

SH7268 群、SH7269 群

用户手册 硬件篇

瑞萨 32 位 RISC 单片机

SuperH™ RISC engine 族 / SH7260 系列

SH7268 R5S7268

SH7269 R5S7269

本资料所记载的内容，均为本资料发行时的信息，瑞萨电子对于本资料所记载的产品或者规格可能会作改动，恕不另行通知。
请通过瑞萨电子的主页确认发布的最新信息。

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.
 - "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.
 - "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.
 - "Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

产品使用时的注意事项

本文对适用于单片机所有产品的“使用时的注意事项”进行说明。有关个别的使用时的注意事项请参照正文。此外，如果在记载上有与本手册的正文有差异之处，请以正文为准。

1. 未使用的引脚的处理

【注意】将未使用的引脚按照正文的“未使用引脚的处理”进行处理。

CMOS产品的输入引脚的阻抗一般为高阻抗。如果在开路的状态下运行未使用的引脚，由于感应现象，外加LSI周围的噪声，在LSI内部产生穿透电流，有可能被误认为是输入信号而引起误动作。未使用的引脚，请按照正文的“未使用引脚的处理”中的指示进行处理。

2. 通电时的处理

【注意】通电时产品处于不定状态。

通电时，LSI内部电路处于不确定状态，寄存器的设定和各引脚的状态不定。通过外部复位引脚对产品进行复位时，从通电到复位有效之前的期间，不能保证引脚的状态。

同样，使用内部上电复位功能对产品进行复位时，从通电到达到复位产生的一定电压的期间，不能保证引脚的状态。

3. 禁止存取保留地址（保留区）

【注意】禁止存取保留地址（保留区）

在地址区域中，有被分配将来用作功能扩展的保留地址（保留区）。因为无法保证存取这些地址时的运行，所以不能对保留地址（保留区）进行存取。

4. 关于时钟

【注意】复位时，请在时钟稳定后解除复位。

在程序运行中切换时钟时，请在要切换成的时钟稳定之后进行。复位时，在通过使用外部振荡器（或者外部振荡电路）的时钟开始运行的系统中，必须在时钟充分稳定后解除复位。另外，在程序运行中，切换成使用外部振荡器（或者外部振荡电路）的时钟时，在要切换成的时钟充分稳定后再进行切换。

5. 关于产品间的差异

【注意】在变更不同型号的产品时，请对每一个产品型号进行系统评价测试。

即使是同一个群的单片机，如果产品型号不同，由于内部ROM、版本模式等不同，在电特性范围内有时特性值、动作容限、噪声耐量、噪声辐射量等不同。因此，在变更不认同型号的产品时，请对每一个型号的产品进行系统评价测试。

本手册的构成

本手册由以下内容构成：

1. 有关产品的一般注意事项
2. 本手册的构成
3. 前言
4. 目录
5. 概要
6. 各功能模块的说明
 - CPU 和系统控制
 - 内部外围模块各模块功能说明的构成因各模块而不同。一般由①特点、②输入 / 输出引脚、③寄存器说明、④运行说明、⑤使用注意事项等章节构成。

设计采用本 LSI 的应用系统时，请在充分确认了注意事项的基础上进行设计。
请务必阅读各章节中有关说明的注意事项和各章节最后的使用时的注意事项（使用时的注意事项根据需要记载）。

7. 寄存器一览表
8. 电特性
9. 引脚状态和处理方法
10. 附录
11. 本版中修改或者追加的部分（仅适用于修订版）

修订履历汇总了对前版内容的修改和追加的主要部分。
记载的内容并非全部修订的内容，请在本手册的正文中确认有关详细内容。

12. 索引

前言

本 LSI 是以瑞萨独创的 RISC 方式 CPU 为核心并且集聚了系统构成上所需外围功能的 RISC 单片机。

对象 本手册以“设计采用本 LSI 的应用系统”的用户为对象。

使用本手册的读者必须具备有关电路，逻辑电路以及单片机的基础知识。

目的 本手册是以“能让用户理解本 LSI 的硬件功能和电特性”为目的。

阅读方法

- 希望了解全部功能时
→ 请按照目录的顺序阅读
本手册按 CPU、系统控制功能、外围功能、电特性等分类依次构成。
- 希望了解 CPU 功能的详细内容时
→ 请参考《SH-2A、SH2A-FPU 用户手册 软件篇》。
- 若知道寄存器名而希望了解详细功能时
→ 在本手册的后面附有“索引”。请从索引搜索页号。
有关地址、位的内容和初始化，汇总在“51. 寄存器一览表”。

2 范例

本手册中使用的寄存器名、位名、数字和符号的表示范例说明如下

(1) 寄存器的表示

相同或类似的功能存在于多个通道时，使用“寄存器名称_通道号”的表示方法。

(例) CMCSR_0

(2) 位的表示

本文中所述位名以左侧为高位、右侧为低位的顺序表示。

(例) CKS1、CKS0

(3) 数字的表示

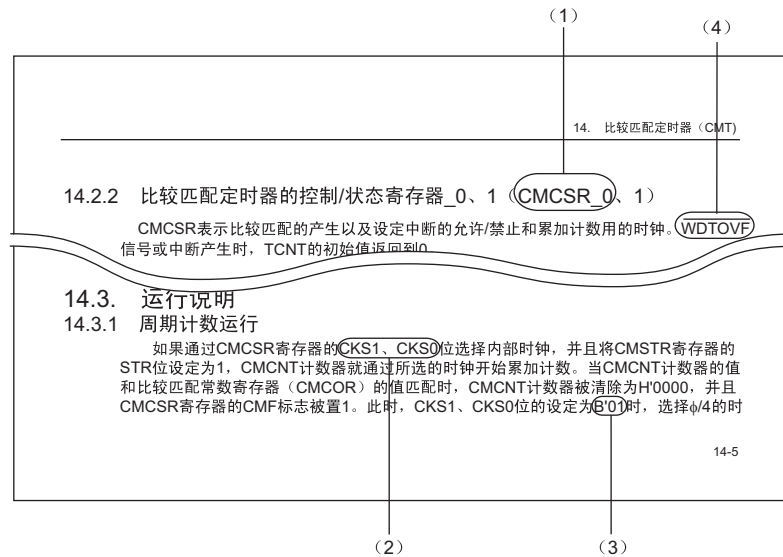
2进制数为B'xxxx或者xxxx，16进制数为H'xxxx，10进制数为xxxx

(例) B'11或者11、H'EFA0、1234

(4) 符号的表示

对低电平有效的信号加上划线。

(例) WDTOVF



【注】 上述图中的位名和文章是举例，与本手册内容无关。

3 位图和位表的表示

各寄存器的说明中，有表示位排列的位图和说明位设定内容的位表。

(1) 位

表示位号码。
32位寄存器按31到0的顺序表示，16位寄存器按15到0的顺序表示。

(2) 位名

表示寄存器的位名或者字段名。
在字段需要明示位的位数时，追加如同ASID[3:0]一样的位数表示法。
保留位用“-”表示。
但也有不描述位名而表示为空白的情况，例如时间计数器等。

(3) 初始值

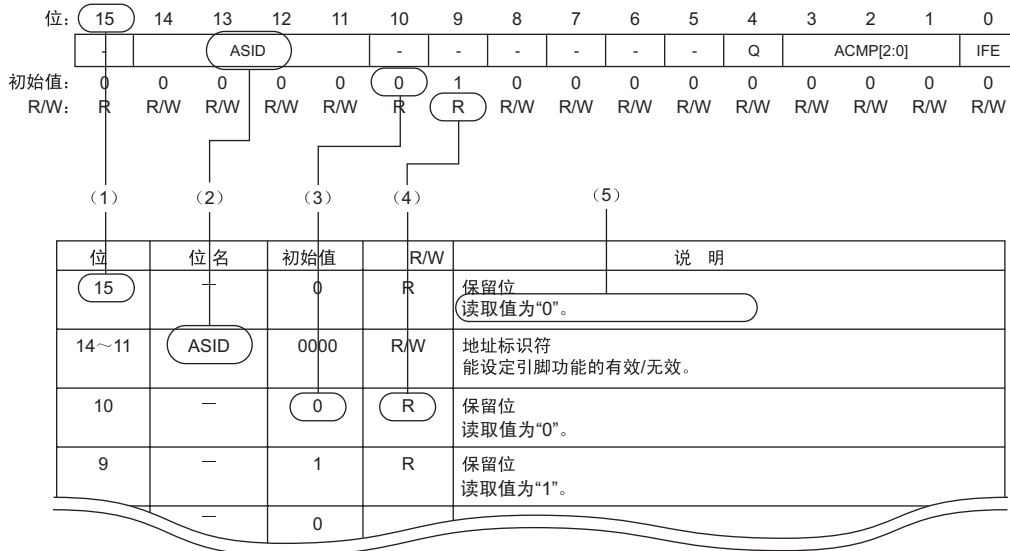
各位上电复位后的值作为初始值表示。
0 : 表示初始值为0。
1 : 表示初始值为1。
- : 表示初始值不定。

(4) R/W

表示各位可读出、写入或者不能读出、写入。
使用的表示法说明如下。
R/W : 可读出以及写入的位或者字段。
R(W) : 可读出以及写入的位或者字段。
但仅是为了清除标志的写入才能进行。
R : 可读出位或者字段。
但保留位全部表示为“R”。需要写入时，
请写入位表指定的值或者初始值。
W : 可写入的位或者字段。
但读出的值是位表记载以外的情况时，不被保证。

(5) 说明

对通过设定位实现的功能进行说明。



【注】 上述图中的位名和文章是举例，与本手册内容无关。

目 录

1. 概要	1
1.1 SH7268/7269 的特点	1
1.2 产品一览表	9
1.3 框图	10
1.4 引脚排列图	11
1.5 引脚功能	14
1.6 引脚一览表	21
2. CPU	41
2.1 寄存器结构	41
2.1.1 通用寄存器	41
2.1.2 控制寄存器	41
2.1.3 系统寄存器	43
2.1.4 寄存器组	43
2.1.5 寄存器的初始值	43
2.2 数据格式	44
2.2.1 寄存器的数据格式	44
2.2.2 存储器的数据格式	44
2.2.3 立即数的数据格式	44
2.3 指令的特点	45
2.3.1 RISC 方式	45
2.3.2 寻址方式	48
2.3.3 指令格式	51
2.4 指令系统	53
2.4.1 指令系统的分类	53
2.4.2 数据传送指令	57
2.4.3 算术运算指令	60
2.4.4 逻辑运算指令	62
2.4.5 移位指令	63
2.4.6 转移指令	64
2.4.7 系统控制指令	65
2.4.8 浮点运算指令	66
2.4.9 FPU 相关的 CPU 指令	68
2.4.10 位操作指令	68
2.5 处理状态	69
3. 浮点单元 (FPU)	70
3.1 特点	70
3.2 数据格式	70
3.2.1 浮点格式	70
3.2.2 非数 (NaN)	72
3.2.3 非规格化数	72
3.3 寄存器的说明	73
3.3.1 浮点寄存器	73
3.3.2 浮点状态 / 控制寄存器 (FPSCR)	74
3.3.3 浮点通信寄存器 (FPUL)	75
3.4 舍入	75
3.5 FPU 异常	76
3.5.1 FPU 异常源	76

3.5.2	FPU 异常处理	76
4.	引导模式	77
4.1	特点	77
4.2	引导模式和引脚功能设定	77
4.3	运行说明	78
4.3.1	引导模式 0 和引导模式 1	78
4.3.2	引导模式 2	78
4.3.3	引导模式 3	80
4.3.4	引导模式 4	81
4.3.5	引导模式 5	82
4.4	注意事项	82
4.4.1	引导相关引脚	82
5.	时钟脉冲振荡器	83
5.1	特点	83
5.2	输入 / 输出引脚	85
5.3	时钟模式	86
5.4	寄存器说明	87
5.4.1	频率控制寄存器 (FRQCR)	87
5.5	频率的更改方法	88
5.5.1	分频比的更改	88
5.6	时钟引脚的使用方法	89
5.6.1	输入外部时钟的情况	89
5.6.2	使用晶体谐振器的情况	89
5.6.3	不使用的情况	90
5.7	振荡稳定时间	90
5.7.1	内部晶体振荡器的振荡稳定时间	90
5.7.2	PLL 电路的振荡稳定时间	90
5.8	设计电路板时的注意事项	90
5.8.1	使用 PLL 振荡电路时的注意事项	90
5.9	SSCG 规格的调制率和调制频率的定义	90
6.	异常处理	91
6.1	概要	91
6.1.1	异常处理的种类和优先级	91
6.1.2	异常处理的运行	92
6.1.3	异常处理向量表	93
6.2	复位	94
6.2.1	输入 / 输出引脚	94
6.2.2	复位的种类	94
6.2.3	上电复位	95
6.2.4	手动复位	95
6.3	地址错误	96
6.3.1	地址错误发生源	96
6.3.2	地址错误的异常处理	96
6.4	寄存器组错误	97
6.4.1	寄存器组的错误发生源	97
6.4.2	寄存器组的错误异常处理	97
6.5	中断	98
6.5.1	中断源	98
6.5.2	中断优先级	98

6.5.3	中断异常处理	98
6.6	由指令引起的异常	99
6.6.1	由指令引起的异常种类	99
6.6.2	陷阱指令	99
6.6.3	槽非法指令	99
6.6.4	一般非法指令	100
6.6.5	整数除法运算异常	100
6.6.6	FPU 异常	100
6.7	不接受异常处理的情况	101
6.8	异常处理后的堆栈状态	101
6.9	使用时的注意事项	102
6.9.1	堆栈指针 (SP) 的值	102
6.9.2	向量基址寄存器 (VBR) 的值	102
6.9.3	在地址错误异常处理的堆栈存取中发生的地址错误	102
6.9.4	通过改变中断屏蔽位进行的中断控制	102
6.9.5	异常处理前的注意事项	102
7.	中断控制器	104
7.1	特点	104
7.2	输入 / 输出引脚	105
7.3	寄存器说明	106
7.3.1	中断优先级设定寄存器 01、02、05 ~ 26 (IPR01、IPR02、IPR05 ~ IPR26)	107
7.3.2	中断控制寄存器 0 (ICR0)	109
7.3.3	中断控制寄存器 1 (ICR1)	110
7.3.4	中断控制寄存器 2 (ICR2)	111
7.3.5	IRQ 中断请求寄存器 (IRQRR)	112
7.3.6	PINT 中断允许寄存器 (PINTER)	113
7.3.7	PINT 中断请求寄存器 (PIRR)	113
7.3.8	存储体控制寄存器 (IBCR)	114
7.3.9	存储体号寄存器 (IBNR)	115
7.4	中断源	116
7.4.1	NMI 中断	116
7.4.2	用户断点中断	116
7.4.3	用户调试接口中断	116
7.4.4	IRQ 中断	117
7.4.5	PINT 中断	117
7.4.6	内部外围模块中断	118
7.5	中断异常处理向量表和优先级	119
7.6	运行说明	125
7.6.1	中断运行的流程	125
7.6.2	中断异常处理结束后的堆栈状态	126
7.7	中断响应时间	127
7.8	寄存器组	132
7.8.1	存储体的对象寄存器和输入 / 输出方式	132
7.8.2	存储体的保存和恢复	133
7.8.3	保存到全部存储体的状态下的保存和恢复	134
7.8.4	寄存器组异常	134
7.8.5	寄存器组的错误异常处理	134
7.9	通过中断请求信号进行的数据传送	135
7.9.1	中断请求信号作为 CPU 中断源而不作为直接存储器存取控制器启动源的情况	135
7.9.2	中断请求信号作为直接存储器存取控制器启动源而不作为 CPU 中断源的情况	135
7.10	使用时的注意事项	136

7.10.1	中断源的清除时序	136
8.	用户断点控制器	137
8.1	特点	137
8.2	输入 / 输出引脚	139
8.3	寄存器说明	139
8.3.1	断点地址寄存器 (BAR)	140
8.3.2	断点地址屏蔽寄存器 (BAMR)	140
8.3.3	断点数据寄存器 (BDR)	141
8.3.4	断点数据屏蔽寄存器 (BDMR)	141
8.3.5	断点总线周期寄存器 (BBR)	142
8.3.6	断点控制寄存器 (BRCCR)	143
8.4	运行说明	145
8.4.1	用户断点运行的流程	145
8.4.2	取指令周期的断点	145
8.4.3	数据存取周期的断点	146
8.4.4	被保存的程序计数器值	146
8.4.5	使用例子	147
8.5	使用时的注意事项	149
9.	高速缓存	150
9.1	特点	150
9.1.1	高速缓存的结构	150
9.2	寄存器说明	151
9.2.1	高速缓存控制寄存器 1 (CCR1)	152
9.2.2	高速缓存控制寄存器 2 (CCR2)	153
9.3	运行说明	155
9.3.1	高速缓存的搜索	155
9.3.2	读操作	156
9.3.3	预取操作 (只限于操作数高速缓存)	156
9.3.4	写操作 (只限于操作数高速缓存)	156
9.3.5	回写缓冲器 (只限于操作数高速缓存)	157
9.3.6	高速缓存和外部存储器 / 大容量内部 RAM 的相关性	157
9.4	存储器映射高速缓存的结构	158
9.4.1	地址阵列	158
9.4.2	数据阵列	158
9.4.3	使用例子	160
9.4.4	注意事项	160
10.	总线状态控制器	161
10.1	特点	161
10.2	输入 / 输出引脚	163
10.3	区域概要	164
10.3.1	地址映像	164
10.3.2	引导模式、各区域的数据总线宽度、字节序和相关引脚的设定	165
10.4	寄存器说明	166
10.4.1	公共控制寄存器 (CMNCR)	167
10.4.2	CSn 空间总线控制寄存器 (CSnBCR) (n=0 ~ 5)	169
10.4.3	CSn 空间等待控制寄存器 (CSnWCR) (n=0 ~ 5)	172
10.4.4	SDRAM 控制寄存器 (SDCR)	191
10.4.5	刷新定时器的控制 / 状态寄存器 (RTCSR)	193
10.4.6	刷新定时器的计数器 (RTCNT)	194

10.4.7	刷新时间常数寄存器 (RTCOR)	195
10.5	运行说明	196
10.5.1	字节序 / 存取长度和数据调整	196
10.5.2	正规空间接口	201
10.5.3	存取等待的控制	206
10.5.4	CSn 有效期间的扩展	208
10.5.5	MPX-I/O 接口	209
10.5.6	SDRAM 接口	211
10.5.7	突发 ROM (异步) 接口	235
10.5.8	带字节选择的 SRAM 接口	237
10.5.9	PCMCIA 接口	241
10.5.10	突发 ROM (时钟同步) 接口	245
10.5.11	存取周期之间的空闲	246
10.5.12	总线仲裁	250
10.5.13	其他	251
11.	直接存储器存取控制器	253
11.1	特点	253
11.2	输入 / 输出引脚	255
11.3	寄存器说明	255
11.3.1	DMA 源地址寄存器 (SAR)	259
11.3.2	DMA 目标地址寄存器 (DAR)	259
11.3.3	DMA 传送计数寄存器 (DMATCR)	259
11.3.4	DMA 通道控制寄存器 (CHCR)	260
11.3.5	DMA 重加载源地址寄存器 (RSAR)	265
11.3.6	DMA 重加载目标地址寄存器 (RDAR)	265
11.3.7	DMA 重加载传送计数寄存器 (RDMATCR)	266
11.3.8	DMA 操作寄存器 (DMAOR)	266
11.3.9	DMA 扩展资源选择器 0 ~ 7 (DMARS0 ~ DMARS7)	268
11.4	运行说明	272
11.4.1	传送流程	272
11.4.2	DMA 传送请求	274
11.4.3	通道优先级	278
11.4.4	DMA 的传送种类	278
11.4.5	总线周期的状态数和 DREQ 引脚的采样时序	284
11.5	使用时的注意事项	287
11.5.1	DACK 输出和 TEND 输出的时序	287
12.	多功能定时器脉冲单元 2	288
12.1	特点	288
12.2	输入 / 输出引脚	292
12.3	寄存器说明	293
12.3.1	定时器的控制寄存器 (TCR)	295
12.3.2	定时器的模式寄存器 (TMDR)	298
12.3.3	定时器的 I/O 控制寄存器 (TIOR)	300
12.3.4	定时器的中断允许寄存器 (TIER)	309
12.3.5	定时器的状态寄存器 (TSR)	311
12.3.6	定时器的缓冲运行传送模式寄存器 (TBTM)	314
12.3.7	定时器的输入捕捉控制寄存器 (TICCR)	315
12.3.8	定时器的 A/D 转换开始请求控制寄存器 (TADCR)	316
12.3.9	定时器的 A/D 转换开始请求周期设定寄存器 (TADCORA/B_4)	318
12.3.10	定时器的 A/D 转换开始请求周期设定缓冲寄存器 (TADCOBRA/B_4)	318

12.3.11	定时器的计数器 (TCNT)	318
12.3.12	定时器的通用寄存器 (TGR)	319
12.3.13	定时器的启动寄存器 (TSTR)	319
12.3.14	定时器的同步寄存器 (TSYR)	320
12.3.15	定时器的读写允许寄存器 (TRWER)	321
12.3.16	定时器的输出主控允许寄存器 (TOER)	322
12.3.17	定时器的输出控制寄存器 1 (TOCR1)	323
12.3.18	定时器的输出控制寄存器 2 (TOCR2)	325
12.3.19	定时器的输出电平缓冲寄存器 (TOLBR)	327
12.3.20	定时器的门控寄存器 (TGCR)	328
12.3.21	定时器的副计数器 (TCNTS)	329
12.3.22	定时器的空载时间数据寄存器 (TDDR)	329
12.3.23	定时器的周期数据寄存器 (TCDR)	329
12.3.24	定时器的周期缓冲寄存器 (TCBR)	330
12.3.25	定时器的中断减少设定寄存器 (TITCR)	330
12.3.26	定时器的中断减少次数计数器 (TITCNT)	331
12.3.27	定时器的缓冲传送设定寄存器 (TBTER)	332
12.3.28	定时器的空载时间允许寄存器 (TDER)	333
12.3.29	定时器的波形控制寄存器 (TWCR)	334
12.3.30	和总线主控器的接口	334
12.4	运行说明	335
12.4.1	基本运行	335
12.4.2	同步运行	340
12.4.3	缓冲运行	341
12.4.4	级联运行	345
12.4.5	PWM 模式	348
12.4.6	相位计数模式	353
12.4.7	复位同步 PWM 模式	359
12.4.8	互补 PWM 模式	361
12.4.9	A/D 转换开始请求的延迟功能	391
12.4.10	在互补 PWM 的“波峰/波谷”进行的 TCNT 捕捉运行	394
12.5	中断源	395
12.5.1	中断源和优先级	395
12.5.2	直接存储器存取控制器的启动	397
12.5.3	A/D 转换器的启动	397
12.6	运行时序	398
12.6.1	输入/输出时序	398
12.6.2	中断信号的时序	404
12.7	使用时的注意事项	407
12.7.1	模块待机模式的设定	407
12.7.2	输入时钟的限制事项	407
12.7.3	设定周期时的注意事项	407
12.7.4	TCNT 的写和清除的竞争	408
12.7.5	TCNT 的写和递增计数的竞争	408
12.7.6	TGR 的写和比较匹配的竞争	409
12.7.7	缓冲寄存器的写和比较匹配的竞争	409
12.7.8	缓冲寄存器的写和 TCNT 清除的竞争	410
12.7.9	TGR 的读和输入捕捉的竞争	410
12.7.10	TGR 的写和输入捕捉的竞争	411
12.7.11	缓冲寄存器的写和输入捕捉的竞争	411
12.7.12	级联中的 TCNT_2 的写和上溢/下溢的竞争	412
12.7.13	互补 PWM 模式停止时的计数器值	413

12.7.14	互补 PWM 模式中的缓冲运行的设定	413
12.7.15	复位同步 PWM 模式中的缓冲运行和比较匹配标志	413
12.7.16	复位同步 PWM 模式的上溢标志	414
12.7.17	上溢 / 下溢和计数器清除的竞争	415
12.7.18	TCNT 的写和上溢 / 下溢的竞争	415
12.7.19	从正常运行或者 PWM 模式 1 转移到复位同步 PWM 模式时的注意事项	416
12.7.20	互补 PWM 模式和复位同步 PWM 模式的输出电平	416
12.7.21	模块待机时的中断	416
12.7.22	级联中的 TCNT_1 和 TCNT_2 的同时输入捕捉	416
12.7.23	在互补 PWM 模式进行同步计数器清除时的输出波形控制的注意事项	416
12.8	多功能定时器脉冲单元 2 的输出引脚的初始化方法	418
12.8.1	运行模式	418
12.8.2	开始复位时的运行	418
12.8.3	因运行过程中的异常等而重新设定时的运行	418
12.8.4	因运行过程中的异常等而对引脚进行初始化的步骤以及模式转移的概要	419
13.	比较匹配定时器	441
13.1	特点	441
13.2	寄存器说明	442
13.2.1	比较匹配定时器的启动寄存器 (CMSTR)	442
13.2.2	比较匹配定时器的控制 / 状态寄存器 (CMCSR)	443
13.2.3	比较匹配计数器 (CMCNT)	443
13.2.4	比较匹配常数寄存器 (CMCOR)	444
13.3	运行说明	444
13.3.1	周期计数运行	444
13.3.2	CMCNT 的计数时序	444
13.4	中断	445
13.4.1	中断源和 DMA 传送请求	445
13.4.2	比较匹配标志的置位时序	445
13.4.3	比较匹配标志的清除时序	445
13.5	使用时的注意事项	446
13.5.1	CMCNT 的写操作和比较匹配的竞争	446
13.5.2	CMCNT 的字写和递增计数的竞争	446
13.5.3	CMCNT 的字节写和递增计数的竞争	447
13.5.4	CMCNT 和 CMCOR 的比较匹配	447
14.	看门狗定时器	448
14.1	特点	448
14.2	输入 / 输出引脚	449
14.3	寄存器说明	449
14.3.1	看门狗定时器的计数器 (WTCNT)	449
14.3.2	看门狗定时器的控制 / 状态寄存器 (WTCSR)	450
14.3.3	看门狗定时器的复位控制 / 状态寄存器 (WRCSR)	451
14.3.4	存取寄存器时的注意事项	452
14.4	使用方法	453
14.4.1	软件待机模式的解除步骤	453
14.4.2	看门狗定时器模式的用法	453
14.4.3	间隔定时器模式的用法	454
14.5	使用时的注意事项	454
14.5.1	定时器的误差	454
14.5.2	禁止将 H'FF 设定为 WTCNT 的初始值	454
14.5.3	间隔定时器上溢标志	454

14.5.4	通过 <u>WDTOVF</u> 信号进行的系统复位	455
14.5.5	看门狗定时器模式的手动复位	455
15.	实时时钟	456
15.1	特点	456
15.2	输入 / 输出引脚	458
15.3	寄存器说明	458
15.3.1	64Hz 计数器 (R64CNT)	459
15.3.2	秒计数器 (RSECCNT)	459
15.3.3	分钟计数器 (RMINCNT)	460
15.3.4	小时计数器 (RHRCNT)	460
15.3.5	星期计数器 (RWKCNT)	461
15.3.6	日计数器 (RDAYCNT)	461
15.3.7	月计数器 (RMONCNT)	462
15.3.8	年计数器 (RYRCNT)	462
15.3.9	秒闹钟寄存器 (RSECAR)	463
15.3.10	分钟闹钟寄存器 (RMINAR)	463
15.3.11	小时闹钟寄存器 (RHRAR)	464
15.3.12	星期闹钟寄存器 (RWKAR)	464
15.3.13	日闹钟寄存器 (RDAYAR)	465
15.3.14	月闹钟寄存器 (RMONAR)	465
15.3.15	年闹钟寄存器 (RYRAR)	466
15.3.16	控制寄存器 1 (RCR1)	467
15.3.17	控制寄存器 2 (RCR2)	468
15.3.18	控制寄存器 3 (RCR3)	469
15.3.19	控制寄存器 5 (RCR5)	469
15.3.20	频率寄存器 H/L (RFRH/L)	470
15.4	运行说明	471
15.4.1	接通电源后的寄存器的初始设定	471
15.4.2	时间的设定步骤	471
15.4.3	时间的读取步骤	472
15.4.4	闹钟功能	473
15.5	使用时的注意事项	474
15.5.1	有关计数过程中的寄存器的写操作	474
15.5.2	有关实时时钟周期中断的使用	474
15.5.3	有关寄存器设定后的待机转移	474
15.5.4	读写寄存器时的注意事项	474
16.	FIFO 内置型串行通信接口	475
16.1	特点	475
16.2	输入 / 输出引脚	476
16.3	寄存器说明	477
16.3.1	接收移位寄存器 (SCRSR)	479
16.3.2	接收 FIFO 数据寄存器 (SCFRDR)	479
16.3.3	发送移位寄存器 (SCTSR)	480
16.3.4	发送 FIFO 数据寄存器 (SCFTDR)	480
16.3.5	串行模式寄存器 (SCSMR)	480
16.3.6	串行控制寄存器 (SCSCR)	482
16.3.7	串行状态寄存器 (SCFSR)	484
16.3.8	位速率寄存器 (SCBRR)	487
16.3.9	FIFO 控制寄存器 (SCFCR)	491
16.3.10	FIFO 数据计数置位寄存器 (SCFDR)	493

16.3.11	串行端口寄存器 (SCSPTR)	493
16.3.12	线路状态寄存器 (SCLSR)	495
16.3.13	串行扩展模式寄存器 (SCEMR)	496
16.4	运行说明	497
16.4.1	概要	497
16.4.2	异步模式中的运行	498
16.4.3	时钟同步模式中的运行	506
16.5	中断	513
16.6	使用时的注意事项	514
16.6.1	SCFTDR 的写操作和 TDFE 标志	514
16.6.2	SCFRDR 的读操作和 RDF 标志	514
16.6.3	使用直接存储器存取控制器时的注意事项	514
16.6.4	中止的检测和处理	514
16.6.5	中止的发送	514
16.6.6	异步模式的接收数据采样时序和接收容限	515
16.6.7	异步基本时钟的选择	515
17.	瑞萨串行外围接口	516
17.1	特点	516
17.2	输入 / 输出引脚	518
17.3	寄存器说明	519
17.3.1	控制寄存器 (SPCR)	520
17.3.2	从属选择极性寄存器 (SSLP)	521
17.3.3	引脚控制寄存器 (SPPCR)	522
17.3.4	状态寄存器 (SPSR)	523
17.3.5	数据寄存器 (SPDR)	525
17.3.6	顺序控制寄存器 (SPSCR)	526
17.3.7	顺序状态寄存器 (SPSSR)	526
17.3.8	位速率寄存器 (SPBR)	527
17.3.9	数据控制寄存器 (SPDCR)	528
17.3.10	时钟延迟寄存器 (SPCKD)	529
17.3.11	从属选择无效延迟寄存器 (SSLND)	530
17.3.12	下次存取延迟寄存器 (SPND)	531
17.3.13	命令寄存器 (SPCMD)	532
17.3.14	缓冲控制寄存器 (SPBFCR)	534
17.3.15	缓冲器数据计数置位寄存器 (SPBFDR)	535
17.4	运行说明	536
17.4.1	运行概要	536
17.4.2	引脚的控制	537
17.4.3	系统结构例子	538
17.4.4	传送格式	540
17.4.5	数据格式	542
17.4.6	错误检测	548
17.4.7	初始化	550
17.4.8	SPI 运行	551
17.4.9	错误处理	561
17.4.10	环回模式	562
17.4.11	中断源	562
18.	瑞萨四路串行外围接口	563
18.1	特点	563
18.2	输入 / 输出引脚	564

18.3	寄存器说明	564
18.3.1	控制寄存器 (SPCR)	566
18.3.2	从属选择极性寄存器 (SSLP)	566
18.3.3	引脚控制寄存器 (SPPCR)	567
18.3.4	状态寄存器 (SPSR)	568
18.3.5	数据寄存器 (SPDR)	569
18.3.6	顺序控制寄存器 (SPSCR)	569
18.3.7	顺序状态寄存器 (SPSSR)	570
18.3.8	位速率寄存器 (SPBR)	571
18.3.9	数据控制寄存器 (SPDCR)	572
18.3.10	时钟延迟寄存器 (SPCKD)	572
18.3.11	从属选择无效延迟寄存器 (SSLND)	573
18.3.12	下次存取延迟寄存器 (SPND)	574
18.3.13	命令寄存器 n (SPCMDn) (n=0、1、2、3)	575
18.3.14	缓冲控制寄存器 (SPBFCR)	578
18.3.15	缓冲数据计数置位寄存器 (SPBDCR)	579
18.3.16	传送数据长度倍数设定寄存器 n (SPBMULn) (n=0、1、2、3)	579
18.4	运行说明	580
18.4.1	运行概要	580
18.4.2	引脚的控制	581
18.4.3	传送格式	582
18.4.4	传送数据	585
18.4.5	异常运行	589
18.4.6	初始化	589
18.4.7	SPI 运行	590
18.4.8	中断源	600
18.4.9	环回模式	600
19.	SPI 多 I/O 总线控制器	601
19.1	特点	601
19.2	框图	602
19.3	输入 / 输出引脚	602
19.4	寄存器说明	603
19.4.1	公共控制寄存器 (CMNCR)	604
19.4.2	SSL 延迟寄存器 (SSLDR)	607
19.4.3	位速率设定寄存器 (SPBCR)	608
19.4.4	数据读控制寄存器 (DRCR)	609
19.4.5	数据读命令设定寄存器 (DRCMR)	610
19.4.6	数据读扩展地址设定寄存器 (DREAR)	611
19.4.7	数据读选项设定寄存器 (DROPR)	612
19.4.8	数据读允许设定寄存器 (DRENr)	613
19.4.9	SPI 模式控制寄存器 (SMCR)	615
19.4.10	SPI 模式命令设定寄存器 (SMCMR)	616
19.4.11	SPI 模式地址设定寄存器 (SMADR)	616
19.4.12	SPI 模式选项设定寄存器 (SMOPR)	617
19.4.13	SPI 模式允许设定寄存器 (SMENR)	618
19.4.14	SPI 模式读数据寄存器 0 (SMRDR0)	620
19.4.15	SPI 模式读数据寄存器 1 (SMRDR1)	621
19.4.16	SPI 模式写数据寄存器 0 (SMWDR0)	621
19.4.17	SPI 模式写数据寄存器 1 (SMWDR1)	622
19.4.18	公共状态寄存器 (CMNSR)	622
19.5	运行说明	623

19.5.1	系统结构	623
19.5.2	地址映像	624
19.5.3	串行闪存 32 位地址	624
19.5.4	数据调整	625
19.5.5	运行模式	625
19.5.6	外部地址空间读模式	625
19.5.7	读高速缓存	630
19.5.8	SPI 运行模式	631
19.5.9	传送格式	634
19.5.10	数据格式	635
19.5.11	数据引脚控制	639
19.5.12	SPBSSL 引脚控制	640
19.5.13	标志	640
20.	I ² C 总线接口 3	641
20.1	特点	641
20.2	输入 / 输出引脚	643
20.3	寄存器说明	644
20.3.1	I ² C 总线控制寄存器 1 (ICCR1)	645
20.3.2	I ² C 总线控制寄存器 2 (ICCR2)	647
20.3.3	I ² C 总线模式寄存器 (ICMR)	648
20.3.4	I ² C 总线中断允许寄存器 (ICIER)	649
20.3.5	I ² C 总线状态寄存器 (ICSR)	650
20.3.6	从属地址寄存器 (SAR)	651
20.3.7	I ² C 总线发送数据寄存器 (ICDRT)	652
20.3.8	I ² C 总线接收数据寄存器 (ICDRR)	652
20.3.9	I ² C 总线移位寄存器 (ICDRS)	652
20.3.10	NF2CYC 寄存器 (NF2CYC)	653
20.4	运行说明	654
20.4.1	I ² C 总线格式	654
20.4.2	主控发送	655
20.4.3	主控接收	656
20.4.4	从属发送	658
20.4.5	从属接收	660
20.4.6	时钟同步串行格式	661
20.4.7	噪声消除电路	663
20.4.8	使用例子	664
20.5	中断请求	667
20.6	位同步电路	668
20.7	使用时的注意事项	669
20.7.1	用于多主控时的注意事项	669
20.7.2	主控接收模式时的注意事项	669
20.7.3	在主控接收模式中设定 ACKBT 时的注意事项	669
20.7.4	仲裁失败时的 MST 位和 TRN 位的状态的注意事项	669
20.7.5	I ² C 总线接口模式的主控接收模式中的注意事项	670
20.7.6	有关 IICRST 位和 BBSY 位的注意事项	670
20.7.7	主控发送模式并且将 ACKE 位设定为 “1” 时的停止条件的发行注意事项	670
21.	串行音频接口	671
21.1	特点	671
21.2	输入 / 输出引脚	673
21.3	寄存器说明	673

21.3.1	控制寄存器 (SSICR)	675
21.3.2	状态寄存器 (SSISR)	680
21.3.3	发送数据寄存器 (SSITDR)	682
21.3.4	接收数据寄存器 (SSIRDR)	682
21.3.5	FIFO 控制寄存器 (SSIFCR)	683
21.3.6	FIFO 状态寄存器 (SSIFSR)	685
21.3.7	发送 FIFO 数据寄存器 (SSIFTDR)	686
21.3.8	接收 FIFO 数据寄存器 (SSIFRDR)	687
21.3.9	TDM 模式寄存器 (SSITDMR)	687
21.4	运行说明	688
21.4.1	总线格式	688
21.4.2	非压缩模式	688
21.4.3	TDM 模式	696
21.4.4	WS 继续模式	697
21.4.5	运行模式	698
21.4.6	发送运行	698
21.4.7	接收运行	701
21.4.8	串行位时钟控制	703
21.5	使用时的注意事项	703
21.5.1	在 DMA 运行过程中发生下溢或者上溢时的限制事项	703
21.5.2	进行主控收发器到主控接收器的模式切换时的注意事项	703
21.5.3	TDM 模式和 WS 继续模式的限制事项	703
22.	带 FIFO 的时钟同步串行 I/O	704
22.1	特点	704
22.2	输入 / 输出引脚	705
22.3	寄存器说明	705
22.3.1	模式寄存器 (SIMDR)	706
22.3.2	控制寄存器 (SICTR)	708
22.3.3	发送数据寄存器 (SITDR)	710
22.3.4	接收数据寄存器 (SIRDR)	710
22.3.5	状态寄存器 (SISTR)	711
22.3.6	中断允许寄存器 (SIER)	713
22.3.7	FIFO 控制寄存器 (SIFCTR)	715
22.3.8	时钟选择寄存器 (SISCR)	716
22.3.9	发送数据分配寄存器 (SITDAR)	717
22.3.10	接收数据分配寄存器 (SIRDAR)	718
22.4	运行说明	719
22.4.1	串行时钟	719
22.4.2	串行时序	720
22.4.3	传送的数据格式	721
22.4.4	传送数据的寄存器分配	722
22.4.5	FIFO	723
22.4.6	发送 / 接收步骤	724
22.4.7	中断	728
22.4.8	发送 / 接收时序	729
23.	控制器局域网	732
23.1	特点	732
23.1.1	特点	732
23.1.2	本单片机的特点	733
23.2	结构	733

23.2.1	框图	733
23.2.2	各块的功能	734
23.2.3	引脚结构	735
23.2.4	存储器映像	736
23.3	信箱	737
23.3.1	信箱结构	737
23.3.2	信息控制字段	740
23.3.3	本地接收滤波器屏蔽 (LAFM)	743
23.3.4	信息数据字段	744
23.3.5	时戳	745
23.3.6	发送触发时间 (TTT) 和时间触发控制	746
23.4	控制寄存器	749
23.4.1	主控控制寄存器 (MCR)	749
23.4.2	通用状态寄存器 (GSR)	753
23.4.3	位配置寄存器 0、1 (BCR0、BCR1)	755
23.4.4	中断请求寄存器 (IRR)	758
23.4.5	中断屏蔽寄存器 (IMR)	763
23.4.6	发送错误计数器 (TEC) / 接收错误计数器 (REC)	763
23.5	信箱寄存器	764
23.5.1	发送等待寄存器 1、0 (TXPR1、TXPR0)	765
23.5.2	发送取消寄存器 1、0 (TXCR1、TXCR0)	767
23.5.3	发送应答寄存器 1、0 (TXACK1、TXACK0)	768
23.5.4	中止应答寄存器 1、0 (ABACK1、ABACK0)	769
23.5.5	数据帧接收结束寄存器 1、0 (RXPR1、RXPR0)	770
23.5.6	远程帧接收结束寄存器 1、0 (RFPR1、RFPR0)	771
23.5.7	信箱中断屏蔽寄存器 1、0 (MBIMR1、MBIMR0)	772
23.5.8	未读信息状态寄存器 1、0 (UMSR1、UMSR0)	773
23.6	定时器的寄存器	774
23.6.1	时间触发控制寄存器 0 (TTCR0)	774
23.6.2	最大周期 /Tx_Enable_Window 寄存器 (CMAX_TEW)	776
23.6.3	基准触发偏移寄存器 (RFTROFF)	777
23.6.4	定时器的状态寄存器 (TSR)	778
23.6.5	周期计数寄存器 (CCR)	780
23.6.6	定时器的计数寄存器 (TCNTR)	781
23.6.7	周期时间寄存器 (CYCTR)	781
23.6.8	基准标记寄存器 (RFMK)	782
23.6.9	定时器的比较匹配寄存器 0 ~ 2 (TCMR0 ~ 2)	782
23.6.10	发送触发时间选择寄存器 (TTTSEL)	784
23.7	运行说明	785
23.7.1	此模块的设定	785
23.7.2	测试模式的设定	788
23.7.3	信息发送顺序	789
23.7.4	信息接收顺序	798
23.7.5	信箱的重新设定	799
23.8	中断源	801
23.9	直接存储器存取控制器接口	802
23.10	CAN 总线接口	803
23.11	引脚端口的设定	803
23.12	使用时的注意事项	805
23.12.1	有关 1 个通道 64 个或者 96 个信箱的端口设定注意事项	805
24.	IEBus™ 控制器	806

24.1	特点	806
24.1.1	IEBus 通信协议	807
24.1.2	传输协议	809
24.1.3	传输数据（数据字段的内容）	813
24.1.4	位格式	816
24.1.5	结构	817
24.2	输入 / 输出引脚	818
24.3	寄存器说明	819
24.3.1	IEBus 控制寄存器（IECTR）	820
24.3.2	IEBus 命令寄存器（IECMR）	821
24.3.3	IEBus 主控控制寄存器（IEMCR）	822
24.3.4	IEBus 本站地址寄存器 1（IEAR1）	823
24.3.5	IEBus 本站地址寄存器 2（IEAR2）	823
24.3.6	IEBus 从属地址设定寄存器 1（IESA1）	824
24.3.7	IEBus 从属地址设定寄存器 2（IESA2）	824
24.3.8	IEBus 发送信息长度寄存器（IETBFL）	824
24.3.9	IEBus 接收主控地址寄存器 1（IEMA1）	825
24.3.10	IEBus 接收主控地址寄存器 2（IEMA2）	825
24.3.11	IEBus 接收控制字段寄存器（IERCTL）	826
24.3.12	IEBus 接收信息长度寄存器（IERBFL）	826
24.3.13	IEBus 锁定地址寄存器 1（IELA1）	826
24.3.14	IEBus 锁定地址寄存器 2（IELA2）	827
24.3.15	IEBus 通用标志寄存器（IEFLG）	827
24.3.16	IEBus 发送状态寄存器（IETSR）	829
24.3.17	IEBus 发送中断允许寄存器（IEIET）	831
24.3.18	IEBus 接收状态寄存器（IERSR）	832
24.3.19	IEBus 接收中断允许寄存器（IEIER）	835
24.3.20	IEBus 时钟选择寄存器（IECKSR）	836
24.3.21	IEBus 发送数据缓冲器 001 ~ 128（IETB001 ~ IETB128）	837
24.3.22	IEBus 接收数据缓冲器 001 ~ 128（IERB001 ~ IERB128）	837
24.4	数据格式	838
24.4.1	发送格式	838
24.4.2	接收格式	839
24.5	软件控制流程	840
24.5.1	初始设定	840
24.5.2	主控发送	841
24.5.3	从属接收	842
24.5.4	主控接收	843
24.5.5	从属发送	844
24.6	运行时序	845
24.6.1	主控发送	845
24.6.2	从属接收	846
24.6.3	主控接收	847
24.6.4	从属发送	848
24.7	中断源	849
24.8	使用时的注意事项	850
24.8.1	在最大传输字节长度内通信没有结束时的注意事项	850
25.	瑞萨 SPDIF 接口	852
25.1	概要	852
25.2	特点	852
25.3	功能框图	853

25.4	输入 / 输出引脚	853
25.5	瑞萨 SPDIF (IEC60958) 帧格式	854
25.6	寄存器结构	855
25.7	寄存器说明	855
25.7.1	控制寄存器 (CTRL)	856
25.7.2	状态寄存器 (STAT)	859
25.7.3	发送模块通道 1 的音频寄存器 (TLCA)	861
25.7.4	发送模块通道 2 的音频寄存器 (TRCA)	862
25.7.5	发送模块 DMA 的音频数据寄存器 (TDAD)	862
25.7.6	发送用户数据寄存器 (TUI)	863
25.7.7	发送模块通道 1 的状态寄存器 (TLCS)	864
25.7.8	发送模块通道 2 的状态寄存器 (TRCS)	865
25.7.9	接收模块通道 1 的音频寄存器 (RLCA)	866
25.7.10	接收模块通道 2 的音频寄存器 (RRCA)	867
25.7.11	接收模块 DMA 的音频数据 (RDAD)	867
25.7.12	接收用户数据寄存器 (RUI)	868
25.7.13	接收模块通道 1 的状态寄存器 (RLCS)	869
25.7.14	接收模块通道 2 的状态寄存器 (RRCS)	870
25.8	功能说明—发送模块	872
25.8.1	发送模块	872
25.8.2	发送模块的初始化	872
25.8.3	发送模块的初始设定	872
25.8.4	发送模块的数据传送	873
25.9	功能说明—接收模块	874
25.9.1	接收模块	874
25.9.2	接收模块的初始化	874
25.9.3	接收模块的数据传送	875
25.10	模块的停止	877
25.10.1	发送模块和接收模块的空闲状态	877
25.11	压缩模式的数据	877
25.12	参考	877
25.13	使用时的注意事项	877
25.13.1	TUIR 的清除	877
25.13.2	音频输入时钟的频率	877
26.	CD-ROM 解码器	878
26.1	特点	878
26.1.1	数据格式	878
26.2	框图	879
26.3	寄存器说明	882
26.3.1	允许控制寄存器 (CROMEN)	884
26.3.2	同步码的同步控制寄存器 (CROMSY0)	885
26.3.3	解码模式控制寄存器 (CROMCTL0)	886
26.3.4	EDC、ECC 校验控制寄存器 (CROMCTL1)	887
26.3.5	解码处理自动停止控制寄存器 (CROMCTL3)	888
26.3.6	解码选项设定控制寄存器 (CROMCTL4)	889
26.3.7	HEAD20 ~ 22 表示控制寄存器 (CROMCTL5)	890
26.3.8	同步码状态寄存器 (CROMST0)	890
26.3.9	ECC 后的标头错误状态寄存器 (CROMST1)	891
26.3.10	ECC 后的副标头错误状态寄存器 (CROMST3)	891
26.3.11	标头 / 副标头数据有效性判定状态寄存器 (CROMST4)	892
26.3.12	模式判定结果和链路扇区检测状态寄存器 (CROMST5)	893

26.3.13	ECC、EDC 错误状态寄存器 (CROMST6)	894
26.3.14	缓冲状态寄存器 (CBUFST0)	894
26.3.15	解码中止源状态寄存器 (CBUFST1)	895
26.3.16	缓冲器上溢状态寄存器 (CBUFST2)	895
26.3.17	ECC 校正前标头部 -MINUTES 数据寄存器 (HEAD00)	896
26.3.18	ECC 校正前标头部 -SECONDS 数据寄存器 (HEAD01)	896
26.3.19	ECC 校正前标头部 -FRAMES (1/75 秒) 数据寄存器 (HEAD02)	896
26.3.20	ECC 校正前标头部 -MODE 数据寄存器 (HEAD03)	896
26.3.21	ECC 校正前副标头部 - 文件号 (BYTE-16) 数据寄存器 (SHEAD00)	897
26.3.22	ECC 校正前副标头部 - 通道号 (BYTE-17) 数据寄存器 (SHEAD01)	897
26.3.23	ECC 校正前副标头部 - 子模式 (BYTE-18) 数据寄存器 (SHEAD02)	897
26.3.24	ECC 校正前副标头部 - 数据类型 (BYTE-19) 数据寄存器 (SHEAD03)	897
26.3.25	ECC 校正前副标头部 - 文件号 (BYTE-20) 数据寄存器 (SHEAD04)	898
26.3.26	ECC 校正前副标头部 - 通道号 (BYTE-21) 数据寄存器 (SHEAD05)	898
26.3.27	ECC 校正前副标头部 - 子模式 (BYTE-22) 数据寄存器 (SHEAD06)	898
26.3.28	ECC 校正前副标头部 - 数据类型 (BYTE-23) 数据寄存器 (SHEAD07)	898
26.3.29	ECC 校正后标头部 -MINUTES 数据寄存器 (HEAD20)	899
26.3.30	ECC 校正后标头部 -SECONDS 数据寄存器 (HEAD21)	899
26.3.31	ECC 校正后标头部 -FRAMES (1/75 秒) 数据寄存器 (HEAD22)	899
26.3.32	ECC 校正后标头部 -MODE 数据寄存器 (HEAD23)	899
26.3.33	ECC 校正后副标头部 - 文件号 (BYTE-16) 数据寄存器 (SHEAD20)	900
26.3.34	ECC 校正后副标头部 - 通道号 (BYTE-17) 数据寄存器 (SHEAD21)	900
26.3.35	ECC 校正后副标头部 - 子模式 (BYTE-18) 数据寄存器 (SHEAD22)	900
26.3.36	ECC 校正后副标头部 - 数据类型 (BYTE-19) 数据寄存器 (SHEAD23)	900
26.3.37	ECC 校正后副标头部 - 文件号 (BYTE-20) 数据寄存器 (SHEAD24)	901
26.3.38	ECC 校正后副标头部 - 通道号 (BYTE-21) 数据寄存器 (SHEAD25)	901
26.3.39	ECC 校正后副标头部 - 子模式 (BYTE-22) 数据寄存器 (SHEAD26)	901
26.3.40	ECC 校正后副标头部 - 数据类型 (BYTE-23) 数据寄存器 (SHEAD27)	901
26.3.41	自动缓冲设定控制寄存器 (CBUFCTL0)	902
26.3.42	自动缓冲起始扇区设定 -MINUTES 控制寄存器 (CBUFCTL1)	903
26.3.43	自动缓冲起始扇区设定 -SECONDS 控制寄存器 (CBUFCTL2)	903
26.3.44	自动缓冲起始扇区设定 -FRAMES 控制寄存器 (CBUFCTL3)	903
26.3.45	ISY 中断源屏蔽控制寄存器 (CROMST0M)	904
26.3.46	CD-ROM 解码器模块复位控制寄存器 (ROMDECRST)	904
26.3.47	CD-ROM 解码器模块复位状态寄存器 (RSTSTAT)	905
26.3.48	串行音频接口数据控制寄存器 (SSI)	906
26.3.49	中断标志寄存器 (INTHOLD)	907
26.3.50	中断源屏蔽控制寄存器 (INHINT)	908
26.3.51	CD-ROM 解码器流数据输入寄存器 (STRMDIN0)	908
26.3.52	CD-ROM 解码器流数据输入寄存器 (STRMDIN2)	909
26.3.53	CD-ROM 解码器流数据输出寄存器 (STRMDOUT0)	909
26.4	运行说明	910
26.4.1	输入流数据字节序的转换功能	910
26.4.2	同步码保护功能	911
26.4.3	纠错	914
26.4.4	自动解码的停止功能	915
26.4.5	缓冲格式	915
26.4.6	目标扇区的缓冲功能	917
26.5	中断源	919
26.5.1	中断和 DMA 传送请求信号	919
26.5.2	状态寄存器的更新时序	920
26.6	使用时的注意事项	920

26.6.1	在进行解码的状态下只停止或者重新开始缓冲时的注意点	920
26.6.2	设定同步码状态寄存器 (CROMST0) 时的注意点	920
26.6.3	链路块的注意点	920
26.6.4	停止和重新开始 CD-DSP 时的注意点	921
26.6.5	清除 IREADY 标志时的注意点	921
26.6.6	传送流数据时的注意点 (1)	921
26.6.7	传送流数据时的注意点 (2)	921
27.	A/D 转换器	922
27.1	特点	922
27.2	输入 / 输出引脚	923
27.3	寄存器说明	924
27.3.1	A/D 数据寄存器 A ~ H (ADDRA ~ ADDRH)	924
27.3.2	A/D 控制 / 状态寄存器 (ADCSR)	925
27.4	运行说明	927
27.4.1	单通道模式	927
27.4.2	多通道模式	929
27.4.3	扫描模式	931
27.4.4	通过外部触发或者多功能定时器脉冲单元 2 进行的 A/D 转换器启动	933
27.4.5	输入采样和 A/D 转换时间	933
27.4.6	外部触发的输入时序	934
27.5	中断源和 DMA 传送请求	935
27.6	A/D 转换精度的定义	936
27.7	使用时的注意事项	937
27.7.1	模块待机模式的设定	937
27.7.2	模拟电压的设定	937
27.7.3	电路板设计的注意事项	937
27.7.4	模拟输入引脚的处理	937
27.7.5	容许信号源阻抗	939
27.7.6	对绝对精度的影响	939
27.7.7	深度待机模式时的 A/D 转换	939
27.7.8	使用扫描模式或者多通道模式时的注意事项	939
28.	NAND 闪存控制器	940
28.1	特点	940
28.2	输入 / 输出引脚	942
28.3	寄存器说明	943
28.3.1	公共控制寄存器 (FLCMNCR)	944
28.3.2	命令控制寄存器 (FLCMDCR)	945
28.3.3	命令码寄存器 (FLCMCDR)	947
28.3.4	地址寄存器 (FLADR)	947
28.3.5	地址寄存器 2 (FLADR2)	948
28.3.6	数据计数寄存器 (FLDTCNTR)	949
28.3.7	数据寄存器 (FLDATAR)	950
28.3.8	中断 DMA 控制寄存器 (FLINTDMACR)	951
28.3.9	就绪 / 忙超时设定寄存器 (FLBSYTMR)	953
28.3.10	就绪 / 忙超时计数器 (FLSYCNT)	954
28.3.11	数据 FIFO 寄存器 (FLDTFIFO)	954
28.3.12	管理码 FIFO 寄存器 (FLECFIFO)	955
28.3.13	传送控制寄存器 (FLTRCR)	955
28.3.14	总线占有期间设定寄存器 (FLHOLDRCR)	956
28.4	运行说明	957

28.4.1	存取步骤	957
28.4.2	运行模式	957
28.4.3	寄存器设定步骤	958
28.4.4	命令存取模式	959
28.4.5	扇区存取模式	962
28.4.6	状态的读操作	965
28.5	中断处理	965
28.6	DMA 传送的设定	966
28.7	使用时的注意事项	967
28.7.1	外部总线权释放的时序	967
28.7.2	有关 SNAND 位的注意事项	967
29.	USB2.0 主机 / 功能模块	968
29.1	特点	968
29.2	输入 / 输出引脚	970
29.3	寄存器说明	971
29.3.1	系统配置控制寄存器 (SYSCFG)	973
29.3.2	CPU 总线等待寄存器 (BUSWAIT)	976
29.3.3	系统配置状态寄存器 (SYSSTS)	977
29.3.4	设备状态控制寄存器 (DVSTCTR)	978
29.3.5	测试模式寄存器 (TESTMODE)	981
29.3.6	DMA-FIFO 总线配置寄存器 (D0FBCFG、D1FBCFG)	982
29.3.7	FIFO 端口寄存器 (CFIFO、D0FIFO、D1FIFO)	983
29.3.8	FIFO 端口选择寄存器 (CFIFOSEL、D0FIFOSEL、D1FIFOSEL)	985
29.3.9	FIFO 端口控制寄存器 (CFIFOCTR、D0FIFOCTR、D1FIFOCTR)	989
29.3.10	中断允许寄存器 0 (INTENB0)	990
29.3.11	中断允许寄存器 1 (INTENB1)	991
29.3.12	BRDY 中断允许寄存器 (BRDYENB)	993
29.3.13	NRDY 中断允许寄存器 (NRDYENB)	994
29.3.14	BEMP 中断允许寄存器 (BEMPENB)	995
29.3.15	SOF 输出配置寄存器 (SOFCFG)	996
29.3.16	中断状态寄存器 0 (INTSTS0)	997
29.3.17	中断状态寄存器 1 (INTSTS1)	1000
29.3.18	BRDY 中断状态寄存器 (BRDYSTS)	1003
29.3.19	NRDY 中断状态寄存器 (NRDYSTS)	1004
29.3.20	BEMP 中断状态寄存器 (BEMPSTS)	1005
29.3.21	帧号寄存器 (FRMNUM)	1006
29.3.22	μ 帧号寄存器 (UFRMNUM)	1007
29.3.23	USB 地址寄存器 (USBADDR)	1007
29.3.24	USB 请求类型寄存器 (USBREQ)	1008
29.3.25	USB 请求值寄存器 (USBVAL)	1008
29.3.26	USB 请求变址寄存器 (USBINDX)	1009
29.3.27	USB 请求长度寄存器 (USBLENG)	1009
29.3.28	DCP 配置寄存器 (DCPCFG)	1010
29.3.29	DCP 最大信息包长度寄存器 (DCPMAXP)	1011
29.3.30	DCP 控制寄存器 (DCPCTR)	1012
29.3.31	管道窗口选择寄存器 (PIPESEL)	1016
29.3.32	管道配置寄存器 (PIPECFG)	1017
29.3.33	管道缓冲器指定寄存器 (PIPEBUF)	1020
29.3.34	管道最大信息包长度寄存器 (PIPEMAXP)	1022
29.3.35	管道周期控制寄存器 (PIPEPERI)	1023
29.3.36	管道 n 控制寄存器 (PIPEnCTR) (n=1 ~ 9)	1024

29.3.37	管道 n 事务计数允许寄存器 (PIPE _n TRE) (n=1 ~ 5)	1033
29.3.38	管道 n 事务计数寄存器 (PIPE _n TRN) (n=1 ~ 5)	1034
29.3.39	设备地址 n 的配置寄存器 (DEVADD _n) (n=0 ~ A)	1035
29.4	运行说明	1036
29.4.1	系统控制和振荡控制	1036
29.4.2	中断功能	1039
29.4.3	管道控制	1052
29.4.4	FIFO 缓冲存储器	1057
29.4.5	控制传送 (DCP)	1065
29.4.6	批量传送 (管道 1 ~ 5)	1067
29.4.7	中断传送 (管道 6 ~ 9)	1068
29.4.8	等时传送 (管道 1 和管道 2)	1069
29.4.9	SOF 内插功能	1076
29.4.10	管道的安排	1077
29.5	使用时的注意事项	1078
29.5.1	USB 收发器部电源	1078
30.	数字视频解码器	1079
30.1	特点	1079
30.2	框图	1080
30.3	输入 / 输出引脚	1081
30.4	寄存器说明	1081
30.4.1	ADC 控制寄存器 1 (ADCCR1)	1084
30.4.2	时序生成控制寄存器 (1) (TGCR1)	1084
30.4.3	时序生成控制寄存器 (2) (TGCR2)	1085
30.4.4	时序生成控制寄存器 (3) (TGCR3)	1085
30.4.5	同步分离控制寄存器 (1) (SYNSCR1)	1088
30.4.6	同步分离控制寄存器 (2) (SYNSCR2)	1092
30.4.7	同步分离控制寄存器 (3) (SYNSCR3)	1093
30.4.8	同步分离控制寄存器 (4) (SYNSCR4)	1094
30.4.9	同步分离控制寄存器 (5) (SYNSCR5)	1095
30.4.10	水平 AFC 控制寄存器 (1) (HAFCCR1)	1096
30.4.11	水平 AFC 控制寄存器 (2) (HAFCCR2)	1098
30.4.12	水平 AFC 控制寄存器 (3) (HAFCCR3)	1099
30.4.13	垂直递减计数控制寄存器 (1) (VCDWCR1)	1100
30.4.14	数字钳位控制寄存器 (1) (DCPCR1)	1102
30.4.15	数字钳位控制寄存器 (2) (DCPCR2)	1103
30.4.16	数字钳位控制寄存器 (3) (DCPCR3)	1104
30.4.17	数字钳位控制寄存器 (4) (DCPCR4)	1104
30.4.18	数字钳位控制寄存器 (5) (DCPCR5)	1105
30.4.19	数字钳位控制寄存器 (6) (DCPCR6)	1105
30.4.20	数字钳位控制寄存器 (7) (DCPCR7)	1106
30.4.21	数字钳位控制寄存器 (8) (DCPCR8)	1106
30.4.22	噪声检测控制寄存器 (NSDCR)	1107
30.4.23	色同步锁定控制 / 色度解码控制寄存器 (BTLCR)	1108
30.4.24	色同步选通脉冲控制寄存器 (BTGPCR)	1111
30.4.25	ACC 控制寄存器 (1) (ACCCR1)	1112
30.4.26	ACC 控制寄存器 (2) (ACCCR2)	1115
30.4.27	ACC 控制寄存器 (3) (ACCCR3)	1116
30.4.28	TINT 控制寄存器 (TINTCR)	1117
30.4.29	Y/C 延迟控制、色度解码控制寄存器 (YCDCR)	1118
30.4.30	AGC 控制寄存器 (1) (AGCCR1)	1120

30.4.31	AGC 控制寄存器 (2) (AGCCR2)	1122
30.4.32	峰值限制器的控制寄存器 (PKLIMITCR)	1123
30.4.33	超范围控制寄存器 (1) (RGORCR1)	1125
30.4.34	超范围控制寄存器 (2) (RGORCR2)	1126
30.4.35	超范围控制寄存器 (3) (RGORCR3)	1126
30.4.36	超范围控制寄存器 (4) (RGORCR4)	1127
30.4.37	超范围控制寄存器 (5) (RGORCR5)	1127
30.4.38	超范围控制寄存器 (6) (RGORCR6)	1128
30.4.39	超范围控制寄存器 (7) (RGORCR7)	1129
30.4.40	水平 AFC 相位比较器的反馈量调整寄存器 (AFPCFCR)	1130
30.4.41	寄存器的更新控制寄存器 (RUPDCR)	1131
30.4.42	同步分离状态、垂直周期读寄存器 (VSYNCR)	1132
30.4.43	水平周期读寄存器 (HSYNCSR)	1133
30.4.44	数字钳位读寄存器 (1) (DCPSR1)	1134
30.4.45	数字钳位读寄存器 (2) (DCPSR2)	1134
30.4.46	噪声检测读寄存器 (NSDSR)	1135
30.4.47	色度解码读寄存器 (1) (CROMASR1)	1136
30.4.48	色度解码读寄存器 (2) (CROMASR2)	1137
30.4.49	同步分离读寄存器 (SYNCSSR)	1138
30.4.50	AGC 控制读寄存器 (1) (AGCCSR1)	1138
30.4.51	AGC 控制读寄存器 (2) (AGCCSR2)	1139
30.4.52	Y/C 分离控制寄存器 (3) (YCSCR3)	1140
30.4.53	Y/C 分离控制寄存器 (4) (YCSCR4)	1141
30.4.54	Y/C 分离控制寄存器 (5) (YCSCR5)	1142
30.4.55	Y/C 分离控制寄存器 (6) (YCSCR6)	1142
30.4.56	Y/C 分离控制寄存器 (7) (YCSCR7)	1143
30.4.57	Y/C 分离控制寄存器 (8) (YCSCR8)	1144
30.4.58	Y/C 分离控制寄存器 (9) (YCSCR9)	1145
30.4.59	Y/C 分离控制寄存器 (11) (YCSCR11)	1146
30.4.60	Y/C 分离控制寄存器 (12) (YCSCR12)	1147
30.4.61	数字钳位控制寄存器 (9) (DCPCR9)	1149
30.4.62	Y/C 分离的色度信号滤波器 TAP 系数 (WA_F0 ~ F8) 寄存器 (YCTWA_F0 ~ F8)	1150
30.4.63	Y/C 分离的色度滤信号波器 TAP 系数 (WB_F0 ~ F8) 寄存器 (YCTWB_F0 ~ F8)	1151
30.4.64	Y/C 分离的色度信号滤波器 TAP 系数 (NA_F0 ~ F8) 寄存器 (YCTNA_F0 ~ F8)	1152
30.4.65	Y/C 分离的色度信号滤波器 TAP 系数 (NB_F0 ~ F8) 寄存器 (YCTNB_F0 ~ F8)	1153
30.4.66	亮度 (Y) 信号增益调整控制寄存器 (YGAINCR)	1154
30.4.67	色差 (Cb) 信号增益调整控制寄存器 (CBGAINCR)	1154
30.4.68	色差 (Cr) 信号增益调整控制寄存器 (CRGAINCR)	1155
30.4.69	PGA 部寄存器更新 (PGA_UPDATE)	1155
30.4.70	PGA 控制寄存器 (PGACR)	1156
30.4.71	ADC 控制寄存器 2 (ADCCR2)	1157
30.5	运行说明	1158
30.5.1	概要	1158
30.5.2	用于输入图像信号的 A/D 转换器	1159
30.5.3	同步分离电路	1161
30.5.4	Burst Controlled Oscillator (BCO)	1164
30.5.5	Y/C 分离电路	1165
30.5.6	色度解码	1173
30.5.7	数字钳位电路	1174
30.5.8	输出调整电路	1175
30.6	推荐设定	1176
30.7	连接例子	1180

31.	视频显示控制器 4 (1) 概要	1181
31.1	特点	1181
31.2	框图	1183
31.3	输入 / 输出引脚	1184
31.4	时钟	1184
31.5	水平 / 垂直同步信号	1184
32.	视频显示控制器 4 (2) 输入控制部	1187
32.1	输入控制功能	1187
32.1.1	功能概要	1187
32.1.2	外部输入部和同步信号调整部的寄存器更新控制	1187
32.1.3	输入选择	1188
32.1.4	外部输入的图像信号控制	1188
32.1.5	外部输入的时钟边沿选择	1189
32.1.6	外部输入的同步信号反相控制	1189
32.1.7	外部输入图像信号的位分配	1190
32.1.8	BT601 的标准信号时序	1193
32.1.9	BT656 的标准信号时序	1196
32.1.10	BT656 的 SAV 码和 EAV 码	1198
32.1.11	BT656/BT601 的设定	1201
32.1.12	YCbCr444/RGB888/666/565 的外部输入时序	1202
32.1.13	场判别和垂直同步相位调整	1204
32.1.14	垂直同步信号的行延迟调整	1205
32.1.15	同步延迟调整	1205
32.1.16	水平噪声的降低	1206
32.1.17	色彩矩阵	1208
32.2	寄存器说明	1211
32.2.1	外部输入部的寄存器更新控制寄存器 (INP_UPDATE)	1212
32.2.2	输入选择控制寄存器 (INP_SEL_CNT)	1213
32.2.3	外部输入同步信号控制寄存器 (INP_EXT_SYNC_CNT)	1214
32.2.4	垂直同步信号相位调整寄存器 (INP_VSYNC_PH_ADJ)	1215
32.2.5	同步信号延迟调整寄存器 (INP_DLY_ADJ)	1216
32.2.6	图像质量调整部的寄存器更新控制寄存器 (IMGCNT_UPDATE)	1216
32.2.7	NR 控制寄存器 0 (IMGCNT_NR_CNT0)	1217
32.2.8	NR 控制寄存器 1 (IMGCNT_NR_CNT1)	1218
32.2.9	图像质量调整部的矩阵模式寄存器 (IMGCNT_MTX_MODE)	1219
32.2.10	图像质量调整部的矩阵 YG 调整寄存器 0 (IMGCNT_MTX_YG_ADJ0)	1220
32.2.11	图像质量调整部的矩阵 YG 调整寄存器 1 (IMGCNT_MTX_YG_ADJ1)	1220
32.2.12	图像质量调整部的矩阵 CBB 调整寄存器 0 (IMGCNT_MTX_CBB_ADJ0)	1221
32.2.13	图像质量调整部的矩阵 CBB 调整寄存器 1 (IMGCNT_MTX_CBB_ADJ1)	1221
32.2.14	图像质量调整部的矩阵 CRR 调整寄存器 0 (IMGCNT_MTX_CRR_ADJ0)	1222
32.2.15	图像质量调整部的矩阵 CRR 调整寄存器 1 (IMGCNT_MTX_CRR_ADJ1)	1222
32.3	使用方法	1223
32.3.1	输入格式的调整方法	1223
32.3.2	色彩矩阵变换的使用方法	1225
33.	视频显示控制器 4 (3) 缩放部	1226
33.1	缩放功能	1226
33.1.1	功能概要	1226
33.1.2	寄存器控制	1227
33.1.3	同步控制	1228
33.1.4	视角大小的设定	1232

33.1.5	缩放设定	1235
33.1.6	水平前置滤波器	1236
33.1.7	水平缩小处理	1236
33.1.8	垂直缩小处理	1237
33.1.9	水平放大处理	1239
33.1.10	垂直放大处理	1240
33.1.11	IP 转换	1241
33.1.12	切边	1243
33.1.13	图像合成	1244
33.1.14	帧缓冲器的写图像格式选择	1244
33.1.15	水平镜像和旋转处理	1245
33.1.16	帧缓冲器的写处理	1246
33.1.17	放大处理和图形 (1) 处理的选择	1249
33.1.18	帧缓冲器的读处理	1251
33.2	寄存器说明	1251
33.2.1	SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE)	1253
33.2.2	屏蔽处理寄存器 (SCL0_FRC1)	1254
33.2.3	欠缺补偿寄存器 (SCL0_FRC2)	1254
33.2.4	输出同步选择寄存器 (SCL0_FRC3)	1255
33.2.5	自激周期寄存器 (SCL0_FRC4)	1255
33.2.6	输出延迟控制寄存器 (SCL0_FRC5)	1256
33.2.7	全屏幕垂直尺寸寄存器 (SCL0_FRC6)	1257
33.2.8	全屏幕水平尺寸寄存器 (SCL0_FRC7)	1258
33.2.9	同步检测寄存器 (SCL0_FRC9)	1259
33.2.10	缩小控制寄存器 (SCL0_DS1)	1260
33.2.11	捕捉垂直尺寸寄存器 (SCL0_DS2)	1261
33.2.12	捕捉水平尺寸寄存器 (SCL0_DS3)	1262
33.2.13	水平缩小寄存器 (SCL0_DS4)	1263
33.2.14	垂直初始相位寄存器 (SCL0_DS5)	1264
33.2.15	垂直缩放寄存器 (SCL0_DS6)	1265
33.2.16	缩小控制部输出尺寸寄存器 (SCL0_DS7)	1266
33.2.17	放大控制寄存器 (SCL0_US1)	1267
33.2.18	输出图像垂直尺寸寄存器 (SCL0_US2)	1268
33.2.19	输出图像水平尺寸寄存器 (SCL0_US3)	1269
33.2.20	放大控制部输入尺寸寄存器 (SCL0_US4)	1270
33.2.21	水平放大寄存器 (SCL0_US5)	1270
33.2.22	水平放大初始相位寄存器 (SCL0_US6)	1271
33.2.23	切边寄存器 (SCL0_US7)	1271
33.2.24	帧缓冲器的读选择寄存器 (SCL0_US8)	1272
33.2.25	背景色寄存器 (SCL0_OVR1)	1273
33.2.26	SCL1 寄存器的更新控制寄存器 (SCL1_UPDATE)	1274
33.2.27	写操作模式寄存器 (SCL1_WR1)	1275
33.2.28	写地址寄存器 1 (SCL1_WR2)	1276
33.2.29	写地址寄存器 2 (SCL1_WR3)	1277
33.2.30	写地址寄存器 3 (SCL1_WR4)	1278
33.2.31	帧减少寄存器 (SCL1_WR5)	1279
33.2.32	位压缩寄存器 (SCL1_WR6)	1280
33.2.33	写检测寄存器 (SCL1_WR7)	1280
33.2.34	图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE)	1281
33.2.35	帧缓冲器的读控制寄存器 (图形 (1)) (GR1_FLM_RD)	1281
33.2.36	帧缓冲控制寄存器 1 (图形 (1)) (GR1_FLM1)	1282
33.2.37	帧缓冲控制寄存器 2 (图形 (1)) (GR1_FLM2)	1283

33.2.38	帧缓冲控制寄存器 3 (图形 (1)) (GR1_FLM3)	1284
33.2.39	帧缓冲控制寄存器 4 (图形 (1)) (GR1_FLM4)	1285
33.2.40	帧缓冲控制寄存器 5 (图形 (1)) (GR1_FLM5)	1285
33.2.41	帧缓冲控制寄存器 6 (图形 (1)) (GR1_FLM6)	1286
33.2.42	α 混合控制寄存器 1 (图形 (1)) (GR1_AB1)	1287
33.2.43	α 混合控制寄存器 2 (图形 (1)) (GR1_AB2)	1288
33.2.44	α 混合控制寄存器 3 (图形 (1)) (GR1_AB3)	1289
33.2.45	α 混合控制寄存器 7 (图形 (1)) (GR1_AB7)	1290
33.2.46	α 混合控制寄存器 8 (图形 (1)) (GR1_AB8)	1290
33.2.47	α 混合控制寄存器 9 (图形 (1)) (GR1_AB9)	1291
33.2.48	α 混合控制寄存器 10 (图形 (1)) (GR1_AB10)	1292
33.2.49	α 混合控制寄存器 11 (图形 (1)) (GR1_AB11)	1293
33.2.50	背景色控制寄存器 (图形 (1)) (GR1_BASE)	1293
33.2.51	CLUT 表控制寄存器 (图形 (1)) (GR1_CLUT)	1294
33.3	使用方法	1295
33.3.1	显示输入图像时的缩放设定例子	1295
33.3.2	显示图形时的缩放设定例子	1298
33.3.3	放大显示图形时的缩放设定例子	1300
34.	视频显示控制器 4 (4) 图像质量改善部	1302
34.1	图像质量改善功能	1302
34.1.1	功能概要	1302
34.1.2	寄存器的更新控制	1302
34.1.3	黑色信号展宽	1302
34.1.4	增强器	1304
34.1.5	色彩矩阵	1309
34.2	寄存器说明	1311
34.2.1	图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE)	1312
34.2.2	黑色信号展宽部寄存器 (ADJ_BKSTR_SET)	1313
34.2.3	增强器的时序调整寄存器 1 (ADJ_ENH_TIM1)	1314
34.2.4	增强器的时序调整寄存器 2 (ADJ_ENH_TIM2)	1314
34.2.5	增强器的时序调整寄存器 3 (ADJ_ENH_TIM3)	1315
34.2.6	增强器的清晰度寄存器 1 (ADJ_ENH_SHP1)	1315
34.2.7	增强器的清晰度寄存器 2 (ADJ_ENH_SHP2)	1316
34.2.8	增强器的清晰度寄存器 3 (ADJ_ENH_SHP3)	1317
34.2.9	增强器的清晰度寄存器 4 (ADJ_ENH_SHP4)	1318
34.2.10	增强器的清晰度寄存器 5 (ADJ_ENH_SHP5)	1318
34.2.11	增强器的清晰度寄存器 6 (ADJ_ENH_SHP6)	1319
34.2.12	增强器的 LTI 寄存器 1 (ADJ_ENH_LTI1)	1320
34.2.13	增强器的 LTI 寄存器 2 (ADJ_ENH_LTI2)	1321
34.2.14	图像质量改善部的矩阵模式寄存器 (ADJ_MTX_MODE)	1322
34.2.15	图像质量改善部的矩阵 YG 调整寄存器 0 (ADJ_MTX_YG_ADJ0)	1322
34.2.16	图像质量改善部的矩阵 YG 调整寄存器 1 (ADJ_MTX_YG_ADJ1)	1323
34.2.17	图像质量改善部的矩阵 CBB 调整寄存器 0 (ADJ_MTX_CBB_ADJ0)	1323
34.2.18	图像质量改善部的矩阵 CBB 调整寄存器 1 (ADJ_MTX_CBB_ADJ1)	1324
34.2.19	图像质量改善部的矩阵 CRR 调整寄存器 0 (ADJ_MTX_CRR_ADJ0)	1324
34.2.20	图像质量改善部的矩阵 CRR 调整寄存器 1 (ADJ_MTX_CRR_ADJ1)	1325
34.3	使用方法	1326
34.3.1	黑色信号展宽的使用方法	1326
34.3.2	增强器的 LTI 处理	1326
34.3.3	增强器的清晰度处理	1327
34.3.4	色彩矩阵的数据转换设定方法	1328

35. 视频显示控制器 4 (5) 图像合成部	1329
35.1 图像合成功能	1329
35.1.1 功能概要	1329
35.1.2 图形数据读控制	1330
35.1.3 图形区的设定	1339
35.1.4 指定行中断的生成	1340
35.1.5 帧缓冲器读信号格式和 α 混合的对应	1340
35.1.6 显示选择	1341
35.1.7 背景色显示处理	1342
35.1.8 下层图形显示处理	1342
35.1.9 当前图形显示处理	1342
35.1.10 矩形区域 α 混合显示处理	1343
35.1.11 RGB 参照色键显示处理	1345
35.1.12 CLUT 参照色键显示处理	1346
35.1.13 以像素为单位的 α 混合显示处理	1347
35.1.14 α 混合运算式	1347
35.1.15 CLUT 表	1347
35.2 寄存器说明	1348
35.2.1 图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE)	1350
35.2.2 帧缓冲器的读控制寄存器 (图形 (2)) (GR2_FLM_RD)	1350
35.2.3 帧缓冲控制寄存器 1 (图形 (2)) (GR2_FLM1)	1351
35.2.4 帧缓冲控制寄存器 2 (图形 (2)) (GR2_FLM2)	1352
35.2.5 帧缓冲控制寄存器 3 (图形 (2)) (GR2_FLM3)	1353
35.2.6 帧缓冲控制寄存器 4 (图形 (2)) (GR2_FLM4)	1354
35.2.7 帧缓冲控制寄存器 5 (图形 (2)) (GR2_FLM5)	1354
35.2.8 帧缓冲控制寄存器 6 (图形 (2)) (GR2_FLM6)	1355
35.2.9 α 混合控制寄存器 1 (图形 (2)) (GR2_AB1)	1356
35.2.10 α 混合控制寄存器 2 (图形 (2)) (GR2_AB2)	1357
35.2.11 α 混合控制寄存器 3 (图形 (2)) (GR2_AB3)	1358
35.2.12 α 混合控制寄存器 4 (图形 (2)) (GR2_AB4)	1359
35.2.13 α 混合控制寄存器 5 (图形 (2)) (GR2_AB5)	1359
35.2.14 α 混合控制寄存器 6 (图形 (2)) (GR2_AB6)	1360
35.2.15 α 混合控制寄存器 7 (图形 (2)) (GR2_AB7)	1361
35.2.16 α 混合控制寄存器 8 (图形 (2)) (GR2_AB8)	1362
35.2.17 α 混合控制寄存器 9 (图形 (2)) (GR2_AB9)	1362
35.2.18 α 混合控制寄存器 10 (图形 (2)) (GR2_AB10)	1363
35.2.19 α 混合控制寄存器 11 (图形 (2)) (GR2_AB11)	1363
35.2.20 背景色控制寄存器 (图形 (2)) (GR2_BASE)	1364
35.2.21 CLUT 表控制寄存器 (图形 (2)) (GR2_CLUT)	1365
35.2.22 状态监视寄存器 (图形 (2)) (GR2_MON)	1365
35.2.23 图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE)	1366
35.2.24 帧缓冲器的读控制寄存器 (图形 (3)) (GR3_FLM_RD)	1366
35.2.25 帧缓冲控制寄存器 1 (图形 (3)) (GR3_FLM1)	1367
35.2.26 帧缓冲控制寄存器 2 (图形 (3)) (GR3_FLM2)	1368
35.2.27 帧缓冲控制寄存器 3 (图形 (3)) (GR3_FLM3)	1369
35.2.28 帧缓冲控制寄存器 4 (图形 (3)) (GR3_FLM4)	1370
35.2.29 帧缓冲控制寄存器 5 (图形 (3)) (GR3_FLM5)	1370
35.2.30 帧缓冲控制寄存器 6 (图形 (3)) (GR3_FLM6)	1371
35.2.31 α 混合控制寄存器 1 (图形 (3)) (GR3_AB1)	1372
35.2.32 α 混合控制寄存器 2 (图形 (3)) (GR3_AB2)	1373
35.2.33 α 混合控制寄存器 3 (图形 (3)) (GR3_AB3)	1374
35.2.34 α 混合控制寄存器 4 (图形 (3)) (GR3_AB4)	1375

35.2.35	α 混合控制寄存器 5 (图形 (3)) (GR3_AB5)	1375
35.2.36	α 混合控制寄存器 6 (图形 (3)) (GR3_AB6)	1376
35.2.37	α 混合控制寄存器 7 (图形 (3)) (GR3_AB7)	1377
35.2.38	α 混合控制寄存器 8 (图形 (3)) (GR3_AB8)	1378
35.2.39	α 混合控制寄存器 9 (图形 (3)) (GR3_AB9)	1378
35.2.40	α 混合控制寄存器 10 (图形 (3)) (GR3_AB10)	1379
35.2.41	α 混合控制寄存器 11 (图形 (3)) (GR3_AB11)	1379
35.2.42	背景色控制寄存器 (图形 (3)) (GR3_BASE)	1380
35.2.43	CLUT 表中断控制寄存器 (图形 (3)) (GR3_CLUT_INT)	1381
35.2.44	状态监视寄存器 (图形 (3)) (GR3_MON)	1382
35.3	使用方法	1382
35.3.1	静噪	1382
35.3.2	矩形区域 α 混合	1382
36.	视频显示控制器 4 (6) 输出控制部	1383
36.1	输出控制功能	1383
36.1.1	功能概要	1383
36.1.2	寄存器的更新控制	1383
36.1.3	路径选择	1384
36.1.4	亮度调整	1384
36.1.5	对比度调整	1385
36.1.6	灰度校正	1385
36.1.7	抖动处理	1389
36.1.8	输出格式的转换	1391
36.1.9	LCD TCON	1398
36.2	寄存器说明	1410
36.2.1	灰度校正部的寄存器更新控制寄存器 G (GAM_G_UPDATE)	1413
36.2.2	灰度校正部的功能切换寄存器 (GAM_SW)	1413
36.2.3	灰度校正部的表设定寄存器 G1 ~ 16 (GAM_G_LUT1 ~ 16)	1414
36.2.4	灰度校正部的区域设定寄存器 G1 (GAM_G_AREA1)	1416
36.2.5	灰度校正部的区域设定寄存器 G2 (GAM_G_AREA2)	1417
36.2.6	灰度校正部的区域设定寄存器 G3 (GAM_G_AREA3)	1418
36.2.7	灰度校正部的区域设定寄存器 G4 (GAM_G_AREA4)	1419
36.2.8	灰度校正部的区域设定寄存器 G5 (GAM_G_AREA5)	1420
36.2.9	灰度校正部的区域设定寄存器 G6 (GAM_G_AREA6)	1421
36.2.10	灰度校正部的区域设定寄存器 G7 (GAM_G_AREA7)	1422
36.2.11	灰度校正部的区域设定寄存器 G8 (GAM_G_AREA8)	1423
36.2.12	灰度校正部的寄存器更新控制寄存器 B (GAM_B_UPDATE)	1424
36.2.13	灰度校正部的表设定寄存器 B1 ~ 16 (GAM_B_LUT1 ~ 16)	1424
36.2.14	灰度校正部的区域设定寄存器 B1 (GAM_B_AREA1)	1427
36.2.15	灰度校正部的区域设定寄存器 B2 (GAM_B_AREA2)	1428
36.2.16	灰度校正部的区域设定寄存器 B3 (GAM_B_AREA3)	1429
36.2.17	灰度校正部的区域设定寄存器 B4 (GAM_B_AREA4)	1430
36.2.18	灰度校正部的区域设定寄存器 B5 (GAM_B_AREA5)	1431
36.2.19	灰度校正部的区域设定寄存器 B6 (GAM_B_AREA6)	1432
36.2.20	灰度校正部的区域设定寄存器 B7 (GAM_B_AREA7)	1433
36.2.21	灰度校正部的区域设定寄存器 B8 (GAM_B_AREA8)	1434
36.2.22	灰度校正部的寄存器更新控制寄存器 R (GAM_R_UPDATE)	1435
36.2.23	灰度校正部的表设定寄存器 R1 ~ 16 (GAM_R_LUT1 ~ 16)	1435
36.2.24	灰度校正部的区域设定寄存器 R1 (GAM_R_AREA1)	1438
36.2.25	灰度校正部的区域设定寄存器 R2 (GAM_R_AREA2)	1439
36.2.26	灰度校正部的区域设定寄存器 R3 (GAM_R_AREA3)	1440

36.2.27	灰度校正部的区域设定寄存器 R4 (GAM_R_AREA4)	1441
36.2.28	灰度校正部的区域设定寄存器 R5 (GAM_R_AREA5)	1442
36.2.29	灰度校正部的区域设定寄存器 R6 (GAM_R_AREA6)	1443
36.2.30	灰度校正部的区域设定寄存器 R7 (GAM_R_AREA7)	1444
36.2.31	灰度校正部的区域设定寄存器 R8 (GAM_R_AREA8)	1445
36.2.32	TCON 寄存器的更新控制寄存器 (TCON_UPDATE)	1446
36.2.33	TCON 基准时序设定寄存器 (TCON_TIM)	1446
36.2.34	TCON 垂直时序设定寄存器 A1 (TCON_TIM_STVA1)	1447
36.2.35	TCON 垂直时序设定寄存器 A2 (TCON_TIM_STVA2)	1448
36.2.36	TCON 垂直时序设定寄存器 B1 (TCON_TIM_STVB1)	1449
36.2.37	TCON 垂直时序设定寄存器 B2 (TCON_TIM_STVB2)	1450
36.2.38	TCON 水平时序设定寄存器 STH1 (TCON_TIM_STH1)	1451
36.2.39	TCON 水平时序设定寄存器 STH2 (TCON_TIM_STH2)	1452
36.2.40	TCON 水平时序设定寄存器 STB1 (TCON_TIM_STB1)	1453
36.2.41	TCON 水平时序设定寄存器 STB2 (TCON_TIM_STB2)	1454
36.2.42	TCON 水平时序设定寄存器 CPV1 (TCON_TIM_CPV1)	1455
36.2.43	TCON 水平时序设定寄存器 CPV2 (TCON_TIM_CPV2)	1456
36.2.44	TCON 水平时序设定寄存器 POLA1 (TCON_TIM_POLA1)	1457
36.2.45	TCON 水平时序设定寄存器 POLA2 (TCON_TIM_POLA2)	1458
36.2.46	TCON 水平时序设定寄存器 POLB1 (TCON_TIM_POLB1)	1459
36.2.47	TCON 水平时序设定寄存器 POLB2 (TCON_TIM_POLB2)	1460
36.2.48	TCON 数据允许极性设定寄存器 (TCON_TIM_DE)	1461
36.2.49	输出控制部的寄存器更新控制寄存器 (OUT_UPDATE)	1461
36.2.50	输出接口寄存器 (OUT_SET)	1462
36.2.51	亮度 (DC) 校正寄存器 1 (OUT_BRIGHT1)	1463
36.2.52	亮度 (DC) 校正寄存器 2 (OUT_BRIGHT2)	1464
36.2.53	对比度 (增益) 校正寄存器 (OUT_CONTRAST)	1464
36.2.54	显示屏抖动寄存器 (OUT_PDTHA)	1465
36.2.55	输出相位控制寄存器 (OUT_CLK_PHASE)	1466
36.3	使用方法	1467
36.3.1	灰度校正的调整方法	1467
36.3.2	抖动的使用方法	1467
36.3.3	输出格式的调整方法	1468
37.	视频显示控制器 4 (7) 系统控制部	1470
37.1	系统控制功能	1470
37.1.1	功能概要	1470
37.1.2	中断控制	1470
37.1.3	显示屏时钟控制	1473
37.1.4	CLUT 表的读选择信号状态标志	1474
37.2	寄存器说明	1474
37.2.1	中断控制寄存器 1 (SYSCNT_INT1)	1475
37.2.2	中断控制寄存器 2 (SYSCNT_INT2)	1476
37.2.3	中断控制寄存器 3 (SYSCNT_INT3)	1477
37.2.4	中断控制寄存器 4 (SYSCNT_INT4)	1478
37.2.5	显示屏时钟控制寄存器 (SYSCNT_PANEL_CLK)	1479
37.2.6	CLUT 表的读选择信号状态标志寄存器 (SYSCNT_CLUT)	1480
38.	失真校正引擎	1481
39.	显示输出比较单元 (DISCOM)	1482
39.1	特点	1482

39.2	框图	1482
39.3	寄存器说明	1483
39.3.1	控制寄存器 (DOCMCR)	1484
39.3.2	状态寄存器 (DOCMSTR)	1485
39.3.3	状态清除寄存器 (DOCMCLSTR)	1485
39.3.4	中断允许寄存器 (DOCMIENR)	1486
39.3.5	工作参数设定寄存器 (DOCMPMR)	1487
39.3.6	CRC 码期待值寄存器 (DOCMECR)	1488
39.3.7	CRC 码计算值寄存器 (DOCMCCR)	1488
39.3.8	水平方向起始位置设定寄存器 (DOCMSPXR)	1489
39.3.9	垂直方向起始位置设定寄存器 (DOCMSPYR)	1489
39.3.10	水平方向尺寸寄存器 (DOCMSZXR)	1490
39.3.11	垂直方向尺寸寄存器 (DOCMSZYR)	1490
39.3.12	显示输出比较的 CRC 初始值寄存器 (DOCMCIRCIR)	1491
39.4	运行说明	1491
39.4.1	运行概要	1491
39.4.2	系统结构	1491
39.4.3	CRC 计算方法	1492
39.4.4	CRC 码生成图形数据的选择	1492
39.4.5	像素格式	1492
39.4.6	矩形区域的设定	1493
39.4.7	CRC 计算期间和比较时序	1494
39.4.8	寄存器的更新时序	1495
39.4.9	运行流程	1496
39.5	中断	1497
39.6	使用时的注意事项	1497
39.6.1	CRC 期待值	1497
39.6.2	放大控制功能	1497
40.	OpenVG 瑞萨图形处理器	1498
41.	JPEG 编解码器单元 (JCU)	1499
41.1	特点	1499
41.2	寄存器说明	1501
41.2.1	JPEG 码模式寄存器 (JCMOD)	1502
41.2.2	JPEG 码命令寄存器 (JCCMD)	1503
41.2.3	JPEG 码量化表号寄存器 (JCQTN)	1504
41.2.4	JPEG 码霍夫曼表号寄存器 (JCHTN)	1504
41.2.5	JPEG 码 DRI 高位寄存器 (JCDRIU)	1505
41.2.6	JPEG 码 DRI 低位寄存器 (JCDRID)	1505
41.2.7	JPEG 码垂直方向尺寸高位寄存器 (JCVSZU)	1505
41.2.8	JPEG 码垂直方向尺寸低位寄存器 (JCVSZD)	1506
41.2.9	JPEG 码水平方向尺寸高位寄存器 (JCHSZU)	1506
41.2.10	JPEG 码水平方向尺寸低位寄存器 (JCHSZD)	1506
41.2.11	JPEG 码数据计数高位寄存器 (JCDTCU)	1507
41.2.12	JPEG 码数据计数中位寄存器 (JCDTCM)	1507
41.2.13	JPEG 码数据计数低位寄存器 (JCDTCD)	1507
41.2.14	JPEG 中断允许寄存器 0 (JINTE0)	1508
41.2.15	JPEG 中断状态寄存器 0 (JINTS0)	1509
41.2.16	JPEG 码解码错误寄存器 (JCDERR)	1509
41.2.17	JPEG 码重新启动寄存器 (JCRST)	1510
41.2.18	JPEG 接口压缩控制寄存器 (JIFECNT)	1510

41.2.19	JPEG 接口压缩源地址寄存器 (JIFESA)	1512
41.2.20	JPEG 接口压缩行偏移寄存器 (JIFESOFST)	1512
41.2.21	JPEG 接口压缩目标地址寄存器 (JIFEDA)	1513
41.2.22	JPEG 接口压缩源行计数寄存器 (JIFESLC)	1513
41.2.23	JPEG 接口压缩目标计数寄存器 (JIFEDDC)	1514
41.2.24	JPEG 接口解压控制寄存器 (JIFDCNT)	1515
41.2.25	JPEG 接口解压源地址寄存器 (JIFDSA)	1517
41.2.26	JPEG 接口解压行偏移寄存器 (JIFDDOFST)	1517
41.2.27	JPEG 接口解压目标地址寄存器 (JIFDDA)	1518
41.2.28	JPEG 接口解压源数据计数寄存器 (JIFSDC)	1518
41.2.29	JPEG 接口解压目标行计数寄存器 (JIFDDL)	1519
41.2.30	JPEG 接口解压 α 设定寄存器 (JIFDADT)	1519
41.2.31	JPEG 中断允许寄存器 1 (JINTE1)	1520
41.2.32	JPEG 中断状态寄存器 1 (JINTS1)	1521
41.3	运行说明	1522
41.3.1	压缩	1522
41.3.2	解压	1528
41.3.3	解压时的输出像素格式	1533
41.3.4	图像数据的保存	1537
41.4	中断	1537
41.4.1	压缩解压处理中断请求 (JEDI)	1537
41.4.2	数据传送处理中断请求 (JDTI)	1538
41.5	总线复位处理	1539
41.6	使用时的注意事项	1539
41.6.1	像素格式 YCbCr	1539
42.	采样率转换器	1540
42.1	特点	1540
42.2	寄存器说明	1541
42.2.1	输入数据寄存器 (SRCID)	1541
42.2.2	输出数据寄存器 (SRCOD)	1542
42.2.3	输入数据控制寄存器 (SRCIDCTRL)	1543
42.2.4	输出数据控制寄存器 (SRCODCTRL)	1544
42.2.5	控制寄存器 (SRCCTRL)	1545
42.2.6	状态寄存器 (SRCSTAT)	1548
42.3	运行说明	1550
42.3.1	初始设定	1550
42.3.2	数据输入	1550
42.3.3	数据输出	1552
42.4	中断	1553
42.5	使用时的注意事项	1553
42.5.1	存取寄存器时的注意点	1553
42.5.2	有关转储清除的注意点	1553
43.	音频发生器	1554
43.1	特点	1554
43.2	输入 / 输出引脚	1555
43.3	寄存器说明	1555
43.3.1	音频发生器的控制寄存器 1 (SGCR1)	1556
43.3.2	音频发生器的控制状态寄存器 (SGCSR)	1557
43.3.3	音频发生器的控制寄存器 2 (SGCR2)	1557
43.3.4	音频发生器的音量寄存器 (SGLR)	1558

43.3.5	音频发生器的音频寄存器 (SGTFR)	1558
43.3.6	音频发生器的基准频率寄存器 (SGSFR)	1558
43.4	运行说明	1559
43.4.1	基本运行	1559
43.4.2	TONE 频率的设定	1562
43.4.3	自动衰减功能	1563
43.4.4	输出波形	1563
43.5	中断源	1564
43.6	使用时的注意事项	1564
43.6.1	模块停止模式的设定	1564
44.	SD 主机接口	1565
45.	MMC 主机接口	1566
45.1	特点	1566
45.2	输入 / 输出引脚	1566
45.3	寄存器说明	1567
45.3.1	命令设定寄存器 (CE_CMD_SET)	1568
45.3.2	自变量寄存器 (CE_ARG)	1570
45.3.3	自动 CMD12 的自变量寄存器 (CE_ARG_CMD12)	1570
45.3.4	命令控制寄存器 (CE_CMD_CTRL)	1571
45.3.5	传送块设定寄存器 (CE_BLOCK_SET)	1571
45.3.6	时钟控制寄存器 (CE_CLK_CTRL)	1572
45.3.7	缓冲器存取设定寄存器 (CE_BUF_ACC)	1573
45.3.8	应答寄存器 3 ~ 0 (CE_RESP3 ~ 0)	1574
45.3.9	自动 CMD12 的应答寄存器 (CE_RESP_CMD12)	1575
45.3.10	数据寄存器 (CE_DATA)	1575
45.3.11	中断标志寄存器 (CE_INT)	1576
45.3.12	中断允许寄存器 (CE_INT_EN)	1580
45.3.13	状态寄存器 1 (CE_HOST_STS1)	1582
45.3.14	状态寄存器 2 (CE_HOST_STS2)	1583
45.3.15	DMA 模式设定寄存器 (CE_DMA_MODE)	1585
45.3.16	卡检测 / 端口控制寄存器 (CE_DETECT)	1586
45.3.17	特殊模式设定寄存器 (CE_ADD_MODE)	1587
45.3.18	版本寄存器 (CE_VERSION)	1587
45.4	中断请求的说明	1588
45.5	DMA 规格	1588
45.5.1	写缓冲器的 DMA 说明	1588
45.5.2	读缓冲器的 DMA 说明	1589
45.6	运行说明	1589
45.6.1	命令 / 应答的格式	1589
45.6.2	数据块的格式	1590
45.6.3	缓冲器结构和缓冲器的存取	1591
45.6.4	CMD12 的自动发行	1592
45.6.5	发生错误或者超时的此模块的处理	1593
45.7	设定例子	1594
45.7.1	范例说明	1594
45.7.2	命令发送的设定例子	1594
45.7.3	命令发送 → 应答接收的设定例子	1595
45.7.4	命令发送 → 应答接收 (应答忙) 的设定例子	1596
45.7.5	单块读的设定例子	1598
45.7.6	多块读的设定例子	1599

45.7.7	多块读（有自动 CMD12）的设定例子	1600
45.7.8	单块写的设定例子	1601
45.7.9	多块写的设定例子	1602
45.7.10	多块写（有自动 CMD12）的设定例子	1603
45.7.11	强制结束的设定例子	1604
45.7.12	CE_CMD_SET 的设定值	1605
45.8	使用时的注意事项	1606
45.8.1	卡检测	1606
46.	马达控制 PWM 定时器	1607
46.1	特点	1607
46.2	输入 / 输出引脚	1608
46.3	寄存器说明	1608
46.3.1	PWM 控制寄存器 $_n$ (PWCR $_n$) (n=1、2)	1609
46.3.2	PWM 极性寄存器 $_n$ (PWPR $_n$) (n=1、2)	1610
46.3.3	PWM 计数器 $_n$ (PWCNT $_n$) (n=1、2)	1610
46.3.4	PWM 周期寄存器 $_n$ (PWCYR $_n$) (n=1、2)	1610
46.3.5	PWM 占空比寄存器 $_nA$ 、 nC 、 nE 、 nG (PWDTR $_nA$ 、PWDTR $_nC$ 、 PWDTR $_nE$ 、PWDTR $_nG$) (n=1、2)	1611
46.3.6	PWM 缓冲寄存器 $_nA$ 、 nC 、 nE 、 nG (PWBFR $_nA$ 、PWBFR $_nC$ 、 PWBFR $_nE$ 、PWBFR $_nG$)	1613
46.3.7	PWM 缓冲传送控制寄存器 (PWBTCR)	1614
46.4	和总线主控的接口	1615
46.4.1	16 位数据寄存器	1615
46.4.2	8 位数据寄存器	1615
46.5	运行说明	1616
46.5.1	PWM 的运行	1616
46.5.2	缓冲器的传送控制	1617
46.6	使用注意事项	1617
46.6.1	缓冲寄存器的写和比较匹配的竞争	1617
47.	内部 RAM	1618
47.1	特点	1618
47.2	使用时的注意事项	1620
47.2.1	页面竞争	1620
47.2.2	RAME 位和 RAMWE 位	1620
47.2.3	数据保持	1621
48.	通用输入 / 输出端口	1622
48.1	特点	1622
48.2	寄存器说明	1629
48.2.1	端口 A 的 IO 寄存器 0 (PAIOR0)	1631
48.2.2	端口 A 的数据寄存器 0 (PADR0)	1631
48.2.3	端口 A 的端口寄存器 0 (PAPR0)	1632
48.2.4	端口 B 的控制寄存器 0 ~ 5 (PBCR0 ~ PBCR5)	1633
48.2.5	端口 B 的 IO 寄存器 0、1 (PBIOR0、PBIOR1)	1640
48.2.6	端口 B 的数据寄存器 0、1 (PBDR0、PBDR1)	1640
48.2.7	端口 B 的端口寄存器 0、1 (PBPR0、PBPR1)	1642
48.2.8	端口 C 的控制寄存器 0 ~ 2 (PCCR0 ~ PCCR2)	1643
48.2.9	端口 C 的 IO 寄存器 0 (PCIOR0)	1646
48.2.10	端口 C 的数据寄存器 0 (PCDR0)	1646
48.2.11	端口 C 的端口寄存器 0 (PCPR0)	1647

48.2.12	端口 D 的控制寄存器 0 ~ 3 (PDCR0 ~ PDCR3)	1648
48.2.13	端口 D 的 IO 寄存器 0 (PDIOR0)	1652
48.2.14	端口 D 的数据寄存器 0 (PDDR0)	1652
48.2.15	端口 D 的端口寄存器 0 (PDPR0)	1653
48.2.16	端口 E 的控制寄存器 0、1 (PECR0、PECR1)	1654
48.2.17	端口 E 的 IO 寄存器 0 (PEIOR0)	1656
48.2.18	端口 E 的数据寄存器 0 (PEDR0)	1656
48.2.19	端口 E 的端口寄存器 0 (PEPR0)	1657
48.2.20	端口 F 的控制寄存器 0 ~ 6 (PFCR0 ~ PFCR6)	1658
48.2.21	端口 F 的 IO 寄存器 0、1 (PFIOR0、PFIOR1)	1665
48.2.22	端口 F 的数据寄存器 0、1 (PFDR0、PFDR1)	1666
48.2.23	端口 F 的端口寄存器 0、1 (PFPR0、PFPR1)	1668
48.2.24	端口 G 的控制寄存器 0 ~ 6 (PGCR0 ~ PGCR6)	1670
48.2.25	端口 G 的 IO 寄存器 0、1 (PGIOR0、PGIOR1)	1678
48.2.26	端口 G 的数据寄存器 0、1 (PGDR0、PGDR1)	1679
48.2.27	端口 G 的端口寄存器 0、1 (PGPR0、PGPR1)	1681
48.2.28	端口 H 的控制寄存器 0、1 (PHCR0、PHCR1)	1683
48.2.29	端口 H 的端口寄存器 0 (PHPR0)	1685
48.2.30	端口 J 的控制寄存器 0 ~ 7 (PJCR0 ~ PJCR7: 只限于 SH7269)	1686
48.2.31	端口 J 的 IO 寄存器 0、1 (PJIOR0、PJIOR1: 只限于 SH7269)	1694
48.2.32	端口 J 的数据寄存器 0、1 (PJDR0、PJDR1: 只限于 SH7289)	1695
48.2.33	端口 J 的端口寄存器 0、1 (PJPR0、PJPR1: 只限于 SH7269)	1697
48.2.34	串行瑞萨接口噪声消除器的控制寄存器 (SNCR)	1699
49.	低功耗模式	1700
49.1	特点	1700
49.1.1	低功耗模式的种类	1700
49.2	寄存器说明	1701
49.2.1	待机控制寄存器 1 (STBCR1)	1702
49.2.2	待机控制寄存器 2 (STBCR2)	1703
49.2.3	待机控制寄存器 3 (STBCR3)	1704
49.2.4	待机控制寄存器 4 (STBCR4)	1705
49.2.5	待机控制寄存器 5 (STBCR5)	1706
49.2.6	待机控制寄存器 6 (STBCR6)	1707
49.2.7	待机控制寄存器 7 (STBCR7)	1708
49.2.8	待机控制寄存器 8 (STBCR8)	1709
49.2.9	待机控制寄存器 9 (STBCR9)	1710
49.2.10	待机控制寄存器 10 (STBCR10)	1711
49.2.11	软件复位控制寄存器 1 (SWRSTCR1)	1712
49.2.12	软件复位控制寄存器 2 (SWRSTCR2)	1713
49.2.13	系统控制寄存器 1 (SYSCR1)	1714
49.2.14	系统控制寄存器 2 (SYSCR2)	1715
49.2.15	系统控制寄存器 3 (SYSCR3)	1716
49.2.16	系统控制寄存器 4 (SYSCR4)	1717
49.2.17	系统控制寄存器 5 (SYSCR5)	1718
49.2.18	用于保持的内部 RAM 保持区指定寄存器 (RRAMKP)	1719
49.2.19	深度待机控制寄存器 (DSCTR)	1720
49.2.20	深度待机解除源选择寄存器 (DSSSR)	1721
49.2.21	深度待机解除边沿选择寄存器 (DSESR)	1722
49.2.22	深度待机解除源标志寄存器 (DSFR)	1724
49.2.23	XTAL 晶体振荡器的增益控制寄存器 (XTALCTR)	1725
49.3	运行说明	1726

49.3.1	睡眠模式	1726
49.3.2	软件待机模式	1726
49.3.3	软件待机模式的应用例子	1728
49.3.4	深度待机模式	1729
49.3.5	模块待机功能	1733
49.3.6	XTAL 晶体振荡器的增益调整功能	1733
49.4	使用时的注意事项	1734
49.4.1	设定寄存器时的注意	1734
49.4.2	未使用实时时钟时的注意	1734
50.	用户调试接口	1735
50.1	特点	1735
50.2	输入 / 输出引脚	1736
50.3	边界扫描 TAP 控制器的寄存器说明	1736
50.3.1	旁路寄存器 (BSBPR)	1736
50.3.2	指令寄存器 (BSIR)	1737
50.3.3	边界扫描寄存器 (SDBSR)	1737
50.3.4	ID 寄存器 (BSID)	1741
50.4	仿真 TAP 控制器的寄存器说明	1741
50.4.1	旁路寄存器 (SDBPR)	1741
50.4.2	指令寄存器 (SDIR)	1742
50.5	运行说明	1743
50.5.1	TAP 控制器	1743
50.5.2	复位结构	1744
50.5.3	TDO 输出时序	1744
50.5.4	用户调试接口复位	1745
50.5.5	用户调试接口中断	1745
50.6	边界扫描	1745
50.6.1	支持的命令	1745
50.6.2	注意事项	1746
50.7	使用时的注意事项	1746
51.	寄存器一览	1747
51.1	寄存器地址一览表 (按功能模块和手册章节号的顺序)	1748
51.2	寄存器位一览表	1786
51.3	各运行模式中的寄存器状态一览表	1885
52.	电特性	1887
52.1	绝对最大额定值	1887
52.2	接通和切断电源的顺序	1887
52.3	DC 特性	1888
52.4	AC 特性	1895
52.4.1	时钟时序	1895
52.4.2	控制信号时序	1899
52.4.3	总线时序	1901
52.4.4	UBC 时序	1931
52.4.5	直接存储器存取控制器的时序	1931
52.4.6	多功能定时器脉冲单元 2 的时序	1932
52.4.7	看门狗定时器的时序	1932
52.4.8	FIFO 内置型串行通信接口的时序	1933
52.4.9	瑞萨串行外围接口的时序	1934
52.4.10	瑞萨四路串行外围接口的时序	1936

52.4.11	SPI 多 I/O 总线控制器的时序	1938
52.4.12	I ² C 总线接口 3 的时序	1940
52.4.13	串行音频接口的时序	1941
52.4.14	带 FIFO 的时钟同步串行 I/O 的时序	1942
52.4.15	A/D 转换器的时序	1944
52.4.16	NAND 闪存控制器的时序	1945
52.4.17	USB2.0 主机 / 功能模块的时序	1949
52.4.18	视频显示控制器 4 的时序	1951
52.4.19	SD 主机接口的时序	1953
52.4.20	MMC 主机接口的时序	1954
52.4.21	通用输入 / 输出端口的时序	1955
52.4.22	用户调试接口的时序	1955
52.4.23	AC 特性的测量条件	1957
52.5	A/D 转换器特性	1957
52.6	用于输入视频信号的 A/D 转换器特性	1958
53.	引脚状态和处理方法	1959
53.1	引脚状态	1959
53.2	未使用引脚的处理	1967
53.3	深度待机模式中的引脚处理	1968
53.4	旁路电容的推荐组合	1969
附录	1971
附录 1.	封装尺寸图.....	1971
索引	1974

1. 概要

1.1 SH7268/7269 的特点

本 LSI 是以瑞萨独创的 RISC（精简指令集计算机）方式 CPU 为核心并且集聚了系统构成上所需外围功能的 RISC 单片机。

本 LSI 的 CPU 采用了在目标代码级上和 SH-1、SH-2 和 SH-2E 的单片机有向上兼容的 SH-2A CPU。本 LSI 的 CPU 有 RISC 方式的指令系统，因为采用了超标量体系结构和哈佛体系结构，所以飞跃地提高了指令的执行速度。另外，采用了独立的内部 32 位总线结构，强化了数据处理能力。对于传统的单片机不能实现的高速性实时控制等应用系统，本 LSI 的 CPU 能构成更低成本、高性能和高功能的系统。

本 LSI 内置浮点单元和高速缓存。作为系统构成上需要的外围功能，本 LSI 内置 64KB 的高速内部 RAM 和 2.5MB 的大容量 RAM（其中 128KB 和用于保持数据的 RAM 共享）、用于保持数据的 RAM、多功能定时器脉冲单元 2、比较匹配定时器、实时时钟、FIFO 内置型串行通信接口、I²C 总线接口 3、串行音频接口、带 FIFO 的时钟同步串行 I/O*2、控制器局域网 *2、IEBusTM*1 控制器、瑞萨 SPDIF 接口、瑞萨串行外围接口、瑞萨四路串行外围接口、SPI 多 I/O 总线控制器、CD-ROM 解码器、A/D 转换器、NAND 闪存控制器、USB2.0 主机 / 功能模块、数字视频解码器、视频显示控制器 4、失真校正引擎、显示输出比较单元、OpenVG 瑞萨图形处理器、JPEG 编解码器单元、采样率转换器、音频发生器、SD 主机接口、MMC 主机接口、马达控制 PWM 定时器、中断控制器、通用输入 / 输出端口等。

另外，本 LSI 能通过外部存储器存取支持功能直接连接存储器和外围 LSI，从而能大幅度地降低系统的成本。

本 LSI 的特点如表 1.1 所示。

【注】 *1 IEBus（Inter Equipment Bus）是瑞萨电子公司的商标。

*2 根据产品型号，分为搭载产品和非搭载产品。

表 1.1 SH7268/7269 的特点

项目	特点
CPU	<ul style="list-style-type: none"> • 瑞萨独创的 SuperH 体系结构 • 在目标代码级上和 SH-1、SH-2、SH-2E 兼容。 • 32 位内部数据总线 • 通用寄存器体系结构 <ul style="list-style-type: none"> 16 个 32 位通用寄存器 4 个 32 位控制寄存器 4 个 32 位系统寄存器 • 用于高速中断应答的寄存器组 • RISC 型指令系统（和 SH 系列向上兼容） <ul style="list-style-type: none"> 指令长度：用于提高代码效率的 16 位基本指令以及用于提高性能和可用性的 32 位指令 装入 - 存储结构 延迟转移指令 基于 C 语言的指令系统 • 同时执行 2 条指令（包括浮点单元）的超标量体系结构 • 指令的执行时间：最多 2 条指令 / 周期 • 地址空间：4G 字节 • 内置乘法器 • 5 段流水线 • 哈佛体系结构
浮点单元	<ul style="list-style-type: none"> • 内置浮点协处理器。 • 支持单精度（32 位）和双精度（64 位）。 • 支持符合 IEEE754 规格的数据类型和浮点异常。 • 舍入模式：向接近的值和“0”方向舍入 • 非规格化数的处理：向“0”舍去 • 浮点寄存器 <ul style="list-style-type: none"> 16 个 32 位浮点寄存器（单精度 ×16 个字或者双精度 ×8 个字） 2 个 32 位浮点系统寄存器 • 支持 FMAC（乘法和累加）指令。 • 支持 FDIV（除法）和 FSQRT（平方根）指令。 • 支持 FLDI0 和 FLDI1（装入常数 0/1）指令。 • 指令的执行时间 <ul style="list-style-type: none"> 等待时间（FMAC/FADD/FSUB/FMUL）：3 个周期（单精度）、8 个周期（双精度） 节距（FMAC/FADD/FSUB/FMUL）：1 个周期（单精度）、6 个周期（双精度） • 【注】FMAC 只支持单精度。 • 5 段流水线
高速缓存	<ul style="list-style-type: none"> • 指令的高速缓存：8K 字节 • 操作数的高速缓存：8K 字节 • 各自的 128 个入口 / 路、4 路组相联方式、16 字节块长 • 回写、直写、LRU 替换算法 • 有路锁定功能（只限于操作数的高速缓存）：能锁定路 2 和路 3
中断控制器	<ul style="list-style-type: none"> • 17 个外部中断引脚（NMI、IRQ7 ~ IRQ0、PINT7 ~ PINT0） • 内部外围中断：给各模块设定优先级 • 能设定 16 个优先级。 • 能通过寄存器组对中断处理的寄存器高速地进行保存和恢复。

项目	特点
总线状态控制器	<ul style="list-style-type: none"> 地址空间分别支持最大 64MB 的 6 个区域 0 ~ 5。 能分别给各区域设定以下功能： 总线宽度（8 位、16 位、32 位）（但是各区域支持的宽度不同） 存取等待周期数（有些区域能在读写存取周期设定独立的等待周期） 空闲等待周期的设定（同一区域 / 不同区域） 能通过指定连接各区域的存储器，直接连接 SRAM、带字节选择的 SRAM、SDRAM、突发 ROM（时钟同步 / 异步）。另外，还支持地址 / 数据多路复用 I/O（MPX）接口。 支持 PCMCIA 接口。 将片选信号（$\overline{CS0} \sim \overline{CS5}$）输出到对象区域。 （能通过编程选择 \overline{CS} 的有效或者无效） SDRAM 刷新功能 支持自动刷新模式和自刷新模式。 SDRAM 突发存取功能
直接存储器存取控制器	<ul style="list-style-type: none"> 16 个通道，其中 1 个通道能进行外部请求。 能从内部外围模块启动。 突发模式和周期挪用模式 支持间歇模式（支持 16/64 个周期）。 能自动重新加载传送信息。
时钟脉冲振荡器	<ul style="list-style-type: none"> 时钟模式：可选择外部输入（EXTAL）或者晶体谐振器作为输入时钟。 能通过内部 PLL 电路将输入时钟最大递增 20 倍。 能通过内部 SSCG 电路减低 EMI 噪声峰值。 生成 4 种时钟： CPU 时钟（$I\phi$）：最大 266.67MHz 内部总线时钟（$B\phi$）：最大 133.33MHz 外围时钟 1（$P1\phi$）：最大 66.67MHz 外围时钟 0（$P0\phi$）：最大 33.33MHz
看门狗定时器	<ul style="list-style-type: none"> 1 个通道的看门狗定时器 能通过计数器的上溢对本 LSI 进行复位。
低功耗模式	<ul style="list-style-type: none"> 为了降低本 LSI 的功耗，支持以下 4 种低功耗模式： 睡眠模式 软件待机模式 深度待机模式 模块待机模式
多功能定时器脉冲单元 2	<ul style="list-style-type: none"> 能以 5 个通道的 16 位定时器为基础，最多输入或者输出 16 种脉冲。 18 个输出比较寄存器兼输入捕捉寄存器 输入捕捉功能 脉冲输出模式 交替 / PWM / 互补 PWM / 复位同步 PWM 多个计数器的同步功能 互补 PWM 输出模式 输出用于控制 3 相反相器的非重叠波形。 自动设定空载时间。 能任意设定 0 ~ 100% 的 PWM 占空比。 A/D 转换请求的延迟功能 波峰 / 波谷的中断减少功能 复位同步 PWM 模式 3 相输出任意占空比的正相 / 反相 PWM 波形。 相位计数模式 能进行 2 相编码器的计数处理。

项目	特点
比较匹配定时器	<ul style="list-style-type: none"> • 2 个通道的 16 位计数器 • 可选择 4 种时钟 (P0φ/8、P0φ/32、P0φ/128、P0φ/512)。 • 在比较匹配时, 可选择产生 DMA 传送请求或者中断请求。
实时时钟	<ul style="list-style-type: none"> • 内部时钟、日历功能、闹钟功能 • 1/64 秒的最大分辨率 (中断周期) 的内部 32.768kHz 晶体振荡器电路
FIFO 内置型串行通信接口	<ul style="list-style-type: none"> • 8 个通道 • 可选择时钟同步模式或者异步模式。 • 能同时进行发送和接收 (全双工)。 • 内置专用的波特率发生器。 • 内置 16 字节发送 FIFO 和 16 字节接收 FIFO。 • 调制解调器控制功能 (在异步模式中, SH7268: 通道 1, SH7269: 通道 1、5、7)
瑞萨串行外围接口	<ul style="list-style-type: none"> • 2 个通道 • SPI 运行 • 支持主控模式和从属模式。 • 能选择可编程的位长、时钟极性和时钟相位。 • 能按顺序进行传送。 • 可选择 MSB first 或者 LSB first。 • 最大传送率: 33.33Mbps
瑞萨四路串行外围接口	<ul style="list-style-type: none"> • 2 个通道 • 能连接对应多 I/O (Single/Dual/Quad) 的串行闪存。 • 能选择可编程的位长、时钟极性和时钟相位。 • 能按顺序进行传送。 • 可选择 MSB first 或者 LSB first。 • 最大传送率: 266.67Mbps
SPI 多 I/O 总线控制器	<ul style="list-style-type: none"> • 最多能连接 2 个对应多 I/O (Single/Dual/Quad) 的串行闪存。 • 外部地址空间读模式 (内置读高速缓存) • SPI 运行模式 • 可选择时钟极性和时钟相位。 • 可选择 MSB first 或者 LSB first。 • 最大传送率: 533.33Mbps (连接 2 个串行闪存时)
I ² C 总线接口 3	<ul style="list-style-type: none"> • SH7268: 2 个通道, SH7269: 4 个通道 • 内置主控模式和从属模式。
串行音频接口	<ul style="list-style-type: none"> • SH7268: 4 个通道, SH7269: 6 个通道的双向串行传送 • 能进行双工通信 (通道 0)。 • 支持多种串行音频格式。 • 支持主控功能和从属功能。 • 可编程的字时钟和位时钟生成功能 • 多通道格式功能 • 支持 8 位、16 位、18 位、20 位、22 位、24 位、32 位的数据格式。 • 内置用于发送和接收的 8 段 FIFO。 • 支持 TDM 模式。 • 支持不让 SSIWS 信号停止运行的 WS 继续模式。
带 FIFO 的时钟同步串行 I/O 【注】 SH7268 为非搭载	<ul style="list-style-type: none"> • 内置分别用于发送和接收的 32 位 × 16 段 FIFO。 • 支持 8 位单声道、16 位单声道和 16 位立体声音频的输入 / 输出。 • 能连接线性 / 音频 /A-Law 和 μ-Law CODEC 芯片。 • 支持主控功能和从属功能。

项目	特点
控制器局域网 【注】 根据产品型号，分为搭载产品和非搭载产品。	<ul style="list-style-type: none"> • 3 个通道 • 全部通道支持 TTCAN 级 1。 • 对应 Bosch 2.0B active。 • 缓冲器大小：发送 / 接收 ×31、接收 FIFO×1 • 用于增加 32 个通道输入的缓冲器的多个控制器局域网能将通道分配到 1 个总线。 • 能设定为发送或者接收的 31 个信箱
IEBus™ 控制器	<ul style="list-style-type: none"> • 支持 IEBus 的协议控制（层 2）。 半双工异步通信 多主控方式 广播通信功能 可选择不同传输速度的 3 种模式。 • 内置能最多连续发送和接收 128 字节数据（模式 2 的最大传输字节数）的缓冲器（双端口 RAM）。 工作频率： 在将 12MHz 和 12.58MHz（P0ϕ 或者 AUDIO_X1、AUDIO_X2）的时钟进行 1/2 分频后使用。 在将 18MHz 和 18.87MHz（P0ϕ 或者 AUDIO_X1、AUDIO_X2）的时钟进行 1/3 分频后使用。 在将 24MHz 和 25.16MHz（P0ϕ 或者 AUDIO_X1、AUDIO_X2）的时钟进行 1/4 分频后使用。 在将 30MHz 和 31.45MHz（P0ϕ 或者 AUDIO_X1、AUDIO_X2）的时钟进行 1/5 分频后使用。 在将 36MHz 和 37.74MHz（AUDIO_X1、AUDIO_X2）的时钟进行 1/6 分频后使用。 在将 42MHz 和 44.03MHz（AUDIO_X1、AUDIO_X2）的时钟进行 1/7 分频后使用 在将 48MHz（AUDIO_X1、AUDIO_X2）的时钟进行 1/8 分频后使用。
瑞萨 SPDIF 接口	<ul style="list-style-type: none"> • 符合 IEC60958 规格（只限于立体声和民用模式）。 • 采样频率：32kHz、44.1kHz、48kHz • 音频字长：16 ~ 24 位 / 样本 • 双相传号方式的编码 • 数据双缓冲器 • 带奇偶校验的串行数据 • 能同时发送和接收。 • 接收模块能自动检测 IEC61937 压缩模式的数据。
CD-ROM 解码器	<ul style="list-style-type: none"> • 支持 Mode0、Mode1、Mode2、Mode2 Form1、Mode2 Form2 共 5 种格式。 • 同步码的检测和保护 （保护：在无法检测时自动插入） • 解扰码 • ECC 校正 P、Q、PQ、QP 校正 最多能重复进行 3 次 PQ 校正和 QP 校正。 • EDC 校验 在 ECC 前后进行校验。 • Mode、Form 的自动检测 • 链路扇区的自动检测 • 缓冲数据控制 对于同步码以后的 CD-ROM 数据，以特定格式传送解扰码、ECC 校正和 EDC 校验后的数据。

项目	特点
NAND 闪存控制器	<ul style="list-style-type: none"> • 直接连接 NAND 型闪存的存储器接口 • 以扇区为单位的读写操作 • 命令存取模式和扇区存取模式 (512 字节数据 +16 字节管理代码) 共 2 种传送模式 • 有中断请求和直接存储器存取控制器传送请求。 • 支持 5 字节地址 (大于等于 2G 位) 的闪存。
USB2.0 主机 / 功能模块	<ul style="list-style-type: none"> • 符合 USB2.0 主机功能模块版本 2.0 标准。 • 支持 480Mbps、12Mbps 和 1.5Mbps 的传送率 (主机模式)。 • 支持 480Mbps 和 12Mbps 的传送率 (功能模式)。 • 内置用作通信缓冲器的 8K 字节 RAM。
数字视频解码器	<ul style="list-style-type: none"> • 图像输入 复合图像输入 (CVBS) • 用于输入图像信号的 A/D 转换器 选择 VIN1 引脚和 VIN2 引脚的输入。 内置同步端钳位。 内置 Programmable Gain Amp (PGA) (1.835dB ~ 8.023dB)。 内置 10 位精度流水线方式的 A/D 转换器。 • 同步分离 噪声消除 LPF、自动电平控制同步限幅器、水平 Auto Frequency Control (AFC)、垂直递减计数、隔行扫描判定、Auto Gain Control (AGC) / 峰值限制器控制 • Y/C 分离 支持 NTSC 2D、PAL 2D 和 SECAM 1D。 • 色度解扰码 支持 NTSC、PAL 和 SECAM。 消色器、Auto Color Control (ACC) 控制、TINT 校正、R-Y 轴校正 • 数字钳位 消隐脉冲钳位 (Y)、中心钳位 (Cb/Cr)、噪声检测 • 输出增益调整 对比度调整: 0 ~ 约 2 倍 色彩调整 (Cb/Cr 独立): 0 ~ 约 2 倍
视频显示控制器 4	<ul style="list-style-type: none"> • 图像输入接口: 从下面选择 1 个通道 BT601、BT656 格式 (NTSC/PAL) 输入: 输入时钟 27MHz/54MHz 数字引脚输入: YCbCr444、RGB888、RGB666、RGB565 数字引脚输入尺寸: 能设定的输入图像尺寸*: 水平 1024 个像素、垂直 1024 行 【注】* 取决于和连接设备的 AC 特性。 输入图像尺寸例子 SVGA (800×600)、WVGA (800×480)、VGA (640×480)、WQVGA (480×240)、QVGA (320×240、240×320) 输入由数字视频解码器解码的复合视频 (CVBS) 信号。 • 输入图像控制 通过水平噪声降低 (NR) 和矩阵运算进行亮度和对比度调整。 • 缩放控制 能以任意倍率对输入图像进行垂直方向和水平方向的放大或者缩小处理。 (也能放大图形) 放大率: 1 ~ 8 倍, 缩小率: 1/8 ~ 1 倍 内插方法: 可选择保持内插和线性内插。 2D IP 转换: 能通过指定每个 TOP/BOTTOM 场的初始相位来进行 2D IP 转换。

项目	特点
视频显示控制器 4	<ul style="list-style-type: none"> • 图像录像 输出像素格式: YCbCr422、RGB888、RGB565 输出场速率: 1/1、1/2、1/4、1/8 场 旋转功能: 在 YCbCr422 和 RGB565 时, 能进行水平镜像和 90/180/270 度旋转。 (但是, 不能和控制失真校正引擎一起控制) 能保存的最大图像尺寸: 和输入图像同倍 • 输出图像控制 黑色信号展宽功能: 根据 Y 信号的状态延伸黑色信号区域 增强器功能: 对 Y 信号进行 LTI (瞬态改善) 和清晰度 (轮廓增强) 处理 • 图层: 3 面 (其中 1 面和输入图像共享) 对应的输入像素格式 1bit/pixel: CLUT1、4bit/pixel: CLUT4、8bit/pixel: CLUT8 16bit/pixel: YCbCr422、RGB565、ARGB1555、ARGB4444 32bit/pixel: ARGB8888、RGB888 • 叠加功能 矩形区域 α 混合功能: 在设定的区域中 α, 根据透明度 α 进行输入图像的层 1、层 2 的混合 (有淡入和淡出功能)。 色键功能: 根据透明度 α 并且通过设定的 RGB 色彩和 CULT 值进行混合。 像素单位 α 功能: 根据透明度 α 并且以像素为单位进行混合。 • 显示屏输出控制 显示屏的输出校正功能: 亮度 / 对比度调整、灰度校正、显示屏抖动处理 TCON: 能通过 7 个垂直和垂直显示屏驱动信号, 输出用于驱动 LCD 显示屏的各种时序。 显示屏输出像素格式: RGB888、RGB666、RGB565、串行 RGB 输出图像尺寸: 能设定的输出图像尺寸*: 水平 1999 个像素、垂直 2035 行 【注】* 取决于和连接显示屏的 AC 特性。 输出图像尺寸例子 SVGA (800×600)、WVGA (800×480)、VGA (640×480)、WQVGA (480×240)、 QVGA (320×240、240×320)
失真校正引擎	<ul style="list-style-type: none"> • 参照作为 2 维纹理数据的视频捕捉数据, 对分割成 3 角形目标的任意形状进行纹理映射和绘图。 • 显示列表方式 • 绘图功能 纹理映射、双线性滤波器、自动坐标生成功能 (+ 相对坐标输入功能) • 指令系统 绘图指令: 绘制 3 角形的指令 TRI 控制指令: TRAP、INT、NOP、SYNCM、SYNCW、WTL、WTS • 绘图空间 目标坐标: $0 \leq X \leq 1023$, $0 \leq Y \leq 1023$ • 源坐标: $0 \leq u \leq 1023$, $0 \leq v \leq 1023$
显示输出比较单元	<ul style="list-style-type: none"> • 算出任意图形面的 CRC 码并且和已有的 CRC 码进行比较。 • 能通过从视频显示控制器 4 的 3 面图层、α 混合后的 1 面数据中选择的任意图形面来指定矩形区域。 • 像素格式 32bit/pixel: ARGB8888/RGB888/RGB666, 16bit/pixel: RGB565
OpenVG 瑞萨图形处理器	<ul style="list-style-type: none"> • 对应开放式 2D 向量图形 API 的 OpenVG。 • 能通过专用硬件和可编程着色来加速 OpenVG 的 Stage2 ~ Stage8 的各种处理。

项目	特点
JPEG 编解码器单元	<ul style="list-style-type: none"> 符合 JPEG 基准 * 的压缩解压方式。 *: 记载范围内的标准 运算精度: 符合 JPEG Part2 和 ISO-IEC10918-2 标准 像素格式 压缩: YCbCr422 解压: YCbCr422、YCbCr420 但是, 输出像素格式只限于 YCbCr422、ARGB8888 和 RGB565。 量化表: 内置 4 个表 霍夫曼表: 内置 4 个表 (2 个 AC 系数表和 2 个 DC 系数表) 对象标记: SOI、SOF0、SOS、DQT、DHT、DRI、RSTm、EOI 图像数据速率: 133.33MB/s (以 66.67MHz 运行时)
采样率转换器	<ul style="list-style-type: none"> 3 个通道 数据格式: 立体声 32 位 (L/R 各 16 位)、单声道 16 位 输入采样率: 8/11.025/12/16/22.05/24/32/44.1/48kHz 输出采样率: 32/44.1/48kHz、8kHz/16kHz (输入采样率为 44.1kHz 的情况)
音频发生器	<ul style="list-style-type: none"> 能通过 8 位 PWM 输出来调整音量。 可选择 4 种运行时钟 (P0ϕ/2、P0ϕ/4、P0ϕ/8、P0ϕ/16)。 能在不超过 1% 输出精度并且在 31Hz ~ 20kHz 范围内设定频率。 支持自动衰减功能。
SD 主机接口	<ul style="list-style-type: none"> 2 个通道 SD 存储器 /IO 卡接口 (1 位和 4 位 SD 总线) 错误校验功能: CRC7 (命令)、CRC16 (数据) 中断请求: 卡存取中断、SDIO 存取中断、卡检测中断 DMA 传送请求: 写 SD_BUF、读 SD_BUF 支持卡检测功能和写保护。
MMC 主机接口	<ul style="list-style-type: none"> 和 Multi Media Card (MMC) 的接口 数据总线: 支持 1 位、4 位、8 位 MMC 模式 中断请求: 卡检测中断、错误 / 超时中断、正常运行中断 DMA 传送请求: CE_DATA 写请求、CE_DATA 读请求 支持卡检测功能
通用输入 / 输出端口	<ul style="list-style-type: none"> SH7268: 101 个输入 / 输出引脚、4 个带漏极开路输出的输入引脚、6 个输入引脚 SH7269: 133 个输入 / 输出引脚、8 个带漏极开路输出的输入引脚、8 个输入引脚 能按位对输入 / 输出端口进行输入 / 输出切换。
A/D 转换器	<ul style="list-style-type: none"> 分辨率: 10 位 输入: SH7268: 6 个通道, SH7269: 8 个通道 能通过外部触发或者定时器触发来启动 A/D 转换。
马达控制 PWM 定时器	<ul style="list-style-type: none"> 内置 2 个通道的 10 位 PWM (有 8 个输出)。
用户断点控制器	<ul style="list-style-type: none"> 断点通道 \times 2 个通道 能将地址、数据值、存取格式和数据长度设定为断点条件。
用户调试接口	<ul style="list-style-type: none"> 支持 E10A 仿真器。 JTAG 标准引脚排列
内部 RAM	<ul style="list-style-type: none"> 内置用于高速存取的 64K 字节存储器 (16KB\times4 面)。 内置用于图像显示 / 录像和工作区的 2.5MB 大容量存储器 (其中 128KB 和用于保持数据的 RAM 共享)。 内置用于保持数据的 128K 字节存储器 (16KB\times2 面、32KB\times1 面、64KB\times1 面)。

项目	特点
引导模式	<ul style="list-style-type: none"> SH7268 / SH7269: 6 个引导模式 (引导模式 0 ~ 5) 引导模式 0: 从连接 $\overline{CS0}$ 空间的存储器 (总线宽度为 16 位) 引导 引导模式 1: 从连接 $\overline{CS0}$ 空间的存储器 (总线宽度为 32 位) 引导 引导模式 2: 从 NAND 闪存引导 引导模式 3: 从串行闪存引导 引导模式 4: 从 SD 控制器的内部 NAND 闪存引导 引导模式 5: 从 MMC 控制器的内部 NAND 闪存引导
电源电压	<ul style="list-style-type: none"> V_{CC}: 1.15 ~ 1.35V, PV_{CC}: 3.0 ~ 3.6V
封装	<ul style="list-style-type: none"> SH7268 208 引脚 QFP、28mm 平方、0.5mm 节距 JEITA Package Code: P-LQFP208-28×28-0.50 RENESAS Code: PLQP0208KB-A SH7269 (QFP) 256 引脚 QFP、28mm 平方、0.4mm 节距 JEITA Package Code: P-LQFP256-28×28-0.40 RENESAS Code: PLQP0256LB-A SH7269 (BGA) 272 引脚 BGA (16 散热锡球)、17mm 平方、0.80mm 节距 JEITA Package Code: P-LFBGA272-17×17-0.80 RENESAS Code: PRBG0272GA-A

1.2 产品一览表

表 1.2 产品一览表

产品分类	产品型号	控制器局域网	温度范围	封装
SH7268 群	R5S72680W266FP	非搭载	标准温度范围 (-20 ~ +85°C)	208 引脚 QFP
	R5S72681W266FP	搭载		
	R5S72680P266FP	非搭载	大温度范围 (-40 ~ +85°C)	
	R5S72681P266FP	搭载		
SH7269 群	R5S72690W266FP	非搭载	标准温度范围 (-20 ~ +85°C)	256 引脚 QFP
	R5S72691W266FP	搭载		
	R5S72690P266FP	非搭载	大温度范围 (-40 ~ +85°C)	
	R5S72691P266FP	搭载		
	R5S72690W266BG	非搭载	标准温度范围 (-20 ~ +85°C)	272 引脚 BGA
	R5S72691W266BG	搭载		
	R5S72690P266BG	非搭载	大温度范围 (-40 ~ +85°C)	
	R5S72691P266BG	搭载		

1.3 框图

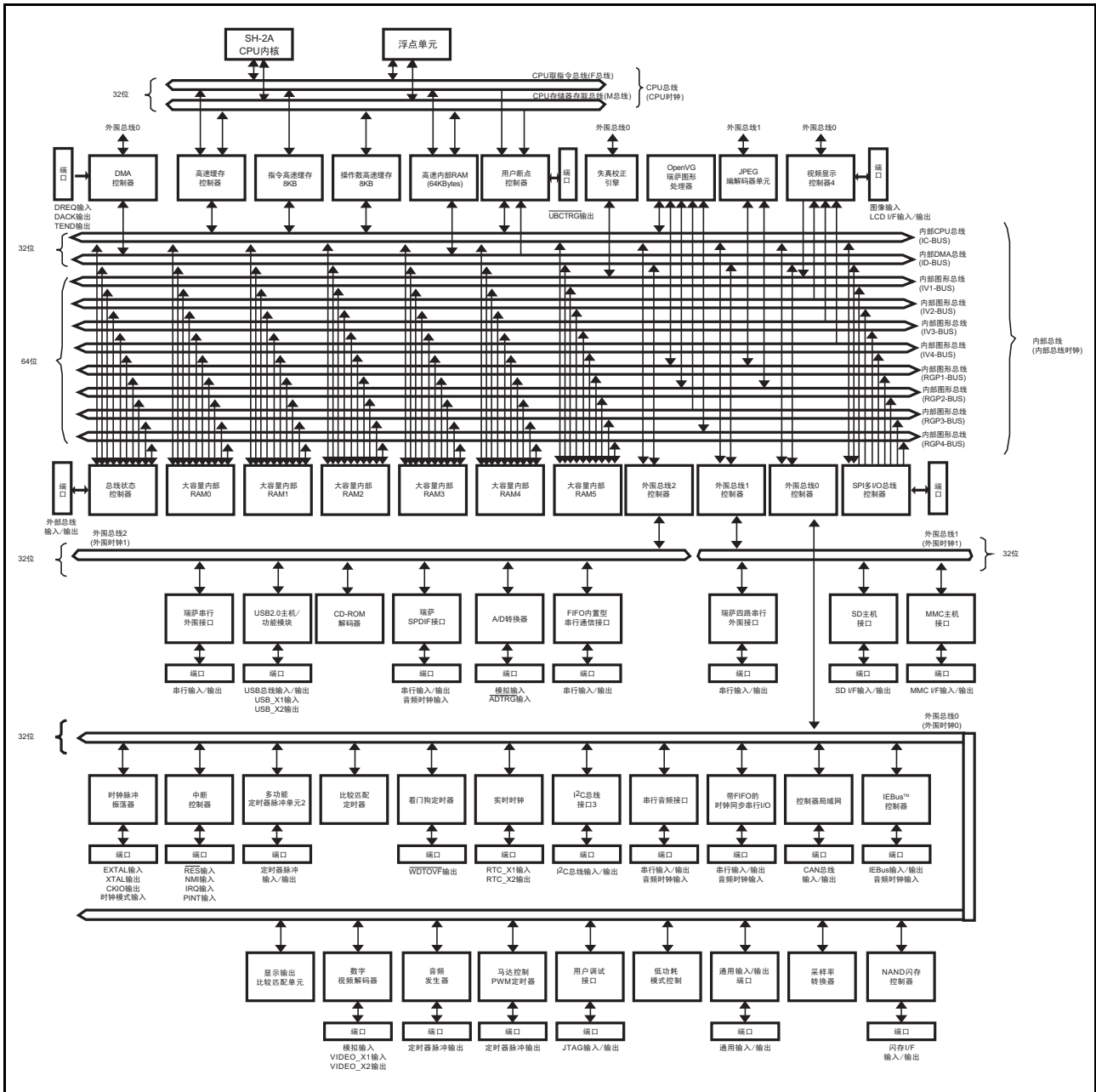


图 1.1 框图

1.4 引脚排列图

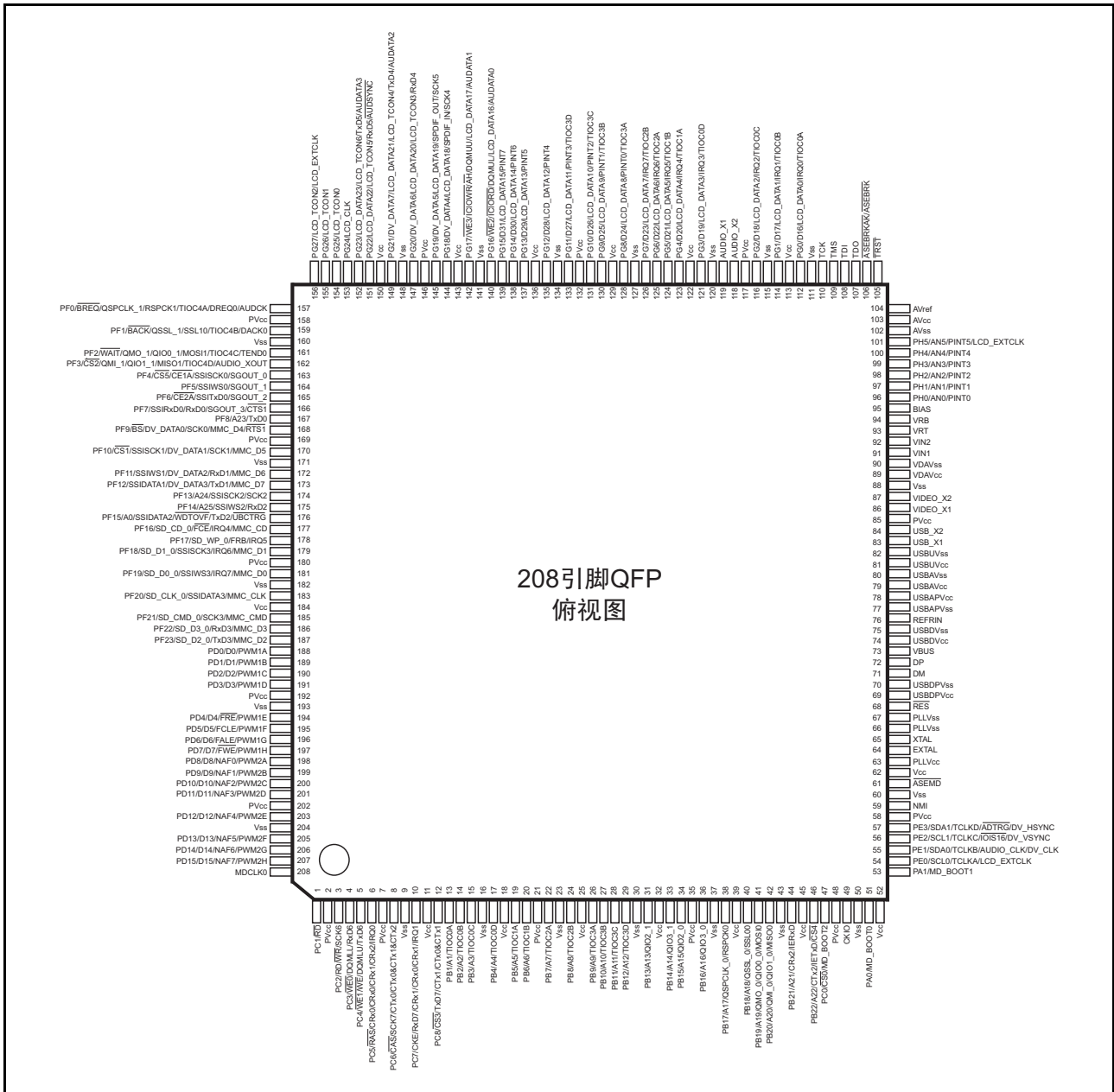
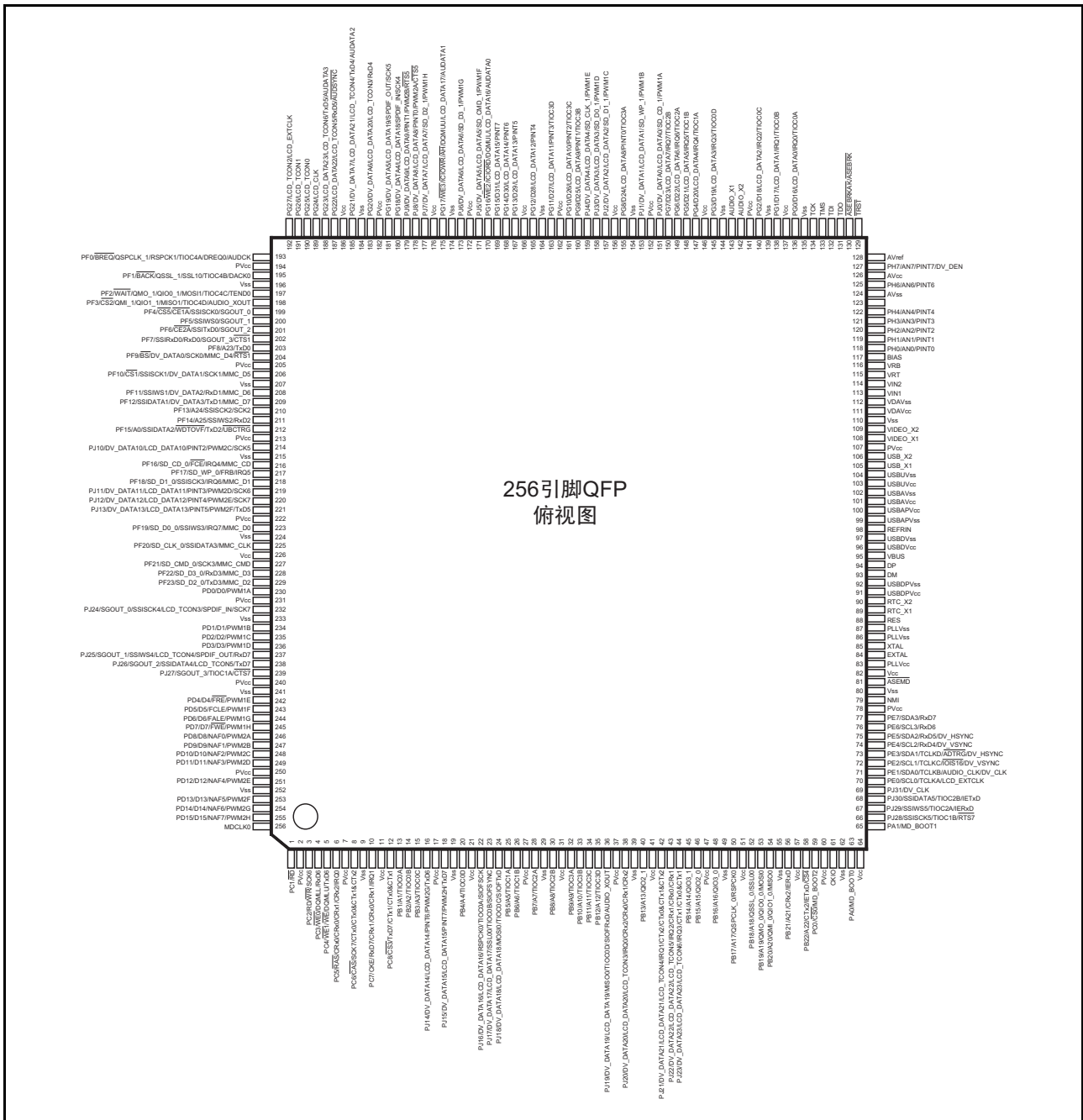


图 1.2 (1) SH7268 引脚排列图



256引脚QFP
俯视图

图 1.2 (2) SH7269 (QFP 版) 引脚排列图

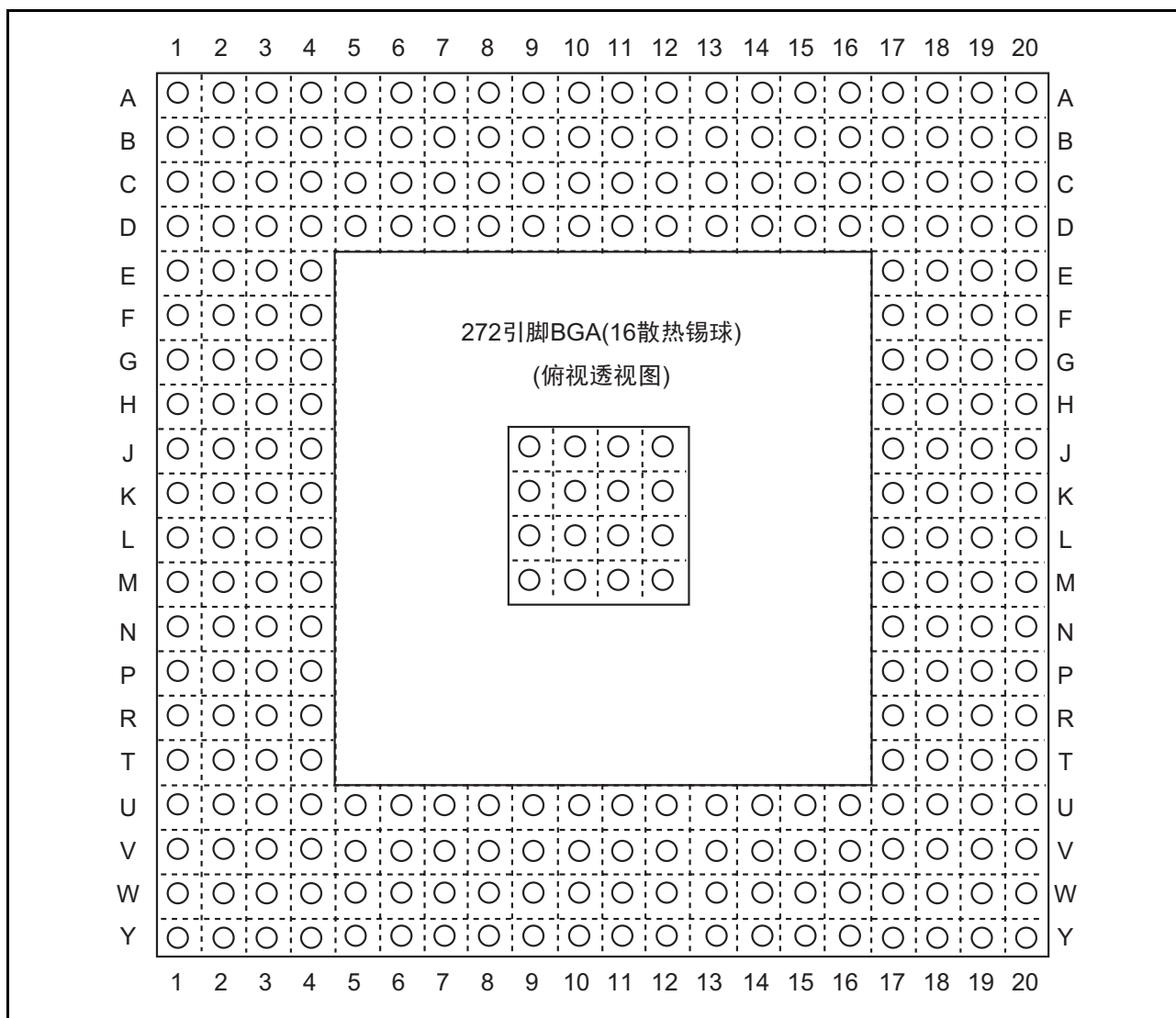


图 1.2 (3) SH7269 (BGA 版) 引脚排列图

1.5 引脚功能

表 1.3 引脚功能

分类	引脚名	输入 / 输出	名称	功能
电源	Vcc	输入	电源	电源引脚 必须将全部 Vcc 引脚连接系统电源。如果有开路引脚，就不运行。
	Vss	输入	接地	接地引脚 必须将全部 Vss 引脚连接系统电源（0V）。如果有开路引脚，就不运行。
	PVcc	输入	输入 / 输出电路电源	用于输入 / 输出引脚的电源引脚 必须将全部 PVcc 引脚连接系统电源。如果有开路引脚，就不运行。
	PLLvcc	输入	PLL 电源	用于内部 PLL 振荡器的电源
	PLLvss 【注】 SH7269 (BGA) 没有此引脚。	输入	PLL 接地	用于内部 PLL 振荡器的接地引脚
时钟	EXTAL	输入	外部时钟	连接晶体谐振器。EXTAL 引脚也能输入外部时钟。
	XTAL	输出	晶体	连接晶体谐振器。
	CKIO	输出	系统时钟输出	给外部设备提供系统时钟。
	AUDIO_CLK	输入	音频外部时钟	输入音频外部时钟。 在选择 AUDIO_X1 和 AUDIO_X2 的振荡时钟后输入到分频器。
	AUDIO_X1	输入	用于音频的晶体谐振器 / 外部时钟	连接用于音频的晶体谐振器。AUDIO_X1 引脚还能输入外部时钟。 在选择 AUDIO_CLK 输入时钟后输入到分频器。
	AUDIO_X2	输出		
	AUDIO_XOUT	输出	AUDIO_X1 时钟输出	将 AUDIO_X1 引脚的内部晶体振荡器信号或者外部时钟输入信号进行输出。
运行模式控制	MD_BOOT2、 MD_BOOT1、 MD_BOOT0	输入	模式设定	设定运行模式。在 $\overline{\text{RES}}$ 引脚有效期间以及在 $\overline{\text{RES}}$ 引脚无效到模式确定的期间，不能更改运行模式。
	MD_CLK0	输入	时钟模式设定	设定 SSCG 电路运行的 ON/OFF。
	$\overline{\text{ASEMD}}$	输入	ASE 模式	如果在 $\overline{\text{RES}}$ 引脚有效期间输入低电平，就进入 ASE 模式；如果输入高电平，就进入产品芯片模式。在 ASE 模式中，E10A-USB 仿真器功能有效。如果不使用仿真器功能，就必须固定为高电平。
系统控制	$\overline{\text{RES}}$	输入	上电复位	如果此引脚为低电平，就进入上电复位状态。
	$\overline{\text{WDTOVF}}$	输出	看门狗定时器上溢	看门狗定时器的上溢输出信号
	$\overline{\text{BREQ}}$	输入	总线权请求	在外部设备请求释放总线权时，将此引脚置为低电平。
	$\overline{\text{BACK}}$	输出	总线权请求应答	表示已将总线权释放给外部设备。输出 $\overline{\text{BREQ}}$ 信号的设备能通过接受 $\overline{\text{BACK}}$ 信号得知获得总线权。

分类	引脚名	输入 / 输出	名称	功能
中断	NMI	输入	非屏蔽中断	非屏蔽中断请求引脚 在不使用时，必须将此引脚固定为高电平。
	IRQ7 ~ IRQ0	输入	中断请求 7 ~ 0	可屏蔽中断请求引脚 可选择电平输入和边沿输入。在边沿输入的情况下，可选择上升沿、下降沿和双边沿。
	PINT7 ~ PINT0	输入	中断请求 7 ~ 0	可屏蔽中断请求引脚 只能选择电平输入。
地址总线	A25 ~ A0	输出	地址总线	输出地址。
数据总线	D31 ~ D0	输入 / 输出	数据总线	双向数据总线
总线控制	$\overline{CS5} \sim \overline{CS0}$	输出	片选 5 ~ 0	用于外部存储器或者设备的片选信号
	\overline{RD}	输出	读	表示从外部设备进行读操作。
	$\overline{RD}/\overline{WR}$	输出	读 / 写	读信号或者写信号
	\overline{BS}	输出	总线开始	总线周期开始信号
	\overline{AH}	输出	地址保持	多路复用 I/O 时的地址保持信号
	\overline{WAIT}	输入	等待	给存取外部空间时的总线周期插入等待周期的输入引脚
	$\overline{WE0}$	输出	字节指定	表示写外部存储器或者设备的数据 bit7 ~ 0。
	$\overline{WE1}$	输出	字节指定	表示写外部存储器或者设备的数据 bit15 ~ 8。
	$\overline{WE2}$	输出	字节指定	表示写外部存储器或者设备的数据 bit23 ~ 16。
	$\overline{WE3}$	输出	字节指定	表示写外部存储器或者设备的数据 bit31 ~ 24。
总线控制	DQMLL	输出	字节指定	在连接 SDRAM 时选择 D7 ~ D0。
	DQMLU	输出	字节指定	在连接 SDRAM 时选择 D15 ~ D8。
	DQMUL	输出	字节指定	在连接 SDRAM 时选择 D23 ~ D16。
	DQMUU	输出	字节指定	在连接 SDRAM 时选择 D31 ~ D24。
	\overline{RAS}	输出	RAS	在连接 SDRAM 时连接 \overline{RAS} 引脚。
	\overline{CAS}	输出	CAS	在连接 SDRAM 时连接 \overline{CAS} 引脚。
	CKE	输出	CK 允许	在连接 SDRAM 时连接 CKE 引脚。
	$\overline{CE1A}$	输出	PCMCIA 卡选择低位	连接 PCMCIA 卡选择信号 D7 ~ D0。
	$\overline{CE2A}$	输出	PCMCIA 卡选择高位	连接 PCMCIA 卡选择信号 D15 ~ D8。
	\overline{ICIWR}	输出	PCMCIA 写选通	连接 PCMCIA I/O 写选通信号。
	\overline{ICIORD}	输出	PCMCIA 读选通	连接 PCMCIA I/O 读选通信号。
	\overline{WE}	输出	PCMCIA 存储器写选通	连接 PCMCIA 存储器写选通信号。
	$\overline{IOIS16}$	输入	PCMCIA 动态总线调整	表示 PCMCIA 的 16 位 I/O。
直接存储器存取控制器	DREQ0	输入	DMA 传送请求	外部 DMA 传送请求的输入引脚
	DACK0	输出	DMA 传送请求接受	请求接受外部 DMA 传送请求的输出引脚
	TEND0	输出	DMA 传送结束输出	DMA 传送结束的输出信号

分类	引脚名	输入 / 输出	名称	功能
多功能定时器 脉冲单元 2	TCLKA、TCLKB、 TCLKC、TCLKD	输入	定时器时钟输入	定时器的外部时钟输入引脚
	TIOC0A、TIOC0B、 TIOC0C、TIOC0D	输入 / 输出	输入捕捉 / 输出 比较 (通道 0)	TGRA_0 ~ TGRD_0 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 输出引脚
	TIOC1A、TIOC1B	输入 / 输出	输入捕捉 / 输出 比较 (通道 1)	TGRA_1 和 TGRB_1 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 输出引脚
	TIOC2A、TIOC2B	输入 / 输出	输入捕捉 / 输出 比较 (通道 2)	TGRA_2 和 TGRB_2 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 输出引脚
	TIOC3A、TIOC3B、 TIOC3C、TIOC3D	输入 / 输出	输入捕捉 / 输出 比较 (通道 3)	TGRA_3 ~ TGRD_3 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 输出引脚
	TIOC4A、TIOC4B、 TIOC4C、TIOC4D	输入 / 输出	输入捕捉 / 输出 比较 (通道 4)	TGRA_4 和 TGRB_4 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 输出引脚
实时时钟	RTC_X1	输入	用于实时时钟的 晶体谐振器 / 外 部时钟	连接 32.768kHz 的晶体谐振器。RTC_X1 引 脚也能输入外部时钟。
	RTC_X2	输出		
FIFO 内置型 串行通信接口	TxD7 ~ TxD0	输出	发送数据	数据的输出引脚。
	RxD7 ~ RxD0	输入	接收数据	数据的输入引脚
	SCK7 ~ SCK0	输入 / 输出	串行时钟	时钟的输入 / 输出引脚
	RTS7、RTS5、RTS1	输出	发送请求	调制解调器的控制引脚 SH7268 只有 RTS1。
	CTS7、CTS5、CTS1	输入	可发送	调制解调器的控制引脚 SH7268 只有 CTS1。
瑞萨串行外围 接口	MOSI1、MOSI0	输入 / 输出	数据	数据的输入 / 输出引脚
	MISO1、MISO0	输入 / 输出	数据	数据的输入 / 输出引脚
	RSPCK1、RSPCK0	输入 / 输出	时钟	时钟的输入 / 输出引脚
	SSL00、SSL10	输入 / 输出	从属选择	从属选择的输入 / 输出引脚
瑞萨四路串行 外围接口	QMO_0/QIO0_0、 QMO_1/QIO0_1	输入 / 输出	数据	数据的输入 / 输出引脚
	QMI_0/QIO0_0、 QMI_1/QIO0_1	输入 / 输出	数据	数据的输入 / 输出引脚
	QIO2_0、QIO2_1	输入 / 输出	数据	数据的输入 / 输出引脚
	QIO3_0、QIO3_1	输入 / 输出	数据	数据的输入 / 输出引脚
	QSPCLK_0、 QSPCLK_1	输出	时钟	时钟的输出引脚
	QSSL_0、QSSL_1	输出	从属选择	从属选择的输出引脚

分类	引脚名	输入 / 输出	名称	功能
SPI 多 I/O 总线 控制器	SPBCLK	输出	时钟	时钟的输出引脚
	SPBSSL	输出	从属选择	从属选择的输出引脚
	SPBMO_0/ SPBIO0_0、 SPBBI_0/SPBIO1_0、 SPBIO2_0、 SPBIO3_0	输入 / 输出	数据	数据的输入 / 输出引脚
	SPBMO_1/ SPBIO0_1、 SPBBI_1/SPBIO1_1、 SPBIO2_1、 SPBIO3_1	输入 / 输出	数据	数据的输入 / 输出引脚
I ² C 总线接口 3	SCL3 ~ SCL0	输入 / 输出	串行时钟引脚	串行时钟的输入 / 输出引脚 SH7268 只有 SCL1 和 SCL0。
	SDA3 ~ SDA0	输入 / 输出	串行数据引脚	串行数据的输入 / 输出引脚 SH7268 只有 SDA1 和 SDA0。
串行音频接口	SSITxD0	输出	数据输出	串行数据的输出引脚
	SSIRxD0	输入	数据输入	串行数据的输入引脚
	SSIDATA5 ~ SSIDATA1	输入 / 输出	数据输入 / 输出	串行数据的输入 / 输出引脚 SH7268 只有 SSIDATA3 ~ SSIDATA1。
	SSISCK5 ~ SSISCK0	输入 / 输出	时钟输入 / 输出	串行时钟的输入 / 输出引脚 SH7268 只有 SSISCK3 ~ SSISCK0。
	SSIWS5 ~ SSIWS0	输入 / 输出	时钟 LR 输入 / 输出	字选择的输入 / 输出引脚 SH7268 只有 SSIWS3 ~ SSIWS0。
带 FIFO 的时钟 周期串行 I/O	SIOFTxD	输出	输出数据	数据的输出引脚
	SIOFRxD	输入	输入数据	数据的接收引脚
	SIOFSCK	输入 / 输出	输入 / 输出时钟	时钟的输入 / 输出引脚
	SIOFSYNC	输入 / 输出	输入 / 输出片选 选择	片选的输入 / 输出引脚
控制器局域网	CTx2 ~ CTx0	输出	CAN 总线发送 数据	CAN 总线的发送引脚
	CRx2 ~ CRx0	输入	CAN 总线接收 数据	CAN 总线的接收引脚
IEBus™ 控制器	IETxD	输出	IEBus™ 控制器 发送数据	IEBus™ 控制器的发送数据输出引脚
	IERxD	输入	IEBus™ 控制器 接收数据	IEBus™ 控制器的接收数据输入引脚
瑞萨 SPDIF 接口	SPDIF_OUT	输出	输出数据	发送数据的输出引脚
	SPDIF_IN	输入	输入数据	接收数据的输入引脚

分类	引脚名	输入 / 输出	名称	功能
NAND 闪存 控制器	FALE	输出	闪存地址锁存允许	此引脚在输出地址时为有效电平，在输入或者输出数据时为无效电平。
	$\overline{\text{FRE}}$	输出	闪存读允许	在下降沿读数据。
	$\overline{\text{FCE}}$	输出	闪存芯片允许	允许连接本 LSI 的闪存。
	FCLE	输出	闪存命令锁存允许	此引脚在输出命令时为有效电平。
	FRB	输入	闪存就绪 / 忙	高电平表示就绪状态，低电平表示忙状态。
	$\overline{\text{FWE}}$	输出	闪存写允许	在下降沿，闪存锁存命令、地址和数据。
	NAF7 ~ NAF0	输入 / 输出	闪存数据	数据的输入 / 输出引脚
USB2.0 主机 / 功能模块	DP	输入 / 输出	USB2.0 主机 / 功能模块 D+ 数据	USB2.0 主机 / 功能模块总线的 D+ 数据
	DM	输入 / 输出	USB2.0 主机 / 功能模块 D- 数据	USB2.0 主机 / 功能模块总线的 D- 数据
	VBUS	输入	VBUS 输入	必须连接 USB2.0 主机 / 功能模块总线的 Vbus。
	REFRIN	输入	基准输入	必须通过 5.6kΩ±1% 电阻连接 USBAPVSS。
	USB_X1	输入	USB2.0 主机 / 功能模块的晶体谐振器 / 外部时钟	连接用于 USB2.0 主机 / 功能模块的晶体谐振器。USB_X1 引脚也能输入外部时钟。
	USB_X2	输出		
	USBAPVcc	输入	收发器部模拟引脚电源	用于引脚的电源
	USBAPVss 【注】 SH7269 (BGA) 没有此引脚。	输入	收发器部模拟引脚接地	用于引脚的接地
	USBDPVcc 【注】 SH7269 (BGA) 没有此引脚。	输入	收发器部数字引脚电源	用于引脚的电源
	USBDPVss 【注】 SH7269 (BGA) 没有此引脚。	输入	收发器部数字引脚接地	用于引脚的接地
	USBAVcc	输入	收发器部模拟核电源	用于内核的电源
	USBAVss 【注】 SH7269 (BGA) 没有此引脚。	输入	收发器部模拟内核接地	用于内核的接地
	USBDVcc 【注】 SH7269 (BGA) 没有此引脚。	输入	收发器部数字内核电源	用于内核的电源
	USBDVss 【注】 SH7269 (BGA) 没有此引脚。	输入	收发器部数字内核接地	用于内核的接地
	USBUVcc 【注】 SH7269 (BGA) 没有此引脚。	输入	用于 USB2.0 主机 / 功能模块的 480MHz 电源	480MHz 工作部电源
USBUVss 【注】 SH7269 (BGA) 没有此引脚。	输入	用于 USB2.0 主机 / 功能模块的 480MHz 接地	480MHz 工作部接地	

分类	引脚名	输入 / 输出	名称	功能
数字视频解码器	VIN1、VIN2	输入	复合图像信号 (CVBS) 输入	复合图像信号 (CVBS) 的输入引脚
	VIDEO_X1	输入	用于数字视频解码器的晶体谐振器 / 外部时钟	连接用于数字视频解码器的晶体谐振器。VIDEO_X1 引脚也能输入外部时钟。
	VIDEO_X2	输出		
	VRT	输出	TOP 基准电压	用于输入图像信号的 A/D 转换器的 TOP 基准电压引脚 必须通过 0.1 μ F 电容器连接 VDAV _{SS} 。
	VRB	输出	BOTTOM 基准电压	用于输入图像信号的 A/D 转换器的 BOTTOM 基准电压引脚 必须通过 0.1 μ F 的电容器连接 VDAV _{SS} 。
	BIAS	输入	基准电压	用于输入图像信号的 A/D 转换器的基准电压引脚 必须通过 24k Ω ±1% 电阻连接 VDAV _{SS} 。
	VDAV _{CC}	输入	模拟电源	用于输入图像信号的 A/D 转换器的电源引脚
	VDAV _{SS}	输入	模拟接地	用于输入图像信号的 A/D 转换器的接地引脚
视频显示控制器 4	LCD_DATA23 ~ LCD_DATA0	输出	输出数据	用于显示屏的数据输出引脚
	LCD_TCON6 ~ LCD_TCON0	输出	显示屏时序调整输出	用于调整显示屏时序的输出引脚
	LCD_CLK	输出	显示屏时钟	显示屏时钟的输出引脚
	LCD_EXTCLK	输入	显示屏时钟源	显示屏时钟源的输入引脚
	DV_DATA23 ~ 0	输入	输入数据	图形数据的输入引脚
	DV_VSYNC	输入	VSYNC 输入	VSYNC 输入引脚
	DV_HSYNC	输入	HSYNC 输入	HSYNC 输入引脚
	DV_CLK	输入	输入时钟	图形数据的时钟输入引脚
音频发生器	SGOUT3 ~ SGOUT0	输出	音频发生器输出	音频发生器的输出引脚
SD 主机接口	SD_CLK_0 SD_CLK_1	输出	SD 时钟	SD 时钟的输出引脚 SH7268 只有 SD_CLK_0。
	SD_CMD_0 SD_CMD_1	输入 / 输出	SD 命令	SD 命令输出 / 应答输入信号 SH7268 只有 SD_CMD_0。
	SD_D3_0 ~ SD_D0_0 SD_D3_1 ~ SD_D0_1	输入 / 输出	SD 数据	SD 数据总线信号 SH7268 只有 SD_D3_0 ~ SD_D0_0。
	SD_CD_0 SD_CD_1	输入	SD 卡检测	SD 卡检测引脚 SH7268 只有 SD_CD_0。
	SD_WP_0 SD_WP_1	输入	SD 写保护	SD 写保护信号 SH7268 只有 SD_WP_0。
MMC 主机接口	MMC_CLK	输出	MMC 时钟	MMC 时钟的输出引脚
	MMC_CMD	输入 / 输出	MMC 命令	MMC 命令输出 / 应答输入信号
	MMC_D7 ~ MMC_D0	输入 / 输出	MMC 数据	MMC 数据总线信号
	MMC_CD	输入	MMC 卡检测	MMC 卡检测引脚
马达控制 PWM 定时器	PWM1H ~ PWM1A PWM2H ~ PWM2A	输出	定时器输出	PWM 输出引脚

分类	引脚名	输入 / 输出	名称	功能
A/D 转换器	AN7 ~ AN0	输入	模拟输入引脚	模拟输入引脚 SH7268 只有 AN5 ~ AN0。
	ADTRG	输入	A/D 转换触发输入	用于开始 A/D 转换的外部触发输入引脚
	AVcc	输入	模拟电源	A/D 转换器的电源引脚
	AVss	输入	模拟接地	A/D 转换器的接地引脚
	AVref	输入	模拟基准电压	A/D 转换器的基准电压引脚
通用输入 / 输出端口	PA1、PA0、 PB22 ~ PB1、 PC8 ~ PC0、 PD15 ~ PD0、 PF23 ~ PF0、 PG27 ~ PG0、 PJ31 ~ PJ0	输入 / 输出	通用端口	SH7268 有 101 个通用输入 / 输出端口引脚， SH7269 有 133 个通用输入 / 输出端口引脚。 SH7268 只有 PA1、PA0、PB22 ~ PB1、 PC8 ~ PC0、PD15 ~ PD0、PF23 ~ PF0、 PG27 ~ PG0。
	PE7 ~ PE0	输入 / 输出	通用端口	8 个带漏极开路输出的输入端口引脚 SH7268 只有 PE3 ~ 0。
	PH7 ~ PH0	输入	通用端口	8 个通用输入端口引脚 SH7268 只有 PH5 ~ 0。
用户调试接口	TCK	输入	测试时钟	测试时钟的输入引脚
	TMS	输入	测试模式选择	测试模式选择信号的输入引脚
	TDI	输入	测试数据输入	指令和数据的串行输入引脚
	TDO	输出	测试数据输出	指令和数据的串行输出引脚
	TRST	输入	测试复位	初始化信号的输入引脚
仿真器接口	AUDATA3 ~ AUDATA0	输出	数据	转移目标 / 转移源的地址输出引脚
	AUDCK	输出	时钟	同步时钟的输出引脚
	AUDSYNC	输出	同步信号	数据起始位置识别信号的输出引脚
	ASEBRKAK	输出	暂停模式应答	表示 E10A-USB 仿真器已进入暂停模式。
	ASEBRK	输入	暂停请求	E10A-USB 仿真器暂停的输入引脚
用户断点控制器	UBCTRG	输出	用户断点触发输出	用户断点条件成立的触发输出引脚

1.6 引脚一览表

表 1.4 引脚一览表

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 1		功能 2		功能 3		功能 4	
			引脚名	I/O	引脚名	I/O	引脚名	I/O	引脚名	I/O
1	1	C1	PC1	I(s)/O	\overline{RD}	O	—	—	—	—
2	2	—	PVcc							
3	3	E2	PC2	I(s)/O	RD/\overline{WR}	O	SCK6	I(s)/O	—	—
4	4	E3	PC3	I(s)/O	$\overline{WE0}/DQMLL$	O	RxD6	I(s)	—	—
5	5	F4	PC4	I(s)/O	$\overline{WE1}/\overline{WE}/DQMLU$	O	TxD6	O	—	—
6	6	D1	PC5	I(s)/O	RAS	O	—	—	CRx0	I(s)
7	7	—	PVcc							
8	8	F2	PC6	I(s)/O	CAS	O	SCK7	I(s)/O	CTx0	O
9	9	—	VSS							
10	10	F3	PC7	I(s)/O	CKE	O	RxD7	I(s)	CRx1	I(s)
11	11	—	Vcc							
12	12	E1	PC8	I(s)/O	$\overline{CS3}$	O	TxD7	O	CTx1	O
13	13	G4	PB1	I(s)/O	A1	O	TIOC0A	I(s)/O	—	—
14	14	G2	PB2	I(s)/O	A2	O	TIOC0B	I(s)/O	—	—
15	15	G3	PB3	I(s)/O	A3	O	TIOC0C	I(s)/O	—	—

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 5		功能 6		功能 7		简易电路图 图 1.3
			引脚名	I/O	引脚名	I/O	引脚名	I/O	
1	1	C1	—	—	—	—	—	—	(7)
2	2	—							
3	3	E2	—	—	—	—	—	—	(7)
4	4	E3	—	—	—	—	—	—	(7)
5	5	F4	—	—	—	—	—	—	(7)
6	6	D1	CRx0/CRx1/CRx2	I(s)	IRQ0	I(s)	—	—	(7)
7	7	—							
8	8	F2	CTx0&CTx1&CTx2	O	—	—	—	—	(7)
9	9	—							
10	10	F3	CRx0/CRx1	I(s)	IRQ1	I(s)	—	—	(7)
11	11	—							
12	12	E1	CTx0&CTx1	O	—	—	—	—	(7)
13	13	G4	—	—	—	—	—	—	(7)
14	14	G2	—	—	—	—	—	—	(7)
15	15	G3	—	—	—	—	—	—	(7)

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 1		功能 2		功能 3		功能 4	
			引脚名	I/O	引脚名	I/O	引脚名	I/O	引脚名	I/O
—	16	F1	PJ14	I(s)/O	DV_DATA14	I(s)	LCD_DATA14	O	PINT6	I(s)
—	17	—	PVcc							
—	18	H2	PJ15	I(s)/O	DV_DATA15	I(s)	LCD_DATA15	O	PINT7	I(s)
16	19	—	Vss							
17	20	H3	PB4	I(s)/O	A4	O	TIOC0D	I(s)/O	—	—
18	21	—	Vcc							
—	22	G1	PJ16	I(s)/O	DV_DATA16	I(s)	LCD_DATA16	O	RSPCK0	I(s)/O
—	23	J3	PJ17	I(s)/O	DV_DATA17	I(s)	LCD_DATA17	O	SSL00	I(s)/O
—	24	J2	PJ18	I(s)/O	DV_DATA18	I(s)	LCD_DATA18	O	MOSI0	I(s)/O
19	25	H1	PB5	I(s)/O	A5	O	TIOC1A	I(s)/O	—	—
20	26	K2	PB6	I(s)/O	A6	O	TIOC1B	I(s)/O	—	—
21	27	—	PVcc							
22	28	J1	PB7	I(s)/O	A7	O	TIOC2A	I(s)/O	—	—
23	29	—	Vss							
24	30	K3	PB8	I(s)/O	A8	O	TIOC2B	I(s)/O	—	—

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 5		功能 6		功能 7		简易电路图 图 1.3
			引脚名	I/O	引脚名	I/O	引脚名	I/O	
—	16	F1	PWM2G	O	TxD6	O	—	—	(7)
—	17	—							
—	18	H2	PWM2H	O	TxD7	O	—	—	(7)
16	19	—							
17	20	H3	—	—	—	—	—	—	(7)
18	21	—							
—	22	G1	TIOC0A	I(s)/O	SIOFCK	I(s)/O	—	—	(7)
—	23	J3	TIOC0B	I(s)/O	SIOFSYNC	I(s)/O	—	—	(7)
—	24	J2	TIOC0C	I(s)/O	SIOFTxD	O	—	—	(7)
19	25	H1	—	—	—	—	—	—	(7)
20	26	K2	—	—	—	—	—	—	(7)
21	27	—							
22	28	J1	—	—	—	—	—	—	(7)
23	29	—							
24	30	K3	—	—	—	—	—	—	(7)

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 1		功能 2		功能 3		功能 4	
			引脚名	I/O	引脚名	I/O	引脚名	I/O	引脚名	I/O
25	31	—	Vcc							
26	32	K1	PB9	I(s)/O	A9	O	TIOC3A	I(s)/O	—	—
27	33	L1	PB10	I(s)/O	A10	O	TIOC3B	I(s)/O	—	—
28	34	L2	PB11	I(s)/O	A11	O	TIOC3C	I(s)/O	—	—
29	35	M1	PB12	I(s)/O	A12	O	TIOC3D	I(s)/O	—	—
—	36	L3	PJ19	I(s)/O	DV_DATA19	I(s)	LCD_DATA19	O	MISO0	I(s)/O
—	37	—	PVcc							
—	38	M2	PJ20	I(s)/O	DV_DATA20	I(s)	LCD_DATA20	O	LCD_TCON3	O
30	39	—	Vss							
31	40	N1	PB13	I(s)/O	A13	O	QIO2_1	I(s)/O	—	—
32	41	—	Vcc							
—	42	N2	PJ21	I(s)/O	DV_DATA21	I(s)	LCD_DATA21	O	LCD_TCON4	O
—	43	M3	PJ22	I(s)/O	DV_DATA22	I(s)	LCD_DATA22	O	LCD_TCON5	O
—	44	P1	PJ23	I(s)/O	DV_DATA23	I(s)	LCD_DATA23	O	LCD_TCON6	O
33	45	P2	PB14	I(s)/O	A14	O	QIO3_1	I(s)/O	—	—

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 5		功能 6		功能 7		简易电路图 图 1.3
			引脚名	I/O	引脚名	I/O	引脚名	I/O	
25	31	—							
26	32	K1	—	—	—	—	—	—	(7)
27	33	L1	—	—	—	—	—	—	(7)
28	34	L2	—	—	—	—	—	—	(7)
29	35	M1	—	—	—	—	—	—	(7)
—	36	L3	TIOC0D	I(s)/O	SIOFRxD	I(s)	AUDIO_XOUT	O	(7)
—	37	—							
—	38	M2	IRQ0	I(s)	CRx2	I(s)	CRx0/CRx1/CRx2	I(s)	(7)
30	39	—							
31	40	N1	—	—	—	—	SPBIO2_1	I(s)/O	(7)
32	41	—							
—	42	N2	IRQ1	I(s)	CTx2	O	CTx0&CTx1&CTx2	O	(7)
—	43	M3	IRQ2	I(s)	CRx1	I(s)	CRx0/CRx1	I(s)	(7)
—	44	P1	IRQ3	I(s)	CTx1	O	CTx0&CTx1	O	(7)
33	45	P2	—	—	—	—	SPBIO3_1	I(s)/O	(7)

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 1		功能 2		功能 3		功能 4	
			引脚名	I/O	引脚名	I/O	引脚名	I/O	引脚名	I/O
34	46	R1	PB15	I(s)/O	A15	O	QIO2_0	I(s)/O	—	—
35	47	—	PVcc							
36	48	R2	PB16	I(s)/O	A16	O	QIO3_0	I(s)/O	—	—
37	49	—	Vss							
38	50	T1	PB17	I(s)/O	A17	O	QSPCLK_0	O	RSPCK0	I(s)/O
39	51	—	Vcc							
40	52	P3	PB18	I(s)/O	A18	O	QSSL_0	O	SSL00	I(s)/O
41	53	P4	PB19	I(s)/O	A19	O	QMO_0/QIO0_0	I(s)/O	MOSI0	I(s)/O
42	54	T2	PB20	I(s)/O	A20	O	QMI_0/QIO1_0	I(s)/O	MISO0	I(s)/O
43	55	—	Vss							
44	56	R3	PB21	I(s)/O	A21	O	CRx2	I(s)	IERxD	I(s)
45	57	—	Vcc							
46	58	U2	PB22	I(s)/O	A22	O	CTx2	O	IETxD	O
47	59	T3	PC0	I(s)/O	$\overline{CS0}$	O	MD_BOOT2	I(s)	—	—
48	60	—	PVcc							

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 5		功能 6		功能 7		简易电路图 图 1.3
			引脚名	I/O	引脚名	I/O	引脚名	I/O	
34	46	R1	—	—	—	—	SPBIO2_0	I(s)/O	(7)
35	47	—							
36	48	R2	—	—	—	—	SPBIO3_0	I(s)/O	(7)
37	49	—							
38	50	T1	—	—	—	—	SPBCLK	O	(7)
39	51	—							
40	52	P3	—	—	—	—	SPBSSL	O	(7)
41	53	P4	—	—	—	—	SPBMO_0/SPBIO0_0	I(s)/O	(7)
42	54	T2	—	—	—	—	SPBMI_0/SPBIO1_0	I(s)/O	(7)
43	55	—							
44	56	R3	—	—	—	—	—	—	(7)
45	57	—							
46	58	U2	$\overline{CS4}$	O	—	—	—	—	(7)
47	59	T3	—	—	—	—	—	—	(7)
48	60	—							

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 1		功能 2		功能 3		功能 4	
			引脚名	I/O	引脚名	I/O	引脚名	I/O	引脚名	I/O
49	61	V1	CKIO	O	—	—	—	—	—	—
50	62	—	V _{SS}							
51	63	R4	PA0	I(s)/O	MD_BOOT0	I(s)	—	—	—	—
52	64	—	V _{CC}							
53	65	U5	PA1	I(s)/O	MD_BOOT1	I(s)	—	—	—	—
—	66	Y2	PJ28	I(s)/O	—	—	SSISCK5	I(s)/O	—	—
—	67	W3	PJ29	I(s)/O	—	—	SSIWS5	I(s)/O	—	—
—	68	V4	PJ30	I(s)/O	—	—	SSIDATA5	I(s)/O	—	—
—	69	Y3	PJ31	I(s)/O	DV_CLK	I(s)	—	—	—	—
54	70	W4	PE0	I(s)/O(o)	SCL0	I(s)/O(o)	TCLKA	I(s)	LCD_EXTCLK	I(s)
55	71	V6	PE1	I(s)/O(o)	SDA0	I(s)/O(o)	TCLKB	I(s)	AUDIO_CLK	I(s)
56	72	Y4	PE2	I(s)/O(o)	SCL1	I(s)/O(o)	TCLKC	I(s)	$\overline{\text{IOIS16}}$	I(s)
57	73	V5	PE3	I(s)/O(o)	SDA1	I(s)/O(o)	TCLKD	I(s)	$\overline{\text{ADTRG}}$	I(s)
—	74	V7	PE4	I(s)/O(o)	SCL2	I(s)/O(o)	RxD4	I(s)	DV_VSYNC	I(s)
—	75	W5	PE5	I(s)/O(o)	SDA2	I(s)/O(o)	RxD5	I(s)	DV_HSYNC	I(s)

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 5		功能 6		功能 7		简易电路图 图 1.3
			引脚名	I/O	引脚名	I/O	引脚名	I/O	
49	61	V1	—	—	—	—	—	—	(5)
50	62	—							
51	63	R4	—	—	—	—	—	—	(7)
52	64	—							
53	65	U5	—	—	—	—	—	—	(7)
—	66	Y2	TIOC1B	I(s)/O	$\overline{\text{RTS7}}$	I(s)/O	—	—	(7)
—	67	W3	TIOC2A	I(s)/O	IERxD	I(s)	—	—	(7)
—	68	V4	TIOC2B	I(s)/O	IETxD	O	—	—	(7)
—	69	Y3	—	—	—	—	—	—	(7)
54	70	W4	—	—	—	—	—	—	(9)
55	71	V6	DV_CLK	I(s)	—	—	—	—	(9)
56	72	Y4	DV_VSYNC	I(s)	—	—	—	—	(9)
57	73	V5	DV_HSYNC	I(s)	—	—	—	—	(9)
—	74	V7	—	—	—	—	—	—	(9)
—	75	W5	—	—	—	—	—	—	(9)

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 1		功能 2		功能 3		功能 4	
			引脚名	I/O	引脚名	I/O	引脚名	I/O	引脚名	I/O
—	76	W7	PE6	I(s)/O(o)	SCL3	I(s)/O(o)	RxD6	I(s)	—	—
—	77	U7	PE7	I(s)/O(o)	SDA3	I(s)/O(o)	RxD7	I(s)	—	—
58	78	—	PV _{CC}							
59	79	V8	NMI	I(s)	—	—	—	—	—	—
60	80	—	V _{SS}							
61	81	W6	$\overline{\text{ASEMD}}$	I(s)	—	—	—	—	—	—
62	82	—	V _{CC}							
63	83	U6	PLL _{VCC}							
64	84	Y5	EXTAL	I	—	—	—	—	—	—
65	85	Y6	XTAL	O	—	—	—	—	—	—
66	86	—	PLL _{VSS}							
67	87	—	PLL _{VSS}							
68	88	W9	$\overline{\text{RES}}$	I(s)	—	—	—	—	—	—
—	89	W8	RTC_X1	I	—	—	—	—	—	—
—	90	Y8	RTC_X2	O	—	—	—	—	—	—

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 5		功能 6		功能 7		简易电路图 图 1.3
			引脚名	I/O	引脚名	I/O	引脚名	I/O	
—	76	W7	—	—	—	—	—	—	(9)
—	77	U7	—	—	—	—	—	—	(9)
58	78	—							
59	79	V8	—	—	—	—	—	—	(3)
60	80	—							
61	81	W6	—	—	—	—	—	—	(1)
62	82	—							
63	83	U6							
64	84	Y5	—	—	—	—	—	—	(10)
65	85	Y6	—	—	—	—	—	—	
66	86	—							
67	87	—							
68	88	W9	—	—	—	—	—	—	(1)
—	89	W8	—	—	—	—	—	—	(11)
—	90	Y8	—	—	—	—	—	—	

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 1		功能 2		功能 3		功能 4	
			引脚名	I/O	引脚名	I/O	引脚名	I/O	引脚名	I/O
69	91	—	USBDPV _{CC}							
70	92	—	USBDPV _{SS}							
71	93	Y9	DM	I/O	—	—	—	—	—	—
72	94	Y10	DP	I/O	—	—	—	—	—	—
73	95	W10	VBUS	I	—	—	—	—	—	—
74	96	—	USBDV _{CC}							
75	97	—	USBDV _{SS}							
76	98	U11	REFRIN	I	—	—	—	—	—	—
77	99	—	USBAPV _{SS}							
78	100	V11	USBAPV _{CC}							
79	101	V12	USBAV _{CC}							
80	102	—	USBAV _{SS}							
81	103	—	USBV _{CC}							
82	104	—	USBV _{SS}							
83	105	Y12	USB_X1	I	—	—	—	—	—	—
84	106	W12	USB_X2	O	—	—	—	—	—	—

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 5		功能 6		功能 7		简易电路图 图 1.3
			引脚名	I/O	引脚名	I/O	引脚名	I/O	
69	91	—							
70	92	—							
71	93	Y9	—	—	—	—	—	—	
72	94	Y10	—	—	—	—	—	—	
73	95	W10	—	—	—	—	—	—	
74	96	—							
75	97	—							
76	98	U11	—	—	—	—	—	—	
77	99	—							
78	100	V11							
79	101	V12							
80	102	—							
81	103	—							
82	104	—							
83	105	Y12	—	—	—	—	—	—	(10)
84	106	W12	—	—	—	—	—	—	

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 1		功能 2		功能 3		功能 4	
			引脚名	I/O	引脚名	I/O	引脚名	I/O	引脚名	I/O
85	107	—	PVcc							
86	108	Y14	VIDEO_X1	I	—	—	—	—	—	—
87	109	W14	VIDEO_X2	O	—	—	—	—	—	—
88	110	—	Vss							
89	111	U13	VDAVcc							
90	112	U15	VDAVss							
91	113	Y15	VIN1	I(a)	—	—	—	—	—	—
92	114	Y16	VIN2	I(a)	—	—	—	—	—	—
93	115	W15	VRT	O	—	—	—	—	—	—
94	116	W13	VRB	O	—	—	—	—	—	—
95	117	W16	BIAS	O	—	—	—	—	—	—
96	118	V14	PH0	I(s)	AN0	I(a)	PINT0	I(s)	—	—
97	119	V15	PH1	I(s)	AN1	I(a)	PINT1	I(s)	—	—
98	120	Y17	PH2	I(s)	AN2	I(a)	PINT2	I(s)	—	—

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 5		功能 6		功能 7		简易电路图 图 1.3
			引脚名	I/O	引脚名	I/O	引脚名	I/O	
85	107	—							
86	108	Y14	—	—	—	—	—	—	(10)
87	109	W14	—	—	—	—	—	—	
88	110	—							
89	111	U13							
90	112	U15							
91	113	Y15	—	—	—	—	—	—	
92	114	Y16	—	—	—	—	—	—	
93	115	W15	—	—	—	—	—	—	
94	116	W13	—	—	—	—	—	—	
95	117	W16	—	—	—	—	—	—	
96	118	V14	—	—	—	—	—	—	(4)
97	119	V15	—	—	—	—	—	—	(4)
98	120	Y17	—	—	—	—	—	—	(4)

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 1		功能 2		功能 3		功能 4	
			引脚名	I/O	引脚名	I/O	引脚名	I/O	引脚名	I/O
99	121	V17	PH3	I(s)	AN3	I(a)	PINT3	I(s)	—	—
100	122	V16	PH4	I(s)	AN4	I(a)	PINT4	I(s)	—	—
101	123	W17	PH5	I(s)	AN5	I(a)	PINT5	I(s)	LCD_EXTCLK	I(s)
102	124	W18	AVSS							
—	125	U14	PH6	I(s)	AN6	I(a)	PINT6	I(s)	—	—
103	126	Y18	AVCC							
—	127	U16	PH7	I(s)	AN7	I(a)	PINT7	I(s)	—	—
104	128	Y19	AVref							
105	129	V20	TRST	I(s)	—	—	—	—	—	—
106	130	U19	ASEBRKAK/ ASEBRK	I(s)/O	—	—	—	—	—	—
107	131	U20	TDO	O	—	—	—	—	—	—
108	132	T18	TDI	I	—	—	—	—	—	—
109	133	R17	TMS	I	—	—	—	—	—	—
110	134	T19	TCK	I	—	—	—	—	—	—
111	135	—	VSS							

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 5		功能 6		功能 7		简易电路图 图 1.3
			引脚名	I/O	引脚名	I/O	引脚名	I/O	
99	121	V17	—	—	—	—	—	—	(4)
100	122	V16	—	—	—	—	—	—	(4)
101	123	W17	—	—	—	—	—	—	(4)
102	124	W18							
—	125	U14	—	—	—	—	—	—	(4)
103	126	Y18							
—	127	U16	—	—	—	—	—	—	(4)
104	128	Y19							
105	129	V20	—	—	—	—	—	—	(3)
106	130	U19	—	—	—	—	—	—	(7)
107	131	U20	—	—	—	—	—	—	(5)
108	132	T18	—	—	—	—	—	—	(2)
109	133	R17	—	—	—	—	—	—	(2)
110	134	T19	—	—	—	—	—	—	(2)
111	135	—							

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 1		功能 2		功能 3		功能 4	
			引脚名	I/O	引脚名	I/O	引脚名	I/O	引脚名	I/O
112	136	R18	PG0	I(s)/O	D16	I/O	LCD_DATA0	O	IRQ0	I(s)
113	137	—	V _{CC}							
114	138	R19	PG1	I(s)/O	D17	I/O	LCD_DATA1	O	IRQ1	I(s)
115	139	—	V _{SS}							
116	140	P18	PG2	I(s)/O	D18	I/O	LCD_DATA2	O	IRQ2	I(s)
117	141	—	PV _{CC}							
118	142	T20	AUDIO_X2	O	—	—	—	—	—	—
119	143	R20	AUDIO_X1	I	—	—	—	—	—	—
120	144	—	V _{SS}							
121	145	P17	PG3	I(s)/O	D19	I/O	LCD_DATA3	O	IRQ3	I(s)
122	146	—	V _{CC}							
123	147	N18	PG4	I(s)/O	D20	I/O	LCD_DATA4	O	IRQ4	I(s)
124	148	P19	PG5	I(s)/O	D21	I/O	LCD_DATA5	O	IRQ5	I(s)
125	149	P20	PG6	I(s)/O	D22	I/O	LCD_DATA6	O	IRQ6	I(s)
126	150	M18	PG7	I(s)/O	D23	I/O	LCD_DATA7	O	IRQ7	I(s)

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 5		功能 6		功能 7		简易电路图 图 1.3
			引脚名	I/O	引脚名	I/O	引脚名	I/O	
112	136	R18	TIOC0A	I(s)/O	—	—	—	—	(8)
113	137	—							
114	138	R19	TIOC0B	I(s)/O	—	—	—	—	(8)
115	139	—							
116	140	P18	TIOC0C	I(s)/O	—	—	—	—	(8)
117	141	—							
118	142	T20	—	—	—	—	—	—	(10)
119	143	R20	—	—	—	—	—	—	
120	144	—							
121	145	P17	TIOC0D	I(s)/O	—	—	—	—	(8)
122	146	—							
123	147	N18	TIOC1A	I(s)/O	—	—	—	—	(8)
124	148	P19	TIOC1B	I(s)/O	—	—	—	—	(8)
125	149	P20	TIOC2A	I(s)/O	—	—	—	—	(8)
126	150	M18	TIOC2B	I(s)/O	—	—	—	—	(8)

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 1		功能 2		功能 3		功能 4	
			引脚名	I/O	引脚名	I/O	引脚名	I/O	引脚名	I/O
—	151	N19	PJ0	I(s)/O	DV_DATA0	I(s)	LCD_DATA0	O	SD_CD_1	I(s)
—	152	—	PV _{CC}							
—	153	N20	PJ1	I(s)/O	DV_DATA1	I(s)	LCD_DATA1	O	SD_WP_1	I(s)
127	154	—	V _{SS}							
128	155	L18	PG8	I(s)/O	D24	I/O	LCD_DATA8	O	PINT0	I(s)
129	156	—	V _{CC}							
—	157	M19	PJ2	I(s)/O	DV_DATA2	I(s)	LCD_DATA2	O	SD_D1_1	I(s)/O
—	158	M20	PJ3	I(s)/O	DV_DATA3	I(s)	LCD_DATA3	O	SD_D0_1	I(s)/O
—	159	L20	PJ4	I(s)/O	DV_DATA4	I(s)	LCD_DATA4	O	SD_CLK_1	O
130	160	L19	PG9	I(s)/O	D25	I/O	LCD_DATA9	O	PINT1	I(s)
131	161	K20	PG10	I(s)/O	D26	I/O	LCD_DATA10	O	PINT2	I(s)
132	162	—	PV _{CC}							
133	163	K18	PG11	I(s)/O	D27	I/O	LCD_DATA11	O	PINT3	I(s)
134	164	—	V _{SS}							
135	165	K19	PG12	I(s)/O	D28	I/O	LCD_DATA12	O	PINT4	I(s)

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 5		功能 6		功能 7		简易电路图 图 1.3
			引脚名	I/O	引脚名	I/O	引脚名	I/O	
—	151	N19	PWM1A	O	—	—	—	—	(7)
—	152	—							
—	153	N20	PWM1B	O	—	—	—	—	(7)
127	154	—							
128	155	L18	TIOC3A	I(s)/O	—	—	—	—	(8)
129	156	—							
—	157	M19	PWM1C	O	—	—	—	—	(7)
—	158	M20	PWM1D	O	—	—	—	—	(7)
—	159	L20	PWM1E	O	—	—	—	—	(7)
130	160	L19	TIOC3B	I(s)/O	—	—	—	—	(8)
131	161	K20	TIOC3C	I(s)/O	—	—	—	—	(8)
132	162	—							
133	163	K18	TIOC3D	I(s)/O	—	—	—	—	(8)
134	164	—							
135	165	K19	—	—	—	—	—	—	(8)

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 1		功能 2		功能 3		功能 4	
			引脚名	I/O	引脚名	I/O	引脚名	I/O	引脚名	I/O
136	166	—	Vcc							
137	167	J20	PG13	I(s)/O	D29	I/O	LCD_DATA13	O	PINT5	I(s)
138	168	J19	PG14	I(s)/O	D30	I/O	LCD_DATA14	O	PINT6	I(s)
139	169	H20	PG15	I(s)/O	D31	I/O	LCD_DATA15	O	PINT7	I(s)
140	170	H19	PG16	I(s)/O	$\overline{\text{WE2/ICIORD/}}/$ DQMUL	O	LCD_DATA16	O	—	—
—	171	G20	PJ5	I(s)/O	DV_DATA5	I(s)	LCD_DATA5	O	SD_CMD_1	I(s)/O
—	172	—	PVcc							
—	173	H18	PJ6	I(s)/O	DV_DATA6	I(s)	LCD_DATA6	O	SD_D3_1	I(s)/O
141	174	—	Vss							
142	175	H17	PG17	I(s)/O	$\overline{\text{WE3/ICIOWR/}}/$ AH/DQMUU	O	LCD_DATA17	O	—	—
143	176	—	Vcc							
—	177	G17	PJ7	I(s)/O	DV_DATA7	I(s)	LCD_DATA7	O	SD_D2_1	I(s)/O
—	178	G19	PJ8	I(s)/O	DV_DATA8	I(s)	LCD_DATA8	O	PINT0	I(s)
—	179	F20	PJ9	I(s)/O	DV_DATA9	I(s)	LCD_DATA9	O	PINT1	I(s)
144	180	G18	PG18	I(s)/O	DV_DATA4	I(s)	LCD_DATA18	O	SPDIF_IN	I(s)

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 5		功能 6		功能 7		简易电路图 图 1.3
			引脚名	I/O	引脚名	I/O	引脚名	I/O	
136	166	—							
137	167	J20	—	—	—	—	—	—	(8)
138	168	J19	—	—	—	—	—	—	(8)
139	169	H20	—	—	—	—	—	—	(8)
140	170	H19	—	—	—	—	AUDATA0	O	(7)
—	171	G20	PWM1F	O	—	—	—	—	(7)
—	172	—							
—	173	H18	PWM1G	O	—	—	—	—	(7)
141	174	—							
142	175	H17	—	—	—	—	AUDATA1	O	(7)
143	176	—							
—	177	G17	PWM1H	O	—	—	—	—	(7)
—	178	G19	PWM2A	O	$\overline{\text{CTS5}}$	I(s)/O	—	—	(7)
—	179	F20	PWM2B	O	$\overline{\text{RTS5}}$	I(s)/O	—	—	(7)
144	180	G18	SCK4	I(s)/O	—	—	—	—	(7)

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 1		功能 2		功能 3		功能 4	
			引脚名	I/O	引脚名	I/O	引脚名	I/O	引脚名	I/O
145	181	F19	PG19	I(s)/O	DV_DATA5	I(s)	LCD_DATA19	O	SPDIF_OUT	O
146	182	—	PV _{CC}							
147	183	E20	PG20	I(s)/O	DV_DATA6	I(s)	LCD_DATA20	O	LCD_TCON3	O
148	184	—	V _{SS}							
149	185	F17	PG21	I(s)/O	DV_DATA7	I(s)	LCD_DATA21	O	LCD_TCON4	O
150	186	—	V _{CC}							
151	187	F18	PG22	I(s)/O	—	—	LCD_DATA22	O	LCD_TCON5	O
152	188	E19	PG23	I(s)/O	—	—	LCD_DATA23	O	LCD_TCON6	O
153	189	D20	PG24	I(s)/O	—	—	LCD_CLK	O	—	—
154	190	E18	PG25	I(s)/O	—	—	LCD_TCON0	O	—	—
155	191	C20	PG26	I(s)/O	—	—	LCD_TCON1	O	—	—
156	192	D19	PG27	I(s)/O	—	—	LCD_TCON2	O	LCD_EXTCLK	I(s)
157	193	A18	PF0	I(s)/O	$\overline{\text{BREQ}}$	I(s)	QSPCLK_1	O	RSPCK1	I(s)/O
158	194	—	PV _{CC}							
159	195	C17	PF1	I(s)/O	$\overline{\text{BACK}}$	O	QSSL_1	O	SSL10	I(s)/O

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 5		功能 6		功能 7		简易电路图 图 1.3
			引脚名	I/O	引脚名	I/O	引脚名	I/O	
145	181	F19	SCK5	I(s)/O	—	—	—	—	(7)
146	182	—							
147	183	E20	RxD4	I(s)	—	—	—	—	(7)
148	184	—							
149	185	F17	TxD4	O	—	—	AUDATA2	O	(7)
150	186	—							
151	187	F18	RxD5	I(s)	—	—	$\overline{\text{AUDSYNC}}$	O	(7)
152	188	E19	TxD5	O	—	—	AUDATA3	O	(7)
153	189	D20	—	—	—	—	—	—	(7)
154	190	E18	—	—	—	—	—	—	(7)
155	191	C20	—	—	—	—	—	—	(7)
156	192	D19	—	—	—	—	—	—	(7)
157	193	A18	TIOC4A	I(s)/O	DREQ0	I(s)	AUDCK	O	(7)
158	194	—							
159	195	C17	TIOC4B	I(s)/O	DACK0	O	—	—	(7)

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 1		功能 2		功能 3		功能 4	
			引脚名	I/O	引脚名	I/O	引脚名	I/O	引脚名	I/O
160	196	—	V _{SS}							
161	197	C15	PF2	I(s)/O	$\overline{\text{WAIT}}$	I(s)	QMO_1/QIO0_1	I(s)/O	MOSI1	I(s)/O
162	198	A17	PF3	I(s)/O	$\overline{\text{CS2}}$	O	QMI_1/QIO1_1	I(s)/O	MISO1	I(s)/O
163	199	B16	PF4	I(s)/O	$\overline{\text{CS5/CE1A}}$	O	SSISCK0	I(s)/O	—	—
164	200	B17	PF5	I(s)/O	—	—	SSIWS0	I(s)/O	—	—
165	201	D14	PF6	I(s)/O	$\overline{\text{CE2A}}$	O	SSITxD0	O	—	—
166	202	A16	PF7	I(s)/O	—	—	SSIRxD0	I(s)	—	—
167	203	C16	PF8	I(s)/O	A23	O	—	—	—	—
168	204	B15	PF9	I(s)/O	$\overline{\text{BS}}$	O	—	—	DV_DATA0	I(s)
169	205	—	PV _{CC}							
170	206	A15	PF10	I(s)/O	$\overline{\text{CS1}}$	O	SSISCK1	I(s)/O	DV_DATA1	I(s)
171	207	—	V _{SS}							
172	208	C14	PF11	I(s)/O	—	—	SSIWS1	I(s)/O	DV_DATA2	I(s)
173	209	D13	PF12	I(s)/O	—	—	SSIDATA1	I(s)/O	DV_DATA3	I(s)
174	210	B14	PF13	I(s)/O	A24	O	SSISCK2	I(s)/O	—	—

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 5		功能 6		功能 7		简易电路图 图 1.3
			引脚名	I/O	引脚名	I/O	引脚名	I/O	
160	196	—							
161	197	C15	TIOC4C	I(s)/O	TEND0	O	SPBMO_1/SPBIO0_1	I(s)/O	(7)
162	198	A17	TIOC4D	I(s)/O	AUDIO_XOUT	O	SPBMI_1/SPBIO1_1	I(s)/O	(7)
163	199	B16	—	—	SGOUT_0	O	—	—	(7)
164	200	B17	—	—	SGOUT_1	O	—	—	(7)
165	201	D14	—	—	SGOUT_2	O	—	—	(7)
166	202	A16	RxD0	I(s)	SGOUT_3	O	$\overline{\text{CTS1}}$	I(s)/O	(7)
167	203	C16	TxD0	O	—	—	—	—	(7)
168	204	B15	SCK0	I(s)/O	MMC_D4	I(s)/O	$\overline{\text{RTS1}}$	I(s)/O	(7)
169	205	—							
170	206	A15	SCK1	I(s)/O	MMC_D5	I(s)/O	—	—	(7)
171	207	—							
172	208	C14	RxD1	I(s)	MMC_D6	I(s)/O	—	—	(7)
173	209	D13	TxD1	O	MMC_D7	I(s)/O	—	—	(7)
174	210	B14	SCK2	I(s)/O	—	—	—	—	(7)

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 1		功能 2		功能 3		功能 4	
			引脚名	I/O	引脚名	I/O	引脚名	I/O	引脚名	I/O
175	211	A14	PF14	I(s)/O	A25	O	SSIWS2	I(s)/O	—	—
176	212	B13	PF15	I(s)/O	A0	O	SSIDATA2	I(s)/O	WDTOVF	O
—	213	—	PVcc							
—	214	C13	PJ10	I(s)/O	DV_DATA10	I(s)	LCD_DATA10	O	PINT2	I(s)
—	215	—	Vss							
177	216	A13	PF16	I(s)/O	SD_CD_0	I(s)	—	—	FCE	O
178	217	A12	PF17	I(s)/O	SD_WP_0	I(s)	—	—	FRB	I(s)
179	218	B12	PF18	I(s)/O	SD_D1_0	I(s)/O	SSISCK3	I(s)/O	—	—
—	219	C11	PJ11	I(s)/O	DV_DATA11	I(s)	LCD_DATA11	O	PINT3	I(s)
—	220	A11	PJ12	I(s)/O	DV_DATA12	I(s)	LCD_DATA12	O	PINT4	I(s)
—	221	B11	PJ13	I(s)/O	DV_DATA13	I(s)	LCD_DATA13	O	PINT5	I(s)
180	222	—	PVcc							
181	223	C12	PF19	I(s)/O	SD_D0_0	I(s)/O	SSIWS3	I(s)/O	—	—
182	224	—	Vss							
183	225	A10	PF20	I(s)/O	SD_CLK_0	O	SSIDATA3	I(s)/O	—	—

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 5		功能 6		功能 7		简易电路图 图 1.3
			引脚名	I/O	引脚名	I/O	引脚名	I/O	
175	211	A14	RxD2	I(s)	—	—	—	—	(7)
176	212	A13	TxD2	O	UBCTRG	O	—	—	(7)
—	213	—							
—	214	C13	PWM2C	O	SCK5	I(s)/O	—	—	(7)
—	215	—							
177	216	A13	IRQ4	I(s)	MMC_CD	I(s)	—	—	(7)
178	217	A12	IRQ5	I(s)	—	—	—	—	(7)
179	218	B12	IRQ6	I(s)	MMC_D1	I(s)/O	—	—	(7)
—	219	C11	PWM2D	O	SCK6	I(s)/O	—	—	(7)
—	220	A11	PWM2E	O	SCK7	I(s)/O	—	—	(7)
—	221	B11	PWM2F	O	TxD5	O	—	—	(7)
180	222	—							
181	223	C12	IRQ7	I(s)	MMC_D0	I(s)/O	—	—	(7)
182	224	—							
183	225	A10	—	—	MMC_CLK	O	—	—	(7)

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 1		功能 2		功能 3		功能 4	
			引脚名	I/O	引脚名	I/O	引脚名	I/O	引脚名	I/O
184	226	—	Vcc							
185	227	B10	PF21	I(s)/O	SD_CMD_0	I(s)/O	—	—	—	—
186	228	A9	PF22	I(s)/O	SD_D3_0	I(s)/O	—	—	—	—
187	229	D9	PF23	I(s)/O	SD_D2_0	I(s)/O	—	—	—	—
188	230	B9	PD0	I/O	D0	I/O	—	—	PWM1A	O
—	231	—	PVcc							
—	232	A8	PJ24	I(s)/O	SGOUT_0	O	SSISCK4	I(s)/O	LCD_TCON3	O
—	233	—	Vss							
189	234	C9	PD1	I/O	D1	I/O	—	—	PWM1B	O
190	235	B8	PD2	I/O	D2	I/O	—	—	PWM1C	O
191	236	A7	PD3	I/O	D3	I/O	—	—	PWM1D	O
—	237	B7	PJ25	I(s)/O	SGOUT_1	O	SSIWS4	I(s)/O	LCD_TCON4	O
—	238	C8	PJ26	I(s)/O	SGOUT_2	O	SSIDATA4	I(s)/O	LCD_TCON5	O
—	239	A6	PJ27	I(s)/O	SGOUT_3	O	—	—	—	—
192	240	—	PVcc							

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 5		功能 6		功能 7		简易电路图 图 1.3
			引脚名	I/O	引脚名	I/O	引脚名	I/O	
184	226	—							
185	227	B10	SCK3	O	MMC_CMD	I(s)/O	—	—	(7)
186	228	A9	RxD3	I(s)	MMC_D3	I(s)/O	—	—	(7)
187	229	D9	TxD3	I(s)/O	MMC_D2	I(s)/O	—	—	(7)
188	230	B9	—	—	—	—	—	—	(6)
—	231	—							
—	232	A8	SPDIF_IN	I(s)	SCK7	I(s)/O	—	—	(7)
—	233	—							
189	234	C9	—	—	—	—	—	—	(6)
190	235	B8	—	—	—	—	—	—	(6)
191	236	A7	—	—	—	—	—	—	(6)
—	237	B7	SPDIF_OUT	O	RxD7	I(s)	—	—	(7)
—	238	C8	—	—	TxD7	O	—	—	(7)
—	239	A6	TIOC1A	I(s)/O	CTS7	I(s)/O	—	—	(7)
192	240	—							

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 1		功能 2		功能 3		功能 4	
			引脚名	I/O	引脚名	I/O	引脚名	I/O	引脚名	I/O
193	241	—	V _{SS}							
194	242	B6	PD4	I/O	D4	I/O	FRE	O	PWM1E	O
195	243	D8	PD5	I/O	D5	I/O	FCLE	O	PWM1F	O
196	244	C7	PD6	I/O	D6	I/O	FALE	O	PWM1G	O
197	245	A5	PD7	I/O	D7	I/O	FWE	O	PWM1H	O
198	246	B5	PD8	I/O	D8	I/O	NAF0	I/O	PWM2A	O
199	247	C5	PD9	I/O	D9	I/O	NAF1	I/O	PWM2B	O
200	248	D7	PD10	I/O	D10	I/O	NAF2	I/O	PWM2C	O
201	249	A4	PD11	I/O	D11	I/O	NAF3	I/O	PWM2D	O
202	250	—	PV _{CC}							
203	251	C6	PD12	I/O	D12	I/O	NAF4	I/O	PWM2E	O
204	252	—	V _{SS}							
205	253	C4	PD13	I/O	D13	I/O	NAF5	I/O	PWM2F	O
206	254	B4	PD14	I/O	D14	I/O	NAF6	I/O	PWM2G	O
207	255	A3	PD15	I/O	D15	I/O	NAF7	I/O	PWM2H	O
208	256	B3	MD_CLK0	I(s)	—	—	—	—	—	—

SH7268 No.	SH7269 No.	SH7269 BGA No.	功能 5		功能 6		功能 7		简易电路图 图 1.3
			引脚名	I/O	引脚名	I/O	引脚名	I/O	
193	241	—							
194	242	B6	—	—	—	—	—	—	(6)
195	243	D8	—	—	—	—	—	—	(6)
196	244	C7	—	—	—	—	—	—	(6)
197	245	A5	—	—	—	—	—	—	(6)
198	246	B5	—	—	—	—	—	—	(6)
199	247	C5	—	—	—	—	—	—	(6)
200	248	D7	—	—	—	—	—	—	(6)
201	249	A4	—	—	—	—	—	—	(6)
202	250	—							
203	251	C6	—	—	—	—	—	—	(6)
204	252	—							
205	253	C4	—	—	—	—	—	—	(6)
206	254	B4	—	—	—	—	—	—	(6)
207	255	A3	—	—	—	—	—	—	(6)
208	256	B3	—	—	—	—	—	—	(1)

【符号说明】

- (s): 施密特
(a): 模拟
(o): 漏极开路

【注】：SH7269 (BGA) 的 PV_{CC}、V_{CC}、V_{SS} 引脚如下：

PV_{CC}: A19、B1、B18、C2、D2、D3、D11、D12、D15、D16、E4、J17、J18、N3、N4、T17、U18、V19、W20、Y11

V_{CC}: A2、B20、C19、D5、D6、D18、E17、H4、J4、M17、N17、T4、U3、U10、V2、V10、W1

V_{SS}: A1、A20、B2、B19、C3、C10、C18、D4、D10、D17、J9、J10、J11、J12、K4、K9、K10、K11、K12、K17、L4、L9、L10、L11、L12、L17、M4、M9、M10、M11、M12、U1、U4、U8、U9、U12、U17、V3、V9、V13、V18、W2、W11、W19、Y1、Y7、Y13、Y20

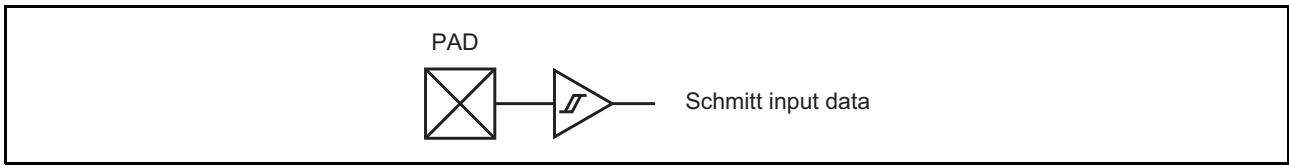


图 1.3 (1) 简易电路图 (施密特输入缓冲器)

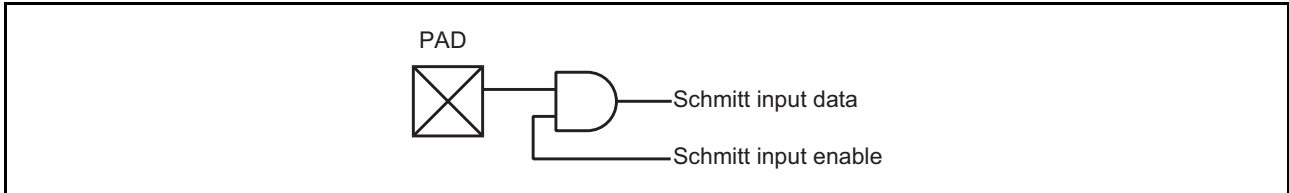


图 1.3 (2) 简易电路图 (TTL AND 输入缓冲器)

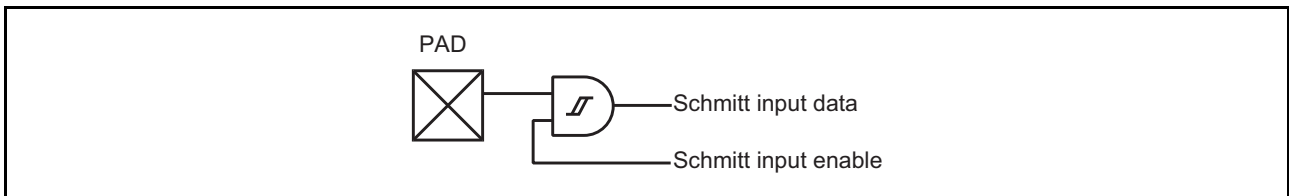


图 1.3 (3) 简易电路图 (施密特 AND 输入缓冲器)

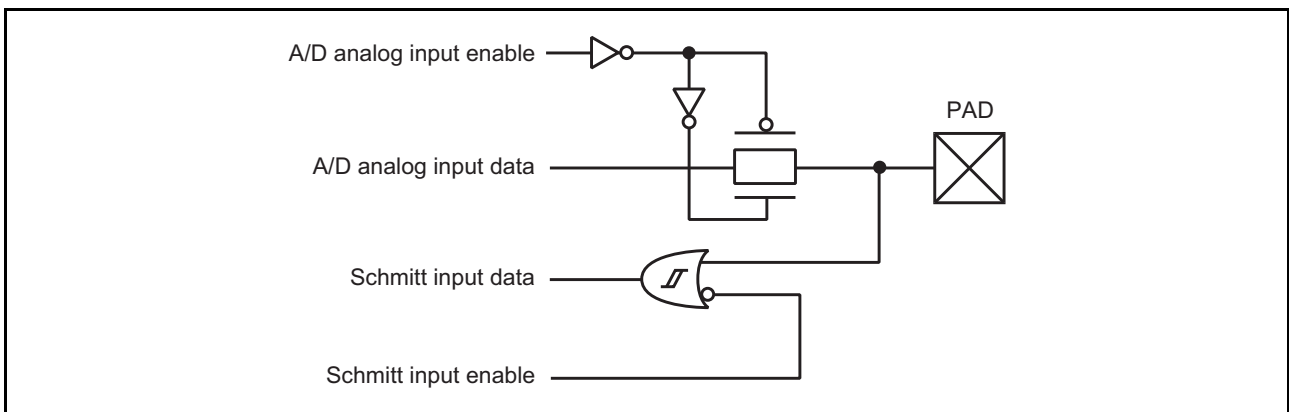


图 1.3 (4) 简易电路图 (施密特 OR 输入 /A/D 输入兼用缓冲器)

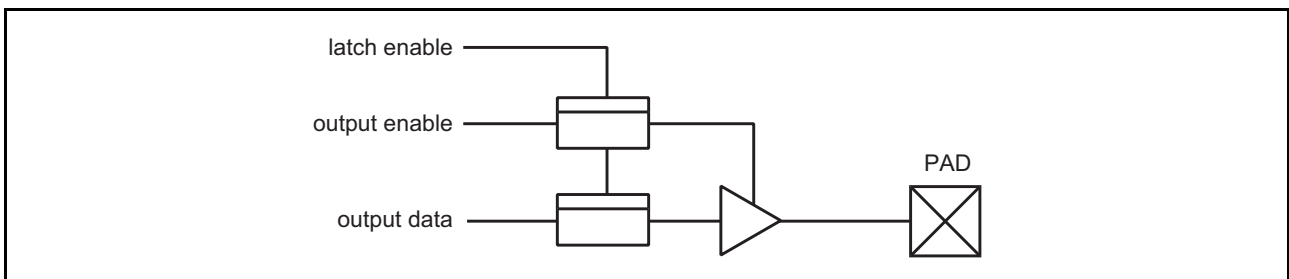


图 1.3 (5) 简易电路图 (带允许的输出缓冲器、带锁存器)

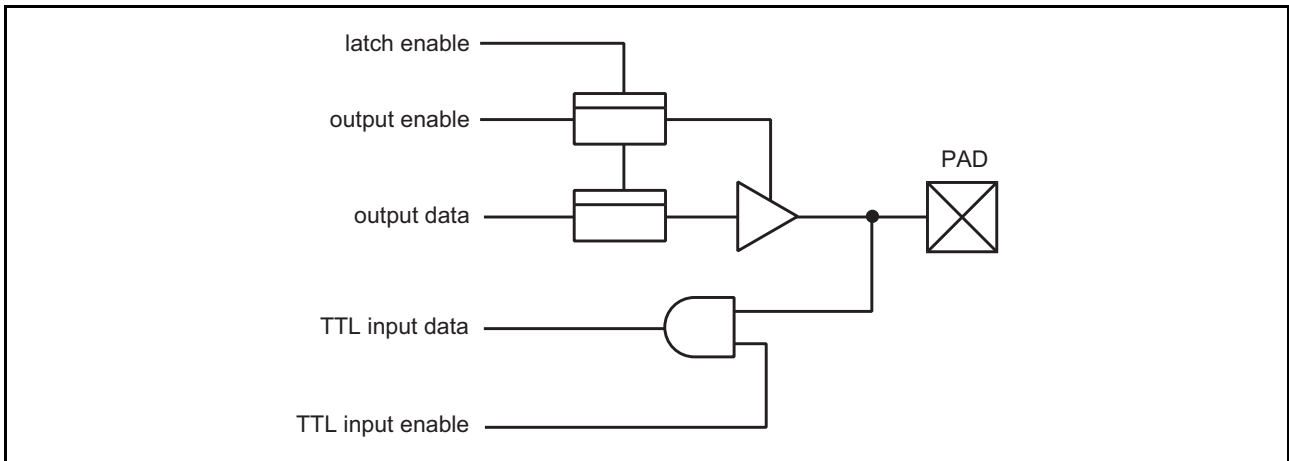


图 1.3 (6) 简易电路图 (双向缓冲器、TTL AND 输入、带锁存器)

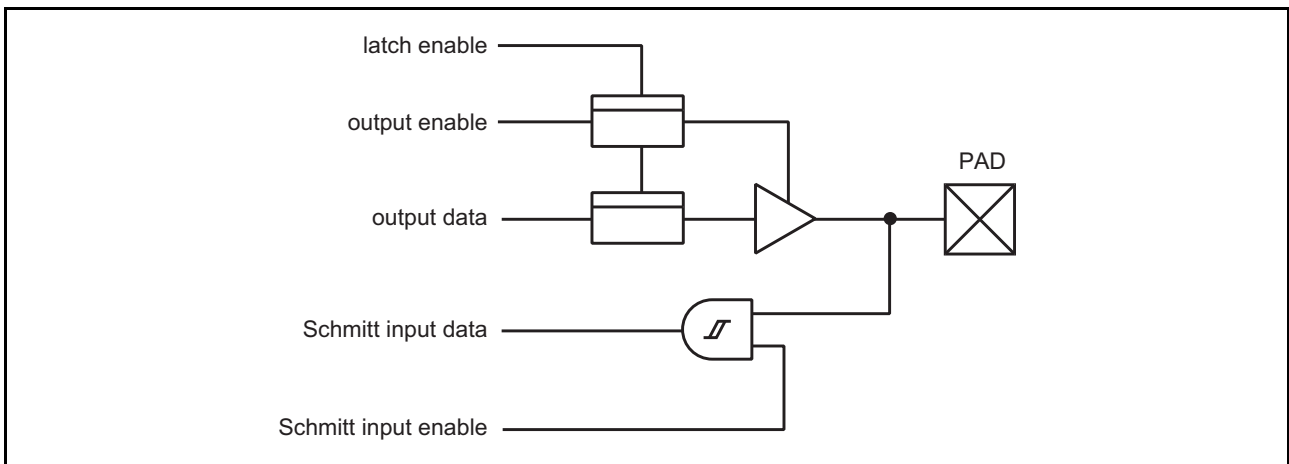


图 1.3 (7) 简易电路图 (双向缓冲器、施密特 AND 输入、带锁存器)

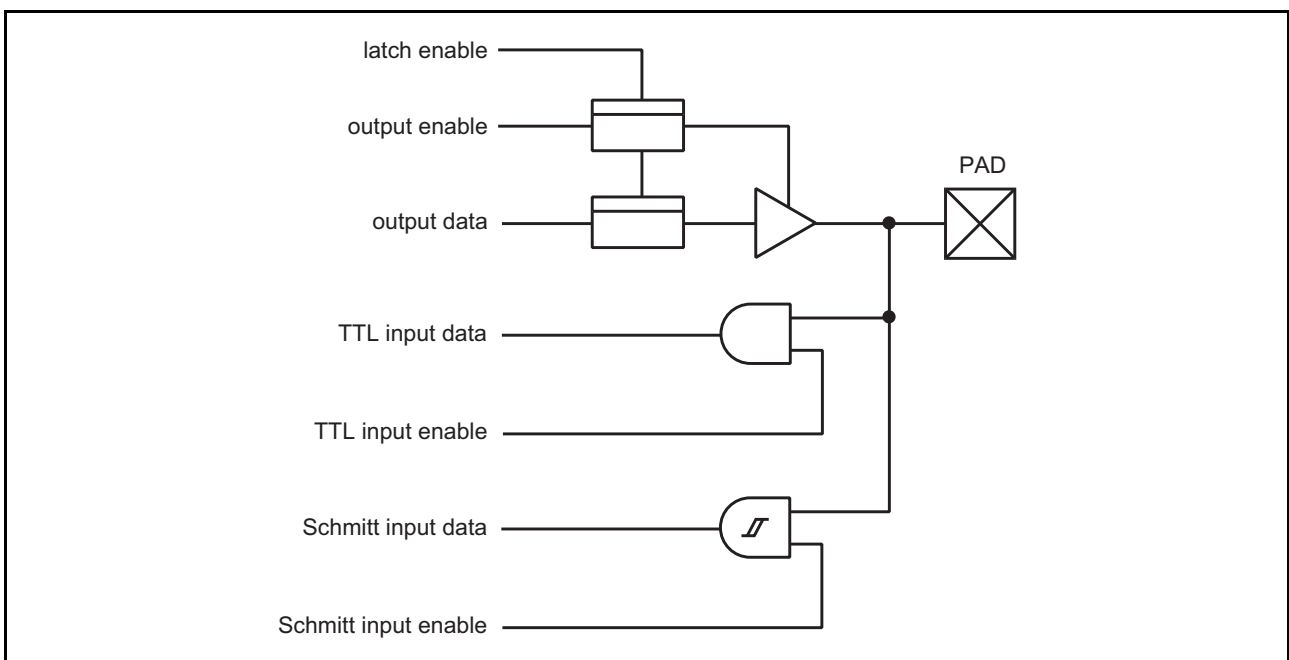


图 1.3 (8) 简易电路图 (双向缓冲器、TTL AND 输入、施密特 AND 输入、带锁存器)

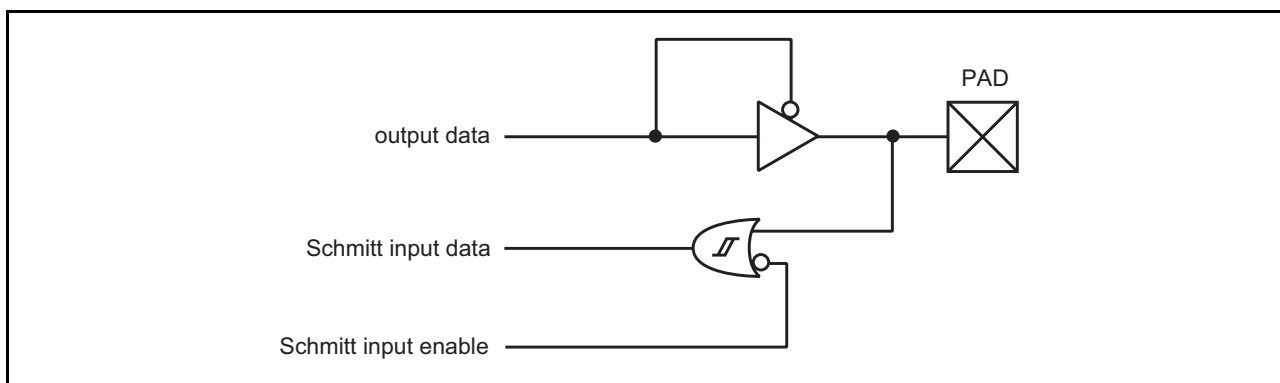


图 1.3 (9) 简易电路图 (漏极开路输出、施密特 OR 输入)

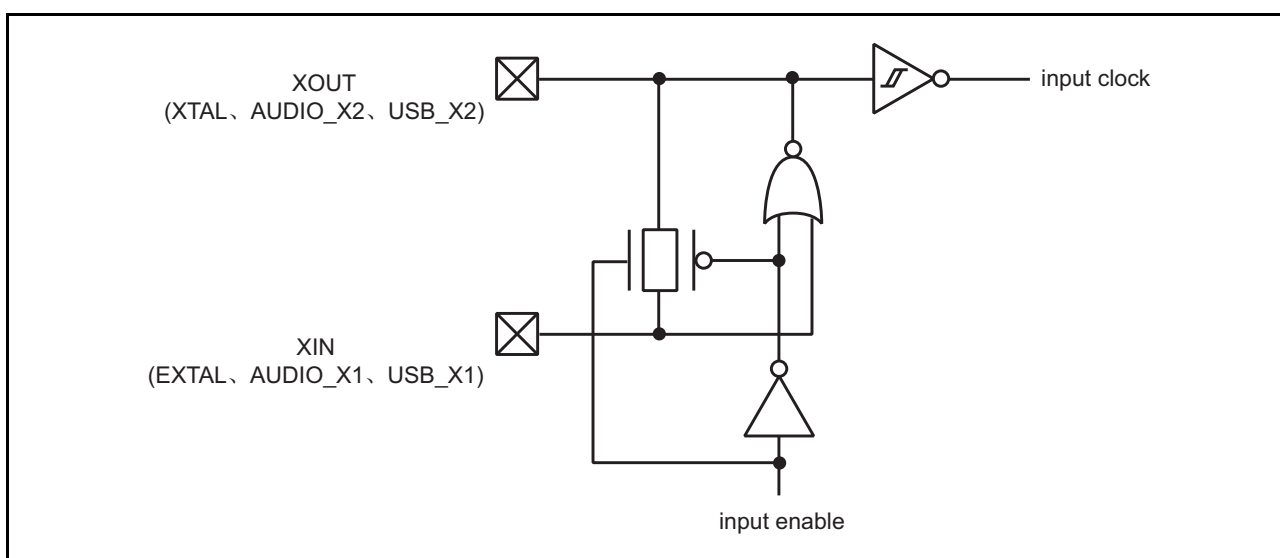


图 1.3 (10) 简易电路图 (振荡缓冲器 1)

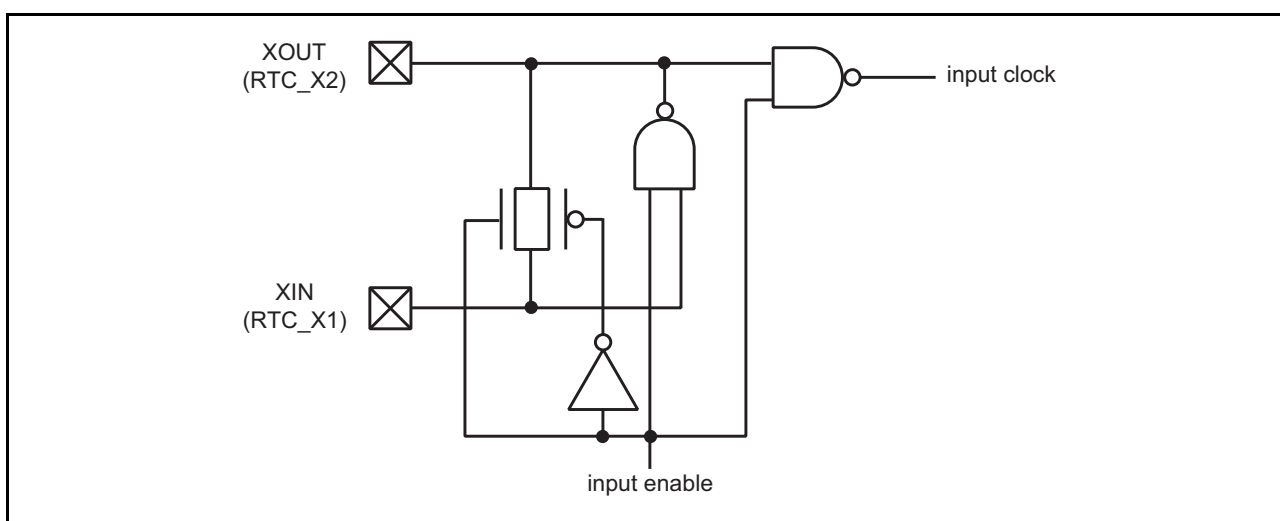


图 1.3 (11) 简易电路图 (振荡缓冲器 2)

2. CPU

2.1 寄存器结构

寄存器有通用寄存器（32 位 × 16 个）、控制寄存器（32 位 × 4 个）和系统寄存器（32 位 × 4 个）共 3 种。

2.1.1 通用寄存器

通用寄存器如图 2.1 所示。通用寄存器为 32 位长，从 R0 到 R15 共有 16 个，用于数据处理和地址计算。R0 也用作变址寄存器，有些指令能使用的寄存器固定为 R0。R15 用作硬件堆栈指针（SP），使用 R15 参照堆栈，保存或者恢复异常处理中的状态寄存器（SR）和程序计数器（PC）。

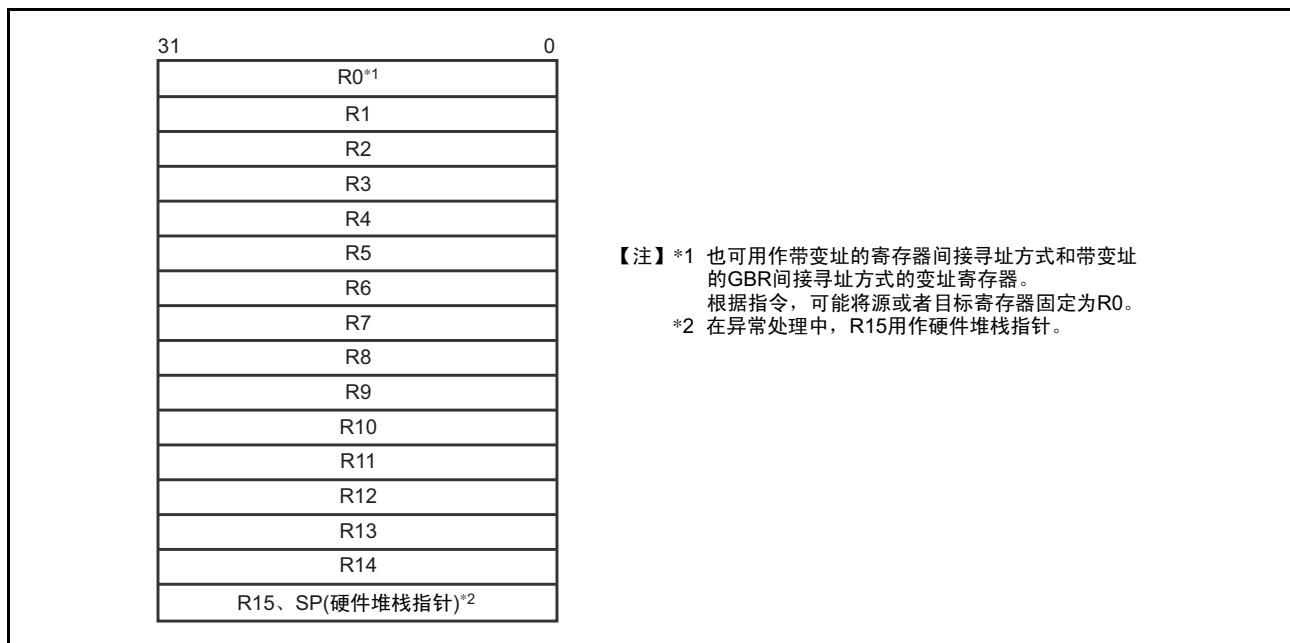


图 2.1 通用寄存器

2.1.2 控制寄存器

控制寄存器为 32 位长，有状态寄存器（SR）、全局基址寄存器（GBR）、向量基址寄存器（VBR）和转移表基址寄存器（TBR）共 4 个。

SR 表示各种指令的处理状态。

GBR 用作 GBR 间接寻址方式的基址，用于内部外围模块寄存器的数据传送等。

VBR 用作包含中断在内的异常处理向量区的基址。

TBR 用作函数表区域的基址。

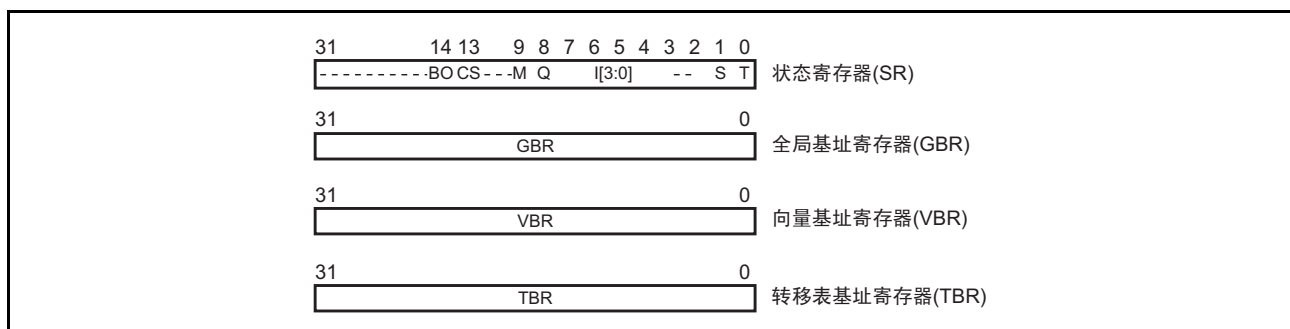


图 2.2 控制寄存器

(1) 状态寄存器 (SR)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	BO	CS	—	—	—	M	Q	I[3:0]			—	—	S	T	
初始值:	0	0	0	0	0	0	—	—	1	1	1	1	0	0	—	—
R/W:	R	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 15	—	全 0	R	保留位 读写值都为“0”。
14	BO	0	R/W	BO 位 表示寄存器组发生上溢。
13	CS	0	R/W	CS 位 表示在执行 CLIP 指令的过程中超过饱和上限值或者低于饱和下限值。
12 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9	M	—	R/W	M 位
8	Q	—	R/W	Q 位 用于 DIV0S、DIV0U、DIV1 指令。
7 ~ 4	I[3:0]	1111	R/W	中断屏蔽级
3、2	—	全 0	R	保留位 读写值都为“0”。
1	S	—	R/W	S 位 指定 MAC 指令的饱和运行。
0	T	—	R/W	T 位 真 / 伪条件或者进位 / 借位

(2) 全局基址寄存器 (GBR)

GBR 作为参照 GBR 的 MOV 指令的基址而被参照。

(3) 向量基址寄存器 (VBR)

在发生异常和中断时，VBR 作为转移目标的基址而被参照。

(4) 转移表基址寄存器 (TBR)

在参照表的子程序调用指令 JSR/N@@(disp8,TBR) 中，转移表基址寄存器作为分配在存储器中的函数表的起始地址而被参照。

2.1.3 系统寄存器

系统寄存器为 32 位长，有乘加寄存器（MACH 和 MACL）、过程寄存器（PR）和程序计数器（PC）共 4 个。MACH 和 MACL 保存乘法或者乘加运算的结果；PR 保存子程序过程的返回地址；PC 表示当前执行指令的 4 字节地址，控制处理的流程。

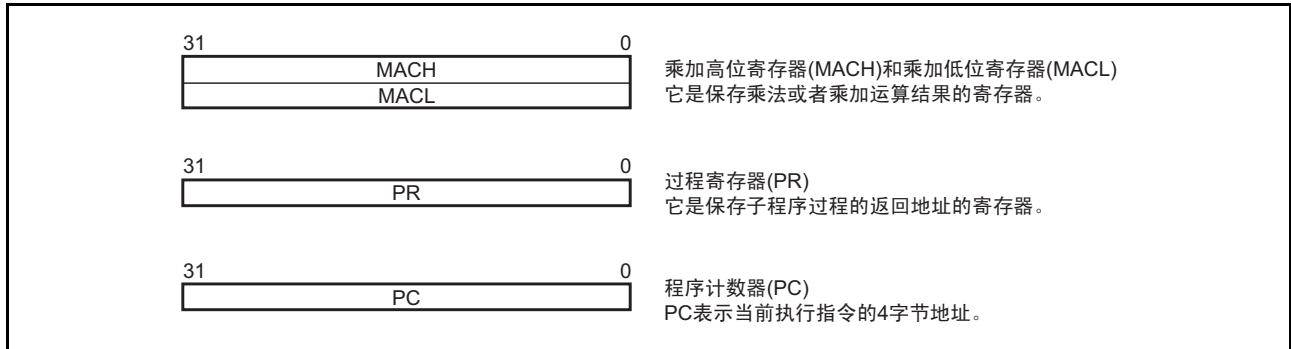


图 2.3 系统寄存器

(1) 乘加高位寄存器（MACH）和乘加低位寄存器（MACL）

MACH 和 MACL 用作 MAC 指令的加法运算值，同时也用于保存 MAC 指令和 MUL 指令的运算结果。

(2) 过程寄存器（PR）

PR 保存使用 BSR、BSRF、JSR 指令的子程序调用的返回地址，并且能通过子程序的返回指令（RTS）进行参照。

(3) 程序计数器（PC）

PC 表示当前执行指令的 4 字节地址。

2.1.4 寄存器组

通用寄存器的 R0 ~ R14、控制寄存器的 GBR 以及系统寄存器的 MACH、MACL 和 PR 共 19 个 32 位寄存器能使用寄存器组进行高速的寄存器保存和寄存器恢复。在 CPU 接受使用寄存器组的中断后，寄存器组的内容自动保存到存储体；通过在中断处理程序中发行 RESBANK 指令，从存储体恢复寄存器组的内容。

本 LSI 有 15 个存储体，详细内容请参照《SH-2A、SH2A-FPU 用户手册 软件篇》和“7.8 寄存器组”。

2.1.5 寄存器的初始值

复位后的寄存器值如表 2.1 所示。

表 2.1 寄存器的初始值

分类	寄存器	初始值
通用寄存器	R0 ~ R14	不定值
	R15 (SP)	向量地址表中的 SP 值
控制寄存器	SR	I[3:0] 为“1111”(HF)，BO 和 CS 为“0”，保留位为“0”，其他位为不定值。
	GBR、TBR	不定值
	VBR	H'00000000
系统寄存器	MACH、MACL、PR	不定值
	PC	向量地址表中的 PC 值

2.2 数据格式

2.2.1 寄存器的数据格式

寄存器操作数的数据长度总是长字（32 位）。在将存储器中的数据装入到寄存器时，如果存储器操作数的数据长度为字节（8 位）或者字（16 位），就将数据扩展（符号扩展）为长字，并且保存到寄存器。



图 2.4 寄存器的数据格式

2.2.2 存储器的数据格式

数据格式有字节、字和长字。存储器能通过 8 位字节、16 位字和 32 位长字中的任意一种格式存取存储器。不满 32 位的存储器操作数在进行符号扩展或者零扩展后被保存到寄存器。

对于字操作数，必须从字边界（2 字节的偶数地址：地址 $2n$ ）进行存取；对于长字操作数，必须从长字边界（4 字节的偶数地址：地址 $4n$ ）进行存取，否则发生地址错误。对于字节操作数，能从任意地址进行存取。

数据格式只能选择大端法的字节序。

存储器的数据格式如图 2.5 所示。

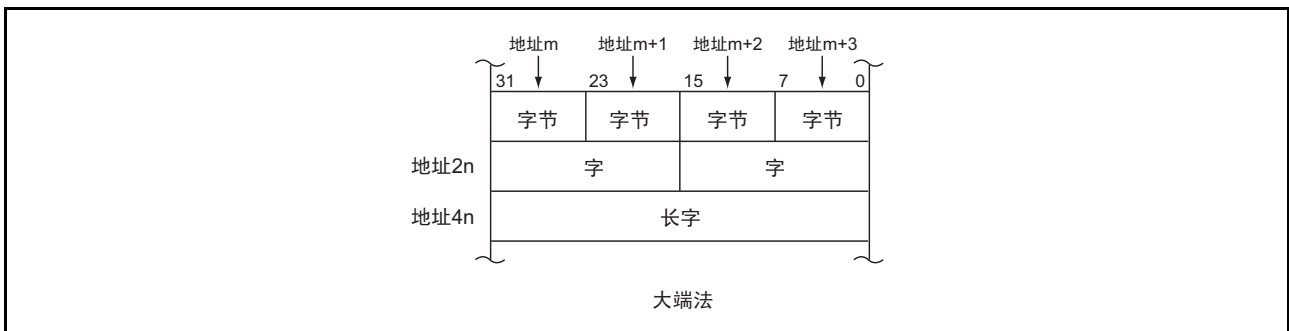


图 2.5 存储器的数据格式

2.2.3 立即数的数据格式

字节（8 位）的立即数分配在指令码中。

MOV、ADD、CMP/EQ 指令在将立即数进行符号扩展后用长字进行运算，TST、AND、OR、XOR 指令在将立即数进行零扩展后用长字进行运算。因此，当 AND 指令使用立即数时，总是清除目标寄存器的高 24 位。

20 位的立即数分配在 32 位长的传送指令 MOVI20 和 MOVI20S 的指令码中。MOVI20 指令在将立即数进行符号扩展后保存到目标寄存器；MOVI20S 指令在将立即数向高位移动 8 位并且进行符号扩展后保存到目标寄存器。

字和长字的立即数不分配在指令码中而分配在存储器的表中。通过带位移量的 PC 相对寻址方式的立即数传送指令（MOV），参照存储器的表。

具体例子请参照“2.3.1(10) 立即数”。

2.3 指令的特点

2.3.1 RISC 方式

指令为 RISC 方式，特点如下。

(1) 16 位固定长指令

基本指令的长度固定为 16 位，因此能提高程序的代码效率。

(2) 32 位固定长指令

在 SH-2A 中追加了 32 位固定长指令，因此提高了性能和可用性。

(3) 1 条指令 / 1 个状态

采用流水线方式，能以 1 个状态执行 1 条基本指令。

(4) 数据长度

运算的基本数据长度是长字，存储器的存取长度能选择字节、字或者长字。对于存储器的字节数据和字数，在进行符号扩展后用长字进行运算；对于立即数，在算术运算时进行符号扩展而在逻辑运算时进行零扩展，然后用长字进行运算。

表 2.2 字数据的符号扩展

SH-2A CPU	说明	其他 CPU 的例子
MOV.W @ (disp,PC),R1 ADD R1,R0DATA.W H'1234	将数据扩展（符号扩展）为 32 位，R1 为“H'00001234”。然后用 ADD 指令进行运算。	ADD.W #H'1234,R0

【注】用 @ (disp,PC) 参照立即数。

(5) 装入 - 存储结构

在寄存器之间进行基本运算。在将数据装入寄存器后，与存储器进行运算（装入 - 存储结构）。但是，AND 等的位操作指令直接对存储器进行运算。

(6) 延迟转移

除了一部分指令以外，无条件转移指令等是延迟转移指令。在使用延迟转移指令时，先执行紧接在延迟转移指令之后的指令，然后进行转移。因此，减少了转移时的流水线混乱。

延迟转移的转移动作发生在执行槽指令之后。但是，指令的执行（更新寄存器等）仍然按延迟转移指令 → 延迟槽指令的顺序进行。例如，即使在延迟槽中更改保存了转移地址的寄存器，但是转移地址仍然是更改前的寄存器内容。

表 2.3 延迟转移指令

SH-2A CPU	说明	其他 CPU 的例子
BRA TRGET ADD R1,R0	在转移到 TRGET 前执行 ADD。	ADD.W R1,R0 BRA TRGET

(7) 无延迟槽的无条件转移指令

SH-2A 中追加了不执行延迟槽指令的无条件转移指令。因此能减少不必要的 NOP 指令，并且能减少代码长度。

(8) 乘法 / 乘加运算

以 1 ~ 2 个状态执行 $16 \times 16 \rightarrow 32$ 的乘法运算，以 2 ~ 3 个状态执行 $16 \times 16 + 64 \rightarrow 64$ 的乘加运算，以 2 ~ 4 个状态执行 $32 \times 32 \rightarrow 64$ 的乘法运算和 $32 \times 32 + 64 \rightarrow 64$ 的乘加运算。

(9) T 位

比较结果反映在状态寄存器 (SR) 的 T 位，根据真伪进行条件转移。只用最小限度的指令改变 T 位，提高了处理速度。

表 2.4 T 位

SH-2A CPU	说明	其他 CPU 的例子
CMP/GE R1,R0	当 $R0 \geq R1$ 时，T 位被置位。	CMP.W R1,R0
BT TRGET0	当 $R0 \geq R1$ 时，转移到 TRGET0。	BGE TRGET0
BF TRGET1	当 $R0 < R1$ 时，转移到 TRGET1。	BLT TRGET1
ADD #-1,R0	ADD 不改变 T 位。	SUB.W #1,R0
CMP/EQ #0,R0	当 $R0=0$ 时，T 位被置位。	BEQ TRGET
BT TRGET	当 $R0=0$ 时，进行转移。	

(10) 立即数

字节的立即数分配在指令码中。字和长字的立即数不分配在指令码中而分配在存储器的表中。通过带位移量的 PC 相对寻址方式的立即数传送指令 (MOV)，参照存储器的表。

在 SH-2A 中，17 ~ 28 位的立即数也能分配在指令码中。但是，对于 21 ~ 28 位的立即数，需要在传送寄存器后执行 OR 指令。

表 2.5 立即数的参照

区分	SH-2A CPU	其他的 CPU 例子
8 位立即数	MOV #H'12,R0	MOV.B #H'12,R0
16 位立即数	MOVI20 #H'1234,R0	MOV.W #H'1234,R0
20 位立即数	MOVI20 #H'12345,R0	MOV.L #H'12345,R0
28 位立即数	MOVI20S #H'12345,R0 OR #H'67,R0	MOV.L #H'1234567,R0
32 位立即数	MOV.L @(disp,PC),R0DATA.L H'12345678	MOV.L #H'12345678,R0

【注】 用 @(disp,PC) 参照立即数。

(11) 绝对地址

当用绝对地址参照数据时，预先将绝对地址的值分配到存储器的表中。在执行指令时，使用立即数的装入方法将该值传送到寄存器，并且通过寄存器间接寻址方式参照数据。

另外，在 SH-2A 中使用不超过 28 位的绝对地址参照数据时，也能将分配在指令码中的立即数转送到寄存器，并且通过寄存器间接寻址方式参照数据。但是，在使用 21 ~ 28 位的绝对地址参照数据时，需要在寄存器传送后使用 OR 指令。

表 2.6 绝对地址的参照

区分	SH-2A CPU	其他 CPU 的例子
不超过 20 位	MOVI20 #H'12345,R1 MOV.B @R1,R0	MOV.B @H'12345,R0
21 ~ 28 位	MOVI20S #H'12345,R1 OR #H'67,R1 MOV.B @R1,R0	MOV.B @H'1234567,R0
不少于 29 位	MOV.L @(disp,PC),R1 MOV.B @R1,R0DATA.L H'12345678	MOV.B @H'12345678,R0

(12) 16 位 /32 位的位移量

当用 16 位或者 32 位的位移量参照数据时，预先将位移量的值分配到存储器的表中。在执行指令时，使用装入立即数的方法将该值传送到寄存器，并且通过带变址的寄存器间接寻址方式参照数据。


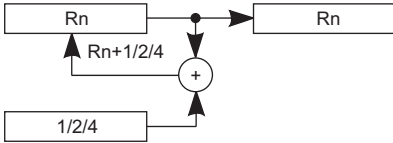
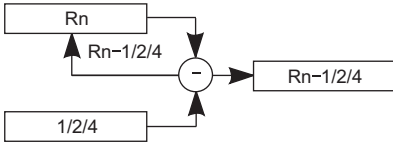
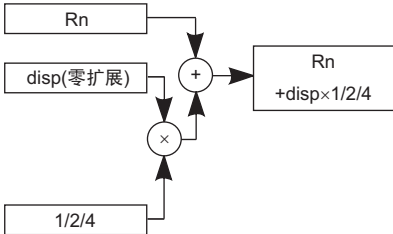
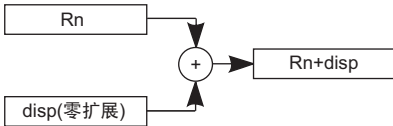
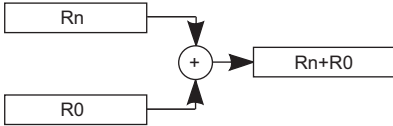
表 2.7 位移量的参照

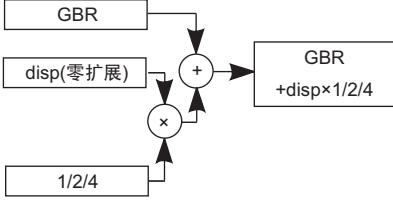
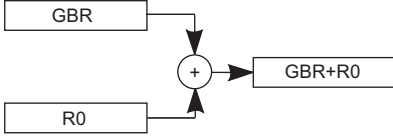
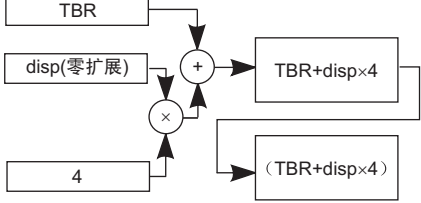
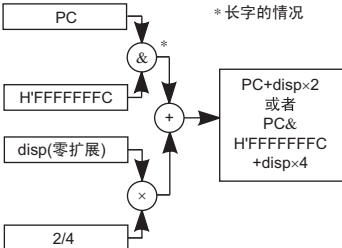
区分	SH-2A CPU	其他 CPU 的例子
16 位的位移量	MOV.W @(disp,PC),R0 MOV.W @(R0,R1),R2DATA.W H'1234	MOV.W @(H'1234,R1),R2

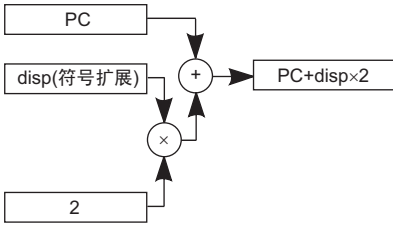
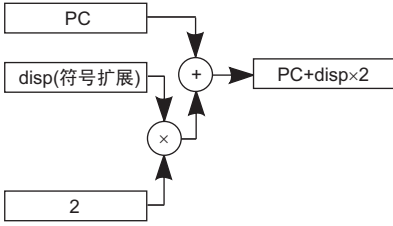
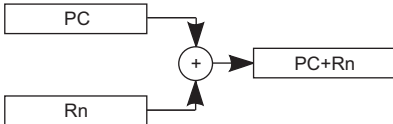
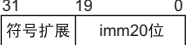
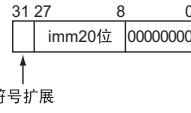
2.3.2 寻址方式

寻址方式和有效地址的计算方法如下所示。

表 2.8 寻址方式和有效地址

寻址方式	指令格式	有效地址的计算方法	计算式
寄存器直接	Rn	有效地址为寄存器 Rn。 (操作数为寄存器 Rn 的内容。)	—
寄存器间接	@Rn	有效地址为寄存器 Rn 的内容。 	Rn
后增寄存器间接	@Rn+	有效地址为寄存器 Rn 的内容。在执行指令后 Rn 加上常数。常数在操作数长度为字节时是“1”，为字时是“2”，为长字时是“4”。 	Rn 指令执行后 字节: Rn+1→Rn 字: Rn+2→Rn 长字: Rn+4→Rn
先减寄存器间接	@-Rn	有效地址为先减去常数后的寄存器 Rn 的内容。常数在操作数长度为字节时是“1”，为字时是“2”，为长字时是“4”。 	字节: Rn-1→Rn 字: Rn-2→Rn 长字: Rn-4→Rn (通过计算后的 Rn 执行指令)
带位移量的寄存器间接	@(disp:4,Rn)	有效地址为寄存器 Rn 加上 4 位位移量 disp 后的内容。disp 在进行零扩展后乘常数，常数在操作数长度为字节时是“1”，为字时是“2”，为长字时是“4”。 	字节: Rn+disp 字: Rn+disp×2 长字: Rn+disp×4
带位移量的寄存器间接	@(disp:12,Rn)	有效地址为寄存器 Rn 加上 12 位位移量 disp 后的内容。disp 进行零扩展。 	字节: Rn+disp 字: Rn+disp 长字: Rn+disp
带变址的寄存器间接	@(R0,Rn)	有效地址为寄存器 Rn 加上 R0 后的内容。 	Rn+R0

寻址方式	指令格式	有效地址的计算方法	计算式
带位移量的 GBR 间接	@(disp:8,GBR)	有效地址为寄存器 GBR 加上 8 位位移量 disp 后的内容。disp 在进行零扩展后乘常数，常数在操作数长度为字节时是“1”，为字时是“2”，为长字时是“4”。 	字节: $GBR + disp$ 字: $GBR + disp \times 2$ 长字: $GBR + disp \times 4$
带变址的 GBR 间接	@(R0,GBR)	有效地址为寄存器 GBR 加上 R0 后的内容。 	$GBR + R0$
带位移量的 TBR 双重间接	@@(disp:8,TBR)	有效地址为寄存器 TBR 加上 8 位位移量 disp 后的内容。disp 在进行零扩展后乘 4。 	($TBR + disp \times 4$) 地址的内容
带位移量的 PC 相对	@(disp:8,PC)	有效地址为寄存器 PC 加上 8 位位移量 disp 后的内容。disp 在进行零扩展后乘常数，常数在操作数长度为字时是“2”，为长字时是“4”。另外，在操作数长度为长字时，屏蔽 PC 的低 2 位。 	字: $PC + disp \times 2$ 长字: $PC \& H'FFFFFFFC + disp \times 4$

寻址方式	指令格式	有效地址的计算方法	计算式
PC 相对	disp:8	有效地址为寄存器 PC 加上 8 位位移量 disp (符号扩展后的 disp) 乘 2 后的内容。 	$PC+disp \times 2$
	disp:12	有效地址为寄存器 PC 加上 12 位位移量 disp (符号扩展后的 disp) 乘 2 后的内容。 	$PC+disp \times 2$
	Rn	有效地址为寄存器 PC 加上 Rn 后的内容。 	$PC+Rn$
立即数	#imm:20	将 MOVI20 指令的 20 位立即数 imm 进行符号扩展。 	—
		将 MOVI20S 指令的 20 位立即数 imm 向左移 8 位, 高位进行符号扩展, 低位填零。 	—
	#imm:8	将 TST、AND、OR、XOR 指令的 8 位立即数 imm 进行零扩展。	—
	#imm:8	将 MOV、ADD、CMP/EQ 指令的 8 位立即数 imm 进行符号扩展。	—
	#imm:8	将 TRAPA 指令的 8 位立即数 imm 进行零扩展后乘“4”。	—
	#imm:3	BAND、BOR、BXOR、BST、BLD、BSET、BCLR 指令的 3 位立即数 imm 表示位的位置。	—

2.3.3 指令格式

本节说明指令格式、源操作数和目标操作数。操作数的含义因指令码而不同。符号如下：

xxxx：指令码

mmmm：源寄存器

nnnn：目标寄存器

iiii：立即数

dddd：位移量

表 2.9 指令格式

指令格式	源操作数	目标操作数	指令例子
0 格式 	—	—	NOP
n 格式 	—	nnnn: 寄存器直接	MOV T Rn
	控制寄存器 或者系统寄存器	nnnn: 寄存器直接	STS MACH,Rn
	R0 (寄存器直接)	nnnn: 寄存器直接	DIVU R0,Rn
	控制寄存器 或者系统寄存器	nnnn: 先减寄存器间接	STC.L SR,@-Rn
	mmmm: 寄存器直接	R15 (先减寄存器间接)	MOVMU.L Rm,@-R15
	R15 (后增寄存器间接)	nnnn: 寄存器直接	MOVMU.L @R15+,Rn
	R0 (寄存器直接)	nnnn: 后增寄存器间接	MOV.L R0,@Rn+
m 格式 	mmmm: 寄存器直接	控制寄存器 或者系统寄存器	LDC Rm,SR
	mmmm: 后增寄存器间接	控制寄存器 或者系统寄存器	LDC.L @Rm+,SR
	mmmm: 寄存器间接	—	JMP @Rm
	mmmm: 先减寄存器间接	R0 (寄存器直接)	MOV.L @-Rm,R0
	mmmm: 使用 Rm 的 PC 相对	—	BRAF Rm
nm 格式 	mmmm: 寄存器直接	nnnn: 寄存器直接	ADD Rm,Rn
	mmmm: 寄存器直接	nnnn: 寄存器间接	MOV.L Rm,@Rn
	mmmm: 后增寄存器间接 (乘加运算) nnnn: * 后增寄存器间接 (乘加运算)	MACH、MACL	MAC.W @Rm+,@Rn+
	mmmm: 后增寄存器间接	nnnn: 寄存器直接	MOV.L @Rm+,Rn
	mmmm: 寄存器直接	nnnn: 先减寄存器间接	MOV.L Rm,@-Rn
	mmmm: 寄存器直接	nnnn: 带变址的寄存器间接	MOV.L Rm,@(R0,Rn)

指令格式		源操作数	目标操作数	指令例子
md 格式		mmmmmddd: 带位移量的寄存器间接	R0 (寄存器直接)	MOV.B @ (disp,Rm),R0
nd4 格式		R0 (寄存器直接)	nnnnddd: 带位移量的寄存器间接	MOV.B R0,@ (disp,Rn)
nmd 格式		mmmm: 寄存器直接	nnnnddd: 带位移量的寄存器间接	MOV.L Rm,@ (disp,Rn)
		mmmmddd: 带位移量的寄存器间接	nnnn: 寄存器直接	MOV.L @ (disp,Rm),Rn
nmd12 格式		mmmm: 寄存器直接	nnnnddd: 带位移量的寄存器间接	MOV.L Rm,@ (disp12,Rn)
		mmmmddd: 带位移量的寄存器间接	nnnn: 寄存器直接	MOV.L @ (disp12,Rm),Rn
d 格式		ddddddd: 带位移量的 GBR 间接	R0 (寄存器直接)	MOV.L @ (disp,GBR),R0
		R0 (寄存器直接)	ddddddd: 带位移量的 GBR 间接	MOV.L R0,@ (disp,GBR)
		ddddddd: 带位移量的 PC 相对	R0 (寄存器直接)	MOVA @ (disp,PC),R0
		ddddddd: 带位移量的 TBR 双重间接	—	JSR/N @@ (disp8,TBR)
		ddddddd: PC 相对	—	BF label
d12 格式		ddddddddddd: PC 相对	—	BRA label (label=disp+PC)
nd8 格式		ddddddd: 带位移量的 PC 相对	nnnn: 寄存器直接	MOV.L @ (disp,PC),Rn
i 格式		iiiiiii: 立即数	带变址的 GBR 间接	AND.B #imm,@ (R0,GBR)
		iiiiiii: 立即数	R0 (寄存器直接)	AND #imm,R0
		iiiiiii: 立即数	—	TRAPA #imm
ni 格式		iiiiiii: 立即数	nnnn: 寄存器直接	ADD #imm,Rn
ni3 格式		nnnn: 寄存器直接 iii: 立即数	—	BLD #imm3,Rn
		—	nnnn: 寄存器直接 iii: 立即数	BST #imm3,Rn
ni20 格式		iiiiiiiiiiiiiiiiiiii: 立即数	nnnn: 寄存器直接	MOVI20 #imm20,Rn
nid 格式		nnnnddddddddddd: 带位移量的寄存器间接 iii: 立即数	—	BLD.B #imm3,@ (disp12,Rn)
		—	nnnnddddddddddd: 带位移量的寄存器间接 iii: 立即数	BST.B #imm3,@ (disp12,Rn)

【注】 * 在乘加指令中，nnnn 为源寄存器。

2.4 指令系统

2.4.1 指令系统的分类

指令的分类如表 2.10 所示。

表 2.10 指令的分类

分类	指令的种类	操作码	功能	指令数
数据传送指令	13	MOV	数据传送 立即数的传送 外围模块数据的传送 结构体数据的传送 反向堆栈的传送	62
		MOVA	执行地址的传送	
		MOVI20	20 位立即数的传送	
		MOVI20S	20 位立即数的传送 左移 8 位	
		MOVML	R0 ~ Rn 寄存器的保存和恢复	
		MOV MU	Rn ~ R14、PR 寄存器的保存和恢复	
		MOV RT	将 T 位取反并且传送到 Rn。	
		MOV T	T 位的传送	
		MOV U	无符号数据的传送	
		NOT T	T 位的取反	
		PREF	将操作数预取到高速缓存。	
		SWAP	交换高位和低位。	
		XTRCT	抽出连接寄存器的中间部分。	
算术运算指令	26	ADD	2 进制加法	40
		ADDC	带进位的 2 进制加法	
		ADDV	带上溢的 2 进制加法	
		CMP/cond	比较	
		CLIPS	带符号的饱和值比较	
		CLIPU	无符号的饱和值比较	
		DIVS	带符号除法 (32÷32)	
		DIVU	无符号除法 (32÷32)	
		DIV1	单步除法	
		DIV0S	带符号的单步除法的初始化	

分类	指令的种类	操作码	功能	指令数
算术运算指令	26	DIV0U	无符号的单步除法的初始化	40
		DMULS	带符号的双精度乘法	
		DMULU	无符号的双精度乘法	
		DT	递减和测试	
		EXTS	符号扩展	
		EXTU	零扩展	
		MAC	乘加运算、双精度乘加运算	
		MUL	双精度乘法	
		MULR	结果保存到 Rn 的带符号乘法	
		MULS	带符号乘法	
		MULU	无符号乘法	
		NEG	符号取反	
		NEGC	带借位的符号取反	
		SUB	2 进制减法	
		SUBC	带借位的 2 进制减法	
SUBV	带下溢的 2 进制减法			
逻辑运算指令	6	AND	逻辑与运算	14
		NOT	位的取反	
		OR	逻辑或运算	
		TAS	存储器测试和置位	
		TST	逻辑与运算的 T 位置位	
		XOR	逻辑异或运算	
移位指令	12	ROTL	左循环 1 位	16
		ROTR	右循环 1 位	
		ROTCL	带 T 位的左循环 1 位	
		ROTCR	带 T 位的右循环 1 位	
		SHAD	动态算术移位	
		SHAL	算术左移 1 位	
		SHAR	算术右移 1 位	
		SHLD	动态逻辑移位	
		SHLL	逻辑左移 1 位	
		SHLLn	逻辑左移 n 位	
		SHLR	逻辑右移 1 位	
SHLRn	逻辑右移 n 位			

分类	指令的种类	操作码	功能	指令数
转移指令	10	BF	条件转移、带延迟的条件转移 (当 T=0 时进行转移)	15
		BT	条件转移、带延迟的条件转移 (当 T=1 时进行转移)	
		BRA	带延迟的无条件转移	
		BRAF	带延迟的无条件转移	
		BSR	转移到子程序过程 (带延迟)。	
		BSRF	转移到子程序过程 (带延迟)。	
		JMP	带延迟的无条件转移	
		JSR	转移到子程序过程。 转移到子程序过程 (带延迟)。	
		RTS	从子程序过程返回。 从子程序过程返回 (带延迟)。	
	RTV/N	从子程序过程返回 (带 Rm→R0 传送)。		
系统控制指令	14	CLRT	T 位的清除	36
		CLRMAC	MAC 寄存器的清除	
		LDBANK	指定寄存器组入口的寄存器恢复	
		LDC	装入到控制寄存器。	
		LDS	装入到系统寄存器。	
		NOP	无操作	
		RESBANK	寄存器组的寄存器恢复	
		RTE	从异常处理返回。	
		SETT	T 位的置位	
		SLEEP	转移到低功耗状态。	
		STBANK	将寄存器保存到指定的寄存器组入口。	
		STC	控制寄存器的保存	
		STS	系统寄存器的保存	
	TRAPA	陷阱异常处理		
浮点运算指令	19	FABS	浮点数绝对值	48
		FADD	浮点数加法	
		FCMP	浮点数比较	
		FCNVDS	将双精度转换为单精度。	
		FCNVSD	将单精度转换为双精度。	
		FDIV	浮点数除法	
		FLDI0	浮点数装入立即数 0	
		FLDI1	浮点数装入立即数 1	
		FLDS	将浮点数装入到系统寄存器 FPUL。	
		FLOAT	将整数转换为浮点数。	
		FMAC	浮点数乘加运算	

分类	指令的种类	操作码	功能	指令数
浮点运算指令	19	FMOV	浮点数传送	48
		FMUL	浮点数乘法	
		FNEG	浮点数符号取反	
		FSCHG	SZ 位的取反	
		FSQRT	浮点平方根	
		FSTS	系统寄存器 FPUL 的浮点数保存	
		FSUB	浮点数减法	
		FTRC	转换（舍去）为浮点数的整数。	
FPU 相关的 CPU 指令	2	LDS	装入到浮点系统寄存器。	8
		STS	浮点系统寄存器的保存	
位操作指令	10	BAND	位逻辑与	14
		BCLR	位清除	
		BLD	位装入	
		BOR	位逻辑或	
		BSET	置位	
		BST	位保存	
		BXOR	位逻辑异或	
		BANDNOT	位逻辑与非	
		BORNOT	位逻辑或非	
		BLDNOT	位逻辑非装入	
	计 112			253

通过以下格式按照分类顺序说明指令的指令码、操作和执行状态。

指令	指令码	操作概要	执行状态	T 位
用助记符表示。 【符号说明】 Rm : 源寄存器 Rn : 目标寄存器 imm : 立即数 disp : 位移量 *2	按照 MSB←→LSB 的顺序表示。 【符号说明】 mmmm: 源寄存器 nnnn: 目标寄存器 0000: R0 0001: R1 …… 1111: R15 iiii: 立即数 dddd: 位移量	表示操作的概要。 【符号说明】 →、←: 传送方向 (xx) : 存储器操作数 M/Q/T : SR 内的标志位 & : 每位的逻辑与 : 每位的逻辑或 ^ : 每位的逻辑异或 ~ : 每位的逻辑非 <<n : 左移 n 位 >>n : 右移 n 位	无等待时的值 *1	表示执行指令后的 T 位的值。 【符号说明】 —: 不变

【注】 *1 有关指令的执行状态

表中所示的执行状态为最小值。实际上根据以下条件，指令执行的状态数将会增加。

(1) 当取指令和数据存取发生竞争时

(2) 当装入指令（存储器→寄存器）的目标寄存器和紧接着的指令所使用的寄存器相同时

*2 根据指令的操作数长度等决定倍数（×1、×2、×4）。

详细内容请参照《SH-2A、SH2A-FPU 用户手册 软件篇》。

2.4.2 数据传送指令

表 2.11 数据传送指令

指令	指令码	操作	执行状态	T 位	适用指令		
					SH2、SH2E	SH4	SH-2A
MOV #imm,Rn	1110nnnniiiiiiii	imm→符号扩展→Rn	1	—	○	○	○
MOV.W @(disp,PC),Rn	1001nnnnddddddd	(disp×2+PC)→符号扩展→Rn	1	—	○	○	○
MOV.L @(disp,PC),Rn	1101nnnnddddddd	(disp×4+PC)→Rn	1	—	○	○	○
MOV Rm,Rn	0110nnnnmmmm0011	Rm→Rn	1	—	○	○	○
MOV.B Rm,@Rn	0010nnnnmmmm0000	Rm→(Rn)	1	—	○	○	○
MOV.W Rm,@Rn	0010nnnnmmmm0001	Rm→(Rn)	1	—	○	○	○
MOV.L Rm,@Rn	0010nnnnmmmm0010	Rm→(Rn)	1	—	○	○	○
MOV.B @Rm,Rn	0110nnnnmmmm0000	(Rm)→符号扩展→Rn	1	—	○	○	○
MOV.W @Rm,Rn	0110nnnnmmmm0001	(Rm)→符号扩展→Rn	1	—	○	○	○
MOV.L @Rm,Rn	0110nnnnmmmm0010	(Rm)→Rn	1	—	○	○	○
MOV.B Rm,@-Rn	0010nnnnmmmm0100	Rn-1→Rn, Rm→(Rn)	1	—	○	○	○
MOV.W Rm,@-Rn	0010nnnnmmmm0101	Rn-2→Rn, Rm→(Rn)	1	—	○	○	○
MOV.L Rm,@-Rn	0010nnnnmmmm0110	Rn-4→Rn, Rm→(Rn)	1	—	○	○	○
MOV.B @Rm+,Rn	0110nnnnmmmm0100	(Rm)→符号扩展→Rn, Rm+1→Rm	1	—	○	○	○
MOV.W @Rm+,Rn	0110nnnnmmmm0101	(Rm)→符号扩展→Rn, Rm+2→Rm	1	—	○	○	○
MOV.L @Rm+,Rn	0110nnnnmmmm0110	(Rm)→Rn, Rm+4→Rm	1	—	○	○	○
MOV.B R0,@(disp,Rn)	10000000nnnnddd	R0→(disp+Rn)	1	—	○	○	○
MOV.W R0,@(disp,Rn)	10000001nnnnddd	R0→(disp×2+Rn)	1	—	○	○	○
MOV.L Rm,@(disp,Rn)	0001nnnnmmmmddd	Rm→(disp×4+Rn)	1	—	○	○	○
MOV.B @(disp,Rm),R0	10000100mmmmddd	(disp+Rm)→符号扩展→R0	1	—	○	○	○
MOV.W @(disp,Rm),R0	10000101mmmmddd	(disp×2+Rm)→符号扩展→R0	1	—	○	○	○
MOV.L @(disp,Rm),Rn	0101nnnnmmmmddd	(disp×4+Rm)→Rn	1	—	○	○	○
MOV.B Rm,@(R0,Rn)	0000nnnnmmmm0100	Rm→(R0+Rn)	1	—	○	○	○
MOV.W Rm,@(R0,Rn)	0000nnnnmmmm0101	Rm→(R0+Rn)	1	—	○	○	○
MOV.L Rm,@(R0,Rn)	0000nnnnmmmm0110	Rm→(R0+Rn)	1	—	○	○	○
MOV.B @(R0,Rm),Rn	0000nnnnmmmm1100	(R0+Rm)→符号扩展→Rn	1	—	○	○	○
MOV.W @(R0,Rm),Rn	0000nnnnmmmm1101	(R0+Rm)→符号扩展→Rn	1	—	○	○	○
MOV.L @(R0,Rm),Rn	0000nnnnmmmm1110	(R0+Rm)→Rn	1	—	○	○	○
MOV.B R0,@(disp,GBR)	11000000ddddddd	R0→(disp+GBR)	1	—	○	○	○
MOV.W R0,@(disp,GBR)	11000001ddddddd	R0→(disp×2+GBR)	1	—	○	○	○
MOV.L R0,@(disp,GBR)	11000010ddddddd	R0→(disp×4+GBR)	1	—	○	○	○
MOV.B @(disp,GBR),R0	11000100ddddddd	(disp+GBR)→符号扩展→R0	1	—	○	○	○
MOV.W @(disp,GBR),R0	11000101ddddddd	(disp×2+GBR)→符号扩展→R0	1	—	○	○	○
MOV.L @(disp,GBR),R0	11000110ddddddd	(disp×4+GBR)→R0	1	—	○	○	○
MOV.B R0,@Rn+	0100nnnn10001011	R0→(Rn), Rn+1→Rn	1	—			○

指令	指令码	操作	执行状态	T 位	适用指令		
					SH2、SH2E	SH4	SH-2A
MOV.W R0,@Rn+	0100nnnn10011011	R0→(Rn), Rn+2→Rn	1	—			○
MOV.L R0,@Rn+	0100nnnn10101011	R0→(Rn), Rn+4→Rn	1	—			○
MOV.B @-Rm,R0	0100mmmm11001011	Rm-1→Rm, (Rm)→符号扩展→R0	1	—			○
MOV.W @-Rm,R0	0100mmmm11011011	Rm-2→Rm, (Rm)→符号扩展→R0	1	—			○
MOV.L @-Rm,R0	0100mmmm11101011	Rm-4→Rm, (Rm)→R0	1	—			○
MOV.B Rm,@(disp12,Rn)	0011nnnnmmmm0001 0000ddddddddddd	Rm→(disp+Rn)	1	—			○
MOV.W Rm,@(disp12,Rn)	0011nnnnmmmm0001 0001ddddddddddd	Rm→(disp×2+Rn)	1	—			○
MOV.L Rm,@(disp12,Rn)	0011nnnnmmmm0001 0010ddddddddddd	Rm→(disp×4+Rn)	1	—			○
MOV.B @(disp12,Rm),Rn	0011nnnnmmmm0001 0100ddddddddddd	(disp+Rm)→符号扩展→Rn	1	—			○
MOV.W @(disp12,Rm),Rn	0011nnnnmmmm0001 0101ddddddddddd	(disp×2+Rm)→符号扩展→Rn	1	—			○
MOV.L @(disp12,Rm),Rn	0011nnnnmmmm0001 0110ddddddddddd	(disp×4+Rm)→Rn	1	—			○
MOVA @(disp,PC),R0	11000111ddddddd	disp×4+PC→R0	1	—	○	○	○
MOVI20 #imm20,Rn	0000nnnniiii0000 iiiiiiiiiiiiiiii	imm→符号扩展→Rn	1	—			○
MOVI20S #imm20,Rn	0000nnnniiii0001 iiiiiiiiiiiiiiii	imm<<8→符号扩展→Rn	1	—			○
MOVML.L Rm,@-R15	0100mmmm11110001	R15-4→R15, Rm→(R15) R15-4→R15, Rm-1→(R15) : R15-4→R15, R0→(R15) ※当 Rm=R15 时, 将 Rm 读作 PR。	1~ 16	—			○
MOVML.L @R15+,Rn	0100nnnn11110101	(R15)→R0, R15+4→R15 (R15)→R1, R15+4→R15 : (R15)→Rn ※当 Rn=R15 时, 将 Rn 读作 PR。	1~ 16	—			○
MOVML.L Rm,@-R15	0100mmmm11110000	R15-4→R15, PR→(R15) R15-4→R15, R14→(R15) : R15-4→R15, Rm→(R15) ※当 Rm=R15 时, 将 Rm 读作 PR。	1~ 16	—			○

指令	指令码	操作	执行状态	T 位	适用指令		
					SH2、SH2E	SH4	SH-2A
MOVML @R15+,Rn	0100nnnn11110100	(R15)→Rn, R15+4→R15 (R15)→Rn+1, R15+4→R15 : (R15)→R14, R15+4→R15 (R15)→PR ※当 Rn=R15 时, 将 Rn 读作 PR。	1 ~ 16	—			○
MOVRT Rn	0000nnnn00111001	~T→Rn	1	—			○
MOVT Rn	0000nnnn00101001	T→Rn	1	—	○	○	○
MOVU.B @(disp12,Rm),Rn	0011nnnnmmmm0001 1000ddddddddddd	(disp+Rm)→零扩展→Rn	1	—			○
MOVU.W @(disp12,Rm),Rn	0011nnnnmmmm0001 1001ddddddddddd	(disp×2+Rm)→零扩展→Rn	1	—			○
NOTT	000000001101000	~T→T	1	运算结果			○
PREF @Rn	0000nnnn10000011	(Rn)→操作数高速缓存	1	—		○	○
SWAP.B Rm,Rn	0110nnnnmmmm1000	Rm→低位 2 字节的高低字节交换→Rn	1	—	○	○	○
SWAP.W Rm,Rn	0110nnnnmmmm1001	Rm→交换高低字→Rn	1	—	○	○	○
XTRCT Rm,Rn	0010nnnnmmmm1101	Rm:Rn 的中间 32 位→Rn	1	—	○	○	○

2.4.3 算术运算指令

表 2.12 算术运算指令

指令	指令码	操作	执行状态	T 位	适用指令		
					SH2、SH2E	SH4	SH-2A
ADD Rm,Rn	0011nnnnmmmm1100	$Rn+Rm \rightarrow Rn$	1	—	○	○	○
ADD #imm,Rn	0111nnnniiiiiii	$Rn+imm \rightarrow Rn$	1	—	○	○	○
ADDC Rm,Rn	0011nnnnmmmm1110	$Rn+Rm+T \rightarrow Rn$, 进位 $\rightarrow T$	1	进位	○	○	○
ADDV Rm,Rn	0011nnnnmmmm1111	$Rn+Rm \rightarrow Rn$, 上溢 $\rightarrow T$	1	上溢	○	○	○
CMP/EQ #imm,R0	10001000iiiiiii	当 $R0=imm$ 时 $1 \rightarrow T$, 否则 $0 \rightarrow T$ 。	1	比较结果	○	○	○
CMP/EQ Rm,Rn	0011nnnnmmmm0000	当 $Rn=Rm$ 时 $1 \rightarrow T$, 否则 $0 \rightarrow T$ 。	1	比较结果	○	○	○
CMP/HS Rm,Rn	0011nnnnmmmm0010	当无符号 $Rn \geq Rm$ 时 $1 \rightarrow T$, 否则 $0 \rightarrow T$ 。	1	比较结果	○	○	○
CMP/GE Rm,Rn	0011nnnnmmmm0011	当有符号 $Rn \geq Rm$ 时 $1 \rightarrow T$, 否则 $0 \rightarrow T$ 。	1	比较结果	○	○	○
CMP/HI Rm,Rn	0011nnnnmmmm0110	当无符号 $Rn > Rm$ 时 $1 \rightarrow T$, 否则 $0 \rightarrow T$ 。	1	比较结果	○	○	○
CMP/GT Rm,Rn	0011nnnnmmmm0111	当有符号 $Rn > Rm$ 时 $1 \rightarrow T$, 否则 $0 \rightarrow T$ 。	1	比较结果	○	○	○
CMP/PL Rn	0100nnnn00010101	当 $Rn > 0$ 时 $1 \rightarrow T$, 否则 $0 \rightarrow T$ 。	1	比较结果	○	○	○
CMP/PZ Rn	0100nnnn00010001	当 $Rn \geq 0$ 时 $1 \rightarrow T$, 否则 $0 \rightarrow T$ 。	1	比较结果	○	○	○
CMP/STR Rm,Rn	0010nnnnmmmm1100	当任意字节相等时 $1 \rightarrow T$, 否则 $0 \rightarrow T$ 。	1	比较结果	○	○	○
CLIPS.B Rn	0100nnnn10010001	当 $Rn > (H'000007F)$ 时, ($H'000007F$) $\rightarrow Rn, 1 \rightarrow CS$ 。 当 $Rn < (H'FFFFFF80)$ 时, ($H'FFFFFF80$) $\rightarrow Rn, 1 \rightarrow CS$ 。	1	—			○
CLIPS.W Rn	0100nnnn10010101	当 $Rn > (H'00007FFF)$ 时, ($H'00007FFF$) $\rightarrow Rn, 1 \rightarrow CS$ 。 当 $Rn < (H'FFFF8000)$ 时, ($H'FFFF8000$) $\rightarrow Rn, 1 \rightarrow CS$ 。	1	—			○
CLIPU.B Rn	0100nnnn10000001	当 $Rn > (H'000000FF)$ 时, ($H'000000FF$) $\rightarrow Rn, 1 \rightarrow CS$ 。	1	—			○
CLIPU.W Rn	0100nnnn10000101	当 $Rn > (H'0000FFFF)$ 时, ($H'0000FFFF$) $\rightarrow Rn, 1 \rightarrow CS$ 。	1	—			○
DIV1 Rm,Rn	0011nnnnmmmm0100	单步除法 ($Rn \div Rm$)	1	计算结果	○	○	○
DIV0S Rm,Rn	0010nnnnmmmm0111	Rn 的 MSB $\rightarrow Q$, Rm 的 MSB $\rightarrow M$, $M^A Q \rightarrow T$	1	计算结果	○	○	○
DIV0U	0000000000011001	$0 \rightarrow M/Q/T$	1	0	○	○	○

指令	指令码	操作	执行状态	T 位	适用指令		
					SH2、SH2E	SH4	SH-2A
DIVS R0,Rn	0100nnnn10010100	带符号 $Rn \div R0 \rightarrow Rn$ 32÷32→32 位	36	—			○
DIVU R0,Rn	0100nnnn10000100	无符号 $Rn \div R0 \rightarrow Rn$ 32÷32→32 位	34	—			○
DMULS.L Rm,Rn	0011nnnnmmmm1101	带符号 $Rn \times Rm \rightarrow MACH,$ MACL 32×32→64 位	2	—	○	○	○
DMULU.L Rm,Rn	0011nnnnmmmm0101	无符号 $Rn \times Rm \rightarrow MACH,$ MACL 32×32→64 位	2	—	○	○	○
DT Rn	0100nnnn00010000	$Rn-1 \rightarrow Rn$ 当 Rn 为 0 时 1→T, 否则 0→T。	1	比较结果	○	○	○
EXTS.B Rm,Rn	0110nnnnmmmm1110	将 Rm 的字节进行符号扩展 →Rn	1	—	○	○	○
EXTS.W Rm,Rn	0110nnnnmmmm1111	将 Rm 的字进行符号扩展 →Rn	1	—	○	○	○
EXTU.B Rm,Rn	0110nnnnmmmm1100	将 Rm 的字节进行零扩展 →Rn	1	—	○	○	○
EXTU.W Rm,Rn	0110nnnnmmmm1101	将 Rm 的字进行零扩展 →Rn	1	—	○	○	○
MAC.L @Rm+,@Rn+	0000nnnnmmmm1111	带符号 $(Rn) \times (Rm) + MAC \rightarrow$ MAC 32×32+64→64 位	4	—	○	○	○
MAC.W @Rm+,@Rn+	0100nnnnmmmm1111	带符号 $(Rn) \times (Rm) + MAC \rightarrow$ MAC 16×16+64→64 位	3	—	○	○	○
MUL.L Rm,Rn	0000nnnnmmmm0111	$Rn \times Rm \rightarrow MACL$ 32×32→32 位	2	—	○	○	○
MULR R0,Rn	0100nnnn10000000	$R0 \times Rn \rightarrow Rn$ 32×32→32 位	2				○
MULS.W Rm,Rn	0010nnnnmmmm1111	带符号 $Rn \times Rm \rightarrow MACL$ 16×16→32 位	1	—	○	○	○
MULU.W Rm,Rn	0010nnnnmmmm1110	无符号 $Rn \times Rm \rightarrow MACL$ 16×16→32 位	1	—	○	○	○
NEG Rm,Rn	0110nnnnmmmm1011	$0-Rm \rightarrow Rn$	1	—	○	○	○
NEGC Rm,Rn	0110nnnnmmmm1010	$0-Rm-T \rightarrow Rn,$ 借位 →T	1	借位	○	○	○
SUB Rm,Rn	0011nnnnmmmm1000	$Rn-Rm \rightarrow Rn$	1	—	○	○	○
SUBC Rm,Rn	0011nnnnmmmm1010	$Rn-Rm-T \rightarrow Rn,$ 借位 →T	1	借位	○	○	○
SUBV Rm,Rn	0011nnnnmmmm1011	$Rn-Rm \rightarrow Rn,$ 下溢 →T	1	上溢	○	○	○

2.4.4 逻辑运算指令

表 2.13 逻辑运算指令

指令	指令码	操作	执行状态	T 位	适用指令		
					SH2、SH2E	SH4	SH-2A
AND Rm,Rn	0010nnnnmmmm1001	$Rn \& Rm \rightarrow Rn$	1	—	○	○	○
AND #imm,R0	11001001iiiiiiii	$R0 \& imm \rightarrow R0$	1	—	○	○	○
AND.B #imm,@(R0,GBR)	11001101iiiiiiii	$(R0+GBR) \& imm \rightarrow (R0+GBR)$	3	—	○	○	○
NOT Rm,Rn	0110nnnnmmmm0111	$\sim Rm \rightarrow Rn$	1	—	○	○	○
OR Rm,Rn	0010nnnnmmmm1011	$Rn Rm \rightarrow Rn$	1	—	○	○	○
OR #imm,R0	11001011iiiiiiii	$R0 imm \rightarrow R0$	1	—	○	○	○
OR.B #imm,@(R0,GBR)	11001111iiiiiiii	$(R0+GBR) imm \rightarrow (R0+GBR)$	3	—	○	○	○
TAS.B @Rn	0100nnnn00011011	当 (Rn) 为 0 时 1→T, 否则 0→T。 1→MSB of(Rn)	3	测试结果	○	○	○
TST Rm,Rn	0010nnnnmmmm1000	$Rn \& Rm$ 当结果为 0 时 1→T, 否则 0→T。	1	测试结果	○	○	○
TST #imm,R0	11001000iiiiiiii	$R0 \& imm$ 当结果为 0 时 1→T, 否则 0→T。	1	测试结果	○	○	○
TST.B #imm,@(R0,GBR)	11001100iiiiiiii	$(R0+GBR) \& imm$ 当结果为 0 时 1→T, 否则 0→T。	3	测试结果	○	○	○
XOR Rm,Rn	0010nnnnmmmm1010	$Rn \wedge Rm \rightarrow Rn$	1	—	○	○	○
XOR #imm,R0	11001010iiiiiiii	$R0 \wedge imm \rightarrow R0$	1	—	○	○	○
XOR.B #imm,@(R0,GBR)	11001110iiiiiiii	$(R0+GBR) \wedge imm \rightarrow (R0+GBR)$	3	—	○	○	○

2.4.5 移位指令

表 2.14 移位指令

指令	指令码	操作	执行状态	T 位	适用指令		
					SH2、SH2E	SH4	SH-2A
ROTL Rn	0100nnnn00000100	$T \leftarrow Rn \leftarrow \text{MSB}$	1	MSB	○	○	○
ROTR Rn	0100nnnn00000101	$\text{LSB} \rightarrow Rn \rightarrow T$	1	LSB	○	○	○
ROTCL Rn	0100nnnn00100100	$T \leftarrow Rn \leftarrow T$	1	MSB	○	○	○
ROTCR Rn	0100nnnn00100101	$T \rightarrow Rn \rightarrow T$	1	LSB	○	○	○
SHAD Rm,Rn	0100nnnnmmmm1100	当 $Rm \geq 0$ 时 $Rn \ll Rm \rightarrow Rn$, 当 $Rm < 0$ 时 $Rn \gg Rm \rightarrow [MSB \rightarrow Rn]$ 。	1	—		○	○
SHAL Rn	0100nnnn00100000	$T \leftarrow Rn \leftarrow 0$	1	MSB	○	○	○
SHAR Rn	0100nnnn00100001	$MSB \rightarrow Rn \rightarrow T$	1	LSB	○	○	○
SHLD Rm,Rn	0100nnnnmmmm1101	当 $Rm \geq 0$ 时 $Rn \ll Rm \rightarrow Rn$, 当 $Rm < 0$ 时 $Rn \gg Rm \rightarrow [0 \rightarrow Rn]$ 。	1	—		○	○
SHLL Rn	0100nnnn00000000	$T \leftarrow Rn \leftarrow 0$	1	MSB	○	○	○
SHLR Rn	0100nnnn00000001	$0 \rightarrow Rn \rightarrow T$	1	LSB	○	○	○
SHLL2 Rn	0100nnnn00001000	$Rn \ll 2 \rightarrow Rn$	1	—	○	○	○
SHLR2 Rn	0100nnnn00001001	$Rn \gg 2 \rightarrow Rn$	1	—	○	○	○
SHLL8 Rn	0100nnnn00011000	$Rn \ll 8 \rightarrow Rn$	1	—	○	○	○
SHLR8 Rn	0100nnnn00011001	$Rn \gg 8 \rightarrow Rn$	1	—	○	○	○
SHLL16 Rn	0100nnnn00101000	$Rn \ll 16 \rightarrow Rn$	1	—	○	○	○
SHLR16 Rn	0100nnnn00101001	$Rn \gg 16 \rightarrow Rn$	1	—	○	○	○

2.4.6 转移指令

表 2.15 转移指令

指令	指令码	操作	执行状态	T 位	适用指令		
					SH2、SH2E	SH4	SH-2A
BF label	10001011dddddddd	当 T=0 时 $\text{disp} \times 2 + \text{PC} \rightarrow \text{PC}$, 当 T=1 时 nop。	3/1*	—	○	○	○
BF/S label	10001111dddddddd	延迟转移 当 T=0 时 $\text{disp} \times 2 + \text{PC} \rightarrow \text{PC}$, 当 T=1 时 nop。	2/1*	—	○	○	○
BT label	10001001dddddddd	当 T=1 时 $\text{disp} \times 2 + \text{PC} \rightarrow \text{PC}$, 当 T=0 时 nop。	3/1*	—	○	○	○
BT/S label	10001101dddddddd	延迟转移 当 T=1 时 $\text{disp} \times 2 + \text{PC} \rightarrow \text{PC}$, 当 T=0 时 nop。	2/1*	—	○	○	○
BRA label	1010dddddddddddd	延迟转移 $\text{disp} \times 2 + \text{PC} \rightarrow \text{PC}$	2	—	○	○	○
BRAF Rm	0000mmmm00100011	延迟转移 $\text{Rm} + \text{PC} \rightarrow \text{PC}$	2	—	○	○	○
BSR label	1011dddddddddddd	延迟转移 $\text{PC} \rightarrow \text{PR}$, $\text{disp} \times 2 + \text{PC} \rightarrow \text{PC}$	2	—	○	○	○
BSRF Rm	0000mmmm00000011	延迟转移 $\text{PC} \rightarrow \text{PR}$, $\text{Rm} + \text{PC} \rightarrow \text{PC}$	2	—	○	○	○
JMP @Rm	0100mmmm00101011	延迟转移 $\text{Rm} \rightarrow \text{PC}$	2	—	○	○	○
JSR @Rm	0100mmmm00001011	延迟转移 $\text{PC} \rightarrow \text{PR}$, $\text{Rm} \rightarrow \text{PC}$	2	—	○	○	○
JSR/N @Rm	0100mmmm01001011	$\text{PC} - 2 \rightarrow \text{PR}$, $\text{Rm} \rightarrow \text{PC}$	3	—			○
JSR/N @@(disp8,TBR)	10000011dddddddd	$\text{PC} - 2 \rightarrow \text{PR}$, $(\text{disp} \times 4 + \text{TBR}) \rightarrow \text{PC}$	5	—			○
RTS	0000000000001011	延迟转移 $\text{PR} \rightarrow \text{PC}$	2	—	○	○	○
RTS/N	000000001101011	$\text{PR} \rightarrow \text{PC}$	3	—			○
RTV/N Rm	0000mmmm01111011	$\text{Rm} \rightarrow \text{R0}$, $\text{PR} \rightarrow \text{PC}$	3	—			○

【注】 * 在不转移时为 1 个状态。

2.4.7 系统控制指令

表 2.16 系统控制指令

指令	指令码	操作	执行状态	T 位	适用指令		
					SH2、SH2E	SH4	SH-2A
CLRT	000000000001000	0→T	1	0	○	○	○
CLRMACH	000000000101000	0→MACH,MACL	1	—	○	○	○
LDBANK @Rm,R0	0100mmmm11100101	(指定寄存器组入口) →R0	6	—			○
LDC Rm,SR	0100mmmm00001110	Rm→SR	3	LSB	○	○	○
LDC Rm,TBR	0100mmmm01001010	Rm→TBR	1	—			○
LDC Rm,GBR	0100mmmm00011110	Rm→GBR	1	—	○	○	○
LDC Rm,VBR	0100mmmm00101110	Rm→VBR	1	—	○	○	○
LDC.L @Rm+,SR	0100mmmm00000111	(Rm)→SR, Rm+4→Rm	5	LSB	○	○	○
LDC.L @Rm+,GBR	0100mmmm00010111	(Rm)→GBR, Rm+4→Rm	1	—	○	○	○
LDC.L @Rm+,VBR	0100mmmm00100111	(Rm)→VBR, Rm+4→Rm	1	—	○	○	○
LDS Rm,MACH	0100mmmm00001010	Rm→MACH	1	—	○	○	○
LDS Rm,MACL	0100mmmm00011010	Rm→MACL	1	—	○	○	○
LDS Rm,PR	0100mmmm00101010	Rm→PR	1	—	○	○	○
LDS.L @Rm+,MACH	0100mmmm00000110	(Rm)→MACH, Rm+4→Rm	1	—	○	○	○
LDS.L @Rm+,MACL	0100mmmm00010110	(Rm)→MACL, Rm+4→Rm	1	—	○	○	○
LDS.L @Rm+,PR	0100mmmm00100110	(Rm)→PR, Rm+4→Rm	1	—	○	○	○
NOP	000000000001001	无操作	1	—	○	○	○
RESBANK	000000001011011	存储体 →R0 ~ R14, GBR, MACH, MACL, PR	9*	—			○
RTE	000000000101011	延迟转移 堆栈区 →PC/SR	6	—	○	○	○
SETT	000000000011000	1→T	1	1	○	○	○
SLEEP	000000000011011	睡眠	5	—	○	○	○
STBANK R0,@Rn	0100nnnn11100001	R0→(指定寄存器组入口)	7	—			○
STC SR,Rn	0000nnnn00000010	SR→Rn	2	—	○	○	○
STC TBR,Rn	0000nnnn01001010	TBR→Rn	1	—			○
STC GBR,Rn	0000nnnn00010010	GBR→Rn	1	—	○	○	○
STC VBR,Rn	0000nnnn00100010	VBR→Rn	1	—	○	○	○
STC.L SR,@-Rn	0100nnnn00000011	Rn-4→Rn, SR→(Rn)	2	—	○	○	○
STC.L GBR,@-Rn	0100nnnn00010011	Rn-4→Rn, GBR→(Rn)	1	—	○	○	○
STC.L VBR,@-Rn	0100nnnn00100011	Rn-4→Rn, VBR→(Rn)	1	—	○	○	○
STS MACH,Rn	0000nnnn00001010	MACH→Rn	1	—	○	○	○
STS MACL,Rn	0000nnnn00011010	MACL→Rn	1	—	○	○	○
STS PR,Rn	0000nnnn00101010	PR→Rn	1	—	○	○	○
STS.L MACH,@-Rn	0100nnnn00000010	Rn-4→Rn, MACH→(Rn)	1	—	○	○	○
STS.L MACL,@-Rn	0100nnnn00010010	Rn-4→Rn, MACL→(Rn)	1	—	○	○	○
STS.L PR,@-Rn	0100nnnn00100010	Rn-4→Rn, PR→(Rn)	1	—	○	○	○

指令	指令码	操作	执行状态	T 位	适用指令		
					SH2、SH2E	SH4	SH-2A
TRAPA #imm	11000011iiiiiiii	PC/SR→堆栈区, (imm×4+VBR)→PC	5	—	○	○	○

【注】 有关指令的执行状态

表中所示的执行状态为最小值。实际上根据以下条件，指令执行的状态数将会增加。

(1) 当取指令和数据存取发生竞争时

(2) 当装入指令（存储器→寄存器）的目标寄存器和紧接着的指令使用的寄存器相同时

* 当存储体发生上溢时，状态数为 19。

2.4.8 浮点运算指令

表 2.17 浮点运算指令

指令	指令码	操作	执行状态	T 位	适用指令		
					SH2E	SH4	SH-2A/ SH2A- FPU
FABS FRn	1111nnnn01011101	FRn →FRn	1	—	○	○	○
FABS DRn	1111nnn001011101	DRn →DRn	1	—		○	○
FADD FRm,FRn	1111nnnnmmmm0000	FRn+FRm→FRn	1	—	○	○	○
FADD DRm,DRn	1111nnn0mmmm0000	DRn+DRm→DRn	6	—		○	○
FCMP/EQ FRm,FRn	1111nnnnmmmm0100	(FRn=FRm)? 1:0→T	1	比较结果	○	○	○
FCMP/EQ DRm,DRn	1111nnn0mmmm00100	(DRn=DRm)? 1:0→T	2	比较结果		○	○
FCMP/GT FRm,FRn	1111nnnnmmmm0101	(FRn > FRm)? 1:0→T	1	比较结果	○	○	○
FCMP/GT DRm,DRn	1111nnn0mmmm00101	(DRn > DRm)? 1:0→T	2	比较结果		○	○
FCNVDS DRm,FPUL	1111mmmm010111101	(float)DRm→FPUL	2	—		○	○
FCNVSD FPUL,DRn	1111nnn010101101	(double)FPUL→DRn	2	—		○	○
FDIV FRm,FRn	1111nnnnmmmm0011	FRn/FRm→FRn	10	—	○	○	○
FDIV DRm,DRn	1111nnn0mmmm00011	DRn/DRm→DRn	23	—		○	○
FLDI0 FRn	1111nnnn10001101	0×00000000→FRn	1	—	○	○	○
FLDI1 FRn	1111nnnn10011101	0×3F800000→FRn	1	—	○	○	○
FLDS FRm,FPUL	1111mmmm00011101	FRm→FPUL	1	—	○	○	○
FLOAT FPUL,FRn	1111nnnn00101101	(float)FPUL→FRn	1	—	○	○	○
FLOAT FPUL,DRn	1111nnn000101101	(double)FPUL→DRn	2	—		○	○
FMAC FR0,FRm,FRn	1111nnnnmmmm1110	FR0×FRm+FRn→FRn	1	—	○	○	○
FMOV FRm,FRn	1111nnnnmmmm1100	FRm→FRn	1	—	○	○	○
FMOV DRm,DRn	1111nnn0mmmm01100	DRm→DRn	2	—		○	○
FMOV.S @(R0,Rm),FRn	1111nnnnmmmm0110	(R0+Rm)→FRn	1	—	○	○	○
FMOV.D @(R0,Rm),DRn	1111nnn0mmmm0110	(R0+Rm)→DRn	2	—		○	○
FMOV.S @Rm+,FRn	1111nnnnmmmm1001	(Rm)→FRn,Rm+=4	1	—	○	○	○

指令	指令码	操作	执行 状态	T 位	适用指令		
					SH2E	SH4	SH-2A/ SH2A- FPU
FMOV.D @Rm+,DRn	1111nnn0mmmm1001	(Rm)→DRn,Rm+=8	2	—		○	○
FMOV.S @Rm,FRn	1111nnnnmmmm1000	(Rm)→FRn	1	—	○	○	○
FMOV.D @Rm,DRn	1111nnn0mmmm1000	(Rm)→DRn	2	—		○	○
FMOV.S @(disp12,Rm),FRn	0011nnnnmmmm0001 0111ddddddddddd	(disp×4+Rm)→FRn	1	—			○
FMOV.D @(disp12,Rm),DRn	0011nnn0mmmm0001 0111ddddddddddd	(disp×8+Rm)→DRn	2	—			○
FMOV.S FRm,@(R0,Rn)	1111nnnnmmmm0111	FRm→(R0+Rn)	1	—	○	○	○
FMOV.D DRm,@(R0,Rn)	1111nnnnmmmm0011	DRm→(R0+Rn)	2	—		○	○
FMOV.S FRm,@-Rn	1111nnnnmmmm1011	Rn=4,FRm→(Rn)	1	—	○	○	○
FMOV.D DRm,@-Rn	1111nnnnmmmm0101	Rn=8,DRm→(Rn)	2	—		○	○
FMOV.S FRm,@Rn	1111nnnnmmmm1010	FRm→(Rn)	1	—	○	○	○
FMOV.D DRm,@Rn	1111nnnnmmmm0100	DRm→(Rn)	2	—		○	○
FMOV.S FRm,@(disp12,Rn)	0011nnnnmmmm0001 0011ddddddddddd	FRm→(disp×4+Rn)	1	—			○
FMOV.D DRm,@(disp12,Rn)	0011nnnnmmmm0000 0011ddddddddddd	DRm→(disp×8+Rn)	2	—			○
FMUL FRm,FRn	1111nnnnmmmm0010	FRn×FRm→FRn	1	—	○	○	○
FMUL DRm,DRn	1111nnn0mmmm00010	DRn×DRm→DRn	6	—		○	○
FNEG FRn	1111nnnn01001101	-FRn→FRn	1	—	○	○	○
FNEG DRn	1111nnn001001101	-DRn→DRn	1	—		○	○
FSCHG	111100111111101	FPSCR.SZ=FPSCR.SZ	1	—		○	○
FSQRT FRn	1111nnnn01101101	√FRn→FRn	9	—		○	○
FSQRT DRn	1111nnn001101101	√DRn→DRn	22	—		○	○
FSTS FPUL,FRn	1111nnnn00001101	FPUL→FRn	1	—	○	○	○
FSUB FRm,FRn	1111nnnnmmmm0001	FRn-FRm→FRn	1	—	○	○	○
FSUB DRm,DRn	1111nnn0mmmm00001	DRn-DRm→DRn	6	—		○	○
FTRC FRm,FPUL	1111mmmm00111101	(long)FRm→FPUL	1	—	○	○	○
FTRC DRm,FPUL	1111mmm000111101	(long)DRm→FPUL	2	—		○	○

2.4.9 FPU 相关的 CPU 指令

表 2.18 FPU 相关的 CPU 指令

指令	指令码	操作	执行状态	T 位	适用指令			
					SH2E	SH4	SH-2A/ SH2A- FPU	
LDS	Rm,FPSCR	0100mmmm01101010	Rm→FPSCR	1	—	○	○	○
LDS	Rm,FPUL	0100mmmm01011010	Rm→FPUL	1	—	○	○	○
LDS.L	@Rm+,FPSCR	0100mmmm01100110	(Rm)→FPSCR,Rm+=4	1	—	○	○	○
LDS.L	@Rm+,FPUL	0100mmmm01010110	(Rm)→FPUL,Rm+=4	1	—	○	○	○
STS	FPSCR,Rn	0000nnnn01101010	FPSCR→Rn	1	—	○	○	○
STS	FPUL,Rn	0000nnnn01011010	FPUL→Rn	1	—	○	○	○
STS.L	FPSCR,@-Rn	0100nnnn01100010	Rn-=4,FPSCR→(Rn)	1	—	○	○	○
STS.L	FPUL,@-Rn	0100nnnn01010010	Rn-=4,FPUL→(Rn)	1	—	○	○	○

2.4.10 位操作指令

表 2.19 位操作指令

指令	指令码	操作	执行状态	T 位	适用指令			
					SH2、 SH2E	SH4	SH-2A	
BAND.B	#imm3,@(disp12,Rn)	0011nnnn0iii1001 0100ddddddddddd	(imm of (disp+Rn))&T → T	3	运算 结果			○
BANDNOT.B	#imm3,@(disp12,Rn)	0011nnnn0iii1001 1100ddddddddddd	~(imm of (disp+Rn))&T → T	3	运算 结果			○
BCLR.B	#imm3,@(disp12,Rn)	0011nnnn0iii1001 0000ddddddddddd	0→(imm of (disp+Rn))	3	—			○
BCLR	#imm3,Rn	10000110nnnn0iii	0→imm of Rn	1	—			○
BLD.B	#imm3,@(disp12,Rn)	0011nnnn0iii1001 0011ddddddddddd	(imm of (disp+Rn))→T	3	运算 结果			○
BLD	#imm3,Rn	10000111nnnnliii	imm of Rn→T	1	运算 结果			○
BLDNOT.B	#imm3,@(disp12,Rn)	0011nnnn0iii1001 1011ddddddddddd	~(imm of (disp+Rn))→T	3	运算 结果			○
BOR.B	#imm3,@(disp12,Rn)	0011nnnn0iii1001 0101ddddddddddd	(imm of (disp+Rn)) T→T	3	运算 结果			○
BORNOT.B	#imm3,@(disp12,Rn)	0011nnnn0iii1001 1101ddddddddddd	~(imm of (disp+Rn)) T→T	3	运算 结果			○
BSET.B	#imm3,@(disp12,Rn)	0011nnnn0iii1001 0001ddddddddddd	1→(imm of (disp+Rn))	3	—			○
BSET	#imm3,Rn	10000110nnnnliii	1→imm of Rn	1	—			○
BST.B	#imm3,@(disp12,Rn)	0011nnnn0iii1001 0010ddddddddddd	T→(imm of (disp+Rn))	3	—			○
BST	#imm3,Rn	10000111nnnn0iii	T→imm of Rn	1	—			○
BXOR.B	#imm3,@(disp12,Rn)	0011nnnn0iii1001 0110ddddddddddd	(imm of (disp+Rn)) ^ T→T	3	运算 结果			○

2.5 处理状态

CPU 的处理状态有复位状态、异常处理状态、总线权释放状态、程序执行状态和低功耗状态共 5 种。状态间的转移如图 2.6 所示。

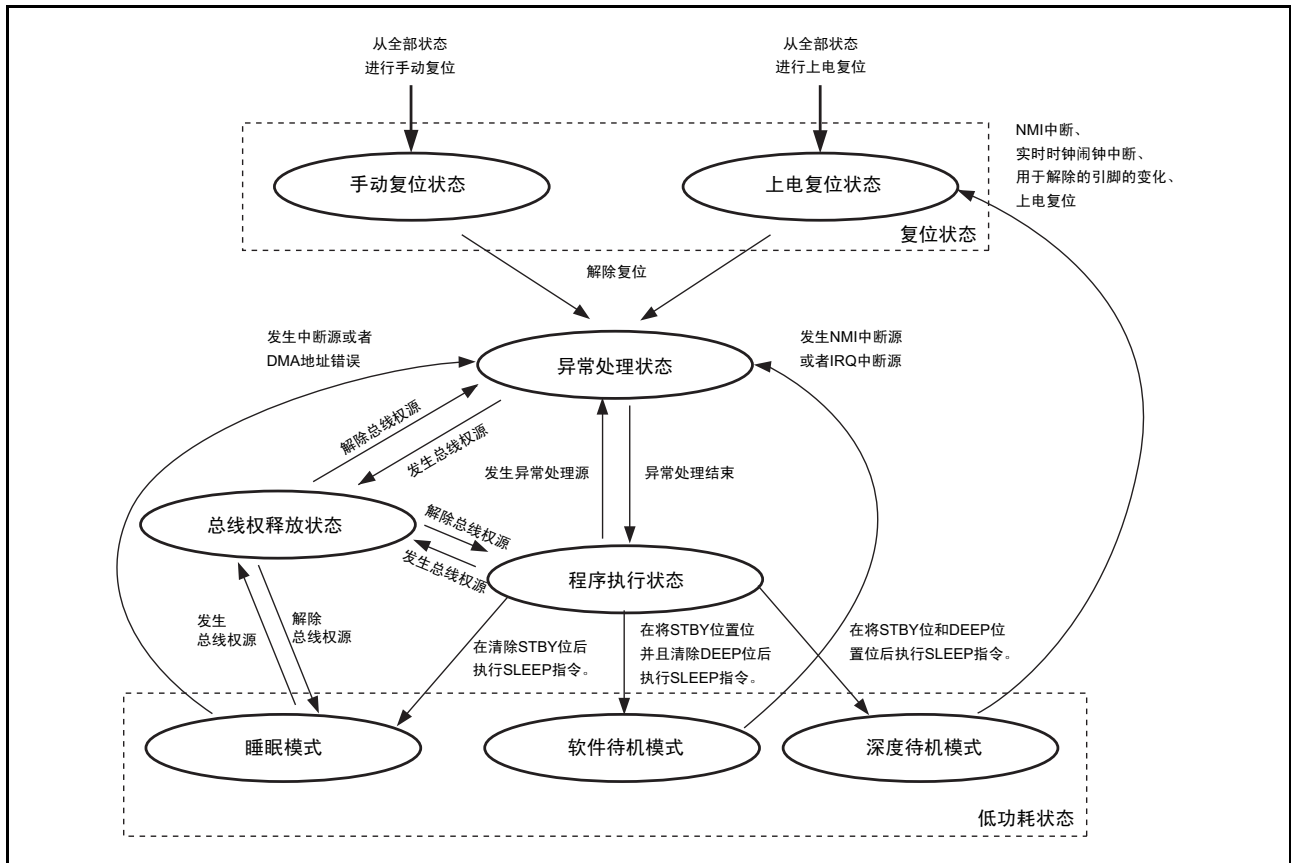


图 2.6 处理状态的状态转移图

(1) 复位状态

这是对 CPU 进行复位的状态。复位有上电复位和手动复位共 2 种。

(2) 异常处理状态

这是 CPU 因复位和中断等异常处理源而改变处理状态流程时的过渡状态。

在复位时，从异常处理向量表分别取出并且保存作为程序计数器（PC）初始值的执行起始地址和堆栈指针（SP）的初始值，转移到起始地址后开始执行程序。

在发生中断时，参照 SP，然后将 PC 和状态寄存器（SR）保存到堆栈区。从异常处理向量表取出异常服务程序的起始地址，转移到该地址后开始执行程序。

之后，处理状态为程序执行状态。

(3) 程序执行状态

这是 CPU 依次执行程序的状态。

(4) 低功耗状态

这是 CPU 停止工作并且处于低功耗的状态。通过睡眠指令进入睡眠模式、软件待机模式或者深度待机模式。

(5) 总线权释放状态

这是 CPU 对请求总线权的设备释放总线的状态。

3. 浮点单元 (FPU)

3.1 特点

FPU 有以下的特点：

- 符合 IEEE754 规格。
- 16 个单精度浮点寄存器（也能作为 8 个双精度寄存器进行参照）
- 2 种舍入模式：向接近的值舍入和向 0 方向舍入。
- 非规格化数处理模式：刷新为“0”。
- 5 个异常源：
无效运算、被零除运算、上溢、下溢、不精确
- 综合指令：
单精度、双精度、系统控制

3.2 数据格式

3.2.1 浮点格式

浮点由以下 3 个字段构成：

- 符号 (s)
- 指数 (e)
- 尾数 (f)

本 LSI 能使用图 3.1 和图 3.2 所示的格式处理单精度和双精度浮点。

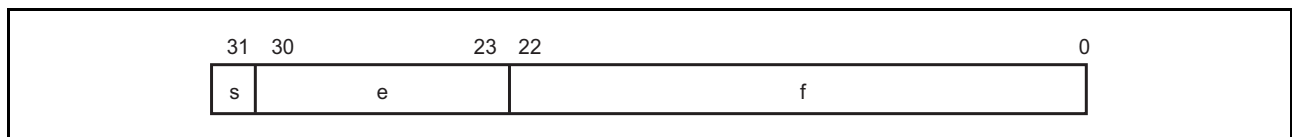


图 3.1 单精度浮点格式

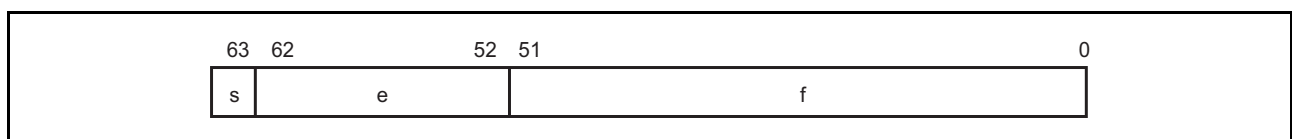


图 3.2 双精度浮点格式

指数带偏置，表示如下：

$$e = E + \text{bias}$$

无偏置的指数 E 的范围是 $E_{\min}-1 \sim E_{\max}+1$ 。 $E_{\min}-1$ 和 $E_{\max}+1$ 两个值的区别如下： $E_{\min}-1$ 表示 0（正和负的符号）和非规格化数， $E_{\max}+1$ 表示正或者负无穷大和非数（NaN）。 E_{\min} 和 E_{\max} 的值如表 3.1 所示。

表 3.1 浮点的格式和参数

参数	单精度	双精度
总位宽	32 位	64 位
符号位	1 位	1 位
指数字段	8 位	11 位
尾数字段	23 位	52 位
精度	24 位	53 位
偏置	+127	+1023
E_{\max}	+127	+1023
E_{\min}	-126	-1022

浮点数值 v 的决定如下：

当 $E=E_{\max}+1$ 并且 $f \neq 0$ 时，与符号 s 无关 v 为非数 (NaN)。

当 $E=E_{\max}+1$ 并且 $f=0$ 时， v 为 $(-1)^s$ (无穷大) “正或者负无穷大”。

当 $E_{\min} \leq E \leq E_{\max}$ 时， v 为 $(-1)^s 2^E (1.f)$ “规格化数”。

当 $E=E_{\min}-1$ 并且 $f \neq 0$ 时， v 为 $(-1)^s 2^{E_{\min}} (0.f)$ “非规格化数”。

当 $E=E_{\min}-1$ 并且 $f=0$ 时， v 为 $(-1)^s 0$ “正或者负的 0”。

16 进制的各数的范围如表 3.2 所示。

表 3.2 浮点的范围

类型	单精度	双精度
信令非数	H'7FFF FFFF ~ H'7FC0 0000	H'7FFF FFFF FFFF FFFF ~ H'7FF8 0000 0000 0000
静态非数	H'7FBF FFFF ~ H'7F80 0001	H'7FF7 FFFF FFFF FFFF ~ H'7FF0 0000 0000 0001
正无穷大	H'7F80 0000	H'7FF0 0000 0000 0000
正的规格化数	H'7F7F FFFF ~ H'0080 0000	H'7FEF FFFF FFFF FFFF ~ H'0010 0000 0000 0000
正的非规格化数	H'007F FFFF ~ H'0000 0001	H'000F FFFF FFFF FFFF ~ H'0000 0000 0000 0001
正 0	H'0000 0000	H'0000 0000 0000 0000
负 0	H'8000 0000	H'8000 0000 0000 0000
负的非规格化数	H'8000 0001 ~ H'807F FFFF	H'8000 0000 0000 0001 ~ H'800F FFFF FFFF FFFF
负的规格化数	H'8080 0000 ~ H'FF7F FFFF	H'8010 0000 0000 0000 ~ H'FFEF FFFF FFFF FFFF
负无穷大	H'FF80 0000	H'FFF0 0000 0000 0000
静态非数	H'FF80 0001 ~ H'FFBF FFFF	H'FFF0 0000 0000 0001 ~ H'FFF7 FFFF FFFF FFFF
信令非数	H'FFC0 0000 ~ H'FFFF FFFF	H'FFF8 0000 0000 0000 ~ H'FFFF FFFF FFFF FFFF

3.2.2 非数 (NaN)

非数 (NaN) 的位格式如图 3.3 所示。以下情况时值为 NaN。

- 符号位: Don't care
- 指数字段: 全部位为 “1”。
- 尾数字段: 至少 1 位为 “1”。

当尾数字段的 MSB 为 “1” 时, NaN 为信令非数 (sNaN); 当尾数字段的 MSB 为 “0” 时, NaN 为静态非数 (qNaN)。

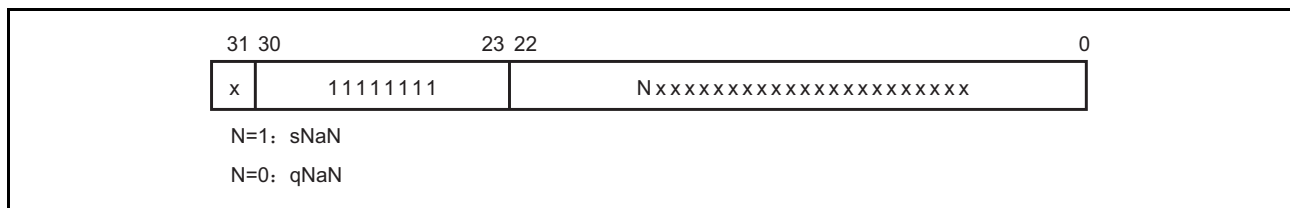


图 3.3 单精度的 NaN 位格式

在生成除复制、FABS 和 FNEG 以外的浮点值的运算时输入 sNaN。

- 当 FPSCR 的 EN.V 位为 “0” 时, 运算结果 (输出) 为 qNaN。
- 当 FPSCR 的 EN.V 位为 “1” 时, 发生因无效运算异常引起的 FPU 异常处理。此时, 不改变运算目标寄存器的内容。

如果在生成浮点值的运算时输入 qNaN 并且在此运算中未输入 sNaN, 就总是输出 qNaN, 与 FPSCR 的 EN.V 位的设定无关。此时不发生异常。

作为运算结果而生成的 qNaN 总是为以下的值:

- 单精度 qNaN: H'7FBF FFFF
- 双精度 qNaN: H'7FF7 FFFF FFFF FFFF

有关输入非数 (NaN) 时的浮点运算的详细内容, 请参照各指令的说明。

3.2.3 非规格化数

非规格化数的浮点值用指数字段为 “0” 而尾数字段不为 “0” 的值来表示。

在 SH2A-FPU 中, 因为状态寄存器 FPSCR 的 DN 位总是为 “1”, 所以非规格化数 (源操作数或者运算结果) 在生成 (复制、FNEG、FABS 除外的运算) 值的浮点运算时总是被刷新为 “0”。

有关输入非规格化数时的浮点运算的详细内容, 请参照各指令的说明。

3.3 寄存器的说明

3.3.1 浮点寄存器

浮点寄存器的结构如图 3.4 所示，有 16 个 32 位浮点寄存器 FPR0 ~ FPR15。这 16 个寄存器作为 FR0 ~ FR15、DR0/2/4/6/8/10/12/14 而被参照。FPR_n 和参照名的对应取决于 FPSCR 的 PR 位和 SZ 位。请参照图 3.4。

1. 浮点寄存器：FPR_i (16个寄存器)
FPR0、FPR1、FPR2、FPR3、FPR4、FPR5、FPR6、FPR7、
FPR8、FPR9、FPR10、FPR11、FPR12、FPR13、FPR14、FPR15
2. 单精度浮点寄存器：FR_i (16个寄存器)
FR0~FR15被分配到FPR0~FPR15。
3. 双精度浮点寄存器或者单精度浮点寄存器对 DR_i (8个寄存器)
DR寄存器由2个FR寄存器构成。
DR0={FR0, FR1}、DR2={FR2, FR3}、DR4={FR4, FR5}、DR6={FR6, FR7}、
DR8={FR8, FR9}、DR10={FR10, FR11}、DR12={FR12, FR13}、DR14={FR14, FR15}

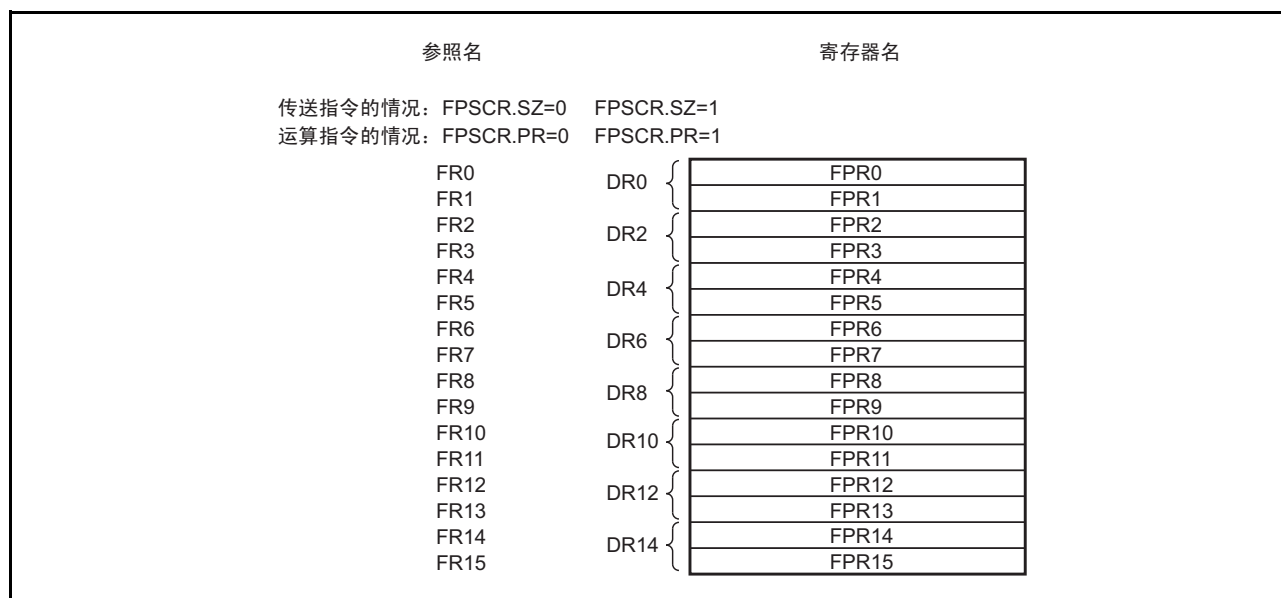


图 3.4 浮点寄存器

3.3.2 浮点状态 / 控制寄存器 (FPSCR)

FPSCR 是 32 位寄存器，控制浮点指令、设定 FPU 异常以及选择舍入的使用方法。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	QIS	—	SZ	PR	DN	Cause	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Cause				Enable						Flag				RM1	RM0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 23	—	全 0	R	保留位 读写值都为“0”。
22	QIS	0	R/W	非数处理模式 0: 直接处理将 qNaN 或者 $\pm\infty$ 1: 将 qNaN 或者 $\pm\infty$ 进行与 sNaN 同样的处理 (只有在 FPSCR.Enable.V 为“1”时有效)
21	—	0	R	保留位 读写值都为“0”。
20	SZ	0	R/W	传送长度模式 0: FMOV 指令的数据长度为 32 位 1: FMOV 指令的数据长度为 32 位对 (64 位)
19	PR	0	R/W	精度模式 0: 以单精度运算执行浮点指令 1: 以双精度运算执行浮点指令
18	DN	1	R	非规格化模式 (在 SH2A-FPU 中, 总是固定为“1”) 1: 将非规格化数作为“0”处理
17 ~ 12	Cause	H'00	R/W	FPU 异常源字段
11 ~ 7	Enable	H'00	R/W	FPU 异常允许字段
6 ~ 2	Flag	H'00	R/W	FPU 异常标志字段 如果执行 FPU 运算指令, 就在最初将 FPU 异常源字段清“0”。然后, 如果发生因浮点运算引起的 FPU 异常, 就将 FPU 异常源字段和 FPU 异常标志字段的对应位置“1”。FPU 异常标志字段将“1”的状态保持到被软件清“0”为止。 如果将 FPU 异常允许字段的对应位置“1”, 就进行 FPU 异常处理。 有关各字段位的分配, 请参照表 3.3。
1	RM1	0	R/W	舍入模式
0	RM0	1	R/W	选择舍入的方法。 00: 向接近的值舍入 01: 向 0 方向舍入 10: 保留 11: 保留

表 3.3 与 FPU 异常处理有关的位的分配

		FPU 错误 (E)	无效运算 (V)	被零除 (Z)	上溢 (O)	下溢 (U)	不精确 (I)
Cause	FPU 异常源字段	bit17	bit16	bit15	bit14	bit13	bit12
Enable	FPU 异常允许字段	无	bit11	bit10	bit9	bit8	bit7
Flag	FPU 异常标志字段	无	bit6	bit5	bit4	bit3	bit2

【注】 在 SH2A-FPU 中，不发生 FPU 错误。

3.3.3 浮点通信寄存器 (FPUL)

通过 FPUL 进行 FPU 和 CPU 之间的信息传递。FPUL 是 32 位系统寄存器，CPU 也能通过 LDS 指令和 STS 指令进行存取。例如，将保存在通用寄存器 R1 的整数转换为单精度浮点的处理流程如下：

R1 → (LDS 指令) → FPUL → (单精度 FLOAT 指令) → FR1

3.4 舍入

在浮点指令中，从中间结果生成最终运算结果时进行舍入。因此，FMAC 等组合指令的结果不同于只使用 FADD、FSUB、FMUL 等基本指令的结果，这是因为 FMAC 进行 1 次舍入，而 FADD、FSUB、FMUL 进行 2 次舍入，舍入的次数不同。

舍入有 2 种方法，使用方法取决于 FPSCR 的 RM 字段。

FPSCR.RM[1:0]=00：向接近的值舍入

FPSCR.RM[1:0]=01：向 0 方向舍入

(1) 向接近的值舍入

运算结果被舍入为最近的能表现的值。如果有 2 个最近的能表现的值，就选择 LSB 为“0”的值。

如果舍入前的值大于等于 $2^{E_{max}(2-2-p)}$ ，结果就为和舍入前符号相同的无穷大。在此， E_{max} 和 p 在单精度时分别为 127 和 24，在双精度时分别为 1023 和 53。

(2) 向 0 方向舍入

舍去舍入位以后的位（将舍入前的值）。

但是，如果舍入前的值大于能表现的最大绝对值，就为能表现的最大绝对值。

3.5 FPU 异常

3.5.1 FPU 异常源

在浮点指令中，可能发生 FPU 异常，异常源如下：

- FPU 错误 (E) : 当 FPSCR 的 DN 位为 “0” 并且输入非规格化数时 (在 SH2A-FPU 中不发生)
- 无效运算 (V) : 当发生如同 NaN 输入的无效运算时
- 被零除运算 (Z) : 除数为 “0” 的除法运算
- 上溢 (O) : 当运算结果发生上溢时
- 下溢 (U) : 当运算结果发生下溢时
- 不精确异常 (I) : 当发生上溢、下溢或者舍入时

FPSCR 的 FPU 异常源字段包含上叙 E、V、Z、O、U、I 的全部对应位，FPSCR 的标志和允许字段包含 V、Z、O、U、I 的对应位，但是不包含 E 的对应位。这样就无法禁止 FPU 错误。

如果发生 FPU 异常，FPU 异常源字段的对应位就被置 “1”，并且给 FPU 异常源字段的对应位加 1。如果不发生 FPU 异常，FPU 异常源字段的对应位就被清 “0”，而 FPU 异常标志字段的对应位不变。

3.5.2 FPU 异常处理

在以下情况下发生 FPU 异常：

- FPU 错误 (E) : 当 FPSCR 的 DN 位为 “0” 并且输入非规格化数时 (SH2A-FPU 中不发生)
- 无效运算 (V) : 当 FPSCR 的 Enable 的 V 位为 “1” 并且进行无效运算时
- 被零除运算 (Z) : 当 FPSCR 的 Enable 的 Z 位为 “1” 并且进行除数为 “0” 的除法运算时
- 上溢 (O) : FPSCR 的 Enable 的 O 位为 “1” 并且运算结果可能发生上溢的指令
- 下溢 (U) : FPSCR 的 Enable 的 U 位为 “1” 并且运算结果可能发生下溢的指令
- 不精确异常 (I) : FPSCR 的 Enable 的 I 位为 “1” 并且运算结果可能不精确的指令

在各指令的说明中叙述了浮点运算引起各种异常处理的可能性。浮点运算引起的全部异常事件分配为同一个异常事件。通过软件读 FPSCR 并且解释所保持的信息，决定浮点运算引起异常的含义。在发生 FPU 异常处理时，不更改目标寄存器的内容。

除上述以外，还将 V、Z、O、U、I 的对应位置 “1”，并且作为运算结果，生成默认值。

- 无效运算 (V) : 作为结果，生成 qNaN。
- 被零除运算 (Z) : 生成带符号 (和舍入前的符号相同) 的无穷大。
- 上溢 (O) :
 - 当向 0 方向舍入时，生成带符号 (和舍入前的符号相同) 的最大规格化数。
 - 当向接近的值舍入时，生成带符号 (和舍入前的符号相同) 的无穷大。
- 下溢 (U) : 生成带符号 (和舍入前的符号相同) 的 “0”。
- 不精确异常 (I) : 生成不精确的结果。

4. 引导模式

本 LSI 能从连接 CS0 空间的存储器、NAND 闪存、串行闪存、SD 控制器内部 NAND 闪存以及 MMC 控制器内部 NAND 闪存引导。

4.1 特点

- 6个引导模式
 - 引导模式0: 从连接CS0空间的存储器（总线宽度为16bit）引导。
 - 引导模式1: 从连接CS0空间的存储器（总线宽度为32bit）引导。
 - 引导模式2: 从NAND闪存引导。
 - 引导模式3: 从串行闪存引导。
 - 引导模式4: 从SD控制器的内部NAND闪存*引导。
 - *能从SD Specification Part 1 eSD Addendum(Version 2.10)规格定义的“eSD (Embedded SD)”引导。
 - 引导模式5: 从MMC控制器的内部NAND闪存*引导。
 - *只能从对应JEDEC STANDARD JESD84 A44(MMCA 4.4)规格的 Boot operation mode 的 eMMC设备引导（不能从MMC卡引导）。

4.2 引导模式和引脚功能设定

本 LSI 能在 $\overline{\text{RES}}$ 为“L”的状态下使用外部引脚来决定引导模式。决定引导模式的外部引脚设定如表 4.1 所示。

表 4.1 外部引脚（MD_BOOT2 ~ 0）设定和引导模式的关系

MD_BOOT2	MD_BOOT1	MD_BOOT0	引导模式
*	0	0	引导模式 0 此模式从连接 CS0 空间的存储器（总线宽度为 16bit）引导。
*	1	0	引导模式 1 此模式从连接 CS0 空间的存储器（总线宽度为 32bit）引导。
0	0	1	引导模式 2 此模式从连接 NAND 闪存控制器的 NAND 闪存引导。
1	0	1	引导模式 3 此模式从连接瑞萨串行外围接口的通道 0（PB20 ~ 17）的串行闪存引导。不能从瑞萨串行外围接口的通道 0（PJ19 ~ 16）引导。
0	1	1	引导模式 4 此模式从连接 SD 主机接口的通道 0 的 SD 控制器内部闪存引导。
1	1	1	引导模式 5 此模式从连接 MMC 主机接口的 MMC 控制器内部闪存引导。

4.3 运行说明

4.3.1 引导模式 0 和引导模式 1

引导模式 0 和引导模式 1 从连接 CS0 空间的存储器引导。在这些模式中，本 LSI 的运行如下。

在解除上电复位后，从连接 CS0 空间的存储器中的异常处理向量表取出程序计数器（PC）的初始值（执行的起始地址）和堆栈指针（SP）的初始值，开始执行程序。

4.3.2 引导模式 2

引导模式 2 从连接 NAND 闪存控制器的 NAND 闪存引导。对应的 NAND 闪存是以大容量块（2048+64）为单位的 5 字节地址（不小于 2Gbit 的容量）产品。引导模式 2 的启动流程如下。

(1) 引导启动内部 ROM 程序的执行

在解除上电复位后，CPU 执行置于引导启动内部 ROM（非公开）的程序。

(2) 装入程序的传送

将 8KB 装入程序从连接 NAND 闪存控制器的 NAND 闪存传送到高速内部 RAM 的起始地址。装入程序的传送和检查如下：

(a) 检索装入程序的保存块（BlockAddress0 ~ 1023max）。

(b) 读 8KB（16 个扇区）装入程序，传送到高速内部 RAM。

在装入程序传送结束后，为了开始执行已传送的装入程序，CPU 跳转到高速内部 RAM。

(3) 应用程序的传送（任意）

在装入程序中，使用 NAND 闪存控制器将要展开的数据从 NAND 闪存装入到内部 RAM 或者外部 RAM。

引导模式 2 的规格概略图如图 4.1 所示。

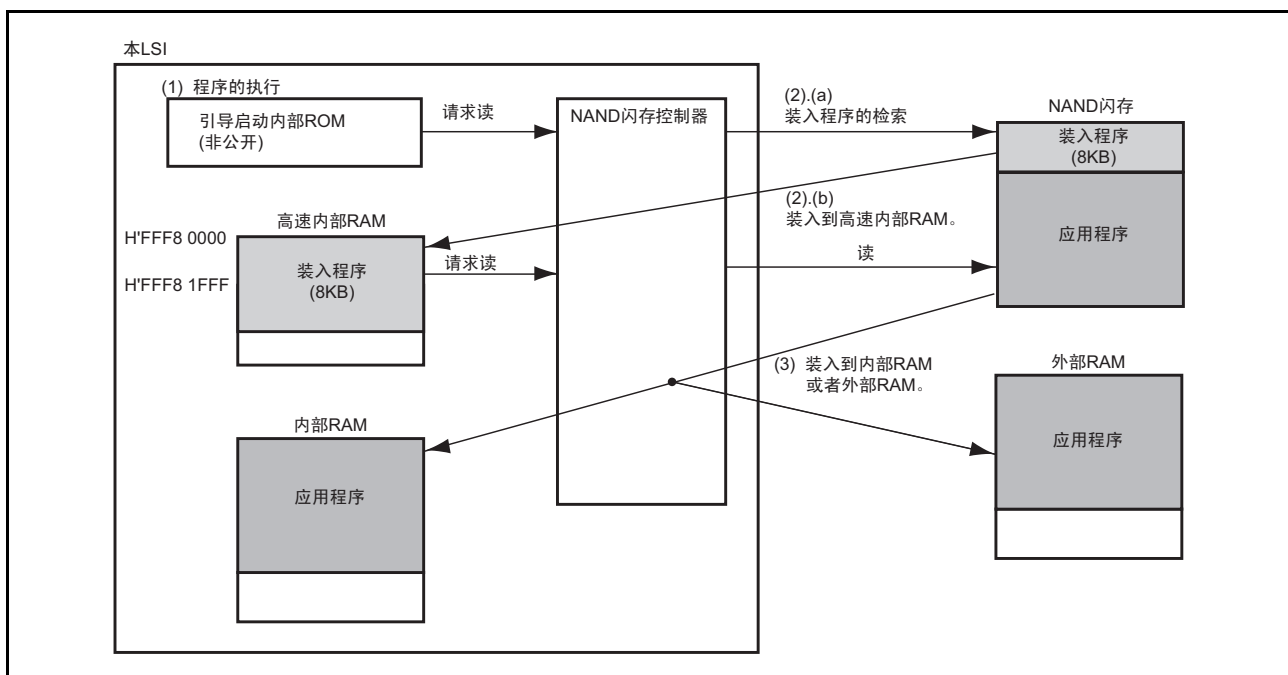


图 4.1 引导模式 2 的规格概略图

装入程序的保存位置如图 4.2 所示。

必须将装入程序保存到装入程序块的扇区 0 ~ 15，以扇区为单位读装入程序。

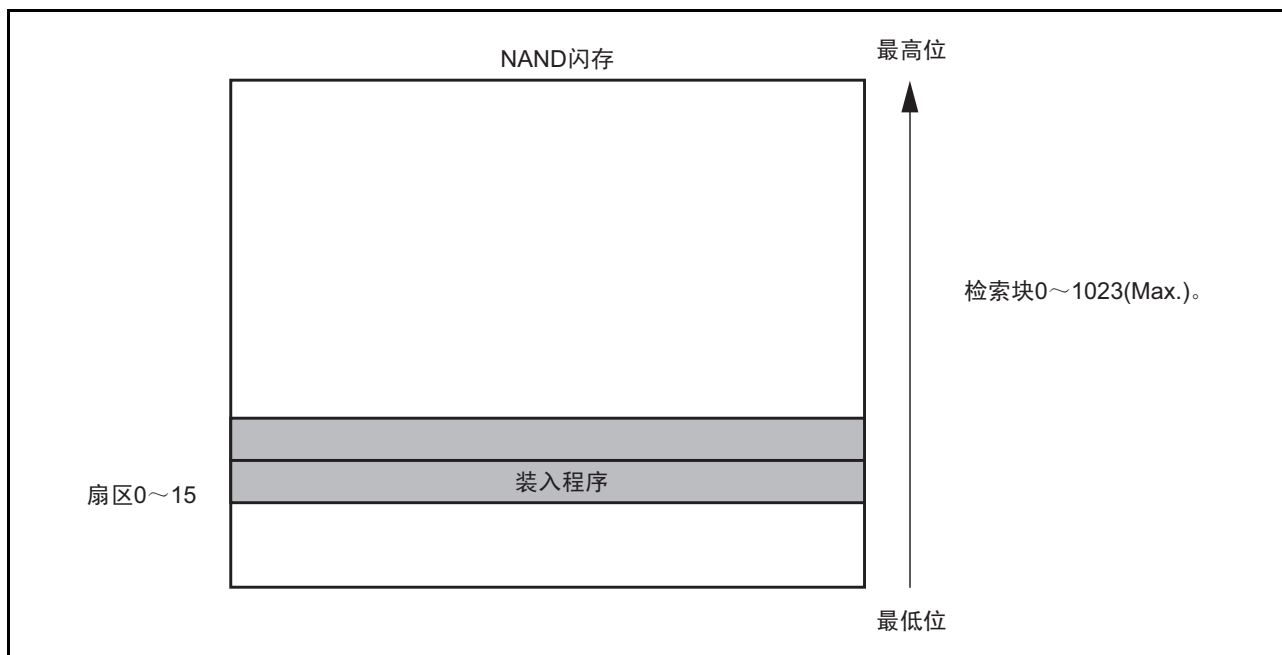


图 4.2 装入程序的保存位置

4.3.3 引导模式 3

引导模式 3 从连接瑞萨串行外围接口的通道 0 的串行闪存引导，其启动流程如下。

(1) 引导启动内部 ROM 程序的执行

在解除上电复位后，CPU 执行置于引导启动内部 ROM（非公开）的程序。

(2) 装入程序的传送

将 8KB 装入程序从连接瑞萨串行外围接口通道 0 的串行闪存的起始地址传送到高速内部 RAM 的起始地址。在装入程序传送结束后，为了开始执行已传送的装入程序，CPU 跳转到高速内部 RAM。

(3) 应用程序的传送（任意）

在装入程序中，使用瑞萨串行外围接口将要展开的数据从串行闪存装入到内部 RAM 或者外部 RAM。引导模式 3 的规格概略图如图 4.3 所示。

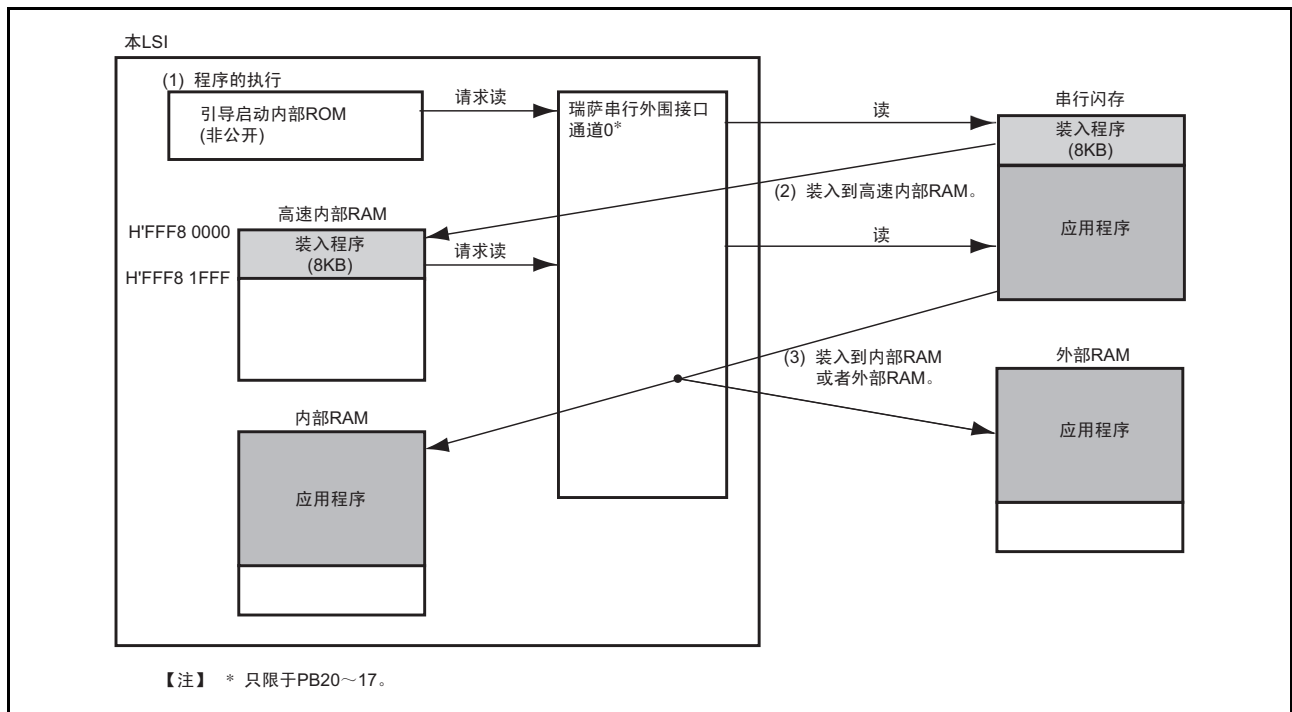


图 4.3 引导模式 3 的规格概略图

4.3.4 引导模式 4

引导模式 4 从连接 SD 主机接口的通道 0 的 SD 控制器内部闪存引导。引导模式 4 的启动流程如下。

(1) 引导启动内部 ROM 程序的执行

在解除上电复位后，CPU 执行置于引导启动内部 ROM（非公开）的程序。

(2) 装入程序的传送

将 16KB 装入程序从连接 SD 主机接口的通道 0 的 SD 控制器内部闪存传送到高速内部 RAM 的起始地址（页面 0）。另外，作为引导处理的工作存储器，也使用高速内部 RAM 的页面 1。

在装入程序传送结束后，为了开始执行已传送的装入程序，CPU 跳转到高速内部 RAM 的页面 0。

(3) 应用程序的传送（任意）

在装入程序中，使用 SD 主机接口要展开的将数据从 SD 控制器内部闪存装入到内部 RAM 或者外部 RAM。引导模式 4 的规格概略图如图 4.4 所示。

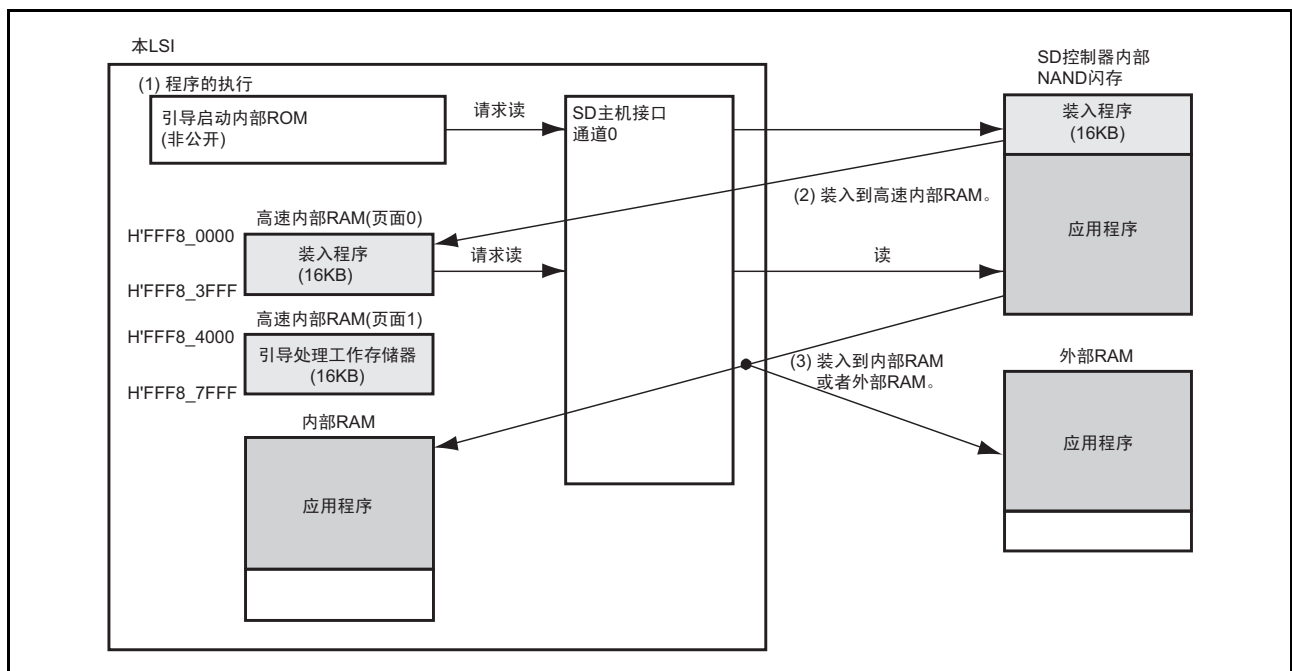


图 4.4 引导模块 4 的规格概略图

4.3.5 引导模式 5

引导模式 5 从连接 MMC 主机接口的 MMC 控制器内部闪存引导，其启动流程如下。

(1) 引导启动内部 ROM 程序的执行

在解除上电复位后，CPU 执行置于引导启动内部 ROM（非公开）的程序。

(2) 装入程序的传送

通过 4 位 MMC 数据总线宽度将 16KB 装入程序从连接 MMC 主机接口的 MMC 控制器内部闪存传送到高速内部 RAM 的起始地址（页面 0）。另外，作为引导处理的工作存储器，也使用高速内部 RAM 的页面 1。

在装入程序传送结束后，为了开始执行已传送的装入程序，CPU 跳转到高速内部 RAM 的页面 0。

(3) 应用程序的传送（任意）

在装入程序中，使用 MMC 主机接口将要展开的数据从 MMC 控制器内部闪存装入到内部 RAM 或者外部 RAM。

引导模式 5 的规格概略图如图 4.5 所示。

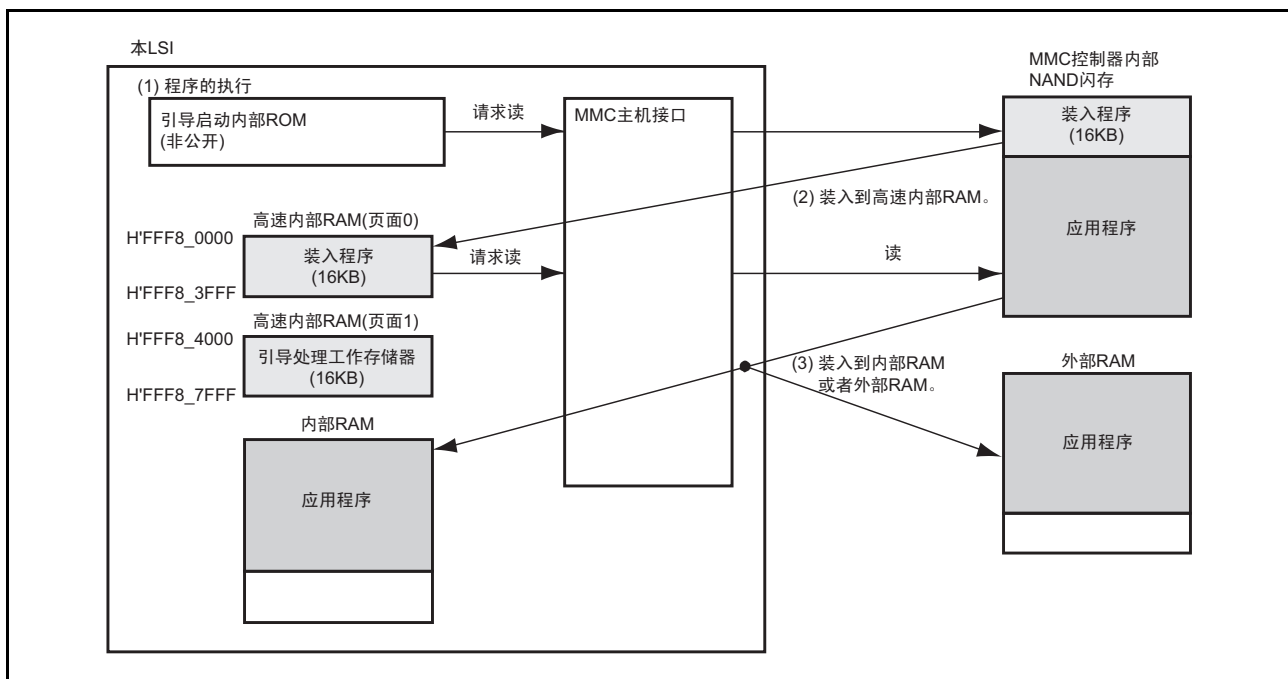


图 4.5 引导模块 5 的规格概略图

4.4 注意事项

4.4.1 引导相关引脚

读 CS0 空间的存储器、NAND 闪存控制器、瑞萨串行外围接口的通道 0、SD 主机接口的通道 0、MMC 主机接口的相关引脚初始状态以及深度待机时的引脚输出状态因引导模式而不同。

有关详细内容，请分别参考“10. 总线状态控制器”、“48. 通用输入 / 输出端口”、“49. 低功耗模式”的章节。

5. 时钟脉冲振荡器

本 LSI 内置时钟脉冲振荡器，生成 CPU 时钟 ($I\phi$)、内部总线时钟 ($B\phi$)、外围时钟 0 ($P0\phi$) 和外围时钟 1 ($P1\phi$)。时钟脉冲振荡器由晶体振荡器、PLL 电路和分频电路构成。

5.1 特点

- 4种时钟
能独立生成用于 CPU 和高速缓存的 CPU 时钟 ($I\phi$)、用于 I-Bus 的内部总线时钟 ($B\phi$)、用于外围模块的外围时钟 0 ($P0\phi$) 以及用于外部总线接口的外围时钟 1 ($P1\phi=CKIO$)。
- 频率变更功能
能通过此模块内部的 PLL (Phase Locked Loop) 电路和分频电路，独立更改 CPU 时钟和内部总线时钟的频率。通过软件设定频率控制寄存器 (FRQCR) 来更改频率。
- 低功耗模式的控制
能通过睡眠模式、软件待机模式和深度待机模式停止时钟的振荡，并且能通过模块待机功能停止特定模块的运行。有关低功耗模式的控制，请参照“49. 低功耗模式”。
- SSCG 功能
在 CPU 内部的 PLL (Phase Locked Loop) 电路中内置了 SSCG (Spread Spectrum Clock Generator)。SSCG 能通过稍微改变输出频率并且使其振荡 (调制频率)，抑制 EMI (电磁干扰) 噪声的峰值。
本 LSI 使用的 SSCG 规格如下：
 - SSCG 规格
 1. 调制波形 (调制分布): 三角波
 2. 扩频类型: 向下扩频
 3. 调制率: -2.5% (固定)
 4. 调制频率: 20.00 ~ 26.67kHz (EXTAL 频率 +500)

框图如图 5.1 所示。

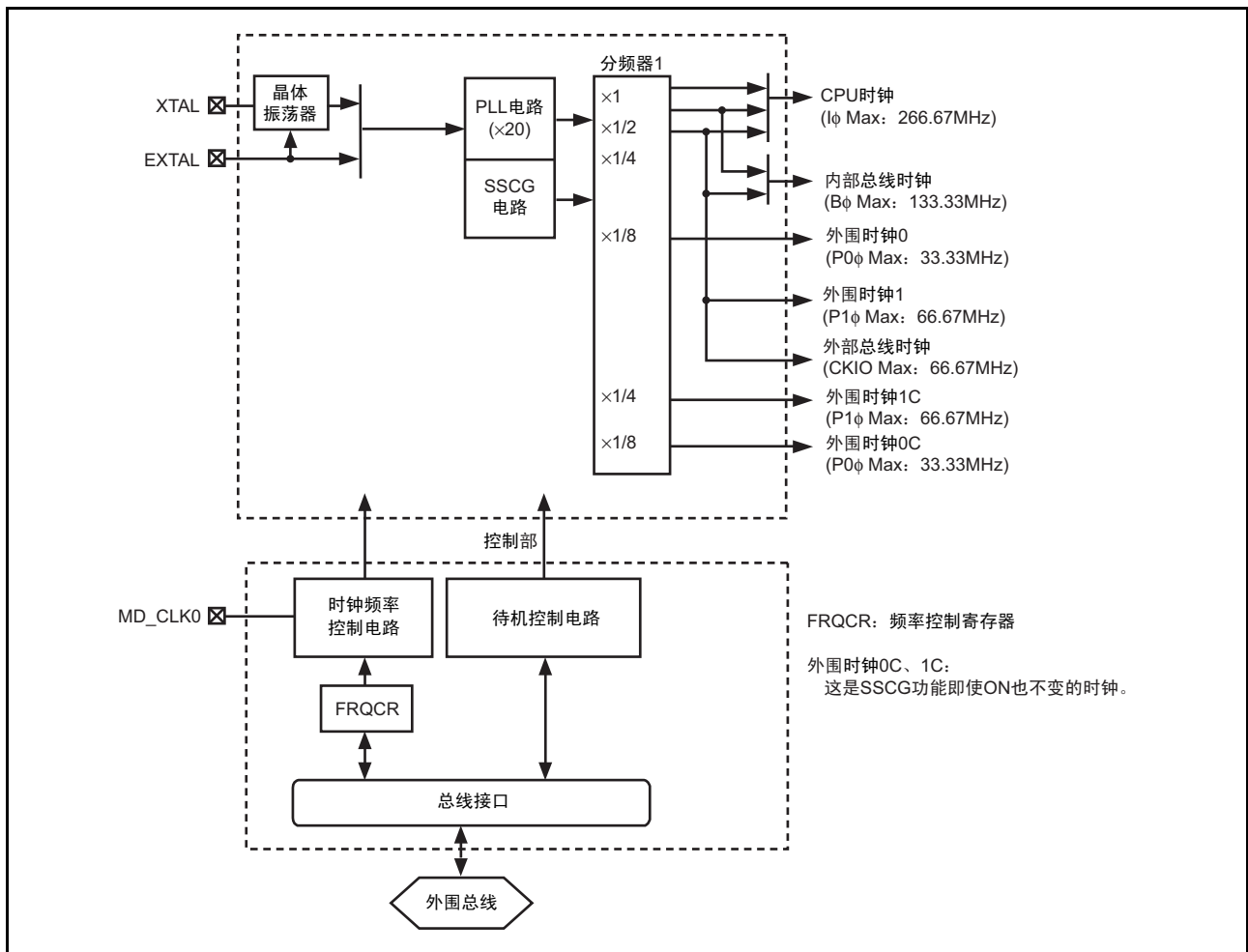


图 5.1 框图

此模块的各块功能如下。

(1) 晶体振荡器

当将晶体谐振器连接到 XTAL 引脚和 EXTAL 引脚时，使用晶体振荡器。

(2) PLL 电路

PLL 电路有将晶体振荡器或者 EXTAL 引脚输入的时钟频率递增 20 倍的功能。

(3) 分频器 1

分频器 1 生成 CPU 时钟、内部总线时钟、外围时钟 0 和外围时钟 1 使用的运行频率时钟。通过频率控制寄存器设定 CPU 时钟和内部总线时钟的分频比。外围时钟 0 的分频比固定为 1/4，外围时钟 1 的分频比固定为 1/8。

(4) 时钟频率的控制电路

时钟频率控制电路通过频率控制寄存器（FRQCR）来控制时钟频率。

(5) 待机控制电路

在时钟切换时或者在睡眠模式、软件待机模式、深度待机模式中，待机控制电路控制内部振荡电路的状态以及其他模块的状态。

另外，还有控制其他模块的低功耗模式的待机控制寄存器。有关待机控制寄存器，请参照“49. 低功耗模式”。

(6) 频率控制寄存器（FRQCR）

频率控制寄存器（FRQCR）中分配了在软件待机模式中是否从 CKIO 引脚输出时钟的控制位以及 CPU 时钟（I ϕ ）和内部总线时钟（B ϕ ）的频率分频比控制位。

(7) SSCG 电路

SSCG 电路通过 MD_CLK0 引脚进行功能 ON 和功能 OFF 的控制。当 SSCG 功能为 OFF 时，全部的内部时钟都不变而为固定频率的时钟；当 SSCG 功能为 ON 时，除了以下所示的提供给部分外围模块的时钟以外，内部时钟为变动的频率时钟。

需要提供固定频率时钟的外围模块：

IEBus 控制器、多功能定时器脉冲单元 2、FIFO 内置型串行通信接口、控制器局域网、比较匹配定时器、马达控制 PWM 定时器、音频发生器

5.2 输入 / 输出引脚

引脚结构和功能如表 5.1 所示。

表 5.1 引脚结构与功能

名称	引脚名	输入 / 输出	功能
模式控制引脚	MD_CLK0	输入	设定 SSCG 电路运行的 ON 或者 OFF。
晶体输入 / 输出引脚 (时钟输入引脚)	XTAL	输出	连接晶体谐振器。 (当不使用晶体谐振器时，必须将引脚置为开路。)
	EXTAL	输入	连接晶体谐振器或者用作外部时钟的输入引脚。
时钟输出引脚	CKIO	输出	为时钟输出引脚。

5.3 时钟模式

时钟输入 / 输出的关系和能使用的频率范围分别如表 5.2 和表 5.3 所示。

表 5.2 时钟输入 / 输出的关系

时钟输入 / 输出		PLL 电路	CKIO 的频率
供给源	输出		
EXTAL/ 晶体谐振器	CKIO	ON (×20)	(EXTAL/ 晶体) ×5

从 EXTAL 引脚或者晶体振荡器输入时钟，通过 PLL 电路形成波形以及倍增频率，提供给本 LSI。EXTAL 引脚的输入和晶体谐振器都能使用 10MHz ~ 13.333MHz 的振荡频率，CKIO 的频率范围是 50MHz ~ 66.67MHz。

表 5.3 能设定的频率范围

FRQCR 寄存器 设定值 *1	PLL 倍频率 PLL 电路	内部时钟比 (I:B:P1:P0) *2	能设定的频率范围 (MHz)					
			输入时钟 *3	输出时钟 (CKIO 引脚)	CPU 时钟 (I ϕ)	内部总线时钟 (B ϕ)	外围时钟 1 (P1 ϕ)	外围时钟 0 (P0 ϕ)
H'x015	ON (×20)	20:10:5:5/2	10 ~ 13.333	50 ~ 66.67	200 ~ 266.66	100 ~ 133.33	50 ~ 66.67	25 ~ 33.33
H'x035	ON (×20)	20:5:5:5/2	10 ~ 13.333	50 ~ 66.67	200 ~ 266.66	50 ~ 66.67	50 ~ 66.67	25 ~ 33.33
H'x115	ON (×20)	10:10:5:5/2	10 ~ 13.333	50 ~ 66.67	100 ~ 133.33	100 ~ 133.33	50 ~ 66.67	25 ~ 33.33
H'x135	ON (×20)	10:5:5:5/2	10 ~ 13.333	50 ~ 66.67	100 ~ 133.33	50 ~ 66.67	50 ~ 66.67	25 ~ 33.33
H'x335	ON (×20)	5:5:5:5/2	10 ~ 13.333	50 ~ 66.67	50 ~ 66.67	50 ~ 66.67	50 ~ 66.67	25 ~ 33.33

【注】 *1 FRQCR 寄存器的设定值 x 取决于 bit14、12、13 的设定值。

*2 这是输入时钟频率为 1 时的时钟比。

*3 这是 EXTAL 引脚的时钟输入或者晶体谐振器的频率。

【注意事项】

在使用本 LSI 时不能设定表 5.3 以外的频率。

在 $\overline{\text{RES}}$ 为 L 电平的条件下，本 LSI 通过 MD_CLK0 引脚的设定值切换 SSCG 运行的 ON 和 OFF。设定值对应的运行如下所示。但是，PLL 的倍频率和各时钟的分频比不会因设定值而变。

表 5.4 SSCG 运行设定

MD_CLK0 引脚的设定值	SSCG 运行
0	OFF
1	ON

5.4 寄存器说明

寄存器结构如表 5.5 所示。

表 5.5 寄存器结构

寄存器名	略称	R/W	初始值	地址	存取长度
频率控制寄存器	FRQCR	R/W	H'0335	H'FFFE0010	16

5.4.1 频率控制寄存器 (FRQCR)

FRQCR 是 16 位可读写寄存器，能在正常运行、释放总线权、更改 XTAL 晶体振荡器的增益、软件待机模式以及解除待机模式时指定是否从 CKIO 引脚输出时钟以及 CPU 时钟 (I ϕ) 和内部总线时钟 (P ϕ) 的分频比。只能对 FRQCR 进行字存取。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	CKOEN2	CKOEN[1:0]	—	—	IFC[1:0]	—	—	—	—	BFC[1:0]	—	—	—	—	—
初始值:	0	0	0	0	0	0	1	1	0	0	1	1	0	1	0	1
R/W:	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R	R

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14	CKOEN2	0	R/W	时钟输出允许 2 在更改 XTAL 晶体振荡器的增益时，CKOEN2 指定是从 CKIO 引脚输出时钟还是将 CKIO 引脚固定为低电平。 当设定为“1”时，在更改 XTAL 晶体振荡器的增益期间，将 CKIO 引脚固定为低电平。因此，能在更改 XTAL 晶体振荡器的增益期间防止因不稳定的 CKIO 时钟而导致的外部电路误动作。 0: 输出不稳定的时钟 1: 输出低电平
13、12	CKOEN[1:0]	00	R/W	时钟输出允许 在正常运行、释放总线权、深度待机模式、待机模式以及解除待机模式时，CKOEN[1:0] 指定是从 CKIO 引脚输出时钟还是固定 CKIO 引脚的电平或者将 CKIO 引脚置为高阻抗状态 (Hi-Z)。 当设定为“01”时，在深度待机模式、软件待机模式以及解除软件待机模式的期间，将 CKIO 引脚固定为低电平。因此，能在解除软件待机模式期间，防止因不稳定的 CKIO 时钟而导致的外部电路误动作。 CKOEN[1:0] 位的设定内容如表 5.6 所示。
11、10	—	全 0	R	保留位 读写值都为“0”。
9、8	IFC[1:0]	11	R/W	CPU 时钟频率的分频比 指定对 PLL 电路输出频率的 CPU 时钟频率的分频比。 00: $\times 1$ 倍 01: $\times 1/2$ 倍 10: 保留 (禁止设定) 11: $\times 1/4$ 倍

位	位名	初始值	R/W	说明
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	BFC[1:0]	11	R/W	内部总线时钟频率的分频比 指定对 PLL 电路输出频率的内部总线时钟频率的分频比。 00: 保留（禁止设定） 01: ×1/2 倍 10: 保留（禁止设定） 11: ×1/4 倍
3	—	0	R	保留位 读写值都为“0”。
2	—	1	R	保留位 读写值都为“1”。
1	—	0	R	保留位 读写值都为“0”。
0	—	1	R	保留位 读写值都为“1”。

表 5.6 CKOEN[1:0] 位的设定内容

设定值	正常运行	释放总线权	软件待机模式	深度待机模式 *
00	输出	输出截至 (Hi-Z)	输出截至 (Hi-Z)	输出截至 (Hi-Z)
01	输出	输出	输出低电平	输出低电平
10	输出	输出	输出 (输出不稳定的时钟)	输出低电平或者高电平
11	输出截至 (Hi-Z)	输出截至 (Hi-Z)	输出截至 (Hi-Z)	输出截至 (Hi-Z)

【注】 * 但是，在解除深度待机模式时，输出的 CKIO 时钟的开始部分可能有欠缺。

5.5 频率的更改方法

要更改 CPU 时钟 ($I\phi$) 和内部总线时钟 ($B\phi$) 的频率时，改变分频器的分频比。通过软件设定频率控制寄存器 (FRQCR) 来控制分频比的更改。

5.5.1 分频比的更改

更改分频比的步骤如下：

1. 在初始状态下，IFC[1:0]为“B'11”，BFC[1:0]为“B'11”。
2. 给IFC[1:0]和BFC[1:0]设定目标值。必须注意：如果设定错误值，就会导致本LSI的误动作。
3. 如果设定寄存器的各位（IFC[1:0]和BFC[1:0]），就切换到所设定的时钟。

【注】 要在更改频率后执行 SLEEP 指令时，必须在读 3 次频率控制寄存器 (FRQCR) 后执行 SLEEP 指令。

5.6 时钟引脚的使用方法

作为能连接晶体谐振器或者时钟输入的引脚，本 LSI 有表 5.7 所示的引脚。

对于这些引脚，必须注意以下事项。本文中的 X_{in} 引脚和 X_{out} 引脚表示表 5.7 的引脚。

表 5.7 时钟引脚

X _{in} 引脚（用作连接晶体谐振器或者时钟输入的引脚）	X _{out} 引脚（连接晶体谐振器）
EXTAL	XTAL
USB_X1	USB_X2
AUDIO_X1	AUDIO_X2
RTC_X1	RTC_X2
VIDEO_X1	VIDEO_X2

5.6.1 输入外部时钟的情况

输入外部时钟的连接例子如图 5.2 所示。当将 X_{out} 引脚置为开路状态时，寄生电容不能超过 10pF。

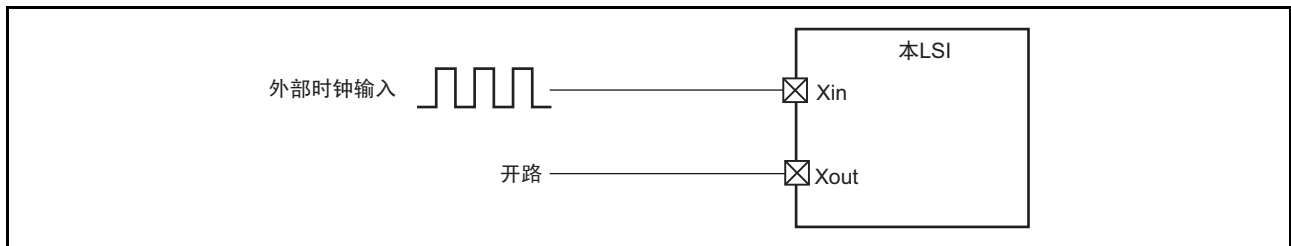


图 5.2 外部时钟的连接例子

5.6.2 使用晶体谐振器的情况

晶体谐振器的连接例子如图 5.3 所示。

尽量将晶体谐振器和电容 CL1、CL2 设置在 X_{in} 引脚和 X_{out} 引脚的附近。为了避免感应而进行正常的振荡，附加在晶体谐振器两侧的电容必须连接同一个接地，并且不能在这些器件的附近设置布线图形。

因为有关晶体谐振器的各种特性与用户的电路板设计密切相关，所以请用户参照本章介绍的晶体谐振器的连接例子，在给予充分的评估后使用。晶体谐振器的电路额定值因晶体谐振器和安装电路的寄生电容而不同，因此必须和谐振器厂商仔细商谈后决定。时钟引脚的外加电压不能超过最大额定值。尽管本 LSI 内置了反馈电阻，但是根据晶体谐振器的特性，可能还需要外接反馈电阻，请用户在给予充分的评估后设定参数（电阻值和电容值）。

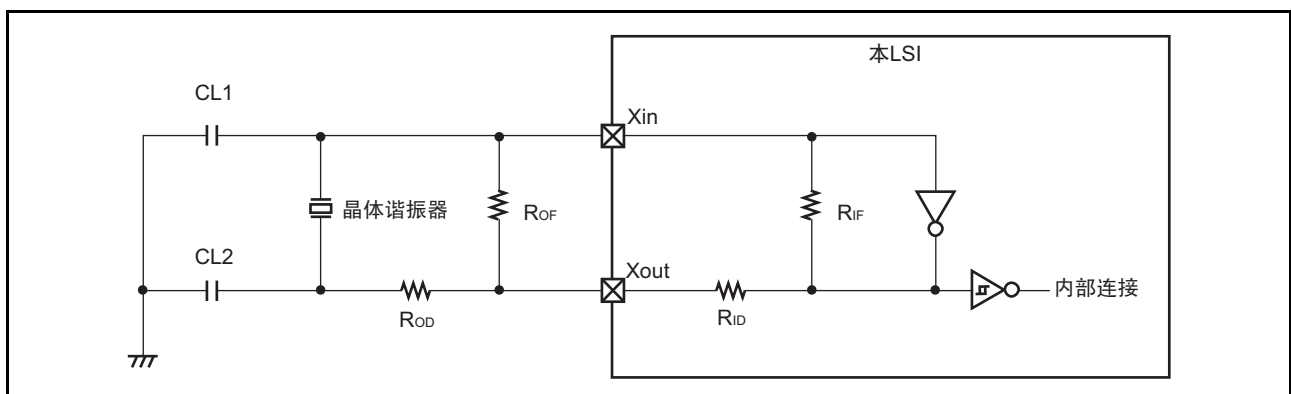


图 5.3 晶体谐振器的连接例子

5.6.3 不使用的情况

不使用时，必须固定 X_{in} 引脚的电平（上拉 / 下拉 / 连接电源 / 接地），并且将 X_{out} 引脚置为开路。

5.7 振荡稳定时间

5.7.1 内部晶体振荡器的振荡稳定时间

在使用晶体谐振器时，为了确保内部晶体振荡器的振荡稳定时间，必须在以下情况下至少等待振荡稳定时间（在输入外部时钟时不需要）：

- 在接通电源时
- 在用 \overline{RES} 引脚解除软件待机模式或者深度待机模式时
- 在通过上电复位或者设定寄存器使振荡从停止状态变为振荡状态时（ $AUDIO_X1$ 和 RTC_X1 ）
- 在通过 \overline{RES} 引脚的上电复位来改变晶体振荡器的增益时（ $EXTAL$ ）

5.7.2 PLL 电路的振荡稳定时间

因为将 $EXTAL$ 输入的信号提供给 PLL，所以对于 $EXTAL$ 无论是连接晶体谐振器还是输入外部时钟，为了确保 PLL 电路的振荡稳定时间，必须在以下情况下至少等待振荡稳定时间：

- 在接通电源（在使用晶体谐振器时）/ 开始输入外部时钟（在输入外部时钟时）时
- 在用 \overline{RES} 引脚解除软件待机模式或者深度待机模式时

【备注】

在以下情况下，通过本 LSI 的内部计数器确保振荡稳定时间：

- 在用 \overline{RES} 引脚解除软件待机模式或者深度待机模式时
- 在通过设定寄存器来改变晶体振荡器的增益时（ $EXTAL$ ）

5.8 设计电路板时的注意事项

5.8.1 使用 PLL 振荡电路时的注意事项

对于 PLL 的 $PLLV_{CC}$ 的连接图形，必须缩短来自电路板电源提供引脚的布线长度，并且加宽图形宽度，减少感应成分。

由于 PLL 的模拟电源对噪声很敏感，因此可能会因其他电源的干扰而导致整个系统的误动作。所以，在电路板上绝对不能给此模拟电源和数字电源 V_{CC} 、 PV_{CC} 提供同一电源。

5.9 SSCG 规格的调制率和调制频率的定义

SSCG 电路能通过稍微改变输出的频率进行频率的调制，抑制辐射噪声的峰值。将此时的频率变化率定义为调制率，将输入时钟的频率变化的频率定义为调制频率。调制率和调制频率如图 5.4 所示。

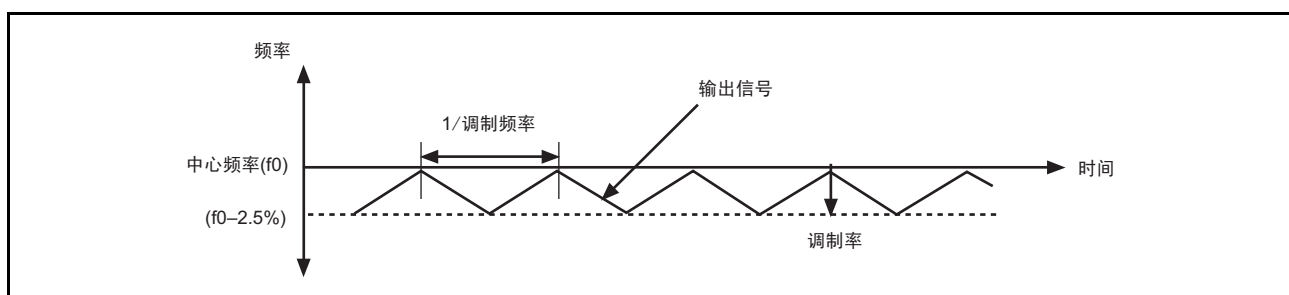


图 5.4 SSCG 调制率和调制频率的定义

6. 异常处理

6.1 概要

6.1.1 异常处理的种类和优先级

如表 6.1 所示，因复位、地址错误、寄存器组错误、中断和指令的各异常源而启动异常处理。异常源如表 6.1 所示，设有优先级，如果同时发生多个异常源，就按照此优先级接受处理。

表 6.1 异常源的种类和优先级

种类	异常处理	优先级
复位	上电复位	高   低
	手动复位	
地址错误	CPU 地址错误	
	DMA 地址错误	
指令	FPU 异常	
	整数除法运算异常（被零除）	
	整数除法运算异常（上溢）	
寄存器组错误	存储体下溢	
	存储体上溢	
中断	NMI	
	用户断点	
	用户调试接口	
	IRQ	
	PINT	
指令	陷阱指令（TRAPA 指令）	
	一般非法指令（未定义代码）	
	槽非法指令（分配在紧接着延迟转移指令 *1 之后的未定义代码（包括 FPU 模块待机时的 FPU 指令和 FPU 相关的 CPU 指令）、PC 改写指令 *2、32 位指令 *3、RESBANK 指令、DIVS 指令或者 DIVU 指令）	

【注】 *1 延迟转移指令：JMP、JSR、BRA、BSR、RTS、RTE、BF/S、BT/S、BSRF、BRAFL

*2 PC 改写指令：JMP、JSR、BRA、BSR、RTS、RTE、BT、BF、TRAPA、BF/S、BT/S、BSRF、BRAFL、JSR/N、RTV/N

*3 32 位指令：BAND.B、BANDNOT.B、BCLR.B、BLD.B、BLDNOT.B、BOR.B、BORNOT.B、BSET.B、BST.B、BXOR.B、MOV.B@disp12、MOV.W@disp12、MOV.L@disp12、MOVI20、MOVI20S、MOVU.B、MOVU.W

6.1.2 异常处理的运行

在如表 6.2 所示的时序中检测到各异常源后开始处理。

表 6.2 异常源检测和异常处理的开始时序

异常处理		异常源检测和异常处理的开始时序
复位	上电复位	在 RES 引脚从低电平变为高电平时或者在设置用户调试接口复位 Assert 命令后设置用户调试接口复位 Negate 命令时或者在看门狗定时器发生上溢时，开始处理。
	手动复位	在看门狗定时器发生上溢时，开始处理。
地址错误		在进行指令解码时检测到错误，在执行中的指令结束后开始处理。
中断		
寄存器组错误	存储体下溢	在没有保存到寄存器组的情况下执行 RESBANK 指令时，开始处理。
	存储体上溢	当设定为中断控制器接受寄存器组上溢异常（中断控制器的 IBNR 的 BOVE 位为“1”）时，如果发生使用寄存器组的中断并且 CPU 接受此中断，就在保存到寄存器组的全部区域后开始处理。
指令	陷阱指令	通过执行 TRAPA 指令开始处理。
	一般非法指令	当对不紧接在延迟转移指令（延迟槽）之后的未定义代码（包括 FPU 模块待机时的 FPU 指令和 FPU 相关的 CPU 指令）进行解码时，开始处理。
	槽非法指令	当对紧接在延迟转移指令（延迟槽）之后的未定义代码（包括 FPU 模块待机时的 FPU 指令和 FPU 相关的 CPU 指令）、PC 改写指令、32 位指令、RESBANK 指令、DIVS 指令或者 DIVU 指令进行解码时，开始处理。
	整数除法运算异常	当检测到被零除运算异常或者因用“-1”除负数的最大值(H'80000000)引起的上溢异常时，开始处理。
	FPU 异常	在发生浮点运算指令的无效运算异常（IEEE754 规定）、被零除运算异常、上溢、下溢或者不精确异常时，开始处理。另外，如果在 FPSCR 的 QIS 位被置位的状态下给浮点运算指令源输入 qNaN 或者 $\pm\infty$ ，就开始处理。

当启动异常处理时，CPU 的运行如下。

(1) 由复位引起的异常处理

从异常处理向量表中（PC 和 SP 在上电复位时分别为地址 H'00000000 和 H'00000004；在手动复位时分别为地址 H'00000008 和 H'0000000C）取程序计数器（PC）和堆栈指针（SP）的初始值。有关异常处理向量表，请参照“6.1.3 异常处理向量表”。然后，将向量基址寄存器（VBR）初始化为“H'00000000”，将状态寄存器（SR）的中断屏蔽级位（I3 ~ I0）初始化为“HF”（B'1111），将 BO 位和 CS 位初始化为“0”，并且将中断控制器（中断控制器）的 IBNR 的 BN 位初始化为“0”。而且在上电复位时还要将 FPSCR 初始化为“H'00040001”。从异常处理向量表中取到的 PC 地址开始执行程序。

(2) 由地址错误、寄存器组错误、中断和指令引起的异常处理

将 SR 和 PC 保存到 R15 指向的堆栈。在 NMI 和用户断点以外的中断异常处理中，如果设定了使用寄存器组，就将通用寄存器 R0 ~ R14、控制寄存器 GBR、系统寄存器 MACH、MACL、PR 以及执行中断异常处理的向量表地址偏移量保存到寄存器组，而在由地址错误、寄存器组错误、NMI 中断、用户断点中断、指令引起的异常处理时，不保存到寄存器组。另外，如果已保存到寄存器组的全部存储体，就替代寄存器组自动保存到堆栈。此时，需要事先将中断控制器设定为不接受寄存器组上溢异常（中断控制器的 IBNR 的 BOVE 位为“0”）。如果设定为接受寄存器组上溢异常（中断控制器的 IBNR 的 BOVE 位为“1”），就发生寄存器组上溢异常。在中断异常处理时，将中断优先级写到 SR 的 I3 ~ I0 位。在由地址错误或者指令引起的异常处理时，I3 ~ I0 位不受影响。然后，从异常处理向量表中取到异常服务程序的起始地址，从该地址开始执行程序。

6.1.3 异常处理向量表

执行异常处理前，需要预先将异常处理向量表设定到存储器，并且将异常服务程序的起始地址保存到异常处理向量表中（将 PC 和 SP 的初始值保存到复位异常处理表中）。

分别给各异常源分配了不同的向量号和向量表地址偏移量。从对应的向量号和向量表地址偏移量计算向量表地址。在异常处理中，从此向量表地址指向的异常处理向量表中取异常服务程序的起始地址。

向量号和向量表地址偏移量、向量表地址的计算方法分别如表 6.3 和表 6.4 所示。

表 6.3 异常处理向量表

异常源		向量号	向量表地址偏移量
上电复位	PC	0	H'00000000 ~ H'00000003
	SP	1	H'00000004 ~ H'00000007
手动复位	PC	2	H'00000008 ~ H'0000000B
	SP	3	H'0000000C ~ H'0000000F
一般非法指令		4	H'00000010 ~ H'00000013
(系统保留)		5	H'00000014 ~ H'00000017
槽非法指令		6	H'00000018 ~ H'0000001B
(系统保留)		7	H'0000001C ~ H'0000001F
		8	H'00000020 ~ H'00000023
CPU 地址错误		9	H'00000024 ~ H'00000027
DMA 地址错误		10	H'00000028 ~ H'0000002B
中断	NMI	11	H'0000002C ~ H'0000002F
	用户断点	12	H'00000030 ~ H'00000033
FPU 异常		13	H'00000034 ~ H'00000037
用户调试接口		14	H'00000038 ~ H'0000003B
存储体上溢		15	H'0000003C ~ H'0000003F
存储体下溢		16	H'00000040 ~ H'00000043
整数除法运算异常 (被零除)		17	H'00000044 ~ H'00000047
整数除法运算异常 (上溢)		18	H'00000048 ~ H'0000004B
(系统保留)		19	H'0000004C ~ H'0000004F
		31	H'0000007C ~ H'0000007F
陷阱指令 (用户向量)		32	H'00000080 ~ H'00000083
		63	H'000000FC ~ H'000000FF
外部中断 (IRQ、PINT)、内部外围模块 *		64	H'00000100 ~ H'00000103
		511	H'000007FC ~ H'000007FF

【注】 * 有关外部中断、各内部外围模块的中断向量号和向量表地址偏移量，请参照“7. 中断控制器”的表 7.4。

表 6.4 异常处理向量表地址的计算方法

异常源	向量表地址的计算方法
复位	向量表地址 = (向量表地址偏移量) = (向量号) × 4
地址错误、寄存器组错误、中断、指令	向量表地址 = VBR + (向量表地址偏移量) = VBR + (向量号) × 4

- 【注】 1. 向量表地址偏移量：参照表 6.3。
2. 向量号：参照表 6.3。

6.2 复位

6.2.1 输入 / 输出引脚

引脚结构如表 6.5 所示。

表 6.5 引脚结构

名称	引脚名	输入 / 输出	功能
上电复位	RES	输入	通过给引脚输入低电平，转移到上电复位处理。

6.2.2 复位的种类

复位是优先级最高的异常处理源，有上电复位和手动复位共 2 种。如表 6.6 所示，在上电复位或者手动复位时，对 CPU 状态进行初始化。在上电复位时，对 FPU 状态进行初始化，而在手动复位时不进行初始化。另外，在上电复位时，除了部分寄存器以外，对内部外围模块的寄存器进行初始化，而在手动复位时不进行初始化。

表 6.6 复位状态

种类	转移到复位状态的条件			内部状态				
	RES	用户调试接口命令	看门狗定时器的上溢	CPU	CPU 以外的模块	高速内部 RAM	大容量内部 RAM (用于保持的内部 RAM 除外)	用于保持的内部 RAM
上电复位	低电平	—	—	初始化	初始化	初始化 or 保持 *2	初始化 or 保持 *3	初始化 or 保持 *4、*5
	高电平	设置用户调试接口复位 Assert 命令	—	初始化	初始化	初始化 or 保持 *2	初始化 or 保持 *3	初始化 or 保持 *4
	高电平	设置用户调试接口复位 Assert 以外的命令	上电	初始化	*1	初始化 or 保持 *2	初始化 or 保持 *3	初始化 or 保持 *4
手动复位	高电平	设置用户调试接口复位 Assert 以外的命令	手动	初始化	*1	保持	保持	保持

- 【注】 *1 请参照“51.3 各运行模式中的寄存器状态一览表”。
*2 通过将 RAME 位或者 RAMWE 位置为无效，继续保持数据。
*3 通过将 VRAME 位或者 VRAMWE 位置为无效，继续保持数据。
*4 通过将 VRAME 位、VRAMWE 位或者 RRAMWE 位置为无效，继续保持数据。
*5 如果通过上电复位解除深度待机模式，就无法保持数据。

6.2.3 上电复位

(1) 由 $\overline{\text{RES}}$ 引脚引起的上电复位

如果将 $\overline{\text{RES}}$ 引脚置为低电平，本 LSI 就进入上电复位状态。为了确保本 LSI 的复位， $\overline{\text{RES}}$ 引脚的低电平在接通电源或者软件待机时（在时钟停止时），必须至少保持振荡稳定时间；在时钟运行时，必须至少保持 20tcyc。在上电复位的状态下，对 CPU 的内部状态和内部外围模块的寄存器全部进行初始化。有关上电复位状态下的各引脚状态，请参照“53.1 引脚状态”。

在上电复位的状态下，如果将 $\overline{\text{RES}}$ 引脚在保持一定期间的低电平后变为高电平，就开始上电复位的异常处理。此时 CPU 的运行如下：

1. 从异常处理向量表中取程序计数器（PC）的初始值（执行起始地址）。
2. 从异常处理向量表中取堆栈指针（SP）的初始值。
3. 将向量基址寄存器（VBR）清除为“H'00000000”，将状态寄存器（SR）的中断屏蔽级位（I3～I0）初始化为“HF”（B'1111），将BO位和CS位初始化为“0”，并且将中断控制器的IBNR的BN位初始化为“0”，将FPSCR初始化为“H'00040001”。
4. 分别给PC和SP设定从异常处理向量表中取到的值，然后开始执行程序。

另外，必须在接通系统电源时进行上电复位处理。

(2) 由用户调试接口复位 Assert 命令引起的上电复位

如果设置用户调试接口复位 Assert 命令，就进入上电复位状态。用户调试接口复位 Assert 命令和由 $\overline{\text{RES}}$ 引脚引起的上电复位同等。能通过设置用户调试接口复位 Negate 命令来解除上电复位状态。用户调试接口复位 Assert 命令和用户调试接口复位 Negate 命令之间所需的时间，与为了开始上电复位而保持 $\overline{\text{RES}}$ 引脚低电平的时间相同。在通过用户调试接口复位 Assert 命令进行上电复位的状态下，如果设置用户调试接口复位 Negate 命令，就开始上电复位异常处理。此时 CPU 的运行和由 $\overline{\text{RES}}$ 引脚引起上电复位时的运行相同。

(3) 由看门狗定时器引起的上电复位

如果设定为在看门狗定时器的看门狗定时器模式中发生上电复位并且看门狗定时器的 WTCNT 发生上溢，就进入上电复位状态。

此时，由看门狗定时器引起的复位信号不对看门狗定时器的 WRCSR 和时钟脉冲发生器的 FRQCR 进行初始化。

另外，如果同时发生由 $\overline{\text{RES}}$ 引脚、用户调试接口复位 Assert 命令引起的复位和由看门狗定时器上溢引起的复位，就优先进行由 $\overline{\text{RES}}$ 引脚、用户调试接口复位 Assert 命令引起的复位，并且将 WRCSR 的 WOVF 位清“0”。当开始由看门狗定时器引起的上电复位异常处理时，CPU 的运行和由 $\overline{\text{RES}}$ 引脚引起上电复位时的运行相同。

6.2.4 手动复位

(1) 由看门狗定时器引起的手动复位

如果设定为在看门狗定时器的看门狗定时器模式中产生手动复位，就在看门狗定时器的 WTCNT 发生上溢时进入手动复位状态。

在开始由看门狗定时器引起的手动复位异常处理时，CPU 的运行如下：

1. 从异常处理向量表中取程序计数器（PC）的初始值（执行起始地址）。
2. 从异常处理向量表中取堆栈指针（SP）的初始值。
3. 将向量基址寄存器（VBR）清除为“H'00000000”，将状态寄存器（SR）的中断屏蔽级位（I3～I0）初始化为“HF”（B'1111），将BO位和CS位初始化为“0”，并且将中断控制器的IBNR的BN位初始化为“0”。
4. 分别给PC和SP设定从异常处理向量表中取到的值，然后开始执行程序。

(2) 手动复位时的注意事项

在发生手动复位时，保持总线周期。如果在释放总线权或者直接存储器存取控制器的突发传送过程中发生手动复位，就将手动复位异常处理保留到 CPU 获得总线权为止。在手动复位时，对 CPU 和中断控制器的 IBNR 的 BN 位进行初始化，而不对 FPU 和其他模块进行初始化。

6.3 地址错误

6.3.1 地址错误发生源

如表 6.7 所示，在取指令或者读写数据时发生地址错误。

表 6.7 总线周期和地址错误

总线周期		总线周期的内容	地址错误的发生
种类	总线主控		
取指令	CPU	从偶数地址开始取指令。	无（正常）
		从奇数地址开始取指令。	发生地址错误。
		从内部外围模块空间*、内部 RAM 空间* 的 H'F0000000 ~ H'F5FFFFFFF 以外的空间取指令。	无（正常）
		从内置外围模块空间*、内置 RAM 空间* 的 H'F0000000 ~ H'F5FFFFFFF 取指令。	发生地址错误。
读写数据	CPU 或者 直接存储器 存取控制器	从偶数地址开始存取字数据。	无（正常）
		从奇数地址开始存取字数据。	发生地址错误。
		从长字边界开始存取长字数据。	无（正常）
		从非长字边界开始存取长字数据。	发生地址错误。
		从双长字边界开始存取双长字数据。	无（正常）
		从非双长字边界开始存取双长字数据。	发生地址错误。
		在内部外围模块空间* 存取字数据和字节数据。	无（正常）
		在 16 位内部外围模块空间* 存取长字数据。	无（正常）
在 8 位内部外围模块空间* 存取长字数据。	无（正常）		

【注】* 有关内部外围模块空间和内部 RAM 空间，请参照“10. 总线状态控制器”。

6.3.2 地址错误的异常处理

当发生地址错误时，结束发生地址错误的总线周期，并且在结束正在执行的指令后开始地址错误的异常处理。此时 CPU 的运行如下：

1. 从异常处理向量表中取对应发生地址错误的异常服务程序的起始地址。
2. 将状态寄存器（SR）压栈。
3. 将程序计数器（PC）压栈。被压栈的 PC 值是最后执行指令的下一个指令起始地址。
4. 转移到从异常处理向量表中取到的异常服务程序的起始地址，开始执行程序。此时的转移不是延迟转移。

6.4 寄存器组错误

6.4.1 寄存器组的错误发生源

(1) 存储体上溢

在将中断控制器设定为接受寄存器组上溢异常（中断控制器的 IBNR 的 BOVE 位为“1”）的情况下发生使用寄存器组的中断并且 CPU 接受此中断时，如果已保存到寄存器组的全部区域，就发生存储体上溢。

(2) 存储体下溢

如果在未保存到寄存器组的情况下执行 RESBANK 指令，就发生存储体下溢。

6.4.2 寄存器组的错误异常处理

当发生寄存器组错误时，开始寄存器组错误的异常处理。此时，CPU 的运行如下：

1. 从异常处理向量表中取对应发生寄存器组错误的异常服务程序的起始地址。
2. 将状态寄存器（SR）压栈。
3. 将程序计数器（PC）压栈。被压栈的 PC 值在存储体发生上溢时是最后执行指令的下一个指令起始地址；在存储体发生下溢时是执行的 RESBANK 指令起始地址。
为了防止在存储体上溢时发生多重中断，将存储体上溢源的中断级写到状态寄存器（SR）的中断屏蔽级位（I3～I0）。
4. 转移到从异常处理向量表中取到的异常服务程序的起始地址，开始执行程序。此时的转移不是延迟转移。

6.5 中断

6.5.1 中断源

启动异常处理的中断源有 NMI、用户断点、用户调试接口、IRQ、PINT 和内部外围模块。

分别给各异常源分配了不同的向量号和向量表地址偏移量。有关向量号和向量表地址偏移量，请参照“7. 中断控制器”的表 7.4。

6.5.2 中断优先级

中断源设有优先级，如果同时发生多个中断（多重中断），就通过中断控制器判断优先级，并且按照该判断结果启动异常处理。

用优先级 0 ~ 16 表示中断源的优先级，优先级 0 为最低，优先级 16 为最高。NMI 中断是优先级 16 并且不能屏蔽的最优先的中断，随时被接受。用户断点中断和用户调试接口的优先级为 15。能通过中断控制器的中断优先级设定寄存器 01、02、05 ~ 26（IPR01、IPR02、IPR05 ~ IPR26）自由设定 IRQ 中断、PINT 中断和内部外围模块中断的优先级（表 6.8）。能设定的优先级为 0 ~ 15，不能设定优先级 16。有关 IPR01、IPR02、IPR05 ~ IPR26，请参照“7.3.1 中断优先级设定寄存器 01、02、05 ~ 26（IPR01、IPR02、IPR05 ~ IPR26）”。

表 6.8 中断优先级

种类	优先级	备注
NMI	16	固定优先级，不能屏蔽。
用户断点	15	固定优先级
用户调试接口	15	固定优先级
IRQ	0 ~ 15	通过中断优先级设定寄存器 01、02、05 ~ 26（IPR01、IPR02、IPR05 ~ IPR26）来设定。
PINT		
内部外围模块		

6.5.3 中断异常处理

如果发生中断时，就通过中断控制器判断优先级。NMI 随时被接受，但是其他中断只在其优先级高于设定在状态寄存器（SR）的中断屏蔽级位（I3 ~ I0）的优先级时才被接受。

如果接受中断，就开始中断异常处理。在中断异常处理中，CPU 从异常处理向量表中取对应接受中断的异常服务程序的起始地址，并且将 SR 和程序计数器（PC）压栈。在 NMI 和用户断点以外的中断异常处理中，如果设定了使用寄存器组，就将通用寄存器 R0 ~ R14、控制寄存器 GBR、系统寄存器 MACH、MACL、PR 以及执行异常处理的向量表地址偏移量保存到寄存器组，而在由地址错误、NMI 中断、用户断点中断、指令引起的异常处理时，不保存到寄存器组。另外，如果已保存到寄存器组的全部存储体（0 ~ 14），就代替寄存器组自动保存到堆栈。此时，需要事先将中断控制器设定为不接受寄存器组上溢异常（中断控制器的 IBNR 的 BOVE 位为“0”）。如果设定为接受寄存器组上溢异常（中断控制器的 IBNR 的 BOVE 位为“1”），就发生寄存器组上溢异常。然后将接受中断的优先级值写到 SR 的 I3 ~ I0 位。但是，在 NMI 时优先级为 16，而给 I3 ~ I0 位设定的值为“HF”（优先级 15）。之后，转移到从异常处理向量表中取到的中断异常服务程序的起始地址，开始执行程序。此时的转移不是延迟转移。有关中断异常处理的详细内容，请参照“7.6 运行说明”。

6.6 由指令引起的异常

6.6.1 由指令引起的异常种类

如表 6.9 所示，启动异常处理的指令有陷阱指令、槽非法指令、一般非法指令、整数除法运算异常和 FPU 异常。

表 6.9 由指令引起的异常种类

种类	源指令	备注
陷阱指令	TRAPA	
槽非法指令	紧接在延迟转移指令（延迟槽）之后的未定义代码（包括 FPU 模块待机时的 FPU 指令和 FPU 相关的 CPU 指令）、PC 改写指令、32 位指令、RESBANK 指令、DIVS 指令或者 DIVU 指令	延迟转移指令：JMP、JSR、BRA、BSR、RTS、RTE、BF/S、BT/S、BSRF、BRA PC 改写指令：JMP、JSR、BRA、BSR、RTS、RTE、BT、BF、TRAPA、BF/S、BT/S、BSRF、BRA、JSR/N、RTV/N 32 位指令：BAND.B、BANDNOT.B、BCLR.B、BLD.B、BLDNOT.B、BOR.B、BORNOT.B、BSET.B、BST.B、BXOR.B、MOV.B@disp12、MOV.W@disp12、MOV.L@disp12、MOVI20、MOVI20S、MOVU.B、MOVU.W
一般非法指令	不在延迟槽的未定义代码（包括 FPU 模块待机的 FPU 指令和 FPU 相关的 CPU 指令）	
整数除法运算异常	被零除	DIVU、DIVS
	负的最大值 $\div(-1)$	DIVS
FPU 异常	IEEE754 规格定义的无效运算异常或者引起被零除运算异常的指令或者可能引起上溢、下溢和不精确异常的指令	FADD、FSUB、FMUL、FDIV、FMAC、FCMP/EQ、FCMP/GT、FLOAT、FTRC、FCNVDS、FCNVSD、FSQRT

6.6.2 陷阱指令

当执行 TRAPA 指令时，开始陷阱指令的异常处理。此时，CPU 的运行如下：

1. 从异常处理向量表中取与 TRAPA 指令指定的向量号对应的异常服务程序的起始地址。
2. 将状态寄存器（SR）压栈。
3. 将程序计数器（PC）压栈。被压栈的 PC 值是 TRAPA 指令的下一个指令起始地址。
4. 转移到从异常处理向量表中取到的异常服务程序的起始地址，开始执行程序。此时的转移不是延迟转移。

6.6.3 槽非法指令

将紧接在延迟转移指令之后的指令称为“分配在延迟槽的指令”。在分配给延迟槽的指令为未定义代码（包括 FPU 模块待机时的 FPU 指令和 FPU 相关的 CPU 指令）、PC 改写指令、32 位指令、RESBANK 指令、DIVS 指令或者 DIVU 指令时，如果对这些指令进行解码，就开始槽非法指令的异常处理。而且，在将 FPU 置为模块待机状态时，浮点指令和 FPU 相关的 CPU 指令作为未定义代码处理，如果分配在延迟槽并且对该指令进行解码，就开始槽非法指令的异常处理。

在进行槽非法指令的异常处理时，CPU 的运行如下：

1. 从异常处理向量表中取异常服务程序的起始地址。

2. 将状态寄存器（SR）压栈。
3. 将程序计数器（PC）压栈。被压栈的PC值为未定义代码、PC改写指令、32位指令、RESBANK指令、DIVS指令或者DIVU指令之前的延迟转移指令的转移地址。
4. 转移到从异常处理向量表中取到的异常服务程序的起始地址，开始执行程序。此时的转移不是延迟转移。

6.6.4 一般非法指令

如果对不紧接在延迟转移指令（延迟槽）之后的未定义代码（包括FPU模块待机时的FPU指令和FPU相关的CPU指令）进行解码，就开始一般非法指令的异常处理。另外，在将FPU置为模块待机状态时，浮点指令和FPU相关的CPU指令作为未定义代码处理，如果不分配在紧接着延迟转移指令（延迟槽）之后并且对该指令进行解码，就开始一般非法指令的异常处理。

在进行一般非法指令的异常处理时，CPU按照和槽非法指令的异常处理相同的步骤运行。但是，被压栈的PC值和槽非法指令的异常处理不同，是此未定义代码的起始地址。

6.6.5 整数除法运算异常

如果整数除法指令进行被零除运算或者整数除法运算的结果发生上溢，就发生整数除法运算异常。被零除运算异常源的指令是DIVU和DIVS，上溢异常源的指令只有DIVS，只在用“-1”除负的最大值时才发生异常。在整数除法运算异常发生时，CPU的运行如下：

1. 从异常处理向量表中取与发生的整数除法运算异常对应的异常服务程序的起始地址。
2. 将状态寄存器（SR）压栈。
3. 将程序计数器（PC）压栈。被压栈的PC值为发生异常的整数除法指令的起始地址。
4. 转移到从异常处理向量表中取到的异常服务程序的起始地址，开始执行程序。此时的转移不是延迟转移。

6.6.6 FPU异常

当浮点状态/控制寄存器（FPSCR）的FPU异常允许字段（Enable）中的V、Z、O、U或者I位被置位时，发生FPU异常。这表示浮点运算指令引起了IEEE754规格定义的有效运算异常、被零除运算异常、上溢（有可能的指令）、下溢（有可能的指令）以及不精确异常（有可能的指令）。

FPU异常处理发生源的浮点运算指令如下所示：

FADD、FSUB、FMUL、FDIV、FMAC、FCMP/EQ、FCMP/GT、FLOAT、FTRC、FCNVDS、FCNVSD、FSQRT

只在对应的FPU异常允许位（Enable）被置位时才发生FPU异常。当FPU检测到浮点运算引起的异常源时，停止FPU的运行并且通知CPU发生FPU异常处理。当开始异常处理时，CPU的运行如下：

1. 从异常处理向量表中取与发生的FPU异常对应的异常服务程序的起始地址。
2. 将状态寄存器（SR）压栈。
3. 将程序计数器（PC）压栈。被压栈的PC值为最后执行指令的下一个指令起始地址。
4. 转移到从异常处理向量表中取到的异常服务程序的起始地址，开始执行程序。此时的转移不是延迟转移。

和是否接受FPU异常无关，随时更新FPSCR的FPU异常标志字段（Flag），并且将置位的状态保持到用户明确地用指令清除为止。每执行一次浮点运算指令，FPSCR的FPU异常源字段（Cause）就会变化。

另外，在FPSCR的FPU异常允许字段（Enable）中的V位被置位并且FPSCR的QIS位被置位的状态下，如果给浮点运算指令源输入qNaN或者 $\pm\infty$ ，就发生FPU异常。

6.7 不接受异常处理的情况

如表 6.10 所示，如果在紧接延迟转移指令之后发生地址错误、FPU 异常、寄存器组错误（上溢）和中断，就可能不立即被接受而被保留。当对能接受异常的指令进行解码时，接受这些异常。

表 6.10 在紧接着延迟转移指令之后发生的异常源

发生时间	异常源			
	地址错误	浮点单元异常	寄存器组错误（上溢）	中断
紧接在延迟转移指令*之后	×	×	×	×

【符号说明】×：不接受

【注】* 延迟转移指令：JMP、JSR、BRA、BSR、RTS、RTE、BF/S、BT/S、BSRF、BRA#

6.8 异常处理后的堆栈状态

异常处理结束后的堆栈状态如表 6.11 所示。

表 6.11 异常处理结束后的堆栈状态

种类	堆栈状态	种类	堆栈状态
地址错误		中断	
寄存器组错误（上溢）		寄存器组错误（下溢）	
陷阱指令		槽非法指令	
一般非法指令		整数除法运算异常	
FPU 异常			

6.9 使用时的注意事项

6.9.1 堆栈指针（SP）的值

SP 的值必须是 4 的倍数。否则，如果在异常处理中存取堆栈，就会发生地址错误。

6.9.2 向量基址寄存器（VBR）的值

VBR 的值必须是 4 的倍数。否则，如果在异常处理中存取堆栈，就会发生地址错误。

6.9.3 在地址错误异常处理的堆栈存取中发生的地址错误

如果 SP 不是 4 的倍数，就会在异常处理（中断等）的堆栈存取中发生地址错误，并且在该异常处理结束后转移到地址错误的异常处理。即使在地址错误异常处理的堆栈存取中也会发生地址错误，但是为了不使地址错误异常处理的堆栈存取无限制继续，不接受此时的地址错误。因此，能将程序的控制转移到地址错误的异常服务程序，进行错误处理。

如果在异常处理的堆栈存取中发生地址错误，就执行堆栈存取的总线周期（写）。在 SR 和 PC 的堆栈存取中 SP 分别减 4，因此即使在堆栈存取结束后，SP 的值也不是 4 的倍数。另外，堆栈存取时输出的地址值是 SP 的值，因此输出发生错误的地址。此时，堆栈存取的写数据为不定值。

6.9.4 通过改变中断屏蔽位进行的中断控制

当用 LDC 指令和 LDC.L 指令操作状态寄存器（SR）的中断屏蔽位（I3 ~ I0）的值使中断从禁止变为允许时，可能在允许中断的指令后的 5 条指令执行期间不接受中断。

因此，当用 LDC 指令和 LDC.L 指令操作状态寄存器（SR）的中断屏蔽位（I3 ~ I0）的值使中断从禁止变为允许时，必须在允许中断的指令和禁止中断的指令之间至少配置 5 条指令。

6.9.5 异常处理前的注意事项

在异常处理前，需要事先将异常处理向量表保存到存储器，使 CPU 能存取该存储器。因此，如果在以下的状态下发生异常处理，CPU 就会取到非预期的值，作为程序的执行起始地址，从非预期的地址开始执行程序。

- 例 1：将异常处理向量表保存在外部地址空间，但是还没有设定用于存取该外部地址空间的总线状态控制器和通用输入/输出端口。
- 例 2：将异常处理向量表保存在内部 RAM，但是还没有将向量基址寄存器（VBR）改为内部 RAM 的地址。

(1) 手动复位

在结束存取外部 CS0 空间所需的设定前，不能发生手动复位。否则，CPU 总是从手动复位的向量表地址偏移量（H'00000008）的位置（即从外部 CS0 空间）取程序的执行起始地址。

另外，即使在引导模式 2 ~ 5 中外部 CS0 空间不连接存储器，也不能发生手动复位。

(2) NMI 中断

在将异常处理向量表保存到存储器并且结束存取该存储器的设定前，不能设定为允许中断。

尤其在引导模式 2 ~ 5 中执行引导的过程中（在装入程序传送结束后转移到高速内部 RAM 前），VBR 保持初始值 H'00000000 的状态。在装入程序中更改 VBR 或者结束存取外部地址空间的设定前，不能设定为允许中断。

(3) NMI 以外的中断

在将异常处理向量表保存到存储器并且结束存取该存储器的设定前，不能设定为允许中断。

(4) 其他异常

在将异常处理向量表保存到存储器并且结束存取该存储器的设定前，不能发生异常处理。

7. 中断控制器

中断控制器判断中断源的优先级以及控制对 CPU 的中断请求。本模块有设定各中断优先级的寄存器，用户根据该寄存器设定的优先级处理中断请求。

7.1 特点

- 能将中断优先级设定为 16 级。
能通过 24 个中断优先级设定寄存器，按请求源将 IRQ 中断、PINT 中断和内部外围模块中断的优先级设定为 16 级。
- NMI 噪声消除功能
有表示 NMI 引脚状态的 NMI 输入电平位。能通过中断异常服务程序读此位来确认引脚状态，并且能用作噪声消除功能。
- 寄存器组
本 LSI 内置寄存器组，能在进行中断处理的同时，高速进行寄存器的保存和恢复。

框图如图 7.1 所示。

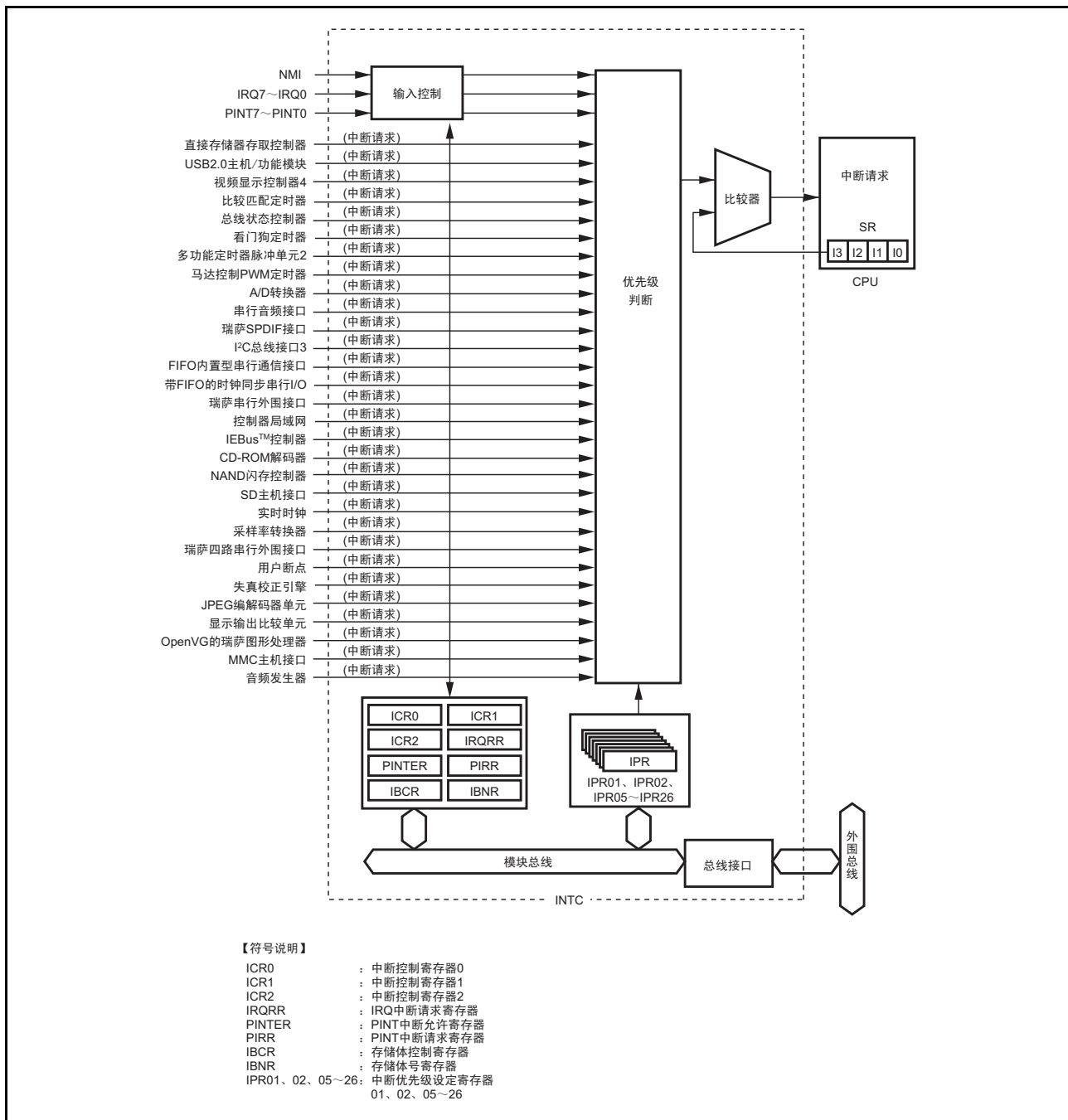


图 7.1 框图

7.2 输入/输出引脚

引脚结构如表 7.1 所示

表 7.1 引脚结构

名称	引脚名	输入/输出	功能
非屏蔽中断输入引脚	NMI	输入	输入不能屏蔽的中断请求信号。
中断请求输入引脚	IRQ7 ~ IRQ0	输入	输入能屏蔽的中断请求信号。
	PINT7 ~ PINT0	输入	

7.3 寄存器说明

寄存器结构如表 7.2 所示。通过这些寄存器进行中断优先级的设定以及外部中断输入信号的检测控制等。

表 7.2 寄存器结构

寄存器名	略称	R/W	初始值	地址	存取长度
中断控制寄存器 0	ICR0	R/W	*1	H'FFFE0800	16、32
中断控制寄存器 1	ICR1	R/W	H'0000	H'FFFE0802	16、32
中断控制寄存器 2	ICR2	R/W	H'0000	H'FFFE0804	16、32
IRQ 中断请求寄存器	IRQRR	R/(W)*2	H'0000	H'FFFE0806	16、32
PINT 中断允许寄存器	PINTER	R/W	H'0000	H'FFFE0808	16、32
PINT 中断请求寄存器	PIRR	R	H'0000	H'FFFE080A	16、32
存储体控制寄存器	IBCR	R/W	H'0000	H'FFFE080C	16、32
存储体号寄存器	IBNR	R/W	H'0000	H'FFFE080E	16、32
中断优先级设定寄存器 01	IPR01	R/W	H'0000	H'FFFE0818	16、32
中断优先级设定寄存器 02	IPR02	R/W	H'0000	H'FFFE081A	16、32
中断优先级设定寄存器 05	IPR05	R/W	H'0000	H'FFFE0820	16、32
中断优先级设定寄存器 06	IPR06	R/W	H'0000	H'FFFE0C00	16、32
中断优先级设定寄存器 07	IPR07	R/W	H'0000	H'FFFE0C02	16、32
中断优先级设定寄存器 08	IPR08	R/W	H'0000	H'FFFE0C04	16、32
中断优先级设定寄存器 09	IPR09	R/W	H'0000	H'FFFE0C06	16、32
中断优先级设定寄存器 10	IPR10	R/W	H'0000	H'FFFE0C08	16、32
中断优先级设定寄存器 11	IPR11	R/W	H'0000	H'FFFE0C0A	16、32
中断优先级设定寄存器 12	IPR12	R/W	H'0000	H'FFFE0C0C	16、32
中断优先级设定寄存器 13	IPR13	R/W	H'0000	H'FFFE0C0E	16、32
中断优先级设定寄存器 14	IPR14	R/W	H'0000	H'FFFE0C10	16、32
中断优先级设定寄存器 15	IPR15	R/W	H'0000	H'FFFE0C12	16、32
中断优先级设定寄存器 16	IPR16	R/W	H'0000	H'FFFE0C14	16、32
中断优先级设定寄存器 17	IPR17	R/W	H'0000	H'FFFE0C16	16、32
中断优先级设定寄存器 18	IPR18	R/W	H'0000	H'FFFE0C18	16、32
中断优先级设定寄存器 19	IPR19	R/W	H'0000	H'FFFE0C1A	16、32
中断优先级设定寄存器 20	IPR20	R/W	H'0000	H'FFFE0C1C	16、32
中断优先级设定寄存器 21	IPR21	R/W	H'0000	H'FFFE0C1E	16、32
中断优先级设定寄存器 22	IPR22	R/W	H'0000	H'FFFE0C20	16、32
中断优先级设定寄存器 23	IPR23	R/W	H'0000	H'FFFE0C22	16、32
中断优先级设定寄存器 24	IPR24	R/W	H'0000	H'FFFE0C24	16、32
中断优先级设定寄存器 25	IPR25	R/W	H'0000	H'FFFE0C26	16、32
中断优先级设定寄存器 26	IPR26	R/W	H'0000	H'FFFE0C28	16、32

【注】 *1 当 NMI 引脚为高电平时是“H'8000”，否则为“H'0001”。

*2 为了清除标志，只能在读“1”后写“0”。

7.3.1 中断优先级设定寄存器 01、02、05 ~ 26 (IPR01、IPR02、IPR05 ~ IPR26)

IPR01、IPR02、IPR05 ~ IPR26 分别是 16 位可读写寄存器，设定 IRQ 中断、PINT 中断和内部外围模块中断的优先级（级 0 ~ 15）。中断请求源和 IPR01、IPR02、IPR05 ~ IPR26 各位的对应关系如表 7.3 所示。

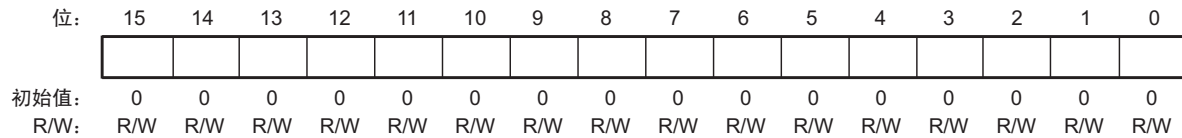


表 7.3 中断请求源和 IPR01、IPR02、IPR05 ~ IPR26

寄存器名	位			
	15 ~ 12	11 ~ 8	7 ~ 4	3 ~ 0
IPR01	IRQ0	IRQ1	IRQ2	IRQ3
IPR02	IRQ4	IRQ5	IRQ6	IRQ7
IPR05	PINT7 ~ PINT0	保留	保留	保留
IPR06	直接存储器存取 控制器 0	直接存储器存取 控制器 1	直接存储器存取 控制器 2	直接存储器存取 控制器 3
IPR07	直接存储器存取 控制器 4	直接存储器存取 控制器 5	直接存储器存取 控制器 6	直接存储器存取 控制器 7
IPR08	直接存储器存取 控制器 8	直接存储器存取 控制器 9	直接存储器存取 控制器 10	直接存储器存取 控制器 11
IPR09	直接存储器存取 控制器 12	直接存储器存取 控制器 13	直接存储器存取 控制器 14	直接存储器存取 控制器 15
IPR10	USB2.0 主机 / 功能模块	视频显示控制器 4	视频显示控制器 4	视频显示控制器 4
IPR11	失真校正引擎	JPEG 编解码器单元	显示输出比较单元	OpenVG 的 瑞萨图形处理器
IPR12	比较匹配通道 0	比较匹配通道 1	总线状态控制器	看门狗定时器
IPR13	多功能定时器脉冲单元 2 通道 0 (TGI0A ~ TGI0D)	多功能定时器脉冲单元 2 通道 0 (TGI0V、TGI0E、TGI0F)	多功能定时器脉冲单元 2 通道 1 (TGI1A、TGI1B)	多功能定时器脉冲单元 2 通道 1 (TGI1V、TGI1U)
IPR14	多功能定时器脉冲单元 2 通道 2 (TGI2A、TGI2B)	多功能定时器脉冲单元 2 通道 2 (TGI2V、TGI2U)	多功能定时器脉冲单元 2 通道 3 (TGI3A ~ TGI3D)	多功能定时器脉冲单元 2 通道 3 (TGI3V)
IPR15	多功能定时器脉冲单元 2 通道 4 (TGI4A ~ TGI4D)	多功能定时器脉冲单元 2 通道 4 (TGI4V)	马达控制 PWM 定时器 通道 1	马达控制 PWM 定时器 通道 2
IPR16	音频发生器 通道 0	音频发生器 通道 1	音频发生器 通道 2	音频发生器 通道 3
IPR17	A/D 转换器	串行音频接口 通道 0	串行音频接口 通道 1	串行音频接口 通道 2
IPR18	串行音频接口 通道 3	串行音频接口 通道 4	串行音频接口 通道 5	瑞萨 SPDIF 接口
IPR19	I ² C 总线接口 3 通道 0	I ² C 总线接口 3 通道 1	I ² C 总线接口 3 通道 2	I ² C 总线接口 3 通道 3
IPR20	FIFO 内置型串行通信接口 通道 0	FIFO 内置型串行通信接口 通道 1	FIFO 内置型串行通信接口 通道 2	FIFO 内置型串行通信接口 通道 3
IPR21	FIFO 内置型串行通信接口 通道 4	FIFO 内置型串行通信接口 通道 5	FIFO 内置型串行通信接口 通道 6	FIFO 内置型串行通信接口 通道 7

寄存器名	位			
	15 ~ 12	11 ~ 8	7 ~ 4	3 ~ 0
IPR22	带 FIFO 的时钟同步 串行 I/O	控制器局域网 通道 0	控制器局域网 通道 1	控制器局域网 通道 2
IPR23	瑞萨串行外围接口 通道 0	瑞萨串行外围接口 通道 1	瑞萨四路串行外围接口 通道 0	瑞萨四路串行外围接口 通道 1
IPR24	IEBus™ 控制器	CD-ROM 解码器	NAND 闪存控制器	MMC 主机接口
IPR25	SD 主机接口 通道 0	SD 主机接口 通道 1	实时时钟	保留
IPR26	采样率转换器 通道 0	采样率转换器 通道 1	采样率转换器 通道 2	保留

如表 7.3 所示，通过给 bit15 ~ 12、bit11 ~ 8、bit7 ~ 4、bit3 ~ 0 的各 4 位置 H'0 (0000) ~ H'F (1111) 范围的值，设定各自对应的中断优先级。中断优先级在置“H'0”时为优先级 0（最低），在置“H'F”时为优先级 15（最高）。

7.3.2 中断控制寄存器 0 (ICR0)

ICR0 是 16 位寄存器，屏蔽或者允许外部中断输入引脚 NMI，设定输入信号的检测模式，并且表示 NMI 引脚的输入电平和中断请求。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NMIL	—	—	—	—	—	—	NMIE	—	—	—	—	—	—	NMIF	NMIM
初始值:	*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/(W)*2

【注】*1 当NMI引脚是高电平时为“1”，当NMI引脚是低电平时为“0”。

*2 只在写“0”时有效。

位	位名	初始值	R/W	说明
15	NMIL	*	R	NMI 输入电平 设定 NMI 引脚输入信号的电平。能通过读此位得知 NMI 引脚的电平。 写操作无效。 0: NMI 引脚输入低电平 1: NMI 引脚输入高电平
14 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	NMIE	0	R/W	NMI 边沿选择 选择在 NMI 输入的下降沿或者上升沿检测中断请求信号。 0: 在 NMI 输入的下降沿检测中断请求 1: 在 NMI 输入的上升沿检测中断请求
7 ~ 2	—	全 0	R	保留位 读写值都为“0”。
1	NMIF	0	R	NMI 中断请求 表示 NMI 引脚的中断请求状态。写操作无效。 0: 没有检测到 NMI 中断请求 [清除条件] • 当更改 NMIE 位的设定值时 • 当执行 NMI 中断异常处理时 1: 检测到 NMI 中断请求 [置位条件] • 当 NMI 引脚发生 NMIE 对应的边沿时
0	NMIM	1	R/(W)*2	NMI 屏蔽 指定是屏蔽还是允许输入 NMI 引脚的中断请求。 0: 允许 NMI 引脚中断 1: 屏蔽 NMI 引脚中断

7.3.3 中断控制寄存器 1 (ICR1)

ICR1 是 16 位寄存器，分别对外部中断输入引脚 IRQ7 ~ IRQ0 指定低电平、下降沿、上升沿或者双边沿的检测模式。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRQ71S	IRQ70S	IRQ61S	IRQ60S	IRQ51S	IRQ50S	IRQ41S	IRQ40S	IRQ31S	IRQ30S	IRQ21S	IRQ20S	IRQ11S	IRQ10S	IRQ01S	IRQ00S
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	IRQ71S	0	R/W	IRQ 检测选择 选择是在低电平还是在下降沿、上升沿、双边沿检测 IRQ7 ~ IRQ0 引脚的中断信号。 00: 在 IRQn 输入的低电平检测中断请求 01: 在 IRQn 输入的下降沿检测中断请求 10: 在 IRQn 输入的上升沿检测中断请求 11: 在 IRQn 输入的双边沿检测中断请求
14	IRQ70S	0	R/W	
13	IRQ61S	0	R/W	
12	IRQ60S	0	R/W	
11	IRQ51S	0	R/W	
10	IRQ50S	0	R/W	
9	IRQ41S	0	R/W	
8	IRQ40S	0	R/W	
7	IRQ31S	0	R/W	
6	IRQ30S	0	R/W	
5	IRQ21S	0	R/W	
4	IRQ20S	0	R/W	
3	IRQ11S	0	R/W	
2	IRQ10S	0	R/W	
1	IRQ01S	0	R/W	
0	IRQ00S	0	R/W	

【符号说明】 n=7 ~ 0

7.3.4 中断控制寄存器 2 (ICR2)

ICR2 是 16 位寄存器，分别给外部中断输入引脚 PINT7 ~ PINT0 指定低电平或者高电平的检测模式。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PINT7S	PINT6S	PINT5S	PINT4S	PINT3S	PINT2S	PINT1S	PINT0S
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 8	—	全 0	R	保留位 读写值都为“0”。
7	PINT7S	0	R/W	PINT 的检测选择 选择是在低电平还是在高电平检测 PINT7 ~ PINT0 引脚的中断信号。 0: 在 PINTn 输入的低电平检测中断请求 1: 在 PINTn 输入的高电平检测中断请求
6	PINT6S	0	R/W	
5	PINT5S	0	R/W	
4	PINT4S	0	R/W	
3	PINT3S	0	R/W	
2	PINT2S	0	R/W	
1	PINT1S	0	R/W	
0	PINT0S	0	R/W	

【符号说明】 n=7 ~ 0

7.3.5 IRQ 中断请求寄存器 (IRQRR)

IRQRR 是 16 位寄存器，表示外部中断输入引脚 IRQ7 ~ IRQ0 的中断请求。当将 IRQ7 ~ IRQ0 中断设定为边沿检测时，通过在读 IRQ7F ~ IRQ0F 为“1”后给 IRQ7 ~ IRQ0 写“0”，能取消被保持的中断请求。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	IRQ7F	IRQ6F	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

【注】* 为了清除标志，只能在读“1”后写“0”。

位	位名	初始值	R/W	说明
15 ~ 8	—	全 0	R	保留位 读写值都为“0”。
7	IRQ7F	0	R/(W)*	IRQ 中断请求 表示 IRQ7 ~ IRQ0 中断请求的状态。 电平检测时 0: 不存在 IRQn 中断请求 [清除条件] • 当 IRQn 输入为高电平时 1: 存在 IRQn 中断请求 [置位条件] • 当 IRQn 输入为低电平时 边沿检测时 0: 未检测到 IRQn 中断请求 [清除条件] • 在读 IRQnF 为“1”的状态后写“0”时 • 当执行 IRQn 中断异常处理时 1: 检测到 IRQn 中断请求 [置位条件] • 当 IRQn 引脚发生 ICR1 的 IRQn1S 或者 IRQn0S 对应的边沿时
6	IRQ6F	0	R/(W)*	
5	IRQ5F	0	R/(W)*	
4	IRQ4F	0	R/(W)*	
3	IRQ3F	0	R/(W)*	
2	IRQ2F	0	R/(W)*	
1	IRQ1F	0	R/(W)*	
0	IRQ0F	0	R/(W)*	

【符号说明】 n=7 ~ 0

7.3.6 PINT 中断允许寄存器 (PINTER)

PINTER 是 16 位寄存器，允许对外部中断输入引脚 PINT7 ~ PINT0 输入中断请求。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PINT7E	PINT6E	PINT5E	PINT4E	PINT3E	PINT2E	PINT1E	PINT0E
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 8	—	全 0	R	保留位 读写值都为“0”。
7	PINT7E	0	R/W	PINT 允许位 选择是否允许对 PINT7 ~ PINT0 引脚输入中断请求。 0: 禁止 PINTn 输入中断请求 1: 允许 PINTn 输入中断请求
6	PINT6E	0	R/W	
5	PINT5E	0	R/W	
4	PINT4E	0	R/W	
3	PINT3E	0	R/W	
2	PINT2E	0	R/W	
1	PINT1E	0	R/W	
0	PINT0E	0	R/W	

【符号说明】 n=7 ~ 0

7.3.7 PINT 中断请求寄存器 (PIRR)

PIRR 是 16 位寄存器，表示外部中断 PINT7 ~ PINT0 的中断请求。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PINT7R	PINT6R	PINT5R	PINT4R	PINT3R	PINT2R	PINT1R	PINT0R
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 8	—	全 0	R	保留位 读写值都为“0”。
7	PINT7R	0	R	PINT 中断请求 表示 PINT7 ~ PINT0 中断请求。 0: PINTn 引脚没有中断请求 1: PINTn 引脚有中断请求
6	PINT6R	0	R	
5	PINT5R	0	R	
4	PINT4R	0	R	
3	PINT3R	0	R	
2	PINT2R	0	R	
1	PINT1R	0	R	
0	PINT0R	0	R	

【符号说明】 n=7 ~ 0

7.3.8 存储体控制寄存器 (IBCR)

对中断优先级，IBCR 能允许或者禁止使用寄存器组。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	E15	E14	E13	E12	E11	E10	E9	E8	E7	E6	E5	E4	E3	E2	E1	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

位	位名	初始值	R/W	说明
15	E15	0	R/W	允许 设定允许或者禁止中断优先级 15 ~ 1 的中断使用寄存器组。但是，总是禁止用户断点中断使用寄存器组。 0: 禁止使用寄存器组 1: 允许使用寄存器组
14	E14	0	R/W	
13	E13	0	R/W	
12	E12	0	R/W	
11	E11	0	R/W	
10	E10	0	R/W	
9	E9	0	R/W	
8	E8	0	R/W	
7	E7	0	R/W	
6	E6	0	R/W	
5	E5	0	R/W	
4	E4	0	R/W	
3	E3	0	R/W	
2	E2	0	R/W	
1	E1	0	R/W	
0	—	0	R	保留位 读写值都为“0”。

7.3.9 存储体号寄存器 (IBNR)

IBNR 允许或者禁止使用寄存器组，允许或者禁止寄存器组的上溢异常，并且通过 BN3 ~ BN0 表示下次要保存的存储体号。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BE[1:0]		BOVE	—	—	—	—	—	—	—	—	BN[3:0]				
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15、14	BE[1:0]	00	R/W	寄存器组允许 设定允许或者禁止使用寄存器组。 00: 禁止全部中断使用存储体，忽视 IBCR 的设定。 01: 允许 NMI 和用户断点以外的全部中断使用存储体，忽视 IBCR 的设定。 10: 保留（禁止设定） 11: 寄存器组的使用情况取决于 IBCR 的设定
13	BOVE	0	R/W	寄存器组上溢允许 设定允许或者禁止寄存器组的上溢异常。 0: 禁止寄存器组发生上溢异常 1: 允许寄存器组发生上溢异常
12 ~ 4	—	全 0	R	保留位 读写值都为“0”。
3 ~ 0	BN[3:0]	0000	R	存储体号 表示下次要保存的存储体号。当接受使用寄存器组的中断时，保存到 BN3 ~ BN0 所示的寄存器组，然后 BN 加 1。通过执行寄存器组恢复指令，在 BN 减 1 后从寄存器组恢复。

7.4 中断源

中断源分为 NMI、用户断点、用户调试接口、IRQ、PINT、内部外围模块共 6 种。各中断的优先级用中断优先级的值（0 ~ 16）表示，优先级 0 为最低，优先级 16 为最高。如果设定为优先级 0，该中断总是被屏蔽。

7.4.1 NMI 中断

NMI 中断是优先级 16 的中断，如果将中断控制寄存器 0（ICR0）的 NMI 屏蔽位（NMIE）设定为允许，就总是被接受。检测 NMI 引脚的输入边沿，检测边沿能通过设定中断控制寄存器 0（ICR0）的 NMI 边沿选择位（NMIE），选择上升沿或者下降沿。

NMI 中断优先级为 16，但是能通过 NMI 中断异常处理将状态寄存器（SR）的中断屏蔽位（I3 ~ I0）设定为 15。

当 ICR0 的 NMIM 位为“1”（屏蔽 NMI 中断）时，不发生 NMI 中断，但是能检测到 ICR0 的 NMI 边沿选择位（NMIE）选择的边沿，并且保留中断请求。能通过读 NMI 中断请求位（NMIF），确认是否检测到 NMI 中断请求。如果在 NMIF 位为“1”的状态下给 NMIM 位写“0”（允许 NMI 中断），就接受被保留的 NMI 中断。NMIM 位是只能写“0”的位，因此在允许 NMI 中断后无法再次将其设定为屏蔽。另外，如果更改 NMIE 位的值，就在此时取消被保留的 NMI 中断请求。

要通过 NMI 解除软件待机模式时，必须在转移到软件待机模式前，通过 NMIF 位确认没有检测到或者没有保留中断请求，然后给 NMIM 位写“0”（允许 NMI 中断）。如果在 NMIM 位为“1”（屏蔽 NMI 中断）的状态下转移到软件待机模式，就无法通过 NMI 解除软件待机模式。此时，因为在软件待机模式中不检测 NMI 边沿，所以即使通过其他中断源解除软件待机模式并且将 NMIM 位置为允许，也不发生 NMI 中断。要在软件待机模式中检测 NMI 边沿时，必须将 NMI 引脚的输入电平（在检测到下降沿时为低电平，在检测到上升沿时为高电平）保持到由其他中断源解除软件待机模式为止。

如果转移到深度待机模式，就能通过 NMI 解除深度待机模式，与转移到深度待机模式前的 NMIM 位的设定无关。在解除深度待机模式后的上电复位时，将 NMIM 位初始化为“1”（屏蔽 NMI 中断）。

要在给 NMIM 位写“0”（允许 NMI 中断）后执行 SLEEP 指令时，必须在读 NMIM 位后执行 SLEEP 指令。

7.4.2 用户断点中断

用户断点中断是在用户断点控制器设定的断点条件成立时发生的中断，优先级为 15。通过用户断点中断异常处理，将 SR 的 I3 ~ I0 位设定为 15。有关用户断点，请参照“8. 用户断点控制器”。

7.4.3 用户调试接口中断

用户调试接口中断是在串行输入用户调试接口中断指令时发生的中断，优先级为 15。在边沿检测到用户调试接口中断请求，并且被保持到接受为止。通过用户调试接口中断异常处理，将 SR 的 I3 ~ I0 位设定为 15。有关用户调试接口中断，请参照“50. 用户调试接口”。

7.4.4 IRQ 中断

IRQ 中断是由 IRQ7 ~ IRQ0 引脚输入引起的中断。IRQ 中断能通过设定中断控制寄存器 1 (ICR1) 的 IRQ 检测选择位 (IRQ71S ~ IRQ01S 和 IRQ70S ~ IRQ00S)，给每个引脚选择低电平、下降沿、上升沿或者双边沿检测。能通过中断优先级设定寄存器 01 和中断优先级设定寄存器 02 (IPR01 和 IPR02) 给每个引脚设定 0 ~ 15 的优先级。

如果将 IRQ 中断设定为低电平检测，就在 IRQ7 ~ IRQ0 引脚为低电平期间将中断请求信号送给中断控制器，而在 IRQ7 ~ IRQ0 引脚变为高电平时不将中断请求信号送给中断控制器。能通过读 IRQ 中断请求寄存器 (IRQRR) 的 IRQ 中断请求位 (IRQ7F ~ IRQ0F) 确认中断请求。

如果将 IRQ 中断设定为边沿检测，就通过 IRQ7 ~ IRQ0 引脚的变化检测到中断请求，并且将中断请求信号送给中断控制器。IRQ 中断请求的检测结果一直保持到该中断请求被接受为止。另外，能通过读 IRQRR 的 IRQ7F ~ IRQ0F 位确认是否检测到 IRQ 中断请求，能通过在读“1”后写“0”来取消 IRQ 中断请求的检测结果。

通过 IRQ 中断异常处理，将 SR 的 I3 ~ I0 位设定为被接受的 IRQ 中断优先级的值。

在从 IRQ 中断异常服务程序返回时，为了避免再次误接受，必须在通过 IRQ 中断请求寄存器 (IRQRR) 确认中断请求已被清除后执行 RTE 指令。

7.4.5 PINT 中断

PINT 中断是由 PINT7 ~ PINT0 引脚输入引起的中断。通过 PINT 中断允许寄存器 (PINTER) 的 PINT 允许位 (PINT7E ~ PINT0E) 来允许中断请求的输入。PINT7 ~ PINT0 能通过设定中断控制寄存器 2 (ICR2) 的 PINT 检测选择位 (PINT7S ~ PINT0S)，给每个引脚选择低电平检测或者高电平检测。能通过中断优先级设定寄存器 05 (IPR05) 的 bit15 ~ 12，一次性地给 PINT7 ~ PINT0 设定 0 ~ 15 的优先级。

如果将 PINT7 ~ PINT0 设定为低电平检测，就在 PINT 引脚为低电平期间将中断请求信号送给中断控制器，而在 PINT 引脚变为高电平时不将中断请求信号送给中断控制器。能通过读 PINT 中断请求寄存器 (PIRR) 的 PINT 中断请求位 (PINT7R ~ PINT0R) 确认中断请求优先级。在设定为高电平检测时，除了极性相反的情况以外都相同。通过 PINT 中断异常处理，将 SR 的 I3 ~ I0 设定为 PINT 中断优先级的值。

在从 PINT 中断异常服务程序返回时，为了避免再次误接受，必须在通过 PINT 中断请求寄存器 (PIRR) 确认中断请求已被清除后执行 RTE 指令。

7.4.6 内部外围模块中断

内部外围模块中断是由以下所示的内部外围模块发生的中断：

- 直接存储器存取控制器
- USB2.0 主机/功能模块
- 视频显示控制器 4
- 失真校正引擎
- 显示输出比较单元
- JPEG 编解码器单元
- OpenVG 的瑞萨图形处理器
- 比较匹配定时器
- 总线状态控制器
- 看门狗定时器
- 多功能定时器脉冲单元 2
- 马达控制 PWM 定时器
- 音频发生器
- A/D 转换器
- 串行音频接口
- 瑞萨 SPDIF 接口
- I²C 总线接口 3
- FIFO 内置型串行通信接口
- 带 FIFO 的时钟同步串行 I/O
- 瑞萨串行外围接口
- 瑞萨四路串行外围接口
- 控制器局域网
- IEBus™ 控制器
- CD-ROM 解码器
- NAND 闪存控制器
- SD 主机接口
- MMC 主机接口
- 实时时钟
- 采样率转换器

因为给各中断源分配了不同的中断向量，所以不需要在异常服务程序中判断中断源。能通过中断优先级设定寄存器 05 ~ 26 (IPR05 ~ IPR26)，给每个模块设定 0 ~ 15 的优先级。通过内部外围模块中断的异常处理，将 SR 的 I3 ~ I0 位设定被接受的内部外围模块中断优先级的值。

7.5 中断异常处理向量表和优先级

中断源、向量号、向量地址偏移量和中断优先级如表 7.4 所示。

给各中断源分别分配了不同的向量号和向量地址偏移量，从向量号和向量地址偏移量算出向量地址。通过中断异常处理从向量地址所示的向量表中取中断异常服务程序的起始地址。有关向量地址的计算方法，请参照“6. 异常处理”表 6.4 的异常处理向量地址的计算方法。

能通过中断优先级设定寄存器 01、02、05 ~ 26 (IPR01、IPR02、IPR05 ~ IPR26)，给每个引脚或者模块任意设定 IRQ 中断、PINT 中断和内部外围模块中断的 0 ~ 15 的优先级。但是，在 IPR05 ~ IPR26 中发生同一个 IPR 内的多个中断时，优先级取决于表 7.4 所示的 IPR 设定单位内的优先级并且不能被更改。通过上电复位，将 IRQ 中断、PINT 中断和内部外围模块中断的优先级设定为“0”。在将多个中断源设定为相同优先级的情况下，如果同时发生这些中断，就根据如表 7.4 所示的默认优先级进行处理。

表 7.4 中断异常向量和优先级

中断源	中断向量		中断优先级 (初始值)	对应的 IPR (位)	IPR 设定 单位内的 优先级	默认 优先级	
	向量	向量地址偏移量					
NMI	11	H'0000002C ~ H'0000002F	16	—	—	高 ↑	
用户断点	12	H'00000030 ~ H'00000033	15	—	—		
用户调试接口	14	H'00000038 ~ H'0000003B	15	—	—		
IRQ	IRQ0	64	H'00000100 ~ H'00000103	0 ~ 15(0)	IPR01(15 ~ 12)		—
	IRQ1	65	H'00000104 ~ H'00000107	0 ~ 15(0)	IPR01(11 ~ 8)		—
	IRQ2	66	H'00000108 ~ H'0000010B	0 ~ 15(0)	IPR01(7 ~ 4)		—
	IRQ3	67	H'0000010C ~ H'0000010F	0 ~ 15(0)	IPR01(3 ~ 0)		—
	IRQ4	68	H'00000110 ~ H'00000113	0 ~ 15(0)	IPR02(15 ~ 12)		—
	IRQ5	69	H'00000114 ~ H'00000117	0 ~ 15(0)	IPR02(11 ~ 8)		—
	IRQ6	70	H'00000118 ~ H'0000011B	0 ~ 15(0)	IPR02(7 ~ 4)		—
	IRQ7	71	H'0000011C ~ H'0000011F	0 ~ 15(0)	IPR02(3 ~ 0)		—
PINT	PINT0	80	H'00000140 ~ H'00000143	0 ~ 15(0)	IPR05(15 ~ 12)		1
	PINT1	81	H'00000144 ~ H'00000147				2
	PINT2	82	H'00000148 ~ H'0000014B				3
	PINT3	83	H'0000014C ~ H'0000014F			4	
	PINT4	84	H'00000150 ~ H'00000153			5	
	PINT5	85	H'00000154 ~ H'00000157			6	
	PINT6	86	H'00000158 ~ H'0000015B			7	
	PINT7	87	H'0000015C ~ H'0000015F			8	
直接存储器存取控制器	通道 0	DEI0	108	0 ~ 15(0)	IPR06(15 ~ 12)	1	
		HEI0	109			H'000001B4 ~ H'000001B7	2
	通道 1	DEI1	112	0 ~ 15(0)	IPR06(11 ~ 8)	1	
		HEI1	113			H'000001C4 ~ H'000001C7	2
	通道 2	DEI2	116	0 ~ 15(0)	IPR06(7 ~ 4)	1	
		HEI2	117			H'000001D4 ~ H'000001D7	2
	通道 3	DEI3	120	0 ~ 15(0)	IPR06(3 ~ 0)	1	
		HEI3	121			H'000001E4 ~ H'000001E7	2
	通道 4	DEI4	124	0 ~ 15(0)	IPR07(15 ~ 12)	1	
		HEI4	125			H'000001F4 ~ H'000001F7	2
	通道 5	DEI5	128	0 ~ 15(0)	IPR07(11 ~ 8)	1	
		HEI5	129			H'00000204 ~ H'00000207	2
	通道 6	DEI6	132	0 ~ 15(0)	IPR07(7 ~ 4)	1	
		HEI6	133			H'00000214 ~ H'00000217	2

中断源		中断向量		中断优先级 (初始值)	对应的 IPR (位)	IPR 设定 单位内的 优先级	默认 优先级	
		向量	向量表地址偏移量					
直接存储器存取控制器	通道 7	DEI7	136	H'00000220 ~ H'00000223	0 ~ 15(0)	IPR07(3 ~ 0)	1	↑ 高
		HEI7	137	H'00000224 ~ H'00000227			2	
	通道 8	DEI8	140	H'00000230 ~ H'00000233	0 ~ 15(0)	IPR08(15 ~ 12)	1	
		HEI8	141	H'00000234 ~ H'00000237			2	
	通道 9	DEI9	144	H'00000240 ~ H'00000243	0 ~ 15(0)	IPR08(11 ~ 8)	1	
		HEI9	145	H'00000244 ~ H'00000247			2	
	通道 10	DEI10	148	H'00000250 ~ H'00000253	0 ~ 15(0)	IPR08(7 ~ 4)	1	
		HEI10	149	H'00000254 ~ H'00000257			2	
	通道 11	DEI11	152	H'00000260 ~ H'00000263	0 ~ 15(0)	IPR08(3 ~ 0)	1	
		HEI11	153	H'00000264 ~ H'00000267			2	
	通道 12	DEI12	156	H'00000270 ~ H'00000273	0 ~ 15(0)	IPR09(15 ~ 12)	1	
		HEI12	157	H'00000274 ~ H'00000277			2	
	通道 13	DEI13	160	H'00000280 ~ H'00000283	0 ~ 15(0)	IPR09(11 ~ 8)	1	
		HEI13	161	H'00000284 ~ H'00000287			2	
	通道 14	DEI14	164	H'00000290 ~ H'00000293	0 ~ 15(0)	IPR09(7 ~ 4)	1	
		HEI14	165	H'00000294 ~ H'00000297			2	
	通道 15	DEI15	168	H'000002A0 ~ H'000002A3	0 ~ 15(0)	IPR09(3 ~ 0)	1	
		HEI15	169	H'000002A4 ~ H'000002A7			2	
USB2.0 主机 / 功能模块	USBI	170	H'000002A8 ~ H'000002AB	0 ~ 15(0)	IPR10(15 ~ 12)	—		
视频显示控制器 4	VI_VSYNC	171	H'000002AC ~ H'000002AF	0 ~ 15(0)	IPR10(11 ~ 8)	1		
	LO_VSYNC	172	H'000002B0 ~ H'000002B3			2		
	VSYNCERR	173	H'000002B4 ~ H'000002B7			3		
	VLINE	174	H'000002B8 ~ H'000002BB	0 ~ 15(0)	IPR10(7 ~ 4)	1		
	VFIELD	175	H'000002BC ~ H'000002BF			2		
	VBUFERR1	176	H'000002C0 ~ H'000002C3	0 ~ 15(0)	IPR10(3 ~ 0)	1		
	VBUFERR2	177	H'000002C4 ~ H'000002C7			2		
	VBUFERR3	178	H'000002C8 ~ H'000002CB			3		
VBUFERR4	179	H'000002CC ~ H'000002CF	4					
失真校正引擎	IMRI	180	H'000002D0 ~ H'000002D3	0 ~ 15(0)	IPR11(15 ~ 12)	—		
JPEG 编解码器单元	JEDI	181	H'000002D4 ~ H'000002D7	0 ~ 15(0)	IPR11(11 ~ 8)	1		
	JDTI	182	H'000002D8 ~ H'000002DB			2		
显示输出比较单元	CMPI	183	H'000002DC ~ H'000002DF	0 ~ 15(0)	IPR11(7 ~ 4)	—		
OpenV G 的瑞萨图形处理器	INT3	184	H'000002E0 ~ H'000002E3	0 ~ 15(0)	IPR11(3 ~ 0)	1		
	INT2	185	H'000002E4 ~ H'000002E7			2		
	INT1	186	H'000002E8 ~ H'000002EB			3		
	INT0	187	H'000002EC ~ H'000002EF			4		
比较匹配定时器	通道 0	CMIO	188	H'000002F0 ~ H'000002F3	0 ~ 15(0)	IPR12(15 ~ 12)	—	
	通道 1	CMI1	189	H'000002F4 ~ H'000002F7	0 ~ 15(0)	IPR12(11 ~ 8)	—	
							↓ 低	

中断源		中断向量		中断优先级 (初始值)	对应的 IPR (位)	IPR 设定 单位内的 优先级	默认 优先级		
		向量	向量表地址偏移量						
总线状态控制器	CMI	190	H'000002F8 ~ H'000002FB	0 ~ 15(0)	IPR12(7 ~ 4)	—	高 ↑		
看门狗定时器	ITI	191	H'000002FC ~ H'000002FF	0 ~ 15(0)	IPR12(3 ~ 0)	—			
多功能定时器脉冲单元 2	通道 0	TGI0A	192	H'00000300 ~ H'00000303	0 ~ 15(0)	IPR13(15 ~ 12)	1		
		TGI0B	193	H'00000304 ~ H'00000307			2		
		TGI0C	194	H'00000308 ~ H'0000030B			3		
		TGI0D	195	H'0000030C ~ H'0000030F			4		
		TCI0V	196	H'00000310 ~ H'00000313			0 ~ 15(0)	IPR13(11 ~ 8)	1
		TGI0E	197	H'00000314 ~ H'00000317			2		
	通道 1	TGI0F	198	H'00000318 ~ H'0000031B	3				
		TGI1A	199	H'0000031C ~ H'0000031F	0 ~ 15(0)	IPR13(7 ~ 4)	1		
		TGI1B	200	H'00000320 ~ H'00000323	2				
		TCI1V	201	H'00000324 ~ H'00000327	0 ~ 15(0)	IPR13(3 ~ 0)	1		
	通道 2	TCI1U	202	H'00000328 ~ H'0000032B	2				
		TGI2A	203	H'0000032C ~ H'0000032F	0 ~ 15(0)	IPR14(15 ~ 12)	1		
		TGI2B	204	H'00000330 ~ H'00000333	2				
		TCI2V	205	H'00000334 ~ H'00000337	0 ~ 15(0)	IPR14(11 ~ 8)	1		
	通道 3	TCI2U	206	H'00000338 ~ H'0000033B	2				
		TGI3A	207	H'0000033C ~ H'0000033F	0 ~ 15(0)	IPR14(7 ~ 4)	1		
		TGI3B	208	H'00000340 ~ H'00000343	2				
		TGI3C	209	H'00000344 ~ H'00000347	3				
		TGI3D	210	H'00000348 ~ H'0000034B	4				
	通道 4	TCI3V	211	H'0000034C ~ H'0000034F	0 ~ 15(0)	IPR14(3 ~ 0)	—		
TGI4A		212	H'00000350 ~ H'00000353	0 ~ 15(0)	IPR15(15 ~ 12)	1			
TGI4B		213	H'00000354 ~ H'00000357	2					
TGI4C		214	H'00000358 ~ H'0000035B	3					
TGI4D		215	H'0000035C ~ H'0000035F	4					
马达控制 PWM 定时器	TCI4V	216	H'00000360 ~ H'00000363	0 ~ 15(0)	IPR15(11 ~ 8)	—			
	通道 1	217	H'00000364 ~ H'00000367	0 ~ 15(0)	IPR15(7 ~ 4)	—			
音频发生器	通道 2	218	H'00000368 ~ H'0000036B	0 ~ 15(0)	IPR15(3 ~ 0)	—			
	通道 0	219	H'0000036C ~ H'0000036F	0 ~ 15(0)	IPR16(15 ~ 12)	—			
	通道 1	220	H'00000370 ~ H'00000373	0 ~ 15(0)	IPR16(11 ~ 8)	—			
	通道 2	221	H'00000374 ~ H'00000377	0 ~ 15(0)	IPR16(7 ~ 4)	—			
A/D 转换器	通道 3	222	H'00000378 ~ H'0000037B	0 ~ 15(0)	IPR16(3 ~ 0)	—			
	ADI	223	H'0000037C ~ H'0000037F	0 ~ 15(0)	IPR17(15 ~ 12)	—			
串行音频接口	通道 0	SSIF0	224	H'00000380 ~ H'00000383	0 ~ 15(0)	IPR17(11 ~ 8)	1		
		SSIRXI0	225	H'00000384 ~ H'00000387			2		
		SSITXI0	226	H'00000388 ~ H'0000038B			3		
	通道 1	SSII1	227	H'0000038C ~ H'0000038F			0 ~ 15(0)	IPR17(7 ~ 4)	1
		SSIRT11	228	H'00000390 ~ H'00000393			2		
	通道 2	SSII2	229	H'00000394 ~ H'00000397			0 ~ 15(0)	IPR17(3 ~ 0)	1
	SSIRT12	230	H'00000398 ~ H'0000039B	2					

低
↓

中断源		中断向量		中断优先级 (初始值)	对应的 IPR (位)	IPR 设定 单位内的 优先级	默认 优先级	
		向量	向量表地址偏移量					
串行音频接口	通道 3	SSII3	231	H'0000039C ~ H'0000039F	0 ~ 15(0)	IPR18(15 ~ 12)	1	↑ 高
		SSIRTI3	232	H'000003A0 ~ H'000003A3			2	
	通道 4	SSII4	233	H'000003A4 ~ H'000003A7	0 ~ 15(0)	IPR18(11 ~ 8)	1	
		SSIRTI4	234	H'000003A8 ~ H'000003AB			2	
	通道 5	SSII5	235	H'000003AC ~ H'000003AF	0 ~ 15(0)	IPR18(7 ~ 4)	1	
		SSIRTI5	236	H'000003B0 ~ H'000003B3			2	
瑞萨 SPDIF 接口	SPDIFI		237	H'000003B4 ~ H'000003B7	0 ~ 15(0)	IPR18(3 ~ 0)	—	
I ² C 总线接口 3	通道 0	STPI0	238	H'000003B8 ~ H'000003BB	0 ~ 15(0)	IPR19(15 ~ 12)	1	
		NAKI0	239	H'000003BC ~ H'000003BF			2	
		RXI0	240	H'000003C0 ~ H'000003C3			3	
		TXI0	241	H'000003C4 ~ H'000003C7			4	
		TEI0	242	H'000003C8 ~ H'000003CB			5	
	通道 1	STPI1	243	H'000003CC ~ H'000003CF	0 ~ 15(0)	IPR19(11 ~ 8)	1	
		NAKI1	244	H'000003D0 ~ H'000003D3			2	
		RXI1	245	H'000003D4 ~ H'000003D7			3	
		TXI1	246	H'000003D8 ~ H'000003DB			4	
		TEI1	247	H'000003DC ~ H'000003DF			5	
	通道 2	STPI2	248	H'000003E0 ~ H'000003E3	0 ~ 15(0)	IPR19(7 ~ 4)	1	
		NAKI2	249	H'000003E4 ~ H'000003E7			2	
		RXI2	250	H'000003E8 ~ H'000003EB			3	
		TXI2	251	H'000003EC ~ H'000003EF			4	
		TEI2	252	H'000003F0 ~ H'000003F3			5	
	通道 3	STPI3	253	H'000003F4 ~ H'000003F7	0 ~ 15(0)	IPR19(3 ~ 0)	1	
		NAKI3	254	H'000003F8 ~ H'000003FB			2	
		RXI3	255	H'000003FC ~ H'000003FF			3	
		TXI3	256	H'00000400 ~ H'00000403			4	
		TEI3	257	H'00000404 ~ H'00000407			5	
	FIFO 内置型串行通信接口	通道 0	BRI0	258	H'00000408 ~ H'0000040B	0 ~ 15(0)	IPR20(15 ~ 12)	1
ERI0			259	H'0000040C ~ H'0000040F	2			
RXI0			260	H'00000410 ~ H'00000413	3			
TXI0			261	H'00000414 ~ H'00000417	4			
通道 1		BRI1	262	H'00000418 ~ H'0000041B	0 ~ 15(0)	IPR20(11 ~ 8)	1	
		ERI1	263	H'0000041C ~ H'0000041F			2	
		RXI1	264	H'00000420 ~ H'00000423			3	
		TXI1	265	H'00000424 ~ H'00000427			4	
通道 2		BRI2	266	H'00000428 ~ H'0000042B	0 ~ 15(0)	IPR20(7 ~ 4)	1	
		ERI2	267	H'0000042C ~ H'0000042F			2	
		RXI2	268	H'00000430 ~ H'00000433			3	
		TXI2	269	H'00000434 ~ H'00000437			4	
通道 3		BRI3	270	H'00000438 ~ H'0000043B	0 ~ 15(0)	IPR20(3 ~ 0)	1	
		ERI3	271	H'0000043C ~ H'0000043F			2	
		RXI3	272	H'00000440 ~ H'00000443			3	
		TXI3	273	H'00000444 ~ H'00000447			4	

中断源		中断向量		中断优先级 (初始值)	对应的 IPR (位)	IPR 设定 单位内的 优先级	默认 优先级	
		向量	向量表地址偏移量					
FIFO 内 置型串 行通信 接口	通道 4	BRI4	274	H'00000448 ~ H'0000044B	0 ~ 15(0)	IPR21(15 ~ 12)	1	↑ 高
		ERI4	275	H'0000044C ~ H'0000044F			2	
		RXI4	276	H'00000450 ~ H'00000453			3	
		TXI4	277	H'00000454 ~ H'00000457			4	
	通道 5	BRI5	278	H'00000458 ~ H'0000045B	0 ~ 15(0)	IPR21(11 ~ 8)	1	
		ERI5	279	H'0000045C ~ H'0000045F			2	
		RXI5	280	H'00000460 ~ H'00000463			3	
		TXI5	281	H'00000464 ~ H'00000467			4	
	通道 6	BRI6	282	H'00000468 ~ H'0000046B	0 ~ 15(0)	IPR21(7 ~ 4)	1	
		ERI6	283	H'0000046C ~ H'0000046F			2	
		RXI6	284	H'00000470 ~ H'00000473			3	
		TXI6	285	H'00000474 ~ H'00000477			4	
	通道 7	BRI7	286	H'00000478 ~ H'0000047B	0 ~ 15(0)	IPR21(3 ~ 0)	1	
		ERI7	287	H'0000047C ~ H'0000047F			2	
		RXI7	288	H'00000480 ~ H'00000483			3	
		TXI7	289	H'00000484 ~ H'00000487			4	
带 FIFO 的时钟 同步串 行 I/O	SIOFI	290	H'00000488 ~ H'0000048B	0 ~ 15(0)	IPR22(15 ~ 12)	—		
控制器 局域网	通道 0	ERS0	291	H'0000048C ~ H'0000048F	0 ~ 15(0)	IPR22(11 ~ 8)	1	
		OVR0	292	H'00000490 ~ H'00000493			2	
		RM00	293	H'00000494 ~ H'00000497			3	
		RM10	294	H'00000498 ~ H'0000049B			4	
		SLE0	295	H'0000049C ~ H'0000049F			5	
	通道 1	ERS1	296	H'000004A0 ~ H'000004A3	0 ~ 15(0)	IPR22(7 ~ 4)	1	
		OVR1	297	H'000004A4 ~ H'000004A7			2	
		RM01	298	H'000004A8 ~ H'000004AB			3	
		RM11	299	H'000004AC ~ H'000004AF			4	
		SLE1	300	H'000004B0 ~ H'000004B3			5	
控制器 局域网	通道 2	ERS2	301	H'000004B4 ~ H'000004B7	0 ~ 15(0)	IPR22(3 ~ 0)	1	
		OVR2	302	H'000004B8 ~ H'000004BB			2	
		RM02	303	H'000004BC ~ H'000004BF			3	
		RM12	304	H'000004C0 ~ H'000004C3			4	
		SLE2	305	H'000004C4 ~ H'000004C7			5	
瑞萨串 行外围 接口	通道 0	SPEI0	306	H'000004C8 ~ H'000004CB	0 ~ 15(0)	IPR23(15 ~ 12)	1	
		SPRI0	307	H'000004CC ~ H'000004CF			2	
		SPTI0	308	H'000004D0 ~ H'000004D3			3	
	通道 1	SPEI1	309	H'000004D4 ~ H'000004D7	0 ~ 15(0)	IPR23(11 ~ 8)	1	
		SPRI1	310	H'000004D8 ~ H'000004DB			2	
		SPTI1	311	H'000004DC ~ H'000004DF			3	
瑞萨四 路串行 外围接 口	通道 0	SPEI0	312	H'000004E0 ~ H'000004E3	0 ~ 15(0)	IPR23(7 ~ 4)	1	
		SPRI0	313	H'000004E4 ~ H'000004E7			2	
		SPTI0	314	H'000004E8 ~ H'000004EB			3	
	通道 1	SPEI1	315	H'000004EC ~ H'000004EF	0 ~ 15(0)	IPR23(3 ~ 0)	1	
		SPRI1	316	H'000004F0 ~ H'000004F3			2	
		SPTI1	317	H'000004F4 ~ H'000004F7			3	
							↓ 低	

中断源		中断向量		中断优先级 (初始值)	对应的 IPR (位)	IPR 设定 单位内的 优先级	默认 优先级	
		向量	向量表地址偏移量					
IEBus™ 控制器	IEB	318	H'000004F8 ~ H'000004FB	0 ~ 15(0)	IPR24(15 ~ 12)	—	高	
CD-ROM 解码器	ISY	319	H'000004FC ~ H'000004FF	0 ~ 15(0)	IPR24(11 ~ 8)	1	↑	
	IERR	320	H'00000500 ~ H'00000503			2		
	IARG	321	H'00000504 ~ H'00000507			3		
	ISEC	322	H'00000508 ~ H'0000050B			4		
	IBUF	323	H'0000050C ~ H'0000050F			5		
	IREADY	324	H'00000510 ~ H'00000513			6		
NAND 闪存控 制器	FLSTEI	325	H'00000514 ~ H'00000517	0 ~ 15(0)	IPR24(7 ~ 4)	1		
	FLTENDI	326	H'00000518 ~ H'0000051B			2		
	FLTREQ0I	327	H'0000051C ~ H'0000051F			3		
	FLTREQ1I	328	H'00000520 ~ H'00000523			4		
MMC 主 机接口	MMC0	329	H'00000524 ~ H'00000527	0 ~ 15(0)	IPR24(3 ~ 0)	1		
	MMC1	330	H'00000528 ~ H'0000052B			2		
	MMC2	331	H'0000052C ~ H'0000052F			3		
SD 主机 接口	通道 0	SDHI0_3	332	H'00000530 ~ H'00000533	0 ~ 15(0)	IPR25(15 ~ 12)	1	↑
		SDHI0_0	333	H'00000534 ~ H'00000537			2	
		SDHI0_1	334	H'00000538 ~ H'0000053B			3	
	通道 1	SDHI1_3	335	H'0000053C ~ H'0000053F	0 ~ 15(0)	IPR25(11 ~ 8)	1	
		SDHI1_0	336	H'00000540 ~ H'00000543			2	
		SDHI1_1	337	H'00000544 ~ H'00000547			3	
实时时 钟	ARM	338	H'00000548 ~ H'0000054B	0 ~ 15(0)	IPR25(7 ~ 4)	1		
	PRD	339	H'0000054C ~ H'0000054F			2		
	CUP	340	H'00000550 ~ H'00000553			3		
采样率 转换器	通道 0	OVF0	341	H'00000554 ~ H'00000557	0 ~ 15(0)	IPR26(15 ~ 12)	1	
		UDF0	342	H'00000558 ~ H'0000055B			2	
		CEF0	343	H'0000055C ~ H'0000055F			3	
		ODFI0	344	H'00000560 ~ H'00000563			4	
		IDEI0	345	H'00000564 ~ H'00000567			5	
采样率 转换器	通道 1	OVF1	346	H'00000568 ~ H'0000056B	0 ~ 15(0)	IPR26(11 ~ 8)	1	↓
		UDF1	347	H'0000056C ~ H'0000056F			2	
		CEF1	348	H'00000570 ~ H'00000573			3	
		ODFI1	349	H'00000574 ~ H'00000577			4	
		IDEI1	350	H'00000578 ~ H'0000057B			5	
	通道 2	OVF2	351	H'0000057C ~ H'0000057F	0 ~ 15(0)	IPR26(7 ~ 4)	1	
		UDF2	352	H'00000580 ~ H'00000583			2	
		CEF2	353	H'00000584 ~ H'00000587			3	
		ODFI2	354	H'00000588 ~ H'0000058B			4	
		IDEI2	355	H'0000058C ~ H'0000058F			5	

7.6 运行说明

7.6.1 中断运行的流程

以下说明发生中断时的运行流程，运行流程如图 7.2 所示。

1. 将各中断请求源的中断请求信号送给中断控制器。
2. 中断控制器按照中断优先级设定寄存器 01、02、05～26 (IPR01、IPR02、IPR05～IPR26)，从送来的中断请求中选择优先级最高的中断，并且忽视*优先级低的中断。此时，如果发生多个被设定为相同优先级的中断或者多个同一 IPR 设定内的中断，就根据表 7.4 所示的默认优先级和 IPR 设定单位内的优先级，选择优先级最高的中断。
3. 将中断控制器选择的中断优先级和 CPU 状态寄存器 (SR) 的中断屏蔽级位 (I3～I0) 进行比较。忽视相同或者低于 I3～I0 位所设优先级的中断，而只接受高于 I3～I0 位的优先级的中断，并且将中断请求信号送给 CPU。
4. 在 CPU 对要执行的指令进行解码时，检测到从中断控制器送来的中断请求，并且将该指令的执行替换为中断异常处理 (参照图 7.4)。
5. 从对应接受中断的异常处理向量表中取中断异常服务程序的起始地址。
6. 将状态寄存器 (SR) 压栈，并且将接受中断的优先级写到 SR 的 I3～I0 位。
7. 将程序计数器 (PC) 压栈。
8. 转移到取得的中断异常服务程序的起始地址，开始执行程序。此时的转移不是延迟转移。

【注】 必须在中断处理程序中清除中断源标志。从清除中断源标志到实际取消给 CPU 的中断源的期间，需要表 7.5 记载的“从发生中断请求，到通过中断控制器判断优先级并且和 SR 屏蔽位进行比较，然后将中断请求信号送给 CPU 为止的时间”。因此，为了避免再次误接受应该被清除的中断源，必须在清除后读中断源标志，然后执行 RTE 指令。

- * 设定为边沿检测的中断请求被保留到接受为止。但是，如果是 IRQ 中断，就能通过存取 IRQ 中断请求寄存器 (IRQRR) 取消中断。详细内容请参照“7.4.4 IRQ 中断”。
另外，通过上电复位清除被保留的边沿检测中断。

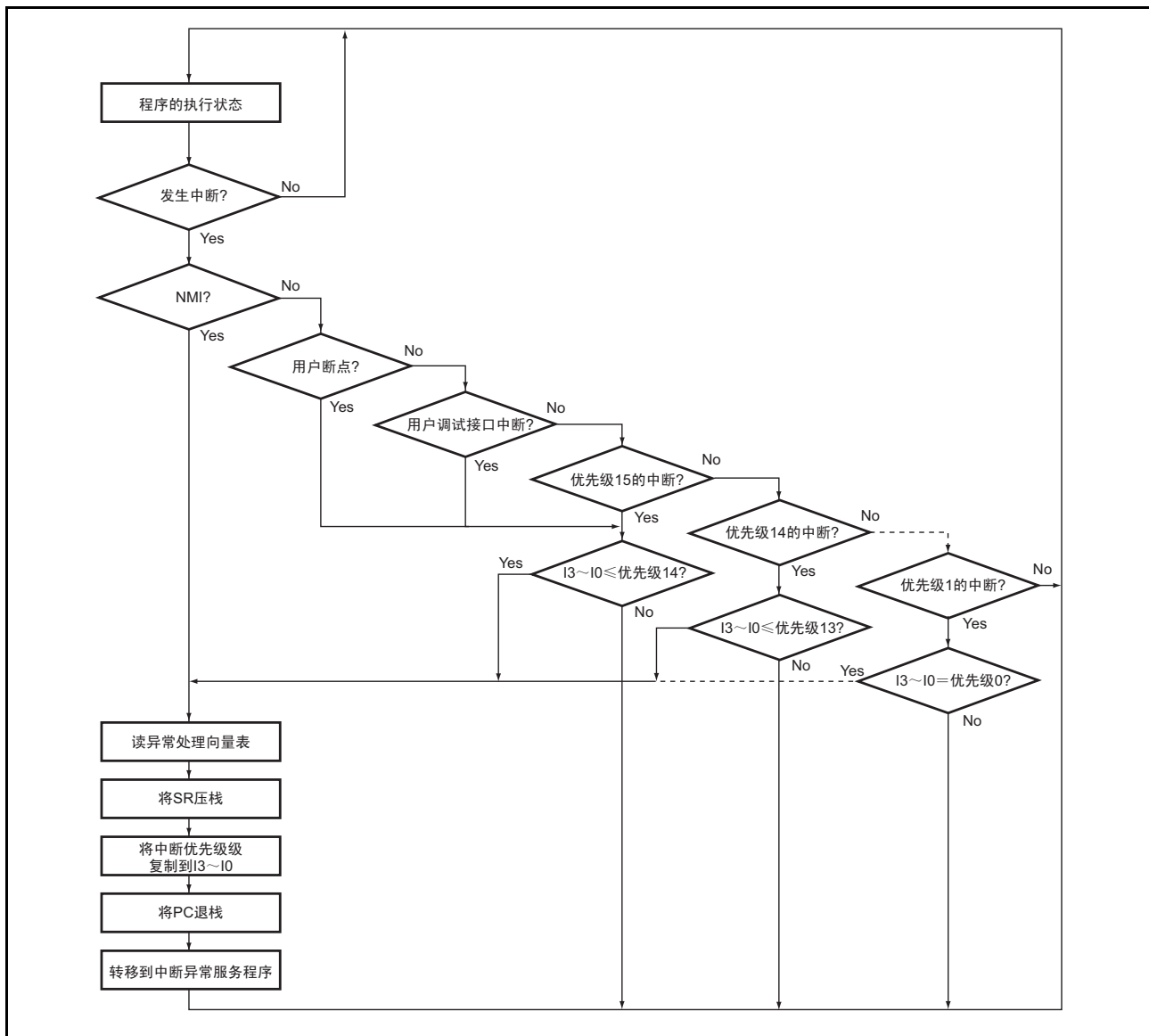


图 7.2 中断运行流程

7.6.2 中断异常处理结束后的堆栈状态

中断异常处理结束后的堆栈状态如图 7.3 所示。

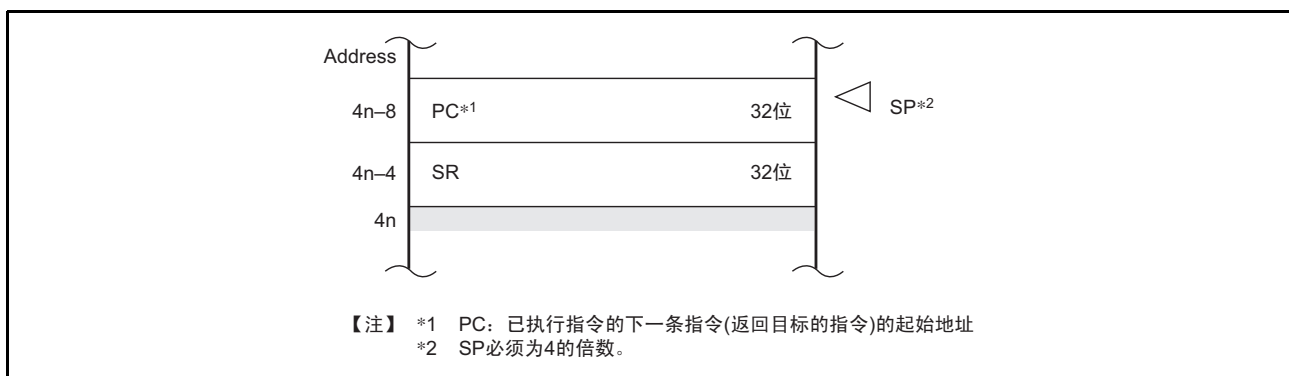


图 7.3 中断异常处理结束后的堆栈状态

7.7 中断响应时间

从发生中断请求到进行中断异常处理以及开始取中断异常服务程序第1条指令为止的时间（中断响应时间）如表 7.5 所示。在无寄存器组时或者在有寄存器组并且寄存器组不发生上溢时或者在有寄存器组并且寄存器组发生上溢时，中断处理的运行不同。无寄存器组时的流水线运行例子如图 7.4 和图 7.5 所示，有寄存器组并且寄存器组不发生上溢时的流水线运行例子如图 7.6 和图 7.7 所示，有寄存器组并且寄存器组发生上溢时的流水线运行例子如图 7.8 和图 7.9 所示。

表 7.5 中断响应时间

项目			状态数								备注		
			NMI	用户断点	用户调试接口	IRQ、PINT	USB2.0 主机 / 功能模块	FIFO 内置型串行通信接口	外围模块群 1 *3	外围模块群 2 *4			
从发生中断请求，到通过中断控制器判断优先级并且和SR屏蔽位进行比较，然后将中断请求信号送给 CPU 为止的时间			2lcyc+2P1cyc+1P0cyc	3lcyc	2lcyc+1P0cyc	2lcyc+3P1cyc+1P0cyc	2lcyc+2Bcyc+2P1cyc	2lcyc+4Bcyc+1P1cyc	2lcyc+4Bcyc+1P0cyc	2lcyc+2Bcyc			
从给 CPU 输入中断请求信号，到结束正在执行的响应顺序后开始中断异常处理，并且取中断异常服务程序第1条指令为止的时间	无寄存器组	最小值	3lcyc+m1+m2								最小值是指中断等待时间为0的情况。最大值是指在中断异常处理中发生更高优先级中断请求的情况。		
		最大值	4lcyc+2(m1+m2)+m3										
	有寄存器组，寄存器组不发生上溢。	最小值	—		3lcyc+m1+m2							最小值是指中断等待时间为0的情况。最大值是指在 RESBANK 指令执行过程中发生中断请求的情况。	
		最大值	—		12lcyc+m1+m2								
	有寄存器组，寄存器组发生上溢。	最小值	—		3lcyc+m1+m2								最小值是指中断等待时间为0的情况。最大值是指在 RESBANK 指令执行过程中发生中断请求的情况。
		最大值	—		3lcyc+m1+m2+19(m4)								
响应时间	无寄存器组。	最小值	5lcyc+2P1cyc+1P0cyc+m1+m2	6lcyc+m1+m2	5lcyc+1P0cyc+m1+m2	5lcyc+3P1cyc+1P0cyc+m1+m2	5lcyc+2Bcyc+2P1cyc+m1+m2	5lcyc+4Bcyc+1P1cyc+m1+m2	5lcyc+4Bcyc+1P0cyc+m1+m2	5lcyc+2Bcyc+m1+m2	以 266MHz 运行时 *1*2: 0.029μs ~ 0.101μs		
		最大值	6lcyc+2P1cyc+1P0cyc+2(m1+m2)+m3	7lcyc+2(m1+m2)+m3	6lcyc+1P0cyc+2(m1+m2)+m3	6lcyc+3P1cyc+1P0cyc+2(m1+m2)+m3	6lcyc+2Bcyc+2P1cyc+2(m1+m2)+m3	6lcyc+4Bcyc+1P1cyc+2(m1+m2)+m3	6lcyc+4Bcyc+1P0cyc+2(m1+m2)+m3	6lcyc+2Bcyc+2(m1+m2)+m3			

项目			状态数							备注
			NMI	用户断点	用户调试接口	IRQ、PINT	USB2.0 主机 / 功能模块	FIFO 内置型串行通信接口	外围模块群 1 *3	
响应时间	有寄存器组，寄存器组不发生上溢。	最小值	—	5lcyc+ 1P0cyc+ m1+m2	5lcyc+ 3P1cyc+ 1P0cyc+ m1+m2	5lcyc+ 2Bcyc+ 2P1cyc+ m1+m2	5lcyc+ 4Bcyc+ 1P1cyc+ m1+m2	5lcyc+ 4Bcyc+ 1P0cyc+ m1+m2	5lcyc+ 2Bcyc+ m1+m2	以 266MHz 运行时 *1*2: 0.041μs ~ 0.101μs
		最大值	—	14lcyc+ 1P0cyc+ m1+m2	14lcyc+ 3P1cyc+ 1P0cyc+ m1+m2	14lcyc+ 2Bcyc+ 2P1cyc+ m1+m2	14lcyc+ 4Bcyc+ 1P1cyc+ m1+m2	14lcyc+ 4Bcyc+ 1P0cyc+ m1+m2	14lcyc+ 2Bcyc+ m1+m2	以 266MHz 运行时 *1*2: 0.075μs ~ 0.135μs
	有寄存器组，寄存器组发生上溢。	最小值	—	5lcyc+ 1P0cyc+ m1+m2	5lcyc+ 3P1cyc+ 1P0cyc+ m1+m2	5lcyc+ 2Bcyc+ 2P1cyc+ m1+m2	5lcyc+ 4Bcyc+ 1P1cyc+ m1+m2	5lcyc+ 4Bcyc+ 1P0cyc+ m1+m2	5lcyc+ 2Bcyc+ m1+m2	以 266MHz 运行时 *1*2: 0.041μs ~ 0.102μs
		最大值	—	5lcyc+ 1P0cyc+ m1+m2+ 19(m4)	5lcyc+ 3P1cyc+ 1P0cyc+ m1+m2+ 19(m4)	5lcyc+ 2Bcyc+ 2P1cyc+ m1+m2+ 19(m4)	5lcyc+ 4Bcyc+ 1P1cyc+ m1+m2+ 19(m4)	5lcyc+ 4Bcyc+ 1P0cyc+ m1+m2+ 19(m4)	5lcyc+ 2Bcyc+ m1+m2+ 19(m4)	以 266MHz 运行时 *1*2: 0.112μs ~ 0.173μs

【注】 m1 ~ m4 是存取存储器时需要的状态数，如下所示：

m1: 读向量地址（读长字）。

m2: 将 SR 压栈（写长字）。

m3: 将 PC 压栈（写长字）。

m4: 存储体的对象寄存器（R0 ~ R14、GBR、MACH、MACL、PR）的退栈

*1 : 这是 m1=m2=m3=m4=1lcyc 的情况。

*2 : 这是 (f_{clk}、B_{clk}、P1_{clk}、P0_{clk})=(266.67MHz、133.33MHz、66.67MHz、33.33MHz) 的情况。

*3 : 比较匹配定时器、马达控制 PWM 定时器、多功能定时器脉冲单元 2、控制器局域网、IEBus™ 控制器、音频发生器

*4 : USB2.0 主机 / 功能模块、FIFO 内置型串行通信接口、外围模块群 1 以外的外围模块

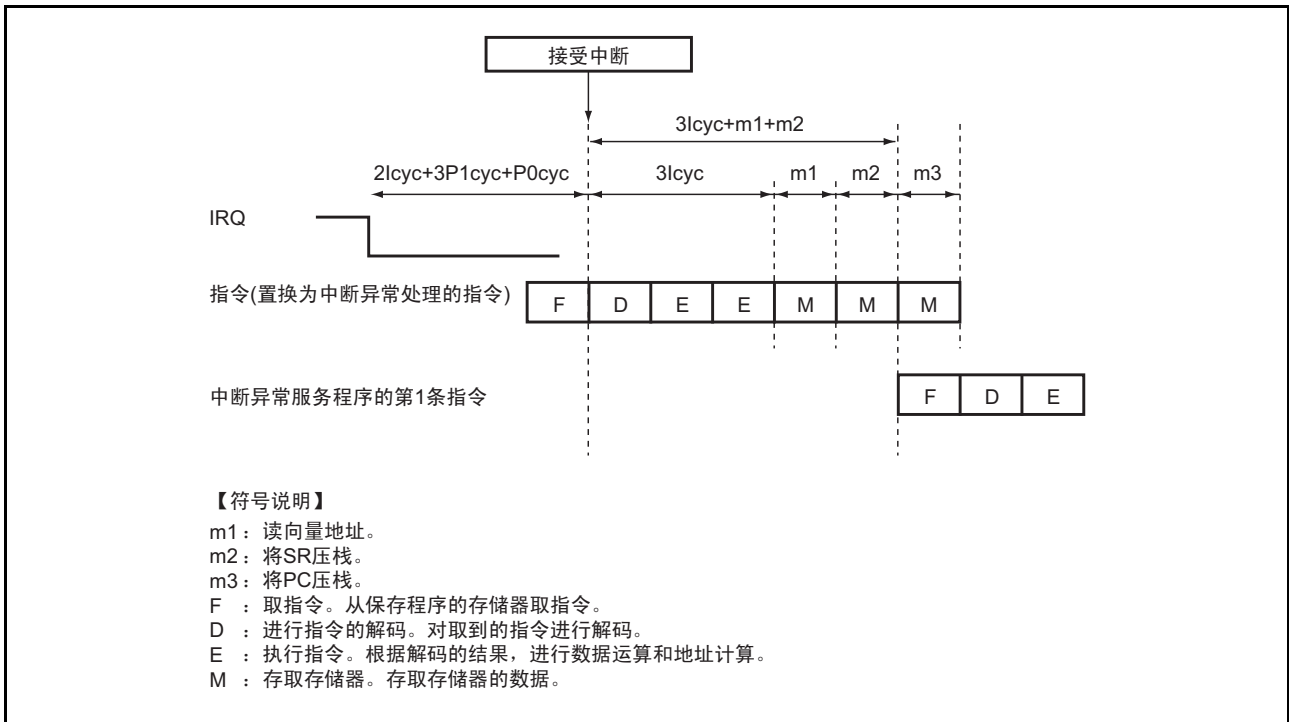


图 7.4 接受 IRQ 中断时的流水线运行例子（无寄存器组）

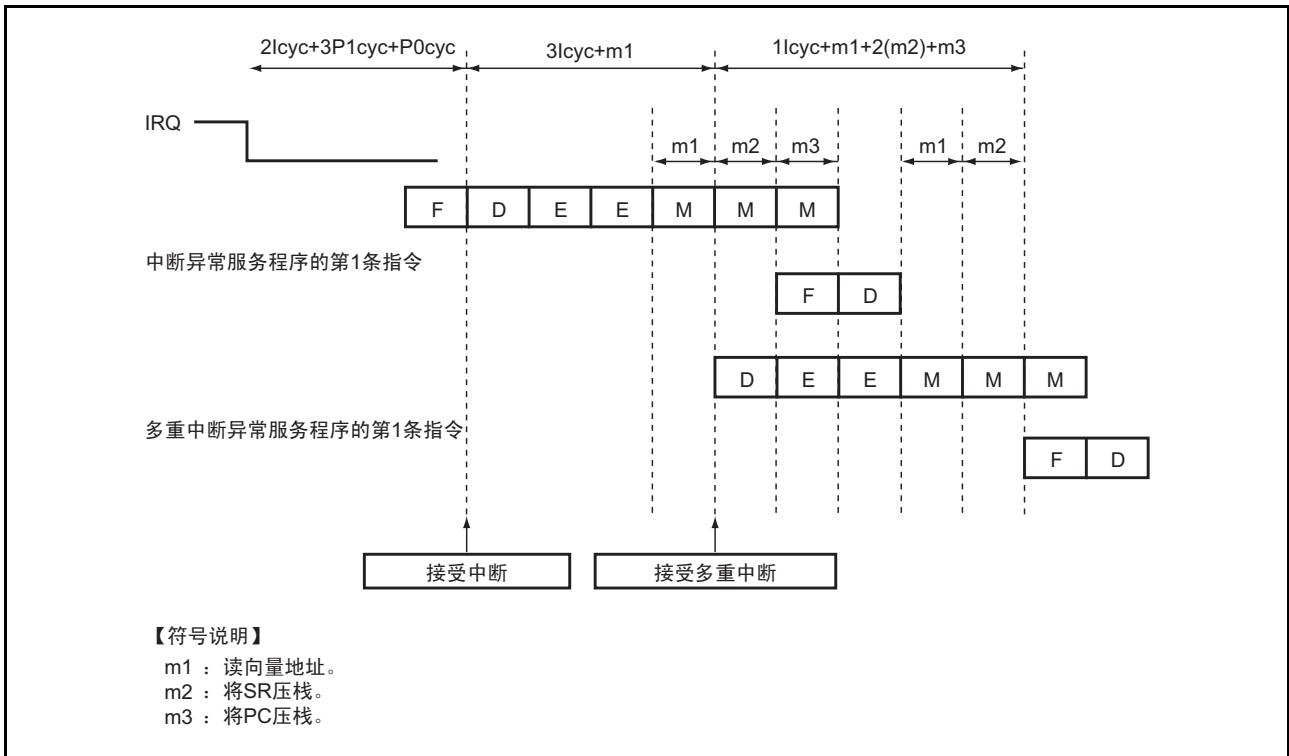


图 7.5 多重中断时的流水线运行例子（无寄存器组）

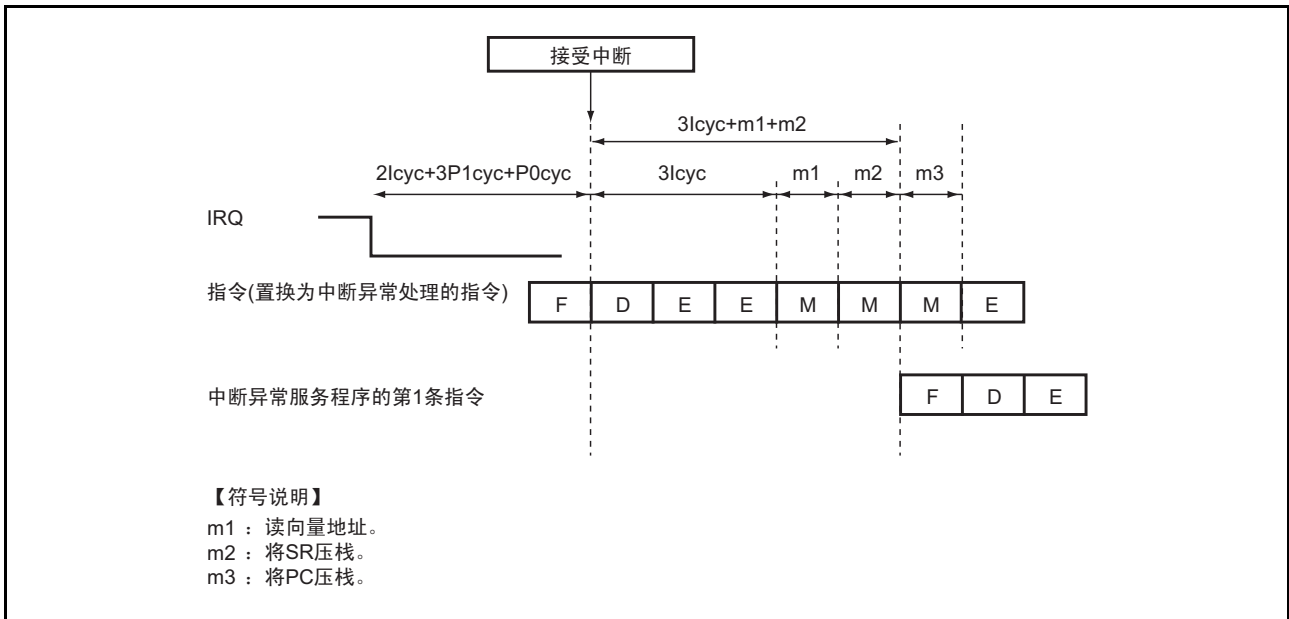


图 7.6 接受 IRQ 中断时的流水线运行例子（有寄存器组，寄存器组不发生上溢）

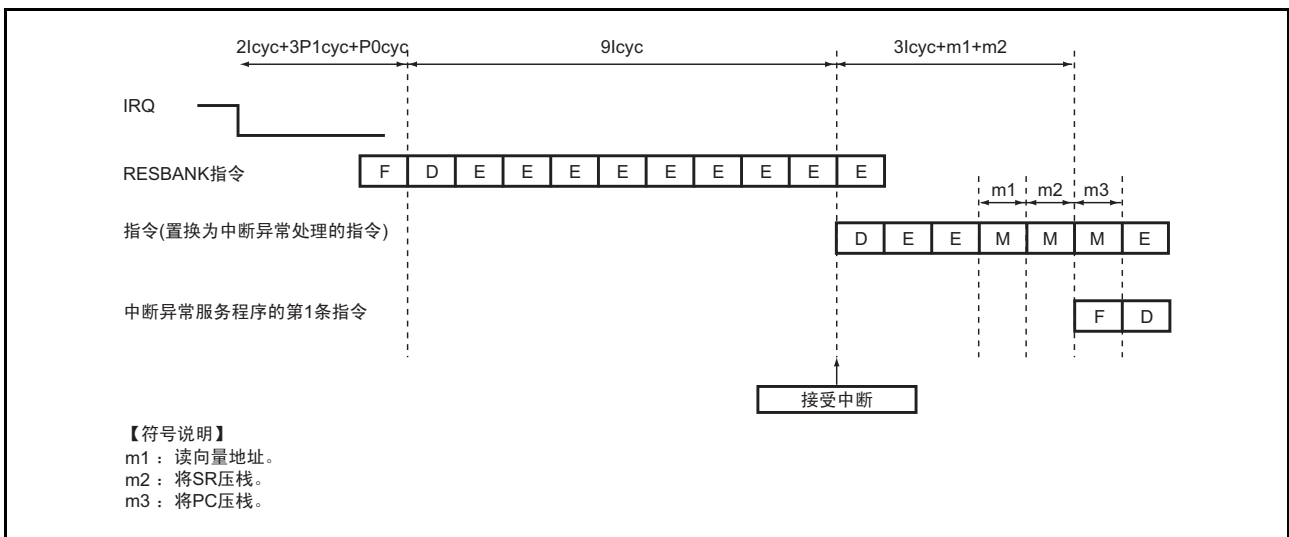


图 7.7 在执行 RESBANK 指令的过程中接受中断时的流水线运行例子（有寄存器组，寄存器组不发生上溢）

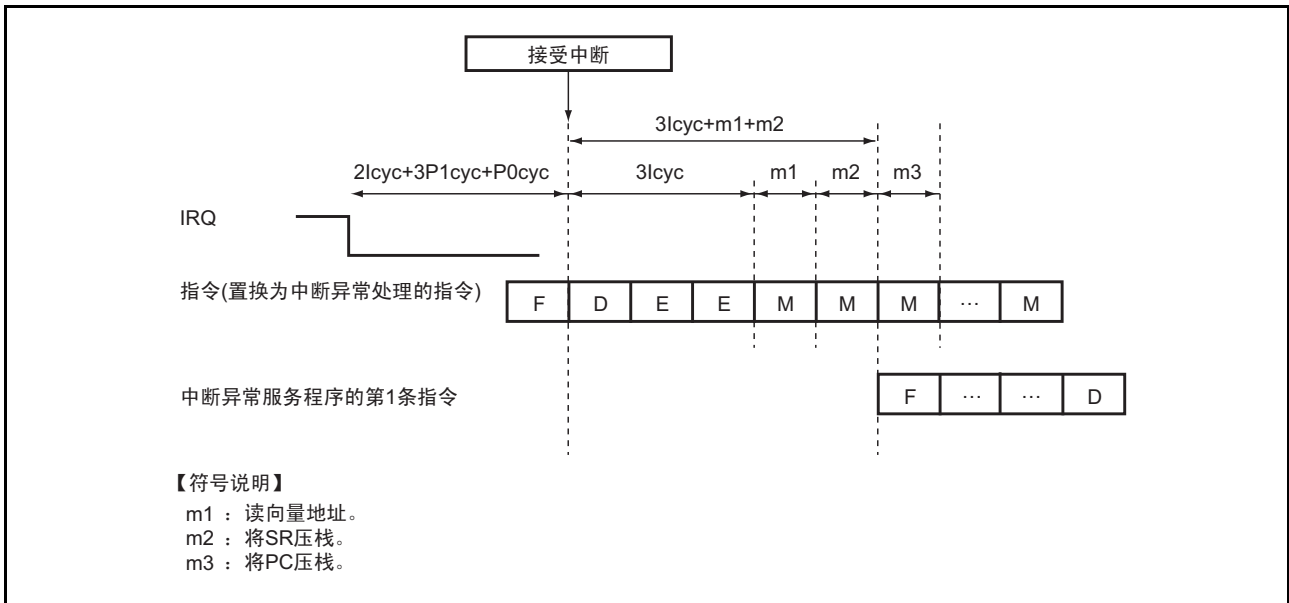


图 7.8 接受 IRQ 中断时的流水线运行例子（有寄存器组，寄存器组发生上溢）

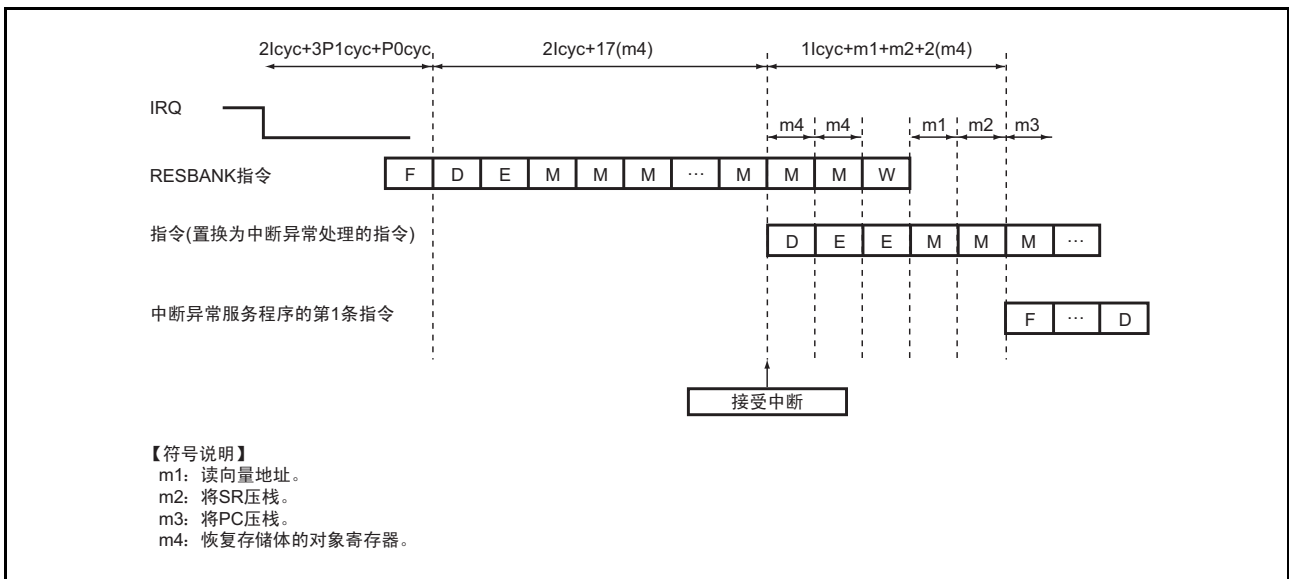


图 7.9 在执行 RESBANK 指令的过程中接受中断时的流水线运行例子（有寄存器组，寄存器组发生上溢）

7.8 寄存器组

本 LSI 内置 15 个寄存器组，在中断处理时高速进行寄存器的保存和恢复。寄存器组的结构如图 7.10 所示。

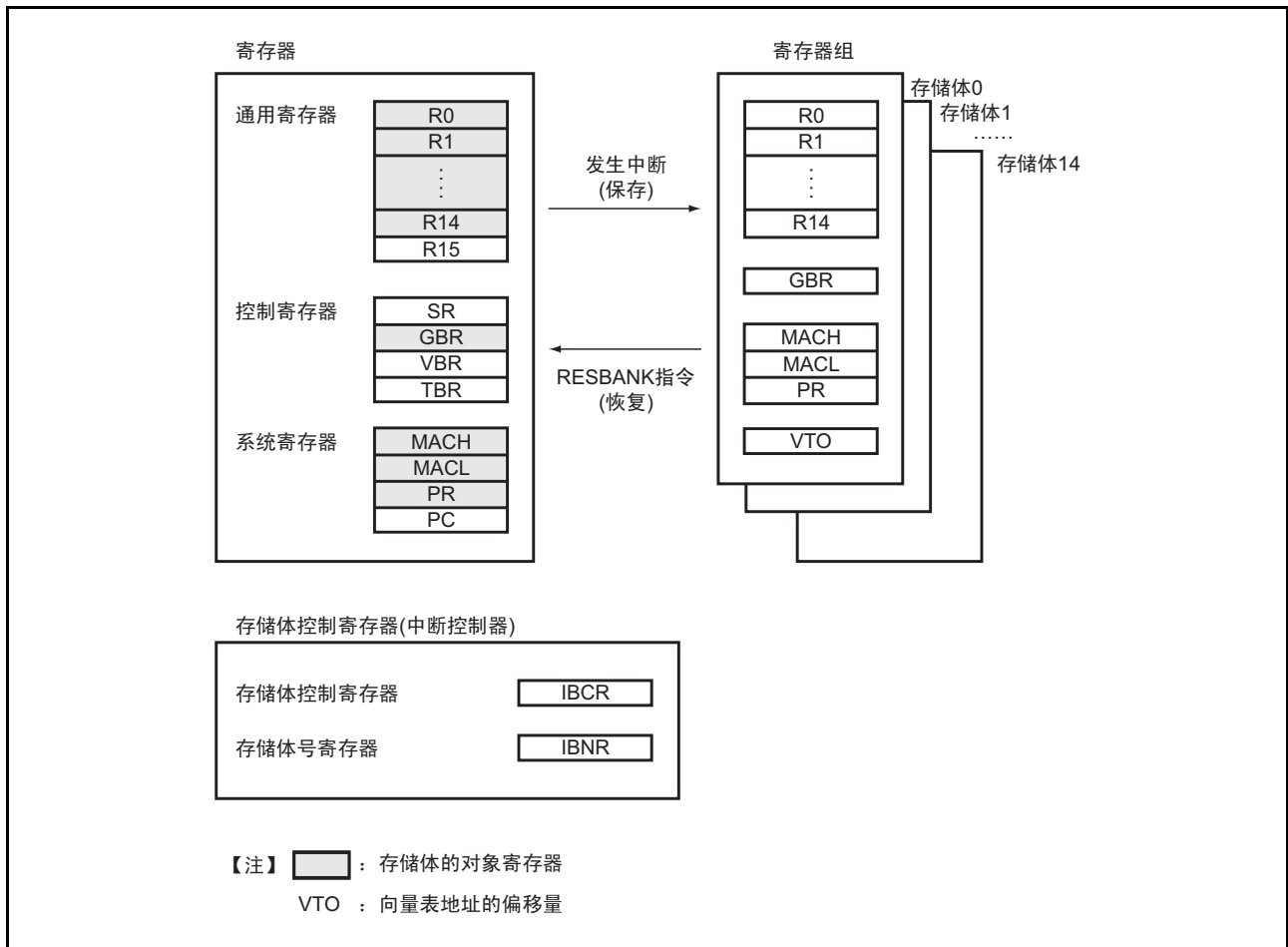


图 7.10 寄存器组结构的概要

7.8.1 存储体的对象寄存器和输入 / 输出方式

(1) 存储体的对象寄存器

存储体的对象寄存器是通用寄存器 (R0 ~ R14)、全局基址寄存器 (GBR)、乘加寄存器 (MACH 和 MACL)、过程寄存器 (PR) 和向量表地址偏移量。

(2) 存储体的输入 / 输出方式

寄存器组有 15 个存储体 (存储体 0 ~ 存储体 14)。寄存器组为先进后出 (FILO) 式的堆栈，从存储体 0 开始按顺序保存，从最后保存的存储体开始按顺序恢复。

7.8.2 存储体的保存和恢复

(1) 存储体的保存

- 保存到寄存器组的运行如图 7.11 所示。当发生中断并且允许 CPU 所接受的中断使用寄存器组时，运行如下：
- (a) 假设发生中断前的存储体号寄存器（IBNR）的存储体号位（BN）的值为 i 。
 - (b) 将寄存器 R0 ~ R14、GBR、MACH、MACL、PR 和被接受的中断向量地址偏移量（VTO）保存到 BN 所示的存储体 i 。
 - (c) 将 BN 的值加 1。

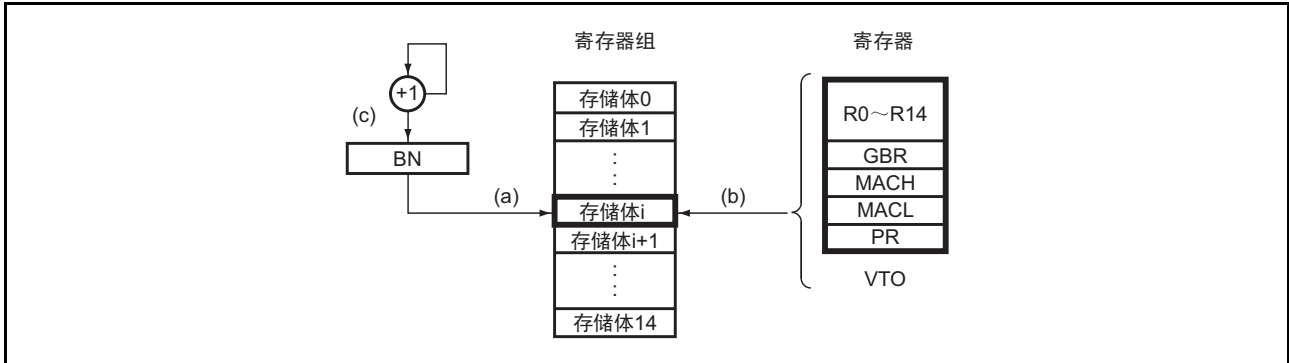


图 7.11 存储体的保存

寄存器组的保存时序如图 7.12 所示。在开始中断异常处理到开始取中断异常服务程序第 1 条指令的期间，保存到寄存器组。

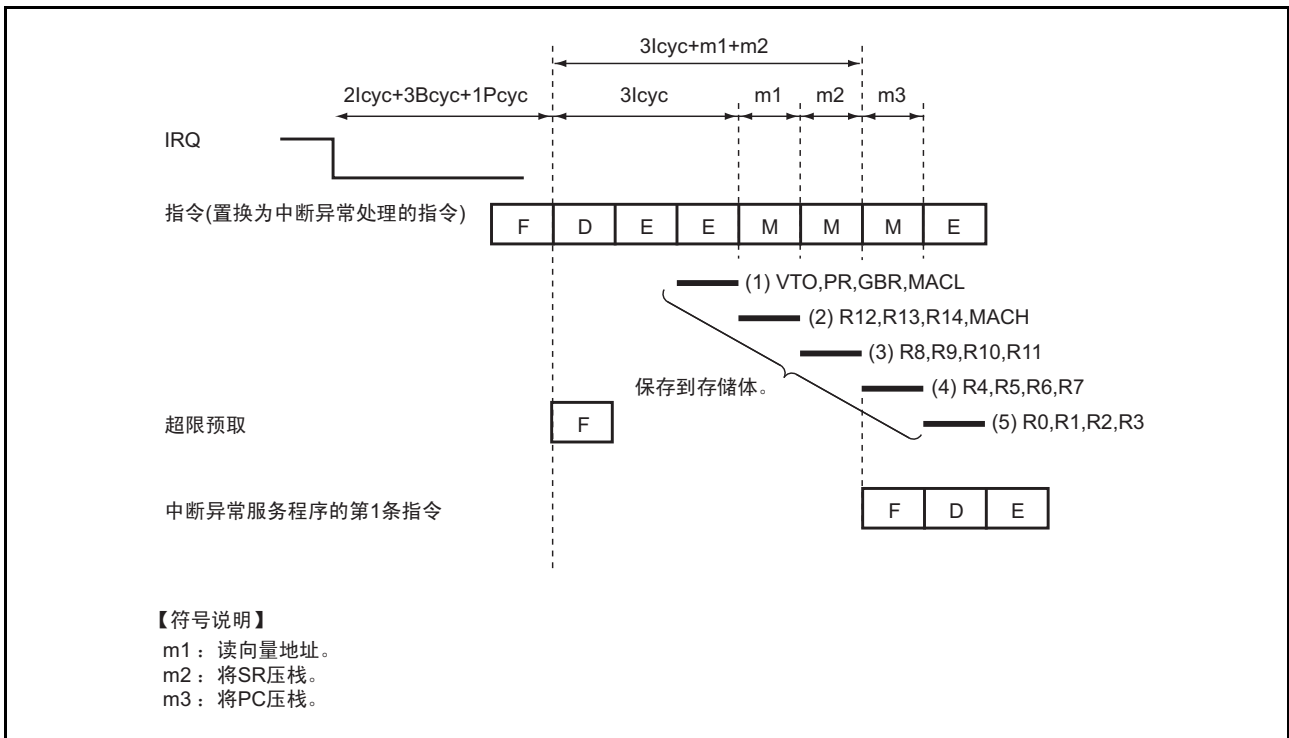


图 7.12 存储体的保存时序

(2) 存储体的恢复

要恢复保存到存储体的数据时，使用存储体恢复指令 **RESBANK**。必须在中断异常服务程序的最后，通过 **RESBANK** 指令进行存储体的恢复，然后用 **RTE** 指令从中断异常服务程序返回。

7.8.3 保存到全部存储体的状态下的保存和恢复

如果在保存到寄存器组的全部存储体的状态下发生中断并且允许 CPU 所接受的中断使用寄存器组，就在存储体号寄存器（**IBNR**）的 **BOVE** 位为“0”时，代替寄存器组自动保存到堆栈。当 **IBNR** 的 **BOVE** 位被置为“1”时，发生寄存器组上溢异常而不保存到堆栈。

压栈和退栈如下：

(1) 压栈

1. 在中断异常处理时，将状态寄存器（**SR**）和程序计数器（**PC**）压栈。
2. 将存储体对象寄存器（**R0**～**R14**、**GBR**、**MACH**、**MACL**、**PR**）压栈。被压栈的寄存器顺序为：**MACL**、**MACH**、**GBR**、**PR**、**R14**、**R13**、……、**R1**、**R0**
3. 将 **SR** 的寄存器组上溢位（**BO**）置“1”。
4. 存储体号寄存器（**IBNR**）的存储体号位（**BN**）不变而保持为最大值 15。

(2) 退栈

在将 **SR** 的寄存器组上溢位（**BO**）置“1”的状态下执行存储体恢复指令 **RESBANK** 时，运行如下：

1. 将存储体对象寄存器（**R0**～**R14**、**GBR**、**MACH**、**MACL**、**PR**）退栈。被退栈的寄存器顺序为：**R0**、**R1**、……、**R13**、**R14**、**PR**、**GBR**、**MACH**、**MACL**。
2. 存储体号寄存器（**IBNR**）的存储体号位（**BN**）不变而保持为最大值 15。

7.8.4 寄存器组异常

寄存器组异常（寄存器组错误）有寄存器组上溢和寄存器组下溢共 2 种。

(1) 寄存器组上溢

如果在保存到寄存器组的全部存储体的状态下发生中断并且允许 CPU 所接受的中断使用寄存器组，就在存储体号寄存器（**IBNR**）的 **BOVE** 位被置为“1”时发生寄存器组上溢异常。此时，存储体号寄存器（**IBNR**）的存储体号位（**BN**）不变而保持为存储体数 15，不保存到寄存器组。

(2) 寄存器组下溢

在完全没有保存到寄存器组的状态下执行寄存器组恢复指令时发生寄存器组下溢异常。此时 **R0**～**R14**、**GBR**、**MACH**、**MACL**、**PR** 的值不变，并且存储体号寄存器（**IBNR**）的存储体号位（**BN**）不变而保持为“0”。

7.8.5 寄存器组的错误异常处理

如果发生寄存器组错误，就开始寄存器组的错误异常处理。此时，CPU 的运行如下：

1. 从与发生寄存器组错误对应的异常处理向量表中取异常服务程序的起始地址。
2. 将状态寄存器（**SR**）压栈。
3. 将程序计数器（**PC**）压栈。寄存器组发生上溢时保存的 **PC** 值为最后执行指令的下一个指令起始地址，寄存器组发生下溢时保存的 **PC** 值为该 **RESBANK** 指令的起始地址。另外，在发生寄存器组上溢时，为了防止多重中断，将寄存器组上溢源的中断级写到状态寄存器（**SR**）的中断屏蔽级位（**I3**～**I0**）。
4. 从异常服务程序的起始地址开始执行程序。

7.9 通过中断请求信号进行的数据传送

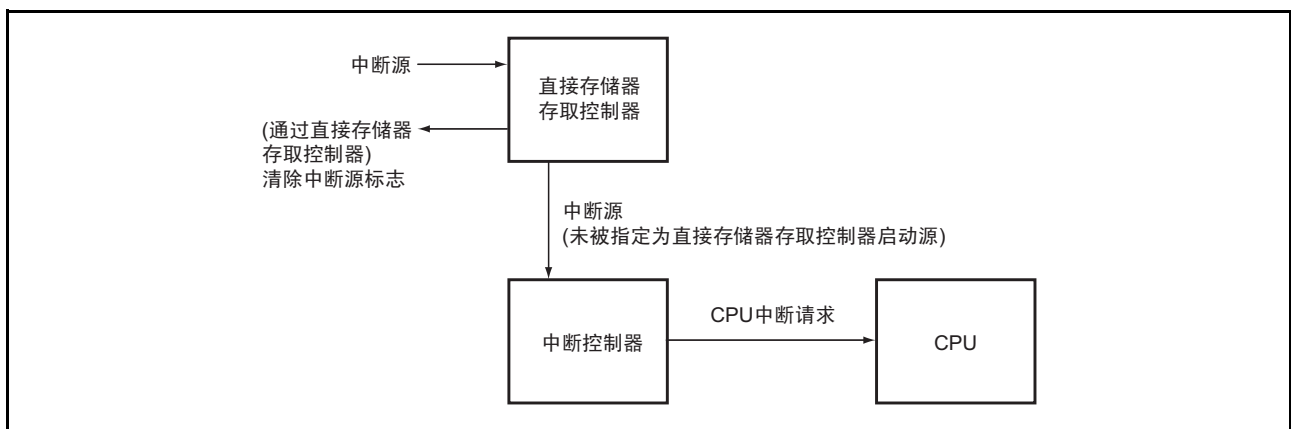
能通过中断请求信号启动直接存储器存取控制器，进行传送数据。

屏蔽中断源中被指定为直接存储器存取控制器启动源的中断源不输入到中断控制器。屏蔽条件如下：

屏蔽条件 = $DME \cdot (DE0 \cdot \text{中断源选择 } 0 + DE1 \cdot \text{中断源选择 } 1 + DE2 \cdot \text{中断源选择 } 2 + DE3 \cdot \text{中断源选择 } 3 + DE4 \cdot \text{中断源选择 } 4 + DE5 \cdot \text{中断源选择 } 5 + DE6 \cdot \text{中断源选择 } 6 + DE7 \cdot \text{中断源选择 } 7 + DE8 \cdot \text{中断源选择 } 8 + DE9 \cdot \text{中断源选择 } 9 + DE10 \cdot \text{中断源选择 } 10 + DE11 \cdot \text{中断源选择 } 11 + DE12 \cdot \text{中断源选择 } 12 + DE13 \cdot \text{中断源选择 } 13 + DE14 \cdot \text{中断源选择 } 14 + DE15 \cdot \text{中断源选择 } 15)$

中断控制框图如图 7.13 所示。

在此，DME 为直接存储器存取控制器的 DMAOR 的 bit0，DEN (n=0 ~ 15) 为直接存储器存取控制器的 CHCR0 ~ CHCR15 的 bit0，详细内容请参照“11. 直接存储器存取控制器”。



7.9.1 中断请求信号作为 CPU 中断源而不作为直接存储器存取控制器启动源的情况

1. 不选择直接存储器存取控制器的启动源，或者将 DME 位清“0”。当选择了直接存储器存取控制器的启动源时，将直接存储器存取控制器的对应通道的 DE 位清“0”。
2. 如果发生中断，就向 CPU 请求中断。
3. 在中断异常服务程序中，CPU 清除中断源并且进行必要的处理。

7.9.2 中断请求信号作为直接存储器存取控制器启动源而不作为 CPU 中断源的情况

1. 选择直接存储器存取控制器的启动源，并且将 DE 位和 DME 位置“1”。与中断优先级设定寄存器的设定无关，屏蔽 CPU 中断源。
2. 如果发生中断，就将启动源分配给直接存储器存取控制器。
3. 在传送时，直接存储器存取控制器清除启动源。

7.10 使用时的注意事项

7.10.1 中断源的清除时序

必须在中断异常服务程序中清除中断源标志。从清除中断源标志到实际取消给 CPU 的中断源的期间，需要表 7.5 记载的“从发生中断请求，到通过中断控制器判断优先级并且和 SR 屏蔽位进行比较，然后将中断请求信号送给 CPU 为止的时间”。因此为了避免再次误接受应该被清除的中断源，必须在清除后读 * 中断源标志，然后执行 RTE 指令。

【注】 * 要清除 USB2.0 主机 / 功能模块的中断源标志时，必须在清除后读 3 次中断源标志。

8. 用户断点控制器

用户断点控制器提供简单地调试程序的功能。通过使用此功能，能容易地建立自监控调试程序，即使不使用内部电路仿真器，也能通过本 LSI 简单地调试程序。作为能给此模块设定的断点条件，支持取指令或者数据的读写（读写数据时为总线周期（CPU、直接存储器存取控制器）、数据长度、数据内容、地址值以及取指令时的停止时序。由于本 LSI 采用哈佛体系结构，因此在 CPU 总线（C 总线）上取指令时，给取指令总线（F 总线）发行总线周期；在 C 总线上存取数据时，给存储器存取总线（M 总线）发行总线周期。另外，内部总线（I 总线）有内部 CPU 总线和内部 DMA 总线，CPU 给内部 CPU 总线发行总线周期，直接存储器存取控制器给内部 DMA 总线发行总线周期。此模块监控这些 C 总线和 I 总线。

8.1 特点

1. 能设定以下的断点比较条件：

断点通道数：2个通道（通道0和通道1）

通道0和通道1能独立设定用户断点。

- 地址
能按位屏蔽32位（地址）的比较。
能选择4种地址总线（F总线地址（FAB）、M总线地址（MAB）、内部CPU总线地址（ICAB）、内部DMA总线地址（IDAB））。
 - 数据
能按位屏蔽32位（数据）的比较。
可选择3种数据总线（M总线数据（MDB）、内部CPU总线数据（ICDB）、内部DMA总线数据（IDDB））中的一种。
 - 选择I总线时的总线选择
内部CPU总线或者内部DMA总线
 - 总线周期
取指令（只在选择C总线时）或者存取数据
 - 读或者写
 - 操作数长度
支持字节、字和长字。
2. 在取指令周期中，能指定用户断点中断异常处理的开始执行是设定在执行指令前还是在执行指令后。
 3. 能将断点条件成立输出到UBCTRG引脚。

框图如图 8.1 所示。

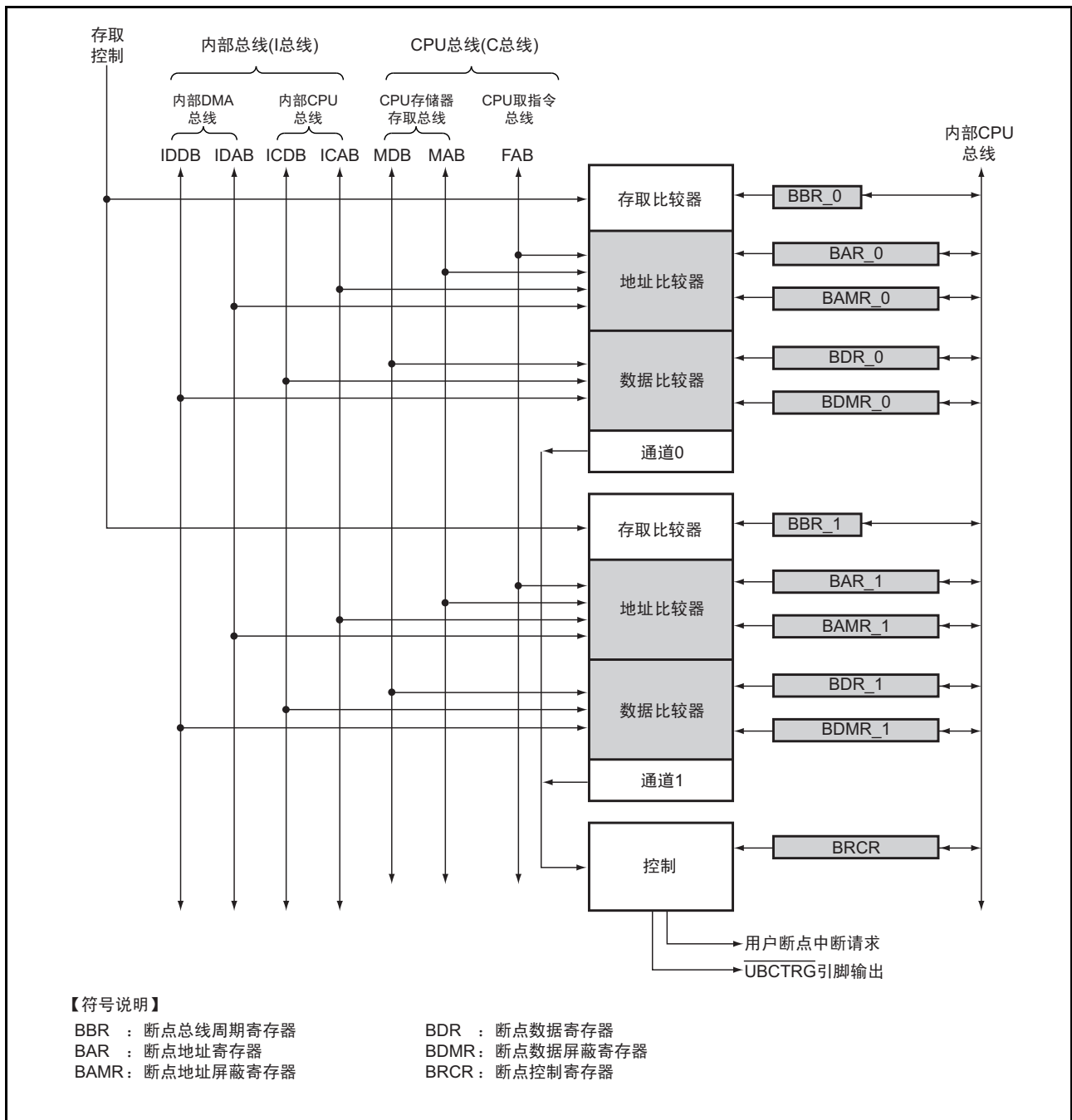


图 8.1 框图

8.2 输入 / 输出引脚

引脚结构如表 8.1 所示。

表 8.1 引脚结构

名称	引脚名	输入 / 输出	功能
用户断点触发输出	UBCTR \bar{G}	输出	表示此模块的通道 0 或者通道 1 的设定条件成立。

8.3 寄存器说明

寄存器结构如表 8.2 所示。每通道有 5 个控制寄存器，通道 0 和通道 1 有 1 个共用的控制寄存器。各通道的寄存器的记载例子：通道 0 的 BAR 记为 BAR₀。

表 8.2 寄存器结构

通道	寄存器名	略称	R/W	初始值	地址	存取长度
0	断点地址寄存器 _0	BAR_0	R/W	H'00000000	H'FFFC0400	32
	断点地址屏蔽寄存器 _0	BAMR_0	R/W	H'00000000	H'FFFC0404	32
	断点总线周期寄存器 _0	BBR_0	R/W	H'0000	H'FFFC04A0	16
	断点数据寄存器 _0	BDR_0	R/W	H'00000000	H'FFFC0408	32
	断点数据屏蔽寄存器 _0	BDMR_0	R/W	H'00000000	H'FFFC040C	32
1	断点地址寄存器 _1	BAR_1	R/W	H'00000000	H'FFFC0410	32
	断点地址屏蔽寄存器 _1	BAMR_1	R/W	H'00000000	H'FFFC0414	32
	断点总线周期寄存器 _1	BBR_1	R/W	H'0000	H'FFFC04B0	16
	断点数据寄存器 _1	BDR_1	R/W	H'00000000	H'FFFC0418	32
	断点数据屏蔽寄存器 _1	BDMR_1	R/W	H'00000000	H'FFFC041C	32
共用	断点控制寄存器	BRCR	R/W	H'00000000	H'FFFC04C0	32

8.3.1 断点地址寄存器 (BAR)

BAR 是 32 位可读写寄存器。BAR 指定作为各通道断点条件的地址。断点条件的对象地址总线有 4 种，通过断点总线周期寄存器 (BBR) 的控制位 CD[1:0] 和 CP[1:0] 进行选择。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BA31	BA30	BA29	BA28	BA27	BA26	BA25	BA24	BA23	BA22	BA21	BA20	BA19	BA18	BA17	BA16
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BA15	BA14	BA13	BA12	BA11	BA10	BA9	BA8	BA7	BA6	BA5	BA4	BA3	BA2	BA1	BA0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 0	BA31 ~ BA0	全 0	R/W	断点地址 保存指定断点条件的 CPU 地址总线 (FAB 或者 MAB) 或者内部地址总线 (ICAB 或者 IDAB) 的地址。 当通过 BBR 选择 C 总线和取指令周期时, 给 BA31 ~ BA0 指定 FAB 的地址。 当通过 BBR 选择 C 总线和数据存取周期时, 给 BA31 ~ BA0 指定 MAB 的地址。 当通过 BBR 选择 I 总线和内部 CPU 总线时, 给 BA31 ~ BA0 指定 ICAB 的地址。 当通过 BBR 选择 I 总线和内部 DMA 总线时, 给 BA31 ~ BA0 指定 IDAB 的地址。

【注】 当将取指令周期设定为断点条件时, 必须将 BAR 的 LSB 清 “0”。

8.3.2 断点地址屏蔽寄存器 (BAMR)

BAMR 是 32 位可读写寄存器。在 BAR 指定的断点地址位中, BAMR 指定要屏蔽的位。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BAM31	BAM30	BAM29	BAM28	BAM27	BAM26	BAM25	BAM24	BAM23	BAM22	BAM21	BAM20	BAM19	BAM18	BAM17	BAM16
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BAM15	BAM14	BAM13	BAM12	BAM11	BAM10	BAM9	BAM8	BAM7	BAM6	BAM5	BAM4	BAM3	BAM2	BAM1	BAM0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 0	BAM31 ~ BAM0	全 0	R/W	断点地址屏蔽 在 BAR (BA31 ~ BA0) 指定的断点地址位中, 指定要屏蔽的位。 0: 断点地址位 BAn 包含在断点条件中 1: 断点地址位 BAn 被屏蔽, 不包含在断点条件中。 【注】 n=31 ~ 0

8.3.3 断点数据寄存器 (BDR)

BDR 是 32 位可读写寄存器。断点条件的对象数据总线有 3 种，通过断点总线周期寄存器 (BBR) 的控制位 CD[1:0] 和 CP[1:0] 进行选择。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BD31	BD30	BD29	BD28	BD27	BD26	BD25	BD24	BD23	BD22	BD21	BD20	BD19	BD18	BD17	BD16
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BD15	BD14	BD13	BD12	BD11	BD10	BD9	BD8	BD7	BD6	BD5	BD4	BD3	BD2	BD1	BD0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 0	BD31 ~ BD0	全 0	R/W	断点数据位 保存指定断点条件的数据。 当通过 BBR 选择 C 总线时，给 BD31 ~ BD0 指定 MDB 的断点数据。 当通过 BBR 选择 I 总线和内部 CPU 总线时，给 BD31 ~ BD0 指定 ICDB 的地址。 当通过 BBR 选择 I 总线和内部 DMA 总线时，给 BD31 ~ BD0 指定 IDDB 的地址。

- 【注】
1. 在断点条件中包含数据总线的值时，必须指定操作数长度。
 2. 在将字节大小指定为断点条件时，作为 BDR 的断点数据，必须给 bit31 ~ 24、23 ~ 16、15 ~ 8、7 ~ 0 设定相同的字节数据。在将字大小指定为断点条件时，必须给 bit31 ~ 16、15 ~ 0 设定相同的字数据。

8.3.4 断点数据屏蔽寄存器 (BDMR)

BDMR 是 32 位可读写寄存器。在 BDR 指定的断点数据位中，BDMR 指定要屏蔽的位。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BDM31	BDM30	BDM29	BDM28	BDM27	BDM26	BDM25	BDM24	BDM23	BDM22	BDM21	BDM20	BDM19	BDM18	BDM17	BDM16
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BDM15	BDM14	BDM13	BDM12	BDM11	BDM10	BDM9	BDM8	BDM7	BDM6	BDM5	BDM4	BDM3	BDM2	BDM1	BDM0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 0	BDM31 ~ BDM0	全 0	R/W	断点数据屏蔽 在 BDR (BD31 ~ BD0) 指定的断点数据位中，指定要屏蔽的位。 0: 断点数据位 BDn 包含在断点条件中 1: 断点数据位 BDn 被屏蔽，不包含在断点条件中。 【注】 n=31 ~ 0

- 【注】
1. 在断点条件中包含数据总线的值时，必须指定操作数长度。
 2. 在将字节大小指定为断点条件时，作为 BDMR 的断点屏蔽数据，必须给 bit31 ~ 24、23 ~ 16、15 ~ 8、7 ~ 0 设定相同的字节数据。在将字大小指定为断点条件时，必须给 bit31 ~ 16、15 ~ 0 设定相同的字数据。

8.3.5 断点总线周期寄存器 (BBR)

BBR 是 16 位可读写寄存器，将以下内容指定为断点条件：

- (1) 禁止或者允许用户断点中断请求
- (2) 是否包含数据总线的值
- (3) 内部 CPU 总线或者内部 DMA 总线
- (4) C 总线周期或者 I 总线周期
- (5) 取指令或者数据存取
- (6) 读或者写
- (7) 操作数长度

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	UBID	DBE	—	—	CP[1:0]		CD[1:0]		ID[1:0]		RW[1:0]		SZ[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13	UBID	0	R/W	用户断点中断禁止 指定在条件成立时是禁止还是允许用户断点中断请求。 0: 允许用户断点中断请求 1: 禁止用户断点中断请求
12	DBE	0	R/W	数据断点允许 选择断点条件是否包含数据总线条件。 0: 断点条件不包含数据总线条件 1: 断点条件包含数据总线条件
11、10	—	全 0	R	保留位 读写值都为“0”。
9、8	CP[1:0]	00	R/W	I 总线的总线选择 选择断点条件的总线周期为 I 总线周期时的总线。但是，当总线周期为 C 总线周期时，此位无效（只限于 CPU 周期）。 00: 不进行条件比较 01: 断点条件为内部 CPU 总线 10: 断点条件为内部 DMA 总线 11: 断点条件为内部 CPU 总线
7、6	CD[1:0]	00	R/W	C 总线周期 / I 总线周期的选择 选择 C 总线周期或者 I 总线周期为断点条件的总线周期。 00: 不进行条件比较 01: 断点条件为 C 总线（F 总线、M 总线）周期 10: 断点条件为 I 总线周期 11: 断点条件为 C 总线（F 总线、M 总线）周期
5、4	ID[1:0]	00	R/W	取指令 / 数据存取的选择 选择取指令周期或者数据存取周期为断点条件的总线周期。如果选择取指令周期，就必须选择 C 总线周期。 00: 不进行条件比较 01: 断点条件为取指令周期 10: 断点条件为数据存取周期 11: 断点条件为取指令周期或者数据存取周期

位	位名	初始值	R/W	说明
3、2	RW[1:0]	00	R/W	读写选择 选择读周期或者写周期为断点条件的总线周期。 00: 不进行条件比较 01: 断点条件为读周期 10: 断点条件为写周期 11: 断点条件为读周期或者写周期
1、0	SZ[1:0]	00	R/W	操作数长度选择 选择断点条件总线周期的操作数长度。 00: 断点条件不包含操作数长度 01: 断点条件为字节存取 10: 断点条件为字存取 11: 断点条件为长字存取

8.3.6 断点控制寄存器 (BRCR)

BRCR 设定如下条件:

1. 在取指令周期中, 指定用户断点中断异常处理的开始执行是设定在执行指令前还是在执行指令后。
2. 设定在断点条件成立时 $\overline{\text{UBCTR}}\text{G}$ 输出的脉冲宽度。
3. 设定在断点条件成立时是否进行 $\overline{\text{UBCTR}}\text{G}$ 输出。

BRCR 是 32 位可读写寄存器, 有设定断点条件匹配标志和其他断点条件的位。因为只有 bit15 ~ 12 的条件匹配标志是写“1”无效(保持之前的值)而只能写“0”的位, 所以必须在清除时给要清除的标志位写“0”, 给其他的标志位写“1”。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	UTOD1	UTOD0	CKS[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCMFC ₀	SCMFC ₁	SCMFD ₀	SCMFD ₁	—	—	—	—	—	PCB1	PCB0	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 20	—	全 0	R	保留位 读写值都为“0”。
19	UTOD1	0	R/W	$\overline{\text{UBCTR}}\text{G}$ 输出禁止 1 指定在通道 1 的条件成立时是否进行 $\overline{\text{UBCTR}}\text{G}$ 输出。 0: 在通道 1 的条件成立时进行 $\overline{\text{UBCTR}}\text{G}$ 输出 1: 在通道 1 的条件成立时不进行 $\overline{\text{UBCTR}}\text{G}$ 输出
18	UTOD0	0	R/W	$\overline{\text{UBCTR}}\text{G}$ 输出禁止 0 指定在通道 0 的条件成立时是否进行 $\overline{\text{UBCTR}}\text{G}$ 输出。 0: 在通道 0 的条件成立时进行 $\overline{\text{UBCTR}}\text{G}$ 输出 1: 在通道 0 的条件成立时不进行 $\overline{\text{UBCTR}}\text{G}$ 输出

位	位名	初始值	R/W	说明
17、16	CKS[1:0]	00	R/W	<p>时钟选择</p> <p>指定在条件成立时 $\overline{\text{UBCTRG}}$ 输出的脉冲宽度。</p> <p>00: 将 $\overline{\text{UBCTRG}}$ 的脉冲宽度设定为总线时钟 1 的 1 个周期</p> <p>01: 将 $\overline{\text{UBCTRG}}$ 的脉冲宽度设定为总线时钟 1 的 2 个周期</p> <p>10: 将 $\overline{\text{UBCTRG}}$ 的脉冲宽度设定为总线时钟 1 的 4 个周期</p> <p>11: 将 $\overline{\text{UBCTRG}}$ 的脉冲宽度设定为总线时钟 1 的 8 个周期</p>
15	SCMFC0	0	R/W	<p>C 总线周期条件匹配标志 0</p> <p>如果满足通道 0 设定的断点条件的 C 总线周期条件, 就将此标志置“1”。 要清除此标志时, 给此位写“0”。</p> <p>0: 通道 0 的 C 总线周期条件不匹配</p> <p>1: 通道 0 的 C 总线周期条件匹配</p>
14	SCMFC1	0	R/W	<p>C 总线周期条件匹配标志 1</p> <p>如果满足通道 1 设定的断点条件的 C 总线周期条件, 就将此标志置“1”。 要清除此标志时, 给此位写“0”。</p> <p>0: 通道 1 的 C 总线周期条件不匹配</p> <p>1: 通道 1 的 C 总线周期条件匹配</p>
13	SCMFD0	0	R/W	<p>I 总线周期条件匹配标志 0</p> <p>如果满足通道 0 设定的断点条件的 I 总线周期条件, 就将此标志置“1”。 要清除此标志时, 给此位写“0”。</p> <p>0: 通道 0 的 I 总线周期条件不匹配</p> <p>1: 通道 0 的 I 总线周期条件匹配</p>
12	SCMFD1	0	R/W	<p>I 总线周期条件匹配标志 1</p> <p>如果满足通道 1 设定的断点条件的 I 总线周期条件, 就将此标志置“1”。 要清除此标志时, 给此位写“0”。</p> <p>0: 通道 1 的 I 总线周期条件不匹配</p> <p>1: 通道 1 的 I 总线周期条件匹配</p>
11 ~ 7	—	全 0	R	<p>保留位</p> <p>读写值都为“0”。</p>
6	PCB1	0	R/W	<p>PC 断点选择 1</p> <p>选择通道 1 取指令周期的断点时序是在执行指令前还是在执行指令后。</p> <p>0: 将通道 1 的 PC 断点设定在执行指令前</p> <p>1: 将通道 1 的 PC 断点设定在执行指令后</p>
5	PCB0	0	R/W	<p>PC 断点选择 0</p> <p>选择通道 0 取指令周期的断点时序是在执行指令前还是在执行指令后。</p> <p>0: 将通道 0 的 PC 断点设定在执行指令前</p> <p>1: 将通道 0 的 PC 断点设定在执行指令后</p>
4 ~ 0	—	全 0	R	<p>保留位</p> <p>读写值都为“0”。</p>

8.4 运行说明

8.4.1 用户断点运行的流程

从设定断点条件到用户断点中断异常处理的运行流程如下：

1. 给断点地址寄存器（BAR）指定断点地址，给断点地址屏蔽寄存器（BAMR）指定要屏蔽的地址，给断点数据寄存器（BDR）指定断点数据，给断点数据屏蔽寄存器（BDMR）指定要屏蔽的数据，并且给断点总线周期寄存器（BBR）指定总线断点条件。如果BBR的3对控制位（即C总线周期或者I总线周期的选择、取指令或者数据存取的选择和读写的选择）中的任意1对控制位为“00”，就不产生用户断点。给BRCCR的位指定断点控制，必须在设定全部断点相关的寄存器后设定BBR，并且在读最后写的寄存器后进行转移。转移目标的指令新写的寄存器的值有效。
2. 当满足断点条件并且设定允许用户断点中断请求时，此模块在将用户断点中断请求通知给中断控制器的同时，将各通道的C总线条件匹配标志（SCMFC）和I总线条件匹配标志（SCMFD）置位，并且按照CKS[1:0]位设定的脉冲宽度输出到UBCTRG引脚。另外，能通过将BBR的UBID位置“1”，在外部监视触发输出而不请求用户断点中断。
3. 如果接受用户断点中断请求信号，中断控制器就判断优先级。用户断点中断为优先级15，因此在状态寄存器（SR）的中断屏蔽级位（I3～I0）不超过优先级14时，接受该中断。当I3～I0位为优先级15时，不接受用户断点中断，但是进行条件判断，如果匹配就将条件匹配标志置位。有关优先级判断，请参照“7. 中断控制器”。
4. 能使用对应的条件匹配标志（SCMFC、SCMFD）检查设定条件是否匹配。必须在用户断点中断异常处理程序中清除条件匹配标志位。如果不清除就会再次产生中断。
5. 通道0和通道1设定的断点可能几乎同时发生，此时对于中断控制器，即使只有1个用户断点中断请求，也可能将2个断点通道匹配标志置位。
6. 如果选择I总线为断点条件，就必须注意以下事项：
 - 高速缓存的设定决定CPU在C总线上发行的存取是否也在内部CPU总线上发行。有关高速缓存条件的I总线运行，请参照“9. 高速缓存”的表9.8。
 - 当给I总线设定了断点条件时，只监视数据存取周期而不监视取指令周期（包含高速缓存更新周期）。
 - 内部DMA总线周期只为数据存取周期。
 - 在给I总线设定了断点条件的情况下，即使在由CPU执行指令引起的内部CPU总线周期中条件匹配时，也不能决定是哪条指令接受用户断点中断请求。

8.4.2 取指令周期的断点

1. 如果将“C总线/取指令/读/字或者长字”设定到断点总线周期寄存器（BBR），断点条件就为FAB总线的取指令。能通过对对应通道的断点控制寄存器（BRCCR）的PCB0或者PCB1位，选择是在执行指令前还是在执行指令后开始执行用户断点中断异常处理。在将取指令周期设定为断点条件时，必须将断点地址寄存器（BAR）的BA0位置“0”。如果将此位置“1”，就不产生断点。
2. 如果在将取指令的断点设定在执行该指令前的状态下条件匹配，就取指令并且在确定要执行指令时产生断点。因此，通过预取（在转移或者中断转换中取出但不执行的指令）取得的指令不产生断点。如果对延迟转移指令的延迟槽设定这种断点条件，就不会在执行转移目标的第1条指令前接受用户断点中断请求。

【注】如果延迟转移指令不发生转移，就不将后续指令视为延迟槽。

3. 如果在断点条件中将断点设定在执行指令后，就在执行和断点条件匹配的指令后并且在执行下一条指令前产生断点。和执行前的断点情况一样，预取指令不产生断点。如果对延迟转移指令及其延迟槽设定这种断点条件，就不会在执行转移目标的第1条指令前接受用户断点中断请求。
4. 如果设定取指令周期，就忽视断点数据寄存器（BDR）。因此，不能给取指令周期的断点设定断点数据。
5. 如果给取指令周期的断点设定I总线，就无效。

8.4.3 数据存取周期的断点

- 对数据存取断点，如果将C总线指定为断点条件，就对因执行指令而存取的地址（和数据）进行条件比较，并且产生断点；如果将I总线指定为断点条件，就对I总线选择指定的总线上的数据存取周期地址（和数据）进行条件比较，并且产生断点。有关在内部CPU总线上发行的CPU总线周期，请参照“8.4.1 用户断点运行的流程”的第6项。
- 有关数据存取周期地址和各操作数长度的比较条件关系如表8.3所示。

表 8.3 数据存取周期地址和各操作数长度的比较条件

存取长度	比较地址
长字	断点地址寄存器的 bit31 ~ 2 和地址总线的 bit31 ~ 2 进行比较。
字	断点地址寄存器的 bit31 ~ 1 和地址总线的 bit31 ~ 1 进行比较。
字节	断点地址寄存器的 bit31 ~ 0 和地址总线的 bit31 ~ 0 进行比较。

例如，当给断点地址寄存器（BAR）设定地址H'00001003时，满足断点条件的总线周期（假设满足其他全部条件）有以下含义：

H'00001000中的长字存取
H'00001002中的字存取
H'00001003中的字节存取

- 断点条件包含数据值的情况
当断点条件包含数据值时，给断点总线周期寄存器（BBR）指定长字、字或者字节作为操作数长度，并且在地址条件和数据条件匹配时产生断点。此时，为了指定字节数据，必须给断点数据寄存器（BDR）和断点数据屏蔽寄存器（BDMR）的 bit31 ~ 24、bit23 ~ 16、bit15 ~ 8、bit7 ~ 0 这4个字节指定相同的数据。为了指定字数据，必须给 bit31 ~ 16、bit15 ~ 0 这2个字指定相同的数据。
- 将PREF指令作为无存取数据的长字节读存取进行处理。因此，如果PREF指令的断点条件包含数据值，就不产生断点。
- 如果选择数据存取周期，就不能特别指定发生暂停的指令。

8.4.4 被保存的程序计数器值

在接受用户断点中断请求时，将需要继续执行的指令地址压栈，然后转移到异常处理状态。如果将C总线（FAB）/取指令指定为断点条件，就能决定产生断点的指令。如果将C总线/数据存取周期或者I总线/数据存取周期指定为断点条件，就不能决定产生断点的指令。

- 将C总线（FAB）/取指令（执行指令前）指定为断点条件的情况
将与断点条件匹配的指令地址压栈。不执行条件匹配的指令，而在执行前产生断点。但是，如果延迟槽指令发生条件匹配，就执行这些指令，并且将转移目标地址压栈。
- 将C总线（FAB）/取指令（执行指令后）指定为断点条件的情况
将与断点条件匹配指令的下一条指令地址压栈。执行条件匹配的指令，并且在执行下一条指令前产生断点。如果延迟转移指令或者延迟槽发生条件匹配，就执行这些指令，并且将转移目标地址压栈。
- 将C总线/数据存取周期或者I总线/数据存取周期指定为断点条件的情况
将与断点条件匹配的多条指令执行后的地址压栈。

8.4.5 使用例子

(1) 指定为 C 总线取指令周期的断点条件

(例 1-1)

- 寄存器指定

BAR_0=H'00000404、BAMR_0=H'00000000、BBR_0=H'0054、BAR_1=H'00008010、
BAMR_1=H'00000006、BBR_1=H'0054、BDR_1=H'00000000、BDMR_1=H'00000000、
BRCR=H'00000020

<通道 0 >

地址: H'00000404, 地址屏蔽: H'00000000

总线周期: C 总线/取指令 (执行指令后) /读 (断点条件不包含操作数长度)

<通道 1 >

地址: H'00008010, 地址屏蔽: H'00000006

数据: H'00000000, 数据屏蔽: H'00000000

总线周期: C 总线/取指令 (执行指令前) /读 (断点条件不包含操作数长度)

在执行地址 H'00000404 的指令后或者在执行地址 H'00008010 ~ H'00008016 的指令前产生用户断点。

(例 1-2)

- 寄存器指定

BAR_0=H'00027128、BAMR_0=H'00000000、BBR_0=H'005A、BAR_1=H'00031415、
BAMR_1=H'00000000、BBR_1=H'0054、BDR_1=H'00000000、BDMR_1=H'00000000、
BRCR=H'00000000

<通道 0 >

地址: H'00027128, 地址屏蔽: H'00000000

总线周期: C 总线/取指令 (执行指令前) /写/字

<通道 1 >

地址: H'00031415, 地址屏蔽: H'00000000

数据: H'00000000, 数据屏蔽: H'00000000

总线周期: C 总线/取指令 (执行指令前) /读 (断点条件不包含操作数长度)

在通道 0 中, 因为取指令不是写周期, 所以不产生用户断点; 在通道 1 中, 因为对偶数地址取指令, 所以也不产生用户断点。

(例 1-3)

- 寄存器指定

BAR_0=H'00008404、BAMR_0=H'00000FFF、BBR_0=H'0054、BAR_1=H'00008010、
BAMR_1=H'00000006、BBR_1=H'0054、BDR_1=H'00000000、BDMR_1=H'00000000、
BRCR=H'00000020

<通道 0 >

地址: H'00008404, 地址屏蔽: H'00000FFF

总线周期: C 总线/取指令 (执行指令后) /读 (断点条件不包含操作数长度)

<通道 1 >

地址: H'00008010, 地址屏蔽: H'00000006

数据: H'00000000, 数据屏蔽: H'00000000

总线周期: C 总线/取指令 (执行指令前) /读 (断点条件不包含操作数长度)

在执行地址 H'00008000 ~ H'00008FFE 的指令后或者在执行地址 H'00008010 ~ H'00008016 的指令前产生用户断点。

(2) 指定为 C 总线数据存取周期的断点条件

(例 2-1)

• 寄存器指定

BAR_0=H'00123456、BAMR_0=H'00000000、BBR_0=H'0064、BAR_1=H'000ABCDE、
BAMR_1=H'000000FF、BBR_1=H'106A、BDR_1=H'A512A512、BDMR_1=H'00000000、
BRCCR=H'00000000

<通道 0 >

地址: H'00123456, 地址屏蔽: H'00000000

总线周期: C 总线/数据存取/读 (断点条件不包含操作数长度)

<通道 1 >

地址: H'000ABCDE, 地址屏蔽: H'000000FF

数据: H'0000A512, 数据屏蔽: H'00000000

总线周期: C 总线/数据存取/写/字

在通道 0 中, 在读地址 H'00123456 的长字、或者地址 H'00123456 的字、或者地址 H'00123456 的字节时产生用户断点; 在通道 1 中, 在给 H'000ABC00 ~ H'000ABCFE 写字数据 H'A512 时产生用户断点。

(3) 指定为 I 总线数据存取周期的断点条件

(例 3-1)

• 寄存器指定:

BAR_0=H'00314156、BAMR_0=H'00000000、BBR_0=H'0194、BAR_1=H'00055555、
BAMR_1=H'00000000、BBR_1=H'12A9、BDR_1=H'78787878、BDMR_1=H'0F0F0F0F、
BRCCR=H'00000000

<通道 0 >

地址: H'00314156, 地址屏蔽: H'00000000

总线周期: 内部 CPU 总线/取指令/读 (断点条件不包含操作数长度)

<通道 1 >

地址: H'00055555, 地址屏蔽: H'00000000

数据: H'00000078, 数据屏蔽: H'0000000F

总线周期: 内部 DMA 总线/数据存取/写/字节

在通道 0 中, 内部 CPU 总线上的取指令设定无效。在通道 1 中, 直接存储器存取控制器在内部 DMA 总线上将字节 H'7x 写到地址 H'00055555 时产生用户断点 (在内部 CPU 总线的存取时不发生)。

8.5 使用时的注意事项

1. 通过内部CPU总线读写此模块的寄存器。因此，在执行改写此模块寄存器的指令后到实际反映该值的期间，可能不产生想要的断点。为得知此模块寄存器的更改时序，必须读最后写的寄存器。此后的指令对新写的寄存器值有效。
2. 此模块不能用相同的通道监视C总线周期、内部CPU总线周期和内部DMA总线周期。
3. 如果在同一条指令中发生用户断点中断请求和其他异常源，就按照“6. 异常处理”的表6.1规定的优先级进行判断。如果发生更高优先级的异常源，就不接受用户断点中断请求。
4. 在延迟槽中产生断点时，有以下注意事项：
如果给延迟槽指令设定指令执行前的断点，就在执行其转移目标前不接受用户断点中断请求。
5. 在模块待机模式中，不能使用用户断点功能，并且不能读写此模块的寄存器，否则就不保证其值。
6. 不能将存放中断优先级为大于等于15（含用户断点中断）的中断异常处理程序的地址设定为断点地址。
7. 对SLEEP指令和延迟槽为SLEEP指令的转移指令，不能设定指令执行后的断点。
8. 要给32位指令设定断点地址时，必须设定在前16位侧。如果将断点地址设定在后16位侧，即使断点条件设定在执行指令前也在执行指令后进行处理。
9. 不能给DIVU指令或者DIVS指令的下一条指令设定指令执行前的断点。如果给DIVU指令或者DIVS指令的下一条指令设定指令执行前的断点，就会在DIVU指令或者DIVS指令的执行过程中发生异常或中断，并且即使中止执行DIVU指令或者DIVS指令，也会在执行下一条指令前产生断点。

9. 高速缓存

9.1 特点

- 容量
指令高速缓存：8K 字节
操作数高速缓存：8K 字节
- 结构：指令/数据分离、4 路组相联
- 路锁定功能（只限于操作数高速缓存）：能锁定路 2 和路 3
- 行大小：16 字节
- 入口数：128 个入口/路
- 写方式：可选择回写方式和直写方式
- 替换方式：LRU 替换算法

9.1.1 高速缓存的结构

高速缓存为指令 / 数据分离型的 4 路组相联方式。由 4 个路（存储体）构成，各路分成地址和数据。

地址和数据的每 1 路分别由 128 个入口构成。入口的数据称为行。1 行为 16 字节（4 字节 × 4）。每 1 路的数据容量为 2K 字节（16 字节 × 128 个入口），高速缓存整体容量（4 路）为 8K 字节。

操作数高速缓存的结构如图 9.1 所示。除了没有 U 位以外，指令高速缓存的结构和操作数高速缓存相同。

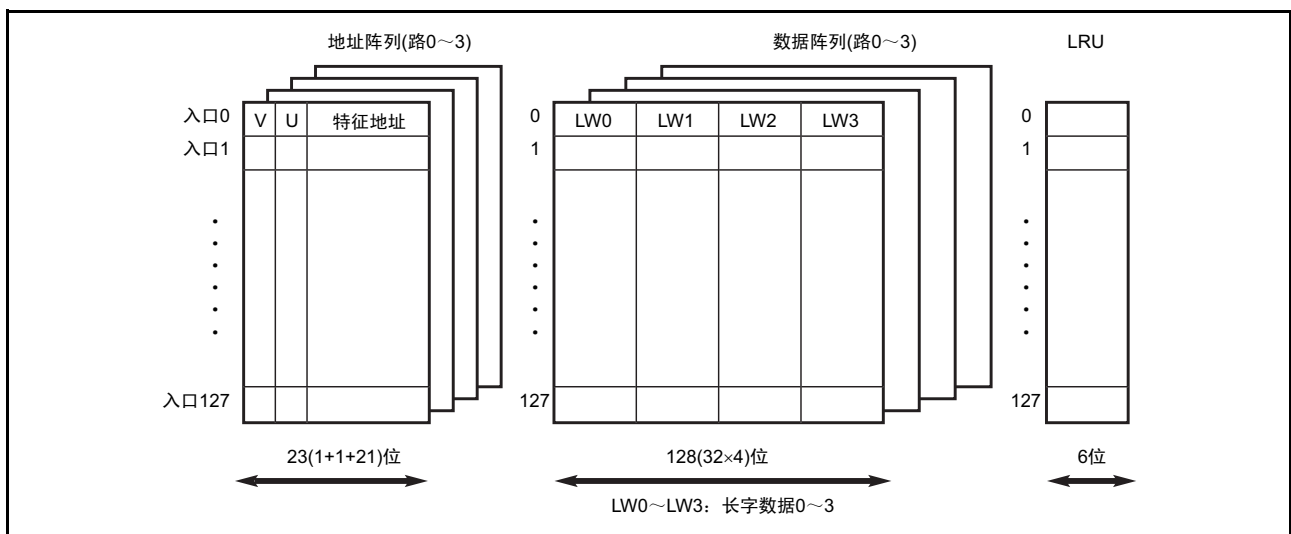


图 9.1 操作数高速缓存的结构

(1) 地址阵列

V 位表示入口的数据是否有效。V 位为“1”表示有效，为“0”表示无效。

U 位（只限于操作数高速缓存），表示在回写模式中该入口的写操作。U 位为“1”表示进行了写操作，为“0”表示没有进行写操作。

特征地址保持用于外部存储器 / 大容量内部 RAM 的存取地址，由 21 个位（地址 31 ~ 11）构成，这些位用于高速缓存搜索时的比较。本 LSI 中，高速缓存有效空间的地址为 H'00000000 ~ H'1FFFFFFF（参照“10. 总线状态控制器”），因此特征地址的高 3 位为“0”。

在上电复位时，将 V 位和 U 位初始化为“0”，但是在手动复位和软件待机模式时不进行初始化。在上电复位、手动复位和软件待机模式时，不对特征地址进行初始化。

(2) 数据阵列

数据阵列保存 16 字节的指令或者数据。以行为单位（以 16 字节为单位）注册高速缓存的入口。
在上电复位、手动复位和软件待机模式时，不对数据阵列进行初始化。

(3) LRU

通过 4 路组相联方式，最多能将 4 个相同入口地址的指令或者数据注册到高速缓存。在注册入口时，LRU 位表示注册到 4 路中的哪一路。LRU 位由 6 位构成，通过硬件控制。作为路选择算法，使用选择最近存取的路的 LRU（Least Recently Used）算法。

通过 6 位的 LRU 位指定高速缓存未命中时被替换的路。不使用高速缓存锁定功能（只限于操作数高速缓存）时的 LRU 位和被替换的路的关系如表 9.1 所示（在使用高速缓存锁定功能时，请参照“9.2.2 高速缓存控制寄存器 2（CCR2）”）。如果通过软件指定表 9.1 所示以外的 LRU 位，高速缓存就不能正常运行。在通过软件更改 LRU 位时，必须设定如表 9.1 所示的模式。

在上电复位时，将 LRU 位初始化为“B'000000”，但是在手动复位和软件待机模式时不进行初始化。

表 9.1 LRU 位和被替换的路（不使用高速缓存锁定功能的情况）

LRU (bit5 ~ 0)	被替换的路
000000、000100、010100、100000、110000、110100	3
000001、000011、001011、100001、101001、101011	2
000110、000111、001111、010110、011110、011111	1
111000、111001、111011、111100、111110、111111	0

9.2 寄存器说明

寄存器结构如表 9.2 所示。

表 9.2 寄存器结构

寄存器名	略称	R/W	初始值	地址	存取长度
高速缓存控制寄存器 1	CCR1	R/W	H'00000000	H'FFFC1000	32
高速缓存控制寄存器 2	CCR2	R/W	H'00000000	H'FFFC1004	32

9.2.1 高速缓存控制寄存器 1 (CCR1)

通过 ICE 位指定允许或者禁止指令高速缓存，ICF 位控制指令高速缓存全部入口的无效化，通过 OCE 位指定允许或者禁止操作数高速缓存，OCF 位控制操作数高速缓存全部入口的无效化，WT 位切换操作数高速缓存的直写模式和回写模式。

必须将更改 CCR1 内容的程序分配在高速缓存的无效空间，在读 CCR1 的内容后存取高速缓存的有效空间。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	ICF	—	—	ICE	—	—	—	—	OCF	—	WT	OCE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R	R	R/W	R	R	R	R	R/W	R	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 12	—	全 0	R	保留位 读写值都为“0”。
11	ICF	0	R/W	指令高速缓存转储清除 如果给此位写“1”，就将指令高速缓存全部入口的 V 位和 LRU 位清“0”（转储清除）。读取值为“0”。在转储清除时不回写外部存储器 / 大容量内部 RAM。
10、9	—	全 0	R	保留位 读写值都为“0”。
8	ICE	0	R/W	指令高速缓存有效 指定允许或者禁止指令高速缓存功能。 0: 禁止指令高速缓存 1: 允许指令高速缓存
7 ~ 4	—	全 0	R	保留位 读写值都为“0”。
3	OCF	0	R/W	操作数高速缓存转储清除 如果给此位写“1”，就将操作数高速缓存全部入口的 V 位、U 位和 LRU 位清“0”（转储清除）。读取值为“0”。在转储清除时不回写外部存储器 / 大容量内部 RAM。
2	—	0	R	保留位 读写值都为“0”。
1	WT	0	R/W	直写 进行回写和直写的切换。 0: 回写模式 1: 直写模式
0	OCE	0	R/W	操作数高速缓存有效 指定允许或者禁止操作数高速缓存功能。 0: 禁止操作数高速缓存 1: 允许操作数高速缓存

9.2.2 高速缓存控制寄存器 2 (CCR2)

CCR2 是控制操作数高速缓存的高速缓存锁定功能的寄存器。高速缓存锁定功能只在高速缓存锁定模式时有效，高速缓存锁定模式是指 CCR2 的锁定允许位 (LE 位) 为“1”的状态。在非高速缓存锁定模式中，高速缓存锁定功能无效。

如果在高速缓存锁定模式中执行预取指令 (PREF @Rn) 而高速缓存未命中，就根据 CCR2 的 bit9 和 bit8 (W3LOAD 位和 W3LOCK 位) 以及 bit1 和 bit0 (W2LOAD 位和 W2LOCK 位) 的设定，将 Rn 所示的 1 行数据取到高速缓存。执行预取指令时的各位设定和被替换的路的关系如表 9.3 所示。如果执行预取指令并且高速缓存命中，就不取新的数据而保持已经有效的入口。例如，在 Rn 所示的 1 行数据已存在于路 0 的状态下，如果在高速缓存锁定模式中将 W3LOAD 位和 W3LOCK 位都置“1”并且执行预取指令，就为高速缓存命中而不将数据取到路 3。

除了高速缓存锁定模式中的预取指令以外，高速缓存的存取受制于通过 W3LOCK 位和 W2LOCK 位被替换的路。CCR2 各位的设定与被替换的路的关系如表 9.4 所示。

必须将更改 CCR2 内容的程序分配在高速缓存的无效空间，在读 CCR2 的内容后存取高速缓存的有效空间。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	W3 LOAD*	W3 LOCK	—	—	—	—	—	—	W2 LOAD*	W2 LOCK
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

【注】* 不能将W3LOAD和W2LOAD同时置“1”。

位	位名	初始值	R/W	说明
31 ~ 17	—	全 0	R	保留位 读写值都为“0”。
16	LE	0	R/W	锁定允许 控制高速缓存锁定模式。 0: 非高速缓存锁定模式 1: 高速缓存锁定模式
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 8	W3LOAD* W3LOCK	0 0	R/W R/W	路 3 装入 路 3 锁定 在高速缓存锁定模式并且 W3LOCK 位和 W3LOAD 位都为“1”时，通过预取指令发生的高速缓存未命中的数据总是被读取到路 3。在其他的全部条件下，高速缓存未命中的数据被读取到 LRU 所示的路。
7 ~ 2	—	全 0	R	保留位 读写值都为“0”。
1 0	W2LOAD* W2LOCK	0 0	R/W R/W	路 2 装入 路 2 锁定 在高速缓存锁定模式并且 W2LOCK 位和 W2LOAD 位都为“1”时，通过预取指令发生的高速缓存未命中的数据总是被读取到路 2。在其他的全部条件下，高速缓存未命中的数据被读取到 LRU 所示的路。

【注】* 不能将 W3LOAD 位和 W2LOAD 位同时置“1”。

表 9.3 在 PREF 指令的情况下高速缓存未命中时被替换的路

LE	W3LOAD*	W3LOCK	W2LOAD*	W2LOCK	被替换的路
0	x	x	x	x	根据 LRU (表 9.1)
1	x	0	x	0	根据 LRU (表 9.1)
1	x	0	0	1	根据 LRU (表 9.5)
1	0	1	x	0	根据 LRU (表 9.6)
1	0	1	0	1	根据 LRU (表 9.7)
1	0	x	1	1	路 2
1	1	1	0	x	路 3

【符号说明】 x: Don't care

【注】 * 不能将 W3LOAD 位和 W2LOAD 位同时置“1”。

表 9.4 在不是 PREF 指令的情况下高速缓存未命中时被替换的路

LE	W3LOAD*	W3LOCK	W2LOAD*	W2LOCK	被替换的路
0	×	×	×	×	根据 LRU (表 9.1)
1	×	0	×	0	根据 LRU (表 9.1)
1	×	0	×	1	根据 LRU (表 9.5)
1	×	1	×	0	根据 LRU (表 9.6)
1	×	1	×	1	根据 LRU (表 9.7)

【记号说明】 ×: Don't care

【注】 * 不能将 W3LOAD 位和 W2LOAD 位同时设置“1”。

表 9.5 LRU 位和被替换的路 (W2LOCK 位为“1”并且 W3LOCK 位为“0”的情况)

LRU (bit5 ~ 0)	被替换的路
000000、000001、000100、010100、100000、100001、110000、110100	3
000011、000110、000111、001011、001111、010110、011110、011111	1
101001、101011、111000、111001、111011、111100、111110、111111	0

表 9.6 LRU 位和被替换的路 (W2LOCK 位为“0”并且 W3LOCK 位为“1”的情况)

LRU (bit5 ~ 0)	被替换的路
000000、000001、000011、001011、100000、100001、101001、101011	2
000100、000110、000111、001111、010100、010110、011110、011111	1
110000、110100、111000、111001、111011、111100、111110、111111	0

表 9.7 LRU 位和被替换的路 (W2LOCK 位为“1”并且 W3LOCK 位为“1”的情况)

LRU (bit5 ~ 0)	被替换的路
000000、000001、000011、000100、000110、000111、001011、001111、010100、010110、011110、011111	1
100000、100001、101001、101011、110000、110100、111000、111001、111011、111100、111110、111111	0

9.3 运行说明

本节对操作数高速缓存进行说明。除了地址阵列中没有U位并且没有预取操作、写操作和回写缓冲器以外，指令高速缓存和操作数高速缓存相同。

9.3.1 高速缓存的搜索

如果在允许操作数高速缓存时（CCR1 寄存器的 OCE 位为“1”）存取高速缓存有效空间的数据，就搜索高速缓存，调查目标数据是否存在于高速缓存中。高速缓存搜索方法的概念图如图 9.2 所示。

通过存储器存取地址的 bit10 ~ 4 选择入口，并且读该入口的特征地址。此时，特征地址的高 3 位总是为“0”。将存储器存取地址的 bit31 ~ 11 和读取的特征地址进行比较，4 路都进行地址的比较。如果相同并且被比较的入口有效（V位为“1”），就为高速缓存命中，否则就为高速缓存未命中。路1命中时的情况如图9.2所示。

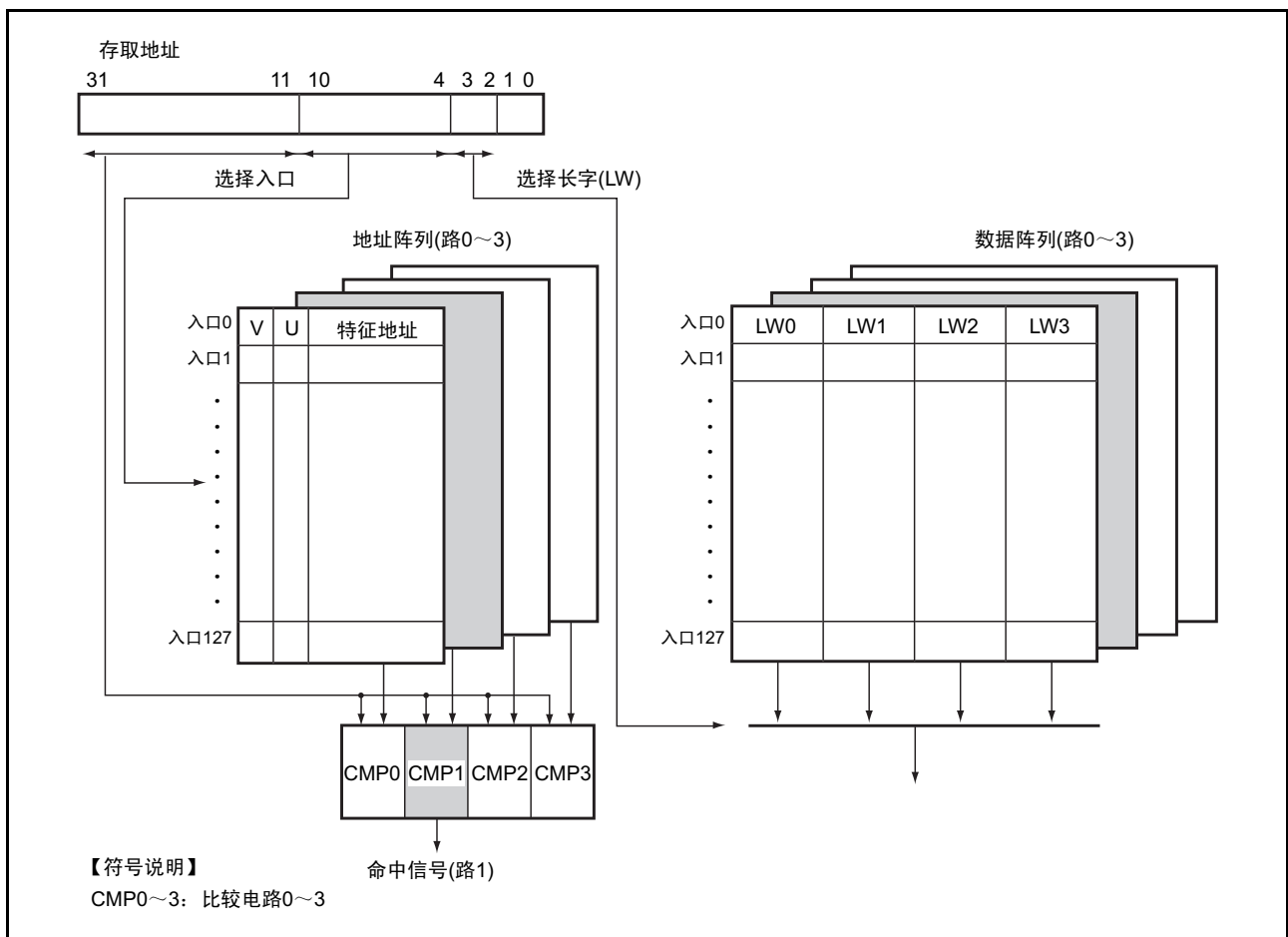


图 9.2 高速缓存的搜索方法

9.3.2 读操作

(1) 读命中

将数据从高速缓存传送到 CPU，更新 LRU 以便命中的路为最新。

(2) 读未命中

启动外部总线周期并且更新入口。被替换的路如表 9.4 所示。入口的更新单位为 16 字节。在将目标数据从外部存储器 / 大容量内部 RAM 注册到高速缓存的同时，将该数据传送到 CPU。在注册到高速缓存时，将 V 位置 “1”，并且更新 LRU 以便被替换的路为最新。在操作数高速缓存的情况下，还需要将 U 位置 “0”，如果在回写模式中因更新入口而被替换的入口的 U 位为 “1”，就在将该入口传送到回写缓冲器后开始高速缓存更新周期。在高速缓存更新周期结束后，将传送到回写缓冲器的入口回写到存储器，回写的单位为 16 字节，并且以缓绕方式更新高速缓存并且回写存储器。例如，当发生读未命中的地址低 4 位为 “H'4” 时，以地址低 4 位为 H'4→H'8→H'C→H'0 的顺序更新高速缓存并且回写存储器。

9.3.3 预取操作（只限于操作数高速缓存）

(1) 预取命中

更新 LRU 以便命中的路为最新，但是不更改其他高速缓存的内容，也不将数据传送到 CPU。

(2) 预取未命中

不将数据传送到 CPU，替换的路如表 9.3 所示。其他的运行和读未命中时相同。

9.3.4 写操作（只限于操作数高速缓存）

(1) 写命中

在回写模式中，将数据写到高速缓存，但是不给外部存储器 / 大容量内部 RAM 发行写周期。将被写入口的 U 位置 “1”，并且更新 LRU 以便命中的路为最新。

在直写模式中，将数据写到高速缓存，并且给外部存储器 / 大容量内部 RAM 发行写周期。不更新被写入口的 U 位，但是更新 LRU 以便命中的路为最新。

(2) 写未命中

在回写模式中写未命中时，启动外部总线周期并且更新入口。替换的路如表 9.4 所示。如果因更新入口而被替换的入口的 U 位为 “1”，就在将该入口传送到回写缓冲器后开始高速缓存更新周期。将数据写到高速缓存，将 U 位和 V 位置 “1”，并且更新 LRU 以便被替换的路为最新。在高速缓存更新周期结束后，将传送到回写缓冲器的入口回写到存储器，回写的单位为 16 字节，并且以缓绕方式更新高速缓存并且回写存储器。例如，当发生写未命中的地址低 4 位为 “H'4” 时，以地址低 4 位为 H'4→H'8→H'C→H'0 的顺序更新高速缓存并且回写存储器。

在直写模式中写未命中时，不写高速缓存而只写外部存储器 / 大容量内部 RAM。

9.3.5 回写缓冲器（只限于操作数高速缓存）

在回写模式中被替换的入口的 U 位为“1”时，需要回写外部存储器 / 大容量内部 RAM。为了提高性能，首先将被替换的入口传送到回写缓冲器，优先回写被取到高速缓存的新入口。在将新入口取到高速缓存后，将回写缓冲器的入口回写到外部存储器 / 大容量内部 RAM。在此回写过程中，能存取高速缓存。

回写缓冲器能保持高速缓存的 1 行数据（16 字节）及其地址。回写缓冲器的结构如图 9.3 所示。

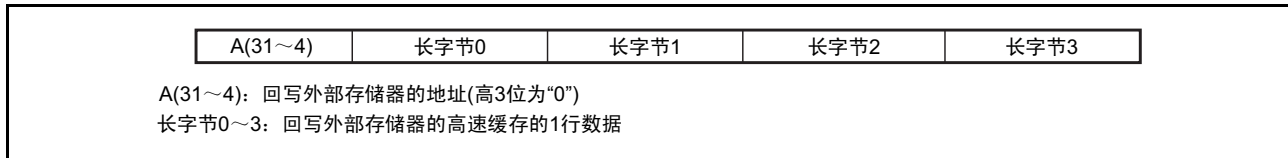


图 9.3 回写缓冲器的结构

上述 9.3.2 ~ 9.3.5 的运行汇总于表 9.8。

表 9.8 高速缓存运行的汇总

高速缓存的种类	CPU 周期	命中 / 未命中	回写模式 / 直写模式	U 位	对外部存储器 / 大容量内部 RAM 的存取（通过内部总线）	高速缓存的内容
指令高速缓存	取指令	命中	—	—	不发生。	不更新。
		未命中	—	—	产生高速缓存更新周期。	根据左记更新周期的内容进行更新。
操作数高速缓存	预取 / 读	命中	任意模式	x	不发生。	不更新。
		未命中	直写模式	—	产生高速缓存更新周期。	根据左记更新周期的内容进行更新。
			回写模式	0	产生高速缓存更新周期。	根据左记更新周期的内容进行更新。
				1	产生高速缓存更新周期，然后产生回写缓冲器的回写周期。	根据左记更新周期的内容进行更新。
	写	命中	直写模式	—	产生 CPU 发行的写周期。	根据 CPU 发行的写周期的内容进行更新。
			回写模式	x	不发生。	根据 CPU 发行的写周期的内容进行更新。
		未命中	直写模式	—	产生 CPU 发行的写周期。	不更新。*
			回写模式	0	产生高速缓存更新周期。	在根据左记更新周期的内容进行更新后，根据 CPU 发行的写周期的内容进行更新。
1	产生高速缓存更新周期，然后产生回写缓冲器的回写周期。	在根据左记更新周期的内容进行更新后，根据 CPU 发行的写周期的内容进行更新。				

【符号说明】×：Don't care

【注】 高速缓存更新周期：16 字节的读存取

回写缓冲器的回写周期：16 字节的写存取

* 也不更新 LRU。在其他情况下更新 LRU。

9.3.6 高速缓存和外部存储器 / 大容量内部 RAM 的相关性

必须通过软件保证高速缓存和外部存储器 / 大容量内部 RAM 的相关性。

当要将本 LSI 和其他器件的共享存储器分配在高速缓存的有效空间时，必须根据需要进行存储器映射高速缓存的操作，进行无效化和回写。对于本 LSI 内的 CPU 和直接存储器存取控制器的共享存储器也必须进行同样的操作。

9.4 存储器映射高速缓存的结构

因为通过软件管理高速缓存，所以能通过 MOV 指令读写高速缓存的内容。指令高速缓存的地址阵列分配在 H'F000 0000 ~ H'F07F FFFF，数据阵列分配在 H'F100 0000 ~ H'F17F FFFF。操作数高速缓存的地址阵列分配在 H'F080 0000 ~ H'F0FF FFFF，数据阵列分配在 H'F180 0000 ~ H'F1FF FFFF。地址阵列和数据阵列的存取长度都固定为长字，不进行取指令。

9.4.1 地址阵列

在存取地址阵列时，需要指定 32 位地址（读写时）和 32 位数据（写时）。

给地址指定要选择入口的入口地址、要选择路的 W 位以及指定有无关联运行的 A 位。W 位的 B'00、B'01、B'10 和 B'11 分别表示路 0、路 1、路 2 和路 3。因为存取长度固定为长字，所以必须给地址的 bit1 ~ 0 指定“B'00”。

给数据指定特征地址、LRU 位、U 位（只限于操作数高速缓存）和 V 位。必须将特征地址的高 3 位（bit31 ~ 29）指定为“0”。

有关地址和数据的格式，请参照图 9.4。

能对地址阵列进行以下 3 种操作。

(1) 读地址阵列

从地址指定的入口地址以及与路对应的入口读特征地址、LRU 位、U 位（只限于操作数高速缓存）和 V 位。读时，无论地址指定的关联位（A 位）是“1”还是“0”，都不进行关联运行。

(2) 写地址阵列（无关联）

如果在将地址的关联位（A 位）置“0”后写地址阵列，就对地址指定的入口地址以及与路对应的入口，写数据指定的特征地址、LRU 位、U 位（只限于操作数高速缓存）和 V 位。如果对 U 位为“1”并且 V 位为“1”的高速缓存行进行了对操作数高速缓存地址阵列的写操作，就在回写该高速缓存行后写数据指定的特征地址、LRU 位、U 位和 V 位。但是，在给 V 位写“0”时，也必须给该入口的 U 位写“0”。以地址低 4 位为 H'0→H'4→H'8→H'C 的顺序回写存储器。

(3) 写地址阵列（有关联）

如果在将地址的关联位（A 位）置“1”后写地址阵列，就判断地址指定的入口的 4 路地址是否和数据指定的特征地址相同。对于命中（比较结果相同）的路，给入口写数据指定的 U 位（只限于操作数高速缓存）和 V 位。但是，不更改特征地址和 LRU 位。当哪个路都不命中时，不进行写操作而为空操作。此运行用于高速缓存特定入口的无效化。

对于操作数高速缓存，如果命中入口的 U 位为“1”，就进行回写。但是，在给 V 位写“0”时，也必须给该入口的 U 位写“0”。以地址低 4 位为 H'0→H'4→H'8→H'C 的顺序回写存储器。

9.4.2 数据阵列

在存取数据阵列时，需要指定 32 位地址（读写时）和 32 位数据（写时）。给地址指定要选择存取入口的信息，给数据指定要写数据阵列的长字数据。

给地址指定要选择入口的入口地址、表示 1 行（16 字节）中的长字位置的 L 位以及要指定路的 W 位。L 位的 B'00、B'01、B'10 和 B'11 分别表示长字 0、长字 1、长字 2 和长字 3，W 位的 B'00、B'01、B'10 和 B'11 分别表示路 0、路 1、路 2 和路 3。因为存取长度固定为长字，所以必须给地址的 bit1 ~ 0 指定“B'00”。

有关地址和数据的格式，请参照图 9.4。

能对数据阵列进行以下 2 种操作，这些操作不会改变地址阵列的信息。

(1) 读数据阵列

从地址指定的入口地址以及与路对应的入口，读地址的 L 位指定的数据。

(2) 写数据阵列

在地址指定的入口地址以及与路对应的入口中，给地址的 L 位指定的位置写数据指定的长字数据。

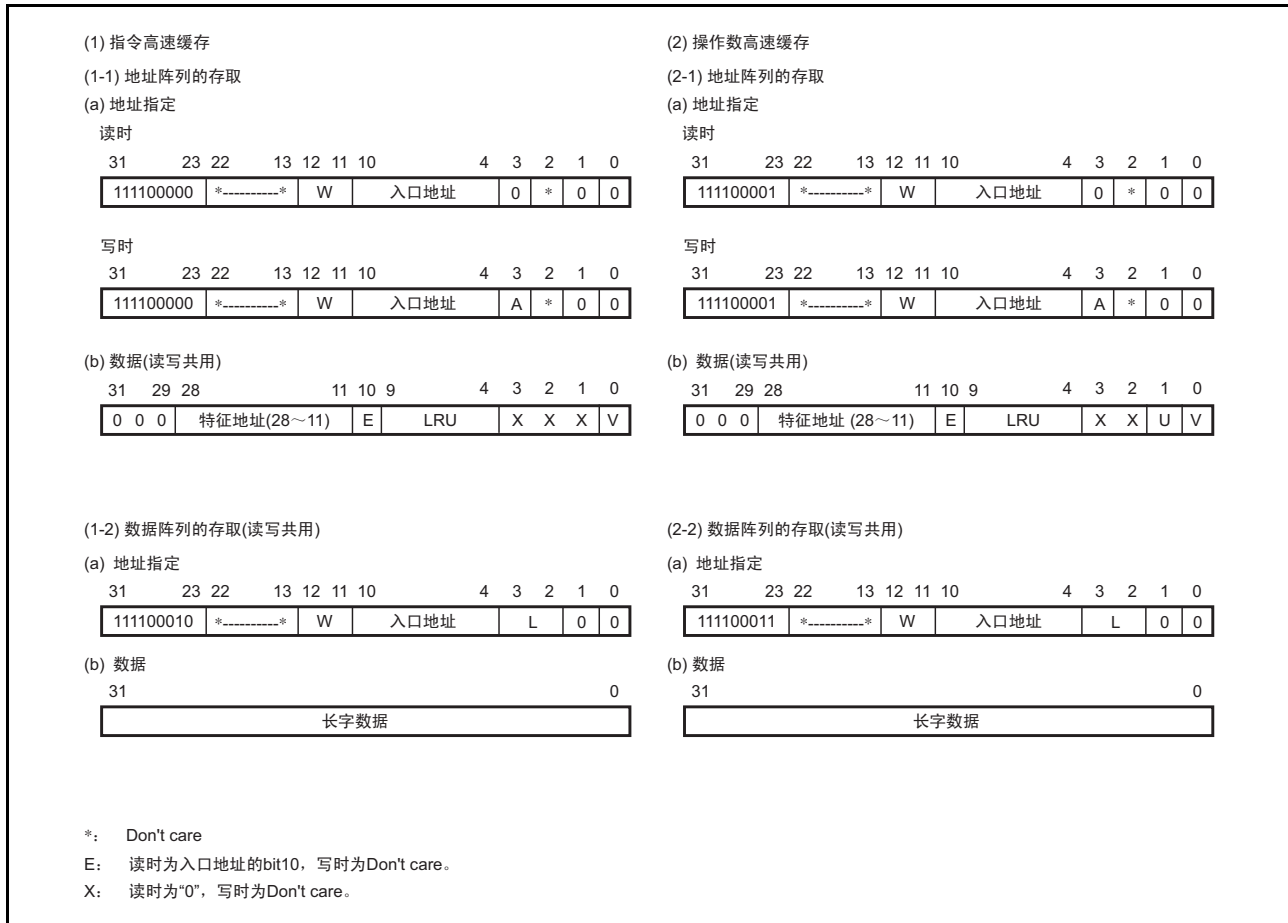


图 9.4 存储器映射高速缓存存取的地址和数据的指定方法

9.4.3 使用例子

(1) 特定入口的无效化

在存取存储器映射高速缓存中，能通过给该入口的 V 位写“0”实现高速缓存特定入口的无效化。假设 A 位为“1”，将写数据指定的特征地址和入口地址选择的高速缓存中的特征地址进行比较。相同时，写 V 位和 U 位（由写数据指定）；不相同时空操作。如果将地址阵列某个入口的 V 位置“0”，就在该入口的 U 位为“1”时回写该入口。

以下是给 R0 指定写数据并且给 R1 指定地址的例子：

```

; R0=H'0110 0010 ; 特征地址 (28 ~ 11) =B'0 0001 0001 0000 0000 0、U=0、V=0
; R1=H'F080 0088 ; 操作数高速缓存地址阵列的存取、入口 =B'000 1000、A=1
;
MOV.L R0, @R1

```

(2) 读特定入口的数据

能通过存取存储器映射高速缓存，读特定入口的数据。将图 9.4 的数据阵列的数据所示的长字读取到寄存器。

以下是给 R0 指定地址并且读到 R1 的例子。

```

; R0=H'F100 004C ; 指令高速缓存数据阵列的存取、入口 =B'000 0100、路 =0、
长字节地址 =3
;
MOV.L @R0, R1

```

9.4.4 注意事项

1. 必须将存取操作数高速缓存的存储器映射高速缓存的程序分配在高速缓存的无效空间；必须将存取指令高速缓存的存储器映射高速缓存的程序分配在高速缓存的无效空间，并且需要在程序的起始位置和最后位置分别对内部外围模块或者外部地址空间（高速缓存无效地址）进行不少于2次的读存取。
2. 禁止将地址阵列的内容改写为多个路同时命中，否则不保证运行。
3. 只能通过CPU而不能通过直接存储器存取控制器存取寄存器和存储器映射高速缓存。

10. 总线状态控制器

总线状态控制器对连接外部地址空间的各种存储器和外部设备输出控制信号，能直接连接 SRAM、SDRAM 等各种存储器和外部设备。

10.1 特点

1. 外部地址空间
 - CS0～CS5 各空间最多支持 64M 字节。
 - 每个空间都能指定正规空间接口、带字节选择的 SRAM 接口、突发 ROM（时钟同步或者异步）、MPX-I/O、SDRAM 的存储器种类和 PCMCIA 接口。
 - 每个空间都能选择数据总线宽度（8 位、16 位或者 32 位）。
 - 每个空间都能控制等待状态的插入。
 - 每次读写存取都能控制等待状态的插入。
 - 在连续存取为读-写（同一空间或者不同空间）、读-读（同一空间或者不同空间）以及起始周期为写周期的 5 种情况下，能独立设定空闲周期。
2. 正规空间接口
 - 支持能直接连接 SRAM 的接口。
3. 突发 ROM（异步）接口
 - 能高速存取具有页模式功能的 ROM
4. MPX-I/O 接口
 - 能直接连接地址/数据多路复用所需的外围 LSI。
5. SDRAM 接口
 - 最多能在 2 个区域设定 SDRAM。
 - 支持行地址或者列地址的多路复用输出。
 - 能通过单次读或者单次写进行高效存取。
 - 能通过存储体激活模式进行高速存取。
 - 支持自动刷新和自刷新。
 - 支持掉电模式。
 - 支持 MRS 命令和 EMRS 命令的发行。
6. PCMCIA 接口
 - 支持 JEIDA 规格 Ver4.2（PCMCIA2.1 Rev2.1）规定的 IC 存储卡和 I/O 卡接口。
 - 能通过程序控制等待状态的插入。
7. 带字节选择的 SRAM 接口
 - 支持能直接连接带字节选择的 SRAM 接口。
8. 突发 ROM（时钟同步）接口
 - 能直接连接时钟同步型的突发 ROM。
9. 总线仲裁
 - 与其他 CPU 共享全部资源，能接受来自外部的总线权请求并且能输出总线使用允许信号。
10. 刷新功能
 - 支持自动刷新和自刷新。
 - 能通过刷新计数器或者时钟选择来设定刷新闻隔。
 - 能通过设定刷新次数（1、2、4、6 和 8）进行集中刷新。
11. 将刷新计数器用作间隔定时器。
 - 能通过比较匹配定时器产生中断请求。

此模块的框图如图 10.1 所示。

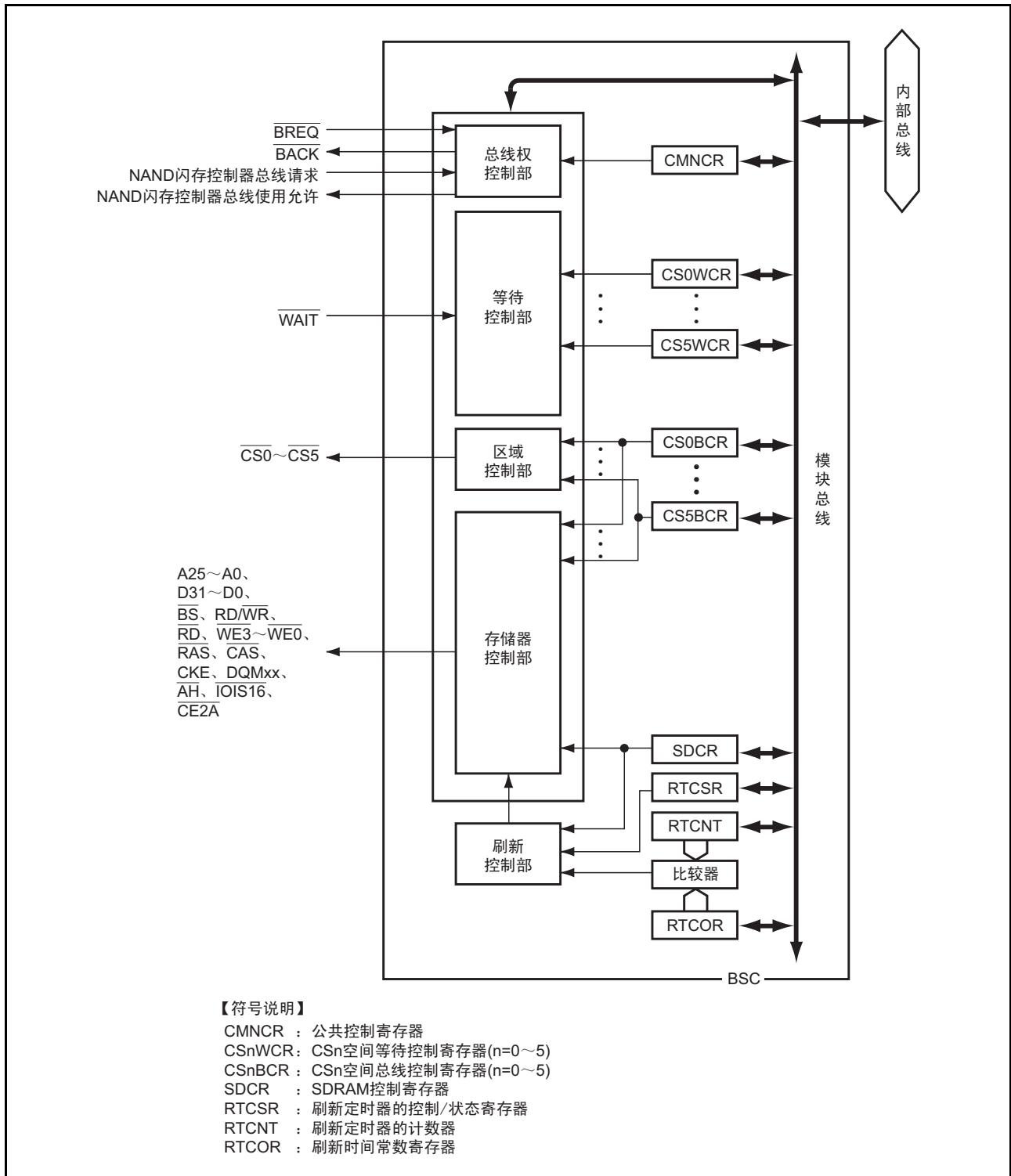


图 10.1 总线状态控制器的框图

10.2 输入 / 输出引脚

引脚结构如表 10.1 所示。

表 10.1 引脚结构

引脚名	输入 / 输出	功能
A25 ~ A0	输出	地址总线
D31 ~ D0	输入 / 输出	数据总线
\overline{BS}	输出	表示总线周期开始的信号
$\overline{CS0} \sim \overline{CS4}$	输出	片选
$\overline{CS5}/\overline{CE1A}$	输出	片选 在使用 PCMCIA 时，此引脚为对应 D7 ~ D0 的 PCMCIA 卡选择信号。
$\overline{CE2A}$	输出	此引脚为对应 D15 ~ D8 的 PCMCIA 卡选择信号。
$\overline{RD}/\overline{WR}$	输出	读写信号 在连接 SDRAM 或者带字节选择的 SRAM 时，将此引脚连接到 \overline{WE} 引脚。
\overline{RD}	输出	读脉冲信号（读数据的输出允许信号） 在使用 PCMCIA 时，此引脚为表示存储器读周期的选通信号。
$\overline{WE3}/\overline{DQMUU}/\overline{ICIORW}/\overline{AH}$	输出	对应 D31 ~ D24 的字节写指示 在连接带字节选择的 SRAM 时，将此引脚连接到字节选择引脚。 在连接 SDRAM 时，此引脚为对应 D31 ~ D24 的选择信号。 在使用 PCMCIA 时，此引脚为表示 I/O 写操作的选通信号。 在使用 MPX-I/O 时，此引脚为保持地址的信号。
$\overline{WE2}/\overline{DQMUL}/\overline{ICIOR}$	输出	对应 D23 ~ D16 的字节写指示 在连接带字节选择的 SRAM 时，将此引脚连接到字节选择引脚。 在连接 SDRAM 时，此引脚为对应 D23 ~ D16 的选择信号。 在使用 PCMCIA 时，此引脚为表示 I/O 读操作的选通信号。
$\overline{WE1}/\overline{DQMLU}/\overline{WE}$	输出	对应 D15 ~ D8 的字节写指示 在连接带字节选择的 SRAM 时，将此引脚连接到字节选择引脚。 在连接 SDRAM 时，此引脚为对应 D15 ~ D8 的选择信号。 在使用 PCMCIA 时，此引脚为表示存储器写周期的选通信号。
$\overline{WE0}/\overline{DQMLL}$	输出	对应 D7 ~ D0 的字节写指示 在连接带字节选择的 SRAM 时，将此引脚连接到字节选择引脚。 在连接 SDRAM 时，此引脚为对应 D7 ~ D0 的选择信号。
\overline{RAS}	输出	在连接 SDRAM 时，将此引脚连接到 \overline{RAS} 引脚。
\overline{CAS}	输出	在连接 SDRAM 时，将此引脚连接到 \overline{CAS} 引脚。
\overline{CKE}	输出	在连接 SDRAM 时，将此引脚连接到 \overline{CKE} 引脚
\overline{WAIT}	输入	外部等待的输入
\overline{BREQ}	输入	总线权请求的输入
\overline{BACK}	输出	总线使用允许的输入
$\overline{IOIS16}$	输入	表示 PCMCIA16 位 I/O 的信号 只在小端法时有效。在大端法时，必须置为低电平。

10.3 区域概要

10.3.1 地址映像

作为体系结构，本 LSI 拥有 32 位地址空间。通过高位分配为高速缓存有效空间、高速缓存无效空间、内部空间（内部 RAM、内部外围模块、保留）。

CS0 ~ CS5 的外部地址空间在内部地址 A29 为“0”时为高速缓存有效，在 A29 为“1”时为高速缓存无效。按各空间指定连接的存储器种类和数据总线宽度。外部地址空间的地址映像如下表所示：

表 10.2 地址映像

地址	空间	存储器种类	高速缓存
H'00000000 ~ H'03FFFFFF	CS0	正规空间、带字节选择的 SRAM、突发 ROM（异步、同步）	有效
H'04000000 ~ H'07FFFFFF	CS1	正规空间、带字节选择的 SRAM	
H'08000000 ~ H'0BFFFFFF	CS2	正规空间、带字节选择的 SRAM、SDRAM	
H'0C000000 ~ H'0FFFFFFF	CS3	正规空间、带字节选择的 SRAM、SDRAM	
H'10000000 ~ H'13FFFFFF	CS4	正规空间、带字节选择的 SRAM、突发 ROM（异步）	
H'14000000 ~ H'17FFFFFF	CS5	正规空间、带字节选择的 SRAM、MPX-I/O、PCMCIA	
H'18000000 ~ H'1FFFFFFF	其他	SPI 多路 I/O 总线空间、大容量内部 RAM、保留区 *	
H'20000000 ~ H'23FFFFFF	CS0	正规空间、带字节选择的 SRAM、突发 ROM（异步、同步）	无效
H'24000000 ~ H'27FFFFFF	CS1	正规空间、带字节选择的 SRAM	
H'28000000 ~ H'2BFFFFFF	CS2	正规空间、带字节选择的 SRAM、SDRAM	
H'2C000000 ~ H'2FFFFFFF	CS3	正规空间、带字节选择的 SRAM、SDRAM	
H'30000000 ~ H'33FFFFFF	CS4	正规空间、带字节选择的 SRAM、突发 ROM（异步）	
H'34000000 ~ H'37FFFFFF	CS5	正规空间、带字节选择的 SRAM、MPX-I/O、PCMCIA	
H'38000000 ~ H'3FFFFFFF	其他	SPI 多路 I/O 总线空间、大容量内部 RAM、保留区 *	
H'40000000 ~ H'FFFFFFF	其他	内部 RAM、内部外围模块、保留区域 *	—

【注】 * 大容量内部 RAM 空间和高速内部 RAM 空间必须存取“47. 内部 RAM”所示的地址。内部外围模块空间必须存取“51. 寄存器一览”所示的地址。不能存取没有记载的地址，否则就不能保证运行。

10.3.2 引导模式、各区域的数据总线宽度、字节序和相关引脚的设定

数据总线宽度和字节序的指定、此模块相关引脚设定的初始值因引导模式而不同。有关引导模式，请参照“4. 引导模式”。

在引导模式 0 和引导模式 1 中，从连接区域 0 的 ROM 启动，因此为不能在 16 位或者 32 位总线宽度和大端法的状态下更改区域 0 的状态。虽然区域 1 ~ 5 的初始状态也和区域 0 相同，但是能通过程序改变总线宽度和字节序的指定。在这些模式中的上电复位后，作为初始功能，只自动选择读区域 0 的 ROM 所需的部分地址总线和部分数据总线、 $\overline{CS0}$ 和 \overline{RD} 引脚功能，而其他功能为通用端口，在通过程序设定引脚功能前不能使用。不能在结束引脚的设定前对区域 0 进行读存取。

在引导模式 2 ~ 5 中，从 NAND 闪存、串行闪存、SD 控制器内部 NAND 闪存以及 MMC 控制器内部 NAND 闪存启动，因此能通过程序从初始状态开始更改区域 0 ~ 5 的状态。另外，不自动设定此模块的相关引脚，所以需要通程序进行设定。不能在结束引脚的设定前存取外部地址空间。

引导模式和各区域的初始状态如表 10.3 所示。

在本章记载的存取波形例子中，出现了 \overline{BS} 、 $\overline{RD}/\overline{WR}$ 、 \overline{WEn} 等引脚，这些是通过通用输入 / 输出端口设定引脚功能的例子。例如，在引导模式 1 中，当使用 16 位总线时需要设定 A1 引脚，当使用 8 位总线时需要设定 A1 引脚和 A0 引脚。

引脚设定的详细内容请参照“48. 通用输入 / 输出端口”。

表 10.3 引导模式和各区域的初始状态

引导模式	项目	区域 0	区域 1 ~ 5
0	数据总线宽度	固定为 16 位总线，不能更改。	初始值为 16 位总线宽度，能通过程序更改。
	字节序的指定	固定为大端法，不能更改。	初始值为大端法，能通过程序更改。
	此模块相关引脚的设定	只自动设定 A20 ~ A1、D15 ~ D0、 $\overline{CS0}$ 、 \overline{RD} 引脚功能，而其他引脚需要通过程序进行设定。	
1	数据总线宽度	固定为 32 位总线，不能更改。	初始值为 32 位总线，能通过程序更改。
	字节序的指定	固定为大端法，不能更改。	初始值为大端法，能通过程序更改。
	此模块相关引脚的设定	只自动设定 A20 ~ A2、D31 ~ D0、 $\overline{CS0}$ 、 \overline{RD} 引脚功能，而其他引脚需要通过程序进行设定。	
2 ~ 5	数据总线宽度	初始值为 32 位总线，能通过程序更改。	
	字节序的指定	初始值为大端法，能通过程序更改。	
	此模块相关引脚的设定	初始值为通用端口功能。要存取外部总线时，需要通过程序设定其他所需的引脚。	

【注】 *1 在引导模式 0 或者引导模式 1 中，要连接使用 A21 以上的地址线的引导 ROM 时，需要在电路板上对 A21 以上的地址线进行下拉处理。

*2 根据使用的存储器种类，可能会限制数据总线宽度。详细内容请参照“10.4.2 CSn 空间总线控制寄存器 (CSnBCR) (n=0 ~ 5)”。

*3 由于将 $\overline{CS4}$ 和 A22 功能分配给同一个引脚，因此不能和使用区域 4 和 A22 以上的地址线的设备一起使用。

10.4 寄存器说明

寄存器结构如表 10.4 所示。

不能在结束和存储器连接的接口设定前存取对应的区域。

表 10.4 寄存器结构

寄存器名	略称	R/W	初始值	地址	存取长度
公共控制寄存器	CMNCR	R/W	H'00001010	H'FFFC0000	32
CS0 空间总线控制寄存器	CS0BCR	R/W	H'36DB0400*	H'FFFC0004	32
CS1 空间总线控制寄存器	CS1BCR	R/W	H'36DB0400*	H'FFFC0008	32
CS2 空间总线控制寄存器	CS2BCR	R/W	H'36DB0400*	H'FFFC000C	32
CS3 空间总线控制寄存器	CS3BCR	R/W	H'36DB0400*	H'FFFC0010	32
CS4 空间总线控制寄存器	CS4BCR	R/W	H'36DB0400*	H'FFFC0014	32
CS5 空间总线控制寄存器	CS5BCR	R/W	H'36DB0400*	H'FFFC0018	32
CS0 空间等待控制寄存器	CS0WCR	R/W	H'00000500	H'FFFC0028	32
CS1 空间等待控制寄存器	CS1WCR	R/W	H'00000500	H'FFFC002C	32
CS2 空间等待控制寄存器	CS2WCR	R/W	H'00000500	H'FFFC0030	32
CS3 空间等待控制寄存器	CS3WCR	R/W	H'00000500	H'FFFC0034	32
CS4 空间等待控制寄存器	CS4WCR	R/W	H'00000500	H'FFFC0038	32
CS5 空间等待控制寄存器	CS5WCR	R/W	H'00000500	H'FFFC003C	32
SDRAM 控制寄存器	SDCR	R/W	H'00000000	H'FFFC004C	32
刷新定时器的控制 / 状态寄存器	RTCSR	R/W	H'00000000	H'FFFC0050	32
刷新定时器的计数器	RTCNT	R/W	H'00000000	H'FFFC0054	32
刷新时间常数寄存器	RTCOR	R/W	H'00000000	H'FFFC0058	32

【注】 * 在引导模式 0 中，为“H'36DB0400”；在引导模式 1～5 中，为“H'36DB0600”。

10.4.1 公共控制寄存器 (CMNCR)

CMNCR 是共同控制各区域的 32 位寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	BLOCK	DPRTY[1:0]	DMAIW[2:0]			DMA IWA	—	—	—	HIZ MEM	HIZ CNT*	
初始值:	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12	—	1	R	保留位 读写值都为“1”。
11	BLOCK	0	R/W	总线锁定位 指定是否接受 $\overline{\text{BREQ}}$ 。 0: 接受 $\overline{\text{BREQ}}$ 。 1: 不接受 $\overline{\text{BREQ}}$ 。
10、9	DPRTY[1:0]	00	R/W	DMA 突发传送优先级 这些位指定在 DMA 突发传送过程中刷新请求和总线权使用请求的优先级。 00: 在 DMA 突发传送过程中接受刷新请求和总线权使用请求。 01: 在 DMA 突发传送过程中接受刷新请求，但是不接受总线权使用请求。 10: 在 DMA 突发传送过程中不接受刷新请求和总线权使用请求。 11: 保留（禁止设定）
8 ~ 6	DMAIW[2:0]	000	R/W	DMA 单地址传送时存取周期之间的等待指定 在 DMA 单地址传送时，这些位指定在从带 DACK 的外部设备输出数据后要插入的空闲周期数。空闲周期的插入方法因后述的 DMAIWA 位的指定而不同。 000: 不插入空闲周期 001: 插入 1 个空闲周期 010: 插入 2 个空闲周期 011: 插入 4 个空闲周期 100: 插入 6 个空闲周期 101: 插入 8 个空闲周期 110: 插入 10 个空闲周期 111: 插入 12 个空闲周期

位	位名	初始值	R/W	说明
5	DMAIWA	0	R/W	DMA 单地址传送时存取周期之间的等待插入方法指定 此位指定空闲周期（由 DMAIW[2:0] 位指定）的插入方法。 当此位为“0”时，在带 DACK 的外部设备驱动数据总线后并且在其他设备（包括本 LSI）驱动数据总线时插入空闲周期。如果带 DACK 的外部设备连续驱动数据总线，就不插入空闲周期。当此位为“1”时，即使连续存取带 DACK 的外部设备，也必须在结束 1 次存取后插入空闲周期。 0: 在带 DACK 的外部设备驱动数据总线后并且在其他设备驱动数据总线时插入空闲周期 1: 总是在存取带 DACK 的外部设备后插入空闲周期
4	—	1	R	保留位 读写值都为“1”。
3、2	—	全 0	R	保留位 读写值都为“0”。
1	HIZMEM	0	R/W	High-Z 存储器控制 此位指定软件待机模式和深度待机模式时的 A25 ~ A0、 \overline{BS} 、 \overline{CSn} 、 $\overline{CE2A}$ 、 $\overline{RD/WR}$ 、 $\overline{WE\bar{n}/DQMxx/AH}$ 、 \overline{RD} 引脚状态。在释放总线时，与此位无关，这些引脚为高阻抗状态。 0: 在软件待机模式和深度待机模式时引脚为高阻抗状态 1: 在软件待机模式和深度待机模式时引脚为驱动状态
0	HIZCNT*	0	R/W	High-Z 控制 此位指定软件待机模式、深度待机模式和释放总线权时的 \overline{CKE} 、 \overline{RAS} 、 \overline{CAS} 引脚状态。 0: 在软件待机模式、深度待机模式和释放总线权时 \overline{CKE} 、 \overline{RAS} 、 \overline{CAS} 引脚为高阻抗状态 1: 在软件待机模式、深度待机模式和释放总线权时 \overline{CKE} 、 \overline{RAS} 、 \overline{CAS} 引脚为驱动状态

【注】* CKIO 的 High-Z 控制请参照“5. 时钟脉冲振荡器”。

10.4.2 CSn 空间总线控制寄存器 (CSnBCR) (n=0 ~ 5)

CSnBCR 设定连接各空间的存储器种类、空间的数据总线宽度以及存取周期之间的等待数。

不能在结束寄存器的初始设定和引脚设定前存取对应的区域的外部存储器。

即使指定为无空闲周期，也可能插入空闲周期。详细内容请参照“10.5.11 存取周期之间的空闲”。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	IWW[2:0]			IWRWD[2:0]			IWRWS[2:0]			IWRRD[2:0]			IWRRS[2:0]		
初始值:	0	0	1	1	0	1	1	0	1	1	0	1	1	0	1	1
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TYPE[2:0]			ENDIAN	BSZ[1:0]		—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	1*	0*	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R

【注】* 在引导模式0中为“B'10”，在引导模式1~5中为“B'11”。

位	位名	初始值	R/W	说明
31	—	0	R	保留位 读写值都为“0”。
30 ~ 28	IWW[2:0]	011	R/W	写 - 读 / 写 - 写周期之间的空闲指定 这些位指定在存取已连接空间的存储器后要插入的空闲周期数。对象周期是写 - 读和写 - 写周期的情况。 000: 不插入空闲周期 001: 插入 1 个空闲周期 010: 插入 2 个空闲周期 011: 插入 4 个空闲周期 100: 插入 6 个空闲周期 101: 插入 8 个空闲周期 110: 插入 10 个空闲周期 111: 插入 12 个空闲周期
27 ~ 25	IWRWD[2:0]	011	R/W	不同空间的读 - 写周期之间的空闲指定 这些位指定在存取已连接空间的存储器后要插入的空闲周期数。对象周期是不同空间的连续读 - 写周期。 000: 不插入空闲周期 001: 插入 1 个空闲周期 010: 插入 2 个空闲周期 011: 插入 4 个空闲周期 100: 插入 6 个空闲周期 101: 插入 8 个空闲周期 110: 插入 10 个空闲周期 111: 插入 12 个空闲周期

位	位名	初始值	R/W	说明
24 ~ 22	IWRWS[2:0]	011	R/W	<p>同一空间的读 - 写周期之间的空闲指定</p> <p>这些位指定在存取已连接空间的存储器后要插入的空闲周期数。对象周期是同一空间的连续读 - 写周期。</p> <p>000: 不插入空闲周期 001: 插入 1 个空闲周期 010: 插入 2 个空闲周期 011: 插入 4 个空闲周期 100: 插入 6 个空闲周期 101: 插入 8 个空闲周期 110: 插入 10 个空闲周期 111: 插入 12 个空闲周期</p>
21 ~ 19	IWRRD[2:0]	011	R/W	<p>不同空间的读 - 读周期之间的空闲指定</p> <p>这些位指定在存取已连接空间的存储器后要插入的空闲周期数。对象周期是不同空间的连续读 - 读周期。</p> <p>000: 不插入空闲周期 001: 插入 1 个空闲周期 010: 插入 2 个空闲周期 011: 插入 4 个空闲周期 100: 插入 6 个空闲周期 101: 插入 8 个空闲周期 110: 插入 10 个空闲周期 111: 插入 12 个空闲周期</p>
18 ~ 16	IWRRS[2:0]	011	R/W	<p>同一空间的读 - 读周期之间的空闲指定</p> <p>这些位指定在存取已连接空间的存储器后要插入的空闲周期数。对象周期是同一空间的连续读 - 读周期。</p> <p>000: 不插入空闲周期 001: 插入 1 个空闲周期 010: 插入 2 个空闲周期 011: 插入 4 个空闲周期 100: 插入 6 个空闲周期 101: 插入 8 个空闲周期 110: 插入 10 个空闲周期 111: 插入 12 个空闲周期</p>
15	—	0	R	<p>保留位</p> <p>读写值都为“0”。</p>

位	位名	初始值	R/W	说明
14 ~ 12	TYPE[2:0]	000	R/W	<p>这些位设定连接空间的存储器种类。</p> <p>000: 正规空间 001: 突发 ROM (异步) 010: MPX-I/O 011: 带字节选择的 SRAM 100: SDRAM 101: PCMCIA 110: 保留 (禁止设定) 111: 突发 ROM (时钟同步)</p> <p>各区域的存储器种类请参照表 10.2。</p> <p>【注】 在引导模式 0 和引导模式 1 中将突发 ROM 连接到 CS0 空间时, 必须在通过使用 CS0WCR 寄存器的突发 ROM 更改为所需的设定后, 将 TYPE[2:0] 设定为突发 ROM。在引导模式 2 ~ 5 的情况下, 必须在设定 CS0BCR 和 CS0WCR 后存取存储器。</p>
11	ENDIAN	0	R/W	<p>字节序指定 此位指定空间的数据排序。</p> <p>0: 以大端法排序 1: 以小端法排序</p> <p>【注】 在引导模式 0 和引导模式 1 中, 不能对区域 0 设定小端法。此时, CS0BCR 的此位的读写值都为“0”。</p>
10、9	BSZ[1:0]	10*	R/W	<p>数据总线的宽度指定 这些位指定空间的数据总线宽度。</p> <p>00: 保留 (禁止设定) 01: 8 位 10: 16 位 11: 32 位</p> <p>在 MPX-I/O 的情况下, 通过地址选择总线宽度。</p> <p>【注】</p> <ol style="list-style-type: none"> 1. 当将区域 5 设定为 MPX-I/O 时, 如果将此位设定为“11”, 就根据 CS5WCR 的 SZSEL 指定的地址选择总线宽度 (8 位或者 16 位)。另外, 在固定总线宽度时, 能设定 8 位或者 16 位。 2. 在引导模式 0 和引导模式 1 中, 忽视 CS0BCR 的 BSZ[1:0] 位的写操作。 3. 当将区域 5 设定为 PCMCIA 空间时, 总线宽度能选择 8 位或者 16 位。 4. 当将区域 2 或者区域 3 设定为 SDRAM 空间时, 总线宽度能选择 16 位或者 32 位。 5. 当将区域 0 设定为时钟同步突发 ROM 空间时, 总线宽度能选择 16 位或者 32 位。
8 ~ 0	—	全 0	R	<p>保留位 读写的值总是 0。</p>

【注】 * 在引导模式 0 中为“B'10”, 在引导模式 1 ~ 5 中为“B'11”。

10.4.3 CSn 空间等待控制寄存器 (CSnWCR) (n=0 ~ 5)

CSnWCR 设定有关存取存储器的各种等待周期。此寄存器的位结构因 CSn 空间总线控制寄存器 (CSnBCR) 设定的存储器种类 (TYPE[2:0]) 而发生以下的变化。必须在存取前设定对象区域, 并且在设定 CSnBCR 寄存器后设定 CSnWCR 寄存器。

(1) 正规空间、带字节选择的 SRAM、MPX-I/O

• CS0WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—*	BAS	—	—	—*	—*
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SW[1:0]		WR[3:0]			WM	—	—	—	—	HW[1:0]		—
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 22	—	全 0	R	保留位 读写值都为“0”。
21	—*	0	R/W	保留位 在正规空间、带字节选择的 SRAM 接口时, 必须将此位置“0”。
20	BAS	0	R/W	带字节选择的 SRAM 字节存取选择 此位设定带字节选择的 SRAM 接口时的 \overline{WEn} 和 $\overline{RD}/\overline{WR}$ 信号时序。 0: \overline{WEn} 在读写时有效, $\overline{RD}/\overline{WR}$ 在写存取周期中有效。 1: \overline{WEn} 在读写存取周期中有效, $\overline{RD}/\overline{WR}$ 在写时有效。
19、18	—	全 0	R	保留位 读写值都为“0”。
17、16	—*	全 0	R/W	保留位 在正规空间、带字节选择的 SRAM 接口时, 必须将此位置“0”。
15 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12、11	SW[1:0]	00	R/W	地址和 $\overline{CS0}$ 有效 $\rightarrow \overline{RD}$ 和 \overline{WEn} 有效的延迟周期数 这些位指定从地址和 $\overline{CS0}$ 有效到 \overline{RD} 和 \overline{WEn} 有效的延迟周期数。 00: 0.5 个周期 01: 1.5 个周期 10: 2.5 个周期 11: 3.5 个周期

位	位名	初始值	R/W	说明
10 ~ 7	WR[3:0]	1010	R/W	<p>存取的等待周期数</p> <p>这些位指定读写存取所需的等待周期数。</p> <p>0000: 不插入等待周期</p> <p>0001: 1 个周期</p> <p>0010: 2 个周期</p> <p>0011: 3 个周期</p> <p>0100: 4 个周期</p> <p>0101: 5 个周期</p> <p>0110: 6 个周期</p> <p>0111: 8 个周期</p> <p>1000: 10 个周期</p> <p>1001: 12 个周期</p> <p>1010: 14 个周期</p> <p>1011: 18 个周期</p> <p>1100: 24 个周期</p> <p>1101: 保留 (禁止设定)</p> <p>1110: 保留 (禁止设定)</p> <p>1111: 保留 (禁止设定)</p>
6	WM	0	R/W	<p>外部等待屏蔽的指定</p> <p>此位指定是将外部等待输入置为有效还是忽视外部等待输入。即使存取等待周期数是“0”，此位的设定也有效。</p> <p>0: 外部等待输入有效</p> <p>1: 忽视外部等待输入</p>
5 ~ 2	—	全 0	R	<p>保留位</p> <p>读写值都为“0”。</p>
1、0	HW[1:0]	00	R/W	<p>\overline{RD} 和 \overline{WEn} 无效 → 地址和 $\overline{CS0}$ 无效的延迟周期数</p> <p>这些位指定从 \overline{RD} 和 \overline{WEn} 无效到地址和 $\overline{CS0}$ 无效的延迟周期数。</p> <p>00: 0.5 个周期</p> <p>01: 1.5 个周期</p> <p>10: 2.5 个周期</p> <p>11: 3.5 个周期</p>

【注】* 在引导模式 0 和引导模式 1 中，如果将突发 ROM 连接到 CS0 空间并且在启动后切换为突发 ROM 接口，就必须在通过 bit21 和 bit20 指定突发数并且通过 bit17 和 bit16 指定突发等待周期数后，设定 CS0BCR 的 TYPE[2:0]。不能给上述以外的保留位写“1”。

• CS1WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	BAS	—	WW[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SW[1:0]		WR[3:0]			WM	—	—	—	—	HW[1:0]		
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 21	—	全 0	R	保留位 读写值都为“0”。
20	BAS	0	R/W	带字节选择的 SRAM 字节存取选择 此位设定带字节选择的 SRAM 接口时的 \overline{WEn} 和 $\overline{RD/WR}$ 信号时序。 0: \overline{WEn} 在读写时有效, $\overline{RD/WR}$ 在写存取周期中有效。 1: \overline{WEn} 在读写存取周期中有效, $\overline{RD/WR}$ 在写时有效。
19	—	0	R	保留位 读写值都为“0”。
18 ~ 16	WW[2:0]	000	R/W	写存取的等待周期数 这些位指定写存取所需的周期数。 000: 和 WR[3:0] 设定的 (读存取等待数) 周期相同 001: 不插入等待周期 010: 1 个周期 011: 2 个周期 100: 3 个周期 101: 4 个周期 110: 5 个周期 111: 6 个周期
15 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12、11	SW[1:0]	00	R/W	地址和 \overline{CSn} 有效 $\rightarrow \overline{RD}$ 和 \overline{WEn} 有效的延迟周期数 这些位指定从地址和 \overline{CSn} 有效到 \overline{RD} 和 \overline{WEn} 有效的延迟周期数。 00: 0.5 个周期 01: 1.5 个周期 10: 2.5 个周期 11: 3.5 个周期

位	位名	初始值	R/W	说明
10 ~ 7	WR[3:0]	1010	R/W	<p>读存取的等待周期数</p> <p>这些位指定读存取所需的等待周期数。</p> <p>0000: 不插入等待周期</p> <p>0001: 1 个周期</p> <p>0010: 2 个周期</p> <p>0011: 3 个周期</p> <p>0100: 4 个周期</p> <p>0101: 5 个周期</p> <p>0110: 6 个周期</p> <p>0111: 8 个周期</p> <p>1000: 10 个周期</p> <p>1001: 12 个周期</p> <p>1010: 14 个周期</p> <p>1011: 18 个周期</p> <p>1100: 24 个周期</p> <p>1101: 保留 (禁止设定)</p> <p>1110: 保留 (禁止设定)</p> <p>1111: 保留 (禁止设定)</p>
6	WM	0	R/W	<p>外部等待屏蔽的指定</p> <p>此位指定是将外部等待输入置为有效还是忽视外部等待输入。即使存取等待周期数是“0”，此位的设定也有效。</p> <p>0: 外部等待输入有效</p> <p>1: 忽视外部等待输入</p>
5 ~ 2	—	全 0	R	<p>保留位</p> <p>读写值都为“0”。</p>
1、0	HW[1:0]	00	R/W	<p>\overline{RD} 和 \overline{WEn} 无效 → 地址和 \overline{CSn} 无效的延迟周期数</p> <p>这些位指定从 \overline{RD} 和 \overline{WEn} 无效到地址和 \overline{CSn} 无效的延迟周期数。</p> <p>00: 0.5 个周期</p> <p>01: 1.5 个周期</p> <p>10: 2.5 个周期</p> <p>11: 3.5 个周期</p>

• CS2WCR、CS3WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	BAS	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	WR[3:0]			WM	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 21	—	全 0	R	保留位 读写值都为“0”。
20	BAS	0	R/W	带字节选择的 SRAM 字节存取选择 此位设定带字节选择的 SRAM 接口时的 \overline{WEn} 和 $\overline{RD/WR}$ 信号时序。 0: \overline{WEn} 在读写时有效, $\overline{RD/WR}$ 在写存取周期中有效。 1: \overline{WEn} 在读写存取周期中有效, $\overline{RD/WR}$ 在写时有效。
19 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 7	WR[3:0]	1010	R/W	存取的等待周期数 这些位指定读写存取所需的等待周期数。 0000: 不插入等待周期 0001: 1 个周期 0010: 2 个周期 0011: 3 个周期 0100: 4 个周期 0101: 5 个周期 0110: 6 个周期 0111: 8 个周期 1000: 10 个周期 1001: 12 个周期 1010: 14 个周期 1011: 18 个周期 1100: 24 个周期 1101: 保留 (禁止设定) 1110: 保留 (禁止设定) 1111: 保留 (禁止设定)
6	WM	0	R/W	外部等待屏蔽的指定 此位指定是将外部等待输入置为有效还是忽视外部等待输入。即使存取等待周期数是“0”, 此位的设定也有效。 0: 外部等待输入有效 1: 忽视外部等待输入
5 ~ 0	—	全 0	R	保留位 读写值都为“0”。

• CS4WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	BAS	—	WW[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SW[1:0]		WR[3:0]			WM	—	—	—	—	HW[1:0]		
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 21	—	全 0	R	保留位 读写值都为“0”。
20	BAS	0	R/W	带字节选择的 SRAM 字节存取选择 此位设定带字节选择的 SRAM 接口时的 \overline{WEn} 和 $\overline{RD/WR}$ 信号时序。 0: \overline{WEn} 在读写时有效, $\overline{RD/WR}$ 在写存取周期中有效。 1: \overline{WEn} 在读写存取周期中有效, $\overline{RD/WR}$ 在写时有效。
19	—	0	R	保留位 读写值都为“0”。
18 ~ 16	WW[2:0]	000	R/W	写存取的等待周期数 这些位指定写存取所需的等待周期数。 000: 和 WR[3:0] 设定的 (读存取等待数) 周期相同 001: 不插入等待周期 010: 1 个周期 011: 2 个周期 100: 3 个周期 101: 4 个周期 110: 5 个周期 111: 6 个周期
15 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12、11	SW[1:0]	00	R/W	地址和 $\overline{CS4}$ 有效 $\rightarrow \overline{RD}$ 和 \overline{WEn} 有效的延迟周期数 这些位指定从地址和 $\overline{CS4}$ 有效到 \overline{RD} 和 \overline{WEn} 有效的延迟周期数。 00: 0.5 个周期 01: 1.5 个周期 10: 2.5 个周期 11: 3.5 个周期

位	位名	初始值	R/W	说明
10 ~ 7	WR[3:0]	1010	R/W	<p>读存取的等待周期数</p> <p>这些位指定读存取所需的等待周期数。</p> <p>0000: 不插入等待周期</p> <p>0001: 1 个周期</p> <p>0010: 2 个周期</p> <p>0011: 3 个周期</p> <p>0100: 4 个周期</p> <p>0101: 5 个周期</p> <p>0110: 6 个周期</p> <p>0111: 8 个周期</p> <p>1000: 10 个周期</p> <p>1001: 12 个周期</p> <p>1010: 14 个周期</p> <p>1011: 18 个周期</p> <p>1100: 24 个周期</p> <p>1101: 保留 (禁止设定)</p> <p>1110: 保留 (禁止设定)</p> <p>1111: 保留 (禁止设定)</p>
6	WM	0	R/W	<p>外部等待屏蔽的指定</p> <p>此位指定是将外部等待输入置为有效还是忽视外部等待输入。即使存取等待周期数是“0”，此位的设定也有效。</p> <p>0: 外部等待输入有效</p> <p>1: 忽视外部等待输入</p>
5 ~ 2	—	全 0	R	<p>保留位</p> <p>读写值都为“0”。</p>
1、0	HW[1:0]	00	R/W	<p>\overline{RD} 和 \overline{WEn} 无效 → 地址和 $\overline{CS4}$ 无效的延迟周期数</p> <p>这些位指定从 \overline{RD} 和 \overline{WEn} 无效到地址和 $\overline{CS4}$ 无效的延迟周期数。</p> <p>00: 0.5 个周期</p> <p>01: 1.5 个周期</p> <p>10: 2.5 个周期</p> <p>11: 3.5 个周期</p>

• CS5WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	SZSEL	MPXW/ BAS	—	WW[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SW[1:0]		WR[3:0]			WM	—	—	—	—	HW[1:0]		
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明																		
31 ~ 22	—	全 0	R	保留位 读写值都为“0”。																		
21	SZSEL	0	R/W	MPX-I/O 接口总线宽度的指定位 此位指定在将 CS5BCR 的 BSZ[1:0] 设定为“11”时要选择总线宽度的地址。此设定只在将区域 5 设定为 MPX-I/O 时有效。 0: 通过地址 A14 选择总线宽度 1: 通过地址 A21 选择总线宽度 SZSEL 位和通过 A14、A21 选择总线宽度的关系如下所示: <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>SZSEL</th><th>A14</th><th>A21</th><th>说明</th></tr> </thead> <tbody> <tr> <td rowspan="2">0</td><td>0</td><td>没有影响</td><td>8位总线宽度</td></tr> <tr> <td>1</td><td>没有影响</td><td>16位总线宽度</td></tr> <tr> <td rowspan="2">1</td><td>没有影响</td><td>0</td><td>8位总线宽度</td></tr> <tr> <td>没有影响</td><td>1</td><td>16位总线宽度</td></tr> </tbody> </table>	SZSEL	A14	A21	说明	0	0	没有影响	8位总线宽度	1	没有影响	16位总线宽度	1	没有影响	0	8位总线宽度	没有影响	1	16位总线宽度
SZSEL	A14	A21	说明																			
0	0	没有影响	8位总线宽度																			
	1	没有影响	16位总线宽度																			
1	没有影响	0	8位总线宽度																			
	没有影响	1	16位总线宽度																			
20	MPXW	0	R/W	MPX-I/O 接口地址的等待 此设定只在将区域 5 设定为 MPX-I/O 时有效。此位设定 MPX-I/O 接口地址周期的插入等待。 0: 不插入等待 1: 插入 1 个周期等待																		
	BAS	0	R/W	带字节选择的 SRAM 字节存取选择 此设定只在将区域 5 设定为带字节选择的 SRAM 时有效。 此位设定带字节选择的 SRAM 接口时的 \overline{WEn} 和 $\overline{RD/WR}$ 信号时序。 0: \overline{WEn} 在读写时有效, $\overline{RD/WR}$ 在写存取周期中有效。 1: \overline{WEn} 在读写存取周期中有效, $\overline{RD/WR}$ 在写时有效。																		
19	—	0	R	保留位 读写值都为“0”。																		
18 ~ 16	WW[2:0]	000	R/W	写存取的等待周期数 这些位指定写存取所需的周期数。 000: 和 WR[3:0] 设定的 (读存取等待数) 周期相同 001: 不插入等待周期 010: 1 个周期 011: 2 个周期 100: 3 个周期 101: 4 个周期 110: 5 个周期 111: 6 个周期																		

位	位名	初始值	R/W	说明
15 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12、11	SW[1:0]	00	R/W	地址和 $\overline{CS5}$ 有效 → \overline{RD} 和 \overline{WEn} 有效的延迟周期数 这些位指定从地址和 $\overline{CS5}$ 有效到 \overline{RD} 和 \overline{WEn} 有效的延迟周期数。 00: 0.5 个周期 01: 1.5 个周期 10: 2.5 个周期 11: 3.5 个周期
10 ~ 7	WR[3:0]	1010	R/W	读存取的等待周期数 这些位指定读存取所需的等待周期数。 0000: 不插入等待周期 0001: 1 个周期 0010: 2 个周期 0011: 3 个周期 0100: 4 个周期 0101: 5 个周期 0110: 6 个周期 0111: 8 个周期 1000: 10 个周期 1001: 12 个周期 1010: 14 个周期 1011: 18 个周期 1100: 24 个周期 1101: 保留 (禁止设定) 1110: 保留 (禁止设定) 1111: 保留 (禁止设定)
6	WM	0	R/W	外部等待屏蔽的指定 此位指定是将外部等待输入置为有效还是忽视外部等待输入。即使存取等待周期数是“0”，此位的设定也有效。 0: 外部等待输入有效 1: 忽视外部等待输入
5 ~ 2	—	全 0	R	保留位 读写值都为“0”。
1、0	HW[1:0]	00	R/W	\overline{RD} 和 \overline{WEn} 无效 → 地址和 $\overline{CS5}$ 无效的延迟周期数 这些位指定从 \overline{RD} 和 \overline{WEn} 无效到地址和 $\overline{CS5}$ 无效的延迟周期数。 00: 0.5 个周期 01: 1.5 个周期 10: 2.5 个周期 11: 3.5 个周期

(2) 突发 ROM (异步)

• CS0WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	BST[1:0]		—	—	BW[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	W[3:0]			WM	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

位	位名	初始值	R/W	说明																		
31 ~ 22	—	全 0	R	保留位 读写值都为“0”。																		
21、20	BST[1:0]	00	R/W	突发数指定 这些位指定在发生 16 字节存取时的突发数。保留 BST[1:0] 为“B'11”的设定，因此不能设定。 <table border="1" data-bbox="769 949 1305 1223"> <thead> <tr> <th>总线宽度</th><th>BST[1:0]</th><th>突发数</th></tr> </thead> <tbody> <tr> <td rowspan="2">8位</td><td>00</td><td>16个突发×1次</td></tr> <tr> <td>01</td><td>4个突发×4次</td></tr> <tr> <td rowspan="3">16位</td><td>00</td><td>8个突发×1次</td></tr> <tr> <td>01</td><td>2突发×4次</td></tr> <tr> <td>10</td><td>4-4或者2-4-2个突发</td></tr> <tr> <td>32位</td><td>xx</td><td>4个突发×1次</td></tr> </tbody> </table>	总线宽度	BST[1:0]	突发数	8位	00	16个突发×1次	01	4个突发×4次	16位	00	8个突发×1次	01	2突发×4次	10	4-4或者2-4-2个突发	32位	xx	4个突发×1次
总线宽度	BST[1:0]	突发数																				
8位	00	16个突发×1次																				
	01	4个突发×4次																				
16位	00	8个突发×1次																				
	01	2突发×4次																				
	10	4-4或者2-4-2个突发																				
32位	xx	4个突发×1次																				
19、18	—	全 0	R	保留位 读写值都为“0”。																		
17、16	BW[1:0]	00	R/W	突发的等待周期数 这些位指定要插入到突发存取时的第 2 次以后存取周期的等待周期数。 00: 不插入等待周期 01: 1 个周期 10: 2 个周期 11: 3 个周期																		
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。																		

位	位名	初始值	R/W	说明
10 ~ 7	W[3:0]	1010	R/W	<p>存取的等待周期数</p> <p>这些位指定要插入到第 1 次存取周期的等待周期数。</p> <p>0000: 不插入等待周期</p> <p>0001: 1 个周期</p> <p>0010: 2 个周期</p> <p>0011: 3 个周期</p> <p>0100: 4 个周期</p> <p>0101: 5 个周期</p> <p>0110: 6 个周期</p> <p>0111: 8 个周期</p> <p>1000: 10 个周期</p> <p>1001: 12 个周期</p> <p>1010: 14 个周期</p> <p>1011: 18 个周期</p> <p>1100: 24 个周期</p> <p>1101: 保留 (禁止设定)</p> <p>1110: 保留 (禁止设定)</p> <p>1111: 保留 (禁止设定)</p>
6	WM	0	R/W	<p>外部等待屏蔽的指定</p> <p>此位指定是将外部等待输入置为有效还是忽视外部等待输入。即使存取等待周期数是“0”，此位的设定也有效。</p> <p>0: 外部等待输入有效</p> <p>1: 忽视外部等待输入</p>
5 ~ 0	—	全 0	R	<p>保留位</p> <p>读写的值总是 0。</p>

• CS4WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	BST[1:0]		—	—	BW[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SW[1:0]		W[3:0]			WM	—	—	—	—	HW[1:0]		
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明																		
31 ~ 22	—	全 0	R	保留位 读写值都为“0”。																		
21、20	BST[1:0]	00	R/W	突发数指定 这些位指定在发生 16 字节存取时的突发数。保留 BST[1:0] 为“B'11”的设定，因此不能设定。 <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>总线宽度</th><th>BST[1:0]</th><th>突发数</th></tr> </thead> <tbody> <tr> <td rowspan="2">8位</td><td>00</td><td>16个突发×1次</td></tr> <tr> <td>01</td><td>4个突发×4次</td></tr> <tr> <td rowspan="3">16位</td><td>00</td><td>8个突发×1次</td></tr> <tr> <td>01</td><td>2个突发×4次</td></tr> <tr> <td>10</td><td>4-4或者2-4-2个突发</td></tr> <tr> <td>32位</td><td>xx</td><td>4个突发×1次</td></tr> </tbody> </table>	总线宽度	BST[1:0]	突发数	8位	00	16个突发×1次	01	4个突发×4次	16位	00	8个突发×1次	01	2个突发×4次	10	4-4或者2-4-2个突发	32位	xx	4个突发×1次
总线宽度	BST[1:0]	突发数																				
8位	00	16个突发×1次																				
	01	4个突发×4次																				
16位	00	8个突发×1次																				
	01	2个突发×4次																				
	10	4-4或者2-4-2个突发																				
32位	xx	4个突发×1次																				
19、18	—	全 0	R	保留位 读写值都为“0”。																		
17、16	BW[1:0]	00	R/W	突发的等待周期数 这些位指定要插入到突发存取时的第 2 次以后存取周期的等待周期数。 00: 不插入等待周期 01: 1 个周期 10: 2 个周期 11: 3 个周期																		
15 ~ 13	—	全 0	R	保留位 读写值都为“0”。																		
12、11	SW[1:0]	00	R/W	地址和 CS4 有效 → RD 和 WEn 有效的延迟周期数 这些位指定从地址和 CS4 有效到 RD 和 WEn 有效的延迟周期数。 00: 0.5 个周期 01: 1.5 个周期 10: 2.5 个周期 11: 3.5 个周期																		

位	位名	初始值	R/W	说明
10 ~ 7	W[3:0]	1010	R/W	<p>存取的等待周期数</p> <p>这些位指定要插入到第 1 次存取周期的等待周期数。</p> <p>0000: 不插入等待周期</p> <p>0001: 1 个周期</p> <p>0010: 2 个周期</p> <p>0011: 3 个周期</p> <p>0100: 4 个周期</p> <p>0101: 5 个周期</p> <p>0110: 6 个周期</p> <p>0111: 8 个周期</p> <p>1000: 10 个周期</p> <p>1001: 12 个周期</p> <p>1010: 14 个周期</p> <p>1011: 18 个周期</p> <p>1100: 24 个周期</p> <p>1101: 保留 (禁止设定)</p> <p>1110: 保留 (禁止设定)</p> <p>1111: 保留 (禁止设定)</p>
6	WM	0	R/W	<p>外部等待屏蔽的指定</p> <p>此位指定是将外部等待输入置为有效还是忽视外部等待输入。即使存取等待周期数是“0”，此位的设定也有效。</p> <p>0: 外部等待输入有效</p> <p>1: 忽视外部等待输入</p>
5 ~ 2	—	全 0	R	<p>保留位</p> <p>读写值都为“0”。</p>
1、0	HW[1:0]	00	R/W	<p>\overline{RD} 和 \overline{WEn} 无效 → 地址和 $\overline{CS4}$ 无效的延迟周期数</p> <p>这些位指定从 \overline{RD} 和 \overline{WEn} 无效到地址和 $\overline{CS4}$ 无效的延迟周期数。</p> <p>00: 0.5 个周期</p> <p>01: 1.5 个周期</p> <p>10: 2.5 个周期</p> <p>11: 3.5 个周期</p>

(3) SDRAM*

• CS2WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	A2CL[1:0]	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10	—	1	R	保留位 读写值都为“1”。
9	—	0	R	保留位 读写值都为“0”。
8、7	A2CL[1:0]	10	R/W	区域 2 的 CAS 等待时间 这些位指定区域 2 的 CAS 等待时间。 00: 1 个周期 01: 2 个周期 10: 3 个周期 11: 4 个周期
6 ~ 0	—	全 0	R	保留位 读写值都为“0”。

【注】 * 当只将 1 个区域连接 SDRAM 时，必须将区域 3 设定为 SDRAM，并且将区域 2 设定为正规空间或者带字节选择的 SRAM。

• CS3WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	WTRP[1:0]*	—	WTRCD[1:0]*	—	A3CL[1:0]	—	—	—	—	—	TRWL[1:0]*	—	—	WTRC[1:0]*	—
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R	R/W	R/W	R	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W

【注】* 当将区域2和区域3都设定为SDRAM时，WTRP[1:0]、WTRCD[1:0]、TRWL[1:0]、WTRC[1:0]位的设定是共用的。

位	位名	初始值	R/W	说明
31 ~ 15	—	全 0	R	保留位 读写值都为“0”。
14、13	WTRP[1:0]*	00	R/W	预充电结束的等待周期数 指定以下情况的等待预充电结束的最小周期数。 <ul style="list-style-type: none"> 从启动自动预充电到对同一存储体发行 ACTV 命令 从发行 PRE/PALL 命令到对同一存储体发行 ACTV 命令 在转移到掉电模式 / 深度掉电模式前 从自动刷新时的发行 PALL 命令到发行 REF 命令 从自刷新时的发行 PALL 命令到发行 SELF 命令 区域 2 和区域 3 为共用的设定。 00: 不插入等待周期 01: 1 个周期 10: 2 个周期 11: 3 个周期
12	—	0	R	保留位 读写值都为“0”。
11、10	WTRCD[1:0]*	01	R/W	ACTV 命令 → READ(A)/WRIT(A) 命令之间的等待周期数 这些位指定从发行 ACTV 命令到发行 READ(A)/WRIT(A) 命令的最小等待周期数。 区域 2 和区域 3 为共用的设定。 00: 不插入等待周期 01: 1 个周期 10: 2 个周期 11: 3 个周期
9	—	0	R	保留位 读写值都为“0”。
8、7	A3CL[1:0]	10	R/W	区域 3 的 CAS 的等待时间 这些位指定区域 3 的 CAS 等待时间。 00: 1 个周期 01: 2 个周期 10: 3 个周期 11: 4 个周期
6、5	—	全 0	R	保留位 读写值都为“0”。

位	位名	初始值	R/W	说明
4、3	TRWL[1:0]*	00	R/W	<p>预充电启动的等待周期数 指定以下情况的等待预充电启动的最小周期数。</p> <ul style="list-style-type: none"> 从本 LSI 发行 WRITA 命令到在 SDRAM 内启动自动预充电的周期数 这是从发行 WRITEA 命令到对同一存储体发行 ACTV 命令的周期数。 另外，必须通过各 SDRAM 的数据表确认在 SDRAM 内从接受 WRITA 命令到启动自动预充电需要几个周期。该周期数不能超过此位指定的周期数。 本 LSI 从发行 WRIT 命令到发行 PRE 命令的周期数 这是在存储体激活模式中存取同一存储体内的不同行地址的情况。 <p>区域 2 和区域 3 为共用的设定。 00: 不插入等待周期 01: 1 个周期 10: 2 个周期 11: 3 个周期</p>
2	—	0	R	<p>保留位 读写值都为“0”。</p>
1、0	WTRC[1:0]*	00	R/W	<p>REF 命令 / 自刷新解除 →ACTV/REF/MRS 命令之间的空闲周期数 指定以下情况的命令之间的最小空闲周期数。</p> <ul style="list-style-type: none"> 从发行 REF 命令到发行 ACTV/REF/MRS 命令 从解除自刷新到发行 ACTV/REF/MRS 命令 <p>区域 2 和区域 3 为共用的设定。 00: 2 个周期 01: 3 个周期 10: 5 个周期 11: 8 个周期</p>

【注】 * 当将区域2和区域3都设定为SDRAM时，WTRP[1:0]、WTRCD[1:0]、TRWL[1:0]、WTRC[1:0]位为共用的设定。
当只将区域1连接SDRAM时，必须将区域3设定为SDRAM，并且将区域2设定为正规空间或者带字节选择的SRAM。

(4) PCMCIA

• CS5WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	SA[1:0]	—	—	—	—	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TED[3:0]			PCW[3:0]			WM	—	—	TEH[3:0]			—	—	—
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 22	—	全 0	R	保留位 读写值都为“0”。
21、20	SA[1:0]	00	R/W	空间属性的指定 在设定 PCMCIA 接口时，通过这些位指定存储卡接口或者 I/O 卡接口。 • SA1 0: 将 A25 为“1”的空间指定为存储卡接口 1: 将 A25 为“1”的空间指定为 I/O 卡接口 • SA0 0: 将 A25 为“0”的空间指定为存储卡接口 1: 将 A25 为“0”的空间指定为 I/O 卡接口
19 ~ 15	—	全 0	R	保留位 读写值都为“0”。
14 ~ 11	TED[3:0]	0000	R/W	地址 $\overline{\text{RD}}/\overline{\text{WE}}$ 有效的延迟 这些位设定从 PCMCIA 接口的地址输出到存储器卡的 $\overline{\text{RD}}/\overline{\text{WE}}$ 有效或者 I/O 卡的 ICIORD/ICIOWR 有效的延迟时间。 0000: 0.5 个周期 0001: 1.5 个周期 0010: 2.5 个周期 0011: 3.5 个周期 0100: 4.5 个周期 0101: 5.5 个周期 0110: 6.5 个周期 0111: 7.5 个周期 1000: 8.5 个周期 1001: 9.5 个周期 1010: 10.5 个周期 1011: 11.5 个周期 1100: 12.5 个周期 1101: 13.5 个周期 1110: 14.5 个周期 1111: 15.5 个周期

位	位名	初始值	R/W	说明
10 ~ 7	PCW[3:0]	1010	R/W	<p>存取等待周期数</p> <p>这些位设定要插入的等待状态数。</p> <p>0000: 3 个周期 0001: 6 个周期 0010: 9 个周期 0011: 12 个周期 0100: 15 个周期 0101: 18 个周期 0110: 22 个周期 0111: 26 个周期 1000: 30 个周期 1001: 33 个周期 1010: 36 个周期 1011: 38 个周期 1100: 52 个周期 1101: 60 个周期 1110: 64 个周期 1111: 80 个周期</p>
6	WM	0	R/W	<p>外部等待屏蔽的指定</p> <p>此位指定是将外部等待输入置为有效还是忽视外部等待输入。即使存取等待周期数是“0”，此位的设定也有效。</p> <p>0: 外部等待输入有效 1: 忽视外部等待输入</p>
5、4	—	全 0	R	<p>保留位</p> <p>读写值都为“0”。</p>
3 ~ 0	TEH[3:0]	0000	R/W	<p>$\overline{RD}/\overline{WE}$ 无效 - 地址延迟</p> <p>这些位设定 PCMCIA 接口的存储器卡的 $\overline{RD}/\overline{WE}$ 无效或者 I/O 卡的 $\overline{ICIORD}/\overline{ICIOWR}$ 无效后的地址保持时间。</p> <p>0000: 0.5 个周期 0001: 1.5 个周期 0010: 2.5 个周期 0011: 3.5 个周期 0100: 4.5 个周期 0101: 5.5 个周期 0110: 6.5 个周期 0111: 7.5 个周期 1000: 8.5 个周期 1001: 9.5 个周期 1010: 10.5 个周期 1011: 11.5 个周期 1100: 12.5 个周期 1101: 13.5 个周期 1110: 14.5 个周期 1111: 15.5 个周期</p>

(5) 突发 ROM (时钟同步)

• CS0WCR

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BW[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	W[3:0]			WM	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 18	—	全 0	R	保留位 读写值都为“0”。
17、16	BW[1:0]	00	R/W	突发的等待周期数 这些位指定要插入到突发存取时的第 2 次以后存取周期的等待周期数。 00: 不插入等待周期 01: 1 个周期 10: 2 个周期 11: 3 个周期
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 7	W[3:0]	1010	R/W	存取的等待周期数 这些位指定要插入到第 1 次存取周期的等待周期数。 0000: 不插入等待周期 0001: 1 个周期 0010: 2 个周期 0011: 3 个周期 0100: 4 个周期 0101: 5 个周期 0110: 6 个周期 0111: 8 个周期 1000: 10 个周期 1001: 12 个周期 1010: 14 个周期 1011: 18 个周期 1100: 24 个周期 1101: 保留 (禁止设定) 1110: 保留 (禁止设定) 1111: 保留 (禁止设定)
6	WM	0	R/W	外部等待屏蔽的指定 此位指定是将外部等待输入置为有效还是忽视外部等待输入。即使存取等待周期数是“0”，此位的设定也有效。 0: 外部等待输入有效 1: 忽视外部等待输入
5 ~ 0	—	全 0	R	保留位 读写值都为“0”。

10.4.4 SDRAM 控制寄存器 (SDCR)

SDCR 指定 SDRAM 的刷新方法、存取方法以及要连接的 SDRAM 种类。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	A2ROW[1:0]	—	—	A2COL[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	DEEP	—	RFSH	RMODE	PDOWN	BACTV	—	—	—	A3ROW[1:0]	—	—	A3COL[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 21	—	全 0	R	保留位 读写值都为“0”。
20、19	A2ROW[1:0]	00	R/W	区域 2 行地址的位数 这些位指定区域 2 行地址的位数。 00: 11 位 01: 12 位 10: 13 位 11: 保留 (禁止设定)
18	—	0	R	保留位 读写值都为“0”。
17、16	A2COL[1:0]	00	R/W	区域 2 列地址的位数 这些位指定区域 2 列地址的位数。 00: 8 位 01: 9 位 10: 10 位 11: 保留 (禁止设定)
15、14	—	全 0	R	保留位 读写值都为“0”。
13	DEEP	0	R/W	深度掉电模式 此位只对低功耗 SDRAM 有效。如果在将此位置“1”的状态下将 RFSH 位或者 RMODE 位置“1”，发行深度掉电入口命令，使低功率 SDRAM 转移到深度掉电模式。 0: 自刷新模式 1: 深度掉电模式
12	—	0	R	保留位 读写值都为“0”。
11	RFSH	0	R/W	刷新控制 此位指定是否刷新 SDRAM。 0: 不刷新 1: 刷新

位	位名	初始值	R/W	说明
10	RMODE	0	R/W	刷新控制 在 RFSH 位为“1”时，此位指定是进行自动刷新还是进行自刷新。当将 RFSH 位和此位都设定为“1”时，就在设定后立即进入自刷新模式。 如果将 RFSH 位设定为“1”而将此位设定为“0”，就根据 RTCSR、RTCNT 和 RTCOR 寄存器设定的内容进行自动刷新。 0: 自动刷新 1: 自刷新
9	PDOWN	0	R/W	掉电模式 此位指定在结束存取 SDRAM 后是否将 SDRAM 置为掉电模式。如果将此位设定为“1”时，就在结束存取后将 CKE 引脚置为低电平，使 SDRAM 进入掉电模式。 0: 在结束存取后，不将 SDRAM 置为掉电模式。 1: 在结束存取后，将 SDRAM 置为掉电模式。
8	BACTV	0	R/W	存储体激活模式 此位指定是通过自动预充电模式（使用 READA 命令和 WRITA 命令）还是通过存储体激活模式（使用 READ 命令和 WRIT 命令）进行存取。 0: 自动预充电模式（使用 READA 命令和 WRITA 命令） 1: 存储体激活模式（使用 READ 命令和 WRIT 命令） 【注】 只有区域 3 才能设定存储体激活模式。当将区域 2 和区域 3 都设定为 SDRAM 时，必须设定为自动预充电模式。
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4、3	A3ROW[1:0]	00	R/W	区域 3 行地址的位数 这些位指定区域 3 行地址的位数。 00: 11 位 01: 12 位 10: 13 位 11: 保留（禁止设定）
2	—	0	R	保留位 读写值都为“0”。
1、0	A3COL[1:0]	00	R/W	区域 3 列地址的位数 这些位指定区域 3 列地址的位数。 00: 8 位 01: 9 位 10: 10 位 11: 保留（禁止设定）

10.4.5 刷新定时器的控制 / 状态寄存器 (RTCSR)

RTCSR 进行有关刷新 SDRAM 的各种设定。

在写 RTCSR 时，必须通过将写数据的高 16 位设定为“H'A55A”来解除写保护。

必须注意：因为对刷新定时器的计数器 (RTCNT) 进行递增计数的时钟只有在上电复位时才进行相位的匹配，所以从 CKS[2:0] 被设定为“B'000”以外的值开始到使定时器运行后最初设定比较匹配标志的期间有误差。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CMF	CMIE	CKS[2:0]			RRC[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 8	—	全 0	R	保留位 读写值都为“0”。
7	CMF	0	R/W	比较匹配标志 此位是表示刷新定时器的计数器 (RTCNT) 和刷新时间常数寄存器 (RTCOR) 的值相同的状态标志。在以下项目下进行置位和清除。 0: 清除条件: 在 CMF 为“1”的状态下, 读 RTCSR 后给 CMF 写“0”时 1: 置位条件: 当 RTCNT 和 RTCOR 相等时
6	CMIE	0	R/W	比较匹配的中断允许 此位设定在 RTCSR 的 CMF 位被置“1”时是否允许由 CMF 产生的中断请求。 0: 禁止由 CMF 产生的中断请求 1: 允许由 CMF 产生的中断请求
5 ~ 3	CKS[2:0]	000	R/W	时钟选择 这些位选择对刷新定时器的计数器 (RTCNT) 进行递增计数的时钟。 000: 停止递增计数 001: CKIO ϕ /4 010: CKIO ϕ /16 011: CKIO ϕ /64 100: CKIO ϕ /256 101: CKIO ϕ /1024 110: CKIO ϕ /2048 111: CKIO ϕ /4096

位	位名	初始值	R/W	说明
2 ~ 0	RRC[2:0]	000	R/W	刷新次数 这些位指定在刷新定时器的计数器（RTCNT）和刷新时间常数寄存器（RTCOR）的值相同后发生刷新请求时的连续刷新次数。通过此功能可延长发生刷新的周期。 000: 1 次 001: 2 次 010: 4 次 011: 6 次 100: 8 次 101: 保留（禁止设定） 110: 保留（禁止设定） 111: 保留（禁止设定）

10.4.6 刷新定时器的计数器（RTCNT）

RTCNT 是 8 位计数器，对 RTCSR 的 CKS[2:0] 位选择的时钟进行递增计数。如果 RTCNT 和 RTCOR 的值相同，就将 RTCNT 清“0”。当递增计数到 255 时，恢复为“0”。在写 RTCNT 时，必须通过将写数据的高 16 位设定为“H'A55A”来解除写保护。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—								
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 8	—	全 0	R	保留位 读取值都为“0”。
7 ~ 0		全 0	R/W	8 位计数器

10.4.7 刷新时间常数寄存器 (RTCOR)

RTCOR是8位寄存器。如果RTCOR和RTCNT的值相同,就将RTCSR的CMF位置“1”,并且将RTCNT清“0”。

如果SDCR的RFSH位已被置“1”,就通过此匹配信号产生刷新请求。刷新请求被保持到实际刷新为止。如果在下一个匹配信号前还没有处理刷新请求,前一个请求就无效。

如果RTCSR的CMIE位已被置“1”,就通过该匹配信号产生中断请求。将中断请求连续输出到RTCSR的CMF位被清除为止。CMF位的清除只影响中断而不会清除刷新请求。因此,能设定为边进行刷新边使用中 断对刷新请求的次数进行计数等,也能同时设定刷新和间隔定时器的中断。

在写RTCOR时,必须通过将写数据的高16位设定为“H'A55A”来解除写保护。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—								
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 8	—	全 0	R	保留位 读取值都为“0”。
7 ~ 0		全 0	R/W	8 位寄存器

10.5 运行说明

10.5.1 字节序 / 存取长度和数据调整

对于字节数据排序方法，本 LSI 支持高位字节（MSB）为地址 0 的大端法并且也支持低位字节（LSB）为地址 0 的小端法。在上电复位后，全区域的初始状态为大端法。在不存取对象空间时，能通过设定 CSnBCR 寄存器来改变字节序。

正规存储器和带字节选择的 SRAM 能选择 8 位、16 位或者 32 位的数据总线，SDRAM 能选择 16 位或者 32 位的数据总线，PCMCIA 接口能选择 8 位或者 16 位的数据总线，MPX-I/O 为 8 位或者 16 位固定的数据总线，或者根据存取地址为 8 位或者 16 位可变的数据总线。

根据引导模式，字节序的指定和数据总线宽度有限制。详细内容请参照“10.3.2 引导模式、各区域的数据总线宽度、字节序和相关引脚的设置”

根据各设备的数据总线宽度进行数据的调整。因此，当要从 8 位宽度的设备读长字数据时，需要读 4 次。本 LSI 在各接口之间自动进行数据的调整以及数据长度的转换。

设备的数据宽度和存取单位的关系如表 10.5 ~ 表 10.10 所示。必须注意：32 位总线宽度和 16 位总线宽度时的选通信号的对应地址因大端法和小端法而不同。例如，在 16 位总线宽度的情况下，当为大端法时， $\overline{WE1}$ 指向地址 0；当为小端法时， $\overline{WE0}$ 指向地址 0。

如果在取指令时 32 位存取和 16 位存取混在一起，就难以配置到小端法区，因此必须从大端法区执行指令。

表 10.5 大端法的 32 位外部设备的存取和数据调整

操作	数据总线				选通信号			
	D31 ~ D24	D23 ~ D16	D15 ~ D8	D7 ~ D0	$\overline{WE3}$ 、 DQMUU	$\overline{WE2}$ 、 DQMUL	$\overline{WE1}$ 、 DQMLU	$\overline{WE0}$ 、 DQMLL
地址 0 的字节存取	数据 7 ~ 0	—	—	—	有效	—	—	—
地址 1 的字节存取	—	数据 7 ~ 0	—	—	—	有效	—	—
地址 2 的字节存取	—	—	数据 7 ~ 0	—	—	—	有效	—
地址 3 的字节存取	—	—	—	数据 7 ~ 0	—	—	—	有效
地址 0 的字存取	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效	—	—
地址 2 的字存取	—	—	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效
地址 0 的长字存取	数据 31 ~ 24	数据 23 ~ 16	数据 15 ~ 8	数据 7 ~ 0	有效	有效	有效	有效

表 10.6 大端法的 16 位外部设备的存取和数据调整

操作		数据总线				选通信号			
		D31 ~ D24	D23 ~ D16	D15 ~ D8	D7 ~ D0	$\overline{\text{WE3}}$ 、 DQMUU	$\overline{\text{WE2}}$ 、 DQMUL	$\overline{\text{WE1}}$ 、 DQMLU	$\overline{\text{WE0}}$ 、 DQMLL
地址 0 的字节存取		—	—	数据 7 ~ 0	—	—	有效	—	
地址 1 的字节存取		—	—	—	数据 7 ~ 0	—	—	有效	
地址 2 的字节存取		—	—	数据 7 ~ 0	—	—	有效	—	
地址 3 的字节存取		—	—	—	数据 7 ~ 0	—	—	有效	
地址 0 的字存取		—	—	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效
地址 2 的字存取		—	—	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效
地址 0 的 长字存取	第 1 次 (地址 0)	—	—	数据 31 ~ 24	数据 23 ~ 16	—	—	有效	有效
	第 2 次 (地址 2)	—	—	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效

表 10.7 大端法的 8 位外部设备的存取和数据调整

操作		数据总线				选通信号			
		D31 ~ D24	D23 ~ D16	D15 ~ D8	D7 ~ D0	$\overline{\text{WE3}}$ 、 DQMUU	$\overline{\text{WE2}}$ 、 DQMUL	$\overline{\text{WE1}}$ 、 DQMLU	$\overline{\text{WE0}}$ 、 DQMLL
地址 0 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 1 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 2 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 3 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 0 的 字存取	第 1 次 (地址 0)	—	—	—	数据 15 ~ 8	—	—	—	有效
	第 2 次 (地址 1)	—	—	—	数据 7 ~ 0	—	—	—	有效
地址 2 的 字存取	第 1 次 (地址 0)	—	—	—	数据 15 ~ 8	—	—	—	有效
	第 2 次 (地址 1)	—	—	—	数据 7 ~ 0	—	—	—	有效
地址 0 的 长字存取	第 1 次 (地址 0)	—	—	—	数据 31 ~ 24	—	—	—	有效
	第 2 次 (地址 1)	—	—	—	数据 23 ~ 16	—	—	—	有效
	第 3 次 (地址 2)	—	—	—	数据 15 ~ 8	—	—	—	有效
	第 4 次 (地址 3)	—	—	—	数据 7 ~ 0	—	—	—	有效

表 10.8 小端法的 32 位外部设备的存取和数据调整

操作	数据总线				选通信号			
	D31 ~ D24	D23 ~ D16	D15 ~ D8	D7 ~ D0	$\overline{\text{WE3}}$ 、 DQMUU	$\overline{\text{WE2}}$ 、 DQMUL	$\overline{\text{WE1}}$ 、 DQMLU	$\overline{\text{WE0}}$ 、 DQMLL
地址 0 的字节存取	—	—	—	数据 7 ~ 0	—	—	—	有效
地址 1 的字节存取	—	—	数据 7 ~ 0	—	—	—	有效	—
地址 2 的字节存取	—	数据 7 ~ 0	—	—	—	有效	—	—
地址 3 的字节存取	数据 7 ~ 0	—	—	—	有效	—	—	—
地址 0 的字存取	—	—	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效
地址 2 的字存取	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效	—	—
地址 0 的长字存取	数据 31 ~ 24	数据 23 ~ 16	数据 15 ~ 8	数据 7 ~ 0	有效	有效	有效	有效

表 10.9 小端法的 16 位外部设备的存取和数据调整

操作	数据总线				选通信号			
	D31 ~ D24	D23 ~ D16	D15 ~ D8	D7 ~ D0	$\overline{\text{WE3}}$ 、 DQMUU	$\overline{\text{WE2}}$ 、 DQMUL	$\overline{\text{WE1}}$ 、 DQMLU	$\overline{\text{WE0}}$ 、 DQMLL
地址 0 的字节存取	—	—	—	数据 7 ~ 0	—	—	—	有效
地址 1 的字节存取	—	—	数据 7 ~ 0	—	—	—	有效	—
地址 2 的字节存取	—	—	—	数据 7 ~ 0	—	—	—	有效
地址 3 的字节存取	—	—	数据 7 ~ 0	—	—	—	有效	—
地址 0 的字存取	—	—	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效
地址 2 的字存取	—	—	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效
地址 0 的 长字存取	第 1 次 (地址 0)	—	数据 15 ~ 8	数据 7 ~ 0	—	—	有效	有效
	第 2 次 (地址 0)	—	数据 31 ~ 24	数据 23 ~ 16	—	—	有效	有效

表 10.10 小端法的 8 位外部设备的存取和数据调整

操作		数据总线				选通信号			
		D31 ~ D24	D23 ~ D16	D15 ~ D8	D7 ~ D0	$\overline{\text{WE3}}$ 、 DQMUU	$\overline{\text{WE2}}$ 、 DQMUL	$\overline{\text{WE1}}$ 、 DQMLU	$\overline{\text{WE0}}$ 、 DQMLL
地址 0 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 1 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 2 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 3 的字节存取		—	—	—	数据 7 ~ 0	—	—	—	有效
地址 0 的 字存取	第 1 次 (地址 0)	—	—	—	数据 7 ~ 0	—	—	—	有效
	第 2 次 (地址 1)	—	—	—	数据 15 ~ 8	—	—	—	有效
地址 2 的 字存取	第 1 次 (地址 0)	—	—	—	数据 7 ~ 0	—	—	—	有效
	第 2 次 (地址 1)	—	—	—	数据 15 ~ 8	—	—	—	有效
地址 0 的 长字存取	第 1 次 (地址 0)	—	—	—	数据 7 ~ 0	—	—	—	有效
	第 2 次 (地址 1)	—	—	—	数据 15 ~ 8	—	—	—	有效
	第 3 次 (地址 2)	—	—	—	数据 23 ~ 16	—	—	—	有效
	第 4 次 (地址 3)	—	—	—	数据 31 ~ 24	—	—	—	有效

10.5.2 正规空间接口

(1) 基本时序

正规空间的存取主要针对直接连接没有字节选择引脚的 SRAM 而输出选通信号。在使用带字节选择引脚的 SRAM 时，请参照“10.5.8 带字节选择的 SRAM 接口”。正规空间存取的基本时序如图 10.2 所示。没有等待的正规存取以 2 个周期结束。 \overline{BS} 信号表示总线周期的开始而且为 1 个周期有效。

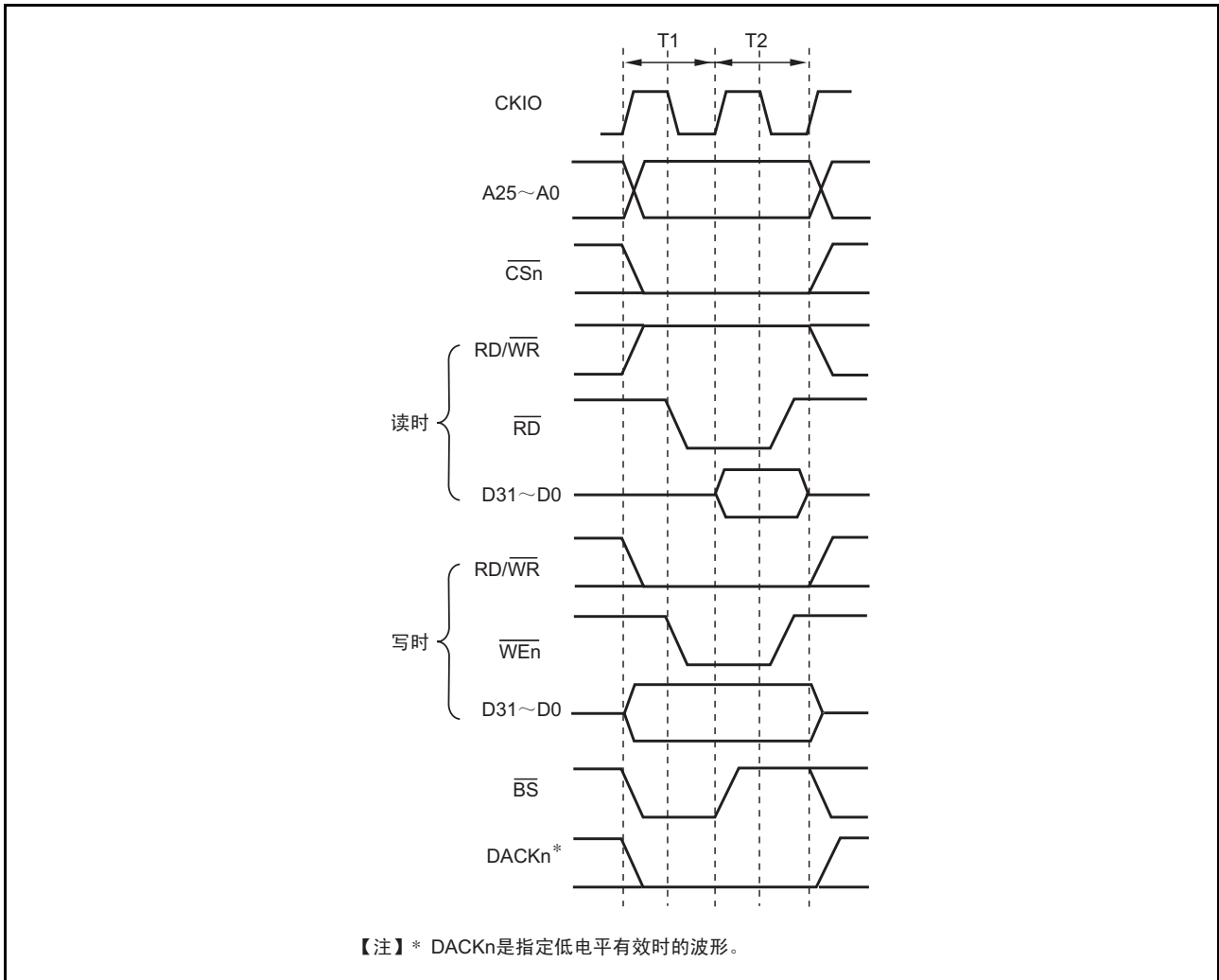


图 10.2 正规空间的基本存取（存取等待 0）

在进行读操作时，没有给外部总线指定存取长度。因为将正确的存取起始地址输出到地址的最低位却没有指定存取长度，所以 32 位设备总是读 32 位数据，而 16 位设备总是读 16 位数据。在进行写操作时，只有写字节的 \overline{WE} 信号才有效。

在给数据总线设置缓冲器时，需要使用 \overline{RD} 进行读数据的输出控制。必须注意：因为 $\overline{RD}/\overline{WR}$ 信号在没有存取时为读状态（高电平输出），如果使用 $\overline{RD}/\overline{WR}$ 信号控制外接数据缓冲器，就可能发生输出冲突。

正规空间的连续存取例子如图 10.3 和图 10.4 所示。当将 \overline{CSnWCR} 的 WM 位设定为“0”时，为了在存取所设定的 \overline{CSn} 空间后评价外部等待，插入 1 个 T_{nop} 周期（图 10.3）。但是，当将 \overline{CSnWCR} 的 WM 位设定为“1”时，忽视外部等待并且能抑制 T_{nop} 周期的插入（图 10.4）。

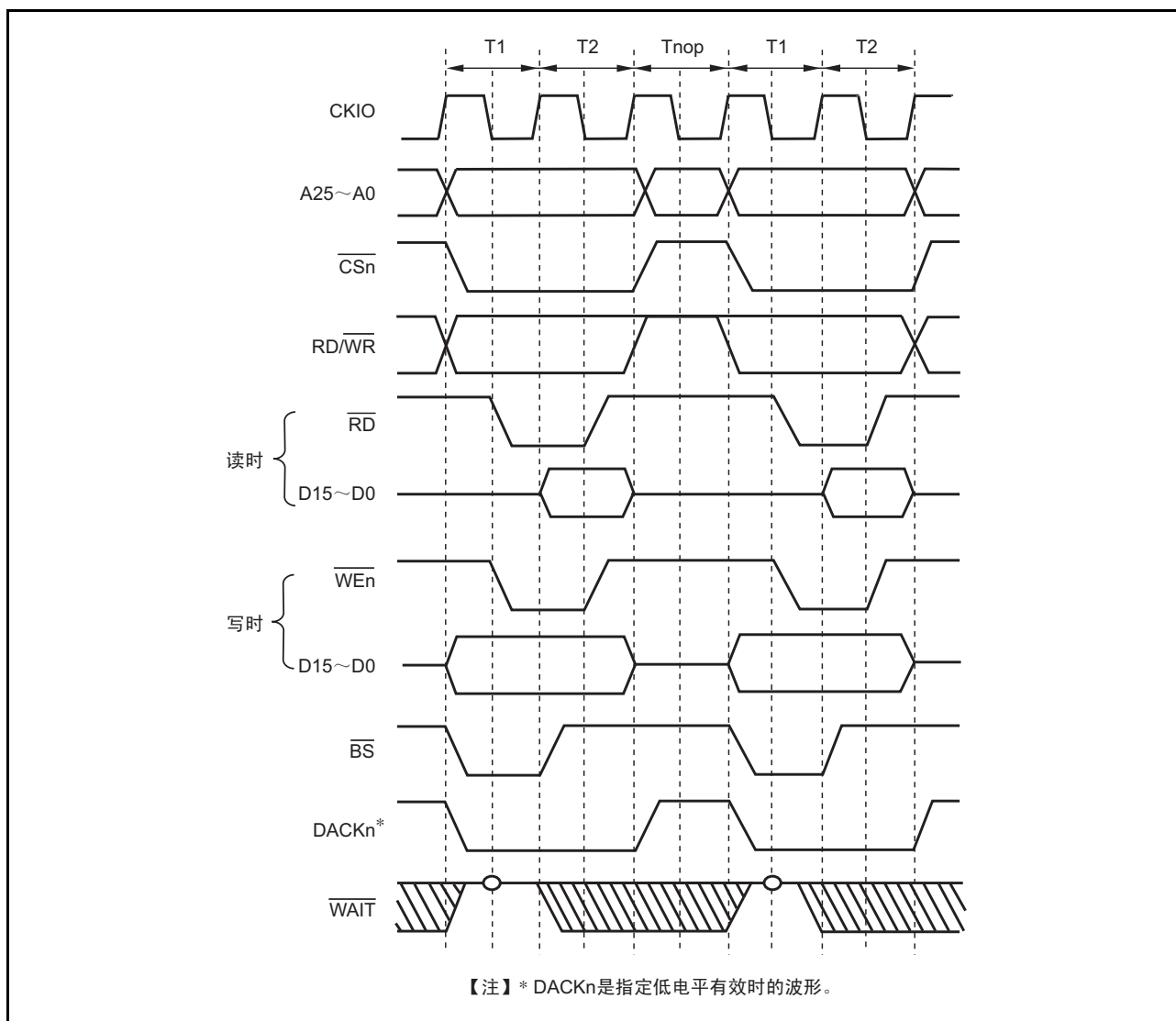


图 10.3 正规空间的连续存取例子 1
16 位总线宽度、长字存取、CSnWCR.WM 位 =0 (存取等待 0、周期之间的等待 0)

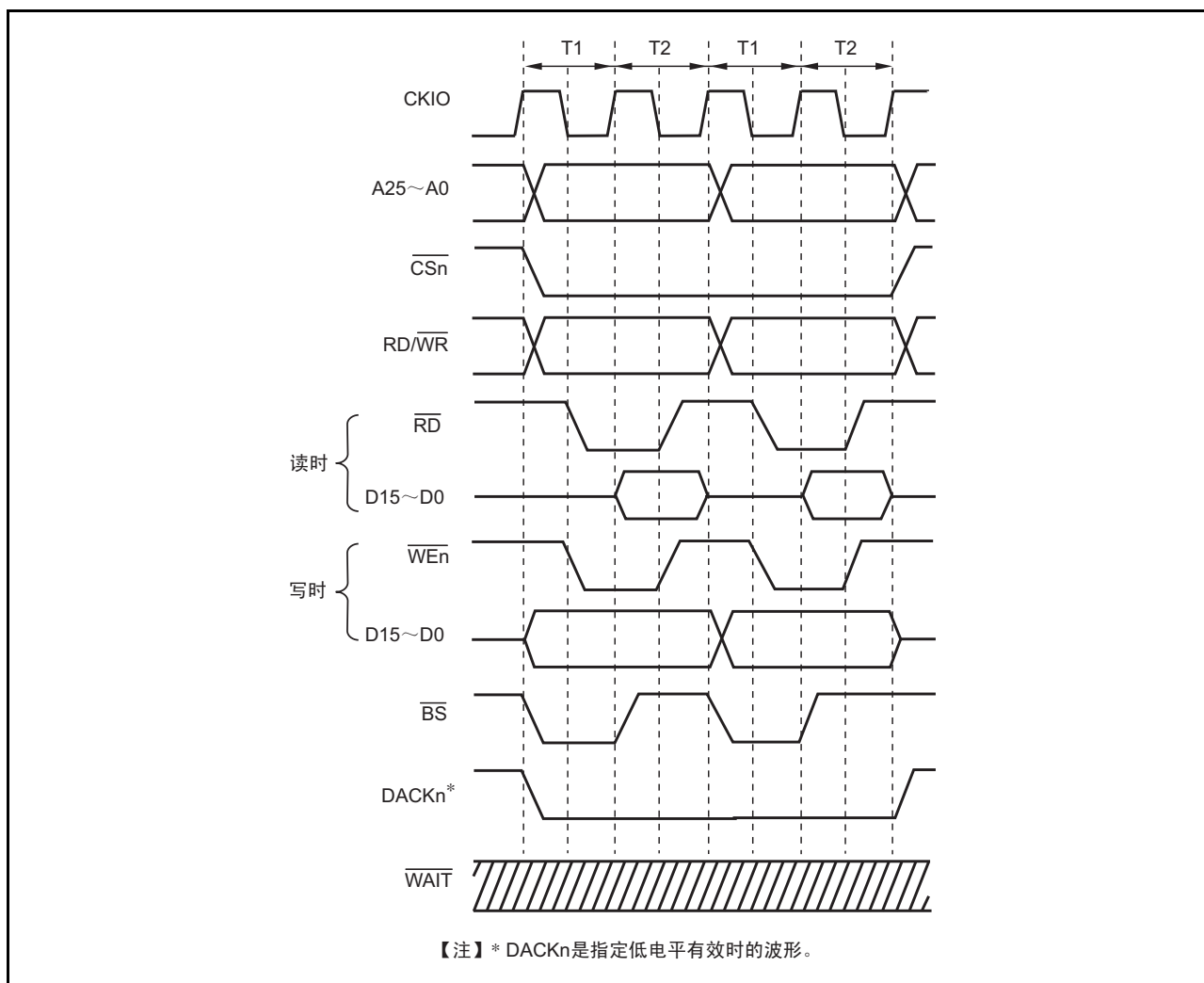


图 10.4 正规空间的连续存取例子 2
16 位总线宽度、长字存取、CSnWCR.WM 位 =1 (存取等待 0、周期之间的等待 0)

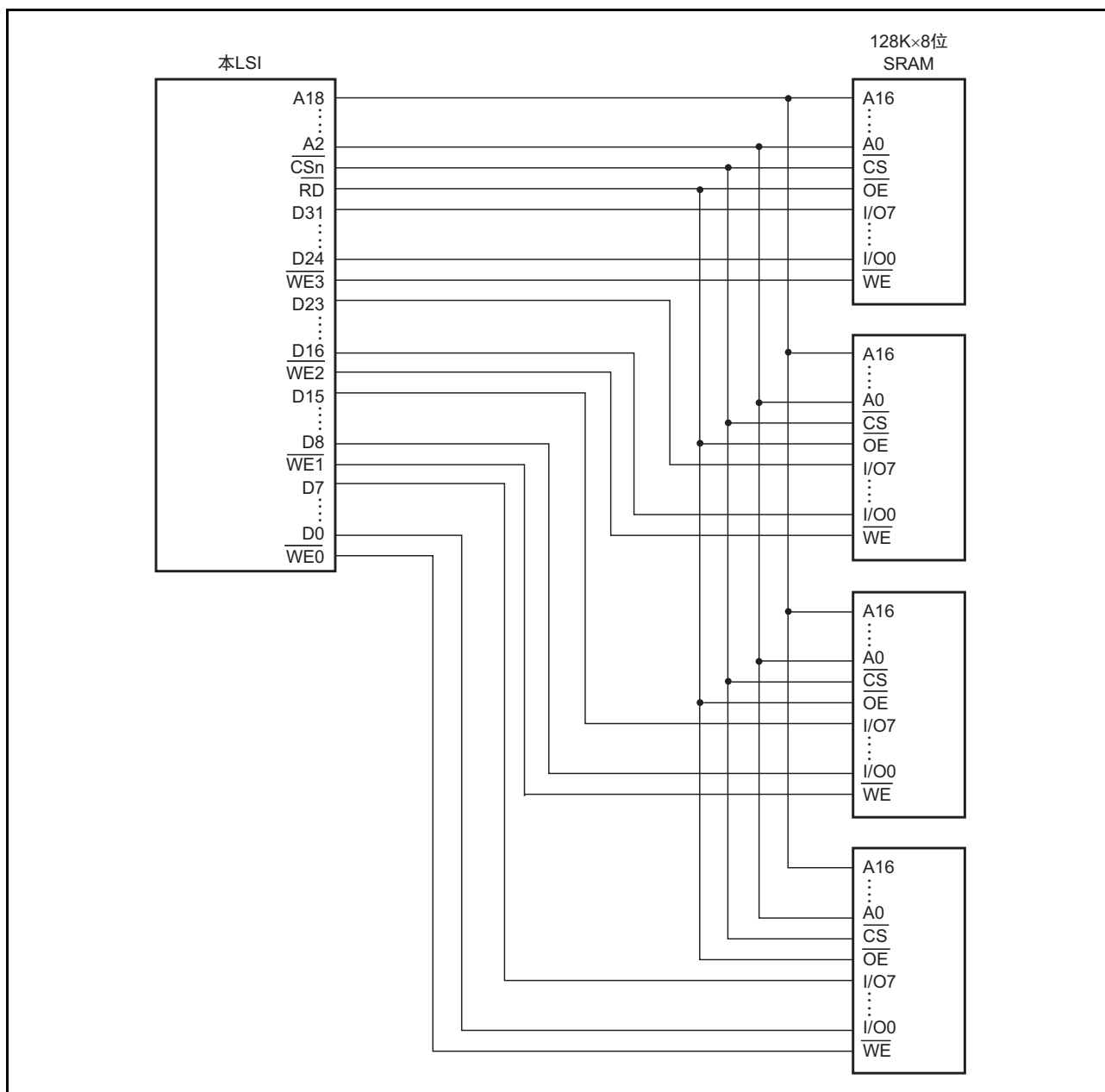


图 10.5 32 位数据宽度的 SRAM 连接例子

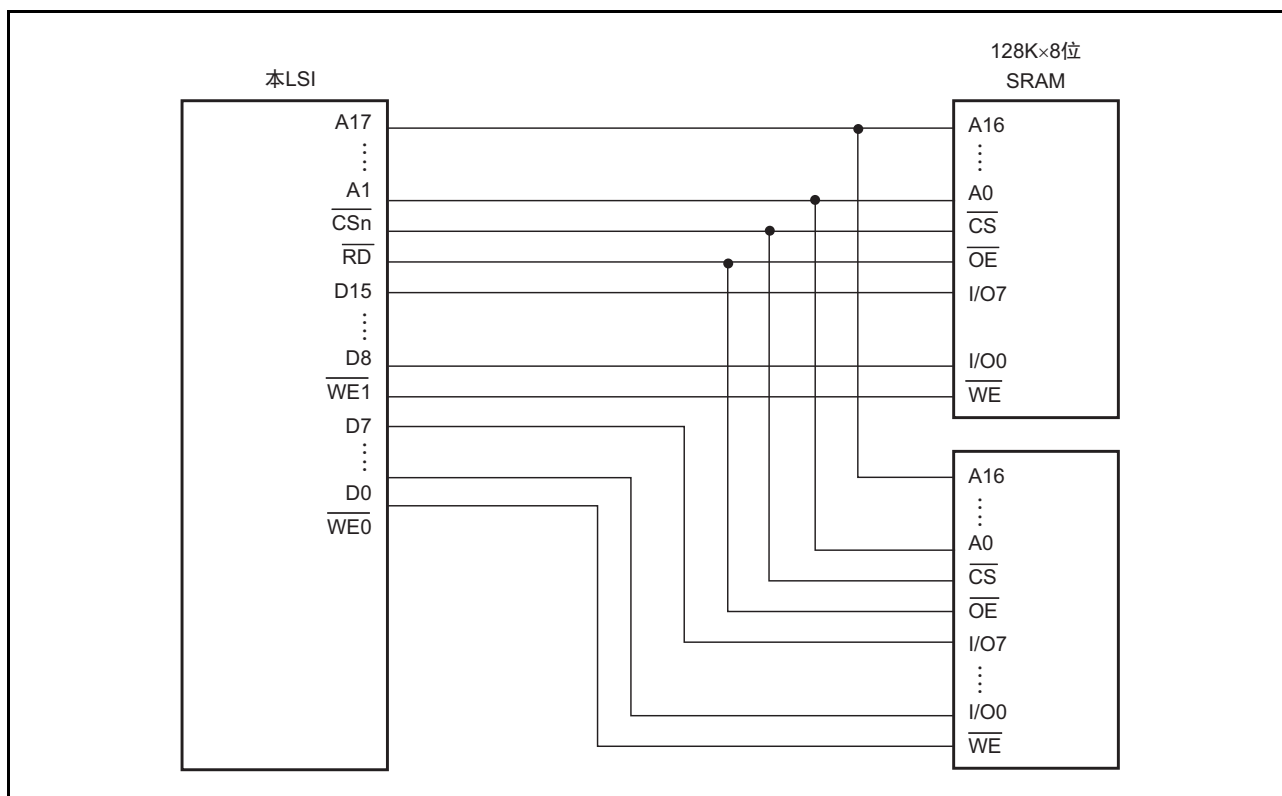


图 10.6 16 位数据宽度的 SRAM 连接例子

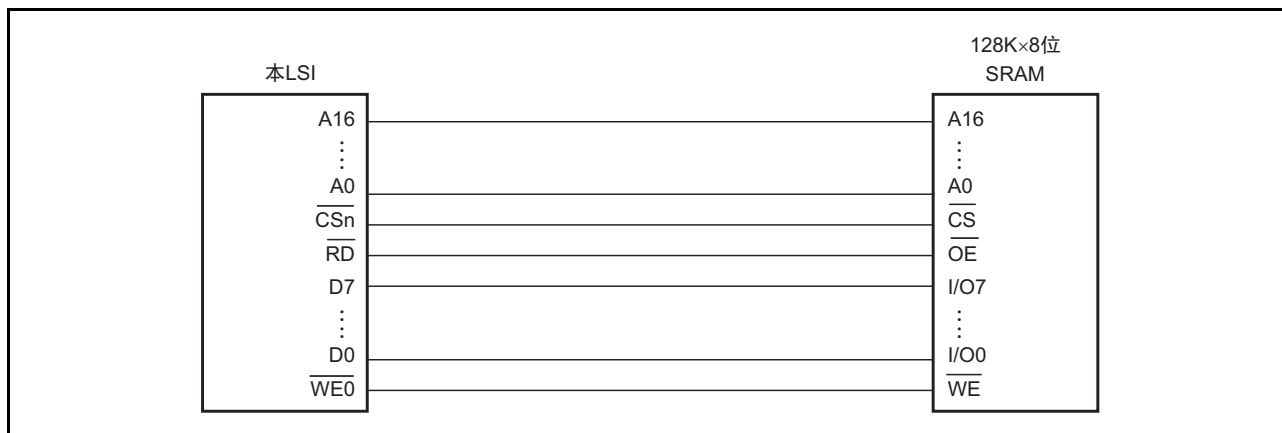


图 10.7 8 位数据宽度的 SRAM 连接例子

10.5.3 存取等待的控制

能通过设定 CSnWCR 的 WR[3:0] 位，控制正规空间存取的等待周期的插入。对于区域 1、区域 4 和区域 5，能在读写存取中独立插入等待周期。区域 0、区域 2 和区域 3 的存取等待在读写周期中是共用的。如图 10.8 所示，在正规空间存取中只插入被指定为等待周期的 Tw 周期数。

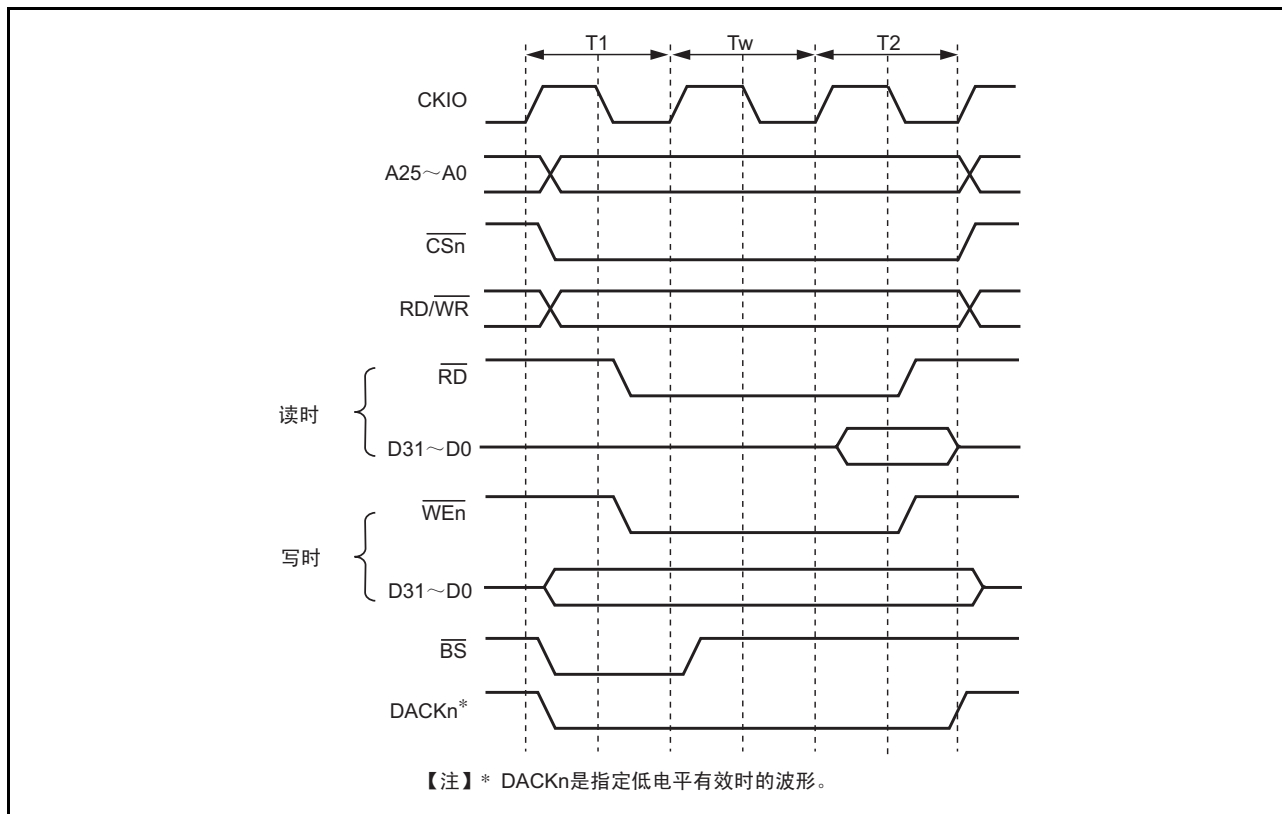


图 10.8 正规空间存取的等待时序（只限于软件等待）

在将 CSnWCR 的 WM 位设定为“0”时，还对来自外部的等待输入 $\overline{\text{WAIT}}$ 信号进行采样， $\overline{\text{WAIT}}$ 信号的采样如图 10.9 所示。将 2 个周期的等待指定为软件等待，在从 T1 周期或者 Tw 周期转移到 T2 周期时，在 CKIO 的下降沿对 $\overline{\text{WAIT}}$ 信号进行采样。

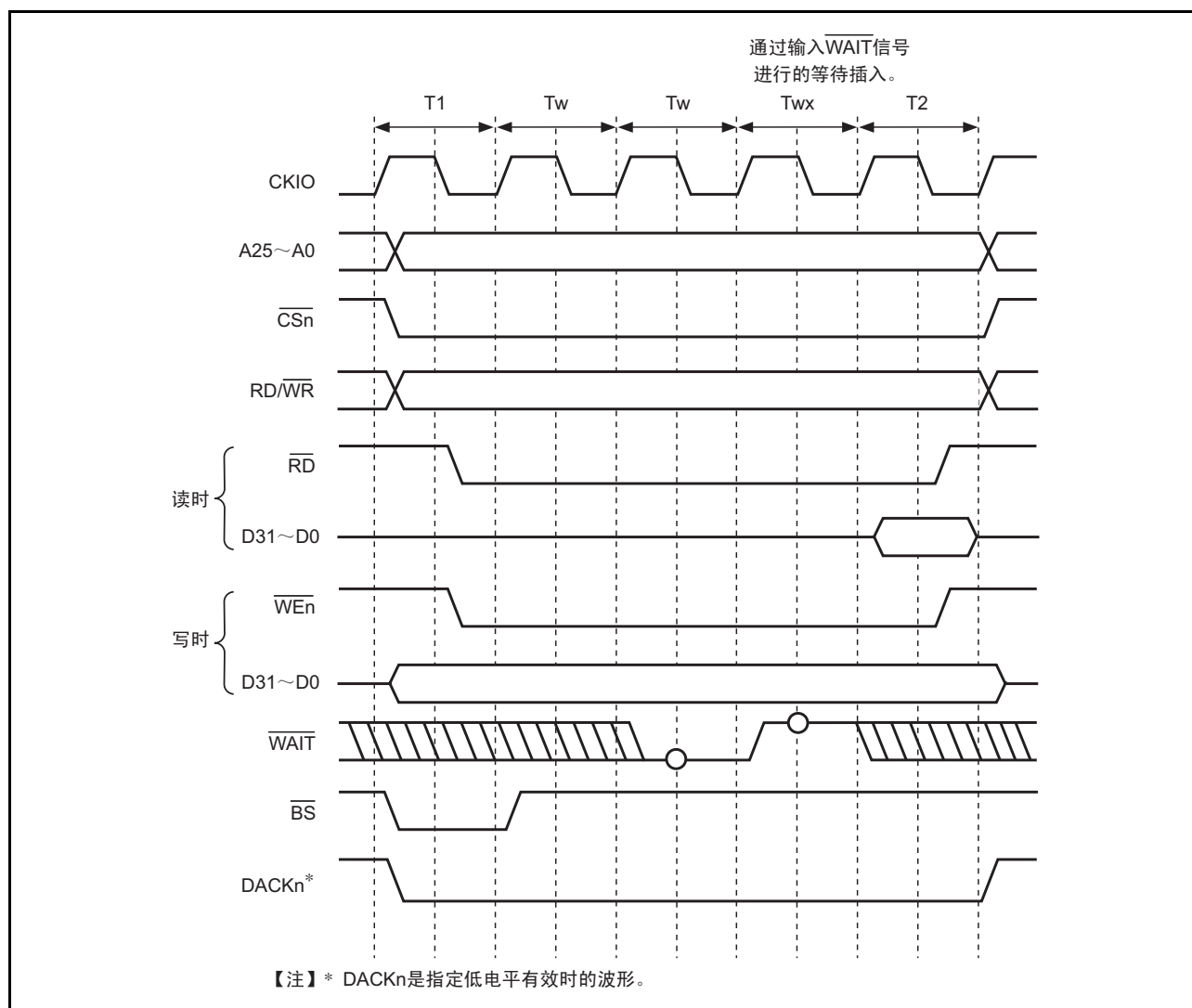


图 10.9 正规空间存取的等待时序（通过 $\overline{\text{WAIT}}$ 信号进行的等待插入）

10.5.4 \overline{CSn} 有效期间的扩展

能通过设定 \overline{CSnWCR} 的 $SW[1:0]$ 位，指定从 \overline{CSn} 有效到 \overline{RD} 和 \overline{WEn} 有效的周期数。还能通过设定 $HW[1:0]$ 位，指定从 \overline{RD} 和 \overline{WEn} 无效到 \overline{CSn} 无效的周期数。因此，能柔软地对应各种外部设备的接口。在图 10.10 所示的例子中，将 T_h 周期和 T_f 周期分别附加在正规周期的前后。在这些周期中，除了 \overline{RD} 和 \overline{WEn} 无效以外，其他信号都有效。另外，因为数据被延长输出到 T_f 周期为止，所以对写操作较慢的设备有效。

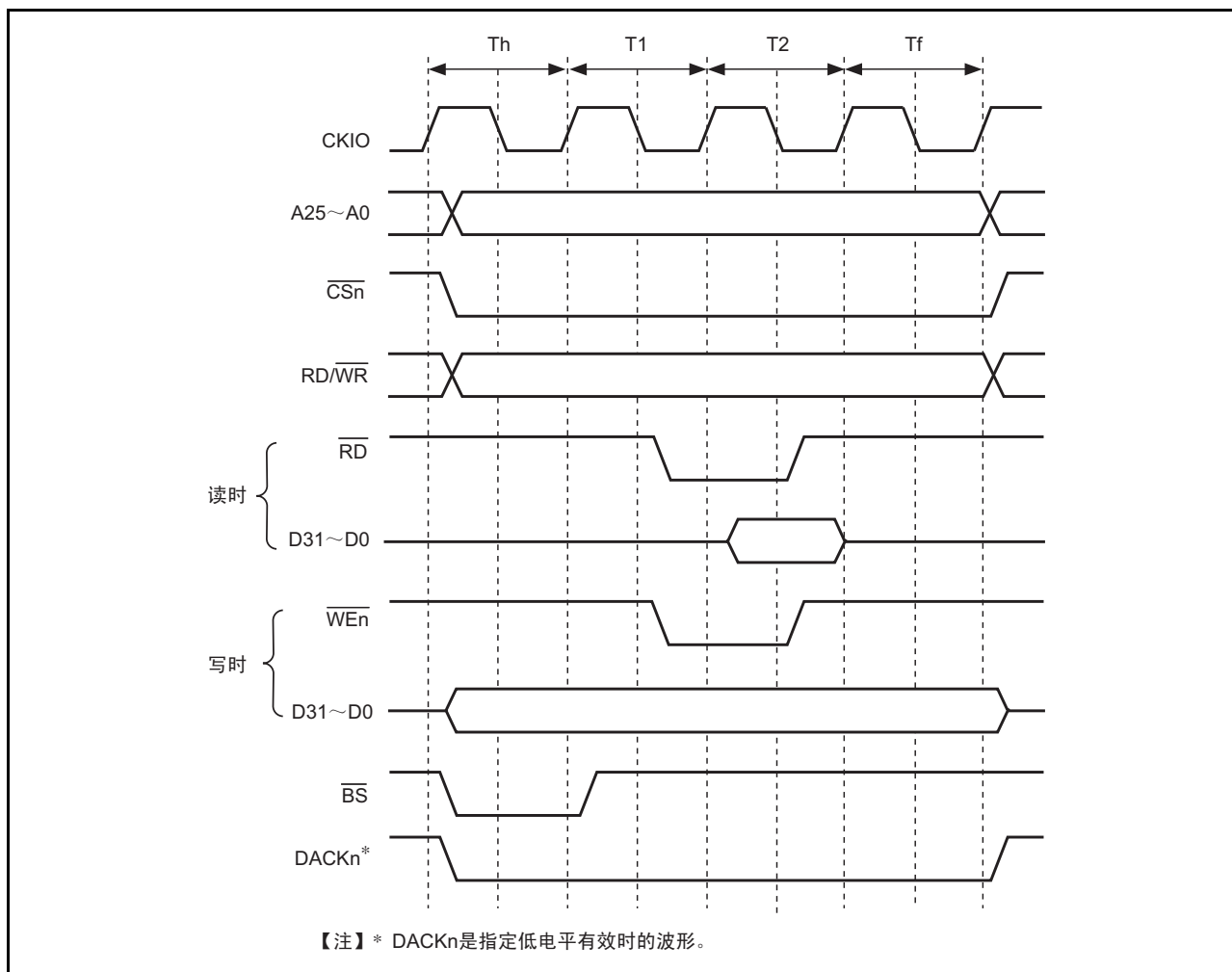


图 10.10 \overline{CSn} 有效期间的扩展

10.5.5 MPX-I/O 接口

MPX 空间的存取时序如下所示。在 MPX 空间中，通过 $\overline{CS5}$ 、 \overline{AH} 、 \overline{RD} 和 \overline{WEn} 信号控制存取。MPX 空间的基本存取在进行 2 个周期的地址输出后继续存取正规空间。地址输出周期和数据输入 / 输出周期的总线宽度固定为 8 位或者 16 位，或者根据存取地址为 8 位或 16 位可变总线宽度。

在 Ta2 周期到 Ta3 周期之间从 D15 ~ D0 或者 D7 ~ D0 输出地址，Ta1 周期为高阻抗状态，即使在连续存取时不插入空闲周期，也能防止地址和数据的冲突。另外，通过将 CS5WCR 的 MPXW 位设定为“1”，地址输出变为 3 个周期。

$\overline{RD}/\overline{WR}$ 信号和 $\overline{CS5}$ 信号同时输出，在读周期中输出高电平，而在写周期中输出低电平。

数据周期和正规空间的存取周期相同。

时序图如图 10.11 ~ 图 10.13 所示。

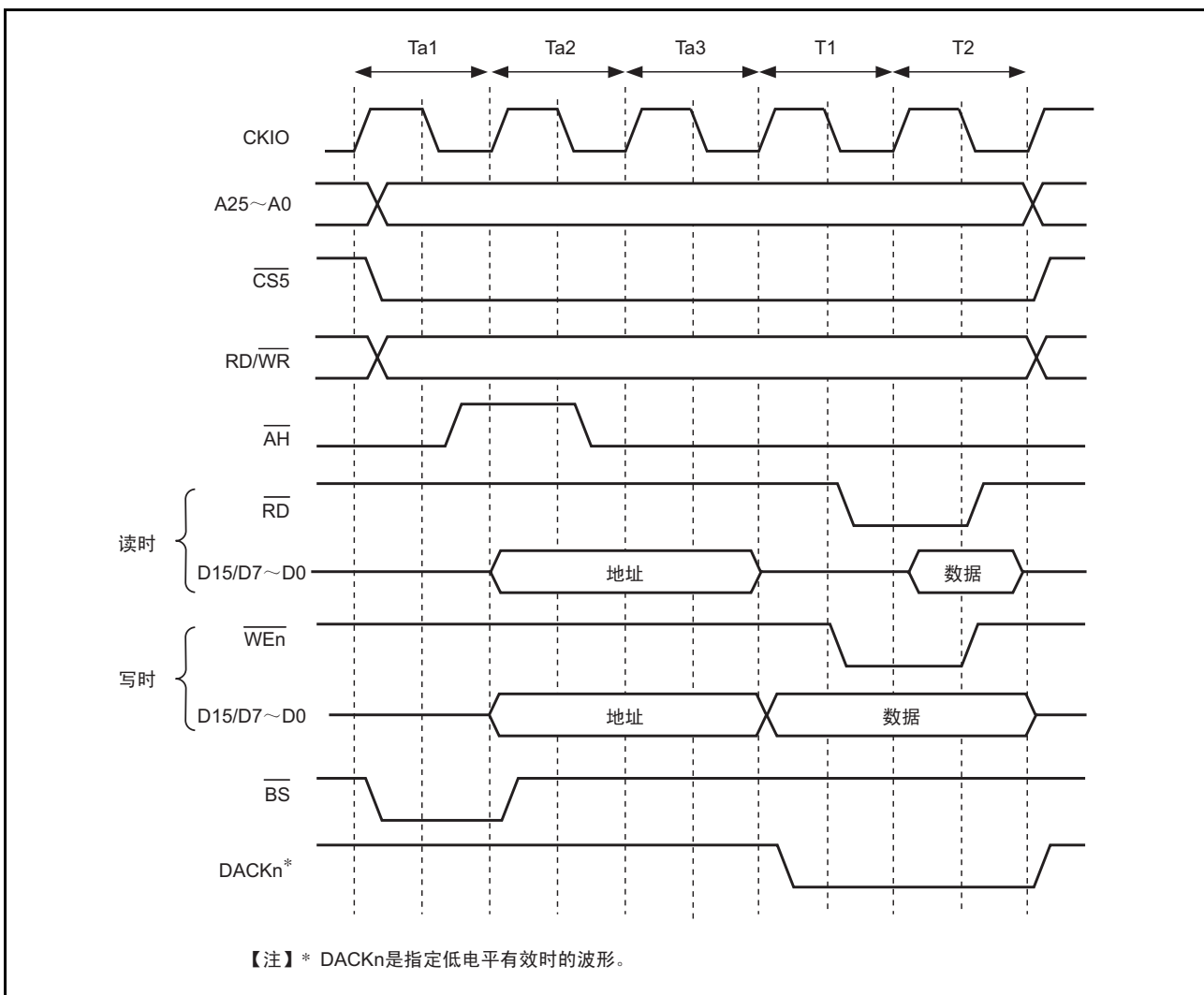


图 10.11 MPX 空间的存取时序（地址周期无等待、数据周期无等待）

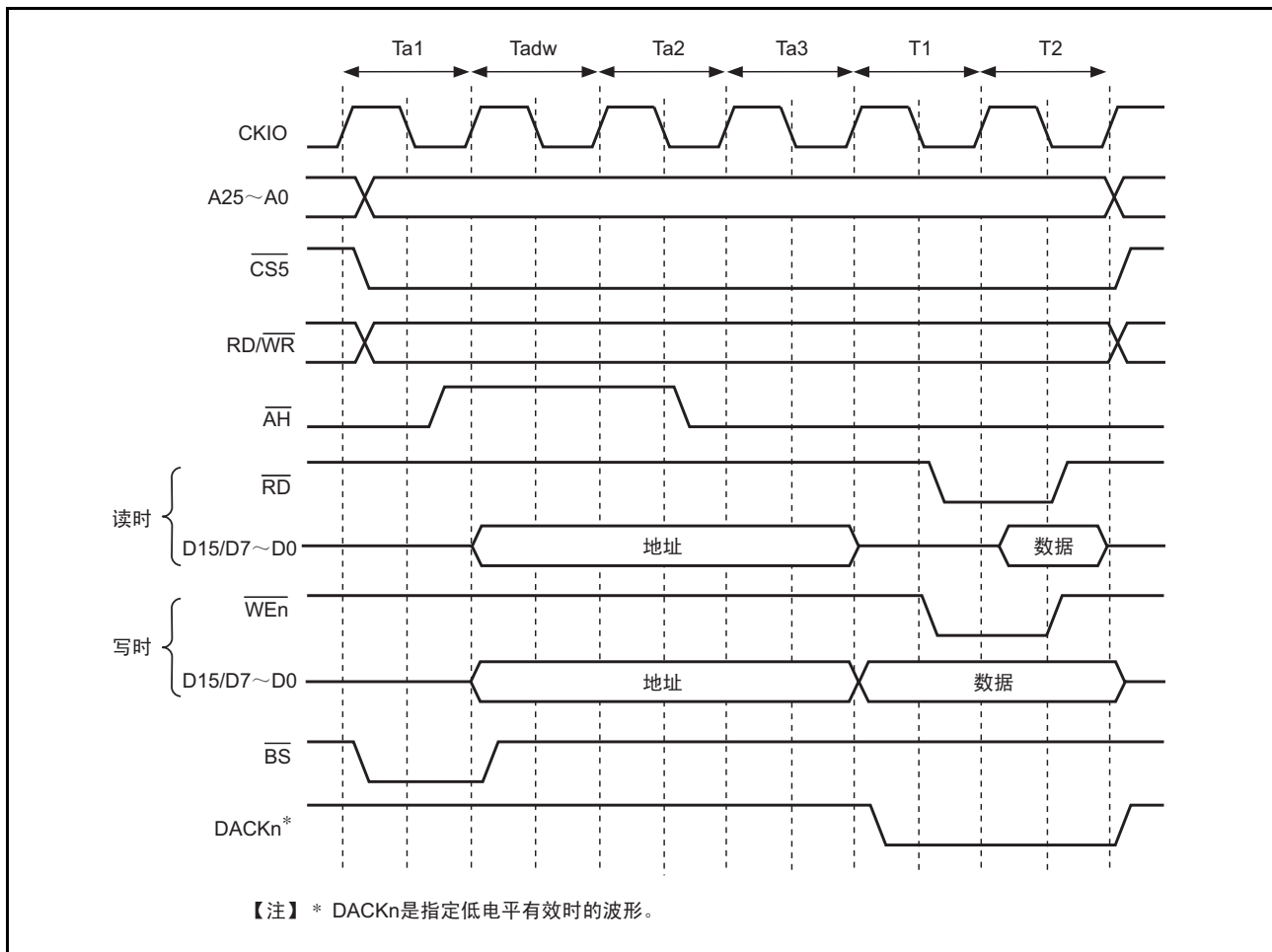


图 10.12 MPX 空间的存取时序（地址周期等待 1、数据周期无等待）

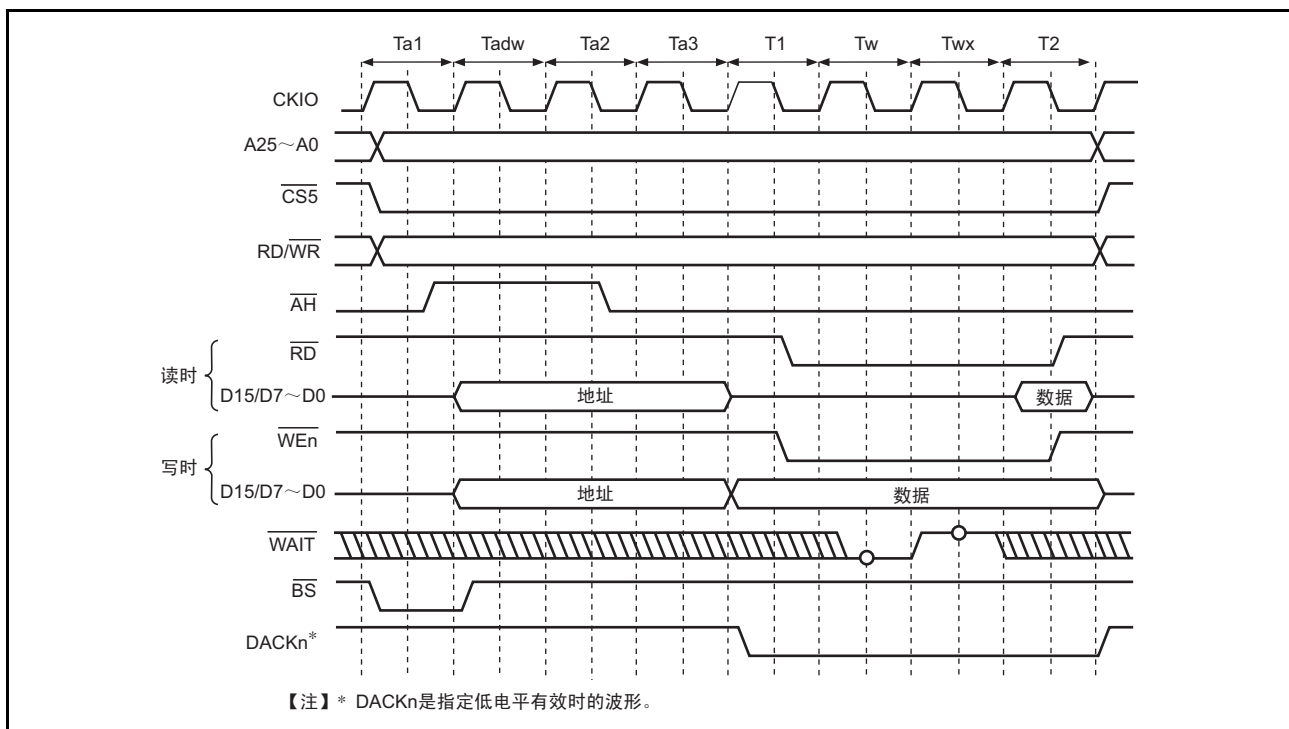


图 10.13 MPX 空间的存取时序（地址周期的存取等待 1、数据周期等待 1、外部等待 1）

10.5.6 SDRAM 接口

(1) 直接连接 SDRAM 的接口

能连接本 LSI 的 SDRAM 产品是：11/12/13 位的行地址，8/9/10 位的列地址，存储体数最多为 4 个，并且在读写命令周期中使用 A10 引脚设定预充电模式。

用于直接连接 SDRAM 的控制信号有 $\overline{\text{RAS}}$ 、 $\overline{\text{CAS}}$ 、 $\overline{\text{RD}}/\overline{\text{WR}}$ 、DQM_{UU}、DQM_{UL}、DQML_U、DQML_L、CKE、 $\overline{\text{CS2}}$ 和 $\overline{\text{CS3}}$ 。 $\overline{\text{CS2}}$ 和 $\overline{\text{CS3}}$ 以外的其他信号对各空间是共用的，并且 CKE 以外的其他信号只在 $\overline{\text{CS2}}$ 或者 $\overline{\text{CS3}}$ 有效时才有效。最多能将 SDRAM 连接到 2 个空间。能将连接 SDRAM 空间的数据总线宽度设定为 32 位或者 16 位。

SDRAM 的运行模式支持突发读 / 单次写（突发长度 1）和突发读 / 突发写（突发长度 1）。

通过 $\overline{\text{RAS}}$ 、 $\overline{\text{CAS}}$ 、 $\overline{\text{RD}}/\overline{\text{WR}}$ 和特定的地址信号指定 SDRAM 的命令。支持 NOP、自动刷新（REF）、自刷新（SELF）、全部存储体的预充电（PALL）、指定存储体的预充电（PRE）、存储体激活（ACTV）、读（READ）、带预充电的读（READA）、写（WRIT）、带预充电的写（WRITA）和模式寄存器的写（MRS、EMRS）等命令。

通过 DQM_{UU}、DQM_{UL}、DQML_U 和 DQML_L 指定要存取的字节。对相应的 DQM_{xx} 为低电平的字节进行读写。有关 DQM_{xx} 和要存取的字节的关系，请参照“10.5.1 字节序 / 存取长度和数据调整”。

本 LSI 和 SDRAM 的连接例子如图 10.14 和图 10.15 所示。

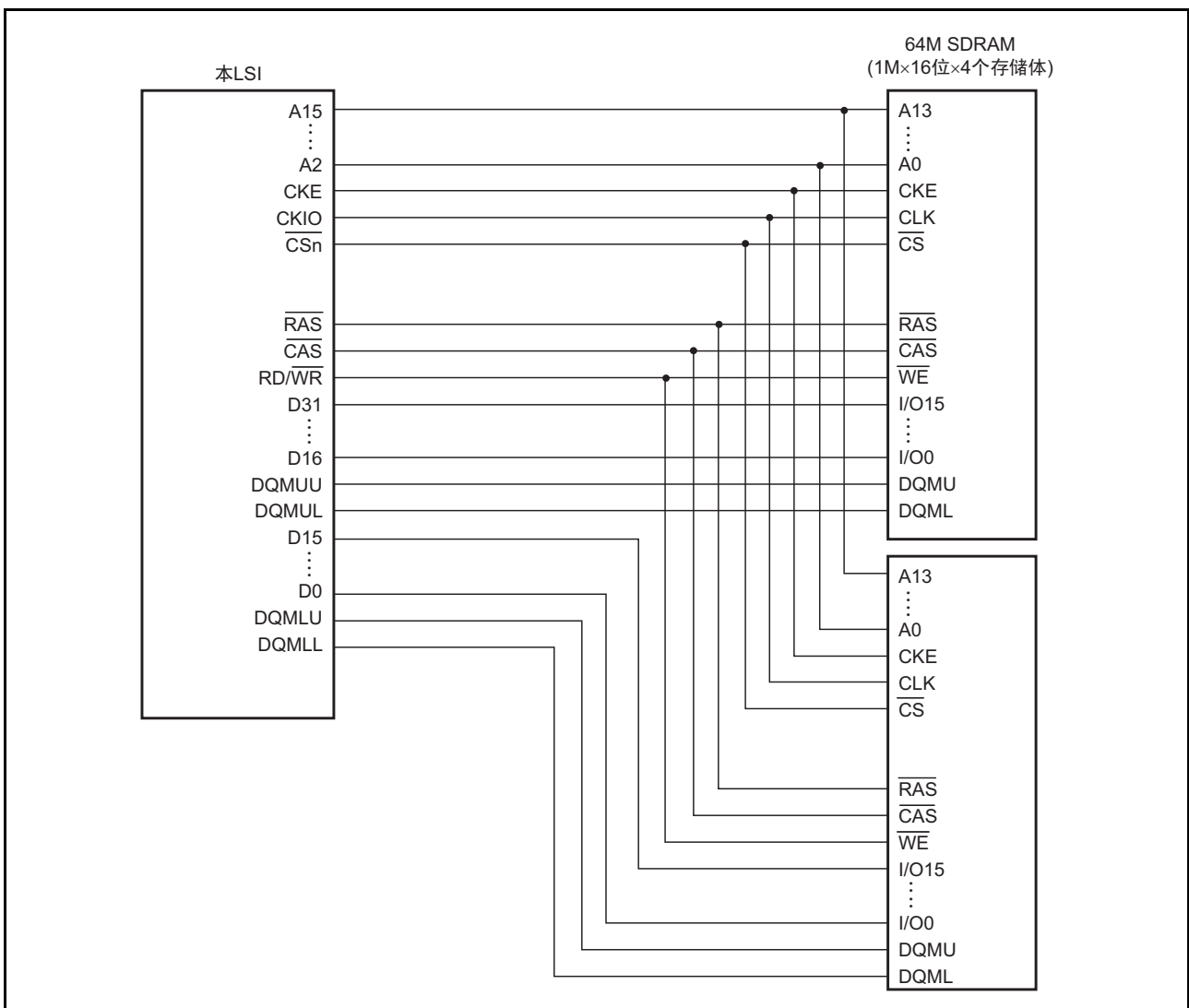


图 10.14 32 位数据宽度的 SDRAM 连接例子

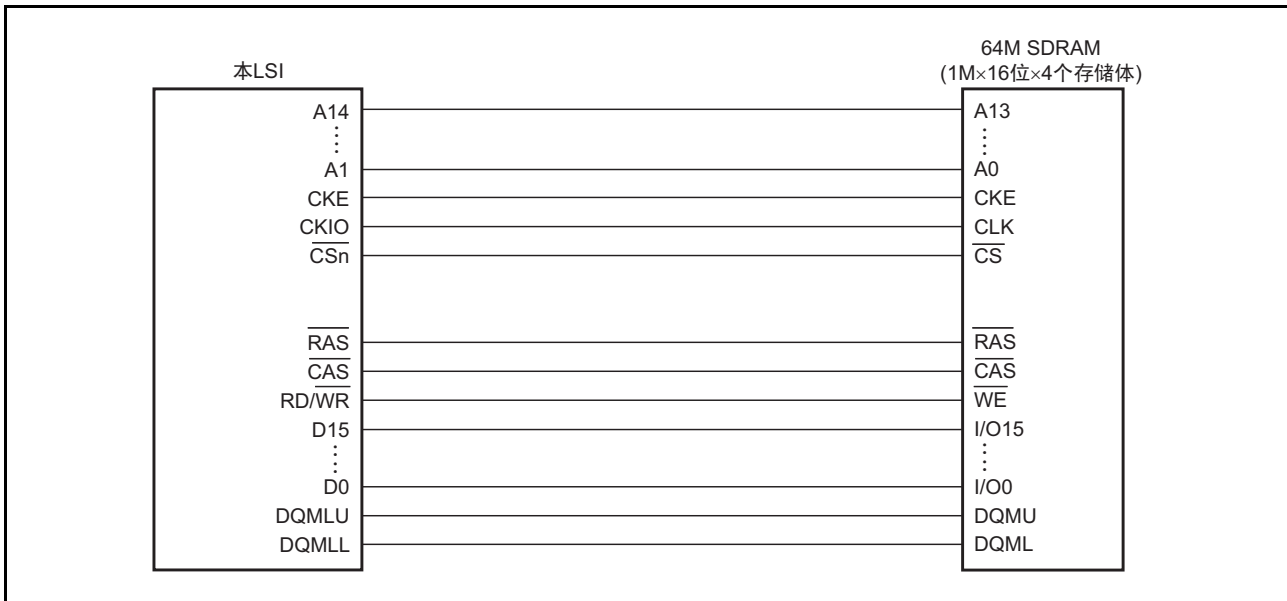


图 10.15 16 位数据宽度的 SDRAM 连接例子

(2) 地址的多路复用

根据 CSnBCR 的 BSZ[1:0] 位、SDCR 的 A2ROW[1:0] 位、A2COL[1:0] 位、A3ROW[1:0] 位和 A3COL[1:0] 位的设定进行地址的多路复用，以便不需要外接地址多路复用电路就能连接 SDRAM。BSZ[1:0]、A2ROW[1:0] 位、A2COL[1:0] 位、A3ROW[1:0] 位和 A3COL[1:0] 位的设定和输出到地址引脚的位的关系如表 10.11 ~ 表 10.16 所示。不能进行此表以外的设定，否则就不保证运行。A25 ~ A18 不进行多路复用而总是输出原地址。

当数据总线宽度为 16 位时 (BSZ[1:0]=B'10)，SDRAM 的 A0 引脚指定字地址，因此必须将 SDRAM 的 A0 引脚连接到本 LSI 的 A1 引脚，并且将 SDRAM 的 A1 引脚连接到本 LSI 的 A2 引脚，后面的引脚连接以此类推。当数据总线宽度为 32 位时 (BSZ[1:0]=B'11)，SDRAM 的 A0 引脚指定长字地址，因此必须将 SDRAM 的 A0 引脚连接到本 LSI 的 A2 引脚，并且将 SDRAM 的 A1 引脚连接到本 LSI 的 A3 引脚，后面的引脚连接以此类推。

表 10.11 BSZ[1:0]、A2/3ROW[1:0]、A2/3COL[1:0] 和地址多路复用的关系 (1)

设定			SDRAM 的引脚	功能	设定			SDRAM 的引脚	功能
BSZ[1:0]	A2/3ROW [1:0]	A2/3COL [1:0]			BSZ[1:0]	A2/3ROW [1:0]	A2/3COL [1:0]		
11 (32 位)	00 (11 位)	00 (8 位)			11 (32 位)	01 (12 位)	00 (8 位)		
本 LSI 的 输出引脚	行地址的 输出周期	列地址的 输出周期			本 LSI 的 输出引脚	行地址的 输出周期	列地址的 输出周期		
A17	A25	A17		未使用	A17	A25	A17		未使用
A16	A24	A16			A16	A24	A16		
A15	A23	A15			A15	A23*2	A23*2	A13(BA1)	指定 存储体
A14	A22*2	A22*2	A12(BA1)	指定 存储体	A14	A22*2	A22*2	A12(BA0)	
A13	A21*2	A21*2	A11(BA0)		A13	A21	A13	A11	地址
A12	A20	L/H*1	A10/AP	指定地址 / 预充电	A12	A20	L/H*1	A10/AP	指定地址 / 预充电
A11	A19	A11	A9	地址	A11	A19	A11	A9	地址
A10	A18	A10	A8		A10	A18	A10	A8	
A9	A17	A9	A7		A9	A17	A9	A7	
A8	A16	A8	A6		A8	A16	A8	A6	
A7	A15	A7	A5		A7	A15	A7	A5	
A6	A14	A6	A4		A6	A14	A6	A4	
A5	A13	A5	A3		A5	A13	A5	A3	
A4	A12	A4	A2		A4	A12	A4	A2	
A3	A11	A3	A1		A3	A11	A3	A1	
A2	A10	A2	A0		A2	A10	A2	A0	
A1	A9	A1		未使用	A1	A9	A1		未使用
A0	A8	A0			A0	A8	A0		
连接存储器的例子					连接存储器的例子				
64M 位产品 (512K 字 ×32 位 ×4 个存储体、列 8 位) 1 个 16M 位产品 (512K 字 ×16 位 ×2 个存储体、列 8 位) 2 个					128M 位产品 (1M 字 ×32 位 ×4 个存储体、列 8 位) 1 个 64M 位产品 (1M 字 ×16 位 ×4 个存储体、列 8 位) 2 个				

【注】 *1 L/H 是用于指定命令的位，根据存取模式，被固定为低电平或者高电平。

*2 指定存储体的地址。

表 10.12 BSZ[1:0]、A2/3ROW[1:0]、A2/3COL[1:0] 和地址多路复用的关系 (2)

设定			SDRAM 的引脚	功能	设定			SDRAM 的引脚	功能
BSZ[1:0]	A2/3ROW [1:0]	A2/3COL [1:0]			BSZ[1:0]	A2/3ROW [1:0]	A2/3COL [1:0]		
11 (32 位)	01 (12 位)	01 (9 位)			11 (32 位)	01 (12 位)	10 (10 位)		
本 LSI 的 输出引脚	行地址的 输出周期	列地址的 输出周期			本 LSI 的 输出引脚	行地址的 输出周期	列地址的 输出周期		
A17	A26	A17		未使用	A17	A27	A17		未使用
A16	A25	A16			A16	A26	A16		
A15	A24*2	A24*2	A13(BA1)	指定 存储体	A15	A25*2	A25*2	A13(BA1)	指定 存储体
A14	A23*2	A23*2	A12(BA0)		A14	A24*2	A24*2	A12(BA0)	
A13	A22	A13	A11	地址	A13	A23	A13	A11	地址
A12	A21	L/H*1	A10/AP	指定地址 / 预充电	A12	A22	L/H*1	A10/AP	指定地址 / 预充电
A11	A20	A11	A9	地址	A11	A21	A11	A9	地址
A10	A19	A10	A8		A10	A20	A10	A8	
A9	A18	A9	A7		A9	A19	A9	A7	
A8	A17	A8	A6		A8	A18	A8	A6	
A7	A16	A7	A5		A7	A17	A7	A5	
A6	A15	A6	A4		A6	A16	A6	A4	
A5	A14	A5	A3		A5	A15	A5	A3	
A4	A13	A4	A2		A4	A14	A4	A2	
A3	A12	A3	A1		A3	A13	A3	A1	
A2	A11	A2	A0		A2	A12	A2	A0	
A1	A10	A1		未使用	A1	A11	A1		未使用
A0	A9	A0			A0	A10	A0		
连接存储器的例子					连接存储器的例子				
256M 位产品 (2M 字 ×32 位 ×4 个存储体、列 9 位) 1 个 128M 位产品 (2M 字 ×16 位 ×4 个存储体、列 9 位) 2 个					512M 位产品 (4M 字 ×32 位 ×4 个存储体、列 10 位) 1 个 256M 位产品 (4M 字 ×16 位 ×4 个存储体、列 10 位) 2 个				

【注】 *1 L/H 是用于指定命令的位，根据存取模式，被固定为低电平或者高电平。

*2 指定存储体的地址。

表 10.13 BSZ[1:0]、A2/3ROW[1:0]、A2/3COL[1:0] 和地址多路复用的关系 (3)

设定					
BSZ[1:0]	A2/3ROW[1:0]	A2/3COL[1:0]			
11 (32 位)	10 (13 位)	01 (9 位)			
本 LSI 的输出引脚	行地址的输出周期	列地址的输出周期	SDRAM 的引脚	功能	
A17	A26	A17		未使用	
A16	A25*2	A25*2	A14(BA1)	指定存储体	
A15	A24*2	A24*2	A13(BA0)		
A14	A23	A14	A12	地址	
A13	A22	A13	A11		
A12	A21	L/H*1	A10/AP	指定地址 / 预充电	
A11	A20	A11	A9	地址	
A10	A19	A10	A8		
A9	A18	A9	A7		
A8	A17	A8	A6		
A7	A16	A7	A5		
A6	A15	A6	A4		
A5	A14	A5	A3		
A4	A13	A4	A2		
A3	A12	A3	A1		
A2	A11	A2	A0		
A1	A10	A1			未使用
A0	A9	A0			
连接存储器的例子					
512M 位产品 (4M 字 × 32 位 × 4 个存储体、列 9 位) 1 个					
256M 位产品 (4M 字 × 16 位 × 4 个存储体、列 9 位) 2 个					

【注】 *1 L/H 是用于指定命令的位，根据存取模式，被固定为低电平或者高电平。

*2 指定存储体的地址。

表 10.14 BSZ[1:0]、A2/3ROW[1:0]、A2/3COL[1:0] 和地址多路复用的关系 (4)

设定					设定				
BSZ [1:0]	A2/3ROW [1:0]	A2/3COL [1:0]		功能	BSZ [1:0]	A2/3ROW [1:0]	A2/3COL [1:0]		功能
10 (16 位)	00 (11 位)	00 (8 位)			10 (16 位)	01 (12 位)	00 (8 位)		
本 LSI 的 输出引脚	行地址的 输出周期	列地址的 输出周期	SDRAM 的引脚	功能	本 LSI 的 输出引脚	行地址的 输出周期	列地址的 输出周期	SDRAM 的引脚	功能
A17	A25	A17		未使用	A17	A25	A17		未使用
A16	A24	A16			A16	A24	A16		
A15	A23	A15			A15	A23	A15		
A14	A22	A14			A14	A22*2	A22*2	A13(BA1)	指定 存储体
A13	A21	A21			A13	A21*2	A21*2	A12(BA0)	指定 存储体
A12	A20*2	A20*2	A11(BA0)	指定 存储体	A12	A20	A12	A11	地址
A11	A19	L/H*1	A10/AP	指定地址 / 预充电	A11	A19	L/H*1	A10/AP	指定地址 / 预充电
A10	A18	A10	A9	地址	A10	A18	A10	A9	地址
A9	A17	A9	A8		A9	A17	A9	A8	
A8	A16	A8	A7		A8	A16	A8	A7	
A7	A15	A7	A6		A7	A15	A7	A6	
A6	A14	A6	A5		A6	A14	A6	A5	
A5	A13	A5	A4		A5	A13	A5	A4	
A4	A12	A4	A3		A4	A12	A4	A3	
A3	A11	A3	A2		A3	A11	A3	A2	
A2	A10	A2	A1		A2	A10	A2	A1	
A1	A9	A1	A0		A1	A9	A1	A0	
A0	A8	A0		未使用	A0	A8	A0		未使用
连接存储器的例子					连接存储器的例子				
16M 位产品 (512K 字 ×16 位 ×2 个存储体、列 8 位) 1 个					64M 位产品 (1M 字 ×16 位 ×4 个存储体、列 8 位) 1 个				

【注】 *1 L/H 是用于指定命令的位，根据存取模式，被固定为低电平或者高电平。

*2 指定存储体的地址。

表 10.15 BSZ[1:0]、A2/3ROW[1:0]、A2/3COL[1:0] 和地址多路复用的关系 (5)

设定			SDRAM 的引脚	功能	设定			SDRAM 的引脚	功能
BSZ[1:0]	A2/3ROW [1:0]	A2/3COL [1:0]			BSZ[1:0]	A2/3ROW [1:0]	A2/3COL [1:0]		
10 (16 位)	01 (12 位)	01 (9 位)			10 (16 位)	01 (12 位)	10 (10 位)		
本 LSI 的 输出引脚	行地址的 输出周期	列地址的 输出周期			本 LSI 的 输出引脚	行地址的 输出周期	列地址的 输出周期		
A17	A26	A17		未使用	A17	A27	A17		未使用
A16	A25	A16			A16	A26	A16		
A15	A24	A15			A15	A25	A15		
A14	A23*2	A23*2	A13(BA1)	指定	A14	A24*2	A24*2	A13(BA1)	指定
A13	A22*2	A22*2	A12(BA0)	存储体	A13	A23*2	A23*2	A12(BA0)	存储体
A12	A21	A12	A11	地址	A12	A22	A12	A11	地址
A11	A20	L/H*1	A10/AP	指定地址 / 预充电	A11	A21	L/H*1	A10/AP	指定地址 / 预充电
A10	A19	A10	A9	地址	A10	A20	A10	A9	地址
A9	A18	A9	A8		A9	A19	A9	A8	
A8	A17	A8	A7		A8	A18	A8	A7	
A7	A16	A7	A6		A7	A17	A7	A6	
A6	A15	A6	A5		A6	A16	A6	A5	
A5	A14	A5	A4		A5	A15	A5	A4	
A4	A13	A4	A3		A4	A14	A4	A3	
A3	A12	A3	A2		A3	A13	A3	A2	
A2	A11	A2	A1		A2	A12	A2	A1	
A1	A10	A1	A0		A1	A11	A1	A0	
A0	A9	A0		未使用	A0	A10	A0		未使用
连接存储器的例子					连接存储器的例子				
128M 位产品 (2M 字 ×16 位 ×4 个存储体、列 9 位) 1 个					256M 位产品 (4M 字 ×16 位 ×4 个存储体、列 10 位) 1 个				

【注】 *1 L/H 是用于指定命令的位，根据存取模式，被固定为低电平或者高电平。

*2 指定存储体的地址。

表 10.16 BSZ[1:0]、A2/3ROW[1:0]、A2/3COL[1:0] 和地址多路复用的关系 (6)

设定			SDRAM 的引脚	功能	设定			SDRAM 的引脚	功能
BSZ[1:0]	A2/3ROW [1:0]	A2/3COL [1:0]			BSZ[1:0]	A2/3ROW [1:0]	A2/3COL [1:0]		
10 (16 位)	10 (13 位)	01 (9 位)			10 (16 位)	10 (13 位)	10 (10 位)		
本 LSI 的 输出引脚	行地址的 输出周期	列地址的 输出周期			本 LSI 的 输出引脚	行地址的 输出周期	列地址的 输出周期		
A17	A26	A17		未使用	A17	A27	A17		未使用
A16	A25	A16			A16	A26	A16		
A15	A24*2	A24*2	A14(BA1)	指定 存储体	A15	A25*2	A25*2	A14(BA1)	指定 存储体
A14	A23*2	A23*2	A13(BA0)		A13(BA0)				
A13	A22	A13	A12	地址	A13	A23	A13	A12	地址
A12	A21	A12	A11		A12	A22	A12	A11	
A11	A20	L/H*1	A10/AP	指定地址 / 预充电 地址	A11	A21	L/H*1	A10/AP	指定地址 / 预充电 地址
A10	A19	A10	A9		A10	A20	A10	A9	
A9	A18	A9	A8		A9	A19	A9	A8	
A8	A17	A8	A7		A8	A18	A8	A7	
A7	A16	A7	A6		A7	A17	A7	A6	
A6	A15	A6	A5		A6	A16	A6	A5	
A5	A14	A5	A4		A5	A15	A5	A4	
A4	A13	A4	A3		A4	A14	A4	A3	
A3	A12	A3	A2		A3	A13	A3	A2	
A2	A11	A2	A1		A2	A12	A2	A1	
A1	A10	A1	A0	A1	A11	A1	A0		
A0	A9	A0		未使用	A0	A10	A0		未使用
连接存储器的例子					连接存储器的例子				
256M 位产品 (4M 字 × 16 位 × 4 个存储体、列 9 位) 1 个					512M 位产品 (8M 字 × 16 位 × 4 个存储体、列 10 位) 1 个				

【注】 *1 L/H 是用于指定命令的位，根据存取模式，被固定为低电平或者高电平。

*2 指定存储体的地址。

(3) 突发读

本 LSI 发生突发读的条件如下：

1. 当读的存取长度大于数据总线宽度时
2. 在高速缓存未命中的情况下进行 16 字节传送时
3. 当通过直接存储器存取控制器进行 16 字节传送时
4. 当通过 OpenVG 瑞萨图形处理器、失真校正引擎和视频显示控制器 4 进行 32 字节传送时
5. 当通过视频显示控制器 4 进行 128 字节传送时

本 LSI 总是通过突发长度 1 进行 SDRAM 的存取。例如，当从连接 16 位数据总线的 SDRAM 连续读 16 字节的数据时，连续进行 8 次突发长度 1 的读操作，此时的存取称为突发数 8 的突发读。存取长度和突发数的关系如表 10.17 所示。

表 10.17 存取长度和突发数的关系

总线宽度	存取长度	突发数
16 位	8 位	1
	16 位	1
	32 位	2
	16 字节	8
	32 字节	16
	128 字节	64
32 位	8 位	1
	16 位	1
	32 位	1
	16 字节	4
	32 字节	8
	128 字节	32

突发读时的时序图如图 10.16 和图 10.17 所示。在突发读过程中，在 T_r 周期输出 ACTV 命令，接着在 T_{c1} 、 T_{c2} 、 T_{c3} 周期发行 READ 命令，在 T_{c4} 周期发行 READA 命令，并且在从 T_{d1} 到 T_{d4} 的周期中在外部时钟（CKIO）的上升沿接收读数据。 T_{ap} 周期是在 SDRAM 内部等待自动预充电（由 READA 命令引起的）结束的周期，在此期间不对同一存储体发行新的命令，但是能存取不同 CS 空间或者同一 SDRAM 的不同存储体。通过指定 CS3WCR 的 WTRP[1:0] 位决定 T_{ap} 的周期数。

本 LSI 为了用各种频率连接 SDRAM，能通过设定 CS3WCR 的各个位插入等待周期。各种等待的设定例子如图 10.17 所示。能通过 CS3WCR 的 WTRCD[1:0]，指定从 ACTV 命令输出周期 T_r 到 READ 命令输出周期 T_{c1} 的周期数。当将 WTRCD[1:0] 至少设定为 1 个周期时，就在 T_r 周期和 T_{c1} 周期之间插入 NOP 命令发行的 T_{rw} 周期。能通过 CS2WCR 的 A2CL[1:0] 位和 CS3WCR 的 A3CL[1:0] 位，在 CS2 和 CS3 空间内分别独立指定从 READ 命令输出周期 T_{c1} 到读数据输入周期 T_{d1} 的周期数。此周期数相当于 SDRAM 的 CAS 等待时间。SDRAM 的 CAS 等待时间最多为 3 个周期，但是本 LSI 能设定 1 ~ 4 个周期。这是为了在本 LSI 和 SDRAM 之间连接含锁存器的电路。

T_{de} 周期是将读数据传送到本 LSI 内部时所需的空闲周期，在进行突发读或者单次读时，总是产生 1 个 T_{de} 周期。

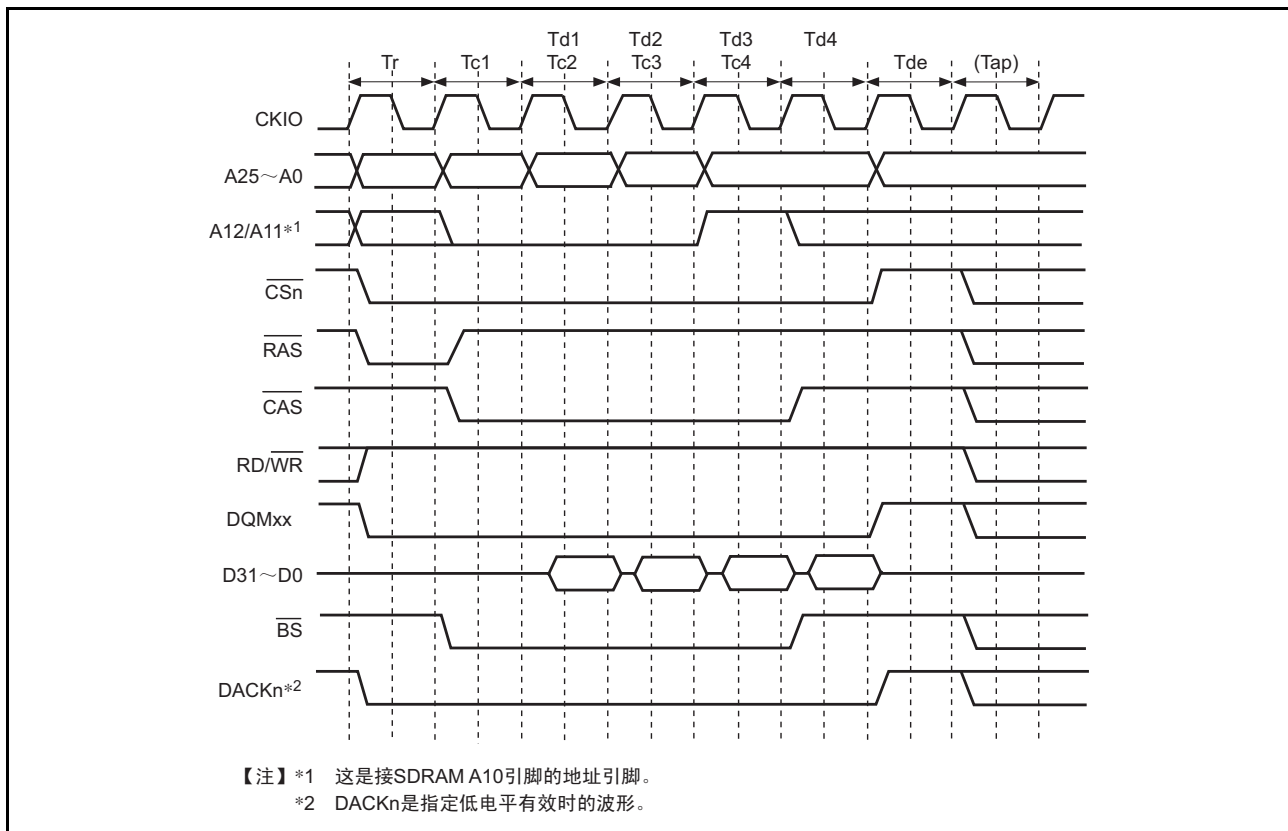


图 10.16 突发读的基本时序（CAS 等待时间 1、自动预充电）

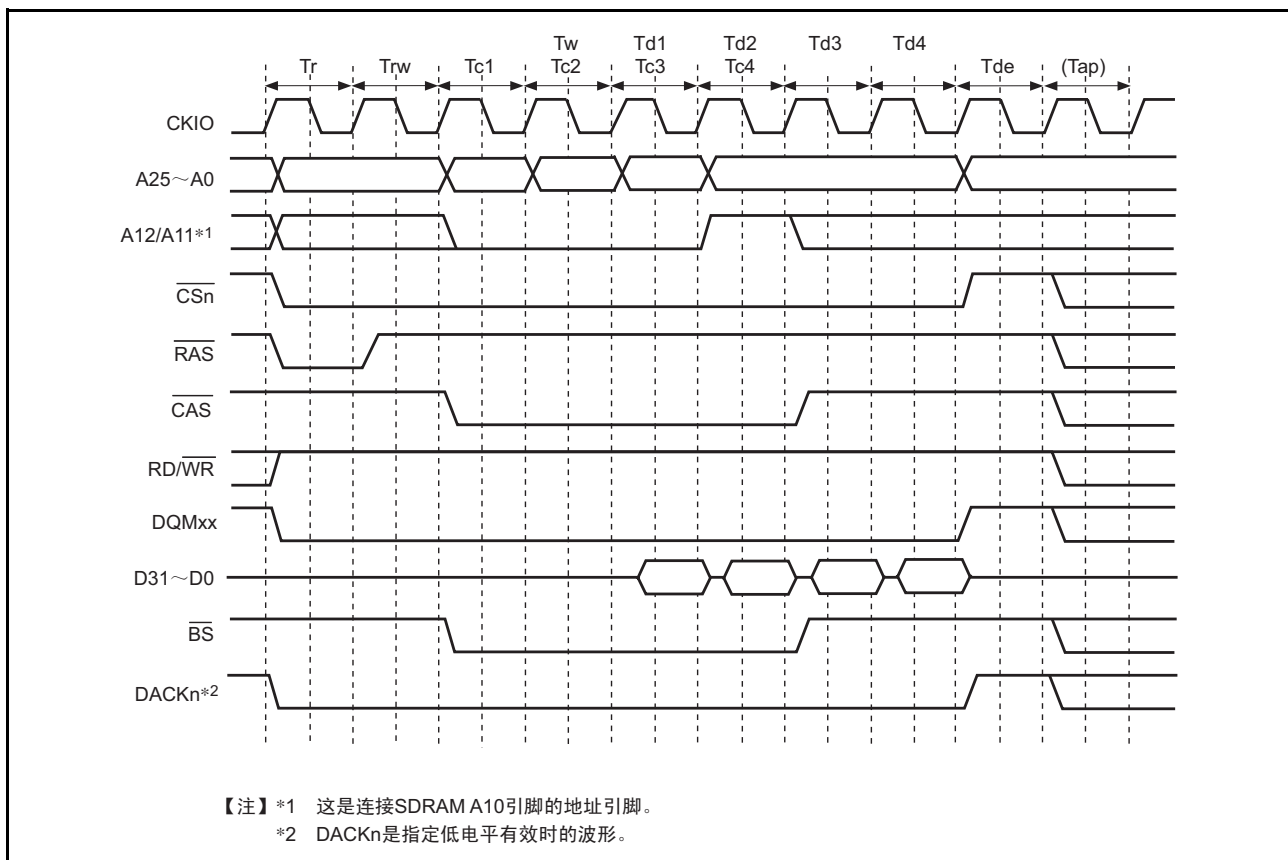


图 10.17 突发读的等待指定时序（CAS 等待时间 2、WTRCD[1:0]=1 个周期、自动预充电）

(4) 单次读

在高速缓存无效空间中数据总线宽度不小于存取长度时，以 1 次读存取结束。因为设定为突发长度 1 的突发读，所以 SDRAM 只输出需要的数据。进行一次就结束的读存取称为单次读。

单次读的基本时序图如图 10.18 所示。

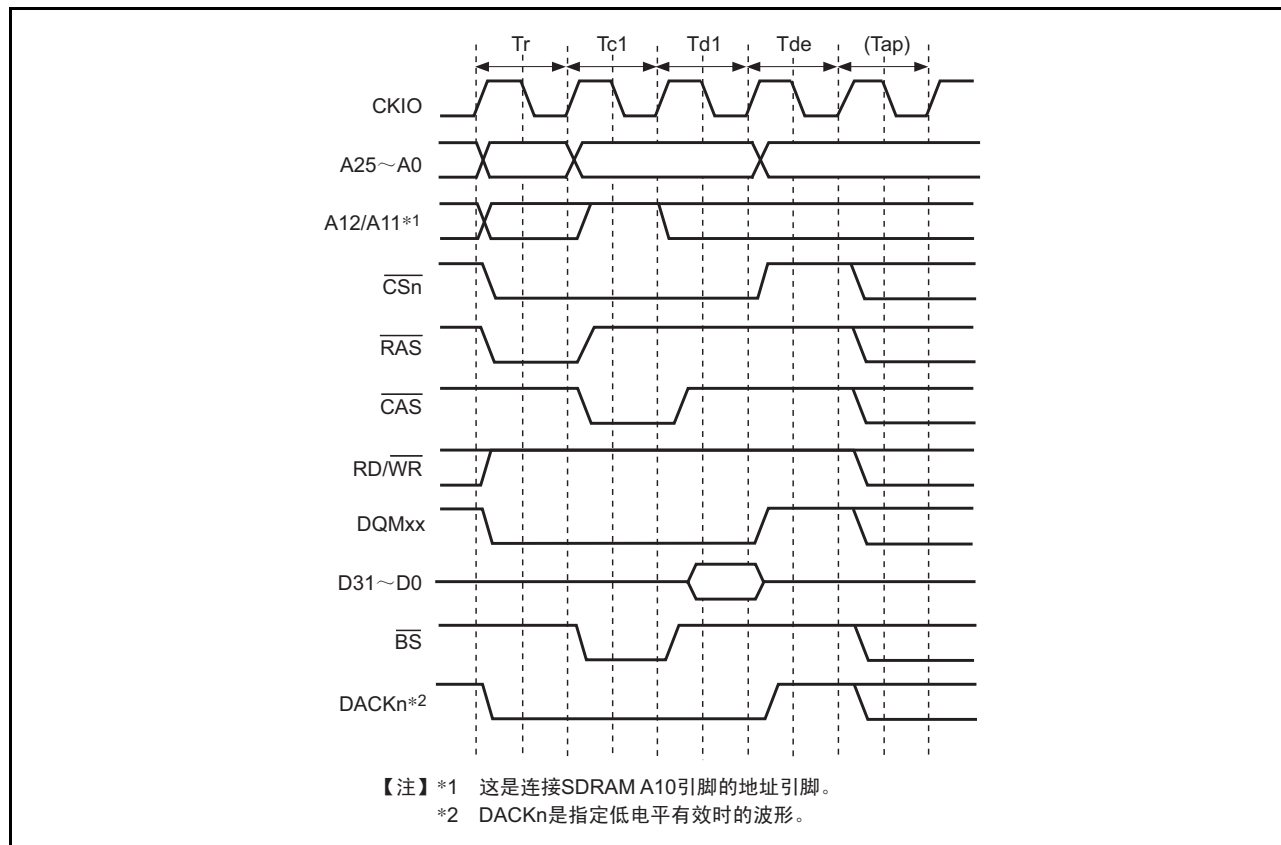


图 10.18 单次读的基本时序（CAS 等待时间 1、自动预充电）

(5) 突发写

本 LSI 发生突发写的条件如下：

1. 当写的存取长度大于数据总线宽度时
2. 当发生高速缓存的回写时
3. 当通过直接存储器存取控制器进行 16 字节传送时
4. 当通过视频显示控制器 4 进行 32 字节传送时
5. 当通过视频显示控制器 4 进行 128 字节传送时

本 LSI 总是通过突发长度 1 进行 SDRAM 的存取。例如，当从连接 16 位数据总线的 SDRAM 连续写 16 字节的数据时，连续进行 8 次突发长度 1 的写操作，此时的存取称为突发数 8 的突发写。存取长度和突发数的关系如表 10.17 所示。突发写的时序图如图 10.19 所示。在突发写过程中，在 T_r 周期输出 ACTV 命令，接着在 T_{c1} 、 T_{c2} 、 T_{c3} 周期发行 WRIT 命令，在 T_{c4} 周期发行进行自动预充电的 WRITA 命令。在写周期中，同时输出写数据和写命令。在输出带自动预充电的写命令后，连续出现等待自动预充电启动的 T_{rw1} 周期和等待自动预充电结束的 T_{ap} 周期。 T_{ap} 周期是在 SDRAM 内部等待自动预充电结束（由 WRITA 命令引起的）的周期。在 T_{rw1} 周期和 T_{ap} 周期之间不对同一存储体发行新的命令，但是能存取不同 CS 空间或者同一 SDRAM 的不同存储体。通过指定 CS3WCR 的 TRWL[1:0] 位决定 T_{rw1} 周期，通过指定 CS3WCR 的 WTRP[1:0] 位决定 T_{ap} 周期。

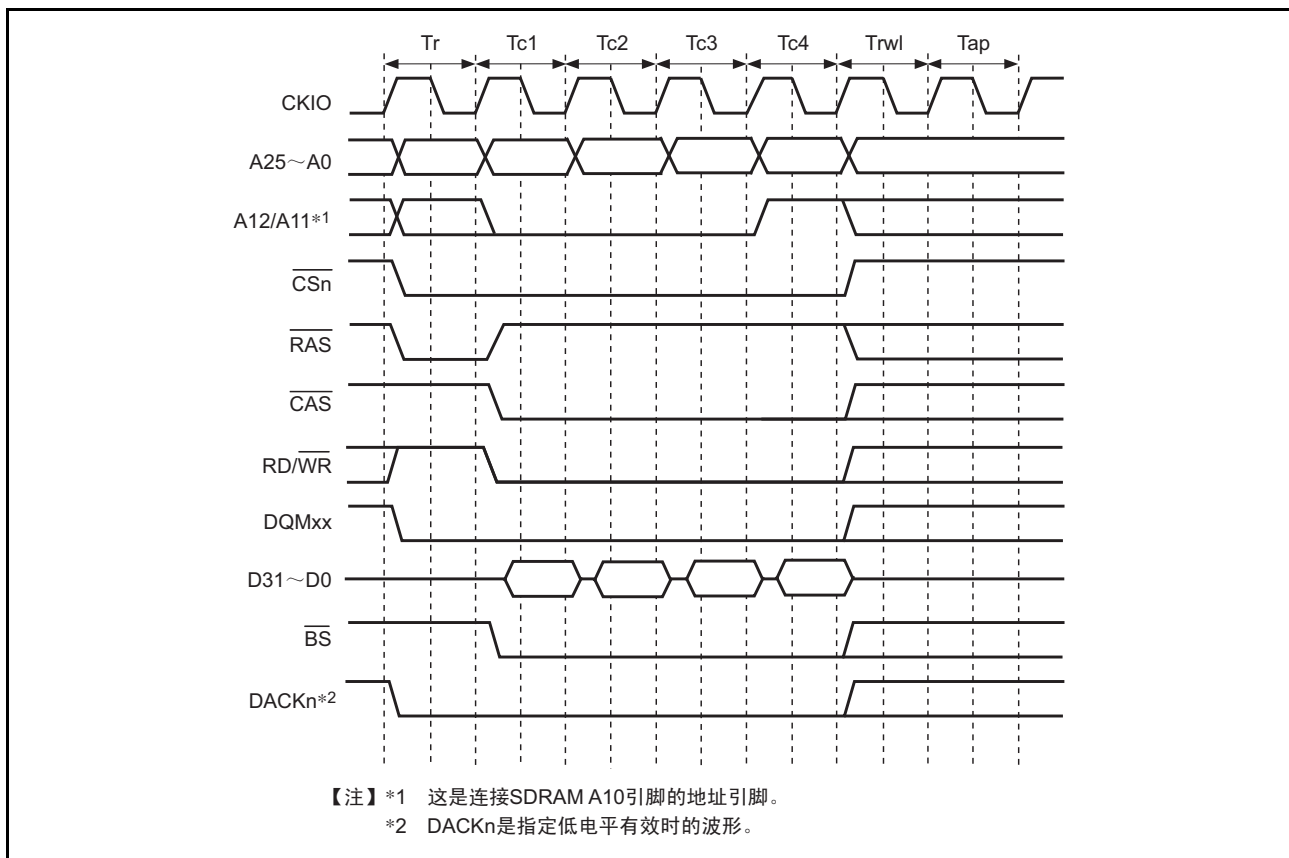


图 10.19 突发写的基本时序（自动预充电）

(6) 单次写

在高速缓存无效空间中数据总线宽度不小于存取长度时，以 1 次写存取结束。因为设定为单次写或突发长度 1 的突发写，所以 SDRAM 只写需要的数据。进行 1 次就结束的写存取称为单次写。单次写的基本时序图如图 10.20 所示。

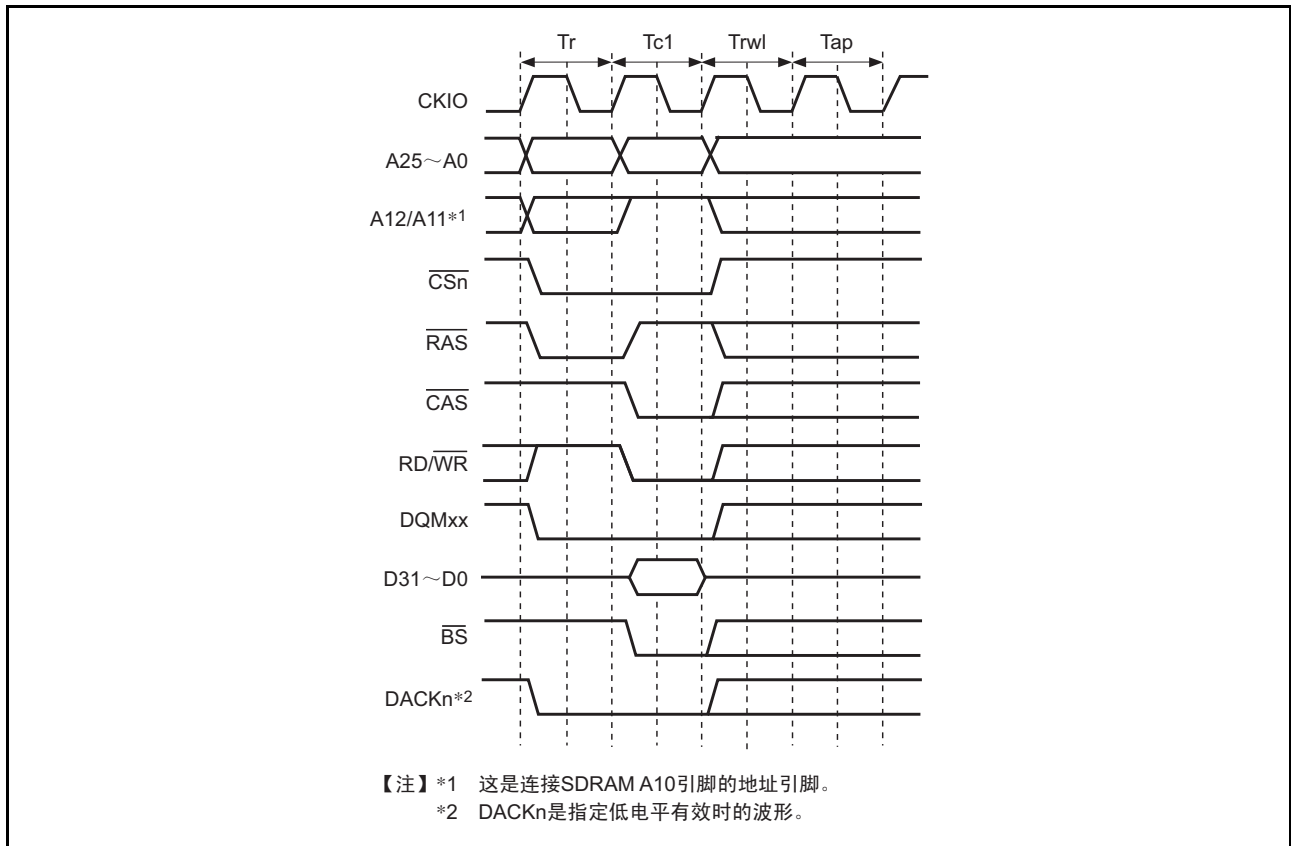


图 10.20 单次写的基本时序（自动预充电）

(7) 存储体激活

能使用 SDRAM 的存储体功能对同一行地址执行高速存取。当 SDCR 的 BACTV 位为“1”时，使用不自动预充电的命令（READ 或者 WRIT）进行存取，称之为存储体激活功能。但是，存储体激活功能只对区域 3 有效。当将区域 3 设定为存储体激活模式时，必须将区域 2 空间设定为正规空间或者带字节选择的 SRAM；当将区域 2 和区域 3 都设定为 SDRAM 时，必须设定为自动预充电模式。

当使用存储体激活功能时，即使存取结束也不进行预充电。如果存取同一存储体的同一行地址，就能不发行 ACTV 命令而直接发行 READ 命令或者 WRIT 命令。因为 SDRAM 内部被分成多个存储体，所以能预先将各存储体的每个行地址设定为激活状态。如果下一次存取是不同的行地址，就先通过发行 PRE 命令进行相应存储体的预充电，在预充电结束后按顺序发行 ACTV 命令、READ 命令或者 WRIT 命令。当继续存取不同的行地址时，可能因为在发生存取请求后进行预充电，反而延长了存取时间。通过 CS3WCR 的 WTRP[1:0] 位指定从发行 PRE 命令到发行 ACTV 命令的周期数。

如果在写操作时进行自动预充电，就不能在发行 WRITA 命令后的 $Trwl+Tap$ 周期内对同一存储体发行命令。当使用存储体激活模式时，如果是同一行地址，就能连续发行 READ 命令或者 WRIT 命令。因此，每写 1 次就能缩短 $Trwl+Tap$ 周期的周期数。

各存储体处于激活状态的时间（tRAS）有限制。根据程序的执行，如果不能保证在维持此限制的周期中存取不同的行地址，就需要将刷新周期设定为小于等于 tRAS。

不自动预充电的突发读周期、同一行地址的突发读周期和不同行地址的突发读周期分别如图 10.21、图 10.22 和图 10.23 所示，不自动预充电的单次写周期、同一行地址的单次写周期和不同行地址的单次写周期分别如图 10.24、图 10.25 和图 10.26 所示。

在图 10.22 中，在发行 READ 命令的 Tc 周期前插入不进行任何操作的 $Tnop$ 周期，这是为了在从 SDRAM 读数据时，使指定读取字节的 DQM_{xx} 信号保持 2 个周期的等待时间。当 CAS 等待时间至少为 2 个周期时，即使在 Tc 周期以后 DQM_{xx} 信号有效，也能保持 2 个周期的等待时间，所以不插入 $Tnop$ 周期。

只从存取被设定存储体激活功能的空间的各存储体来看，只要连续存取同一行地址，就从图 10.21 或者图 10.24 开始，重复图 10.22 或者图 10.25。在此期间，即使存取不同空间或者不同存储体也不影响。如果在存储体激活模式中存取其他行地址，就执行图 10.23 或者图 10.26 的总线周期，而不是图 10.22 或者图 10.25 的总线周期。即使在存储体激活模式中，也在刷新周期后或者在通过总线仲裁释放总线后，全部存储体都为非激活状态。

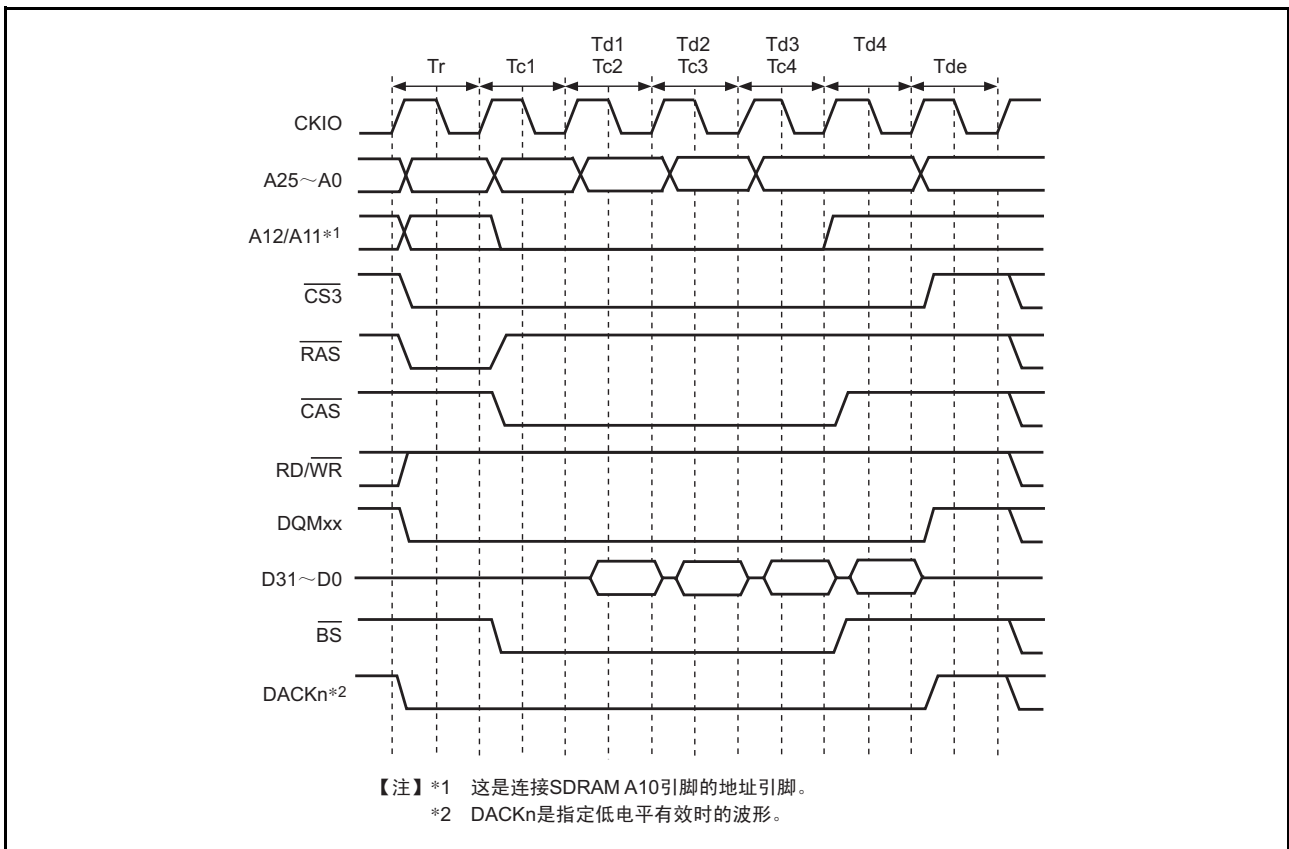


图 10.21 突发读的时序（存储体激活、不同存储体、CAS 等待时间 1）

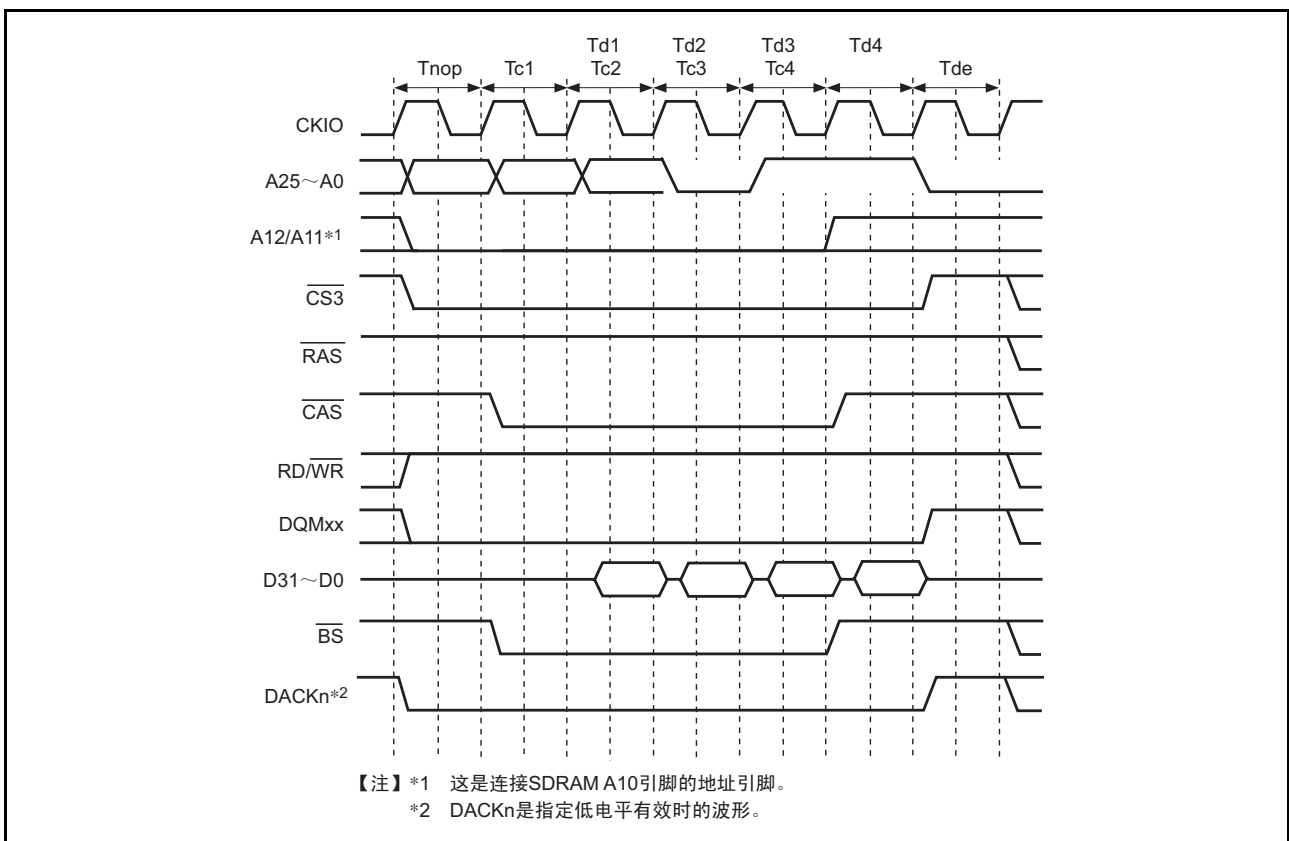


图 10.22 突发读的时序（存储体激活、同一存储体的同一行地址、CAS 等待时间 1）

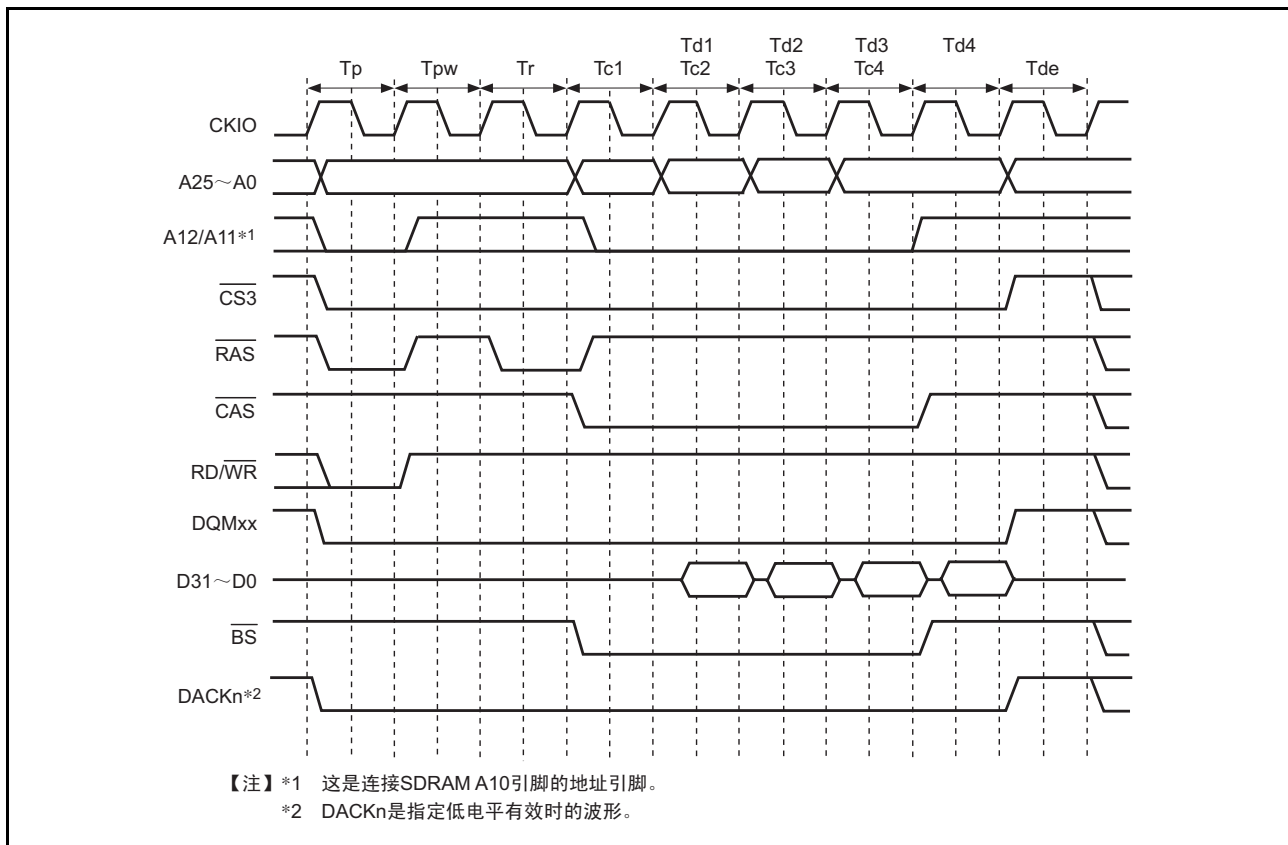


图 10.23 突发读的时序（存储体激活、同一存储体的不同行地址、CAS 等待时间 1）

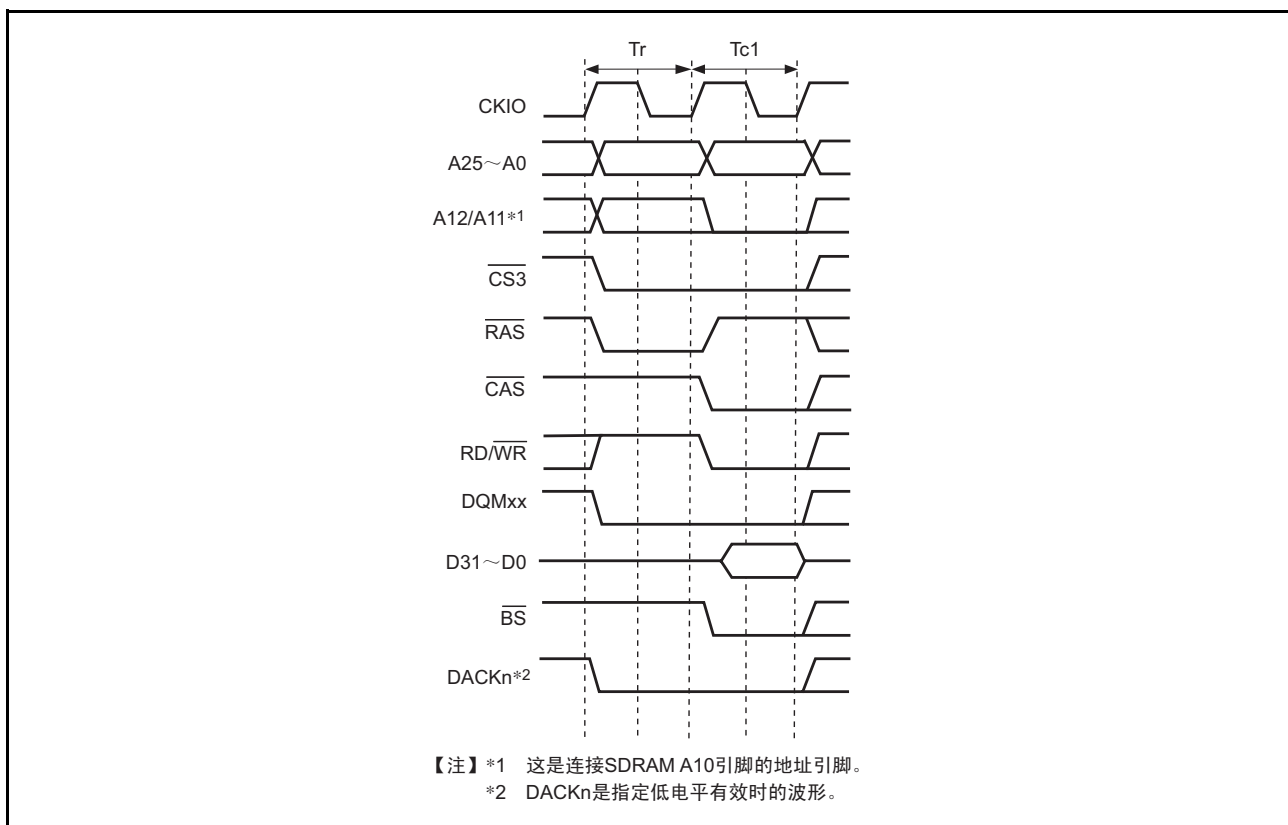


图 10.24 单次写的时序（存储体激活、不同存储体）

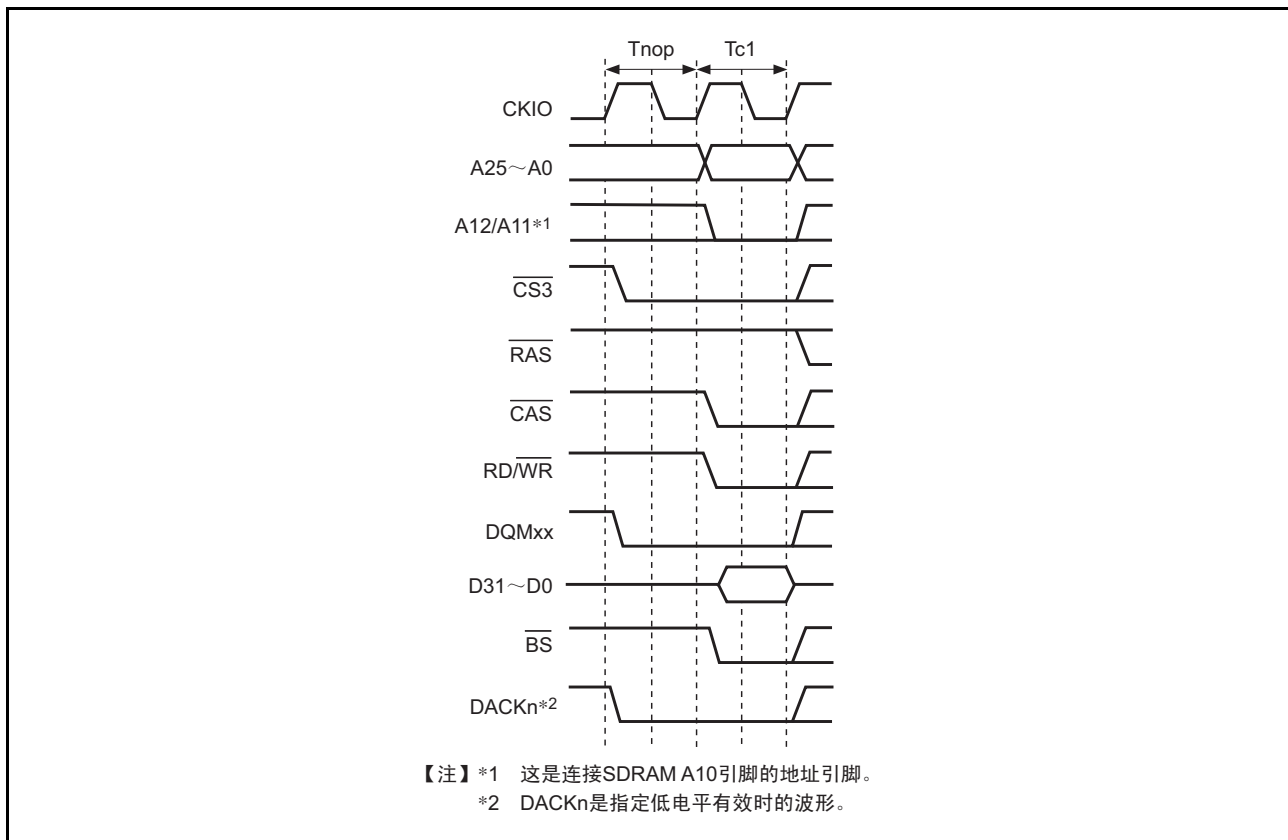


图 10.25 单次写的时序（存储体激活、同一存储体的同一行地址）

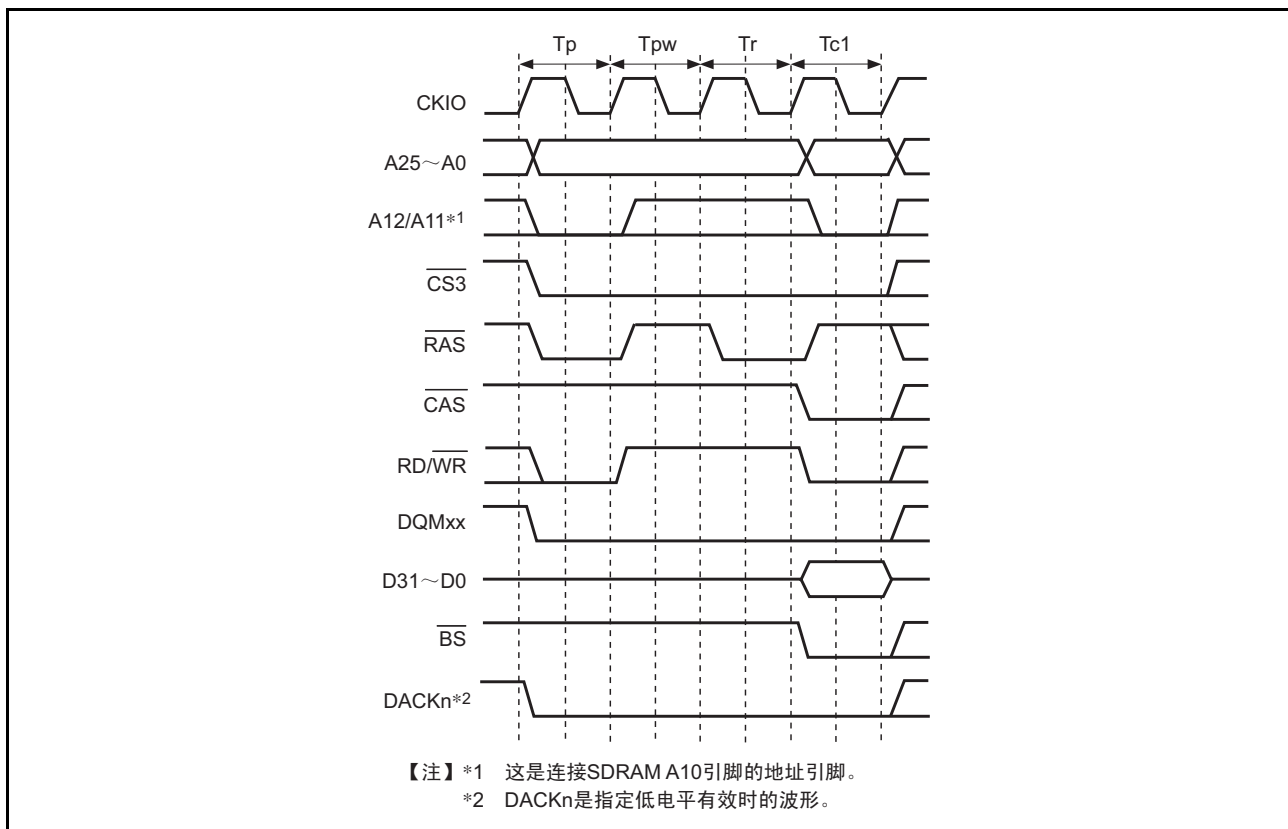


图 10.26 单次写的时序（存储体激活、同一存储体的不同行地址）

(8) 刷新

此模块具有控制 SDRAM 刷新的功能。能通过将 SDCR 的 RFSH 位设定为“1”并且将 RMODE 位设定为“0”进行自动刷新，还能通过设定 RTCSR 的 RRC[2:0] 位进行连续刷新。另外，在长时间不存取 SDRAM 时，能通过将 RFSH 位和 RMODE 位同时置“1”，启动功耗小的自刷新。

(a) 自动刷新

以 RTCSR 的 CKS[2:0] 位选择的输入时钟和 RTCOR 的设定值所定的间隔，进行 RTCSR 的 RRC[2:0] 位设定次数的刷新。为了符合所使用的 SDRAM 的刷新闻隔规定，必须设定各寄存器。先设定 RTCOR、RTCNT、SDCR 的 RFSH 位和 RMODE 位，然后设定 RTCSR 的 CKS[2:0] 位和 RRC[2:0] 位。如果通过 CKS[2:0] 位选择输入时钟，RTCNT 就从当时的值开始递增计数。RTCNT 的值随时和 RTCOR 的值进行比较，如果两者的值相同，就产生刷新请求并且执行 RRC[2:0] 位设定次数的自动刷新，同时将 RTCNT 清“0”，重新开始递增计数。

自动刷新周期的时序如图 10.27 所示。如果在启动自动刷新时有正在预充电的存储体，就在等待其结束后在 T_p 周期发行 PALL 命令，使全部的存储体从激活状态进入预充电状态。接着，插入 CS3WCR 的 WTRP[1:0] 位所指定的空闲周期数，然后在 T_{tr} 周期发行 REF 命令。在 T_{tr} 周期之后和 CS3WCR 的 WTRC[1:0] 位指定的周期数之间不发行新命令。为了符合 SDRAM 刷新周期时间的规定 (t_{RC})，需要设定 WTRC[1:0] 位。当 CS3WCR 的 WTRP[1:0] 位的设定值至少是 1 个周期时，就在 T_p 周期和 T_{tr} 周期之间插入空闲周期。

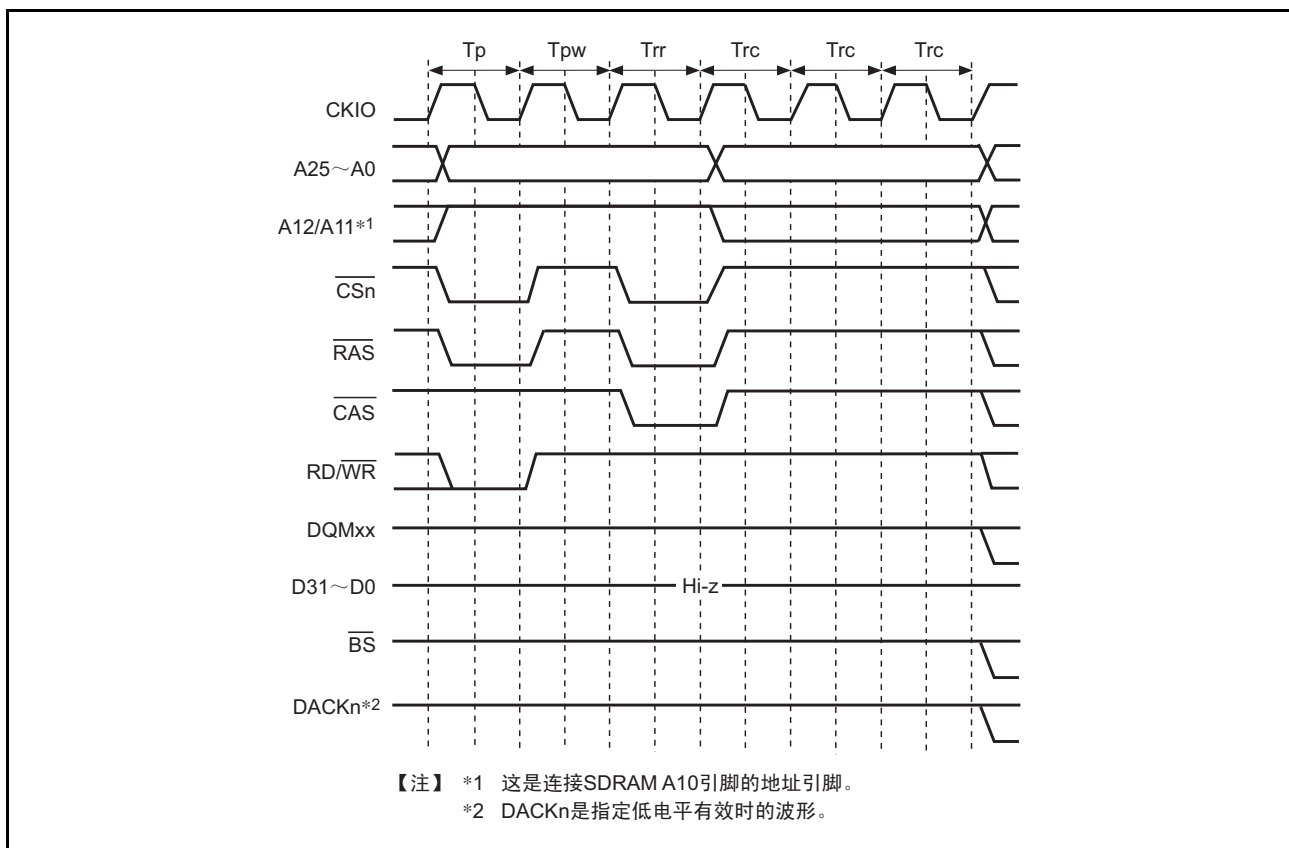


图 10.27 自动刷新的时序

(b) 自刷新

自刷新是在 SDRAM 内部生成刷新时序和刷新地址的一种待机模式。通过将 SDCR 的 RFSH 位和 RMODE 位同时置“1”进行启动。如果在启动自刷新时有正在预充电的存储体，就等待其结束后在 T_p 周期发行 PALL 命令。接着，在插入 CS3WCR 的 WTRP[1:0] 位设定的空闲周期后发行 SELF 命令。不能在自刷新状态期间存取 SDRAM。通过将 RMODE 位置“0”来解除自刷新，在解除自刷新之后和 CS3WCR 的 WTRC[1:0] 位指定的周期数之间不发行命令。

自刷新的时序如图 10.28 所示。在解除自刷新后，必须立即进行设定以便能以正确的间隔执行自动刷新。当从自动刷新状态进入自刷新状态时，如果在解除自刷新时 RFSH 为“1”并且 RMODE 为“0”，就重新开始自动刷新。在从解除自刷新到开始自动刷新需要时间的情况下，能通过将 (RTCOR 的值 -1) 设定到 RTCNT，立即开始自动刷新。

在设定为自刷新后，即使将 LSI 设定为待机状态还继续保持自刷新状态，并且在通过中断从待机状态返回后也保持自刷新状态。但是，将 CMNCR 寄存器的 HIZCNT 位置“1”，并且即使在待机状态下也需要驱动 CKE 引脚。

在更改 PLL 电路的倍频率时，CKIO 输出不稳定而变为低电平。有关 CKIO 输出，请参照“5. 时钟脉冲振荡器”。能通过更改倍频率前设置为自刷新状态，保持 SDRAM 的内容。

不能通过手动复位解除自刷新状态。

在上电复位时，因为对此模块的寄存器进行初始化，所以解除自刷新的状态。

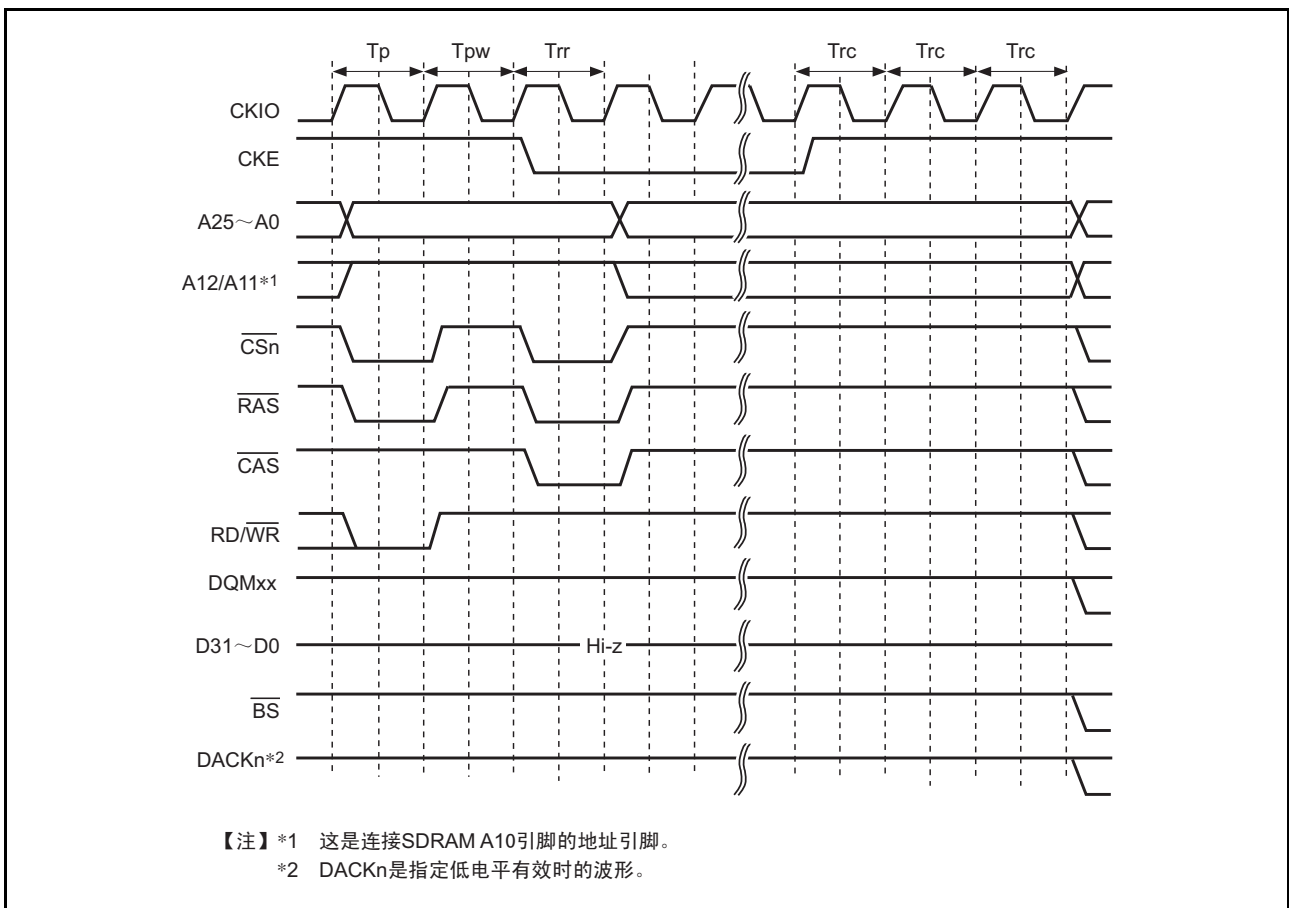


图 10.28 自刷新的时序

(9) 刷新请求和总线周期的关系

如果在执行总线周期的过程中发生刷新请求，就在等到总线周期结束后执行刷新；如果在通过总线仲裁功能释放总线时发生刷新请求，就在等到获得总线权后执行刷新。必须注意：如果在超过刷新间隔的规定时间后外部设备仍然没有返回总线权，就不能进行刷新并且无法保证 SDRAM 的内容。

如果在等待执行刷新的状态下发生新的刷新请求，就取消前一个刷新请求。为了正确地进行刷新，需要长于刷新间隔的总线周期或者不占有总线权。

即使在自刷新时发生总线权请求，也不在解除自刷新前释放总线。

(10) 掉电模式

如果将 SDCR 的 PDOWN 位置“1”，就在不进行存取时将 CKE 置为 L 电平，使 SDRAM 转移到掉电模式，从而能在不进行存取时大幅度地抑制功耗。但是，在发生存取时，为了解除 SDRAM 的掉电模式，需要插入使 CKE 有效的周期，因此会增加 1 个周期的开销。掉电模式的存取时序如图 10.29 所示。

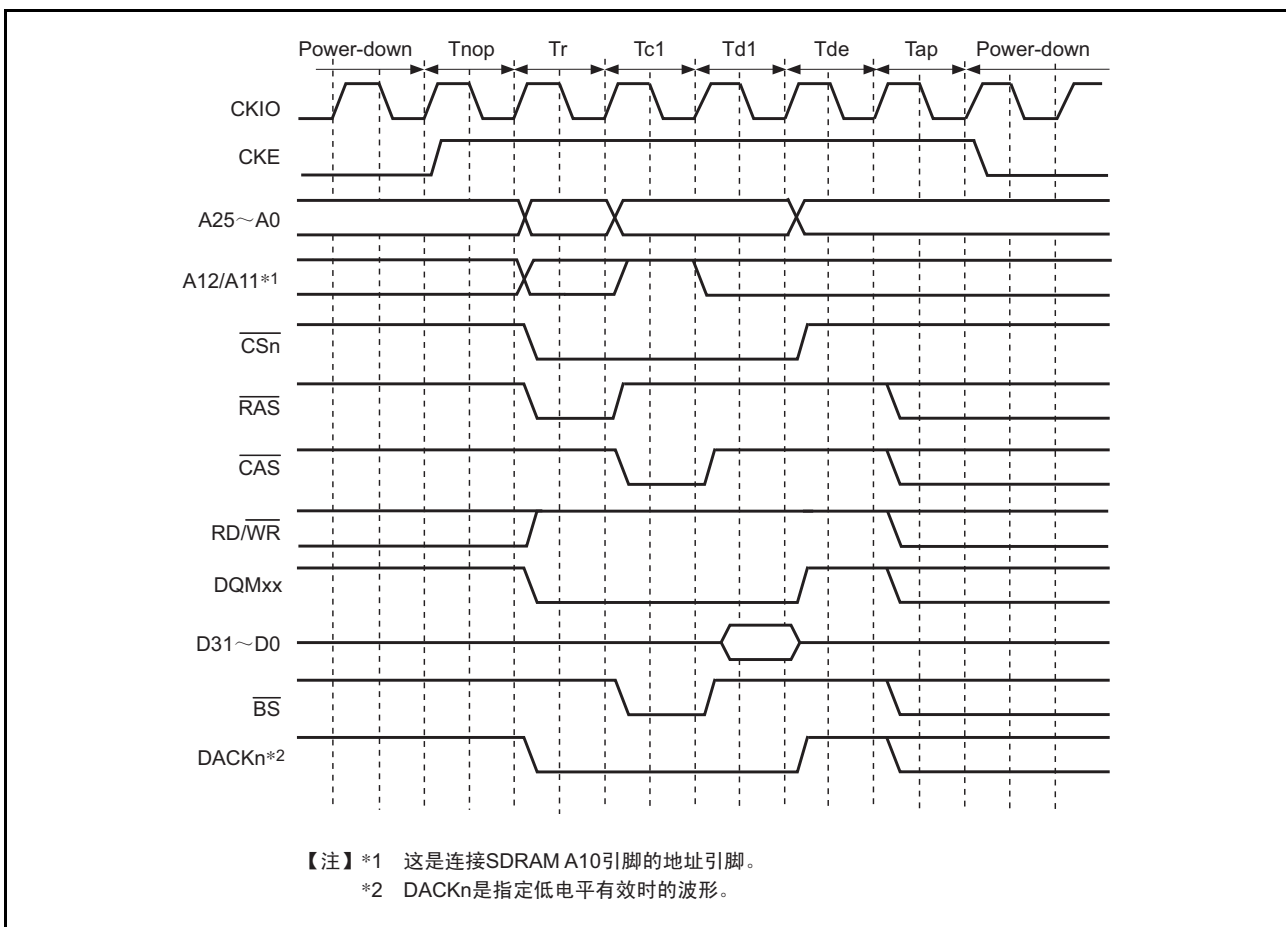


图 10.29 掉电模式的存取时序

(11) 上电顺序

为了使用 SDRAM，需要在上电后并且在所使用的 SDRAM 规定的暂停间隔后对 SDRAM 进行模式的设定。必须通过上电复位生成电路或者软件等实现暂停间隔。

为了正确地对 SDRAM 进行初始化，先设定此模块的寄存器，然后写 SDRAM 的模式寄存器。在设定 SDRAM 的模式寄存器时，通过 \overline{CSn} 、 \overline{RAS} 、 \overline{CAS} 和 $\overline{RD}/\overline{WR}$ 的组合，将当时的地址信号值取到 SDRAM。假设要设定的值为 X，通过以字为单位对 X+(区域 2: 地址 H'FFFC4000, 区域 3: 地址 H'FFFC5000) 进行写操作，就能将值 X 写到 SDRAM 的模式寄存器，此时忽视写数据。要设定本 LSI 支持的突发读/单次写（突发长度 1）或者突发读/突发写（突发长度 1）、CAS 等待时间为 2~3、Wrap 型为顺序以及突发长度 1 时，将任意数据字写到表 10.18 所示的存取地址，此时给外部地址引脚 A12 以上的位输出“0”。

表 10.18 写 SDRAM 模式寄存器时的存取地址

- 区域 2 的设定

突发读/单次写（突发长度 1）的情况

数据总线宽度	CAS 等待时间	存取地址	外部地址引脚
16 位	2	H'FFFC4440	H'0000440
	3	H'FFFC4460	H'0000460
32 位	2	H'FFFC4880	H'0000880
	3	H'FFFC48C0	H'00008C0

突发读/突发写（突发长度 1）的情况

数据总线宽度	CAS 等待时间	存取地址	外部地址引脚
16 位	2	H'FFFC4040	H'0000040
	3	H'FFFC4060	H'0000060
32 位	2	H'FFFC4080	H'0000080
	3	H'FFFC40C0	H'00000C0

- 区域 3 的设定

突发读/单次写（突发长度 1）的情况

数据总线宽度	CAS 等待时间	存取地址	外部地址引脚
16 位	2	H'FFFC5440	H'0000440
	3	H'FFFC5460	H'0000460
32 位	2	H'FFFC5880	H'0000880
	3	H'FFFC58C0	H'00008C0

突发读/突发写（突发长度 1）的情况

数据总线宽度	CAS 等待时间	存取地址	外部地址引脚
16 位	2	H'FFFC5040	H'0000040
	3	H'FFFC5060	H'0000060
32 位	2	H'FFFC5080	H'0000080
	3	H'FFFC50C0	H'00000C0

模式寄存器的设定时序如图 10.30 所示。首先发行全部存储体的预充电命令 (PALL)，其次发行 8 次自动刷新命令 (REF)，最后发行模式寄存器的写命令 (MRS)。在 PALL 和第 1 次 REF 之间插入 CS3WCR 的 WTRP[1:0] 位所设个数的空闲周期，在 REF 和 REF 之间以及第 8 次 REF 和 MRS 之间插入 CS3WCR 的 WTRC[1:0] 位所设个数的空闲周期。另外，在 MRS 和下一次发行的命令之间至少插入 1 个空闲周期。

SDRAM 必须在接通电源后并且在进行全部存储体的预充电 (PALL) 前确保一定的空闲时间，所需的空闲时间请参照 SDRAM 的使用手册。如果复位信号的脉冲宽度长于此空闲时间，立即设定模式寄存器也没关系，但是如果短于空闲时间就需要注意。

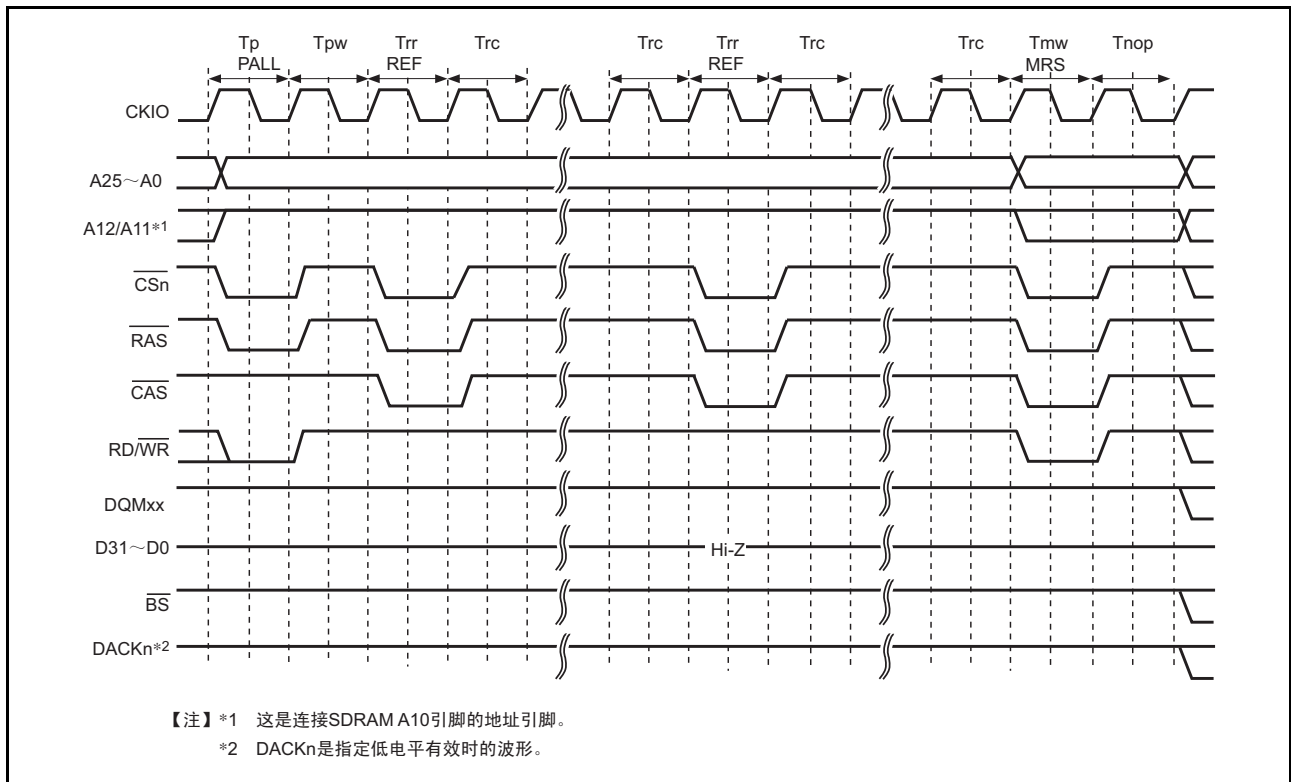


图 10.30 SDRAM 模式寄存器的写时序 (符合 JEDEC)

(12) 低功耗 SDRAM

低功耗 SDRAM 是能通过和正规 SDRAM 相同的协议进行存取的存储器。

低功耗 SDRAM 和正规 SDRAM 规格的不同点是：对于在自刷新时将部分存储器置为自刷新状态的部分刷新以及根据用户使用条件 (温度) 进行的刷新时实现低功率化等的控制，能进行详细的设定。部分刷新对即使在工作区 (某个特定区域除外) 内丢失数据也没问题的系统有效。详细内容请参照所使用的低功耗 SDRAM 数据表。

低功耗 SDRAM 除了有和正规 SDRAM 相同的模式寄存器以外，还有扩展模式寄存器。扩展模式寄存器写命令称为 EMRS，本 LSI 支持 EMRS 命令的发行。

按下表发行 EMRS。例如，当用长字给 H'FFFC5XX0 写数据 "H'0YYYYYYY" 时，就按 PALL→REF×8→MRS→EMRS 的顺序对 CS3 空间发行命令。此时，发行 MRS 时的地址为 "H'0000XX0"，发行 EMRS 时的地址为 "H'YYYYYYY"。当用长字给 H'FFFC5XX0 写数据 "H'1YYYYYYY" 时，就按 PALL→MRS→EMRS 的顺序对 CS3 空间发行命令。

表 10.19 发行 EMRS 命令时的输出地址

发行的命令	存取地址	存取数据	写存取长度	发行 MRS 命令时的地址	发行 EMRS 命令时的地址
CS2 MRS	H'FFFC4XX0	H'*****	16 位	H'0000XX0	-----
CS3 MRS	H'FFFC5XX0	H'*****	16 位	H'0000XX0	-----
CS2 MRS+EMRS (有刷新)	H'FFFC4XX0	H'0YYYYYYYY	32 位	H'0000XX0	H'YYYYYYYY
CS3 MRS+EMRS (有刷新)	H'FFFC5XX0	H'0YYYYYYYY	32 位	H'0000XX0	H'YYYYYYYY
CS2 MRS+EMRS (无刷新)	H'FFFC4XX0	H'1YYYYYYYY	32 位	H'0000XX0	H'YYYYYYYY
CS3 MRS+EMRS (无刷新)	H'FFFC5XX0	H'1YYYYYYYY	32 位	H'0000XX0	H'YYYYYYYY

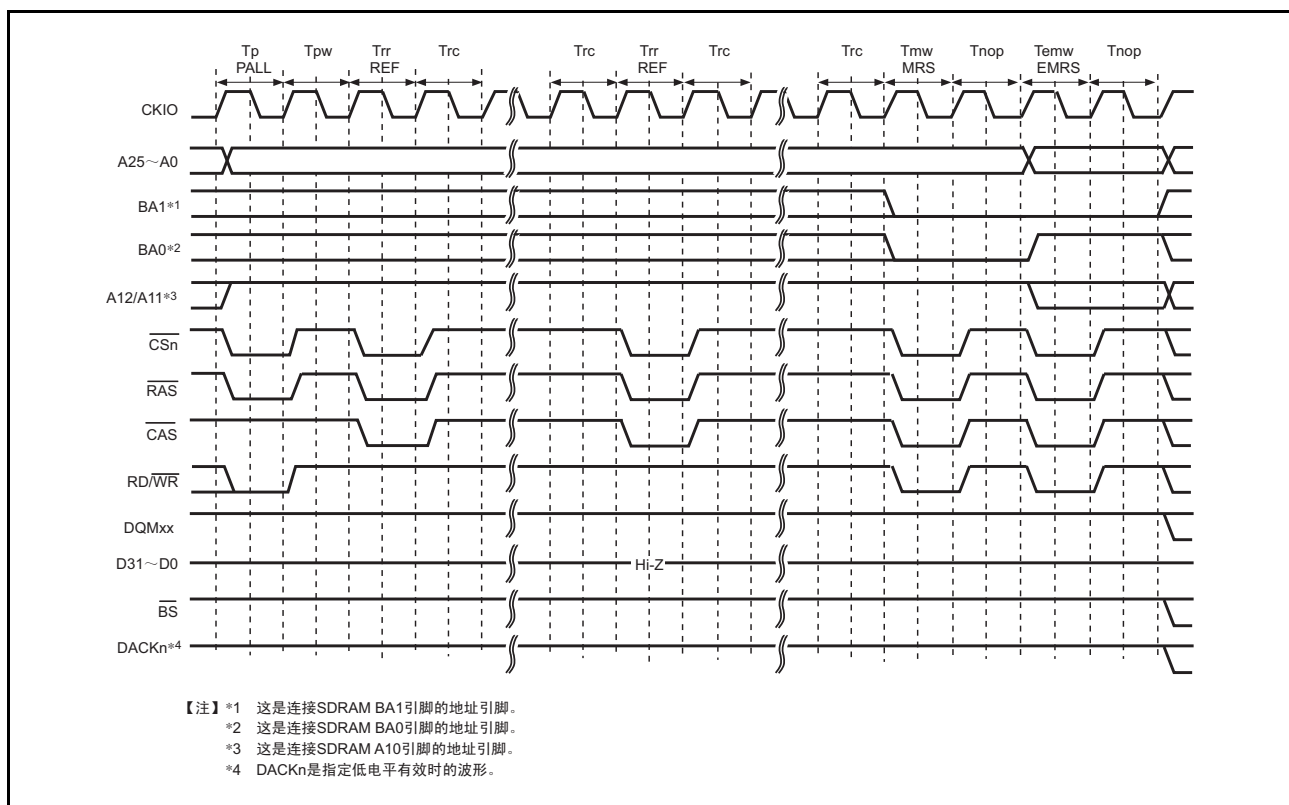


图 10.31 发行 EMRS 命令的时序

- 深度掉电模式

低功率 SDRAM 有称为深度掉电模式的低功耗模式。

部分自刷新只对某个特定区域进行自刷新，而在深度掉电模式中对整个存储器不进行自刷新。

此模式对将整个存储器作为工作区域的系统有效。

如果在将 SDCR 的 DEEP 位置 “1” 并且将 RFSH 位置 “1” 的状态下给 RMODE 位写 “1”，低功率 SDRAM 就转移到深度掉电模式。如果给 RMODE 位写 “0”，CKE 就变为高电平，解除深度掉电模式。在解除深度掉电后，必须先重新进行上电顺序的处理，然后才能进行存取。

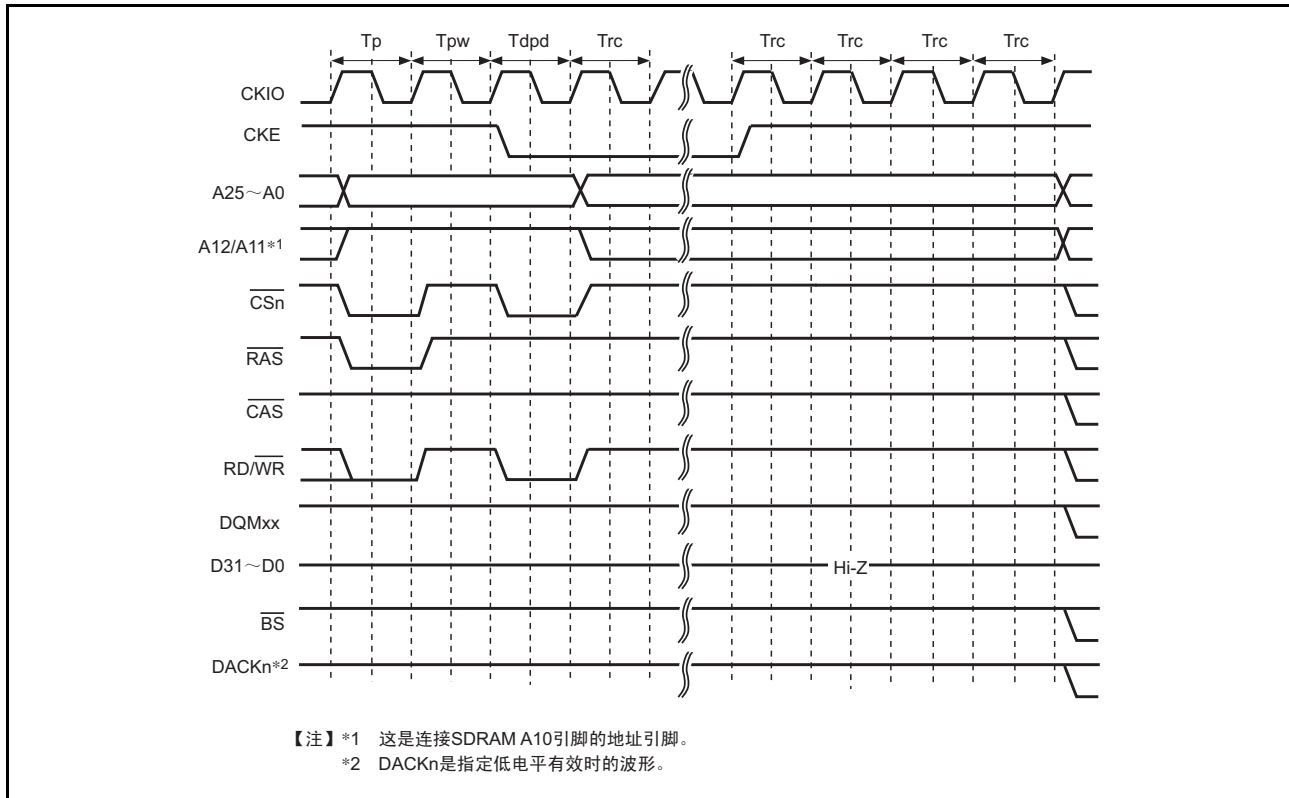


图 10.32 深度掉电模式的转移时序

10.5.7 突发 ROM（异步）接口

突发 ROM（异步）接口通过称为突发模式或者页模式等的地址转换，用于存取具有高速读功能的存储器。基本上进行和正规空间相同的存取，但是在最初的周期结束时，不将 \overline{RD} 信号置为无效而只转换地址，进行第 2 次以后的存取。第 2 次以后的存取在 CKIO 的下降沿发生地址变化。

在最初的存取周期中插入 CSnWCR 的 W[3:0] 位所设个数的等待周期。在第 2 次以后的存取周期中插入 CSnWCR 的 BW[1:0] 位所设个数的等待周期。

当存取突发 ROM（异步）时， \overline{BS} 信号只对最初的存取周期有效，并且外部等待输入也只对最初的存取周期有效。

当通过突发 ROM（异步）接口进行单次存取（不进行突发运行）和写存取时，存取时序和正规空间相同。总线宽度、存取长度和突发数的关系如表 10.20 所示，时序图如图 10.33 所示。

表 10.20 总线宽度、存取长度和突发数的关系

总线宽度	存取长度	CSnWCR.BST[1:0] 位	突发数	存取次数
8 位	8 位	没有影响	1	1
	16 位	没有影响	2	1
	32 位	没有影响	4	1
	16 字节	00	16	1
		01	4	4
16 位	8 位	没有影响	1	1
	16 位	没有影响	1	1
	32 位	没有影响	2	1
	16 字节	00	8	1
		01	2	4
		10*	4	2
			2、4、2	3
32 位	8 位	没有影响	1	1
	16 位	没有影响	1	1
	32 位	没有影响	1	1
	16 位	没有影响	4	1

【注】* 在 16 位总线宽度、16 字节的存取长度并且 CSnWCR.BST[1:0] 位为“10”时，突发数和存取次数因存取起始地址而不同。当地址为“H'xxx0”或者“H'xxx8”时为 4-4 突发存取，当地址为“H'xxx4”或者“H'xxxC”时为 2-4-2 突发存取。

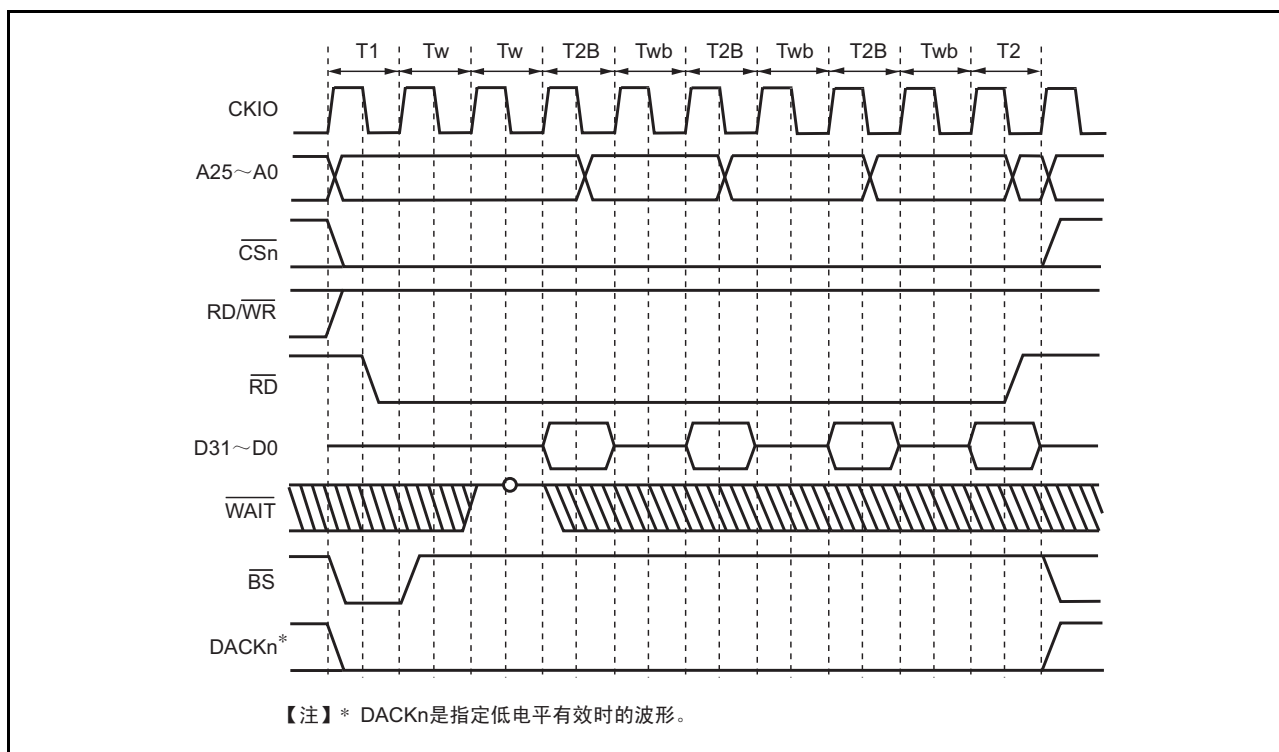


图 10.33 突发 ROM（异步）的存取
 （32 位总线宽度、16 字节传送（突发数 4）、初次存取等待 2、第 2 次以后的存取等待 1）

10.5.8 带字节选择的 SRAM 接口

带字节选择的 SRAM 接口是在读或者写的总线周期中输出字节选择引脚 (\overline{WEn}) 的存储器接口。此接口有 16 位数据引脚，用于存取有如 UB 或者 LB 的高位字节选择引脚和低位字节选择引脚的 SRAM。

当 CSnWCR 的 BAS 位为 “0” (初始值) 时，带字节选择的 SRAM 接口的写存取时序和正规空间接口相同。然而在进行读操作时， \overline{WEn} 引脚的时序和正规空间接口不同，从 \overline{WEn} 引脚输出字节选择信号。基本存取时序如图 10.34 所示。尤其在进行写操作时，按照字节选择引脚 (\overline{WEn}) 的时序写存储器，所以必须确认所使用的存储器数据表。

当 CSnWCR 的 BAS 位为 “1” 时， \overline{WEn} 引脚和 RD/WR 引脚的时序发生变化。基本存取时序如图 10.35 所示。尤其在进行写操作时，按照写允许引脚 ($\overline{RD/WR}$) 时序写存储器，所以必须通过设定 CSnWCR 的 HW[1:0] 位，确保从 RD/WR 无效到写数据的保持时间。软件等待设定时的存取时序如图 10.36 所示。

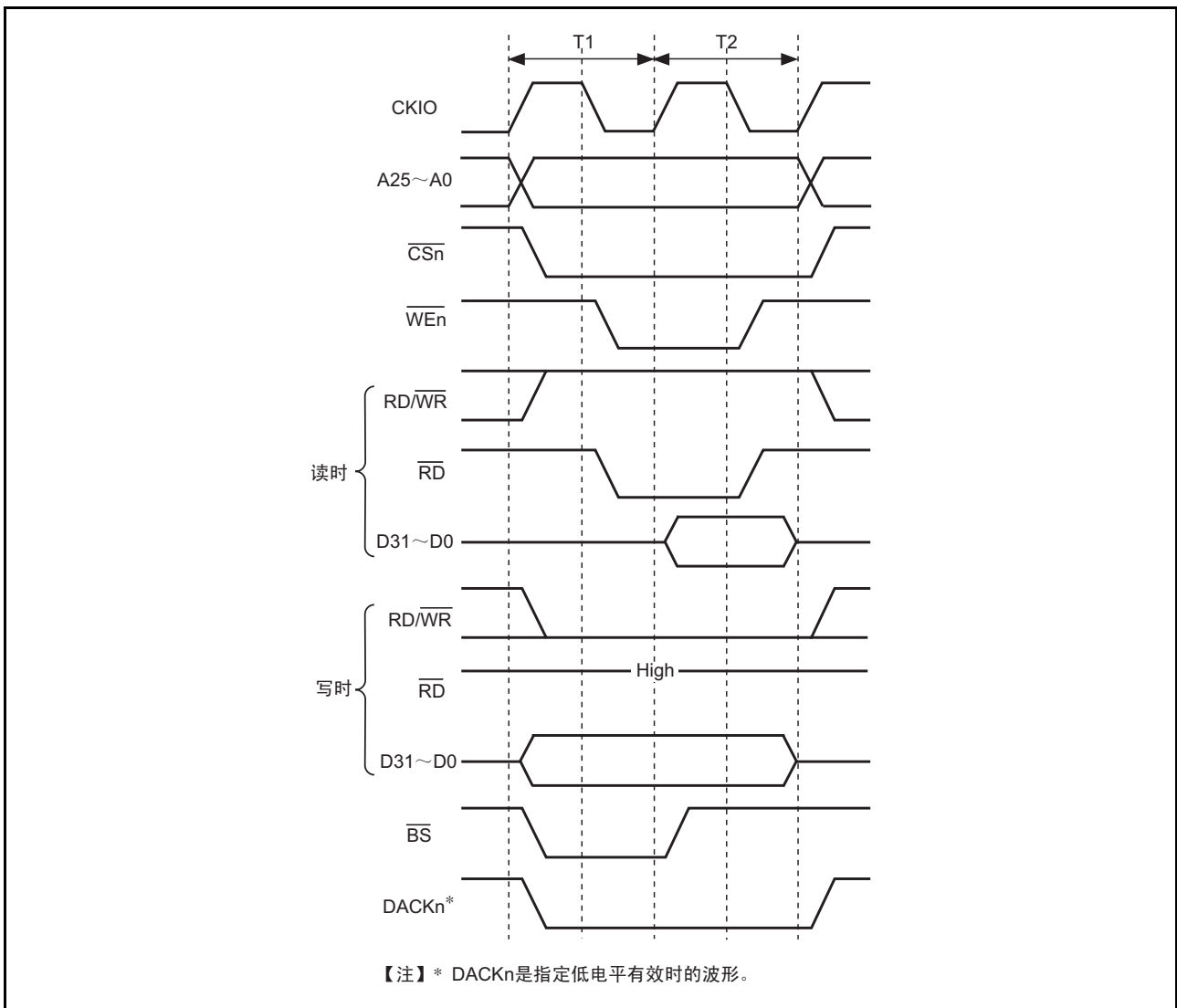


图 10.34 带字节选择的 SRAM 的基本存取时序 (BAS=0)

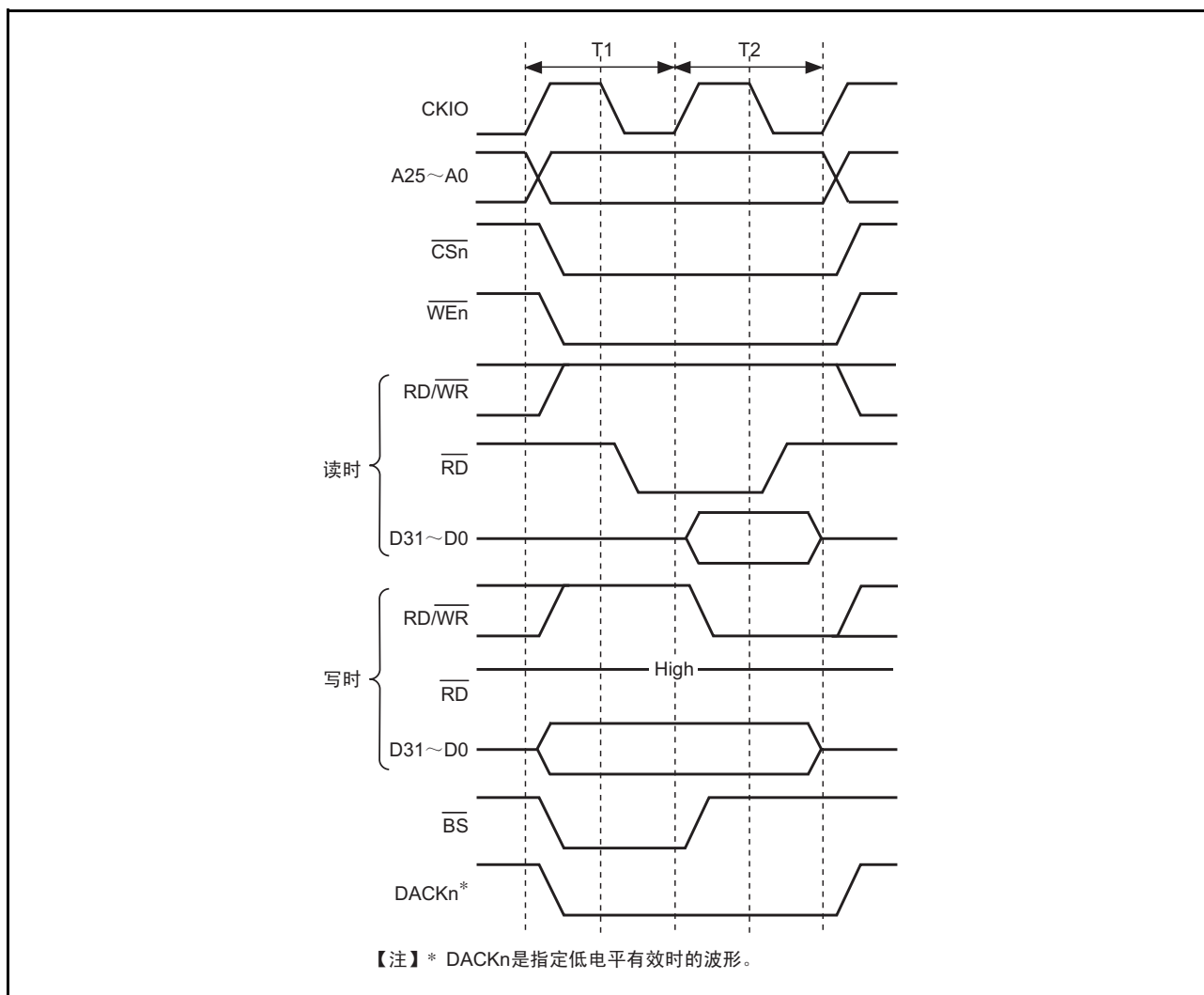


图 10.35 带字节选择的 SRAM 的基本存取时序 (BAS=1)

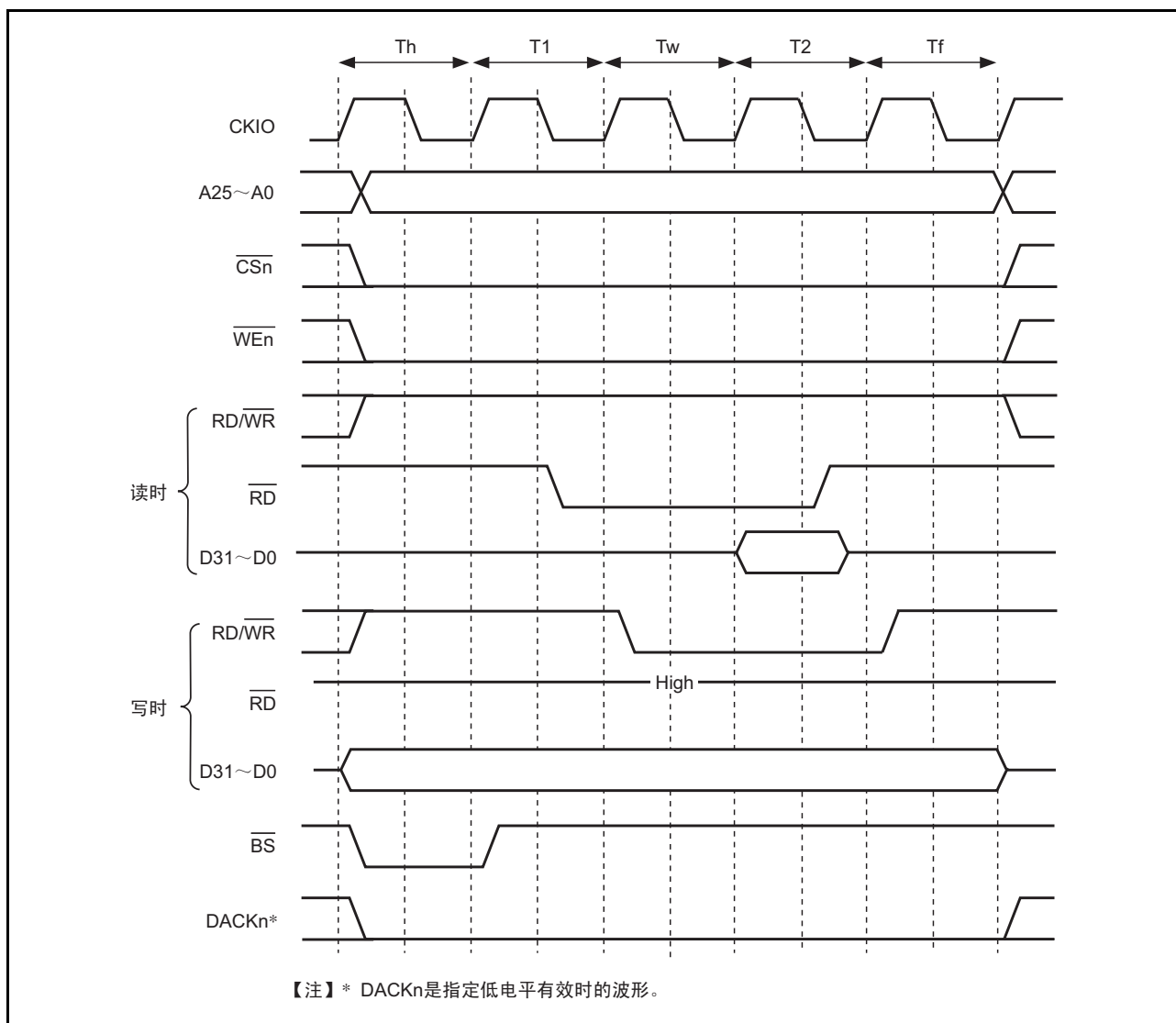


图 10.36 带字节选择的 SRAM 的等待时序 (BAS=1, SW[1:0]=01, WR[3:0]=0001, HW[1:0]=01)

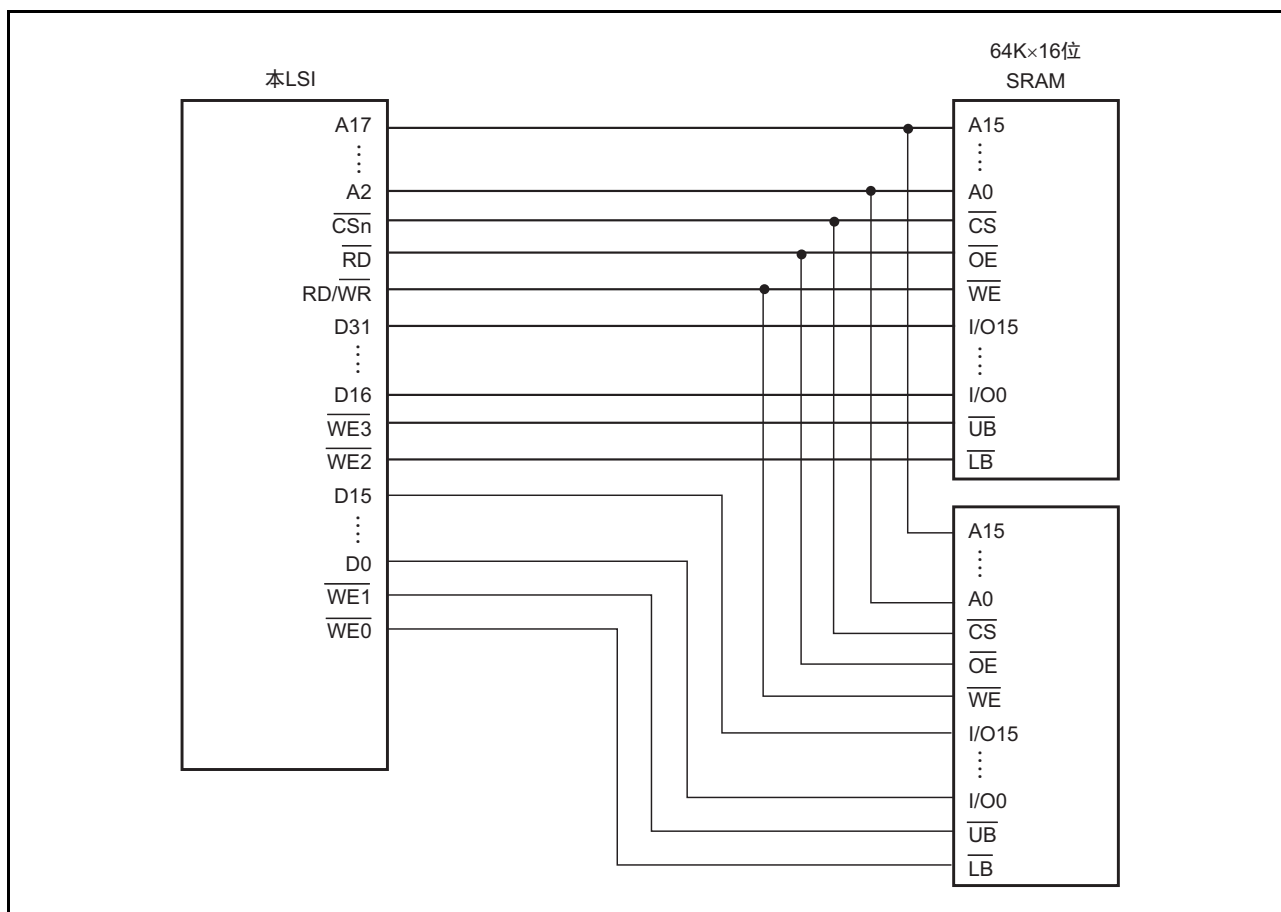


图 10.37 32 位数据带字节选择的 SRAM 连接例子

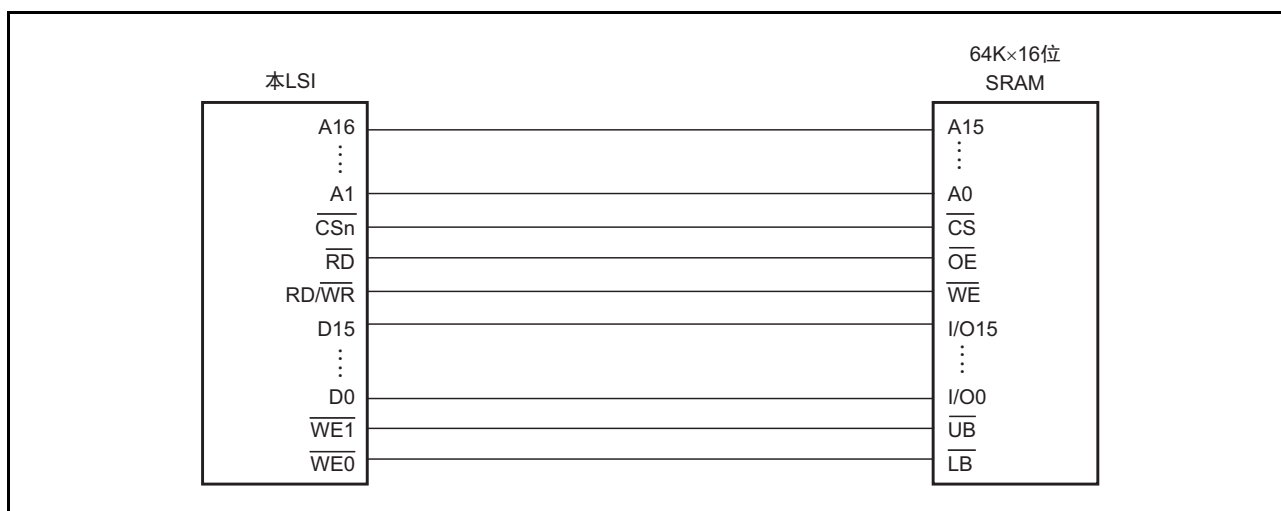


图 10.38 16 位数据带字节选择的 SRAM 连接例子

10.5.9 PCMCIA 接口

本LSI通过将CS5BCR的TYPE[2:0]位设定为“B'101”，将区域5设定为JEIDA规格Ver4.2(PCMCIA2.1Rev2.1)规定的IC存储卡和I/O卡接口，还能通过CS5WCR的SA[1:0]位，将各区域5的前32MB和后32MB设定为IC存储卡或者I/O卡接口。例如，当将CS5WCR的SA1位设定为“1”并且将CS5WCR的SA0位设定为“0”时，区域5的前32MB为IC存储卡接口而后32MB为I/O卡接口。

在使用PCMCIA接口时，必须通过CS5BCR的BSZ[1:0]位，将总线宽度设定为8位或者16位。

本LSI和PCMCIA卡的连接例子如图10.39所示。为了能进行PCMCIA卡的热插拔（在提供系统电源时插拔卡），需要在本LSI的总线接口和PCMCIA卡之间连接3态缓冲器。

因为在JEIDA和PCMCIA的规格中没有明确规定大端法模式的运行规格，所以本LSI大端法模式的PCMCIA接口为独自规定的接口规格。

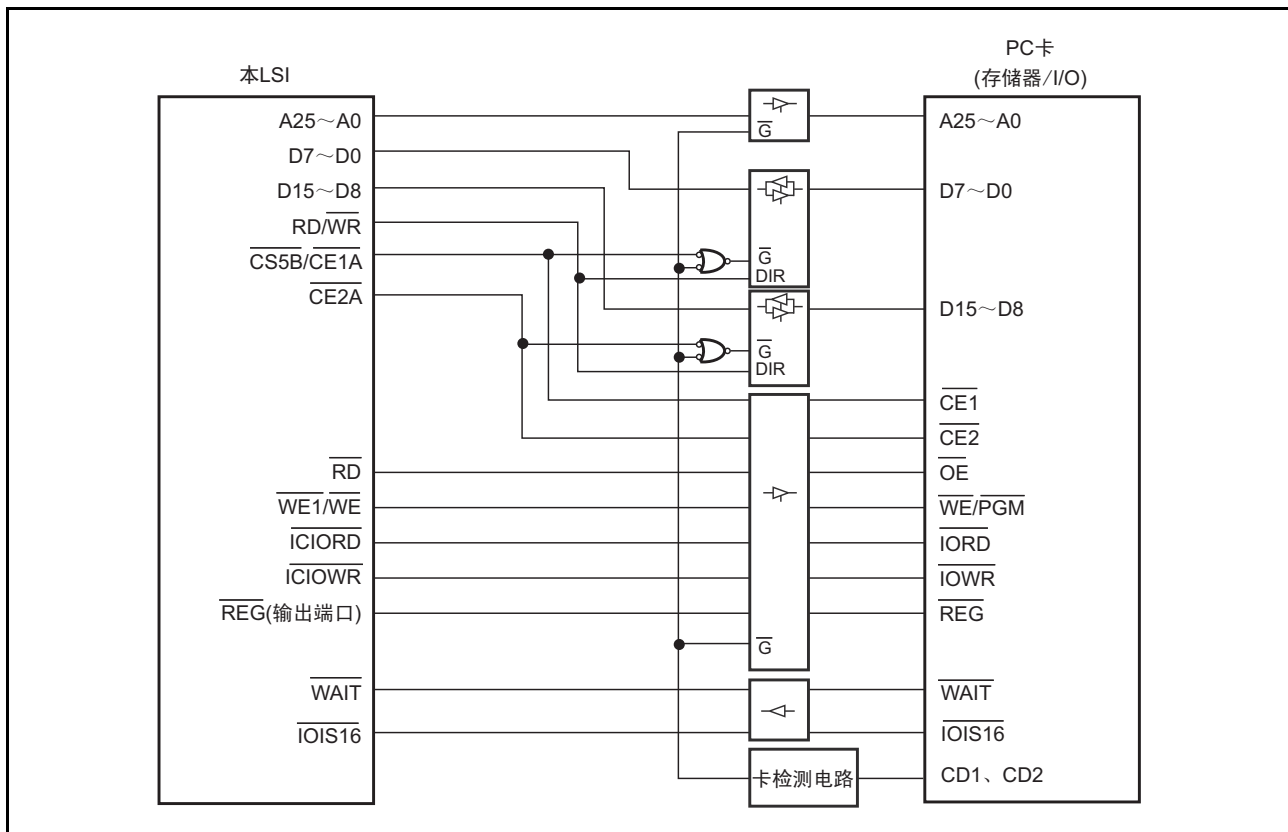


图 10.39 PCMCIA 接口的连接例子

(1) 存储卡接口的基本时序

PCMCIA 的 IC 存储卡接口的基本时序如图 10.40 所示。如果将区域 5 设定为 PCMCIA 接口，就能根据 CS5WCR 的 SA[1:0] 位的设定，能作为 IC 存储卡接口进行总线存取。如果外部总线频率（CKIO）变高，对于 \overline{RD} 和 \overline{WE} 来说，地址（A25 ~ A0）、卡的允许（ $\overline{CE1A}$ 、 $\overline{CE2A}$ ）以及写周期中的写数据（D15 ~ D0）的准备时间和保持时间就会不足。对此，本 LSI 能通过 CS5WCR 给区域 5 设定准备时间和保持时间。另外，和正规空间接口一样，能进行软件等待以及通过 \overline{WAIT} 引脚进行硬件等待。PCMCIA 存储器的总线等待时序如图 10.41 所示。

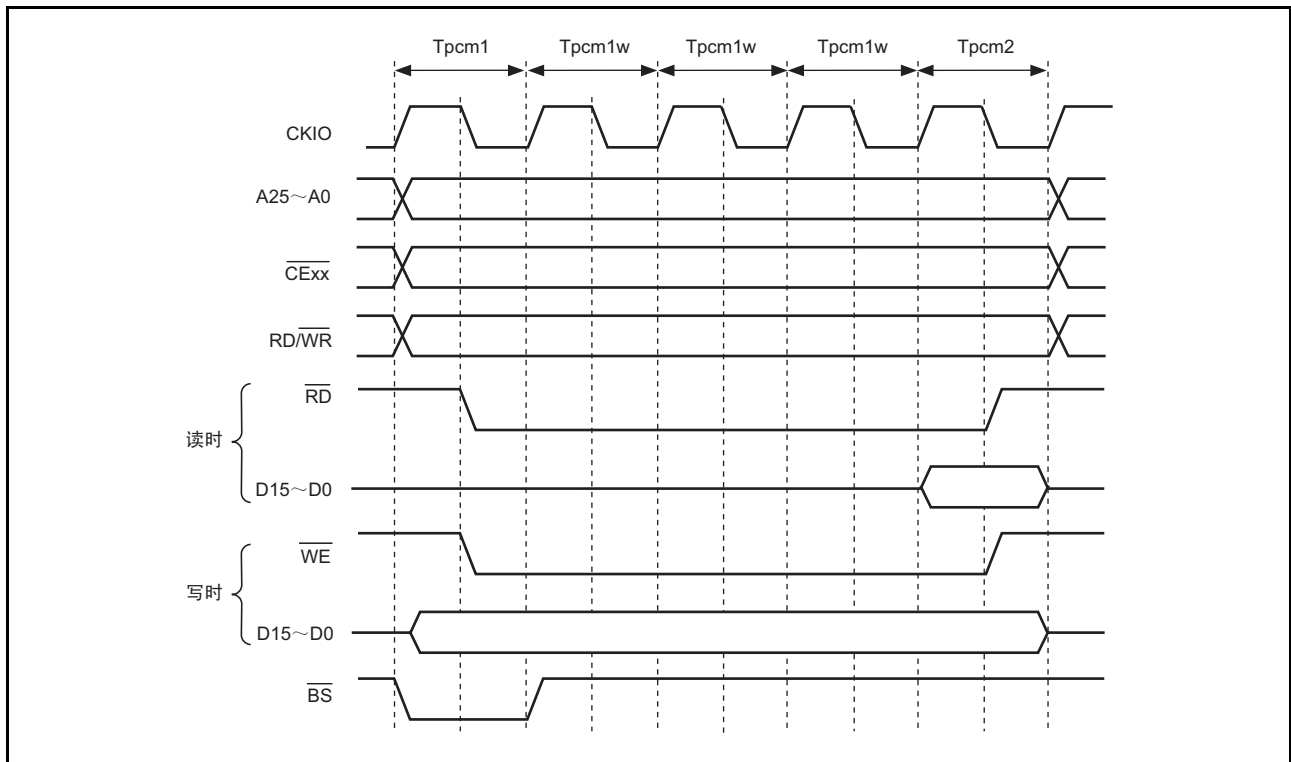


图 10.40 PCMCIA 存储卡接口的基本时序

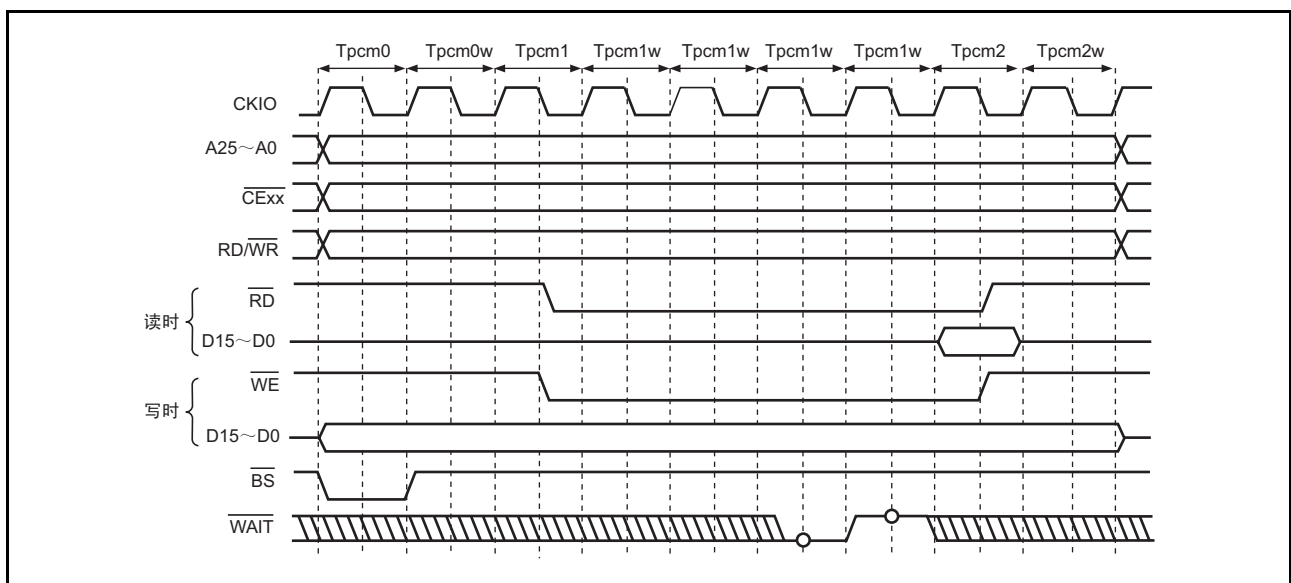


图 10.41 PCMCIA 存储卡接口的等待时序
(TED[3:0]=B'0010, PCW[3:0]=B'0000, TEH[3:0]=B'0001, 硬件等待 1)

利用端口等生成公共存储器和属性存储器的转换信号 $\overline{\text{REG}}$ 。另外，如图 10.42 所示，在公共存储器和属性存储器的合计存储空间不超过 32M 字节并且够用的情况下，通过将 A24 引脚用作 $\overline{\text{REG}}$ 信号，能将存储空间用作 16M 字节的公共存储空间和 16M 字节的属性存储空间。

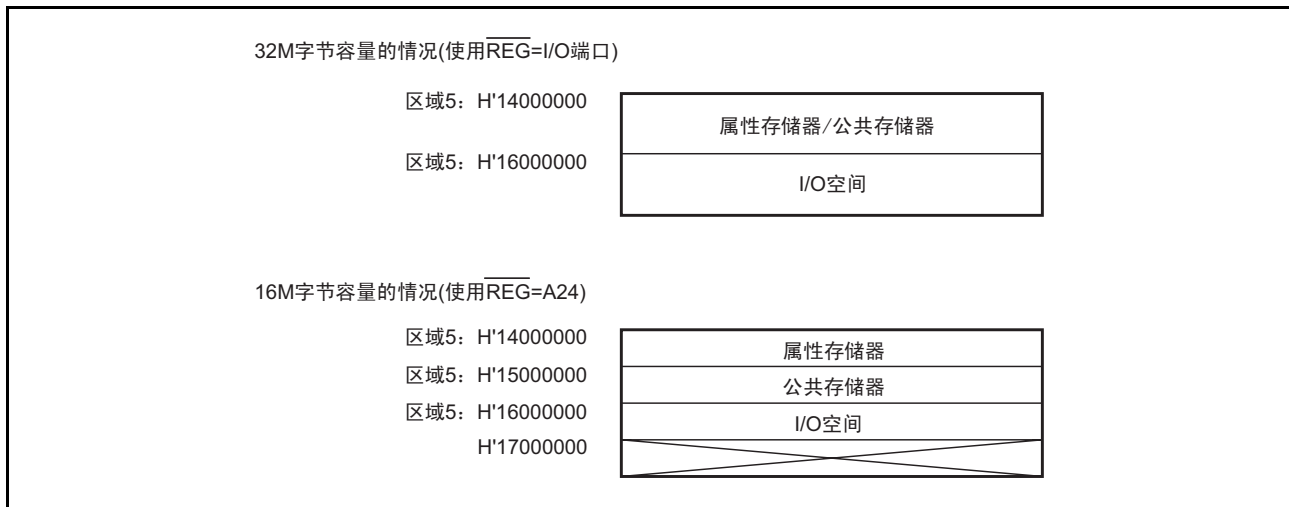


图 10.42 PCMCIA 空间的分配例子 (CS5WCR.SA[1:0]=B'10)

(2) I/O 卡接口的基本时序

PCMCIA 的 I/O 卡接口时序如图 10.43 和图 10.44 所示。

要存取 PCMCIA 的 I/O 卡时，必须存取高速缓存的无效区域。

根据 CS5WCR 的 SA[1:0] 位的设定，通过要存取的地址进行 I/O 卡接口和 IC 存储器卡接口的切换。

要在小端法模式中将 PCMCIA 卡作为 I/O 卡接口存取时，使用 $\overline{\text{IOIS16}}$ 引脚调整 I/O 总线宽度的动态总线。如果将区域 5 的总线宽度设定为 16 位，就在字长度的 I/O 卡总线周期中 $\overline{\text{IOIS16}}$ 信号为高电平时，总线宽度被视为 8 位，并且在执行中的 I/O 卡总线周期中只进行 8 位数据的存取，自动地继续进行剩余的 8 位数据的存取。

在将 TED[3:0] 位至少设定为 1.5 个周期时的 Tpci0 和全部 Tpci0w 周期中，在 CKIO 的下降沿，对 $\overline{\text{IOIS16}}$ 信号进行采样，并且在从 Tpci0 的采样点经过 CKIO 的 1.5 个周期后反映到 $\overline{\text{CE2A}}$ 信号。必须在全部的采样点确定 $\overline{\text{IOIS16}}$ 信号，并且不能在中途发生变化。

必须设定 TED[3:0] 位，以便满足所用 PC 卡的 $\overline{\text{ICIORD}}$ 和 $\overline{\text{ICIOWR}}$ 到 $\overline{\text{CE1}}$ 的准备时间的规格。

动态总线调整的基本波形如图 10.44 所示。

在大端法模式中，不支持 $\overline{\text{IOIS16}}$ 信号。此时，必须将 $\overline{\text{IOIS16}}$ 信号固定为低电平。

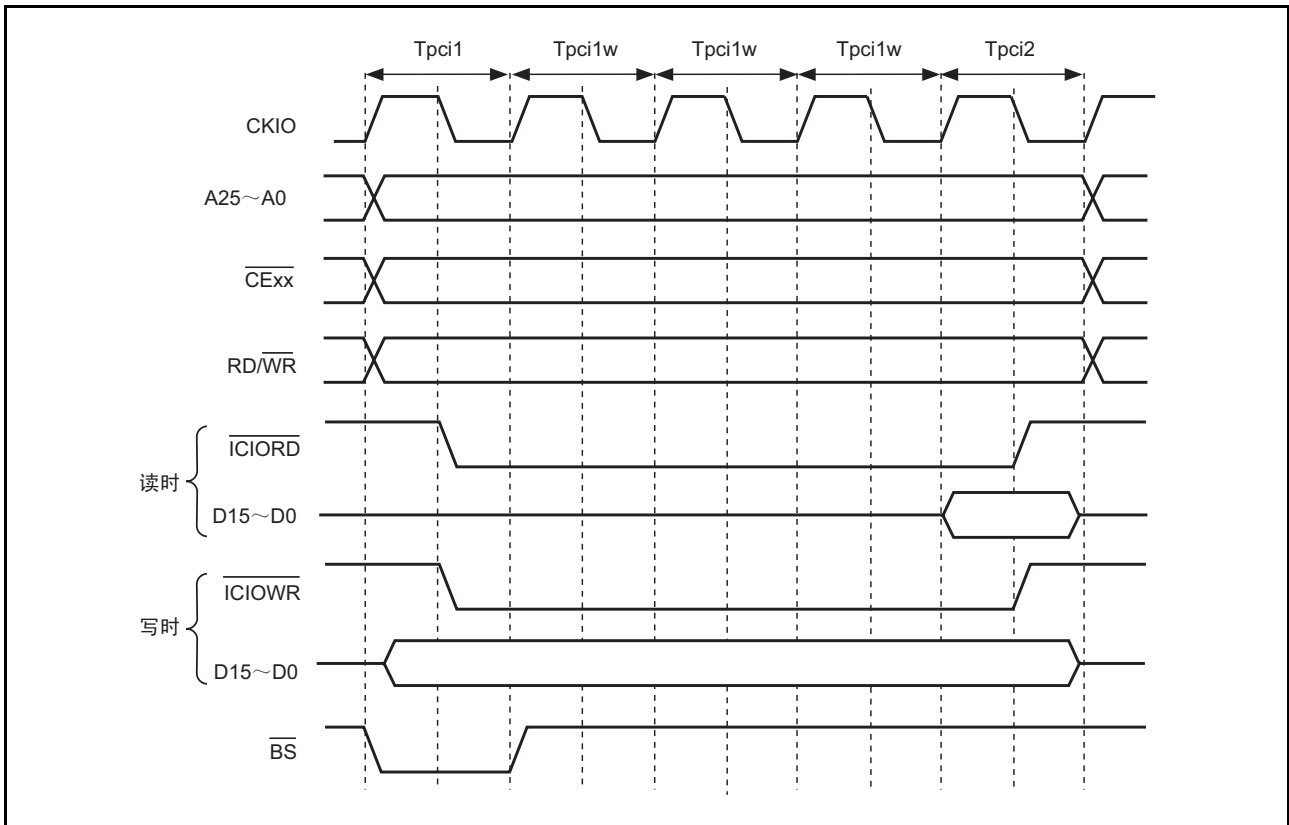


图 10.43 PCMCIA I/O 卡接口的基本时序

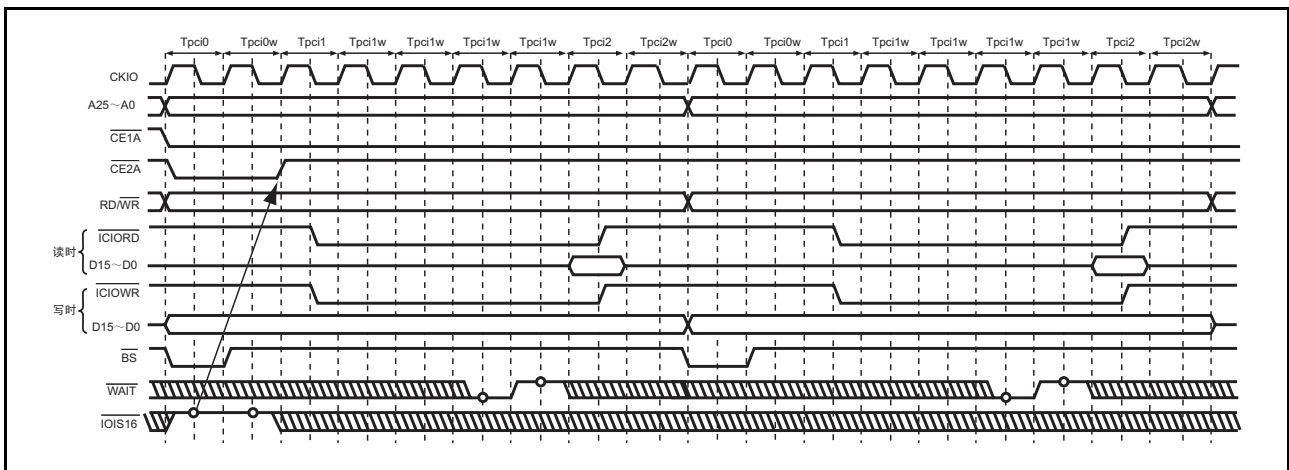


图 10.44 PCMCIA I/O 卡接口的动态总线调整时序
(TED[3:0]=B'0010, PCW[3:0]=B'0000, TEH[3:0]=B'0001, 硬件等待 1)

10.5.10 突发 ROM（时钟同步）接口

突发 ROM（时钟同步）接口用于高速存取具有同步突发功能的 ROM。基本上进行和正规空间相同的存取，此接口只能设定在区域 0。

在最初的存取周期中插入 CS0WCR 的 W[3:0] 位所设个数的等待周期，在第 2 次以后的存取周期中插入 CS0WCR 的 BW[1:0] 位所设个数的等待周期。

当存取突发 ROM（时钟同步）时， \overline{BS} 信号只对最初的存取周期有效，并且外部等待输入也只对最初的存取周期有效。

当总线宽度为 16 位时，必须将突发长度设定为 8；当总线宽度为 32 位时，必须将突发长度设定为 4。不支持 8 位总线。

此接口对全部读存取进行突发运行。例如，当以 16 位总线宽度进行长字存取时，在读 2 个需要的数据后，空读剩余的 6 个不需要的数据。

因为这样的空读周期会增加存储器的存取时间，降低程序的执行速度和 DMA 的传送速度，所以必须有效地利用由高速缓存有效空间中的高速缓存填充进行 16 字节读操作或者由 DMA 进行的 16 字节读操作。在写存取时，和正规空间的存取时序相同。

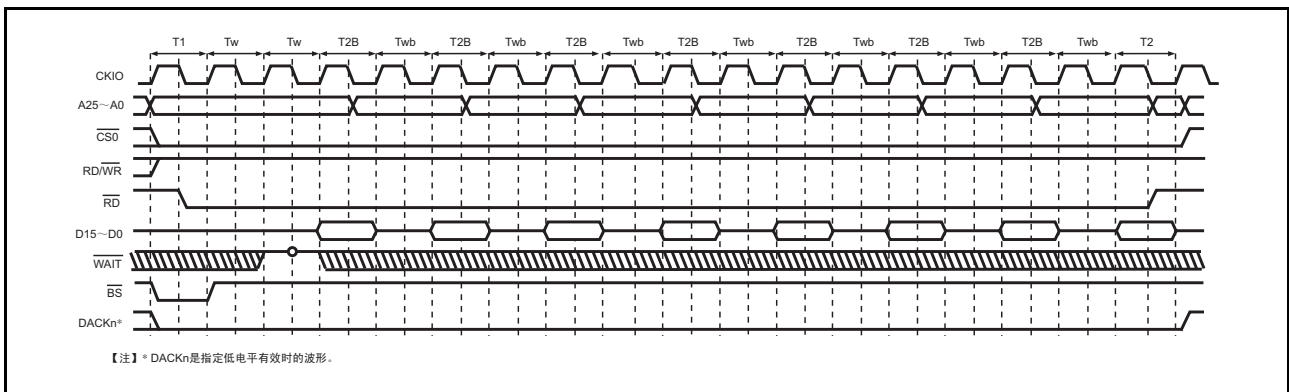


图 10.45 突发 ROM（时钟同步）的存取（突发长度 8、初次存取等待 2、第 2 次以后的存取等待 1）

10.5.11 存取周期之间的空闲

由于 LSI 工作频率的提高，可能在低速设备的数据输出结束时来不及关闭数据缓冲器，导致和下一个设备的数据输出发生冲突，从而降低设备的可靠性或者引起误动作。为了防止这种情况，在连续的存取周期之间插入空闲（等待），以避免数据的冲突。

通过 CSnWCR 的 WM 位以及 CSnBCR 的 IWW[2:0] 位、IWRWD[2:0] 位、IWRWS[2:0] 位、IWRRD[2:0] 位、IWRRS[2:0] 位和 CMNCR 的 DMAIW[2:0] 位、DMAIWA 位指定存取周期之间的空闲周期数。能在存取周期之间插入空闲的条件如下：

1. 连续存取为写-读或者写-写
2. 连续存取为不同空间的读-写
3. 连续存取为同一空间的读-写
4. 连续存取为不同空间的读-读
5. 连续存取为同一空间的读-读
6. 在DMA单地址传送中的外部设备的数据输出周期后从其他设备（包括本LSI）输出数据时（DMAIWA=0）
7. 在DMA单地址传送中的外部设备的数据输出周期后发生存取时（DMAIWA=1）

有关在上述存取周期之间空闲周期数的指定，请参照各寄存器的说明。

除了由这些寄存器指定的存取周期之间的等待空闲周期以外，为了确保连接内部总线的接口或者多路复用引脚（WEn）的最小脉冲宽度，可能需要插入空闲周期。以下说明的是空闲周期的详细内容和空闲周期数的计算方法。

对从 $\overline{\text{CSn}}$ 无效到 $\overline{\text{CSn}}$ 或者 $\overline{\text{CSm}}$ 有效的外部总线空闲周期数进行说明。在此， $\overline{\text{CSn}}$ 和 $\overline{\text{CSm}}$ 中含有 PCMCIA 的 CE2A。

决定外部总线空闲周期数的项目有 8 项，如表 10.21 所示。它们之间的关系如图 10.46 所示。

表 10.21 决定空闲周期数的项目

No	内容	说明	范围	注意事项
(1)	CMNCR.DMAIW[2:0] 的设定	指定 DMA 单地址传送时的空闲周期数。此项目只在单地址传送时有效，是在存取结束后产生的空闲周期。	0 ~ 12	必须注意：如果将空闲数设定为“0”，DACK 信号就可能连续有效，发生带 DACK 的设备的识别周期数和 DMAC 传送数不相同的情况，导致误动作。
(2)	CSnBCR.IW***[2:0] 的设定	指定非单地址传送时的空闲周期数。能按前后周期的各种组合指定空闲周期数。例如，在读 CS1 空间后读其他 CS 空间时，如果要将空闲数至少设定为 6 个周期，就将 CS1BCR.IWRRD[2:0] 设定为“B'100”。此项目只在非单地址传送时有效，是在存取结束后产生的空闲周期。	0 ~ 12	必须注意：如果是不能连续存取的存储器，就不能设定为“0”。
(3)	CSnWCR 的 SDRAM 相关设定	指定存取 SDRAM 时的预充电结束 / 启动等待和命令之间的空闲数。此项目只在存取 SDRAM 时有效，是在存取结束后产生的空闲周期。	0 ~ 3	必须根据所使用的 SDRAM 的规格设定。
(4)	CSnWCR.WM 位的设定	在 SDRAM 以外的存储器中，能将外部 $\overline{\text{WAIT}}$ 引脚输入设定为有效或者无效。当此位为“0”（外部 $\overline{\text{WAIT}}$ 有效）时，为了评价外部 $\overline{\text{WAIT}}$ 引脚状态，在存取结束后插入 1 个空闲周期；当此位为“1”（无效）时，不发生此空闲周期。	0 ~ 1	

No	内容	说明	范围	注意事项
(5)	读数据的传送周期	这是在读存取结束后产生的 1 个空闲周期。在分割存取的最初和中途的存取中不产生空闲周期，并且在 CSnWCR.HW[1:0] 不为“B'00”时也不产生空闲周期。	0 ~ 1	在 SDRAM 或者 PCMCIA 的读周期中肯定会产生 1 个空闲周期。
(6)	内部总线空闲等	通过内部总线，传递来自 CPU、直接存储器存取控制器等的外部总线存取请求和结果。在内部总线的空闲周期以及外部总线以外的存取中，外部总线为空闲状态。在存取长度超过外部数据总线宽度时，通过此模块进行分割存取，但是在分割周期之间，没有内部总线空闲周期等的影响。	0 ~	根据 Iφ:Bφ:CKIOφ 的时钟比，内部总线空闲数可能不为“0”。时钟比和内部总线的最小空闲数的关系如表 10.22 和表 10.23 所示。
(7)	写数据到达的等待周期	当写存取时，在等待写数据到达后产生外部总线的写周期。此写数据的等待为写周期前产生的空闲周期。但是，在前一个周期为写周期并且内部总线空闲短于前一个存取的写周期时，为了与前一个存取并行处理，不产生此空闲周期（写缓冲器效果）。	0 ~ 1	在写 → 写以及写 → 读存取时，通过左边所述的写缓冲器效果，容易进行连续存取。在不能连续存取时，必须通过 CSnBCR 等指定周期之间的最小空闲数。
(8)	不同存储器之间的空闲	为了确保多路引脚的最小脉冲宽度，在切换存储器类型后的存取前，可能会插入空闲周期。根据存储器类型，即使不切换存储器类型，也会产生空闲周期。	0 ~ 2	取决于各存储器类型。请参照表 10.24。

由于项 (1)/(2)（其中一项有效）、项 (3)/(4)（其中一项有效）、项 (5)+(6)+(7)（因为按顺序发生，所以被累计）和项 (8) 共 4 项平行发生，因此其中最大的一项为外部总线空闲数。要确保最小空闲数时，必须进行项 (1)/(2) 寄存器的设定。

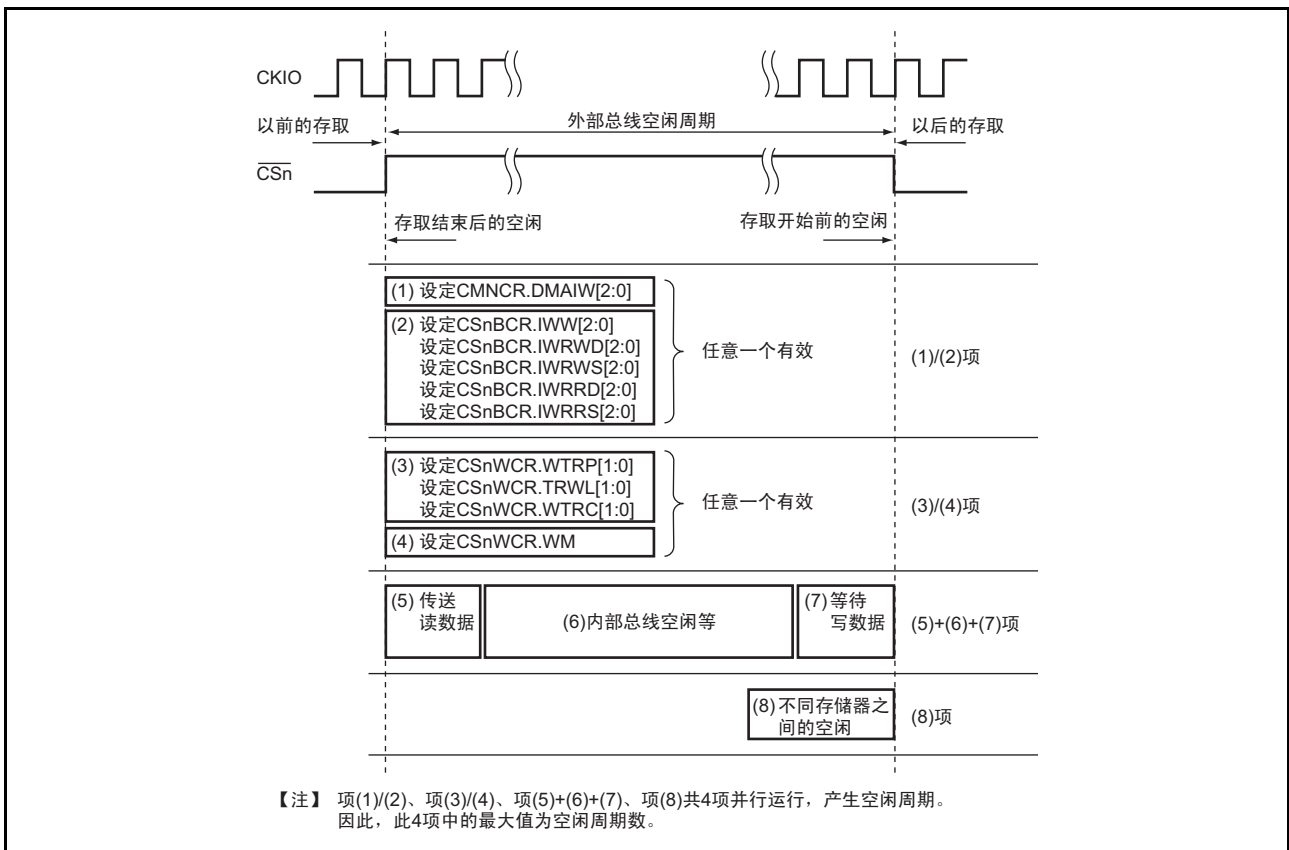


图 10.46 空闲周期的结构

表 10.22 内部总线的最小空闲数 (CPU 运行)

CPU 运行	时钟比 (I _φ :B _φ :CKIO _φ)				
	4:2:1	4:1:1	2:2:1	2:1:1	1:1:1
写 → 写	1	2	2	2	3
写 → 读	0	0	0	0	0
读 → 写	1	2	2	2	3
读 → 读	0	0	1	0	1

表 10.23 内部总线的最小空闲数 (直接存储器存取控制器)

传送模式	读写操作	时钟比 (B _φ :CKIO _φ)	
		2:1	1:1
双地址	写 → 写 *1	0	0
	写 → 读	0	0
	读 → 写	0	0
	读 → 读 *1	0	0
单地址 (DREQ 为电平检测) *2	写 → 写	3	6
	读 → 读	2	5
单地址 (DREQ 为边沿检测) *2	写 → 写	0	1
	读 → 读	1	2

【注】 *1. 双地址的写 → 写、读 → 读的操作在被分割的周期中进行。

*2. 单地址的“写”是指带 DACK 的设备 → 外部存储器的传送，“读”是指外部存储器 → 带 DACK 的设备的传送。

表 10.24 在不同存储器之间的存取之前插入的空闲周期数

		后一个周期							
		SRAM	突发 ROM (异步)	MPX-I/O	字节 SRAM (BAS=0)	字节 SRAM (BAS=1)	SDRAM	PCMCIA	突发 ROM (同步)
前 一 个 周 期	SRAM	0	0	1	0	1	1	0	0
	突发 ROM (异步)	0	0	1	0	1	1	0	0
	MPX-I/O	1	1	0	1	1	1	1	1
	字节 SRAM (BAS=0)	0	0	1	0	1	1	0	0
	字节 SRAM (BAS=1)	1	1	2	1	0	0	1	1
	SDRAM	1	1	2	1	0	0	1	1
	PCMCIA	0	0	1	0	1	1	0	0
	突发 ROM (同步)	0	0	1	0	1	1	0	0

周期之间最小空闲数的估算例子如图 10.47 所示。在实际运行中，根据写缓冲器的效果，空闲周期可能小于估算值，而且可能因 CPU 指令的执行或者 CPU 寄存器的竞争而引起的分割使内部总线产生空闲周期，导致空闲周期大于估算值。因此，在使用估算值时，必须考虑这些误差。

周期之间空闲数的估算例子

考虑通过CPU的存取将数据从CS1空间传送到CS2空间的例子。假设重复进行CS1读→CS1读→CS2写→CS2写→CS1读→…的传送。

• 条件
 将CS1BCR和CS2BCR的周期之间的空闲全部指定为“0”。
 CS1WCR和CS2WCR的WM位为“1”(外部WAIT引脚无效)，HW[1:0]位为“00”(不延长CS的无效)。
 假设 $t_{\phi}:\text{B}\phi:\text{CKIO}\phi$ 为4:1:1，在传送时不进行其他处理。
 CS1和CS2连接正规SRAM，总线宽度为32位，存取长度也为32位。

在每个周期之间估算决定空闲数的项目。下表中的R表示读，W表示写。

项目	R→R	R→W	W→W	W→R	备注
(1)/(2)	0	0	0	0	因为将CSnBCR设定为“0”。
(3)/(4)	0	0	0	0	因为WM位为“1”。
(5)	1	1	0	0	在读周期后发生。
(6)	0	2	2	0	请参照表10.22的 $t_{\phi}:\text{B}\phi:\text{CKIO}\phi=4:1:1$ 的部分。
(7)	0	1	0	0	因为写缓冲器的效果，不产生第2次的空闲周期。
(5)+(6)+(7)	1	4	2	0	
(8)	0	0	0	0	因为SRAM→SRAM。
估算空闲周期数	1	4	2	0	项(1)/(2)、项(3)/(4)、项(5)+(6)+(7)、项(8)中的最大值
实际产生的空闲数	1	4	2	1	在W→R时出现不相同的原因是将(6)的内部空闲数估算为“0”，而实际上执行了循环判断指令，产生了内部空闲。

图 10.47 空闲周期数的估算例子和实际的比较

10.5.12 总线仲裁

本 LSI 中的总线仲裁在正常状态下占有总线权，在接受其它设备的总线权请求后释放总线。

在总线周期的边界传递总线权。在有总线权请求时，如果没有在执行总线周期，就立即释放总线权。如果正在执行总线周期，就在等到总线周期结束后释放总线权。从 LSI 外部看，即使没有在执行总线周期，也可能因在存取周期之间插入等待而在内部开始执行总线周期，所以不能从 $\overline{\text{CSn}}$ 信号和其他总线控制信号来直接判断总线是否被释放。不进行总线释放的状态如下所示：

1. 当因高速缓存未命中而正在进行 16 字节传送时
2. 当正在回写高速缓存时
3. 在 TAS 指令的读周期和写周期之间
4. 因数据总线宽度小于存取长度而产生的多个总线周期（例如：在对 8 位数据总线宽度的存储器进行长字存取时的总线周期之间）
5. 当通过直接存储器存取控制器进行传送 16 字节时
6. 当通过 OpenVG 瑞萨图形处理器、失真校正引擎和视频显示控制器 4 进行 32 字节传送时
7. 当通过视频显示控制器 4 进行 128 字节传送时
8. 当将 CMNCR 的 BLOCK 位置“1”时
9. 当通过 NAND 闪存控制器正在进行外部闪存的存取时

通过 CMNCR 的 DPRTY[1:0] 位，能在直接存储器存取控制器的突发传送过程中选择是否接受总线权使用请求。

本 LSI 只要不接受总线权请求就占有总线权。接受外部的总线权请求 $\overline{\text{BREQ}}$ 的有效电平（低电平），在当前执行中的总线周期结束后释放总线，并且将总线使用允许 $\overline{\text{BACK}}$ 置为有效（低电平）。在接受表示外部设备已释放总线的 $\overline{\text{BREQ}}$ 无效电平（高电平）后，将 $\overline{\text{BACK}}$ 置为无效（高电平），重新开始使用总线。

在使用 SDRAM 接口时，如果有激活的存储体，就发行全部存储体预充电命令（PALL），并且在命令结束后释放总线。

具体的总线释放顺序如下所示。首先，与 CKIO 的上升沿同步，将地址总线 and 数据总线置为高阻抗。在 0.5 个周期后与 CKIO 的下降沿同步，将总线使用允许信号置为有效。接着，在 CKIO 的上升沿，将总线控制信号（ $\overline{\text{BS}}$ 、 $\overline{\text{CSn}}$ 、 $\overline{\text{RAS}}$ 、 $\overline{\text{CAS}}$ 、 $\overline{\text{CKE}}$ 、 $\overline{\text{DQMxx}}$ 、 $\overline{\text{WEn}}$ 、 $\overline{\text{RD}}$ 和 $\overline{\text{RD/WR}}$ ）置为高阻抗。这些总线控制信号最迟也需要在置为高阻抗的 1 个周期前变为高电平。在 CKIO 的下降沿对总线权请求信号进行采样。而且，通过设定 CMNCR 的 HIZCNT 位，即使在总线释放过程中也能用前值继续驱动 $\overline{\text{CKE}}$ 、 $\overline{\text{RAS}}$ 、 $\overline{\text{CAS}}$ 。

再次从外部设备获得总线权的顺序如下所示。在 CKIO 的下降沿检测到 $\overline{\text{BREQ}}$ 的无效电平，在 1.5 个周期后开始以高电平驱动总线控制信号。接着，在时钟的下降沿，将总线使用允许信号置为无效。在 CKIO 的上升沿开始驱动地址总线 and 数据总线。从将总线控制信号置为有效到实际开始执行总线周期，最快的情况也是在与驱动地址和数据信号的相同时钟的上升沿。总线仲裁时序如图 10.48 所示。

在释放总线权的过程中需要刷新 SDRAM 时，需要返回总线权。在超过刷新间隔的规定时间后还没返回总线权时，因为不能进行刷新，所以无法确保 SDRAM 的内容。

在本 LSI 获取总线权前，不执行释放总线权过程中的 SLEEP 指令（向睡眠模式、深度待机模式或者软件待机模式的转移）。释放总线权过程中的手动复位也一样。

在软件待机模式或者深度待机模式中，忽视 $\overline{\text{BREQ}}$ 的输入并且 $\overline{\text{BACK}}$ 输出为高阻抗状态。在此状态下需要请求总线权时，通过将 $\overline{\text{BACK}}$ 引脚下拉，在转移到软件待机模式或者深度待机模式的同时，变为总线权释放状态。

如果发生总线权请求（ $\overline{\text{BREQ}}$ 为低电平有效），就必须在允许总线使用（ $\overline{\text{BACK}}$ 为低电平有效）后释放总线权（ $\overline{\text{BREQ}}$ 为高电平无效）。如果在 $\overline{\text{BACK}}$ 有效前将 $\overline{\text{BREQ}}$ 置为无效，就根据 $\overline{\text{BREQ}}$ 无效时序， $\overline{\text{BACK}}$ 只有 1 个周期有效，可能在外部设备和本 LSI 之间发生总线冲突。

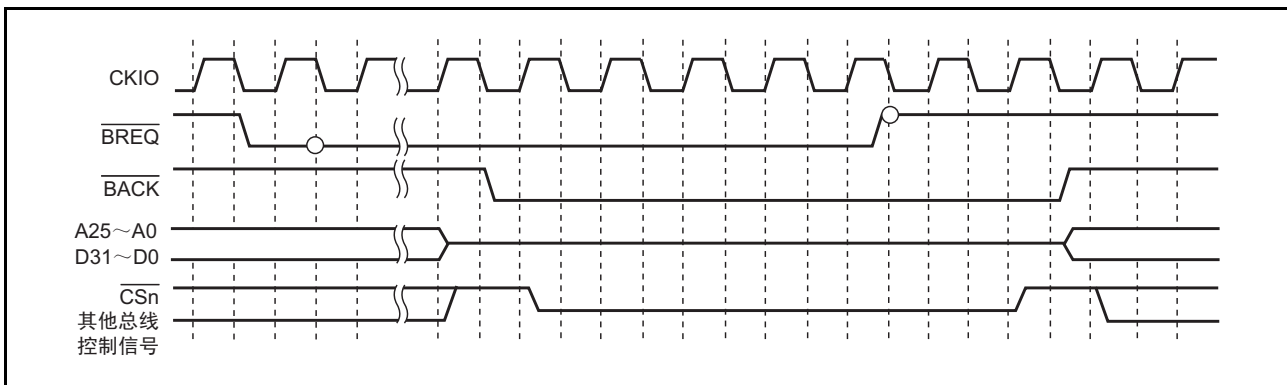


图 10.48 总线仲裁

10.5.13 其他

(1) 复位

此模块只在上电复位时才被完全初始化。在上电复位时进行内部时钟同步，然后将全部信号置为无效，关闭数据输出缓冲器，并且对全部控制寄存器进行初始化，与是否在执行总线周期无关。在软件待机模式、睡眠和手动复位中，不对总线状态控制器的控制寄存器进行初始化。如果进行手动复位，就让当前正在执行的总线周期执行结束。在手动复位信号有效时，RTCNT 也能递增计数，因此能产生刷新请求，启动刷新周期。

(2) LSI 内部总线主控的存取

本 LSI 内部分为 CPU 总线、内部总线和外围总线共 3 种总线。将 CPU 和高速缓存存储器连接到 CPU 总线，将 CPU 以外的内部总线主控器和此模块连接到内部总线，并且将低速外围模块连接到外围总线。另外，高速缓存存储器以外的内部存储器连接到 CPU 总线和内部总线。能从 CPU 总线存取内部总线，但是不能从内部总线存取 CPU 总线。为此，发生以下的问题。

能从除 CPU 以外的内部总线主控（直接存储器存取控制器等）存取高速缓存存储器以外的内部存储器，但是不能存取高速缓存存储器。通过除 CPU 以外的内部总线主控写外部存储器，但是结果有可能发生外部存储器的内容和高速缓存的内容不符的情况。如果通过除 CPU 以外的内部总线主控写外部存储器，就在传送到该地址的数据可能存在于高速缓存时需要通过软件清除高速缓存存储器。

如果 CPU 在高速缓存有效空间的情况下开始读存取，就搜索高速缓存。当数据保留在高速缓存中时，就取数据并且结束存取；当高速缓存中没有数据时，就通过内部总线进行高速缓存数据的填充，因此启动 4 个连续的长字读操作。对于在存取字节或者字操作数时以及转移到奇数字边界（ $4n+2$ ）时的命中和未命中，总是通过芯片外部接口，用 4 个长字存取进行填充。在高速缓存无效空间的情况下，根据实际的存取地址进行存取。在对偶数字边界（ $4n$ ）取指令时为长字存取，在对奇数字边界（ $4n+2$ ）取指令时为字存取。

在内部外围模块读周期的情况下，通过内部总线和外围总线启动读周期。读数据经过外围总线、内部总线、CPU 总线传送到 CPU。

高速缓存有效空间的写周期运行因高速缓存的写方式而不同。

在回写模式中，搜索高速缓存。如果有相应地址的数据，就写高速缓存。在替换相应地址前不写实际的存储器。如果没有相应地址的数据，就更新高速缓存。首先，将替换对象的数据保存到内部缓冲器，然后读包含相应地址数据的 16 字节数据，更新相应地址的数据。接着，执行最初保存的 16 字节数据的回写周期。

在直写模式中，搜索高速缓存。如果有相应地址的数据，就在写高速缓存的同时，通过内部总线进行实际的写操作。如果没有相应地址的数据，就不更新高速缓存而只通过内部总线进行实际的写操作。

由于此模块有一段写缓冲器，因此即使在写周期中芯片外部的总线周期还没有结束，也能将内部总线用于其他存取。如果在写芯片外部的低速存储器后读写内部外围模块，就能不等待低速存储器写操作的结束而存取内部外围模块。

在读操作时，CPU 总是等待读操作结束，所以想在确认数据已写到实际的设备后继续处理时，只要继续虚读相同地址，就能确认写操作的结束。

对于从直接存储器存取控制器等其他总线主控的存取，此模块的写缓冲器也同样起作用。因此，在进行双地址的 DMA 传送时，不等待写周期结束就启动下一个读周期。但是，当 DMA 的源地址和目标地址都在外部存储器空间时，就等到前一个写周期结束后才开始下一个读周期。

如果在写缓冲器运行过程中更改此模块的寄存器，就不能正确地进行写存取，因此不能在写存取后立即更改此模块的寄存器。如果需要，就必须在虚读写数据后更改此模块的寄存器。

(3) 内部外围模块的存取

在存取内部外围模块的寄存器时，内部总线至少需要 2 个周期的外围模块时钟 ($P0\phi$ 或者 $P1\phi$)。另一方面，在 CPU 写内部外围模块时，CPU 不等待寄存器写操作的结束而执行后续的指令。

以为了降低功耗而转移到软件待机模式为例进行说明。为实现转移，需要在将 STBCR 寄存器的 STBY 位置“1”后执行 SLEEP 指令，但是必须在执行 SLEEP 指令前虚读 STBCR 寄存器。如果不虚读，CPU 就在将 STBY 位置“1”前执行 SLEEP 指令，转移到睡眠模式而不转移到软件待机模式。为了等待 STBY 位的写操作，需要虚读 STBCR 寄存器。

如此例所示，如果想在执行后续指令时反映内部外围寄存器所发生的变化，必须在执行寄存器的写指令后虚读相同的寄存器，然后再执行后续指令。

(4) 由 NAND 闪存控制器进行的外部闪存的存取

在本产品中，部分外部数据总线和 NAND 闪存控制器的数据总线兼用。通过 NAND 闪存控制器控制数据总线的使用。在结束本模块的外部设备存取后开始 NAND 闪存控制器的存取。在 NAND 闪存控制器的存取过程中，此模块外部设备总线的存取等待 NAND 闪存控制器存取的结束。

当 NAND 闪存控制器的存取请求和外部总线的解除请求发生竞争时，优先先接受的存取请求。在同时请求的情况下，优先 NAND 闪存控制器的存取。

即使在 NAND 闪存控制器的存取过程中也执行 SDRAM 的自动刷新和自刷新。

11. 直接存储器存取控制器

直接存储器存取控制器能代替 CPU 在带 DACK（传送请求的接收信号）的外部设备、外部存储器、内部存储器、存储器映射外部设备以及内部外围模块之间进行高速数据传送。

11.1 特点

- 通道数：有 CH0～CH15 共 16 个通道。其中，CH0 能接受外部请求。
- 地址空间：在体系结构上为 4G 字节。
- 传送数据单位：字节、字（2 字节）、长字（4 字节）、16 字节（长字×4）
- 最大传送次数：16777216（24 位）次
- 地址模式：可选择单地址模式或者双地址模式
- 传送请求：可选择外部请求、内部外围模块请求、自动请求共 3 种。
能发行内部外围模块请求的模块如下：
 - FIFO 内置型串行通信接口：16 个源
 - I²C 总线接口 3：8 个源
 - A/D 转换器：1 个源
 - 多功能定时器脉冲单元 2：5 个源
 - 比较匹配定时器：2 个源
 - USB2.0 主机/功能模块：2 个源
 - NAND 闪存控制器：2 个源
 - 控制器局域网：3 个源
 - 串行音频接口：7 个源
 - 采样率转换器：6 个源
 - 音频发生器：4 个源
 - 瑞萨 SPDIF 接口：2 个源
 - CD-ROM 解码器：1 个源
 - SD 主机接口：4 个源
 - MMC 主机接口：2 个源
 - 瑞萨串行外围接口：4 个源
 - 瑞萨四路串行外围接口：4 个源
 - 带 FIFO 的时钟同步串行 I/O：2 个源
 - 马达控制 PWM 定时器：2 个源
- 总线模式：可选择周期挪用模式（普通模式和间歇模式）以及突发模式。
- 优先级：可选择 2 种通道优先级的固定模式。
- 中断请求：能在结束 1/2 的数据传送或者结束数据传送时向 CPU 请求中断。
通过 CHCR 的 HE 位和 HIE 位，在 DMA 传送初始设定的 1/2 次结束时对 CPU 设定中断。
- 外部请求的检测：可选择 DREQ 输入的低/高电平检测或者上升/下降沿检测。
- 传送请求的接受信号和传送结束信号：DACK 和 TEND 能设定有效电平。
- 由于 DMA 设定的寄存器有重加载功能，因此能通过正在执行中的 DMA 传送的相同设定，反复执行 DMA 传送而不需要重新设定。在 DMA 传送过程中，通过预先设定重加载寄存器，能用不同的设定执行下一轮的 DMA 传送。
能对各通道的各重加载寄存器设定此重加载功能的 ON/OFF。

框图如图 11.1 所示。

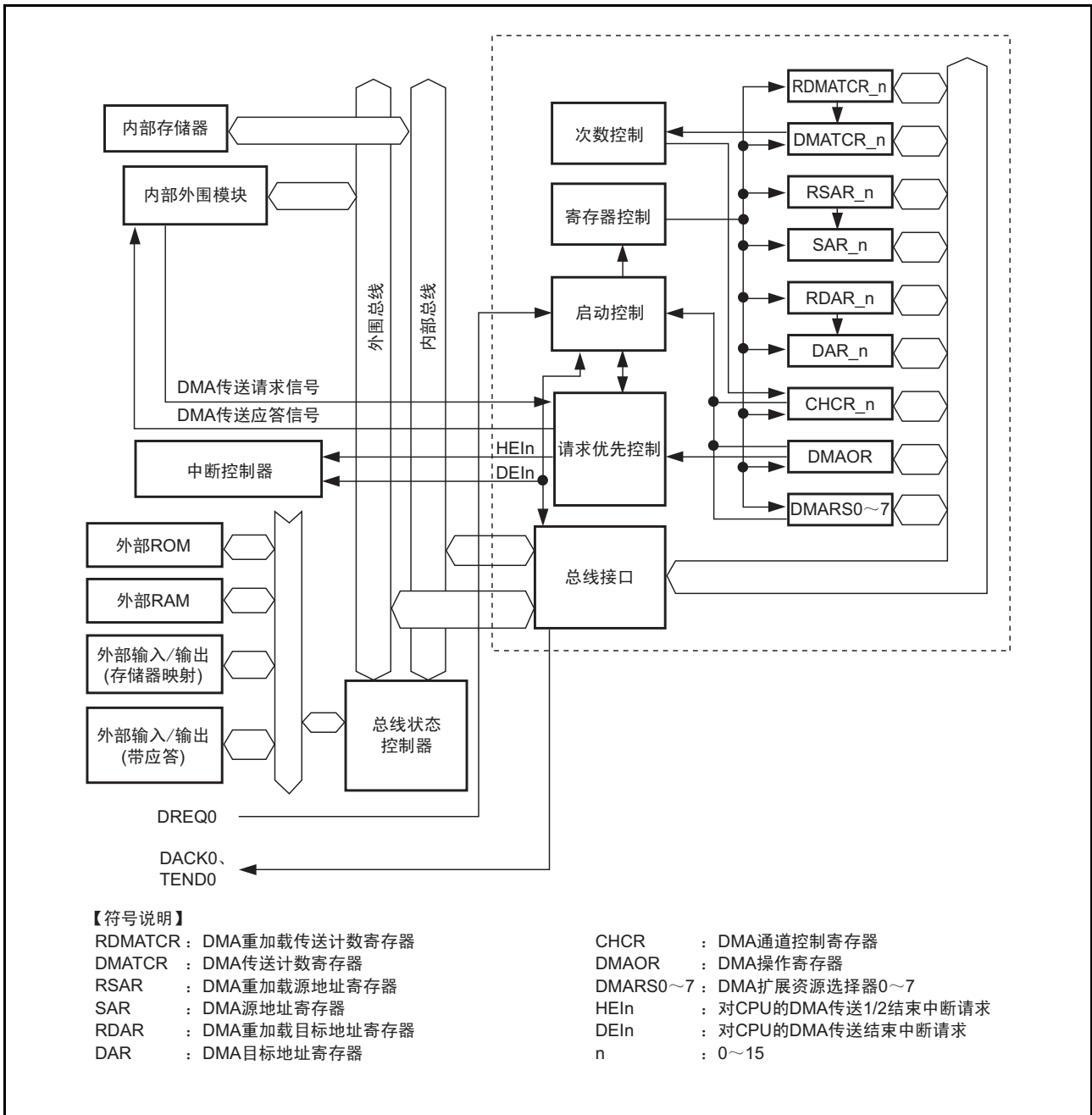


图 11.1 框图

11.2 输入 / 输出引脚

引脚结构如表 11.1 所示。此模块有 1 个通道的引脚（CH0）用于外部总线。

表 11.1 引脚结构

通道	名称	引脚名	输入 / 输出	功能
0	DMA 传送请求	DREQ0	输入	将 DMA 传送请求从外部设备输入到通道 0。
	DMA 传送请求的接受	DACK0	输出	将 DMA 传送请求的接受从此模块通道 0 输出到外部设备。
	DMA 传送结束	TEND0	输出	输出此模块通道 0 的 DMA 传送结束。

11.3 寄存器说明

寄存器结构如表 11.2 所示。各通道有 4 个控制寄存器和 3 个重加载寄存器，全部通道有 1 个共用的控制寄存器，并且每 2 通道有 1 个扩展资源选择器的寄存器。各通道的寄存器的记载例子：通道 0 的 SAR 记为 SAR_0。

表 11.2 寄存器结构

通道	寄存器名	略称	R/W	初始值	地址	存取长度
0	DMA 源地址寄存器 _0	SAR_0	R/W	H'00000000	H'FFFE1000	16、32
	DMA 目标地址寄存器 _0	DAR_0	R/W	H'00000000	H'FFFE1004	16、32
	DMA 传送计数寄存器 _0	DMATCR_0	R/W	H'00000000	H'FFFE1008	16、32
	DMA 通道控制寄存器 _0	CHCR_0	R/W*1	H'00000000	H'FFFE100C	8、16、32
	DMA 重加载源地址寄存器 _0	RSAR_0	R/W	H'00000000	H'FFFE1100	16、32
	DMA 重加载目标地址寄存器 _0	RDAR_0	R/W	H'00000000	H'FFFE1104	16、32
	DMA 重加载传送计数寄存器 _0	RDMATCR_0	R/W	H'00000000	H'FFFE1108	16、32
1	DMA 源地址寄存器 _1	SAR_1	R/W	H'00000000	H'FFFE1010	16、32
	DMA 目标地址寄存器 _1	DAR_1	R/W	H'00000000	H'FFFE1014	16、32
	DMA 传送计数寄存器 _1	DMATCR_1	R/W	H'00000000	H'FFFE1018	16、32
	DMA 通道控制寄存器 _1	CHCR_1	R/W*1	H'00000000	H'FFFE101C	8、16、32
	DMA 重加载源地址寄存器 _1	RSAR_1	R/W	H'00000000	H'FFFE1110	16、32
	DMA 重加载目标地址寄存器 _1	RDAR_1	R/W	H'00000000	H'FFFE1114	16、32
	DMA 重加载传送计数寄存器 _1	RDMATCR_1	R/W	H'00000000	H'FFFE1118	16、32
2	DMA 源地址寄存器 _2	SAR_2	R/W	H'00000000	H'FFFE1020	16、32
	DMA 目标地址寄存器 _2	DAR_2	R/W	H'00000000	H'FFFE1024	16、32
	DMA 传送计数寄存器 _2	DMATCR_2	R/W	H'00000000	H'FFFE1028	16、32
	DMA 通道控制寄存器 _2	CHCR_2	R/W*1	H'00000000	H'FFFE102C	8、16、32
	DMA 重加载源地址寄存器 _2	RSAR_2	R/W	H'00000000	H'FFFE1120	16、32
	DMA 重加载目标地址寄存器 _2	RDAR_2	R/W	H'00000000	H'FFFE1124	16、32
	DMA 重加载传送计数寄存器 _2	RDMATCR_2	R/W	H'00000000	H'FFFE1128	16、32

通道	寄存器名	略称	R/W	初始值	地址	存取长度
3	DMA 源地址寄存器_3	SAR_3	R/W	H'00000000	H'FFFE1030	16、32
	DMA 目标地址寄存器_3	DAR_3	R/W	H'00000000	H'FFFE1034	16、32
	DMA 传送计数寄存器_3	DMATCR_3	R/W	H'00000000	H'FFFE1038	16、32
	DMA 通道控制寄存器_3	CHCR_3	R/W*1	H'00000000	H'FFFE103C	8、16、32
	DMA 重加载源地址寄存器_3	RSAR_3	R/W	H'00000000	H'FFFE1130	16、32
	DMA 重加载目标地址寄存器_3	RDAR_3	R/W	H'00000000	H'FFFE1134	16、32
	DMA 重加载传送计数寄存器_3	RDMATCR_3	R/W	H'00000000	H'FFFE1138	16、32
4	DMA 源地址寄存器_4	SAR_4	R/W	H'00000000	H'FFFE1040	16、32
	DMA 目标地址寄存器_4	DAR_4	R/W	H'00000000	H'FFFE1044	16、32
	DMA 传送计数寄存器_4	DMATCR_4	R/W	H'00000000	H'FFFE1048	16、32
	DMA 通道控制寄存器_4	CHCR_4	R/W*1	H'00000000	H'FFFE104C	8、16、32
	DMA 重加载源地址寄存器_4	RSAR_4	R/W	H'00000000	H'FFFE1140	16、32
	DMA 重加载目标地址寄存器_4	RDAR_4	R/W	H'00000000	H'FFFE1144	16、32
	DMA 重加载传送计数寄存器_4	RDMATCR_4	R/W	H'00000000	H'FFFE1148	16、32
5	DMA 源地址寄存器_5	SAR_5	R/W	H'00000000	H'FFFE1050	16、32
	DMA 目标地址寄存器_5	DAR_5	R/W	H'00000000	H'FFFE1054	16、32
	DMA 传送计数寄存器_5	DMATCR_5	R/W	H'00000000	H'FFFE1058	16、32
	DMA 通道控制寄存器_5	CHCR_5	R/W*1	H'00000000	H'FFFE105C	8、16、32
	DMA 重加载源地址寄存器_5	RSAR_5	R/W	H'00000000	H'FFFE1150	16、32
	DMA 重加载目标地址寄存器_5	RDAR_5	R/W	H'00000000	H'FFFE1154	16、32
	DMA 重加载传送计数寄存器_5	RDMATCR_5	R/W	H'00000000	H'FFFE1158	16、32
6	DMA 源地址寄存器_6	SAR_6	R/W	H'00000000	H'FFFE1060	16、32
	DMA 目标地址寄存器_6	DAR_6	R/W	H'00000000	H'FFFE1064	16、32
	DMA 传送计数寄存器_6	DMATCR_6	R/W	H'00000000	H'FFFE1068	16、32
	DMA 通道控制寄存器_6	CHCR_6	R/W*1	H'00000000	H'FFFE106C	8、16、32
	DMA 重加载源地址寄存器_6	RSAR_6	R/W	H'00000000	H'FFFE1160	16、32
	DMA 重加载目标地址寄存器_6	RDAR_6	R/W	H'00000000	H'FFFE1164	16、32
	DMA 重加载传送计数寄存器_6	RDMATCR6	R/W	H'00000000	H'FFFE1168	16、32
7	DMA 源地址寄存器_7	SAR_7	R/W	H'00000000	H'FFFE1070	16、32
	DMA 目标地址寄存器_7	DAR_7	R/W	H'00000000	H'FFFE1074	16、32
	DMA 传送计数寄存器_7	DMATCR_7	R/W	H'00000000	H'FFFE1078	16、32
	DMA 通道控制寄存器_7	CHCR_7	R/W*1	H'00000000	H'FFFE107C	8、16、32
	DMA 重加载源地址寄存器_7	RSAR_7	R/W	H'00000000	H'FFFE1170	16、32
	DMA 重加载目标地址寄存器_7	RDAR_7	R/W	H'00000000	H'FFFE1174	16、32
	DMA 重加载传送计数寄存器_7	RDMATCR_7	R/W	H'00000000	H'FFFE1178	16、32
8	DMA 源地址寄存器_8	SAR_8	R/W	H'00000000	H'FFFE1080	16、32
	DMA 目标地址寄存器_8	DAR_8	R/W	H'00000000	H'FFFE1084	16、32
	DMA 传送计数寄存器_8	DMATCR_8	R/W	H'00000000	H'FFFE1088	16、32
	DMA 通道控制寄存器_8	CHCR_8	R/W*1	H'00000000	H'FFFE108C	8、16、32
	DMA 重加载源地址寄存器_8	RSAR_8	R/W	H'00000000	H'FFFE1180	16、32
	DMA 重加载目标地址寄存器_8	RDAR_8	R/W	H'00000000	H'FFFE1184	16、32
	DMA 重加载传送计数寄存器_8	RDMATCR_8	R/W	H'00000000	H'FFFE1188	16、32

通道	寄存器名	略称	R/W	初始值	地址	存取长度
9	DMA 源地址寄存器 _9	SAR_9	R/W	H'00000000	H'FFFE1090	16、32
	DMA 目标地址寄存器 _9	DAR_9	R/W	H'00000000	H'FFFE1094	16、32
	DMA 传送计数寄存器 _9	DMATCR_9	R/W	H'00000000	H'FFFE1098	16、32
	DMA 通道控制寄存器 _9	CHCR_9	R/W*1	H'00000000	H'FFFE109C	8、16、32
	DMA 重加载源地址寄存器 _9	RSAR_9	R/W	H'00000000	H'FFFE1190	16、32
	DMA 重加载目标地址寄存器 _9	RDAR_9	R/W	H'00000000	H'FFFE1194	16、32
	DMA 重加载传送计数寄存器 _9	RDMATCR_9	R/W	H'00000000	H'FFFE1198	16、32
10	DMA 源地址寄存器 _10	SAR_10	R/W	H'00000000	H'FFFE10A0	16、32
	DMA 目标地址寄存器 _10	DAR_10	R/W	H'00000000	H'FFFE10A4	16、32
	DMA 传送计数寄存器 _10	DMATCR_10	R/W	H'00000000	H'FFFE10A8	16、32
	DMA 通道控制寄存器 _10	CHCR_10	R/W*1	H'00000000	H'FFFE10AC	8、16、32
	DMA 重加载源地址寄存器 _10	RSAR_10	R/W	H'00000000	H'FFFE11A0	16、32
	DMA 重加载目标地址寄存器 _10	RDAR_10	R/W	H'00000000	H'FFFE11A4	16、32
	DMA 重加载传送计数寄存器 _10	RDMATCR_10	R/W	H'00000000	H'FFFE11A8	16、32
11	DMA 源地址寄存器 _11	SAR_11	R/W	H'00000000	H'FFFE10B0	16、32
	DMA 目标地址寄存器 _11	DAR_11	R/W	H'00000000	H'FFFE10B4	16、32
	DMA 传送计数寄存器 _11	DMATCR_11	R/W	H'00000000	H'FFFE10B8	16、32
	DMA 通道控制寄存器 _11	CHCR_11	R/W*1	H'00000000	H'FFFE10BC	8、16、32
	DMA 重加载源地址寄存器 _11	RSAR_11	R/W	H'00000000	H'FFFE11B0	16、32
	DMA 重加载目标地址寄存器 _11	RDAR_11	R/W	H'00000000	H'FFFE11B4	16、32
	DMA 重加载传送计数寄存器 _11	RDMATCR_11	R/W	H'00000000	H'FFFE11B8	16、32
12	DMA 源地址寄存器 _12	SAR_12	R/W	H'00000000	H'FFFE10C0	16、32
	DMA 目标地址寄存器 _12	DAR_12	R/W	H'00000000	H'FFFE10C4	16、32
	DMA 传送计数寄存器 _12	DMATCR_12	R/W	H'00000000	H'FFFE10C8	16、32
	DMA 通道控制寄存器 _12	CHCR_12	R/W*1	H'00000000	H'FFFE10CC	8、16、32
	DMA 重加载源地址寄存器 _12	RSAR_12	R/W	H'00000000	H'FFFE11C0	16、32
	DMA 重加载目标地址寄存器 _12	RDAR_12	R/W	H'00000000	H'FFFE11C4	16、32
	DMA 重加载传送计数寄存器 _12	RDMATCR_12	R/W	H'00000000	H'FFFE11C8	16、32
13	DMA 源地址寄存器 _13	SAR_13	R/W	H'00000000	H'FFFE10D0	16、32
	DMA 目标地址寄存器 _13	DAR_13	R/W	H'00000000	H'FFFE10D4	16、32
	DMA 传送计数寄存器 _13	DMATCR_13	R/W	H'00000000	H'FFFE10D8	16、32
	DMA 通道控制寄存器 _13	CHCR_13	R/W*1	H'00000000	H'FFFE10DC	8、16、32
	DMA 重加载源地址寄存器 _13	RSAR_13	R/W	H'00000000	H'FFFE11D0	16、32
	DMA 重加载目标地址寄存器 _13	RDAR_13	R/W	H'00000000	H'FFFE11D4	16、32
	DMA 重加载传送计数寄存器 _13	RDMATCR_13	R/W	H'00000000	H'FFFE11D8	16、32
14	DMA 源地址寄存器 _14	SAR_14	R/W	H'00000000	H'FFFE10E0	16、32
	DMA 目标地址寄存器 _14	DAR_14	R/W	H'00000000	H'FFFE10E4	16、32
	DMA 传送计数寄存器 _14	DMATCR_14	R/W	H'00000000	H'FFFE10E8	16、32
	DMA 通道控制寄存器 _14	CHCR_14	R/W*1	H'00000000	H'FFFE10EC	8、16、32
	DMA 重加载源地址寄存器 _14	RSAR_14	R/W	H'00000000	H'FFFE11E0	16、32
	DMA 重加载目标地址寄存器 _14	RDAR_14	R/W	H'00000000	H'FFFE11E4	16、32
	DMA 重加载传送计数寄存器 _14	RDMATCR_14	R/W	H'00000000	H'FFFE11E8	16、32

通道	寄存器名	略称	R/W	初始值	地址	存取长度
15	DMA 源地址寄存器 _15	SAR_15	R/W	H'00000000	H'FFFE10F0	16、32
	DMA 目标地址寄存器 _15	DAR_15	R/W	H'00000000	H'FFFE10F4	16、32
	DMA 传送计数寄存器 _15	DMATCR_15	R/W	H'00000000	H'FFFE10F8	16、32
	DMA 通道控制寄存器 _15	CHCR_15	R/W*1	H'00000000	H'FFFE10FC	8、16、32
	DMA 重加载源地址寄存器 _15	RSAR_15	R/W	H'00000000	H'FFFE11F0	16、32
	DMA 重加载目标地址寄存器 _15	RDAR_15	R/W	H'00000000	H'FFFE11F4	16、32
	DMA 重加载传送计数寄存器 _15	RDMATCR_15	R/W	H'00000000	H'FFFE11F8	16、32
共用	DMA 操作寄存器	DMAOR	R/W*2	H'0000	H'FFFE1200	8、16
0/1	DMA 扩展资源选择器 0	DMARS0	R/W	H'0000	H'FFFE1300	16
2/3	DMA 扩展资源选择器 1	DMARS1	R/W	H'0000	H'FFFE1304	16
4/5	DMA 扩展资源选择器 2	DMARS2	R/W	H'0000	H'FFFE1308	16
6/7	DMA 扩展资源选择器 3	DMARS3	R/W	H'0000	H'FFFE130C	16
8/9	DMA 扩展资源选择器 4	DMARS4	R/W	H'0000	H'FFFE1310	16
10/11	DMA 扩展资源选择器 5	DMARS5	R/W	H'0000	H'FFFE1314	16
12/13	DMA 扩展资源选择器 6	DMARS6	R/W	H'0000	H'FFFE1318	16
14/15	DMA 扩展资源选择器 7	DMARS7	R/W	H'0000	H'FFFE131C	16

【注】 *1 为了清除标志，CHCR_n 的 HE 和 TE 位只能在读“1”后写“0”。

*2 为了清除标志，DMAOR 的 AE 和 NMIF 位只能在读“1”后写“0”。

11.3.1 DMA 源地址寄存器 (SAR)

SAR 是 32 位可读写寄存器，指定 DMA 传送源的地址，在 DMA 运行中表示下一个传送源地址。在单地址模式中，当传送源是带 DACK 的外部设备时，忽视 SAR。

在以字（2 字节）、长字（4 字节）、16 字节为单位进行数据传送时，必须分别指定 2 字节、4 字节、16 字节的边界地址。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

11.3.2 DMA 目标地址寄存器 (DAR)

DAR 是 32 位可读写寄存器，指定 DMA 传送目标的地址，在 DMA 运行中表示下一个传送目标地址。在单地址模式中，当传送目标是带 DACK 的外部设备时，忽视 DAR。

在以字（2 字节）、长字（4 字节）、16 字节为单位进行数据传送时，必须分别指定 2 字节、4 字节、16 字节的边界地址。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

11.3.3 DMA 传送计数寄存器 (DMATCR)

DMATCR 是 32 位可读写寄存器，指定 DMA 的传送次数。当设定值是“H'00000001”时，传送次数为 1 次；当设定值是“H'00FFFFFF”时，传送次数为 16777215 次；当设定值是“H'00000000”时，传送次数为 16777216 次（最大传送次数）。在 DMA 传送中，DMATCR 表示剩余的传送次数。

DMATCR 高 8 位的读写值都为“0”。

在进行 16 字节传送时，传送 1 次 16 字节（128 位）就进行 1 次计数。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

11.3.4 DMA 通道控制寄存器 (CHCR)

CHCR 是 32 位可读写寄存器，控制 DMA 的传送模式。

对于决定外部引脚 DREQ、DACK、TEND 规格的位 (DO、AM、AL、DL、DS、TL)，在通道 0 中可读写，但是在通道 1 ~ 15 中对应的位是保留位。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TC	—	RLD SAR	RLD DAR	—	DAF	SAF	—	DO	TL	—	TE MASK	HE	HIE	AM	AL
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R/W	R/W	R	R/W	R/W	R	R/W	R/W	R	R/W	R/(W)*	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DM[1:0]		SM[1:0]		RS[3:0]			DL	DS	TB	TS[1:0]		IE	TE	DE	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/(W)*	R/W

【注】* 为了清除标志，只能在读“1”后写“0”。

位	位名	初始值	R/W	说明
31	TC	0	R/W	传送计数模式 设定 1 次传送请求是传送 1 次还是传送 DMATCR 的设定次数。此功能只在内部外围模块请求时有效。如果将 TC 位置“0”，就不能将 TB 位置“1”（突发模式）。如果将传送请求源设定为多功能定时器脉冲单元 2、比较匹配定时器、控制器局域网、CD-ROM 解码器和 A/D 转换器以外的模块，就不能将 TC 位置“1”。 0: 1 次传送请求传送 1 次 1: 1 次传送请求传送 DMATCR 的设定次数
30	—	0	R	保留位 读写值都为“0”。
29	RLDSAR	0	R/W	SAR 重加载功能 ON/OFF 将 SAR 和 DMATCR 的重加载功能设定为有效 (ON) 或者无效 (OFF)。 0: 将 SAR 和 DMATCR 的重加载功能设定为无效 (OFF) 1: 将 SAR 和 DMATCR 的重加载功能设定为有效 (ON)
28	RLDDAR	0	R/W	DAR 重加载功能 ON/OFF 将 DAR 和 DMATCR 的重加载功能设定为有效 (ON) 或者无效 (OFF)。 0: 将 DAR 和 DMATCR 的重加载功能设定为无效 (OFF) 1: 将 DAR 和 DMATCR 的重加载功能设定为有效 (ON)
27	—	0	R	保留位 读写值都为“0”。
26	DAF	0	R/W	目标地址固定 16 字节传送 传送长度 (TS[1:0]) 为 16 字节，在目标地址模式 (DM[1:0]) 为固定地址时有效。 0: 将 16 字节的数据传送到 DAR 寄存器设定的地址 写地址为 DAR 寄存器设定的地址 +H'0、+H'4、+H'8、+H'C。 1: 将 4 字节的数据传送 4 次到 DAR 寄存器设定的地址 写地址固定为 DAR 寄存器设定的地址。此功能只能用于 CD-ROM 解码器、USB2.0 主机 / 功能模块、采样率转换器、瑞萨四路串行外围接口、SD 主机接口、MMC 主机接口。

位	位名	初始值	R/W	说明
25	SAF	0	R/W	源地址固定 16 字节传送 传送长度 (TS[1:0]) 为 16 字节, 在源地址模式 (SM[1:0]) 为固定地址时有效。 0: 从 SAR 寄存器设定的地址传送 16 字节的数据 读地址为 SAR 寄存器设定的地址 +H'0、+H'4、+H'8、+H'C。 1: 从 SAR 寄存器设定的地址传送 4 次 4 字节的数据 读地址固定为 SAR 寄存器设定的地址。此功能只能用于 CD-ROM 解码器、USB2.0 主机 / 功能模块、采样率转换器、瑞萨四路串行外围接口、SD 主机接口、MMC 主机接口。
24	—	0	R	保留位 读写值都为“0”。
23	DO	0	R/W	DMA 超限 选择是用超限 0 检测 DREQ 还是用超限 1 检测 DREQ。此位只在 CHCR_0 的电平检测时有效, 在 CHCR_1 ~ 15 中是保留位, 读写值都为“0”。 0: 用超限 0 检测 DREQ 1: 用超限 1 检测 DREQ
22	TL	0	R/W	传送结束电平 指定 TEND 信号是高电平有效还是低电平有效。此位只在 CHCR_0 有效, 在 CHCR_1 ~ 15 中是保留位, 读写值都为“0”。 0: TEND 为输出低电平有效 1: TEND 为输出高电平有效
21	—	0	R	保留位 读写值都为“0”。
20	TEMASK	0	R/W	TE 设置屏蔽 当 TE 位被置“1”时, 指示 DMA 不停止传送。通过和 SAR 重加载功能或者 DAR 重加载功能一起设定此位, 能在停止传送请求前的期间内执行 DMA 传送。当检测到自动请求和外部请求的上升 / 下降沿时, 忽视此位的设定, 如果 TE 位被置位, 就停止 DMA 传送。另外, 此功能在 RLDSAR 位或者 RLDDAR 位被置“1”时有效。 0: 如果 TE 位被置位, 就结束 DMA 传送。 1: 即使 TE 位被置位, 也继续 DMA 传送。
19	HE	0	R/(W)*	半结束标志 如果传送次数大于等于开始传送前设定的 DMATCR 值的一半, 就将 HE 位置“1”。在传送次数小于开始传送前设定的 DMATCR 值的一半的状态下, 如果因 NMI 中断或者 DMA 地址错误引起传送结束, 或者通过清除 DE 位和 DMAOR 的 DME 位来结束传送, 就不将 HE 位置位。在 HE 位被置位后, 如果因 NMI 中断或者 DMA 地址错误引起传送结束, 或者通过清除 DE 位和 DMAOR 的 DME 位来结束传送, HE 位就保持置位的状态。要清除 HE 位时, 必须在读 HE 位的“1”后写“0”。 0: 在 DMA 传送中或者在中止 DMA 传送时, $DMATCR > (\text{传送前设定的 } DMATCR)/2$ [清除条件] • 在读 HE 位的“1”后写“0”时 1: $DMATCR \leq (\text{传送前设定的 } DMATCR)/2$

位	位名	初始值	R/W	说明
18	HIE	0	R/W	<p>半结束中断允许</p> <p>在传送次数为开始传送前设定的 DMATCR 值的一半时，指定是否向 CPU 请求中断。如果在将 HIE 位置“1”的状态下 HE 位被置位，就向 CPU 请求中断。</p> <p>0: 在 DMATCR=(传送前设定的 DMATCR)/2 时，禁止中断请求。 1: 在 DMATCR=(传送前设定的 DMATCR)/2 时，允许中断请求。</p>
17	AM	0	R/W	<p>应答模式</p> <p>在双地址模式中，选择是在数据读周期还是数据写周期输出 DACK 和 TEND。</p> <p>在单地址模式中，与此位的指定无关，总是输出 DACK 和 TEND。</p> <p>此位只在 CHCR_0 中有效，在 CHCR_1~15 中是保留位，读写值都为“0”。</p> <p>0: 在读周期中输出 DACK 和 TEND (双地址模式) 1: 在写周期中输出 DACK 和 TEND (双地址模式)</p>
16	AL	0	R/W	<p>应答电平</p> <p>指定 DACK 信号是高电平有效还是低电平有效。此位只在 CHCR_0 中有效，在 CHCR_1~15 中是保留位，读写值都为“0”。</p> <p>0: DACK 为输出低电平有效 1: DACK 为输出高电平有效</p>
15、14	DM[1:0]	00	R/W	<p>目标地址模式</p> <p>指定 DMA 传送目标地址的增减 (在单地址模式中，当将数据传送到带 DACK 的外部设备时，忽视 DM1 位和 DM0 位)。</p> <p>00: 目标地址固定 01: 目标地址增加 (在以字节为单位传送时 +1；在以字为单位传送时 +2；在以长字为单位传送时 +4；在以 16 字节为单位传送时 +16) 10: 目标地址减少 (在以字节为单位传送时 -1；在以字为单位传送时 -2；在以长字为单位传送时 -4；在以 16 字节为单位传送时禁止设定) 11: 禁止设定</p>
13、12	SM[1:0]	00	R/W	<p>源地址模式</p> <p>指定 DMA 传送源地址的增减 (在单地址模式中，当从带 DACK 的外部设备传送数据时，忽视 SM1 位和 SM0 位)。</p> <p>00: 源地址固定 01: 源地址增加 (在以字节为单位传送时 +1；在以字为单位传送时 +2；在以长字为单位传送时 +4；在以 16 字节为单位传送时 +16) 10: 源地址减少 (在以字节为单位传送时 -1；在以字为单位传送时 -2；在以长字为单位传送时 -4；在以 16 字节为单位传送时禁止设定) 11: 禁止设定</p>

位	位名	初始值	R/W	说明
11 ~ 8	RS[3:0]	0000	R/W	<p>资源选择</p> <p>指定输入到此模块的传送请求源。必须在 DMA 允许位 (DE) 为“0”的状态下更改传送请求源。</p> <p>0000: 外部请求、双地址模式</p> <p>0001: 禁止设定</p> <p>0010: 外部请求、单地址模式 外部地址空间 → 带 DACK 的外部设备</p> <p>0011: 外部请求、单地址模式 带 DACK 的外部设备 → 外部地址空间</p> <p>0100: 自动请求</p> <p>0101: 禁止设定</p> <p>0110: 禁止设定</p> <p>0111: 禁止设定</p> <p>1000: DMA 扩展资源选择器</p> <p>1001: 控制器局域网 通道 0</p> <p>1010: 控制器局域网 通道 1</p> <p>1011: 禁止设定</p> <p>1100: 禁止设定</p> <p>1101: 禁止设定</p> <p>1110: 禁止设定</p> <p>1111: 禁止设定</p> <p>【注】 外部请求的指定只在 CHCR_0 中有效。在 CHCR_1 ~ 15 中, 不能指定外部请求。</p>
7 6	DL DS	0 0	R/W R/W	<p>DREQ 电平</p> <p>DREQ 边沿选择</p> <p>选择 DREQ 输入的检测方法和检测电平。</p> <p>这些位只在 CHCR_0 中有效, 在 CHCR_1 ~ 15 中是保留位, 读写值都为“0”。</p> <p>如果将传送请求源指定为内部外围模块或者自动请求时, 这些位就无效。</p> <p>00: 低电平检测</p> <p>01: 下降沿检测</p> <p>10: 高电平检测</p> <p>11: 上升沿检测</p>
5	TB	0	R/W	<p>传送总线模式</p> <p>选择 DMA 传送的总线模式。但是, 如果将 TC 位置“0”, 就不能设定突发模式。</p> <p>0: 周期挪用模式</p> <p>1: 突发模式</p>
4、3	TS[1:0]	00	R/W	<p>传送长度</p> <p>选择 DMA 的传送单位。当传送源或者传送目标是已指定传送长度的内部外围模块的寄存器时, 必须选择该传送长度。</p> <p>00: 以字节为单位</p> <p>01: 以字 (2 字节) 为单位</p> <p>10: 以长字 (4 字节) 为单位</p> <p>11: 以 16 字节为单位 (长字 ×4)</p>

位	位名	初始值	R/W	说明
2	IE	0	R/W	<p>中断允许</p> <p>指定在结束 DMA 传送时是否向 CPU 请求中断。如果在将 IE 位置“1”的状态下 TE 位被置位，就向 CPU 请求中断（DEI）。</p> <p>0: 禁止中断请求 1: 允许中断请求</p>
1	TE	0	R/(W)*	<p>传送结束标志</p> <p>当 DMATCR 的值为“0”并且结束 DMA 传送时，就将 TE 位置“1”。在 DMATCR 的值不为“0”时，如果因 NMI 中断或者 DMA 地址错误引起传送结束，或者通过清除 DE 位和 DMA 操作寄存器（DMAOR）的 DME 位来结束传送，就不将 TE 位置位。要清除 TE 位时，必须在读 TE 位的“1”后写“0”。</p> <p>当 TEMASK 位为“0”并且 TE 位被置位时，即使将 DE 位置“1”也不允许传送。</p> <p>0: 正在进行 DMA 传送或者中止 DMA 传送 [清除条件]</p> <ul style="list-style-type: none"> 在读 TE 位的“1”后写“0”时 <p>1: (DMATCR=0) 结束 DMA 传送</p>
0	DE	0	R/W	<p>DMA 允许</p> <p>允许或者禁止 DMA 传送。在自动请求模式中，如果将 DE 位和 DMAOR 的 DME 位置“1”，就开始传送。但是，TE 位、DMAOR 的 NMIF 位和 AE 位必须全部为“0”。在外部请求和外围模块请求时，如果在将 DE 位和 DME 位置“1”后还有相应设备或者相应外围模块的 DMA 传送请求，就开始传送。在外部请求的低电平检测或者高电平检测以及外围模块请求的情况下，当 TEMASK 位为“1”时，NMIF 位和 AE 位必须为“0”。当 TEMASK 位为“0”时，TE 位也必须为“0”。在外部请求的上升沿检测或者下降沿检测时，自动请求模式相同，TE 位、NMIF 位和 AE 位必须全部为“0”。如果清除 DE 位，就能中止传送。</p> <p>0: 禁止 DMA 传送 1: 允许 DMA 传送</p>

【注】* 为了清除标志，只能在读“1”后写“0”。

11.3.5 DMA 重加载源地址寄存器 (RSAR)

RSAR 是 32 位可读写寄存器。

如果将 SAR 重加载功能置为 ON，就在结束当前的 DMA 传送时将 RSAR 的内容写到源地址寄存器 (SAR)。此时，在 DMA 传送过程中，能通过预先设定来预置下一次 DMA 传送的设定。当将 SAR 重加载功能置为 OFF 时，对运行没有任何影响。

在以字 (2 字节)、长字 (4 字节)、16 字节为单位进行数据传送时，必须分别指定 2 字节、4 字节、16 字节的边界地址。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

11.3.6 DMA 重加载目标地址寄存器 (RDAR)

RDAR 是 32 位可读写寄存器。

如果将 DAR 重加载功能置为 ON，就在结束当前的 DMA 传送时将 RDAR 的内容写到目标地址寄存器 (DAR)。此时，在 DMA 传送过程中，能通过预先设定来预置下一次 DMA 传送的设定。当将 DAR 重加载功能置为 OFF 时，对运行没有任何影响。

在以字 (2 字节)、长字 (4 字节)、16 字节为单位进行数据传送时，必须分别指定 2 字节、4 字节、16 字节的边界地址。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

11.3.7 DMA 重加载传送计数寄存器 (RDMATCR)

RDMATCR 是 32 位可读写寄存器。

如果将 SAR 重加载功能或者 DAR 重加载功能置为 ON，就在结束当前的 DMA 传送时将 RDMATCR 的内容写到传送计数寄存器 (DMATCR)。此时，在 DMA 传送过程中，能通过预先设定来预置下一次 DMA 传送的设定。当将 SAR 重加载功能或者 DAR 重加载功能置为 OFF 时，对运行没有任何影响。

RDMATCR 高 8 位的读写值都为“0”。

和 DMATCR 相同，当设定值为“H'00000001”时，传送次数为 1 次；当设定值为“H'00FFFFFF”时，传送次数为 16777215 次；当设定值为“H'00000000”时，传送次数为 16777216 次（最大传送次数）。在进行 16 字节传送时，传送 1 次 16 字节（128 位）就计数 1 次。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

11.3.8 DMA 操作寄存器 (DMAOR)

DMAOR 是 16 位可读写寄存器，指定 DMA 传送时的通道优先级，并且表示 DMA 的传送状态（状态）。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CMS[1:0]	—	—	PR[1:0]	—	—	—	—	—	—	—	AE	NMIF	DME
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R	R	R/(W)*	R/(W)*

【注】* 为了清除标志，只能在读“1”后写“0”。

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13、12	CMS[1:0]	00	R/W	周期挪用模式选择 选择周期挪用模式中的正常模式或者间歇模式。 为了将间歇模式置为有效，全部通道的总线模式都必须是周期挪用模式。 00: 正常模式 01: 禁止设定 10: 间歇模式 16 B ϕ 时钟的每 16 个时钟（周期）执行 1 次 DMA 传送 11: 间歇模式 64 B ϕ 时钟的每 64 个时钟（周期）执行 1 次 DMA 传送
11、10	—	全 0	R	保留位 读写值都为“0”。

位	位名	初始值	R/W	说明
9、8	PR[1:0]	00	R/W	<p>优先级模式</p> <p>当多个通道同时发生传送请求时，决定要执行的通道优先级。</p> <p>00: 固定模式 1: CH0 > CH1 > CH2 > CH3 > CH4 > CH5 > CH6 > CH7 > CH8 > CH9 > CH10 > CH11 > CH12 > CH13 > CH14 > CH15</p> <p>01: 固定模式 2: CH0 > CH8 > CH1 > CH9 > CH2 > CH10 > CH3 > CH11 > CH4 > CH12 > CH5 > CH13 > CH6 > CH14 > CH7 > CH15</p> <p>10: 禁止设定</p> <p>11: 禁止设定</p>
7 ~ 3	—	全 0	R	<p>保留位</p> <p>读写值都为“0”。</p>
2	AE	0	R(W)*	<p>地址错误标志</p> <p>表示发生由此模块引起的地址错误。当 AE 位被置位时，即使将 CHCR 的 DE 位和 DMAOR 的 DME 位置“1”，也不允许 DMA 传送。为了清除 AE 位，必须在读 AE 位的“1”后写“0”。</p> <p>0: 没有发生此模块引起的地址错误</p> <p>1: 发生此模块引起的地址错误</p> <p>[清除条件]</p> <ul style="list-style-type: none"> 在读 AE 位的“1”后写“0”时
1	NMIF	0	R(W)*	<p>NMI 标志</p> <p>表示发生 NMI 中断。当 NMIF 位被置位时，即使将 CHCR 的 DE 位和 DMAOR 的 DME 位置“1”，也不允许 DMA 传送。为了清除 NMIF 位，必须在读 NMIF 位的“1”后写“0”。</p> <p>当 NMI 输入时，对于正在执行的 DMA 传送，进行 1 个传送单位的传送。当此模块不在运行时，即使输入 NMI 中断，也将 NMIF 位置“1”。</p> <p>0: 没有发生 NMI 中断</p> <p>1: 发生 NMI 中断</p> <p>[清除条件]</p> <ul style="list-style-type: none"> 在读 NMIF 位的“1”后写“0”时
0	DME	0	R/W	<p>DMA 主控允许</p> <p>允许或者禁止全部通道的 DMA 传送。如果将 DME 位和 CHCR 的 DE 位置“1”，就允许 DMA 传送。但是，进行传送的通道 CHCR 的 TE 位、DMAOR 的 NMIF 位和 AE 位必须全部为“0”。如果清除 DME 位，就中止全部通道的 DMA 传送。</p> <p>0: 禁止全部通道的 DMA 传送</p> <p>1: 允许全部通道的 DMA 传送</p>

【注】* 为了清除标志，只能在读“1”后写“0”。

如果在结束一次传送后改变优先权模式位的设定，就对优先级进行初始化。

例如，在固定模式 2 中重新设定时，优先级为 CH0 > CH8 > CH1 > CH9 > CH2 > CH10 > CH3 > CH11 > CH4 > CH12 > CH5 > CH13 > CH6 > CH14 > CH7 > CH15；在固定模式 1 中重新设定时，优先级为 CH0 > CH1 > CH2 > CH3 > CH4 > CH5 > CH6 > CH7 > CH8 > CH9 > CH10 > CH11 > CH12 > CH13 > CH14 > CH15。

在发生地址错误时，此模块的内部处理运行如下：

- 当不发生地址错误时：Read（传送源 → 此模块内部）→ Write（此模块内部 → 传送目标）
- 在源地址发生地址错误时：Nop → Nop
- 在目标地址发生地址错误时：Read → Nop

11.3.9 DMA 扩展资源选择器 0 ~ 7 (DMARS0 ~ DMARS7)

DMARS 是 16 位可读写寄存器，按各通道指定外围模块的 DMA 传送请求源。DMARS0 设定为通道 0 和通道 1，DMARS1 设定为通道 2 和通道 3。能设定的组合如表 11.3 所示。

此寄存器能设定接受以下启动源的传送请求。

能发行内部外围模块请求的模块如下：

- FIFO 内置型串行通信接口：16 个源
- I²C 总线接口 3：8 个源
- A/D 转换器：1 个源
- 多功能定时器脉冲单元 2：5 个源
- 比较匹配定时器：2 个源
- USB2.0 主机/功能模块：2 个源
- NAND 闪存控制器：2 个源
- 控制器局域网：3 个源
- 串行音频接口：7 个源
- 采样率转换器：6 个源
- 音频发生器：4 个源
- 瑞萨 SPDIF 接口：2 个源
- CD-ROM 解码器：1 个源
- SD 主机接口：4 个源
- MMC 主机接口：2 个源
- 瑞萨串行外围接口：4 个源
- 瑞萨四路串行外围接口：4 个源
- 带 FIFO 的时钟同步串行 I/O：2 个源
- 马达控制 PWM 定时器：2 个源

对于控制器局域网 3 个源的传送请求的接受，能通过 DMA 通道控制寄存器的 RS[3:0] 进行设定，而不需要设定 DMA 扩展资源选择器。

• DMARS0

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH1 MID[5:0]					CH1 RID[1:0]		CH0 MID[5:0]					CH0 RID[1:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

• DMARS1

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH3 MID[5:0]					CH3 RID[1:0]		CH2 MID[5:0]					CH2 RID[1:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

• DMARS2

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH5 MID[5:0]					CH5 RID[1:0]		CH4 MID[5:0]					CH4 RID[1:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

• DMARS3

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH7 MID[5:0]					CH7 RID[1:0]		CH6 MID[5:0]					CH6 RID[1:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

• DMARS4

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH9 MID[5:0]					CH9 RID[1:0]		CH8 MID[5:0]					CH8 RID[1:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

• DMARS5

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH11 MID[5:0]					CH11 RID[1:0]		CH10 MID[5:0]					CH10 RID[1:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

• DMARS6

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH13 MID[5:0]					CH13 RID[1:0]		CH12 MID[5:0]					CH12 RID[1:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

• DMARS7

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH15 MID[5:0]					CH15 RID[1:0]		CH14 MID[5:0]					CH14 RID[1:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

各模块的传送请求设定以下的 MID 和 RID。

表 11.3 DMARS 的设定

外围模块	1 个通道的设定值 ({MID、RID})	MID	RID	功能
USB2.0 主机 / 功能模块	H'03	B'000000	B'11	通道 0 FIFO
	H'07	B'000001	B'11	通道 1 FIFO
瑞萨 SPDIF 接口	H'09	B'000010	B'01	发送
	H'0A	B'000010	B'10	接收
SD 主机接口 0	H'11	B'000100	B'01	写 SD_BUF
	H'12		B'10	读 SD_BUF
SD 主机接口 1	H'A9	B'101010	B'01	写 SD_BUF
	H'AA		B'10	读 SD_BUF
MMC 主机接口	H'AD	B'101011	B'01	发送
	H'AE		B'10	接收
带 FIFO 的时钟同步串行 I/O	H'19	B'000110	B'01	发送
	H'1A		B'10	接收
串行音频接口 通道 0	H'21	B'001000	B'01	发送
	H'22		B'10	接收

外围模块	1 个通道的设定值 ({MID、RID})	MID	RID	功能
串行音频接口 通道 1	H'27	B'001001	B'11	—
串行音频接口 通道 2	H'2B	B'001010	B'11	—
串行音频接口 通道 3	H'2F	B'001011	B'11	—
串行音频接口 通道 4	H'B9	B'101110	B'01	—
串行音频接口 通道 5	H'BD	B'101111	B'01	—
马达控制 PWM 定时器 通道 1	H'33	B'001100	B'11	—
马达控制 PWM 定时器 通道 2	H'37	B'001101	B'11	—
音频发生器 0	H'C5	B'110001	B'01	—
音频发生器 1	H'C9	B'110010	B'01	—
音频发生器 2	H'CD	B'110011	B'01	—
音频发生器 3	H'D5	B'110101	B'01	—
采样率转换器 通道 0	H'41	B'010000	B'01	输入数据空
	H'42		B'10	输出数据满
采样率转换器 通道 1	H'45	B'010001	B'01	输入数据空
	H'46		B'10	输出数据满
采样率转换器 通道 2	H'B5	B'101101	B'01	输入数据空
	H'B6		B'10	输出数据满
瑞萨串行外围接口 通道 0	H'51	B'010100	B'01	发送
	H'52		B'10	接收
瑞萨串行外围接口 通道 1	H'55	B'010101	B'01	发送
	H'56		B'10	接收
瑞萨四路串行外围接口 通道 0	H'A1	B'101000	B'01	发送
	H'A2		B'10	接收
瑞萨四路串行外围接口 通道 1	H'A5	B'101001	B'01	发送
	H'A6		B'10	接收
I ² C 总线接口 3 通道 0	H'61	B'011000	B'01	发送
	H'62		B'10	接收
I ² C 总线接口 3 通道 1	H'65	B'011001	B'01	发送
	H'66		B'10	接收
I ² C 总线接口 3 通道 2	H'69	B'011010	B'01	发送
	H'6A		B'10	接收
I ² C 总线接口 3 通道 3	H'C1	B'110000	B'01	发送
	H'C2		B'10	接收
CD-ROM 解码器	H'73	B'011100	B'11	—

外围模块	1 个通道的设定值 ({MID、RID})	MID	RID	功能
FIFO 内置型串行通信接口 通道 0	H'81	B'100000	B'01	发送
	H'82		B'10	接收
FIFO 内置型串行通信接口 通道 1	H'85	B'100001	B'01	发送
	H'86		B'10	接收
FIFO 内置型串行通信接口 通道 2	H'89	B'100010	B'01	发送
	H'8A		B'10	接收
FIFO 内置型串行通信接口 通道 3	H'8D	B'100011	B'01	发送
	H'8E		B'10	接收
FIFO 内置型串行通信接口 通道 4	H'91	B'100100	B'01	发送
	H'92		B'10	接收
FIFO 内置型串行通信接口 通道 5	H'95	B'100101	B'01	发送
	H'96		B'10	接收
FIFO 内置型串行通信接口 通道 6	H'99	B'100110	B'01	发送
	H'9A		B'10	接收
FIFO 内置型串行通信接口 通道 7	H'9D	B'100111	B'01	发送
	H'9E		B'10	接收
A/D 转换器	H'B3	B'101100	B'11	—
NAND 闪存控制器	H'BB	B'101110	B'11	数据的发送 / 接收
	H'BF	B'101111	B'11	管理码的发送 / 接收
多功能定时器脉冲单元 通道 0	H'E3	B'111000	B'11	—
多功能定时器脉冲单元 通道 1	H'E7	B'111001	B'11	—
多功能定时器脉冲单元 通道 2	H'EB	B'111010	B'11	—
多功能定时器脉冲单元 通道 3	H'EF	B'111011	B'11	—
多功能定时器脉冲单元 通道 4	H'F3	B'111100	B'11	—
比较匹配定时器 通道 0	H'FB	B'111110	B'11	—
比较匹配定时器 通道 1	H'FF	B'111111	B'11	—

不能保证设定表 11.3 以外的 MID 或者 RID 时的运行。DMARS 寄存器的传送请求只在将 CHCR_0 ~ 15 寄存器的资源选择位 (RS[3:0]) 置为 “B'1000” 时有效, 当资源选择位不为 “B'1000” 时, 即使设定 DMARS 也不接受传送请求源。

11.4 运行说明

当发生此模块的传送请求时，按照决定的通道优先级开始传送，在满足传送结束条件时结束传送。传送请求有自动请求、外部请求和内部外围模块请求共 3 种模式。总线模式能选择突发模式或者周期挪用模式。

11.4.1 传送流程

在给 DMA 源地址寄存器 (SAR)、DMA 目标地址寄存器 (DAR)、DMA 传送计数寄存器 (DMATCR)、DMA 通道控制寄存器 (CHCR)、DMA 操作寄存器 (DMAOR)、3 个重加载寄存器 (RSAR、RDAR、RDMATCR) 和 DMA 扩展资源选择器 (DMARS) 设定目标传送条件后，此模块按照以下步骤进行数据传送：

1. 检查是否为传送允许状态 (DE=1、DME=1、TEMASK=0 并且 TE=0 或者 TEMASK=1、AE=0、NMIF=0)。
2. 如果在传送允许状态下发生传送请求，就传送 1 个传送单位的数据 (取决于 TS[1:0] 位的设定)。在自动请求模式中，如果将 DE 位和 DME 位置 “1”，就自动开始传送。每进行 1 次传送，DMATCR 的值就减 1。具体的传送流程因地址模式和总线模式而不同。
3. 如果传送次数超过指定次数的一半 (DMATCR 的值为初始值的 1/2) 并且 CHCR 的 HIE 位已被置 “1”，就向 CPU 请求 HEI 中断。
4. 如果在 TEMASK 位为 “0” 时结束指定次数的传送 (DMATCR 的值为 “0”)，就正常结束传送。此时如果 CHCR 的 IE 位已被置 “1”，就向 CPU 请求 DEI 中断。如果在 TEMASK 位为 “1” 时 DMATCR 的值变为 “0”，就在将 TE 位置 “1” 后将指定的 RSAR、RDAR 和 RDMATCR 的值重新加载到 SAR、DAR 和 DMATCR，继续进行传送，直到取消传送请求为止。
5. 如果发生此模块引起的地址错误或者 NMI 中断，就中止传送。即使将 CHCR 的 DE 位或者 DMAOR 的 DME 位清 “0”，也中止传送。

上述的流程图如图 11.2 所示。

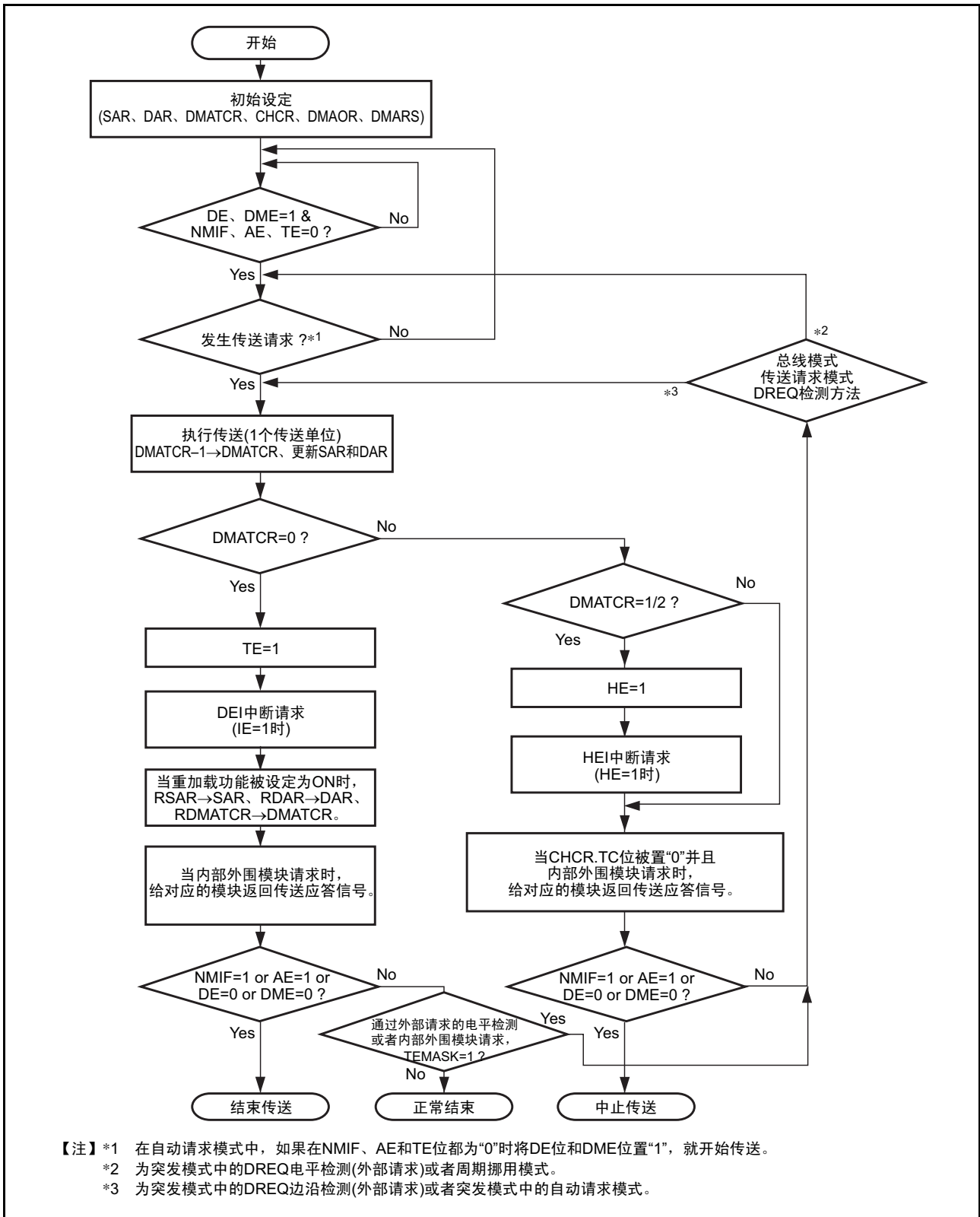


图 11.2 DMA 传送流程图

11.4.2 DMA 传送请求

基本的使用方法是让数据的传送源或者传送目标产生 DMA 传送请求，但是也有让既不是传送源又不是传送目标的外部设备或者内部外围模块产生 DMA 传送请求的使用方法。

传送请求有自动请求、外部请求和内部外围模块请求共 3 种。通过 CHCR_0 ~ CHCR_15 的 RS[3:0] 位和 DMARS0 ~ DMARS7 寄存器选择传送请求。

(1) 自动请求模式

如同存储器之间的传送或者不能产生传送请求的内部外围模块和存储器之间的传送，当没有来自外部的传送请求信号时，自动请求模式在此模块内部自动产生传送请求信号。如果将 CHCR_0 ~ CHCR_15 的 DE 位和 DMAOR 的 DME 位置“1”，就开始传送。但是，CHCR_0 ~ CHCR_15 的 TE 位、DMAOR 的 AE 和 NMIF 位必须全部为“0”。

(2) 外部请求模式

外部请求模式是通过 LSI 外部设备的传送请求信号（DREQ0）开始传送的模式。根据系统，从表 11.4 所示的模式中选择 1 种使用。如果在允许 DMA 传送（在电平检测时，DE=1、DME=1、TEMASK=0 并且 TE=0 或者 TEMASK=1、AE=0、NMIF=0；在边沿检测时，DE=1、DME=1、TE=0、AE=0、NMIF=0）时输入 DREQ，就开始 DMA 传送。

表 11.4 通过 RS 位选择外部请求模式

RS[3]	RS[2]	RS[1]	RS[0]	地址模式	传送源	传送目标
0	0	0	0	双地址模式	任意	任意
0	0	1	0	单地址模式	外部存储器或者存储器映射外部设备	带 DACK 的外部设备
			1		带 DACK 的外部设备	外部存储器或者存储器映射外部设备

如表 11.5 所示，通过 CHCR_0 的 DL 位和 DS 位，选择是通过边沿还是通过电平来检测 DREQ。传送请求源不一定需要是数据的传送源或者传送目标。在上升沿或者下降沿检测到突发模式时，1 次传送请求连续进行传送，直到 DMATCR 位变为“0”为止；在周期挪用模式中 1 次传送请求进行 1 次传送。

表 11.5 通过 DL 或者 DS 位选择外部请求检测

CHCR		外部请求的检测方法
DL	DS	
0	0	低电平检测
	1	下降沿检测
1	0	高电平检测
	1	上升沿检测

如果接受 DREQ，DREQ 引脚就处于不能接受请求的状态（非感应区）。在对已接受的 DREQ 输出应答 DACK 后，DREQ 引脚再次变为能接受请求的状态。

在通过电平检测使用 DREQ 时，根据从输出 DACK 到检测下一个 DREQ 的时序，有在执行和请求相同次数的传送后中止传送（超限 0）的情况以及在执行比请求多 1 次的传送后中止传送（超限 1）的情况。通过 CHCR 的 DO 位选择是超限 0 还是超限 1。

表 11.6 通过 DO 位选择外部请求检测

CHCR 的 DO 位	外部请求
0	超限 0
1	超限 1

(3) 内部外围模块请求模式

在内部外围模块请求模式中，通过内部外围模块的 DMA 传送请求信号进行传送。

内部外围模块对此模块发出的 DMA 传送请求信号一览表如表 11.7 所示。

在选择内部外围模块请求模式时，如果是 DMA 传送允许状态（DE=1、DME=1、TEMASK=0 并且 TE=0 或者 TEMASK=1、AE=0、NMIF=0），就通过传送请求信号进行传送。

内部外围模块请求有固定传送源或者传送目标的情况，请参照表 11.7。

表 11.7 通过 RS3 ~ RS0 位选择内部外围模块请求模式

CHCR	DMARS		DMA 传送请求源	DMA 传送请求信号	传送源	传送目标	总线模式	
	RS[3:0]	MID						RID
1001	任意	任意	控制器局域网通道 0	RM0（接收结束）	MB0	任意	周期挪用	
1010	任意	任意	控制器局域网通道 1	RM0（接收结束）	MB0	任意		
1011	任意	任意	控制器局域网通道 2	RM0（接收结束）	MB0	任意		
1000	000000	11	USB2.0 主机 / 功能模块	USB_DMA0 （通道 0 接收 FIFO 满）	D0FIFO	任意		
				USB_DMA0 （通道 0 发送 FIFO 空）	任意	D0FIFO		
		000001		11	USB_DMA1 （通道 1 接收 FIFO 满）	D1FIFO		任意
					USB_DMA1 （通道 1 发送 FIFO 空）	任意		D1FIFO
	000010	01	瑞萨 SPDIF 接口	SPDIFTXI （发送模块 DMA 传送）	任意	TDAD		
		10		SPDIFRXI （接收模块 DMA 传送）	RDAD	任意		
	000100	01	SD 主机接口 0	SD_BUF 写	任意	数据寄存器		
		10		SD_BUF 读	数据寄存器	任意		
	101010	01	SD 主机接口 1	SD_BUF 写	任意	数据寄存器		
		10		SD_BUF 读	数据寄存器	任意		
	101011	01	MMC 主机接口	发送数据空	任意	数据寄存器		
		10		接收数据满	数据寄存器	任意		
000110	01	带 FIFO 的时钟	TXI（发送数据传送）	任意	SITDR			
	10	同步串行 I/O	RXI（接收数据传送）	SIRDR	任意			

CHCR	DMARS		DMA 传送请求源	DMA 传送请求信号	传送源	传送目标	总线模式
	RS[3:0]	MID					
1000	001000	01	串行音频接口	SSITX10 (发送数据空)	任意	SSIFTDR_0	周期挪用
		10	通道 0	SSIRX10 (接收数据满)	SSIFRDR_0	任意	
	001001	11	串行音频接口 通道 1	SSIRT11 (发送数据空)	任意	SSIFTDR_1	
				SSIRT11 (接收数据满)	SSIFRDR_1	任意	
	001010	11	串行音频接口 通道 2	SSIRT12 (发送数据空)	任意	SSIFTDR_2	
				SSIRT12 (接收数据满)	SSIFRDR_2	任意	
	001011	11	串行音频接口 通道 3	SSIRT13 (发送数据空)	任意	SSIFTDR_3	
				SSIRT13 (接收数据满)	SSIFRDR_3	任意	
	101110	01	串行音频接口 通道 4	SSIRT14 (发送数据空)	任意	SSIFTDR_4	
				SSIRT14 (接收数据满)	SSIFRDR_4	任意	
	101111	01	串行音频接口 通道 5	SSIRT15 (发送数据空)	任意	SSIFTDR_5	
				SSIRT15 (接收数据满)	SSIFRDR_5	任意	
	001100	11	马达控制 PWM 定时器 通道 1	CMI1 (PWM 比较匹配)	任意	PWBFR1	
	001101	11	马达控制 PWM 定时器 通道 2	CMI2 (PWM 比较匹配)	任意	PWBFR2	
	110001	01	音频发生器 0	SGDEI0	任意	SGLR_0	
	110010	01	音频发生器 1	SGDEI1	任意	SGLR_1	
	110011	01	音频发生器 2	SGDEI2	任意	SGLR_2	
	110101	01	音频发生器 3	SGDEI3	任意	SGLR_3	
	010000	01	采样率转换器 通道 0	IDEI0 (输入数据空)	任意	SRCIDR_0	
				ODFI0 (输出数据满)	SRCODR_0	任意	
	010001	01	采样率转换器 通道 1	IDEI1 (输入数据空)	任意	SRCIDR_1	
				ODFI1 (输出数据满)	SRCODR_1	任意	
	101101	01	采样率转换器 通道 2	IDEI2 (输入数据空)	任意	SRCIDR_2	
				ODFI2 (输出数据满)	SRCODR_2	任意	
	010100	01	瑞萨串行外围接口 通道 0	SPTI0 (发送缓冲器空)	任意	SPDR_0	
				SPRI0 (接收缓冲器满)	SPDR_0	任意	
	010101	01	瑞萨串行外围接口 通道 1	SPTI1 (发送缓冲器空)	任意	SPDR_1	
				SPRI1 (接收缓冲器满)	SPDR_1	任意	
	101000	01	瑞萨四路串行外围 接口 通道 0	SPTI0 (发送缓冲器空)	任意	SPDR_0	
				SPRI0 (接收缓冲器满)	SPDR_0	任意	
	101001	01	瑞萨四路串行外围 接口 通道 1	SPTI1 (发送缓冲器空)	任意	SPDR_1	
				SPRI1 (接收缓冲器满)	SPDR_1	任意	
	011000	01	I ² C 总线接口 3 通道 0	TXI0 (发送数据空)	任意	ICDRT_0	
				RXI0 (接收数据满)	ICDRR_0	任意	
	011001	01	I ² C 总线接口 3 通道 1	TXI1 (发送数据空)	任意	ICDRT_1	
				RXI1 (接收数据满)	ICDRR_1	任意	
011010	01	I ² C 总线接口 3 通道 2	TXI2 (发送数据空)	任意	ICDRT_2		
			RXI2 (接收数据满)	ICDRR_2	任意		

CHCR RS[3:0]	DMARS		DMA 传送请求源	DMA 传送请求信号	传送源	传送目标	总线模式
	MID	RID					
1000	110000	01	I ² C 总线接口 3 通道 3	TXI3 (发送数据空)	任意	ICDRT_3	周期挪用
		10		RXI3 (接收数据满)	ICDRR_3	任意	
	011100	11	CD-ROM 解码器	IREADY (解码结束)	STRMDOUT	任意	周期挪用 / 突发
100000	01	01	FIFO 内置型串行 通信接口 通道 0	TXI0 (发送 FIFO 数据空)	任意	SCFTDR_0	周期挪用
		10		RXI0 (接收 FIFO 数据满)	SCFRDR_0	任意	
100001	01	01	FIFO 内置型串行 通信接口 通道 1	TXI1 (发送 FIFO 数据空)	任意	SCFTDR_1	
		10		RXI1 (接收 FIFO 数据满)	SCFRDR_1	任意	
100010	01	01	FIFO 内置型串行 通信接口 通道 2	TXI2 (发送 FIFO 数据空)	任意	SCFTDR_2	
		10		RXI2 (接收 FIFO 数据满)	SCFRDR_2	任意	
100011	01	01	FIFO 内置型串行 通信接口 通道 3	TXI3 (发送 FIFO 数据空)	任意	SCFTDR_3	
		10		RXI3 (接收 FIFO 数据满)	SCFRDR_3	任意	
100100	01	01	FIFO 内置型串行 通信接口 通道 4	TXI4 (发送 FIFO 数据空)	任意	SCFTDR_4	
		10		RXI4 (接收 FIFO 数据满)	SCFRDR_4	任意	
100101	01	01	FIFO 内置型串行 通信接口 通道 5	TXI5 (发送 FIFO 数据空)	任意	SCFTDR_5	
		10		RXI5 (接收 FIFO 数据满)	SCFRDR_5	任意	
100110	01	01	FIFO 内置型串行 通信接口 通道 6	TXI6 (发送 FIFO 数据空)	任意	SCFTDR_6	
		10		RXI6 (接收 FIFO 数据满)	SCFRDR_6	任意	
100111	01	01	FIFO 内置型串行 通信接口 通道 7	TXI7 (发送 FIFO 数据空)	任意	SCFTDR_7	
		10		RXI7 (接收 FIFO 数据满)	SCFRDR_7	任意	
101100	11	11	A/D 转换器	ADI (A/D 转换结束)	ADDR	任意	
101110	11	11	NAND 闪存控制器	数据 发送 FIFO 数据空	任意	FLDTFIFO	周期挪用
				数据 接收 FIFO 数据满	FLDTFIFO	任意	
101111	11	11		管理码 发送 FIFO 数据空	任意	FLECFIFO	
				管理码 接收 FIFO 数据满	FLECFIFO	任意	
111000	11	11	多功能定时器脉冲 单元 2 通道 0	TGI0A (输入捕捉 / 比较匹配)	任意	任意	周期挪用 / 突发
111001	11	11	多功能定时器脉冲 单元 2 通道 1	TGI1A (输入捕捉 / 比较匹配)	任意	任意	
111010	11	11	多功能定时器脉冲 单元 2 通道 2	TGI2A (输入捕捉 / 比较匹配)	任意	任意	
111011	11	11	多功能定时器脉冲 单元 2 通道 3	TGI3A (输入捕捉 / 比较匹配)	任意	任意	
111100	11	11	多功能定时器脉冲 单元 2 通道 4	TGI4A (输入捕捉 / 比较匹配)	任意	任意	
111110	11	11	比较匹配定时器 通道 0	CMIO (比较匹配)	任意	任意	
111111	11	11	比较匹配定时器 通道 1	CM11 (比较匹配)	任意	任意	

11.4.3 通道优先级

如果同时对多个通道发生传送请求，此模块就按照决定的优先级进行传送。通道优先级能从固定模式 1 和固定模式 2 这 2 种模式中选择。

各模式优先级如下：

- 固定模式 1: CH0 > CH1 > CH2 > CH3 > CH4 > CH5 > CH6 > CH7 > CH8 > CH9 > CH10 > CH11 > CH12 > CH13 > CH14 > CH15
- 固定模式 2: CH0 > CH8 > CH1 > CH9 > CH2 > CH10 > CH3 > CH11 > CH4 > CH12 > CH5 > CH13 > CH6 > CH14 > CH7 > CH15

通过 DMAOR 的 PR1 位和 PR0 位选择这些模式。

11.4.4 DMA 的传送种类

根据存取传送源和传送目标的总线周期次数，将 DMA 传送分为单地址模式传送和双地址模式传送。具体的传送时序因总线模式而不同，总线模式有周期挪用模式和突发模式。此模块能支持的传送如表 11.8 示。

表 11.8 能支持的 DMA 传送

传送源	传送目标				
	带 DACK 的外部设备	外部存储器	存储器映射外部设备	内部外围模块	内部存储器
带 DACK 的外部设备	不能	双、单	双、单	不能	不能
外部存储器	双、单	双	双	双	双
存储器映射外部设备	双、单	双	双	双	双
内部外围模块	不能	双	双	双	双
内部存储器	不能	双	双	双	双

- 【注】
1. 双：双地址模式
 2. 单：单地址模式
 3. 内部外围模块只能对允许长字存取的寄存器进行 16 字节的传送。

(1) 地址模式

(a) 双地址模式

双地址模式是用于通过地址存取（选择）传送源和传送目标的模式。传送源和传送目标可以是外部也可以是内部。在此模式中，此模块在读周期中存取传送源，在写周期中存取传送目标，通过 2 个总线周期进行传送。此时，传送数据暂时被保存在此模块中。例如，对于图 11.3 所示的外部存储器之间的传送，在读周期中从一个外部存储器将数据读到此模块，然后在写周期中将该数据写到另一个外部存储器。

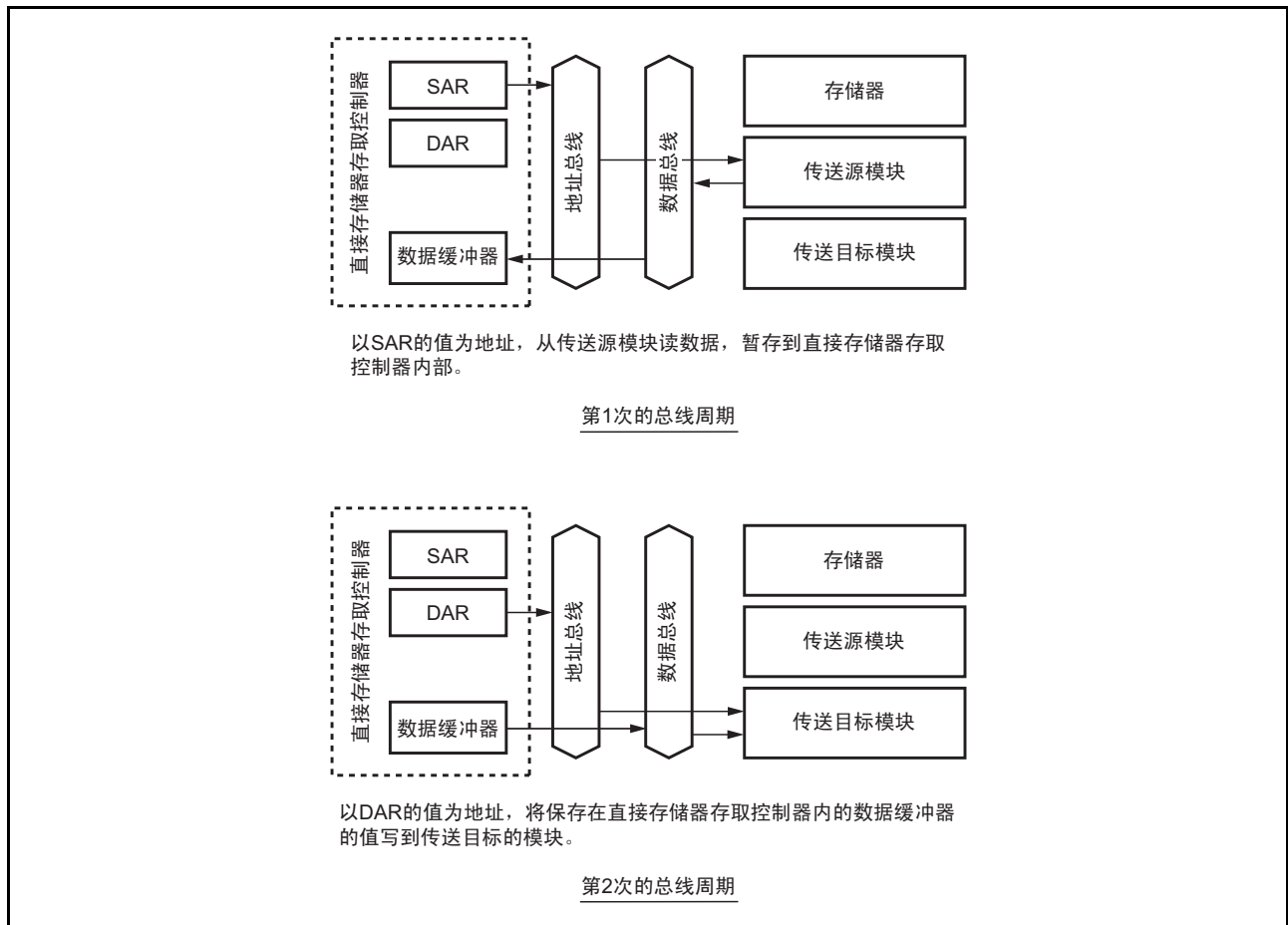


图 11.3 双地址模式的数据流程

传送请求可以是自动请求、外部请求或者内部外围模块请求。在双地址模式中，能在读周期或者写周期中输出 DACK。能通过 CHCR 的 AM 位设定是在读周期中还是在写周期中输出 DACK。

双地址模式中的 DMA 传送时序例子如图 11.4 所示。

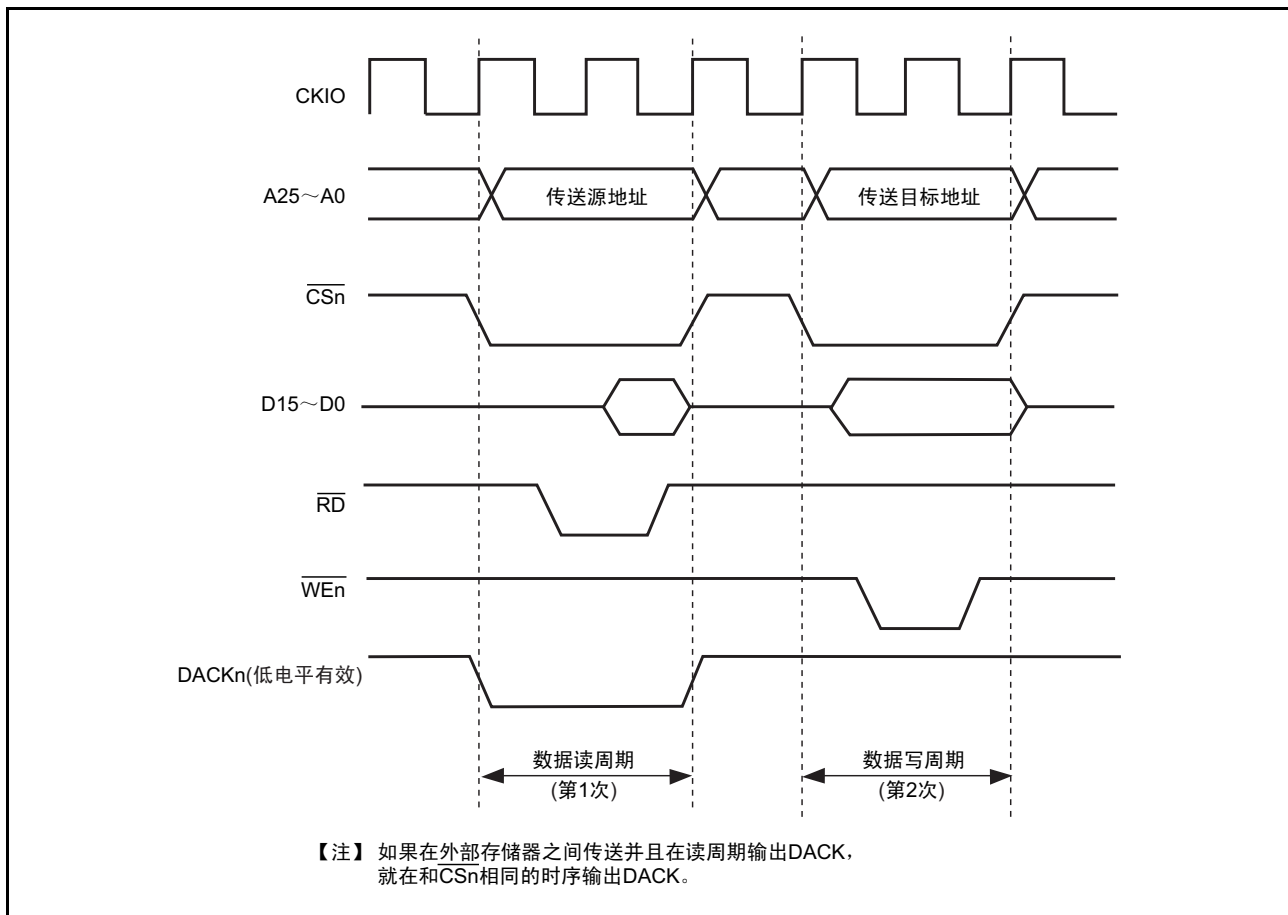


图 11.4 双地址模式中的 DMA 传送时序例子（传送源：正规存储器，传送目标：正规存储器）

(b) 单地址模式

单地址模式用于传送源和传送目标都是外部设备的情况，通过 DACK 信号存取（选择）一个模块而通过地址存取另一个模块。在此模式中，DMAC 在将传送请求的接收信号 DACK 输出到一个外部设备进行存取的同时，将地址输出到传送目标，通过 1 个总线周期进行 DMA 传送。例如，对于图 11.5 所示的外部存储器和带 DACK 的外部设备之间的传送，在外部设备将数据输出到数据总线的相同总线周期中，将该数据写到外部存储器。

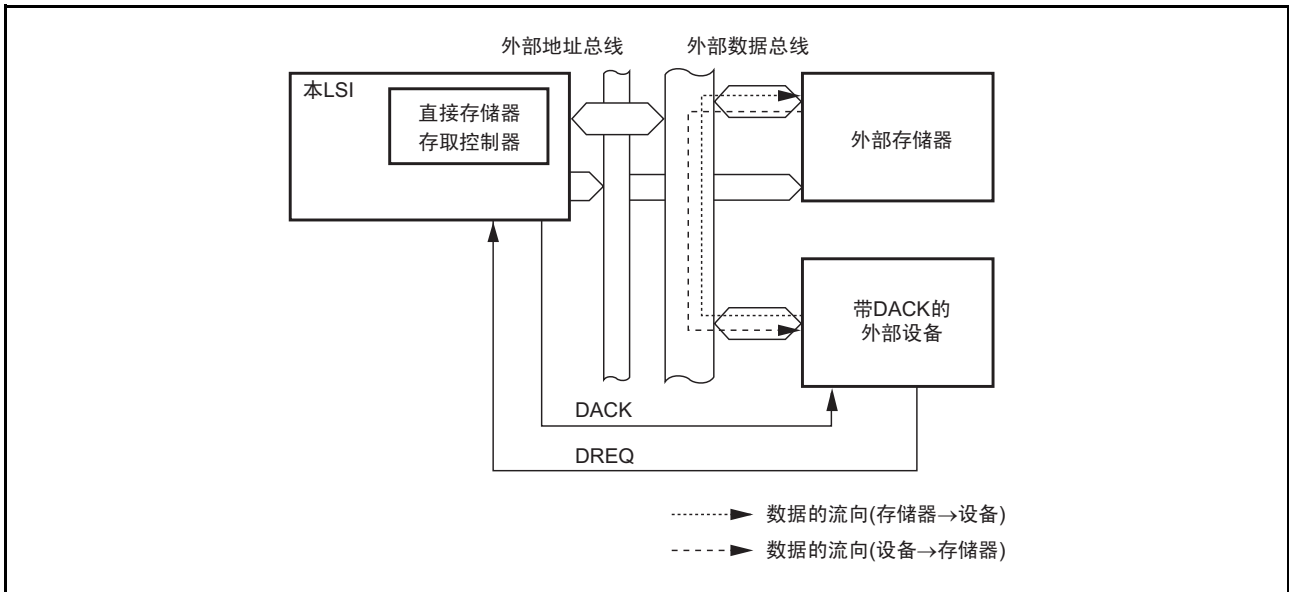


图 11.5 单地址模式的数据流程

能在单地址模式中进行的传送是：(1) 带 DACK 的外部设备和存储器映射外部设备之间的传送，(2) 带 DACK 的外部设备和外部存储器之间的传送。无论哪种情况，传送请求都只有外部请求（DREQ）。

单地址模式中的 DMA 传送时序例子如图 11.6 所示。

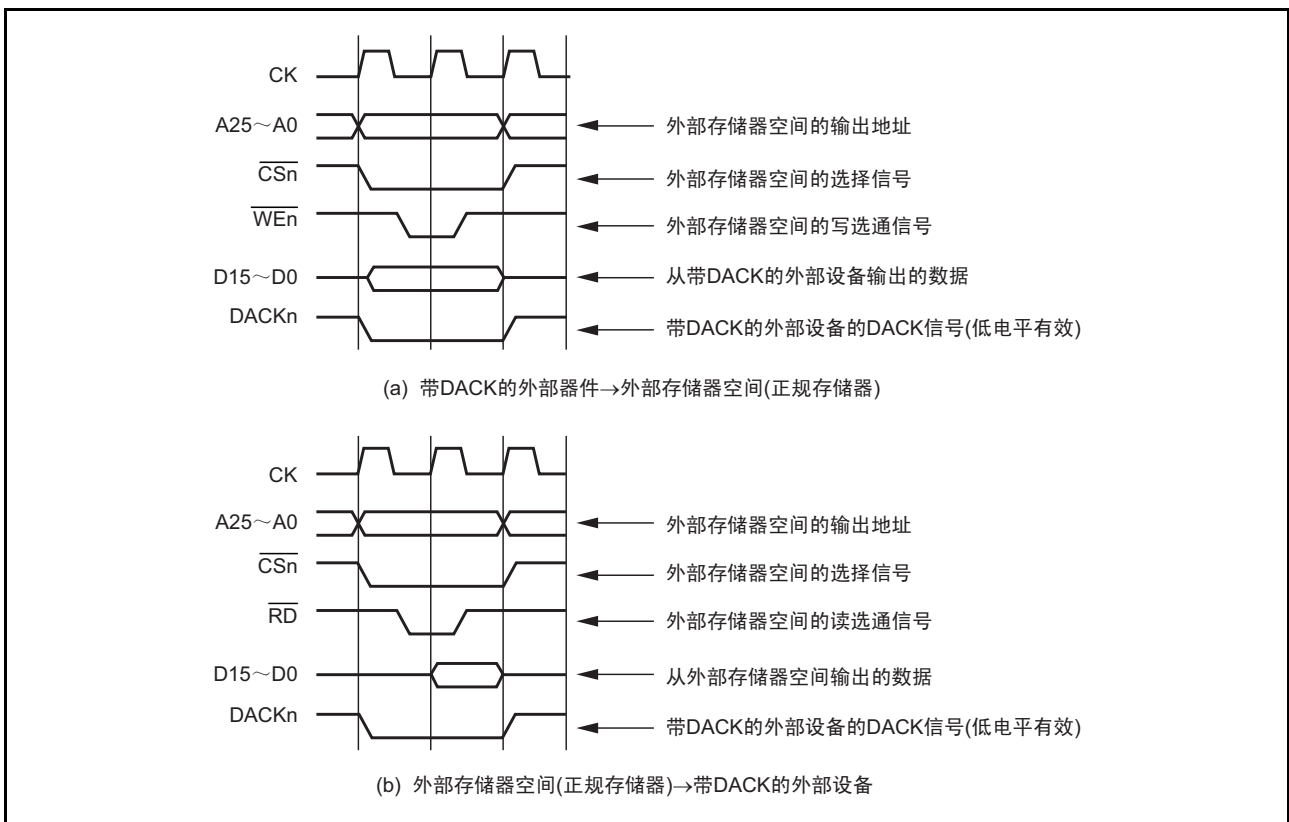


图 11.6 单地址模式中的 DMA 传送时序例子

(2) 总线模式

总线模式有周期挪用模式和突发模式，通过 CHCR 的 TB 位进行模式的选择。

(a) 周期挪用模式

• 正常模式

在周期挪用的正常模式中，每当1个传送单位（以字节、字、长字或者16字节为单位）的传送结束时，此模块就将总线权交给其他总线主控。如果此后发生传送请求，就从其他总线主控取回总线权，重新进行1个传送单位的传送，在该传送结束时又将总线权交给其他总线主控。重复此操作，直到满足传送结束条件为止。

周期挪用的正常模式与传送请求源、传送源和传送目标无关，能在全部传送区间内使用。

周期挪用正常模式中的DMA传送时序例子如图11.7所示，图中的传送条件如下：

- 双地址模式
- DREQ低电平检测

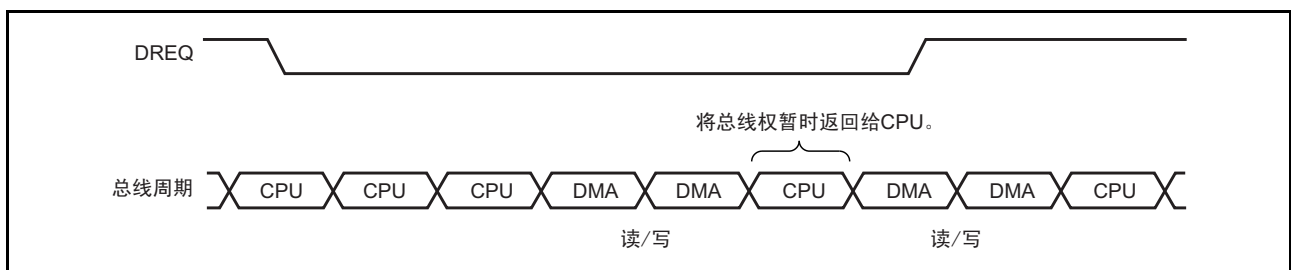


图 11.7 周期挪用正常模式中的 DMA 传送例子（双地址、DREQ 低电平检测）

• 间歇模式 16 和间歇模式 64

在周期挪用的间歇模式中，每当1个传送单位（以字节、字、长字或者16字节为单位）的传送结束时，此模块就将总线权交给其他总线主控。如果此后发生传送请求，就通过 $B\phi$ 时钟计数等待16个时钟或者64个时钟，然后从其他总线主控取回总线权，重新进行1个传送单位的传送，在该传送结束时又将总线权交给其他总线主控。重复此操作，直到满足结束条件为止。因此，和周期挪用的正常模式相比，能降低DMA传送的总线占有率。

在此模块重新取回总线权时，对于因高速缓存未命中而引起的入口更新等情况，可能会让DMA传送延迟。

间歇模式与传送请求源、传送源和传送目标无关，能在全部传送区间内使用，但是全部通道的总线模式都必须是周期挪用模式。

周期挪用间歇模式中的DMA传送时序例子如图11.8所示，图中的传送条件如下：

- 双地址模式
- DREQ低电平检测

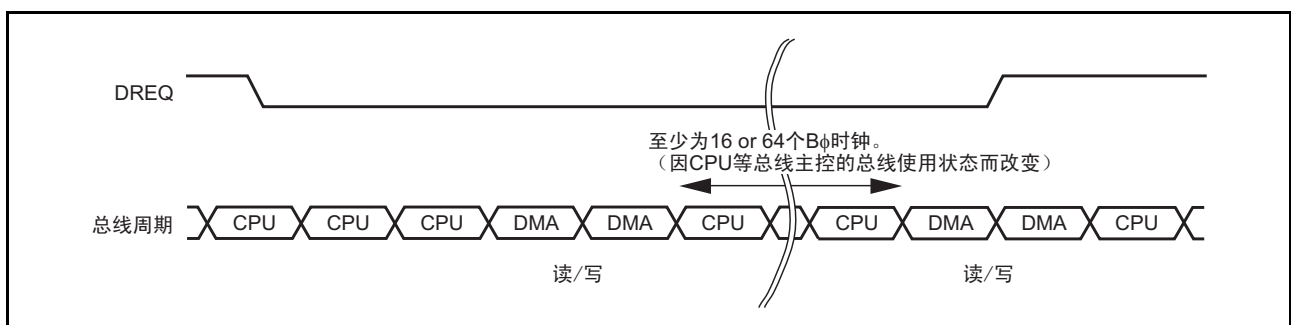


图 11.8 周期挪用间歇模式中的 DMA 传送例子（双地址、DREQ 低电平检测）

(b) 突发模式

在突发模式中，此模块一旦取得总线权，就在满足传送结束条件前不释放总线权而连续进行传送。但是，在外部请求模式中通过电平检测 DREQ 时，如果 DREQ 变为无效电平，即使尚未满足传送结束条件，也在结束已接受请求的 DMA 传送请求后将总线权交给其他总线主控。

突发模式中的 DMA 传送时序如图 11.9 所示。

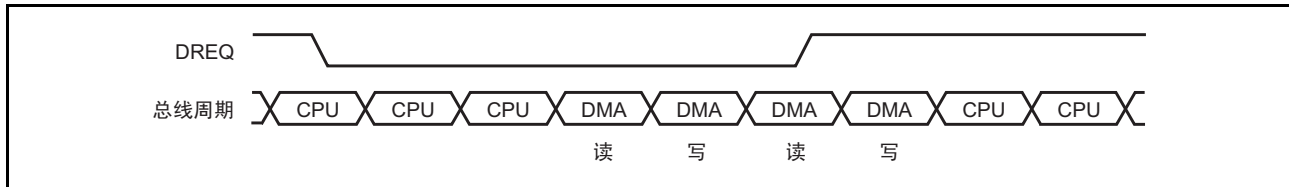


图 11.9 突发模式中的 DMA 传送例子（双地址、DREQ 低电平检测）

(3) DMA 传送区间和请求模式、总线模式的关系

DMA 传送区间和请求模式、总线模式等相关事项如表 11.9 所示。

表 11.9 DMA 传送区间和请求模式、总线模式等相关事项一览表

地址模式	传送区间	请求模式	总线模式	传送长度（位）	能使用的通道
双地址模式	带 DACK 的外部设备和外部存储器	外部	B/C	8/16/32/128	0
	带 DACK 的外部设备和存储器映射外部设备	外部	B/C	8/16/32/128	0
	外部存储器和外部存储器	均可 *4	B/C	8/16/32/128	0 ~ 15*3
	外部存储器和存储器映射外部设备	均可 *4	B/C	8/16/32/128	0 ~ 15*3
	存储器映射外部设备和存储器映射外部设备	均可 *4	B/C	8/16/32/128	0 ~ 15*3
	外部存储器和内部外围模块	均可 *1	B/C*5	8/16/32/128*2	0 ~ 15*3
	存储器映射外部设备和内部外围模块	均可 *1	B/C*5	8/16/32/128*2	0 ~ 15*3
	内部外围模块和内部外围模块	均可 *1	B/C*5	8/16/32/128*2	0 ~ 15*3
	内部存储器和内部存储器	均可 *4	B/C	8/16/32/128	0 ~ 15*3
	内部存储器和存储器映射外部设备	均可 *4	B/C	8/16/32/128	0 ~ 15*3
	内部存储器和内部外围模块	均可 *1	B/C*5	8/16/32/128*2	0 ~ 15*3
	内部存储器和外部存储器	均可 *4	B/C	8/16/32/128	0 ~ 15*3
单地址模式	带 DACK 的外部设备和外部存储器	外部	B/C	8/16/32/128	0
	带 DACK 的外部设备和存储器映射外部设备	外部	B/C	8/16/32/128	0

【符号说明】

B: 突发

C: 周期挪用

【注】 *1 外部请求、自动请求和内部外围模块请求均可。

但是，在内部外围模块请求的情况下，传送请求源除了多功能定时器脉冲单元 2 和比较匹配以外，传送源或者传送目标需要是为各自的请求源寄存器。

*2 这是通过传送源或者传送目标的内部外围模块寄存器允许的存取长度。

*3 在传送请求为外部请求时，只限于通道 0。

*4 外部请求、自动请求和内部外围模块请求均可。

但是，在内部外围模块请求的情况下，只限于多功能定时器脉冲单元 2 和比较匹配。

*5 在内部外围模块请求的情况下，传送请求源除了 CD-ROM 解码器、多功能定时器脉冲单元 2 和比较匹配以外，只限于周期挪用。

(4) 总线模式和通道优先级

在优先级固定模式 (CH0 > CH1) 中, 如果优先级更高的通道 0 发生传送请求, 即使通道 1 正在以突发模式进行传送, 也立即开始通道 0 的传送。

此时, 如果通道 0 也是突发模式, 就在优先级高的通道 0 的传送全部结束后通道 1 继续传送。

如果通道 0 是周期挪用模式, 首先在优先级高的通道 0 进行 1 个传送单位的传送, 然后不释放总线权而连续进行通道 1 的传送。此后, 按照通道 0→通道 1→通道 0→通道 1 进行交替传送, 即总线状态是在周期挪用模式的传送结束后 CPU 周期被替换为突发模式传送的状态 (以下, 称为突发模式的优先执行)。此例如图 11.10 所示。如果发生竞争的突发模式有多个通道, 就优先执行优先级最高的通道。

如果在多个通道进行 DMA 传送, 就不在发生竞争的突发传送全部结束前将总线权释放给总线主控。

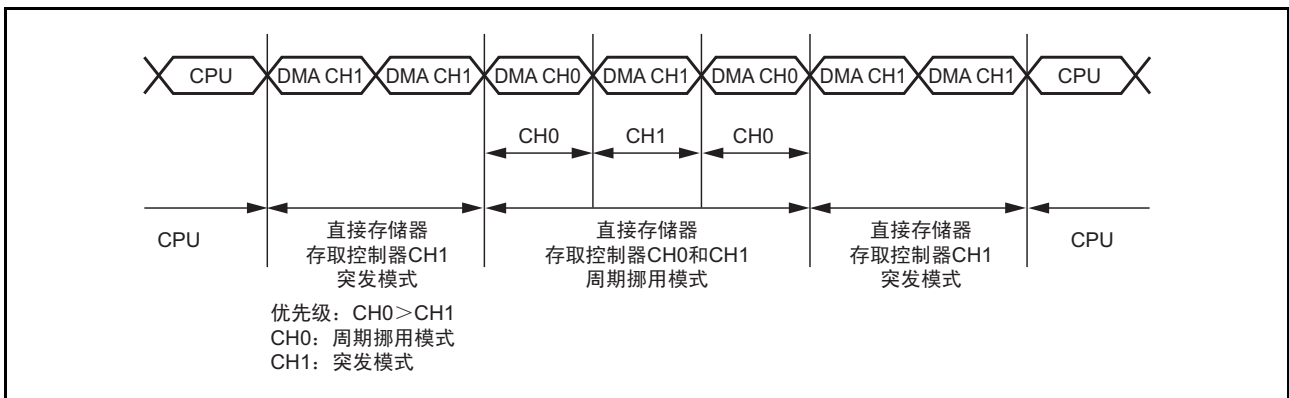


图 11.10 多个通道运行时的总线状态

11.4.5 总线周期的状态数和 DREQ 引脚的采样时序

(1) 总线周期的状态数

此模块为总线主控的情况和 CPU 为总线主控的情况一样, 通过总线状态控制器控制总线周期的状态数, 详细内容请参照“10. 总线状态控制器”。

(2) DREQ 引脚的采样时序

各总线模式中的 DREQ 输入的采样时序如图 11.11 ~ 图 11.14 所示。

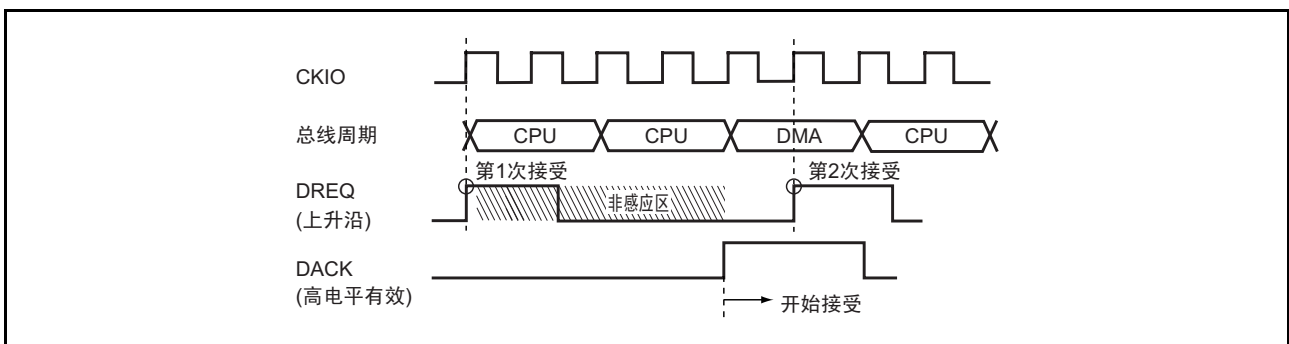


图 11.11 周期挪用模式、边沿检测时的 DREQ 输入检测时序

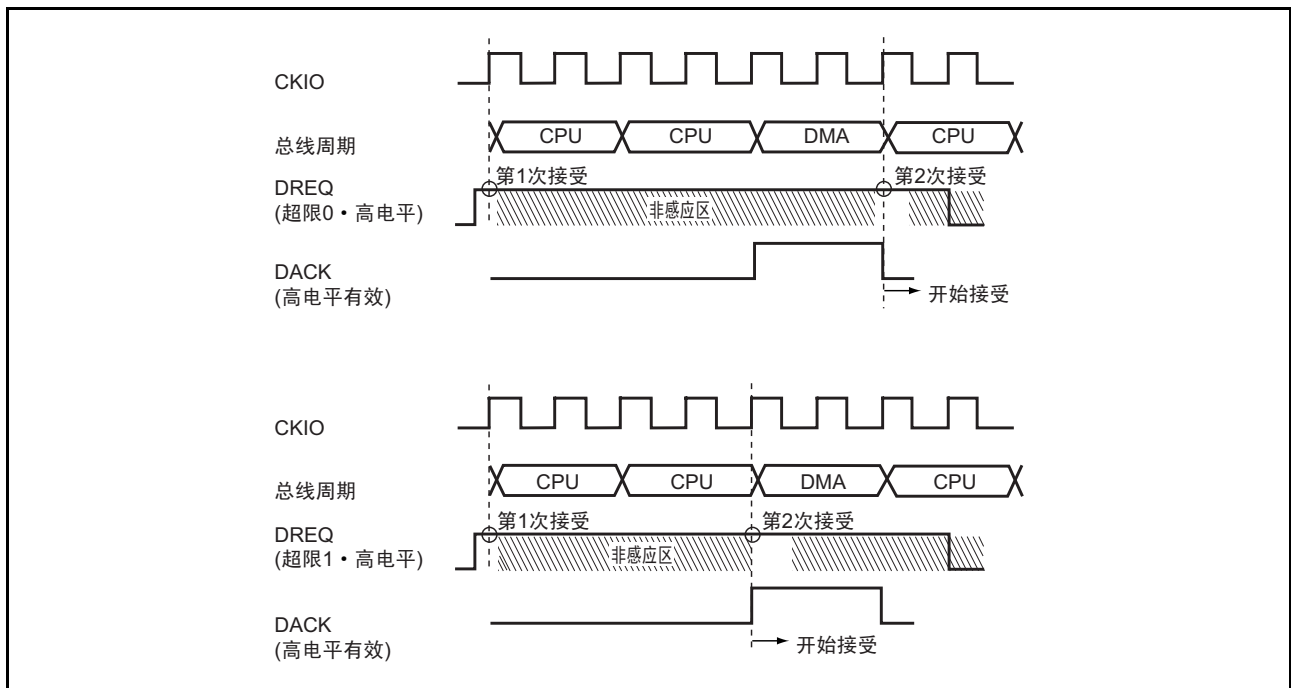


图 11.12 周期挪用模式、电平检测时的 DREQ 输入检测时序

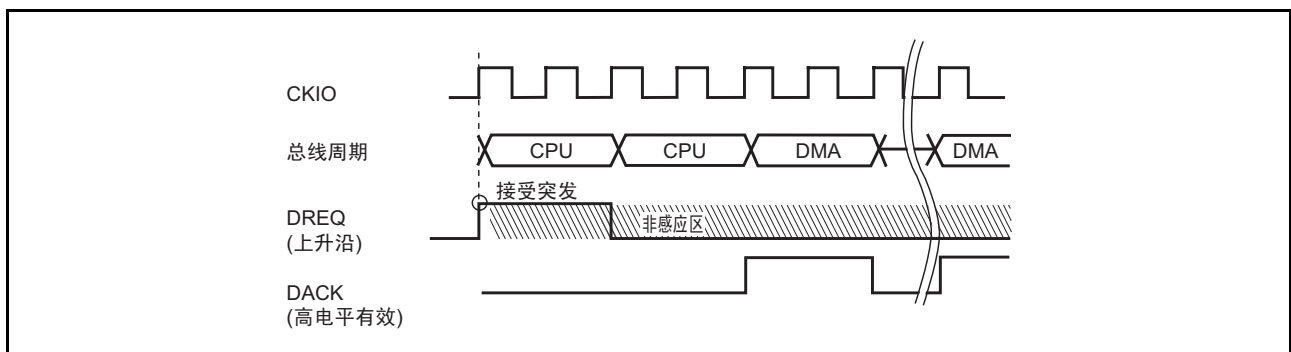


图 11.13 突发模式、边沿检测时的 DREQ 输入检测时序

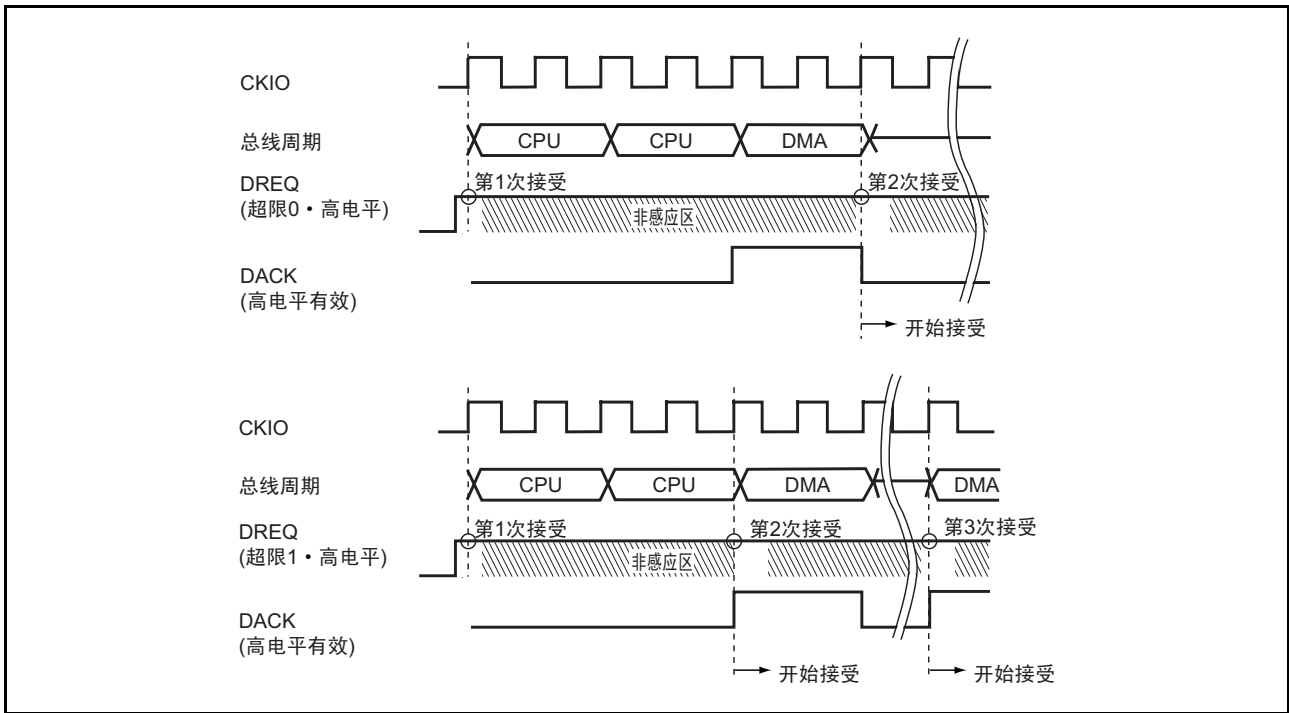


图 11.14 突发模式、电平检测时的 DREQ 输入检测时序

TEND 的输出时序如图 11.15 所示。

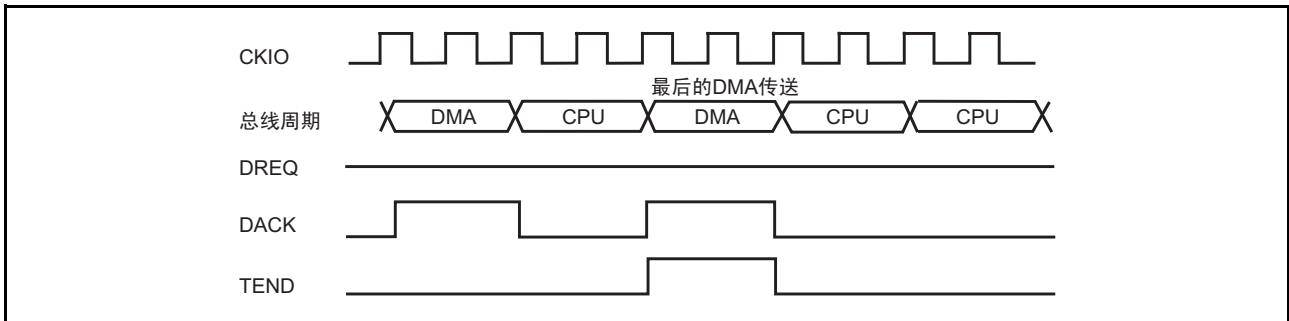


图 11.15 DMA 传送结束信号时序 (周期挪用、电平检测)

在对 8 位、16 位、32 位外部设备进行 16 字节的传送时，或者在对 8 位、16 位外部设备进行长字存取时，或者对 8 位外部设备进行字存取时，DMA 传送单位被分为多个总线周期。必须注意：在 DMA 传送单位被分为多个总线周期并且在总线周期之间将 \overline{CS} 置为无效时，为了调整数据，和 \overline{CS} 一样 DACK 输出和 TEND 输出被分割。此例如图 11.16 所示。另外，在 DMA 传送时 DACK 和 TEND 不被分割的情况如图 11.11 ~ 图 11.15 所示。

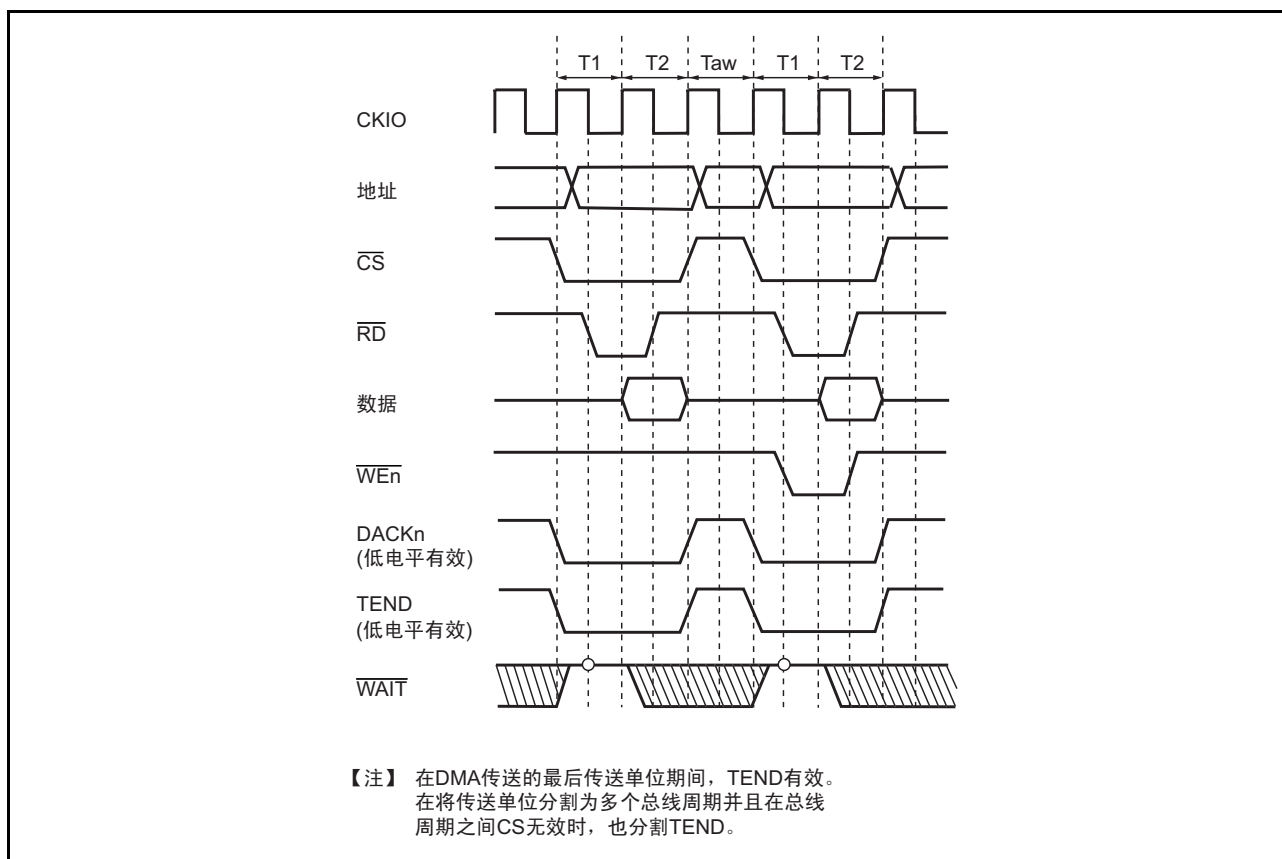


图 11.16 总线状态控制器的正规存储器存取（无等待、空闲周期为 1、16 位设备的长字存取）

11.5 使用时的注意事项

11.5.1 DACK 输出和 TEND 输出的时序

当外部存储器为 MPX-I/O 时，在数据周期的时序将 DACK 输出置为有效。详细内容参照“10. 总线状态控制器”的“10.5.5 MPX-I/O 接口”。

在 MPX-I/O 以外的存储器的情况下，DACK 输出的有效时序和相应的 CS 相同。

与存储器无关，TEND 输出的有效时序总是和相应的 CS 相同。

12. 多功能定时器脉冲单元 2

本 LSI 内置由 5 个通道的 16 位定时器构成的多功能定时器脉冲单元 2。

12.1 特点

- 能进行最多 16 个脉冲输入/输出。
- 各通道可选择 8 种计数器输入时钟。
- 能设定以下运行：通过比较匹配进行的波形输出、输入捕捉功能、计数器的清除运行、多个定时器计数器（TCNT）的同时写、通过比较匹配/输入捕捉进行的同时清除、通过计数器的同步运行进行各寄存器的同步输入/输出、通过和同步运行组合进行的最多 12 相的 PWM 输出。
- 通道 0、3、4 能设定缓冲运行。
- 通道 1 和通道 2 能分别设定相位计数模式。
- 级联运行
- 通过内部 16 位总线进行高速存取。
- 28 种中断源
- 能自动传送寄存器数据。
- 能生成 A/D 转换器的转换开始触发。
- 能设定模块待机模式。
- 能通过 CH3 和 CH4 的联动运行，设定互补 PWM 或者复位 PWM 的正负 3 相共 6 相的波形输出。
- 能通过 CH0、3、4 的联动运行，设定使用互补 PWM 或者复位 PWM 的 AC 同步马达（无刷 DC 马达）驱动模式，并且可选择 2 种（斩波和电平）波形输出。
- 在互补 PWM 模式中，能减少计数器波峰/波谷的中断和 A/D 转换器的转换开始触发。

表 12.1 多功能定时器脉冲单元 2 的功能一览表

项目	通道 0	通道 1	通道 2	通道 3	通道 4
计数时钟	P0φ/1 P0φ/4 P0φ/16 P0φ/64 TCLKA TCLKB TCLKC TCLKD	P0φ/1 P0φ/4 P0φ/16 P0φ/64 P0φ/256 TCLKA TCLKB	P0φ/1 P0φ/4 P0φ/16 P0φ/64 P0φ/1024 TCLKA TCLKB TCLKC	P0φ/1 P0φ/4 P0φ/16 P0φ/64 P0φ/256 P0φ/1024 TCLKA TCLKB	P0φ/1 P0φ/4 P0φ/16 P0φ/64 P0φ/256 P0φ/1024 TCLKA TCLKB
通用寄存器 (TGR)	TGRA_0 TGRB_0 TGRE_0	TGRA_1 TGRB_1	TGRA_2 TGRB_2	TGRA_3 TGRB_3	TGRA_4 TGRB_4
通用寄存器 / 缓冲寄存器	TGRC_0 TGRD_0 TGRF_0	—	—	TGRC_3 TGRD_3	TGRC_4 TGRD_4
输入 / 输出引脚	TIOC0A TIOC0B TIOC0C TIOC0D	TIOC1A TIOC1B	TIOC2A TIOC2B	TIOC3A TIOC3B TIOC3C TIOC3D	TIOC4A TIOC4B TIOC4C TIOC4D
计数器清除功能	TGR 的比较匹配 或者输入捕捉	TGR 的比较匹配 或者输入捕捉	TGR 的比较匹配 或者输入捕捉	TGR 的比较匹配 或者输入捕捉	TGR 的比较匹配 或者输入捕捉
比较匹配输出	输出低电平	○	○	○	○
	输出高电平	○	○	○	○
	交替输出	○	○	○	○
输入捕捉功能	○	○	○	○	○
同步运行	○	○	○	○	○
PWM 模式 1	○	○	○	○	○
PWM 模式 2	○	○	○	—	—
互补 PWM 模式	—	—	—	○	○
复位 PWM 模式	—	—	—	○	○
AC 同步马达驱动模式	○	—	—	○	○
相位计数模式	—	○	○	—	—
缓冲运行	○	—	—	○	○
直接存储器存取控制器的启动	TGR 的比较匹配 或者输入捕捉	TGR 的比较匹配 或者输入捕捉	TGR 的比较匹配 或者输入捕捉	TGR 的比较匹配 或者输入捕捉	TGR 的比较匹配 或者输入捕捉、 TCNT 上溢 / 下溢
A/D 转换开始触发	TGRA_0 的 比较匹配或者 输入捕捉、 TGRE_0 的 比较匹配	TGRA_1 的 比较匹配或者 输入捕捉	TGRA_2 的 比较匹配或者 输入捕捉	TGRA_3 的 比较匹配或者 输入捕捉	TGRA_4 的 比较匹配或者 输入捕捉、 互补 PWM 模式 中的 TCNT_4 下 溢 (波谷)

项目	通道 0	通道 1	通道 2	通道 3	通道 4
中断源	7 个源 • 比较匹配 / 输入捕捉 0A • 比较匹配 / 输入捕捉 0B • 比较匹配 / 输入捕捉 0C • 比较匹配 / 输入捕捉 0D • 比较匹配 0E • 比较匹配 0F • 上溢	4 个源 • 比较匹配 / 输入捕捉 1A • 比较匹配 / 输入捕捉 1B • 上溢 • 下溢	4 个源 • 比较匹配 / 输入捕捉 2A • 比较匹配 / 输入捕捉 2B • 上溢 • 下溢	5 个源 • 比较匹配 / 输入捕捉 3A • 比较匹配 / 输入捕捉 3B • 比较匹配 / 输入捕捉 3C • 比较匹配 / 输入捕捉 3D • 上溢	5 个源 • 比较匹配 / 输入捕捉 4A • 比较匹配 / 输入捕捉 4B • 比较匹配 / 输入捕捉 4C • 比较匹配 / 输入捕捉 4D • 上溢 / 下溢
A/D 转换的开始请求延迟功能	—	—	—	—	• 在 TADCORA_4 和 TCNT_4 匹配时请求开始 A/D 转换 • 在 TADCORB_4 和 TCNT_4 匹配时请求开始 A/D 转换
中断减少功能	—	—	—	• 减少 TGRA_3 比较匹配中断	• 减少 TCIV_4 中断

【符号说明】

- ：能
 —：不能

框图如图 12.1 所示。

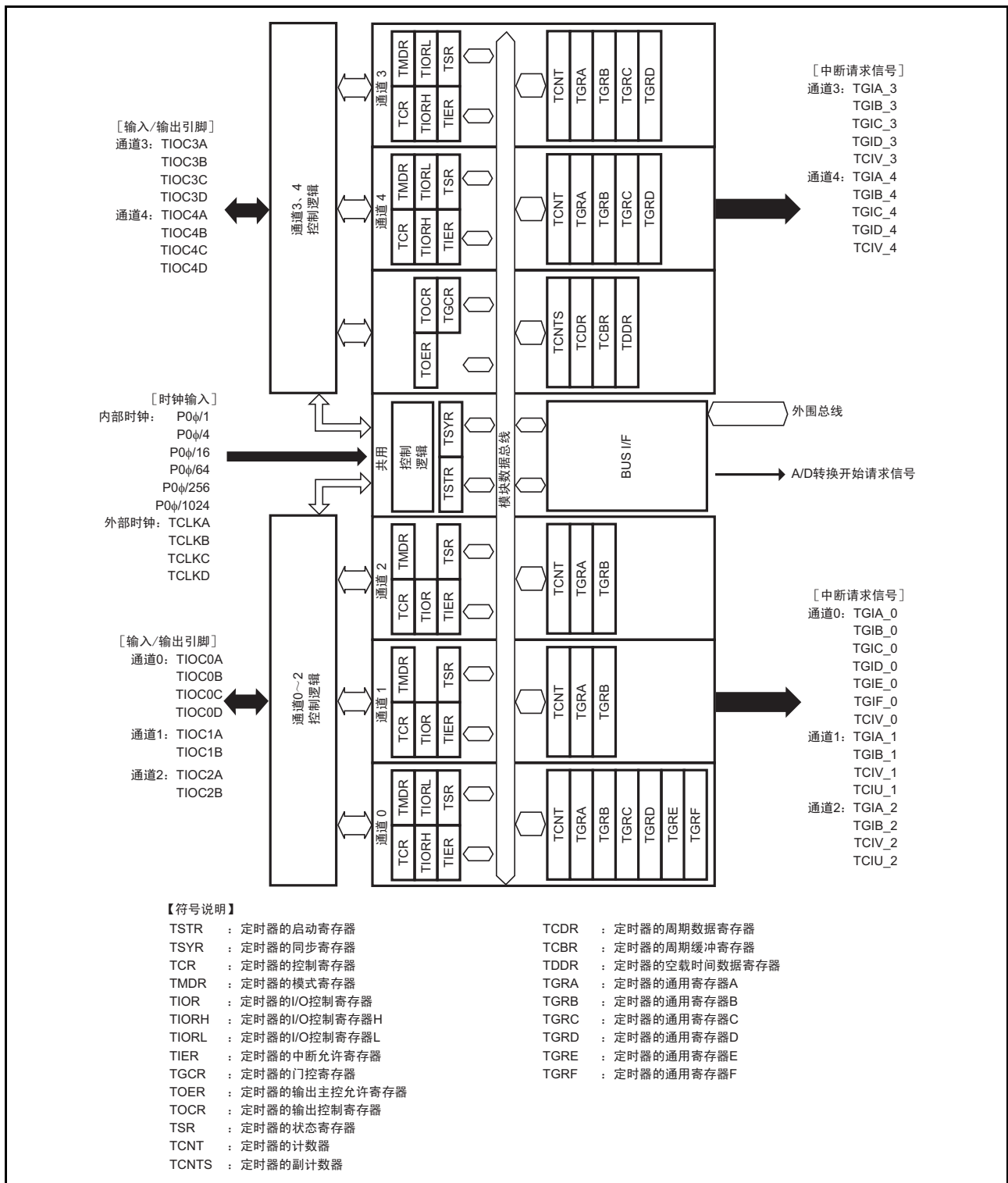


图 12.1 框图

12.2 输入 / 输出引脚

表 12.2 引脚构成

通道	引脚名	输入 / 输出	功能
共用	TCLKA	输入	外部时钟 A 的输入引脚 (通道 1 的相位计数模式的 A 相输入)
	TCLKB	输入	外部时钟 B 的输入引脚 (通道 1 的相位计数模式的 B 相输入)
	TCLKC	输入	外部时钟 C 的输入引脚 (通道 2 的相位计数模式的 A 相输入)
	TCLKD	输入	外部时钟 D 的输入引脚 (通道 2 的相位计数模式的 B 相输入)
0	TIOC0A	输入 / 输出	TGRA_0 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	TIOC0B	输入 / 输出	TGRB_0 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	TIOC0C	输入 / 输出	TGRC_0 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	TIOC0D	输入 / 输出	TGRD_0 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
1	TIOC1A	输入 / 输出	TGRA_1 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	TIOC1B	输入 / 输出	TGRB_1 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
2	TIOC2A	输入 / 输出	TGRA_2 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	TIOC2B	输入 / 输出	TGRB_2 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
3	TIOC3A	输入 / 输出	TGRA_3 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	TIOC3B	输入 / 输出	TGRB_3 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	TIOC3C	输入 / 输出	TGRC_3 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	TIOC3D	输入 / 输出	TGRD_3 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
4	TIOC4A	输入 / 输出	TGRA_4 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	TIOC4B	输入 / 输出	TGRB_4 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	TIOC4C	输入 / 输出	TGRC_4 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	TIOC4D	输入 / 输出	TGRD_4 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚

【注】 有关互补 PWM 模式的引脚结构, 请参照 “12.4.8 互补 PWM 模式” 的表 12.54。

12.3 寄存器说明

寄存器结构如表 12.3 所示。有关各通道的寄存器名，将通道 0 的 TCR 记为 TCR_0。

表 12.3 寄存器结构

通道	寄存器名	略称	R/W	初始值	地址	存取长度
0	定时器的控制寄存器 _0	TCR_0	R/W	H'00	H'FFFE4300	8
	定时器的模式寄存器 _0	TMDR_0	R/W	H'00	H'FFFE4301	8
	定时器的 I/O 控制寄存器 H_0	TIORH_0	R/W	H'00	H'FFFE4302	8
	定时器的 I/O 控制寄存器 L_0	TIORL_0	R/W	H'00	H'FFFE4303	8
	定时器的中断允许寄存器 _0	TIER_0	R/W	H'00	H'FFFE4304	8
	定时器的状态寄存器 _0	TSR_0	R/W	H'C0	H'FFFE4305	8
	定时器的计数器 _0	TCNT_0	R/W	H'0000	H'FFFE4306	16
	定时器的通用寄存器 A_0	TGRA_0	R/W	H'FFFF	H'FFFE4308	16
	定时器的通用寄存器 B_0	TGRB_0	R/W	H'FFFF	H'FFFE430A	16
	定时器的通用寄存器 C_0	TGRC_0	R/W	H'FFFF	H'FFFE430C	16
	定时器的通用寄存器 D_0	TGRD_0	R/W	H'FFFF	H'FFFE430E	16
	定时器的通用寄存器 E_0	TGRE_0	R/W	H'FFFF	H'FFFE4320	16
	定时器的通用寄存器 F_0	TGRF_0	R/W	H'FFFF	H'FFFE4322	16
	定时器的中断允许寄存器 2_0	TIER2_0	R/W	H'00	H'FFFE4324	8
	定时器的状态寄存器 2_0	TSR2_0	R/W	H'C0	H'FFFE4325	8
	定时器的缓冲运行传送模式寄存器 _0	TBTM_0	R/W	H'00	H'FFFE4326	8
1	定时器的控制寄存器 _1	TCR_1	R/W	H'00	H'FFFE4380	8
	定时器的模式寄存器 _1	TMDR_1	R/W	H'00	H'FFFE4381	8
	定时器的 I/O 控制寄存器 _1	TIOR_1	R/W	H'00	H'FFFE4382	8
	定时器的中断允许寄存器 _1	TIER_1	R/W	H'00	H'FFFE4384	8
	定时器的状态寄存器 _1	TSR_1	R/W	H'C0	H'FFFE4385	8
	定时器的计数器 _1	TCNT_1	R/W	H'0000	H'FFFE4386	16
	定时器的通用寄存器 A_1	TGRA_1	R/W	H'FFFF	H'FFFE4388	16
	定时器的通用寄存器 B_1	TGRB_1	R/W	H'FFFF	H'FFFE438A	16
	定时器的输入捕捉控制寄存器	TICCR	R/W	H'00	H'FFFE4390	8
2	定时器的控制寄存器 _2	TCR_2	R/W	H'00	H'FFFE4000	8
	定时器的模式寄存器 _2	TMDR_2	R/W	H'00	H'FFFE4001	8
	定时器的 I/O 控制寄存器 _2	TIOR_2	R/W	H'00	H'FFFE4002	8
	定时器的中断允许寄存器 _2	TIER_2	R/W	H'00	H'FFFE4004	8
	定时器的状态寄存器 _2	TSR_2	R/W	H'C0	H'FFFE4005	8
	定时器的计数器 _2	TCNT_2	R/W	H'0000	H'FFFE4006	16
	定时器的通用寄存器 A_2	TGRA_2	R/W	H'FFFF	H'FFFE4008	16
	定时器的通用寄存器 B_2	TGRB_2	R/W	H'FFFF	H'FFFE400A	16

通道	寄存器名	略称	R/W	初始值	地址	存取长度
3	定时器的控制寄存器 _3	TCR_3	R/W	H'00	H'FFFE4200	8
	定时器的模式寄存器 _3	TMDR_3	R/W	H'00	H'FFFE4202	8
	定时器的 I/O 控制寄存器 H_3	TIORH_3	R/W	H'00	H'FFFE4204	8
	定时器的 I/O 控制寄存器 L_3	TIORL_3	R/W	H'00	H'FFFE4205	8
	定时器的中断允许寄存器 _3	TIER_3	R/W	H'00	H'FFFE4208	8
	定时器的状态寄存器 _3	TSR_3	R/W	H'C0	H'FFFE422C	8
	定时器的计数器 _3	TCNT_3	R/W	H'0000	H'FFFE4210	16
	定时器的通用寄存器 A_3	TGRA_3	R/W	H'FFFF	H'FFFE4218	16
	定时器的通用寄存器 B_3	TGRB_3	R/W	H'FFFF	H'FFFE421A	16
	定时器的通用寄存器 C_3	TGRC_3	R/W	H'FFFF	H'FFFE4224	16
	定时器的通用寄存器 D_3	TGRD_3	R/W	H'FFFF	H'FFFE4226	16
	定时器的缓冲运行传送模式寄存器 _3	TBTM_3	R/W	H'00	H'FFFE4238	8
4	定时器的控制寄存器 _4	TCR_4	R/W	H'00	H'FFFE4201	8
	定时器的模式寄存器 _4	TMDR_4	R/W	H'00	H'FFFE4203	8
	定时器的 I/O 控制寄存器 H_4	TIORH_4	R/W	H'00	H'FFFE4206	8
	定时器的 I/O 控制寄存器 L_4	TIORL_4	R/W	H'00	H'FFFE4207	8
	定时器的中断允许寄存器 _4	TIER_4	R/W	H'00	H'FFFE4209	8
	定时器的状态寄存器 _4	TSR_4	R/W	H'C0	H'FFFE422D	8
	定时器的计数器 _4	TCNT_4	R/W	H'0000	H'FFFE4212	16
	定时器的通用寄存器 A_4	TGRA_4	R/W	H'FFFF	H'FFFE421C	16
	定时器的通用寄存器 B_4	TGRB_4	R/W	H'FFFF	H'FFFE421E	16
	定时器的通用寄存器 C_4	TGRC_4	R/W	H'FFFF	H'FFFE4228	16
	定时器的通用寄存器 D_4	TGRD_4	R/W	H'FFFF	H'FFFE422A	16
	定时器的缓冲运行传送模式寄存器 _4	TBTM_4	R/W	H'00	H'FFFE4239	8
	定时器的 A/D 转换开始请求控制寄存器	TADCR	R/W	H'0000	H'FFFE4240	16
	定时器的 A/D 转换开始请求周期设定寄存器 A_4	TADCORA_4	R/W	H'FFFF	H'FFFE4244	16
	定时器的 A/D 转换开始请求周期设定寄存器 B_4	TADCORB_4	R/W	H'FFFF	H'FFFE4246	16
	定时器的 A/D 转换开始请求周期设定缓冲寄存器 A_4	TADCOBRA_4	R/W	H'FFFF	H'FFFE4248	16
定时器的 A/D 转换开始请求周期设定缓冲寄存器 B_4	TADCOBRB_4	R/W	H'FFFF	H'FFFE424A	16	
共用	定时器的启动寄存器	TSTR	R/W	H'00	H'FFFE4280	8
	定时器的同步寄存器	TSYR	R/W	H'00	H'FFFE4281	8
	定时器的读写允许寄存器	TRWER	R/W	H'01	H'FFFE4284	8

通道	寄存器名	略称	R/W	初始值	地址	存取长度
3/4 共用	定时器的输出主控允许寄存器	TOER	R/W	H'00	H'FFFE420A	8
	定时器的输出控制寄存器 1	TOCR1	R/W	H'00	H'FFFE420E	8
	定时器的输出控制寄存器 2	TOCR2	R/W	H'00	H'FFFE420F	8
	定时器的门控寄存器	TGCR	R/W	H80	H'FFFE420D	8
	定时器的周期数据寄存器	TCDR	R/W	H'FFFF	H'FFFE4214	16
	定时器的空载时间数据寄存器	TDDR	R/W	H'FFFF	H'FFFE4216	16
	定时器的副计数器	TCNTS	R	H'0000	H'FFFE4220	16
	定时器的周期缓冲寄存器	TGBR	R/W	H'FFFF	H'FFFE4222	16
	定时器的中断减少设定寄存器	TITCR	R/W	H'00	H'FFFE4230	8
	定时器的中断减少次数计数器	TITCNT	R	H'00	H'FFFE4231	8
	定时器的缓冲传送设定寄存器	TBTER	R/W	H'00	H'FFFE4232	8
	定时器的空载时间允许寄存器	TDER	R/W	H'01	H'FFFE4234	8
	定时器的波形控制寄存器	TWCR	R/W	H'00	H'FFFE4260	8
	定时器的输出电平缓冲寄存器	TOLBR	R/W	H'00	H'FFFE4236	8

12.3.1 定时器的控制寄存器 (TCR)

TCR 是控制各通道 TCNT 的 8 位可读写寄存器。在此模块中，通道 0 ~ 4 各有 1 个，共计 5 个 TCR。必须在 TCNT 停止运行的状态下设定 TCR。

位:	7	6	5	4	3	2	1	0
	CCLR[2:0]			CKEG[1:0]		TPSC[2:0]		
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 5	CCLR[2:0]	000	R/W	计数器清除 2、1、0 选择 TCNT 计数器的清除源，详细内容请参照表 12.4 和表 12.5。
4、3	CKEG[1:0]	00	R/W	时钟边沿 1、0 选择输入时钟的边沿。如果在双边沿对内部时钟进行计数，输入时钟的周期就为 1/2（例如：P0φ/4 的双边沿 = P0φ/2 的上升沿）。在通道 1 和通道 2 使用相位计数模式时，忽视此设定而优先设定相位计数模式。内部时钟的边沿选择在输入时钟为 P0φ/4 或者慢于 P0φ/4 时有效。如果选择 P0φ/1 或者其他通道的上溢 / 下溢作为输入时钟，虽然能写值，但是在运行时为初始值。 00: 在上升沿进行计数 01: 在下降沿进行计数 1x: 在双边沿进行计数
2 ~ 0	TPSC[2:0]	000	R/W	定时器的预分频器 2、1、0 选择 TCNT 计数器的时钟。各通道能独立选择时钟源，详细内容请参照表 12.6 ~ 表 12.9。

【符号说明】 x: Don't care

表 12.4 CCLR2 ~ CCLR0 (通道 0、3、4)

通道	bit7	bit6	bit5	说明
	CCLR2	CCLR1	CCLR0	
0、3、4	0	0	0	禁止清除 TCNT。
	0	0	1	在发生 TGRA 的比较匹配 / 输入捕捉时清除 TCNT。
	0	1	0	在发生 TGRB 的比较匹配 / 输入捕捉时清除 TCNT。
	0	1	1	在清除正在进行同步清除或者同步运行的其他通道的计数器时清除 TCNT*1。
	1	0	0	禁止清除 TCNT。
	1	0	1	在发生 TGRC 的比较匹配 / 输入捕捉时清除 TCNT*2。
	1	1	0	在发生 TGRD 的比较匹配 / 输入捕捉时清除 TCNT*2。
	1	1	1	在清除正在进行同步清除或者同步运行的其他通道的计数器时清除 TCNT*1。

【注】 *1 通过将 TSYR 的 SYNC 位置 “1”，设定为同步运行。

*2 在将 TGRC 或者 TGRD 用作缓冲寄存器时，因为优先设定缓冲寄存器而且不发生比较匹配 / 输入捕捉，所以不清除 TCNT。

表 12.5 CCLR2 ~ CCLR0 (通道 1 和通道 2)

通道	bit7	bit6	bit5	说明
	保留 *2	CCLR1	CCLR0	
1、2	0	0	0	禁止清除 TCNT。
	0	0	1	在发生 TGRA 的比较匹配 / 输入捕捉时清除 TCNT。
	0	1	0	在发生 TGRB 的比较匹配 / 输入捕捉时清除 TCNT。
	0	1	1	在清除正在进行同步清除或者同步运行的其他通道的计数器时清除 TCNT*1。

【注】 *1 通过将 TSYR 的 SYNC 位置 “1”，设定为同步运行。

*2 在通道 1 和通道 2 中，bit7 为保留位，读取值总是 “0”，写操作无效。

表 12.6 TPSC2 ~ TPSC0 (通道 0)

通道	bit2	bit1	bit0	说明
	TPSC2	TPSC1	TPSC0	
0	0	0	0	内部时钟：通过 P0φ/1 进行计数
	0	0	1	内部时钟：通过 P0φ/4 进行计数
	0	1	0	内部时钟：通过 P0φ/16 进行计数
	0	1	1	内部时钟：通过 P0φ/64 进行计数
	1	0	0	外部时钟：通过 TCLKA 引脚输入进行计数
	1	0	1	外部时钟：通过 TCLKB 引脚输入进行计数
	1	1	0	外部时钟：通过 TCLKC 引脚输入进行计数
	1	1	1	外部时钟：通过 TCLKD 引脚输入进行计数

表 12.7 TPSC2 ~ TPSC0 (通道 1)

通道	bit2	bit1	bit0	说明
	TPSC2	TPSC1	TPSC0	
1	0	0	0	内部时钟: 通过 P0 ϕ /1 进行计数
	0	0	1	内部时钟: 通过 P0 ϕ /4 进行计数
	0	1	0	内部时钟: 通过 P0 ϕ /16 进行计数
	0	1	1	内部时钟: 通过 P0 ϕ /64 进行计数
	1	0	0	外部时钟: 通过 TCLKA 引脚输入进行计数
	1	0	1	外部时钟: 通过 TCLKB 引脚输入进行计数
	1	1	0	内部时钟: 通过 P0 ϕ /256 进行计数
	1	1	1	通过 TCNT_2 的上溢 / 下溢进行计数

【注】 在通道 1 为相位计数模式时, 此设定无效。

表 12.8 TPSC2 ~ TPSC0 (通道 2)

通道	bit2	bit1	bit0	说明
	TPSC2	TPSC1	TPSC0	
2	0	0	0	内部时钟: 通过 P0 ϕ /1 进行计数
	0	0	1	内部时钟: 通过 P0 ϕ /4 进行计数
	0	1	0	内部时钟: 通过 P0 ϕ /16 进行计数
	0	1	1	内部时钟: 通过 P0 ϕ /64 进行计数
	1	0	0	外部时钟: 通过 TCLKA 引脚输入进行计数
	1	0	1	外部时钟: 通过 TCLKB 引脚输入进行计数
	1	1	0	外部时钟: 通过 TCLKC 引脚输入进行计数
	1	1	1	内部时钟: 通过 P0 ϕ /1024 进行计数

【注】 在通道 2 为相位计数模式时, 此设定无效。

表 12.9 TPSC2 ~ TPSC0 (通道 3 和通道 4)

通道	bit2	bit1	bit0	说明
	TPSC2	TPSC1	TPSC0	
3、4	0	0	0	内部时钟: 通过 P0 ϕ /1 进行计数
	0	0	1	内部时钟: 通过 P0 ϕ /4 进行计数
	0	1	0	内部时钟: 通过 P0 ϕ /16 进行计数
	0	1	1	内部时钟: 通过 P0 ϕ /64 进行计数
	1	0	0	内部时钟: 通过 P0 ϕ /256 进行计数
	1	0	1	内部时钟: 通过 P0 ϕ /1024 进行计数
	1	1	0	外部时钟: 通过 TCLKA 引脚输入进行计数
	1	1	1	外部时钟: 通过 TCLKB 引脚输入进行计数

12.3.2 定时器的模式寄存器 (TMDR)

TMDR 是 8 位可读寄存器，设定各通道的运行模式。在此模块中，通道 0 ~ 4 各有 1 个，共计 5 个 TMDR。必须在 TCNT 停止运行的状态下设定 TMDR。

位:	7	6	5	4	3	2	1	0
	—	BFE	BFB	BFA	MD[3:0]			
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	—	0	R	保留位 读写值都为“0”。
6	BFE	0	R/W	缓冲运行 E 选择 TGRE_0 和 TGRF_0 是进行正常运行还是进行缓冲运行。即使将 TGRF 用作缓冲寄存器，也会发生 TGRF 的比较匹配。在通道 1、2、3、4 中，此位为保留位，读写值都为“0”。 0: TGRE_0 和 TGRF_0 进行正常运行 1: TGRE_0 和 TGRF_0 进行缓冲运行
5	BFB	0	R/W	缓冲运行 B 设定 TGRB 是进行正常运行还是 TGRB 和 TGRD 组合进行缓冲运行。如果将 TGRD 用作缓冲寄存器，除了互补 PWM 模式以外，不发生 TGRD 的输入捕捉 / 输出比较。在没有 TGRD 的通道 1、2 中，此位为保留位，读写值都为“0”。 0: TGRB 和 TGRD 进行正常运行 1: TGRB 和 TGRD 进行缓冲运行
4	BFA	0	R/W	缓冲运行 A 设定 TGRA 是进行正常运行还是 TGRA 和 TGRC 组合进行缓冲运行。如果将 TGRC 用作缓冲寄存器，除了互补 PWM 模式以外，不发生 TGRC 的输入捕捉 / 输出比较。但是，在互补 PWM 模式中发生 TGRC 的比较匹配。如果在互补 PWM 模式的 Tb 区间发生通道 4 的比较匹配，TGFC 就被置位，因此必须将定时器的中断允许寄存器_4 (TIER_4) 的 TGIEC 位置“0”。 在没有 TGRC 的通道 1、2 中，此位为保留位，读写值都为“0”。 0: TGRA 和 TGRC 进行正常运行 1: TGRA 和 TGRC 进行缓冲运行
3 ~ 0	MD[3:0]	0000	R/W	模式 3 ~ 0 MD3 ~ MD0 设定定时器的运行模式。 详细内容请参照表 12.10。

表 12.10 通过 MD3 ~ MD0 位设定的运行模式

bit3	bit2	bit1	bit0	说明
MD3	MD2	MD1	MD0	
0	0	0	0	正常运行。
0	0	0	1	禁止设定。
0	0	1	0	PWM 模式 1
0	0	1	1	PWM 模式 2*1
0	1	0	0	相位计数模式 1*2
0	1	0	1	相位计数模式 2*2
0	1	1	0	相位计数模式 3*2
0	1	1	1	相位计数模式 4*2
1	0	0	0	复位同步 PWM 模式 *3
1	0	0	1	禁止设定。
1	0	1	x	禁止设定。
1	1	0	0	禁止设定。
1	1	0	1	互补 PWM 模式 1（在波峰进行传送）*3
1	1	1	0	互补 PWM 模式 2（在波谷进行传送）*3
1	1	1	1	互补 PWM 模式 3（在波峰和波谷进行传送）*3

【符号说明】 x: Don't care

【注】 *1 在通道 3 和通道 4 中，不能设定 PWM 模式 2。

*2 在通道 0、3、4 中，不能设定相位计数模式。

*3 只能在通道 3 中设定复位同步 PWM 模式或者互补 PWM 模式。

如果将通道 3 设定为复位同步 PWM 模式或者互补 PWM 模式，通道 4 的设定就无效并且自动服从通道 3 的设定。

但是，不能将通道 4 设定为复位同步 PWM 模式或者互补 PWM 模式。

在通道 0、1、2 中，不能设定复位同步 PWM 模式或者互补 PWM 模式。

12.3.3 定时器的 I/O 控制寄存器 (TIOR)

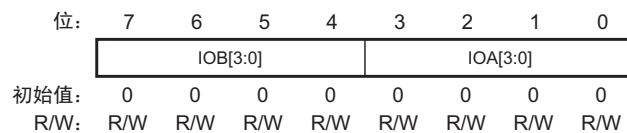
TIOR 是控制 TGR 的 8 位可读写寄存器。在此模块中，通道 0、3、4 各有 2 个，通道 1 和通道 2 各有 1 个，共计 8 个 TIOR。

在 TMDR 为正常运行、PWM 模式或者相位计数模式时，设定 TIOR。

TIOR 指定的初始输出在计数器停止（将 TSTR 的 CST 位清“0”）的状态下有效。在 PWM 模式 2 中，指定计数器清“0”时的输出。

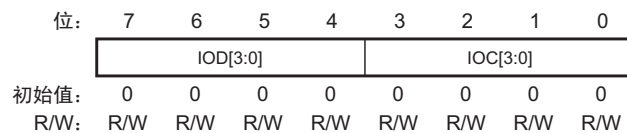
如果将 TGRC 或者 TGRD 设定为缓冲运行，此设定就无效并且作为缓冲寄存器运行。

- TIORH_0、TIOR_1、TIOR_2、TIORH_3、TIORH_4



位	位名	初始值	R/W	说明
7 ~ 4	IOB[3:0]	0000	R/W	I/O 控制 B3 ~ B0 IOB3 ~ IOB0 位设定 TGRB 的功能。 请参照下表： TIORH_0: 表 12.11 TIOR_1: 表 12.13 TIOR_2: 表 12.14 TIORH_3: 表 12.15 TIORH_4: 表 12.17
3 ~ 0	IOA[3:0]	0000	R/W	I/O 控制 A3 ~ A0 IOA3 ~ IOA0 位设定 TGRA 的功能。 请参照下表： TIORH_0: 表 12.19 TIOR_1: 表 12.21 TIOR_2: 表 12.22 TIORH_3: 表 12.23 TIORH_4: 表 12.25

- TIORL_0、TIORL_3、TIORL_4



位	位名	初始值	R/W	说明
7 ~ 4	IOD[3:0]	0000	R/W	I/O 控制 D3 ~ D0 IOD3 ~ IOD0 位设定 TGRD 的功能。 请参照下表： TIORL_0: 表 12.12 TIORL_3: 表 12.16 TIORL_4: 表 12.18
3 ~ 0	IOC[3:0]	0000	R/W	I/O 控制 C3 ~ C0 IOC3 ~ IOC0 位设定 TGRC 的功能。 请参照下表： TIORL_0: 表 12.20 TIORL_3: 表 12.24 TIORL_4: 表 12.26

表 12.11 TIORH_0 (通道 0)

bit7	bit6	bit5	bit4	说明	
IOB3	IOB2	IOB1	IOB0	TGRB_0 的功能	TIOC0B 引脚的功能
0	0	0	0	输出比较寄存器	保持输出。*
0	0	0	1		初始输出为 Low 电平，在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平，在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平，在比较匹配时进行交替输出。
0	1	0	0		保持输出。
0	1	0	1		初始输出为 High 电平，在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平，在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平，在比较匹配时进行交替输出。
1	0	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	0	0	1		在下降沿进行输入捕捉。
1	0	1	x		在双边沿进行输入捕捉。
1	1	x	x		捕捉输入源是通道 1/ 计数时钟。 在 TCNT_1 进行递增或者递减计数时进行输入捕捉。

【符号说明】 x: Don't care

【注】 * 从上电复位后到设定 TIOR 为止，输出 Low 电平。

表 12.12 TIORL_0 (通道 0)

bit7	bit6	bit5	bit4	说明	
IOD3	IOD2	IOD1	IOD0	TGRD_0 的功能	TIOC0D 引脚的功能
0	0	0	0	输出比较寄存器 *2	保持输出 *1。
0	0	0	1		初始输出为 Low 电平，在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平，在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平，在比较匹配时进行交替输出。
0	1	0	0		保持输出。
0	1	0	1		初始输出为 High 电平，在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平，在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平，在比较匹配时进行交替输出。
1	0	0	0	输入捕捉寄存器 *2	在上升沿进行输入捕捉。
1	0	0	1		在下降沿进行输入捕捉。
1	0	1	x		在双边沿进行输入捕捉。
1	1	x	x		捕捉输入源是通道 1/ 计数时钟。 在 TCNT_1 进行递增或者递减计数时进行输入捕捉。

【符号说明】 x: Don't care

【注】 *1 从上电复位后到设定 TIOR 为止，输出 Low 电平。

*2 如果将 TMDR_0 的 BFB 位置“1”使 TGRD_0 用作缓冲寄存器，此设定就无效并且不发生输入捕捉/输出比较。

表 12.13 TIOR_1 (通道 1)

bit7	bit6	bit5	bit4	说明	
IOB3	IOB2	IOB1	IOB0	TGRB_1 的功能	TIOC1B 引脚的功能
0	0	0	0	TGRB_1 为 输出比较寄存器	保持输出。*
0	0	0	1		初始输出为 Low 电平，在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平，在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平，在比较匹配进行交替输出。
0	1	0	0		保持输出。
0	1	0	1		初始输出为 High 电平，在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平，在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平，在比较匹配进行交替输出。
1	0	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	0	0	1		在下降沿进行输入捕捉。
1	0	1	x		在双边沿进行输入捕捉。
1	1	x	x		在发生 TGRC_0 的比较匹配 / 输入捕捉时进行输入捕捉。

【符号说明】 x: Don't care

【注】 * 从上电复位后到设定 TIOR 为止，输出 Low 电平。

表 12.14 TIOR_2 (通道 2)

bit7	bit6	bit5	bit4	说明	
IOB3	IOB2	IOB1	IOB0	TGRB_2 的功能	TIOC2B 引脚的功能
0	0	0	0	TGRB_2 为 输出比较寄存器	保持输出。*
0	0	0	1		初始输出为 Low 电平，在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平，在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平，在比较匹配时进行交替输出。
0	1	0	0		保持输出。
0	1	0	1		初始输出为 High 电平，在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平，在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平，在比较匹配时进行交替输出。
1	x	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

【符号说明】 x: Don't care

【注】 * 从上电复位后到设定 TIOR 为止，输出 Low 电平。

表 12.15 TIORH_3 (通道 3)

bit7	bit6	bit5	bit4	说明	
IOB3	IOB2	IOB1	IOB0	TGRB_3 的功能	TIOC3B 引脚的功能
0	0	0	0	输出比较寄存器	保持输出。*
0	0	0	1		初始输出为 Low 电平，在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平，在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平，在比较匹配时进行交替输出。
0	1	0	0		保持输出。
0	1	0	1		初始输出为 High 电平，在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平，在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平，在比较匹配进行交替输出。
1	x	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

【符号说明】 x: Don't care

【注】 * 从上电复位后到设定 TIOR 为止，输出 Low 电平。

表 12.16 TIORL_3 (通道 3)

bit7	bit6	bit5	bit4	说明	
IOD3	IOD2	IOD1	IOD0	TGRD_3 的功能	TIOC3D 引脚的功能
0	0	0	0	输出比较寄存器 *2	保持输出。*1
0	0	0	1		初始输出为 Low 电平，在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平，在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平，在比较匹配时进行交替输出。
0	1	0	0		保持输出。
0	1	0	1		初始输出为 High 电平，在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平，在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平，在比较匹配时进行交替输出。
1	x	0	0	输入捕捉寄存器 *2	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

【符号说明】 x: Don't care

【注】 *1 从上电复位后到设定 TIOR 为止，输出 Low 电平。

*2 如果将 TMDR_3 的 BFB 位置“1”使 TGRD_3 用作缓冲寄存器，此设定就无效并且不发生输入捕捉/输出比较。

表 12.17 TIORH_4 (通道 4)

bit7	bit6	bit5	bit4	说明	
IOB3	IOB2	IOB1	IOB0	TGRB_4 的功能	TIOC4B 引脚的功能
0	0	0	0	TGRB_4 为 输出比较寄存器	保持输出。*
0	0	0	1		初始输出为 Low 电平，在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平，在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平，在比较匹配时进行交替输出。
0	1	0	0		保持输出。
0	1	0	1		初始输出为 High 电平，在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平，在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平，在比较匹配时进行交替输出。
1	x	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

【符号说明】 x: Don't care

【注】 * 从上电复位后到设定 TIOR 为止，输出 Low 电平。

表 12.18 TIORL_4 (通道 4)

bit7	bit6	bit5	bit4	说明	
IOD3	IOD2	IOD1	IOD0	TGRD_4 的功能	TIOC4D 引脚的功能
0	0	0	0	输出比较寄存器 *2	保持输出。*1
0	0	0	1		初始输出为 Low 电平，在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平，在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平，在比较匹配时进行交替输出。
0	1	0	0		保持输出。
0	1	0	1		初始输出为 High 电平，在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平，在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平，在比较匹配时进行交替输出。
1	x	0	0	输入捕捉寄存器 *2	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

【符号说明】 x: Don't care

【注】 *1 从上电复位后到设定 TIOR 为止，输出 Low 电平。

*2 如果将 TMDR_4 的 BFB 位置“1”使 TGRD_4 用作缓冲寄存器，此设定就无效并且不发生输入捕捉/输出比较。

表 12.19 TIORH_0 (通道 0)

bit3	bit2	bit1	bit0	说明	
IOA3	IOA2	IOA1	IOA0	TGRA_0 的功能	TIOC0A 引脚的功能
0	0	0	0	输出比较寄存器	保持输出。*
0	0	0	1		初始输出为 Low 电平，在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平，在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平，在比较匹配时进行交替输出。
0	1	0	0		保持输出。
0	1	0	1		初始输出为 High 电平，在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平，在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平，在比较匹配时进行交替输出。
1	0	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	0	0	1		在下降沿进行输入捕捉。
1	0	1	x		在双边沿进行输入捕捉。
1	1	x	x		捕捉输入源是通道 1/ 计数时钟。 在 TCNT_1 进行递增或者递减计数时进行输入捕捉。

【符号说明】 x: Don't care

【注】 * 从上电复位后到设定 TIOR 为止，输出 Low 电平。

表 12.20 TIORL_0 (通道 0)

bit3	bit2	bit1	bit0	说明	
IOC3	IOC2	IOC1	IOC0	TGRC_0 的功能	TIOC0C 引脚的功能
0	0	0	0	输出比较寄存器 *2	保持输出。*1
0	0	0	1		初始输出为 Low 电平，在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平，在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平，在比较匹配时进行交替输出。
0	1	0	0		保持输出。
0	1	0	1		初始输出为 High 电平，在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平，在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平，在比较匹配时进行交替输出。
1	0	0	0	输入捕捉寄存器 *2	在上升沿进行输入捕捉。
1	0	0	1		在下降沿进行输入捕捉。
1	0	1	x		在双边沿进行输入捕捉。
1	1	x	x		捕捉输入源是通道 1/ 计数时钟。 在 TCNT_1 进行递增或者递减计数时进行输入捕捉。

【符号说明】 x: Don't care

【注】 *1 从上电复位后到设定 TIOR 为止，输出 Low 电平。

*2 如果将 TMDR_0 的 BFA 位置“1”使 TGRC_0 用作缓冲寄存器，此设定就无效并且不发生输入捕捉 / 输出比较。

表 12.21 TIOR_1 (通道 1)

bit3	bit2	bit1	bit0	说明	
IOA3	IOA2	IOA1	IOA0	TGRA_1 的功能	TIOC1A 引脚的功能
0	0	0	0	输出比较寄存器	保持输出。*
0	0	0	1		初始输出为 Low 电平，在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平，在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平，在比较匹配时进行交替输出。
0	1	0	0		保持输出。
0	1	0	1		初始输出为 High 电平，在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平，在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平，在比较匹配时进行交替输出。
1	0	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	0	0	1		在下降沿进行输入捕捉。
1	0	1	x		在双边沿进行输入捕捉。
1	1	x	x		在发生 TGRA_0 的比较匹配 / 输入捕捉时进行输入捕捉。

【符号说明】 x: Don't care

【注】 * 从上电复位后到设定 TIOR 为止，输出 Low 电平。

表 12.22 TIOR_2 (通道 2)

bit3	bit2	bit1	bit0	说明	
IOA3	IOA2	IOA1	IOA0	TGRA_2 的功能	TIOC2A 引脚的功能
0	0	0	0	输出比较寄存器	保持输出。*
0	0	0	1		初始输出为 Low 电平，在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平，在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平，在比较匹配时进行交替输出。
0	1	0	0		保持输出。
0	1	0	1		初始输出为 High 电平，在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平，在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平，在比较匹配时进行交替输出。
1	x	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

【符号说明】 x: Don't care

【注】 * 从上电复位后到设定 TIOR 为止，输出 Low 电平。

表 12.23 TIORH_3 (通道 3)

bit3	bit2	bit1	bit0	说明	
IOA3	IOA2	IOA1	IOA0	TGRA_3 的功能	TIOC3A 引脚的功能
0	0	0	0	输出比较寄存器	保持输出。*
0	0	0	1		初始输出为 Low 电平，在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平，在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平，在比较匹配时进行交替输出。
0	1	0	0		保持输出。
0	1	0	1		初始输出为 High 电平，在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平，在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平，在比较匹配时进行交替输出。
1	x	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

【符号说明】 x: Don't care

【注】 * 从上电复位后到设定 TIOR 为止，输出 Low 电平。

表 12.24 TIORL_3 (通道 3)

bit3	bit2	bit1	bit0	说明	
IOC3	IOC2	IOC1	IOC0	TGRC_3 的引脚	TIOC3C 引脚的功能
0	0	0	0	输出比较寄存器 *2	保持输出。*1
0	0	0	1		初始输出为 Low 电平，在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平，在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平，在比较匹配时进行交替输出。
0	1	0	0		保持输出。
0	1	0	1		初始输出为 High 电平，在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平，在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平，在比较匹配时进行交替输出。
1	x	0	0	输入捕捉寄存器 *2	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

【符号说明】 x: Don't care

【注】 *1 从上电复位后到设定 TIOR 为止，输出 Low 电平。

*2 如果将 TMDR_3 的 BFA 位置“1”使 TGRC_3 用作缓冲寄存器，此设定就无效并且不发生输入捕捉 / 输出比较。

表 12.25 TIORH_4 (通道 4)

bit3	bit2	bit1	bit0	说明	
IOA3	IOA2	IOA1	IOA0	TGRA_4 的功能	TIOC4A 引脚的功能
0	0	0	0	输出比较寄存器	保持输出。*
0	0	0	1		初始输出为 Low 电平，在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平，在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平，在比较匹配时进行交替输出。
0	1	0	0		保持输出。
0	1	0	1		初始输出为 High 电平，在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平，在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平，在比较匹配时进行交替输出。
1	x	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

【符号说明】 x: Don't care

【注】 * 从上电复位后到设定 TIOR 为止，输出 Low 电平。

表 12.26 TIORL_4 (通道 4)

bit3	bit2	bit1	bit0	说明	
IOC3	IOC2	IOC1	IOC0	TGRC_4 的功能	TIOC4C 引脚的功能
0	0	0	0	输出比较寄存器 *2	保持输出。*1
0	0	0	1		初始输出为 Low 电平，在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平，在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平，在比较匹配时进行交替输出。
0	1	0	0		保持输出。
0	1	0	1		初始输出为 High 电平，在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平，在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平，在比较匹配时进行交替输出。
1	x	0	0	输入捕捉寄存器 *2	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

【符号说明】 x: Don't care

【注】 *1 从上电复位后到设定 TIOR 为止，输出 Low 电平。

*2 如果将 TMDR_4 的 BFA 位置“1”使 TGRC_4 用作缓冲寄存器，此设定就无效并且不发生输入捕捉/输出比较。

12.3.4 定时器的中断允许寄存器 (TIER)

TIER 是 8 位可读写寄存器，控制允许或者禁止各通道的中断请求。在此模块中，通道 0 有 2 个，通道 1～4 各有 1 个，共计 6 个 TIER。

- TIER_0、TIER_1、TIER_2、TIER_3、TIER_4

位:	7	6	5	4	3	2	1	0
	TTGE	TTGE2	TCIEU	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	TTGE	0	R/W	A/D 转换开始请求允许 允许或者禁止因 TGRA 的输入捕捉 / 比较匹配而产生的 A/D 转换器开始请求。 0: 禁止产生 A/D 转换开始请求 1: 允许产生 A/D 转换开始请求
6	TTGE2	0	R/W	A/D 转换开始请求允许 2 在互补 PWM 模式中，允许或者禁止因 TCNT_4 的下溢（波谷）而产生的 A/D 转换请求。在通道 0～3 中，此位为保留位，读写值都为“0”。 0: 禁止因 TCNT_4 的下溢（波谷）而产生的 A/D 转换请求 1: 允许因 TCNT_4 的下溢（波谷）而产生的 A/D 转换请求
5	TCIEU	0	R/W	下溢中断允许 在通道 1 和通道 2 中，当 TSR 的 TCFU 标志被置“1”时，允许或者禁止 TCFU 标志的中断请求（TCIU）。在通道 0、3、4 中，此位为保留位，读写值都为“0”。 0: 禁止 TCFU 的中断请求（TCIU） 1: 允许 TCFU 的中断请求（TCIU）
4	TCIEV	0	R/W	上溢中断允许 当 TSR 的 TCFV 标志被置“1”时，允许或者禁止 TCFV 标志的中断请求（TCIV）。 0: 禁止 TCFV 的中断请求（TCIV） 1: 允许 TCFV 的中断请求（TCIV）
3	TGIED	0	R/W	TGR 中断允许 D 在通道 0、3、4 中，当 TSR 的 TGFD 位被置“1”时，允许或者禁止 TGFD 位的中断请求（TGID）。 在通道 1 和通道 2 中，此位为保留位，读写值都为“0”。 0: 禁止 TGFD 位的中断请求（TGID） 1: 允许 TGFD 位的中断请求（TGID）
2	TGIEC	0	R/W	TGR 中断允许 C 在通道 0、3、4 中，当 TSR 的 TGFC 位被置“1”时，允许或者禁止 TGFC 位的中断请求（TGIC）。 在通道 1 和通道 2 中，此位为保留位，读写值都为“0”。 0: 禁止 TGFC 位的中断请求（TGIC） 1: 允许 TGFC 位的中断请求（TGIC）

位	位名	初始值	R/W	说明
1	TGIEB	0	R/W	TGR 中断允许 B 当 TSR 的 TGFB 位被置“1”时，允许或者禁止 TGFB 位的中断请求 (TGIB)。 0: 禁止 TGFB 位的中断请求 (TGIB) 1: 允许 TGFB 位的中断请求 (TGIB)
0	TGIEA	0	R/W	TGR 中断允许 A 当 TSR 的 TGFA 位被置“1”时，允许或者禁止 TGFA 位的中断请求 (TGIA)。 0: 禁止 TGFA 位的中断请求 (TGIA) 1: 允许 TGFA 位的中断请求 (TGIA)

- TIER2_0

位:	7	6	5	4	3	2	1	0
	TTGE2	—	—	—	—	—	TGIEF	TGIEE
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明
7	TTGE2	0	R/W	A/D 转换开始请求允许 2 允许或者禁止因 TCNT_0 和 TGRE_0 的比较匹配而产生的 A/D 转换开始请求。 0: 禁止因 TCNT_0 和 TGRE_0 的比较匹配而产生的 A/D 转换开始请求 1: 允许因 TCNT_0 和 TGRE_0 的比较匹配而产生的 A/D 转换开始请求
6 ~ 2	—	全 0	R	保留位 读写值都为“0”。
1	TGIEF	0	R/W	TGR 中断允许 F 允许或者禁止因 TCNT_0 和 TGRF_0 的比较匹配而产生的中断请求。 0: 禁止 TGFE 位的中断请求 (TGIF) 1: 允许 TGFE 位的中断请求 (TGIF)
0	TGIEE	0	R/W	TGR 中断允许 E 允许或者禁止因 TCNT_0 和 TGRE_0 的比较匹配而产生的中断请求。 0: 禁止 TGEE 位的中断请求 (TGIE) 1: 允许 TGEE 位的中断请求 (TGIE)

12.3.5 定时器的状态寄存器 (TSR)

TSR 是 8 位可读写寄存器，表示各通道的状态。在此模块中，通道 0 有 2 个，通道 1 ~ 4 各有 1 个，共计 6 个 TSR。

- TSR_0、TSR_1、TSR_2、TSR_3、TSR_4

位:	7	6	5	4	3	2	1	0
	TCFD	—	TCFU	TCFV	TGFD	TGFC	TGFB	TGFA
初始值:	1	1	0	0	0	0	0	0
R/W:	R	R	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1

【注】*1 为了清除标志，只能在读“1”后写“0”。

位	位名	初始值	R/W	说明
7	TCFD	1	R	计数方向标志 这是表示通道 1 ~ 4 的 TCNT 计数方向的状态标志。 在通道 0 中，此位为保留位，读写值都为“1”。 0: TCNT 进行递减计数 1: TCNT 进行递增计数
6	—	1	R	保留位 读写值都为“1”。
5	TCFU	0	R/(W)*1	下溢标志 这是表示在通道 1 和通道 2 为相位计数模式时 TCNT 发生下溢的状态标志。 为了清除标志，只能写“0”。 在通道 0、3、4 中，此位为保留位，读写值都为“0”。 [清除条件] • 在 TCFU 位为“1”的状态下读 TCFU 后给 TCFU 位写“0”时*2 [置位条件] • 当 TCNT 的值发生下溢 (H'0000→H'FFFF) 时
4	TCFV	0	R/(W)*1	上溢标志 • 这是表示 TCNT 发生上溢的状态标志。为了清除标志，只能写“0”。 [清除条件] • 在 TCFV 位为“1”的状态下读 TCFV 位后给 TCFV 位写“0”时*2 [置位条件] • 当 TCNT 的值发生上溢 (H'FFFF→H'0000) 时 对于通道 4，如果在互补 PWM 模式中 TCNT_4 的值发生下溢 (H'0001→H'0000)，就将此标志置位。
3	TGFD	0	R/(W)*1	输入捕捉 / 输出比较标志 D 这是表示通道 0、3、4 的 TGRD 发生输入捕捉或者比较匹配的状态标志。 为了清除标志，只能写“0”。 在通道 1 和通道 2 中，此位为保留位，读写值都为“0”。 [清除条件] • 在 TGFD 位为“1”的状态下读 TGFD 位后给 TGFD 位写“0”时*2 [置位条件] • 在 TGRD 用作输出比较寄存器的情况下 TCNT 变为 TGRD 时 • 在 TGRD 用作输入捕捉寄存器的情况下通过输入捕捉信号将 TCNT 的值传送到 TGRD 时

位	位名	初始值	R/W	说明
2	TGFC	0	R/(W)*1	<p>输入捕捉 / 输出比较标志 C</p> <p>这是表示通道 0、3、4 的 TGRC 发生输入捕捉或比较匹配的状态标志。为了清除标志，只能写“0”。</p> <p>在通道 1 和通道 2 中，此位为保留位，读写值都为“0”。</p> <p>[清除条件]</p> <ul style="list-style-type: none"> 在 TGFC 位为“1”的状态下读 TGFC 位后给 TGFC 位写“0”时*2 <p>[置位条件]</p> <ul style="list-style-type: none"> 在 TGRC 用作输出比较寄存器的情况下 TCNT 变为 TGRC 时 在 TGRC 用作输入捕捉寄存器的情况下通过输入捕捉信号将 TCNT 的值传送到 TGRC 时
1	TGFB	0	R/(W)*1	<p>输入捕捉 / 输出比较标志 B</p> <p>这是表示 TGRB 发生输入捕捉或比较匹配的状态标志。为了清除标志，只能写“0”。</p> <p>[清除条件]</p> <ul style="list-style-type: none"> 在 TGFB 位为“1”的状态下读 TGFB 位后给 TGFB 写“0”时*2 <p>[置位条件]</p> <ul style="list-style-type: none"> 在 TGRB 用作输出比较寄存器的情况下 TCNT 变为 TGRB 时 在 TGRB 用作输入捕捉寄存器的情况下通过输入捕捉信号将 TCNT 的值传送到 TGRB 时
0	TGFA	0	R/(W)*1	<p>输入捕捉 / 输出比较标志 A</p> <p>这是表示 TGRA 发生输入捕捉或比较匹配的状态标志。为了清除标志，只能写“0”。</p> <p>[清除条件]</p> <ul style="list-style-type: none"> 当通过 TGIA 中断启动直接存储器存取控制器时 在 TGFA 位为“1”的状态下读 TGFA 位后给 TGFA 位写“0”时*2 <p>[置位条件]</p> <ul style="list-style-type: none"> 在 TGRA 用作输出比较寄存器的情况下 TCNT 变为 TGRA 时 在 TGRA 用作输入捕捉寄存器的情况下通过输入捕捉信号将 TCNT 的值传送到 TGRA 时

【注】 *1 为了清除标志，只能在读“1”后写“0”。

*2 如果在读“1”后到写“0”前发生下一位的标志置位，即使写“0”也不清除标志。必须在重新读“1”后写“0”。

• TSR2_0

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TGFF	TGFE
初始值:	1	1	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/(W)*1	R/(W)*1

【注】 *1 为了清除标志，只能在读“1”后写“0”。

位	位名	初始值	R/W	说明
7、6	—	全 1	R	保留位 读写值都为“1”。
5~2	—	全 0	R	保留位 读写值都为“0”。
1	TGFF	0	R/(W)*1	比较匹配标志 F 这是表示 TCNT_0 和 TGRF_0 发生比较匹配的状态标志。 [清除条件] • 在 TGFF 位为“1”的状态下读 TGFF 位后给 TGFF 为写“0”时*2 [置位条件] • 在将 TGRF_0 用作比较寄存器的情况下 TCNT_0 变为 TGRF_0 时
0	TGFE	0	R/(W)*1	比较匹配标志 E 这是表示 TCNT_0 和 TGRE_0 发生比较匹配的状态标志。 [清除条件] • 在 TGFE 位为“1”的状态下读 TGFE 位后给 TGFE 位写“0”时*2 [置位条件] • 在将 TGRE_0 用作比较寄存器的情况下 TCNT_0 变为 TGRE_0 时

【注】 *1 为了清除标志，只能在读“1”后写“0”。

*2 如果在读“1”后到写“0”前发生下一次的标志置位，即使写“0”也不清除标志。必须在重新读“1”后写“0”。

12.3.6 定时器的缓冲运行传送模式寄存器 (TBTM)

TBTM 是 8 位可读写寄存器，设定在 PWM 模式中缓冲寄存器到定时器的通用寄存器的传送时序。在此模块中，通道 0、3、4 各有 1 个，共计 3 个 TBTM。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	TTSE	TTSB	TTSA
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 3	—	全 0	R	保留位 读写值都为“0”。
2	TTSE	0	R/W	时序选择 E 设定在缓冲运行时 TGRF_0 到 TGRE_0 的传送时序。 在通道 3 和通道 4 中，此位为保留位，读写值都为“0”。 在 PWM 以外的模式中使用通道 0 时，不能将此位置“1”。 0: 在通道 0 发生比较匹配 E 时 1: 在清除 TCNT_0 时
1	TTSB	0	R/W	时序选择 B 设定在各通道进行缓冲运行时 TGRD 到 TGRB 的传送时序。 对于在 PWM 以外的模式使用的通道，不能将此位置“1”。 0: 在各通道发生比较匹配 B 时 1: 在清除各通道的 TCNT 时
0	TTSA	0	R/W	时序选择 A 设定在各通道进行缓冲运行时 TGRC 到 TGRA 的传送时序。 对于在 PWM 以外的模式使用的通道，不能将此位置“1”。 0: 在各通道发生比较匹配 A 时 1: 在清除各通道的 TCNT 时

12.3.7 定时器的输入捕捉控制寄存器 (TICCR)

TICCR 是 8 位可读写寄存器，控制 TCNT_1 和 TCNT_2 级联时的输入捕捉条件。在此模块中，通道 1 有 1 个 TICCR。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	I2BE	I2AE	I1BE	I1AE
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7~4	—	全 0	R	保留位 读写值都为“0”。
3	I2BE	0	R/W	输入捕捉允许 选择是否将 TIOC2B 引脚追加到 TGRB_1 的输入捕捉条件。 0: 不将 TIOC2B 引脚追加到 TGRB_1 的输入捕捉条件 1: 将 TIOC2B 引脚追加到 TGRB_1 的输入捕捉条件
2	I2AE	0	R/W	输入捕捉允许 选择是否将 TIOC2A 引脚追加到 TGRA_1 的输入捕捉条件。 0: 不将 TIOC2A 引脚追加到 TGRA_1 的输入捕捉条件 1: 将 TIOC2A 引脚追加到 TGRA_1 的输入捕捉条件
1	I1BE	0	R/W	输入捕捉允许 选择是否将 TIOC1B 引脚追加到 TGRB_2 的输入捕捉条件。 0: 不将 TIOC1B 引脚追加到 TGRB_2 的输入捕捉条件 1: 将 TIOC1B 引脚追加到 TGRB_2 的输入捕捉条件
0	I1AE	0	R/W	输入捕捉允许 选择是否将 TIOC1A 引脚追加到 TGRA_2 的输入捕捉条件。 0: 不将 TIOC1A 引脚追加到 TGRA_2 的输入捕捉条件 1: 将 TIOC1A 引脚追加到 TGRA_2 的输入捕捉条件

12.3.8 定时器的 A/D 转换开始请求控制寄存器 (TADCR)

TADCR 是 16 位可读写寄存器，允许或者禁止 A/D 转换开始请求以及设定 A/D 转换开始请求是否联动中断减少功能。在此模块中，通道 4 有 1 个 TADCR。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BF[1:0]		—	—	—	—	—	—	UT4AE	DT4AE	UT4BE	DT4BE	ITA3AE	ITA4VE	ITB3AE	ITB4VE
初始值:	0	0	0	0	0	0	0	0	0	0*	0	0*	0*	0*	0*	0*
R/W:	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15、14	BF[1:0]	00	R/W	TADCOBRA/B_4 传送时序选择 选择 TADCOBRA/B_4 到 TADCORA/B_4 的传送时序。 详细内容请参照表 12.27。
13 ~ 8	—	全 0	R	保留位 读写值都为“0”。
7	UT4AE	0	R/W	递增计数 TRG4AN 允许 允许或者禁止在 TCNT_4 进行递增计数时发生 A/D 转换的开始请求 (TRG4AN)。 0: 禁止在 TCNT_4 进行递增计数时发生 A/D 转换的开始请求 (TRG4AN) 1: 允许在 TCNT_4 进行递增计数时发生 A/D 转换的开始请求 (TRG4AN)
6	DT4AE	0*	R/W	递减计数 TRG4AN 允许 允许或者禁止在 TCNT_4 进行递减计数时发生 A/D 转换的开始请求 (TRG4AN)。 0: 禁止在 TCNT_4 进行递减计数时发生 A/D 转换的开始请求 (TRG4AN) 1: 允许在 TCNT_4 进行递减计数时发生 A/D 转换的开始请求 (TRG4AN)
5	UT4BE	0	R/W	递增计数 TRG4BN 允许 允许或者禁止在 TCNT_4 进行递增计数时发生 A/D 转换的开始请求 (TRG4BN)。 0: 禁止在 TCNT_4 进行递增计数时发生 A/D 转换的开始请求 (TRG4BN) 1: 允许在 TCNT_4 进行递增计数时发生 A/D 转换的开始请求 (TRG4BN)
4	DT4BE	0*	R/W	递减计数 TRG4BN 允许 允许或者禁止在 TCNT_4 进行递减计数时发生 A/D 转换的开始请求 (TRG4BN)。 0: 禁止在 TCNT_4 进行递减计数时发生 A/D 转换的开始请求 (TRG4BN) 1: 允许在 TCNT_4 进行递减计数时发生 A/D 转换的开始请求 (TRG4BN)
3	ITA3AE	0*	R/W	TGIA_3 中断减少联动允许 选择 A/D 转换的开始请求 (TRG4AN) 是否联动 TGIA_3 中断减少功能。 0: 不联动 TGIA_3 中断减少功能 1: 联动 TGIA_3 中断减少功能

位	位名	初始值	R/W	说明
2	ITA4VE	0*	R/W	TCIV_4 中断减少联动允许 选择 A/D 转换的开始请求 (TRG4AN) 是否联动 TCIV_4 中断减少功能。 0: 不联动 TCIV_4 中断减少功能 1: 联动 TCIV_4 中断减少功能
1	ITB3AE	0*	R/W	TGIA_3 中断减少联动允许 选择 A/D 转换的开始请求 (TRG4BN) 是否联动 TGIA_3 中断减少功能。 0: 不联动 TGIA_3 中断减少功能 1: 联动 TGIA_3 中断减少功能
0	ITB4VE	0*	R/W	TCIV_4 中断减少联动允许 选择 A/D 转换的开始请求 (TRG4BN) 是否联动 TCIV_4 中断减少功能。 0: 不联动 TCIV_4 中断减少功能 1: 联动 TCIV_4 中断减少功能

- 【注】
- 禁止以 8 位为单位存取 TADCR，必须以 16 位为单位进行存取。
 - 当禁止中断减少功能时（将定时器的中断减少设定寄存器 (TITCR) 的 T3AEN 位和 T4VEN 位置“0”或者将 TITCR 的减少次数设定 (3ACOR 和 4VCOR) 置“0”时)，必须设定为不联动中断减少功能（将定时器的 A/D 转换开始请求控制寄存器 (TADCR) 的 ITA3AE、ITA4VE、ITB3AE、ITB4VE 位置“0”）。
 - 在禁止中断减少功能时，如果设定为联动中断减少功能，就不进行 A/D 转换的开始请求。
- * 除了互补 PWM 模式以外，不能置“1”。

表 12.27 通过 BF1 位和 BF0 位设定的传送时序

bit7	bit6	说明
BF1	BF0	
0	0	不从周期设定缓冲寄存器传送到周期设定寄存器。
0	1	在 TCNT_4 的波峰从周期设定缓冲寄存器传送到周期设定寄存器 *1。
1	0	在 TCNT_4 的波谷从周期设定缓冲寄存器传送到周期设定寄存器 *2。
1	1	在 TCNT_4 的波峰和波谷从周期设定缓冲寄存器传送到周期设定寄存器 *2。

- 【注】
- *1 在互补 PWM 模式中，在 TCNT_4 的波峰从周期设定缓冲寄存器传送到周期设定寄存器；在复位同步 PWM 模式中，当 TCNT_3 和 TGRA_3 发生比较匹配时，从周期设定缓冲寄存器传送到周期设定寄存器；在 PWM 模式 1 或者正常运行模式中，当 TCNT_4 和 TGRA_4 发生比较匹配时，从周期设定缓冲寄存器传送到周期设定寄存器。
 - *2 除了互补 PWM 模式以外，禁止此设定。

12.3.9 定时器的 A/D 转换开始请求周期设定寄存器 (TADCORA/B_4)

TADCORA/B_4 是 16 位可读写寄存器。当 TADCORA/B_4 和 TCNT_4 相同时，发生对应的 A/D 转换开始请求。TADCORA/B_4 的初始值为 “H'FFFF”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

【注】禁止以 8 位为单位存取 TADCORA/B_4，必须以 16 位为单位进行存取。

12.3.10 定时器的 A/D 转换开始请求周期设定缓冲寄存器 (TADCOBRA/B_4)

TADCOBRA/B_4 是 16 位可读写寄存器。在波峰或者波谷将 TADCORA/B_4 的缓冲寄存器的值传送到 TADCORA/B_4。

TADCOBRA/B_4 的初始值为 “H'FFFF”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

【注】禁止以 8 位为单位存取 TADCOBRA/B_4，必须以 16 位为单位进行存取。

12.3.11 定时器的计数器 (TCNT)

TCNT 是 16 位可读写计数器。通道 0 ~ 4 各有 1 个，共计 5 个 TCNT。

禁止以 8 位为单位存取 TCNT，必须以 16 位为单位进行存取。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

【注】禁止以 8 位为单位存取 TCNT，必须以 16 位为单位进行存取。

12.3.12 定时器的通用寄存器 (TGR)

TGR 是 16 位可读写寄存器。通道 0 有 6 个，通道 1 和通道 2 各有 2 个，通道 3 和通道 4 各有 4 个，共计 18 个通用寄存器。

TGRA、TGRB、TGRC、TGRD 是输出比较 / 输入捕捉兼用的寄存器。能将通道 0、3、4 的 TGRC 和 TGRD 用作缓冲寄存器。TGR 和缓冲寄存器的组合为 TGRA-TGRC、TGRB-TGRD。

TGRE_0 和 TGRF_0 用作比较寄存器，当 TCNT_0 和 TGRE_0 相同时，能产生 A/D 转换开始请求。能将 TGRF 用作缓冲寄存器。TGR 和缓冲寄存器的组合为 TGRE-TGRF。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

【注】禁止以 8 位为单位存取 TGR，必须以 16 位为单位进行存取。TGR 的初始值为“H'FFFF”。

12.3.13 定时器的启动寄存器 (TSTR)

TSTR 是 8 位可读写寄存器，选择通道 0 ~ 4 的 TCNT 的运行或者停止。

在给 TMDR 设定运行模式或者给 TCR 设定 TCNT 的计数时钟时，必须在停止 TCNT 计数器后进行设定。

位:	7	6	5	4	3	2	1	0
	CST4	CST3	—	—	—	CST2	CST1	CST0
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	CST4	0	R/W	计数器启动 4、3
6	CST3	0	R/W	选择 TCNT 的运行或者停止。 在 TIOC 引脚为输出状态下运行时，如果给 CST 位写“0”，就停止计数器的计数，但是保持 TIOC 引脚的输出比较的输出电平。如果在 CST 位为“0”的状态下写 TIOR，就将引脚的输出电平更新为所设定的初始输出值。 0: TCNT_4 和 TCNT_3 停止计数 1: TCNT_4 和 TCNT_3 计数
5 ~ 3	—	全 0	R	保留位 读写值都为“0”。
2	CST2	0	R/W	计数器启动 2 ~ 0
1	CST1	0	R/W	选择 TCNT 的运行或者停止。
0	CST0	0	R/W	在 TIOC 引脚为输出状态下运行时，如果给 CST 位写“0”，就停止计数器的计数，但是保持 TIOC 引脚的输出比较的输出电平。如果在 CST 位为“0”的状态下写 TIOR，就将引脚的输出电平更新为所设定的初始输出值。 0: TCNT_2 ~ TCNT_0 停止计数 1: TCNT_2 ~ TCNT_0 计数

12.3.14 定时器的同步寄存器 (TSYR)

TSYR 是 8 位可读写寄存器，选择通道 0 ~ 4 的 TCNT 进行独立运行或者同步运行。对应位为“1”的通道进行同步运行。

位:	7	6	5	4	3	2	1	0
	SYNC4	SYNC3	—	—	—	SYNC2	SYNC1	SYNC0
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	SYNC4	0	R/W	定时器同步 4、3
6	SYNC3	0	R/W	选择是进行独立运行还是与其他通道同步运行。 如果选择同步运行，就能对多个 TCNT 进行同步预置并且通过清除其他通道的计数器进行同步清除。 要设定同步运行时，至少需要将 2 个通道的 SYNC 位置“1”；要设定同步清除时，除了 SYNC 位以外，还需要通过 TCR 的 CCLR2 ~ CCLR0 位设定 TCNT 的清除源。 0: TCNT_4 和 TCNT_3 独立运行 (TCNT 的预置 / 清除与其他通道无关) 1: TCNT_4 和 TCNT_3 进行同步运行 能进行 TCNT 的同步预置 / 同步清除。
5 ~ 3	—	全 0	R	保留位 读写值都为“0”。
2	SYNC2	0	R/W	定时器同步 2 ~ 0
1	SYNC1	0	R/W	选择是进行独立运行还是与其他通道同步运行。
0	SYNC0	0	R/W	如果选择同步运行，就能对多个 TCNT 进行同步预置并且通过清除其他通道的计数器进行同步清除。 要设定同步运行时，至少需要将 2 个通道的 SYNC 位置“1”；要设定同步清除时，除了 SYNC 位以外，还需要通过 TCR 的 CCLR2 ~ CCLR0 位设定 TCNT 的清除源。 0: TCNT_2 ~ TCNT_0 独立运行 (TCNT 的预置 / 清除与其他通道无关) 1: TCNT_2 ~ TCNT_0 进行同步运行 能进行 TCNT 的同步预置 / 同步清除。

12.3.15 定时器的读写允许寄存器 (TRWER)

TRWER 是 8 位可读写寄存器，用于设定允许或者禁止存取通道 3 和通道 4 的误写防止对象寄存器 / 计数器。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	RWE
初始值:	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
7 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	RWE	1	R/W	读写允许 允许或者禁止读写误写防止对象寄存器。 0: 禁止读写寄存器 1: 允许读写寄存器 [清除条件] • 在 RWE 位为“1”的状态下读 RWE 位后给 RWE 位写“0”时

- 误写防止对象寄存器/计数器
TCR_3、4、TMDR_3、4、TIORH_3、4、TIORL_3、4、TIER_3、4、TGRA_3、4、TGRB_3、4、TOER、TOCR1、TOCR2、TGCR、TCDR、TDDR和TCNT_3、4，共计22个寄存器。

12.3.16 定时器的输出主控允许寄存器 (TOER)

TOER 是 8 位可读写寄存器，允许或者禁止输出引脚的 TIOC4D、TIOC4C、TIOC3D、TIOC4B、TIOC4A 和 TIOC3B 的输出设定。如果不设定 TOER 的各位，就无法正确地输出这些引脚。在通道 3 和通道 4 中，必须在设定通道 3 通道 4 的 TIOR 前给 TOER 设定值。

位:	7	6	5	4	3	2	1	0
	—	—	OE4D	OE4C	OE3D	OE4B	OE4A	OE3B
初始值:	1	1	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7、6	—	全 1	R	保留位 读写值都为“1”。
5	OE4D	0	R/W	主控允许 TIOC4D 允许或者禁止 TIOC4D 引脚的此模块的输出。 0: 禁止此模块的输出 (无效电平) * 1: 允许此模块的输出
4	OE4C	0	R/W	主控允许 TIOC4C 允许或者禁止 TIOC4C 引脚的此模块的输出。 0: 禁止此模块的输出 (无效电平) * 1: 允许此模块的输出
3	OE3D	0	R/W	主控允许 TIOC3D 允许或者禁止 TIOC3D 引脚的此模块的输出。 0: 禁止此模块的输出 (无效电平) * 1: 允许此模块的输出
2	OE4B	0	R/W	主控允许 TIOC4B 允许或者禁止 TIOC4B 引脚的此模块的输出。 0: 禁止此模块的输出 (无效电平) * 1: 允许此模块的输出
1	OE4A	0	R/W	主控允许 TIOC4A 允许或者禁止 TIOC4A 引脚的此模块的输出。 0: 禁止此模块的输出 (无效电平) * 1: 允许此模块的输出
0	OE3B	0	R/W	主控允许 TIOC3B 允许或者禁止 TIOC3B 引脚的此模块的输出。 0: 禁止此模块的输出 (无效电平) * 1: 允许此模块的输出

【注】* 无效电平取决于定时器的输出控制寄存器 1/2 (TOCR1/2) 的设定，详细内容请参照“12.3.17 定时器的输出控制寄存器 1 (TOCR1)”和“12.3.18 定时器的输出控制寄存器 2 (TOCR2)”。除了互补 PWM 模式 / 复位同步 PWM 模式以外，在进行此模块的输出时必须置“1”。如果置“0”就输出低电平。

12.3.17 定时器的输出控制寄存器 1 (TOCR1)

TOCR1 是 8 位可读写寄存器，允许或者禁止与互补 PWM 模式 / 复位同步 PWM 模式的 PWM 周期同步进行交替输出以及对 PWM 输出电平进行反相控制。

位:	7	6	5	4	3	2	1	0
	—	PSYE	—	—	TOCL	TOCS	OLSN	OLSP
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R	R/(W)*3	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	—	0	R	保留位 读写值都为“0”。
6	PSYE	0	R/W	PWM 同步输出允许 允许或者禁止与 PWM 周期同步进行交替输出。 0: 禁止交替输出 1: 允许交替输出
5、4	—	全 0	R	保留位 读写值都为“0”。
3	TOCL	0	R/(W)*3	TOC 寄存器的写禁止位 *1 设定禁止或者允许写 TOCR1 寄存器的 TOCS 位、OLSN 位和 OLSP 位。 0: 允许写 TOCS 位、OLSN 位和 OLSP 位 1: 禁止写 TOCS 位、OLSN 位和 OLSP 位
2	TOCS	0	R/W	TOC 选择位 选择互补 PWM 模式 / 复位同步 PWM 模式的输出电平是 TOCR1 的设定有效还是 TOCR2 的设定有效。 0: TOCR1 的设定有效 1: TOCR2 的设定有效
1	OLSN	0	R/W	输出电平选择 N*2 在复位同步 PWM 模式 / 互补 PWM 模式中，选择反相的输出电平。 请参照表 12.28。
0	OLSP	0	R/W	输出电平选择 P*2 在复位同步 PWM 模式 / 互补 PWM 模式中，选择正相的输出电平。 请参照表 12.29。

【注】 *1 能通过将 TOCL 位置“1”，防止 CPU 失控时的误写。

*2 通过将 TOCS 位置“0”，使此设定变为有效。

*3 在上电复位后，只能写 1 次“1”。写“1”后不能写“0”。

表 12.28 输出电平的选择功能

bit1	功能			
OLSN	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	高电平	低电平	高电平	低电平
1	低电平	高电平	低电平	高电平

【注】 在开始计数并且经过空载时间后，反相波形的初始输出值变为有效电平。

表 12.29 输出电平的选择功能

bit0	功能			
OLSP	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	高电平	低电平	低电平	高电平
1	低电平	高电平	高电平	低电平

OLSN 位为“1”并且 OLSP 位为“1”时的互补 PWM 模式的输出例子（1 相）如图 12.2 所示。

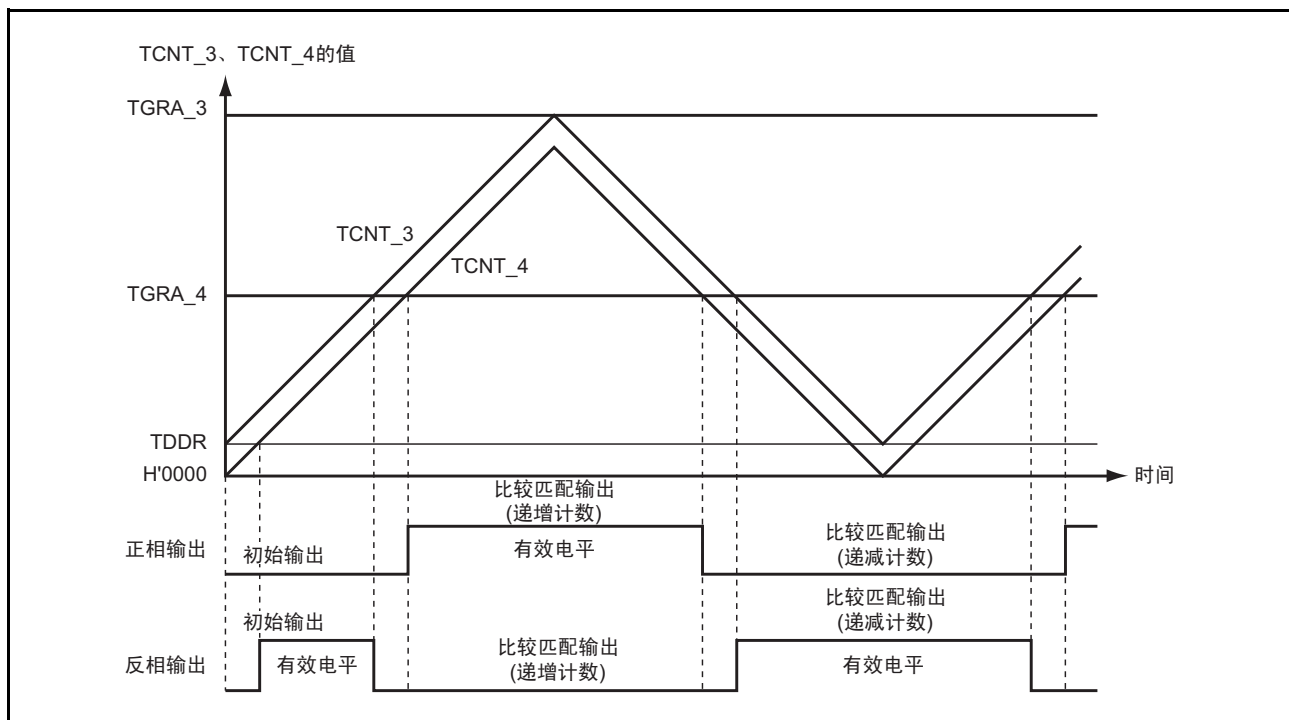


图 12.2 互补 PWM 模式的输出电平例子

12.3.18 定时器的输出控制寄存器 2 (TOCR2)

TOCR2 是 8 位可读写寄存器，用于对互补 PWM 模式 / 复位同步 PWM 模式的 PWM 输出电平进行反相控制。

位:	7	6	5	4	3	2	1	0
	BF[1:0]	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P	
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7、6	BF[1:0]	00	R/W	TOLBR 缓冲传送时序选择 选择从 TOLBR 到 TOCR2 的缓冲传送时序。 详细内容请参照表 12.30。
5	OLS3N	0	R/W	输出电平选择 3N* 在复位同步 PWM 模式 / 互补 PWM 模式中，选择 TIOC4D 的输出电平， 请参照表 12.31。
4	OLS3P	0	R/W	输出电平选择 3P* 在复位同步 PWM 模式 / 互补 PWM 模式中，选择 TIOC4B 的输出电平， 请参照表 12.32。
3	OLS2N	0	R/W	输出电平选择 2N* 在复位同步 PWM 模式 / 互补 PWM 模式中，选择 TIOC4C 的输出电平， 请参照表 12.33。
2	OLS2P	0	R/W	输出电平选择 2P* 在复位同步 PWM 模式 / 互补 PWM 模式中，选择 TIOC4A 的输出电平， 请参照表 12.34。
1	OLS1N	0	R/W	输出电平选择 1N* 在复位同步 PWM 模式 / 互补 PWM 模式中，选择 TIOC3D 的输出电平， 请参照表 12.35。
0	OLS1P	0	R/W	输出电平选择 1P* 在复位同步 PWM 模式 / 互补 PWM 模式中，选择 TIOC3B 的输出电平， 请参照表 12.36。

【注】* 通过将 TOCR1 的 TOCS 位置 “1”，使此设定变为有效。

表 12.30 BF1 位和 BF0 位的设定

bit7	bit6	说明	
BF1	BF0	互补 PWM 模式	复位 PWM 模式
0	0	不从缓冲寄存器 (TOLBR) 传送到 TOCR2。	不从缓冲寄存器 (TOLBR) 传送到 TOCR2。
0	1	在 TCNT_4 的波峰从缓冲寄存器 (TOLBR) 传送到 TOCR2。	在清除 TCNT_3/4 计数器时从缓冲寄存器 (TOLBR) 传送到 TOCR2。
1	0	在 TCNT_4 的波谷从缓冲寄存器 (TOLBR) 传送到 TOCR2。	禁止设定。
1	1	在 TCNT_4 的波峰和波谷从缓冲寄存器 (TOLBR) 传送到 TOCR2。	禁止设定。

表 12.31 TIOC4D 输出电平的选择功能

bit5	功能			
OLS3N	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	高电平	低电平	高电平	低电平
1	低电平	高电平	低电平	高电平

【注】 在开始计数并且经过空载时间后，反相波形的初始输出值变为有效电平。

表 12.32 TIOC4B 输出电平的选择功能

bit4	功能			
OLS3P	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	高电平	低电平	低电平	高电平
1	低电平	高电平	高电平	低电平

表 12.33 TIOC4C 输出电平的选择功能

bit3	功能			
OLS2N	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	高电平	低电平	高电平	低电平
1	低电平	高电平	低电平	高电平

【注】 在开始计数并且经过空载时间后，反相波形的初始输出值变为有效电平。

表 12.34 TIOC4A 输出电平的选择功能

bit2	功能			
OLS2P	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	高电平	低电平	低电平	高电平
1	低电平	高电平	高电平	低电平

表 12.35 TIOC3D 输出电平的选择功能

bit1	功能			
OLS1N	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	高电平	低电平	高电平	低电平
1	低电平	高电平	低电平	高电平

【注】 在开始计数并且经过空载时间后，反相波形的初始输出值变为有效电平。

表 12.36 TIOC4B 输出电平的选择功能

bit0	功能			
	OLS1P	初始输出	有效电平	比较匹配输出
递增计数				递减计数
0	高电平	低电平	低电平	高电平
1	低电平	高电平	高电平	低电平

12.3.19 定时器的输出电平缓冲寄存器 (TOLBR)

TOLBR 是 TOCR2 的缓冲寄存器，用于设定互补 PWM 模式/复位同步 PWM 模式的 PWM 输出电平。TOLBR 是 8 位可读写寄存器。

位:	7	6	5	4	3	2	1	0
	—	—	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7、6	—	全 0	R	保留位 读写值都为“0”。
5	OLS3N	0	R/W	必须给 TOCR2 的 OLS3N 位设定缓冲传送的值。
4	OLS3P	0	R/W	必须给 TOCR2 的 OLS3P 位设定缓冲传送的值。
3	OLS2N	0	R/W	必须给 TOCR2 的 OLS2N 位设定缓冲传送的值。
2	OLS2P	0	R/W	必须给 TOCR2 的 OLS2P 位设定缓冲传送的值。
1	OLS1N	0	R/W	必须给 TOCR2 的 OLS1N 位设定缓冲传送的值。
0	OLS1P	0	R/W	必须给 TOCR2 的 OLS1P 位设定缓冲传送的值。

在缓冲运行过程中设定 PWM 输出电平时的设定步骤例子如图 12.3 所示。

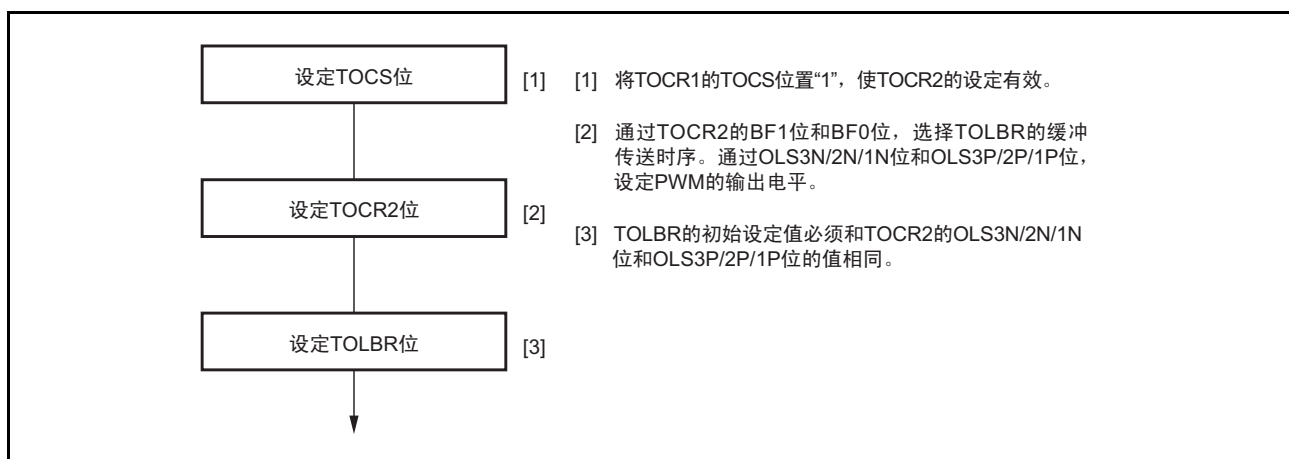


图 12.3 在缓冲运行过程中设定 PWM 输出电平时的设定步骤例子

12.3.20 定时器的门控寄存器 (TGCR)

TGCR 是 8 位可读写寄存器，在复位同步 PWM 模式 / 互补 PWM 模式中对控制无刷 DC 马达控制所需的波形输出进行控制。除了互补 PWM 模式 / 复位同步 PWM 模式以外，此寄存器的设定无效。

位:	7	6	5	4	3	2	1	0
	—	BDC	N	P	FB	WF	VF	UF
初始值:	1	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	—	1	R	保留位 读写值都为“1”。
6	BDC	0	R/W	无刷 DC 马达 选择此寄存器功能的有效或者无效。 0: 正常输出 1: 此寄存器的功能有效
5	N	0	R/W	反相输出 (N) 控制 在输出反相引脚 (TIOC3D 引脚、TIOC4C 引脚、TIOC4D 引脚) 时，选择是进行电平输出还是进行复位同步 PWM/ 互补 PWM 输出。 0: 电平输出 1: 复位同步 PWM/ 互补 PWM 输出
4	P	0	R/W	正相输出 (P) 控制 在输出正相引脚 (TIOC3B 引脚、TIOC4A 引脚、TIOC4B 引脚) 时，选择是进行电平输出还是进行复位同步 PWM/ 互补 PWM 输出。 0: 电平输出 1: 复位同步 PWM/ 互补 PWM 输出
3	FB	0	R/W	外部反馈信号允许 选择是通过此模块 / 通道 0 的 TGRA、TGRB、TGRC 的输入捕捉信号自动进行正反相输出的转换，还是通过给 TGCR 的 bit2 ~ 0 写“0”或者“1”进行正反相输出的转换。 0: 通过外部输入进行输出转换 (输入源为通道 0 的 TGRA、TGRB、TGRC 的输入捕捉信号) 1: 通过软件进行输出转换 (TGCR 的 UF、VF、WF 的设定值)
2	WF	0	R/W	输出相的切换 2 ~ 0
1	VF	0	R/W	设定正相 / 反相的输出相 ON/OFF。这些位的设定只在此寄存器的 FB 位为“1”时有效。此时，bit2 ~ 0 的设定取代外部输入，请参照表 12.37。
0	UF	0	R/W	

表 12.37 输出电平的选择功能

bit2	bit1	bit0	功能					
			TIOC3B	TIOC4A	TIOC4B	TIOC3D	TIOC4C	TIOC4D
WF	VF	UF	U 相	V 相	W 相	U 相	V 相	W 相
0	0	0	OFF	OFF	OFF	OFF	OFF	OFF
0	0	1	ON	OFF	OFF	OFF	OFF	ON
0	1	0	OFF	ON	OFF	ON	OFF	OFF
0	1	1	OFF	ON	OFF	OFF	OFF	ON
1	0	0	OFF	OFF	ON	OFF	ON	OFF
1	0	1	ON	OFF	OFF	OFF	ON	OFF
1	1	0	OFF	OFF	ON	ON	OFF	OFF
1	1	1	OFF	OFF	OFF	OFF	OFF	OFF

12.3.21 定时器的副计数器（TCNTS）

TCNTS 是只用于互补 PWM 模式的 16 位只读计数器，初始值为“H'0000”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

【注】禁止以 8 位为单位存取 TCNTS，必须以 16 位为单位进行存取。

12.3.22 定时器的空载时间数据寄存器（TDDR）

TDDR 是只用于互补 PWM 模式的 16 位寄存器，在互补 PWM 模式中设定 TCNT_3 和 TCNT_4 计数器的偏移值。在互补 PWM 模式中清除 TCNT_3 和 TCNT_4 计数器后重新开始计数时，将 TDDR 寄存器的值装入到 TCNT_3 计数器并且开始计数。TDDR 的初始值为“H'FFFF”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

【注】禁止以 8 位为单位存取 TDDR，必须以 16 位为单位进行存取。

12.3.23 定时器的周期数据寄存器（TCDR）

TCDR 是只用于互补 PWM 模式的 16 位寄存器，必须将 TCDR 寄存器设定为 1/2 个 PWM 载波周期。在互补 PWM 模式中，此寄存器随时和 TCNTS 计数器进行比较，如果两者的值相同，TCNTS 计数器就切换计数方向（递减计数 → 递增计数）。TCDR 的初始值为“H'FFFF”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

【注】禁止以 8 位为单位存取 TCDR，必须以 16 位为单位进行存取。

12.3.24 定时器的周期缓冲寄存器 (TCBR)

TCBR 是只用于互补 PWM 模式的 16 位寄存器，用作 TCDR 寄存器的缓冲寄存器。在 TMDR 寄存器设定的传送时序将 TCBR 寄存器的值传送到 TCDR 寄存器。TCBR 的初始值为“H'FFFF”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

【注】禁止以8位为单位存取TCBR，必须以16位为单位进行存取。

12.3.25 定时器的中断减少设定寄存器 (TITCR)

TITCR 是 8 位可读写寄存器，控制禁止或者允许中断减少功能以及设定中断减少次数。此模块有 1 个 TITCR。

位:	7	6	5	4	3	2	1	0
	T3AEN	3ACOR[2:0]			T4VEN	4VCOR[2:0]		
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	T3AEN	0	R/W	T3AEN 设定禁止或者允许减少 TGIA_3 中断。 0: 禁止减少 TGIA_3 中断 1: 允许减少 TGIA_3 中断
6 ~ 4	3ACOR[2:0]	000	R/W	将 TGIA_3 的中断减少次数设定为 0 ~ 7 次 *， 详细内容请参照表 12.38。
3	T4VEN	0	R/W	T4VEN 此位设定禁止或者允许减少 TCIV_4 中断。 0: 禁止减少 TCIV_4 中断 1: 允许减少 TCIV_4 中断
2 ~ 0	4VCOR[2:0]	000	R/W	将 TCIV_4 的中断减少次数设定为 0 ~ 7 次 *， 详细内容请参照表 12.39。

【注】* 如果将中断减少次数设定为“0”，就不减少中断。
另外，必须在更改中断减少次数前，通过将 T3AEN 位和 T4VEN 位置“0”，清除中断减少次数计数器 (TITCNT)。

表 12.38 通过 3ACOR2 ~ 3ACOR0 位设定的中断减少次数

bit6	bit5	bit4	说明
3ACOR2	3ACOR1	3ACOR0	
0	0	0	不减少 TGIA_3 的中断。
0	0	1	将 TGIA_3 的中断减少次数设定为 1 次。
0	1	0	将 TGIA_3 的中断减少次数设定为 2 次。
0	1	1	将 TGIA_3 的中断减少次数设定为 3 次。
1	0	0	将 TGIA_3 的中断减少次数设定为 4 次。
1	0	1	将 TGIA_3 的中断减少次数设定为 5 次。
1	1	0	将 TGIA_3 的中断减少次数设定为 6 次。
1	1	1	将 TGIA_3 的中断减少次数设定为 7 次。

表 12.39 通过 4VCOR2 ~ 4VCOR0 位设定的中断减少次数

bit2	bit1	bit0	说明
4VCOR2	4VCOR1	4VCOR0	
0	0	0	不减少 TCIV_4 的中断。
0	0	1	将 TCIV_4 的中断减少次数设定为 1 次。
0	1	0	将 TCIV_4 的中断减少次数设定为 2 次。
0	1	1	将 TCIV_4 的中断减少次数设定为 3 次。
1	0	0	将 TCIV_4 的中断减少次数设定为 4 次。
1	0	1	将 TCIV_4 的中断减少次数设定为 5 次。
1	1	0	将 TCIV_4 的中断减少次数设定为 6 次。
1	1	1	将 TCIV_4 的中断减少次数设定为 7 次。

12.3.26 定时器的中断减少次数计数器 (TITCNT)

TITCNT 是 8 位可读计数器。此模块有 1 个 TITCNT。在 TCNT_3 和 TCNT_4 停止计数后，TITCNT 保持原来的值。

位:	7	6	5	4	3	2	1	0
	—	3ACNT[2:0]			—	4VCNT[2:0]		
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7	—	0	R	保留位 读取值总是“0”。
6 ~ 4	3ACNT[2:0]	000	R	TGIA_3 中断计数器 如果将 TITCR 的 T3AEN 位置“1”，就在发生 TGIA_3 中断源时进行递增计数（加 1）。 [清除条件] • 当 TITCR 的 3ACOR2 ~ 3ACOR0 和 TITCNT 的 3ACNT2 ~ 3ACNT0 相同时 • 当 TITCR 的 T3AEN 位为“0”时 • 当 TITCR 的 3ACOR2 ~ 3ACOR0 为“0”时
3	—	0	R	保留位 读取值总是“0”。
2 ~ 0	4VCNT[2:0]	000	R	TCIV_4 中断计数器 如果将 TITCR 的 T4VEN 位置“1”，就在发生 TCIV_4 中断源时进行递增计数（加 1）。 [清除条件] • 当 TITCR 的 4VCOR2 ~ 4VCOR0 和 TITCNT 的 4VCNT2 ~ 4VCNT0 相同时 • 当 TITCR 的 T4VEN 位为“0”时 • 当 TITCR 的 4VCOR2 ~ 4VCOR0 为“0”时

【注】 要清除 TITCNT 的值时，必须将 TITCR 的 T3AEN 位和 T4VEN 位清“0”。

12.3.27 定时器的缓冲传送设定寄存器 (TBTER)

TBTER 是 8 位可读写寄存器，设定是否抑制用于互补 PWM 模式的缓冲寄存器 * 到暂存器的传送以及是否联动中断减少功能。此模块有 1 个 TBTER。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	BTE[1:0]	
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 2	—	全 0	R	保留位 读写值都为“0”。
1、0	BTE[1:0]	00	R/W	设定是否抑制用于互补 PWM 模式缓冲寄存器 * 到暂存器的传送以及是否联动中断减少功能， 详细内容请参照表 12.40。

【注】 * 对象缓冲寄存器：
TGRC_3、TGRD_3、TGRC_4、TGRD_4、TCBR

表 12.40 BTE1 位和 BTE0 位的设定

bit1	bit0	说明
BTE1	BTE0	
0	0	不抑制缓冲寄存器到暂存器的传送 *1，也不联动中断减少功能。
0	1	抑制缓冲寄存器到暂存器的传送。
1	0	缓冲寄存器到暂存器的传送联动中断减少功能 *2。
1	1	禁止设定。

【注】 *1 根据 TMDR 的 MD3 ~ MD0 的设定进行传送，详细内容请参照“12.4.8 互补 PWM 模式”。

*2 当禁止中断减少功能时（将定时器的中断减少设定寄存器 (TITCR) 的 T3AEN 位和 T4VEN 位置“0”或者将 TITCR 的减少次数设定位 (3ACOR 和 4VCOR) 置“0”时)，必须设定为缓冲传送不联动中断减少功能（将定时器的缓冲传送寄存器 (TBTER) 的 BTE1 置“0”）。
在禁止中断减少功能时，如果设定为缓冲传送联动中断减少功能，就不进行缓冲传送。

12.3.28 定时器的空载时间允许寄存器 (TDER)

TDER 是 8 位可读写寄存器。通道 3 有 1 个 TDER，能控制互补 PWM 模式的空载时间的生成。此模块有 1 个 TDER。必须在 TCNT 停止运行的状态下设定 TDER。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TDER
初始值:	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R/(W)

位	位名	初始值	R/W	说明
7 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	TDER	1	R/(W)	空载时间允许寄存器 设定是否生成空载时间。 0: 不生成空载时间 1: 生成空载时间* [清除条件] • 在 TDER 位为“1”的状态下读 TDER 位后给 TDER 位写“0”时

【注】 * 必须设定为 $TDDR \geq 1$ 。

12.3.29 定时器的波形控制寄存器 (TWCR)

TWCR 是 8 位可读写寄存器，用于控制在互补 PWM 模式中发生 TNCT_3 和 TNCT_4 的同步计数清除时的输出波形以及设定是否通过 TGRA_3 的比较匹配清除计数器。必须在 TCNT 停止运行的状态下设定 TWCR 的 CCE 位和 WRE 位。

位:	7	6	5	4	3	2	1	0
	CCE	—	—	—	—	—	—	WRE
初始值:	0*	0	0	0	0	0	0	0
R/W:	R/(W)	R	R	R	R	R	R	R/(W)

位	位名	初始值	R/W	说明
7	CCE	0*	R/(W)	比较匹配清除允许 设定在互补 PWM 模式中是否通过 TGRA_3 的比较匹配清除计数器。 0: 不通过 TGRA_3 比较匹配清除计数器 1: 通过 TGRA_3 的比较匹配清除计数器 [置位条件] • 在 CCE 位为“0”的状态下读 CCE 位后给 CCE 写“1”时
6~1	—	全 0	R	保留位 读写值都为“0”。
0	WRE	0	R/(W)	初始输出抑制允许 选择在互补 PWM 模式中发生同步计数清除时的输出波形。只有在互补 PWM 模式的波谷 Tb 区间发生同步清除时，才能通过此功能抑制初始输出。如果在其他区间发生同步清除，就输出 TOCR 寄存器设定的初始值，与 WRE 位的设定无关。即使在 TCNT_3 和 TCNT_4 开始计数器后的波谷 Tb 区间发生同步清除时，也输出 TOCR 寄存器设定的初始值。 有关互补 PWM 模式波谷的 Tb 区间，请参照图 12.40。 0: 输出 TOCR 寄存器设定的初始输出值 1: 抑制初始输出 [置位条件] • 在 WRE 位为“0”的状态下读 WRE 位后给 WRE 位写“1”时

【注】* 除了互补 PWM 模式 1 以外，不能置“1”。

12.3.30 和总线主控器的接口

定时器的计数器 (TCNT)、通用寄存器 (TGR)、定时器的副计数器 (TCNTS)、定时器的周期缓冲寄存器 (TCBR)、定时器的空载时间数据寄存器 (TDDR)、定时器的周期数据寄存器 (TCDR)、定时器的 A/D 转换开始请求控制寄存器 (TADCRC)、定时器的 A/D 转换开始请求周期设定寄存器 (TADCOR) 和定时器的 A/D 转换开始请求周期设定缓冲寄存器 (TADCOBR) 都是 16 位寄存器。因为和总线主控之间的数据总线为 16 位，所以能以 16 位为单位进行读写，而不能以 8 位为单位进行读写。必须总是以 16 位为单位进行存取。

上述以外的寄存器是 8 位寄存器。因为和 CPU 之间的数据总线为 16 位，所以既能以 16 位为单位进行读写，也能以 8 位为单位进行读写。

12.4 运行说明

12.4.1 基本运行

各通道有 TCNT 和 TGR。TCNT 进行递增计数，能进行自由运行、周期计数器运行或者外部事件计数运行。TGR 能分别用作输入捕捉寄存器或者输出比较寄存器。必须通过通用输入 / 输出端口设定此模块的外部引脚功能。

(1) 计数器的运行

如果将 TSTR 的 CST0 ~ CST4 位置“1”，对应通道的 TCNT 就开始计数运行。能用作自由运行计数器和周期计数器等。

(a) 计数运行的设定步骤例子

计数运行的设定步骤例子如图 12.4 所示。

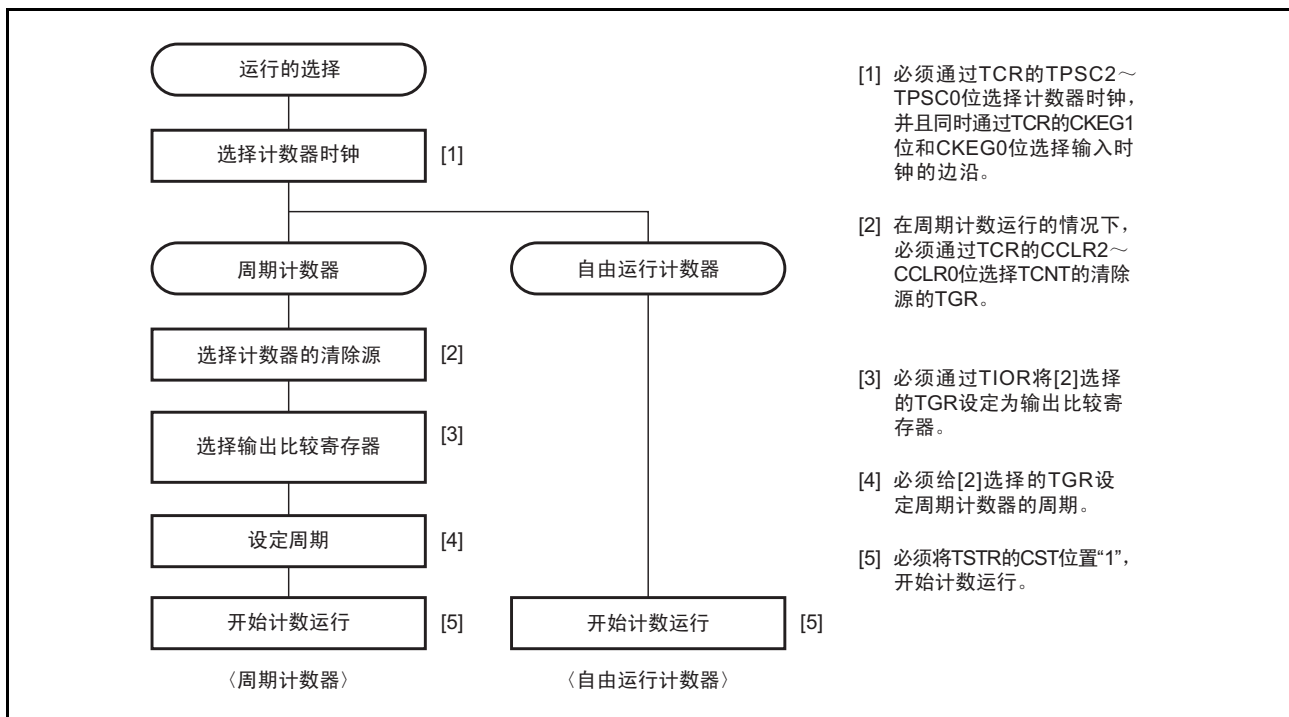


图 12.4 计数器运行的设定步骤例子

(b) 自由运行计数的运行和周期计数的运行

此模块的 TCNT 在复位后立即被全部设定为自由运行计数器。如果将 TSTR 的对应位置“1”，就作为自由运行计数器开始递增计数。如果 TCNT 发生上溢（H'FFFF→H'0000），就将 TSR 的 TCFV 位置“1”。此时，如果对应 TIER 的 TCIEV 位为“1”，此模块就请求中断。TCNT 在发生上溢后从“H'0000”开始继续递增计数。

自由运行计数器的运行如图 12.5 所示。

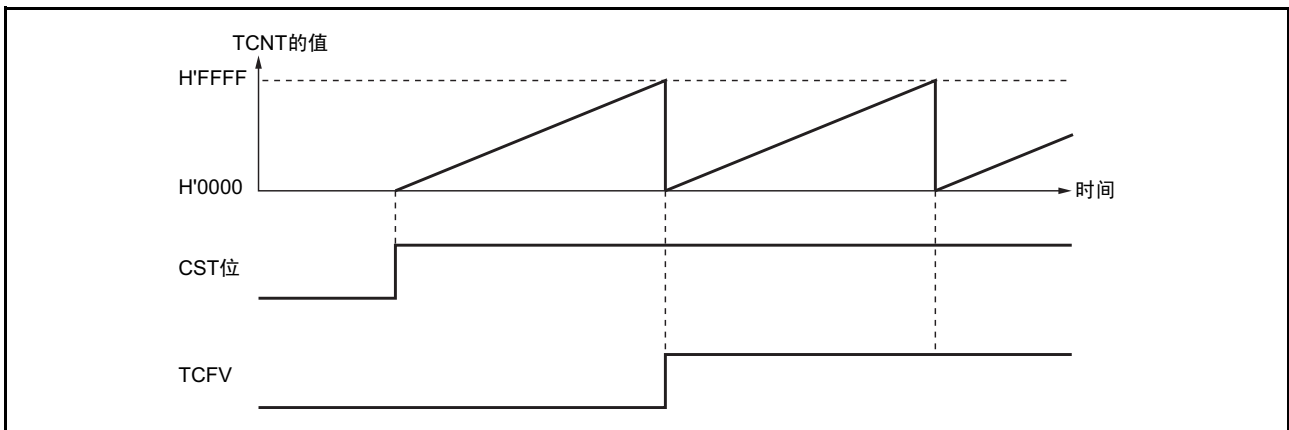


图 12.5 自由运行计数器的运行

当选择比较匹配作为 TCNT 的清除源时，对应通道的 TCNT 进行周期计数。将用于设定周期的 TGR 设定为输出比较寄存器，并且由 TCR 的 CCLR2 ~ CCLR0 位选择通过比较匹配进行计数器清除。如果在设定后将 TSTR 的对应位置“1”，就作为周期计数器开始递增计数。如果计数值和 TGR 的值相同，就将 TSR 的 TGF 位置“1”并且将 TCNT 清“H'0000”。

此时，如果对应 TIER 的 TGIE 位为“1”，此模块就请求中断。TCNT 在比较匹配后从“H'0000”开始继续递增计数。

周期计数器的运行如图 12.6 所示。

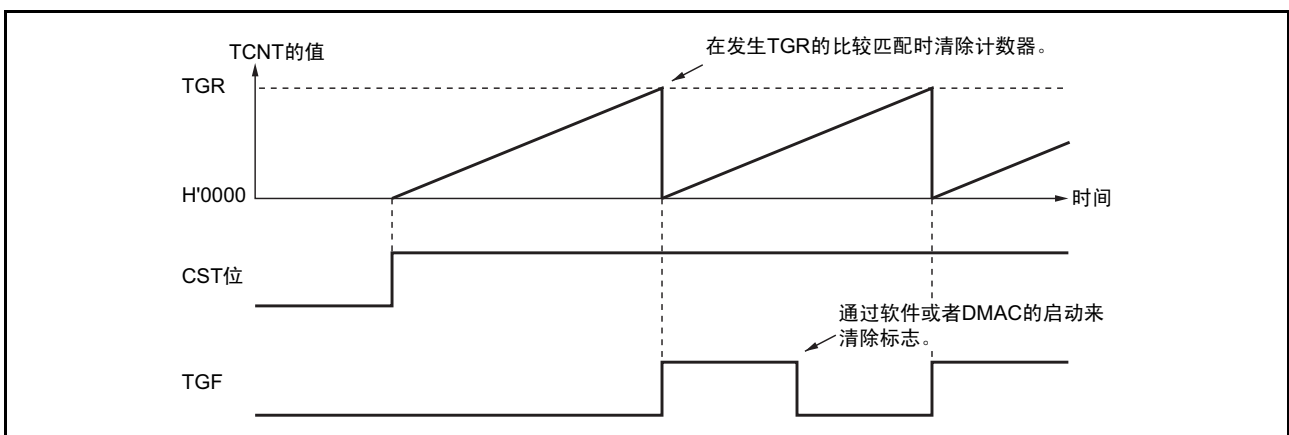


图 12.6 周期计数器的运行

(2) 通过比较匹配进行的波形输出功能

此模块能通过比较匹配从对应的输出引脚输出 Low 电平、High 电平或者进行交替输出。

(a) 通过比较匹配进行波形输出的设定步骤例子

通过比较匹配进行波形输出的设定步骤例子如图 12.7 所示。

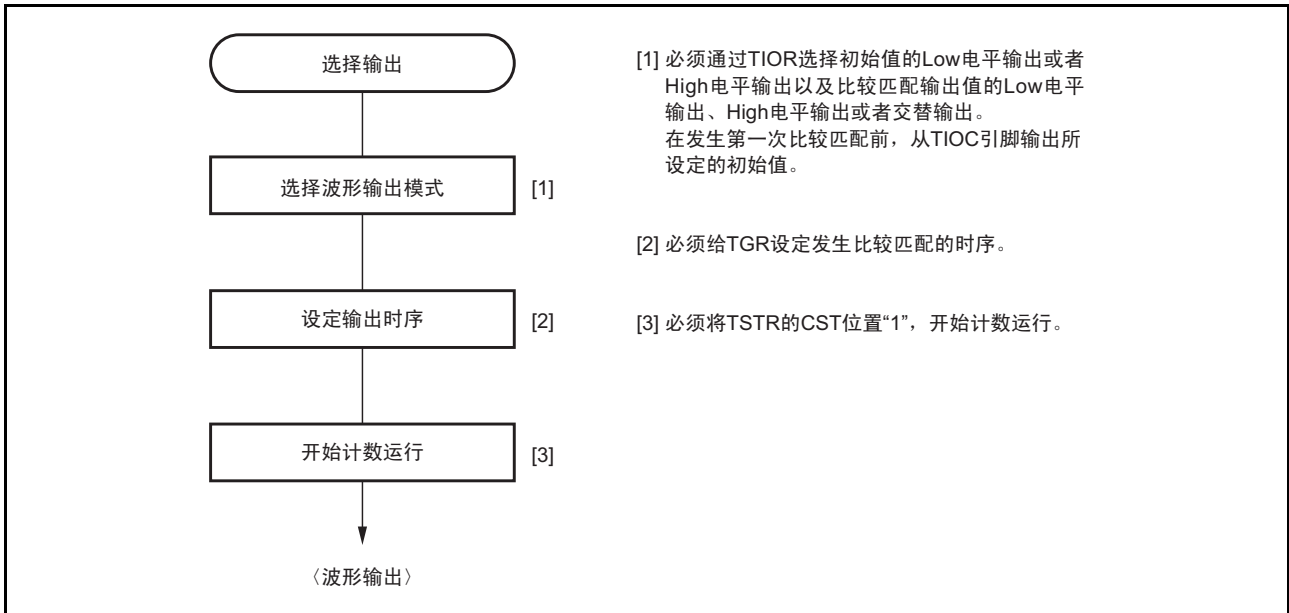


图 12.7 通过比较匹配进行波形输出的设定步骤例子

(b) 波形输出的运行例子

输出 Low 电平 /High 电平的运行例子如图 12.8 所示。

在此例子中，假设 TCNT 进行自由运行的计数，在比较匹配 A 时输出 High 电平，在比较匹配 B 时输出 Low 电平。如果设定的电平和引脚的电平相同，引脚的电平就不变。

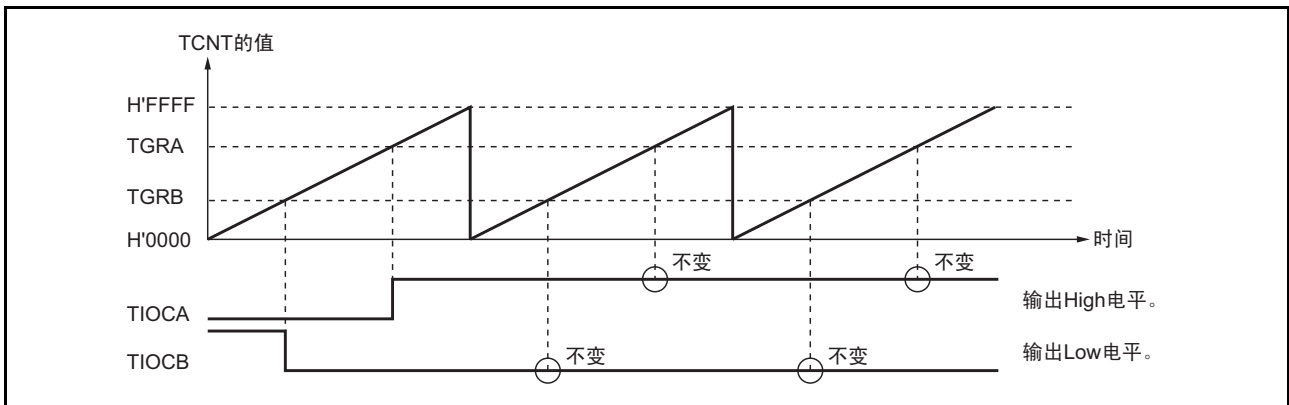


图 12.8 输出 Low 电平 /High 电平的运行例子

交替输出的运行例子如图 12.9 所示。

在此例子中，假设 TCNT 进行周期计数（通过比较匹配 B 进行计数器清除），并且比较匹配 A 和比较匹配 B 都进行交替输出。

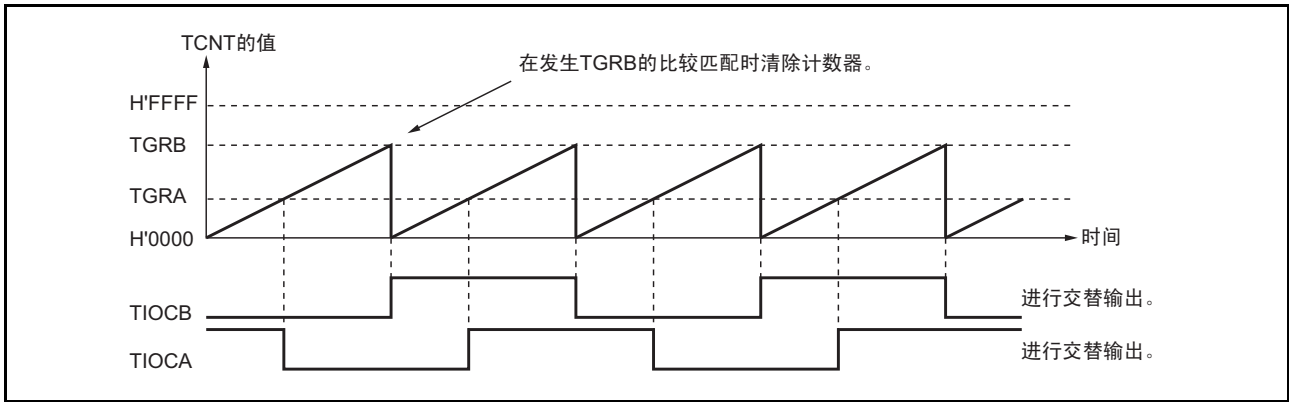


图 12.9 交替输出的运行例子

(3) 输入捕捉功能

能在检测到 TIOC 引脚的输入边沿后将 TCNT 的值传送到 TGR。

检测边沿可选择上升沿、下降沿或者双边沿，而在通道 0 和通道 1 中也能将其他通道的计数器输出时钟或者比较匹配信号作为输入捕捉源。

【注】如果在通道 0 和通道 1 中将其他通道的计数器输入时钟作为输入捕捉的输入引脚，就不能选择 P0φ/1 作为输入捕捉的计数器输入时钟，否则就不发生输入捕捉。

(a) 输入捕捉的设定步骤例子

输入捕捉的设定步骤例子如图 12.10 所示。

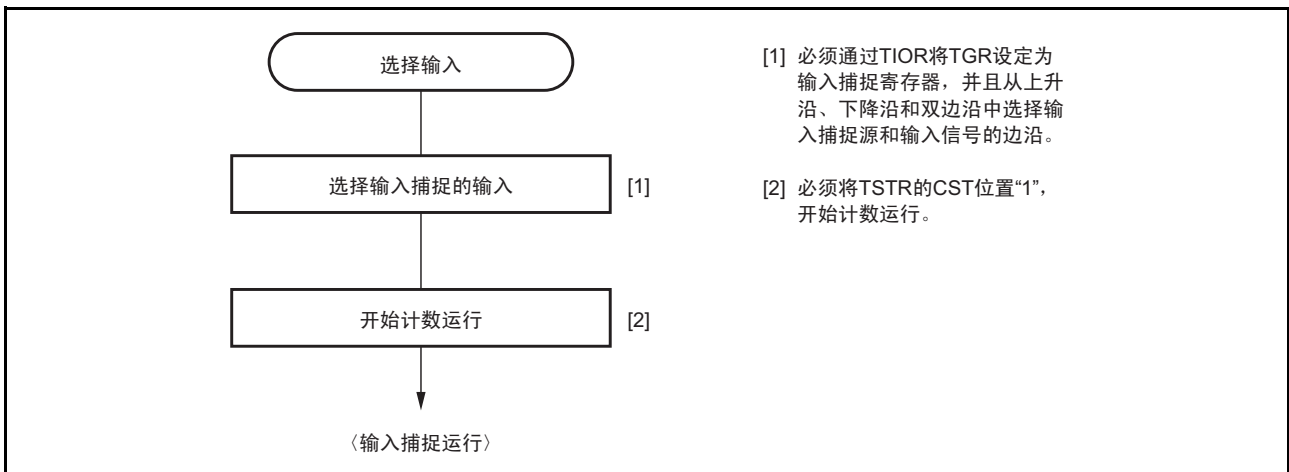


图 12.10 输入捕捉的设定步骤例子

(b) 输入捕捉的运行例子

输入捕捉的运行例子如图 12.11 所示。

在此例子中，假设选择上升 / 下降的双边沿作为 TIOCA 引脚的输入捕捉的输入边沿，选择下降沿作为 TIOCB 引脚的输入捕捉的输入边沿，并且在发生 TGRB 的输入捕捉时清除 TCNT 计数器。

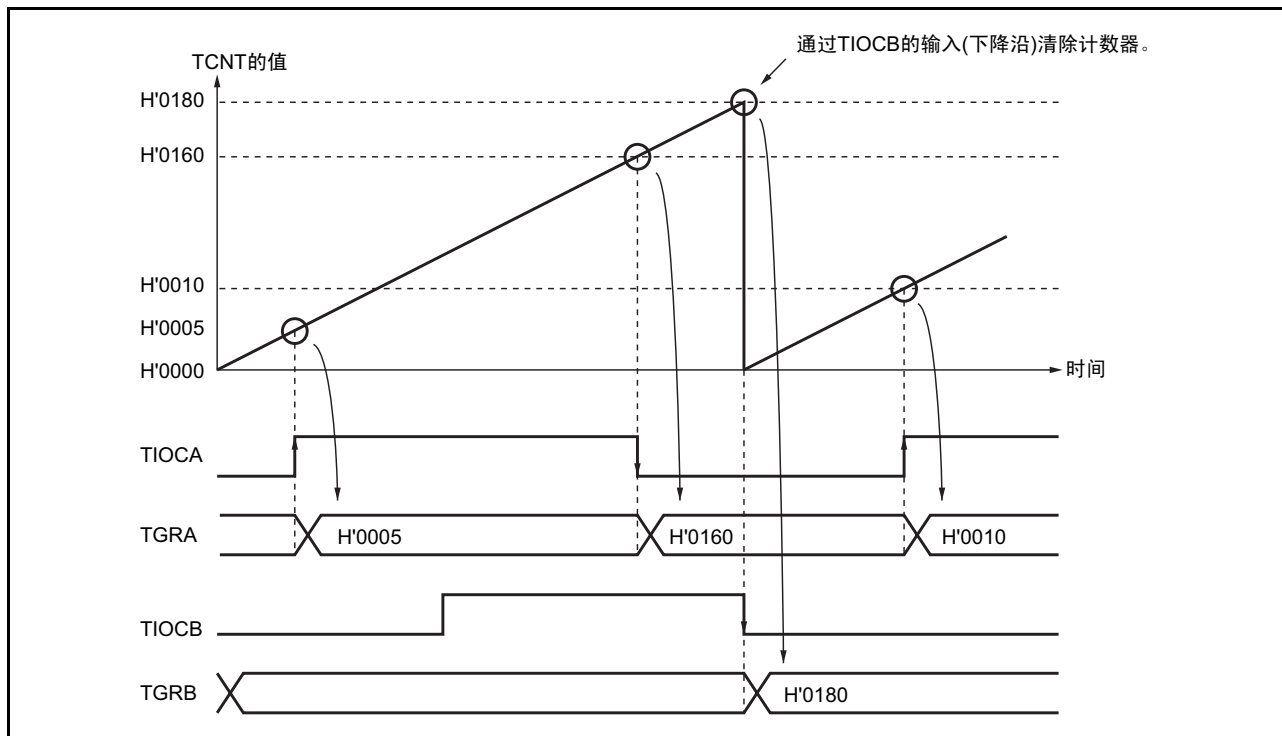


图 12.11 输入捕捉的运行例子

12.4.2 同步运行

能使用同步运行同时改写多个 TCNT 的值（同步预置），还能通过设定 TCR 同时清除多个 TCNT（同步清除）。

能通过同步运行，对 1 个时基增加要运行的 TGR 个数。

通道 0 ~ 4 都能设定为同步运行。

通道 5 不能进行同步运行。

(1) 同步运行的设定步骤例子

同步运行的设定步骤例子如图 12.12 所示。

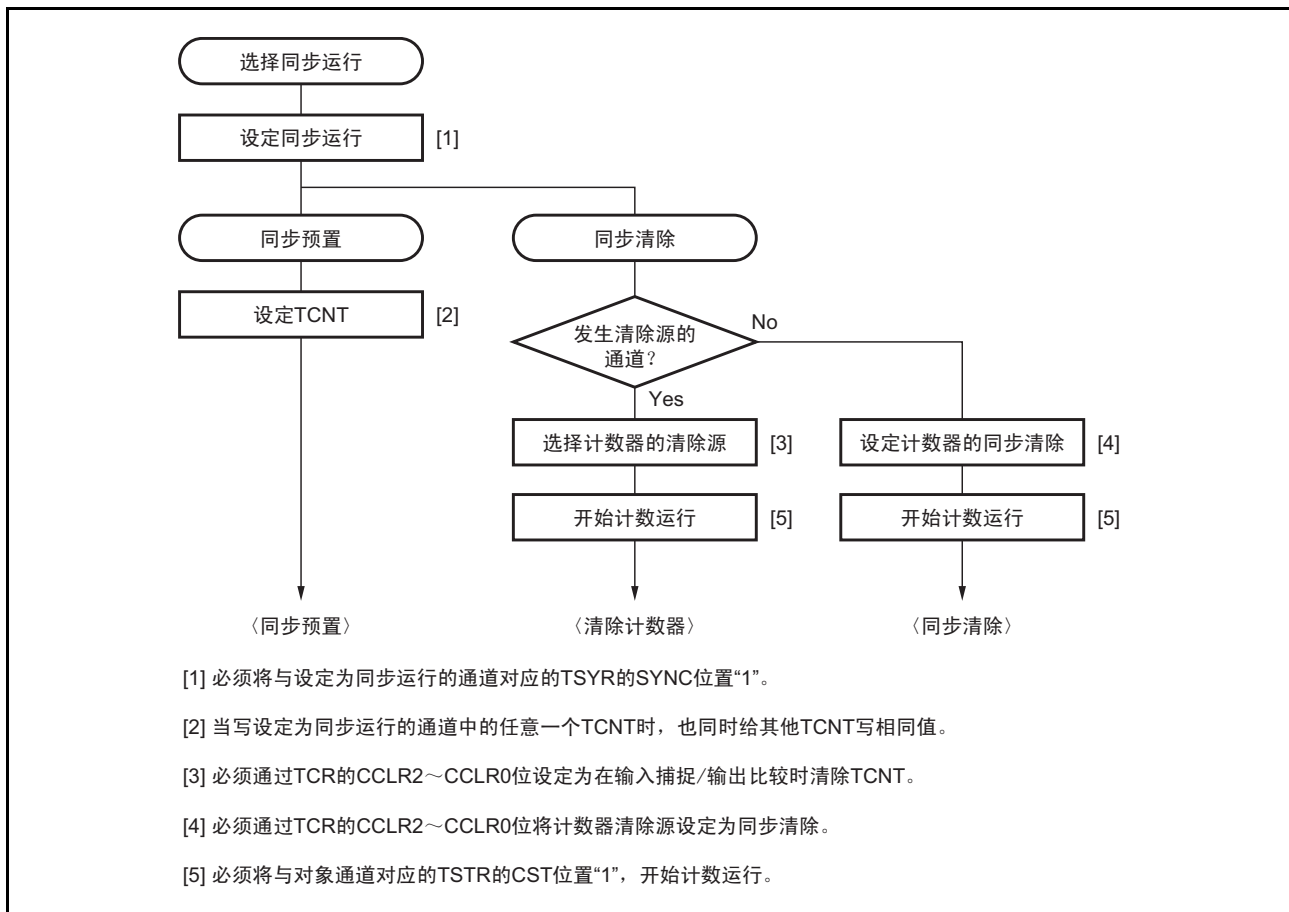


图 12.12 同步运行的设定步骤例子

(2) 同步运行的例子

同步运行的例子如图 12.13 所示。

在此例子中，假设将通道 0 ~ 2 设定为同步运行和 PWM 模式 1，将通道 0 的计数器清除源设定为 TGRB_0 的比较匹配，将通道 1 和通道 2 的计数器清除源设定为同步清除。

从 TIOC0A、TIOC1A、TIOC2A 引脚输出 3 相 PWM 波形。此时，通道 0 ~ 2 的 TCNT 进行同步预置并且通过 TGRB_0 的比较匹配进行同步清除，TGRB_0 设定的数据为 PWM 周期。

有关 PWM 模式，请参照“12.4.5 PWM 模式”。

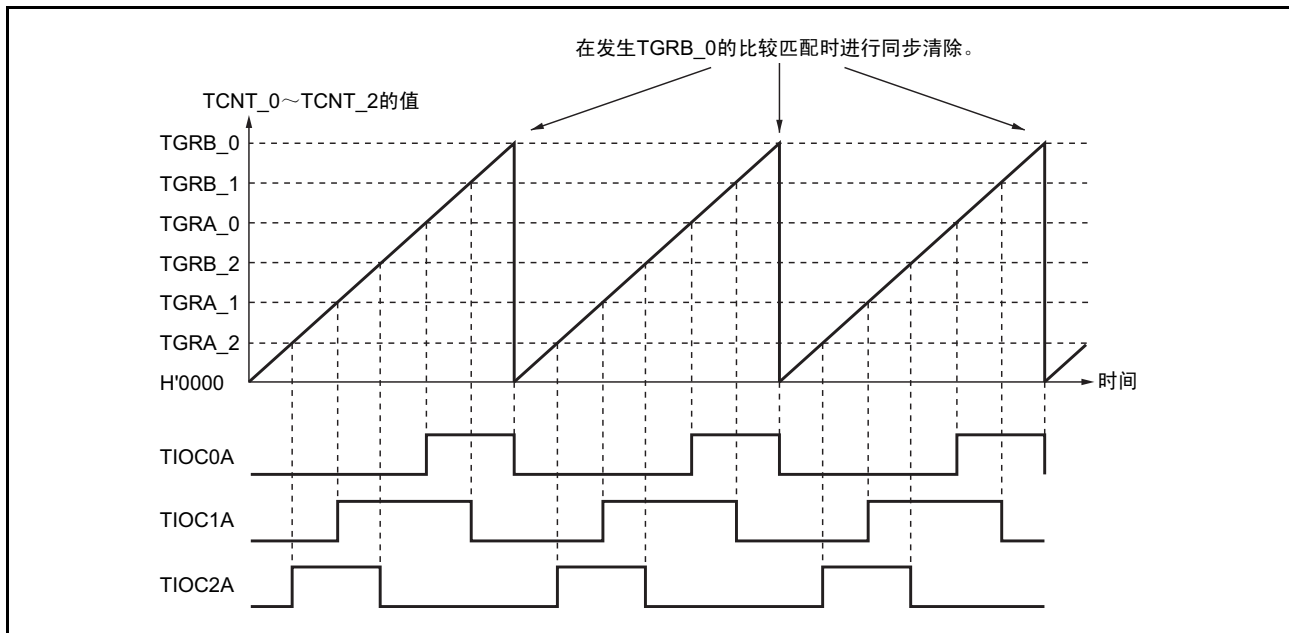


图 12.13 同步运行的例子

12.4.3 缓冲运行

缓冲运行是通道 0、3、4 具有的功能。能将 TGRC 和 TGRD 用作缓冲寄存器，在通道 0 中也能将 TGRF 用作缓冲寄存器。

将 TGR 设定为输入捕捉寄存器和比较匹配寄存器时的缓冲运行内容不同。

【注】 不能将 TGRE_0 设定为输入捕捉寄存器，TGRE_0 只用作比较匹配寄存器。

缓冲运行时的寄存器组合如表 12.41 所示。

表 12.41 寄存器的组合

通道	定时器的通用寄存器	缓冲寄存器
0	TGRA_0	TGRC_0
	TGRB_0	TGRD_0
	TGRE_0	TGRF_0
3	TGRA_3	TGRC_3
	TGRB_3	TGRD_3
4	TGRA_4	TGRC_4
	TGRB_4	TGRD_4

- TGR 为输出比较寄存器的情况
如果发生比较匹配，就将对应通道的缓冲寄存器的值传送到定时器的通用寄存器。
此运行如图 12.14 所示。

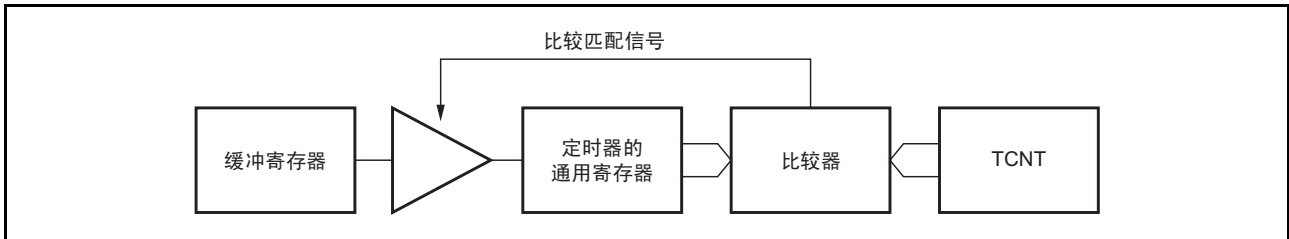


图 12.14 比较匹配的缓冲运行

- TGR 为输入捕捉寄存器的情况
如果发生输入捕捉，就在将 TCNT 的值传送到 TGR 的同时，将以前保存的 TGR 值传送到缓冲寄存器。
此运行如图 12.15 所示。

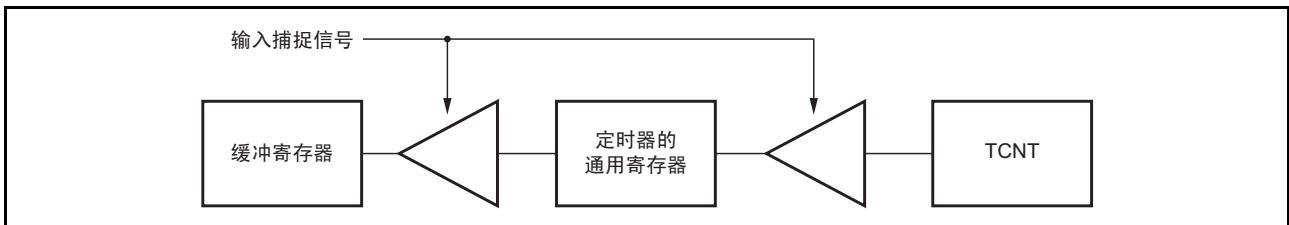


图 12.15 输入捕捉的缓冲运行

(1) 缓冲运行的设定步骤例子

缓冲运行的设定步骤例子如图 12.16 所示。

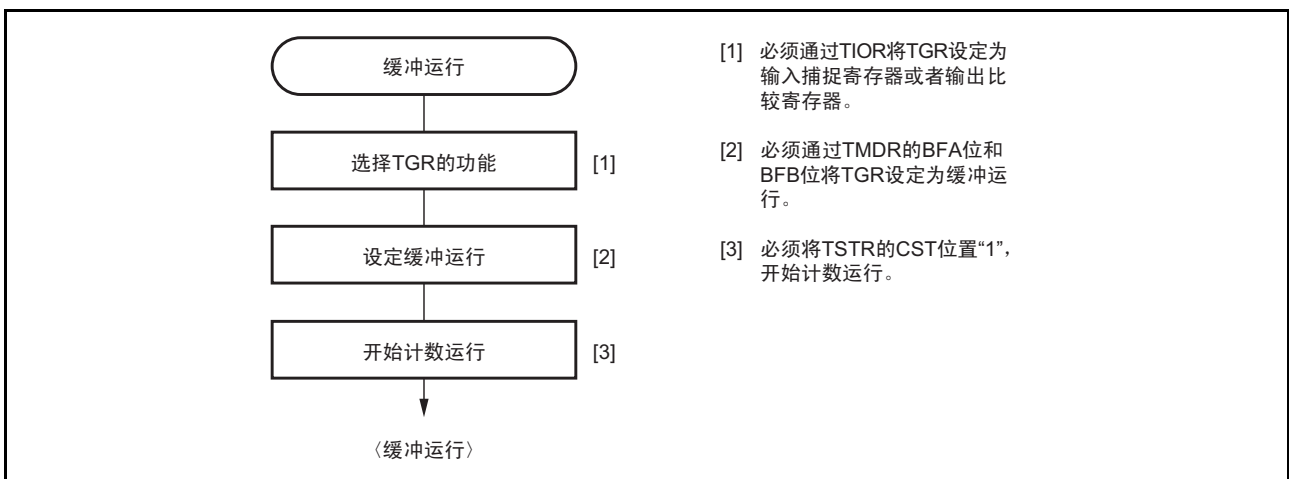


图 12.16 缓冲运行的设定步骤例子

(2) 缓冲运行的例子

(a) TGR 为输出比较寄存器的情况

将通道 0 设定为 PWM 模式 1 并且将 TGRA 和 TGRC 设定为缓冲运行时的运行例子如图 12.17 所示。在此例子中，假设通过比较匹配 B 清除 TCNT，在比较匹配 A 输出 High 电平，在比较匹配 B 时输出 Low 电平，将 TBTM 的 TTSA 位置“0”。

因为设定了缓冲运行，所以当发生比较匹配 A 时，就在输出发生变化的同时将缓冲寄存器 TGRC 的值传送到定时器的通用寄存器 TGRA。每当发生比较匹配 A 时，重复此操作。

有关 PWM 模式，请参照“12.4.5 PWM 模式”。

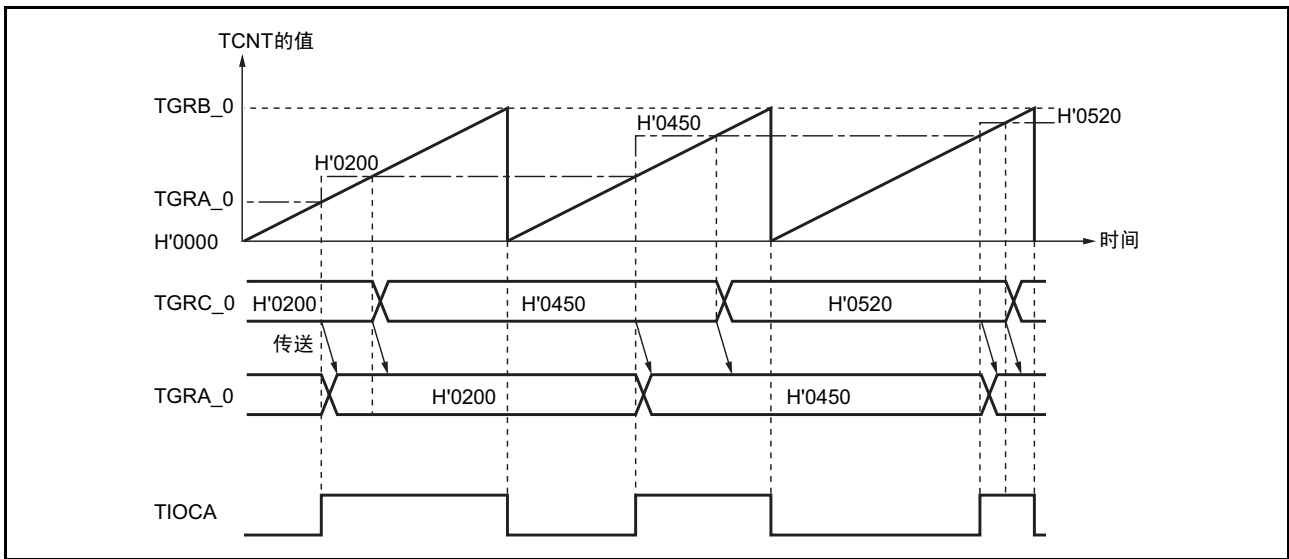


图 12.17 缓冲运行的例子 (1)

(b) TGR 为输入捕捉寄存器的情况

将 TGRA 设定为输入捕捉寄存器并且将 TGRA 和 TGRC 设定为缓冲运行时的运行例子如图 12.18 所示。

在 TGRA 的输入捕捉时清除 TCNT 计数器，选择上升沿 / 下降沿的双边沿为 TIOCA 引脚的输入捕捉的输入边沿。

因为设定了缓冲运行，所以在通过输入捕捉 A 将 TCNT 的值保存到 TGRA 的同时，将以前保存在 TGRA 的值传送到 TGRC。

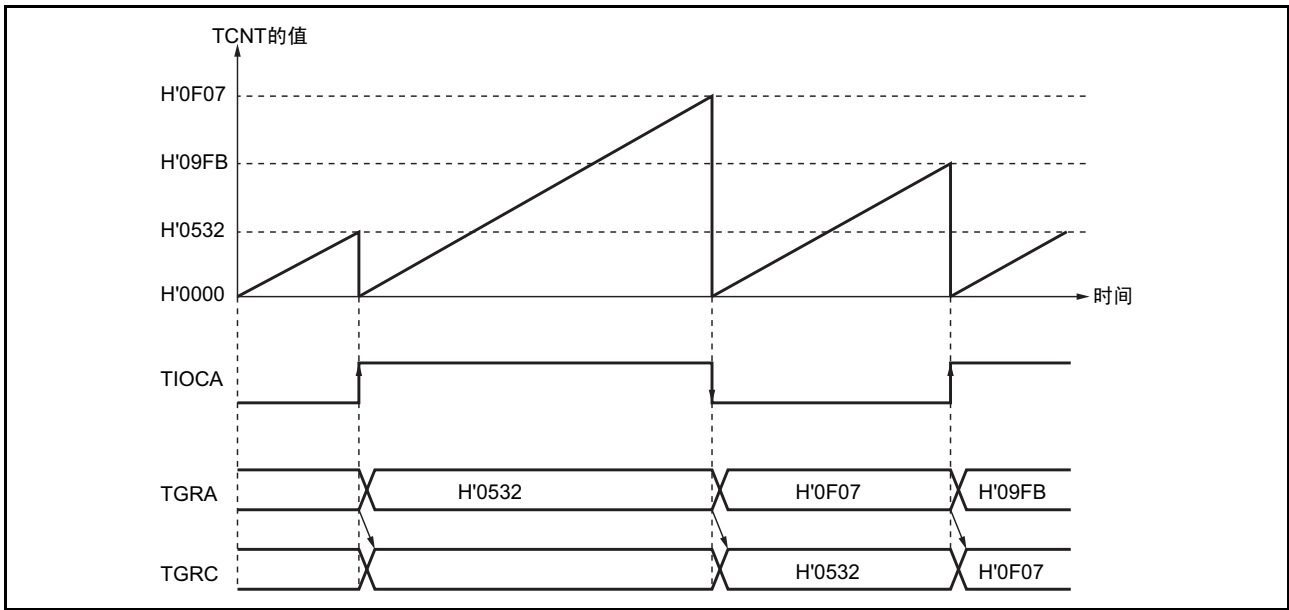


图 12.18 缓冲运行的例子 (2)

(3) 缓冲运行时的缓冲寄存器到定时器的通用寄存器的传送时序选择

能通过设定缓冲运行传送模式寄存器 (TBTM_0、TBTM_3 和 TBTM_4)，选择通道 0 为 PWM 模式 1、2 时以及通道 3 和通道 4 为 PWM 模式 1 时的缓冲寄存器到定时器的通用寄存器的传送时序。能选择在发生比较匹配时 (初始值) 或者在清除 TCNT 时进行缓冲传送。在此，清除 TCNT 时是指以下的任意一个条件成立时。

- 当 TCNT 发生上溢 (H'FFFF→H'0000) 时
- 在计数器运行过程中给 TCNT 写 “H'0000” 时
- 通过 TCR 的 CCLR2~CCLR0 位设定的清除源使 TCNT 变为 “H'0000” 时

【注】 必须在 TCNT 停止运行的状态下设定 TBTM。

将通道 0 设定为 PWM 模式 1 并且将 TGRA_0 和 TGRC_0 设定为缓冲运行时的运行例子如图 12.19 所示。在此例子中，通过比较匹配 B 清除 TCNT_0，在比较匹配 A 时输出 High 电平，在比较匹配 B 时输出 Low 电平，将 TBTM_0 的 TTSA 位置 “1”。

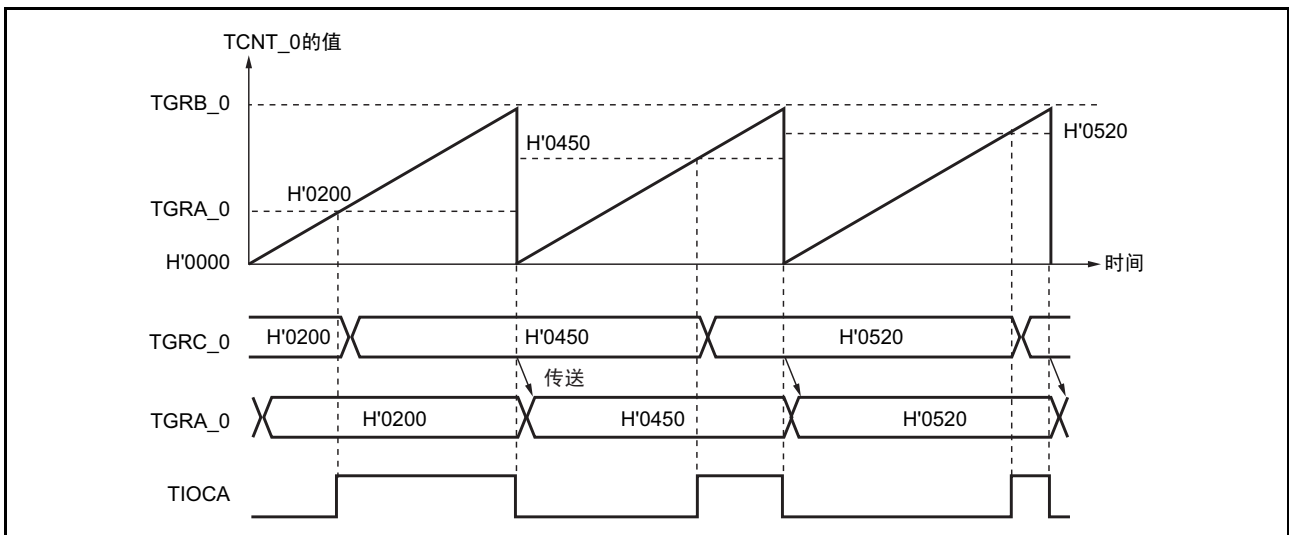


图 12.19 选择在清除 TCNT_0 时进行 TGRC_0 到 TGRA_0 的缓冲传送的运行例子

12.4.4 级联运行

级联运行是将 2 个通道的 16 位计数器连接为 32 位计数器的功能。

通过 TCR 的 TPSC2 ~ TPSC0 位将通道 1 的计数器时钟设定为“通过 TCNT_2 的上溢 / 下溢进行计数”，实现级联运行的功能。

只在低 16 位的 TCNT 为相位计数模式时发生下溢。

级联的组合如表 12.42 所示。

【注】 如果将通道 1 设定为相位计数模式，计数器时钟的设定无效，并且在相位计数模式中独立运行。

表 12.42 级联的组合

组合	高 16 位	低 16 位
通道 1 和通道 2	TCNT_1	TCNT_2

如果在级联运行时进行 TCNT_1 和 TCNT_2 的同时输入捕捉，就能通过输入捕捉控制寄存器（TICCR）进行设定，追加作为输入捕捉条件的输入引脚。有关级联时的输入捕捉，请参照“12.7.22 级联中的 TCNT_1 和 TCNT_2 的同时输入捕捉”。

TICCR 的设定值和输入捕捉的输入引脚的对应如表 12.43 所示。

表 12.43 TICCR 的设定值和输入捕捉的输入引脚的对应

对象输入捕捉	TICCR 的设定值	输入捕捉的输入引脚
从 TCNT_1 到 TGRA_1 的输入捕捉	I2AE 位为“0”（初始值）。	TIOC1A
	I2AE 位为“1”。	TIOC1A、TIOC2A
从 TCNT_1 到 TGRB_1 的输入捕捉	I2BE 位为“0”（初始值）。	TIOC1B
	I2BE 位为“1”。	TIOC1B、TIOC2B
从 TCNT_2 到 TGRA_2 的输入捕捉	I1AE 位为“0”（初始值）。	TIOC2A
	I1AE 位为“1”。	TIOC2A、TIOC1A
从 TCNT_2 到 TGRB_2 的输入捕捉	I1BE 位为“0”（初始值）。	TIOC2B
	I1BE 位为“1”。	TIOC2B、TIOC1B

(1) 级联运行的设定步骤例子

级联运行的设定步骤例子如图 12.20 所示。

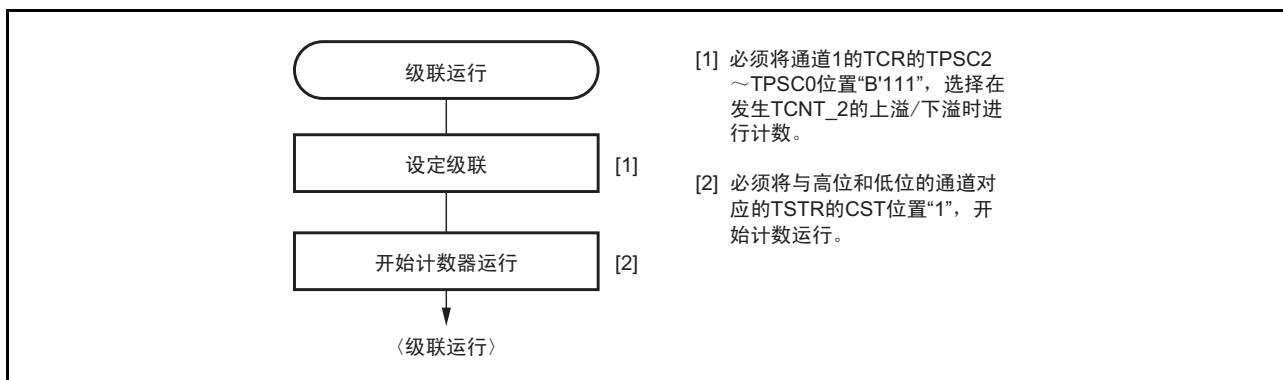


图 12.20 级联运行的设定步骤例子

(2) 级联运行的例子 (a)

TCNT_1 通过 TCNT_2 的上溢 / 下溢进行计数并且将通道 2 设定为相位计数模式时的运行如图 12.21 所示。TCNT_1 通过 TCNT_2 的上溢进行递增计数，通过 TCNT_2 的下溢进行递减计数。

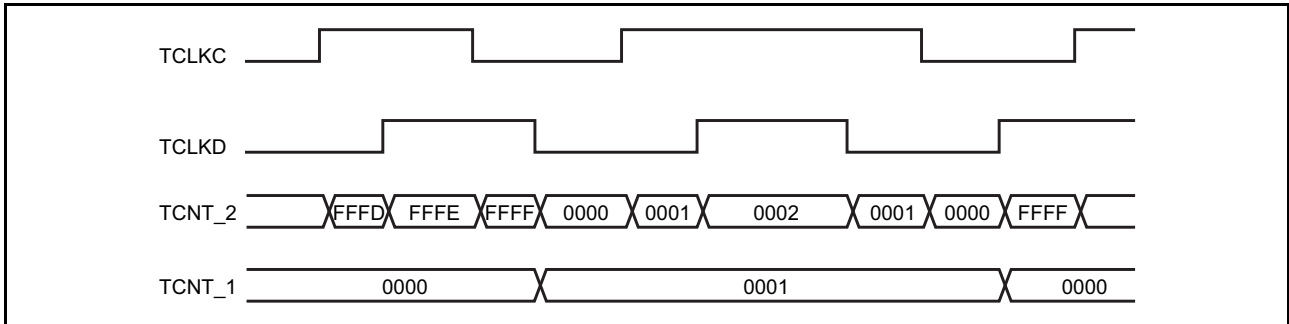


图 12.21 级联运行的例子 (a)

(3) 级联运行的例子 (b)

将 TCNT_1 和 TCNT_2 进行级联，并且通过将 TICCR 的 I2AE 位置“1”使 TIOC2A 引脚追加到 TGRA_1 的输入捕捉条件时的运行如图 12.22 所示。在此例子中，将 TIOR_1 的 IOA0 ~ IOA3 设定为在 TIOC1A 的上升沿进行输入捕捉，将 TIOR_2 的 IOA0 ~ IOA3 设定为在 TIOC2A 的上升沿进行输入捕捉。

此时，TIOC1A 和 TIOC2A 的上升沿被设定为 TGRA_1 的输入捕捉条件，TIOC2A 的上升沿为 TGRA_2 的输入捕捉条件。

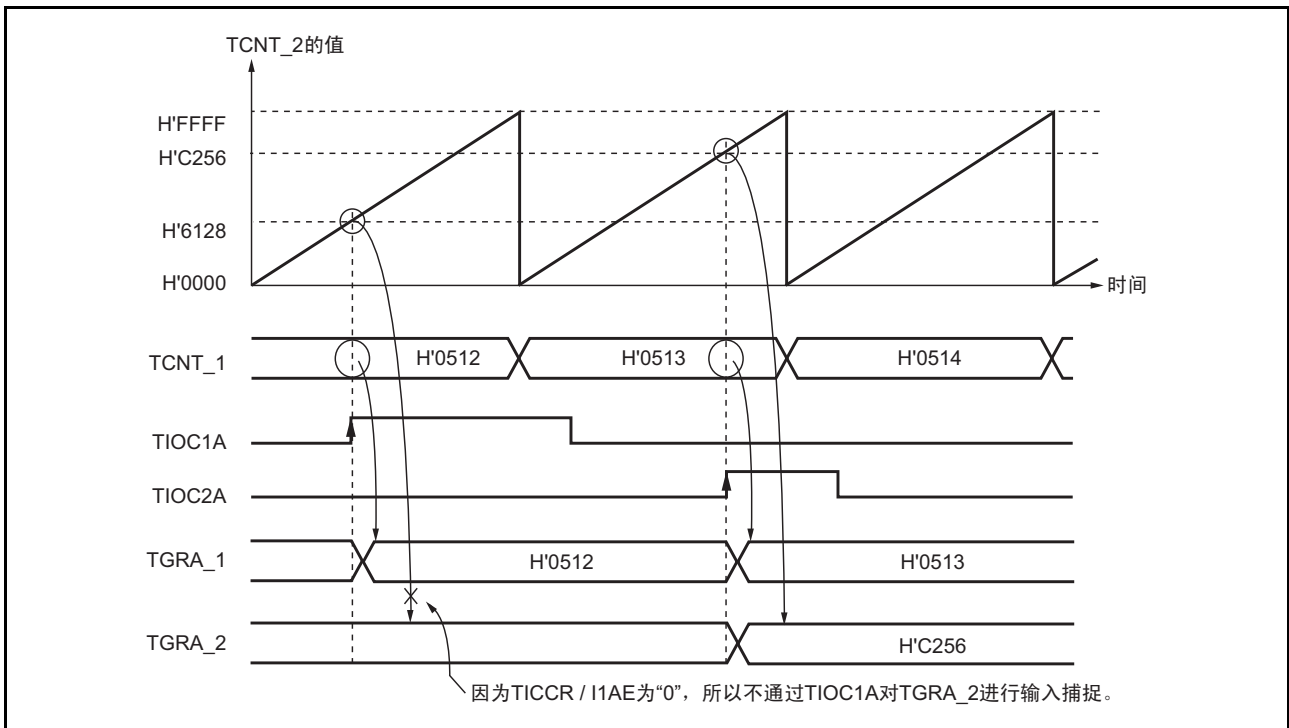


图 12.22 级联运行的例子 (b)

(4) 级联运行的例子 (c)

将 TCNT_1 和 TCNT_2 进行级联，并且通过将 TICCR 的 I2AE 位和 I1AE 位置“1”使 TIOC2A 引脚追加到 TGRA_1 的输入捕捉条件以及使 TIOC1A 引脚追加到 TGRA_2 的输入捕捉条件时的运行如图 12.23 所示。在此例子中，将 TIOR_1 和 TIOR_2 的 IOA0 ~ IOA3 都设定为在 TIOC1A 和 TIOC2A 的双边沿进行输入捕捉。此时，TIOC1A 和 TIOC2A 输入的 OR 为 TGRA_1 和 TGRA_2 的输入捕捉条件。

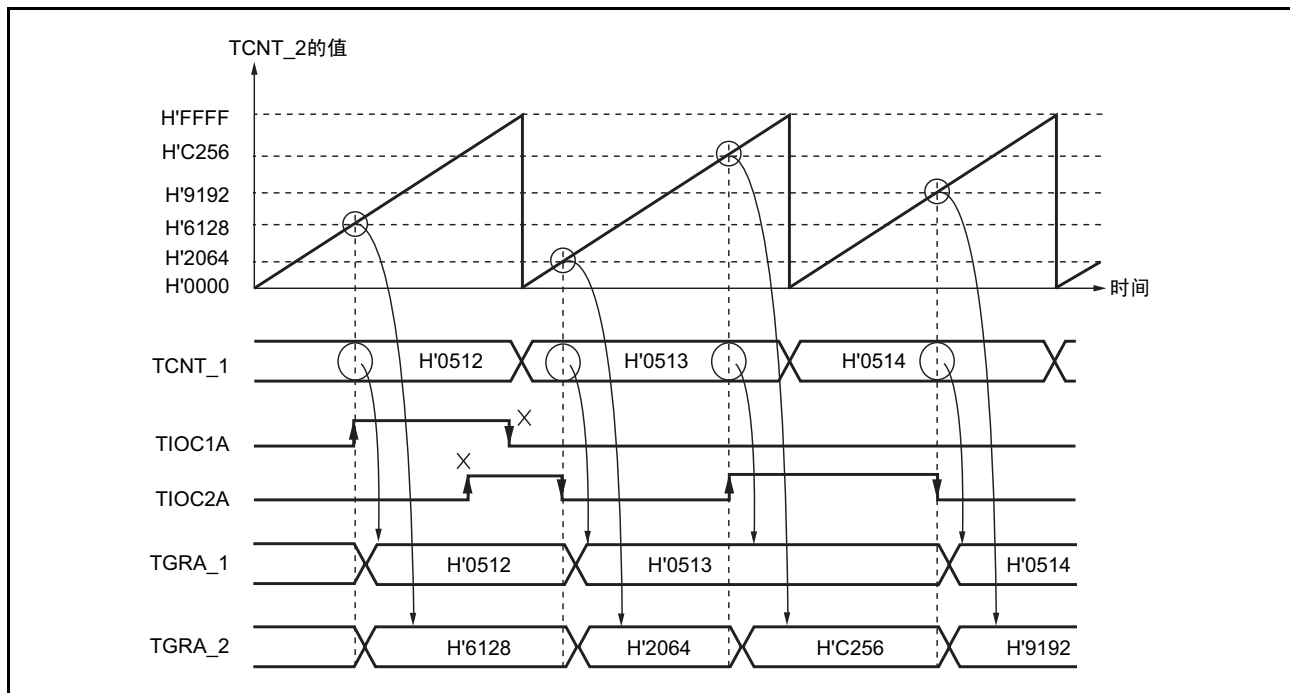


图 12.23 级联运行的例子 (c)

(5) 级联运行的例子 (d)

将 TCNT_1 和 TCNT_2 进行级联，并且通过将 TICCR 的 I2AE 位置“1”使 TIOC2A 引脚追加到 TGRA_1 的输入捕捉条件时的运行如图 12.24 所示。在此例子中，将 TIOR_1 的 IOA0 ~ IOA3 设定为在发生 TGRA_0 的比较匹配/输入捕捉时进行输入捕捉，将 TIOR_2 的 IOA0 ~ IOA3 设定为在 TIOC2A 的上升沿进行输入捕捉。

此时，因为将 TIOR_1 设定为在发生 TGRA_0 的比较匹配/输入捕捉时进行输入捕捉，所以即使将 TICCR 的 I2AE 位置“1”，TIOC2A 的边沿也不会成为 TGRA_1 的输入捕捉条件。

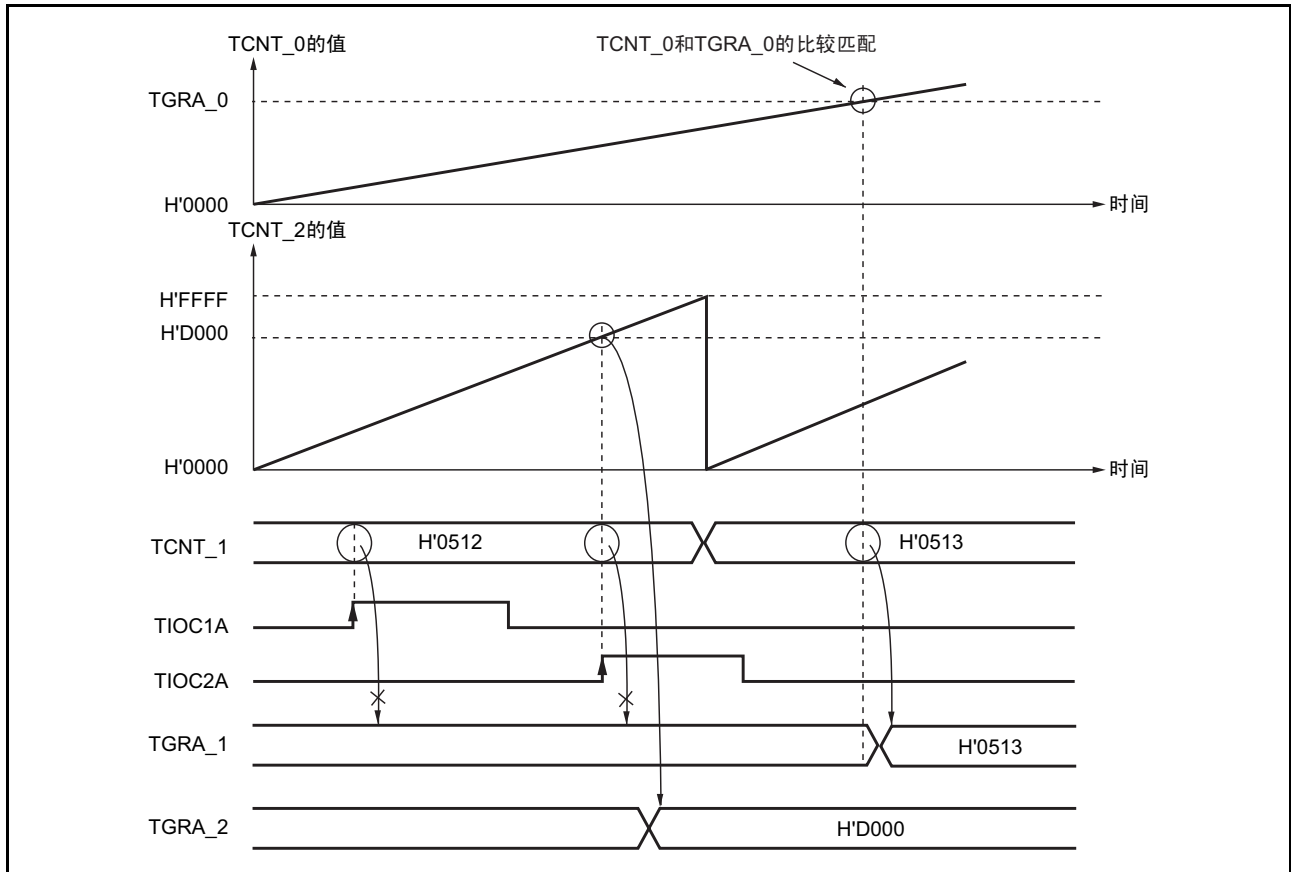


图 12.24 级联运行的例子 (d)

12.4.5 PWM 模式

PWM 模式是从输出引脚分别输出 PWM 波形的模式。各 TGR 比较匹配的输出电平可选择 Low 电平输出、High 电平输出或者交替输出。

能通过设定各 TGR，输出占空比为 0 ~ 100% 的 PWM 波形。

能通过将 TGR 的比较匹配设定为计时器清除源，给该寄存器设定周期。能将全部通道分别设定为 PWM 模式，也能进行同步运行。

PWM 模式有以下 2 种。

- PWM 模式 1

TGRA 和 TGRB、TGRC 和 TGRD 配对使用，从 TIOCA 引脚和 TIOCC 引脚生成 PWM 输出。通过比较匹配 A 和比较匹配 C 从 TIOCA 引脚和 TIOCC 引脚进行 TIOR 的 IOA3 ~ IOA0 和 IOC3 ~ IOC0 位指定的输出，并且通过比较匹配 B 和比较匹配 D 进行 TIOR 的 IOB3 ~ IOB0 和 IOD3 ~ IOD0 位指定的输出。初始输出值为 TGRA 和 TGRC 的设定值。当配对使用的 TGR 的设定值相同时，即使发生比较匹配，输出值也不变。

在 PWM 模式 1 中，能进行最多 8 相的 PWM 输出。

- PWM 模式 2

将 1 个 TGR 用作周期寄存器而将其他 TGR 用作占空比寄存器，生成 PWM 输出。通过比较匹配进行 TIOR 指定的输出，并且通过同步寄存器的比较匹配进行计数器清除，各引脚的输出值为 TIOR 设定的初始值。如果周期寄存器和占空比寄存器的设定值相同，即使发生比较匹配，输出值也不变。

在 PWM 模式 2 中，能通过和同步运行的并用，进行最多 8 相的 PWM 输出。

PWM 输出引脚和寄存器的对应如表 12.44 所示。

表 12.44 各 PWM 输出的寄存器和输出引脚

通道	寄存器	输出引脚	
		PWM 模式 1	PWM 模式 2
0	TGRA_0	TIOC0A	TIOC0A
	TGRB_0		TIOC0B
	TGRC_0	TIOC0C	TIOC0C
	TGRD_0		TIOC0D
1	TGRA_1	TIOC1A	TIOC1A
	TGRB_1		TIOC1B
2	TGRA_2	TIOC2A	TIOC2A
	TGRB_2		TIOC2B
3	TGRA_3	TIOC3A	不能设定
	TGRB_3		
	TGRC_3	TIOC3C	
	TGRD_3		
4	TGRA_4	TIOC4A	
	TGRB_4		
	TGRC_4	TIOC4C	
	TGRD_4		

【注】 在 PWM 模式 2 中，不能对已设定周期的 TGR 进行 PWM 输出。

(1) PWM 模式的设定步骤例子

PWM 模式的设定步骤例子如图 12.25 所示。

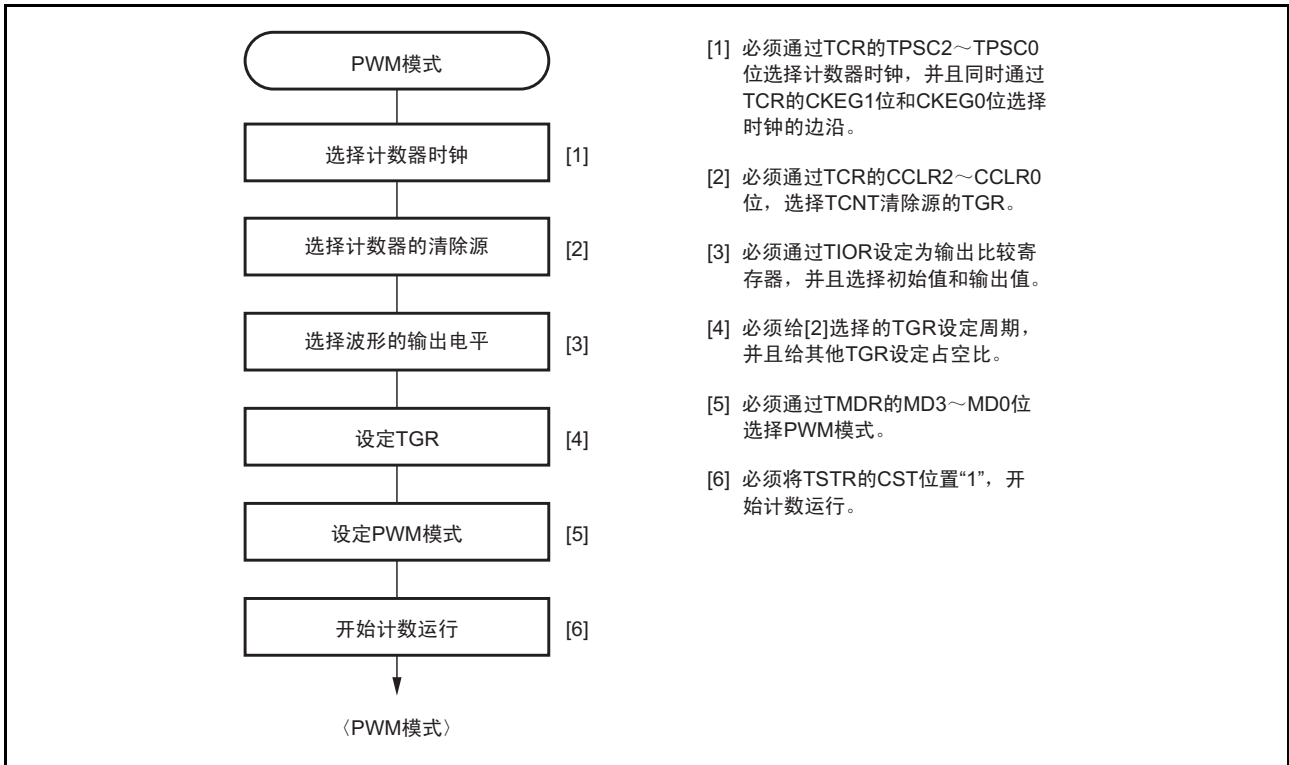


图 12.25 PWM 模式的设定步骤例子

(2) PWM 模式的运行例子

PWM 模式 1 的运行例子如图 12.26 所示。

在此例子中，假设将 TGRA 的比较匹配设定为 TCNT 的清除源，将 TGRA 的初始输出值和输出值置“0”，将 TGRB 的输出值置“1”。

此时，TGRA 的设定值为周期，TGRB 的设定值为占空比。

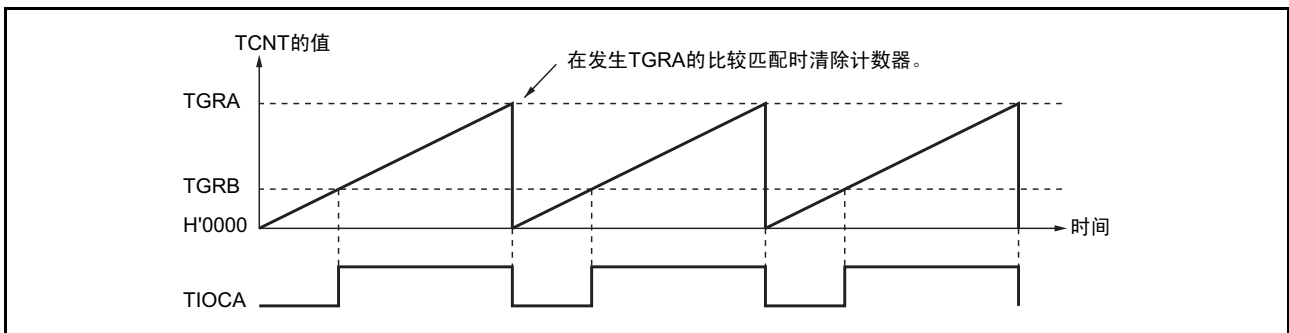


图 12.26 PWM 模式的运行例子

PWM 模式 2 的运行例子如图 12.27 所示。

在此例子中，假设使通道 0 和 1 同步运行，将 TGRB_1 的比较匹配设定为 TCNT 的清除源，将其他 TGR (TGRA_0 ~ TGRD_0、TGRA_1) 的初始输出值置“0”，将输出值置“1”，输出 5 相的 PWM 波形。

此时，TGR1B 的设定值为周期，其他 TGR 的设定值为占空比。

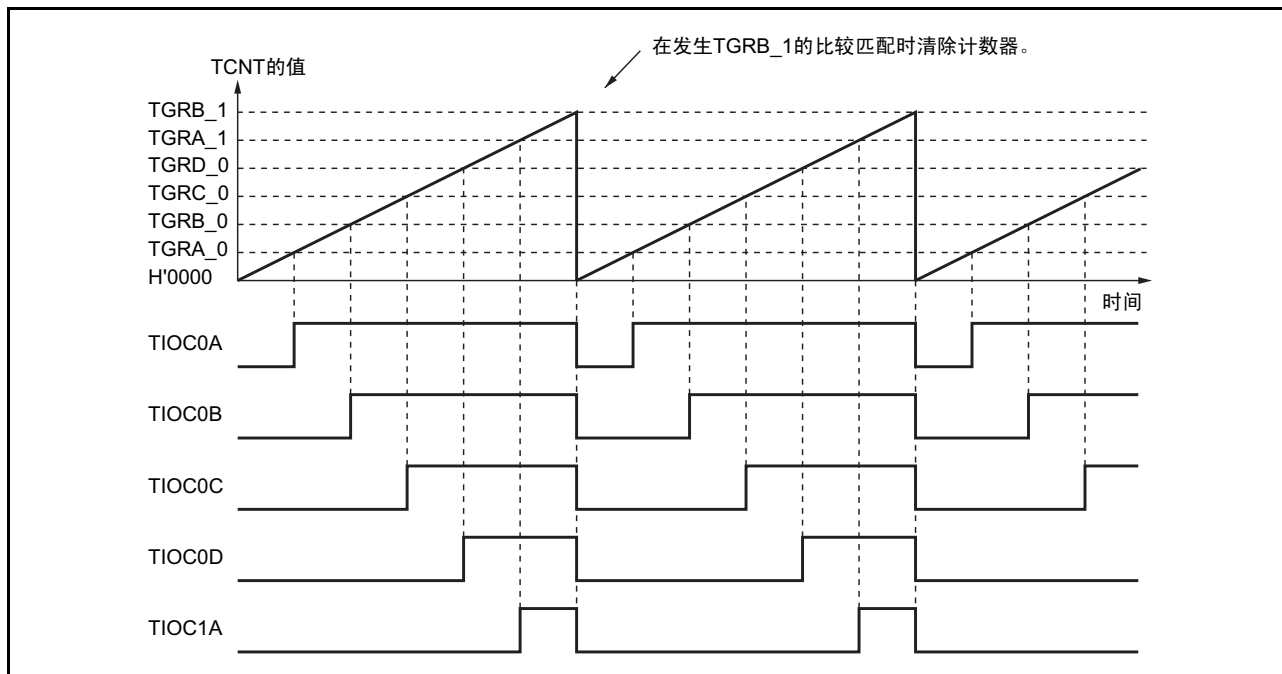


图 12.27 PWM 模式的运行例子

在 PWM 模式中，占空比为 0% 和 100% 的 PWM 波形输出例子如图 12.28 所示。

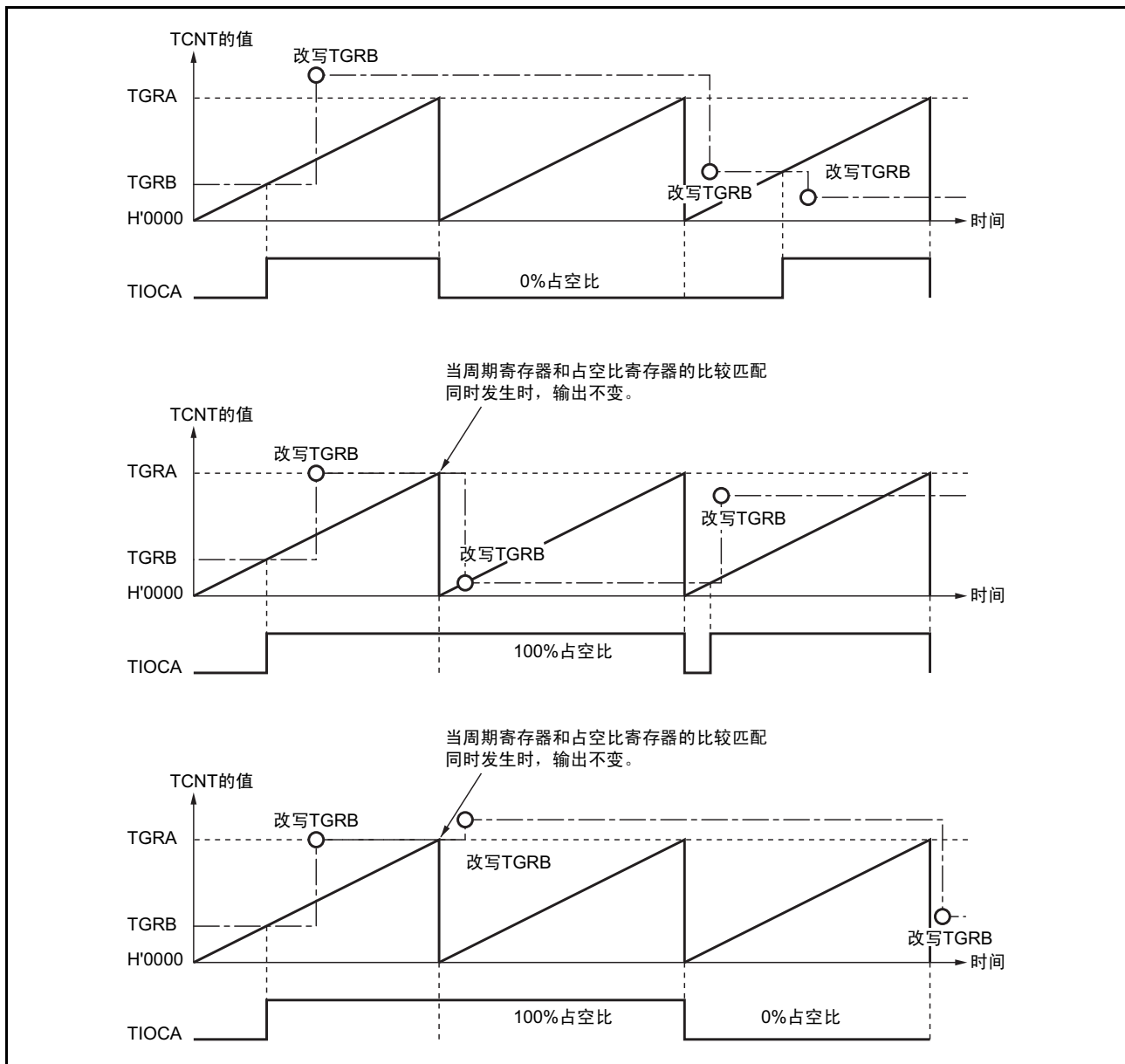


图 12.28 PWM 模式的运行例子

12.4.6 相位计数模式

在相位计数模式中，通过设定通道 1 和通道 2，检测 2 个外部时钟输入的相位差，并且 TCNT 进行递增 / 递减计数。

如果设定为相位计数模式，就选择外部时钟作为计数器输入时钟，并且 TCNT 作为递增 / 递减计数器运行，与 TCR 的 TPSC2 ~ TPSC0 位、CKEG1 位和 CKEG0 位的设定无关。但是，因为 TCR 的 CCLR1、CCLR0 位、TIOR、TIER、TGR 的功能有效，所以能使用输入捕捉 / 比较匹配功能和中断功能。

能用作 2 相编码器脉冲的输入。

如果在 TCNT 进行递增计数时发生上溢，就将 TSR 的 TCFV 标志置位；如果在 TCNT 进行递减计数时发生下溢，就将 TCFU 标志置位。

TSR 的 TCFD 位是计数方向标志。能通过读 TCFD 标志，确认 TCNT 是在进行递增计数还是在进行递减计数。

外部时钟引脚和通道的对应如表 12.45 所示。

表 12.45 相位计数模式的时钟输入引脚

通道	外部时钟引脚	
	A 相	B 相
将通道 1 设定为相位计数模式的情况	TCLKA	TCLKB
将通道 2 设定为相位计数模式的情况	TCLKC	TCLKD

(1) 相位计数模式的设定步骤例子

相位计数模式的设定步骤例子如图 12.29 所示。

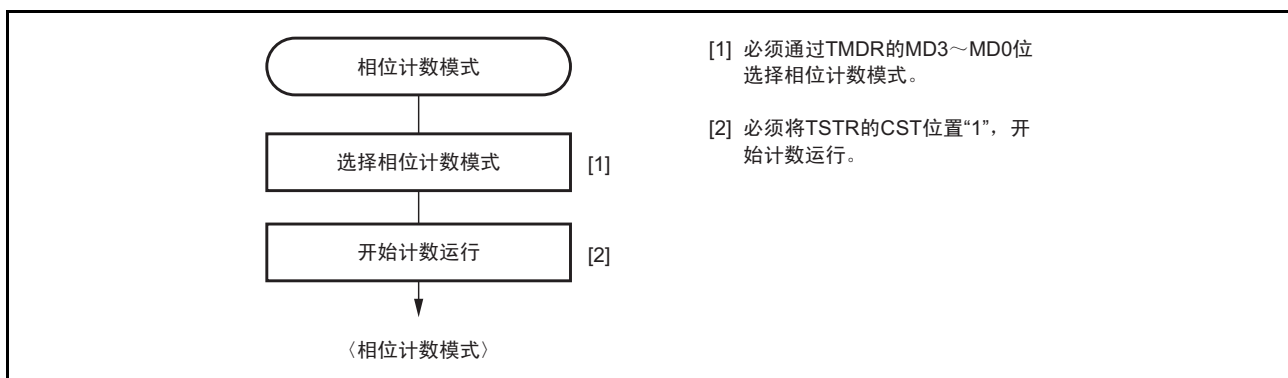


图 12.29 相位计数模式的设定步骤例子

(2) 相位计数模式的运行例子

在相位计数模式中，TCNT 通过 2 个外部时钟的相位差进行递增计数或者递减计数。根据计数条件，有 4 种模式。

(a) 相位计数模式 1

相位计数模式 1 的运行例子如图 12.30 所示，TCNT 的递增 / 递减计数条件如表 12.46 所示。

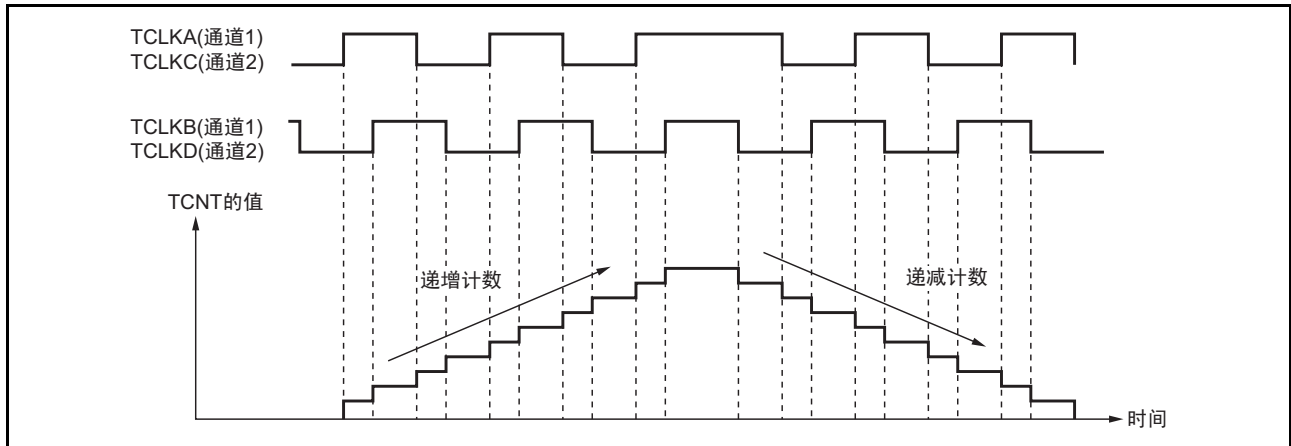


图 12.30 相位计数模式 1 的运行例子

表 12.46 相位计数模式 1 的递增 / 递减计数条件

TCLKA (通道 1) TCLKC (通道 2)	TCLKB (通道 1) TCLKD (通道 2)	运行内容
High 电平	上升沿	递增计数
Low 电平	下降沿	
上升沿	Low 电平	
下降沿	High 电平	
High 电平	下降沿	递减计数
Low 电平	上升沿	
上升沿	High 电平	
下降沿	Low 电平	

【符号说明】

上升沿

下降沿

(b) 相位计数模式 2

相位计数模式 2 的运行例子如图 12.31 所示，TCNT 的递增 / 递减计数条件如表 12.47 所示。

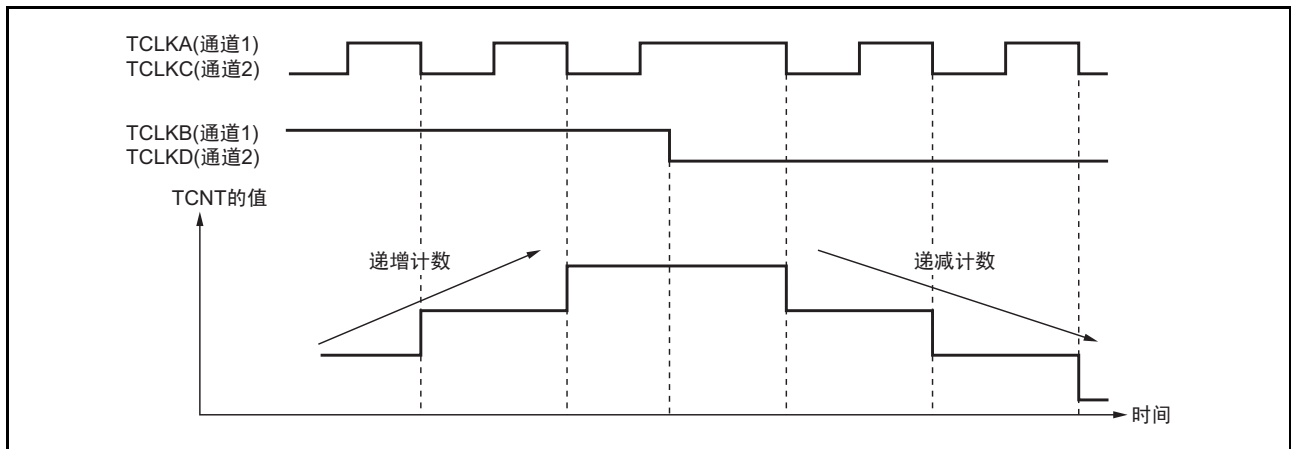


图 12.31 相位计数模式 2 的运行例子

表 12.47 相位计数模式 2 的递增 / 递减计数条件

TCLKA (通道 1) TCLKC (通道 2)	TCLKB (通道 1) TCLKD (通道 2)	运行内容
High 电平	\uparrow	不计数 (Don't care)
Low 电平	\downarrow	不计数 (Don't care)
\uparrow	Low 电平	不计数 (Don't care)
\downarrow	High 电平	递增计数
High 电平	\downarrow	不计数 (Don't care)
Low 电平	\uparrow	不计数 (Don't care)
\uparrow	High 电平	不计数 (Don't care)
\downarrow	Low 电平	递减计数

【符号说明】

\uparrow : 上升沿

\downarrow : 下降沿

(c) 相位计数模式 3

相位计数模式 3 的运行例子如图 12.32 所示，TCNT 的递增 / 递减计数条件如表 12.48 所示。

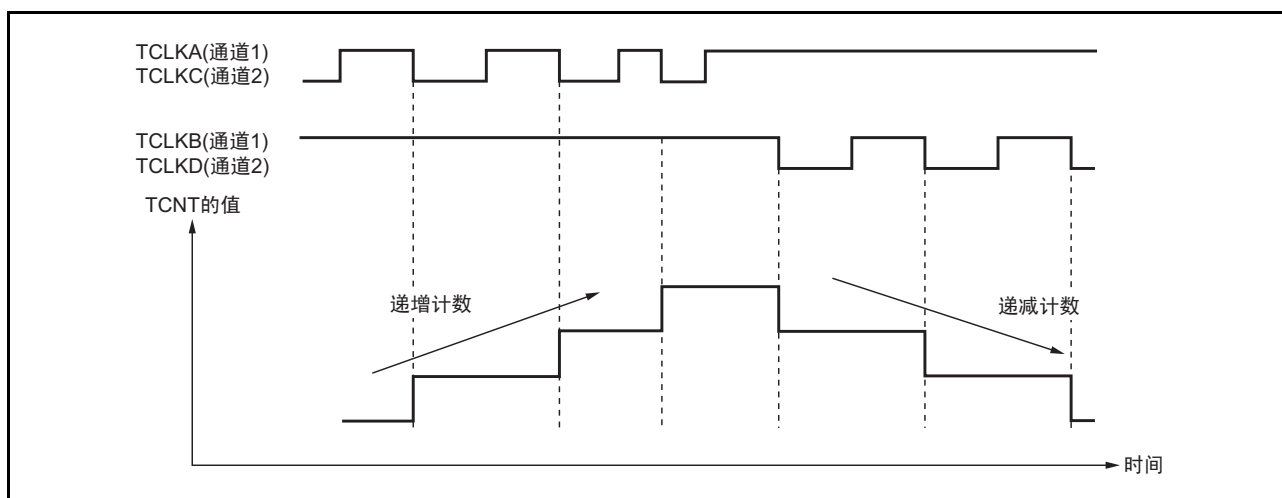


图 12.32 相位计数模式 3 的运行例子

表 12.48 相位计数模式 3 的递增 / 递减计数条件

TCLKA (通道 1) TCLKC (通道 2)	TCLKB (通道 1) TCLKD (通道 2)	运行内容
High 电平	上升沿	不计数 (Don't care)
Low 电平	下降沿	不计数 (Don't care)
上升沿	Low 电平	不计数 (Don't care)
下降沿	High 电平	递增计数
High 电平	下降沿	递减计数
Low 电平	上升沿	不计数 (Don't care)
上升沿	High 电平	不计数 (Don't care)
下降沿	Low 电平	不计数 (Don't care)

【符号说明】

上升沿

下降沿

(d) 相位计数模式 4

相位计数模式 4 的运行例子如图 12.33 所示，TCNT 的递增 / 递减计数条件如表 12.49 所示。

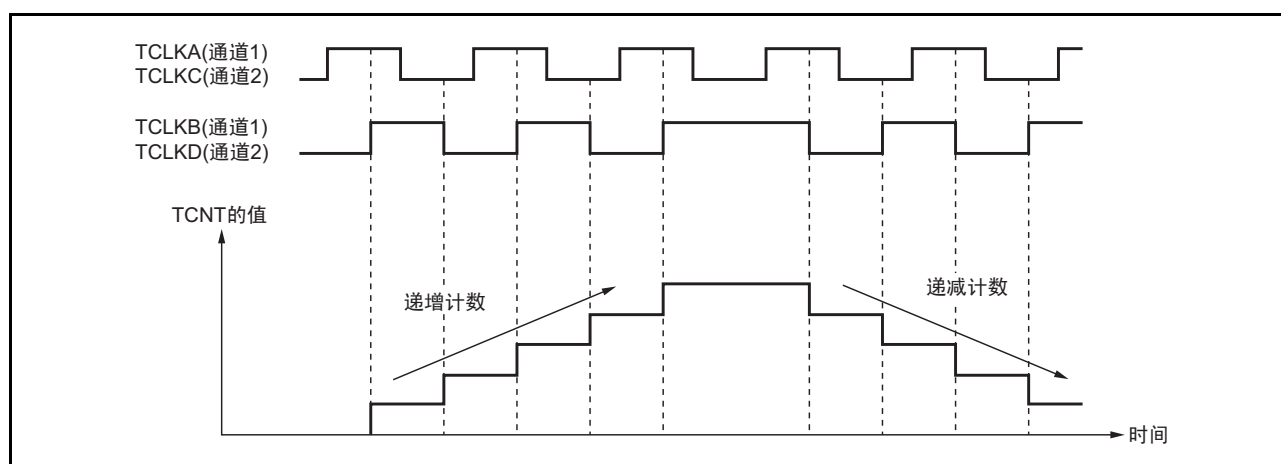


图 12.33 相位计数模式 4 的运行例子

表 12.49 相位计数模式 4 的递增 / 递减计数条件

TCLKA (通道 1) TCLKC (通道 2)	TCLKB (通道 1) TCLKD (通道 2)	运行内容
High 电平	上升沿	递增计数
Low 电平	下降沿	
上升沿	Low 电平	不计数 (Don't care)
下降沿	High 电平	
High 电平	下降沿	递减计数
Low 电平	上升沿	
上升沿	High 电平	不计数 (Don't care)
下降沿	Low 电平	

【符号说明】

上升沿

下降沿

(3) 相位计数模式的应用例子

将通道 1 设定为相位计数模式，并且在与通道 0 组合输入伺服马达 2 相编码器脉冲后检测位置或者速度的例子如图 12.34 所示。

将通道 1 设定为相位计数模式 1，给 TCLKA 和 TCLKB 输入编码器脉冲的 A 相和 B 相。

通道 0 通过 TGRC_0 的比较匹配进行 TCNT 计数器清除，TGRA_0 和 TGRC_0 用于比较匹配功能，设定速度控制周期和位置控制周期。TGRB_0 用于输入捕捉功能，使 TGRB_0 和 TGRD_0 进行缓冲运行。将通道 1 的计数器输入时钟作为 TGRB_0 的输入捕捉源，检测 2 相解码器的 4 倍频脉冲的脉宽。

将通道 1 的 TGRA_1 和 TGRB_1 设定为输入捕捉功能，选择通道 0 的 TGRA_0 和 TGRC_0 的比较匹配作为输入捕捉源，保存各控制周期时的递增 / 递减计数器的值。

能用此方法能检测正确的位置 / 速度。

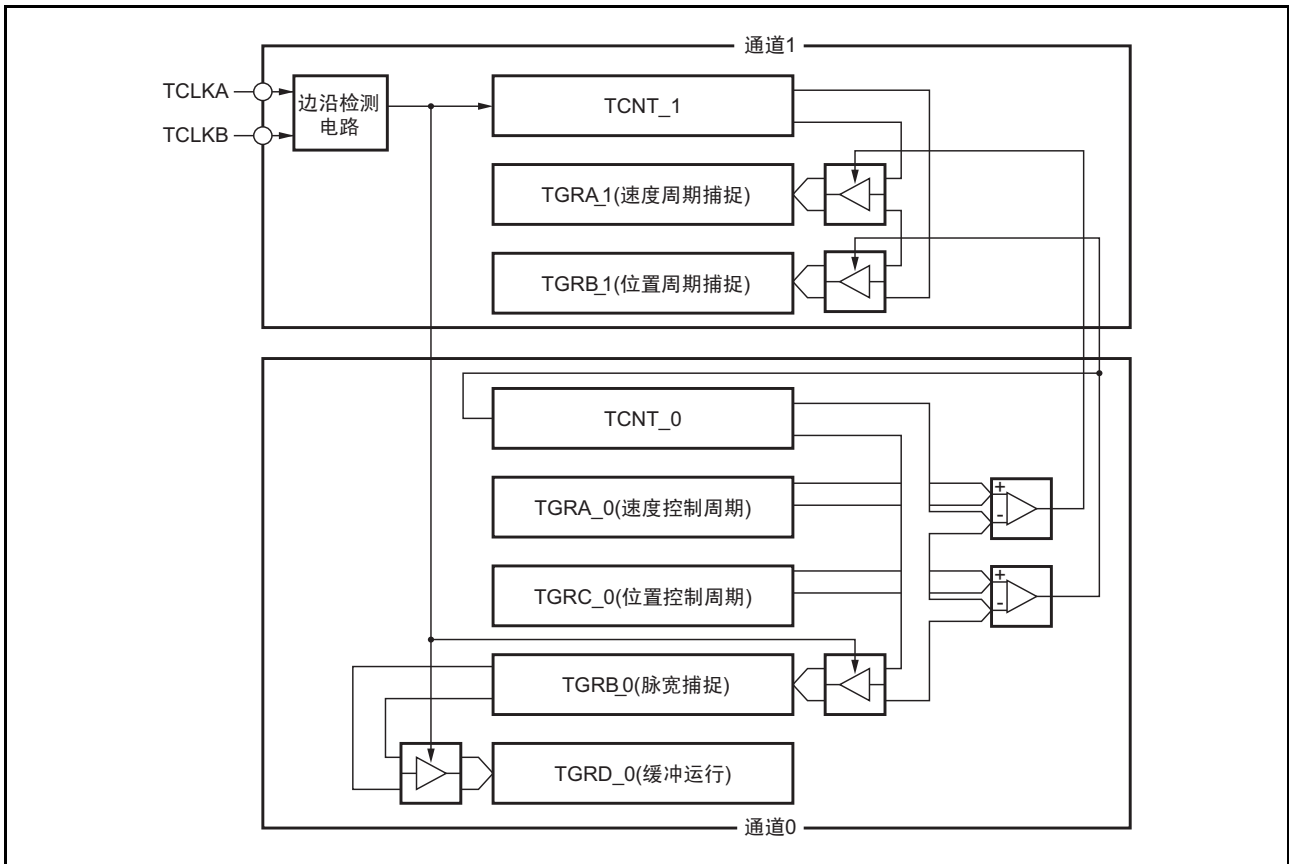


图 12.34 相位计数模式的应用例子

12.4.7 复位同步 PWM 模式

在复位同步 PWM 模式中，通过组合通道 3 和通道 4，将一方的波形变化点作为共同关系的 PWM 波形（正相和反相）进行 3 相输出。

如果设定为复位同步 PWM 模式，TIOC3B、TIOC3D、TIOC4A、TIOC4C、TIOC4B 和 TIOC4D 引脚就为 PWM 输出引脚，定时器的计数器 3（TCNT_3）用作递增计数器。

使用的 PWM 输出引脚和寄存器设定分别如表 12.50 和表 12.51 所示。

表 12.50 复位同步 PWM 模式中的输出引脚

通道	输出引脚	说明
3	TIOC3B	PWM 输出引脚 1
	TIOC3D	PWM 输出引脚 1'（PWM 输出 1 的反相波形）
4	TIOC4A	PWM 输出引脚 2
	TIOC4C	PWM 输出引脚 2'（PWM 输出 2 的反相波形）
	TIOC4B	PWM 输出引脚 3
	TIOC4D	PWM 输出引脚 3'（PWM 输出 3 的反相波形）

表 12.51 复位同步 PWM 模式中的寄存器设定

寄存器	设定内容
TCNT_3	初始设定“H'0000”。
TCNT_4	初始设定“H'0000”。
TGRA_3	设定 TCNT_3 的计数周期。
TGRB_3	设定从 TIOC3B 引脚和 TIOC3D 引脚输出的 PWM 波形的变化点。
TGRA_4	设定从 TIOC4A 引脚和 TIOC4C 引脚输出的 PWM 波形的变化点。
TGRB_4	设定从 TIOC4B 引脚和 TIOC4D 引脚输出的 PWM 波形的变化点。

(1) 复位同步 PWM 模式的设定步骤例子

复位同步 PWM 模式的设定步骤例子如图 12.35 所示。



图 12.35 复位同步 PWM 模式的设定步骤例子

(2) 复位同步 PWM 模式的运行例子

复位同步 PWM 模式的运行例子如图 12.36 所示。

在复位同步 PWM 模式中，TCNT_3 和 TCNT_4 作为递增计数器运行。如果 TCNT_3 和 TGRA_3 发生比较匹配，就清除计数器，并且从“H'0000”重新开始递增计数。每当各 TGRB_3、TGRA_4 和 TGRB_4 发生比较匹配并且进行计数器清除时，PWM 输出引脚就进行交替输出。

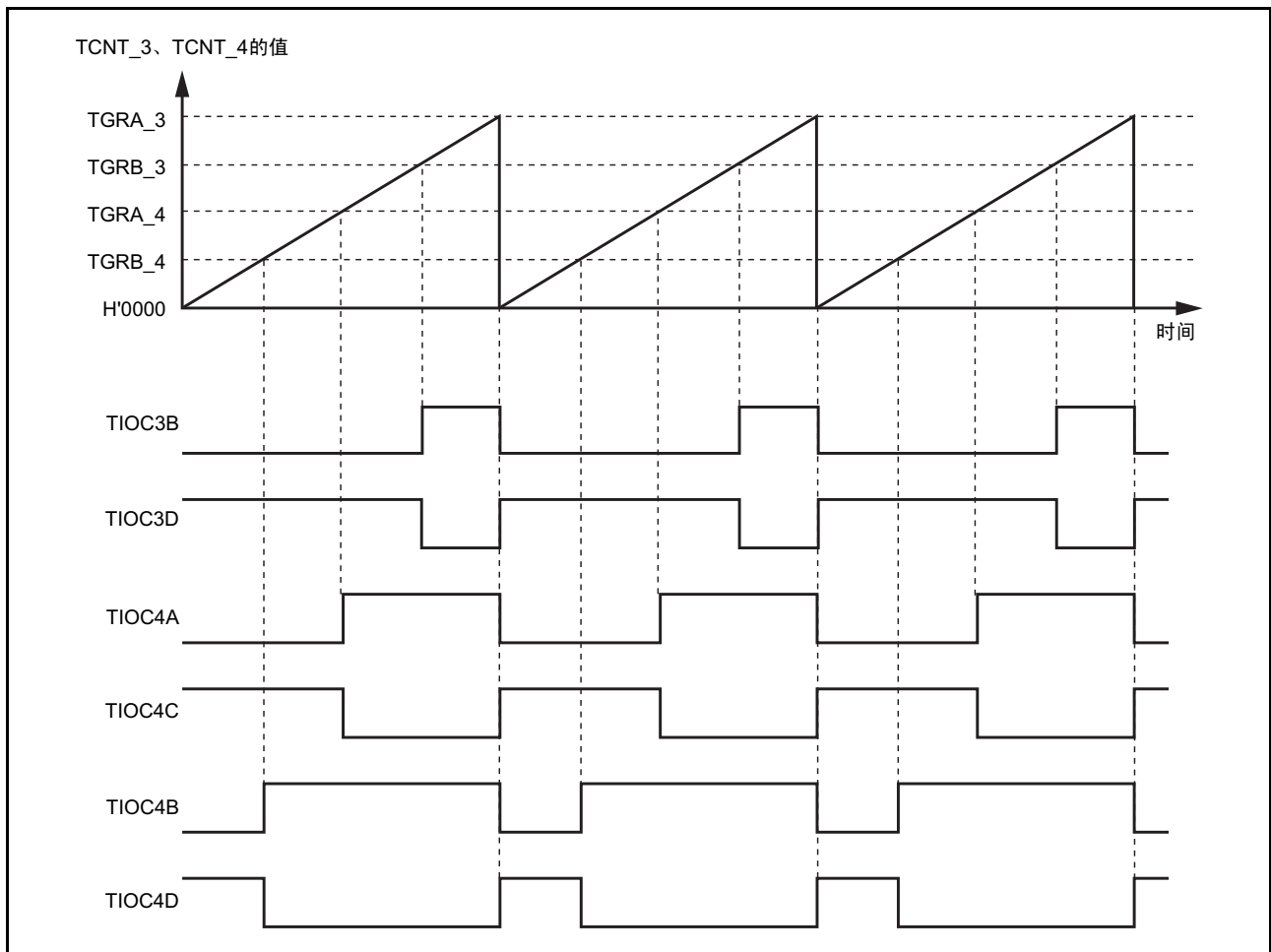


图 12.36 复位同步 PWM 模式的运行例子（TOCR 的 OLSN 位为“1”并且 OLSP 位为“1”的情况）

12.4.8 互补 PWM 模式

在互补 PWM 模式中，通过组合通道 3 和通道 4，将正相和反相为非重叠关系的 PWM 波形进行 3 相输出，也能设定为没有非重叠时间的波形。

如果设定为互补 PWM 模式，TIOC3B、TIOC3D、TIOC4A、TIOC4B、TIOC4C 和 TIOC4D 引脚就为 PWM 输出引脚，能将 TIOC3A 引脚设定为与 PWM 周期同步进行交替输出。

TCNT_3 和 TCNT_4 用作递增 / 递减计数器。

使用的 PWM 输出引脚和寄存器设定分别如表 12.52 和表 12.53 所示。

表 12.52 互补 PWM 模式中的输出引脚

通道	输出引脚	说明
3	TIOC3A	与 PWM 周期同步的交替输出（或者输入 / 输出端口）
	TIOC3B	PWM 输出引脚 1
	TIOC3C	输入 / 输出端口 *
	TIOC3D	PWM 输出引脚 1'（与 PWM 输出 1 有非重叠关系的反相波形，也能设定为没有非重叠时间的波形）
4	TIOC4A	PWM 输出引脚 2
	TIOC4C	PWM 输出引脚 2'（与 PWM 输出 2 有非重叠关系的反相波形，也能设定为没有非重叠时间的波形）
	TIOC4B	PWM 输出引脚 3
	TIOC4D	PWM 输出引脚 3'（与 PWM 输出 3 有非重叠关系的反相波形，也能设定为没有非重叠时间的波形）

【注】 * 在互补 PWM 模式中，不能将 TIOC3C 引脚设定为定时器的输入 / 输出引脚。

表 12.53 互补 PWM 模式的寄存器设定

通道	计数器 / 寄存器	说明	由 CPU 进行读写
3	TCNT_3	从空载时间寄存器的设定值开始递增计数。	能通过设定 TRWER* 进行屏蔽。
	TGRA_3	设定 TCNT_3 的上限值（1/2 个载波周期 + 空载时间）。	能通过设定 TRWER* 进行屏蔽。
	TGRB_3	PWM 输出 1 的比较寄存器	能通过设定 TRWER* 进行屏蔽。
	TGRC_3	TGRA_3 的缓冲寄存器	随时能读写。
	TGRD_3	PWM 输出 1/TGRB_3 的缓冲寄存器	随时能读写。
4	TCNT_4	初始设定“H'0000”并且开始递增计数。	能通过设定 TRWER* 进行屏蔽。
	TGRA_4	PWM 输出 2 的比较寄存器	能通过设定 TRWER* 进行屏蔽。
	TGRB_4	PWM 输出 3 的比较寄存器	能通过设定 TRWER* 进行屏蔽。
	TGRC_4	PWM 输出 2/TGRA_4 的缓冲寄存器	随时能读写。
	TGRD_4	PWM 输出 3/TGRB_4 的缓冲寄存器	随时能读写。
定时器的空载时间数据寄存器 (TDDR)		设定 TCNT_4 和 TCNT_3 的偏移值（空载时间的值）。	能通过设定 TRWER* 进行屏蔽。
定时器的周期数据寄存器 (TCDR)		设定 TCNT_4 的上限值（1/2 个载波周期）。	能通过设定 TRWER* 进行屏蔽。
定时器的周期缓冲寄存器 (TCBR)		TCDR 的缓冲寄存器	随时能读写。
副计数器 (TCNTS)		用于生成空载时间的副计数器	只能读。
暂存器 1 (TEMP1)		PWM 输出 1/TGRB_3 的暂存器	不能读写。
暂存器 2 (TEMP2)		PWM 输出 2/TGRA_4 的暂存器	不能读写。
暂存器 3 (TEMP3)		PWM 输出 3/TGRB_4 的暂存器	不能读写。

【注】 * 能通过设定 TRWER（定时器的读写允许寄存器），允许或者禁止存取。

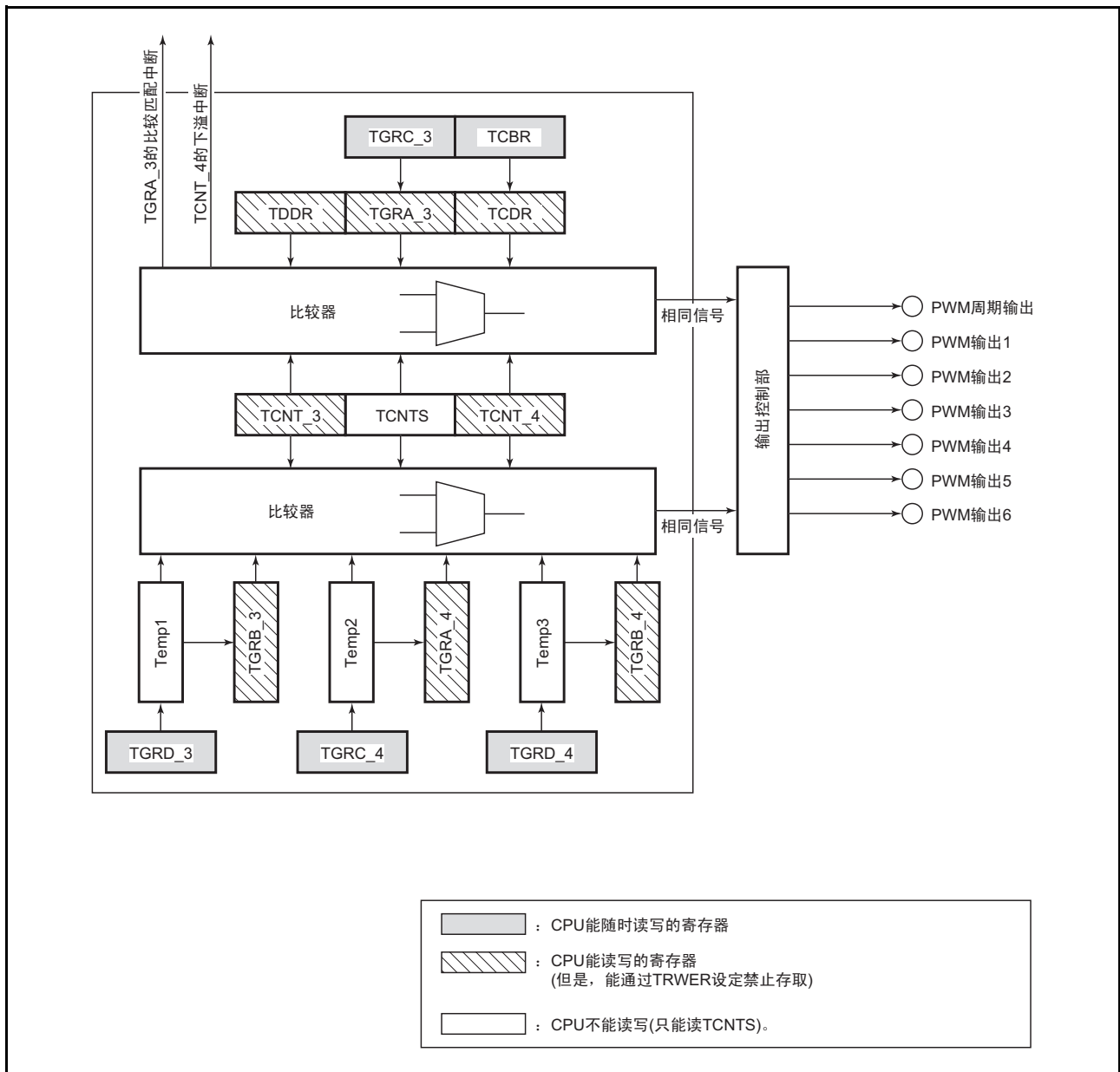


图 12.37 互补 PWM 模式中的通道 3 和通道 4 的框图

(1) 互补 PWM 模式的设定步骤例子

互补 PWM 模式的设定步骤例子如图 12.38 所示。



图 12.38 互补 PWM 模式的设定步骤例子

(2) 互补 PWM 模式的运行概要

在互补 PWM 模式中能进行 6 相的 PWM 输出。互补 PWM 模式的计数器运行和运行例子分别如图 12.39 和图 12.40 所示。

(a) 计数器的运行

在互补 PWM 模式中，TCNT_3、TCNT_4 和 TCNTS 这 3 个计数器进行递增和递减计数。

在设定为互补 PWM 模式并且 TSTR 的 CST 位为“0”时，将 TDDR 的设定值作为初始值，自动设定给 TCNT_3。

如果将 CST 位置“1”，TCNT_3 就进行递增计数，计数到 TGRA_3 的设定值为止。一旦 TCNT_3 和 TGRA_3 的值相同，TCNT_3 就改为递减计数。此后，一旦 TCNT_3 和 TDDR 的值相同，TCNT_3 就改为递增计数，重复此运行。

将 TCNT_4 的初始值设定为“H'0000”。

如果将 CST 位置“1”，TCNT_4 就与 TCNT_3 同步运行并且进行递增计数。一旦 TCNT_4 和 TCDR 的值相同，TCNT_4 就改为递减计数。此后，一旦 TCNT_4 计数到“H'0000”，TCNT_4 就改为递增计数，重复此运行。

TCNTS 是只读计数器，不需要设定初始值。

在 TCNT_3 和 TCNT_4 进行递增计数时，一旦 TCNT_3 和 TCDR 的值相同，TCNTS 就开始递减计数。一旦 TCNTS 和 TCDR 的值相同，TCNTS 就改为递增计数。一旦 TCNTS 和 TGRA_3 的值相同，TCNTS 就被清“H'0000”。

在 TCNT_3 和 TCNT_4 进行递减计数时，一旦 TCNT_4 和 TDDR 的值相同，TCNTS 就开始递增计数。一旦 TCNTS 和 TDDR 的值相同，TCNTS 就改为递减计数。一旦 TCNTS 计数到“H'0000”，TCNTS 就被设定为 TGRA_3 的值。

TCNTS 只在计数运行期间与设定 PWM 占空比的比较寄存器和暂存器进行比较。

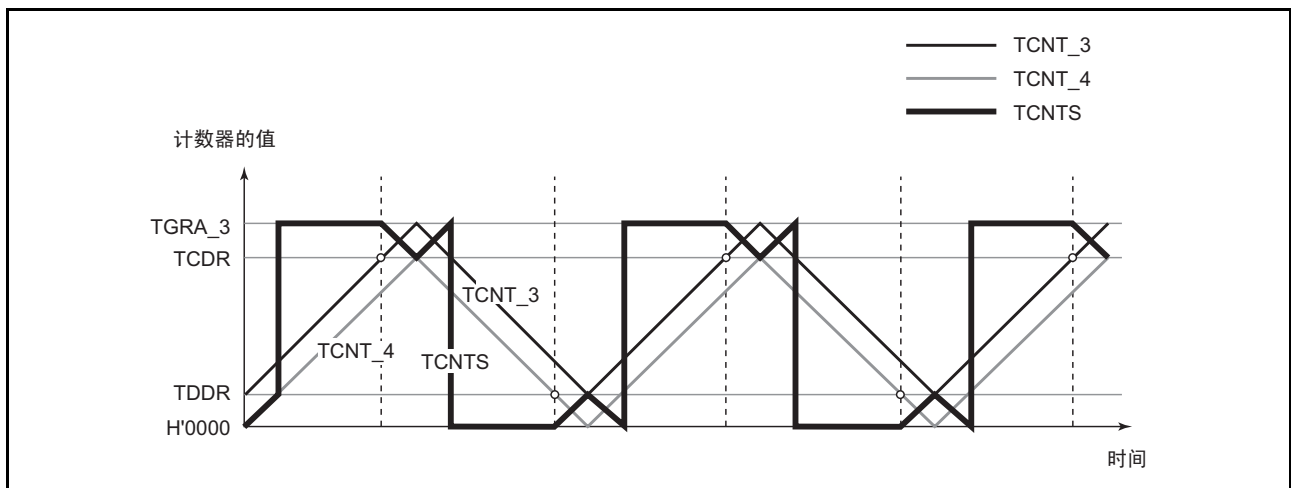


图 12.39 互补 PWM 模式的计数器运行

(b) 寄存器的运行

在互补 PWM 模式中，使用比较寄存器、缓冲寄存器和暂存器等 9 个寄存器。互补 PWM 模式的运行例子如图 12.40 所示。

为了进行 PWM 输出，寄存器 TGRB_3、TGRA_4、TGRB_4 总是和计数器进行比较。如果这些寄存器和计数器的值相同，就输出定时器的输出控制寄存器（TOCR）的 OLSN 位和 OLSP 位的设定值。

这些比较寄存器的缓冲寄存器为 TGRD_3、TGRC_4 和 TGRD_4。

在缓冲寄存器和比较寄存器之间有暂存器，但是 CPU 不能存取暂存器。

要更改比较寄存器的数据时，必须给对应的缓冲寄存器写要更改的数据。随时能读写缓冲寄存器。

总是在 T_a 区间将写到缓冲寄存器的数据传送到暂存器，而在 T_b 区间不传送到暂存器。在 T_b 区间结束后，将在此区间写到缓冲寄存器的数据传送到暂存器。

如果 T_b 区间结束的 TCNTS 在递增计数时计数值和 TGRA_3 的值相同，或者在递减计数时计数到“H'0000”，就将传送到暂存器的值传送到比较寄存器。能通过定时器的模式寄存器（TMDR）的 MD3 ~ MD0 位选择暂存器到比较寄存器的传送时序。选择在波谷进行更改的例子如图 12.40 所示。

在不向暂存器传送数据的 T_b （图 12.40 中为 T_{b1} ）区间，暂存器具有和比较寄存器相同的功能，和计数器进行比较。在此区间，对于 1 相输出有 2 个比较匹配寄存器，比较寄存器保存变更前的数据，暂存器保存要变更的新数据。在此区间，TCNT_3、TCNT_4 和 TCNTS 这 3 个计数器与比较寄存器和暂存器进行比较，控制 PWM 输出。

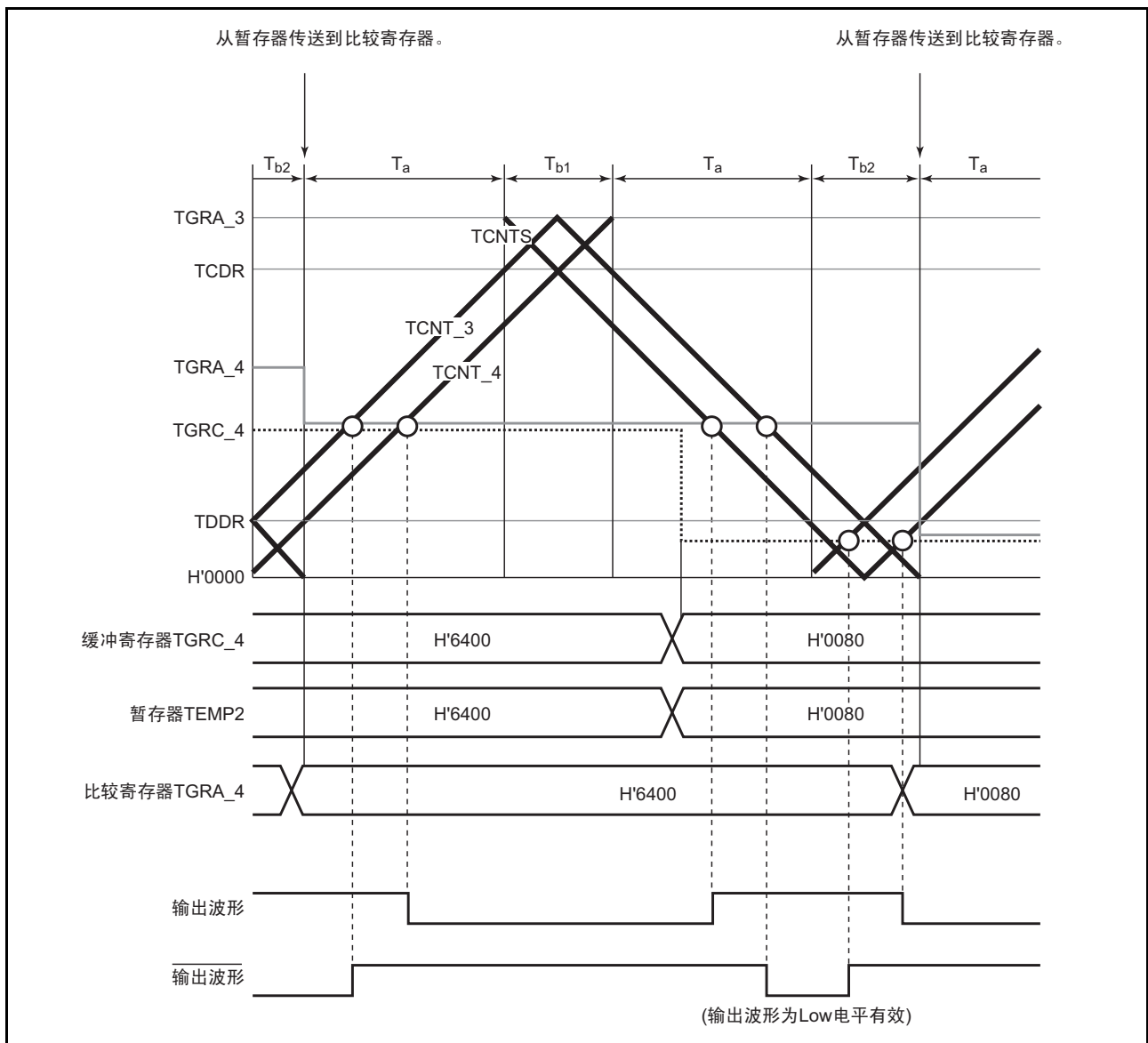


图 12.40 互补 PWM 模式的运行例子

(c) 初始设定

在互补 PWM 模式中，有 6 个需要初始设定的寄存器，还有 1 个用于设定是否生成空载时间的寄存器（只能在不生成空载时间的情况下进行设定）。

在通过定时器的模式寄存器（TMDR）的 MD3 ~ MD0 位设定为互补 PWM 模式前，必须设定以下寄存器的初始值。

TGRC_3 用作 TGRA_3 的缓冲寄存器，设定 1/2 个 PWM 载波周期 + 空载时间 Td。定时器的周期缓冲寄存器（TCBR）用作定时器的周期数据寄存器（TCDR）的缓冲寄存器，设定 1/2 个 PWM 载波周期，并且给定定时器的空载时间数据寄存器（TDDR）设定空载时间 Td。

如果不生成空载时间，就将定时器的空载时间允许寄存器（TDER）的 TDER 位置“0”，给 TGRC_3 和 TGRA_3 设定 1/2 个 PWM 载波周期 +1，并且将 TDDR 置“1”。

给缓冲寄存器 TGRD_3、TGRC_4 和 TGRD_4 这 3 个寄存器分别设定 PWM 占空比的初始值。

在设定为互补 PWM 模式的同时，将 TDDR 除外的 5 个缓冲寄存器的设定值分别传送到对应的比较寄存器。

必须在设定为互补 PWM 模式前将 TCNT_4 置“H'0000”。

表 12.54 需要初始设定的寄存器和计数器

寄存器 / 计数器	设定值
TGRC_3	1/2 个 PWM 载波周期 + 空载时间 Td (如果通过 TDER 设定为不生成空载时间，就为 1/2 个 PWM 载波周期 +1)
TDDR	空载时间 Td (如果通过 TDER 设定为不生成空载时间，就为“1”)
TCBR	1/2 个 PWM 载波周期
TGRD_3、TGRC_4、TGRD_4	各相 PWM 占空比的初始值
TCNT_4	H'0000

【注】 TGRC_3 的设定值必须为 TCBR 设定的 1/2 个 PWM 载波周期和 TDDR 设定的空载时间 Td 值的和。如果通过 TDER 设定为不生成空载时间，就必须为 1/2 个 PWM 载波周期 +1。

(d) PWM 输出电平的设定

在互补 PWM 模式中，通过定时器的输出控制寄存器 1（TOCR1）的 OLSN 位和 OLSP 位或者定时器的输出控制寄存器 2（TOCR2）的 OLS1P ~ OLS3P 位和 OLS1N ~ OLS3N 位，设定 PWM 脉冲的输出电平。

能按 6 相输出中的 3 相正相和 3 相反相设定输出电平。

必须在解除互补 PWM 模式的状态下设定或者更改输出电平。

(e) 空载时间的设定

在互补 PWM 模式中，输出正相和反相为非重叠关系的 PWM 脉冲，此非重叠时间称为空载时间。

将非重叠时间设定到定时器的空载时间数据寄存器（TDDR）。TDDR 的设定值为 TCNT_3 计数器的初始值，生成 TCNT_3 和 TCNT_4 的非重叠关系。必须在解除互补 PWM 模式的状态下更改 TDDR 的内容。

(f) 不生成空载时间的设定

通过将定时器的空载时间允许寄存器（TDER）的 TDER 位置“0”，设定为不生成空载时间。只能在 TDER 位为“1”的状态下读 TDER 位后给 TDER 位写“0”时，将 TDER 置“0”。

给 TGRA_3 和 TGRC_3 设定 1/2 个 PWM 载波周期 +1，并且将定时器的空载时间数据寄存器（TDDR）置“1”。

如果设定为不生成空载时间，就能输出无空载时间的 PWM 波形。不生成空载时间的运行例子如图 12.41 所示。

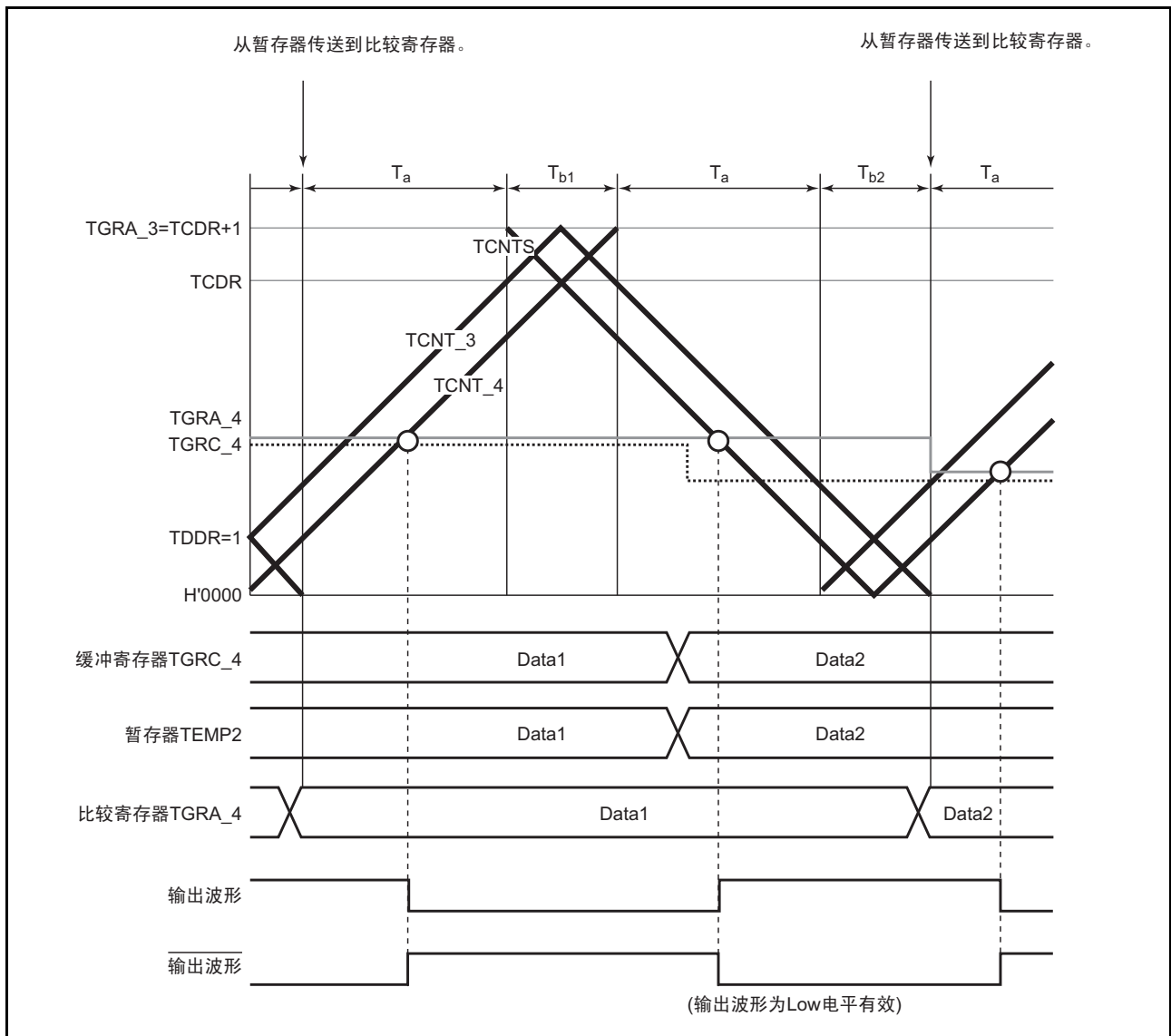


图 12.41 不生成空载时间的运行例子

(g) PWM 周期的设定

在互补 PWM 模式中，将 PWM 脉冲周期设定到 TGRA_3（设定 TCNT3 的上限值）和 TCDR（设定 TCNT_4 的上限值）2 个寄存器。必须将这 2 个寄存器设定为如下的关系：

生成空载时间：TGRA_3 的设定值 = TCDR 的设定值 + TDDR 的设定值

不生成空载时间：TGRA_3 的设定值 = TCDR 的设定值 + 1

必须通过给缓冲寄存器的 TGRC_3 和 TCBR 设定值，设定 TGRA_3 和 TCDR。在通过定时器的模式寄存器（TMDR）的 MD3 ~ MD0 选择的传送时序，将 TGRC_3 和 TCBR 的设定值同时传送到 TGRA_3 和 TCDR。

如果在波峰更新数据，就从下一个周期反映被更改的 PWM 周期；如果在波谷更新数据，就从该周期反映被更改的 PWM 周期。在波峰更改 PWM 周期时的运行例子如图 12.42 所示。

有关各缓冲寄存器数据的更新方法，请参照下面的“(h) 寄存器数据的更新”。

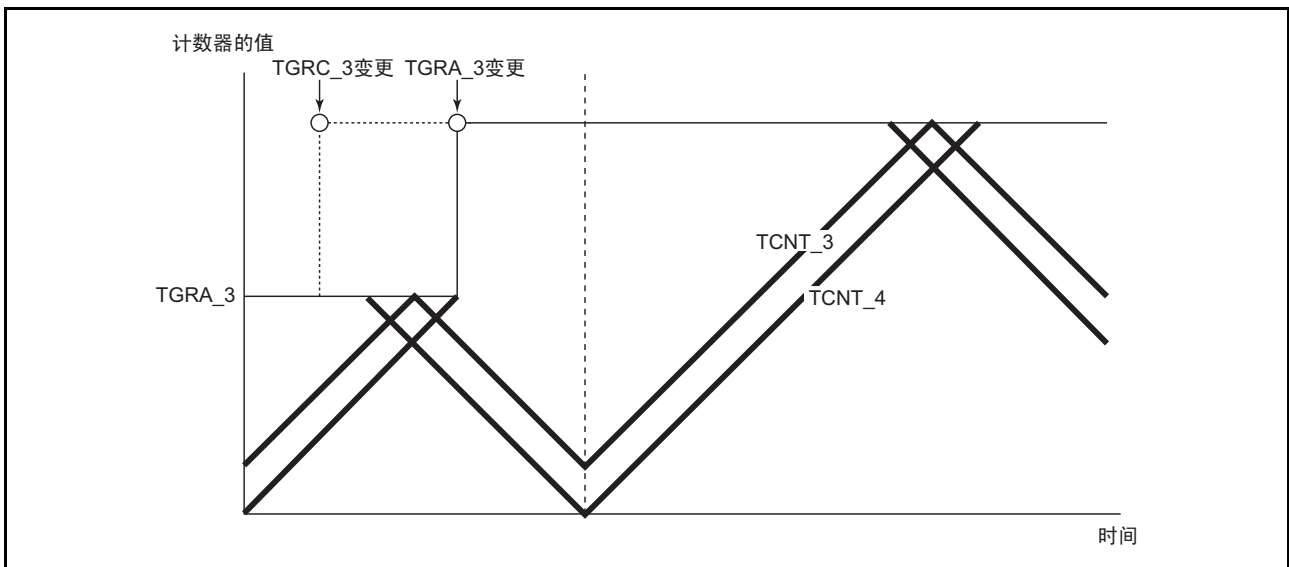


图 12.42 PWM 周期的变更例子

(h) 寄存器数据的更新

在互补 PWM 模式中更新比较寄存器的数据时，使用缓冲寄存器。更新的数据随时能写到缓冲寄存器。另外，在有缓冲寄存器的运行过程中能更改的寄存器为 5 个用于 PWM 占空比和载波周期的寄存器。

在这些寄存器和缓冲寄存器之间有各自的暂存器。在副计数器 TCNTS 不进行计数的期间更新缓冲寄存器的数据时，也会改写暂存器的值。在 TCNTS 计数过程中，不进行缓冲寄存器到暂存器的传送，而在 TCNTS 停止计数后传送写到缓冲寄存器的值。

在通过定时器的模式寄存器（TMDR）的 MD3 ~ MD0 位设定的数据更新时序，将暂存器的值传送到比较寄存器。互补 PWM 模式的数据更新例子如图 12.43 所示。此图是在计数器的波峰和波谷更新数据的例子。

在改写缓冲寄存器的数据时，最后必须写 TGRD_4。在写 TGRD_4 后，5 个寄存器的数据同时从缓冲寄存器传送到暂存器。

如果不全部更新 5 个寄存器或者不更新 TGRD_4 的数据，就必须在写要更新的寄存器数据后写 TGRD_4。此时，给 TGRD_4 写的的数据必须和写之前的数据相同。

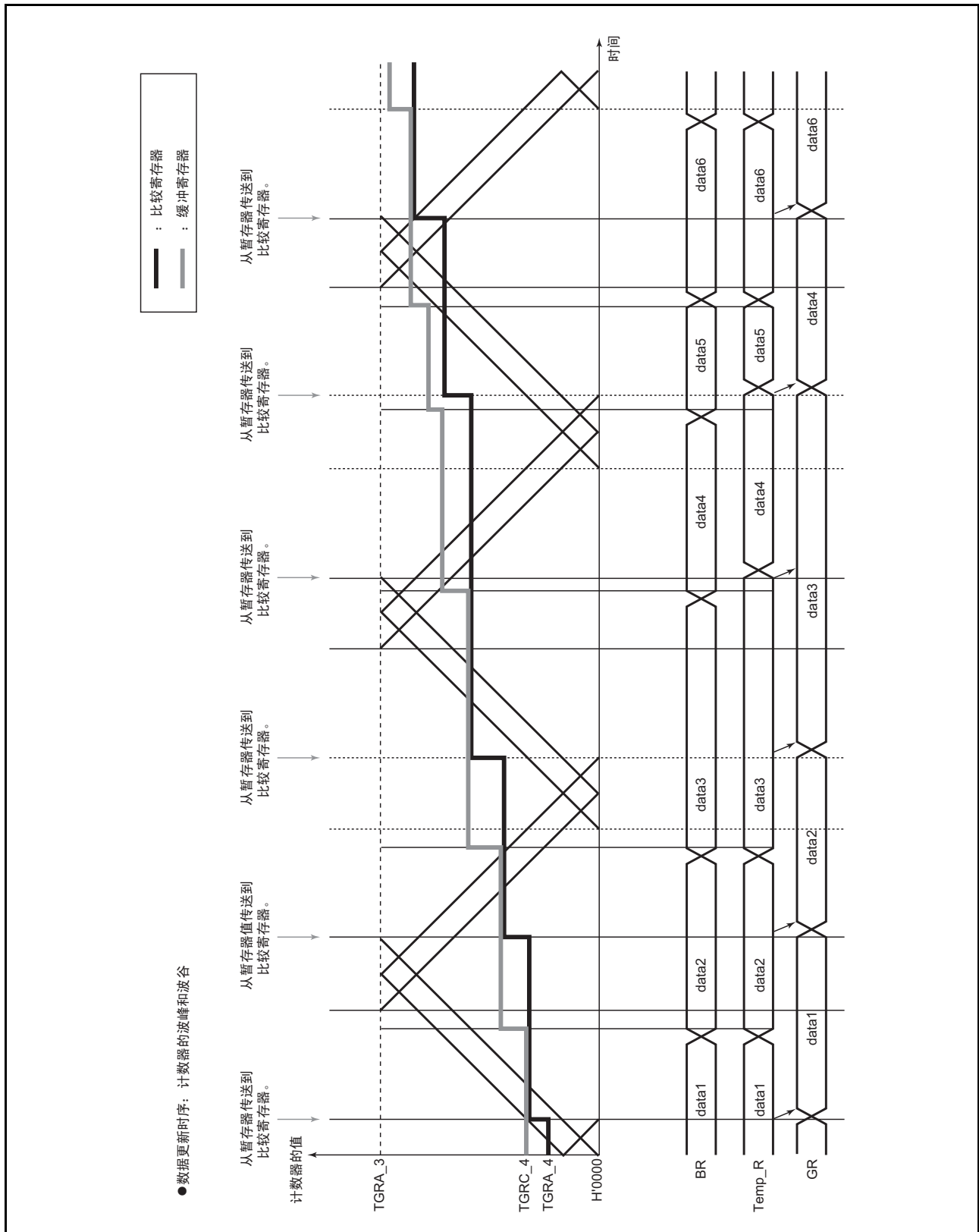


图 12.43 互补 PWM 模式的数据更新例子

(i) 互补 PWM 模式的初始输出

在互补 PWM 模式中，通过设定定时器的输出控制寄存器 1（TOCR1）的 OLSN 和 OLSP 位或者定时器的输出控制寄存器 2（TOCR2）的 OLS1N ~ OLS3N 位和 OLS1P ~ OLS3P 位，决定初始输出。

此初始输出为 PWM 脉冲的无效电平，从通过定时器的模式寄存器（TMDR）设定互补 PWM 模式到 TCNT_4 大于空载时间寄存器（TDDR）的设定值为止，输出此初始输出。互补 PWM 模式的初始输出例子如图 12.44 所示。

PWM 占空比的初始值小于 TDDR 值的波形例子如图 12.45 所示。

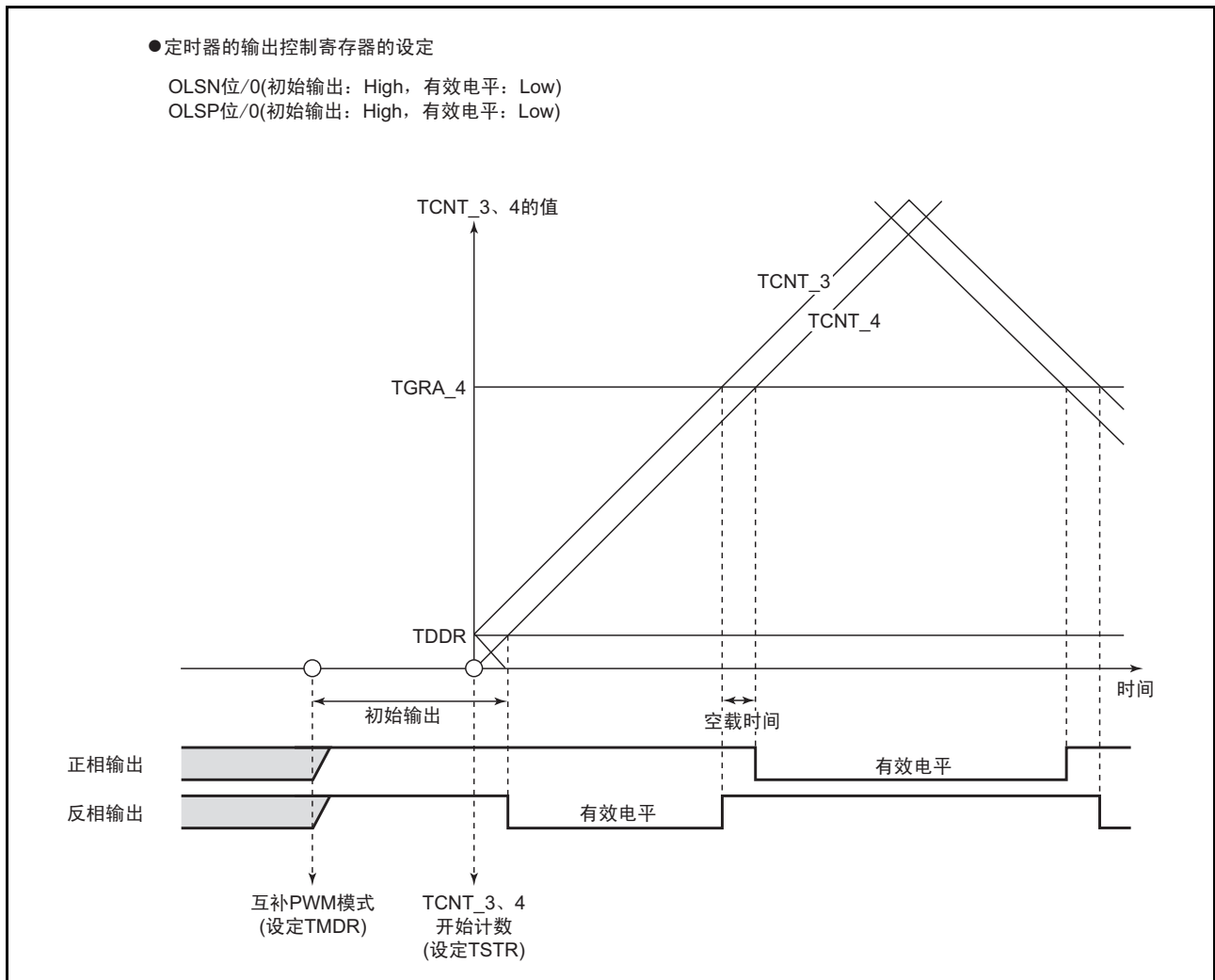


图 12.44 互补 PWM 模式的初始输出例子 (1)

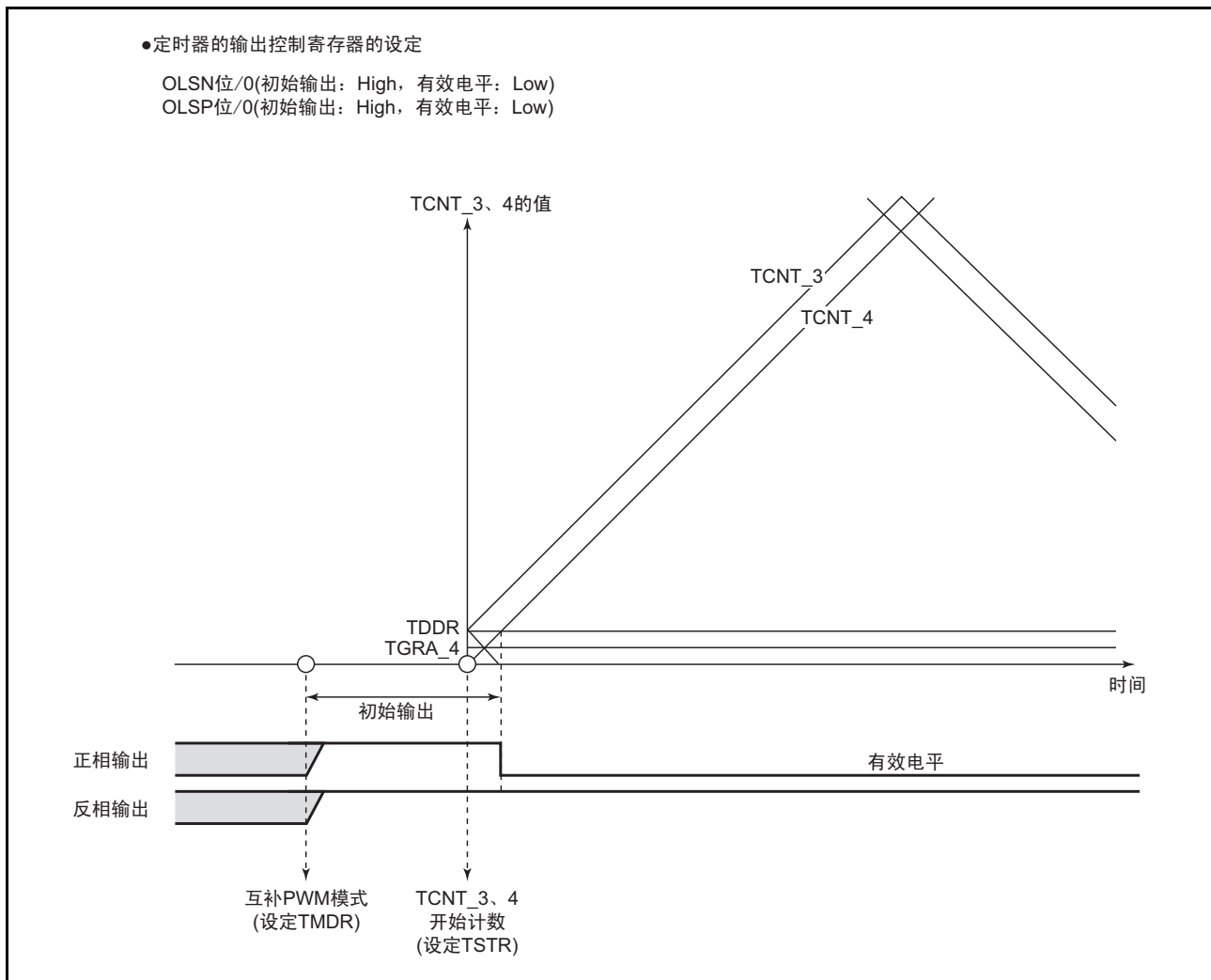


图 12.45 互补 PWM 模式的初始输出例子 (2)

(j) 互补 PWM 模式的 PWM 输出生成方法

在互补 PWM 模式中，将正相和反相有非重叠时间的 PWM 波形进行 3 相输出，此非重叠时间称为空载时间。

在计数器和数据寄存器发生比较匹配时，通过输出定时器的输出控制寄存器选择的输出电平，生成 PWM 波形。在 TCNTS 进行计数的期间，因为生成 0 ~ 100% 连续的 PWM 脉冲，所以同时比较数据寄存器和暂存器的值。此时，ON/OFF 比较匹配的发生时序会有前后，为了确保空载时间并且使正相 / 反相的 ON 时间不重叠，必须优先各相 OFF 的比较匹配。互补 PWM 模式的波形生成例子如图 12.46 ~ 图 12.48 所示。

通过和实线计数器的比较匹配，生成正相 / 反相的 OFF 时序，通过和点线计数器（比实线计数器迟空载时间）的比较匹配，生成 ON 时序。在 T1 期间，最优先使反相 OFF 的 a 的比较匹配，忽视比 a 先发生的比较匹配。在 T2 期间，最优先使正相 OFF 的 c 的比较匹配，忽视比 c 先发生的比较匹配。

如图 12.46 所示，通常比较匹配的发生顺序是 a→b→c→d（或者 c→d→a'→b'）。

当比较匹配的发生顺序不是 a→b→c→d 时，因为反相的 OFF 时间短于 2 倍的空载时间，所以表示正相不为 ON；当比较匹配的发生顺序不是 c→d→a'→b' 时，因为正相的 OFF 时间短于 2 倍的空载时间，所以表示反相不为 ON。

如图 12.47 所示，如果在 a 的比较匹配之后先发生 c 的比较匹配，就忽视 b 的比较匹配，而通过 d 的比较匹配使反相 OFF。这是因为比 b 的比较匹配（正相 ON 时序）先发生正相 OFF 的 c 的比较匹配，而优先使正相 OFF（因为正相从 OFF 变为 OFF，所以波形不变）。

同样地，在图 12.48 所示的例子中，比 c 的比较匹配先发生和暂存器的新数据比较匹配的 a'，但是在发生使正相 OFF 的 c 之前忽视其他比较匹配。因此不使反相 ON。

如此，在互补 PWM 模式中，优先 OFF 时序的比较匹配，即使 ON 时序的比较匹配比 OFF 先发生，也被忽视。

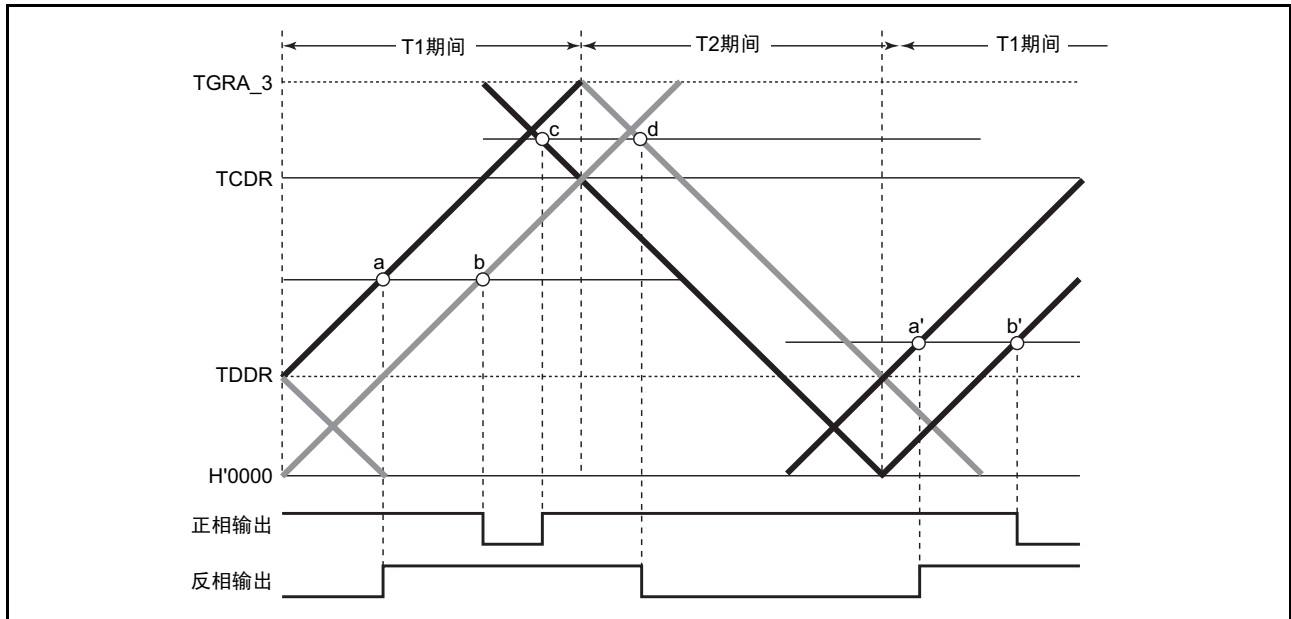


图 12.46 互补 PWM 模式的波形输出例子 (1)

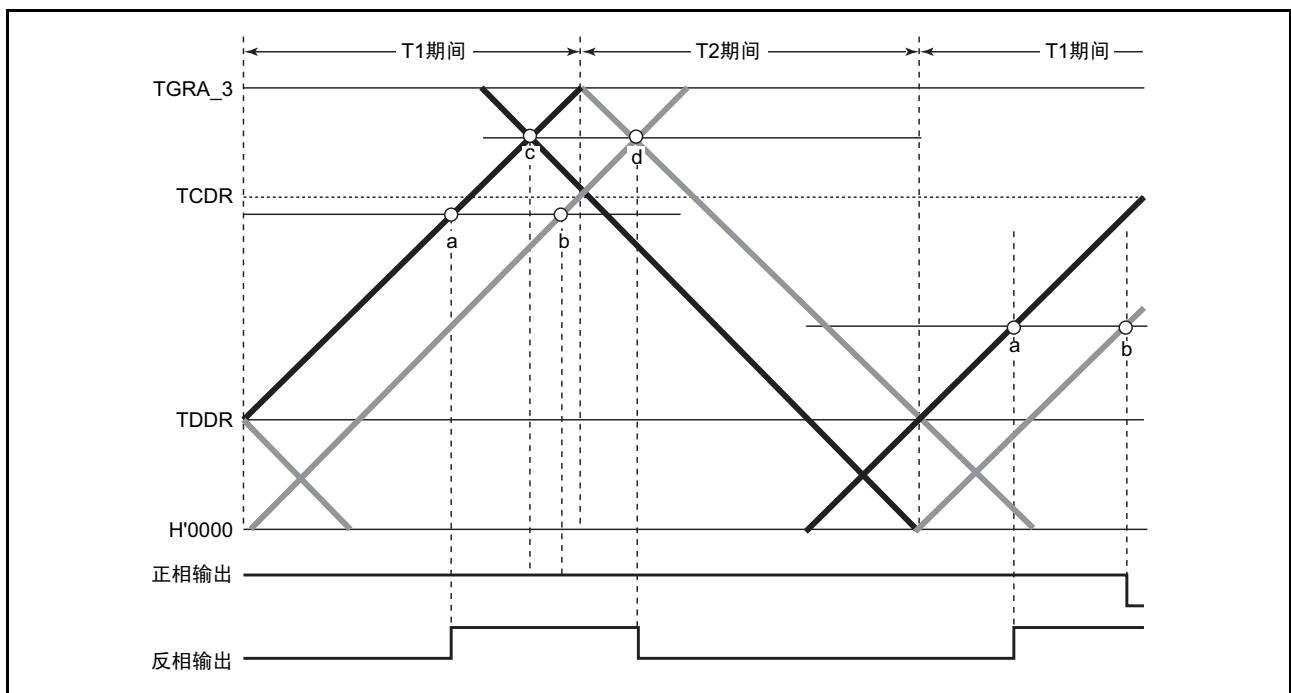


图 12.47 互补 PWM 模式的波形输出例子 (2)

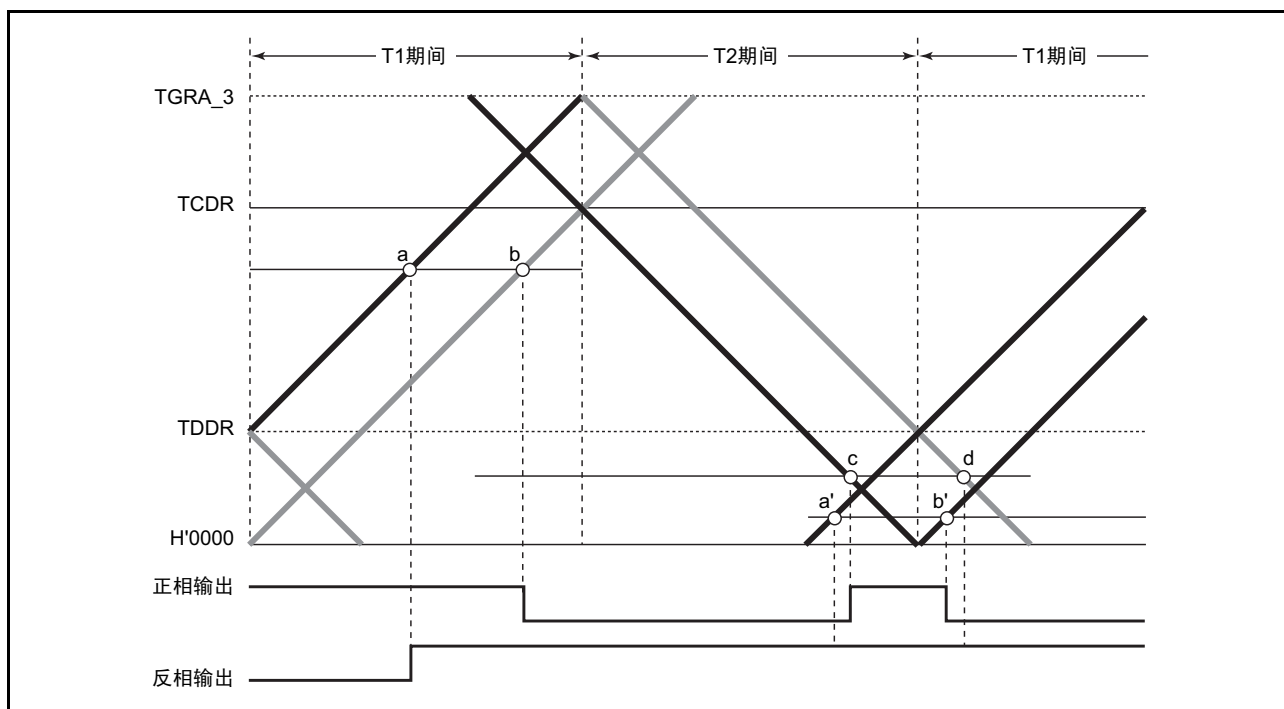


图 12.48 互补 PWM 模式的波形输出例子 (3)

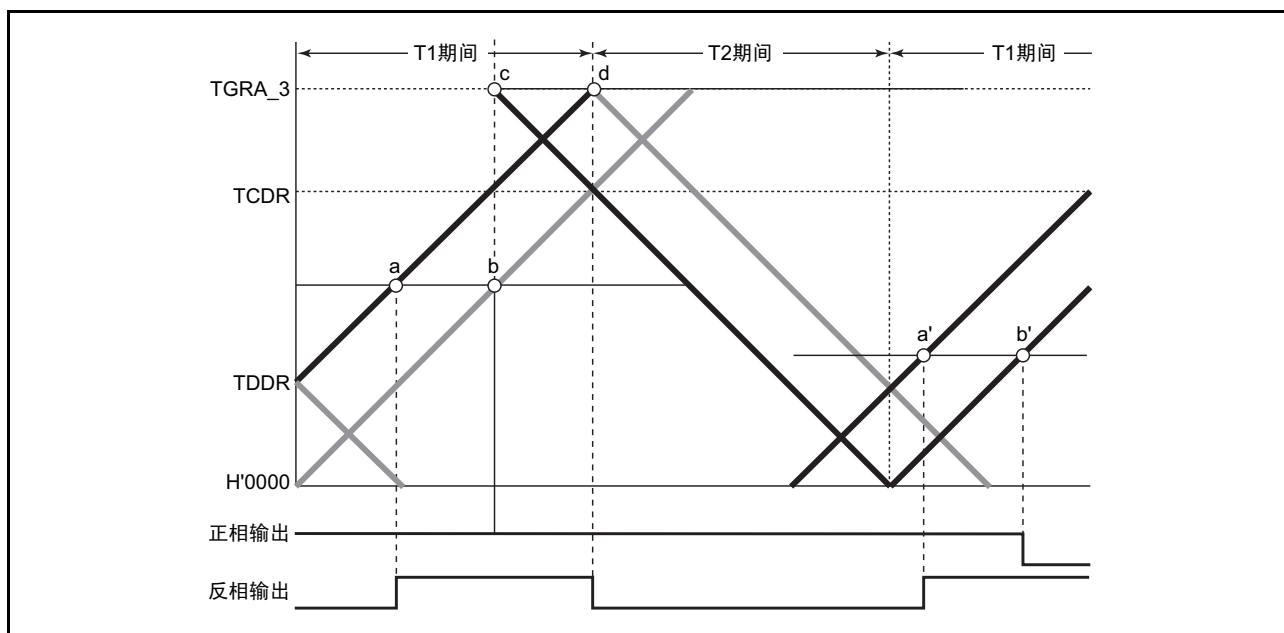


图 12.49 互补 PWM 模式的 0%、100% 波形输出例子 (1)

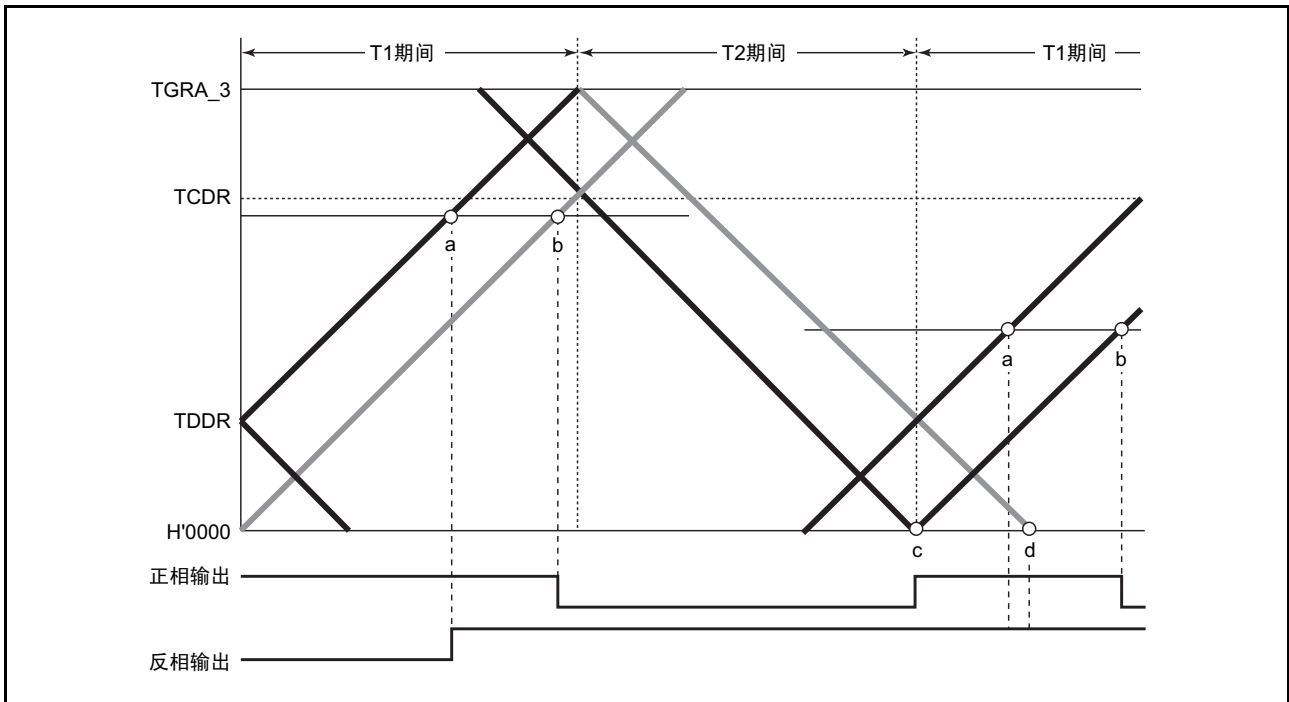


图 12.50 互补 PWM 模式的 0%、100% 波形输出例子 (2)

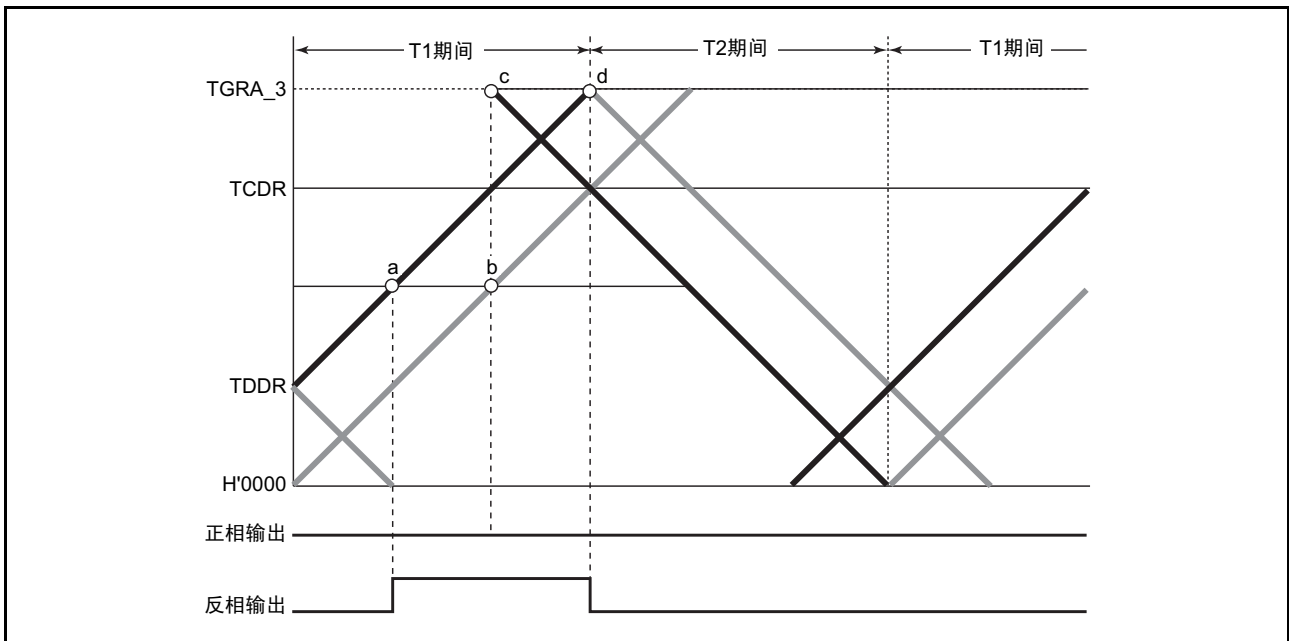


图 12.51 互补 PWM 模式的 0%、100% 波形输出例子 (3)

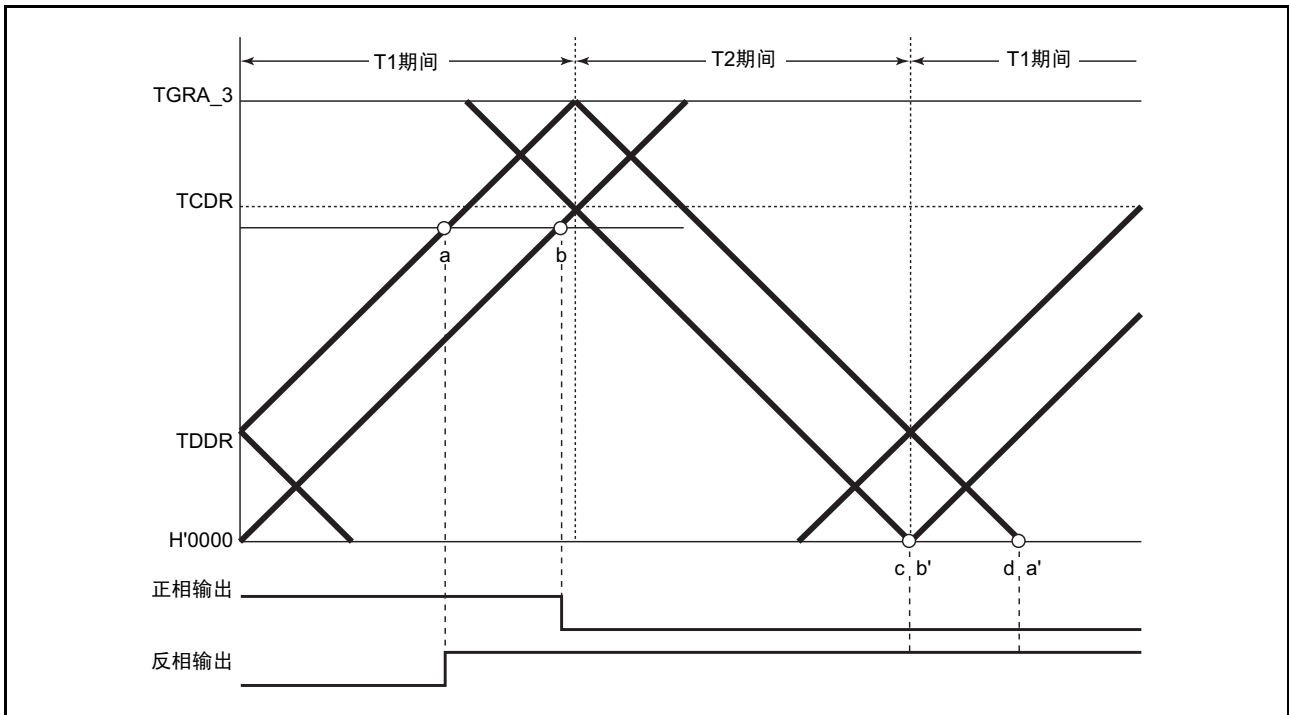


图 12.52 互补 PWM 模式的 0%、100% 波形输出例子 (4)

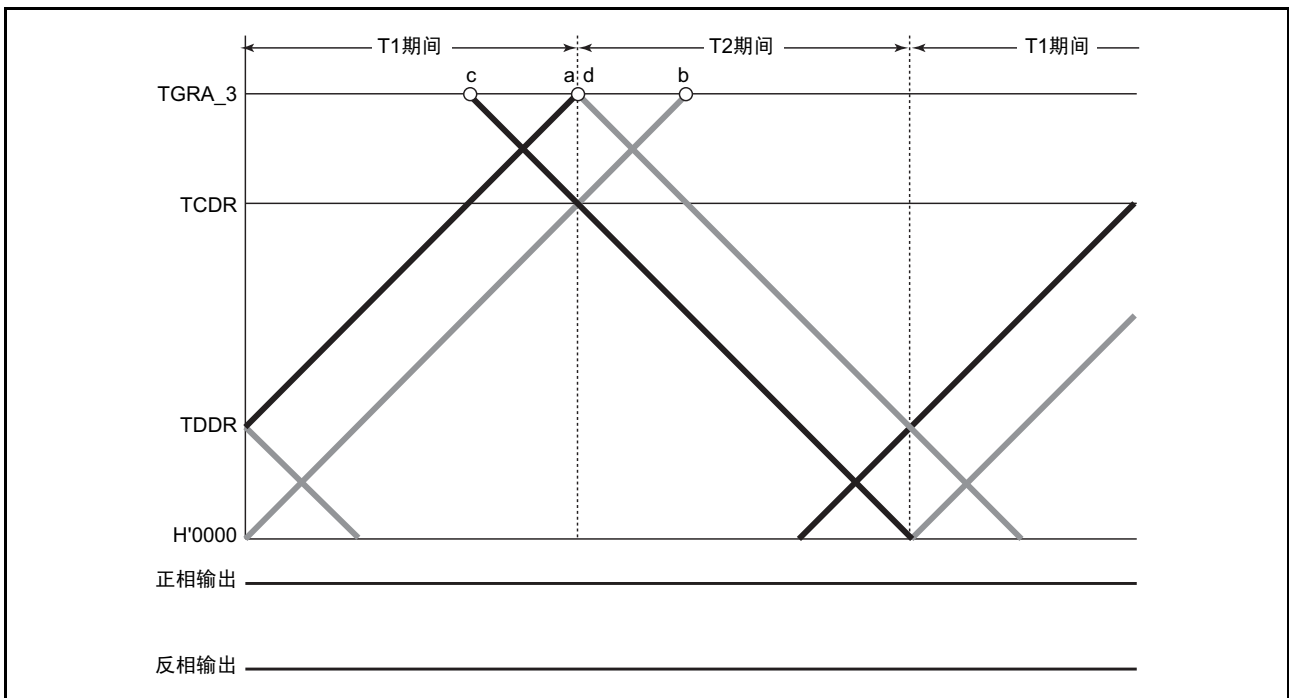


图 12.53 互补 PWM 模式的 0%、100% 波形输出例子 (5)

(k) 互补 PWM 模式的 0%、100% 占空比输出

在互补 PWM 模式中，能任意输出 0%、100% 的占空比，输出例子如图 12.49 ~ 图 12.53 所示。

如果将数据寄存器的值设定为“H'0000”，就输出 100% 的占空比，此时的波形是正相为 100%ON 状态的波形。如果将数据寄存器的值设定为和 TGRA_3 相同的值，就输出 0% 的占空比，此时的波形是正相为 100% OFF 状态的波形。

此时，同时发生 ON 和 OFF 的比较匹配，如果同相的 ON 比较匹配和 OFF 比较匹配同时发生，双方的比较匹配就都被忽视而波形不变。

(l) 与 PWM 周期同步的交替输出

在互补 PWM 模式中，能通过将定时器的输出控制寄存器（TOCR）的 PSYE 位置“1”，与 PWM 载波周期同步进行交替输出。交替输出的波形例子如图 12.54。

通过 TCNT_3 和 TGRA_3 的比较匹配以及 TCNT4 和“H'0000”的比较匹配进行交替输出。

交替输出的输出引脚为 TIOC3A 引脚，初始输出为 High 电平。

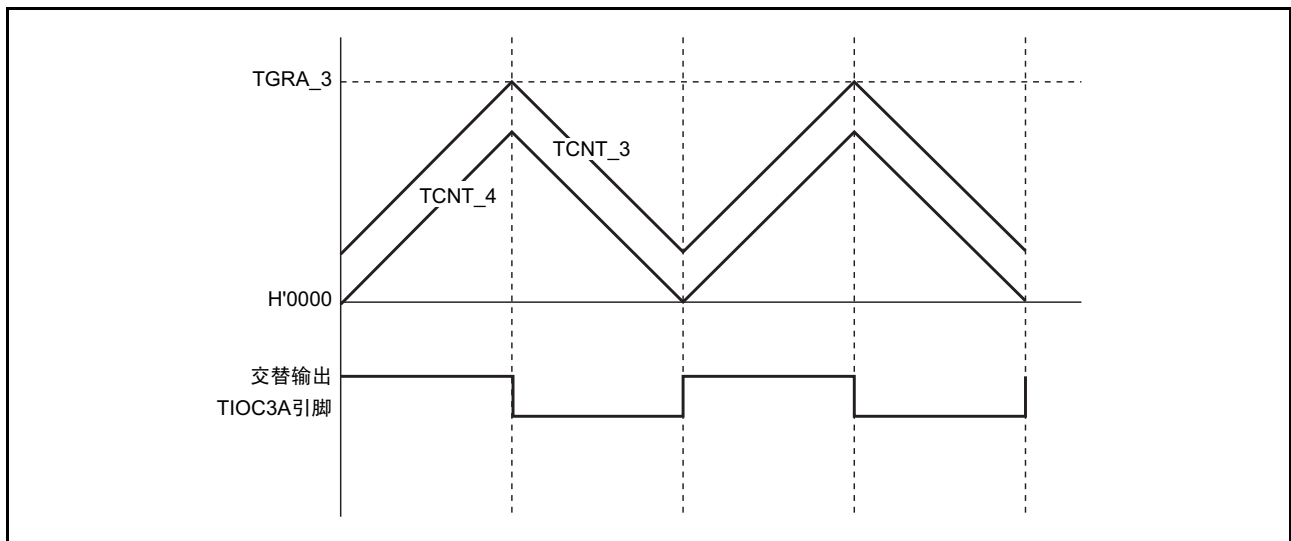


图 12.54 与 PWM 输出同步进行交替输出的波形例子

(m) 其他通道的计数器清除

在互补 PWM 模式中，当通过定时器的同步寄存器（TSYR）设定为与其他通道同步的模式并且通过定时器的控制寄存器（TCR）的 CCLR2 ~ CCLR0 选择同步清除时，能由其他通道进行 TCNT_3、TCNT_4 和 TCNTS 的清除。

运行例子如图 12.55 所示。

使用此功能，能通过外部信号进行计数器的清除和重新开始。

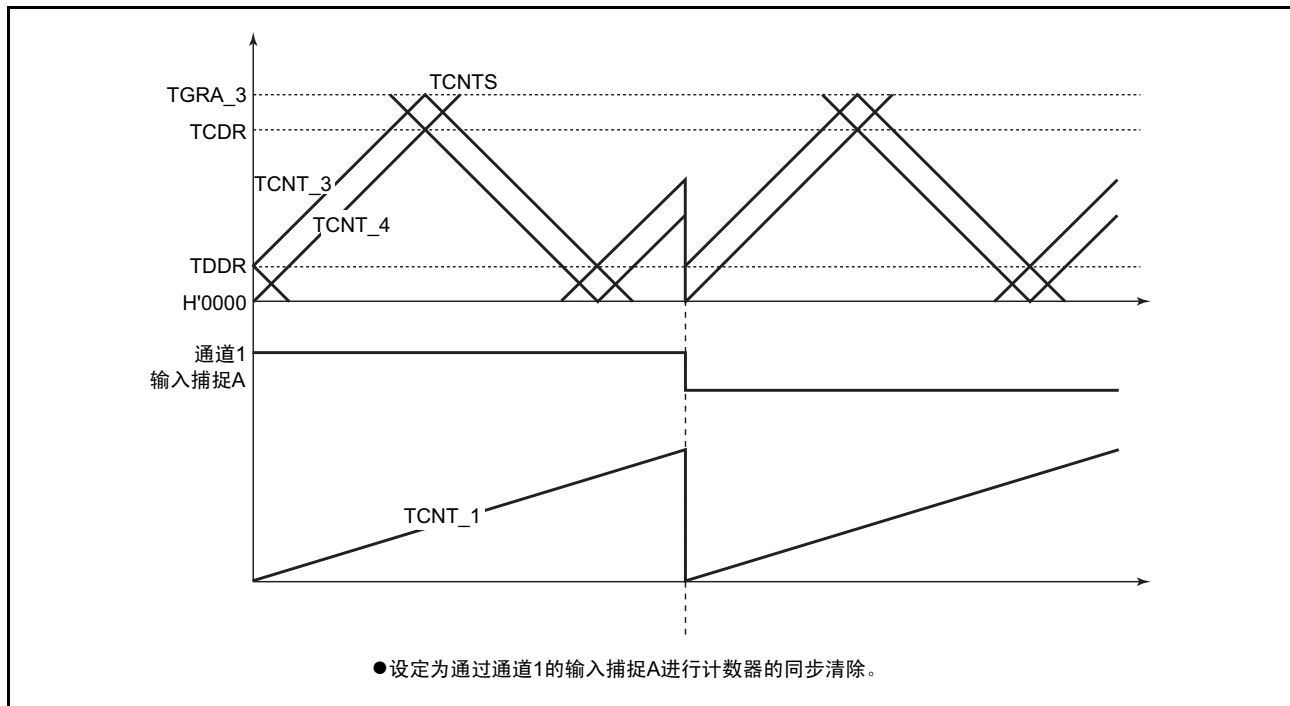


图 12.55 与其他通道同步的计数器清除

(n) 互补 PWM 模式的同步计数器清除时的输出波形控制

能通过将 TWCR 寄存器的 WRE 位置 “1”，抑制在互补 PWM 模式的波谷 Tb 区间发生同步计数器清除时的初始输出。因此，能抑制同步计数器清除时占空比的急剧变化。

只有在如图 12.56 的⑩、⑪ 的波谷 Tb 区间进行同步清除时，才能通过将 WRE 位置 “1” 来抑制初始输出。如果在其他时序中发生同步清除，就输出 TOCR 寄存器的 OLS 位设定的初始值。即使在波谷的 Tb 区间，如果在图 12.56 的①所示的计数器刚开始计数后的初始输出期间发生同步清除，也不抑制初始输出。

要抑制初始输出时，必须将比较寄存器 TGRB_3、TGRA_4、TGRB_4 全部设定为至少是 2 倍的空载时间数据寄存器 TDDR。如果在小于 2 倍的 TDDR 的状态下进行同步清除，PWM 输出的空载时间就可能会变短（消失），或者可能在抑制初始输出的期间从 PWM 反相输出引脚输出不正确的有效电平。详细内容请参照“12.7.23 在互补 PWM 模式进行同步计数器清除时的输出波形控制的注意事项”

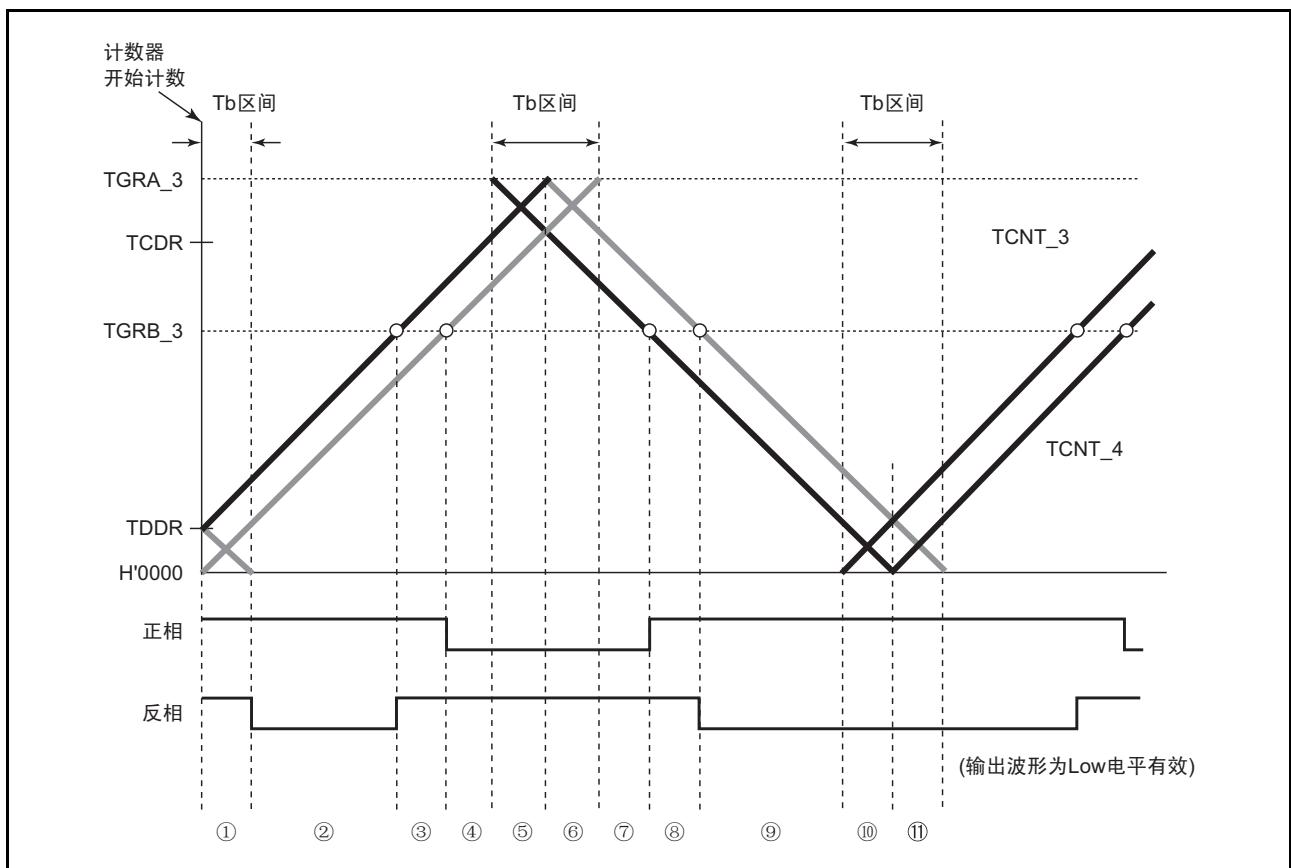


图 12.56 同步计数器的清除时序

- 互补 PWM 模式的同步计数器清除时输出波形控制的设定步骤例子
互补 PWM 模式的同步计数器清除时输出波形控制的设定步骤例子如图 12.57 所示。

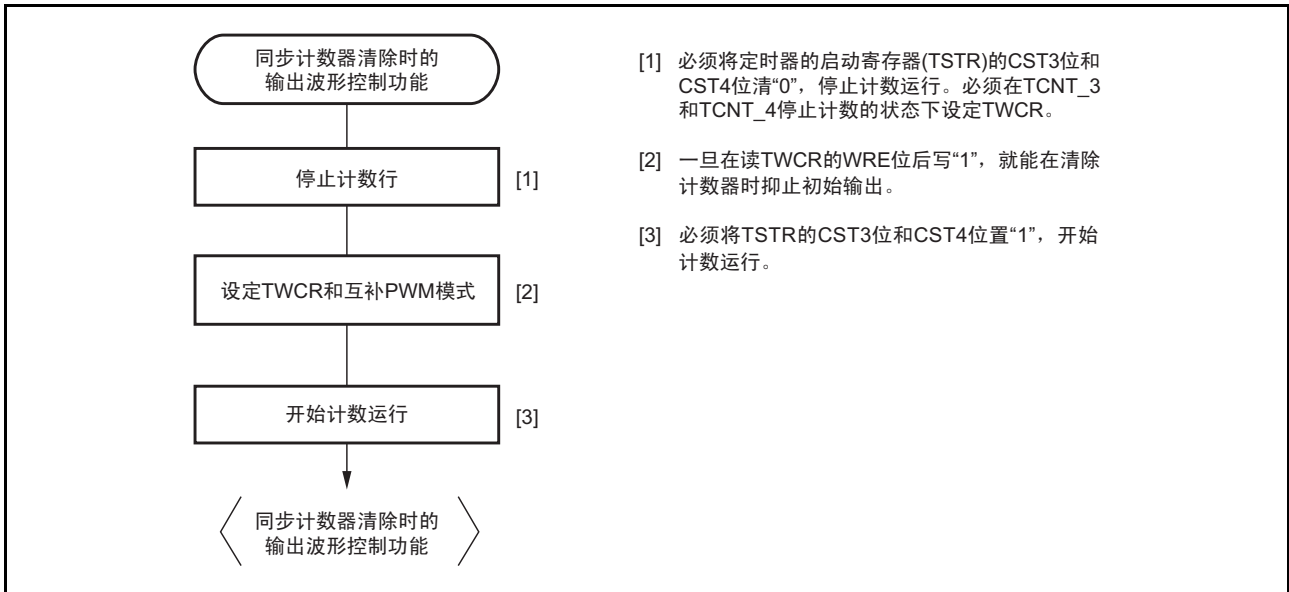


图 12.57 互补 PWM 模式的同步计数器清除时输出波形控制的设定步骤例子

- 互补 PWM 模式的同步计数器清除时输出波形控制的运行例子
在将 TWCR 的 WRE 位置 “1” 的状态下使此模块进行互补 PWM 运行并且进行同步计数器清除时的运行例子如图 12.58 ~ 图 12.61 所示。在此，图 12.58 ~ 图 12.61 的同步计数器清除时序分别是图 12.56 的 ③、⑥、⑧、⑩ 所示的时序。

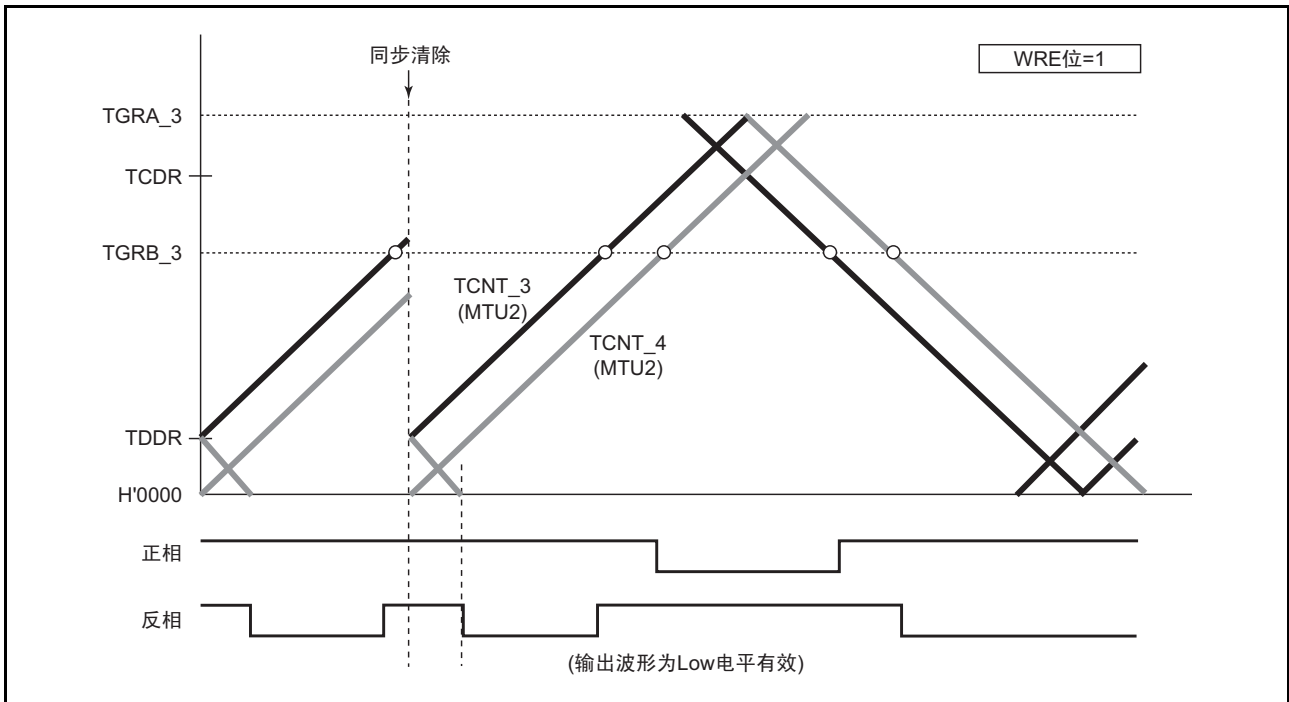


图 12.58 在递增计数过程中的空载时间发生同步清除的情况
(图 12.56 的时序③，此模块的 TWCR 寄存器的 WRE 位为 “1”)

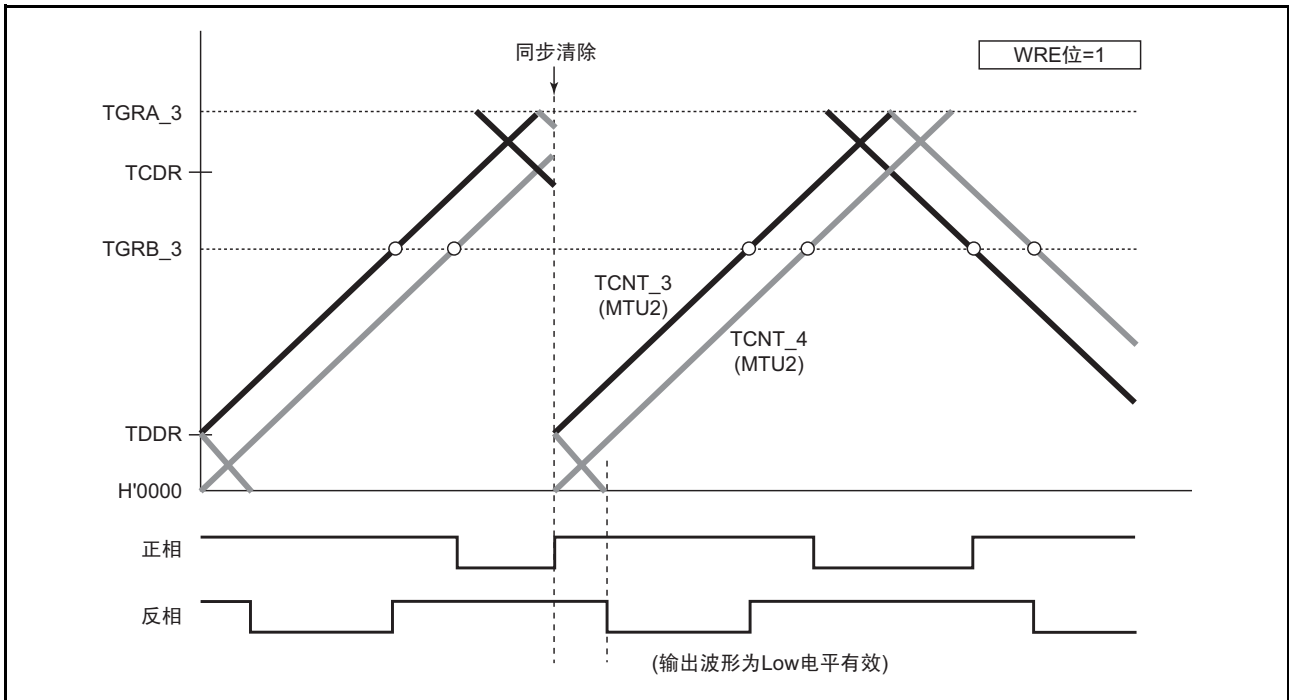


图 12.59 在波峰 Tb 区间发生同步清除的情况
 (图 12.56 的时序⑥, 此模块的 TWCR 寄存器的 WRE 位为 “1”)

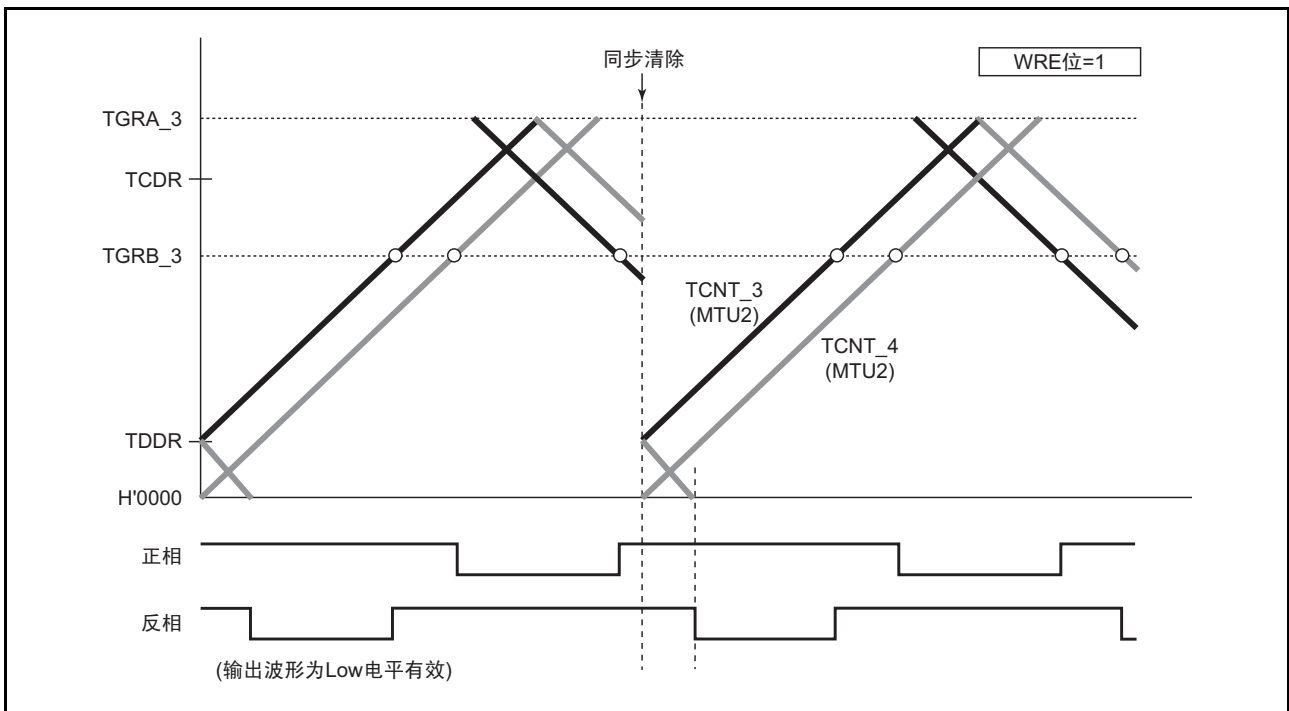


图 12.60 在递减计数过程中的空载时间发生同步清除的情况
 (图 12.56 的时序⑧, TWCR 寄存器的 WRE 位为 “1”)

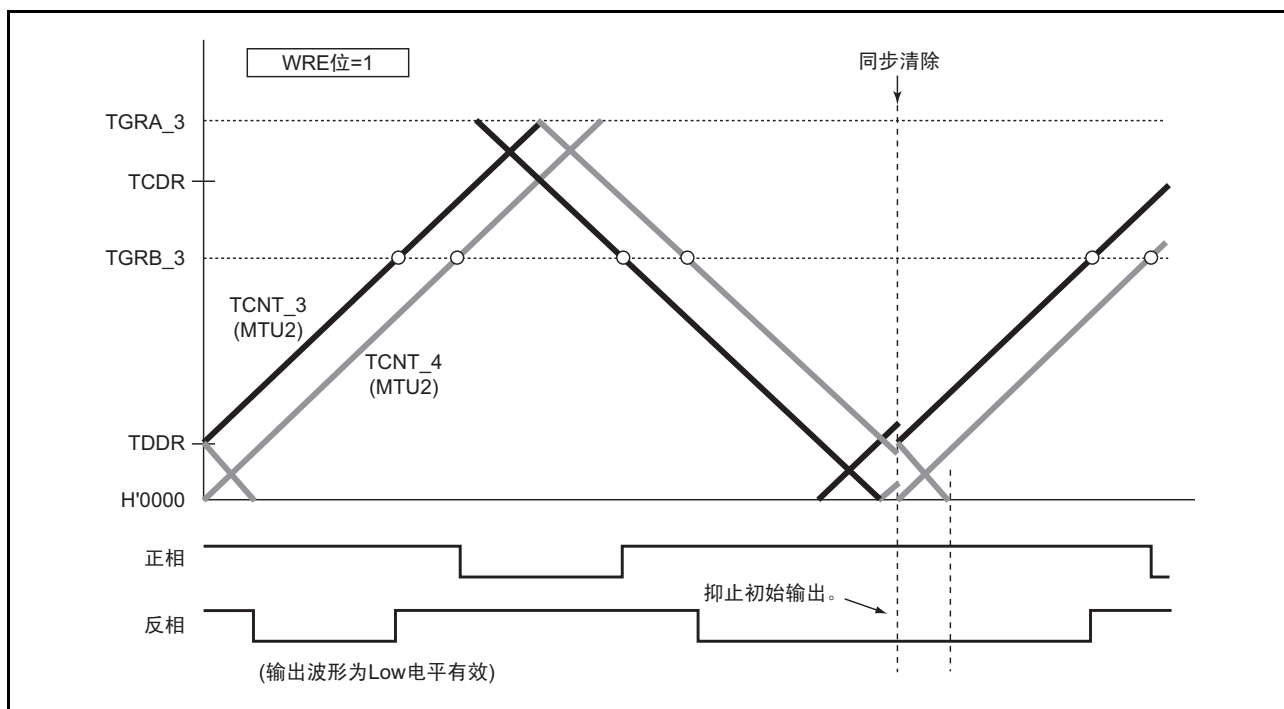


图 12.61 在波谷 Tb 区间发生同步清除的情况
 (图 12.56 的时序 ①, TWCR 寄存器的 WRE 位为 “1”)

(o) 通过 TGRA_3 的比较匹配进行的计数器清除

在互补 PWM 模式中，能通过设定定时器的波形控制寄存器（TWCR）的 CCE 位，在 TGRA_3 的比较匹配时清除 TCNT_3、TCNT_4 和 TCNTS。

运行例子如图 12.62 所示。

- 【注】
1. 只能在互补 PWM 模式 1（在波峰进行传送）中使用。
 2. 不能设定为与其他通道同步的清除功能（不能将定时器的同步寄存器（TSYR）的 SYNC0~SYNC4 位置“1”）。
 3. 不能将 PWM 占空比设定为“H'0000”。
 4. 不能将定时器的输出控制寄存器 1（TOCR1）的 PSYE 位置“1”。

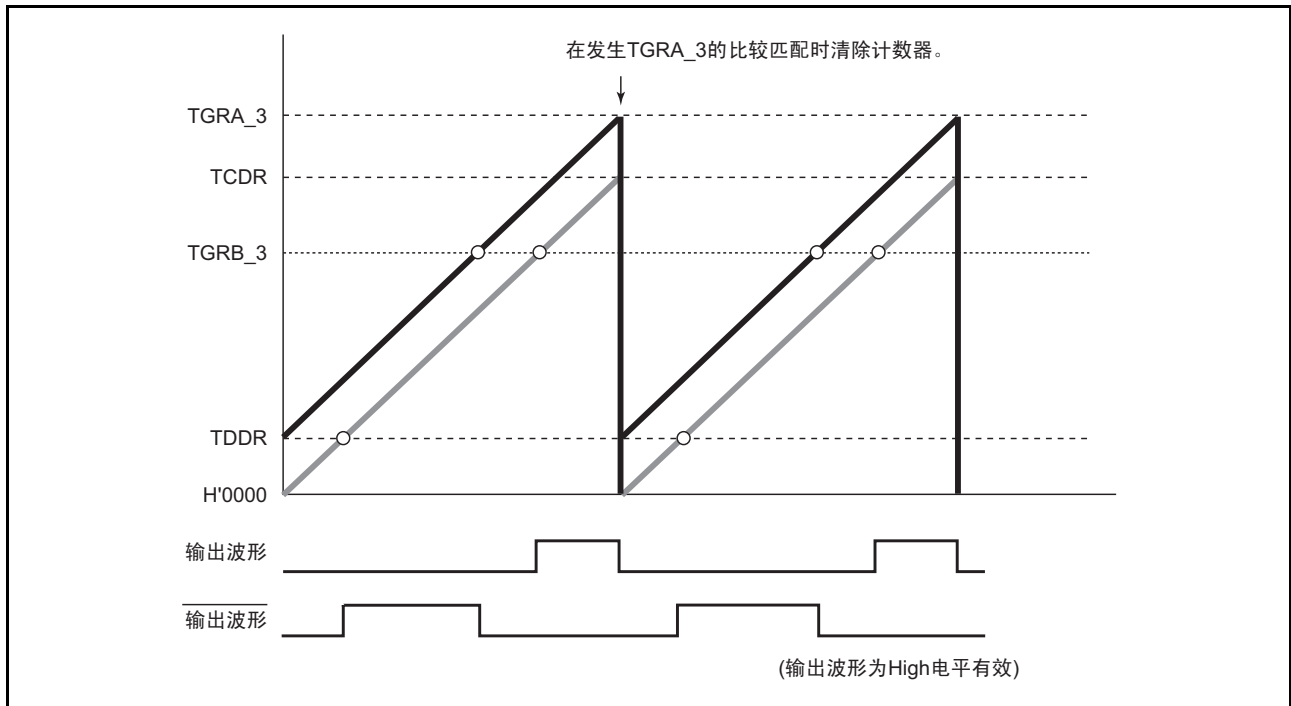


图 12.62 通过 TGRA_3 的比较匹配进行计数器清除的运行例子

(p) AC 同步马达（无刷 DC 马达）的驱动波形输出例子

在互补 PWM 模式中，能使用定时器的门控寄存器（TGCR）简单地控制无刷 DC 马达。使用 TGCR 无刷 DC 马达的驱动波形例子如图 12.63 ~ 图 12.66 所示。

在通过使用霍尔器件等检测到的外部信号进行 3 相无刷 DC 马达的输出相转换时，必须将 TGCR 的 FB 位置“0”。此时，将表示磁极位置的外部信号输入到通道 0 的定时器输入引脚 TIOC0A、TIOC0B 和 TIOC0C（必须通过通用输入 / 输出端口进行设定）。如果在 TIOC0A、TIOC0B、TIOC0C 这 3 个引脚产生边沿，输出的 ON/OFF 就自动进行转换。

在 FB 位为“1”的情况下，如果将 TGCR 的 UF 位、VF 位和 WF 位置“0”或者“1”，就进行输出的 ON/OFF 转换。

从互补 PWM 模式的 6 相输出引脚输出驱动波形。

对于此 6 相输出，能通过将 N 位或者 P 位置“1”，在 ON 输出时使用互补 PWM 模式的输出，进行斩波输出。如果 N 位或者 P 位为“0”，就为电平输出。

6 相输出的有效电平（ON 输出时的电平）与 N 位和 P 位的设定无关，能通过定时器的输出控制寄存器（TOCR）的 OLSN 位和 OLSP 位进行设定。

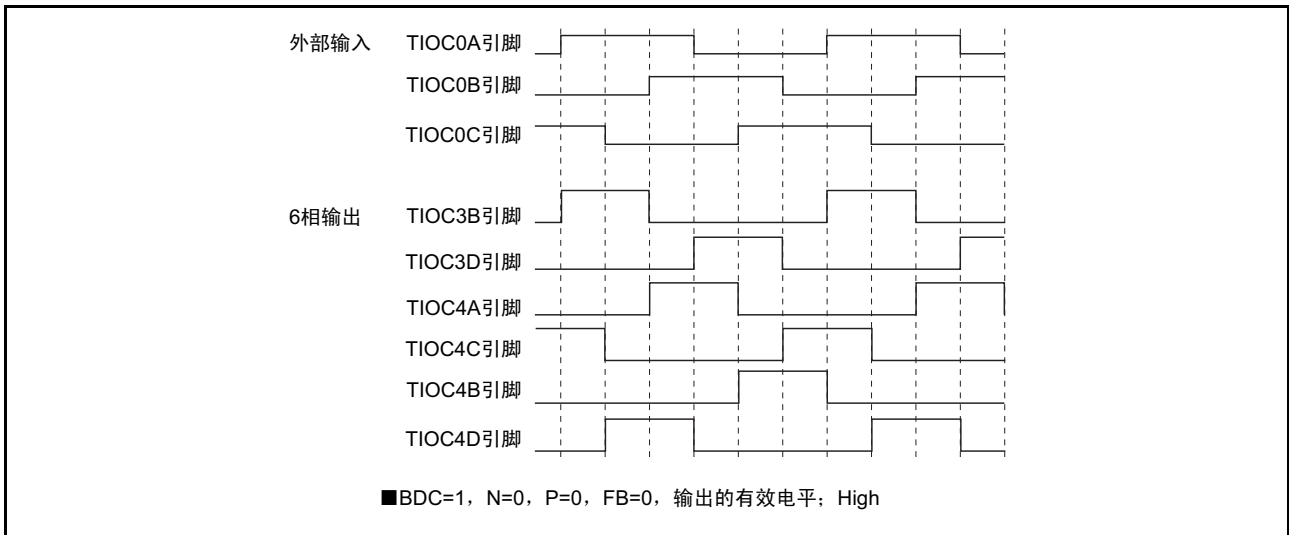


图 12.63 通过外部输入进行输出相转换的运行例子（1）

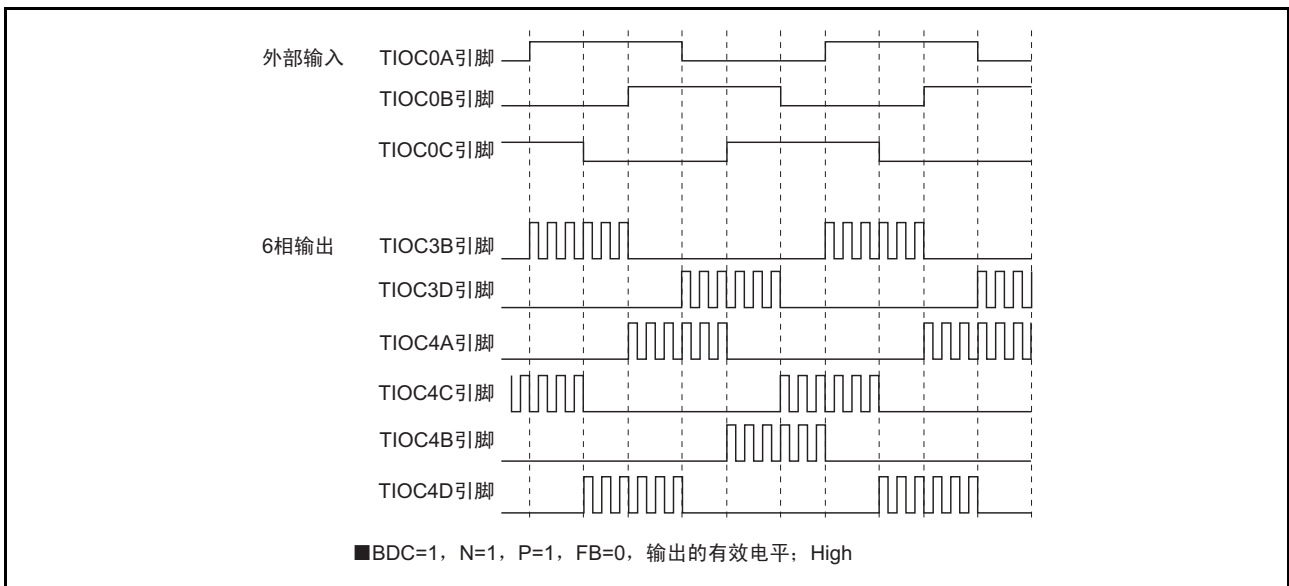


图 12.64 通过外部输入进行输出相转换的运行例子（2）

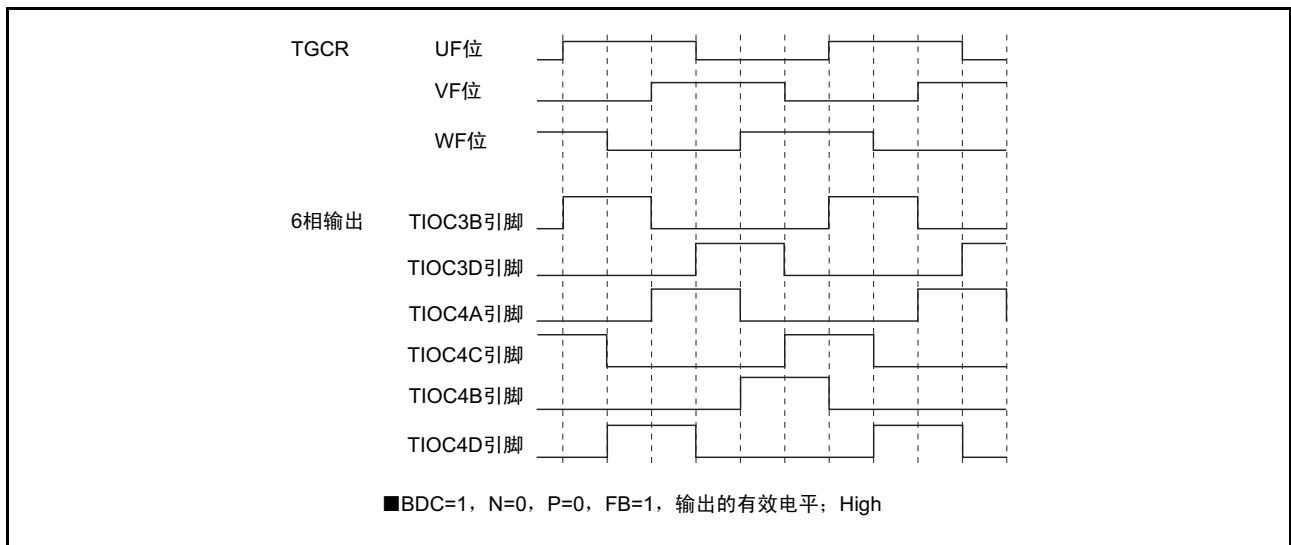


图 12.65 通过设定 UF 位、VF 位和 WF 位进行输出相转换的运行例子 (1)

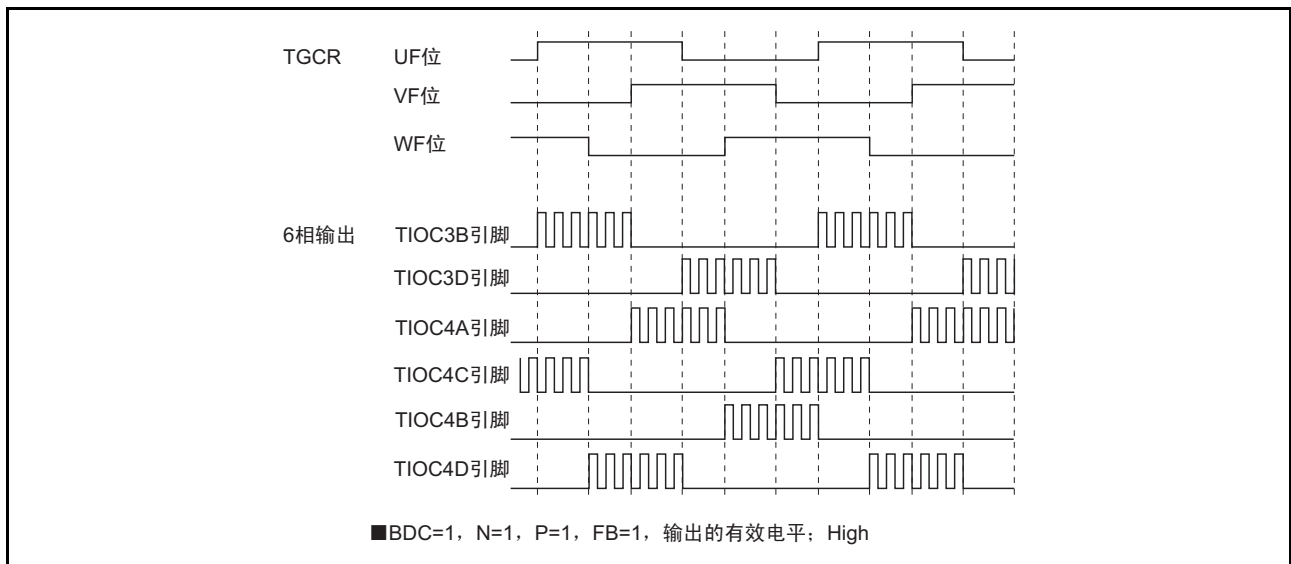


图 12.66 通过设定 UF 位、VF 位和 WF 位进行输出相转换的运行例子 (2)

(q) A/D 转换开始请求的设定

在互补 PWM 模式中，能使用 TGRA_3 的比较匹配、TCNT_4 的下溢（波谷）以及通道 3 和通道 4 以外通道的比较匹配，请求开始 A/D 转换。

如果使用 TGRA_3 的比较匹配设定开始请求，就能在 TCNT_3 的波峰开始 A/D 转换。

能通过将定时器的中断允许寄存器（TIER）的 TTGE 位置“1”，设定 A/D 转换的开始请求，并且能通过将 TIER_4 的 TTGE2 位置“1”，设定 TCNT_4 下溢（波谷）的 A/D 转换开始请求。

(3) 互补 PWM 模式的中断减少功能

能通过设定定时器的中断减少设定寄存器（TITCR），使通道 3 和通道 4 的 TGIA_3（波峰的中断）以及 TCIV_4（波谷的中断）最多减少 7 次中断。

能通过设定定时器的缓冲传送寄存器（TBTER），联动缓冲寄存器到暂存器 / 比较寄存器的数据传送来减少中断。有关和缓冲寄存器的联动，请参照“(c) 联动中断减少功能的缓冲传送控制”。

能通过设定定时器的 A/D 转换请求控制寄存器（TADCR），联动 A/D 转换开始请求延迟功能的 A/D 转换开始请求来减少中断。有关和 A/D 转换开始请求延迟功能的联动，请参照“12.4.9 A/D 转换开始请求的延迟功能”。

必须通过设定 TIER_3 寄存器和 TIER_4 寄存器，在禁止 TGIA_3 和 TCIV_4 中断请求的状态下并且在不发生由比较匹配产生的 TGFA_3 标志和 TCFV_4 标志置位的状态下，设定定时器的中断减少设定寄存器（TITCR）。在更改减少次数前，必须将 T3AEN 和 T4VEN 位置“0”，清除减少计数器。

(a) 中断减少功能的设定步骤例子

中断减少功能的设定步骤例子和中断减少次数的可更改期间分别如图 12.67 和图 12.68 所示。

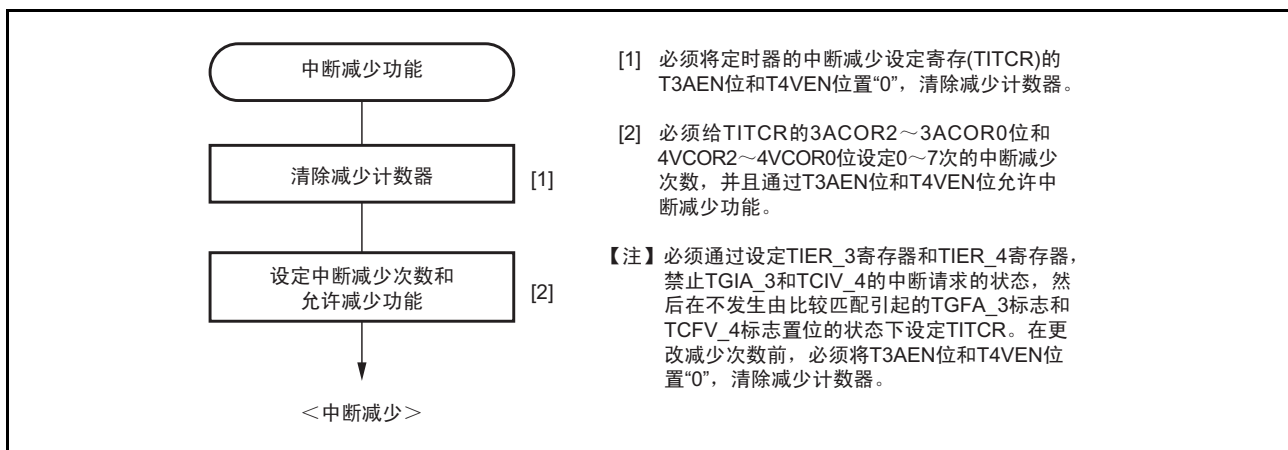


图 12.67 中断减少功能的设定步骤例子

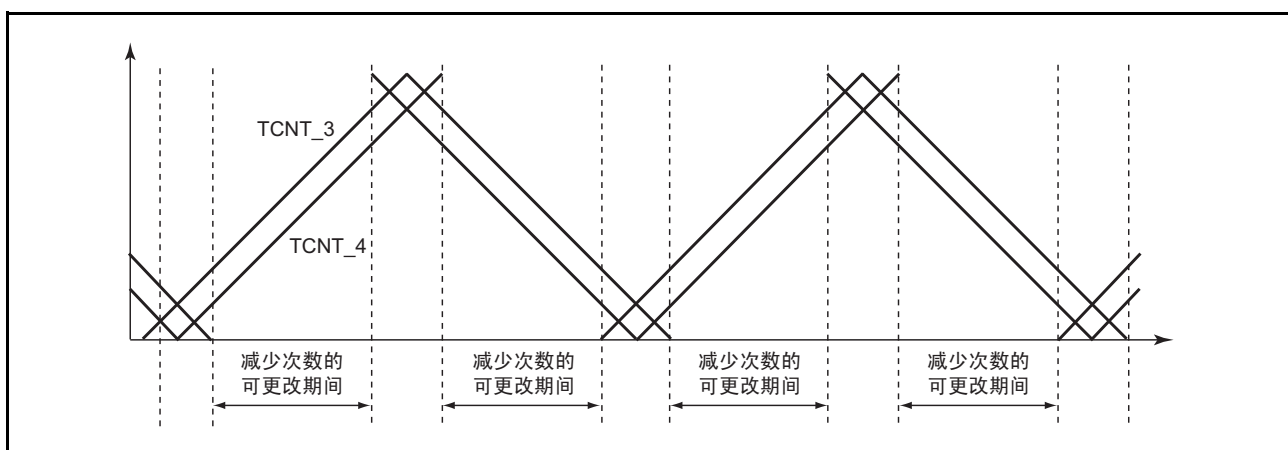


图 12.68 中断减少次数的可更改期间

(b) 中断减少功能的运行例子

通过定时器的中断减少设定寄存器（TITCR）的 3ACOR 位将中断减少次数设定为 3 次并且将 T3AEN 位置“1”时的 TGIA_3 中断减少的运行例子如图 12.69 所示。

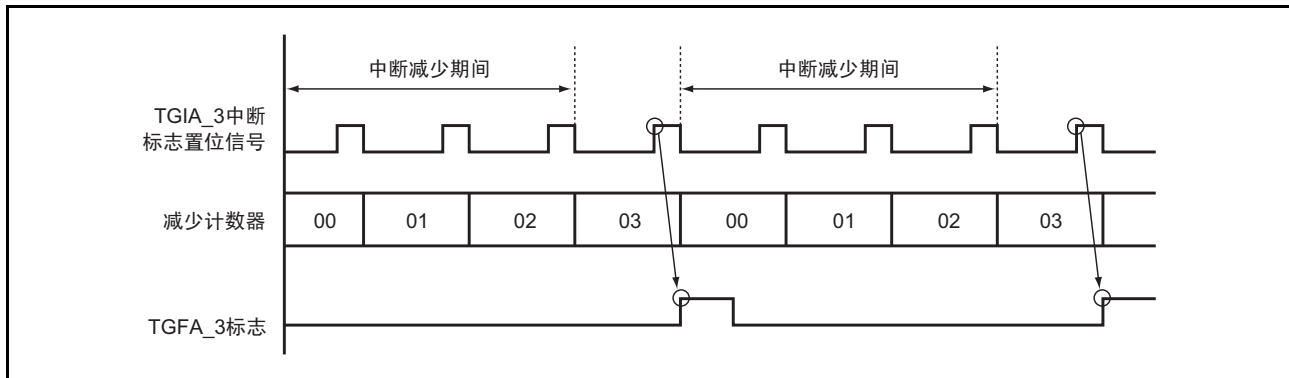


图 12.69 中断减少功能的运行例子

(c) 联动中断减少功能的缓冲传送控制

能通过设定定时器的缓冲传送设定寄存器（TBTER）的 BTE1 位和 BTE0 位，选择在互补 PWM 模式中是否进行缓冲寄存器到暂寄存器的缓冲传送以及是否联动中断减少功能。

设定为抑制缓冲传送（BTE1 位为“0”，BTE0 位为“1”）时的运行例子如图 12.70 所示。在此设定期间中，不进行缓冲寄存器到暂寄存器的传送。

设定为缓冲传送联动中断减少功能（BTE1 位为“1”，BTE0 位为“0”）时的运行例子如图 12.71 所示。此时，除了缓冲传送允许期间以外，不进行缓冲寄存器到暂寄存器的传送。根据发生中断后的缓冲存储器的改写时序，有缓冲存储器到暂寄存器以及暂寄存器到通用寄存器的 2 种传送时序。

当将定时器的中断减少设定寄存器（TITCR）的 T3AEN 位或者 T4VEN 位置“1”，或者将 T3AEN 位和 T4VEN 位置“1”时，各种情况的缓冲传送允许期间不同。TITCR 的 T3AEN 位、T4VEN 位的设定和缓冲传送允许期间的关系如图 12.72 所示。

【注】 此功能必须和中断减少功能配合使用。

在禁止中断减少功能时（将定时器的中断减少设定寄存器（TITCR）的 T3AEN 位和 T4VEN 位置“0”时，或者将 TITCR 的减少次数设定位（3ACOR 和 4VCOR）置“0”时），必须设定为缓冲传送不联动中断减少功能（将定时器的缓冲传送设定寄存器（TBTER）的 BTE1 位置“0”）。否则，就不进行缓冲传送。

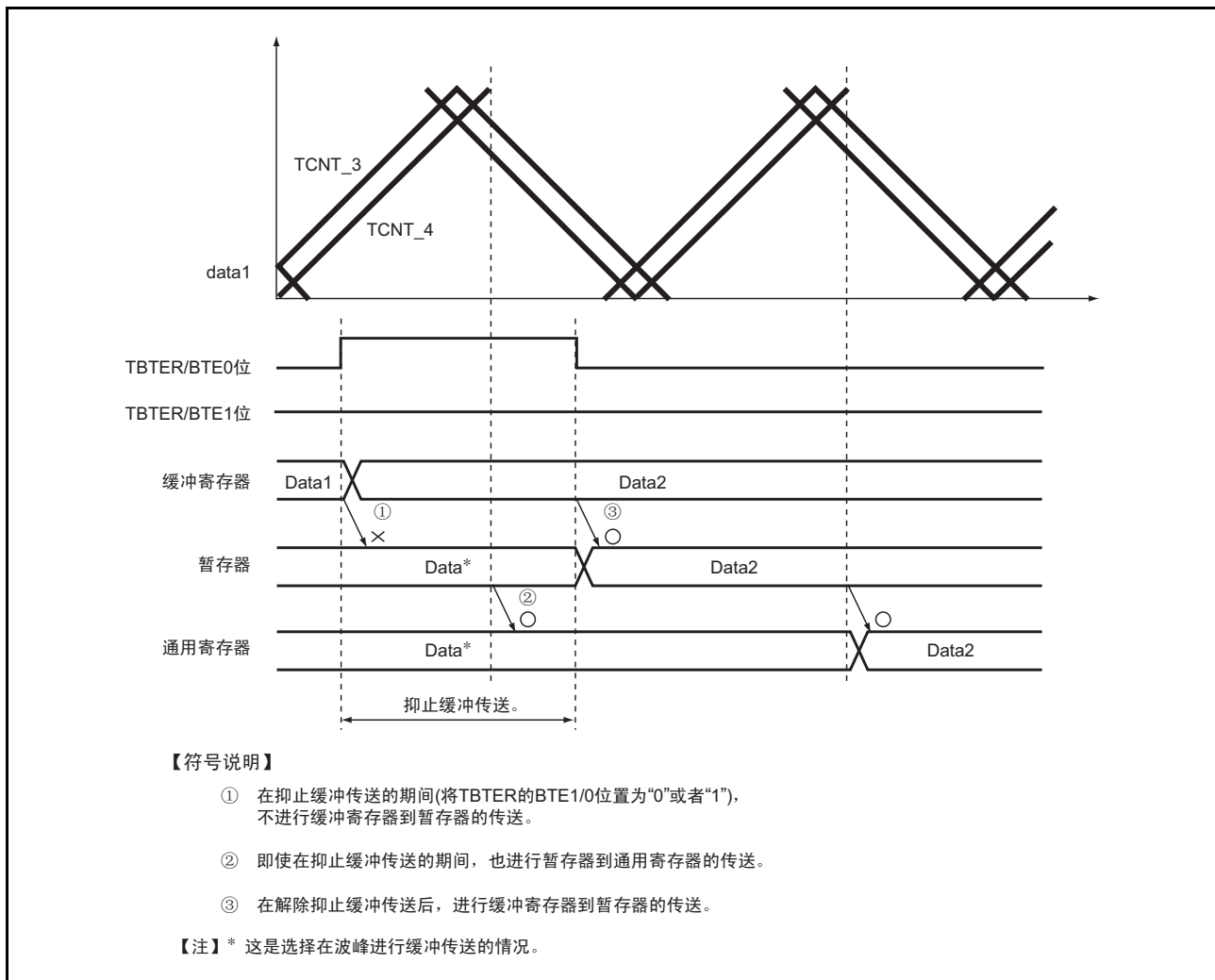


图 12.70 设定为抑制缓冲传送 (BTE1 位为 “0”, BTE0 位为 “1”) 时的运行例子

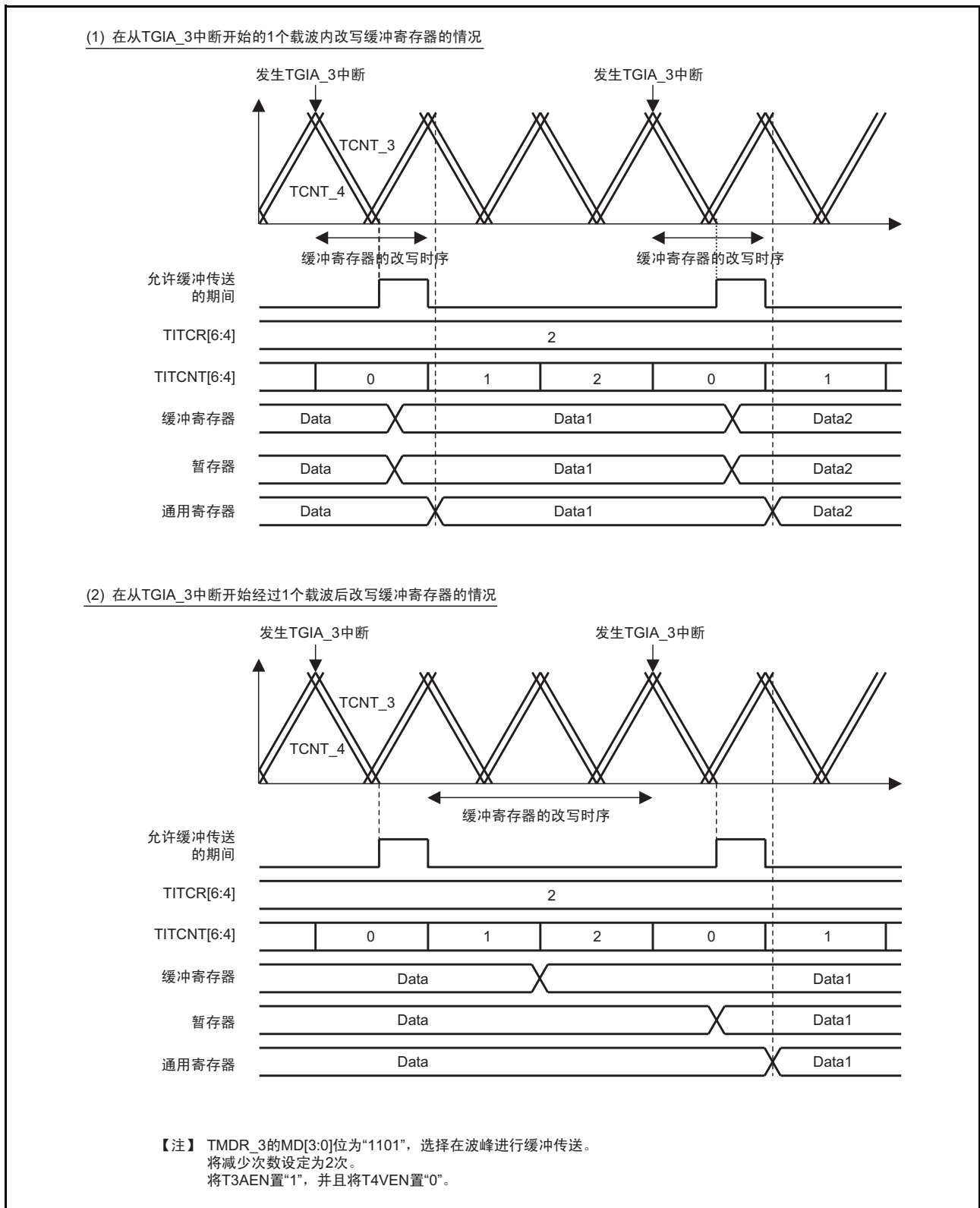


图 12.71 设定为缓冲传送联动中断减少功能（BTE1 位为“1”，BTE0 位为“0”）时的运行例子

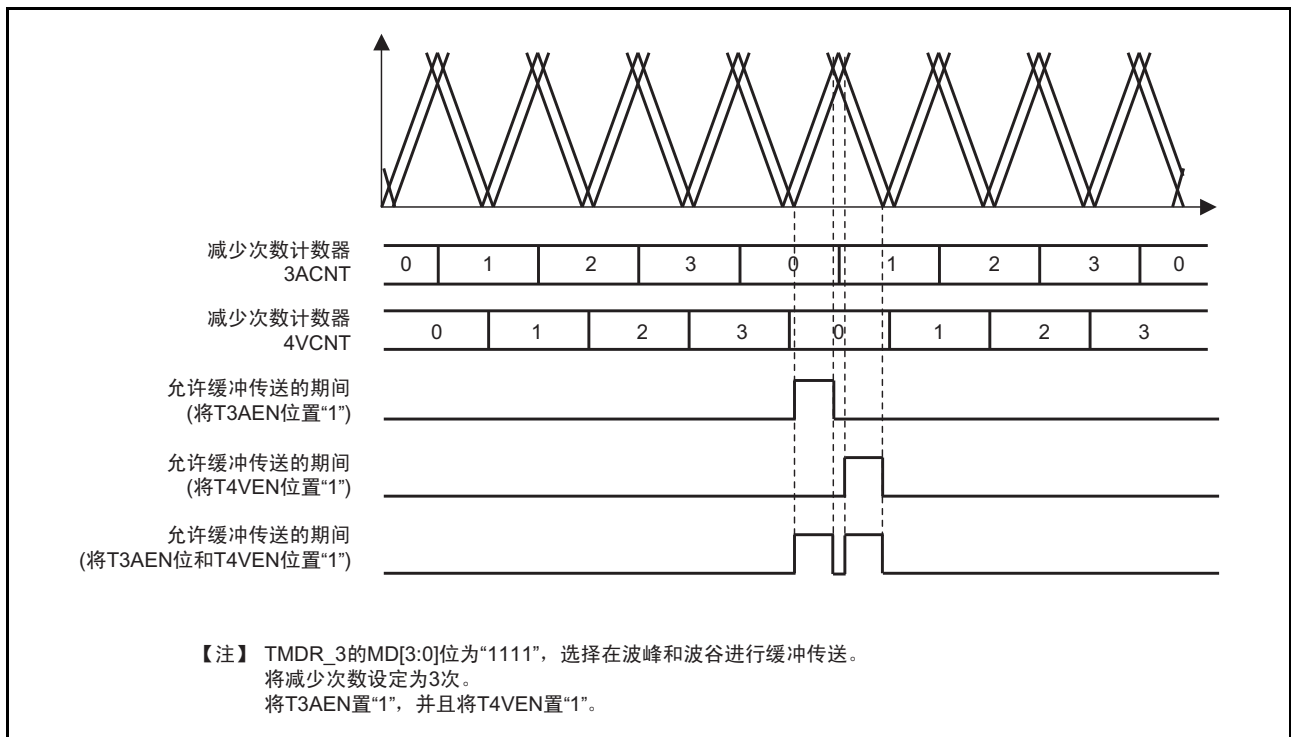


图 12.72 定时器的中断减少设定寄存器（TITCR）的 T3AEN 位和 T4VEN 位的设定与缓冲传送允许期间的关系

(4) 互补 PWM 模式的输出保护功能

互补 PWM 模式的输出具有以下保护功能。

(a) 寄存器和计数器的误写防止功能

用于互补 PWM 模式的寄存器和计数器中，除了能随时改写的缓冲寄存器以外，能通过设定定时器的读写允许寄存器（TRWER）的 RWE 位，选择允许或者禁止 CPU 存取模式寄存器、控制寄存器、比较寄存器和计数器。对象寄存器为通道 3 和通道 4 的一部分寄存器，适用于以下寄存器：

TCR_3 和 TCR_4、TMDR_3 和 TMDR_4、TIORH_3 和 TIORH_4、TIORL_3 和 TIORL_4、TIER_3 和 TIER_4、TCNT_3 和 TCNT_4、TGRA_3 和 TGRA_4、TGRB_3 和 TGRB_4、TOER、TOCR、TGCR、TCDR、TDDR 共计 21 个寄存器。

通过此功能，能设定为禁止 CPU 存取模式寄存器、控制寄存器和计数器，以防止因 CPU 失控而发生的误写。在禁止存取的状态下读对象寄存器时，读取值为不定值，写操作无效。

12.4.9 A/D 转换开始请求的延迟功能

能通过设定通道 4 的定时器的 A/D 转换开始请求控制寄存器 (TADCR)、定时器的 A/D 启动请求周期寄存器 (TADCORA_4 和 TADCORB_4) 和定时器的 A/D 启动请求周期缓冲寄存器 (TADCOBRA_4、TADCOBRB_4)，进行 A/D 转换的开始请求。

A/D 转换开始请求的延迟功能是将 TCNT_4 和 TADCORA_4、TADCORB_4 进行比较，如果 TCNT_4 和 TADCORA_4、TADCORB_4 相同，就进行各自的 A/D 转换开始请求 (TRG4AN 和 TRG4BN)。

能通过设定 TADCR 的 ITA3AE、ITA4VE、ITB3AE 和 ITB4VE 位，联动中断减少功能来减少 A/D 转换的开始请求 (TRG4AN 和 TRG4BN)。

- A/D 转换开始请求延迟功能的设定步骤例子

A/D 转换开始请求延迟功能的设定步骤例子如图 12.73 所示。

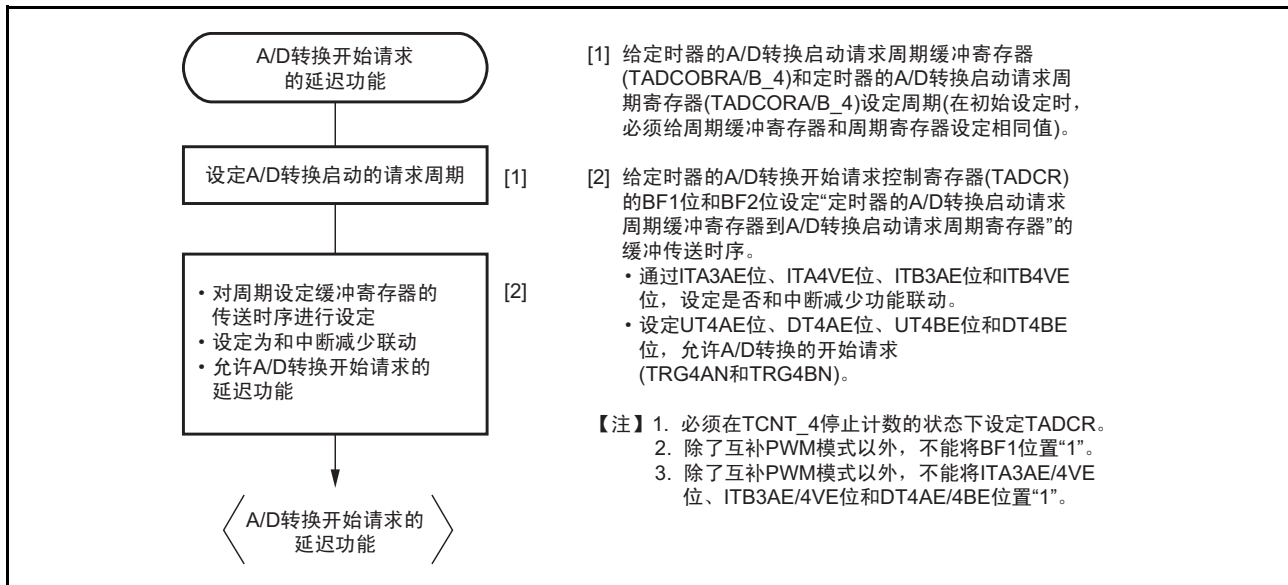


图 12.73 A/D 转换开始请求延迟功能的设定步骤例子

- A/D 转换开始请求延迟功能的基本运行例子

将缓冲传送时序设定为 TCNT_4 的波谷并且在 TCNT_4 进行递减计数时输出 A/D 转换开始请求信号 (TRG4AN) 时的 A/D 转换开始请求信号 (TRG4AN) 的基本运行例子如图 12.74 所示。

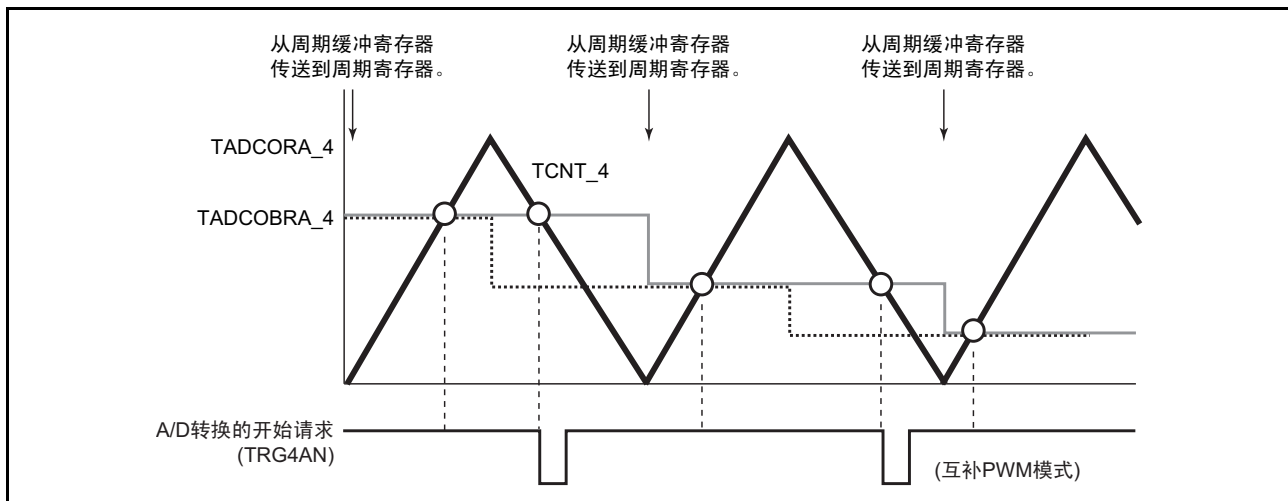


图 12.74 A/D 转换开始请求信号 (TRG4AN) 的基本运行例子

- 缓冲传送

通过将数据写到定时器的 A/D 启动请求周期设定缓冲寄存器 (TADCOBRA/B_4)，更新定时器的 A/D 启动请求周期设定寄存器 (TADCORA/B_4) 的数据。能通过设定定时器的 A/D 转换开始请求控制寄存器 (TADCR_4) 的 BF1 位和 BF0 位，选择定时器的 A/D 启动请求周期设定缓冲寄存器到定时器的 A/D 启动请求周期设定寄存器的传送时序。

- 联动中断减少功能的 A/D 转换开始请求延迟功能

能通过设定定时器的 A/D 转换开始请求控制寄存器 (TADCR) 的 ITA3AE、ITA4VE、ITB3AE 和 ITB4VE 位，联动中断减少功能进行 A/D 转换的开始请求 (TRG4AN 和 TRG4BN)。在 TCNT_4 进行递增计数和递减计数时允许 TRG4AN 输出并且联动中断减少功能时的 A/D 转换开始请求信号 (TRG4AN) 的运行例子如图 12.75 所示。

在 TCNT_4 进行递增计数时允许 TRG4AN 输出并且联动中断减少功能时的 A/D 转换开始请求信号 (TRG4AN) 的运行例子如图 12.76 所示。

【注】 此功能必须和中断减少功能配合使用。

在禁止中断减少功能时 (将定时器的中断减少设定寄存器 (TITCR) 的 T3AEN 位和 T4VEN 位置 “0” 时，或者将 TITCR 的减少次数设定 (3ACOR 和 4VCOR) 置 “0” 时)，必须设定为不联动中断减少功能 (将定时器的 A/D 转换开始请求控制寄存器 (TADCR) 的 ITA3AE、ITA4VE、ITB3AE 和 ITB4VE 位置 “0”)。

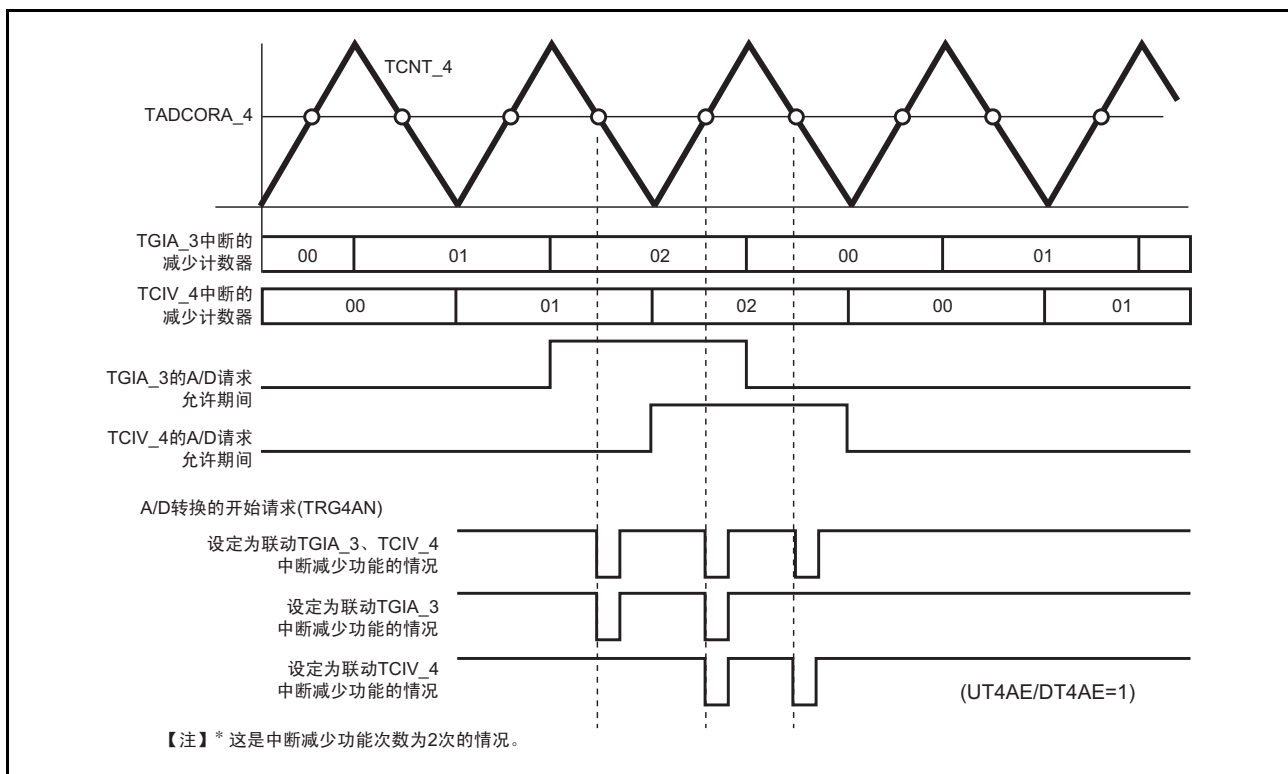


图 12.75 联动中断减少功能时的 A/D 转换开始请求信号 (TRG4AN) 的运行例子

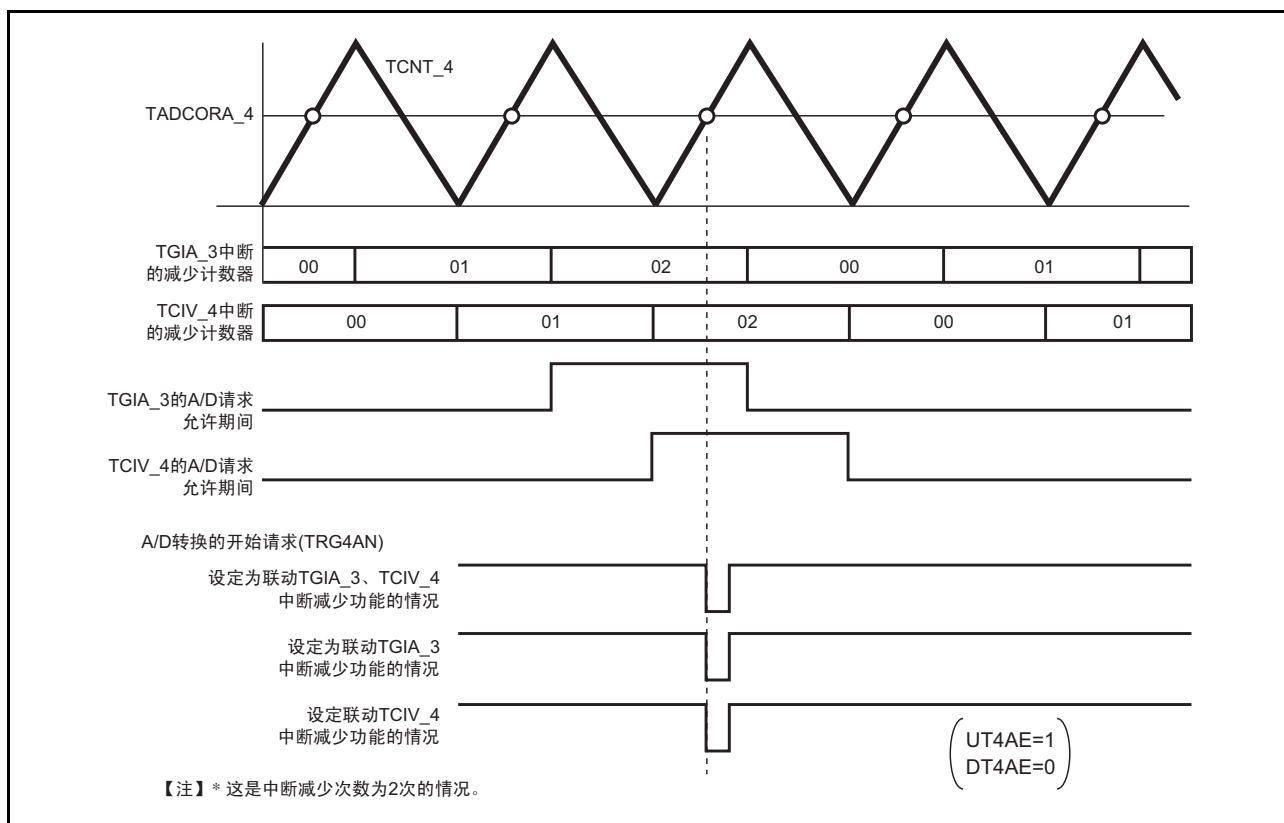


图 12.76 联动中断减少功能时的 A/D 转换开始请求信号 (TRG4AN) 的运行例子

12.4.10 在互补 PWM 的“波峰/波谷”进行的 TCNT 捕捉运行

在互补 PWM 运行时，在“波峰、波谷、波峰和波谷”将 TCNT 的值保存到 TGR，通过 TIOR 选择要捕捉到 TGR 的时序转换。

TCNT 用作自由运行计数器而不被清除并且在设定的“波峰、波谷”对 TGR 进行捕捉的运行例子如图 12.77 所示。

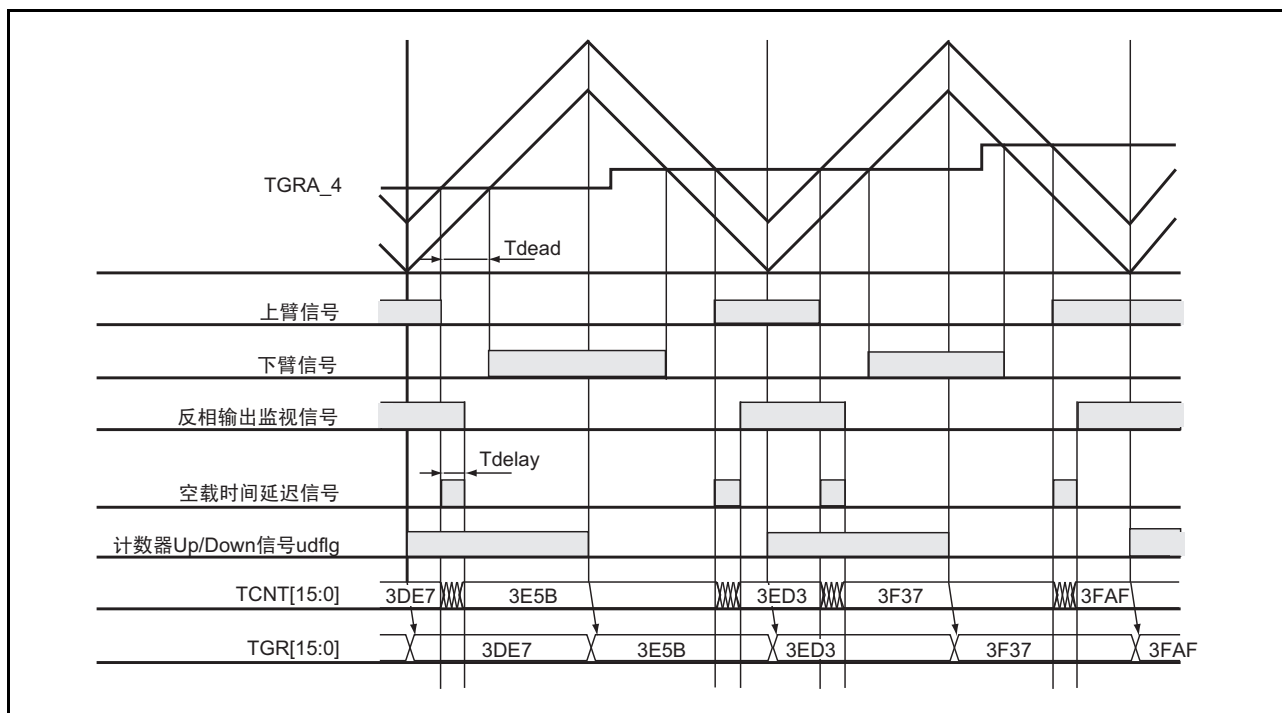


图 12.77 在互补 PWM 的“波峰/波谷”进行的 TCNT 捕捉运行

12.5 中断源

12.5.1 中断源和优先级

此模块的中断源有 TGR 的输入捕捉 / 比较匹配、TCNT 的上溢和 TCNT 的下溢共 3 种中断源。因为各中断源有各自专用的状态标志和允许 / 禁止位，所以能独立允许或者禁止中断请求信号的产生。

如果发生中断源，就将 TSR 对应的状态标志置“1”。此时，如果 TIER 对应的允许 / 禁止位已被置“1”，就请求中断。通过将状态标志清“0”，解除中断请求。

能通过中断控制器更改通道之间的优先级，通道内的优先级是固定的，详细内容请参照“7. 中断控制器”。此模块的中断源一览表如表 12.55 所示。

表 12.55 多功能定时器脉冲单元 2 的中断源

通道	名称	中断源	中断标志	直接存储器存取控制器的启动	优先级
0	TGIA_0	TGRA_0 的输入捕捉 / 比较匹配	TGFA_0	能	高 ↑
	TGIB_0	TGRB_0 的输入捕捉 / 比较匹配	TGFB_0	不能	
	TGIC_0	TGRC_0 的输入捕捉 / 比较匹配	TGFC_0	不能	
	TGID_0	TGRD_0 的输入捕捉 / 比较匹配	TGFD_0	不能	
	TCIV_0	TCNT_0 的上溢	TCFV_0	不能	
	TGIE_0	TGRE_0 的比较匹配	TGFE_0	不能	
	TGIF_0	TGRF_0 的比较匹配	TGFF_0	不能	
1	TGIA_1	TGRA_1 的输入捕捉 / 比较匹配	TGFA_1	能	↑ ↓ 低
	TGIB_1	TGRB_1 的输入捕捉 / 比较匹配	TGFB_1	不能	
	TCIV_1	TCNT_1 的上溢	TCFV_1	不能	
	TCIU_1	TCNT_1 的下溢	TCFU_1	不能	
2	TGIA_2	TGRA_2 的输入捕捉 / 比较匹配	TGFA_2	能	
	TGIB_2	TGRB_2 的输入捕捉 / 比较匹配	TGFB_2	不能	
	TCIV_2	TCNT_2 的上溢	TCFV_2	不能	
	TCIU_2	TCNT_2 的下溢	TCFU_2	不能	
3	TGIA_3	TGRA_3 的输入捕捉 / 比较匹配	TGFA_3	能	
	TGIB_3	TGRB_3 的输入捕捉 / 比较匹配	TGFB_3	不能	
	TGIC_3	TGRC_3 的输入捕捉 / 比较匹配	TGFC_3	不能	
	TGID_3	TGRD_3 的输入捕捉 / 比较匹配	TGFD_3	不能	
	TCIV_3	TCNT_3 的上溢	TCFV_3	不能	
4	TGIA_4	TGRA_4 的输入捕捉 / 比较匹配	TGFA_4	能	
	TGIB_4	TGRB_4 的输入捕捉 / 比较匹配	TGFB_4	不能	
	TGIC_4	TGRC_4 的输入捕捉 / 比较匹配	TGFC_4	不能	
	TGID_4	TGRD_4 的输入捕捉 / 比较匹配	TGFD_4	不能	
	TCIV_4	TCNT_4 的上溢 / 下溢	TCFV_4	不能	

【注】表示刚复位后的初始状态。能通过中断控制器更改通道之间的优先级。

(1) 输入捕捉 / 比较匹配中断

在通过各通道的 TGR 输入捕捉 / 比较匹配将 TSR 的 TGF 标志置“1”时，如果 TIER 的 TGIE 位已被置“1”，就请求中断。通过将 TGF 标志清“0”来解除中断请求。在此模块中，通道 0 有 6 个，通道 3 和通道 4 各有 4 个，通道 1 和通道 2 各有 2 个，共计 18 个输入捕捉 / 比较匹配中断。在进行输入捕捉时，不将通道 0 的 TGFE₀ 标志和 TGFF₀ 标志置位。

(2) 上溢中断

在通过各通道的 TCNT 上溢将 TSR 的 TCFV 标志置“1”时，如果 TIER 的 TCIEV 位已被置“1”，就请求中断。通过将 TCFV 标志清“0”来解除中断请求。在此模块中，各通道各有 1 个，共计 5 个上溢中断。

(3) 下溢中断

在通过各通道的 TCNT 下溢将 TSR 的 TCFU 标志置“1”时，如果 TIER 的 TCIEU 位已被置“1”，就请求中断。通过将 TCFU 标志清“0”来解除中断请求。在此模块中，通道 1 和通道 2 各有 1 个，共计 2 个下溢中断。

12.5.2 直接存储器存取控制器的启动

能通过各通道的 TGRA 的输入捕捉 / 比较匹配中断来启动直接存储器存取控制器，详细内容请参照“11. 直接存储器存取控制器”。

在此模块中，通道 0 ~ 4 各有 1 个，共计 5 个 TGRA 寄存器的输入捕捉 / 比较匹配中断，能将这些输入捕捉 / 比较匹配中断作为直接存储器存取控制器的启动源。

12.5.3 A/D 转换器的启动

在此模块中，能通过以下 3 种方法启动 A/D 转换器。

各中断源和 A/D 转换开始请求的对应如表 12.56 所示。

(1) 在 TGRA 的输入捕捉 / 比较匹配时以及在互补 PWM 模式的 TCNT_4 波谷进行的 A/D 转换器启动

能通过各通道的 TGRA 输入捕捉 / 比较匹配来启动 A/D 转换器。另外，如果在将 TIER_4 的 TTGE2 位置“1”的状态下进行互补 PWM 运行，也能在 TCNT_4 的波谷（TCNT_4 为“H'0000”）启动 A/D 转换器。

通过以下所示的条件，对 A/D 转换器产生 A/D 转换开始请求 TRGAN：

- 在通过各通道的 TGRA 输入捕捉 / 比较匹配将 TSR 的 TGFA 标志置“1”的情况下 TIER 的 TTGE 位已被置“1”时
- 在将 TIER_4 的 TTGE2 位置“1”的状态下进行互补 PWM 运行并且 TCNT_4 变为波谷（TCNT_4 为“H'0000”）时

在以上条件下，如果已在 A/D 转换器侧选择此模块的转换开始触发 TRGAN，就开始进行 A/D 转换。

(2) 通过 TCNT_0 和 TGRE_0 的比较匹配进行的 A/D 转换器启动

能通过通道 0 的 TCNT_0 和 TGRE_0 的比较匹配，产生 A/D 转换开始请求 TRG0N 并且启动 A/D 转换器。

在通过通道 0 的 TCNT_0 和 TGRE_0 的比较匹配将 TSR2_0 的 TGFE 标志置“1”时，如果 TIER2_0 的 TTGE2 位已被置“1”，就对 A/D 转换器产生 A/D 转换开始请求 TRG0N。此时，如果已在 A/D 转换器侧选择此模块的转换开始触发 TRG0N，就开始进行 A/D 转换。

(3) 通过 A/D 转换开始请求的延迟功能进行的 A/D 转换器启动

如果将 A/D 转换开始请求控制寄存器（TADCR）的 UT4AE、DT4AE、UT4BE、DT4BE 位置“1”，就能在 TADCORA、TADCORB 和 TCNT_4 相同时产生 TRG4AN 或者 TRG4BN，启动 A/D 转换器。详细内容请参照“12.4.9 A/D 转换开始请求的延迟功能”。

当发生 TRG4AN 时，如果已在 A/D 转换器侧选择此模块的转换开始触发 TRG4AN，就开始进行 A/D 转换。当发生 TRG4BN 时，如果已在 A/D 转换器侧选择此模块的转换开始触发 TRG4BN，就开始进行 A/D 转换。

表 12.56 各中断源和 A/D 转换开始请求的对应

对象	中断源	A/D 转换开始请求
TGRA_0 和 TCNT_0	输入捕捉 / 比较匹配	TRGAN
TGRA_1 和 TCNT_1		
TGRA_2 和 TCNT_2		
TGRA_3 和 TCNT_3		
TGRA_4 和 TCNT_4		
TCNT_4	互补 PWM 模式中的 TCNT_4 波谷	
TGRE_0 和 TCNT_0	比较匹配	TRG0N
TADCORA 和 TCNT_4		TRG4AN
TADCORB 和 TCNT_4		TRG4BN

12.6 运行时序

12.6.1 输入 / 输出时序

(1) TCNT 的计数时序

内部时钟运行时的 TCNT 计数时序如图 12.78 所示，外部时钟运行（正常模式）和外部时钟运行（相位计数模式）时的 TCNT 计数时序分别如图 12.79 和图 12.80 所示。

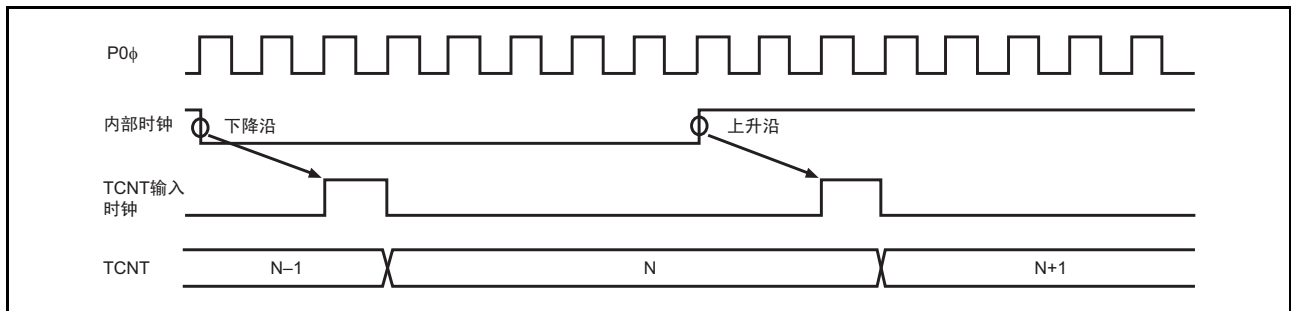


图 12.78 内部时钟运行时的计数时序

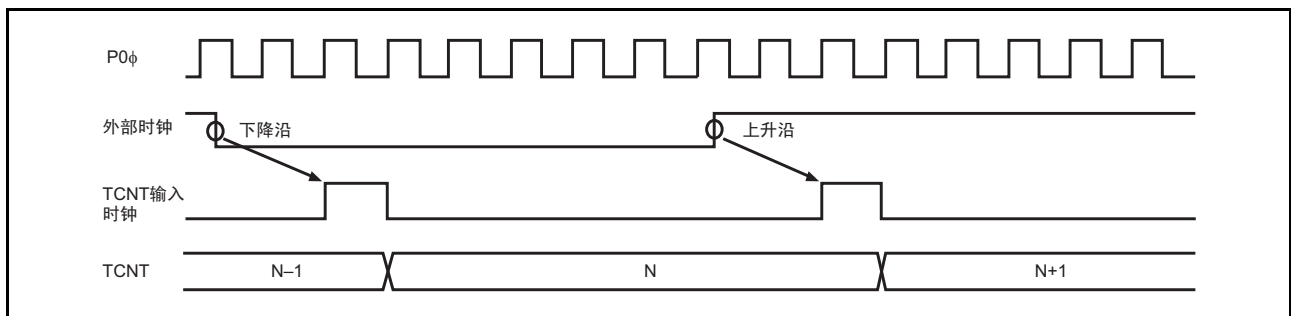


图 12.79 外部时钟运行时的计数时序

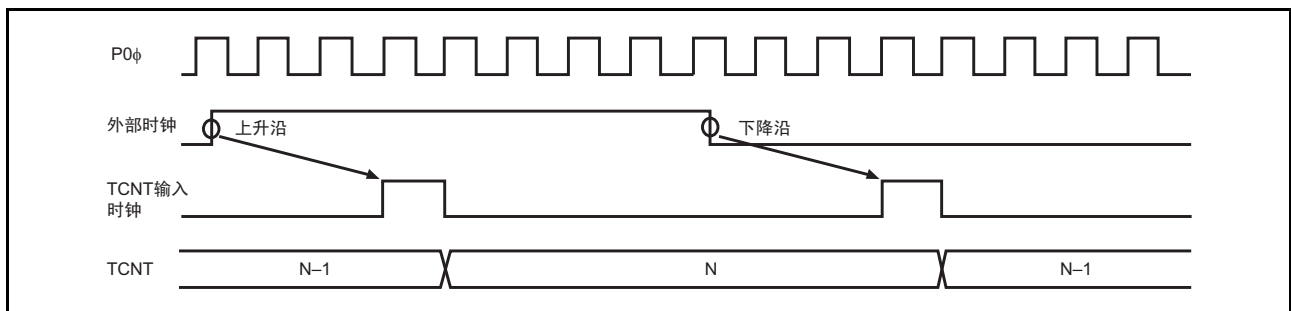


图 12.80 外部时钟运行时的计数时序（相位计数模式）

(2) 输出比较的输出时序

在 TCNT 和 TGR 的值相同的最后状态（在更新 TCNT 相同后的计数值时）产生比较匹配信号。当产生比较匹配信号时，将 TIOR 设定的输出值输出到输出比较的输出引脚（TIOC 引脚）。从 TCNT 和 TGR 的值相同到产生 TCNT 输入时钟为止，不产生比较匹配信号。

输出比较的输出时序（正常模式、PWM 模式）和输出比较的输出时序（互补 PWM 模式、复位同步 PWM 模式）分别如图 12.81 和图 12.82 所示。

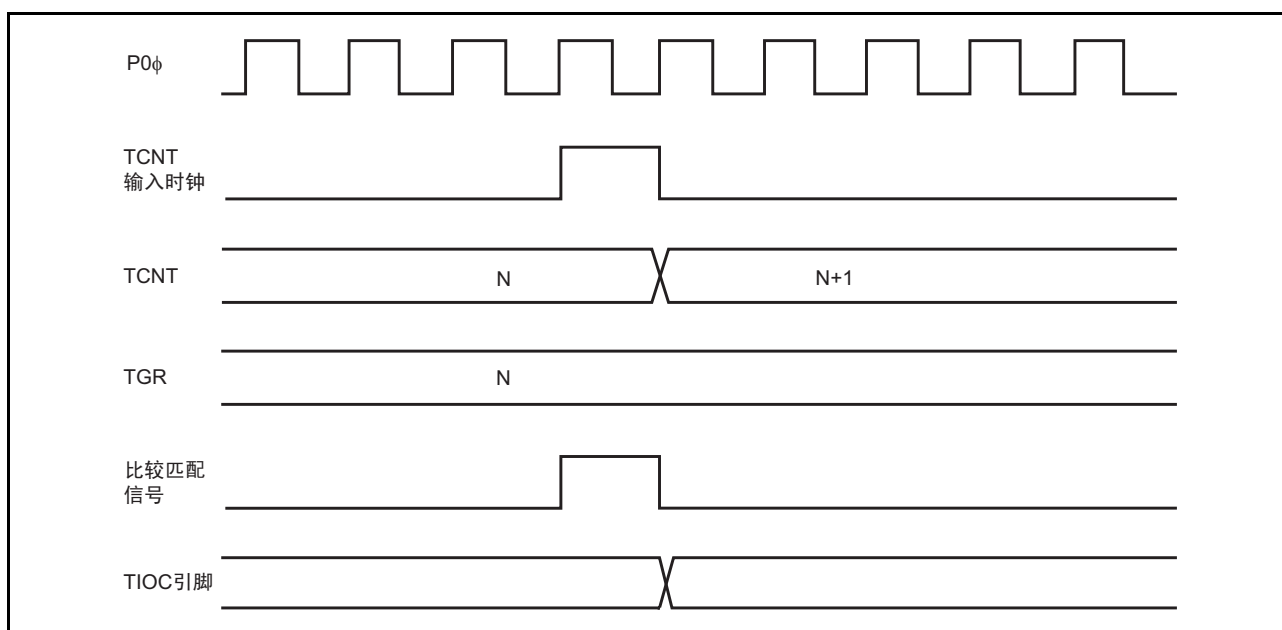


图 12.81 输出比较的输出时序（正常模式、PWM 模式）

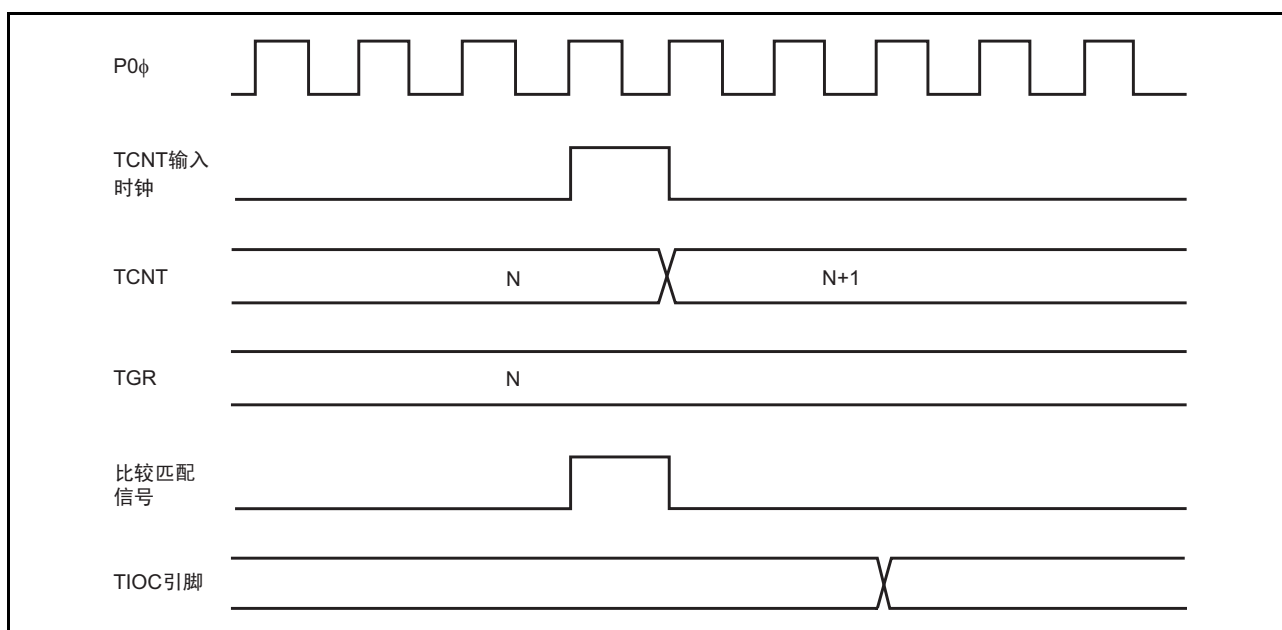


图 12.82 输出比较的输出时序（互补 PWM 模式、复位同步 PWM 模式）

(3) 输入捕捉信号的时序

输入捕捉的时序如图 12.83 所示。

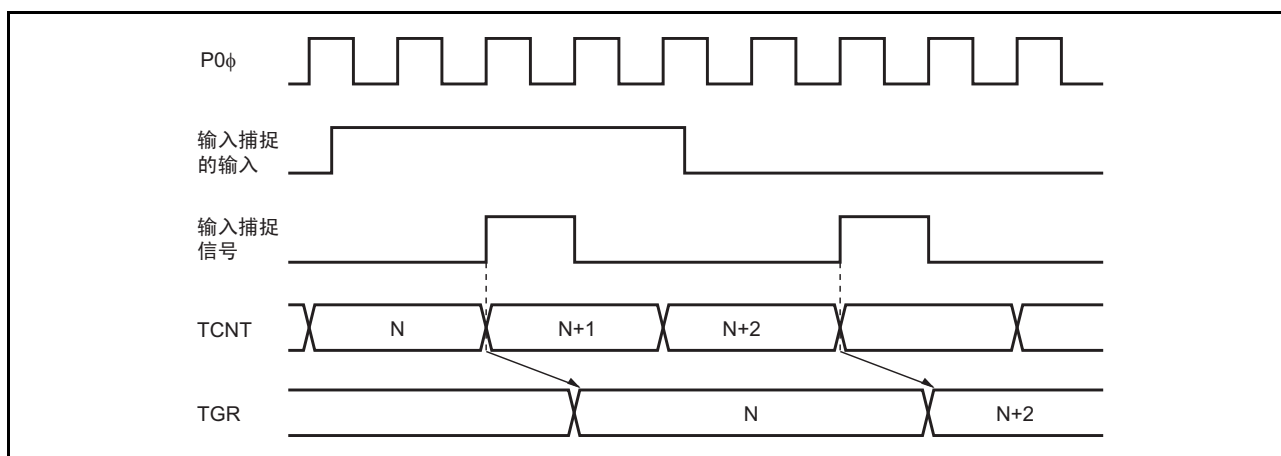


图 12.83 输入捕捉的输入信号时序

(4) 通过比较匹配 / 输入捕捉进行的计数器清除时序

指定通过比较匹配进行计数器清除时的时序如图 12.84 所示，
指定通过输入捕捉进行计数器清除时的时序如图 12.85 所示。

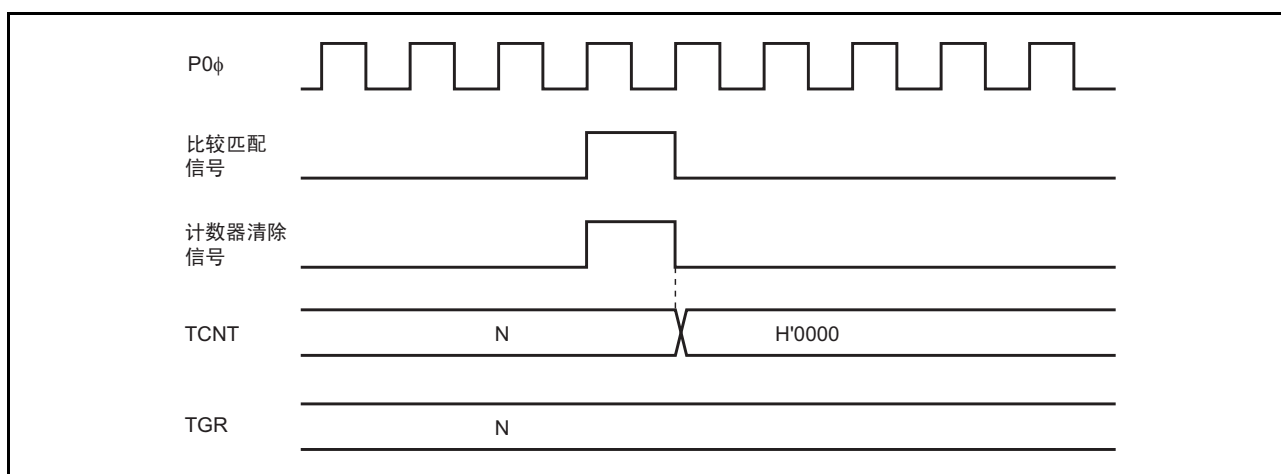


图 12.84 计数器清除的时序（比较匹配）

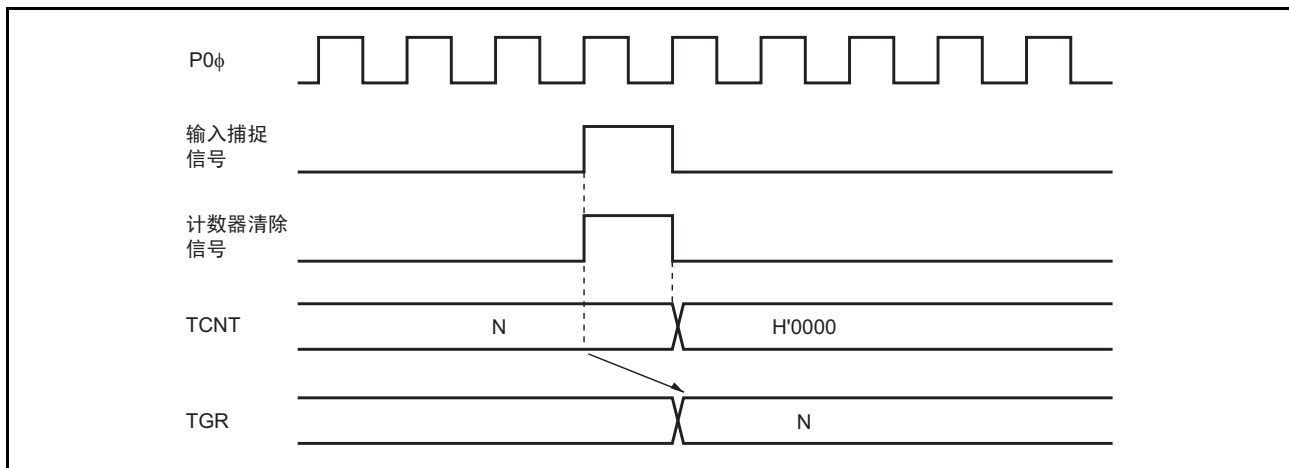


图 12.85 计数器清除的时序（输入捕捉）

(5) 缓冲运行的时序

缓冲运行的时序如图 12.86 ~ 图 12.88 所示。

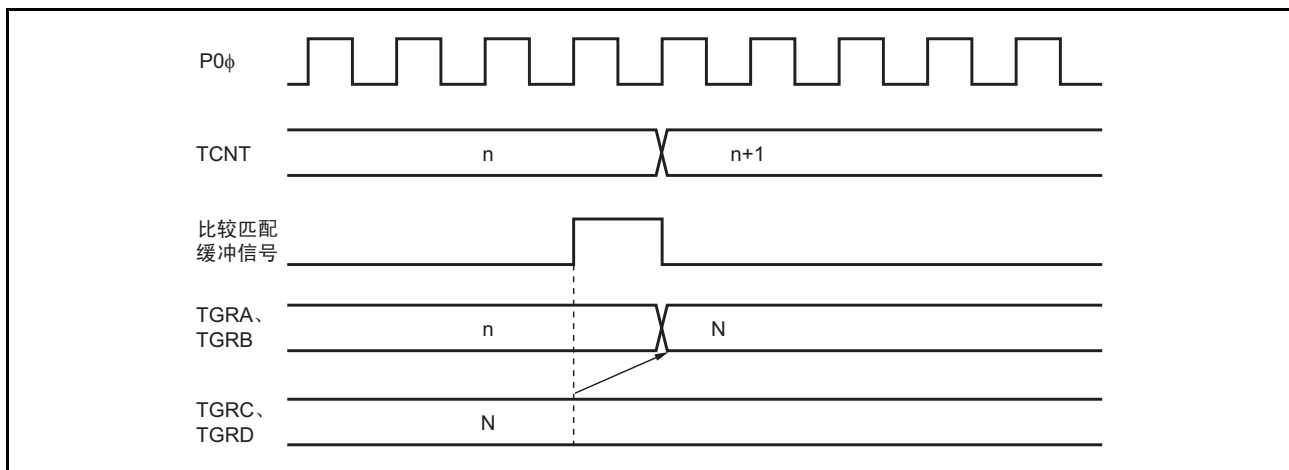


图 12.86 缓冲运行的时序（比较匹配）

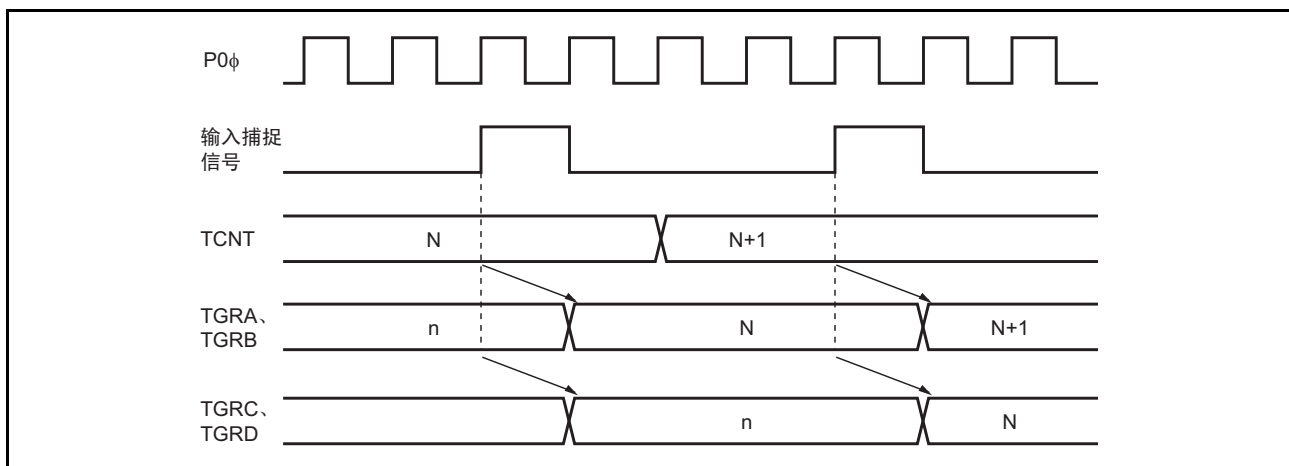


图 12.87 缓冲运行的时序（输入捕捉）

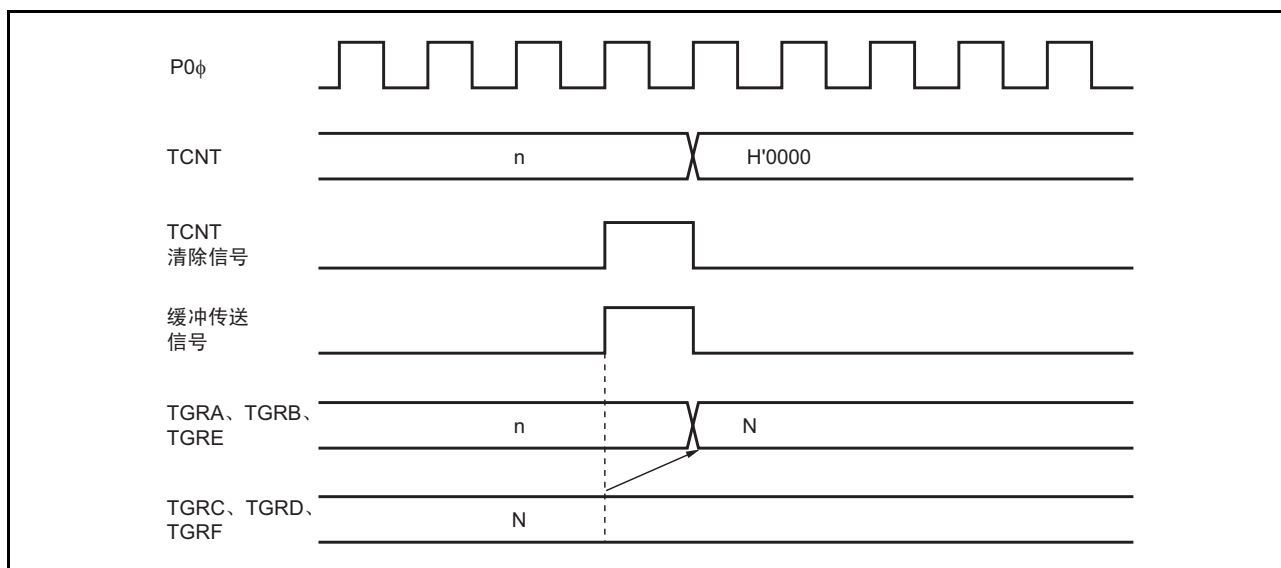


图 12.88 缓冲运行的时序（清除 TCNT 的情况）

(6) 缓冲传送的时序（互补 PWM 模式）

互补 PWM 模式中的缓冲传送时序如图 12.89 ~ 图 12.91 所示。

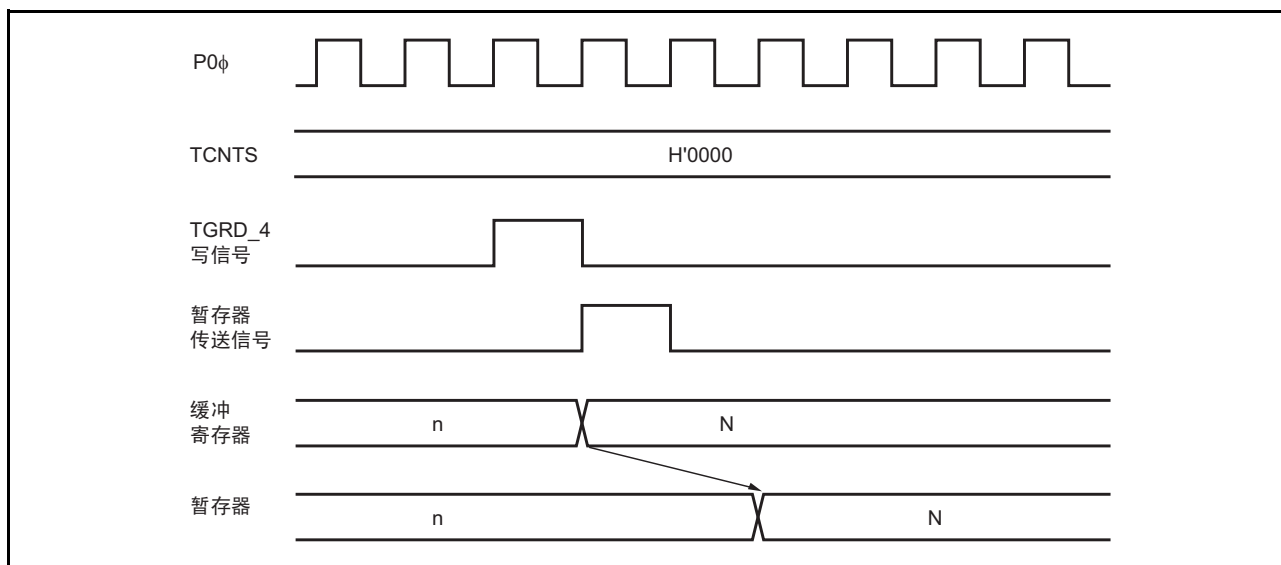


图 12.89 缓冲寄存器到暂存器的传送时序（TCNTS 停止）

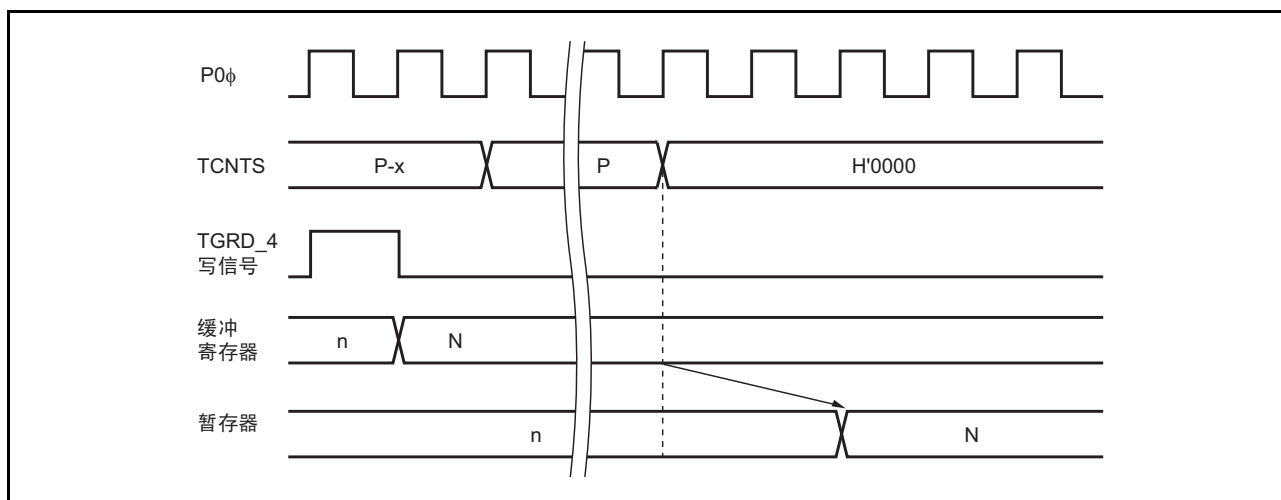


图 12.90 缓冲寄存器到暂存器的传送时序 (TCNTS 正在运行)

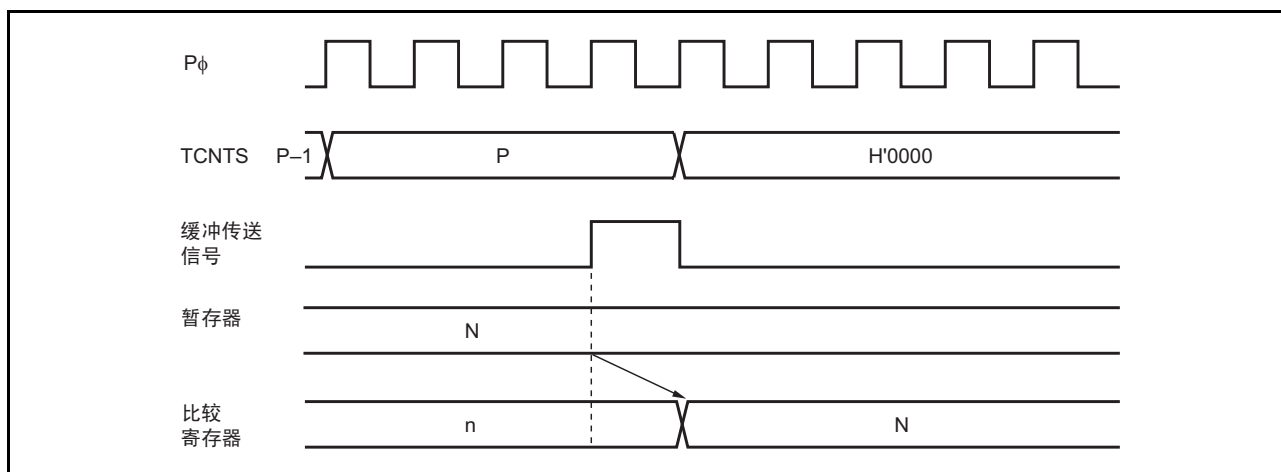


图 12.91 暂存器到比较寄存器的传送时序

12.6.2 中断信号的时序

(1) 比较匹配时的 TGF 标志的置位时序

由比较匹配产生的 TSR 的 TGF 标志置位时序和 TGI 中断请求信号的时序如图 12.92 所示。

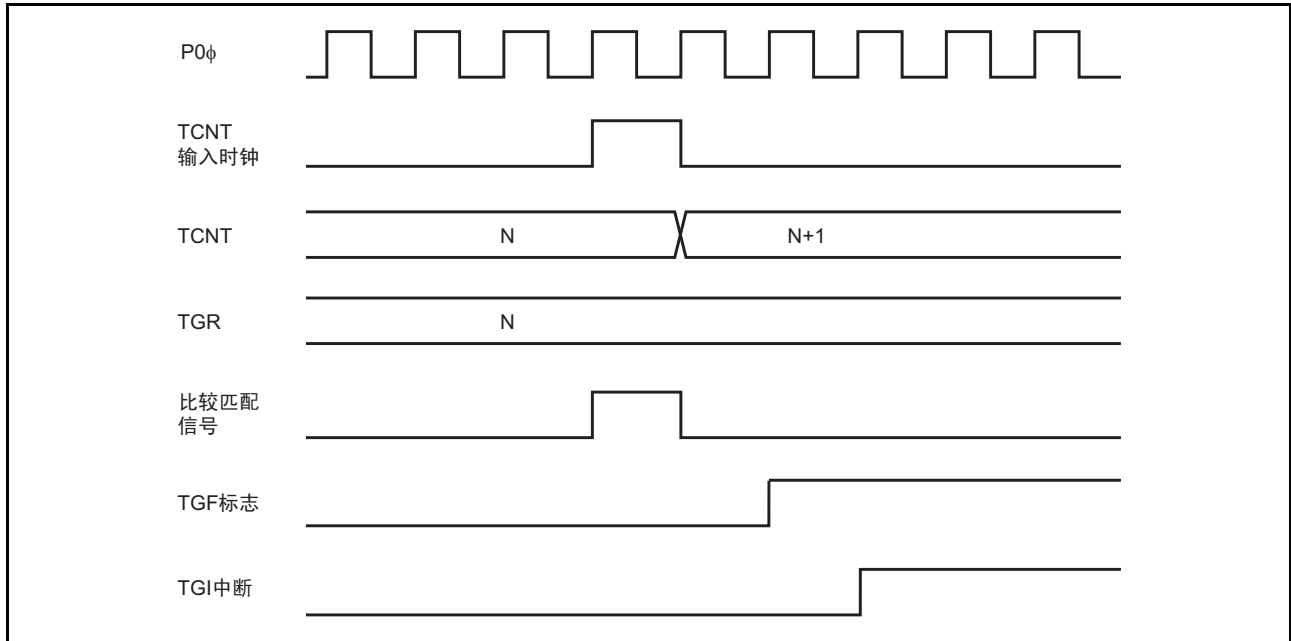


图 12.92 TGI 中断时序（比较匹配）

(2) 输入捕捉时的 TGF 标志的置位时序

由输入捕捉产生的 TSR 的 TGF 标志置位时序和 TGI 中断请求信号的时序如图 12.93 所示。

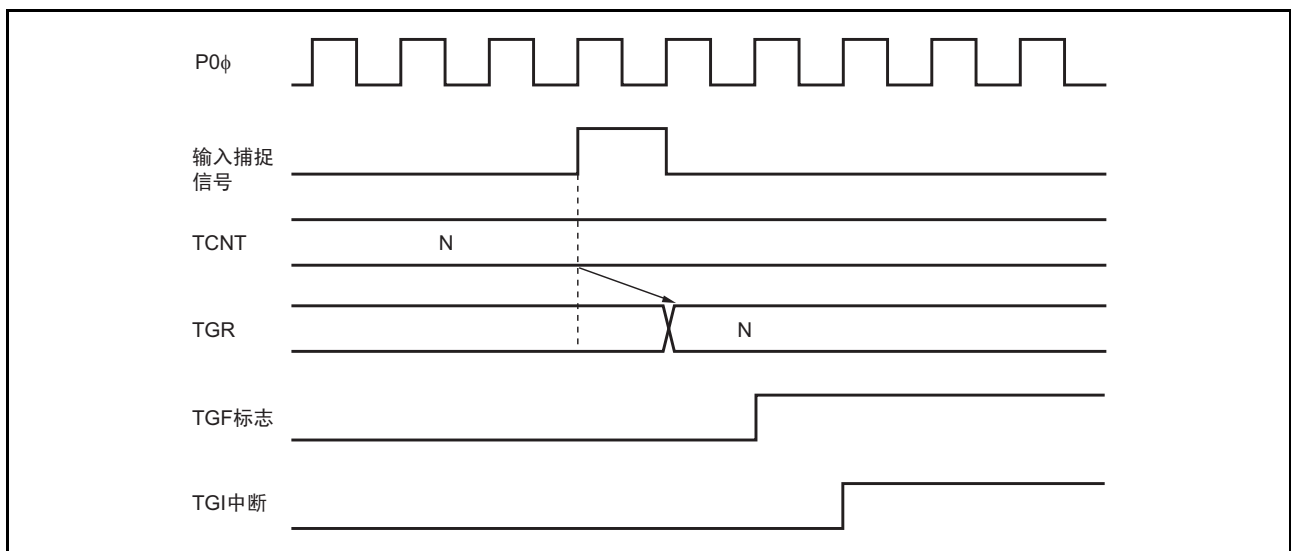


图 12.93 TGI 中断时序（输入捕捉）

(3) TCFV 标志 /TCFU 标志的置位时序

由上溢产生的 TSR 的 TCFV 标志置位时序和 TCIV 中断请求信号的时序如图 12.94 所示，由下溢产生的 TSR 的 TCFU 标志置位时序和 TCIU 中断请求信号的时序如图 12.95 所示。

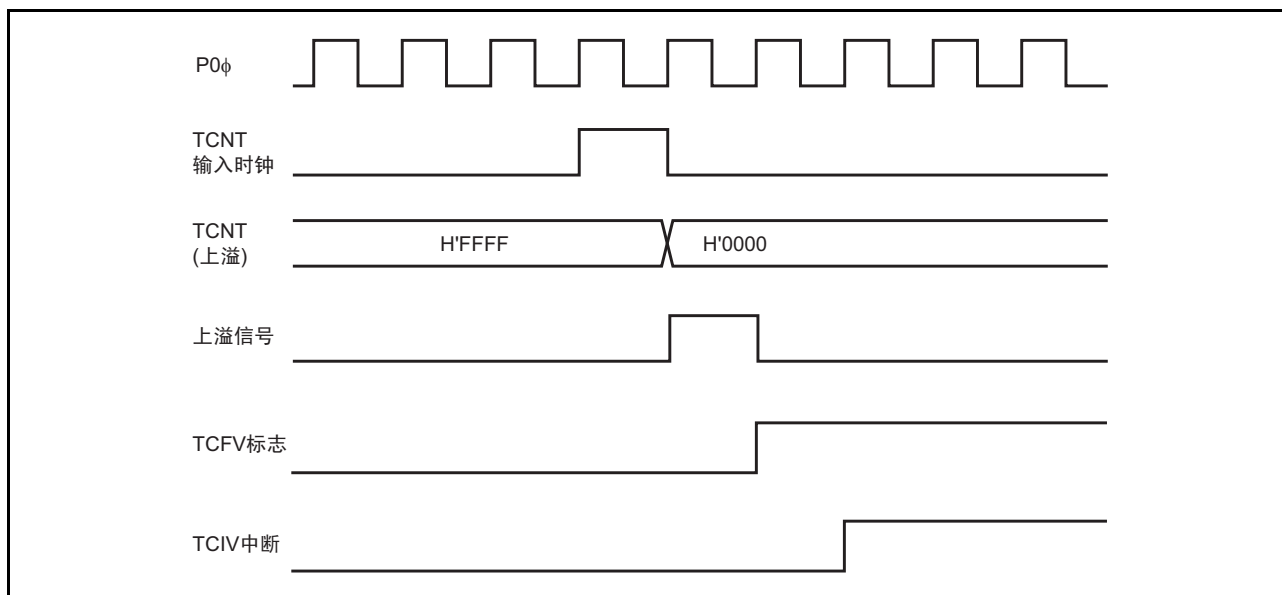


图 12.94 TCIV 中断的置位时序

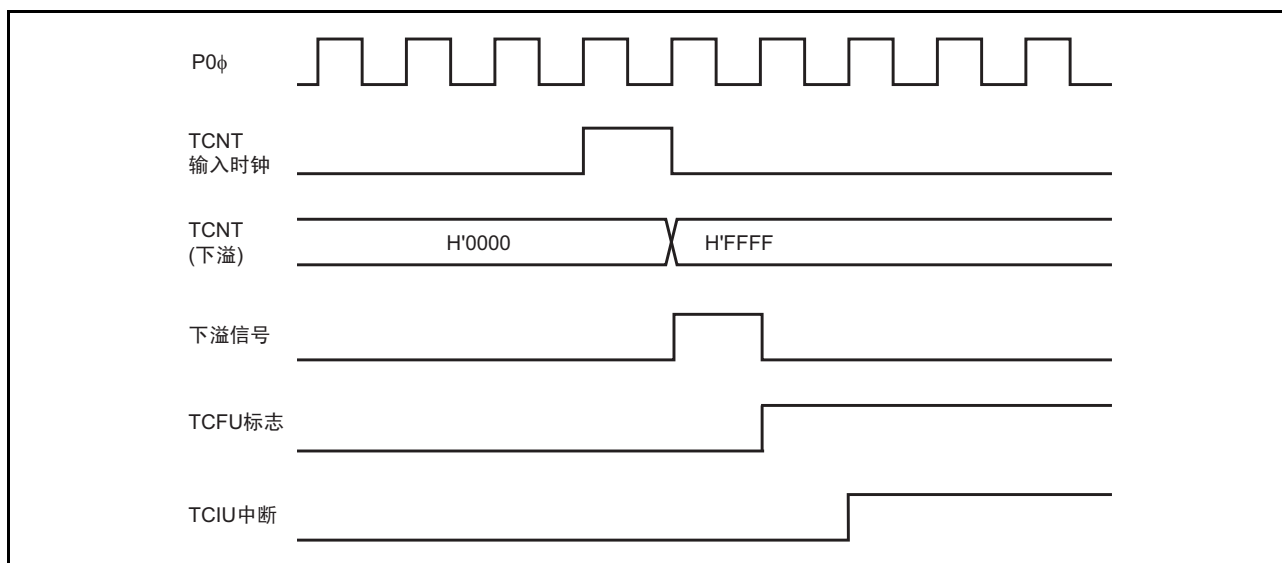


图 12.95 TCIU 中断的置位时序

(4) 状态标志的清除时序

如果在 CPU 读 “1” 的状态后写 “0”，就清除状态标志。在启动直接存储器存取控制器时，能自动清除此状态标志。通过 CPU 和直接存储器存取控制器清除状态标志的时序分别如图 12.96 和图 12.97 所示。

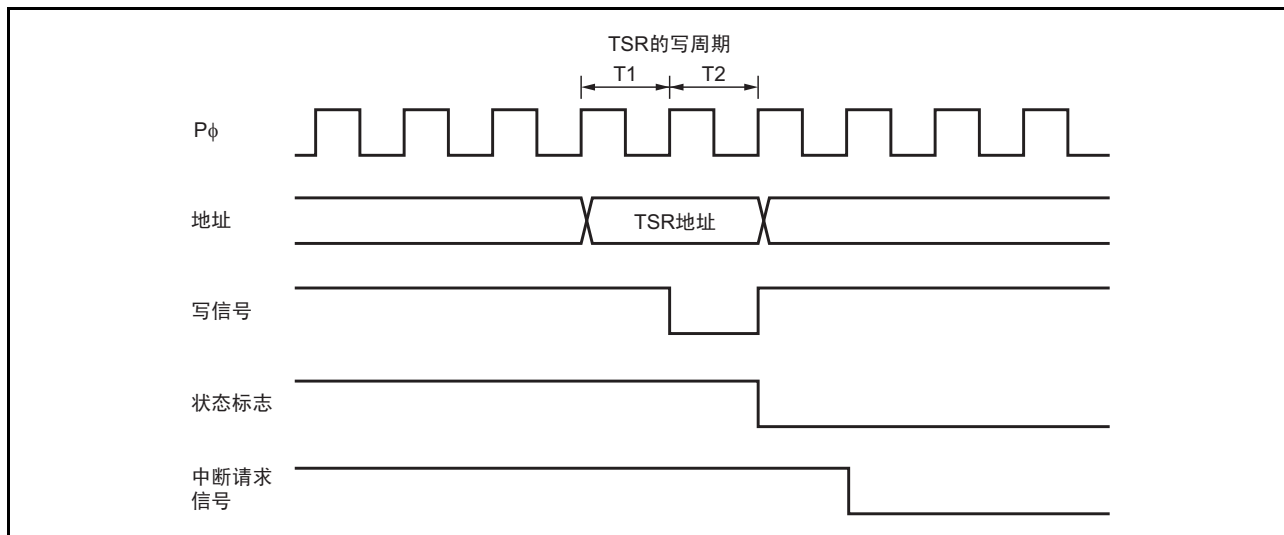


图 12.96 通过 CPU 清除状态标志的时序

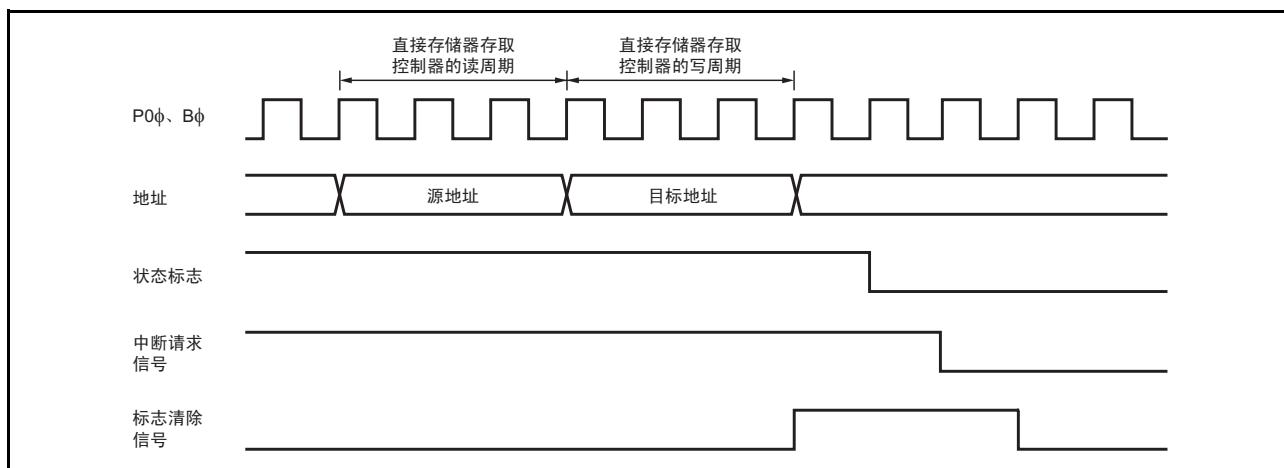


图 12.97 通过启动直接存储器存取控制器来清除状态标志的时序

12.7 使用时的注意事项

12.7.1 模块待机模式的设定

此模块能通过待机控制寄存器设定禁止或者允许此模块的运行，初始值为停止此模块的运行。能通过解除模块待机模式，使寄存器变为可存取的状态，详细内容请参照“49. 低功耗模式”。

12.7.2 输入时钟的限制事项

必须注意：在单边沿的情况下，输入时钟的脉宽至少为 1.5 个状态；在双边沿的情况下，输入时钟的脉宽至少为 2.5 个状态。否则，就不能正常运行。

在相位计数模式中，2 个输入时钟的相位差和重叠都至少为 1.5 个状态，脉宽至少为 2.5 个状态。相位计数模式的输入时钟条件如图 12.98 所示。

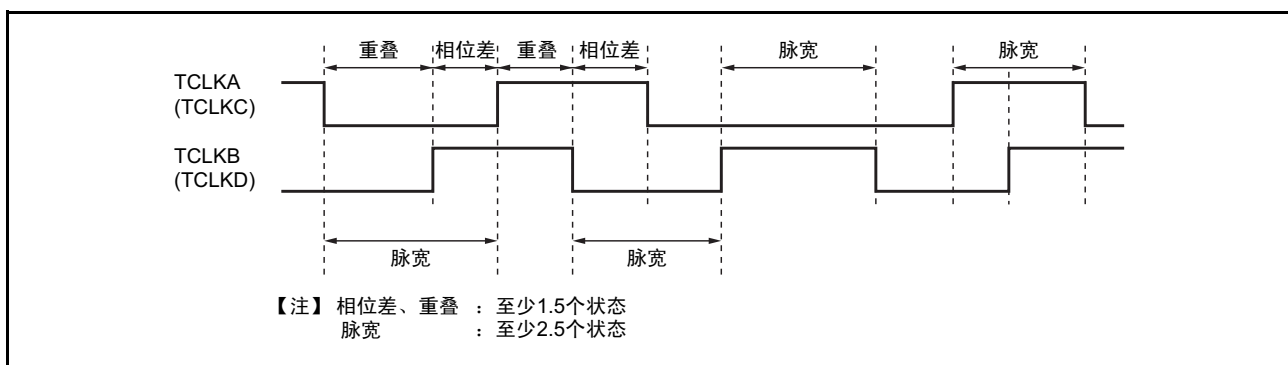


图 12.98 相位计数模式的相位差、重叠和脉宽

12.7.3 设定周期时的注意事项

如果设定为通过比较匹配清除计数器，就在 TCNT 和 TGR 的值相同的最后状态（在更新 TCNT 相同后的计数值时）清除 TCNT。因此，实际的计数器频率用以下计算式表示：

$$f = \frac{P0\phi}{(N+1)}$$

f：计数器频率

P0φ：外围时钟运行频率

N：TGR 的设定值

12.7.4 TCNT 的写和清除的竞争

如果在 TCNT 写周期中的 T2 状态产生计数器清除信号，就不写 TCNT 而优先清除 TCNT。此时序如图 12.99 所示。

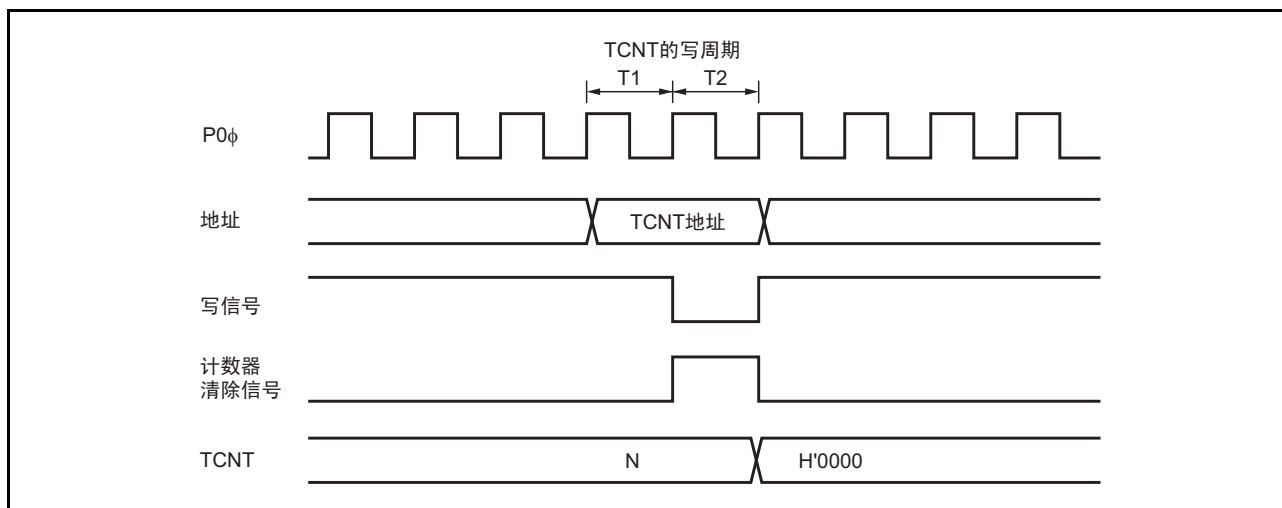


图 12.99 TCNT 的写和清除的竞争

12.7.5 TCNT 的写和递增计数的竞争

即使在 TCNT 写周期中的 T2 状态产生递增计数信号，也不进行递增计数而优先写 TCNT。此时序如图 12.100 所示。

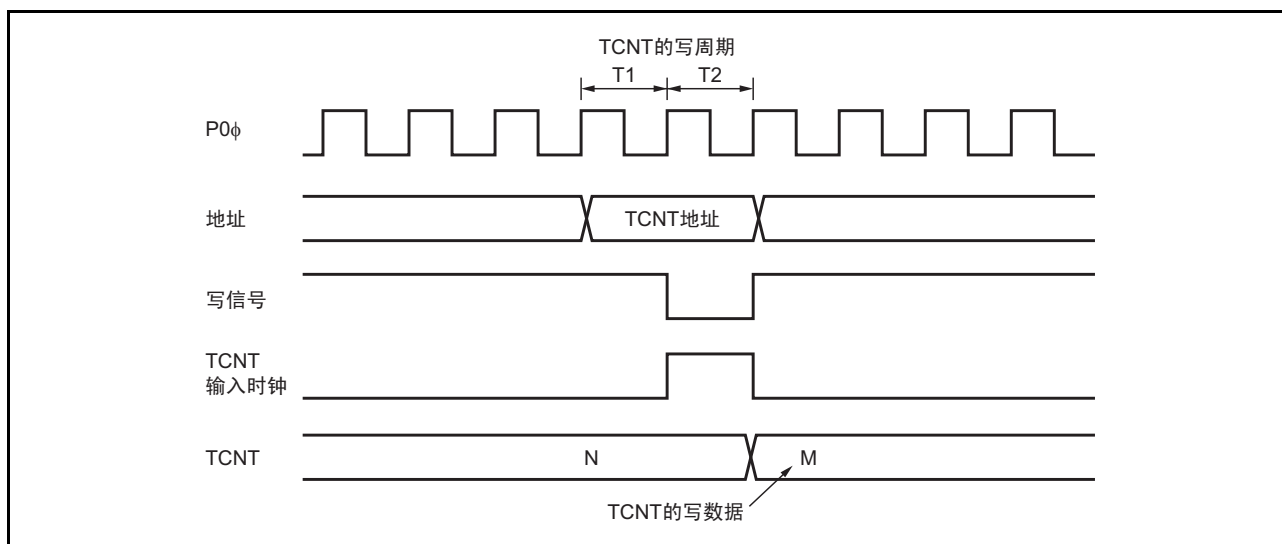


图 12.100 TCNT 的写和递增计数的竞争

12.7.6 TGR 的写和比较匹配的竞争

如果在 TGR 写周期中的 T2 状态产生比较匹配，就写 TGR 并且产生比较匹配信号。此时序如图 12.101 所示。

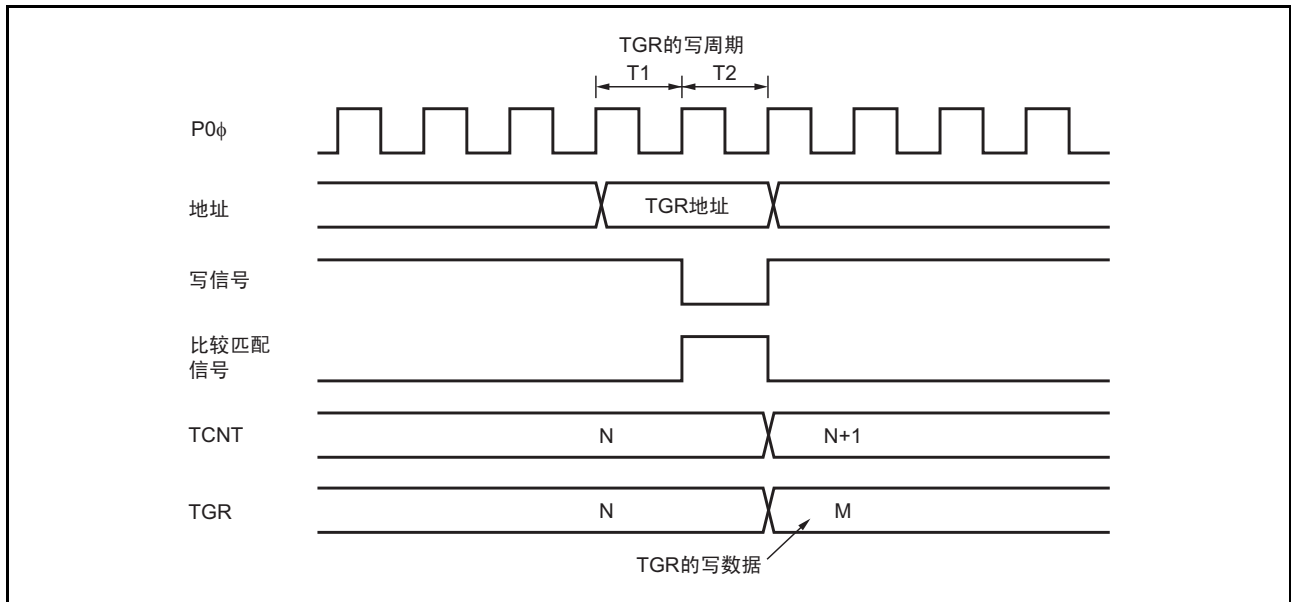


图 12.101 TGR 的写和比较匹配的竞争

12.7.7 缓冲寄存器的写和比较匹配的竞争

如果在 TGR 写周期中的 T2 状态产生比较匹配，通过缓冲运行传送到 TGR 的数据就为写之前的数据。此时序如图 12.102 所示。

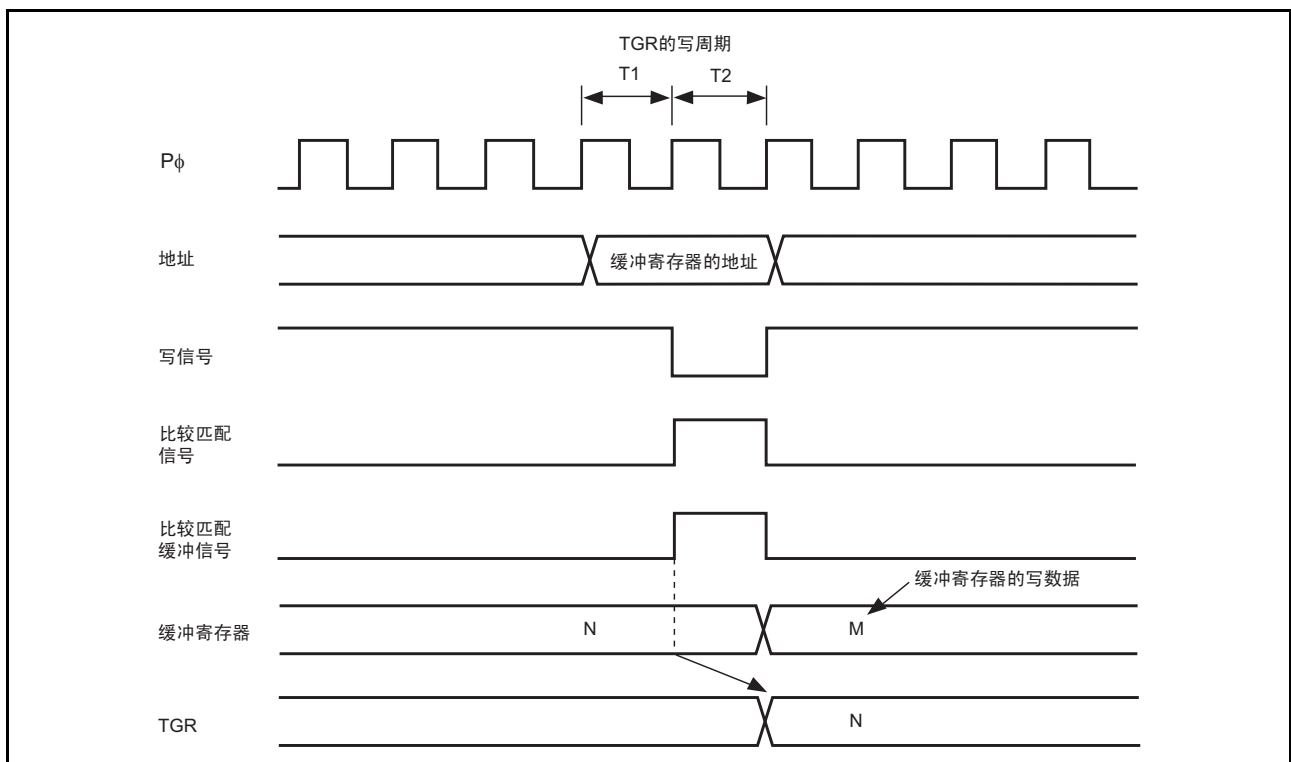


图 12.102 缓冲寄存器的写和比较匹配的竞争

12.7.8 缓冲寄存器的写和 TCNT 清除的竞争

当通过缓冲传送模式寄存器 (TBTM) 将缓冲传送时序设定为清除 TCNT 时, 如果在 TGR 写周期中的 T2 状态产生 TCNT 的清除信号, 通过缓冲运行传送的数据就为写之前的数据。

此时序如图 12.103 所示。

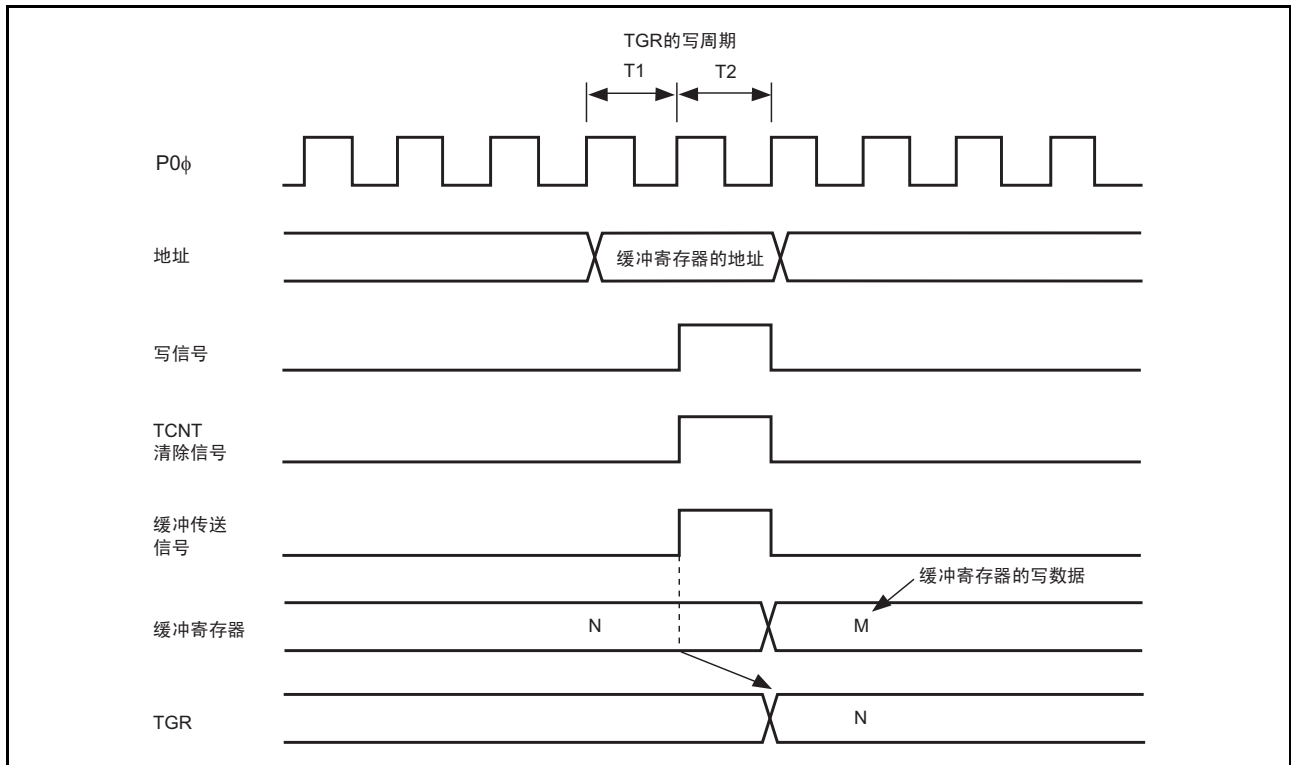


图 12.103 缓冲寄存器的写和 TCNT 清除的竞争

12.7.9 TGR 的读和输入捕捉的竞争

如果在 TGR 读周期中的 T1 状态产生输入捕捉信号, 读到的数据就为输入捕捉传送前的数据。

此时序如图 12.104 所示。

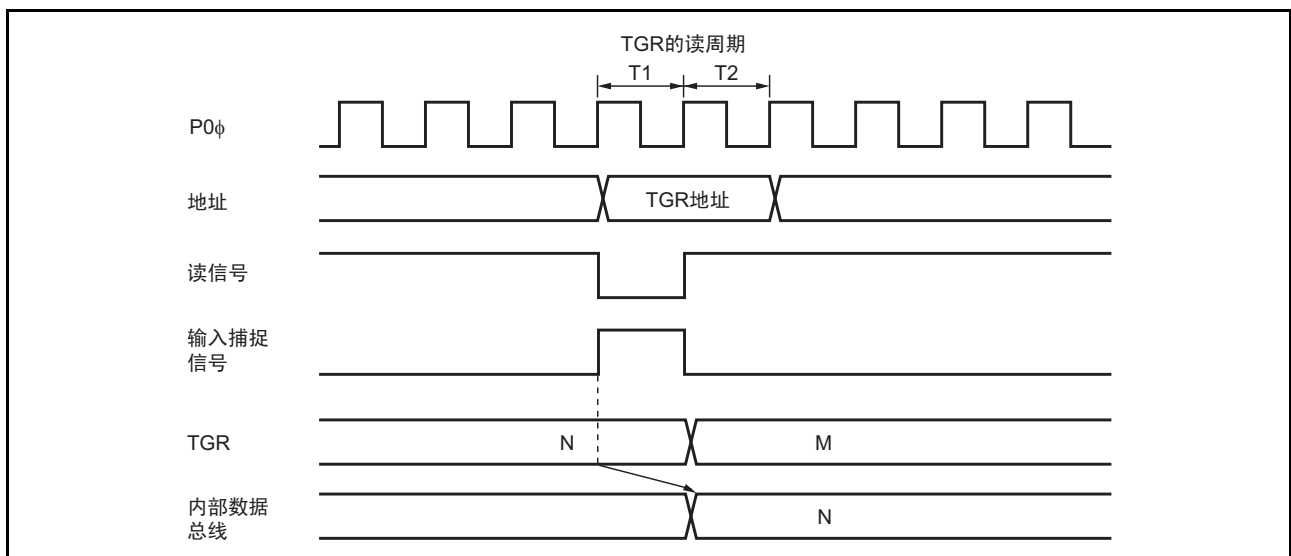


图 12.104 TGR 的读和输入捕捉的竞争

12.7.10 TGR 的写和输入捕捉的竞争

如果在 TGR 写周期中的 T2 状态产生输入捕捉信号，就不写 TGR 而优先输入捕捉。
此时序如图 12.105 所示。

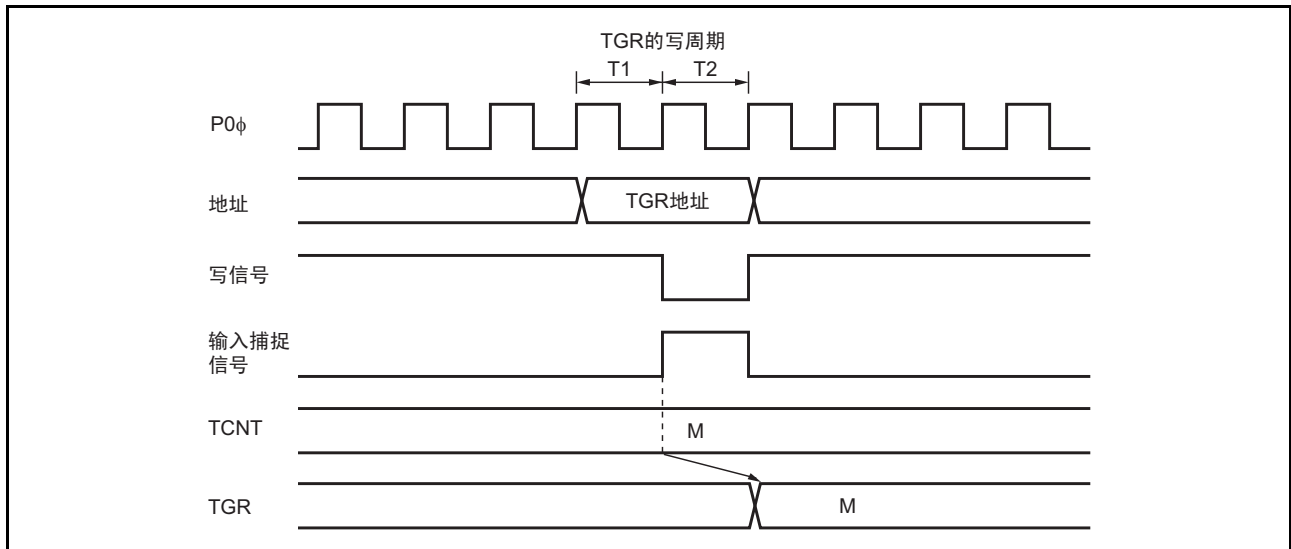


图 12.105 TGR 的写和输入捕捉的竞争

12.7.11 缓冲寄存器的写和输入捕捉的竞争

如果在缓冲器写周期中的 T2 状态产生输入捕捉信号，就不写缓冲寄存器而优先缓冲运行。
此时序如图 12.106 所示。

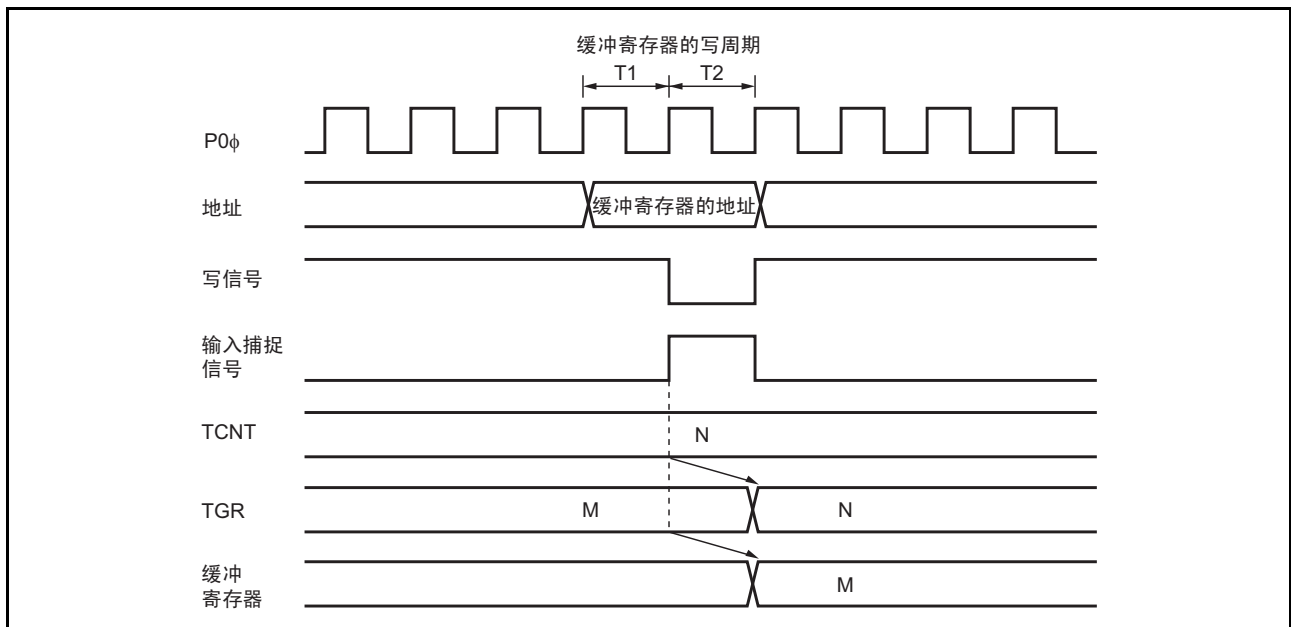


图 12.106 缓冲寄存器的写和输入捕捉的竞争

12.7.12 级联中的 TCNT_2 的写和上溢 / 下溢的竞争

如果级联定时器的计数器 (TCNT_1 和 TCNT_2) 并且在 TCNT_1 进行计数的瞬间 (TCNT_2 发生上溢 / 下溢的瞬间) 和 TCNT_2 写周期中的 T2 状态发生竞争, 就写 TCNT_2 而禁止 TCNT_1 的计数信号。此时, TGRA_1 用作比较匹配寄存器, 如果 TGRA_1 和 TCNT_1 的值相同时, 就产生比较匹配信号。

如果选择 TCNT_1 计数时钟为通道 0 的输入捕捉源, TGRA_0 ~ D_0 就进行输入捕捉运行。如果选择 TGRC_0 的比较匹配 / 输入捕捉为 TGRB_1 的输入捕捉源, TGRB_1 就进行输入捕捉运行。

此时序如图 12.107 所示。

如果在级联运行时设定清除 TCNT, 就必须进行通道 1 和通道 2 的同步设定。

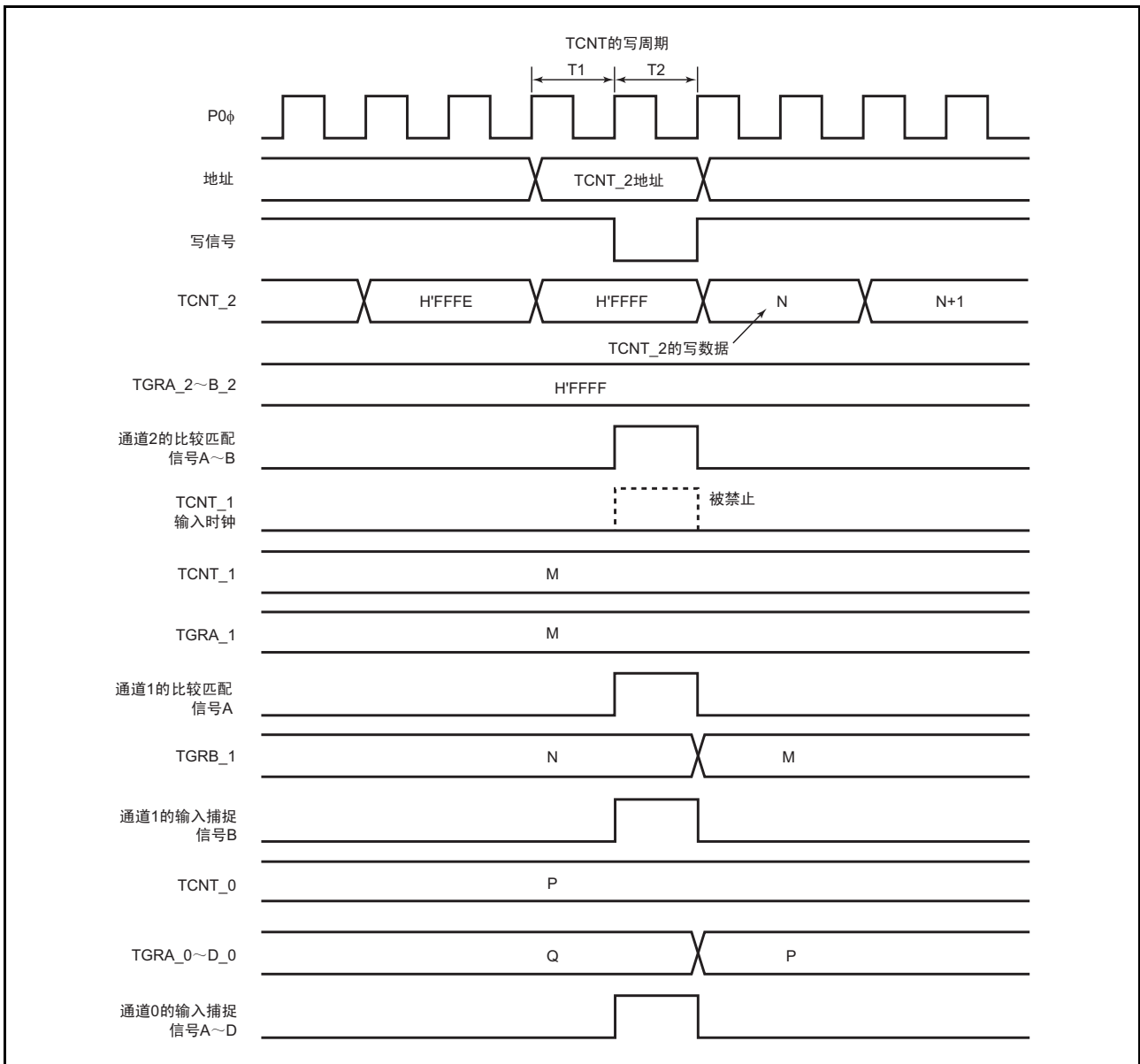


图 12.107 级联中的 TCNT_2 的写和上溢 / 下溢的竞争

12.7.13 互补 PWM 模式停止时的计数器值

如果 TCNT_3 和 TCNT_4 在互补 PWM 模式中运行时停止计数，TCNT_3 就变为定时器的空载时间寄存器 (TDDR) 的值，TCNT_4 变为“H'0000”。

一旦重新开始互补 PWM 模式运行，计数器就自动从初始状态开始计数。

此说明如图 12.108 所示。

如果要在其他运行模式中开始计数，就必须给 TCNT_3 和 TCNT_4 设定计数初始值。

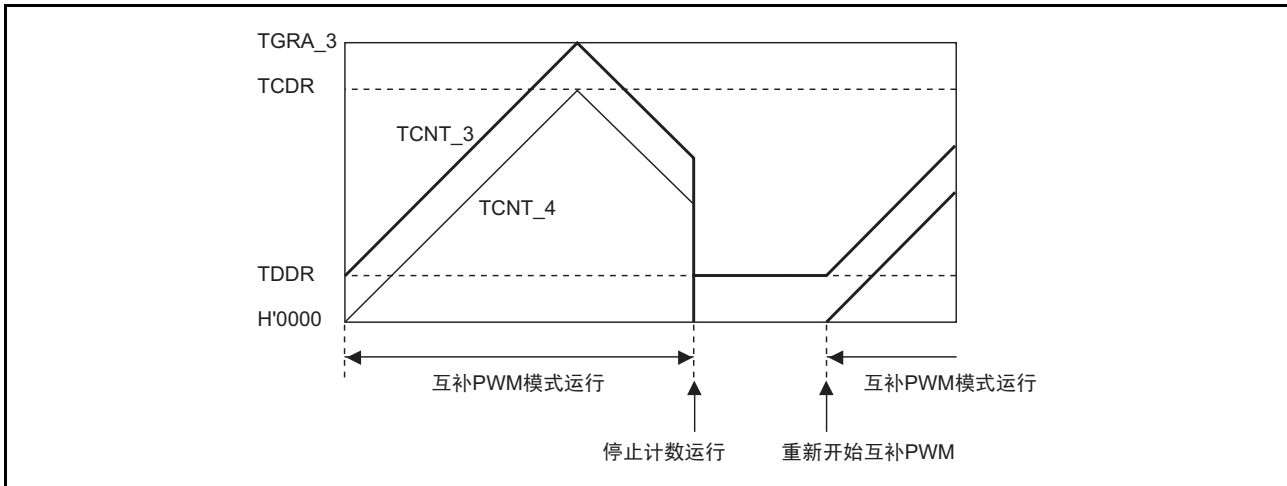


图 12.108 互补 PWM 模式停止时的计数器值

12.7.14 互补 PWM 模式中的缓冲运行的设定

在互补 PWM 模式中，必须通过缓冲运行来改写 PWM 周期设定寄存器 (TGRA_3)、定时器的周期数据寄存器 (TCDR) 和占空比设定寄存器 (TGRB_3、TGRA_4 和 TGRB_4)。

按照 TMDR_3 的 BFA 位和 BFB 位的设定，进行互补 PWM 模式的通道 3 和通道 4 的缓冲运行。如果将 TMDR_3 的 BFA 位置“1”，就在 TGRC_3 用作 TGRA_3 的缓冲寄存器同时，TGRC_4 用作 TGRA_4 的缓冲寄存器，并且 TCBR 用作 TCDR 的缓冲寄存器。

12.7.15 复位同步 PWM 模式中的缓冲运行和比较匹配标志

要设定为在复位同步 PWM 模式中进行缓冲运行时，必须将 TMDR_4 的 BFA 位和 BFB 位置“0”。如果将 TMDR_4 的 BFA 位置“1”，TIOC4C 引脚就不能输出波形。

按照 TMDR_3 的 BFA 位和 BFB 位的设定，进行复位同步 PWM 模式的通道 3 和通道 4 的缓冲运行。例如，如果将 TMDR_3 的 BFA 位置“1”，就在 TGRC_3 用作 TGRA_3 的缓冲寄存器的同时，TGRC_4 用作 TGRA_4 的缓冲寄存器。

当 TGRC_3 和 TGRD_3 用作缓冲寄存器时，不能将 TSR_3 和 TSR_4 的 TGFC 位和 TGFD 位置位。

将 TMDR_3 的 BFA 位和 BFB 位置“1”并且将 TMDR_4 的 BFA 位和 BFB 位置“0”时的 TGR_3、TGR_4、TIOC3 和 TIOC4 的运行例子如图 12.109 所示。

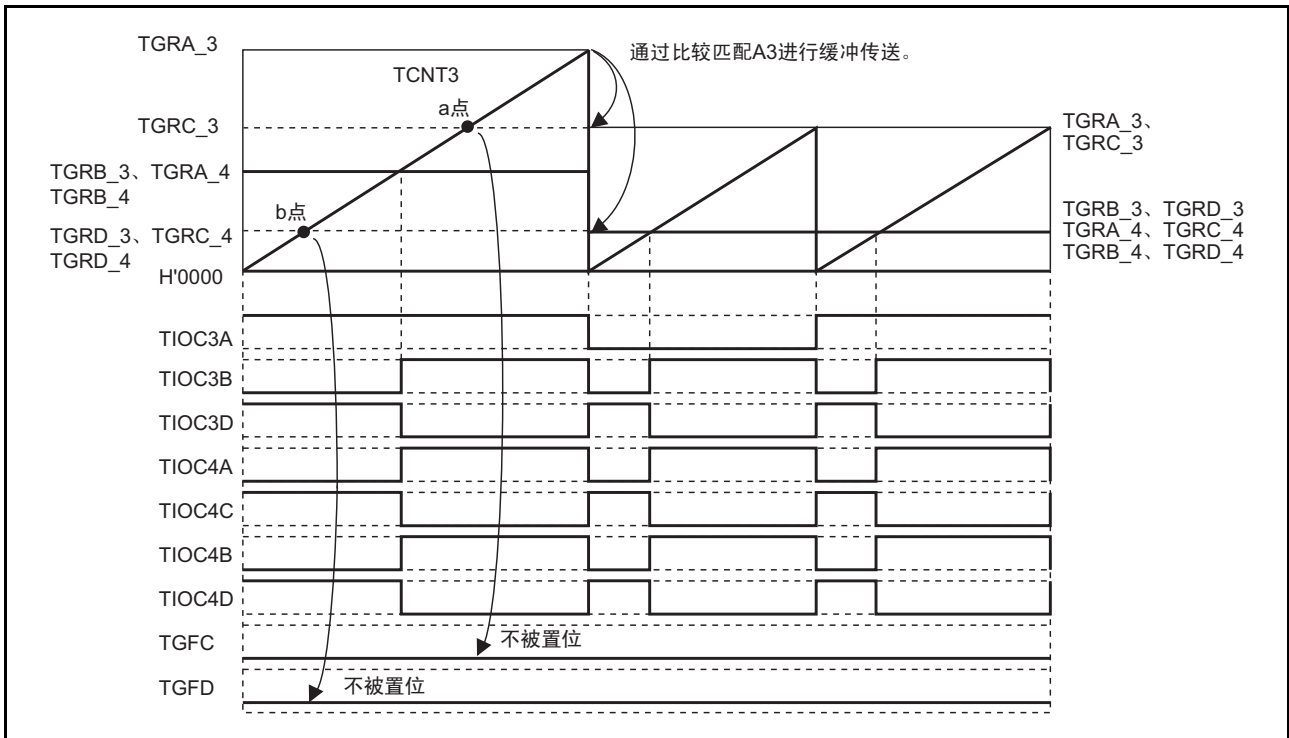


图 12.109 复位同步 PWM 模式的缓冲运行和比较匹配标志

12.7.16 复位同步 PWM 模式的上溢标志

如果设定复位同步 PWM 模式并且将 TSTR 的 CST3 位置“1”，就开始 TCNT_3 和 TCNT_4 的计数运行。此时，TCNT_4 的计数时钟源和计数边取决于 TCR_3 的设定。

在复位同步 PWM 模式中，当周期寄存器 TGRA_3 的设定值为“H'FFFF”并且指定 TGRA_3 的比较匹配为计数器清除源时，如果 TCNT_3 和 TCNT_4 递增计数到“H'FFFF”，就产生和 TGRA_3 的比较匹配，并且同时清除 TCNT_3 和 TCNT_4。此时，不将 TSR 的上溢标志 TCFV 位置位。

在复位同步 PWM 模式中，当周期寄存器 TGRA_3 的设定值为“H'FFFF”并且指定 TGRA_3 的比较匹配为计数器清除源而不进行同步设定时的 TCFV 位的运行例子如图 12.110 所示。

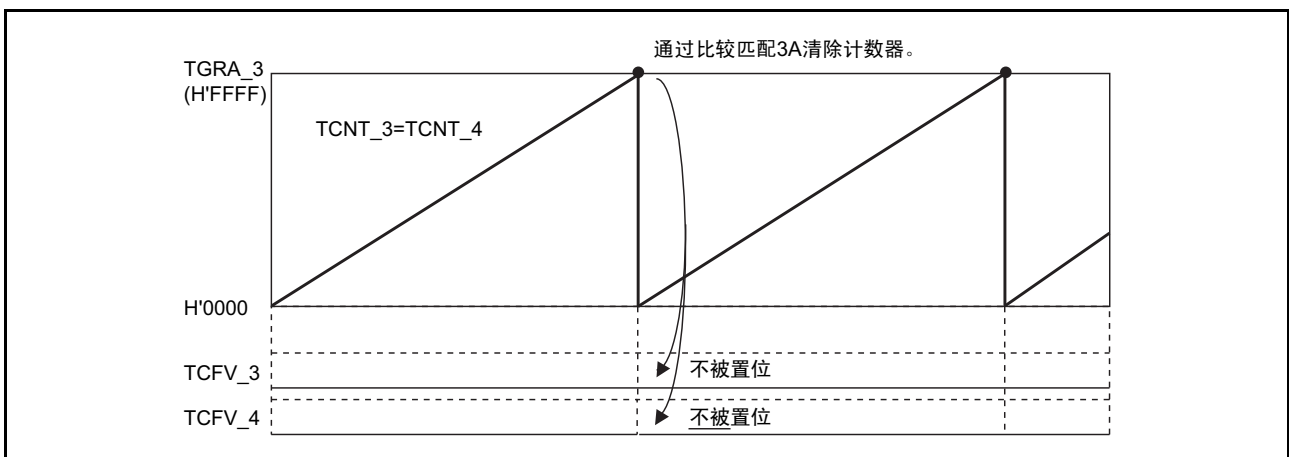


图 12.110 复位同步 PWM 模式的上溢标志

12.7.17 上溢 / 下溢和计数器清除的竞争

如果同时发生上溢 / 下溢和计数器清除，就不将 TSR 的 TCFV/TCFU 标志置位而优先清除 TCNT。将 TGR 的比较匹配作为清除源并且给 TGR 设定 “H'FFFF” 时的运行时序如图 12.111 所示。

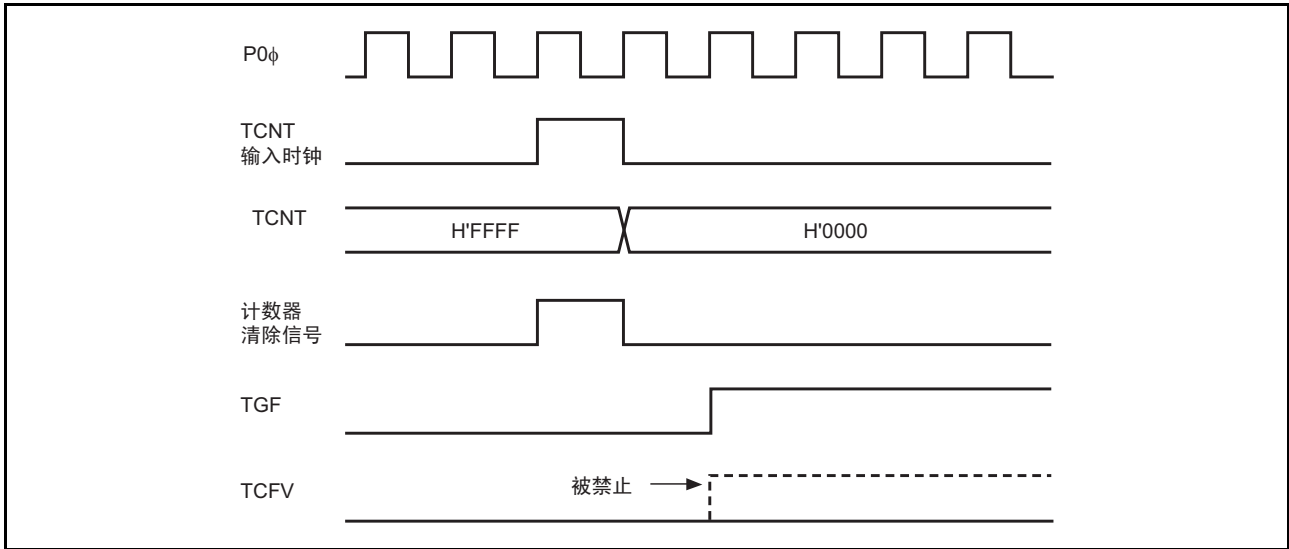


图 12.111 上溢和计数器清除的竞争

12.7.18 TCNT 的写和上溢 / 下溢的竞争

即使在 TCNT 写周期中的 T2 状态发生递增计数 / 递减计数和上溢 / 下溢，也优先写 TCNT 而不将 TSR 的 TCFV/TCFU 标志置位。

TCNT 的写和上溢发生竞争时的运行时序如图 12.112 所示。

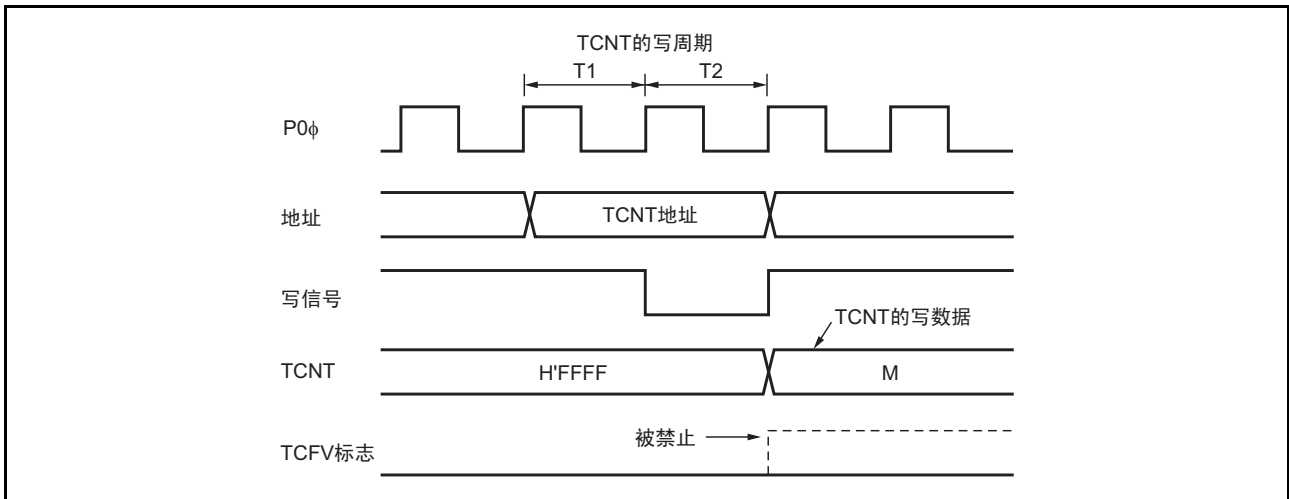


图 12.112 TCNT 的写和上溢的竞争

12.7.19 从正常运行或者 PWM 模式 1 转移到复位同步 PWM 模式时的注意事项

要从通道 3 和通道 4 的正常运行或者 PWM 模式 1 转移到复位同步 PWM 模式时，如果在将输出引脚（TIOC3B、TIOC3D、TIOC4A、TIOC4C、TIOC4B、TIOC4D）置为高电平的状态下停止计数器并且在转移到复位同步 PWM 模式后进行运行，就不能正确地进行引脚的初始输出，因此必须注意。

要从正常运行转移到复位同步 PWM 模式时，必须在给 TIORH_3、TIORL_3、TIORH_4、TIORL_4 寄存器写“H'11”并且将输出引脚初始化为低电平后设定寄存器的初始值“H'00”，然后进行模式转移。

要从 PWM 模式 1 转移到复位同步 PWM 模式时，必须暂且转移到正常运行，在将输出引脚初始化为低电平后设定寄存器的初始值“H'00”，然后转移到复位同步 PWM 模式。

12.7.20 互补 PWM 模式和复位同步 PWM 模式的输出电平

在通道 3 和通道 4 为互补 PWM 模式或者复位同步 PWM 模式时，通过定时器的输出控制寄存器（TOCR）的 OLSF 位和 OLSN 位设定 PWM 波形的输出电平。在互补 PWM 模式或者复位同步 PWM 模式中，必须将 TIOR 置“H'00”。

12.7.21 模块待机时的中断

如果在请求中断的状态下变为模块待机模式，就不能清除 CPU 的中断源或者直接存储器存取控制器的启动源。

必须预先禁止中断，然后设定为模块待机模式。

12.7.22 级联中的 TCNT_1 和 TCNT_2 的同时输入捕捉

在将定时器的计数器 1 和计数器 2（TCNT_1 和 TCNT_2）进行级联后用作 32 位计数器时，即使 TIOC1A 和 TIOC2A 或者 TIOC1B 和 TIOC2B 同时进行输入捕捉的输入，也会与内部时钟同步将输入到 TCNT_1 和 TCNT_2 的外部输入捕捉信号取到内部，此时可能因 TIOC1A 和 TIOC2A 或者 TIOC1B 和 TIOC2B 的取时序发生偏差而不能正确地捕捉级联计数器的值。

例如，TCNT_1（高 16 位的计数器）应该捕捉由 TCNT_2（低 16 位的计数器）的上溢产生的递增计数值，却会捕捉递增计数前的计数值。此时，应该将 TCNT_1 为“H'FFF1”以及 TCNT_2 为“H'0000”的值传送到 TGRA_1 和 TGRA_2 或者 TGRB_1 和 TGRB_2，却会误传送 TCNT_1 为“H'FFF0”和 TCNT_2 为“H'0000”的值。

12.7.23 在互补 PWM 模式进行同步计数器清除时的输出波形控制的注意事项

在互补 PWM 模式中，如果在将同步计数器清除时的输出波形控制置为有效（TWCR 寄存器的 WRE 位为“1”）的状态下满足条件 (1) 或者条件 (2)，就会出现以下现象：

- PWM 输出引脚的空载时间变短（或者消失）。
- 在有效电平的输出期间以外的期间，从 PWM 反相输出引脚输出有效电平。

条件 (1) 在初始输出的抑制期间^⑩并且在 PWM 输出的空载时间期间内进行同步清除（图 12.113）。

条件 (2) 在初始输出的抑制期间^⑩和^⑪并且在 $TGRB_3 \leq TDDR$ 或者 $TGRA_4 \leq TDDR$ 或者 $TGRB_4 \leq TDDR$ 的条件成立的状态下进行同步清除（图 12.114）。

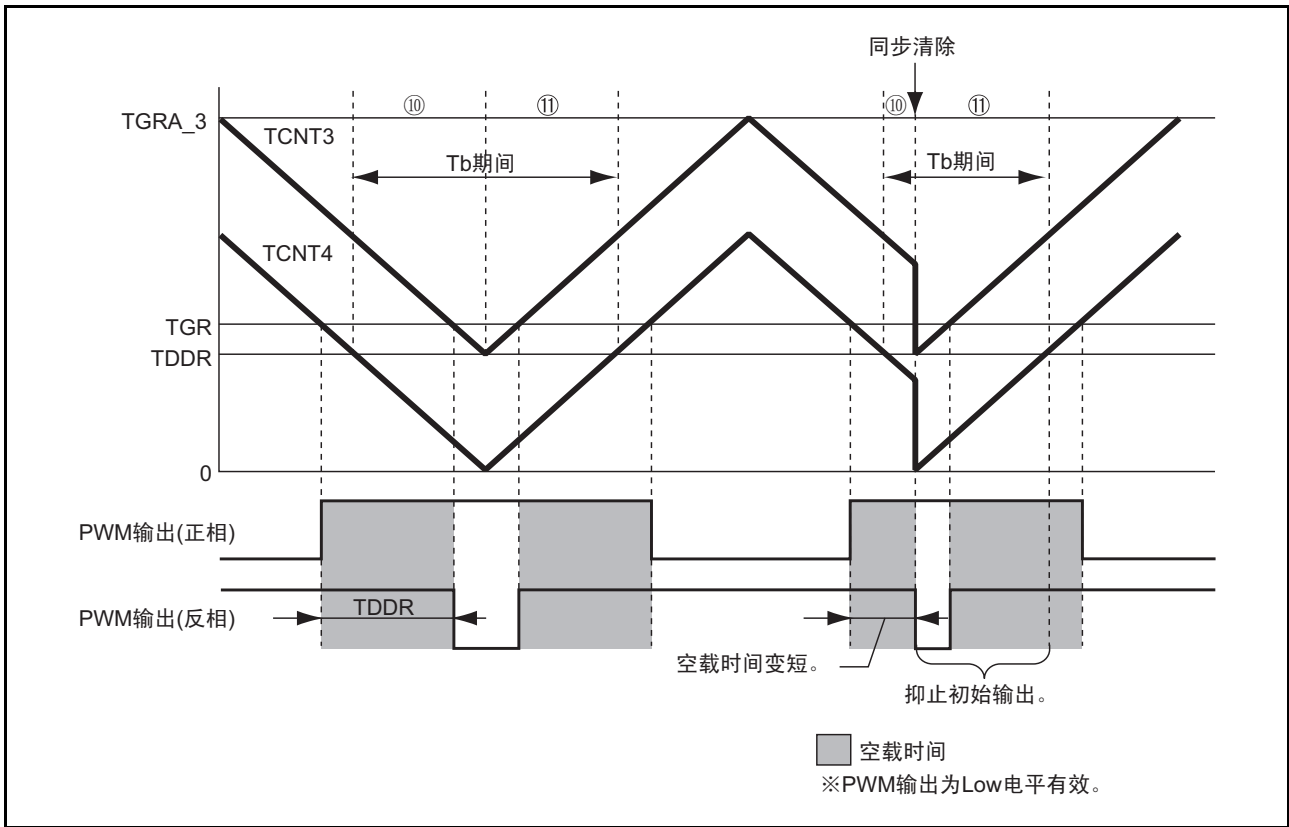


图 12.113 条件 (1) 的同步清除例子

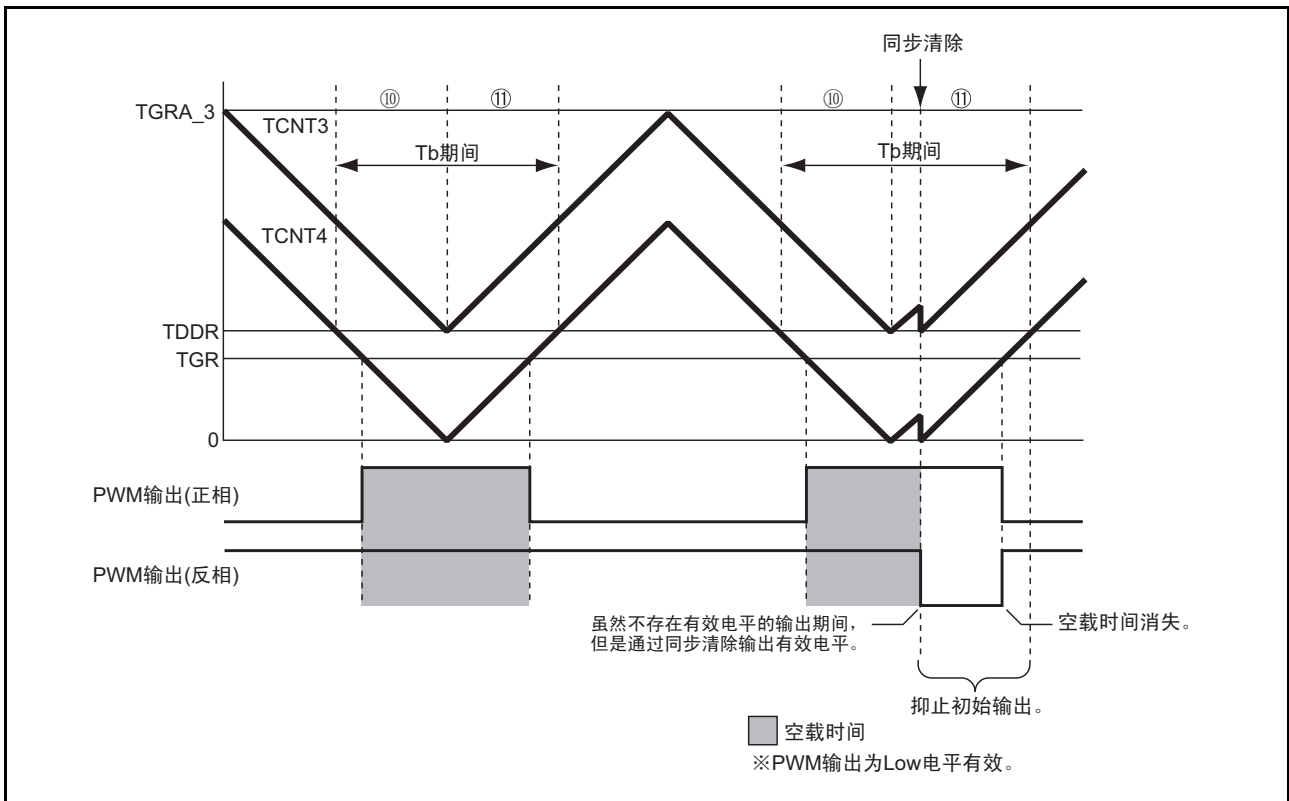


图 12.114 条件 (2) 的同步清除例子

能通过以下方法避免这种现象的发生：

在将比较寄存器 TGRB_3、TGRA_4 和 TGRB_4 全部至少设定为 2 倍的空载时间数据寄存器 TDDR 的状态下进行同步清除。

12.8 多功能定时器脉冲单元 2 的输出引脚的初始化方法

12.8.1 运行模式

此模块有以下 6 种运行模式，能在任意的模式中输出波形。

- 正常模式（通道 0～4）
- PWM 模式 1（通道 0～4）
- PWM 模式 2（通道 0～2）
- 相位计数模式 1～4（通道 1 和通道 2）
- 互补 PWM 模式（通道 3 和通道 4）
- 复位同步 PWM 模式（通道 3 和通道 4）

在此说明各模式中的此模块输出引脚的初始化方法。

12.8.2 开始复位时的运行

在复位或者待机模式时，将此模块的输出引脚（TIOC*）初始化为低电平。因为通过通用输入 / 输出端口选择此模块的引脚功能，所以在设定通用输入 / 输出端口时会将当时的此模块的引脚状态输出到端口。如果在复位后立即通过通用输入 / 输出端口选择此模块的输出，就将此模块输出的初始状态（低电平）输出到端口。当有效电平为低电平时，因为系统已经运行，所以必须在结束此模块的输出引脚的初始设定后设定通用输入 / 输出端口。

【注】 * 为通道编号 + 端口符号。

12.8.3 因运行过程中的异常等而重新设定时的运行

如果在此模块的运行过程中发生异常，就必须通过系统切断此模块的输出。即，通过输入 / 输出端口将引脚的输出转换为端口输出并且反相输出有效电平。因运行过程中的异常等而重新设定时的引脚初始化步骤以及重新设定后在其他运行模式中重新开始时的步骤如下所示。

因为此模块有 6 种运行模式，所以有 36 种模式转移的组合，但是有些转移在通道和模式的组合中不存在，模式转移的组合一览表如表 12.57 所示。

表中使用下述的符号：

Normal：正常模式 PWM1：PWM 模式 1 PWM2：PWM 模式 2
PCM：相位计数模式 1～4 CPWM：互补 PWM 模式 RPWM：复位同步 PWM 模式

表 12.57 模式转移的组合

	Normal	PWM1	PWM2	PCM	CPWM	RPWM
Normal	(1)	(2)	(3)	(4)	(5)	(6)
PWM1	(7)	(8)	(9)	(10)	(11)	(12)
PWM2	(13)	(14)	(15)	(16)	none	none
PCM	(17)	(18)	(19)	(20)	none	none
CPWM	(21)	(22)	none	none	(23) (24)	(25)
RPWM	(26)	(27)	none	none	(28)	(29)

12.8.4 因运行过程中的异常等而对引脚进行初始化的步骤以及模式转移的概要

- 在转移到定时器的 I/O 控制寄存器 (TIOR) 选择的引脚输出电平模式 (Normal、PWM1、PWM2、PCM) 时, 必须通过设定 TIOR 对引脚进行初始化。
- 在 PWM 模式 1 中, 因为不将波形输出到 TIOC*B (TIOC*D) 引脚, 所以即使设定 TIOR 也不对引脚进行初始化。要进行初始化时, 必须在正常模式中进行初始化, 然后转移到 PWM 模式 1。
- 在 PWM 模式 2 中, 因为不将波形输出到周期寄存器的引脚, 所以即使设定 TIOR 也不对引脚进行初始化。要进行初始化时, 必须在正常模式中进行初始化, 然后转移到 PWM 模式 2。
- 在正常模式或者 PWM 模式 2 中, 当 TGRC 和 TGRD 都用作缓冲寄存器时, 即使设定 TIOR 也不对缓冲寄存器的引脚进行初始化。要进行初始化时, 必须在解除缓冲器模式后进行初始化, 然后重新设定缓冲器模式。
- 在 PWM 模式 1 中, 当 TGRC 或者 TGRD 用作缓冲寄存器时, 即使设定 TIOR 也不对 TGRC 的引脚进行初始化。要初始化 TGRC 的引脚时, 必须在解除缓冲器模式后进行初始化, 然后重新设定缓冲器模式。
- 在转移到定时器的输出控制寄存器 (TOCR) 选择的引脚输出电平模式 (CPWM、RPWM) 时, 必须转移到正常模式并且通过 TIOR 进行初始化, 在将 TIOR 恢复为初始值后通过定时器的输出主控允许寄存器 (TOER) 暂时禁止通道 3 和通道 4 的输出, 然后按照模式的设定步骤 (TOCR 的设定、TMDR 的设定、TOER 的设定) 运行。

【注】 此项记述中的 * 为通道号。

根据表 12.57 的组合 No. 对引脚进行初始化的步骤如下所示，有效电平为低电平。

(1) 在正常模式的运行过程中发生异常并且在正常模式中重新开始时的运行

在正常模式中发生异常并且重新设定后在正常模式中重新开始时的说明如图 12.115 所示。

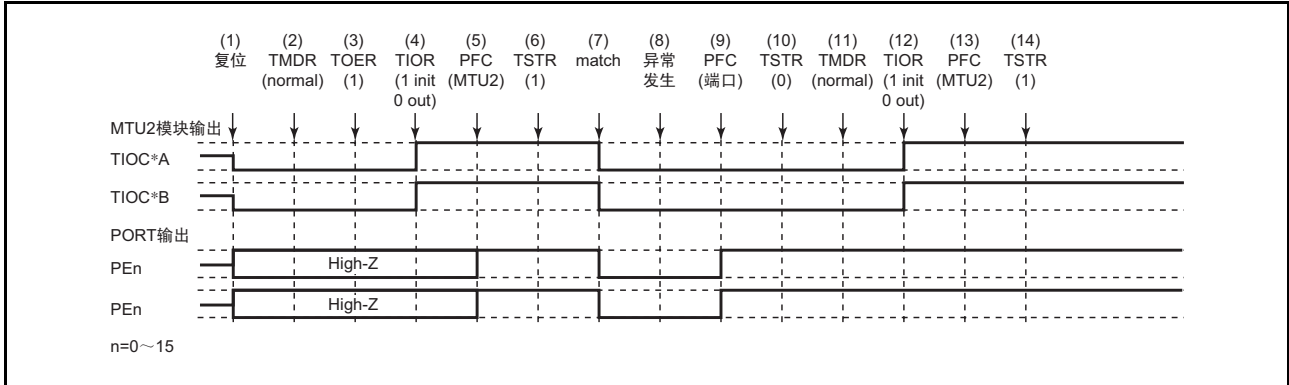


图 12.115 在正常模式中发生异常并且在正常模式中恢复的情况

- (1) 复位后，此模块的输出为低电平，端口为高阻抗状态。
- (2) 复位后，TMDR 被设定为正常模式。
- (3) 在通道3和通道4中，必须在通过TIOR对引脚进行初始化前用TOER允许输出。
- (4) 必须通过TIOR对引脚进行初始化（这是初始输出为高电平并且在比较匹配时输出低电平的例子）。
- (5) 必须通过通用输入/输出端口设定为此模块的输出。
- (6) 通过TSTR开始计数运行。
- (7) 在比较匹配时输出低电平。
- (8) 发生异常。
- (9) 必须通过通用输入/输出端口设定为端口输出并且反相输出有效电平。
- (10) 通过TSTR停止计数运行。
- (11) 在正常模式中重新开始的情况下不需要此步骤。
- (12) 必须通过TIOR对引脚进行初始化。
- (13) 必须通过通用输入/输出端口设定为此模块的输出。
- (14) 通过TSTR重新开始。

(2) 在正常模式的运行过程中发生异常并且在 PWM 模式 1 中重新开始时的运行

在正常模式中发生异常并且重新设定后在 PWM 模式 1 中重新开始时的说明如图 12.116 所示。

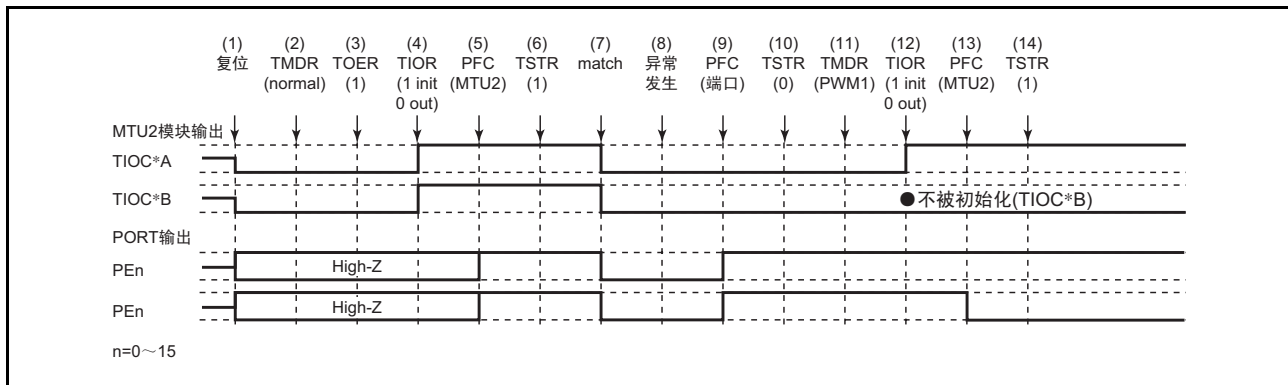


图 12.116 在正常模式中发生异常并且在 PWM 模式 1 中恢复的情况

(1) ~ (10) 和图 12.115 共用。

(11) 设定 PWM 模式 1。

(12) 必须通过 TIOR 对引脚进行初始化（在 PWM 模式 1 中不对 TIOC*B 进行初始化。如果要进行初始化，就必须在正常模式中进行初始化，然后转移到 PWM 模式 1）。

(13) 必须通过通用输入/输出端口设定为此模块的输出。

(14) 通过 TSTR 重新开始。

(3) 在正常模式的运行过程中发生异常并且在 PWM 模式 2 中重新开始时的运行

在正常模式中发生异常并且重新设定后在 PWM 模式 2 中重新开始时的说明如图 12.117 所示。

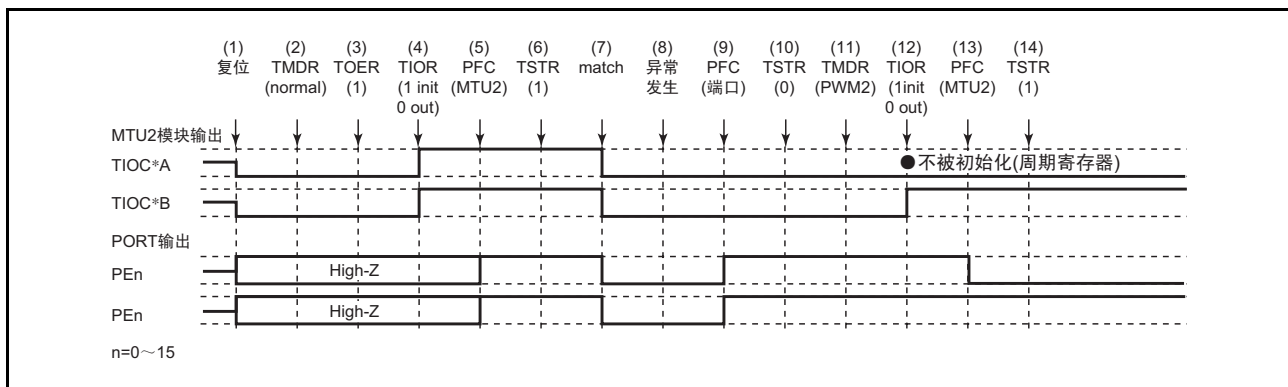


图 12.117 在正常模式中发生异常并且在 PWM 模式 2 中恢复的情况

(1) ~ (10) 和图 12.115 共用。

(11) 设定 PWM 模式 2。

(12) 必须通过 TIOR 对引脚进行初始化（在 PWM 模式 2 中不对周期寄存器的引脚进行初始化。如果要进行初始化，就必须在正常模式中进行初始化，然后转移到 PWM 模式 2）。

(13) 必须通过通用输入/输出端口设定为此模块的输出。

(14) 通过 TSTR 重新开始。

【注】 只有通道 0 ~ 2 能设定 PWM 模式 2，因此不需要设定 TOER。

(4) 在正常模式的运行过程中发生异常并且在相位计数模式中重新开始时的运行

在正常模式中发生异常并且重新设定后在相位计数模式中重新开始时的说明如图 12.118 所示。

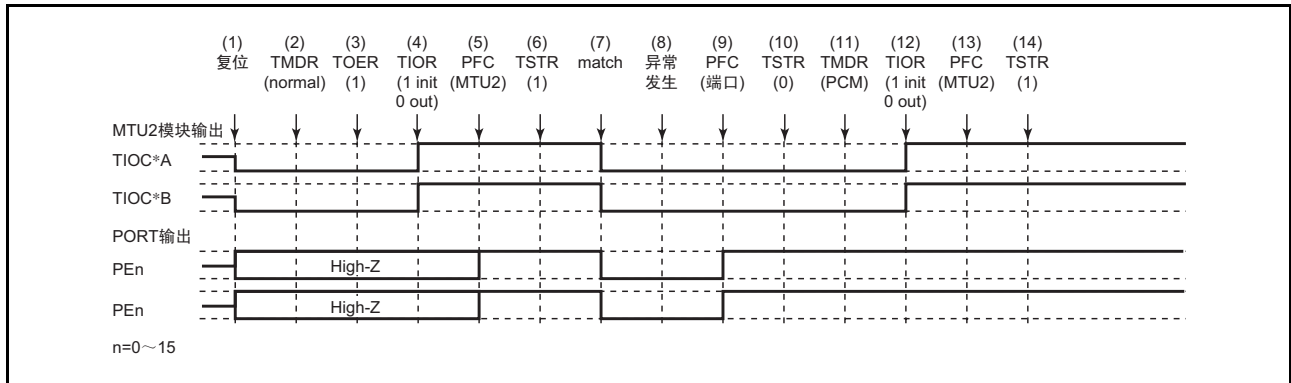


图 12.118 在正常模式中发生异常并且在相位计数模式中恢复的情况

(1) ~ (10) 和图 12.115 共用。

(11) 设定相位计数模式。

(12) 必须通过 TIOR 对引脚进行初始化。

(13) 必须通过通用输入/输出端口设定为此模块的输出。

(14) 通过 TSTR 重新开始。

【注】 只有通道 1 和通道 2 能设定相位计数模式，因此不需要设定 TOER。

(5) 在正常模式的运行过程中发生异常并且在互补 PWM 模式中重新开始时的运行

在正常模式中发生异常并且重新设定后在互补 PWM 模式中重新开始时的说明如图 12.119 所示。

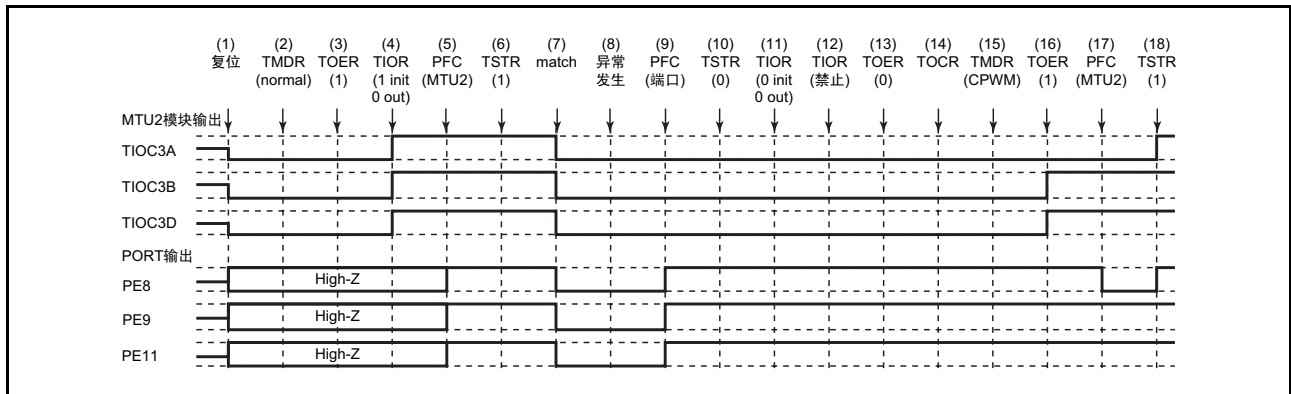


图 12.119 在正常模式中发生异常并且在互补 PWM 模式中恢复的情况

(1) ~ (10) 和图 12.115 共用。

(11) 必须通过 TIOR 对正常模式的波形生成部进行初始化。

(12) 必须通过 TIOR 禁止正常模式的波形生成部的运行。

(13) 必须通过 TOER 禁止通道 3 和通道 4 的输出。

(14) 必须通过 TOCR 选择互补 PWM 的输出电平以及允许或者禁止周期输出。

(15) 设定互补 PWM。

(16) 必须通过 TOER 允许通道 3 和通道 4 的输出。

(17) 必须通过通用输入/输出端口设定为此模块的输出。

(18) 通过 TSTR 重新开始。

(6) 在正常模式的运行过程中发生异常并且在复位同步 PWM 模式中重新开始时的运行

在正常模式中发生异常并且重新设定后在复位同步 PWM 模式中重新开始时的说明如图 12.120 所示。

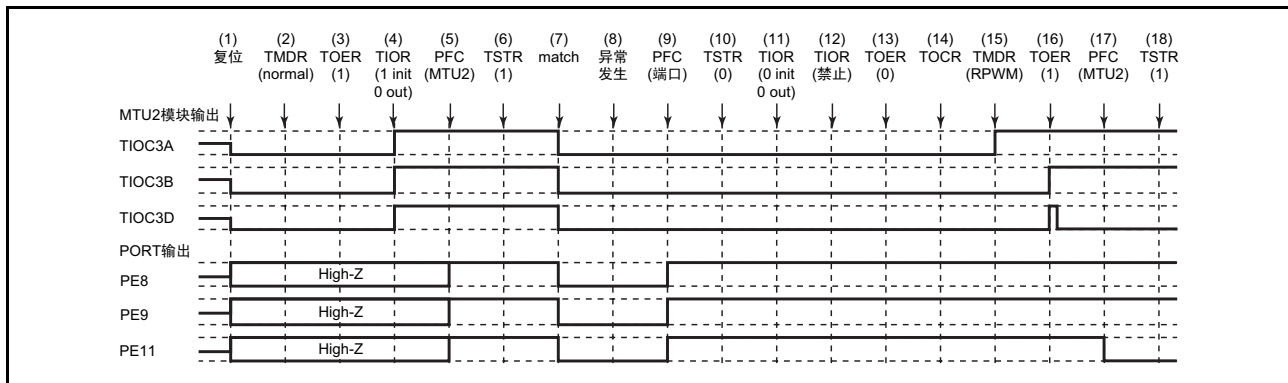


图 12.120 在正常模式中发生异常并且在复位同步 PWM 模式中恢复的情况

(1) ~ (13) 和图 12.115 共用。

(14) 必须通过 TOCR 选择复位同步 PWM 的输出电平以及允许或者禁止周期输出。

(15) 设定复位同步 PWM。

(16) 必须通过 TOER 允许通道 3 和通道 4 的输出。

(17) 必须通过通用输入/输出端口设定为此模块的输出。

(18) 通过 TSTR 重新开始。

(7) 在 PWM 模式 1 的运行过程中发生异常并且在正常模式中重新开始时的运行

在 PWM 模式 1 中发生异常并且重新设定后在正常模式中重新开始时的说明如图 12.121 所示。

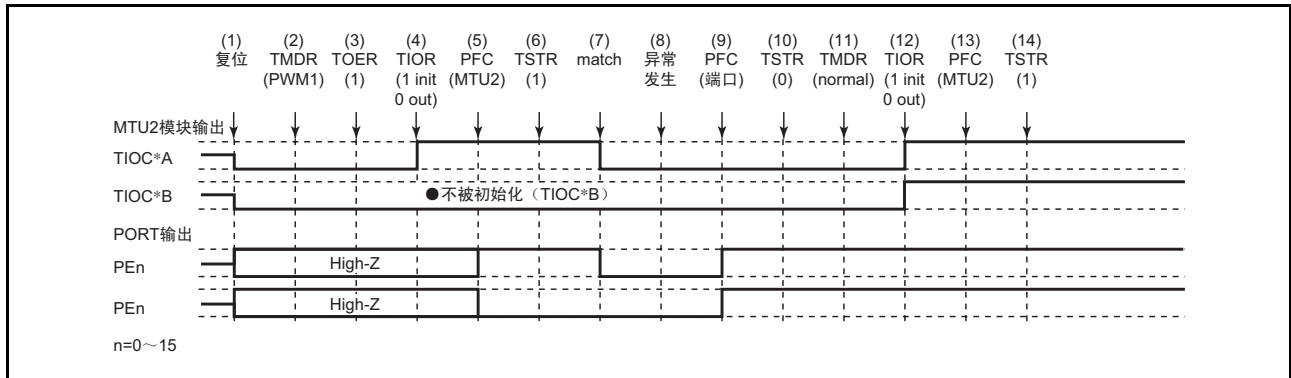


图 12.121 在 PWM 模式 1 中发生异常并且在正常模式中恢复的情况

- (1) 复位后，此模块的输出为低电平，端口为高阻抗状态。
- (2) 必须设定 PWM 模式 1。
- (3) 在通道 3 和通道 4 中，必须在通过 TIOR 对引脚进行初始化前用 TOER 允许输出。
- (4) 必须通过 TIOR 对引脚进行初始化（这是初始输出为高电平并且在比较匹配时输出低电平的例子。在 PWM 模式 1 中不对 TIOC*B 进行初始化）。
- (5) 必须通过通用输入/输出端口设定为此模块的输出。
- (6) 通过 TSTR 开始计数运行。
- (7) 在比较匹配时输出低电平。
- (8) 发生异常。
- (9) 必须通过通用输入/输出端口设定为端口输出并且反相输出有效电平。
- (10) 通过 TSTR 停止计数运行。
- (11) 必须设定正常模式。
- (12) 必须通过 TIOR 对引脚进行初始化。
- (13) 必须通过通用输入/输出端口设定为此模块的输出。
- (14) 通过 TSTR 重新开始。

(8) 在 PWM 模式 1 的运行过程中发生异常并且在 PWM 模式 1 中重新开始时的运行

在 PWM 模式 1 中发生异常并且重新设定后在 PWM 模式 1 中重新开始时的说明如图 12.122 所示。

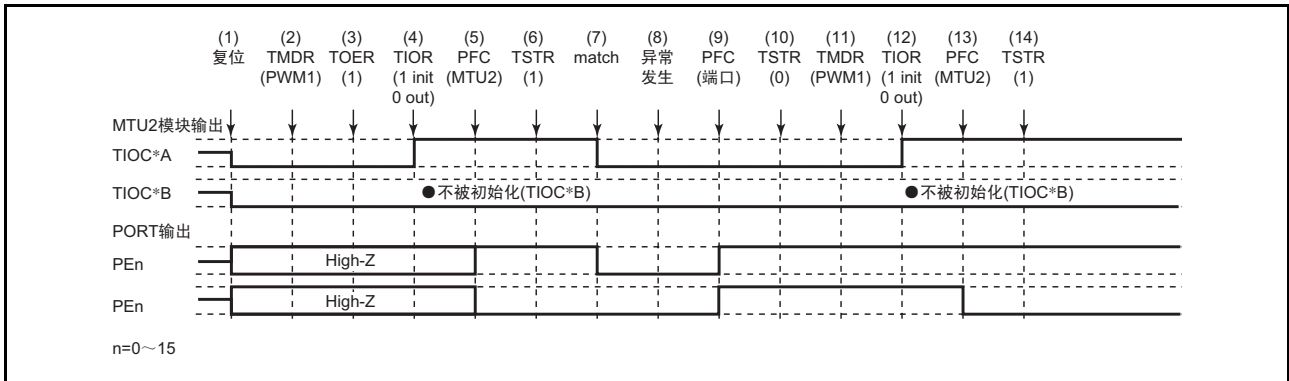


图 12.122 在 PWM 模式 1 中发生异常并且在 PWM 模式 1 中恢复的情况

(1) ~ (10) 和图 12.121 共用。

(11) 在 PWM 模式 1 中重新开始的情况下不需要此步骤。

(12) 必须通过 TIOR 对引脚进行初始化（在 PWM 模式 1 中不对 TIOC*B 进行初始化）。

(13) 必须通过通用输入/输出端口设定为此模块的输出。

(14) 通过 TSTR 重新开始。

(9) 在 PWM 模式 1 的运行过程中发生异常并且在 PWM 模式 2 中重新开始时的运行

在 PWM 模式 1 中发生异常并且重新设定后在 PWM 模式 2 中重新开始时的说明如图 12.123 所示。

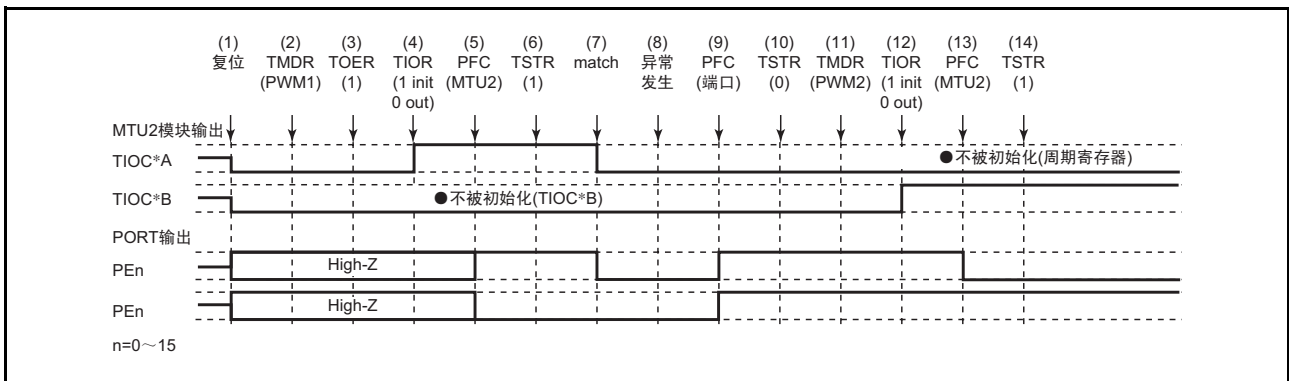


图 12.123 在 PWM 模式 1 中发生异常并且在 PWM 模式 2 中恢复的情况

(1) ~ (10) 和图 12.121 共用。

(11) 设定 PWM 模式 2。

(12) 必须通过 TIOR 对引脚进行初始化（在 PWM 模式 2 中不对周期寄存器的引脚进行初始化）。

(13) 必须通过通用输入/输出端口设定为此模块的输出。

(14) 通过 TSTR 重新开始。

【注】 只有通道 0 ~ 2 能设定 PWM 模式 2，因此不需要设定 TOER。

(10) 在 PWM 模式 1 的运行过程中发生异常并且在相位计数模式中重新开始时的运行

在 PWM 模式 1 中发生异常并且重新设定后在相位计数模式中重新开始时的说明如图 12.124 所示。

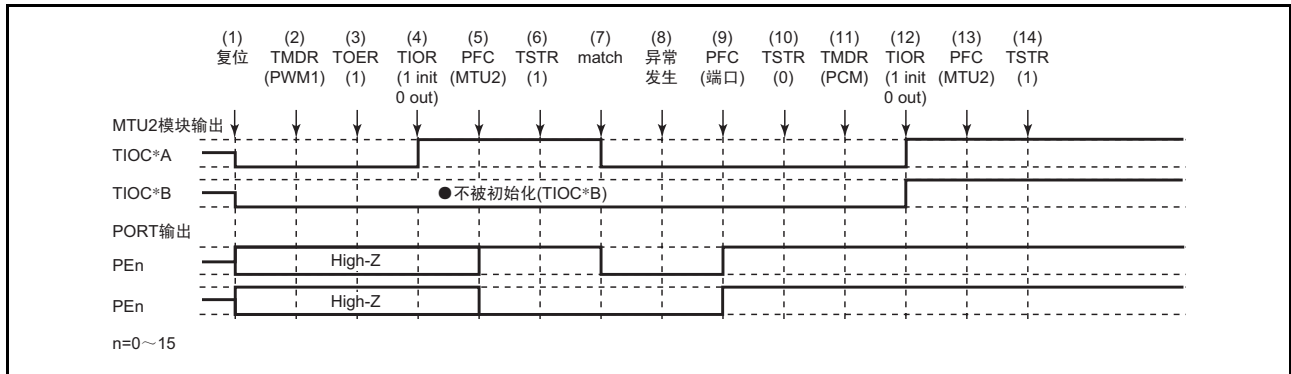


图 12.124 在 PWM 模式 1 中发生异常并且在相位计数模式中恢复的情况

(1) ~ (10) 和图 12.121 共用。

(11) 设定相位计数模式。

(12) 必须通过 TIOR 对引脚进行初始化。

(13) 必须通过通用输入/输出端口设定为此模块的输出。

(14) 通过 TSTR 重新开始。

【注】 只有通道 1 和通道 2 能设定相位计数模式，因此不需要设定 TOER。

(11) 在 PWM 模式 1 的运行过程中发生异常并且在互补 PWM 模式中重新开始时的运行

在 PWM 模式 1 中发生异常并且重新设定后在互补 PWM 模式中重新开始时的说明如图 12.125 所示。

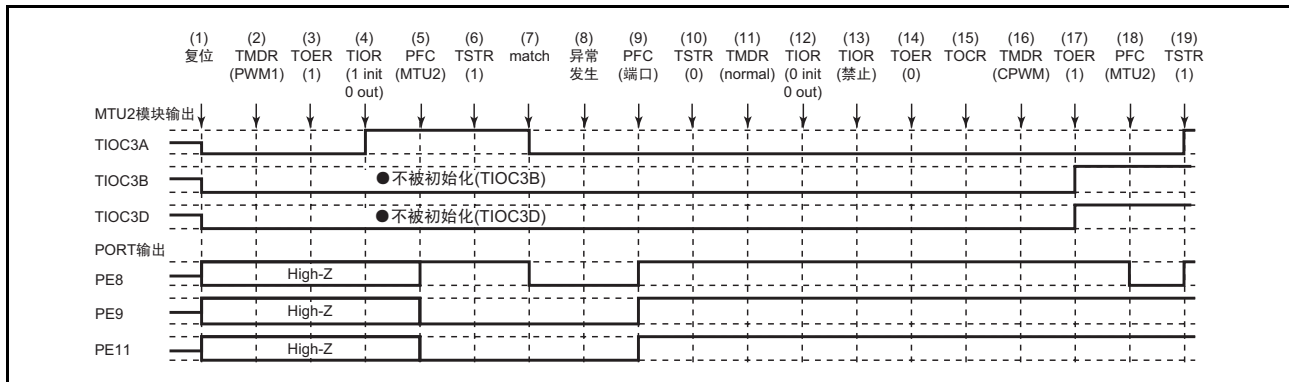


图 12.125 在 PWM 模式 1 中发生异常并且在互补 PWM 模式中恢复的情况

(1) ~ (10) 和图 12.121 共用。

(11) 为了对波形生成部进行初始化，必须设定正常模式。

(12) 必须通过 TIOR 对 PWM 模式 1 的波形生成部进行初始化。

(13) 必须通过 TIOR 禁止 PWM 模式 1 的波形生成部的运行

(14) 必须通过 TOER 禁止通道 3 和通道 4 的输出。

(15) 必须通过 TOCR 选择 PWM 的输出电平以及允许或者禁止周期输出。

(16) 设定互补 PWM。

(17) 必须通过 TOER 允许通道 3 和通道 4 的输出。

(18) 必须通过通用输入/输出端口设定为此模块的输出。

(19) 通过 TSTR 重新开始。

(12) 在 PWM 模式 1 的运行过程中发生异常并且在复位同步 PWM 模式中重新开始时的运行

在 PWM 模式 1 中发生异常并且重新设定后在复位同步 PWM 模式中重新开始时的说明如图 12.126 所示。

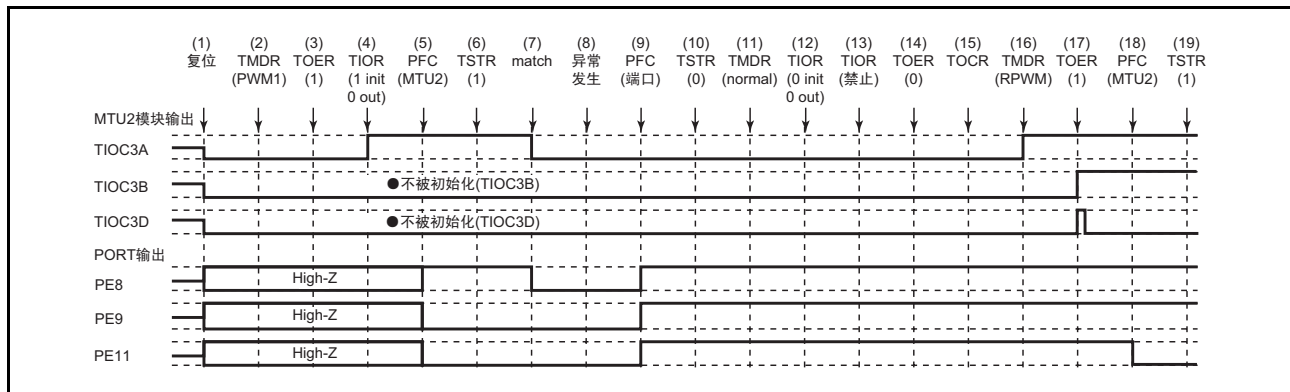


图 12.126 在 PWM 模式 1 中发生异常并且在复位同步 PWM 模式中恢复的情况

(1) ~ (14) 和图 12.125 共用。

(15) 必须通过 TOCR 选择复位同步 PWM 的输出电平以及允许或者禁止周期输出。

(16) 设定复位同步 PWM。

(17) 必须通过 TOER 允许通道 3 和通道 4 的输出。

(18) 必须通过通用输入/输出端口设定为此模块的输出。

(19) 通过 TSTR 重新开始。

(13) 在 PWM 模式 2 的运行过程中发生异常并且在正常模式中重新开始时的运行

在 PWM 模式 2 中发生异常并且重新设定后在正常模式中重新开始时的说明如图 12.127 所示。

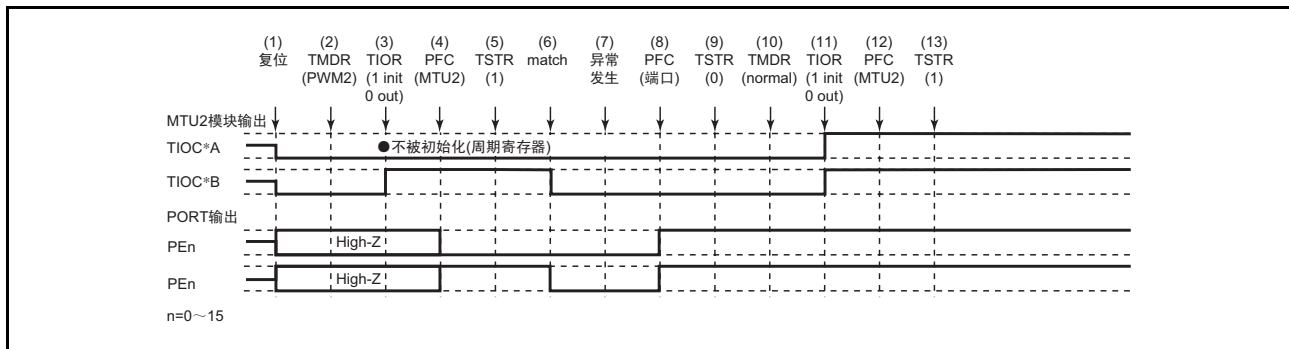


图 12.127 在 PWM 模式 2 中发生异常并且在正常模式中恢复的情况

- (1) 复位后，此模块的输出为低电平，端口为高阻抗状态。
- (2) 必须设定 PWM 模式 2。
- (3) 必须通过 TIOR 对引脚进行初始化（这是初始输出为高电平并且在比较匹配时输出低电平的例子。在 PWM 模式 2 中不对周期寄存器的引脚进行初始化。例如，TIOC*A 为周期寄存器的情况）。
- (4) 必须通过通用输入/输出端口设定为此模块的输出。
- (5) 通过 TSTR 开始计数运行。
- (6) 在比较匹配时输出低电平。
- (7) 发生异常。
- (8) 必须通过通用输入/输出端口设定为端口输出并且反相输出有效电平。
- (9) 通过 TSTR 停止计数运行。
- (10) 必须设定正常模式。
- (11) 必须通过 TIOR 对引脚进行初始化。
- (12) 必须通过通用输入/输出端口设定为此模块的输出。
- (13) 通过 TSTR 重新开始。

(14) 在 PWM 模式 2 的运行过程中发生异常并且在 PWM 模式 1 中重新开始时的运行

在 PWM 模式 2 中发生异常并且重新设定后在 PWM 模式 1 中重新开始时的说明如图 12.128 所示。

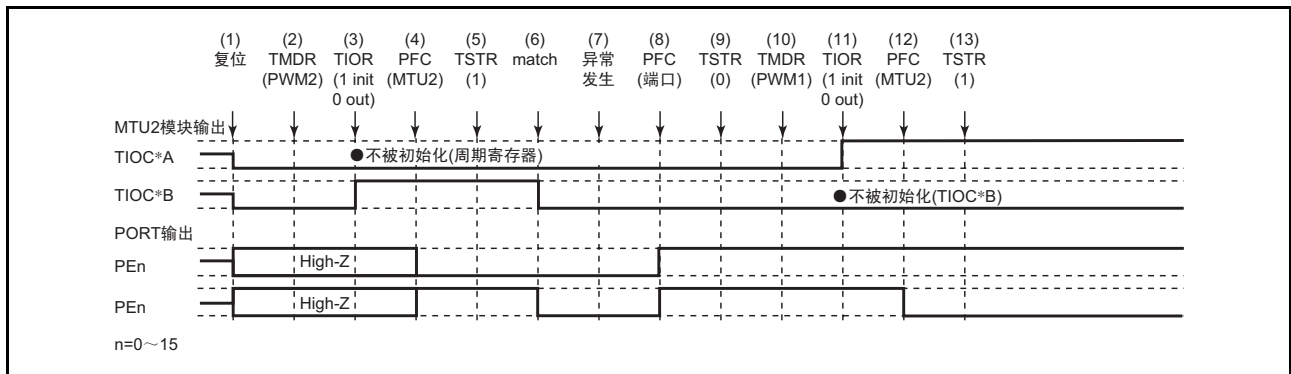


图 12.128 在 PWM 模式 2 中发生异常并且在 PWM 模式 1 中恢复的情况

(1) ~ (9) 和图 12.127 共用。

(10) 设定 PWM 模式 1。

(11) 必须通过 TIOR 对引脚进行初始化（在 PWM 模式 1 中不对 TIOC*B 进行初始化）。

(12) 必须通过通用输入/输出端口设定为此模块的输出。

(13) 通过 TSTR 重新开始。

(15) 在 PWM 模式 2 的运行过程中发生异常并且在 PWM 模式 2 中重新开始时的运行

在 PWM 模式 2 中发生异常并且重新设定后在 PWM 模式 2 中重新开始时的说明如图 12.129 所示。

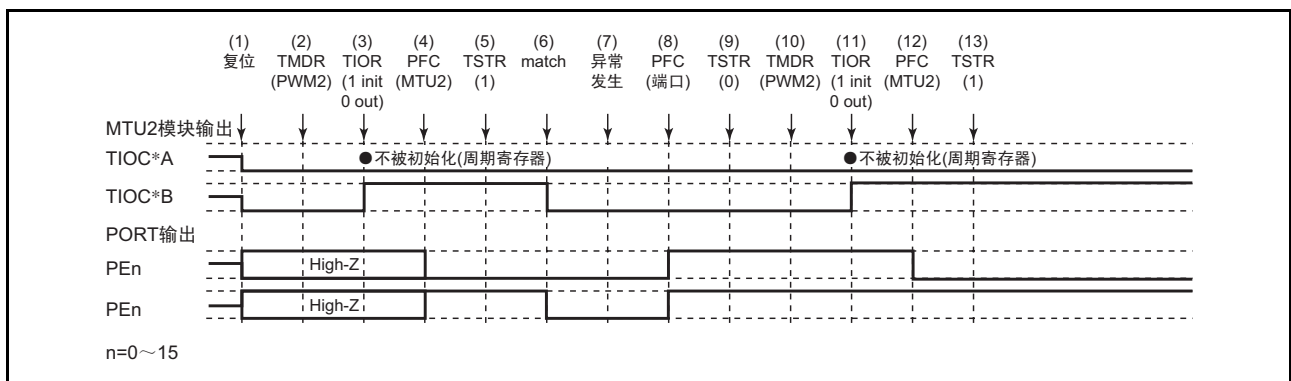


图 12.129 PWM 模式 2 中发生异常并且在 PWM 模式 2 中恢复的情况

(1) ~ (9) 和图 12.127 共用。

(10) 在 PWM 模式 2 中重新开始的情况下不需要此步骤。

(11) 必须通过 TIOR 对引脚进行初始化（在 PWM 模式 2 中不对周期寄存器的引脚进行初始化）。

(12) 必须通过通用输入/输出端口设定为此模块的输出。

(13) 通过 TSTR 重新开始。

(16) 在 PWM 模式 2 的运行过程中发生异常并且在相位计数模式中重新开始时的运行

在 PWM 模式 2 中发生异常并且重新设定后在相位计数模式中重新开始时的说明如图 12.130 所示。

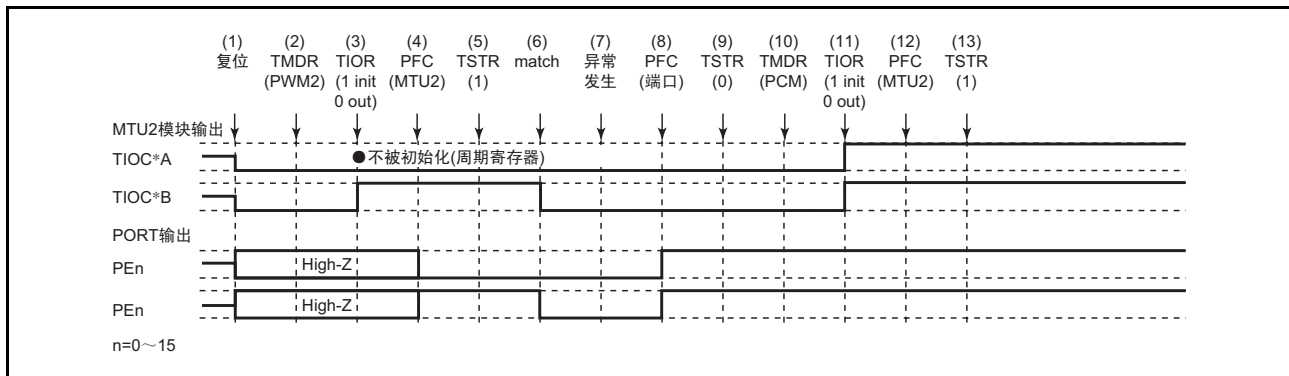


图 12.130 在 PWM 模式 2 中发生异常并且在相位计数模式中恢复的情况

- (1) ~ (9) 和图 12.127 共用。
- (10) 设定相位计数模式。
- (11) 必须通过 TIOR 对引脚进行初始化。
- (12) 必须通过通用输入/输出端口设定为此模块的输出。
- (13) 通过 TSTR 重新开始。

(17) 在相位计数模式的运行过程中发生异常并且在正常模式中重新开始时的运行

在相位计数模式中发生异常并且重新设定后在正常模式中重新开始时的说明如图 12.131 所示。

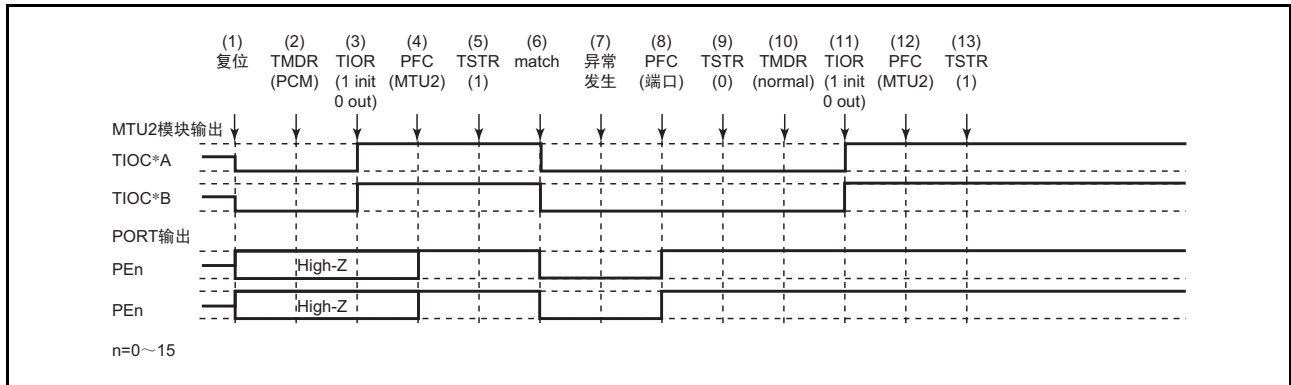


图 12.131 在相位计数模式中发生异常并且在正常模式中恢复的情况

- (1) 复位后，此模块的输出为低电平，端口为高阻抗状态。
- (2) 必须设定相位计数模式。
- (3) 必须通过 TIOR 对引脚进行初始化（这是初始输出为高电平并且在比较匹配时输出低电平的例子）。
- (4) 必须通过通用输入/输出端口设定为此模块的输出。
- (5) 通过 TSTR 开始计数运行。
- (6) 在比较匹配时输出低电平。
- (7) 发生异常。
- (8) 通过通用输入/输出端口设定为端口输出并且反相输出有效电平。
- (9) 通过 TSTR 停止计数运行。
- (10) 必须进行设定正常模式。
- (11) 必须通过 TIOR 对引脚进行初始化。
- (12) 必须通过通用输入/输出端口设定为此模块的输出。
- (13) 通过 TSTR 重新开始。

(18) 在相位计数模式的运行过程中发生异常并且在 PWM 模式 1 中重新开始时的运行

在相位计数模式中发生异常并且重新设定后在 PWM 模式 1 中重新开始时的说明如图 12.132 所示。

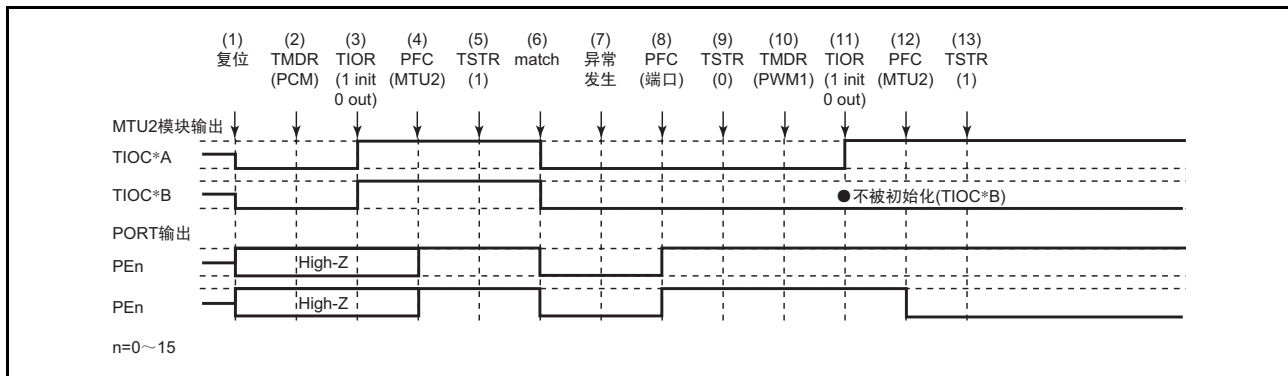


图 12.132 在相位计数模式中发生异常并且在 PWM 模式 1 中恢复的情况

(1) ~ (9) 和图 12.131 共用。

(10) 设定 PWM 模式 1。

(11) 必须通过 TIOR 对引脚进行初始化（在 PWM 模式 1 中不对 TIOC*B 进行初始化）。

(12) 必须通过通用输入/输出端口设定为此模块的输出。

(13) 通过 TSTR 重新开始。

(19) 在相位计数模式的运行过程中发生异常并且在 PWM 模式 2 中重新开始时的运行

在相位计数模式中发生异常并且重新设定后在 PWM 模式 2 中重新开始时的说明如图 12.133 所示。

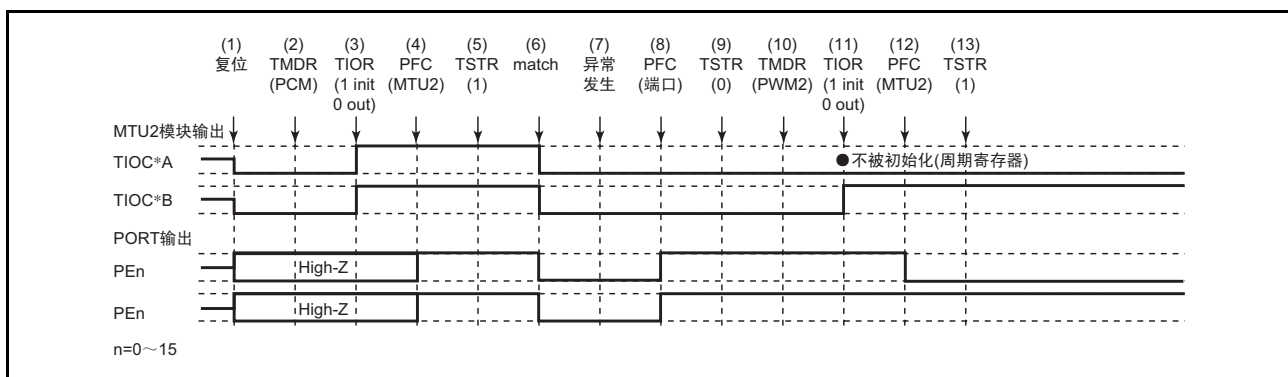


图 12.133 在相位计数模式中发生异常并且在 PWM 模式 2 中恢复的情况

(1) ~ (9) 和图 12.131 共用。

(10) 设定 PWM 模式 2。

(11) 必须通过 TIOR 对引脚进行初始化（在 PWM 模式 2 中不对周期寄存器的引脚进行初始化）。

(12) 必须通过通用输入/输出端口设定为此模块的输出。

(13) 通过 TSTR 重新开始。

(20) 在相位计数模式的运行过程中发生异常并且在相位计数模式中重新开始时的运行

在相位计数模式中发生异常并且重新设定后在相位计数模式中重新开始时的说明如图 12.134 所示。

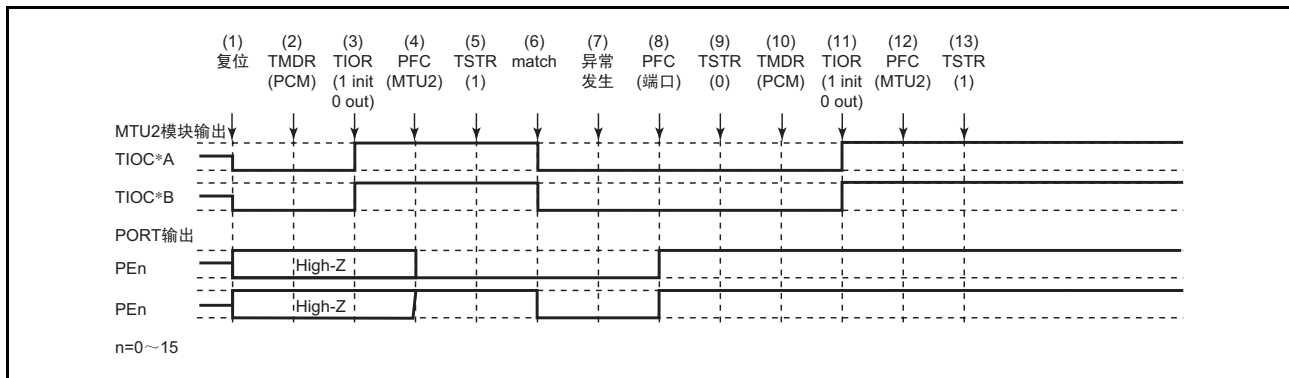


图 12.134 在相位计数模式中发生异常并且在相位计数模式中恢复的情况

- (1) ~ (9) 和图 12.131 共用。
- (10) 在相位计数模式中重新开始的情况下不需要此步骤。
- (11) 必须通过 TIOR 对引脚进行初始化。
- (12) 必须通过通用输入/输出端口设定为此模块的输出。
- (13) 通过 TSTR 重新开始。

(21) 在互补 PWM 模式的运行过程中发生异常并且在正常模式中重新开始时的运行

在互补 PWM 模式中发生异常并且重新设定后在正常模式中重新开始的说明如图 12.135 所示。

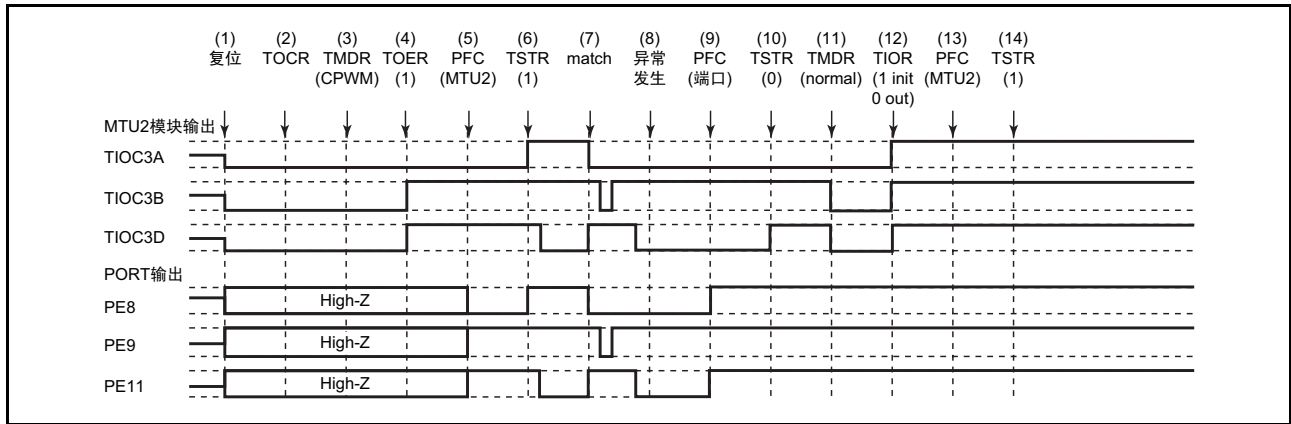


图 12.135 在互补 PWM 模式中发生异常并且在正常模式中恢复的情况

- (1) 复位后，此模块的输出为低电平，端口为高阻抗状态。
- (2) 必须通过 TOCR 选择互补 PWM 的输出电平以及允许或者禁止周期输出。
- (3) 设定互补 PWM。
- (4) 必须通过 TOER 允许通道 3 和通道 4 的输出。
- (5) 必须通过通用输入/输出端口设定为此模块的输出。
- (6) 通过 TSTR 开始计数运行。
- (7) 在比较匹配时输出互补 PWM 波形。
- (8) 发生异常。
- (9) 必须通过通用输入/输出端口设定为端口输出并且反相输出有效电平。
- (10) 通过 TSTR 停止计数运行（此模块的输出为互补 PWM 输出的初始值）。
- (11) 必须设定正常模式（此模块的输出为低电平）。
- (12) 必须通过 TIOR 对引脚进行初始化。
- (13) 必须通过通用输入/输出端口设定为此模块的输出。
- (14) 通过 TSTR 重新开始。

(22) 在互补 PWM 模式的运行过程中发生异常并且在 PWM 模式 1 中重新开始时的运行

在互补 PWM 模式中发生异常并且重新设定后在 PWM 模式 1 中重新开始时的说明如图 12.136 所示。

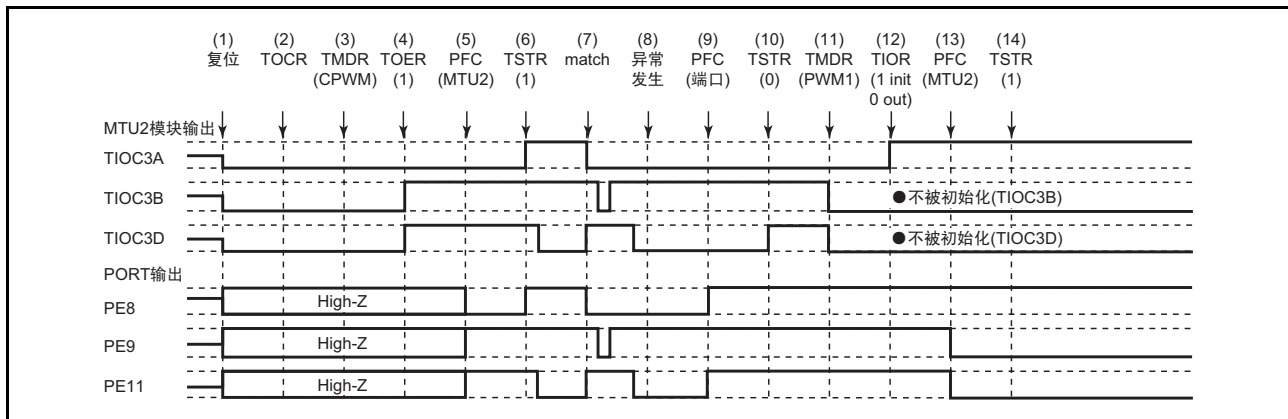


图 12.136 在互补 PWM 模式中发生异常并且在 PWM 模式 1 中恢复的情况

(1) ~ (10) 和图 12.135 共用。

(11) 必须设定 PWM 模式 1（此模块的输出为低电平）。

(12) 必须通过 TIOR 对引脚进行初始化（在 PWM 模式 1 中不对 TIOC*B 进行初始化）。

(13) 必须通过通用输入/输出端口设定为此模块的输出。

(14) 通过 TSTR 重新开始。

(23) 在互补 PWM 模式的运行过程中发生异常并且在互补 PWM 模式中重新开始时的运行

在互补 PWM 模式中发生异常并且重新设定后在互补 PWM 模式中重新开始时的说明如图 12.137 所示（从停止计数时的值开始重新设定周期和占空比的情况）。

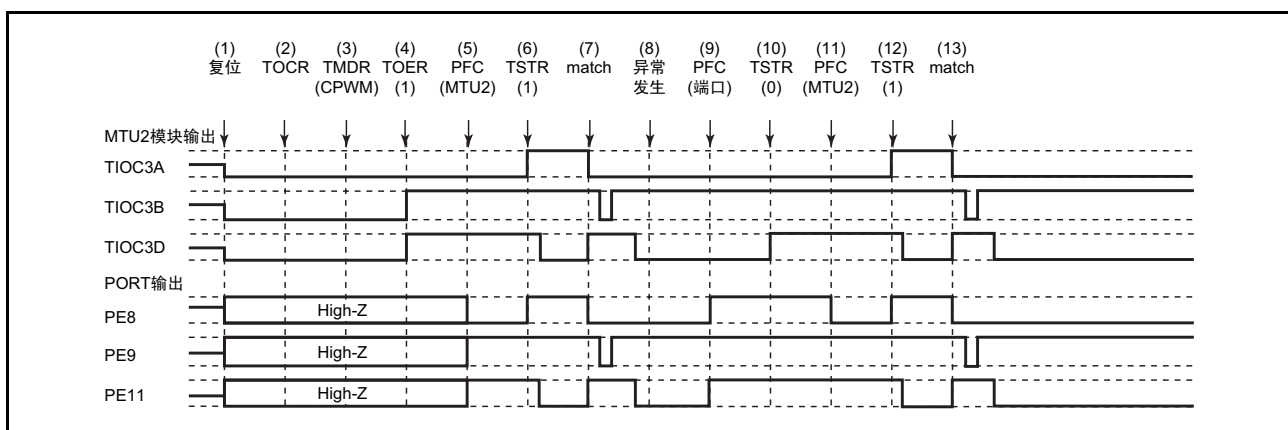


图 12.137 在互补 PWM 模式中发生异常并且在互补 PWM 模式中恢复的情况

(1) ~ (10) 和图 12.135 共用。

(11) 必须通过通用输入/输出端口设定为此模块的输出。

(12) 通过 TSTR 重新开始。

(13) 在比较匹配时输出互补 PWM 波形。

(24) 在互补 PWM 模式的运行过程中发生异常并且在互补 PWM 模式中重新开始时的运行

在互补 PWM 模式中发生异常并且重新设定后在互补 PWM 模式中重新开始时的说明如图 12.138 所示（从全新的值开始重新设定周期和占空比的情况）。

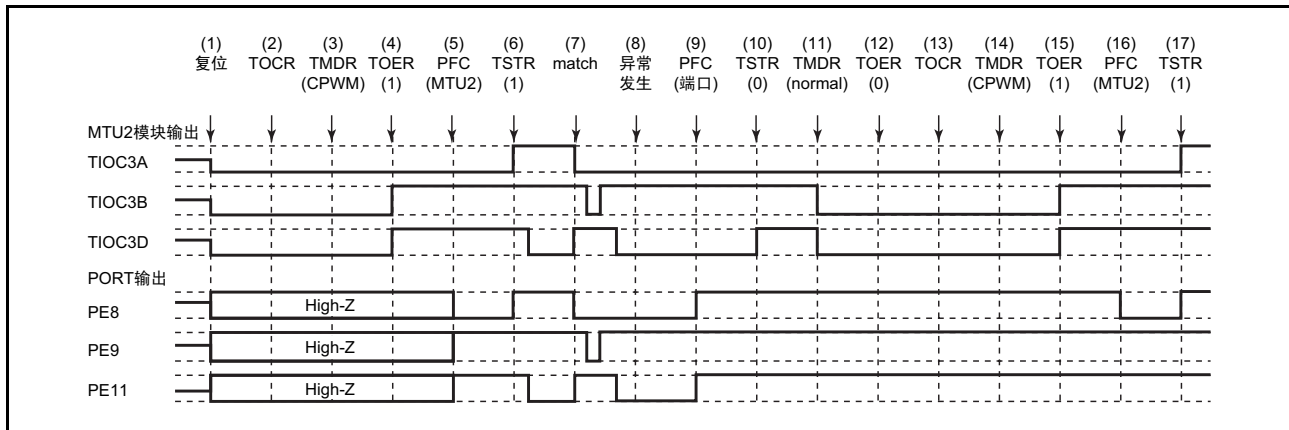


图 12.138 在互补 PWM 模式中发生异常并且在互补 PWM 模式中恢复的情况

(1) ~ (10) 和图 12.135 共用。

(11) 必须设定正常模式并且设定新的设定值（此模块的输出为低电平）。

(12) 必须通过 TOER 禁止通道 3 和通道 4 的输出。

(13) 必须通过 TOCR 选择互补 PWM 模式的输出电平以及允许或者禁止周期输出。

(14) 设定互补 PWM。

(15) 必须通过 TOER 允许通道 3 和通道 4 的输出。

(16) 必须通过通用输入/输出端口设定为此模块的输出。

(17) 通过 TSTR 重新开始。

(25) 在互补 PWM 模式的运行过程中发生异常并且在复位同步 PWM 模式中重新开始时的运行

在互补 PWM 模式中发生异常并且重新设定后在复位同步 PWM 模式中重新开始时的说明如图 12.139 所示。

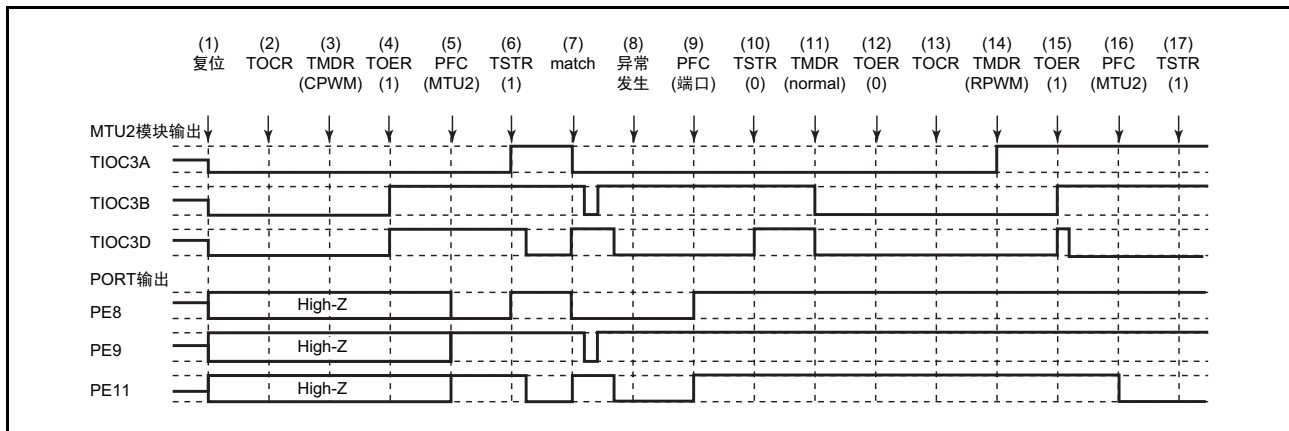


图 12.139 在互补 PWM 模式中发生异常并且在复位同步 PWM 模式中恢复的情况

(1) ~ (10) 和图 12.135 共用。

(11) 必须设定正常模式（此模块的输出为低电平）。

(12) 必须通过 TOER 禁止通道 3 和通道 4 的输出。

(13) 必须通过 TOCR 选择复位同步 PWM 模式的输出电平以及允许或者禁止周期输出。

(14) 设定复位同步 PWM。

(15) 必须通过 TOER 允许通道 3 和通道 4 的输出。

(16) 必须通过通用输入/输出端口设定为此模块的输出。

(17) 通过 TSTR 重新开始。

(26) 在复位同步 PWM 模式的运行过程中发生异常并且在正常模式中重新开始时的运行

在复位同步 PWM 模式中发生异常并且重新设定后在正常模式中重新开始时的说明如图 12.140 所示。

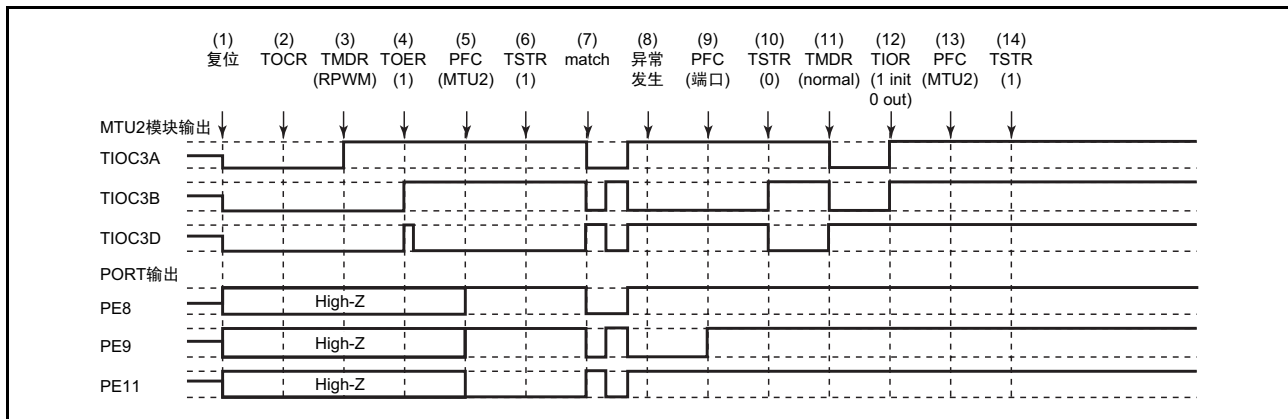


图 12.140 在复位同步 PWM 模式中发生异常并且在正常模式中恢复的情况

- (1) 复位后，此模块的输出为低电平，端口为高阻抗状态。
- (2) 必须通过 TOCR 选择复位同步 PWM 的输出电平以及允许或者禁止周期输出。
- (3) 设定复位同步 PWM。
- (4) 必须通过 TOER 允许通道 3 和通道 4 的输出。
- (5) 必须通过通用输入/输出端口设定为此模块的输出。
- (6) 通过 TSTR 开始计数运行。
- (7) 在比较匹配时输出复位同步 PWM 波形。
- (8) 发生异常。
- (9) 必须通过通用输入/输出端口设定为端口输出并且反相输出有效电平。
- (10) 通过 TSTR 停止计数运行（此模块的输出为复位同步 PWM 输出的初始值）。
- (11) 必须设定正常模式（此模块输出的正相为低电平，反相为高电平）。
- (12) 必须通过 TIOR 对引脚进行初始化。
- (13) 必须通过通用输入/输出端口设定为此模块的输出。
- (14) 通过 TSTR 重新开始。

(27) 在复位同步 PWM 模式的运行过程中发生异常并且在 PWM 模式 1 中重新开始时的运行

在复位同步 PWM 模式中发生异常并且重新设定后在 PWM 模式 1 中重新开始时的说明如图 12.141 所示。

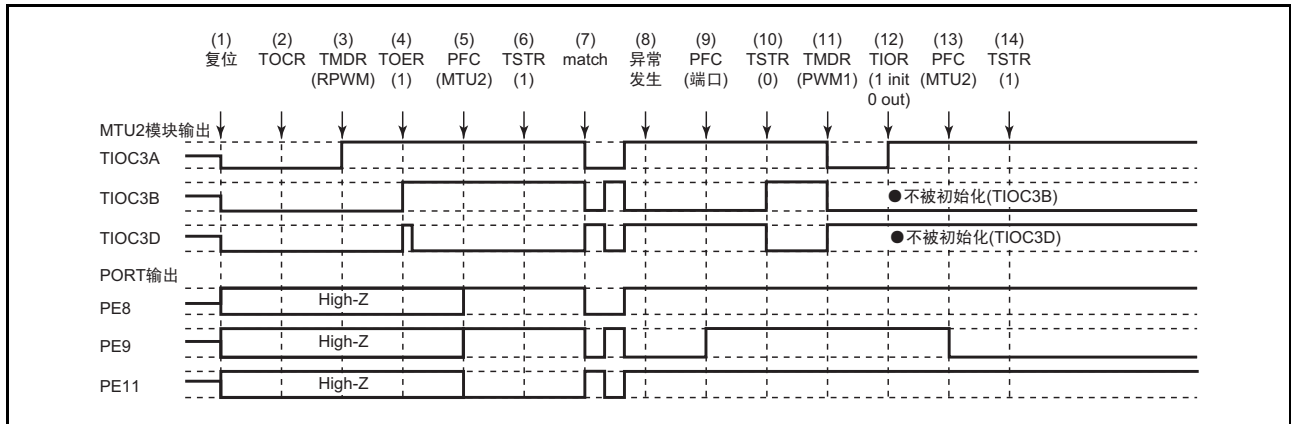


图 12.141 在复位同步 PWM 模式中发生异常并且在 PWM 模式 1 中恢复的情况

(1) ~ (10) 和图 12.140 共用。

(11) 必须设定 PWM 模式 1（此模块输出的正相为低电平，反相为高电平）。

(12) 必须通过 TIOR 对引脚进行初始化（在 PWM 模式 1 中不对 TIOC*B 进行初始化）。

(13) 必须通过通用输入/输出端口设定为此模块的输出。

(14) 通过 TSTR 重新开始。

(28) 在复位同步 PWM 模式的运行发生异常并且在互补 PWM 模式中重新开始时的运行

在复位同步 PWM 模式中发生异常并且重新设定后在互补 PWM 模式中重新开始时的说明如图 12.142 所示。

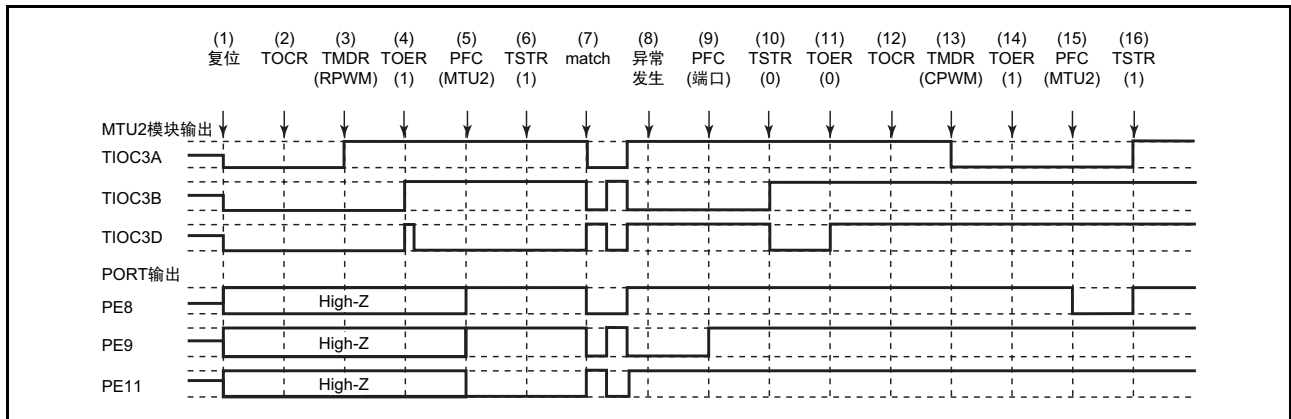


图 12.142 在复位同步 PWM 模式中发生异常并且在互补 PWM 模式中恢复的情况

(1) ~ (10) 和图 12.140 共用。

(11) 必须通过 TOER 禁止通道 3 和通道 4 的输出。

(12) 必须通过 TOCR 选择互补 PWM 的输出电平以及允许或者禁止周期输出。

(13) 设定互补 PWM（此模块的周期输出引脚为低电平）。

(14) 必须通过 TOER 允许通道 3 和通道 4 的输出。

(15) 必须通过通用输入/输出端口设定为此模块的输出。

(16) 通过 TSTR 重新开始。

(29) 在复位同步 PWM 模式的运行过程中发生异常并且在复位同步 PWM 模式中重新开始时的运行

在复位同步 PWM 模式中发生异常并且重新设定后在复位同步 PWM 模式中重新开始时的说明如图 12.143 所示。

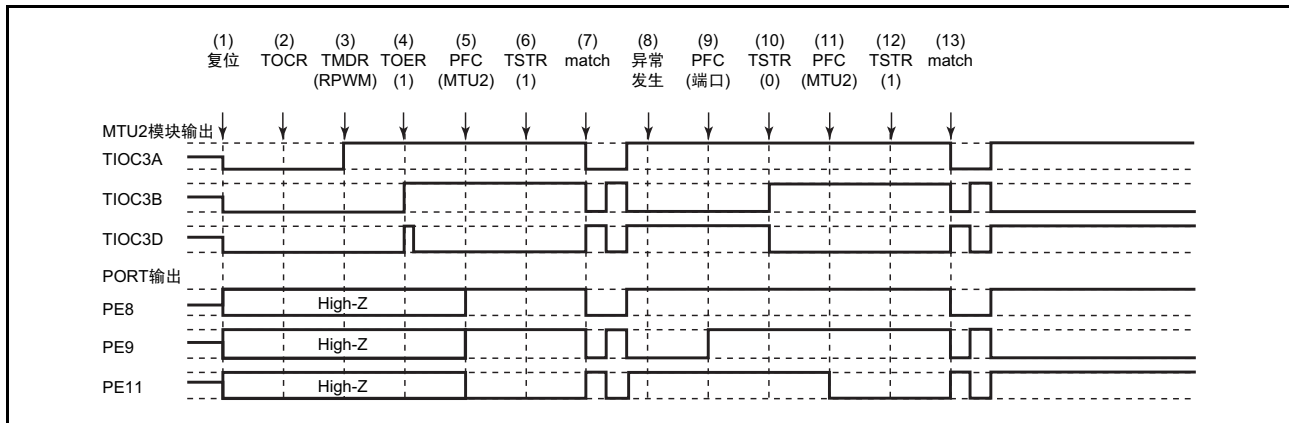


图 12.143 在复位同步 PWM 模式中发生异常并且在复位同步 PWM 模式中恢复的情况

- (1) ~ (10) 和图 12.140 共用。
- (11) 必须通过通用输入/输出端口设定为此模块的输出。
- (12) 通过 TSTR 重新开始。
- (13) 在比较匹配时输出复位同步 PWM 波形。

13. 比较匹配定时器

本 LSI 内置由 2 个通道 16 位定时器构成的比较匹配定时器。此模块有 16 位计数器，能按各设定的周期产生中断。

13.1 特点

- 2 个通道可独立选择 4 种计数器输入时钟。
可选择 4 种内部时钟（ $P0\phi/8$ 、 $P0\phi/32$ 、 $P0\phi/128$ 、 $P0\phi/512$ ）。
- 在比较匹配时，能通过设定直接存储器存取控制器，选择产生 DMAC 传送请求或者中断请求。
- 在不使用此模块时，为了降低功耗，能停止给此模块提供时钟，使其停止运行。

框图如图 13.1 所示。

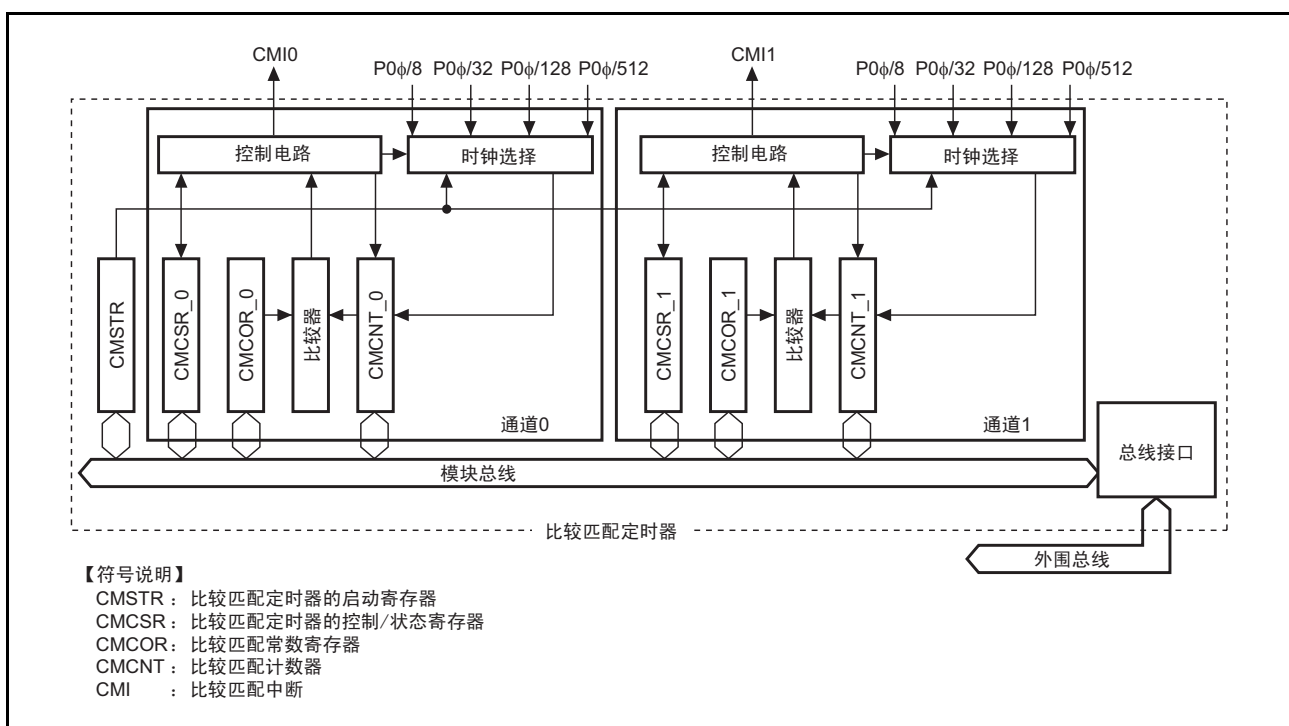


图 13.1 框图

13.2 寄存器说明

寄存器结构如表 13.1 所示。

表 13.1 寄存器结构

通道	寄存器名	略称	R/W	初始值	地址	存取长度
共用	比较匹配定时器的启动寄存器	CMSTR	R/W	H'0000	H'FFFEC000	16
0	比较匹配定时器的控制 / 状态寄存器 _0	CMCSR_0	R/W	H'0000	H'FFFEC002	16
	比较匹配计数器 _0	CMCNT_0	R/W	H'0000	H'FFFEC004	8、16
	比较匹配常数寄存器 _0	CMCOR_0	R/W	H'FFFF	H'FFFEC006	8、16
1	比较匹配定时器的控制 / 状态寄存器 _1	CMCSR_1	R/W	H'0000	H'FFFEC008	16
	比较匹配计数器 _1	CMCNT_1	R/W	H'0000	H'FFFEC00A	8、16
	比较匹配常数寄存器 _1	CMCOR_1	R/W	H'FFFF	H'FFFEC00C	8、16

13.2.1 比较匹配定时器的启动寄存器（CMSTR）

CMSTR 是 16 位寄存器，选择比较匹配计数器（CMCNT）的运行或者停止。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STR1	STR0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 2	—	全 0	R	保留位 读写值都为“0”。
1	STR1	0	R/W	计数开始 1 选择比较匹配计数器 _1 的运行或者停止。 0: CMCNT_1 停止计数 1: CMCNT_1 开始计数
0	STR0	0	R/W	计数开始 0 选择比较匹配计数器 _0 的运行或者停止。 0: CMCNT_0 停止计数 1: CMCNT_0 开始计数

13.2.2 比较匹配定时器的控制 / 状态寄存器 (CMCSR)

CMCSR 是 16 位寄存器，表示比较匹配的发生，允许或者禁止中断以及设定计数器输入时钟。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CMF	CMIE	—	—	—	—	—	CKS[1:0]
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/(W)*	R/W	R	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 8	—	全 0	R	保留位 读写值都为“0”。
7	CMF	0	R/(W)*	比较匹配标志 这是表示 CMCNT 和 CMCOR 的值是否相同的标志。 0: CMCNT 和 CMCOR 的值不相同 [清除条件] • 在读 CMF 位的“1”后给 CMF 写“0”时 1: CMCNT 和 CMCOR 的值相同
6	CMIE	0	R/W	比较匹配中断允许 在 CMCNT 和 CMCOR 的值相同时 (CMF=1)，选择是允许还是禁止发生比较匹配中断 (CMI)。 0: 禁止比较匹配中断 (CMI) 1: 允许比较匹配中断 (CMI)
5 ~ 2	—	全 0	R	保留位 读写值都为“0”。
1、0	CKS[1:0]	00	R/W	时钟选择 从外围时钟 (P0φ) 分频后得到的 4 种内部时钟中选择 CMCNT 的输入时钟。当 CMSTR 的 STR 位被置“1”时，CMCNT 通过 CKS[1:0] 位所选的时钟开始计数。 00: P0φ/8 01: P0φ/32 10: P0φ/128 11: P0φ/512

【注】 * 为了清除标志，只能在读“1”后写“0”。

13.2.3 比较匹配计数器 (CMCNT)

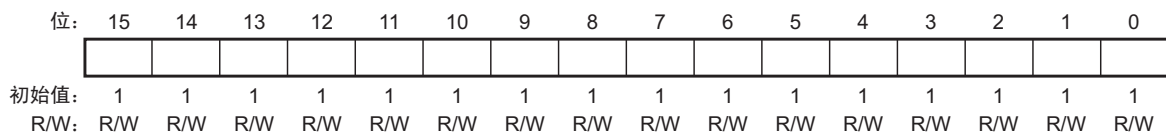
CMCNT 是 16 位寄存器，用作递增计数器。如果通过 CMCSR 的 CKS[1:0] 位选择计数器的输入时钟，并且将 CMST 的 STR 位置“1”，CMCNT 就通过所选的时钟开始计数。当 CMCNT 的值和比较匹配常数寄存 (CMCOR) 的值相同时，将 CMCNT 清“H'0000”，并且将 CMCSR 的 CMF 标志置“1”。

当将比较匹配定时器的启动寄存器 (CMSTR) 对应通道的计数开始位从“1”清“0”时，将 CMCNT 初始化为“H'0000”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

13.2.4 比较匹配常数寄存器 (CMCOR)

CMCOR 是 16 位寄存器，设定和 CMCNT 比较匹配前的时间。



13.3 运行说明

13.3.1 周期计数运行

如果通过 CMCSR 的 CKS[1:0] 位选择内部时钟，并且将 CMSTR 的 STR 位置 “1”，CMCNT 就通过所选的时钟开始递增计数。当 CMCNT 的值和 CMCOR 的值相同时，将 CMCNT 清 “H'0000”，并且将 CMCSR 的 CMF 标志置 “1”。此时，如果 CMCSR 寄存器的 CMIE 位已被置 “1”，就请求比较匹配中断 (CMI)。CMCNT 从 “H'0000” 重新开始递增计数。

比较匹配计数器的运行如图 13.2 所示。

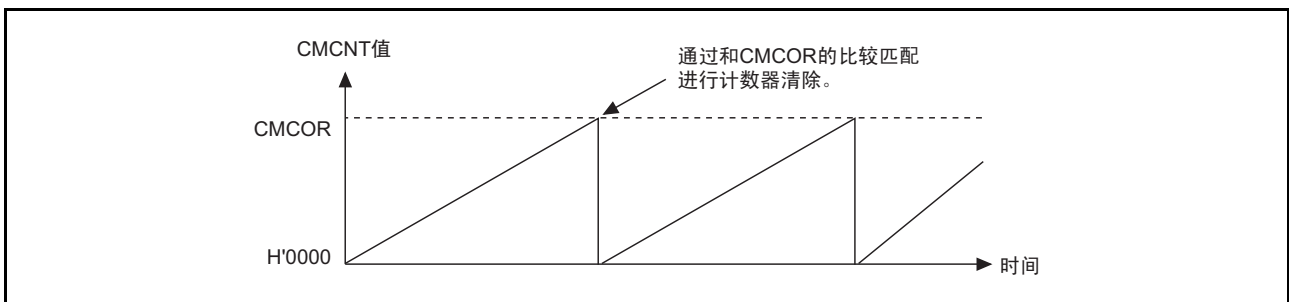


图 13.2 计数器的运行

13.3.2 CMCNT 的计数时序

能通过 CMCSR 的 CKS[1:0] 位，从外围时钟 (P0φ) 分频后得到的 4 种时钟 (P0φ/8、P0φ/32、P0φ/128、P0φ/512) 中选择一种时钟。该时序如图 13.3 所示。

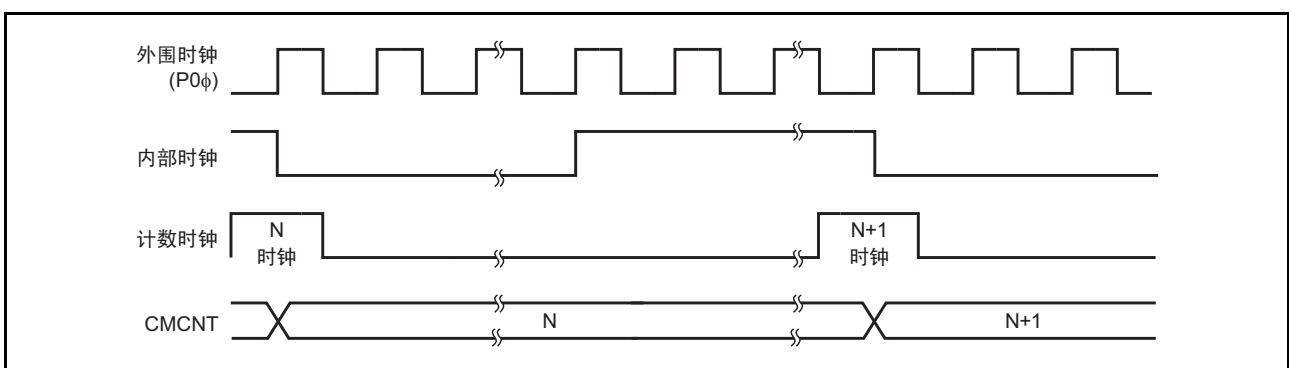


图 13.3 计数时序

13.4 中断

13.4.1 中断源和 DMA 传送请求

此模块有各通道的比较匹配中断，并且分别分配了独立的向量地址。当比较匹配标志（CMF）和中断允许位（CMIE）都被置“1”时，就输出相应的中断请求。在通过中断请求启动 CPU 中断时，能通过设定中断控制器更改通道之间的优先级。详细内容请参照“7. 中断控制器”。

必须在用户异常处理程序中清除 CMF 位，否则会再次产生中断。另外，在请求比较匹配中断时，能通过设定中断控制器启动直接存储器存取控制器，此时不向 CPU 请求中断。在不进行直接存储器存取控制器的启动设定时，向 CPU 请求中断。在通过直接存储器存取控制器进行数据传送时，自动清除 CMF 位。

13.4.2 比较匹配标志的置位时序

如果 CMCOR 和 CMCNT 相同，就在相同的最后状态（将 CMCNT 的值更新为“H'0000”时）产生比较匹配信号，并且将 CMCSR 的 CMF 位置“1”。即，如果在 CMCOR 和 CMCNT 相同后不输入用于 CMCNT 计数器的时钟，就不产生比较匹配信号。CMF 位的置位时序如图 13.4 所示。

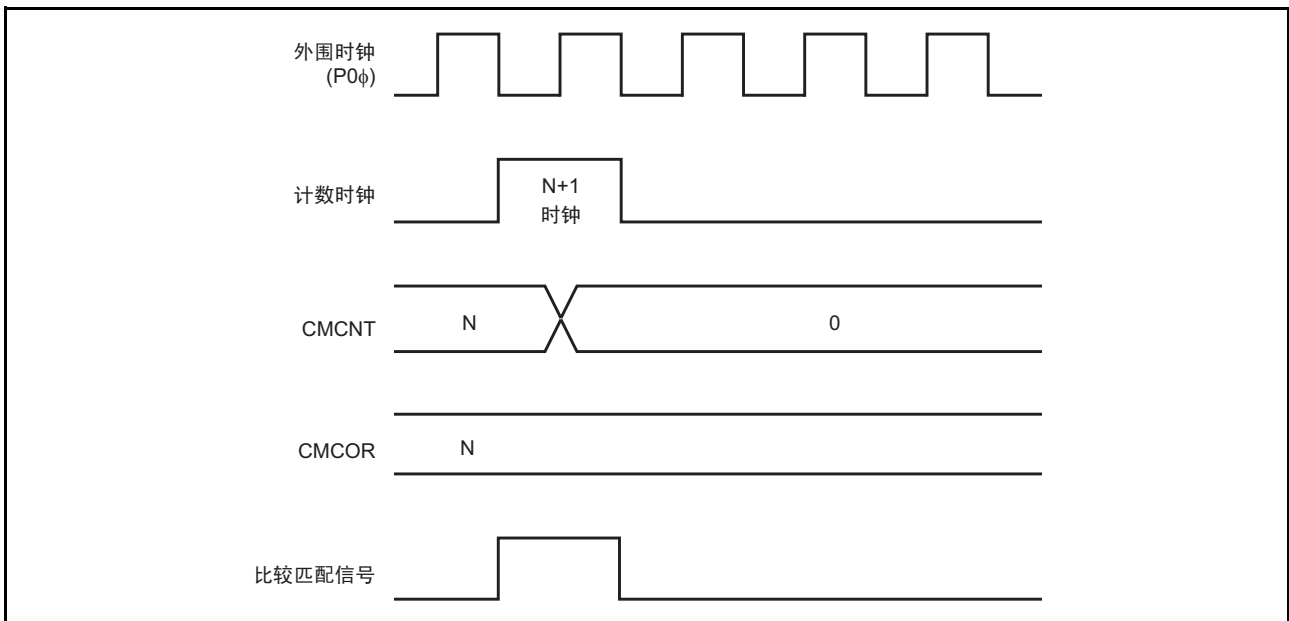


图 13.4 CMF 置位时序

13.4.3 比较匹配标志的清除时序

通过在读 CMF 的“1”后写“0”，清除 CMCSR 的 CMF 位。但是，如果启动直接存储器存取控制器，就在直接存储器存取控制器进行数据传送时自动清除 CMF 位。

13.5 使用时的注意事项

13.5.1 CMCNT 的写操作和比较匹配的竞争

如果在 CMCNT 计数器的写周期中的 T2 状态产生比较匹配信号，就不写 CMCNT 计数器而优先清除 CMCNT 计数器。此时序如图 13.5 所示。

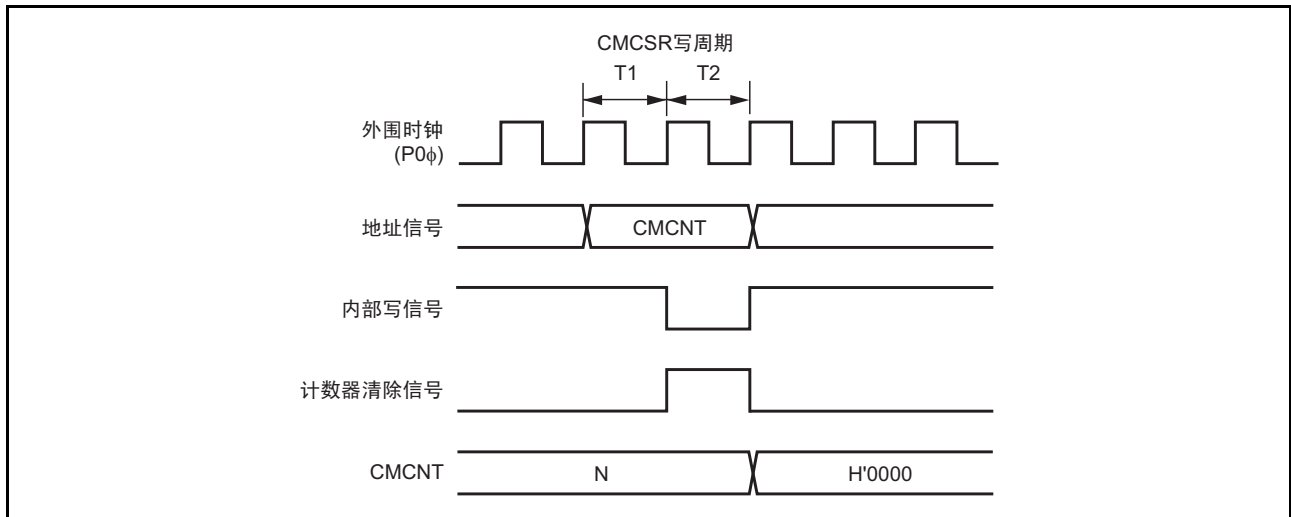


图 13.5 CMCNT 的写操作和比较匹配的竞争

13.5.2 CMCNT 的字写和递增计数的竞争

即使在 CMCNT 计数器的字写周期中的 T2 状态发生递增计数，也不进行递增计数而优先写计数器。此时序如图 13.6 所示。

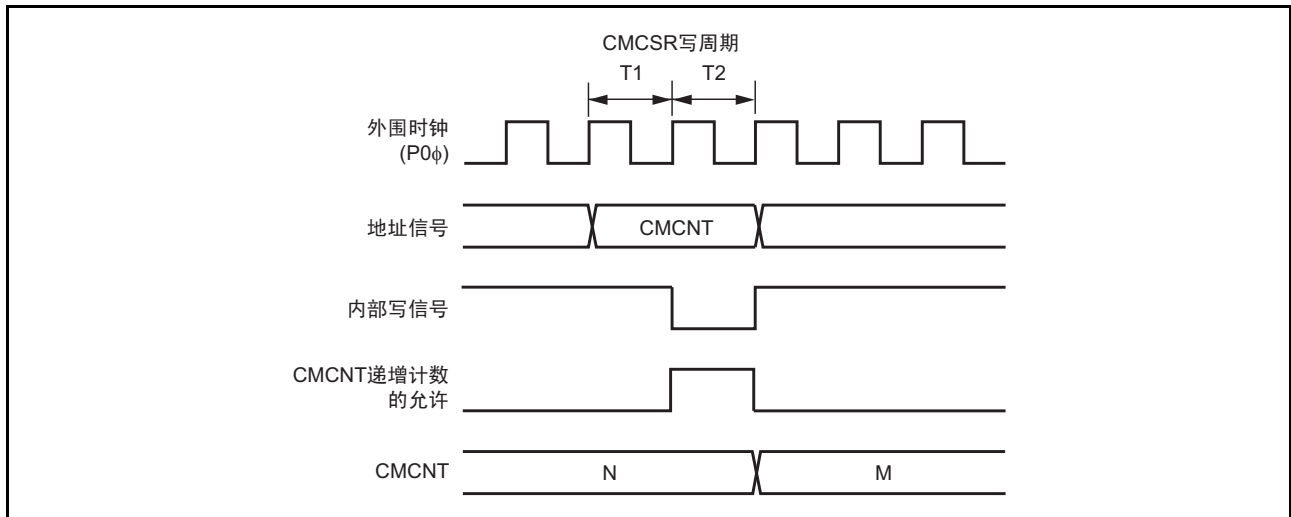


图 13.6 CMCNT 的字写和递增计数的竞争

13.5.3 CMCNT 的字节写和递增计数的竞争

即使在 CMCNT 的字节写周期中的 T2 状态发生递增计数，也不对写对象的写数据进行递增计数而优先写计数器，并且也不对非写对象的字节数据进行递增计数而保持写之前的内容。

在 CMCNTH 写周期中的 T2 状态发生递增计数时的时序如图 13.7 所示。

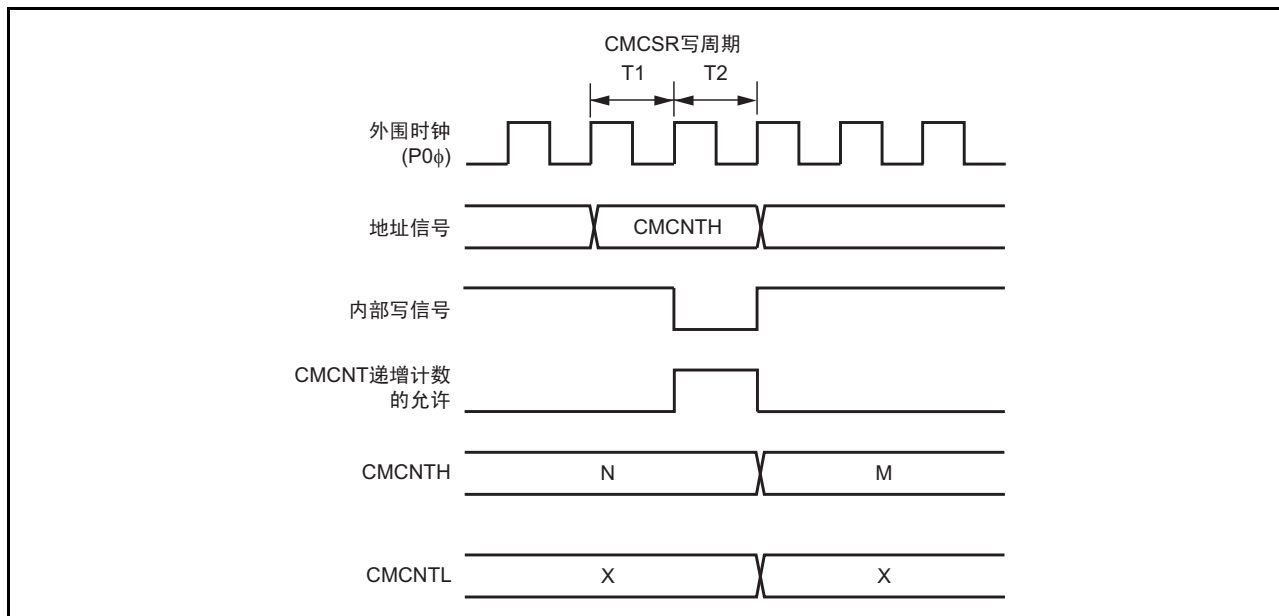


图 13.7 CMCNT 的字节写和递增计数的竞争

13.5.4 CMCNT 和 CMCOR 的比较匹配

在 CMCNT 为计数停止的状态下，不能给 CMCNT 和 CMCOR 设定相同的值。

14. 看门狗定时器

本 LSI 内置看门狗定时器，当因系统失控等原因不能改写计数器的值而发生上溢时，将上溢信号 ($\overline{\text{WDTOVF}}$) 输出到外部，同时能产生本 LSI 的内部复位信号。

WDT 是 1 个通道的定时器，在解除软件待机模式时用于时钟振荡稳定时间的计数，并且还能用作通常的看门狗定时器或者间隔定时器。

14.1 特点

- 能用于确保时钟振荡稳定时间。
在解除软件待机模式时使用。
- 能进行看门狗定时器模式和间隔定时器模式的切换。
- 在看门狗定时器模式中输出 $\overline{\text{WDTOVF}}$ 信号。
如果计数器发生上溢，就将 $\overline{\text{WDTOVF}}$ 信号输出到外部。此时，能选择是否同时对本 LSI 内部进行复位。此内部复位能选择上电复位或者手动复位。
- 在间隔定时器模式中产生中断。
因计数器上溢而产生间隔定时器中断。
- 可选择 8 种计数器的输入时钟。
能从外围时钟分频后得到的 8 种分频时钟 ($\text{P0}\phi\times 1 \sim \times 1/16384$) 中进行选择。

框图如图 14.1 所示。

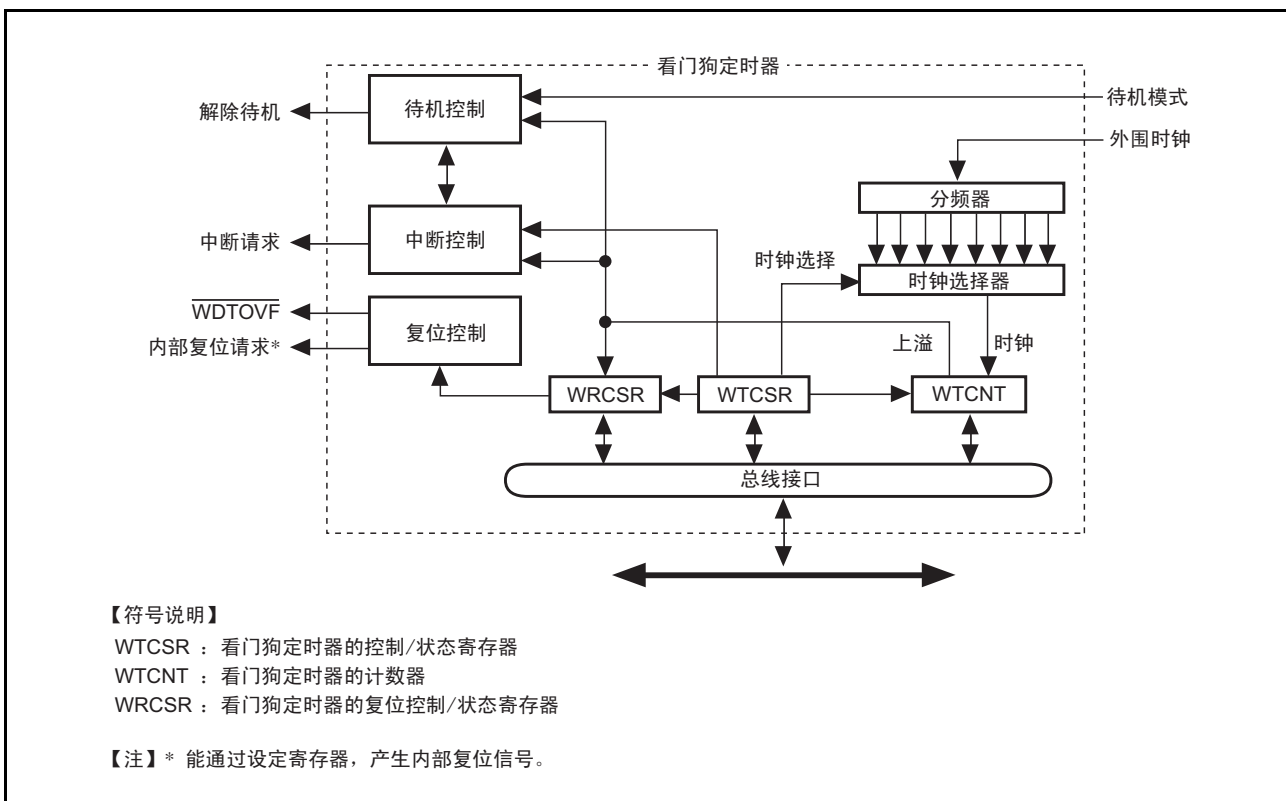


图 14.1 框图

14.2 输入 / 输出引脚

引脚结构如表 14.1 所示。

表 14.1 引脚结构

名称	引脚名	输入 / 输出	功能
看门狗定时器的上溢	WDTOVF	输出	在看门狗定时器模式中，输出计数器的上溢信号。

14.3 寄存器说明

寄存器结构如表 14.2 所示。

表 14.2 寄存器结构

寄存器名	略称	R/W	初始值	地址	存取长度
看门狗定时器的计数器	WTCNT	R/W	H'00	H'FFFE0002	16*
看门狗定时器的控制 / 状态寄存器	WTCSR	R/W	H'18	H'FFFE0000	16*
看门狗定时器的复位控制 / 状态寄存器	WRCSR	R/W	H'1F	H'FFFE0004	16*

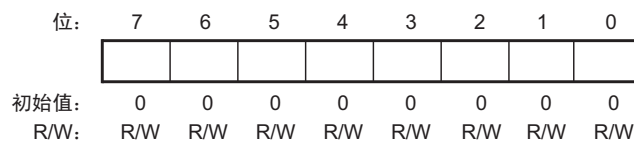
【注】 * 存取长度请参照“14.3.4 存取寄存器时的注意事项”。

14.3.1 看门狗定时器的计数器（WTCNT）

WTCNT 是 8 位可读写寄存器，也是通过所选时钟进行递增计数的计数器。如果发生上溢，就会在看门狗定时器模式中产生看门狗定时器的上溢信号（ $\overline{\text{WDTOVF}}$ ），而在间隔定时器模式中产生中断。

必须在将高位字节置“H'5A”后以字为单位写 WTCNT，并且必须以字节为单位读 WTCNT。

【注】 为了此寄存器不被误改写，写数据的方法和一般寄存器不同。详细内容请参照“14.3.4 存取寄存器时的注意事项”。



14.3.2 看门狗定时器的控制 / 状态寄存器 (WTCSR)

WTCSR 是 8 位可读写寄存器，由选择计数时钟的位、上溢标志位和允许位构成。

当用于软件待机模式解除时的时钟振荡稳定时间的计数时，在计数器发生上溢后仍保持该值。

必须在将高位字节置“H'A5”后以字为单位写 WTCSR，并且必须以字节为单位读 WTCSR。

【注】 为了此寄存器不被误改写，写数据的方法和一般寄存器不同。详细内容请参照“14.3.4 存取寄存器时的注意事项”。

位:	7	6	5	4	3	2	1	0
	IOVF	WT/IT	TME	—	—	CKS[2:0]		
初始值:	0	0	0	1	1	0	0	0
R/W:	R/(W)	R/W	R/W	R	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明																		
7	IOVF	0	R/(W)	间隔定时器上溢 表示在间隔定时器模式中 WTCNT 发生上溢，而在看门狗定时器模式中此位不被置位。 0: 没有发生上溢 1: 在间隔定时器模式中，WTCNT 发生上溢。 [清除条件] • 在读 IOVF 后写“0”																		
6	WT/IT	0	R/W	定时器模式选择 指定是用作看门狗定时器还是用作间隔定时器。 0: 间隔定时器模式 1: 看门狗定时器模式 【注】 在看门狗定时器模式中，当 WTCNT 发生上溢时，将 $\overline{\text{WDTOVF}}$ 信号输出到外部。 如果在此模块的运行过程中改写 WT/IT，就可能无法正确地进行递增计数。																		
5	TME	0	R/W	定时器允许 设定开始或者停止定时器的运行。如果在软件待机模式中或者在更改时钟频率时使用此模块，就必须将此位置“0”。 0: 禁止定时器的运行 停止递增计数，保持 WTCNT 的值。 1: 允许定时器的运行																		
4、3	—	全 1	R	保留位 读写值都为“1”。																		
2~0	CKS[2:0]	000	R/W	时钟选择 从外围时钟 (P0 ϕ) 分频后得到的 8 种时钟中选择用于 WTCNT 计数的时钟。括弧内为外围时钟 (P0 ϕ) =33.33MHz 时的上溢周期值。 <table style="margin-left: 20px; border: none;"> <tr> <td style="padding-right: 20px;">时钟分频比</td> <td>上溢周期</td> </tr> <tr> <td>000: 1×P0ϕ</td> <td>(7.7μs)</td> </tr> <tr> <td>001: 1/64×P0ϕ</td> <td>(490μs)</td> </tr> <tr> <td>010: 1/128×P0ϕ</td> <td>(979μs)</td> </tr> <tr> <td>011: 1/256×P0ϕ</td> <td>(2.0ms)</td> </tr> <tr> <td>100: 1/512×P0ϕ</td> <td>(3.9ms)</td> </tr> <tr> <td>101: 1/1024×P0ϕ</td> <td>(7.8ms)</td> </tr> <tr> <td>110: 1/4096×P0ϕ</td> <td>(31ms)</td> </tr> <tr> <td>111: 1/16384×P0ϕ</td> <td>(125ms)</td> </tr> </table> 【注】 如果在此模块的运行过程中改写 CKS[2:0] 位，就可能无法正确地进行递增计数。在改写 CKS[2:0] 位时，必须停止此模块的运行。	时钟分频比	上溢周期	000: 1×P0 ϕ	(7.7 μ s)	001: 1/64×P0 ϕ	(490 μ s)	010: 1/128×P0 ϕ	(979 μ s)	011: 1/256×P0 ϕ	(2.0ms)	100: 1/512×P0 ϕ	(3.9ms)	101: 1/1024×P0 ϕ	(7.8ms)	110: 1/4096×P0 ϕ	(31ms)	111: 1/16384×P0 ϕ	(125ms)
时钟分频比	上溢周期																					
000: 1×P0 ϕ	(7.7 μ s)																					
001: 1/64×P0 ϕ	(490 μ s)																					
010: 1/128×P0 ϕ	(979 μ s)																					
011: 1/256×P0 ϕ	(2.0ms)																					
100: 1/512×P0 ϕ	(3.9ms)																					
101: 1/1024×P0 ϕ	(7.8ms)																					
110: 1/4096×P0 ϕ	(31ms)																					
111: 1/16384×P0 ϕ	(125ms)																					

14.3.3 看门狗定时器的复位控制 / 状态寄存器 (WRCSR)

WRCSR 是 8 位可读写寄存器，控制因看门狗定时器的计数器 (WTCNT) 上溢产生的内部复位信号。

【注】 为了此寄存器不被误改写，写数据的方法和一般寄存器不同。详细内容请参照“14.3.4 存取寄存器时的注意事项”。

位:	7	6	5	4	3	2	1	0
	WOVF	RSTE	RSTS	—	—	—	—	—
初始值:	0	0	0	1	1	1	1	1
R/W:	R/(W)	R/W	R/W	R	R	R	R	R

位	位名	初始值	R/W	说明
7	WOVF	0	R/(W)	看门狗定时器上溢 表示在看门狗定时器模式中 WTCNT 发生上溢，而在间隔定时器模式中此位不被置位。 0: 没有发生上溢 1: 在看门狗定时器模式中，WTCNT 发生上溢。 [清除条件] • 在读 WOVF 后写“0”
6	RSTE	0	R/W	复位允许 在看门狗定时器模式中 WTCNT 发生上溢时，选择是否产生对本 LSI 内部进行复位的信号。在间隔定时器模式中，忽视此位的设定值。 0: 在 WTCNT 发生上溢时不进行内部复位* 1: 在 WTCNT 发生上溢时进行内部复位 【注】* 不对本 LSI 内部进行复位，但是对此模块内的 WTCNT 和 WTCSR 进行复位。
5	RSTS	0	R/W	复位选择 在看门狗定时器模式中，选择 WTCNT 发生上溢时的复位种类。在间隔定时器模式中，忽视此位的设定值。 0: 上电复位 1: 手动复位
4 ~ 0	—	全 1	R	保留位 读写值都为“1”。

14.3.4 存取寄存器时的注意事项

为了看门狗定时器计数器（WTCNT）、看门狗定时器的控制 / 状态寄存器和看门狗定时器复位的控制 / 状态寄存器（WRCSR）不易被改写，写数据的方法和一般寄存器不同。必须按照以下方法进行读写：

(1) WTCNT 和 WTCSR 的写操作

在写 WTCNT 和 WTCSR 时，必须使用字传送指令，而字节传送指令和长字传送指令不能用于这些寄存器的写操作。

如图 14.2 所示，在写 WTCNT 时，必须在将高位字节置“H'5A”并且将低位字节置为写数据后进行传送；在写 WTCSR 时，必须在将高位字节置“H'A5”并且将低位字节置为写数据后进行传送。如此进行传送，就能将低位字节的数据写到 WTCNT 或者 WTCSR。

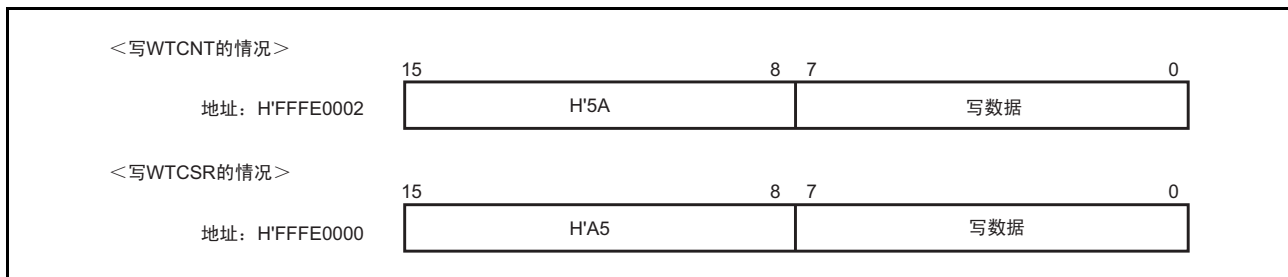


图 14.2 WTCNT 和 WTCSR 的写操作

(2) WRCSR 的写操作

在写 WRCSR 时，必须对地址 H'FFFE0004 进行字传送，而字节传送指令和长字传送指令不能用于此寄存器的写操作。

如图 14.3 所示，给 WOVF 位（bit7）写“0”的方法与写 RSTE 位（bit6）和 RSTS 位（bit5）的方法不同。

在给 WOVF 位写“0”时，必须在将高位字节置“H'A5”并且将低位字节置“H'00”后进行传送，从而能将 WOVF 位清“0”，此时不影响 RSTE 位和 RSTS 位。在写 RSTE 位和 RSTS 位时，必须在将高位字节置“H'5A”并且将低位字节置为写数据后进行传送，从而能将低位字节的 bit6 和 bit5 的值分别写到 RSTE 位和 RSTS 位，此时不影响 WOVF 位。

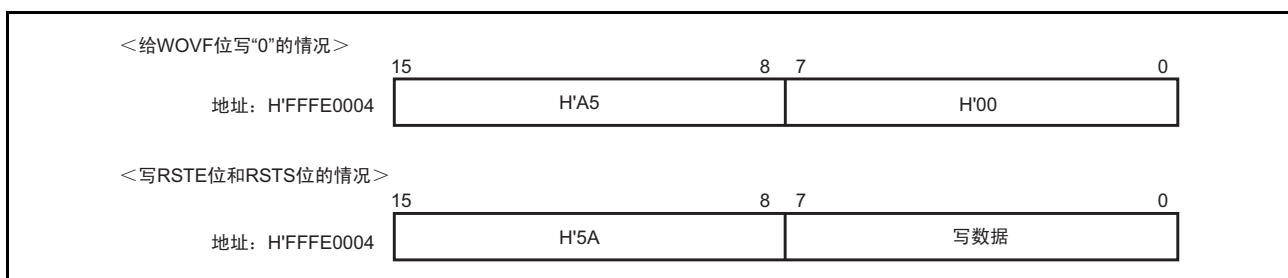


图 14.3 WRCSR 的写操作

(3) WTCNT、WTCSR 和 WRCSR 的读操作

能用和普通寄存器相同的方法读这些寄存器。WTCSR 被分配到地址 H'FFFE0000，WTCNT 被分配到地址 H'FFFE0002，WRCSR 被分配到地址 H'FFFE0004。在读这些寄存器时，必须使用字节传送指令。

14.4 使用方法

14.4.1 软件待机模式的解除步骤

在通过 NMI 等中断解除软件待机模式时使用此模块。步骤如下所示（因为在通过复位进行解除时此模块不运行，所以必须在时钟振荡稳定前保持 RES 引脚的低电平）。

1. 必须在转移到软件待机模式前将 WTCSR 寄存器的 TME 位置 “0”。如果将 TME 位置 “1”，就可能在计数上溢时产生不正常的复位或者间隔定时器中断。
2. 预先给 WTCSR 的 CKS[2:0] 位设定要使用的计数时钟种类和计数器的初始值。在设定这些值时，计数上溢前的时间必须大于等于时钟振荡稳定时间。
3. 在将待机控制寄存器（STBCR1：参照“49. 低功耗模式”）的 STBY 位置 “1” 并且将 DEEP 位置 “0” 后，通过执行 SLEEP 指令转移到软件待机模式，停止时钟的运行。
4. 通过 NMI 信号变化边沿等的中断检测，此模块开始计数。
5. 如果此模块发生计数上溢，时钟脉冲振荡器就开始提供时钟，本 LSI 重新开始运行。此时，WRCSR 的 WOVF 不被置位。

14.4.2 看门狗定时器模式的用法

1. 在将 WTCSR 的 WT/IT 位置 “1” 后，给 CKS[2:0] 设定计数时钟的种类，给 WRCSR 的 RSTE 位设定是否对本 LSI 内部进行复位，给 RSTS 位设定此时的复位类型并且给 WTCNT 设定初始值。
2. 如果将 WTCSR 的 TME 位置 “1”，就在看门狗定时器模式中开始计数。
3. 在看门狗定时器模式中运行时，为了计数器不发生上溢，必须定期地将计数器改写为 “H'00”。
4. 如果计数器发生上溢，此模块就将 WRCSR 的 WOVF 置 “1”，将 WDTOVF 信号输出到外部，如图 14.4 所示。能使用此 WDTOVF 信号对系统进行复位。在 $64 \times P0\phi$ 时钟之间输出 WDTOVF 信号。
5. 如果预先将 WRCSR 的 RSTE 位置 “1”，就能与 WDTOVF 信号同时产生本 LSI 内部的复位信号。能通过设定 WRCSR 的 RSTS 位，选择上电复位或者手动复位。在 $128 \times P0\phi$ 时钟之间输出内部复位信号。
6. 当 RES 引脚的输入信号引起的复位和此模块的上溢引起的复位同时发生时，优先由 RES 引脚引起的复位，并且将 WRCSR 的 WOVF 位清 “0”。

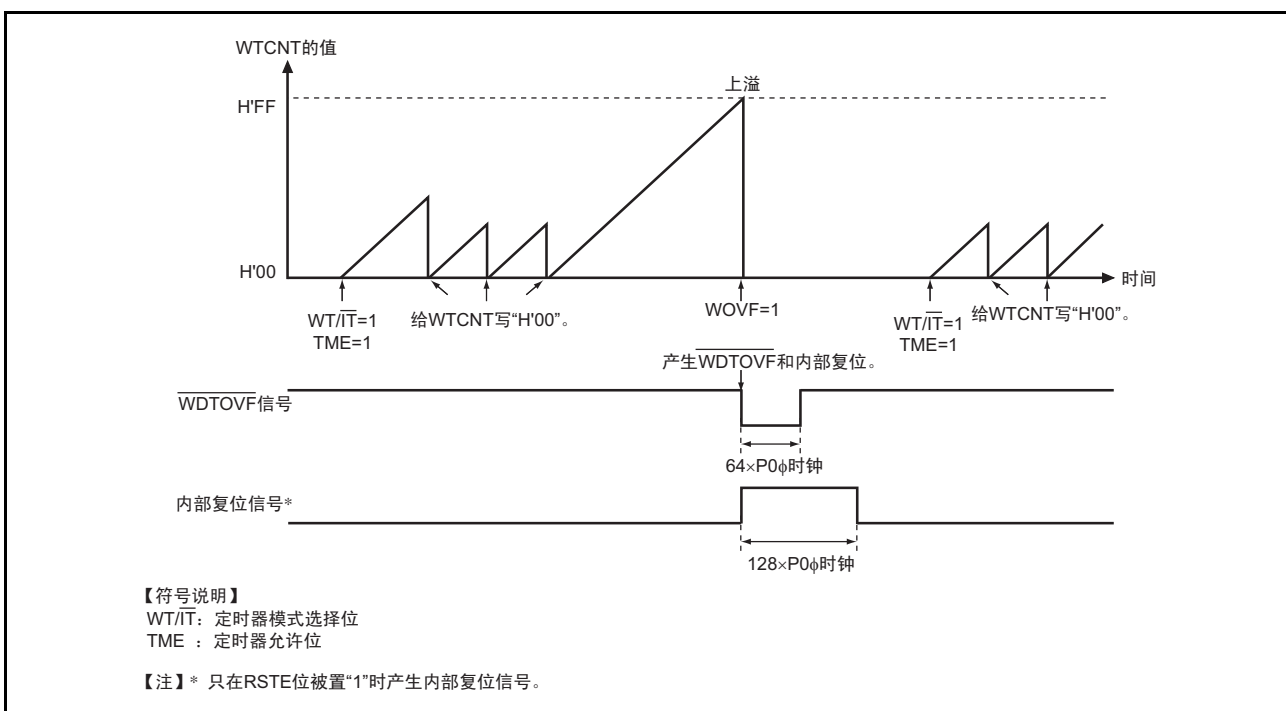


图 14.4 看门狗定时器模式中的运行

14.4.3 间隔定时器模式的用法

在间隔定时器模式中运行时，每当计数器发生上溢时，产生间隔定时器中断。因此能每隔一定时间产生 1 次中断。

1. 在将 WTCSR 的 $\overline{WT/IT}$ 位置 “0” 后，给 CKS[2:0] 位设定计数时钟的种类并且给 WTCNT 设定初始值。
2. 如果将 WTCSR 寄存器的 TME 位置 “1”，就在间隔定时器模式中开始计数。
3. 当计数器发生上溢时，此模块就将 WTCSR 寄存器的 IOVF 标志置 “1”，并且将间隔定时器的中断请求传送到中断控制器，计数器继续计数。

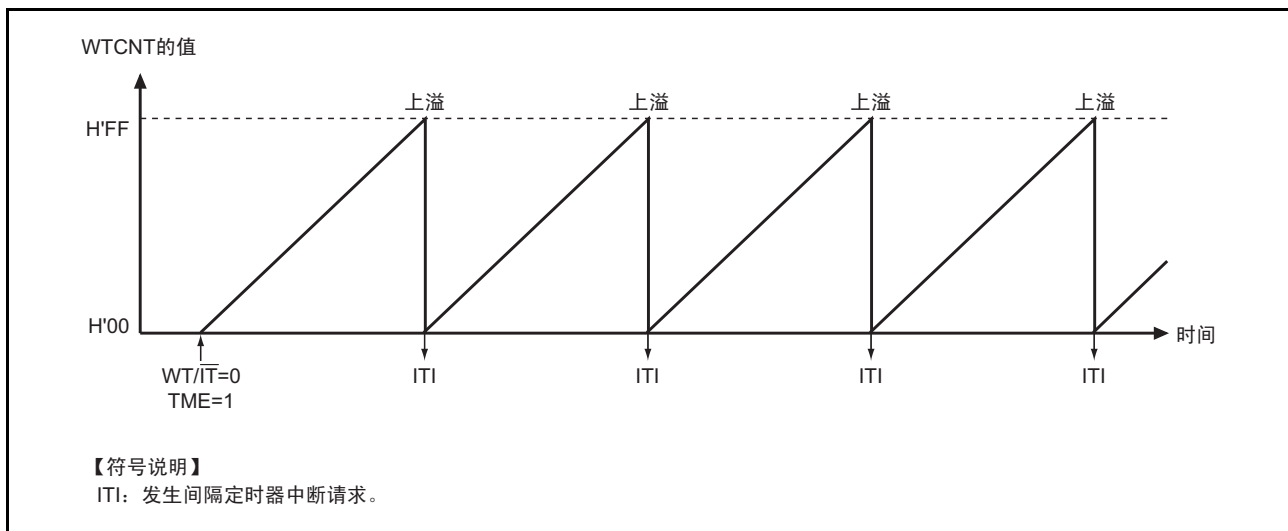


图 14.5 间隔定时器模式中的运行

14.5 使用时的注意事项

在间隔定时器模式和看门狗定时器模式中，有以下注意事项。

14.5.1 定时器的误差

在定时器开始运行后，根据以上电复位为基点的 WTCSR 寄存器的 TME 位的置位时序，WTCNT 寄存器最初的递增计数时序是在从 1 个 $P0\phi$ 周期之后（最短）开始到 CKS[2:0] 所选分频时序（最长）的之间。第 2 次以后的递增计数时序为选择的分频时序。因此，到上述最初的递增计数为止的时间差为定时器的误差。在定时器的运行过程中改写 WTCNT 寄存器后，最初的递增计数时序也一样。

14.5.2 禁止将 H'FF 设定为 WTCNT 的初始值

当 WTCNT 的值变为 “H'FF” 时，此模块判断为发生上溢。因此，如果给 WTCNT 设定 “H'FF”，就立即产生间隔定时器中断或者复位，与 CKS[2:0] 的选择时钟无关。

14.5.3 间隔定时器上溢标志

在 WTCNT 的值为 “H'FF” 时，无法清除 WTCSR 的 IOVF 标志。

必须在 WTCNT 的值变为 “H'00” 或者将 WTCNT 的值改写为 “H'FF” 以外的值后清除 IOVF 标志。

14.5.4 通过 $\overline{\text{WDTOVF}}$ 信号进行的系统复位

如果将 $\overline{\text{WDTOVF}}$ 信号输入到本 LSI 的 $\overline{\text{RES}}$ 引脚，就不能正确地对本 LSI 进行初始化。

不能通过逻辑电路将 $\overline{\text{WDTOVF}}$ 信号输入到本 LSI 的 $\overline{\text{RES}}$ 引脚。要用 $\overline{\text{WDTOVF}}$ 信号对整个系统进行复位时，必须按图 14.6 所示的电路进行。

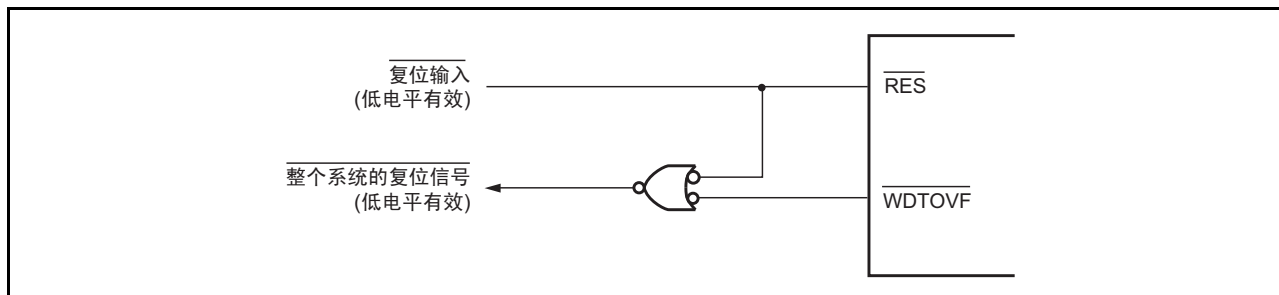


图 14.6 通过 $\overline{\text{WDTOVF}}$ 信号进行系统复位的电路例子

14.5.5 看门狗定时器模式的手动复位

在发生看门狗定时器模式的手动复位时，保持总线周期。如果在总线权处于释放状态中或者在 DMAC 突发传送过程中发生手动复位，就将手动复位异常处理保留到 CPU 获得总线权为止。

15. 实时时钟

本 LSI 内置实时时钟和 32.768kHz 晶体振荡器。

15.1 特点

- 具有时钟和日历功能（BCD表示）
对秒、分钟、小时、星期、日、月、年进行计数。
- 具有1~64Hz定时器（二进制表示）
64Hz计数器表示分频电路中的64Hz~1Hz的状态。
- 开始/停止功能
- 30秒调整功能
- 闹钟中断
闹钟中断条件可选择是否与秒、分钟、小时、星期、日、月或者年进行比较。
- 周期中断
中断周期可选择1/256秒、1/64秒、1/16秒、1/4秒、1/2秒、1秒或者2秒周期。
- 进位中断
这是表示秒计数器发生进位或者在读64Hz计数器时64Hz计数器发生进位的进位中断功能。
- 闰年自动调整功能
- 钟表功能的运行时钟可选择用于内部时钟和钟表专用的外部时钟输入。
- 能通过闹钟中断从深度待机模式返回。

框图如图 15.1 所示。

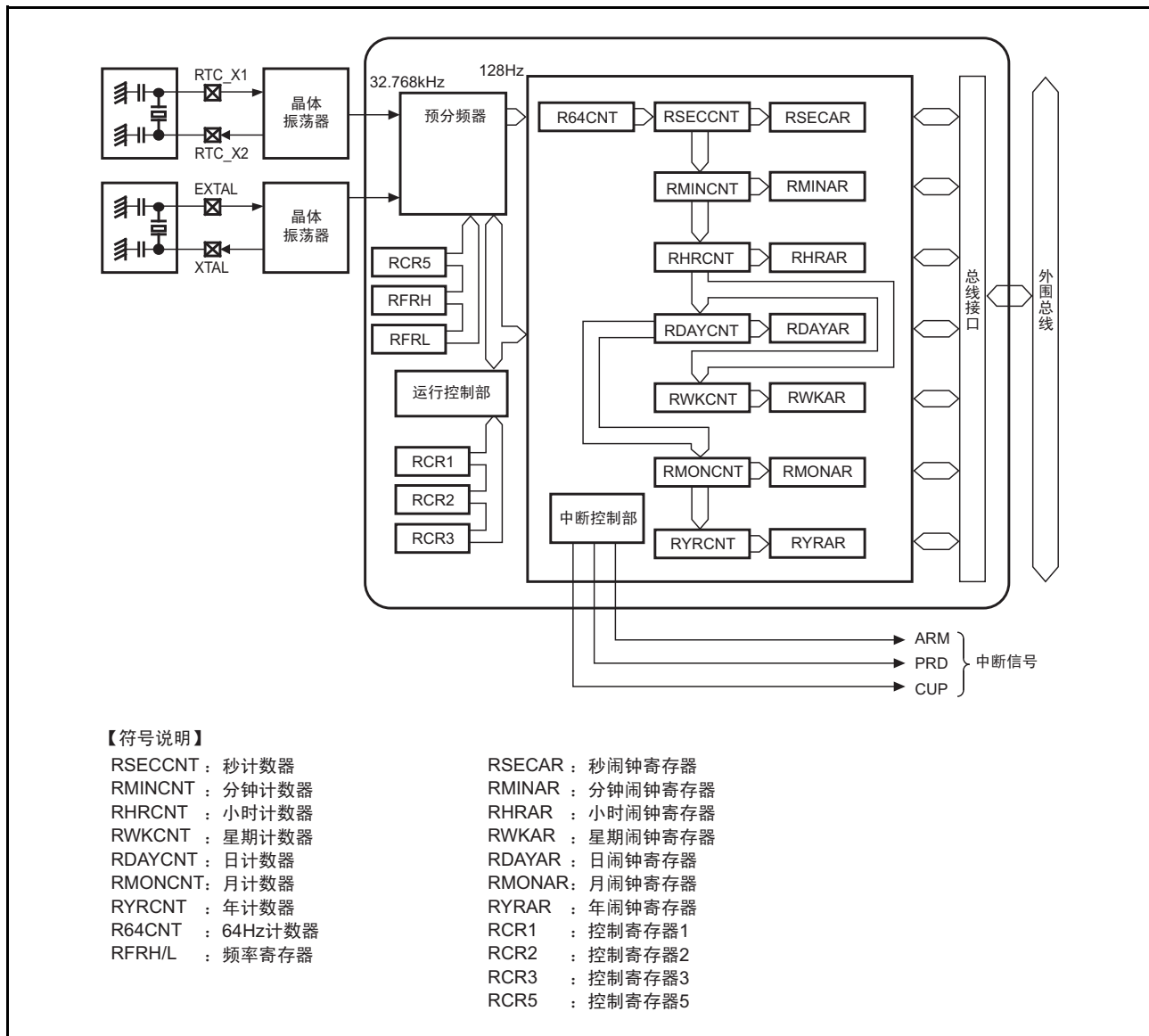


图 15.1 框图

15.2 输入 / 输出引脚

引脚结构如表 15.1 所示。

表 15.1 引脚结构

名称	引脚名	输入 / 输出	功能
用于实时时钟的晶体谐振器 引脚 / 外部时钟	RTC_X1	输入	用于此模块，连接 32.768kHz 的晶体谐振器。 RTC_X1 引脚也能输入外部时钟。
	RTC_X2	输出	
用于内部时钟的晶体谐振器 / 外部时钟	EXTAL	输入	连接用于内部运行的晶体谐振器。 详细内容请参照时钟脉冲振荡器的章节。
	XTAL	输出	

15.3 寄存器说明

寄存器结构如表 15.2 所示。

表 15.2 寄存器结构

寄存器名	略称	R/W	初始值	地址	存取长度
64Hz 计数器	R64CNT	R	H'xx	H'FFFF6000	8
秒计数器	RSECCNT	R/W	H'xx	H'FFFF6002	8
分钟计数器	RMINCNT	R/W	H'xx	H'FFFF6004	8
小时计数器	RHRCNT	R/W	H'xx	H'FFFF6006	8
星期计数器	RWKCNT	R/W	H'0x	H'FFFF6008	8
日计数器	RDAYCNT	R/W	H'xx	H'FFFF600A	8
月计数器	RMONCNT	R/W	H'xx	H'FFFF600C	8
年计数器	RYRCNT	R/W	H'xxxx	H'FFFF600E	16
秒闹钟寄存器	RSECAR	R/W	H'xx	H'FFFF6010	8
分钟闹钟寄存器	RMINAR	R/W	H'xx	H'FFFF6012	8
小时闹钟寄存器	RHRAR	R/W	H'xx	H'FFFF6014	8
星期闹钟寄存器	RWKAR	R/W	H'0x	H'FFFF6016	8
日闹钟寄存器	RDAYAR	R/W	H'xx	H'FFFF6018	8
月闹钟寄存器	RMONAR	R/W	H'xx	H'FFFF601A	8
年闹钟寄存器	RYRAR	R/W	H'xxxx	H'FFFF6020	16
控制寄存器 1	RCR1	R/W	H'xx	H'FFFF601C	8
控制寄存器 2	RCR2	R/W	H'09	H'FFFF601E	8
控制寄存器 3	RCR3	R/W	H'x0	H'FFFF6024	8
控制寄存器 5	RCR5	R/W	H'xx	H'FFFF6026	8
频率寄存器	RFRH	R/W	H'xxxx	H'FFFF602A	16
	RFRL	R/W	H'xxxx	H'FFFF602C	16

15.3.1 64Hz 计数器 (R64CNT)

R64CNT 表示分频电路中的 64Hz ~ 1Hz 的状态。

在 128Hz 分频段发生进位时，如果读此寄存器，就将控制寄存器 1 (RCR1) 的 CF 位置“1”，表示在发生进位的同时读取 64Hz 计数器。此时，读取值无效，因此在给 RCR1 的 CF 位写“0”进行清除后，需要重新读 R64CNT。

如果将控制寄存器 2 (RCR2) 的 RESET 位置“1”或者将 RCR2 的 ADJ 位置“1”，就对分频电路和 R64CNT 进行初始化。

位:	7	6	5	4	3	2	1	0
	—	1Hz	2Hz	4Hz	8Hz	16Hz	32Hz	64Hz
初始值:	0	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7	—	0	R	保留位 读写值都为“0”。
6	1Hz	不定值	R	这些位表示分频电路中的 1Hz ~ 64Hz 的状态。
5	2Hz	不定值	R	
4	4Hz	不定值	R	
3	8Hz	不定值	R	
2	16Hz	不定值	R	
1	32Hz	不定值	R	
0	64Hz	不定值	R	

15.3.2 秒计数器 (RSECCNT)

RSECCNT 是对秒的部分 (BCD 码) 进行设定和计数的计数器，通过 64Hz 计数器按秒的进位而进行计数。

能设定的范围为 10 进制 (BCD) 的 00 ~ 59，否则无法正常运行。另外，必须在通过 RCR2 的 START 位停止计数后进行写处理。

位:	7	6	5	4	3	2	1	0
	—	10秒			1秒			
初始值:	0	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	10 秒	不定值	R/W	秒的十位计数 秒的十位进行 0 ~ 5 的计数，是 60 秒计数。
3 ~ 0	1 秒	不定值	R/W	秒的个位计数 秒的个位按秒进行 0 ~ 9 的计数。如果发生进位，秒的十位就加 1。

15.3.3 分钟计数器 (RMINCNT)

RMINCNT 是对分钟的部分 (BCD 码) 进行设定和计数的计数器, 通过秒计数器按分钟的进位而进行计数。

能设定的范围为 10 进制 (BCD) 的 00 ~ 59, 否则无法正常运行。另外, 必须在通过 RCR2 的 START 位停止计数后进行写处理。

位:	7	6	5	4	3	2	1	0
	—	10分钟			1分钟			
初始值:	0	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	10 分钟	不定值	R/W	分钟的十位计数 分钟的十位进行 0 ~ 5 的计数, 是 60 分钟计数。
3 ~ 0	1 分钟	不定值	R/W	分钟个位计数 分钟个位按分钟进行 0 ~ 9 的计数。如果发生进位, 分钟的十位就加 1。

15.3.4 小时计数器 (RHRCNT)

RHRCNT 是对小时的部分 (BCD 码) 进行设定和计数的计数器, 通过分钟计数器按小时的进位而进行计数。

能设定的范围为 10 进制 (BCD) 的 00 ~ 23, 否则无法正常运行。另外, 必须在通过 RCR2 的 START 位停止计数后进行写处理。

位:	7	6	5	4	3	2	1	0
	—	—	10小时		1小时			
初始值:	0	0	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	10 小时	不定值	R/W	小时的十位计数 小时的十位进行 0 ~ 2 的计数。
3 ~ 0	1 小时	不定值	R/W	小时的个位计数 小时的个位按小时进行 0 ~ 9 的计数。如果发生进位, 小时的十位就加 1。

15.3.5 星期计数器 (RWKCNT)

RWKCNT 是对星期的部分 (BCD 码) 进行设定和计数的计数器, 通过小时计数器按日的进位而进行计数。

能设定的范围为 10 进制 (BCD) 的 0 ~ 6, 否则无法正常运行。另外, 必须在通过 RCR2 的 START 位停止计数后进行写处理。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	星期几		
初始值:	0	0	0	0	0	不定值	不定值	不定值
R/W:	R	R	R	R	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 3	—	全 0	R	保留位 读写值都为“0”。
2 ~ 0	星期	不定值	R/W	星期的计数 用二进制码表示星期。 000: 星期日 001: 星期一 010: 星期二 011: 星期三 100: 星期四 101: 星期五 110: 星期六 111: 保留 (禁止设定)

15.3.6 日计数器 (RDAYCNT)

RDAYCNT 是对日的部分 (BCD 码) 进行设定和计数的计数器, 通过小时计数器按日的进位而进行计数。

能设定的范围为 10 进制 (BCD) 的 01 ~ 31, 否则无法正常运行。另外, 必须在通过 RCR2 的 START 位停止计数后进行写处理。

因为 RDAYCNT 的设定范围因月和闰年而不同, 所以必须在确认后进行设定。闰年是将年计数器 (RYRCNT) 作为西历, 根据是否能被 400、100 或者 4 整除而得来的。

位:	7	6	5	4	3	2	1	0
	—	—	10日		1日			
初始值:	0	0	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	10 日	不定值	R/W	日的十位计数
3 ~ 0	1 日	不定值	R/W	日的个位计数 日的个位按日进行 0 ~ 9 的计数。如果发生进位, 日的十位就加 1。

15.3.7 月计数器 (RMONCNT)

RMONCNT 是对月的部分 (BCD 码) 进行设定和计数的计数器, 通过日计数器按月的进位而进行计数。

能设定的范围为 10 进制 (BCD) 的 01 ~ 12, 否则无法正常运行。另外, 必须在通过 RCR2 的 START 位停止计数后进行写处理。

位:	7	6	5	4	3	2	1	0
	—	—	—	10月	1月			
初始值:	0	0	0	不定值	不定值	不定值	不定值	不定值
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	10 月	不定值	R/W	月的十位计数
3 ~ 0	1 月	不定值	R/W	月的个位计数 月的个位按月进行 0 ~ 9 的计数。如果发生进位, 月的十位就加 1。

15.3.8 年计数器 (RYRCNT)

RYRCNT 是对年的部分 (BCD 码) 进行设定和计数的计数器, 通过月计数器按年的进位而进行计数。

能设定的范围为 10 进制 (BCD) 的 0000 ~ 9999, 否则无法正常运行。另外, 必须在通过 RCR2 的 START 位停止计数后进行写处理。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	1000年				100年				10年				1年			
初始值:	不定值 不定值 不定值 不定值 不定值 不定值 不定值 不定值 不定值 不定值 不定值 不定值 不定值 不定值 不定值 不定值 不定值															
R/W:	R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W R/W															

位	位名	初始值	R/W	说明
15 ~ 12	1000 年	不定值	R/W	年的千位计数
11 ~ 8	100 年	不定值	R/W	年的百位计数
7 ~ 4	10 年	不定值	R/W	年的十位计数
3 ~ 0	1 年	不定值	R/W	年的个位计数

15.3.9 秒闹钟寄存器 (RSECAR)

RSECAR 是与秒部分 (BCD 码) 的计数器 RSECCNT 对应的闹钟寄存器。如果将 ENB 位置 “1”，就和 RSECCNT 的值进行比较。只对 ENB 位为 “1” 的闹钟寄存器 (RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RCR3) 进行计数器和闹钟寄存器的比较，当各自的值全部相同时，将 RCR1 的闹钟标志置 “1”。

能设定的范围为 10 进制 (BCD) 的 00 ~ 59+ENB 位，否则无法正常运行。

位:	7	6	5	4	3	2	1	0
	ENB	10秒			1秒			
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	ENB	不定值	R/W	如果将此位置 “1”，就和 RSECCNT 的值进行比较。
6 ~ 4	10 秒	不定值	R/W	秒的十位设定值
3 ~ 0	1 秒	不定值	R/W	秒的个位设定值

15.3.10 分钟闹钟寄存器 (RMINAR)

RMINAR 是与分钟部分 (BCD 码) 的计数器 RMINCNT 对应的闹钟寄存器。如果将 ENB 位置 “1”，就和 RMINCNT 的值进行比较。只对 ENB 位为 “1” 的闹钟寄存器 (RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RCR3) 进行计数器和闹钟寄存器的比较，当各自的值全部相同时，将 RCR1 的闹钟标志置 “1”。

能设定的范围为 10 进制 (BCD) 的 00 ~ 59+ENB 位，否则无法正常运行。

位:	7	6	5	4	3	2	1	0
	ENB	10分钟			1分钟			
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	ENB	不定值	R/W	如果将此位置 “1”，就和 RMINCNT 的值进行比较。
6 ~ 4	10 分钟	不定值	R/W	分钟的十位设定值
3 ~ 0	1 分钟	不定值	R/W	分钟的个位设定值

15.3.11 小时闹钟寄存器 (RHRAR)

RHRAR 是与小时部分 (BCD 码) 的计数器 RHRCNT 对应的闹钟寄存器。如果将 ENB 位置 “1”，就和 RHRCNT 的值进行比较。只对 ENB 位为 “1” 的闹钟寄存器 (RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RCR3) 进行计数器和闹钟寄存器的比较，当各自的值全部相同时，将 RCR1 的闹钟标志置 “1”。

能设定的范围为 10 进制 (BCD) 的 00 ~ 23+ENB 位，否则无法正常运行。

位:	7	6	5	4	3	2	1	0
	ENB	—	10小时		1小时			
初始值:	不定值	0	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	ENB	不定值	R/W	如果将此位置 “1”，就和 RHRCNT 的值进行比较。
6	—	0	R	保留位 读写值都为 “0”。
5、4	10 小时	不定值	R/W	小时的十位设定值
3 ~ 0	1 小时	不定值	R/W	小时的个位设定值

15.3.12 星期闹钟寄存器 (RWKAR)

RWKAR 是与星期部分 (BCD 码) 的计数器 RWKCNT 对应的闹钟寄存器。如果将 ENB 位置 “1”，就和 RWKCNT 的值进行比较。只对 ENB 位为 “1” 的闹钟寄存器 (RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RCR3) 进行计数器和闹钟寄存器的比较，当各自的值全部相同时，将 RCR1 的闹钟标志置 “1”。

能设定的范围为 10 进制 (BCD) 的 0 ~ 6+ENB 位，否则无法正常运行。

位:	7	6	5	4	3	2	1	0
	ENB	—	—	—	—	星期		
初始值:	不定值	0	0	0	0	不定值	不定值	不定值
R/W:	R/W	R	R	R	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	ENB	不定值	R/W	如果将此位置 “1”，就和 RWKCNT 的值进行比较。
6 ~ 3	—	全 0	R	保留位 读写值都为 “0”。
2 ~ 0	星期	不定值	R/W	星期的设定值 000: 星期日 001: 星期一 010: 星期二 011: 星期三 100: 星期四 101: 星期五 110: 星期六 111: 保留 (禁止设定)

15.3.13 日闹钟寄存器 (RDAYAR)

RDAYAR 是与日部分 (BCD 码) 的计数器 RDAYCNT 对应的闹钟寄存器。如果将 ENB 位置 “1”，就和 RDAYCNT 的值进行比较。只对 ENB 位为 “1” 的闹钟寄存器 (RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RCR3) 进行计数器和闹钟寄存器的比较，当各自的值全部相同时，将 RCR1 的闹钟标志置 “1”。

能设定的范围为 10 进制 (BCD) 的 00 ~ 31+ENB 位，否则无法正常运行。

位:	7	6	5	4	3	2	1	0
	ENB	—	10日		1日			
初始值:	不定值	0	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	ENB	不定值	R/W	如果将此位置 “1”，就和 RDAYCNT 的值进行比较。
6	—	0	R	保留位 读写值都为 “0”。
5、4	10 日	不定值	R/W	日的十位设定值
3 ~ 0	1 日	不定值	R/W	日的个位设定值

15.3.14 月闹钟寄存器 (RMONAR)

RMONAR 是与月部分 (BCD 码) 的计数器 RMONCNT 对应的闹钟寄存器。如果将 ENB 位置 “1”，就和 RMONCNT 的值进行比较。只对 ENB 位为 “1” 的闹钟寄存器 (RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RCR3) 进行计数器和闹钟寄存器的比较，当各自的值全部相同时，将 RCR1 的闹钟标志置 “1”。

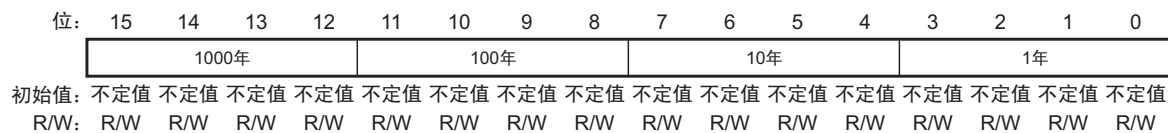
能设定的范围为 10 进制 (BCD) 的 01 ~ 12+ENB 位，否则无法正常运行。

位:	7	6	5	4	3	2	1	0
	ENB	—	—	10月	1月			
初始值:	不定值	0	0	不定值	不定值	不定值	不定值	不定值
R/W:	R/W	R	R	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	ENB	不定值	R/W	如果将此位置 “1”，就和 RMONCNT 的值进行比较。
6、5	—	全 0	R	保留位 读写值都为 “0”。
4	10 月	不定值	R/W	月的十位设定值
3 ~ 0	1 月	不定值	R/W	月的个位设定值

15.3.15 年闹钟寄存器 (RYRAR)

RYRAR 是与年部分 (BCD 码) 的计数器 RYRCNT 对应的闹钟寄存器。
能设定的范围为 10 进制 (BCD) 的 0000 ~ 9999, 否则无法正常运行。



位	位名	初始值	R/W	说明
15 ~ 12	1000 年	不定值	R/W	年的千位设定值
11 ~ 8	100 年	不定值	R/W	年的百位设定值
7 ~ 4	10 年	不定值	R/W	年的十位设定值
3 ~ 0	1 年	不定值	R/W	年的个位设定值

15.3.16 控制寄存器 1 (RCR1)

RCR1 是与进位和闹钟标志有关的寄存器。对于各标志，能选择是否产生中断。

在分频电路被复位（将 RCR2 的 RESET 位和 ADJ 位置“1”）前，CF 标志为不定值。要使用 CF 标志时，必须在使用前对分频电路进行复位。

在设定闹钟寄存器和计数器的值前，AF 标志为不定值。要使用 AF 标志时，必须在使用前设定闹钟寄存器和计数器。

位:	7	6	5	4	3	2	1	0
	CF	—	—	CIE	AIE	—	—	AF
初始值:	不定值	0	0	0	0	0	0	不定值
R/W:	R/W	R	R	R/W	R/W	R	R	R/W

位	位名	初始值	R/W	说明
7	CF	不定值	R/W	进位标志 当此标志被置“1”时，表示秒计数器发生进位或者在读 64Hz 计数器时 64Hz 计数器发生进位，此时不能保证读取的计数寄存器的值。需要重新读取。 0: 秒计数器没有发生进位并且在读 64Hz 计数器时 64Hz 计数器也没有发生进位 [清除条件] 当给 CF 写“0”时 1: 秒计数器发生进位或者在读 64Hz 计数器时 64Hz 计数器发生进位 [置位条件] 秒计数器发生进位或者在读 64Hz 计数器时 64Hz 计数器发生进位，或者给 CF 写“1”时
6、5	—	全 0	R	保留位 读写值都为“0”。
4	CIE	0	R/W	进位中断允许标志 此位允许在进位标志（CF）被置“1”时产生中断。 0: 不允许在 CF 标志被置“1”时产生进位中断。 1: 允许在 CF 标志被置“1”时产生进位中断。
3	AIE	0	R/W	闹钟中断允许标志 此位允许在闹钟标志（AF）被置“1”时产生中断。 0: 不允许在 AF 标志被置“1”时产生闹钟中断。 1: 允许在 AF 标志被置“1”时产生闹钟中断。
2、1	—	全 0	R	保留位 读写值都为“0”。
0	AF	不定值	R/W	闹钟标志 在闹钟寄存器（RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RYRAR）设定的闹钟时间（只限于 ENB 位为“1”的寄存器）和计数器的值相同时，将此标志置“1”。 0: 闹钟寄存器和计数器不相同 [清除条件] 当给 AF 写“0”时 1: 闹钟寄存器和计数器相同* [置位条件] 当闹钟寄存器（只限于 ENB 位为“1”的寄存器）与计数器相同时 【注】 * 当写“1”时，保持原来的值。

15.3.17 控制寄存器 2 (RCR2)

RCR2 是与周期中断控制、30 秒调整、分频电路复位和计数控制有关的寄存器。

在上电复位时或者在深度待机模式中，对 RCR2 进行初始化。在手动复位时，对 RTCEN 位和 START 位以外的位进行初始化。

位:	7	6	5	4	3	2	1	0
	PEF	PES[2:0]			RTCEN	ADJ	RESET	START
初始值:	0	0	0	0	1	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	PEF	0	R/W	周期中断标志 此标志表示在 PES2 ~ PES0 位设定的周期中发生中断。如果此标志被置“1”，就产生周期中断。 0: 在 PES2 ~ PES0 位设定的周期中未产生中断 [清除条件] 当给 PEF 写“0”时 1: 在 PES2 ~ PES0 位设定的周期中产生中断 [置位条件] 在 PES2 ~ PES0 位设定的周期中产生中断时或者给 PEF 写“1”时
6 ~ 4	PES[2:0]	000	R/W	中断允许标志 设定周期中断的周期。 000: 没有产生周期中断 001: 禁止设定 010: 产生周期中断的周期为每 1/64 秒 011: 产生周期中断的周期为每 1/16 秒 100: 产生周期中断的周期为每 1/4 秒 101: 产生周期中断的周期为每 1/2 秒 110: 产生周期中断的周期为每 1 秒 111: 产生周期中断的周期为每 2 秒
3	RTCEN	1	R/W	RTC_X1 时钟控制 控制 RTC_X1 引脚的功能。 0: 停止内部晶体振荡器的振荡 / 禁止外部时钟的输入 1: 开始内部晶体振荡器的振荡 / 允许外部时钟的输入
2	ADJ	0	R/W	30 秒调整 用于 30 秒调整。通过给此位写“1”，将 29 秒以前的时间舍为 00 秒，而将 30 秒以后的时间进行 1 分钟的进位。此时，同时对分频电路（预分频器和 R64CNT）进行复位。因此 ADJ 位自动变为“0”，所以不需要特意写“0”，此位的读取值总是“0”。 0: 正常的时钟运行 1: 进行 30 秒的调整
1	RESET	0	R/W	复位 通过给此位写“1”，对分频电路、R64CNT 寄存器、闹钟寄存器、RCR3 寄存器、RCR1 寄存器的 CF 位和 AF 位、RCR2 寄存器的 PEF 位进行初始化。另外，当给此位写“1”时，就在对以上寄存器进行复位后，此 RESET 位自动变为“0”，因此不需要特意写“0”，此位的读取值总是“0”。 0: 正常的时钟运行 1: 对分频电路进行复位

位	位名	初始值	R/W	说明
0	START	1	R/W	START 位 此位停止并且重新启动计数器（时钟）的运行。 0: 秒、分钟、小时、日、星期、月、年计数器停止计数 1: 秒、分钟、小时、日、星期、月、年计数器正常计数

15.3.18 控制寄存器 3（RCR3）

如果将 ENB 位置“1”，RCR3 就和 RYRCNT 的值进行比较。只对 ENB 位为“1”的闹钟寄存器（RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RCR3）进行计数器和闹钟寄存器的比较，当各自的值全部相同时，将 RCR1 的闹钟标志置“1”。

位:	7	6	5	4	3	2	1	0
	ENB	—	—	—	—	—	—	—
初始值:	不定值	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7	ENB	不定值	R/W	如果将此位置“1”，就和 RYRCNT 的值进行比较。
6~0	—	全 0	R	保留位 读写值都为“0”。

15.3.19 控制寄存器 5（RCR5）

如果将 RCR5 的 RCKSEL[1:0] 位置“00”，就使用 RTC_X1 时钟进行钟表的计数；如果将 RCR5 的 RCKSEL[1:0] 位置“01”，就使用 EXTAL 时钟进行钟表的计数。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RCKSEL[1:0]	
初始值:	0	0	0	0	0	0	不定值	不定值
R/W:	R	R	R	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明
7~2	—	全 0	R	保留位 读写值都为“0”。
1、0	RCKSEL[1:0]	不定值	R/W	运行时钟选择位 能选择 RTC_X1 或者 EXTAL 为运行时钟。不能在运行过程中进行 RCKSEL[1:0] 的切换。 00: 选择 RTC_X1 01: 选择 EXTAL 10: 禁止设定 11: 禁止设定

15.3.20 频率寄存器 H/L (RFRH/L)

RFRH/L 是可读写寄存器，由 16 位构成。

当通过 EXTAL 时钟运行时，为了生成 128Hz 时钟，给 RFC[18:0] 设定“频率比较值”。必须根据 EXTAL 时钟频率更改“频率比较值”。计算方法请参照下述内容。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SEL64	—	—	—	—	—	—	—	—	—	—	—	—	RFC[18:16]		
初始值:	不定值	0	0	0	0	0	0	0	0	0	0	0	0	不定值	不定值	不定值
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFC[15:0]															
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31	SEL64	不定值	R/W	64 分频选择 表示 EXTAL 时钟频率不能被 128Hz 整除而能被 64Hz 整除时的运行时钟。 0: EXTAL 时钟频率能被 128Hz 整除 1: EXTAL 时钟频率不能被 128Hz 整除而能被 64Hz 整除
30 ~ 19	—	全 0	R	保留位 读写值都为“0”。
18 ~ 0	RFC[18:0]	不定值	R/W	频率比较值 为了从 EXTAL 生成运行时钟，设定比较值。

(1) “频率比较值”的计算方法

- EXTAL 时钟频率能被 128Hz 整除的情况

$$RFC[18:0] = (\text{EXTAL 时钟频率}) \div 128$$
 必须将 SEL64 位置“0”。
- EXTAL 时钟频率不能被 128Hz 整除而能被 64Hz 整除的情况

$$RFC[18:0] = (\text{EXTAL 时钟频率}) \div 64$$
 必须将 SEL64 位置“1”。

(2) 设定例子

表 15.3 设定例子

EXTAL	时钟频率	SEL64 的设定值	RFC 的设定值
	10MHZ	0	H'1312D
	11MHZ	1	H'29F63
	12MHZ	0	H'16E36
	13MHZ	1	H'31975

15.4 运行说明

说明此模块的使用例子。

15.4.1 接通电源后的寄存器的初始设定

必须在接通电源后对全部寄存器进行初始设定。

15.4.2 时间的设定步骤

时间的设定步骤例子如图 15.2 所示。

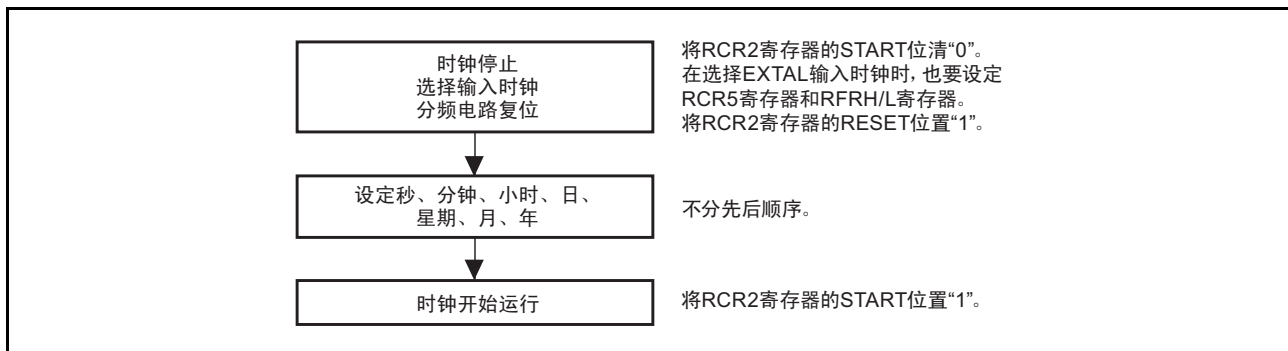


图 15.2 时间的设定步骤

15.4.3 时间的读取步骤

时间的读取步骤如图 15.3 所示。

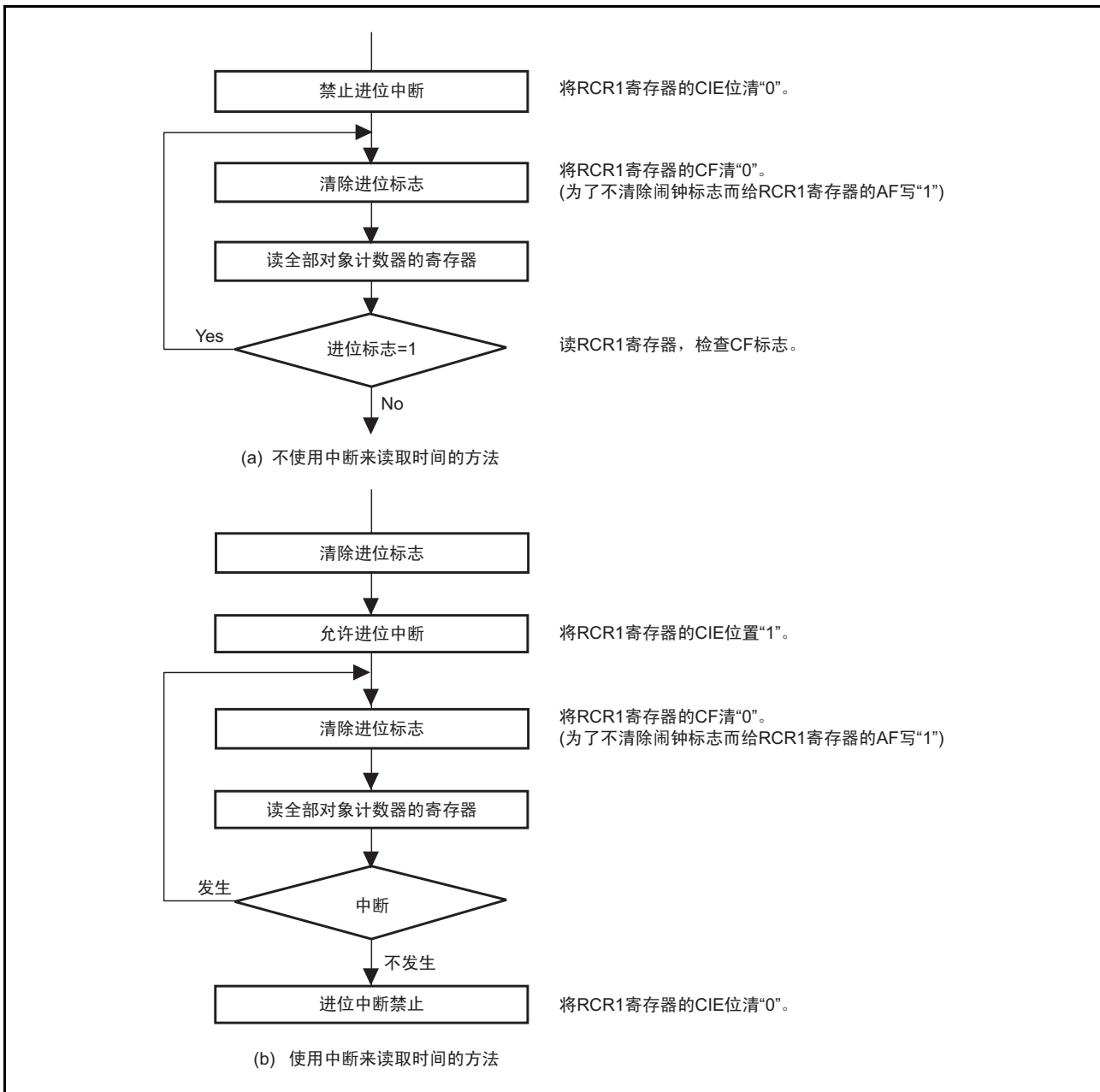


图 15.3 时间的读取步骤

如果在读时间的过程中发生进位，就得不到正确的时间，需要重新读取。不使用中断的方法如图 15.3(a) 所示，使用进位中断的方法如图 15.3(b) 所示。通常，为了容易编写程序，利用不使用中断的方法。

15.4.4 闹钟功能

闹钟功能的使用例子如图 15.4 所示。

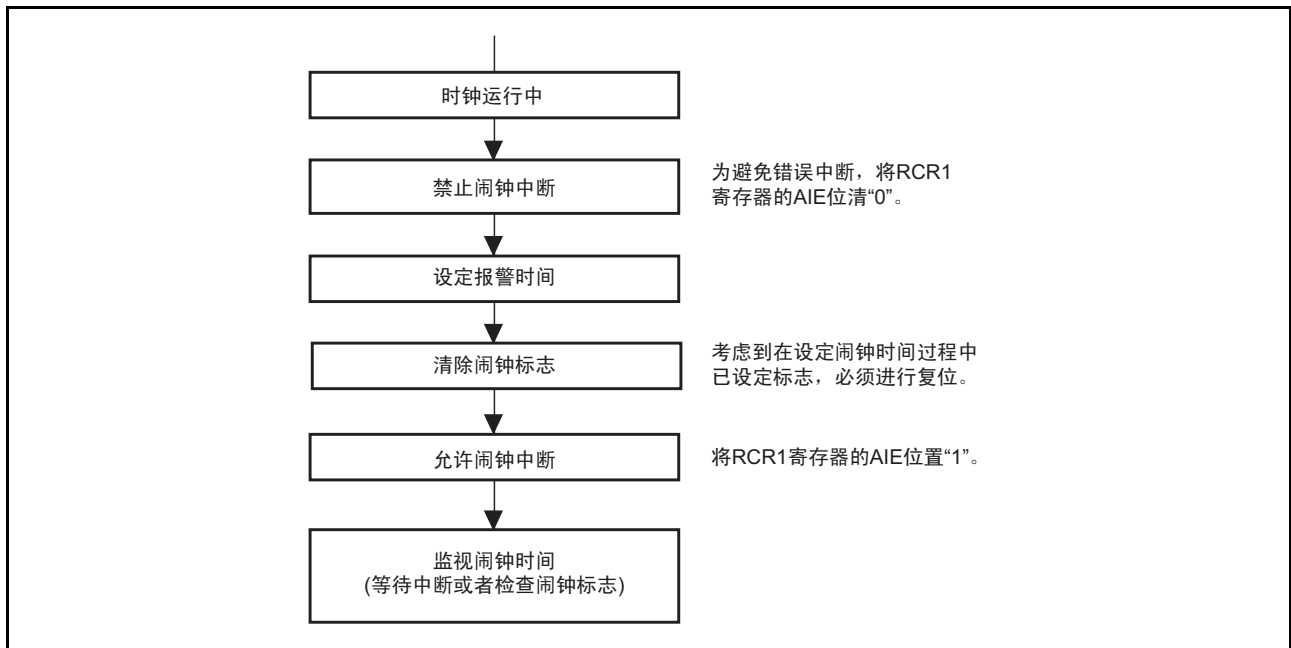


图 15.4 闹钟功能的使用方法

能通过秒、分钟、小时、日、星期、月、年中任意一个或者组合产生闹钟。给闹钟对象的闹钟寄存器的 ENB 位写 “1”，并且给低位设定闹钟时间。给闹钟对象外的寄存器的 ENB 位写 “0”。

如果计数器的值和闹钟时间相同，就将 RCR1 寄存器的 AF 位置 “1”。能通过读此位确认闹钟的检测，但是一般通过中断进行。如果给 RCR1 寄存器的 AIE 位写 “1”，就能产生闹钟中断并且能检测到闹钟。

如果计数器的值和闹钟时间相同，就将闹钟标志置位。但是，如果给闹钟标志写 “0”，就清除闹钟标志。

15.5 使用时的注意事项

15.5.1 有关计数过程中的寄存器的写操作

在计数过程中（RCR2 寄存器的 START 位为“1”），不能写以下寄存器：

RSECCNT、RMINCNT、RHRCNT、RDAYCNT、RWKCNT、RMONCNT、RYRCONT

要写上述寄存器时，必须暂停计数，然后写寄存器。

15.5.2 有关实时时钟周期中断的使用

周期中断的使用方法如图 15.5 所示。

能在 RCR2 寄存器的 PES[2:0] 位设定的周期中定期地产生周期中断。在经过 PES[2:0] 位设定的时间后将 PEF 置“1”。

在设定 PES[2:0] 位以及发生周期中断时，将 PEF 清“0”。虽然能通过读此位确认周期中断的发生，但是一般使用中断功能。

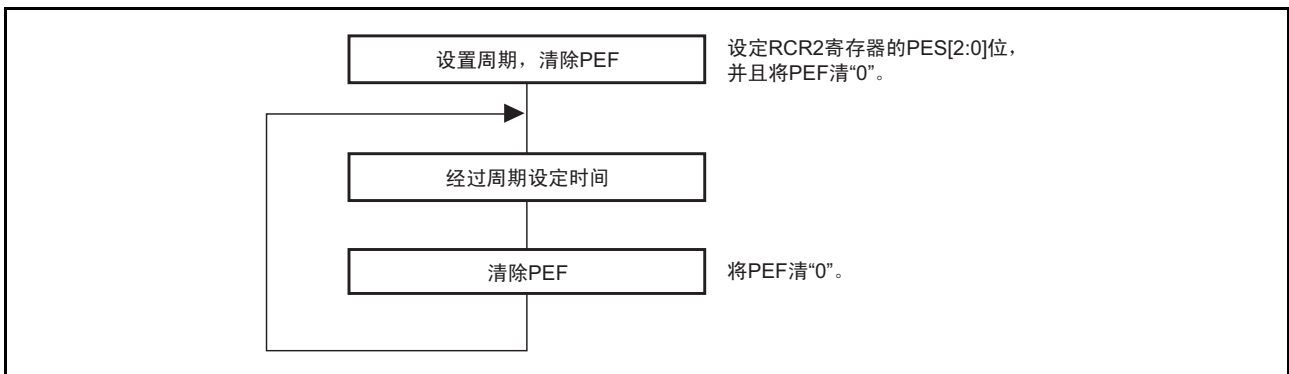


图 15.5 周期中断功能的使用方法

15.5.3 有关寄存器设定后的待机转移

如果在设定此模块内的寄存器后转移到待机状态，就可能无法正确计数。必须在设定寄存器后虚读所设寄存器，然后转移到待机状态。

15.5.4 读写寄存器时的注意事项

- 要在写秒计数器等计数寄存器和 RCR2 寄存器后读这些寄存器时，必须在进行 2 次虚读后读这些寄存器。2 次的虚读值为写之前的值，第 3 次的读取值为写入值。
- 在写上述以外的寄存器后立即读取时，读取值为写入值。

16. FIFO 内置型串行通信接口

本 LSI 有 8 个通道的 FIFO 内置型串行通信接口，支持异步通信和时钟同步通信 2 种方式。各通道还内置分别用于发送和接收的 16 段 FIFO 寄存器，能提高本 LSI 的效率以及实现高速的连续通信。

16.1 特点

- 异步串行通信
通过以字符为单位取得同步的异步方式进行串行数据通信。能与 Universal Asynchronous Receiver/Transmitter (UART) 或者 Asynchronous Communication Interface Adapter (ACIA) 等标准异步通信的 LSI 进行串行数据通信，能从 8 种格式中选择串行数据的通信格式。
数据长度 : 7 位或者 8 位
停止位长 : 1 位或者 2 位
奇偶校验 : 偶校验、奇校验或者无奇偶校验
接收错误的检测 : 检测奇偶校验错误、帧错误和溢出错误。
中止的检测 : 当发生帧错误并且继续有至少 1 帧长的空间 0 (低电平) 时，就检测到中止；当发生帧错误时，能通过直接从串行端口寄存器读取 RxD 引脚的电平来检测到中止。
- 时钟同步串行通信
与时钟同步进行串行数据通信，能和具有时钟同步通信功能的其他 LSI 进行串行数据通信。串行数据的通信格式只有 1 种。
数据长度 : 8 位
接收错误的检测 : 检测溢出错误。
- 能进行全双工通信。
有独立的发送部和接收部，因此能同时进行发送和接收，而且发送部和接收部都是 16 段 FIFO 缓冲结构，所以能高速连续发送和连续接收串行数据。
- 能通过内部波特率发生器选择任意的位速率。
- 内部或者外部的发送/接收时钟源
可选择波特率发生器 (内部时钟) 或者 SCK 引脚 (外部时钟)。
- 4 种中断源
中断源有发送 FIFO 数据空、中止、接收 FIFO 数据满和接收错误共 4 种中断，能分别独立请求中断。
- 在不使用此模块时，为了降低功耗，能停止给此模块提供时钟，使其停止运行。
- 在异步模式中，内置 (只限于通道 1) 调制解调器控制功能 (RTS 和 CTS)。
- 能检测到发送/接收 FIFO 数据寄存器的数据量以及接收 FIFO 数据寄存器的接收数据的接收错误数。
- 在异步模式接收时，能检测到超时错误 (DR)。
- 在异步模式中，可选择 16 倍或者 8 倍位速率的基本时钟运行。
- 在异步模式中，当时钟源为内部时钟并且 SCK 引脚为输入引脚时，能选择波特率发生器的正常模式或者倍速模式。

框图如图 16.1 所示。

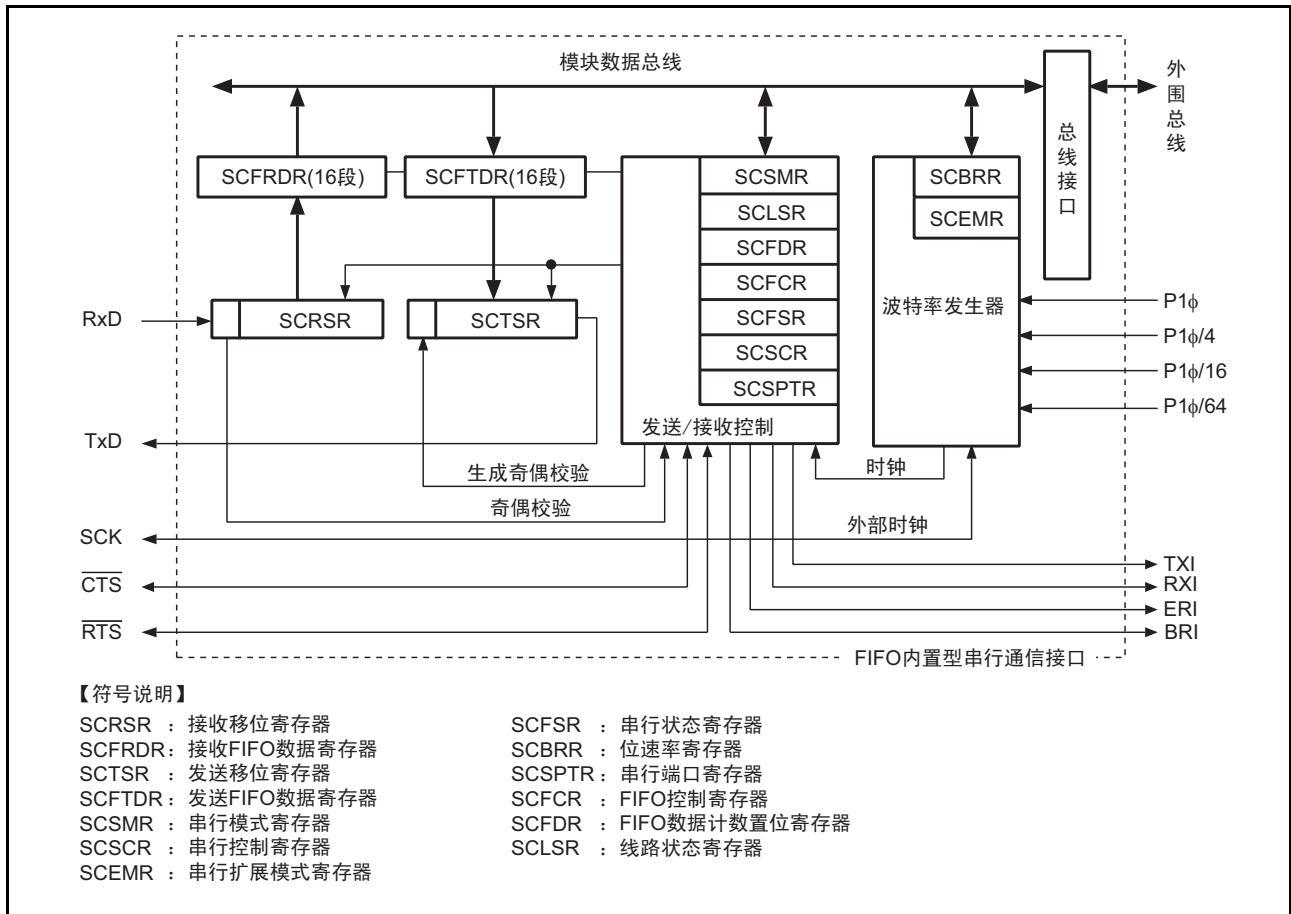


图 16.1 框图

16.2 输入 / 输出引脚

引脚构成如表 16.1 所示。

表 16.1 引脚构成

通道	名称	引脚名称	输入 / 输出	功能
0 ~ 7	串行时钟引脚	SCK0 ~ SCK7	输入 / 输出	时钟的输入 / 输出
	接收数据引脚	RxD0 ~ RxD7	输入	接收数据的输入
	发送数据引脚	TxD0 ~ TxD7	输出	发送数据的输出
1、5、7	请求发送引脚	$\overline{\text{RTS}}1$ 、 $\overline{\text{RTS}}5^*$ 、 $\overline{\text{RTS}}7^*$	输入 / 输出	请求发送
	清除发送引脚	$\overline{\text{CTS}}1$ 、 $\overline{\text{CTS}}5^*$ 、 $\overline{\text{CTS}}7^*$	输入 / 输出	清除发送

【注】 * 在 SH7268 中，不能使用 $\overline{\text{RTS}}5$ 引脚、 $\overline{\text{CTS}}5$ 引脚、 $\overline{\text{RTS}}7$ 引脚和 $\overline{\text{CTS}}7$ 引脚。

16.3 寄存器说明

寄存器结构如表 16.2 所示。

表 16.2 寄存器构成

通道	寄存器名	略称	R/W	初始值	地址	存取长度
0	串行模式寄存器_0	SCSMR_0	R/W	H'0000	H'E8007000	16
	位速率寄存器_0	SCBRR_0	R/W	H'FF	H'E8007004	8
	串行控制寄存器_0	SCSCR_0	R/W	H'0000	H'E8007008	16
	发送 FIFO 数据寄存器_0	SCFTDR_0	W	不定值	H'E800700C	8
	串行状态寄存器_0	SCFSR_0	R/(W)*1	H'0060	H'E8007010	16
	接收 FIFO 数据寄存器_0	SCFRDR_0	R	不定值	H'E8007014	8
	FIFO 控制寄存器_0	SCFCR_0	R/W	H'0000	H'E8007018	16
	FIFO 数据计数置位寄存器_0	SCFDR_0	R	H'0000	H'E800701C	16
	串行端口寄存器_0	SCSPTR_0	R/W	H'0050	H'E8007020	16
	线路状态寄存器_0	SCLSR_0	R/(W)*2	H'0000	H'E8007024	16
	串行扩展模式寄存器_0	SCEMR_0	R/W	H'0000	H'E8007028	16
1	串行模式寄存器_1	SCSMR_1	R/W	H'0000	H'E8007800	16
	位速率寄存器_1	SCBRR_1	R/W	H'FF	H'E8007804	8
	串行控制寄存器_1	SCSCR_1	R/W	H'0000	H'E8007808	16
	发送 FIFO 数据寄存器_1	SCFTDR_1	W	不定值	H'E800780C	8
	串行状态寄存器_1	SCFSR_1	R/(W)*1	H'0060	H'E8007810	16
	接收 FIFO 数据寄存器_1	SCFRDR_1	R	不定值	H'E8007814	8
	FIFO 控制寄存器_1	SCFCR_1	R/W	H'0000	H'E8007818	16
	FIFO 数据计数置位寄存器_1	SCFDR_1	R	H'0000	H'E800781C	16
	串行端口寄存器_1	SCSPTR_1	R/W	H'0050	H'E8007820	16
	线路状态寄存器_1	SCLSR_1	R/(W)*2	H'0000	H'E8007824	16
	串行扩展模式寄存器_1	SCEMR_1	R/W	H'0000	H'E8007828	16
2	串行模式寄存器_2	SCSMR_2	R/W	H'0000	H'E8008000	16
	位速率寄存器_2	SCBRR_2	R/W	H'FF	H'E8008004	8
	串行控制寄存器_2	SCSCR_2	R/W	H'0000	H'E8008008	16
	发送 FIFO 数据寄存器_2	SCFTDR_2	W	不定值	H'E800800C	8
	串行状态寄存器_2	SCFSR_2	R/(W)*1	H'0060	H'E8008010	16
	接收 FIFO 数据寄存器_2	SCFRDR_2	R	不定值	H'E8008014	8
	FIFO 控制寄存器_2	SCFCR_2	R/W	H'0000	H'E8008018	16
	FIFO 数据计数置位寄存器_2	SCFDR_2	R	H'0000	H'E800801C	16
	串行端口寄存器_2	SCSPTR_2	R/W	H'0050	H'E8008020	16
	线路状态寄存器_2	SCLSR_2	R/(W)*2	H'0000	H'E8008024	16
	串行扩展模式寄存器_2	SCEMR_2	R/W	H'0000	H'E8008028	16
3	串行模式寄存器_3	SCSMR_3	R/W	H'0000	H'E8008800	16
	位速率寄存器_3	SCBRR_3	R/W	H'FF	H'E8008804	8
	串行控制寄存器_3	SCSCR_3	R/W	H'0000	H'E8008808	16
	发送 FIFO 数据寄存器_3	SCFTDR_3	W	不定值	H'E800880C	8
	串行状态寄存器_3	SCFSR_3	R/(W)*1	H'0060	H'E8008810	16

通道	寄存器名	略称	R/W	初始值	地址	存取长度
3	接收 FIFO 数据寄存器 _3	SCFRDR_3	R	不定值	H'E8008814	8
	FIFO 控制寄存器 _3	SCFCR_3	R/W	H'0000	H'E8008818	16
	FIFO 数据计数置位寄存器 _3	SCFDR_3	R	H'0000	H'E800881C	16
	串行端口寄存器 _3	SCSPTR_3	R/W	H'0050	H'E8008820	16
	线路状态寄存器 _3	SCLSR_3	R/(W)*2	H'0000	H'E8008824	16
	串行扩展模式寄存器 _3	SCEMR_3	R/W	H'0000	H'E8008828	16
4	串行模式寄存器 _4	SCSMR_4	R/W	H'0000	H'E8009000	16
	位速率寄存器 _4	SCBRR_4	R/W	H'FF	H'E8009004	8
	串行控制寄存器 _4	SCSCR_4	R/W	H'0000	H'E8009008	16
	发送 FIFO 数据寄存器 _4	SCFTDR_4	W	不定值	H'E800900C	8
	串行状态寄存器 _4	SCFSR_4	R/(W)*1	H'0060	H'E8009010	16
	接收 FIFO 数据寄存器 _4	SCFRDR_4	R	不定值	H'E8009014	8
	FIFO 控制寄存器 _4	SCFCR_4	R/W	H'0000	H'E8009018	16
	FIFO 数据计数置位寄存器 _4	SCFDR_4	R	H'0000	H'E800901C	16
	串行端口寄存器 _4	SCSPTR_4	R/W	H'0050	H'E8009020	16
	线路状态寄存器 _4	SCLSR_4	R/(W)*2	H'0000	H'E8009024	16
	串行扩展模式寄存器 _4	SCEMR_4	R/W	H'0000	H'E8009028	16
5	串行模式寄存器 _5	SCSMR_5	R/W	H'0000	H'E8009800	16
	位速率寄存器 _5	SCBRR_5	R/W	H'FF	H'E8009804	8
	串行控制寄存器 _5	SCSCR_5	R/W	H'0000	H'E8009808	16
	发送 FIFO 数据寄存器 _5	SCFTDR_5	W	不定值	H'E800980C	8
	串行状态寄存器 _5	SCFSR_5	R/(W)*1	H'0060	H'E8009810	16
	接收 FIFO 数据寄存器 _5	SCFRDR_5	R	不定值	H'E8009814	8
	FIFO 控制寄存器 _5	SCFCR_5	R/W	H'0000	H'E8009818	16
	FIFO 数据计数置位寄存器 _5	SCFDR_5	R	H'0000	H'E800981C	16
	串行端口寄存器 _5	SCSPTR_5	R/W	H'0050	H'E8009820	16
	线路状态寄存器 _5	SCLSR_5	R/(W)*2	H'0000	H'E8009824	16
	串行扩展模式寄存器 _5	SCEMR_5	R/W	H'0000	H'E8009828	16
6	串行模式寄存器 _6	SCSMR_6	R/W	H'0000	H'E800A000	16
	位速率寄存器 _6	SCBRR_6	R/W	H'FF	H'E800A004	8
	串行控制寄存器 _6	SCSCR_6	R/W	H'0000	H'E800A008	16
	发送 FIFO 数据寄存器 _6	SCFTDR_6	W	不定值	H'E800A00C	8
	串行状态寄存器 _6	SCFSR_6	R/(W)*1	H'0060	H'E800A010	16
	接收 FIFO 数据寄存器 _6	SCFRDR_6	R	不定值	H'E800A014	8
	FIFO 控制寄存器 _6	SCFCR_6	R/W	H'0000	H'E800A018	16
	FIFO 数据计数置位寄存器 _6	SCFDR_6	R	H'0000	H'E800A01C	16
	串行端口寄存器 _6	SCSPTR_6	R/W	H'0050	H'E800A020	16
	线路状态寄存器 _6	SCLSR_6	R/(W)*2	H'0000	H'E800A024	16
	串行扩展模式寄存器 _6	SCEMR_6	R/W	H'0000	H'E800A028	16

通道	寄存器名	略称	R/W	初始值	地址	存取长度
7	串行模式寄存器_7	SCSMR_7	R/W	H'0000	H'E800A800	16
	位速率寄存器_7	SCBRR_7	R/W	H'FF	H'E800A804	8
	串行控制寄存器_7	SCSCR_7	R/W	H'0000	H'E800A808	16
	发送 FIFO 数据寄存器_7	SCFTDR_7	W	不定值	H'E800A80C	8
	串行状态寄存器_7	SCFSR_7	R/(W)*1	H'0060	H'E800A810	16
	接收 FIFO 数据寄存器_7	SCFRDR_7	R	不定值	H'E800A814	8
	FIFO 控制寄存器_7	SCFCR_7	R/W	H'0000	H'E800A818	16
	FIFO 数据计数置位寄存器_7	SCFDR_7	R	H'0000	H'E800A81C	16
	串行端口寄存器_7	SCSPTR_7	R/W	H'0050	H'E800A820	16
	线路状态寄存器_7	SCLSR_7	R/(W)*2	H'0000	H'E800A824	16
	串行扩展模式寄存器_7	SCEMR_7	R/W	H'0000	H'E800A828	16

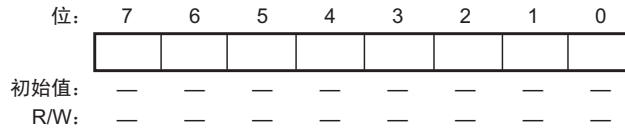
【注】 *1 为了清除标志，只能写“0”。bit15～8、3、2为只读位，不能写。

*2 为了清除标志，只能写“0”。bit15～1为只读位，不能写。

16.3.1 接收移位寄存器（SCRSR）

SCRSR 是接收串行数据的寄存器。按照从 LSB（bit0）开始接收的顺序，将 RxD 引脚输入的串行数据保存到 SCRSR，并且转换为并行数据。当接收完 1 字节的数据时，数据就自动传送到接收 FIFO 数据寄存器（SCFRDR）。

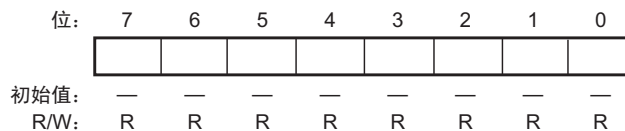
CPU 不能直接读写 SCRSR。



16.3.2 接收 FIFO 数据寄存器（SCFRDR）

SCFRDR 是保存已接收串行数据的 16 段 FIFO 寄存器。当接收完 1 字节的串行数据时，就将接收的串行数据从接收移位寄存器（SCRSR）传送并且保存到 SCFRDR，然后结束接收运行。在结束 16 字节的保存前可连续接收，CPU 能读但不能写 SCFRDR。如果在接收 FIFO 数据寄存器中没有接收数据的状态下读数据，读取值就为不定值。

如果 SCFRDR 为接收数据满，就会丢失以后接收的串行数据。



16.3.3 发送移位寄存器 (SCTSR)

SCTSR 是发送串行数据的寄存器。一旦将发送数据从发送 FIFO 数据寄存器 (SCFTDR) 传送到 SCTSR, 就从 LSB (bit0) 开始按顺序将数据发送到 TxD 引脚, 进行串行数据的发送。当发送完 1 字节的数据时, 下一个发送数据就自动从 SCFTDR 传送到 SCTSR, 开始发送。

CPU 不能直接读写 SCTSR。

位:	7	6	5	4	3	2	1	0
初始值:	—	—	—	—	—	—	—	—
R/W:	—	—	—	—	—	—	—	—

16.3.4 发送 FIFO 数据寄存器 (SCFTDR)

SCFTDR 是保存串行发送数据的 16 段 FIFO 寄存器。如果检测到发送移位寄存器 (SCTSR) 为空, 就将写到 SCFTDR 的发送数据传送到 SCTSR, 开始串行发送。可将串行发送直连续进行到 SCFTDR 变为发送数据空为止。CPU 能随时写 SCFTDR。

如果 SCFTDR 为发送数据满 (16 字节), 就不能写下一个数据。即使写, 数据也被忽视。

位:	7	6	5	4	3	2	1	0
初始值:	—	—	—	—	—	—	—	—
R/W:	W	W	W	W	W	W	W	W

16.3.5 串行模式寄存器 (SCSMR)

SCSMR 是用于设定串行通信格式和选择波特率发生器时钟源的寄存器。

CPU 能随时读写 SCSMR。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	C/A	CHR	PE	O/E	STOP	—	CKS[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 8	—	全 0	R	保留位 读写值都为“0”。
7	C/A	0	R/W	通信模式 从异步模式和时钟同步模式中选择运行模式。 0: 异步模式 1: 时钟同步模式
6	CHR	0	R/W	字符长 从 7 位数据和 8 位数据中选择异步模式的数据长。在时钟同步模式中, 与 CHR 的设定无关, 固定为 8 位数据长。 0: 8 位数据 1: 7 位数据 * 【注】 * 如果选择 7 位数据, 发送 FIFO 数据寄存器的 MSB (bit7) 就不被发送。

位	位名	初始值	R/W	说明
5	PE	0	R/W	<p>奇偶校验允许</p> <p>在异步模式中，选择是否在发送时附加奇偶校验位以及在接收时检查奇偶校验位。在时钟同步模式中，与 PE 位的设定无关，不附加也不检查奇偶校验位。</p> <p>0: 禁止附加和检查奇偶校验位 1: 允许附加和检查奇偶校验位 *</p> <p>【注】 * 如果将 PE 位置 “1”，就在发送时将 $\overline{O/E}$ 位指定的偶校验位或者奇校验位附加给发送数据后进行发送，而在接收时检查接收的奇偶校验位是否为 $\overline{O/E}$ 位指定的偶校验位或者奇校验位。</p>
4	$\overline{O/E}$	0	R/W	<p>奇偶校验模式</p> <p>选择是通过偶校验还是通过奇校验来附加和检查奇偶校验位。只在异步模式中将 PE 位置 “1” 并且允许附加和检查奇偶校验位时，$\overline{O/E}$ 位的设定才有效；在时钟同步模式中或者在异步模式中禁止附加和检查奇偶校验位时，$\overline{O/E}$ 位的指定无效。</p> <p>0: 偶校验 *1 1: 奇校验 *2</p> <p>【注】 *1 如果设定为偶校验，就在发送时附加奇偶校验位，使奇偶校验位和发送字符中的 “1” 的个数为偶数，然后进行发送；在接收时检查奇偶校验位和接收字符中的 “1” 的个数是否为偶数。 *2 如果设定为奇校验，就在发送时附加奇偶校验位，使奇偶校验位和发送字符中的 “1” 的个数为奇数，然后进行发送；在接收时检查奇偶校验位和接收字符中的 “1” 的个数是否为奇数。</p>
3	STOP	0	R/W	<p>停止位长</p> <p>在异步模式中，从 1 位和 2 位中选择停止位长。STOP 位的设定只在异步模式中有效。因为在设定为时钟同步模式时不附加停止位，所以此位的设定无效。另外，在接收时，与 STOP 位的设定无关，只检查接收的第 1 个停止位。当第 2 个停止位是 “1” 时，作为停止位处理，而当第 2 个停止位是 “0” 时，作为下一个发送字符的起始位处理。</p> <p>0: 1 个停止位 发送时，在发送字符的末尾附加 1 位的 “1”（停止位）后进行发送。 1: 2 个停止位 发送时，在发送字符的末尾附加 2 位的 “1”（停止位）后进行发送。</p>
2	—	0	R	<p>保留位</p> <p>读写值都为 “0”。</p>
1、0	CKS[1:0]	00	R/W	<p>时钟选择</p> <p>选择内部波特率发生器的内部时钟源。</p> <p>有关时钟源、位速率寄存器的设定值和波特率的关系，请参照 “16.3.8 位速率寄存器（SCBRR）”。</p> <p>00: P1ϕ 时钟 01: P1ϕ/4 时钟 10: P1ϕ/16 时钟 11: P1ϕ/64 时钟</p> <p>【注】 P1ϕ: 外围时钟</p>

16.3.6 串行控制寄存器 (SCSCR)

SCSCR 是进行发送和接收、允许或者禁止中断请求以及选择发送 / 接收时钟源的寄存器。CPU 能随时读写 SCSCR。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TIE	RIE	TE	RE	REIE	—	—	CKE[1:0]
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 8	—	全 0	R	保留位 读写值都为“0”。
7	TIE	0	R/W	发送中断允许 在串行状态寄存器 (SCFSR) 的 TDFE 标志被置“1”时 (串行发送数据从发送 FIFO 数据寄存器 (SCFTDR) 传送到发送移位寄存器 (SCTSR), 并且发送 FIFO 寄存器的数据量小于指定的发送触发个数), 允许或者禁止发生发送 FIFO 数据空的中断 (TXIF) 请求。 0: 禁止发送 FIFO 数据空的中断 (TXI) 请求 1: 允许发送 FIFO 数据空的中断 (TXI) 请求 * 【注】 * 能通过写发送数据 (数据量大于 SCFTDR 指定的发送触发个数), 在读 TDFE 标志的“1”后将该标志清“0”或者通过将 TIE 清“0”来解除 TXI。
6	RIE	0	R/W	接收中断允许 允许或者禁止以下的中断请求: 1. 串行状态寄存器 (SCFSR) 的 RDF 标志或者 DR 标志被置“1”时的接收 FIFO 数据满中断 (RXI) 请求。 2. SCFSR 的 ER 标志被置“1”时的接收错误中断 (ERI) 请求。 3. SCFSR 的 BRK 标志或者线路状态寄存器 (SCLSR) 的 ORER 标志被置“1”时的中止中断 (BRI) 请求。 0: 禁止接收 FIFO 数据满中断 (RXI) 请求、接收错误中断 (ERI) 请求和中止中断 (BRI) 请求 1: 允许接收 FIFO 数据满中断 (RXI) 请求、接收错误中断 (ERI) 请求和中止中断 (BRI) 请求 * 【注】 * 通过在读 DR 标志或者 RDF 标志的“1”后将该位清“0”, 或者通过将 RIE 位清“0”来解除 RXI 中断。通过在读 ER 标志、BRK 标志或者 ORER 标志的“1”后将这些标志清“0”, 或者通过将 RIE 位和 REIE 位清“0”来解除 ERI 中断和 BRI 中断。
5	TE	0	R/W	发送允许 允许或者禁止串行发送。 0: 禁止发送 1: 允许发送 * 【注】 * 如果在此状态下将发送数据写到 SCFTDR, 就开始串行发送。必须在将 TE 位置“1”前设定 SCSMR 和 SCFCR, 决定发送格式并且对发送 FIFO 进行复位。

位	位名	初始值	R/W	说明
4	RE	0	R/W	<p>接收允许 允许或者禁止串行接收。</p> <p>0: 禁止接收 *1 1: 允许接收 *2</p> <p>【注】 *1 必须注意: 即使将 RE 位清“0”, DR、ER、BRK、RDF、FER、PER 和 ORER 的各位也不受影响而保持状态。 *2 在此状态下, 如果在异步模式中检测到起始位以及在时钟同步模式中检测到同步时钟, 就开始串行接收。必须在将 RE 位置“1”前设定串行模式寄存器 (SCSMR) 和 FIFO 控制寄存器 (SCFCR), 决定接收格式并且对接收 FIFO 进行复位。</p>
3	REIE	0	R/W	<p>接收错误中断允许 允许或者禁止接收错误中断 (ERI) 请求和中止中断 (BRI) 请求。但是, REIE 位的设定只在 RIE 位是“0”时有效。</p> <p>0: 禁止接收错误的中断 (ERI) 请求和中止中断 (BRI) 请求 1: 允许接收错误的中断 (ERI) 请求和中止中断 (BRI) 请求 *</p> <p>【注】 * 通过在读 ER 标志、BRK 标志或者 ORER 标志的“1”后将该标志清“0”, 或者通过将 RIE 位和 REIE 位清“0”来解除 ERI 中断请求和 BRI 中断请求。即使将 RIE 位置“0”, 只要将 REIE 位置“1”, 也产生 ERI 中断和 BRI 中断。</p>
2	—	0	R	<p>保留位 读写值都为“0”。</p>
1、0	CKE[1:0]	00	R/W	<p>时钟允许 选择时钟源以及禁止或者允许 SCK 引脚的输出时钟。根据 CKE[1:0] 位决定 SCK 引脚是为串行时钟的输出引脚还是为串行时钟的输入引脚。要在时钟同步模式中设定为同步时钟输出时, 必须在将 SCSMR 的 C/A 位置“1”后设定 CKE[1:0]。</p> <ul style="list-style-type: none"> • 异步模式 <ul style="list-style-type: none"> 00: 内部时钟 /SCK 引脚为输入引脚 (忽视输入信号) 01: 内部时钟 /SCK 引脚为时钟输出引脚 (输出 16 倍或者 8 倍位速率的时钟) 10: 外部时钟 /SCK 引脚为时钟输入引脚 (输入 16 倍或者 8 倍位速率的时钟) 11: 禁止设定 • 时钟同步模式 <ul style="list-style-type: none"> 00: 内部时钟 /SCK 引脚是同步时钟的输出引脚 01: 内部时钟 /SCK 引脚是同步时钟的输出引脚 10: 外部时钟 /SCK 引脚是同步时钟的输入引脚 11: 禁止设定

16.3.7 串行状态寄存器 (SCFSR)

SCFSR 是 16 位寄存器，高 8 位表示接收 FIFO 数据寄存器数据的接收错误数，低 8 位表示运行状态的状态标志。

CPU 能随时读写 SCFSR，但是不能给 ER、TEND、TDFE、BRK、RDF 和 DR 的各状态标志写“1”。另外，为了将这些标志清“0”，必须预先读取“1”。PER (bit15 ~ 12、2) 标志和 FER (bit11 ~ 8、3) 标志为只读标志，不能写。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PER[3:0]				FER[3:0]				ER	TEND	TDFE	BRK	FER	PER	RDF	DR
初始值:	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R	R	R/(W)*	R/(W)*

【注】* 为了清除标志，只能在读“1”后写“0”。

位	位名	初始值	R/W	说明
15 ~ 12	PER[3:0]	0000	R	奇偶校验错误数 表示在接收 FIFO 数据寄存器 (SCFRDR) 保存的接收数据中发生奇偶校验错误的数量。 在 SCFSR 的 ER 位被置位后，bit15 ~ 12 的值表示发生奇偶校验错误的数量。当 SCFRDR 的 16 字节接收数据都发生奇偶校验错误时，PER[3:0] 为“0000”。
11 ~ 8	FER[3:0]	0000	R	帧错误数 表示在接收 FIFO 数据寄存器 (SCFRDR) 保存的接收数据中发生帧错误的数量。 在 SCFSR 的 ER 位被置位后，bit11 ~ 8 的值表示发生帧错误的数量。当 SCFRDR 的 16 字节接收数据都发生帧错误时，FER[3:0] 为“0000”。
7	ER	0	R/(W)*	接收错误 表示发生帧错误或者在接收含奇偶校验的数据时发生奇偶校验错误。*1 0: 表示正在接收或者接收已正常结束 [清除条件] • 上电复位 • 在读 ER 位的“1”后写“0”时 1: 表示在接收时发生帧错误或者奇偶校验错误 [置位条件] • 在接收完 1 个数据时，检查接收数据的最后停止位是否为“1”，检查结果是最后的停止位为“0” *2。 • 在接收时的接收数据和奇偶校验位的“1”的个数与串行模式寄存器 (SCSMR) 的 O/E 位指定的偶 / 奇校验的设定不同时 【注】 *1 在将 SCSCR 的 RE 位清“0”时，ER 标志不受影响而继续保持原来的状态。即使发生接收错误，还是将接收数据传送到 SCFRDR 并且继续接收。能通过 SCFSR 的 FER 位或者 PER 位来判断在从 SCFRDR 读到的数据中是否有接收错误。 *2 在 2 个停止位模式中，只检查第 1 个停止位而不检查第 2 个停止位。

位	位名	初始值	R/W	说明
6	TEND	1	R/(W)*	<p>发送结束</p> <p>表示在发送字符的最后一位时因 SCFTDR 中无有效数据而结束发送。</p> <p>0: 表示正在发送</p> <p>[清除条件]</p> <ul style="list-style-type: none"> 在给 SCFTDR 写发送数据时或者在读 TEND 位的“1”后给 TEND 标志写“0”时 *1 <p>1: 表示发送结束</p> <p>[置位条件]</p> <ul style="list-style-type: none"> 上电复位 当 SCSCR 的 TE 位是“0”时 在发送 1 字节串行发送字符的最后一位时, SCFTDR 中无发送数据时 <p>【注】 *1 直接存储器存取控制器通过 TXI 中断请求将数据写到 SCFTDR 时, 不能用作发送结束标志。</p>
5	TDFE	1	R/(W)*	<p>发送 FIFO 数据空</p> <p>表示将数据从发送 FIFO 数据寄存器 (SCFTDR) 传送到发送移位寄存器 (SCTSR), 并且 SCFTDR 的数据量少于 FIFO 控制寄存器 (SCFCR) 的 TTRG[1:0] 指定的送触发的数据量, 允许将发送数据写到 SCFTDR。</p> <p>0: 表示写到 SCFTDR 的发送数据量大于指定的发送触发个数</p> <p>[清除条件]</p> <ul style="list-style-type: none"> 在读 TDFE 位的“1”状态后, 将大于指定发送触发个数的数据写到 SCFTDR 并且给 TDFE 写“0”时 当通过发送 FIFO 数据空中断 (TXI) 启动直接存储器存取控制器并且将大于指定发送触发个数的数据写到 SCFTDR 时 <p>1: 表示写到 SCFTDR 的发送数据量小于等于指定的发送触发个数 *1</p> <p>[置位条件]</p> <ul style="list-style-type: none"> 上电复位 当写到发送结果 SCFTDR 的发送数据量小于等于指定的发送触发个数时 <p>【注】 *1 因为 SCFTDR 是 16 字节的 FIFO 寄存器, 所以在 TDFE 位为“1”的状态下能写的最大数据量为 16 减去指定的发送触发个数。即使写更多的数据, 此数据也被忽视。SCFDR 的高 8 位表示 SCFTDR 的数据量。</p>
4	BRK	0	R/(W)*	<p>中止的检测</p> <p>表示在接收数据过程中检测到中止信号。</p> <p>0: 没有中止信号</p> <p>[清除条件]</p> <ul style="list-style-type: none"> 上电复位 在读 BRK 位的“1”后给 BRK 标志写“0”时 <p>1: 接收到中止信号 *1</p> <p>[置位条件]</p> <ul style="list-style-type: none"> 当接收到含有帧错误的数据并且继续有大于等于 1 帧长的空间 0 (低电平) 时 <p>【注】 *1 如果检测到中止, 就在检测后停止将接收数据 (H'00) 传送到 SCFRDR。当中止结束并且接收信号变为标记 1 时, 重新开始传送接收数据。</p>

位	位名	初始值	R/W	说明
3	FER	0	R	<p>表示帧错误</p> <p>在异步模式中，表示在从接收 FIFO 数据寄存器（SCFRDR）读到的数据中是否有帧错误。</p> <p>0：表示在下次从 SCFRDR 读到的接收数据中未发生帧错误 [清除条件]</p> <ul style="list-style-type: none"> • 上电复位 • 下次从 SCFRDR 读到的数据无帧错误 <p>1：表示在下次从 SCFRDR 读到的接收数据中发生帧错误 [置位条件]</p> <ul style="list-style-type: none"> • 在下一个 SCFRDR 读数据中有帧错误
2	PER	0	R	<p>表示奇偶校验错误</p> <p>在异步模式中，表示在从接收 FIFO 数据寄存器（SCFRDR）读到的数据中是否有奇偶校验错误。</p> <p>0：表示在下次从 SCFRDR 读到的接收数据中未发生奇偶校验错误 [清除条件]</p> <ul style="list-style-type: none"> • 上电复位 • 在下一个 SCFRDR 读数据中没有奇偶校验错误 <p>1：表示在下次从 SCFRDR 读到的接收数据中发生奇偶校验错误 [置位条件]</p> <ul style="list-style-type: none"> • 在下一个 SCFRDR 读数据中有奇偶校验错误
1	RDF	0	R/(W)*	<p>接收 FIFO 数据满</p> <p>表示将接收数据传送到接收 FIFO 数据寄存器（SCFRDR）并且 SCFRDR 的接收数据量大于 FIFO 控制寄存器（SCFCR）的 RTRG[1:0] 位指定的接收触发个数。</p> <p>0：表示写到 SCFRDR 的接收数据量小于指定的接收触发个数 [清除条件]</p> <ul style="list-style-type: none"> • 上电复位 • 在读 RDF 位的“1”后，读 SCFRDR，直到 SCFRDR 的接收数据量小于指定接收触发个数为止，并且给 RDF 写“0”时 • 当通过接收 FIFO 数据满中断（RXI）启动直接存储器存取控制器并且读 SCFRDR，直到 SCFRDR 的接收数据量小于指定接收触发个数时 <p>1：表示 SCFRDR 的接收数据量大于等于指定的接收触发个数 [置位条件]</p> <ul style="list-style-type: none"> • 当将大于等于指定接收触发个数的接收数据量保存到 SCFRDR 时 *1 <p>【注】 *1 因为 SCFRDR 是 16 字节的 FIFO 寄存器，所以在 RDF 是“1”时能读的最大数据量为指定的接收触发个数。如果在读 SCFRDR 的全部数据后继续阅读数据，读取值就为不定值。SCFRDR 的低 8 位表示 SCFRDR 的接收数据量。</p>

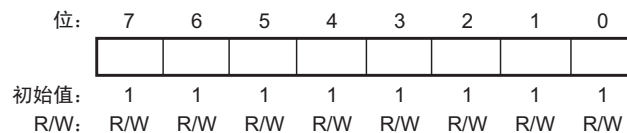
位	位名	初始值	R/W	说明
0	DR	0	R/(W)*	<p>接收数据就绪</p> <p>在异步模式中，表示将小于指定接收触发个数的数据保存到接收 FIFO 数据寄存器（SCFRDR）并且从最后的停止位开始经过 15ETU 的时间后还未接收到下一个数据。在时钟同步模式中，此位不被置位。</p> <p>0：表示正在接收或者在接收正常结束后 SCFRDR 没有接收数据 [清除条件]</p> <ul style="list-style-type: none"> • 上电复位 • 在读 DR 位的“1”后，读 SCFRDR 的全部接收数据，并且写“0”时 • 通过接收 FIFO 数据满中断（RXI）启动直接存储器存取控制器并且读 SCFRDR 的全部接收数据时 <p>1：表示没有接收到下一个接收数据 [置位条件]</p> <ul style="list-style-type: none"> • 当将小于指定接收触发个数的数据保存到 SCFRDR 并且从最后的停止位开始经过 15ETU 的时间 *1 后还未接收到下一个数据时 <p>【注】 *1 相当于 8 位和 1 个停止位格式的 1.5 帧。 (ETU Element Time Unit: 单元时间单位)</p>

【注】 * 为了清除标志，只能在读“1”后写“0”。

16.3.8 位速率寄存器（SCBRR）

SCBRR 是 8 位寄存器，和串行模式寄存器（SCSMR）的 CKS[1:0]、串行扩展模式寄存器（SCEMR）的 BGDM 位及 ABCS 位一起设定串行发送和接收的位速率。

CPU 能随时读写 SCBRR。在上电复位时，将 SCBRR 初始化为“H'FF”。另外，因为每个通道的波特率发生器的控制是独立的，所以能分别给 8 个通道设定不同的值。



通过以下表达式求 SCBRR 的设定值：

【异步模式】

- 波特率发生器正常模式（SCEMR 的 BGDM 位为“0”）

$$N = (P1\phi / (64 \times 2^{2n-1} \times B)) \times 10^6 - 1$$

（以 16 倍位速率的基本时钟运行的情况）

$$N = (P1\phi / (32 \times 2^{2n-1} \times B)) \times 10^6 - 1$$

（以 8 倍位速率的基本时钟运行的情况）
- 波特率发生器倍速模式（SCEMR 的 BGDM 位为“1”）

$$N = (P1\phi / (32 \times 2^{2n-1} \times B)) \times 10^6 - 1$$

（以 16 倍位速率的基本时钟运行的情况）

$$N = (P1\phi / (16 \times 2^{2n-1} \times B)) \times 10^6 - 1$$

（以 8 倍位速率的基本时钟运行的情况）

【时钟同步模式】

$$N = \frac{P1\phi}{8 \times 2^{2n-1} \times B} \times 10^6 - 1$$

B : 位速率 (bit/s)

N : 波特率发生器 SCBRR 的设定值 ($0 \leq N \leq 255$)
(必须设定为满足电特性的值。)

P1 ϕ : 外围模块的工作频率 (MHz)

n : 波特率发生器的输入时钟 (n=0、1、2、3)
(n和时钟的关系请参照表 16.3)

表 16.3 SCSMR 的设定

n	时钟	SCSMR 的设定值	
		CKS[1]	CKS[0]
0	P1 ϕ	0	0
1	P1 ϕ /4	0	1
2	P1 ϕ /16	1	0
3	P1 ϕ /64	1	1

通过以下计算式求异步模式的位速率误差:

- 波特率发生器正常模式 (SCEMR 的 BGDM 位为 “0”)
 - 误差 (%) = $\{((P1\phi \times 10^6) / ((N+1) \times B \times 64 \times 2^{2n-1})) - 1\} \times 100$
(以 16 倍位速率的基本时钟运行的情况)
 - 误差 (%) = $\{((P1\phi \times 10^6) / ((N+1) \times B \times 32 \times 2^{2n-1})) - 1\} \times 100$
(以 8 倍位速率的基本时钟运行的情况)
- 波特率发生器倍速模式 (SCEMR 的 BGDM 位为 “1”)
 - 误差 (%) = $\{((P1\phi \times 10^6) / ((N+1) \times B \times 32 \times 2^{2n-1})) - 1\} \times 100$
(以 16 倍位速率的基本时钟运行的情况)
 - 误差 (%) = $\{((P\phi \times 10^6) / ((N+1) \times B \times 16 \times 2^{2n-1})) - 1\} \times 100$
(以 8 倍位速率的基本时钟运行的情况)

当波特率发生器正常模式（SCEMR 的 BGDM 位为“0”）并且以 16 倍位速率的基本时钟运行（SCEMR 的 ABCS 位为“0”）时，异步模式 SCBRR 的设定例子表 16.4 所示。另外，时钟同步模式 SCBRR 的设定例子如表 16.5 所示。

表 16.4 位速率的 SCBRR 设定例子（异步模式，BGDM 位为“0”，ABCS 位为“0”）

位速率 (bit/s)	P1 ϕ (MHz)								
	50			60			66.67		
	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)
110	3	221	-0.02						
150	3	162	-0.15	3	194	0.16	3	216	0.01
300	3	80	0.47	3	97	-0.35	3	108	-0.45
600	3	40	-0.76	3	48	-0.35	3	53	0.47
1200	2	80	0.47	2	97	-0.35	2	107	0.47
2400	2	40	-0.76	2	48	-0.35	2	53	0.47
4800	1	80	0.47	1	97	-0.35	1	108	-0.45
9600	1	40	-0.76	1	48	-0.35	1	53	0.47
19200	0	80	0.47	0	97	-0.35	0	108	-0.45
31250	0	49	0.00	0	59	0.00	0	66	-0.50
38400	0	40	-0.76	0	48	-0.35	0	53	0.47

【注】 必须尽量将误差设定在 1% 以内。

【符号说明】

空白栏: 不能设定。

— : 能设定, 但是会出现误差。

表 16.5 位速率的 SCBRR 设定例子（时钟同步模式）

位速率 (bit/s)	P1 ϕ (MHz)					
	50		60		66.67	
	n	N	n	N	n	N
500	—	—				
1000	3	194	3	233	—	—
2500	3	77	3	93	3	103
5000	2	155	2	187	2	207
10000	2	77	2	93	2	103
25000	1	124	1	149	1	166
50000	1	62	1	74	1	82
100000	0	124	0	149	0	166
250000	0	49	0	59	0	66
500000	0	24	0	29	—	—
1000000	—	—	0	14	—	—
2000000	—	—	—	—	—	—

【符号说明】

空白栏: 不能设定。

— : 能设定, 但是会出现误差。

使用波特率发生器时的异步模式各频率的最大位速率如表 16.6 所示，外部时钟输入的异步模式的最大位速率如表 16.7 所示，外部时钟输入（ $t_{Scyc}=12t_{pcyc}$ 的情况*）的时钟同步模式的最大位速率如表 16.8 所示。

【注】* 必须确认满足本 LSI 和通信对象的电特性。

表 16.6 使用波特率发生器时的各频率的最大位速率（异步模式）

P1 ϕ (MHz)	设定值				最大位速率 (bit/s)
	BGDM	ABCS	n	N	
50	0	0	0	0	1562500
		1	0	0	3125000
	1	0	0	0	3125000
		1	0	0	6250000
60	0	0	0	0	1875000
		1	0	0	3750000
	1	0	0	0	3750000
		1	0	0	7500000
66.67	0	0	0	0	2083333
		1	0	0	4166667
	1	0	0	0	4166667
		1	0	0	8333333

表 16.7 输入外部时钟时的最大位速率（异步模式）

P1 ϕ (MHz)	外部输入时钟 (MHz)	设定值	最大位速率 (bit/s)
		ABCS	
50	12.5000	0	781250
		1	1562500
60	15.0000	0	937500
		1	1875000
66.67	16.6667	0	1041667
		1	2083333

表 16.8 输入外部时钟时的最大位速率（时钟同步模式， $t_{Scyc}=12t_{pcyc}$ 的情况）

P1 ϕ (MHz)	外部输入时钟 (MHz)	最大位速率 (bit/s)
50	4.1667	4166666.7
60	5.0000	5000000.0
66.67	5.5556	5555555.5

16.3.9 FIFO 控制寄存器 (SCFCR)

SCFCR 是对发送 FIFO 数据寄存器和接收 FIFO 数据寄存器的数据量进行复位并且设定触发个数数据的寄存器，SCFCR 还有环回测试的允许位。

CPU 能随时读写 SCFCR。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RSTRG[2:0]			RTRG[1:0]		TTRG[1:0]		MCE	TFRST	RFRST	LOOP
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 8	RSTRG[2:0]	000	R/W	RTS 输出有效触发 当保存在接收 FIFO 数据寄存器 (SCFRDR) 的接收数据量大于等于下表所示的触发设定数时，RTS 信号为高电平。 000: 15 001: 1 010: 4 011: 6 100: 8 101: 10 110: 12 111: 14
7、6	RTRG[1:0]	00	R/W	接收 FIFO 数据量的触发 设定接收数据量 (指定接收触发个数)，作为串行状态寄存器 (SCFSR) 的 RDF 标志置位的基准。如果保存在接收 FIFO 数据寄存器 (SCFRDR) 的接收数据量大于等于以下所示的设定触发个数，就将 RDF 标志置“1”。 • 异步模式 00: 1 01: 4 10: 8 11: 14 • 时钟同步模式 00: 1 01: 2 10: 8 11: 14 【注】 在时钟同步模式中，要通过直接存储器存取控制器传送接收数据时，必须将接收触发个数设定为“1”。否则，需要通过 CPU 读 SCFRDR 中剩余的接收数据。

位	位名	初始值	R/W	说明
5、4	TTRG[1:0]	00	R/W	<p>发送 FIFO 数据量的触发</p> <p>设定发送数据量（指定发送触发个数），作为串行状态寄存器（SCFSR）TDFE 标志的置位基准。如果保存到发送 FIFO 数据寄存器（SCFTDR）的发送数据量小于等于以下所示的设定触发个数，就将 TDFE 标志置“1”。</p> <p>00: 8 (8) *</p> <p>01: 4 (12) *</p> <p>10: 2 (14) *</p> <p>11: 0 (16) *</p> <p>【注】 * () 内的数值表示 TDFE 标志被置“1”时的 SCFTDR 寄存器的空字节数。</p>
3	MCE	0	R/W	<p>调制解调器的控制允许</p> <p>允许或者禁止调制解调器的控制信号 $\overline{\text{CTS}}$ 和 $\overline{\text{RTS}}$。</p> <p>在通道 0 和通道 2 ~ 7 并且时钟同步模式中，必须将 MCE 置“0”。</p> <p>0: 禁止调制解调器信号 *</p> <p>1: 允许调制解调器信号</p> <p>【注】 * 与输入值无关，$\overline{\text{CTS}}$ 和 $\overline{\text{RTS}}$ 都被固定为 Low 电平有效。</p>
2	TFRST	0	R/W	<p>发送 FIFO 数据寄存器复位</p> <p>将发送 FIFO 数据寄存器内的发送数据置为无效，并且将数据复位成空状态。</p> <p>0: 禁止复位 *</p> <p>1: 允许复位</p> <p>【注】 * 在上电复位时进行复位。</p>
1	RFRST	0	R/W	<p>接收 FIFO 数据寄存器复位</p> <p>将接收 FIFO 数据寄存器内的接收数据置为无效，并且将数据复位成空状态。</p> <p>0: 禁止复位 *</p> <p>1: 允许复位</p> <p>【注】 * 在上电复位时进行复位。</p>
0	LOOP	0	R/W	<p>环回测试</p> <p>允许在内部将发送输出引脚（TxD）和接收输入引脚（RxD）连接并且将 $\overline{\text{RTS}}$ 引脚和 $\overline{\text{CTS}}$ 引脚连接，进行环回测试。</p> <p>0: 禁止环回测试</p> <p>1: 允许环回测试</p>

16.3.10 FIFO 数据计数置位寄存器 (SCFDR)

SCFDR 表示保存在发送 FIFO 数据寄存器 (SCFTDR) 和接收 FIFO 数据寄存器 (SCFRDR) 的数据量。高 8 位表示 SCFTDR 的发送数据量，低 8 位表示 SCFRDR 的接收数据量。CPU 能随时读 SCFDR。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	T[4:0]				—	—	—	R[4:0]					
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12 ~ 8	T[4:0]	00000	R	表示保存在 SCFTDR 的未发送的数据量。 H'00 表示没有发送数据，H'10 表示发送数据全部保存在 SCFTDR。
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4 ~ 0	R[4:0]	00000	R	表示保存在 SCFRDR 的接收数据量。 H'00 表示没有接收数据，H'10 表示接收数据全部保存在 SCFRDR。

16.3.11 串行端口寄存器 (SCSPTR)

SCSPTR 控制被此模块引脚多路复用的端口输入 / 输出和数据。能通过 bit7 和 bit6 读写 $\overline{\text{RTS}}$ 引脚的输入和输出数据，能通过 bit5 和 bit4 读写 CTS 引脚的输入和输出数据；能通过 bit3 和 bit2 读写 SCK 引脚的输入和输出数据；能通过 bit1 和 bit0 读 RxD 引脚的输入数据并且写 TxD 引脚的输出数据，控制串行发送和接收的中止。

CPU 能随时读写 SCSPTR。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPB2IO	SPB2DT
初始值:	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 8	—	全 0	R	保留位 读写值都为“0”
7	RTSIO	0	R/W	$\overline{\text{RTS}}$ 端口输入 / 输出 指定串行端口的 $\overline{\text{RTS}}$ 引脚的输入 / 输出。实际上，要将 $\overline{\text{RTS}}$ 引脚作为端口输出引脚来输出 RTSDT 位的设定值时，必须将 SCFCR 的 MCE 位置“0”。 0: 不将 RTSDT 位的值输出到 $\overline{\text{RTS}}$ 引脚 1: 将 RTSDT 位的值输出到 $\overline{\text{RTS}}$ 引脚
6	RTSDT	1	R/W	$\overline{\text{RTS}}$ 端口数据 指定串行端口的 $\overline{\text{RTS}}$ 引脚的输入 / 输出数据。通过 RTSIO 位指定输入或者输出。在输出时，将 RTSDT 位的值输出到 $\overline{\text{RTS}}$ 引脚。与 RTSIO 位的值无关，从 RTSDT 位读 $\overline{\text{RTS}}$ 引脚的值。但是，需要预先通过 PFC 设定为 $\overline{\text{RTS}}$ 输入 / 输出。 0: 输入 / 输出数据为低电平 1: 输入 / 输出数据为高电平

位	位名	初始值	R/W	说明
5	CTSIO	0	R/W	<p>CTS 端口输入 / 输出</p> <p>指定串行端口的 $\overline{\text{CTS}}$ 引脚的输入 / 输出。实际上, 要将 $\overline{\text{CTS}}$ 引脚作为端口输出引脚来输出 CTSDT 位的设定值时, 必须将 SCFCR 的 MCE 位置“0”。</p> <p>0: 不将 CTSDT 位的值输出到 $\overline{\text{CTS}}$ 引脚</p> <p>1: 将 CTSDT 位的值输出到 $\overline{\text{CTS}}$ 引脚</p>
4	CTSDT	1	R/W	<p>CTS 端口数据</p> <p>指定串行端口的 $\overline{\text{CTS}}$ 引脚的输入 / 输出数据。通过 CTSIO 位指定输入或者输出。在输出时, 将 CTSDT 位的值输出到 $\overline{\text{CTS}}$ 引脚。与 CTSIO 位的值无关, 从 CTSDT 位读 $\overline{\text{CTS}}$ 引脚的值。但是, 需要预先通过 PFC 设定为 $\overline{\text{CTS}}$ 输入 / 输出。</p> <p>0: 输入 / 输出数据为低电平</p> <p>1: 输入 / 输出数据为高电平</p>
3	SCKIO	0	R/W	<p>SCK 端口输入 / 输出</p> <p>指定串行端口的 SCK 引脚的输入 / 输出。实际上, 要将 SCK 引脚作为端口输出引脚来输出 SCKDT 位的设定值时, 必须将 SCSCR 的 CKE1 位和 CKE0 位置“0”。</p> <p>0: 不将 SCKDT 位的值输出到 SCK 引脚</p> <p>1: 将 SCKDT 位的值输出到 SCK 引脚</p>
2	SCKDT	0	R/W	<p>SCK 端口数据</p> <p>指定串行端口的 SCK 引脚的输入 / 输出数据。通过 SCKIO 位指定输入或者输出。在输出时, 将 SCKDT 位的值输出到 SCK 引脚。与 SCKIO 位的值无关, 从 SCKDT 位读 SCK 引脚的值。但是, 需要预先通过 PFC 设定为 SCK 输入 / 输出。</p> <p>0: 输入 / 输出数据为低电平</p> <p>1: 输入 / 输出数据为高电平</p>
1	SPB2IO	0	R/W	<p>串行端口的中止输入 / 输出</p> <p>指定串行端口的 TxD 引脚的输出条件。实际上, 要将 TxD 引脚作为端口输出引脚来输出 SPB2DT 位的设定值时, 必须将 SCSCR 的 TE 位置“0”。</p> <p>0: 不将 SPB2DT 位的值输出到 TxD 引脚</p> <p>1: 将 SPB2DT 位的值输出到 TxD 引脚</p>
0	SPB2DT	0	R/W	<p>串行端口的中止数据</p> <p>指定串行端口的 RxD 引脚的输入数据和 TxD 引脚的输出数据。通过 SPB2IO 位指定输入或者输出。当将 TxD 引脚设定为输出时, 将 SPB2DT 位的值输出到 TxD 引脚。与 SPB2IO 位的值无关, 从 SPB2DT 位读 RxD 引脚的值。但是, 需要预先通过 PFC 设定为 RxD 输入和 TxD 输出。</p> <p>0: 输入 / 输出数据为低电平</p> <p>1: 输入 / 输出数据为高电平</p>

16.3.12 线路状态寄存器 (SCLSR)

CPU 能随时读写 SCLSR，但是不能给 ORER 的状态标志写“1”。为了清“0”，必须预先读“1”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ORER
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/(W)*

【注】* 为了清除标志，只能在读“1”后写“0”。

位	位名	初始值	R/W	说明
15 ~ 1	—	全 0	R	保留位 读写值都为“0”
0	ORER	0	R/(W)*	溢出错误 表示在接收时发生溢出错误而异常结束。 0: 表示正在接收或者接收正常结束 *1 [清除条件] • 上电复位 • 在读 ORER 位的“1”后写“0”时 1: 表示在接收时发生溢出错误 *2 [置位条件] • 在接收 FIFO 为满 (接收到 16 字节数据) 的状态下结束下一次串行接收时 【注】 *1 在将串行控制寄存器 (SCSCR) 的 RE 位清“0”时，ORER 标志不受影响而保持原来的状态。 *2 接收 FIFO 数据寄存器 (SCFRDR) 保持发生溢出错误前的接收数据，丢失以后接收的数据。另外，在将 ORER 位置“1”的状态下，不能继续进行以后的串行接收。

16.3.13 串行扩展模式寄存器 (SCEMR)

CPU 能随时读写 SCEMR。在异步模式 (SCSMR 的 $\overline{C/A}$ 位为 “0”) 中时钟源为内部时钟并且 SCK 引脚为输入引脚 (SCSCR 的 $\overline{CKE[1:0]}$ 位为 “00”) 时, 通过将 BGDM 位置 “1”, 此模块内部的波特率发生器以倍速模式运行。

另外, 能通过改变 ABCS 位的设定, 选择异步模式中的 1 位期间的基本时钟。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	BGDM	—	—	—	—	—	—	ABCS
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
15 ~ 8	—	全 0	R	保留位 读写值都为 “0”。
7	BGDM	0	R/W	波特率发生器倍速模式 如果将 BGDM 位置 “1”, 此模块内部的波特率发生器就以倍速模式运行。BGDM 的设定只在异步模式 (SCSMR 的 $\overline{C/A}$ 位为 “0”) 中时钟源为内部时钟并且 SCK 引脚为输入引脚 (SCSCR 的 $\overline{CKE[1:0]}$ 位为 “00”) 时有效。在进行上述以外的设定时, 必须使用正常模式。 0: 正常模式 1: 倍速模式
6 ~ 1	—	全 0	R	保留位 读写值都为 “0”。
0	ABCS	0	R/W	异步基本时钟选择 选择异步模式中的 1 位期间的基本时钟。ABCS 位的设定只在异步模式 (SCSMR 的 $\overline{C/A}$ 位为 “0”) 中有效。 0: 以 16 倍位速率的基本时钟运行 1: 以 8 倍位速率的基本时钟运行

16.4 运行说明

16.4.1 概要

此模块能以异步模式（以字符为单位边取得同步边通信）和时钟同步模式（通过时钟脉冲边取得同步边通信）共 2 种方式进行串行通信。

此模块内置分别用于发送和接收的 16 段 FIFO 寄存器，能减少 CPU 的开销并且能连续进行高速通信。另外，通道 1 还具有用作调制解调器控制信号的 $\overline{\text{RTS}}$ 和 $\overline{\text{CTS}}$ 信号。通过串行模式寄存器（SCSMR）选择发送 / 接收格式，如表 16.9 所示。时钟源取决于串行控制寄存器（SCSCR）的 $\text{CKE}[1:0]$ 位的组合，如表 16.10 所示。

(1) 异步模式

- 数据长度：可选择 7 位或者 8 位。
- 可选择附加奇偶校验以及附加 1 位或者 2 位的停止位（根据这些组合，决定发送 / 接收格式和字符长）
- 在接收时，能检测到帧错误、奇偶校验错误、接收 FIFO 数据满、溢出错误、接收数据就绪和中止。
- 表示各发送 / 接收 FIFO 寄存器所保存的数据量。
- 时钟源：可选择内部时钟或者外部时钟。
当选择内部时钟时，通过波特率发生器时钟运行。
当选择外部时钟时，需要输入频率为 16/8 倍位速率的时钟（不使用内部波特率发生器）

(2) 时钟同步模式

- 发送 / 接收格式：固定为 8 位数据。
- 在接收时，能检测到溢出错误。
- 时钟源：可选择内部时钟或者外部时钟。
当选择内部时钟时，通过波特率发生器的时钟运行，并且将此时钟作为同步时钟输出到外部。
当选择外部时钟时，不使用内部波特率发生器，而使用输入的外部同步时钟运行。

表 16.9 SCSMR 的设定值和发送 / 接收格式

SCSMR				模式	发送 / 接收格式		
bit7 $\overline{\text{C/A}}$	bit6 CHR	bit5 PE	bit3 STOP		数据长度	奇偶校验位	停止位长
0	0	0	0	异步模式	8 位	无	1 位
			1				2 位
		1	0			有	1 位
			1				2 位
	1	0	0		7 位	无	1 位
			1				2 位
		1	0			有	1 位
			1				2 位
1	x	x	x	时钟同步模式	8 位	无	无

【符号说明】 x: Don't care

表 16.10 SCSMR、SCSCR 的设定值和时钟源的选择

SCSMR	SCSCR	模式	时钟源	SCK 引脚的功能
bit7	bit1、0			
$\overline{C/A}$	CKE[1:0]			
0	00	异步模式	内部	不使用 SCK 引脚。
	01			输出 16 倍或者 8 倍位速率的时钟。
	10		外部	输出 16 倍或者 8 倍位速率的时钟。
	11		禁止设定	
1	0x	时钟同步模式	内部	输出同步时钟。
	10		外部	输入同步时钟。
	11		禁止设定	

【符号说明】 x: Don't care

【注】 在使用波特率发生器倍速模式（BGDM 位为“1”）时，必须设定异步模式（ $\overline{C/A}$ 位为“0”）并且时钟源为内部时钟以及不使用 SCK 引脚（CKE[1:0] 位为“00”）。

16.4.2 异步模式中的运行

异步模式是以字符为单位边取得同步边进行串行通信的模式，在给数据附加表示通信开始的起始位和表示通信结束的停止位后进行字符的发送和接收。

此模块内部的发送部和接收部独立，因此能进行全双工通信，而且发送部和接收部都是 16 段 FIFO 缓冲结构，所以能在发送和接收过程中读写数据，连续进行发送和接收。

异步串行通信的一般格式如图 16.2 所示。

在异步串行通信中，通信线路一般保持为标记状态（高电平）。监视通信线路，将空状态（低电平）视为起始位，开始串行通信。

以起始位（低电平）开始，数据（LSB first: 从最低位开始）、奇偶校验位（高 / 低电平），最后为停止位（高电平）的顺序构成串行通信的 1 个字符。

在异步模式中，在接收时用起始位的下降沿进行同步。另外，因为在 1 位期间的 16 倍或者 8 倍位速率时钟的第 8/4 个时钟脉冲对数据进行采样，所以在各位的中央取得通信数据。

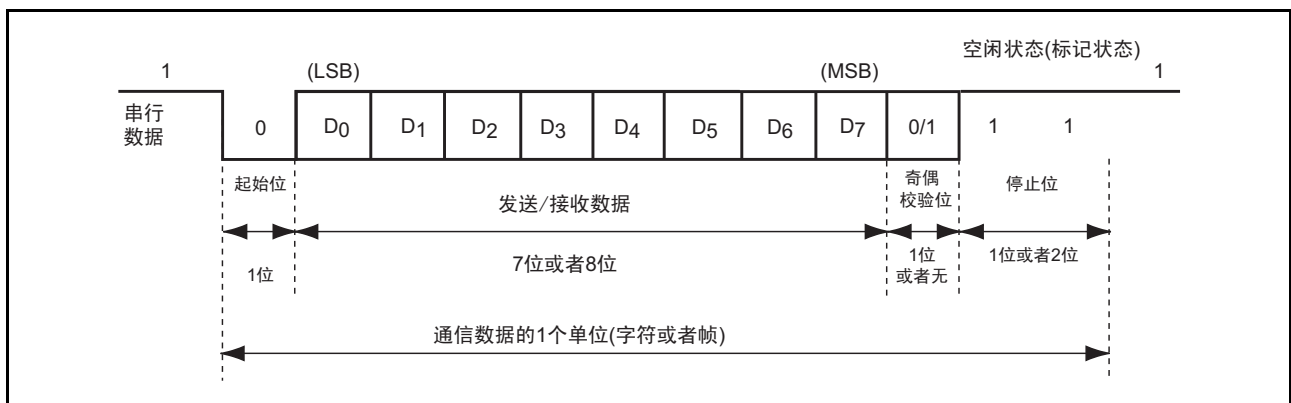


图 16.2 异步通信的数据格式（8 位数据 / 有奇偶校验 / 2 个停止位的例子）

(1) 发送 / 接收格式

能设定的发送 / 接收格式如表 16.11 所示。

发送 / 接收格式有 8 种，能通过设定串行模式寄存器（SCSMR）进行选择。

表 16.11 串行发送 / 接收格式（异步模式）

SCSMR 的设定			串行发送 / 接收格式和帧长											
CHR	PE	STOP	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	START 8位数据								STOP			
		1	START 8位数据								STOP STOP			
	1	0	START 8位数据								P	STOP		
		1	START 8位数据								P	STOP STOP		
1	0	0	START 7位数据							STOP				
		1	START 7位数据							STOP STOP				
	1	0	START 7位数据							P	STOP			
		1	START 7位数据							P	STOP STOP			

【符号说明】

START : 起始位

STOP : 停止位

P : 奇偶校验位

(2) 时钟

能通过设定 SCSMR 的 $\overline{C/A}$ 位和串行控制寄存器（SCSCR）的 CKE[1:0] 位，从内部波特率发生器生成的内部时钟或者从 SCK 引脚输入的外部时钟中选择此模块的发送/接收时钟。有关时钟源的选择，请参照表 16.10。

要将外部时钟输入到 SCK 引脚时，必须输入频率是 16 倍或者 8 倍位速率的时钟。

在通过内部时钟运行时，能从 SCK 引脚输出时钟。此时，输出的时钟频率是位速率的 16 倍或者 8 倍。

(3) 数据的发送和接收

• 初始化（异步模式）

在发送和接收数据前，必须在将串行控制寄存器（SCSCR）的 TE 位和 RE 位清“0”后按以下顺序进行初始化。

在更改运行模式和通信格式等时，必须在将 TE 位和 RE 位清“0”后按照以下步骤进行。如果将 TE 位清“0”，就对发送移位寄存器（SCTSR）进行初始化。但是，必须注意：即使将 TE 位和 RE 位清“0”，串行状态寄存器（SCFSR）、发送 FIFO 数据寄存器（SCFTDR）、接收 FIFO 数据寄存器（SCFRDR）也不进行初始化而保持其内容。必须在发送全部的发送数据并且 SCFSR 的 TEND 标志被置位后，将 TE 位清“0”。即使在发送过程中也能将 TE 位清“0”，但是在清“0”后发送数据变为标记状态。另外，在通过将 TE 位置“1”重新开始发送前，必须在将 SCFCR 的 TFRST 位置“1”后对 SCFTDR 进行复位。

在使用外部时钟时，不能在含有初始化的运行中停止时钟，否则运行就不稳定。初始化的流程图例子如图 16.3 所示。

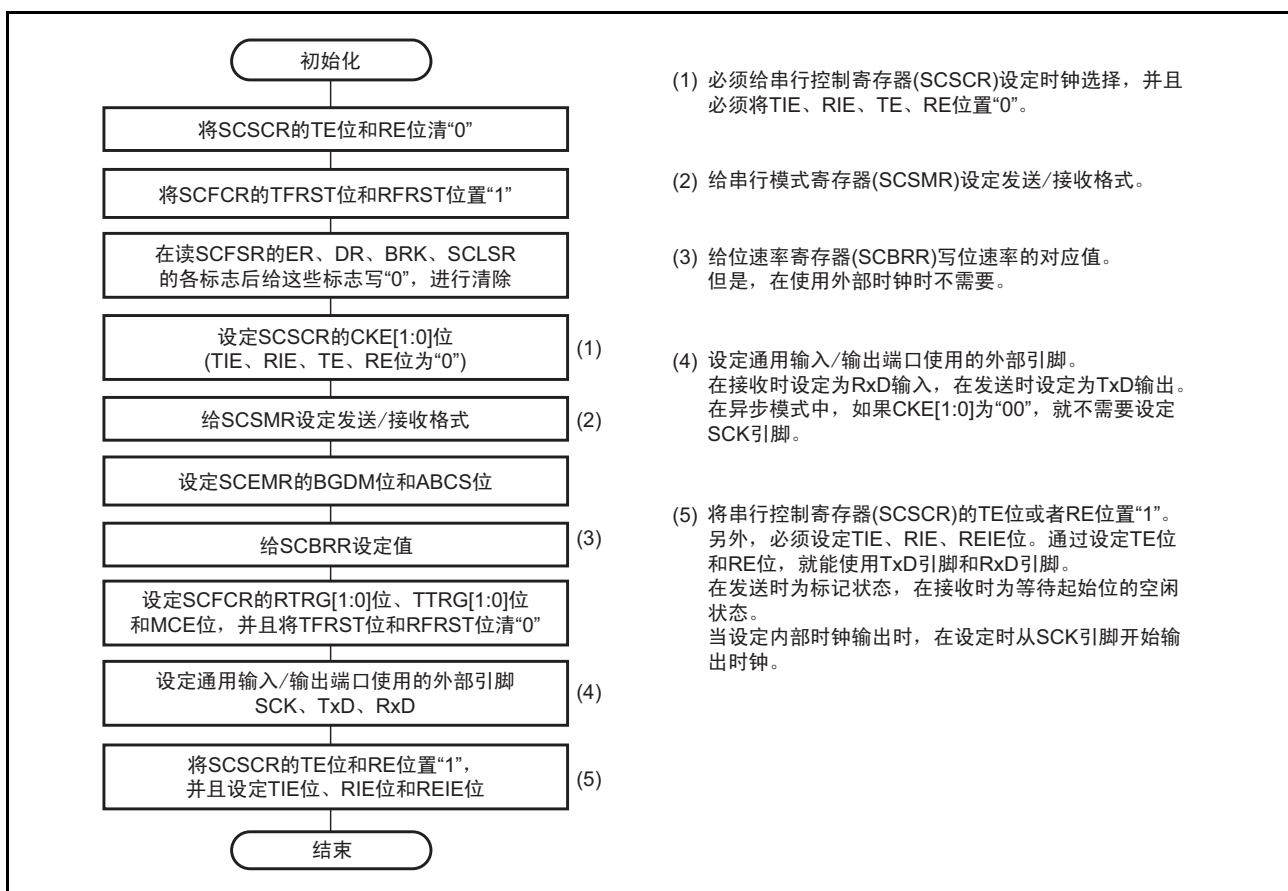


图 16.3 初始化的流程图例子

• 串行数据的发送（异步模式）

串行发送的流程图例子如图 16.4 所示。

必须在设定为可发送的状态后按照以下步骤发送串行数据。

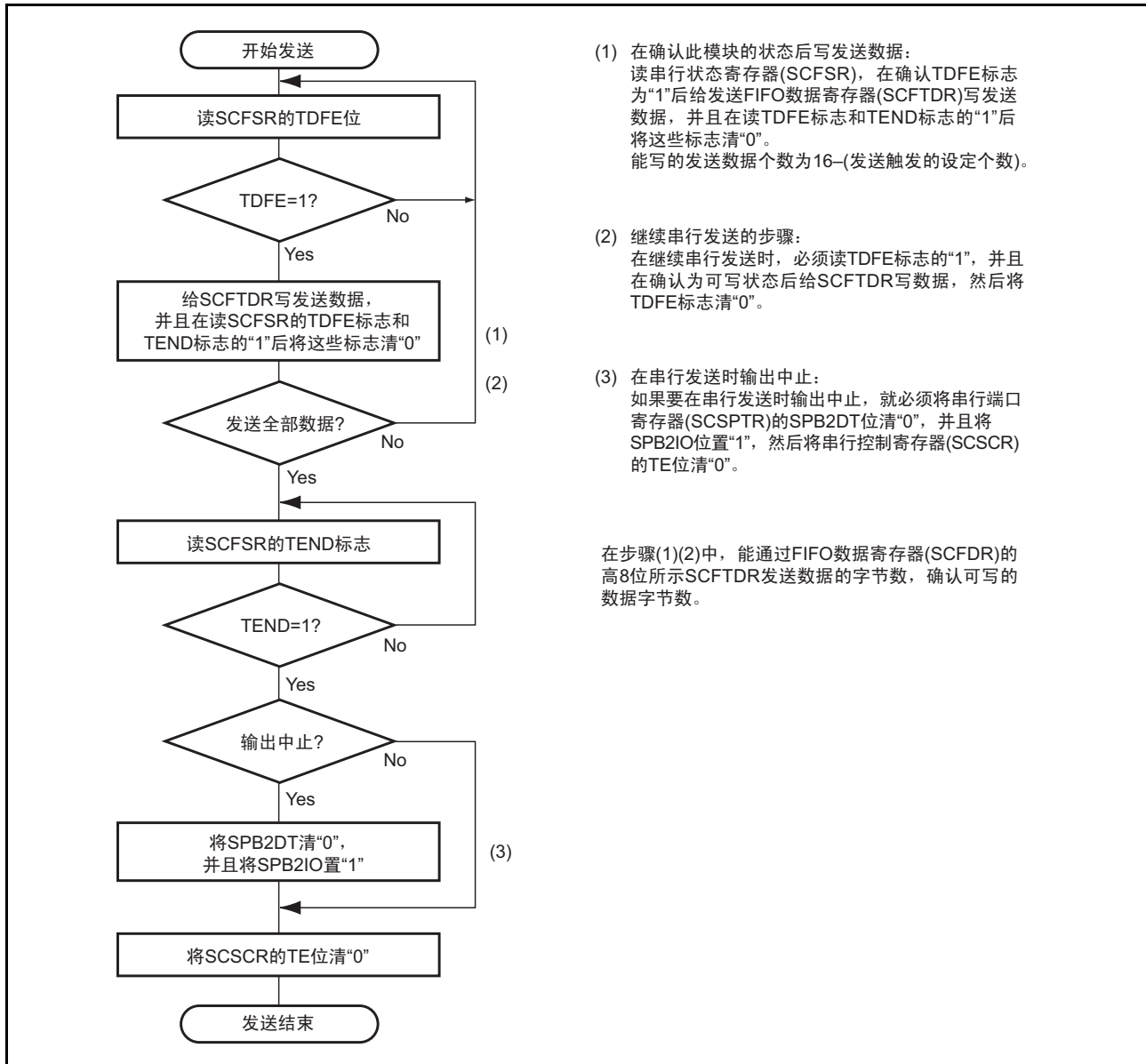


图 16.4 串行发送的流程图例子

串行发送时的运行如下：

1. 如果将数据写到发送FIFO数据寄存器（SCFTDR），数据就从SCFTDR传送到发送移位寄存器（SCTSR）。在将发送数据写到SCFTDR前，必须确认串行状态寄存器（SCFSR）的TDFE标志被置“1”。能写的发送数据字节数为16减去发送触发的设定数。
2. 如果在将数据从SCFTDR传送到SCTSR后开始发送，就进行连续发送，直到SCFTDR中没有发送数据为止。如果SCFTDR的发送数据字节数小于等于FIFO控制寄存器（SCFCR）设定的发送触发个数，就将TDFE标志置位。此时，如果串行控制寄存器（SCSCR）的TIE位被置“1”，就产生发送FIFO数据空中断（TXIF）请求。

按照以下顺序，从TxD引脚发送串行发送数据：

- a. 起始位：输出1位的“0”。

- b. 发送数据：从LSB开始，按顺序输出8位或者7位数据。
 - c. 奇偶校验位：输出1位的奇偶校验位（偶校验或者奇校验）
（另外，也能选择不输出奇偶校验位的格式）。
 - d. 停止位：输出1位或者2位的“1”（停止位）。
 - e. 标记状态：连续输出“1”，直到输出开始下一次发送的起始位为止。
3. 在输出停止位时检查SCFTDR的发送数据。如果有数据，就将数据从SCFTDR传送到SCTSR，在输出停止位后，开始下一帧的串行发送。

异步模式中的发送运行例子如图16.5所示。

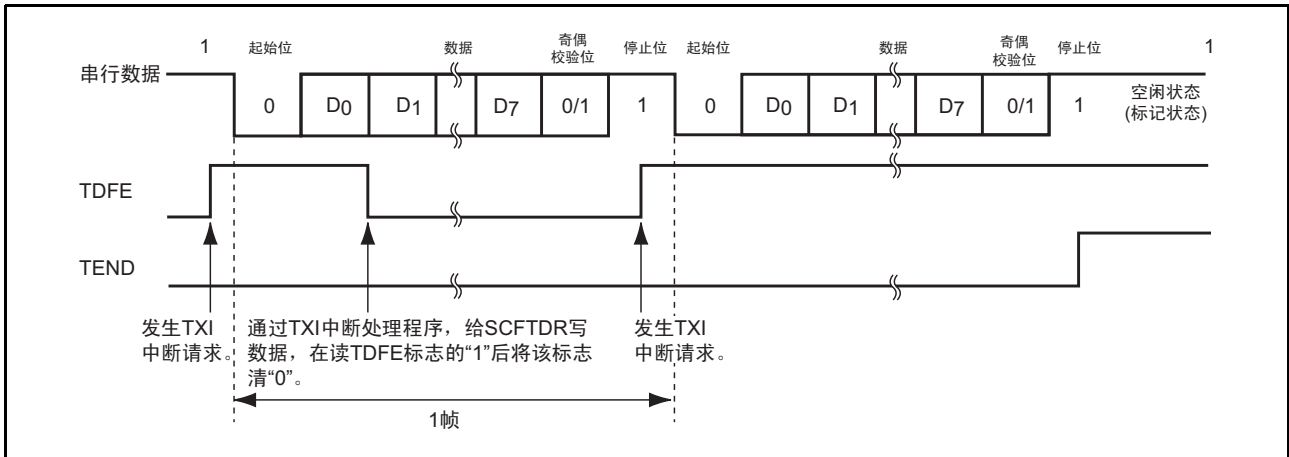


图 16.5 发送的运行例子（8 位数据 / 有奇偶校验 / 1 个停止位的例子）

4. 如果在通道1中允许调制解调器控制，就能通过 $\overline{\text{CTS}}$ 输入值停止或者重新开始发送。当 $\overline{\text{CTS}}$ 被置“1”时，如果正在发送，就在发送完1帧后变为标记状态。当 $\overline{\text{CTS}}$ 被置“0”时，从起始位开始输出下一个发送数据。

使用调制解调器控制的运行例子如图16.6所示。

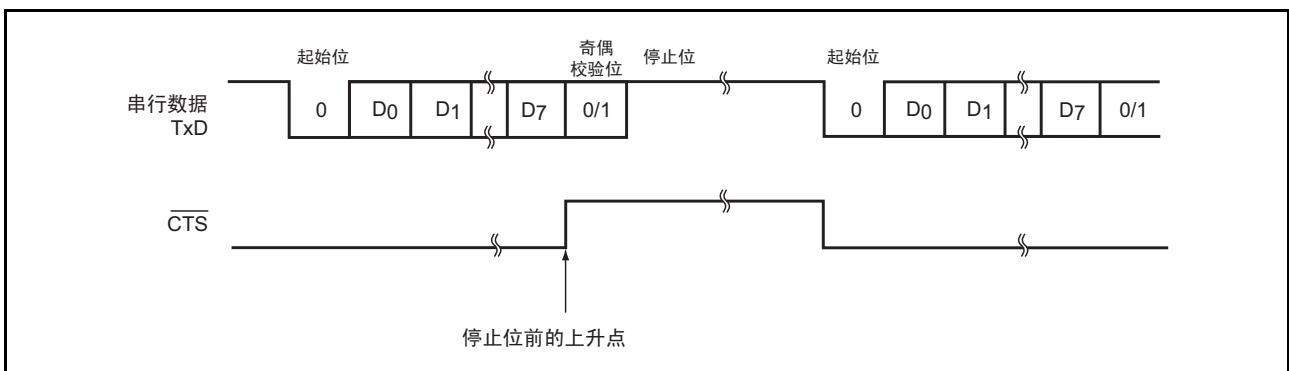


图 16.6 使用调制解调器控制的运行例子（ $\overline{\text{CTS}}$ ）

• 串行数据的接收（异步模式）

串行接收的流程图例子如图 16.7 和图 16.8 所示。

必须在设定为可接收状态后按照以下步骤接收串行数据。

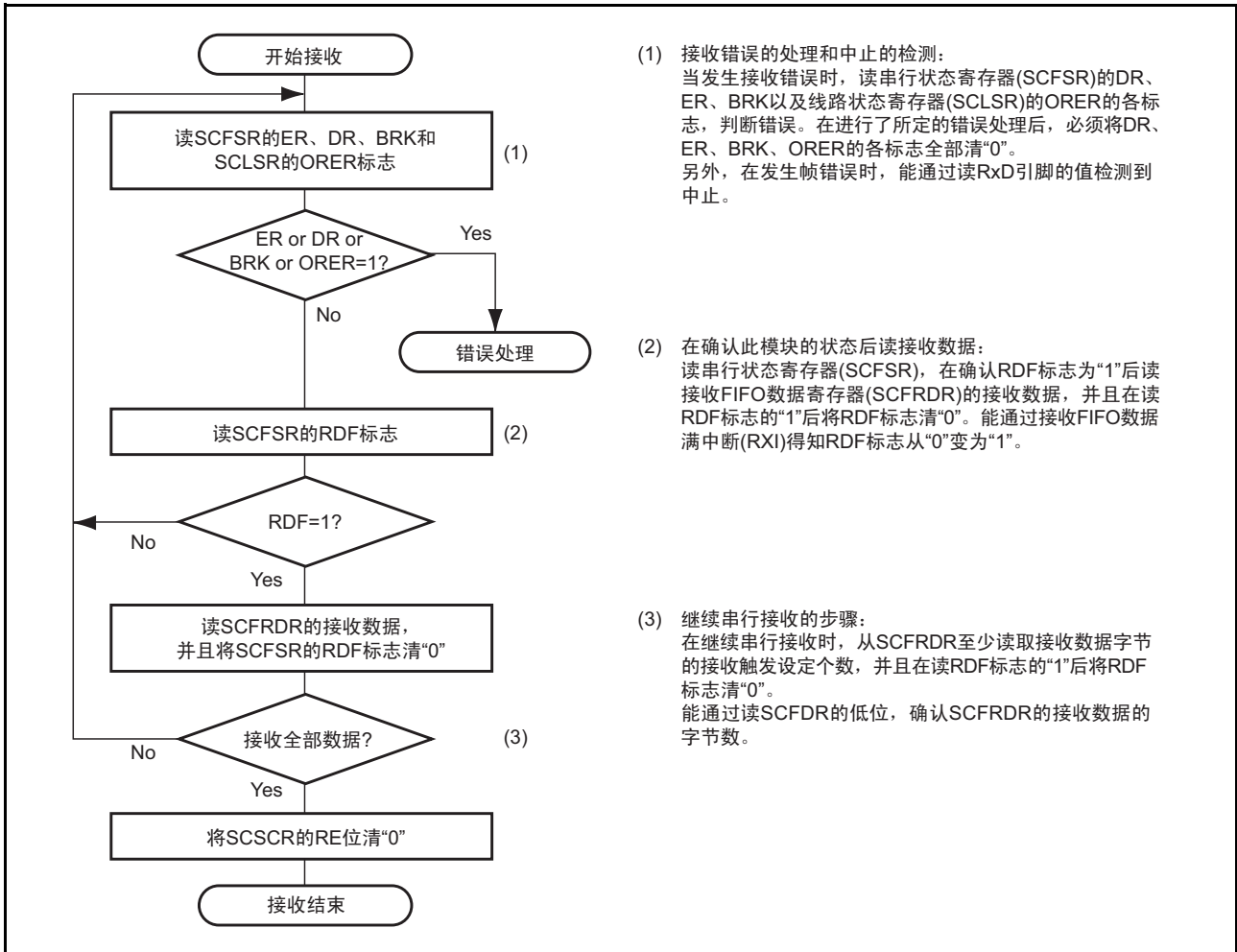


图 16.7 串行接收的流程图例子（1）

- (1) 接收错误的处理和中止的检测：
当发生接收错误时，读串行状态寄存器(SCFSR)的DR、ER、BRK以及线路状态寄存器(SCLSR)的ORER的各标志，判断错误。在进行了所定的错误处理后，必须将DR、ER、BRK、ORER的各标志全部清“0”。另外，在发生帧错误时，能通过读RxD引脚的值检测到中止。
- (2) 在确认此模块的状态后读接收数据：
读串行状态寄存器(SCFSR)，在确认RDF标志为“1”后读接收FIFO数据寄存器(SCFRDR)的接收数据，并且在读RDF标志的“1”后将RDF标志清“0”。能通过接收FIFO数据满中断(RXI)得知RDF标志从“0”变为“1”。
- (3) 继续串行接收的步骤：
在继续串行接收时，从SCFRDR至少读取接收数据字节的接收触发设定个数，并且在读RDF标志的“1”后将RDF标志清“0”。能通过读SCFRDR的低位，确认SCFRDR的接收数据的字节数。

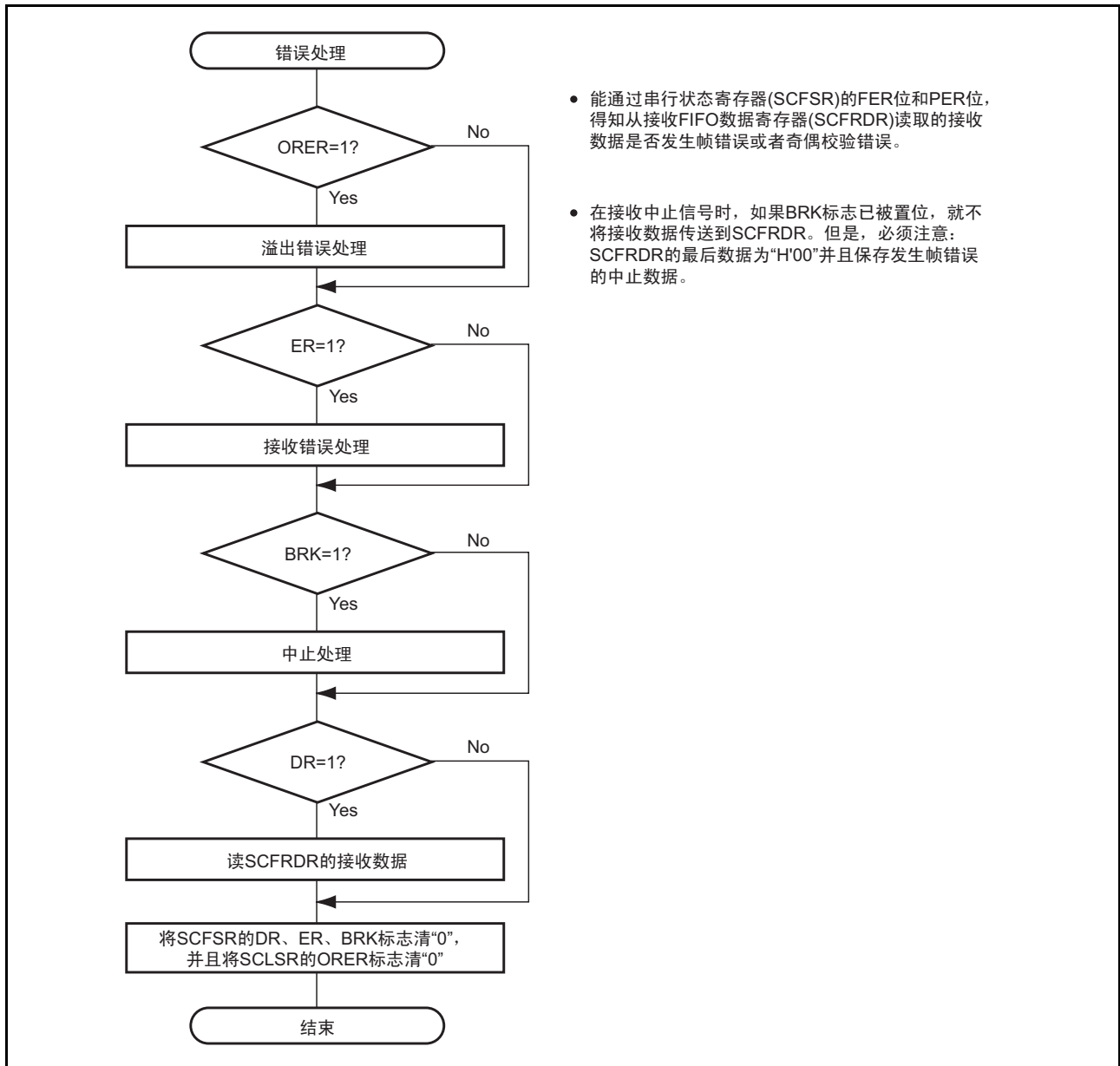


图 16.8 串行接收的流程图例子 (2)

接收时的运行如下：

1. 监视通信线路，如果检测到起始位的“0”，就和内部同步并且开始接收。
2. 按照从SCRSR的LSB到MSB的顺序保存接收的数据。
3. 接收奇偶校验位和停止位。

在接收后，进行以下检查：

- a. 停止位的检查：检查停止位是否为“1”。但是，在2个停止位的情况下，只检查第1个停止位。
- b. 检查是否能为将接收数据从接收移位寄存器（SCRSR）传送到SCFRDR的状态。
- c. 溢出错误的检查：检查表示未发生溢出错误的ORER标志是否为“0”。
- d. 中止的检查：检查表示未设定中止状态的BRK标志是否为“0”。

在以上检查全部通过后，将接收数据保存到SCFRDR。

【注】即使发生奇偶校验错误或者帧错误，也能继续接收。

- 在 RDF 或者 DR 标志变为“1”时，如果 SCSCR 的 RIE 位已被置“1”，就产生接收 FIFO 数据满中断（RXI）请求；在 ER 标志变为“1”时，如果 SCSCR 的 RIE 位或者 REIE 位已被置“1”，就产生接收错误中断（ERI）请求；在 BRK 标志或者 ORER 标志变为“1”时，如果 SCSCR 的 RIE 位或者 REIE 位已被置1，就产生中止接收的中断（BRI）请求。
异步模式接收时的运行例子如图 16.9 所示。

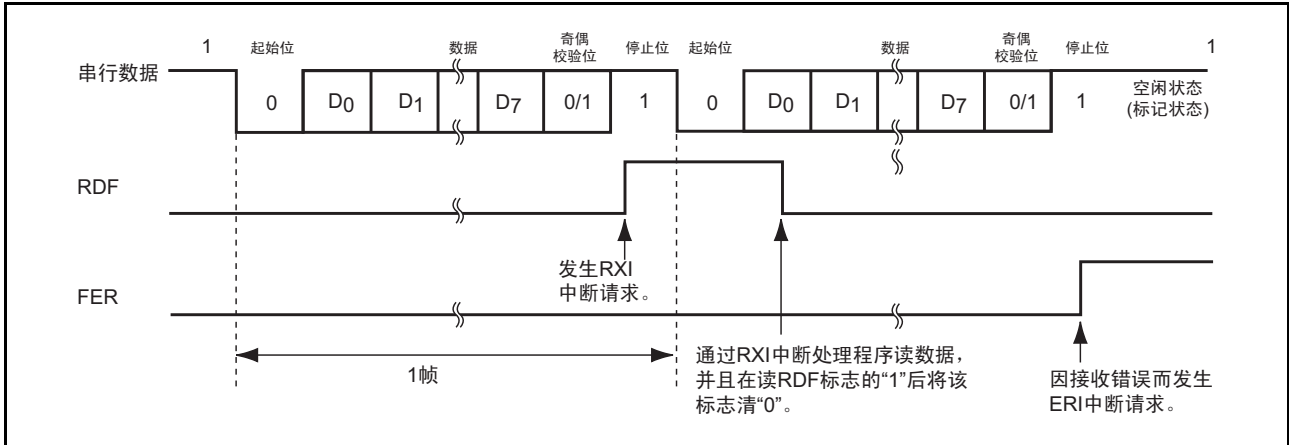


图 16.9 接收时的运行例子（8 位数据 / 有奇偶校验 / 1 个停止位的例子）

- 如果在通道 3 中调制解调器控制有效，就在 SCFRDR 为空时输出 $\overline{\text{RTS}}$ 信号。 $\overline{\text{RTS}}$ 为“0”时表示为可接收状态， $\overline{\text{RTS}}$ 为“1”时表示 SCFRDR 的数据量大于等于 RTS 输出有效触发的设定数。

使用调制解调器控制时的运行例子如图 16.10 所示。

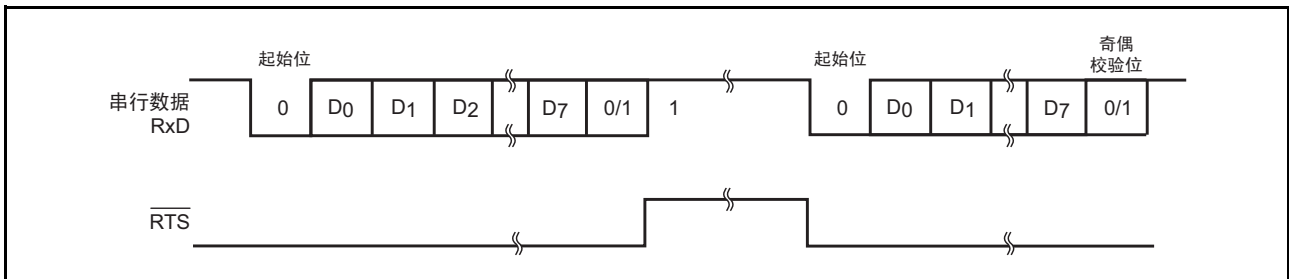


图 16.10 使用调制解调器控制时的运行例子（ $\overline{\text{RTS}}$ ）

16.4.3 时钟同步模式中的运行

时钟同步模式是和时钟脉冲同步进行数据发送和接收的模式，适用于高速串行通信。

此模块内部的发送部和接收部独立，因此能通过共享时钟进行全双工通信，而且发送部和接收部都是 16 段 FIFO 缓冲结构，所以能在发送和接收过程中读写数据，进行连续的发送和接收。

时钟同步串行通信的一般格式如图 16.11 所示。

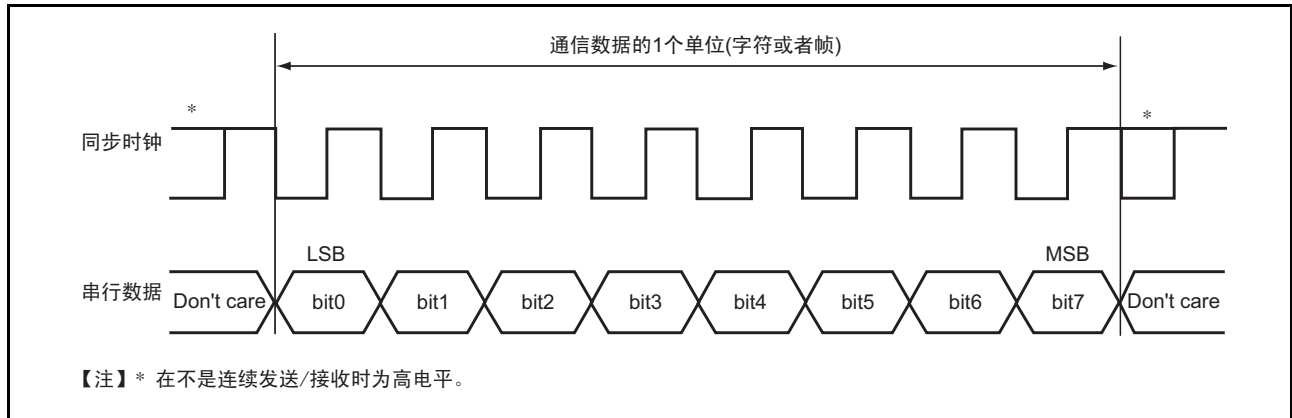


图 16.11 时钟同步通信的数据格式

在时钟同步串行通信中，在同步时钟的下降沿和下一个下降沿之间输出通信线路的数据，保证在同步时钟的上升沿数据有效。

以数据的 LSB 开始到 MSB（最后）的顺序输出串行通信的 1 个字符。输出 MSB 后的通信线路状态保持 MSB 的状态。

在时钟同步模式中，和同步时钟的上升沿同步接收数据。

(1) 发送 / 接收格式

固定为 8 位数据。

不能附加奇偶校验位。

(2) 时钟

能通过设定 SCSMR 的 $\overline{C/A}$ 位和 SCSCR 的 CKE[1:0] 位，选择内部波特率发生器生成的内部时钟或者从 SCK 引脚输入的外部同步时钟。

在通过内部时钟运行时，从 SCK 引脚输出同步时钟。同步时钟在发送和接收 1 个字符时输出 8 个脉冲，而在不进行发送和接收时固定为高电平。如果在只接收时选择内部时钟，就在 SCSCR 的 RE 位为“1”的期间输出时钟脉冲，直到接收 FIFO 内的数据量达到接收触发的设定数为止。

(3) 数据的发送和接收

初始化（时钟同步模式）

在发送和接收数据前，必须在将串行控制寄存器（SCSCR）的 TE 和 RE 位清“0”后按照以下步骤进行初始化。

在改变模式和通信格式等时，必须在将 TE 和 RE 位清“0”后按照以下步骤进行。如果将 TE 位清“0”，就对发送移位寄存器（SCTSR）进行初始化。必须注意：即使将 RE 位清“0”，也保持 RDF、PER、FER、ORER 各标志和接收数据寄存器（SCRDR）的内容。

初始化的流程图例子如图 16.12 所示。

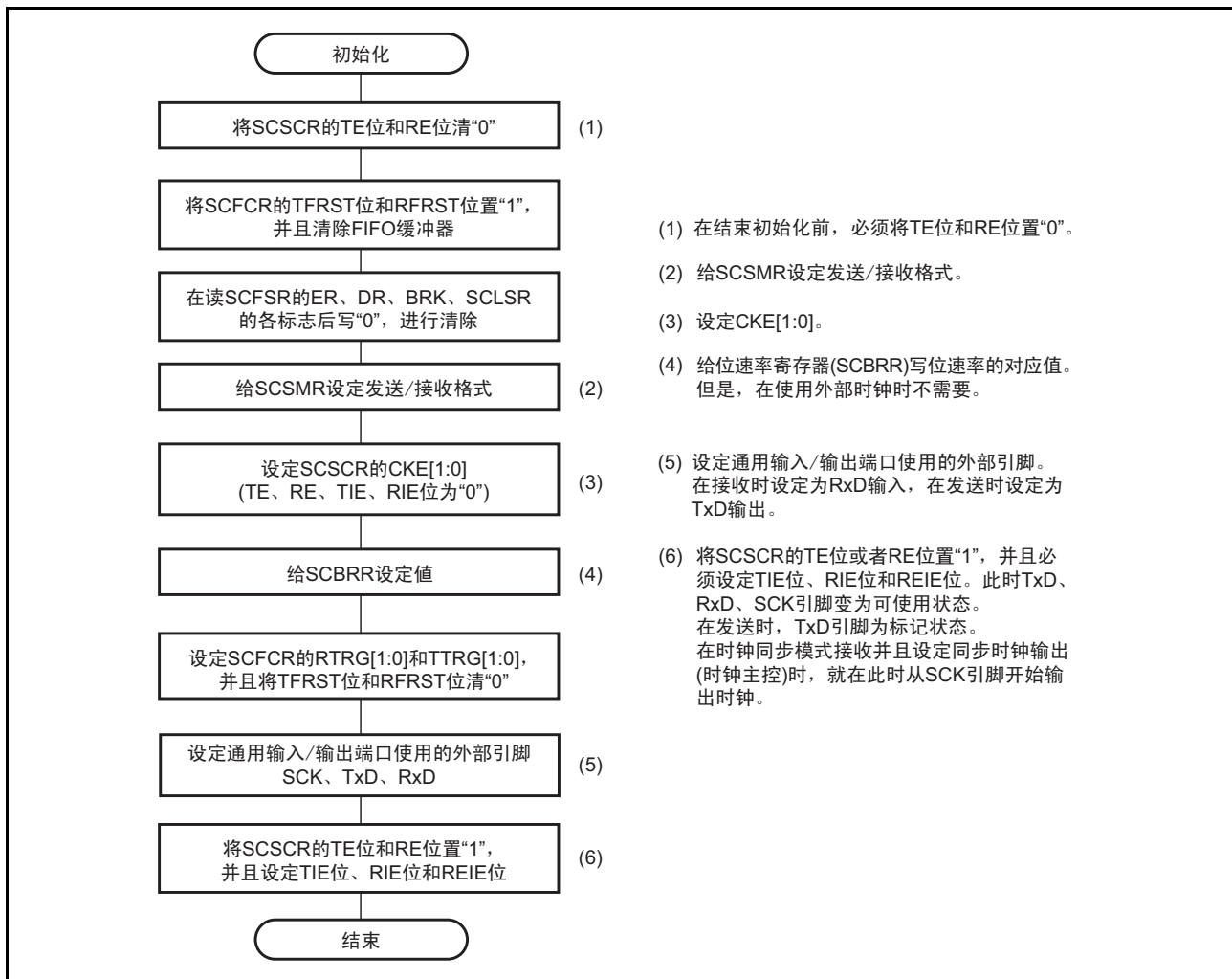


图 16.12 初始化的流程图例子

- 串行数据的发送（时钟同步模式）

串行发送的流程图例子如图 16.13 所示。

必须在设定为可发送的状态后按照以下步骤发送串行数据。

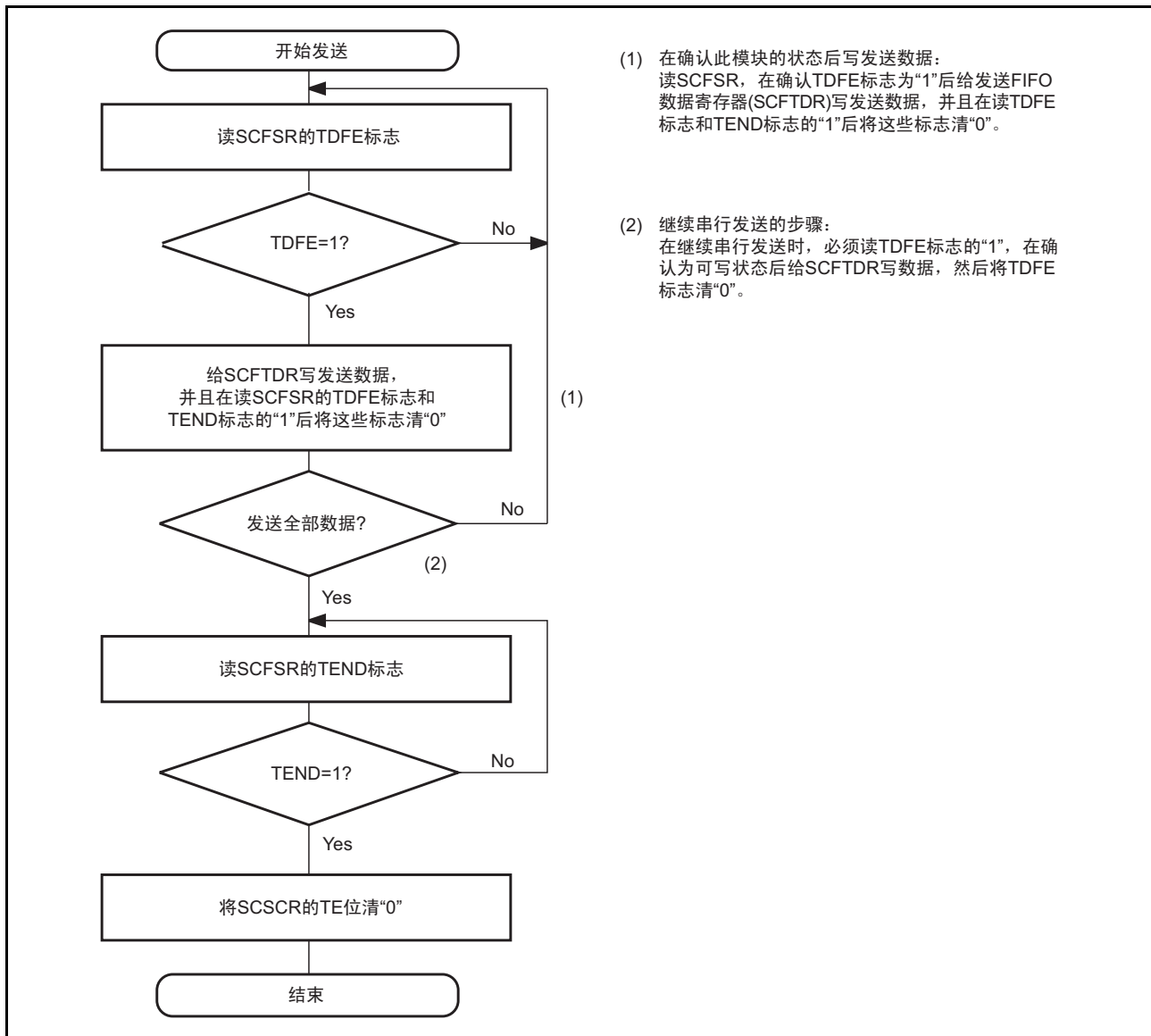


图 16.13 串行发送的流程图例子

串行发送时的运行如下：

1. 如果将数据写到发送FIFO数据寄存器（SCFTDR），数据就从SCFTDR传送到发送移位寄存器（SCTSR）。在将发送数据写到SCFTDR前，必须确认串行状态寄存器（SCFSR）的TDFE标志被置“1”。能写的发送数据字节数为16减去发送触发的设定数。
2. 如果在将数据从SCFTDR传送到SCTSR后开始发送，就进行连续发送，直到SCFTDR中没有发送数据为止。如果SCFTDR的发送数据字节数小于等于FIFO控制寄存器（SCFCR）设定的发送触发个数，就将TDFE标志置位。此时，如果串行控制寄存器（SCSCR）的TIE位被置“1”，就产生发送FIFO数据空中断（TXI）请求。

当设定为时钟输出模式时，输出同步时钟的8个脉冲；当设定为外部时钟时，和输入时钟同步输出数据。按照LSB（bit0）～MSB（bit7）的顺序从TxD引脚输出串行发送数据。

3. 在输出最后位时检查SCFTDR发送数据。如果有发送数据，就将数据从SCFTDR传送到SCTSR，开始下一帧的串行发送；如果没有数据，就将SCFSR的TEND标志置“1”，在输出最后位后，TXD引脚保持状态。
4. 在结束串行发送后，SCK引脚固定为高电平。

发送的运行例子如图 16.14 所示。

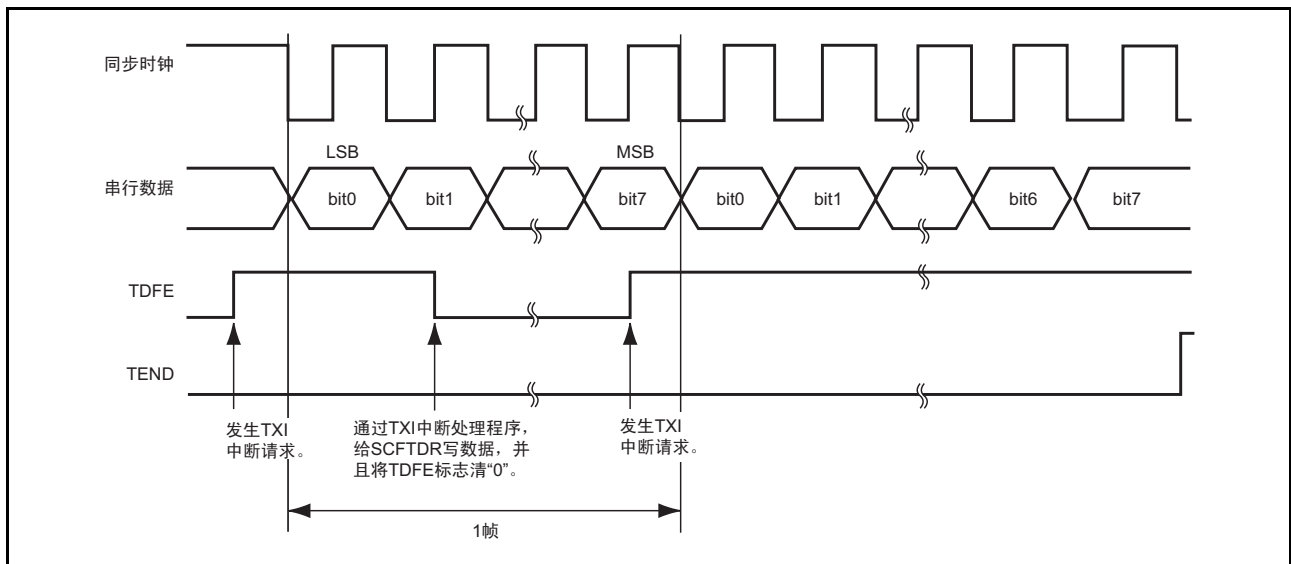


图 16.14 发送的运行例子

• 接收串行数据（时钟同步模式）

串行接收的流程图例子如图 16.15 和图 16.16 所示。

必须在设定为可接收的状态后按照以下步骤接收串行数据。

在不进行初始化而将运行模式从异步模式转换为时钟同步模式时，必须确认已将 ORER、PER、FER 各标志清“0”。

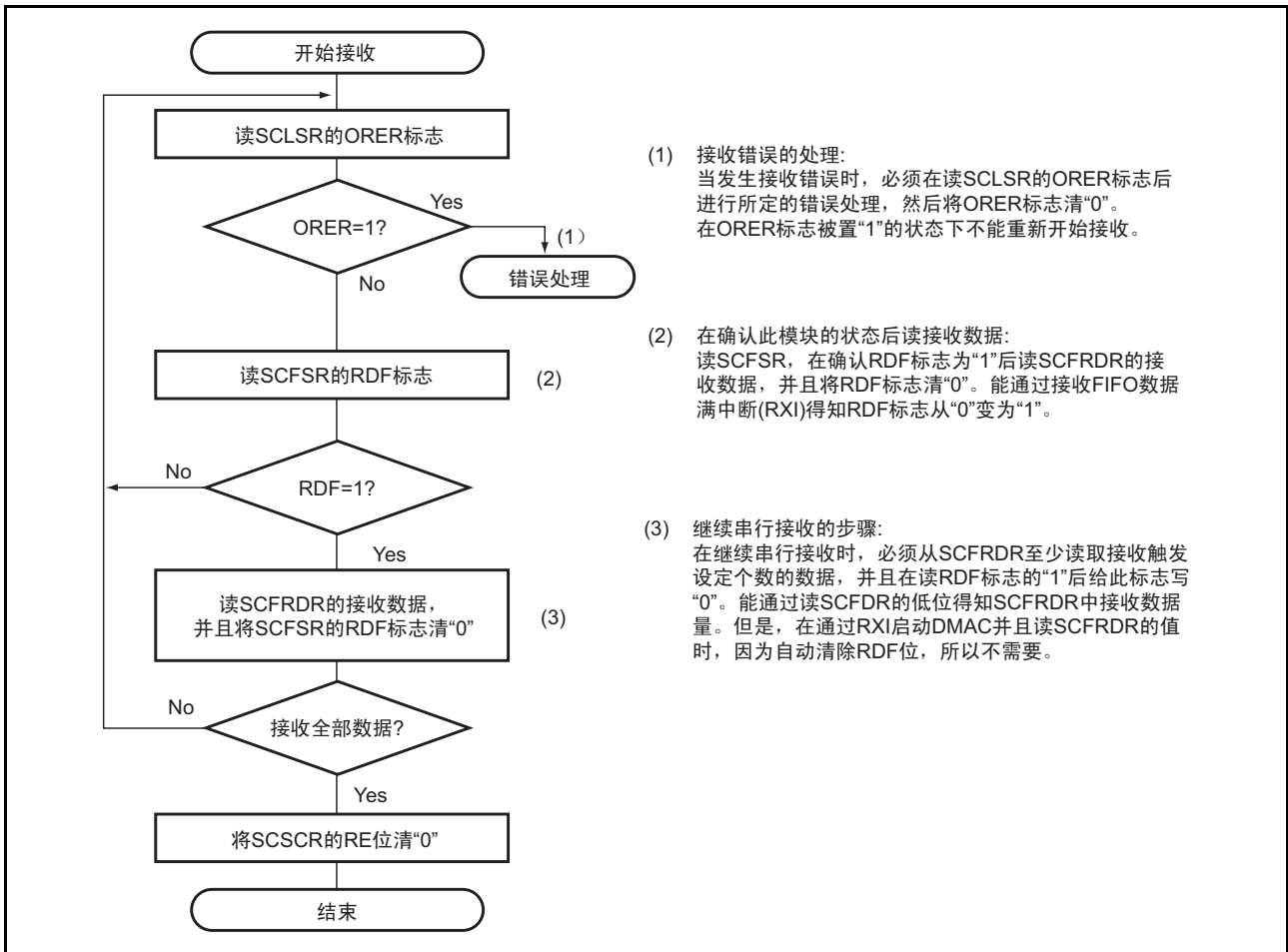


图 16.15 串行接收的流程图例子（1）

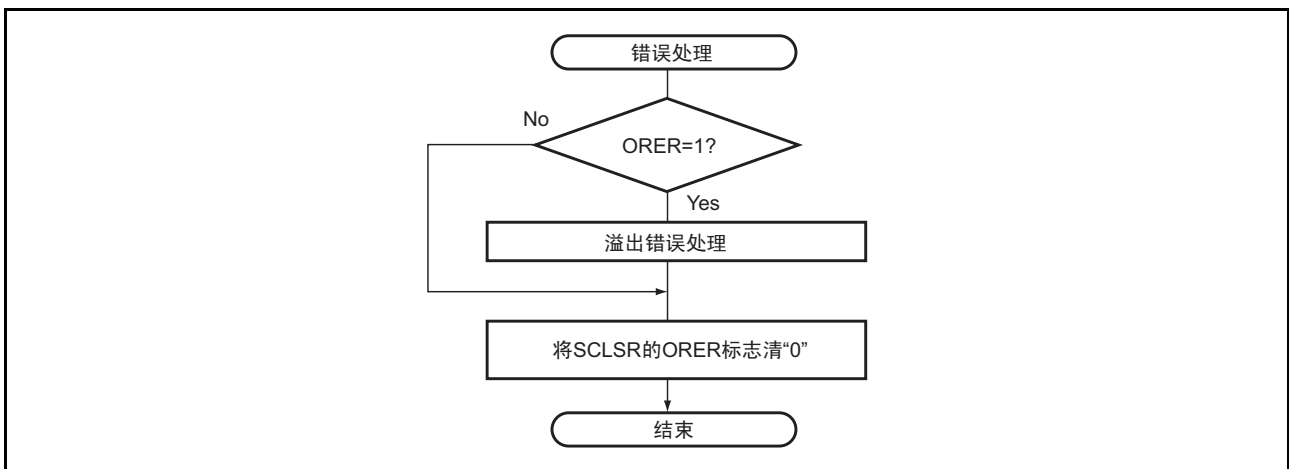


图 16.16 串行接收的流程图例子（2）

串行接收时的运行如下：

1. 和同步时钟的输入或者输出同步，开始接收。
2. 按照接收移位寄存器（SCRSR）的 LSB 到 MSB 的顺序保存接收数据。在接收后，检查是否为能将接收数据从 SCRSR 传送到 SCFRDR 的状态。在此检查通过后，将 RDF 标志置“1”并且将接收数据保存到 SCFRDR。如果在错误检查时检测到溢出错误，就不能进行以后的接收。
3. 在 RDF 标志变为“1”时，如果串行控制寄存器（SCSCR）的 RIE 位已被置“1”，就产生接收 FIFO 数据满中断（RXI）请求；在 ORER 标志变为“1”时，如果 SCSCR 的 RIE 位或者 REIE 位已被置“1”，就产生中止的中断（BRI）请求。

接收的运行例子如图 16.17。

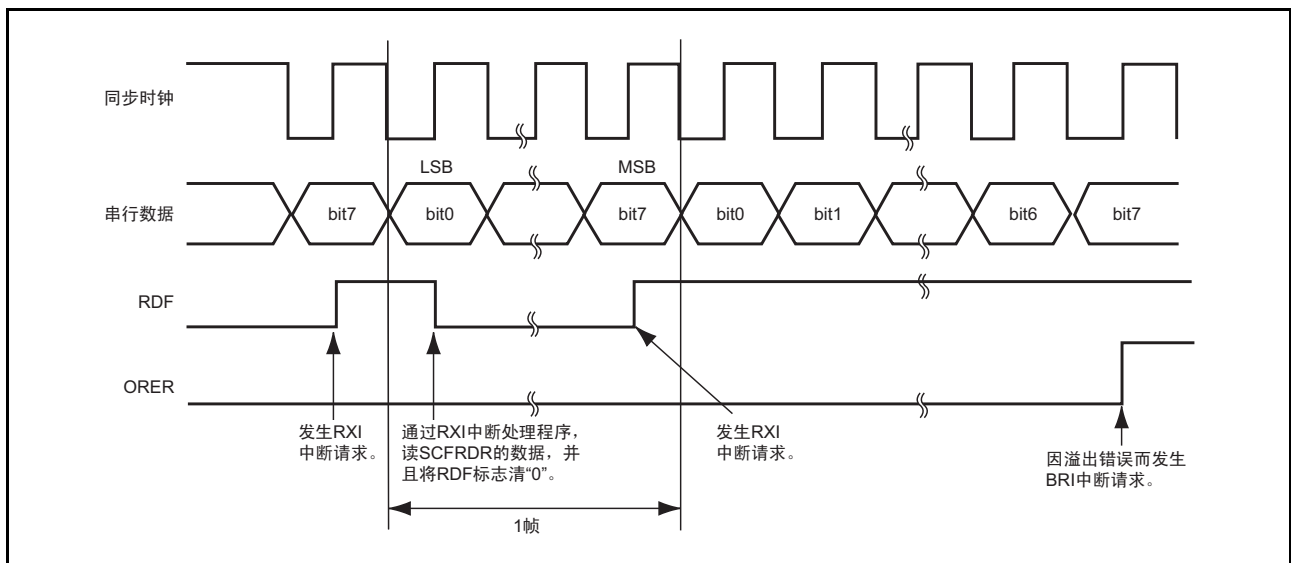


图 16.17 接收的运行例子

• 串行数据的同时发送和接收（时钟同步模式）

同时串行发送和接收的流程图例子如图 16.18 所示。

必须在设定为可发送和接收的状态后按照以下步骤同时发送和接收串行数据。

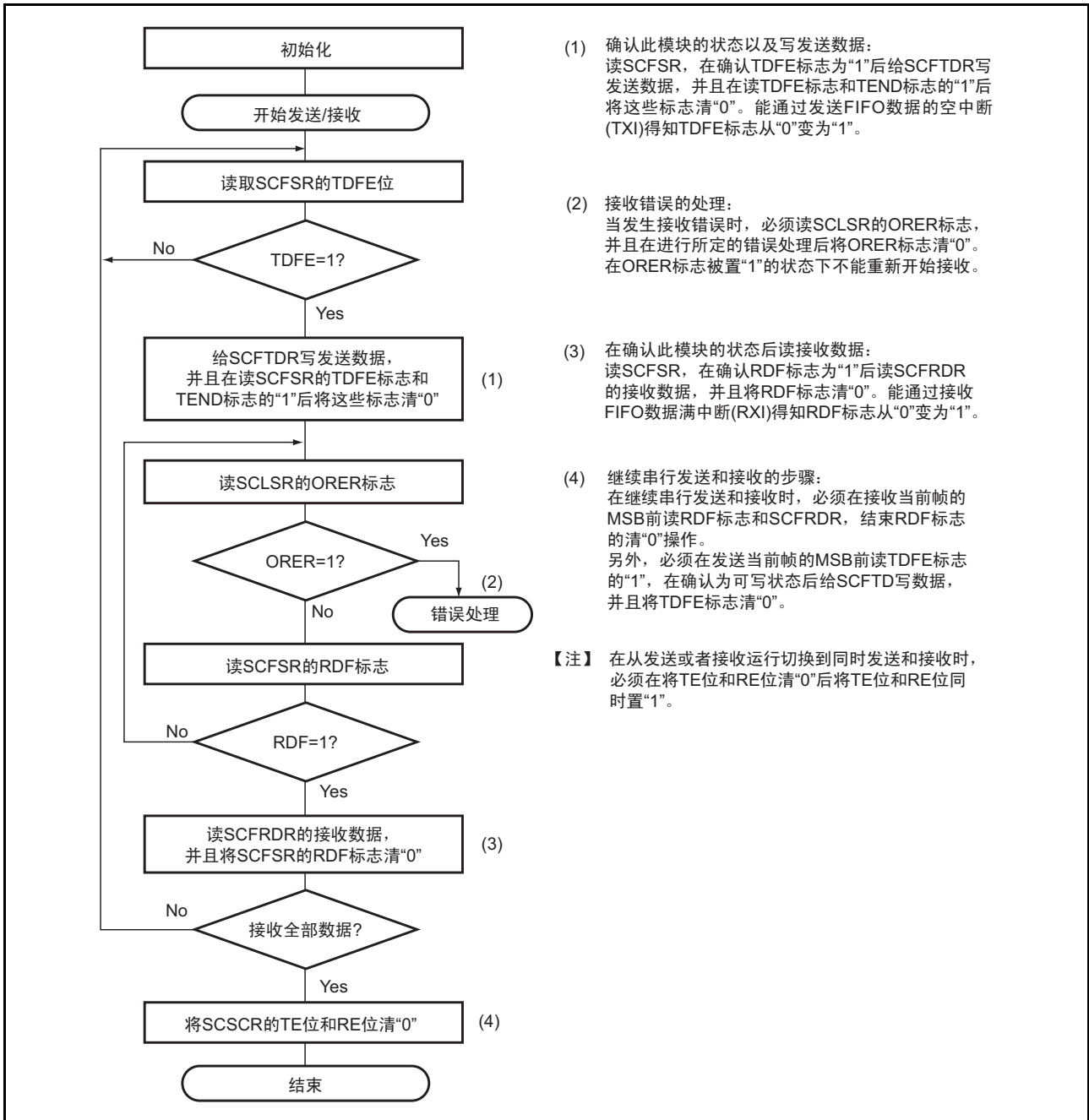


图 16.18 串行数据发送和接收的流程图例子

16.5 中断

此模块有发送 FIFO 数据空中断 (TXI) 请求、接收错误中断 (ERI) 请求、接收 FIFO 数据满中断 (RXI) 请求和中止中断 (BRI) 请求共 4 种中断源。

各中断源和优先级如表 16.12 所示。能通过 SCSCR 的 TIE 位、RIE 位、REIE 位允许或者禁止各中断源。另外，分别将各中断请求独立传送到中断控制器。

如果在通过 TIE 位允许 TXI 中断的情况下将串行状态寄存器 (SCFSR) 的 TDFE 标志置“1”，就产生 TXI 中断。能通过此 TXI 中断请求启动直接存储器存取控制器，进行数据传送。此时，不向 CPU 发出中断请求。

如果在通过 RIE 位允许 RXI 中断的情况下将 SCFSR 的 RDF 标志或者 DR 标志置“1”，就产生 RXI 中断请求。能通过此 RXI 中断请求启动直接存储器存取控制器，进行数据传送。此时，不向 CPU 发出中断请求。只在异步模式中才能通过将 DR 标志置“1”产生 RXI 中断请求。

如果将 SCSCR 的 RIE 位置“0”并且将 REIE 位置“1”，就能不产生 RXI 中断请求而只产生 ERI 中断请求或者 BRI 中断请求。

另外，TXI 表示能写发送数据，RXI 表示 SCFRDR 中有接收数据。

表 16.12 中断源

中断源	内容	直接存储器存取控制器的启动	解除复位时的优先级
BRI	中止 (BRK) 或者溢出 (ORER) 引起的中断	不能	高 ▲ ▼ 低
ERI	接收错误 (ER) 引起的中断	不能	
RXI	接收 FIFO 数据满 (RDF) 或者数据就绪 (DR) 引起的中断	能	
TXI	发送 FIFO 数据空 (TDFE) 引起的中断	能	

16.6 使用时的注意事项

在使用此模块时，必须注意以下几点。

16.6.1 SCFTDR 的写操作和 TDFE 标志

如果写到发送 FIFO 数据寄存器 (SCFTDR) 的发送数据字节数小于 FIFO 控制寄存器 (SCFCR) 的 TTRG[1:0] 设定的发送触发个数，就将串行状态寄存器 (SCFSR) 的 TDFE 标志置位。在 TDFE 标志被置位后，能在 SCFTDR 变空之前写发送数据，进行高效率的连续发送。

但是，如果写到 SCFTDR 的数据字节数小于等于发送触发个数，即使在读 TDFE 标志的“1”后清“0”，TDFE 标志也将被再次置“1”。因此，必须在写到 SCFTDR 的数据字节数超过发送触发个数时读 TDFE 标志的“1”后清“0”。

能通过 FIFO 数据计数寄存器 (SCFDR) 的高 8 位得知 SCFTDR 的发送数据字节数。

16.6.2 SCFRDR 的读操作和 RDF 标志

如果接收 FIFO 数据寄存器 (SCFRDR) 的接收数据字节数大于等于 FIFO 控制寄存器 (SCFCR) 的 RTRG[1:0] 设定的接收触发个数，就将串行状态寄存器 (SCFSR) 的 RDF 标志置位。在 RDF 标志被置位后，能从 SCFRDR 读触发个数的接收数据，进行高效率的连续接收。

但是，如果 SCFRDR 的数据字节数超过接收触发个数，即使在读 RDF 标志的“1”后清“0”，RDF 标志也将被再次置“1”。因此，必须读接收数据，使接收 FIFO 数据寄存器 (SCFRDR) 内的数据量小于触发个数，然后在读 RDF 标志的“1”后清“0”。能从 FIFO 数据计数寄存器 (SCFDR) 低 8 位得知 SCFRDR 的接收数据字节数。

16.6.3 使用直接存储器存取控制器时的注意事项

如果通过 TXI 中断请求由直接存储器存取控制器将数据写到 SCFTDR，TEND 标志的状态就为不定值。因此，此时不能将 TEND 标志用作传送结束标志。

16.6.4 中止的检测和处理

在检测帧错误 (FER) 时，能通过直接读 RxD 引脚的值检测到中止。在中止时，RxD 引脚的输入始终为“0”，因此将 FER 标志置位并且也可能将奇偶校验错误标志 (PER) 置位。

如果检测到中止，就停止向 SCFRDR 传送接收数据，但是能继续接收。

16.6.5 中止的发送

能通过串行端口寄存器 (SCSPTR) 的 SPB2IO 位和 SPB2DR 位决定 TxD 引脚的输入 / 输出的条件和电平，因此能利用这一功能发送中止。

从进行串行发送的初始化到将 TE 位置“1” (能发送)，TxD 引脚不工作。在此期间，标记状态被 SPB2DT 位的值替代。因此，要先将 SPB2IO 和 SPB2DT 位置“1” (输出高电平)。

为了在串行发送时发送中止信号，必须在将 SPB2DT 位清“0” (指定低电平) 后将 TE 位清“0” (停止发送)。如果将 TE 位清“0”，就对发送部进行初始化并且从 TxD 引脚输出“0”，与当前的发送状态无关。

16.6.6 异步模式的接收数据采样时序和接收容限

此模块以 16 倍或者 8 倍位速率的基本时钟运行。在接收时，通过基本时钟对起始位的下降沿进行采样，取得内部的同步，而在基本时钟的第 8/4 个时钟脉冲的上升沿将接收数据取到内部。以 16 倍位速率的基本时钟运行时的时序如图 16.19 所示。

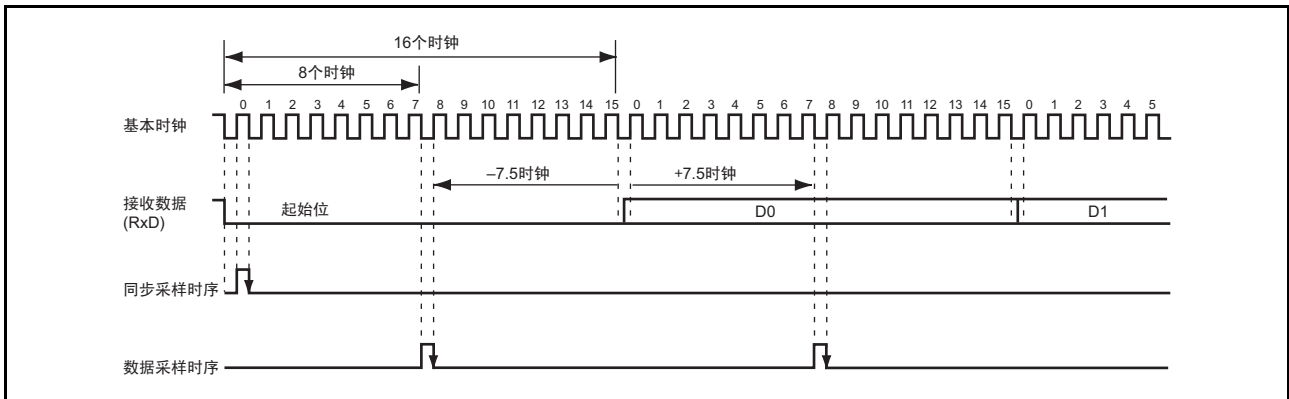


图 16.19 异步模式中的接收数据采样时序（以 16 倍位速率的基本时钟运行）

因此，能用计算式 (1) 表示异步模式的接收容限。

$$M = \left| \left(0.5 - \frac{1}{2N} \right) - (L - 0.5)F - \frac{|D - 0.5|}{N} (1 + F) \right| \times 100[\%] \quad \cdots\cdots\text{计算式 (1)}$$

M: 接收容限 (%)

N: 时钟频率和位速率的比 (N=16/8)

D: 时钟占空比 (D:0 ~ 1.0)

L: 帧长 (L=9 ~ 12)

F: 时钟频率的偏差绝对值

如果计算式 (1) 中 F=0、D=0.5 并且 N=16，根据计算式 (2)，接收容限为 46.875%。

当 D=0.5 并且 F=0 时，

$$M = (0.5 - 1/(2 \times 16)) \times 100\% = 46.875\% \quad \cdots\cdots\text{计算式 (2)}$$

但是，此值为计算值，在系统设计时必须留有 20 ~ 30% 的余地。

16.6.7 异步基本时钟的选择

本 LSI 能通过更改串行扩展模式寄存器 (SCEMR) 的 ABCS 位，将异步模式中的 1 位期间的基本时钟频率设定为位速率的 16 倍或者 /8 倍。

但是，必须注意：由“16.6.6 异步模式的接收数据采样时序和接收容限”的计算式 (1) 可知，如果基本时钟的频率低于位速率的 8 倍，就会减低接收容限。

如果只通过串行模式寄存器 (SCSMR) 的 CKS[1:0] 和位速率寄存器 (SCBRR) 就能设定期望的位速率，建议将 1 位期间的基本时钟频率设定为位速率的 16 倍 (SCEMR 的 ABCS 位为“0”)。如果时钟源为内部时钟并且不使用 SCK 引脚，就能通过设定为波特率发生器倍速模式 (SCEMR 的 BGD M 位为“1”)，不降低接收容限而提高位速率。

17. 瑞萨串行外围接口

本 LSI 有独立的 2 个通道的瑞萨串行外围接口。
此模块能进行全双工同步串行通信。

17.1 特点

本 LSI 的此模块有以下特点：

- **SPI 传送功能**
 - 能使用 MOSI (Master Out Slave In)、MISO (Master In Slave Out)、SSL (Slave Select)、RSPCK (SPI Clock) 信号，通过 SPI 运行 (4 线式) 进行串行通信。
 - 能在主控模式或者从属模式中进行串行通信。
 - 能进行模式故障错误检测 (只限于设定为 SPI 从属模式的情况)。
 - 能进行溢出错误检测 (只限于设定为 SPI 从属模式的情况)。
 - 能更改串行传送时钟的极性。
 - 能更改串行传送时钟的相位。
- **数据格式**
 - 能切换 MSB first 和 LSB first。
 - 能将传送位长改为 8 位、16 位或者 32 位。
- **位速率**
 - 主控模式时的 RSPCK 最大为 4096 分频。
 - 通过内部波特率发生器将 P1 ϕ 分频后生成 RSPCK。
 - 将外部输入时钟用作串行时钟。
- **缓冲器结构**
 - 发送缓冲器为 8Byte，接收缓冲器为 32Byte。
- **SSL 控制功能**
 - 每个通道有 1 个 SSL 信号。
 - 在设定为主控模式时，输出 SSL 信号。
 - 在设定为从属模式时，输入 SSL 信号。
 - 能设定从 SSL 输出有效到 RSPCK 运行为止的延迟 (RSPCK 延迟)。
 - 设定范围：1~8 个 RSPCK 设定单位：1 个 RSPCK
 - 能设定从 RSPCK 停止到 SSL 输出无效为止的延迟 (SSL 无效延迟)。
 - 设定范围：1~8 个 RSPCK 设定单位：1 个 RSPCK
 - 能设定下次存取的 SSL 输出有效的等待 (下次存取延迟)。
 - 设定范围：1~8 个 RSPCK 设定单位：1 个 RSPCK
- **SSL 极性变更功能**
- **主控传送时的控制方式**
 - 能按顺序循环执行最多由 4 个命令构成的传送。
 - 能给各命令设定以下项目：
 - SSL 信号值、位速率、RSPCK 极性/相位、传送数据长度、LSB first/MSB first、突发、RSPCK 延迟、SSL 无效延迟、下次存取延迟。
 - 能通过写发送缓冲器来启动传送。
 - 能通过清除 SPTEF 位来启动传送。
 - 能设定 SSL 无效时的 MOSI 信号值。

- 中断源
 - 可屏蔽的中断源
 - 接收中断（接收缓冲器满）
 - 发送中断（发送缓冲器空）
 - 错误中断（模式故障、溢出）
- 其他
 - 环回模式
 - 禁止（初始化）功能

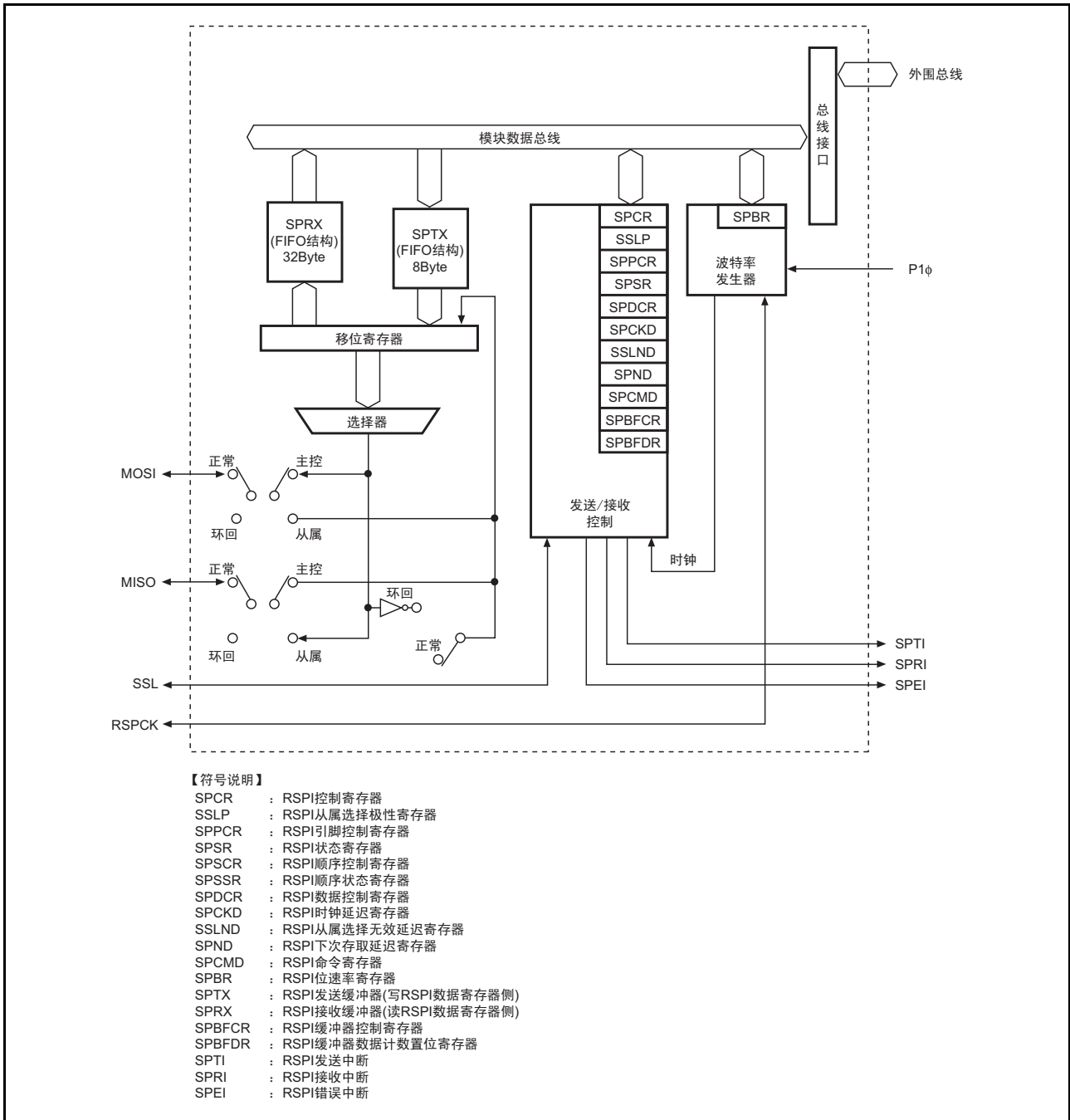


图 17.1 框图（1 个通道）

17.2 输入 / 输出引脚

引脚结构如表 17.1 所示。在设定为主控模式时，此模块自动将 SSL 引脚的输入 / 输出方向切换为输出；在设定为从属模式时，此模块自动将 SSL 引脚的输入 / 输出方向切换为输入。根据主控模式 / 从属模式的设定和 SSL 的输入电平，此模块自动切换 RSPCK、MOSI 和 MISO 的输入 / 输出方向（参照“17.4.2 引脚的控制”）。

表 17.1 引脚结构

通道	名称	引脚名	输入 / 输出	功能
0	时钟引脚	RSPCK0	输入 / 输出	时钟输入 / 输出
	主控发送数据引脚	MOSI0	输入 / 输出	主控发送数据
	从属发送数据引脚	MISO0	输入 / 输出	从属发送数据
	从属选择 0 引脚	SSL00	输入 / 输出	从属选择
1	时钟引脚	RSPCK1	输入 / 输出	时钟输入 / 输出
	主控发送数据引脚	MOSI1	输入 / 输出	主控发送数据
	从属发送数据引脚	MISO1	输入 / 输出	从属发送数据
	从属选择 0 引脚	SSL10	输入 / 输出	从属选择

【注】 本章节中省略通道，略称为 RSPCK、MOSI、MISO 和 SSL。

17.3 寄存器说明

寄存器结构如表17.2所示。能通过这些寄存器指定主控模式/从属模式和传送格式以及控制发送部/接收部。

表 17.2 寄存器结构

通道	名称	略称 *1	R/W	初始值	地址	存取长度
0	控制寄存器 _0	SPCR_0	R/W	H'00	H'E800E000	8、16
	从属选择极性寄存器 _0	SSLP_0	R/W	H'00	H'E800E001	8、16
	引脚控制寄存器 _0	SPPCR_0	R/W	H'00	H'E800E002	8、16
	状态寄存器 _0	SPSR_0	R/(W)*2	H'60	H'E800E003	8、16
	数据寄存器 _0	SPDR_0	R/W	不定值	H'E800E004	8、16、32
	顺序控制寄存器 _0	SPSCR_0	R/W	H'00	H'E800E008	8、16
	顺序状态寄存器 _0	SPSSR_0	R	H'00	H'E800E009	8、16
	位速率寄存器 _0	SPBR_0	R/W	H'FF	H'E800E00A	8、16
	数据控制寄存器 _0	SPDCR_0	R/W	H'20	H'E800E00B	8、16
	时钟延迟寄存器 _0	SPCKD_0	R/W	H'00	H'E800E00C	8、16
	从属选择无效延迟寄存器 _0	SSLND_0	R/W	H'00	H'E800E00D	8、16
	下次存取延迟寄存器 _0	SPND_0	R/W	H'00	H'E800E00E	8
	命令寄存器 _00	SPCMD_00	R/W	H'070D	H'E800E010	16
	命令寄存器 _01	SPCMD_01	R/W	H'070D	H'E800E012	16
	命令寄存器 _02	SPCMD_02	R/W	H'070D	H'E800E014	16
	命令寄存器 _03	SPCMD_03	R/W	H'070D	H'E800E016	16
	缓冲控制寄存器 _0	SPBFCR_0	R/W	H'00	H'E800E020	8、16
	缓冲器数据计数置位寄存器 _0	SPBFDR_0	R	H'0000	H'E800E022	16
	1	控制寄存器 _1	SPCR_1	R/W	H'00	H'E800E800
从属选择极性寄存器 _1		SSLP_1	R/W	H'00	H'E800E801	8、16
引脚控制寄存器 _1		SPPCR_1	R/W	H'00	H'E800E802	8、16
状态寄存器 _1		SPSR_1	R/(W)*2	H'60	H'E800E803	8、16
数据寄存器 _1		SPDR_1	R/W	不定值	H'E800E804	16、32
顺序控制寄存器 _1		SPSCR_1	R/W	H'00	H'E800E808	8、16
顺序状态寄存器 _1		SPSSR_1	R	H'00	H'E800E809	8、16
位速率寄存器 _1		SPBR_1	R/W	H'FF	H'E800E80A	8、16
数据控制寄存器 _1		SPDCR_1	R/W	H'20	H'E800E80B	8、16
时钟延迟寄存器 _1		SPCKD_1	R/W	H'00	H'E800E80C	8、16
从属选择无效延迟寄存器 _1		SSLND_1	R/W	H'00	H'E800E80D	8、16
下次存取延迟寄存器 _1		SPND_1	R/W	H'00	H'E800E80E	8
命令寄存器 _10		SPCMD_10	R/W	H'070D	H'E800E810	16
命令寄存器 _11		SPCMD_11	R/W	H'070D	H'E800E812	16
命令寄存器 _12		SPCMD_12	R/W	H'070D	H'E800E814	16
命令寄存器 _13		SPCMD_13	R/W	H'070D	H'E800E816	16
缓冲控制寄存器 _1		SPBFCR_1	R/W	H'00	H'E800E820	8、16
缓冲器数据计数置位寄存器 _1		SPBFDR_1	R	H'0000	H'E800E822	16

【注】 *1 本章节中的寄存器名省略通道。

*2 为了清除标志，只能写“0”。

17.3.1 控制寄存器 (SPCR)

SPCR 是设定运行模式的寄存器。如果在 SPE 位为“1”并且允许此模块功能的状态下改写 MSTR 位和 MODFEN 位的设定值，就不保证以后的运行。

位:	7	6	5	4	3	2	1	0
	SPRIE	SPE	SPTIE	SPEIE	MSTR	MOD FEN	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R

位	位名	初始值	R/W	说明
7	SPRIE	0	R/W	接收中断允许 当接收缓冲器 (SPRX) 的接收数据量大于等于指定的接收触发数并且状态寄存器 (SPSR) 的 SPRF 标志被置“1”时，设定允许或者禁止接收中断 (SPRI) 请求的产生。 0: 禁止接收中断请求的产生 1: 允许接收中断请求的产生
6	SPE	0	R/W	功能允许 通过将此位置“1”，使此模块功能变为有效。当状态寄存器 (SPSR) 的 MODF 位为“1”时，不能将 SPE 位置“1” (参照“17.4.6 错误检测”)。 如果将 SPE 位置“0”，就将此模块功能置为无效并且对一部分模块功能进行初始化 (参照“17.4.7 初始化”)。 0: 将此模块的功能置为无效 1: 将此模块的功能置为有效
5	SPTIE	0	R/W	发送中断允许 当发送缓冲器 (SPTX) 的发送数据量小于等于指定的发送触发数并且状态寄存器 (SPSR) 的 SPTEF 标志被置“1”时，设定允许或者禁止发送中断 (SPTI) 请求的产生。 0: 禁止发送中断请求的产生 1: 允许发送中断请求的产生
4	SPEIE	0	R/W	错误中断允许 在此模块检测到模式故障错误后将状态寄存器 (SPSR) 的 MODF 位置“1”或者在此模块检测到溢出错误后将 SPSR 的 OVRF 位置“1”时，此位允许或者禁止错误中断请求的产生 (参照“17.4.6 错误检测”)。 0: 禁止错误中断请求的产生 1: 允许错误中断请求的产生 【注】 此位只在设定为 SPI 从属模式时有效。
3	MSTR	0	R/W	主控 / 从属模式选择 此位选择主控模式或者从属模式。此模块根据 MSTR 位的设定，决定 RSPCK、MOSI、MISO 和 SSL 引脚的方向。 0: 从属模式 1: 主控模式
2	MODFEN	0	R/W	模式故障错误检测允许 此位允许或者禁止模式故障错误检测 (参照“17.4.6 错误检测”)。 0: 禁止模式故障错误检测 1: 允许模式故障错误检测 【注】 此位只在设定为 SPI 从属模式时有效。 在将 MSTR 位设定为主控模式时，必须将 MODFEN 位置“0”。
1、0	—	全 0	R	保留位 写入值总是“0”。不保证写“1”时的运行。

17.3.2 从属选择极性寄存器 (SSLP)

SSLP 是设定 SSL 信号极性的寄存器。如果在控制寄存器 (SPCR) 的 SPE 位为 “1” 并且允许此模块功能的状态下改写 SSL0P 位，就不保证以后的运行。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SSL0P
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
7 ~ 1	—	全 0	R	保留位 写入值总是 “0”。不保证写 “1” 时的运行。
0	SSL0P	0	R/W	SSL 信号极性设定 此位设定 SSL 信号的极性。SSL0P 的设定值表示 SSL 信号的有效极性。 0: SSL 信号为低电平有效 1: SSL 信号为高电平有效

17.3.3 引脚控制寄存器 (SPPCR)

SPPCR 是设定引脚模式的寄存器。如果在 SPCR 的 SPE 位为“1”并且允许此模块功能的状态下改写 SPPCR，就不保证以后的运行。

位:	7	6	5	4	3	2	1	0
	—	—	MOIFE	MOIFV	—	—	—	SPLP
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R	R/W

位	位名	初始值	R/W	说明
7、6	—	全 0	R	保留位 写入值总是“0”。不保证写“1”时的运行。
5	MOIFE	0	R/W	MOSI 空闲值固定允许 在主控模式的 SSL 无效期间（包括突发传输中的 SSL 保持期间），此模块使用此位固定 MOSI 输出值。当 MOIFE 为“0”时，此模块在 SSL 无效期间将上次串行传输的最后数据输出到 MOSI；当 MOIFE 为“1”时，此模块将 MOIFV 位设定的固定值输出到 MOSI。 0: MOSI 输出值为上次传输的最后数据 1: MOSI 输出值为 MOIFV 位的设定值
4	MOIFV	0	R/W	MOSI 空闲固定值 在主控模式中，当 MOIFE 位为“1”时，此模块根据 MOIFV 位的设定，决定 SSL 无效期间（包括突发传输中的 SSL 保持期间）的 MOSI 信号值。 0: MOSI 空闲固定值为“0” 1: MOSI 空闲固定值为“1”
3~1	—	全 0	R	保留位 写入值总是“0”。不保证写“1”时的运行。
0	SPLP	0	R/W	环回 如果将 SPLP 位置“1”，此模块就切断 MISO 引脚和移位寄存器之间以及 MOSI 引脚和移位寄存器之间的线路，连接移位寄存器的输入线路和输出线路（反相）。 0: 正常模式 1: 环回模式

17.3.4 状态寄存器 (SPSR)

SPSR 保存表示运行状态的标志。

位:	7	6	5	4	3	2	1	0
	SPRF	TEND	SPTEF	—	—	MODF	—	OVRF
初始值:	0	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/(W)*	R	R/(W)*

【注】* 为了清除标志，只能在读“1”后写“0”。

位	位名	初始值	R/W	说明
7	SPRF	0	R	接收缓冲器满标志 此位表示接收缓冲器 (SPRX) 的接收数据量大于等于缓冲控制寄存器 (SPBFCR) 的接收缓冲器的数据量触发 (RXTRG) 指定的值。 0: 接收缓冲器的接收数据量小于接收缓冲器的数据量触发指定的值 1: 接收缓冲器的接收数据量大于等于接收缓冲器的数据量触发指定的值 SPRF 位清 “0” 的条件如下: <ul style="list-style-type: none"> • 读接收缓冲器，直到接收缓冲器的接收数据量小于接收缓冲器的数据量触发指定的值时 • 当将接收缓冲器的数据复位置为有效时 • 上电复位 SPRF 位置 “1” 的条件如下: <ul style="list-style-type: none"> • 当接收缓冲器的接收数据量大于等于接收缓冲器的数据量触发指定的值时
6	TEND	1	R	发送结束 此位表示发送结束。当 TEND 位为 “1” 时，表示发送结束；当 TEND 位为 “0” 时，表示发送未结束。 TEND 位清 “0” 的条件如下: <ul style="list-style-type: none"> • 当将发送数据从发送寄存器传送到移位寄存器时 TEND 位置 “1” 的条件如下: <ul style="list-style-type: none"> • 在串行传送结束时发送缓冲器 (SPTX) 的数据保存个数为空
5	SPTEF	1	R	发送缓冲器空标志 此位表示发送缓冲器 (SPTX) 的发送数据量小于等于缓冲控制寄存器 (SPBFCR) 的发送缓冲器的数据量触发 (TXTRG) 指定的值。 0: 发送缓冲器的发送数据量大于等于发送缓冲器的数据量触发指定的值 1: 发送缓冲器的发送数据量小于发送缓冲器的数据量触发指定的值 SPTEF 位清 “0” 的条件如下: <ul style="list-style-type: none"> • 写发送缓冲器，直到发送缓冲器的发送数据量大于发送缓冲器的数据量触发指定的值时 SPTEF 位置 “1” 的条件如下: <ul style="list-style-type: none"> • 当发送缓冲器的发送数据量小于发送缓冲器的数据量触发指定的值时 • 当将发送缓冲器的数据复位置为有效时 • 上电复位
4、3	—	全 0	R	保留位 写入值总是 “0”。不保证写 “1” 时的运行。

位	位名	初始值	R/W	说明
2	MODF	0	R/(W)*	<p>模式故障错误标志</p> <p>此位表示模式故障错误的发生情况。如果此模块在从属模式中将 MODFEN 位置“1”并且在数据传送所需的 RSPCK 周期结束前将 SSL 引脚置为无效，此模块就检测到模式故障错误。SSL 信号的有效电平取决于从属选择极性寄存器（SSLP）的 SSL0P 位。MODF 位的清除条件如下：</p> <ul style="list-style-type: none"> • 在 MODF 被置“1”的状态下读 SPSR 后给 MODF 写“0”时 • 上电复位 <p>0: 未发生模式故障错误 1: 发生模式故障错误</p> <p>【注】 此位只在设定为 SPI 从属模式时有效。</p>
1	—	0	R	<p>保留位</p> <p>写入值总是“0”。不保证写“1”时的运行。</p>
0	OVRF	0	R/(W)*	<p>溢出错误标志</p> <p>此位表示溢出错误的发生状况。如果在接收缓冲器（SPRX）没有接收数据长度空间的状态下结束串行传送，此模块就检测到溢出错误并且将 OVRF 位置“1”。</p> <p>OVRF 位的清除条件如下：</p> <ul style="list-style-type: none"> • 在 OVRF 被置“1”的状态下读 SPSR 后给 OVRF 写“0”时 • 上电复位 <p>0: 未发生溢出错误 1: 发生溢出错误</p> <p>【注】 此位只在设定为 SPI 从属模式时有效。</p>

【注】 * 为了清除标志，只能在读“1”后写“0”。

17.3.5 数据寄存器 (SPDR)

SPDR 是保存发送 / 接收数据的缓冲器。

发送缓冲器 (SPTX) 和接收缓冲器 (SPRX) 是独立的缓冲器, 这些缓冲器被映像到 SPDR。

必须通过设定数据控制寄存器 (SPDCR) 的存取宽度设定位 (SPLW), 用字节、字或者长字读写 SPDR。

SPDR 使用的位长取决于命令寄存器 (SPCMD) 的数据长度设定位 (SPB3 ~ 0)。

如果发送缓冲器有 SPDR 存取宽度的空间, 就将数据从 SPDR 写到发送缓冲器, 否则就不写数据。即使写数据也被忽视。

一旦从 SPDR 读数据, 就能读到接收缓冲器中的接收数据。如果在接收缓冲器没有接收数据的状态下读数据, 读取值为不定值。

在以长字、字或者字节的存取宽度写 SPDR 时, 必须将发送数据写到以下的位, 否则不保证所写的的数据。

- 长字: 31 ~ 0 位
- 字: 31 ~ 16 位
- 字节: 31 ~ 24 位

在以长字、字或者字节的存取宽度读 SPDR 时, 必须从以下的位读接收数据, 否则所读的数据为不定值。

- 长字: 31 ~ 0 位
- 字: 31 ~ 16 位
- 字节: 31 ~ 24 位

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24	SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

17.3.6 顺序控制寄存器 (SPSCR)

SPSCR 设定此模块主控运行时的顺序控制方式。如果在控制寄存器 (SPCR) 的 MSTR 位和 SPE 位都为“1”并且在主控模式中允许此模块功能的状态下改写 SPSCR，就不保证以后的运行。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPS LN1	SPS LN0
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明															
7 ~ 2	—	全 0	R	保留位 写入值总是“0”。不保证写“1”时的运行。															
1	SPSLN1	0	R/W	顺序长度设定 在主控模式中，此位设定此模块进行顺序运行时的顺序长度。在主控模式中，此模块根据 SPSLN1 和 SPSLN0 设定的顺序长度，更改要参照的命令寄存器 0 ~ 3 (SPCMD0 ~ 3) 和参照顺序。SPSLN1 和 SPSLN0 的设定值与顺序长度、此模块要参照的 SPCMD0 ~ 3 的关系如下。另外，在从属模式中，此模块能随时参照 SPCMD0。 <table style="margin-left: 20px; border: none;"> <tr> <td style="padding-right: 20px;">顺序长度</td> <td style="padding-right: 20px;">要参照的 SPCMD 寄存器 (序号)</td> <td></td> </tr> <tr> <td>00:</td> <td>1</td> <td>0→0→……</td> </tr> <tr> <td>01:</td> <td>2</td> <td>0→1→0→……</td> </tr> <tr> <td>10:</td> <td>3</td> <td>0→1→2→0→……</td> </tr> <tr> <td>11:</td> <td>4</td> <td>0→1→2→3→0→……</td> </tr> </table>	顺序长度	要参照的 SPCMD 寄存器 (序号)		00:	1	0→0→……	01:	2	0→1→0→……	10:	3	0→1→2→0→……	11:	4	0→1→2→3→0→……
顺序长度	要参照的 SPCMD 寄存器 (序号)																		
00:	1	0→0→……																	
01:	2	0→1→0→……																	
10:	3	0→1→2→0→……																	
11:	4	0→1→2→3→0→……																	
0	SPSLN0	0	R/W																

17.3.7 顺序状态寄存器 (SPSSR)

SPSSR 表示此模块主控运行时的顺序控制状态。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPCP1	SPCP0
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明								
7 ~ 2	—	全 0	R	保留位 写入值总是“0”。不保证写“1”时的运行。								
1	SPCP1	0	R	命令指针 在顺序控制过程中，此位表示当前指针指向的命令寄存器 0 ~ 3 (SPCMD0 ~ 3)。 SPCP1 和 SPCP0 的值与 SPCMD0 ~ SPCMD3 的关系如下。 有关顺序控制，请参照“17.4.8(1) (c) 顺序控制”。 <table style="margin-left: 20px; border: none;"> <tr> <td>00:</td> <td>SPCMD0</td> </tr> <tr> <td>01:</td> <td>SPCMD1</td> </tr> <tr> <td>10:</td> <td>SPCMD2</td> </tr> <tr> <td>11:</td> <td>SPCMD3</td> </tr> </table>	00:	SPCMD0	01:	SPCMD1	10:	SPCMD2	11:	SPCMD3
00:	SPCMD0											
01:	SPCMD1											
10:	SPCMD2											
11:	SPCMD3											
0	SPCP0	0	R									

17.3.8 位速率寄存器 (SPBR)

SPBR 用于设定主控模式中的位速率。如果在控制寄存器 (SPCR) 的 MSTR 位和 SPE 位都为“1”并且在主控模式中允许此模块功能的状态下改写 SPBR，就不保证以后的运行。

位:	7	6	5	4	3	2	1	0
	SPR7	SPR6	SPR5	SPR4	SPR3	SPR2	SPR1	SPR0
初始值:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

在从属模式中使用此模块时，位速率与 SPBR 和 BRDV 的设定无关，取决于输入时钟的位速率。

位速率取决于 SPBR 的设定值和命令寄存器 (SPCMD0 ~ 3) 的 BRDV1 ~ 0 位的设定值的组合。位速率的计算式如下。在计算式中，n 为 SPBR 的设定值 (0、1、2、……、255)，N 为 BRDV1 ~ 0 位的设定值 (0、1、2、3)。

$$\text{位速率} = \frac{f(P1\phi)}{2 \times (n + 1) \times 2^N}$$

SPBR 寄存器、BRDV1 ~ 0 位的设定值和位速率的关系例子如表 17.3 所示。

表 17.3 SPBR 寄存器、BRDV1 ~ 0 位的设定值和位速率

SPBR 的设定值 (n)	BRDV1 ~ 0 的 设定值 (N)	分频比	位速率		
			P1φ=50MHz	P1φ=60MHz	P1φ=66.67MHz
0	0	2	25.0Mbps	30.0Mbps	33.33Mbps
1	0	4	12.5Mbps	15.0Mbps	16.67Mbps
2	0	6	8.33Mbps	10.0Mbps	11.11Mbps
3	0	8	6.25Mbps	7.50Mbps	8.33Mbps
4	0	10	5.00Mbps	6.00Mbps	6.67Mbps
5	0	12	4.16Mbps	5.00Mbps	5.56Mbps
5	1	24	2.08Mbps	2.50Mbps	2.78Mbps
5	2	48	1.04Mbps	1.25kbps	1.39Mbps
5	3	96	520kbps	625kbps	694.48kbps
255	3	4096	12.20kbps	14.64kbps	16.28kbps

17.3.9 数据控制寄存器 (SPDCR)

SPDCR 将 SPDR 寄存器的存取宽度设定为字节存取、长字存取或者字存取，并且在将此模块设定为主控模式时设定虚拟数据发送允许 (TXDMY)。

如果在状态寄存器 (SPCR) 的 TEND 位为发送未结束的状态下改写 SPDCR，就不保证以后的运行。

位:	7	6	5	4	3	2	1	0
	TXDMY	SPLW1	SPLW0	—	—	—	—	—
初始值:	0	0	1	0	0	0	0	0
R/W:	R/W	R/W	R/W	R	R	R	R	R

位	位名	初始值	R/W	说明
7	TXDMY	0	R/W	虚拟数据发送允许 此位设定允许或者禁止虚拟数据的发送。 如果将 TXDMY 置“1”进行通信，即使发送缓冲器没有发送数据，也能从 MOSI 引脚发送虚拟数据进行串行通信。 在发送缓冲器没有发送数据并且 TXDMY 被置“1”时，将虚拟数据传送到移位寄存器。虚拟数据为刚从引脚发送的数据。如果在初始化后将 TXDMY 置“1”并且进行传送，发送的虚拟数据就为不定值。 0: 禁止发送虚拟数据 1: 允许发送虚拟数据 【注】 此模块只在设定为主控模式时有效。
6 5	SPLW1 SPLW0	0 1	R/W R/W	存取宽度设定 设定数据寄存器 (SPDR) 的存取宽度。如果数据寄存器 (SPDR) 的传送数据长度和 SPLW 设定的存取宽度不同，就不保证运行。 00: 禁止设定 01: 对 SPDR 寄存器进行字节存取 10: 对 SPDR 寄存器进行字存取 11: 对 SPDR 寄存器进行长字存取
4~0	—	全 0	R	保留位 写入值总是“0”。不保证写“1”时的运行。

17.3.10 时钟延迟寄存器 (SPCKD)

在命令寄存器 (SPCMD) 的 SCKDEN 位为 “1” 的状态下, SPCKD 设定从 SSL 信号有效开始到 RSPCK 振荡的期间 (RSPCK 延迟)。如果在控制寄存器 (SPCR) 的 MSTR 位和 SPE 位都为 “1” 并且在主控模式中允许此模块的状态下改写 SPCKD, 就不保证以后的运行。

在从属模式中使用此模块时, 必须将 SCKDL2 ~ SCKDL0 置 “B'000”。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SCK DL2	SCK DL1	SCK DL0
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 3	—	全 0	R	保留位 写入值总是 “0”。不保证写 “1” 时的运行。
2	SCKDL2	0	R/W	RSPCK 延迟设定 这些位设定 SPCMD 的 SCKDEN 位为 “1” 时的 RSPCK 延迟值。 SCKDL2 ~ SCKDL0 的设定值和 RSPCK 延迟值的关系如下: 000: 1 个 RSPCK 001: 2 个 RSPCK 010: 3 个 RSPCK 011: 4 个 RSPCK 100: 5 个 RSPCK 101: 6 个 RSPCK 110: 7 个 RSPCK 111: 8 个 RSPCK
1	SCKDL1	0	R/W	
0	SCKDL0	0	R/W	

17.3.11 从属选择无效延迟寄存器 (SSLND)

在主控模式中，SSLND 设定此模块从发送串行传送最后的 RSPCK 边沿到将 SSL 信号置为无效的期间 (SSL 无效延迟)。如果在控制寄存器 (SPCR) 的 MSTR 位和 SPE 位都为“1”并且在主控模式中允许此模块的状态下改写 SSLND，就不保证以后的运行。

在从属模式中使用此模块时，必须将 SLNDL2 ~ SLNDL0 置“B'000”。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SLN DL2	SLN DL1	SLN DL0
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 3	—	全 0	R	保留位 写入值总是“0”。不保证写“1”时的运行。
2	SLNDL2	0	R/W	SSL 无效延迟设定
1	SLNDL1	0	R/W	这些位设定 SPCMD 的 SLNDEN 位为“1”时的 SSL 无效延迟。
0	SLNDL0	0	R/W	SLNDL2 ~ SLNDL0 的设定值和 SSL 无效延迟值的关系如下： 000: 1 个 RSPCK 001: 2 个 RSPCK 010: 3 个 RSPCK 011: 4 个 RSPCK 100: 5 个 RSPCK 101: 6 个 RSPCK 110: 7 个 RSPCK 111: 8 个 RSPCK

17.3.12 下次存取延迟寄存器 (SPND)

在命令寄存器 (SPCMD) 的 SPNDEN 位为 “1” 的状态下，SPND 设定串行传送结束后的 SSL 信号的无效期间 (下次存取延迟)。如果在控制寄存器 (SPCR) 的 MSTR 位和 SPE 位都为 “1” 并且在主控模式中允许此模块的状态下改写 SPND，就不保证以后的运行。

在从属模式中使用此模块时，必须将 SPNDL2 ~ SPNDL0 置 “B'000”。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SPN DL2	SPN DL1	SPN DL0
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 3	—	全 0	R	保留位 写入值总是 “0”。不保证写 “1” 时的运行。
2	SPNDL2	0	R/W	下次存取延迟设定 这些位设定 SPCMD 的 SPNDEN 位为 “1” 时的下次存取延迟。 SPNDL2 ~ SPNDL0 的设定值和下次存取延迟值的关系如下： 000: 1 个 RSPCK+2 个 P1φ 001: 2 个 RSPCK+2 个 P1φ 010: 3 个 RSPCK+2 个 P1φ 011: 4 个 RSPCK+2 个 P1φ 100: 5 个 RSPCK+2 个 P1φ 101: 6 个 RSPCK+2 个 P1φ 110: 7 个 RSPCK+2 个 P1φ 111: 8 个 RSPCK+2 个 P1φ
1	SPNDL1	0	R/W	
0	SPNDL0	0	R/W	

17.3.13 命令寄存器 (SPCMD)

1 个通道有 4 个命令寄存器 (SPCMD: SPCMD0 ~ SPCMD3)。SPCMD0 ~ SPCMD3 用于设定主控模式的传送格式。SPCMD0 的一部分位也用于设定从属模式的传送格式。在**主控模式**中，此模块根据顺序控制寄存器 (SPSCR) 的 SPSLN1 位和 SPSLN0 位的设定，按顺序参照 SPCMD0 ~ 3 并且执行所参照的 SPCMD 设定的串行传送。

在状态寄存器 (SPSR) 的 TEND 位为发送未结束的状态下，如果改写此模块正在参照的 SPCMD，就不保证以后的运行。能通过顺序状态寄存器 (SPSSR) 的 SPCP1 位和 SPCP0 位确认此模块在**主控模式**中正在参照的 SPCMD。如果在从属模式中允许此模块的状态下改写 SPCMD0，就不保证以后的运行。

位:	15	14	13	12	11	10	9	8
	SCK DEN	SLN DEN	SPN DEN	LSBF	SPB3	SPB2	SPB1	SPB0
初始值:	0	0	0	0	0	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	7	6	5	4	3	2	1	0
	SSLKP	—	—	—	BRDV1	BRDV0	CPOL	CPHA
初始值:	0	0	0	0	1	1	0	1
R/W:	R/W	R	R	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	SCKDEN	0	R/W	RSPCK 延迟设定允许 在**主控模式**中，此位设定此模块从将 SSL 信号置为有效到 RSPCK 振荡为止的期间 (RSPCK 延迟)。 当 SCKDEN 为“0”时，此模块将 RSPCK 延迟设定为 1 个 RSPCK；当 SCKDEN 为“1”时，此模块在经过时钟延迟寄存器 (SPCKD) 设定的 RSPCK 延迟后开始 RSPCK 的振荡。 在从属模式中使用此模块时，必须将 SCKDEN 置“0”。 0: RSPCK 延迟为 1 个 RSPCK 1: RSPCK 延迟为 RSPCK 延迟寄存器 (SPCKD) 的设定值
14	SLNDEN	0	R/W	SSL 无效延迟设定允许 在**主控模式**中，此位设定此模块从 RSPCK 停止振荡到将 SSL 信号置为无效的期间 (SSL 无效延迟)。 当 SLNDEN 为“0”时，此模块将 SSL 无效延迟设定为 1 个 RSPCK；当 SLNDEN 为“1”时，此模块在经过从属选择无效延迟寄存器 (SSLND) 设定的 RSPCK 延迟后将 SSL 置为无效。 在从属模式中使用此模块时，必须将 SLNDEN 置“0”。 0: SSL 无效延迟为 1 个 RSPCK 1: SSL 无效延迟为从属选择无效延迟寄存器 (SSLND) 的设定值
13	SPNDEN	0	R/W	下次存取延迟允许 在**主控模式**中，此位设定此模块在结束串行发送后将 SSL 信号置为无效到下次存取的 SSL 信号有效为止的期间 (下次存取延迟)。 当 SPNDEN 为“0”时，此模块将下次存取延迟设定为 1 个 RSPCK+2 个 P1 ϕ 。当 SPNDEN 为“1”时，此模块插入下次存取延迟寄存器 (SPND) 设定的下次存取延迟。 在从属模式中使用此模块时，必须将 SPNDEN 置“0”。 0: 下次存取延迟为 1 个 RSPCK+2 个 P1 ϕ 1: 下次存取延迟为下次存取延迟寄存器 (SPND) 的设定值

位	位名	初始值	R/W	说明
12	LSBF	0	R/W	LSB first 此位设定将主控模式 / 从属模式的数据格式设定为 MSB first 或者 LSB first。 0: MSB first 1: LSB first
11 10 9 8	SPB3 SPB2 SPB1 SPB0	0 1 1 1	R/W R/W R/W R/W	数据长度设定 这些位设定主控模式 / 从属模式的传送数据长度。 0100 ~ 0111: 8 位 1111: 16 位 0010、0011: 32 位 其他: 禁止设定
7	SSLKP	0	R/W	SSL 信号电平保持 此模块在 主控模式中进行串行传送时, 此位设定在当前命令对应的 SSL 无效时序到下一个命令对应的 SSL 有效时序的期间是保持当前命令的 SSL 信号电平还是将 SSL 信号电平置为无效。 在从属模式中使用此模块时, 必须将 SSLKP 置“0”。 0: 在传送结束时将 SSL 信号置为无效 1: 在传送结束后到下次存取开始前保持 SSL 信号电平
6 ~ 4	—	全 0	R	保留位 写入值总是“0”。不保证写“1”时的运行。
3 2	BRDV1 BRDV0	1 1	R/W R/W	位速率分频设定 这是用于决定位速率的寄存器。位速率取决于 BRDV1 位、BRDV0 位和位速率寄存器 (SPBR) 的设定值的组合 (参照“17.3.8 位速率寄存器 (SPBR)”)。 SPBR 的设定值决定基本位速率。BRDV1 ~ 0 位的设定值用于选择基本位速率的无分频、2 分频、4 分频和 8 分频的位速率。能给 SPCMD0 ~ 3 分别设定不同的 BRDV1 位和 BRDV0 位。因此, 各命令能以不同的位速率进行串行传送。 00: 选择基本位速率 01: 选择基本位速率的 2 分频 10: 选择基本位速率的 4 分频 11: 选择基本位速率的 8 分频
1	CPOL	0	R/W	RSPCK 极性设定 此位设定主控模式 / 从属模式的 RSPCK 极性。在进行此模块之间的数据通信时, 需要在模块之间设定相同的 RSPCK 极性。 0: 空闲时的 RSPCK 为“0” 1: 空闲时的 RSPCK 为“1”
0	CPHA	1	R/W	RSPCK 相位设定 此位设定主控模式 / 从属模式的 RSPCK 相位。在进行此模块之间的数据通信时, 需要在模块之间设定相同的 RSPCK 相位。 0: 在奇数边沿进行数据采样, 在偶数边沿数据发生变化。 1: 在奇数边沿数据发生变化, 在偶数边沿进行数据采样。

17.3.14 缓冲控制寄存器 (SPBFCR)

SPBFCR 设定发送缓冲器 (SPTX) / 接收缓冲器 (SPRX) 的数据量复位和触发数据量。

位:	7	6	5	4	3	2	1	0
	TXRST	RXRST	TXTRG[1:0]		—	RXTRG[2:0]		
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	TXRST	0	R/W	发送缓冲器数据复位 将发送缓冲器内的发送数据置为无效并且复位到没有任何数据的状态。 0: 禁止复位 * 1: 允许复位 【注】 * 在上电复位时进行复位。
6	RXRST	0	R/W	接收缓冲器数据复位 将接收缓冲器内的接收数据置为无效并且复位到没有任何数据的状态。 0: 禁止复位 * 1: 允许复位 【注】 * 在上电复位时进行复位。
5、4	TXTRG	00	R/W	发送缓冲器的数据量触发 设定发送缓冲器为空的时序，作为状态寄存器的 SPTEF 标志基准。如果发送缓冲器 (SPTX) 保存的数据字节数小于等于以下所示的设定触发数，就将 SPTEF 标志置“1”。 00: 7Byte (1) * 01: 6Byte (2) * 10: 4Byte (4) * 11: 0Byte (8) * 【注】 * () 内的数值表示发送缓冲器 (SPTX) 的空字节数。
3	—	0	R	保留位 读写值都为“0”。
2~0	RXTRG	000	R/W	接收缓冲器的数据量触发 设定接收缓冲器为满的时序，作为状态寄存器的 SPRF 标志基准值。如果接收缓冲器 (SPRX) 保存的数据字节数大于等于以下所示的设定触发数，就将 SPRF 标志置“1”。 000: 1Byte (31) * 001: 2Byte (30) * 010: 4Byte (28) * 011: 8Byte (24) * 100: 16Byte (16) * 101: 24Byte (8) * 110: 32Byte (0) * 111: 5Byte (27) * 【注】 * () 内的数值表示接收缓冲器 (SPRX) 的空字节数。

17.3.15 缓冲器数据计数置位寄存器 (SPBFDR)

SPBFDR 表示发送缓冲器 (SPTX) 和接收缓冲器 (SPRX) 保存的数据量。高 8 位表示 SPTX 的发送数据字节数，低 8 位表示 SPRX 的接收数据字节数。

位:	15	14	13	12	11	10	9	8
	—	—	—	—	T[3:0]			
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
位:	7	6	5	4	3	2	1	0
	—	—	R[5:0]					
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 12	—	全 0	R	保留位 读写值都为“0”。
11 ~ 8	T[3:0]	0000	R	表示 SPTX 保存的未发送数据字节数。 B'0000 表示 SPTX 为空状态。 B'1000 表示 SPTX 为满状态。
7、6	—	全 0	R	保留位 读写值都为“0”。
5 ~ 0	R[5:0]	000000	R	表示 SPRX 保存的接收数据字节数。 B'000000 表示 SPRX 为空状态。 B'100000 表示 SPRX 为满状态。

17.4 运行说明

在本节中使用的术语“串行传送期间”是指从开始驱动有效数据到取最后有效数据为止的期间。

17.4.1 运行概要

此模块能进行从属模式和主控模式的串行传送。能通过控制寄存器（SPCR）的 MSTR 位设定模式。模式和 SPCR 设定的关系以及各模式的概要如表 17.4 所示。

表 17.4 模式和 SPCR 设定的关系以及各模式的概要

模式	从属（SPI 运行）	主控（SPI 运行）
MSTR 位的设定	0	1
MODFEN 位的设定	0 or 1	0
RSPCK 信号	输入	输出
MOSI 信号	输入	输出
MISO 信号	输出 /Hi-Z	输入
SSL 信号	输入	输出
SSL 极性变更功能	有	有
传送率	~ P1 ϕ /8	~ P1 ϕ /2
时钟源	RSPCK 输入	内部波特率发生器
时钟极性	2 种	2 种
时钟相位	2 种	2 种
开始传送的位	MSB/LSB	MSB/LSB
传送数据长度	8 位、16 位、32 位	8 位、16 位、32 位
突发传送	能（CPHA=1）	能（CPHA=0、1）
RSPCK 延迟控制	无	有
SSL 无效延迟控制	无	有
下次存取延迟控制	无	有
传送启动方法	SSL 输入有效或者 RSPCK 振荡	在 SPE 位为“1”时写发送缓冲器
顺序控制	无	有
发送缓冲器空检测	有	有
接收缓冲器满检测	有	有
溢出错误检测	有	无
模式故障错误检测	有（MODFEN=1）	无

17.4.2 引脚的控制

根据控制寄存器（SPCR）的 MSTR 位的设定，此模块自动切换引脚方向和输出模式。引脚状态和各位设定值的关系如表 17.5 所示。

表 17.5 引脚状态和控制位设定值的关系

模式	引脚	引脚状态
主控（SPI 运行）（MSTR=1）	RSPCK	CMOS 输出
	SSL	CMOS 输出
	MOSI	CMOS 输出
	MISO	输入
从属（SPI 运行）（MSTR=0）	RSPCK	输入
	SSL	输入
	MOSI	输入
	MISO*	CMOS 输出 /Hi-Z

【注】 * 当 SSL 为无效电平或者 SPCR 的 SPE 位为 “0” 时，引脚状态为 Hi-Z。

如表 17.6 所示，在主导模式（SPI 运行）中，此模块根据 SPPCR 的 MOIFE 位和 MOIFV 位的设定，决定 SSL 无效期间（包括突发传送中的 SSL 保持期间）的 MOSI 信号值。

表 17.6 SSL 无效期间的 MOSI 信号值的决定方法

MOIFE	MOIFV	SSL 无效期间的 MOSI 信号值
0	0、1	上次传送的最后数据
1	0	总是 “0”。
1	1	总是 “1”。

17.4.3 系统结构例子

(1) 主控 / 从属（本 LSI 为主控设备）

本 LSI 用作主控设备时的主控 / 从属系统结构例子如图 17.2 所示。在主控 / 从属的结构中，不使用本 LSI（主控设备）的 SSL 输出。将从属设备的 SSL 输入固定为低电平，使从属设备总是被选择的状态。在类似控制寄存器（SPCR）的 CPHA 位为“0”的传送格式中，有些从属设备不能将 SSL 信号固定为有效电平。在不能固定 SSL 信号时，必须将本 LSI 的 SSL 输出连接到从属设备的 SSL 输入。

本 LSI（主控设备）驱动 RSPCK 和 MOSI，从属设备驱动 MISO。

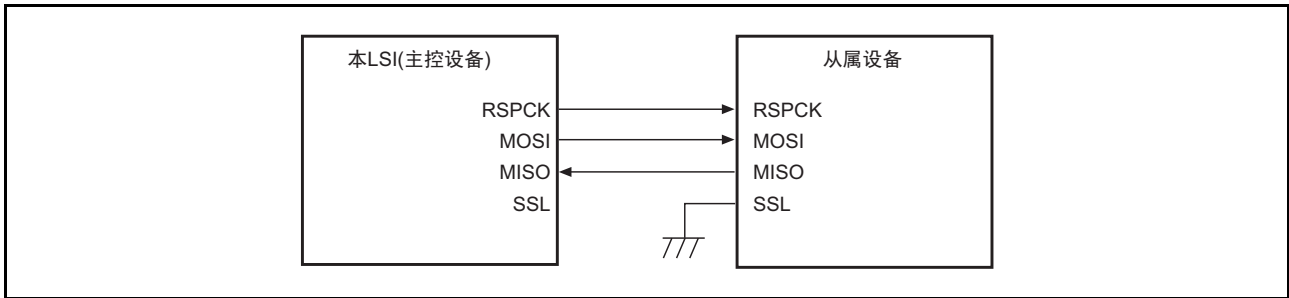


图 17.2 主控 / 从属的结构例子（本 LSI 为主控设备）

(2) 主控 / 从属（本 LSI 为从属设备）

本 LSI 用作从属设备时的主控 / 从属系统结构例子如图 17.3 所示。在本 LSI 用作从属设备时，将 SSL 引脚用作 SSL 输入。主控设备驱动 RSPCK 和 MOSI，本 LSI（从属设备）驱动 MISO。在 SSL 为无效电平时，引脚状态为 Hi-Z。

在将命令寄存器（SPCMD）的 CPHA 位置“1”的单从属结构中，也能将本 LSI（从属设备）的 SSL 输入固定为低电平，使本 LSI（从属设备）总是被选择的状态，进行串行传送（图 17.4）。

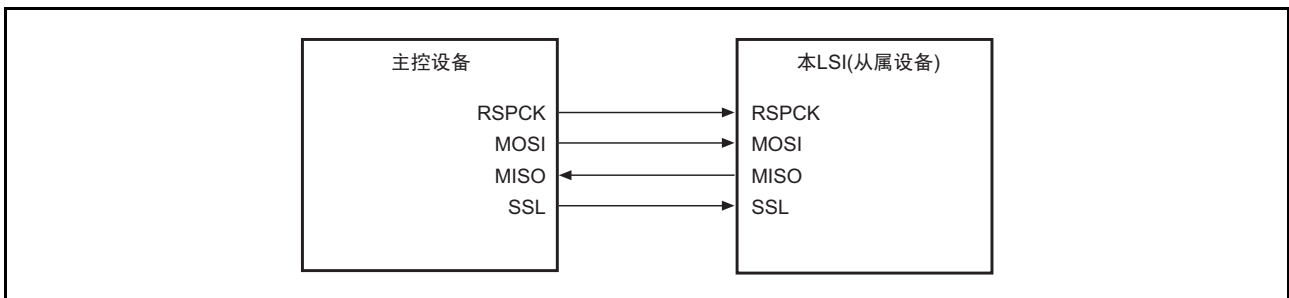


图 17.3 主控 / 从属的结构例子（本 LSI 为从属设备）

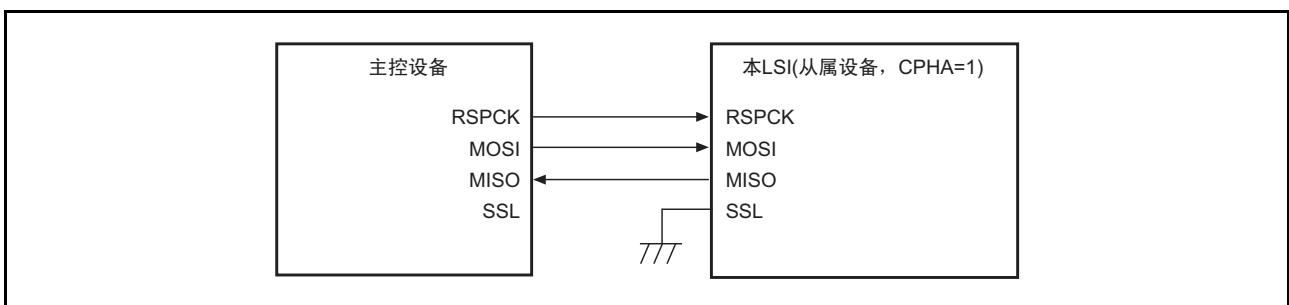


图 17.4 主控 / 从属的结构例子（本 LSI 为从属设备，CPHA 位为“1”）

(3) 主控 / 多从属（本 LSI 为从属设备）

本 LSI 用作从属设备时的主控 / 多从属系统结构例子如图 17.5 所示。在图 17.5 的例子中，系统由主控设备和 2 个本 LSI（从属设备 X 和从属设备 Y）构成。

主控设备的 RSPCK 输出和 MOSI 输出连接本 LSI（从属设备 X 和从属设备 Y）的 RSPCK 输入和 MOSI 输入，本 LSI（从属设备 X 和从属设备 Y）的 MISO 输出连接主控设备的 MISO 输入，主控设备的 SSLX 输出和 SSLY 输出连接本 LSI（从属设备 X 和从属设备 Y）的 SSL 输入。

主控设备驱动 RSPCK、MOSI、SSLX 和 SSLY。在本 LSI（从属设备 X 和从属设备 Y）中，SSL0 输入为低电平的从属设备驱动 MISO。

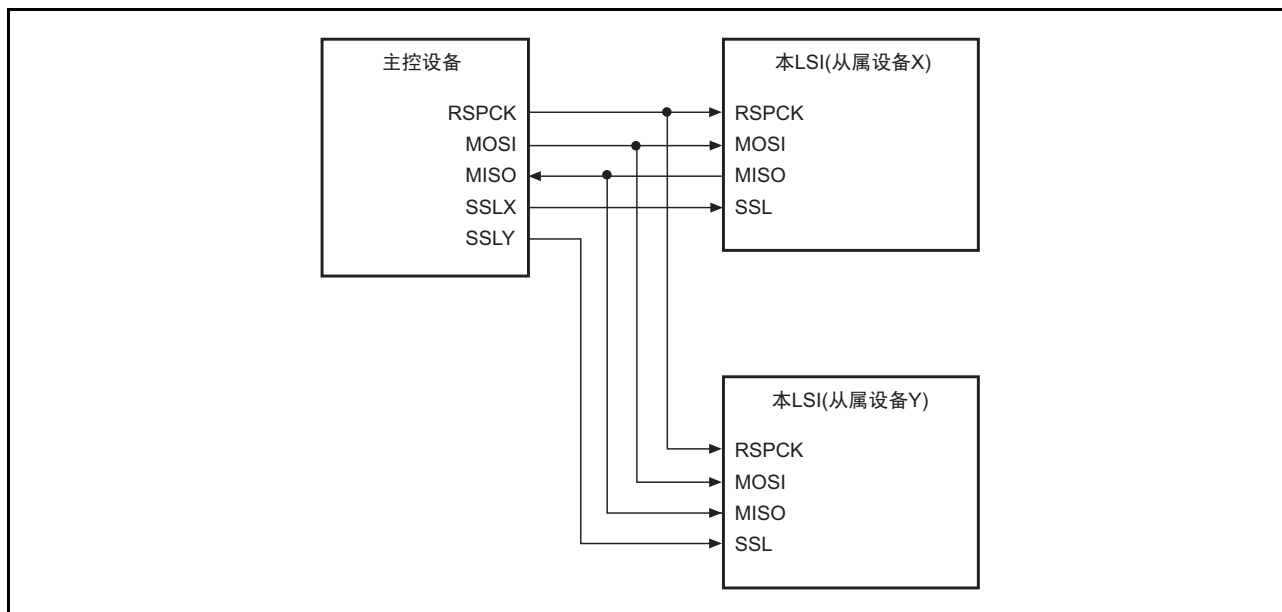


图 17.5 单主控 / 多从属的结构例子（本 LSI 为从属设备）

17.4.4 传送格式

(1) CPHA 位为“0”的情况

在命令寄存器（SPCMD）的 CPHA 位为“0”的情况下，对 8 位数据进行串行传送时的传送格式例子如图 17.6 所示。在图 17.6 中，RSPCK（CPOL=0）是 SPCMD 的 CPOL 位为“0”时的 RSPCK 信号波形，RSPCK（CPOL=1）是 CPOL 位为“1”时的 RSPCK 信号波形。采样时序表示此模块将串行传送数据取到移位寄存器的时序。各信号的输入/输出方向取决于此模块的设定，详细内容请参照“17.4.2 引脚的控制”。

当 CPHA 位为“0”并且在 SSL 信号有效时，开始驱动 MOSI 信号和 MISO 信号的有效数据。在 SSL 信号有效后产生的第一个 RSPCK 信号变化时序为第一个传送数据的取时序。在此时序之后，按每个 RSPCK 周期对数据进行采样。MOSI 信号和 MISO 信号的变化时序总是在传送数据取时序的 1/2 个 RSPCK 周期之后。CPOL 位的设定值不影响 RSPCK 信号的运行时序而只影响信号的极性。

t1 是从 SSL 信号有效到 RSPCK 振荡的期间（RSPCK 延迟），t2 是从 RSPCK 停止振荡到 SSL 信号无效的期间（SSL 无效延迟），t3 是在串行传送结束后抑制下次传送的 SSL 信号有效的期间（下次存取延迟）。t1、t2 和 t3 受系统上的主控设备的控制。有关此模块为主控模式时的 t1、t2 和 t3，请参照“17.4.3(1) 主控/从属（本 LSI 为主控设备）”。

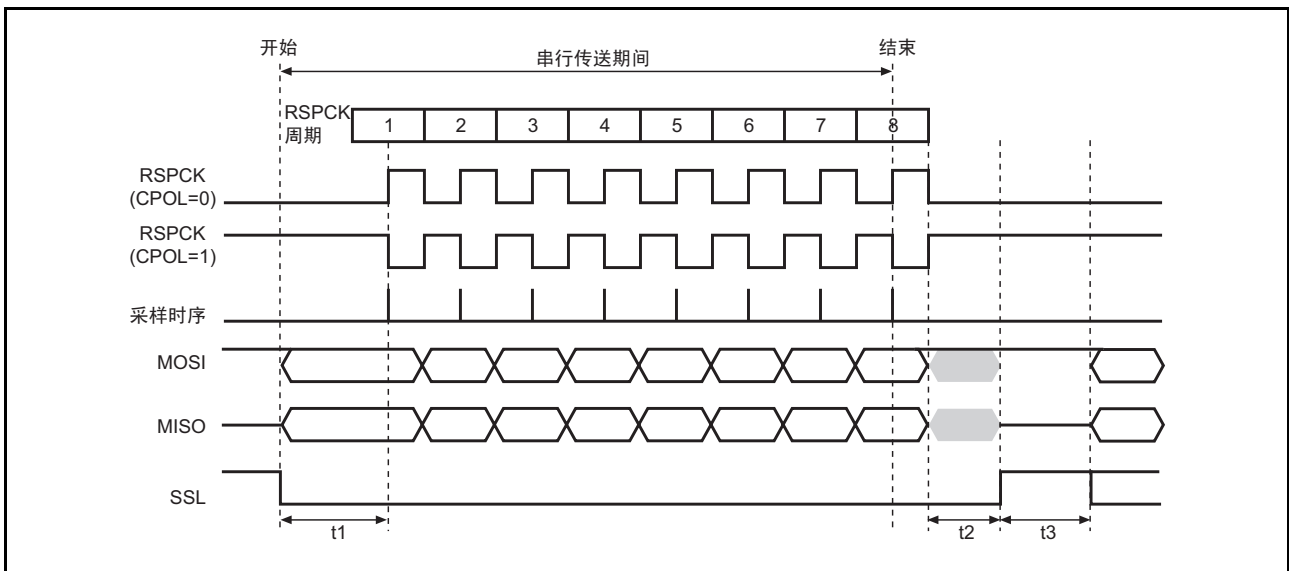


图 17.6 传送格式（CPHA 位为“0”）

(2) CPHA 位为“1”的情况

在命令寄存器（SPCMD）的 CPHA 位为“1”的情况下，对 8 位数据进行串行传送时的传送格式例子如图 17.7 所示。在图 17.7 中，RSPCK（CPOL=0）是 SPCMD 的 CPOL 位为“0”时的 RSPCK 信号波形，RSPCK（CPOL=1）是 CPOL 位为“1”时的 RSPCK 信号波形。采样时序表示此模块将串行传送数据取到移位寄存器的时序。各信号的输入/输出方向取决于模式（主控/从属），详细内容请参照“17.4.2 引脚的控制”。

当 CPHA 位为“1”并且在 SSL 信号有效时，开始驱动 MOSI 信号和 MISO 信号的无效数据。在 SSL 信号有效后产生的第一个 RSPCK 信号变化时序，开始驱动 MOSI 信号和 MISO 信号的有效数据。在此时序之后，按每个 RSPCK 周期更新数据。传送数据的取时序总是在此时序的 1/2 个 RSPCK 周期之后。CPOL 位的设定值不影响 RSPCK 信号的运行时序而只影响信号的极性。

t1、t2、t3 的内容和 CPHA 位为“0”的情况相同。有关此模块为主控模式时的 t1、t2 和 t3，请参照“17.4.3 (1) 主控/从属（本 LSI 为主控设备）”。

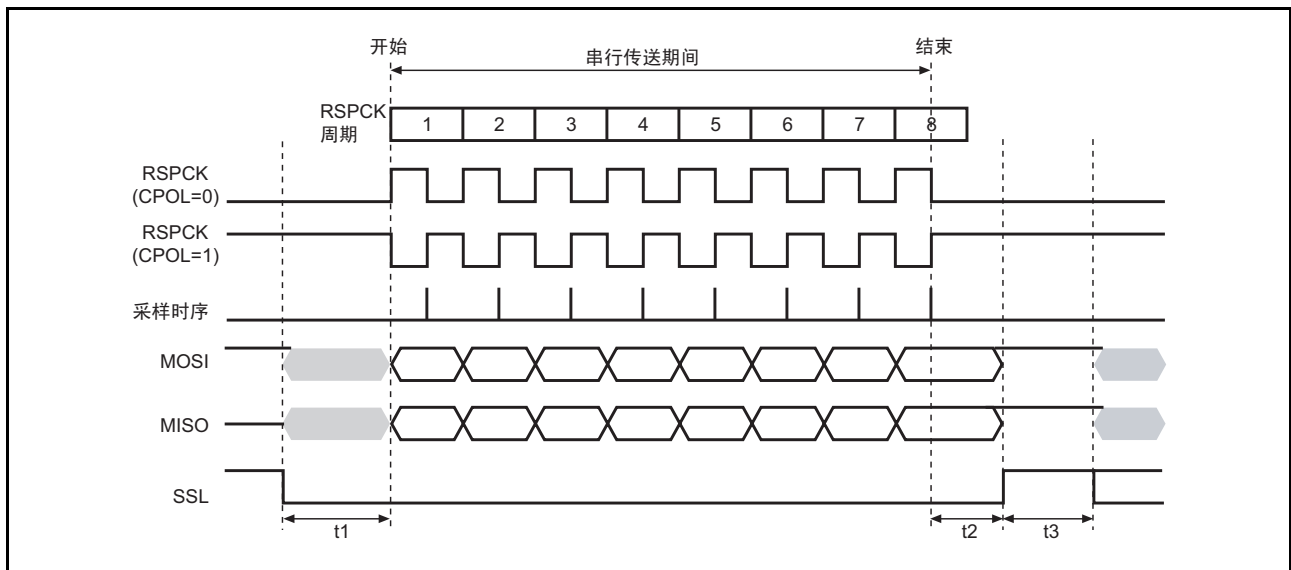


图 17.7 传送格式（CPHA 位为“1”）

17.4.5 数据格式

数据格式取决于命令寄存器（SPCMD）的设定值。与 MSB/LSB first 无关，此模块将从数据寄存器（SPDR）的 LSB 开始的设定数据长度的范围作为传送数据。

(1) MSB first 传送（32 位数据）

此模块进行 32 位数据 MSB first 传送时的发送缓冲器（SPTX）和移位寄存器的运行内容如图 17.8 所示。

CPU 或者直接存储器存取控制器将 T31 ~ T00 写到 SPDR 的发送缓冲器。如果移位寄存器为空，此模块就将发送缓冲器的数据复制到移位寄存器，使移位寄存器变满。一旦串行传送开始，此模块就从移位寄存器的 MSB（bit31）开始输出数据，从移位寄存器的 LSB（bit0）开始移入数据。如果经过 32 位串行传送所需的 RSPCK 周期，数据 R31 ~ R00 就被保存到移位寄存器。在此状态下，此模块将数据从移位寄存器复制到接收缓冲器，使移位寄存器变空。在将接收数据从移位寄存器复制到接收缓冲器后，如果接收缓冲器没有接收数据长度的空间，就不开始下一次串行传送。要开始下一次串行传送时，必须从接收缓冲器读接收数据长度的数据，确保接收缓冲器的空间。

如果在 CPU 或者直接存储器存取控制器写发送缓冲器前启动下一次串行传送，就从移位寄存器移出接收数据 R31 ~ R00。

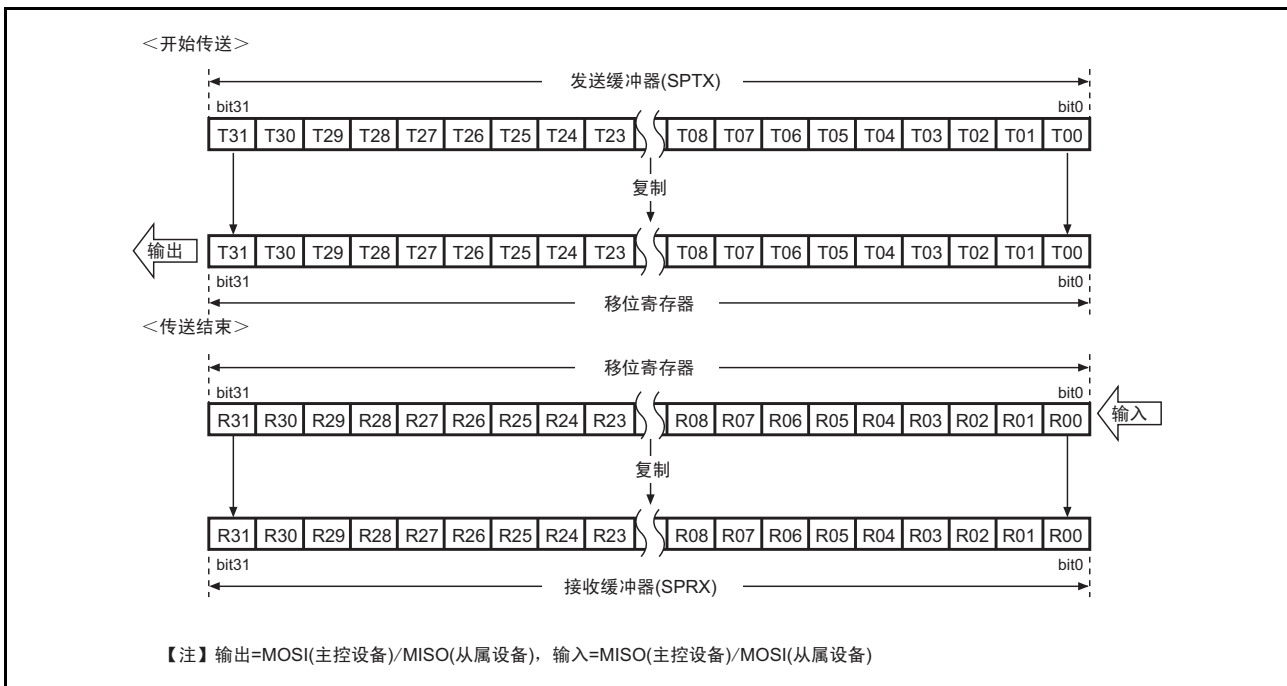


图 17.8 MSB first 传送（32 位数据）

(2) MSB first 传送 (16 位数据)

此模块进行 16 位数据传送时的发送缓冲器 (SPTX) 和移位寄存器的运行内容如图 17.9 所示。

CPU 或者直接存储器存取控制器将 T15 ~ T00 写到发送缓冲器。如果移位寄存器为空, 此模块就将发送缓冲器的数据复制到移位寄存器, 使移位寄存器变满。一旦串行传送开始, 此模块就从移位寄存器的 bit15 开始输出数据, 从移位寄存器的 LSB (bit0) 开始移入数据。如果经过 16 位串行传送所需的 RSPCK 周期, 接收数据 R15 ~ R00 就被保存到移位寄存器的 bit15 ~ 0, 串行传送结束后的移位寄存器的 bit31 ~ 16 保持传送前的数据。在此状态下, 此模块将数据从移位寄存器复制到接收缓冲器, 使移位寄存器变空。在将接收数据从移位寄存器复制到接收缓冲器后, 如果接收缓冲器没有接收数据长度的空间, 就不开始下一次串行传送。要开始下一次串行传送时, 必须从接收缓冲器读接收数据长度的数据, 确保接收缓冲器的空间。

如果在 CPU 或者直接存储器存取控制器写发送缓冲器前启动下一次串行传送, 就从移位寄存器移出接收数据 R15 ~ R00。

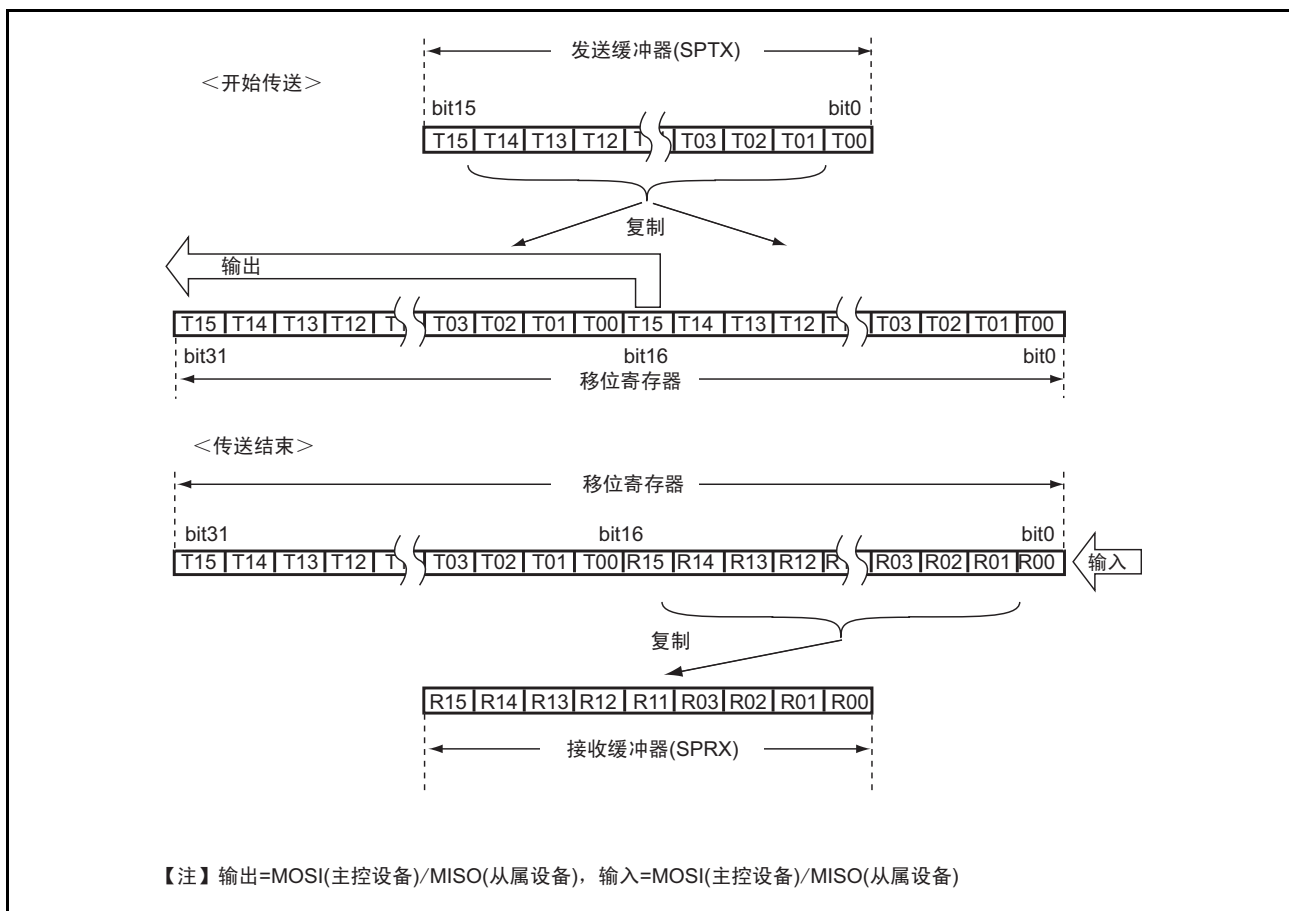


图 17.9 MSB first 传送 (16 位数据)

(3) MSB first 传送 (8 位数据)

此模块进行 8 位数据传送时的数据寄存器 (SPDR) 和移位寄存器的运行内容如图 17.10 所示。

CPU 或者直接存储器存取控制器将 T07 ~ T00 写到发送缓冲器。如果移位寄存器为空, 此模块就将发送缓冲器的数据复制到移位寄存器, 使移位寄存器变满。一旦串行传送开始, 此模块就从移位寄存器的 bit7 开始输出数据, 从移位寄存器的 LSB 开始 (bit0) 移入数据。如果经过 8 位串行传送所需的 RSPCK 周期, 接收数据 R07 ~ R00 就被保存到移位寄存器的位 bit7 ~ 0, 串行传送结束后的移位寄存器的 bit31 ~ 8 保持传送前的数据。在此状态下, 此模块将数据从移位寄存器复制到接收缓冲器, 使移位寄存器变空。在将接收数据从移位寄存器复制到接收缓冲器后, 如果接收缓冲器没有接收数据长度的空间, 就不开始下一次串行传送。要开始下一次串行传送时, 必须从接收缓冲器读接收数据长度的数据, 确保接收缓冲器的空间。

如果在 CPU 或者直接存储器存取控制器写发送缓冲器前启动下一次串行传送, 就从移位寄存器移出接收数据 R07 ~ R00。

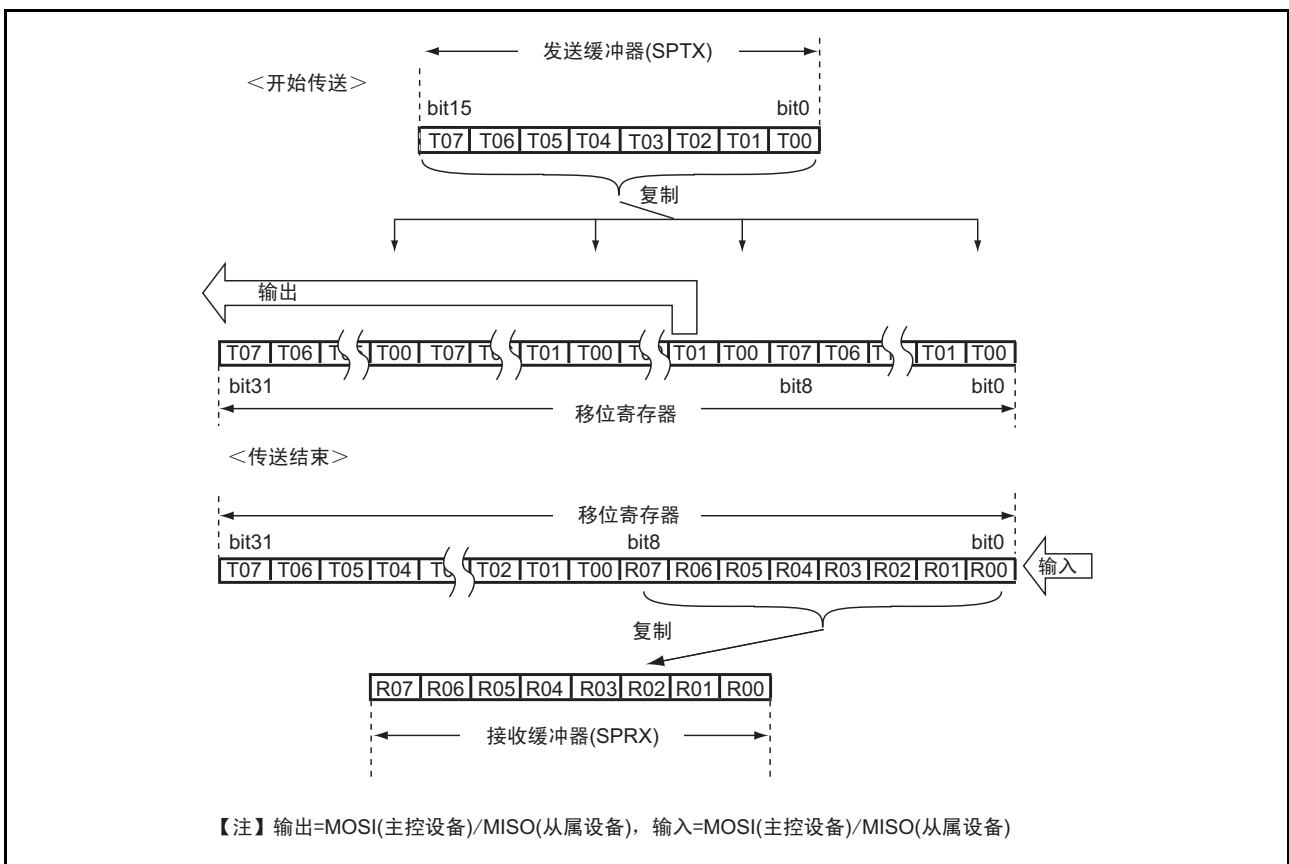


图 17.10 MSB first 传送 (8 位数据)

(4) LSB first 传送 (32 位数据)

此模块进行 32 位数据 LSB first 传送时的发送缓冲器 (SPTX) 和移位寄存器的运行内容如图 17.11 所示。

CPU 或者直接存储器存取控制器将 T31 ~ T00 写到发送缓冲器。如果移位寄存器为空, 此模块就在将发送缓冲器的发送数据的位序颠倒后复制到移位寄存器, 使移位寄存器变满。一旦串行传送开始, 此模块就从移位寄存器的 MSB (bit31) 开始输出发送数据, 从移位寄存器的 LSB (bit0) 开始移入接收数据。如果经过 32 位串行传送所需的 RSPCK 周期, 接收数据 R00 ~ R31 就被保存到移位寄存器。在此状态下, 此模块将位序颠倒的数据从移位寄存器复制到接收缓冲器, 使移位寄存器变空。

在将接收数据从移位寄存器复制到接收缓冲器后, 如果接收缓冲器没有接收数据长度的空间, 就不开始下一次串行传送。要开始下一次串行传送时, 必须从接收缓冲器读接收数据长度的数据, 确保接收缓冲器的空间。

如果在 CPU 或者直接存储器存取控制器写 SPDR 的发送缓冲器前启动下一次串行传送, 就从移位寄存器移出接收数据 R00 ~ R31。

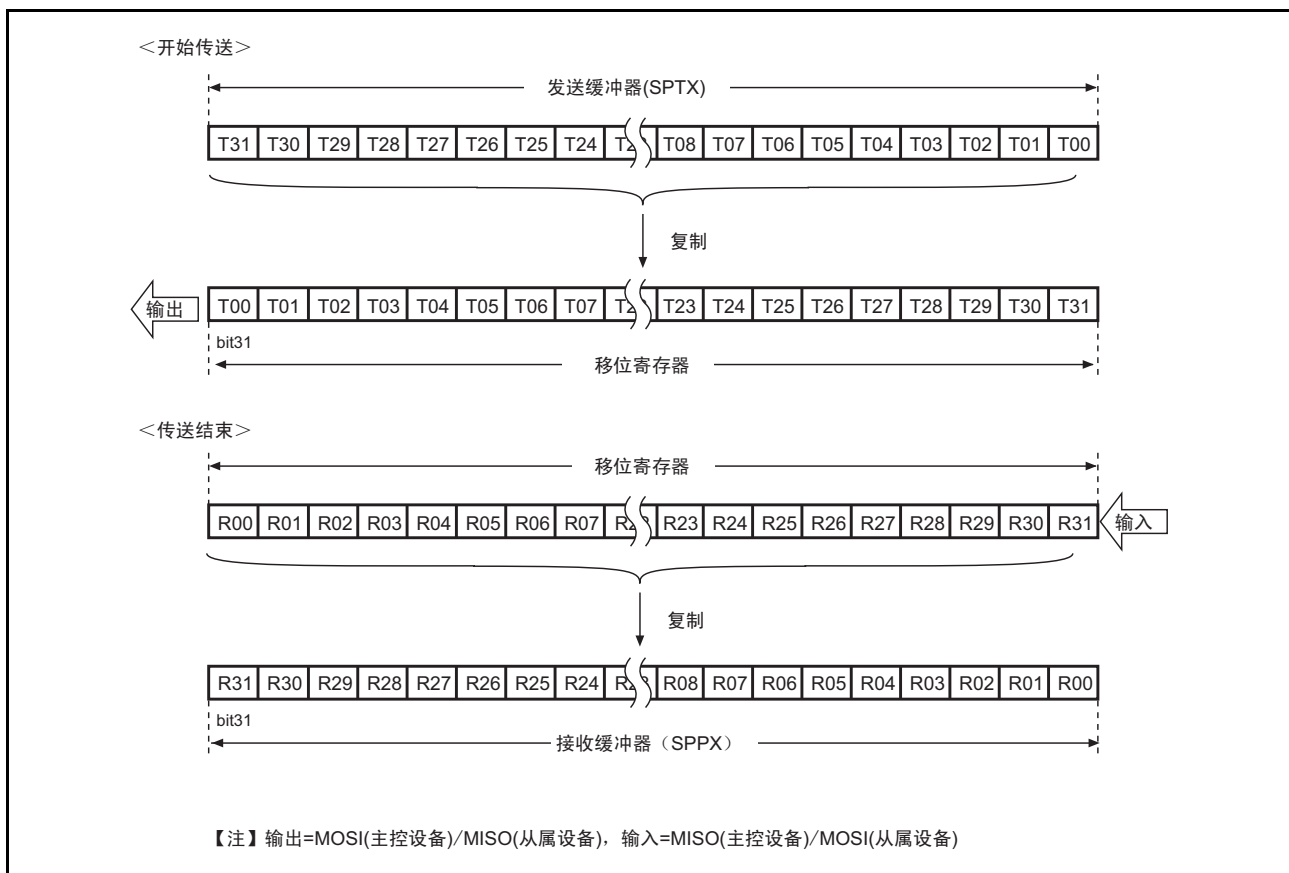


图 17.11 LSB first 传送 (32 位数据)

(5) LSB first 传送 (16 位数据)

此模块进行 16 位数据传送时的发送缓冲器 (SPTX) 和移位寄存器的运行内容如图 17.12 所示。

CPU 或者直接存储器存取控制器将 T15 ~ T00 写到发送缓冲器。如果移位寄存器为空, 此模块就在将发送缓冲器的发送数据的位序颠倒后复制到移位寄存器, 使移位寄存器变满。一旦串行传送开始, 此模块就从移位寄存器的 MSB (bit31) 开始输出数据, 从移位寄存器的 bit16 开始移入数据。如果经过 16 位串行传送所需的 RSPCK 周期, 接收数据 R00 ~ R15 就被保存到移位寄存器的 bit31 ~ 16。串行传送结束后的移位寄存器的 bit15 ~ 0 保持传送前的数据。在此状态下, 此模块将位序颠倒的数据从移位寄存器复制到 SPDR 的接收缓冲器, 使移位寄存器变空。在将接收数据从移位寄存器复制到接收缓冲器后, 如果接收缓冲器没有接收数据长度的空间, 就不开始下一次串行传送。要开始下一次串行传送时, 必须从接收缓冲器读接收数据长度的数据, 确保接收缓冲器的空间。

如果在 CPU 或者直接存储器存取控制器写 SPDR 的发送缓冲器前启动下一次串行传送, 就从移位寄存器移出接收数据 R00 ~ R15。

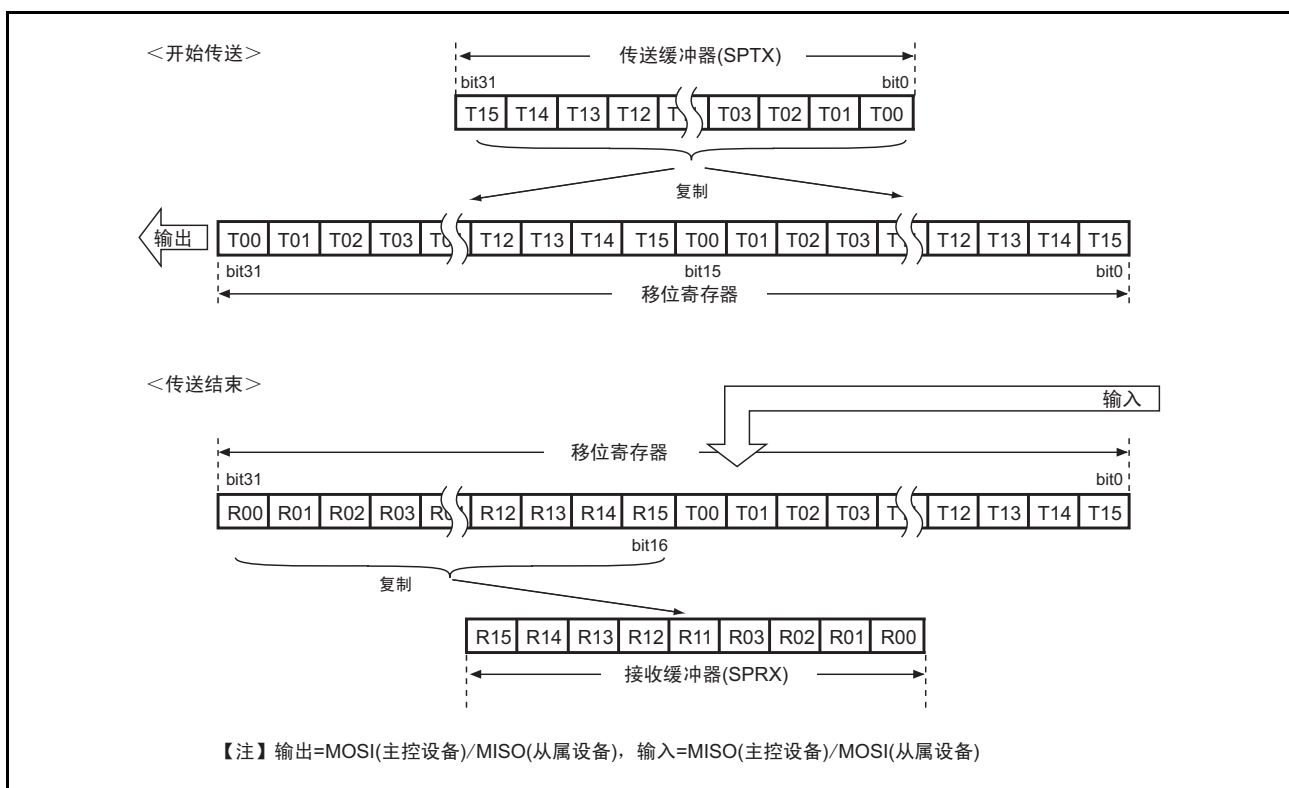


图 17.12 LSB first (16 位数据)

(6) LSB first 传送 (8 位数据)

此模块进行 8 位数据传送时的发送缓冲器 (SPTX) 和移位寄存器的运行内容如图 17.13 所示。

CPU 或者直接存储器存取控制器将 T07 ~ T00 写到发送缓冲器。如果移位寄存器为空, 此模块就将发送缓冲器的发送数据的位序颠倒后复制到移位寄存器, 使移位寄存器变满。一旦串行传送开始, 此模块就从移位寄存器的 MSB (bit31) 开始输出数据, 从移位寄存器的 bit24 开始移入数据。如果经过 8 位串行传送所需的 RSPCK 周期, 接收数据 R00 ~ R07 就被保存到移位寄存器的 bit31 ~ 24。串行传送结束后的移位寄存器的 bit23 ~ 0 保持传送前的数据。在此状态下, 此模块将位序颠倒的数据从移位寄存器复制到 SPDR 的接收缓冲器, 使移位寄存器变空。在将接收数从据移位寄存器复制到接收缓冲器后, 如果接收缓冲器没有接收数据长度的空间, 就不开始下一次串行传送。要开始下一次串行传送时, 必须从接收缓冲器读接收数据长度的数据, 确保接收缓冲器的空间。

如果在 CPU 或者直接存储器存取控制器写 SPDR 的发送缓冲器前启动下一次串行传送, 就从移位寄存器移出接收数据 R00 ~ R07。

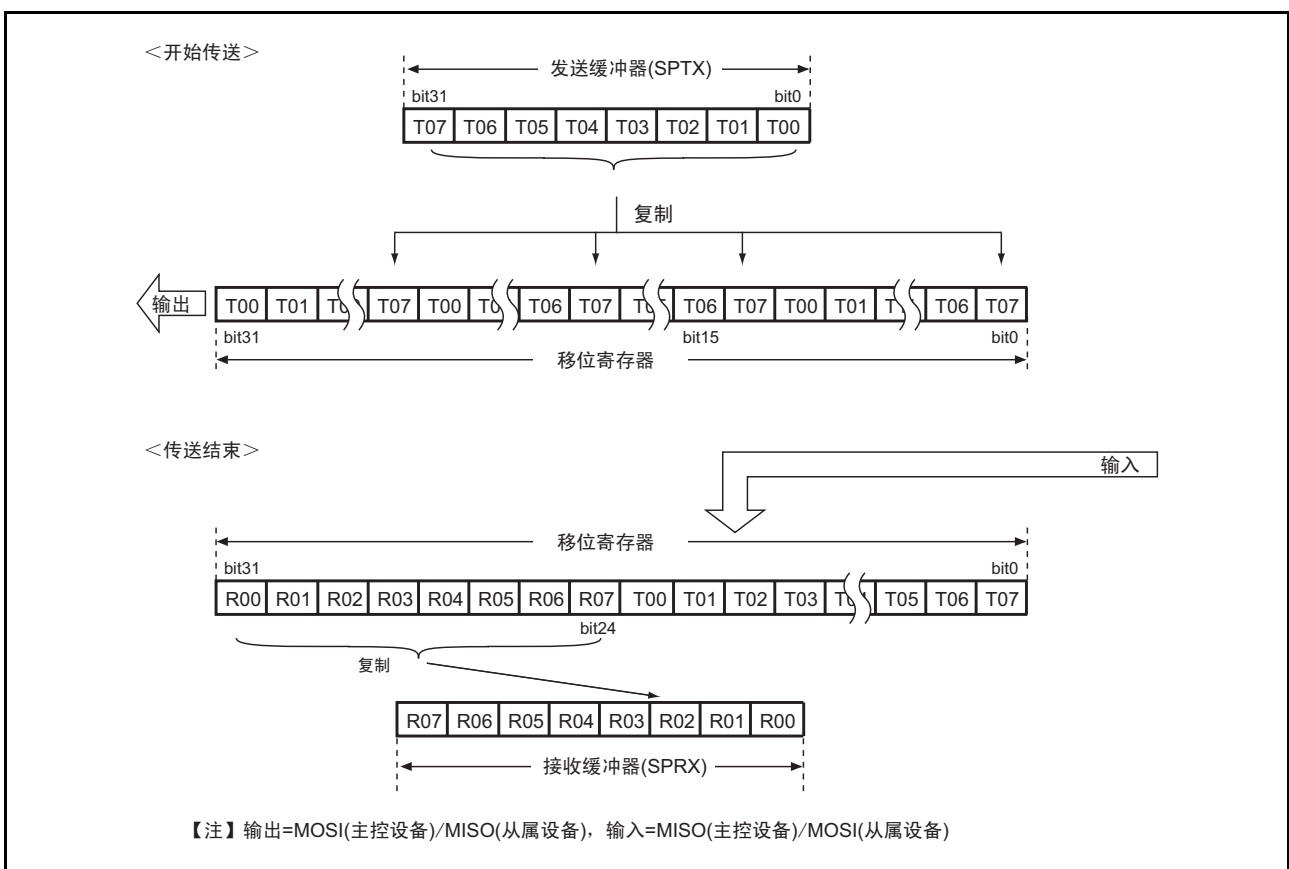


图 17.13 LSB first (8 位数据)

17.4.6 错误检测

正常的串行传送能对写在数据寄存器（SPDR）的发送缓冲器中的数据进行串行发送，并且能从 SPDR 的接收缓冲器读串行接收的数据。根据存取 SPDR 时的发送 / 接收缓冲器的状态和串行传送开始 / 结束时的状态，有可能发生异常传送。

如果发生部分异常传送，此模块就检测为溢出错误或者模式故障错误。异常传送和错误检测功能的关系如表 17.7 所示。

表 17.7 异常传送的发生条件和错误检测功能

	发生条件	运行	错误检测
A	在发送缓冲器满的状态下写 SPDR。	写数据丢失。	无
B	在从属模式中未将发送数据设定到移位寄存器的状态下开始串行传送。	将上次串行传送时的接收数据进行串行发送。	无
C	在接收缓冲器没有接收数据的状态下读 SPDR。	输出不确定。	无
D	在接收缓冲器的接收数据满的状态下结束串行传送。	串行接收数据丢失。	溢出错误检测 (只限于从属模式)
E	在从属模式的串行传送过程中，SSL 输入信号无效。	中止串行传送。 发送 / 接收数据丢失。 禁止运行。	模式故障错误检测

此模块不对表 17.7 的 A 所示的运行进行错误检测，能通过缓冲器数据计数置位寄存器（SPBFDR）的 T[3:0] 位确认是否能写 SPDR。

此模块也不对 B 所示的运行进行错误检测。在更新移位寄存器前启动的串行传送中，此模块发送上次串行传送时的接收数据，而不将 B 所示的运行作为错误进行处理。因为上次串行传送时的接收数据保存在 SPDR 的接收缓冲器，所以能正常读取。

此模块也不对 C 所示的运行进行错误检测。为了只读需要的数据，必须通过缓冲器数据计数置位寄存器（SPBFDR）的 R[5:0] 位确认保存在接收缓冲器的接收数据量。

有关 D 所示的溢出错误，在“17.4.6(1) 溢出错误”中详细说明；有关 E 所示的模式故障错误，在“17.4.6(2) 模式故障错误”中详细说明。

(1) 溢出错误

如果在数据寄存器（SPDR）的接收缓冲器满的状态下结束串行传送，此模块就在检测到溢出错误后将 SPSR 的 OVRF 位置“1”。因为在 OVRF 位为“1”的状态下此模块不将移位寄存器的数据复制到接收缓冲器，所以接收缓冲器保存发生错误前的数据。要将 SPSR 的 OVRF 位置“0”时，需要进行上电复位，或者在 OVRF 位为“1”的状态下读 SPSR 后给 OVRF 写“0”。

SPSR 的 SPRF 位和 OVRF 位的运行如图 17.14 所示。图 17.14 中记载的 SPSR 存取和 SPDR 存取表示各自的存取状况。I 表示空闲状态，W 表示写周期，R 表示读周期。在图 17.14 的例子中，命令寄存器（SPCMD）的 CPHA 为“1”并且 CPOL 为“0”，此模块进行 8 位串行传送。在 RSPCK 波形下面记载的数字表示 RSPCK 周期数（传送位数）。

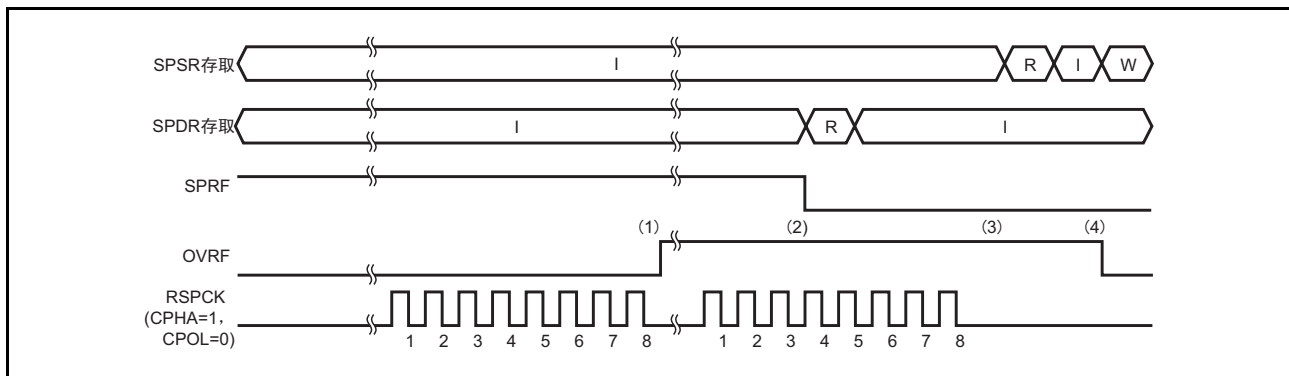


图 17.14 SPRF 位和 OVRF 位的运行例子

以下说明图中 (1) ~ (4) 所示的时序中的标志内容：

1. 如果在接收缓冲器没有接收数据长度空间的状态下结束串行传送，此模块就检测到溢出错误并且将 OVRF 位置“1”。此模块不将移位寄存器的数据复制到接收缓冲器。
2. 即使读 SPDR 并且小于等于缓冲控制寄存器（SPBFCR）的接收缓冲器数据量触发（RXTRG）位设定的触发数，也不清除 OVRF 位。
3. 如果在发生溢出错误的状态下结束串行传送，此模块就判断移位寄存器为空状态，处于可将数据从发送缓冲器传送到移位寄存器的状态。
4. 如果在 OVRF 位为“1”的状态下读 SPSR 后给 OVRF 写“0”，此模块就清除 OVRF 位。

能通过读 SPSR 或者通过错误中断以及读 SPSR，确认溢出的发生。在使用错误中断时，必须将控制寄存器（SPCR）的 SPEIE 位置“1”。在不使用错误中断进行串行传送时，必须通过在读 SPDR 后立即读 SPSR 等方法，尽快检测到溢出错误的发生。

将 OVRF 位清“0”的条件如下：

- 在 OVRF 被置“1”的状态下读 SPSR 后给 OVRF 写“0”时
- 上电复位

【注】如果在发生溢出错误的状态下接收缓冲器有接收数据保存区，此模块就接收数据。

(2) 模式故障错误

当 MSTR 位为“0”时，此模块以从属模式运行。在从属模式中 MODFEN 位为“1”的情况下，如果在串行传送期间（从开始驱动有效数据到取最后的有效数据为止）SSL 输入信号无效，此模块就检测到模式故障错误。

如果此模块检测到模式故障错误，就停止驱动输出信号并且清除 SPCR 寄存器的 SPE 位。如果 SPE 位变为“0”，就将此模块功能置为无效，停止此模块驱动的外部信号。有关通过将 SPE 位置“0”使此模块功能无效的详细内容，请参照“17.4.7 初始化”。

能通过读 SPSR 或者通过错误中断以及读 SPSR，确认模式故障错误的发生。在使用错误中断时，必须将控制寄存器（SPCR）的 SPEIE 位置“1”。在不使用错误中断来检测模式故障错误时，需要轮询 SPSR。

在 MODF 位为“1”的状态下，忽视给 SPE 位写“1”。为了使此模块功能在检测到模式故障错误后有效，必须将 MODF 位置“0”。

将 MODF 位清“0”的条件如下：

- 在 MODF 为“1”的状态下读 SPSR 后给 MODF 写“0”时
- 上电复位

17.4.7 初始化

如果给控制寄存器（SPCR）的 SPE 位写“0”或者此模块通过模式故障错误检测将 SPE 位置“0”，此模块就将模块功能置为无效并且对部分模块功能进行初始化。如果发生上电复位，此模块就对全部模块功能进行初始化。以下说明通过 SPE 位的清除进行的初始化。

(1) 通过清除 SPE 位进行的初始化

如果清除 SPCR 的 SPE 位，此模块就进行以下所示的初始化：

- 中止正在执行的串行传送。
- 如果是从属模式，就停止驱动输出信号（Hi-Z）。
- 对内部状态进行初始化。
- 对 SPSR 的 TEND 位进行初始化。

在通过清除 SPE 位进行的初始化中，不对此模块的控制位进行初始化。因此，只要重新将 SPE 位置“1”，就能用和初始化前相同的传送模式来启动此模块。

17.4.8 SPI 运行

(1) 主控模式的运行

以下说明多主控模式的运行。

(a) 串行传送的开始

串行传送的开始条件是：将发送数据从发送缓冲器复制到移位寄存器，使移位寄存器变满，并且接收缓冲器有接收数据长度的空间。如果已将发送数据写到移位寄存器，就不将数据从发送缓冲器复制到移位寄存器。

有关传送格式的详细内容，请参照“17.4.4 传送格式”。

(b) 串行传送的结束

与命令寄存器（SPCMD）的 CPHA 位无关，如果此模块发送最后采样时序所对应的 RSPCK 边沿，就结束串行传送。在串行传送结束后将接收数据从移位寄存器复制到接收缓冲器。在将接收数据从移位寄存器复制到接收缓冲器后，如果接收缓冲器没有接收数据长度的空间，就不开始下一次串行传送。要开始下一次串行传送时，必须从接收缓冲器读接收数据长度的数据，确保接收数据长度的空间。

最后的采样时序因传送数据的位长而变，主控模式的数据长度取决于命令寄存器（SPCMD）的 SPB3 ~ SPB0 位的设定值。有关传送格式的详细内容，请参照“17.4.4 传送格式”。

(c) 顺序控制

主控模式中的传送格式取决于顺序控制寄存器（SPSCR）、命令寄存器 0 ~ 3（SPCMD0 ~ 3）、位速率寄存器（SPBR）、时钟延迟值寄存器（SPCKD）、从属选择无效延迟寄存器（SSLND）和下次存取延迟值寄存器（SPND）。

在主控模式中，SPSCR 决定此模块要执行的串行传送的顺序结构。给 SPCMD0 ~ 3 设定 SSL 输出信号值、MSB/LSB first、数据长度、一部分位速率、RSPCK 极性 / 相位以及设定是否参照 SPCKD、SSLND 和 SPND；给 SPBR 设定一部分位速率，给 SPCKD 设定时钟延迟值，给 SSLND 设定 SSL 无效延迟并且给 SPND 设定下次存取延迟值。

此模块根据 SPSCR 设定的顺序长度，由部分或者全部 SPCMD0 ~ 3 构成顺序。此模块有构成顺序的 SPCMD 的对应指针，能通过读顺序状态寄存器（SPSSR）的 SPCP1 位和 SPCP0 位来确认此指针的值。如果通过将控制寄存器（SPCR）的 SPE 位置“1”允许此模块的功能，此模块就将指向命令的指针设定到 SPCMD0，并且在开始串行传送时将 SPCMD0 的设定内容反映到传送格式。每当各数据传送的下次存取延迟期间结束时，此模块将指针进行递增。当构成顺序的最后命令对应的串行传送结束时，此模块将指针设定到 SPCMD0，重复执行顺序。

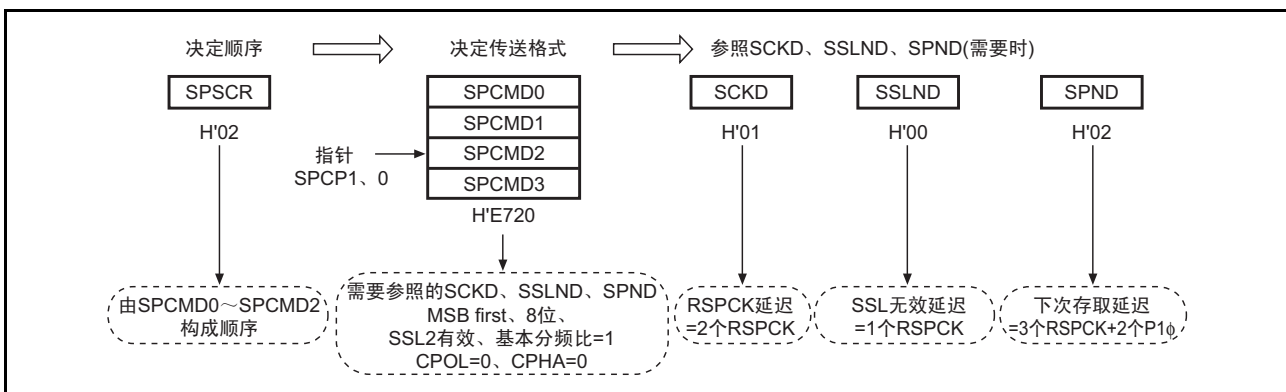


图 17.15 主控模式中的串行传送方式的决定方法

(d) 突发传送

当此模块正在进行的串行传送所参照的命令寄存器（SPCMD）的 SSLKP 位为“1”时，此模块将串行传送过程中的 SSL 信号电平保持到下一次串行传送的 SSL 信号开始有效为止。如果下一次串行传送的 SSL 信号电平和正在进行的串行传送中的 SSL 信号电平相同，此模块就能在保持 SSL 信号有效的状态下，连续进行串行传送（突发传送）。

通过设定 SPCMD0 和 SPCMD1 来实现突发传送时的 SSL 信号例子如图 17.16 所示。以下说明图 17.16 中记载的 (1) ~ (7) 的运行内容。SSL 输出信号的极性取决于从属选择极性寄存器（SSLP）的设定值。

1. 根据 SPCMD0，使 SSL 信号有效并且插入 RSPCK 延迟。
2. 根据 SPCMD0 进行串行传送。
3. 插入 SSL 无效延迟。
4. 因为 SPCMD0 的 SSLKP 位为“1”，所以保持 SPCMD0 中的 SSL 信号值，最短的保持期间至少要和 SPCMD0 的下次存取延迟相同。如果在经过最短期间后移位寄存器变空，就将此期间继续保持到下次要传送的发送数据被保存到移位寄存器为止。
5. 根据 SPCMD1，使 SSL 信号有效并且插入 RSPCK 延迟。
6. 根据 SPCMD1 进行串行传送。
7. 因为 SPCMD1 的 SSLKP 位为“0”，所以将 SSL 信号置为无效。根据 SPCMD1 插入下次存取延迟。

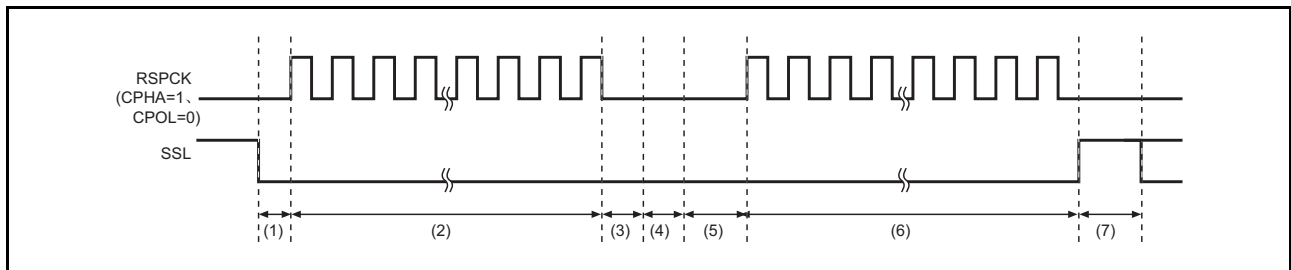


图 17.16 使用 SSLKP 位的突发传送运行例子

如果在 SSLKP 位被置“1”的 SPCMD 中设定的 SSL 信号输出和在下次传送所使用的 SPCMD 中设定的 SSL 信号输出不同，此模块就在下次传送命令所对应的 SSL 信号有效时（图 17.16 的 (5)）切换 SSL 信号状态。必须注意：如果切换 SSL 信号，就可能因驱动 MISO 的从属设备发生竞争而引起信号电平的冲突。

在主控模式中，此模块在模块内部参照不使用 SSLKP 时的 SSL 信号。即使 SPCMD 的 CPHA 位为“0”，此模块也能使用在内部检测到的下次传送的有效 SSL 信号，正确地开始串行传送。因此，与 CPHA 位的设定值无关，能进行主控模式的突发传送（参照“17.4.8(2) 从属模式的运行”）。

(e) RSPCK 延迟 (t1)

主控模式的 RSPCK 延迟值取决于命令寄存器 (SPCMD) 的 SCKDEN 位和时钟延迟寄存器 (SPCKD) 的设定。此模块通过指针控制来决定串行传送时要参照的 SPCMD，并且使用所选 SPCMD 的 SCKDEN 位和 SPCKD，决定表 17.8 所示的串行传送时的 RSPCK 延迟值。有关 RSPCK 延迟的定义，请参照“17.4.4 传送格式”。

表 17.8 SCKDEN、SPCKD 和 RSPCK 延迟值的关系

SCKDEN	SPCKD	RSPCK 延迟值
0	000 ~ 111	1 个 RSPCK
1	000	1 个 RSPCK
	001	2 个 RSPCK
	010	3 个 RSPCK
	011	4 个 RSPCK
	100	5 个 RSPCK
	101	6 个 RSPCK
	110	7 个 RSPCK
	111	8 个 RSPCK

(f) SSL 无效延迟 (t2)

主控模式的 SSL 无效延迟值取决于命令寄存器 (SPCMD) 的 SLNDEN 位和 SSL 无效延迟寄存器 (SSLND) 的设定。此模块通过指针控制来决定串行传送时要参照的 SPCMD，并且使用所选 SPCMD 的 SLNDEN 位和 SSLND，决定表 17.9 所示的串行传送时的 SSL 无效延迟值。有关 SSL 无效延迟的定义，请参照“17.4.4 传送格式”。

表 17.9 SSLND 和 SSL 无效延迟值的关系

SLNDEN	SSLND	SSL 无效延迟值
0	000 ~ 111	1 个 RSPCK
1	000	1 个 RSPCK
	001	2 个 RSPCK
	010	3 个 RSPCK
	011	4 个 RSPCK
	100	5 个 RSPCK
	101	6 个 RSPCK
	110	7 个 RSPCK
	111	8 个 RSPCK

(g) 下次存取延迟 (t₃)

主控模式的下次存取延迟取决于命令寄存器 (SPCMD) 的 SPNDEN 位和下次存取延迟寄存器 (SPND) 的设定。此模块通过指针控制来决定串行传送时要参照的 SPCMD, 并且使用所选 SPCMD 的 SPNDEN 位和 SPND, 决定表 17.10 所示的串行传送时的 RSPCK 延迟。有关下次存取延迟的定义, 请参照“17.4.4 传送格式”。

表 17.10 SPNDEN、SPND 和下次存取延迟值的关系

SPNDEN	SPND	下次存取延迟值
0	000 ~ 111	1 个 RSPCK+2 个 P1φ
1	000	1 个 RSPCK+2 个 P1φ
	001	2 个 RSPCK+2 个 P1φ
	010	3 个 RSPCK+2 个 P1φ
	011	4 个 RSPCK+2 个 P1φ
	100	5 个 RSPCK+2 个 P1φ
	101	6 个 RSPCK+2 个 P1φ
	110	7 个 RSPCK+2 个 P1φ
	111	8 个 RSPCK+2 个 P1φ

(h) 初始化流程

在 SPI 运行时，主控模式中使用此模块时的初始化流程例子如图 17.17 所示。有关中断控制器、直接存储器存取控制器和输入 / 输出端口的设定方法，请参照各块的说明。

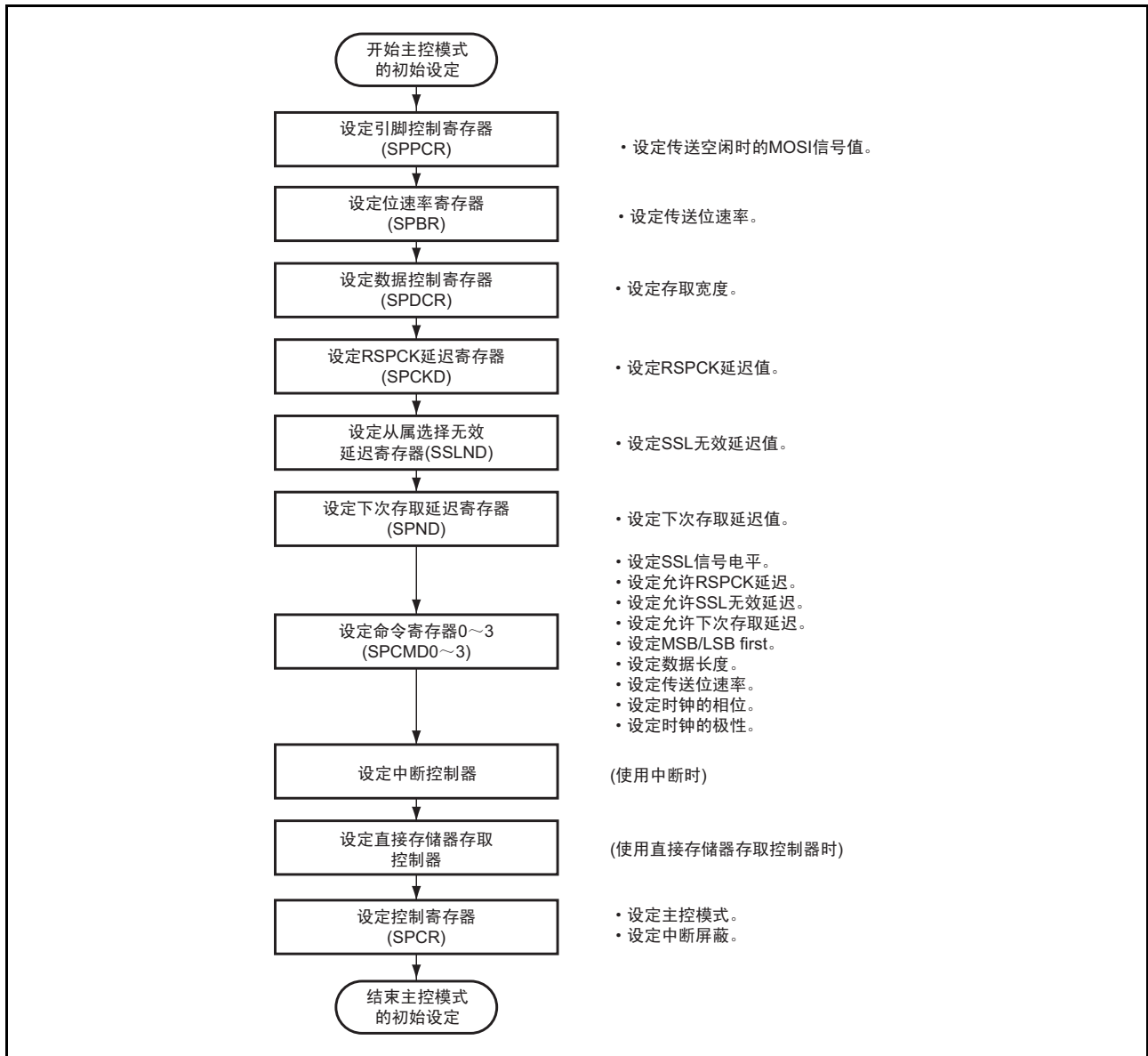


图 17.17 主控模式中的初始化流程例子

(i) 传送流程

在 SPI 运行时，主控模式中的传送流程如图 17.18 所示。

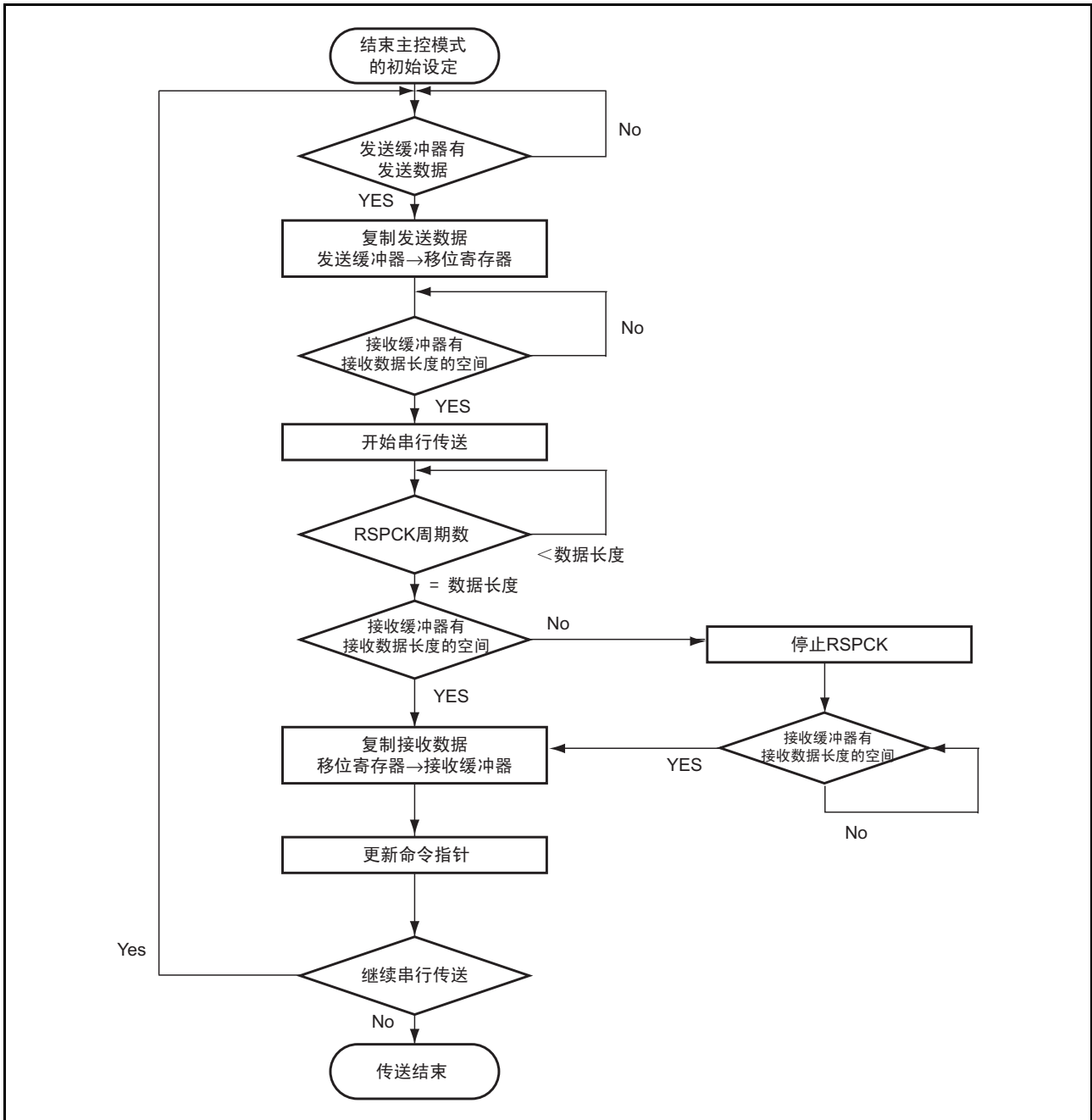


图 17.18 主控模式中的传送流程

(2) 从属模式的运行

(a) 串行传送的开始

当命令寄存器 0 (SPCMD0) 的 CPHA 位为“0”时，如果此模块检测到 SSL 输入信号有效，就需要开始驱动 MISO 输出信号的有效数据。因此，当 CPHA 为“0”时，SSL 输入信号的有效为串行传送的开始触发。

当 CPHA 位为“1”时，如果此模块在 SSL 输入信号有效的状态下检测到第一个 RSPCK 边沿，就需要开始驱动 MISO 输出信号的有效数据。因此，当 CPHA 为“1”时，SSL 信号有效状态下的第一个 RSPCK 边沿为串行传送的开始触发。

如果此模块在移位寄存器为空的状态下检测到串行传送的开始，就将移位寄存器改为满状态，以便不能在串行传送过程中将数据从发送缓冲器复制到移位寄存器。如果在开始串行传送前移位寄存器已满，此模块就保持移位寄存器的满状态。

与 CPHA 位的设定无关，此模块开始驱动 MISO 输出信号的时序为 SSL 信号的有效时序。此模块输出的数据是否有效因 CPHA 位的设定而不同。

有关此模块传送格式的详细内容，请参照“17.4.4 传送格式”。SSL 输入信号的极性取决于从属选择极性寄存器 (SSLP) 的 SSLP 位的设定值。

(b) 串行传送的结束

与命令寄存器 0 (SPCMD0) 的 CPHA 位无关，如果此模块检测到相当于最后采样时序的 RSPCK 边沿，就结束串行传送。在接收缓冲器有能保存接收数据的空间的情况下，此模块在串行传送结束后将接收数据从移位寄存器复制到数据寄存器 (SPDR) 的接收缓冲器。与 SPRF 位的值无关，此模块在结束串行传送后将移位寄存器改为空状态。在串行传送开始到串行传送结束的期间，如果此模块检测到 SSL 输入信号无效，就发生模式故障错误 (参照“17.4.6 错误检测”)。

最后的采样时序取因传送数据的位长而变，从属模式的数据长度取决于 SPCMD0 的 SPB3 ~ SPB0 位的设定值，SSL 输入信号的极性取决于从属选择极性寄存器 (SSLP) 的 SSLP 位的设定值。有关传送格式的详细内容，请参照“17.4.4 传送格式”。

(c) 从属模式时的注意点

当命令寄存器 0 (SPCMD0) 的 CPHA 位为“0”时，如果此模块检测到 SSL 输入信号的有效边沿，就开始串行传送。在图 17.4 所示的结构中，在从属模式中使用此模块时，因为 SSL 输入信号总是被固定为有效状态，所以 CPHA 位被置“0”的此模块无法正常开始串行传送。在 SSL 输入信号被固定为有效状态的结构中，要正常进行从属模式的发送和接收，必须将 CPHA 位置“1”。如果需要将 CPHA 位置“0”，就不能固定 SSL 输入信号。

(d) 突发传送

当命令寄存器 0 (SPCMD0) 的 CPHA 位为“1”时，能在保持 SSL 输入信号有效的状态下进行连续的串行传送 (突发传送)。当 CPHA 位为“1”时，从 SSL 输入信号有效状态的第一个 RSPCK 边沿到最后位的接收采样时序的期间相当于串行传送期间。因为能检测到存取的开始，所以即使 SSL 输入信号保持有效电平，也能进行突发传送。

当 CPHA 位为“0”时，原因和“17.4.8(2) (c) 从属模式时的注意点”相同，所以不能正确进行突发传送第 2 次以后的串行传送。

(e) 初始化流程

在 SPI 运行时，从属模式中使用此模块时的初始化流程例子如图 17.19 所示。有关中断控制器、直接存储器存取控制器和输入 / 输出端口的设定方法，请参照各块的说明。

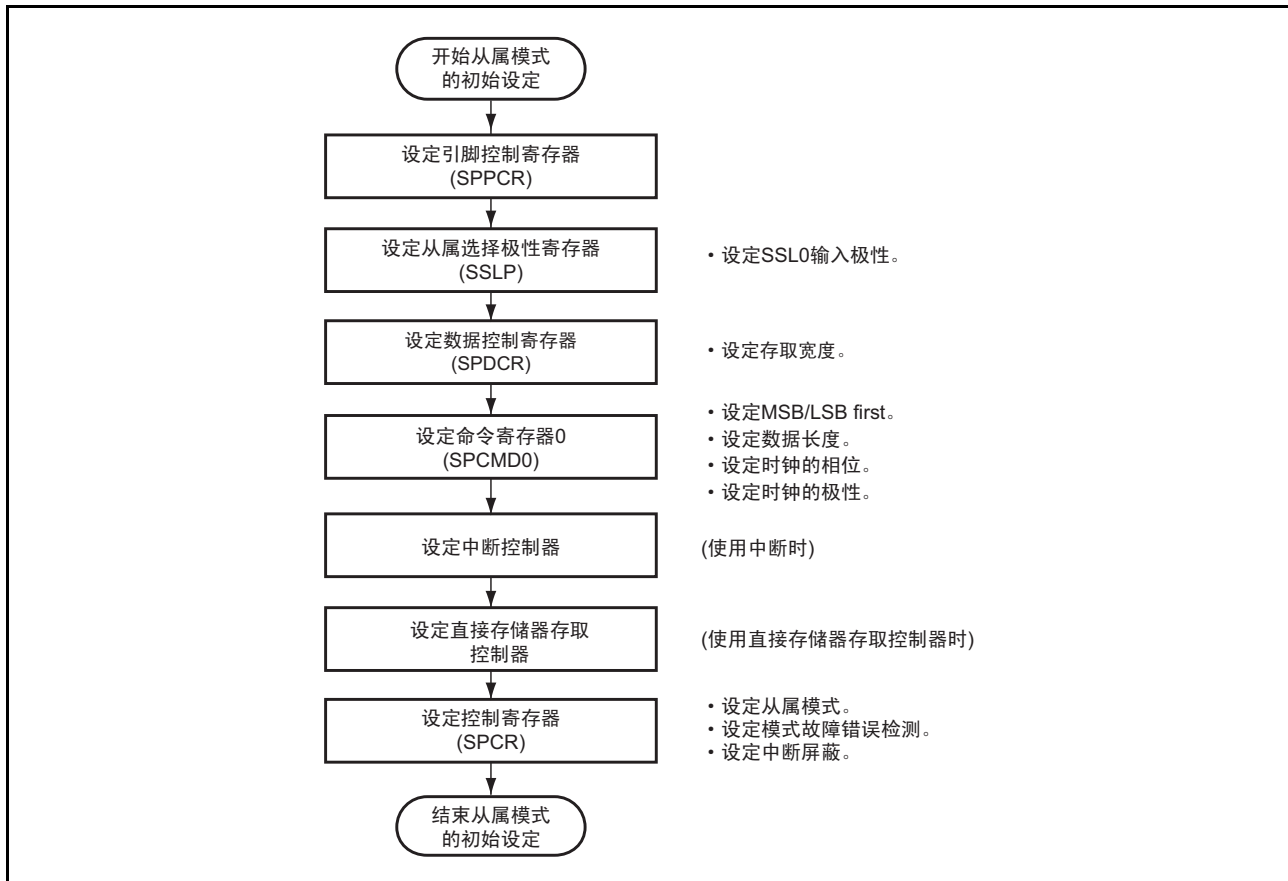


图 17.19 从属模式中的初始化流程例子

(f) 传送流程 (CPHA 位为“0”)

在 SPI 运行时，从属模式中（将命令寄存器 0 (SPCMD0) 的 CPHA 位置“0”的情况）的传送运行流程如图 17.20 所示。

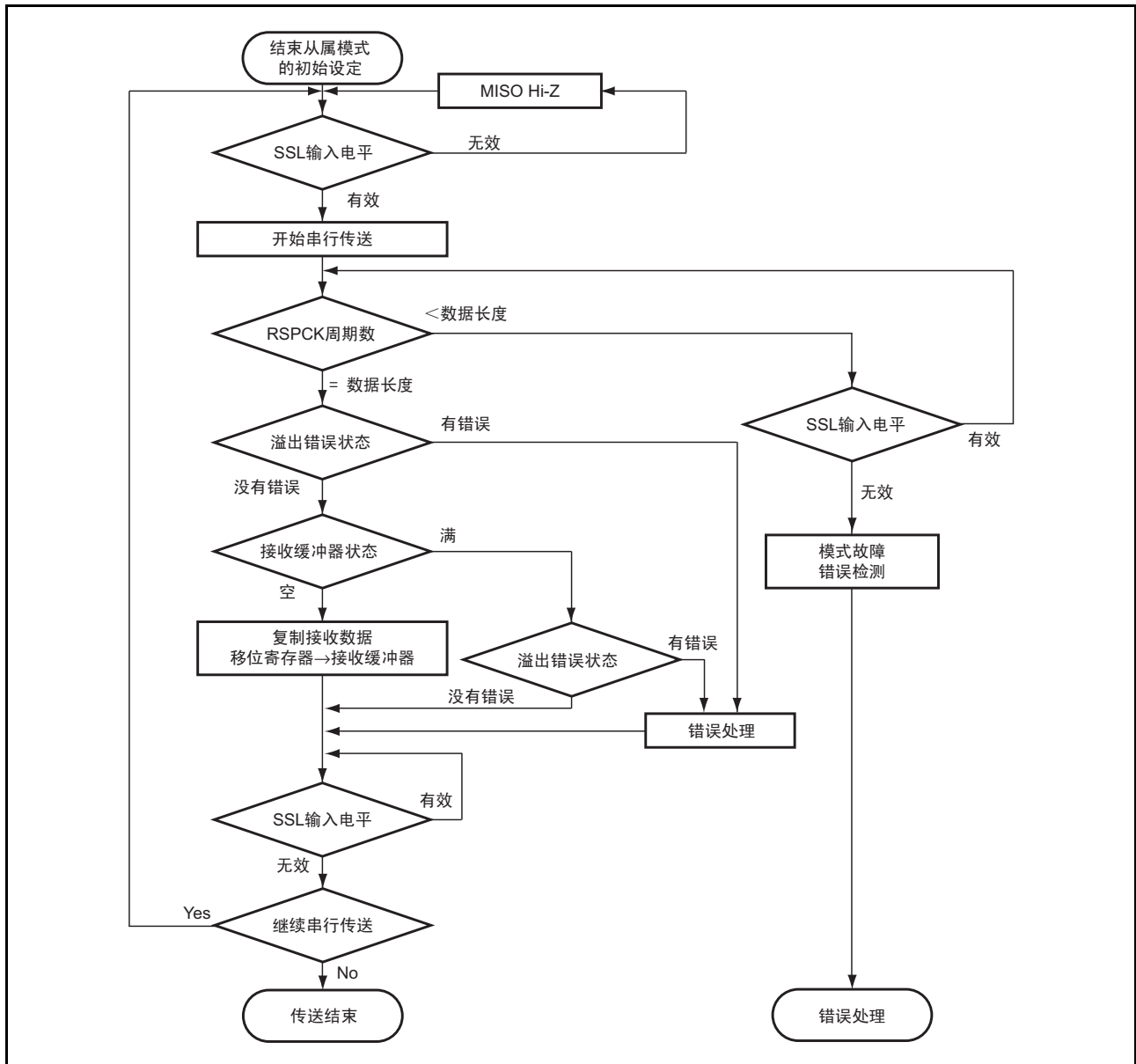


图 17.20 从属模式中的传送流程 (CPHA 位为“0”)

(g) 传送流程 (CPHA 位为“1”)

在 SPI 运行时，从属模式中（将命令寄存器 0 (SPCMD0) 的 CPHA 位被置“1”并且将控制寄存器 (SPCR) 的 MODFEN 位置“1”的情况）的传送运行流程如图 17.21 所示。如果在 MODFEN 位被置“0”的状态下开始串行传送，并且在 RSPCK 周期数短于数据长度的状态下将 SSL 输入电平置为无效，就不保证以后的运行。

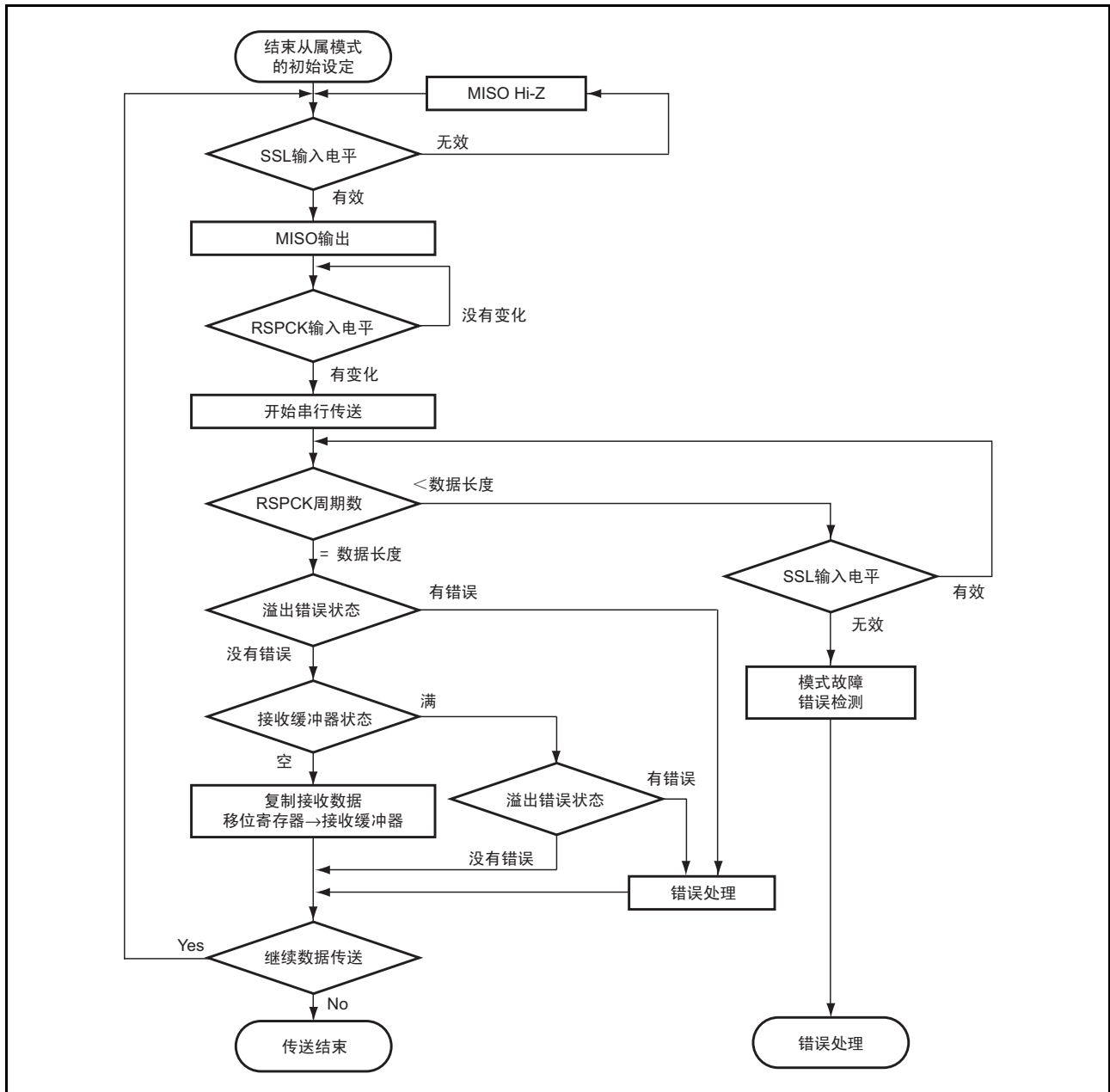


图 17.21 从属模式中的传送流程 (CPHA 位为“1”)

17.4.9 错误处理

错误处理如图 17.22 和图 17.23 所示。对于在主控模式和从属模式中发生的错误，能通过以下的错误处理从错误状态返回。



图 17.22 错误处理（溢出错误）

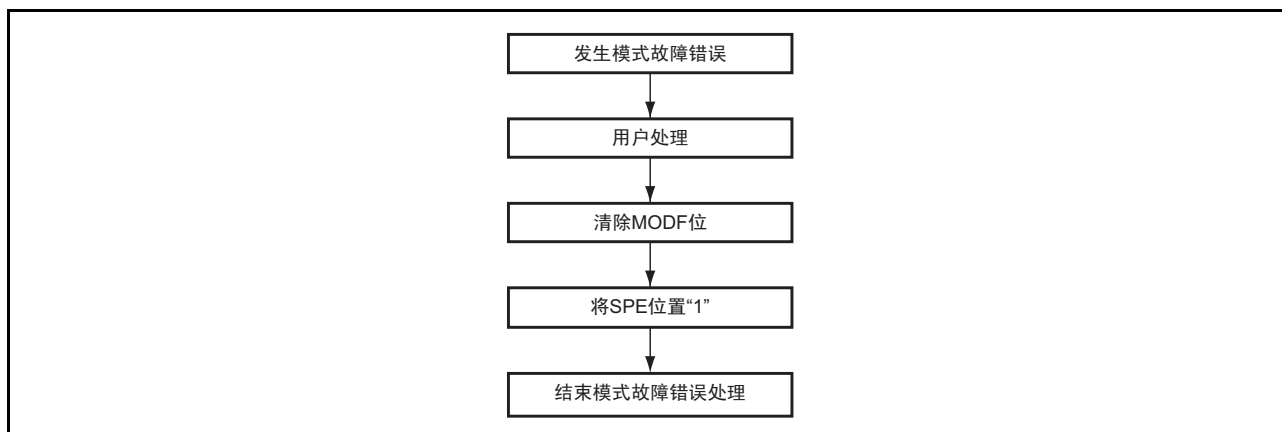


图 17.23 错误处理（模式故障错误）

17.4.10 环回模式

如果给引脚控制寄存器（SPPCR）的 SPLP 位写“1”，此模块就切断 MISO 引脚和移位寄存器之间以及 MOSI 引脚和移位寄存器之间的线路，连接移位寄存器的输入线路和输出线路（反相），此模式称为环回模式。如果在环回模式中进行串行传送，发送数据就为接收数据。在主机模式中，将此模块设定为环回模式时的移位寄存器的输入 / 输出线路的结构如图 17.24 所示。

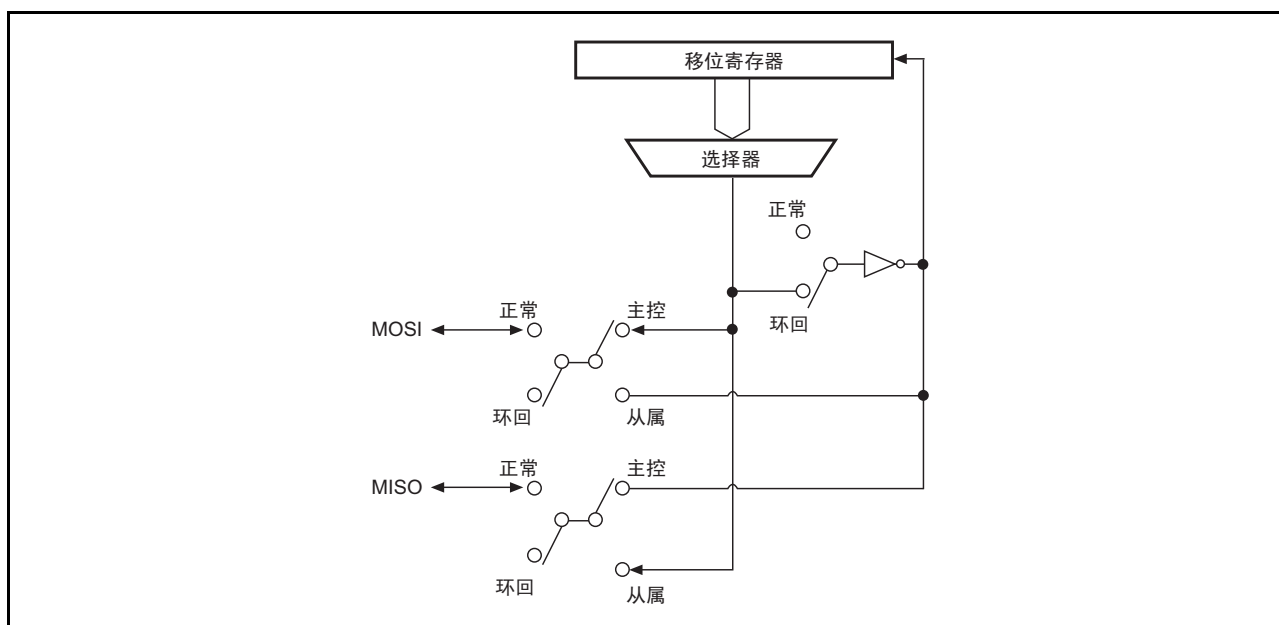


图 17.24 环回模式中的移位寄存器的输入 / 输出结构（主控模式）

17.4.11 中断源

中断源有接收缓冲器满、发送缓冲器空、模式故障和溢出。能通过接收缓冲器满和发送缓冲器空的中断请求，启动直接存储器存取控制器进行数据传送。

中断源如表 17.11 所示。

一旦表 17.11 中的中断条件成立，就产生中断。必须在通过 CPU 或者直接存储器存取控制器进行的数据传送中清除中断源。

表 17.11 中断源

名称	中断源	略称	中断条件	直接存储器存取控制器的启动
SPRI	接收缓冲器满	RXI	$(SPRIE=1) \cdot (SPRF=1)$	○
SPTI	发送缓冲器空	TXI	$(SPTIE=1) \cdot (SPTEF=1)$	○
SPEI	模式故障	MOI	$(SPEIE=1) \cdot (MODF=1)$	—
	溢出	OVI	$(SPEIE=1) \cdot (OVRF=1)$	—

18. 瑞萨四路串行外围接口

本 LSI 有独立的 2 个通道的瑞萨四路串行外围接口。

18.1 特点

此模块有以下特点：

- 能通过 Single/Dual/Quad-SPI 运行，和串行闪存进行通信。

Single-SPI 运行

使用 QMO (MasterOut)、QMI (MasterIn)、QSSL (Slave Select) 和 QSPCLK (SPI Clock) 共 4 条线存取串行闪存。

QMO 引脚为输出引脚，QMI 引脚为输入引脚，QSSL 引脚和 QSPCLK 引脚为输出引脚。

Dual-SPI 运行

使用 QIO1-0、QSSL、QSPCLK 共 4 条线存取串行闪存。

QSSL 引脚和 QSPCLK 引脚为输出引脚，QIO1-0 引脚为双向引脚。

Quad-SPI 运行

使用 QIO3-0、QSSL 和 QSPCLK 共 6 条线存取串行闪存。

QSSL 引脚和 QSPCLK 引脚为输出引脚，QIO3-0 引脚为双向引脚。

- 传送数据长度

能通过 8 位 ~ 128G 位进行传送。

但是，以 8 位、16 位或者 32 位传送为 1 个基本传送，重复进行 1 ~ 4294967296 次的连续传送。

- 位速率

能设定 1 ~ 4080 的 QSPCLK 分频比（但是，发送时的分频比至少为 2 分频）。

通过内部波特率发生器对 P1 ϕ 进行分频，生成 QSPCLK。

- 缓冲器结构

发送缓冲器：8 位 \times 32 个

接收缓冲器：8 位 \times 32 个

- 移位寄存器

分别有发送专用和接收专用的 32 位移位寄存器。

- QSSL 引脚的控制

能设定从 QSSL 信号有效到 QSPCLK 运行为止的延迟（时钟延迟）。

设定范围：0、1.5 ~ 8.5 个 QSPCLK 设定单位：1 个 QSPCLK

能设定从 QSPCLK 停止到 QSSL 输出无效为止的延迟（QSSL 无效延迟）。

设定范围：0 ~ 8 个 QSPCLK 设定单位：1 个 QSPCLK

能设定下次存取的 QSSL 输出有效的等待（下次存取延迟）。

设定范围：0 ~ 8 个 QSPCLK 设定单位：1 个 QSPCLK

在传送结束后到下次存取为止的期间，能保持 QSSL 的输出值。

能更改 QSSL 的极性。

- 传送控制

能按顺序循环执行最多由 4 个命令构成的传送方式。

Single-SPI 或者 Dual/Quad-SPI 的写操作：在 SPI 功能有效的状态下，通过给发送缓冲器写数据来启动传送。

Dual/Quad-SPI 的读操作：在接收缓冲器有传送数据长度的空间并且 SPI 功能有效的状态下启动传送。

在 QSSL 无效期间，能设定 QIO3-0 引脚输出和 QMO 输出。

在 Single/Dual-SPI 模式中，能设定 QIO3-2 引脚输出。

- 中断源
 - 可屏蔽的中断源
 - 接收缓冲器满中断
 - 发送缓冲器空中断
- 其他
 - 环回模式
 - 有初始化功能。

18.2 输入 / 输出引脚

引脚结构如表 18.1 所示。

表 18.1 引脚结构

通道	名称	引脚名	输入 / 输出	功能
0	时钟引脚	QSPCLK_0	输出	时钟输出
	主控发送数据引脚 / 数据 0 引脚 *2	QMO_0/QIO0_0	输入 / 输出	主控发送数据 / 数据 0
	主控输入数据引脚 / 数据 1 引脚 *2	QMI_0/QIO1_0	输入 / 输出	主控输入数据 / 数据 1
	数据 2 引脚 *3	QIO2_0	输入 / 输出	数据 2
	数据 3 引脚 *3	QIO3_0	输入 / 输出	数据 3
	从属选择引脚	QSSL_0	输出	从属选择
1	时钟引脚	QSPCLK_1	输出	时钟输出
	主控发送数据引脚 / 数据 0 引脚 *2	QMO_1/QIO0_1	输入 / 输出	主控发送数据 / 数据 0
	主控输入数据引脚 / 数据 1 引脚 *2	QMI_1/QIO1_1	输入 / 输出	主控输入数据 / 数据 1
	数据 2 引脚 *3	QIO2_1	输入 / 输出	数据 2
	数据 3 引脚 *3	QIO3_1	输入 / 输出	数据 3
	从属选择引脚	QSSL_1	输出	从属选择

【注】 *1 本章节中省略引脚的通道，记述为 QSPCLK、QMO/QIO0、QMI/QIO1、QIO2、QIO3、QSSL。

*2 在 Single-SPI 模式中为 QMO/QMI，在 Dual/Quad-SPI 中为 QIO0/QIO1。

*3 在 Single/Dual-SPI 中通过设定寄存器输出固定值，在 Quad-SPI 模式中为 QIO2/QIO3。

18.3 寄存器说明

寄存器结构如表 18.2 所示。

表 18.2 寄存器结构

通道	名称	略称	R/W	初始值	地址	存取长度
0	控制寄存器 _0	SPCR_0	R/W	H'00	H'E8033800	8、16、32
	从属选择极性寄存器 _0	SSLP_0	R/W	H'00	H'E8033801	8、16、32
	引脚控制寄存器 _0	SPPCR_0	R/W	H'06	H'E8033802	8、16、32
	状态寄存器 _0	SPSR_0	R/(W)*2	H'60	H'E8033803	8、16、32
	数据寄存器 _0	SPDR_0	R/W	不定值	H'E8033804	8、16、32
	顺序控制寄存器 _0	SPSCR_0	R/W	H'00	H'E8033808	8、16、32
	顺序状态寄存器 _0	SPSSR_0	R	H'00	H'E8033809	8、16、32
	位速率寄存器 _0	SPBR_0	R/W	H'FF	H'E803380A	8、16、32

通道	名称	略称	R/W	初始值	地址	存取长度
0	数据控制寄存器 _0	SPDCR_0	R/W	H'00	H'E803380B	8、16、32
	时钟延迟寄存器 _0	SPCKD_0	R/W	H'00	H'E803380C	8、16、32
	从属选择无效延迟寄存器 _0	SSLND_0	R/W	H'00	H'E803380D	8、16、32
	下次存取延迟寄存器 _0	SPND_0	R/W	H'00	H'E803380E	8、16、32
	命令寄存器 0_0	SPCMD0_0	R/W	H'E001	H'E8033810	16、32
	命令寄存器 1_0	SPCMD1_0	R/W	H'E001	H'E8033812	16、32
	命令寄存器 2_0	SPCMD2_0	R/W	H'E001	H'E8033814	16、32
	命令寄存器 3_0	SPCMD3_0	R/W	H'E001	H'E8033816	16、32
	缓冲数据计数置位寄存器 _0	SPBDCR_0	R	H'0000	H'E803381A	16、32
	数据长度倍数设定寄存器 0_0	SPBMUL0_0	R/W	H'00000001	H'E803381C	32
	数据长度倍数设定寄存器 1_0	SPBMUL1_0	R/W	H'00000001	H'E8033820	32
	数据长度倍数设定寄存器 2_0	SPBMUL2_0	R/W	H'00000001	H'E8033824	32
	数据长度倍数设定寄存器 3_0	SPBMUL3_0	R/W	H'00000001	H'E8033828	32
1	控制寄存器 _1	SPCR_1	R/W	H'00	H'E8034000	8、16、32
	从属选择极性寄存器 _1	SSLP_1	R/W	H'00	H'E8034001	8、16、32
	引脚控制寄存器 _1	SPPCR_1	R/W	H'06	H'E8034002	8、16、32
	状态寄存器 _1	SPSR_1	R/(W)*2	H'60	H'E8034003	8、16、32
	数据寄存器 _1	SPDR_1	R/W	不定值	H'E8034004	8、16、32
	顺序控制寄存器 _1	SPSCR_1	R/W	H'00	H'E8034008	8、16、32
	顺序状态寄存器 _1	SPSSR_1	R	H'00	H'E8034009	8、16、32
	位速率寄存器 _1	SPBR_1	R/W	H'FF	H'E803400A	8、16、32
	数据控制寄存器 _1	SPDCR_1	R/W	H'00	H'E803400B	8、16、32
	时钟延迟寄存器 _1	SPCKD_1	R/W	H'00	H'E803400C	8、16、32
	从属选择无效延迟寄存器 _1	SSLND_1	R/W	H'00	H'E803400D	8、16、32
	命令寄存器 0_1	SPCMD0_1	R/W	H'E001	H'E8034010	16、32
	命令寄存器 1_1	SPCMD1_1	R/W	H'E001	H'E8034012	16、32
	命令寄存器 2_1	SPCMD2_1	R/W	H'E001	H'E8034014	16、32
	命令寄存器 3_1	SPCMD3_1	R/W	H'E001	H'E8034016	16、32
	缓冲控制寄存器 _1	SPBFCR_1	R/W	H'00	H'E8034018	8、16、32
	缓冲数据计数置位寄存器 _1	SPBDCR_1	R	H'0000	H'E803401A	16、32
	数据长度倍数设定寄存器 0_1	SPBMUL0_1	R/W	H'00000001	H'E803401C	32
	数据长度倍数设定寄存器 1_1	SPBMUL1_1	R/W	H'00000001	H'E8034020	32
	数据长度倍数设定寄存器 2_1	SPBMUL2_1	R/W	H'00000001	H'E8034024	32
数据长度倍数设定寄存器 3_1	SPBMUL3_1	R/W	H'00000001	H'E8034028	32	

【注】 *1 本章节中的寄存器名省略通道。

*2 为了清除标志，只能写“0”。

18.3.1 控制寄存器（SPCR）

控制寄存器（SPCR）是设定运行模式的寄存器。

位:	7	6	5	4	3	2	1	0
	SPRIE	SPE	SPTIE	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R	R/W	R	R	R

位	位名	初始值	R/W	说明
7	SPRIE	0	R/W	接收中断允许 当接收缓冲器的接收数据量大于等于指定的触发数并且状态寄存器的标志为“1”时，允许或者禁止接收中断请求。 0: 禁止中断请求 1: 允许中断请求
6	SPE	0	R/W	SPI 功能允许 通过将此位置“1”，将 SPI 模块的功能置为有效。 如果将此位置“0”，就对部分模块进行初始化。 0: 功能无效 1: 功能有效
5	SPTIE	0	R/W	发送中断允许 当发送缓冲器的发送数据量小于等于指定的触发数并且状态寄存器的标志为“1”时，允许或者禁止发送中断请求。 0: 禁止中断请求 1: 允许中断请求
4	—	0	R	保留位 读写值都为“0”。
3	—	0	R/W	保留位 写入值总是“1”。不保证写“0”时的运行。
2~0	—	全0	R	保留位 读写值都为“0”。

18.3.2 从属选择极性寄存器（SSLP）

从属选择极性寄存器（SSLP）是设定 QSSL 信号极性的寄存器。如果在控制寄存器（SPCR）的 SPI 功能允许位（SPE）为“1”的状态下改写此寄存器，就不保证以后的运行。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SSLP
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
7~1	—	全0	R	保留位 读写值都为“0”。
0	SSLP	0	R/W	QSSL 信号极性设定 此位设定 QSSL 信号的极性。 0: QSSL 信号为 Low 电平有效 1: QSSL 信号为 High 电平有效

18.3.3 引脚控制寄存器 (SPPCR)

引脚控制寄存器 (SPPCR) 是设定引脚模式的寄存器。如果在控制寄存器 (SPCR) 的 SPI 功能允许位 (SPE) 为 “1” 的状态下改写此寄存器, 就不保证以后的运行。

位:	7	6	5	4	3	2	1	0
	—	—	MOIFE	MOIFV	—	IO3FV	IO2FV	SPLP
初始值:	0	0	0	0	0	1	1	0
R/W:	R	R	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7、6	—	全 0	R	保留位 读写值都为 “0”。
5	MOIFE	0	R/W	数据输出空闲值固定允许 在 QSSL 无效期间或者突发传输时的 QSSL 保持期间, 此位用于固定输出引脚的值。 在 Single-SPI 模式中, 此位对 QMO 有效; 在 Dual-SPI 模式中, 此位对 QIO1-0 有效; 在 Quad-SPI 模式中, 此位对 QIO3-0 有效。 0: 输出值为上次传输的最后数据 1: 输出值为 MOIFV 位的设定值 【注】 在 Dual/Quad-SPI 模式中, 读后的 QIO1-0/QIO3-0 与此位的设定无关, 为 Hi-Z (参照 “18.4.2 引脚的控制”)。
4	MOIFV	0	R/W	数据输出空闲时的固定值 当 MOIFE 位为 “1” 时, 根据此位的设定值决定 QSSL 无效期间的输出引脚的值。 0: 输出引脚空闲值为 “0” 1: 输出引脚空闲值为 “1”
3	—	0	R	保留位 读写值都为 “0”。
2	IO3FV	1	R/W	Single/Dual-SPI 时的 QIO3 输出固定值 在 Single/Dual-SPI 模式中, 此位固定 QIO3 引脚的输出方向。此位只在 Single/Dual-SPI 模式中有效, 并且此时不影响 MOIFE 位和 MOIFV 位的值。 0: QIO3 固定为 “0” 1: QIO3 固定为 “1”
1	IO2FV	1	R/W	Single/Dual-SPI 时的 QIO2 输出固定值 在 Single/Dual-SPI 模式中, 此位固定 QIO2 引脚的输出方向。此位只在 Single/Dual-SPI 模式中有效, 并且此时不影响 MOIFE 位和 MOIFV 位的设定值。 0: QIO2 固定为 “0” 1: QIO2 固定为 “1”
0	SPLP	0	R/W	环回模式 如果将 SPLP 位置 “1”, 此模块就切断数据输入 / 输出引脚和发送 / 接收移位寄存器之间的线路, 连接发送 / 接收移位寄存器之间的线路。 0: 正常模式 1: 环回模式 【注】 如果在 Dual/Quad-SPI 中设定为环回模式, 就必须将命令寄存器 0 ~ 3 (SPCMD0 ~ 3) 的 SPRW 位置 “0” (写操作)。

18.3.4 状态寄存器 (SPSR)

状态寄存器 (SPSR) 保存表示此模块运行状态的标志。

位:	7	6	5	4	3	2	1	0
	SPRFF	TEND	SPTEF	—	—	—	—	—
初始值:	0	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7	SPRFF	0	R	接收缓冲器满标志 表示接收缓冲器的接收数据量大于等于缓冲控制寄存器的接收缓冲器的数据量触发指定的值。 0: 接收缓冲器的接收数据量小于接收缓冲器的数据量触发指定的值 1: 接收缓冲器的接收数据量大于等于接收缓冲器的数据量触发指定的值 [清除条件] <ul style="list-style-type: none"> • 读接收缓冲器, 直到接收缓冲器的接收数据量小于接收缓冲器的数据量触发指定的值时 • 当将接收缓冲器的数据复位置为有效时 • 上电复位 [置位条件] <ul style="list-style-type: none"> • 当接收缓冲器的接收数据量大于等于接收缓冲器的数据量触发指定的值时
6	TEND	1	R	通信结束标志 当此位为“1”时, 表示通信结束; 当此位为“0”时, 表示通信未结束。 [清除条件] <ul style="list-style-type: none"> • 当将发送数据从发送寄存器传送到发送移位寄存器时 • 在 Dual/Quad-SPI 模式中开始接收时 [置位条件] <ul style="list-style-type: none"> • 在结束串行传送时发送缓冲器的数据保存个数为空 (虚拟发送允许位 (TXDMY) 为“1”的情况除外)。 • 在串行传送结束时接收缓冲器没有下一次接收数据长度的空间。
5	SPTEF	1	R	发送缓冲器空标志 表示发送缓冲器的发送数据量小于等于缓冲控制寄存器的发送缓冲器的数据量触发指定的值。 0: 发送缓冲器的发送数据量大于发送缓冲器的数据量触发指定的值 1: 发送缓冲器的发送数据量小于等于发送缓冲器的数据量触发指定的值 [清除条件] <ul style="list-style-type: none"> • 写发送缓冲器, 直到发送缓冲器的发送数据量大于发送缓冲器的数据量触发指定的值时 [置位条件] <ul style="list-style-type: none"> • 当发送缓冲器的发送数据量小于等于发送缓冲器的数据量触发指定的值时 • 当将发送缓冲器的数据复位置为有效时 • 上电复位
4 ~ 0	—	全 0	R	保留位 读写值都为“0”。

18.3.5 数据寄存器 (SPDR)

数据寄存器 (SPDR) 是存取发送 / 接收数据缓冲器的寄存器。

发送缓冲器 (SPTXB) 和接收缓冲器 (SPRXB) 是独立的缓冲器, 这些缓冲器被映像到 SPDR。如果写此寄存器, 就将数据写到发送缓冲器 (SPTXB); 如果读此寄存器, 就从接收缓冲器 (SPRXB) 读数据。

必须用字节、字或者长字存取此寄存器。

各存取长度的有效数据如下:

长字: 31 ~ 0 位

字: 31 ~ 16 位

字节: 31 ~ 24 位

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24	SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

18.3.6 顺序控制寄存器 (SPSCR)

顺序控制寄存器 (SPSCR) 设定此模块的顺序控制方式。如果在控制寄存器 (SPCR) 的 SPI 功能允许位 (SPE) 为“1”的状态下改写此寄存器, 就不保证以后的运行。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPSC1	SPSC0
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 2	—	全 0	R	保留位 读写值都为“0”。
1	SPSC1	0	R/W	顺序控制设定 设定此模块的顺序运行。 根据 SPSC[1:0] 位的设定值, 按顺序参照命令寄存器 0 ~ 3 (SPCMD0 ~ 3)。 要参照的命令寄存器 (SPCMD) 序号: 00: 0→0→…… 01: 0→1→0→…… 10: 0→1→2→0→…… 11: 0→1→2→3→0→……
0	SPSC0	0	R/W	

18.3.7 顺序状态寄存器 (SPSSR)

顺序状态寄存器 (SPSSR) 是表示顺序状态的寄存器。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPSS1	SPSS0
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 2	—	全 0	R	保留位 读写值都为“0”。
1	SPSS1	0	R	顺序状态 在顺序控制中表示当前正在参照的命令寄存器 0 ~ 3 (SPCMD0 ~ 3)。 00: SPCMD0 01: SPCMD1 10: SPCMD2 11: SPCMD3
0	SPSS0	0	R	

18.3.8 位速率寄存器（SPBR）

位速率寄存器（SPBR）是设定位速率的寄存器。如果在控制寄存器（SPCR）的 SPI 功能允许位（SPE）为“1”的状态下改写此寄存器，就不保证以后的运行。

位:	7	6	5	4	3	2	1	0
	SPBR7	SPBR6	SPBR5	SPBR4	SPBR3	SPBR2	SPBR1	SPBR0
初始值:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位速率取决于此寄存器的设定值和命令寄存器 0 ~ 3（SPCMD0 ~ 3）的位速率分频设定位（BRDV[1:0]）的设定值。

如果将此寄存器置“0”，就为 1 分频。

SPBR≠0 时的位速率的计算式如下。

在计算式中，n 为此寄存器的设定值（1、……、255），N 为位速率分频位（BRDV[1:0]）的设定值（0、1、2、3）。

$$\text{位速率} = \frac{f(P1\phi)}{2 \times n \times 2^N}$$

SPBR 寄存器、BRDV[1:0] 位的设定值和位速率的关系如表 18.3 所示。

表 18.3 SPBR 寄存器、BRDV[1:0] 位的设定值和位速率

SPBR 的设定值 (n)	BRDV[1:0] 的设定值 (N)	分频比	位速率	
			P1φ=50MHz	P1φ=66.67MHz
0	0	1	50.0Mbps	66.67Mbps
1	0	2	25.00Mbps	33.33Mbps
2	0	4	12.50Mbps	16.76Mbps
3	0	6	8.33Mbps	11.11Mbps
4	0	8	6.25Mbps	8.33Mbps
5	0	10	5.00Mbps	6.67Mbps
6	0	12	4.16Mbps	5.56Mbps
6	1	24	2.08Mbps	2.78Mbps
6	2	48	1.04Mbps	1.39Mbps
6	3	96	520kbps	694.48kbps
255	3	4080	12.25kbps	16.34kbps

【注】 在发送时，必须设定位速率寄存器（SPBR）和位速率分频位（BRDV），使 QSPCLK 不小于 2 分频。

18.3.9 数据控制寄存器 (SPDCR)

数据控制寄存器 (SPDCR) 是设定虚拟数据发送允许的寄存器。

位:	7	6	5	4	3	2	1	0
	TXDMY	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7	TXDMY	0	R/W	虚拟数据发送允许 在 Single-SPI 运行时, 允许或者禁止在发送缓冲器没有发送数据时从 QMO 引脚输出虚拟数据。 当发送缓冲器没有发送数据并且 TXDMY 被置“1”时, QMO 引脚输出“0”作为虚拟数据。 能在状态寄存器 (SPSR) 的通信结束标志 (TEND) 为“1”时更改此位。如果在其他状态下更改此位, 就不保证运行。 0: 禁止发送虚拟数据 1: 允许发送虚拟数据
6~0	—	全 0	R	保留位 读写值都为“0”。

18.3.10 时钟延迟寄存器 (SPCKD)

时钟延迟寄存器 (SPCKD) 设定从 QSSL 信号有效开始到 QSPCLK 振荡的期间 (时钟延迟)。当命令寄存器 0~3 (SPCMD0~3) 的时钟延迟设定允许位 (SCKDEN) 为“1”时, 此寄存器的设定值有效。如果在控制寄存器 (SPCR) 的 SPI 功能允许位 (SPE) 为“1”的状态下改写此寄存器, 就不保证以后的运行。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SCKDL2	SCKDL1	SCKDL0
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7~3	—	全 0	R	保留位 读写值都为“0”。
2	SCKDL2	0	R/W	时钟延迟设定 当命令寄存器 0~3 (SPCMD0~3) 的 SCKDEN 位为“1”时, 设定从 QSSL 信号有效开始到 QSPCLK 振荡的期间 (时钟延迟)。 000: 1.5 个 QSPCLK 001: 2.5 个 QSPCLK 010: 3.5 个 QSPCLK 011: 4.5 个 QSPCLK 100: 5.5 个 QSPCLK 101: 6.5 个 QSPCLK 110: 7.5 个 QSPCLK 111: 8.5 个 QSPCLK
1	SCKDL1	0	R/W	
0	SCKDL0	0	R/W	

18.3.11 从属选择无效延迟寄存器 (SSLND)

从属选择无效延迟寄存器 (SSLND) 设定从发送串行传送最后的 QSPCLK 边沿到将 QSSL 信号置为无效的期间 (QSSL 无效延迟)。当命令寄存器 0 ~ 3 (SPCMD0 ~ 3) 的 QSSL 无效延迟设定允许位 (SLNDEN) 为 “1” 时, 此寄存器的设定值有效。如果在控制寄存器 (SPCR) 的 SPI 功能允许位 (SPE) 为 “1” 的状态下改写此寄存器, 就不保证以后的运行。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SLNDL2	SLNDL1	SLNDL0
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 3	—	全 0	R	保留位 读写值都为 “0”。
2	SLNDL2	0	R/W	QSSL 无效延迟设定 当命令寄存器 0 ~ 3 (SPCMD0 ~ 3) 的 SLNDEN 位为 “1” 时, 设定从发送串行传送最后的 QSPCLK 边沿到将 QSSL 信号置为无效的期间 (QSSL 无效延迟)。 000: 1 个 QSPCLK 001: 2 个 QSPCLK 010: 3 个 QSPCLK 011: 4 个 QSPCLK 100: 5 个 QSPCLK 101: 6 个 QSPCLK 110: 7 个 QSPCLK 111: 8 个 QSPCLK
1	SLNDL1	0	R/W	
0	SLNDL0	0	R/W	

18.3.12 下次存取延迟寄存器 (SPND)

下次存取延迟寄存器 (SPND) 设定从串行传送结束到下一次串行传送的期间 (下次存取延迟)。当命令寄存器 0 ~ 3 (SPCMD0 ~ 3) 的下次存取延迟设定允许位 (SPNDEN) 为 “1” 时, 此寄存器的设定值有效。如果在控制寄存器 (SPCR) 的 SPI 功能允许位 (SPE) 为 “1” 的状态下改写此寄存器, 就不保证以后的运行。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SPNDL2	SPNDL1	SPNDL0
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 3	—	全 0	R	保留位 读写值都为 “0”。
2	SPNDL2	0	R/W	下次存取延迟设定 当命令寄存器 0 ~ 3 (SPCMD0 ~ 3) 的下次存取延迟设定允许位 (SPNDEN) 为 “1” 时, 设定从串行传送结束到下一次串行传送开始的期间 (下次存取延迟)。 000: 1 个 QSPCLK 001: 2 个 QSPCLK 010: 3 个 QSPCLK 011: 4 个 QSPCLK 100: 5 个 QSPCLK 101: 6 个 QSPCLK 110: 7 个 QSPCLK 111: 8 个 QSPCLK
1	SPNDL1	0	R/W	
0	SPNDL0	0	R/W	

18.3.13 命令寄存器 n (SPCMDn) (n=0、1、2、3)

此模块有 4 个命令寄存器 (SPCMD0 ~ 3)，这些寄存器用于设定传送格式。根据顺序控制寄存器 (SPSCR) 的设定，参照 SPCMD0 ~ 3，执行所参照的命令寄存器设定的串行传送。

在状态寄存器 (SPSR) 的通信结束标志 (TEND) 为通信未结束的状态下，如果改写正在参照的命令寄存器，就不保证以后的运行。能通过顺序状态寄存器 (SPSSR) 确认正在参照的命令寄存器。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0	SSLKP	SPIMOD ₁	SPIMOD ₀	SPRW	BRDV1	BRDV0	CPOL	CPHA
初始值:	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	SCKDEN	1	R/W	<p>时钟延迟设定允许</p> <p>设定从将 QSSL 信号置为有效到 QSPCLK 振荡的期间 (时钟延迟)。当此位为“0”时，时钟延迟期间为 0 个 QSPCLK；当此位为“1”时，根据时钟延迟寄存器 (SPCKD) 的设定值，开始 QSPCLK 的振荡。只有在通过多个命令使 QSSL 持续有效的第 2 次以后的连续存取中，才能将此位置“0”，否则就必须置“1”。</p> <p>0: 时钟延迟期间为 0 个 QSPCLK 1: 时钟延迟期间为时钟延迟寄存器 (SPCKD) 的设定值</p>
14	SLNDEN	1	R/W	<p>QSSL 无效延迟设定允许</p> <p>设定从 QSPCLK 停止振荡到将 QSSL 信号置为无效的期间 (QSSL 无效延迟)。当此位为“0”时，QSSL 无效延迟期间为 0 个 QSPCLK；当此位为“1”时，根据 QSSL 无效延迟寄存器 (SSLND) 的设定值，将 QSSL 置为无效。在通过多个命令使 QSSL 持续有效的连续存取中并且不是最后的命令时，能将此位置“0”，否则就必须置“1”。</p> <p>0: QSSL 无效延迟期间为 0 个 QSPCLK 1: QSSL 无效延迟期间为 QSSL 无效延迟寄存器 (SSLND) 的设定值</p>
13	SPNDEN	1	R/W	<p>下次存取延迟设定允许</p> <p>设定从串行传送结束到下一次串行传送开始的期间 (下次存取延迟)。当此位为“0”时，下次存取延迟期间为 0 个 QSPCLK；当此位为“1”时，根据下次存取延迟寄存器 (SPND) 的设定值，开始下一次串行传送。在通过多个命令使 QSSL 持续有效的连续存取中并且不是最后的命令时，能将此位置“0”，否则就必须置“1”。</p> <p>0: 下次存取延迟期间为 0 个 QSPCLK 1: 下次存取延迟期间为下次存取延迟寄存器 (SPND) 的设定值</p>
12	LSBF	0	R/W	<p>LSB first 设定</p> <p>此位将数据格式设定为 MSB first 或者 LSB first。</p> <p>0: MSB first 1: LSB first</p>

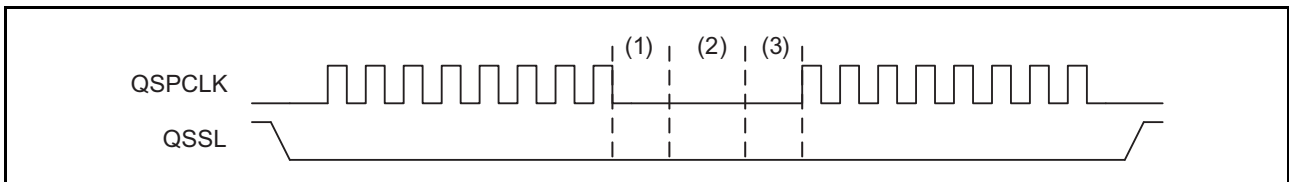
位	位名	初始值	R/W	说明
11 10 9 8	SPB3 SPB2 SPB1 SPB0	0 0 0 0	R/W R/W R/W R/W	<p>传送数据长度设定</p> <p>设定串行传送的基本传送数据长度。当以 LSB first 进行传送时，以此位设定的数据宽度将传送数据的位序颠倒。实际传送的数据长度为此位的设定值乘传送数据长度倍数设定寄存器 0 ~ 3 (SPBMUL0 ~ 3) 的设定值后的值。</p> <p>0000: 8 位 (1 字节) 0001: 16 位 (2 字节) 0010: 32 位 (4 字节) 0011 ~ 1111: 禁止设定</p>
7	SSLKP	0	R/W	<p>QSSL 信号电平保持</p> <p>设定在当前命令对应的传送结束到下一个命令对应的传送开始的期间是否保持 QSSL 信号。能通过将此位置“1”，在 QSSL 信号有效的状态下进行下次存取。</p> <p>0: 在传送结束后将 QSSL 信号置为无效 1: 在从传送结束后到下次存取开始前保持 QSSL 信号电平</p>
6 5	SPIMOD1 SPIMOD0	0 0	R/W R/W	<p>SPI 运行模式</p> <p>设定运行模式。</p> <p>运行模式从 Single/Dual/Quad-SPI 中选择。</p> <p>00: Single-SPI 01: Dual-SPI 10: Quad-SPI 11: 禁止设定</p>
4	SPRW	0	R/W	<p>SPI 读写存取设定</p> <p>设定 Dual/Quad-SPI 模式中的存取方向。</p> <p>在 Single-SPI 模式中，此位无效。</p> <p>0: 写操作 (QIO1-0/QIO3-0 为输出) 1: 读操作 (QIO1-0/QIO3-0 为输入)</p>
3 2	BRDV1 BRDV0	0 0	R/W R/W	<p>位速率分频设定</p> <p>位速率取决于此位和位速率寄存器 (SPBR) 的设定值的组合。根据位速率寄存器 (SPBR) 的设定值，决定基本位速率。此位的设定值用于选择基本位速率的无分频、2 分频、4 分频或者 8 分频。能给命令寄存器 0 ~ 3 分别设定不同的 BRDV[1:0]。因此，各命令能以不同位速率进行串行传送。</p> <p>00: 基本位速率 01: 基本位速率的 2 分频 10: 基本位速率的 4 分频 11: 基本位速率的 8 分频</p> <p>【注】 在发送时，必须以 QSPCLK 大于等于 2 分频的组合，设定位速率寄存器 (SPBR) 和此位的值。</p>
1	CPOL	0	R/W	<p>QSPCLK 极性设定</p> <p>设定 QSPCLK 的极性。必须在进行通信的模块之间设定相同的极性。</p> <p>0: 正极性 (空闲时 QSPCLK 为“0”) 1: 负极性 (空闲时 QSPCLK 为“1”)</p>

位	位名	初始值	R/W	说明
0	CPHA	1	R/W	<p>QSPCLK 相位设定</p> <p>设定进行传送数据的锁存 / 移位的 QSPCLK 边沿。必须在进行通信的模块之间设定相同的相位。</p> <p>0: 在奇数边沿进行数据锁存, 在偶数边沿进行数据移位。</p> <p>1: 在奇数边沿进行数据移位, 在偶数边沿进行数据锁存。</p> <p>【注】 将 QSPCLK 的第一个边沿作为第一个边沿。</p>

【参考】 根据串行闪存的数据表, QSPCLK 的设定有可能表现为 SPI 模式 0 ~ 3。如果为 SPI 模式 [1:0], 本书中的 CPOL 和 CPHA 就分别对应 CPOL=SPI 模式 [1] 和 CPHA=SPI 模式 [0]。

此模块的 CPOL/CPHA 的初始值为 CPOL=0、CPHA=1, SPI 模式 1 为初始值。

- 【注】** 1. 要将时钟延迟期间、QSSL 无效延迟期间、下次存取延迟期间中的某一个或者全部期间设定为“0”时, 必须将 SSLKP 置“1”, 设定为将 QSSL 有效的连续存取, 否则就不保证运行。在 QSSL 有效的连续存取中, 各延迟期间的设定方法请参照下述内容。
2. 在 QSSL 有效的连续存取的情况下, 在 QSPCLK 时钟停止后 QSSL 无效延迟期间、下次存取延迟期间和下一个命令的时钟延迟期间连在一起。此时, 如果要将设定允许 (SPCKDEN)、QSSL 无效延迟设定允许 (SLNDEN) 或者下次存取延迟设定允许 (SPNDEN) 置“0”, 就必须从后面的期间开始置“0”。

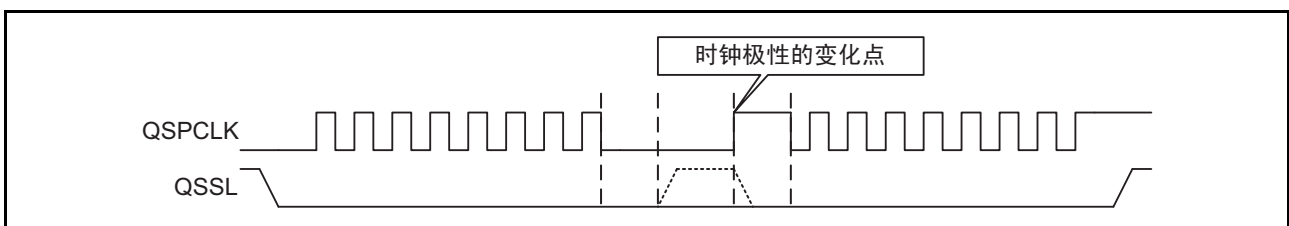


在上图中, (1) 为 QSSL 无效延迟期间, (2) 为下次存取延迟期间, (3) 为下一个命令的时钟延迟期间。如果要将其中的一个置“0”, 就必须从 (3) 开始置“0”。

即, 禁止像 {(1)、(2)、(3)}={0、0、1}、{0、1、1}、{0、1、0}……那样, 在“0”之后设定“1”。能设定的是 {(1)、(2)、(3)}={1、1、1}、{1、1、0}、{1、0、0}、{0、0、0}。

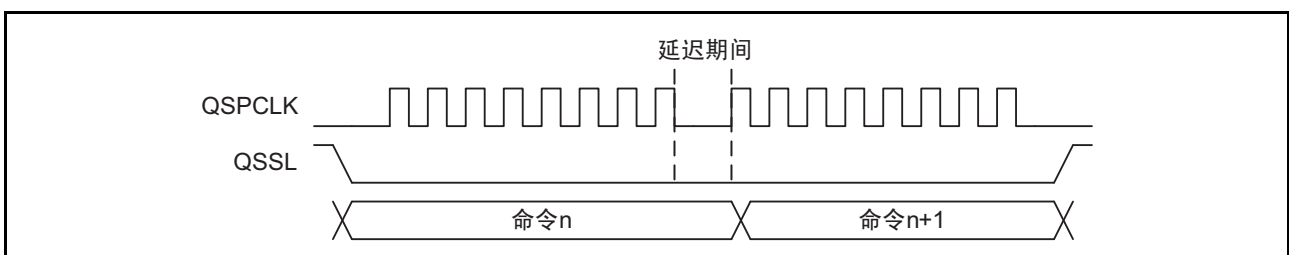
否则就不保证运行。

- 【注】** 1. 在保持 QSSL 的连续传送中按命令更改 BRDV[1:0] 和 CPOL 时, 必须在命令之间的延迟期间插入 QSSL 无效延迟期间、下次存取延迟期间和时钟延迟期间, 否则不保证运行。
2. 在保持 QSSL 的状态下更改 CPOL 时, 必须考虑到时钟极性的变化点有可能被检测为时钟边沿。



【注】 在保持 QSSL 的连续传送中按命令更改 SPIMOD[1:0] 和 CPHA 时, 必须在命令之间的延迟期间至少插入 1 个周期, 否则不保证运行 (也包括切换 Dual/Quad-SPI 的写 / 读的情况)。

在下图中, 在命令 n 为 Dual/Quad-SPI 的写操作时, 驱动命令 n 的期间中的数据线的。



18.3.14 缓冲控制寄存器 (SPBFCR)

缓冲控制寄存器 (SPBFCR) 设定发送缓冲器 (SPTXB) 和接收缓冲器 (SPRXB) 内的数据量复位和触发数据量。

位:	7	6	5	4	3	2	1	0
	TXRST	RXRST	TXTRG1	TXTRG0	—	RXTRG2	RXTRG1	RXTRG0
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	TXRST	0	R/W	发送缓冲器复位 将发送缓冲器的发送数据置为无效并且将缓冲器复位到空的状态。 0: 发送缓冲器正常运行 1: 发送缓冲器复位
6	RXRST	0	R/W	接收缓冲器复位 将接收缓冲器的接收数据置为无效并且将缓冲器复位到空的状态。 0: 接收缓冲器正常运行 1: 接收缓冲器复位
5 4	TXTRG1 TXTRG0	0 0	R/W R/W	发送缓冲器的数据量触发 设定发送缓冲器为空的时序, 作为状态寄存器的发送缓冲器空标志 (SPTEF) 基准。当发送缓冲器 (SPTXB) 保存的数据字节数小于等于设定的触发数时, 将 SPTEF 标志置 “1”。 00: 31 字节 (有 1 字节的空间) 01: 30 字节 (有 2 字节的空间) 10: 28 字节 (有 4 字节的空间) 11: 0 字节 (有 32 字节的空间)
3	—	0	R	保留位 读写值都为 “0”。
2 1 0	RXTRG2 RXTRG1 RXTRG0	0 0 0	R/W R/W R/W	接收缓冲器数据量触发 设定接收缓冲器为满的时序, 作为状态寄存器的接收缓冲器满标志 (SPRFF) 基准。当接收缓冲器 (SPRXB) 保存的数据字节数大于等于设定的触发数, 就将 SPRFF 标志置 “1”。 000: 1 字节 (有 31 字节空间) 001: 2 字节 (有 30 字节空间) 010: 4 字节 (有 28 字节空间) 011: 5 字节 (有 27 字节空间) 100: 8 字节 (有 24 字节空间) 101: 16 字节 (有 16 字节空间) 110: 24 字节 (有 8 字节空间) 111: 32 字节 (有 0 字节空间)

18.3.15 缓冲数据计数置位寄存器 (SPBDCR)

缓冲数据计数置位寄存器 (SPBDCR) 表示发送缓冲器 (SPTXB) 和接收缓冲器 (SPRXB) 保存的数据量。高 8 位表示发送缓冲器的发送数据字节数, 低 8 位表示接收缓冲器的接收数据字节数。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TXBC5	TXBC4	TXBC3	TXBC2	TXBC1	TXBC0	—	—	RXBC5	RXBC4	RXBC3	RXBC2	RXBC1	RXBC0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13 ~ 8	TXBC[5:0]	000000	R	发送字节数计数器 表示发送数据缓冲器 (SPTXB) 保存的数据字节数。 当为“B'000000”时, 表示 SPTXB 为空。 当为“B'100000”时, 表示 SPTXB 为满。
7、6	—	全 0	R	保留位 读写值都为“0”。
5 ~ 0	RXBC[5:0]	000000	R	接收字节数计数器 表示接收数据缓冲器 (SPRXB) 保存的数据字节数。 当为“B'000000”时, 表示 SPRXB 为空。 当为“B'100000”时, 表示 SPRXB 为满。

18.3.16 传送数据长度倍数设定寄存器 n (SPBMULn) (n=0、1、2、3)

传送数据长度倍数设定寄存器 0 ~ 3 (SPBMUL0 ~ 3) 设定命令寄存器 0 ~ 3 (SPCMD0 ~ 3) 的传送数据长度设定 (SPB[3:0]) 设定的数据长度传送的重复次数。此寄存器有 4 个, 分别对应命令寄存器 0 ~ 3。

在状态寄存器 (SPSR) 的通信结束标志 (TEND) 为通信未结束的状态下, 如果改写与正在参照的命令寄存器对应的此寄存器, 就不保证以后的运行。能通过顺序状态寄存器 (SPSSR) 确认正在参照的命令寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SPBMUL [31:24]								SPBMUL [23:16]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPBMUL [15:8]								SPBMUL [7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 0	SPBMUL [31:0]	H'00000001	R/W	传送数据长度倍数设定位 设定传送数据长度的倍数。 这些位设定命令寄存器 (SPCMD0 ~ 3) 的传送数据长度设定 (SPB[3:0]) 设定的传送数据长度的重复次数。 传送数据长度取决于 SPB[3:0]×SPBMUL[31:0]。 如果将此位置“H'00000000”, 重复次数就为 4294967296 次。

18.4 运行说明

在本节中，从开始驱动有效数据到取最后数据为止的期间为串行传送期间，QSSL 无效期间为空闲期间。

18.4.1 运行概要

此模块能在 Single/Dual/Quad-SPI 中进行串行传送。Single/Dual/Quad-SPI 的特点如表 18.4 所示。

表 18.4 各 SPI 模式的特点

	Single-SPI	Dual-SPI	Quad-SPI
数据线条数	1 条输入线、1 条输出线	2 条输入 / 输出线	4 条输入 / 输出线
数据线方向	单向	双向	双向
发送和接收的同时运行	○	×	×

运行概要如表 18.5 所示。

表 18.5 运行概要

项目	特点
QSPCLK 信号	输出
QMO 信号 (Single-SPI)	输出
QMI 信号 (Single-SPI)	输入
QIO1-0 (Dual-SPI) / QIO3-0 (Quad-SPI)	输入 / 输出
QSSL 信号	输出
QSSL 极性更改	○
传送率	~ P1φ
时钟源	内部波特率发生器
时钟极性	正 / 负
时钟相位	(上升沿锁存 / 下降沿输出) (下降沿锁存 / 上升沿输出)
传送位序	MSB first/LSB first
传送数据长度	(8/16/32)×(1 ~ 4294967296) 位
突发传送	○
QSPCLK 延迟控制	○
QSSL 无效延迟控制	○
下次存取延迟控制	○
传送启动方法	在 SPE 位为 “1” 时写发送缓冲器。 在 SPE 位为 “1” 时接收缓冲器有空间*。
顺序控制	○
发送缓冲器空检测	○
接收缓冲器满检测	○

【注】 * 在进行 Single-SPI 和 Dual/Quad-SPI 的写操作的情况下，在 SPE 位为 “1” 并且写发送缓冲器时启动传送。
在进行 Dual/Quad-SPI 的读操作的情况下，在 SPE 位为 “1” 并且接收缓冲器有传送数据长度的空间时启动传送。

18.4.2 引脚的控制

根据 Single-SPI/Dual/Quad-SPI 的写 / 读传送后的状态，此模块自动切换引脚状态。空闲时的数据引脚（QMO /QMI/QIO[3:0]）状态因引脚控制寄存器（SPPCR）的数据输出空闲值固定允许位（MOIFE） / 数据输出空闲时的固定值位（MOIFV）和 Single/Dual-SPI 时的 QIO3 输出固定值位（IO3FV） /Single/Dual-SPI 时的 QIO2 输出固定值位（IO2FV）的设定而不同。Single-SPI的引脚状态和Dual/Quad-SPI的引脚状态分别如表18.6和表18.7所示。

表 18.6 Single-SPI 的引脚状态

	Single-SPI
QSSL	输出
QSPCLK	输出
QMO	输出
QMI	输入
空闲时的 QMO	MOIFE=0: 最后的输出值 MOIFE=1: MOIFV 的设定值
空闲时的 QMI	—
QIO2	输出 IO2FV 的设定值或者不使用
QIO3	输出 IO3FV 的设定值或者不使用

表 18.7 Dual/Quad-SPI 的引脚状态

	Dual-SPI	Quad-SPI
QSSL	输出	输出
QSPCLK	输出	输出
QIO0	输入 / 输出	输入 / 输出
QIO1	输入 / 输出	输入 / 输出
QIO2	输出 IO2FV 的设定值或者不使用	输入 / 输出
QIO3	输出 IO3FV 的设定值或者不使用	输入 / 输出
空闲时的 QIO0	写后： MOIFE=0: 最后输出值 MOIFE=1: MOIFV 的设定值 读后: Hi-Z	写后： MOIFE=0: 最后输出值 MOIFE=1: MOIFV 的设定值 读后: Hi-Z
空闲时的 QIO1	写后： MOIFE=0: 最后的输出值 MOIFE=1: MOIFV 的设定值 读后: Hi-Z	写后： MOIFE=0: 最后的输出值 MOIFE=1: MOIFV 的设定值 读后: Hi-Z
空闲时的 QIO2	输出 IO2FV 的设定值或者不使用	写后： MOIFE=0: 最后的输出值 MOIFE=1: MOIFV 的设定值 读后: Hi-Z
空闲时的 QIO3	输出 IO3FV 的设定值或者不使用	写后： MOIFE=0: 最后的输出值 MOIFE=1: MOIFV 的设定值 读后: Hi-Z

18.4.3 传送格式

在 SPI 中，根据设定命令寄存器 0 ~ 3 (SPCMD0 ~ 3) 的 QSPCLK 极性设定 (CPOL) 和 QSPCLK 相位设定 (CPHA)，存在 4 个时钟的设定。以 8 位 MSB first 传送为例的各种设定的数据锁存 / 移位时序如图 18.1 所示。图中的“L”表示锁存时序，“S”表示移位时序。在 Single-SPI 模式中，DATA 对应 QMI/QMO；在 Dual-SPI 模式中，DATA 对应 QIO1-0；在 Quad-SPI 模式中，DATA 对应 QIO3-0。 t_{ckd} 表示 SPCMD0 ~ 3 的时钟延迟设定允许位 (SCKDEN) 为“1”时的时钟延迟期间。同样， t_{slnd} 表示 SPCMD0 ~ 3 的 QSSL 无效延迟设定允许位 (SLNDEN) 为“1”时的 QSSL 无效延迟期间， t_{spnd} 表示 SPCMD0 ~ 3 的下次存取延迟允许位 (SPNDEN) 为“1”时的下次存取延迟期间。

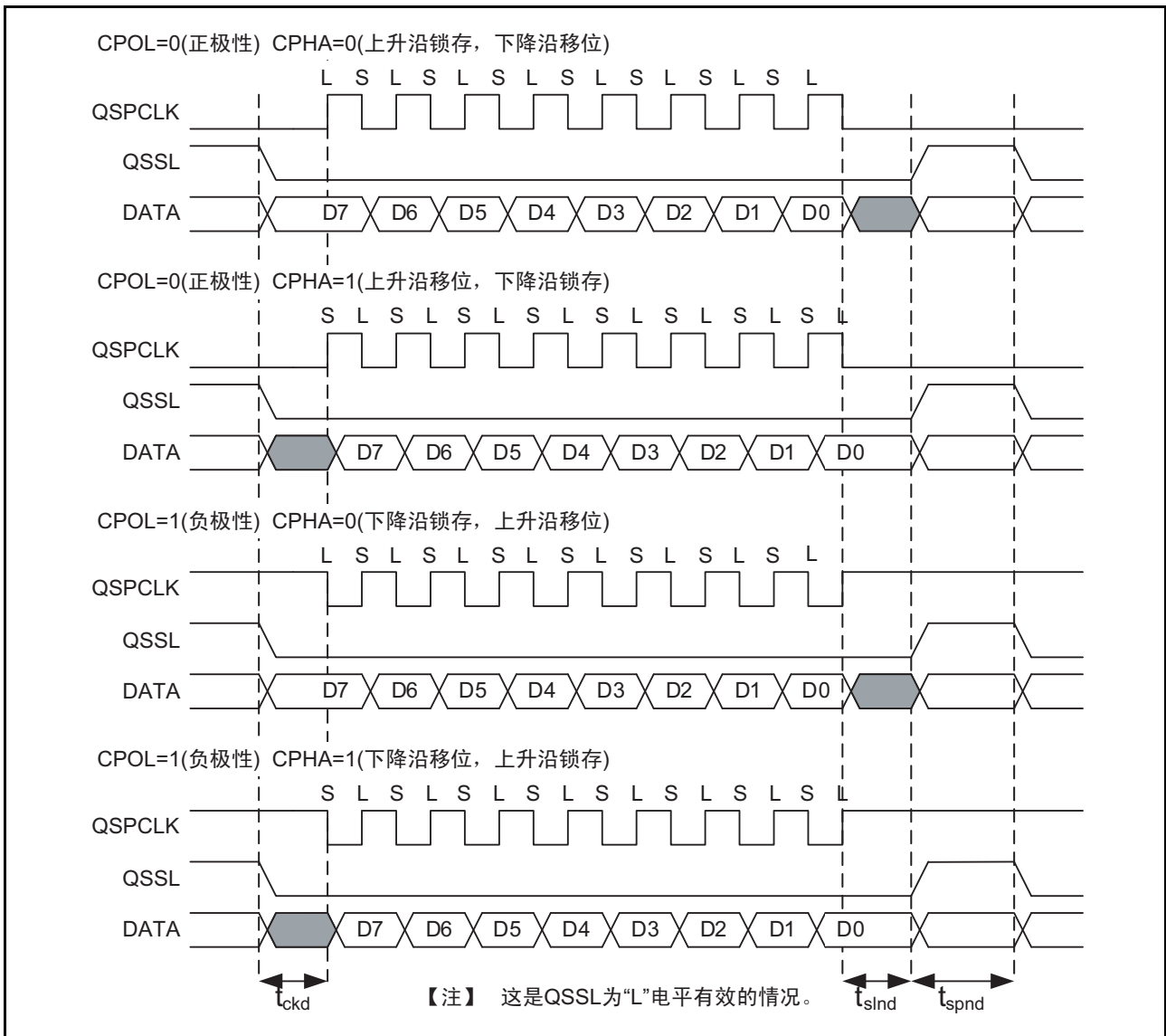


图 18.1 SPI 的时钟设定和传送时序

但是，在 1 分频的情况下，不支持发送运行和 CPHA 位为“0”时的接收运行。以下以 CPOL 位为“0”并且 CPHA 位为“0”的设定为例说明 Single/Dual/Quad-SPI 中的 8 位 MSB first 传送。

(1) Single-SPI 模式

Single-SPI 模式的传送格式如图 18.2 所示，在 Single-SPI 中同时进行发送和接收。因为发送和接收分别用 1 条数据线进行串行通信，所以通信速度为 1 位 / 1 个 QSPCLK。通过命令寄存器 0 ~ 3（SPCMD0 ~ 3）设定传送数据，传送数据的详细内容请参照“18.4.4 传送数据”。

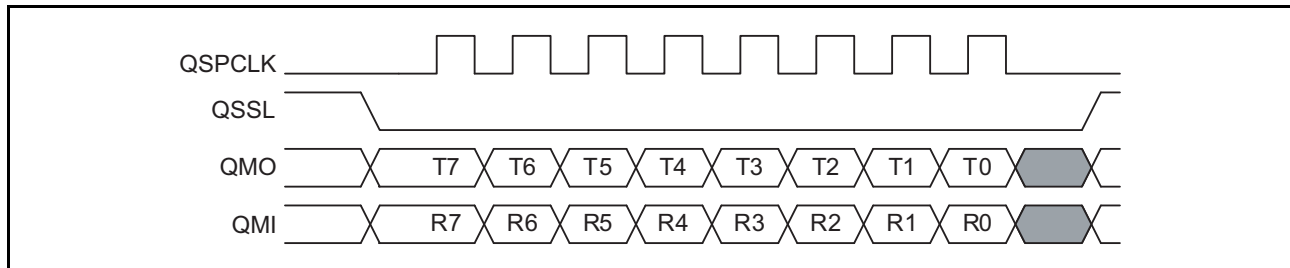


图 18.2 Single-SPI 模式的传送格式

(2) Dual-SPI 模式

Dual-SPI 模式的传送格式如图 18.3 所示，在 Dual-SPI 中只进行发送或者接收的单向通信。通过 SPCMD0 ~ 3 的 SPI 读写存取设定位（SPRW）设定发送或者接收。通过写操作进行数据发送，通过读操作进行数据接收。QIO1-0 通过各自的运行，切换输出和输入。因为发送和接收都用 2 条数据线进行串行通信，所以通信速度为 2 位 / 1 个 QSPCLK。从 QIO1 开始传送数据的起始位。通过 SPCMD0 ~ 3 设定传送数据，传送数据的详细内容请参照“18.4.4 传送数据”。

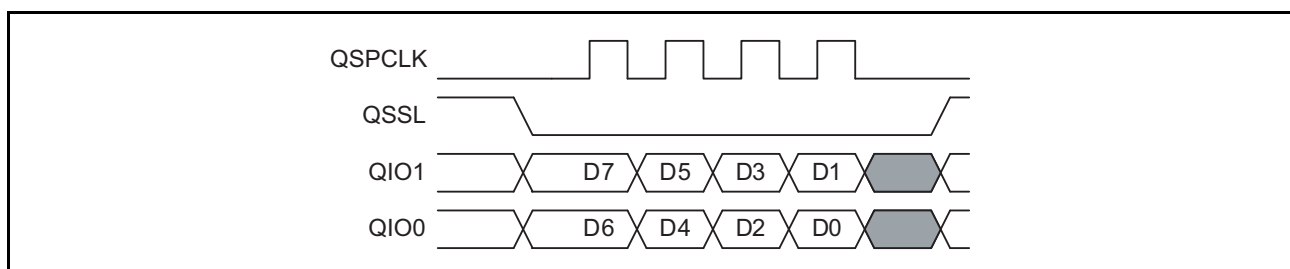


图 18.3 Dual-SPI 模式的传送格式

(3) Quad-SPI 模式

Quad-SPI 模式的传送格式如图 18.4 所示，在 Quad-SPI 中只能进行发送或者接收的单向通信。通过命令寄存器 0 ~ 3 (SPCMD0 ~ 3) 的 SPI 读写存取设定位 (SPRW) 设定发送或者接收。通过写操作进行数据发送，通过读操作进行数据接收。QIO3-0 通过各自的运行，切换输出和输入。因为发送和接收都用 4 条数据线进行串行通信，所以通信速度为 4 位 / 1 个 QSPCLK。从 QIO3 开始传送数据的起始位。通过 SPCMD0 ~ 3 设定传送数据，传送数据的详细内容请参照“18.4.4 传送数据”。

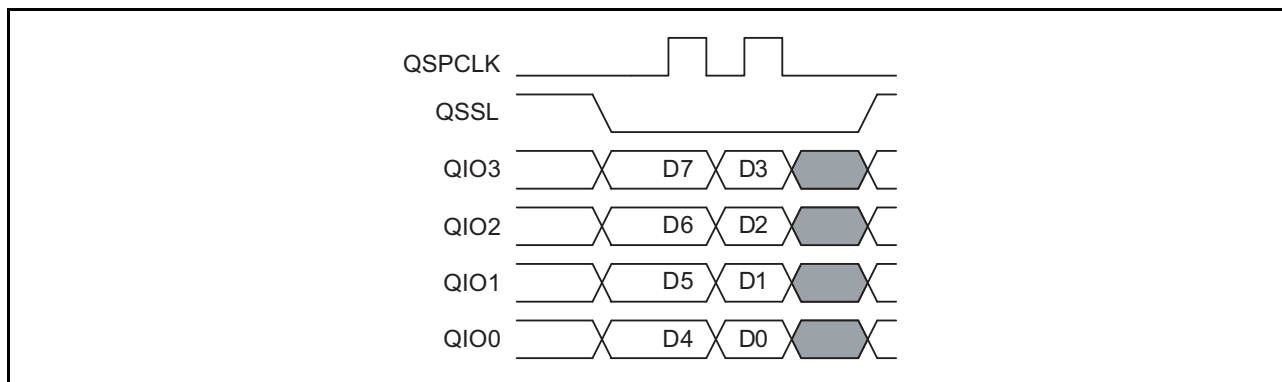


图 18.4 Quad-SPI 模式的传送格式

18.4.4 传送数据

数据格式取决于命令寄存器 0 ~ 3 (SPCMD0 ~ 3) 的传送数据长度设定位 (SPB[3:0])、LSB first 设定位 (LSBF) 和传送数据长度倍数设定寄存器 0 ~ 3 (SPBMUL0 ~ 3)。此模块与 MSB/LSB first 无关, 将从发送移位寄存器的 MSB 开始的传送数据长度的数据作为发送数据, 而将从接收移位寄存器的 LSB 开始的传送字节长度的数据作为接收数据。以下按照传送数据长度为 32 位、16 位、8 位的基本传送进行说明。

(1) MSB first 传送 (32 位数据)

以 MSB first 发送和接收 32 位数据时的发送缓冲器和发送移位寄存器、接收移位寄存器和接收缓冲器的运行如图 18.5 所示。

在发送时, CPU 或者直接存储器存取控制器将 32 位发送数据写到发送缓冲器 (SPTXB)。当发送移位寄存器为空时, 往 MSB 靠紧复制发送数据, 使发送移位寄存器变满。一旦发送开始, 就从发送移位寄存器的 MSB (bit31) 开始输出数据。如果经过 32 位串行传送所需的 QSPCLK 周期, 发送移位寄存器就变空。

在接收时, 从接收移位寄存器的 LSB (bit0) 开始写从数据引脚接收的数据。如果经过 32 位串行传送所需的 QSPCLK 周期, 接收移位寄存器就变满。如果接收缓冲器 (SPRXB) 有 32 位空间, 就从接收移位寄存器的 LSB 开始将 32 位数据复制到接收缓冲器, 使接收移位寄存器变空, 否则就不接收。要开始接收时, 必须读接收缓冲器, 以确保 32 位空间。在实际的传送中, 此运行的重复次数为传送数据长度倍数设定寄存器 0 ~ 3 (SPBMUL0 ~ 3) 的设定值。

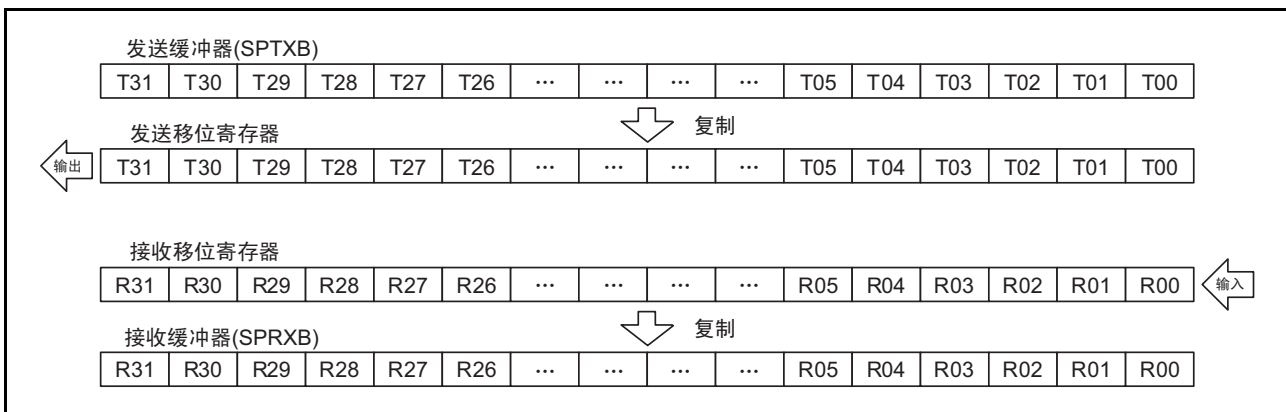


图 18.5 MSB first 传送 (32 位数据)

(2) MSB first 传送 (16 位数据)

以 MSB first 发送和接收 16 位数据时的发送缓冲器和发送移位寄存器、接收移位寄存器和接收缓冲器的运行如图 18.6 所示。

在发送时, CPU 或者直接存储器存取控制器将 16 位发送数据写到发送缓冲器 (SPTXB)。当发送移位寄存器为空时, 往 MSB 靠紧复制发送数据, 使发送移位寄存器变满。一旦发送开始, 就从发送移位寄存器的 MSB (bit31) 开始输出数据。如果经过 16 位串行传送所需的 QSPCLK 周期, 发送移位寄存器就变空。

在接收时, 从接收移位寄存器的 LSB (bit0) 开始写从数据引脚接收的数据。如果经过 16 位串行传送所需的 QSPCLK 周期, 接收移位寄存器就变满。如果接收缓冲器 (SPRXB) 有 16 位空间, 就从接收移位寄存器的 LSB 开始将 16 位数据复制到接收缓冲器, 使接收移位寄存器变空, 否则就不接收。要开始接收时, 必须读接收缓冲器, 以确保 16 位空间。在实际的传送中, 此运行的重复次数为传送数据长度倍数设定寄存器 0 ~ 3 (SPBMUL0 ~ 3) 的设定值。

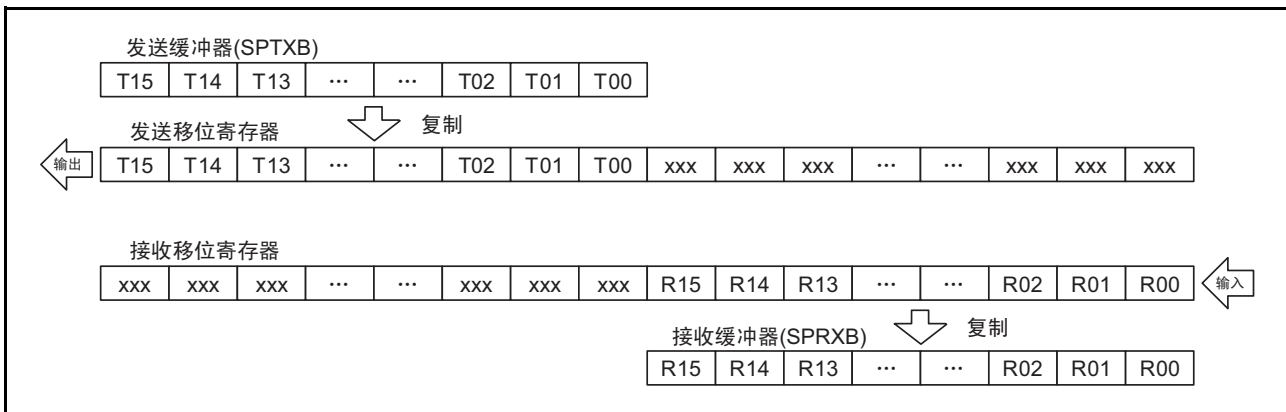


图 18.6 MSB first 传送（16 位数据）

(3) MSB first 传送（8 位数据）

以 MSB first 发送和接收 8 位数据时的发送缓冲器和发送移位寄存器、接收移位寄存器和接收缓冲器的运行如图 18.7 所示。

在发送时，CPU 或者直接存储器存取控制器将 8 位发送数据写到发送缓冲器（SPTXB）。当发送移位寄存器为空时，往 MSB 靠紧复制发送数据，使发送移位寄存器满。一旦发送开始，就从发送移位寄存器的 MSB（bit31）开始输出数据。如果经过 8 位串行传送所需的 QSPCLK 周期，发送移位寄存器就变空。

在接收时，从接收移位寄存器的 LSB（bit0）开始写从数据引脚接收的数据。如果经过 8 位串行传送所需的 QSPCLK 周期，接收移位寄存器就变满。如果接收缓冲器（SPRXB）有 8 位空间，就从接收移位寄存器的 LSB 开始将 8 位数据复制到接收缓冲器，使接收移位寄存器变空，否则就不接收。要开始接收时，必须读取接收缓冲器，以确保 8 位空间。

在实际的传送中，此运行的重复次数为传送数据长度倍数设定寄存器 0～3（SPBMUL0～3）的设定值。

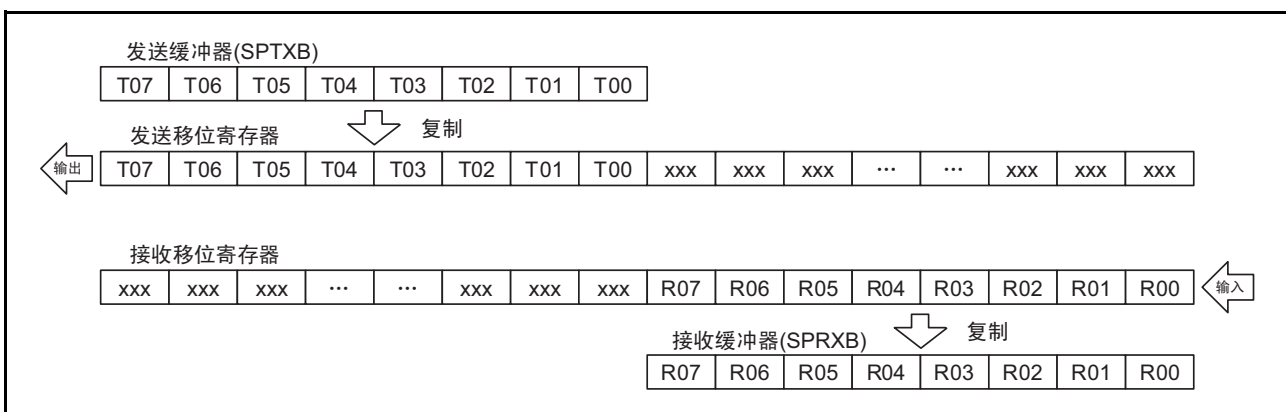


图 18.7 MSB first 传送（8 位数据）

(4) LSB first 传送 (32 位数据)

以 LSB first 发送和接收 32 位数据时的发送缓冲器和发送移位寄存器、接收移位寄存器和接收缓冲器的运行如图 18.8 所示。

在发送时，CPU 或者直接存储器存取控制器将 32 位发送数据写到发送缓冲器 (SPTXB)。当发送移位寄存器为空时，往 MSB 靠紧复制位序颠倒的 32 位发送数据，使发送移位寄存器满。一旦发送开始，就从发送移位寄存器的 MSB (bit31) 开始输出数据。如果经过 32 位串行传送所需的 QSPCLK 周期，发送移位寄存器就变空。

在接收时，从接收移位寄存器的 LSB (bit0) 开始写从数据引脚接收的数据。如果经过 32 位串行传送所需的 QSPCLK 周期，接收移位寄存器变满。如果接收缓冲器 (SPRXB) 有 32 位空间，就从接收移位寄存器的 LSB 开始将位序颠倒的 32 位数据复制到接收缓冲器，使接收移位寄存器变空，否则就不接收。要开始接收时，必须读接收缓冲器，以确保 32 位空间。

在实际的传送中，此运行的重复次数为传送数据长度倍数设定寄存器 0 ~ 3 (SPBMUL0 ~ 3) 的设定值。

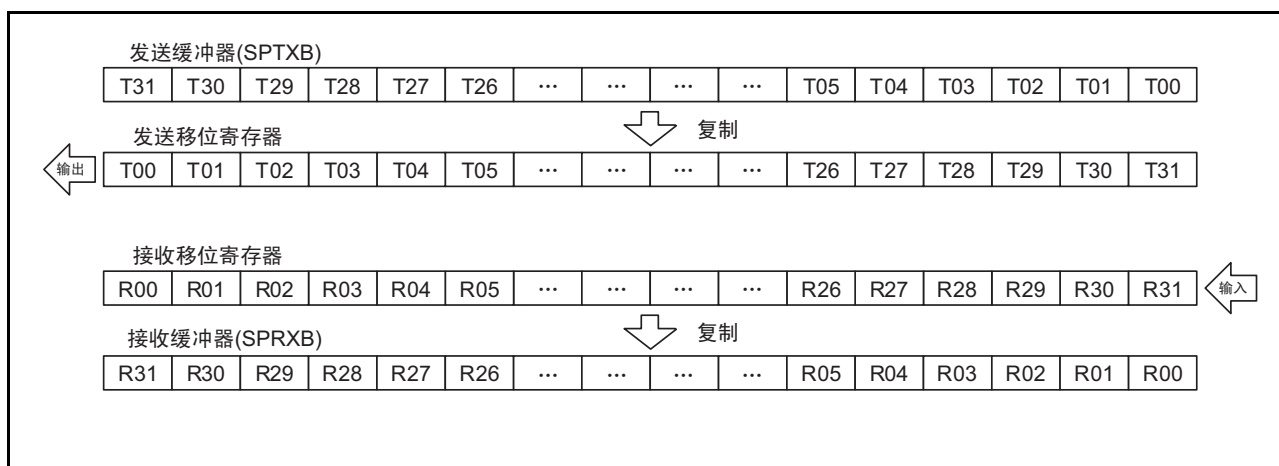


图 18.8 LSB first 传送 (32 位数据)

(5) LSB first 传送 (16 位数据)

以 LSB first 发送和接收 16 位数据时的发送缓冲器和发送移位寄存器、接收移位寄存器和接收缓冲器的运行如图 18.9 所示。

在发送时，CPU 或者直接存储器存取控制器将 16 位发送数据写到发送缓冲器 (SPTXB)。当发送移位寄存器为空时，往 MSB 靠紧复制位序颠倒的 16 位发送数据，使发送移位寄存器满。一旦发送开始，就从发送移位寄存器的 MSB (bit31) 开始输出数据。如果经过 16 位串行传送所需要的 QSPCLK 周期，发送移位寄存器就变空。

在接收时，从接收移位寄存器的 LSB (bit0) 开始写从数据引脚接收的数据。如果经过 16 位串行传送所需的 QSPCLK 周期，接收移位寄存器就变满。如果接收缓冲器 (SPRXB) 有 16 位空间，就从接收移位寄存器的 LSB 开始将位序颠倒的 16 位数据复制到接收缓冲器，使接收移位寄存器变空，否则就不接收。要开始接收时，必须读接收缓冲器，以确保 16 位空间。

在实际的传送中，此运行的重复次数为传送数据长度倍数设定寄存器 0 ~ 3 (SPBMUL0 ~ 3) 的设定值。

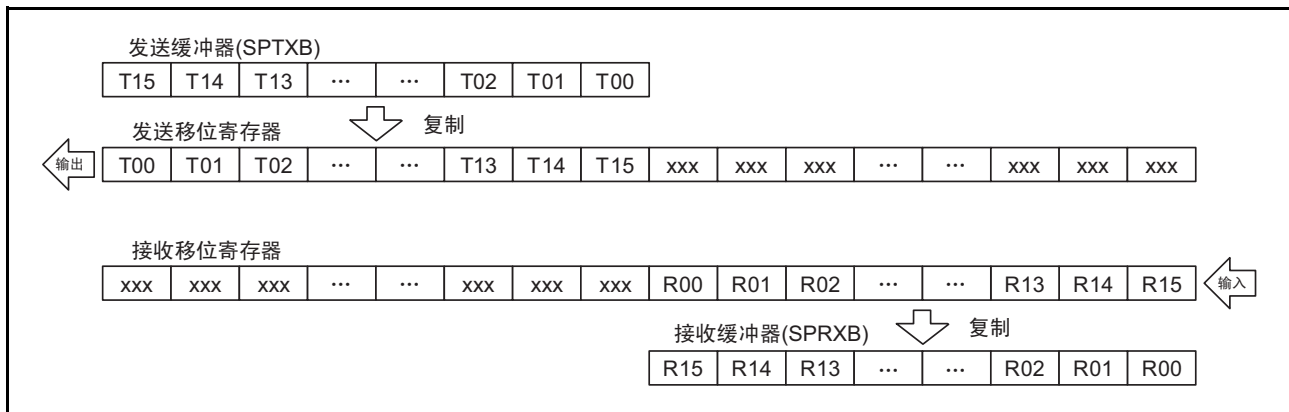


图 18.9 LSB first 传送（16 位数据）

(6) LSB first 传送（8 位数据）

以 LSB first 发送和接收 8 位数据时的发送缓冲器和发送移位寄存器、接收移位寄存器和接收缓冲器的运行如图 18.10 所示。

在发送时，CPU 或者直接存储器存取控制器将 8 位发送数据写到发送缓冲器（SPTXB）。当发送移位寄存器为空时，往 MSB 靠紧复制位序颠倒的 8 位发送数据，使发送移位寄存器变满。一旦发送开始，就从发送移位寄存器的 MSB (bit31) 开始输出数据。如果经过 8 位串行传送所需的 QSPCLK 周期，发送移位寄存器就变空。

在接收时，从接收移位寄存器的 LSB (bit0) 开始写从数据引脚接收的数据。如果经过 8 位串行传送所需的 QSPCLK 周期，接收移位寄存器就变满。如果接收缓冲器（SPRXB）有 8 位空间，就从接收移位寄存器的 LSB 开始将位序颠倒的 8 位数据复制到接收缓冲器，使接收移位寄存器变空，否则就不接收。要开始接收时，必须读接收缓冲器，以确保 8 位空间。

在实际的传送中，此运行的重复次数为传送数据长度倍数设定寄存器 0 ~ 3（SPBMUL0 ~ 3）的设定值。

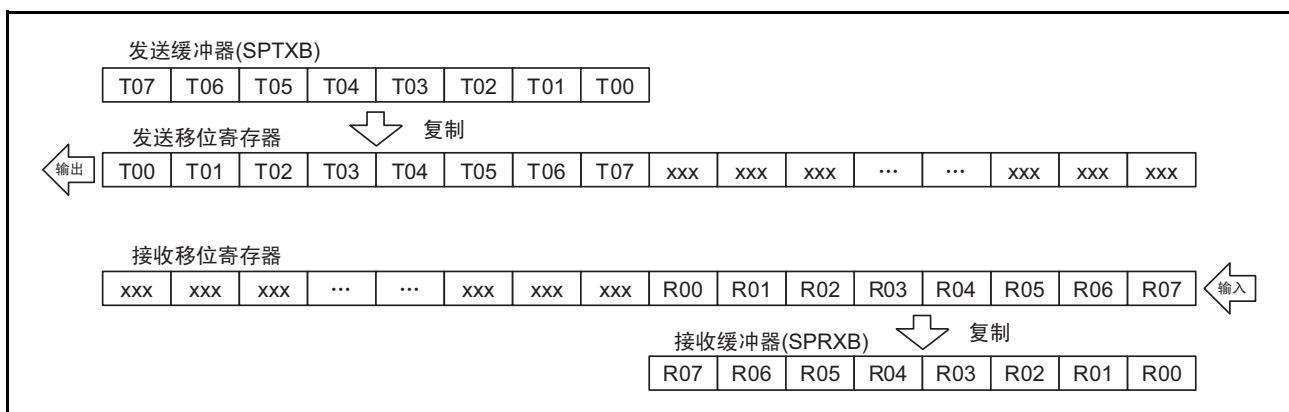


图 18.10 LSB first 传送（8 位数据）

18.4.5 异常运行

在正常的串行传送中，能将数据寄存器（SPDR）写到发送缓冲器的数据进行串行发送，并且能通过读 SPDR 从接收缓冲器读串行接收的数据。但是，根据存取 SPDR 时的发送缓冲器和接收缓冲器的状态，有可能进行异常传送。

异常传送如表 18.8 所示。

表 18.8 异常传送的产生条件

	产生条件	运行
A	在发送缓冲器满的状态下写 SPDR。	丢失写数据。
B	在接收缓冲器空的状态下读 SPDR。	输出不定值。

在表 18.8 的 A 所示的运行中，必须通过缓冲数据计数置位寄存器（SPBDCR）的发送字节数计数器位（TXBC[5:0]）确认是否能写 SPDR。

在 B 所示的运行中，必须通过 SPBDCR 的接收字节数计数器位（RXBC[5:0]）确认是否已将有效数据保存到接收缓冲器。

18.4.6 初始化

如果给控制寄存器（SPCR）的 SPI 功能允许位（SPE）写“0”，此模块就使功能无效并且对一部分功能进行初始化。如果发生上电复位，此模块就对全部模块功能进行初始化。

当将 SPCR 的 SPE 位清“0”时，此模块进行以下的初始化：

- 中止正在进行的串行传送。
- 对发送移位寄存器和接收移位寄存器进行初始化。
- 对内部状态机进行初始化。
- 对顺序进行初始化。
- 对 SPSR 的通信结束位（TEND）进行初始化。

在通过将 SPE 位清“0”进行初始化时，不对此模块的控制位和发送 / 接收缓冲器进行初始化。因此，能通过将 SPE 位置“1”，在和 SPE 位被清“0”前相同的状态下重新开始传送。但是，当将 SPE 位清“0”时，由于对发送移位寄存器和接收移位寄存器进行初始化，因此放弃传送中途的传送数据。

18.4.7 SPI 运行

此模块的运行模式如下：

- Single-SPI 模式
- Dual-SPI 模式/Quad-SPI 模式

以下说明各模式的运行。

(1) Single-SPI 模式

(a) 串行传送的开始

串行传送的开始条件是：发送缓冲器有传送数据长度的数据并且接收缓冲器有传送数据长度的空间。

(b) 串行传送的结束

串行传送的结束条件是：与时钟设定无关，发送最后采样时序对应的 QSPCLK 边沿。一旦串行传送正常结束，就将接收数据从接收移位寄存器复制到接收缓冲器。在串行传送结束并且将接收数据从接收移位寄存器复制到接收缓冲器后，如果接收缓冲器没有下次要传送的数据长度的空间，就不开始下一次传送。

(c) 顺序控制

在 Single-SPI 模式中，能通过设定顺序控制寄存器（SPSCR），用命令寄存器 0～3（SPCMD0～3）和部分或者全部传送数据长度倍数设定寄存器 0～3（SPBMUL0～3）构成顺序。此模块有命令寄存器对应的指针，并且能通过读顺序状态寄存器（SPSSR），确认此指针的值。

如果通过将控制寄存器（SPCR）的 SPI 功能允许位（SPE）置“1”使此模块功能有效，此模块就将命令寄存器对应的指针设定到 SPCMD0，并且在开始串行传送时将 SPCMD0 和 SPBMUL0 的设定内容反映到传送格式。每当与正在参照的 SPCMD0～3 对应的传送的下次存取延迟期间结束时，此模块将命令寄存器的指针递增。当构成顺序的最后命令对应的传送结束时，此模块将指针恢复到 SPCMD0，重复执行顺序。

能给 SPCMD0～3 分别设定基本传送数据长度、MSB/LSB first、时钟、一部分位速率、SPI 传送模式、传送方向（只限于 Dual/Quad-SPI 模式）、QSSL 保持、时钟延迟期间、QSSL 无效延迟期间和下次存取延迟期间。在此设定的基本传送数据长度乘上 SPBMUL0～3 设定值后的值为总传送数据长度。

将顺序控制寄存器设定为“H'02”并且使用 SPCMD0～2 构成顺序时的运行例子如图 18.11 所示。在图中，QMO/QMI 的灰色部分表示无效数据。图中 (1)～(3) 表示如下：

1. 时钟延迟期间（SPCKD）设定值为“B'000”（1.5个QSPCLK）。
2. QSSL无效延迟期间（SSLND）设定值为“B'000”（1个QSPCLK）。
3. 下次存取延迟期间（SPND）设定值为“B'000”（1个QSPCLK）。

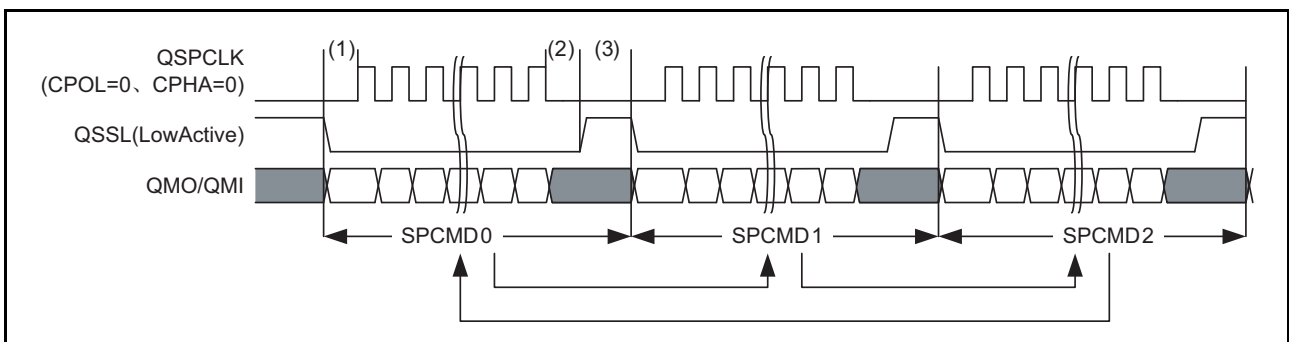


图 18.11 顺序控制运行例子

(d) 突发传送

在 Single-SPI 模式中，能通过 2 种方法进行突发传送。

一种是使用命令寄存器 0 ~ 3 (SPCMD0 ~ 3) 的传送数据长度设定位 (SPB[3:0]) 和传送数据长度倍数设定寄存器 0 ~ 3 (SPBMUL0 ~ 3) 进行突发传送。通过将 SPB[3:0] 位设定为 8 位、16 位或者 32 位并且将 SPBMUL0 ~ 3 设定为 1 ~ 4294967296，以 SPB[3:0] 位设定长度的传送为 1 次传送，只重复执行 SPBMUL0 ~ 3 设定次数的连续传送。如果在传送过程中发送缓冲器 (SPTXB) 没有发送数据或者接收缓冲器 (SPRXB) 没有 SPB[3:0] 位设定的数据长度的空间，就停止时钟，等待重新开始传送。将 SPB[3:0] 位设定为 32 位并且将 SPBMUL 设定为 4 以及将总传送数据长度设定为 128 位时的突发传送例子如图 18.12 所示。图中 (1) ~ (4) 的内容说明如下：

1. 这是第 1 次的 32 位传送。
2. 这是第 2 次的 32 位传送。
3. 因为发送缓冲器没有发送数据或者接收缓冲器没有 32 位空间，所以停止时钟。在此期间，QMO 输出保持之前的值。一旦写发送数据或者接收缓冲器有空间，就在发送时钟后重新开始传送。
4. 这是第 3 次和第 4 次的 32 位传送。

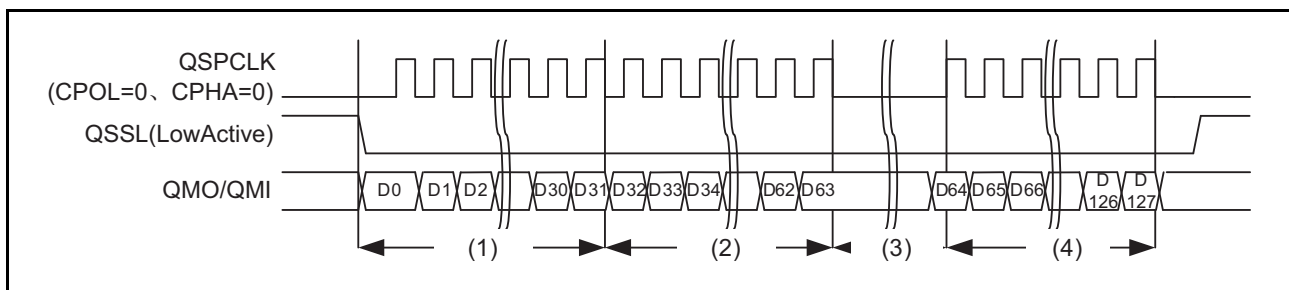


图 18.12 将传送数据长度设定为 128 位时的突发传送例子 (Single-SPI 模式)

另一种方法是通过在串行传送结束后到下一次串行传送的期间使 QSSL 持续有效进行突发传送。如果将命令寄存器 0 ~ 3 (SPCMD0 ~ 3) 的 QSSL 信号电平保持位 (SSLKP) 置“1”，QSSL 信号就在该命令寄存器对应的传送结束后到下一次传送的期间持续有效。使用 QSSL 信号电平保持功能的突发传送例子如图 18.13 所示。图中 (1) ~ (6) 运行内容的说明如下：

1. 这是 SPCMD0 设定的时钟延迟期间。在突发传送的最初传送中，必须至少设定为 1.5 个 QSPCLK。
2. 这是 SPCMD0 设定的 QSSL 无效延迟期间。因为 SSLKP 为“1”，所以在 QSSL 无效延迟期间后也不将 QSSL 置为无效。通过设定 SPCMD0 的 QSSL 无效延迟设定允许位 (SLNDEN) 来改变期间长度。如果将 SLNDEN 置“1”，期间长度就取决于从属选择无效延迟寄存器 (SSLND) 的设定值；如果将 SLNDEN 置“0”，期间长度就为 0 个 QSPCLK。
3. 这是 SPCMD0 设定的下次存取延迟期间。因为 SSLKP 为“1”，所以在此期间也不将 QSSL 置为无效。通过设定 SPCMD0 的下次存取延迟设定允许位 (SPNDEN) 来改变期间长度。如果将 SPNDEN 置“1”，期间长度就取决于下次存取延迟寄存器 (SPND) 的设定值；如果将 SPNDEN 置“0”，期间长度就为 0 个 QSPCLK。
4. 这是 SPCMD1 设定的时钟延迟期间。通过设定 SPCMD1 的时钟延迟设定允许位 (SCKDEN) 来改变期间长度。如果将 SCKDEN 置“1”，期间长度就取决于时钟延迟设定寄存器 (SPCKD) 的设定值；如果将 SCKDEN 置“0”，期间长度就为 0 个 QSPCLK。
5. 这是 SPCMD1 设定的 QSSL 无效延迟期间。在突发传送的最后传送中，必须至少设定为 1 个 QSPCLK。因为 SPCMD1 的 SSLKP 为“0”，所以在 QSSL 无效延迟期间后将 QSSL 置为无效。
6. 这是 SPCMD1 设定的下次存取延迟期间。在突发传送的最后传送中，必须至少设定为 1 个 QSPCLK，而且必须通过将 SSLKP 置“0”，使 QSSL 无效。

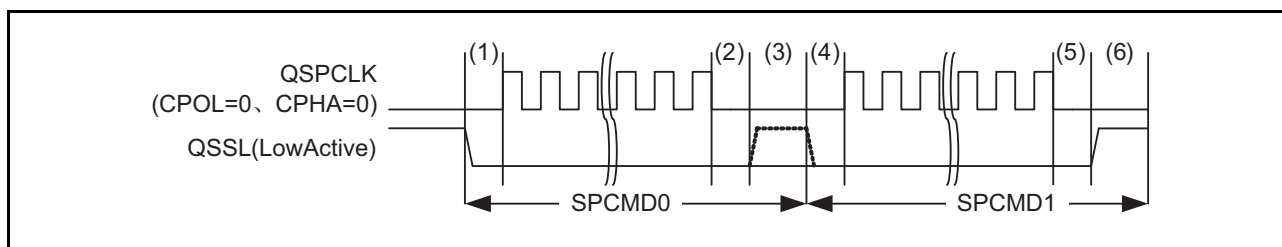


图 18.13 使用 QSSL 信号电平保持功能的突发传送例子 (Single-SPI 模式)

在通过此方法设定突发传送时，必须注意以下内容：

在通过更新命令来更改时钟分频比或者时钟极性时，必须插入 (2) ~ (4) 的期间。

如果更改时钟分频比，(4) 的期间有可能在设定值的前或者后。

在通过更新命令来更改时钟相位或者 Single/Dual/Quad-SPI 的传送模式时，必须至少插入 (2) 的期间 (Dual/Quad-SPI 也包括读写的更改)。

(e) 初始设定流程

Single-SPI 模式中的初始设定流程例子如图 18.14 所示。有关中断控制器和直接存储器存取控制器的设定，请参照各章节。

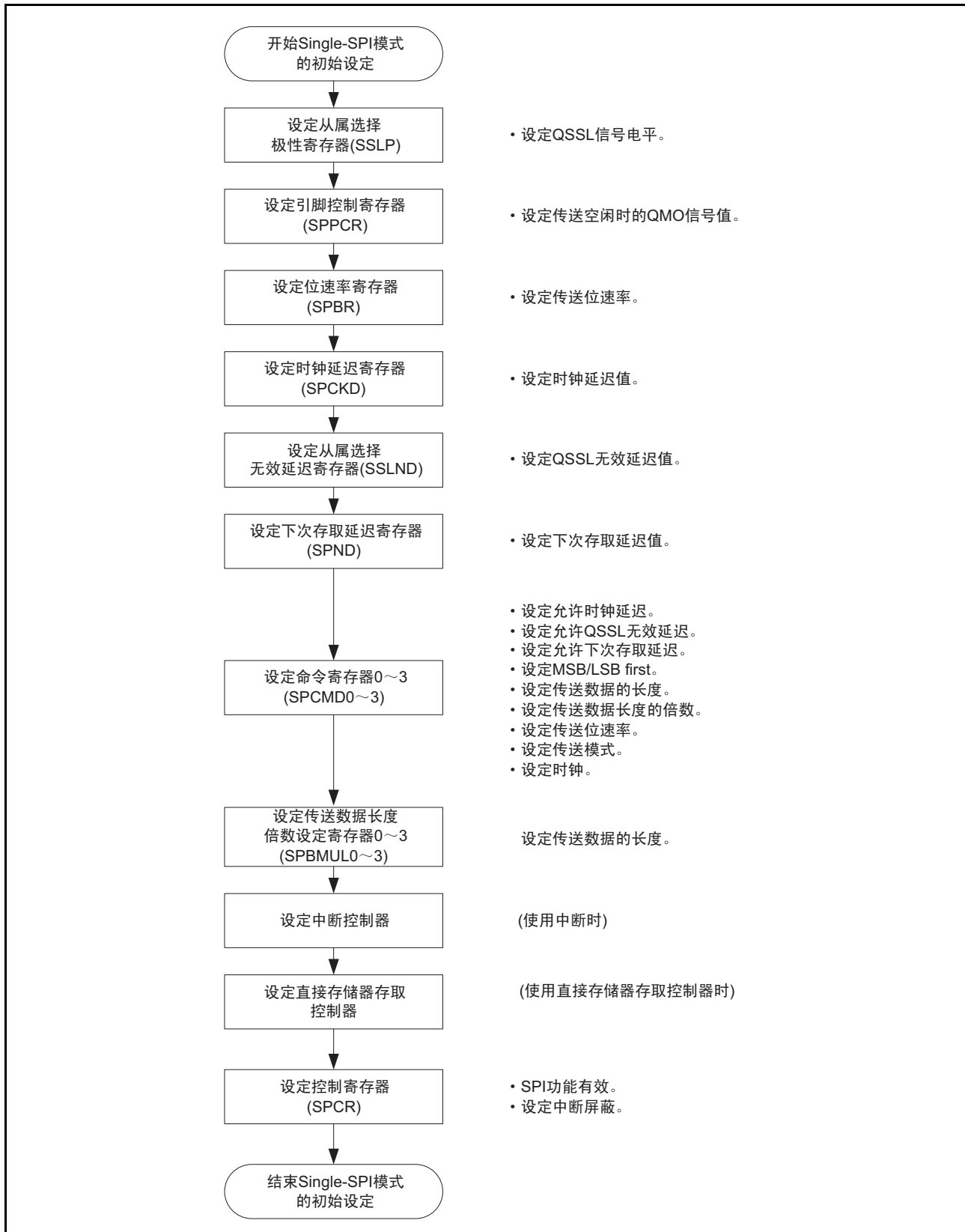


图 18.14 Single-SPI 模式中的初始设定流程例子

(f) 传送流程

Single-SPI 模式时的运行流程如图 18.15 所示。通过设定传送数据长度进行的突发传也以此流程为基准。

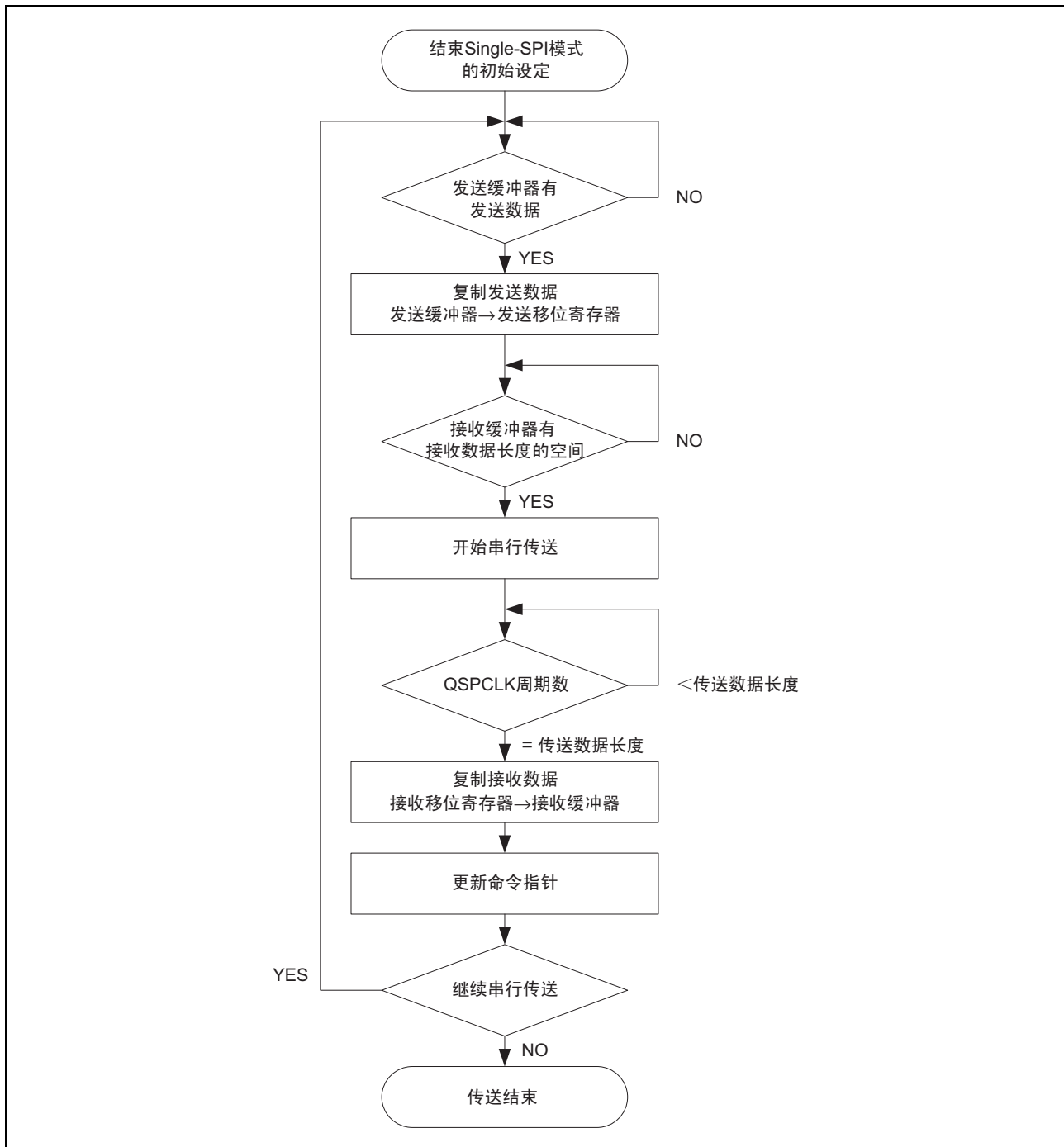


图 18.15 Single-SPI 模式的运行流程

(2) Dual-SPI 模式 /Quad-SPI 模式

(a) 串行传送的开始

在发送和接收数据时，Dual-SPI 模式和 Quad-SPI 模式的串行传送开始条件不同。

发送数据时的开始条件是发送缓冲器有传送数据长度的数据。

接收数据时的开始条件是接收缓冲器有传送数据长度的空间。

(b) 串行传送的结束

与发送和接收无关，如果发送最后采样时序所对应的 QSPCLK 边沿，就结束串行传送。在写后和读后，Dual /Quad-SPI 模式中的空闲时的 QIO 引脚控制不同。在写后，通过设定寄存器，输出最后输出数据或者固定值；在读后，将 QIO 引脚置为 Hi-Z。以 Quad-SPI 模式为例的存取结束后的引脚状态如图 18.16 所示。图中 (1) 和 (2) 的说明如下：

1. 在写时 QIO0~3 为输出，如果在写操作结束后将 QSSL 置为无效，就通过设定引脚控制寄存器（SPPCR）的数据输出空闲值固定允许位（MOIFE）来改变输出值。当 MOIFE 为“1”时，输出由数据输出空闲时固定值位（MOIFV）设定的值；当 MOIFE 为“0”时，输出最后输出数据的值。
2. 在读时 QIO0~3 为输入，如果在读操作结束后将 QSSL 置为无效，就为 Hi-Z，与 MOIFE 和 MOIFV 的值无关。

有关 Dual/Quad-SPI 模式的引脚控制，请参照“18.4.2 引脚的控制”。

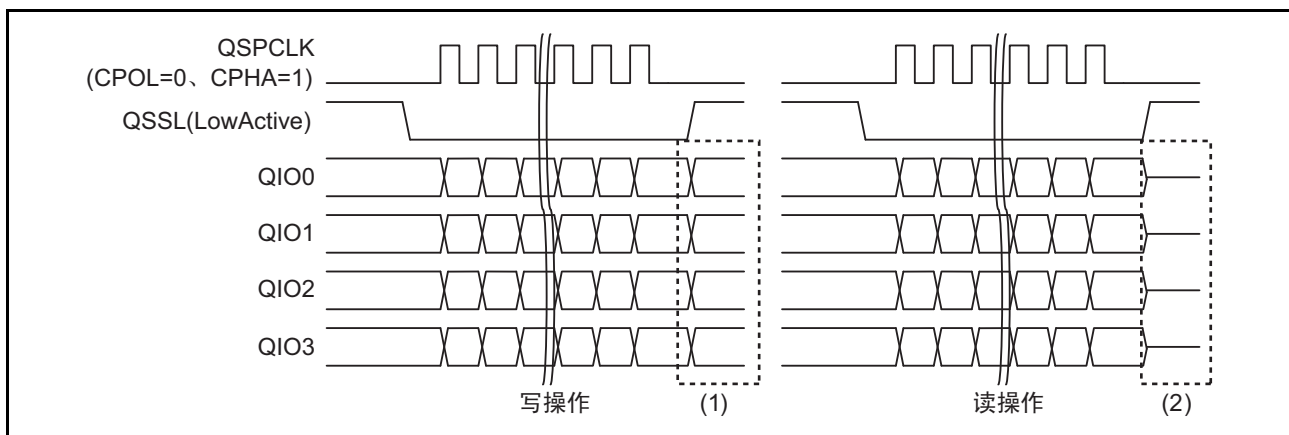


图 18.16 Dual/Quad-SPI 模式的传送结束后的引脚状态（Quad-SPI 的例子）

(c) 顺序控制

Dual/Quad-SPI 模式和 Single-SPI 模式一样，能使用顺序控制寄存器（SPSCR）、命令寄存器 0~3（SPCMD0~3）和传送数据长度倍数设定寄存器 0~3（SPBMUL0~3）构成顺序。运行的详细内容请参照“18.4.7 SPI 运行(1) Single-SPI 模式(c) 顺序控制”。

因为在 Dual/Quad-SPI 模式中只能进行串行传送或者串行接收的单向通信，所以通过 SPCMD0~3 的 SPI 读写存取设定位（SPRW）设定发送或者接收。通过 SPCMD0~3 的 SPI 运行模式设定位（SPIMOD[1:0]）设定包括 Dual/Quad-SPI 模式和 Single-SPI 模式在内的 3 种运行模式。通过使用这些设定，能在顺序控制中切换 Single-SPI 模式、Dual-SPI 模式的发送和接收以及 Quad-SPI 模式的发送和接收。切换传送模式的构成顺序的例子如图 18.17 所示。

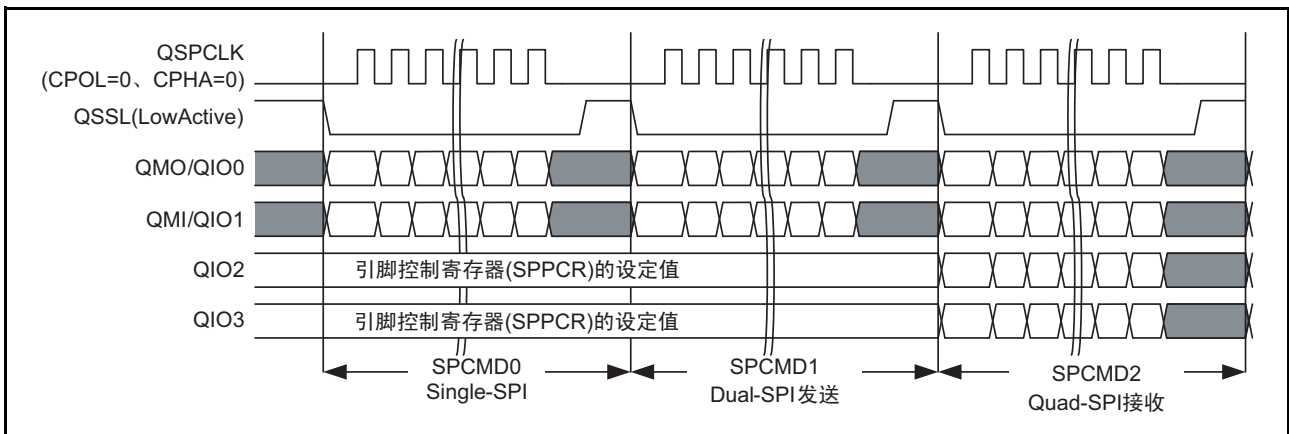


图 18.17 切换传送模式后的顺序结构例子

【注】 在 Dual/Quad-SPI 模式中构成顺序时，必须注意以下内容：

当构成顺序的命令全部为 Dual/Quad-SPI 读操作时，只在接收缓冲器有接收数据长度的空间时执行顺序运行。

要结束读操作时，必须在接收所需数据长度的数据后将控制寄存器（SPCR）的 SPI 功能允许位（SPE）清“0”或者将最后的顺序作设定为写操作，使发送缓冲器变空。

(d) 突发传送

在 Dual/Quad-SPI 模式中，能通过 2 种方法进行突发传送。

一种是使用命令寄存器 0 ~ 3（SPCMD0 ~ 3）的传送数据长度设定位（SPB[3:0]）和传送数据长度倍数设定寄存器 0 ~ 3（SPBMUL0 ~ 3）进行突发传送。和 Single-SPI 模式一样，通过将 SPB[3:0] 位设定为 8 位、16 位或者 32 位并且将 SPBMUL0 ~ 3 设定为 1 ~ 4294967296，只重复执行 SPBMUL0 ~ 3 设定次数的 SPB[3:0] 位设定长度的连续传送。但是，如果在写过程中发送缓冲器（SPTXB）没有发送数据或者在读过程中接收缓冲器（SPRXB）没有 SPB[3:0] 位设定数据长度的空间，就停止时钟，等待重新开始传送。在 Dual/Quad-SPI 模式中要传送大量的数据时，此方法有效。将 SPB[3:0] 位设定为 32 位并且将 SPBMUL 设定为 4 以及将总传送数据长度设定为 128 位时的 Quad-SPI 的突发传送例子如图 18.18 所示。图中 (1) ~ (4) 的内容说明如下：

1. 这是第 1 次的 32 位传送。
2. 这是第 2 次的 32 位传送。
3. 因为发送缓冲器没有发送数据或者接收缓冲器没有 32 位空间，所以停止时钟。在此期间，当 QIO3-0 为输出时，保持之前的值；当 QIO3-0 为输入时，取决于通信对象设备的输出。如果通过写发送数据或者使接收缓冲器有空间来调整传送的重新开始条件，就通过内部时钟输出发送数据。
4. 这是第 3 次和第 4 次的 32 位传送。

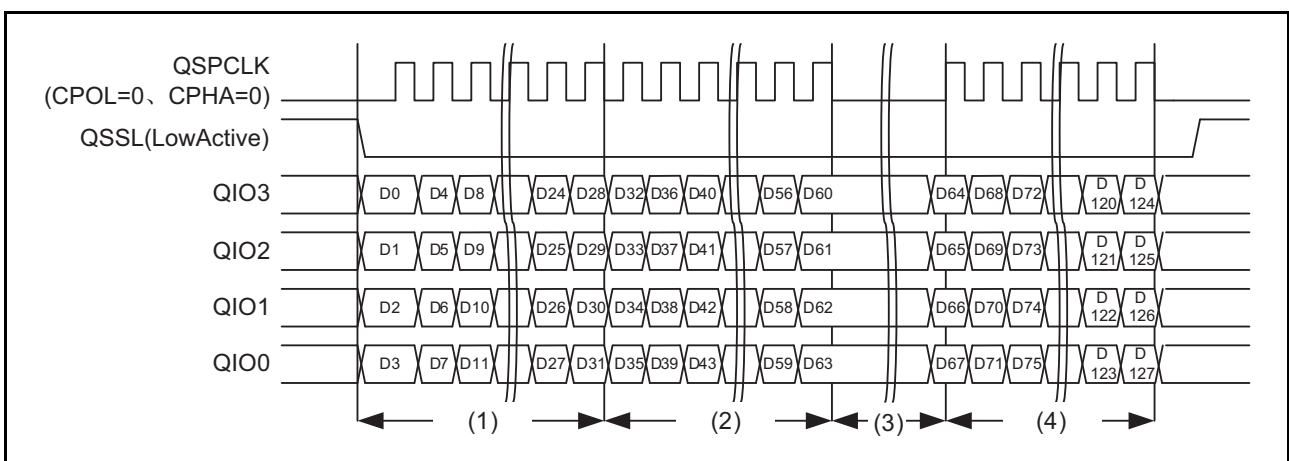


图 18.18 将传送数据长度设定为 128 位时的突发传送例子（Quad-SPI 模式）

另一种方法和 Single-SPI 模式一样，使用 QSSL 信号电平保持功能。因为能在传送中途更改 SPI 传送模式（Single/Dual/Quad-SPI），所以在 Single-SPI 模式中将命令数据写到串行闪存或者在 Quad-SPI 模式中写存储器的保存数据等情况下，此方法有效。但是，如果更改 SPI 传送模式，就必须在传送期间至少插入 1 个周期的延迟期间。Single-SPI 和 Quad-SPI 同时存在的突发传送例子如图 18.19 所示。图中 (1) ~ (6) 运行内容的说明如下：

1. 这是 SPCMD0 设定的时钟延迟期间。在突发传送的最初传送中，必须至少设定为 1.5 个 QSPCLK。
2. 这是 SPCMD0 设定的 QSSL 无效延迟期间。因为 SPCMD0 的 QSSL 信号电平保持位（SSLKP）为“1”，所以在 QSSL 无效延迟期间后也不将 QSSL 置为无效。通过设定 SPCMD0 的 QSSL 无效延迟设定允许位（SLNDEN）来改变期间长度。如果将 SLNDEN 置“1”，期间长度就取决于从属选择无效延迟寄存器（SSLND）的设定值；如果将 SLNDEN 置“0”，期间长度就为 0 个 QSPCLK。
3. 这是 SPCMD0 设定的下次存取延迟期间。因为 SSLKP 为“1”，所以在此期间也不将 QSSL 置为无效。通过设定 SPCMD0 的下次存取延迟设定允许位（SPNDEN）来改变期间长度。如果将 SPNDEN 置“1”，期间长度就取决于下次存取延迟寄存器（SPND）的设定值；如果将 SPNDEN 置“0”，期间长度就为 0 个 QSPCLK。到此期间为止，根据 SPCMD0 的设定，驱动数据引脚。
4. 这是 SPCMD1 设定的时钟延迟期间。通过设定 SPCMD1 的时钟延迟设定允许位（SCKDEN）来改变期间的长度。如果将 SCKDEN 置“1”，期间长度就取决于时钟延迟设定寄存器（SPCKD）的设定值；如果将 SCKDEN 置“0”，期间长度就为 0 个 QSPCLK。
5. 这是 SPCMD1 设定的 QSSL 无效延迟期间。在突发传送的最后传送中，必须至少设定为 1 个 QSPCLK。因为 SPCMD1 的 SSLKP 为“0”，所以在 QSSL 无效延迟期间后将 QSSL 置为无效。
6. 这是 SPCMD1 设定的下次存取延迟期间。在突发传送的最后传送中，必须至少设定为 1 个 QSPCLK，而且必须通过将 SSLKP 置“0”，使 QSSL 无效。

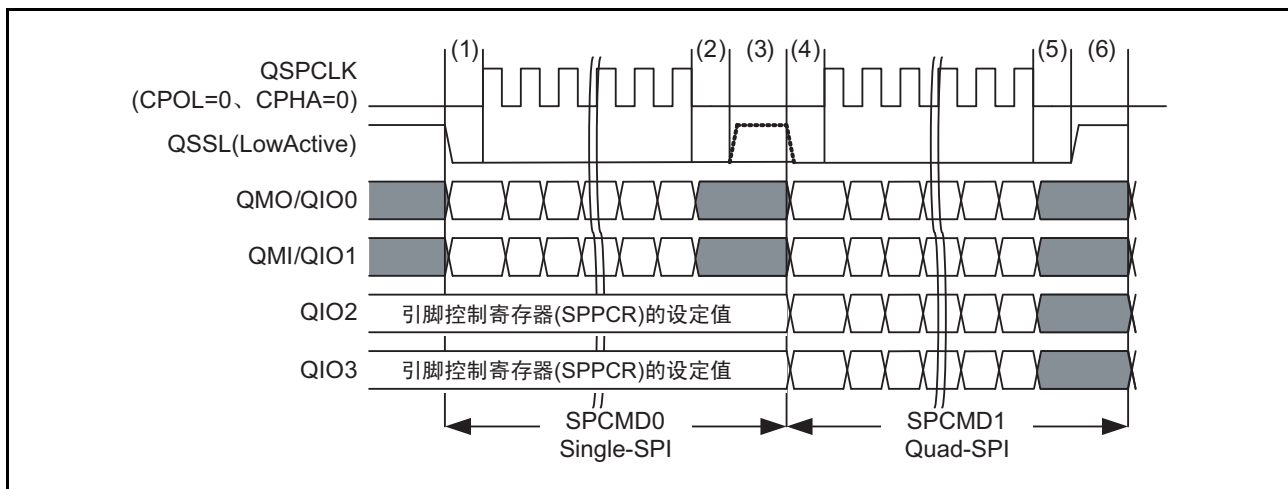


图 18.19 使用 QSSL 信号电平保持功能的突发传送例子（Single-SPI 和 Quad-SPI 同时存在）

在通过此方法设定突发传送时，必须注意以下内容：

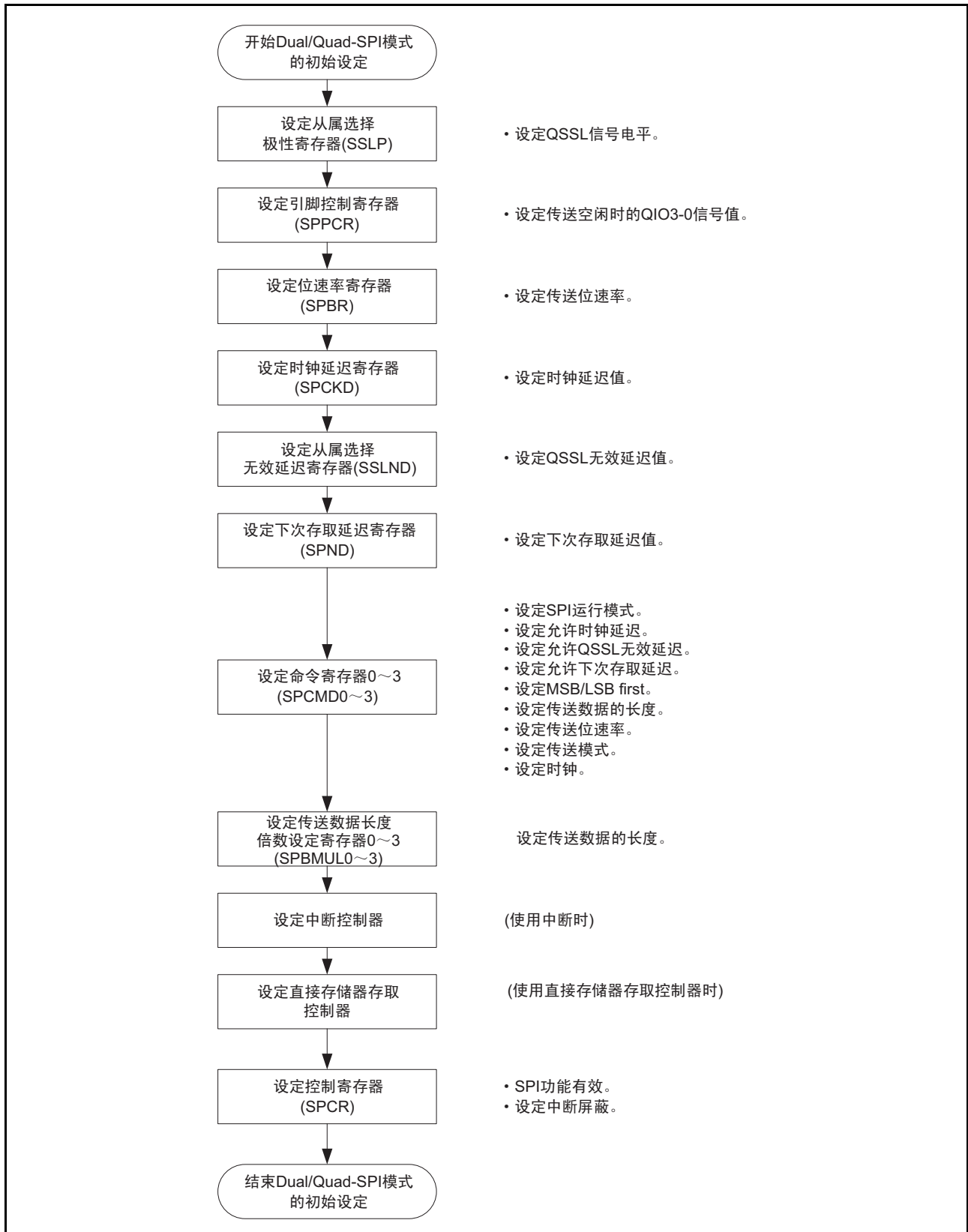
在通过更新命令来更改时钟频比或者时钟极性时，必须插入 (2) ~ (4) 的期间。

如果更改时钟频比，(4) 的期间有可能在设定值的前或者后。

在通过更新命令来更改时钟相位或者 Single/Dual/Quad-SPI 的传送模式时，必须至少插入 (2) 的期间（Dual/Quad-SPI 包括读写的更改）。

(e) 初始设定流程

Dual/Quad-SPI 模式中的初始设定流程例子如图 18.20 所示。有关中断控制器和直接存储器存取控制器的设定，请参照各章节。



(f) 传送运行流程

Dual/Quad-SPI 模式中的运行流程如图 18.21 所示。

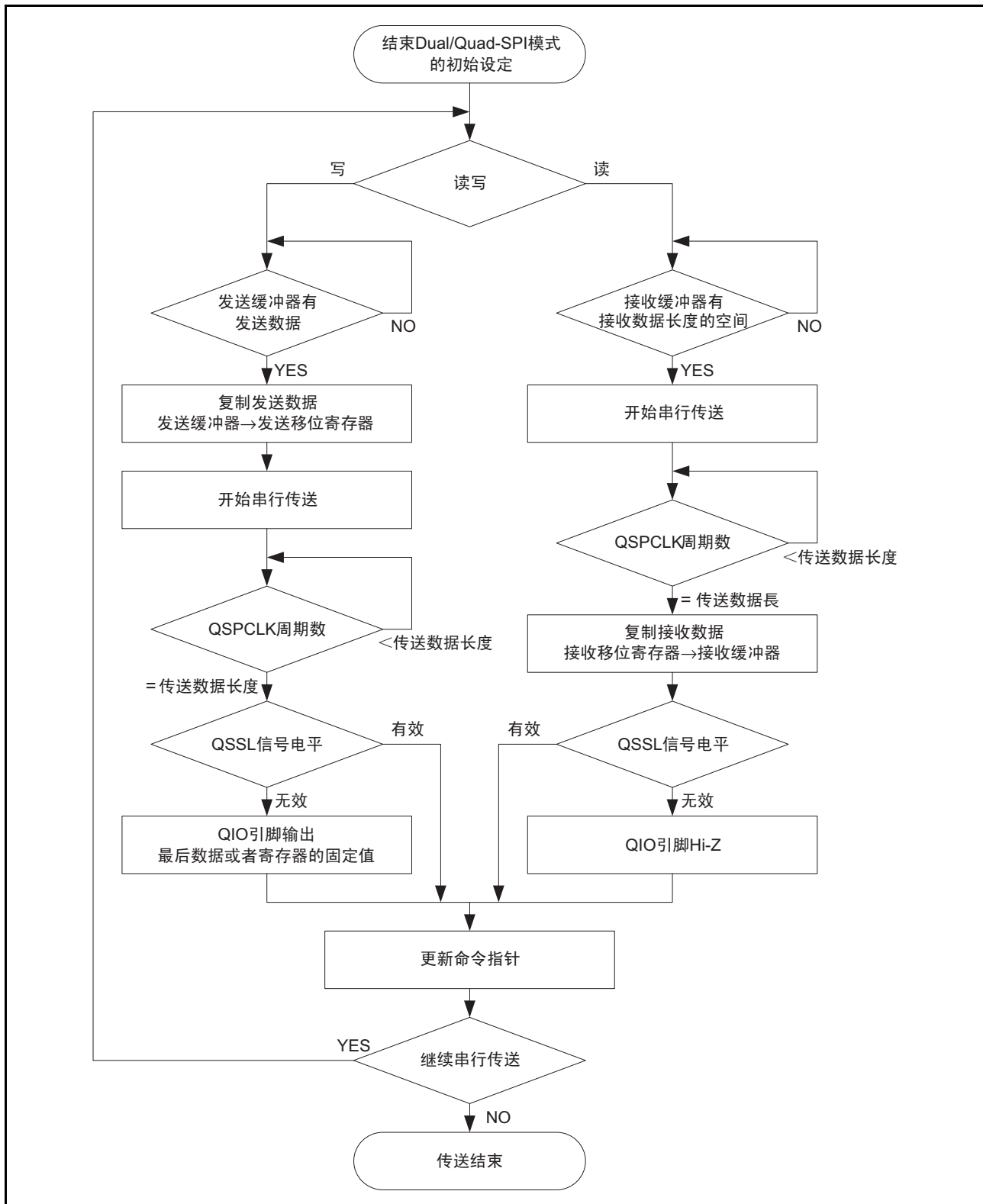


图 18.21 Dual/Quad-SPI 模式中的运行流程

18.4.8 中断源

此模块的中断源有接收缓冲器满和发送缓冲器空。能通过接收缓冲器满中断和发送缓冲器空中断来启动直接存储器存取控制器，进行数据传送。

中断源如表 18.9 所示。一旦表中的中断条件成立，就产生中断。必须在通过 CPU 或者直接存储器存取控制器进行的数据传送中清除中断源。

表 18.9 中断源

名称	中断源	略称	中断条件	直接存储器存取控制器的启动
SPRI	接收缓冲器满	RXI	(SPRIE=1) · (SPRFF=1)	○
SPTI	发送缓冲器空	TXI	(SPTIE=1) · (SPTEF=1)	○

18.4.9 环回模式

此模块有作为测试功能的环回模式。通过将引脚控制寄存器（SPPCR）的环回模式位（SPLP）置“1”，进入环回模式。在环回模式中，切断发送/接收移位寄存器和 QMI/QMO 引脚以及 QIO3-0 之间的线路，连接发送移位寄存器的输出线路和接收移位寄存器的输入线路。环回模式的内部连线概略图如图 18.22 所示。

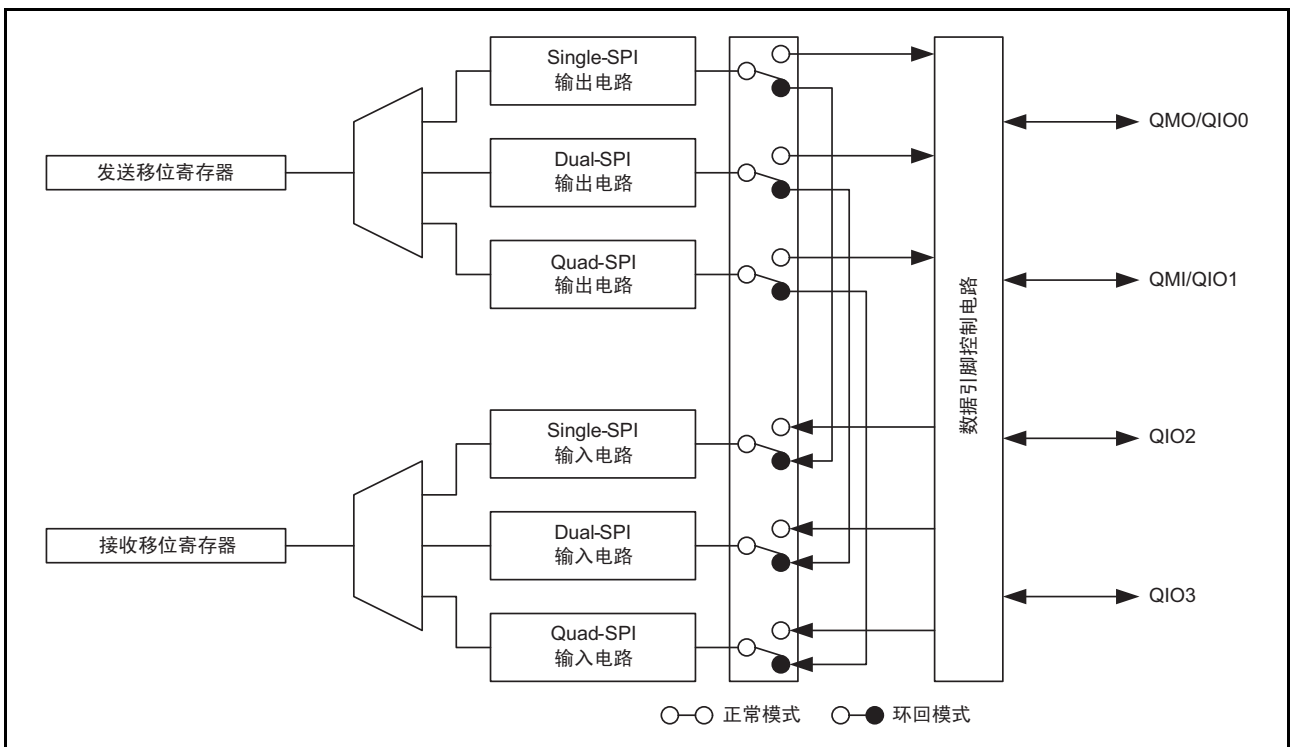


图 18.22 环回模式的内部连线概略图

19. SPI 多 I/O 总线控制器

SPI 多 I/O 总线控制器将控制信号输出到连接 SPI 多 I/O 总线空间的串行闪存，从而能直接连接串行闪存。

19.1 特点

对于连接的串行闪存，此模块能直接读 SPI 多 I/O 总线空间以及通过 SPI 模式发送和接收数据。

1. 串行闪存接口
 - 最多能连接2个串行闪存。
 - 对于1个串行闪存，可选择1位、2位或者4位的数据总线宽度。
2. 外部地址空间读模式
 - 支持最大8G字节的地址空间（连接2个串行闪存的情况）。
 - 能通过监视存取地址，自动控制SPBSSL引脚。
 - 能通过内置读高速缓存（64位块大小×16个入口），高效率地接收数据。
3. SPI运行模式
 - 能任意读写串行闪存。
4. 位速率
 - 在通过内部波特率发生器将B ϕ 分频后生成SPBCLK。
 - 只能将SPBCLK分频比设定为2分频。
5. SPBSSL引脚控制
 - 能设定从SPBSSL信号有效到SPBCLK运行为止的延迟（时钟延迟）。
设定范围：1~8个SPBCLK 设定单位：1个SPBCLK
 - 能更改SPBSSL的极性。

19.2 框图

此模块的整体框图如图 19.1 所示。

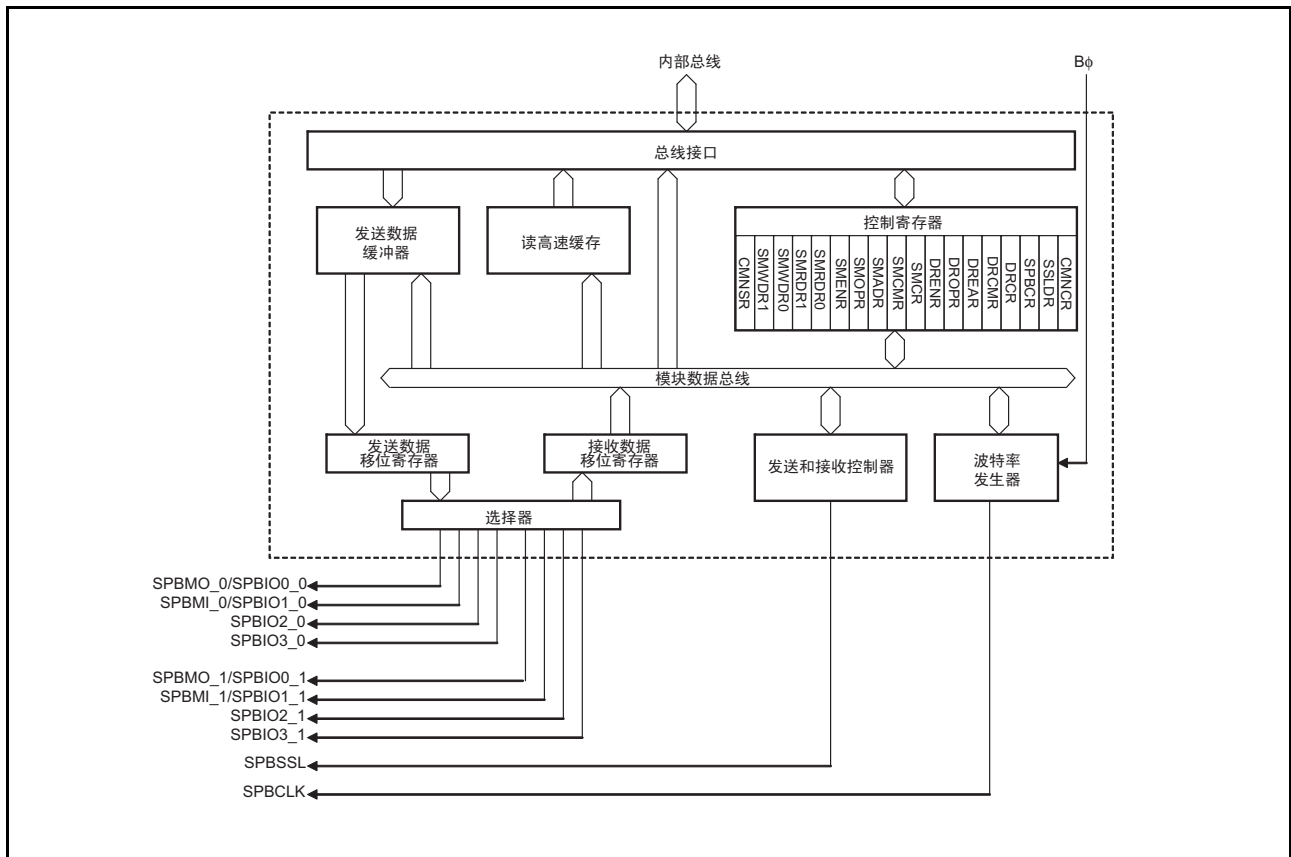


图 19.1 框图

19.3 输入 / 输出引脚

引脚结构如表 19.1 所示。

表 19.1 引脚结构

通道	名称	引脚名	输入 / 输出	功能
共用	时钟引脚	SPBCLK	输出	时钟输出
	从属选择引脚	SPBSSL	输出	从属选择
0	数据 0 引脚	SPBMO_0/SPBIO0_0	输入 / 输出	主控发送数据 / 数据 0
	数据 1 引脚	SPBMI_0/SPBIO1_0	输入 / 输出	主控输入数据 / 数据 1
	数据 2 引脚	SPBIO2_0	输入 / 输出	数据 2
	数据 3 引脚	SPBIO3_0	输入 / 输出	数据 3
1	数据 0 引脚	SPBMO_1/SPBIO0_1	输入 / 输出	主控发送数据 / 数据 0
	数据 1 引脚	SPBMI_1/SPBIO1_1	输入 / 输出	主控输入数据 / 数据 1
	数据 2 引脚	SPBIO2_1	输入 / 输出	数据 2
	数据 3 引脚	SPBIO3_1	输入 / 输出	数据 3

19.4 寄存器说明

寄存器结构如表 19.2 所示。

表 19.2 寄存器结构

寄存器名	略称	初始值	R/W	地址	存取长度
公共控制寄存器	CMNCR	H'00AA4000	R/W	H'FFFC1C00	32
SSL 延迟寄存器	SSLDR	H'00000000	R/W	H'FFFC1C04	32
位速率设定寄存器	SPBCR	H'00000003	R/W	H'FFFC1C08	32
数据读控制寄存器	DRCR	H'00000000	R/W	H'FFFC1C0C	32
数据读命令设定寄存器	DRCMR	H'00000000	R/W	H'FFFC1C10	32
数据读扩展地址设定寄存器	DREAR	H'00000000	R/W	H'FFFC1C14	32
数据读选项设定寄存器	DROPR	H'00000000	R/W	H'FFFC1C18	32
数据读允许设定寄存器	DRENDR	H'00004700	R/W	H'FFFC1C1C	32
SPI 模式控制寄存器	SMCR	H'00000000	R/W	H'FFFC1C20	32
SPI 模式命令设定寄存器	SMCMR	H'00000000	R/W	H'FFFC1C24	32
SPI 模式地址设定寄存器	SMADR	H'00000000	R/W	H'FFFC1C28	32
SPI 模式选项设定寄存器	SMOPR	H'00000000	R/W	H'FFFC1C2C	32
SPI 模式允许设定寄存器	SMENR	H'00004000	R/W	H'FFFC1C30	32
SPI 模式读数据寄存器 0	SMRDR0	不定值	R	H'FFFC1C38	8、16、32
SPI 模式读数据寄存器 1	SMRDR1	不定值	R	H'FFFC1C3C	8、16、32
SPI 模式写数据寄存器 0	SMWDR0	H'00000000	R/W	H'FFFC1C40	8、16、32
SPI 模式写数据寄存器 1	SMWDR1	H'00000000	R/W	H'FFFC1C44	8、16、32
公共状态寄存器	CMNSR	H'00000001	R	H'FFFC1C48	32

19.4.1 公共控制寄存器 (CMNCR)

CMNCR 是控制 SPI 多 I/O 总线控制器的 32 位寄存器，此寄存器的设定被反映到外部地址空间读模式和 SPI 运行模式。

必须在 CMNSR 寄存器的 SSLF 标志为“0”时更改此寄存器，CMNSR 寄存器的 SSLF 标志为“1”时不保证运行。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MD	—	—	—	—	—	—	—	MOII03[1:0]	MOII02[1:0]	MOII01[1:0]	MOII00[1:0]				
初始值:	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0
R/W:	R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IO3FV[1:0]	IO2FV[1:0]	—	—	IO0FV[1:0]	—	CPHAT	CPHAR	SSLP	CPOL	—	BSZ[1:0]				
初始值:	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R	R/W	R/W

位	位名	初始值	R/W	说明
31	MD	0	R/W	运行模式的切换 切换运行模式。 0: 外部地址空间读模式 1: SPI 运行模式
30 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23、22	MOII03 [1:0]	10	R/W	SPBSSL 输出空闲值固定 SPBIO3_0、SPBIO3_1 在 SPBSSL 无效期间固定 SPBIO3_0 和 SPBIO3_1 的输出值。 00: 输出值为“0” 01: 输出值为“1” 10: 输出值为上次传送的最后位（如果上次传送的最后位为 Hi-Z，输出值就为 Hi-Z） 11: 输出值为 Hi-Z
21、20	MOII02 [1:0]	10	R/W	SPBSSL 输出空闲值固定 SPBIO2_0、SPBIO2_1 在 SPBSSL 无效期间固定 SPBIO2_0 和 SPBIO2_1 的输出值。 00: 输出值为“0” 01: 输出值为“1” 10: 输出值为上次传送的最后位（如果上次传送的最后位为 Hi-Z，输出值就为 Hi-Z） 11: 输出值为 Hi-Z
19、18	MOII01 [1:0]	10	R/W	SPBSSL 输出空闲值固定 SPBIO1_0、SPBIO1_1 在 SPBSSL 无效期间固定 SPBIO1_0 和 SPBIO1_1 的输出值。 00: 输出值为“0” 01: 输出值为“1” 10: 输出值为上次传送的最后位（如果上次传送的最后位为 Hi-Z，输出值就为 Hi-Z） 11: 输出值为 Hi-Z

位	位名	初始值	R/W	说明
17、16	MOIIO0 [1:0]	10	R/W	SPBSSL 输出空闲值固定 SPBIO0_0、SPBIO0_1 在 SPBSSL 无效期间固定 SPBIO0_0 和 SPBIO0_1 的输出值。 00: 输出值为 “0” 01: 输出值为 “1” 10: 输出值为上次传送的最后位 (如果上次传送的最后位为 Hi-Z, 输出值就为 Hi-Z) 11: 输出值为 Hi-Z
15、14	IO3FV[1:0]	01	R/W	1 位或者 2 位宽时的 SPBIO3_0、SPBIO3_1 固定值 在 1 位或者 2 位宽时固定 SPBIO3_0 引脚和 SPBIO3_1 引脚的输出值。 00: 输出值固定为 “0” 01: 输出值固定为 “1” 10: 输出值为上次传送的最后位 (如果上次传送的最后位为 Hi-Z, 输出值就为 Hi-Z) 11: 输出值为 Hi-Z
13、12	IO2FV[1:0]	00	R/W	1 位或者 2 位宽时的 SPBIO2_0、SPBIO2_1 固定值 在 1 位或者 2 位宽时固定 SPBIO2_0 引脚和 SPBIO2_1 引脚的输出值。 00: 输出值固定为 “0” 01: 输出值固定为 “1” 10: 输出值为上次传送的最后位 (如果上次传送的最后位为 Hi-Z, 输出值就为 Hi-Z) 11: 输出值为 Hi-Z
11、10	—	全 0	R	保留位 读写值都为 “0”。
9、8	IO0FV[1:0]	00	R/W	输入 1 位宽时的 SPBIO0_0、SPBIO0_1 固定值 在输入 1 位宽时固定 SPBIO0_0 引脚和 SPBIO0_1 引脚的输出值。 00: 输出值固定为 “0” 01: 输出值固定为 “1” 10: 输出值为上次传送的最后位 (如果上次传送的最后位为 Hi-Z, 输出值就为 Hi-Z) 11: 输出值为 Hi-Z
7	—	0	R	保留位 读写值都为 “0”。
6	CPHAT	0	R/W	输出移位的设定 设定输出数据的 SPBCLK 边沿。 CPHAT 和 CPHAR 的设定请参照 CPHAR 说明。 0: 在偶数边沿发送数据 1: 在奇数边沿发送数据

位	位名	初始值	R/W	说明															
5	CPHAR	0	R/W	输入锁存的设定 设定接收数据的 SPBCLK 边沿。 CPHAT 和 CPHAR 的设定请参照下表。 0: 在奇数边沿接收数据 1: 在偶数边沿接收数据 CPHAT和CPHAR的设定值 <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>CPHAT</th> <th>CPHAR</th> <th></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>能设定</td> </tr> <tr> <td>0</td> <td>1</td> <td>能设定</td> </tr> <tr> <td>1</td> <td>0</td> <td>禁止设定</td> </tr> <tr> <td>1</td> <td>1</td> <td>能设定</td> </tr> </tbody> </table>	CPHAT	CPHAR		0	0	能设定	0	1	能设定	1	0	禁止设定	1	1	能设定
CPHAT	CPHAR																		
0	0	能设定																	
0	1	能设定																	
1	0	禁止设定																	
1	1	能设定																	
4	SSLP	0	R/W	SPBSSL 信号的极性设定 设定 SPBSSL 信号的极性。 0: SPBSSL 信号为 Low 电平有效 1: SPBSSL 信号为 High 电平有效															
3	CPOL	0	R/W	SPBSSL 无效期间的 SPBCLK 输出方向 设定 SPBSSL 无效期间的 SPBCLK 输出方向。 0: 在 SPBSSL 无效时 SPBCLK 输出低电平 1: 在 SPBSSL 无效时 SPBCLK 输出高电平															
2	—	0	R	保留位 读写值都为“0”。															
1、0	BSZ[1:0]	00	R/W	数据总线宽度的指定 指定连接的串行闪存个数。 00: 1 个 01: 2 个 1X: 禁止设定 【注】 如果更改这些位, 就必须通过将 DRCR 寄存器的 RCF 位置“1”, 清除读高速缓存的全部入口。															

19.4.2 SSL 延迟寄存器 (SSLDLDR)

SSLDLDR 是 32 位寄存器，调整 SPBSSL 信号和 SPBCLK 信号之间的时序。

此寄存器的设定被反映到外部地址空间读模式和 SPI 运行模式。

必须在 CMNSR 寄存器的 SSLF 标志为“0”时更改此寄存器，CMNSR 寄存器的 SSLF 标志为“1”时不保证运行。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	SPNDL[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SLNDL[2:0]			—	—	—	—	—	SCKDL[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 19	—	全 0	R	保留位 读写值都为“0”。
18 ~ 16	SPNDL[2:0]	000	R/W	下次存取延迟的设定 设定从传送结束到开始下一次传送的期间（下次存取）。 000: 1 个 SPBCLK 上述以外: 禁止设定
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 8	SLNDL[2:0]	000	R/W	SPBSSL 无效延迟的设定 设定从传送的最后 SPBCLK 边沿到将 SPBSSL 引脚置为无效的期间（SPBSSL 无效延迟）。 000: 1.5 个 SPBCLK 上述以外: 禁止设定
7 ~ 3	—	全 0	R	保留位 读写值都为“0”。
2 ~ 0	SCKDL[2:0]	000	R/W	时钟延迟的设定 设定从 SPBSSL 引脚有效到 SPBCLK 振荡的期间（时钟延迟）。 000: 1 个 SPBCLK 001: 2 个 SPBCLK 010: 3 个 SPBCLK 011: 4 个 SPBCLK 100: 5 个 SPBCLK 101: 6 个 SPBCLK 110: 7 个 SPBCLK 111: 8 个 SPBCLK

19.4.3 位速率设定寄存器 (SPBCR)

SPBCR 是设定位速率的 32 位寄存器。

此寄存器的设定被反映到外部地址空间读模式和 SPI 运行模式。

必须在 CMNSR 寄存器的 SSLF 标志为“0”时更改此寄存器，CMNSR 寄存器的 SSLF 标志为“1”时不保证运行。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPBR[7:0]								—	—	—	—	—	—	BRDV[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 16	—	全 0	R	保留位 读写值都为“0”。
15 ~ 8	SPBR[7:0]	全 0	R/W	位速率的设定 设定位速率。位速率取决于这些位和 BRDV[1:0] 位的组合，详细内容请参照表 19.3。
7 ~ 2	—	全 0	R	保留位 读写值都为“0”。
1、0	BRDV[1:0]	11	R/W	位速率分频的设定 设定位速率。位速率取决于这些位和 SPBR[7:0] 位的组合。SPBR 的设定值决定基准位速率。这些位的设定值用于选择基准位速率的无分频或者 2 分频后的位速率。 00: 选择基准位速率 01: 选择基准位速率的 2 分频 上述以外: 禁止设定

(1) 位速率

通过 SPBR[7:0] 和 BRDV[1:0] 设定位速率。禁止 2 分频以外的设定。

表 19.3 SPBR[7:0]、BRDV[1:0] 位的设定值和位速率

SPBR[7:0] 的设定值	BRDV[1:0] 的设定值	分频比	位速率	
			B ϕ =66.67MHz	B ϕ =133.33MHz
0	1	2	33.33Mbps	66.66Mbps
1	0	2	33.33Mbps	66.66Mbps

19.4.4 数据读控制寄存器 (DRCR)

DRCR 是 32 位寄存器，设定外部地址空间读模式的运行。

必须在 CMNSR 寄存器的 SSLF 标志为“0”时更改此寄存器，CMNSR 寄存器的 SSLF 标志为“1”时不保证运行。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	RBURST[3:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RCF	RBE	—	—	—	—	—	—	—	SSLE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	W	R/W	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 20	—	全 0	R	保留位 读写值都为“0”。
19 ~ 16	RBURST [3:0]	0000	R/W	读数据的突发长度 设定读数据时的突发长度。 这些位在 RBE 位为“1”时有效。 0000: 连续的 1 个数据长度 0001: 连续的 2 个数据长度 : 1110: 连续的 15 个数据长度 1111: 连续的 16 个数据长度 1 个数据长度是 64 位。
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9	RCF	0	W	读高速缓存的转储清除 如果给此位写“1”，就清除读高速缓存的全部入口。 读取值总是“0”。
8	RBE	0	R/W	读突发 设定读时的突发 ON/OFF。 0: 根据存取长度进行读操作 1: 读高速缓存有效并且读 RBURST[3:0] 位指定突发次数的数据
7 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	SSLE	0	R/W	SPBSSL 无效设定 设定读突发时的 SPBSSL 无效条件。 在正常读的情况下，在每次存取时将 SPBSSL 置为无效。 0: 每当以突发长度设定的数据传送结束时将 SPBSSL 置为无效 1: 当存取地址与上次的传送地址不连续时将 SPBSSL 置为无效

19.4.5 数据读命令设定寄存器 (DRCMR)

DRCMR 是 32 位寄存器，设定外部地址空间读模式的发行命令。

必须在 CMNSR 寄存器的 TEND 标志为“1”时更改此寄存器，CMNSR 寄存器的 TEND 标志为“0”时不保证运行。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CMD[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	OCMD[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23 ~ 16	CMD[7:0]	H'00	R/W	命令 设定命令。
15 ~ 8	—	全 0	R	保留位 读写值都为“0”。
7 ~ 0	OCMD[7:0]	H'00	R/W	可选命令 设定可选命令。

19.4.6 数据读扩展地址设定寄存器 (DREAR)

DREAR 是 32 位寄存器，设定以 32 位输出串行闪存地址时的地址。

必须在 CMNSR 寄存器的 TEND 标志为“1”时更改此寄存器，CMNSR 寄存器的 TEND 标志为“0”时不保证运行。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	EAV[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	EAC[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23 ~ 16	EAV[7:0]	H'00	R/W	32 位扩展高位地址固定值的设定 在以 32 位输出串行闪存地址时，此寄存器设定由 EAC[2:0] 位指定的外部地址的高位地址位值。 bit0 对应串行闪存的地址位 [25]，bit7 对应 bit[32]。 此设定在 DRENr 寄存器的 ADE[3] 位为“1”时有效。 当 EAC[2:0] 为“000”时，将串行闪存地址 [32:25] 的固定值设定到 EAV[7:0]。 当 EAC[2:0] 为“001”时，将串行闪存地址 [32:26] 的固定值设定到 EAV[7:1]。 (1) 当 CMNCR 寄存器的 BSZ[1:0] 位为“00”（连接 1 个串行闪存）时使用串行闪存的地址 [31:0] 进行存取。 (2) 当 CMNCR 寄存器的 BSZ[1:0] 位为“01”（连接 2 个串行闪存）时使用串行闪存的地址 [32:1] 进行存取。
15 ~ 3	—	全 0	R	保留位 读写值都为“0”。
2 ~ 0	EAC[2:0]	000	R/W	32 位扩展外部地址有效范围的设定 在以 32 位输出串行闪存地址时，设定用作串行闪存地址的外部地址范围。 此设定在 DRENr 寄存器的 ADE[3] 位为“1”时有效。 000：外部地址的 bit[24:0] 有效 001：外部地址的 bit[25:0] 有效 上述以外：禁止设定

19.4.7 数据读选项设定寄存器 (DROPR)

DROPR 是 32 位寄存器，设定读外部地址空间时的选项数据。

必须在 CMNSR 寄存器的 TEND 标志为“1”时更改此寄存器，CMNSR 寄存器的 TEND 标志为“0”时不保证运行。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OPD3[7:0]								OPD2[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OPD1[7:0]								OPD0[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	OPD3[7:0]	H'00	R/W	选项数据 3 设定选项数据 3。
23 ~ 16	OPD2[7:0]	H'00	R/W	选项数据 2 设定选项数据 2。
15 ~ 8	OPD1[7:0]	H'00	R/W	选项数据 1 设定选项数据 1。
7 ~ 0	OPD0[7:0]	H'00	R/W	选项数据 0 设定选项数据 0。

【注】 输出顺序为 OPD3→OPD2→OPD1→OPD0。

19.4.8 数据读允许设定寄存器 (DRENr)

DRENr 是 32 位寄存器，设定读外部地址空间时的命令、可选命令、地址、选项数据和读数据的位宽以及设定读数据以外的输出允许。

必须在 CMNSR 寄存器的 TEND 标志为“1”时更改此寄存器，CMNSR 寄存器的 TEND 标志为“0”时不保证运行。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CDB[1:0]		OCDB[1:0]		—	—	ADB[1:0]		—	—	OPDB[1:0]		—	—	DRDB[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	CDE	—	OCDE	ADE[3:0]			OPDE[3:0]			—	—	—	—		
初始值:	0	1	0	0	0	1	1	1	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

位	位名	初始值	R/W	说明
31、30	CDB[1:0]	00	R/W	命令位宽 设定命令的位宽。 00: 1 位 01: 2 位 10: 4 位 11: 禁止设定
29、28	OCDB[1:0]	00	R/W	可选命令位宽 设定可选命令的位宽。 00: 1 位 01: 2 位 10: 4 位 11: 禁止设定
27、26	—	全 0	R	保留位 读写值都为“0”。
25、24	ADB[1:0]	00	R/W	地址位宽 设定地址的位宽。 00: 1 位 01: 2 位 10: 4 位 11: 禁止设定
23、22	—	全 0	R	保留位 读写值都为“0”。
21、20	OPDB[1:0]	00	R/W	选项数据位宽 设定选项数据的位宽。 00: 1 位 01: 2 位 10: 4 位 11: 禁止设定
19、18	—	全 0	R	保留位 读写值都为“0”。

位	位名	初始值	R/W	说明
17、16	DRDB[1:0]	00	R/W	读数据的位宽 设定读数据的位宽。 00: 1 位 01: 2 位 10: 4 位 11: 禁止设定
15	—	0	R	保留位 读写值都为“0”。
14	CDE	1	R/W	命令允许 设定命令的输出。 0: 不输出 1: 输出
13	—	0	R	保留位 读写值都为“0”。
12	OCDE	0	R/W	可选命令允许 设定可选命令的输出。 0: 不输出 1: 输出
11 ~ 8	ADE[3:0]	0111	R/W	地址允许 设定地址的输出。 必须在设定以下内容时使用这些位，否则不保证运行。 (1) CMNCR 寄存器的 BSZ[1:0] 位为“00”（连接 1 个串行闪存）的情况 0000: 不输出 0111: 输出 Address[23:0] 1111: 输出 Address[31:0] 上述以外: 禁止设定 (2) CMNCR 寄存器的 BSZ[1:0] 位为“01”（连接 2 个串行闪存）的情况 0000: 不输出 0111: 输出 Address[24:1] 1111: 输出 Address[32:1] 上述以外: 禁止设定
7 ~ 4	OPDE[3:0]	0000	R/W	选项数据允许 设定选项数据的输出。 必须在设定以下内容时使用这些位，否则不保证运行。 0000: 不输出 1000: 输出 OPD3 1100: 输出 OPD3 和 OPD2 1110: 输出 OPD3、OPD2 和 OPD1 1111: 输出 OPD3、OPD2、OPD1 和 OPD0 上述以外: 禁止设定
3 ~ 0	—	全 0	R	保留位 读写值都为“0”。

19.4.9 SPI 模式控制寄存器 (SMCR)

SMCR 是 32 位寄存器，设定 SPI 运行模式的运行。

必须在 CMNSR 寄存器的 TEND 标志为“1”时更改此寄存器，CMNSR 寄存器的 TEND 标志为“0”时不保证运行。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SSLKP	—	—	—	—	—	SPIRE	SPIWE	SPIE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R/W	R/W	W

位	位名	初始值	R/W	说明
31 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	SSLKP	0	R/W	SPBSSL 信号电平的保持 决定传送结束后的 SPBSSL 状态。 0: 在传送结束时将 SPBSSL 信号置为无效 1: 从传送结束到开始下次存取为止保持 SPBSSL 信号电平 【注】 如果通过 SPIDB[1:0] 位将传送数据位宽设定为 2 位或者 4 位，就不能同时将 SPIRE 位和 SSLKP 位置“1”。
7 ~ 3	—	全 0	R	保留位 读写值都为“0”。
2	SPIRE	0	R/W	数据读允许 允许 SPI 运行模式中的读操作。 0: 不读数据 1: 读数据 【注】 如果通过 SPIDB[1:0] 位将传送数据位宽设定为 2 位或者 4 位，就不能同时将 SPIRE 位和 SPIWE 位置“1”。
1	SPIWE	0	R/W	数据写允许 允许 SPI 运行模式中的写操作。 0: 不写数据 1: 写数据 【注】 如果通过 SPIDB[1:0] 位将传送数据位宽设定为 2 位或者 4 位，就不能同时将 SPIRE 位和 SPIWE 位置“1”。
0	SPIE	0	W	SPI 数据传送的允许 通过将此位置“1”进行数据传送。 此位只在 CMNSR 寄存器的 TEND 位为“1”时有效。如果将 CMNSR 寄存器的 TEND 位置“0”，就不保证置“1”后的运行。 如果在 t3 期间将此位置“1”，就在 t3 期间结束后开始传送数据。 读取值总是“0”。 【注】 即使 SPIRE 位和 SPIWE 位为“0”，也输出被设定为允许输出的命令、可选命令、地址和选项数据。

19.4.10 SPI 模式命令设定寄存器 (SMCMR)

SMCMR 是 32 位寄存器，设定 SPI 运行模式的发行命令。

必须在 CMNSR 寄存器的 TEND 标志为“1”时更改此寄存器，CMNSR 寄存器的 TEND 标志为“0”时不保证运行。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—								CMD[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—								OCMD[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23 ~ 16	CMD[7:0]	H'00	R/W	命令 设定命令。
15 ~ 8	—	全 0	R	保留位 读写值都为“0”。
7 ~ 0	OCMD[7:0]	H'00	R/W	可选命令 设定可选命令。

19.4.11 SPI 模式地址设定寄存器 (SMADR)

SMADR 是 32 位寄存器，设定 SPI 运行模式的地址。

必须在 CMNSR 寄存器的 TEND 标志为“1”时更改此寄存器，CMNSR 寄存器的 TEND 标志为“0”时不保证运行。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADR[31:24]								ADR[23:16]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADR[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	ADR[31:24]	H'00	R/W	地址 设定以 32 位输出串行闪存地址时的 bit31 ~ 24。 此设定在 SMENR 寄存器的 ADE[3] 位为“1”时有效。
23 ~ 0	ADR[23:0]	H'000000	R/W	地址 设定地址。

19.4.12 SPI 模式选项设定寄存器 (SMOPR)

SMOPR 是 32 位寄存器，设定 SPI 运行模式的选项数据。

必须在 CMNSR 寄存器的 TEND 标志为“1”时更改此寄存器，CMNSR 寄存器的 TEND 标志为“0”时不保证运行。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OPD3[7:0]								OPD2[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OPD1[7:0]								OPD0[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	OPD3[7:0]	H'00	R/W	选项数据 3 设定选项数据 3。
23 ~ 16	OPD2[7:0]	H'00	R/W	选项数据 2 设定选项数据 2。
15 ~ 8	OPD1[7:0]	H'00	R/W	选项数据 1 设定选项数据 1。
7 ~ 0	OPD0[7:0]	H'00	R/W	选项数据 0 设定选项数据 0。

【注】 输出顺序为 OPD3→OPD2→OPD1→OPD0。

19.4.13 SPI 模式允许设定寄存器 (SMENR)

SMENR 是 32 位寄存器，设定 SPI 运行模式的命令、可选命令、地址、选项数据和传送数据的位宽和允许。禁止将命令、可选命令、地址、选项数据和传送数据全部设定为禁止，必须至少允许其中 1 个。

必须在 CMNSR 寄存器的 TEND 标志为“1”时更改此寄存器，CMNSR 寄存器的 TEND 标志为“0”时不保证运行。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CDB[1:0]		OCDB[1:0]		—	—	ADB[1:0]		—	—	OPDB[1:0]		—	—	SPIDB[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	CDE	—	OCDE	ADE[3:0]				OPDE[3:0]				SPIDE[3:0]			
初始值:	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31、30	CDB[1:0]	00	R/W	命令位宽 设定命令的位宽。 00: 1 位 01: 2 位 10: 4 位 11: 禁止设定
29、28	OCDB[1:0]	00	R/W	可选命令位宽 设定可选命令的位宽。 00: 1 位 01: 2 位 10: 4 位 11: 禁止设定
27、26	—	全 0	R	保留位 读写值都为“0”。
25、24	ADB[1:0]	00	R/W	地址位宽 设定地址的位宽。 00: 1 位 01: 2 位 10: 4 位 11: 禁止设定
23、22	—	全 0	R	保留位 读写值都为“0”。
21、20	OPDB[1:0]	00	R/W	选项数据位宽 设定选项数据的位宽。 00: 1 位 01: 2 位 10: 4 位 11: 禁止设定
19、18	—	全 0	R	保留位 读写值都为“0”。

位	位名	初始值	R/W	说明
17、16	SPIDB[1:0]	00	R/W	传送数据位宽 设定传送数据的位宽。 00: 1 位 01: 2 位 10: 4 位 11: 禁止设定
15	—	0	R	保留位 读写值都为“0”。
14	CDE	1	R/W	命令的允许 设定命令的输出。 0: 不输出 1: 输出
13	—	0	R	保留位 读写值都为“0”。
12	OCDE	0	R/W	可选命令的允许 设定可选命令的输出。 0: 不输出 1: 输出
11 ~ 8	ADE[3:0]	0000	R/W	地址的允许 设定地址的输出。 必须在设定以下内容时使用这些位，否则不保证运行。 0000: 不输出 0100: 输出 ADR[23:16] 0110: 输出 ADR[23:8] 0111: 输出 ADR[23:0] 1111: 输出 ADR[31:0] 上述以外: 禁止设定
7 ~ 4	OPDE[3:0]	0000	R/W	选项数据的允许 设定选项数据的输出。 必须在设定以下内容时使用这些位，否则不保证运行。 0000: 不输出 1000: 输出 OPD3 1100: 输出 OPD3 和 OPD2 1110: 输出 OPD3、OPD2 和 OPD1 1111: 输出 OPD3、OPD2、OPD1 和 OPD0 上述以外: 禁止设定

位	位名	初始值	R/W	说明
3 ~ 0	SPIDE[3:0]	0000	R/W	传送数据的允许 设定传送数据的有效。 有效数据因 CMNCR 寄存器的 BSZ[1:0] 位而不同。 必须在设定以下内容时使用这些位，否则不保证运行。 (1) CMNCR 寄存器的 BSZ[1:0] 位为“00”（连接 1 个串行闪存）的情况 0000：不传送 1000：8 位传送（将 DATA[31:24] 置为有效） 1100：16 位传送（将 DATA[31:16] 置为有效） 1111：32 位传送（将 DATA[31:0] 置为有效） 上述以外：禁止设定 (2) CMNCR 寄存器的 BSZ[1:0] 位为“01”（连接 2 个串行闪存）的情况 0000：不传送 1000：16 位传送（将 DATA[63:48] 置为有效） 1100：32 位传送（将 DATA[63:32] 置为有效） 1111：64 位传送（将 DATA[63:0] 置为有效） 上述以外：禁止设定

19.4.14 SPI 模式读数据寄存器 0（SMRDR0）

SMRDR0 是 32 位寄存器，保存 SPI 运行模式的读数据。

必须在 CMNSR 寄存器的 TEND 标志为“1”时读此寄存器，CMNSR 寄存器的 TEND 标志为“0”时不保证运行。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDATA0[31:16]															
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDATA0[15:0]															
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 0	RDATA0 [31:0]	不定值	R	读数据 保存 SPI 运行模式的读数据。 数据位因 CMNCR 寄存器的 BSZ[1:0] 位而不同。 当 BSZ[1:0] 为“00”时，为读数据 [31:0]。 当 BSZ[1:0] 为“01”时，为读数据 [63:32]。

一旦结束 SPI 运行模式的接收，就改写此寄存器和 SMRDR1 寄存器的内容。

当 SPI 运行模式的接收结束时，必须读数据。

19.4.15 SPI 模式读数据寄存器 1 (SMRDR1)

SMRDR1 是 32 位寄存器，保存 SPI 运行模式的读数据。

此寄存器的值在 CMNCR 寄存器的 BSZ[1:0] 位为“01”（连接 2 个串行闪存）时有效，在 BSZ[1:0] 位为“00”（连接 1 个串行闪存）时无效。

必须在 CMNSR 寄存器的 TEND 标志为“1”时读此寄存器，CMNSR 寄存器的 TEND 标志为“0”时不保证运行。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDATA1[31:16]															
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDATA1[15:0]															
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 0	RDATA1 [31:0]	不定值	R	读数据 保存 SPI 运行模式的读数据。 这些位的值只在 CMNCR 寄存器的 BSZ[1:0] 位为“01”（连接 2 个串行闪存）时有效，在 BSZ[1:0] 位为“00”（连接 1 个串行闪存）无效。 当 BSZ[1:0] 为“01”时，为读数据 [31:0]。

19.4.16 SPI 模式写数据寄存器 0 (SMWDR0)

SMWDR0 是 32 位寄存器，设定 SPI 运行模式的写数据。

必须在 CMNSR 寄存器的 TEND 标志为“1”时更改此寄存器，CMNSR 寄存器的 TEND 标志为“0”时不保证运行。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	WDATA0[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WDATA0[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 0	WDATA0 [31:0]	全 0	R/W	写数据 保存 SPI 运行模式的写数据。 数据位因 CMNCR 寄存器的 BSZ[1:0] 位而不同。 当 BSZ[1:0] 为“00”时，为写数据 [31:0]。 当 BSZ[1:0] 为“01”时，为写数据 [63:32]。

19.4.17 SPI 模式写数据寄存器 1 (SMWDR1)

SMWDR1 是 32 位寄存器，设定 SPI 运行模式的写数据。

此寄存器的值在 CMNCR 寄存器的 BSZ[1:0] 位为“01”（连接 2 个串行闪存）有效，在 BSZ[1:0] 位为“00”（连接 1 个串行闪存）时无效。

必须在 CMNSR 寄存器的 TEND 标志为“1”时更改此寄存器，CMNSR 寄存器的 TEND 标志为“0”时不保证运行。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	WDATA1[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WDATA1[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 0	WDATA1 [31:0]	全 0	R/W	写数据 保存 SPI 运行模式的写数据。 这些位的值只在 CMNCR 寄存器的 BSZ[1:0] 位为“01”（连接 2 个串行闪存）时有效在 BSZ[1:0] 位为“00”（连接 1 个串行闪存）无效。 当 BSZ[1:0] 为“01”时，为写数据 [31:0]。

19.4.18 公共状态寄存器 (CMNSR)

CMNSR 是保存运行状态标志的 32 位寄存器。

此寄存器的设定被反映到外部地址空间读模式和 SPI 运行模式。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSLF	TEND
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 2	—	全 0	R	保留位 读写值都为“0”。
1	SSLF	0	R	SPBSSL 引脚监视器 0: SPBSSL 引脚无效 1: SPBSSL 引脚有效
0	TEND	1	R	传送结束标志 表示数据传送已结束。 0: 表示正在传送 1: 表示传送已结束

19.5 运行说明

19.5.1 系统结构

此模块能直接连接 1 个或者 2 个串行闪存（数据宽度为 1 位、2 位或者 4 位）。通过 CMNCR 寄存器的 BSZ[1:0] 位切换连接个数。

连接 1 个或者 2 个串行闪存的系统结构例子如图 19.2 和图 19.3 所示。

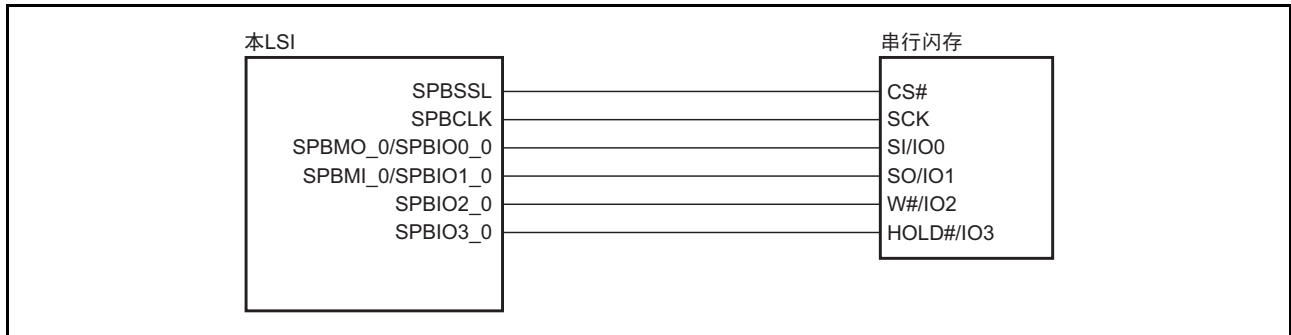


图 19.2 数据宽度为 4 位并且连接 1 个串行闪存的连接例子（CMNCR 寄存器的 BSZ[1:0] 位为“00”）

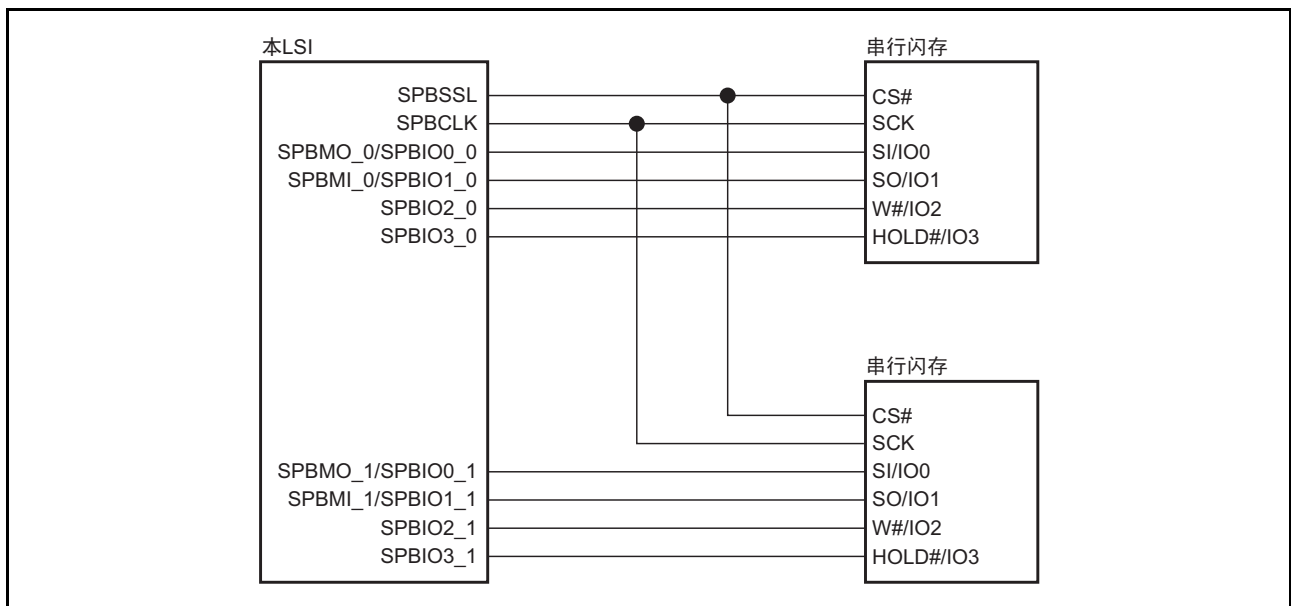


图 19.3 数据宽度为 4 位并且连接 2 个串行闪存的连接例子（CMNCR 寄存器的 BSZ[1:0] 位为“01”）

19.5.2 地址映像

在外部地址空间读模式中，将连接的串行闪存分配到 SPI 多 I/O 总线空间。能存取地址空间最大值因串行闪存的连接个数而不同。能通过和 DREAR 寄存器的组合，在连接 1 个串行闪存时最多存取 4G 字节，在连接 2 个串行闪存时最多存取 8G 字节。

表 19.4 地址映像

串行闪存的连接个数	内部地址	高速缓存	最大存取区域
1 个	H'18000000 ~ H'1BFFFFFF	有效	4G 字节
	H'38000000 ~ H'3BFFFFFF	无效	
2 个	H'18000000 ~ H'1BFFFFFF	有效	8G 字节
	H'38000000 ~ H'3BFFFFFF	无效	

19.5.3 串行闪存 32 位地址

因为 SPI 多 I/O 总线空间是 64M 字节，所以能直接存取的空间为串行闪存 32 位地址区域的部分区域。此时，32 位地址的高位为寄存器设定的固定值。

为了以 32 位输出串行闪存地址，将 DRENr 寄存器的 ADE[3] 位置“1”，给 DREAR 寄存器的 EAC[2:0] 设定用作串行闪存地址的外部地址范围，并且给 DREAR 寄存器的 EAV[7:0] 位设定用作 32 位地址高位的固定值。

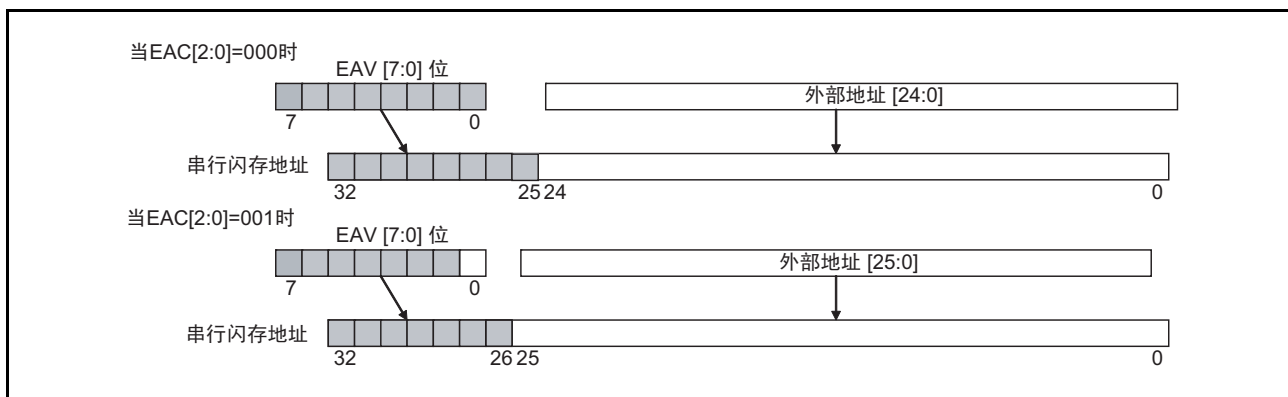


图 19.4 32 位地址的设置

能通过将 DRENr 寄存器的 ADE[3] 位置“1”，以 [31:0] 位输出串行闪存地址。

当将 EAC[2:0] 置“000”时，外部地址 [24:0] 有效。必须给 EAV[7:0] 设定 [32:25] 位。

当将 EAC[2:0] 置“001”时，外部地址 [25:0] 有效。必须给 EAV[7:1] 设定 [32:26] 位。

在实际存取中使用的地址因串行闪存的连接个数而不同。

在连接 1 个串行闪存时使用地址 [31:0]。

在连接 2 个串行闪存时使用地址 [32:1]。

【注】 如果使用未满 4G 字节容量的串行闪存，就必须注意以下事项：

在突发运行（DRCR 寄存器的 RBE 位为“1”）时，如果跨过串行闪存的结束地址进行存取，存取地址和串行闪存的内部地址就会出现差异。

为了防止差异的发生，需要通过软件管理每个所用串行闪存容量能存取的地址区域。

19.5.4 数据调整

如果连接 2 个串行闪存，连接引脚 SPBIO3_0-SPBIO0_0 的串行闪存就为地址 $2n$ ，连接引脚 SPBIO3_1-SPBIO0_1 的串行闪存就为地址 $2n+1$ 。必须至少以字为单位进行存取而不能以字节为单位进行存取。连接 2 个串行闪存时的数据调整如表 19.5 所示。

表 19.5 连接 2 个串行闪存时的数据调整

操作数		串行闪存	
		引脚 SPBIO3_0-SPBIO0_0	引脚 SPBIO3_1-SPBIO0_1
地址 0 的字存取		数据 15 ~ 8	数据 7 ~ 0
地址 2 的字存取		数据 15 ~ 8	数据 7 ~ 0
地址 0 的长字存取	1 个字 (地址 0)	数据 31 ~ 24	数据 23 ~ 16
	2 个字 (地址 2)	数据 15 ~ 8	数据 7 ~ 0
地址 0 的双长字存取	1 个字 (地址 0)	数据 63 ~ 56	数据 55 ~ 48
	2 个字 (地址 2)	数据 47 ~ 40	数据 39 ~ 32
	3 个字 (地址 4)	数据 31 ~ 24	数据 23 ~ 16
	4 个字 (地址 6)	数据 15 ~ 8	数据 7 ~ 0

19.5.5 运行模式

此模块有外部地址空间读模式和 SPI 运行模式共 2 个模式。

外部地址空间读模式将 SPI 多 I/O 总线空间的读操作转换为 SPI 通信并且接收数据，在取到数据后将数据返送给发送源的总线主控。详细内容请参照“19.5.6 外部地址空间读模式”。

SPI 运行模式通过设定寄存器来进行任意的 SPI 通信，详细内容请参照“19.5.8 SPI 运行模式”。

19.5.6 外部地址空间读模式

能通过外部地址空间读模式将 SPI 多 I/O 总线空间的读操作转换为 SPI 通信，还能通过设定寄存器来更改读操作时要发行的命令、可选命令和选项数据。

在外部地址空间读模式中，能选择正常读操作和突发读操作。传送格式取决于公共控制寄存器 (CMNCR)、SSL 延迟寄存器 (SSLDLDR)、位速率设定寄存器 (SPBCR)、数据读控制寄存器 (DRCR)、数据读命令设定寄存器 (DRCMR)、数据读扩展地址设定寄存器 (DREAR)、数据读选项设定寄存器 (DROPR) 和数据读允许设定寄存器 (DRENDR)。

(1) 正常读操作

通过将 DRCR 寄存器的 RBE 位置“0”进行正常读操作。

在正常读操作中，分别以字节、字、长字和双长字读 8 位、16 位、32 位和 64 位数据。但是，只能在连接 1 个串行闪存时进行字节存取。在读数据后，将 SPBSSL 引脚置为无效。

正常读操作的时序图如图 19.5 所示。

t_1 表示从 SPBSSL 引脚有效到 SPBCLK 振荡的期间 (时钟延迟)， t_2 表示从送出传送的最后 SPBCLK 边沿到将 SPBSSL 引脚置为无效的期间 (SPBSSL 无效延迟)， t_3 表示传送结束到开始下次传送的期间 (下次存取)。 t_1 、 t_2 、 t_3 的详细内容请参照“19.5.9 传送格式”。

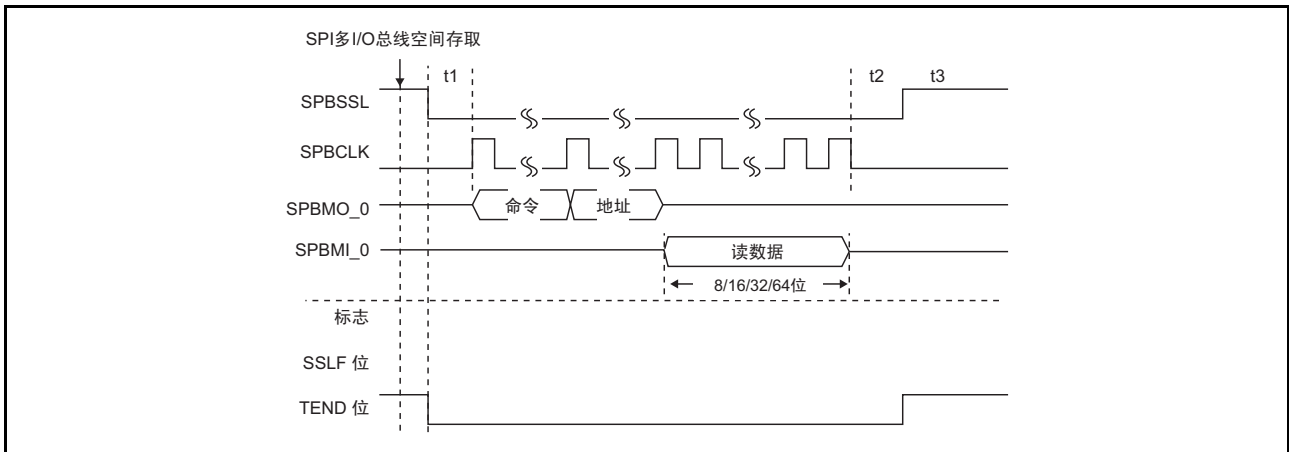


图 19.5 正常读操作时序

(2) 突发读操作

通过将 DRCCR 寄存器的 RBE 位置 “1” 进行突发读操作。

在突发读操作中，读高速缓存有效。有关读高速缓存的运行，请参照“19.5.7 读高速缓存”。

对于以字节、字、长字和双长字为单位进行的读操作，先从读高速缓存参照数据。如果读高速缓存有数据，就从读高速缓存读数据而不存取串行闪存。如果读高速缓存没有数据，就突发读串行闪存并且将读到的数据保存到读高速缓存。此时，数据传送长度为 64 位 \times RBURST[3:0] 位的值，必须从 64 位边界读数据。

能通过 DRCCR 寄存器的 SSLE 位选择数据传送后的 SPBSSL 引脚状态。如果 SSLE 位为 “0”，就在传送数据后将 SPBSSL 引脚置为无效。有关 SSLE 位为 “1” 的运行，请参照“19.5.6(3) 突发读操作 SPBSSL 自动无效”。

此操作的模式图以及 SSLE 位为 “0” 时的突发读操作时序如图 19.6 和图 19.7 所示。

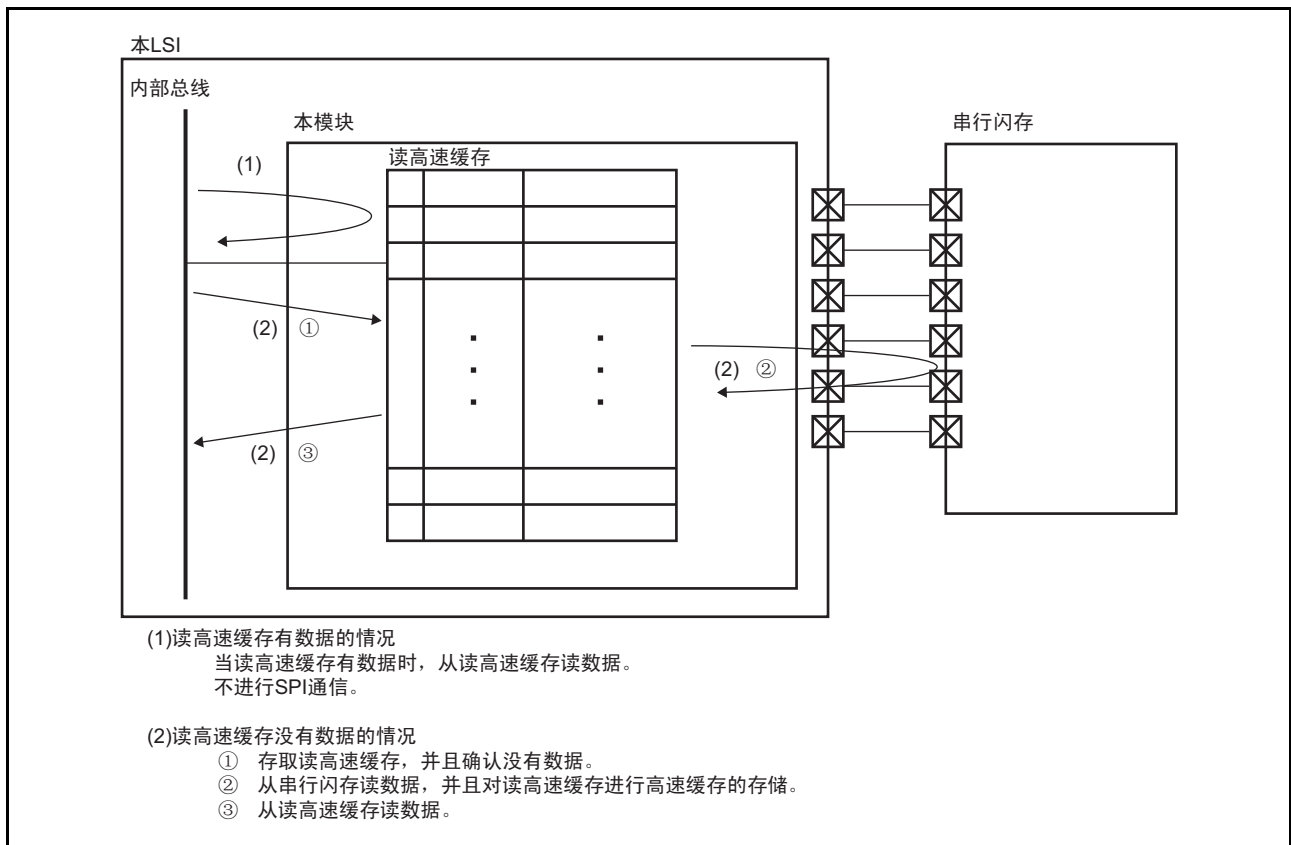


图 19.6 突发读操作

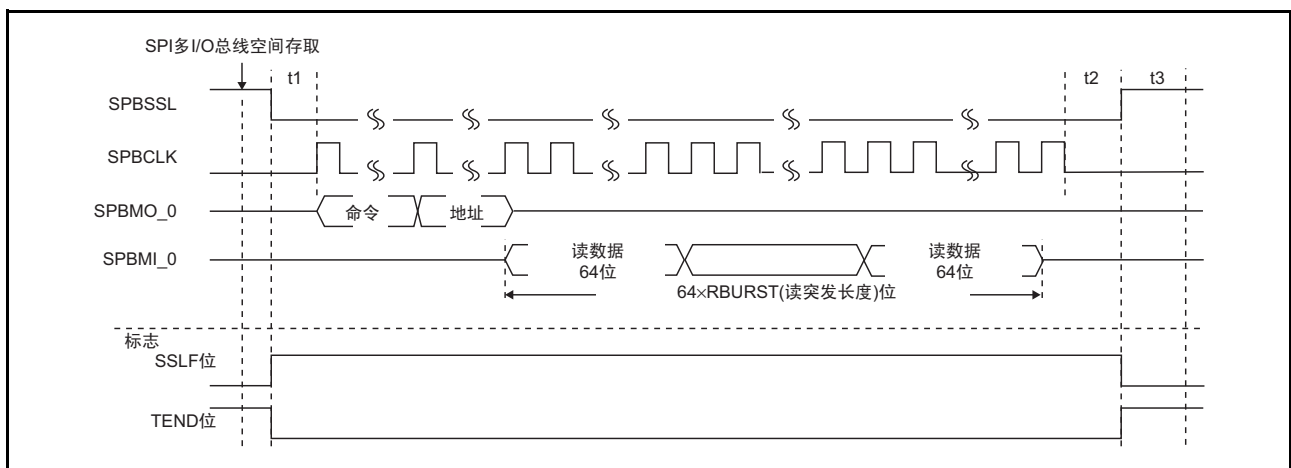


图 19.7 突发读操作时序 (SSLE 位为“0”)

(3) 突发读操作 SPBSSL 自动无效

当将 DRCR 寄存器的 SSLE 位置“1”时，此模块在突发读传送后不将 SPBSSL 引脚置为无效。在下次存取时，如果地址与上次的读地址连续，就不发命令、可选命令、地址和选项数据而进行突发读。如果地址不连续，就暂时将 SPBSSL 引脚置为无效，在发命令、可选命令、地址和选项数据后进行突发读。

连续地址和非连续地址时的突发读时序如图 19.8 和图 19.9 所示。

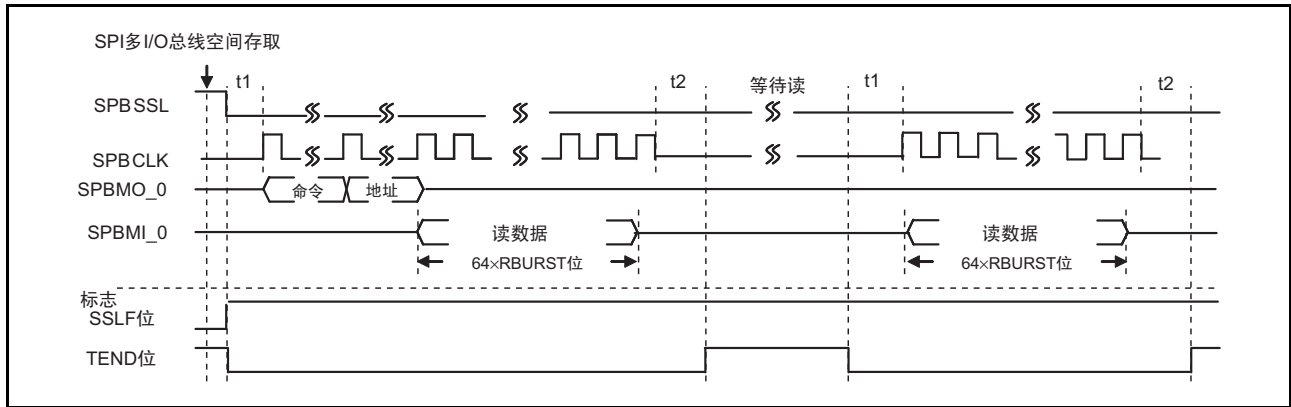


图 19.8 连续地址的突发读操作时序（SSLE 位为“1”）

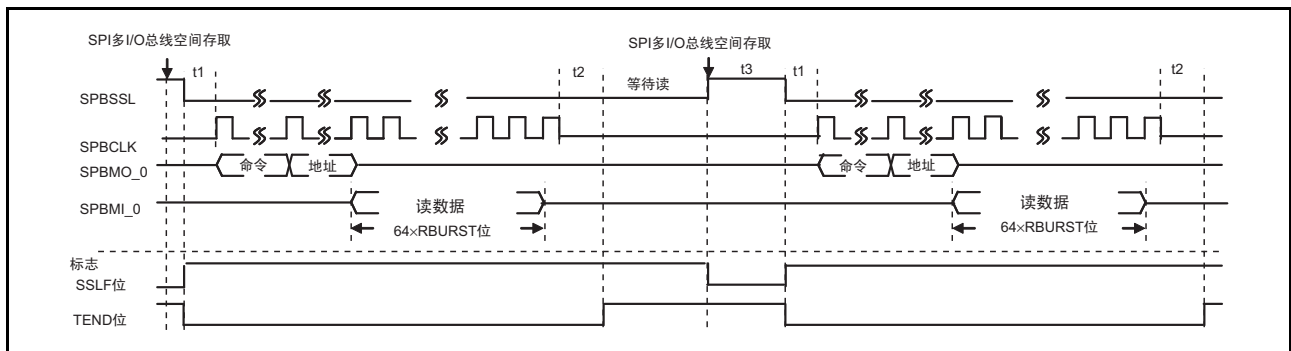


图 19.9 非连续地址的突发读操作时序（SSLE 位为“1”）

在通过将 SPBSSL 置为无效来结束此操作时，必须遵循以下步骤：

1. 将 DRCR 寄存器的 SSLE 位清“0”。
2. 对读高速缓存进行转储清除。
3. 进行（虚拟）读存取。

(4) 初始设定流程

外部地址空间读模式的初始设定流程例子如图 19.10 所示。

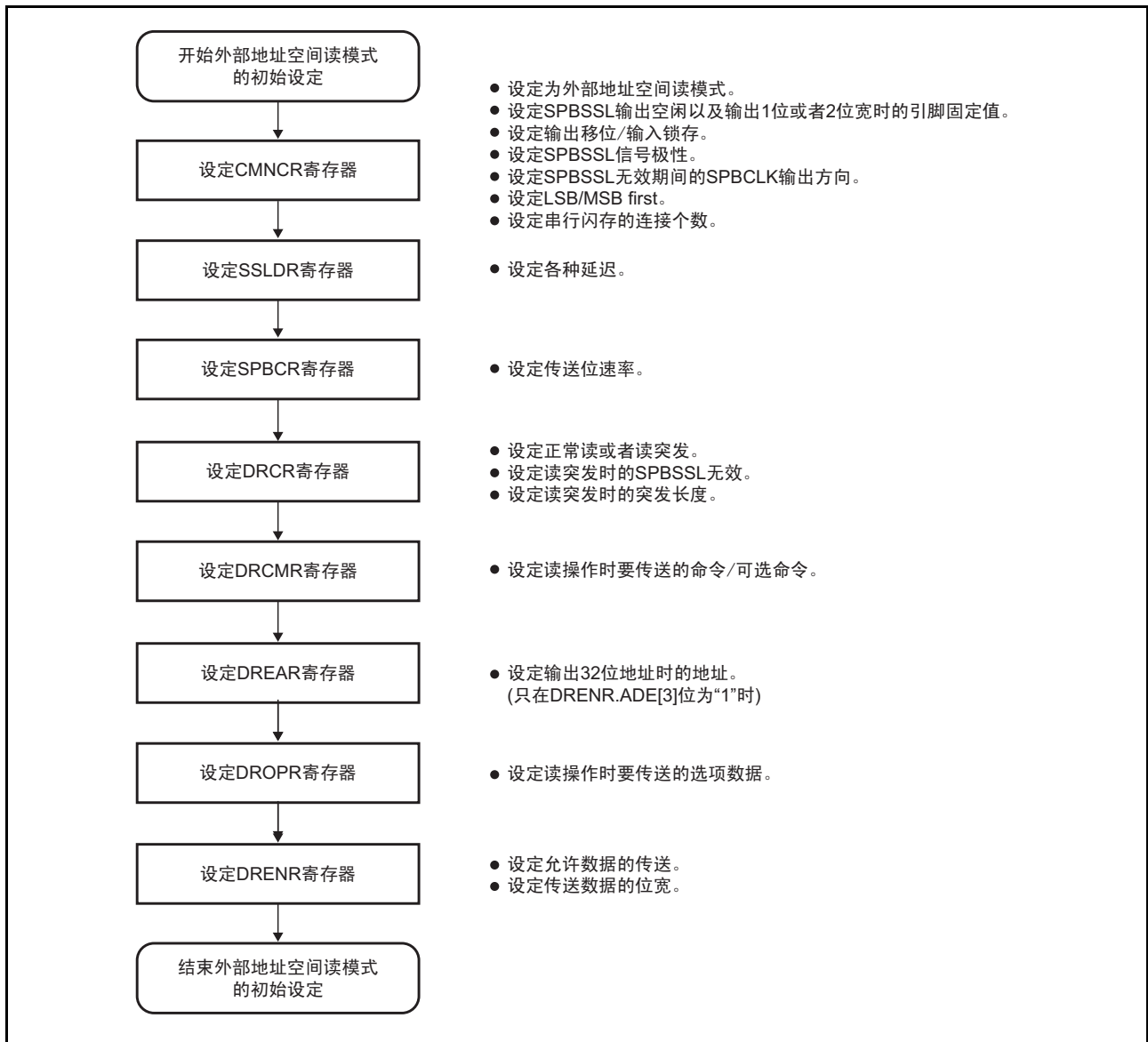


图 19.10 外部地址空间读模式的初始设定流程例子

19.5.7 读高速缓存

此模块内置简易的读高速缓存，能在外部地址空间读模式和突发读操作时使用高速缓存。读高速缓存由 64 位行大小和 16 个入口构成。

读高速缓存的结构如图 19.11 所示。

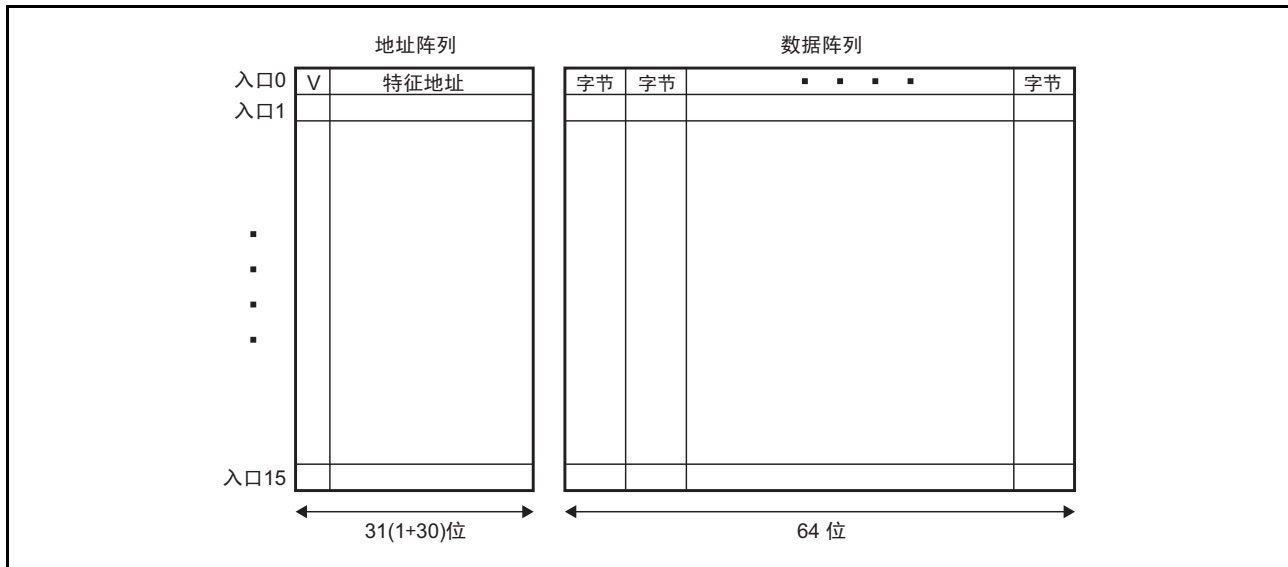


图 19.11 读高速缓存的结构

(1) 地址阵列

V 位表示入口数据是否有效。当 V 位为“1”时，表示数据有效；当 V 位为“0”时，表示数据无效。

特征地址保持串行闪存使用的地址，由地址 32 ~ 3 位构成。在地址输出为 24 位的情况下，当连接 1 个串行闪存时地址 23 ~ 3 有效，当连接 2 个串行闪存时地址 24 ~ 3 有效。

在地址输出为 32 位的情况下，当连接 1 个串行闪存时地址 31 ~ 3 有效，当连接 2 个串行闪存时地址 32 ~ 3 有效。

(2) 数据阵列

保持 64 位读数据。以行为单位注册到读高速缓存。

(3) 读操作

如果读命中，就从读高速缓存读数据；如果读未命中，就从串行闪存读 $64 \times \text{RBURST}$ （读突发长度）的数据，并且在更新读高速缓存后将数据返送给总线主控。

(4) 数据替换

通过写指针来管理数据的更新。在读未命中时，从写指针指向的入口开始替换 RBURST （读突发长度）的数据，即按照保存的旧顺序进行替换。不影响是否参照数据。

19.5.8 SPI 运行模式

此模块能通过设定寄存器进行任意的 SPI 运行。

传送格式取决于公共控制寄存器 (CMNCR)、SSL 延迟寄存器 (SSLDR)、位速率设定寄存器 (SPBCR)、SPI 模式控制寄存器 (SMCR)、SPI 模式命令设定寄存器 (SMCMR)、SPI 模式地址设定寄存器 (SMADR)、SPI 模式选项设定寄存器 (SMOPR)、SPI 模式允许设定寄存器 (SMENR)、SPI 模式读数据寄存器 (SMRDR)、SPI 模式写数据寄存器 (SMWDR)，能用于读写串行闪存的状态。

(1) 传送开始

能通过将 SMCR 寄存器的 SPIE 位置“1”，以设定的传送格式开始传送。在允许写的情况下，将 SPI 模式写数据寄存器的数据发送到串行闪存；在允许读的情况下，将从串行闪存读到的数据保存到 SPI 模式读数据寄存器。

时序图如图 19.12 所示。

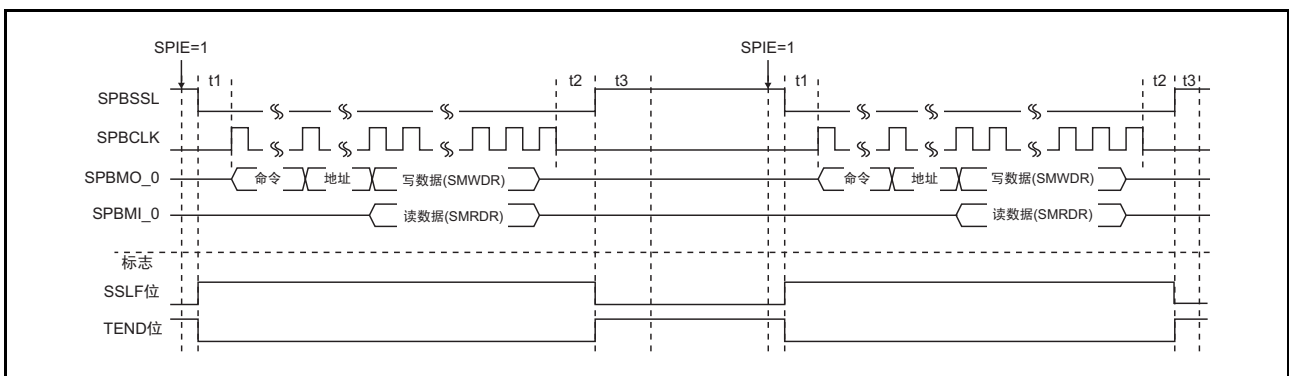


图 19.12 SPI 运行时序图

(2) 读写允许

读：能通过将 SMCR 寄存器的 SPIRE 位置“1”来读数据。将读到的数据保存到 SMRDR 寄存器。

写：能通过将 SMCR 寄存器的 SPIWE 位置“1”来写数据。输出被保存在 SMWDR 寄存器的数据。

在将 SMENR 寄存器的 SPIDB[1:0] 位的位宽设定为 1 位时，能通过将 SPIRE 位和 SPIWE 位置“1”进行发送和接收。如果将 SPIDB[1:0] 位的位宽设定为 2 位或者 4 位，就必须将 SPIRE 位或者 SPIWE 位置为有效。如果将两个位都置为有效，就不保证运行。

(3) SPBSSL 引脚有效的保持

通过将 SMCR 寄存器的 SSLKP 位置“1”，使 SPBSSL 引脚持续有效到下次传送为止。能通过此功能，在 SPBSSL 有效的状态下进行连续传送。

【注】 如果在 SPI 模式中并且传送数据的位宽为 2 位或者 4 位时进行读操作，就不能使用此功能。

使用 SSLKP 位的传送时序如图 19.13 所示。

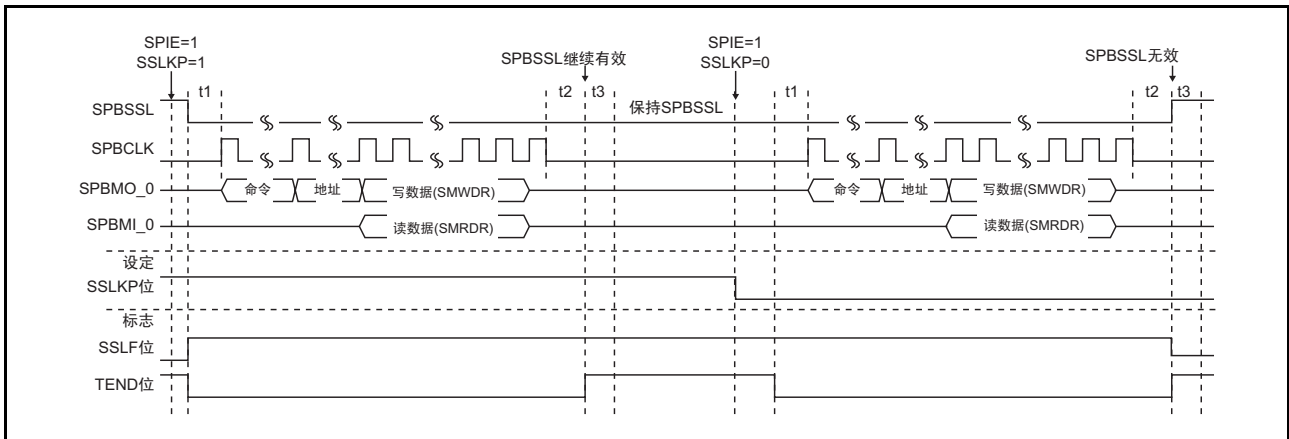


图 19.13 使用 SSLKP 位的传送时序图

(4) 初始设定流程

SPI 运行模式的初始设定流程例子如图 19.14 所示。

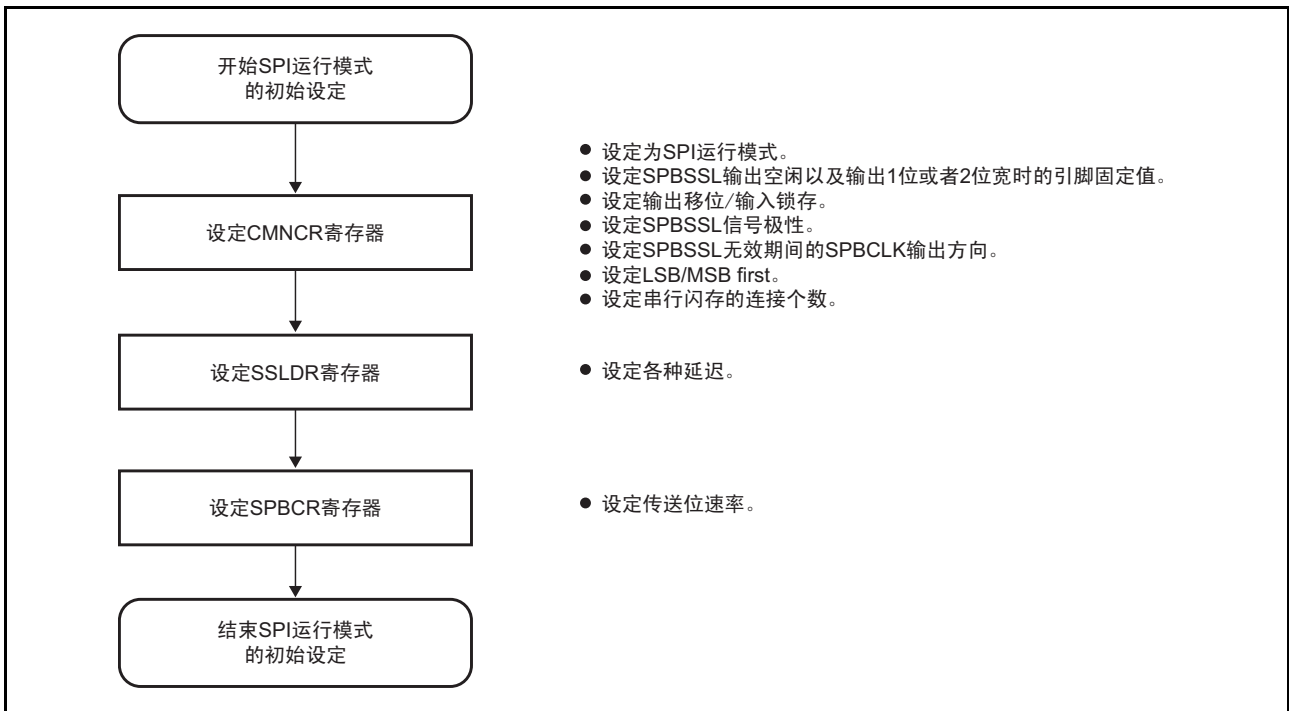


图 19.14 SPI 运行模式的初始设定流程例子

(5) 数据传送设定流程

SPI 运行模式的数据传送设定流程例子如图 19.15 所示。

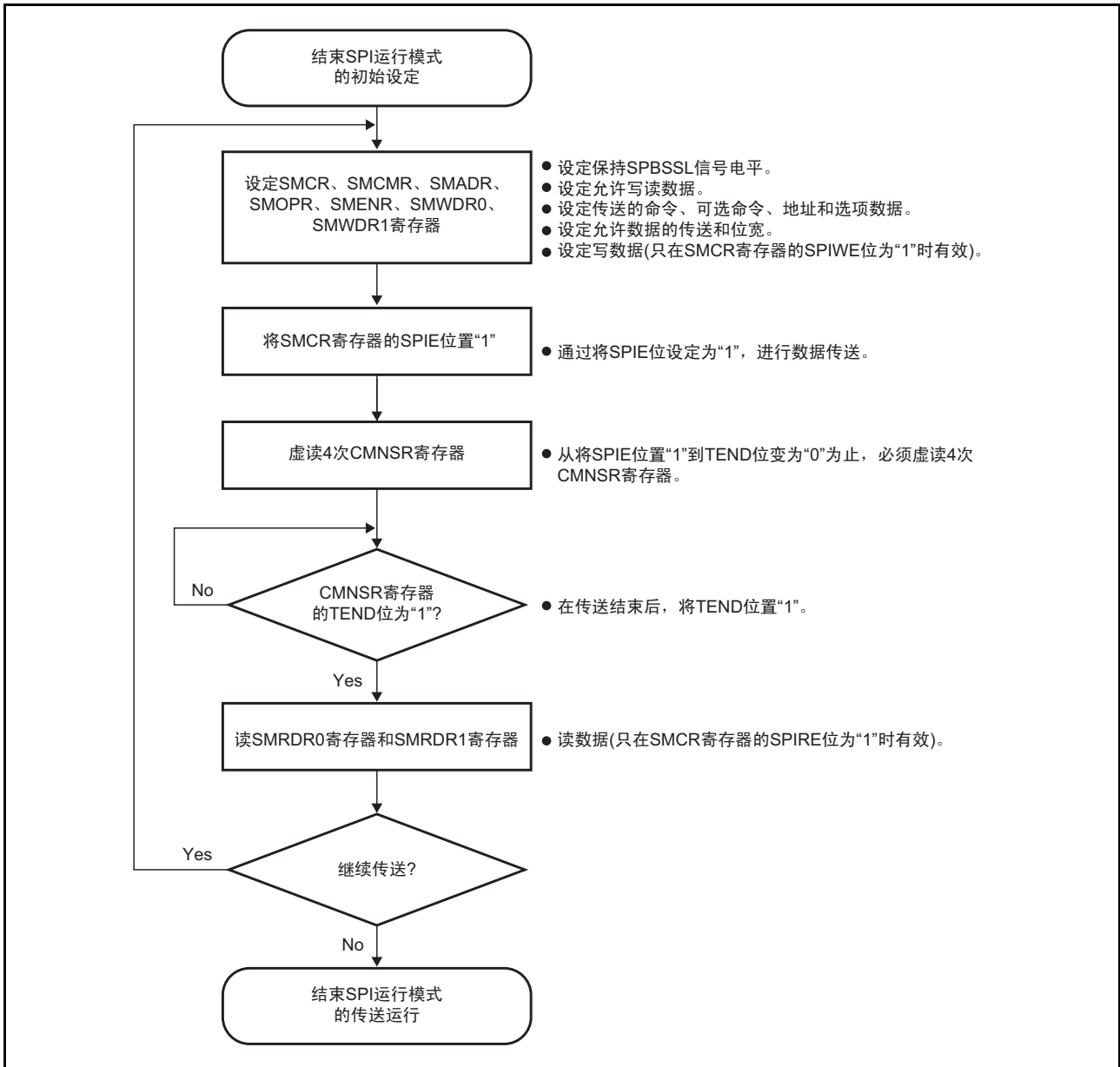


图 19.15 SPI 运行模式的数据传送设定流程例子

19.5.9 传送格式

(1) SPBSSL 引脚的允许极性控制

能通过 CMNCR 寄存器的 SSLP 位更改 SPBSSL 的允许极性。

(2) SPBCLK 输出

能通过 CMNCR 寄存器的 CPOL 位，设定 SPBSSL 无效时的 SPBCLK 输出方向。

(3) 数据发送和接收时序

能通过 CMNCR 寄存器的 CPHAT 位将数据发送时序设定为奇数或者偶数。同样，能通过 CMNCR 寄存器的 CPHAR 位将数据接收时序设定为奇数或者偶数。

(4) 延迟期间

t_1 是从 SPBSSL 引脚有效到 SPBCLK 振荡的期间（时钟延迟），能通过 SSLDR 寄存器的 SCKDL[2:0] 位进行设定。 t_2 是从 SPBCLK 停止振荡到 SPBSSL 信号无效的期间（SPBSSL 无效延迟）， t_3 是在传送结束后为了下次传送而抑制 SPBSSL 信号有效的期间（下次存取延迟）。 t_2 为 1.5 个 SPBCLK 周期， t_3 为 1 个 SPBCLK 周期。

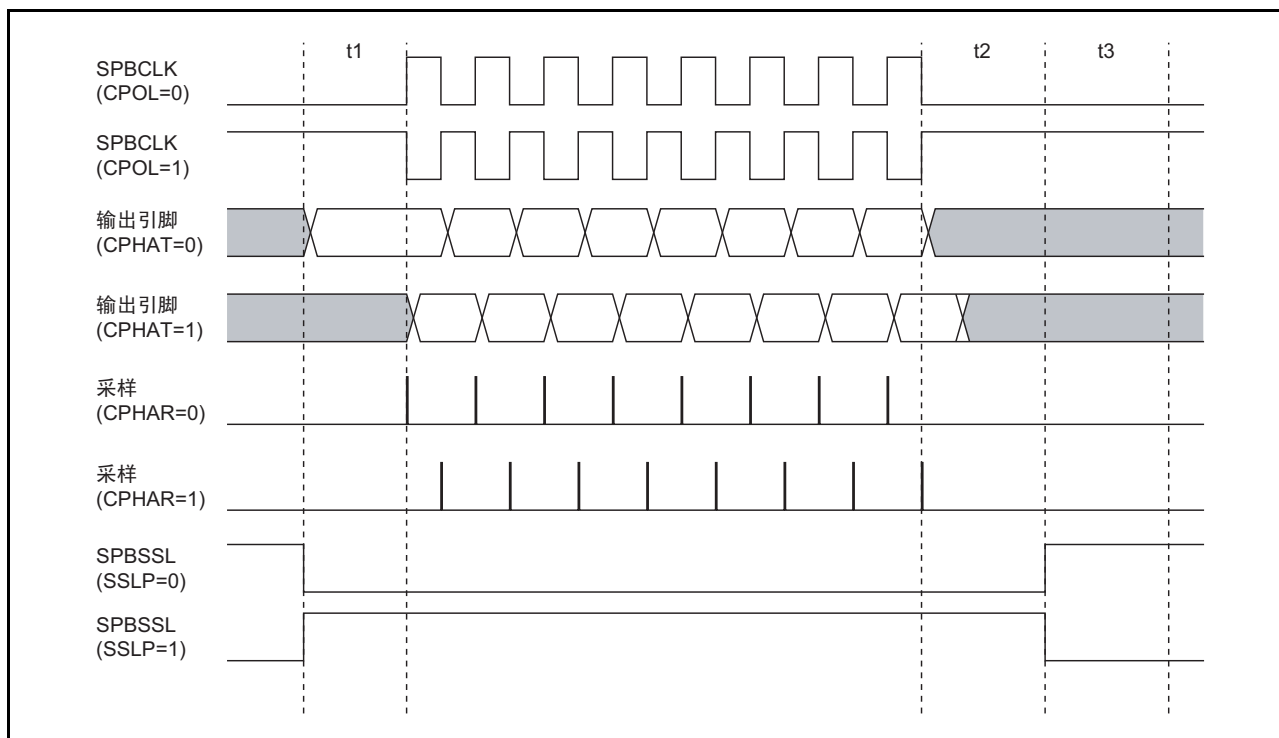


图 19.16 传送格式

19.5.10 数据格式

此模块按命令、可选命令、地址、选项数据、数据的顺序输入或者输出数据。

(1) 数据寄存器

输入 / 输出数据如表 19.6 所示。

表 19.6 数据寄存器

数据		外部地址空间读操作	SPI 运行
命令 (8 位)		DRCMR.CMD[7:0] 位	SMCMR.CMD[7:0] 位
可选命令 (8 位)		DRCMR.OCMD[7:0] 位	SMCMR.OCMD[7:0] 位
地址 (32 位 / 24 位)	BSZ[1:0]=00 (连接 1 个)	32 位: DREAR.EAV[6:1 ~ 0] 位 + 读到的低位地址 [25 ~ 24:0] 位 24 位: 读到的低位地址 [23:0] 位	32 位: SMADR.ADR[31:0] 位 24 位: SMADR.ADR[23:0] 位
	BSZ[1:0]=01 (连接 2 个)	32 位: DREAR.EAV[7:1 ~ 0] 位 + 读到的低位地址 [25 ~ 24:1] 位 24 位: 读到的低位地址 [24:1] 位	
选项数据 (8 位 × 4)		DROPR 寄存器	SMOPR 寄存器
传送数据		正常读: 8/16/32/64 位 突发读: 64×RBURST 位	读: SMRDR0 寄存器和 SMRDR1 寄存器 写: SMWDR0 寄存器和 SMWDR1 寄存器

(2) 数据允许

在读外部地址空间时, 能通过 DRENr 寄存器的 CDE、OCDE、ADE[3:0] 和 OPDE[3:0] 位允许或者禁止命令、可选命令、地址和选项数据的传送。同样, 在 SPI 运行模式中, 能通过 SMENr 寄存器的 CDE、OCDE、ADE[3:0]、OPDE[3:0] 和 SPIDE[3:0] 位允许或者禁止命令、可选命令、地址、选项数据和传送数据。但是, 在 SPI 运行模式中, 不能将全部数据设定为禁止, 必须至少允许 1 个数据。对于读外部地址空间时的地址和选项数据以及 SPI 运行模式中的地址、选项数据和传送数据, 通过传送数据长度决定能设定的允许位。有关能设定的允许位的组合, 请参照各寄存器说明。

如果设定为禁止, 就跳过该数据进行下一个数据的输入和输出。随时输出命令、可选命令、地址和选项数据。在读外部地址空间时, 随时输入数据; 在 SPI 运行模式中, 通过 SMCR 寄存器的 SPIRE 位和 SPIWE 位的设定决定数据的输入和输出。

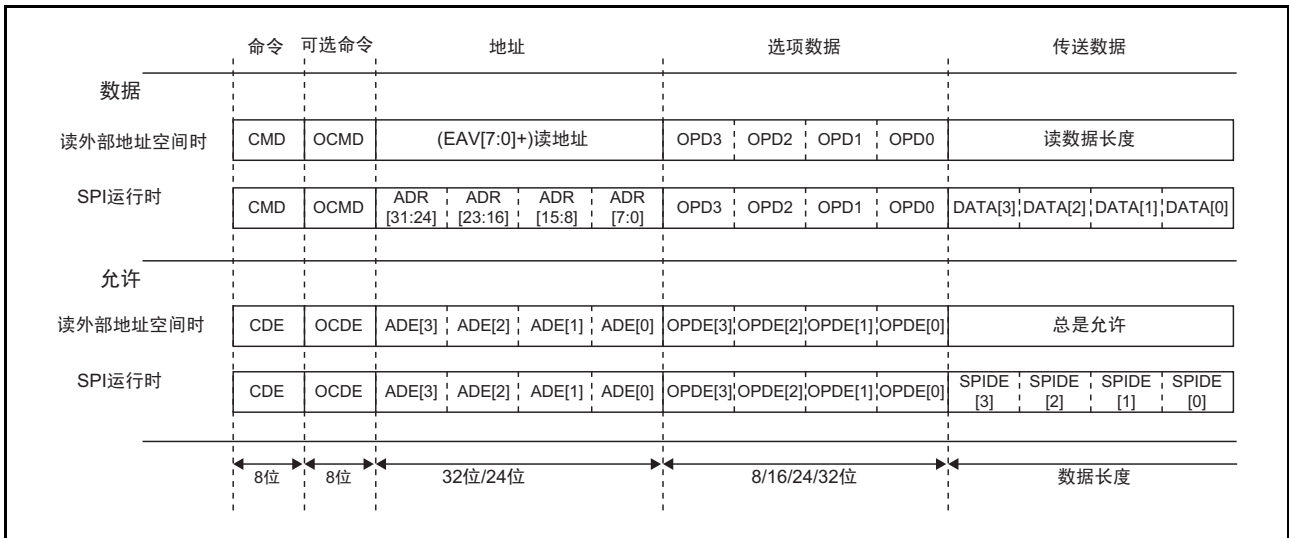


图 19.17 数据和允许

(3) 位宽

在读外部地址空间时，能通过 DRENr 寄存器的 CDB[1:0]、OCDB[1:0]、ADB[1:0]、OPDB[1:0]、DRDB[1:0] 位分别控制命令、可选命令、地址、选项数据和读数据的位宽。

同样，在 SPI 运行模式中，能通过 SMENr 寄存器的 CDB[1:0]、OCDB[1:0]、ADB[1:0]、OPDB[1:0]、SPIDB[1:0] 位控制命令、可选命令、地址、选项数据和读写数据的位宽。

(a) 1 位宽

如果将位宽设定为 1 位，SPBMI_0 引脚和 SPBMI_1 引脚就为输入，SPBMO_0 引脚和 SPBMO_1 引脚就为输出。不使用 SPBIO2_0、SPBIO2_1、SPBIO3_0 和 SPBIO3_1 引脚。

传送格式例子如图 19.18 和图 19.19 所示。

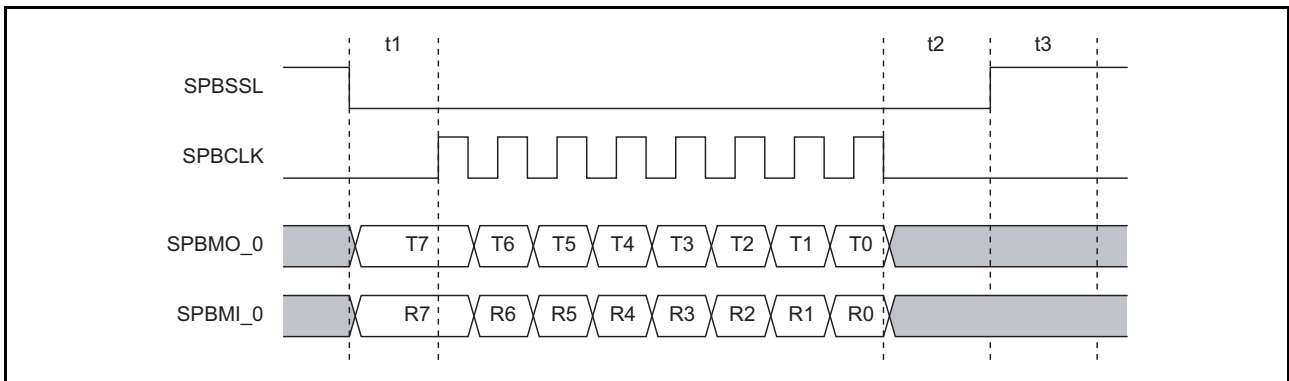


图 19.18 1 位宽并且连接 1 个串行闪存的传送格式例子

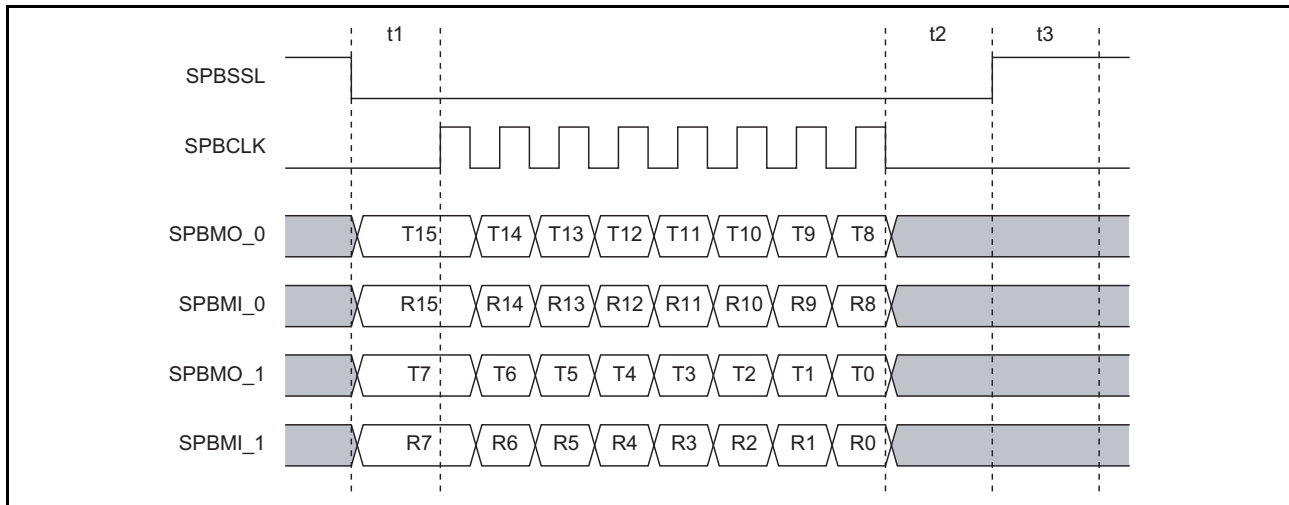


图 19.19 1 位宽并且连接 2 个串行闪存的传送格式例子

(b) 2 位宽

如果将位宽设定为 2 位，SPBIO0_0 和 SPBIO0_1 引脚、SPBIO1_0 和 SPBIO1_1 引脚就为输入或者输出。不使用 SPBIO2_0、SPBIO2_1、SPBIO3_0 和 SPBIO3_1 引脚。

传送格式例子如图 19.20 和图 19.21 所示。

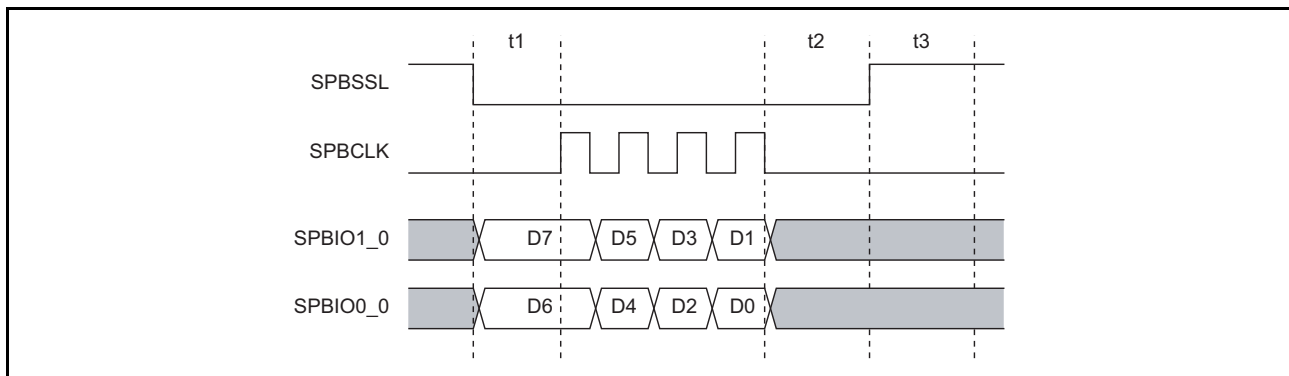


图 19.20 2 位宽并且连接 1 个串行闪存的传送格式例子

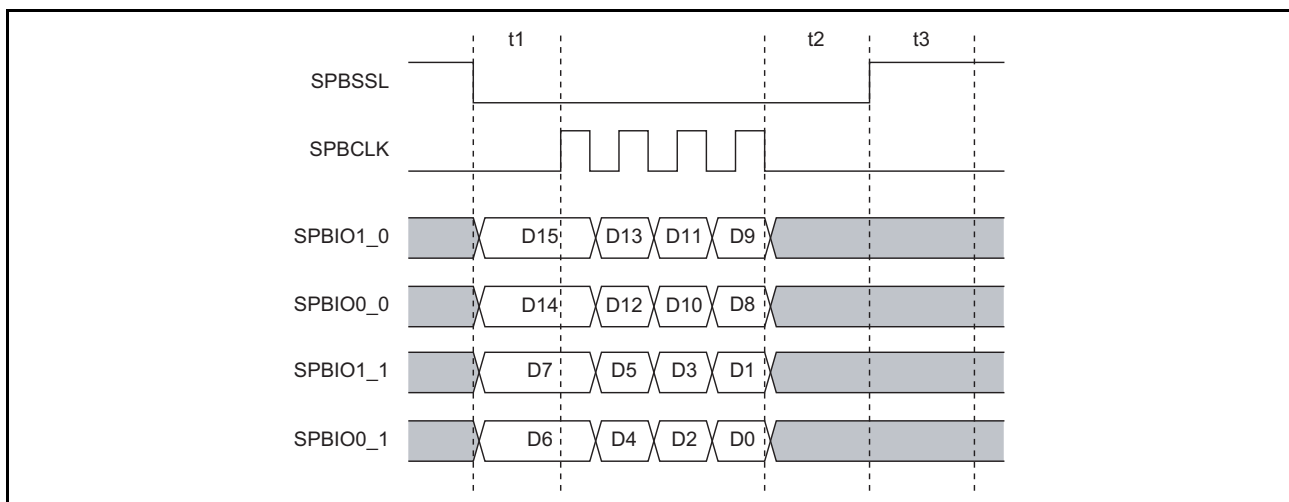


图 19.21 2 位宽并且连接 2 个串行闪存的传送格式例子

(c) 4 位宽

如果将位宽设定为 4 位，SPBIO0_0、SPBIO0_1、SPBIO1_0、SPBIO1_1、SPBIO2_0、SPBIO2_1、SPBIO3_0 和 SPBIO3_1 引脚就为输入或者输出。传送格式例子如图 19.22 和图 19.23 所示。

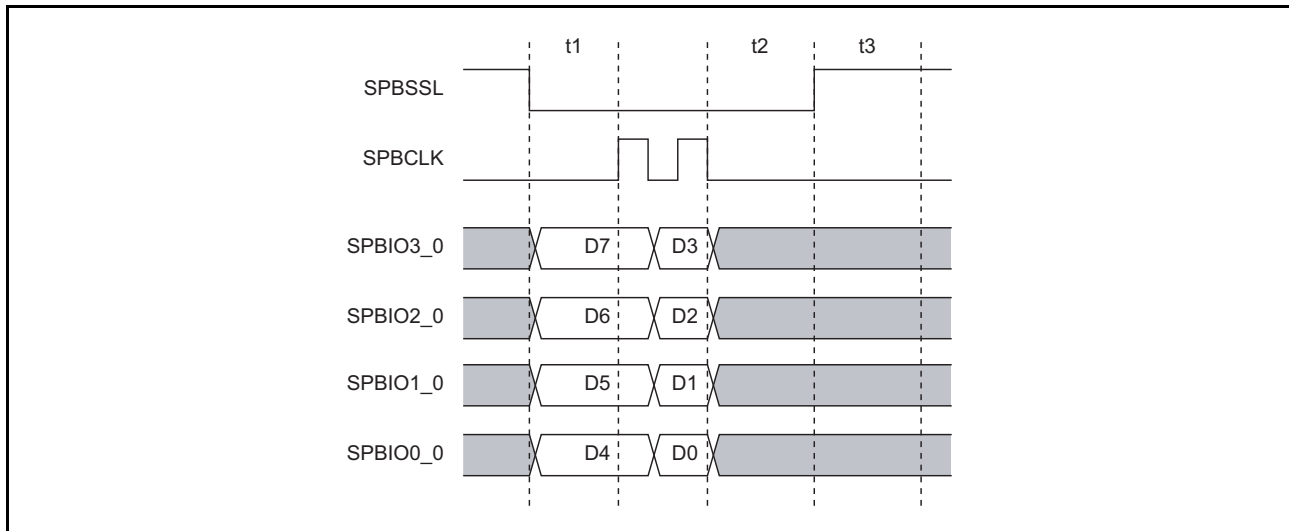


图 19.22 4 位宽并且连接 1 个串行闪存的传送格式例子

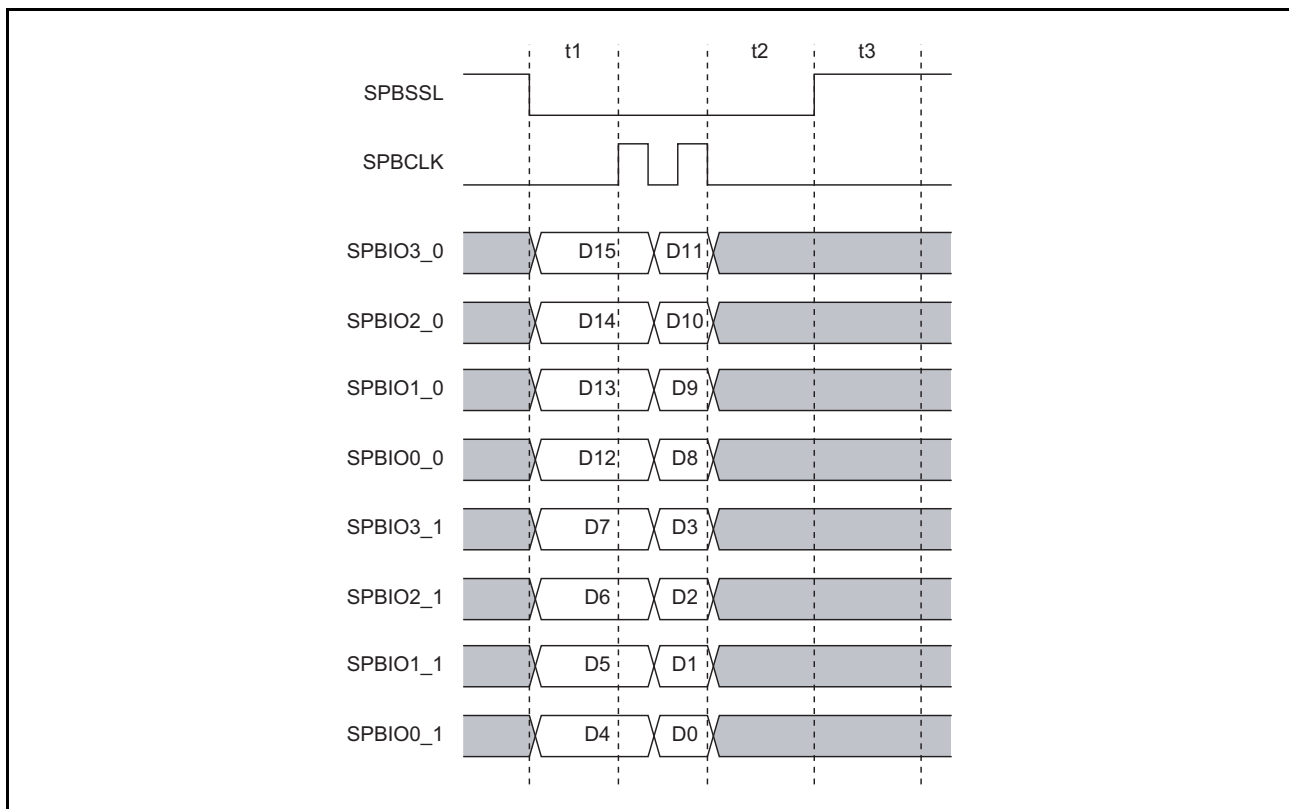


图 19.23 4 位宽并且连接 2 个串行闪存的传送格式例子

19.5.11 数据引脚控制

此模块能通过设定要使用的数据宽度、读操作、写操作来自动切换引脚状态，还能通过 CMNCR 寄存器的 MOII03、MOII02、MOII01、MOII00 位设定 SPBSSL 无效的引脚状态。

SPBSSL 引脚和 SPBCLK 引脚总是为输出引脚，各引脚状态表如表 19.7 ~ 表 19.9 所示。

表 19.7 引脚状态 (1)

引脚	SPBSSL 无效	SPBSSL 有效		
		命令、可选命令、地址、选项数据		
		1 位	2 位	4 位
SPBMO_0/SPBIO0_0、 SPBMO_1/SPBIO0_1	MOII00 位的设定值	输出	输出	输出
SPBMI_0/SPBIO1_0、 SPBMI_1/SPBIO1_1	MOII01 位的设定值	Hi-Z	输出	输出
SPBIO2_0、SPBIO2_1	MOII02 位的设定值	IO2FV 位的设定值	IO2FV 位的设定值	输出
SPBIO3_0、SPBIO3_1	MOII03 位的设定值	IO3FV 位的设定值	IO3FV 位的设定值	输出

表 19.8 引脚状态 (2)

引脚	传送数据					
	读外部地址空间			SPI 运行		
	1 位	2 位	4 位	SPIRE 位 =1、SPIWE 位 =0		
				1 位	2 位	4 位
SPBMO_0/SPBIO0_0、 SPBMO_1/SPBIO0_1	IO0FV 位的 设定值	输入	输入	IO0FV 位 的设定值	输入	输入
SPBMI_0/SPBIO1_0、 SPBMI_1/SPBIO1_1	输入	输入	输入	输入	输入	输入
SPBIO2_0、 SPBIO2_1	MOII02 位 的设定值	MOII02 位 的设定值	输入	MOII02 位 的设定值	MOII02 位 的设定值	输入
SPBIO3_0、 SPBIO3_1	MOII03 位 的设定值	MOII03 位 的设定值	输入	MOII03 位 的设定值	MOII03 位 的设定值	输入

表 19.9 引脚状态 (3)

引脚	传送数据					
	SPI 运行					
	SPIRE 位 =0、SPIWE 位 =1			SPIRE 位 =1、SPIWE 位 =1		
	1 位	2 位	4 位	1 位	2 位	4 位
SPBMO_0/SPBIO0_0、 SPBMO_1/SPBIO0_1	输出	输出	输出	输出	禁止设定	禁止设定
SPBMI_0/SPBIO1_0、 SPBMI_1/SPBIO1_1	Hi-Z	输出	输出	输入	禁止设定	禁止设定
SPBIO2_0、 SPBIO2_1	MOII02 位 的设定值	MOII02 位 的设定值	输出	MOII02 位 的设定值	禁止设定	禁止设定
SPBIO3_0、 SPBIO3_1	MOII03 位 的设定值	MOII03 位 的设定值	输出	MOII03 位 的设定值	禁止设定	禁止设定

19.5.12 SPBSSL 引脚控制

SPBSSL 引脚的无效条件如下所示。

(1) 读外部地址空间

(a) 正常读操作 (DRCR 寄存器的 RBE 位为 “0”)

在数据传送结束并且经过 t_2 周期后, SPBSSL 引脚无效。

(b) 突发读 SPBSSL 不自动无效

(DRCR 寄存器的 RBE 位为 “1”, DRCR 寄存器的 SSLE 位为 “0”)

在数据传送结束并且经过 t_2 周期后, SPBSSL 引脚无效。

(c) 突发读 SPBSSL 自动无效

(DRCR 寄存器的 RBE 位为 “1”, DRCR 寄存器的 SSLE 位为 “1”)

- 当读地址与上次读到的地址不连续时, 经过 t_2 周期后 SPBSSL 引脚无效。

(2) SPI 运行模式

(a) SPBSSL 引脚不保持有效 (SMCR 寄存器的 SSLKP 位为 “0”)

在数据传送结束并且经过 t_2 周期后, SPBSSL 引脚无效。

(b) SPBSSL 引脚保持有效 (SMCR 寄存器的 SSLKP 位为 “1”)

不将 SPBSSL 引脚置为无效。

如果要将 SPBSSL 引脚置为无效, 就必须在将 SSLKP 位置 “0” 后传送数据。

19.5.13 标志

此模块的 CMNSR 寄存器有 SSLF 位和 TEND 位, 这 2 个位是只读位。

(1) SSLF 位

此位表示 SPBSSL 的引脚状态。如果将 SPBSSL 置为有效, 就读到 “1”; 如果置为无效, 就读到 “0”。

(2) TEND 位

此位表示数据正在传送或者数据传送已经结束。

在 t_1 期间表示数据传送, 在 t_2 期间表示数据正在传送, 并且 TEND 位的读取值为 “0”。

在上述以外的情况下, 表示数据传送已经结束并且 TEND 位的读取值为 “1”。

(3) 寄存器的改写

寄存器的改写时序取决于 SSLF 位和 TEND 位的状态, 各种寄存器的改写时序如表 19.10 所示。

表 19.10 各种寄存器的改写时序

标志	能改写的寄存器	备注
SSLF=0	CMNCR、SSLDR、SPBCR、DRCR 寄存器	当 SPBSSL 引脚无效时
TEND=1	DRCMR、DREAR、DROPR、DRENr、SMCR、SMCMR、SMADR、SMOPR、SMENR、SMWDR0、SMWDR1 寄存器	当传送结束时

20. I²C 总线接口 3

I²C 总线接口 3 以飞利浦公司提倡的 I²C 总线（Inter IC Bus）接口方式为基准，具有子集功能。但是，必须注意：控制 I²C 总线的一部分寄存器结构和飞利浦公司不同。

有 4 个通道。

20.1 特点

- 可选择 I²C 总线格式或者时钟同步串行格式。
- 能连续发送和接收。
因为移位寄存器、发送数据寄存器和接收数据寄存器各自独立，所以能连续发送和接收。

I²C 总线格式

- 在主机模式中，自动生成开始条件和停止条件。
- 在接收时，可选择应答的输出电平。
- 在发送时，自动装入应答位。
- 内置位同步功能。
在主机模式中，按位监控 SCL 的状态，自动取得同步。在没有准备好传送时，将 SCL 置为低电平，进入待机状态。
- 中断源：6 种
发送数据空（包含从属地址匹配时）、发送结束、接收数据满（包含从属地址匹配时）、仲裁失败、NACK 检测、停止条件检测
- 能通过发送数据空中断和接收数据满中断，启动直接存储器存取控制器进行数据传送。
- 能直接驱动总线。
在选择总线驱动功能时，SCL 和 SDA 的 2 个引脚为 NMOS 漏极开路输出。

时钟同步串行格式

- 中断源：4 种
发送数据空、发送结束、接收数据满、溢出错误
- 能通过发送数据空中断和接收数据满中断，启动直接存储器存取控制器进行数据传送。

框图如图 20.1 所示。

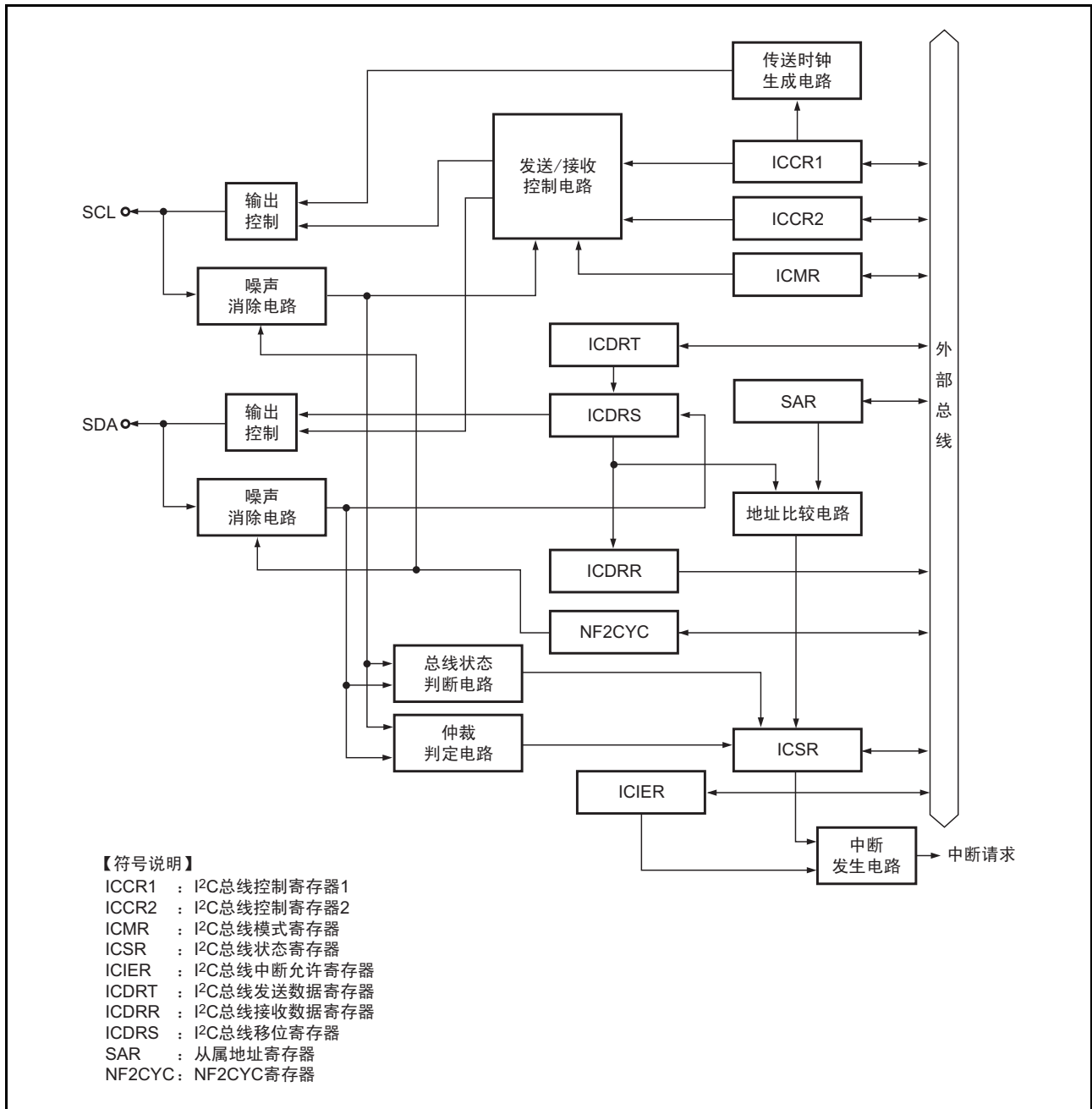


图 20.1 框图

20.2 输入 / 输出引脚

引脚结构如表 20.1 所示。

表 20.1 引脚结构

名称	引脚名	输入 / 输出	功能
串行时钟引脚	SCL0 ~ SCL3	输入 / 输出	I ² C 串行时钟的输入 / 输出引脚
串行数据引脚	SDA0 ~ SDA3	输入 / 输出	I ² C 串行数据的输入 / 输出引脚

输入 / 输出引脚的外部电路连接例子如图 20.2 所示。

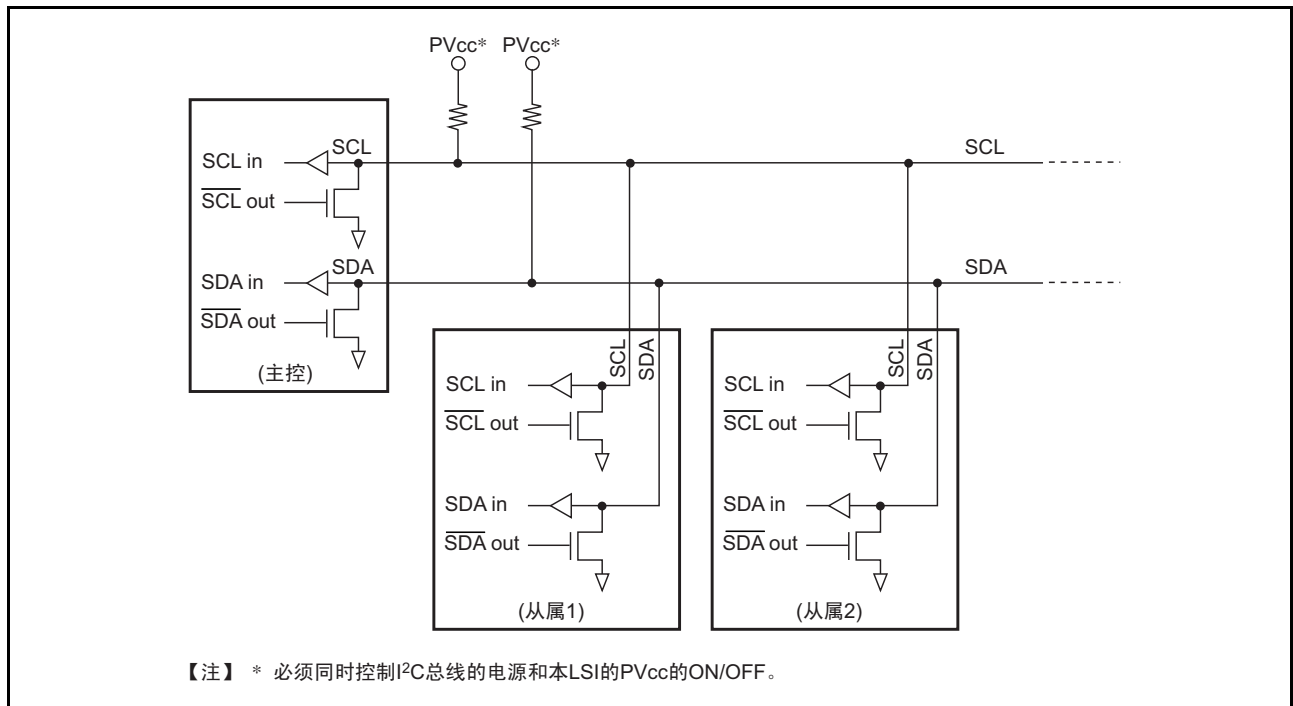


图 20.2 输入 / 输出引脚的外部电路连接例子

20.3 寄存器说明

寄存器结构如表 20.2 所示。

表 20.2 寄存器结构

通道	寄存器名	略称	R/W	初始值	地址	存取长度
0	I ² C 总线控制寄存器 1	ICCR1_0	R/W	H'00	H'FFFEE000	8
	I ² C 总线控制寄存器 2	ICCR2_0	R/W	H'7D	H'FFFEE001	8
	I ² C 总线模式寄存器	ICMR_0	R/W	H'38	H'FFFEE002	8
	I ² C 总线中断允许寄存器	ICIER_0	R/W	H'00	H'FFFEE003	8
	I ² C 总线状态寄存器	ICSR_0	R/W	H'00	H'FFFEE004	8
	从属地址寄存器	SAR_0	R/W	H'00	H'FFFEE005	8
	I ² C 总线发送数据寄存器	ICDRT_0	R/W	H'FF	H'FFFEE006	8
	I ² C 总线接收数据寄存器	ICDRR_0	R/W	H'FF	H'FFFEE007	8
	NF2CYC 寄存器	NF2CYC_0	R/W	H'00	H'FFFEE008	8
1	I ² C 总线控制寄存器 1	ICCR1_1	R/W	H'00	H'FFFEE400	8
	I ² C 总线控制寄存器 2	ICCR2_1	R/W	H'7D	H'FFFEE401	8
	I ² C 总线模式寄存器	ICMR_1	R/W	H'38	H'FFFEE402	8
	I ² C 总线中断允许寄存器	ICIER_1	R/W	H'00	H'FFFEE403	8
	I ² C 总线状态寄存器	ICSR_1	R/W	H'00	H'FFFEE404	8
	从属地址寄存器	SAR_1	R/W	H'00	H'FFFEE405	8
	I ² C 总线发送数据寄存器	ICDRT_1	R/W	H'FF	H'FFFEE406	8
	I ² C 总线接收数据寄存器	ICDRR_1	R/W	H'FF	H'FFFEE407	8
	NF2CYC 寄存器	NF2CYC_1	R/W	H'00	H'FFFEE408	8
2	I ² C 总线控制寄存器 1	ICCR1_2	R/W	H'00	H'FFFEE800	8
	I ² C 总线控制寄存器 2	ICCR2_2	R/W	H'7D	H'FFFEE801	8
	I ² C 总线模式寄存器	ICMR_2	R/W	H'38	H'FFFEE802	8
	I ² C 总线中断允许寄存器	ICIER_2	R/W	H'00	H'FFFEE803	8
	I ² C 总线状态寄存器	ICSR_2	R/W	H'00	H'FFFEE804	8
	从属地址寄存器	SAR_2	R/W	H'00	H'FFFEE805	8
	I ² C 总线发送数据寄存器	ICDRT_2	R/W	H'FF	H'FFFEE806	8
	I ² C 总线接收数据寄存器	ICDRR_2	R/W	H'FF	H'FFFEE807	8
	NF2CYC 寄存器	NF2CYC_2	R/W	H'00	H'FFFEE808	8
3	I ² C 总线控制寄存器 1	ICCR1_3	R/W	H'00	H'FFFEEC00	8
	I ² C 总线控制寄存器 2	ICCR2_3	R/W	H'7D	H'FFFEEC01	8
	I ² C 总线模式寄存器	ICMR_3	R/W	H'38	H'FFFEEC02	8
	I ² C 总线中断允许寄存器	ICIER_3	R/W	H'00	H'FFFEEC03	8
	I ² C 总线状态寄存器	ICSR_3	R/W	H'00	H'FFFEEC04	8
	从属地址寄存器	SAR_3	R/W	H'00	H'FFFEEC05	8
	I ² C 总线发送数据寄存器	ICDRT_3	R/W	H'FF	H'FFFEEC06	8
	I ² C 总线接收数据寄存器	ICDRR_3	R/W	H'FF	H'FFFEEC07	8
	NF2CYC 寄存器	NF2CYC_3	R/W	H'00	H'FFFEEC08	8

20.3.1 I²C 总线控制寄存器 1 (ICCR1)

ICCR1 是 8 位可读写寄存器，选择 I²C 总线接口 3 的运行 / 停止、发送 / 接收控制、主控模式 / 从属模式、发送 / 接收以及主控模式传送时钟频率。

位:	7	6	5	4	3	2	1	0
	ICE	RCVD	MST	TRS	CKS[3:0]			
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	ICE	0	R/W	I ² C 总线接口 3 允许 0: 禁止 SCL/SDA 输出 (SCL/SDA 输入有效) 1: 此模块为可传送状态 (SCL/SDA 为总线驱动状态)
6	RCVD	0	R/W	接收禁止 设定在 TRS 位为“0”的状态下读 ICDRR 时是继续还是禁止下一次运行。 0: 继续下一次接收 1: 禁止下一次接收
5 4	MST TRS	0 0	R/W R/W	主控 / 从属选择 发送 / 接收选择 在 I ² C 总线格式的主控模式中，如果总线竞争失败，就通过硬件对 MST 位和 TRS 位进行复位，变为从属接收模式。必须在传送帧的期间更改 TRS 位。另外，在从属接收模式中，如果开始条件后的 7 位和 SAR 设定的从属地址相同并且第 8 位是“1”，TRS 位就自动置“1”。在时钟同步串行格式的主控接收模式中，如果发生溢出错误，就将 MST 位清“0”，变为从属接收模式。 根据 MST 位和 TRS 位的组合，变为以下运行模式。如果选择时钟同步串行格式，就在 MST 位为“1”时变为时钟输出。 00: 从属接收模式 01: 从属发送模式 10: 主控接收模式 11: 主控发送模式
3 ~ 0	CKS[3:0]	0000	R/W	传送时钟选择 在 主控模式中，必须根据需要的传送率 (参照表 20.3) 进行设定。

表 20.3 传送率

NF2CYC	ICCR1				时钟	传送率 (kHz)		
	bit4	bit3	bit2	bit1		bit0	P ϕ 25.00MHz	P ϕ 33.33MHz
CKS4	CKS[3]	CKS[2]	CKS[1]	CKS[0]				
0	0	0	0	0	0	P0 ϕ /44	568.18	757.50
				1	0	P0 ϕ /52	480.77	640.96
			1	0	0	P0 ϕ /64	390.63	520.78
				1	0	P0 ϕ /72	347.22	462.92
		1	0	0	0	P0 ϕ /84	297.62	396.79
				1	0	P0 ϕ /92	271.74	362.28
			1	0	0	P0 ϕ /100	250.00	333.30
				1	0	P0 ϕ /108	231.48	308.61
	1	0	0	0	0	P0 ϕ /176	142.05	189.38
				1	0	P0 ϕ /208	120.19	160.24
			1	0	0	P0 ϕ /256	97.66	130.20
				1	0	P0 ϕ /288	86.81	115.73
		1	0	0	0	P0 ϕ /336	74.40	99.20
				1	0	P0 ϕ /368	67.93	90.57
			1	0	0	P0 ϕ /400	62.50	83.33
				1	0	P0 ϕ /432	57.87	77.15
1	0	0	0	0	0	P0 ϕ /352	71.02	94.69
				1	0	P0 ϕ /416	60.10	80.12
			1	0	0	P0 ϕ /512	48.83	65.10
				1	0	P0 ϕ /576	43.40	57.86
		1	0	0	0	P0 ϕ /672	37.20	49.60
				1	0	P0 ϕ /736	33.97	45.29
			1	0	0	P0 ϕ /800	31.25	41.66
				1	0	P0 ϕ /864	28.94	38.58
	1	0	0	0	0	P0 ϕ /704	35.51	47.34
				1	0	P0 ϕ /832	30.05	40.06
			1	0	0	P0 ϕ /1024	24.41	32.55
				1	0	P0 ϕ /1152	21.70	28.93
		1	0	0	0	P0 ϕ /1344	18.60	24.80
				1	0	P0 ϕ /1472	16.98	22.64
			1	0	0	P0 ϕ /1600	15.63	20.83
				1	0	P0 ϕ /1728	14.47	19.29

【注】 设定的内容必须满足外部规格。

20.3.2 I²C 总线控制寄存器 2 (ICCR2)

ICCR2 为 8 位可读写寄存器，控制开始 / 停止条件的发行、SDA 引脚的操作、SCL 引脚的监控以及 I²C 总线控制部的复位。

位:	7	6	5	4	3	2	1	0
	BBSY	SCP	SDAO	SDAOP	SCLO	—	IICRST	—
初始值:	0	1	1	1	1	1	0	1
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R

位	位名	初始值	R/W	说明
7	BBSY	0	R/W	总线忙 有表示 I ² C 总线占有 / 释放状态的标志功能和主控模式的开始 / 停止条件的发行功能。在时钟同步串行格式时，此位的读取值总是“0”。在 I ² C 总线格式时并且在 SCL 为 High 电平的状态下，如果 SDA 从 High 电平变为 Low 电平，就认为发行了开始条件并且将此位置“1”。如果在 SCL 为 High 电平的状态下 SDA 从 Low 电平变为 High 电平，就认为发行了停止条件并且将此位清“0”。在发行开始条件时，给 BBSY 位写“1”并且给 SCP 位写“0”，在重新发行开始条件时也同样；在发行停止条件时，给 BBSY 位和 SCP 位写“0”。
6	SCP	1	R/W	开始 / 停止条件发行禁止 在主控模式中，SCP 位控制开始条件 / 停止条件的发行。在发行开始条件时，给 BBSY 位写“1”并且给 SCP 位写“0”，在重新发行开始条件时也同样；在发行停止条件时，给 BBSY 位和 SCP 位写“0”。此位的读取值总是“1”。即使写“1”，也不保存数据。
5	SDAO	1	R/W	SDA 输出值控制 要更改 SDA 的输出电平时，SDAO 位和 bit4 的 SDAOP 位配合使用。不能在传送过程中操作此位。 0: 在读操作时，SDA 引脚输出 Low 电平。 在写操作时，将 SDA 引脚输出更改为 Low 电平。 1: 在读操作时，SDA 引脚输出 High 电平。 在写操作时，将 SDA 引脚输出更改为 Hi-Z（通过外部上拉电阻，输出 High 电平）
4	SDAOP	1	R/W	SDAO 写保护 通过改写 SDAOP 位来控制 SDA 引脚输出电平的更改。要更改输出电平时，给 SDAOP 位和 SDAOP 位写“0”，或者给 SDAOP 位写“1”并且给 SDAOP 位写“0”。此位的读取值总是“1”。
3	SCLO	1	R	SCL 输出电平 SCLO 位监控 SCL 的输出电平。在读操作时，如果 SCLO 位是“1”，SCL 引脚就输出 High 电平；如果 SCLO 位是“0”，SCL 引脚就输出 Low 电平。
2	—	1	R	保留位 读写值都为“1”。
1	IICRST	0	R/W	控制部复位 IICRST 位对 ICMR 的 BC[2:0] 位和内部电路进行复位。在 I ² C 总线运行中由于通信不正常等引起中止时，如果将 IICRST 位置“1”，就能对 ICMR 的 BC[2:0] 位和内部电路进行复位。
0	—	1	R	保留位 读写值都为“1”。

20.3.3 I²C 总线模式寄存器 (ICMR)

ICMR 为 8 位可读写寄存器，选择 MSB first/LSB first 和传送位数。

通过 ICCR2 的 IICRST 位，将 BC[2:0] 位初始化为 “H'0”。

位:	7	6	5	4	3	2	1	0
	MLS	—	—	—	BCWP	BC[2:0]		
初始值:	0	0	1	1	1	0	0	0
R/W:	R/W	R	R	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明																		
7	MLS	0	R/W	MSB first/LSB first 选择 0: MSB first 1: LSB first 当用于 I ² C 总线格式时，必须将此位置 “0”。																		
6	—	0	R	保留位 读写值都为 “0”。																		
5、4	—	全 1	R	保留位 读写值都为 “1”。																		
3	BCWP	1	R/W	BC 写保护 控制 BC[2:0] 位的写操作。在改写 BC[2:0] 位时，必须将此位置 “0”。在时钟同步串行格式时，不能改写 BC[2:0] 位。 0: 在写操作时，设定 BC[2:0] 位的值。 1: 在读操作时，读取值总是 “1”。 在写操作时，BC[2:0] 位的设定值无效。																		
2 ~ 0	BC[2:0]	000	R/W	<p>位计数器 指定下一次要传送的数据位数。如果读这些位，就能得知剩余的传送位数。在 I²C 总线格式时，给数据加上 1 个应答位后进行传送。必须在传送帧的期间设定这些位。如果设定 “B'000” 以外的值，就必须在 SCL 为 Low 电平的状态下进行设定。在包含应答的数据传送结束后，这些位自动恢复到 “B'000”。在检测到停止条件后，自动变为 “B'111”。通过将上电复位、软件待机模式、模块待机模式以及将 ICCR2 的 IICRST 位置 “1” 来清除这些位。如果选择时钟同步串行格式，就不能改写这些位。</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 50%; border: none;">I²C 总线格式</td> <td style="width: 50%; border: none;">时钟同步串行格式</td> </tr> <tr> <td style="border: none;">000: 9 位</td> <td style="border: none;">000: 8 位</td> </tr> <tr> <td style="border: none;">001: 2 位</td> <td style="border: none;">001: 1 位</td> </tr> <tr> <td style="border: none;">010: 3 位</td> <td style="border: none;">010: 2 位</td> </tr> <tr> <td style="border: none;">011: 4 位</td> <td style="border: none;">011: 3 位</td> </tr> <tr> <td style="border: none;">100: 5 位</td> <td style="border: none;">100: 4 位</td> </tr> <tr> <td style="border: none;">101: 6 位</td> <td style="border: none;">101: 5 位</td> </tr> <tr> <td style="border: none;">110: 7 位</td> <td style="border: none;">110: 6 位</td> </tr> <tr> <td style="border: none;">111: 8 位</td> <td style="border: none;">111: 7 位</td> </tr> </table>	I ² C 总线格式	时钟同步串行格式	000: 9 位	000: 8 位	001: 2 位	001: 1 位	010: 3 位	010: 2 位	011: 4 位	011: 3 位	100: 5 位	100: 4 位	101: 6 位	101: 5 位	110: 7 位	110: 6 位	111: 8 位	111: 7 位
I ² C 总线格式	时钟同步串行格式																					
000: 9 位	000: 8 位																					
001: 2 位	001: 1 位																					
010: 3 位	010: 2 位																					
011: 4 位	011: 3 位																					
100: 5 位	100: 4 位																					
101: 6 位	101: 5 位																					
110: 7 位	110: 6 位																					
111: 8 位	111: 7 位																					

20.3.4 I²C 总线中断允许寄存器 (ICIER)

ICIER 是 8 位可读写寄存器，允许各种中断源、选择应答的有效 / 无效、设定发送应答以及确认接收应答。

位:	7	6	5	4	3	2	1	0
	TIE	TEIE	RIE	NAKIE	STIE	ACKE	ACKBR	ACKBT
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W

位	位名	初始值	R/W	说明
7	TIE	0	R/W	发送中断允许 当 ICSR 的 TDRE 位被置位时，允许或者禁止发送数据空中断 (TXI)。 0: 禁止发送数据空中断请求 (TXI) 1: 允许发送数据空中断请求 (TXI)
6	TEIE	0	R/W	发送结束中断允许 在 ICSR 的 TDRE 位为“1”的状态下第 9 个时钟上升时，TEIE 位允许或者禁止发送结束中断 (TEI)。能通过将 TEND 位清“0”或者将 TEIE 位清“0”来解除 TEI。 0: 禁止发送结束中断请求 (TEI) 1: 允许发送结束中断请求 (TEI)
5	RIE	0	R/W	接收中断允许 当接收数据从 ICDRS 传送到 ICDRR 并且 ICSR 的 RDRF 位被置“1”时，RIE 允许或者禁止接收数据满中断请求 (RXI)。能通过将 RDRF 位清“0”或者将 RIE 清位“0”来解除 RXI。 0: 禁止接收数据满中断请求 (RXI) 1: 允许接收数据满中断请求 (RXI)
4	NAKIE	0	R/W	NACK 接收中断允许 当 ICSR 的 NACKF 位或者 AL/OVE 位被置位时，NAKIE 位允许或者禁止 NACK 检测仲裁失败 / 溢出错误中断请求 (NAKI)。能通过将 NACKF 位或者 AL/OVE 位清“0”或者将 NAKIE 位清“0”来解除 NAKI。 0: 禁止 NACK 接收中断请求 (NAKI) 1: 允许 NACK 接收中断请求 (NAKI)
3	STIE	0	R/W	停止条件检测中断允许 当 ICSR 的 STOP 位被置位时，STIE 允许或者禁止停止条件检测中断请求 (STPI)。 0: 禁止停止条件检测中断请求 (STPI) 1: 允许停止条件检测中断请求 (STPI)
2	ACKE	0	R/W	应答位判定选择 0: 忽视接收应答的内容而进行连续传送 1: 当接收应答为“1”时中止传送
1	ACKBR	0	R	接收应答 在发送模式中，此位保存从接收设备接收到的应答位内容。此位的写操作无效。如果将 ICCR2 的 BBSY 位置“1”，就清除此位。 0: 接收应答为“0” 1: 接收应答为“1”
0	ACKBT	0	R/W	发送应答 在接收模式中，设定在应答时要发送的位。 0: 在应答时发送“0” 1: 在应答时发送“1”

20.3.5 I²C 总线状态寄存器 (ICSR)

ICSR 是 8 位可读写寄存器，确认各中断请求标志和状态。

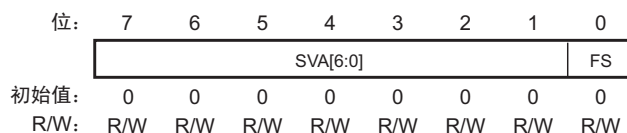
位:	7	6	5	4	3	2	1	0
	TDRE	TEND	RDRF	NACKF	STOP	AL/OVE	AAS	ADZ
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	TDRE	0	R/W	发送数据空 [清除条件] • 在读“1”后写“0”时 • 当给 ICDRT 写数据时 [置位条件] • 当将数据从 ICDRT 传送到 ICDRS 并且 ICDRT 变为空时 • 当将 TRS 位置位时 • 当发行开始条件（包含重新发行）时 • 在从属模式中从接收模式变为发送模式时
6	TEND	0	R/W	发送结束 [清除条件] • 在读“1”后写“0”时 • 当给 ICDRT 写数据时 [置位条件] • I ² C 总线格式的情况：在 TDRE 位为“1”的状态下 SCL 的第 9 个时钟上升时 • 时钟同步串行格式的情况：当发出发送帧的最后位时
5	RDRF	0	R/W	接收数据满 [清除条件] • 在读“1”后写“0”时 • 当读 ICDRR 时 [置位条件] • 当将接收数据从 ICDRS 传送到 ICDRR 时
4	NACKF	0	R/W	无应答检测标志 [清除条件] • 在读“1”后写“0”时 [置位条件] • 在 ICIEP 的 ACKE 位为“1”的状态下发送时，接收设备没有应答。
3	STOP	0	R/W	停止条件检测标志 [清除条件] • 在读“1”后写“0”时 [置位条件] • 在结束帧传送后检测到停止条件时

位	位名	初始值	R/W	说明
2	AL/OVE	0	R/W	<p>仲裁失败标志 / 溢出错误标志</p> <p>在 I²C 总线格式情况下，AL/OVE 位表示在主机模式中总线竞争失败；在时钟同步串行格式的情况下，AL/OVE 位表示在 RDRF 位为“1”的状态下接收到最后位。</p> <p>在多个主机几乎同时要占有总线时，如果 I²C 总线接口 3 监控到 SDA 的数据和自己发送的数据不同，就将 AL 标志置“1”，表示总线已被其它主机占有。</p> <p>[清除条件]</p> <ul style="list-style-type: none"> 在读“1”后写“0”时 <p>[置位条件]</p> <ul style="list-style-type: none"> 主机发送模式的情况：在 SCL 的上升沿内部 SDA 和 SDA 引脚的电平不相同 主机模式的情况：当检测到开始条件并且 SDA 引脚为 High 电平时 时钟同步串行格式的情况：在 RDRF 位为“1”的状态下接收到最后位时
1	AAS	0	R/W	<p>从属地址识别标志</p> <p>在从属接收模式中，如果在开始条件后的第一帧和 SAR 的 SVA6 ~ SVA0 相同时，就将此标志置位。</p> <p>[清除条件]</p> <ul style="list-style-type: none"> 在读“1”后写“0”时 <p>[置位条件]</p> <ul style="list-style-type: none"> 在从属接收模式中检测到从属地址时 在从属接收模式中检测到一般调用地址时
0	ADZ	0	R/W	<p>一般调用地址识别标志</p> <p>在 I²C 总线格式的从属接收模式中有效</p> <p>[清除条件]</p> <ul style="list-style-type: none"> 在读“1”后写“0”时 <p>[置位条件]</p> <ul style="list-style-type: none"> 在从属接收模式中检测到一般调用地址时

20.3.6 从属地址寄存器 (SAR)

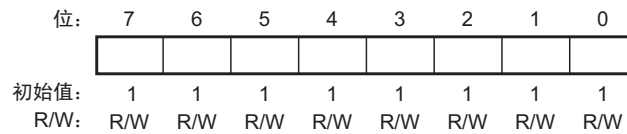
SAR 是 8 位可读写寄存器，选择格式并且设定从属地址。在 I²C 总线格式的从属模式中，如果在开始条件后送来的第 1 帧的高 7 位和 SAR 的高 7 位相同，就作为从属设备运行。



位	位名	初始值	R/W	说明
7 ~ 1	SVA[6:0]	0000000	R/W	<p>从属地址</p> <p>设定与连接到 I²C 总线的其它从属设备不同的唯一地址。</p>
0	FS	0	R/W	<p>格式选择</p> <p>0: 选择 I²C 总线格式</p> <p>1: 选择时钟同步串行格式</p>

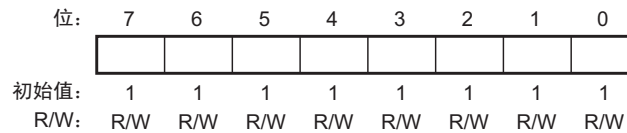
20.3.7 I²C 总线发送数据寄存器 (ICDRT)

ICDRT 是保存发送数据的 8 位可读写寄存器。如果检测到移位寄存器 (ICDRS) 为空, 就将写到 ICDRT 的发送数据传送到 ICDRS, 开始发送数据。在 ICDRS 的数据发送过程中, 如果将下一个发送数据预先写到 ICDRT, 就能连续发送。



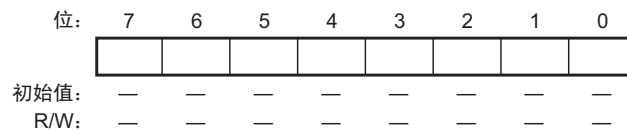
20.3.8 I²C 总线接收数据寄存器 (ICDRR)

ICDRR 是保存接收数据的 8 位寄存器。如果结束 1 字节的数据接收, 就将接收到的数据从 ICDRS 传送到 ICDRR, 变为能接收下一个数据的状态。因为 ICDRR 是接收专用寄存器, 所以 CPU 不能写此寄存器。



20.3.9 I²C 总线移位寄存器 (ICDRS)

ICDRS 是用于发送或者接收数据的寄存器。在发送时, 将发送数据从 ICDRT 传送到 ICDRS, 然后从 SDA 引脚发送数据。在接收时, 如果结束 1 字节的数据接收, 就将数据从 ICDRS 传送到 ICDRR。CPU 不能直接读此寄存器。



20.3.10 NF2CYC 寄存器 (NF2CYC)

NF2CYC 是 8 位可读写寄存器，能选择传送时钟、SCL 引脚和 SDA 引脚的噪声消除范围。详细内容请参照“20.4.7 噪声消除电路”。

位:	7	6	5	4	3	2	1	0
	—	—	—	CKS4	—	—	PRS	NF2 CYC
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	CKS4	0	R/W	传送时钟选择 在主控模式中，必须根据需要的传送率（参照表 20.3）进行设定。 在 1MB 版中，此位是保留位，读写值都为“0”。
3、2	—	全 0	R	保留位 读写值都为“0”。
1	PRS	0	R/W	脉宽比率选择 指定 SCL 的 High 电平宽度和 Low 电平宽度的比率 0: High:Low=0.5:0.5 1: High:Low≈0.4:0.6
0	NF2CYC	0	R/W	噪声消除范围选择 0: 能除去 1 个外围时钟周期以内的噪声 1: 能除去 2 个外围时钟周期以内的噪声

20.4 运行说明

能通过设定 SAR 的 FS 位，I²C 总线接口 3 以 I²C 总线格式或者时钟同步串行格式进行通信。

20.4.1 I²C 总线格式

I²C 总线格式和 I²C 总线时序分别如图 20.3 和图 20.4 所示。开始条件后的第 1 帧一定由 8 位构成。

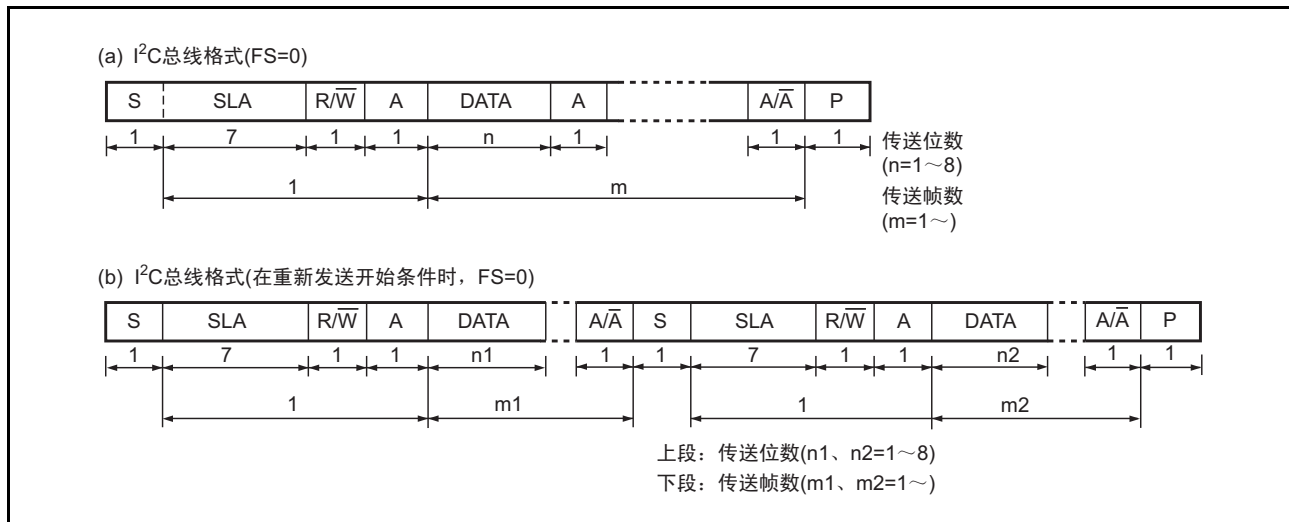


图 20.3 I²C 总线格式

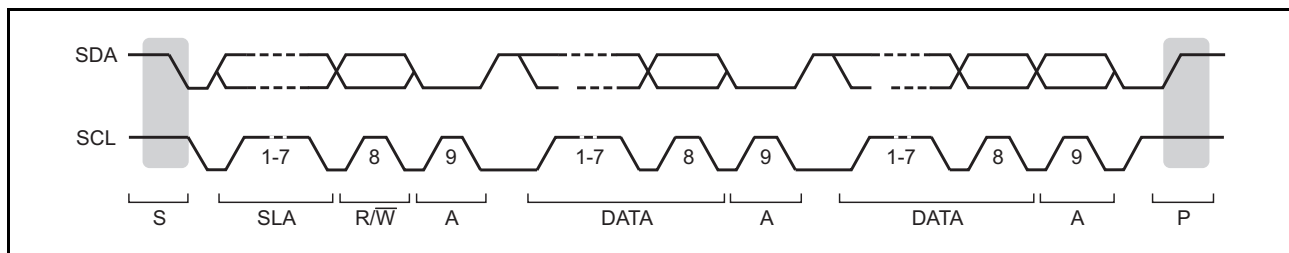


图 20.4 I²C 总线时序

【符号说明】

- S : 开始条件。在 SCL 为 High 电平的状态下，主控设备将 SDA 从 High 电平变为 Low 电平。
- SLA : 从属地址
- R/W : 表示发送和接收的方向。当 R/W 为“1”时，将数据从从属设备发送到主控设备；当 R/W 为“0”时，将数据从主控设备发送到从属设备。
- A : 应答。接收设备将 SDA 置为 Low 电平。
- DATA : 发送和接收数据
- P : 停止条件。在 SCL 为 High 电平的状态下，主控设备将 SDA 从 Low 电平变为 High 电平。

20.4.2 主控发送

在主控发送模式中，主控设备输出发送时钟和发送数据，从属设备返回应答。有关主控发送模式的运行顺序，请参照图 20.5 和图 20.6。主控发送模式的发送步骤和运行如下所示：

1. 将ICCR1的ICE位置“1”，并且设定ICCR1的CKS[3:0]位等（初始设定）。
2. 读ICCR2的BBSY标志，在确认总线处于释放状态后将ICCR1的MST位和TRS位设定为主控发送模式，然后给BBSY位写“1”并且给SCP位写“0”（发送开始条件），生成开始条件。
3. 在确认ICSR的TDRE位被置位后将发送数据（第1个字节表示从属地址和R/W的数据）写到ICDRT。此时，TDRE位自动清“0”，并且在将数据从ICDRT传送到ICDRS后重新将TDRE位置位。
4. 在TDRE位被置位的状态下结束1字节的数据发送，并且在发送时钟的第9个时钟的上升沿将ICSR的TEND位置位。读ICIER的ACKBR位，在确认选择了从属设备后将第2个字节的数据写到ICDRT。因为在ACKBR位为“1”时无法确认从属设备，所以发行停止条件。通过给BBSY位写“0”和给SCP位写“0”，发行停止条件。另外，在完成数据准备前或者在发行停止条件前，将SCL固定为Low电平。
5. 每当TDRE位被置位时，就将第2个字节以后的发送数据写到ICDRT。
6. 如果将要发送的字节数写到ICDRT，就在TDRE位被置位的状态下等待TEND位的置位（最后字节发送结束），或者在ICIER的ACKE位被置位的状态下等待接收设备的NACK（ICSR的NACKF位为“1”）。然后，发行停止条件，清除TEND位或者NACKF位。
7. 如果ICSR的STOP位被置位，就返回到从属接收模式。

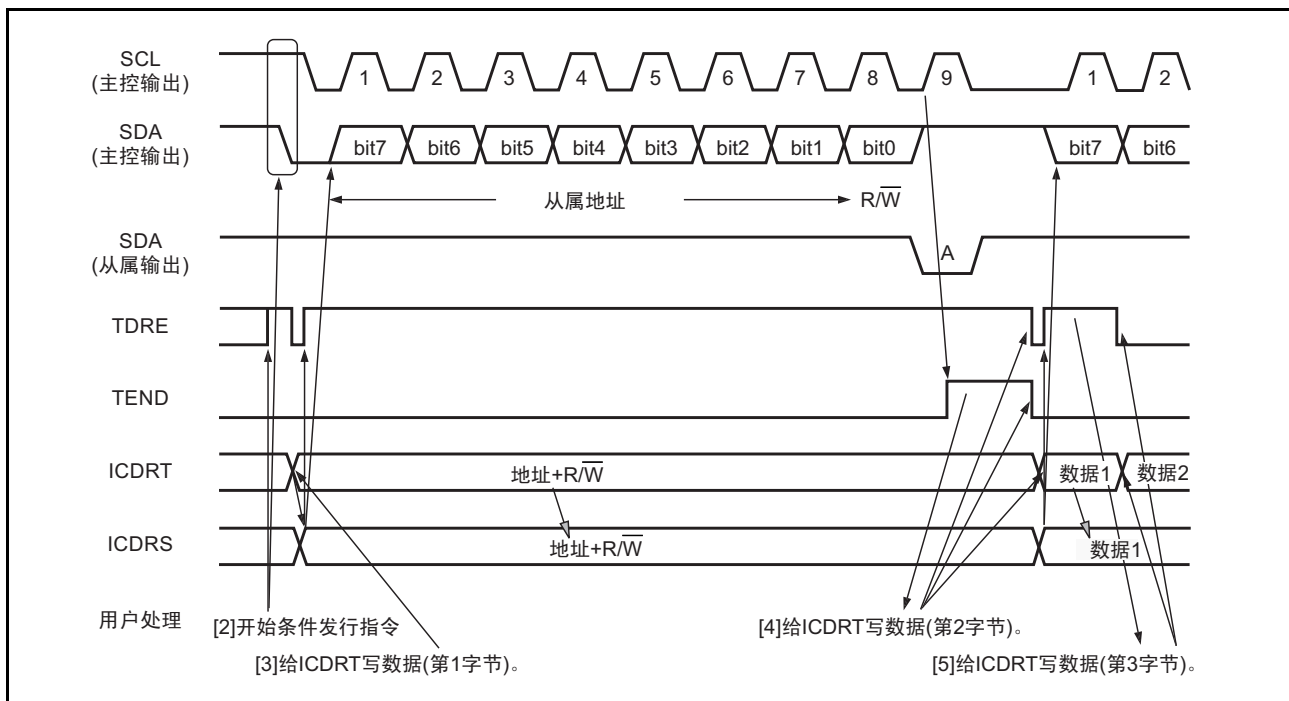


图 20.5 主控发送模式的运行时序（1）

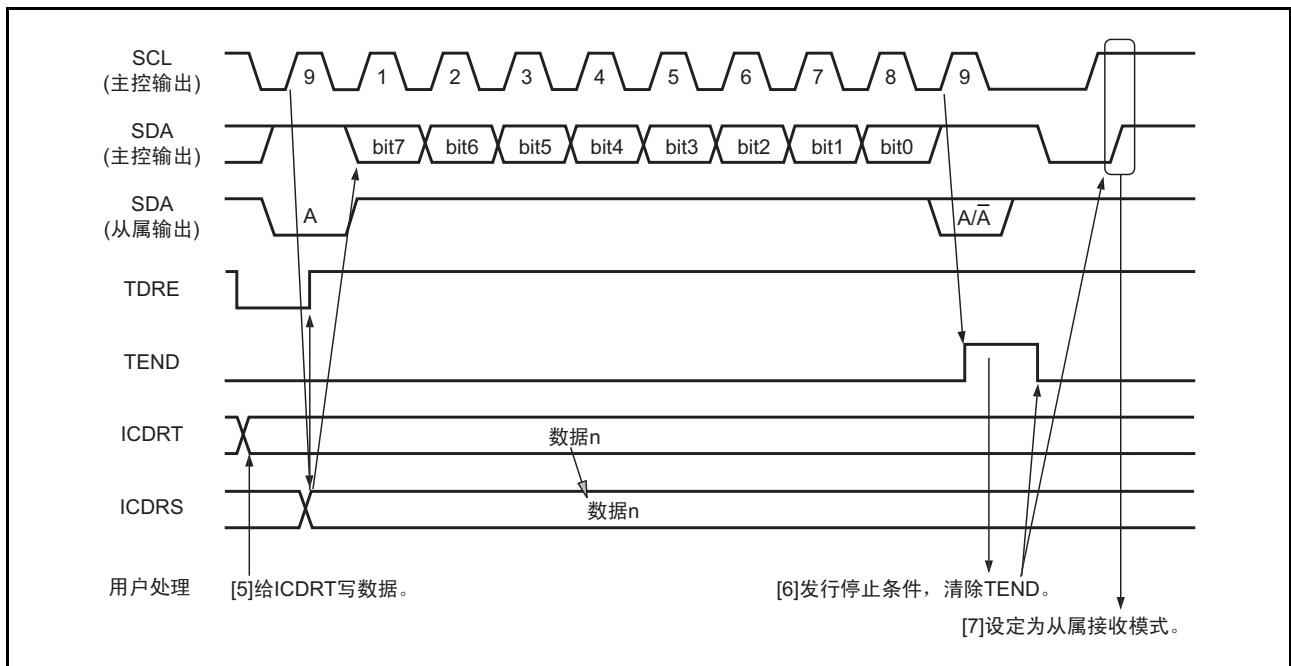


图 20.6 主控发送模式的运行时序 (2)

20.4.3 主控接收

在主控接收模式中，主控设备输出接收时钟，接收从属设备的数据并且返回应答。有关主控接收模式的运行时序，请参照图 20.7 和图 20.8。

主控接收模式的接收步骤和运行如下所示：

1. 在清除 ICSR 的 TEND 位后清除 ICCR1 的 TRS 位，从主控发送模式切换为主控接收模式，然后清除 TDRE 位。
2. 如果虚读 ICDRR，就开始接收*，与内部时钟同步输出接收时钟并且接收数据。在接收时钟的第 9 个时钟，主控设备将 ICIEP 的 ACKBT 位设定的电平输出到 SDA。
3. 结束 1 帧数据的接收，在接收时钟的第 9 个时钟的上升沿将 ICSR 的 RDRF 位置位。此时，能通过读 ICDRR 来读取接收到的数据，同时清除 RDRF 位。
4. 每当 RDRF 位被置位时，都能通过读 ICDRR 进行连续接收。在 RDRF 位被置位的状态下，如果因其它处理而推迟 ICDRR 的读操作并且第 8 个时钟下降，就在读 ICDRR 前将 SCL 固定为 Low 电平。
5. 当下一次接收为最后帧时，就在读 ICDRR 前将 ICCR1 的 RCVD 位置位，从而能在下一次接收后变为可发行停止条件的状态。
6. 如果在接收时钟的第 9 个时钟的上升沿将 RDRF 位置位，就发行停止条件。
7. 如果 ICSR 的 STOP 位被置位，就读 ICDRR，然后清除 RCVD 位。
8. 返回到从属接收模式。

【注】 * 如果只需要接收 1 字节，就必须在将 ICCR1 的 RCVD 位置位后虚读 ICDRR。

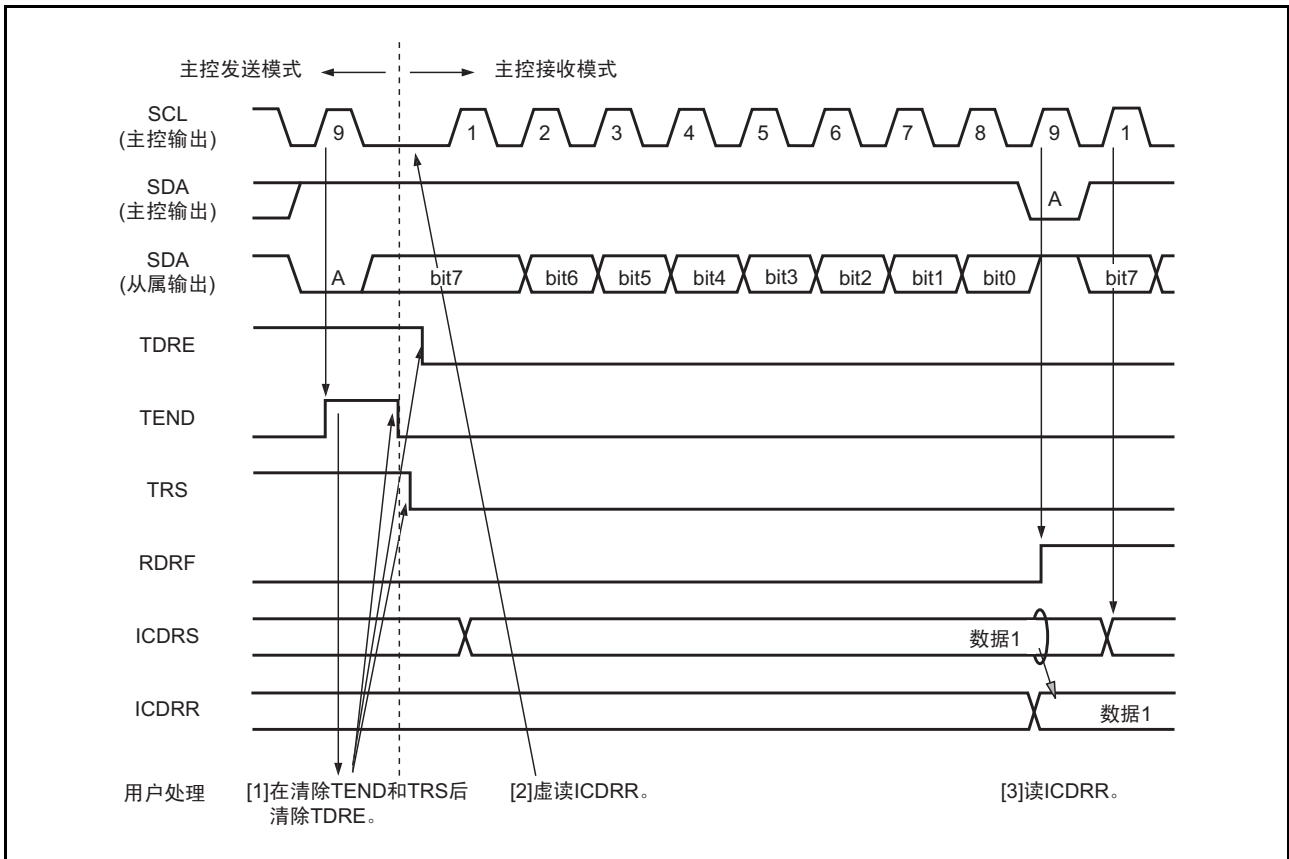


图 20.7 主控接收模式的运行时序 (1)

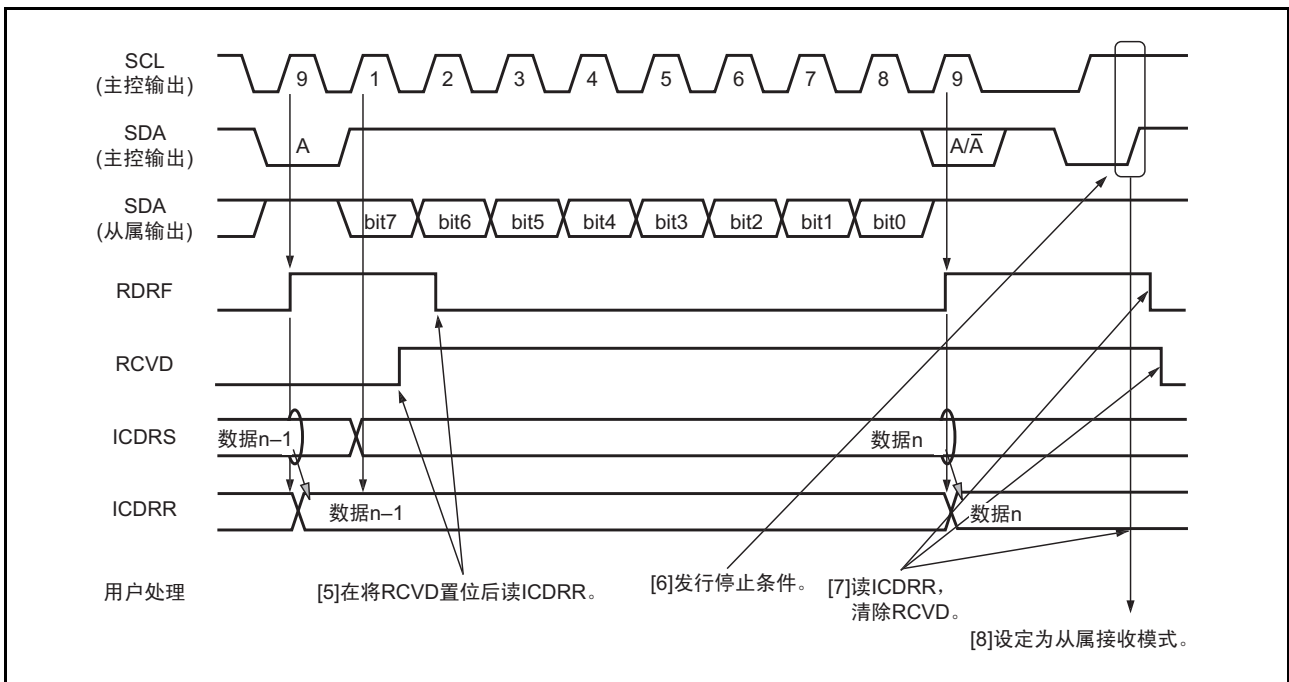


图 20.8 主控接收模式的运行时序 (2)

20.4.4 从属发送

在从属发送模式中，从属设备输出发送数据，主控设备输出接收时钟并且返回应答。有关从属发送模式的运行时序，请参照图 20.9 和图 20.10。

从属发送模式的发送步骤和运行如下所示：

1. 将 ICCR1 的 ICE 位置 “1”，并且设定 ICCR1 的 CKS[3:0] 位等（初始设定）。将 ICCR1 的 MST 位和 TRS 位设定为从属接收模式，等待从属地址相同。
2. 在检测到开始条件后的第 1 帧，如果从属地址相同，从属设备就在第 9 个时钟的上升沿将 ICIER 的 ACKBT 位设定的电平输出到 SDA。此时，当第 8 位的数据（R/W）为 “1” 时，就将 ICCR1 的 TRS 位和 ICSR 的 TDRE 位置位，自动切换为从属发送模式。如果在每次 TDRE 位被置位时将发送数据写到 ICDRT，就能连续发送。
3. 如果在将最后的发送数据写到 ICDRT 后将 TDRE 位置位，就在 TDRE 位为 “1” 的状态下等待 ICSR 的 TEND 位的置位。如果 TEND 位被置位，就清除 TEND 位。
4. 为了结束处理，清除 TRS 位并且虚读 ICDRR，从而释放 SCL。
5. 清除 TDRE 位。

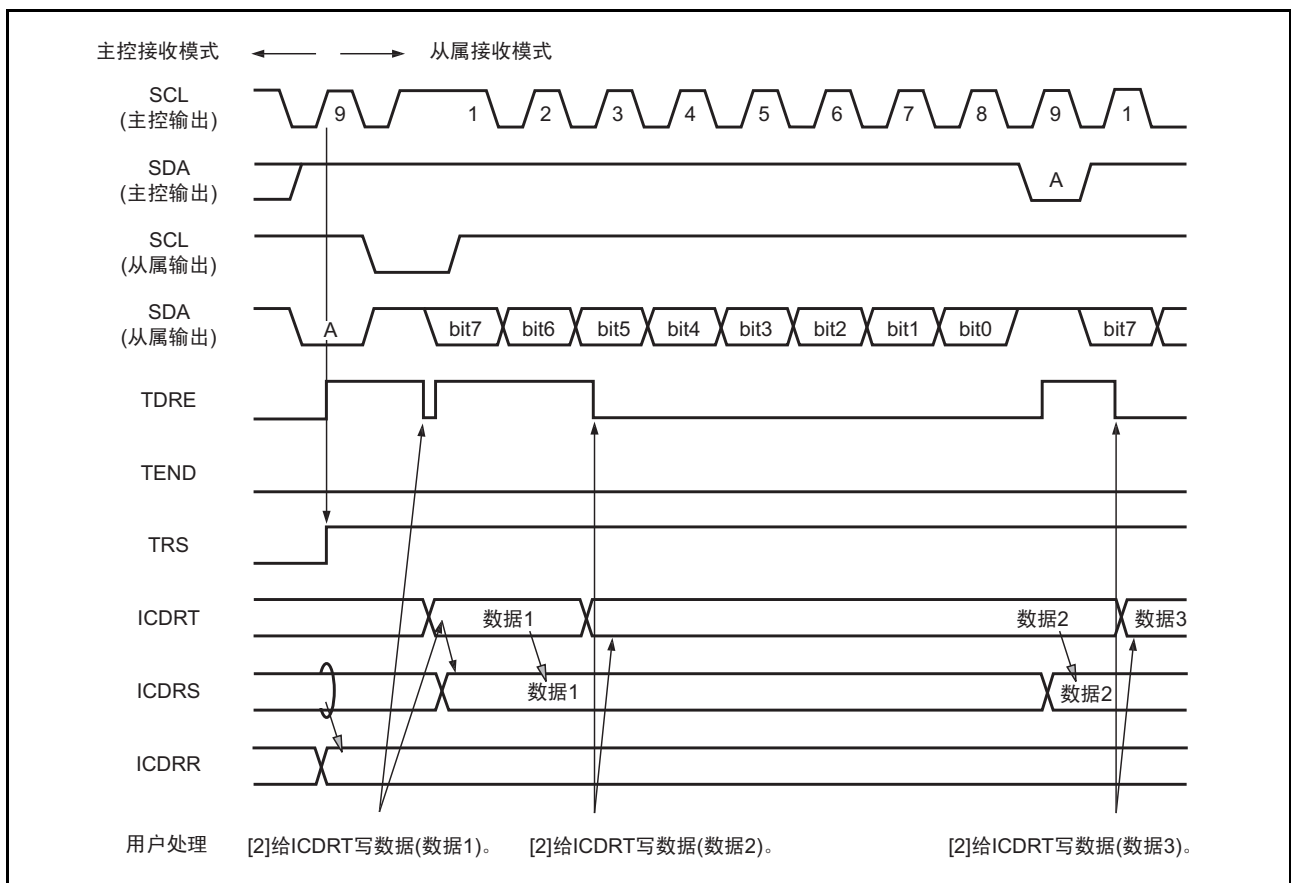


图 20.9 从属发送模式的运行时序 (1)

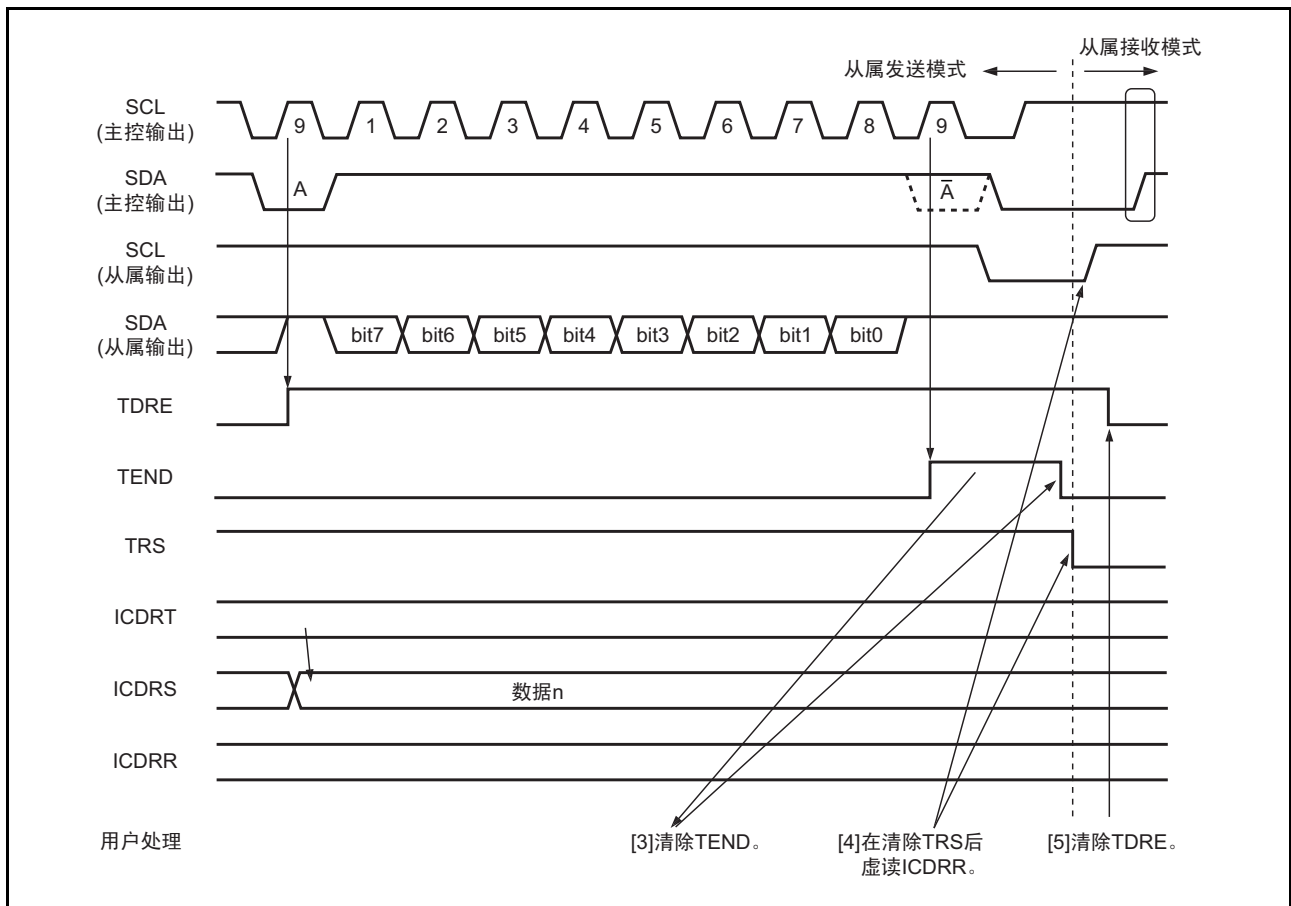


图 20.10 从属发送模式的运行时序 (2)

20.4.5 从属接收

在从属接收模式中，主控设备输出发送时钟和发送数据，从属设备返回应答。有关从属接收模式的运行时序，请参照图 20.11 和图 20.12。从属接收模式的接收步骤和运行如下所示：

1. 将ICCR1的ICE位置“1”，并且设定ICCR1的CKS[3:0]位等（初始设定）。将ICCR1的MST位和TRS位设为从属接收模式，等待从属地址相同。
2. 在检测到开始条件后的第一帧，如果从属地址相同，从属设备就在第 9 个时钟的上升沿将 ICIER 的 ACKBT 位设定的电平输出到 SDA。因为 ICSR 的 RDRF 位同时被置位，所以虚读 ICDRR（因为读到的数据表示从属地址+R/W，所以不需要）。
3. 每当 RDRF 位被置位时，读 ICDRR。如果在 RDRF 位被置位的状态下第 8 个时钟下降，就在读 ICDRR 前将 SCL 固定为 Low 电平。读 ICDRR 前所更改的应答设定（返回给主控设备）反映在下一个传送帧。
4. 同样，通过读 ICDRR 来读取最后字节。

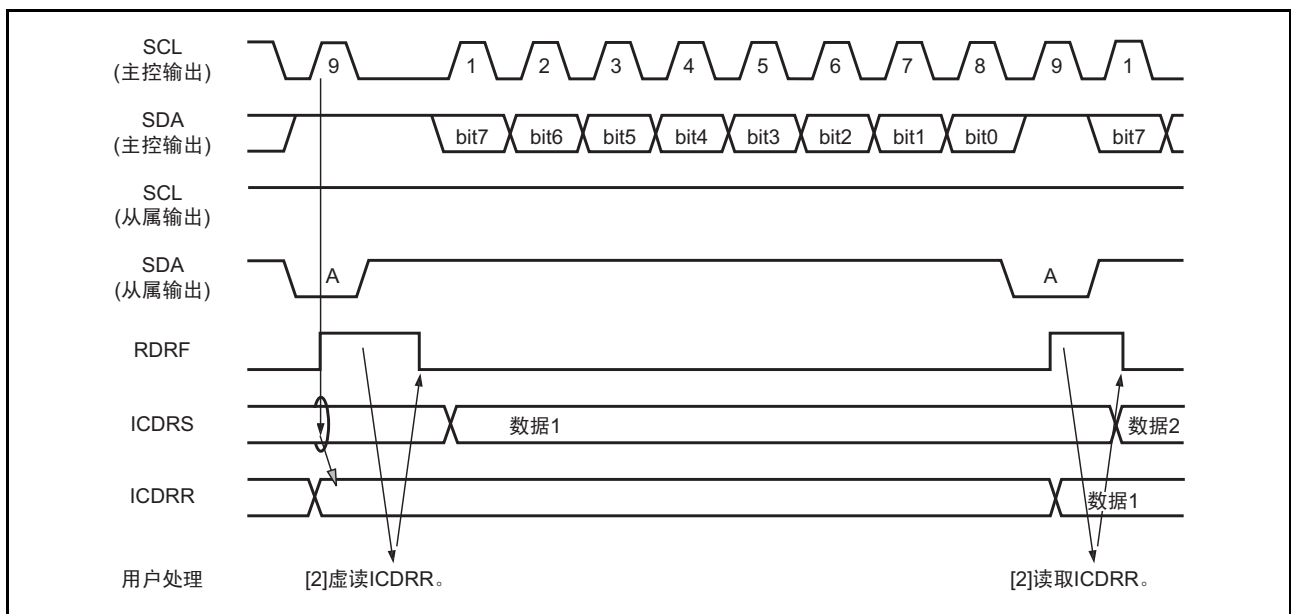


图 20.11 从属接收模式的运行时序（1）

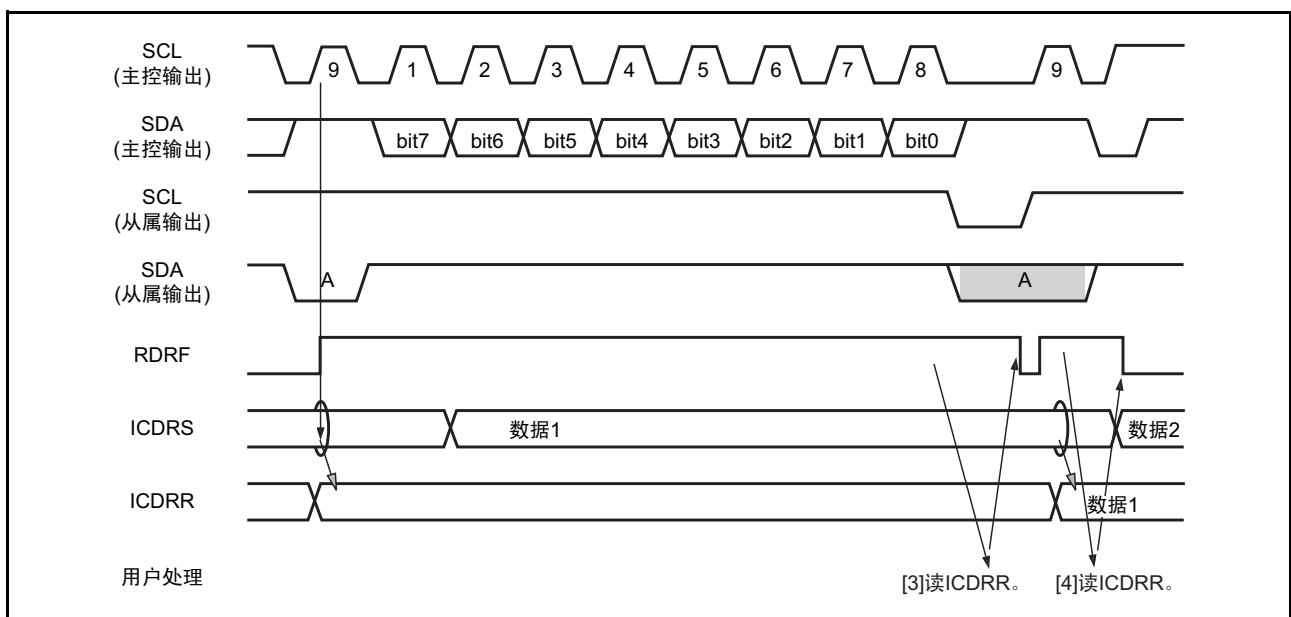


图 20.12 从属接收模式的运行时序（2）

20.4.6 时钟同步串行格式

能通过将 SAR 的 FS 位置“1”，此模块以时钟同步串行格式运行。当 ICCR1 的 MST 位为“1”时，从 SCL 输出传送时钟；当 MST 位为“0”时，输入传送时钟。

(1) 数据传送格式

时钟同步串行格式的数据传送格式如图 20.13 所示。

在 SCL 时钟的一个下降沿和下一个下降沿之间输出传送数据并且在 SCL 时钟上升沿确保数据。数据的传送顺序能通过 ICMR 的 MLS 位选择 MSB first 或者 LSB first。在传送待机中，能通过 ICCR2 的 SDAO 位更改 SDA 的输出电平。

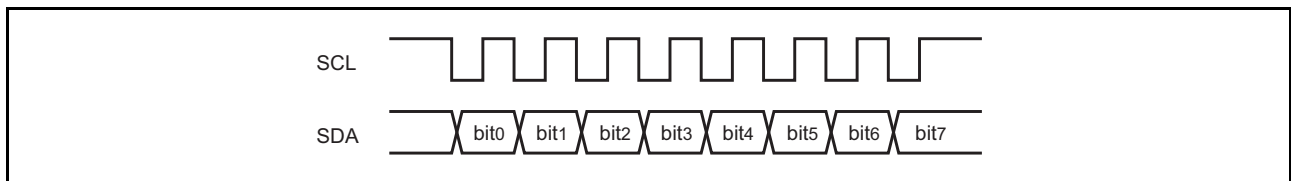


图 20.13 时钟同步串行格式的数据传送格式

(2) 发送运行

在发送模式中，与传送时钟的下降沿同步，从 SDA 输出发送数据。当 ICCR1 的 MST 位为“1”时，输出传送时钟；当 MST 位为“0”时，输入传送时钟。有关发送模式的运行时序，请参照图 20.14。发送模式的步骤和运行如下所示：

1. 将 ICCR1 的 ICE 位置“1”，并且设定 ICCR1 的 MST 位和 CKS[3:0] 位等（初始设定）。
2. 设定 ICCR1 的 TRS 位，使其为发送模式，从而 ICSR 的 TDRE 位被置位。
3. 如果确认 TDRE 位被置位，就将发送数据写到 ICDRT，数据从 ICDRT 被传送到 ICDRS 并且 TDRE 位自动置位。如果在每次 TDRE 被置位时将数据写到 ICDRT，就能连续发送。要从发送模式切换为接收模式时，必须在 TDRE 位被置位的状态下清除 TRS 位。

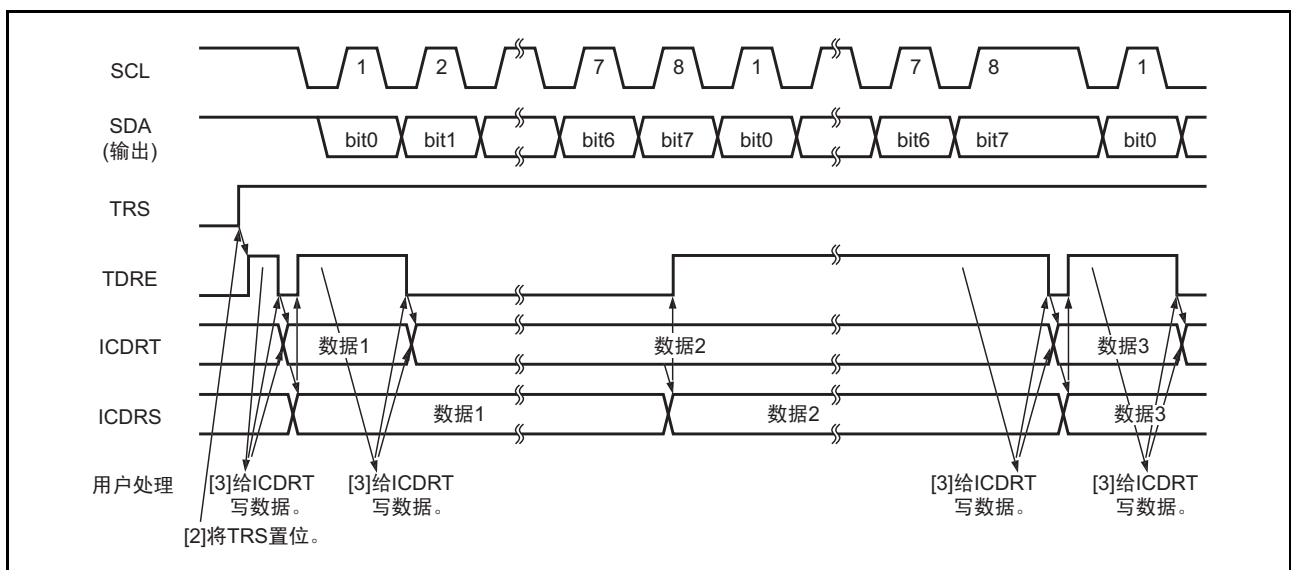


图 20.14 发送模式的运行时序

(3) 接收

在接收模式中，在传送时钟的上升沿锁存数据。当 ICCR1 的 MST 位为“1”时，输出传送时钟；当 MST 位为“0”时，输入传送时钟。有关接收模式的运行时序，请参照图 20.15。接收模式的步骤和运行如下所示：

1. 将 ICCR1 的 ICE 位置“1”，并且设定 ICCR1 的 CKS[3:0] 位等（初始设定）。
2. 在输出传送时钟时，将 MST 位置“1”，开始输出接收时钟。
3. 当接收结束时，将数据从 ICDRS 传送到 ICDRR 并且将 ICSR 的 RDRF 置位。当 MST 位为“1”时，因为可接收下一个字节的状态，所以连续输出时钟。如果在每次 RDRF 被置位时读 ICDRR，就能连续接收。如果在 RDRF 被置位的状态下第 8 个时钟上升，就检测到溢出，将 ICSR 的 AL/OVE 位置位。此时，ICDRR 的值保持前一个接收数据。
4. 当 MST 位为“1”时，为了停止接收，将 ICCR1 的 RCVD 位置位，然后读 ICDRR。于是，在结束下一个字节的接收后将 SCL 固定为 High 电平。

【注】当 MST 位为“1”并且只需要接收 1 字节时，必须按照以下步骤进行。有关运行时序请参照图 20.16。

1. 将 ICCR1 的 ICE 位置“1”，并且设定 ICCR1 的 CKS[3:0] 位等（初始设定）。
2. 在 ICCR1 的 RCVD 位为“0”的状态下将 MST 位置“1”，开始输出接收时钟。
3. 必须在确认 ICMR 的 BC[2] 位被置“1”后将 ICCR1 的 RCVD 位置“1”。因此，在输出 1 字节的接收时钟后将 SCL 固定为 High 电平。

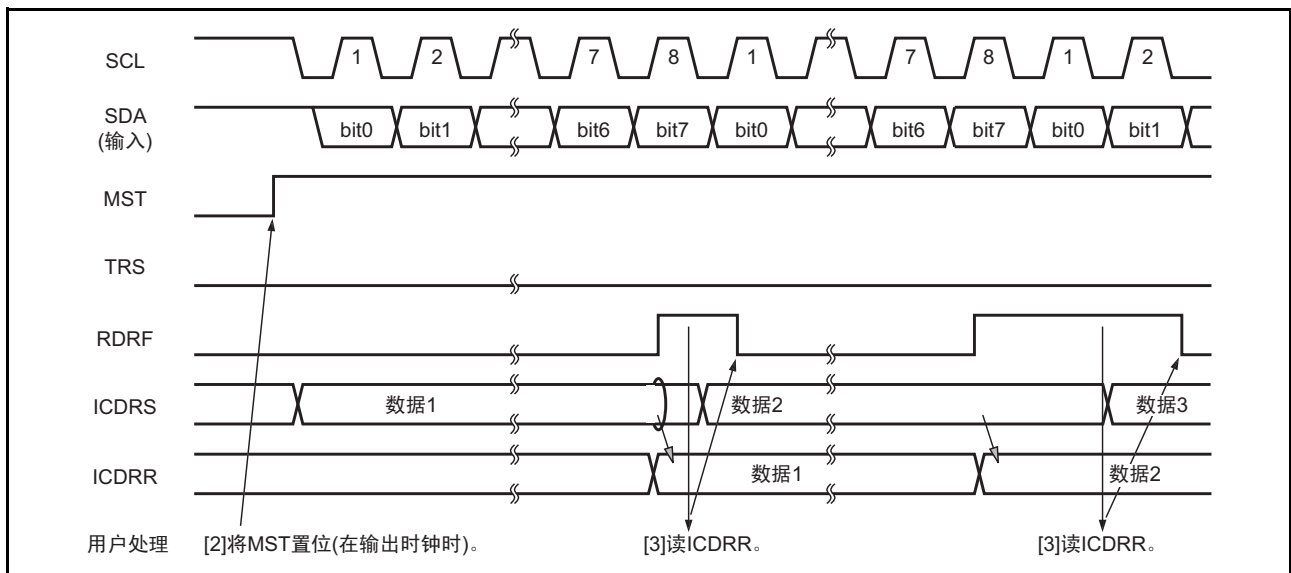


图 20.15 接收模式的运行时序

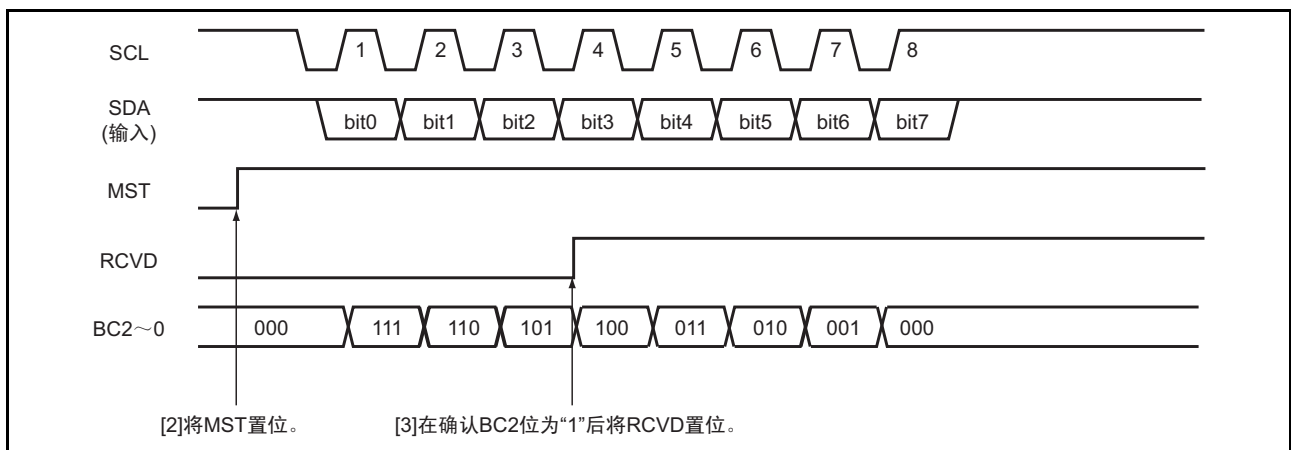


图 20.16 接收 1 字节的运行时序（MST 位为“1”）

20.4.7 噪声消除电路

SCL 引脚和 SDA 引脚的状态经过噪声消除电路被取到内部。噪声消除电路的框图如图 20.17 所示。

噪声消除电路由 3 段串联的锁存电路和匹配检测电路构成。通过外部时钟对 SCL 引脚的输入信号（或者 SDA 引脚的输入信号）进行采样。当 NF2CYC 寄存器为“0”并且 2 个锁存器的输出匹配时，将电平发送给后面的电路；当 NF2CYC 寄存器为“1”并且 3 个锁存器的输出匹配时，将电平发送给后面的电路。当不匹配时，保持以前的值。

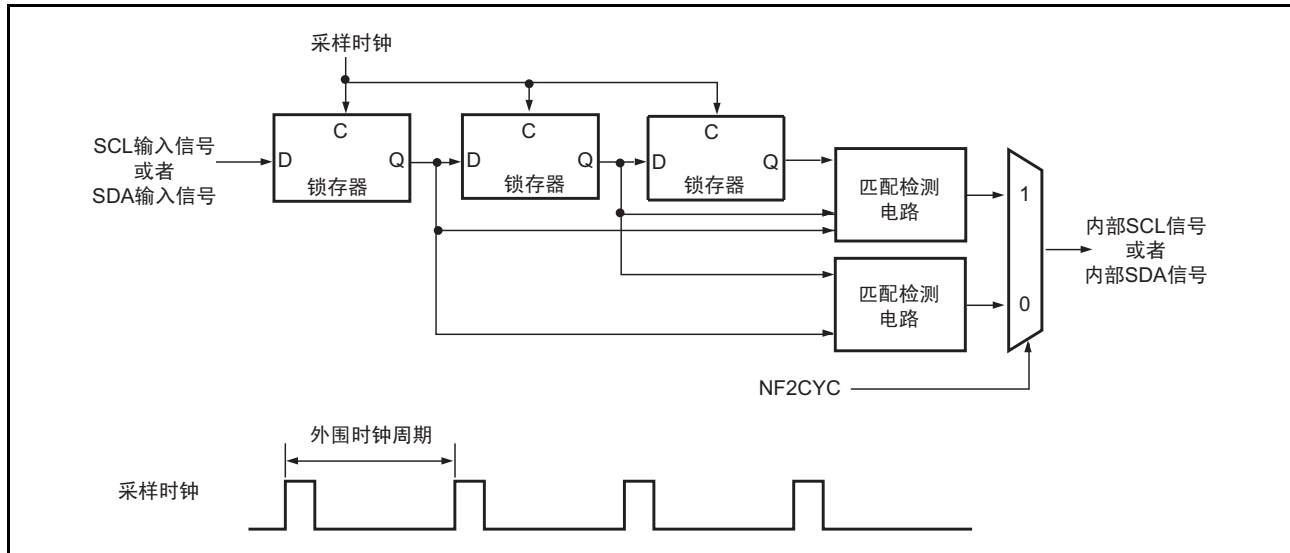


图 20.17 噪声消除电路的框图

20.4.8 使用例子

在使用 I²C 总线接口 3 时，各模式的流程图例子如图 20.18 ~ 图 20.21 所示。

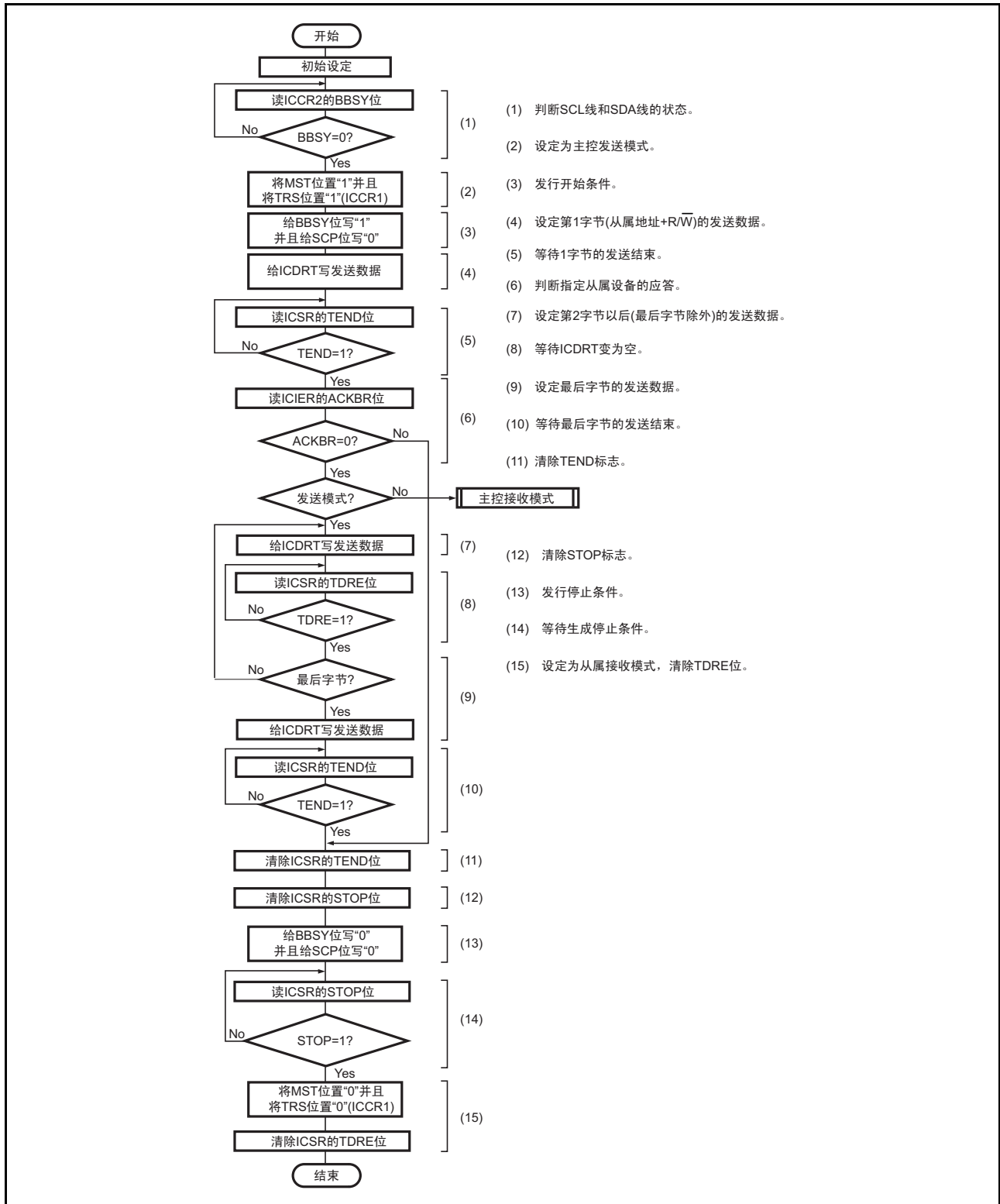


图 20.18 主控发送模式的流程图例子

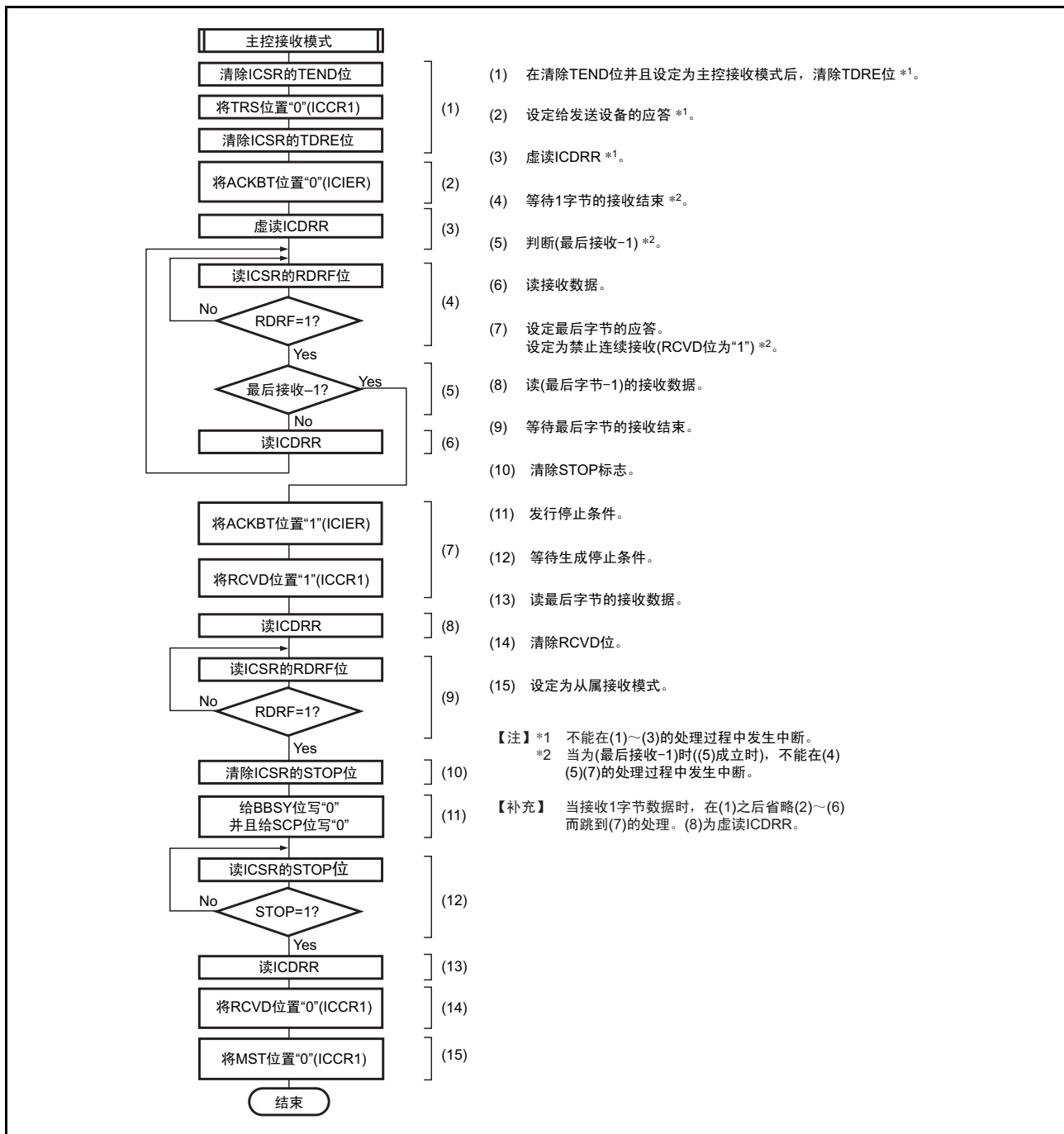


图 20.19 主控接收模式的流程图例子

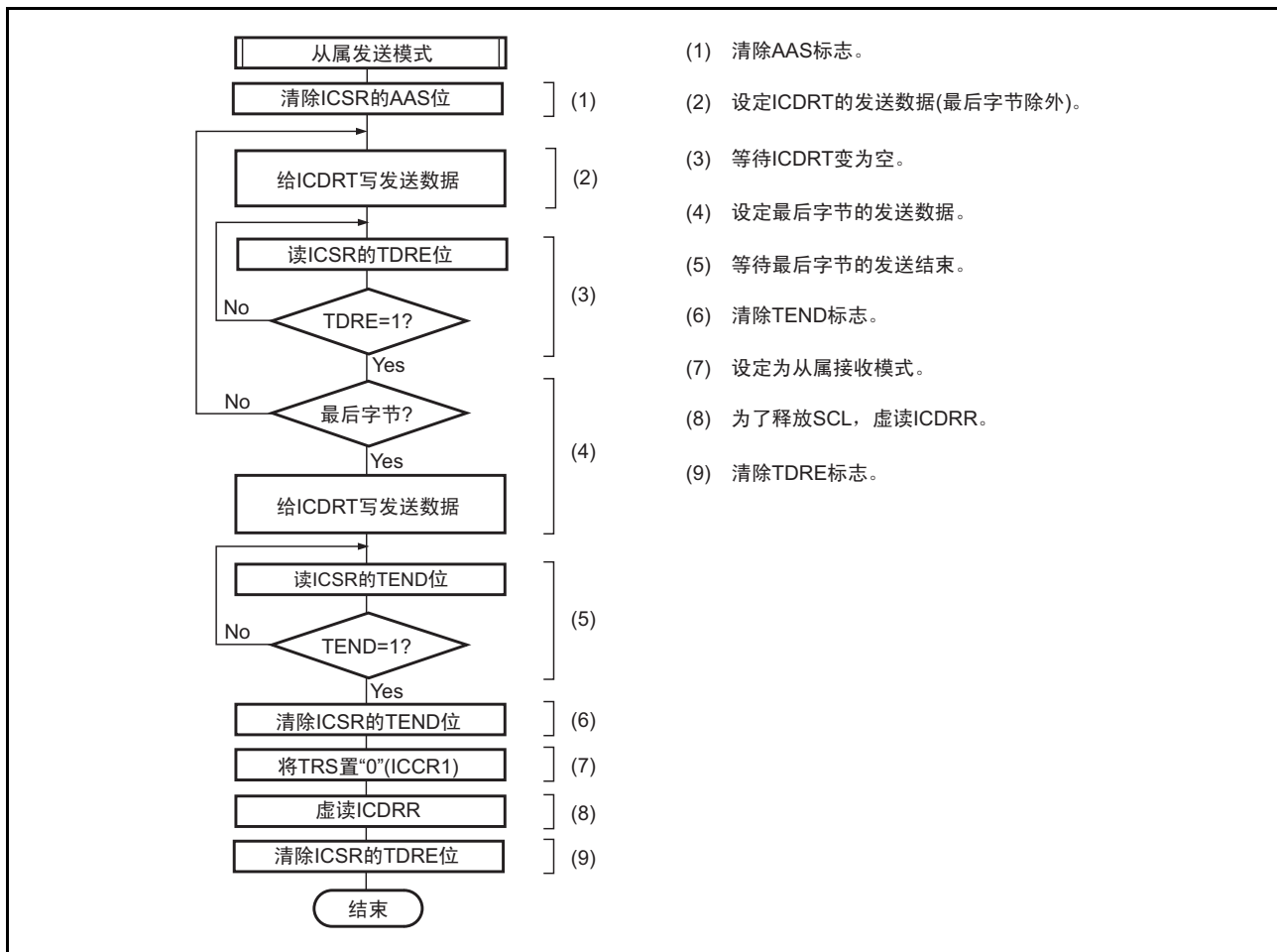


图 20.20 从属发送模式的流程图例子

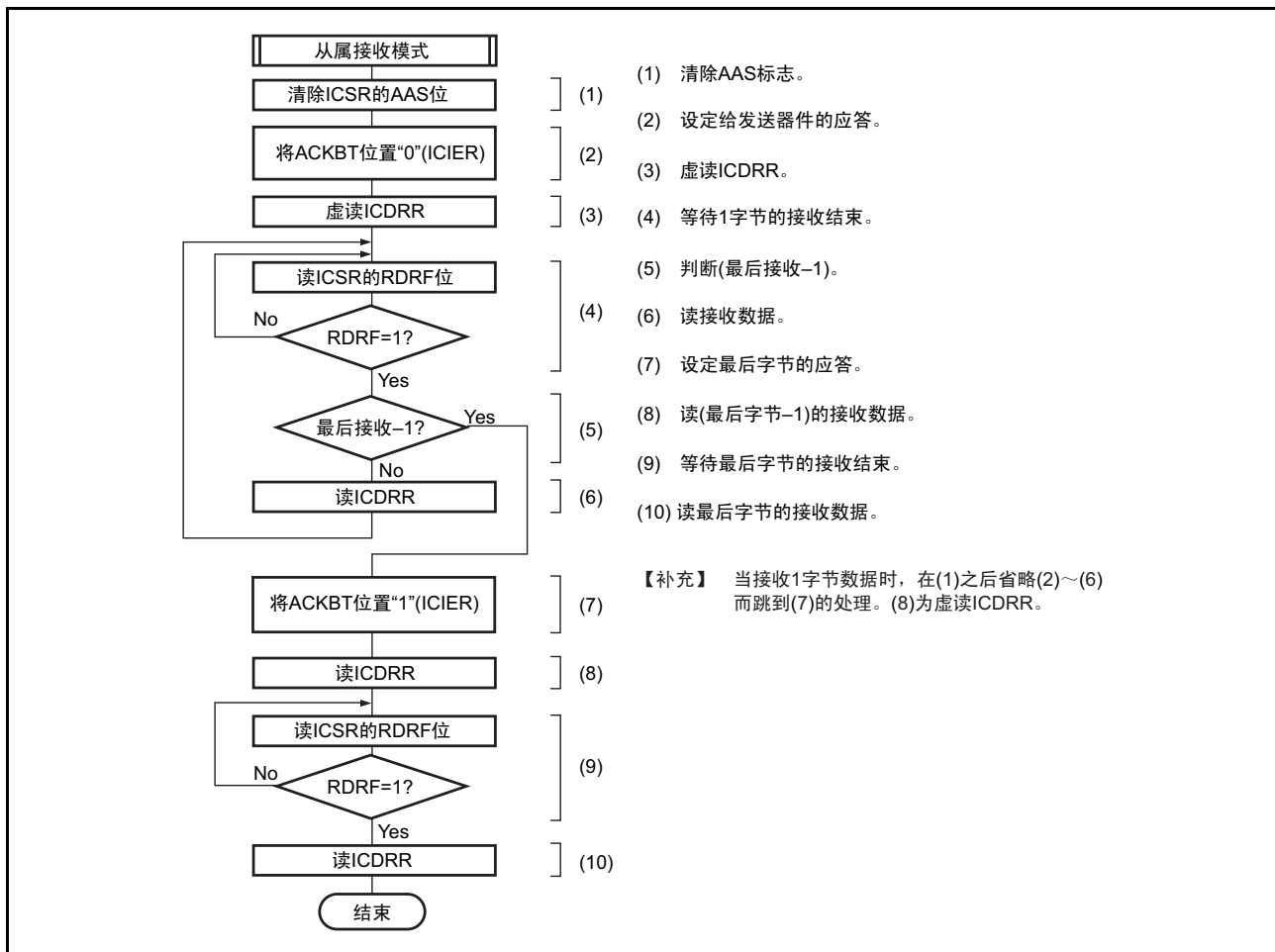


图 20.21 从属接收模式的流程图例子

20.5 中断请求

此模块的中断请求有发送数据空、发送结束、接收数据满、NACK 检测、停止条件检测、仲裁失败 / 溢出错误共 6 种。各中断请求的内容如表 20.4 所示。

表 20.4 中断请求一览表

中断请求	略称	中断条件	I ² C 总线格式	时钟同步串行格式
发送数据空	TXI	(TDRE=1) • (TIE=1)	○	○
发送结束	TEI	(TEND=1) • (TEIE=1)	○	○
接收数据满	RXI	(RDRF=1) • (RIE=1)	○	○
停止条件检测	STPI	(STOP=1) • (STIE=1)	○	×
NACK 检测	NAKI	{(NACKF=1)+(AL=1)} • (NAKIE=1)	○	×
仲裁失败 / 溢出错误			○	○

当表 20.4 的中断条件为“1”时，CPU 执行中断异常处理。如果对直接存储器存取控制器进行了启动设定，TXI 和 RXI 就能启动直接存储器存取控制器。此时，不向 CPU 请求中断。在异常处理中，必须清除各种中断源。但是，必须注意：在将发送数据写到 ICDRT 时，自动清除 TDRE 位和 TEND 位；在读 ICDRR 时，自动清除 RDRF。尤其是在将发送数据写到 ICDRT 的同时 TDRE 位再次被置位，此时如果清除 TDRE 位，就可能多发送 1 个字节。

20.6 位同步电路

在设定为主控模式时，此模块在以下 2 种状态下可能缩短 High 电平期间，因此需要监视 SCL，边按位取得同步边进行通信。

- 当 SCL 被从属设备拉到 Low 电平时
- 因 SCL 线的负载（负载电容、上拉电阻）而使 SCL 的上升变得缓慢时

位同步电路的时序如图 20.22 所示。SCL 从 Low 电平输出变为 Hi-Z 到监控 SCL 为止的时间如表 20.5 所示。

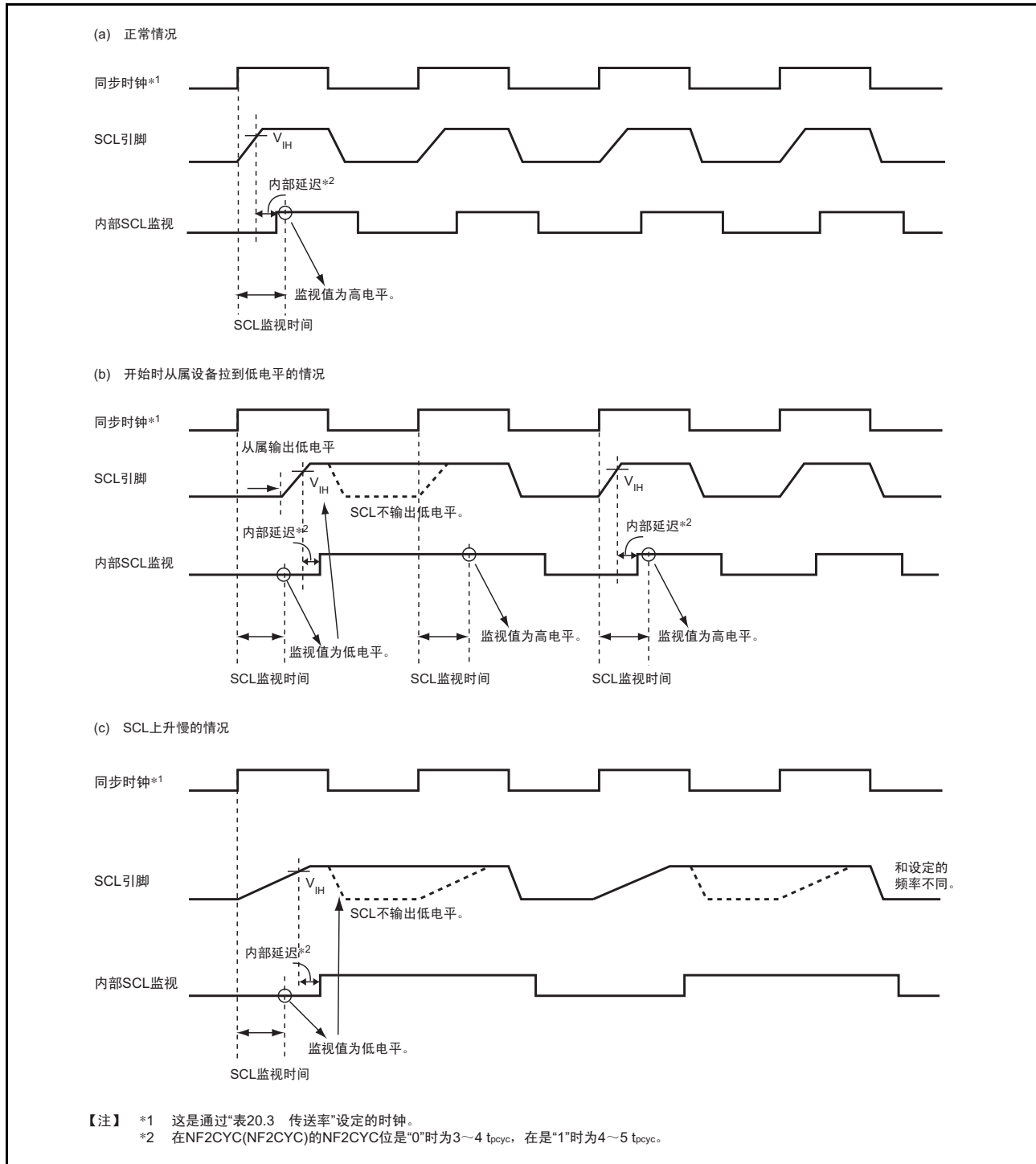


图 20.22 位同步电路的时序

表 20.5 监控 SCL 的时间

CKS4	CKS[3]	CKS[2]	监控 SCL 的时间
0	0	0	9tpcyc
		1	21tpcyc
	1	0	39tpcyc
		1	87tpcyc
1	0	0	79tpcyc
		1	175tpcyc
	1	0	159tpcyc
		1	351tpcyc

【注】 * tpcyc 表示外围时钟 0 (P0 ϕ) 的同步。

20.7 使用时的注意事项

20.7.1 用于多主控时的注意事项

在用于多主控并且此模块设定的传送率 (ICCR1 的 CKS[3:0] 位) 比其他主控慢时, 偶尔会输出预料外宽度的 SCL。

需要将传送率设定为大于等于其他主控中最高传送率的 1/1.8。

20.7.2 主控接收模式时的注意事项

如果在第 8 个时钟的下降沿附近读 ICDRR, 就可能取不到接收数据。

另外, 如果在接收缓冲器为满的状态下并且在第 8 个时钟的下降沿附近将 RCVD 位置 “1”, 就可能无法发行停止条件。

必须使用以下 1. 或者 2. 的方法解决。

1. 在主控接收模式中, 在 8 个时钟的下降沿前读 ICDRR。
2. 在主控接收模式中, 必须将 RCVD 位置 “1”, 按各字节进行通信处理。

20.7.3 在主控接收模式中设定 ACKBT 时的注意事项

在主控接收模式运行时, 必须在连续传送的最后数据的第 8 个 SCL 下降前设定 ACKBT, 否则从属发送设备可能发生溢出。

20.7.4 仲裁失败时的 MST 位和 TRN 位的状态的注意事项

在用于多主控时, 如果按 MST 位和 TRS 位的顺序进行位操作来设定为主控发送, 就根据 TRS 的位操作指令执行过程中的仲裁失败发生时序, 有可能出现 ICSR 的 AL 位为 “1” 而运行模式为主控发送模式 (MST=1, TRS=1) 的矛盾状态。

为了避免这种现象, 使用以下方法:

- 在用于多主控时, 必须使用 MOV 指令设定 MST 位和 TRS 位。
- 当发生仲裁失败时, 必须确认 MST 位和 TRS 位是否都为 “0”, 如果不是都为 “0”, 就必须重新将 MST 位和 TRS 位置 “0”。

20.7.5 I²C 总线接口模式的主控接收模式中的注意事项

必须在主控接收后确认 SCL 的第 9 个时钟的下降沿，然后发行停止条件或者重新发行开始条件。

20.7.6 有关 IICRST 位和 BBSY 位的注意事项

如果给 ICCR2 的 IICRST 位写“1”，本 LSI 就释放 SCL 引脚和 SDA 引脚。此时，如果在 SCL 为 High 电平的状态下 SDA 从 Low 电平变为 High 电平，就认为是停止条件并且将 ICCR2 的 BBSY 位清“0”。

20.7.7 主控发送模式并且将 ACKE 位设定为“1”时的停止条件的发行注意事项

当通过主控发送模式并且将 I²C 总线中断允许寄存器（ICIER）的 ACKE 位设定为“1”来发行停止条件时，根据发行时序可能不能正常输出停止条件。

为了防止此现象的发生，必须在确认第 9 个时钟的下降沿后发行停止条件。能通过检查 I²C 控制寄存器 2（ICCR2）的 SCLO 位来确认第 9 个时钟的下降沿。

21. 串行音频接口

串行音频接口是和 I²S 总线兼容的各种设备进行音频数据的发送和接收的模块。串行音频接口不仅支持其他一般格式，而且还支持多通道模式。

21.1 特点

- 通道个数：6个通道
- 运行模式：非压缩模式
非压缩模式支持被分到各通道的串行音频流。
- 可用作发送器或者接收器。
通道0能进行全双工通信。
- 能使用串行总线格式。
- 数据缓冲器和移位寄存器之间进行异步传送。
- 可选择用于串行总线接口的时钟分频比。
- 能通过DMA传送或者中断来控制数据的发送和接收。
- 可从以下引脚中选择过采样时钟：
AUDIO_CLK（1~50MHz）
AUDIO_X1、AUDIO_X2（连接晶体谐振器：10~50MHz，输入外部时钟：1~50MHz）
- 发送部和接收部内置8段FIFO缓冲器。
- 支持SSIWS只在最初的系统字区间为高电平的多通道模式（TDM模式）。
- 支持在不停止SSIWS的状态下运行的模式（WS继续模式）。

框图如图 21.1 所示。

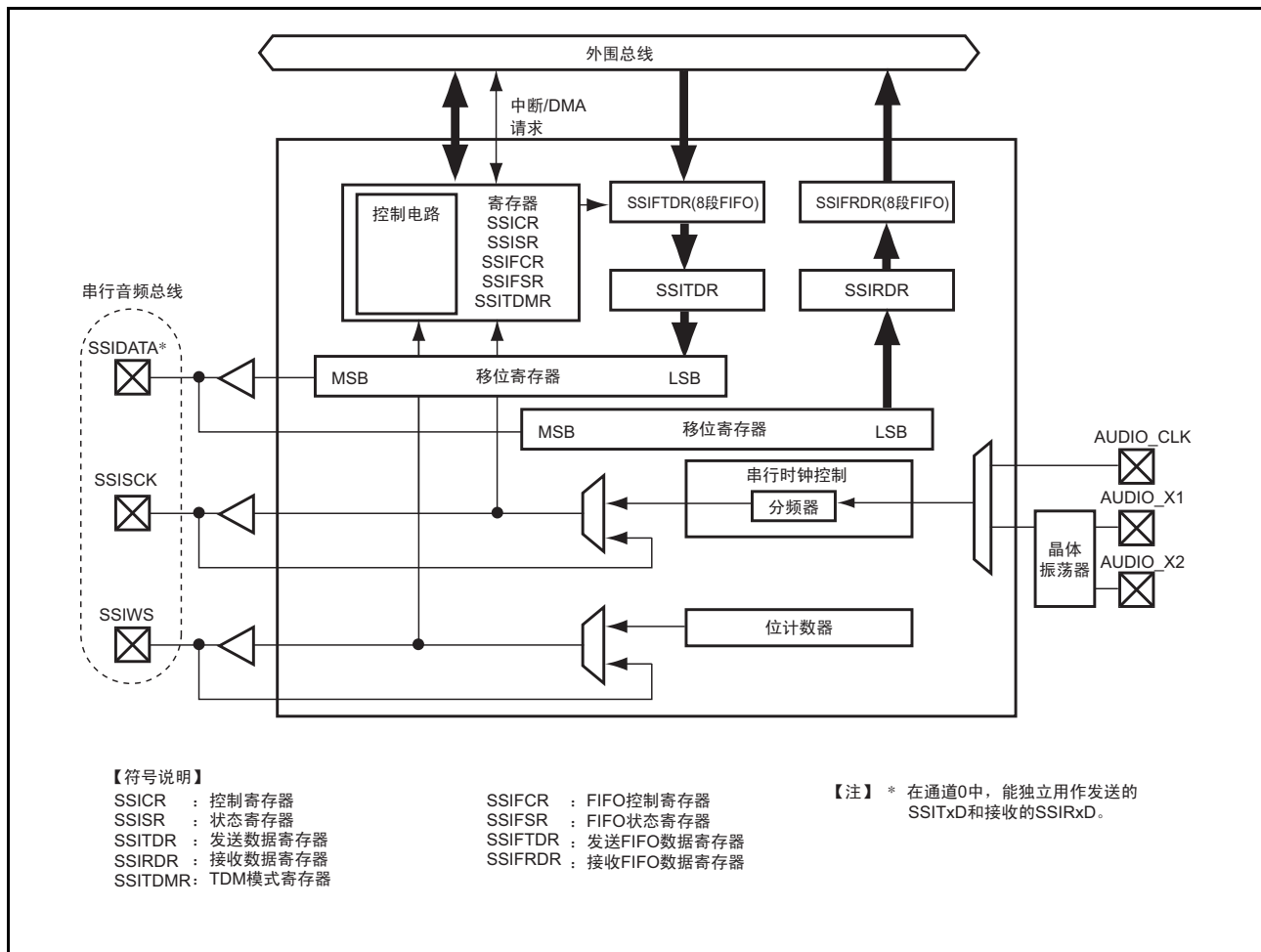


图 21.1 框图

21.2 输入 / 输出引脚

引脚结构如表 21.1 所示。

表 21.1 引脚结构

通道	名称	输入 / 输出	功能
0	SSISCK0	输入 / 输出	串行位时钟
	SSIWS0	输入 / 输出	字选择
	SSITxD0	输出	串行数据的输出
	SSIRxD0	输入	串行数据的输入
1 ~ 5	SSISCK1 ~ SSISCK5	输入 / 输出	串行位时钟
	SSIWS1 ~ SSIWS5	输入 / 输出	字选择
	SSIDATA1 ~ SSIDATA5	输入 / 输出	串行数据的输入 / 输出
共用	AUDIO_CLK	输入	用于音频的外部时钟（输入过采样时钟）
	AUDIO_X1	输入	用于音频的晶体谐振器 / 外部时钟（输入过采样时钟）
	AUDIO_X2	输出	

21.3 寄存器说明

寄存器结构如表 21.2 所示。另外，在本章节的说明中省略各通道的区别。

表 21.2 寄存器结构

通道	寄存器名	略称	R/W	初始值	地址	存取长度
0	控制寄存器 0	SSICR_0	R/W	H'00000000	H'FFFF0000	8、16、32
	状态寄存器 0	SSISR_0	R/W*1	H'02000013	H'FFFF0004	8、16、32
	FIFO 控制寄存器 0	SSIFCR_0	R/W	H'00000000	H'FFFF0010	8、16、32
	FIFO 状态寄存器 0	SSIFSR_0	R/(W)*2	H'00010000	H'FFFF0014	8、16、32
	发送 FIFO 数据寄存器 0	SSIFTDR_0	W	不定值	H'FFFF0018	32
	接收 FIFO 数据寄存器 0	SSIFRDR_0	R	不定值	H'FFFF001C	32
	TDM 模式寄存器 0	SSITDMR_0	R/W	H'00000000	H'FFFF0020	8、16、32
1	控制寄存器 1	SSICR_1	R/W	H'00000000	H'FFFF0800	8、16、32
	状态寄存器 1	SSISR_1	R/W*1	H'02000013	H'FFFF0804	8、16、32
	FIFO 控制寄存器 1	SSIFCR_1	R/W	H'00000000	H'FFFF0810	8、16、32
	FIFO 状态寄存器 1	SSIFSR_1	R/(W)*2	H'00010000	H'FFFF0814	8、16、32
	发送 FIFO 数据寄存器 1	SSIFTDR_1	W	不定值	H'FFFF0818	32
	接收 FIFO 数据寄存器 1	SSIFRDR_1	R	不定值	H'FFFF081C	32
	TDM 模式寄存器 1	SSITDMR_1	R/W	H'00000000	H'FFFF0820	8、16、32
2	控制寄存器 2	SSICR_2	R/W	H'00000000	H'FFFF1000	8、16、32
	状态寄存器 2	SSISR_2	R/W*1	H'02000013	H'FFFF1004	8、16、32
	FIFO 控制寄存器 2	SSIFCR_2	R/W	H'00000000	H'FFFF1010	8、16、32
	FIFO 状态寄存器 2	SSIFSR_2	R/(W)*2	H'00010000	H'FFFF1014	8、16、32
	发送 FIFO 数据寄存器 2	SSIFTDR_2	W	不定值	H'FFFF1018	32
	接收 FIFO 数据寄存器 2	SSIFRDR_2	R	不定值	H'FFFF101C	32
	TDM 模式寄存器 2	SSITDMR_2	R/W	H'00000000	H'FFFF1020	8、16、32

通道	寄存器名	略称	R/W	初始值	地址	存取长度
3	控制寄存器 3	SSICR_3	R/W	H'00000000	H'FFFF1800	8、16、32
	状态寄存器 3	SSISR_3	R/W*1	H'02000013	H'FFFF1804	8、16、32
	FIFO 控制寄存器 3	SSIFCR_3	R/W	H'00000000	H'FFFF1810	8、16、32
	FIFO 状态寄存器 3	SSIFSR_3	R/(W)*2	H'00010000	H'FFFF1814	8、16、32
	发送 FIFO 数据寄存器 3	SSIFTDR_3	W	不定值	H'FFFF1818	32
	接收 FIFO 数据寄存器 3	SSIFRDR_3	R	不定值	H'FFFF181C	32
	TDM 模式寄存器 3	SSITDMR_3	R/W	H'00000000	H'FFFF1820	8、16、32
4	控制寄存器 4	SSICR_4	R/W	H'00000000	H'FFFF2000	8、16、32
	状态寄存器 4	SSISR_4	R/W*1	H'02000013	H'FFFF2004	8、16、32
	FIFO 控制寄存器 4	SSIFCR_4	R/W	H'00000000	H'FFFF2010	8、16、32
	FIFO 状态寄存器 4	SSIFSR_4	R/(W)*2	H'00010000	H'FFFF2014	8、16、32
	发送 FIFO 数据寄存器 4	SSIFTDR_4	W	不定值	H'FFFF2018	32
	接收 FIFO 数据寄存器 4	SSIFRDR_4	R	不定值	H'FFFF201C	32
	TDM 模式寄存器 4	SSITDMR_4	R/W	H'00000000	H'FFFF2020	8、16、32
5	控制寄存器 5	SSICR_5	R/W	H'00000000	H'FFFF2800	8、16、32
	状态寄存器 5	SSISR_5	R/W*1	H'02000013	H'FFFF2804	8、16、32
	FIFO 控制寄存器 5	SSIFCR_5	R/W	H'00000000	H'FFFF2810	8、16、32
	FIFO 状态寄存器 5	SSIFSR_5	R/(W)*2	H'00010000	H'FFFF2814	8、16、32
	发送 FIFO 数据寄存器 5	SSIFTDR_5	W	不定值	H'FFFF2818	32
	接收 FIFO 数据寄存器 5	SSIFRDR_5	R	不定值	H'FFFF281C	32
	TDM 模式寄存器 5	SSITDMR_5	R/W	H'00000000	H'FFFF2820	8、16、32

【注】 *1 此寄存器的 bit29 ~ 26 是可读写位，其他位是只读位，详细内容请参照“21.3.2 状态寄存器（SSISR）”。

*2 为了清除标志，只能给此寄存器的 bit16 和 bit0 写“0”，其他位是只读位，详细内容请参照“21.3.6 FIFO 状态寄存器（SSIFSR）”。

21.3.1 控制寄存器 (SSICR)

SSICR 是 32 位可读写寄存器，控制 IRQ，选择各极性的状态以及设定运行模式。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IEN	—	CHNL[1:0]	DWL[2:0]		SWL[2:0]				
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]			MUEN	—	TEN	REN	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

位	位名	初始值	R/W	说明
31	—	0	R	保留位 读取值为不定值，写入值总是“0”。
30	CKS	0	R/W	过采样时钟的选择 选择过采样时钟的供给源。 0: 输入 AUDIO_X1 1: 输入 AUDIO_CLK
29	TUIEN	0	R/W	发送下溢中断的允许 0: 禁止下溢中断 1: 允许下溢中断
28	TOIEN	0	R/W	发送上溢中断的允许 0: 禁止上溢中断 1: 允许上溢中断
27	RUIEN	0	R/W	接收下溢中断的允许 0: 禁止下溢中断 1: 允许下溢中断
26	ROIEN	0	R/W	接收上溢中断的允许 0: 禁止上溢中断 1: 允许上溢中断
25	IEN	0	R/W	空闲模式中中断的允许 0: 禁止空闲模式中中断 1: 允许空闲模式中中断
24	—	0	R	保留位 读取值为不定值，写入值总是“0”。

位	位名	初始值	R/W	说明
23、22	CHNL[1:0]	00	R/W	<p>通道</p> <p>【TDM 为“0”的情况】 表示各系统字的通道个数。 00: 各系统字由 1 个通道构成 01: 各系统字由 2 个通道构成 10: 各系统字由 3 个通道构成 11: 各系统字由 4 个通道构成</p> <p>【TDM 为“1”的情况】 表示 TDM 帧的系统字数。 00: 禁止设定 01: TDM 帧由 4 个系统字构成 10: TDM 帧由 6 个系统字构成 11: TDM 帧由 8 个系统字构成。</p>
21 ~ 19	DWL[2:0]	000	R/W	<p>数据字长</p> <p>表示数据字的位数。</p> <p>000: 8 位 001: 16 位 010: 18 位 011: 20 位 100: 22 位 101: 24 位 110: 32 位 111: 禁止设定</p>
18 ~ 16	SWL[2:0]	000	R/W	<p>系统字长</p> <p>表示系统字的位数。</p> <p>000: 8 位 001: 16 位 010: 24 位 011: 32 位 100: 48 位 101: 64 位 110: 128 位 111: 256 位</p>
15	SCKD	0	R/W	<p>串行位时钟方向</p> <p>0: 串行位时钟输入、从属模式 1: 串行位时钟输出、主控模式</p> <p>【注】 只能将 (SCKD、SWSD) 设定为 (0、0) 和 (1、1)，禁止其他设定。</p>
14	SWSD	0	R/W	<p>串行 WS 方向</p> <p>0: 串行字选择输入、从属模式 1: 串行字选择输出、主控模式</p> <p>【注】 只能将 (SCKD、SWSD) 设定为 (0、0) 和 (1、1)，禁止其他设定。</p>

位	位名	初始值	R/W	说明															
13	SCKP	0	R/W	串行位时钟极性 0: SSIWS 和 SSIDATA 在 SSISCK 的下降沿发生变化 (在 SCK 上升沿采样) 1: SSIWS 和 SSIDATA 在 SSISCK 的上升沿发生变化 (在 SCK 下降沿采样) <table border="1" data-bbox="699 477 1359 770" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th></th> <th>SCKP=0</th> <th>SCKP=1</th> </tr> </thead> <tbody> <tr> <td>接收时 SSIDATA输入采样时序</td> <td>SSISCK 上升沿</td> <td>SSISCK 下降沿</td> </tr> <tr> <td>发送时 SSIDATA输出变化时序</td> <td>SSISCK 下降沿</td> <td>SSISCK 上升沿</td> </tr> <tr> <td>从属模式中 (SWSD=0) SSIWS输入采样时序</td> <td>SSISCK 上升沿</td> <td>SSISCK 下降沿</td> </tr> <tr> <td>主模式中 (SWSD=1) SSIWS输出变化时序</td> <td>SSISCK 下降沿</td> <td>SSISCK 上升沿</td> </tr> </tbody> </table>		SCKP=0	SCKP=1	接收时 SSIDATA输入采样时序	SSISCK 上升沿	SSISCK 下降沿	发送时 SSIDATA输出变化时序	SSISCK 下降沿	SSISCK 上升沿	从属模式中 (SWSD=0) SSIWS输入采样时序	SSISCK 上升沿	SSISCK 下降沿	主模式中 (SWSD=1) SSIWS输出变化时序	SSISCK 下降沿	SSISCK 上升沿
	SCKP=0	SCKP=1																	
接收时 SSIDATA输入采样时序	SSISCK 上升沿	SSISCK 下降沿																	
发送时 SSIDATA输出变化时序	SSISCK 下降沿	SSISCK 上升沿																	
从属模式中 (SWSD=0) SSIWS输入采样时序	SSISCK 上升沿	SSISCK 下降沿																	
主模式中 (SWSD=1) SSIWS输出变化时序	SSISCK 下降沿	SSISCK 上升沿																	
12	SWSP	0	R/W	串行 WS 极性 【TDM 位为 “0” 的情况】 0: SSIWS 在第 1 通道为低电平, 在第 2 通道为高电平。 1: SSIWS 在第 1 通道为高电平, 在第 2 通道为低电平。 【TDM 位为 “1” 的情况】 0: SSIWS 只在系统字 1 的区间为高电平, 在其他区间为低电平。 1: 禁止设定															
11	SPDP	0	R/W	串行填充极性 0: 填充位为低电平 1: 填充位为高电平															
10	SDTA	0	R/W	串行数据调整 0: 按串行数据、填充位的顺序进行发送和接收 1: 按填充位、串行数据的顺序进行发送和接收															

位	位名	初始值	R/W	说明																																																																																																																								
9	PDTA	0	R/W	<p>并行数据调整</p> <p>当数据字长为 32 位时，此位无意义。</p> <p>此位适用于接收模式的 SSIRDR 和发送模式的 SSITDR。</p> <p>【数据字长为 8 位或者 16 位的情况】</p> <p>0: 先发送和接收并行数据 (SSITDR 和 SSIRDR) 的低位</p> <p>1: 先发送和接收并行数据 (SSITDR 和 SSIRDR) 的高位</p> <p>【数据字长为 18 位、20 位、22 位或者 24 位的情况】</p> <p>0: 将并行数据 (SSITDR、SSIRDR) 向左靠紧</p> <p>1: 将并行数据 (SSITDR、SSIRDR) 向右靠紧</p> <p>• PDTA = 0</p> <table border="1" data-bbox="815 658 1254 1077"> <thead> <tr> <th>DWL[2:0]</th> <th>SSITDR/SSIRDR[31:0]</th> </tr> </thead> <tbody> <tr> <td>000</td> <td> <table border="1"> <tr> <td>31</td> <td>24 23</td> <td>16 15</td> <td>8 7</td> <td>0</td> </tr> <tr> <td colspan="2">第4个字</td> <td colspan="2">第3个字</td> <td>第1个字</td> </tr> </table> </td> </tr> <tr> <td>001</td> <td> <table border="1"> <tr> <td>31</td> <td>16 15</td> <td>0</td> </tr> <tr> <td colspan="2">第2个字</td> <td>第1个字</td> </tr> </table> </td> </tr> <tr> <td>010</td> <td> <table border="1"> <tr> <td>31</td> <td>14 13</td> <td>0</td> </tr> <tr> <td colspan="2">有效</td> <td>无效</td> </tr> </table> </td> </tr> <tr> <td>011</td> <td> <table border="1"> <tr> <td>31</td> <td>12 11</td> <td>0</td> </tr> <tr> <td colspan="2">有效</td> <td>无效</td> </tr> </table> </td> </tr> <tr> <td>100</td> <td> <table border="1"> <tr> <td>31</td> <td>10 9</td> <td>0</td> </tr> <tr> <td colspan="2">有效</td> <td>无效</td> </tr> </table> </td> </tr> <tr> <td>101</td> <td> <table border="1"> <tr> <td>31</td> <td>8 7</td> <td>0</td> </tr> <tr> <td colspan="2">有效</td> <td>无效</td> </tr> </table> </td> </tr> <tr> <td>110</td> <td> <table border="1"> <tr> <td>31</td> <td>0</td> </tr> <tr> <td colspan="2">有效</td> </tr> </table> </td> </tr> </tbody> </table> <p>• PDTA = 1</p> <table border="1" data-bbox="823 1135 1246 1541"> <thead> <tr> <th>DWL[2:0]</th> <th>SSITDR/SSIRDR[31:0]</th> </tr> </thead> <tbody> <tr> <td>000</td> <td> <table border="1"> <tr> <td>31</td> <td>24 23</td> <td>16 15</td> <td>8 7</td> <td>0</td> </tr> <tr> <td>第1个字</td> <td>第2个字</td> <td>第3个字</td> <td>第4个字</td> <td></td> </tr> </table> </td> </tr> <tr> <td>001</td> <td> <table border="1"> <tr> <td>31</td> <td>16 15</td> <td>0</td> </tr> <tr> <td colspan="2">第1个字</td> <td>第2个字</td> </tr> </table> </td> </tr> <tr> <td>010</td> <td> <table border="1"> <tr> <td>31</td> <td>18 17</td> <td>0</td> </tr> <tr> <td>无效</td> <td>有效</td> <td></td> </tr> </table> </td> </tr> <tr> <td>011</td> <td> <table border="1"> <tr> <td>31</td> <td>20 19</td> <td>0</td> </tr> <tr> <td>无效</td> <td>有效</td> <td></td> </tr> </table> </td> </tr> <tr> <td>100</td> <td> <table border="1"> <tr> <td>31</td> <td>22 21</td> <td>0</td> </tr> <tr> <td>无效</td> <td>有效</td> <td></td> </tr> </table> </td> </tr> <tr> <td>101</td> <td> <table border="1"> <tr> <td>31</td> <td>24 23</td> <td>0</td> </tr> <tr> <td>无效</td> <td>有效</td> <td></td> </tr> </table> </td> </tr> <tr> <td>110</td> <td> <table border="1"> <tr> <td>31</td> <td>0</td> </tr> <tr> <td colspan="2">有效</td> </tr> </table> </td> </tr> </tbody> </table>	DWL[2:0]	SSITDR/SSIRDR[31:0]	000	<table border="1"> <tr> <td>31</td> <td>24 23</td> <td>16 15</td> <td>8 7</td> <td>0</td> </tr> <tr> <td colspan="2">第4个字</td> <td colspan="2">第3个字</td> <td>第1个字</td> </tr> </table>	31	24 23	16 15	8 7	0	第4个字		第3个字		第1个字	001	<table border="1"> <tr> <td>31</td> <td>16 15</td> <td>0</td> </tr> <tr> <td colspan="2">第2个字</td> <td>第1个字</td> </tr> </table>	31	16 15	0	第2个字		第1个字	010	<table border="1"> <tr> <td>31</td> <td>14 13</td> <td>0</td> </tr> <tr> <td colspan="2">有效</td> <td>无效</td> </tr> </table>	31	14 13	0	有效		无效	011	<table border="1"> <tr> <td>31</td> <td>12 11</td> <td>0</td> </tr> <tr> <td colspan="2">有效</td> <td>无效</td> </tr> </table>	31	12 11	0	有效		无效	100	<table border="1"> <tr> <td>31</td> <td>10 9</td> <td>0</td> </tr> <tr> <td colspan="2">有效</td> <td>无效</td> </tr> </table>	31	10 9	0	有效		无效	101	<table border="1"> <tr> <td>31</td> <td>8 7</td> <td>0</td> </tr> <tr> <td colspan="2">有效</td> <td>无效</td> </tr> </table>	31	8 7	0	有效		无效	110	<table border="1"> <tr> <td>31</td> <td>0</td> </tr> <tr> <td colspan="2">有效</td> </tr> </table>	31	0	有效		DWL[2:0]	SSITDR/SSIRDR[31:0]	000	<table border="1"> <tr> <td>31</td> <td>24 23</td> <td>16 15</td> <td>8 7</td> <td>0</td> </tr> <tr> <td>第1个字</td> <td>第2个字</td> <td>第3个字</td> <td>第4个字</td> <td></td> </tr> </table>	31	24 23	16 15	8 7	0	第1个字	第2个字	第3个字	第4个字		001	<table border="1"> <tr> <td>31</td> <td>16 15</td> <td>0</td> </tr> <tr> <td colspan="2">第1个字</td> <td>第2个字</td> </tr> </table>	31	16 15	0	第1个字		第2个字	010	<table border="1"> <tr> <td>31</td> <td>18 17</td> <td>0</td> </tr> <tr> <td>无效</td> <td>有效</td> <td></td> </tr> </table>	31	18 17	0	无效	有效		011	<table border="1"> <tr> <td>31</td> <td>20 19</td> <td>0</td> </tr> <tr> <td>无效</td> <td>有效</td> <td></td> </tr> </table>	31	20 19	0	无效	有效		100	<table border="1"> <tr> <td>31</td> <td>22 21</td> <td>0</td> </tr> <tr> <td>无效</td> <td>有效</td> <td></td> </tr> </table>	31	22 21	0	无效	有效		101	<table border="1"> <tr> <td>31</td> <td>24 23</td> <td>0</td> </tr> <tr> <td>无效</td> <td>有效</td> <td></td> </tr> </table>	31	24 23	0	无效	有效		110	<table border="1"> <tr> <td>31</td> <td>0</td> </tr> <tr> <td colspan="2">有效</td> </tr> </table>	31	0	有效	
DWL[2:0]	SSITDR/SSIRDR[31:0]																																																																																																																											
000	<table border="1"> <tr> <td>31</td> <td>24 23</td> <td>16 15</td> <td>8 7</td> <td>0</td> </tr> <tr> <td colspan="2">第4个字</td> <td colspan="2">第3个字</td> <td>第1个字</td> </tr> </table>	31	24 23	16 15	8 7	0	第4个字		第3个字		第1个字																																																																																																																	
31	24 23	16 15	8 7	0																																																																																																																								
第4个字		第3个字		第1个字																																																																																																																								
001	<table border="1"> <tr> <td>31</td> <td>16 15</td> <td>0</td> </tr> <tr> <td colspan="2">第2个字</td> <td>第1个字</td> </tr> </table>	31	16 15	0	第2个字		第1个字																																																																																																																					
31	16 15	0																																																																																																																										
第2个字		第1个字																																																																																																																										
010	<table border="1"> <tr> <td>31</td> <td>14 13</td> <td>0</td> </tr> <tr> <td colspan="2">有效</td> <td>无效</td> </tr> </table>	31	14 13	0	有效		无效																																																																																																																					
31	14 13	0																																																																																																																										
有效		无效																																																																																																																										
011	<table border="1"> <tr> <td>31</td> <td>12 11</td> <td>0</td> </tr> <tr> <td colspan="2">有效</td> <td>无效</td> </tr> </table>	31	12 11	0	有效		无效																																																																																																																					
31	12 11	0																																																																																																																										
有效		无效																																																																																																																										
100	<table border="1"> <tr> <td>31</td> <td>10 9</td> <td>0</td> </tr> <tr> <td colspan="2">有效</td> <td>无效</td> </tr> </table>	31	10 9	0	有效		无效																																																																																																																					
31	10 9	0																																																																																																																										
有效		无效																																																																																																																										
101	<table border="1"> <tr> <td>31</td> <td>8 7</td> <td>0</td> </tr> <tr> <td colspan="2">有效</td> <td>无效</td> </tr> </table>	31	8 7	0	有效		无效																																																																																																																					
31	8 7	0																																																																																																																										
有效		无效																																																																																																																										
110	<table border="1"> <tr> <td>31</td> <td>0</td> </tr> <tr> <td colspan="2">有效</td> </tr> </table>	31	0	有效																																																																																																																								
31	0																																																																																																																											
有效																																																																																																																												
DWL[2:0]	SSITDR/SSIRDR[31:0]																																																																																																																											
000	<table border="1"> <tr> <td>31</td> <td>24 23</td> <td>16 15</td> <td>8 7</td> <td>0</td> </tr> <tr> <td>第1个字</td> <td>第2个字</td> <td>第3个字</td> <td>第4个字</td> <td></td> </tr> </table>	31	24 23	16 15	8 7	0	第1个字	第2个字	第3个字	第4个字																																																																																																																		
31	24 23	16 15	8 7	0																																																																																																																								
第1个字	第2个字	第3个字	第4个字																																																																																																																									
001	<table border="1"> <tr> <td>31</td> <td>16 15</td> <td>0</td> </tr> <tr> <td colspan="2">第1个字</td> <td>第2个字</td> </tr> </table>	31	16 15	0	第1个字		第2个字																																																																																																																					
31	16 15	0																																																																																																																										
第1个字		第2个字																																																																																																																										
010	<table border="1"> <tr> <td>31</td> <td>18 17</td> <td>0</td> </tr> <tr> <td>无效</td> <td>有效</td> <td></td> </tr> </table>	31	18 17	0	无效	有效																																																																																																																						
31	18 17	0																																																																																																																										
无效	有效																																																																																																																											
011	<table border="1"> <tr> <td>31</td> <td>20 19</td> <td>0</td> </tr> <tr> <td>无效</td> <td>有效</td> <td></td> </tr> </table>	31	20 19	0	无效	有效																																																																																																																						
31	20 19	0																																																																																																																										
无效	有效																																																																																																																											
100	<table border="1"> <tr> <td>31</td> <td>22 21</td> <td>0</td> </tr> <tr> <td>无效</td> <td>有效</td> <td></td> </tr> </table>	31	22 21	0	无效	有效																																																																																																																						
31	22 21	0																																																																																																																										
无效	有效																																																																																																																											
101	<table border="1"> <tr> <td>31</td> <td>24 23</td> <td>0</td> </tr> <tr> <td>无效</td> <td>有效</td> <td></td> </tr> </table>	31	24 23	0	无效	有效																																																																																																																						
31	24 23	0																																																																																																																										
无效	有效																																																																																																																											
110	<table border="1"> <tr> <td>31</td> <td>0</td> </tr> <tr> <td colspan="2">有效</td> </tr> </table>	31	0	有效																																																																																																																								
31	0																																																																																																																											
有效																																																																																																																												
8	DEL	0	R/W	<p>串行数据延迟</p> <p>0: 在 SSIWS 和 SSIDATA 之间延迟 1 个时钟周期</p> <p>1: 在 SSIWS 和 SSIDATA 之间无延迟</p>																																																																																																																								

位	位名	初始值	R/W	说明
7 ~ 4	CKDV[3:0]	0000	R/W	<p>串行过采样时钟分频比</p> <p>设定过采样时钟 (AUDIOϕ) 的串行位时钟分频比。当 SCKD 位为 “0” 时，忽视这些位。串行位时钟用于移位寄存器，由 SSISCK 引脚提供。</p> <p>0000: AUDIOϕ</p> <p>0001: AUDIOϕ/2</p> <p>0010: AUDIOϕ/4</p> <p>0011: AUDIOϕ/8</p> <p>0100: AUDIOϕ/16</p> <p>0101: AUDIOϕ/32</p> <p>0110: AUDIOϕ/64</p> <p>0111: AUDIOϕ/128</p> <p>1000: AUDIOϕ/6</p> <p>1001: AUDIOϕ/12*</p> <p>1010: AUDIOϕ/24</p> <p>1011: AUDIOϕ/48*</p> <p>1100: AUDIOϕ/96*</p> <p>1101: 禁止设定</p> <p>1110: 禁止设定</p> <p>1111: 禁止设定</p> <p>【注】 * 只有通道 0 能设定，通道 1 ~ 5 禁止设定。</p>
3	MUEN	0	R/W	<p>静噪允许</p> <p>0: 此模块不处于静噪状态</p> <p>1: 此模块处于静噪状态</p> <p>【注】 在静噪状态下将要输出的串行数据替换为 “0”，但是不停止模块内部的数据传送。发送 FIFO 的数据量逐渐减少，为了防止发送溢出，必须将虚拟数据写到 SSIFTDR。</p>
2	—	0	R	<p>保留位</p> <p>读取值为不定值，写入值总是 “0”。</p>
1	TEN	0	R/W	<p>发送允许</p> <p>0: 禁止发送</p> <p>1: 允许发送</p>
0	REN	0	R/W	<p>接收允许</p> <p>0: 禁止接收</p> <p>1: 允许接收</p>

21.3.2 状态寄存器 (SSISR)

SSISR 由表示此模块运行状态的状态标志以及表示当前通道号和字号的位构成。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—	—	—	—	—	—	—	—
初始值:	不定值	不定值	0	0	0	0	1	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	TCHNO[1:0]	TSWNO	RCHNO[1:0]	RSWNO	IDST		
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	0	0	1	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

【注】* 可读写。如果写“0”，此位就被初始化，但是忽视写“1”。

位	位名	初始值	R/W	说明
31、30	—	不定值	R	保留位 读取值为不定值，写入值总是“0”。
29	TUIRQ	0	R/(W)*	发送下溢错误的中断状态标志 此状态标志表示以低于请求的速率提供发送数据。 与TUIEN位的设定无关，此位被置“1”。要将此位清“0”时，必须写“0”。 当TUIRQ位为“1”并且TUIEN位为“1”时产生中断。 当TUIRQ位为“1”时，表示在发送前没有将发送数据写到SSITDR。因此，有可能多发送1次相同的数据，导致破坏多通道的数据，结果输出错误的SSI数据。 【注】 如果发生下溢错误，就在写下一个数据前发送数据缓冲器的数据。
28	TOIRQ	0	R/(W)*	发送上溢错误的中断状态标志 此状态标志表示以高于请求的速率提供发送数据。 与TOIEN位的设定无关，此位被置“1”。要将此位清“0”时，必须写“0”。 当TOIRQ位为“1”并且TOIEN位为“1”时产生中断。 当TOIRQ位为“1”时，表示在发送FIFO满（TDC位为“H'8”）的状态下写SSIFTDR。因此，有可能丢失数据，导致破坏多通道的数据。
27	RUIRQ	0	R/(W)*	接收下溢错误的中断状态标志 此状态标志表示以低于请求的速率提供接收数据。 与RUIEN位的设定无关，此位被置“1”。要将此位清“0”时，必须写“0”。 当RUIRQ位为“1”并且RUIEN位为“1”时产生中断。 当RUIRQ位为“1”时，表示在接收FIFO空（RDC位为“H'0”）的状态下读SSIFRDR。此时，有可能保存无效的接收数据，导致破坏多通道的数据。
26	ROIRQ	0	R/(W)*	接收上溢错误的中断状态标志 此状态标志表示以高于请求的速率提供接收数据。 与ROIEN位的设定无关，此位被置“1”。要将此位清“0”时，必须写“0”。 当ROIRQ位为“1”并且ROIEN位为“1”时产生中断。 当ROIRQ位为“1”时，表示在将新的未读数据写到SSIRDR前没有读以前的未读数据。因此，有可能丢失数据，导致破坏多通道的数据。 【注】 如果发生上溢错误，就将SSI接口送来的下一个数据重写到数据缓冲器。

位	位名	初始值	R/W	说明
25	IIRQ	1	R	空闲模式的中断状态标志 此状态标志表示此模块是否处于空闲状态。为了能进行查询，与 I IEN 位的设定无关，此位被置“1”。 能通过将 I IEN 位清“0”来屏蔽中断，即使给此位写“0”也无法清除中断。 当 IIRQ 位为“1”并且 I IEN 位为“1”时产生中断。 0: 此模块不处于空闲状态 1: 此模块处于空闲状态
24 ~ 7	—	不定值	R	保留位 读取值为不定值，写入值总是“0”。
6、5	TCHNO [1:0]	00	R	发送通道号 表示当前的通道。 此位表示应将哪个通道的数据写到 SSITDR。如果将数据复制到移位寄存器，此值就会发生变化，与是否写到 SSITDR 无关。 如果 TDM 位或者 CONT 位为“1”，就不能使用这些位。
4	TSWNO	1	R	发送串行字号 表示当前的字号。 此位表示应将哪个系统字写到 SSITDR。如果将数据复制到移位寄存器，此位的值就会发生变化，与是否写到 SSITDR 无关。 如果 TDM 位或者 CONT 位为“1”，就不能使用此位。
3、2	RCHNO [1:0]	00	R	接收通道号 表示当前的通道。 此位表示 SSIRDR 的当前数据是哪个通道的数据。如果通过移位寄存器的传来更新 SSIRDR 的数据，此位的值就会发生变化。 如果 TDM 位或者 CONT 位为“1”，就不能使用这些位。
1	RSWNO	1	R	接收串行字号 表示当前的字号。 此位表示 SSIRDR 的当前数据是哪个系统字。与是否读到 SSIRDR 无关，如果通过移位寄存器的传来更新 SSIRDR 的数据，此位的值就会发生变化。 如果 TDM 位或者 CONT 位为“1”，就不能使用此位。
0	IDST	1	R	空闲模式的状态标志 此状态标志表示串行总线处于停止状态。 如果在 TEN 位或者 REN 位为“1”的状态下串行总线正在运行，就清除此位。 如果 TEN 位和 REN 位都被清除并且当前系统字的通信已结束，此位就自动置“1”。 【注】 在当前的系统字结束前，如果外部设备停止串行总线时钟，此位就不被置位。

【注】 * 可读写。如果写“0”，此位就被初始化，但是忽视写“1”。

21.3.3 发送数据寄存器 (SSITDR)

SSITDR 是保存发送数据的 32 位寄存器。要保存的发送数据从发送 FIFO 数据寄存器自动传送到 SSITDR。

如果有发送请求，就将此寄存器的数据传送到移位寄存器。当数据字长小于 32 位时，根据 SSICR 的 PDTA 控制位的设定进行调整。

CPU 不能读写 SSITDR。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
初始值:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

21.3.4 接收数据寄存器 (SSIRDR)

SSIRDR 是保存接收数据的 32 位寄存器。将 SSIRDR 保存的接收数据自动传送到接收 FIFO 数据寄存器。

每当接收到数据字时，从移位寄存器传送此寄存器的数据。当数据字长小于 32 位时，根据 SSICR 的 PDTA 控制位的设定进行调整。

CPU 不能读写 SSIRDR。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
初始值:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

21.3.5 FIFO 控制寄存器 (SSIFCR)

SSIFCR 是设定发送 / 接收 FIFO 数据寄存器的数据触发个数、复位和中断请求允许的寄存器。CPU 能随时读写 SSIFCR。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TTRG[1:0]	RTRG[1:0]	TIE	RIE	TFRST	RFRST		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 8	—	全 0	R	保留位 读写值都为“0”。
7、6	TTRG[1:0]	00	R/W	发送数据量的触发 在发送时, 设定作为 FIFO 状态寄存器 (SSIFSR) 的 TDE 标志置位基准的发送数据量 (指定的发送触发个数)。如果发送 FIFO 数据寄存器 (SSIFTDR) 保存的发送数据量小于等于以下所示的设定触发个数, 就将 TDE 标志置“1”。 00: 7 (1) * 01: 6 (2) * 10: 4 (4) * 11: 2 (6) * 【注】 * () 内的数值表示 TDE 标志被置“1”时的 SSIFTDR 寄存器的空的段数。
5、4	RTRG[1:0]	00	R/W	接收数据量的触发 在接收时, 设定作为 FIFO 状态寄存器 (SSIFSR) 的 RDF 标志置位基准的接收数据量 (指定的接收触发个数)。如果接收 FIFO 数据寄存器 (SSIFRDR) 保存的接收数据量大于等于以下所示的设定触发个数, 就将 RDF 标志置“1”。 00: 1 01: 2 10: 4 11: 6
3	TIE	0	R/W	发送中断的允许 在发送过程中, 因将串行发送数据从发送 FIFO 数据寄存器 (SSIFTDR) 传送到发送数据寄存器 (SSITDR) 并且发送 FIFO 数据寄存器的数据量小于指定的发送触发个数而 FIFO 状态寄存器 (SSIFSR) 的 TDE 标志被置“1”时, 允许或者禁止发送数据空中断 (TXI) 请求的产生。 0: 禁止发送数据空中断 (TXI) 请求 1: 允许发送数据空中断 (TXI) 请求 * 【注】 * 能通过将 TDE 标志清“0” (详细内容请参照 TDE 位的说明) 或者将 TIE 清“0”来解除 TXI。

位	位名	初始值	R/W	说明
2	RIE	0	R/W	<p>接收中断的允许</p> <p>在接收过程中 FIFO 状态寄存器 (SSIFSR) 的 RDF 标志被置“1”时, 允许或者禁止接收数据满中断 (RXI) 请求的产生。</p> <p>0: 禁止接收数据满中断 (RXI) 请求 1: 允许接收数据满中断 (RXI) 请求 *</p> <p>【注】 * 能通过将 RDF 标志清“0” (详细内容请参照 RDF 位的说明) 或者将 RIE 位清“0”来清除 RXI 中断请求。</p>
1	TFRST	0	R/W	<p>发送 FIFO 数据寄存器的复位</p> <p>将发送 FIFO 数据寄存器的数据置为无效并且将该寄存器复位到空状态。</p> <p>0: 禁止复位 * 1: 允许复位</p> <p>【注】 * 在上电复位时进行复位。</p>
0	RFRST	0	R/W	<p>接收 FIFO 数据寄存器的复位</p> <p>将接收 FIFO 数据寄存器的数据置为无效并且将该寄存器复位到空状态。</p> <p>0: 禁止复位 * 1: 允许复位</p> <p>【注】 * 在上电复位时进行复位。</p>

21.3.6 FIFO 状态寄存器 (SSIFSR)

SSIFSR 由表示发送 / 接收 FIFO 数据寄存器运行状态的状态标志构成。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	—	—	—	—	TDC[3:0]				—	—	—	—	—	—	—	—	TDE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/(W)*	
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	—	—	—	—	RDC[3:0]				—	—	—	—	—	—	—	—	RDF
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/(W)*	

位	位名	初始值	R/W	说明
31 ~ 28	—	全 0	R	保留位 读写值都为 “0”。
27 ~ 24	TDC[3:0]	0000	R	表示 SSIFTDR 保存的数据量。 H'0 表示没有发送数据， H'8 表示 32 字节发送数据保存在 SSIFTDR。
23 ~ 17	—	全 0	R	保留位 读写值都为 “0”。
16	TDE	1	R/(W)*	发送数据空 表示将数据从发送 FIFO 数据寄存器 (SSIFTDR) 传送到发送数据寄存器 (SSITDR)， SSIFTDR 的数据量小于 FIFO 控制寄存器 (SSIFCR) 的 TTRG[1:0] 指定的发送触发个数并且允许将发送数据写到 SSIFTDR。 0: 表示 SSIFTDR 的发送数据量大于指定的发送触发个数 [清除条件] • 当将大于指定发送触发个数的数据写到 SSIFTDR 并且给 TDE 写 “0” 时 • 当通过发送数据空中断 (TXI) 启动直接存储器存取控制器并且将大于指定发送触发个数的数据写到 SSIFTDR 时 1: 表示 SSIFTDR 的发送数据量小于等于指定的发送触发个数 *1 [置位条件] • 上电复位 • 当 SSIFTDR 保存的发送数据量小于等于指定的发送触发个数时 【注】 *1 因为 SSIFTDR 是 8 段 FIFO 寄存器，所以在 TDE 位为 “1” 的状态下能写的最大数据量为 “8 减去指定的发送触发个数”。即使想写更多的数据，也忽视此数据。用 SSIFSR 的 TDC 位表示 SSIFTDR 的数据量。
15 ~ 12	—	全 0	R	保留位 读写值都为 “0”。
11 ~ 8	RDC[3:0]	0000	R	表示 SSIFRDR 保存的数据量。 H'0 表示没有接收数据， H'8 表示 32 字节接收数据保存在 SSIFRDR。
7 ~ 1	—	全 0	R	保留位 读写值都为 “0”。

位	位名	初始值	R/W	说明
0	RDF	0	R/(W)*	<p>接收数据满</p> <p>表示将接收数据传送到接收 FIFO 数据寄存器 (SSIFRDR) 并且 SSIFRDR 的数据量大于 FIFO 控制寄存器 (SSIFCR) 的 RTRG[1:0] 指定的接收触发个数。</p> <p>0: 表示 SSIFRDR 的接收数据量小于指定的接收触发个数 [清除条件]</p> <ul style="list-style-type: none"> • 上电复位 • 在通过给 RFRST 写 “1” 使接收 FIFO 变空后给 RDF 写 “0” 时 • 当读 SSIFRDR 使 SSIFRDR 的接收数据量小于指定的接收触发个数并且给 RDF 写 “0” 时 • 当通过接收数据满中断 (RXI) 启动直接存储器存取控制器并且读 SSIFRDR 使 SSIFRDR 的接收数据量小于指定的接收触发个数时 <p>1: 表示 SSIFRDR 的接收数据量大于等于指定的接收触发个数 [置位条件]</p> <ul style="list-style-type: none"> • 当将大于等于指定接收触发个数的接收数据量保存到 SSIFRDR 时 *1 <p>【注】 *1 因为 SSIFRDR 是 8 段 FIFO 寄存器, 所以在 RDF 位为 “1” 时能读到的最大数据量为指定的接收触发个数。如果在读 SSIFRDR 的全部数据后还继续读, 数据就为不定值。用 SSIFSR 的 RDC 位表示 SSIFRDR 的数据量。</p>

【注】 * 可读写。如果写 “0”, 此位就被初始化, 但是忽视写 “1”。

21.3.7 发送 FIFO 数据寄存器 (SSIFTDR)

SSIFTDR 是保存串行发送数据的 32 位 8 段 FIFO 寄存器。如果检测到发送数据寄存器 (SSITDR) 为空, SSI 就在将写到 SSIFTDR 的发送数据传送到 SSITDR 后开始串行发送。能在 SSIFTDR 的发送数据变空前连续进行串行发送。CPU 能随时写 SSIFTDR。

如果 SSIFTDR 发送数据满 (32 字节), 就不能写下一个数据。即使写也被忽视, 并且发生上溢。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
初始值:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

【注】 * 不能在 FIFO 接收运行时写数据。

21.3.8 接收 FIFO 数据寄存器 (SSIFRDR)

SSIFRDR 是保存接收串行数据的 32 位 8 段 FIFO 寄存器。一旦 4 字节串行数据的接收结束，此模块就将接收的串行数据从接收数据寄存器 (SSIRDR) 传送到 SSIFRDR，保存并结束接收。能在结束 32 字节数据的保存前连续接收。CPU 能读但不能写 SSIFRDR。如果在接收 FIFO 数据寄存器没有接收数据的状态下读数据，读取值就为不定值，并且发生接收下溢。

如果 SSIFRDR 的接收数据满，就会丢失以后接收的串行数据并且发生接收上溢。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
初始值:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

21.3.9 TDM 模式寄存器 (SSITDMR)

SSITDMR 是 32 位可读写寄存器，用于设定 TDM 模式和 WS 继续模式。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CONT	—	—	—	—	—	—	—	TDM
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	CONT	0	R/W	WS 继续模式 0: 禁止 WS 继续模式 1: 允许 WS 继续模式 【注】 只能在主控模式 (SCKD 位为“1”并且 SWSD 位为“1”) 中进行设定。
7 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	TDM	0	R/W	TDM 模式 0: 禁止 TDM 模式 1: 允许 TDM 模式

21.4 运行说明

21.4.1 总线格式

此模块能用作发送器或者接收器，无论在何种模式中都能使用多种串行总线格式。
能从表 21.3 所示的 12 个主要模式中选择总线格式。

表 21.3 总线格式

	TEN	REN	SCKD	SWSD	TDM	MUEN	IEN	TOIEN	TUIEN	ROIEN	RUIEN	CONT	SWSP	DEL	PDTA	SDTA	SPDP	SCKP	SWL[2:0]	DWL[2:0]	CHNL[1:0]	
非压缩从属接收器	0	1	0	0	0	控制位						配置位										
非压缩从属发送器	1	0	0	0	0																	
非压缩从属收发器	1	1	0	0	0																	
非压缩主控接收器	0	1	1	1	0																	
非压缩主控发送器	1	0	1	1	0																	
非压缩主控收发器	1	1	1	1	0																	
TDM 从属接收器	0	1	0	0	1							0	配置位									
TDM 从属发送器	1	0	0	0	1							0										
TDM 从属收发器	1	1	0	0	1							0										
TDM 主控接收器	0	1	1	1	1							0										
TDM 主控发送器	1	0	1	1	1							0										
TDM 主控收发器	1	1	1	1	1							0										

21.4.2 非压缩模式

非压缩模式支持被分到各通道的串行音频流。不仅支持 I²S 兼容格式，而且支持多个改良版格式。

(1) 从属接收器

在此模式中，能从其他设备接收串行数据。外部设备提供用于串行数据流的时钟和字选择信号。如果这些信号和此模块的设定格式不同，就不保证运行。

(2) 从属发送器

在此模式中，能将串行数据发送到其他设备。外部设备提供用于串行数据流的时钟和字选择信号。如果这些信号和此模块的设定格式不同，就不保证运行。

(3) 从属收发器

在此模式中，能和其他设备进行串行数据的发送和接收。外部设备提供用于串行数据流的时钟和字选择信号。如果这些信号和此模块的设定格式不同，就不保证运行。

(4) 主控接收器

在此模式中，能从其他设备接收串行数据。在内部由过采样时钟生成时钟和字选择信号。这些信号的格式取决于此模块的设定。如果从其他设备发送的数据和设定的格式不同，就不保证运行。

(5) 主控发送器

在此模式中，能将串行数据发送到其他设备。在内部由过采样时钟生成时钟和字选择信号。这些信号的格式取决于此模块配置位的设定。

(6) 主控收发器

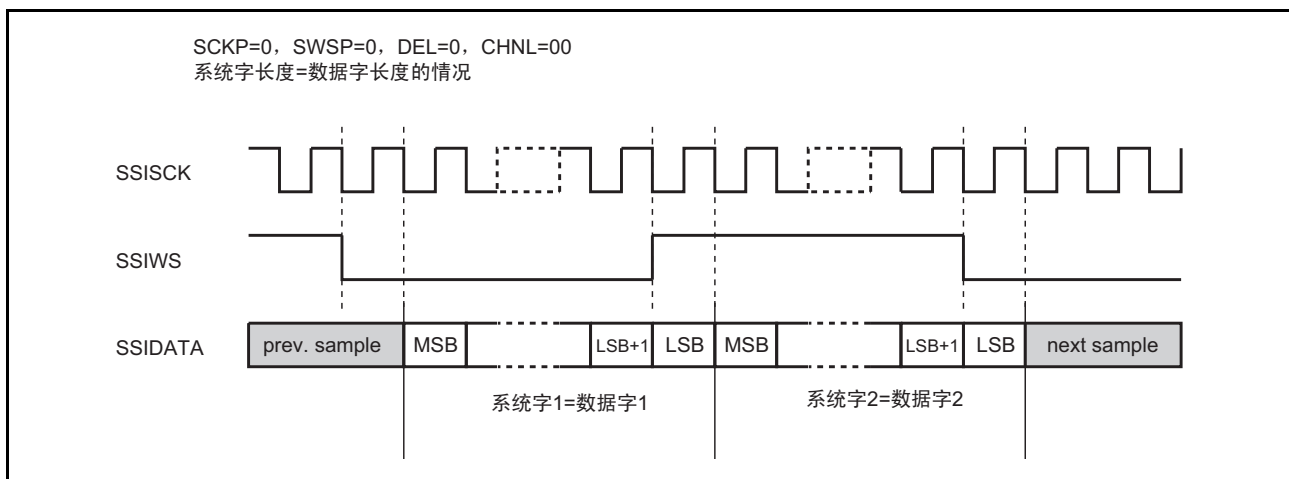
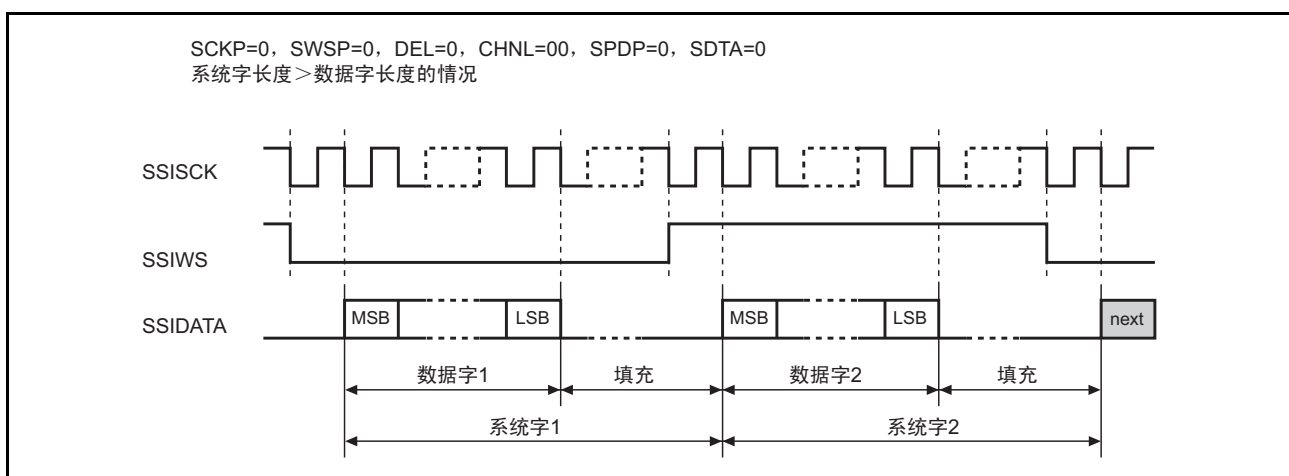
在此模式中，能和其他设备进行串行数据的发送和接收。在内部由过采样时钟生成时钟和字选择信号。这些信号的格式取决于此模块配置位的设定。

(7) 运行设定和字长的相关性

在非压缩模式中，与 SSICR 的字长有关的位全部有效。此模块能支持多个配置，但是在此只说明 I²S 兼容格式、MSB first 和向左靠紧格式以及 MSB first 和向右靠紧格式。

- I²S 兼容格式

无填充和有填充的 I²S 兼容格式分别如图 21.2 和图 21.3 所示。在数据字长度短于系统字长度时发生填充。

图 21.2 I²S 兼容格式（无填充）图 21.3 I²S 兼容格式（有填充）

MSB first 和向左靠紧格式、MSB first 和向右靠紧格式分别如图 21.4 和图 21.5 所示。

- MSB first和向左靠紧格式

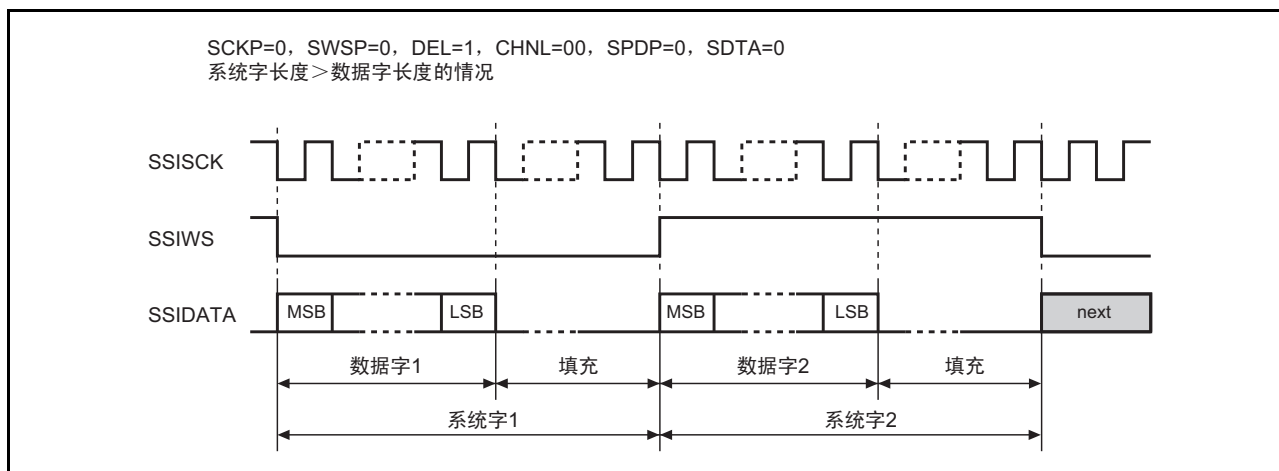


图 21.4 MSB first 和向左靠紧格式（按串行数据、填充位的顺序进行发送和接收）

- MSB first和向右靠紧格式

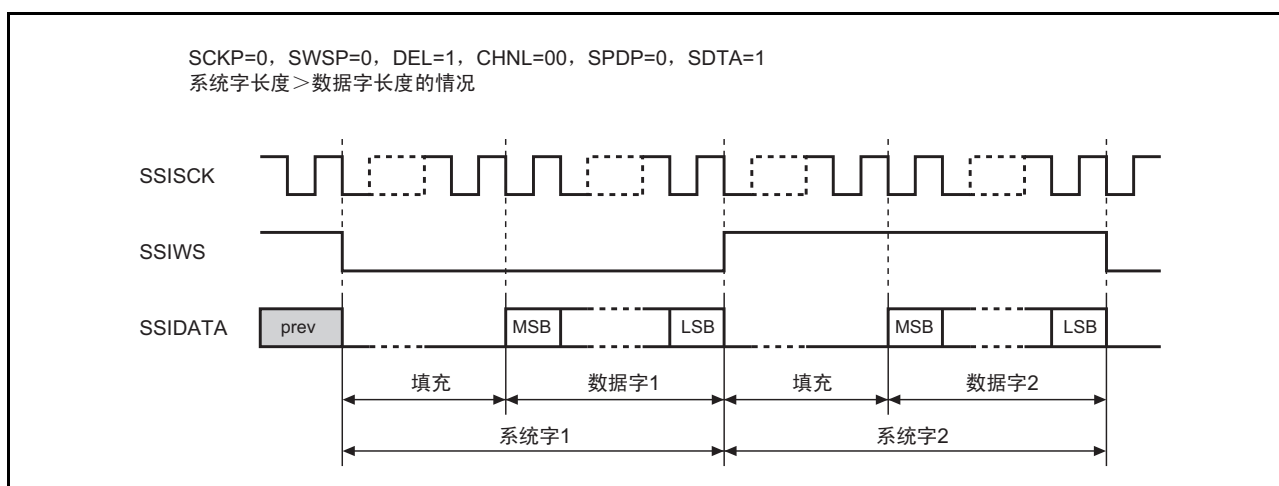


图 21.5 MSB first 和向右靠紧格式（按填充位、串行数据的顺序进行发送和接收）

(8) 多通道格式

有些设备类型扩展了 I²S 总线规格的定义并且在 2 个系统字中进行多于 2 个通道的传送。

此模块使用 CHNL 位、SWL 位和 DWL 位进行 4 个、6 个或者 8 个通道的传送。但是，只限于系统字长度 (SWL) 大于等于数据字长度 (DWL) 乘通道个数 (CHNL) 后的长度的情况。

有效设定和填充位数如表 21.4 所示，无效设定用 “—” 表示而不用数字表示。

表 21.4 有效设定和填充位数

各系统字的填充位数			DWL[2:0]	000	001	010	011	100	101	110
CHNL [1:0]	各系统字的解码通道	SWL [2:0]	解码后的字长	8	16	18	20	22	24	32
00	1	000	8	0	—	—	—	—	—	—
		001	16	8	0	—	—	—	—	—
		010	24	16	8	6	4	2	0	—
		011	32	24	16	14	12	10	8	0
		100	48	40	32	30	28	26	24	16
		101	64	56	48	46	44	42	40	32
		110	128	120	112	110	108	106	104	96
		111	256	248	240	238	236	234	232	224
01	2	000	8	—	—	—	—	—	—	—
		001	16	0	—	—	—	—	—	—
		010	24	8	—	—	—	—	—	—
		011	32	16	0	—	—	—	—	—
		100	48	32	16	12	8	4	0	—
		101	64	48	32	28	24	20	16	0
		110	128	112	96	92	88	84	80	64
		111	256	240	224	220	216	212	208	192
10	3	000	8	—	—	—	—	—	—	—
		001	16	—	—	—	—	—	—	—
		010	24	0	—	—	—	—	—	—
		011	32	8	—	—	—	—	—	—
		100	48	24	0	—	—	—	—	—
		101	64	40	16	10	4	—	—	—
		110	128	104	80	74	68	62	56	32
		111	256	232	208	202	196	190	184	160
11	4	000	8	—	—	—	—	—	—	—
		001	16	—	—	—	—	—	—	—
		010	24	—	—	—	—	—	—	—
		011	32	0	—	—	—	—	—	—
		100	48	16	—	—	—	—	—	—
		101	64	32	0	—	—	—	—	—
		110	128	96	64	56	48	40	32	0
		111	256	224	192	184	176	168	160	128

当此模块用作发送器时，将写到 SSITDR 的各字按写入顺序发送到串行音频总线；当此模块用作接收器时，按从 SSIRDR 接收的顺序读串行音频总线接收的各字。

将 4 个、6 个和 8 个通道的数据传送到串行音频总线的具体例子如图 21.6 ~ 图 21.8 所示，图 21.6 是无填充的例子，图 21.7 是向左靠紧的例子，图 21.8 是向右靠紧的例子。这些例子全部是任意例子。

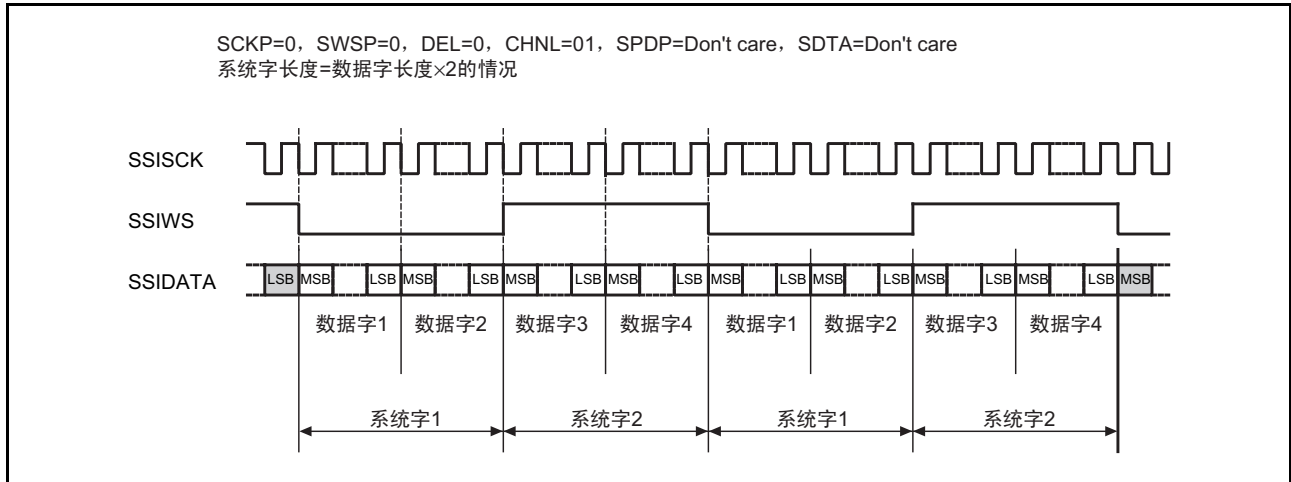


图 21.6 多通道格式（4 个通道，无填充）

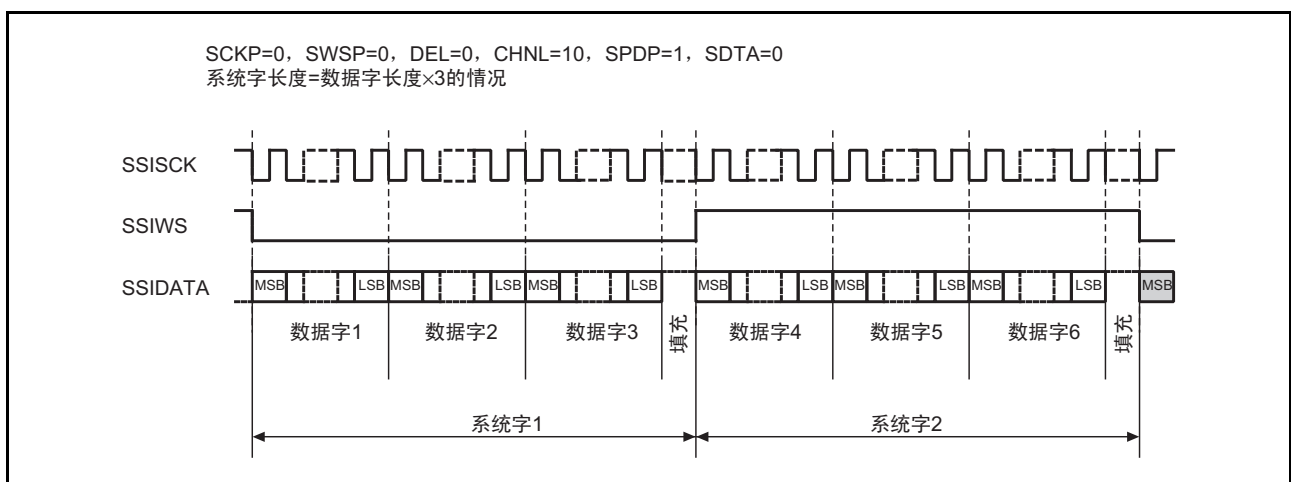


图 21.7 多通道格式（6 个通道，填充 High 电平）

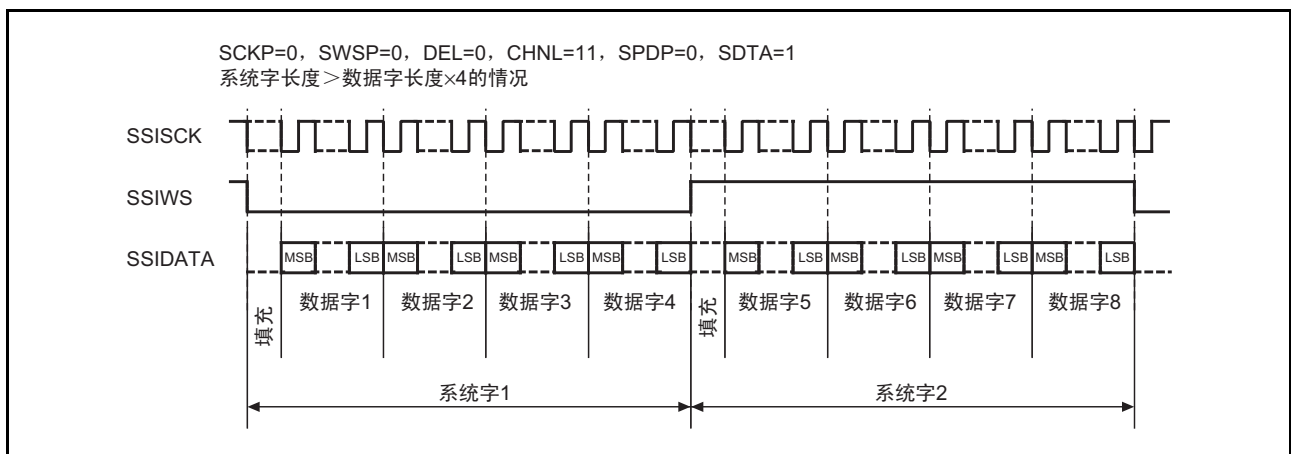


图 21.8 多通道格式（按 8 个通道、填充位、串行数据的顺序进行的发送和接收，有填充）

(9) 运行设定格式的设置

非压缩模式的其他配置位如下所示。这些位不互相排斥，但是根据组合可能有不实用的设定。参照图 21.9 的基本格式例子，对这些配置位进行以下的说明。

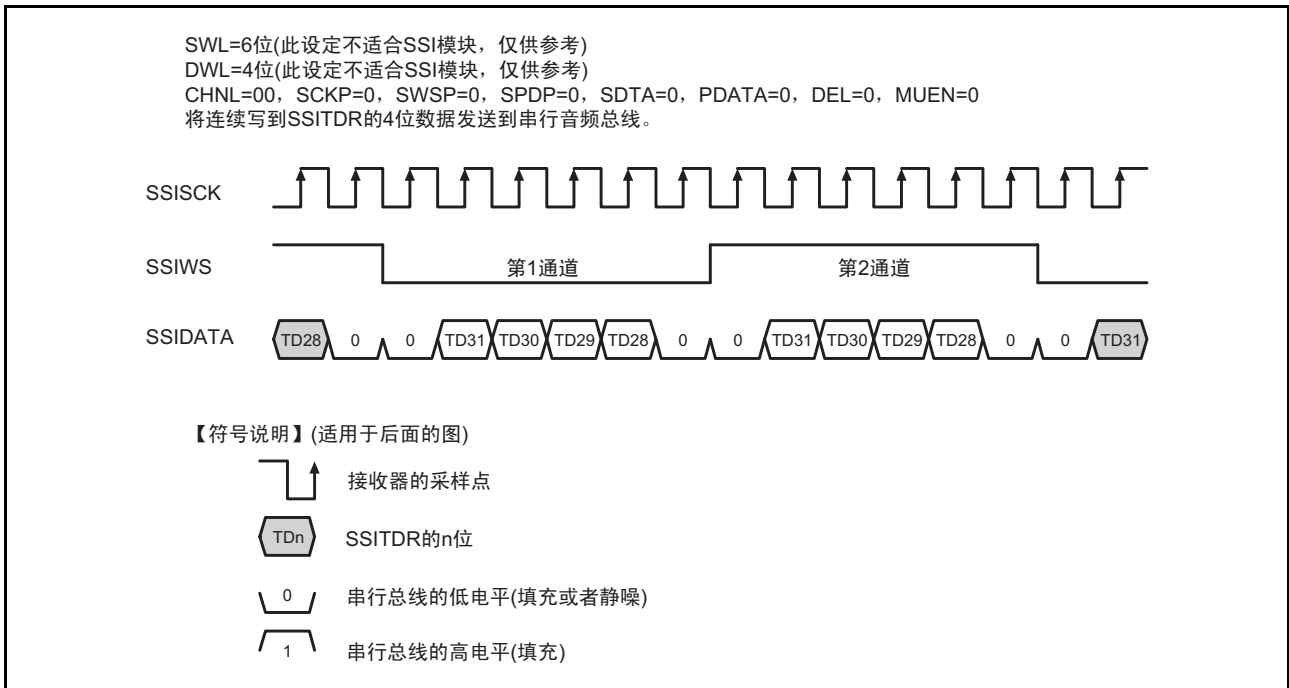


图 21.9 基本格式例子 (发送模式、任意的系统字 / 数据字长度)

图 21.9 的例子使用了 6 位系统字和 4 位数据字。虽然这些设定不可能在此模块中实现，但是为了说明其他设定，在此使用了此例。

- 反相时钟

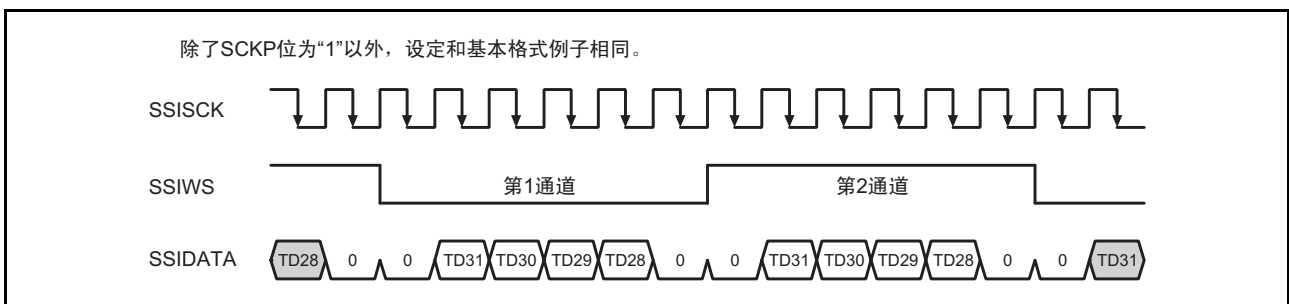


图 21.10 反相时钟

- 反相字选择信号

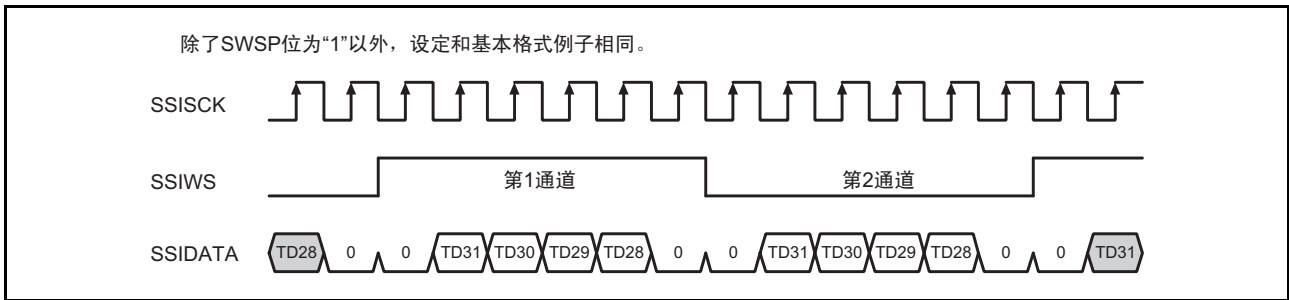


图 21.11 反相字选择信号

- 反相填充极性

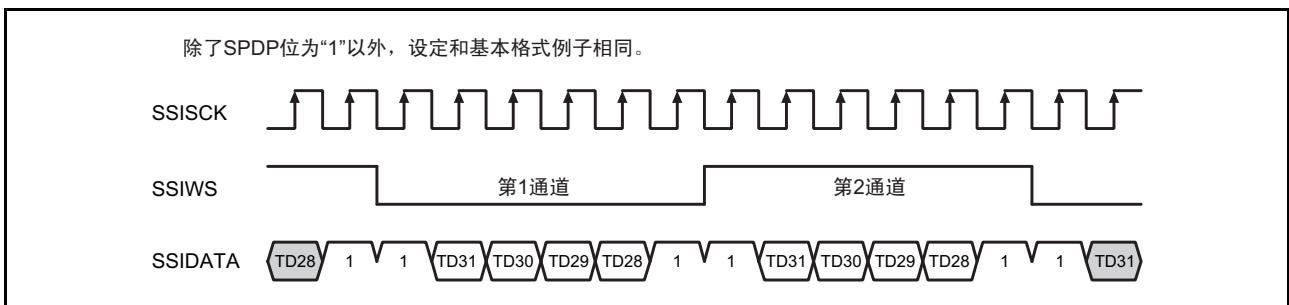


图 21.12 反相填充极性

- 按填充位、串行数据的顺序进行的发送和接收，有延迟

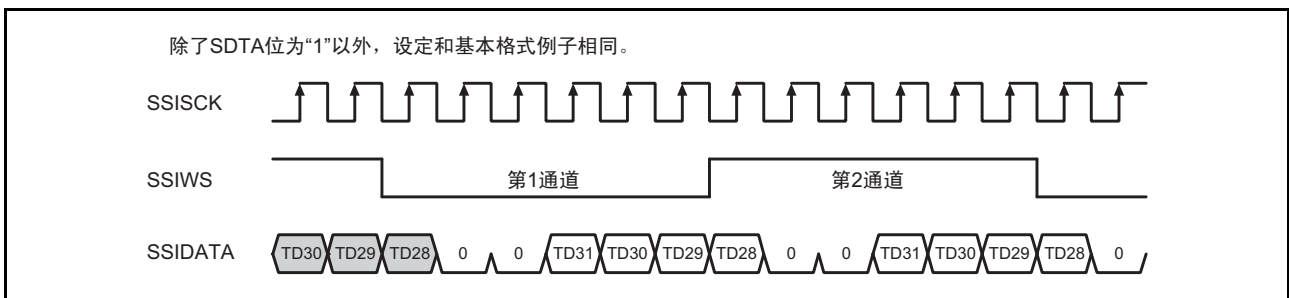


图 21.13 按填充位、串行数据的顺序进行的发送和接收，有延迟

- 按填充位、串行数据的顺序进行的发送和接收，无延迟

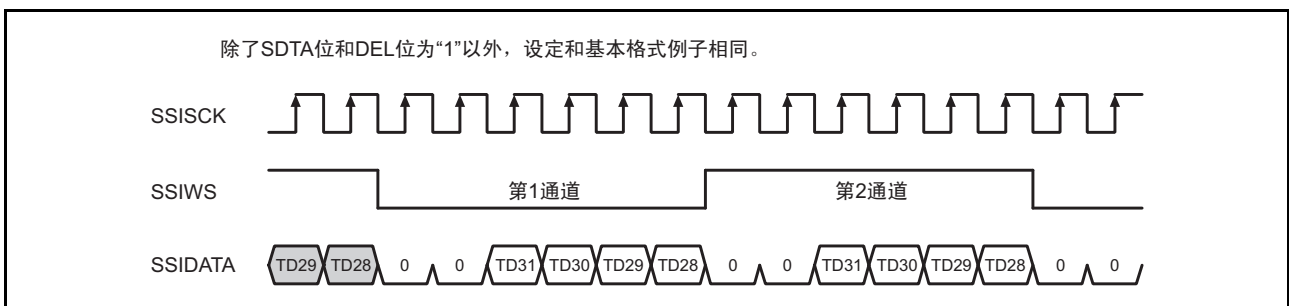


图 21.14 按填充位、串行数据的顺序进行的发送和接收，无延迟

- 按串行数据、填充位的顺序进行的发送和接收，无延迟

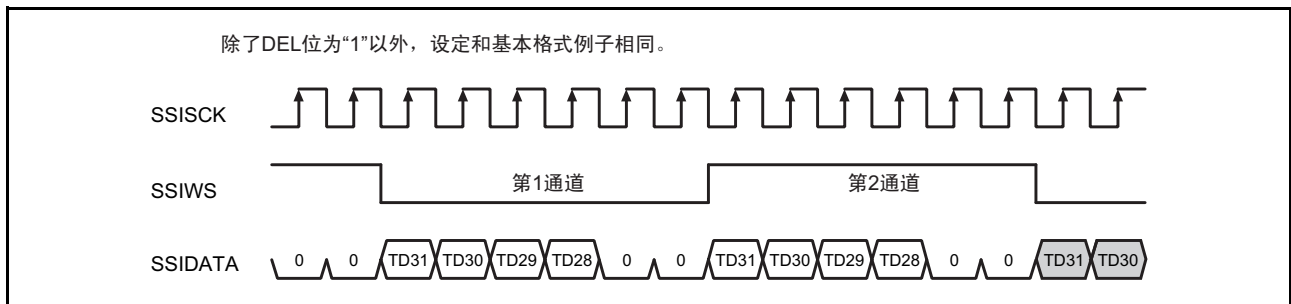


图 21.15 按串行数据、填充位的顺序进行的发送和接收，无延迟

- 并行数据的向右靠紧，有延迟

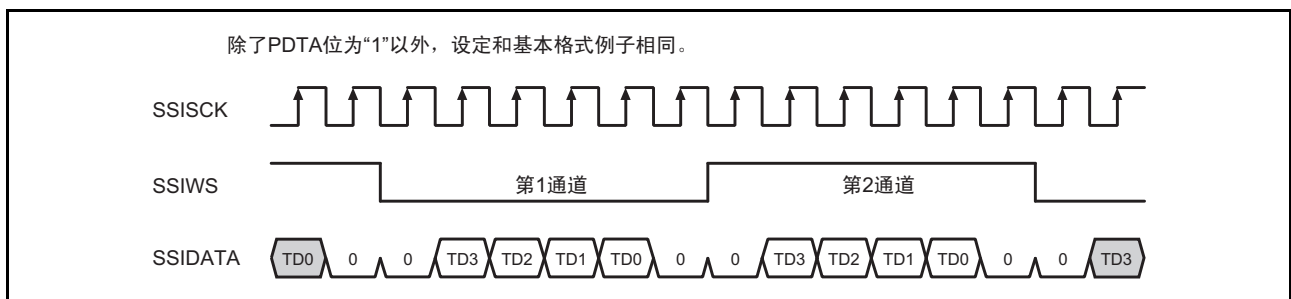


图 21.16 并行数据的向右靠紧，有延迟

- 静噪有效

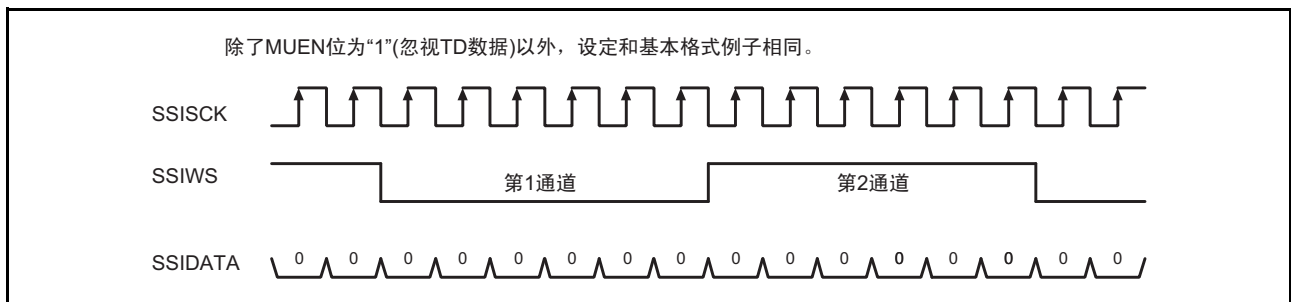


图 21.17 静噪有效

21.4.3 TDM 模式

TDM 模式是连接 TDM 对应的多通道和设备的模式，能通过 TDM 模式寄存器（SSITDMR）的 TDM 位进行设定。一旦设定为此模式，SSIWS 信号就只在系统字 1 的区间为高电平，在其他区间为低电平。将此 SSIWS 信号产生的脉冲定义为 SYNC 脉冲。SYNC 脉冲只有正极性（只在系统字 1 的区间为高电平）。

无填充和有填充的 TDM 格式分别如图 21.18 和图 21.19 所示。

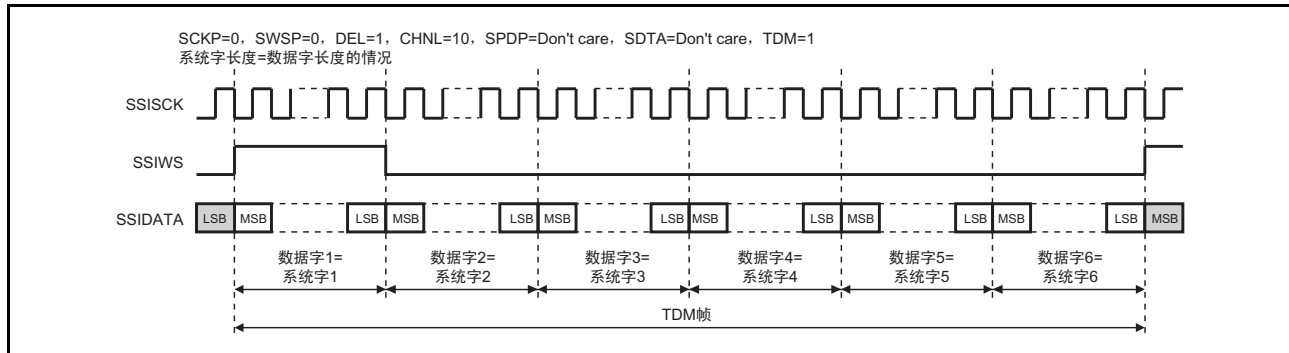


图 21.18 TDM 格式（6 个系统字，无填充）

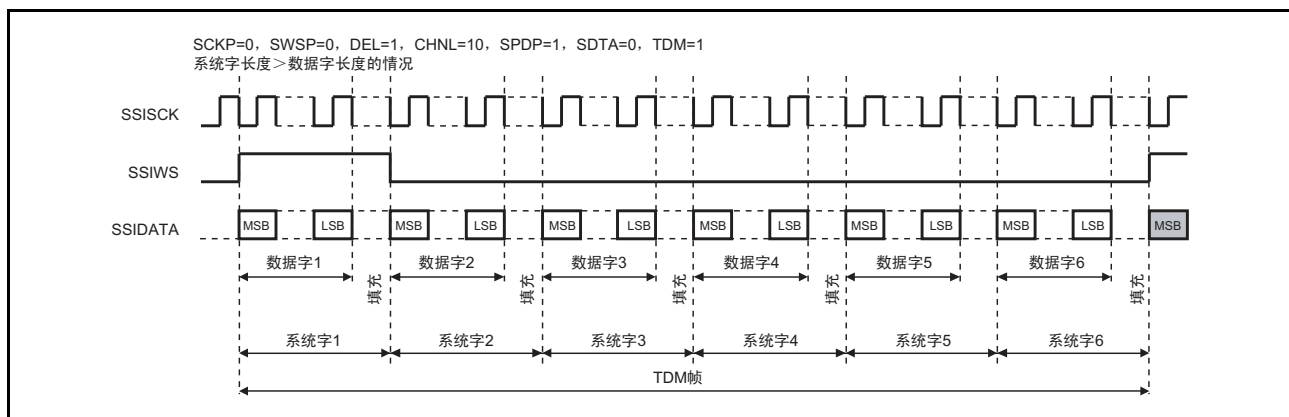


图 21.19 TDM 格式（6 个系统字，有填充）

21.4.4 WS 继续模式

WS 继续模式是继续输出 SSIWS 信号的模式，与允许或者禁止数据传送无关。能通过 TDM 模式寄存器 (SSITDMR) 的 CONT 位设定此模式。在允许此模式的情况下，即使将 SSICR.TEN 位和 SSICR.REN 位置“0”（停止传送），SSIWS 信号也不停止而继续输出；在禁止此模式的情况下，如果将 SSICR.TEN 位或者 SSICR.REN 位置“0”，SSIWS 信号就停止输出。

允许和禁止 WS 继续模式时的运行分别如图 21.20 和图 21.21 所示。

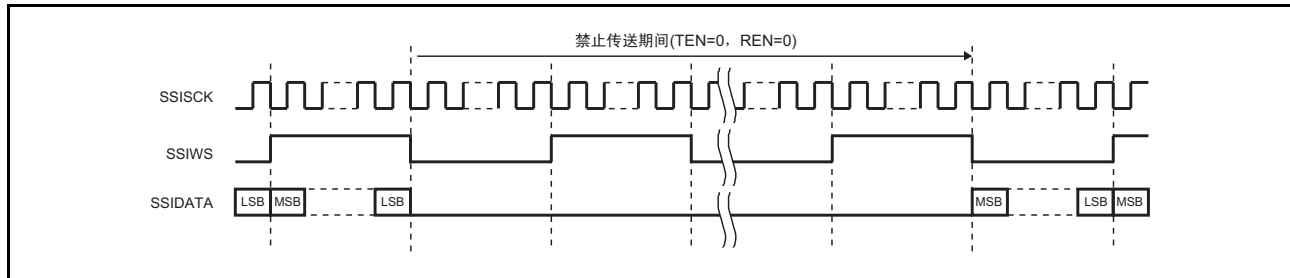


图 21.20 允许 WS 继续模式

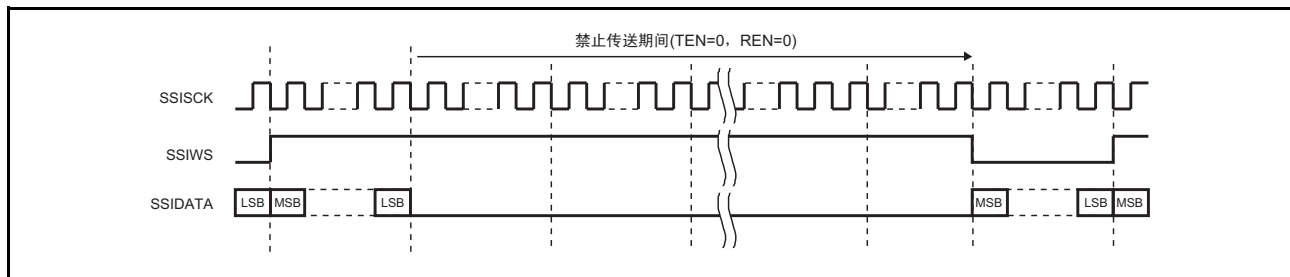


图 21.21 禁止 WS 继续模式

21.4.5 运行模式

运行模式有配置模式、有效模式和无效模式共 3 个，运行模式转移图如图 21.22 所示。

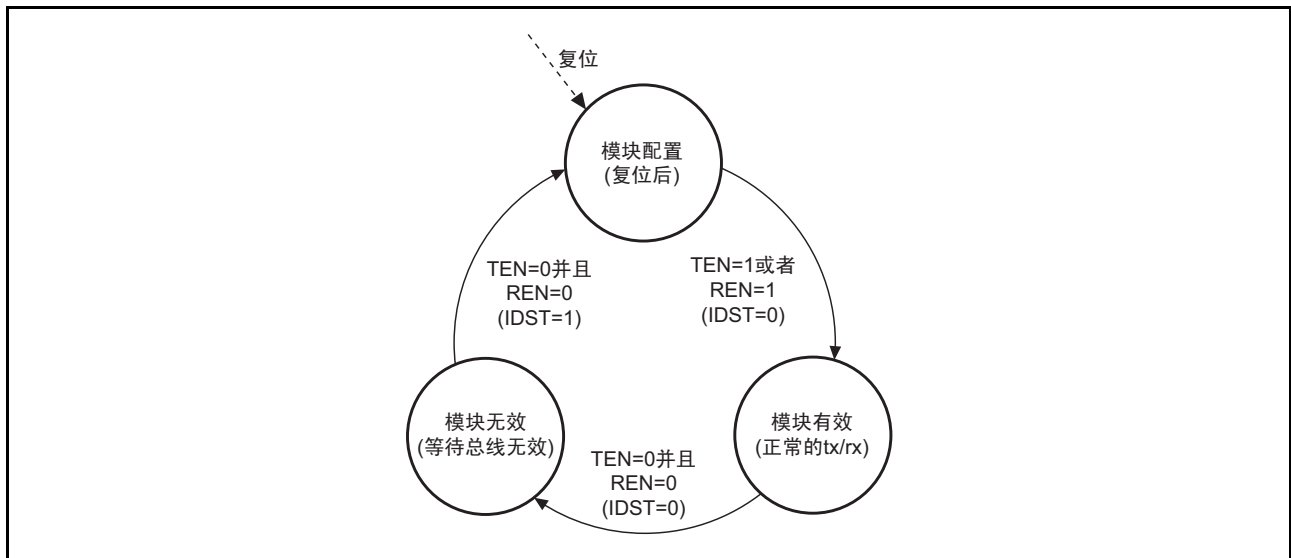


图 21.22 运行模式转移图

(1) 配置模式

在解除复位后进入此模式。在通过将 TEN 位或者 REN 位置位使此模块有效前，需要对此模式的控制寄存器进行设定。

如果将 TEN 位或者 REN 位置位，此模块就转移到模块有效模式。

(2) 模块有效模式

此模式的运行取决于所选择的运行模式，详细内容请参照“21.4.6 发送运行”和“21.4.7 接收运行”。

21.4.6 发送运行

能通过 DMA 传送或者中断来控制发送。

DMA 控制能降低 CPU 的负担。在 DMA 控制模式中，通过中断来通知数据的下溢/上溢或者 DMA 传送结束。

另外的控制方法是使用此模块根据需要为提供数据而生成的中断。

要将此模块置为无效时，必须连续提供时钟*，直到 IIRQ 位表示空闲状态为止。

DMA 控制模式和中断控制模式的发送分别如图 21.23 和图 21.24 所示。

【注】 * 当 SCKD 位为“0”时，提供 SSISCK 引脚的输入时钟。
当 SCKD 位为“1”时，提供过采样时钟。

(1) 使用直接存储器存取控制器的发送

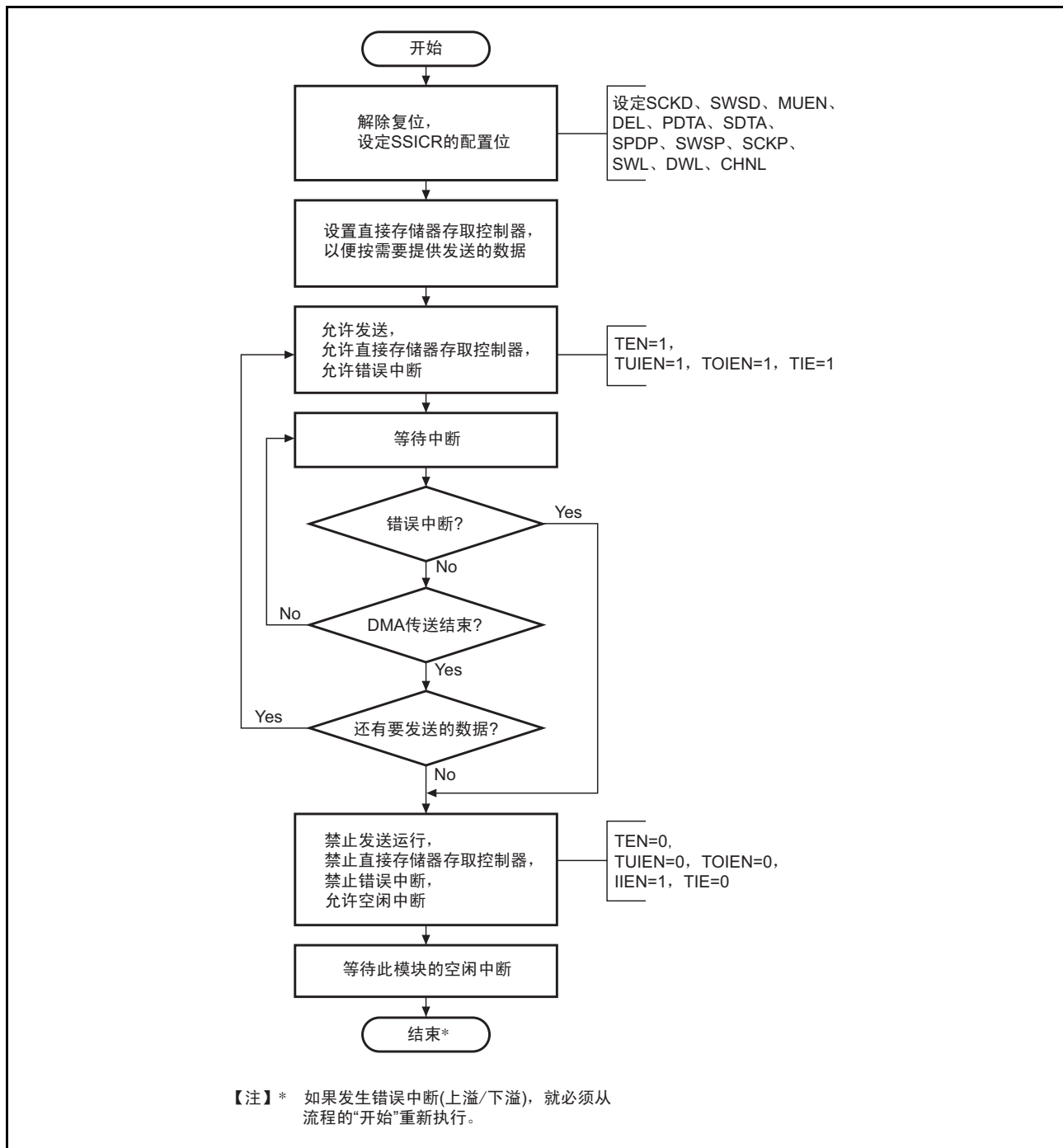


图 21.23 使用直接存储器存取控制器的发送

(2) 使用中断数据流程控制的发送

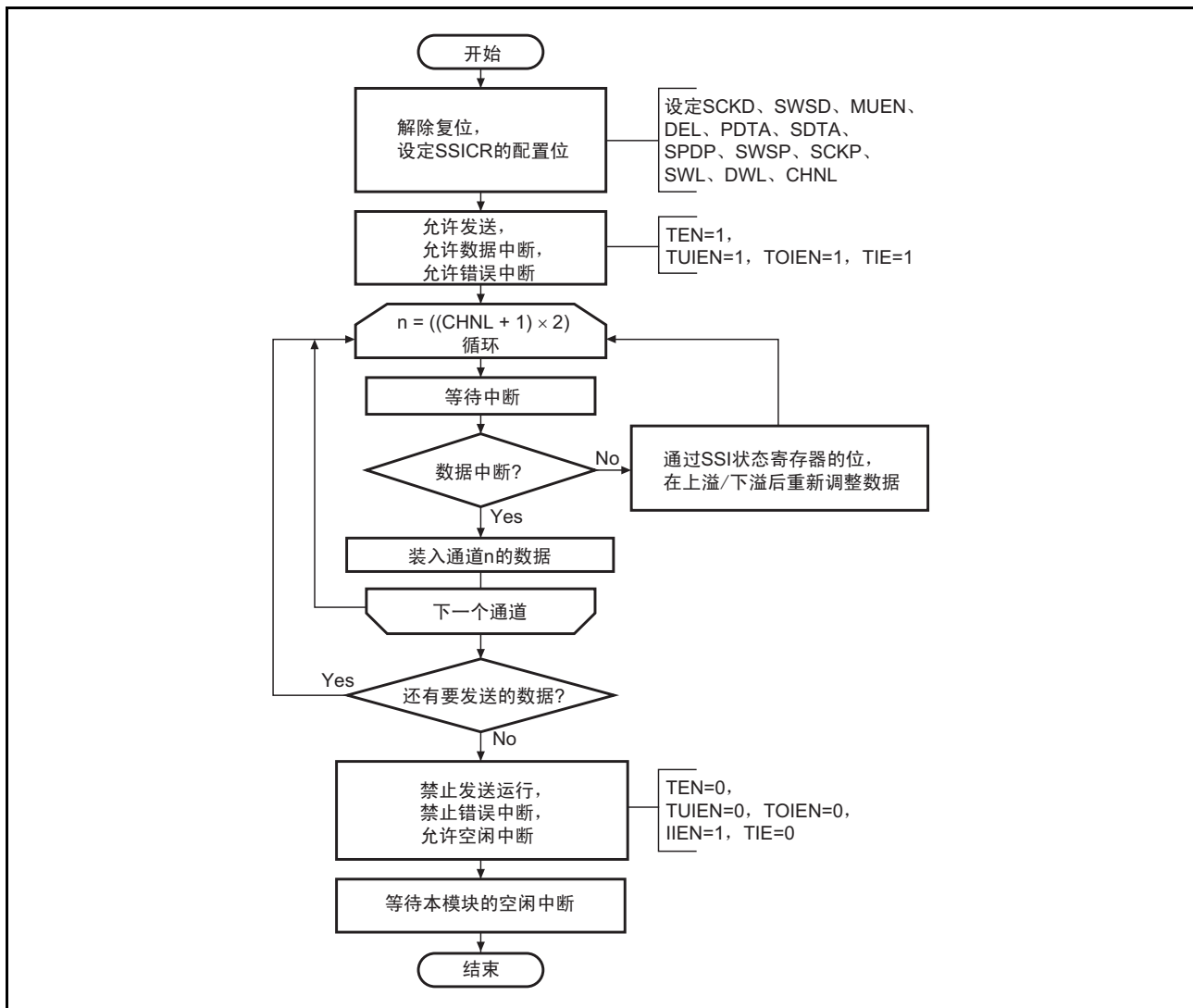


图 21.24 使用中断数据流程控制的发送

21.4.7 接收运行

和发送一样，也能通过 DMA 传送或者中断来控制接收。

各运行流程图分别如图 21.25 和图 21.26 所示。

要将此模块置为无效时，必须继续提供时钟*，直到 IIRQ 位表示空闲状态为止。

【注】* 当 SCKD 位为“0”时，提供 SSISCK 引脚的输入时钟。
当 SCKD 位为“1”时，提供过采样时钟。

(1) 使用直接存储器存取控制器的接收

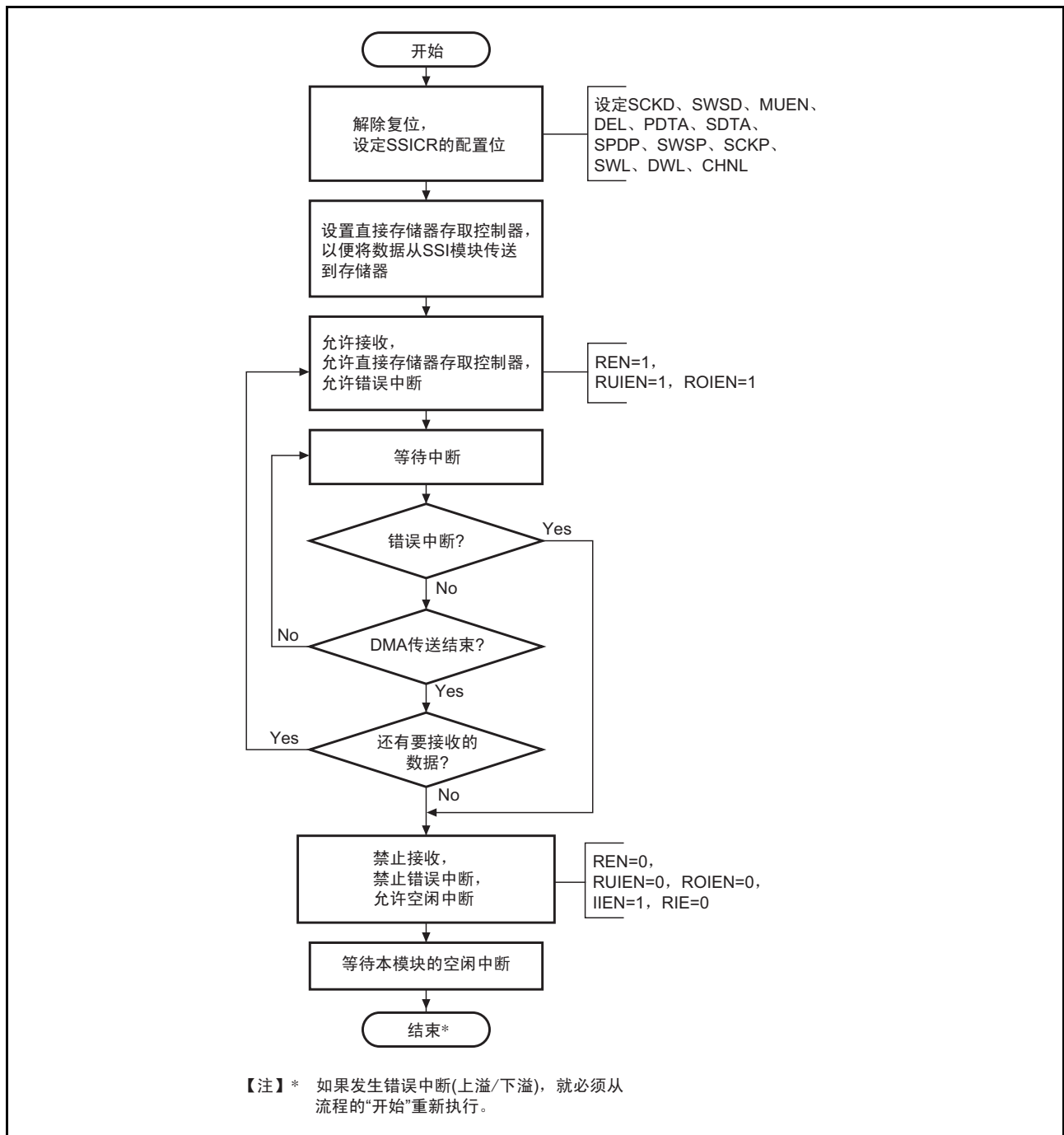


图 21.25 使用直接存储器存取控制器的接收

(2) 使用中断数据流程控制的接收

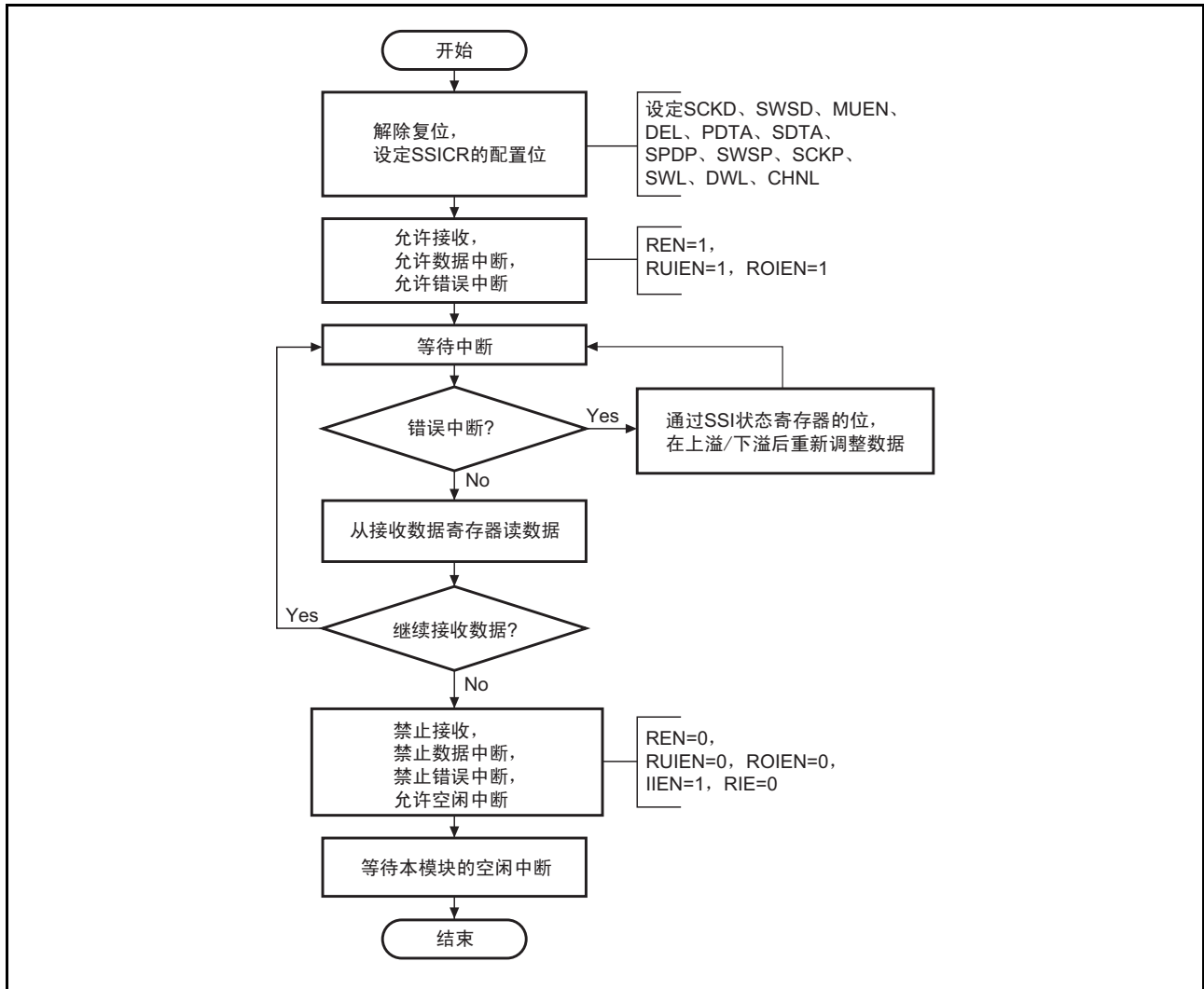


图 21.26 使用中断数据流程控制的接收

如果下溢条件和上溢条件相同，就能在发送时使用 TCHNO[1:0] 位和 TSWNO 位并且在接收时使用 RCHNO[1:0] 位和 RSWNO 位，将此模块恢复到条件相同前的状态。一旦发生下溢或者上溢，主机 CPU 就能通过读通道号和系统字数，得知串行音频流到达的位置。当作为发送器运行时，主机 CPU 能在此模块到达下次发送的预定数据前跳读发送数据，因此能再次和音频数据流同步。当作为接收器运行时，在能保存表示此模块下一次接收的数据前，主机 CPU 能通过保存空数据，取得接收数据量的一致性，再次和音频数据流同步。

21.4.8 串行位时钟控制

使用串行位时钟功能，控制和选择串行总线接口所用的时钟。

如果将串行位时钟方向设定为输入（SCKD 位为“0”），此模块就为时钟从属模式，移位寄存器使用的位时钟为 SSISCK 引脚的输入时钟。

如果将串行位时钟方向设定为输出（SCKD 位为“1”），此模块就为时钟主控模式，移位寄存器使用的位时钟为过采样时钟或者其分频时钟。以 SSICR 的串行过采样时钟分频比（CKDV）位设定的比率将过采样时钟进行分频，分频后的时钟用作移位寄存器的位时钟。

无论上述哪种情况，SSISCK 引脚的输出都和位时钟相同。

21.5 使用时的注意事项

21.5.1 在 DMA 运行过程中发生下溢或者上溢时的限制事项

如果在 DMA 运行过程中发生下溢或者上溢，就需要重新启动模块。发送缓冲器和接收缓冲器由 L 通道和 R 通道共用的 32 位寄存器构成。例如，在将控制寄存器（SSICR）的数据字长度（DWL2 ~ DWL0）设定为 32 位并且将系统字长度（SWL2 ~ SWL0）设定为 32 位时，如果发生下溢或者上溢，应该在 L 通道发送和接收的数据就可能在 R 通道发送和接收。

如果发生发送下溢、发送上溢、接收下溢、接收上溢这 4 种错误中断或者通过这些中断对应的错误状态标志（SSISR 的 TUIRQ、TOIRQ、RUIRQ、ROIQR 的各位）确认到发生错误，就必须给 SSICR 的 TEN 位或者 REN 位写“0”，禁止 DMA 传送请求并且停止运行（此时，也必须停止设定直接存储器存取控制器）。然后，必须给错误状态标志位写“0”，清除错误状态，并且再次设定直接存储器存取控制器，重新开始传送。

21.5.2 进行主控收发器到主控接收器的模式切换时的注意事项

在禁止 WS 继续模式（SSITDMR.CONT 位为“0”）的状态下，如果在主控收发器模式的运行过程中因发生发送下溢并且给 SSICR 的 TEN 位写“0”而停止发送，就暂停 SSIWS 的输出。要在切换为主控接收器模式后不暂停而连续接收时，必须将虚拟数据写到 SSIFTDR，以避免发送下溢的发生。

21.5.3 TDM 模式和 WS 继续模式的限制事项

如果改变 TDM 模式和 WS 继续模式的设定，就不保证改变后的 SSISCK 信号和 SSIWS 信号的动作。如果影响连接的设备，就不能动态改变设定。

22. 带 FIFO 的时钟同步串行 I/O

本 LSI 内置带 FIFO 的时钟同步串行 I/O。

22.1 特点

- 串行传送
FIFO 容量：32 位×16 段（独立发送和接收）
支持 8 位单声道、16 位单声道和 16 位立体声的输入/输出。
从 MSB 开始发送数据（MSB First）。
支持最大 48kHz 的采样率。
同步方法支持帧同步脉冲。
能连接线性、音频、A-Law 和 μ -Law CODEC 芯片。
支持主控模式和从属模式。
- 串行时钟
可从 AUDIO_CLK 和 AUDIO_X1 中选择时钟源。
- 中断：1 种
- DMA 传送：2 种
发送 FIFO 传送请求、接收 FIFO 传送请求

框图如图 22.1 所示。

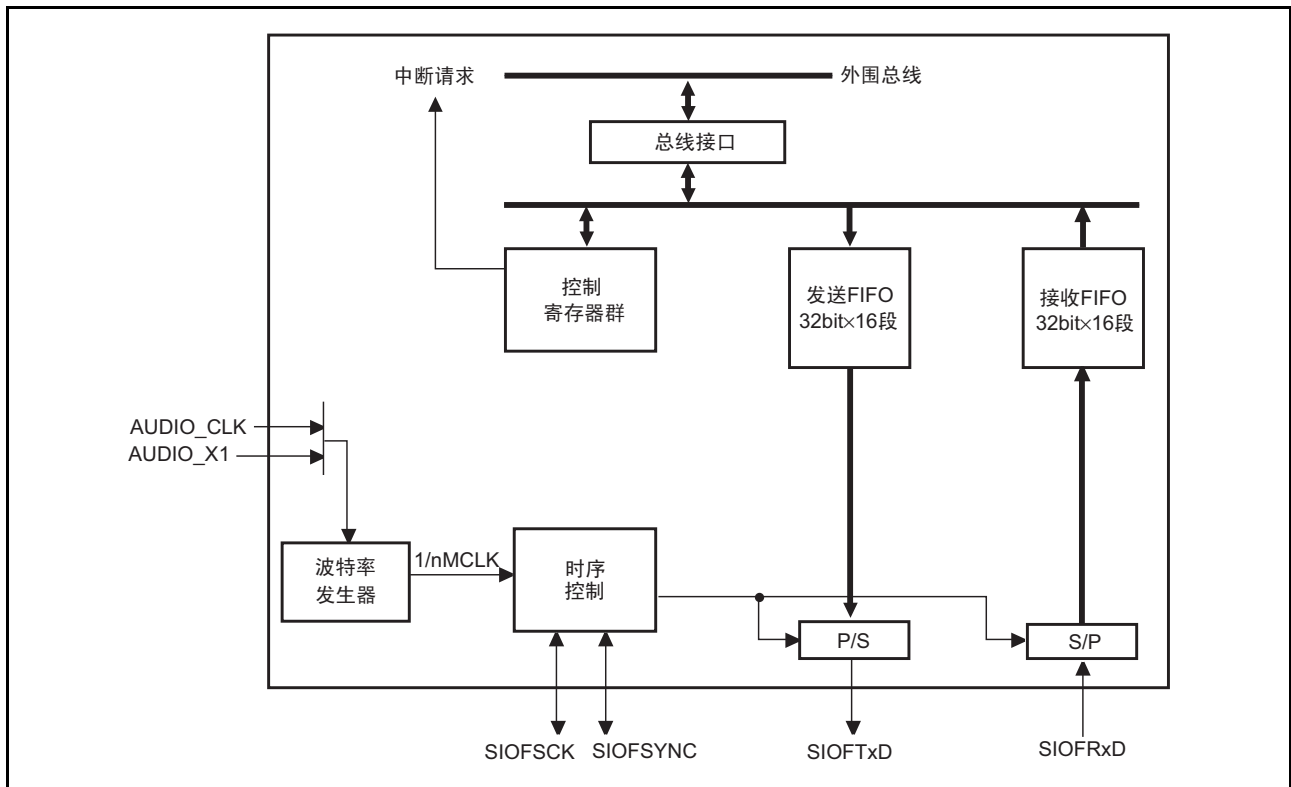


图 22.1 框图

22.2 输入 / 输出引脚

引脚结构如表 22.1 所示。

表 22.1 引脚结构

引脚名	输入 / 输出	功能
AUDIO_CLK	输入	音频外部时钟
AUDIO_X1	输入	音频晶体谐振器 / 外部时钟
AUDIO_X2	输出	
SIOFSCK	输入 / 输出	串行时钟（发送 / 接收共用）
SIOFSYNC	输入 / 输出	帧同步信号（发送 / 接收共用）
SIOFTxD	输出	发送数据
SIOFRxD	输入	接收数据

22.3 寄存器说明

寄存器结构如表 22.2 所示。

表 22.2 寄存器结构

寄存器名	略称	R/W	初始值	地址	存取长度
模式寄存器	SIMDR	R/W	H'8000	H'FFFF4800	16
时钟选择寄存器	SISCR	R/W	H'8000	H'FFFF4802	16
发送数据分配寄存器	SITDAR	R/W	H'0000	H'FFFF4804	16
接收数据分配寄存器	SIRDAR	R/W	H'0000	H'FFFF4806	16
控制寄存器	SICTR	R/W	H'0000	H'FFFF480C	16
FIFO 控制寄存器	SIFCTR	R/W*	H'1000	H'FFFF4810	16
状态寄存器	SISTR	R/W*	H'0000	H'FFFF4814	16
中断允许寄存器	SIIER	R/W	H'0000	H'FFFF4816	16
发送数据寄存器	SITDR	W	不定值	H'FFFF4820	8、16、32
接收数据寄存器	SIRD R	R	不定值	H'FFFF4824	8、16、32

【注】 * 此寄存器的位有读写位和只读位，详细内容请参照各寄存器的章节。

22.3.1 模式寄存器 (SIMDR)

SIMDR 设定此模块的运行模式。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TRMD1	TRMD0	SYNCAT	REDG	FL3	FL2	FL1	FL0	TXDIZ	—	SYNCAC	SYNCDL	—	—	—	—
初始值:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R	R	R	R

位	位名	初始值	R/W	说明
15	TRMD1	1	R/W	传送模式 1、传送模式 0
14	TRMD0	0	R/W	选择传送模式。 00: 从属模式 01: 禁止设定 10: 主控模式 11: 禁止设定
13	SYNCAT	0	R/W	SIOFSYNC 引脚有效时序 表示 SIOFSYNC 信号的输出位置。 此位在主控模式时有效。 0: 帧的起始位数据 1: 槽的最后位数据 【注】 如果将此位置“1”，就必须将有效数据设定为发送/接收或者发送。
12	REDG	0	R/W	接收数据采样边沿 此位在主控模式时有效。 0: 在 SIOFSCK 的下降沿对 SIOFRxD 进行采样 (在 SIOFSCK 的上升沿发送 SIOFTxD) 1: 在 SIOFSCK 的上升沿对 SIOFRxD 进行采样 (在 SIOFSCK 的下降沿发送 SIOFTxD)
11	FL3	0	R/W	帧长 3 ~ 0
10	FL2	0	R/W	00xx: 数据长度为 8 位, 帧长为 8 位。
9	FL1	0	R/W	0100: 数据长度为 8 位, 帧长为 16 位。
8	FL0	0	R/W	0101: 数据长度为 8 位, 帧长为 32 位。 0110: 数据长度为 8 位, 帧长为 64 位。 0111: 数据长度为 8 位, 帧长为 128 位。 10xx: 数据长度为 16 位, 帧长为 16 位。 1100: 数据长度为 16 位, 帧长为 32 位。 1101: 数据长度为 16 位, 帧长为 64 位。 1110: 数据长度为 16 位, 帧长为 128 位。 1111: 数据长度为 16 位, 帧长为 256 位。 【注】 x: Don't care
7	TXDIZ	0	R/W	发送无效时 *SIOFTxD 引脚的输出 0: 无效时输出“1” 1: 无效时为高阻抗状态 【注】 * 无效时是指禁止发送的情况以及输出未分配发送数据或者控制数据的槽的情况。
6	—	0	R	保留位 读写值都为“0”。

位	位名	初始值	R/W	说明
5	SYNCAC	0	R/W	SIOFSYNC 引脚极性 此位在主控模式时有效。 0: 高电平有效 1: 低电平有效
4	SYNCDL	0	R/W	对 SIOFSYNC 引脚的数据引脚位延迟 在从属模式中, 只有 1 位延迟有效。 0: 无位延迟 1: 1 位延迟
3 ~ 0	—	全 0	R	保留位 读写值都为“0”。

22.3.2 控制寄存器 (SICTR)

SICTR 设定此模块的运行状态。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCKE	FSE	—	—	—	—	TXE	RXE	—	—	—	—	—	—	TXRST	RXRST
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15	SCKE	0	R/W	串行时钟的输出允许 此位在主控模式时有效。 0: 禁止输出 SIOFSCK (输出 “0”) 1: 允许输出 SIOFSCK • 如果将此位置 “1”, 此模块就对波特率发生器进行初始化, 并且在开始运行的同时将波特率发生器生成的时钟输出到 SIOFSCK。
14	FSE	0	R/W	帧同步信号的输出允许 此位在主控模式时有效。 0: 禁止输出 SIOFSYNC (输出 “0”) 1: 允许输出 SIOFSYNC • 如果将此位置 “1”, 此模块就对帧计数器进行初始化并且开始运行。
13 ~ 10	—	全 0	R	保留位 读写值都为 “0”。
9	TXE	0	R/W	发送允许 0: 禁止从 SIOFTxD 发送数据 1: 允许从 SIOFTxD 发送数据 • 此位的设定在下一帧开始 (帧同步信号的上升沿) 时有效。 • 如果将此位置 “1”, 此模块就根据 SIFCTR 寄存器的 TFWM 位的设定, 发行发送传送请求。如果将数据保存到发送 FIFO, 就从 SIOFTxD 开始发送数据。 • 在发送复位时, 对此位进行初始化。
8	RXE	0	R/W	接收允许 0: 禁止从 SIOFRxD 接收数据 1: 允许从 SIOFRxD 接收数据 • 此位的设定在下一帧开始 (帧同步信号的上升沿) 时有效。 • 如果将此位置 “1”, 此模块就开始从 SIOFRxD 取接收数据。如果将数据保存到接收 FIFO, 就根据 SIFCTR 的 RFWM 位的设定, 发行接收传送请求。 • 在接收复位时进行初始化。
7 ~ 2	—	全 0	R	保留位 读写值都为 “0”。

位	位名	初始值	R/W	说明
1	TXRST	0	R/W	<p>发送复位</p> <p>0: 不对发送运行进行复位 1: 对发送运行进行复位</p> <ul style="list-style-type: none"> 此位的设定立即有效。在设定被初始化的寄存器前，必须将此位清“0”。 如果将此位置“1”，此模块就将 SIOFTxD 的输出置“1”并且对下述寄存器和数据进行初始化。 <ol style="list-style-type: none"> SITDR 寄存器 发送 FIFO 的有效数据 SISTR 寄存器的 TFEMP 位和 TDREQ 位 TXE 位 <p>【注】 必须将此位为“1”的状态至少保持 1 个传送时钟周期。</p>
0	RXRST	0	R/W	<p>接收复位</p> <p>0: 不对接收运行进行复位 1: 对接收运行进行复位</p> <ul style="list-style-type: none"> 此位的设定立即有效。在设定被初始化的寄存器前，必须将此位清“0”。 如果将此位置“1”，此模块就停止从 SIOFRxD 接收数据并且对下述寄存器和数据进行初始化。 <ol style="list-style-type: none"> SIRDR 寄存器 接收 FIFO 的有效数据 SISTR 寄存器的 RFFUL 位和 RDREQ 位 RXE 位 <p>【注】 必须将此位为“1”的状态至少保持 1 个传送时钟周期。</p>

22.3.3 发送数据寄存器 (SITDR)

SITDR 设定此模块的发送数据。将此寄存器的设定数据保存到发送 FIFO。
在通过 SICTR 的 TXRST 位进行发送复位时，对此寄存器进行初始化。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SITDL[15:0]															
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SITDR[15:0]															
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

位	位名	初始值	R/W	说明
31 ~ 16	SITDL[15:0]	不定值	W	左 ch. 发送数据 将 SIOFTxD 发送的数据设定为左 ch. 数据。发送帧中的左 ch. 数据的位置取决于 SITDAR 的 TDLA 位的设定值。 • 如果将 SITDAR 的 TDLE 位置 “1”，这些位就有效。
15 ~ 0	SITDR[15:0]	不定值	W	右 ch. 发送数据 将 SIOFTxD 发送的数据设定为右 ch. 数据。发送帧中的右 ch. 数据的位置取决于 SITDAR 的 TDRA 位的设定值。 • 如果将 SITDAR 的 TDRE 位置 “1” 并且将 SITDAR 的 TLREP 位清 “0”，这些位就有效。

22.3.4 接收数据寄存器 (SIRDR)

SIRDR 读此模块的接收数据。将接收 FIFO 的数据保存到此寄存器。
在通过 SICTR 的 RXRST 位进行接收复位时，对此寄存器进行初始化。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SIRDL[15:0]															
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SIRDR[15:0]															
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 16	SIRDL[15:0]	不定值	R	左 ch. 接收数据 将 SIOFRxD 接收的数据保存为左 ch. 数据。接收帧中的左 ch. 数据的位置取决于 SIRDAR 的 RDLA 位的设定值。 • 如果将 SIRDAR 的 RDLE 位置 “1”，这些位就有效。
15 ~ 0	SIRDR[15:0]	不定值	R	右 ch. 接收数据 将 SIOFRxD 接收的数据保存为右 ch. 数据。接收帧中的右 ch. 数据的位置取决于 SIRDAR 的 RDRA 位的设定值。 • 如果将 SIRDAR 的 RDRE 位置 “1”，这些位就有效。

22.3.5 状态寄存器 (SISTR)

SISTR 表示此模块的状态。如果将 SIIEP 寄存器的对应位置“1”，此寄存器的各位就为此模块的中断请求。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TFEMP	TDREQ	—	—	RFFUL	RDREQ	—	—	—	FSERR	TFOVF	TFUDF	RFUDF	RFOVF
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14	—	0	R	保留位 读取值为不定值，写入值总是“0”。
13	TFEMP	0	R	发送 FIFO 空 0: 发送 FIFO 不是空状态 1: 发送 FIFO 是空状态 • 此位在 SICTR 的 TXE 位为“1”时有效。 • 如果将数据写到 SITDR，就通过此模块将此位清“0”。 【注】 当此位被置“1”时，发送 FIFO 可能已经发生下溢。因此，在写发送数据寄存器时，不能使用此位。
12	TDREQ	0	R	发送 FIFO 传送请求 0: 发送 FIFO 的空区域小于 SIFCTR 的 TFWM 位的设定值 1: 发送 FIFO 的空区域大于等于 SIFCTR 的 TFWM 位的设定值 如果发送 FIFO 的空区域超过 SIFCTR 的 TFWM 位的设定值，就发行发送 FIFO 传送请求。 在通过直接存储器存取控制器进行发送数据的传送时，必须通过存取直接存储器存取控制器将此位清“0”。如果在存取直接存储器存取控制器后仍满足此位的设定条件，就再次将此位置“1”。 • 此位在 SICTR 的 TXE 位为“1”时有效。 • 如果发送 FIFO 的空区域小于 SIFCTR 的 TFWM 位的设定值，就通过此模块将此位清“0”。
11、10	—	全 0	R	保留位 读写值都为“0”。
9	RFFUL	0	R	接收 FIFO 满 0: 接收 FIFO 不是满状态 1: 接收 FIFO 是满状态 • 此位在 SICTR 的 RXE 位为“1”时有效。 • 如果读 SIRDR，就通过此模块将此位清“0”。

位	位名	初始值	R/W	说明
8	RDREQ	0	R	<p>接收 FIFO 传送请求</p> <p>0: 接收 FIFO 的有效区域小于 SIFCTR 的 RFWM 位的设定值 1: 接收 FIFO 的有效区域大于等于 SIFCTR 的 RFWM 位的设定值</p> <p>如果接收 FIFO 的有效区域超过 SIFCTR 的 RFWM 位的设定值, 就发行接收 FIFO 传送请求。</p> <p>在通过直接存储器存取控制器进行接收数据的传送时, 必须通过存取直接存储器存取控制器将此位清“0”。如果在存取直接存储器存取控制器后仍满足此位的设定条件, 就再次将此位置“1”。</p> <ul style="list-style-type: none"> • 此位在 SICTR 的 RXE 位为“1”时有效。 • 如果接收 FIFO 的有效区域小于 SIFCTR 的 RFWM 位的设定值, 就通过此模块将此位清“0”。
7 ~ 5	—	全 0	R	<p>保留位</p> <p>读写值都为“0”。</p>
4	FSERR	0	R/W	<p>帧同步错误</p> <p>0: 未发生帧同步错误 1: 发生帧同步错误</p> <p>帧同步错误是指在结束传送数据的发送和接收前进入下一帧同步时序。在发生帧同步错误时, 此模块只对能传送的槽进行发送和接收。</p> <ul style="list-style-type: none"> • 此位在 SICTR 的 TXE 位或者 RXE 位为“1”时有效。 • 如果将此位置“1”, 就通过此模块将此位清“0”。此位置“0”的操作无效。
3	TFOVF	0	R/W	<p>发送 FIFO 上溢</p> <p>0: 发送 FIFO 未发生上溢 1: 发送 FIFO 发生上溢</p> <p>发送 FIFO 上溢是指在发送 FIFO 满的状态下写 SITDR。</p> <p>在发送 FIFO 发生上溢时, 此模块将上溢时的写操作置为无效。</p> <ul style="list-style-type: none"> • 此位在 SICTR 的 TXE 位为“1”时有效。 • 如果将此位置“1”, 就通过此模块将此位清“0”。此位置“0”的操作无效。
2	TFUDF	0	R/W	<p>发送 FIFO 下溢</p> <p>0: 发送 FIFO 未发生下溢 1: 发送 FIFO 发生下溢</p> <p>发送 FIFO 下溢是指在发送 FIFO 空的状态下因发送而进行了装入。在发送 FIFO 发生下溢时, 此模块重复发送上次的发送数据。</p> <ul style="list-style-type: none"> • 此位在 SICTR 的 TXE 位为“1”时有效。 • 如果将此位置“1”, 就通过此模块将此位清“0”。此位置“0”的操作无效。
1	RFUDF	0	R/W	<p>接收 FIFO 下溢</p> <p>0: 接收 FIFO 未发生下溢 1: 接收 FIFO 发生下溢</p> <p>接收 FIFO 下溢是指在接收 FIFO 空的状态下读 SIRDR。</p> <p>在接收 FIFO 发生下溢时, 不保证从 SIRDR 读到的数据值。</p> <ul style="list-style-type: none"> • 此位在 SICTR 的 RXE 位为“1”时有效。 • 如果将此位置“1”, 就通过此模块将此位清“0”。此位置“0”的操作无效。

位	位名	初始值	R/W	说明
0	RFOVF	0	R/W	接收 FIFO 上溢 0: 接收 FIFO 未发生上溢 1: 接收 FIFO 发生上溢 接收 FIFO 上溢是指在接收 FIFO 满的状态下因接收而进行了写操作。 在接收 FIFO 发生上溢时, 上溢的接收数据丢失。 • 如果将此位置“1”, 就通过此模块将此位清“0”。将此位置“0”的操作无效。

22.3.6 中断允许寄存器 (SIIER)

SIIER 允许发行此模块的中断。如果将此寄存器的各位置“1”并且将 SISTR 对应的各位置“1”, 就发行中断。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TDMAE	—	TFEMPE	TDREQE	RDMAE	—	RFFULE	RDREQE	—	—	—	FSERRE	TFOVFE	TFUDFE	RFUDFE	RFOVFE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	TDMAE	0	R/W	发送 FIFO 的 DMA 传送请求允许 将发送 FIFO 传送请求作为中断请求 /DMA 传送请求进行传送。 0: 用作向 CPU 请求的中断 1: 用作向直接存储器存取控制器请求的 DMA 传送
14	—	0	R	保留位 读写值都为“0”。
13	TFEMPE	0	R/W	发送 FIFO 空允许 0: 禁止由发送空引起的中断 1: 允许由发送空引起的中断
12	TDREQE	0	R/W	发送 FIFO 传送请求允许 0: 禁止由发送 FIFO 传送请求引起的中断请求 /DMA 传送请求 1: 允许由发送 FIFO 传送请求引起的中断请求 /DMA 传送请求
11	RDMAE	0	R/W	接收 FIFO 的 DMA 传送请求允许 将接收 FIFO 传送请求作为中断请求 /DMA 传送请求进行传送。 0: 用作向 CPU 请求的中断 1: 用作向直接存储器存取控制器请求的 DMA 传送
10	—	0	R	保留位 读写值都为“0”。
9	RFFULE	0	R/W	接收 FIFO 满允许 0: 禁止由接收 FIFO 满引起的中断 1: 允许由接收 FIFO 满引起的中断
8	RDREQE	0	R/W	接收 FIFO 传送请求允许 0: 禁止由接收 FIFO 传送请求引起的中断请求 /DMA 传送请求 1: 允许由接收 FIFO 传送请求引起的中断请求 /DMA 传送请求
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。

位	位名	初始值	R/W	说明
4	FSERRE	0	R/W	帧同步错误允许 0: 禁止帧同步错误引起的中断 1: 允许帧同步错误引起的中断
3	TFOVFE	0	R/W	发送 FIFO 上溢允许 0: 禁止由发送 FIFO 上溢引起的中断 1: 允许由发送 FIFO 上溢引起的中断
2	TFUDFE	0	R/W	发送 FIFO 下溢允许 0: 禁止由发送 FIFO 下溢引起的中断 1: 允许由发送 FIFO 下溢引起的中断
1	RFUDFE	0	R/W	接收 FIFO 下溢允许 0: 禁止由接收 FIFO 下溢引起的中断 1: 允许由接收 FIFO 下溢引起的中断
0	RFOVFE	0	R/W	接收 FIFO 上溢允许 0: 禁止由接收 FIFO 上溢引起的中断 1: 允许由接收 FIFO 上溢引起的中断

22.3.7 FIFO 控制寄存器 (SIFCTR)

SIFCTR 表示能进行发送 / 接收 FIFO 传送的区域。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TFWM2	TFWM1	TFWM0	TFUA4	TFUA3	TFUA2	TFUA1	TFUA0	RFWM2	RFWM1	RFWM0	RFUA4	RFUA3	RFUA2	RFUA1	RFUA0
初始值:	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R

位	位名	初始值	R/W	说明
15	TFWM2	0	R/W	发送 FIFO 水位标记
14	TFWM1	0	R/W	000: 在发送 FIFO 的空区域为 16 段时发行传送请求
13	TFWM0	0	R/W	001: 禁止设定 010: 禁止设定 011: 禁止设定 100: 在发送 FIFO 的空区域大于等于 12 段时发行传送请求 101: 在发送 FIFO 的空区域大于等于 8 段时发行传送请求 110: 在发送 FIFO 的空区域大于等于 4 段时发行传送请求 111: 在发送 FIFO 的空区域大于等于 1 段时发行传送请求 • 通过 SISTR 的 TDREQE 位发行发送 FIFO 的传送请求。 • 与这些位的设定无关, 发送 FIFO 总是作为 16 段 FIFO 运行。
12	TFUA4	1	R	发送 FIFO 能使用的区域
11	TFUA3	0	R	用 B'00000 (满) ~ B'10000 (空) 表示能传送的 FIFO 段数。
10	TFUA2	0	R	
9	TFUA1	0	R	
8	TFUA0	0	R	
7	RFWM2	0	R/W	接收 FIFO 水位标记
6	RFWM1	0	R/W	000: 在接收 FIFO 的有效数据区大于等于 1 段时发行传送请求
5	RFWM0	0	R/W	001: 禁止设定 010: 禁止设定 011: 禁止设定 100: 在接收 FIFO 的有效数据区大于等于 4 段时发行传送请求 101: 在接收 FIFO 的有效数据区大于等于 8 段时发行传送请求 110: 在接收 FIFO 的有效数据区大于等于 12 段时发行传送请求 111: 在接收 FIFO 的有效数据区大于等于 16 段时发行传送请求 • 通过 SISTR 的 RDREQE 位发行接收 FIFO 的传送请求。 • 与这些位的设定无关, 接收 FIFO 总是作为 16 段 FIFO 运行。
4	RFUA4	0	R	接收 FIFO 能使用的区域
3	RFUA3	0	R	用 B'00000 (空) ~ B'10000 (满) 表示能传送的 FIFO 段数。
2	RFUA2	0	R	
1	RFUA1	0	R	
0	RFUA0	0	R	

22.3.8 时钟选择寄存器 (SISCR)

SISCR 设定主控模式时的串行时钟生成条件。当 SIMDR 的 TRMD1 位和 TRMD0 位为“B'10”时，此寄存器的设定有效。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSSEL	—	—	BRPS4	BRPS3	BRPS2	BRPS1	BRPS0	—	—	—	—	—	BRDV2	BRDV1	BRDV0
初始值:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	MSSEL	1	R/W	主控时钟源选择 0: 将 AUDIO_X1 用作主控时钟 1: 将 AUDIO_CLK 用作主控时钟 主控时钟是指输入到波特率发生器的时钟。
14、13	—	全 0	R	保留位 读写值都为“0”。
12	BRPS4	0	R/W	预分频器设定 通过波特率发生器的预分频器的计数值设定主控时钟的分频比。 设定的范围为 B'00000 ($\times 1/1$) ~ B'11111 ($\times 1/32$)。
11	BRPS3	0	R/W	
10	BRPS2	0	R/W	
9	BRPS1	0	R/W	
8	BRPS0	0	R/W	
7 ~ 3	—	全 0	R	保留位 读写值都为“0”。
2	BRDV2	0	R/W	波特率发生器分频比 设定波特率发生器中的输出段的分频比。 000: 预分频器输出 $\times 1/2$ 001: 预分频器输出 $\times 1/4$ 010: 预分频器输出 $\times 1/8$ 011: 预分频器输出 $\times 1/16$ 100: 预分频器输出 $\times 1/32$ 101: 禁止设定 110: 禁止设定 111: 禁止设定 波特率发生器的最后分频比取决于 BRPS \times BRDV (最大为 1/1024)。
1	BRDV1	0	R/W	
0	BRDV0	0	R/W	

22.3.9 发送数据分配寄存器 (SITDAR)

SITDAR 设定帧内的发送数据位置 (槽号)。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TDLE	—	—	—	TDLA3	TDLA2	TDLA1	TDLA0	TDRE	TLREP	—	—	TDRA3	TDRA2	TDRA1	TDRA0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	TDLE	0	R/W	发送左 ch. 数据允许 0: 禁止发送左 ch. 数据 1: 允许发送左 ch. 数据
14 ~ 12	—	全 0	R	保留位 读写值都为 “0”。
11	TDLA3	0	R/W	发送左 ch. 数据分配 3 ~ 0 用 B'0000 (0) ~ B'1110 (14) 设定发送帧中的左 ch. 数据的位置。 1111: 禁止设定 • 给 SITDR 的 SITDL 位设定左 ch. 数据的发送数据。
10	TDLA2	0	R/W	
9	TDLA1	0	R/W	
8	TDLA0	0	R/W	
7	TDRE	0	R/W	发送右 ch. 数据允许 0: 禁止发送右 ch. 数据 1: 允许发送右 ch. 数据
6	TLREP	0	R/W	发送左 ch. 重复 0: 将 SITDR 的 SITDR 位的设定值作为右 ch. 数据进行发送 1: 将 SITDR 的 SITDL 位的设定值作为右 ch. 数据进行重复发送 • 此位的设定在 TDRE 位为 “1” 时有效。 • 如果将此位置 “1”，就忽视 SITDR 的设定。
5、4	—	全 0	R	保留位 读写值都为 “0”。
3	TDRA3	0	R/W	发送右 ch. 数据分配 3 ~ 0 用 B'0000 (0) ~ B'1110 (14) 设定发送帧中的右 ch. 数据的位置。 1111: 禁止设定 • 给 SITDR 的 SITDR 位设定右 ch. 数据的发送数据。
2	TDRA2	0	R/W	
1	TDRA1	0	R/W	
0	TDRA0	0	R/W	

22.3.10 接收数据分配寄存器 (SIRDAR)

SIRDAR 设定帧内的接收数据的位置 (槽号)。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDLE	—	—	—	RDLA3	RDLA2	RDLA1	RDLA0	RDRE	—	—	—	RDRA3	RDRA2	RDRA1	RDRA0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	RDLE	0	R/W	接收左 ch. 数据允许 0: 禁止接收左 ch. 数据 1: 允许接收左 ch. 数据
14 ~ 12	—	全 0	R	保留位 读写值都为 “0”。
11	RDLA3	0	R/W	接收左 ch. 数据分配 3 ~ 0 用 B'0000 (0) ~ B'1110 (14) 设定接收帧中的左 ch. 数据的位置。 1111: 禁止设定 • 将左 ch. 数据的接收数据保存到 SIRDR 的 SIRDL 位。
10	RDLA2	0	R/W	
9	RDLA1	0	R/W	
8	RDLA0	0	R/W	
7	RDRE	0	R/W	接收右 ch. 数据允许 0: 禁止接收右 ch. 数据 1: 允许接收右 ch. 数据
6 ~ 4	—	全 0	R	保留位 读写值都为 “0”。
3	RDRA3	0	R/W	接收右 ch. 数据分配 3 ~ 0 用 B'0000 (0) ~ B'1110 (14) 设定接收帧中的右 ch. 数据的位置。 1111: 禁止设定 • 将右 ch. 数据的接收数据保存到 SIRDR 的 SIRDR 位。
2	RDRA2	0	R/W	
1	RDRA1	0	R/W	
0	RDRA0	0	R/W	

22.4 运行说明

22.4.1 串行时钟

(1) 主控模式 / 从属模式

此模块的时钟模式有以下 2 种：

- 从属模式：SIOFSCK 和 SIOFSYNC 为输入
- 主控模式：SIOFSCK 和 SIOFSYNC 为输出

(2) 波特率发生器

当此模块为主控模式时，使用波特率发生器（BRG）生成串行时钟。波特率发生器的分频比为 $1/2 \sim 1/1024$ 。串行时钟的供给系统图如图 22.2 所示。

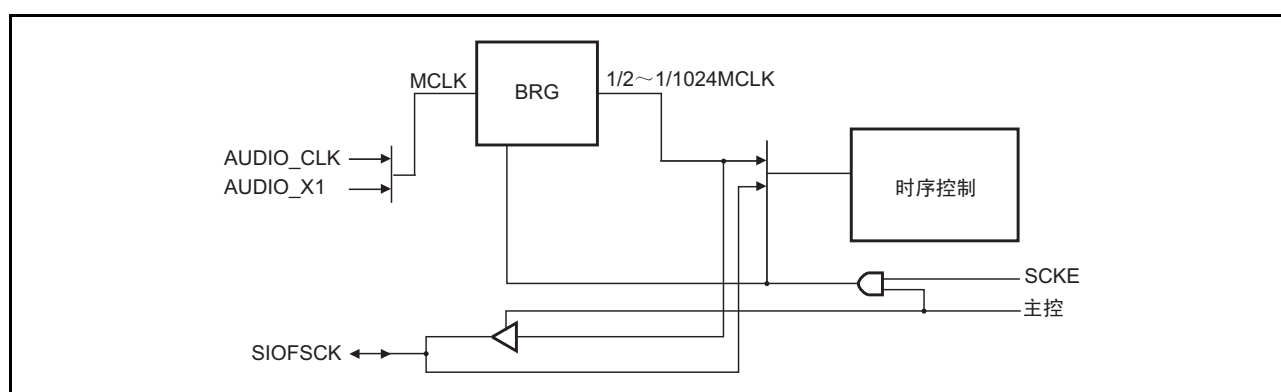


图 22.2 串行时钟的供给

串行时钟的频率例子如表 22.3 所示。

表 22.3 串行时钟的频率例子

帧长	采样率		
	8kHz	44.1kHz	48kHz
32 位	256kHz	1.4112MHz	1.536MHz
64 位	512kHz	2.8224MHz	3.072MHz
128 位	1.024MHz	5.6448MHz	6.144MHz
256 位	2.048MHz	11.289MHz	12.289MHz

22.4.2 串行时序

(1) SIOFSYNC

SIOFSYNC 是帧同步信号。

SIOFSYNC 的同步时序如图 22.3 所示。

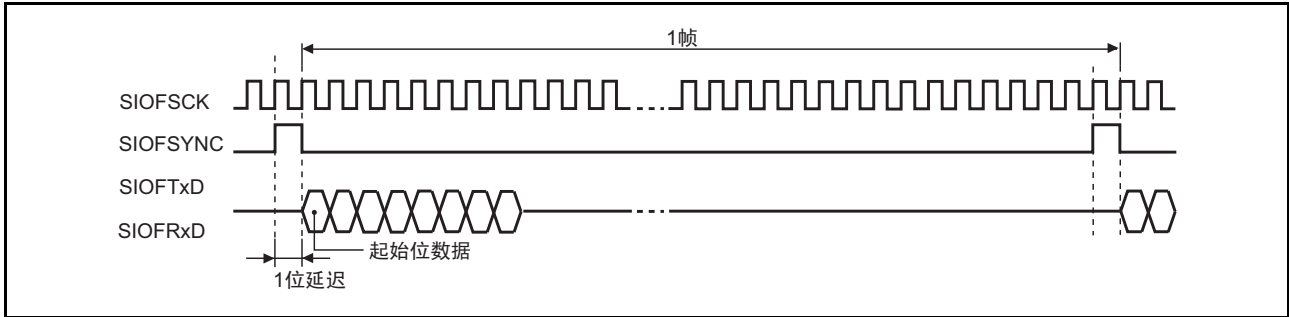


图 22.3 串行数据的同步时序

(2) 发送 / 接收时序

对于 SIOFSCK，能将 SIOFTxD 发送时序和 SIOFRxD 接收时序设定为以下的采样时序。通过 SIMDR 的 REDG 位设定发送 / 接收时序。

- 下降沿采样
- 上升沿采样（只限于主控模式）

发送 / 接收时序如图 22.4 所示。

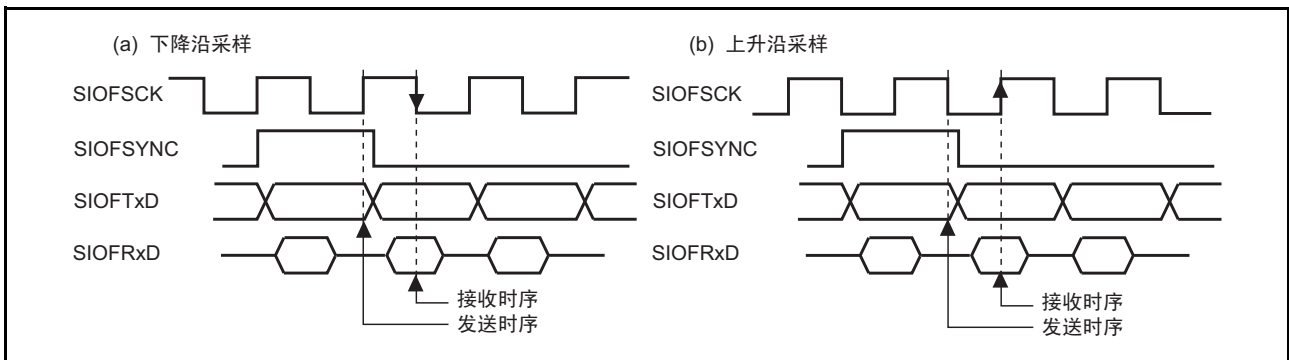


图 22.4 发送 / 接收时序

22.4.3 传送的数据格式

此模块进行以下传送：

- 发送/接收数据：8位单声道、16位单声道和16位立体声的传送数据

(1) 传送模式

此模块有表 22.4 所示的 2 种传送模式。给 SIMDR 的 TRMD1 ~ TRMD0 位设定传送模式。

表 22.4 串行传送模式

传送模式	SIOFSYNC	位延迟
从属模式	同步脉冲	SYNCDL 位
主控模式		

(2) 帧长

给 SIMDR 的 FL3 ~ FL0 位设定此模块要传送的帧长。设定值和帧长的关系如表 22.5 所示。

表 22.5 帧长

FL3 ~ FL0	槽长	每帧的位	对应的传送数据
00xx	8	8	8 位单声道
0100	8	16	8 位单声道
0101	8	32	8 位单声道
0110	8	64	8 位单声道
0111	8	128	8 位单声道
10xx	16	16	16 位单声道
1100	16	32	16 位单声道 / 立体声
1101	16	64	16 位单声道 / 立体声
1110	16	128	16 位单声道 / 立体声
1111	16	256	16 位单声道 / 立体声

【注】 x: Don't care

(3) 槽位置

此模块能用槽号分别设定 1 帧中的发送数据和接收数据的位置。对以下所示的寄存器进行设定。

- 发送数据：SITDAR
- 接收数据：SIRDAR

22.4.4 传送数据的寄存器分配

对下述寄存器进行发送 / 接收数据的读写。

- 写发送数据：SITDR（8位、16位、32位存取）
- 读接收数据：SIRDR（8位、16位、32位存取）

发送 / 接收数据和 SITDR、SIRDR 的位调整如图 22.5 所示。

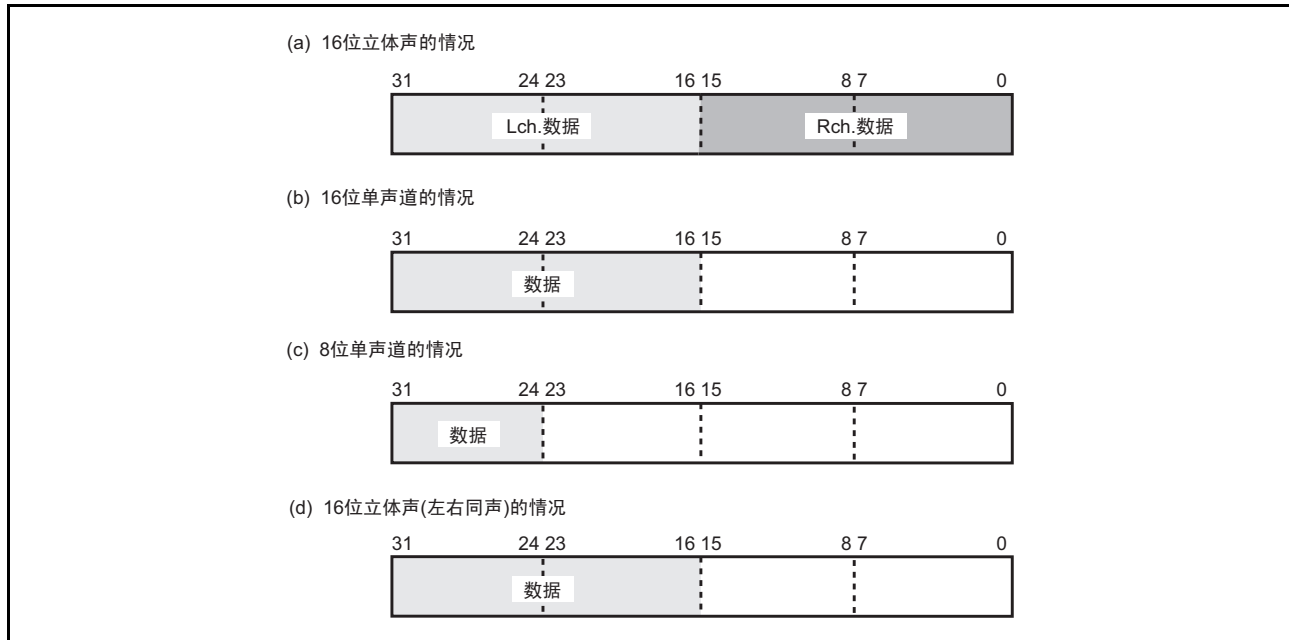


图 22.5 发送 / 接收数据的位调整

【注】 只发送和接收阴影部分的有效数据，因此阴影部分以外的数据不是发送和接收的对象。

给 SITDAR 的 TDLE 位和 TDRE 位设定发送数据的单声道或者立体声，给 SIRDAR 的 RDLE 位和 RDRE 位设定接收数据的单声道或者立体声，并且给 SITDAR 的 TLREP 位设定发送数据立体声时的左右同声输出。发送数据和接收数据的声音模式的设定分别如表 22.6 和表 22.7 所示。

表 22.6 发送数据的声音模式

模式 \ 位	TDLE	TDRE	TLREP
单声道	1	0	x
立体声	1	1	0
左右同声	1	1	1

【注】 x: Don't care

表 22.7 接收数据的声音模式

模式 \ 位	RDLE	RDRE
单声道	1	0
立体声	1	1

【注】 接收数据没有左右同声模式。

在通过单声道进行发送和接收时，必须使用左通道。

22.4.5 FIFO

(1) 概要

此模块的发送 / 接收 FIFO 的特点如下所示：

- 发送/接收FIFO的容量都为32位×16段。
- 与存取长度无关，使用FIFO缓冲器的1段（不能对1段分多次进行存取）。

(2) 传送请求

能通过下述请求将 FIFO 的传送请求输出到 CPU 和直接存储器存取控制器。

- 发送请求：TDREQ（发送FIFO传送请求）
- 接收请求：RDREQ（接收FIFO传送请求）

能分别设定发送 / 接收 FIFO 传送请求的发行条件。给 SIFCTR 的 TFWM2 ~ TFWM0 位设定发送请求的条件，并且给 RFWM2 ~ RFWM0 位设定接收 FIFO 的传送请求。发送请求和接收请求的发行条件分别如表 22.8 和表 22.9 所示。

表 22.8 发送请求的发行条件

TFWM2 ~ 0	请求段数	发送请求的发行	使用区域
000	1	空区域为 16 段	小 ↑ ↓ 大
100	4	空区域大于等于 12 段	
101	8	空区域大于等于 8 段	
110	12	空区域大于等于 4 段	
111	16	空区域大于等于 1 段	

表 22.9 接收请求的发行条件

RFWM2 ~ 0	请求段数	接收请求的发行	使用区域
000	1	有效数据大于等于 1 段	小 ↑ ↓ 大
100	4	有效数据大于等于 4 段	
101	8	有效数据大于等于 8 段	
110	12	有效数据大于等于 12 段	
111	16	有效数据为 16 段	

FIFO 的段数为 16 段。如果有效数据区或者空区域超过 16 段，就发行上溢错误或者下溢错误。即使 FIFO 不是空状态或者满状态，也在不满足上述条件时解除传送请求。

(3) 段数表示

用以下所示的寄存器表示发送 / 接收 FIFO 的使用状况。

- 发送FIFO：SIFCTR的TFUA4~TFUA0位表示空区域的段数
- 接收FIFO：SIFCTR的RFUA4~RFUA0位表示有效数据的段数

上述内容表示 CPU 或者直接存储器存取控制器能传送的数据量。

22.4.6 发送 / 接收步骤

(1) 主控模式中的发送步骤

此模块为主控模式中的发送设定例子和运行如图 22.6 所示。

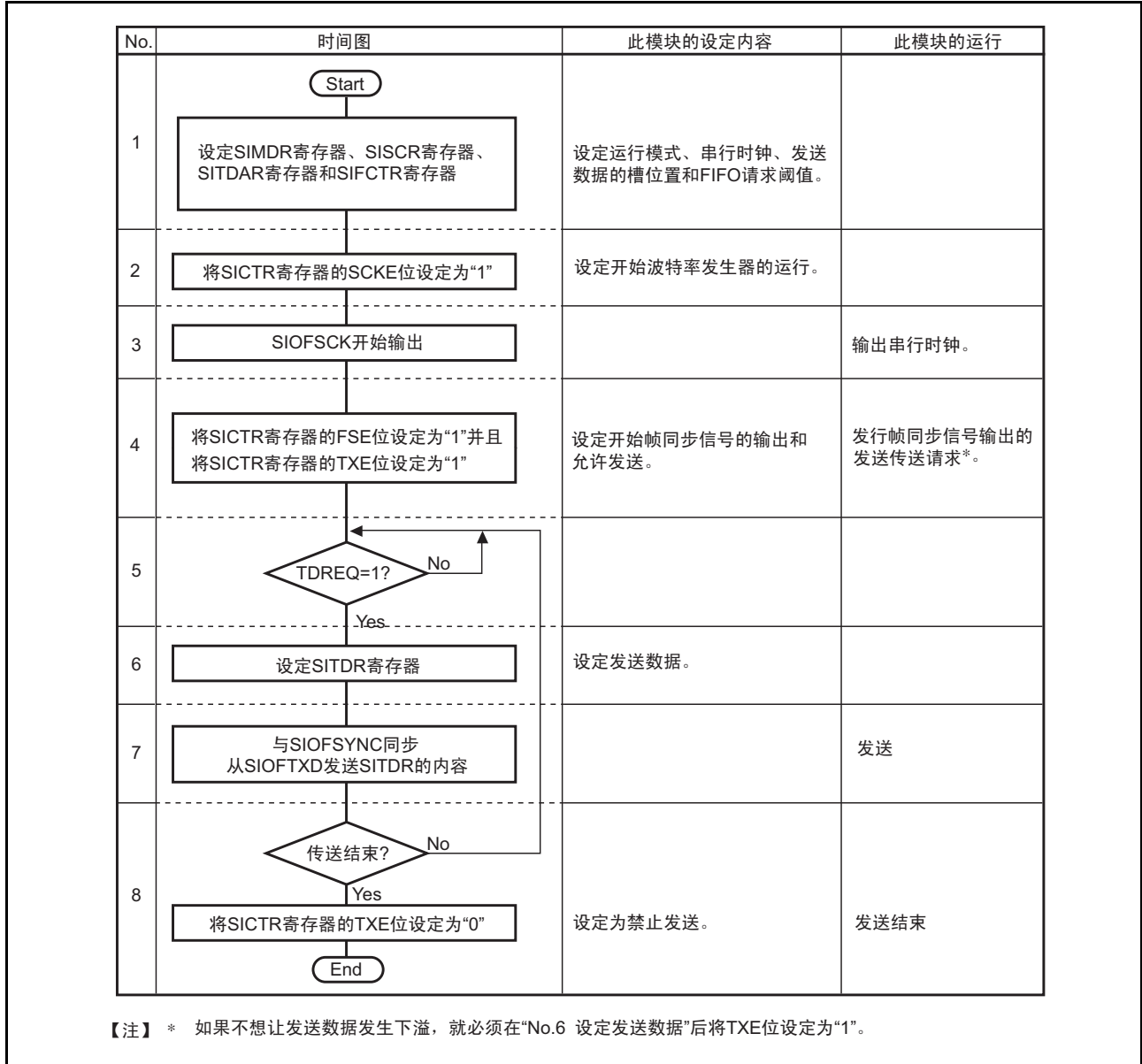


图 22.6 主控模式中的发送运行例子

(2) 主控模式中的接收步骤

此模块为主控模式中的接收设定例子和运行如图 22.7 所示。

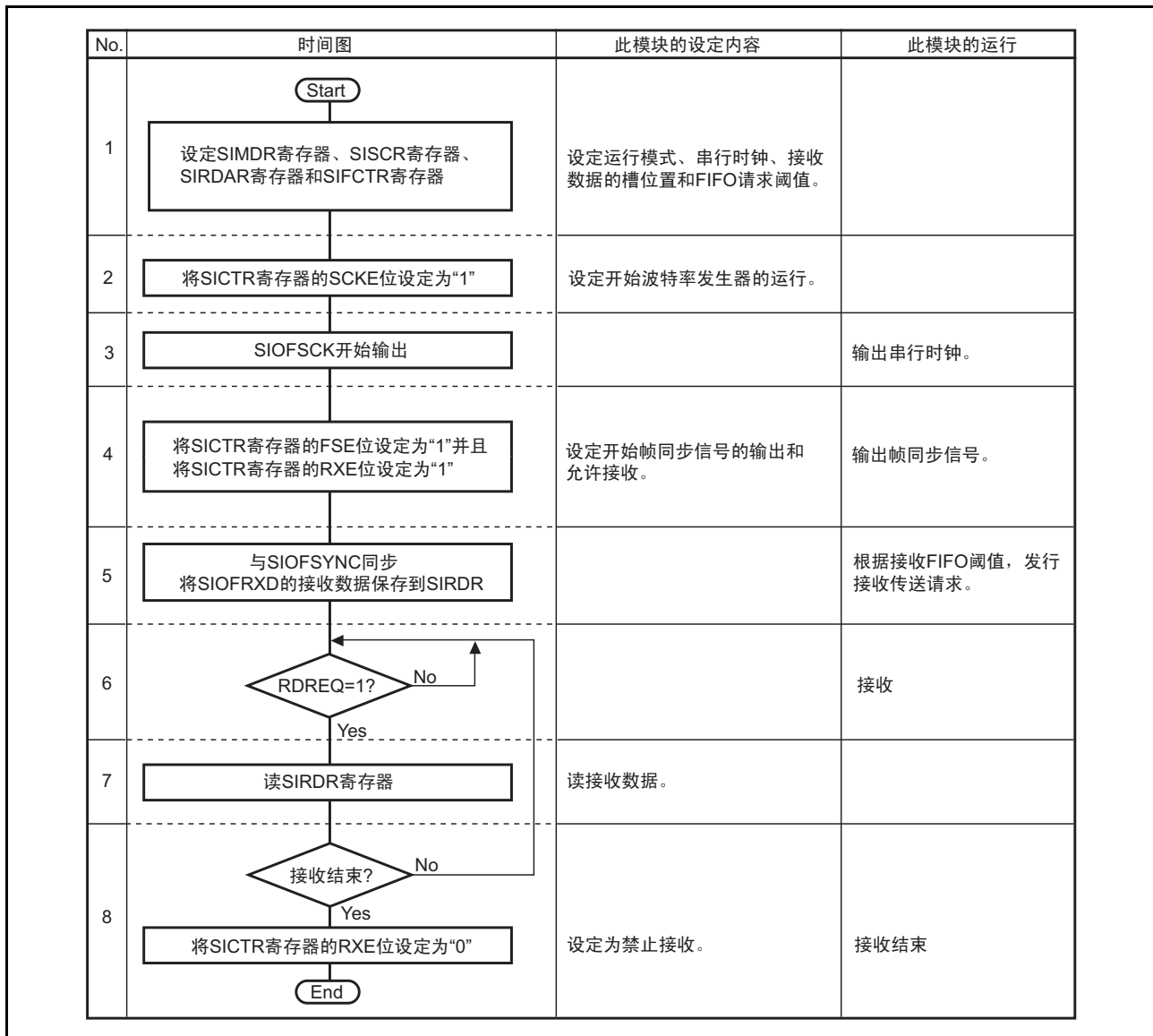


图 22.7 主控模式中的接收运行例子

(3) 从属模式中的发送

此模块为从属模式中的发送设定例子和运行如图 22.8 所示。

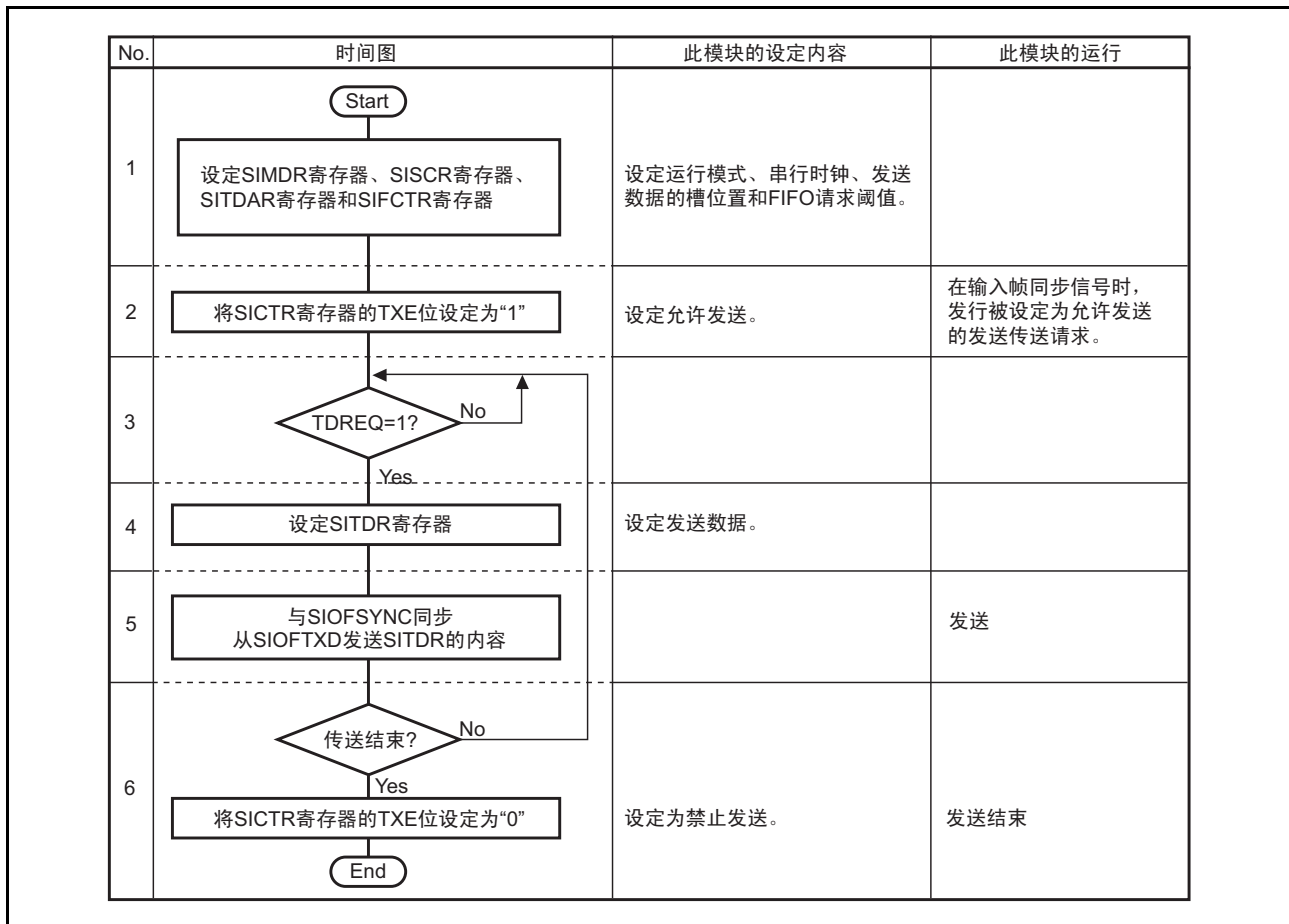


图 22.8 从属模式中的发送运行例子

(4) 从属模式中的接收

此模块为从属模式中的接收设定例子和运行如图 22.9 所示。

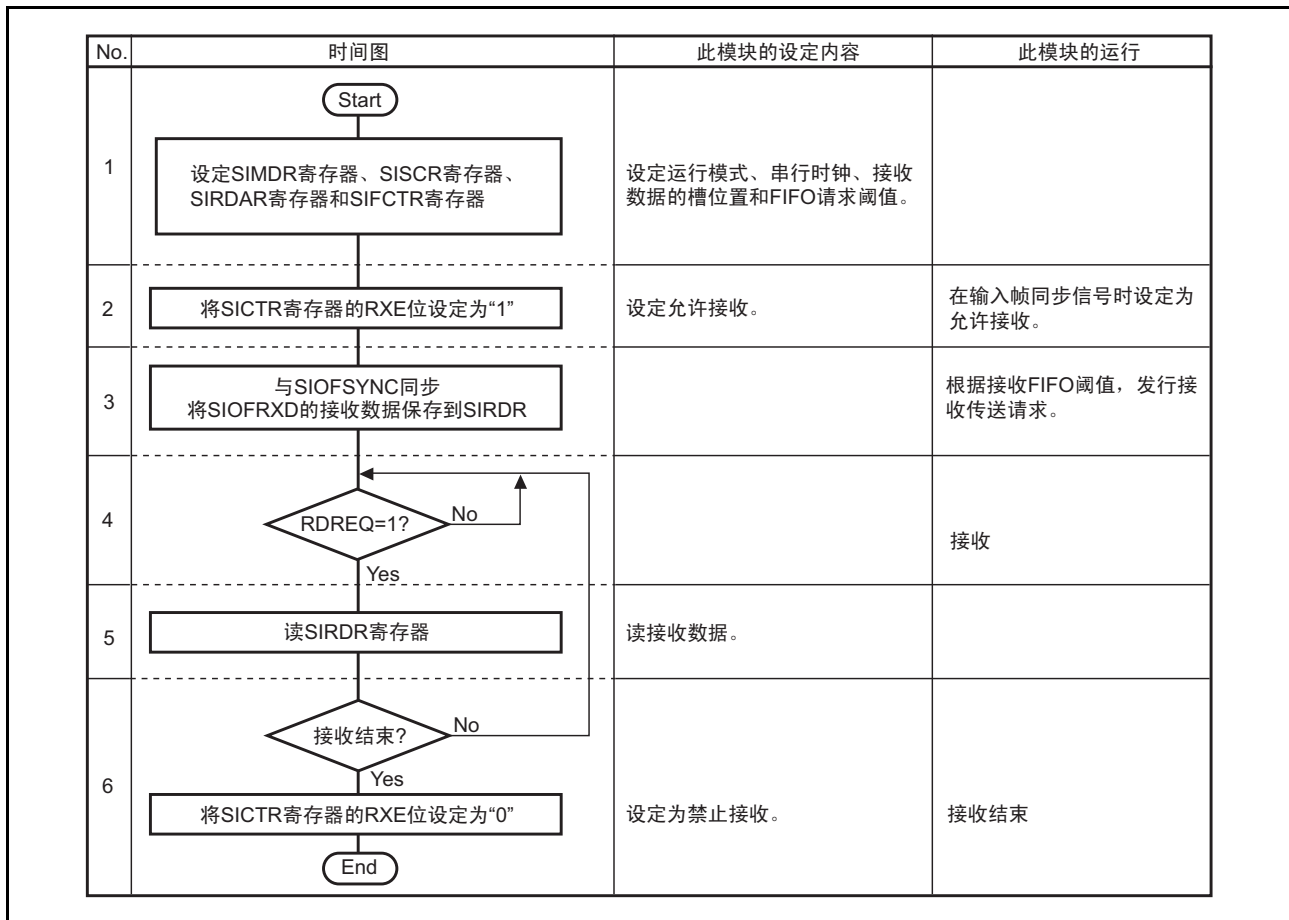


图 22.9 从属模式中的接收运行例子

(5) 发送 / 接收复位

此模块能通过将下述位置“1”，对发送部和接收部分别进行复位。

- 发送复位：（SICTR的TXRST位）
- 接收复位：（SICTR的RXRST位）

通过发送 / 接收复位进行初始化的内容如表 22.10 所示。

表 22.10 发送 / 接收复位

种类	初始化对象
发送复位	SITDR 发送 FIFO 的有效数据 SISTR 的 TFEMP 位、TDREQ 位 SICTR 的 TXE 位
接收复位	SIRDR 接收 FIFO 的有效数据 SISTR 的 RFFUL 位、RDREQ 位 SICTR 的 RXE 位

22.4.7 中断

此模块有 1 种中断。

(1) 中断请求

能分别通过多个请求发行中断。作为此模块的状态，SISTR 表示各请求。中断请求一览表如表 22.11 所示。

表 22.11 中断请求

No.	区分	位名	功能名称	内容
1	发送	TDREQ	发送 FIFO 传送请求	大于等于设定值的数据保存在发送 FIFO。
2		TFEMP	发送 FIFO 空	发送 FIFO 为空状态。
3	接收	RDREQ	接收 FIFO 传送请求	大于等于设定值的数据保存在接收 FIFO。
4		RFFUL	接收 FIFO 满	接收 FIFO 为满状态。
5	错误	TFUDF	发送 FIFO 下溢	在发送 FIFO 空的状态下进入串行数据的发送时序。
6		TFOVF	发送 FIFO 上溢	在发送 FIFO 满的状态下写发送 FIFO。
7		RFOVF	接收 FIFO 上溢	在接收 FIFO 满的状态下接收串行数据。
8		RFUDF	接收 FIFO 下溢	在接收 FIFO 空的状态下读接收 FIFO。
9		FSERR	FS 错误	在设定的位数前输入了同步信号（从属模式时）。

由 SIIER 的设定来决定是否通过中断请求发行中断。在 SIIER 的对应位为允许状态下发行这些位对应的中断请求时，发行此模块的中断。

(2) 发送和接收的区分

发送区分的请求和接收区分的请求是表示状态的信号，一旦将此位置“1”，如果在此后发送 / 接收 FIFO 的状态发生变化，此模块也自动将此位清“0”。

如果使用 DMA 传送，就必须通过存取直接存储器存取控制器将此位清“0”。如果在存取直接存储器存取控制器后仍满足设定条件，就再次将此位置“1”。

(3) 发生错误时的处理

在发生 SISTR 所示状态的各错误时，此模块进行以下运行：

- 发送 FIFO 下溢（TFUDF）
重新发送上次的发送数据。
- 发送 FIFO 上溢（TFOVF）
保存发送 FIFO 的内容，忽视上溢的写数据。
- 接收 FIFO 上溢（RFOVF）
放弃上溢的数据。
- 接收 FIFO 下溢（RFUDF）
读取值为不定值。
- FS 错误（FSERR）
根据错误的同步信号，对内部计数器进行复位。

22.4.8 发送 / 接收时序

此模块的串行发送 / 接收例子如图 22.10 ~ 图 22.15 所示。

(1) 8 位单声道 (1)

下降沿采样，发送数据和接收数据为槽 No.0，帧长为 8 位。

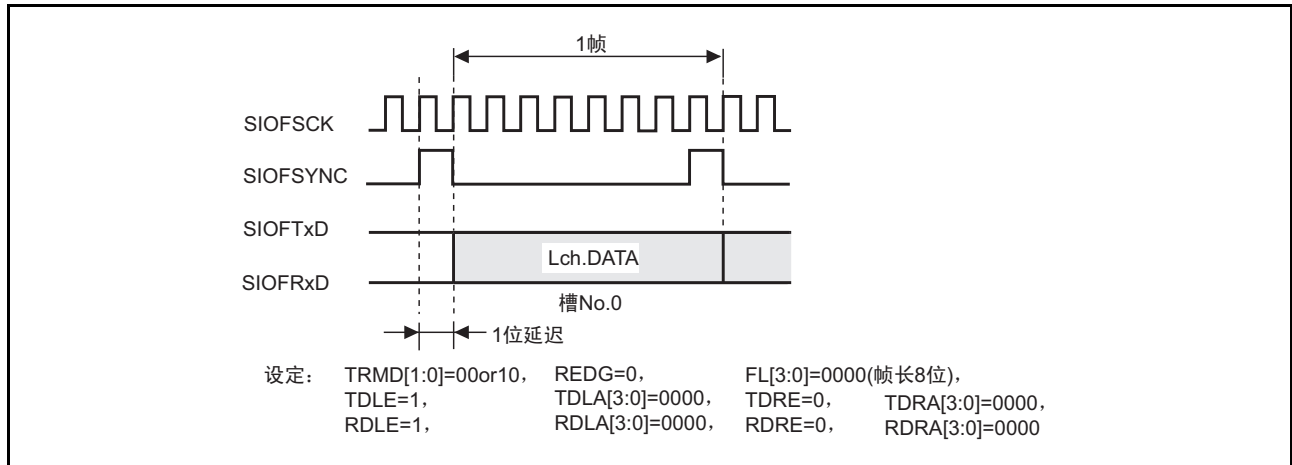


图 22.10 发送 / 接收时序 (8 位单声道 -1)

(2) 8 位单声道 (2)

下降沿采样，发送数据和接收数据为槽 No.0，帧长为 16 位。

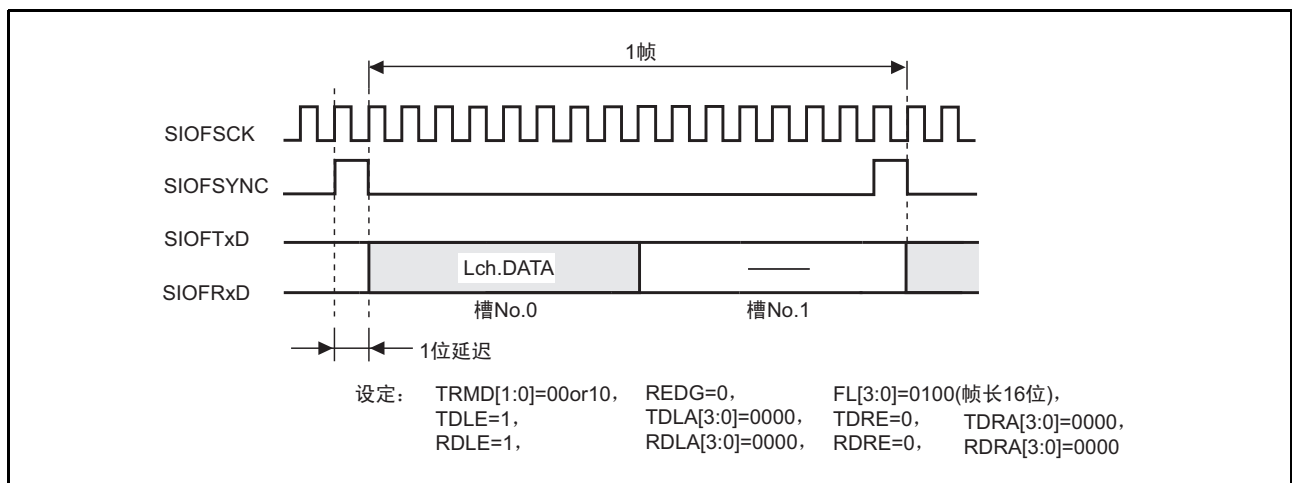


图 22.11 发送 / 接收时序 (8 位单声道 -2)

(3) 16 位单声道

下降沿采样，发送数据和接收数据为槽 No.0，帧长为 64 位。

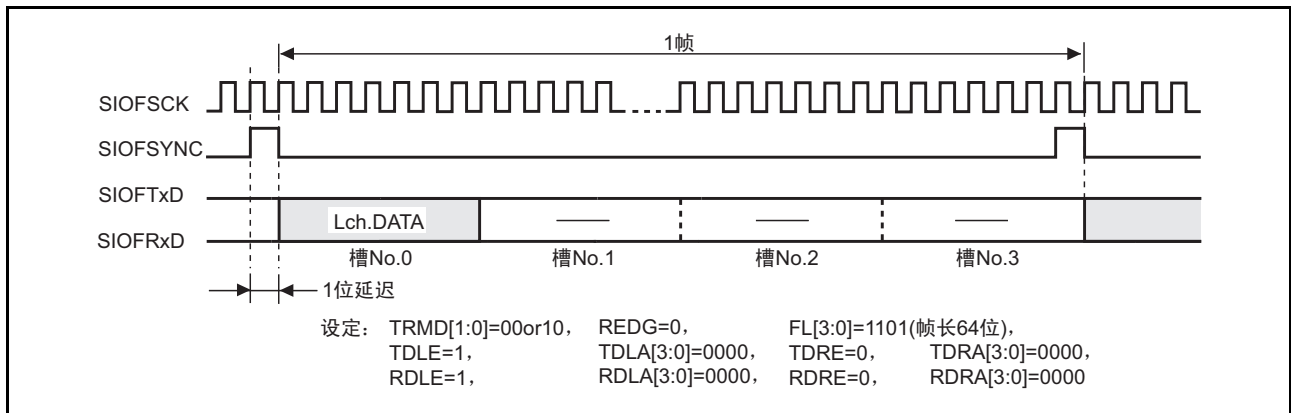


图 22.12 发送 / 接收时序 (16 位单声道)

(4) 16 位立体声 (1)

下降沿采样，Lch. 数据为槽 No.0，Rch. 数据为槽 No.1，帧长为 128 位。

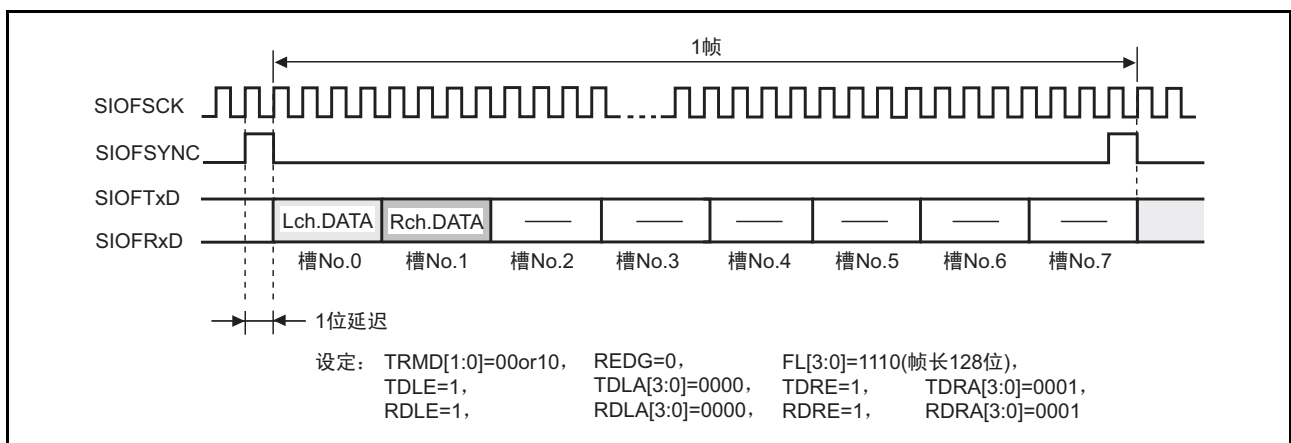
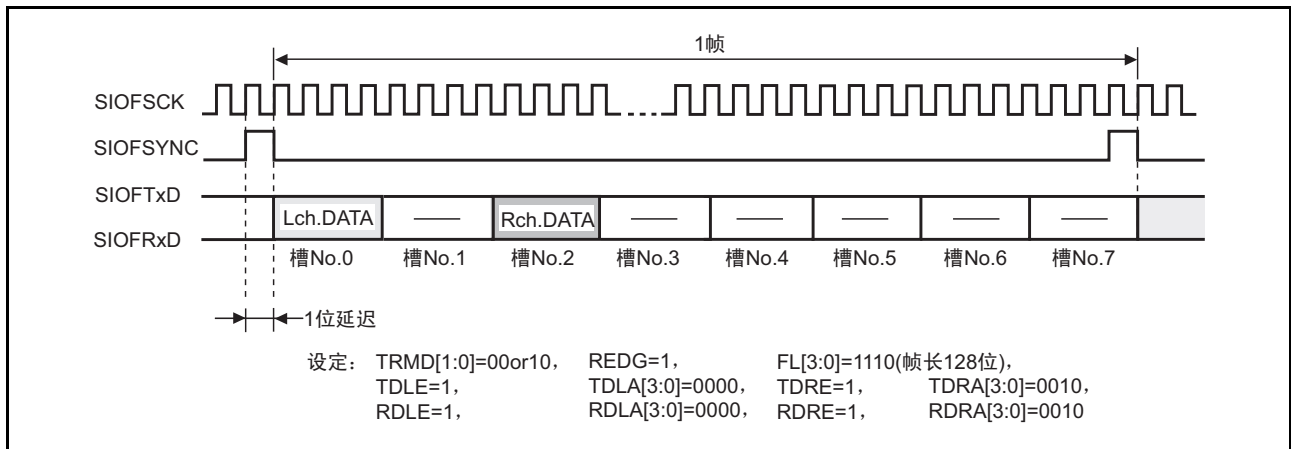


图 22.13 发送 / 接收时序 (16 位立体声 -1)

(5) 16 位立体声 (2)

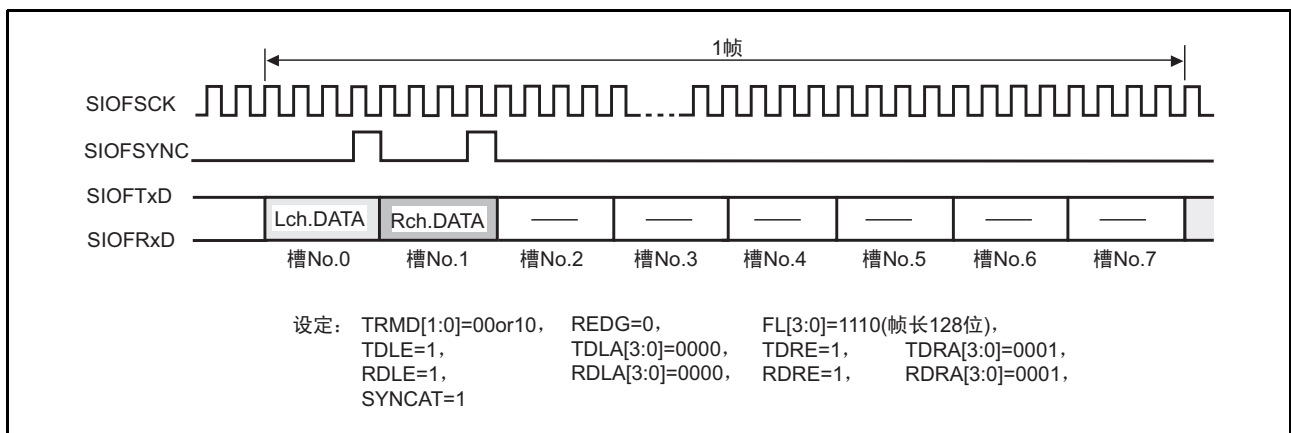
下降沿采样，Lch. 数据为槽 No.0，Rch. 数据为槽 No.2，帧长为 128 位。



(6) 各槽结束时输出同步脉冲模式的情况 (SYNCCAT 位为 “1”)

下降沿采样，Lch. 数据为槽 No.0，Rch. 数据为槽 No.1，帧长为 128 位。

在此模式中，必须将有效数据设定到槽 No.0，而且必须将有效数据设定为发送 / 接收或者发送。



23. 控制器局域网

控制器局域网（Renesas CAN Time Trigger Level 1）是控制以汽车和工业设备系统等实时通信为目的的 CAN（Controller Area Network）的模块。

本章说明程序接口。

有关 CAN 数据链路控制器的功能，请参照下述参考文献：

[参考文献]

1. CAN Specification Version 2.0 part A, Robert Bosch GmbH, 1991
2. CAN Specification Version 2.0 part B, Robert Bosch GmbH, 1991
3. Implementation Guide for the CAN Protocol, CAN Specification 2.0 Addendum, CAN In Automation, Erlangen, Germany, 1997
4. Road vehicles-Controller area network (CAN) : Part 1: Data link layer and physical signaling(ISO-11898-1, 2003)
5. Road vehicles - Controller area network (CAN) : Part 4: Time triggered communication(ISO-11898-4, 2004)

23.1 特点

23.1.1 特点

- 支持CAN规格2.0B。
- 位定时符合ISO-11898规格。
- 32个信箱
- 时钟频率：最大33.33MHz
- 31个可编程的发送/接收信箱和1个接收信箱
- 低功耗的CAN睡眠模式以及通过检测CAN总线有效来自动解除CAN的睡眠模式。
- 对应全部信箱的可编程接收滤波器屏蔽（标准ID和扩展ID）
- 最大1Mbps的可编程CAN数据速率
- 具有优先级内部区分功能的传送信息队列，以便防止因实时应用所引起的优先级倒序。
- 丰富的中断源
- 内置测试功能（监听专用模式和错误被动模式）。
- 16位的自由运行定时器（多种时钟源、预分频器、3个定时器的比较匹配寄存器）
- 用于发送时间触发的6位周期计数器（Basic Cycle）
- 通过定时器的比较匹配寄存器产生的中断
- 能清除并且设定定时器的计数器。
- 用于时间触发的寄存器：Local_Time、Cycle_time、Ref_Mark、Tx_Enable_Window、Ref_Trigger_Offset
- 发送和接收都支持SOF的灵活的时戳功能。
- 除了事件触发发送以外，还支持时间触发发送和固定周期发送。
- 能将周期计数器（Basic Cycle）嵌入CAN帧后进行发送。

23.1.2 本单片机的特点

- 搭载了32个信箱×3个通道
- 3个通道能通过设定通用输入/输出口进行以下连接使用：
32个信箱×3个通道
64个信箱×1个通道（RCAN_0、1）和32个信箱×1个通道（RCAN_2）
96个信箱×1个通道
- 通道0、1、2都能通过信箱0的接收信息启动直接存储器存取控制器。
- 能通过CRxn（PC5、PC7、PJ20、PJ22）引脚的变化解除深度待机模式。
- 详细内容请参照“49. 低功耗模式”。

【注】* 在64个和96个信箱的结构中，有使用时的注意事项，请参照“23.12 使用时的注意事项”。

23.2 结构

23.2.1 框图

此模块由支持 CAN2.0B Active 和 ISO-11898 的 CAN 帧构成，提供控制自由度大并且简练的方法。此模块根据功能分为微处理器接口（MPI）、信箱、信箱控制、定时器和 CAN 接口共 5 块。

框图如图 23.1 所示。

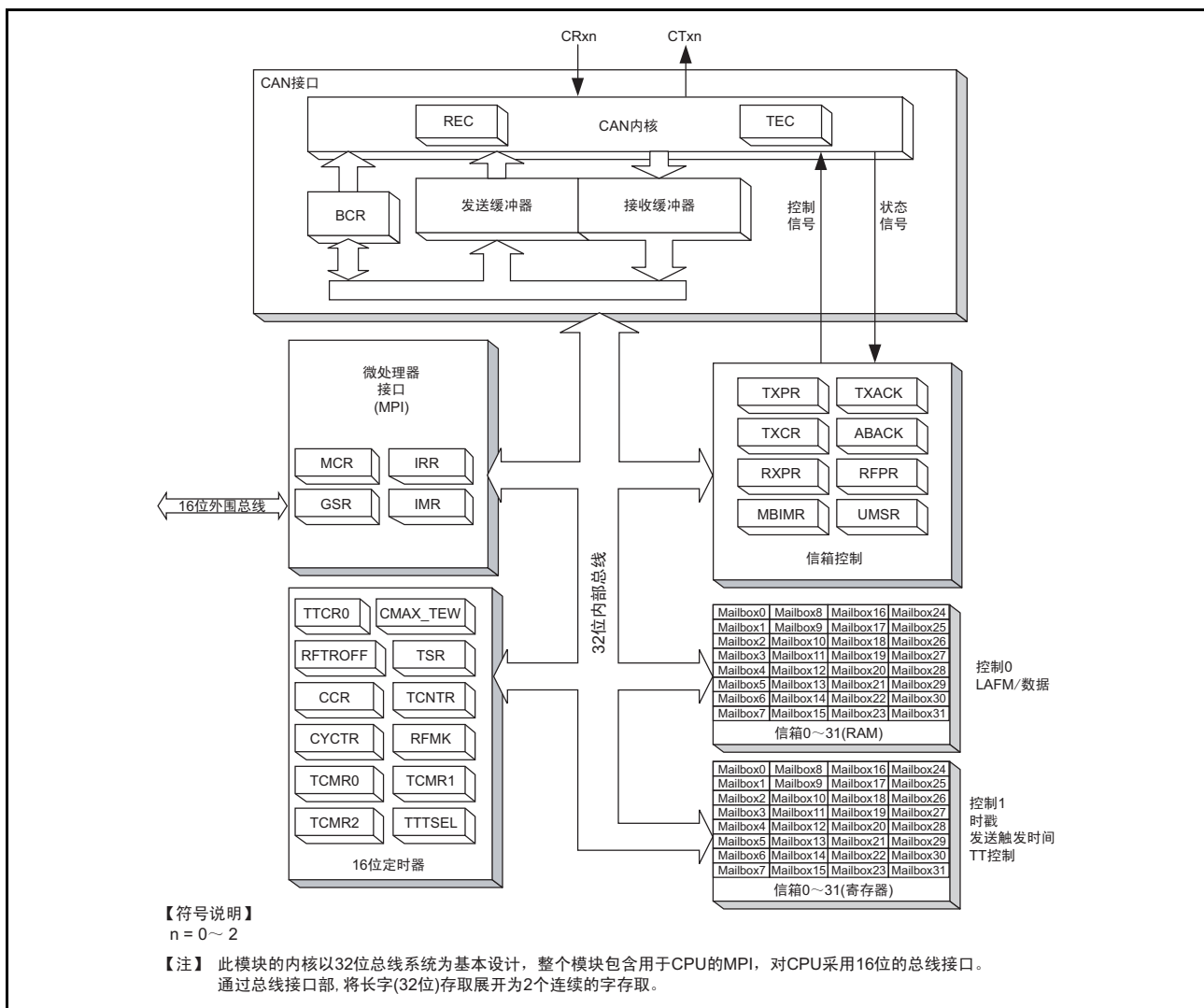


图 23.1 框图（每个通道）

23.2.2 各块的功能

(1) 微处理器接口 (MPI)

MPI 能进行 CPU 和此模块的寄存器或者信箱之间的通信，并且控制存储器接口。MPI 有唤醒控制逻辑，用于检测 CAN 总线有效，并且将 CAN 总线有效通知 MPI 和此模块以外的其他模块，因此此模块能自动解除 CAN 睡眠模式。另外，MPI 的寄存器有 MCR、IRR、GSR 和 IMR。

(2) 信箱

信箱作为信息缓冲器，分配在 RAM 和寄存器中。

RAM 和寄存器分别有 32 个信箱，保存以下信息：

[RAM]

- CAN 信息控制 (ID、RTR、IDE 等)
- CAN 信息数据 (用于 CAN 数据帧)
- 用于接收的本地接收滤波器屏蔽 (LAFM)

[寄存器]

- CAN 信息控制 (DLC)
- 用于发送/接收信息的时戳
- 3 位的信箱配置、自动重新发送无效位、用于远程请求的自动发送位、新信息控制位
- 发送触发时间

(3) 信箱控制

信箱控制具有以下功能：

- 在接收信息时，比较 ID，生成用于将来自 CAN 接口的信息保存到信箱的 RAM 地址和数据，并且进行对应寄存器的置位或者清除。
- 在发送事件触发信息时，进行内部仲裁，选择正确的优先级信息，并且将信息从信箱装入到 CAN 接口的发送缓冲器，然后进行对应寄存器的置位或者清除。在发送时间触发时，通过发送触发时间的比较匹配进行信息的装入。
- 对 CPU 和信箱控制之间的信箱存取进行仲裁。
- 寄存器有 TXPR、TXCR、TXACK、ABACK、RXPR、RFPR、MBIMR 和 UMSR。

(4) 定时器

此定时器是用于在特定的时间范围内发送信息以及记录结果的功能块，是能由 CPU 控制的 16 位自由运行递增计数器。有 1 个和本地时间进行比较的 16 位比较匹配寄存器、2 个和周期时间进行比较的比较匹配寄存器。这些比较匹配寄存器能产生中断信号并且清除计数器。能从系统时钟生成的多个时钟周期中选择此定时器的时钟，但是也能通过编程，让定时器用 CAN 总线的 1 位定时进行计数。此定时器的寄存器有 TCNTR、TTCR0、CMA_X_TEW、RETROFF、TSR、CCR、CYCTR、RFMK、TCMR0、TCMR1、TCMR2 和 TTTSEL。

(5) CAN 接口

此块支持参考文献 [2] 和 [4] 的 CAN 总线数据链路控制器规格，它满足 OSI 模型规定的的数据链路控制器的全部功能。另外，还提供用于 CAN 总线的寄存器和逻辑，具体有接收错误计数器、发送错误计数器、位配置寄存器等各种测试模式，而且还有保存 CAN 数据链路控制器的发送 / 接收数据的功能。

23.2.3 引脚结构

引脚结构如表 23.1 所示。

表 23.1 引脚结构

名称	引脚名	输入 / 输出	功能
发送数据引脚	CTxn	输出	用于 CAN 总线发送的引脚
接收数据引脚	CRxn	输入	用于 CAN 总线接收的引脚

【注】 n=0、1、2

23.2.4 存储器映像

存储器映像如图 23.2 所示。

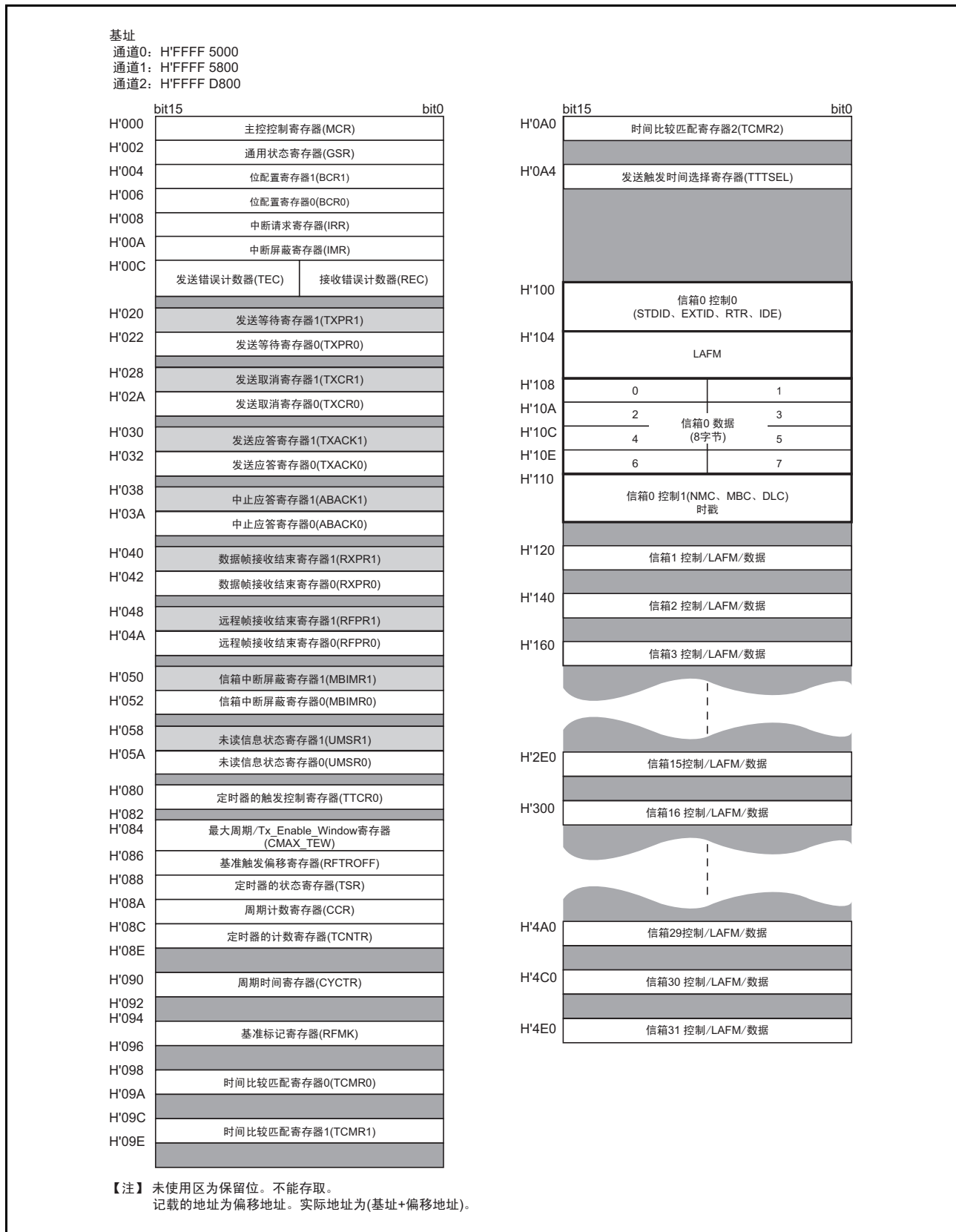


图 23.2 存储器映像 (每个通道)

23.3 信箱

23.3.1 信箱结构

信箱用作发送和接收 CAN 帧的信息缓冲器。各信箱由信息控制、本地接收滤波器屏蔽（LAFM）、信息数据的 3 个保存字段构成。而且还有具备时戳、时间触发配置和时间触发控制字段的信箱。

各信箱的信息控制、LAFM、信息数据、时戳、发送触发时间和时间触发控制的地址映像如表 23.2 所示。

表 23.2 各信箱的地址映像（每个通道）

信箱	地址						
	控制 0	LAFM	数据	控制 1	时戳	触发时间	TT 控制
	4 字节	4 字节	8 字节	2 字节	2 字节	2 字节	2 字节
0（只接收）	100-103	104-107	108-10F	110-111	112-113	—	—
1	120-123	124-127	128-12F	130-131	132-133	—	—
2	140-143	144-147	148-14F	150-151	152-153	—	—
3	160-163	164-167	168-16F	170-171	172-173	—	—
4	180-183	184-187	188-18F	190-191	192-193	—	—
5	1A0-1A3	1A4-1A7	1A8-1AF	1B0-1B1	1B2-1B3	—	—
6	1C0-1C3	1C4-1C7	1C8-1CF	1D0-1D1	1D2-1D3	—	—
7	1E0-1E3	1E4-1E7	1E8-1EF	1F0-1F1	1F2-1F3	—	—
8	200-203	204-207	208-20F	210-211	212-213	—	—
9	220-223	224-227	228-22F	230-231	232-233	—	—
10	240-243	244-247	248-24F	250-251	252-253	—	—
11	260-263	264-267	268-26F	270-271	272-273	—	—
12	280-283	284-287	288-28F	290-291	292-293	—	—
13	2A0-2A3	2A4-2A7	2A8-2AF	2B0-2B1	2B2-2B3	—	—
14	2C0-2C3	2C4-2C7	2C8-2CF	2D0-2D1	2D2-2D3	—	—
15	2E0-2E3	2E4-2E7	2E8-2EF	2F0-2F1	2F2-2F3	—	—
16	300-303	304-307	308-30F	310-311	—	—	—
17	320-323	324-327	328-32F	330-331	—	—	—
18	340-343	344-347	348-34F	350-351	—	—	—
19	360-363	364-367	368-36F	370-371	—	—	—
20	380-383	384-387	388-38F	390-391	—	—	—
21	3A0-3A3	3A4-3A7	3A8-3AF	3B0-3B1	—	—	—
22	3C0-3C3	3C4-3C7	3C8-3CF	3D0-3D1	—	—	—
23	3E0-3E3	3E4-3E7	3E8-3EF	3F0-3F1	—	—	—
24	400-403	404-407	408-40F	410-411	—	414-415	416-417
25	420-423	424-427	428-42F	430-431	—	434-435	436-437
26	440-443	444-447	448-44F	450-451	—	454-455	456-457
27	460-463	464-467	468-46F	470-471	—	474-475	476-477
28	480-483	484-487	488-48F	490-491	—	494-495	496-497
29	4A0-4A3	4A4-4A7	4A8-4AF	4B0-4B1	—	4B4-4B5	4B6-4B7
30	4C0-4C3	4C4-4C7	4C8-4CF	4D0-4D1	4D2-4D3 (本地时间)	4D4-4D5	—
31	4E0-4E3	4E4-4E7	4E8-4EF	4F0-4F1	4F2-4F3 (本地时间)	—	—

信箱 0 为接收（信息）专用信箱，其他全部信箱都能通过设定信息控制的 MBC（信箱结构）位进行发送和接收。信箱结构的详细内容如图 23.3 ~ 图 23.5 所示。

表 23.3 信箱的作用

	事件触发		时间触发		备注	
	发送	接收	发送	接收	时戳寄存器	发送触发时间寄存器
MB31	能设定	能设定	—	接收时间基准	有	—
MB30	能设定	能设定	在主控时间模式中发送时间基准	在从属时间模式中接收	有	有
MB29 ~ MB24	能设定	能设定	能设定	能设定	—	有
MB23 ~ MB16	能设定	能设定	— (ET)	能设定	—	—
MB15 ~ MB1	能设定	能设定	— (ET)	能设定	有	—
MB0	—	能设定	—	能设定	有	—

【注】 ET：能在时间触发模式中的合并仲裁窗口内进行发送。

• MB0(带时戳的接收信箱)																	
地址	数据总线															存取长度	字段名
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1		
H'100+N*32	IDE	RTR	0	STDID[10:0]										EXTID[17:16]	16/32	控制0	
H'102+N*32	EXTID[15:0]															16	
H'104+N*32	IDE	LAFM	0	0	STDID_LAFM[10:0]										EXTID_LAFM[17:16]	16/32	LAFM
H'106+N*32	EXTID_LAFM[15:0]															16	
H'108+N*32	MSG_DATA_0 (最初的Rx/Tx字节)							MSG_DATA_1							8/16/32	数据	
H'10A+N*32	MSG_DATA_2							MSG_DATA_3							8/16		
H'10C+N*32	MSG_DATA_4							MSG_DATA_5							8/16/32		
H'10E+N*32	MSG_DATA_6							MSG_DATA_7							8/16		
H'110+N*32	0	0	NMC	0	0	MBC[2:0]*	0	0	0	0	0	0	0	DLC[3:0]	8/16		控制1
H'112+N*32	TimeStamp[15:0] (CYCTR[15:0] or CCR[5:0]/CYCTR[15:6] at SOF)															16	时戳
• MB15~1(带时戳的信箱)																	
地址	数据总线															存取长度	字段名
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1		
H'100+N*32	IDE	RTR	0	STDID[10:0]										EXTID[17:16]	16/32	控制0	
H'102+N*32	EXTID[15:0]															16	
H'104+N*32	IDE	LAFM	0	0	STDID_LAFM[10:0]										EXTID_LAFM[17:16]	16/32	LAFM
H'106+N*32	EXTID_LAFM[15:0]															16	
H'108+N*32	MSG_DATA_0 (最初的Rx/Tx字节)							MSG_DATA_1							8/16/32	数据	
H'10A+N*32	MSG_DATA_2							MSG_DATA_3							8/16		
H'10C+N*32	MSG_DATA_4							MSG_DATA_5							8/16/32		
H'10E+N*32	MSG_DATA_6							MSG_DATA_7							8/16		
H'110+N*32	0	0	NMC	ATX	DART	MBC[2:0]	0	0	0	0	0	0	0	DLC[3:0]	8/16		控制1
H'112+N*32	TimeStamp[15:0] (CYCTR[15:0] or CCR[5:0]/CYCTR[15:6] at SOF)															16	时戳

【注】* MBC[1]固定为“1”。

1. 灰色表示的位为保留位。写入值为“0”，读取值不一定为“0”，而且没有保证。
2. 在信箱0中不支持ATX和DART，并且限制信箱0的MBC的设定值。
3. 因为MCR15位的初始值为“1”，所以信息控制、LAFM的STDID、RTR、IDE、EXTID的顺序和HCAN2不同。

图 23.3 信箱（0 ~ 15）的结构

• MB23~16(邮箱: 无时戳)

地址	数据总线															存取长度	字段名
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1		
H'100+N*32	IDE	RTR	0	STDID[10:0]										EXTID[17:16]	16/32	控制0	
H'102+N*32	EXTID[15:0]																16
H'104+N*32	IDE	LAFM	0	0	STDID_LAFM[10:0]										EXTID_LAFM[17:16]	16/32	LAFM
H'106+N*32	EXTID_LAFM[15:0]															16	
H'108+N*32	MSG_DATA_0 (最初的Rx/Tx字节)							MSG_DATA_1							8/16/32	数据	
H'10A+N*32	MSG_DATA_2							MSG_DATA_3							8/16		
H'10C+N*32	MSG_DATA_4							MSG_DATA_5							8/16/32		
H'10E+N*32	MSG_DATA_6							MSG_DATA_7							8/16		
H'110+N*32	0	0	NMC	ATX	DART	MBC[2:0]			0	0	0	0	DLC[3:0]				8/16

• MB29~24(在时间触发模式中发送时间触发)

地址	数据总线															存取长度	字段名			
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1			0		
H'100+N*32	IDE	RTR	0	STDID[10:0]										EXTID[17:16]	16/32	控制0				
H'102+N*32	EXTID[15:0]																16			
H'104+N*32	IDE	LAFM	0	0	STDID_LAFM[10:0]										EXTID_LAFM[17:16]	16/32	LAFM			
H'106+N*32	EXTID_LAFM[15:0]															16				
H'108+N*32	MSG_DATA_0 (最初的Rx/Tx字节)							MSG_DATA_1							8/16/32	数据				
H'10A+N*32	MSG_DATA_2							MSG_DATA_3							8/16					
H'10C+N*32	MSG_DATA_4							MSG_DATA_5							8/16/32					
H'10E+N*32	MSG_DATA_6							MSG_DATA_7							8/16					
H'110+N*32	0	0	NMC	ATX	DART	MBC[2:0]			0	0	0	0	DLC[3:0]				8/16	控制1		
H'112+N*32	保留															—	—			
H'114+N*32	发送触发时间(TTT)															16	触发时间			
H'116+N*32	TTW[1:0]			偏移							0	0	0	0	0	Rep_Factor			16	TT控制

【注】 1. 灰色表示的位为保留位。写入值为“0”，读取值不一定为“0”，而且没有保证。
2. 因为MCR15位的初始值为“1”，所以信息控制、LAFM的STDID、RTR、IDE、EXTID的顺序和HCAN2不同。

图 23.4 信箱（16～29）的结构

• MB30(在时间触发模式中发送时间基准)

地址	数据总线															存取长度	字段名
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1		
H'100+N*32	IDE	RTR	0	STDID[10:0]										EXTID[17:16]	16/32	控制0	
H'102+N*32	EXTID[15:0]																16
H'104+N*32	IDE	LAFM	0	0	STDID_LAFM[10:0]										EXTID_LAFM[17:16]	16/32	LAFM
H'106+N*32	EXTID_LAFM[15:0]															16	
H'108+N*32	MSG_DATA_0 (最初的Rx/Tx字节)							MSG_DATA_1							8/16/32	数据	
H'10A+N*32	MSG_DATA_2							MSG_DATA_3							8/16		
H'10C+N*32	MSG_DATA_4							MSG_DATA_5							8/16/32		
H'10E+N*32	MSG_DATA_6							MSG_DATA_7							8/16		
H'110+N*32	0	0	NMC	ATX	DART	MBC[2:0]			0	0	0	0	DLC[3:0]				8/16
H'112+N*32	TimeStamp[15:0] (在SOF的TCNTR值)															16	时戳
H'114+N*32	作为时间基准的发送触发时间(TTT)															16	触发时间

• MB31(在时间触发模式中接收时间基准)

地址	数据总线															存取长度	字段名
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1		
H'100+N*32	IDE	RTR	0	STDID[10:0]										EXTID[17:16]	16/32	控制0	
H'102+N*32	EXTID[15:0]																16
H'104+N*32	IDE	LAFM	0	0	STDID_LAFM[10:0]										EXTID_LAFM[17:16]	16/32	LAFM
H'106+N*32	EXTID_LAFM[15:0]															16	
H'108+N*32	MSG_DATA_0 (最初的Rx/Tx字节)							MSG_DATA_1							8/16/32	数据	
H'10A+N*32	MSG_DATA_2							MSG_DATA_3							8/16		
H'10C+N*32	MSG_DATA_4							MSG_DATA_5							8/16/32		
H'10E+N*32	MSG_DATA_6							MSG_DATA_7							8/16		
H'110+N*32	0	0	NMC	ATX	DART	MBC[2:0]			0	0	0	0	DLC[3:0]				8/16
H'112+N*32	TimeStamp[15:0] (在SOF的TCNTR值)															16	时戳

【注】 1. 灰色表示的位为保留位。写入值为“0”，读取值不一定为“0”，而且没有保证。
2. 因为MCR15位的初始值为“1”，所以信息控制、LAFM的STDID、RTR、IDE、EXTID的顺序和HCAN2不同。

图 23.5 信箱（30、31）的结构

23.3.2 信息控制字段

寄存器名	地址	位	位名	说明
MB[x]. CONTROL0H	H'100+N*32	15	IDE	ID 扩展 区别 CAN 数据帧和远程帧是标准格式还是扩展格式。 0: 标准格式 1: 扩展格式
		14	RTR	远程发送请求 区别数据帧和远程帧。根据是数据帧还是远程帧, 接收 CAN 帧改写此位。 【重要】 如果在 MBC 为“B'001”时将数据帧自动发送 (ATX) 位置位, 就不将 RTR 置位。如果接收远程帧, 就通过对应的 RFPR 位或者 IRR2 位 (远程帧接收中断) 通知 CPU。但是, 因为此模块需要将当前的信息作为数据帧进行发送, 所以 RTR 位不变。 【重要】 在 MBC 为“B'001”时将 ATX 置“1”, 接收到远程帧的信箱自动被设定为发送。此时, 必须将 RTR 置“0”, 以发送数据帧。 0: 数据帧 1: 远程帧
		13	—	保留位 初始值为不定值。写入值总是“0”。
		12 ~ 2	STDID [10:0]	标准 ID 此位设定数据帧和远程帧的 ID (标准 ID)。
		1、0	EXTID [17:16]	扩展 ID 此位设定数据帧和远程帧的 ID (扩展 ID)。
MB[x]. CONTROL0L	H'102+N*32	15 ~ 0	EXTID [15:0]	

• 信箱 0

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	NMC	0	0	MBC[2:0]			0	0	0	0	DLC[3:0]			
初始值:	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R	R	R/W	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

【注】 MBC[1]的值总是“1”。

• 信箱 1 ~ 31

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	NMC	ATX	DART	MBC[2:0]			0	0	0	0	DLC[3:0]			
初始值:	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

寄存器名	地址	位	位名	说明
MB[x]. CONTROL1	H'110+N*32	15、14	—	保留位 读写值都为“0”。
		13	NMC	新信息控制 如果此位被置“0”，RXPR 或者 RFPR 被置位的信箱就不保存新的信息而保持之前的信息，并且将 UMSR 的对应位置位。 如果此位被置“1”，RXPR 或者 RFPR 被置位的信箱就重写新的信息，并且将 UMSR 的对应位置位。 【重要】 如果通过数据帧改写远程帧或者通过远程帧改写数据帧，RXPR 标志和 RFPR 标志（和 USMR 一起）就可能被设定为同一信箱。此时，也重写信箱控制字段中的 RTR 位。 【重要】 在使用时间触发模式时，必须将信箱 31 的 NMC 置“1”，即使不清除 RXPR[31] 也必须设定为能接收全部基准信息。 0: 超限模式 1: 重写模式
	H'110+N*32	12	ATX	数据帧自动发送 如果将此位置“1”并且通过信箱接收远程帧，就保存 DLC 并且 TXPR 自动进行置位。使用由相同信箱改写的 DLC，并且自动发送数据帧。 设定为自动发送的信箱根据信息发送优先级位（MCR2）的设定，按照 ID 优先级或者信箱优先级进行调度。要使用此功能时，必须将 MBC[2:0] 置“B'001”。在使用此功能进行发送时，所用数据长度码（DLC）为接收到的内容，因此远程帧的 DLC 需要对应所要求的数据帧的 DLC。 【重要】 1. 在使用 ATX 并且 MBC 为“B'001”时，和应答信息的情况相同，远程帧的 ID 必须和数据帧的 ID 完全相同，因此不能使用 IDE 位的滤波器。 2. 在使用此功能时，与接收远程帧无关，不将 RTR 位置位。如果接收远程帧，就通过被置位的 PFPR 通知 CPU，但是此模块需要将当前信息作为数据帧进行发送，所以不更改 RTR 位。 3. 有时会不开始自动发送远程帧。必须注意超限状态（NMC 为“0”时的 UMSR 置位）。 4. 如果 ATX 为“1”的信箱因远程帧而变为超限状态，就可能接受旧信息的自动发送请求。 0: 数据帧的自动发送无效 1: 数据帧的自动发送有效 【注】 此位只限于信箱 1～31。在信箱 0 中，此位是保留位。
	11	DART	自动重新发送无效 如果将此位置“1”，就在发生 CAN 总线错误事件或者 CAN 总线仲裁失败时将信息的自动重新发送置为无效。实际上，在使用此功能时，对应的 TXCR 位在开始发送时自动置位。如果将此位置“0”，此模块就通过 TXCR 取消发送请求或者在发送正常结束之前请求发送。 0: 重新发送有效 1: 重新发送无效 【注】 此位只限于信箱 1～31。在信箱 0 中，此位是保留位。	

寄存器名	地址	位	位名	说明
MB[x]. CONTROL1	H'110+N*32	10 ~ 8	MBC[2:0]	信箱配置 如表 23.4 所示，这些位设定各信箱的功能。当 MBC 为 “B'111” 时，信箱无效。即：与 TXPR 或者其他的设定无关，不进行信息的发送和接收。 禁止将 MBC 置 “B'100”、“B'101” 和 “B'110”。如果将 MBC 的值置 “B'000” 或者其他值，就能使用 LAFM 字段。 在将 MBC 设定为接收时，不能将 TXPR 置位。没有硬件保护功能，TXPR 保持被置位的状态。信箱 0 的 MBC[1] 为接收专用，因此被硬件固定为 “1”。
		7 ~ 4	—	保留位 读写值都为 “0”。
		3 ~ 0	DLC[3:0]	数据长度码 通过数据帧对发送的数据字节数（0 ~ 8）进行编码。 0000：数据长度为 0 字节 0001：数据长度为 1 字节 0010：数据长度为 2 字节 0011：数据长度为 3 字节 0100：数据长度为 4 字节 0101：数据长度为 5 字节 0110：数据长度为 6 字节 0111：数据长度为 7 字节 1xxx：数据长度为 8 字节 【注】 x: Don't care

表 23.4 信箱功能的设定

MBC2	MBC1	MBC0	数据帧的发送	远程帧的发送	数据帧的接收	远程帧的接收	说明
0	0	0	能	能	不能	不能	<ul style="list-style-type: none"> 不能使用信箱 0。 能使用时间触发发送。
0	0	1	能	能	不能	能	<ul style="list-style-type: none"> 能由 ATX 使用*。 不能使用信箱 0。 能使用 LAFM。
0	1	0	不能	不能	能	能	<ul style="list-style-type: none"> 能使用信箱 0。 能使用 LAFM。
0	1	1	不能	不能	能	不能	<ul style="list-style-type: none"> 能使用信箱 0。 能使用 LAFM。
1	0	0	禁止设定				
1	0	1	禁止设定				
1	1	0	禁止设定				
1	1	1	信箱无效（初始值）				

【注】 * 为了支持自动重新发送，必须在 MBC 为 “B'001” 并且 ATX 为 “1” 时将 RTR 置 “0”。在将 ATX 置 “1” 后使用时，不能使用 IDE 的滤波器。

23.3.3 本地接收滤波器屏蔽 (LAFM)

在 MBC 为 “B'001”、“B'010” 或者 “B'011” 时，此字段用作接收 LAFM。LAFM 允许信箱接受多个接收 ID。如图 23.6 所示，LAFM 由 2 个 16 位可读写区域构成。

寄存器名	地址	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	存取长度	字段名
MB[0].LAFMH	H'104+N*32	IDE_LAFM	0	0	STDID_LAFM[10:0]												EXTID_LAFM[17:16]	16/32	LAFM
MB[0].LAFML	H'106+N*32	EXTID_LAFM[15:0]															16		

图 23.6 本地接收滤波器屏蔽 (LAFM)

如果将 1 个位设定为 LAFM，就在通过和此模块相同 CAN-ID 搜索信箱时忽视接收的 CAN 的 ID 的对应位。如果清除位，接收的 CAN 的 ID 的对应位必须和设定为保存信箱的 STDID/IDE/EXTID 相同。LAFM 的结构和信箱的信息控制相同。在不使用此功能时，必须写 “0”。

- 【注】
1. 此模块从信箱 31 到信箱 0，搜索相同的 ID。如果搜索到相同的 ID，此模块就立即结束搜索，与 NMC 或者 RXPR/RFPR 标志无关。这表示：即使使用 LAFM，接收的信息也只保存在 1 个信箱中。
 2. 如果接收到一条信息并且发现相同的信箱，就将全部信息保存到信箱。在使用 LAFM 时，将 STDID、RTR、IDE、EXTID 更新为接收信息的 STDID、RTR、IDE、EXTID，因此可能和接收前的设定不同。

寄存器名	地址	位	位名	说明
MB[x]. LAFMH	H'104+N*32	15	IDE_LAFM	IDE 位的滤波器屏蔽位 0: 对应的 IDE 位有效 1: 对应的 IDE 位无效
		14、13	—	保留位 初始值为不定值。写入值总是 “0”。
		12 ~ 2	STDID_LAFM [10:0]	STDID[10:0] 的滤波器屏蔽位 0: 对应的 STDID 位有效 1: 对应的 STDID 位无效
		1、0	EXTID_LAFM [17:16]	EXTID[17:0] 的滤波器屏蔽位 0: 对应的 EXTID 位有效 1: 对应的 EXTID 位无效
MB[x]. LAFML	H'106+N*32	15 ~ 0	EXTID_LAFM [15:0]	

23.3.4 信息数据字段

保存发送和接收的 CAN 信息。MSG_DATA_0 对应发送和接收的第一个数据字节。CAN 总线上的位的排列顺序为 bit7 到 bit0。

- 发送时间触发时的注意事项

如果在 CMAX 不为 “B'111” 并且 MBC[30] 为 “B'000” 时将 TXPR[30] 置 “1”，就能将信箱 30 设定为用于发送时间基准。此时，必须将 DLC 置大于 “0” 的值并且将 RTR 置 “0”（TTCAN Level 1 的规定）。在信箱 30 开始发送时，在将周期计数器的值（CCR）填入数据字段起始字节 MSG_DATA_0[5:0] 后发送信息。

MSG_DATA_0[7:6] 发送保存在信箱的值。

当需要发送 Next_is_Gap 时，用户能通过将 MSG_DATA_0[7] 置 “1” 进行发送。

必须注意：此时，CCR 的值在被发送的帧中，但是不保存到信箱 30 的信息数据字段。

如果在 CMAX 不为 “B'111” 并且 MBC[31] 为 “B'011” 时进行设定，就将信箱 31 设定为用于接收时间基准。如果接收有效的基准信息（ID 相同并且 DLC > 0），此模块就通过改写 RFMK 和周期计数器（CCR）使 Cycle_Time 和 Basic_Cycle 同步。

MB30、31		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
H'108+N*32	Next_is_Gap/Cycle_Counter(最初的Rx/Tx字节)	MSG_DATA_1										8/16/32位		数据					
H'10A+N*32	MSG_DATA_2	MSG_DATA_3										8/16位							
H'10C+N*32	MSG_DATA_4	MSG_DATA_5										8/16/32位							
H'10E+N*32	MSG_DATA_6	MSG_DATA_7										8/16位							

图 23.7 信息数据字段

23.3.5 时戳

保存发送 / 接收信息记录的时戳。时戳能用于监视是否按计划发送或者接收了信息。

- 时戳

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TS15	TS14	TS13	TS12	TS11	TS10	TS9	TS8	TS7	TS6	TS5	TS4	TS3	TS2	TS1	TS0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

(1) 接收信息

根据时间触发控制寄存器 0 (TTCR0) bit14 的设定, 信箱 0 ~ 15 的接收信息时戳在接收信息的 SOF 处捕捉并且保存 CYCTR[15:0] (周期时间寄存器) 的值或者 CCR[5:0] (周期计数器) +CYCTR[15:6] (周期时间寄存器) 的值。

信箱 30 和信箱 31 的接收信息时戳在接收信息的 SOF 处捕捉并且保存定时器计数器寄存器 (TCNTR) 的值。

(2) 发送信息

根据时间触发控制寄存器 0 (TTCR0) 的 bit14 的设定, 信箱 1 ~ 15 的发送信息时戳在发送信息的 SOF 处捕捉并且保存 CYCTR[15:0] (周期时间寄存器) 的值或者 CCR[5:0] (周期计数器) +CYCTR[15:6] (周期时间寄存器) 的值。

信箱 30 和信箱 31 的发送信息时戳在发送信息的 SOF 处捕捉定时器的计数器寄存器 (TCNTR) 的值。

【重要】 时戳保存在暂存器。如果正常进行发送或者接收, 就将该值复制到信箱的字段。另外, 在因超限而将 UMSR[N] 置位的同时, CPU 清除 RXPR[N]/RFPR[N], 此时可能只更新时戳。因此, 能读到将 RXPR[N]/RFPR[N] 清除前的正确的时戳值。

23.3.6 发送触发时间 (TTT) 和时间触发控制

在信箱 24 ~ 29 中，如果在时间触发模式 (CMAX 不为 “B'111”) 中将 MBC 置 “B'000”，就将发送触发时间用作决定时间窗口边界的 Time_Mark。

发送触发时间 (TTT) 和时间触发控制 (TT 控制) 由以下 2 个 16 位读写寄存器构成。

在信箱 30 中，没有时间触发控制 (TT 控制)，用作 Time_Ref。

当不将信箱 24 ~ 30 用于时间触发模式的发送时，能用作接收信箱。但是，在使用时间触发模式时，不会成为事件触发发送的对象。

- 发送触发时间 (TTT)

发送触发时间 (TTT) 指定周期时间内的开始发送信息的时间。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TTT15	TTT14	TTT13	TTT12	TTT11	TTT10	TTT9	TTT8	TTT7	TTT6	TTT5	TTT4	TTT3	TTT2	TTT1	TTT0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- 时间触发控制 (TT 控制)

时间触发控制 (TT 控制) 指定窗口的属性、开始发送的系统矩阵内的周期计数 (Basic Cycle) 以及周期发送的频率。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TTW[1:0]		Offset[5:0]					0	0	0	0	0	rep_factor[2:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

支持时间触发模式的全部信箱之间的不同点如图 23.8 所示。

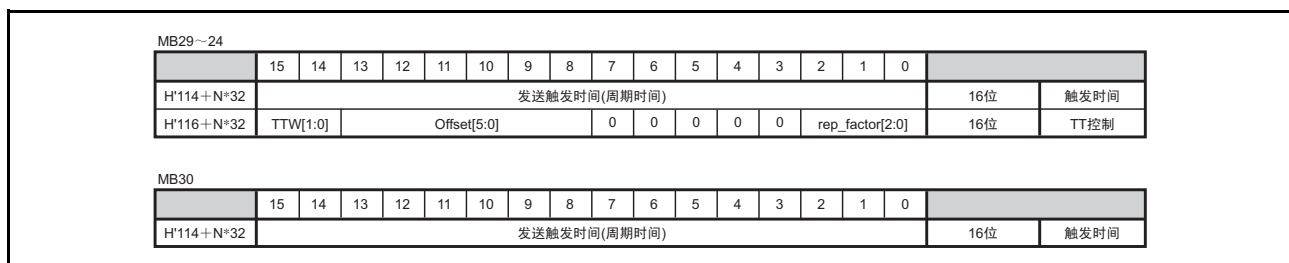


图 23.8 发送触发控制字段

- TTW[1:0] (时间触发窗口)

表示时间窗口的属性。必须将 TTW 置 “B'11”，结束通过将 TTW 置 “B'10” 而开始的合并仲裁窗口 (Merged Arbitrating Window)。在合并仲裁窗口的开始到结束的期间，能使用多个 TTW 为 “B'10” 的信息。

TTW[1]	TTW[0]	说明
0	0	互斥窗口 : exclusive window (初始值)
0	1	仲裁窗口 : arbitrating window
1	0	合并仲裁窗口的开始 : Start of Merged arbitrating window
1	1	合并仲裁窗口的结束 : End of Merged arbitrating window

如果 TTT 寄存器的值和 CYCTR 的值相同并且 Offset 的值和 CCR 的值相同，就从对应的信箱尝试发送。为了使此功能有效，必须将 CMAX 置 “B'111” 以外的值，使定时器 (TCNTR) 运行 (TTCR0 的 bit15 为 “1”)，

将对应信箱的 MBC 置 “B'000” 并且将对应的 TXPR 位置 “1”。因为在通过软件将 TXPR 的位置位后继续进行周期发送，所以此模块不消除 TXPR 的位（信箱 24 ~ 30）。

要停止周期发送时，必须通过 TXCR 清除 TXPR。此时，如果在发送结束后不立即清除 TXACK，就可能将相同信箱的 TXACK 和 ABACK 都置位（参照图 23.9）。

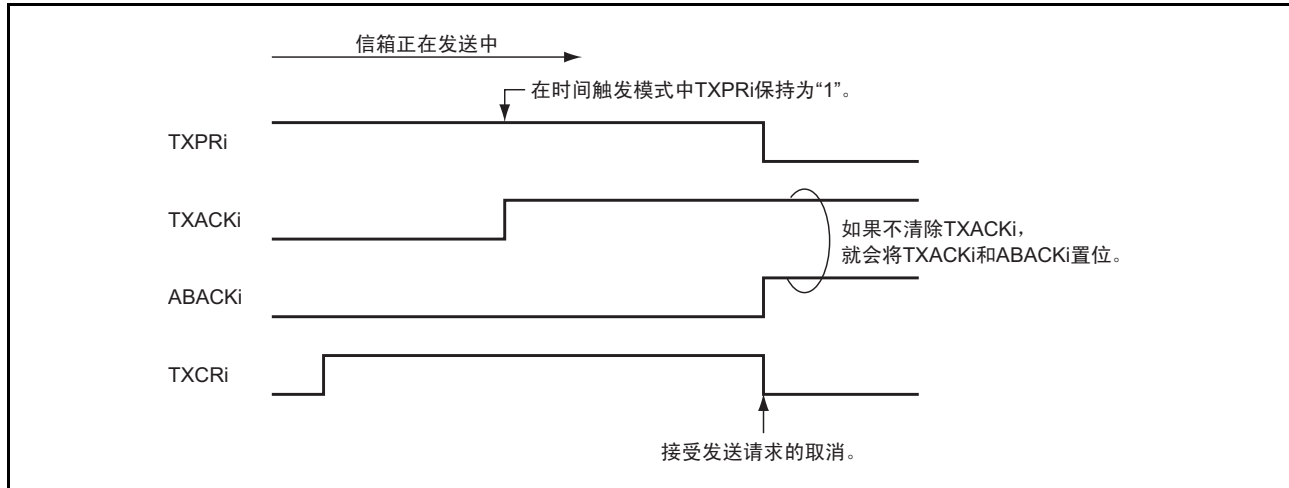


图 23.9 发送时间触发时的 TXACK 和 ABACK

在信箱 30 中，TTW 固定为 “B'01”，Offset 固定为 “B'000000” 并且 rep_factor 固定为 “B'000”。rep_factor 和 Offset 的组合如下表所示。

rep_factor	说明
B'000	每 1 个周期计数（初始值）
B'001	每 2 个周期计数
B'010	每 4 个周期计数
B'011	每 8 个周期计数
B'100	每 16 个周期计数
B'101	每 32 个周期计数
B'110	每 64 个周期计数（每个系统矩阵进行 1 次）
B'111	保留

Offset 字段决定时间触发信箱能开始发送信息的第一个周期的计数器值。

Offset	说明
B'000000	偏移量 (Offset) = 第 1 个周期计数（初始值）
B'000001	偏移量 (Offset) = 第 2 个周期计数
B'000010	偏移量 (Offset) = 第 3 个周期计数
B'000011	偏移量 (Offset) = 第 4 个周期计数
B'000100	偏移量 (Offset) = 第 5 个周期计数
...	...
B'111110	偏移量 (Offset) = 第 63 个周期计数
B'111111	偏移量 (Offset) = 第 64 个周期计数

必须满足以下关系：

$$\text{Cycle_Count_Maximum} + 1 \geq \text{Repeat_Factor} > \text{Offset}$$

$$\text{Cycle_Count_Maximum} = 2^{\text{CMAX}-1}$$

$$\text{Repeat_Factor} = 2^{\text{rep_factor}}$$

CMAX、Repeat_Factor 和 Offset 为寄存器的值。

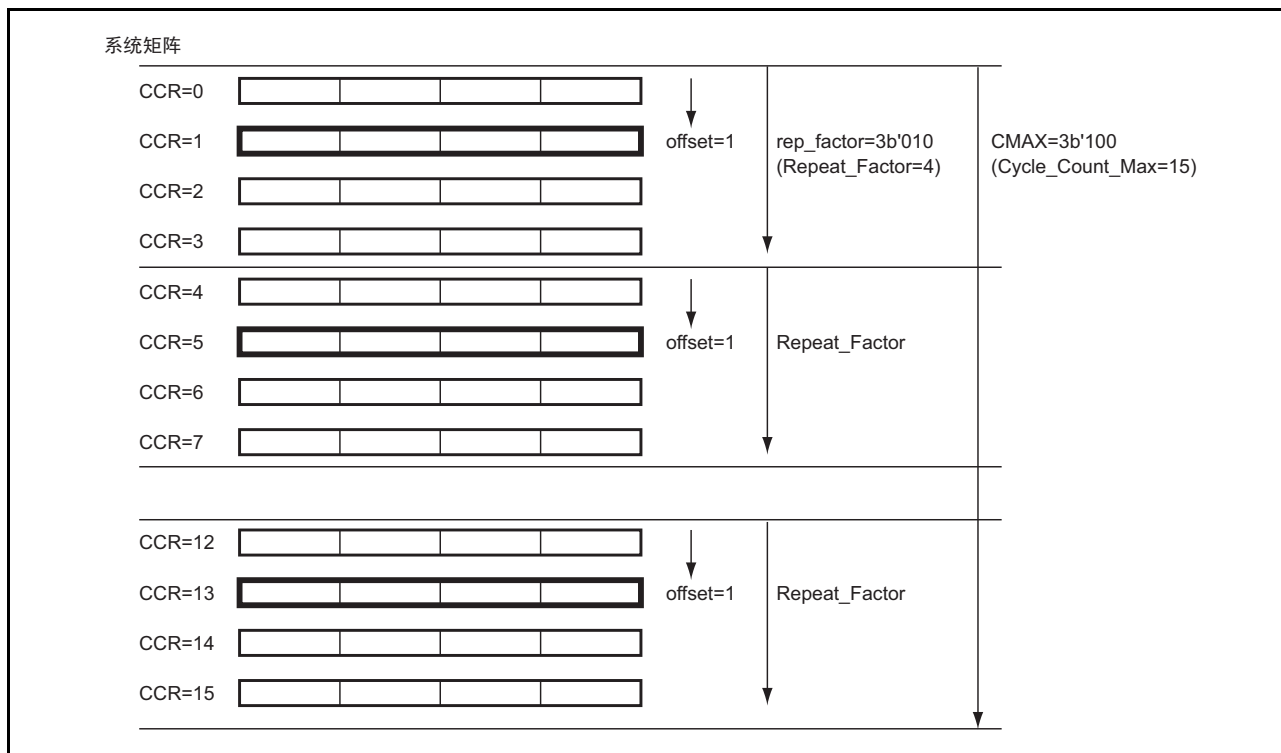


图 23.10 系统矩阵

必须将发送触发时间设定为升序。发送触发时间之间的差必须满足：

$$\text{TTT} (\text{信箱 } i) - \text{TTT} (\text{信箱 } i-1) > \text{TEW} + \text{最大帧长} + 9$$

23.4 控制寄存器

说明控制寄存器。控制寄存器只能进行字（16 位）存取。
控制寄存器如表 23.5 所示。

表 23.5 控制寄存器的结构

寄存器名	略称	地址	存取长度（位）
主控控制寄存器	MCR	H'000	16
通用状态寄存器	GSR	H'002	16
位配置寄存器 1	BCR1	H'004	16
位配置寄存器 0	BCR0	H'006	16
中断请求寄存器	IRR	H'008	16
中断屏蔽寄存器	IMR	H'00A	16
发送错误计数器 / 接收错误计数器	TEC/REC	H'00C	16

23.4.1 主控控制寄存器（MCR）

MCR 是 16 位可读写寄存器，控制此模块。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MCR15	MCR14	—	—	—	TST[2:0]		MCR7	MCR6	MCR5	—	—	MCR2	MCR1	MCR0	
初始值:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	MCR15	1	R/W	ID 排序更改 此位能设定和 HCAN2 兼容的信息控制以及 LAFM 的 STDID、RTR、IDE、EXTID 的顺序。此位只能在复位模式中进行更改。有关 ID 排列的顺序请参照图 23.11。 0: 此模块和 HCAN2 的顺序相同 1: 此模块和 HCAN2 的顺序不同
14	MCR14	0	R/W	自动总线 OFF 停机 如果同时设定此位和 MCR6, 就在此模块进入总线 OFF 状态时, MCR1 立即自动置位。此位只能在复位模式中进行更改。 0: 通过正常的返回顺序 (128×11 隐性位) 保持此模块总线的 OFF 状态 1: 如果将 MCR6 置位, 此模块就在进入总线 OFF 状态后立即进入停机模式。
13~11	—	全 0	R	保留位 读写值都为“0”。

位	位名	初始值	R/W	说明
10 ~ 8	TST[2:0]	000	R/W	<p>测试模式</p> <p>此位设定测试模式是否有效。必须注意：在启动测试模式前需要将此模块设定为停机模式或者复位模式。这是为了避免在转移到测试模式时影响正在进行的发送和接收。详细内容请参照“23.7.2 测试模式的设定”。</p> <p>测试模式只用于诊断和测试，不能用于此模块的正常运行。</p> <p>000：正常模式 001：监听专用模式（接收专用模式） 010：自测试模式 1（外部） 011：自测试模式 2（内部） 100：写错误计数器 101：错误被动模式 110：禁止设定 111：禁止设定</p>
7	MCR7	0	R/W	<p>自动唤醒模式</p> <p>此位设定 CAN 睡眠模式的自动唤醒模式是否有效。如果设定此位，此模块就在检测到 CAN 总线有效（显性位）后自动解除 CAN 睡眠模式（MCR5）。如果清除此位，此模块就不自动解除 CAN 睡眠模式。</p> <p>此模块不能保存唤醒的信息。</p> <p>0：将通过 CAN 总线有效进行的自动唤醒模式置为无效 1：将通过 CAN 总线有效进行的自动唤醒模式置为有效</p> <p>【注】 不能在 CAN 睡眠模式中更改 MCR7 位。</p>
6	MCR6	0	R/W	<p>总线 OFF 时停机</p> <p>此位设定在总线 OFF 时 MCR1 被置位的情况下是否立即进入停机模式。此位只能在复位或者停机模式中进行更改。必须注意：如果在总线 OFF 时进入停机模式，CAN 控制器也立即恢复到错误主动模式。</p> <p>0：在总线 OFF 时不进入停机模式而等待恢复顺序的结束 1：在总线 OFF 时通过设定 MCR1 进行的停机模式转移有效</p>
5	MCR5	0	R/W	<p>CAN 睡眠模式</p> <p>此位设定 CAN 睡眠模式的转移是否有效。如果在此模块进入停机模式时将此位置位，CAN 睡眠模式的转移就有效。在进入停机模式后允许设定此位。在 CAN 睡眠模式期间，2 个错误计数器（REC 和 TEC）不变。有 2 种解除 CAN 睡眠模式的方法：</p> <ul style="list-style-type: none"> • 给 MCR5 位写“0”。 • 在 MCR7 有效时，检测到 CAN 总线上的显性位。 <p>如果自动睡眠解除模式无效，此模块就在 CAN 睡眠模式结束前忽视全部 CAN 总线的运行。在解除 CAN 睡眠模式时，此模块在进入 CAN 总线运行前检查 11 个隐性位，取得和 CAN 总线的同步。也就是说，在使用第 2 种方法时，此模块无法接收最初的信息。同样，CAN 收发器在待机模式结束时也无法接收最初的信息，因此需要采用此方法设计软件。</p> <p>只能在 CAN 睡眠模式中存取 MCR、GSR、IRR、IMR 寄存器，详细内容请参照“23.7.1(3) CAN 睡眠模式”。</p> <p>0：解除 CAN 睡眠模式 1：CAN 睡眠模式的转移有效</p> <p>【注】 此模块能在最初设定为停机模式后转移到 CAN 睡眠模式。因此，在转移到 CAN 睡眠模式前，CPU 能清除处于等待状态的全部中断。此模块必须在清除全部中断的同时从停机模式转移到 CAN 睡眠模式（就是说，在给 MCR5 写“1”的同时给 MCR1 写“0”。）</p>

位	位名	初始值	R/W	说明
4、3	—	全 0	R	保留位 读写值都为“0”。
2	MCR2	0	R/W	<p>信息发送优先级</p> <p>此位选择等待中的发送数据的发送顺序。如果将此位置“1”，就按照发送等待寄存器（TXPR）的位的顺序发送数据。</p> <p>最优先开始信箱 31 的发送，一直进行到信箱 1 为止（信箱被设定为发送信箱的情况）。</p> <p>必须注意：此功能不能用于信箱 24 ~ 30 的时间触发发送。如果清除此位，就按照 ID 优先级（根据内部仲裁的运行结果）发送全部发送信息。最优先的信息持有最小数字的仲裁字段（STDID+IDE 位 +EXTID（IDE 为“1”的情况）+RTR 位），先发送。</p> <p>内部仲裁包含 RTR 位和 IDE 位（内部仲裁的运行方法和在 2 个 CAN 节点之间 CAN 总线上的仲裁同时开始转移的情况相同）。</p> <p>此位只能在复位或者停机模式中进行更改。</p> <p>0：按照信息 ID 优先级进行发送 1：按照信箱号的顺序（信箱 31→信箱 1）进行发送</p>
1	MCR1	0	R/W	<p>停机请求</p> <p>如果将此位置位，CAN 控制器就在当前的运行结束后进入停机模式（在此，从 CAN 总线分离）。此模块的停机模式保持到此位被清除为止。</p> <p>在停机模式中，CAN 接口与 CAN 总线的运行无关，不保存也不发送信息。除了用于将停机状态通知 CPU 的 IRR0 和 GSR4 以外，保持用户寄存器（包含信箱内容和 TEC/REC）的内容。</p> <p>在 CAN 总线为空闲状态或者间歇状态时，与 MCR6 无关，此模块在 1 位时间内变为停机模式。如果将 MCR6 置位，总线 OFF 中的停机请求也在 1 位时间内运行。对于其他情况，在总线 OFF 恢复顺序结束前不进入停机模式。如果变为停机模式，就通过 IRR0 和 GSR4 进行通知。</p> <p>如果将 MCR14 和 MCR6 置位，就在此模块转移到总线 OFF 状态时立即自动将此位置位。</p> <p>在停机模式中，因为此模块与总线运行无关，所以除了能设定定时以外，还能更改此模块的结构。为了再次加入 CAN 总线运行，需要将此位清“0”。在清除后，此模块等待检测 11 个隐性位，然后加入 CAN 总线。</p> <p>0：清除停机模式的请求 1：请求停机模式的转移</p> <p>【注】</p> <ol style="list-style-type: none"> 1. 在发行停机请求后并且转移到停机模式前（通过 IRR0 和 GSR4 进行通知），CPU 不能存取 TXPR 和 TXCR，也不能清除此位。在将 MCR1 置位后，只能在进入停机模式后或者通过复位（软件或者硬件）解除此状态。 2. 只有在给 BCR1 寄存器和 BCR0 寄存器设定了适当的波特率时，才能转移到停机模式或者从停机模式转移。

位	位名	初始值	R/W	说明
0	MCR0	1	R/W	<p>复位请求</p> <p>此位控制此模块的复位。当此位从“0”变为“1”时，CAN 控制器进入复位程序，对内部逻辑进行初始化。然后，为了通知复位模式，将 GSR3 和 IRR0 置位。对全部用户寄存器进行初始化。</p> <p>在此为被置位的期间，能重新构成此模块。为了加入 CAN 总线，需要在给此位写“0”后清除。在清除后，此模块等待检测 11 个隐性位，然后加入 CAN 总线。</p> <p>为采样 CAN 总线上的值，需要设定适当的波特率。在上电复位后，总是将此位和 GSR3 置位。这表示请求复位并且需要构成此模块。</p> <p>复位请求相当于上电复位，通过软件进行控制。</p> <p>0: 清除复位模式的请求 [清除条件] 在对此模块进行复位后写“0”时 1: 请求 CAN 接口复位模式的转移</p>

• MCR15(ID排序)=0																存取长度	字段名	
地址	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
H'100+N*32	0	STDID[10:0]										RTR	IDE	EXTID[17:16]	16/32	控制0		
H'102+N*32	EXTID[15:0]																16	
H'104+N*32	0	STDID_LAFM[10:0]										0	IDE_LAFM	EXTID_LAFM [17:16]	16/32	LAFM字段		
H'106+N*32	EXTID_LAFM[15:0]																16	
• MCR15(ID排序)=1																存取长度	字段名	
地址	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
H'100+N*32	IED	RTR	0	STDID[10:0]										EXTID[17:16]		16/32	控制0	
H'102+N*32	EXTID[15:0]															16		
H'104+N*32	IDE_LAFM	0	0	STDID_LAFM[10:0]										EXTID_LAFM [17:16]		16/32	LAFM字段	
H'106+N*32	EXTID_LAFM[15:0]															16		

图 23.11 ID 排序

23.4.2 通用状态寄存器 (GSR)

GSR 是 16 位只读寄存器，表示此模块的状态。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	GSR5	GSR4	GSR3	GSR2	GSR1	GSR0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 6	—	全 0	R	保留位 读写值都为“0”。
5	GSR5	0	R	<p>错误被动状态</p> <p>表示 CAN 接口是否为错误被动状态。当此模块变为错误被动状态时，就立即将此位置“1”；当再次变为错误主动状态时，就清除此位。这表示在错误被动状态下或者在总线 OFF 状态下，GSR5 保持为“1”。因此，要得知正确的状态时，必须检查 GSR5 和 GSR0。</p> <p>0: 此模块不为错误被动状态或者总线 OFF 状态 [清除条件] 在此模块处于错误被动状态期间 1: 此模块为错误被动状态 (GSR0 为“0”) 或者总线 OFF 状态 (GSR0 为“1”) [置位条件] 当 $TEC \geq 128$ 或者 $REC \geq 128$ 时，或者在测试模式中选择错误被动模式时</p>
4	GSR4	0	R	<p>停机 / 睡眠状态</p> <p>表示 CAN 控制器是否为停机 / 睡眠状态。必须注意：此标志的清除时间和 IRR12 的置位时间不同。此标志是反映 CAN 控制器状态的标志，而不是反映此模块全部状态的标志。如果结束 CAN 睡眠模式并且清除 MCR5，此模块就变为可存取状态。在经过传送时钟 2 个位后，结束 CAN 控制器的 CAN 睡眠模式。</p> <p>0: 此模块既不是停机模式也不是 CAN 睡眠模式 1: 是停机模式 (MCR1 为“1”) 或者 CAN 睡眠模式 (MCR5 为“1”) [置位条件] • 当 MCR1 被置位并且 CAN 总线为间歇状态或者空闲状态时 • 当 MCR5 被置位并且此模块为停机模式时 • 当 MCR14 和 MCR6 被置位并且此模块转移到总线 OFF 状态时</p>
3	GSR3	1	R	<p>复位状态</p> <p>表示此模块是否为复位状态。</p> <p>0: 此模块不是复位状态 1: 此模块是复位状态 [置位条件] 在对此模块进行软件复位或者硬件复位后</p>

位	位名	初始值	R/W	说明
2	GSR2	1	R	<p>信息发送标志</p> <p>此标志向 CPU 表示以下某种状态：</p> <ul style="list-style-type: none"> • 此模块处于总线 OFF 状态。 • 在发送信息。 • 正在发送因发送过程中检测到错误而产生的错误 / 过载标志。 <p>TXACK 的置位时序和 GSR2 的清除时序不同。通过 EOF 的第 7 位将 TXACK 置位。当没有等待发送的信息时，在间歇的第 3 位将 GSR2 置位。也可通过仲裁失败、总线空闲、接收、复位、停机转移进行置位。</p> <p>0: 此模块为总线 OFF 状态或者正在发送信息 1: [置位条件] 不是总线 OFF 状态并且也没有发送信息</p>
1	GSR1	0	R	<p>发送 / 接收警告标志</p> <p>此位表示错误警告的标志。</p> <p>0: [清除条件] 当 $TEC < 96$ 并且 $REC < 96$ 时或者当总线 OFF 时 1: [置位条件] 当 $96 \leq TEC < 256$ 或者 $96 \leq REC < 256$ 时</p> <p>【注】 因为 REC 对总线 OFF 恢复顺序所需的 11 个隐性位的重复次数进行计数，所以在总线 OFF 状态下递增。但是，此位在总线 OFF 状态下不被置位。</p>
0	GSR0	0	R	<p>总线 OFF 标志</p> <p>此位表示此模块为总线 OFF 状态。</p> <p>0: [清除条件] 在从总线 OFF 状态恢复后或者在进行硬件 / 软件复位后 1: [置位条件] $TEC \geq 256$ (总线 OFF 状态)</p>

23.4.3 位配置寄存器 0、1 (BCR0、BCR1)

BCR0 和 BCR1 分别是 16 位可读写寄存器，设定 CAN 位定时参数和 CAN 接口的波特率预分频器。时间量子定义如下：

$$\text{时间量子} = 2 \times \text{BRP} / f_{\text{clk}}$$

BRP (波特率预分频器) 为 BCR0 保存值 +1 的值， f_{clk} 为外围时钟 0 的频率。

(1) BCR1

有关 TSEG1 和 TSEG2 的设定，请参照表 23.6。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG1[3:0]			—	TSG2[2:0]			—	—	SJW[1:0]		—	—	—	BSP	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W

位	位名	初始值	R/W	说明
15 ~ 12	TSG1[3:0]	0000	R/W	时间段 1 为了用正相误差补偿 CAN 总线上的边沿，这些位用于设定段 TSEG1 (=PRSEG+PHSEG1)。能设定 4 ~ 16 个时间量子。 0000: 禁止设定 0001: 禁止设定 0010: 禁止设定 0011: PRSEG+PHSEG1=4 个时间量子 0100: PRSEG+PHSEG1=5 个时间量子 : : 1111: PRSEG+PHSEG1=16 个时间量子
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	TSG2[2:0]	000	R/W	时间段 2 为了用负相误差补偿 CAN 总线上的边沿，这些位用于设定段 TSEG2 (=PRSEG2)。能设定 2 ~ 8 个时间量子。 000: 禁止设定 001: PHSEG2=2 个时间量子 (根据条件禁止设定。请参照表 23.6) 010: PHSEG2=3 个时间量子 011: PHSEG2=4 个时间量子 100: PHSEG2=5 个时间量子 101: PHSEG2=6 个时间量子 110: PHSEG2=7 个时间量子 111: PHSEG2=8 个时间量子
7、6	—	全 0	R	保留位 读写值都为“0”。

位	位名	初始值	R/W	说明
5、4	SJW[1:0]	00	R/W	再同步转移宽度 设定同步转移宽度。 00: 同步转移宽度 =1 个时间量子 01: 同步转移宽度 =2 个时间量子 10: 同步转移宽度 =3 个时间量子 11: 同步转移宽度 =4 个时间量子
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	BSP	0	R/W	位采样点 此位设定采样数据的点。 0: 在一处进行位采样 (时间量子 1 的最后) 1: 在三处进行位采样 (PHSEG1 的最后 3 个时钟周期的上升沿)

(2) BCR0

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	BRP[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 8	—	全 0	R	保留位 读写值都为“0”。
7 ~ 0	BRP[7:0]	00000000	R/W	波特率预分频器 这些位设定对应 1 个时间量子的外围时钟 0 的个数。 0000000: 2× 外围时钟 0 0000001: 4× 外围时钟 0 0000010: 6× 外围时钟 0 : : 2×(寄存器的值 +1)× 外围时钟 0 1111111: 512× 外围时钟 0

- 有关位配置寄存器

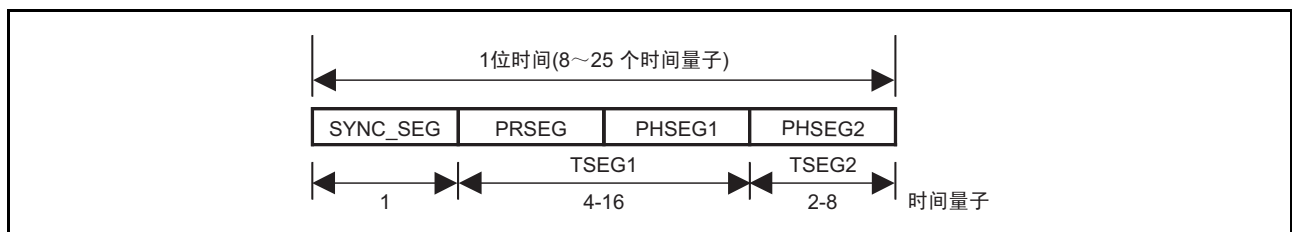


图 23.12 1 位时间的结构

SYNC_SEG: 和 CAN 总线上的节点同步的段
(在此段产生正常的位边沿转移)

PRSEG : 调整网络间物理延迟的段

PHSEG1 : 相位漂移 (正方向) 的缓冲段 (在同步或者再次同步时, 进行扩展)

PHSEG2 : 定位漂移 (负方向) 的缓冲段 (在同步或者再次同步时, 进行缩小)

TSEG1 : TSG1+1

TSEG2 : TSG2+1

位速率的计算如下:

$$\text{位速率} = \text{fclk} / (2 \times (\text{BRP} + 1) \times (\text{TSEG1} + \text{TSEG2} + 1))$$

BRP 由寄存器的值提供, TSEG1 和 TSEG2 是上表中计算的值而不是实际设定值。上述位速率计算式的时间段的“+1”是根据 SYNC_SEG 的 1 个时间量子得出的。

fclk= 外围时钟 0

设定 BCR 时的限制事项:

$$\text{TSEG1}(\text{Min.}) > \text{TSEG2} \geq \text{SJW}(\text{Max.}) \quad (\text{SJW}=1 \sim 4)$$

$$8 \leq \text{TSEG1} + \text{TSEG2} + 1 \leq 25 \text{ 个时间量子} \quad (\text{TSEG1} + \text{TSEG2} + 1 \neq 7)$$

$$\text{TSEG2} \geq 2$$

如果位配置寄存器 TSEG1 和 TSEG2 的值在表 23.6 所示的设定值的范围内, 就满足上述限制事项。表 23.6 的值表示 SJW 能设定的值。“不能”表示不能进行 TSEG1 和 TSEG2 的组合。

表 23.6 TSG 和 TSEG 的设定

		001	010	011	100	101	110	111	TSG2
		2	3	4	5	6	7	8	TSEG2
TSG1	TSEG1								
0011	4	不能	1-3	不能	不能	不能	不能	不能	
0100	5	1-2	1-3	1-4	不能	不能	不能	不能	
0101	6	1-2	1-3	1-4	1-4	不能	不能	不能	
0110	7	1-2	1-3	1-4	1-4	1-4	不能	不能	
0111	8	1-2	1-3	1-4	1-4	1-4	1-4	不能	
1000	9	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1001	10	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1010	11	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1011	12	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1100	13	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1101	14	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1110	15	1-2	1-3	1-4	1-4	1-4	1-4	1-4	
1111	16	1-2	1-3	1-4	1-4	1-4	1-4	1-4	

例 1: 在 fclk 为 25MHz 并且位速率为 250kbps 时, 满足 BRP=4, TSEG1=5, TSEG2=4 的条件。
此时, 给 BCR1 写 “H'4300”, 给 BCR0 写 “H'0004”。

例 2: 在 fclk 为 33MHz 并且位速率为 500kbps 时, 满足 BRP=2、TSEG1=6、TSEG2=4 的条件。
此时, 给 BCR1 写 “H'5300”, 给 BCR0 写 “H'0002”。

23.4.4 中断请求寄存器 (IRR)

IRR 是 16 位可读写寄存器，由各中断源的状态标志构成。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRR15	IRR14	IRR13	IRR12	IRR11	IRR10	IRR9	IRR8	IRR7	IRR6	IRR5	IRR4	IRR3	IRR2	IRR1	IRR0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W

位	位名	初始值	R/W	说明
15	IRR15	0	R/W	定时器比较匹配中断 1 表示定时器的比较匹配寄存器 1 (TCMR1) 发生比较匹配。当 TCMR1 的设定值和周期时间相同时 (TCMR1=CYCTR)，将此位置位。 0: TCMR1 定时器没有发生比较匹配 [清除条件] 写“1”。 1: TCMR1 定时器发生比较匹配 [置位条件] TCMR1 和周期时间相同 (TCMR1=CYCTR)
14	IRR14	0	R/W	定时器比较匹配中断 0 表示定时器的比较匹配寄存器 0 (TCMR0) 发生比较匹配。当 TCMR0 的设定值和本地时间相同时 (TCMR0=TCNTR)，将此位置位。 0: TCMR0 定时器没有发生比较匹配 [清除条件] 写“1”。 1: TCMR0 定时器发生比较匹配 [置位条件] TCMR0 和定时器的值相同 (TCMR0=TCNTR)
13	IRR13	0	R/W	定时器超限中断 / Next_is_Gap 接收中断 / 信息错误中断 此中断的含义因此模块的模式而不同，表示以下状态： <ul style="list-style-type: none"> 在事件触发模式 (含测试模式) 中运行时，定时器 (TCNTR) 发生超限。 在时间触发模式 (含测试模式) 中运行时，Next_is_Gap 接收所设置的时间基准信息。 在测试模式中发生信息错误。但是，即使在测试模式中发生信息过载，也不将此位置位。 0: 在事件触发模式 (包括测试模式) 中定时器 (TCNTR) 没有发生超限 在时间触发模式 (包括含测试模式) 中，没有接收到带 Next_is_Gap 的时间基准信息。 在测试模式中发生信息错误。 [清除条件] 写“1”。 1: [置位条件] 在事件触发模式 (包括测试模式) 中定时器 (TCNTR) 发生超限并且从 H'FFFF 变为 H'0000 在定时触发模式 (包括测试模式) 中，接收到带 Next_is_Gap 的时间基准信息。 在信息测试模式中发生信息错误。

位	位名	初始值	R/W	说明
12	IRR12	0	R/W	<p>CAN 睡眠模式中总线有效</p> <p>此位表示 CAN 总线是否有效。在此模块为 CAN 睡眠模式时，如果在 CAN 总线上检测到显性位，就将此位置位。如果给此位写“1”，就清除中断，忽视写“0”。当不使用自动唤醒功能并且不接受此中断时，必须通过对应的屏蔽位将此中断置为无效。如果在不使用自动唤醒功能时请求此中断，就必须从 CAN 睡眠模式返回后清除此位。这是为了防止因接收线上的新的下降沿而再次将中断置位。</p> <p>必须注意：此中断的置位时间和 GSR4 的清除时间不同。</p> <p>0: 总线空闲状态 [清除条件] 写“1”。</p> <p>1: 在 CAN 睡眠模式中检测到 CAN 总线有效 [置位条件] 在 CAN 睡眠模式中，在 CRxn 上检测到显性位的状态变化。</p>
11	IRR11	0	R/W	<p>定时器的比较匹配中断 2</p> <p>表示定时器的比较匹配寄存器 2 (TCMR2) 发生比较匹配。当 TCMR2 的设定值和周期时间相同时 (TCMR2=CYCTR)，就将此位置位。</p> <p>0: TCMR2 定时器没有发生比较匹配 [清除条件] 写“1”。</p> <p>1: TCMR2 定时器发生比较匹配 [置位条件] TCMR2 和周期时间相同 (TCMR2=CYCTR)。</p>
10	IRR10	0	R/W	<p>系统矩阵开始中断</p> <p>表示下一个系统矩阵的开始。当 CCR 为“0”时，在结束时间基准信息的发送 / 接收时将此位置位；当 CMAX 为“0”时，在每次周期计数 (Basic Cycle) 时将此中断置位。</p> <p>0: 不是新系统矩阵的开头 [清除条件] 写“1”。</p> <p>1: 周期计数器为“0” [置位条件] 在 CMAX 不为“B'111”而 CCR 为“0”时结束时间基准信息的发送 / 接收。</p>
9	IRR9	0	R	<p>信息超限 / 重写中断标志</p> <p>此位表示：虽然接收到信息，但是因对应的 RXPR 或者 RFPR 被置“1”而不读信箱内的信息，由于没有通过 CPU 清除此位，因此根据 NMC 位的设定，放弃 (超限) 或者重写新接收的信息。如果给 UMSR 寄存器的对应位写“1”，就清除此位。另外，也可通过给 MBIMR 对应的全部位写“1”清除此位。此位的写操作无效。</p> <p>0: 没有信息超限 / 重写通知 [清除条件] 清除 UMSR 的全部位或者将与被置位的全部 UMSR 的对应 MBIMR 置位。</p> <p>1: 因发生超限而放弃接收信息，或者重写信息。 [置位条件] 在对应的 PXPR 或者 RFPR 为“1”并且 MBIMR 为“0”时接收信息。</p>

位	位名	初始值	R/W	说明
8	IRR8	0	R	<p>信箱空中断标志</p> <p>如果正常发送 1 个发送信息（对应的 TXACK 标志被置位）或者中止发送（取消发送的信息对应的 ABACK 标志被置位），就将此位置位。此时，在事件触发模式中，清除对应的 TXPR，信箱能接收下一个用于发送的信息数据。</p> <p>在时间触发模式中，为了连续进行每个程控周期计数（Basic Cycle）的发送，在发送结束后不清除信箱 24 ~ 30 的 TXPR。</p> <p>实际上，通过未被 MBIMR 标志屏蔽的 TXACK 位和 ABACK 位的 OR 信号，将此位置位。因此，如果清除全部 TXACK 位和 ABACK 位，就自动清除此位。另外，也能通过给 MBIMR 对应的全部位写“1”来清除此位。此位的写操作无效。</p> <p>0: 没有在发送或者没有在处理取消发送的信息 [清除条件] 清除全部 TXACK 位和 ABACK 位或者将被置位的全部 TXACK 位和 ABACK 位对应的 MBIMR 置位。 1: 发送信息或者中止发送（取消发送）并且为能保存下一个信息（在时间触发模式中，信箱 24 ~ 30 只在中止时能保存下一个信息） [置位条件] 当将 TXACK 位或者 ABACK 位置位时（对应的 MBIMR 为“0”时）</p>
7	IRR7	0	R/W	<p>过载帧</p> <p>此标志表示此模块检测到过载帧的发送。如果给 IRR7 写“1”，就清除此位。忽视写“0”。</p> <p>0: [清除条件] 写“1” 1: [置位条件] 检测到过载条件</p>
6	IRR6	0	R/W	<p>总线 OFF 中断标志</p> <p>在此模块为总线 OFF 状态时或者从总线 OFF 状态返回到错误主动状态时，将此位置位。因此，中断源为节点的 $TEC \geq 256$ 或者结束总线 OFF 恢复顺序（连续 128 次接收到 11 个隐性位）或者从总线 OFF 转移到停止状态（自动或者手动）。</p> <p>在此模块解除总线 OFF 后，此位保持被置位的状态，因此需要通过软件进行清除。必须通过软件读 GSR0，判断此模块是总线 OFF 状态还是错误主动状态。即使节点为总线 OFF 状态，只要写“1”就清除此位。忽视写“0”。</p> <p>0: [清除条件] 写“1” 1: 因发送错误而转移到总线 OFF 状态或者从总线 OFF 状态返回到错误主动状态 [置位条件] 在 $TEC \geq 256$ 或者连续 128 次接收到 11 个隐性位后结束总线 OFF 或者从总线 OFF 转移到停止状态。</p>
5	IRR5	0	R/W	<p>错误被动中断标志</p> <p>表示通过接收/发送错误计数器进入错误被动状态或者通过测试模式强制进入错误被动状态。如果给此位写“1”，就清除此位。忽视写“0”。即使清除此位，节点也可能保持错误被动状态。需要通过软件检查 GSR0 和 GSR5，判断此模块是错误被动状态还是总线 OFF 状态。</p> <p>0: [清除条件] 写“1” 1: 因发送/接收错误而进入错误被动状态 [置位条件] $TEC \geq 128$ 或者 $REC \geq 128$ 或者使用错误被动测试模式。</p>

位	位名	初始值	R/W	说明
4	IRR4	0	R/W	<p>接收错误计数器警告中断标志</p> <p>如果在此模块不为总线 OFF 状态时接收错误计数器 (REC) 超过 95, 就将此位置位。如果给此位写 “1”, 就清除中断。忽视写 “0”。</p> <p>0: [清除条件] 写 “1”</p> <p>1: 因接收错误而进入错误警告状态</p> <p>[置位条件] 在此模块不为总线 OFF 状态的情况下 $REC \geq 96$ 时</p>
3	IRR3	0	R/W	<p>发送错误计数器警告中断标志</p> <p>如果在此模块不为总线 OFF 状态时发送错误计数器 (TEC) 超过 95, 就将此位置位。如果给此位写 “1”, 就清除中断。忽视写 “0”。</p> <p>0: [清除条件] 写 “1”</p> <p>1: 因发送错误而进入错误警告状态</p> <p>[置位条件] $TEC \geq 96$</p>
2	IRR2	0	R	<p>远程帧接收中断标志</p> <p>表示信箱接收到远程帧。当至少有 1 个没被置位的对应 MBIMR 的信箱保存了远程帧发送请求时, 将此位置位。如果清除远程帧接收结束寄存器 (RFPR) 的全部位时, 就清除此位。另外, 当给 MBIMR 对应的全部位写 “1” 时, 就清除此位。此位的写操作无效。</p> <p>0: [清除条件] 清除 RFPR 的全部位</p> <p>1: 至少有 1 个远程帧请求为处理等待状态</p> <p>[置位条件] 在对应的 MBIMR 为 “0” 时接收远程帧。</p>
1	IRR1	0	R	<p>数据帧接收中断标志</p> <p>表示有处理等待状态的接收数据帧。当此位为 “1” 时, 至少有 1 个信箱保存了未处理的信息。如果清除数据帧接收结束寄存器 (RXPR) 的全部位 (哪个接收信箱都没有未处理的信息), 就清除此位。此位是 MBIMR 不为 “0” 的各接收信箱的 RXPR 标志的逻辑与。</p> <p>另外, 如果给 MBIMR 对应的全部位写 “1”, 就清除此位。此位的写操作无效。</p> <p>0: [清除条件] 清除 RXPR 的全部位</p> <p>1: 接收数据帧并且保存到信箱</p> <p>[置位条件] 在对应的 MBIMR 为 “0” 时接收数据。</p>

位	位名	初始值	R/W	说明
0	IRR0	1	R/W	<p>复位 / 停机 / 睡眠中断标志</p> <p>通过以下 3 种情况将此位置位：</p> <ul style="list-style-type: none"> 在软件复位（MCR0）或者硬件复位后转移到复位模式。 在请求停机（MCR1）后转移到停机模式。 在停机模式中发行睡眠请求（MCR5）后转移到 CAN 睡眠模式。 <p>在将此位进行置位后读 GSR，能得知此模块的状态。</p> <p>【注】 需要发行睡眠模式请求时，必须事先使用停机模式。请参照 MCR5 的说明和图 23.17。</p> <p>在 GSR3 或者 GSR4 从“0”变为“1”时或者在从停机模式转移到 CAN 睡眠模式时，将 IRR0 置位。因此，如果此模块在解除停机模式后立即再次转移到停机模式，就不清除 GSR4，所以不将 IRR0 置位。同样，即使从 CAN 睡眠模式直接转移到停机模式，也将 IRR0 置位。当从停机模式 /CAN 睡眠模式转移到发送 / 接收运行时，就在清除 GSR4 前需要 [1 位时间 - TSEG2] ~ [1 位时间 × 2 - TSEG2] 的时间。</p> <p>在复位模式中，虽然将 IRR0 置位，但是因初始化而将 IMR0 自动置位，所以不允许向 CPU 请求中断。</p> <p>0: [清除条件] 写“1”。</p> <p>1: 转移到软件复位模式、停机模式或者 CAN 睡眠模式</p> <p>[置位条件] 在请求复位（MCR0 或硬件）、停机模式（MCR1）或者 CAN 睡眠模式（MCR5）后转移到复位 / 停机 /CAN 睡眠模式。</p>

23.4.5 中断屏蔽寄存器 (IMR)

IMR 是 16 位可读写寄存器，屏蔽中断请求寄存器 (IRR) 各位对应的中断 IRQ 输出信号的生成。如果将位置“1”，就屏蔽对应的中断请求。IMR 直接控制 IRQ 的生成，但是不禁止 IRR 对应位的位置。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IMR15	IMR14	IMR13	IMR12	IMR11	IMR10	IMR9	IMR8	IMR7	IMR6	IMR5	IMR4	IMR3	IMR2	IMR1	IMR0
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 0	IMR[15:0]	H'FFFF	R/W	IRR[15:0] 各位对应的中断源屏蔽 如果将此位置位，即使 IRR 的对应位被置位，也不生成此中断信号。 0: 不屏蔽对应的 IRR (如果发生中断源，就生成 IRQ) 1: 屏蔽 IRR 的对应中断

23.4.6 发送错误计数器 (TEC) / 接收错误计数器 (REC)

TEC 和 REC 都是 16 位可读可条件写寄存器，是表示 CAN 接口的发送 / 接收信息错误数的计数器。计数值由上述参考文献 [1]、[2]、[3]、[4] 规定。除了写错误计数器测试模式以外，此寄存器为只读寄存器，只能通过 CAN 接口进行改写。通过复位请求 (MCR0) 或者转移到总线 OFF 状态来清除此寄存器。

在写错误计数器测试模式 (TST[2:0] 为“B'100”) 中，能写此寄存器。只能给 TEC 和 REC 写相同的值，将 TEC 的写入值设定到 TEC 和 REC。在写此寄存器时，需要将此模块设定为停机模式。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TEC7	TEC6	TEC5	TEC4	TEC3	TEC2	TEC1	TEC0	REC7	REC6	REC5	REC4	REC3	REC2	REC1	REC0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

【注】* 只在MCR的TST[2:0]位为“B'100”的测试模式中才能写此位。在总线OFF的状态下，REC对总线OFF恢复顺序所需的11个隐性位的接收次数进行计数。

位	位名	初始值	R/W	说明
15 ~ 8	TEC[7:0]	H'00	R/W*	发送错误计数器 如果在发送时检测到基于 CAN 规格的错误，此寄存器的值就递增。
7 ~ 0	REC[7:0]	H'00	R/W*	接收错误计数器 如果在接收时检测到基于 CAN 规格的错误，此寄存器的值就递增。

23.5 信箱寄存器

说明此模块的信箱寄存器。信箱寄存器控制各信箱并且表示信箱的状态。

信箱寄存器如表 23.7 所示。

【重要】 能作为 2 个连续的字存取进行长字存取。

表 23.7 信箱寄存器

寄存器名	略称	地址	存取长度 (位)
发送等待寄存器 1	TXPR1	H'020	32
发送等待寄存器 0	TXPR0	H'022	—
		H'024	
		H'026	
发送取消寄存器 1	TXCR1	H'028	16/32
发送取消寄存器 0	TXCR0	H'02A	16
		H'02C	
		H'02E	
发送应答寄存器 1	TXACK1	H'030	16/32
发送应答寄存器 0	TXACK0	H'032	16
		H'034	
		H'036	
中止应答寄存器 1	ABACK1	H'038	16/32
中止应答寄存器 0	ABACK0	H'03A	16
		H'03C	
		H'03E	
数据帧接收结束寄存器 1	RXPR1	H'040	16/32
数据帧接收结束寄存器 0	RXPR0	H'042	16
		H'044	
		H'046	
远程帧接收结束寄存器 1	RFPR1	H'048	16/32
远程帧接收结束寄存器 0	RFPR0	H'04A	16
		H'04C	
		H'04E	
信箱中断屏蔽寄存器 1	MBIMR1	H'050	16/32
信箱中断屏蔽寄存器 0	MBIMR0	H'052	16
		H'054	
		H'056	
未读信息状态寄存器 1	UMSR1	H'058	16/32
未读信息状态寄存器 0	UMSR0	H'05A	16
		H'05C	
		H'05E	

23.5.1 发送等待寄存器 1、0 (TXPR1、TXPR0)

TXPR1 和 TXPR0 连接，构成保存 CAN 模块发送等待标志的 32 位寄存器。在 16 位总线接口时，能作为 2 个连续的字存取来进行长字存取。

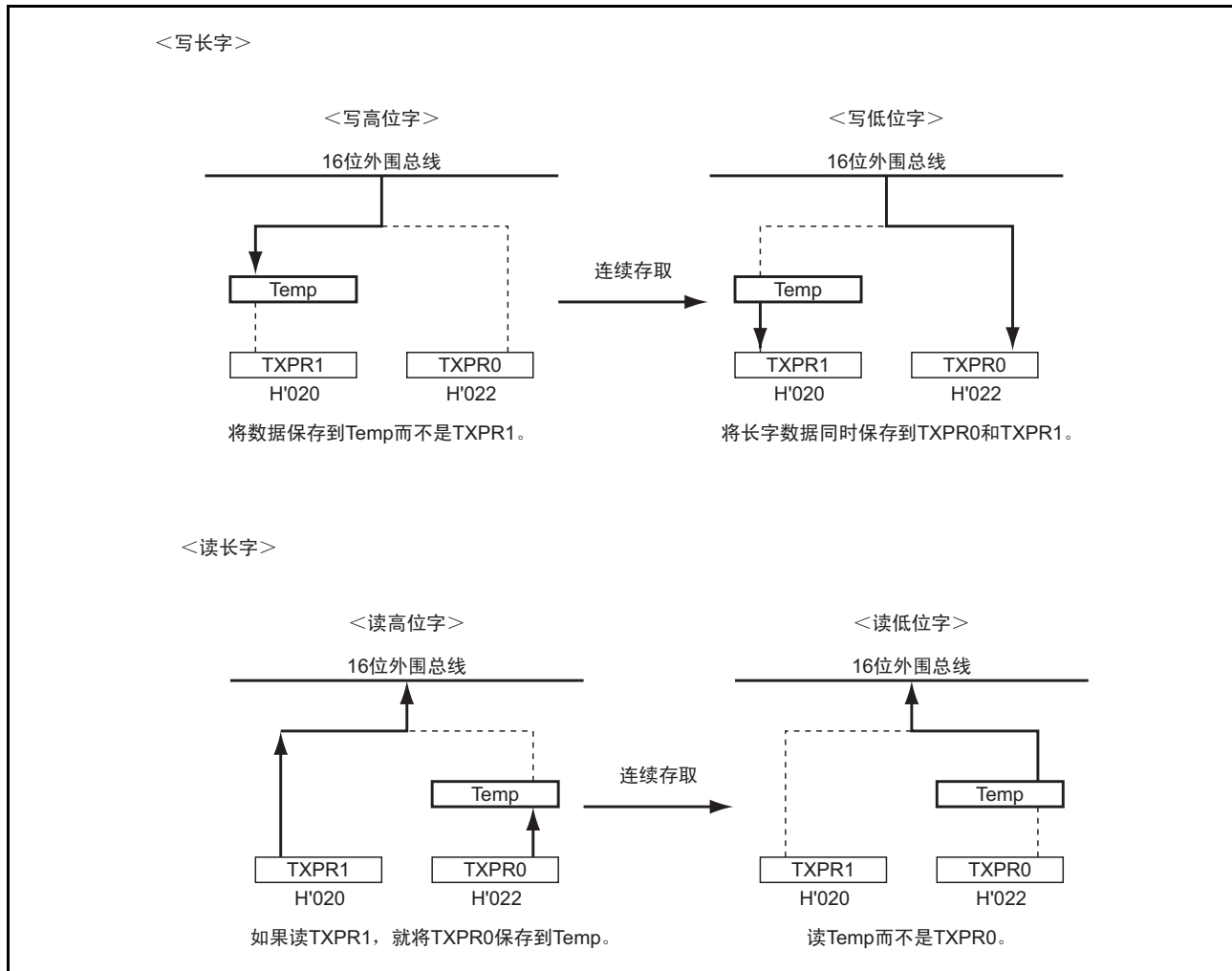


图 23.13 16 位总线接口时的长字存取

TXPR1 寄存器控制信箱 31 ~ 16, TXPR0 寄存器控制信箱 15 ~ 1。通过给位写“1”, CPU 能操作对应的发送信息。写“0”无效。要清除 TXPR 时, 必须将 TXCR 的对应位置位。通过读 TXPR, CPU 能得知发送等待的信箱以及正在发送的信箱。实际上, 除了信箱 0 以外的全部信箱都有发送等待位。另外, 禁止给未设定为发送信箱的对应位写“1”。

在事件触发模式中, 此模块在正常发送信息后或者在中止 TXCR 的发送后, 清除对应的发送等待标志。在时间触发模式中, 为了连续进行每个程控周期计数 (Basic Cycle) 的发送, 在发送结束后不清除信箱 30 ~ 24 的 TXPR。当因 CAN 节点仲裁失败或者 CAN 总线发生错误而不发送信息时, 不清除 TXPR 的标志。此时, 如果对应信箱的信息控制字段的 DART (自动重新发送无效) 位没有被置位, 此模块就自动尝试重新发送; 如果 DART 被置位, 就清除发送, 并且通过信箱空中断标志 (IRR8) 和中止应答寄存器 (ABACK) 的对应位通知 CPU。

当 TXPR 的状态发生变化时, 即使总线仲裁失败或者 CAN 总线发生错误, 也总是发送 ID 所示最高优先级 (MCR2 为“0”) 的信息。详细内容请参照“23.7 运行说明”。

当此模块将 TXPR 位的状态更改为“0”时, 就可能生成信箱空中断标志 (IRR8)。这表示信箱的发送正常结束或者中止。如果信息发送正常结束, 就反映在 TXACK; 如果信息发送中止, 就反映在 ABACK。通过检查这些寄存器, 能将对应信箱的信息数据字段的内容改写为在下次发送时使用。

(1) TXPR1

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXPR1[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

【注】* 只能给设定为发送信箱的对应位写“1”。

位	位名	初始值	R/W	说明
15 ~ 0	TXPR1[15:0]	H'0000	R/W	向对应的信箱请求发送 CAN 帧。bit15 ~ 0 对应信箱 31 ~ 16。根据 MCR2 位的设定，多位被置位时的发送顺序为信息 ID 优先级的顺序或者信箱号顺序。 0: 对应的信箱为发送信息空闲状态 [清除条件] 结束信息发送（事件触发信息时）或者中止信息发送（自动进行清除） 1: 向对应的信箱请求发送

(2) TXPR0

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXPR0[15:1]															—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R

【注】* 只能给设定为发送信箱的对应位写“1”。
必须通过长字存取来读写TXPR1/TXPR0。

位	位名	初始值	R/W	说明
15 ~ 1	TXPR0[15:1]	H'0000	R/W	表示向对应的信箱请求发送 CAN 帧。bit15 ~ 1 对应信箱 15 ~ 1。根据 MCR2 位的设定，多位被置位时的发送顺序为信息 ID 优先级的顺序或者信箱号顺序。 0: 对应的信箱为发送信息空闲状态 [清除条件] 结束信息发送（事件触发信息时）或者中止信息发送（自动进行清除）。 1: 向对应的信箱请求发送
0	—	0	R	保留位 因为是接收专用信箱，所以此位总是“0”。忽视写“1”，读取值为“0”。

23.5.2 发送取消寄存器 1、0 (TXCR1、TXCR0)

TXCR1 和 TXCR0 是 16 位可读可条件写寄存器，TXCR1 控制信箱 31 ~ 16，TXCR0 控制信箱 15 ~ 1。CPU 使用 TXCR 取消 TXPR 的发送请求。要清除 TXPR 的位时，必须从 CPU 给对应的 TXCR 的位写“1”。忽视写“0”。

如果正常中止，CAN 控制器就清除对应的 TXPR 和 TXCR 的位并且将 ABACK 位置位。但是，在信箱开始发送后，不能通过此寄存器的位来取消。此时，如果发送正常结束，CAN 控制器就清除 TXPR 和 TXCR 的位并且将 TXACK 的位置位。但是，如果因仲裁失败或者总线发生错误而不能发送，CAN 控制器就清除对应的 TXPR 和 TXCR 的位并且将 ABACK 位置位。即使 CPU 要清除不是发送等待状态的信箱发送，也被忽视。此时，CPU 不能将 TXCR 的标志置位。

(1) TXCR1

位: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

TXCR1[15:0]

初始值: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

R/W: R/W* R/W* R/W* R/W* R/W* R/W* R/W* R/W* R/W* R/W* R/W* R/W* R/W* R/W* R/W*

【注】* 只能给接受到发送请求的发送信箱的对应位写“1”。

位	位名	初始值	R/W	说明
15 ~ 0	TXCR1[15:0]	H'0000	R/W	向进入发送队列的信箱请求取消发送。bit15 ~ 0 对应信箱 31 ~ 16 (以及 TXPR1[15:0])。 0: 对应的信箱为发送信息取消空闲状态 [清除条件] 结束发送信息的取消 (自动进行清除)。 1: 向对应的信箱请求发送取消

(2) TXCR0

位: 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

TXCR0[15:1] —

初始值: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

R/W: R/W* R/W* R/W* R/W* R/W* R/W* R/W* R/W* R/W* R/W* R/W* R/W* R/W* R/W* R/W* R

【注】* 只能给接受到发送请求的发送信箱的对应位写“1”。

位	位名	初始值	R/W	说明
15 ~ 1	TXCR0[15:1]	H'0000	R/W	向进入发送队列的信箱请求取消发送。bit15 ~ 1 对应信箱 15 ~ 1 (以及 TXPR0[15:1])。 0: 对应的信箱为发送信息取消空闲状态 [清除条件] 结束发送信息的取消 (自动进行清除)。 1: 向对应的信箱请求发送取消
0	—	0	R	保留位 因为是接收专用信箱，所以此位总是“0”。忽视写“1”，读取值为“0”。

23.5.3 发送应答寄存器 1、0 (TXACK1、TXACK0)

TXACK1 和 TXACK0 是 16 位可读可条件写寄存器，用于将信箱发送正常结束通知 CPU。如果发送正常进行，此模块就将 TXACK 寄存器的对应位置位。CPU 能通过写“1”清除 TXACK 的位。忽视写“0”。

(1) TXACK1

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXACK1[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

【注】* 为了清除位，只能写“1”。

位	位名	初始值	R/W	说明
15 ~ 0	TXACK1[15:0]	H'0000	R/W	通知对应的信箱所请求的发送已正常进行。bit15 ~ 0 对应信箱 31 ~ 16。 0: [清除条件] 写“1” 1: 正常发送对应信箱的信息（数据帧或者远程帧） [置位条件] 结束对应信箱的信息发送。

(2) TXACK0

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXACK0[15:1]															—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	—

【注】* 为了清除位，只能写“1”。

位	位名	初始值	R/W	说明
15 ~ 1	TXACK0[15:1]	H'0000	R/W	通知对应的信箱所请求的发送已正常进行。bit15 ~ 1 对应信箱 15 ~ 1。 0: [清除条件] 写“1” 1: 正常发送对应信箱的信息（数据帧或者远程帧） [置位条件] 结束对应信箱的信息发送。
0	—	0	R	保留位 因为是接收专用信箱，所以此位总是“0”。忽视写“1”，读取值为“0”。

23.5.4 中止应答寄存器 1、0 (ABACK1、ABACK0)

ABACK1 和 ABACK0 是 16 位可读可条件写寄存器，用于将已按请求中止信箱的发送通知 CPU。如果进行中止，此模块就将 ABACK 寄存器的对应位置位。CPU 能通过写“1”清除 ABACK 的位。忽视写“0”。通过此模块将 ABACK 的位置位，确认通过对应的 TXCR 位清除了 TXPR 位。

(1) ABACK1

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ABACK1[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

【注】* 为了清除位，只能写“1”。

位	位名	初始值	R/W	说明
15 ~ 0	ABACK1[15:0]	H'0000	R/W	通知对应的信箱所请求的发送取消已正常进行。bit15 ~ 0 对应于信箱 31 ~ 16。 0: [清除条件] 写“1” 1: 取消对应信箱的信息（数据帧或者远程帧）发送 [置位条件] 取消对应信箱的信息发送。

(2) ABACK0

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ABACK0[15:1]															—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R

【注】* 为了清除位，只能写“1”。

位	位名	初始值	R/W	说明
15 ~ 1	ABACK0[15:1]	H'0000	R/W	通知对应的信箱所请求的发送取消已正常进行。bit15 ~ 1 对应于信箱 15 ~ 1。 0: [清除条件] 写“1” 1: 取消对应信箱的信息（数据帧或者远程帧）发送 [置位条件] 取消对应信箱的信息发送。
0	—	0	R	保留位 因为是接收专用信箱，所以此位总是“0”。忽视写“1”，读取值为“0”。

23.5.5 数据帧接收结束寄存器 1、0 (RXPR1、RXPR0)

RXPR1 和 RXPR0 是 16 位可读可条件写寄存器，由表示接收信箱接收到数据帧的标志构成。如果将 CAN 数据帧正常地保存到接收信箱，就将 RXPR 的对应位置位。如果写“1”，就清除对应的位，写“0”无效。但是，只在通过 MBC（信箱配置）将信箱设定为接收数据帧时才将位置位。当 RXPR 的位被置位时，如果没有将对应的 MBIMR 置位，就将 IRR1（数据帧接收中断标志）置位，而且如果 IMR1 没有被置位，就生成中断信号。此寄存器的位只在接收数据帧时被置位，而在接收远程帧时不被置位。

(1) RXPR1

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RXPR1[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

【注】* 为了清除位，只能写“1”。

位	位名	初始值	R/W	说明
15 ~ 0	RXPR1[15:0]	H'0000	R/W	bit15 ~ 0 对应接收信箱号 31 ~ 16 的信箱。 0: [清除条件] 写“1” 1: 对应的信箱接收到 CAN 数据帧 [置位条件] 结束对应信箱的数据帧接收。

(2) RXPR0

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RXPR0[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

【注】* 为了清除位，只能写“1”。

位	位名	初始值	R/W	说明
15 ~ 0	RXPR0[15:0]	H'0000	R/W	bit15 ~ 0 对应接收信箱号 15 ~ 0 的信箱。 0: [清除条件] 写“1” 1: 对应的信箱接收到 CAN 数据帧 [置位条件] 结束对应信箱的数据帧接收。

23.5.6 远程帧接收结束寄存器 1、0 (RFPR1、RFPR0)

RFPR1 和 RFPR0 是 16 位可读可条件写寄存器，由表示接收信箱接收到远程帧的标志构成。如果将 CAN 远程帧正常地保存到接收信箱，就将 RFPR 的对应位置位。如果写“1”，就清除对应的位，写“0”无效。全部信箱都有位，但是只在通过 MBC（信箱配置）将信箱设定为接收远程帧时才将位置位。当 RFPR 的位被置位时，如果没有将对应的 MBIMR 置位，就将 IRR2（数据帧接收中断标志）置位，而且如果 IMR2 没有被置位，就生成中断信号。此寄存器的位只在接收远程帧时被置位，而在接收数据帧时不被置位。

(1) RFPR1

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFPR1[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

【注】* 为了清除位，只能写“1”。

位	位名	初始值	R/W	说明
15 ~ 0	RFPR1[15:0]	H'0000	R/W	这些位是信箱 31 ~ 16 的远程请求接收标志。 0: [清除条件] 写“1” 1: 对应信箱接收到 CAN 远程帧 [置位条件] 结束对应信箱的远程帧接收。

(2) RFPR0

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFPR0[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

【注】* 为了清除位，只能写“1”。

位	位名	初始值	R/W	说明
15 ~ 0	RFPR0[15:0]	H'0000	R/W	这些位是信箱 15 ~ 0 的远程请求接收标志。 0: [清除条件] 写“1” 1: 对应信箱接收到 CAN 远程帧 [置位条件] 结束对应信箱的远程帧接收。

23.5.7 信箱中断屏蔽寄存器 1、0 (MBIMR1、MBIMR0)

MBIMR1 和 MBIMR0 是 16 位可读写寄存器。MBIMR 是屏蔽信箱运行的相关 IRR (IRR1: 数据帧接收中断、IRR2: 远程帧接收中断、IRR8: 信箱空中断、IRR9: 信息超限 / 重写中断) 的寄存器。当设定为接收信箱时, 屏蔽由接收中断标志 (IRR1、IRR2、IRR9) 产生的中断。不影响 RXPR、RFPR、UMSR 位的置位。当设定为发送信箱时, 屏蔽由发送或者中止发送 (IRR8) 产生的中断或者信箱空中断。不影响由发送产生的 TXPR/TXCR 位的清除和 TXACK 位的置位以及由中止发送产生的 TXPR/TXCR 位的清除和 ABACK 位的置位。

通过给要屏蔽的信箱所对应的位写“1”来设定屏蔽。在复位时, 屏蔽全部信箱的中断。

(1) MBIMR1

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MBIMR1[15:0]															
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 0	MBIMR1[15:0]	H'FFFF	R/W	允许或者禁止信箱 31 ~ 16 的中断请求。 0: 允许 IRR1、IRR2、IRR8、IRR9 的中断请求 1: 禁止 IRR1、IRR2、IRR8、IRR9 的中断请求

(2) MBIMR0

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MBIMR0[15:0]															
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 0	MBIMR0[15:0]	H'FFFF	R/W	允许或者禁止信箱 15 ~ 0 的中断请求。 0: 允许 IRR1、IRR2、IRR8、IRR9 的中断请求 1: 禁止 IRR1、IRR2、IRR8、IRR9 的中断请求

23.5.8 未读信息状态寄存器 1、0 (UMSR1、UMSR0)

UMSR1 和 UMSR0 是 16 位可读可条件写寄存器，在接收新信息时记录没有通过 CPU/ 直接存储器存取控制器存取内容的接收信箱。在接收信箱的新信息时，如果没有通过 CPU 清除 RXPR 或者 RFPR 的对应位，就将 UMSR 的位置位。如果写“1”，就进行清除，写“0”无效。

当设定为发送信箱时，不将对应的 UMSR 的位置位。

(1) UMSR1

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UMSR1[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

【注】* 为了清除位，只能写“1”。

位	位名	初始值	R/W	说明
15 ~ 0	UMSR1[15:0]	H'0000	R/W	表示信箱 31 ~ 16 的未读接收信息发生重写或者超限。 0: [清除条件] 写“1” 1: 未读接收信息被新信息重写或者发生超限 [置位条件] 在清除 RXPR 或者 RFPR 前接收到新的信息。

(2) UMSR0

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UMSR0[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

【注】* 为了清除位，只能写“1”。

位	位名	初始值	R/W	说明
15 ~ 0	UMSR0[15:0]	H'0000	R/W	表示信箱 15 ~ 0 的未读接收信息发生重写或者超限。 0: [清除条件] 写“1” 1: 未读接收信息被新信息重写或者发生超限 [置位条件] 在清除 RXPR 或者 RFPR 前接收到新的信息。

23.6 定时器的寄存器

此模块的定时器是 16 位定时器，支持多种时钟源。能通过预分频计数器降低时钟的速度。另外，还有 3 个比较匹配寄存器（TCMR2、TCMR1、TCMR0）。地址映像如下所示：

【重要】 对定时器的寄存器只能进行字（16 位）存取。

表 23.8 定时器的寄存器结构

寄存器名	略称	地址	存取长度（位）
时间触发控制寄存器 0	TTCR0	H'080	16
最大周期 /Tx_Enable_Window 寄存器	CMAX_TEW	H'084	16
基准触发偏移寄存器	RFTROFF	H'086	16
定时器的状态寄存器	TSR	H'088	16
周期计数寄存器	CCR	H'08A	16
定时器的计数寄存器	TCNTR	H'08C	16
周期时间寄存器	CYCTR	H'090	16
基准标记寄存器	RFMK	H'094	16
定时器的比较匹配寄存器 0	TCMR0	H'098	16
定时器的比较匹配寄存器 1	TCMR1	H'09C	16
定时器的比较匹配寄存器 2	TCMR2	H'0A0	16
发送触发时间选择寄存器	TTTSEL	H'0A4	16

23.6.1 时间触发控制寄存器 0（TTCR0）

TTCR0 是 16 位可读写寄存器。

控制定时器的运行。必须在设定此寄存器并且运行定时器后进行固定周期的发送和监视寄存器的设定。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCR15	TCR14	TCR13	TCR12	TCR11	TCR10	—	—	—	TCR6	TPSC5	TPSC4	TPSC3	TPSC2	TPSC1	TPSC0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	TCR15	0	R/W	定时器允许 如果将此位置位，就运行 TCNTR。如果清除此位，就清除 TCNTR 和 CCR。 0: 清除 TCNTR 和 CCR 并且停止运行 1: 定时器运行
14	TCR14	0	R/W	时戳值 指定是将周期时间（CYCTR）还是将 CCR[5:0]+CYCTR[15:6] 作为信箱 15 ~ 0 的发送和接收的时戳。此功能便于在时间触发模式中监视 Rx_Trigger。 此寄存器不影响信箱 30 和信箱 31 的时戳。 0: 使用 CYCTR[15:0] 作为信箱 15 ~ 0 的时戳 1: 使用 CCR[5:0]+CYCTR[15:6] 作为信箱 15 ~ 0 的时戳

位	位名	初始值	R/W	说明
13	TCR13	0	R/W	通过 TCMR2 进行的取消 在将此位和 bit12 置位的状态下, 如果此模块在停机模式以外的模式中发生比较匹配, 就自动将该 TXPR 位对应的全部 TXCR 位置位, 从而取消发送队列中的信息。 0: 禁止通过 TCMR2 比较匹配取消发送 1: 允许通过 TCMR2 比较匹配取消发送
12	TCR12	0	R/W	比较匹配允许 如果将此位置位, 就通过 TCMR2 的比较匹配将 IRR11 置位。 0: 不通过 TCMR2 的比较匹配将 IRR11 置位 1: 通过 TCMR2 的比较匹配将 IRR11 置位
11	TCR11	0	R/W	比较匹配允许 如果将此位置位, 就通过 TCMR1 的比较匹配将 IRR15 置位。 0: 不通过 TCMR1 的比较匹配将 IRR15 置位 1: 通过 TCMR1 的比较匹配将 IRR15 置位
10	TCR10	0	R/W	比较匹配允许 如果将此位置位, 就通过 TCMR0 的比较匹配将 IRR14 置位。 0: 不通过 TCMR0 的比较匹配将 IRR14 置位 1: 通过 TCMR0 的比较匹配将 IRR14 置位
9 ~ 7	—	全 0	R	保留位 读写值都为“0”。
6	TCR6	0	R/W	通过 TCMR0 进行的定时器清除控制 此位指定在 TCNTR 和 TCMR0 相同时是否将定时器清除为“H'0000”。另外, TCMR0 也能通过 IRR14 给 CPU 产生中断信号。 【注】 此模块在 TTCAN 模式中运行时 (CMAX 不为“B'111”), 为了不清除本地时间, 必须将此位置“0”。 0: 不通过 TCMR0 清除定时器 1: 通过 TCMR0 清除定时器
5 ~ 0	TPSC5 ~ TPSC0	H'00	R/W	定时器的预分频器 能通过设定这些位来分频定时器的源时钟 (4×[此模块的系统时钟])。预分频器只在事件触发模式中有效。在 TTCAN 模式中 (CMAX 不为“B'111”), 自动选择 1 位定时 (=CAN 总线上的 1 位长度) 作为 TCNTR 的源时钟。 源时钟周期与定时器周期的关系如下所示: 000000: 1X 源时钟 000001: 2X 源时钟 000010: 3X 源时钟 000011: 4X 源时钟 000100: 5X 源时钟 : 111111: 64X 源时钟

23.6.2 最大周期 /Tx_Enable_Window 寄存器 (CMAX_TEW)

CMAX_TEW 是 16 位可读写寄存器。

CMAX 指定用于时间触发发送的周期计数器 (CCR) 的最大值，从而设定系统矩阵的周期计数值 (Basic Cycle)。如果周期计数器为最大值 (CCR=CMAX)，就在进行 1 个周期计数后将周期计数器清“0”并且产生 IRR10 中断。

TEW 指定 Tx_Enable_Window 的宽度。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CMAX[2:0]			—	—	—	—	TEW[3:0]			
初始值:	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 8	CMAX[2:0]	111	R/W	周期计数器最大值 表示 CCR 的最大值。 周期计数器的最大值 = $2^{CMAX} - 1$ (CMAX = 寄存器值) 在用于时间触发发送的矩阵周期中能进行周期计数的值为 (周期计数器的最大值 + 1)。当不将 CMAX 设定为“B'111”时，此模块为时间触发模式，能使用时间触发功能。 当将 CMAX 设定为“B'111”时，此模块为事件触发模式。 000: 周期计数器的最大值 = 0 001: 周期计数器的最大值 = 1 010: 周期计数器的最大值 = 3 011: 周期计数器的最大值 = 7 100: 周期计数器的最大值 = 15 101: 周期计数器的最大值 = 31 110: 周期计数器的最大值 = 63 111: 此模块为事件触发模式 (清除 CCR) 【注】 在使用事件触发模式时，必须将 CMAX 设定为“B'111”。
7 ~ 4	—	全 0	R	保留位 读写值都为“0”。
3 ~ 0	TEW[3:0]	0000	R/W	Tx_Enable_Window 设定 Tx_Enable_Window 的宽度。 当 TEW 为“B'0000”时，窗口宽度为 1 位定时。 能设定 B'0000 ~ B'1111 的全部值。 0000: Tx_Enable_Window 的宽度 = 1 0001: Tx_Enable_Window 的宽度 = 2 0010: Tx_Enable_Window 的宽度 = 3 0011: Tx_Enable_Window 的宽度 = 4 : 1111: Tx_Enable_Window 的宽度 = 16 【注】 因为此模块的 CAN 数据链路控制器从发送请求开始发送，所以总是需要 CAN 总线 1 ~ 2 位定时。因此，上述的值未考虑此精度。

23.6.3 基准触发偏移寄存器 (RFTROFF)

RFTROFF 是 8 位可读写寄存器。

信箱 30 的发送触发时间 (TTT) 能设定的偏移为 $-127 \sim +127$ 。在附加 RFTROFF 寄存器值的符号扩展值后, 将信箱 30 的 TTT 和 CYCTR 进行比较。

但是, 不更改 TTT 的值。此偏移值只影响信箱 30。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFTROFF[7:0]								—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 8	RFTROFF [7:0]	H'00	R/W	这些位表示基准触发的偏移值。 00000000: ref_trigger_offset=+0 00000001: ref_trigger_offset=+1 00000010: ref_trigger_offset=+2 : 01111111: ref_trigger_offset=+127 : 11111111: ref_trigger_offset=-1 11111110: ref_trigger_offset=-2 : 10000001: ref_trigger_offset=-127
7 ~ 0	—	全 0	R	保留位 读写值都为“0”。

23.6.4 定时器的状态寄存器 (TSR)

TSR 是 16 位只读寄存器，从而 CPU 能监视定时器比较匹配的状态和时间超限状态。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	TSR4	TSR3	TSR2	TSR1	TSR0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R*	R*	R*	R*	R*

位	位名	初始值	R/W	说明
15 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	TSR4	0	R*	系统矩阵开始 表示开始下一个系统矩阵。 当 CCR 为“0”时，在结束时间基准信息的发送 / 接收时将此位置位。 0: 不是下一个系统矩阵的开头 [清除条件] 给 IRR10 (周期计数器上溢中断标志) 写“1”。 1: 周期计数器为“0” [置位条件] 周期计数器的值从最大值 (CMAX) 变为“H'0”。 在 CMAX 不为“B'111”而 CCR 为“0”时结束时间基准信息的发送 / 接收。
3	TSR3	0	R*	定时器比较匹配标志 2 表示定时器的比较匹配寄存器 2 (TCMR2) 发生比较匹配。在 TCMR2 的设定值和周期时间寄存器相同 (TCMR2=CYCTR) 时，如果 TTCR0 的 bit12 被置“1”，就将此位置位。此位是只读位，如果清除 IRR11 (定时器比较匹配中断 2 标志)，就清除此位。 0: TCMR2 定时器没有发生比较匹配 [清除条件] 给 IRR11 (定时器比较匹配中断 2 标志) 写“1”。 1: TCMR2 定时器发生比较匹配 [置位条件] 在 TTCR0 的 bit12 为“1”时 TCMR2 和周期时间相同 (TCMR2=CYCTR)。
2	TSR2	0	R*	定时器比较匹配标志 1 表示定时器的比较匹配寄存器 1 (TCMR1) 发生比较匹配。在 TCMR1 的设定值和周期时间寄存器相同 (TCMR1=CYCTR) 时，如果 TTCR0 的 bit11 被置“1”，就将此位置位。此位是只读位，如果清除 IRR15 (定时器比较匹配中断 1 标志)，就清除此位。 0: TCMR1 定时器没有发生比较匹配 [清除条件] 给 IRR15 (定时器比较匹配中断 1 标志) 写“1”。 1: TCMR1 定时器发生比较匹配 [置位条件] 在 TTCR0 的 bit11 为“1”时 TCMR1 和周期时间相同 (TCMR1=CYCTR)。

位	位名	初始值	R/W	说明
1	TSR1	0	R*	<p>定时器比较匹配标志 0</p> <p>表示定时器的比较匹配寄存器 0 (TCMR0) 发生比较匹配。在 TCM02 的设定值和定时器的值相同 (TCMR0=TCNTR) 时, 如果 TTCR0 的 bit10 被置“1”, 就将此位置位。此位是只读位, 如果清除 IRR14 (定时器比较匹配中断 0 标志), 就清除此位。</p> <p>0: TCMR1 定时器没有发生比较匹配 [清除条件] 给 IRR14 (定时器比较匹配中断 0 标志) 写“1”。</p> <p>1: TCMR0 定时器发生比较匹配 [置位条件] 在 TTCR0 的 bit10 为“1”时 TCMR0 和定时器的值相同 (TCMR0=TCNTR)。</p>
0	TSR0	0	R*	<p>定时器超限 / Next_is_Gap 接收 / 信息错误</p> <p>此标志分为 3 种不同功能, 表示在事件触发模式中运行时发生定时器超限, 在时间触发模式中运行时接收到 Next_is_Gap 被置位的时间基准信息, 在测试模式中检测到 CAN 总线发生错误。</p> <p>测试模式的优先级高于其他设定。</p> <p>0: 在事件触发模式中没有发生定时器 (TCNTR) 超限 在时间触发模式中没有接收到带 Next_is_Gap 的时间基准信息。 在测试模式中没有发生信息错误。</p> <p>[清除条件] 给 IRR13 写“1”。</p> <p>1: [置位条件] 在事件触发模式中定时器 (TCNTR) 发生超限并且从“H'FFFF”变为“H'0000”。 在时间触发模式中接收到带 Next_is_Gap 的时间基准信息。 在测试模式中发生信息错误。</p>

【注】 * 此位是只读位, CPU 能监视周期计数器、定时器和比较匹配寄存器的状态。此位的写操作无效。

23.6.5 周期计数寄存器 (CCR)

CCR 是 6 位可读写寄存器。

这是表示时间触发发送的周期计数值 (Basic Cycle) 的寄存器。

根据此模块是作为潜在时间主控运行还是作为时间从属运行，CCR 值有不同的更新方法。CCR 的更新如下。

此模块作为时间主控 (潜在) 运行的情况：

- 每次周期时间 (CYCTR) 和信箱 30 的发送触发时间相同时加 1。

或者

- 每接收一次有效的基准信息时重写信箱 31 的 MSG_DATA_0[5:0] 的值。

此模块作为时间从属运行的情况：

- 每接收一次有效的基准信息时重写信箱 31 的 MSG_DATA_0[5:0] 的值。

CMAX=B'111 并且 TTCR0[15]=0 的情况：

- CCR 的值总是为 “B'000000”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CCR[5:0]					
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 6	—	全 0	R	保留位 读写值都为 “0”。
5 ~ 0	CCR[5:0]	H'00	R/W	周期计数寄存器 表示用于时间触发发送的矩阵周期的当前周期计数值。

23.6.6 定时器的计数寄存器 (TCNTR)

TCNTR 是 16 位可读写寄存器。

TCNTR 为自由运行定时器计数器。通过将 TTCR0[15] 置“1”，定时器进行计数。

定时器计数器的预分频器因此模块的触发模式而不同。

- 事件触发模式 (C_{MAX}=B'111)：TTCR0[5:0]的TPSC值
- 时间触发模式 (C_{MAX}≠B'111)：CAN总线的1位定时

在 TTCR0[6] 为“1”时，如果 TCNTR 的值和定时器的比较匹配寄存器 0 (TCMR0) 相同，就将 TCNTR 清除为“H'0000”并且进行累加计数。

在时间触发模式中，TCNTR 用作本地时间 (Local time)。因此，必须将 TTCR0[6] 置“0”，作为自由运行定时器运行。

【注】 1. 当 TTCR0[15] 为“0”时，TCNTR 总是为“H'0000”。

2. 从设定允许定时器 (TTCR0[15] 为“1”) 到 TCNTR 开始计数为止，会发生多个时钟周期的延迟，这是由预分频器的内部逻辑引起的。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCNTR[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

【注】 * 只在设定为允许定时器时(TTCR0[15]为“1”)才能写此寄存器。
在时间触发模式(C_{MAX}不为“B'111”)中，禁止写此寄存器。

位	位名	初始值	R/W	说明
15 ~ 0	TCNTR[15:0]	H'0000	R/W	表示自由运行定时器的值。

23.6.7 周期时间寄存器 (CYCTR)

CYCTR 是表示周期时间的 16 位只读寄存器。

Cycle_Time(CYCTR) = Local_Time(TCNTR) - Reference_Mark(RFMK)

在事件触发模式中，RFMK 固定为“0”，因此此寄存器为 TCNTR 的副本。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CYCTR[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 0	CYCTR[15:0]	H'0000	R	表示周期时间。

23.6.8 基准标记寄存器 (RFMK)

RFMK 是 16 位只读寄存器。在正常发送或者接收基准信息时，用于捕捉基准信息 SOF 处的本地时间 (TCNTR)。在事件触发模式中，不使用此寄存器而总是“0”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFMK[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 0	RFMK[15:0]	H'0000	R	基准标记寄存器 表示时间基准信息 SOF 处的 TCNTR 值。

23.6.9 定时器的比较匹配寄存器 0 ~ 2 (TCMR0 ~ 2)

TCMR0 ~ 2 是 16 位可读写寄存器。

能产生中断信号并且清除定时器值 (只支持 TCMR0) 和发送请求 (只支持 TCMR2)。

TCMR0 和 TCNTR 进行比较, TCMR1、TCMR2 和 CYCTR 进行比较。能分别单独设定各比较匹配寄存器的比较值。要将比较匹配的标志置位时, 需要将 TTCR0 的 bit12 ~ 10 置位。在时间触发模式中, TCMR0 用于 Init_Watch_Trigger, TCMR2 用于 Watch_Trigger。

(1) 中断功能

通过设定 TTCR0 的 bit12、bit11 和 bit10 来允许各寄存器的中断标志。

当发生比较匹配时, 将 IRR 对应的中断标志 (bit11、bit15、bit14) 置位。

能通过 IMR 的 bit11、bit15 和 bit14 来禁止产生中断信号。

当发生比较匹配时, 如果 IRR11 (或者 IRR15 或者 IRR14) 被置位, 就将定时器的状态寄存器 (TSR) 的 bit3、bit2 或者 bit1 置位。另外, 如果清除 IRR 的位, 就清除对应的 TSR 的位。

(2) 定时器清除功能

只有 TCMR0 具有通过比较匹配清除定时器值 (TCNTR) 的功能, 在 TTCR0[6] 为“1”时有效。TCMR1 和 TCMR2 没有此功能。

(3) 请求发送的信息的取消功能

当 TTCR0[13:12] 为“B'11”时, 如果在停机模式以外的模式中 TCMR2 发生比较匹配, 就进行 TXPR 设定所对应 TXCR 的设定。因此, 除了正在发送的信息以外, 取消发送等待请求。

TCMR1 和 TCMR0 没有此功能。

• TCMR0

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCMR0[15:0]															
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 0	TCMR0[15:0]	H'FFFF	R/W	定时器的比较匹配寄存器 表示发生比较匹配时的 TCNTR 的值。

- TCMR1

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCMR1[15:0]															
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 0	TCMR1[15:0]	H'FFFF	R/W	定时器的比较匹配寄存器 表示发生比较匹配时的 CYCTR 的值。

- TCMR2

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCMR2[15:0]															
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 0	TCMR2[15:0]	H'FFFF	R/W	定时器的比较匹配寄存器 表示发生比较匹配时的 CYCTR 的值。

23.6.10 发送触发时间选择寄存器 (TTTSEL)

TTTSEL 是 16 位可读写寄存器，指定周期时间和比较匹配的发送触发时间。只能将 1 个位置位。不能将多个位置位并且不能清除全部位。此寄存器只能在配置模式中进行更改。更改时的算法如图 23.14 所示。

TTTSEL 是测试和诊断专用的寄存器，不能在正常运行时写此寄存器，也不保证读取值。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	—	TTTSEL[14:8]								—	—	—	—	—	—	—	—
初始值:	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	

【注】 只能将1位置位。

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 8	TTTSEL [14:8]	B'1000000	R/W	指定 CYCTR 和比较匹配的发送触发时间。bit14 ~ 8 对应信箱 30 ~ 24。
7 ~ 0	—	全 0	R	保留位 读写值都为“0”。

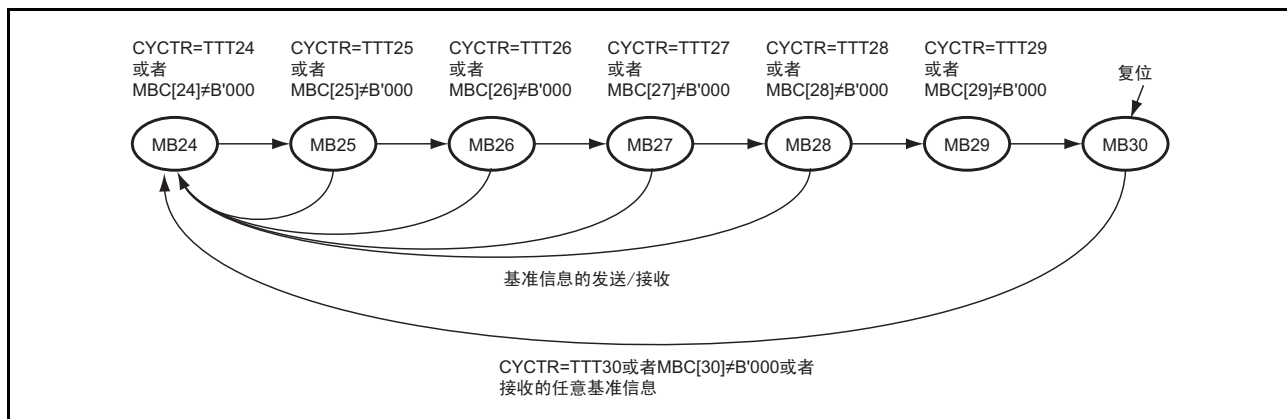


图 23.14 更改 TTTSEL 时的算法

23.7 运行说明

23.7.1 此模块的设定

说明硬件复位（上电复位）或者软件复位（MCR0）后的配置模式和停机模式中的此模块设定。无论是哪种情况，此模块都不能加入 CAN 总线有效。另外，此模块设定的更改不影响 CAN 总线的通信。

(1) 复位顺序

软件复位或者硬件复位后的此模块的设定步骤如图 23.15 所示。在复位后，对全部寄存器进行初始化，因此必须在加入 CAN 总线有效前设定此模块。详细内容请参照图中的注释。

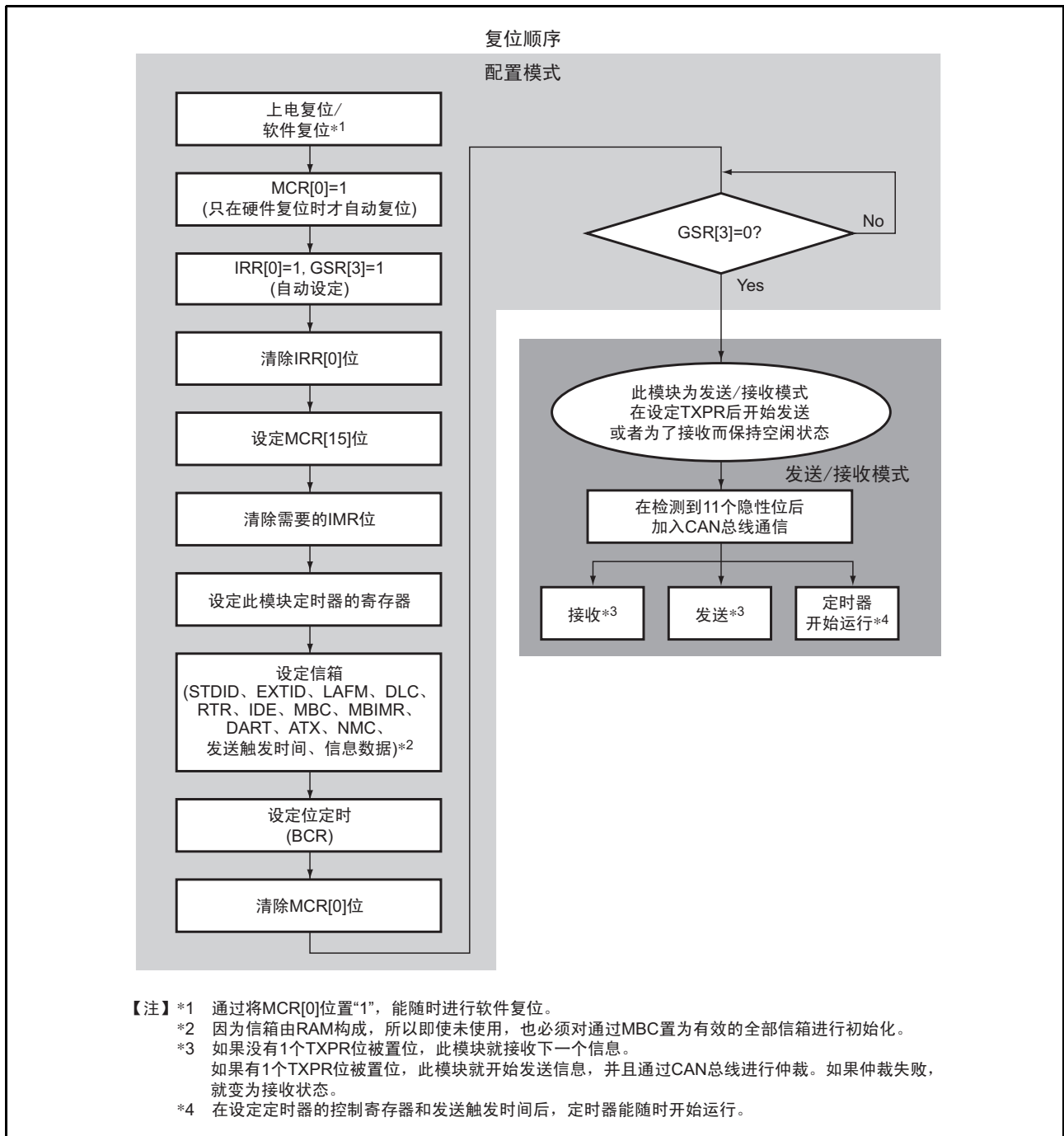


图 23.15 复位顺序

(2) 停机模式

在停机模式中，此模块不能加入 CAN 总线有效，因此用户能在不影响 CAN 总线通信的情况下更改所需的寄存器的设定。在此，重要的是在更改寄存器前等待此模块进入停机模式。不一定要立刻转移到停机模式（在 CAN 总线为空闲状态或者睡眠状态时转移）。如果此模块转移到停机模式，就将 GSR4 位置位。

需要在设定寄存器后解除停机请求。在解除停机请求后（MCR1 为“0”或者 GSR4 为“0”），此模块在检测到 CAN 总线上的 11 个隐性位后加入 CAN 总线有效。

(3) CAN 睡眠模式

在 CAN 睡眠模式中，为了降低消耗电流而停止此模块的主要模块时钟，因此只能存取 MCR、GSR、IRR 和 IMR 寄存器。在 CAN 睡眠模式中，不能清除发送和接收有关的中断，因此必须事先进行清除。

CAN 睡眠模式的流程图如图 23.16 所示。

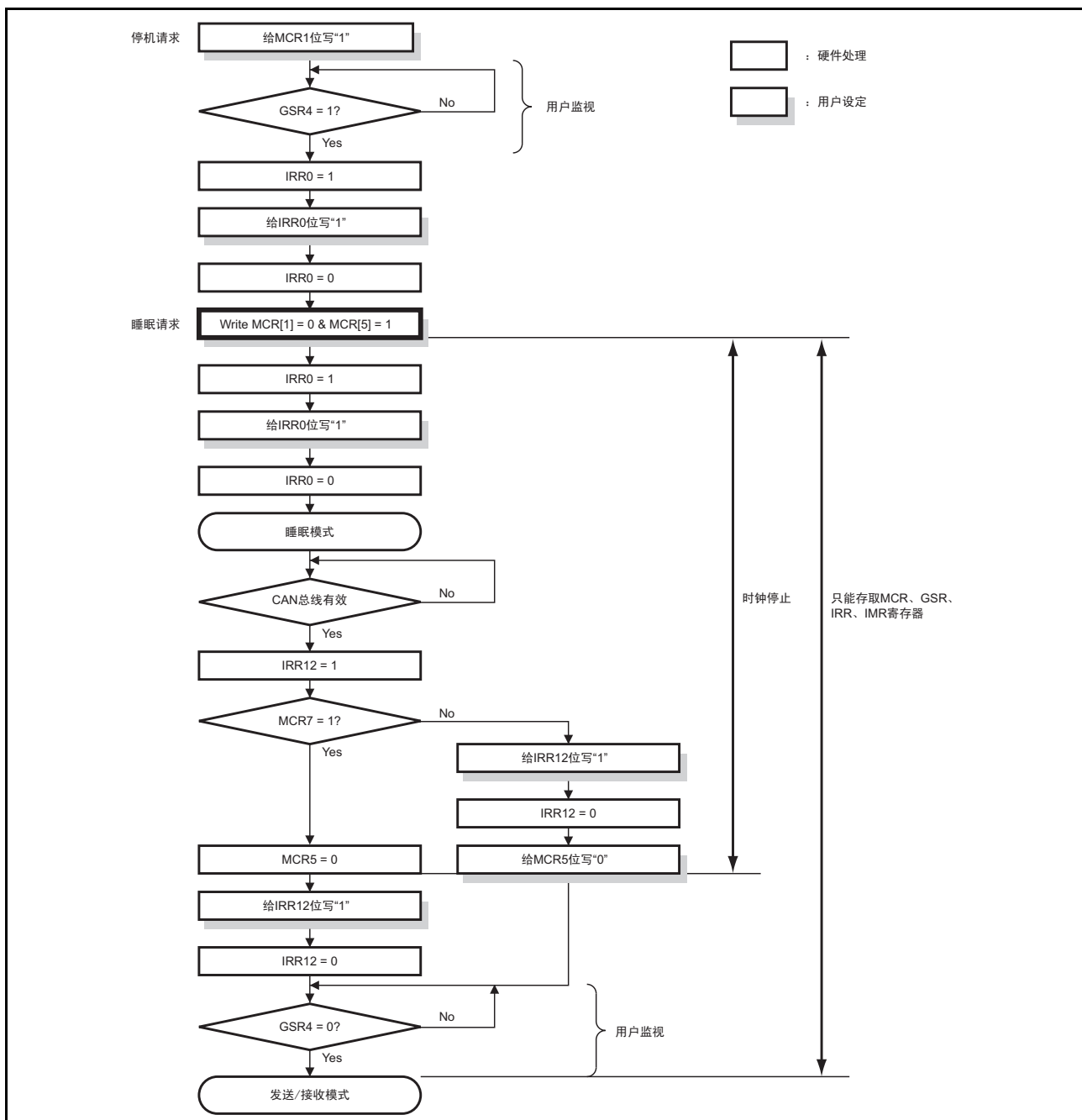


图 23.16 CAN 睡眠模式的流程图

可能的状态转移如图 23.17 所示。

不能在停机模式以外的模式中将 MCR5（CAN 睡眠模式）位置位。

另外，在将 MCR1 位置位后并且在清除 MCR1 前，必须确认 GSR4 的置位，然后将此模块设定为停机模式。

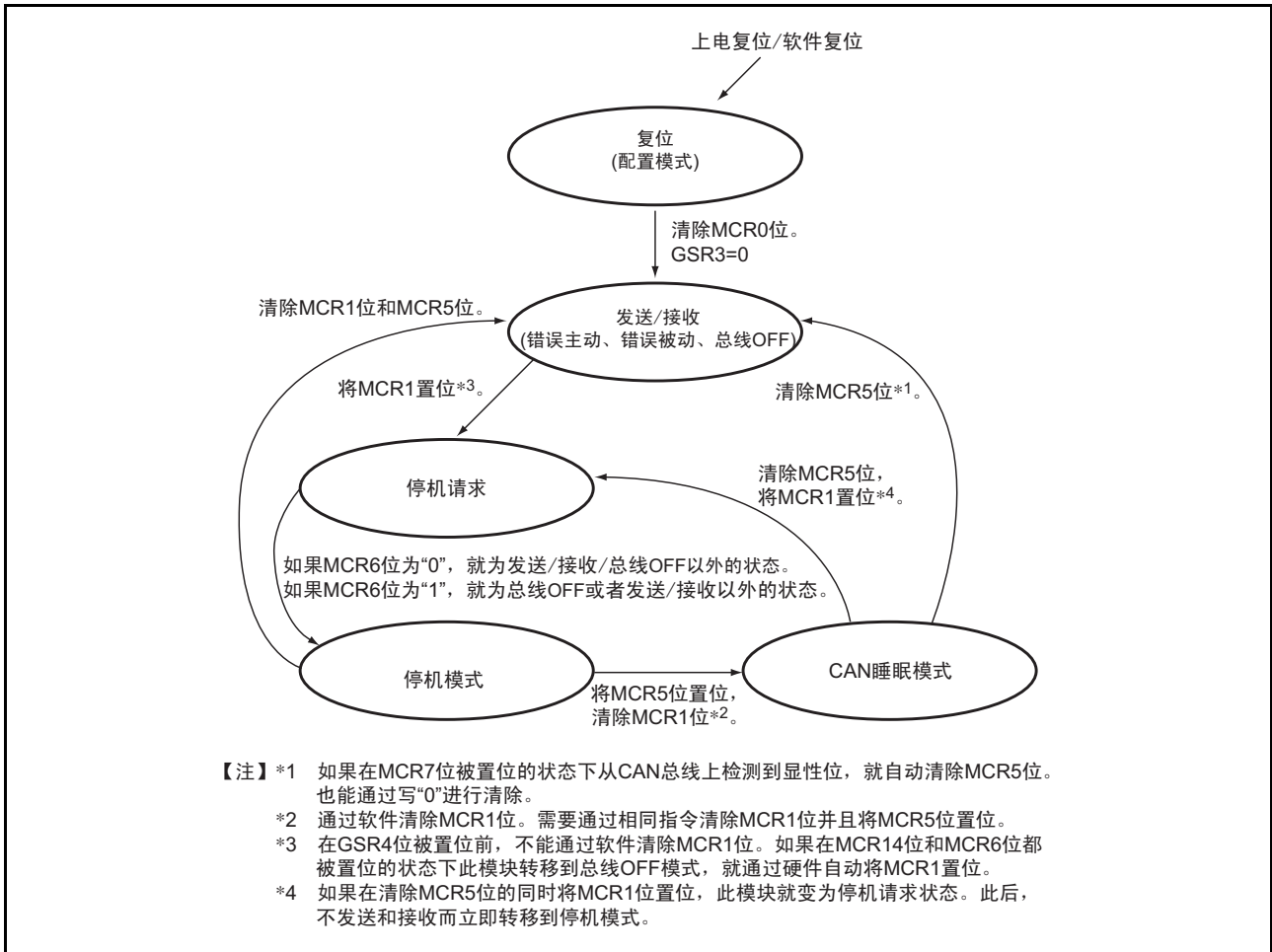


图 23.17 状态转移图

允许在各模式中存取的条件如表 23.9 所示。

表 23.9 能存取的寄存器

状态模式	寄存器									
	MCR、GSR	IRR、IMR	BCR	MBIMR 定时器 TT 寄存器	标志 寄存器	信箱 (控制 0、 LAFM)	信箱 (数据)	信箱 (控制 1)	信箱 (触发时间、 TT 控制)	
复位	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	
发送 / 接收	Yes	Yes	No	Yes	Yes	No Yes*	Yes*	No Yes*	Yes*	
停机请求	Yes	Yes	No	Yes	Yes	No Yes*	Yes*	No Yes*	Yes*	
停机模式	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	
CAN 睡眠模式	Yes	Yes	No	No	No	No	No	No	No	

【符号说明】

Yes: 允许存取的寄存器

No: 禁止存取的寄存器

【注】* 这是没有将 TXPR0 置位的情况。

23.7.2 测试模式的设定

此模块有各种测试模式。通过 MCR 寄存器的 TST[2:0] 位选择测试模式，此模块的默认值（初始值）是以正常模式运行。

测试模式的设定如表 23.10 所示。

只能在配置模式中选择测试模式。要执行所选的测试模式时，必须在选择测试模式后解除配置模式（确认 BCR0/BCR1 的设定）。

表 23.10 测试模式的设定

TST2	TST1	TST0	模式
0	0	0	正常模式
0	0	1	监听专用模式（接收专用模式）
0	1	0	自测试模式 1（外部）
0	1	1	自测试模式 2（内部）
1	0	0	写错误计数器
1	0	1	错误被动模式
1	1	0	禁止设定
1	1	1	禁止设定

- 正常模式
此模块进行正常的运行。
- 监听专用模式
这是用于波特率检测并且由 ISO-11898 要求的模式。在清除错误计数器后禁止运行，而且不递增 TEC/REC 的值。另外，禁止 CT_{xn}（n=0、1、2）的输出并且禁止此模块生成错误帧或者应答位。如果发生信息错误，就将 IRR13 置位。
- 自测试模式（外部）
此模块生成应答位，如果需要就将信息保存到接收信箱。CR_{xn}/CT_{xn}（n=0、1、2）引脚必须连接 CAN 总线。
- 自测试模式（内部）
此模块生成应答位，如果需要就将信息保存到接收信箱。因为内部 CT_{xn}（n=0、1、2）环回到内部 CR_{xn}（n=0、1、2），所以不需要将 CR_{xn}/CT_{xn}（n=0、1、2）引脚连接 CAN 总线或者其他外部设备。CT_{xn}（n=0、1、2）引脚只输出隐性位，而 CR_{xn}（n=0、1、2）引脚无效。
- 写错误计数器
能在此模式中写 TEC/REC。能通过给错误计数器写大于 127 的值，将此模块强制置为错误被动模式。因为 TEC 写入值也被写到 REC，所以总是将 TEC 和 REC 设定为相同值。同样，通过写大于 95 的值，能将此模块强制置为错误警告模式。
必须在此模块为停机模式（在写错误计数器时 MCR1 为“1”）时写 TEC/REC。根据 CAN 的规格，需要在解除停机模式前解除此测试模式，使 TEC/REC 重新运行。
- 错误被动模式
能将此模块强制置为错误被动模式。
虽然错误被动模式的执行不会改变 REC 的值，但是一旦在错误被动模式中运行，只要接收错误，REC 的值就会和正常情况一样进行递增。如果在此模式中 TEC 的值达到 256，此模块就变为总线 OFF 状态，但是如果使用此模式，此模块就不能变为错误主动状态。因此，此模块在总线 OFF 恢复顺序的最后转移到错误被动状态而不是错误主动状态。

当发生信息错误时，在全部测试模式中将 IRR13 位置位。

23.7.3 信息发送顺序

(1) 信息发送请求

将 CAN 帧发送到总线的顺序例子如图 23.18 所示。

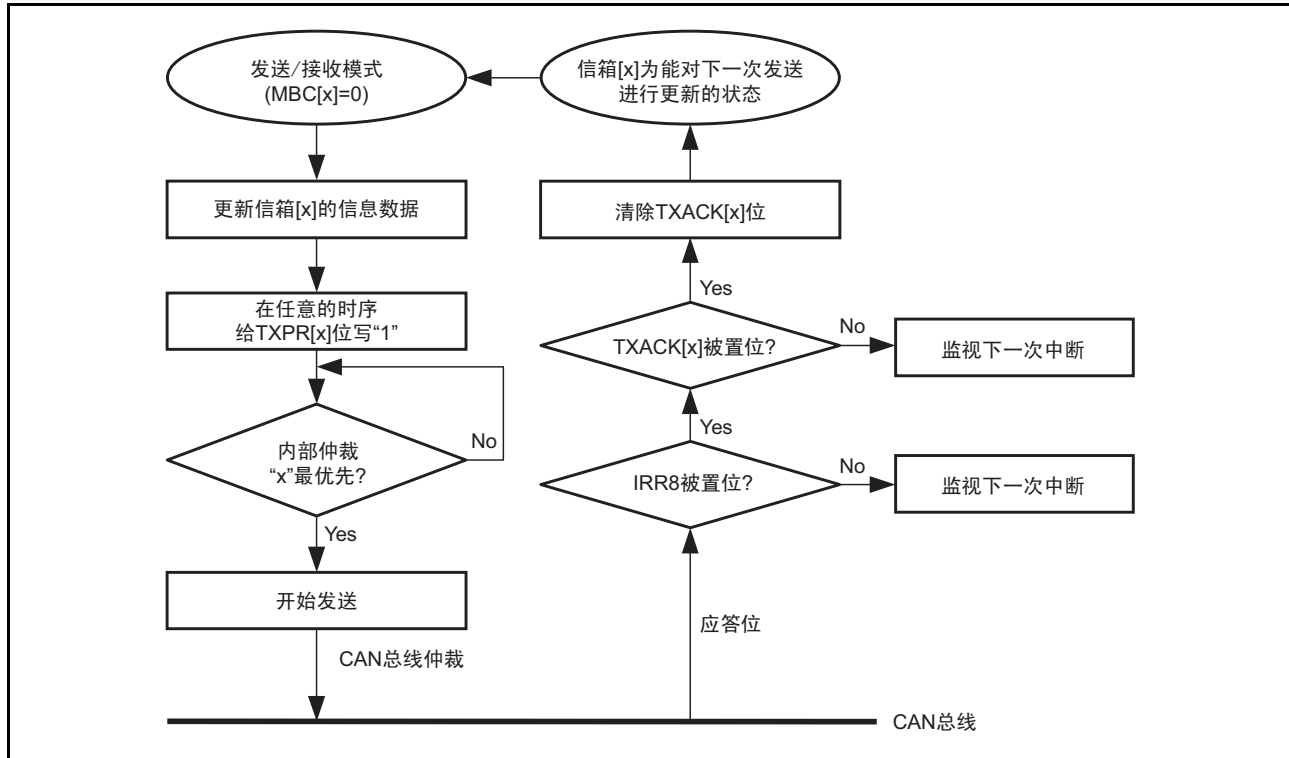


图 23.18 信息发送请求

必须注意：如同寄存器章节的说明，如果 TXACK 位或者 ABACK 位的任意一个位被置位，就将 IRR8 置位。这意味着任意信箱结束发送或者发送中止并且更新为能进行下次发送的状态。另外，GSR2 表示现在没有发生发送请求（全部 TXPR 标志没有被置位）。

(2) 发送的内部仲裁

图 23.19 说明此模块如何按照信息 ID 的顺序进行发送请求的信息调度。通过内部仲裁，在有发送请求的信息中取出优先级最高的信息。

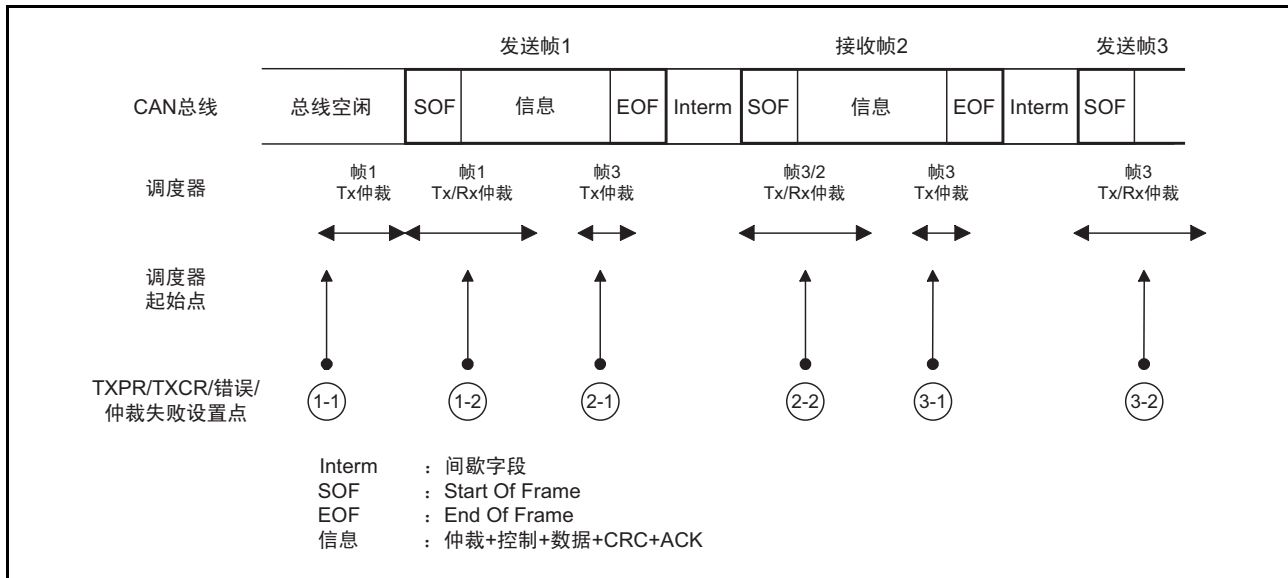


图 23.19 发送的内部仲裁

- 1-1: 如果在 CAN 总线为空闲状态时将 TXPR 位置位，就立即进行内部仲裁并且开始发送。
- 1-2: 在 SOF 处开始发送和接收。因为不是接收帧，所以此模块进行发送。
- 2-1: 通过 CRC 的定界符开始内部仲裁，寻找下一次要发送的信息。
- 2-2: 在 SOF 处开始发送和接收。因为是优先级高的接收帧，所以此模块进行接收。因此，不发送帧 3 而进行接收。
- 3-1: 通过 CRC 的定界符开始内部仲裁，寻找下一次要发送的信息。
- 3-2: 在 SOF 处开始发送和接收。因为发送帧的优先级高于接收帧，所以此模块进行发送。

如果在 CAN 总线上检测到错误，就在各错误定界符的开头进行下次发送的内部仲裁，并且在过载帧后的错误定界符的开头也进行下次发送的内部仲裁。

因为通过 CRC 的定界符进行发送仲裁，所以当 ATX 为“1”的信箱接收到远程帧请求时，根据此请求的发送信息加入发送仲裁的时序为此后的总线空闲、CRC 定界符或者错误定界符。

根据 CAN 总线的状态，在将 TXCR 置位后进行对应最大 1 个 CAN 帧延迟后的信息中止处理。

(3) 时间触发发送

此模块通过硬件支持符合 ISO-11898-4 TTCAN Level 1 规格的时间触发模式通信。

在此说明使用时间触发模式的基本步骤。

• 时间触发模式的设定

为了设置时间触发模式，需要进行以下设定：

- 将 CMAX_TEW 的 CMAX 设定为“B'111”以外的值。
- 将 TTCR0 的 bit15 置位，开始 TCNTR 的运行。
- 为了在比较匹配时不清除 TCNTR，清除 TTCR0 的 bit6。
- 为了进行周期发送，不清除进行时间触发发送的信箱的 TXPR。

- 各寄存器的作用

能使用此模块的用户寄存器进行 TTCAN 规格所要求的主要功能的处理。

TCNTR	Local_Time
RFMK	Ref_Mark
CYCTR	周期时间 = TCNTR - RFMK
RTROFF	信箱 30 的 Ref_Trigger_Offset
信箱 31	时间基准信息的专用接收信箱
信箱 30	时间基准信息的专用发送信箱（作为潜在时间主控运行的情况）
信箱 29 ~ 24	时间触发发送对应的信箱
信箱 23 ~ 16	无时戳接收对应的信箱（也能用作事件触发发送的信箱）
信箱 15 ~ 0	带时戳接收对应的信箱（也能用作事件触发发送的信箱）
Tx-Trigger Time	指定发送信息时序的 Time_Mark
CMAX	指定作为潜在时间主控运行时的周期计数值（Basic Cycle）的最大值。
TEW	指定 Tx_Enable_Window 的宽度。
TCMR0	Init_Watch_Trigger（和本地时间进行比较匹配）
TCMR1	用于用户指定事件监视，和周期时间进行比较匹配
TCMR2	Watch_Trigger（和周期时间进行比较匹配） 能设定为中止全部处于等待状态的发送。
TTW	指定用于发送的时间窗口的属性。
TTTSEL	指定等待下次发送的信箱。

- 时间主控/时间从属

此模块能设定为作为网络的潜在时间主控运行或者时间从属运行。各模式所需的设定和此模块自动进行的处理如下表所示：

模式	设定	功能
时间从属	TXPR[30]=0 & MBC[30]≠B'000 & CMAX≠B'111 & MBC[31]=B'011	在每次检测到 CAN 总线上 SOF 时对 TCNTR 进行采样，并且保存到内部寄存器。如果信箱 31 接收到有效时间基准信息，就将 TCNTR 的值（通过 SOF 保存）复制到 Ref_Mark（RFMK）。 将接收到的基准信息的周期计数器值（CCR）复制到自节点的周期计数寄存器（CCR）。 如果 Next_is_Gap 为“1”，就将 IRR13 置位。
（潜在）时间主控	TXPR[30]=1 & MBC[30]=B'000 & DLC[30]>0 & CMAX≠B'111 & MBC[31]=B'011	有以下 2 种情况： 1. 如果信箱 31 接收到有效的的时间基准信息，就将通过 SOF 保存在内部寄存器的 TCNTR 值复制到 Ref_Mark（RFMK）。将接收到的基准信息的周期计数器值（CCR）复制到自节点的周期计数寄存器（CCR），如果 Next_is_Gap 为“1”，就将 IRR13 置位。 2. 如果从信箱 30 发送时间基准信息，就将通过 SOF 保存在内部寄存器的 TCNTR 值复制 Ref_Mark（RFMK）。如果信箱 30 的 TTT 和 CYCTR 相同，就将周期计数器（CCR）递增。 将周期计数器（CCR）嵌入时间基准信息的第 1 个数据字节（Data0[7:6]、CCR[5:0]）。

• 发送触发时间的设定

必须按以下所示的升序设定发送触发时间 (TTT)。TTT 间的差需要满足以下计算式。计算式中的 TEW 为寄存器的值。

$$TTT(\text{信箱}24) < TTT(\text{信箱}25) < TTT(\text{信箱}26) < TTT(\text{信箱}27) < TTT(\text{信箱}28) < TTT(\text{信箱}29) < TTT(\text{信箱}30) \text{ 并且}$$

$$TTT(\text{信箱}i) - TTT(\text{信箱}i - 1) > TEW + \text{最大帧长度} + 9$$

如果作为潜在时间主控运行, TTT (信箱 24) ~ TTT (信箱 29) 就对应 Time_Mark 并且 TTT (信箱 30) 对应表示 Basic_Cycle length 的 Time_Ref。

上述限制只适用于设定为时间触发发送的信箱。

【重要】 根据发送触发时间的设定限制, 只有 1 个能分配 1 个时间窗口的信箱。

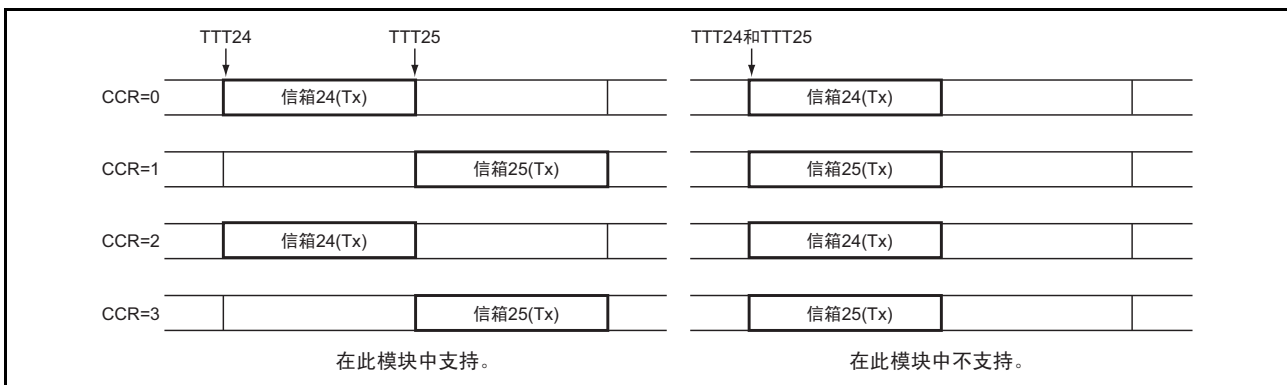


图 23.20 发送触发时间的限制

作为 Watch_Trigger 的 TCMR2 的值必须大于表示 1 个 Basic_Cycle length 的 TTT (信箱 30)。

(潜在) 时间主控和时间从属的设定例子如图 23.21 和图 23.22 所示。图中的 L 为时间基准信息的时间长度。

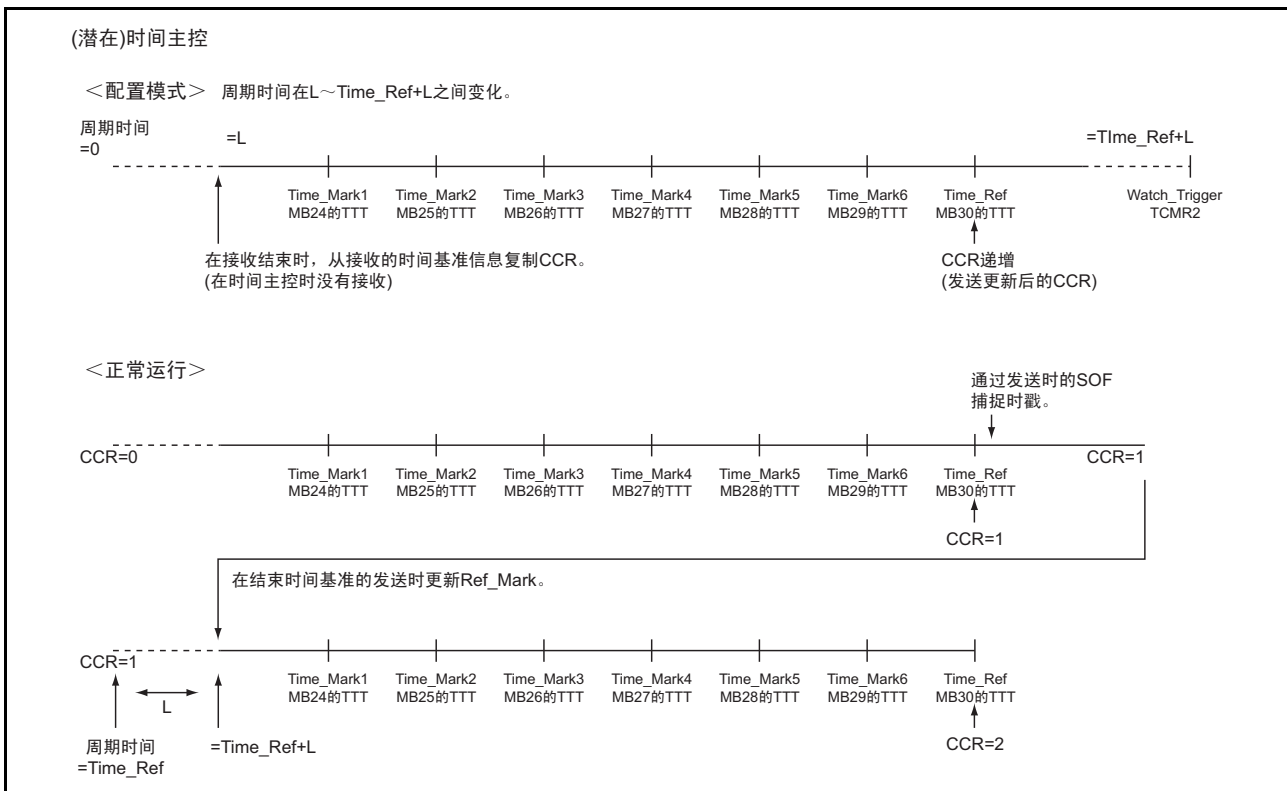


图 23.21 (潜在) 时间主控

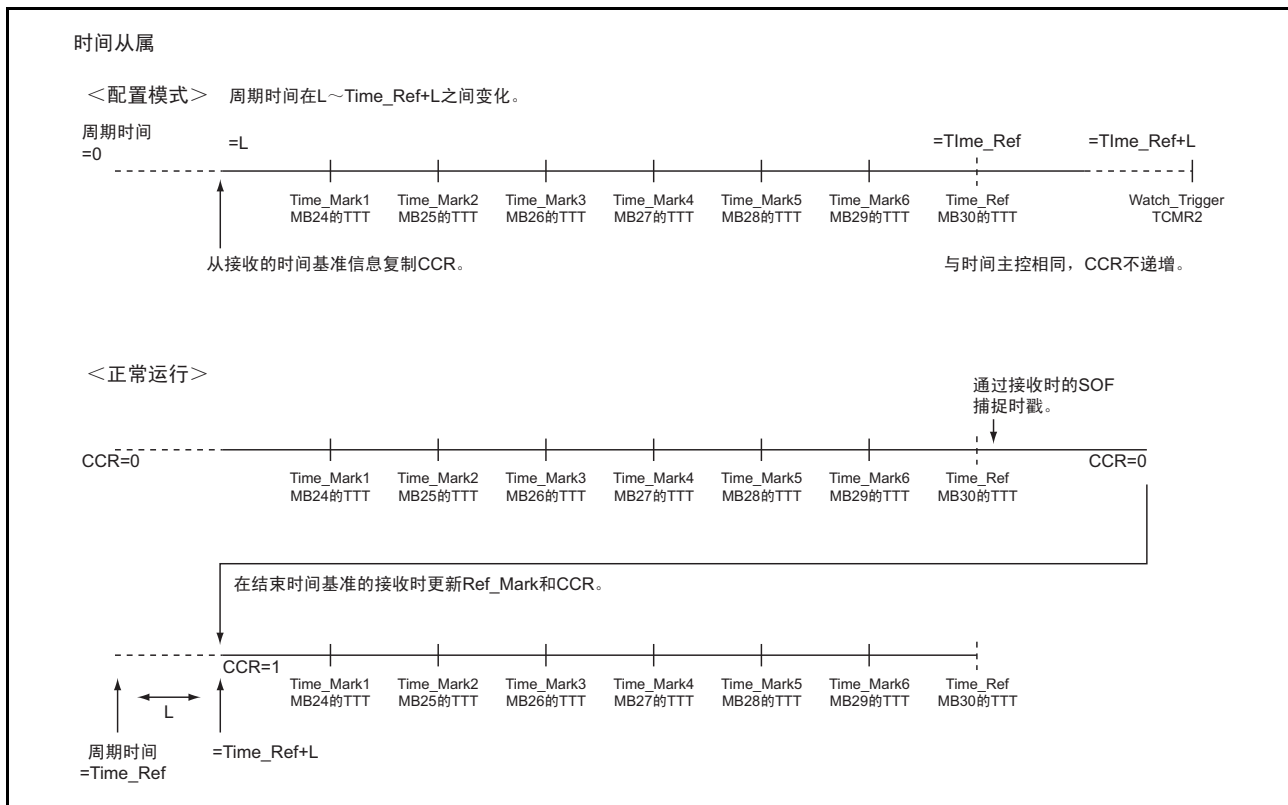


图 23.22 时间从属

- 通过软件实现的功能

在 TTCAN 的功能中有需要通过软件实现的功能。主要功能如下所示。详细内容请参照 ISO-11898-4。

- 从 Init_Watch_Trigger 变为 Watch_Trigger

作为 Init_Watch_Trigger 和 Watch_Trigger 的硬件支持，此模块分别准备了 TCMR0 寄存器和 TCMR2 寄存器。在检测到 CAN 总线上的最初的基准信息前，通过软件将 TCMR0 置为有效而将 TCMR2 置为无效，然后通过软件控制将 TCMR0 置为无效而将 TCMR2 置为有效。（同步调度状态机）

只支持 Next_is_Gap 接收中断。需要通过应用软件将对应的 TXCR 标志置位，以便在结束当前的 Basic_Cycle 时停止全部的发送。

主控 / 从属模式控制

只支持自动周期时间同步和 CCR 的递增。

- 信息状态计数

需要通过软件对互斥窗口的周期信息有关的调度错误进行计数。

- 时间触发通信的信息发送请求

使用时间触发模式时，必须满足 ISO11898-4 的要求事项。

必须进行以下步骤：

1. 将此模块置为复位或者停机模式。
2. 给TCMR0设定Init_Watch_Trigger (H'FFFF)。
3. 通过TTCR0的bit10将由TCMR0产生的比较匹配置为有效。
4. 给TCMR2设定指定的Watch_Trigger值。
5. 将TTCR0的bit12保持为“0”，将由TCMR2产生的比较匹配置为无效。
6. 给CMAX设定所需的值 (B'111以外)。
7. 给TEW设定所需的值。
8. 将信箱设定为时间触发的发送和接收。
9. 给低3位设定信箱31的LAFM。
10. 给MCR、BCR1、BCR0设定所需的值。
11. 作为潜在时间主控运行时的设定如下：
 - 给RFTROFF设定所需的Init_Ref_Offset值。
 - 将信箱30的TXPR置位。
 - 给TTTSEL写“H'4000”。
12. 通过TTCR0的bit15将定时器TCNTR置为允许。
13. 转移到正常模式。
14. 等待发送或者接收有效的基准信息，或者等待TCMR0的比较匹配。
15. 如果本地时间 (TCNTR) 和TCMR0的值相同，就表示达到Init_Watch_Trigger，通过应用程序将信箱30的TXCR置位，重新开始。
16. 如果发送了基准信息 (TXACK[30]被置位)，就将RFTROFF置“0”。
17. 如果接收到有效的基准信息 (RXPR[31]被置位)，就进行以下操作：
 - 如果信箱31的ID低3位的优先级高于信箱30的低3位 (作为潜在时间主控运行的情况)，就将RFTROFF的值保持为Init_Ref_Offset。
 - 如果信箱31的ID低3位的优先级低于信箱30的低3位 (作为潜在时间主控运行的情况)，就将RFTROFF的值减1。
18. 清除TTCR0的bit10，将由TCMR0产生的比较匹配置为无效。
19. 将TTCR0的bit12置位，将由TCMR2产生的比较匹配置为有效。
20. 在检测到CAN总线上的2个基准信息 (发送或者接收) 后，能通过应用程序将其他时间触发信箱的TXPR置位。

如果在CAN总线上没有检测到基准信息而周期时间CYCTR达到TCMR2的值，此模块就立即自动中止全部等待状态的发送 (包括基准信息)。

在时间触发模式中再次请求发送时的顺序如下所示：

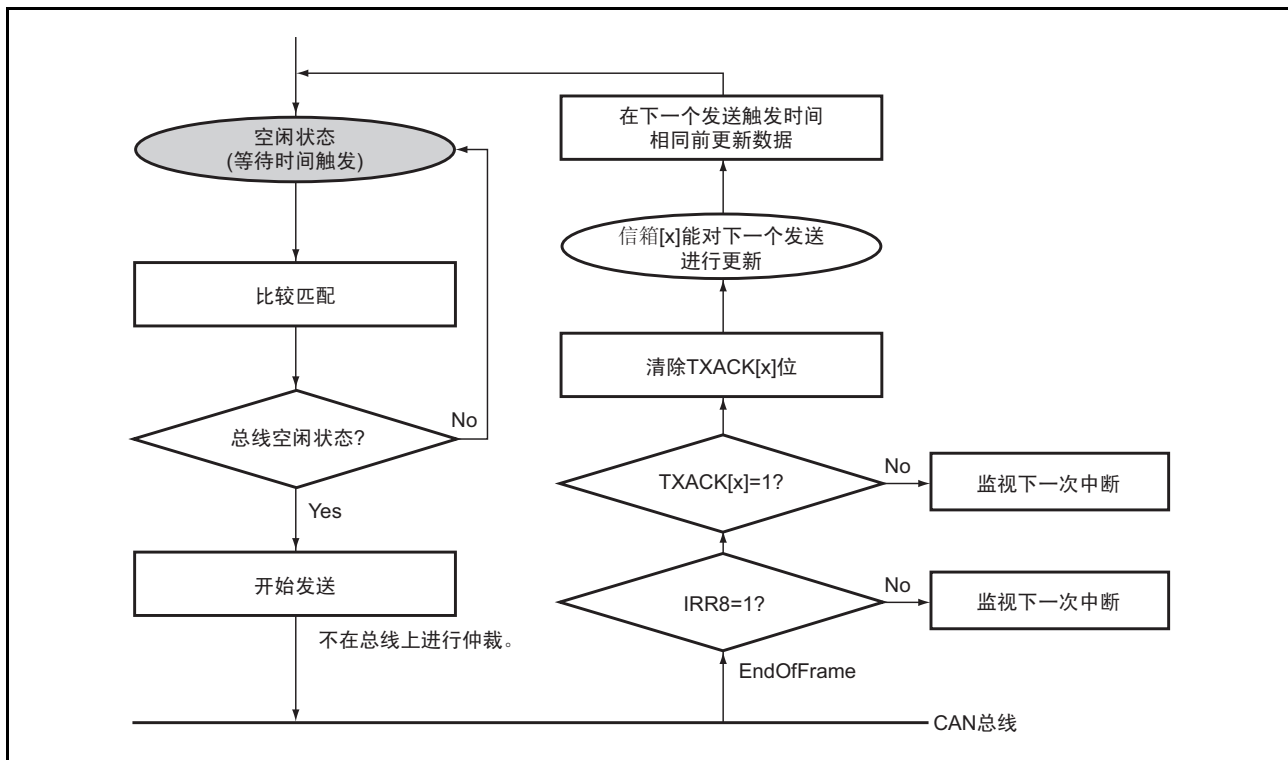


图 23.23 信息发送请求

在发生发送触发前，软件必须正确地更新信息。如果 $CYCTR$ 达到信箱的 TTT （发送触发时间）并且 CCR 和程控发送周期相同，此模块就立即将信息传送到发送缓冲器。

此时，此模块在指定的 $Tx_Enable\ Window$ 中尝试发送。如果错过此时间槽，就在发送为固定周期（信箱 24 ~ 30）时此模块将对应的 $TXPR$ 位保持为“1”，并且将发送请求保留到下一个发送触发为止。

此模块错过上述时间槽的原因有 3 个：

1. 正在使用 CAN 总线。
2. 在发送时间触发信息过程中 CAN 总线发生错误。
3. 在发送时间触发信息过程中发生仲裁失败。

在合并仲裁窗口（Merged Arbitrating Window）时，此发送槽为从窗口开始（ TTW 为“B'10”）的信箱的发送触发到窗口结束（ TTW 为“B'11”）的信箱的 TEW 最后。能随时更改 $TXPR$ 。此模块总是正确地调度发送时间触发信息，但是为了保证正确的调度，有以下所示的重要规则：

- 能在配置模式中更改 TTT （发送触发时间）。
- 不能将 TTT 设定为超过指定 $Basic_Cycle\ length$ 周期计数器（Basic Cycle）长度的 $Time_Ref$ 。如果违反此规定，调度就会出现问題。
- 在周期发送时不自动清除 $TXPR$ 。需要取消周期发送时，通过应用程序将对应的 $TXCR$ 位置位。

- 时间触发系统的例子

使用时间从属模式的此模块，使时间触发系统运行的简单例子如下图所示。

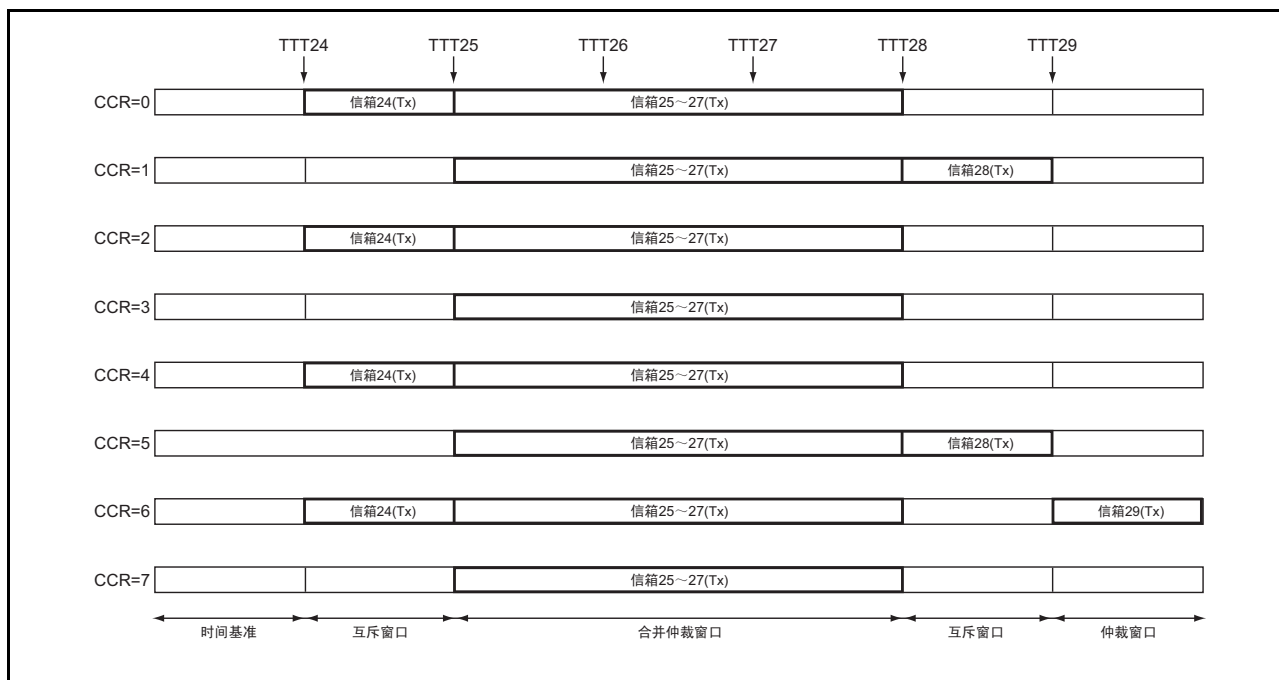


图 23.24 作为时间从属的时间触发系统例子

图 23.24 的例子中使用的值如表 23.11 所示。

表 23.11 例子中使用的设定值

	rep_factor (寄存器)	Offset	TTW[1:0]	MBC[2:0]
信箱 24	B'001	B'000000	B'00	B'000
信箱 25	B'000	B'000000	B'10	B'000
信箱 26	B'000	B'000000	B'10	B'000
信箱 27	B'000	B'000000	B'11	B'000
信箱 28	B'010	B'000001	B'00	B'000
信箱 29	B'011	B'000110	B'01	B'000
信箱 30	—	—	—	B'111
信箱 31	—	—	—	B'011

【注】 CMAX 为 “B'011” 并且 TXPR[30] 为 “0”。

在合并仲裁窗口中，通过 FCFS (First Come First Served) 处理时间触发发送请求。例如：如果信箱 25 不能在发送触发时间 25 (TTT25) 和发送触发时间 26 (TTT26) 之间进行发送，就在 TTT26-TTT28 之间信箱 25 的优先级高于信箱 26。

要将时间触发发送置为无效时，将 MBC 设定为 “B'111”。在此模块为时间主控时，必须将 MBC[30] 设定为 “B'000”，自动将时间基准窗口认为是仲裁窗口。

• 定时器运行

定时器的时序图如图 23.25 所示。假设发送触发时间设定为 n，就在 CYCTR=n+2 和 CYCTR=n+3 之间开始时间触发发送。

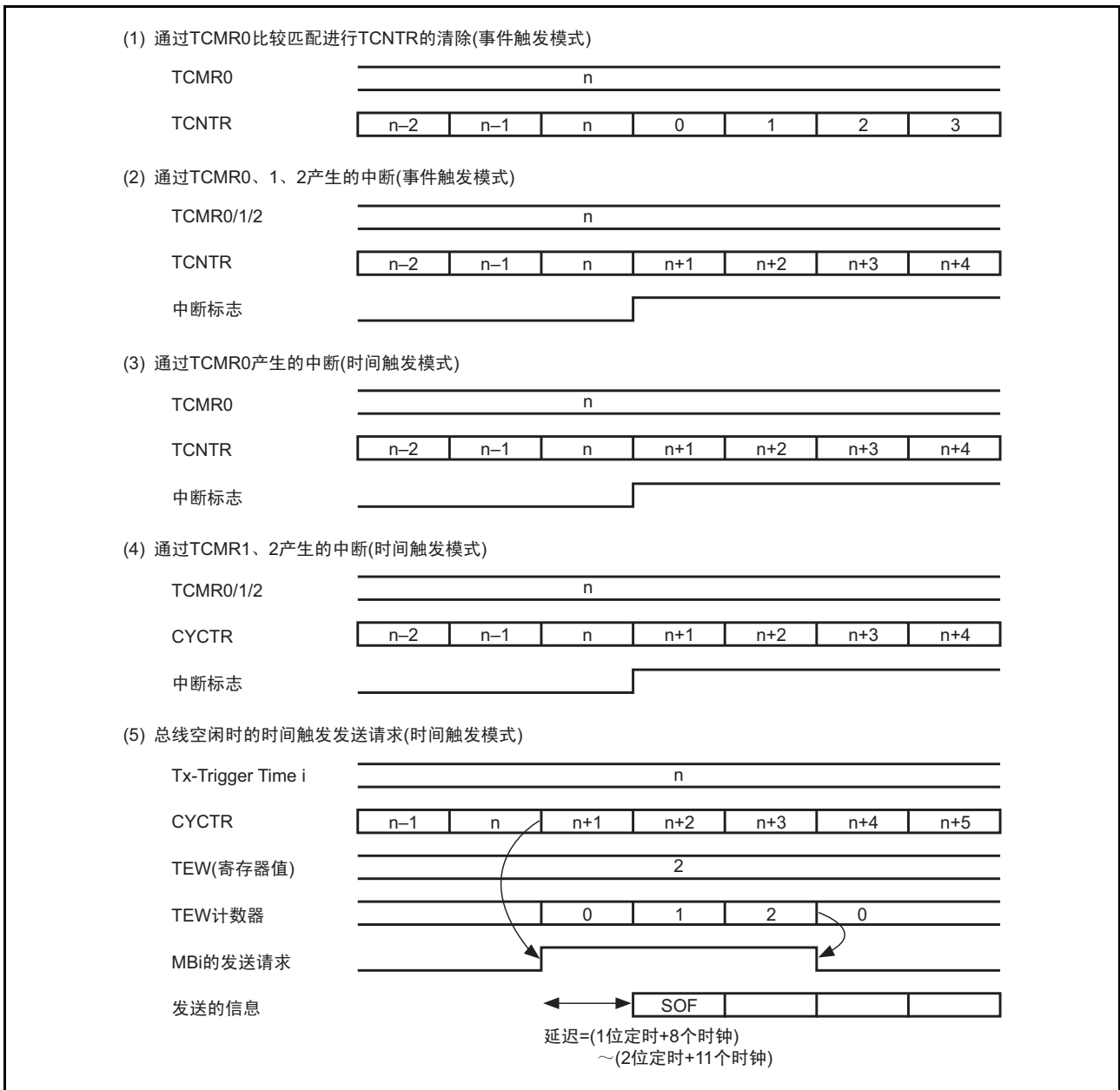


图 23.25 定时器的时序图

在合并仲裁窗口中，在结束定时器的触发发送后处理事件触发发送。例如：如果在结束信箱 25 的发送时 CYCTR 未达到 TTT26，就根据 MCR2 指定的信息发送的优先级，开始事件触发发送。在发送结束后不清除定时器触发发送的 TXPR 而清除事件触发发送的 TXPR。

如果不将关闭合并仲裁窗口的信箱的 TXPR 置位，就在此信箱的 TTT 后的 TEW 的最后关闭合并仲裁窗口。

请参照“23.3.1 信箱结构”的表 23.3。

23.7.4 信息接收顺序

信息接收顺序如图 23.26 所示。

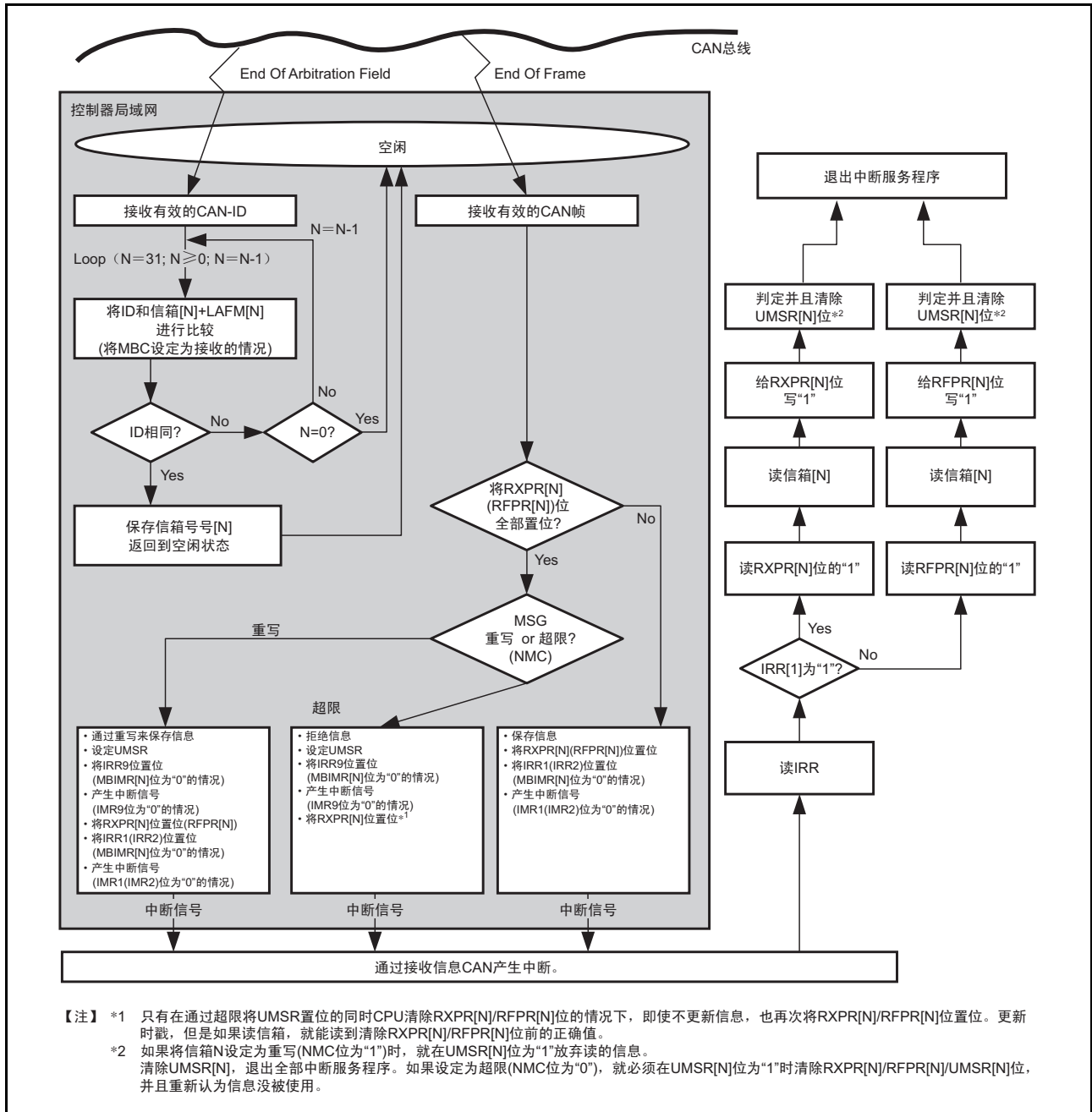


图 23.26 信息接收顺序

如果此模块在接收信息的过程中认识到仲裁字段的最后，就开始将接收的 ID 和信箱设定的 ID 进行比较。比较顺序为从信箱 31 到信箱 0 的顺序。首先检查 MBC 并且调查是否设定为接收信箱，然后读 LAFM，接着读信箱 31（设定为接收的情况）的 ID，和接收的 ID 进行比较。如果不相同，就对信箱 30（设定为接收的情况）进行同样的检查。如果发现相同的 ID，此模块就将此信箱号（N）保存到内部缓冲器，并且在停止搜索后返回到空闲状态，等待 EOF（End Of Frame）。如果被通知 EOF 的第 6 位，就根据 NMC 位的设定，放弃或者写接收信息。

不能在通信过程中更改此模块的信息 ID 和 LAFM 的设定。作为更改设定的 1 种办法，有停机模式和配置模式。在给对应的信箱写接收信息时，因为包含信息 ID，所以在使用 LAFM 时 CAN-ID 可能被不同的 CAN-ID 重写。这也意味着在接收的信息 ID 和多个信箱的 ID+LAFM 相同时，总是将接收信息保存在信箱号最大的信箱，号小的信箱不保存信息。因此，需要特别注意选择 ID 和 LAFM 的设定值。

在如图 23.11 所示的接收数据和远程帧时，为了在执行中断服务程序过程中检测信息被相同信箱保存的新信息重写（NMC 为“1”的情况），需要在读 IRR 后清除 UMSR 标志。如果在检测 UMSR 的最后时检测到重写，就需要放弃信息并且重新读。

当信箱被设定为超限（NMC 为“0”）时，UMSR 被置位的信息有效。但是，在 CAN 总线上被监视的不是最新信息而是旧信息。必须在清除相关的 RXPR/RFPFR 标志前读信息。

必须注意：如果接收的远程帧被数据帧重写，就将远程帧接收中断（IRR2）和数据帧接收中断（IRR1）置位，并且也将接收标志（RXPR 和 RFPFR）置位。同样，如果数据帧被远程帧重写，也将 IRR2 和 IRR1 置位。

如果接收信息并且保存到信箱，就将未接收的数据字段全部保存为“0”。接收标准 ID 的情况也相同。能给扩展 ID（EXTID[17:0]）写“0”。

23.7.5 信箱的重新设定

当需要重新设定信箱时，必须按以下步骤进行：

(1) 发送箱设定的更改

有以下 2 种情况：

- ID、RTR、IDE、LAFM、数据、DLC、NMC、ATX、DART 的更改
只能在 MBC 为“B'000”时进行更改。必须确认对应的 TXPR 位没有被置位。能随时更改设定（MBC 除外）。
- 从发送箱到接收箱的更改
必须确认对应的 TXPR 位没有被置位。只能在停机模式或者复位状态下进行更改。此模块在接收或者发送信息的过程中可能需要一定的时间才能转移到停机状态（这是因为需等到接收或者发送结束后才转移到停机状态）。必须注意：不能在停机状态下发送和接收信息。
在此模块为总线 OFF 状态时，根据 MCR 寄存器的 bit6 和 bit14 的设定进行停机状态的转移。

(2) 接收箱的 ID、RTR、IDE、LAFM、数据、DLC、NMC、ATX、DART、MBC 设定的更改以及从接收箱到发送箱的更改

只能在停机模式中进行设定的更改。当信息存在于 CAN 总线上并且此模块为接收模式时，不会丢失此信息。此模块在结束正在进行的接收后转移到停机模式。此模块在接收或者发送信息的过程中可能需要一定的时间才能转移到停机状态（这是因为需等到接收或者发送结束后才转移到停机状态）。必须注意：不能在停机状态下发送和接收信息。

在此模块为总线 OFF 状态时，根据 MCR 寄存器的 bit6 和 bit14 的设定进行停机状态的转移。

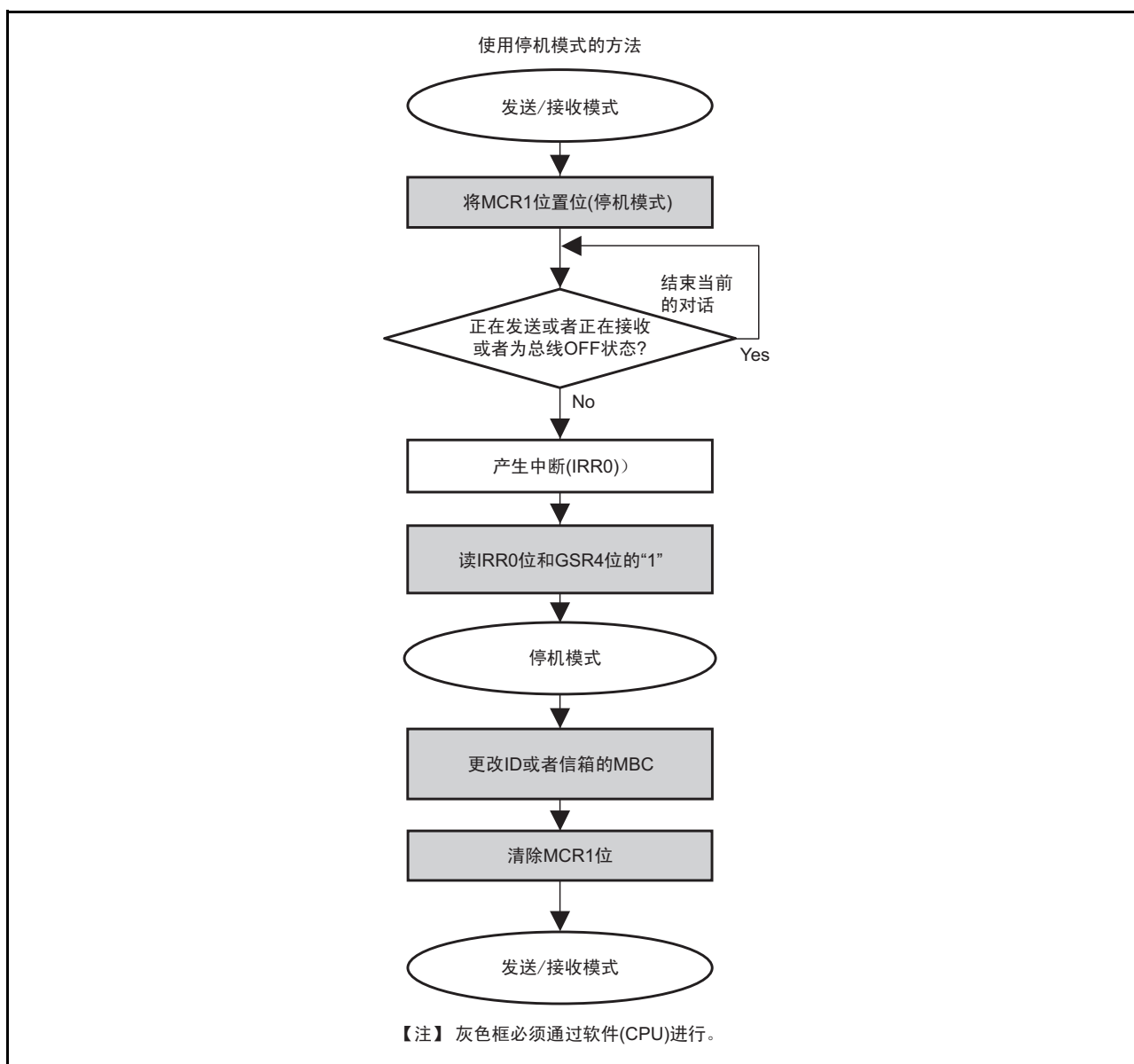


图 23.27 接收箱 ID 的更改 / 从接收箱到发送箱的更改

23.8 中断源

此模块有表 23.12 所示的中断源，能屏蔽这些中断源。要屏蔽时，使用信箱中断屏蔽寄存器（MBIMR）和中断屏蔽寄存器（IMR）。有关各中断请求的中断向量，请参照“7. 中断控制器”。

表 23.12 中断源

名称	中断源	中断标志	直接存储器存取控制器的启动
ERSn*1	错误被动（TEC \geq 128 或者 REC \geq 128）	IRR5	不能
	总线 OFF（TEC \geq 256）/ 从总线 OFF 的返回	IRR6	
	错误警告（TEC \geq 96）	IRR3	
	错误警告（REC \geq 96）	IRR4	
OVRn*1	复位 / 停机 / CAN 睡眠转移	IRR0	
	过载帧发送	IRR7	
	未读信息的重写（超限）	IRR9	
	系统矩阵的开始	IRR10	
	TCMR2 的比较匹配	IRR11	
	CAN 睡眠中 CAN 总线运行的检测	IRR12	
	定时器超限 / Next_is_Gap / 信息错误	IRR13	
	TCMR0 的比较匹配	IRR14	
	TCMR1 的比较匹配	IRR15	
RMn0*1*2	数据帧的接收	IRR1*3	能*4
RMn1*1*2	远程帧的接收	IRR2*3	
SLEn*1	信息的发送 / 发送取消（槽空）	IRR8	不能

【注】 *1 n=0、1、2

*2 RM0 是由信箱 0 的远程帧接收标志（RFPR0[0]）或者数据帧接收标志（RXPR0[0]）产生的中断，RM1 是由信箱 n（n=1～31）的远程帧接收标志（RFPR0[n]）或者数据帧接收标志（RXPR0[n]）产生的中断。

*3 IRR1 是信箱 0～31 的数据帧接收标志，IRR2 是信箱 0～31 的远程帧接收标志。

*4 只能通过 RMn0 中断来启动直接存储器存取控制器。

23.9 直接存储器存取控制器接口

如果将信息接收到各此模块的信箱 0，就能启动直接存储器存取控制器。如果设定直接存储器存取控制器的启动并且结束 DMA 传送，就自动清除 RXPR0 寄存器和 RFPR0 寄存器的标志。此时，在此模块发生接收中断时不会向 CPU 请求中断。DMA 传送的流程图如图 23.28 所示。

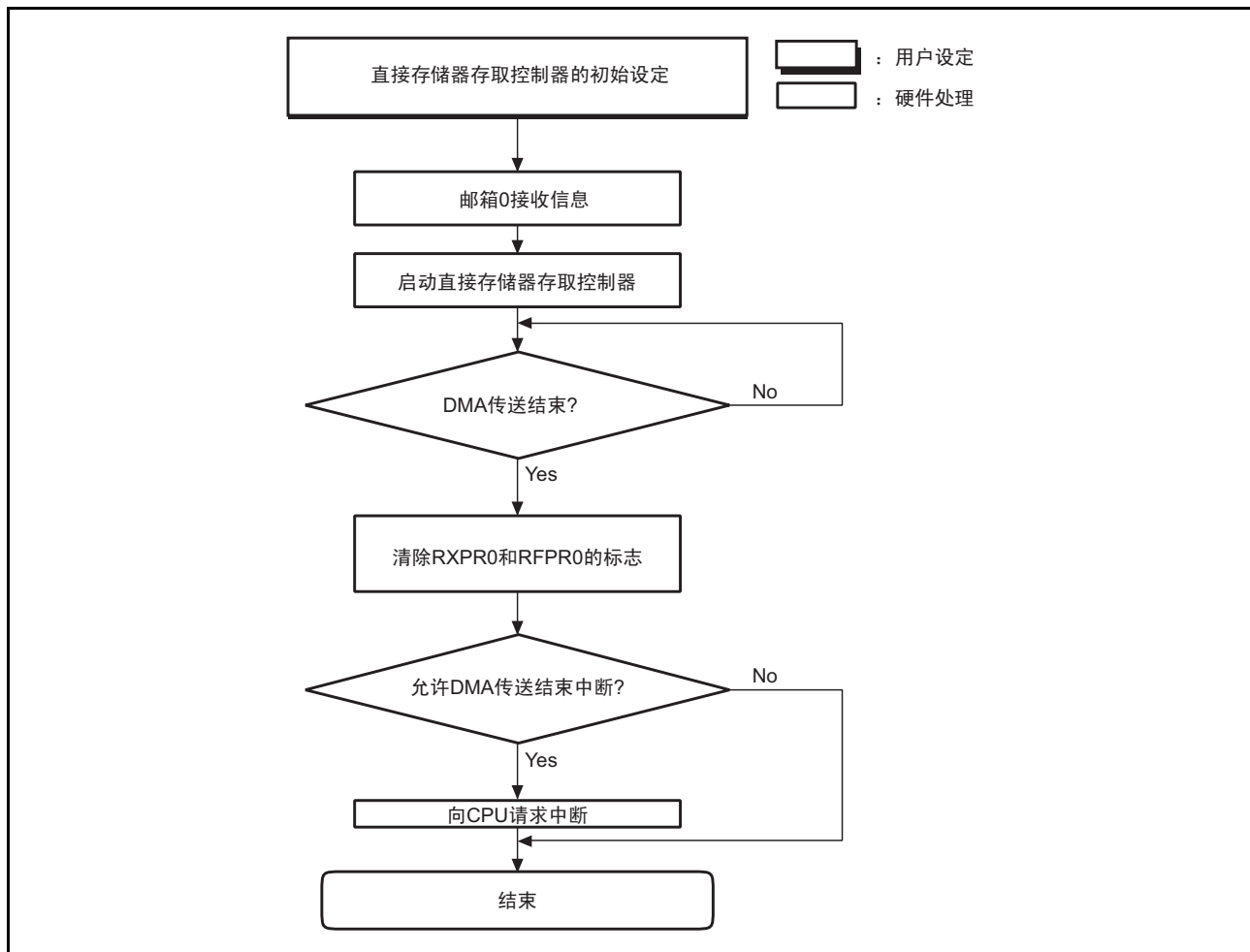


图 23.28 DMA 传送的流程图

23.10 CAN 总线接口

要将本 LSI 和 CAN 总线连接时，需要总线收发器 IC。建议使用瑞萨 HA13721 的收发器 IC。如果使用 HA13721 以外的产品，就必须使用与 HA13721 兼容的产品。连接例子如图 23.29 所示。

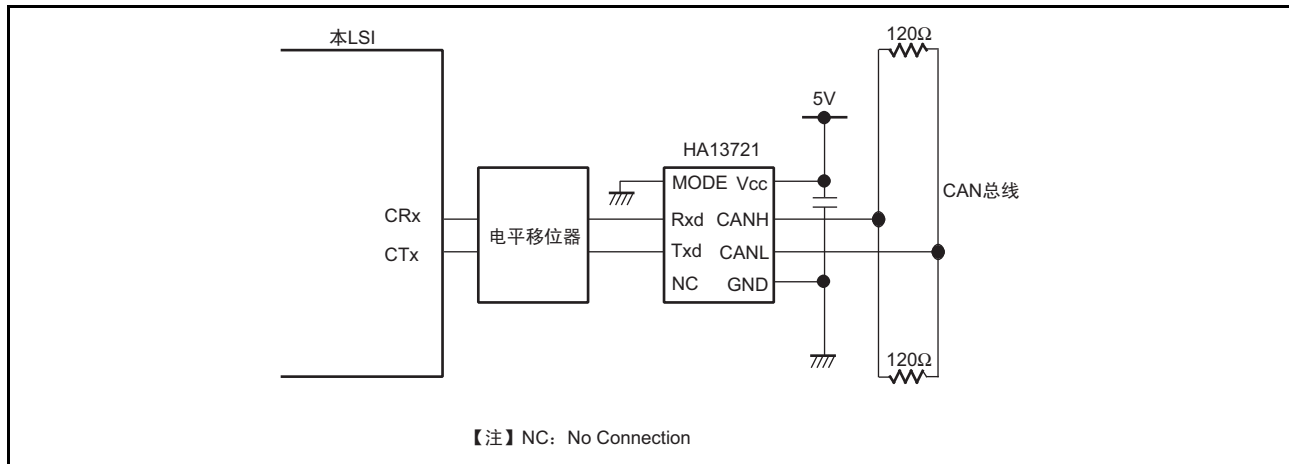


图 23.29 使用 HA13721 的高速 CAN 接口

23.11 引脚端口的设定

必须在配置模式中或者在进入配置模式前设定端口。有关端口设定方法的详细内容，请参照“48. 通用输入/输出端口”。本 LSI 内置的此模块有 3 个通道，有 3 种使用方法：

- 3 个通道的 32 个信箱
- 1 个通道的 64 个信箱（RCAN_0、1）和 1 个通道的 32 个信箱（RCAN_2）
- 1 个通道的 96 个信箱

【注】 在使用 64 个或者 96 个信箱时需要注意。请阅读“23.12.1 有关 1 个通道 64 个或者 96 个信箱的端口设定注意事项”。

各端口设定的连接例子如图 23.30、图 23.31 和图 23.32 所示。

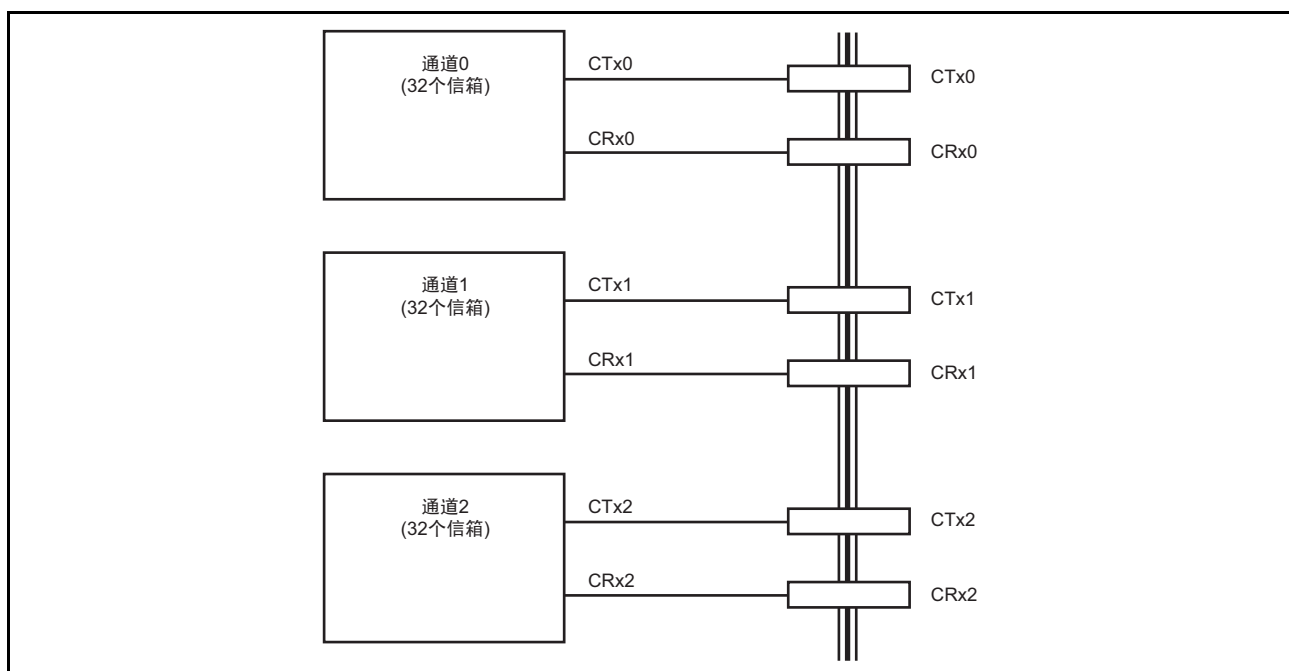


图 23.30 将通道 0、1、2 用作个别通道时的连接例子

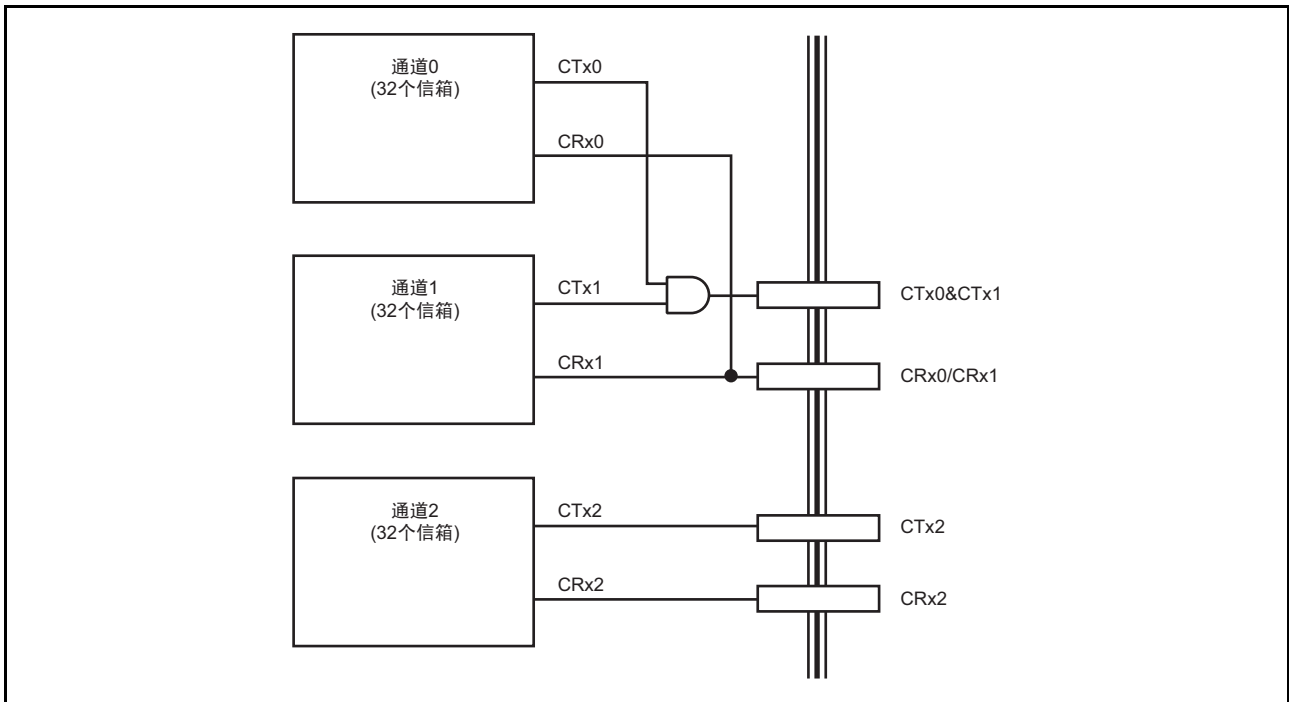


图 23.31 将通道 0、1 用作 64 个信箱的 1 个通道并且将通道 2 用作 32 个信箱的 1 个通道时的连接例子

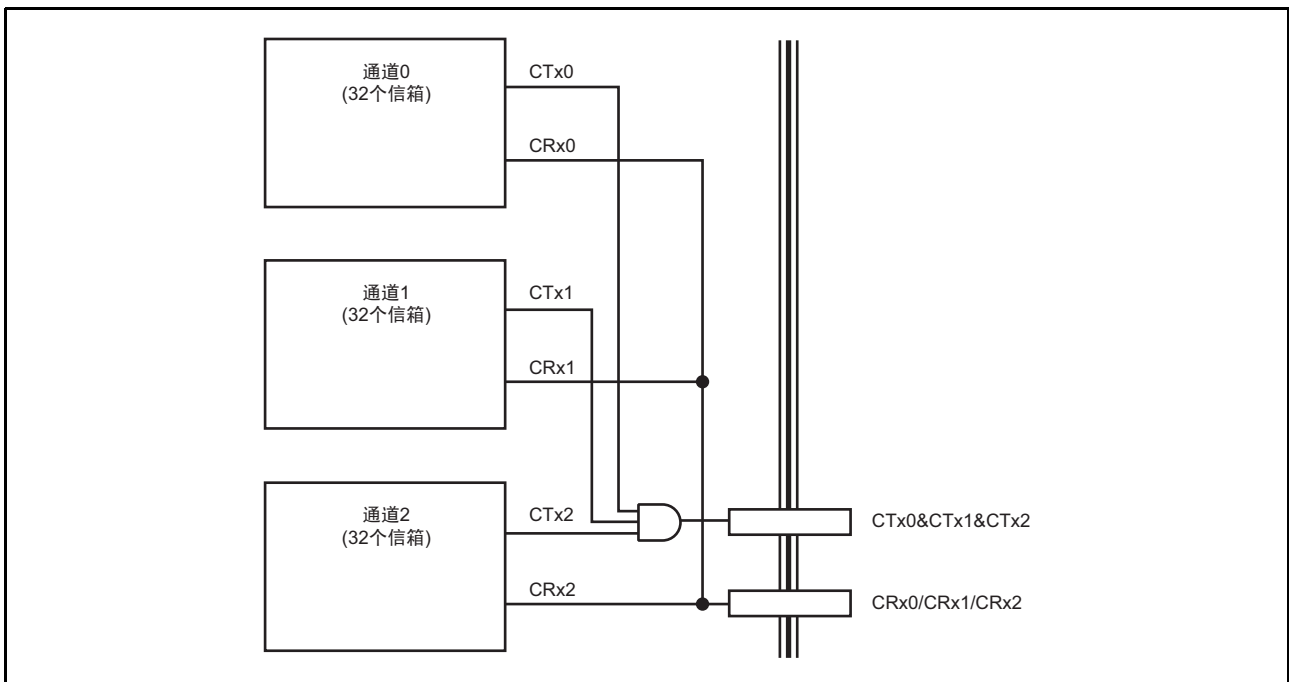


图 23.32 将通道 0、1、2 用作 96 个信箱的 1 个通道时的连接例子

23.12 使用时的注意事项

23.12.1 有关 1 个通道 64 个或者 96 个信箱的端口设定注意事项

本 LSI 内置的此模块有 3 个通道。在用于 1 个通道 64 个信箱的设定时，有以下注意事项：

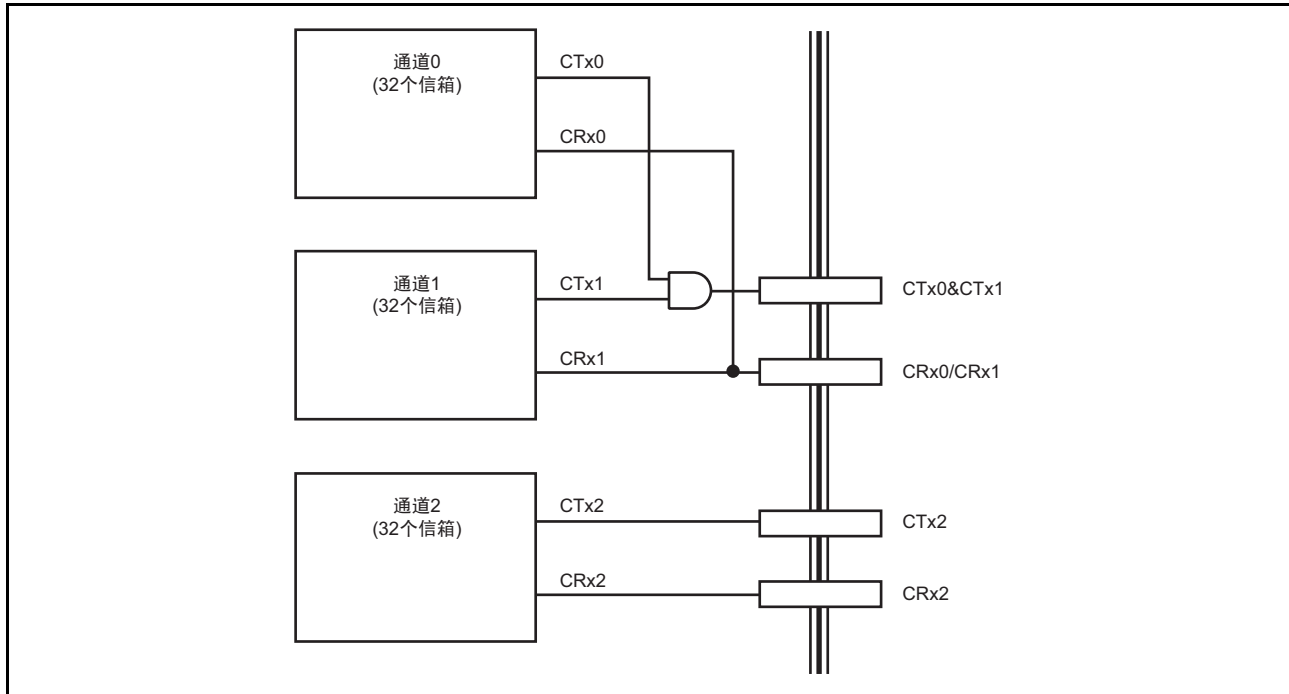


图 23.33 将通道 0、1 用作 64 个信箱的 1 个通道并且将通道 2 用作 32 个信箱的 1 个通道时的连接例子

1. 如果在 CAN 总线不连接其他节点的状态下发送信息，就不发生 ACK 错误。这是因为在从上图的通道 0 发送信息时，通道 1 在 ACK 字段发送 ACK。通道 1 接收 CAN 总线上的信息，然后根据 CAN 协议，在 ACK 字段发送 ACK，通道 0 接收此 ACK。对应的方法是必须将不发送信息的通道 1 置为测试模式的监听专用模式状态（TST[2:0] 为“B'001”）或者复位状态（将 MCR0 设定为“1”）。从而，不发送信息的通道就不发送 ACK。
2. 分别通过通道 0 和通道 1 单独进行决定发送顺序的内部仲裁。能发送的缓冲器有 31 个信箱/通道，但是在 62 个缓冲器的范围内不进行内部仲裁。
3. 不能给通道 0 和通道 1 设定相同的发送信息 ID。在 CAN 总线上进行仲裁后从 2 个通道发送信息。

24. IEBus™ 控制器

本 LSI 内置 1 个通道的 IEBus 控制器。IEBus™ (Inter Equipment Bus™) * 是以设备之间传送数据为目的的小规模数字数据传送系统。

本 LSI 没有内置 IEBus 驱动器 / 接收器，因此需要外接专用的驱动器 / 接收器。另外，因为 IERxD 引脚和 IETxD 引脚是 3V 的规格，所以需要外接专用的电平移位器。

【注】 * IEBus™ (Inter Equipment Bus™) 是瑞萨电子的商标。

24.1 特点

- 支持 IEBus 的协议控制 (层 2)。
 - 半双工异步通信
 - 多主控方式
 - 广播通信功能
 - 可选择 3 种不同传输速度的模式。
- 内置用于数据发送和接收的缓冲器。
 - 发送缓冲器和接收缓冲器各为 128 字节。
 - 能在模式 2 中连续进行最大传输字节数为 128 字节的发送和接收。
- 运行频率
 - 12MHz、12.58MHz (使用 AUDIO_X1* 和 AUDIO_X2* 的 1/2 分频时钟)
 - 18MHz、18.87MHz (使用 AUDIO_X1* 和 AUDIO_X2* 的 1/3 分频时钟)
 - 24MHz、25.16MHz (使用 P0φ 或者 AUDIO_X1* 和 AUDIO_X2* 的 1/4 分频时钟)
 - 30MHz、31.45MHz (使用 P0φ 或者 AUDIO_X1* 和 AUDIO_X2* 的 1/5 分频时钟)
 - 36MHz、37.74MHz (使用 AUDIO_X1* 和 AUDIO_X2* 的 1/6 分频时钟)
 - 42MHz、44.03MHz (使用 AUDIO_X1* 和 AUDIO_X2* 的 1/7 分频时钟)
 - 48MHz (使用 AUDIO_X1* 和 AUDIO_X2* 的 1/8 分频时钟)

【注】 * 只在不用作串行音频接口、带 FIFO 的时钟同步串行 I/O、瑞萨 SPDIF 接口的时钟输入时，才能用作此模块的时钟输入。

- 能设定模块待机模式

24.1.1 IEBus 通信协议

IEBus 的概要如下：

- 通信方式：半双工异步通信
- 多主控方式
能实现连接 IEBus 的全部单元到其他单元的数据传输。
- 广播通信功能（1 个单元对多个单元的通信）
组广播通信：对组单元的广播通信
同时广播通信：对全部单元的广播通信
- 可选择 3 种不同传输速度的模式。

表 24.1 3 种模式

模式	IEB ϕ *1=12MHz、18MHz、 24MHz*2、30MHz、36MHz、 42MHz、48MHz	IEB ϕ *1=12.58MHz、18.87MHz*2、 25.16MHz、31.45MHz、 37.74MHz、44.03MHz	最大传输字节数 (字节/帧)
0	约 3.9kbps	约 4.1kbps	16
1	约 17kbps	约 18kbps	32
2	约 26kbps	约 27kbps	128

【注】 *1 这是外围时钟 0 (P0 ϕ) 或者 AUDIO_X1 和 AUDIO_X2 的时钟。

*2 这是使用本 LSI 时的振荡频率。

- 存取控制：CSMA/CD (Carrier Sense Multiple Access with Collision Detection)
总线占有的优先级如下：
 1. 广播通信（1 个单元对多个单元的通信）优先于正常通信（1 个单元对 1 个单元的通信）。
 2. 主控地址小的优先。
- 通信规模
单元数：最大 50
电缆长：最长 150m（使用双绞线电缆的情况）

【注】 实际系统的通信规模因外接 IEBus 驱动器 / 接收器的特点和使用的电缆特点而不同。

(1) 总线占有权的决定（仲裁）

连接 IEBus 的设备为了在控制其他设备时占有总线而进行运行。该运行称为仲裁。在仲裁处理中，如果多个单元同时开始发送，就对其中的 1 个单元允许占有总线。

因为通过仲裁只能使 1 个设备获得总线占有权，所以决定了以下总线占有的优先条件。

(a) 由通信类型决定的优先

广播通信（1 个单元对多个单元的通信）优先于正常通信（1 个单元对 1 个单元的通信）。

(b) 由主控地址决定的优先

在通信类型相同时，主控地址最小的单元优先。

例：主控地址由 12 位构成，H'000 的单元拥有最高的优先级，而 H'FFF 的单元拥有最低的优先级。

【注】 如果仲裁失败，就能自动进入重新发送模式（重新发送次数能通过 IEMCR 的 RN 位设定为 0 ~ 7 次）。

(2) 通信模式

IEBus 有 3 种不同传输速度的通信模式。各通信模式的传输速度和 1 个通信帧中的最大传输字节数如表 24.2 所示。

表 24.2 各通信模式的传输速度和最大传输字节数

通信模式	最大传输字节数 (字节 / 帧)	实际传输速度 *1 (kbps)	
		IEB ϕ *2=12MHz、18MHz、 24MHz、30MHz、36MHz、 42MHz、48MHz*3	IEB ϕ *2=12.58MHz、18.87MHz、 25.16MHz、31.45MHz、 37.74MHz、44.03MHz*3
0	16	约 3.9	约 4.1
1	32	约 17	约 18
2	128	约 26	约 27

【注】 在通信前预先选择连接 IEBus 的各设备的通信模式。如果主控单元和其通信对象（从属单元）的通信模式不同，就不能进行正确的通信。

在 IEB ϕ 为 6MHz 的设备和 IEB ϕ 为 6.29MHz 的设备之间，即使通信模式相同也不能进行正确的通信。必须使用相同的振荡频率进行通信。

*1 这是传输最大传输字节数时的实际传输速度。

*2 这是外围时钟 0 (P0 ϕ) 或者 AUDIO_X1 和 AUDIO_X2 的时钟。

*3 这是使用本 LSI 时的振荡频率。

(3) 通信地址

IEBus 给各设备分配了 12 位固有的通信地址，通信地址的构成如下：

高 4 位：组号（各设备所属组的识别号）

低 8 位：单元号（组内各设备的识别号）

(4) 广播通信

在正常发送时，主控单元和其通信对象（从属单元）都是 1 个单元，能进行 1 对 1 的发送或者接收。在广播通信时，存在多个从属单元，主控单元对多个从属单元进行发送。由于从属单元有多个，因此在通信中从属单元不返回应答。

是进行广播通信还是进行正常通信取决于广播位（有关广播位，请参照“24.1.2(1)(b) 广播位”）。

有以下 2 种广播通信：

(a) 组广播通信

对通信地址高 4 位的组号相同的组内设备进行广播通信。

(b) 同时广播通信

与组号的值无关，对全部设备进行广播通信。

通过从属地址的值，识别是组广播还是同时广播（有关从属地址，请参照“24.1.2(3) 从属地址字段”）。

24.1.2 传输协议

IEBus 的传输信号格式如图 24.1 所示。

将通信数据作为被称为通信帧的一连串信号进行传输。1 个通信帧能传输的数据量和传输速度因通信模式而不同。

(IEB ϕ =12MHz、18MHz、24MHz、30MHz、36MHz时)																						
字段名	标头		主控地址 字段		从属地址 字段			控制 字段			信息长度 字段			数据 字段								
位数	1	1	12	1	12	1	1	4	1	1	8	1	1	8	1	1	8	1	1	8	1	1
	起始 位	广播 位	主控 地址	P	从属 地址	P	A	控制位	P	A	信息 长度位	P	A	数据位	P	A	数据位	P	A	数据位	P	A
传输时间																						
模式0	大约7330 μ s												大约1590 \times N μ s									
模式1	大约2090 μ s												大约410 \times N μ s									
模式2	大约1590 μ s												大约300 \times N μ s									
P: 奇偶校验位(1位) A: 应答位(1位) 当A=0时: ACK 当A=1时: NAK N: 数据字节数																						
【注】 在广播通信时, 忽视应答位的值。																						

图 24.1 传输信号格式

(1) 标头

标头由起始位和广播位构成。

(a) 起始位

起始位是向其他单元通知数据传输开始的信号。

要开始数据传输的单元输出规定时间的低电平信号（起始位），然后输出广播位。

要输出起始位时，如果其他单元已输出起始位，就不输出起始位而等待其他单元起始位的输出结束，然后与该结束时序同步输出广播位。

没有开始发送的单元检测此起始位，并且进入接收状态。

(b) 广播位

广播位用于识别广播通信和正常通信。

在广播位为“0”时表示广播通信；在为“1”时表示正常通信。广播通信有组广播和同时广播，根据从属地址的值进行识别（有关从属地址，请参照“24.1.2(3) 从属地址字段”）。

在广播通信时，因为有多多个通信对象（从属单元），所以不返回(2)以后各字段中的应答位。

当2个或者2个以上的单元以相同时序开始发送通信帧时，广播通信优先于正常通信而赢得仲裁。

(2) 主控地址字段

主控地址字段是用于将自己的单元地址（主控地址）发送到其他单元的字段，由主控地址位和奇偶校验位构成。

主控地址由 12 位构成，从 MSB 开始输出。

当 2 个和 2 个以上的单元以相同时序开始发送相同值的广播位时，仲裁的判定取决于主控地址字段。

在主控地址字段中每发送 1 位时，自己输出的数据和总线上的数据进行比较。如果自己输出的主控地址和总线上的数据不同，就判断为仲裁失败而中止发送，进入接收状态。

因为 IEBus 由线“与”构成，所以在加入仲裁的单元（仲裁主控）中最小主控地址的单元赢得仲裁。

最后，在输出 12 位的主控地址后只有 1 个单元作为主控单元保持发送状态。

然后，此主控单元输出奇偶校验位*，对其他单元确定主控地址，并且开始输出从属地址字段。

【注】* 奇偶校验使用偶校验，当主控地址位中的 1 的位数是奇数时，奇偶校验位为“1”。

(3) 从属地址字段

从属地址字段是用于发送要通信的单元（从属单元）地址（从属地址）的字段，由从属地址位、奇偶校验位和应答位构成。

从属地址由 12 位构成，从 MSB 开始输出。在发送 12 位从属地址后，为了避免错误接收从属地址，输出奇偶校验位。然后，为了确认从属单元是否在总线上，主控单元检测来自从属单元的应答信号，如果检测到应答信号，就输出控制字段。但是，在广播通信时，不检测应答位而直接输出控制字段。

如果从属地址相同并且检测到主控地址和从属地址的奇偶校验都是偶数，从属单元就输出应答信号。如果从属单元的奇偶校验是奇数，就认为没有正确地接收到主控地址或者从属地址而不输出应答信号。此时，主控单元进入待机（监视）状态并且结束通信。

在广播通信时，从属地址用于识别组广播和同时广播，具体内容如下：

从属地址是“H'FFF”：同时广播

从属地址不是“H'FFF”：组广播

【注】组广播通信时的组号为从属地址高 4 位的值。

(4) 控制字段

控制字段是用于发送以下数据字段的种类和方向的字段，由控制位、奇偶校验位和应答位构成。

控制位由 4 位构成，从 MSB 开始输出。

在控制位后，输出奇偶校验位。当奇偶校验为偶数并且从属单元能执行主控单元请求的功能时，从属单元输出应答信号，并且转移到下一个信息长度字段。但是，当奇偶校验为偶数而从属单元不能执行主控单元的请求时或者当奇偶校验是奇数时，从属单元不输出应答信号而返回到待机（监视）状态。

主控单元在确认应答信号后转移到下一个信息长度字段。

当不能确认应答信号时，主控单元进入待机状态并且结束通信。但是，在广播通信时，主控单元不确认应答信号而直接转移到下一个信息长度字段。

有关控制位的内容请参照表 24.4。

(5) 信息长度字段

信息长度字段是用于指定通信数据字节数的字段，由信息长度位、奇偶校验位和应答位构成。

信息长度位由 8 位构成，从 MSB 开始输出。通信数据的字节数如表 24.3 所示。

表 24.3 信息长度位的内容

信息长度位	发送数据和字节数
H'01	1 字节
H'02	2 字节
·	·
·	·
H'FF	255 字节
H'00	256 字节

【注】 根据通信模式，如果设定大于等于 1 帧的最大传输字节数，就为多帧通信。此时，第 2 次以后的信息长度位为剩余通信数据的字节数。在本 LSI 中，不能将信息长度位设定为大于等于 1 帧的最大传输字节数。必须在以下范围内进行设定：

模式 0：1～16 字节

模式 1：1～32 字节

模式 2：1～128 字节

在主控发送时（控制位的 bit3 为“1”）和在主控接收时（控制位的 bit3 为“0”），此字段的运行不同。

(a) 主控发送

主控单元输出信息长度位和奇偶校验位。如果从属单元检测到奇偶校验为偶数，就输出应答信号，转移到下一个数据字段。但是，在广播通信时，从属单元不输出应答信号。

当奇偶校验为奇数时，从属单元就认为没有正确地接收到信息长度位，不输出应答信号而返回到待机（监视）状态。此时，主控单元也返回到待机状态并且结束通信。

(b) 主控接收

从属单元输出信息长度位和奇偶校验位。如果主控单元检测到奇偶校验为偶数，就输出应答信号。

当奇偶校验为奇数时，主控单元就认为没有正确地接收到信息长度位，不输出应答信号而返回到待机状态。此时，从属单元也返回到待机状态并且结束通信。

(6) 数据字段

数据字段是用于对从属单元进行数据发送和数据接收的字段。主控单元使用数据字段给从属单元发送数据并且从从属单元接收数据。数据字段由数据位、奇偶校验位和应答位构成。

数据位由 8 位构成，从 MSB 开始输出。

在数据位后，分别从主控单元和从属单元输出奇偶校验位和应答位。

只在主控单元的发送运行时进行广播通信。此时，忽视应答信号。

主控发送和主控接收时的运行如下：

(a) 主控发送

当主控单元写从属单元时，主控单元将数据位和奇偶校验位发送到从属单元。从属单元接收数据位和奇偶校验位，如果奇偶校验为偶数并且接收缓冲器为空，就输出应答信号。如果奇偶校验为奇数或者接收缓冲器不为空，从属单元就拒绝接收对应的数据并且不输出应答信号。

如果从属单元不输出应答信号，主控单元就再次发送相同的数据。此运行持续到检测到从属单元的应答信号或者超过数据最大传输字节数为止。

当奇偶校验为偶数并且从属单元输出应答信号时，如果还有数据并且没有超过最大传输字节数，主控单元就发送下一个数据。

在广播通信时，从属单元不输出应答信号，主控单元按字节传送数据。

(b) 主控接收

当主控单元读从属单元时，主控单元输出与全部读取位对应的同步信号。

从属单元根据主控单元的同步信号，将数据和奇偶校验位的内容输出到总线。

主控单元读从属单元输出的数据和奇偶校验位，确认奇偶校验。

如果奇偶校验为奇数或者接收缓冲器不为空，主控单元就拒绝接收该数据并且不输出应答信号。如果 1 个通信帧中的数据量在能发送的最大传输字节数以内，主控单元就反复读相同的数据。如果奇偶校验为偶数并且接收缓冲器为空，主控单元就接受数据并且返回应答信号。如果 1 个通信帧中的数据量在能发送的最大传输字节数以内，主控单元就读取下一个数据。

(7) 奇偶校验位

奇偶校验位用于确认传输数据没有错误。

对主控地址位、从属地址位、控制位、信息长度位、数据位的各数据附加奇偶校验位。

奇偶校验为偶校验。当数据中的 1 的位数是奇数时，奇偶校验位为“1”；当数据中的 1 的位数是偶数时，奇偶校验位为“0”。

(8) 应答位

在正常通信（1 个单元对 1 个单元的通信）中，为了确认是否正确接收了数据，在以下位置附加应答位：

- 从属地址字段的最后
- 控制字段的最后
- 信息长度字段的最后
- 数据字段的最后

应答位的定义如下：

- 0：表示识别出传输数据（ACK）。
- 1：表示没有识别出传输数据（NAK）。

但是，在广播通信时，忽视应答位的内容。

(a) 从属地址字段最后的应答位

在以下情况下，从属地址字段最后的应答位为 NAK，并且中止传输。

- 当主控地址位或者从属地址位的奇偶校验不正确时
- 当发生时序错误（位格式有错）时
- 当不存在从属单元时

(b) 控制字段最后的应答位

在以下情况下，控制字段最后的应答位为 NAK，并且中止传输。

- 当控制位的奇偶校验不正确时
- 在从属接收缓冲器*不为空的情况下控制位的bit3为“1”（写）时
- 在从属发送缓冲器*为空的情况下控制位为读数据（H'3、H'7）时
- 在被锁定的情况下没被锁定的单元请求控制位的H'3、H'6、H'7、H'A、H'B、H'E、H'F时
- 在没被锁定的情况下控制位为读锁定地址（H'4、H'5）时
- 当发生时序错误时
- 当有未定义的控制位时

【注】 * 请参照“24.1.3(1) 从属状态（SSR）的读操作（控制位：H'0、H'6）”。

(c) 信息长度字段最后的应答位

在以下情况下，信息长度字段最后的应答位为 NAK，并且中止传输。

- 当信息长度位的奇偶校验不正确时
- 当发生时序错误时

(d) 数据字段最后的应答位

在以下情况下，数据字段最后的应答位为 NAK，并且中止传输。

- 当数据位的奇偶校验不正确时*
- 在传输前一个应答位后发生时序错误时
- 当接收缓冲器满而不能再接收数据时*

【注】* 在这种情况下，如果发送侧的 1 帧中的数据量在能传输的最大传输字节数以内，就在达到最大传输字节数前再次发送该数据字段。

24.1.3 传输数据（数据字段的内容）

数据字段的内容为控制位表示的数据。

表 24.4 控制位的内容

设定值	bit3*1	bit2	bit1	bit0	功能*2
H'0	0	0	0	0	读从属状态（SSR）。
H'1	0	0	0	1	未定义
H'2	0	0	1	0	未定义
H'3	0	0	1	1	读数据并且锁定。
H'4	0	1	0	0	读锁定地址（低 8 位）。
H'5	0	1	0	1	读锁定地址（高 4 位）。
H'6	0	1	1	0	读从属状态（SSR）并且解除锁定。
H'7	0	1	1	1	读数据。
H'8	1	0	0	0	未定义
H'9	1	0	0	1	未定义
H'A	1	0	1	0	写命令并且锁定。
H'B	1	0	1	1	写数据并且锁定。
H'C	1	1	0	0	未定义
H'D	1	1	0	1	未定义
H'E	1	1	1	0	写命令。
H'F	1	1	1	1	写数据。

【注】*1 以后的信息长度字段的信息长度位和数据字段的数据传送方向因 bit3（MSB）的值而不同。

当 bit3 为“1”时，进行主控单元到从属单元的传送。

当 bit3 为“0”时：进行从属单元到主控单元的传送。

*2 H'3、H'6、H'A、H'B 是指定设定锁定和解除锁定的控制位。

当发送 H'1、H'2、H'8、H'9、H'C、H'D 的未定义值时，不返回应答。

如果在表 24.5 中没有从不请求锁定的主控单元接收到的控制位，被主控单元锁定的单元就拒绝接收此控制位，并且不输出应答位。

表 24.5 被锁定的从属单元的控制字段

设定值	bit3	bit2	bit1	bit0	功能
H'0	0	0	0	0	读从属状态。
H'4	0	1	0	0	读锁定地址（低 8 位）。
H'5	0	1	0	1	读锁定地址（高 4 位）。

(1) 从属状态（SSR）的读操作（控制位：H'0、H'6）

通过读从属状态（H'0，H'6），主控单元能得知从属单元没有返回应答位（ACK）的理由。从属单元最后进行的通信结果决定从属状态，全部从属单元都能提供从属状态的信息。从属状态如图 24.2 所示。

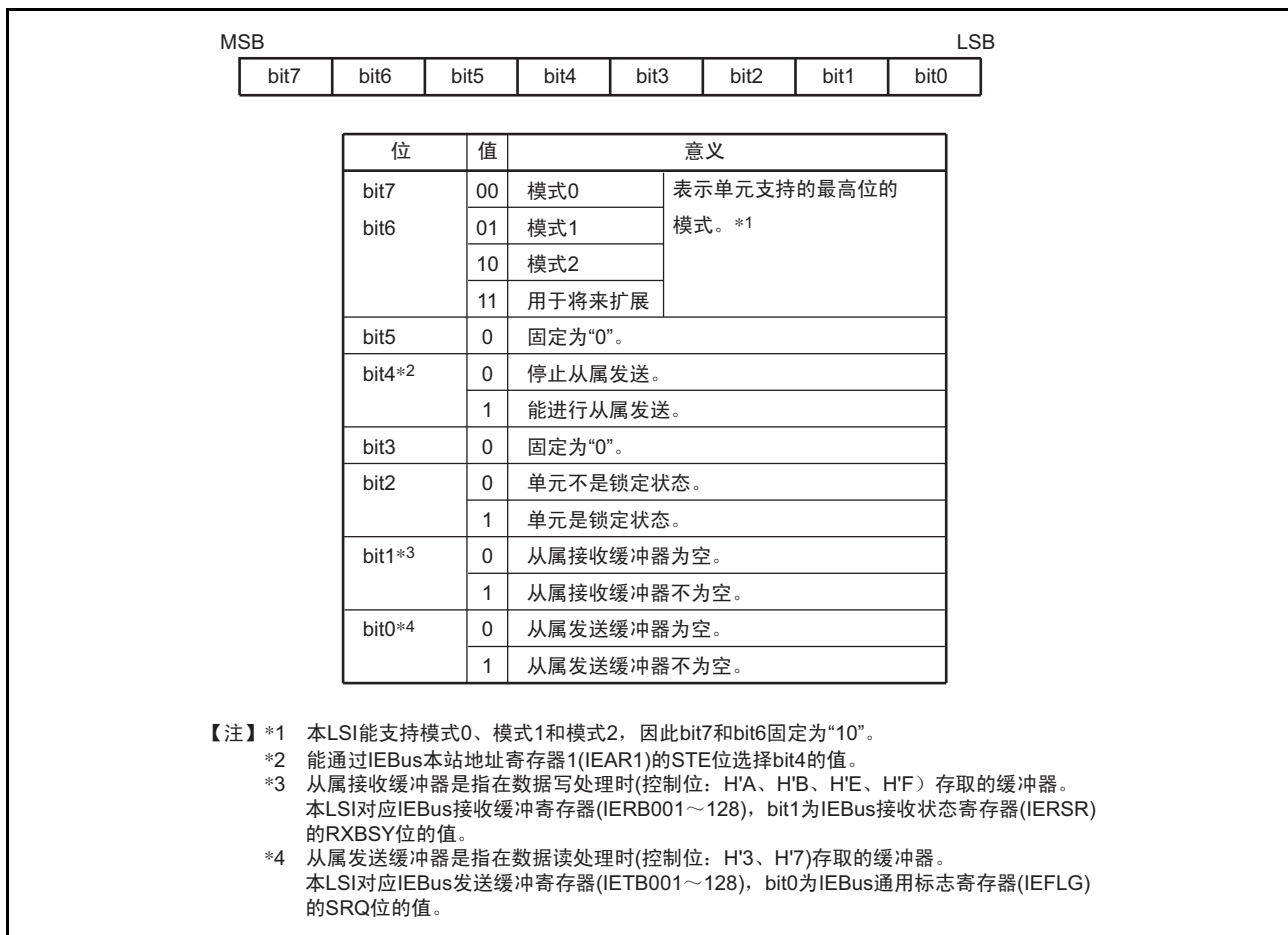


图 24.2 从属状态（SSR）的位结构

(2) 数据命令的传送（控制位：读（H'3、H'7）、写（H'A、H'B、H'E、H'F））

如果读数据（H'3、H'7），就将从属单元数据缓冲器中的数据读到主控单元。如果写数据（H'B、H'F）或者写命令（H'A、H'E），就根据该从属单元的运行规定处理从属单元接收的数据。

- 【注】** 1. 用户能根据系统自由选择数据和命令。
2. H'3、H'A、H'B 可能因通信条件和状态而被锁定。

(3) 锁定地址的读操作（控制位：H'4、H'5）

在对锁定地址进行读处理（H'4、H'5）时，发行锁定指令的主控单元的地址（12 位）如下所示，由 1 字节单位构成并且被读取。

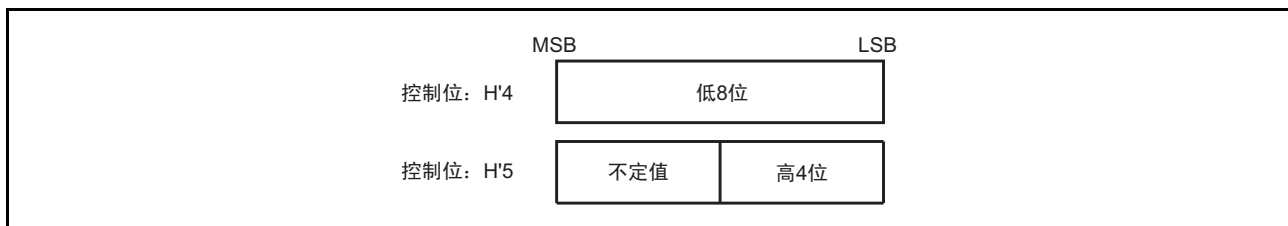


图 24.3 锁定地址的结构

(4) 锁定的设定和解除（控制位：设定（H'3、H'A、H'B）、解除（H'6））

锁定功能用于通过多个通信帧传送信息。锁定单元不接收没被锁定的单元的数据。锁定的设定和解除如下所示。

(a) 锁定的设定

通过指定锁定的控制位（H'3、H'A、H'B），在结束信息长度字段的应答位 0 的发送和接收后，当没有结束信息长度位指定字节数的数据发送和接收而结束通信帧时，主控单元锁定从属单元，并且同时将从属状态字节中的锁定相关位（bit2）置“1”。

只在超过帧的最大传输字节数时才设定锁定，在其他错误结束时不设定锁定。

(b) 锁定的解除

通过指定锁定的控制位（H'3、H'A、H'B）或者指定锁定解除的控制位（H'6），在 1 个通信帧内结束信息长度位指定字节数的数据发送和接收后，主控单元解除锁定的从属单元，并且同时将从属状态字节中的锁定相关位（bit2）复位为“0”。

在广播通信中，不进行锁定的设定和解除。

【注】 要使指定的单元自身解除锁定时，有以下 3 种方法：

- 进行上电复位。
- 进入深度待机模式。
- 通过 IEBus 命令寄存器（IECMR）发行锁定状态解除命令。

另外，能通过 IEFLG 的 LCK 位确认是否为锁定状态。

24.1.4 位格式

构成 IEBus 通信帧的位格式（概念）如图 24.4 所示。

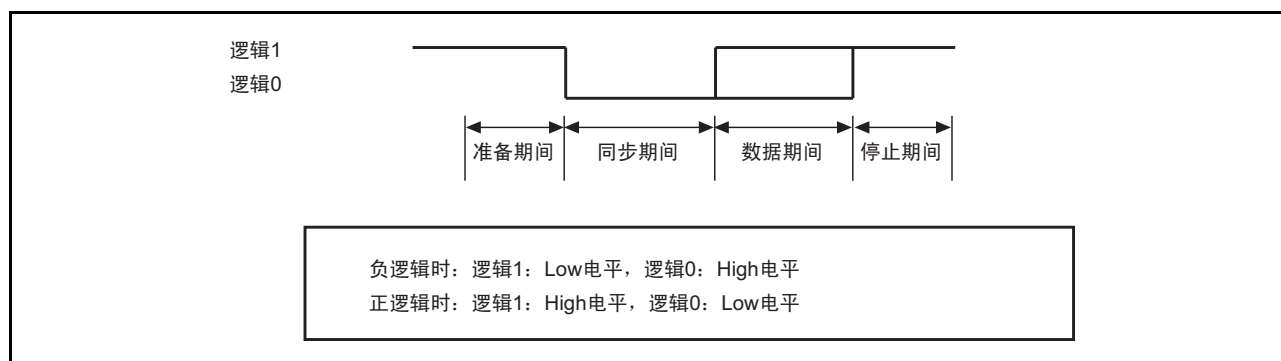


图 24.4 IEBus 的位格式（概念）

以下说明在正逻辑时位格式的各个期间。

准备期间: 最初的逻辑 1 期间 (High 电平)

同步期间: 下一个的逻辑 0 期间 (Low 电平)

数据期间: 表示位值的期间 (逻辑 1: High 电平、逻辑 0: Low 电平)

停止期间: 最后的逻辑 1 期间 (High 电平)

在负逻辑时, 为正逻辑的反相电平。

同步期间和数据期间的长度几乎相等。

IEBus 按位取得同步。有关位的整体时间规格和分配在该位内的期间时间规格, 因传输位的种类和单元种类 (主控单元或者从属单元) 而不同。

24.1.5 结构

整体框图如图 24.5 所示，各块的功能如表 24.6 所示。

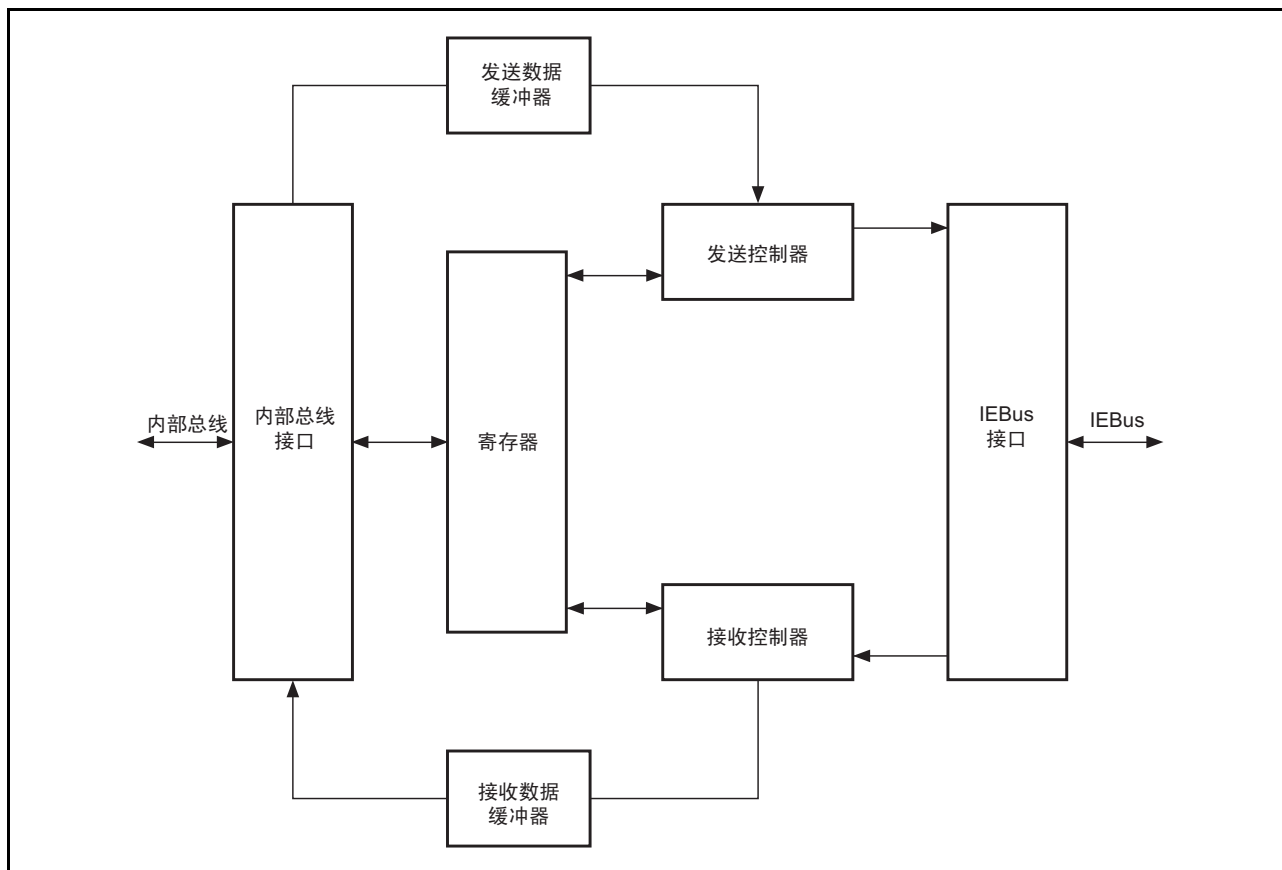


图 24.5 框图

表 24.6 各块的功能

块	功能
内部总线接口	内部总线的接口功能
	<ul style="list-style-type: none"> • 数据宽度为 8 位。 • 寄存器存取
IEBus 接口	符合 IEBus 规格的接口功能
	<ul style="list-style-type: none"> • 按照 IEBus 规格的位格式，从发送控制器将数据发送到 IEBus。 • 从 IEBus 规格的位格式抽出帧数据并且传送到接收控制器。
寄存器	控制寄存器
	<ul style="list-style-type: none"> • 用于控制此模块的寄存器 • 能通过内部总线进行读写。
发送控制器	将缓冲器的数据发送到 IEBus。
	<ul style="list-style-type: none"> • 从寄存器的标头信息和发送缓冲器的数据生成发送帧，并且发送此帧。 • 检测发送错误。
接收控制器	将 IEBus 的数据保存到接收缓冲器。
	<ul style="list-style-type: none"> • 将接收的帧标头信息保存到寄存器，将接收的帧数据保存到接收缓冲器。 • 检测接收错误。
发送数据缓冲器	用于发送数据的缓冲器
	<ul style="list-style-type: none"> • 此缓冲器保存发送给 IEBus 的数据。 • 缓冲器容量为 128 字节。
接收数据缓冲器	用于接收数据的缓冲器
	<ul style="list-style-type: none"> • 此缓冲器保存从 IEBus 接收的数据。 • 缓冲器容量为 128 字节。

24.2 输入 / 输出引脚

表 24.7 引脚结构

名称	引脚名	输入 / 输出	功能
IEBus 接收数据引脚	IERxD	输入	接收数据的输入引脚
IEBus 发送数据引脚	IETxD	输出	发送数据的输出引脚

24.3 寄存器说明

寄存器结构如表 24.8 所示。

表 24.8 寄存器结构

寄存器名	略称	R/W	初始值	地址	存取长度
IEBus 控制寄存器	IECTR	R/W	H'00	H'FFFEF000	8
IEBus 命令寄存器	IECMR	W	H'00	H'FFFEF001	8
IEBus 主控控制寄存器	IEMCR	R/W	H'00	H'FFFEF002	8
IEBus 本站地址寄存器 1	IEAR1	R/W	H'00	H'FFFEF003	8
IEBus 本站地址寄存器 2	IEAR2	R/W	H'00	H'FFFEF004	8
IEBus 从属地址设定寄存器 1	IESA1	R/W	H'00	H'FFFEF005	8
IEBus 从属地址设定寄存器 2	IESA2	R/W	H'00	H'FFFEF006	8
IEBus 发送信息长度寄存器	IETBFL	R/W	H'00	H'FFFEF007	8
IEBus 接收主控地址寄存器 1	IEMA1	R	H'00	H'FFFEF009	8
IEBus 接收主控地址寄存器 2	IEMA2	R	H'00	H'FFFEF00A	8
IEBus 接收控制字段寄存器	IERCTL	R	H'00	H'FFFEF00B	8
IEBus 接收信息长度寄存器	IERBFL	R	H'00	H'FFFEF00C	8
IEBus 锁定地址寄存器 1	IELA1	R	H'00	H'FFFEF00E	8
IEBus 锁定地址寄存器 2	IELA2	R	H'00	H'FFFEF00F	8
IEBus 通用标志寄存器	IEFLG	R	H'00	H'FFFEF010	8
IEBus 发送状态寄存器	IETSR	R/(W)*	H'00	H'FFFEF011	8
IEBus 发送中断允许寄存器	IEIET	R/W	H'00	H'FFFEF012	8
IEBus 接收状态寄存器	IERSR	R/(W)*	H'00	H'FFFEF014	8
IEBus 接收中断允许寄存器	IEIER	R/W	H'00	H'FFFEF015	8
IEBus 时钟选择寄存器	IECKSR	R/W	H'01	H'FFFEF018	8
IEBus 发送数据缓冲器 001 ~ 128	IETB001 ~ IETB128	W	不定值	H'FFFEF100 ~ H'FFFEF17F	8
IEBus 接收数据缓冲器 001 ~ 128	IERB001 ~ IERB128	R	不定值	H'FFFEF200 ~ H'FFFEF27F	8

【注】 * 为了清除标志，只能写“1”。

24.3.1 IEBus 控制寄存器 (IECTR)

IECTR 是设定此模块运行控制的寄存器。

位:	7	6	5	4	3	2	1	0
	—	IOL	DEE	—	RE	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R	R/W	R	R	R

位	位名	初始值	R/W	说明
7	—	0	R	保留位 读写值都为“0”。
6	IOL	0	R/W	输入 / 输出电平 选择 IERxD 引脚和 IETxD 引脚的输入 / 输出电平（正逻辑、负逻辑）。 0: 引脚的输入 / 输出为负逻辑 (逻辑 1 为 Low 电平, 逻辑 0 为 High 电平) 1: 引脚的输入 / 输出为正逻辑 (逻辑 1 为 High 电平, 逻辑 0 为 Low 电平)
5	DEE	0	R/W	广播接收错误中断允许 如果将此位置“1”，就在广播接收时并且接收缓冲器为不能接收的状态（RE 位没有被置“1”或者 RXBSY 标志被置位的状态）下产生广播接收错误中断。此时，保存 IEBus 接收主控地址寄存器 1、2。 如果此位为“0”，就在广播接收时并且接收缓冲器为不能接收的状态下不产生广播接收错误中断而中止接收，进入待机状态。此时，不保存主控地址。 0: 控制字段为止的数据没有发生广播接收错误。 1: 控制字段为止的数据发生广播接收错误。
4	—	0	R	保留位 读写值都为“0”。
3	RE	0	R/W	接收允许 允许或者禁止接收。必须在接收帧前的初始设定时设定此位。 0: 禁止接收 1: 允许接收
2~0	—	全 0	R	保留位 读写值都为“0”。

24.3.2 IEBus 命令寄存器 (IECMR)

IECMR 发行控制通信的命令。此寄存器是只写寄存器，读取值为不定值。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	CMD		
初始值:	0	0	0	0	0	0	0	0
R/W:	—	—	—	—	—	W	W	W

位	位名	初始值	R/W	说明
7 ~ 3	—	全 0	—	保留位 读取值为不定值，写入值总是“0”。
2 ~ 0	CMD	000	W	命令位 发送控制通信的命令。在发行命令后，IEFLG 的 CMX 位被置位的期间为执行命令的期间。在 CMX 变为“0”后，转移到运行状态。 000: 空操作（不影响运行） 001: 解除锁定状态（其他单元的请求）*1 010: 作为主控单元，请求通信。 011: 中止主控通信 *2 100: 未定义 *4 101: 请求从属单元发送数据 110: 中止从属单元发送数据 *3 111: 未定义 *4

【注】 *1 在从属通信状态下禁止执行。

*2 只在主控通信（MRQ 位为“1”）时，此命令有效。否则，即使发行该命令也被忽视。如果在主控通信中发行此命令，通信控制器就立即进入待机状态，并且同时结束发行的主控发送请求（MRQ 位为“0”）。

*3 只在从属发送（SRQ 位为“1”）时，此命令有效。否则，即使发行该命令也被忽视。如果在从属发送中发行此命令，就在进行从属发送前 SRQ 位为“0”，因此不响应主控单元的发送请求。如果在进行从属发送的状态下发行命令，就中止发送运行，进入待机状态（SRQ 位为“0”）。

*4 这是未定义位。即使发行此命令也不影响运行。

24.3.3 IEBus 主控控制寄存器 (IEMCR)

IEMCR 设定主控通信时的通信条件。

位:	7	6	5	4	3	2	1	0
	SS	RN			CTL*1			
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	SS	0	R/W	广播 / 正常通信的选择 选择主控通信时的广播通信或者正常通信。 0: 在主控通信时进行广播通信 1: 在主控通信时进行正常通信
6 ~ 4	RN	000	R/W	重新发送次数 如果在主控通信中仲裁失败, 就自动设定重新发送次数。如果仲裁失败, 就将 IETSR 的 TXEAL 位置位, 并且发送错误结束。 000: 0 次 001: 1 次 010: 2 次 011: 3 次 100: 4 次 101: 5 次 110: 6 次 111: 7 次
3 ~ 0	CTL*1	0000	R/W	控制位 设定主控发送时的控制字段的控制位。 0000: 读从属状态 0001: 未定义 *3 0010: 未定义 *3 0011: 读并且锁定数据 *2 0100: 读锁定地址 (低 8 位) 0101: 读锁定地址 (高 4 位) 0110: 读从属状态并且解除锁定 *2 0111: 读数据 1000: 未定义 *3 1001: 未定义 *3 1010: 写并且锁定命令 *2 1011: 写并且锁定数据 *2 1100: 未定义 *3 1101: 未定义 *3 1110: 写命令 1111: 写数据

【注】 *1 以后的信息长度字段的信息长度位和数据字段的数据传送方向因 bit3 的值而不同。

当 bit3 为 “1” 时, 从主控单元传送到从属单元。

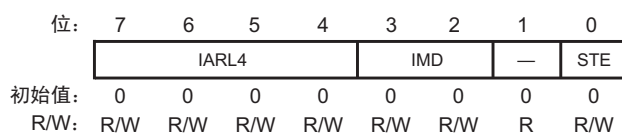
当 bit3 为 “0” 时, 从从属单元传送到主控单元。

*2 这是指定锁定的设定和解除的控制位。

*3 禁止使用。

24.3.4 IEBus 本站地址寄存器 1 (IEAR1)

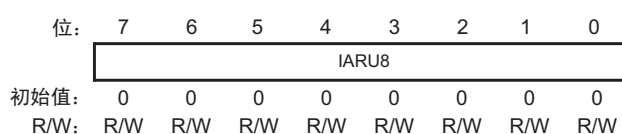
IEAR1 设定本站地址的低 4 位和通信模式。在主导通信时，本站地址为主导地址字段的值；在从属通信时，本站地址和接收的从属地址字段进行比较。



位	位名	初始值	R/W	说明
7 ~ 4	IARL4	0000	R/W	IEBus 本站地址的低 4 位 设定本站地址的低 4 位，为主导地址字段的值。在从属通信时，和接收的从属地址进行比较。
3、2	IMD	00	R/W	IEBus 通信模式 选择 IEBus 通信模式。 00: 通信模式 0 01: 通信模式 1 10: 通信模式 2 11: 禁止设定
1	—	0	R	保留位 读写值都为“0”。
0	STE	0	R/W	从属发送设定 设定从属状态寄存器 bit4 的值。如果将此位置“1”，就通过发送从属状态寄存器，将能进行从属发送的状态信息通知主控单元。此位只设定从属状态寄存器的值而不直接影响从属发送的运行。 0: 从属状态的 bit4 为“0”（为停止从属发送的状态） 1: 从属状态的 bit4 为“1”（为能进行从属发送的状态）

24.3.5 IEBus 本站地址寄存器 2 (IEAR2)

IEAR2 设定本站地址的高 8 位。在主导通信时，此寄存器为主导地址字段的值；在从属通信时，此寄存器和接收的从属地址字段进行比较。



位	位名	初始值	R/W	说明
7 ~ 0	IARU8	全 0	R/W	IEBus 本站地址的高 8 位 设定本站地址的高 8 位，为主导地址字段的值。在从属通信时，和接收的从属地址进行比较。

24.3.6 IEBus 从属地址设定寄存器 1 (IESA1)

IESA1 设定通信对象从属单元地址的低 4 位。

位:	7	6	5	4	3	2	1	0
	ISAL4				—	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 4	ISAL4	0000	R/W	IEBus 从属地址的低 4 位 设定通信对象从属单元地址的低 4 位。
3 ~ 0	—	全 0	R	保留位 读写值都为“0”。

24.3.7 IEBus 从属地址设定寄存器 2 (IESA2)

IESA2 设定通信对象从属单元地址的高 8 位。

位:	7	6	5	4	3	2	1	0
	ISAU8							
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 0	ISAU8	全 0	R/W	IEBus 从属地址的高 8 位 设定通信对象从属单元地址的高 8 位。

24.3.8 IEBus 发送信息长度寄存器 (IETBFL)

IETBFL 设定主控发送和从属发送时的信息长度。

位:	7	6	5	4	3	2	1	0
	IBFL							
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 0	IBFL	全 0	R/W	发送信息长度位 设定主控发送时的信息长度的值。必须设定通信模式的最大传输字节数以内的值。 H'01: 1 字节 H'02: 2 字节 : H'7F: 127 字节 H'80: 128 字节 H'81: 未定义 * : H'FF: 未定义 * H'00: 未定义 *

【注】 * 禁止设定。

24.3.9 IEBus 接收主控地址寄存器 1 (IEMA1)

IEMA1 表示从属广播接收时的通信对象主控单元地址的低 4 位。

位:	7	6	5	4	3	2	1	0
	IMAL4				—	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 4	IMAL4	0000	R	IEBus 接收主控地址的低 4 位 表示从属广播接收时的通信对象主控单元地址的低 4 位。当开始从属广播接收时，此寄存器有效（在 RXS 位被置位时改写内容）。 如果通过 IECTR 的 DEE 位选择广播接收错误，就在接收控制字段时接收缓冲器变为不能接收信息的状态下产生接收错误中断，并且将主控地址的低 4 位保存到 IEMA1。
3 ~ 0	—	全 0	R	保留位 读写值都为“0”。

24.3.10 IEBus 接收主控地址寄存器 2 (IEMA2)

IEMA2 读从属广播接收时的通信对象主控单元地址的高 8 位。当开始从属广播接收时，此寄存器有效（在 IERSR 的 RXS 位被置位时改写内容）。

如果通过 IECTR 的 DEE 位选择广播接收错误中断，就在接收控制字段时接收缓冲器变为不能接收信息的状态下产生接收错误中断，并且将主控地址的高 8 位保存到 IEMA2。此寄存器的写操作无效。

位:	7	6	5	4	3	2	1	0
	IMAU8							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 0	IMAU8	全 0	R	IEBus 接收主控地址的高 8 位 表示从属广播接收时的通信对象主控单元地址的高 8 位。当开始从属广播接收时，此寄存器有效（在 RXS 位被置位时改写内容）。 如果通过 IECTR 的 DEE 位选择广播接收错误，就在接收控制字段时接收缓冲器变为不能接收信息的状态下产生接收错误中断，并且将主控地址的高 8 位保存到 IEMA2。

24.3.11 IEBus 接收控制字段寄存器 (IERCTL)

IERCTL 读从属广播接收时的控制字段的值。当开始从属广播接收时，此寄存器有效（在 IERSR 的 RXS 位被置位时改写内容）。

此寄存器的写操作无效。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	RCTL			
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 4	—	全 0	R	保留位 读写值都为“0”。
3 ~ 0	RCTL	0000	R	IEBus 接收控制字段 表示从属广播接收时的控制字段的值。当开始从属广播接收时，此寄存器有效（在 RXS 位被置位时改写内容）。

24.3.12 IEBus 接收信息长度寄存器 (IERBFL)

IERBFL 读从属广播接收时的信息长度字段。当开始从属广播接收时，此寄存器有效（在 IERSR 的 RXS 位被置位时改写内容）。

此寄存器的写操作无效。

位:	7	6	5	4	3	2	1	0
	RBFL							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 0	RBFL	全 0	R	IEBus 接收信息长度 读从属广播接收时的信息长度字段的内容。

24.3.13 IEBus 锁定地址寄存器 1 (IELA1)

IELA1 表示锁定时的锁定地址的低 8 位。

位:	7	6	5	4	3	2	1	0
	ILAL8							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 0	ILAL8	全 0	R	IEBus 锁定地址的低 8 位 表示被锁定的主控单元地址的低 8 位。只有在 IEFLG 的 LCK 位被置位时才有效。

24.3.14 IEBus 锁定地址寄存器 2 (IELA2)

IELA2 表示锁定时的锁定地址的高 4 位。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	ILAU4			
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 4	—	全 0	R	保留位 读写的值总是 0。
3 ~ 0	ILAU4	0000	R	IEBus 锁定地址的高 4 位 保存被锁定的主控单元地址的高 4 位。只有在 IEFLG 的 LCK 位被置位时才有效。

24.3.15 IEBus 通用标志寄存器 (IEFLG)

IEFLG 表示此模块的命令执行状态、锁定状态、从属地址匹配、广播接收的检测。

位:	7	6	5	4	3	2	1	0
	CMX	MRQ	SRQ	SRE	LCK	—	RSS	GG
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7	CMX	0	R	命令执行状态 表示命令的执行状态。 0: 命令执行结束 1: 正在执行命令 [置位条件] • 在 MRQ、SRQ、SRE 任意一位被置位的条件下发行主控通信请求命令 或者从属发送请求命令时 [清除条件] • 当命令执行结束时
6	MRQ	0	R	主控通信请求 表示主控单元是否处于通信请求期间。 0: 主控单元不处于通信请求期间 1: 主控单元处于通信请求期间 [置位条件] • 当发行主控通信请求命令并且 CMX 位变为“0”时。 [清除条件] • 当主控通信结束时

位	位名	初始值	R/W	说明
5	SRQ	0	R	<p>从属发送请求 表示从属单元是否处于发送请求期间。</p> <p>0: 从属单元不处于发送请求期间 1: 从属单元处于发送请求期间</p> <p>[置位条件]</p> <ul style="list-style-type: none"> 当发行从属发送请求命令并且 CMX 位变为“0”时。 <p>[清除条件]</p> <ul style="list-style-type: none"> 当从属发送结束时
4	SRE	0	R	<p>从属接收状态 表示从属广播接收的执行状态。</p> <p>0: 不在进行从属广播接收 1: 正在进行从属广播接收</p> <p>[置位条件]</p> <ul style="list-style-type: none"> 在 IECTR 的 RE 位为“1”的状态下开始从属广播接收时 <p>[清除条件]</p> <ul style="list-style-type: none"> 当从属广播接收结束时
3	LCK	0	R	<p>锁定状态表示 如果从主控单元接收到锁定请求并且设定锁定, 就将此位置位。在此位被置位时, IELA1 和 IELA2 的值有效。</p> <p>0: 解除锁定 1: 锁定状态</p> <p>[置位条件]</p> <ul style="list-style-type: none"> 当从主控单元接收到设定锁定的控制位并且没有接收到信息长度指定的数据时 (LCK 位只在超过帧最大传输字节数时被置位, 而在其他错误结束时不被置位)。 <p>[清除条件]</p> <ul style="list-style-type: none"> 当锁定解除条件成立或者发行锁定解除命令时。
2	—	0	R	<p>保留位 读写值都为“0”。</p>
1	RSS	0	R	<p>接收广播位 表示接收的广播位的值。当开始从属广播接收时, 此位有效 (在 RXS 位被置位时改写内容)。 将上一次的值保持到开始从属广播接收为止。</p> <p>0: 接收的广播位为“0” 1: 接收的广播位为“1”</p>
0	GG	0	R	<p>同时广播接收识别 在广播接收时, 如果识别到从属地址为“H'FFF”, 就将此位置位。和接收广播位相同, 当开始从属广播接收时, 此位有效 (在 RXS 位被置位时改写内容)。 将上一次的值保持到开始从属广播接收为止。在从属正常接收时为“0”。</p> <p>0: (1) 为从属接收 (2) 在广播接收时, 在从属地址字段中没有识别到“H'FFF”。 1: 在广播接收时, 在从属地址字段中识别到“H'FFF”。</p>

24.3.16 IEBus 发送状态寄存器 (IETSR)

IETSR 检测发送开始、发送正常结束、发送错误结束等的状态。

各状态源有对应 IEBus 发送中断允许寄存器 (IEIET) 的位，能设定禁止或者允许中断。通过给各位写“1”来清除该位。

位:	7	6	5	4	3	2	1	0
	—	TXS	TXF	—	TXEAL	TXETTME	TXERO	TXEACK
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R/(W)*	R/(W)*	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*

位	位名	初始值	R/W	说明
7	—	0	—	保留位 读写值都为“0”。
6	TXS	0	R/(W)*	发送开始 表示此模块开始发送。 [置位条件] • 在主控发送的情况下赢得仲裁并且标头到主控地址字段的数据发送结束时。 [清除条件] • 当写“1”时
5	TXF	0	R/(W)*	发送正常结束 检测到信息长度位所定数据量的发送正常结束。 [置位条件] • 当信息长度位所定数据量的发送结束时 [清除条件] • 当写“1”时
4	—	0	R	保留位 读写值都为“0”。
3	TXEAL	0	R/(W)*	仲裁失败 如果在主控通信时仲裁失败，此模块就重新从起始位开始发送，发送次数为 IEMCR 的 RN 位的设定次数。如果设定次数的仲裁全部失败，就将此位置位并且进入待机状态。如果在进行设定次数的重新发送时赢得仲裁，就不将此位置“1”。在仲裁失败并且通信变为待机状态时将此位置位。 [置位条件] • 表示在数据发送过程中发生仲裁失败而发送结束。 [清除条件] • 当写“1”时
2	TXETTME	0	R/(W)*	发送时序错误 在数据发送过程中没有按 IEBus 协议规定的时序传送数据时，将此位置位。此模块在将此位置位后进入待机状态。 [置位条件] • 表示在数据发送过程中发生时序错误。 [清除条件] • 当写“1”时

位	位名	初始值	R/W	说明
1	TXERO	0	R/(W)*	<p>发送帧最大传输字节数超过</p> <p>表示在发送数据时发生以下某种情况：</p> <ul style="list-style-type: none"> • 因从接收单元接收到 NAK 而重新发送，进行了通信模式所定最大字节长度的发送。 • 因信息长度的值大于最大传输量的值而发送没有结束。 <p>此模块在将此位置位后进入待机状态。</p> <p>[置位条件]</p> <ul style="list-style-type: none"> • 表示进行了通信模式所定最大字节数的发送而发送没有结束。 <p>[清除条件]</p> <ul style="list-style-type: none"> • 当写“1”时
0	TXEACK	0	R/(W)*	<p>应答位</p> <p>表示数据字段应答位接收的数据。</p> <ul style="list-style-type: none"> • 数据字段以外的应答位 如果接收到 NAK，就中止发送并且进入待机状态，并且将此位置“1”。 • 数据字段的应答位 如果在发送数据字段时从接收单元接收到 NAK，就进行重新发送，直到从接收单元接收到 ACK 或者达到通信模式定义的最大字节数为止。此时，如果在发送过程中从接收单元接收到 ACK，就不设定此位而继续发送。如果没有接收到 ACK 而结束通信，就将此位置“1”。 <p>【注】 此标志在广播通信时无效。</p> <p>[置位条件]</p> <ul style="list-style-type: none"> • 表示检测到应答位为“1”（NAK）。 <p>[清除条件]</p> <ul style="list-style-type: none"> • 当写“1”时

【注】 * 为了清除标志，只能写“1”。

24.3.17 IEBus 发送中断允许寄存器 (IEIET)

IEIET 能禁止或者允许 IETSR 的发送开始、发送正常结束、发送错误结束等的各中断源的中断。

位:	7	6	5	4	3	2	1	0
	—	TXSE	TXFE	—	TXEALE	TXE TTMEE	TXEROE	TXE ACKE
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	—	0	R	保留位 读写值都为“0”。
6	TXSE	0	R/W	发送开始中断允许 禁止或者允许发送开始 (TXS) 中断。 0: 禁止发送开始 (TXS) 中断 1: 允许发送开始 (TXS) 中断
5	TXFE	0	R/W	发送正常结束中断允许 禁止或者允许发送正常结束 (TXF) 中断。 0: 禁止发送正常结束 (TXF) 中断 1: 允许发送正常结束 (TXF) 中断
4	—	0	R	保留位 读写值都为“0”。
3	TXEALE	0	R/W	仲裁失败中断允许 禁止或者允许仲裁失败 (TXEAL) 中断。 0: 禁止仲裁失败 (TXEAL) 中断 1: 允许仲裁失败 (TXEAL) 中断
2	TXETTMEE	0	R/W	发送时序错误中断允许 禁止或者允许发送时序错误 (TXETTMEE) 中断。 0: 禁止发送时序错误 (TXETTMEE) 中断 1: 允许发送时序错误 (TXETTMEE) 中断
1	TXEROE	0	R/W	发送帧最大传输字节数超过中断允许 禁止或者允许发送帧最大传输字节数超过 (TXEROE) 中断。 0: 禁止发送帧最大传输字节数超过 (TXEROE) 中断 1: 允许发送帧最大传输字节数超过 (TXEROE) 中断
0	TXEACKE	0	R/W	应答位中断允许 禁止或者允许应答位 (TXEACKE) 中断。 0: 禁止应答位 (TXEACKE) 中断 1: 允许应答位 (TXEACKE) 中断

24.3.18 IEBus 接收状态寄存器 (IERSR)

IERSR 检测接收忙、接收开始、接收正常结束、接收错误结束等的状态。各状态源有对应 IEIER 的位，能禁止或者允许中断。能通过给各位写“1”来清除该位。

位:	7	6	5	4	3	2	1	0
	RXBSY	RXS	RXF	RXEDE	RXEOVE	RXE RTME	RXEDLE	RXEPE
初始值:	0	0	0	0	0	0	0	0
R/W:	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

位	位名	初始值	R/W	说明
7	RXBSY	0	R/(W)*	接收忙 表示将接收数据保存到接收数据缓冲器 (IERB001 ~ IERB128)。必须在读全部接收数据后清除此位。在此位被置位的期间，不能接收下一个接收数据。 [置位条件] • 当接收数据全部被写到接收数据缓冲器时 [清除条件] • 当写“1”时
6	RXS	0	R/(W)*	接收开始 表示此模块开始接收。 [置位条件] • 在从属接收的情况下从主控单元正常接收了标头到信息长度字段的数据时 [清除条件] • 当写“1”时
5	RXF	0	R/(W)*	接收正常结束 表示信息长度位所定数据量的接收正常结束。 [置位条件] • 1: 当信息长度位所定字节数的数据接收结束时 [清除条件] • 当写“1”时
4	RXEDE	0	R/(W)*	广播接收错误 在广播接收时接收控制字段期间，表示因接收缓冲器不为能接收的状态 (在 RE 位没被置“1”或者 RXBSY 标志被置位的状态) 而不能接收数据。此位在 IECTR 的 DEE 位为“1”时有效。 [置位条件] • 在广播接收中不能接收数据时 [清除条件] • 当写“1”时

位	位名	初始值	R/W	说明
3	RXEOVE	0	R/(W)*	<p>接收溢出标志</p> <p>此标志表示在数据接收过程中发生溢出。</p> <p>在没有清除 RXBSY 标志的状态下（即在没有读接收数据的状态下），如果开始接收下一个数据，此模块就将此位置位。此时，此模块判断为发生溢出错误，给通信对象返回 NAK。然后，通信对象进行重新发送，直到帧最大传输字节数为止。但是，在 RXBSY 标志被置位的状态下，此模块继续发送 NAK。如果 RXBSY 标志被清除，此模块就发送 ACK 并且取下一个数据。在广播接收时，如果在开始接收数据时 RXBSY 位已被置位，就立即进入待机状态。此标志在接收开始标志（RXS）被置位后开始有效。</p> <p>[置位条件]</p> <ul style="list-style-type: none"> • 在没有清除 RXBSY 标志的状态下接收到下一个数据时 <p>[清除条件]</p> <ul style="list-style-type: none"> • 当写“1”时
2	RXERTME	0	R/(W)*	<p>接收时序错误</p> <p>在数据接收过程中，如果不能根据 IEBus 协议规定的时序正确接收数据，就将此位置位。此模块在将此位置位后进入待机状态。此标志在接收开始标志（RXS）被置位后开始有效。如果在接收开始标志被置位前发生错误，就中止通信并且进入待机状态。此时，不将此位置位。</p> <p>[置位条件]</p> <ul style="list-style-type: none"> • 表示在数据接收过程中发生时序错误。 <p>[清除条件]</p> <ul style="list-style-type: none"> • 当写“1”时
1	RXEDLE	0	R/(W)*	<p>接收帧最大传输字节数超过</p> <p>表示在接收数据时发生以下某种情况：</p> <ul style="list-style-type: none"> • 因发生奇偶校验错误或者溢出错误而重新接收，导致在通信模式定义的最大字节长度内接收没有结束。 • 因信息长度的值大于最大传输字节数的值而使接收没有结束。 <p>此模块在将此位置位后进入待机状态。此标志在接收开始标志（RXS）被置位后开始有效。如果在将接收开始标志置位前发生错误，就中止通信并且进入待机状态。此时，不将此位置位。</p> <p>[置位条件]</p> <ul style="list-style-type: none"> • 表示在通信模式定义的最大字节数内接收没有结束。 <p>[清除条件]</p> <ul style="list-style-type: none"> • 当写“1”时

位	位名	初始值	R/W	说明
0	RXEPE	0	R/(W)*	<p>奇偶校验错误</p> <p>表示在接收数据字段的过程中发生奇偶校验错误。如果在接收数据字段前发生奇偶校验错误，此模块就立即进入待机状态，而不将 RXEPE 位置位。在接收数据字段的过程中，如果发生奇偶校验错误并且在达到帧最大传输字节数前没有进行接收，就不将 RXEPE 位置位。当发生奇偶校验错误时，此模块通过应答位给通信对象返回 NAK。此时，通信对象进行重新发送，直到帧最大传输字节数为止。但是，在重新接收过程中，如果解除了奇偶校验错误并且正常接收，就不将 RXEPE 位置位。在没有接收到信息长度设定的字节数而中止接收并且进入待机状态时，如果奇偶校验错误没被解除，就将 RXEPE 位置位。在广播接收时，如果在接收数据字段的过程中发生奇偶校验错误，此模块就将 RXEPE 位置位，并且立即进入待机状态。此标志在接收开始标志（RXS）被置位后开始有效。如果在接收开始标志被置位前发生错误，就中止通信并且进入待机状态。此时，不将此位置位。</p> <p>[置位条件]</p> <ul style="list-style-type: none"> 在接收到帧最大传输字节数为止而且最后接收字节的数据字段奇偶校验位不是偶校验时 <p>[清除条件]</p> <ul style="list-style-type: none"> 当写“1”时

【注】 * 为了清除标志，只能写“1”。

24.3.19 IEBus 接收中断允许寄存器 (IEIER)

IEIER 能禁止或者允许 IERSR 的接收忙、接收开始、接收正常结束、接收错误结束等中断源的中断。

位:	7	6	5	4	3	2	1	0
	RXBSYE	RXSE	RXFE	RXEDEE	RXE OVEE	RXE RTMEE	RXE DLEE	RXEPEE
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	RXBSYE	0	R/W	接收忙中断允许 禁止或者允许接收忙 (RXBSY) 中断。 0: 禁止接收忙 (RXBSY) 中断 1: 允许接收忙 (RXBSY) 中断
6	RXSE	0	R/W	接收开始中断允许 禁止或者允许接收开始 (RXS) 中断。 0: 禁止接收开始 (RXS) 中断 1: 允许接收开始 (RXS) 中断
5	RXFE	0	R/W	接收正常结束中断允许 禁止或者允许接收正常结束 (RXF) 中断。 0: 禁止接收正常结束 (RXF) 中断 1: 允许接收正常结束 (RXF) 中断
4	RXEDEE	0	R/W	广播接收错误中断允许 禁止或者允许广播接收错误 (RXEDE) 中断。 0: 禁止广播接收错误 (RXEDE) 中断 1: 允许广播接收错误 (RXEDE) 中断
3	RXEOVEE	0	R/W	溢出控制标志中断允许 禁止或者允许溢出控制标志 (RXEOVE) 中断。 0: 禁止溢出控制标志 (RXEOVE) 中断 1: 允许溢出控制标志 (RXEOVE) 中断
2	RXERTMEE	0	R/W	接收时序错误中断允许 禁止或者允许接收时序错误 (RXERTME) 中断。 0: 禁止接收时序错误 (RXERTME) 中断 1: 允许接收时序错误 (RXERTME) 中断
1	RXEDLEE	0	R/W	接收帧最大传输字节数超过中断允许 禁止或者允许接收帧最大传输字节数超过 (RXEDLE) 中断。 0: 禁止接收帧最大传输字节数超过 (RXEDLE) 中断 1: 允许接收帧最大传输字节数超过 (RXEDLE) 中断
0	RXEPEE	0	R/W	奇偶校验错误中断允许 禁止或者允许奇偶校验错误 (RXEPE) 中断。 0: 禁止奇偶校验错误 (RXEPE) 中断 1: 允许奇偶校验错误 (RXEPE) 中断

24.3.20 IEBus 时钟选择寄存器 (IECKSR)

IECKSR 是 8 位可读写寄存器，设定此模块使用的时钟。

位:	7	6	5	4	3	2	1	0
	—	—	—	CKS3	—	CKS[2:0]		
初始值:	0	0	0	0	0	0	0	1
R/W:	R	R	R	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	CKS3	0	R/W	输入时钟选择 3*1*2 选择此模块使用的时钟。 0: 使用外围时钟 0 (p0φ) 1: 使用 AUDIO_X1 和 AUDIO_X2 的时钟
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	CKS[2:0]	001	R/W	输入时钟选择 2 ~ 0*1 选择此模块使用的时钟分频。 000: 禁止设定 001: 使用 CKS3 选择的时钟 (IEBφ) 的 1/2 时钟 (IEBφ=12MHz、12.58MHz) 010: 使用 CKS3 选择的时钟 (IEBφ) 的 1/3 时钟 (IEBφ=18MHz、18.87MHz) 011: 使用 CKS3 选择的时钟 (IEBφ) 的 1/4 时钟 (IEBφ=24MHz、25.16MHz) 100: 使用 CKS3 选择的时钟 (IEBφ) 的 1/5 时钟 (IEBφ=30MHz、31.45MHz) 101: 使用 CKS3 选择的时钟 (IEBφ) 的 1/6 时钟 (IEBφ=36MHz、37.74MHz) 110: 使用 CKS3 选择的时钟 (IEBφ) 的 1/7 时钟 (IEBφ=42MHz、44.03MHz) 111: 使用 CKS3 选择的时钟 (IEBφ) 的 1/8 时钟 (IEBφ=48MHz)

【注】 *1 不能在 IEBus 发送和接收过程中更改 CKS3 位和 CKS0[2:0] 位的设定。

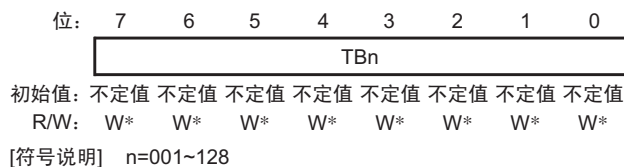
*2 在 CKS3 位为“1”时，必须将 STBCR3 寄存器的 MSTP36 位置“0”。

有关 STBCR3 寄存器的设定方法，请参照“49. 低功耗模式”。

24.3.21 IEBus 发送数据缓冲器 001 ~ 128 (IETB001 ~ IETB128)

这是在主控发送时写发送数据的 128 字节 (8×128) 缓冲器。

IETB001 ~ IETB128 的初始值为不定值。



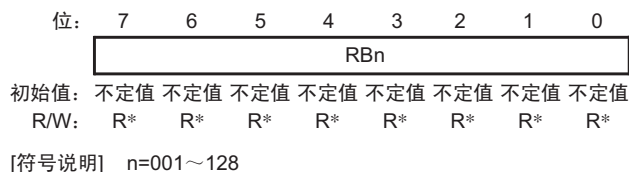
位	位名	初始值	R/W	说明
7 ~ 0	TBn	不定值	W*	IEBus 发送数据缓冲器 在主控发送时, 将数据字段发送的数据写到 TB001 ~ TB128。按 TB001 为 1 字节的起始数据, 按 TB002、TB003……的顺序写发送数据。TB128 为 128 字节发送时的最后数据。

【注】 * 在主控发送 (IEFLG 的 MRQ 位为 “1”) 时禁止写。

24.3.22 IEBus 接收数据缓冲器 001 ~ 128 (IERB001 ~ IERB128)

这是在从属接收时保存发送数据的 128 字节 (8×128) 缓冲器。

IERB001 ~ IERB128 的初始值为不定值。



位	位名	初始值	R/W	说明
7 ~ 0	RBn	不定值	R*	IEBus 接收数据缓冲器 在 IEBus 接收状态寄存器 (IERSR) 的 RXBSY 位为 “1” 时, 能读 RB001 ~ RB128。通过 RB001 ~ RB128 读到的数据为从属接收时的数据字段的值。RB001 为 1 字节的起始数据, 按 RB002、RB003……的顺序写接收数据。RB128 为 128 字节接收时的最后数据。

【注】 * 在从属接收 (IEFLG 的 SRE 位为 “1” 并且 IERSR 的 RXBSY 位为 “0”) 时禁止读 (读取值为不定值)。

24.4 数据格式

24.4.1 发送格式

IEBus 发送数据时的传输格式和各寄存器的关系如图 24.6 所示。

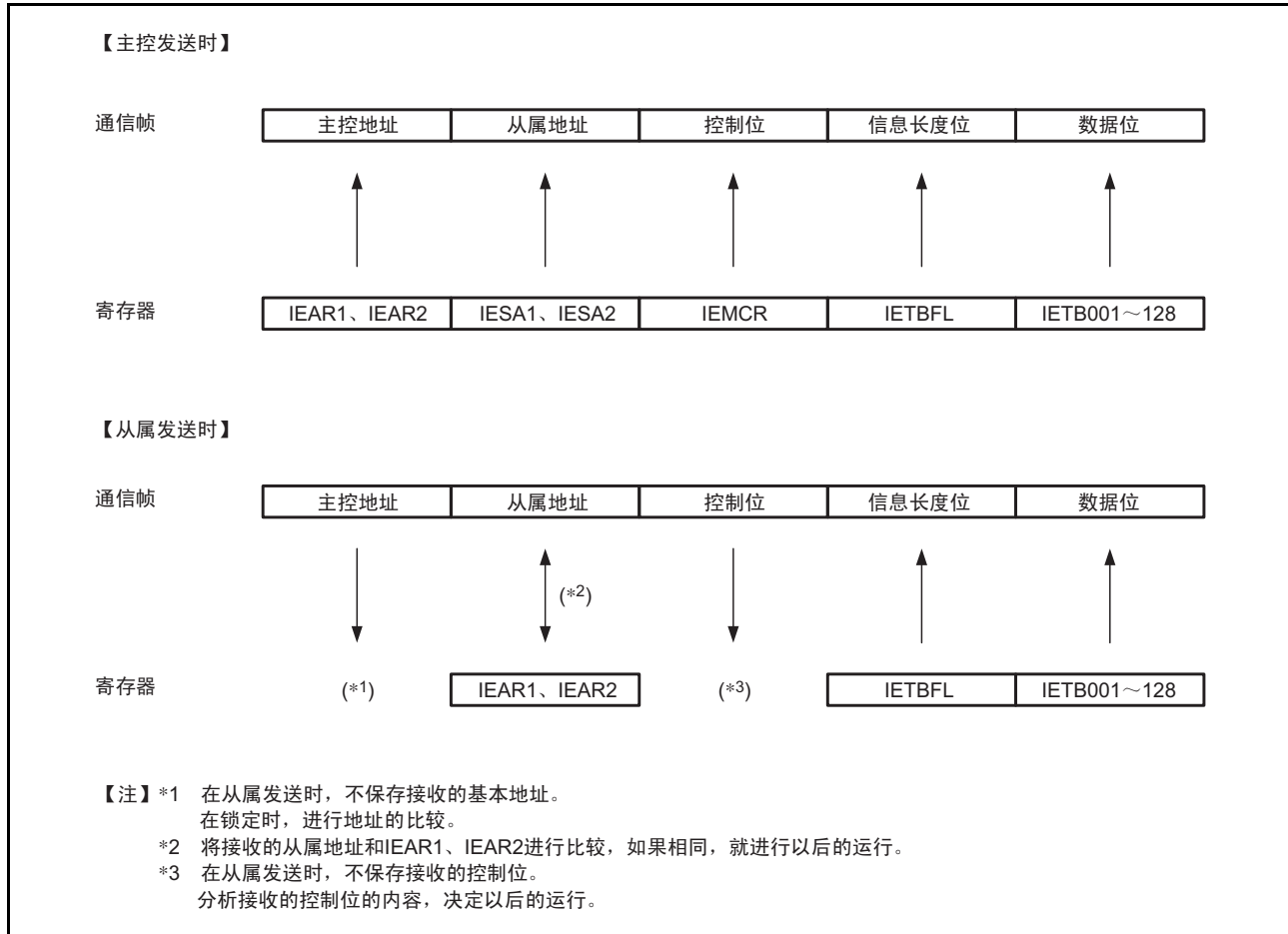


图 24.6 发送时的传输信号格式和各寄存器的关系

24.4.2 接收格式

IEBus 接收数据时的传输格式和各寄存器的关系如图 24.7 所示。

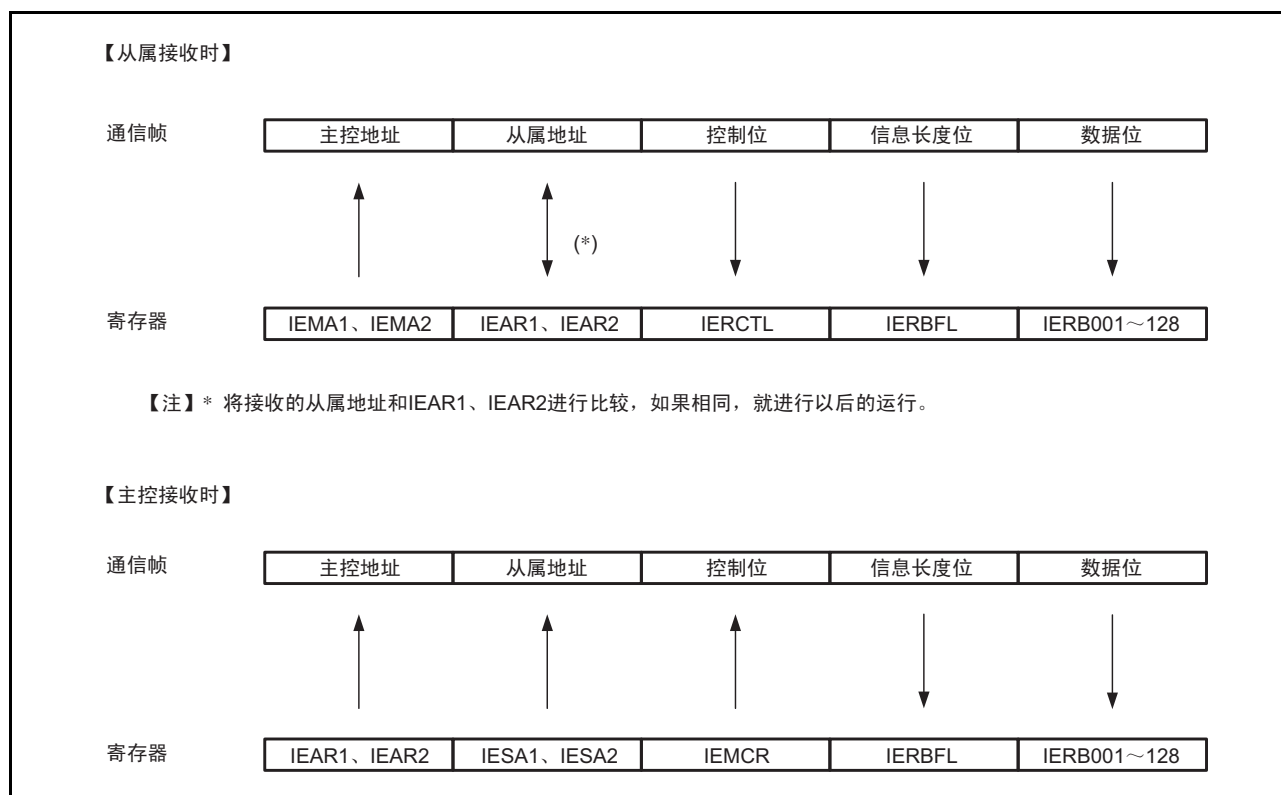


图 24.7 接收时的传输信号格式和各寄存器的关系

24.5 软件控制流程

24.5.1 初始设定

初始设定流程如图 24.8 所示。

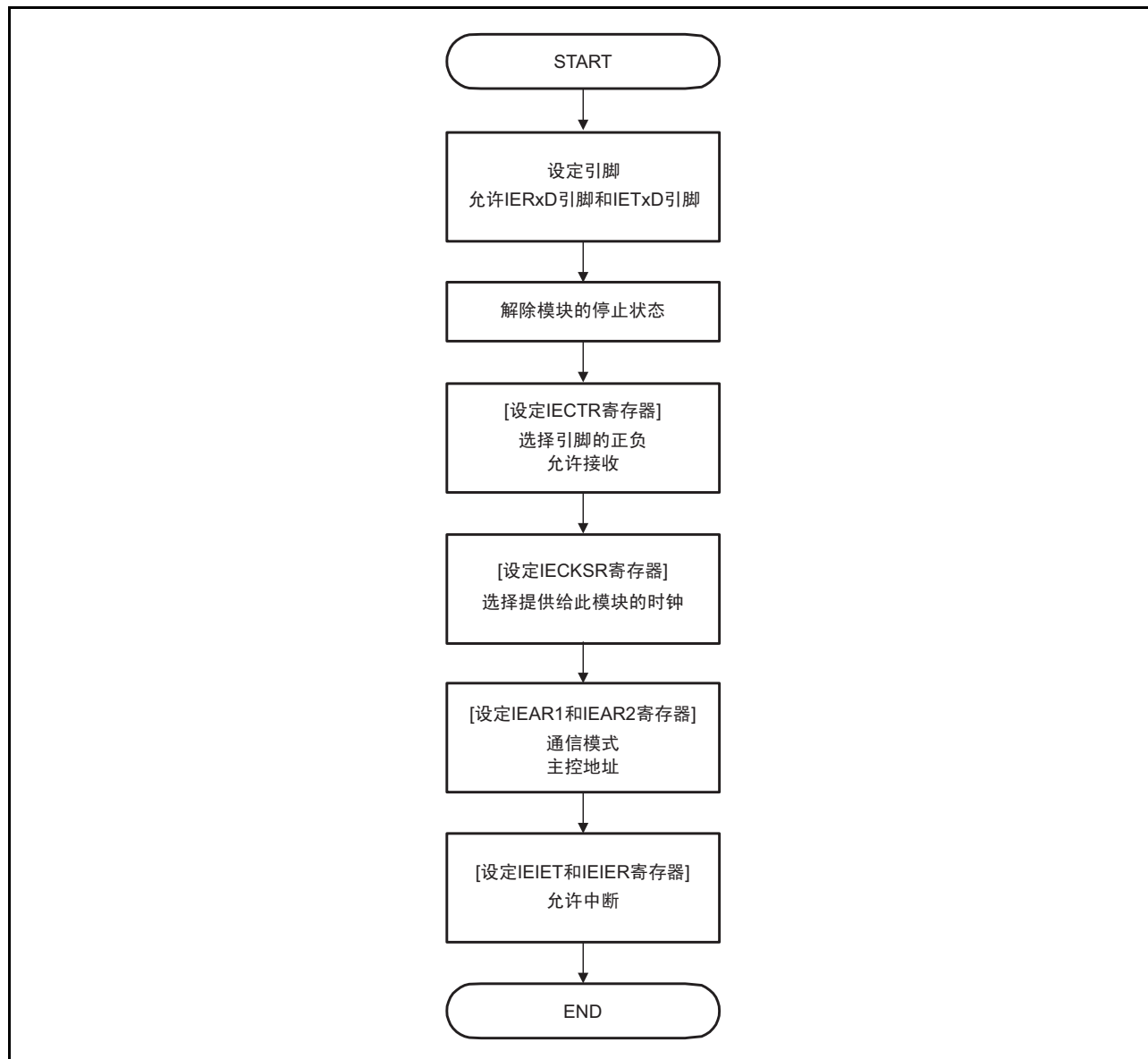


图 24.8 初始设定流程

24.5.2 主控发送

主控发送流程如图 24.9 所示。

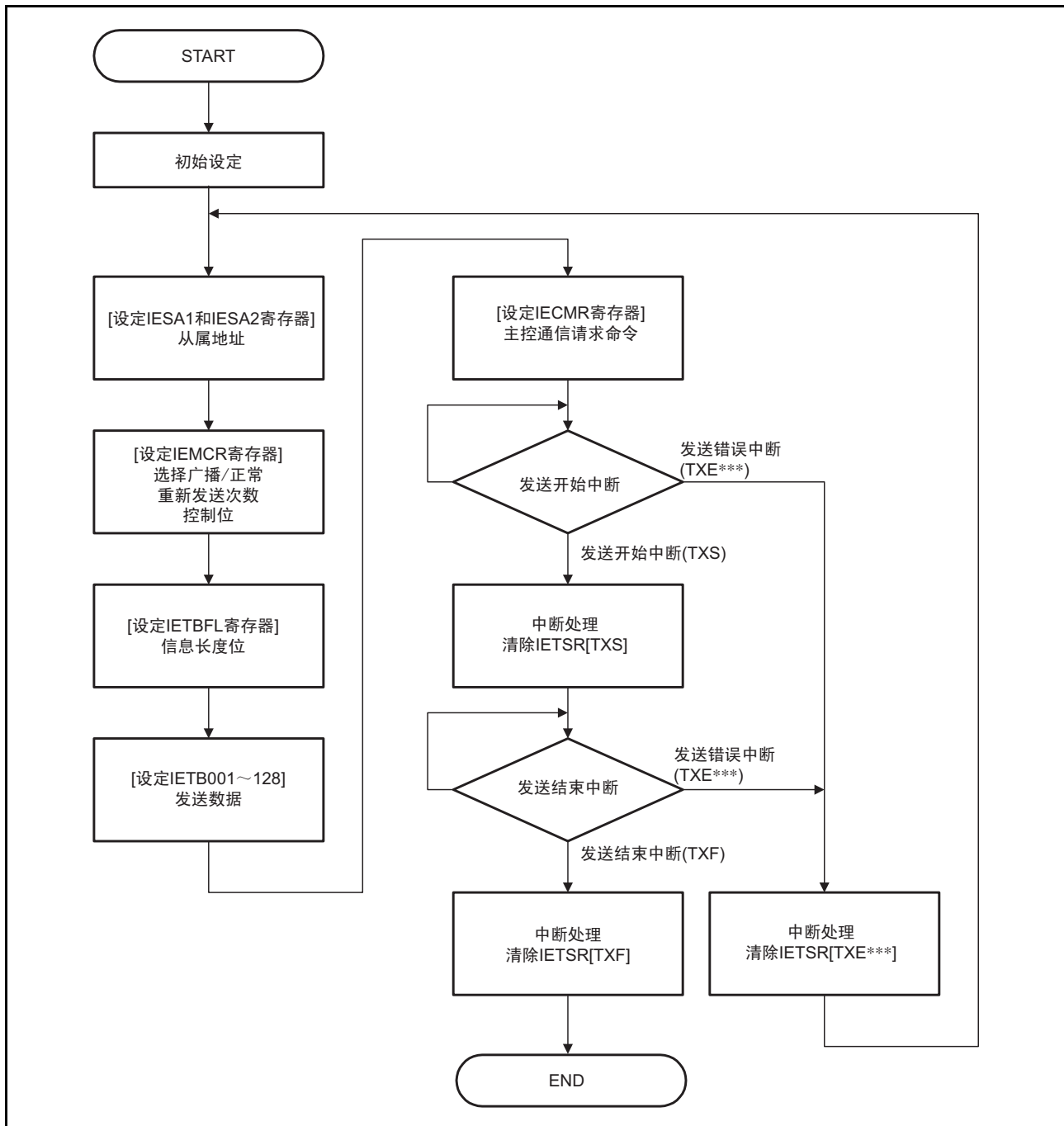


图 24.9 主控发送流程

24.5.3 从属接收

从属接收流程如图 24.10 所示。

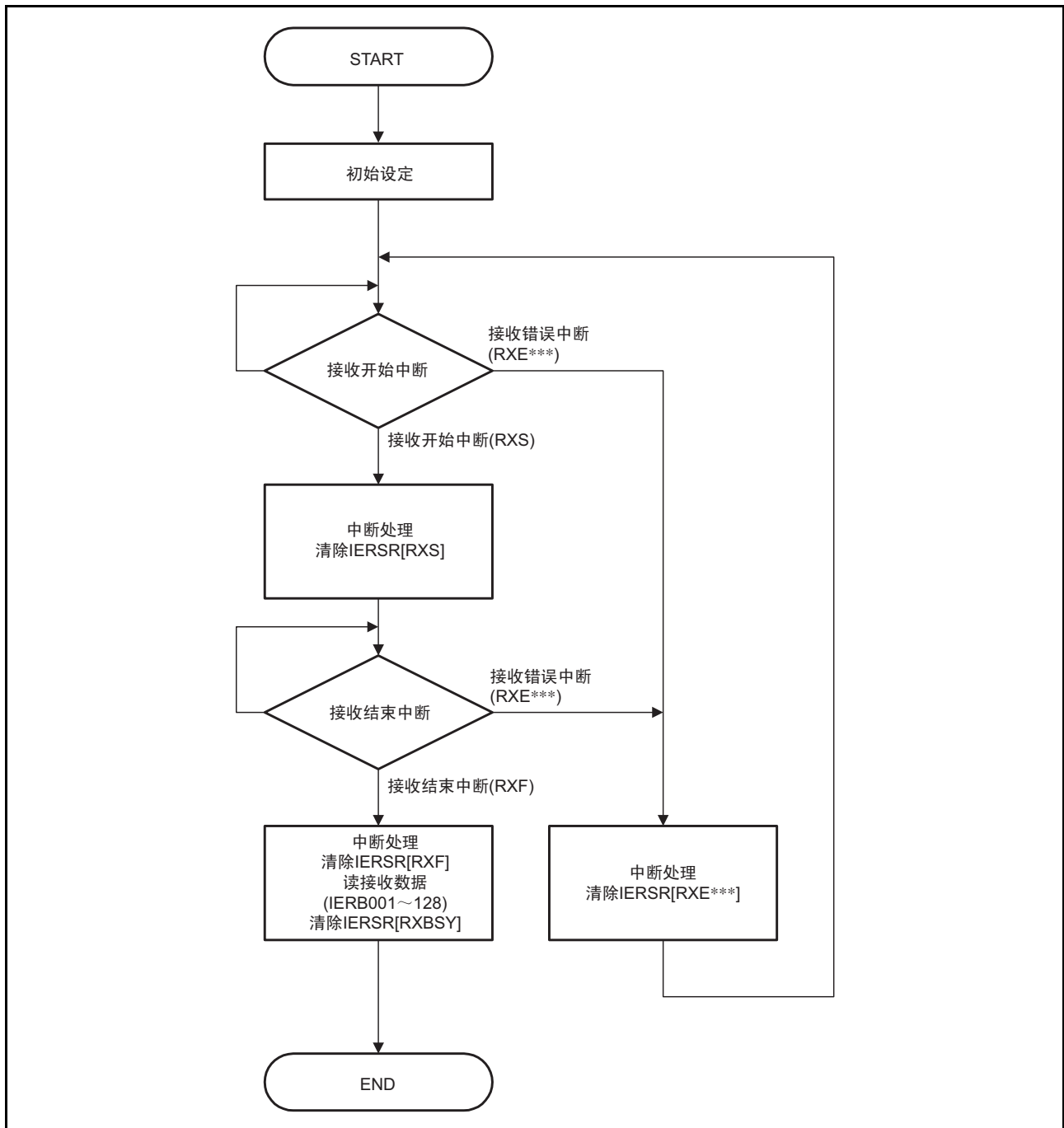


图 24.10 从属接收流程

24.5.4 主控接收

主控接收流程如图 24.11 所示。

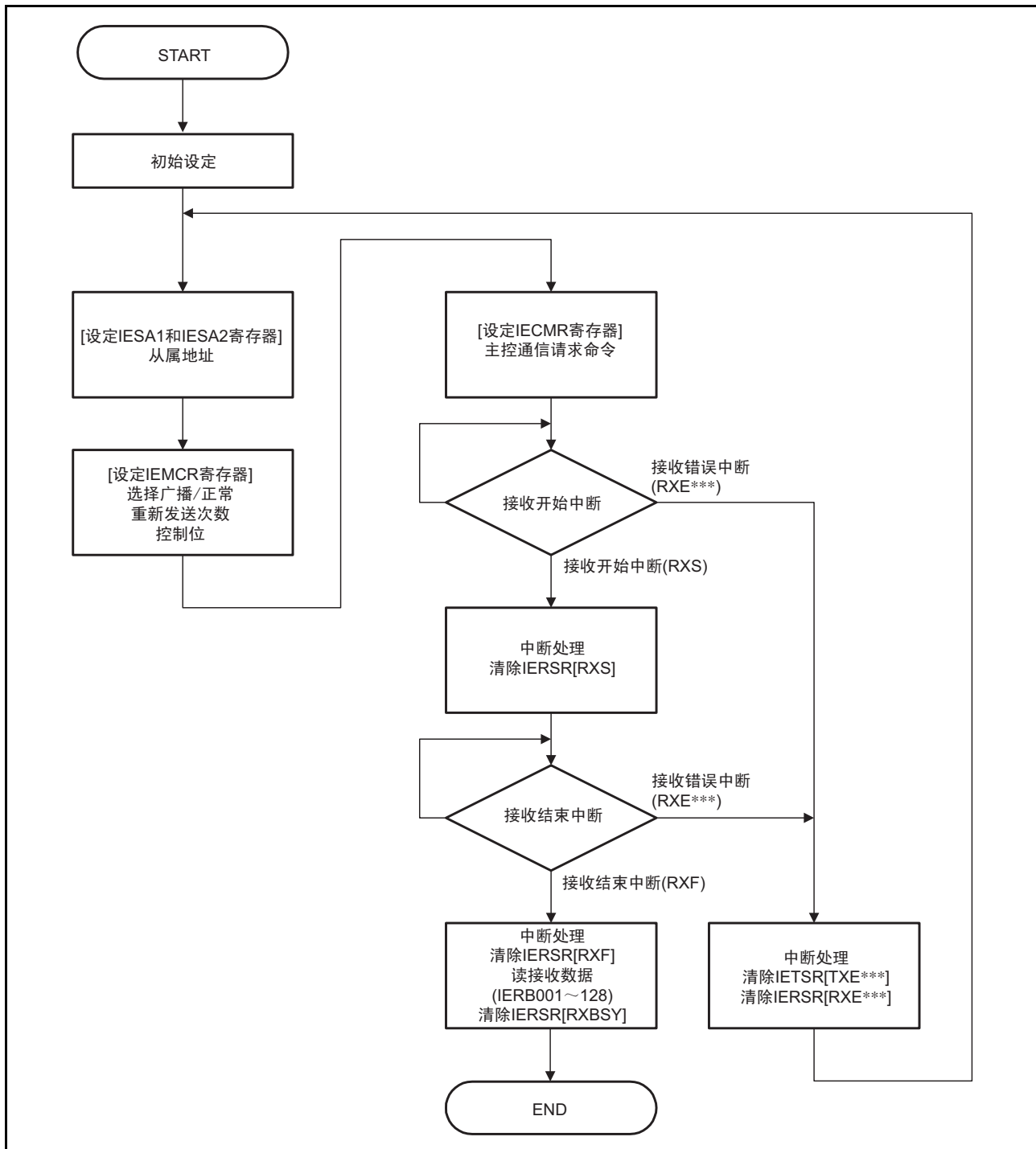


图 24.11 主控接收流程

24.5.5 从属发送

从属发送流程如图 24.12 所示。

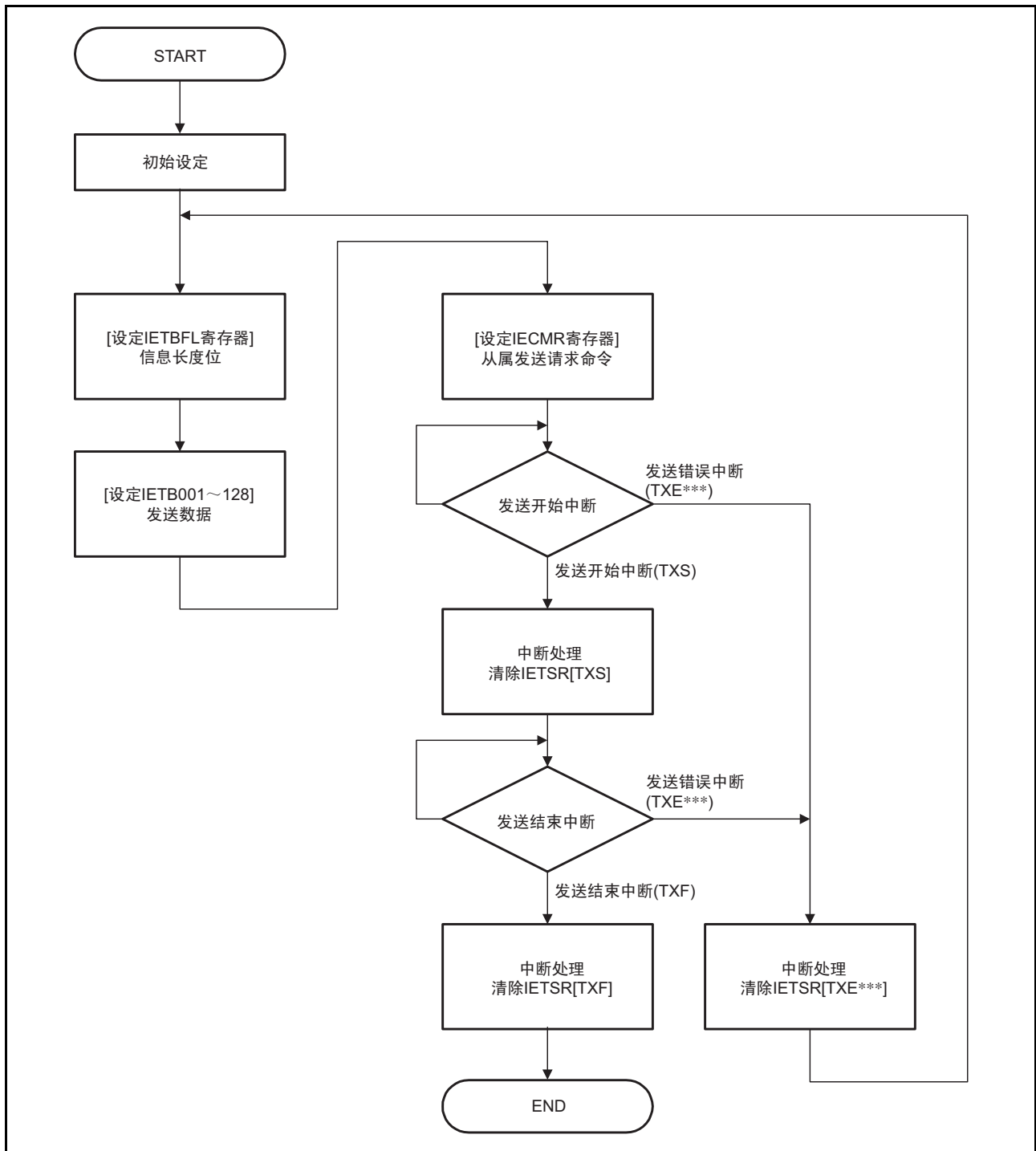


图 24.12 从属发送流程

24.6 运行时序

24.6.1 主控发送

主控发送时序如图 24.13 所示。

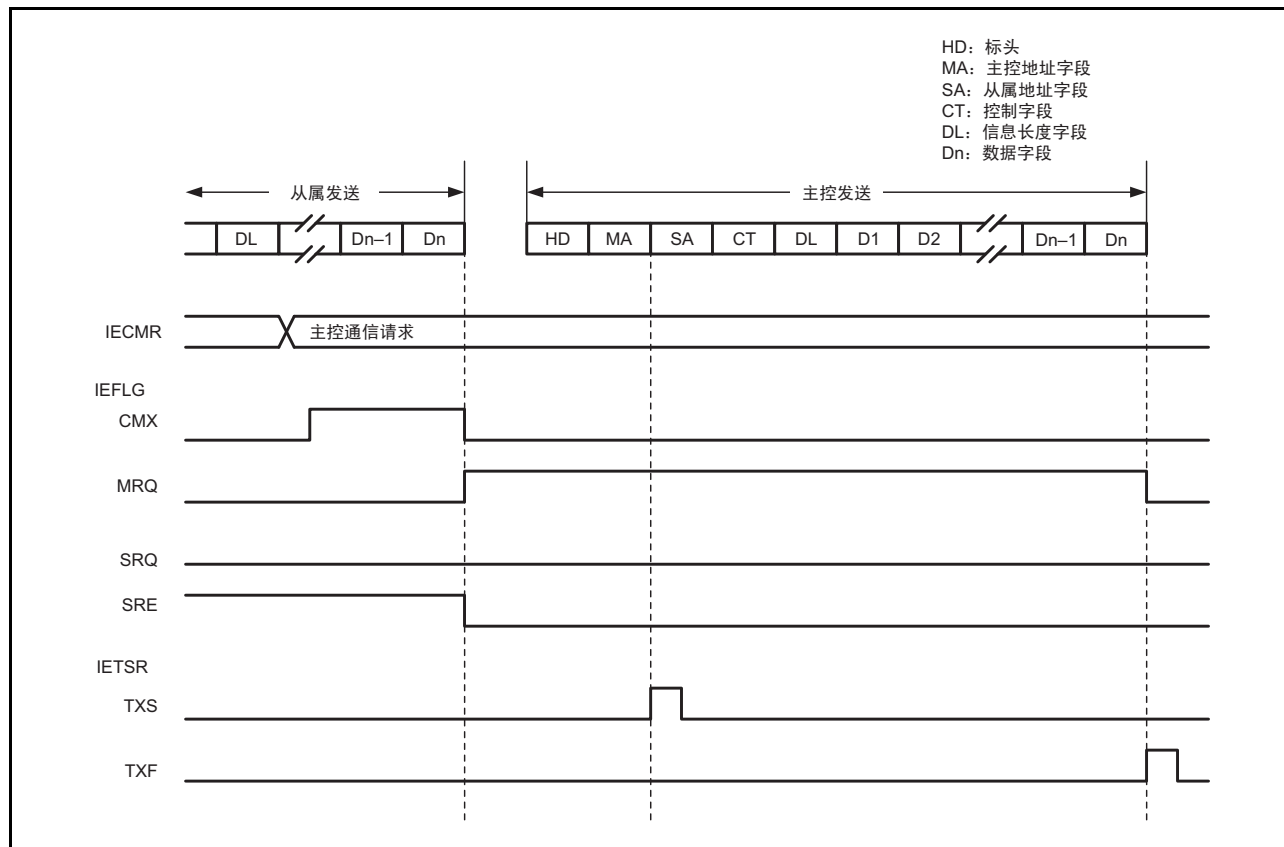


图 24.13 主控发送时序

24.6.2 从属接收

从属接收时序如图 24.14 所示。

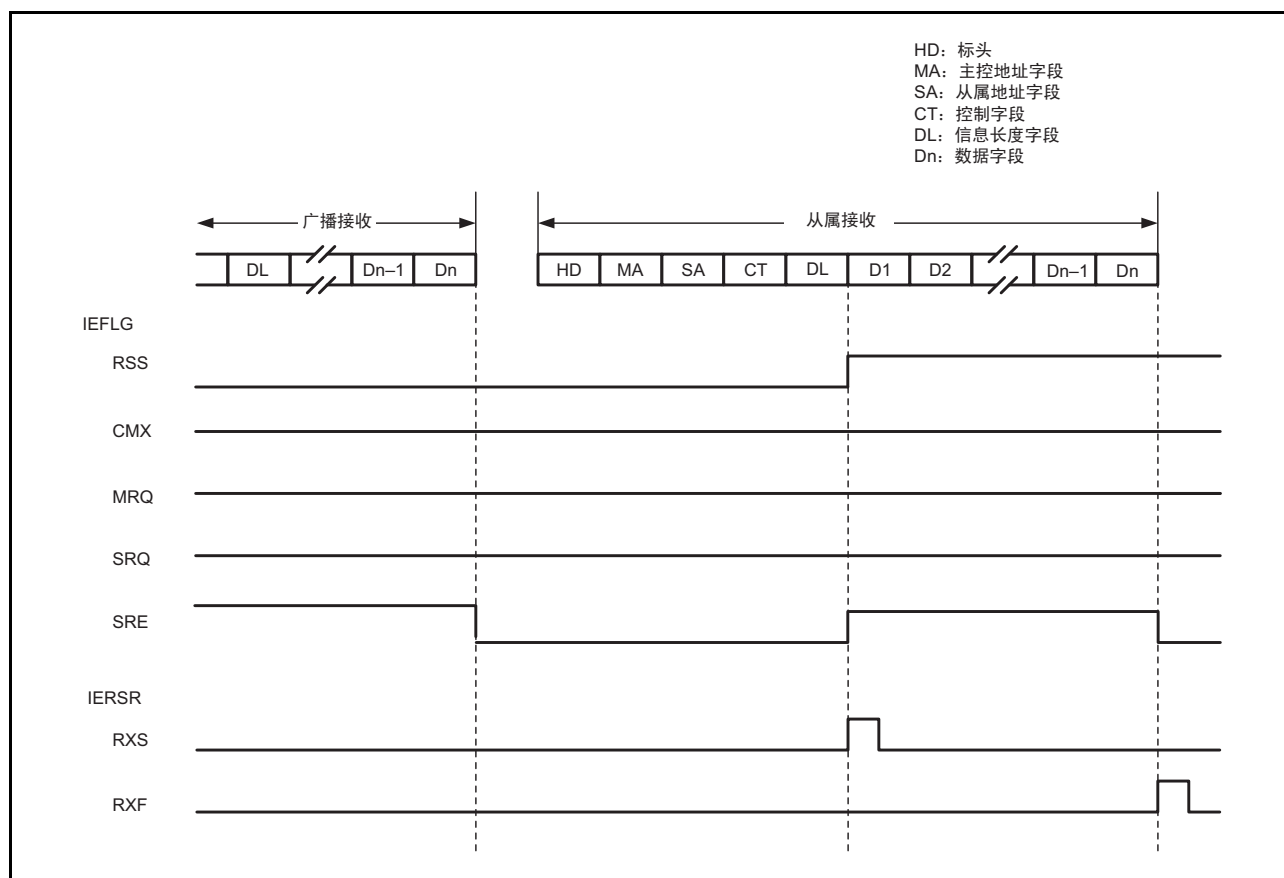


图 24.14 从属接收时序

24.6.3 主控接收

主控接收时序如图 24.15 所示。

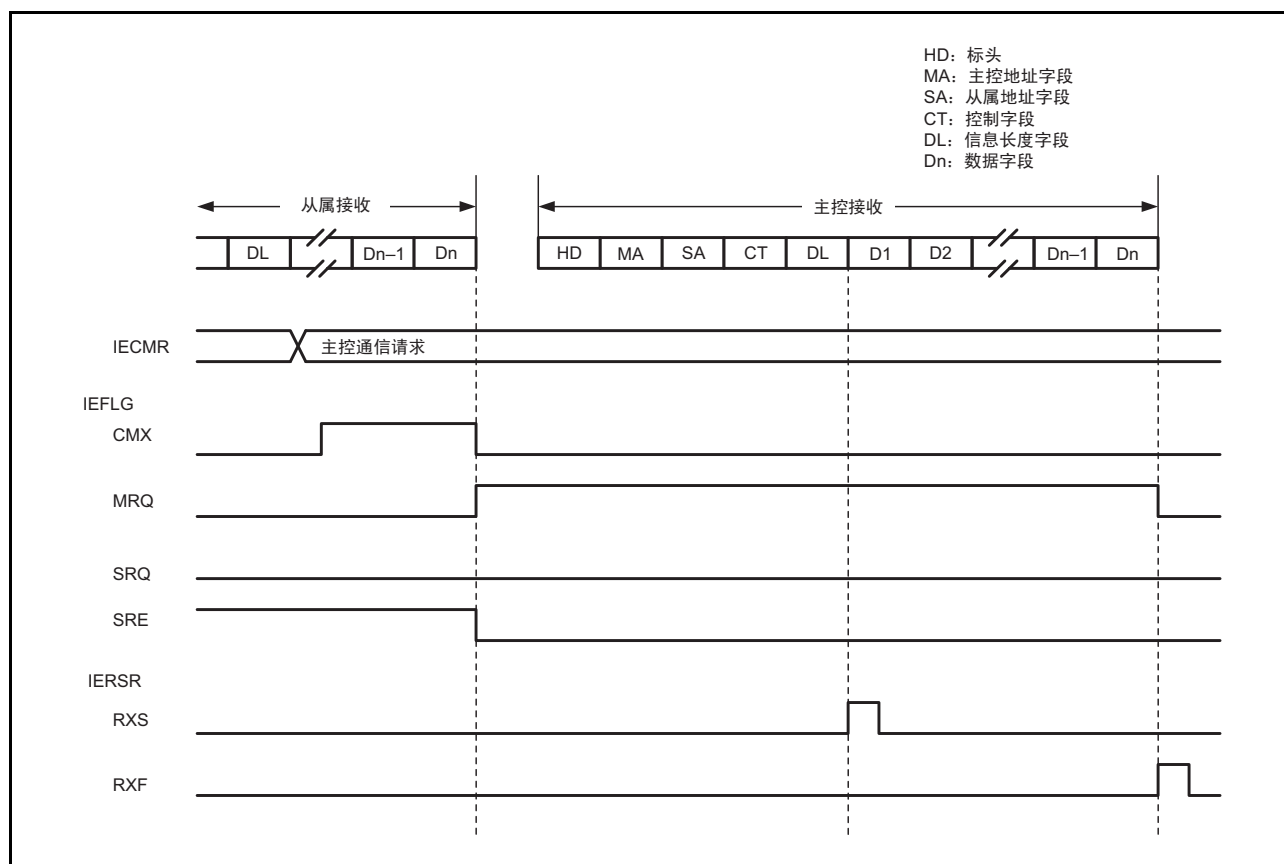


图 24.15 主控接收时序

24.6.4 从属发送

从属发送时序如图 24.16 所示。

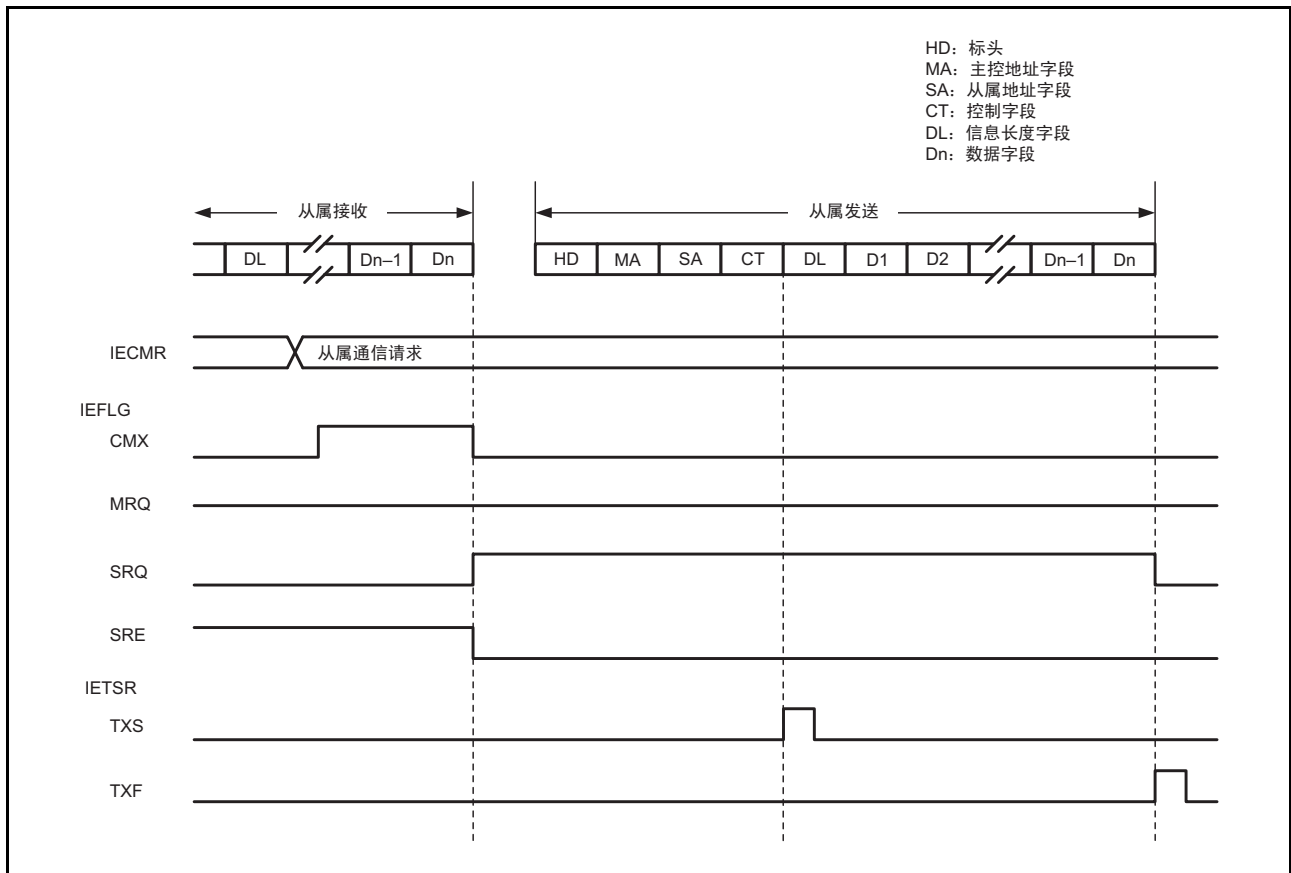


图 24.16 从属发送时序

24.7 中断源

此模块的中断有发送开始 (TXS)、发送正常结束 (TXF)、仲裁失败 (TXEAL)、发送时序错误 (TXETTME)、发送帧最大传输字节数超过 (TXERO)、应答位 (TXEACK)、接收忙 (RXBSY)、接收开始 (RXS)、接收正常结束 (RXF)、广播接收错误 (RXEDE)、接收溢出标志 (RXEOVE)、接收时序错误 (RXERTME)、接收帧最大传输字节数超过 (RXEDLE)、奇偶校验错误 (RXEPE)。

各中断源有对应 IEBus 发送中断允许寄存器 (IEIET) 和 IEBus 接收中断允许寄存器 (IEIER) 的位, 能禁止或者允许中断, 并且还有对应 IEBus 发送状态寄存器 (IETSR) 和 IEBus 接收状态寄存器 (IERSR) 的状态标志, 能通过读状态标志判断中断源。

中断源的关系如图 24.17 所示。

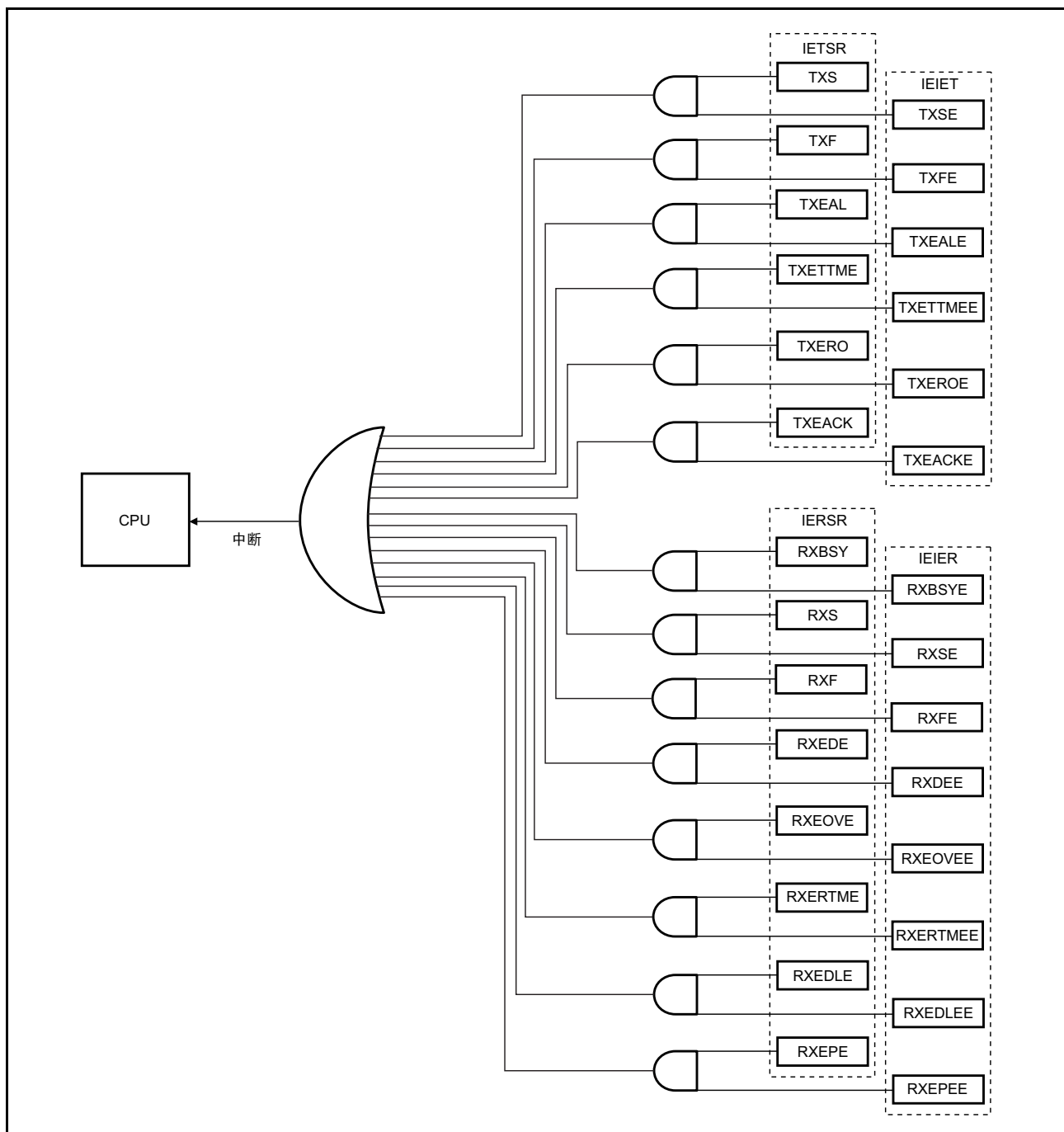


图 24.17 中断源的关系

24.8 使用时的注意事项

24.8.1 在最大传输字节长度内通信没有结束时的注意事项

(1) 数据发送

在发送数据时，如果发生以下某种情况，就将 IETSR 的错误标志置位，并且进入待机状态。

- 因从接收单元接收到 NAK 而进行了通信模式定义的最大字节长度的发送。
- 因信息长度的值大于最大传输字节数的值而在最大字节长度内发送没有结束。

此时，进行第 (最大传输字节数 +1) 字节的发送。然后，如果通过第 (最大传输字节数 +1) 字节的应答位接收到 NAK，就将 TXERO 标志置位。如果接收到的不是 NAK 而是 ACK，就不将 TXF 标志置位。

在最大字节长度内发送没有结束时的运行时序如图 24.18 所示。

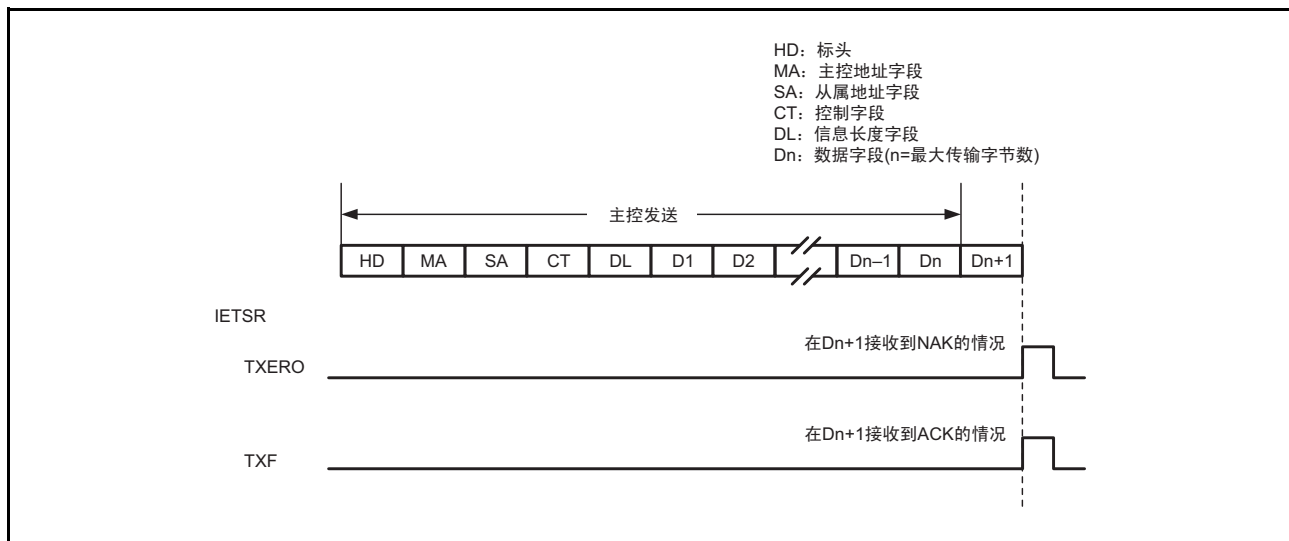


图 24.18 在最大字节长度内发送没有结束时的运行时序

(2) 数据接收

在接收数据时，如果发生以下某种情况，就将 IERSR 的错误标志置位，然后进入待机状态。

- 因发生奇偶校验错误或者溢出错误而由重新发送引起接收，导致在通信模式定义的最大字节长度内接收没有结束。
- 因信息长度的值大于最大传输字节数的值而接收没有结束。

此时，为第 (最大传输字节数 +1) 字节的数据接收等待状态。因此，如果不能接收第 (最大传输字节数 +1) 字节的数据，就检测到接收时序错误，并且将 RXERTME 标志置位。此时，不将 RXEDLE 标志置位，而在接收到第 (最大传输字节数 +1) 字节的数据时将 RXEDLE 标志置位。

接收到最大传输字节长度的数据并且奇偶校验错误没被解除的情况也一样，如果不能接收第 (最大传输字节数 +1) 字节的数据，就将 RXERTME 标志置位。此时，不将 RXEPE 标志置位，而在接收到 (最大传输字节数 +1) 字节的数据时将 RXEPE 标志置位。

在最大字节长度内接收没有结束时的运行时序如图 24.19 所示。

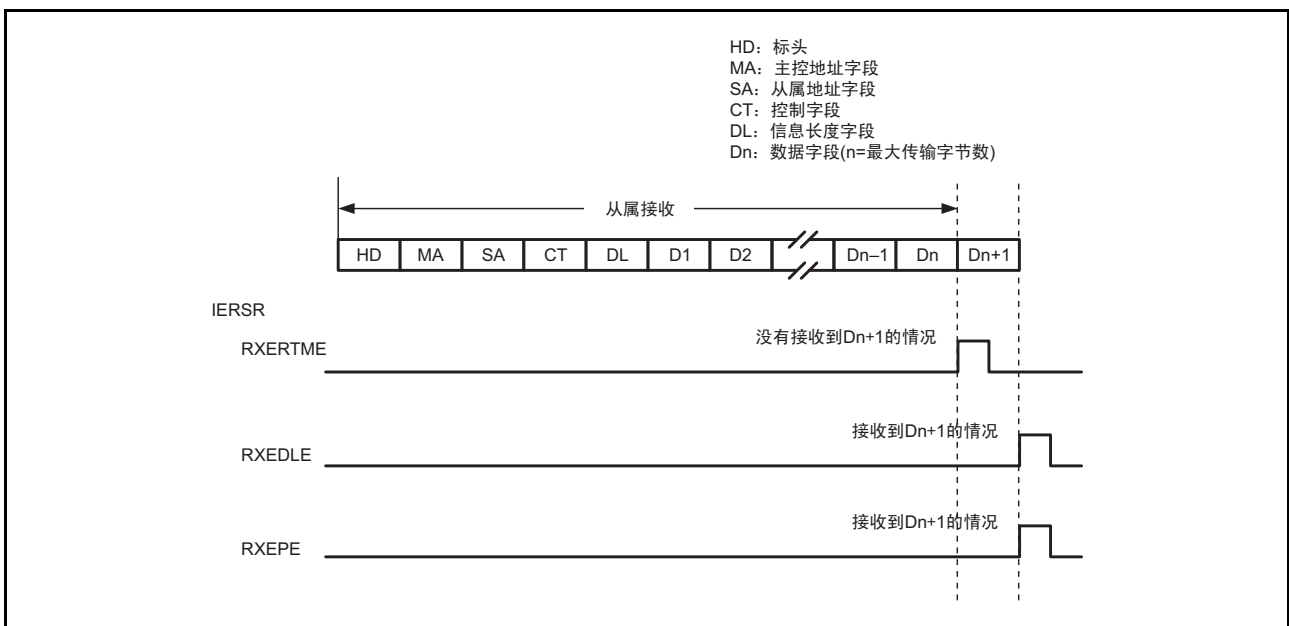


图 24.19 在最大字节长度内接收没有结束时的运行时序

25. 瑞萨 SPDIF 接口

25.1 概要

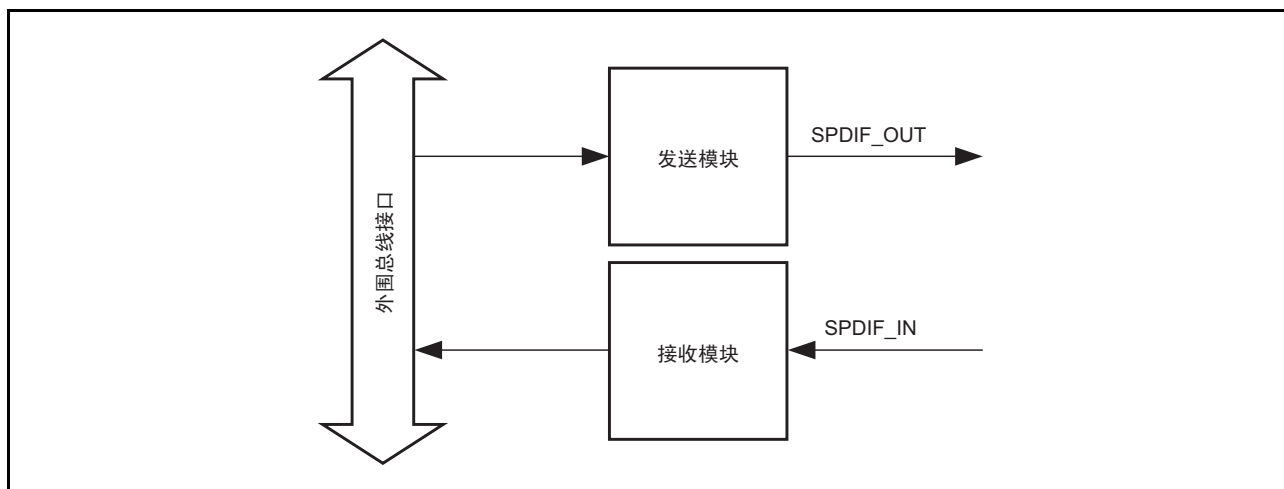


图 25.1 概要框图

25.2 特点

- 符合 IEC60958 规格（只限于立体声和民用模式）。
- 采样频率：32kHz、44.1kHz、48kHz
- 音频字长：16~24 位/样本
- 双相传号方式的编码
- 数据的双缓冲
- 带奇偶校验的串行数据
- 能同时发送和接收。
- 接收模块能自动检测到 IEC61937 压缩模式的数据。

25.3 功能框图

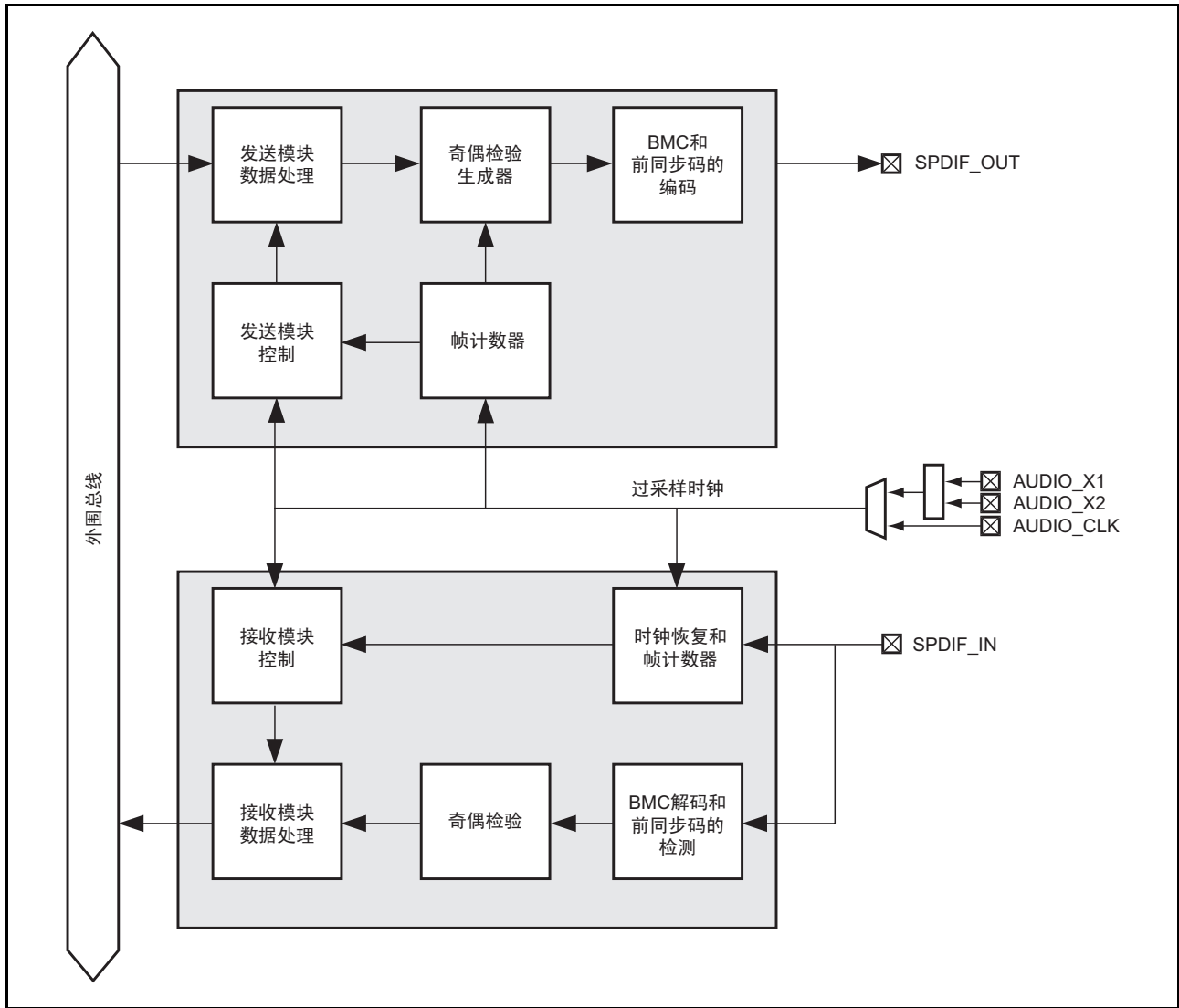


图 25.2 功能框图

25.4 输入 / 输出引脚

引脚结构如表 25.1 所示。

表 25.1 引脚结构

通道	名称	输入 / 输出	功能
0	SPDIF_IN	输入	发送侧双相传号方式的编码 SPDIF 比特流
1	SPDIF_OUT	输出	接收侧双相传号方式的编码 SPDIF 比特流
0、1 共用	AUDIO_CLK	输入	音频外部时钟
	AUDIO_X1	输入	音频晶体谐振器 / 外部时钟
	AUDIO_X2	输出	

25.6 寄存器结构

寄存器结构如表 25.3 所示。

表 25.3 寄存器结构

通道	名称	略称	地址	存取长度
0 (发送)	发送模块通道 1 的音频寄存器	TLCA	H'E801 2000	32
	发送模块通道 2 的音频寄存器	TRCA	H'E801 2004	32
	发送模块通道 1 的状态寄存器	TLCS	H'E801 2008	32
	发送模块通道 2 的状态寄存器	TRCS	H'E801 200C	32
	发送用户数据寄存器	TUI	H'E801 2010	32
1 (接收)	接收模块通道 1 的音频寄存器	RLCA	H'E801 2014	32
	接收模块通道 2 的音频寄存器	RRCA	H'E801 2018	32
	接收模块通道 1 的状态寄存器	RLCS	H'E801 201C	32
	接收模块通道 2 的状态寄存器	RRCS	H'E801 2020	32
	接收用户数据寄存器	RUI	H'E801 2024	32
0、1 共用	控制寄存器	CTRL	H'E801 2028	32
	状态寄存器	STAT	H'E801 202C	32
0、1 共用	发送模块 DMA 的音频数据寄存器	TDAD	H'E801 2030	32
	接收模块 DMA 的音频数据寄存器	RDAD	H'E801 2034	32

【注】 因为寄存器全部是长字寄存器，所以必须总是用长字进行存取。

对于各寄存器图中值为“0”的位，只能写“0”（可写的情况）。此时，读取值总是为“0”（可读的情况）。

25.7 寄存器说明

【寄存器的符号说明】

初始值：复位后的寄存器值

—：不定值

R/W：可读写。能读到写入值。

R：只能读。写入值总是“0”。

R/WC0：可读写。如果写“0”，该位就被初始化，但是忽视写“1”。

R/WC1：可读写。如果写“1”，该位就被初始化，但是忽视写“0”。

W：只能写，禁止读。在保留位的情况下，写入值总是“0”。

—/W：只能写。读取值为不定值。

25.7.1 控制寄存器 (CTRL)

位:	31	30	29	28	27	26	25	24
	—	—	—	CKS	—	PB	RASS	
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R/W	R/W	R/W
位:	23	22	21	20	19	18	17	16
	TASS		RDE	TDE	NCSI	AOS	RME	TME
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8
	REIE	TEIE	UBOI	UBUI	CREI	PAEI	PREI	CSEI
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	7	6	5	4	3	2	1	0
	ABOI	ABUI	RUII	TUII	RCSI	RCBI	TCSI	TCBI
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 29	—	全 0	R	保留位
28	CKS	0	R/W	过采样时钟选择位 选择过采样时钟的供给源。 0: AUDIO_X1 1: AUDIO_CLK
27	—	0	R	保留位
26	PB	0	R/W	回传 将发送模块 SPDIF 的输出送回到 SPDIF 模块的 SPDIF 接收模块 (回传)。 0: 回传无效 1: 回传有效
25、24	RASS	全 0	R/W	接收模块音频样本位长 指定用于数据调整的接收模块音频样本位长 (16 位、20 位或者 24 位)。 00: 16 位样本 01: 20 位样本 10: 24 位样本 11: 保留
23、22	TASS	全 0	R/W	发送模块音频样本位长 指定用于数据调整的发送模块音频样本位长 (16 位、20 位或者 24 位)。 00: 16 位样本 01: 20 位样本 10: 24 位样本 11: 保留
21	RDE	0	R/W	接收模块 DMA 允许 允许接收模块的 DMA 请求。 0: 禁止接收模块 DMA 1: 允许接收模块 DMA

位	位名	初始值	R/W	说明
20	TDE	0	R/W	发送模块 DMA 允许 允许发送模块的 DMA 请求。 0: 禁止发送模块 DMA 1: 允许发送模块 DMA
19	NCSI	0	R/W	新通道状态信息 当发送模块中有应该更改的新通道状态信息时, 将此位置“1”。 0: 没有新通道状态信息 1: 有新通道状态信息
18	AOS	0	R/W	音频单独采样 当用户数据寄存器有用户信息时, 将此位置“0”。如果置“1”, 就将用户位全部置“0”。 0: 有用户信息 1: 没有用户信息
17	RME	0	R/W	接收模块允许 将接收模块置为有效。 0: 接收模块无效 1: 接收模块有效
16	TME	0	R/W	发送模块允许 将发送模块置为有效。 0: 发送模块无效 1: 发送模块有效
15	REIE	0	R/W	接收模块错误中断允许 如果清除此位, 就屏蔽全部接收模块的错误中断; 如果将此位置位, 就允许全部接收模块的错误中断。 0: 禁止接收模块错误中断 1: 允许接收模块错误中断
14	TEIE	0	R/W	发送模块错误中断允许 如果清除此位, 就屏蔽全部发送模块的错误中断; 如果将此位置位, 就允许全部发送模块的错误中断。 0: 禁止发送模块错误中断 1: 允许发送模块错误中断
13	UBOI	0	R/W	用户缓冲器超限中断允许 允许用户缓冲器超限中断。 0: 禁止用户缓冲器超限中断 1: 允许用户缓冲器超限中断
12	UBUI	0	R/W	用户缓冲器欠载中断允许 允许用户缓冲器欠载中断。 0: 禁止用户缓冲器欠载中断 1: 允许用户缓冲器欠载中断
11	CREI	0	R/W	时钟恢复错误中断允许 允许时钟恢复错误中断。 0: 禁止时钟恢复错误中断 1: 允许时钟恢复错误中断
10	PAEI	0	R/W	奇偶校验错误中断允许 允许奇偶校验错误中断。 0: 禁止奇偶校验错误中断 1: 允许奇偶校验错误中断

位	位名	初始值	R/W	说明
9	PREI	0	R/W	前同步码错误中断允许 允许前同步码校验错误中断。 0: 禁止前同步码错误中断 1: 允许前同步码错误中断
8	CSEI	0	R/W	通道状态错误中断允许 允许通道状态错误中断。 0: 禁止通道状态错误中断 1: 允许通道状态错误中断
7	ABOI	0	R/W	音频缓冲器超限中断允许 允许接收模块的音频缓冲器超限中断。 0: 禁止音频缓冲器超限中断 1: 允许音频缓冲器超限中断
6	ABUI	0	R/W	音频缓冲器欠载中断允许 允许发送模块的音频缓冲器欠载中断。 0: 禁止音频缓冲器欠载中断 1: 允许音频缓冲器欠载中断
5	RUII	0	R/W	接收模块用户信息中断允许 允许接收模块的用户信息寄存器满中断。 0: 禁止接收模块的用户信息中断 1: 允许接收模块的用户信息中断
4	TUII	0	R/W	发送模块用户信息中断允许 允许发送模块的用户信息寄存器空中断。 0: 禁止发送模块的用户信息中断 1: 允许发送模块的用户信息中断
3	RCSI	0	R/W	接收模块通道状态中断允许 允许接收模块的通道状态寄存器空中断。 0: 禁止接收模块的通道状态中断 1: 允许接收模块的通道状态中断
2	RCBI	0	R/W	接收模块通道缓冲器中断允许 允许接收模块的音频通道缓冲器空中断。 0: 禁止接收模块的音频通道中断 1: 允许接收模块的音频通道中断
1	TCSI	0	R/W	发送模块通道状态中断允许 允许发送模块的通道状态寄存器空中断。 0: 禁止发送模块的通道状态中断 1: 允许发送模块的通道状态中断
0	TCBI	0	R/W	发送模块通道缓冲器中断允许 允许发送模块的音频通道缓冲器空中断。 0: 禁止发送模块的音频通道中断 1: 允许发送模块的音频通道中断

25.7.2 状态寄存器 (STAT)

位:	31	30	29	28	27	26	25	24
	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位:	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	CMD
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位:	15	14	13	12	11	10	9	8
	RIS	TIS	UBO	UBU	CE	PARE	PREE	CSE
初始值:	1	1	0	0	0	0	0	0
R/W:	R	R	R/WC0	R/WC0	R/WC0	R/WC0	R/WC0	R/WC0

位:	7	6	5	4	3	2	1	0
	ABO	ABU	RUIR	TUIR	CSRX	CBRX	CSTX	CBTX
初始值:	0	0	0	0	0	0	0	0
R/W:	R/WC0	R/WC0	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 17	—	全 0	R	保留位
16	CMD	0	R	压缩模式数据 当接收数据为压缩模式数据 (V 标志和通道状态的 bit1 都为 “1”) 时, 将此位置位。 0: 不是压缩模式数据 1: 是压缩模式数据
15	RIS	1	R	接收模块空闲状态 当接收模块为空闲状态时, 将此位置位。 0: 接收模块不是空闲状态 1: 接收模块是空闲状态
14	TIS	1	R	发送模块空闲状态 当发送模块为空闲状态时, 将此位置位。 0: 发送模块不是空闲状态 1: 发送模块是空闲状态
13	UBO	0	R/WC0	用户缓冲器超限 * 如果接收模块的用户缓冲器发生超限, 就将此位置位; 如果给此位写 “0”, 就清除此位。如果控制寄存器的 REIE 位和 UBOI 位都被置位, 就产生中断。 0: 用户缓冲器未发生超限 1: 用户缓冲器发生超限
12	UBU	0	R/WC0	用户缓冲器欠载 * 如果发送模块的用户缓冲器发生欠载, 就将此位置位; 如果给此位写 “0”, 就清除此位。如果控制寄存器的 TEIE 位和 UBUI 位都被置位, 就产生中断。 0: 用户缓冲器未发生欠载 1: 用户缓冲器发生欠载

位	位名	初始值	R/W	说明
11	CE	0	R/WC0	时钟错误 * 如果时钟恢复不同步, 就将此位置位; 如果给此位写“0”, 就清除此位。 如果控制寄存器的 REIE 位和 CREI 位都被置位, 就产生中断。 0: 时钟恢复正常 1: 时钟恢复错误
10	PARE	0	R/WC0	奇偶校验错误 * 如果奇偶校验结果发生错误, 就将此位置位; 如果给此位写“0”, 就清除此位。如果控制寄存器的 REIE 位和 PAEI 位都被置位, 就产生中断。 0: 奇偶校验正常 1: 奇偶校验错误
9	PREE	0	R/WC0	前同步码错误 * 如果出现未定义的前同步码或者前同步码没有出现在正确的位置, 就将此位置位; 如果给此位写“0”, 就清除此位。如果控制寄存器的 REIE 位和 PREI 位都被置位, 就产生中断。 【注】 只能在块开始的前同步码之后将此位置位。 0: 前同步码出现在正确的位置 1: 前同步码错误
8	CSE	0	R/WC0	通道状态错误 * 如果在发送状态的第 32 帧前写通道状态信息, 就将此位置位; 如果给此位写“0”, 就清除此位。如果控制寄存器的 TEIE 位和 CSEI 位都被置位, 就产生中断。 0: 通道状态正常 1: 通道状态错误
7	ABO	0	R/WC0	音频缓冲器超限 * 表示接收模块的音频缓冲器在第 1 阶段和第 2 阶段都为满的状态下重写数据。如果给此位写“0”, 就清除此位。如果控制寄存器的 REIE 位和 ABOI 位都被置位, 就产生中断。 0: 接收模块的音频缓冲器未发生超限 1: 接收模块的音频缓冲器发生超限
6	ABU	0	R/WC0	音频缓冲器欠载 * 表示发送模块的音频缓冲器在第 1 阶段和第 2 阶段都为空状态下重复传送最后的数据; 如果给此位写“0”, 就清除此位。如果控制寄存器的 TEIE 位和 ABUI 位都被置位, 就产生中断。 0: 发送模块的音频缓冲器未发生欠载 1: 发送模块的音频缓冲器发生欠载
5	RUIR	0	R	接收模块用户信息寄存器 表示接收模块用户信息寄存器的状态。如果读接收模块的用户寄存器, 就清除此位。如果控制寄存器的 RUII 位被置位, 就产生中断。 0: 接收模块用户信息寄存器空 1: 接收模块用户信息寄存器满
4	TUIR	0	R	发送模块用户信息寄存器 表示发送模块用户信息寄存器的状态。如果写发送模块的用户寄存器, 就清除此位。如果控制寄存器的 TUII 位被置位, 就产生中断。 0: 发送模块用户信息寄存器满 1: 发送模块用户信息寄存器空

位	位名	初始值	R/W	说明
3	CSRX	0	R	通道 1 和通道 2 的状态—接收模块 表示接收模块的通道状态寄存器的状态。如果读接收模块的通道状态寄存器，就清除此位。如果控制寄存器的 RCSI 位被置位，就产生中断。 0: 接收模块通道状态寄存器空 1: 接收模块通道状态寄存器满
2	CBRX	0	R	通道 1 和通道 2 的缓冲器—接收模块 表示接收模块音频通道寄存器的状态。如果读接收模块的音频通道寄存器，就清除此位。如果控制寄存器的 RCBI 位被置位，就产生中断。 0: 接收模块音频通道寄存器空 1: 接收模块音频通道寄存器满
1	CSTX	0	R	通道 1 和通道 2 的状态—发送模块 表示发送模块通道状态寄存器的状态。如果写发送模块通道状态寄存器，就清除此位。如果控制寄存器的 TCSI 位被置位，就产生中断。 0: 发送模块通道状态寄存器满 1: 发送模块通道状态寄存器空
0	CBTX	0	R	通道 1 和通道 2 的缓冲器—发送模块 表示发送模块音频通道寄存器的状态。如果写发送模块的音频通道寄存器，就清除此位。如果控制寄存器的 TCBI 位被置位，就产生中断。 0: 发送模块音频通道寄存器满 1: 发送模块音频通道寄存器空

【注】* 如果在 DMA 传送的过程中检测到错误位，就需要重新设定 DMA 传送。此时，必须将瑞萨 SPDIF 模块允许位（RME 位或者 TME 位）和 DMA 允许位（RDE 位或者 TDE 位）设定为“禁止”，并且在解除错误状态后重新设定直接存储器存取控制器。此后，能将模块允许位设定为“允许”，重新开始 DMA 传送。

25.7.3 发送模块通道 1 的音频寄存器（TLCA）

位:	31	30	29	28	27	26	25	24
	—	—	—	—	—	—	—	—
初始值:	—	—	—	—	—	—	—	—
R/W:	W	W	W	W	W	W	W	W
位:	23	22	21	20	19	18	17	16
	音频PCM数据							
初始值:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
位:	15	14	13	12	11	10	9	8
	音频PCM数据							
初始值:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
位:	7	6	5	4	3	2	1	0
	音频PCM数据							
初始值:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W

位	位名	初始值	R/W	说明
31 ~ 24	—	—	W	保留位
23 ~ 0	音频 PCM 数据	全 0	W	音频 PCM 数据 这是进行 PCM 编码后的音频数据（向 LSB 靠紧保存）。

25.7.4 发送模块通道 2 的音频寄存器 (TRCA)

位:	31	30	29	28	27	26	25	24
	—	—	—	—	—	—	—	—
初始值:	—	—	—	—	—	—	—	—
R/W:	W	W	W	W	W	W	W	W
位:	23	22	21	20	19	18	17	16
	音频PCM数据							
初始值:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
位:	15	14	13	12	11	10	9	8
	音频PCM数据							
初始值:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
位:	7	6	5	4	3	2	1	0
	音频PCM数据							
初始值:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W

位	位名	初始值	R/W	说明
31 ~ 24	—	—	W	保留位
23 ~ 0	音频 PCM 数据	全 0	W	音频 PCM 数据位 这是进行 PCM 编码后的音频数据（向 LSB 靠紧保存）。

25.7.5 发送模块 DMA 的音频数据寄存器 (TDAD)

位:	31	30	29	28	27	26	25	24
	—	—	—	—	—	—	—	—
初始值:	—	—	—	—	—	—	—	—
R/W:	W	W	W	W	W	W	W	W
位:	23	22	21	20	19	18	17	16
	音频PCM数据							
初始值:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
位:	15	14	13	12	11	10	9	8
	音频PCM数据							
初始值:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
位:	7	6	5	4	3	2	1	0
	音频PCM数据							
初始值:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W

位	位名	初始值	R/W	说明
31 ~ 24	—	—	W	保留位
23 ~ 0	音频 PCM 数据	全 0	W	音频 PCM 数据位 这是进行 PCM 编码后的音频数据（向 LSB 靠紧保存）。

25.7.6 发送用户数据寄存器 (TUI)

给此寄存器写副帧的 U 位数据。因为按副帧 1、副帧 2 的顺序发送 U 位数据，所以必须每 16 帧更新一次数据。有关用户字节的内容，请参照所用设备对应的用户信息规格。从 LSB 开始按顺序设定要发送的用户位。

位:	31	30	29	28	27	26	25	24
	用户字节4							
初始值:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
位:	23	22	21	20	19	18	17	16
	用户字节3							
初始值:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
位:	15	14	13	12	11	10	9	8
	用户字节2							
初始值:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
位:	7	6	5	4	3	2	1	0
	用户字节1							
初始值:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W

位	位名	初始值	R/W	说明
31 ~ 24	用户字节 4	全 0	W	保存 U 位的信息。
23 ~ 16	用户字节 3	全 0	W	
15 ~ 8	用户字节 2	全 0	W	
7 ~ 0	用户字节 1	全 0	W	

25.7.7 发送模块通道 1 的状态寄存器 (TLCS)

将要发送的通道状态信息保存到 30 位寄存器。每个通道的 1 帧的通道状态信息由 192 位构成，因为设定到以下寄存器的所需数据只有 30 位，所以在发送开始的 30 位数据后发送“0”。

位:	31	30	29	28	27	26	25	24
	—	—	CLAC[1:0]		FS[3:0]			
初始值:	—	—	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
位:	23	22	21	20	19	18	17	16
	CHNO[3:0]				SRCNO[3:0]			
初始值:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
位:	15	14	13	12	11	10	9	8
	CATCD[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
位:	7	6	5	4	3	2	1	0
	—	—	CTL[4:0]				—	
初始值:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W

位	位名	初始值	R/W	说明
31、30	—	—	W	保留位
29、28	CLAC[1:0]	全 0	W	时钟精度 00: 2 级 01: 1 级 10: 3 级 11: 保留
27 ~ 24	FS[3:0]	全 0	W	采样频率 (FS) 0000: 44.1kHz 0010: 48kHz 0011: 32kHz
23 ~ 20	CHNO[3:0]	全 0	W	通道号 0000: Don't care 0001: A (左通道) 0010: B (右通道) 0011: C
19 ~ 16	SRCNO[3:0]	全 0	W	发送源号 0000: Don't care 0001: 1 0010: 2 0011: 3
15 ~ 8	CATCD[7:0]	全 0	W	种类代码 (例) 00000000: 2 通道通用格式 00000001: 2 通道光盘 (IEC908) 00000010: 2 通道 PCM 编码器 / 解码器 00000011: 2 通道数字音频磁带录音机

位	位名	初始值	R/W	说明
7、6	—	全 0	W	保留位 写入值总是“0”。
5 ~ 1	CTL[4:0]	全 0	W	控制 从发送源复制控制位（参照 IEC60958 规格）。
0	—	0	W	保留位 写入值总是“0”。

25.7.8 发送模块通道 2 的状态寄存器（TRCS）

将要发送的通道状态信息保存到 30 位寄存器。每个通道的 1 帧的通道状态信息由 192 位构成，因为设定到以下寄存器的所需数据只有 30 位，所以在发送开始的 30 位数据后发送“0”。

位:	31	30	29	28	27	26	25	24
	—	—	CLAC[1:0]	FS[3:0]				
初始值:	—	—	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
位:	23	22	21	20	19	18	17	16
	CHNO[3:0]				SRCNO[3:0]			
初始值:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
位:	15	14	13	12	11	10	9	8
	CATCD[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
位:	7	6	5	4	3	2	1	0
	—	—	CTL[4:0]					—
初始值:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W

位	位名	初始值	R/W	说明
31、30	—	—	W	保留位
29、28	CLAC[1:0]	全 0	W	时钟精度 00: 2 级 01: 1 级 10: 3 级 11: 保留
27 ~ 24	FS[3:0]	全 0	W	采样频率 (FS) 0000: 44.1kHz 0010: 48kHz 0011: 32kHz
23 ~ 20	CHNO[3:0]	全 0	W	通道号 0000: Don't care 0001: A (左通道) 0010: B (右通道) 0011: C

位	位名	初始值	R/W	说明
19 ~ 16	SRCNO[3:0]	全 0	W	发送源号 0000: Don't care 0001: 1 0010: 2 0011: 3
15 ~ 8	CATCD[7:0]	全 0	W	种类代码 (例) 00000000: 2 通道通用格式 00000001: 2 通道光盘 (IEC908) 00000010: 2 通道 PCM 编码器 / 解码器 00000011: 2 通道数字音频磁带录音机
7、6	—	全 0	W	保留位 写入值总是“0”。
5 ~ 1	CTL[4:0]	全 0	W	控制 从发送源复制控制位 (参照 IEC60958 规格)。
0	—	0	W	保留位 写入值总是“0”。

25.7.9 接收模块通道 1 的音频寄存器 (RLCA)

位:	31	30	29	28	27	26	25	24
	—	—	—	—	—	—	—	—
初始值:	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R
位:	23	22	21	20	19	18	17	16
	音频PCM数据							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8
	音频PCM数据							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
位:	7	6	5	4	3	2	1	0
	音频PCM数据							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 24	—	—	R	保留位
23 ~ 0	音频 PCM 数据	全 0	R	音频 PCM 数据 这是进行 PCM 编码后的音频数据 (向 LSB 靠紧保存)。

25.7.10 接收模块通道 2 的音频寄存器 (RRCA)

位:	31	30	29	28	27	26	25	24
	—	—	—	—	—	—	—	—
初始值:	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R
位:	23	22	21	20	19	18	17	16
	音频PCM数据							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8
	音频PCM数据							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
位:	7	6	5	4	3	2	1	0
	音频PCM数据							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 24	—	—	R	保留位
23 ~ 0	音频 PCM 数据	全 0	R	音频 PCM 数据位 这是进行 PCM 编码后的音频数据 (向 LSB 靠紧保存)。

25.7.11 接收模块 DMA 的音频数据 (RDAD)

位:	31	30	29	28	27	26	25	24
	—	—	—	—	—	—	—	—
初始值:	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R
位:	23	22	21	20	19	18	17	16
	音频PCM数据							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8
	音频PCM数据							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
位:	7	6	5	4	3	2	1	0
	音频PCM数据							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 24	—	—	R	保留位
23 ~ 0	音频 PCM 数据	全 0	R	音频 PCM 数据位 这是进行 PCM 编码后的音频数据 (向 LSB 靠紧保存)。

25.7.12 接收用户数据寄存器 (RUI)

此寄存器保存由瑞萨 SPDIF 接收的副帧的 U 位数据。因为从 LSB 开始按副帧 1、副帧 2 的顺序保存 U 位信息，所以必须每 16 帧读一次数据。有关用户字节的内容，请参照所用设备对应的规格。

位:	31	30	29	28	27	26	25	24
	用户字节4							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
位:	23	22	21	20	19	18	17	16
	用户字节3							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8
	用户字节2							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
位:	7	6	5	4	3	2	1	0
	用户字节1							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 24	用户字节 4	全 0	R	保存 U 位的信息。
23 ~ 16	用户字节 3	全 0	R	
15 ~ 8	用户字节 2	全 0	R	
7 ~ 0	用户字节 1	全 0	R	

25.7.13 接收模块通道 1 的状态寄存器 (RLCS)

按照从块开始接收的副帧 1 的顺序，从寄存器的 LSB 位开始保存通道状态。通道状态的内容请参照 IEC-60958。

位:	31	30	29	28	27	26	25	24
	—	—	CLAC[1:0]		FS[3:0]			
初始值:	—	—	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
位:	23	22	21	20	19	18	17	16
	CHNO[3:0]				SRCNO[3:0]			
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8
	CATCD[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
位:	7	6	5	4	3	2	1	0
	—	—	CTL[4:0]				—	
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31、30	—	—	R	保留位
29、28	CLAC[1:0]	全 0	R	时钟精度 00: 2 级 01: 1 级 10: 3 级 11: 保留
27 ~ 24	FS[3:0]	全 0	R	采样频率 (fs) 0000: 44.1kHz 0010: 48kHz 0011: 32kHz
23 ~ 20	CHNO[3:0]	全 0	R	通道号 0000: Don't care 0001: A (左通道) 0010: B (右通道) 0011: C
19 ~ 16	SRCNO[3:0]	全 0	R	发送源号 0000: Don't care 0001: 1 0010: 2 0011: 3
15 ~ 8	CATCD[7:0]	全 0	R	种类代码 (例) 00000000: 2 通道通用格式 00000001: 2 通道光盘 (IEC908) 00000010: 2 通道 PCM 编码器 / 解码器 00000011: 2 通道数字音频磁带录音机
7、6	—	全 0	R	保留位

位	位名	初始值	R/W	说明
5 ~ 1	CTL[4:0]	全 0	R	控制位 从发送源复制控制位（参照 IEC60958 规格）。
0	—	0	R	保留位

25.7.14 接收模块通道 2 的状态寄存器（RRCS）

按照从块开始接收的副帧 2 的顺序，从寄存器的 LSB 位开始保存通道状态。通道状态的内容请参照 IEC-60958。

位:	31	30	29	28	27	26	25	24
	—	—	CLAC[1:0]		FS[3:0]			
初始值:	—	—	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
位:	23	22	21	20	19	18	17	16
	CHNO[3:0]				SRCNO[3:0]			
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8
	CATCD[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
位:	7	6	5	4	3	2	1	0
	—	—	CTL[4:0]				—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31、30	—	—	R	保留位
29、28	CLAC[1:0]	全 0	R	时钟精度 00: 2 级 01: 1 级 10: 3 级 11: 保留
27 ~ 24	FS[3:0]	全 0	R	采样频率 (fs) 0000: 44.1kHz 0010: 48kHz 0011: 32kHz
23 ~ 20	CHNO[3:0]	全 0	R	通道号 0000: Don't care 0001: A (左通道) 0010: B (右通道) 0011: C
19 ~ 16	SRCNO[3:0]	全 0	R	发送源号 0000: Don't care 0001: 1 0010: 2 0011: 3

位	位名	初始值	R/W	说明
15 ~ 8	CATCD[7:0]	全 0	R	种类代码 (例) 00000000: 2 通道通用格式 00000001: 2 通道光盘 (IEC908) 00000010: 2 通道 PCM 编码器 / 解码器 00000011: 2 通道数字音频磁带录音机
7、6	—	全 0	R	保留位
5 ~ 1	CTL[4:0]	全 0	R	控制位 从发送源复制控制位 (参照 IEC60958 规格)。
0	—	0	R	保留位

25.8 功能说明—发送模块

25.8.1 发送模块

发送模块将要发送的 PCM 数据和辅助信息进行编码，使之符合 IEC60958 规格（SPDIF）的双相传号方式的调制数据后进行发送。

发送模块的时钟是外部提供的过采样时钟。此时钟通常选择以双相传号方式编码时所需时钟频率的 8 倍频率进行过采样的值。此时，副帧内 32 个时隙的发送时所需的时钟频率为音频数据采样频率的 512 倍。

先将音频数据和通道状态信息写到模块的通道 1，然后写到通道 2。在一般情况下，只在信息发生变化时才需要写通道状态。因为 SPDIF 模块在 30 帧后（当前通道状态数据的发送全部结束时）请求写通道状态，所以必须在从 31 帧到下一块（第 192 帧）开始前进行写操作。

通过双缓冲结构保存音频数据。确认在双缓冲的第 1 阶段数据丢失的方法有 2 种，一种是发送中断请求的方法，另一种是读状态寄存器的方法。DMA 传送通过第 1 个请求发送通道 1 的音频数据，通过第 2 个请求发送通道 2 的音频数据。

给通道 1 和通道 2 的 30 位寄存器设定要发送的通道状态信息。在 1 块中，每个通道有 192 位通道状态信息，因为需要的数据只有 30 位，所以在瑞萨 SPDIF 模块发送开始的 30 位数据后连续发送“0”，直到块结束为止。

用户数据是 32 位双缓冲数组。通过发送中断请求或者读状态寄存器，确认双缓冲的第 1 阶段为空状态。一般情况下，块中的用户数据信息的数据长度不足，按照通道 1、通道 2 的顺序发送，在发送 384 位 1 块中的用户数据后，也继续发送下一块。

通过瑞萨 SPDIF 模块处理的音频数据为线性 PCM，能设定的最大长度是 24 位。因此，V 标志总是为“0”，表示音频数据为线性 PCM。不能通过寄存器设定 V 标志。串行输出数据的每 32 位（不包括前同步码）生成 1 个偶校验。

【注】 如果发送模块的用户缓冲器发生欠载，就在写下一个数据前发送 SPDIF 缓冲器中的当前数据。

25.8.2 发送模块的初始化

发送模块在复位后默认为空闲状态，也能通过给 CTRL 寄存器的 TME 位写“0”，转移到空闲状态。在空闲状态下，发送模块的设定如下：

- 将发送模块的空闲状态位（TIS）置“1”，而将其他状态位全部清“0”。
- 前同步码的生成无效。
- 将通道 1 和通道 2 的同步置“0”（通道 1：0，通道 2：1）。
- 字计数和帧计数变为“0”。
- 从双相传号方式的编码器输出“0”。

通道状态寄存器、用户数据寄存器和音频数据寄存器保持转移到空闲状态前的值。要从空闲状态返回时，必须给 CTRL 寄存器的 TME 位写“1”。

25.8.3 发送模块的初始设定

如果将 TME 位置“1”，就将 TUIR 位和 CSTX 位都置“1”。此时，如果按照 (1)TUI、(2)TLCS、TRCS 的顺序写数据，就发生通道状态错误。为了避免发生错误，必须按照 (1)TLCS、TRCS、(2)TUI 的顺序写数据。

在将 TME 位置“1”后，必须在写 TLCS、TRCS 和 TUI 后确认已清除 CSTX 位和 TUIR，然后写开始的音频数据（通过 CPU 的存取写 TLCA 和 TRCA，或者通过 DMA 传送写 TDAD）。

25.8.4 发送模块的数据传送

如果发送模块从空闲状态返回，就能进行数据传送。能通过中断、DMA 请求和读状态寄存器 3 种方法中的任意一种开始数据传送。有发送和接收的共用中断信号以及用于发送模块的独立的 DMA 请求信号。

使用中断的发送模块的数据传送如图 25.5 所示。

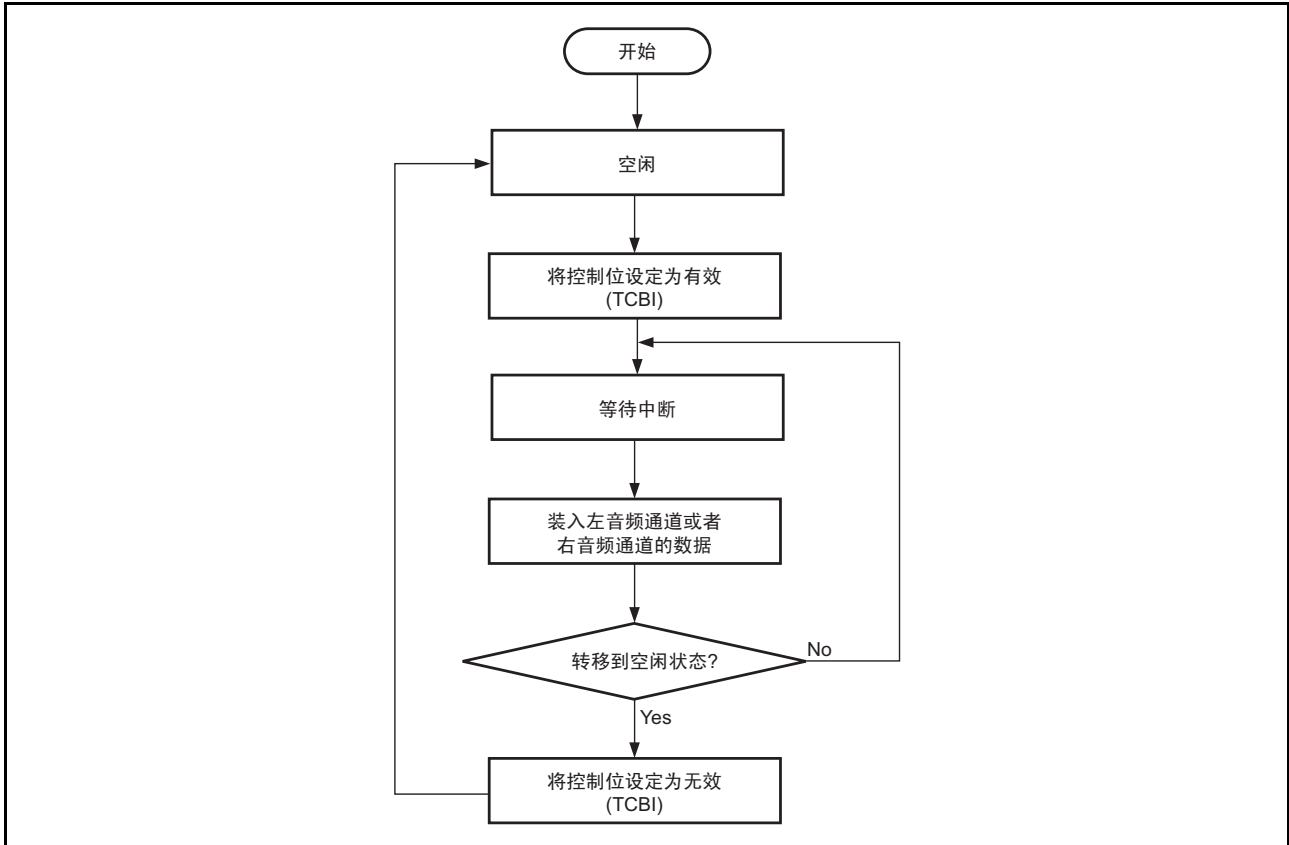


图 25.5 发送模块的数据传送流程（中断的方法）

使用 DMA 请求的发送模块的数据传送如图 25.6 所示。

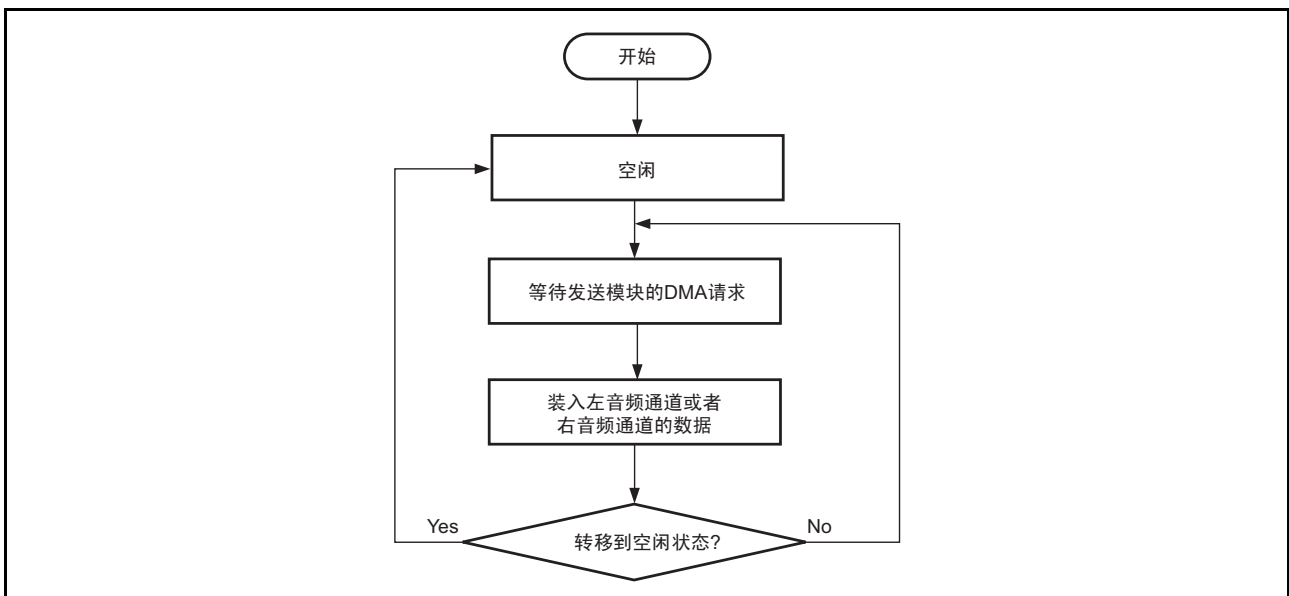


图 25.6 发送模块的数据传送流程（DMA 请求的方法）

在通道状态信息发生变化时更新该信息。在更新通道状态信息时，必须在发送下一个块前进行，所以必须在 30 帧发送结束后写要更新的通道状态。能通过中断通知 30 帧的发送结束或者通过读状态位来检测 30 帧的发送结束。如果在 30 帧发送结束前（正在发送当前的信息）写通道状态信息，状态寄存器的通道状态错误位（CSE）就被置位并且产生中断。

【注】 能通过 30 帧发送 1 个通道状态块的全部有效信息。

25.9 功能说明—接收模块

25.9.1 接收模块

接收模块对按 IEC60958 规格编码的输入进行数据和时钟的解调。解调数据为线性 PCM 格式的数据，保存在音频数据寄存器，将作为辅助信息同时接收的通道状态和用户信息保存到寄存器。

接收模块的主时钟是外部提供的过采样时钟。接收模块以过采样时钟的 4 倍频率运行。

【注】 发送模块和接收模块的过采样时钟相同。

使用脉宽计数器和平均滤波器进行时钟恢复，在输入数据各位的中间点产生采样脉冲。时钟错误状态位表示和时钟不同步。当输入数据中出现第一个前同步码时，就取得同步。能通过继续调整，抑制抖动和时钟变动在时钟恢复规定范围的期间对该时钟恢复的影响。

如果时钟恢复成功，双相传号方式的解码器就开始检测前同步码。解码器检索块开始的前同步码（参照表 25.2）。前同步码错误状态位表示下一个前同步码没有出现在正确的时序中，这有可能是传送失败或者干扰引起的。

对解码的数据进行偶校验。如果不同，奇偶校验错误状态位就被置位。

SPDIF 模块取出音频数据以外的用户数据和通道状态信息。通过双缓冲结构保存音频数据。能通过发送缓冲器满状态引起的中断请求或者读状态位，检测到能读的数据。DMA 传送通过第 1 个请求传送通道 1 的音频数据，通过第 2 个请求传送通道 2 的音频数据。

将通道状态信息保存到 30 位寄存器。因为以每 1 个副帧 1 位的形式接收通道状态信息，所以在接收到各通道共 30 帧前，不结束寄存器的保存。将新的通道状态信息和当前数据进行比较，只有数据发生变化时，CPU 才读新的信息。同时也接收用户数据，因为用户数据按副帧保存到寄存器，所以在接收到 16 帧后结束。

【注】 1. 不能对通道状态信息数据请求进行 DMA 传送。

2. 如果接收模块的用户缓冲器发生超限，就用从 SPDIF 接口输入的下一个数据重写 SPDIF 缓冲器的当前数据。

25.9.2 接收模块的初始化

接收模块在复位后默认为空闲状态，也能通过给 CTRL 寄存器的 RME 位写“0”，转移到空闲状态。在空闲状态下，接收模块的设定如下：

- 将接收模块的空闲状态位置“1”，而将其他状态位全部清“0”。
- 将通道1和通道2的同步置“0”（通道1:0，通道2:1）。
- 字计数和帧计数变为“0”。

通道状态寄存器、用户数据寄存器和音频数据寄存器保持转移到空闲状态前的值。要从空闲状态返回时，必须给 CTRL 寄存器的 RME 位写“1”。

25.9.3 接收模块的数据传送

如果接收模块从空闲状态返回，就能进行数据传送。能通过中断、读状态寄存器和 DMA 请求 3 种方法中的任意一种开始数据传送。有发送和接收的共用中断信号和用于接收模块的独立的 DMA 请求信号。如果在将数据传送到接收模块的过程中因以下源而产生错误信号，就可能产生中断。

1. 时钟恢复错误
2. 传送错误或者障碍—通过前同步码错误进行通知
3. 奇偶校验错误

在传送错误或者障碍的情况下，副帧开始或者块开始的前同步码的位置发生偏差或者不出现。在奇偶校验位出错时，发生奇偶校验错误。上述的任何原因都有可能引起这种错误。

- 时钟恢复的容许偏差
用以下计算式表示时钟恢复的接收余量。

$$M = \left| \left(0.5 - \frac{1}{2N} \right) - (L - 0.5) F - \frac{|D - 0.5|}{N} (1 + F) \right| \times 100\%$$

M : 接收余量

N : 过采样速度

L : 帧长 =33

D : 占空比周期 =0.6

F : 过采样时钟偏差 =II 级精度 =1000×10e⁻⁶

接收余量 M 如图 25.7 所示。

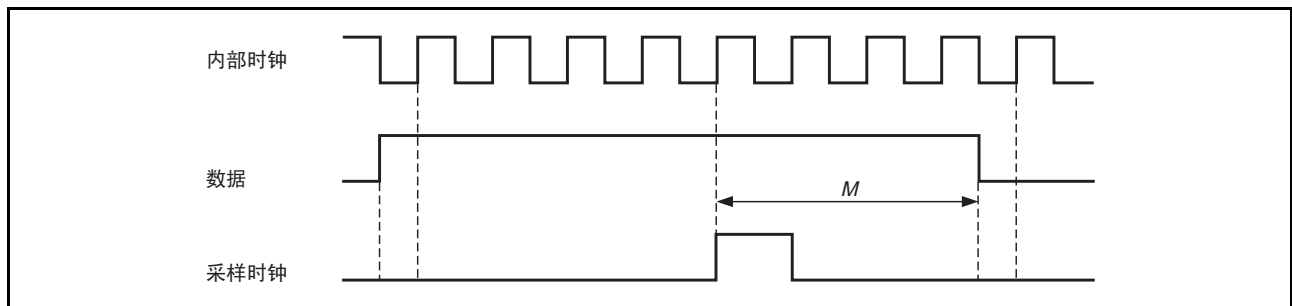


图 25.7 接收余量

在上述计算式中导入偏差时，用以下不等式表示。

$$j \leq \left| \left(0.5 - \frac{1}{2N} \right) - (L - 0.5) F - \frac{|D-0.5|}{N} (1 + F) \right| \times 100\%$$

J : 时钟偏差

8 倍过采样速度: 接收余量 = 39.25%

4 倍过采样速度: 接收余量 = 31.75%

2 倍过采样速度: 接收余量 = 16.75%

最大采样频率为 48kHz。此时，需要的时钟频率为 $128 \times 48\text{kHz} = 6.144\text{MHz}$ 。1 个周期中最坏情况的偏差规定为 40ns（周期的 24.5%）。因此，如果是大于等于 4 的过采样速度，就满足上述不等式。

使用中断的接收模块的数据传送如图 25.8 所示。

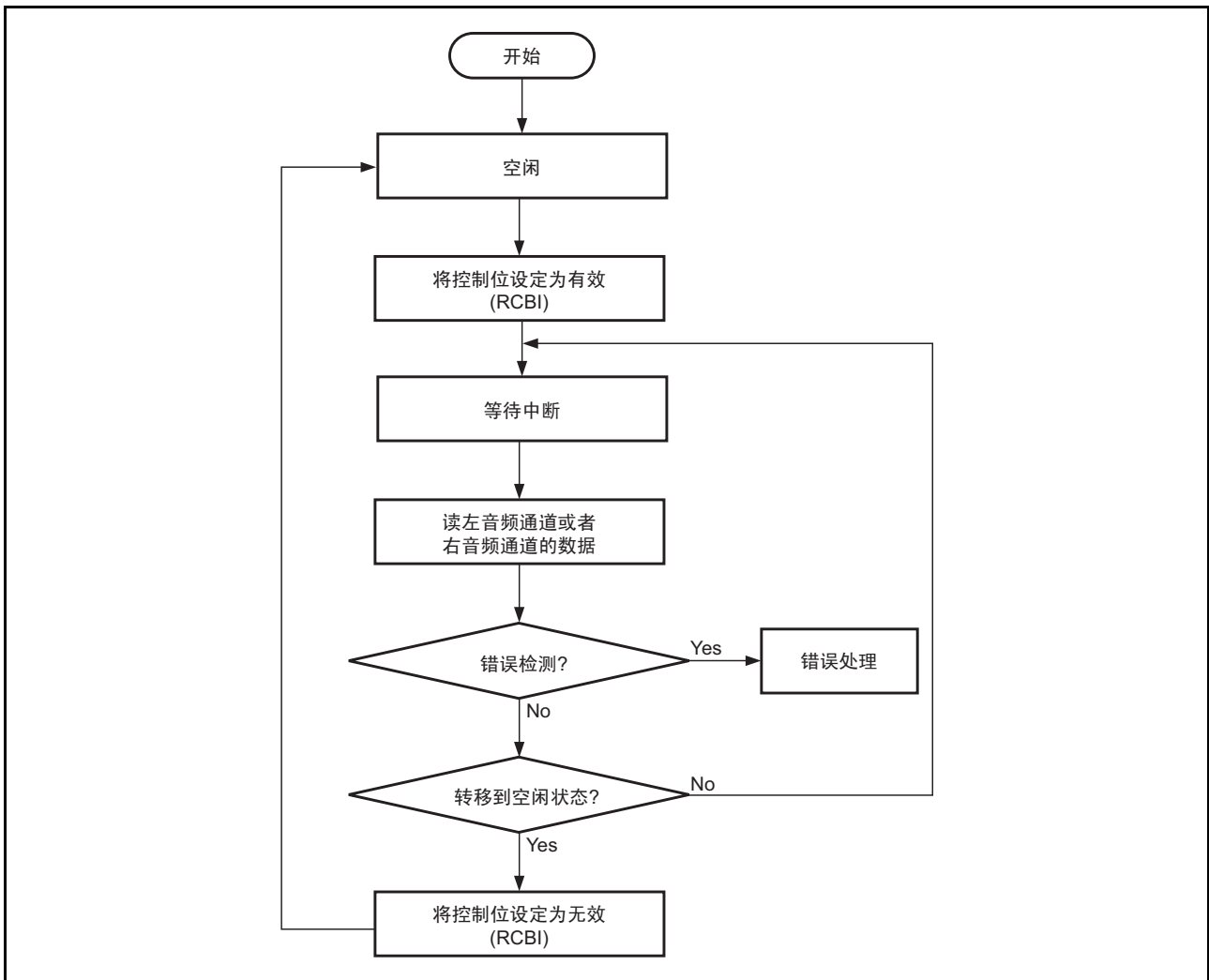


图 25.8 接收模块的数据传送流程（中断的方法）

只在第 30 帧接收结束后信息发生变化时，才产生通知信息已保存到通道状态信息寄存器的中断。

25.10 模块的停止

25.10.1 发送模块和接收模块的空闲状态

能通过给控制寄存器的空闲位（发送模块为 TME 位，接收模块为 RME 位）写“0”，将发送模块和接收模块的运行置为无效。能通过读状态寄存器的空闲位（发送模块为 TIS 位，接收模块为 RIS 位），检测到此空闲状态。

25.11 压缩模式的数据

IEC61937 规格中定义了压缩模式的数据。SPDIF 模块只检测压缩模式数据。检查有效性标志（V 标志）和通道状态数据的 bit1，在两者都为“1”的情况下，作为压缩模式数据设定到状态寄存器的 CMD 位。

【注】 只有接收模块检测压缩模式数据。此信息与发送模块无关。

25.12 参考

IEC60958 数字音频接口

IEC61937 压缩模式数字音频接口

25.13 使用时的注意事项

25.13.1 TUIR 的清除

在写 TUI 后到清除 TUIR 前，需要等待最大 1 帧的发送的结束。如果通过发送模块用户信息中断将数据写到 TUI，为了不再次错误地接收中断，必须在确认 TUIR 已被清除后结束中断处理程序。

25.13.2 音频输入时钟的频率

输入到 AUDIO X1、AUDIO X2 或者 AUDIO CLK 的时钟频率必须低于 P1 ϕ 的频率。

26. CD-ROM 解码器

CD-ROM 解码器对 CD-DSP 送来的流数据进行 ROM 解码处理。在 CD-DA*1 情况下，流数据为 PCM 数据，因此不输入到 CD-ROM 解码器；在 CD-ROM*2 情况下，输入流数据，进行同步码的检测和保护、解扰、ECC 校正、EDC 校验，并且输出流数据。

但是，流数据是以通过串行音频接口传送 CD-ROM 数据为前提，因此不进行 CD 规格的子代码处理。

【注】 *1 符合 JIS S 8605 (Red Book)。

*2 符合 JIS X 6281 (Yellow Book)。

26.1 特点

- 同步码的检测和保护
检测 CD-ROM 同步码。在因划伤等检测不到同步码时，保护同步码（自动插入）。
保护的种类有自动同步保护模式、外部同步保护模式、内插同步模式和内插/外部同步保护模式。
- 解扰
- ECC 校正
能进行 P 校正、Q 校正、PQ 校正和 QP 校正。
能重复进行 PQ 校正和 QP 校正，最多能校正 3 次。但是，校正次数取决于 CD 的倍速。例如，如果 CD-ROM 解码器的工作频率为 60MHz 并且为 CD2 倍速，就最多能重复校正 3 次。
ECC 校正有 2 面缓冲器，能在用 1 面接收流数据的同时用另 1 面进行 ECC 校正。
- EDC 校验
在 ECC 校正前后进行 EDC 校验。当 ECC 前的 EDC 校验没有错误时，与校正子计算结果无关，有不进行 ECC 校正的模式。
- 缓冲数据控制
CD-ROM 解码器通过以同步码开头的特定格式将数据输出到缓冲区。

26.1.1 数据格式

本 LSI 支持图 26.1 的 5 种格式。

Mode0	同步 (12字节)	标头 (4字节)	All 0			
Mode1	同步 (12字节)	标头 (4字节)	数据(2048字节)	EDC (4字节)	0 (8字节)	P 奇偶校验 (172字节) Q 奇偶校验 (104字节)
Mode2 (notXA)	同步 (12字节)	标头 (4字节)	数据(2336字节)			
Mode2 Form1	同步 (12字节)	标头 (4字节)	副标头 (8字节)	数据(2048字节)	EDC (4字节)	P 奇偶校验 (172字节) Q 奇偶校验 (104字节)
Mode2 Form2	同步 (12字节)	标头 (4字节)	副标头 (8字节)	数据(2324字节)	EDC (4字节)	

图 26.1 数据格式

26.2 框图

用于实现 CD-ROM 解码功能的本 LSI 的 CD-ROM 解码器功能和连接外围总线的总线桥如图 26.2 所示。

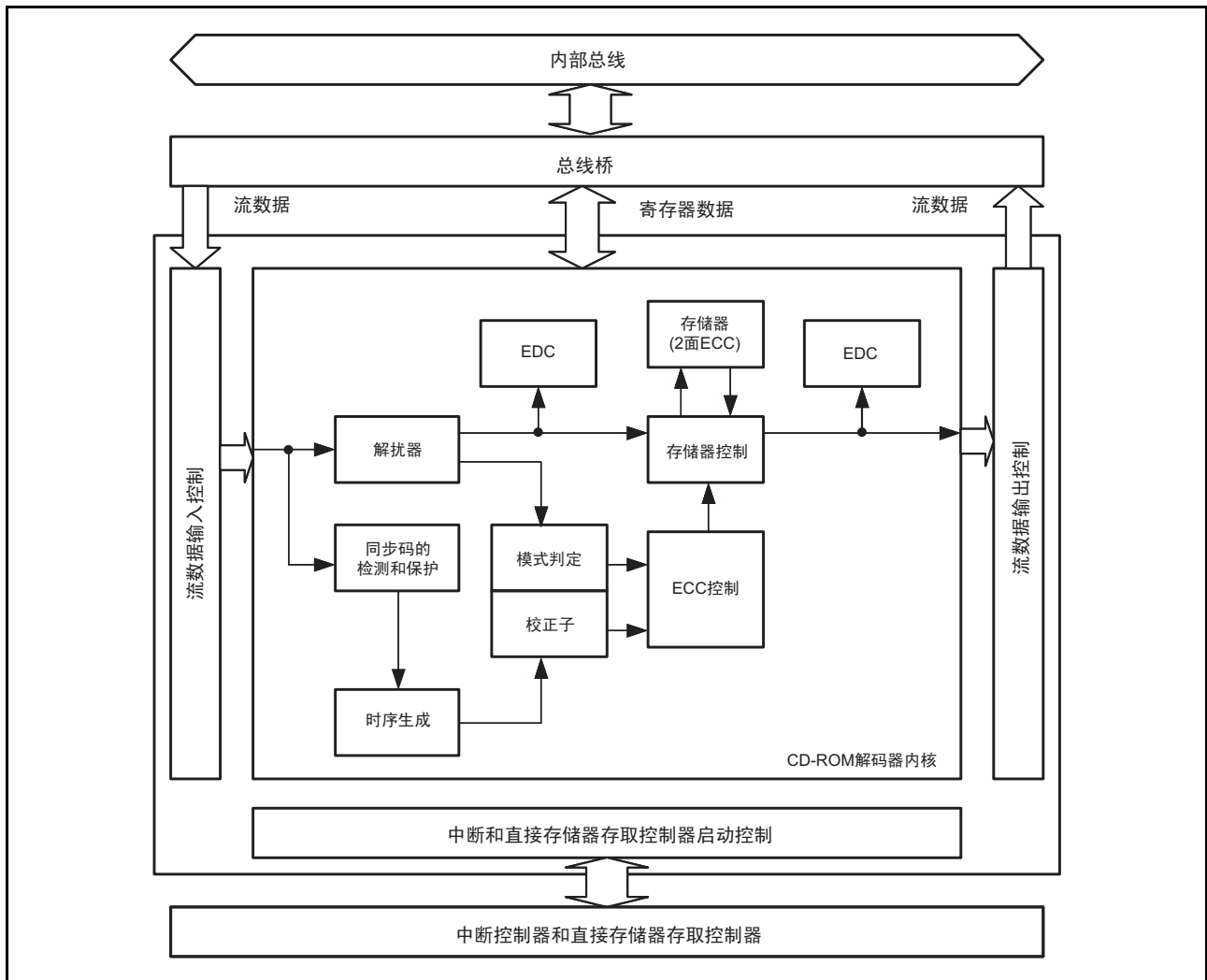


图 26.2 框图

CD-ROM 解码器内核部执行 CD-ROM 解码所需的解扰功能、同步码检测、ECC (P、Q 校正) 功能和 EDC 校验功能。存储器有 2 个扇区。

内部总线的数据为输入和输出各 1 个系统，通过总线桥逻辑，分为寄存器存取端口和流数据端口。

CD-DSP 的流数据通过串行音频接口传送到流数据输入控制块。然后，经过 CD-ROM 解码器，在进行解扰、ECC 校正、EDC 校验后，得到 1 个扇区的数据。之后，通过流数据输出控制块，传送到流数据缓冲器。流数据的传送有通过直接存储器存取控制器传送的方法和通过 CPU 传送的方法。

总线桥的框图如图 26.3 所示。

因为从串行音频接口传送输入流数据，所以传送较慢，但是由于输出流数据已存在于 CD-ROM 解码器内核部内，因此能进行高速传送。另外，输出流数据通过 SDRAM 等进行缓冲，为了降低 SDRAM 的繁忙率，需要进行高速传送。因此，在内部总线的输出流数据读请求被传送到 CD-ROM 解码器前，预先读数据并且将流数据存储在总线桥内的寄存器，如果发生内部总线的请求，就立即变为能将数据输出到内部总线的状态。然而，在将预先读到的输出流数据保存到寄存器的状态下，也可能发生其他寄存器的读请求。为此，此模块分别有用于输出流数据的寄存器和用于读其他寄存器的中继寄存器。

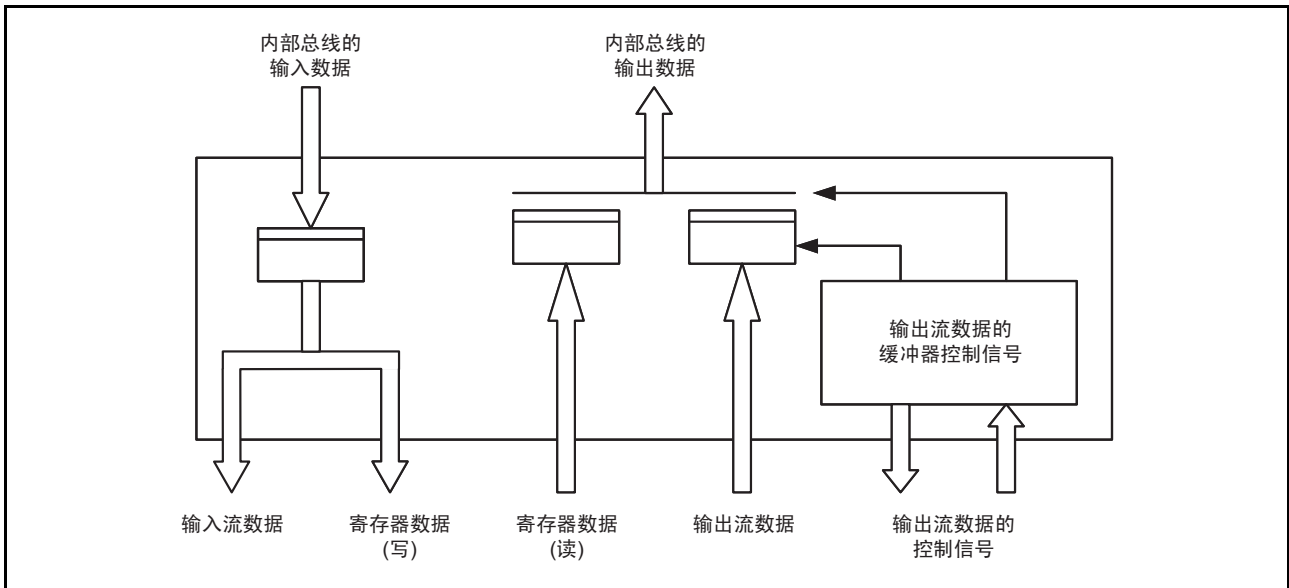


图 26.3 总线桥的框图

流数据的输入控制框图如图 26.4 所示。流数据的输入控制部有输入流数据的控制逻辑和改变 CD-ROM 解码器控制模式的寄存器。

因为字节序因串行音频接口的模式而不同，所以从串行音频接口送来的数据会改变传送顺序或者会传送填充数据。为了对应各种数据，准备了改变运行模式的寄存器以及生成控制 CD-ROM 解码器内核部的控制信号。输入流数据的保持寄存器有 2 面 16 位寄存器。通过寄存器设定的模式，既能将从串行音频接口先送来的 16 位数据先提供给 CD-ROM 解码器内核部，也能将后送来的 16 位数据先提供给 CD-ROM 解码器内核部，而且也能不给 CD-ROM 解码器内核部提供填充数据。

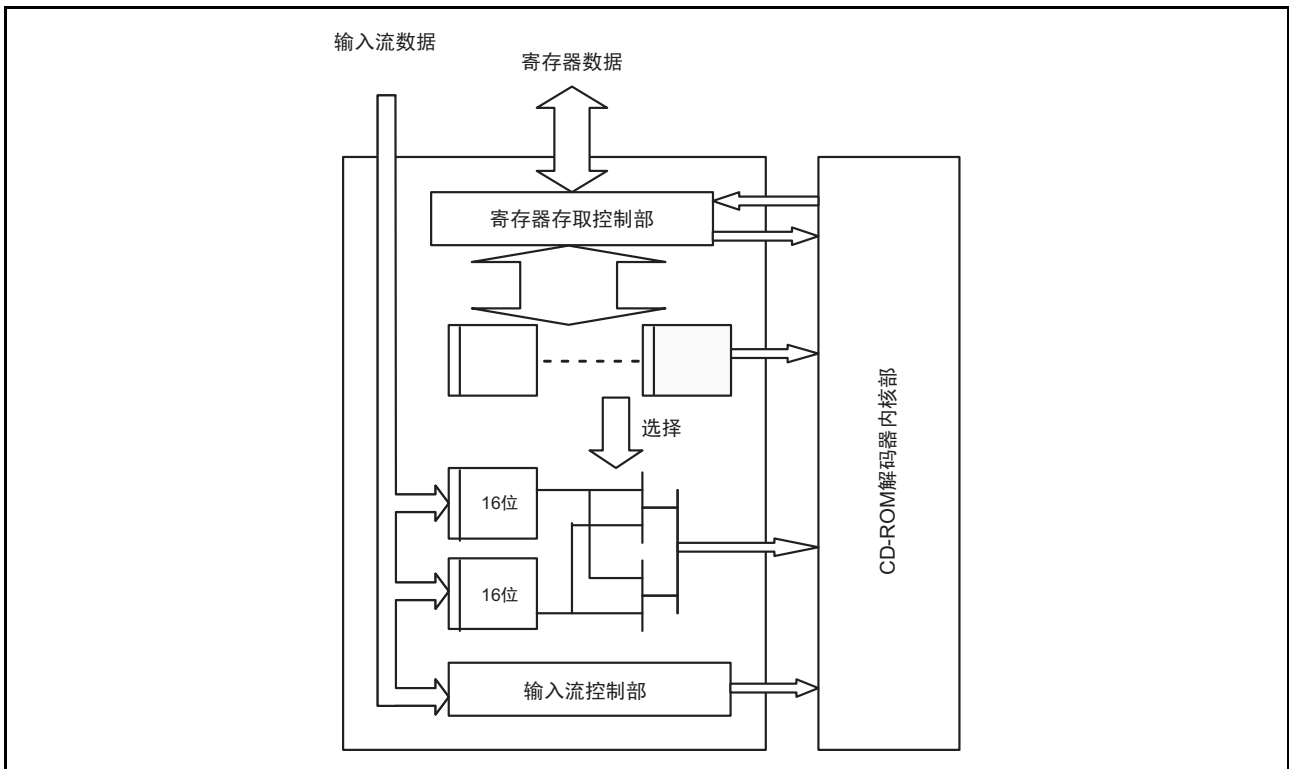


图 26.4 流数据的输入控制框图

流数据的输出控制框图如图 26.5 所示。

在识别出 CD-ROM 解码器内核部准备好 1 个扇区的 CD-ROM 数据并且确认了总线桥部的输出流数据寄存器为空后，从 CD-ROM 解码器内核部取输出流数据。

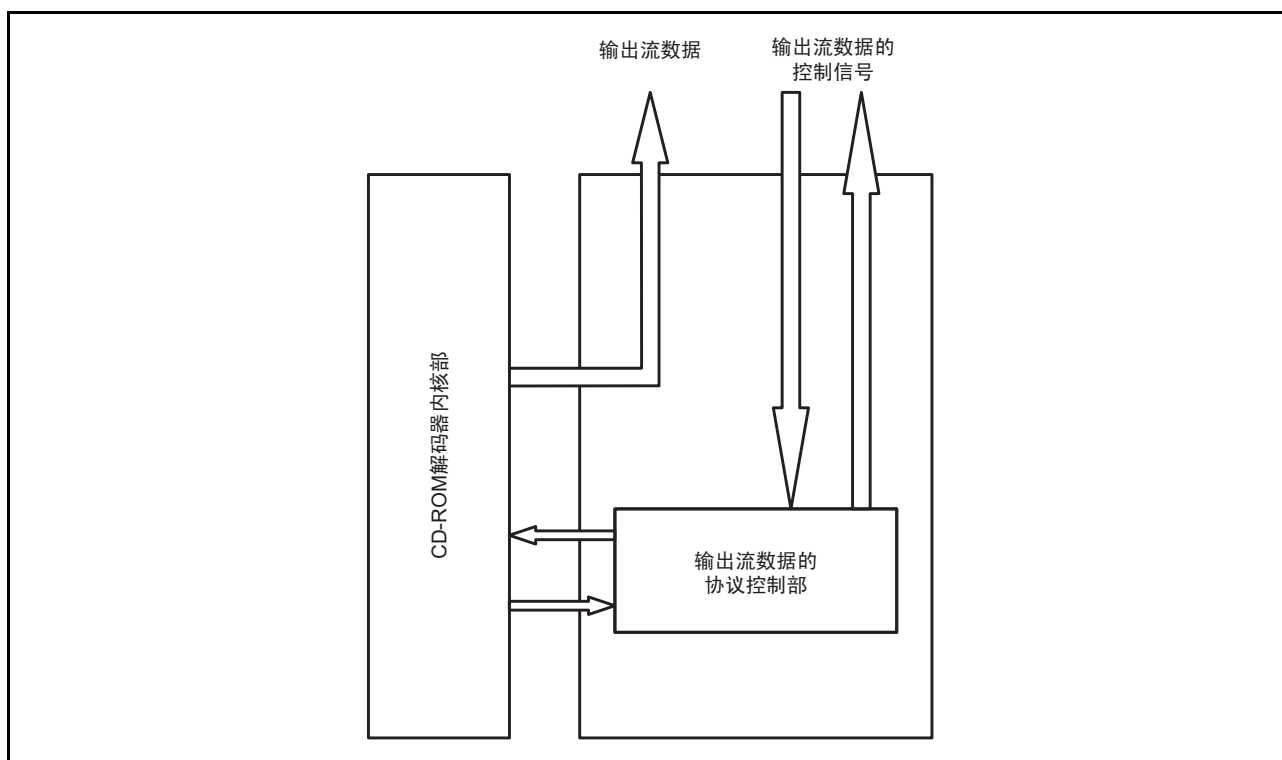


图 26.5 流数据的输出控制框图

中断和直接存储器存取控制器的启动控制有中断的保留和标志清除功能、中断屏蔽功能、直接存储器存取控制器的启动信号有效功能以及检测传送数据量并且将启动信号置为无效的功能。

26.3 寄存器说明

寄存器结构如表 26.1 所示。

表 26.1 寄存器结构

寄存器名	略称	R/W	初始值	地址	存取长度
允许控制寄存器	CROMEN	R/W	H'00	H'E8005000	8
同步码的同步控制寄存器	CROMSY0	R/W	H'89	H'E8005001	8
解码模式控制寄存器	CROMCTL0	R/W	H'82	H'E8005002	8
EDC/ECC 校验控制寄存器	CROMCTL1	R/W	H'D1	H'E8005003	8
解码处理自动停止控制寄存器	CROMCTL3	R/W	H'00	H'E8005005	8
解码选项设定控制寄存器	CROMCTL4	R/W	H'00	H'E8005006	8
HEAD20 ~ 22 表示控制寄存器	CROMCTL5	R/W	H'00	H'E8005007	8
同步码状态寄存器	CROMST0	R	H'00	H'E8005008	8
ECC 后的标头错误状态寄存器	CROMST1	R	H'00	H'E8005009	8
ECC 后的副标头错误状态寄存器	CROMST3	R	H'00	H'E800500B	8
标头 / 副标头数据有效判定状态寄存器	CROMST4	R	H'00	H'E800500C	8
模式判定结果和链路扇区检测状态寄存器	CROMST5	R	H'00	H'E800500D	8
ECC/EDC 错误状态寄存器	CROMST6	R	H'00	H'E800500E	8
缓冲状态寄存器	CBUFST0	R	H'00	H'E8005014	8
解码中止源状态寄存器	CBUFST1	R	H'00	H'E8005015	8
缓冲器上溢状态寄存器	CBUFST2	R	H'00	H'E8005016	8
ECC 校正前标头部 -MINUTES 数据寄存器	HEAD00	R	H'00	H'E8005018	8
ECC 校正前标头部 -SECONDS 数据寄存器	HEAD01	R	H'00	H'E8005019	8
ECC 校正前标头部 -FRAMES (1/75 秒) 数据寄存器	HEAD02	R	H'00	H'E800501A	8
ECC 校正前标头部 -MODE 数据寄存器	HEAD03	R	H'00	H'E800501B	8
ECC 校正前副标头部 - 文件号 (BYTE-16) 数据寄存器	SHEAD00	R	H'00	H'E800501C	8
ECC 校正前副标头部 - 通道号 (BYTE-17) 数据寄存器	SHEAD01	R	H'00	H'E800501D	8
ECC 校正前副标头部 - 子模式 (BYTE-18) 数据寄存器	SHEAD02	R	H'00	H'E800501E	8
ECC 校正前副标头部 - 数据类型 (BYTE-19) 数据寄存器	SHEAD03	R	H'00	H'E800501F	8
ECC 校正前副标头部 - 文件号 (BYTE-20) 数据寄存器	SHEAD04	R	H'00	H'E8005020	8
ECC 校正前副标头部 - 通道号 (BYTE-21) 数据寄存器	SHEAD05	R	H'00	H'E8005021	8
ECC 校正前副标头部 - 子模式 (BYTE-22) 数据寄存器	SHEAD06	R	H'00	H'E8005022	8
ECC 校正前副标头部 - 数据类型 (BYTE-23) 数据寄存器	SHEAD07	R	H'00	H'E8005023	8
ECC 校正后标头部 -MINUTES 数据寄存器	HEAD20	R	H'00	H'E8005024	8

寄存器名	略称	R/W	初始值	地址	存取长度
ECC 校正后标头部 -SECONDS 数据寄存器	HEAD21	R	H'00	H'E8005025	8
ECC 校正后标头部 -FRAMES (1/75 秒) 数据寄存器	HEAD22	R	H'00	H'E8005026	8
ECC 校正后标头部 -MODE 数据寄存器	HEAD23	R	H'00	H'E8005027	8
ECC 校正后副标头部 - 文件号 (BYTE-16) 数据寄存器	SHEAD20	R	H'00	H'E8005028	8
ECC 校正后副标头部 - 通道号 (BYTE-17) 数据寄存器	SHEAD21	R	H'00	H'E8005029	8
ECC 校正后副标头部 - 子模式 (BYTE-18) 数据寄存器	SHEAD22	R	H'00	H'E800502A	8
ECC 校正后副标头部 - 数据类型 (BYTE-19) 数据寄存器	SHEAD23	R	H'00	H'E800502B	8
ECC 校正后副标头部 - 文件号 (BYTE-20) 数据寄存器	SHEAD24	R	H'00	H'E800502C	8
ECC 校正后副标头部 - 通道号 (BYTE-21) 数据寄存器	SHEAD25	R	H'00	H'E800502D	8
ECC 校正后副标头部 - 子模式 (BYTE-22) 数据寄存器	SHEAD26	R	H'00	H'E800502E	8
ECC 校正后副标头部 - 数据类型 (BYTE-23) 数据寄存器	SHEAD27	R	H'00	H'E800502F	8
自动缓冲设定控制寄存器	CBUFCTL0	R/W	H'04	H'E8005040	8
自动缓冲起始扇区设定 -MINUTES 控制寄存器	CBUFCTL1	R/W	H'00	H'E8005041	8
自动缓冲起始扇区设定 -SECONDS 控制寄存器	CBUFCTL2	R/W	H'00	H'E8005042	8
自动缓冲起始扇区设定 -FRAMES 控制寄存器	CBUFCTL3	R/W	H'00	H'E8005043	8
ISY 中断源屏蔽控制寄存器	CROMST0M	R/W	H'00	H'E8005045	8
CD-ROM 解码器模块复位控制寄存器	ROMDECRST	R/W	H'00	H'E8005100	8
CD-ROM 解码器模块复位状态寄存器	RSTSTAT	R	H'00	H'E8005101	8
串行音频接口数据控制寄存器	SSI	R/W	H'18	H'E8005102	8
中断标志寄存器	INTHOLD	R/W	H'00	H'E8005108	8
中断源屏蔽控制寄存器	INHINT	R/W	H'00	H'E8005109	8
CD-ROM 解码器流数据输入寄存器	STRMDIN0	R/W	H'0000	H'E8005200	读: 16 写: 16、32
CD-ROM 解码器流数据输入寄存器	STRMDIN2	R/W	H'0000	H'E8005202	16
CD-ROM 解码器流数据输出寄存器	STRMDOUT0	R	H'0000	H'E8005204	16、32

26.3.1 允许控制寄存器 (CROMEN)

CROMEN 允许 Subcode 处理和 CD-ROM 解码处理以及强制结束 CD-ROM 解码处理。

位:	7	6	5	4	3	2	1	0
	SUBC EN	CROM_ EN	CROM_ STP	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	SUBC_EN	0	R/W	Subcode 处理允许 必须将此位和 CROM_EN 位同时置位和清除。在异常自动停止并且 CROM_STP 位为“1”时，此位自动变为“0”。
6	CROM_EN	0	R/W	CD-ROM 解码处理允许 如果将此位置“1”，就在检测到有效的同步码后开始 CD-ROM 解码处理。如果将此位置“0”，就在当前正在解码的扇区处理结束时中止解码处理。 在通过自动解码停止功能停止并且 CROM_STP 位为“1”时，此位自动变为“0”。
5	CROM_STP	0	R/W	CD-ROM 解码处理强制结束 如果将此位置“1”，就立即停止 CD-ROM 解码处理。 SUBC_EN 位和 CROM_EN 位自动变为“0”。在重新开始解码处理前，必须将此位置“0”。
4~0	—	全 0	R/W	保留位 读写值都为“0”。

26.3.2 同步码的同步控制寄存器 (CROMSY0)

CROMSY0 选择同步码保护功能。

位:	7	6	5	4	3	2	1	0
	SY AUT	SY IEN	SY DEN	—	—	—	—	—
初始值:	1	0	0	0	1	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	SY_AUT	1	R/W	CD-ROM 同步码的自动同步保护模式 如果将此位置“1”，就自动进行 CD-ROM 同步码的同步保护（插入同步码）。当此位为“1”时，SY_IEN 位和 SY_DEN 位的设定无效。
6	SY_IEN	0	R/W	内部同步信号允许 将 CD-ROM 解码器内部计数器生成的内部同步信号置为有效。 当 SY_AUT 位为“0”时，如果将此位置“1”，CD-ROM 数据的同步总是为通过内部计数器进行的内插模式。
5	SY_DEN	0	R/W	和同步信号的外部同步码同步 随时监视从输入数据检测到的同步码，与内部计数器的值无关，总是同步。 此位的设定在 SY_AUT 位为“0”时有效。
4	—	0	R/W	保留位 读写值都为“0”。
3	—	1	R/W	保留位 读写值都为“1”。
2、1	—	全 0	R/W	保留位 读写值都为“0”。
0	—	1	R/W	保留位 读写值都为“1”。

表 26.2 同步码保护功能的设定

SY_AUT	SY_IEN	SY_DEN	运行模式
1	—	—	自动同步保护模式
0	0	1	外部同步模式
0	1	0	内插同步模式
0	1	1	内插 / 外部同步模式
0	0	0	禁止设定

26.3.3 解码模式控制寄存器 (CROMCTL0)

CROMCTL0 控制各种功能，选择 Mode 判定或者 Form 判定并且设定扇区类型。此寄存器的设定在切换扇区时有效。

位:	7	6	5	4	3	2	1	0
	MD_DESC	—	MD_AUTO	MD_AUTOS1	MD_AUTOS2	MD_SEC[2:0]		
初始值:	1	0	0	0	0	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	MD_DESC	1	R/W	解扰功能控制 0: 解扰功能 OFF 1: 解扰功能 ON
6	—	0	R/W	保留位 读写置都为“0”。
5	MD_AUTO	0	R/W	Mode Form 的自动检测功能控制 0: OFF 1: ON 能检测的格式为 Mode0、Mode1、Mode2 (not XA)、Mode2 Form1 和 Mode2 Form2。当无法检测到 Mode Form 时，继续前一个扇区的 Mode Form。当无法检测到解码开始的第一个扇区的 Mode Form 时，将 MD_SEC[2:0] 的设定值用作初始值。
4	MD_AUTOS1	0	R/W	MD_AUTO 位为“1”时的 Mode 判定基准 0: 只在检测到同步码时进行 Mode 判定 1: 随时进行 Mode 判定 此位的设定只在 MD_AUTO 位为“1”时有效。当无法判定时，继续前一个扇区的 Mode。如果将此位置“0”，就只在检测到对应扇区的同步码时进行 Mode 判定。
3	MD_AUTOS2	0	R/W	MD_AUTO 位为“1”时的 Mode2 Form 判定基准 0: 如果在副标头内 2 处 Form 中的代码不相同，就为 not XA。 1: 不进行 XA 判定。第一个 Form 有效，但是对 2 处的 Form 进行是否相同的检查，其结果反映到状态位。 此位的设定只在 MD_AUTO 位为“1”时有效。
2~0	MD_SEC[2:0]	010	R/W	扇区类型设定 000: 禁止设定 001: Mode0 010: Mode1 011: Long (Mode0、Mode1、Mode2, 没有 EDC/ECC 数据) 100: 禁止设定 101: Mode2 Form1 110: Mode2 Form2 111: Mode2 自动格式检测 当设定为“B'111”时，如果无法判定类型，就作为 Mode2 not XA 处理。

26.3.4 EDC、ECC 校验控制寄存器 (CROMCTL1)

CROMCTL1 是控制 EDC、ECC 校验的寄存器。此寄存器的设定在切换扇区时有效。

位:	7	6	5	4	3	2	1	0
	M2F2 EDC	MD_DEC[2:0]			—	—	MD_PQREP[1:0]	
初始值:	1	1	0	1	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	M2F2EDC	1	R/W	在 Mode2 Form2 中，如果 EDC 码为 All 0，EDC 功能就无效。如果将此位置“1”，即使在 Mode2 Form2 中的 EDC 码为“0”时 EDC 校验为 NG，也不产生 IERR 中断。写操作无效
6 ~ 4	MD_DEC[2:0]	101	R/W	选择 EDC、ECC 校验模式 000: 无校验 001: 仅 EDC 010: Q+EDC 011: P+EDC 100: QP+EDC 101: PQ+EDC 110: 禁止设定 111: 禁止设定
3、2	—	全 0	R/W	保留位 读写值都为“0”。
1、0	MD_PQREP [1:0]	01	R/W	PQ、QP 校正的校正重复次数 通过 MD_DEC 位选择设定 PQ、QP 校正时的校正重复次数。 00: 禁止设定 01: 1 次校正 10: 2 次重复校正 11: 3 次重复校正

26.3.5 解码处理自动停止控制寄存器 (CROMCTL3)

当发生解码异常时，CROMCTL3能自动停止解码处理。如果停止，就产生IBUF中断，并且能通过CBUFST1寄存器确认停止源。此寄存器的设定在切换扇区时有效。

位:	7	6	5	4	3	2	1	0
	STP ECC	STP EDC	—	STP MD	STP MIN	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	STP_ECC	0	R/W	如果将 STP_ECC 位置“1”，就在不能进行 ECC 校正时中止 CD-ROM 解码处理。
6	STP_EDC	0	R/W	如果将 STP_EDC 位置“1”，就在 ECC 后的 EDC 发生错误时中止 CD-ROM 解码处理。
5	—	0	R/W	保留位 读写值都为“0”。
4	STP_MD	0	R/W	如果将 STP_MD 位置“1”，就在 Mode Form 和前一个扇区不同时中止解码处理。
3	STP_MIN	0	R/W	如果将 STP_MIN 位置“1”，就在 MINUTES、SECONDS、FRAME (1/75 秒) 中途不连续时中止解码处理。
2~0	—	全 0	R/W	保留位 读写值都为“0”。

26.3.6 解码选项设定控制寄存器 (CROMCTL4)

CROMCTL4 控制链路块的检测，选择状态寄存器的表示并且控制 ECC 校正模式。此寄存器的设定在切换扇区时有效。

位:	7	6	5	4	3	2	1	0
	LINKOFF	LINK2	—	ER0SEL	NO_ECC	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	LINKOFF	0	R/W	链路块检测时的缓冲控制 0: ON 1: OFF 如果将此位置“1”，就不在检测链路块时进行缓冲控制。作为普通的扇区处理。 但是，检测链路块（结果反映到 CROMST5 寄存器的 bit3 ~ 0）。
6	LINK2	0	R/W	链路块检测条件的选择 0: 当检测到 run-out1 和 run-out2 的任意一个或者检测 run-in3 和 run-in4 时，判定为链路块。 1: 在 run-out1 和 run-out2 连接中检测到 2 个时，判定为链路块。 如果将此位置“1”，链路块的缓冲控制就无效（作为普通的扇区处理）。 另外，LINK_ON 位的置位条件是对链路扇区进行解码时。
5	—	0	R/W	保留位 读写值都为“0”。
4	ER0SEL	0	R/W	ROM 数据相关状态寄存器的设定条件选择 0: 表示当前正在解码的扇区信息 1: 表示缓冲结束的最新扇区信息 对象为 CROMST0 寄存器的 bit5 ~ 0、CROMST4、CROMST5 寄存器的 bit7 ~ 1、HEAD00 ~ 02。
3	NO_ECC	0	R/W	ECC 前 EDC 校验 OK 时的 ECC 校正模式选择 如果将此位置“1”，就在 ECC 前 EDC 校验 OK 时不进行 ECC 校正，与校正子计算结果无关。
2 ~ 0	—	全 0	R/W	保留位 读写值都为“0”。

26.3.7 HEAD20 ~ 22 表示控制寄存器 (CROMCTL5)

CROMCTL5 设定 HEAD20 ~ 22 的表示方式。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	MSF_LBA_SEL
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 1	—	全 0	R/W	保留位 读写值都为“0”。
0	MSF_LBA_SEL	0	R/W	HEAD20 ~ 22 的表示方式 0: 直接用 BCD (10 进制) 表示标头的 MSF 1: 用 16 进制表示总扇区的值

26.3.8 同步码状态寄存器 (CROMST0)

CROMST0 表示同步码保护功能时的状态。

位:	7	6	5	4	3	2	1	0
	—	—	ST_SYIL	ST_SYNO	ST_BLK5	ST_BLKL	ST_SECS	ST_SECL
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7、6	—	全 0	R	保留位 读取值总是“0”，写操作无效。
5	ST_SYIL	0	R	表示在字计数器 (测量同步码间隔) 不正确时检测到同步码, 但是被忽视而不进行同步。 此位只在自动同步保护模式或者内插模式中有效。
4	ST_SYNO	0	R	表示在字计数器变为最后值时也没有检测到同步码, 但是通过内插取得了解码器的同步。 此位只在自动同步保护模式或者内插模式中有效。
3	ST_BLK5	0	R	表示在字计数器不正确时检测到同步码并且以此取得了解码器的同步。 此位只在自动同步保护模式或者外部同步模式中有效。
2	ST_BLKL	0	R	表示在字计数器变为最后值时也没有检测到同步码, 但是延长了 1 个扇区的期间。 此位只在外部同步模式中有效。
1	ST_SECS	0	R	表示同步码和内插的同步时序作为短扇区进行对应。如果此位为“1”, 就立即停止解码, 从当前正在解码的前一个扇区重新开始尝试解码。
0	ST_SECL	0	R	表示同步码和内插的同步时序作为长扇区进行对应。如果此位为“1”, 就立即停止解码, 从当前正在解码的前二个扇区重新开始尝试解码。

26.3.9 ECC 后的标头错误状态寄存器 (CROMST1)

CROMST1 表示 ECC 后的扇区标头错误状态。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	ER2 HEAD0	ER2 HEAD1	ER2 HEAD2	ER2 HEAD3
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 4	—	全 0	R	保留位 读取值总是“0”，写操作无效。
3	ER2_HEAD0	0	R	表示 ECC 后的标头部 MINUTES 的错误状态。
2	ER2_HEAD1	0	R	表示 ECC 后的标头部 SECONDS 的错误状态。
1	ER2_HEAD2	0	R	表示 ECC 后的标头部 FRAME (1/75 秒) 的错误状态。
0	ER2_HEAD3	0	R	表示 ECC 后的标头部 MODE 的错误状态。

26.3.10 ECC 后的副标头错误状态寄存器 (CROMST3)

CROMST3 表示 ECC 后的副标头错误状态。

位:	7	6	5	4	3	2	1	0
	ER2 SHEAD0	ER2 SHEAD1	ER2 SHEAD2	ER2 SHEAD3	ER2 SHEAD4	ER2 HEAD5	ER2 HEAD6	ER2 HEAD7
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7	ER2_SHEAD0	0	R	表示 ECC 后的副标头 (文件号) 的错误状态。 表示 SHEAD20 寄存器的错误。
6	ER2_SHEAD1	0	R	表示 ECC 后的副标头 (通道号) 的错误状态。 表示 SHEAD21 寄存器的错误。
5	ER2_SHEAD2	0	R	表示 ECC 后的副标头 (子模式) 的错误状态。 表示 SHEAD22 寄存器的错误。
4	ER2_SHEAD3	0	R	表示 ECC 后的副标头 (数据类型) 的错误状态。 表示 SHEAD23 寄存器的错误。
3	ER2_SHEAD4	0	R	表示 ECC 后的副标头 (文件号) 的错误状态。 表示 SHEAD24 寄存器的错误。
2	ER2_SHEAD5	0	R	表示 ECC 后的副标头 (通道号) 的错误状态。 表示 SHEAD25 寄存器的错误。
1	ER2_SHEAD6	0	R	表示 ECC 后的副标头 (子模式) 的错误状态。 表示 SHEAD26 寄存器的错误。
0	ER2_SHEAD7	0	R	表示 ECC 后的副标头 (数据类型) 的错误状态。 表示 SHEAD27 寄存器的错误。

26.3.11 标头 / 副标头数据有效性判定状态寄存器 (CROMST4)

CROMST4 表示自动模式判定和 Mode2 Form 判定的错误。

位:	7	6	5	4	3	2	1	0
	NG_MD	NG_MDCMP1	NG_MDCMP2	NG_MDCMP3	NG_MDCMP4	NG_MDDEF	NG_MDTIM1	NG_MDTIM2
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7	NG_MD	0	R	表示按照自动模式判定基准而无法判定模式。
6	NG_MDCMP1	0	R	表示在判定 Mode2 的 Form 时文件号 (BYTE-16 和 BYTE-20) 发生比较错误。
5	NG_MDCMP2	0	R	表示在判定 Mode2 的 Form 时通道号 (BYTE-17 和 BYTE-21) 发生比较错误。
4	NG_MDCMP3	0	R	表示在判定 Mode2 的 Form 时子模式 (BYTE-18 和 BYTE-22) 发生比较错误。
3	NG_MDCMP4	0	R	表示在判定 Mode2 的 Form 时数据类型 (BYTE-19 和 BYTE-23) 发生比较错误。
2	NG_MDDEF	0	R	表示 Mode 和 Form 与前一个扇区不同。
1	NG_MDTIM1	0	R	表示标头的 MINUTES、SECONDS、FRAMES (1/75 秒) 的值中途不连续。 在检查下一个扇区的连续性时使用更新值。
0	NG_MDTIM2	0	R	表示标头的 MINUTES、SECONDS、FRAMES 不是 BCD (10 进制) 的值。 此位表示非 BCD 的值 (A ~ F) 并且 HEAD01 的值大于 "H'59" 或者 HEAD02 的值大于 "H'74"。 在检查下一个扇区的连续性时使用内插值。

26.3.12 模式判定结果和链路扇区检测状态寄存器 (CROMST5)

CROMST5 表示自动模式的判定结果和链路块检测。

位:	7	6	5	4	3	2	1	0
	ST_AMD[2:0]			ST_MDX	LINK_ON	LINK_DET	LINK_SDET	LINK_OUT1
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 5	ST_AMD[2:0]	000	R	自动模式判定结果 表示使用自动模式判定功能时的模式判定结果。 000: 不使用自动模式判定功能 001: Mode0 010: Mode1 011: — 100: Mode2 not XA 101: Mode2 Form1 110: Mode2 Form2 111: —
4	ST_MDX	0	R	在不使用自动模式判定而用手动设定时, 表示设定值会和逻辑识别的结果不相同。手动设定值优先。
3	LINK_ON	0	R	在通过链路块判定识别为链路块时此位为“1”。 有关链路块判定基准, 请参照 CROMCTL4 寄存器的 LINK2 位。 当此位为“1”时, 按照 CBUFCTL0 寄存器的 CBUF_LINK 位进行缓冲控制。
2	LINK_DET	0	R	表示检测到链路块 (run-out1 ~ run-in4)。 因为通过 ECC 校正前的数据进行检测, 所以在和链路块相同的代码中发生数据错误时 LINK_DET 位可能变为“1”。
1	LINK_SDET	0	R	表示在解码开始后的 7 个扇区内检测到链路块。
0	LINK_OUT1	0	R	表示在 ECC 处理后判定为 run-out1 扇区。 此位只在不发生 IERR 中断 (正常进行了 ECC 校正) 时有效。

26.3.13 ECC、EDC 错误状态寄存器 (CROMST6)

CROMST6 表示 ECC 处理错误和 ECC 前后的 EDC 校验错误。

位:	7	6	5	4	3	2	1	0
	ST_ERR	—	ST_ECCABT	ST_ECCNG	ST_ECCP	ST_ECCQ	ST_EDC1	ST_EDC2
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7	ST_ERR	0	R	表示在 ECC 后的解码块中即使 1 字节也有错误。
6	—	0	R	保留位 读取值总是“0”，写操作无效。
5	ST_ECCABT	0	R	表示 ECC 处理中途被中止。 在 ECC 校正处理中发生扇区切换时，此位为“1”。根据时序，即使此位为“1”，只要 CBUFST2 寄存器的 BUF_NG 位为“0”，ECC 校正运行就不会出现问题。
4	ST_ECCNG	0	R	表示无法进行错误校正。 在检测到短扇区时，此位也为“1”。
3	ST_ECCP	0	R	表示不能通过 ECC 处理进行 P 系列的校正。 此位只在同步状态正常（不是短扇区和长扇区）时有效。 当 P 系列的校正不是 All0 时，此位为“1”。
2	ST_ECCQ	0	R	表示不能通过 ECC 处理进行 Q 系列的校正。 此位只在同步状态正常（不是短扇区和长扇区）时有效。 当 Q 系列的校正不是 All0 时，此位为“1”。
1	ST_EDC1	0	R	表示 ECC 前的 EDC 校验为 NG。 此位在 EDC 时有效，即使为短扇区时，此位也为“1”。
0	ST_EDC2	0	R	表示 ECC 后的 EDC 校验为 NG。

26.3.14 缓冲状态寄存器 (CBUFST0)

CBUFST0 表示正在检索缓冲开始扇区或者正在进行缓冲。

位:	7	6	5	4	3	2	1	0
	BUF_REF	BUF_ACT	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7	BUF_REF	0	R	表示正在检索缓冲开始扇区。 此位只在使用自动缓冲功能时（CBUF_AUT 位为“1”）有效。
6	BUF_ACT	0	R	表示正在进行缓冲。
5 ~ 0	—	全 0	R	保留位 读取值总是“0”，写操作无效。

26.3.15 解码中止源状态寄存器 (CBUFST1)

CBUFST1 表示因各错误而中止解码 / 缓冲处理。

只在 CROMCTL3 的对应位为“1”时，才将此寄存器置位。

位:	7	6	5	4	3	2	1	0
	BUF ECC	BUF EDC	—	BUF MD	BUF MIN	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7	BUF_ECC	0	R	表示因不能进行 ECC 校正而中止 ROM 解码 / 缓冲处理。
6	BUF_EDC	0	R	表示因在 ECC 校正后的 EDC 校验中检测到错误而中止解码 / 缓冲处理。
5	—	0	R	保留位 读取值总是“0”，写操作无效。
4	BUF_MD	0	R	表示因 Mode 和 Form 与前一个扇区不同而中止解码 / 缓冲处理。
3	BUF_MIN	0	R	表示因 MINUTES、SECONDS、FRAMES (1/75 秒) 在中途不连续而中止解码 / 缓冲处理。
2~0	—	全 0	R	保留位 读取值总是“0”，写操作无效。

26.3.16 缓冲器上溢状态寄存器 (CBUFST2)

CBUFST2 表示在缓冲器的传送还没有结束时发生扇区切换。

位:	7	6	5	4	3	2	1	0
	BUF NG	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7	BUF_NG	0	R	表示在缓冲器的数据传送还没有结束时发生扇区切换。如果不从 CD-ROM 解码器取出输出流数据而输入第 3 个扇区的数据，此位就为“1”，此时不产生中断。当此位为“1”时，只有通过 ROMDECRST 寄存器的 LOGICRST 位进行的复位，才能将此位清“0”。
6~0	—	全 0	R	保留位 读取值总是“0”，写操作无效。

26.3.17 ECC 校正前标头部 -MINUTES 数据寄存器 (HEAD00)

HEAD00 表示 ECC 校正前标头部 MINUTES 的值。

位:	7	6	5	4	3	2	1	0
	HEAD00[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7~0	HEAD00[7:0]	H'00	R	ECC 校正前标头部 MINUTES 的值

26.3.18 ECC 校正前标头部 -SECONDS 数据寄存器 (HEAD01)

HEAD01 表示 ECC 校正前标头部 SECONDS 的值。

位:	7	6	5	4	3	2	1	0
	HEAD01[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7~0	HEAD01[7:0]	H'00	R	ECC 校正前标头部 SECONDS 的值

26.3.19 ECC 校正前标头部 -FRAMES (1/75 秒) 数据寄存器 (HEAD02)

HEAD02 表示 ECC 校正前标头部 FRAMES (1/75 秒) 的值。

位:	7	6	5	4	3	2	1	0
	HEAD02[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7~0	HEAD02[7:0]	H'00	R	ECC 校正前头标部 FRAMES (1/75 秒) 的值

26.3.20 ECC 校正前标头部 -MODE 数据寄存器 (HEAD03)

HEAD03 表示 ECC 校正前标头部 MODE 的值。

位:	7	6	5	4	3	2	1	0
	HEAD03[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7~0	HEAD03[7:0]	H'00	R	ECC 校正前标头部 MODE 的值

26.3.21 ECC 校正前副标头部 - 文件号 (BYTE-16) 数据寄存器 (SHEAD00)

SHEAD00 表示 ECC 校正前副标头部的文件码的值 (BYTE-16)。

位:	7	6	5	4	3	2	1	0
	SHEAD00[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 0	SHEAD00[7:0]	H'00	R	ECC 校正前副标头的文件号的值 (BYTE-16) 表示在不是 Mode2 时相应位置的字节数据。

26.3.22 ECC 校正前副标头部 - 通道号 (BYTE-17) 数据寄存器 (SHEAD01)

SHEAD01 表示 ECC 校正前副标头部的通道号的值 (BYTE-17)。

位:	7	6	5	4	3	2	1	0
	SHEAD01[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 0	SHEAD01[7:0]	H'00	R	ECC 校正前副标头的通道号的值 (BYTE-17) 表示在不是 Mode2 时相应位置的字节数据。

26.3.23 ECC 校正前副标头部 - 子模式 (BYTE-18) 数据寄存器 (SHEAD02)

SHEAD02 表示 ECC 校正前副标头部的子模式的值 (BYTE-18)。

位:	7	6	5	4	3	2	1	0
	SHEAD02[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 0	SHEAD02[7:0]	H'00	R	ECC 校正前副标头的子模式的值 (BYTE-18) 表示在不是 Mode2 时相应位置的字节数据。

26.3.24 ECC 校正前副标头部 - 数据类型 (BYTE-19) 数据寄存器 (SHEAD03)

SHEAD03 表示 ECC 校正前副标头部的数据类型的值 (BYTE-19)。

位:	7	6	5	4	3	2	1	0
	SHEAD03[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 0	SHEAD03[7:0]	H'00	R	ECC 校正前副标头的数据类型值的值 (BYTE-19) 表示在不是 Mode2 时相应位置的字节数据。

26.3.25 ECC 校正前副标头部 - 文件号 (BYTE-20) 数据寄存器 (SHEAD04)

SHEAD04 表示 ECC 校正前副标头部的文件号的值 (BYTE-20)。

位:	7	6	5	4	3	2	1	0
	SHEAD04[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 0	SHEAD04[7:0]	H'00	R	ECC 校正前副标头的文件号的值 (BYTE-20) 表示在不是 Mode2 时相应位置的字节数据。

26.3.26 ECC 校正前副标头部 - 通道号 (BYTE-21) 数据寄存器 (SHEAD05)

SHEAD05 表示 ECC 校正前副标头部的通道号的值 (BYTE-21)。

位:	7	6	5	4	3	2	1	0
	SHEAD05[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 0	SHEAD05[7:0]	H'00	R	ECC 校正前副标头的通道号的值 (BYTE-21) 表示在不是 Mode2 时相应位置的字节数据。

26.3.27 ECC 校正前副标头部 - 子模式 (BYTE-22) 数据寄存器 (SHEAD06)

SHEAD06 表示 ECC 校正前副标头部的子模式的值 (BYTE-22)。

位:	7	6	5	4	3	2	1	0
	SHEAD06[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 0	SHEAD06[7:0]	H'00	R	ECC 校正前副标头的子模式的值 (BYTE-22) 表示在不是 Mode2 时相应位置的字节数据。

26.3.28 ECC 校正前副标头部 - 数据类型 (BYTE-23) 数据寄存器 (SHEAD07)

SHEAD07 表示 ECC 校正前副标头部的数据类型的值 (BYTE-23)。

位:	7	6	5	4	3	2	1	0
	SHEAD07[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 0	SHEAD07[7:0]	H'00	R	ECC 校正前副标头的数据类型值的值 (BYTE-23) 表示在不是 Mode2 时相应位置的字节数据。

26.3.29 ECC 校正后标头部 -MINUTES 数据寄存器 (HEAD20)

HEAD20 表示 ECC 校正后标头部 MINUTES 的值。

位:	7	6	5	4	3	2	1	0
	HEAD20[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 0	HEAD20[7:0]	H'00	R	ECC 校正后标头部 MINUTES 的值 在 MSF_LBA_SEL 位为 “1” 时, 表示 M、S 和 F 的总扇区的值 (1/3)。

26.3.30 ECC 校正后标头部 -SECONDS 数据寄存器 (HEAD21)

HEAD21 表示 ECC 校正后标头部 SECONDS 的值。

位:	7	6	5	4	3	2	1	0
	HEAD21[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 0	HEAD21[7:0]	H'00	R	ECC 校正后标头部 SECONDS 的值 在 MSF_LBA_SEL 位为 “1” 时, 表示 M、S 和 F 的总扇区的值 (2/3)。

26.3.31 ECC 校正后标头部 -FRAMES (1/75 秒) 数据寄存器 (HEAD22)

HEAD22 表示 ECC 校正后标头部 FRAMES (1/75 秒) 的值。

位:	7	6	5	4	3	2	1	0
	HEAD22[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 0	HEAD22[7:0]	H'00	R	ECC 校正后标头部 FRAMES (1/75 秒) 的值 在 MSF_LBA_SEL 位为 “1” 时, 表示 M、S 和 F 的总扇区的值 (3/3)。

26.3.32 ECC 校正后标头部 -MODE 数据寄存器 (HEAD23)

HEAD23 表示 ECC 校正后标头部 MODE 的值。

位:	7	6	5	4	3	2	1	0
	HEAD23[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 0	HEAD23[7:0]	H'00	R	ECC 校正后标头部 MODE 的值

26.3.33 ECC 校正后副标头部 - 文件号 (BYTE-16) 数据寄存器 (SHEAD20)

SHEAD20 表示 ECC 校正后副标头部的文件号的值 (BYTE-16)。

位:	7	6	5	4	3	2	1	0
	SHEAD20[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7~0	SHEAD20[7:0]	H'00	R	ECC 校正后副标头部的文件号的值 (BYTE-16)

26.3.34 ECC 校正后副标头部 - 通道号 (BYTE-17) 数据寄存器 (SHEAD21)

SHEAD21 表示 ECC 校正后副标头部的通道号的值 (BYTE-17)。

位:	7	6	5	4	3	2	1	0
	SHEAD21[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7~0	SHEAD21[7:0]	H'00	R	ECC 校正后副标头部的通道号的值 (BYTE-17)

26.3.35 ECC 校正后副标头部 - 子模式 (BYTE-18) 数据寄存器 (SHEAD22)

SHEAD22 表示 ECC 校正后副标头部的子模式的值 (BYTE-18)。

位:	7	6	5	4	3	2	1	0
	SHEAD22[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7~0	SHEAD22[7:0]	H'00	R	ECC 校正后副标头部的子模式的值 (BYTE-18)

26.3.36 ECC 校正后副标头部 - 数据类型 (BYTE-19) 数据寄存器 (SHEAD23)

SHEAD23 表示 ECC 校正后副标头部的数据类型的值 (BYTE-19)。

位:	7	6	5	4	3	2	1	0
	SHEAD23[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7~0	SHEAD23[7:0]	H'00	R	ECC 校正后副标头部的数据类型值的值 (BYTE-19)

26.3.37 ECC 校正后副标头部 - 文件号 (BYTE-20) 数据寄存器 (SHEAD24)

SHEAD24 表示 ECC 校正后副标头部的文件号的值 (BYTE-20)。

位:	7	6	5	4	3	2	1	0
	SHEAD24[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 0	SHEAD24[7:0]	H'00	R	ECC 校正后副标头部的文件号的值 (BYTE-20)

26.3.38 ECC 校正后副标头部 - 通道号 (BYTE-21) 数据寄存器 (SHEAD25)

SHEAD25 表示 ECC 校正后副标头部的通道号的值 (BYTE-21)。

位:	7	6	5	4	3	2	1	0
	SHEAD25[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 0	SHEAD25[7:0]	H'00	R	ECC 校正后副标头部的通道号的值 (BYTE-21)

26.3.39 ECC 校正后副标头部 - 子模式 (BYTE-22) 数据寄存器 (SHEAD26)

SHEAD26 表示 ECC 校正后副标头部的子模式的值 (BYTE-22)。

位:	7	6	5	4	3	2	1	0
	SHEAD26[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 0	SHEAD26[7:0]	H'00	R	ECC 校正后副标头部的子模式的值 (BYTE-22)

26.3.40 ECC 校正后副标头部 - 数据类型 (BYTE-23) 数据寄存器 (SHEAD27)

SHEAD27 表示 ECC 校正后副标头部的数据类型的值 (BYTE-23)。

位:	7	6	5	4	3	2	1	0
	SHEAD27[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7 ~ 0	SHEAD27[7:0]	H'00	R	ECC 校正后副标头部的数据类型值的值 (BYTE-23)

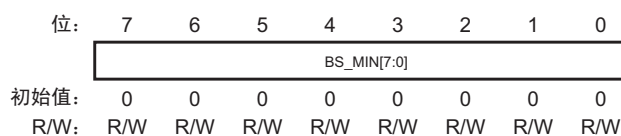
26.3.41 自动缓冲设定控制寄存器 (CBUFCTL0)

位:	7	6	5	4	3	2	1	0
	CBUF_	CBUF_	CBUF_	CBUF_MD[1:0]		CBUF_	CBUF_	—
	AUT	EN	LINK			TS	Q	—
初始值:	0	0	0	0	0	1	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	CBUF_AUT	0	R/W	自动缓冲功能控制 在将 CROM_EN 位置“1”的状态下要将此位置为 ON 或者 OFF 时，也必须同时将 CBUF_EN 位置为 ON 或者 OFF。否则，就不能保证 CBUFST0、CBUFST1、CBUFST2 的表示。 0: 自动缓冲 OFF 1: 自动缓冲 ON
6	CBUF_EN	0	R/W	缓冲器 RAM 的缓冲控制 该位在自动 / 手动缓冲模式中进行缓冲的 ON/OFF。在手动缓冲时，必须在发生 ISEC 中断后将此位置位。 在自动缓冲停止时，此位自动复位。 0: 缓冲 OFF 1: 缓冲 ON
5	CBUF_LINK	0	R/W	链路块检测时的缓冲控制 0: 确保 7 个扇区的区域 1: 填充链路块，进行缓冲。
4、3	CBUF_MD[1:0]	00	R/W	使用自动缓冲功能时的起始扇区检测模式设定 00: 在前一个扇区的检测和当前扇区的检测中，连续性（标头的值）OK。 01: 在当前扇区的检测中，和内插值的连续性 OK。 10: 当前扇区的检测 OK 11: 即使未检测到当前扇区，也根据内插值判定。
2	CBUF_TS	1	R/W	CBUFCTL1 ~ 3 的设定方法 0: CBUFCTL1 ~ 3: BCD（10 进制） 1: 总扇区的值（16 进制）
1	CBUF_Q	0	R/W	QCODE 的 CRC-NG 时的 QCODE 缓冲数据指定 0: 对 CRC-OK 的最终扇区的值进行缓冲 1: 对 NG 数据进行缓冲 【注】 因为本 LSI 不输入子代码，所以必须总是将此位置“1”。
0	—	0	R/W	保留位 读写值都为“0”。

26.3.42 自动缓冲起始扇区设定 -MINUTES 控制寄存器 (CBUFCTL1)

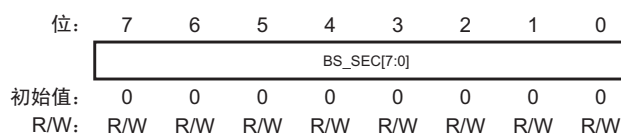
CBUFCTL1 表示缓冲起始扇区的标头 -MINUTES 值。



位	位名	初始值	R/W	说明
7 ~ 0	BS_MIN[7:0]	H'00	R/W	缓冲起始扇区的标头 -MINUTES 值

26.3.43 自动缓冲起始扇区设定 -SECONDS 控制寄存器 (CBUFCTL2)

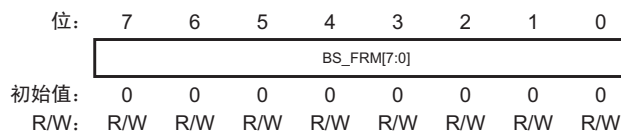
CBUFCTL2 表示缓冲起始扇区的标头 -SECONDS 值。



位	位名	初始值	R/W	说明
7 ~ 0	BS_SEC[7:0]	H'00	R/W	缓冲起始扇区的标头 -SECONDS 值

26.3.44 自动缓冲起始扇区设定 -FRAMES 控制寄存器 (CBUFCTL3)

CBUFCTL3 表示缓冲起始扇区的标头 -FRAMES (1/75 秒) 值。



位	位名	初始值	R/W	说明
7 ~ 0	BS_FRM[7:0]	H'00	R/W	缓冲起始扇区的标头 -FRAMES (1/75 秒) 值

26.3.45 ISY 中断源屏蔽控制寄存器 (CROMST0M)

CROMST0M 屏蔽由同步码状态寄存器 (CROMST0) 的各位引起的 ISY 中断源。

位:	7	6	5	4	3	2	1	0
	—	—	ST_ SYILM	ST_ SYNOM	ST_ BLKSM	ST_ BLKLM	ST_ SECSM	ST_ SECLM
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7、6	—	全 0	R/W	保留位 读写值都为“0”。
5	ST_SYILM	0	R/W	屏蔽 ISY 中断 ST_SYIL (CROMST0 寄存器的 bit5) 源。
4	ST_SYNOM	0	R/W	屏蔽 ISY 中断 ST_SYNO (CROMST0 寄存器的 bit4) 源。
3	ST_BLKSM	0	R/W	屏蔽 ISY 中断 ST_BLK S (CROMST0 寄存器的 bit3) 源。
2	ST_BLKLM	0	R/W	屏蔽 ISY 中断 ST_BLK L (CROMST0 寄存器的 bit2) 源。
1	ST_SECSM	0	R/W	屏蔽 ISY 中断 ST_SECS (CROMST0 寄存器的 bit1) 源。
0	ST_SECLM	0	R/W	屏蔽 ISY 中断 ST_SECL (CROMST0 寄存器的 bit0) 源。

26.3.46 CD-ROM 解码器模块复位控制寄存器 (ROMDECRST)

ROMDECRST 对 CD-ROM 解码器的随机逻辑部进行复位以及清除 CD-ROM 解码器的 RAM。

位:	7	6	5	4	3	2	1	0
	LOGI CRST	RAM RST	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	LOGICRST	0	R/W	CD-ROM 解码器的随机逻辑部的复位信号 在此寄存器为“1”的期间输出复位信号。
6	RAMRST	0	R/W	CD-ROM 解码器的 RAM 清除信号 必须通过 RSTSTAT 寄存器的 RAMCLRST 位确认 RAM 清除是否结束。
5~0	—	全 0	R/W	保留位 读写值都为“0”。

【注】 要将 LOGICRST 置“1”时，必须在确认 RAMRST 位为“0”后给此寄存器写“B'10000000”。

26.3.47 CD-ROM 解码器模块复位状态寄存器 (RSTSTAT)

RSTSTAT 表示 CD-ROM 解码器的 RAM 清除状态。

位:	7	6	5	4	3	2	1	0
	RAM CLRST	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7	RAMCLRST	0	R	在给 ROMDECRST 寄存器的 RAMRST 位写“1”后，如果 RAM 清除结束，此位就为“1”。通过给 RAMRST 位写“0”，清除此位。
6~0	—	全 0	R	保留位 读取值总是“0”，写操作无效。

26.3.48 串行音频接口数据控制寄存器 (SSI)

SSI 进行流数据相关的各种设定。有关设定此寄存器时的运行内容，请参照“26.4.1 输入流数据字节序的转换功能”。

位:	7	6	5	4	3	2	1	0
	BYTEND	BITEND	BUFEND0[1:0]	BUFEND1[1:0]	—	—	—	—
初始值:	0	0	0 1	1 1	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	BYTEND	0	R/W	更改从串行音频接口输入的流数据的字节序。 如果将此位置“1”，就调换STRMDIN0和STRMDIN2的字节0和字节1。
6	BITEND	0	R/W	更改从串行音频接口输入的流数据的位序。 如果将此位置“1”，就将字节内的位序0→7改为7→0。
5、4	BUFEND0[1:0]	01	R/W	选择更改串行音频接口送来的流数据顺序或者抑制流数据。在串行音频接口中，能选择“填充模式”或者“不填充模式”。在“不填充模式”中，作为CD-ROM数据，从串行音频接口传送32位数据。在CD-ROM解码器内部有2组16位的输入流数据寄存器，能调换此顺序。另外，在“填充模式”中，从串行音频接口传送含填充的32位数据。因为填充部没有意义，所以作为流数据需要进行抑制，能通过设定此寄存器进行抑制。CD-ROM解码以16位流数据进行处理，此寄存器抑制串行音频接口送来的32位传送数据中先输入的16位数据。 00: 抑制先处理的流数据 01: 将串行音频接口送来的32位流数据中的高16位数据作为先处理的流数据 10: 将串行音频接口送来的32位流数据中的低16位数据作为先处理的流数据 11: 禁止设定
3、2	BUFEND1[1:0]	10	R/W	选择更改串行音频接口送来的流数据顺序或者抑制流数据。在串行音频接口中，能选择“填充模式”或者“不填充模式”。在“不填充模式”中，作为CD-ROM数据，从串行音频接口传送32位数据。在CD-ROM解码器内部有2组16位的输入流数据寄存器，能调换此顺序。另外，在“填充模式”中，从串行音频接口传送含填充的32位数据。因为填充部没有意义，所以作为流数据需要进行抑制，能通过设定此寄存器进行抑制。CD-ROM解码以16位流数据进行处理，此寄存器抑制串行音频接口送来的32位传送数据中后输入的16位数据。 00: 抑制后处理的流数据 01: 将串行音频接口送来的32位流数据中的高16位数据作为后处理的流数据 10: 将串行音频接口送来的32位流数据中的低16位数据作为后处理的流数据 11: 禁止设定
1、0	—	全0	R/W	保留位 读写值都为“0”。

26.3.49 中断标志寄存器 (INTHOLD)

INTHOLD 由各种中断标志构成。

位:	7	6	5	4	3	2	1	0
	ISEC	ITARG	ISY	IERR	IBUF	IREADY	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	ISEC	0	R/W	ISEC 中断标志 如果读不到“1”，就不能写“0”。
6	ITARG	0	R/W	ITARG 中断标志 如果读不到“1”，就不能写“0”。
5	ISY	0	R/W	ISY 中断标志 如果读不到“1”，就不能写“0”。
4	IERR	0	R/W	IERR 中断标志 如果读不到“1”，就不能写“0”。
3	IBUF	0	R/W	IBUF 中断标志 如果读不到“1”，就不能写“0”。
2	IREADY	0	R/W	IREADY 中断标志 如果读不到“1”，就不能写“0”。
1、0	—	全 0	R/W	保留位 读写值都为“0”。

26.3.50 中断源屏蔽控制寄存器 (INHINT)

INHINT 控制 CD-ROM 解码器的各种中断请求。

位:	7	6	5	4	3	2	1	0
	INH ISEC	INH ITARG	INH ISY	INH IERR	INH IBUF	INH IREADY	PREINH REQDM	PREINH IREADY
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	INHISEC	0	R/W	ISEC 中断屏蔽 当此位为“1”时，禁止 ISEC 中断请求。
6	INHITARG	0	R/W	ITARG 中断屏蔽 当此位为“1”时，禁止 ITARG 中断请求。
5	INHISY	0	R/W	ISY 中断屏蔽 当此位为“1”时，禁止 ISY 中断请求。
4	INHIERR	0	R/W	IERR 中断屏蔽 当此位为“1”时，禁止 IERR 中断请求。
3	INHIBUF	0	R/W	IBUF 中断屏蔽 当此位为“1”时，禁止 IBUF 中断请求。
2	INHIREADY	0	R/W	IREADY 中断屏蔽 当此位为“1”时，禁止 IREADY 中断请求。
1	PREINHREQDM	0	R/W	禁止将输出流数据的 DMA 传送请求中断标志置位。 当此位为“1”时，不保持 DMA 传送请求中断源。
0	PREINHIREADY	0	R/W	禁止将 IREADY 中断标志置位。 当此位为“1”时，IREADY 标志不保持中断源。

26.3.51 CD-ROM 解码器流数据输入寄存器 (STRMDIN0)

STRMDIN0 表示输入到 CD-ROM 解码器的 4 字节数据中 MSB 侧的 2 字节。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STRMDIN[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 0	STRMDIN [31:16]	H'0000	R/W	输入到 CD-ROM 解码器的 4 字节数据中 MSB 侧的 2 字节 CD-ROM 解码器有作为寄存器的 4 字节数据窗口，将输入到该寄存器的数据作为流数据处理。1 个扇区的数据量为 2352 字节。

26.3.52 CD-ROM 解码器流数据输入寄存器 (STRMDIN2)

STRMDIN2 表示输入到 CD-ROM 解码器的 4 字节数据中 LSB 侧的 2 字节。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STRMDIN[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 0	STRMDIN[15:0]	H'0000	R/W	输入到 CD-ROM 解码器的 4 字节数据中 LSB 侧的 2 字节 CD-ROM 解码器有作为寄存器的 4 字节数据窗口，将输入到该寄存器的数据作为流数据处理。1 个扇区的数据量为 2352 字节。

26.3.53 CD-ROM 解码器流数据输出寄存器 (STRMDOUT0)

STRMDOUT0 表示 CD-ROM 解码器输出的 2 字节数据。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STRMDOUT[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 0	STRMDOUT [15:0]	H'0000	R	CD-ROM 解码器输出的 2 字节数据 CD-ROM 解码器有作为寄存器的 2 字节数据窗口，该寄存器输出的数据为 ROM 解码处理后的数据。每次存取该寄存器时，按照另外定义的输出格式，依次输出存取长度的数据。1 个扇区的数据量为 2768 字节，必须读 2768 字节。

26.4 运行说明

26.4.1 输入流数据字节序的转换功能

需要按照 CD-ROM 数据格式规格的顺序将流数据输入到 CD-ROM 解码器内核部。但是，在一些系统中，可能需要调换串行音频接口送来的数据顺序或者传送填充部。因此，流数据输入控制部有调换数据顺序的功能，使填充数据不输入到 CD-ROM 解码器内核部。

这些都是通过串行音频接口数据控制寄存器（SSI）进行控制。作为输入流数据，以 16 位为单位将“填充数据 + 同步码起始 2 字节”（即 H'000000FF）顺序相反后的数据（H'00FF0000）输入到 CD-ROM 解码器，如图 26.6 所示。

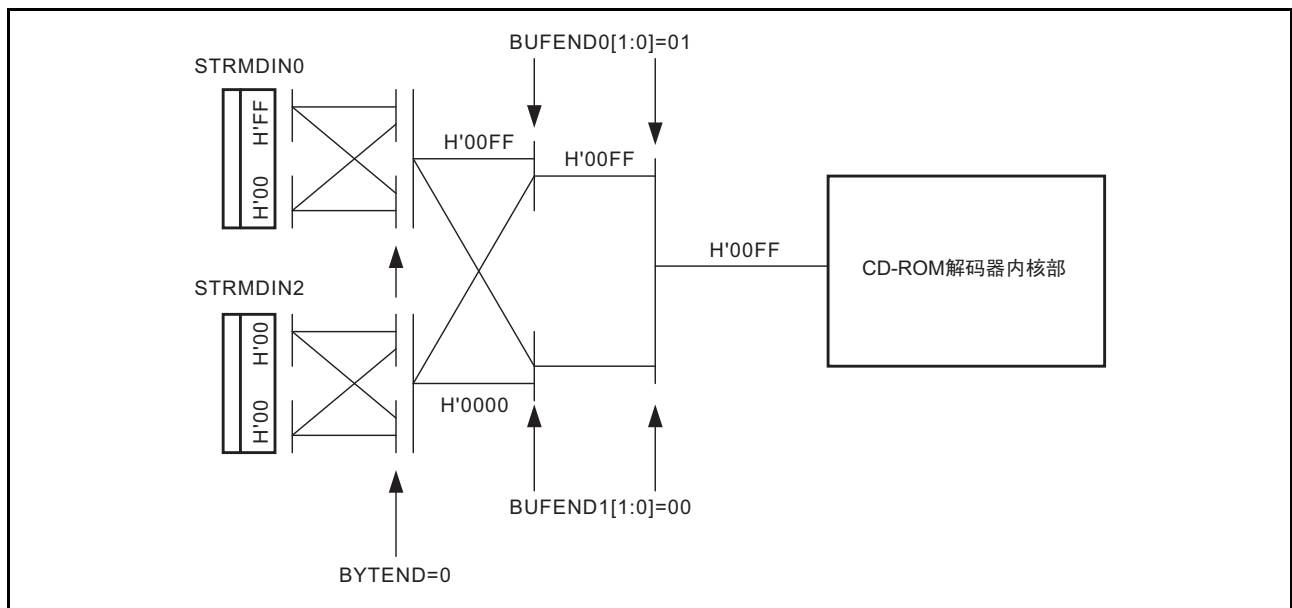


图 26.6 含填充数据的流数据的 SSI 寄存器控制例子

作为流数据，以 16 位为单位将不含填充数据的 H'12345678 顺序相反后的数据（H'56781234）输入到 CD-ROM 解码器，如图 26.7 所示。

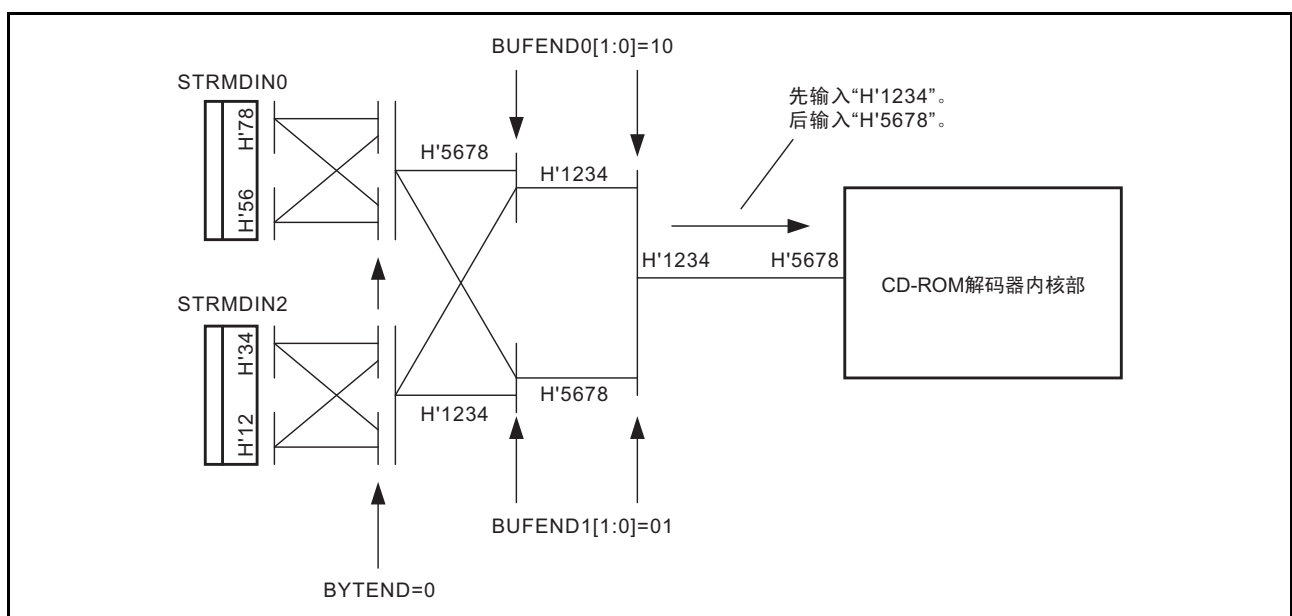


图 26.7 不含填充数据的流数据的 SSI 寄存器控制例子

26.4.2 同步码保护功能

以 H'00FFFFFFFFFFFFFFFFF00（同步码）开始的 2352 字节为 CD-ROM 的 1 个扇区的数据，但是可能因划伤等原因出现识别同步码的时序不正常或者应该检测到同步码时而不能检测到同步码的情况。因此，此 CD-ROM 解码器有以下功能：

- 在异常时序检测到同步码时，忽视该同步码。
- 在应该检测到同步码时而不能检测到同步码的情况下，保护同步码。

同步码保护功能有以下模式。有关设定的详细内容，请参照“26.3.2 同步码的同步控制寄存器（CROMSY0）”和表 26.2。

- 自动同步保护模式
- 外部同步模式
- 内插同步模式
- 内插/外部同步模式

(1) 自动同步保护模式

在自动同步保护模式中，忽视在 1 个扇区（2352 字节）期间内检测到的同步码，如果在下一个扇区的开头检测不到同步码，就保护同步码。另外，如果同步时序发生变化，就在同步时序变化后的 2352 字节内检测到同步码时，再次进行同步。

因此，在排除异常同步模型并且跟随同步时序的变更时有效。但是，对于同步时序变更时的第一个扇区，不能跟随。

在正常情况的运行如图 26.8 所示，在 1 个扇区期间内检测到同步码时的运行如图 26.9 所示，在超过 1 个扇区期间后检测到同步码时的运行如图 26.10 所示。

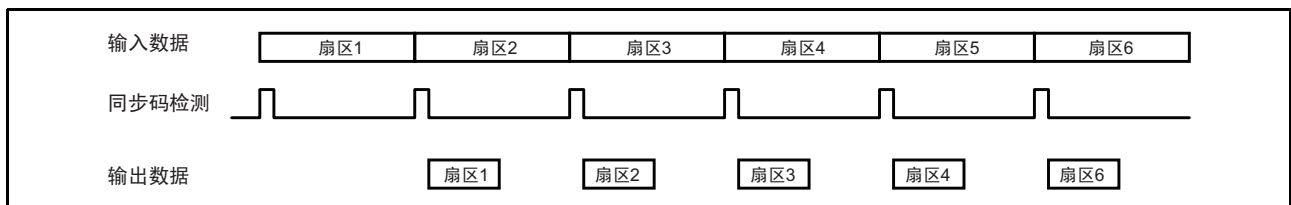


图 26.8 正常时的自动同步保护模式运行

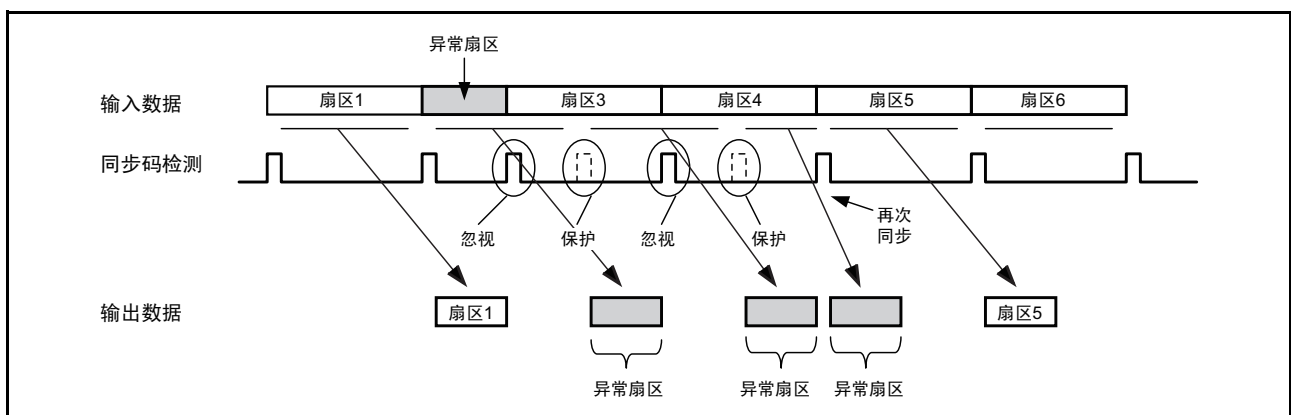


图 26.9 短扇区的自动同步保护模式运行

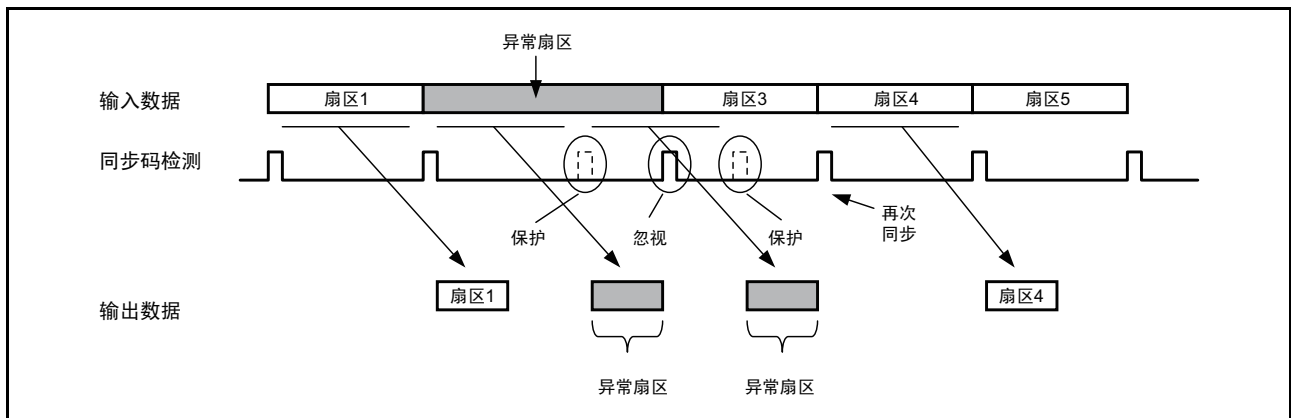


图 26.10 长扇区的自动同步保护模式运行

(2) 外部同步模式

在外部同步模式中，总是通过输入数据的同步码进行同步。如果在 2352 字节内没有发现下一个同步码，就不在检测到同步码前进行解码。

因此，在跟随外部同步时序时有效。但是，如果输入异常同步码模式，就不能进行正常的解码。

外部同步模式中的运行如图 26.11 所示。

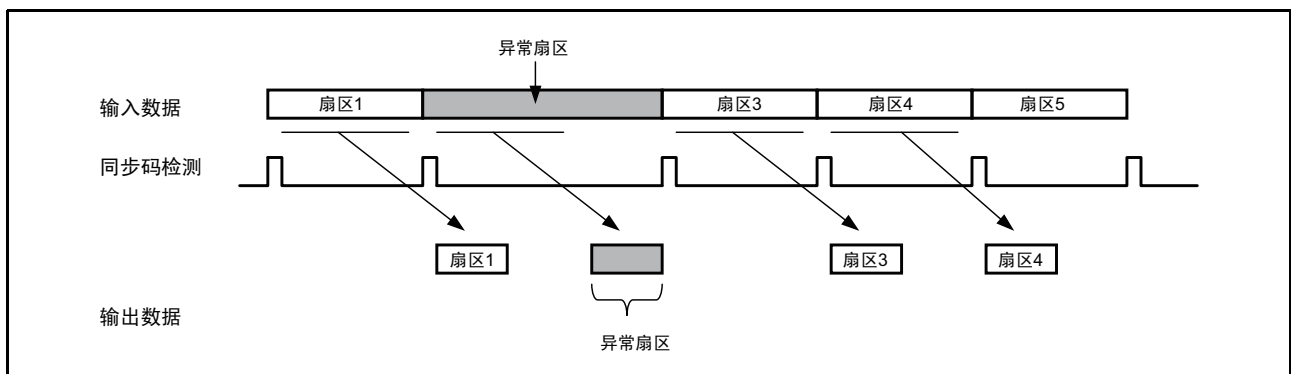


图 26.11 外部同步模式中的运行

(3) 内插同步模式

在内插同步模式中，总是在检测到解码开始时的同步码模式后通过内部计数器进行同步。因此，在同步模型损坏时有效。

但是，如果同步码时序发生变化，就不能跟随，并且不能进行正常的解码。

内插同步模式中的运行如图 26.12 所示。

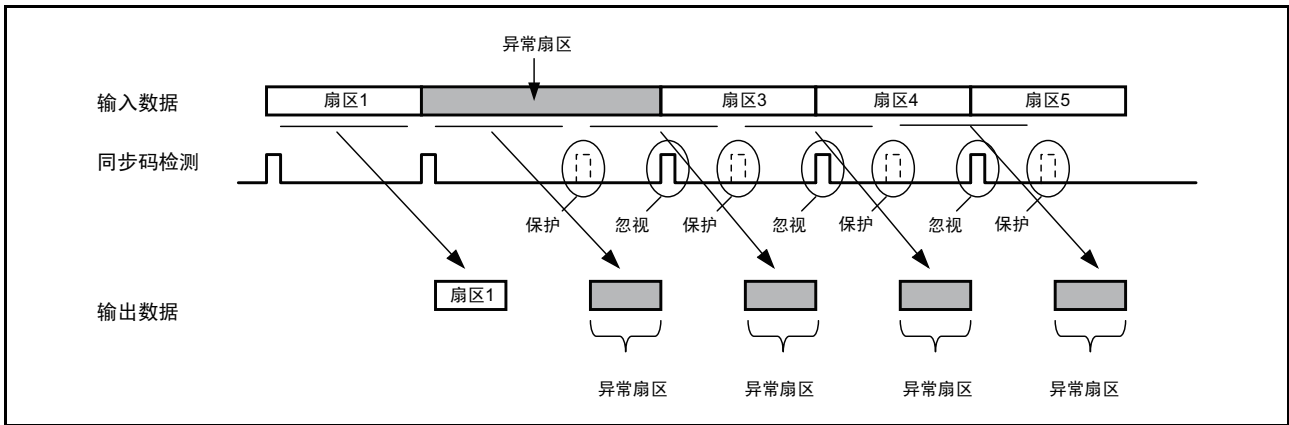


图 26.12 内插同步模式中的运行

(4) 内插 / 外部同步模式

在内插 / 外部同步模式中，如果检测到同步码模式，就与其同步。如果在 2352 字节内没有发现同步模型，就保护同步码。和自动同步保护模式相比，它能立即跟随同步时序的改变。

但是，如果在异常时序进入同步模型，就不能进行正常的解码。

发生短扇区和长扇区时的内插 / 外部同步模式的运行分别如图 26.13 和图 26.14 所示。

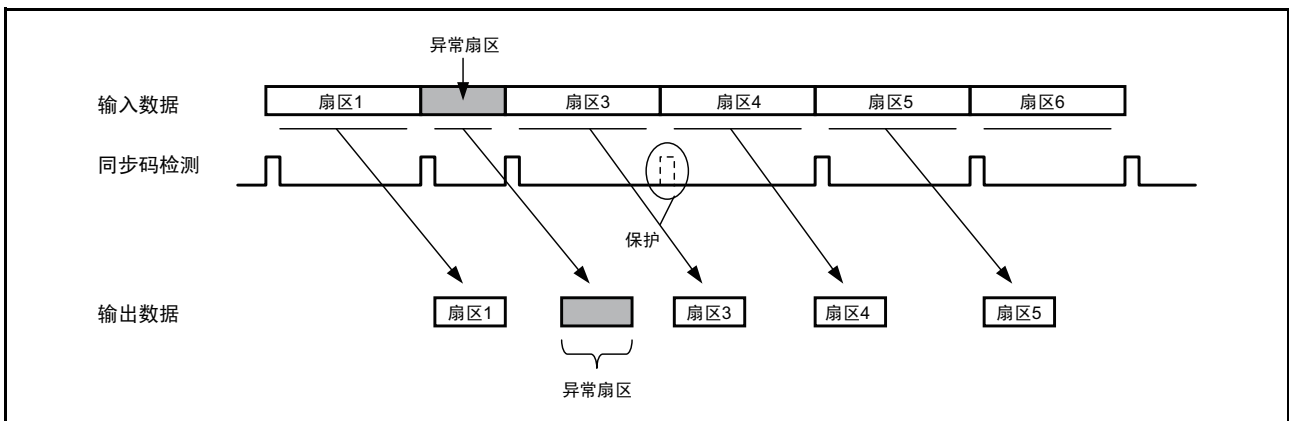


图 26.13 短扇区时的内插 / 外部同步模式的运行

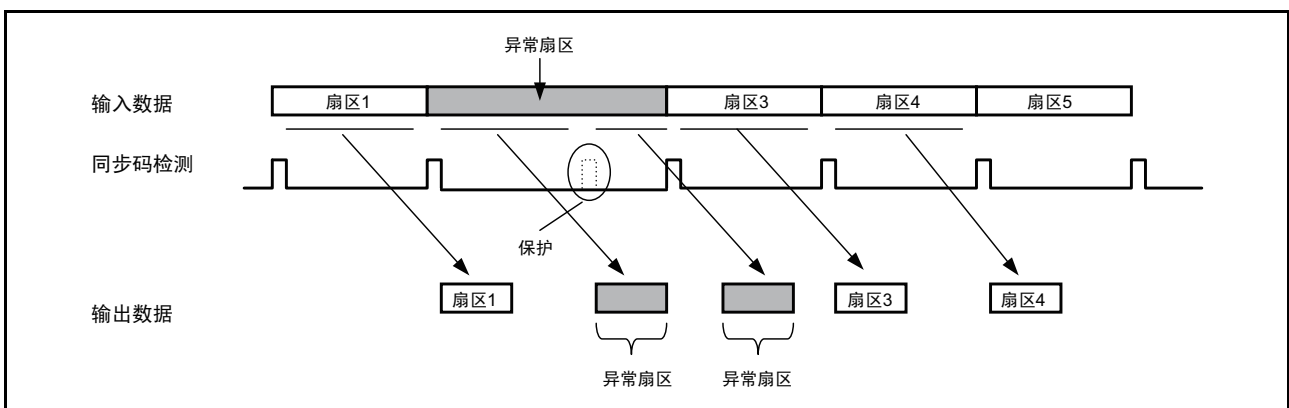


图 26.14 长扇区时的内插 / 外部同步模式的运行

26.4.3 纠错

CD-ROM 解码器有用于纠错的 EDC、P 奇偶校验、Q 奇偶校验的信息格式。此 CD-ROM 解码器有以下纠错功能：

- 校正子计算功能
- ECC 校正功能
- EDC 校验功能

(1) 校正子计算功能

在 CD-ROM 解码器为 Mode1 或者 Mode2 Form1 的情况下，如果在输入 1 个扇区的数据后判定出现错误（校正子计算结果不为“0”），就进行 ECC 校正。在 ECC 校正后，将修正后的数据校正子计算结果输出到 CROMST6 寄存器的 ST_ECCP 位（P 系列）和 ST_ECCQ 位（Q 系列）。

(2) ECC 校正和 EDC 校验

对含 EDC、P 奇偶校验、Q 奇偶校验的 CD-ROM 格式数据进行 ECC 校正和 EDC 校验。ECC 的校正模式支持 P 校正、Q 校正、PQ 校正（在 P 校正后进行 Q 校正）和 QP 校正（在 Q 校正后进行 P 校正），PQ 校正和 QP 校正能最多重复 3 次（根据倍速有限制）。

在 ECC 校正前后进行 2 次 EDC 校验。

通过 CROMCTL1 寄存器的 MD_DEC[2:0] 位设定 ECC 校正和 EDC 校验的校验模式，通过 CROMCTL1 寄存器的 MD_PQREP[1:0] 位设定选择 PQ 校正和 QP 校正模式时的校正重复次数。

在使用自动模式和格式检测功能时，根据判定的模式进行 ECC 校正和 EDC 校验。在判定为没有 P 奇偶校验、Q 奇偶校验和 EDC 的 Mode0 或者 Mode2 时，不进行 ECC 校正和 EDC 校验，并且在判定为 Mode2 Form2 时也不进行 ECC 校正。

(a) ECC 校正

在使用 ECC 校正时，如果判定为无法校正，就产生 IERR 中断，并且 CROMST6 寄存器的 ST_ECCNG 位为“1”。另外，即使检测到短扇区，此位也为“1”。

在 CROMCTL4 寄存器的 NO_ECC 位被设定为“1”时，如果 ECC 前的 EDC 校验 OK，就不进行 ECC 校正，与校正子计算结果无关。

(b) EDC 校验

在使用 EDC 校验时，根据设定的模式和格式进行 EDC 校验。另外，在使用自动模式和格式检测功能时，根据判定的模式和格式进行 EDC 校验。

ECC 校正前后的 EDC 校验结果分别反映在 CROMST6 寄存器的 ST_EDC1 位和 ST_EDC2 位。如果 ECC 校正后的 EDC 校验结果为 NG，就产生 IERR 中断。

26.4.4 自动解码的停止功能

如果在对 CD-ROM 数据进行解码的过程中发生异常，就能自动停止解码。有以下自动停止的停止源。通过 CROMCTL3 寄存器设定有效源。

- 无法进行 ECC 校正时
- ECC 后的 EDC 校验为 NG 时
- 模式和格式发生变化时
- MSF（分、秒、帧（1/75 秒））不连续时

当因上述停止源而自动停止时，就在输出发生停止源的扇区后停止解码。

如果发生 CROMCTL3 寄存器设定的停止源而停止解码，就能通过 CBUFST1 寄存器确认发生自动停止的停止源。

另外，在 CD-ROM 解码器内部有 2 个扇区的缓冲器。如果不读输出流数据而连续输入流数据，就在输入第 3 个扇区的数据时 CD-ROM 解码器停止运行。此时，CBUFST2 寄存器的 BUF_NG 位为“1”，不产生中断。在 CBUFST2 寄存器的 BUF_NG 位变为“1”时，如果不通过 ROMDECRST 寄存器的 LOGICRST 位清除 CD-ROM 解码器功能，就不能恢复。通过将 ROMDECRST 寄存器的 LOGICRST 位置“1”，输出复位信号，并且将设定的寄存器也清除为初始值。

26.4.5 缓冲格式

CD-ROM 解码后的输出流数据格式如图 26.15 所示。

CD-ROM 解码器的输出有 2 字节的窗口寄存器 STRMDOUT0，如果在 CD-ROM 解码结束后存取该寄存器，就从同步码开始按顺序输出。

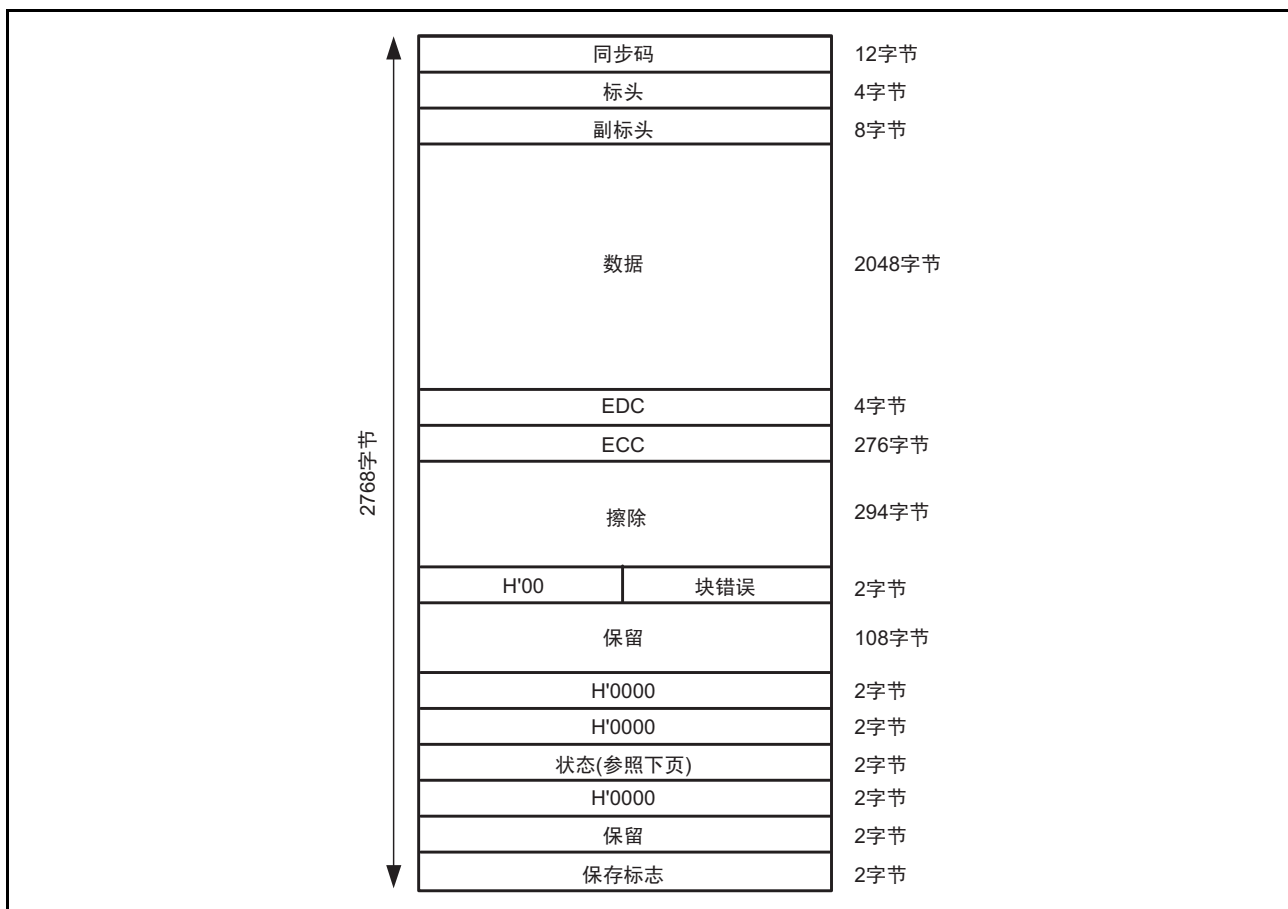


图 26.15 输出流数据格式

图 26.15 中 2 字节状态的含义如下。未分配的位的状态不确定。

状态															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PERR	QERR	EDCE	—	—	—	—	—	SD	SY	FM[2:0]			HD	—	—

PERR: 表示 P 系列有错误。

QERR: 表示 Q 系列有错误。

EDCE: 表示 ECC 校正后的 EDC 校验为 NG。

SD: 表示为短扇区。

SY: 表示内插同步码。

FM: 表示数据格式。

001: Mode0

010: Mode1

011: Long (没有 EDC 和 ECC 的格式)

100: Mode2 notXA

101: Mode2 Form1

111: Mode2 Form21

HD: 标头的连续性 (分、秒、帧 (1/75 秒) 为 NG)

每输出 1 个扇区的数据, 图 26.15 中的保存标志就进行 H'0000 ~ H'FFFF 的递增计数 (为 H'FFFF 后递增计数到 H'0000)。另外, 必须注意, 只有 2 字节的保存标志进行高字节和低字节的交换。

26.4.6 目标扇区的缓冲功能

作为指定要输出扇区的方法，CD-ROM 解码器有以下 2 种功能：

1. 自动缓冲功能：事先设定目标扇区，CD-ROM 解码器自身检测目标扇区。
2. 手动缓冲功能：通过软件边认识 CD-ROM 解码器内部被缓冲的扇区边指示目标扇区的输出。

以下说明用于自动和手动缓冲的 CD-ROM 解码器内部寄存器的设定方法。

(1) 自动缓冲设定

自动缓冲的设定例子如图 26.16 所示。

如果设定 CD-ROM 解码器寄存器并且输入流数据，CD-ROM 解码器就检测目标扇区并且开始输出流数据。

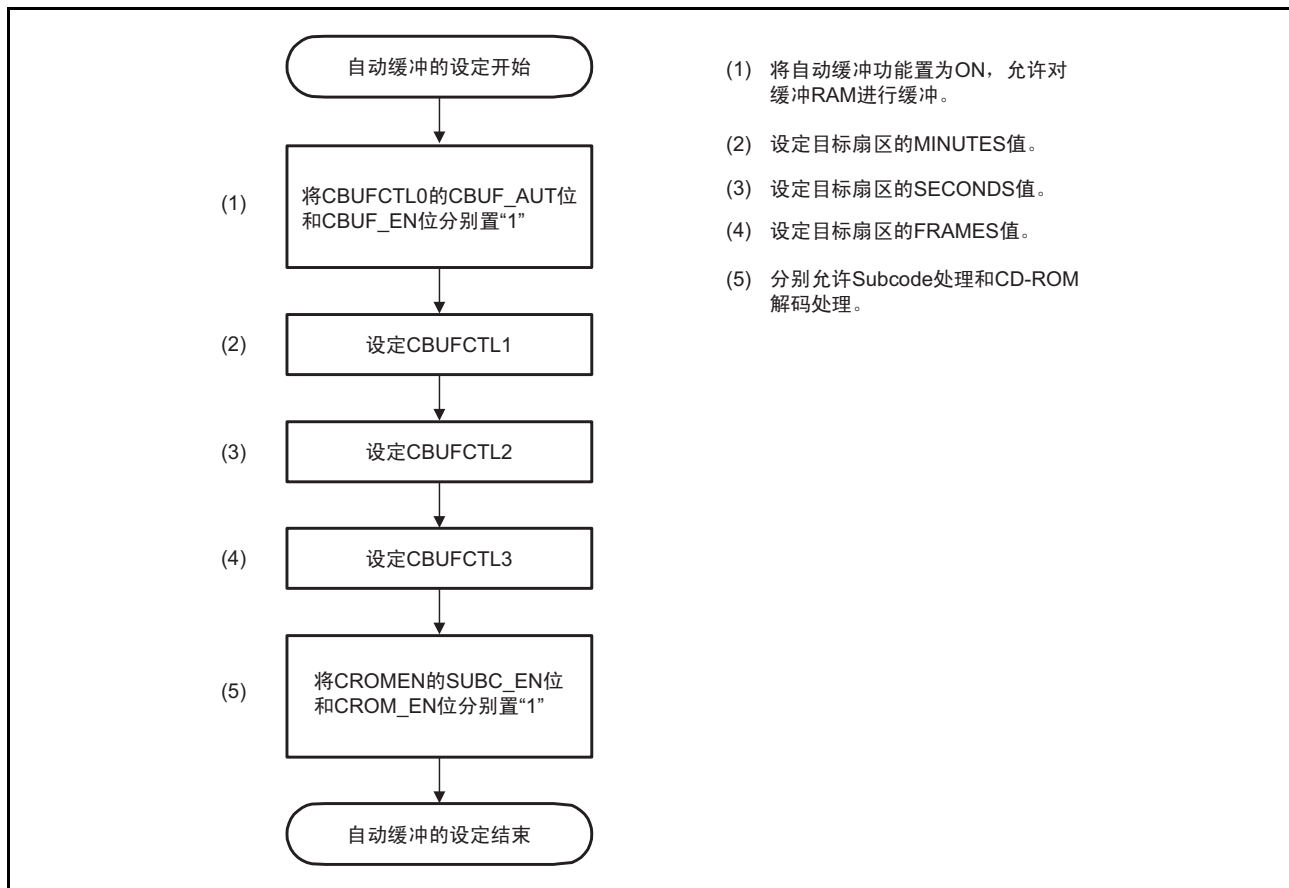


图 26.16 自动缓冲的设定例子

(2) 手动缓冲设定

手动缓冲的设定例子如图 26.17 所示。

每次发生 ISEC 中断时，判定是否是目标扇区，在变为目标扇区后将缓冲置为 ON。

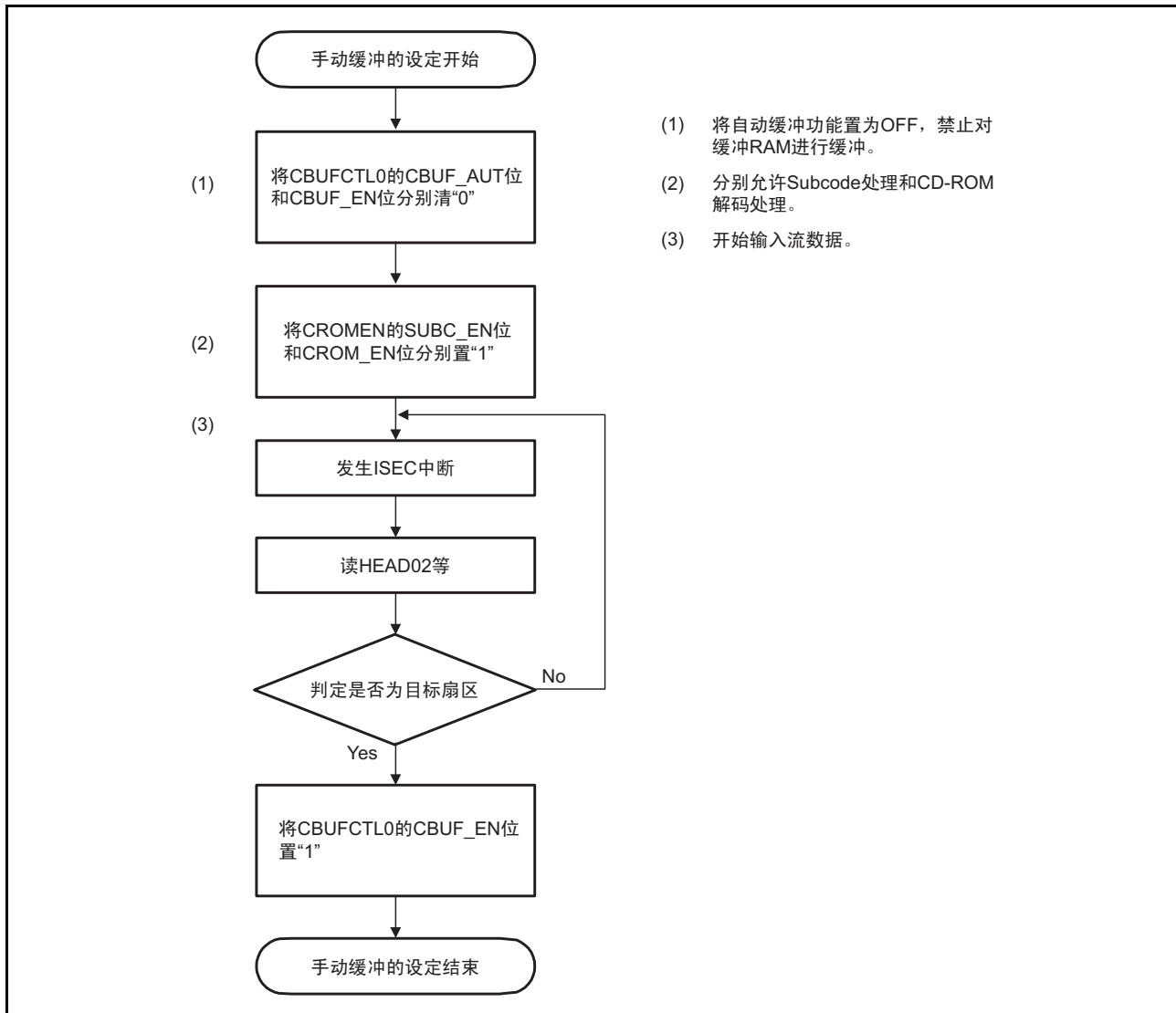


图 26.17 手动缓冲的设定例子

26.5 中断源

26.5.1 中断和 DMA 传送请求信号

CD-ROM 解码器输出的中断信号和请求 DMA 传送信号的含义和连接目标如表 26.3 所示。

表 26.3 中断和 DMA 请求一览

名称	条件	连接目标
ISEC	当切换扇区时	中断控制器
ITARG	当 CD-ROM 的存取扇区不是目标扇区时	中断控制器
ISY	CD-ROM 同步码出现异常时	中断控制器
IERR	不能进行 ECC 校正或者 ECC 校正后的 EDC 校验发生错误时	中断控制器
IBUF	当数据传送到缓冲器的状况发生变化时	中断控制器
IREADY	向 CD-ROM 缓冲器请求数据传送	中断控制器
DMA 传送请求	向 CD-ROM 缓冲器请求数据传送	直接存储器存取控制器

(1) ISEC 中断

在将同步码开始的 1 个扇区切换到下一个扇区时产生中断。

(2) ITARG 中断

在 CD-DSP 送来的流数据不是目标扇区的数据时产生中断。CD-ROM 解码器检查子代码中的时间，在正常状态下希望从目标扇区稍前一些的时序开始数据传送，但是在以下情况下产生中断：

- 当传送了在目标扇区非常前面的数据时
- 当传送了目标扇区后的数据时

此中断从子代码检测 ITARG，但是本 LSI 从串行音频接口传送 CD-ROM 数据，因此此中断没有意义。

(3) ISY 中断

在以下情况下产生中断：

- 在字计数器（同步码间隔检查计数器）不正确时检测到同步码，但是该同步码被忽视时
- 字计数器为最后值，但是不检测同步码而进行内插（保护）时
- 在字计数器不正确时检测到同步码并且通过该同步码重新进行同步时
- 字计数器为最后值，但是不检测同步码而延长 1 个扇区的期间时
- 同步码和内插时序为短扇区时
- 同步码和内插时序为长扇区时

(4) IERR 中断

在以下情况下产生中断：

- ECC 不能校正时
- ECC 校正 OK 而 ECC 校正后的 EDC 检查发生错误时

(5) IBUF 中断

在进行以下切换时产生中断：

- 正在将数据传送到环形缓冲器 → 数据传送结束（正在检索下一个传送数据）
- 正在检索要传送到环形缓冲器的数据 → 开始数据传送

(6) IREADY 中断

在 1 个扇区的 CD-ROM 解码结束时产生中断。在通过 CPU 将输出流数据缓冲到 SDRAM 时，此中断用作启动信号。

(7) DMA 传送请求

直接存储器存取控制器的启动源和 IREADY 相同。如果准备好 1 个扇区的输出流数据，就产生中断请求。一旦传送完如图 26.15 所示的 2768 字节的数据，就将请求信号置为无效。这是因为需要准备下一个扇区的输出流数据的时间而使直接存储器存取控制器暂停数据传送请求。

26.5.2 状态寄存器的更新时序

每当发生 ISEC 中断时，更新 CD-ROM 解码器的状态信息寄存器。通过 CROMCTL4 寄存器的 ER0SEL 位设定反映哪个扇区的的信息。

26.6 使用时的注意事项

26.6.1 在进行解码的状态下只停止或者重新开始缓冲时的注意点

如果保持 CD-ROM 解码器的运行而不读输出流数据，CBUFST2 寄存器的 BUF_NG 位就为“1”，然后 CD-ROM 解码器无法运行。

要只停止缓冲时，将 CBUFCTL0 寄存器的 CBUF_EN 位置“0”。在使用自动缓冲功能时，将 CBUFCTL0 寄存器的 CBUF_AUT 位置“0”。此时，必须读当前正在进行缓冲的扇区。

要重新开始缓冲时，如果使用自动缓冲功能，就必须将 CBUFCTL0 寄存器的 CBUF_AUT 位和 CBUF_EN 位同时置“1”。

26.6.2 设定同步码状态寄存器（CROMST0）时的注意点

1. 如果 CROMST0 寄存器的 ST_SECS 位为“1”，就必须立即停止解码，并且从当前正在解码的前 1 个扇区开始尝试解码。
2. 如果 CROMST0 寄存器的 ST_SECL 位为“1”，就必须立即停止解码，并且从当前正在解码的前 2 个扇区开始尝试解码。

26.6.3 链路块的注意点

此 CD-ROM 解码器的链路块检测功能使用 ECC 校正前的标头信息进行链路块的判断。因此，当输入流数据出现错误时，可能会错误地检测到链路块。为了防止这种情况，必须通过软件采取以下的对策：

- 在进行缓冲的过程中（CBUFST0 寄存器 BUF_ACT 位为“1”），每当发生 ISEC 中断时，必须确认 CROMST5 寄存器 LINK_OUT1 位。如果该位为“1”，就保存 HEAD20～23 寄存器的 MSF 值。此时，必须确认没有发生 IERR 中断。如果发生 IERR 中断，就不保存 MSF 值。
- 在确认上述的 LINK_OUT1 位为“1”后，必须在 7 个扇区（发生 7 次 ISEC 中断）期间进行以下的处理：
 1. 每次发生 ISEC 中断时，确认 CROMST5 寄存器的 LINK_ON 位为“1”，并且在此后的 2 个扇区期间再次检测到 LINK_ON 位为“1”。
 2. 1 次也没检测到 LINK_ON 位为“1”。

在上述 1. 和 2. 的情况下，强制结束解码，并且在将 CROMSY0 寄存器置为外部同步模式后，将上述保存的目标扇区的 MSF 值 +7（紧接在链路块后面）作为目标扇区重新开始尝试解码。

在 CBUFCTL0 寄存器的 CBUF_LINK 位为“0”时，起始扇区地址为 run-out 的保存地址 +7；在 CBUF_LINK 位为“1”时，起始扇区地址为 run-out 的保存地址。

26.6.4 停止和重新开始 CD-DSP 时的注意点

在停止和重新开始将流数据输入到 CD-ROM 解码器时，停止时的流数据不会在同步码前停止，而在此后重新开始时可能被认为是非法的流数据。这是因为停止时输入的流数据和重新开始时输入的流数据混在一起，所以在停止和重新开始输入流数据时必须注意。

26.6.5 清除 IREADY 标志时的注意点

在通过中断处理等清除 IREADY 标志时，必须在读 1 个扇区（2768 字节）的数据后将该标志清“0”。如果在读完 1 个扇区的数据前将 IREADY 标志清“0”，就不能对此后的扇区进行解码。此时，如果要恢复，就必须在通过给 CD-ROM 解码器模块复位控制寄存器（ROMDECRST）的 LOGICRST 位写“1”后写“0”。

26.6.6 传送流数据时的注意点（1）

如果读流数据的速度慢于写流数据的速度，CD-ROM 解码器就陷入缓冲器上溢状态而异常停止。必须注意流数据的读写传送。流数据传送设定的组合例子如下所示。

表 26.4 流数据传送设定的组合例子

流输入	流输出
通过直接存储器存取控制器进行的 LW/ 周期挪用传送（无填充）	(1) 通过直接存储器存取控制器进行的 16 字节 / 周期挪用传送（16 字节*） (2) 通过直接存储器存取控制器进行的突发传送（16 字节*、长字、字）
通过直接存储器存取控制器进行的 LW/ 周期挪用传送（有填充）	(1) 通过直接存储器存取控制器进行的周期挪用传送（16 字节*、长字） (2) 通过直接存储器存取控制器进行的突发传送（16 字节*、长字、字）
通过 CPU 写 LW	(1) 通过直接存储器存取控制器进行的周期挪用传送（16 字节*、长字、字） (2) 通过直接存储器存取控制器进行的突发传送（16 字节*、长字、字）

【注】* 除了通常的 16 字节传送设定以外，还必须将 DMA 通道控制寄存器（CHCR_n）的 bit25 置“1”。

26.6.7 传送流数据时的注意点（2）

必须通过直接存储器存取控制器或者 CPU 读流数据。当直接存储器存取控制器和 CPU 的读操作混在一起时，可能无法将流数据识别为 CD-ROM 的格式。

27. A/D 转换器

本 LSI 内置逐次逼近方式的 10 位精度 A/D 转换器，最多能选择 8 个通道的模拟输入。

27.1 特点

- 分辨率：10 位
- 输入通道：SH7268 为 6 个通道，SH7269 为 8 个通道。
- 最小转换时间：每个通道 6.0 μ s
- 绝对精度： ± 5 LSB
- 运行模式：3 种
 - 单通道模式：1 个通道的 A/D 转换
 - 多通道模式：1~4 个通道的 A/D 转换或者 1~8 个通道的 A/D 转换（SH7268 为 6 个通道）
 - 扫描模式：1~4 个通道的连续 A/D 转换或者 1~8 个通道的连续 A/D 转换（SH7268 为 6 个通道）
- 数据寄存器：8 个
 - 将转换结果保存到各通道对应的 16 位数据寄存器。
- 采样 & 保持功能
- A/D 转换开始方法：3 种
 - 软件
 - 通过多功能定时器脉冲单元 2（多功能定时器脉冲单元 2）的开始转换触发外部触发信号
- 中断源
 - 能在 A/D 转换结束时产生 A/D 转换结束的中断（ADI）请求。
- 能设定模块待机模式。

框图如图 27.1 所示。

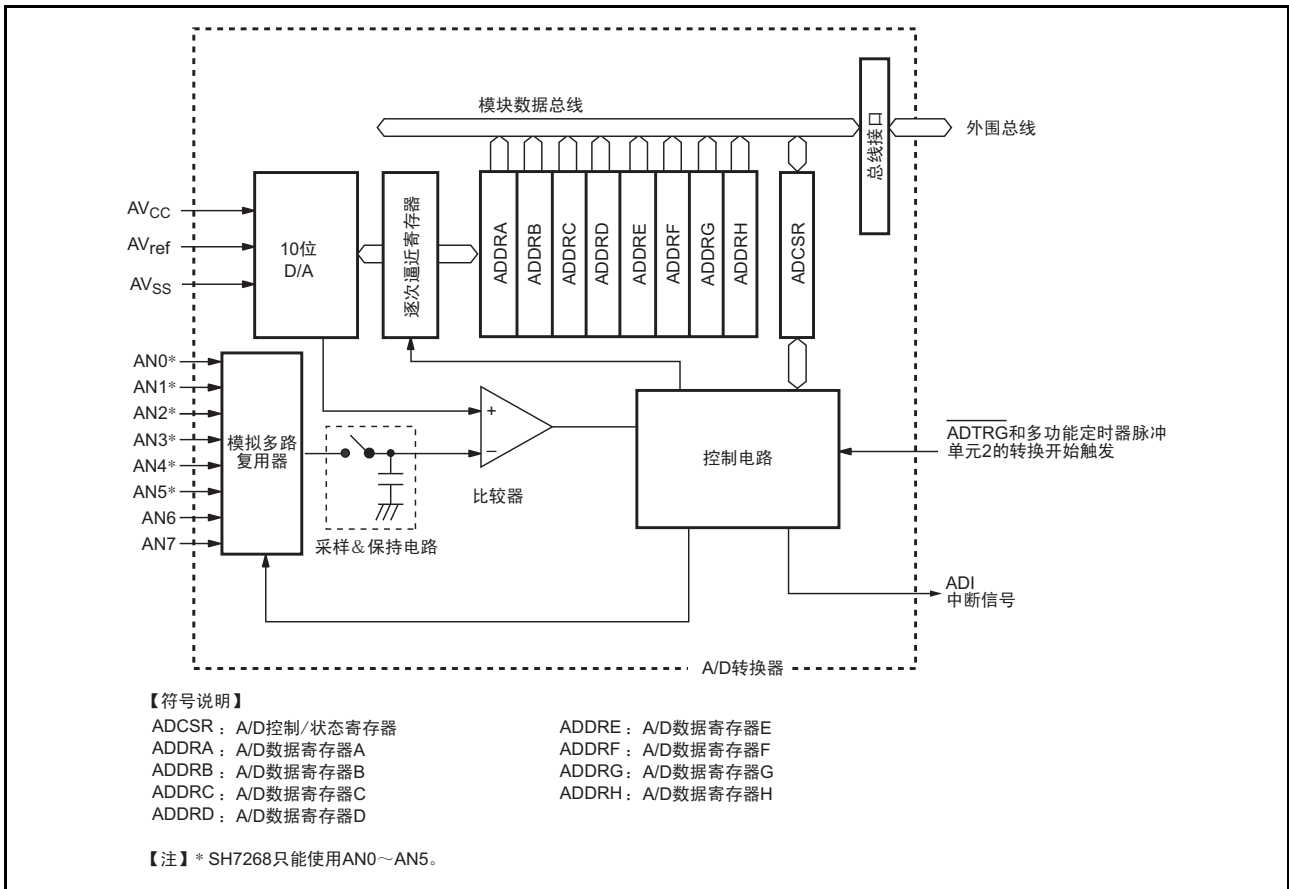


图 27.1 框图

27.2 输入 / 输出引脚

引脚结构如表 27.1 所示。

表 27.1 引脚结构

名称	引脚名	输入 / 输出	功能
模拟电源引脚	AV _{CC}	输入	模拟部的电源引脚
模拟接地引脚	AV _{SS}	输入	模拟部的接地引脚和 A/D 转换的基准接地
模拟基准电压引脚	AV _{ref}	输入	A/D 转换器的基准电压引脚
模拟输入引脚 0*	AN0	输入	模拟输入
模拟输入引脚 1*	AN1	输入	
模拟输入引脚 2*	AN2	输入	
模拟输入引脚 3*	AN3	输入	
模拟输入引脚 4*	AN4	输入	
模拟输入引脚 5*	AN5	输入	
模拟输入引脚 6	AN6	输入	
模拟输入引脚 7	AN7	输入	
A/D 外部触发输入引脚	ADTRG	输入	开始 A/D 转换的外部触发输入

【注】 * SH7268 只能使用模拟输入引脚 0 ~ 5 (AN0 ~ AN5)。

27.3 寄存器说明

寄存器结构如表 27.2 所示。

表 27.2 寄存器结构

寄存器名	略称	R/W	初始值	地址	存取长度
A/D 数据寄存器 A	ADDRA	R	H'0000	H'E8005800	16
A/D 数据寄存器 B	ADDRB	R	H'0000	H'E8005802	16
A/D 数据寄存器 C	ADDRC	R	H'0000	H'E8005804	16
A/D 数据寄存器 D	ADDRD	R	H'0000	H'E8005806	16
A/D 数据寄存器 E	ADDRE	R	H'0000	H'E8005808	16
A/D 数据寄存器 F	ADDRF	R	H'0000	H'E800580A	16
A/D 数据寄存器 G	ADDRG	R	H'0000	H'E800580C	16
A/D 数据寄存器 H	ADDRH	R	H'0000	H'E800580E	16
A/D 控制 / 状态寄存器	ADCSR	R/W	H'0000	H'E8005820	16

27.3.1 A/D 数据寄存器 A ~ H (ADDRA ~ ADDRH)

ADDR 是用于保存 A/D 转换结果的 16 位只读寄存器，有 ADDRA ~ ADDRH 共 8 个寄存器。

A/D 转换的数据为 10 位数据，传送并且保存到转换通道对应的 ADDR 的 bit15 ~ 6。bit5 ~ 0 的读取值总是“0”。

禁止以 8 位为单位存取 ADDR，必须总是以 16 位为单位进行存取。

模拟输入通道和 ADDR 的对应关系如表 27.3 所示。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
											—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 6		全 0	R	位数据 (10 位)
5 ~ 0	—	全 0	R	保留位 读写值都为“0”。

表 27.3 模拟输入通道和 ADDR 的对应关系

模拟输入通道	保存转换结果的 A/D 数据寄存器
AN0*	ADDRA
AN1*	ADDRB
AN2*	ADDRC
AN3*	ADDRD
AN4*	ADDRE
AN5*	ADDRF
AN6	ADDRG
AN7	ADDRH

【注】 * SH7268 只能使用 AN0 ~ AN5。

27.3.2 A/D 控制 / 状态寄存器 (ADCSR)

ADCSR 是 16 位可读写寄存器，选择运行模式，控制 A/D 转换运行以及允许或者禁止通过输入外部触发开始 A/D 转换。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADF	ADIE	ADST	TRGS[3:0]				CKS[1:0]			MDS[2:0]		CH[2:0]			
初始值:	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
R/W:	R/(W)*1	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

【注】 *1 为了清除标志，只能在读“1”后写“0”。

位	位名	初始值	R/W	说明
15	ADF	0	R/(W)	A/D 结束标志 这是表示 A/D 转换结束的状态标志。 [清除条件] • 在 ADF 位为“1”的状态下读 ADF 标志后给 ADF 标志写“0”时 • 通过 ADI 中断启动直接存储器存取控制器并且读 ADDR 时 [置位条件] • 在单通道模式中，当 A/D 转换结束时 • 在多通道模式中，当所选通道的 A/D 转换全部结束时 • 在扫描模式中，当所选通道的 A/D 转换全部结束时
14	ADIE	0	R/W	A/D 中断允许 选择允许或者禁止 A/D 转换结束引起的中断 (ADI) 请求。 必须在停止转换时设定 ADIE 位。 0: 禁止 A/D 转换结束引起的中断 (ADI) 请求 1: 允许 A/D 转换结束引起的中断 (ADI) 请求
13	ADST	0	R/W	A/D 开始 选择开始或者停止 A/D 转换。在 A/D 转换过程中保持“1”。 0: 停止 A/D 转换 1: 单通道模式: 开始 A/D 转换。如果指定通道的 A/D 转换结束，就自动清除。 多通道模式: 开始 A/D 转换。如果指定通道的一次 A/D 转换全部结束，就自动清“0”。 扫描模式时: 开始 A/D 转换。连续进行转换，直到进入软件复位、上电复位、深度待机模式、软件待机模式或者模块待机模式而将此位清“0”为止。
12 ~ 9	TRGS[3:0]	0000	R/W	定时器触发选择 选择允许或者禁止通过触发信号开始 A/D 转换。 0000: 禁止通过外部触发开始 A/D 转换 0001: 通过多功能定时器脉冲单元 2 的转换触发 TRGAN 开始 A/D 转换 0010: 通过多功能定时器脉冲单元 2 的转换触发 TRG0N 开始 A/D 转换 0011: 通过多功能定时器脉冲单元 2 的转换触发 TRG4AN 开始 A/D 转换 0100: 通过多功能定时器脉冲单元 2 的转换触发 TRG4BN 开始 A/D 转换 1001: 通过 ADTRG 开始 A/D 转换 上述以外: 禁止设定

位	位名	初始值	R/W	说明																														
8 ~ 6	CKS[2:0]	000	R/W	<p>时钟选择</p> <p>设定 A/D 转换时间*1。必须在停止转换时 (ADST 位为“0”) 设定转换时间。</p> <p>000: 转换时间 =412t_{cyc} (最大值)</p> <p>001: 转换时间 =480t_{cyc} (最大值)</p> <p>010: 转换时间 =548t_{cyc} (最大值)</p> <p>011、100、101、110、111、: 禁止设定</p>																														
5 ~ 3	MDS[2:0]	000	R/W	<p>多通道扫描模式</p> <p>选择 A/D 转换的运行模式。</p> <p>0xx: 单通道模式</p> <p>100: 多通道模式, 1 ~ 4 个通道的 A/D 转换。</p> <p>101: 多通道模式, 1 ~ 8 个通道的 A/D 转换。</p> <p>110: 扫描模式, 1 ~ 4 个通道的 A/D 转换。</p> <p>111: 扫描模式, 1 ~ 8 个通道的 A/D 转换。</p>																														
2 ~ 0	CH[2:0]	000	R/W	<p>通道选择</p> <p>和 ADCSR 的 MDS 位一起选择模拟输入。</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 33%;"></td> <td style="width: 33%; text-align: center;">MDS=100 或者</td> <td style="width: 33%; text-align: center;">MDS=101 或者</td> </tr> <tr> <td style="text-align: center;">MDS=0xx 时</td> <td style="text-align: center;">MDS=110 时</td> <td style="text-align: center;">MDS=111 时</td> </tr> <tr> <td>000: AN0</td> <td>000: AN0</td> <td>000: AN0</td> </tr> <tr> <td>001: AN1</td> <td>001: AN0、AN1</td> <td>001: AN0、AN1</td> </tr> <tr> <td>010: AN2</td> <td>010: AN0 ~ AN2</td> <td>010: AN0 ~ AN2</td> </tr> <tr> <td>011: AN3</td> <td>011: AN0 ~ AN3</td> <td>011: AN0 ~ AN3</td> </tr> <tr> <td>100: AN4</td> <td>100: AN4</td> <td>100: AN0 ~ AN4</td> </tr> <tr> <td>101: AN5</td> <td>101: AN4、AN5</td> <td>101: AN0 ~ AN5</td> </tr> <tr> <td>110: AN6*2</td> <td>110: AN4 ~ AN6*2</td> <td>110: AN0 ~ AN6*2</td> </tr> <tr> <td>111: AN7*2</td> <td>111: AN4 ~ AN7*2</td> <td>111: AN0 ~ AN7*2</td> </tr> </table>		MDS=100 或者	MDS=101 或者	MDS=0xx 时	MDS=110 时	MDS=111 时	000: AN0	000: AN0	000: AN0	001: AN1	001: AN0、AN1	001: AN0、AN1	010: AN2	010: AN0 ~ AN2	010: AN0 ~ AN2	011: AN3	011: AN0 ~ AN3	011: AN0 ~ AN3	100: AN4	100: AN4	100: AN0 ~ AN4	101: AN5	101: AN4、AN5	101: AN0 ~ AN5	110: AN6*2	110: AN4 ~ AN6*2	110: AN0 ~ AN6*2	111: AN7*2	111: AN4 ~ AN7*2	111: AN0 ~ AN7*2
	MDS=100 或者	MDS=101 或者																																
MDS=0xx 时	MDS=110 时	MDS=111 时																																
000: AN0	000: AN0	000: AN0																																
001: AN1	001: AN0、AN1	001: AN0、AN1																																
010: AN2	010: AN0 ~ AN2	010: AN0 ~ AN2																																
011: AN3	011: AN0 ~ AN3	011: AN0 ~ AN3																																
100: AN4	100: AN4	100: AN0 ~ AN4																																
101: AN5	101: AN4、AN5	101: AN0 ~ AN5																																
110: AN6*2	110: AN4 ~ AN6*2	110: AN0 ~ AN6*2																																
111: AN7*2	111: AN4 ~ AN7*2	111: AN0 ~ AN7*2																																

【符号说明】 x: Don't care

【注】 *1 为了满足 A/D 转换器特性的绝对精度, 必须至少设定最小转换时间的值。

*2 在 SH7268 中, 禁止设定。

27.4 运行说明

A/D 转换器采用逐次逼近方式，分辨率为 10 位。运行模式有单通道模式、多通道模式和扫描模式。当要切换运行模式或者模拟输入通道时，为了避免误动作，必须在 ADCSR 的 ADST 位为“0”的状态下进行切换。能在将 ADST 位置位的同时更改运行模式或者模拟输入通道。

27.4.1 单通道模式

在只进行 1 个通道的 A/D 转换时，选择单通道模式。

在单通道模式中，对指定的 1 个通道的模拟输入进行 1 次 A/D 转换，运行如下：

1. 如果通过软件、多功能定时器脉冲单元 2 或者外部触发输入将 ADCSR 的 ADST 位置“1”，就开始对所选通道进行 A/D 转换。
2. 如果 A/D 转换结束，就将 A/D 转换结果传送到该通道对应的 A/D 数据寄存器。
3. 在 A/D 转换结束后，将 ADCSR 的 ADF 位置“1”。此时，如果 ADIE 位已被置“1”，就产生 ADI 中断请求。
4. ADST 位在 A/D 转换过程中保持“1”，当转换结束时，就自动清“0”并且 A/D 转换器进入待机状态。

要在 A/D 转换过程中切换模式和模拟输入通道时，为了避免误动作，必须通过将 ADST 位清“0”，在停止 A/D 转换的状态下进行切换。如果在更改后将 ADST 位置“1”（能在将 ADST 位置位的同时更改模式或者通道），就重新开始 A/D 转换。

在单通道模式中，选择通道 1（AN1）的运行例子如下所示，此时的运行时序如图 27.2 所示（由 ADCSR 寄存器指定运行例子中的位）。

1. 将运行模式设定为单通道模式，将输入通道设定为 AN1（CH[2:0] 位为“001”）并且设定为允许 A/D 中断请求（ADIE 位为“1”），开始 A/D 转换（ADST 位为“1”）。
2. 如果 A/D 转换结束，就将 A/D 转换结果传送到 ADDR0。同时，ADF 位变为“1”并且 ADST 位变为“0”，A/D 转换器进入转换待机状态。
3. 因为 ADF 位为“1”并且 ADIE 位为“1”，所以产生 ADI 中断请求。
4. 开始执行 A/D 中断处理程序。
5. 在读 ADF 的“1”后给 ADF 写“0”。
6. 读 A/D 转换结果（ADDR0）并且进行处理。
7. 结束 A/D 中断处理程序的执行。然后，如果将 ADST 位置“1”，就开始 A/D 转换，执行步骤 2.~7.。

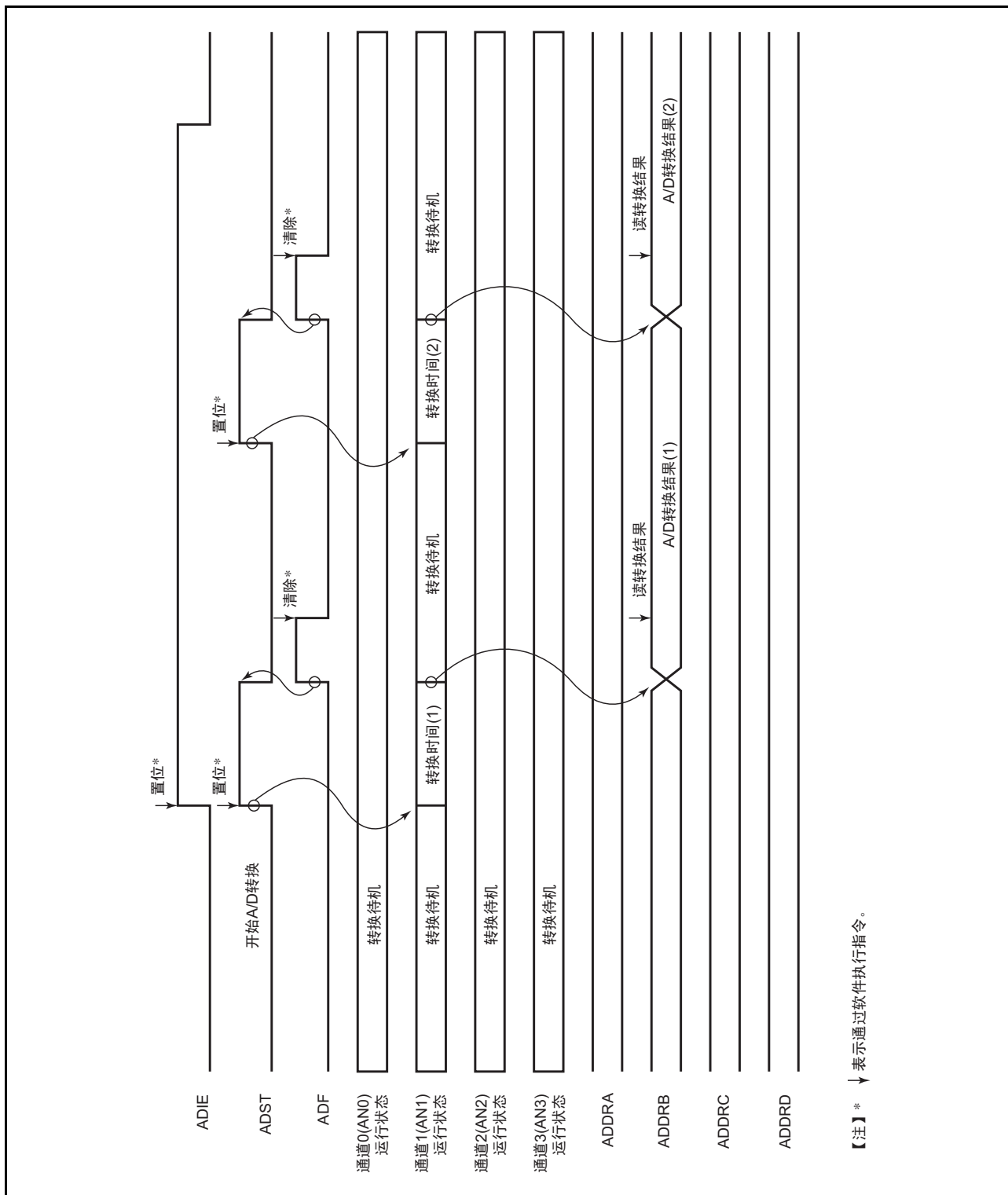


图 27.2 A/D 转换器的运行例子 (选择单通道模式和通道 1 的情况)

27.4.2 多通道模式

多通道模式依次对多个通道（包含 1 个通道）的模拟输入进行 1 次转换。

在多通道模式中，对指定的最多 8 个通道的模拟输入进行 1 次 A/D 转换，运行如下：

1. 如果通过软件、多功能定时器脉冲单元 2 或者外部触发输入将 ADCSR 的 ADST 位置 “1”，就按模拟输入通道号从小到大的顺序（例如：AN0、AN1、……、AN3）进行 A/D 转换。
2. 如果各通道的 A/D 转换结束，就按顺序将 A/D 转换结果传送到该通道对应的 A/D 数据寄存器。
3. 如果所选通道的 A/D 转换全部结束，就将 ADCSR 的 ADF 位置 “1”。此时，如果 ADIE 位已被置 “1”，就产生 ADI 中断请求。
4. ADST 位在 A/D 转换过程中保持 “1”，当转换结束时，自动清 “0” 并且 A/D 转换器进入待机状态。如果在 A/D 转换过程中将 ADST 位清 “0”，就中止转换并且 A/D 转换器进入待机状态。通过在读 ADF 位的 “1” 后给 ADF 位写 “0”，清除 ADF 位。

指定的通道全部进行一次 A/D 转换，将转换的结果传送并且保存到各通道对应的 ADDR。

要在 A/D 转换过程中切换模式和模拟输入通道时，为了避免误动作，必须通过将 ADST 位清 “0”，在停止 A/D 转换的状态下进行切换。如果在更改后将 ADST 位置 “1”（能在将 ADST 位置位的同时更改模式或者通道），就选择第 1 个通道，重新开始 A/D 转换。

在多通道模式中，选择 3 个通道（AN0 ~ AN2）进行 A/D 转换的运行例子如下所示，此时的运行时序如图 27.3 所示。

1. 将运行模式设定为多通道模式（MDS[2]位为 “1”，MDS[1]位为 “0”）并且将模拟输入通道设定为 AN0 ~ AN2（CH[2:0]位为 “010”），开始 A/D 转换（ADST 位为 “1”）。
2. 开始第 1 个通道（AN0）的 A/D 转换，如果 A/D 转换结束，就将转换结果传送到 ADDR。
3. 接着，自动选择第 2 个通道（AN1）开始转换。
4. 同样，转换到第 3 通道（AN2）为止。
5. 如果所选通道（AN0 ~ AN2）的转换全部结束，ADF 位就变为 “1” 并且将 ADST 位置 “0”，结束转换。
6. 此时，如果 ADIE 位为 “1”，就在 A/D 转换结束后产生 ADI 中断。

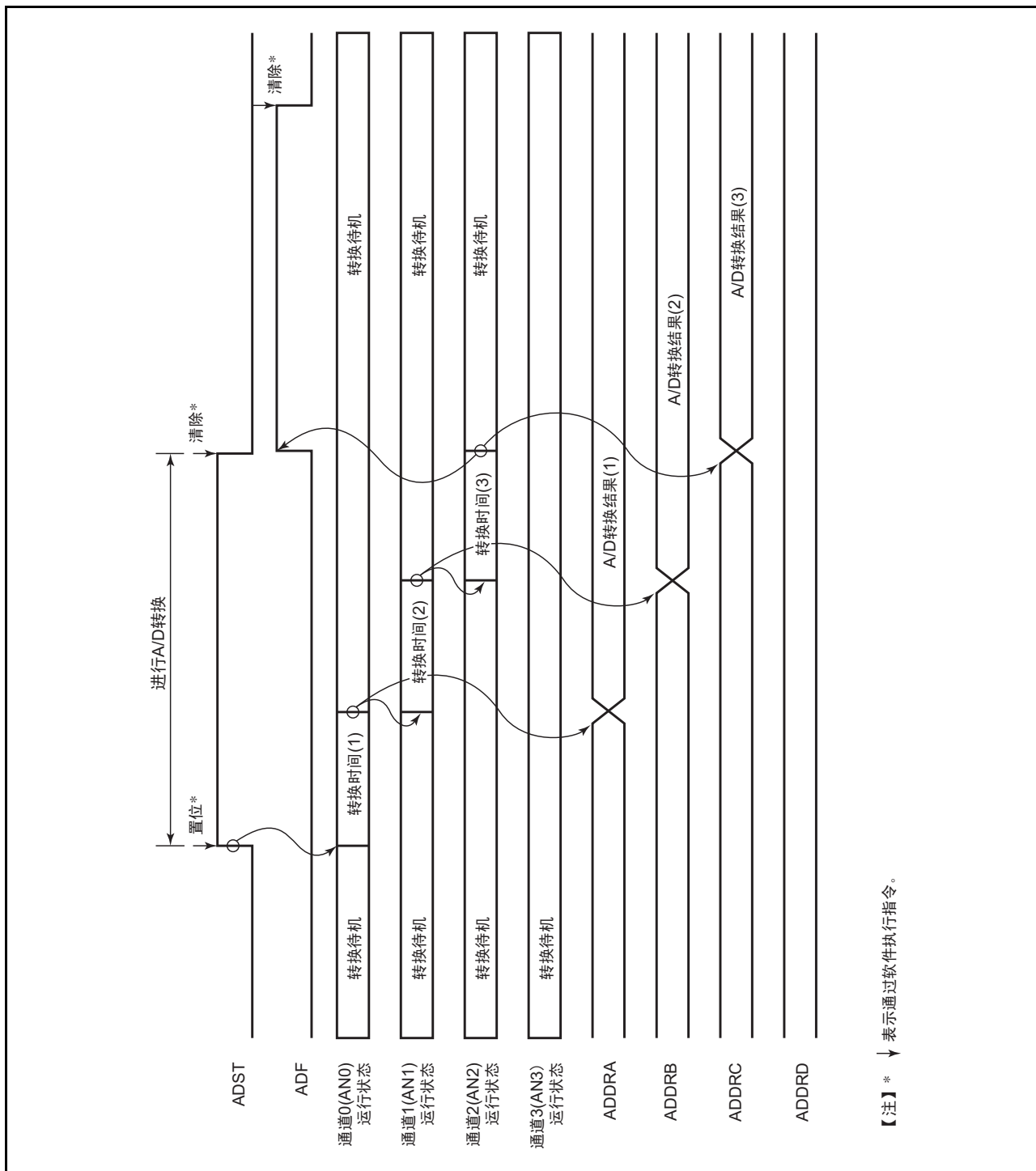


图 27.3 A/D 转换器的运行例子（选择多通道模式和 3 个通道（AN0 ~ AN2）的情况）

27.4.3 扫描模式

扫描模式适用于随时监视多个通道（包含 1 个通道）的模拟输入的系统。在扫描模式中，依次对指定的最多 8 个通道的模拟输入进行连续的 A/D 转换，运行如下：

1. 如果通过软件、多功能定时器脉冲单元 2 或者外部触发输入将 ADCSR 的 ADST 位置 “1”，就按模拟输入通道号从小到大的顺序（例如，AN0、AN1…AN3）进行 A/D 转换。
2. 如果各通道的 A/D 转换结束，就按顺序将 A/D 转换结果传送到该通道对应的 A/D 数据寄存器。
3. 如果所选通道的 A/D 转换全部结束，就将 ADCSR 的 ADF 位置 “1”。此时，如果 ADIE 位已被置 “1”，就产生 ADI 中断请求。A/D 转换器再次按通道号从小到大的顺序进行 A/D 转换。
4. 不自动清除 ADST 位，在 ADST 位为 “1” 的期间重复执行步骤 2.~3.。如果将 ADST 位清 “0”，就中止 A/D 转换并且 A/D 转换器进入待机状态。
通过在读 ADF 位的 “1” 后给 ADF 位写 “0”，清除 ADF 位。

要在 A/D 转换过程中切换模式和模拟输入通道时，为了避免误动作，必须通过将 ADST 位清 “0”，在停止 A/D 转换的状态下进行切换。如果在更改后将 ADST 位置 “1”（能在将 ADST 位置位的同时更改模式或者通道），就选择第 1 个通道，重新开始 A/D 转换。

在扫描模式中，选择 3 个通道（AN0 ~ AN2）进行 A/D 转换的运行例子如下所示，此时的运行时序如图 27.4 所示。

1. 将运行模式设定为扫描模式（MDS[2]位为 “1”，MDS[1]位为 “1”）并且将模拟输入通道设定为 AN0 ~ AN2（CH[2:0]位为 “010”），开始 A/D 转换（ADST 位为 “1”）。
2. 开始第 1 个通道（AN0）的 A/D 转换，如果 A/D 转换结束，就将转换结果传送到 ADDRA。
3. 接着，自动选择第 2 个通道（AN1）开始转换。
4. 同样，转换到第 3 个通道（AN2）为止。
5. 如果所选通道（AN0 ~ AN2）的转换全部结束，ADF 位就变为 “1” 并且重新选择第 1 个通道（AN0），连续进行转换。此时，如果 ADIE 位为 “1”，就在 A/D 转换（第 3 个通道的转换）结束后产生 ADI 中断。
6. 不自动清除 ADST 位，在 ADST 位为 “1” 期间重复执行步骤 2.~4.。在重复期间 ADF 位保持 “1”。如果将 ADST 位清 “0”，就停止 A/D 转换。通过在读 ADF 位的 “1” 后给 ADF 位写 “0”，清除 ADF 位。

在重复步骤 2.~4. 的期间，当 ADF 位和 ADIE 位都为 “1” 时，随时产生 ADI 中断。如果要在第 3 个通道的转换结束时产生中断，就必须在产生中断后将 ADF 位清 “0”。

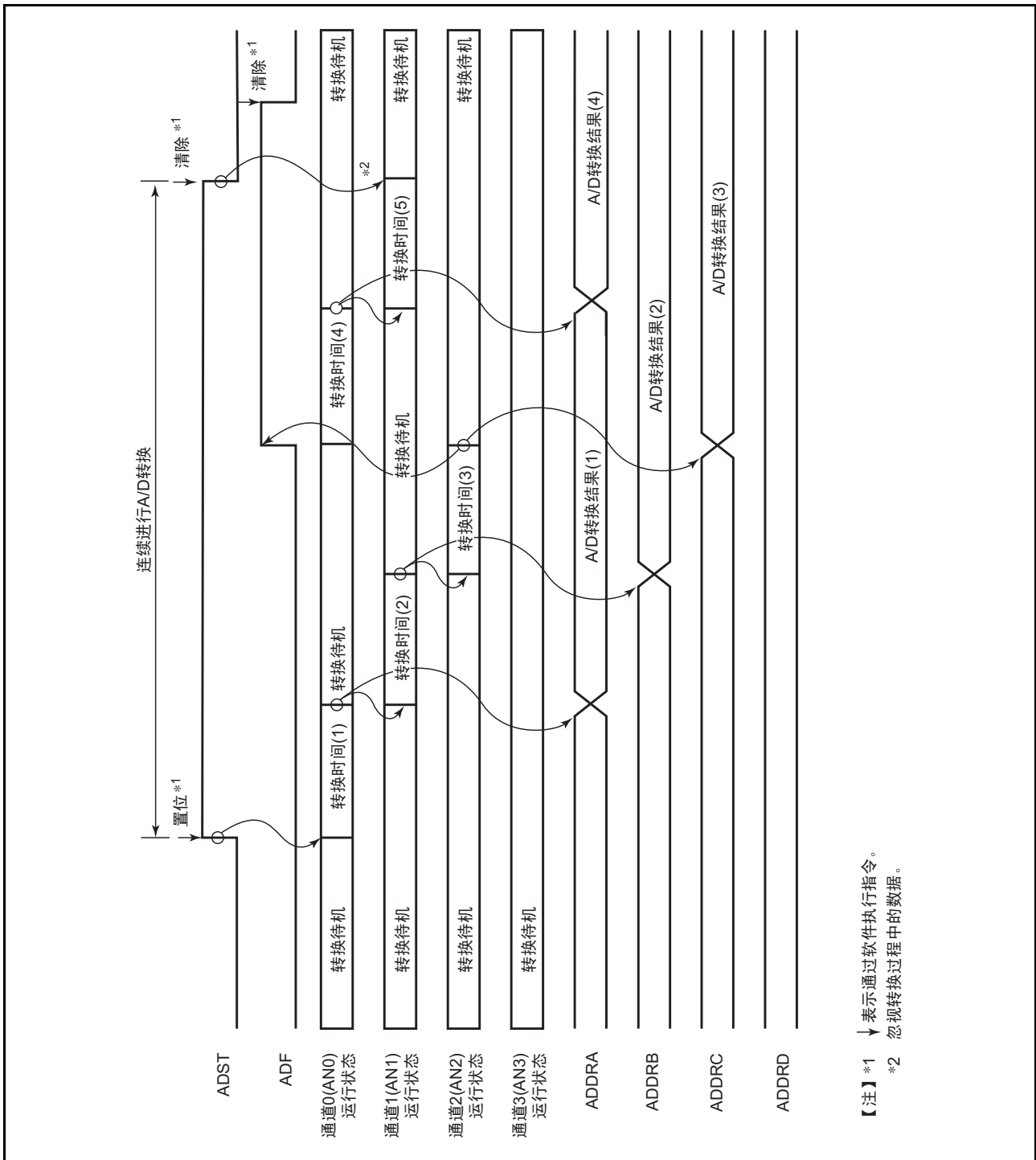


图 27.4 A/D 转换器的运行例子（选择扫描模式和 3 个通道（AN0 ~ AN2）的情况）

27.4.4 通过外部触发或者多功能定时器脉冲单元 2 进行的 A/D 转换器启动

能通过外部触发或者多功能定时器脉冲单元 2 的 A/D 转换请求，独立启动 A/D 转换器。在通过外部触发或者多功能定时器脉冲单元 2 启动 A/D 转换器时，需要设定 A/D 触发允许位（TRGS3 ~ TRGS0）。如果在此状态下发生外部触发或者多功能定时器脉冲单元 2 的 A/D 转换请求，就将 ADST 位置“1”，开始 A/D 转换。要转换的通道取决于 ADCSR 的 CH2 ~ CH0 位。从将 ADST 位置“1”到开始 A/D 转换为止的时序和通过软件给 ADST 位写“1”的情况相同。

27.4.5 输入采样和 A/D 转换时间

A/D 转换器内置采样 & 保持电路。如果在将 ADCSR 的 ADST 位置“1”后经过 A/D 转换开始延迟时间（ t_D ），A/D 转换器就对输入进行采样并且开始转换。A/D 转换时序如图 27.5 所示，A/D 转换时间如表 27.4 所示。

如图 27.5 所示，A/D 转换时间（ t_{CONV} ）包含 t_D 和输入采样时间（ t_{SPL} ）。在此， t_D 由 ADCSR 的写时序决定，而不是固定值。因此，转换时间在表 27.4 所示的范围内发生变化。

有关多通道模式和扫描模式的转换时间，表 27.4 所示的值为第 1 次的转换时间，表 27.5 所示的值为第 2 次及其以后的转换时间。

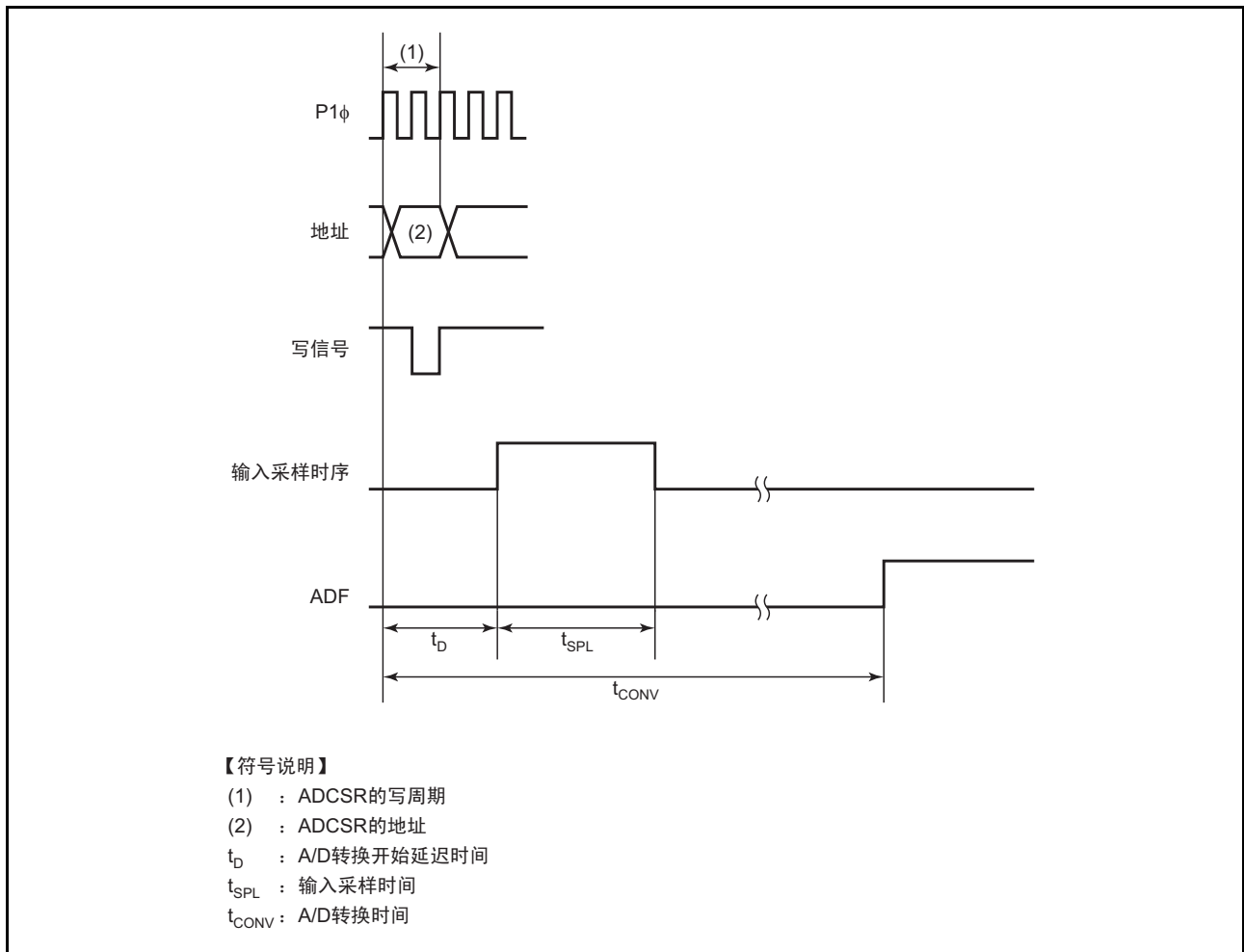


图 27.5 A/D 转换时序

表 27.4 A/D 转换时间（单通道模式）

项目	符号	CKS2=0								
		CKS1=0						CKS1=1		
		CKS0=0			CKS0=1			CKS0=0		
		Min.	Typ.	Max.	Min.	Typ.	Max.	Min.	Typ.	Max.
A/D 转换开始延迟时间	t_D	15	—	26	17	—	30	19	—	34
输入采样时间	t_{SPL}	—	97	—	—	113	—	—	129	—
A/D 转换时间	t_{CONV}	401	—	412	467	—	480	533	—	548

【注】表中数值的单位是 t_{cyc} 。

表 27.5 A/D 转换时间（多通道模式 / 扫描模式）

CKS2	CKS1	CKS0	转换时间 (t_{cyc})
0	0	0	384 (固定)
		1	448 (固定)
	1	0	512 (固定)

【注】表中数值的单位是 t_{cyc} 。

27.4.6 外部触发的输入时序

也能通过外部触发的输入开始 A/D 转换。在 ADCSR 的 TRGS3 ~ TRGS0 位为 “B'1001” 时，从 \overline{ADTRG} 引脚输入外部触发。在 \overline{ADTRG} 的下降沿将 ADCSR 的 ADST 位置 “1”，开始 A/D 转换。其他运行与单通道模式、多通道模式或者扫描模式无关，和通过软件将 ADST 位置 “1” 的情况相同。此时序如图 27.6 所示。

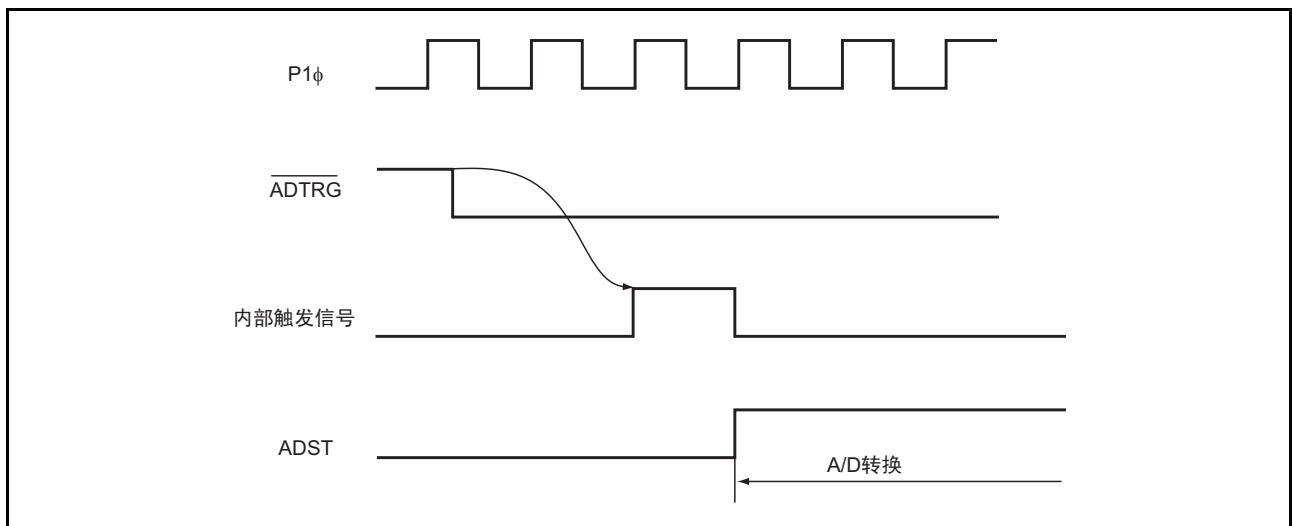


图 27.6 外部触发的输入时序

27.5 中断源和 DMA 传送请求

如果 A/D 转换结束，A/D 转换器就产生 A/D 转换中断（ADI）。在 A/D 转换结束后将 ADCSR 的 ADF 置“1”，如果 ADIE 位已被置“1”，就产生 ADI 中断请求。另外，根据直接存储器存取控制器的设定，ADI 中断能启动直接存储器存取控制器。此时，不向 CPU 请求中断，如果没有进行直接存储器存取控制器的启动设定，就向 CPU 请求中断。如果使用直接存储器存取控制器并且通过 ADI 中断读转换数据，就能不给软件增加负担而进行连续转换。

在单通道模式中，必须设定为通过 ADI 中断进行 1 次 DMA 传送。在通过扫描模式或者多通道模式进行多个通道的 A/D 转换时，如果将 DMA 传送次数设定为 1 次，就在进行 1 个通道的数据传送后结束 DMA 传送。要通过直接存储器存取控制器传送全部转换数据时，必须将传送源地址设定为保存 A/D 转换数据的 ADDR，将传送次数设定为转换通道数。

如果通过 ADI 启动直接存储器存取控制器，就在 DMA 传送时自动清除 ADCSR 的 ADF 位。

表 27.6 中断源和 DMA 传送请求的关系

名称	中断源	中断标志	直接存储器存取控制器的启动
ADI	A/D 转换结束	ADCSR 的 ADF	能

27.6 A/D 转换精度的定义

A/D 转换器将模拟输入通道输入的模拟值一边和模拟基准电压进行比较一边转换为 10 位的数字值。此时的 A/D 转换绝对精度，即输入模拟值和输出数字值的偏差包含以下误差：

1. 偏移误差
2. 满刻度误差
3. 量化误差
4. 非线性误差

按照图 27.7，说明上述 1. ~ 4. 的误差。但是，为了使图容易理解，将 10 位的 A/D 转换器简化为 3 位的 A/D 转换器。偏移误差是指在数字输出值从最小值（零电压）“B'000000000”（图中是“000”）变为“B'000000001”（图中是“001”）时实际 A/D 转换特性与理想 A/D 转换特性的偏差（图 27.7(1)）为。满刻度误差是指在数字输出值从“B'111111110”（图中是“110”）变为最大值（满刻度电压）“B'111111111”（图中是“111”）时实际 A/D 转换特性与理想 A/D 转换特性的偏差（图 27.7(2)）。量化误差是指 A/D 转换器固有的偏差，表示为 $1/2\text{LSB}$ （图 27.7(3)）。非线性误差是指从零电压到满刻度电压之间的实际 A/D 转换特性与理想 A/D 转换特性的偏差（图 27.7(4)），但是不包含偏移误差、满刻度误差和量化误差。

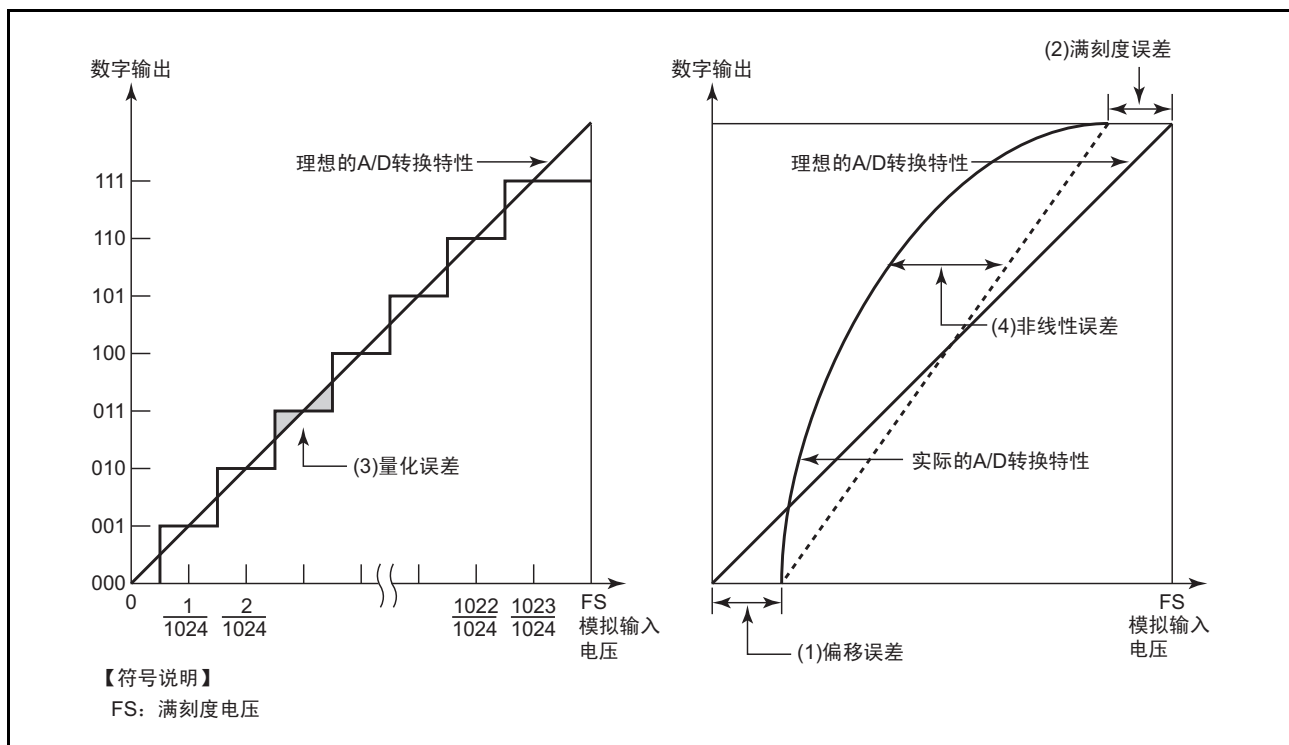


图 27.7 A/D 转换精度的定义

27.7 使用时的注意事项

在使用 A/D 转换器时，必须注意以下事项。

27.7.1 模块待机模式的设定

能通过待机控制寄存器设定允许或者禁止 A/D 转换器的运行，初始值为停止 A/D 转换器的运行。能通过解除模块待机模式，使寄存器变为可存取的状态。详细内容请参照“49. 低功耗模式”。

27.7.2 模拟电压的设定

如果在超过以下电压的设定范围使用 LSI，就会给 LSI 的可靠性带来不良影响。

1. 模拟输入电压的范围

在 A/D 转换过程中，必须将模拟输入引脚 AN_n 的外加电压设定在 $AV_{SS} \leq AN_n \leq AV_{CC}$ 的范围内（n=0~7）。

2. AV_{CC} 和 AV_{SS} 输入电压

AV_{CC} 和 AV_{SS} 的输入电压必须为 $PV_{CC} - 0.3V \leq AV_{CC} \leq PV_{CC}$ ， $AV_{SS} = V_{SS}$ 。而且，在不使用 A/D 转换器时或者在软件待机模式中，不能将 AV_{CC} 引脚和 AV_{SS} 引脚置为开路。不使用时，必须将 AV_{CC} 连接到电源（PV_{CC}）并且将 AV_{SS} 接地（V_{SS}）。

3. AV_{ref} 的设定范围

AV_{ref} 引脚的基准电压范围必须为 $3.0V \leq AV_{ref} \leq AV_{CC}$ 。

27.7.3 电路板设计的注意事项

在设定电路板时，必须尽量将数字电路和模拟电路分开，不能使数字电路的信号线和模拟电路的信号线交叉或者靠近。否则会因电感等引起模拟电路的误动作并且给 A/D 转换值带来不良影响。

必须通过模拟接地（AV_{SS}）将模拟输入信号（AN0~AN3）、模拟基准电压（AV_{ref}）、模拟电源（AV_{CC}）必须通过模拟接地（AV_{SS}）和数字电路分开，并且模拟接地（AV_{SS}）必须和电路板上稳定的数字接地（V_{SS}）进行单点连接。

27.7.4 模拟输入引脚的处理

为了防止过大电涌等异常电压对模拟输入引脚（AN0~AN7）的破坏，必须连接如图 27.8 所示的保护电路。此图的电路还有 CR 过滤器功能，能抑制由噪声引起的误差。图中的电路是设计例子，必须在考虑实际的使用条件的基础上决定电路常数。

模拟输入引脚的等效电路如图 27.9 所示，模拟输入引脚的规格如表 27.7 所示。

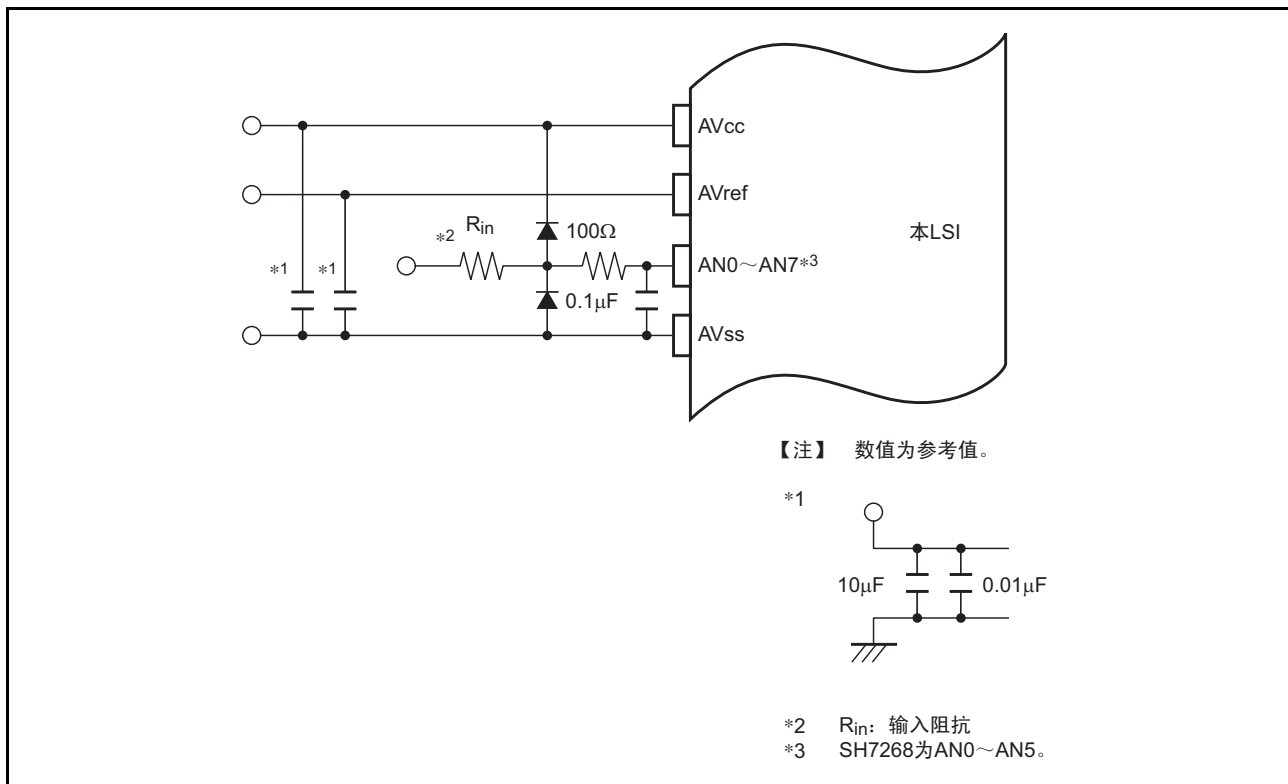


图 27.8 模拟输入引脚的保护电路例子

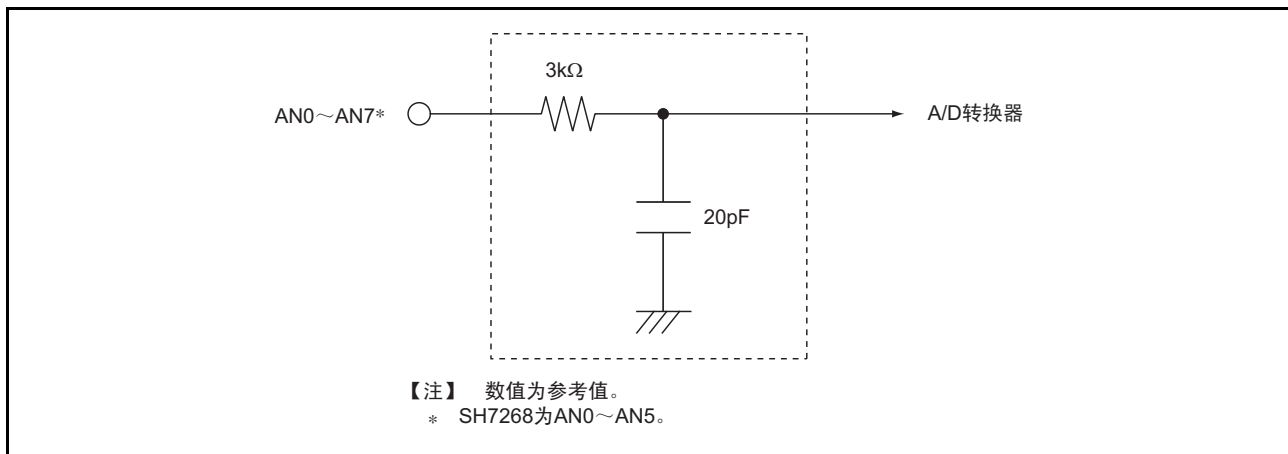


图 27.9 模拟输入引脚的等效电路

表 27.7 模拟输入引脚的规格

项目	Min.	Max.	单位
模拟输入电容	—	20	pF
容许信号源阻抗	—	5	kΩ

27.7.5 容许信号源阻抗

对于信号源阻抗不超过 $5\text{k}\Omega$ 的输入信号，本 LSI 的模拟输入能保证转换精度。这是为了在采样时间内，对 A/D 转换器的采样 & 保持电路的输入电容进行充电而制定的规格。在传感器的输出阻抗超过 $5\text{k}\Omega$ 时，可能发生充电不足并且不能保证 A/D 转换精度的情况。在单通道模式中进行转换并且外接大电容的情况下，因为输入负载实际上只有 $3\text{k}\Omega$ 的内部输入电阻，所以信号源阻抗可忽略不计。但是，由于形成低通滤波器，因此可能无法跟踪大微分系数的模拟信号（例如，大于等于 $5\text{mV}/\mu\text{s}$ ）（图 27.10）。在转换高速模拟信号时或者在扫描模式中进行转换时，必须插入一个低阻抗的缓冲器。

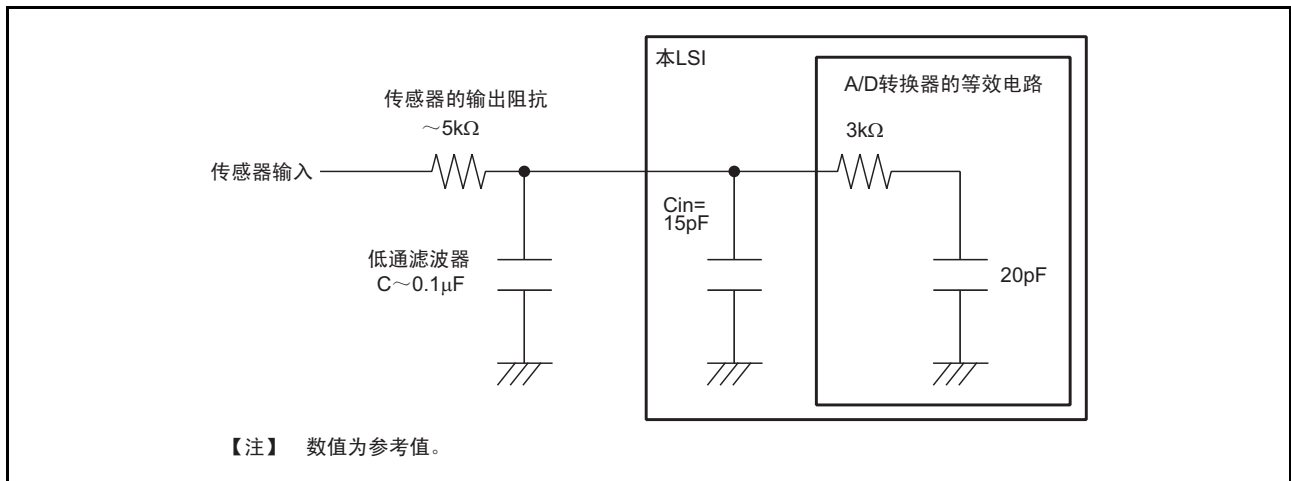


图 27.10 模拟输入电路的例子

27.7.6 对绝对精度的影响

由于附加电容会导致与 GND 的耦合，因此，如果在 GND 中有噪声，就可能降低绝对精度，所以 AV_{SS} 等必须与电稳定的 GND 连接。

另外，必须注意：在安装电路板上滤波器电路不要干扰数字信号也不要充当天线。

27.7.7 深度待机模式时的 A/D 转换

要转移到深度待机模式时，必须将 ADST 位清“0”，禁止 A/D 转换。如果本 LSI 在允许 A/D 转换的状态下变为深度待机模式，就不保证 A/D 的引脚状态。

27.7.8 使用扫描模式或者多通道模式时的注意事项

如果在扫描模式和多通道模式停止后立即开始转换，就可能产生错误的转换结果。

要连续进行转换时，必须在将 ADST 位置“0”后经过 1 个通道的 A/D 转换时间（1 个通道的转换时间因分频寄存器的设定而不同），然后开始转换（ADST 位为“1”）。

28. NAND 闪存控制器

NAND 闪存控制器提供和外接 NAND 型闪存的存储器接口。

28.1 特点

(1) NAND 型闪存的存储器接口

- 能直接连接NAND型闪存的接口
- 以扇区为单位（512+16字节）进行读写。
- 以字节为单位进行读写。
- 支持大容量块（2048+64）的闪存*。
- 通过扩展为5字节地址，支持超过2G位的地址。

【注】* 此模块将512+16字节作为1个扇区处理。对于1页2048+64字节的产品，按512+16字节（1页4个扇区）进行分割处理。

(2) 存取模式：此模块能选择以下2种存取模式：

- 命令存取模式：通过给寄存器指定此模块对闪存发行的命令、地址、输入/输出的数据长度，进行一系列的存取。
- 扇区存取模式：通过指定扇区地址，以扇区为单位进行读写。通过指定扇区数，能对连续的扇区进行读写。

(3) 扇区和管理码

- 扇区是存取的基本单位，由512字节的数据和16字节的管理码构成。
- 能将用户信息写到管理码。

(4) 发生数据错误时

- 在发生编程/擦除错误时，反映到错误源标志。能指定独立源的中断。

(5) 用于数据传送的FIFO和数据寄存器

- 内置224字节的数据FIFO寄存器（FLDTFIFO），用于闪存的数据传送。
- 内置32字节的管理码FIFO寄存器（FLECFIFO），用于管理码的数据传送。

(6) DMA 传送

- 能通过给直接存储器存取控制器分别指定闪存数据和管理码的传送目标，将数据和管理码传送到不同的区域。

(7) 存取时间

- NAND型闪存引脚的运行时钟（FCLK）使用将外围时钟0（P0φ）分频后的时钟。能通过公共控制寄存器（FLCMNCR）的QTSEL位指定分频率。
- 要更改CPG的设定时，必须通过模块停止模式使此模块处于停止状态，然后进行更改。
- 对于NAND型闪存，FRE引脚和FWE引脚以FCLK的工作频率运行。在设定时，不能超过连接存储器的最大工作频率。

框图如图 28.1 所示。

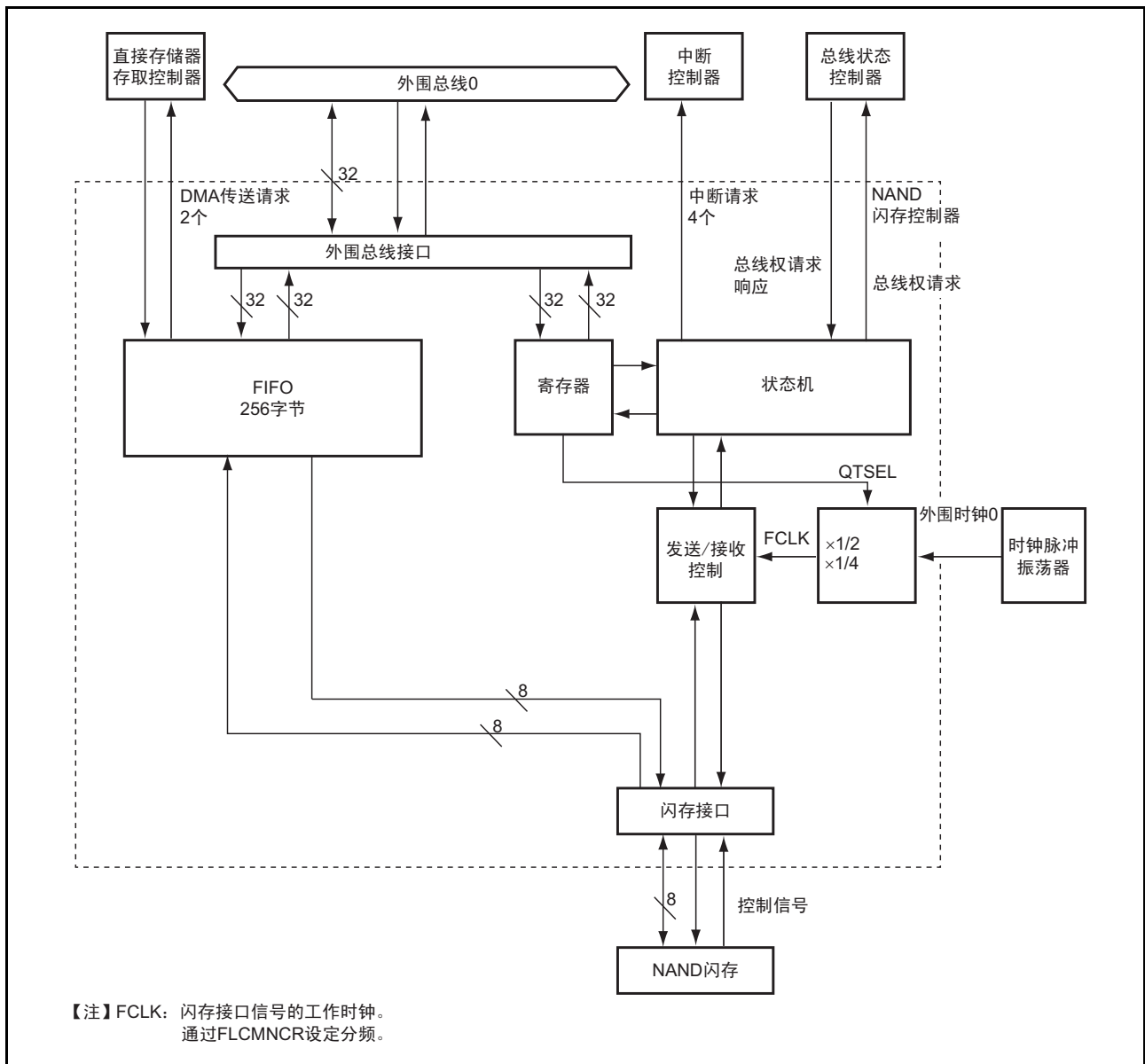


图 28.1 框图

28.2 输入 / 输出引脚

引脚结构如表 28.1 所示。

表 28.1 引脚结构

引脚名	输入 / 输出	对应的闪存引脚	功能
		NAND 型	
$\overline{\text{FCE}}$	输出	$\overline{\text{CE}}$	闪存芯片允许 允许连接本 LSI 的闪存。
NAF7 ~ 0	输入 / 输出	I/O7 ~ I/O0	闪存数据 这是命令、地址、数据的输入 / 输出引脚。
FCLE	输出	CLE	闪存命令锁定允许 在输出命令时为有效电平。
FALE	输出	ALE	闪存地址锁定允许 在输出地址时为有效电平。 在输出数据时为无效电平。
$\overline{\text{FRE}}$	输出	$\overline{\text{RE}}$	闪存读允许 在 $\overline{\text{RE}}$ 的下降沿读数据。
$\overline{\text{FWE}}$	输出	$\overline{\text{WE}}$	闪存写允许 在 $\overline{\text{WE}}$ 的上升沿，闪存锁定命令、地址和数据。
FRB	输入	$\text{R}/\overline{\text{B}}$	闪存就绪 / 忙 高电平表示就绪状态，低电平表示忙状态。
—*	—	$\overline{\text{WP}}$	编程保护 / 复位 在低电平的情况下，保护在接通或者切断电源时的偶发性擦除 / 编程。
—*	—	$\overline{\text{SE}}$	备用区允许 这是能存取备用区的引脚。当使用扇区存取模式时，必须固定为低电平。

【注】* 本 LSI 不支持此引脚。

28.3 寄存器说明

寄存器结构如表 28.2 所示。

表 28.2 寄存器结构

寄存器名	略称	R/W	初始值	地址	存取长度
公共控制寄存器	FLCMNCR	R/W	H'00100001	H'FFFF4000	32
命令控制寄存器	FLCMDCR	R/W	H'00000000	H'FFFF4004	32
命令码寄存器	FLCMCDR	R/W	H'00000000	H'FFFF4008	32
地址寄存器	FLADR	R/W	H'00000000	H'FFFF400C	32
地址寄存器 2	FLADR2	R/W	H'00000000	H'FFFF403C	32
数据寄存器	FLDATAR	R/W	H'00000000	H'FFFF4010	32
数据计数寄存器	FLDTCNTR	R/W	H'00000000	H'FFFF4014	32
中断 DMA 控制寄存器	FLINTDMACR	R/W	H'00000000	H'FFFF4018	32
就绪 / 忙超时设定寄存器	FLBSYTMR	R/W	H'00000000	H'FFFF401C	32
就绪 / 忙超时计数器	FLBSYCNT	R	H'00000000	H'FFFF4020	32
数据 FIFO 寄存器	FLDTFIFO	R/W	H'xxxxxxxx	H'FFFF4050	32
管理码 FIFO 寄存器	FLECFIFO	R/W	H'xxxxxxxx	H'FFFF4060	32
传送控制寄存器	FLTRCR	R/W	H'00	H'FFFF402C	8
总线占有期间设定寄存器	FLHOLDCR	R/W	H'00000000	H'FFFF4038	32

28.3.1 公共控制寄存器 (FLCMNCR)

FLCMNCR 是 32 位可读写寄存器，指定存取模式等。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	BUSYON	—	—	SNAND	QTSEL	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	ACM[1:0]	NANDWF	—	—	—	—	—	—	CE	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R	R	R

位	位名	初始值	R/W	说明
31 ~ 22	—	全 0	R	保留位 读写值都为“0”。
21	BUSYON	0	R/W	忙选择位 指定是否在 FRB 引脚处于忙的状态中释放外部总线权。但是，如果一系列处理结束， \overline{FCE} 引脚就变为无效电平，与忙 / 就绪的状态无关。详细内容请参照“28.7.1 外部总线权释放的时序”。 0: 在 FRB 引脚处于忙的状态中占有总线权 1: 在 FRB 引脚处于忙的状态中释放总线权 【注】 必须注意：根据闪存，有些产品 \overline{FCE} 引脚在忙状态期间不能变为无效电平。
20	—	1	R	保留位 读写值都为“1”。
19	—	0	R	保留位 读写值都为“0”。
18	SNAND	0	R/W	大容量 NAND 闪存选择位 用于 1G 位以上闪存中的 1 页结构为 2048+64 字节的 NAND 闪存。 0: 在使用 1 页结构为 512+16 字节的闪存时，必须置“0”。 1: 在使用 1 页结构为 2048+64 字节的 NAND 型闪存时，必须置“1”。
17	QTSEL	0	R/W	闪存时钟分频选择位 此位选择闪存内使用的时钟 FCLK 的分频。 0: 将 CPG 提供的时钟 (P0 ϕ) 进行 1/2 分频，用作 FCLK。 1: 将 CPG 提供的时钟 (P0 ϕ) 进行 1/4 分频，用作 FCLK。
16 ~ 12	—	全 0	R	保留位 读写值都为“0”。
11、10	ACM[1:0]	00	R/W	存取模式指定位 1、0 指定存取模式。 00: 命令存取模式 01: 扇区存取模式 10: 禁止设定 11: 禁止设定
9	NANDWF	0	R/W	NAND 等待插入运行位 0: 通过 1 个 FCLK 周期进行地址和数据的输入 / 输出 1: 通过 2 个 FCLK 周期进行地址和数据的输入 / 输出

位	位名	初始值	R/W	说明
8 ~ 4	—	全 0	R	保留位 读写值都为“0”。
3	CE	0	R/W	芯片允许位 0: 禁止 ($\overline{\text{FCE}}$ 引脚输出高电平。) 1: 允许 ($\overline{\text{FCE}}$ 引脚输出低电平。)
2、1	—	全 0	R	保留位 读写值都为“0”。
0	—	1	R	保留位 读写值都为“1”。

28.3.2 命令控制寄存器 (FLCMDCR)

FLCMDCR 是 32 位可读写寄存器，能指定是否在命令存取模式中发行命令和地址以及指定数据的输入 / 输出目标。在扇区存取模式中，能指定扇区的传送次数。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	ADR CNT2	SCTCNT[19:16]					ADR MD	CDS RC	DOSR	—	—	SEL RW	DOA DR	ADRCNT[1:0]		DOC MD2	DOC MD1
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCTCNT[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31	ADRCNT2	0	R/W	地址发行字节数指定位 2 指定在地址阶段发行的地址数据的字节数。和 ADRCNT[1:0] 位一起使用。 0: 只发行 ADRCNT[1:0] 位指定的字节数地址。 1: 发行 5 字节的地址 必须将 ADRCNT[1:0] 位置“00”。
30 ~ 27	SCTCNT [19:16]	0000	R/W	扇区传送次数指定位 [19:16] 这是扇区传送次数指定位 SCTCNT[15:0] 的扩展位。 SCTCNT[19:16]和SCTCNT[15:0]一起用作SCTCNT[19:0]的20位计数器。
26	ADRMD	0	R/W	扇区存取地址指定位 此位在命令存取模式中无效，只在扇区存取模式中有效。 0: 将地址寄存器的值作为扇区地址进行处理。在存取扇区时，通常必须使用此值。 1: 将地址寄存器的值作为闪存地址进行输出。 【注】 在存取连续的扇区时，必须置“0”。
25	CDSRC	0	R/W	数据缓冲器指定位 在命令存取模式中，指定在数据阶段要读写的数据缓冲器。 0: 指定 FLDATAR 作为数据缓冲器 1: 指定 FLDTFIFO 作为数据缓冲器
24	DOSR	0	R/W	状态读选定位 在命令存取模式中，指定是否在发行第 2 个命令后读状态。 0: 不读状态 1: 读状态

位	位名	初始值	R/W	说明
23、22	—	全 0	R	保留位 读写值都为“0”。
21	SELRW	0	R/W	数据读写指定位 指定在数据阶段的读写方向。 0: 读 1: 写
20	DOADR	0	R/W	地址阶段执行指定位 在命令存取模式中, 指定是否执行地址阶段。 0: 不执行地址阶段 1: 执行地址阶段
19、18	ADRCNT [1:0]	00	R/W	地址发行字节数指定位 1、0 指定在地址阶段发行的地址数据的字节数。 00: 发行 1 字节的地址 01: 发行 2 字节的地址 10: 发行 3 字节的地址 11: 发行 4 字节的地址
17	DOCMD2	0	R/W	第 2 个命令阶段执行指定位 在命令存取模式中, 指定是否执行第 2 个命令阶段。 0: 不执行第 2 个命令阶段 1: 执行第 2 个命令阶段
16	DOCMD1	0	R/W	第 1 个命令阶段执行指定位 在命令存取模式中, 指定是否执行第 1 个命令阶段。 0: 不执行第 1 个命令阶段 1: 执行第 1 个命令阶段
15 ~ 0	SCTCNT [15:0]	H'0000	R/W	扇区传送次数指定位 [15:0] 指定在扇区存取模式中连续读的扇区数。每当 1 扇区传送结束时, 进行递减计数, 计数到 0 时停止。 和 SCTCNT[19:16] 一起使用。 在命令存取模式的运行过程中, 为“H'00001”。

28.3.3 命令码寄存器 (FLCMCDR)

FLCMCDR 是 32 位可读写寄存器，指定在存取命令或者扇区时要发行的命令值。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMD2[7:0]							CMD1[7:0]								
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 16	—	全 0	R	保留位 读写值都为“0”。
15 ~ 8	CMD2[7:0]	H'00	R/W	第 2 个命令数据位 指定在第 2 个命令阶段发行的命令码。
7 ~ 0	CMD1[7:0]	H'00	R/W	第 1 个命令数据位 指定在第 1 个命令阶段发行的命令码。

28.3.4 地址寄存器 (FLADR)

FLADR 是 32 位可读写寄存器，指定要输出的地址值。从 ADR1 开始按顺序并且以字节为单位，将命令寄存器指定的字节数作为地址进行输出。另外，能通过命令控制寄存器的扇区存取地址指定位 (ADRMD)，指定是否将地址数据位指定的扇区号转换为地址值进行输出。

- ADRMD 位为“1”的情况

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADR4[7:0]							ADR3[7:0]								
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADR2[7:0]							ADR1[7:0]								
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	ADR4[7:0]	H'00	R/W	第 4 个地址数据位 当 ADRMD 位为“1”时，指定输出到闪存的第 4 个数据为地址。
23 ~ 16	ADR3[7:0]	H'00	R/W	第 3 地址数据位 当 ADRMD 位为“1”时，指定输出到闪存的第 3 个数据为地址。
15 ~ 8	ADR2[7:0]	H'00	R/W	第 2 地址数据位 当 ADRMD 位为“1”时，指定输出到闪存的第 2 个数据为地址。
7 ~ 0	ADR1[7:0]	H'00	R/W	第 1 地址数据位 当 ADRMD 位为“1”时，指定输出到闪存的第 1 个数据为地址。

• ADRMD位为“0”的情况

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	ADR[25:16]									
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADR[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 26	—	全 0	R	保留位 读写值都为“0”。
25 ~ 0	ADR[25:0]	H'0000000	R/W	扇区地址指定位 当 ADRMD 位为“0”时，指定要存储的扇区号。将扇区号转换为地址并且将转换后的地址输出到闪存。当 FLCMDCR 的 ADCNT2 位为“1”时，ADR[25:0] 有效；当 ADCNT2 位为“0”时，ADR[17:0] 有效。详细内容请参照图 28.11。 <ul style="list-style-type: none"> 大容量块产品（2048+64 字节） 能通过 ADR[25:2] 指定页地址并且通过 ADR[1:0] 指定以扇区为单位的列地址。 ADR[1:0]=00：第 0 个字节（0 个扇区） ADR[1:0]=01：第 512+16 个字节（1 个扇区） ADR[1:0]=10：第 1024+32 个字节（2 个扇区） ADR[1:0]=11：第 1536+48 个字节（3 个扇区） 小容量块产品（512+16 字节） 只能指定页地址。

28.3.5 地址寄存器 2（FLADR2）

FLADR2 是 32 位可读写寄存器，在 FLCMDCR 的 ADCNT2 位为“1”时有效。在命令存取模式中，指定要输出的地址值。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ADR5[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 8	—	全 0	R	保留位 读写值都为“0”。
7 ~ 0	ADR5[7:0]	H'00	R/W	第 5 个地址数据位 当 ADRMD 位为“1”时，指定输出到闪存的第 5 个数据为地址。

28.3.6 数据计数寄存器 (FLDTCNTR)

FLDTCNTR 是 32 位可读写寄存器。在命令存取模式中，指定要读写的字节数。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECFLW[7:0]								DTFLW[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DTCNT[11:0]											
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	ECFLW[7:0]	H'00	R	FLECFIFO 存取个数位 表示 FLECFIFO 能读写的长字个数。 能在通过 CPU 读写 FLECFIFO 时使用。 在读 FLECFIFO 时，表示在 FLECFIFO 内能读的数据长字个数。 在写 FLECFIFO 时，表示在 FLECFIFO 内能写的空长字个数。
23 ~ 16	DTFLW[7:0]	H'00	R	FLDTFIFO 存取个数位 表示 FLDTFIFO 能读写的长字个数。 能用于通过 CPU 进行的 FLDTFIFO 读写。 在读 FLDTFIFO 时，表示 FLDTFIFO 内能读的数据长字个数。 在写 FLDTFIFO 时，表示 FLDTFIFO 内能写的空长字个数。
15 ~ 12	—	全 0	R	保留位 读写值都为“0”。
11 ~ 0	DTCNT[11:0]	H'000	R/W	数据量指定位 指定命令存取模式中的读写数据的字节数（最多能指定 2048+64 字节）。

28.3.7 数据寄存器 (FLDATAR)

FLDATAR 是 32 位可读写寄存器。在命令存取模式中，用于保存 FLCMDCR 的 CDSRC 位为“0”时使用的输入/输出数据。在读写大于等于 5 字节的连续数据时，不能使用此寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DT4[7:0]								DT3[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DT2[7:0]								DT1[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	DT4[7:0]	H'00	R/W	第 4 个数据位 指定从 NAF7 ~ 0 输入或者输出的第 4 个数据。 写时：指定写数据。 读时：保存读数据。
23 ~ 16	DT3[7:0]	H'00	R/W	第 3 个数据位 指定从 NAF7 ~ 0 输入或者输出的第 3 个数据。 写时：指定写数据。 读时：保存读数据。
15 ~ 8	DT2[7:0]	H'00	R/W	第 2 个数据位 指定从 NAF7 ~ 0 输入或者输出的第 2 个数据。 写时：指定写数据。 读时：保存读数据。
7 ~ 0	DT1[7:0]	H'00	R/W	第 1 个数据位 指定从 NAF7 ~ 0 输入或者输出的第 1 个数据。 写时：指定写数据。 读时：保存读数据。

28.3.8 中断 DMA 控制寄存器 (FLINTDMACR)

FLINTDMACR 是 32 位可读写寄存器，设定 DMA 传送请求以及允许或者禁止中断。在各存取模式开始运行后，此模块对直接存储器控制器产生传送请求。

bit9 ~ 5 是表示是否在存取闪存时发生各种错误和 FIFO 传送请求的标志位，只能给这些标志写“0”。要清除标志时，必须给要清除的标志位写“0”而给其他标志位写“1”。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	FIFOTRG [1:0]	AC1 CLR	AC0 CLR	DREQ1 EN	DREQ0 EN	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ST ERB	BTO ERB	TRR EQF1	TRR EQF0	STER INTE	RBER INTE	TE INTE	TR INTE1	TR INTE0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/W	R/W	R/W	R/W	R/W

【注】 * 只能写“0”。

位	位名	初始值	R/W	说明
31 ~ 22	—	全 0	R	保留位 读写值都为“0”。
21、20	FIFOTRG [1:0]	00	R/W	FIFO 触发设定位 设定 FLDTFIFO 和 FLECFIFO 的传送请求发生条件（字节数）。 <ul style="list-style-type: none"> 读闪存的情况 当 FLDTFIFO（FLECFIFO）的保存字节数大于等于以下值时，就向 CPU 请求中断或者产生 DMA 传送请求。 00: 4（4） 01: 16（16） 10: 128（4） 11: 128（16） 闪存编程的情况 当 FLDTFIFO（FLECFIFO）的空字节数大于等于以下值时，就向 CPU 请求中断或者产生 DMA 传送请求。 00: 4（4） 01: 16（16） 10: 128（4） 11: 128（16） 【注】 * 在进行 FLDTFIFO 的 DMA 传送时，禁止设定“10”和“11”。
19	AC1CLR	0	R/W	FLECFIFO 清除位 清除 FLECFIFO。 0: 保持 FLECFIFO 的值。在存取闪存时，必须置“0”。 1: 清除 FLECFIFO。必须在清除后置“0”。
18	AC0CLR	0	R/W	FLDTFIFO 清除位 清除数据区 FLDTFIFO。 0: 保持 FLDTFIFO 的值。在存取闪存时，必须置“0”。 1: 清除 FLDTFIFO 的值。必须在清除后置“0”。
17	DREQ1EN	0	R/W	FLECFIFODMA 请求允许位 选择允许或者禁止从 FLECFIFO 区发行 DMA 传送请求。 0: 禁止从 FLECFIFO 区发行 DMA 传送请求 1: 允许从 FLECFIFO 区发行 DMA 传送请求

位	位名	初始值	R/W	说明
16	DREQ0EN	0	R/W	FLDTFIFODMA 请求允许位 选择允许或者禁止从 FLDTFIFO 区发行 DMA 传送请求。 0: 禁止从 FLDTFIFO 区发行 DMA 传送请求 1: 允许从 FLDTFIFO 区发行 DMA 传送请求
15 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	STERB	0	R/(W)*	状态错误位 表示读状态的结果。如果在读状态后 FLBSYCNT 的 STAT[7:0] 位的特定定位为“1”，就将此位置“1”。 此位为标志位，因此不能写“1”。为了清除标志，只能写“0”。 0: 没有发生状态错误（表示 FLBSYCNT 的 STAT[7:0] 位的特定定位为“0”） 1: 发生状态错误 有关特定位置，请参照“28.4.6 状态的读操作”
7	BTOERB	0	R/(W)*	R/B 超时错误位 当发生 R/B 超时错误时（FLBSYCNT 的 RBTIMCNT[19:0] 位在递减计数后变为“0”），将此位置“1”。此位为标志位，因此不能写“1”。为了清除标志，只能写“0”。 0: 没有发生 R/B 超时错误 1: 发生 R/B 超时错误。
6	TRREQF1	0	R/(W)*	FLECFIFO 传送请求标志位 表示 FLECFIFO 发生传送请求。 此位为标志位，因此不能写“1”。为了清除标志，只能写“0”。 0: FLECFIFO 没有发生传送请求 1: FLECFIFO 发生传送请求
5	TRREQF0	0	R/(W)*	FLDTFIFO 传送请求标志位 表示 FLDTFIFO 发生传送请求。 此位为标志位，因此不能写“1”。为了清除标志，只能写“0”。 0: FLDTFIFO 没有发生传送请求 1: FLDTFIFO 发生传送请求
4	STERINTE	0	R/W	发生状态错误时的中断允许位 在发生状态错误时，选择允许或者禁止向 CPU 请求中断。 0: 禁止在发生状态错误时向 CPU 请求中断 1: 允许在发生状态错误时向 CPU 请求中断
3	RBERINTE	0	R/W	发生 R/B 超时错误时的中断允许位 在发生 R/B 超时错误时，选择允许或者禁止向 CPU 请求中断。 0: 禁止在发生 R/B 超时错误时向 CPU 请求中断 1: 允许在发生 R/B 超时错误时向 CPU 请求中断
2	TEINTE	0	R/W	传送结束中断允许位 在传送结束（FLTRCR 的 TREND 位）时，选择允许或者禁止向 CPU 请求中断。 0: 禁止在传送结束时向 CPU 请求中断 1: 允许在传送结束时向 CPU 请求中断
1	TRINTE1	0	R/W	向 CPU 请求 FLECFIFO 传送的允许位 在发生 FLECFIFO 传送请求时，选择允许或者禁止向 CPU 请求中断。 0: 禁止在发生 FLECFIFO 传送请求时向 CPU 请求中断 1: 允许在发生 FLECFIFO 传送请求时向 CPU 请求中断 当允许 DMA 传送时，必须将此位置“0”。

位	位名	初始值	R/W	说明
0	TRINTE0	0	R/W	向 CPU 请求 FLDTFIFO 传送的允许位 在发生 FLDTFIFO 传送请求时，选择允许或者禁止向 CPU 请求中断。 0: 禁止在发生 FLDTFIFO 传送请求时向 CPU 请求中断 1: 允许在发生 FLDTFIFO 传送请求时向 CPU 请求中断 当允许 DMA 传送时，必须将此位置“0”。

【注】* 只能写“0”。

28.3.9 就绪 / 忙超时设定寄存器 (FLBSYTMR)

FLBSYTMR 是 32 位可读写寄存器，设定 FRB 引脚处于忙状态时的超时时间。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	RBTMOUT[19:16]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RBTMOUT[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 20	—	全 0	R	保留位 读写值都为“0”。
19 ~ 0	RBTMOUT [19:0]	H'00000	R/W	就绪 / 忙超时位 设定忙状态的超时时间 (Pφ 的时钟数)。如果置“0”，就不发生超时。

28.3.10 就绪 / 忙超时计数器 (FLBSYCNT)

FLBSYCNT 是 32 位只读寄存器。

通过读状态，将读到的闪存状态保存到 STAT[7:0]。

如果 FRB 引脚处于忙状态，就将 FLBSYTMR 的 RBTMOUT[19:0] 位设定的超时时间复制到 RBTIMCNT[19:0] 位，并且开始递减计数。当 RBTIMCNT[19:0] 位的值为“0”时，将 FLINTDMACR 的 BTOERB 位置“1”，并且通知发生超时错误。此时，如果通过 FLINTDMACR 的 RBERINTE 位允许中断，就能产生 FLSTE 中断。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16		
	STAT[7:0]								—	—	—	—	RBTIMCNT[19:16]					
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R		
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
	RBTIMCNT[15:0]																	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R		

位	位名	初始值	R/W	说明
31 ~ 24	STAT[7:0]	H'00	R	这些位表示闪存读到的状态值。
23 ~ 20	—	全 0	R	保留位 读取值都为“0”。
19 ~ 0	RBTIMCNT [19:0]	H'00000	R	就绪 / 忙超时计数器位 当 FRB 引脚处于忙状态时，将 FLBSYTMR 的 RBTMOUT[19:0] 位的设定值复制到这些位。 然后，在 FRB 引脚为忙状态的期间，对这些位的值进行递减计数，如果计数值变为“0”，就发生超时错误。

28.3.11 数据 FIFO 寄存器 (FLDTFIFO)

FLDTFIFO 是用于读写数据 FIFO 区的寄存器。

在 DMA 传送时，必须将此寄存器指定为数据传送目标（传送源）。

FLCMDCR 的 SELRW 位指定的读写方向必须和此寄存器的读写存取方向相同。要改变读写方向时，必须在通过 FLINTDMACR 的 AC0CLR 位清除 FLDTFIFO 后使用。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTFO[31:16]															
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTFO[15:0]															
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 0	DTFO[31:0]	H'xxxxxxxx	R/W	这是用于读写数据 FIFO 区的寄存器。 写时：将数据写到数据 FIFO 区。 读时：从数据 FIFO 区读数据。

28.3.12 管理码 FIFO 寄存器 (FLECFIFO)

FLECFIFO 是用于读写管理码 FIFO 区的寄存器。

在 DMA 传送时，必须将此寄存器指定为管理码传送目标（传送源）。

FLCMDCR 的 SELRW 位指定的读写方向必须和此寄存器的读写存取方向相同。要改变读写方向时，必须在通过 FLINTDMACR 的 AC1CLR 位清除 FLECFIFO 后使用。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECFO[31:16]															
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECFO[15:0]															
初始值:	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值	不定值
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 0	ECFO[31:0]	H'xxxxxxxx	R/W	这是用于读写管理码 FIFO 区的寄存器。 写时：将数据写到管理码 FIFO 区。 读时：从管理码 FIFO 区读数据。

28.3.13 传送控制寄存器 (FLTRCR)

通过将 TRSTRT 位置“1”，开始存取闪存。能通过 TREND 位确认存取的结束。在传送过程中（从将 TRSTRT 位置“1”到 TREND 位被置“1”的期间）不能强制结束（将 TRSTRT 位置“0”）。当读闪存时，在读完闪存时将 TREND 位置位。如果在 FIFO 中留有读数据，就不能在读完 FIFO 前强制结束。另外，在此模块获得外部总线的总线权进行传送的期间，不能执行 SLEEP 指令，直到 TREND 位被置位并且传送结束为止。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	TR STAT	TR END	TR STRT
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 3	—	全 0	R	保留位 读写值都为“0”。
2	TRSTAT	0	R	传送状态位 表示此模块获得外部总线的总线权并且正在进行实际的传送。 0: 没有开始传送 1: 正在传送或者传送结束
1	TREND	0	R/W	处理结束标志位 表示指定存取模式的处理已经结束。 只能写“0”。
0	TRSTRT	0	R/W	传送开始位 在 TREND 为“0”时，通过将 TRSTR 从“0”变为“1”，开始存取模式指定位 ACM[1:0] 指定的存取模式的处理。 0: 停止传送 1: 开始传送

28.3.14 总线占有期间设定寄存器 (FLHOLDCR)

在通过扇区存取模式对闪存进行存取的区间，当其他模块（包含 CPU）存取 BSC 控制的存储器时，此寄存器设定此模块的外部总线权的释放频率。当此寄存器的 HOLDEN 位为“0”时，此模块在和闪存传送过程中占有外部总线。必须注意：可能会因程序码、传送数据的保存位置和使用方法而引起死锁。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	HOLDEN
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	HOLDEN	0	R/W	总线占有允许位 在通过扇区存取模式对闪存进行存取的过程中，设定外部总线权的释放。 0: 在传送期间不释放总线权 1: 在扇区存取模式中转移到 FIFO 空 / 满状态时，释放总线权。 【注】 要在扇区存取模式中使用 FIFO 时，必须将此模块控制程序和传送数据保存到内部 RAM。

28.4 运行说明

28.4.1 存取步骤

此模块将一系列的存取步骤分为独立的阶段。

例如，在对 NAND 型闪存进行编程时，有以下 5 个阶段：

- 第 1 个命令发行阶段（编程设置命令）
- 地址发行阶段（编程地址）
- 数据阶段（输出）
- 第 2 个命令发行阶段（编程开始命令）
- 读状态

通过按顺序执行此 5 个阶段来实现一系列的存取，在最后的阶段（此时为读状态）结束时，结束对闪存的存取。

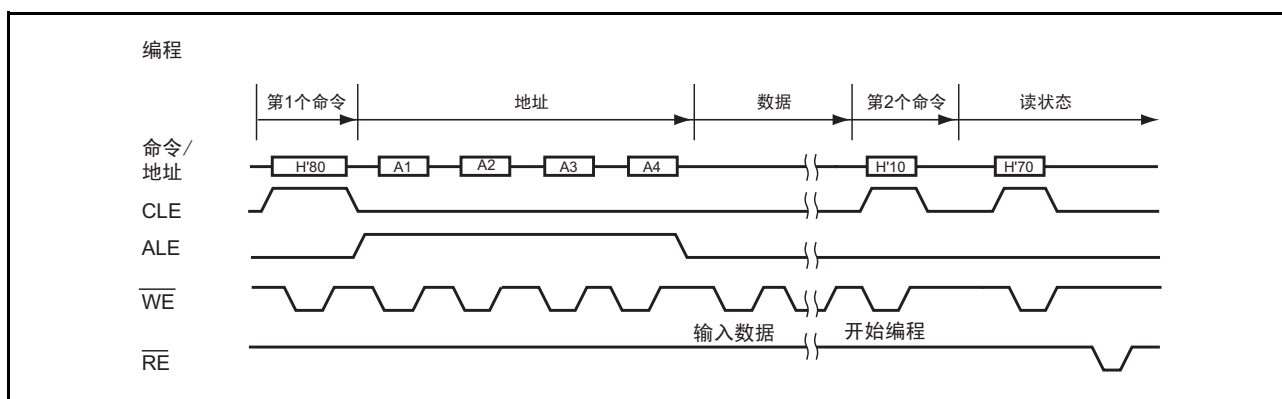


图 28.2 NAND 型闪存编程操作的阶段区分例子

有关详细内容和 NAND 型闪存的读操作，请参照“28.4.4 命令存取模式”。

28.4.2 运行模式

运行模式有以下 2 种模式：

- 命令存取模式
- 扇区存取模式

在扇区存取模式中，进行 ECC 的生成和错误校验。

28.4.3 寄存器设定步骤

存取闪存时所需的寄存器设定流程如图 28.3 所示。

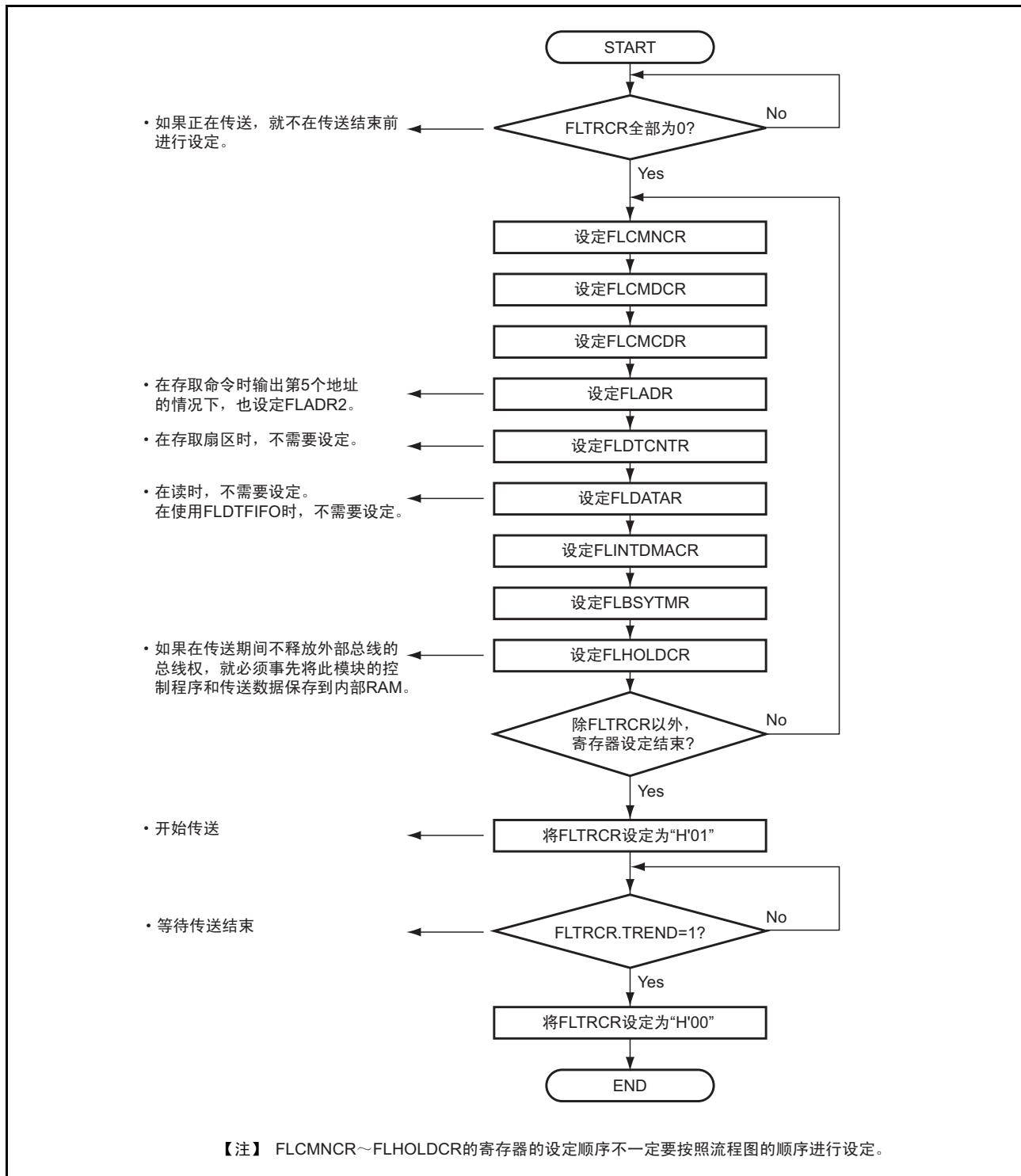


图 28.3 寄存器设定流程

28.4.4 命令存取模式

命令存取模式是通过给寄存器设定对闪存发行的命令、地址、数据、读写方向和次数等进行闪存存取的模式。输入/输出数据能使用 FLDTFIFO 进行 DMA 传送。

(1) NAND 型闪存的存取

NAND 型闪存的读操作例子如图 28.4 所示。在此例子中，将第 1 个命令指定为“H'00”，地址长度指定为 3 字节。这是给数据计数器指定 8 字节的读字节数的情况。

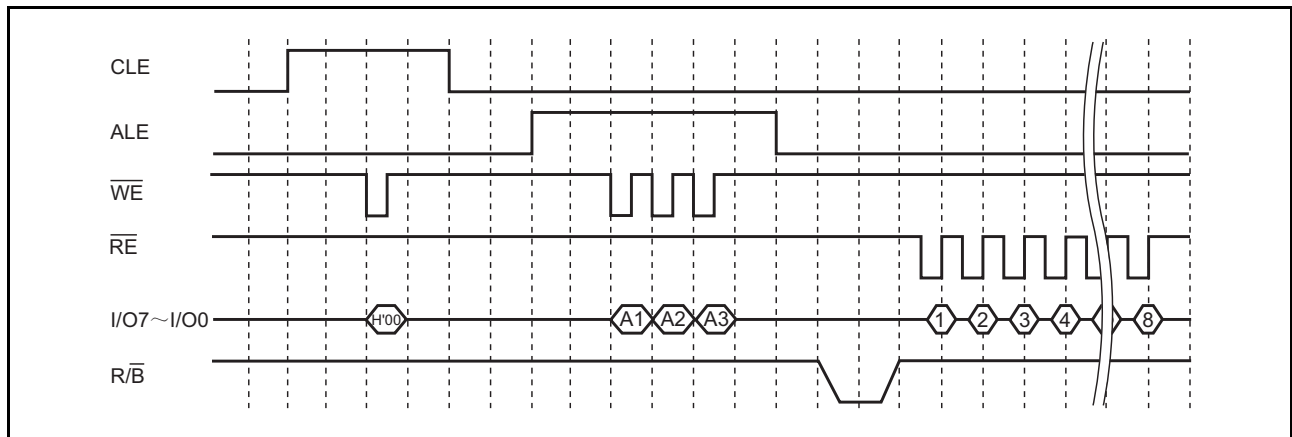


图 28.4 NAND 型闪存的读操作时序

进行 NAND 型闪存编程时的波形如图 28.5 和图 28.6 所示。

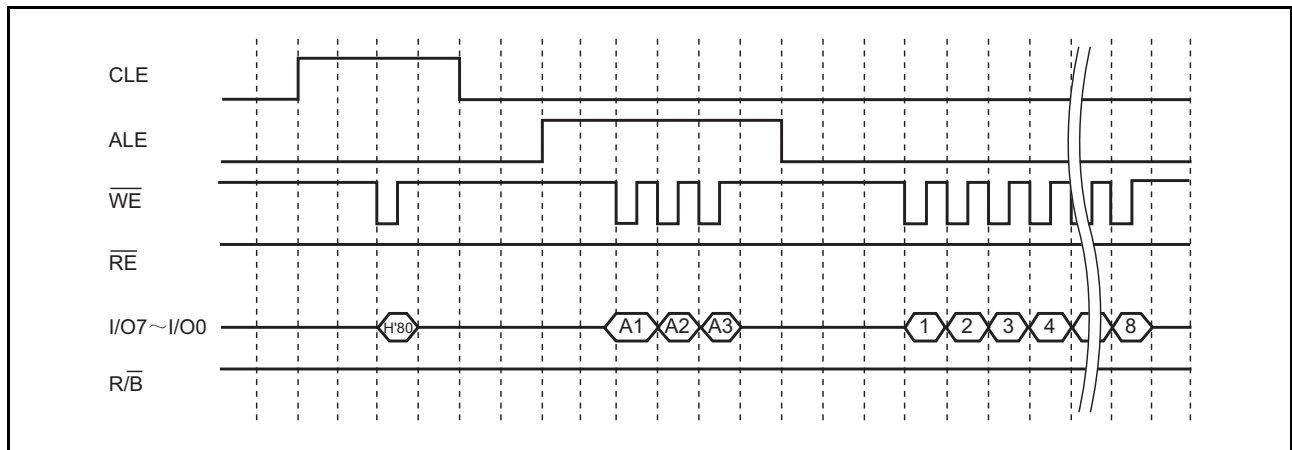


图 28.5 NAND 型闪存的编程时序 (1)

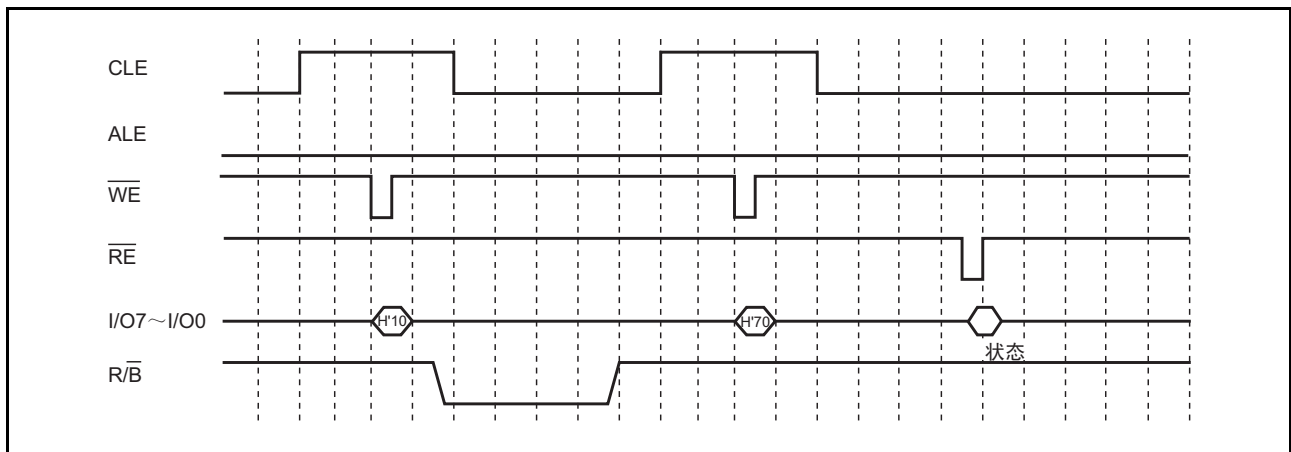


图 28.6 NAND 型闪存的编程时序 (2)

(2) NAND 型闪存 (2048+64 字节) 的存取

NAND 型闪存 (2048+64 字节) 的读操作例子如图 28.7 所示。在此例子中, 将第 1 个命令指定为 “H'00”, 第 2 个命令指定为 “H'30”, 地址长度指定为 4 字节。这是给数据计数器指定 4 字节的读字节数的情况。

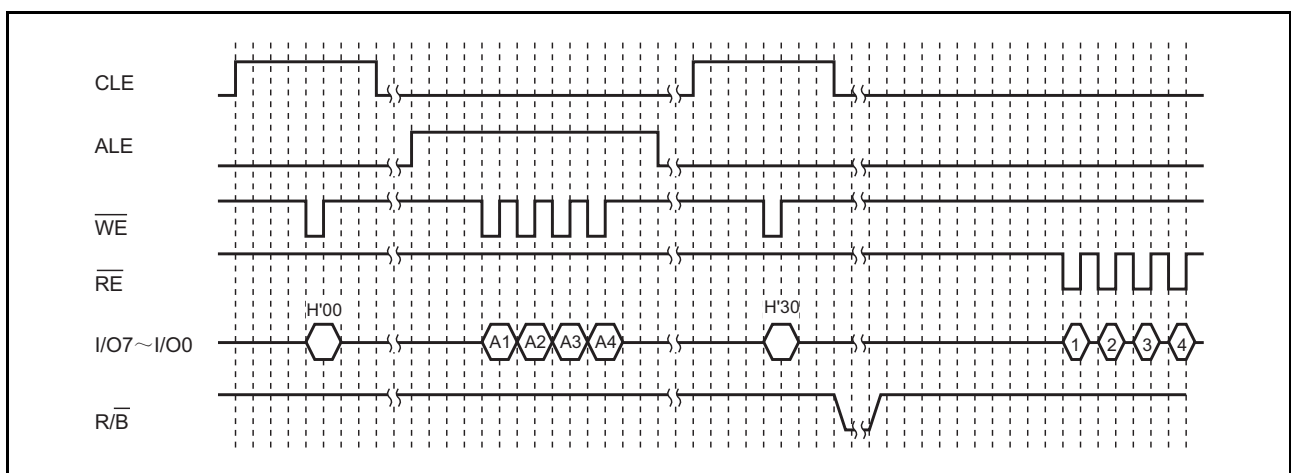


图 28.7 NAND 型闪存的读操作时序

进行 NAND 型闪存（2048+64 字节）编程时的波形如图 28.8 和图 28.9 所示。

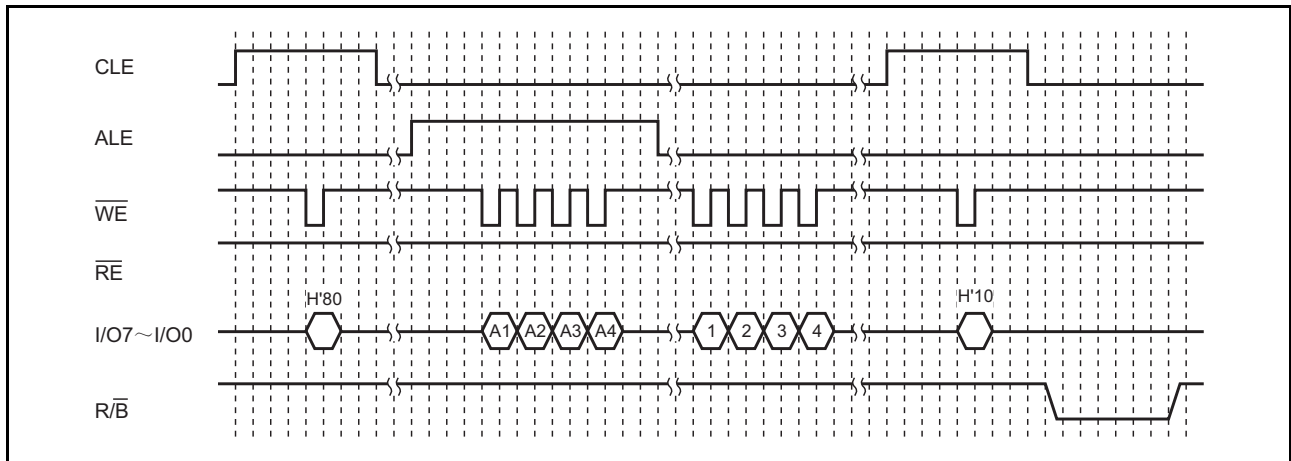


图 28.8 NAND 型闪存的编程时序 (1)

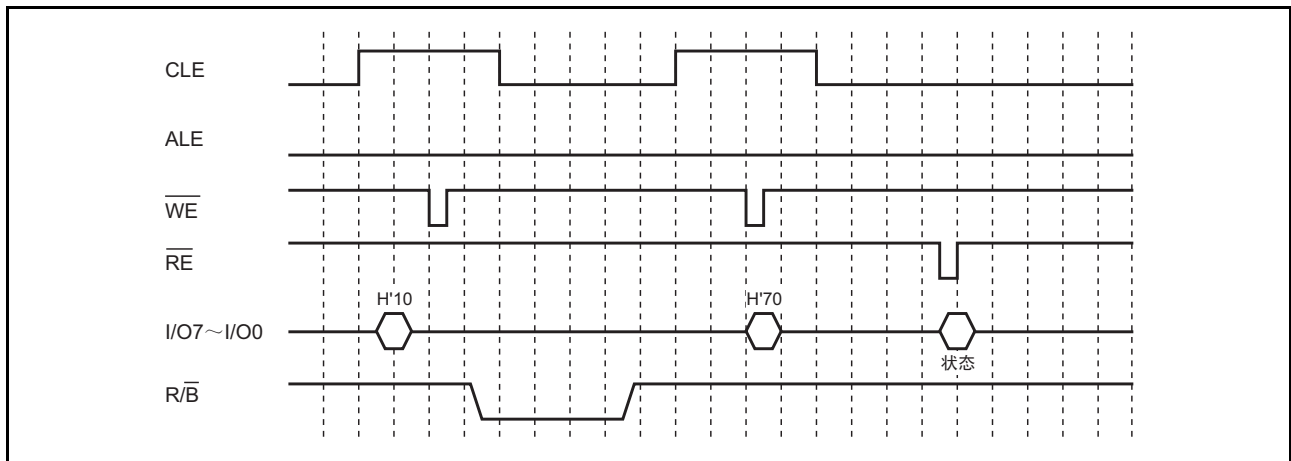


图 28.9 NAND 型闪存的编程时序 (2)

28.4.5 扇区存取模式

在扇区存取模式中，能通过指定要存取的扇区号，以扇区为单位进行读和编程。

512 字节的数据保存在 FLDTFIFO，16 字节的管理码保存在 FLECFIFO，因此设定 FLINTDMACR 的 DREQ1EN 和 DREQ0EN，分别进行 DMA 传送。

闪存内的扇区（数据 + 管理码）和地址空间上的存储器之间的 DMA 传送的关系如图 28.10 所示。

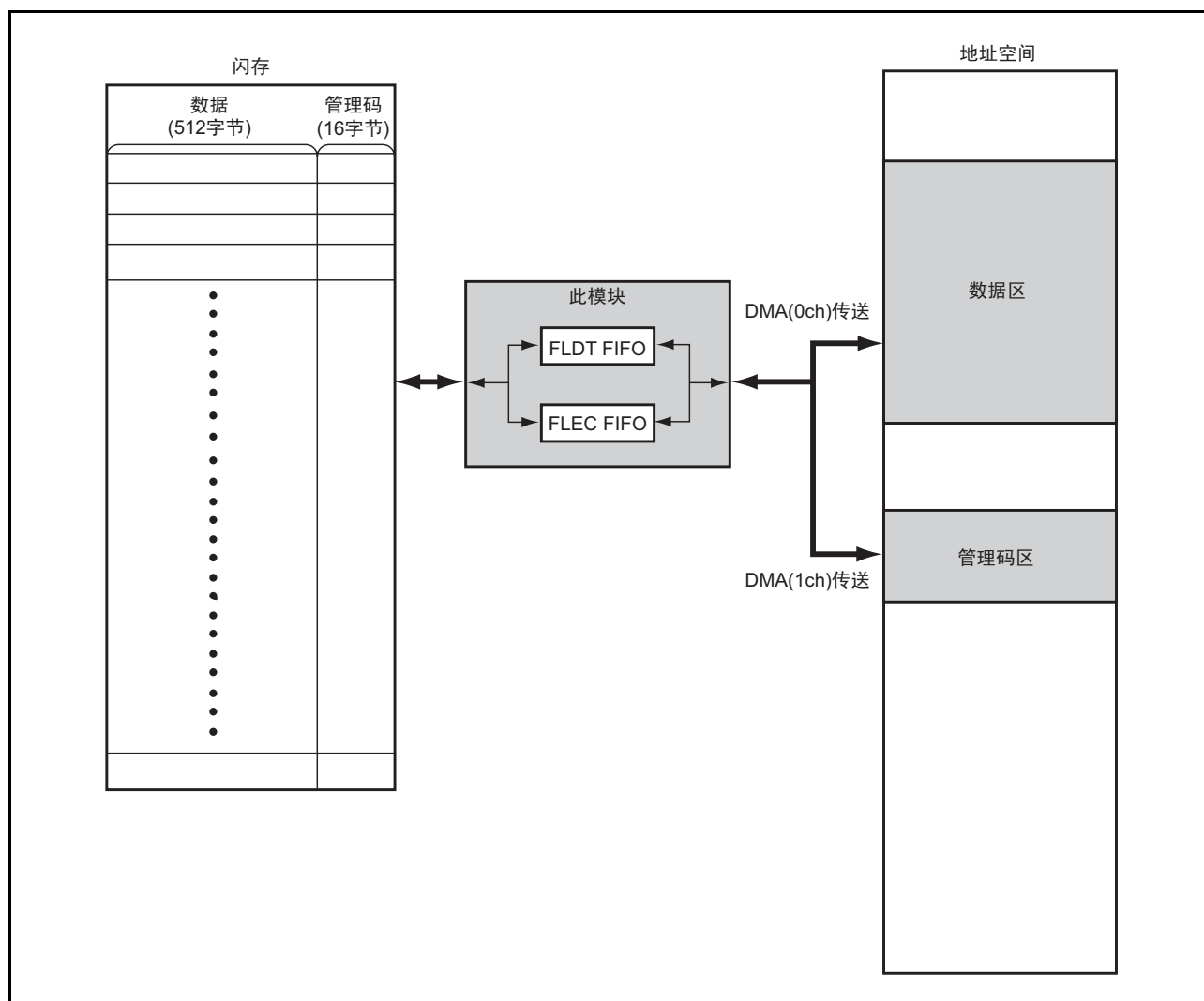


图 28.10 DMA 传送和扇区（数据和管理码）、存储器和 DMA 传送的关系模式图

(1) 扇区地址

NAND 型闪存的物理扇区地址和闪存的地址的关系如图 28.11 所示。

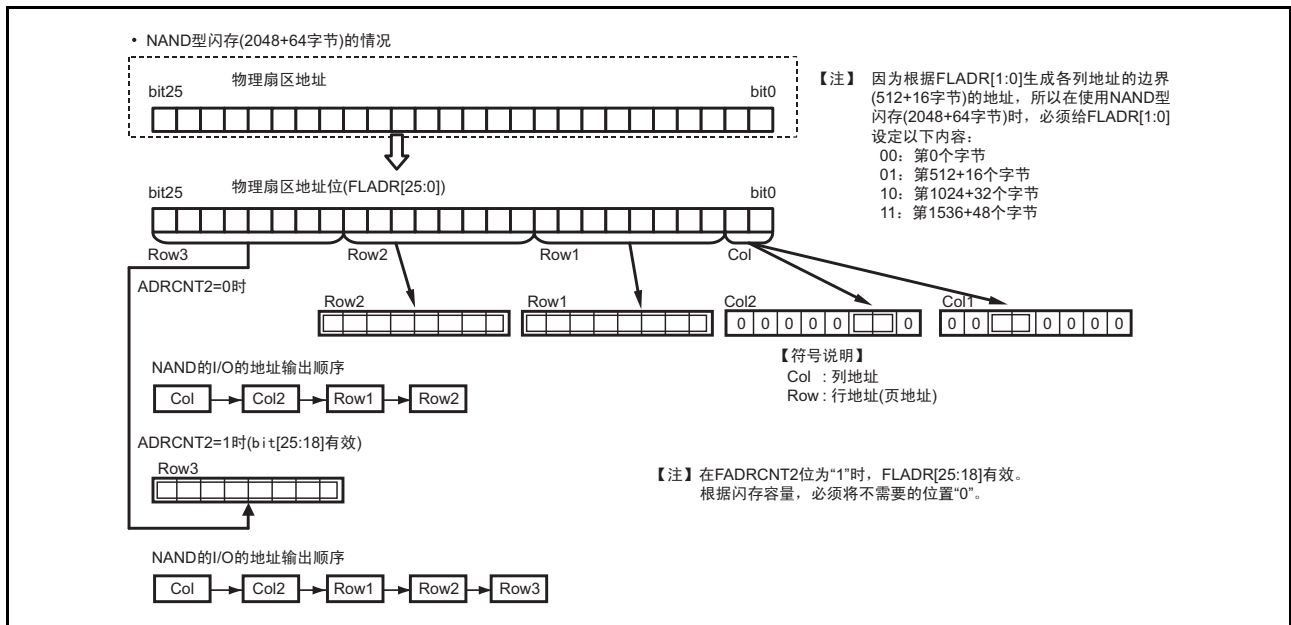


图 28.11 扇区号和 NAND 型闪存地址的展开例子

(2) 连续的扇区存取

能通过指定 NAND 型闪存的起始扇区地址和扇区传送次数进行连续扇区的读和编程。在传送中途存在坏扇区并且物理扇区不连续的 0 ~ 40 的逻辑扇区时, 物理扇区指定寄存器和扇区传送次数指定寄存器的设定例子如图 28.12 所示。

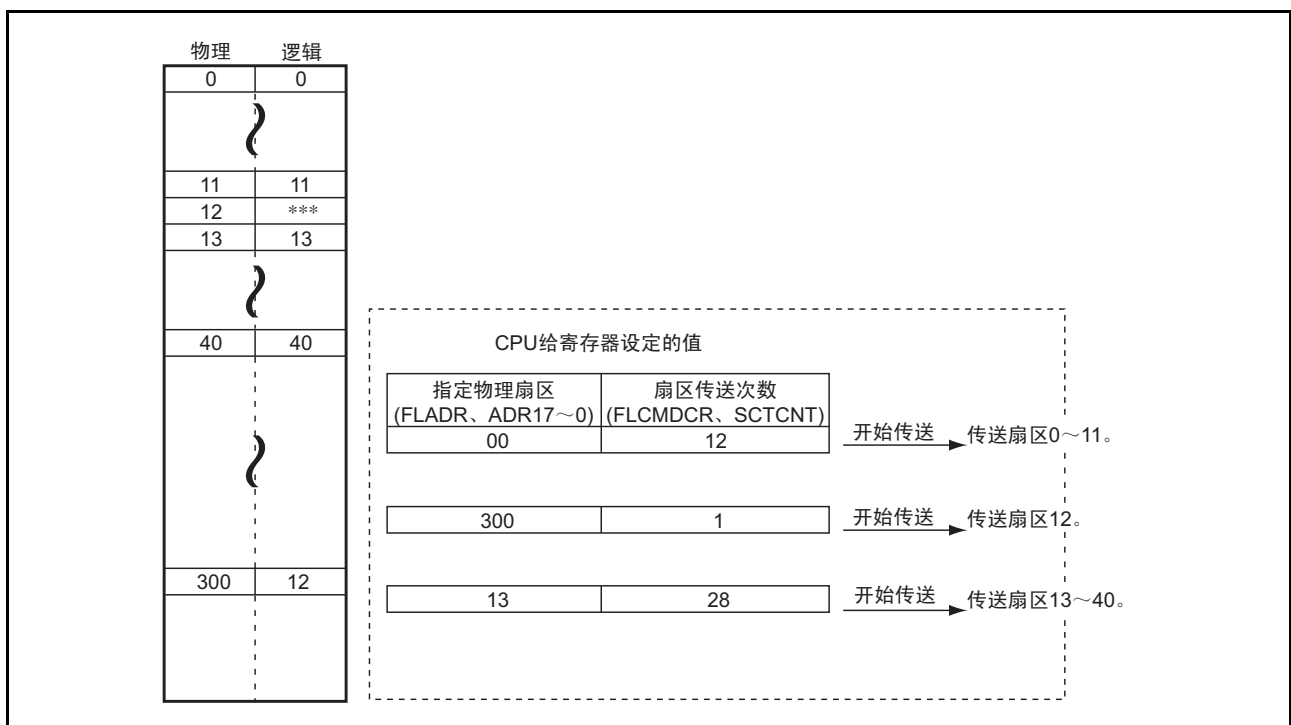


图 28.12 存在坏扇区时的扇区存取例子

(3) 扇区存取模式中的闪存存取

在扇区存取模式中进行 NAND 型闪存的读和编程时的波形如图 28.13 和图 28.14 所示。

图 28.13 是对存储器容量为 1G 位大容量块产品进行编程的波形。在存取跨越多个页的连续扇区时，通过图 28.13 的波形按页（2048+64 字节）进行闪存的编程。

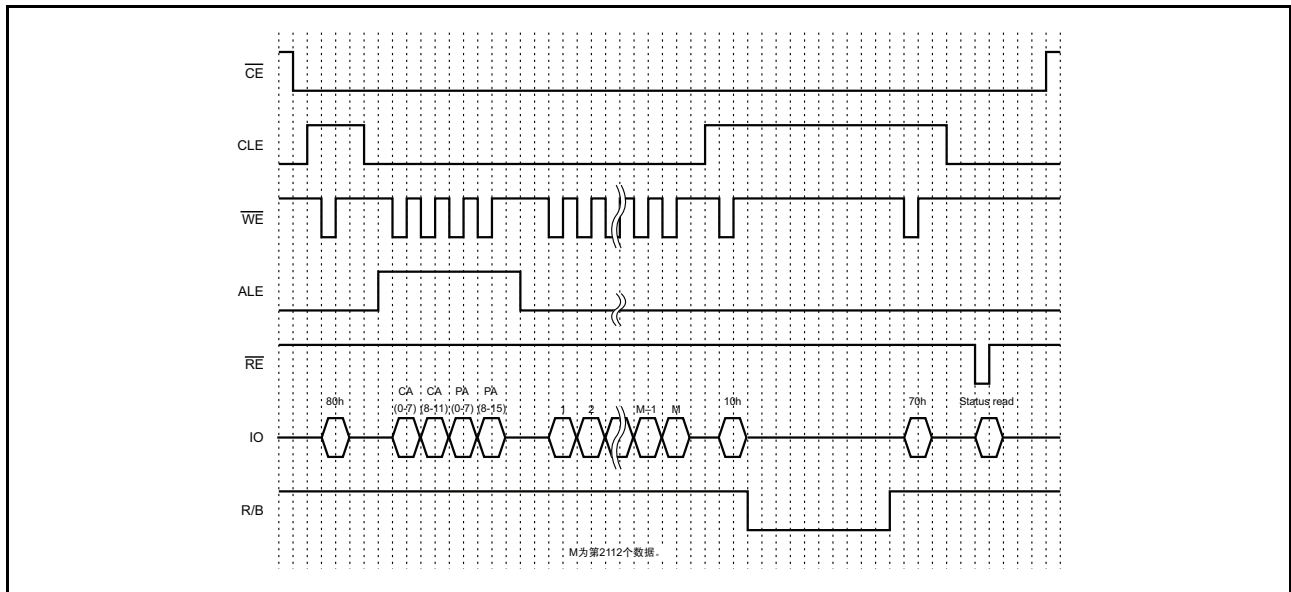


图 28.13 NAND 型闪存的编程时序

图 28.14 是对存储器容量为 1G 位大容量块产品进行读操作的波形。在存取跨越多个页的连续扇区时，通过图 28.14 的波形按页（2048+64 字节）读闪存的数据。

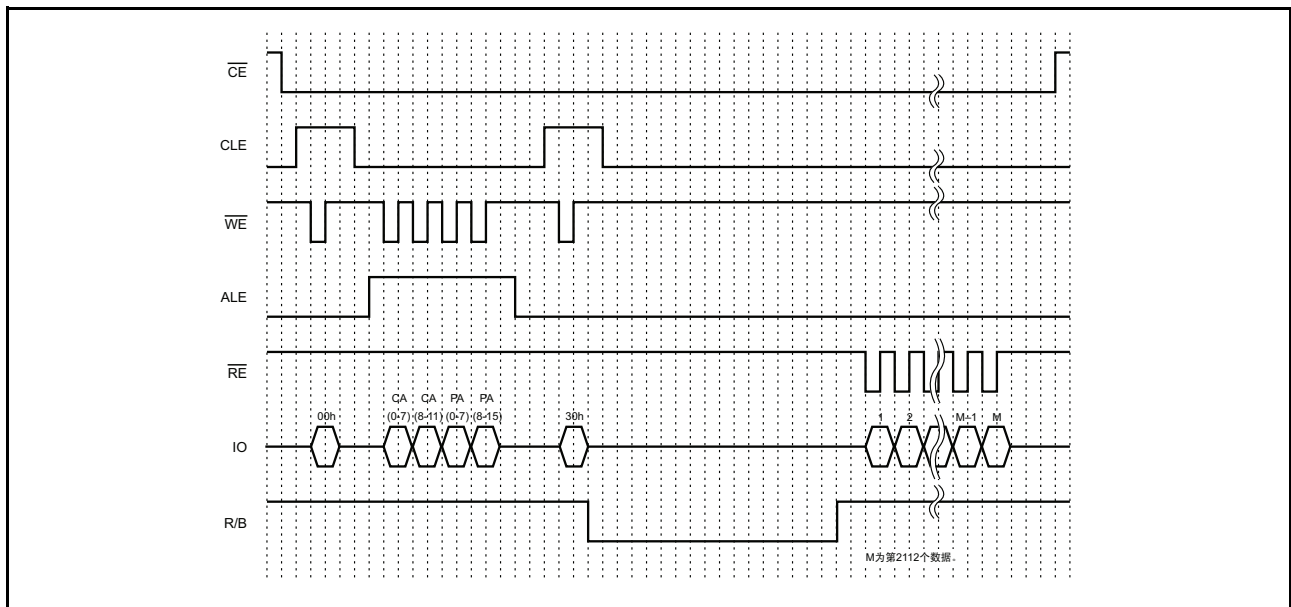


图 28.14 NAND 型闪存的读操作时序（扇区存取模式）

28.4.6 状态的读操作

此模块能读 NAND 型闪存的状态寄存器的值。能从 I/O7 ~ 0 输入状态寄存器的值并且被保存到 FLBSYCNT 的 STAT[7:0] 位，CPU 能读此值。如果在将状态寄存器的值保存到 FLBSYCNT 的 STAT[7:0] 位时检测到编程错误和擦除错误，就将 FLINTDMACR 的 STERB 位置“1”。此时，如果将 FLINTDMACR 的 STERINTE 位设定为允许中断，就向 CPU 请求中断。另外，如果在存取连续扇区的中途发生状态错误，就将 FLTRCR 的 TREND 位置“1”，结束处理。

(1) NAND 型闪存的状态读取

通过给 NAND 型闪存输入命令“H'70”，能读到 NAND 型闪存的状态。如果将 FLCMDCR 的 DOSR 位置“1”并且在命令存取模式或者扇区存取模式中进行编程，此模块就自动将“H'70”输入到 NAND 型闪存，进行状态的读取。在读 NAND 型闪存的状态时，从 I/O7 ~ 0 输入的状态寄存器各位的含义如表 28.3 所示。

表 28.3 NAND 型闪存的状态读取

I/O	状态 (definition)	说明
I/O7	编程保护	0: 不能编程 1: 能编程
I/O6	就绪 / 忙	0: 忙状态 1: 就绪状态
I/O5 ~ 1	保留	—
I/O0	编程 / 擦除	0: Pass (成功) 1: Fail (失败)

28.5 中断处理

此模块有 5 种中断源（状态错误、就绪 / 忙超时错误、传送结束、FIFO0 传送请求、FIFO1 传送请求）。全部的中断源有独立的中断标志，如果通过中断允许位允许中断，就独立向 CPU 请求中断。状态错误与就绪 / 忙超时错误，对 CPU 使用公共的 FLSTE 中断。

表 28.4 中断请求

中断源	中断标志	允许位	含义	优先级
FLSTE 中断	STERB	STERINTE	状态错误	高 ↑ ↓ 低
	BTOERB	RBERINTE	就绪 / 忙超时错误	
FLTEND 中断	TREND	TEINTE	传送结束	
FLTRQ0 中断	TRREQF0	TRINTE0	FIFO0 传送请求	
FLTRQ1 中断	TRREQF1	TRINTE1	FIFO1 传送请求	

28.6 DMA 传送的设定

此模块能分别从数据区 FLDTFIFO 和管理码区 FLECFIFO 请求 DMA 传送。在各存取模式中是否能进行 DMA 传送如表 28.5 所示。

表 28.5 DMA 传送的设定

	扇区存取模式	命令存取模式
FLDTFIFO	能	能
FLECFIFO	能	不能

有关直接存储器存取控制器的设定，请参照“11. 直接存储器存取控制器”。

28.7 使用时的注意事项

28.7.1 外部总线权释放的时序

此模块与就绪 / 忙的状态无关，在一系列处理结束时将 $\overline{\text{FCE}}$ 置为无效电平。必须注意：如果将公共控制寄存器（FLCMNCR）的 bit21（BUSYON）置“0”，即使在忙的期间，也在结束此模块的处理时将 $\overline{\text{FCE}}$ 置为无效电平，释放总线权。当 BUSYON 位为“0”时，通过将命令控制寄存器（FLCMDRCR）的 bit24（DOSR）置“1”进行状态的读取，即使在忙的期间也能获得总线权。

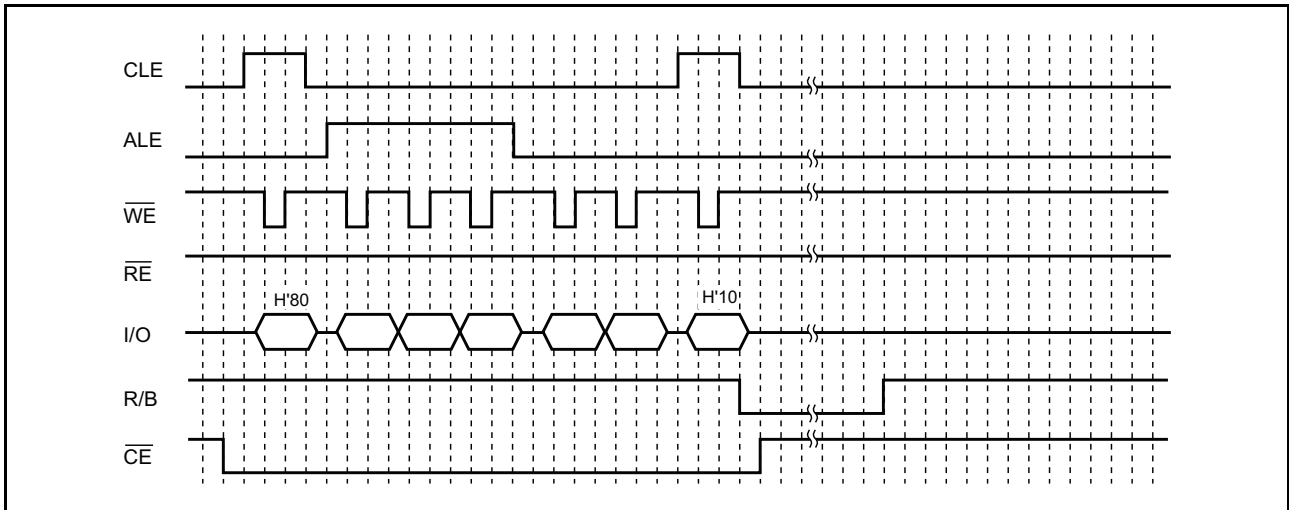


图 28.15 BUSYON 位为“0”并且 DOSR 位为“0”（进行闪存的编程）

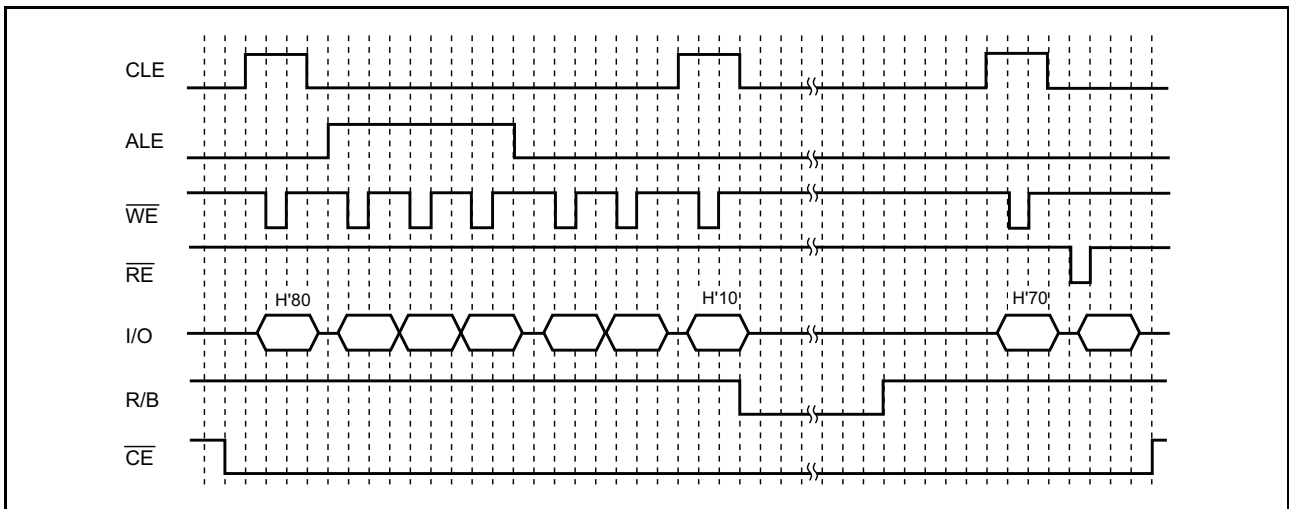


图 28.16 BUSYON 位为“0”并且 DOSR 位为“1”（进行闪存的编程）

28.7.2 有关 SNAND 位的注意事项

在使用公共控制寄存器（FLCMNCR）的 SNAND 位时，与命令控制寄存器（FLCMDRCR）的 DOCMD1 位和 DOCMD2 位的设定无关，只支持第 1 个命令和第 2 个命令。

当不发行命令或者只发行第 1 命令时，必须将 SNAND 位置“0”。

29. USB2.0 主机 / 功能模块

此模块是具有 USB 主机控制器功能和功能控制器功能的 USB 控制器。当选择主机控制器功能时，支持 USB (Universal Serial Bus) 规格 2.0 的高速传送、全速传送和低速传送；当使用功能控制器功能时，支持高速传送和全速传送。此模块还内置 USB 收发器，支持 USB 规格定义的全部传送类型。

此模块内置用于数据传送的 8K 字节缓冲存储器，最多能使用 10 个管道。能给管道 1 ~ 9 分配与进行通信的外围设备以及用户系统相符的任意端点号。

29.1 特点

(1) 内置支持 USB 高速的主机控制器和功能控制器。

- 内置 USB 主机控制器和功能控制器。
- 能通过设定寄存器进行 USB 主机控制器功能和功能控制器功能的切换。
- 内置 USB 收发器。

(2) 能够实现外接元件少并且省空间的安装。

- 内置 D+ 上拉电阻 (功能控制器运行时)。
- 内置 D+/D- 下拉电阻 (主机控制器运行时)。
- 内置 D+/D- 终端电阻 (高速运行时)。
- 内置 D+/D- 输出电阻 (全速运行时)。

(3) 支持 USB 通信的全部数据传送类型。

- 控制传送
- 批量传送
- 中断传送 (不支持 High Bandwidth)
- 等时传送 (不支持 High Bandwidth)

(4) 内部总线接口

- 内置 2 个通道的 DMA 接口。

(5) 管道配置

- 内置 8K 字节用于 USB 通信的缓冲存储器。
- 最多可选择 10 个管道 (包含默认控制管道)。
- 可编程的管道结构
- 能给管道 1 ~ 9 分配任意的端点号。
- 各管道能设定的传送条件如下：
 - 管道 0: 是控制传送的专用管道 (默认控制管道: DCP)，固定为 64 字节的单缓冲器。
 - 管道 1 和管道 2: 是能选择批量传送或者等时传送的管道，连续传送模式，可编程的的缓冲器大小 (最大能指定 2K 字节的双缓冲器)。
 - 管道 3 ~ 5: 是批量传送的专用管道，连续传送模式，缓冲器大小为可编程 (最大能指定 2K 字节的双缓冲器)。
 - 管道 6 ~ 9: 是中断传送的专用管道，固定为 64 字节的单缓冲器。

(6) 选择主机控制器功能时的特点

- 支持高速传送（480Mbps）、全速传送（12Mbps）和低速传送（1.5Mbps）。
- 能通过一段集线器连接多个外围设备进行通信。
- 自动应答复位握手信号。
- 自动安排SOF和信息包的发送时间。
- 能设定等时传送和中断传送的传送间隔。

(7) 选择功能控制器功能时的特点

- 支持高速传送（480Mbps）和全速传送（12Mbps）。
- 能通过自动应答复位握手信号来自动识别高速运行和全速运行。
- 控制传送阶段管理功能
- 设备状态管理功能
- 对SET_ADDRESS请求的自动应答功能
- NAK应答中断功能（NRDY）
- SOF内插功能

(8) 其他功能

- 事务计数产生的传送结束功能
- BRDY中断事件通知时序的变更功能（BFRE）
- 读DnFIFO（n=0、1）端口指定的管道数据后的缓冲存储器自动清除功能（DCLRM）
- 传送结束产生的应答PID的NAK设定功能（SHTNAK）

29.2 输入 / 输出引脚

引脚结构如表 29.1 所示。

表 29.1 引脚结构

区分	名称	引脚名	输入 / 输出	功能
USB 总线接口	USB D+ 数据	DP	输入 / 输出	USB 内部收发器的 D+ 输入 / 输出引脚 必须连接 USB 总线的 D+ 引脚。
	USB D- 数据	DM	输入 / 输出	USB 内部收发器的 D- 输入 / 输出引脚 必须连接 USB 总线的 D- 引脚。
VBUS 监视输入	VBUS 输入	VBUS	输入	USB 电缆连接监视引脚 必须连接 USB 总线的 Vbus。能检测到连接或者切断 Vbus。如果不连接 USB 总线的 Vbus，就必须固定为 5V。当选择主机控制器功能时，也必须提供 5V。 【注】 不能给连接的外围设备提供 Vbus。
基准电阻	基准输入	REFRIN	输入	用于连接基准电阻的引脚 必须通过 $5.6K\Omega \pm 1\%$ 电阻连接 USBAPV _{SS} 。
时钟	用于 USB 的晶体谐振器 / 外部时钟	USB_X1	输入	连接用于 USB 的晶体谐振器。USB_X1 引脚也能输入外部时钟。
		USB_X2	输出	
电源	收发器部模拟引脚电源	USBAPV _{CC}	输入	用于引脚的电源
	收发器部模拟引脚接地 *	USBAPV _{SS}	输入	用于引脚的接地
	收发器部数字引脚电源 *	USBDPV _{CC}	输入	用于引脚的电源
	收发器部数字引脚接地 *	USBDPV _{SS}	输入	用于引脚的接地
	收发器部模拟内核电源	USBAV _{CC}	输入	用于内核的电源
	收发器部模拟内核接地 *	USBAV _{SS}	输入	用于内核的接地
	收发器部数字内核电源 *	USBDV _{CC}	输入	用于内核的电源
	收发器部数字内核接地 *	USBDV _{SS}	输入	用于内核的接地
电源	USB480MHz 电源 *	USBUV _{CC}	输入	480MHz 工作部电源
	USB480MHz 接地 *	USBUV _{SS}	输入	480MHz 工作部接地

【注】 * SH7269 (BGA) 没有此引脚。

29.3 寄存器说明

寄存器结构如表 29.2 所示。

表 29.2 寄存器结构

寄存器名	略称	R/W	地址	存取长度
系统配置控制寄存器	SYSCFG	R/W	H'E801 0000	16
CPU 总线等待设定寄存器	BUSWAIT	R/W	H'E801 0002	16
系统配置状态寄存器	SYSSTS	R	H'E801 0004	16
设备状态控制寄存器	DVSTCTR	R/W	H'E801 0008	16
测试模式寄存器	TESTMODE	R/W	H'E801 000C	16
DMA0-FIFO 总线配置寄存器	D0FBCFG	R/W	H'E801 0010	16
DMA1-FIFO 总线配置寄存器	D1FBCFG	R/W	H'E801 0012	16
CFIFO 端口寄存器	CFIFO	R/W	H'E801 0014	8、16、32
D0FIFO 端口寄存器	D0FIFO	R/W	H'E801 0018	8、16、32
D1FIFO 端口寄存器	D1FIFO	R/W	H'E801 001C	8、16、32
CFIFO 端口选择寄存器	CFIFOSEL	R/W	H'E801 0020	16
CFIFO 端口控制寄存器	CFIFOCTR	R/W	H'E801 0022	16
D0FIFO 端口选择寄存器	D0FIFOSEL	R/W	H'E801 0028	16
D0FIFO 端口控制寄存器	D0FIFOCTR	R/W	H'E801 002A	16
D1FIFO 端口选择寄存器	D1FIFOSEL	R/W	H'E801 002C	16
D1FIFO 端口控制寄存器	D1FIFOCTR	R/W	H'E801 002E	16
中断允许寄存器 0	INTENB0	R/W	H'E801 0030	16
中断允许寄存器 1	INTENB1	R/W	H'E801 0032	16
BRDY 中断允许寄存器	BRDYENB	R/W	H'E801 0036	16
NRDY 中断允许寄存器	NRDYENB	R/W	H'E801 0038	16
BEMP 中断允许寄存器	BEMPENB	R/W	H'E801 003A	16
SOF 输出配置寄存器	SOFCFG	R/W	H'E801 003C	16
中断状态寄存器 0	INTSTS0	R/W	H'E801 0040	16
中断状态寄存器 1	INTSTS1	R/W	H'E801 0042	16
BRDY 中断状态寄存器	BRDYSTS	R/W	H'E801 0046	16
NRDY 中断状态寄存器	NRDYSTS	R/W	H'E801 0048	16
BEMP 中断状态寄存器	BEMPSTS	R/W	H'E801 004A	16
帧号寄存器	FRMNUM	R/W	H'E801 004C	16
μ 帧号寄存器	UFRMNUM	R	H'E801 004E	16
USB 地址寄存器	USBADDR	R	H'E801 0050	16
USB 请求类型寄存器	USBREQ	R/W	H'E801 0054	16
USB 请求值寄存器	USBVAL	R/W	H'E801 0056	16
USB 请求变址寄存器	USBINDX	R/W	H'E801 0058	16
USB 请求长度寄存器	USBLENG	R/W	H'E801 005A	16
DCP 配置寄存器	DCPCFG	R/W	H'E801 005C	16
DCP 最大信息包长度寄存器	DCPMAXP	R/W	H'E801 005E	16
DCP 控制寄存器	DCPCTR	R/W	H'E801 0060	16
管道窗口选择寄存器	PIPESEL	R/W	H'E801 0064	16

寄存器名	略称	R/W	地址	存取长度
管道配置寄存器	PIPECFG	R/W	H'E801 0068	16
管道缓冲器指定寄存器	PIPEBUF	R/W	H'E801 006A	16
管道最大信息包长度寄存器	PIPEMAXP	R/W	H'E801 006C	16
管道周期控制寄存器	PIPEPERI	R/W	H'E801 006E	16
管道 1 的控制寄存器	PIPE1CTR	R/W	H'E801 0070	16
管道 2 的控制寄存器	PIPE2CTR	R/W	H'E801 0072	16
管道 3 的控制寄存器	PIPE3CTR	R/W	H'E801 0074	16
管道 4 的控制寄存器	PIPE4CTR	R/W	H'E801 0076	16
管道 5 的控制寄存器	PIPE5CTR	R/W	H'E801 0078	16
管道 6 的控制寄存器	PIPE6CTR	R/W	H'E801 007A	16
管道 7 的控制寄存器	PIPE7CTR	R/W	H'E801 007C	16
管道 8 的控制寄存器	PIPE8CTR	R/W	H'E801 007E	16
管道 9 的控制寄存器	PIPE9CTR	R/W	H'E801 0080	16
管道 1 的事务计数允许寄存器	PIPE1TRE	R/W	H'E801 0090	16
管道 1 的事务计数寄存器	PIPE1TRN	R/W	H'E801 0092	16
管道 2 的事务计数允许寄存器	PIPE2TRE	R/W	H'E801 0094	16
管道 2 的事务计数寄存器	PIPE2TRN	R/W	H'E801 0096	16
管道 3 的事务计数允许寄存器	PIPE3TRE	R/W	H'E801 0098	16
管道 3 的事务计数寄存器	PIPE3TRN	R/W	H'E801 009A	16
管道 4 的事务计数允许寄存器	PIPE4TRE	R/W	H'E801 009C	16
管道 4 的事务计数寄存器	PIPE4TRN	R/W	H'E801 009E	16
管道 5 的事务计数允许寄存器	PIPE5TRE	R/W	H'E801 00A0	16
管道 5 的事务计数寄存器	PIPE5TRN	R/W	H'E801 00A2	16
设备地址 0 的配置寄存器	DEVADD0	R/W	H'E801 00D0	16
设备地址 1 的配置寄存器	DEVADD1	R/W	H'E801 00D2	16
设备地址 2 的配置寄存器	DEVADD2	R/W	H'E801 00D4	16
设备地址 3 的配置寄存器	DEVADD3	R/W	H'E801 00D6	16
设备地址 4 的配置寄存器	DEVADD4	R/W	H'E801 00D8	16
设备地址 5 的配置寄存器	DEVADD5	R/W	H'E801 00DA	16
设备地址 6 的配置寄存器	DEVADD6	R/W	H'E801 00DC	16
设备地址 7 的配置寄存器	DEVADD7	R/W	H'E801 00DE	16
设备地址 8 的配置寄存器	DEVADD8	R/W	H'E801 00E0	16
设备地址 9 的配置寄存器	DEVADD9	R/W	H'E801 00E2	16
设备地址 A 的配置寄存器	DEVADDA	R/W	H'E801 00E4	16

29.3.1 系统配置控制寄存器 (SYSCFG)

SYSCFG 允许高速运行，选择主机控制器功能或者功能控制器功能，控制 DP 引脚和 DM 引脚，选择输入时钟以及允许此模块的运行。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SCKE	—	—	HSE	DCF	DRPD	DPRPU	UCKFSEL	UCKPSEL	UPLLE	USBE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10	SCKE	0	R/W	USB 模块时钟允许 指定停止或者允许给此模块提供时钟。 0: 停止给 USB 模块提供时钟 1: 允许给 USB 模块提供时钟 当此位为“0”时，只能读写此寄存器和 BUSWAIT 寄存器。 能读但是不能写 USB 模块内的其他寄存器。 如果在通过 UCKFSEL 位选择 12MHz 选择时更改此位，就必须遵守以下规则： • 要将此位置“1”时，必须在将 UPLLE 位置“1”后至少等待 1ms。 • 要转移到软件待机模式或者 USB 模块待机模式时，必须将此位置“0”。
9、8	—	全 0	R	保留位 读写值都为“0”。
7	HSE	0	R/W	高速运行允许 0: 禁止高速运行 当选择功能控制器功能时，只限于全速运行。 当选择主机控制器功能时，进行全速运行或者低速运行。 1: 允许高速运行（此模块检测通信速度） (1) 选择主机控制器功能的情况 当 HSE 位为“0”时，USB 端口进行低速运行或者全速运行。 如果检测到 USB 端口连接了低速外围设备，就必须将 HSE 位置“0”。 当 HSE 位为“1”时，此模块执行复位握手信号协议。根据该结果，USB 端口自动进行高速运行或者全速运行。 必须在检测到连接（检测到 ATTCH 中断）后并且在进行 USB 总线复位前（将 USBRESET 位置“1”前）更改此位。 (2) 选择功能控制器功能的情况 当 HSE 位为“0”时，此模块进行全速运行。 当 HSE 位为“1”时，此模块执行复位握手信号协议。根据该结果，自动进行高速运行或者全速运行。 必须在 DPRPU 位为“0”时改写此位。

位	位名	初始值	R/W	说明
6	DCFM	0	R/W	<p>控制器功能选择 选择此模块的功能。 0: 选择功能控制器功能 1: 选择主机控制器功能 必须在 DPRPU 位为 “0” 并且 DRPD 位为 “0” 时更改此位。</p>
5	DRPD	0	R/W	<p>D+/D- 线路电阻控制 当选择主机控制器功能时, 指定禁止或者允许 D+/D- 线路的下拉。 0: 禁止下拉 1: 允许下拉 当选择主机控制器功能时, 必须将此位置 “1”; 当选择功能控制器功能时, 必须将此位置 “0”。</p>
4	DPRPU	0	R/W	<p>D+ 线路电阻控制 当选择功能控制器功能时, 指定禁止或者允许 D+ 线路的上拉。 0: 禁止上拉 1: 允许上拉 当选择功能控制器功能时, 如果将此位置 “1”, 此模块就能上拉 D+ 线路并且将连接状态通知 USB 主机。通过将此位从 “1” 改为 “0”, 此模块取消 D+ 线路的上拉, 所以能将断开状态通知 USB 主机。 当选择功能控制器功能时, 必须将此位置 “1”; 当选择主机控制器功能时, 必须将此位置 “0”。 【注】 必须在切断 USB 时将此位改为 “0”。在将此位从 “1” 改为 “0” 时, 必须追加进行以下的处理: 1. 将 DPRPU 位置 “0”。 2. 至少等待 1μs。 3. 将 DCFM 位置 “1”。 4. 至少等待 200ns。 5. 将 DCFM 位置 “0”。</p>
3	UCKFSEL	0	R/W	<p>时钟输入频率选择 选择给此模块提供的时钟输入频率。 0: 48MHz 输入 1: 12MHz 输入 必须在 SCKE 位为 “0” 时更改此位。 【注】 在通过 UCKPSEL 位选择 EXTAL 输入时, 必须将此位置 “1”。</p>
2	UCKPSEL	0	R/W	<p>时钟输入引脚选择 选择给此模块提供的时钟输入引脚 0: USB_X1 输入 1: EXTAL 输入 必须在 SCKE 位为 “0” 时更改此位。</p>
1	UPLLE	0	R/W	<p>USB 内部 PLL 运行允许 在通过 UCKFSEL 位选择 12MHz 输入时, 指定禁止或者允许 USB 内部 PLL 的运行。 0: 禁止内部 PLL 运行 1: 允许内部 PLL 运行 如果通过 UCKFSEL 位选择 48MHz 输入, 此位的设定就无效。 必须在 SCKE 位为 “0” 时更改此位。 如果在通过 UCKFSEL 位选择 12MHz 时转移到软件待机模式或者 USB 模块待机模式, 就必须将此位置 “0”。</p>

位	位名	初始值	R/W	说明
0	USBE	0	R/W	<p>USB 模块运行允许 指定禁止或者允许此模块的运行。 0: 禁止 USB 模块运行 1: 允许 USB 模块运行</p> <p>在将此位从“1”改为“0”时被初始化的寄存器和位如表 29.3 和表 29.4 所示。</p> <p>必须在 SCKE 位为“1”时更改此位。</p> <p>当选择主机控制器功能时，必须将 DPRD 位置“1”，在消除 LNST 位的颤动并且确认 USB 总线状态稳定后，将 USBE 位置“1”。</p>

表 29.3 在给 USBE 位写“0”时被初始化的寄存器（选择功能控制器功能的情况）

寄存器名	位名	备注
SYSSTS	LNST	当选择主机控制器功能时，保持原来的值。
DVSTCTR	RHST	
INTSTS0	DVSQ	当选择主机控制器功能时，保持原来的值。
USBADDR	USBADDR	当选择主机控制器功能时，保持原来的值。
USEREQ	BRequest、bmRequestType	当选择主机控制器功能时，保持原来的值。
USBVAL	wValue	当选择主机控制器功能时，保持原来的值。
USBINDX	wIndex	当选择主机控制器功能时，保持原来的值。
USBLENG	wLength	当选择主机控制器功能时，保持原来的值。

表 29.4 在给 USBE 位写“0”时被初始化的寄存器（选择主机控制器功能的情况）

寄存器名	位名	备注
DVSTCTR	RHST	
FRMNUM	FRNM	当选择功能控制器功能时，保持原来的值。
UFRMNUM	UFRNM	当选择功能控制器功能时，保持原来的值。

29.3.2 CPU 总线等待寄存器 (BUSWAIT)

BUSWAIT 指定 CPU 存取此模块时的等待数。

即使在 SYSCFG 寄存器的 SCKE 位为“0”时，也能写此寄存器。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	BWAIT[3:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 4	—	全 0	R	保留位 读写值都为“0”。
3 ~ 0	BWAIT[3:0]	1111	R/W	CPU 总线存取等待的指定 指定寄存器的存取等待数（和 FIFO 端口的存取等待共用），详细内容请参 照“29.4.1(5) 寄存器的存取等待控制”。 0000: 0 个等待（2 个存取周期） : 0010: 2 个等待（4 个存取周期） : 0100: 4 个等待（6 个存取周期） : 1111: 15 个等待（17 个存取周期）

29.3.3 系统配置状态寄存器 (SYSSTS)

SYSSTS 是监视 USB 数据总线线路状态 (D+/D- 线路) 的寄存器。

在上电复位和 USB 总线复位时, 对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LNST[1:0]
初始值:	0	0	0	0	0	不定值	0	0	0	0	0	0	0	0	不定值*	不定值*
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10	—	不定值	R	保留位 读取值为不定值, 写入值总是“0”。
9 ~ 2	—	全 0	R	保留位 读写值都为“0”。
1、0	LNST[1:0]	不定值*	R	USB 数据线路状态监视 表示 USB 数据总线线路 (D+ 线路、D- 线路) 的状态。USB 数据总线线路状态如表 29.5 所示。 当选择功能控制器功能时, 必须在连接处理 (将 DPRPU 位置“1”) 后参照这些位; 当选择主机控制器功能时, 必须在允许下拉 (将 DRPD 位置“1”) 后参照这些位。

【注】* 取决于 DP 引脚和 DM 引脚的状态。

表 29.5 USB 数据总线线路状态表

LNST[1]	LNST[0]	低速运行 (只限于选择主机控 制器功能的情况)	全速运行	高速运行	Chirp 运行
0	0	SE0	SE0	Squelch	Squelch
0	1	K-State	J-State	UnSquelch	Chirp J
1	0	J-State	K-State	Invalid	Chirp K
1	1	SE1	SE1	Invalid	Invalid

【符号说明】

Chirp: 在允许高速运行的状态下 (SYSCFG.HSE 为“1”), 正在执行复位握手信号协议 (RHSP)。

Squelch: SE0 或者空闲状态

UnSquelch: 高速 J-State 或者高速 K-State

Chirp J: Chirp J-State

Chirp K: Chirp K-State

Invalid: 无效

29.3.4 设备状态控制寄存器 (DVSTCTR)

DVSTCTR 是控制和确认 USB 数据总线状态的寄存器。

在上电复位时，对此寄存器进行初始化；在 USB 总线复位时，只对 WKUP 位进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	WKUP	RWUPE	USBRST	RESUME	UACT	—	RHST[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W*	R/W	R/W	R/W	R/W	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	WKUP	0	R/W*	唤醒输出 当选择功能控制器功能时，指定禁止或者允许 USB 总线上的远程唤醒（输出恢复信号）。 0: 不输出远程唤醒信号 1: 输出远程唤醒信号 此模块管理远程唤醒信号的输出时间。如果将此位置“1”，此模块就在输出 10ms 的 K-State 后将此位置“0”。 根据 USB 规格，在发送远程唤醒信号前需要至少保持 5ms 的 USB 总线空闲状态。因此，即使在检测到挂起状态后立即给此位写“1”，此模块也在等待 2ms 后输出 K-State。 只在设备状态为挂起状态（INTSTS0.DVSQ 位为“1xx”）并且允许 USB 主机远程唤醒时才能给此位写“1”。在将此位置“1”时，即使在挂起过程中也不能停止内部时钟（必须在 SCKE 位为“1”的状态下给 WKUP 位写“1”）。 当选择主机控制器功能时，必须给此位写“0”。
7	RWUPE	0	R/W	远程唤醒检测允许 当选择主机控制器功能时，指定禁止或者允许下行端口外围设备的远程唤醒（输出恢复信号）。 0: 禁止下行端口的远程唤醒输出 1: 允许下行端口的远程唤醒 如果将此位置“1”，就在检测到远程唤醒信号时检测下行端口的恢复信号（2.5μs 之间的 K-State）并且进行恢复处理（K-State 驱动）。 如果将此位置“0”，即使此模块检测到 USB 端口连接的外围设备的远程唤醒信号（K-State），也被忽视。 如果将此位置“1”，即使在挂起过程中也不能停止内部时钟（必须保持 SCKE 位为“1”的状态），并且不能从挂起状态进行 USB 总线复位（将 USBRST 位置“1”）。在 USB Specification2.0 的情况下，禁止此操作。 当选择功能控制器功能时，必须将此位置“0”。

位	位名	初始值	R/W	说明
6	USBRST	0	R/W	<p>总线复位输出</p> <p>当选择主机控制器功能时，控制 USB 总线复位信号的输出。</p> <p>0: 不输出 USB 总线复位信号</p> <p>1: 输出 USB 总线复位信号</p> <p>当选择主机控制器功能时，如果将此位置“1”，此模块就对 USB 端口进行 SE0 驱动并且进行 USB 总线复位处理。此时，如果 HSE 位为“1”，就执行复位握手信号协议。</p> <p>在 USBRST 位为“1”的期间（在给 USBRST 位写“0”前），此模块继续 SE0 输出。在 USBRST 位为“1”的期间（USB 总线复位期间），必须确保符合 USB Specification2.0 规格的时间。</p> <p>如果在通过程中（UACT 位为“1”）或者恢复过程中（RESUME 位为“1”）给此位写“1”，此模块就不在 UACT 位变为“0”并且 RESUME 位变为“0”的状态前开始 USB 总线复位。</p> <p>必须在 USB 总线复位处理结束（给 USBRST 位写“0”）的同时给 UACT 位写“1”。</p> <p>当选择功能控制器功能时，必须将此位置“0”。</p>
5	RESUME	0	R/W	<p>恢复输出</p> <p>当选择主机控制器功能时，控制恢复信号的输出。</p> <p>0: 不输出恢复信号</p> <p>1: 输出恢复信号</p> <p>如果将此位置“1”，此模块就对端口进行 K-State 驱动并且输出恢复信号。</p> <p>在 RESUME 位为“1”的期间（给 RESUME 位写“0”前），此模块继续进行 K-State 输出。在 RESUME 位为“1”的期间（恢复期间），必须确保符合 USB Specification2.0 规定的时间。</p> <p>只能在挂起过程中给此位写“1”。</p> <p>必须在恢复处理结束（给 RESUME 位写“0”）的同时给 UACT 位写“1”。</p> <p>当选择功能控制器功能时，必须将此位置“0”。</p>
4	UACT	0	R/W	<p>USB 总线允许</p> <p>当选择主机控制器功能时，允许 USB 总线运行（控制将 SOF 或者 μSOF 包发送到 USB 总线）。</p> <p>0: 禁止下行端口运行（禁止发送 SOF/μSOF）</p> <p>1: 允许下行端口运行（允许发送 SOF/μSOF）</p> <p>如果将此位置“1”，此模块就将 USB 端口设定为 USB 总线允许状态，并且进行 SOF 发送和数据的发送和接收。</p> <p>在给 UACT 位写“1”后，在 1 (μ) 帧时间内开始输出 SOF/μSOF。</p> <p>如果将此位置“0”，此模块就在输出 SOF/μSOF 后转移到空闲状态。</p> <p>在以下的情况下，此模块将此位置“0”：</p> <ul style="list-style-type: none"> 在通过程中（将 UACT 位置“1”）检测到 DTCH 中断时 在通过程中（将 UACT 位置“1”）检测到 EOFERR 中断时 <p>必须在 USB 总线复位处理结束时（给 USBRST 位写“0”）或者从挂起进行的恢复处理结束时（给 RESUME 位写“0”）给此位写“1”。</p> <p>当选择功能控制器功能时，必须将此位置“0”。</p>
3	—	0	R	<p>保留位</p> <p>读写值都为“0”。</p>

位	位名	初始值	R/W	说明
2 ~ 0	RHST[2:0]	000	R	<p>复位握手信号 表示复位握手信号的状态。</p> <p>(1) 选择主机控制器功能的情况 000: 通信速度不确定 (在供电或者切断时) 1xx: 正在进行复位握手信号处理 001: 低速连接时 010: 全速连接时 011: 高速连接时 在给 USBRST 位写 “1” 后, 此位变为 “100”。 对于端口, 当 HSE 位为 “1” 并且此模块检测到外围设备的 ChirpK 时此位为 “111”。 在给 USBRST 位写 “0” 并且此模块结束 SE0 驱动时, 此模块确定 RHST 位的值。 在给 UTST 位写 “1xxx” 时 (设定用于 HOST 测试的参数), 此位为 “011”。</p> <p>(2) 选择功能控制器功能的情况 000: 通信速度不确定 100: 正在进行复位握手信号处理 010: 全速连接时 011: 高速连接时 当 HSE 为 “1” 时, 如果此模块检测到 USB 总线复位, 此位就为 “100”。 然后, 此模块输出 ChirpK, 在检测到 3 次 USB 主机的 ChirpJK 时, 此位为 “011”。如果在输出 ChirpK 后的 2.5ms 内不能确定为高速, 此位就为 “010”。 当 HSE 位为 “0” 时, 如果此模块检测到 USB 总线复位, 此位就为 “010”。 在此模块检测到 USB 总线复位后并且确定 RHST 位为 “010” 或者 “011” 时, 产生 DVST 中断。</p>

【注】 * 只能写 “1”。

29.3.5 测试模式寄存器 (TESTMODE)

TESTMODE 控制高速运行时的 USB 测试信号输出。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	UTST[3:0]			
初始值:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	—	1	R	保留位 读写值都为“1”。
7 ~ 4	—	全 0	R	保留位 读写值都为“0”。
3 ~ 0	UTST[3:0]	0000	R/W	<p>测试模式 通过给此位写值，此模块输出高速运行时的 USB 测试信号。 此模块的测试模式运行表如表 29.6 所示。</p> <p>(1) 选择主机控制器功能的情况 能在给 DRPD 位写“1”后设定此位。通过将 DRPD 位置“1”并且将 UACT 位置“1”，此模块进行波形输出；通过写此位，进行高速终端运行。</p> <ul style="list-style-type: none"> • UTST 位的设定步骤如下： <ol style="list-style-type: none"> 1. 进行上电复位。 2. 启动时钟（在晶体振荡和 USB PLL 稳定后将 SCKE 位置“1”） 3. 将 DCFM 位置“1”并且将 DRPD 位置“1”（不需要将 HSE 位置“1”）。 4. 将 USBE 位置“1”。 5. 给 UTST 位设定符合测试内容的值。 6. 将 UACT 位置“1”。 • UTST 位的变更步骤如下： <ol style="list-style-type: none"> 1.（在上述 6. 的状态下）将 UACT 位置“0”并且将 USBE 位置“0”。 2. 将 USBE 位置“1”。 3. 给 UTST 位设定符合测试内容的值。 4. 将 UACT 位置“1”。 <p>在设定 Test_SE0_NAK（1011）时，即使将 UACT 位置“1”，此模块也不输出 SOF 包。</p> <p>在设定 Test_Force_Enable（1101）时，通过将 UACT 位置“1”，此模块输出 SOF 包。在设定此模式时，即使此模块检测到高速断开（检测到 DTCH 中断），此模块也不进行随之发生的硬件控制。</p> <p>要设定 UTST 位时，必须将全部管道的 PID 位设定为 NAK。 要在设定测试模式后进行正常的 USB 通信时，必须进行上电复位。</p> <p>(2) 选择功能控制器功能的情况 必须根据高速通信时的 USB 主机发出的 SetFeature 请求写此位。 在将此位设定为“0001”~“0100”时，此模块不转移到挂起状态。</p>

表 29.6 测试模式运行表

测试模式	UTST 位的设定	
	选择功能控制器功能的情况	选择主机控制器功能的情况
正常运行	0000	0000
Test_J	0001	1001
Test_K	0010	1010
Test_SE0_NAK	0011	1011
Test_Packet	0100	1100
Test_Force_Enable	—	1101
Reserved	0101 ~ 0111	1110 ~ 1111

29.3.6 DMA-FIFO 总线配置寄存器 (D0FBCFG、D1FBCFG)

D0FBCFG 寄存器是控制 DMA0-FIFO 总线存取的寄存器，D1FBCFG 寄存器是控制 DMA1-FIFO 总线存取的寄存器。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	TENDE	—	—	—	—
初始值:	0	0	0	0	不定值	不定值	不定值	不定值	不定值	不定值	不定值	0	不定值	不定值	不定值	不定值
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 12	—	全 0	R	保留位 读写值都为“0”。
11 ~ 5	—	不定值	R	保留位 读取值为不定值，写入值总是“0”。
4	TENDE	0	R/W	DMA 传送结束采样允许 在 DMA 传送结束时，控制接收从直接存储器存取控制器输出的 DMA 传送结束信号，详细内容请参照“29.4.4(3) DMA 传送 (D0FIFO/D1FIFO 端口)”。 0: 不对 DMA 传送结束信号进行采样 1: 对 DMA 传送结束信号进行采样 如果将 DMA 的传送长度设定为 16 字节，就必须将此位置“0”。
3 ~ 0	—	不定值	R	保留位 读取值为不定值，写入值总是“0”。

29.3.7 FIFO 端口寄存器 (CFIFO、D0FIFO、D1FIFO)

CFIFO、D0FIFO 和 D1FIFO 是读写 FIFO 缓冲存储器数据的端口寄存器。

FIFO 端口有 CFIFO、D0FIFO 和 D1FIFO 共 3 个端口。各 FIFO 端口由读写 FIFO 缓冲存储器数据的 FIFO 端口寄存器 (CFIFO、D0FIFO 和 D1FIFO)、选择 FIFO 端口分配管道的选择寄存器 (CFIFOSEL、D0FIFOSEL 和 D1FIFOSEL) 以及控制寄存器 (CFIFOCTR、D0FIFOCTR 和 D1FIFOCTR) 构成。

各 FIFO 端口有以下特点:

- 必须通过 CFIFO 端口存取 DCP 的 FIFO 缓冲器。
- 必须通过 D0FIFO 端口或者 D1FIFO 端口存取 DMA 传送的 FIFO 缓冲器。
- 也能通过 CPU 存取 D1FIFO 端口或者 D0FIFO 端口。
- 在使用 FIFO 端口的特有功能时, 不能更改 CURPIPE 位设定的管道号 (所选管道) (使用 DMA 传送功能的情况等)。
- 构成 FIFO 端口的寄存器群不影响其他 FIFO 端口。
- 不能将相同的管道分配给多个 FIFO 端口。
- FIFO 缓冲器状态有 2 种: CPU 侧拥有存取权和 SIE 侧拥有存取权。当 SIE 侧拥有 FIFO 缓冲器的存取权时, CPU 就不能存取。

在上电复位时, 对此寄存器进行初始化。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FIFOPORT[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FIFOPORT[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 0	FIFOPORT [31:0]	全 0	R/W	FIFO 端口 通过存取这些位, 读 FIFO 缓冲器的接收数据或者将发送数据写到 FIFO 缓冲器。 只在各控制寄存器 (CFIFOCTR、D0FIFOCTR 或者 D1FIFOCTR) 的 FRDY 位为 “1” 时才能存取此寄存器。 此寄存器的有效位因 MBW 位的设定值和 BIGEND 位的设定值而不同, 有效位如表 29.7 ~ 表 29.9 所示。

表 29.7 进行 32 位存取时的字节序操作表

BIGEND 位	bit31 ~ 24	bit23 ~ 16	bit15 ~ 8	bit7 ~ 0
0	地址 N+3	地址 N+2	地址 N+1	地址 N+0
1	地址 N+0	地址 N+1	地址 N+2	地址 N+3

表 29.8 进行 16 位存取时的字节序操作表

BIGEND 位	bit31 ~ 24	bit23 ~ 16	bit15 ~ 8	bit7 ~ 0
0	写: 无效, 读: 禁止 *		地址 N+1	地址 N+0
1	地址 N+0	地址 N+1	写: 无效, 读: 禁止 *	

【注】 * 禁止以字或者字节为单位读无效寄存器。

表 29.9 进行 8 位存取时的字节序操作表

BIGEND 位	bit31 ~ 24	bit23 ~ 16	bit15 ~ 8	bit7 ~ 0
0	写: 无效, 读: 禁止 *			地址 N+0
1	地址 N+0	写: 无效, 读: 禁止 *		

【注】 * 禁止以字或者字节为单位读无效寄存器。

29.3.8 FIFO 端口选择寄存器 (CFIFOSEL、D0FIFOSEL、D1FIFOSEL)

CFIFOSEL、D0FIFOSEL 和 D1FIFOSEL 选择分配给 FIFO 端口的管道以及控制各 FIFO 端口的存取。

不能给 CFIFOSEL、D0FIFOSEL 和 D1FIFOSEL 的 CURPIPE 位指定相同的管道。当 D0FIFOSEL 寄存器和 D1FIFOSEL 寄存器的 CURPIPE 位的设定值为 “B'000” 时，不指定管道。

不能在 DMA 传送允许状态下更改管道号。

在上电复位时，对此寄存器进行初始化。

(1) CFIFOSEL

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCNT	REW	—	—	MBW[1:0]		—	BIGEND	—	—	ISEL	—	CURPIPE[3:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W*	R	R	R/W	R/W	R	R/W	R	R	R/W	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	RCNT	0	R/W	读计数模式 指定 CFIFOCTR 寄存器的 DTLN 位的读模式。 0: 在读完 CFIFO 的全部接收数据时，将 DTLN 位清 “0”。 (在双缓冲器的情况下，只读完 1 个缓冲器面时) 1: 每读 1 个 CFIFO 接收数据时，对 DTLN 位进行递减计数。
14	REW	0	R/W*	缓冲器指针复位 指定是否对缓冲器指针进行复位。 0: 不对缓冲器指针进行复位 1: 对缓冲器指针进行复位 当所选管道为接收方向时，如果在读 FIFO 缓冲器的过程中将此位置 “1”，就能从 FIFO 缓冲器的最初数据开始读 (在双缓冲器的情况下，能从正在读的 1 个缓冲器面的最初位置开始重新读数据)。 不能在将 REW 位置 “1” 的同时更改 CURPIPE 位的设定。必须在确认 FRDY 位为 “1” 后将 REW 位置 “1”。 对于发送方向的管道，如果要从 FIFO 缓冲器的最初位置重新开始写数据，就必须使用 BCLR 位。
13、12	—	全 0	R	保留位 读写值都为 “0”。
11、10	MBW[1:0]	00	R/W	CFIFO 端口存取位宽度 指定 CFIFO 端口存取位的宽度。 00: 8 位 01: 16 位 10: 32 位 11: 禁止设定 一旦开始缓冲存储器的读处理，就不能在读完全部数据前更改 FIFO 端口存取位的宽度。当指定管道为接收方向时，必须按以下的步骤设定此位： <ul style="list-style-type: none"> 在设定 CURPIPE 位的同时写这些位。 在设定 DCP (CURPIPE 位为 “B'000”) 时，在设定 ISEL 位的同时写这些位。 详细内容请参照 “29.4.4 FIFO 缓冲存储器”。 当所选管道为发送方向时，不能在缓冲存储器的写处理过程中从 8 位切换到 16 位或者 32 位，也不能从 16 位切换到 32 位。

位	位名	初始值	R/W	说明
9	—	0	R	保留位 读写值都为“0”。
8	BIGEND	0	R/W	CFIFO 端口字节序控制 指定 CFIFO 端口的字节序。 0: 小端法 1: 大端法
7、6	—	全 0	R	保留位 读写值都为“0”。
5	ISEL	0	R/W	选择 DCP 时的 CFIFO 端口存取方向 0: 选择读缓冲存储器 1: 选择写缓冲存储器 如果在所选管道为 DCP 时更改此位, 就必须在写此位后进行读操作, 在确认写入值和读取值相同后进行下一个处理。 如果在存取 FIFO 缓冲器的过程中更改此位的设定, 就保持以前的存取, 并且能在回写此位的设定后继续进行存取。 必须在设定 CURPIPE 位的同时设定此位。
4	—	0	R	保留位 读写值都为“0”。
3 ~ 0	CURPIPE [3:0]	0000	R/W	CFIFO 端口存取管道指定 必须通过 CFIFO 端口给要读写的数据设定管道号。 0000: DCP 0001: 管道 1 0010: 管道 2 0011: 管道 3 0100: 管道 4 0101: 管道 5 0110: 管道 6 0111: 管道 7 1000: 管道 8 1001: 管道 9 上述以外: 禁止设定 如果要更改这些位, 就必须在写这些位后进行读操作, 在确认写入值和读取值相同后进行下一个处理。 不能给 CFIFOSEL 寄存器、D0FIFOSEL 寄存器和 D1FIFOSEL 寄存器的 CURPIPE 位设定相同的管道号。 如果在存取 FIFO 缓冲器的过程中更改这些位的设定, 就保持以前的存取, 并且能在回写这些位的设定后继续进行存取。

【注】 * 读取值为“0”，只能写“1”。

(2) D0FIFOSEL、D1FIFOSEL

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCNT	REW	DCLRM	DREQE	MBW[1:0]		—	BIG END	—	—	—	—	CURPIPE[3:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W*	R/W	R/W	R/W	R/W	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	RCNT	0	R/W	<p>读计数模式</p> <p>指定 DnFIFOCTR 寄存器的 DTLN 位的读模式。</p> <p>0: 在读完 DnFIFO 的全部接收数据时, 将 DTLN 位清 “0” (在双缓冲器的情况下, 只读完 1 个缓冲器面的数据时)。</p> <p>1: 每读 1 个 DnFIFO 接收数据时, 对 DTLN 位进行递减计数。</p> <p>如果要在将 BFRE 位置 “1” 后存取 DnFIFO, 就必须将此位置 “0”。</p>
14	REW	0	R/W*	<p>缓冲器指针复位</p> <p>指定是否对缓冲器指针进行复位。</p> <p>0: 不对缓冲器指针进行复位</p> <p>1: 对缓冲器指针进行复位</p> <p>当所选管道为接收方向时, 如果在读 FIFO 缓冲器的过程中将此位置 “1”, 就能从 FIFO 缓冲器的最初数据开始读 (在双缓冲器的情况下, 能从正在读的 1 个缓冲器面的最初位置开始重新读数据)。</p> <p>不能在将 REW 位置 “1” 的同时更改 CURPIPE 位的设定。必须在确认 FRDY 位为 “1” 后将 REW 位置 “1”。</p> <p>对于发送方向的管道, 如果要从 FIFO 缓冲器的最初位置开始重新写数据, 就必须使用 BCLR 位。</p>
13	DCLRM	0	R/W	<p>读所选管道数据后的缓冲存储器自动清除模式</p> <p>在读所选管道的数据后, 指定禁止或者允许缓冲存储器自动清除。</p> <p>0: 禁止缓冲器自动清除模式</p> <p>1: 允许缓冲器自动清除模式</p> <p>当此位为 “1” 时, 如果在分配给所选管道的 FIFO 缓冲器为空的状态下接收 Zero-Length packet 或者在 BFRE 位为 “1” 的状态下接收短包并且读完数据, 此模块就将 FIFO 缓冲器的 BCLR 位置 “1”。</p> <p>要在将 BRDYM 位置 “1” 后使用此模块时, 必须将此位置 “0”。</p>
12	DREQE	0	R/W	<p>DMA 传送请求允许</p> <p>指定禁止或者允许 DMA 传送请求的发行。</p> <p>0: 禁止 DMA 传送请求</p> <p>1: 允许 DMA 传送请求</p> <p>在允许发行 DMA 传送请求时, 必须在设定 CURPIPE 位后将此位置 “1”。</p> <p>必须在将此位置 “0” 后更改 CURPIPE 位的设定。</p>

位	位名	初始值	R/W	说明
11、10	MBW[1:0]	00	R/W	FIFO 端口存取位宽度 指定 DnFIFO 端口存取位的宽度。 00: 8 位 01: 16 位 10: 32 位 11: 禁止设定 一旦开始缓冲存储器的读处理, 就不能在读完全部数据前更改 FIFO 端口存取位的宽度。 当所选管道为接收方向时, 必须同时设定 CURPIPE 位和 MBW 位, 详细内容请参照“29.4.4 FIFO 缓冲存储器”。 当指定管道为发送方向时, 不能在缓冲存储器的写处理过程中从 8 位切换到 16 位或者 32 位, 也不能从 16 位切换到 32 位。
9	—	0	R	保留位 读写值都为“0”。
8	BIGEND	0	R/W	FIFO 端口字节序控制 指定 DnFIFO 端口的字节序。 0: 小端法 1: 大端法
7~4	—	全 0	R	保留位 读写值都为“0”。
3~0	CURPIPE [3:0]	0000	R/W	FIFO 端口存取管道指定 必须通过 D0FIFO/D1FIFO 端口给要读写的数据设定管道号。 0000: 未指定 0001: 管道 1 0010: 管道 2 0011: 管道 3 0100: 管道 4 0101: 管道 5 0110: 管道 6 0111: 管道 7 1000: 管道 8 1001: 管道 9 上述以外: 禁止设定 如果要更改这些位, 就必须在写这些位后进行读操作, 在确认写入值和读取值相同后进行下一个处理。 不能给 CFIFOSEL 寄存器、D0FIFOSEL 寄存器和 D1FIFOSEL 寄存器的 CURPIPE 位设定相同的管道号。 如果在存取 FIFO 缓冲器的过程中更改此位的设定, 就保持以前的存取, 并且能在回写这些位的设定后继续进行存取。

【注】 * 读取值为“0”, 只能写“1”。

29.3.9 FIFO 端口控制寄存器 (CFIFOCTR、D0FIFOCTR、D1FIFOCTR)

CFIFOCTR、D0FIFOCTR 和 D1FIFOCTR 设定缓冲存储器的写操作结束、CPU 侧缓冲器的清除以及是否能存取 FIFO 端口。此寄存器对应各 FIFO 端口。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BVAL	BCLR	FRDY	—	DTLN[11:0]											
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*2	R/W*1	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15	BVAL	0	R/W*2	<p>缓冲存储器有效标志</p> <p>在写完 CURPIPE 指定管道（所选管道）的 CPU 侧的 FIFO 缓冲器时，将此位置“1”。</p> <p>0: 无效</p> <p>1: 写操作结束</p> <p>当所选管道为发送方向时，必须在以下情况下将此位置“1”。此模块将 CPU 侧的 FIFO 缓冲器置为 SIE 侧，并且设定为能发送的状态。</p> <ul style="list-style-type: none"> • 要发送短包时，在写完数据时将此位置“1”。 • 要发送 Zero-Length 包时，在将数据写到 FIFO 缓冲器前将此位置“1”。 • 在给连续传送模式的管道写 MaxPacketSize 的自然数倍并且小于 BufferSize 的数据后将此位置“1”。 <p>如果给非连续传送模式的管道写 MaxPacketSize 的数据，此模块就将此位置“1”，将 CPU 侧的 FIFO 缓冲器置为 SIE 侧，并且设定为能发送的状态。当所选管道为发送方向时，如果同时给 BVAL 位和 BCLR 位写“1”，此模块就清除以前写的的数据，并且将 Zero-Length 包设定为能发送的状态。必须在此模块将 FRDY 位置“1”时给此位写“1”。</p> <p>当所选管道为接收方向时，不能给此位写“1”。</p>
14	BCLR	0	R/W*1	<p>CPU 缓冲器清除</p> <p>要清除所选管道的 CPU 侧的 FIFO 缓冲器时，将此位置“1”。</p> <p>0: 无效</p> <p>1: 清除 CPU 侧的缓冲存储器</p> <p>当分配给所选管道的 FIFO 缓冲器为双缓冲器时，即使为能读 2 个缓冲器面的状态，此模块也只清除 1 个 FIFO 缓冲器面。</p> <p>当所选管道为 DCP 时，与 FIFO 缓冲器是 CPU 侧还是 SIE 侧无关，通过将 BCLR 位置“1”，此模块清除 FIFO 缓冲器。要清除 SIE 侧的缓冲器时，必须在将 DCP 的 PID 位设定为 NAK 后将 BCLR 位置“1”。</p> <p>在所选管道不是 DCP 时，必须在此模块将 FRDY 位置“1”时给此位写“1”。</p>
13	FRDY	0	R	<p>FIFO 端口就绪</p> <p>表示是否能存取 FIFO 端口。</p> <p>0: 不能存取 FIFO 端口</p> <p>1: 能存取 FIFO 端口</p> <p>在以下情况下，此模块将 FRDY 位置“1”，因为没有应该读的数据，所以不能读 FIFO 端口的数据。在这些情况下，必须通过将 BCLR 位置“1”，清除 FIFO 缓冲器，并且设定为能发送和接收下一个数据的状态。</p> <ul style="list-style-type: none"> • 在分配给所选管道的 FIFO 缓冲器为空的状态下接收到 Zero-Length 包。 • 在 BFRE 位为“1”时接收到短包并且读完数据。

位	位名	初始值	R/W	说明
12	—	0	R	保留位 读写值都为“0”。
11 ~ 0	DTLN [11:0]	H'000	R	接收数据长度 表示接收数据的长度。 在读 FIFO 缓冲器的过程中，这些位的值因 RCNT 位的设定值而不同。 • RCNT 位为“0”的情况 在读完 1 个 FIFO 缓冲器面的接收数据前，此模块给这些位设定接收数据的长度。 当 BFRE 位为“1”时，即使已读完数据，此模块也将接收数据的长度保持到将 BCLR 位置“1”为止。 • RCNT 位为“1”的情况 在每次读时，此模块对 DTLN 位进行递减计数（MBW 位为“00”时减 1，MBW 位为“01”时减 2，MBW 位为“10”时减 4）。 在读完 1 个 FIFO 缓冲器面时，此模块将 DTLN 位置“0”。如果在设定为双缓冲器并且读完 1 个 FIFO 缓冲器面的接收数据前接收到另外 1 个 FIFO 缓冲器面的数据，就在读完前 1 个缓冲器面的数据时给 DTLN 位设定后 1 个缓冲器面的接收数据长度。 【注】 在说明 RCNT 位为“1”的情况时，从读 FIFO 端口到更新这些位，需要 10 个总线周期。

【注】 *1 读取值为“0”，只能写“1”。

*2 只能写“1”。

29.3.10 中断允许寄存器 0 (INTENB0)

INTENB0 指定是否允许各中断。如果此模块检测到此寄存器中被置“1”的位所对应的中断，就产生 USB 中断。

与此寄存器的设定值（禁止或者允许中断通知）无关，在满足各中断源的检测条件时，此模块将 INTSTS0 寄存器对应的状态位置“1”。

在各中断源对应的 INTSTS0 寄存器的状态位为“1”的情况下，如果将此寄存器对应的中断允许位从“0”改为“1”，此模块就产生 USB 中断。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VBSE	RSME	SOFE	DVSE	CTRE	BEMPE	NRDYE	BRDYE	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15	VBSE	0	R/W	VBUS 中断允许 在检测到 VBINT 中断时，指定禁止或者允许 USB 中断请求。 0: 禁止中断输出 1: 允许中断输出
14	RSME	0	R/W	恢复中断允许 * 在检测到 RESM 中断时，指定禁止或者允许 USB 中断请求。 0: 禁止中断输出 1: 允许中断输出

位	位名	初始值	R/W	说明
13	SOFE	0	R/W	帧号更新中断允许 在检测到 SOFR 中断时，指定禁止或者允许 USB 中断请求。 0: 禁止中断输出 1: 允许中断输出
12	DVSE	0	R/W	设备状态转移中断允许 * 在检测到 DVST 中断时，指定禁止或者允许 USB 中断请求。 0: 禁止中断输出 1: 允许中断输出
11	CTRE	0	R/W	控制传送阶段转移中断允许 * 在检测到 CTRT 中断时，指定禁止或者允许 USB 中断请求。 0: 禁止中断输出 1: 允许中断输出
10	BEMPE	0	R/W	缓冲器空中断允许 在检测到 BEMP 中断时，指定禁止或者允许 USB 中断请求。 0: 禁止中断输出 1: 允许中断输出
9	NRDYE	0	R/W	缓冲器未就绪应答中断允许 在检测到 NRDY 中断时，指定禁止或者允许 USB 中断请求。 0: 禁止中断输出 1: 允许中断输出
8	BRDYE	0	R/W	缓冲器就绪中断允许 在检测到 BRDY 中断时，指定禁止或者允许 USB 中断请求。 0: 禁止中断输出 1: 允许中断输出
7 ~ 0	—	全 0	R	保留位 读写值都为“0”。

【注】 * 只在选择功能控制器功能时才能设定 RSME 位、DVSE 位和 CTRE 位。在选择主机控制器功能时不能设定为允许。

29.3.11 中断允许寄存器 1 (INTENB1)

INTENB1 设定在选择主机控制器功能时是否允许中断。

如果此模块检测到此寄存器中被置“1”的位所对应的中断，就产生 USB 中断。

与此寄存器的设定值（禁止或者允许中断通知）无关，在满足各中断源的检测条件时，此模块将 INTSTS1 寄存器对应的状态位置“1”。

在各中断源对应的 INTSTS1 寄存器的状态位为“1”的情况下，如果将此寄存器对应的中断允许位从“0”改为“1”，此模块就产生 USB 中断。

当选择功能控制器功能时，不能设定为允许中断。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	BCHGE	—	DTCHE	ATT CHE	—	—	—	—	EOF ERRE	SIGNE	SACKE	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R	R	R	R

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14	BCHGE	0	R/W	USB 总线变化中断允许 在检测到 BCHG 中断时，指定禁止或者允许 USB 中断请求。 0: 禁止中断输出 1: 允许中断输出
13	—	0	R	保留位 读写值都为“0”。
12	DTCHE	0	R/W	断开检测中断允许 在检测到 DTCH 中断时，指定禁止或者允许 USB 中断请求。 0: 禁止中断输出 1: 允许中断输出
11	ATTCHE	0	R/W	连接检测中断允许 在检测到 ATTCH 中断时，指定禁止或者允许 USB 中断请求。 0: 禁止中断输出 1: 允许中断输出
10 ~ 7	—	全 0	R	保留位 读写值都为“0”。
6	EOFERRE	0	R/W	EOF 错误检测中断允许 在检测到 EOFERR 中断时，指定禁止或者允许 USB 中断请求。 0: 禁止中断输出 1: 允许中断输出
5	SIGNE	0	R/W	设置事务错误中断允许 在检测到 SIGN 中断时，指定禁止或者允许 USB 中断请求。 0: 禁止中断输出 1: 允许中断输出
4	SACKE	0	R/W	设置事务正常应答中断允许 在检测到 SACK 中断时，指定禁止或者允许 USB 中断请求。 0: 禁止中断输出 1: 允许中断输出
3 ~ 0	—	全 0	R	保留位 读写值都为“0”。

【注】 只在选择主机控制器功能时才能设定为通过 INTENB1 寄存器允许中断。在选择功能控制器功能时不能设定为允许。

29.3.12 BRDY 中断允许寄存器 (BRDYENB)

在检测到各管道的 BRDY 中断时，BRDYENB 指定禁止或者允许将 INTSTS0 寄存器的 BRDY 位置“1”。

对于此寄存器中被置“1”的管道，如果此模块检测到 BRDY 中断，就将 BRDYSTS 寄存器的 PIPEBRDY 位的对应位置“1”，并且将 INTSTS0 寄存器的 BRDY 位置“1”，产生 BRDY 中断。

在 BRDYSTS 寄存器的 PIPEBRDY 位中至少有 1 位为“1”的状态下，如果将此寄存器对应的中断允许位从“0”改为“1”，此模块就产生 BRDY 中断。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPE9 BRDYE	PIPE8 BRDYE	PIPE7 BRDYE	PIPE6 BRDYE	PIPE5 BRDYE	PIPE4 BRDYE	PIPE3 BRDYE	PIPE2 BRDYE	PIPE1 BRDYE	PIPE0 BRDYE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9	PIPE9BRDYE	0	R/W	管道 9 的 BRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
8	PIPE8BRDYE	0	R/W	管道 8 的 BRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
7	PIPE7BRDYE	0	R/W	管道 7 的 BRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
6	PIPE6BRDYE	0	R/W	管道 6 的 BRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
5	PIPE5BRDYE	0	R/W	管道 5 的 BRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
4	PIPE4BRDYE	0	R/W	管道 4 的 BRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
3	PIPE3BRDYE	0	R/W	管道 3 的 BRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
2	PIPE2BRDYE	0	R/W	管道 2 的 BRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
1	PIPE1BRDYE	0	R/W	管道 1 的 BRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
0	PIPE0BRDYE	0	R/W	管道 0 的 BRDY 中断允许 0: 禁止中断输出 1: 允许中断输出

29.3.13 NRDY 中断允许寄存器 (NRDYENB)

在检测到各管道的 NRDY 中断时，NRDYENB 指定禁止或者允许将 INTSTS0 寄存器的 NRDY 位置“1”。

对于此寄存器中被置“1”的管道，如果此模块检测到 NRDY 中断源，就将 NRDYSTS 寄存器的 PIPENRDY 位的对应位置“1”，并且将 INTSTS0 寄存器的 NRDY 位置“1”，产生 NRDY 中断。

在 NRDYSTS 寄存器的 PIPENRDY 位中至少有 1 位为“1”的状态下，如果将此寄存器对应的中断允许位从“0”改为“1”，此模块就产生 NRDY 中断。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPE9 NRDYE	PIPE8 NRDYE	PIPE7 NRDYE	PIPE6 NRDYE	PIPE5 NRDYE	PIPE4 NRDYE	PIPE3 NRDYE	PIPE2 NRDYE	PIPE1 NRDYE	PIPE0 NRDYE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9	PIPE9NRDYE	0	R/W	管道 9 的 NRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
8	PIPE8NRDYE	0	R/W	管道 8 的 NRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
7	PIPE7NRDYE	0	R/W	管道 7 的 NRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
6	PIPE6NRDYE	0	R/W	管道 6 的 NRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
5	PIPE5NRDYE	0	R/W	管道 5 的 NRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
4	PIPE4NRDYE	0	R/W	管道 4 的 NRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
3	PIPE3NRDYE	0	R/W	管道 3 的 NRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
2	PIPE2NRDYE	0	R/W	管道 2 的 NRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
1	PIPE1NRDYE	0	R/W	管道 1 的 NRDY 中断允许 0: 禁止中断输出 1: 允许中断输出
0	PIPE0NRDYE	0	R/W	管道 0 的 NRDY 中断允许 0: 禁止中断输出 1: 允许中断输出

29.3.14 BEMP 中断允许寄存器 (BEMPENB)

在检测到各管道的 BEMP 中断时，BEMPENB 指定禁止或者允许将 INTSTS0 寄存器的 BEMP 位置“1”。

对于此寄存器中被置“1”的管道，如果此模块检测到 BEMP 中断源，此模块就将 BEMPSTS 寄存器的 PIPEBEMP 位的对应位置“1”，并且将 INTSTS0 寄存器的 BEMP 位置“1”，产生 BEMP 中断。

在 BEMPSTS 寄存器的 PIPEBEMP 位中至少有 1 位为“1”的状态下，如果将此寄存器对应的中断允许位从“0”改为“1”，此模块就产生 BEMP 中断。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPE9 BEMPE	PIPE8 BEMPE	PIPE7 BEMPE	PIPE6 BEMPE	PIPE5 BEMPE	PIPE4 BEMPE	PIPE3 BEMPE	PIPE2 BEMPE	PIPE1 BEMPE	PIPE0 BEMPE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9	PIPE9BEMPE	0	R/W	管道 9 的 BEMP 中断允许 0: 禁止中断输出 1: 允许中断输出
8	PIPE8BEMPE	0	R/W	管道 8 的 BEMP 中断允许 0: 禁止中断输出 1: 允许中断输出
7	PIPE7BEMPE	0	R/W	管道 7 的 BEMP 中断允许 0: 禁止中断输出 1: 允许中断输出
6	PIPE6BEMPE	0	R/W	管道 6 的 BEMP 中断允许 0: 禁止中断输出 1: 允许中断输出
5	PIPE5BEMPE	0	R/W	管道 5 的 BEMP 中断允许 0: 禁止中断输出 1: 允许中断输出
4	PIPE4BEMPE	0	R/W	管道 4 的 BEMP 中断允许 0: 禁止中断输出 1: 允许中断输出
3	PIPE3BEMPE	0	R/W	管道 3 的 BEMP 中断允许 0: 禁止中断输出 1: 允许中断输出
2	PIPE2BEMPE	0	R/W	管道 2 的 BEMP 中断允许 0: 禁止中断输出 1: 允许中断输出
1	PIPE1BEMPE	0	R/W	管道 1 的 BEMP 中断允许 0: 禁止中断输出 1: 允许中断输出
0	PIPE0BEMPE	0	R/W	管道 0 的 BEMP 中断允许 0: 禁止中断输出 1: 允许中断输出

29.3.15 SOF 输出配置寄存器 (SOFCFG)

SOFCFG 指定事务有效期间和 BRDY 中断状态清除时序等。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TRNENSEL	—	BRDYM	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R/W	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	TRNENSEL	0	R/W	事务有效期间转换位 在全速通信或者低速通信过程中，指定此模块在 1 帧内发行权标的期间（事务有效期间）。 0: 不支持低速运行 1: 支持低速运行 此位只在选择主机控制器功能时有效。即使在选择主机控制器功能时，也不影响高速事务的有效期间。 当选择功能控制器功能时，必须将此位置“0”。
7	—	0	R	保留位 读写值都为“0”。
6	BRDYM	0	R/W	各管道 BRDY 中断状态清除时序设定 指定各管道 BRDY 中断状态的清除时序。 必须在此模块的初始设定时（通信前）设定此位。不能更改通信后的设定。 0: 通过写“0”，清除状态。 1: 通过读写 FIFO 缓冲器，此模块自动清除状态。
5 ~ 0	—	全 0	R	保留位 读写值都为“0”。

29.3.16 中断状态寄存器 0 (INTSTS0)

INTSTS0 表示被检测到的各中断状态。

在上电复位时，对此寄存器进行初始化；在 USB 总线复位时，对 PVST 位和 DVSQ[2:0] 位进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VBINT	RESM	SOFR	DVST	CTR1	BEMP	NRDY	BRDY	VBSTS	DVSQ[2:0]			VALID	CTSQ[2:0]		
初始值:	0	0	0	0/1*1	0	0	0	0	0/1*3	0*2	0*2	0/1*2	0	0	0	0
R/W:	R/W*7	R/W*7	R/W*7	R/W*7	R/W*7	R	R	R	R	R	R	R	R/W*7	R	R	R

位	位名	初始值	R/W	说明
15	VBINT	0	R/W*7	VBUS 中断状态 *4*5 0: 未发生 VBUS 中断 1: 发生 VBUS 中断 此模块在检测到 VBUS 引脚输入值的变化（从高电平变为低电平或者从低电平变为高电平）时将此位置“1”，并且将 VBUS 引脚的输入值反映到 VBSTS 位。在发生 VBINT 中断时，必须重复读 VBSTS 位，直到读取值数次相同为止，以消除颤动。
14	RESM	0	R/W*7	恢复中断状态 *4*5*6 0: 未发生恢复中断 1: 发生恢复中断 当设定功能控制器功能时，如果此模块为挂起状态（DVSQ 位为“1XX”）并且检测到 DP 引脚的下降沿，就将此位置“1”。 当选择主机控制器功能时，读取值无效。
13	SOFR	0	R/W*7	帧号更新中断状态 *4 0: 未发生 SOF 中断 1: 发生 SOF 中断 (1) 设定主机控制器功能的情况 在将 UACT 位置“1”的状态下更新帧号时，将此位置“1”（每 1ms 检测 1 次帧号更新中断）。 (2) 设定功能控制器功能的情况 此模块在更新帧号更新时将此位置“1”（每 1ms 检测 1 次帧号更新中断）。即使 USB 主机的 SOF 包发生破损，此模块也能通过内插检测到 SOFR 中断。
12	DVST	0/1*1	R/W*7	设备状态转移中断状态 *4*6 0: 未发生设备状态转移中断 1: 发生设备状态转移中断 当设定功能控制器功能时，如果此模块检测到设备状态的变化，就更新 DVSQ 位的值并且将此位置“1”。 当发生设备状态转移中断时，必须在此模块检测到下一个设备状态转移前清除状态。 当选择主机控制器功能时，读取值无效。

位	位名	初始值	R/W	说明
11	CTRT	0	R/W*7	<p>控制传送阶段转移中断状态 *4*6</p> <p>0: 未发生控制传送阶段转移中断</p> <p>1: 发生控制传送阶段转移中断</p> <p>当设定功能控制器功能时, 如果此模块检测到控制传送的阶段转移, 就更新 CTSQ 位的值并且将此位置“1”。</p> <p>当发生控制传送阶段转移中断时, 必须在此模块检测到控制传送的下一个阶段转移前清除状态。</p> <p>当选择主机控制器功能时, 读取值无效。</p>
10	BEMP	0	R	<p>缓冲器空中断状态</p> <p>0: 未发生 BEMP 中断</p> <p>1: 发生 BEMP 中断</p> <p>当 BEMPENB 寄存器的 PIPEBEMPE 位为“1”的管道对应的 BEMPSTS 寄存器的 PIPEBEMP 位中至少有 1 位为“1”时(此模块至少对允许 BEMP 中断通知的管道中的 1 个管道, 检测到 BEMP 中断状态时), 此模块将此位置“1”。</p> <p>PIPEBEMP 状态的有效条件请参照“29.4.2(3) BEMP 中断”。</p> <p>如果给由 PIPEBEMPE 位设定为允许的管道对应的 PIPEBEMP 位全部写“0”, 此模块就将此位清“0”。</p> <p>即使给此位写“0”, 也无法将此位清“0”。</p>
9	NRDY	0	R	<p>缓冲器未就绪中断状态</p> <p>0: 未发生 NRDY 中断</p> <p>1: 发生 NRDY 中断</p> <p>当 NRDYENB 寄存器的 PIPENRDYE 位为“1”的管道对应的 NRDYSTS 寄存器的 PIPENRDY 位中至少有 1 位为“1”时(此模块至少对允许 NRDY 中断通知的管道中的 1 个管道, 检测到 NRDY 中断状态时), 此模块将此位置“1”。</p> <p>PIPENRDY 状态的有效条件请参照“29.4.2(2) NRDY 中断”。</p> <p>如果给由 PIPENRDYE 位设定为允许的管道对应的 PIPENRDY 位全部写“0”, 此模块就将此位清“0”。</p> <p>即使给此位写“0”, 也无法将此位清“0”。</p>
8	BRDY	0	R	<p>缓冲器就绪中断状态</p> <p>表示 BRDY 中断状态。</p> <p>0: 未发生 BRDY 中断</p> <p>1: 发生 BRDY 中断</p> <p>当 BRDYENB 寄存器的 PIPEBRDYE 位为“1”的管道对应的 BRDYSTS 寄存器的 PIPEBRDY 位中至少有 1 位为“1”时(此模块至少对允许 BRDY 中断通知的管道中的 1 个管道, 检测到 BRDY 中断状态时), 此模块将此位置“1”。</p> <p>PIPEBRDY 状态的有效条件请参照“29.4.2(1) BRDY 中断”。</p> <p>如果给由 PIPEBRDYE 位设定为允许的管道对应的 PIPEBRDY 位全部写“0”, 此模块就将此位清“0”。</p> <p>即使给此位写“0”, 也无法将此位清“0”。</p>
7	VBSTS	0/1*3	R	<p>VBUS 输入状态</p> <p>0: VBUS 引脚为低电平</p> <p>1: VBUS 引脚为高电平</p>

位	位名	初始值	R/W	说明
6 ~ 4	DVSQ[2:0]	000/001*2	R	设备状态 000: 供电状态 001: 默认状态 010: 地址状态 011: 配置状态 1xx: 挂起状态 当选择主机控制器功能时, 读取值无效。
3	VALID	0	R/W*7	USB 请求接收 0: 未检测到 1: 接收到设置包 当选择主机控制器功能时, 读取值无效。
2 ~ 0	CTSQ[2:0]	000	R	控制传送阶段 000: 空闲阶段或者设置阶段 001: 控制读数据阶段 010: 控制读状态阶段 011: 控制写数据阶段 100: 控制写状态阶段 101: 控制写 (NoData) 状态阶段 110: 控制传送顺序错误 111: 禁止设定 当选择主机控制器功能时, 读取值无效。

- 【注】 *1 在上电复位时, 此位为 “B'0” ; 在 USB 总线复位时, 此位为 “B'1”。
- *2 在上电复位时, 此位为 “B'000” ; 在 USB 总线复位时, 此位为 “B'001”。
- *3 在 VBUS 引脚为高电平时, 此位为 “1” ; 在 VBUS 引脚为低电平时, 此位为 “0”。
- *4 要清除 VBINT 位、RESM 位、SOFR 位、DVST 位或者 CTRT 位时, 只能给要清除的位写 “0” 而给其他位写 “1”。不能给为 “0” 的状态位写 “0”。
- *5 此模块也能在时钟停止过程中 (SCKE 位为 “0”) 检测到 VBINT 位和 RESM 位所示的状态变化, 如果允许对应的中断, 就进行中断通知。必须在允许时钟后清除状态。
- *6 只在选择功能控制器功能时, RESM 位、DVST 位和 CTRT 位才发生状态变化。在选择主机控制器功能时, 必须将对应的中断允许位置 “0” (禁止)。
- *7 只能写 “0”。

29.3.17 中断状态寄存器 1 (INTSTS1)

INTSTS1 是确认各中断状态的寄存器。

只在选择主机控制器功能时，才能允许此寄存器各位所示的状态变化引起的中断。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	BCHG	—	DTCH	ATTCH	—	—	—	—	EOF ERR	SIGN	SACK	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W*1	R	R/W*1	R/W*1	R	R	R	R	R/W*1	R/W*1	R/W*1	R	R	R	R

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14	BCHG	0	R/W*1	USB 总线变化中断状态 表示 USB 总线变化中断状态。 0: 未发生 BCHG 中断 1: 发生 BCHG 中断 如果 USB 端口发生全速 / 低速信号电平的状态变化 (从 J-State、K-State 或者 SE0 的任意一种状态变为 J-State、K-State 或者 SE0 的任意一种状态)，此模块就检测到 BCHG 中断并且将此位置“1”。此时，如果已将对应的中断允许位置“1”，此模块就产生中断。 SYSSTS0 寄存器的 LNST 位表示 USB 端口当前的输入状态。在发生 BCHG 引脚中断时，重复读 LNST 位，直到读取值数次相同为止，以消除颤动。 在内部时钟停止状态下也能检测到 USB 总线的变化。 当选择功能控制器功能时，读取值无效。
13	—	0	R	保留位 读写值都为“0”。
12	DTCH	0	R/W*1	USB 断开检测中断状态 当选择主机控制器功能时，表示 USB 断开检测的中断状态。 0: 未发生 DTCH 中断 1: 发生 DTCH 中断 在检测到 USB 总线断开时，此模块检测到 DTCH 中断并且将此位置“1”。此时，如果已将对应的中断允许位置“1”，此模块就产生中断。 此模块按照 USB 2.0 规格的基准检测总线的断开。 此模块在检测到 DTCH 中断 (与对应的中断允许位的设定值无关) 后进行以下的硬件控制。必须通过软件使正在与 USB 端口进行通信的管道全部停止通信，并且转移到 USB 端口的连接 (产生 ATTCH 中断) 等待状态。 (1) 将 UACT 位改为“0”。 (2) 转移到空闲状态。 当选择功能控制功能时，读取值无效。

位	位名	初始值	R/W	说明
11	ATTCH	0	R/W*1	<p>ATTCH 中断状态</p> <p>当选择主机控制器功能时，表示 ATTCH 的中断状态。</p> <p>0: 未发生 ATTCH 中断 1: 发生 ATTCH 中断</p> <p>如果此模块检测到 2.5μs 的全速 / 低速信号电平的 J-State 或者 K-State，就检测到 ATTCH 中断并且将此位置“1”。此时，如果已将对应的中断允许位置“1”，此模块就产生中断。</p> <p>此模块的 ATTCH 中断检测的具体条件如下：</p> <ul style="list-style-type: none"> • 从 K-State、SE0 或者 SE1 变为 J-State 并且 J-State 持续保持 2.5μs 时 • 从 J-State、SE0 或者 SE1 变为 K-State 并且 K-State 持续保持 2.5μs 时 <p>当选择功能控制功能时，读取值无效。</p>
10 ~ 7	—	全 0	R	<p>保留位</p> <p>读写值都为“0”。</p>
6	EOFERR	0	R/W*1	<p>EOF 错误检测中断状态</p> <p>当选择主机控制器功能时，表示 EOFERR 的中断状态。</p> <p>0: 未发生 EOFERR 中断 1: 发生 EOFERR 中断</p> <p>如果此模块在 USB 2.0 规格规定的 EOF2 时检测到通信未结束，就检测到 EOFERR 中断并且将此位置“1”。此时，如果已将对应的中断允许位置“1”，此模块就产生 EOFERR 中断。</p> <p>此模块在检测到 EOFERR 中断（与对应的中断允许位的设定值无关）后进行以下的硬件控制。必须使正在与 USB 端口进行通信的管道全部停止通信，并且重新对 USB 端口进行 Enumeration。</p> <p>(1) 将 UACT 位改为“0” (2) 转移到空闲状态。</p> <p>当选择功能控制功能时，读取值无效。</p>
5	SIGN	0	R/W*1	<p>设置事务错误中断状态</p> <p>当选择主机控制器功能时，表示设置事务错误的中断状态。</p> <p>0: 未发生 SIGN 中断 1: 发生 SIGN 中断</p> <p>在此模块发行的设置事务中，如果外围设备连续 3 次没有 ACK 应答，此模块就检测到 SIGN 中断并且将此位置“1”。此时，如果已将对应的中断允许位置“1”，此模块就产生 SIGN 中断。</p> <p>此模块的 SIGN 中断检测条件是在对连续 3 次的设置事务进行以下任何一种应答时的情况：</p> <ul style="list-style-type: none"> • 此模块在外围设备没有任何应答的状态下检测到超时 • 当 ACK 包发生破损时 • 当接收到 ACK 以外的握手信号（NAK、NYET 或者 STALL）时 <p>当选择功能控制功能时，读取值无效。</p>
4	SACK	0	R/W*1	<p>设置事务正常应答中断状态</p> <p>当选择主机控制器功能时，表示设置事务正常应答的中断状态。</p> <p>0: 未发生 SACK 中断 1: 发生 SACK 中断</p> <p>在此模块发行的设置事务中，如果接收外围设备的 ACK 应答，此模块就检测到 SACK 中断并且将此位置“1”。此时，如果已将对应的中断允许位置“1”，此模块就产生 SACK 中断。</p>

位	位名	初始值	R/W	说明
3 ~ 0	—	全 0	R	保留位 读写值都为“0”。

【注】 *1 只能写“0”。

- *2 此模块也能在时钟停止时（SCKE 位为“0”）检测到 BCHG 位所示的状态变化，只要允许对应的中断也进行中断通知。必须在允许时钟后清除状态。
在时钟停止时（SCKE 位为“0”），不检测 BCHG 位以外的中断。

29.3.18 BRDY 中断状态寄存器 (BRDYSTS)

BRDYSTS 表示各管道的 BRDY 中断状态。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPE9 BRDY	PIPE8 BRDY	PIPE7 BRDY	PIPE6 BRDY	PIPE5 BRDY	PIPE4 BRDY	PIPE3 BRDY	PIPE2 BRDY	PIPE1 BRDY	PIPE0 BRDY
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

位	位名	初始值	R/W	说明
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9	PIPE9BRDY	0	R/W*1	管道 9 的 BRDY 中断状态 *2 0: 未发生中断 1: 发生中断
8	PIPE8BRDY	0	R/W*1	管道 8 的 BRDY 中断状态 *2 0: 未发生中断 1: 发生中断
7	PIPE7BRDY	0	R/W*1	管道 7 的 BRDY 中断状态 *2 0: 未发生中断 1: 发生中断
6	PIPE6BRDY	0	R/W*1	管道 6 的 BRDY 中断状态 *2 0: 未发生中断 1: 发生中断
5	PIPE5BRDY	0	R/W*1	管道 5 的 BRDY 中断状态 *2 0: 未发生中断 1: 发生中断
4	PIPE4BRDY	0	R/W*1	管道 4 的 BRDY 中断状态 *2 0: 未发生中断 1: 发生中断
3	PIPE3BRDY	0	R/W*1	管道 3 的 BRDY 中断状态 *2 0: 未发生中断 1: 发生中断
2	PIPE2BRDY	0	R/W*1	管道 2 的 BRDY 中断状态 *2 0: 未发生中断 1: 发生中断
1	PIPE1BRDY	0	R/W*1	管道 1 的 BRDY 中断状态 *2 0: 未发生中断 1: 发生中断
0	PIPE0BRDY	0	R/W*1	管道 0 的 BRDY 中断状态 *2 0: 未发生中断 1: 发生中断

【注】 *1 只能写“0”。

*2 在将 BRDYM 位置“0”时，必须在存取 FIFO 前清除此中断。

29.3.19 NRDY 中断状态寄存器 (NRDYSTS)

NRDYSTS 表示各管道的 NRDY 中断状态。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPE9 NRDY	PIPE8 NRDY	PIPE7 NRDY	PIPE6 NRDY	PIPE5 NRDY	PIPE4 NRDY	PIPE3 NRDY	PIPE2 NRDY	PIPE1 NRDY	PIPE0 NRDY
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

位	位名	初始值	R/W	说明
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9	PIPE9NRDY	0	R/W*	管道 9 的 NRDY 中断状态 0: 未发生中断 1: 发生中断
8	PIPE8NRDY	0	R/W*	管道 8 的 NRDY 中断状态 0: 未发生中断 1: 发生中断
7	PIPE7NRDY	0	R/W*	管道 7 的 NRDY 中断状态 0: 未发生中断 1: 发生中断
6	PIPE6NRDY	0	R/W*	管道 6 的 NRDY 中断状态 0: 未发生中断 1: 发生中断
5	PIPE5NRDY	0	R/W*	管道 5 的 NRDY 中断状态 0: 未发生中断 1: 发生中断
4	PIPE4NRDY	0	R/W*	管道 4 的 NRDY 中断状态 0: 未发生中断 1: 发生中断
3	PIPE3NRDY	0	R/W*	管道 3 的 NRDY 中断状态 0: 未发生中断 1: 发生中断
2	PIPE2NRDY	0	R/W*	管道 2 的 NRDY 中断状态 0: 未发生中断 1: 发生中断
1	PIPE1NRDY	0	R/W*	管道 1 的 NRDY 中断状态 0: 未发生中断 1: 发生中断
0	PIPE0NRDY	0	R/W*	管道 0 的 NRDY 中断状态 0: 未发生中断 1: 发生中断

【注】 * 只能写“0”。

29.3.20 BEMP 中断状态寄存器 (BEMPSTS)

BEMPSTS 表示各管道的 BEMP 中断状态。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPE9 BEMP	PIPE8 BEMP	PIPE7 BEMP	PIPE6 BEMP	PIPE5 BEMP	PIPE4 BEMP	PIPE3 BEMP	PIPE2 BEMP	PIPE1 BEMP	PIPE0 BEMP
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

位	位名	初始值	R/W	说明
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9	PIPE9BEMP	0	R/W*	管道 9 的 BEMP 中断状态 0: 未发生中断 1: 发生中断
8	PIPE8BEMP	0	R/W*	管道 8 的 BEMP 中断状态 0: 未发生中断 1: 发生中断
7	PIPE7BEMP	0	R/W*	管道 7 的 BEMP 中断状态 0: 未发生中断 1: 发生中断
6	PIPE6BEMP	0	R/W*	管道 6 的 BEMP 中断状态 0: 未发生中断 1: 发生中断
5	PIPE5BEMP	0	R/W*	管道 5 的 BEMP 中断状态 0: 未发生中断 1: 发生中断
4	PIPE4BEMP	0	R/W*	管道 4 的 BEMP 中断状态 0: 未发生中断 1: 发生中断
3	PIPE3BEMP	0	R/W*	管道 3 的 BEMP 中断状态 0: 未发生中断 1: 发生中断
2	PIPE2BEMP	0	R/W*	管道 2 的 BEMP 中断状态 0: 未发生中断 1: 发生中断
1	PIPE1BEMP	0	R/W*	管道 1 的 BEMP 中断状态 0: 未发生中断 1: 发生中断
0	PIPE0BEMP	0	R/W*	管道 0 的 BEMP 中断状态 0: 未发生中断 1: 发生中断

【注】 * 只能写“0”。

29.3.21 帧号寄存器 (FRMNUM)

FRMNUM 判断等时错误的通知源以及表示帧号等。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OVRN	CRCE	—	—	—	FRNM[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15	OVRN	0	R/W*	<p>超限 / 欠载检测状态</p> <p>对于正在进行等时传送的管道，表示是否检测到超限或者欠载错误。</p> <p>0: 未发生错误 1: 发生错误</p> <p>能通过给此位写“0”，将此位清“0”。</p> <p>(1) 选择主机控制器功能的情况</p> <p>在以下任意情况下，此模块将此位置“1”：</p> <ul style="list-style-type: none"> 对传送类型为等时发送方向的管道，在还没有将发送数据完全写到 FIFO 缓冲器的状态下到了发行 OUT 权标的时间时。 对传送类型为等时接收方向的管道，在至少有 1 个 FIFO 缓冲器面不为空的状态下到了发行 IN 权标的时间时。 <p>(2) 选择功能控制器功能的情况</p> <p>在以下任意情况下，此模块将此位置“1”：</p> <ul style="list-style-type: none"> 对传送类型为等时发送方向的管道，在还没有将发送数据完全写到 FIFO 缓冲器的状态下接收到 IN 权标时。 对传送类型为等时接收方向的管道，在至少有 1 个 FIFO 缓冲器面不为空的状态下接收到 OUT 权标时。 <p>【注】 此位用于调试。在设计时，不能使系统发生超限或者欠载。</p>
14	CRCE	0	R/W*	<p>接收数据错误</p> <p>对于正在进行等时传送的管道，表示 CRC 错误和位填充错误的检测状态，同时产生内部 NRDY 中断请求，详细内容请参照“29.4.2 中断功能”。</p> <p>0: 未发生错误 1: 发生错误</p>
13 ~ 11	—	全 0	R	<p>保留位</p> <p>读写值都为“0”。</p>
10 ~ 0	FRNM [10:0]	H'000	R	<p>帧号</p> <p>此模块在每 1ms 发行或者接收 1 次 SOF 时改写这些位，表示最新的帧号。在读这些位时，必须重复读 2 次，确认 2 次的读取值是否相同。</p>

【注】 * 只能写“0”。

29.3.22 μ 帧号寄存器 (UFRMNUM)

UFRMNUM 表示 μ 帧号。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UFRNM[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 3	—	全 0	R	保留位 读写值都为“0”。
2 ~ 0	UFRNM[2:0]	000	R	μ 帧 这些位能确认 μ 帧号。 在高速运行时，此模块将 μ 帧号反映到这些位。 在高速以外的运行时，将这些位置“B'000”。 在读这些位时，必须重复读 2 次，确认 2 次的读取值是否相同。

29.3.23 USB 地址寄存器 (USBADDR)

USBADDR 表示 USB 地址。

此寄存器只在选择功能控制器功能时有效。当选择主机控制器功能时，必须使用 PIPEMAXP 寄存器的 DEVSEL 位设定外围设备地址。

在上电复位和 USB 总线复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	USBADDR[6:0]						
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 7	—	全 0	R	保留位 读写值都为“0”。
6 ~ 0	USBADDR [6:0]	H'00	R	USB 地址 当选择功能控制器功能时，如果 SET_ADRESS 请求被正常处理，就表示由主机分配的 USB 地址。

29.3.24 USB 请求类型寄存器 (USBREQ)

USBREQ 是保存控制传送设置请求的寄存器。

当选择功能控制器功能时，USBREQ 保存接收到的 bRequest 值和 bmRequestType 值；当选择主机控制器功能时，USBREQ 设定要发送的 bRequest 值和 bmRequestType 值。

在上电复位和 USB 总线复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BREQUEST[7:0]								BMREQUESTTYPE[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

位	位名	初始值	R/W	说明
15 ~ 8	BREQUEST [7:0]	H'00	R/W*	请求 保存 USB 请求的 bRequest 值。 (1) 选择主机控制器功能的情况 必须设定要发送的设置事务的 USB 请求数据值。不能在将 SUREQ 位置“1”后并且在读到 SUREQ 位为“0”前改写这些位。 (2) 选择功能控制器功能的情况 表示在设置事务时接收的 USB 请求数据值。这些位的写操作无效。
7 ~ 0	BMREQUEST TYPE[7:0]	H'00	R/W*	请求类型 保存 USB 请求的 bmRequestType 值。 (1) 选择主机控制器功能的情况 必须设定要发送的设置事务的 USB 请求数据值。不能在将 SUREQ 位置“1”后并且在读到 SUREQ 为“0”前改写这些位。 (2) 选择功能控制器功能的情况 表示在设置事务时接收的 USB 请求数据值。这些位的写操作无效。

【注】 * 当选择功能控制器功能，此位是只读位，写操作无效；当选择主机控制器功能时，此位是可读写位。

29.3.25 USB 请求值寄存器 (USBVAL)

当选择功能控制器功能时，USBVAL 保存接收到的 wValue 值；当选择主机控制器功能时，USBVAL 设定要发送的 wValue 值。

在上电复位和 USB 总线复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WVALUE[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

位	位名	初始值	R/W	说明
15 ~ 0	WVALUE [15:0]	H'0000	R/W*	Value 保存 USB 请求的 wValue 值。 (1) 选择主机控制器功能的情况 必须设定要发送的设置事务的 USB 请求 wValue 值。不能在将 SUREQ 位置“1”后并且在读到 SUREQ 位为“0”前改写这些位。 (2) 选择功能控制器功能的情况 表示在设置事务时接收的 USB 请求 wValue 值。这些位的写操作无效。

【注】 * 当选择功能控制器功能时，此位是只读位，写操作无效；当选择主机控制器功能时，此位是可读写位。

29.3.26 USB 请求变址寄存器 (USBINDX)

USBINDX 是保存控制传送设置请求的寄存器。

当选择功能控制器功能时，USBINDX 保存接收到的 wIndex 值；当选择主机控制器功能时，USBINDX 设定要发送的 wIndex 值。

在上电复位和 USB 总线复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WINDEX[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

位	位名	初始值	R/W	说明
15 ~ 0	WINDEX [15:0]	H'0000	R/W*	变址 保存 USB 请求的 wIndex 值。 (1) 选择主机控制器功能的情况 必须设定要发送的设置事务的 USB 请求 wIndex 值。不能在将 SUREQ 位置“1”后并且在读到 SUREQ 位为“0”前改写这些位。 (2) 选择功能控制器功能的情况 表示设置事务时接收到的 USB 请求 wIndex 值。这些位的写操作无效。

【注】 * 当选择功能控制器功能时，此位是只读位，写操作无效；当选择主机控制器功能时，此位是可读写位。

29.3.27 USB 请求长度寄存器 (USBLENG)

USBLENG 是保存控制传送设置请求的寄存器。

当选择功能控制器功能时，USBLENG 保存接收到的 wLength 值；当选择主机控制器功能时，USBLENG 设定要发送的 wLength 值。

在上电复位和 USB 总线复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WLENGTH[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

位	位名	初始值	R/W	说明
15 ~ 0	WLENGTH [15:0]	H'0000	R/W*	长度 保存 USB 请求的 wLength 值。 (1) 选择主机控制器功能的情况 必须设定要发送的设置事务的 USB 请求 wLength 值。不能在将 SUREQ 位置“1”后并且在读到 SUREQ 位为“0”前改写这些位。 (2) 选择功能控制器功能的情况 表示设置事务时接收到的 USB 请求 wLength 值。这些位的写操作无效。

【注】 * 当选择功能控制器功能时，此位是只读位，写操作无效；当选择主机控制器功能时，此位是可读写位。

29.3.28 DCP 配置寄存器 (DCPCFG)

DCPCFG 给默认控制管道 (DCP) 指定数据的传送方向。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	DIR	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	DIR	0	R/W	传送方向 当选择主机控制器功能时，设定控制传送的数据阶段和状态阶段的传送方向。 0: 数据接收方向 1: 数据发送方向 当选择功能控制器功能时，必须将此位置“0”。
3 ~ 0	—	全 0	R	保留位 读写值都为“0”。

29.3.29 DCP 最大信息包长度寄存器 (DCPMAXP)

DCPMAXP 给 DCP 指定最大信息包的长度。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEVSEL[3:0]				—	—	—	—	—	MXPS[6:0]						
初始值:	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R

位	位名	初始值	R/W	说明
15 ~ 12	DEVSEL[3:0]	0000	R/W	<p>设备选择</p> <p>当选择主机控制器功能时，指定作为控制传送通信对象的外围设备的地址。</p> <p>0000: 地址 0000 0001: 地址 0001 : : 1001: 地址 1001 1010: 地址 1010 上述以外: 禁止设定</p> <p>必须在设定与这些位的设定值对应的 DEVADDn 寄存器后设定这些位。 例如，要将 DEVSEL 位置 “0010” 时，必须设定 DEVADD2 寄存器。 必须在 CSSTS 位为 “0” 并且 PID 位为 NAK 并且 SUREQ 位为 “0” 的期间设定这些位。 如果要在将 DCP 的 PID 位从 BUF 改为 NAK 后更改这些位，就必须在确认 CSSTS 位为 “0” 并且 PBUSY 位为 “0” 后进行更改。如果此模块已将 PID 位改为 NAK，就不需要确认 PBUSY 位。 当选择功能控制器功能时，必须将这些位的值设定为 “B'0000”。</p>
11 ~ 7	—	全 0	R	<p>保留位</p> <p>读写值都为 “0”。</p>
6 ~ 0	MXPS[6:0]	H'40	R/W	<p>最大信息包长度</p> <p>必须给这些位设定 DCP 的最大数据有效负载（最大信息包长度）。 初始值为 “H'40”（64 字节）。 MXPS 位的设定值必须符合 USB 规格。 必须在 CSSTS 位为 “0” 并且 PID 位为 NAK 的状态下设定 MXPS 位。 如果要在将 DCP 的 PID 位从 BUF 改为 NAK 后更改这些位，就必须在确认 CSSTS 位为 “0” 并且 PBUSY 位为 “0” 后进行更改。如果此模块已将 PID 位改为 NAK，就不需要确认 PBUSY 位。 不能在 MXPS 位为 “0” 时写 FIFO 缓冲器或者将 PID 位设定为 BUF。</p>

29.3.30 DCP 控制寄存器 (DCPCTR)

DCPCTR 对 DCP 确认缓冲存储器状态，更改并确认数据 PID 顺序位以及设定应答 PID。

在上电复位时，对此寄存器进行初始化；在 USB 总线复位时，对 CCPL 位和 PID[2:0] 位进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	SUREQ	CSCLR	CSSTS	SUREQ CLR	—	—	SQCLR	SQSET	SQMON	PBUSY	PINGE	—	CCPL	PID[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
R/W:	R	R/W*2	R/W*1	R	R/W*1	R	R	R/W*1	R/W*1	R	R	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	BSTS	0	R	<p>缓冲器状态</p> <p>此位表示是否能存取 DCP 的 FIFO 缓冲器的状态。</p> <p>0: 不能存取缓冲器</p> <p>1: 能存取缓冲器</p> <p>此位的含义因 ISEL 位的设定值而不同。</p> <ul style="list-style-type: none"> 当 ISEL 位为 “0” 时，表示是否能读接收数据。 当 ISEL 位为 “1” 时，表示是否能写发送数据。
14	SUREQ	0	R/W*2	<p>设置权标发送</p> <p>当选择主机控制器功能时，通过将此位置 “1”，发送设置包。</p> <p>0: 无效</p> <p>1: 发送设置包</p> <p>在设置事务处理结束后，此模块产生 SACK 中断或者 SIGN 中断，并且将此位清 “0”。</p> <p>通过将 SUREQCLR 位置 “1”，此模块将此位清 “0”。</p> <p>必须在给 DEVSEL 位、USBREQ 寄存器、USBVAL 寄存器、USBINDX 寄存器和 USBLENG 寄存器设定设置事务中要发送的 USB 请求后，将此位置 “1”。必须在将 SUREQ 位置 “1” 前，确认是否已将 DCP 的 PID 位设定为 NAK。在将此位置 “1” 到设置事务结束前（SUREQ 位为 “1”）的期间，不能更改 DEVSEL 位、USBREQ 寄存器、USBVAL 寄存器、USBINDX 寄存器和 USBLENG 寄存器的值。</p> <p>只在发出设置权标时，才必须将此位置 “1”，否则必须置 “0”。</p> <p>当选择功能控制器功能时，必须给此位写 “0”。</p>
13	CSCLR	0	R/W*1	<p>分离事务的 C-SPLIT 状态清除</p> <p>当选择主机控制器功能时，如果在使用分离事务的传送中将此位置 “1”，就能将 CSSTS 位清 “0”。此时，从 S-SPLIT 重新开始 DCP 的下次传送。</p> <p>0: 无效</p> <p>1: 将 CSSTS 位清 “0”</p> <p>如果将此位置 “1”，此模块就将 CSSTS 位清 “0”。</p> <p>在使用分离事务的传送中，如果想强制从 S-SPLIT 重新开始下次传送，就必须将此位置 “1”。正常的分离事务在 C-SPLIT 结束时，此模块自动将 CSSTS 位清 “0”，因此不需要进行清除处理。</p> <p>必须在通过将 UACT 位置 “0” 来停止通信时或者在检测到断开时尚未传送的情况下，通过此位控制 CSSTS 位。</p> <p>在 CSSTS 位为 “0” 时，即使将此位置 “1”，CSSTS 位也保持 “0”。</p> <p>当选择功能控制器功能时，必须给此位写 “0”。</p>

位	位名	初始值	R/W	说明
12	CSSTS	0	R	<p>分离事务的 COMPLETE SPLIT (C-SPLIT) 状态</p> <p>当选择主机控制器功能时, 表示分离事务的 C-SPLIT 状态。</p> <p>0: 正在进行 START-SPLIT (S-SPLIT) 事务处理或者正在处理未使用分离事务的设备</p> <p>1: 正在进行 C-SPLIT 事务处理</p> <p>在 C-SPLIT 开始时, 此模块将此位置 “1”; 在检测到 C-SPLIT 结束时, 此模块将此位置 “0”。</p> <p>当选择功能控制器功能时, 此位的读取值无效。</p>
11	SUREQCLR	0	R/W*1	<p>SUREQ 位清除</p> <p>当选择主机控制器功能时, 如果将此位置 “1”, 就能将 SUREQ 位清 “0”。</p> <p>0: 无效</p> <p>1: 将 SUREQ 位清 “0”</p> <p>此位总是 “0”。</p> <p>在设置事务中, 如果在 SUREQ 位为 “1” 的状态下停止通信, 就必须将此位置 “1”。正常的设置事务在事务结束时, 此模块自动将 SUREQ 位清 “0”, 因此不需要进行清除处理。</p> <p>必须在通过将 UACT 位置 “0” 来停止通信时或者在检测到断开时尚未传送的情况下, 通过此位控制 SUREQ 位。</p> <p>当选择功能控制器功能时, 必须给此位写 “0”。</p>
10、9	—	全 0	R	<p>保留位</p> <p>读写值都为 “0”。</p>
8	SQCLR	0	R/W*1	<p>交替位清除</p> <p>在 DCP 传送中, 能将下一个事务的顺序交替位的期待值设定为 DATA0。</p> <p>0: 无效</p> <p>1: 指定 DATA0</p> <p>此位总是 “0”。</p> <p>不能同时将 SQCLR 位和 SQSET 位置 “1”。</p> <p>必须在 CSSTS 位为 “0” 并且 PID 位为 NAK 时将此位置 “1”。</p> <p>如果要在将 DCP 的 PID 位从 BUF 改为 NAK 后将此位置 “1”, 就必须在确认 CSSTS 位为 “0” 并且 PBUSY 位为 “0” 后进行更改。如果此模块已将 PID 位改为 NAK, 就不需要确认 PBUSY 位。</p>
7	SQSET	0	R/W*1	<p>交替位置位</p> <p>在 DCP 传送中, 能将下一个事务的顺序交替位的期待值设定为 DATA1。</p> <p>0: 无效</p> <p>1: 指定 DATA1</p> <p>不能同时将 SQCLR 位和 SQSET 位置 “1”。</p> <p>必须在 CSSTS 位为 “0” 并且 PID 位为 NAK 时将此位置 “1”。</p> <p>如果要在将 DCP 的 PID 位从 BUF 改为 NAK 后将此位置 “1”, 就必须在确认 CSSTS 位为 “0” 并且 PBUSY 位为 “0” 后进行更改。如果此模块已将 PID 位改为 NAK, 就不需要确认 PBUSY 位。</p>

位	位名	初始值	R/W	说明
6	SQMON	1	R	<p>顺序交替位监视</p> <p>在 DCP 传送中，表示下一个事务的顺序交替位的期待值。</p> <p>0: DATA0 1: DATA1</p> <p>如果事务被正常处理，此模块就将此位取反。如果在进行接收方向的传送时 DATA-PID 不同，就不将此位取反。</p> <p>当选择功能控制器功能时，如果正常接收设置包，此模块就将此位置“1”（将期待值设定为 DATA1）。</p> <p>当选择功能控制器功能时，此模块在状态阶段的 IN/OUT 事务中不参照此位。即使正常结束，也不将此位取反。</p>
5	PBUSY	0	R	<p>管道忙</p> <p>表示在当前事务中是否使用相应的管道。</p> <p>0: 在事务处理时未使用相应的管道 1: 在事务处理时使用相应的管道</p> <p>在开始 DCP 的 USB 事务时，此模块将此位从“0”改为“1”；在 1 个事务正常结束时，将此位从“1”改为“0”。</p> <p>在将 PID 位设定为 NAK 后，能通过读此位来确认是否能更改管道的设定。详细内容请参照“29.4.3(1) 管道控制寄存器的切换步骤”。</p>
4	PINGE	0	R/W	<p>PING 权标发行允许</p> <p>当选择主机控制器功能时，如果将此位置“1”，此模块就在进行发送方向的传送时发行 PING 权标。从 PING 事务开始发送方向的传送。</p> <p>0: 禁止发行 PING 权标 1: 正常 PING 运行</p> <p>如果在 PING 事务中检测到 ACK 握手信号，就在下一个事务中执行 OUT 事务。</p> <p>如果在 OUT 事务中检测到 NAK 握手信号，就在下一个事务中执行 PING 事务。</p> <p>当选择主机控制器功能时，如果将此位置“0”，此模块就不在进行发送方向的传送时发行 PING 权标。全部在 OUT 事务中进行发送方向的传送。必须在 CSSTS 位为“0”并且 PID 位为 NAK 时更改此位。</p> <p>如果要在将 DCP 的 PID 位从 BUF 改为 NAK 后更改此位，就必须在确认 CSSTS 位为“0”并且 PBUSY 位为“0”后进行更改。如果此模块已将 PID 位改为 NAK，就不需要确认 PBUSY 位。</p> <p>当选择功能控制器功能时，必须给此位写“0”。</p>
3	—	0	R	<p>保留位</p> <p>读写值都为“0”。</p>

位	位名	初始值	R/W	说明
2	CCPL	0	R/W*1	<p>控制传送结束允许</p> <p>当选择功能控制器功能时，通过将此位置“1”，设定为允许结束控制传送的状态阶段。</p> <p>0: 无效</p> <p>1: 允许结束控制传送</p> <p>当对应的 PID 位为 BUF 时，如果将此位置“1”，此模块就结束控制传送的阶段。</p> <p>即，在进行控制读传送时，对 USB 主机的 OUT 事务发送 ACK 握手信号；在进行控制写和无数据控制传送时，对 USB 主机的 IN 事务发送 Zero-Length 包。当检测到 SET_ADDRESS 请求时，与此位的设定值无关，此模块在从设置阶段到状态阶段结束的期间进行自动应答。</p> <p>如果接收到新的设置包，此模块就将此位从“1”改为“0”。</p> <p>当 VALID 位为“1”时，不能给此位写“1”。</p> <p>当选择主机控制器功能时，必须给此位写“0”。</p>
1、0	PID[1:0]	00	R/W	<p>应答 PID</p> <p>通过这些位在控制传送中控制此模块的应答。</p> <p>00: NAK 应答</p> <p>01: BUF 应答（取决于缓冲器状态）</p> <p>10: STALL 应答</p> <p>11: STALL 应答</p> <p>(1) 选择主机控制器功能的情况</p> <p>必须按照以下步骤将这些位从 NAK 改为 BUF:</p> <ul style="list-style-type: none"> 当设定为发送方向时 <ul style="list-style-type: none"> 必须在 UACT 位为“1”并且 PID 位为 NAK 的状态下将发送数据全部写到 FIFO 缓冲器，并且给 PID 位写 BUF。在给 PID 位写 BUF 后，此模块执行 OUT 事务（或者 PING 事务）。 当设定为接收方向时 <ul style="list-style-type: none"> 必须在 UACT 位为“1”并且 PID 位为“NAK”的状态下确认 FIFO 缓冲器为空状态（空的状态），并且给 PID 位写 BUF。在给 PID 位写 BUF 后，此模块执行 IN 事务。 <p>在以下情况下，此模块更改这些位的值:</p> <ul style="list-style-type: none"> 在将这些位设定为 BUF 的情况下，如果此模块接收到超过 MaxPacketSize 的数据，就将 PID 位设定为 STALL（11）。 如果连续 3 次检测到 CRC 错误等接收错误，此模块就将 PID 位设定为 NAK。 如果接收到 STALL 握手信号，此模块就将 PID 位设定为 STALL（11）。 <p>即使此模块在对所选管道发行分离事务的 S-SPLIT 后（CSSTS 位为“1”）将这些位改为 NAK，也在 C-SPLIT 结束前执行事务。在 C-SPLIT 结束时，此模块将 PID 位设定为 NAK。</p> <p>(2) 选择功能控制器功能的情况</p> <p>在以下情况下，此模块更改这些位的值:</p> <ul style="list-style-type: none"> 如果此模块接收到设置包，就将 PID 位改为 NAK。此时，此模块将 VALID 位置“1”，在将 VALID 位置“0”前不能更改这些位。 在将这些位设定为 BUF 的情况下，如果此模块接收到超过 MaxPacketSize 的数据，此模块就将 PID 位设定为 STALL（11）。 如果此模块检测到控制传送顺序错误，就将 PID 位设定为 STALL（1x）。 如果此模块检测到 USB 总线复位，就将 PID 位设定为 NAK。 <p>此模块在进行 SET_ADDRESS 请求处理（自动处理）时不参照这些位的设定值。</p>

【注】 *1 读取值总是“0”，只能写“1”。

*2 只能写“1”。

29.3.31 管道窗口选择寄存器 (PIPESEL)

必须通过 PIPESEL、PIPECFG、PIPEBUF、PIPEMAXP、PIPEPERI、PIPECTR、PIPEnTRE 和 PIPEnTRN 寄存器设定管道 1 ~ 9。

在通过 PIPESEL 寄存器选择要使用的管道后，给 PIPECFG、PIPEBUF、PIPEMAXP 和 PIPEPERI 寄存器设定各管道的功能。与 PIPESEL 寄存器选择的管道无关，能设定 PIPECTR、PIPEnTRE 和 PIPEnTRN 寄存器。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	PIPESEL[3:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 4	—	全 0	R	保留位 读写值都为“0”。
3 ~ 0	PIPESEL [3:0]	0000	R/W	管道窗口选择 如果将这些位置“0001”~“1001”，PIPECFG、PIPEBUF、PIPEMAXP、PIPEPERI 寄存器就表示对应管道的信息和设定值。 0000: 未选择 0001: 管道 1 0010: 管道 2 0011: 管道 3 0100: 管道 4 0101: 管道 5 0110: 管道 6 0111: 管道 7 1000: 管道 8 1001: 管道 9 上述以外: 禁止设定 如果将这些位置“0000”，就将 PIPECFG、PIPEBUF、PIPEMAXP、PIPEPERI 寄存器全部置“0”。此时，上述寄存器的写操作无效。

29.3.32 管道配置寄存器 (PIPECFG)

PIPECFG 指定管道 1 ~ 9 的传送类型、缓冲存储器的存取方向和端点号，并且选择是连续传送模式还是非连续传送模式、是单缓冲器还是双缓冲器以及在传送结束时是否禁止管道运行。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TYPE[1:0]		—	—	—	BFRE	DBLB	CNTMD	SHT NAK	—	—	DIR	EPNUM[3:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15、14	TYPE[1:0]	00	R/W	<p>传送类型</p> <p>指定 PIPESEL 位所选管道的传送类型。</p> <ul style="list-style-type: none"> 管道 1 和管道 2 的情况 <ul style="list-style-type: none"> 00: 不能使用管道 01: 批量传送 10: 禁止设定 11: 等时传送 管道 3 ~ 5 的情况 <ul style="list-style-type: none"> 00: 不能使用管道 01: 批量传送 10: 禁止设定 11: 禁止设定 管道 6 ~ 9 的情况 <ul style="list-style-type: none"> 00: 不能使用管道 01: 禁止设定 10: 中断传送 11: 禁止设定 <p>必须在将所选管道的 PID 位设定为 BUF (开始使用所选管道的 USB 通信) 前将这些位设定为“00”以外的值。</p> <p>必须在所选管道的 PID 位为 NAK 的状态下更改这些位。如果要在将所选管道的 PID 位从 BUF 改为 NAK 后更改这些位，就必须在确认 CSSTS 位为“0”并且 PBUSY 位为“0”后进行更改。如果此模块已将 PID 位改为 NAK，就不需要确认 PBUSY 位。</p>
13 ~ 11	—	全 0	R	<p>保留位</p> <p>读写值都为“0”。</p>

位	位名	初始值	R/W	说明
10	BFRE	0	R/W	<p>BRDY 中断运行指定</p> <p>指定此模块向 CPU 发行所选管道相关 BRDY 中断的时序。</p> <p>0: 在发送和接收时发生 BRDY 中断</p> <p>1: 在读完数据时发生 BRDY 中断</p> <p>此位在选择管道 1 ~ 5 时有效。</p> <p>如果将此位置“1”并且将所选管道用于接收方向, 此模块就检测到传送结束并且在读完信息包时产生 BRDY 中断。</p> <p>当通过此设定发生 BRDY 中断时, 需要给 BCLR 位写“1”。在给 BCLR 位写“1”前, 分配给所选管道的 FIFO 缓冲器为不可接收的状态。</p> <p>如果将此位置“1”并且将所选管道用于发送方向, 此模块就不产生 BRDY 中断。</p> <p>详细内容请参照“29.4.2(1) BRDY 中断”。</p> <p>必须在 CSSTS 位为“0”并且 PID 位为 NAK 而未给 CURPIPE 位设定管道的状态下更改此位。</p> <p>如果要在进行使用所选管道的 USB 通信后更改此位, 就必须在上述 3 个寄存器的状态下给 ACLRM 位连续写“1”和“0”, 清除分配给所选管道的 FIFO 缓冲器。</p> <p>如果要在将所选管道的 PID 位从 BUF 改为 NAK 后更改此位, 就必须在确认 CSSTS 位为“0”并且 PBUSY 位为“0”后进行更改。如果此模块已将 PID 位改为 NAK, 就不需要确认 PBUSY 位。</p>
9	DBLB	0	R/W	<p>双缓冲器模式</p> <p>指定所选管道要使用的 FIFO 缓冲器是单缓冲器还是双缓冲器。</p> <p>0: 单缓冲器</p> <p>1: 双缓冲器</p> <p>此位在选择管道 1 ~ 5 时有效。</p> <p>如果将此位置“1”, 此模块就将 PIPEBUF 寄存器的 BUFSIZE 位指定的 2 个 FIFO 缓冲器面分配给所选管道。</p> <p>即, 此模块分配给所选管道的 FIFO 缓冲器容量如下:</p> <p>$(BUFSIZE+1) \times 64 \times (DBLB+1)$ [字节]</p> <p>必须在 CSSTS 位为“0”并且 PID 位为 NAK 而未给 CURPIPE 位设定管道的状态下更改此位。</p> <p>如果要在进行使用所选管道的 USB 通信后更改此位的设定, 就必须在上述 3 个寄存器的状态下给 ACLRM 位连续写“1”和“0”, 清除分配给所选管道的 FIFO 缓冲器。</p> <p>如果要在将所选管道的 PID 位从 BUF 改为 NAK 后更改此位, 就必须在确认 CSSTS 位为“0”并且 PBUSY 位为“0”后进行更改。如果此模块已将 PID 位改为 NAK, 就不需要确认 PBUSY 位。</p>

位	位名	初始值	R/W	说明
8	CNTMD	0	R/W	<p>连续传送模式</p> <p>指定所选管道是否在连续传送模式中进行通信。</p> <p>0: 非连续传送模式 1: 连续传送模式</p> <p>此位在通过 PIPESEL 位选择管道 1 ~ 5 并且选择批量传送 (TYPE 位为“01”) 时有效。</p> <p>必须在 CSSTS 位为“0”并且 PID 位为 NAK 而未给 CURPIPE 位设定管道的状态下更改此位。</p> <p>如果要在进行使用所选管道的 USB 通信后更改此位, 就必须在上述 3 个寄存器的状态下给 ACLRM 位连续写“1”和“0”, 清除分配给所选管道的 FIFO 缓冲器。</p> <p>如果要在将所选管道的 PID 位从 BUF 改为 NAK 后更改此位, 就必须在确认 CSSTS 位为“0”并且 PBUSY 位为“0”后进行更改。如果此模块已将 PID 位改为 NAK, 就不需要确认 PBUSY 位。</p>
7	SHTNAK	0	R/W	<p>传送结束时的管道禁止</p> <p>当所选管道为接收方向时, 指定是否在传送结束时将 PID 位改为 NAK。</p> <p>0: 传送结束时继续管道 1: 传送结束时禁止管道</p> <p>此位在所选管道为管道 1 ~ 管道 5 并且为接收方向时有效。</p> <p>对于接收方向的管道, 如果将此位置“1”, 此模块就在对所选管道判断为传送结束时将所选管道对应的 PID 位改为 NAK。此模块在满足以下条件时判断为传送结束。</p> <ul style="list-style-type: none"> • 当正常接收短包数据 (包括 Zero-Length 包) 时 • 当使用事务计数器并且正常接收事务计数器指定数量的信息包时必须 <p>必须在 CSSTS 位为“0”并且 PID 位为 NAK 的状态下更改此位。</p> <p>如果要在将所选管道的 PID 位从 BUF 改为 NAK 后更改此位, 就必须在确认 CSSTS 位为“0”并且 PBUSY 位为“0”后进行更改。如果此模块已将 PID 位改为 NAK, 就不需要确认 PBUSY 位。</p> <p>对于发送方向的管道, 必须将此位置“0”。</p>
6、5	—	全 0	R	<p>保留位</p> <p>读写值都为“0”。</p>
4	DIR	0	R/W	<p>传送方向</p> <p>指定所选管道的传送方向。</p> <p>0: 接收方向 1: 发送方向</p> <p>在将此位置“0”的情况下, 如果此模块将所选管道设定为接收方向并且将此位置“1”, 此模块就将所选管道用于发送方向。</p> <p>必须在 CSSTS 位为“0”并且 PID 位为 NAK 而未给 CURPIPE 位设定管道的状态下更改此位。</p> <p>如果要在进行使用所选管道的 USB 通信后更改此位, 就必须在上述 3 个寄存器的状态下给 ACLRM 位连续写“1”和“0”, 清除分配给所选管道的 FIFO 缓冲器。</p> <p>如果要在将所选管道的 PID 位从 BUF 改为 NAK 后更改此位, 就必须在确认 CSSTS 位为“0”并且 PBUSY 位为“0”后进行更改。如果此模块已将 PID 位改为 NAK, 就不需要确认 PBUSY 位。</p>

位	位名	初始值	R/W	说明
3 ~ 0	EPNUM[3:0]	0000	R/W	<p>端点号</p> <p>指定所选管道的端点号。</p> <p>如果将这些位置“0000”，就表示是未使用的管道。</p> <p>必须在 CSSTS 位为“0”并且 PID 位为 NAK 的状态下更改此位。</p> <p>如果要在将所选管道的 PID 位从 BUF 改为 NAK 后更改此位，就必须在确认 CSSTS 位为“0”并且 PBUSY 位为“0”后进行更改。如果此模块已将 PID 位改为 NAK，就不需要确认 PBUSY 位。</p> <p>DIR 位和 EPNUM 位的设定组合不能和其他管道的设定重复（能重复将 EPNUM 位置为“0000”）。</p>

29.3.33 管道缓冲器指定寄存器（PIPEBUF）

PIPEBUF 指定管道 1 ~ 9 的缓冲器大小和缓冲器号。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	BUFSIZE[4:0]						—	—	—	BUFNMB[6:0]					
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	<p>保留位</p> <p>读写值都为“0”。</p>
14 ~ 10	BUFSIZE [4:0]	H'00	R/W	<p>缓冲器大小</p> <p>指定 PIPESEL 位所选管道的缓冲器大小。以块为单位，1 个块为 64 字节。</p> <p>00000 (H'00): 64 字节</p> <p>00001 (H'01): 128 字节</p> <p> :</p> <p> :</p> <p>11111 (H'1F): 2K 字节</p> <p>当 DBLB 位为“1”时，此模块将这些位指定的 2 个 FIFO 缓冲器面分配给所选管道。</p> <p>此模块分配给所选管道的 FIFO 缓冲器容量如下：</p> <p>(BUFSIZE+1)×64×(DBLB+1) [字节]</p> <p>这些位能设定的值因所选管道而不同。</p> <p>在管道 1 ~ 5 的情况下，必须将 BUFSIZE 位置“H'00” ~ “H'1F”。</p> <p>在管道 6 ~ 9 的情况下，必须将 BUFSIZE 位置“H'00”。</p> <p>在 CNTMD 位为“1”的情况下使用时，必须给这些位设定 MaxPacketSize 整数倍的值。</p> <p>必须在 CSSTS 位为“0”并且 PID 位为 NAK 并且未给 CURPIPE 位设定管道的状态下更改这些位。</p> <p>如果要在将所选管道的 PID 位从 BUF 改为 NAK 后更改这些位，就必须在确认 CSSTS 位为“0”并且 PBUSY 位为“0”后进行更改。如果此模块已将 PID 位改为 NAK，就不需要确认 PBUSY 位。</p>
9 ~ 7	—	全 0	R	<p>保留位</p> <p>读写值都为“0”。</p>

位	位名	初始值	R/W	说明
6 ~ 0	BUFNMB [6:0]	H'00	R/W	<p>缓冲器号</p> <p>必须指定分配给所选管道的 FIFO 缓冲器中的起始块号。分配给所选管道的 FIFO 缓冲器块如下： 块号：BUFNMB ~ 块号：BUFNMB + (BUFSIZE+1)×(DBLB+1)-1 指定“H'04”~“H'7F”范围内的值。</p> <p>在将管道 1 ~ 5 设定为所选管道时，能给这些位设定适合用户系统的值。 BUFNMB=0 ~ 3 为 DCP 专用。 BUFNMB=4 为管道 6 专用。 但是，在不使用管道 6 时，其他管道能使用。 当所选管道为管道 6 时，这些位的写操作无效，模块自动分配 BUFNMB=4。 BUFNMB=5 为管道 7 专用。 但是，在不使用管道 7 时，其他管道能使用。 当所选管道为管道 7 时，这些位的写操作无效，模块自动分配 BUFNMB=5。 BUFNMB=6 为管道 8 专用。 但是，在不使用管道 8 时，其他管道能使用。 当所选管道为管道 8 时，这些位的写操作无效，模块自动分配 BUFNMB=6。 BUFNMB=7 为管道 9 专用。 但是，在不使用管道 9 时，其他管道能使用。 当所选管道为管道 9 时，这些位的写操作无效，模块自动分配 BUFNMB=7。</p> <p>必须在 CSSTS 位为“0”并且 PID 位为 NAK 并且未给 CURPIPE 位设定管道的状态下更改这些位。 如果要在将所选管道的 PID 位从 BUF 改为 NAK 后更改这些位，就必须在确认 CSSTS 位为“0”并且 PBUSY 位为“0”后进行更改。如果此模块已将 PID 位改为 NAK，就不需要确认 PBUSY 位。</p>

29.3.34 管道最大信息包长度寄存器 (PIPEMAXP)

PIPEMAXP 给管道 1 ~ 9 指定最大信息包的长度。

在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	DEVSEL[3:0]				—	MXPS[10:0]											
初始值:	0	0	0	0	0	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

位	位名	初始值	R/W	说明
15 ~ 12	DEVSEL [3:0]	0000	R/W	<p>设备选择</p> <p>当选择主机控制器功能时，指定通信对象的外围设备的 USB 设备地址。</p> <p>0000: 地址 0000 0001: 地址 0001 0010: 地址 0010 : : 1010: 地址 1010</p> <p>上述以外: 禁止设定</p> <p>必须在设定与这些位的设定值对应的 DEVADDn (n=0 ~ A) 寄存器后设定这些位。</p> <p>例如，要将 DEVSEL 位置 “0010” 时，必须设定 DEVADD2 地址。</p> <p>如果要在将 PID 位从 BUF 改为 NAK 后更改这些位，就必须在确认所选管道的 CSSTS 位为 “0” 并且 PBUSY 位为 “0” 后进行更改。如果此模块已将 PID 位改为 NAK，就不需要确认 PBUSY 位。</p> <p>当选择功能控制器功能时，必须将这些位的值设定为 “B'0000”。</p>
11	—	0	R	<p>保留位</p> <p>读写值都为 “0”。</p>
10 ~ 0	MXPS[10:0]	*	R/W	<p>最大信息包长度</p> <p>指定所选管道的最大数据有效负载（最大信息包长度）。各管道能设定的值的范围如下：</p> <p>管道 1、2: 1 字节 (H'001) ~ 1024 字节 (H'400)</p> <p>管道 3 ~ 5: 8 字节 (H'008)、16 字节 (H'010)、32 字节 (H'020)、64 字节 (H'040)、512 字节 (H'200)</p> <p>(没有 [2:0] 的位。)</p> <p>管道 6 ~ 9: 1 字节 (H'001) ~ 64 字节 (H'040)</p> <p>MXPS 位的设定值必须符合各传送类型的 USB 规格。</p> <p>在分离事务中进行等时管道通信时，必须给 MXPS 位设定不超过 188 字节的值。</p> <p>必须在 CSSTS 位为 “0” 并且 PID 位为 NAK 并且未给 CURPIPE 位设定管道的状态下更改这些位。</p> <p>如果要在将所选管道的 PID 位从 BUF 改为 NAK 后更改这些位，就必须在确认 CSSTS 位为 “0” 并且 PBUSY 位为 “0” 后进行更改。如果此模块已将 PID 位改为 NAK，就不需要确认 PBUSY 位。</p> <p>不能在 MXPS 位为 “0” 时写 FIFO 缓冲器或者将 PID 位设定为 BUF。</p>

【注】 * 在未通过 PIPESEL 寄存器的 PIPESEL 位选择管道时，初始值为 “H'000”；在选择管道时，初始值为 “H'040”。

29.3.35 管道周期控制寄存器 (PIPEPERI)

对管道 1 ~ 9, PIPEPERI 选择在等时 IN 传送过程中发生间隔错误时是否运行缓冲器转储清除功能, 并且设定间隔错误的检测间隔。

在上电复位时, 对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	IFIS	—	—	—	—	—	—	—	—	—	IITV[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12	IFIS	0	R/W	<p>等时 IN 缓冲器转储清除</p> <p>在 PIPESEL 位指定的管道 (所选管道) 为等时 IN 传送时, 此位指定是否进行缓冲器的转储清除。</p> <p>0: 不进行缓冲器转储清除 1: 进行缓冲器转储清除</p> <p>在选择功能控制器功能的情况下, 当所选管道的传送类型为等时传送并且传送方向为 IN 传送时, 如果此模块在 IITV 位设定的各间隔的 (μ) 帧中未接收到 USB 主机的 IN 权标, 就自动清除 FIFO 缓冲器。</p> <p>在设定双缓冲器时 (将 DBLB 位置“1”), 此模块只清除先前使用的 1 个缓冲器面的数据。</p> <p>在应该接收 IN 权标的 (μ) 帧之后接收到 SOF 包时, 清除 FIFO 缓冲器。即使 SOF 包破损, 也通过内插功能在应该接收 SOF 时清除缓冲器。</p> <p>当选择主机控制器功能时, 必须将此位置“0”。</p> <p>在所选管道的传送类型不是等时传送时, 必须将此位置“0”。</p>
11 ~ 3	—	全 0	R	保留位 读写值都为“0”。
2 ~ 0	IITV[2:0]	000	R/W	<p>间隔错误的检测间隔</p> <p>必须将所选管道的间隔错误的检测间隔指定为帧时序的 2 的 n 次方。如下所述, 详细功能根据是选择主机控制器功能还是选择功能控制器功能而不同。</p> <p>必须在 CSSTS 位为“0”并且 PID 位为 NAK 并且未给 CURPIPE 位设定管道时设定这些位。</p> <p>如果要在将所选管道的 PID 位从 BUF 改为 NAK 后更改这些位, 就必须在确认 CSSTS 位为“0”并且 PBUSY 位为“0”后进行更改。如果此模块已将 PID 位改为 NAK, 就不需要确认 PBUSY 位。</p> <p>如果要在设定这些位并且进行 USB 通信后改为其他值, 就必须在将 PID 位设定为 NAK 后将 ACLRM 位置“1”, 对间隔定时器进行初始化。</p> <p>管道 3 ~ 5 没有此位。必须将与管道 3 ~ 5 对应的这些位置“000”。</p>

29.3.36 管道 n 控制寄存器 (PIPEnCTR) (n=1 ~ 9)

PIPEnCTR 对管道 1 ~ 9 确认缓冲存储器的状态, 更改和确认数据 PID 的顺序位, 选择是否设定为自动应答模式以及是否设定为自动缓冲器清除模式, 设定应答 PID。此设定与 PIPESEL 寄存器选择的管道无关。

在上电复位时, 对此寄存器进行初始化; 在 USB 总线复位时, 对 PID[1:0] 位进行初始化。

(1) PIPEnCTR (n=1 ~ 5)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	INBUFM	CSCLR	CSSTS	—	AT REPM	ACLRM	SQCLR	SQSET	SQMON	PBUSY	—	—	—	PID[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W*	R	R	R/W	R/W	R/W*	R/W*	R	R	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15	BSTS	0	R	缓冲器状态 表示相应管道的 FIFO 缓冲器状态。 0: CPU 不能存取缓冲器 1: CPU 能存取缓冲器 如表 29.10 所示, 此位的含义因 DIR 位、BFRE 位和 DCLR M 位的设定值而不同。
14	INBUFM	0	R	发送缓冲器监视 在相应管道为发送方向时, 此位表示该管道的 FIFO 缓冲器状态。 0: 缓冲存储器没有能发送的数据 1: 缓冲存储器有能发送的数据 在将相应管道设定为发送方向 (DIR 位为“1”)时, 如果至少将 1 个缓冲器面的数据写到 FIFO 缓冲器, 此模块就将此位置“1”。 当此模块将写完的 FIFO 缓冲器面的数据全部发送结束时, 将此位置“0”。 如果使用双缓冲器 (将 DBLB 位置“1”), 此模块就在发送完 2 个缓冲器面的数据而未写完 1 个缓冲器面的数据时将此位置“0”。 在将相应管道设定为接收方向 (DIR 位为“0”)时, 此位和 BSTS 位的值相同。
13	CSCLR	0	R/W*	CSPLIT 状态清除位 当选择主机控制器功能时, 如果将此位置“1”, 此模块就将 CSSTS 位清“0”。 0: 写操作无效 1: 清除 CSSTS 位 在使用分离事务的传送中, 如果要强制从 S-SPLIT 重新开始下一次传送, 就必须将此位置“1”。正常的分离事务在 C-SPLIT 结束时, 此模块自动将 CSSTS 位清“0”, 因此不需要进行清除处理。 必须在通过将 UACT 位置“0”来停止通信时或者在检测到断开时尚未传送的情况下, 通过此位控制 CSSTS 位。 在 CSSTS 位为“0”时, 即使将此位置“1”, CSSTS 位也保持“0”。 当选择功能控制器功能时, 必须给此位写“0”。

位	位名	初始值	R/W	说明
12	CSSTS	0	R	<p>CSSTS 状态位</p> <p>当选择主机控制器功能时，此位表示分离事务的 C-SPLIT 状态。</p> <p>0: 正在进行 START-SPLIT (S-SPLIT) 事务处理或者正在进行未使用分离事务的传送</p> <p>1: 正在进行 C-SPLIT 事务处理</p> <p>在 C-SPLIT 开始时，此模块将此位置“1”；在检测到 C-SPLIT 结束时，此模块将此位置“0”。</p> <p>如果在 C-SPLIT 处理过程中断开，此位就可能保持“1”的状态。此时（检测到 DTCH 位为“1”时），必须通过 CSCLR 位清除此位。</p> <p>此位只在选择主机控制器功能时有效。</p>
11	—	0	R	<p>保留位</p> <p>读写值都为“0”。</p>
10	ATREPM	0	R/W	<p>自动应答模式</p> <p>指定禁止或者允许相应管道的自动应答。</p> <p>0: 禁止自动应答</p> <p>1: 允许自动应答</p> <p>当选择功能控制器功能时，如果将相应管道的传送类型设定为批量传送，就能将此位置“1”。</p> <p>如果将此位置“1”，此模块就对 USB 主机的权标进行如下应答：</p> <p>(1) 相应管道为 Bulk-IN 传送（将 TYPE 位置“01”并且将 DIR 位置“1”）的情况</p> <p>在 ATREPM 位为“1”并且 PID 位为 BUF 时，此模块对 IN 权标发送 Zero-Length 包。</p> <p>在每次接收 USB 主机的 ACK 时（1 个事务为接收 IN 权标 → 发送 Zero Length 包 → 接收 ACK），此模块更新（交替）顺序交替位（DATA-PID）。不发生 BRDY 中断和 BEMP 中断。</p> <p>(2) 相应管道为 Bulk-OUT 传送（将 TYPE 位置“01”并且将 DIR 位置“0”）的情况</p> <p>在 ATREPM 位为“1”并且 PID 位为 BUF 时，此模块对 OUT 权标（或者 PING 权标）进行 NAK 应答并且产生 NRDY 中断。</p> <p>必须在 CSSTS 位“0”并且 PID 位为 NAK 时更改此位。</p> <p>如果要在将对应管道的 PID 位从 BUF 改为 NAK 后更改此位，就必须在确认 CSSTS 位为“0”并且 PBUSY 位为“0”后进行更改。如果此模块已将 PID 位改为 NAK，就不需要确认 PBUSY 位。</p> <p>要在将此位置“1”后进行 USB 通信时，必须在 FIFO 缓冲器为空的状态下进行设定。在将此位置“1”到进行 USB 通信的期间，不能写 FIFO 缓冲器。</p> <p>在相应管道的传送类型为等时传送时，必须将此位置“0”。</p> <p>当选择主机控制器功能时，必须给此位写“0”。</p>

位	位名	初始值	R/W	说明
9	ACLRM	0	R/W	<p>缓冲器自动清除模式 指定禁止或者允许相应管道的缓冲器自动清除模式。 0: 禁止 1: 允许 (对全部缓冲器进行初始化)</p> <p>如果要分配给相应管道的 FIFO 缓冲器内容全部删除, 就必须给 ACLRM 位连续写 “1” 和 “0”。</p> <p>在给此位连续写 “1” 和 “0” 时, 此模块要清除的内容如表 29.11 (1) 所示, 需清除的情况如表 29.11 (2) 所示。</p> <p>必须在 CSSTS 位为 “0” 并且 PID 位为 NAK 而未给 CURPIPE 位设定相应管道时更改此位。</p> <p>如果要在将对应管道的 PID 位从 BUF 改为 NAK 后更改此位, 就必须在确认 CSSTS 位为 “0” 并且 PBUSY 位为 “0” 后进行更改。如果此模块已将 PID 位改为 NAK, 就不需要确认 PBUSY 位。</p>
8	SQCLR	0	R/W*	<p>交替位清除 在要将相应管道的下一个事务的顺序交替位的期待值清除为 DATA0 时, 将此位置 “1”。</p> <p>0: 无效 1: 指定 DATA0</p> <p>如果将此位置 “1”, 此模块就将相应管道的顺序交替位的期待值设定为 DATA0。此模块总是将此位置 “0”。</p> <p>当选择主机控制器功能时, 对于 Bulk-Out 传送管道, 如果将此位置 “1”, 此模块就从 PING 权标开始相应管道的下一次传送。</p> <p>必须在 CSSTS 位为 “0” 并且 PID 位为 NAK 时将 SQCLR 位置 “1”。</p> <p>如果要在将对应管道的 PID 位从 BUF 改为 NAK 后将此位置 “1”, 就必须在确认 CSSTS 位为 “0” 并且 PBUSY 位为 “0” 后进行更改。如果此模块已将 PID 位改为 NAK, 就不需要确认 PBUSY 位。</p>
7	SQSET	0	R/W*	<p>交替位置位 在要将相应管道的下一个事务的顺序交替位的期待值设定为 DATA1 时, 将此位置 “1”。</p> <p>0: 无效 1: 指定 DATA1</p> <p>如果将此位置 “1”, 此模块就将相应管道的顺序交替位的期待值设定为 DATA1。此模块总是将此位置 “0”。</p> <p>必须在 CSSTS 位为 “0” 并且 PID 位为 NAK 时将 SQSET 位置 “1”。</p> <p>如果要在将对应管道的 PID 位从 BUF 改为 NAK 后将此位置 “1”, 就必须在确认 CSSTS 位为 “0” 并且 PBUSY 位为 “0” 后进行更改。如果此模块已将 PID 位改为 NAK, 就不需要确认 PBUSY 位。</p>
6	SQMON	0	R	<p>交替位确认 表示相应管道的下一个事务的顺序交替位的期待值。 0: DATA0 1: DATA1</p> <p>当相应管道的传送类型不是等时传送时, 如果事务被正常处理, 此模块就将此位取反。如果在进行接收方向的传送时 DATA-PID 不同, 就不将此位取反。</p>

位	位名	初始值	R/W	说明
5	PBUSY	0	R	<p>管道忙</p> <p>表示在当前事务中是否使用相应管道。</p> <p>0: 在事务中未使用相应管道</p> <p>1: 在事务中使用相应管道</p> <p>在开始相应管道的 USB 事务时, 此模块将此位从“0”改为“1”; 在 1 个事务正常结束时, 将此位从“1”改为“0”。</p> <p>在将 PID 位设定为 NAK 后, 能通过读此位来确认是否能更改管道的设定。详细内容请参照“29.4.3(1) 管道控制寄存器的切换步骤”。</p>
4 ~ 2	—	全 0	R	<p>保留位</p> <p>读写值都为“0”。</p>
1、0	PID[1:0]	00	R/W	<p>应答 PID</p> <p>指定相应管道的下一个事务的应答方法。</p> <p>00: NAK 应答</p> <p>01: BUF 应答 (取决于缓冲器状态)</p> <p>10: STALL 应答</p> <p>11: STALL 应答</p> <p>这些位的默认值为 NAK。在通过相应管道进行 USB 传送时, 必须将这些位改为 BUF。各 PID 位的设定值所对应的此模块基本运行 (通信包中没有错误时的运行) 如表 29.12 和表 29.13 所示。</p> <p>当相应管道正在进行 USB 通信时, 如果将这些位从 BUF 改为 NAK, 就必须在写 NAK 后确认 PBUSY 位为“0”, 以便确认该管道的 USB 传送转移到 NAK 状态。如果此模块已将这些位改为 NAK, 就不需要确认 PBUSY 位。即使在对应管道发行分离事务的 S-Split 后 (CSSTS 位为“1”) 将这些位改为 NAK, 也在 C-Split 结束前执行事务。</p> <p>在以下情况下, 此模块更改这些位的值:</p> <ul style="list-style-type: none"> 当相应管道为接收方向并且已将所选管道的 SHTNAK 位置“1”时, 如果此模块认为传送结束, 就将 PID 位设定为 NAK。 对于相应管道, 如果接收到超过 MaxPacketSize 有效负载的数据包, 此模块就将 PID 位设定为 STALL (11)。 当选择功能控制器功能时, 如果检测到 USB 总线复位, 此模块就将 PID 位设定为 NAK。 当选择主机控制器功能时, 如果连续 3 次检测到 CRC 错误等接收错误, 此模块就将 PID 位设定为 NAK。 当选择主机控制器功能时, 如果接收到 STALL 握手信号, 此模块就将 PID 位设定为 STALL (11)。 <p>必须按照以下步骤设定这些位:</p> <ul style="list-style-type: none"> 在从 NAK (00) 状态转移到 STALL 状态时, 必须写“10”。 在从 BUF (01) 状态转移到 STALL 状态时, 必须写“11”。 在从 STALL (11) 状态转移到 NAK 状态时, 必须在写“10”后写“00”。 在从 STALL 状态转移到 BUF 状态时, 必须在改为 NAK 状态后设定为 BUF 状态。

【注】 * 读取值为“0”, 只能写“1”。

表 29.10 BSTS 位的操作

DIR 位	BFRE 位	DCLRM 位	BSTS 位的含义
0	0	0	在能从 FIFO 缓冲器读接收数据时，此位为“1”；在读完数据时，此位为“0”。
		1	禁止设定此组合。
	1	0	在能从 FIFO 缓冲器读接收数据时，此位为“1”；在读完数据后给 BCLR 位写“1”时此位为“0”。
		1	在能从 FIFO 缓冲器读接收数据时，此位为“1”；在读完数据时，此位为“0”。
1	0	0	在能将发送数据写到 FIFO 缓冲器时，此位为“1”；在写完数据时，此位为“0”。
		1	禁止设定此组合。
	1	0	禁止设定此组合。
		1	禁止设定此组合。

表 29.11 (1) ACLRM 位为“1”时此模块要清除的内容

序号	通过 ACLRM 位操作清除的内容
1	分配给相应管道的 FIFO 缓冲器的全部内容（在设定为双缓冲器时，清除 2 个 FIFO 缓冲器面）
2	当相应管道的传送类型为等时传送时，清除间隔计数值。

表 29.11 (2) ACLRM 位为“1”时需要清除的情况

序号	需要清除的情况
1	要清除分配给相应管道的 FIFO 缓冲器的全部内容时
2	要对间隔计数值进行复位时
3	要更改 BFRE 位的设定值时
4	要更改 DBLB 位的设定值时
5	在强制结束事务计数功能时

表 29.12 PID 位对应的此模块运行一览表（选择主机控制器功能的情况）

PID 位	传送类型	传送方向 (DIR 位)	此模块的运行
00 (NAK)	与设定值无关	与设定值无关	不发行权标。
01 (BUF)	批量或者中断	与设定值无关	如果将 UACT 位置“1”并且相应管道对应的 FIFO 缓冲器为可发送和接收的状态，就发行权标。 如果将 UACT 位置“0”或者为不可发送和接收的状态，就不发行权标。
	等时	与设定值无关	与相应管道对应的 FIFO 缓冲器的状态无关，发行权标。
10 (STALL) 或者 11 (STALL)	与设定值无关	与设定值无关	不发行权标。

表 29.13 PID 位对应的此模块运行一览表（选择功能控制器功能的情况）

PID 位	传送类型	传送方向 (DIR 位)	此模块的运行
00 (NAK)	批量或者中断	与设定值无关	对 USB 主机的权标进行 NAK 应答。 但是, 有关将 ATREPM 位置“1”时的运行请参照 ATREPM 位的说明。
	等时	接收方向 (DIR 位为“0”)	对 USB 主机的权标不进行应答。
		发送方向 (DIR 位为“1”)	对 USB 主机的权标发送 Zero-Length 包。
01 (BUF)	批量	接收方向 (DIR 位为“0”)	对于 USB 主机的 OUT 权标, 如果相应管道对应的 FIFO 缓冲器为可接收的状态, 就接收数据并且进行 ACK 应答; 如果不是可接收的状态, 就进行 NAK 应答。 对于 USB 主机的 PING 权标, 如果相应管道对应的 FIFO 缓冲器为可接收的状态, 就进行 ACK 应答; 如果不是可接收的状态, 就进行 NYET 应答。
		中断	接收方向 (DIR 位为“0”)
	批量或者中断	发送方向 (DIR 位为“1”)	如果对应的 FIFO 缓冲器为可发送的状态, 就对 USB 主机的权标发送数据; 如果不是可发送的状态, 就进行 NAK 应答。
	等时	接收方向 (DIR 位为 0)	对于 USB 主机的 OUT 权标, 如果相应管道对应的 FIFO 缓冲器为可接收的状态, 就接收数据; 如果不是可接收的状态, 就放弃数据。
		发送方向 (DIR 位为“1”)	如果对应的 FIFO 缓冲器为可发送的状态, 就对 USB 主机的权标发送数据; 如果不是可发送的状态, 就发送 Zero-Length 包。
	10 (STALL) 或者 11 (STALL)	批量或者中断	与设定值无关
	等时	与设定值无关	对 USB 主机的权标不进行应答。

(2) PIPEnCTR (n=6 ~ 9)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	—	CSCLR	CSSTS	—	—	ACLRM	SQCLR	SQSET	SQMON	PBUSY	—	—	—	PID[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W*	R/W	R	R	R/W	R/W*	R/W*	R	R	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15	BSTS	0	R	缓冲器状态 表示相应管道的 FIFO 缓冲器状态。 0: 不能存取缓冲器 1: 能存取缓冲器 如表 29.10 所示, 此位的含义因 DIR 位、BFRE 位和 DCLRM 位的设定值而不同。
14	—	0	R	保留位 读写值都为“0”。
13	CSCLR	0	R/W*	CSPLIT 状态清除位 通过将此位置“1”, 清除相应管道的 CSSTS 位。 0: 写操作无效 1: 清除 CSSTS 位 在使用分离事务的传送中, 如果要强制从 S-SPLIT 重新开始下一次传送, 就必须将此位置“1”。正常的分离事务在 C-SPLIT 结束时, 此模块自动将 CSSTS 位清“0”, 因此不需要进行清除处理。 必须在通过将 UACT 位置“0”来停止通信时或者在检测到断开时尚未传送的情况下, 通过此位控制 CSSTS 位。 在 CSSTS 位为“0”时, 即使将此位置“1”, CSSTS 位也保持“0”。 当选择功能控制器功能时, 必须给此位写“0”。
12	CSSTS	0	R/W	当选择主机控制器功能时, 此位表示分离事务的 C-SPLIT 状态。 0: 正在进行 START-SPLIT (S-SPLIT) 事务处理或者正在进行未使用分离事务的传送 1: 正在进行 C-SPLIT 事务处理 在 C-SPLIT 开始时, 此模块将此位置“1”; 在检测到 C-SPLIT 结束时, 此模块将此位置“0”。 此位只在选择主机控制器功能时有效。
11、10	—	全 0	R	保留位 读写值都为“0”。
9	ACLRM	0	R/W	缓冲器自动清除模式 指定禁止或者允许相应管道的缓冲器自动清除模式。 0: 禁止缓冲器自动清除模式 1: 允许缓冲器自动清除模式 (对全部缓冲器进行初始化) 如果要将分配给相应管道的 FIFO 缓冲器内容全部清除, 就必须给 ACLRM 位连续写“1”和“0”。 在给此位连续写“1”和“0”时, 此模块要清除的内容如表 29.14 (1) 所示, 需清除的情况如表 29.14 (2) 所示。 必须在 CSSTS 位为“0”并且 PID 位为 NAK 而未给 CURPIPE 位设定相应管道时更改此位。 如果要在将对应管道的 PID 位从 BUF 改为 NAK 后更改此位, 就必须在确认 CSSTS 位为“0”并且 PBUSY 位为“0”后进行更改。如果此模块已将 PID 位改为 NAK, 就不需要确认 PBUSY 位。

位	位名	初始值	R/W	说明
8	SQCLR	0	R/W*	<p>交替位清除</p> <p>在要将相应管道的下一个事务的顺序交替位的期待值清除为 DATA0 时，将此位置“1”。</p> <p>0: 无效 1: 指定 DATA0</p> <p>如果将此位置“1”，此模块就将相应管道的顺序交替位的期待值设定为 DATA0。此模块总是将此位置“0”。</p> <p>当选择主机控制器功能时，对于 Bulk-OUT 传送管道，如果将此位置“1”，此模块就从 PING 权标开始相应管道的下一次传送。</p> <p>必须在 CSSTS 位为“0”并且 PID 位为 NAK 时将 SQCLR 位置“1”。</p> <p>如果要在将对应管道的 PID 位从 BUF 改为 NAK 后将此位置“1”，就必须在确认 CSSTS 位为“0”并且 PBUSY 位为“0”后进行更改。如果此模块已将 PID 位改为 NAK，就不需要确认 PBUSY 位。</p>
7	SQSET	0	R/W*	<p>交替位置位</p> <p>在要将相应管道的下一个事务的顺序交替位的期待值设定为 DATA1 时，将此位置“1”。</p> <p>0: 无效 1: 指定 DATA1</p> <p>如果将此位置“1”，此模块就将相应管道的顺序交替位的期待值设定为 DATA1。此模块总是将此位置“0”。</p> <p>必须在 CSSTS 位为“0”并且 PID 位为 NAK 时将 SQSET 位置“1”。</p> <p>如果要在将对应管道的 PID 位从 BUF 改为 NAK 后将此位置“1”，就必须在确认 CSSTS 位为“0”并且 PBUSY 位为“0”后进行更改。如果此模块已将 PID 位改为 NAK，就不需要确认 PBUSY 位。</p>
6	SQMON	0	R	<p>交替位确认</p> <p>表示相应管道的下一个事务的顺序交替位的期待值。</p> <p>0: DATA0 1: DATA1</p> <p>在相应管道的传送类型不是等时传送时，如果事务被正常处理，此模块就将此位取反。如果在进行接收方向的传送时 DATA-PID 不同，就不将此位取反。</p>
5	PBUSY	0	R	<p>管道忙</p> <p>表示在当前 USB 总线上是否使用相应管道。</p> <p>0: 在 USB 总线上未使用该管道 1: 在 USB 总线上使用该管道</p> <p>当开始该管道的 USB 事务时，此模块将此位从“0”改为“1”；当 1 个事务正常结束时，将此位从“1”改为“0”。</p> <p>在将 PID 位设定为 NAK 后，能通过读此位来确认是否能更改管道的设定。</p>
4 ~ 2	—	全 0	R	<p>保留位</p> <p>读写值都为“0”。</p>

位	位名	初始值	R/W	说明
1、0	PID[1:0]	00	R/W	<p>应答 PID</p> <p>指定相应管道的下一个事务的应答方法。</p> <p>00: NAK 应答</p> <p>01: BUF 应答 (取决于缓冲器状态)</p> <p>10: STALL 应答</p> <p>11: STALL 应答</p> <p>这些位的默认值为 NAK。在通过相应管道进行 USB 传送时, 必须将这些位改为 BUF。各 PID 位的设定值所对应的此模块基本运行 (通信包中没有错误时的运行) 如表 29.12 和表 29.13 所示。</p> <p>当相应管道正在进行 USB 通信时, 如果将这些位从 BUF 改为 NAK, 就必须在写 NAK 后确认 PBUSY 位为 “0”, 以便确认该管道的 USB 传送转移到 NAK 状态。如果此模块已将这些位改为 NAK, 就不需要确认 PBUSY 位。即使在对相应管道发行分离事务的 S-Split 后 (CSSTS 位为 “1”) 将这些位改为 NAK, 也在 C-Split 结束前执行事务。</p> <p>在以下情况下, 此模块更改这些位的值:</p> <ul style="list-style-type: none"> 在相应管道为接收方向并且已将所选管道的 SHTNAK 位置 “1” 时, 如果此模块认为传送结束, 就将 PID 位设定为 NAK。 对于相应管道, 如果接收到超过 MaxPacketSize 有效负载的数据包, 此模块就将 PID 位设定为 STALL (11)。 当选择功能控制器功能时, 如果检测到 USB 总线复位, 此模块就将 PID 位设定为 NAK。 当选择主机控制器功能时, 如果连续 3 次检测到 CRC 错误等接收错误, 此模块就将 PID 位设定为 NAK。 当选择主机控制器功能时, 如果接收到 STALL 握手信号, 此模块就将 PID 位设定为 STALL (11)。 <p>必须按照以下步骤设定这些位:</p> <ul style="list-style-type: none"> 在从 NAK (00) 状态转移到 STALL 状态时, 必须写 “10”。 在从 BUF (01) 状态转移到 STALL 状态时, 必须写 “11”。 在从 STALL (11) 状态转移到 NAK 状态, 必须在写 “10” 后写 “00”。 在从 STALL 状态转移到 BUF 状态时, 必须在改为 NAK 状态后转移到 BUF 状态。

【注】 * 读取值为 “0”, 只能写 “1”。

表 29.14 (1) ACLRM 位为 “1” 时此模块要清除的内容

序号	通过 ACLRM 位操作清除的内容
1	分配给相应管道的 FIFO 缓冲器的全部内容
2	当选择主机控制器功能时并且所选管道的传送类型为中断传送时, 清除间隔计数值。

表 29.14 (2) ACLRM1 位为 “1” 时需要清除的情况

序号	需要清除的情况
1	要清除分配给相应管道的 FIFO 缓冲器的全部内容时
2	要对间隔计数值进行复位时
3	更改 BFRE 位的设定值时
4	强制结束事务计数功能时

29.3.37 管道 n 事务计数允许寄存器 (PIPE_nTRE) (n=1 ~ 5)

PIPE_nTRE 指定对应管道 1 ~ 5 的事务计数器是否有效以及指定计数器的清除。
在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TRENB	TRCLR	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W*1	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9	TRENB	0	R/W	事务计数器允许 指定事务计数器的无效或者有效。 0: 事务计数功能无效 1: 事务计数功能有效 对于接收管道，如果在给 TRNCNT 位设定总信息包的数量后将此位置“1”，此模块就在接收完 TRNCNT 位设定数量的信息包后进行以下的控制： <ul style="list-style-type: none"> 在使用连续发送和接收模式（将 CNTMD 位置“1”）时，即使在接收结束时 FIFO 缓冲器未滿，也切换给 CPU 侧。 在 SHTNAK 位为“1”时，在接收完 TRNCNT 位设定数量的信息包时将对应管道的 PID 位改为 NAK。 在 BFRE 位为“1”时，在接收完 TRNCNT 位设定数量的信息包并且读完最后的数据时将 BRDY 中断置为有效。 对于发送管道，必须将此位置“0”。 在不使用事务计数功能时，必须将此位置“0”。 在使用事务计数功能时，必须在将此位置“1”前设定 TRNCNT 位。另外，必须在接收事务计数对象的最初信息包前将此位置“1”。
8	TRCLR	0	R/W*1	事务计数器清除 清除相应管道对应的事务计数器的当前计数值并且将此位置“0”。 0: 无效 1: 清除当前计数器
7 ~ 0	—	全 0	R	保留位 读写值都为“0”。

【注】 *1 读取值为“0”，只能写“1”。

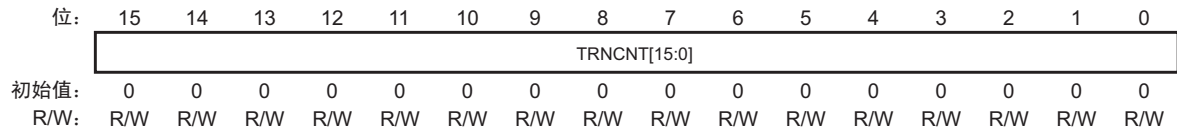
*2 必须在 CSSTS 位为“0”并且 PID 位为 NAK 时更改此寄存器的各位。

如果要在将对应管道的 PID 位从 BUF 改为 NAK 后更改各位的设定值，就必须在确认 CSSTS 位为“0”并且 PBUSY 位为“0”后进行更改。如果此模块已将 PID 位改为 NAK，就不需要确认 PBUSY 位。

29.3.38 管道 n 事务计数寄存器 (PIPE_nTRN) (n=1 ~ 5)

PIPE_nTRN 是对应管道 1 ~ 5 的事务计数器。

在上电复位时，对此寄存器进行初始化；在 USB 总线复位时，PIPE_nTRN 保持设定值。



位	位名	初始值	R/W	说明
15 ~ 0	TRNCNT [15:0]	全 0	R/W	<p>事务计数器</p> <p>写时: 设定 DMA 传送的事务次数。</p> <p>读时: 当 TREN_B 位为 “0” 时，表示设定的事务次数。 当 TREN_B 位为 “1” 时，表示正在计数的事务次数。 如果接收时的状态满足以下全部条件，此模块就将此位的值加 1。</p> <ul style="list-style-type: none"> • TREN_B 位为 “1”。 • 在接收信息包时，TRC_{NT} 设定值不等于当前的计数值 +1。 • 接收的信息包的有效负载和 MXPS 位的设定值相同。 <p>在满足以下条件时，此模块将这些位清 “0”。</p> <ul style="list-style-type: none"> • 当满足以下全部条件时 TREN_B 位为 “1”。 在接收信息包时，TRC_{NT} 设定值等于当前的计数值 +1)。 接收的信息包的有效负载和 MXPS 位的设定值相同。 • 当满足以下全部条件时 TREN_B 位为 “1”。 接收到短包。 • 当满足以下条件时 将 TRCLR 位置 “1”。 <p>对于发送管道，必须将这些位置 “0”。</p> <p>在不使用事务计数功能时，必须将这些位置 “0”。</p> <p>必须在 CSSTS 位为 “0” 并且 PID 位为 NAK 并且 TREN_B 位为 “0” 时更改这些位。</p> <p>如果要在将对应管道的 PID 位从 BUF 改为 NAK 后更改此位，就必须在确认 CSSTS 位为 “0” 并且 P_BUSY 位为 “0” 后进行更改。如果此模块已将 PID 位改为 NAK，就不需要确认 P_BUSY 位。</p> <p>在更改这些位的值时，必须在将 TREN_B 位置 “1” 前将 TRC_{NT} 位置 “1”。</p>

29.3.39 设备地址 n 的配置寄存器 (DEVADDn) (n=0 ~ A)

对于管道 0 ~ A，DEVADDn 指定连接通信对象外围设备的 HUB 地址、端口号和通信速度。

当选择主机控制器功能时，必须在开始各管道的通信前设定此寄存器的各位。

必须在不存在使用各位设定的有效管道时更改此寄存器的各位。有效管道是指满足以下 2 个条件的管道：

1. 在通过设定 DEVSEL 位来指定此寄存器时
2. 在将所选管道的 PID 位设定为 BUF 或者在所选管道为 DCP 并且 SUREQ 位为 “1” 时
在上电复位时，对此寄存器进行初始化。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	UPPHUB[3:0]				HUBPORT[2:0]			USBSPD[1:0]		—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为 “0”。
14 ~ 11	UPPHUB [3:0]	0000	R/W	通信对象连接 HUB 寄存器 设定连接通信对象外围设备的 HUB 的 USB 地址。 0000: 外围设备直接连接本 LSI 的端口 0001 ~ 1010: HUB 的 USB 地址 1011 ~ 1111: 禁止设定 当选择主机控制器功能时，此模块在执行分离事务时参照这些位的设定值来生成信息包。 当选择功能控制器功能时，必须设定为 “0000”。
10 ~ 8	HUBPORT [2:0]	000	R/W	通信对象连接 HUB 端口 设定连接通信对象外围设备的 HUB 的端口号。 000: 外围设备直接连接本 LSI 的端口 001 ~ 111: HUB 的端口号 当选择主机控制器功能时，此模块在执行分离事务时参照这些位的设定值来生成信息包。 当选择功能控制器功能时，必须设定为 “000”。
7、6	USBSPD [1:0]	00	R/W	通信对象设备的传送速度 设定通信对象外围设备的 USB 传送速度。 00: 未使用 DEVADDn 寄存器 01: 低速 10: 全速 11: 高速 当选择主机控制器功能时，此模块参照这些位的设定值来生成信息包。 当选择功能控制器功能时，必须设定为 “00”。
5 ~ 0	—	全 0	R	保留位 读写值都为 “0”。

29.4 运行说明

29.4.1 系统控制和振荡控制

本节说明此模块的初始设定所需的寄存器操作以及功耗控制所需的寄存器。

(1) 复位

此模块的复位种类一览表如表 29.15 所示，有关各种复位运行后的寄存器初始状态，请参照“29.3 寄存器说明”。

表 29.15 复位种类一览表

名称	操作
上电复位	从 $\overline{\text{RES}}$ 引脚输入低电平。
USB 总线复位	当选择功能控制器功能时，此模块从 D+ 线路和 D- 线路开始自动检测。

(2) 控制器功能的选择设定

此模块能选择主机控制器功能或者功能控制器功能。必须通过 SYSCFG 寄存器的 DCFM 位选择控制器功能。必须在上电复位后的初始设定或者禁止 D+ 上拉（DPRPU 位为“0”）并且禁止 D+/D- 下拉（DRPD 位为“0”）的状态下设定 DCFM 位。

(3) 高速运行的允许

此模块能设定 USB 通信速度（通信位速率）。

当选择主机控制器功能时，能设定为高速运行或者全速 / 低速运行；当选择功能控制器功能时，能选择高速运行或者全速运行。如果通过此模块允许高速运行，就必须将 SYSCFG 寄存器的 HSE 位置“1”。在允许高速运行时，此模块执行复位握手信号协议，自动设定 USB 通信速度。能通过 DVSTCTR 寄存器的 RHST 位确认复位握手信号的结果。

在禁止高速运行的情况下，当选择主机控制器功能时，此模块以全速或者低速运行；当选择功能控制器功能时，此模块只以全速运行。

当选择主机控制器功能时，必须在检测到断开后并且在进行总线复位前更改 HSE 位；当选择功能控制器功能时，必须在禁止 D+ 上拉（DPRPU 位为“0”）的状态下更改 HSE 位。

(4) USB 数据总线的电阻控制

此模块和 USB 连接器的连接图如图 29.1 所示。

此模块内置 D+ 信号的上拉电阻和 D+/D- 信号的下拉电阻。必须通过设定 SYSCFG 寄存器的 DPRPU 位和 DRPD 位，设定各信号的上拉和上拉。

此模块对 D+ 信号和 D- 信号进行高速运行时的终端电阻和全速运行时的输出电阻进行控制。此模块通过复位握手信号、挂起和恢复的检测，自动切换和主机控制器或者外围设备连接后的内部电阻。

如果选择功能控制器功能并且在与主机控制器进行通信的过程中将 SYSCFG 寄存器的 DPRPU 位置“0”，就禁止 USB 数据线的上拉电阻（或者终端电阻），因此能将“设备断开”通知 USB 主机。

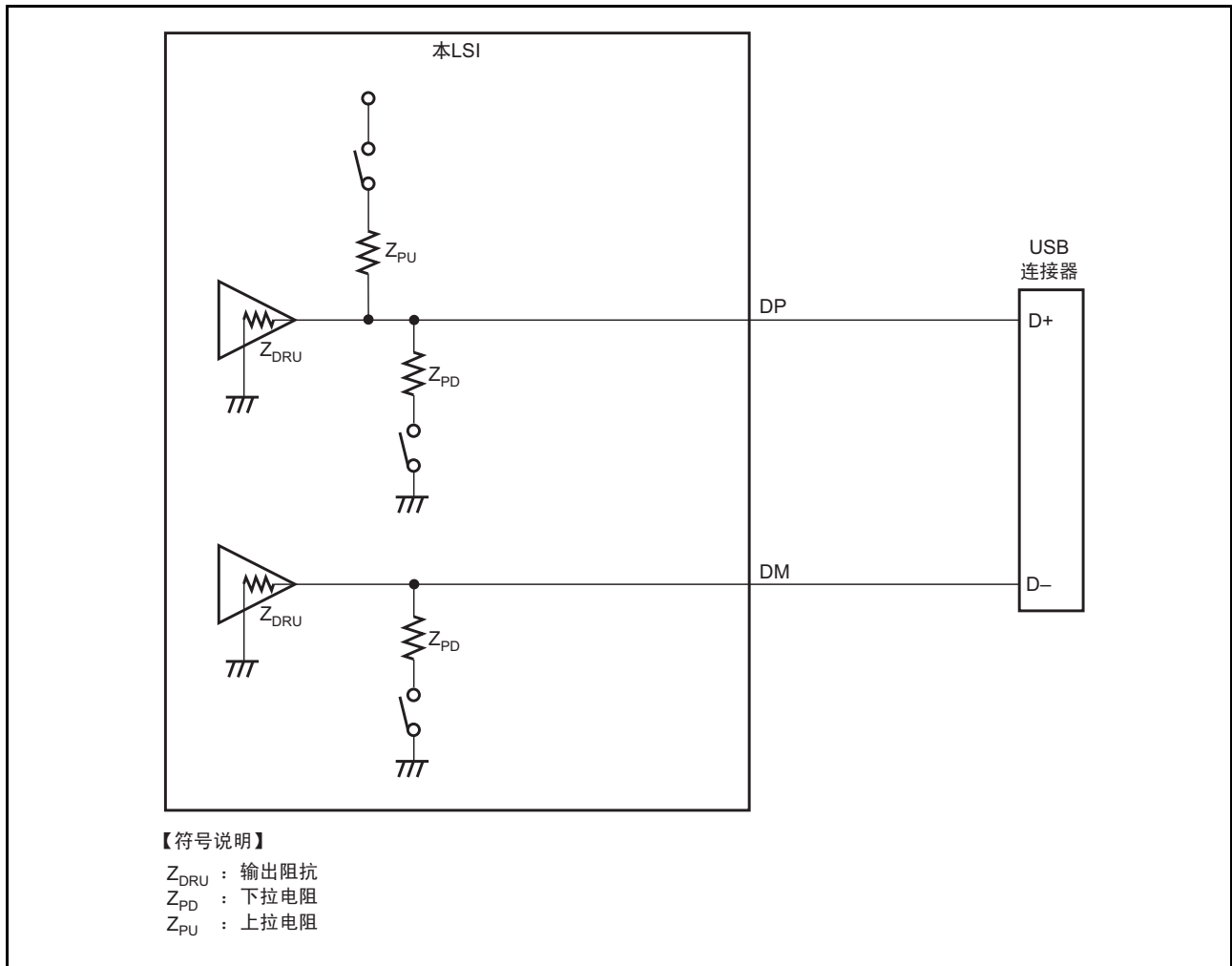


图 29.1 USB 连接器的连接图

(5) 寄存器的存取等待控制

此模块的 SYSSTS 以后的寄存器存取周期有以下限制：

等待限制：此模块的寄存器的连续存取周期必须至少为 4 个 USB 时钟（48MHz）周期的期间（83.33ns）。

为了满足此限制，需要通过 BUSWAIT 寄存器的 BWAIT[3:0] 位控制寄存器的存取等待。初始值为最大值（存取周期为 17 个时钟周期），所以必须选择最佳的设定值。

设定例子 1：连续存取此模块的寄存器的情况

外围时钟 1 的频率：66.67MHz

计算式：(2 个周期 (此模块的寄存器的存取周期) + 1 个周期 (连续存取的间隔期间) + BWAIT) × 1 / 66.67MHz
 $\geq 83.33\text{ns}$

BWAIT = 3

设定例子 2：通过 DMA 传送将数据从高速内部 RAM 传送到 FIFO 端口寄存器的情况

外围时钟 1 的频率：66.67MHz

计算式：(2 个周期 (此模块的寄存器的存取周期) + 2 个周期 (高速内部 RAM 的存取周期) + BWAIT) × 1 /
 $66.67\text{MHz} \geq 83.33\text{ns}$

BWAIT = 2

(6) 输入时钟的选择设定

此模块能选择USB_X1或者EXTAL为输入引脚并且选择48MHz或者12MHz为输入时钟。必须通过SYSCFG寄存器的 UCKPSEL 位和 UCKFSEL 位选择输入引脚和输入频率。必须在停止给 USB 模块提供时钟的状态下 (SCKE 位为 “0”) 设定 UCKPSEL 位和 UCKFSEL 位。

(7) USB 模块的时钟供给设定

给此模块提供时钟的设定方法因时钟输入频率的选择而不同。

(a) 选择 48MHz 输入的情况

必须将 SYSCFG 寄存器的 UCKFSEL 位和 UCKPSEL 位分别置 “0”，在选择 48MHz 输入和 USB_X1 后，通过 SCKE 位进行时钟供给的设定。在本 LSI 中，只在时钟输入引脚为 USB_X1 的情况下，48MHz 输入的指定才有效。

(b) 选择 12MHz 输入的情况

必须将 SYSCFG 寄存器的 UCKFSEL 位置 “1”，在通过 UCKPSEL 位选择时钟输入引脚后，按照以下步骤进行时钟供给的设定：

设定例子 1：允许从上电复位后的初始设定开始提供时钟的情况

1. 将 UPLLE 位置 “1”。
2. 等待 1ms。
3. 将 SCKE 位置 “1”。

设定例子 2：在挂起时停止提供时钟的情况

1. 将 SCKE 位置 “0”。
2. 将 UPLLE 位置 “0”。

设定例子 3：在从挂起恢复时允许提供时钟的情况

1. 将 UPLLE 位置 “1”。
2. 等待 1ms。
3. 将 SCKE 位置 “1”。

【注】 当选择功能控制器功能并且选择高速运行时，如果通过USB复位从挂起恢复，就必须在2.5ms内将SCKE位置“1”。

29.4.2 中断功能

此模块的中断发生条件一览表如表 29.16 所示。

在这些中断发生条件成立并且通过对应的中断允许寄存器设定为允许中断输出时，此模块向中断控制器发行 USB 中断请求。

表 29.16 中断发生条件一览表

位	中断名	中断发生条件	产生的功能	相关状态
VBINT	VBUS 中断	<ul style="list-style-type: none"> 当检测到 VBUS 输入引脚的状态变化时 (L 电平 → H 电平或者 H 电平 → L 电平) 	主机 / 功能	VBSTS
RESM	恢复中断	<ul style="list-style-type: none"> 在挂起状态下检测到 USB 总线的状态变化时 (J-State → K-State 或者 J-State → SE0) 	功能	—
SOFR	帧号更新中断	<ul style="list-style-type: none"> [选择主机控制器功能的情况] 当发送帧号不同的 SOF 包时 [选择功能控制器功能的情况] 当接收帧号不同的 SOF 包时 	主机 / 功能	—
DVST	设备状态转移中断	<ul style="list-style-type: none"> 当检测到设备状态的转移时 USB 总线复位的检测 挂起状态的检测 SET_ADDRESS 请求的接收 SET_CONFIGURATION 请求的接收 	功能	DVSQ
CTRT	控制传送阶段转移中断	<ul style="list-style-type: none"> 当检测到控制传送的阶段转移时 设置阶段结束 控制写传送的状态阶段转移 控制读传送的状态阶段转移 控制传送结束 控制传送顺序错误的发生 	功能	CTSQ
BEMP	缓冲器空中断	<ul style="list-style-type: none"> 在发送缓冲存储器中的全部数据后缓冲器变空时 当接收到超过最大信息包长度的信息包时 	主机 / 功能	BEMPSTS. PIPEBEMP
NRDY	缓冲器未就绪中断	<ul style="list-style-type: none"> [选择主机控制器功能的情况] 对于发行的权标，接收到外围设备侧的 STALL 时 对于发行的权标，无法正常接收外围设备侧的应答时 (连续 3 次不应答或者连续 3 次发生信息包接收错误) 在等时传送过程中发生超限或者欠载时 [选择功能控制器功能的情况] 在 PID 位为 BUF 并且缓冲存储器为不能发送的状态下接收权标时 在等时传送的数据接收过程中发生 CRC 错误和位填充错误时 在等时传送的数据接收过程中发生间隔错误时 	主机 / 功能	NRDYSTS. PIPENRDY
BRDY	缓冲器就绪中断	<ul style="list-style-type: none"> 当缓冲器就绪 (可读写的状态) 时 	主机 / 功能	BRDYSTS. PIPEBRDY
BCHG	总线变化中断	<ul style="list-style-type: none"> 当检测到 USB 总线状态的变化时 	主机	—
DTCH	设备断开检测	<ul style="list-style-type: none"> 当检测到外围设备断开时 	主机	—
ATTCH	设备连接检测	<ul style="list-style-type: none"> 当检测到 USB 总线状态为连续 2.5μs 的 J-STATE 或者连续 2.5μs 的 K-STATE 时。能用于外围设备连接的检测。 	主机	—
EOFERR	EOF 错误检测	<ul style="list-style-type: none"> 当检测到外围设备的 EOF 错误时 	主机	—
SACK	设置正常	<ul style="list-style-type: none"> 当接收到设置事务的正常应答 (ACK) 时 	主机	—
SIGN	设置错误	<ul style="list-style-type: none"> 当连续 3 次检测到设置事务的错误 (不应答或者 ACK 包破损) 时 	主机	—

此模块的中断相关图如图 29.2 所示。

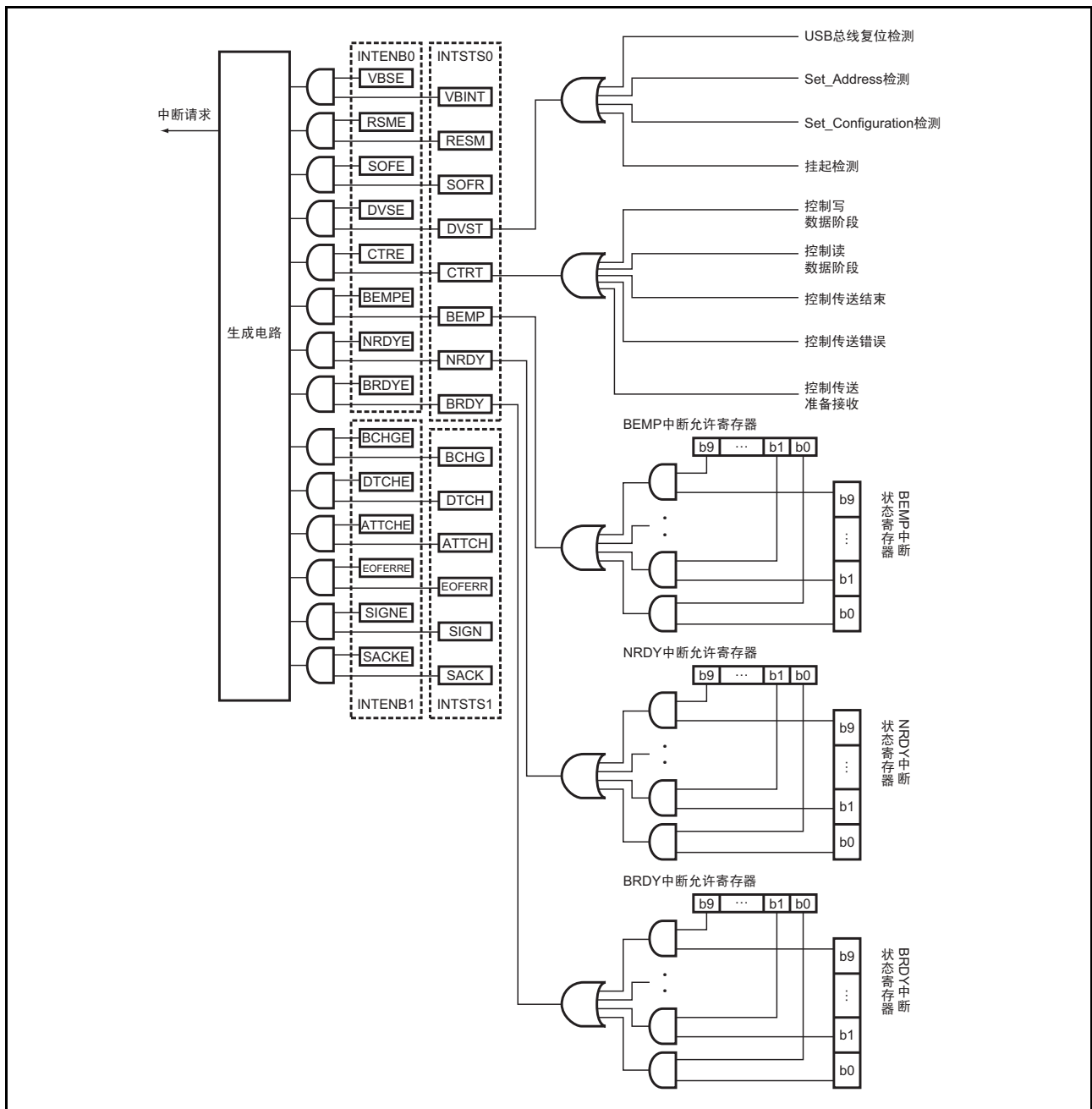


图 29.2 中断的相关图

(1) BRDY 中断

在选择主机控制器功能或者功能控制器功能时，都产生 BRDY 中断。当各管道满足以下条件时，此模块将 BRDYSTS 寄存器的对应位置“1”。此时，如果将相应管道对应的 BRDYENB 寄存器的 PIPEBRDYE 位置“1”并且将 INTENB0 寄存器的 BRDYE 位置“1”，此模块就产生 BRDY 中断。

BRDY 中断的发生条件和清除方法因 BRDYM 位和各管道的 BFRE 位的设定而不同。

(a) BRDYM 位为“0”并且 BFRE 位为“0”的情况

在此设定的情况下，BRDY 中断是表示能存取 FIFO 端口的中断。

在满足以下所示条件时，此模块产生内部 BRDY 中断请求的触发并且将产生请求触发的管道所对应的 PIPEBRDY 位置“1”。

1. 发送方向管道的情况

- 当将 DIR 位从“0”改为“1”时
- 当此模块在 CPU 不能写分配给相应管道的 FIFO 缓冲器的状态下（BSTS 位的读取值为“0”）发送完该管道的信息包时
如果设定为连续发送和接收模式，就在发送完一个 FIFO 缓冲器面的数据时产生请求触发。
- 在 FIFO 缓冲器为双缓冲器的情况下写完一个 FIFO 缓冲器而另一个 FIFO 缓冲器为空白
即使在写 FIFO 缓冲器的过程中另一个缓冲器发送结束，也不在当前正在写的缓冲器面写完数据前产生请求触发。
- 对于传送类型为等时传送的管道，当通过硬件进行缓冲器转储清除时
- 当通过给 ACLRM 位写“1”使 FIFO 缓冲器从不可写状态变为可写状态时
对于 DCP（即，控制传送时的数据发送），不产生请求触发。

2. 接收方向管道的情况

- 在 CPU 不能读分配给相应管道的 FIFO 缓冲器的状态下（BSTS 位的读取值为“0”时）接收正常到信息包并且 FIFO 缓冲器变为可读状态时
对于数据 PID 不同的事务，不产生请求触发。
在连续发送和接收模式的情况下，如果数据长度为 MaxPacketSize 并且缓冲器还有空间，就不产生请求触发。
如果接收到短包，即使 FIFO 缓冲器有空间，也产生请求触发。
在使用事务计数器的情况下，当接收设定数量的信息包时，产生请求触发。此时，即使 FIFO 缓冲器还有空间，也产生请求触发。
- 在 FIFO 缓冲器为双缓冲器的情况下读完一个 FIFO 缓冲器而另一个 FIFO 缓冲器也变为可读状态时
即使在读过程中另一个 FIFO 缓冲器接收结束，也不在当前正在读的缓冲器面读完数据前产生请求触发。

在选择功能控制器功能时的控制传送状态阶段进行通信时不产生此中断。

能通过给相应管道对应的 BRDYSTS 寄存器的 PIPEBRDY 位写“0”，将该管道的 PIPEBRDY 中断状态位清“0”。此时，必须给其他管道的对应位写“1”。

必须在存取 FIFO 缓冲器前清除此中断状态。

(b) BRDYM 位为“0”并且 BFRE 位为“1”的情况

在此设定的情况下，此模块在对接收管道读完一次传送的全部数据时判断为发生了 BRDY 中断，并且将此寄存器的相应管道的对应位置“1”。

在以下任意一种情况下，此模块判断为接收到一次传送的最后数据。

- 当接收到包括 Zero-Length 包的短包时
- 当使用事务计数器（TRNCNT 位）并且接收到 TRNCNT 位设定数量的信息包时

如果在满足上述判断条件后读完此数据，此模块就判断为读完一次传送的全部数据。

如果在 FIFO 缓冲器为空的状态下接收到 Zero-Length 包，此模块就在 FIFO 端口控制寄存器的 FRDY 位变为“1”并且 DTLN 位变为“0”时判断为读完一次传送的全部数据。此时，为了开始下一次传送，必须给对应的 FIFOCTR 寄存器的 BCLR 位写“1”。

在此设定的情况下，此模块不对发送管道检测 BRDY 中断。

能通过给相应管道对应的 PIPEBRDY 位写“0”，将该管道的 PIPEBRDY 中断状态位清“0”。此时，必须给其他管道的对应位写“1”。

在使用此模式时，不能在一次传送处理结束前更新 BFRE 位的设定值。

如果要在中途更改 BFRE 位，就必须通过 ACLRM 位清除全部对应管道的 FIFO 缓冲器。

(c) BRDYM 位为“1”并且 BFRE 位为“0”的情况

在此设定的情况下，PIPEBRDY 位的值和各管道的 BSTS 位连动。即，此模块根据 FIFO 缓冲器的状态，将 BRDY 中断状态位置“1”或者“0”。

1. 发送方向管道的情况

BRDY 中断状态在能给 FIFO 端口写数据的状态下为“1”，在不能写的状态下为“0”。

但是，即使 DCP 的发送管道为可写状态，也不产生 BRDY 中断。

2. 接收方向管道的情况

BRDY 中断状态在能从 FIFO 端口读数据的状态下为“1”，在读完全部数据（不可读状态）时为“0”。

如果在 FIFO 缓冲器为空的状态下接收到 Zero-Length 包，就在给 BCLR 位写“1”前该位为“1”，继续产生 BRDY 中断。

在此设定的情况下，不能将 PIPEBRDY 位清“0”。

当 BRDYM 位为“1”时，必须将 BFRE 位全部（全部管道）置“0”。

BRDY 中断发生时序图如图 29.3 所示。

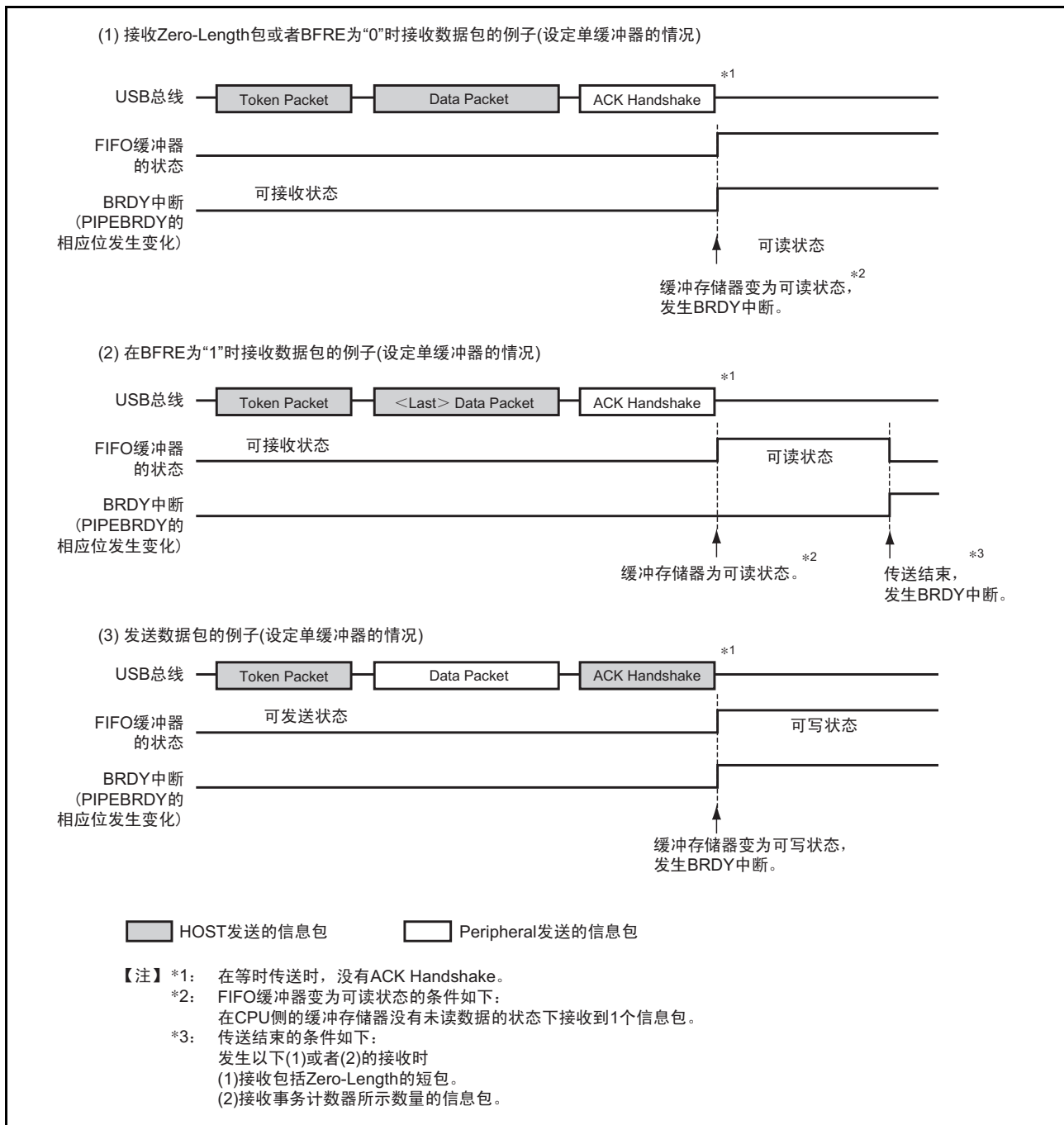


图 29.3 BRDY 中断发生时序图

(2) NRDY 中断

对于将 PID 位设定为 BUF 的管道，如果此模块发生内部 NRDY 中断请求，就将 NRDYSTS 寄存器的 PIPENRDY 位的对应位置“1”。此时，如果已将 NRDYENB 寄存器的对应位置“1”，此模块就将 INTSTS0 寄存器的 NRDY 位置“1”，产生 USB 中断。

对于某个管道，此模块产生内部 NRDY 中断请求的条件如下所示。

但是，在选择主机控制器功能并且执行设置事务时，不适合以下的中断发生条件。在选择主机控制器功能时的设置事务中，检测 SACK 中断或者 SIGN 中断。

在选择功能控制器功能并且执行控制传送状态阶段时，不产生中断请求。

(a) 选择主机控制器功能并且不发生分离事务的连接

1. 发送方向管道的情况

在满足以下任意条件时，此模块检测到 NRDY 中断。

- 对于传送类型为等时传送的管道，在 FIFO 缓冲器没有发送数据的状态下到了发行 OUT 权标时间时
此时，此模块在 OUT 权标后继续发送 Zero-Length 包，并且将 PIPENRDY 位的对应位置“1”，将 OVRN 位置“1”。
- 对于传送类型不是等时传送的管道并且在设置事务以外的通信中，连续 3 次发生外围设备不应答（在未检测到外围设备的握手信号包的情况下检测到超时）或者从外围设备的信息包中检测到错误时
此时，此模块将 PIPENRDY 位的对应位置“1”，并且将对应管道的 PID 位改为 NAK。
- 在设置事务以外的通信中从外围设备接收到 STALL 握手信号时（对于 OUT 和 PING 的 STALL）
此时，此模块将 PIPENRDY 位的对应位置“1”，并且将对应管道的 PID 位改为 STALL（11）。

2. 接收方向管道的情况

- 对于传送类型为等时传送的管道，在 FIFO 缓冲器中没有空间的状态下到了发行 IN 权标的时间时
此时，此模块放弃接收数据（对于 IN 权标）并且将该管道对应的 PIPENRDY 位置“1”，将 OVRN 位置“1”。
另外，如果从接收数据（对于 IN 权标）中检测到信息包错误，就将 CRCE 位置“1”。
- 对于传送类型不是等时传送的管道，连续 3 次发生外围设备对此模块发行的 IN 权标不应答（在未检测到外围设备的 DATA 包的情况下检测到超时）或者从外围设备的信息包中检测到错误时
此时，此模块将该管道对应的 PIPENRDY 位置“1”，并且将对应管道的 PID 位改为 NAK。
- 对于传送类型为等时传送的管道，外围设备对 IN 权标不应答（在未检测到外围设备的 DATA 包的情况下检测到超时）或者从外围设备的信息包中检测到错误时
此时，此模块将该管道对应的 PIPENRDY 位置“1”（不更改对应管道的 PID 位）。
- 对于传送类型为等时传送的管道，从接收的数据包中检测到 CRC 错误或者位填充错误时
此时，此模块将该管道对应的 PIPENRDY 位置“1”，并且将 CRCE 位置“1”。
- 当接收到 STALL 握手信号时
此时，此模块将相应管道对应的 PIPENRDY 位置“1”，并且将对应管道的 PID 位改为 STALL。

(b) 选择主机控制器功能并且发生分离事务的连接

1. 发送方向管道的情况

- 对于传送类型为等时传送的管道，在 FIFO 缓冲器没有发送数据的状态下到了发行 OUT 权标的时间时
此时，此模块在发行 Start-Split 事务（S-SPLIT）时将该管道对应的 PIPENRDY 位置“1”，并且将 OVRN 位置“1”。在 OUT 权标后继续发送 Zero-Length 包。
- 对于传送类型不是等时传送的管道，连续 3 次发生 HUB 对 S-SPLIT 或者 Complete-Split 事务（C-SPLIT）不应答（在未检测到 HUB 的握手信号包的情况下检测到超时）或者从 HUB 的信息包中检测到错误时
此时，此模块将该管道对应的 PIPENRDY 位置“1”，并且将对应管道的 PID 位改为 NAK。
如果在发行 C-SPLIT 时检测到 NRDY 中断，此模块就将 CSSTS 位清“0”。

- 对于C-SPLIT, 接收到STALL握手信号时
此时, 此模块将相应管道对应的 PIPENRDY 位置 “1”, 将对应管道的 PID 位改为 STALL (11) 并且将 CSSTS 位清 “0”。
但是在设置事务中, 不检测此中断。
 - 对于传送类型为中断传送的管道的 microFrame 号为 “4” 的 C-SPLIT, 接收到NYET时
此时, 此模块将该管道对应的 PIPENRDY 位置 “1”, 并且将 CSSTS 位清 “0” (不更改对应管道的 PID 位)。
2. 接收方向管道的情况
- 对于传送类型为等时传送的管道, 在 FIFO 缓冲器没有空间的状态下到了发行 IN 权标的时间时
此时, 此模块在发行 S-SPLIT 时将该管道对应的 PIPENRDY 位置 “1”, 并且将 OVRN 位置 “1”。另外, 放弃接收数据 (对于 IN 权标)。
 - 对于传送类型为批量传送的管道的传送或者 DCP 的设置事务以外的传送, 连续 3 次发生以下任意情况的组合: 在发行 S-SPLIT 或者 C-SPLIT 时 HUB 对此模块发行的 IN 权标不应答 (在未检测到 HUB 的 DATA 包的情况下检测到超时) 或者从 HUB 的信息包检测到错误时
此时, 此模块将该管道对应的 PIPENRDY 位置 “1”, 并且将对应管道的 PID 位改为 NAK。如果在 C-SPLIT 中产生此条件, 此模块就将 CSSTS 位清 “0”。
 - 对于传送类型为等时传送或者中断传送的管道的 C-SPLIT, 连续 3 次发生 HUB 对此模块发行的 IN 权标不应答 (在未检测到 HUB 的 DATA 包的情况下检测到超时) 或者从 HUB 的信息包中检测到错误时
对于传送类型为中断传送的管道, 如果发生此条件, 此模块就将该管道对应的 PIPENRDY 位置 “1”, 将对应管道的 PID 位改为 NAK, 并且将 CSSTS 位清 “0”。
对于传送类型为等时传送的管道, 如果产生此条件, 此模块就将该管道对应的 PIPENRDY 位置 “1”, 将 CRCE 位置 “1”, 并且将 CSSTS 位清 “0” (不更改管道的 PID 位)。
 - 对于传送类型不是等时传送的管道的 C-SPLIT, 接收到STALL握手信号时
此时, 此模块将该管道对应的 PIPENRDY 位置 “1”, 将对应的管道的 PID 位改为 STALL (11), 并且将 CSSTS 位清 “0”。
 - 对于传送类型为等时传送或者中断传送的管道的 C-SPLIT, 在 microFrame 为 “4” 时接收到NYET握手信号时
此时, 此模块将该管道对应的 PIPENRDY 位置 “1”, 将 CRCE 位置 “1”, 并且将 CSSTS 位清 “0” (不更改管道的 PID 位)。

(c) 选择功能控制器功能的情况

1. 发送方向管道的情况
- 在 FIFO 缓冲器没有发送数据的状态下接收到 IN 权标时
在接收 IN 权标时, 此模块产生 NRDY 中断请求并且将 PIPENRDY 位置 “1”。
在发生中断的管道传送类型为等时传送时, 此模块发送 Zero-Length 包并且将 OVRN 位置 “1”。
2. 接收方向管道的情况
- 在 FIFO 缓冲器没有空间的状态下接收到 OUT 权标时
当发生中断的管道传送类型为等时传送时, 此模块在接收 OUT 权标时产生 NRDY 中断请求, 并且将 PIPENRDY 位置 “1”, 将 OVRN 位置 “1”。
当发生中断的管道传送类型不是等时传送时, 此模块在接收到 OUT 权标之后的数据后发送 NAK 握手信号时产生 NRDY 中断请求, 并且将 PIPENRDY 位置 “1”。
但是, 在重新发送时 (DATA-PID 不同时) 不产生 NRDY 中断请求, 而且在 DATA 包中有错误时也不产生 NRDY 中断。
 - 在 FIFO 缓冲器没有空间的状态下接收到 PING 权标时
在接收到 PING 权标时, 此模块产生 NRDY 中断请求并且将 PIPENRDY 位置 “1”。

- 对于传送类型为等时传送的管道，在间隔帧内无法正常接收时在接收SOF时，此模块产生NRDY中断请求并且将PIPENRDY位置“1”。

选择功能控制器功能时的NRDY中断发生时序图如图29.4所示。

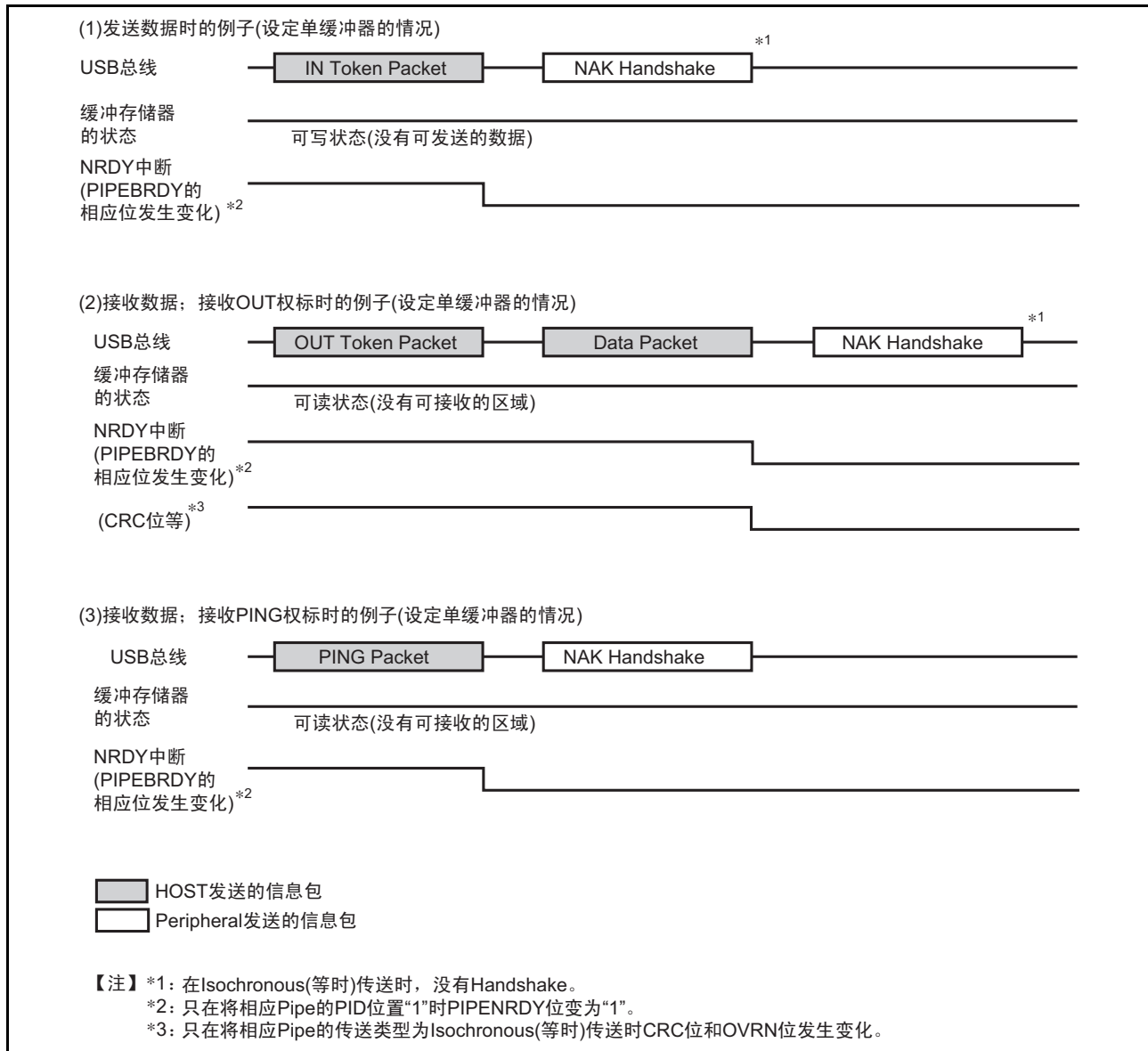


图 29.4 选择功能控制器功能时的NRDY中断发生时序图

(3) BEMP 中断

对于将 PID 位设定为 BUF 的管道，如果此模块检测到 BEMP 中断，就将 BEMPSTS 寄存器的 PIPEBEMP 位的对应位置“1”。此时，如果已将 BEMPENB 寄存器对应的位置“1”，此模块就将 INTSTS0 寄存器的 BEMP 位置“1”，产生 USB 中断。

在以下情况下，此模块产生内部 BEMP 中断请求。

1. 发送方向管道的情况

在发送结束时（包括发送 Zero-Length 包时），对应管道的 FIFO 缓冲器为空时

在设定为单缓冲器时，对于 DCP 以外的管道，在产生 BRDY 中断的同时产生内部 BEMP 中断请求。

但是，在以下情况下不产生内部 BEMP 中断请求：

- 在设定为双缓冲器并且发送完一个缓冲器面的数据时已经开始写 CPU 侧的 FIFO 缓冲器的情况
- 通过给 ACLRM 位或者 BCLR 位写“1”，清除（空）缓冲器。
- 在设定功能控制器功能的情况下，进行控制传送 Status 阶段的 IN 传送（发送 Zero-Length 包）时。

2. 接收方向管道的情况

正常接收到大于 MaxPacketSize 设定值的数据。

此时，此模块产生 BEMP 中断请求，将 PIPEBEMP 位的对应位置“1”，并且放弃接收数据以及将对应管道的 PID 位改为 STALL（11）。

此模块在设定为主机控制器功能时不进行应答，而在设定为功能控制器功能时进行 STALL 应答。

但是，在以下情况下，不产生内部 BEMP 中断请求：

- 从接收数据中检测到 CRC 错误或者位填充错误等时
- 执行设置事务时

能通过给 PIPEBEMP 位写“0”来清除状态。

即使给 PIPEBEMP 位写“1”，也不影响运行。

选择功能控制器功能时的 BEMP 中断发生时序图如图 29.5 所示。

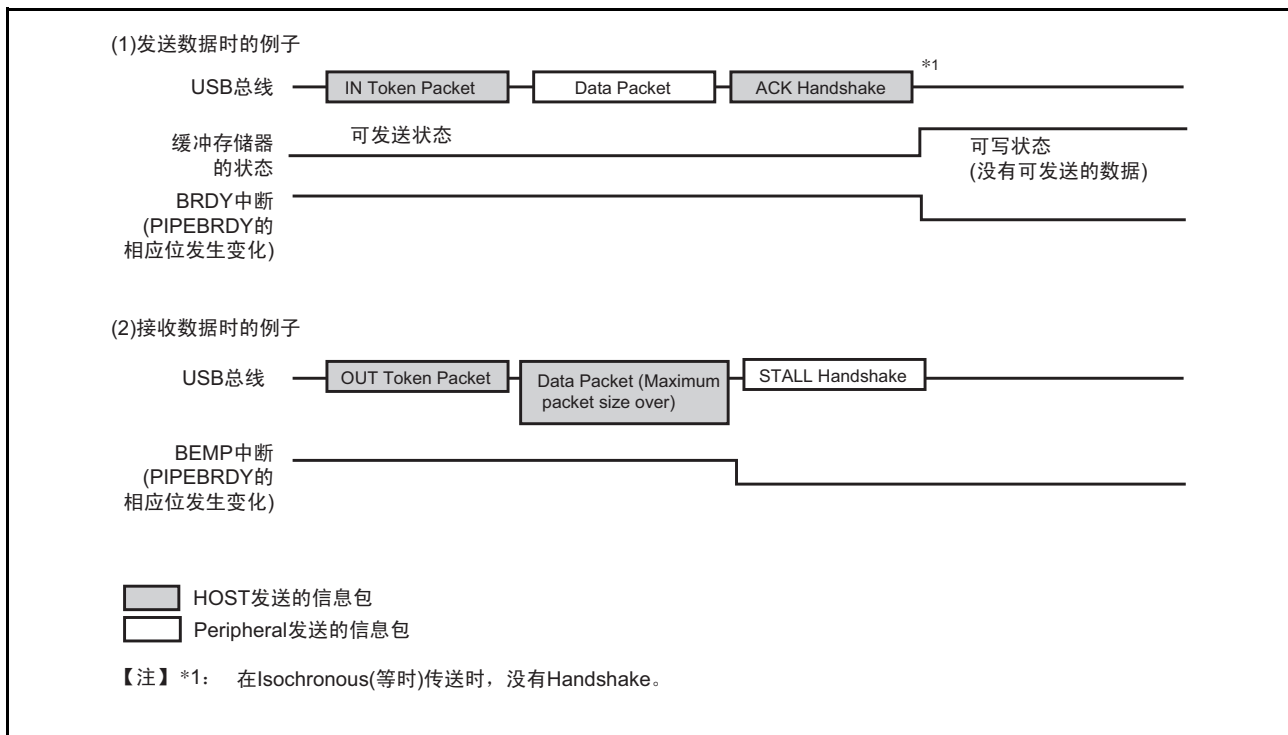


图 29.5 选择功能控制器功能时的 BEMP 中断发生时序图

(4) 设备状态转移中断

此模块的设备状态转移图如图 29.6 所示。此模块管理设备状态并且产生设备状态转移中断，但是通过恢复中断检测从挂起的恢复（恢复信号检测）。能通过 INTENB0 寄存器独立设定允许或者禁止设备状态转移中断，还能通过 INTSTS0 寄存器的 DVSQ 位确认转移后的设备状态。

当转移到默认状态时，在复位握手信号协议结束后产生设备状态转移中断。

只在选择功能控制器功能时，才能管理设备状态并且产生设备状态转移中断。

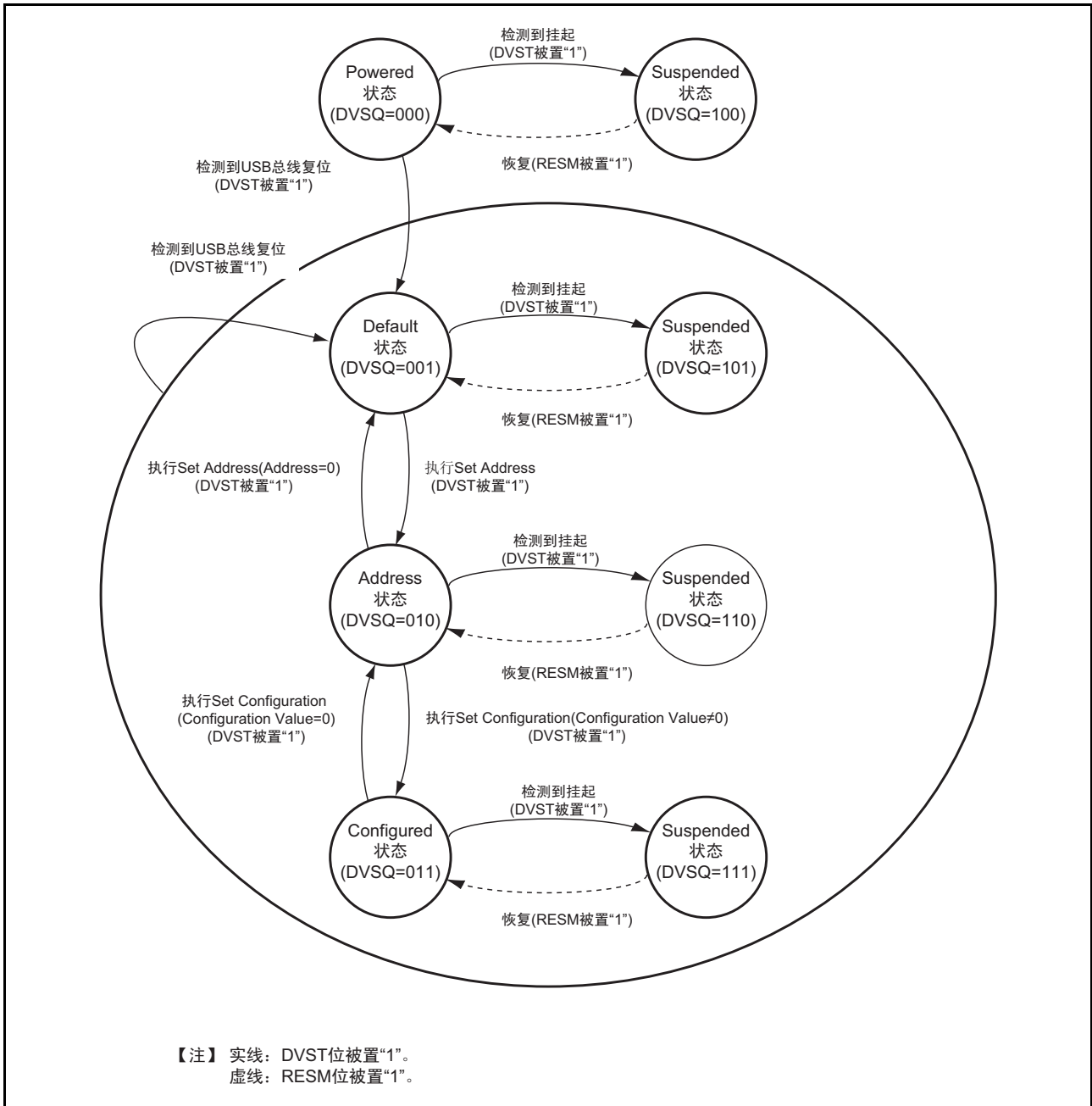


图 29.6 设备状态转移图

(5) 控制传送阶段转移中断（功能控制器功能）

此模块的控制传送阶段转移图如图 29.7 所示。此模块管理控制传送顺序并且产生控制传送阶段转移中断。能通过 INTENB0 寄存器独立设定允许或者禁止控制传送阶段转移中断，还能通过 INTSTS0 寄存器的 CTSQ 位确认转移后的传送阶段。

只在选择功能控制器功能时，才能产生控制传送阶段转移中断。

控制传送的顺序错误如下所示。在发生错误时，DCPCTR 寄存器的 PID 位变为“B'1x”（STALL 应答）。

1. 控制读传送的情况

- 对于数据阶段的 IN 权标，在一次也没有进行数据传送的状态下接收到 OUT 权标或者 PING 权标。
- 在状态阶段接收到 IN 权标。
- 在状态阶段接收到 DATAPID 为 DATA0 的数据包。

2. 控制写传送的情况

- 对于数据阶段的 OUT 权标，在一次也没有进行 ACK 应答的状态下接收到 IN 权标。
- 在数据阶段接收到 DATAPID 为 DATA0 的最初的数据包。
- 在状态阶段接收到 OUT 权标或者 PING 权标。

3. 控制写无数据控制传送的情况

- 在状态阶段接收到 OUT 权标或者 PING 权标。

在控制写传送数据阶段，如果接收数据量超过 USB 请求的 wLength 值，就不能识别是控制传送顺序错误。在控制读传送状态阶段，如果接收到的信息包不是 Zero-Length 包，就进行 ACK 应答并且正常结束。

当因顺序错误而发生 CTRT 中断时，就在系统给 CTRT 位写“0”（清除中断状态）前 CTSQ 位保持“110”的值。因此，在 CTSQ 保持“110”的状态下，即使接收到新的 USB 请求，也不产生设置阶段结束的 CTRT 中断（此模块保持设置阶段结束，在清除中断状态后产生 CTRT 中断）。

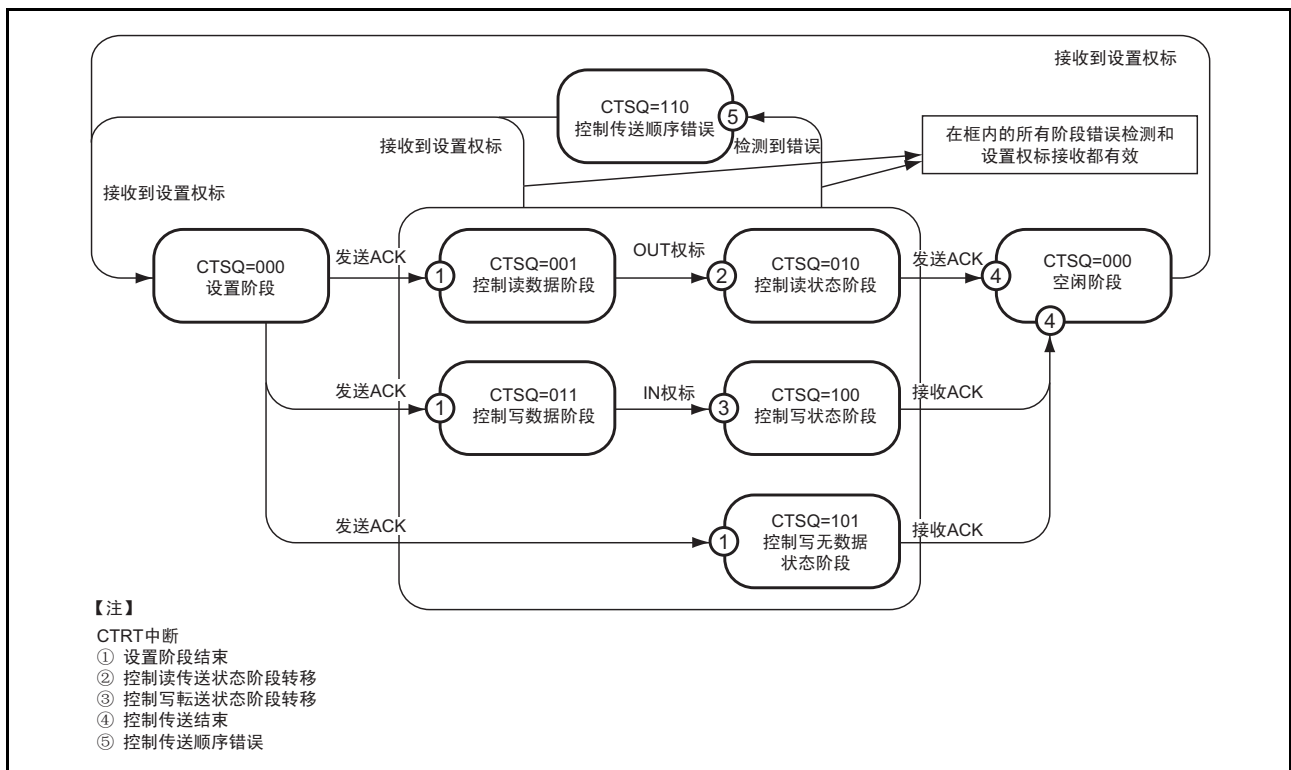


图 29.7 控制传送阶段转移图

(6) 帧号更新中断

此模块的 SOFR 中断输出时序例子如图 29.8 所示。

如果选择主机控制器功能，就在更新帧号时产生中断；如果选择功能控制器功能，就在更新帧号时产生 SOFR 中断。

当选择功能控制器功能时，如果此模块在全速运行中检测到新的 SOF 包，就更新帧号并且产生 SOFR 中断。但是，如果在高速运行中不是 μ SOF 锁定状态，就不更新帧号，也不产生 SOFR 中断，并且 SOF 的内插功能也不运行。 μ SOF 锁定状态是指连续 2 次接收到无错误并且帧号不同的 μ SOF 包。

另外， μ SOF 锁定监视开始条件和 μ SOF 锁定监视停止条件如以下 2 点所示：

1. μ SOF 锁定监视开始条件
USBE 位为 “1”。
2. μ SOF 锁定监视停止条件
USBE 位为 “0”，接收到 USB 总线复位或者检测到挂起。

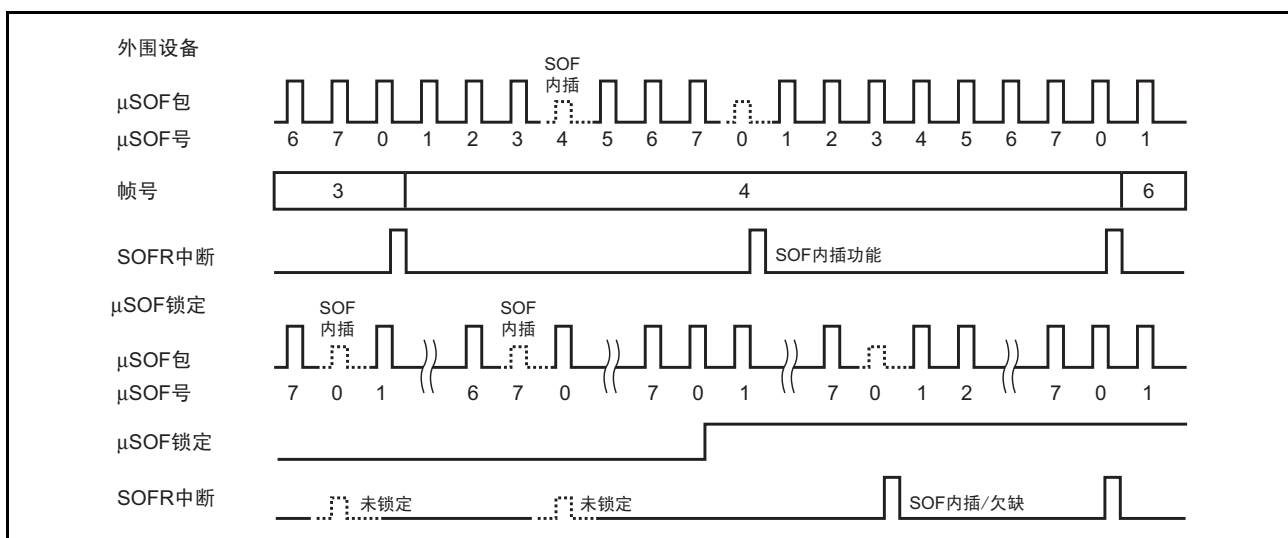


图 29.8 SOFR 中断输出时序例子

(7) VBUS 中断

在 VBUS 引脚发生变化时，产生 VBUS 中断。能通过 INTSTS0 寄存器的 VBSTS 位确认 VBUS 引脚的电平，能通过 VBUS 中断确认主机控制器的连接和切断。但是，在连接主机控制器的状态下启动系统时，VBUS 引脚不变，因此不产生最初的 VBUS 中断。

(8) 恢复中断

当选择功能控制器功能时，如果在设备状态为挂起的状态下 USB 总线状态发生变化 (J-State→K-State 或者 J-State→SE0)，就产生恢复中断。通过恢复中断检测从挂起状态的返回。

当选择主机控制器功能时，不产生恢复中断。必须使用 BCHG 中断来检测 USB 总线的变化。

(9) BCHG 中断

在 USB 总线状态发生变化时，产生 BCHG 中断。此中断用于在选择主机控制器功能时检测外围设备的连接和远程唤醒。在选择主机控制器功能或者功能控制器功能时，都产生 BCHG 中断。

(10) DTCH 中断

当选择主机控制器功能时，如果检测到 USB 总线断开，就产生 DTCH 中断。此模块以 USB Specification2.0 为基准检测总线的断开。

此模块在检测到 DTCH 中断后（与该中断允许位的设定值无关）进行以下的硬件控制。必须结束正在与 USB 端口进行通信的全部管道的通信，并且转移到 USB 端口的连接（产生 ATTCH 中断）等待状态。

- 将 UACT 位改为“0”。
- 转移到空闲状态。

(11) SACK 中断

当选择主机控制器功能时，如果接收到外围设备对发送设置包的 ACK 应答，就产生 SACK 中断。能通过 SACK 中断得知设置事务正常结束。

(12) SIGN 中断

当选择主机控制器功能时，如果不能连续 3 次正常接收到外围设备对发送设置包的 ACK 应答，就产生 SIGN 中断。能检测到外围设备未发送 ACK（不应答）或者 ACK 包破损。

(13) ATTCH 中断

当选择主机控制器功能时，如果在 USB 端口检测到 2.5 μ s 的全速 / 低速信号电平的 J-State 或者 K-State，就产生 ATTCH 中断。具体的 ATTCH 中断检测条件如下：

- 从 K-State、SE0 或者 SE1 变为 J-State，并且 J-State 持续保持 2.5 μ s。
- 从 J-State、SE0 或者 SE1 变为 K-State，并且 K-State 持续保持 2.5 μ s。

(14) EOFERR 中断

如果在 USB Specification2.0 规定的 EOF2 时序检测到通信未结束，就产生 EOFERR 中断。

此模块在检测到 EOFERR 中断后（与该中断允许位的设定值无关）进行以下的硬件控制。必须结束正在与该端口进行通信的全部管道的通信，并且对该端口进行再 Enumeration。

- 将检测到 EOFERR 中断的端口的 UACT 位改为“0”。
- 将发生 EOFERR 中断的端口转移到空闲状态。

29.4.3 管道控制

此模块的管道设定项目一览表如表29.17所示。USB数据传送需要通过称为端点的逻辑管道进行数据通信。此模块有10个用于数据传送的管道，必须按照系统规格设定各管道。

表 29.17 管道设定项目一览表

寄存器名	位名	设定内容	备注
DCPCFG PIPECFG	TYPE	指定传送类型。	管道 1 ~ 9: 能设定
	BFRE	选择 BRDY 中断模式。	管道 1 ~ 5: 能设定
	DBLB	选择双缓冲器。	管道 1 ~ 5: 能设定
	CNTMD	选择连续传送或者非连续传送。	管道 1 和管道 2: 能设定 (只在选择批量传送时才能设定) 管道 3 ~ 5: 能设定
	DIR	选择传送方向。	能设定 IN 或者 OUT。
	EPNUM	端点号	管道 1 ~ 9: 能设定 在使用管道时, 不能设定“0000”。
	SHTNAK	选择在传送结束时禁止管道。	管道 1 和管道 2: 能设定 (只在选择批量传送时才能设定) 管道 3 ~ 5: 能设定
PIPEBUF	BUFSIZE	缓冲存储器大小	DCP: 不能设定 (固定为 256 字节) 管道 1 ~ 5: 能设定 (以 64 字节为单位, 最多能指定 2K 字节) 管道 6 和管道 9: 不能设定 (固定为 64 字节)
	BUFNMB	缓冲存储器号	DCP: 不能设定 (固定区域 H'0 ~ H'3) 管道 1 ~ 5: 能设定 (能指定区域 H'8 ~ H'7F) 管道 6 ~ 9: 不能设定 (固定区域 H'4 ~ H'7)
DCPMAXP	DEVSEL	选择设备。	只在选择主机控制器功能时能参照。
PIPEMAXP	MXPS	最大信息包长度	符合 USB 规格的设定
PIPEPERI	IFIS	缓冲器转储清除。	管道 1 和管道 2: 能设定 (只在选择等时传送选择时才能设定) 管道 3 ~ 5: 不能设定 管道 6 ~ 9: 只在选择主机控制器功能时才能设定。
	IITV	间隔计数器	管道 1、2: 能设定 (只在选择等时传送时才能设定) 管道 3 ~ 5: 不能设定 管道 6 ~ 9: 只在选择主机控制器功能时才能设定。

寄存器名	位名	设定内容	备注
DCPCTR	BSTS	缓冲器状态	DCP 通过 ISEL 位进行接收 / 发送缓冲器状态的切换。
PIPEnCTR	INBUFM	监视 IN 缓冲器。	只内置于管道 3 ~ 5。
	SUREQ	设置请求	只有 DCP 能设定 只在选择主机控制器功能时才能控制。
	SUREQCLR	SUREQ 的清除	只有 DCP 能设定 只在选择主机控制器功能时才能控制。
	CSCLR	CSSTS 的清除	只在选择主机控制器功能时才能控制。
	CSSTS	SPLIT 状态表示	只在选择主机控制器功能时才能参照。
	ATREPM	自动应答模式	管道 1 ~ 5: 能设定 只在选择功能控制器功能时才能设定。
	ACLRM	缓冲器的自动清除	管道 1 ~ 9: 能设定
	SQCLR	顺序的清除	清除数据交替位。
	SQSET	顺序的置位	将数据交替位置位。
	SQMON	顺序的确认	确认数据交替位。
	PBUSY	管道忙的确认为	
	PID	应答 PID	请参照“29.4.3(6) 应答 PID”。
	PIPEnTRE	TRENB	允许事务计数。
TRCLR		清除当前事务计数器。	管道 1 ~ 5: 能设定
PIPEnTRN	TRNCNT	事务计数器	管道 1 ~ 5: 能设定

(1) 管道控制寄存器的切换步骤

只能在不允许 USB 通信 (PID 位为 NAK) 时改写以下管道控制寄存器的位。

在允许 USB 通信 (PID 位为 BUF) 的状态下禁止设定的寄存器:

- DCPMAXP 寄存器的各位
- DCPCTR 寄存器的 SQCLR 位、SQSET 位和 PINGE 位
- PIPECFG 寄存器、PIPEBUF 寄存器、PIPEMAXP 寄存器和 PIPEPERI 寄存器的各位
- PIPEnCTR 寄存器的 ATREPM 位、ACLRM 位、SQCLR 位和 SQSET 位
- PIPEnTRE 寄存器和 PIPEnTRN 寄存器的各位
- DEVADDn 寄存器的各位

【注】 对于 CSCLR 位和 DEVADDn 寄存器的设定, 除了遵守上述规则外, 还要遵守寄存器说明中的设定方法。

在允许 USB 通信 (PID 位为 BUF) 的状态下改写上述位时, 必须按照以下步骤进行:

1. 产生管道控制寄存器的位变更请求。
2. 将相应管道的 PID 位改为 NAK。
3. 等待该管道的 CSSTS 位变为“0”(只限于选择主机控制器功能时)。
4. 等待该管道的 PBUSY 位变为“0”。
5. 开始更改管道控制寄存器的位。

只在还没有给 CFIFOSEL、D0FIFOSEL 或者 D1FIFOSEL 寄存器的 CURPIPE 位设定管道信息时, 才能改写以下管道控制寄存器的位。

在设定 FIFO-PORT 的 CURPIPE 过程中禁止设定的寄存器:

- DCPCFG 寄存器和 DCPMAXP 寄存器的各位
- PIPECFG 寄存器、PIPEBUF 寄存器、PIPEMAXP 寄存器和 PIPEPERI 寄存器的各位
- PIPEnCTR 寄存器的 ACLRM 位

在更改管道信息时，必须将 CURPIPE 位指定为变更管道以外的管道。有关 DCP，必须在修改管道信息后，通过 BCLR 进行缓冲器的清除处理。

(2) 传送类型

通过 PIPEPCFG 寄存器的 TYPE 位设定各管道的传送类型。能给各管道设定的传送类型如下：

- DCP：不需要设定（固定为控制传送）。
- 管道1和管道2：必须设定为批量传送或者等时传送。
- 管道3～5：必须设定批量传送。
- 管道6～9：必须设定中断传送。

(3) 端点号

通过 PIPEPCFG 寄存器的 EPNUM 位设定各管道的端点号。DCP 固定为端点 0。其他管道能设定端点 1～端点 15。

- DCP：不需要设定（固定为端点 0）。
- 管道 1～9：必须设定 1～15。

但是，DIR 位和 EPNUM 位的组合不能重复。

(4) 最大信息包长度的设定

通过 DCPMAXP 寄存器和 PIPEMAXP 寄存器的 MXPS 位设定各管道的最大信息包长度。能给 DCP 和管道 1～5 设定 USB 规格定义的所有最大信息包长度。管道 6～9 最大为 64 字节，是最大信息包长度的上限。必须在开始传送前（PID 位为 BUF）设定最大信息包长度。

- DCP：在高速运行时，必须设定 64。
- DCP：在全速运行时，必须设定 8、16、32 或者 64。
- 管道 1～5：在高速批量传送时，必须设定 512。
- 管道 1～5：在全速批量传送时，必须设定 8、16、32 或者 64。
- 管道 1 和管道 2：在高速等时传送时，必须设定 1～1024 的值。
- 管道 1 和管道 2：在全速等时传送时，必须设定 1～1023 的值。
- 管道 6～9：必须设定 1～64 的值。

不支持中断传送和等时传送的 High Bandwidth。

(5) 事务计数器（管道 1～5 为读方向）

如果以数据包接收方向结束指定次数的事务处理，此模块就能认为是传送结束。在 D0FIFO/D1FIFO 端口选择的管道为从缓冲存储器读数据的方向时，事务计数器工作。事务计数器有指定事务次数的 TRNCNT 寄存器和在内部进行事务计数的当前计数器，如果当前计数器和指定的次数相同，缓冲存储器就为可读的状态。能通过 TRCLR 位对事务计数器功能的当前计数器进行初始化，对事务从头开始计数。TRNCNT 寄存器的读取信息因 TRENb 位的设定而不同。

- TRENb=0：能读取所设定的事务计数器的值。
- TRENb=1：能读取在内部进行计数的当前计数器的值。

TRCLR 位的操作条件如下：

- 在事务计数过程中并且 PID 位为 BUF 时，不能清除当前计数器。
- 在缓冲器中有剩余数据的状态下，不能清除当前计数器。

(6) 应答 PID

通过 DCPCTR 寄存器和 PIPEnCTR 寄存器的 PID 位设定各管道的应答 PID。

在各种设定状态下，此模块的运行如下：

1. 选择主机控制器功能时的应答PID设定
应答PID指定事务的执行。
 - 设定NAK：为管道禁止状态，不执行事务。
 - 设定BUF：根据缓冲存储器的状态执行事务。
在OUT方向时，如果缓冲存储器中有发送数据，就发行OUT权标。
在IN方向时，如果缓冲存储器中有空间并且为接收允许状态，就发行IN权标。
 - 设定STALL：为管道禁止状态，不执行事务。

【注】 通过 SUREQ 位设定 DCP 的设置事务。

2. 选择功能控制器功能时的应答PID设定
应答PID指定对主机事务的应答。
 - 设定NAK：总是对发生的事务进行NAK应答。
 - 设定BUF：根据缓冲存储器的状态对事务进行应答。
 - 设定STALL：总是对发生的事务进行STALL应答。

【注】 与PID的设定无关，总是对设置事务进行ACK应答，并且将USB请求保存到寄存器。

根据事务结果，此模块有可能写PID位。在发生以下情况时，此模块写PID位。

1. 在选择主机控制器功能时由硬件设定应答PID的情况
 - 设定NAK：在以下情况下，PID位为NAK并且自动停止发行权标。
 - 在等时传送以外的传送过程中，在对发送权标的应答中连续3次不应答或者发生位填充错误和CRC错误等接收错误时
 - 在等时传送过程中，在对发送权标的应答中连续3次发生位填充错误和CRC错误等接收错误时
 - 在将DCPCFG寄存器的SHTNAK位置“1”的情况下在控制读传送的数据阶段接收到短包时
 - 在批量传送过程中将PIPECFG寄存器的SHTNAK位置“1”的情况下接收到短包时
 - 在批量传送过程中将SHTNAK位置“1”的情况下结束事务计数器时
 - 设定BUF：此模块不写BUF。
 - 设定STALL：在以下情况下，PID位为STALL并且自动停止发行权标。
对发送的权标，接收到STALL时
当接收的数据包超过最大信息包长度时
2. 在选择功能控制器功能时由硬件设定应答PID的情况
 - 设定NAK：在以下情况下，PID位为NAK并且总是对事务进行NAK应答。
当正常接收到设置权标时（只限于DCP）
在批量传送过程中将PIPECFG寄存器的SHTNAK位置“1”的情况下结束事务计数器或者接收到短包时
 - 设定BUF：此模块不写BUF。
 - 设定STALL：在以下情况下，PID位为STALL并且总是对事务进行STALL应答。
在接收数据包中检测到超过最大信息包长度的错误时
当检测到控制传送顺序错误时（只限于DCP）

(7) 数据 PID 顺序位

在控制传送的数据阶段、批量传送和中断传送中，如果进行正常的传送，就通过此模块自动将数据 PID 的顺序位取反。能通过 DCPCTR 寄存器和 PIPEnCTR 寄存器的 SQMON 位确认下一次发送的数据 PID 顺序位。在发送数据的情况下，在接收 ACK 握手信号时转换顺序位；在接收数据的情况下，在发送 ACK 握手信号时转换顺序位。也能通过 DCPCTR 寄存器和 PIPEnCTR 寄存器的 SQCLR 位、SQSET 位更改数据的 PID 顺序位。

对于选择功能控制器功能时的控制传送，此模块在阶段转移时自动设定顺序位。在设置阶段结束时由 DATA1 应答；在状态阶段不参照顺序位而由 DATA1 应答。因此，不需要进行设定。对于选择主机控制器功能时的控制传送，需要在阶段转移时设定顺序位。

必须注意：无论是选择主机控制器功能还是选择功能控制器功能，在发送或者接收 ClearFeature 请求时，都要设定数据 PID 顺序位。

设定为等时传送的管道不能通过 SQSET 位操作顺序位。

(8) 应答 PID 为 NAK 时的功能

此模块能通过将 PIPECFG 寄存器的 SHTNAK 位置“1”，在接收传送的最后（根据接收的短包或者事务计数器自动识别模块）数据包时禁止管道运行（应答 PID 为 NAK）。

通过使用此功能，在缓冲存储器用作双缓冲器时，能以传送单位接收数据包。在禁止管道运行后，需要重新设定为允许管道运行（应答 PID 为 BUF）。

此功能只能用于批量传送。

(9) 自动应答模式

对于批量传送的管道（管道 1 ~ 5），如果将 PIPEnCTR 寄存器的 ATREPM 位置“1”，就变为自动应答模式。在 OUT 传送时（DIR 位为“0”）变为 OUT-NAK 模式；在 IN 传送时（DIR 位为“1”）变为 Null 自动应答模式。

• OUT-NAK 模式

对于批量 OUT 传送的管道，如果将 ATREPM 位置“1”，就对 OUT 权标或者 PING 权标进行 NAK 应答并且输出 NRDY 中断。为了从正常模式设定为 OUT-NAK 模式，必须在管道运行禁止状态（应答 PID 为 NAK）下设定为 OUT-NAK 模式，然后允许管道运行（应答 PID 为 BUF）。在允许管道运行后，OUT-NAK 模式有效。如果在设定为禁止管道运行前接收到 OUT 权标，就正常接收此权标的的数据，并且对主机进行 ACK 应答。

为了从 OUT-NAK 模式转移到正常模式，必须在管道运行禁止状态（应答 PID 为 NAK）下解除 OUT-NAK 模式，然后允许管道运行（应答 PID 为 BUF）。在正常模式中，能接收 OUT 数据。对于 PING 权标，如果缓冲器为能接收的状态，就返回 ACK。

• Null 自动应答模式

对于批量 IN 传送的管道，如果将 ATREPM 位置“1”，就继续发送 Zero-Length 包。

为了从正常模式设定为 Null 自动应答模式，必须在管道运行禁止状态（应答 PID 为 NAK）下设定为 Null 自动应答模式，然后允许管道运行（应答 PID 为 BUF）。在允许管道运行后，Null 自动应答模式有效。但是，在设定为 Null 自动应答模式时，需要缓冲器为空的状态，因此必须确认 INBUFM 位是否为“0”。当 INBUFM 位为“1”时，因为缓冲器有数据，所以必须通过 ACLRM 位将缓冲器清空。在 Null 自动应答模式的设定过程中，不能给 FIFO 端口写数据。

为了从 Null 自动应答模式转移到正常模式，必须使管道运行禁止状态（应答 PID 为 NAK）保持 1 个 Zero-Length 包的发送等待时间（全速运行时：10 μ s，高速运行时：3 μ s），然后解除 Null 自动应答模式。在正常模式中，能写 FIFO 端口；在允许管道运行（应答 PID 为 BUF）后，能给主机发送信息包。

29.4.4 FIFO 缓冲存储器

(1) FIFO 缓冲存储器的分配

此模块的 FIFO 缓冲存储器映像例子如图 29.9 所示。FIFO 缓冲存储器是 CPU 和此模块共用的区域。FIFO 缓冲存储器的状态有 2 种；系统（CPU 侧）拥有存取权和此模块（SIE 侧）拥有存取权。

FIFO 缓冲存储器给每个各管道设定独立的区域。以 64 字节为 1 块，通过块起始号和块数（PIPEBUF 寄存器的 BUFNMB 位和 BUFSIZE 位）设定存储区。

如果通过 PIPE_nCFG 寄存器的 CNTMD 位选择连续传送模式，就必须将 BUFSIZE 位设定为最大信息包长度的整数倍。如果通过 PIPE_nCFG 寄存器的 DBLB 位选择双缓冲器，就给同一管道分配 2 个由 PIPEBUF 寄存器的 BUFSIZE 位指定的存储区。

缓冲存储器的存取（读写数据）使用 3 个 FIFO 端口。通过 C/D_nFIFOSEL 寄存器的 CURPIPE 位，给分配到 FIFO 端口的管道指定管道号。

能通过 DCPCTR 寄存器和 PIPE_nCTR 寄存器的 BSTS 位、INBUFM 位确认各管道的缓冲器状态，并且能通过 CFIFOCTR 寄存器和 D_nFIFOCTR 寄存器的 FRDY 位确认 FIFO 端口的存取权。

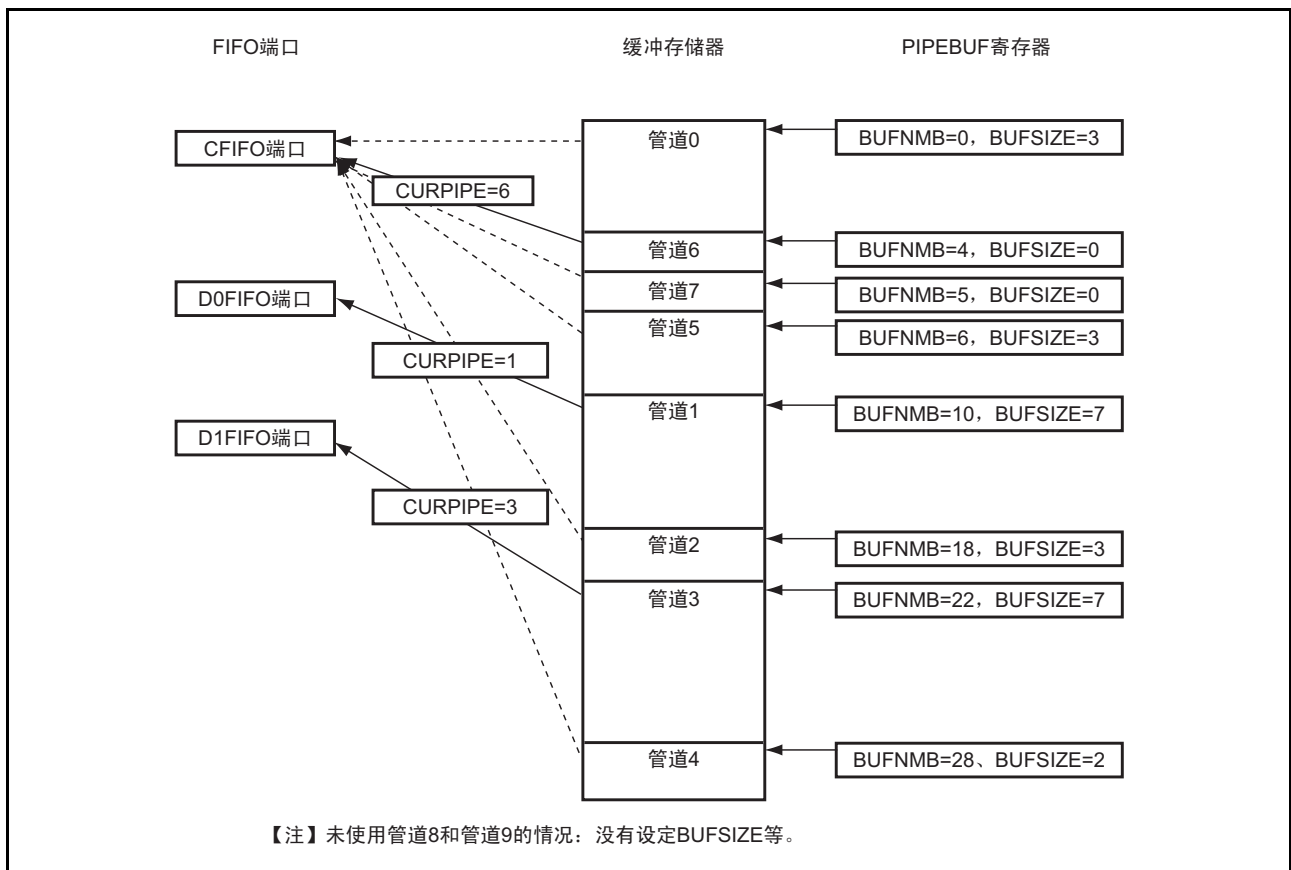


图 29.9 缓冲存储器映像例子

- 缓冲器状态

此模块的缓冲器状态表如表 29.18 和表 29.19 所示。能通过 DCPCTR 寄存器的 BSTS 位和 PIPE_nCTR 寄存器的 INBUFM 位确认缓冲存储器状态，并且能通过 PIPE_nCFG 寄存器的 DIR 位或者 CFIFOSEL 寄存器的 ISEL 位（选择 DCP 时）指定缓冲存储器的存取方向。

INBUFM 位只对发送方向的管道 1~5 有效。

当将发送侧的传送管道设定为双缓冲器时，BSTS 位用于判断 CPU 侧的缓冲器状态，INBUFM 位用于判断 SIE 侧的缓冲器状态。在因 CPU（直接存储器存取控制器）给 FIFO 端口写数据的速度慢而不能通过 BEMP 中断判断缓冲器为空时，能通过 INBUFM 位确认发送结束。

表 29.18 通过 BSTS 位表示的缓冲器状态表

ISEL 或者 DIR	BSTS	缓冲存储器的状态
0（接收方向）	0	没有接收数据或者正在接收。 不能从 FIFO 端口读数据。
0（接收方向）	1	有接收数据或者接收到 Zero-Length 包。 能从 FIFO 端口读数据。 但是，因为在接收 Zero-Length 包时不能读数据，所以需要清除缓冲器。
1（发送方向）	0	发送尚未结束。 不能给 FIFO 端口写数据。
1（发送方向）	1	发送结束。 CPU 能写数据。

表 29.19 通过 INBUFM 位表示的缓冲器状态表

DIR	INBUFM	缓冲存储器的状态
0（接收方向）	无效	无效
1（发送方向）	0	发送完能发送的数据。 没有能发送的数据。
1（发送方向）	1	已给 FIFO 端口写能发送的数据。 有能发送的数据。

- FIFO 缓冲器的清除

通过此模块清除的 FIFO 缓冲存储器一览表如表 29.20 所示，能通过 BCLR 位、DCLRM 位和 ACLRM 位清除缓冲存储器。

表 29.20 缓冲器清除一览表

位名	BCLR	DCLRM	ACLRM
寄存器	CFIFOCTR 寄存器 DnFIFOCTR 寄存器	DnFIFOSEL 寄存器	PIPEnCTR 寄存器
功能	清除 CPU 侧的缓冲存储器。	这是在读指定管道的数据后自动清除缓冲存储器的模式。	这是放弃接收全部信息包的缓冲器自动清除模式。
清除方法	通过写“1”来清除。	1: 模式有效 0: 模式无效	1: 模式有效 0: 模式无效

- 缓冲区

此模块的缓冲存储器映像如表 29.21 所示。缓冲存储器有事先分配给管道的专用固定区和用户能设定的用户区。

DCP 的缓冲器是在控制读传送和控制写传送时使用同一个区域的专用固定区。

事先分配管道 6~9 的区域，在不使用管道 6~9 时，能作为用户区分配给管道 1~5 使用。

各管道的区域不能重叠。尤其要注意在设定双缓冲器时区域为设定值的倍数。

不能用小于最大信息包长度的设定值指定缓冲器大小。

表 29.21 缓冲存储器映像

缓冲存储器号	缓冲器大小	管道设定	备注
H'0 ~ H'3	256 字节	DCP 专用固定区	单缓冲器
H'4	64 字节	管道 6 的固定区	单缓冲器
H'5	64 字节	管道 7 的固定区	单缓冲器
H'6	64 字节	管道 8 的固定区	单缓冲器
H'7	64 字节	管道 9 的固定区	单缓冲器
H'8 ~ H'7F	最大 7616 字节	管道 1 ~ 5 的用户区	能设定双缓冲器，进行连续传送。

- 缓冲器自动清除模式功能

通过将 PIPEnCTR 寄存器的 ACLRM 位置“1”，此模块放弃所接收的全部数据包。但是，当接收到正常的数据包时，就对主机控制器进行 ACK 应答。只有在缓冲存储器为读方向时，才能设定此功能。与存取方向无关，能通过在给 ACLRM 位写“1”后接着写“0”，清除所选管道的缓冲存储器。

- 缓冲存储器的规格（设定单缓冲器/双缓冲器）

管道 1 ~ 5 能通过 PIPEnCFG 寄存器的 DBLB 位选择单缓冲器或者双缓冲器。双缓冲器具有给同一管道分配 2 个由 PIPEBUF 寄存器的 BUFSIZE 位指定的存储区面的功能。此模块的缓冲存储器设定例子如图 29.10 所示。

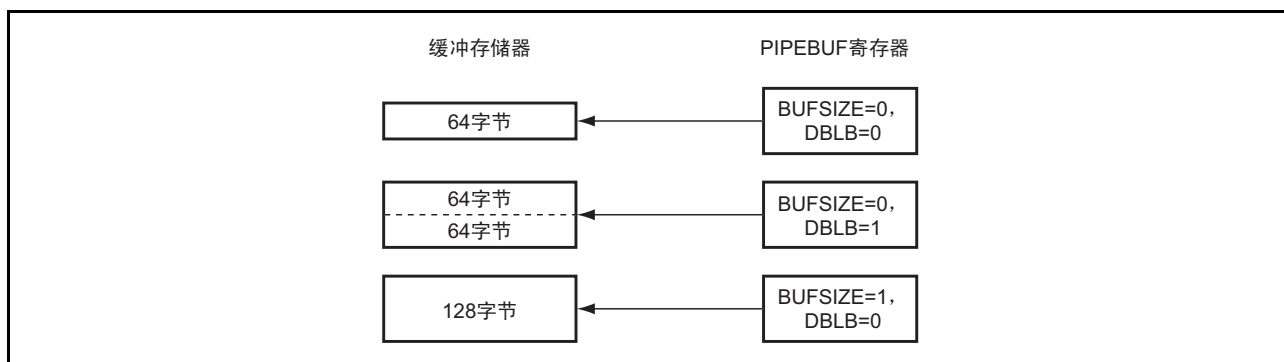


图 29.10 缓冲存储器设定例子

- 缓冲存储器的运行（设定连续传送）

能通过 PIPEnCFG 寄存器的 CNTMD 位选择连续传送模式或者非连续传送模式。此选择只对管道 1 ~ 5 有效。

连续传送模式功能是连续发送和接收多个事务的功能。在设定连续传送模式时，能在不向 CPU 请求中断的情况下进行分配给各管道的缓冲器大小的数据传送。

在连续发送模式中，根据最大信息包长度，将写数据分割后发送。对于小于缓冲器大小的数据发送（短包或者最大信息包长度的整数倍小于缓冲器大小），需要在写发送数据后将 BVAL 位置“1”。

在连续接收模式中，在接收不超过缓冲器大小的信息包或者结束事务计数或者接收短包前不产生中断。

CNTMD 位的设定值和 FIFO 缓冲器的发送/接收结束判定方法的关系如表 29.22 所示。

表 29.22 CNTMD 位的设定值和 FIFO 缓冲器的发送 / 接收结束判定方法的关系

连续传送模式	可读状态和可发送状态的判定方法
非连续传送 (CNTMD 位为 “0”)	在设定为接收方向时 (DIR 位为 “0”), FIFO 缓冲器为可读状态的条件: • 当此模块接收到 1 个信息包时
	在设定为发送方向时 (DIR 位为 “1”), FIFO 缓冲器为可发送状态的条件: 满足以下任意一个条件: • 将最大信息包长度的数据写到 FIFO 缓冲器。 • 将短包数据 (包括 0 字节) 写到 FIFO 缓冲器, 并且给 BVAL 位写 “1”。
连续传送 (CNTMD 位为 “1”)	在设定为接收方向时 (DIR 位为 “0”), FIFO 缓冲器为可读状态的条件: • 当分配给所选管道的 FIFO 缓冲器接收的数据字节数和所分配的字节数 ((BUFSIZE+1)×64) 相等时 • 当此模块接收到 Zero-Length 包以外的短包时 • 在分配给所选管道的 FIFO 缓冲器已保存数据的状态下, 此模块接收到 Zero-Length 包时。 • 当接收到给所选管道设定的事务计数器次数的信息包时
	在设定为发送方向时 (DIR 位为 “1”), FIFO 缓冲器为可发送状态的条件: 满足以下任意一个条件: • 当写的数据量和分配给所选管道的一个 FIFO 缓冲器面大小相等时 • 当将小于分配给所选管道的一个 FIFO 缓冲器面大小的数据量 (包括 0 字节) 写到 FIFO 缓冲器并且将 BVAL 位置 “1” 时 • 当将小于分配给所选管道的一个 FIFO 缓冲器面大小的数据量写到 FIFO 缓冲器并且在最后写时接受到 DMA 传送结束信号时。

此模块的缓冲存储器运行例子如图 29.11 所示。

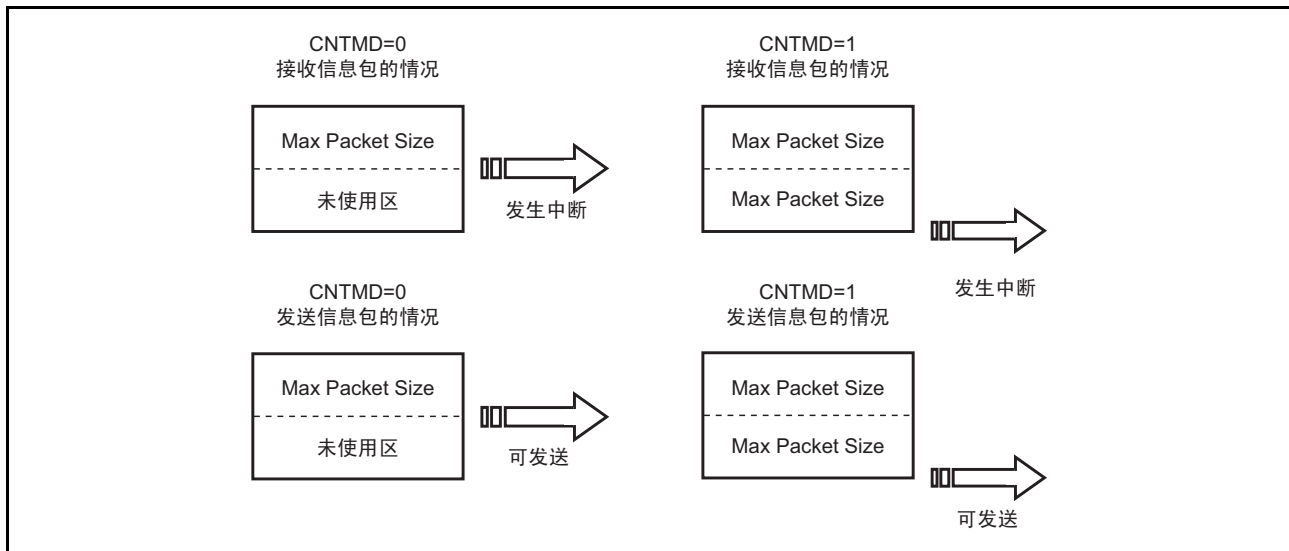


图 29.11 缓冲存储器运行例子

(2) FIFO 端口功能

此模块的 FIFO 端口功能设定表如表 29.23 所示。在进行写数据的存取时，如果写满缓冲器（非连续传送时为最大信息包长度），就自动进入可发送状态。为了将缓冲器存满（非连续传送时为最大信息包长度）前的数据变为可发送状态，需要通过 C/DnFIFOCTR 寄存器的 BVAL 位设定写结束。在发送 Zero-Length 包时，不仅需要相同寄存器的 BCLR 位清除缓冲器，还需要通过 BVAL 位设定写结束。

在进行读存取时，如果读取全部数据，就自动进入可接收新包的状态。但是，在接收 Zero-Length 包时（DTLN 位为“0”），因为无法读取数据，所以需要通过相同寄存器的 BCLR 位清除缓冲器。通过 C/DnFIFOCTR 寄存器的 DTLN 位确认接收数据的长度。

表 29.23 FIFO 端口功能设定表

寄存器名	位名	功能	备注
C/DnFIFOSEL	RCNT	选择 DTLN 读模式。	
	REW	对缓冲存储器进行复位（重新读或者重新写）	
	DCLRM	在读指定管道的接收数据后自动清除。	DnFIFO 专用
	DREQE	允许 DMA 传送。	DnFIFO 专用
	MBW	FIFO 端口的存取位宽	
	BIGEND	选择 FIFO 端口的字节序。	
	ISEL	FIFO 端口的存取方向	DCP 专用
C/DnFIFOCTR	CURPIPE	选择当前管道。	
	BVAL	结束缓冲存储器的写操作。	
	BCLR	清除 CPU 侧的缓冲存储器。	
	DTLN	确认接收数据的长度。	

(a) FIFO 端口的选择

各 FIFO 端口能选择的管道表如表 29.24 所示。通过 C/DnFIFOSEL 寄存器的 CURPIPE 位选择要存取的管道。在选择管道后，必须确认是否能正确地读到所写的 CURPIPE 值（如果读到上一轮的管道号，就表示此控制器正处于管道变更处理中），然后确认 FRDY 位为“1”并且存取 FIFO 端口。

必须通过 MBW 位选择要存取的总线宽度。缓冲存储器的存取方向取决于 PIPEnCFG 寄存器的 DIR 位，但是 DCP 取决于 ISEL 位。

表 29.24 各管道的 FIFO 端口存取表

管道	存取方法	能使用的端口
DCP	CPU 存取	CFIFO 端口寄存器
管道 1 ~ 9	CPU 存取	CFIFO 端口寄存器
	DMA 存取	D0FIFO/D1FIFO 端口寄存器

(b) REW 位

使用 C/DnFIFOSEL 寄存器的 REW 位，能暂停当前存取中的管道存取而对其他管道进行存取，然后再继续当前管道的处理。

如果通过在设定 C/DnFIFOSEL 寄存器的 CURPIPE 位的同时将 REW 位置“1”来选择管道，就对缓冲存储器的读写指针进行复位，因此能从最初的字节开始读写。如果通过将 REW 位置“0”来选择管道，就不对缓冲

存储器的读写指针进行复位，能接着上一次选择继续读写数据。

为了存取 FIFO 端口，需要在选择管道后确认 FRDY 位为“1”。

(c) FIFO 端口的尾数数据存取方法

在读 FIFO 端口时，如果读数据小于 FIFO 端口选择寄存器的 MBW 位指定的位宽，就必须以 MBW 位指定的位宽读数据，并且通过软件删除不需要的数据。

在写 FIFO 端口时，如果写数据小于 FIFO 端口选择寄存器的 MBW 位指定的位宽，就必须按以下例子进行存取。在将 FIFO 端口的存取宽度设定为 32 位（MBW 位为“10”）时写 24 位数据的方法例子如下所示：

- 尾数数据的写操作例子 1：以 16 位和 8 位宽各写 1 次

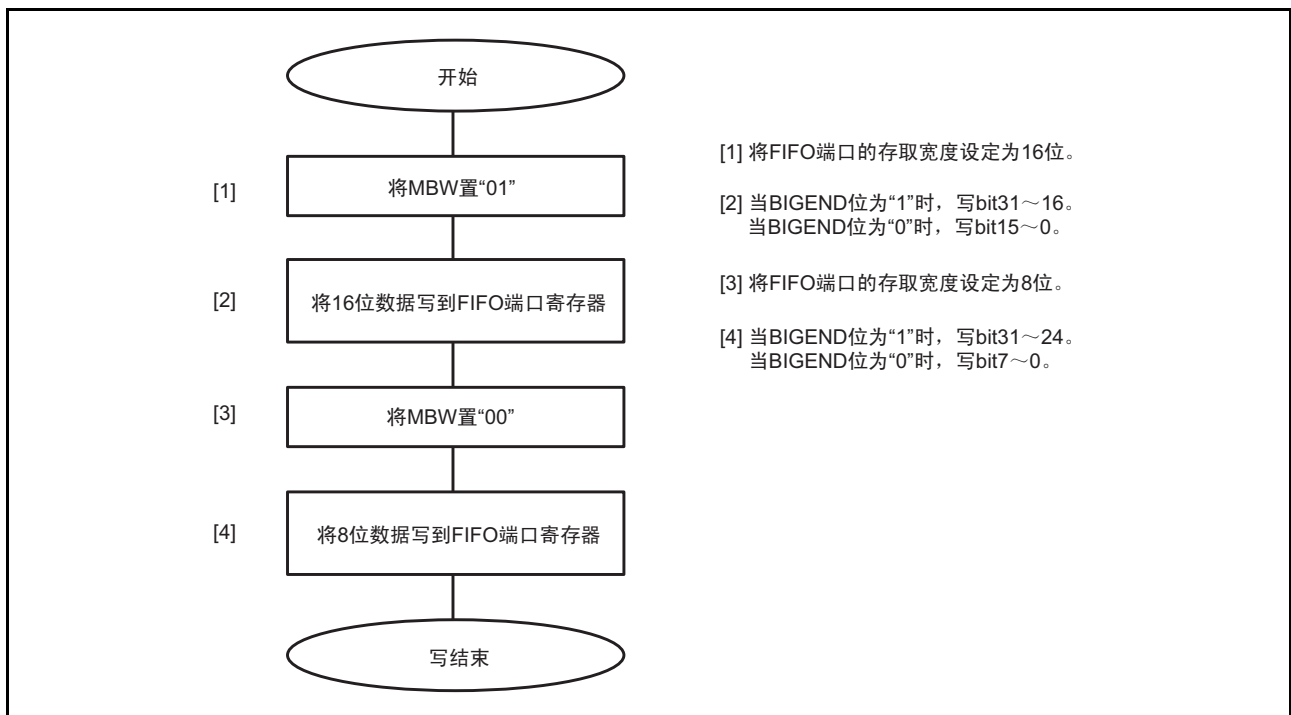


图 29.12 FIFO 端口的尾数数据的写操作例子 1

- 尾数数据的写操作例子 2：以 8 位宽写 3 次

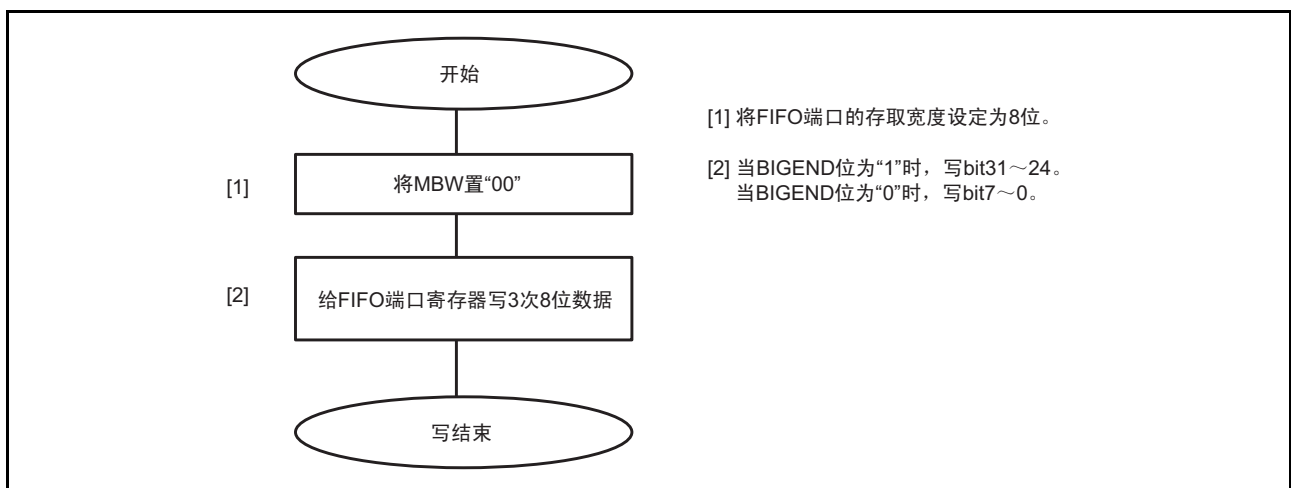


图 29.13 FIFO 端口的尾数数据的写操作例子 2

(d) 管道为接收方向时的 MBW 位变更方法

当指定管道为接收方向时，必须在设定 CURPIPE 位同时写 FIFO 端口选择寄存器 (CFIFOSEL、D0FIFOSEL、D1FIFOSEL) 的 MBW 位。如果通过 CFIFO 寄存器设定 DCP (CURPIPE 位为“000”)，就必须在设定 CURPIPE 位或者 ISEL 位的同时写 MBW 位。

要对当前正在设定的管道只更改 MBW 位时，必须按照以下步骤进行。但是，一旦开始处理缓冲存储器的读操作，就不能在读完全部数据前更改 MBW 位。

当所选 CURPIPE 位为写方向的缓冲存储器时，只能通过设定 MBW 位进行更改。

但是，一旦开始处理缓冲存储器的写操作，就不能从 8 位切换为 16 位或者 32 位，也不能从 16 位切换为 32 位。

- DFIFO0、DFIFO1 或者 CFIFO 的 CURPIPE 位设定为非 DCP (000) 的情况



图 29.14 DFIFO0、DFIFO1 或者 CFIFO 的 CURPIPE 位设定为非 DCP (000) 时的 MBW 更改例子

- CFIFO的CURPIPE位设定为DCP（000）的情况

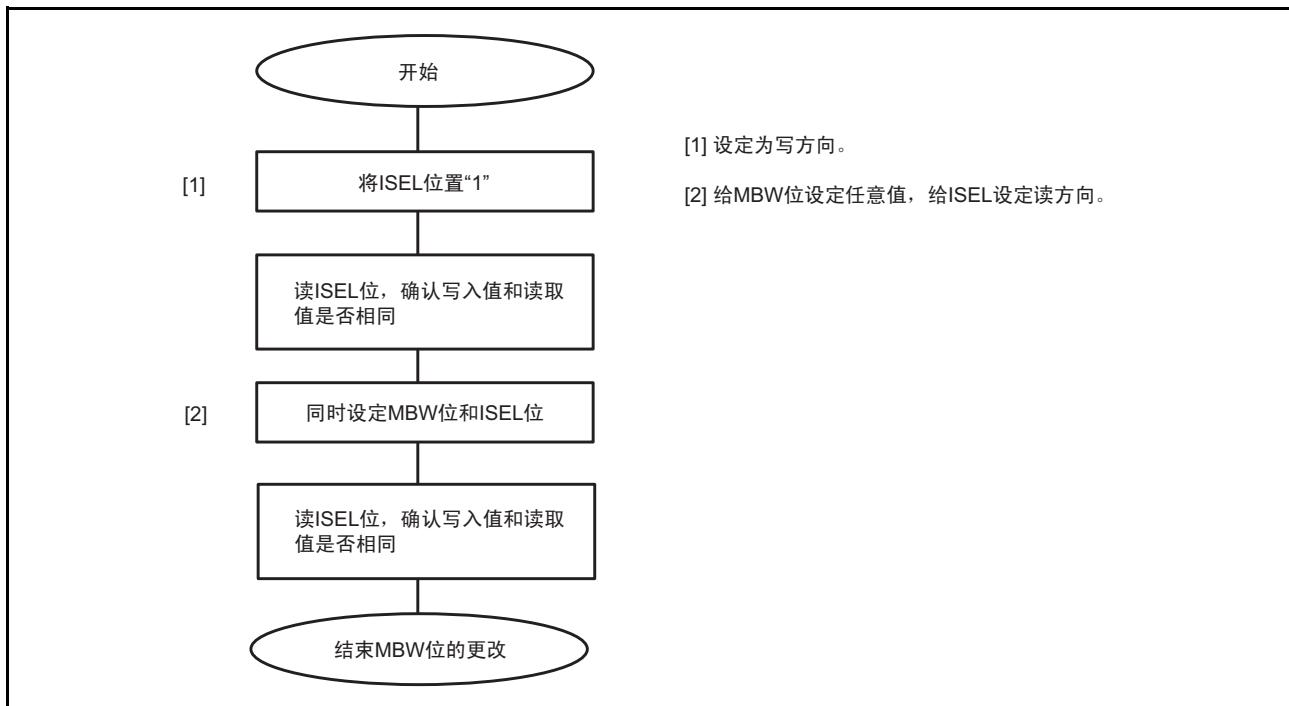


图 29.15 CURPIPE 设定为 DCP（000）时的 MBW 更改例子

(3) DMA 传送（D0FIFO/D1FIFO 端口）

(a) DMA 传送的概要

对于管道 1 ~ 9，能通过直接存储器存取控制器存取 FIFO 端口。当设定为 DMA 传送的管道缓冲器为可存取状态时，输出 DMA 传送请求。

必须通过 DnFIFOSEL 寄存器的 MBW 位选择 FIFO 端口的传送单位，通过 CURPIPE 位选择 DMA 传送的管道。不能在 DMA 传送过程中更改所选的管道。

(b) DMA 传送结束的自动识别

此模块能通过控制 DMA 传送结束信号的输入，结束 DMA 传送的 FIFO 数据写操作。如果进行直接存储器存取控制器的 DMA 传送计数器（DMATCR）所设次数的 DMA 传送，就从直接存储器存取控制器输出 DMA 传送结束信号。如果对传送结束信号进行采样，就将缓冲存储器设定为可发送状态（和 BVAL 位为“1”时的状态相同）。能通过 DnFBCFG 寄存器的 TENDE 位设定是否对 DMA 传送结束信号进行采样。

在将 DMA 的传送长度设定为 16 字节时，不能使用此功能。

(c) DnFIFO 自动清除模式（D0FIFO/D1FIFO 端口为读方向）

通过将 DnFIFOSEL 寄存器的 DCLRM 位置“1”，此模块在读完缓冲存储器的数据时自动清除所选管道的缓冲存储器。

各种设定状态下的接收信息包和清除缓冲存储器的处理关系如表 29.25 所示。如表 29.25 所示，缓冲器的清除条件因 BFRE 位的设定值而不同。即使在需要清除的状态下，只要使用 DCLRM 位，就不需要清除缓冲器，而且不通过软件就能进行 DMA 传送。

只在缓冲存储器为读方向时，才能设定此功能。

表 29.25 接收信息包和清除缓冲存储器的处理关系表

寄存器的设定 接收信息包时的缓冲器状态	DCLR M=0		DCLR M=1	
	BFRE=0	BFRE=1	BFRE=0	BFRE=1
缓冲器满	不需要清除	不需要清除	不需要清除	不需要清除
接收到 Zero-Length 包	需要清除	需要清除	不需要清除	不需要清除
接收到正常的短包	不需要清除	需要清除	不需要清除	不需要清除
事务计数结束	不需要清除	需要清除	不需要清除	不需要清除

29.4.5 控制传送 (DCP)

控制传送的数据阶段的数据传送使用默认控制管道 (DCP)。DCP 的缓冲存储器是控制读和控制写共用的固定区，为 256 字节的单缓冲器。只能通过 CFIFO 端口存取缓冲存储器。

(1) 选择主机控制器功能时的控制传送

(a) 设置阶段

USBREQ 寄存器、USBVAL 寄存器、USBINDX 寄存器和 USBLENG 寄存器用于发送设置事务的 USB 请求。如果将设置包的数据写到寄存器并且给 DCPCTR 寄存器的 SUREQ 位写“1”，就将设定的数据作为设置事务进行发送。一旦事务结束，就将 SUREQ 位清“0”。在 SUREQ 位为“1”期间，不能操作上述的 USB 请求寄存器。通过 DCPMAXP 寄存器的 DEVSEL 位指定设置事务的设备地址。

如果发送事务，就会根据外围设备的应答产生中断请求 (INTSTS1 寄存器的 SIGN 位和 SACK 位)。能通过此中断请求来确认设置事务的结果。

设置事务的数据包与 DCPCTR 寄存器的 SQMON 位的内容无关，随时发送 DATA0 的数据包 (USB 请求)。

(b) 数据阶段

使用 DCP 缓冲存储器进行数据传送。

必须通过 CFIFOSEL 寄存器的 ISEL 位指定 DCP 缓冲存储器的存取方向，通过 DCPCFG 寄存器的 DIR 位指定传送方向。

数据阶段的第 1 个数据包需要将数据 PID 作为 DATA1 进行通信。如果通过 DCPCTR 寄存器的 SQSET 位将数据 PID 设定为 DATA1 并且将 PID 位设定为 BUF，就执行事务。通过 BRDY 中断或者 BEMP 中断检测数据传送的结束。

在控制写传送的情况下，当发送数据为最大信息包长度的整数倍时，必须最后发送 Zero-Length 包。

(c) 状态阶段

这是与数据阶段反方向的 Zero-Length 包的数据传送。和数据阶段相同，使用 DCP 缓冲存储器进行数据传送。以数据阶段相同的步骤执行事务。

状态阶段的数据包需要将数据 PID 作为 DATA1 进行通信。必须通过 DCPCTR 寄存器的 SQSET 位将数据 PID 设定为 DATA1。

对于 Zero-Length 包的接收，必须在发生 BRDY 中断后，通过 CFIFOCTR 寄存器的 DTLN 位确认接收数据长度，然后通过 BCLR 位清除缓冲存储器。

(2) 选择功能控制器功能时的控制传送

(a) 设置阶段

此模块必须对正常的设置包进行 ACK 应答。设置阶段的此模块的运行如下所示：

1. 如果接收到新的设置包，此模块就将以下的位置位：
 - 将 INTSTS0 寄存器的 VALID 位置 “1”。
 - 将 DCPCTR 寄存器的 PID 位设定为 NAK。
 - 将 DCPCTR 寄存器的 CCPL 位置 “0”。
2. 如果接收到设置包之后的数据包，此模块就将 USB 请求的参数保存到 USBREQ 寄存器、USBVAL 寄存器 USBINDX 寄存器和 USBLENG 寄存器。

必须在将 VALID 位置 “0” 后对控制传送进行应答处理。在 VALID 位为 “1” 的状态下，不能将 PID 位设定为 BUF，也不能结束数据阶段。

通过 VALID 位的功能，此模块能在控制传送过程中接收到新的 USB 请求时中止正在处理的请求处理，并且对最新的请求进行应答。

此模块自动判别接收的 USB 请求的方向位 (bmRequestType 的 bit8) 和请求数据的长度 (wLength)，并且识别控制读传送、控制写传送和控制写无数据传送，进行阶段转移的管理。对于错误的顺序，发生控制传送阶段转移中断的顺序错误并且通知软件。有关此模块的阶段管理，请参照图 29.7。

(b) 数据阶段

必须通过 DCP 进行接收的 USB 请求所对应的数据传送。在存取 DCP 缓冲存储器前，必须通过 CFIFOSEL 寄存器的 ISEL 位指定存取方向。

通过将 DCPCTR 寄存器的 PID 位设定为 BUF，执行事务。

通过 BRDY 中断或者 BEMP 中断检测数据传送的结束。

在控制写传送时，必须使用 BRDY 中断；在控制读传送时，必须使用 BEMP 中断。

如果传送数据大于 DCP 缓冲存储器的大小，就在控制写传送时使用 BRDY 中断进行数据传送，在控制读传送时使用 BEMP 中断进行数据传送。

在高速运行时的控制写传送中，根据缓冲存储器的状态进行 NYET 握手信号应答。

(c) 状态阶段

在 DCPCTR 寄存器的 PID 位为 BUF 的状态下，通过将 CCPL 位置 “1” 来结束控制传送。

在进行上述的设定后，根据设置阶段确定的数据传送方向，此模块自动执行状态阶段。具体运行如下：

- 控制读传送的情况
此模块接收 USB 主机的 Zero-Length 包并且发送 ACK 应答。
- 控制写传送和无数据控制传送的情况
此模块发送 Zero-Length 包并且接收 USB 主机的 ACK 应答。

(d) 控制传送自动应答功能

此模块对正常的 SET_ADDRESS 请求进行自动应答。在 SET_ADDRESS 请求中发生以下错误时，需要进行应答。

- bmRequestType≠H'00
- wIndex≠H'00
- wLength≠H'00
- wValue>H'7F
- DVSQ=011 (Configured)

需要对 SET_ADDRESS 以外的全部请求进行应答。

29.4.6 批量传送（管道 1 ~ 5）

批量传送能选择缓冲存储器的使用方法（设定单缓冲器 / 双缓冲器或者连续 / 非连续传送模式）。缓冲存储器大小最大能设定 2K 字节。此模块管理缓冲存储器的状态并且对 PING 包 / NYET 握手信号进行自动应答。

(1) 选择主机控制器功能时的 PING 包控制

通过此模块自动发送 OUT 方向的 PING 包。

初始状态如下所示，在 PING 包的发送状态下，通过接收 ACK 握手信号来发送 OUT 包。如果在 OUT 事务中接收 NAK 或者 NYET，就返回到 PING 包的发送状态。

1. 设定 OUT 数据发送模式。
2. 发送 PING 包。
3. 接收 ACK 握手信号。
4. 发送 OUT 数据包。
5. 接收 ACK 握手信号。
(重复 4. 和 5.)
6. 发送 OUT 数据包。
7. 接收 NAK/NYET 握手信号。
8. 发送 PING 包。

通过上电复位、NYET/NAK 握手信号的接收、顺序交替位的清除（SQCLR）和缓冲器的清除（ACLRM），此模块返回到 PING 包的发送状态。

(2) 选择功能控制器功能时的 NYET 握手信号控制

批量传送和控制传送中的接收权标和应答一览表如表 29.26 所示。在批量传送和控制传送中，如果在接收 OUT 权标时缓冲存储器只有 1 个信息包的空间，此模块就进行 NYET 应答。但是，在接收短包时，即使在此条件下也不进行 NYET 包的应答而进行 ACK 应答。

表 29.26 接收权标的应答一览表

DCPCTR.PID 位的设定值	缓冲存储器的状态	权标	应答	备注
NAK/STALL	—	SETUP	ACK	—
	—	IN/OUT/PING	NAK/STALL	—
BUF	—	SETUP	ACK	—
	RCV-BRDY1	OUT/PING	ACK	在接收 OUT 权标时，接收数据包。
	RCV-BRDY2	OUT	NYET	接收数据包，通知不能接收。
	RCV-BRDY2	OUT (Short)	ACK	接收数据包，通知能接收。
	RCV-BRDY2	PING	ACK	通知能接收。
	RCV-NRDY	OUT/PING	NAK	通知不能接收。
	TRN-BRDY	IN	DATA0/1	发送数据包。
	TRN-NRDY	IN	NAK	通知不能发送。

【符号说明】

RCV-BRDY1: 在接收 OUT/PING 权标时，缓冲存储器至少有 2 个信息包的空间。

RCV-BRDY2: 在接收 OUT 权标时，缓冲存储器只有 1 个信息包的空间。

RCV-NRDY: 在接收 PING 权标时，缓冲存储器没有空间。

TRN-BRDY: 在接收 IN 权标时，缓冲存储器有发送数据。

TRN-NRDY: 在接收 IN 权标时，缓冲存储器没有发送数据。

29.4.7 中断传送（管道 6 ~ 9）

当选择功能控制器功能时，此模块根据主机控制器管理的周期进行中断传送。在中断传送的情况下，忽视 PING 包（不应答），不发送 NYET 握手信号而进行 ACK、NAK、STALL 应答。

当选择主机控制器功能时，能通过间隔计数器设定权标的发行时序。即使在 OUT 方向的传送中，也不发行 PING 权标而发行 OUT 权标。

此模块不支持中断传送的 High-Bandwidth 传送。

(1) 选择主机控制器功能时的中断传送的间隔计数器

在进行中断传送时，给 PIPEPERI 寄存器的 IITV 位设定事务间隔。此控制器根据所设定的间隔发行中断传送的权标。

(a) 计数器的初始化

此控制器对间隔计数器进行初始化的条件如下。

- 上电复位
对 IITV 位进行初始化。
- 通过 ACLRM 进行的缓冲存储器初始化
不对 IITV 位而对计数值进行初始化。通过将 ACLRM 位置“0”，从头开始 IITV 设定值的计数。

必须注意：在以下情况下，不对间隔计数器进行初始化：

- USB 总线复位或者 USB 挂起
不对 IITV 位进行初始化。通过将 UACT 位置“1”，从进入 USB 总线复位或者 USB 挂起状态前的值开始计数。

(b) 在产生权标时不能发送和接收的运行

在以下情况下，即使在产生权标时也不产生权标，而在下一个间隔尝试执行事务。

- 将 PID 位设定为 NAK 或者 STALL。
- 在 IN 方向（接收）的传送中发送权标时，缓冲存储器没有空间。
- 在 OUT 方向（发送）的传送中发送权标时，缓冲存储器没有发送数据。

29.4.8 等时传送（管道 1 和管道 2）

此模块具有以下等时传送的功能：

- 等时传送的错误信息通知
- 间隔计数器（由 IITV 位指定）
- 等时 IN 传送数据的设置控制（IDLY 功能）
- 等时 IN 传送缓冲器的转储清除功能（由 IFIS 位指定）

此模块不支持等时传送的 High-Bandwidth 传送。

当选择主机控制器功能时，如果 2 个管道同时用作等时传送，就必须遵循 Universal Serial Bus Revision 2.0 Specification “5.6.3 Isochronous Transfer Packet Size Constraints” 中记载的信息包限制。

(1) 等时传送的错误检测

此模块具有检测以下错误信息的功能，以便能通过软件管理发生的等时传送错误。此模块检测错误的优先级以及错误检测引起的中断如表 29.27 和表 29.28 所示。

1. PID 错误
 - 接收包的 PID 不正确。
2. CRC 错误和位填充错误
 - 接收包的 CRC 有错误或者位填充不正确。
3. 超过最大信息包长度
 - 接收包的数据长度超过最大信息包长度的设定值。
4. 超限和欠载错误
 - 选择主机控制器功能的情况
在 IN 方向（接收）的传送中发送权标时，缓冲存储器没有空间。
在 OUT 方向（发送）的传送中发送权标时，缓冲存储器没有数据。
 - 选择功能控制器功能的情况
在 IN 方向（发送）的传送中接收 IN 权标时，缓冲存储器没有数据。
在 OUT 方向（接收）的传送中接收 OUT 权标时，缓冲存储器没有空间。
5. 间隔错误
 - 当选择功能控制器功能时，将以下情况视为间隔错误：
 - 在等时 IN 传送时，不能在间隔帧中接收 IN 权标。
 - 在等时 OUT 传送时，在非间隔帧中接收到 OUT 权标。

表 29.27 接收权标时的错误检测

检测的 优先级	错误	产生的中断和状态
1	PID 错误	在选择主机控制器功能或者功能控制器功能时，都不产生中断（作为破损包被忽视）。
2	CRC 错误和位填充错误	在选择主机控制器功能或者功能控制器功能时，都不产生中断（作为破损包被忽视）。
3	超限和欠载错误	在选择主机控制器功能或者功能控制器功能时，都产生中断，并且将 OVRN 位置位。当选择主机控制器功能时，不发送权标；当选择功能控制器功能时，对 IN 权标发送 Zero-Length 包。对于 OUT 权标，不接收数据包。
4	间隔错误	当选择功能控制器功能时，产生 NRDY 中断；当选择主机控制器功能时，不产生 NRDY 中断。

表 29.28 接收数据包时的错误检测

检测的 优先级	错误	产生的中断和状态
1	PID 错误	不产生中断（作为破损包被忽视）。
2	CRC 错误和位填充错误	在选择主机控制器或者功能控制器功能时，都产生 NRDY 中断，并且将 CRCE 位置位。
3	最大信息包长度超过错误	在选择主机控制器或者功能控制器功能时，都产生 BEMP 中断，并且将 PID 位设定为 STALL。

(2) DATA-PID

此模块不支持 High-Bandwidth 传送。当选择功能控制器功能时，对接收的 PID 进行以下的处理：

1. IN 方向
 - DATA0: 作为数据包的PID进行发送。
 - DATA1: 不发送。
 - DATA2: 不发送。
 - mData: 不发送。
2. OUT 方向（全速运行时）
 - DATA0: 作为数据包的PID，正常接收。
 - DATA1: 作为数据包的PID，正常接收。
 - DATA2: 忽视此信息包。
 - mData: 忽视此信息包。
3. OUT 方向（高速运行时）
 - DATA0: 作为数据包的PID，正常接收。
 - DATA1: 作为数据包的PID，正常接收。
 - DATA2: 作为数据包的PID，正常接收。
 - mData: 作为数据包的PID，正常接收。

(3) 间隔计数器

能通过 PIPEPERI 寄存器的 IITV 位设定等时传送的间隔。当选择功能控制器功能时，通过间隔计数器实现表 29.29 所示的功能；当选择主机控制器功能时，通过间隔计数器生成权标的发行时序。选择主机控制器功能时的间隔计数器的运行和中断传送的运行相同。

表 29.29 选择功能控制器功能时的间隔计数器的功能

传送方向	功能	检测条件
IN	发送缓冲器转储清除功能	在等时 IN 传送时，不能在间隔帧中正常接收 IN 权标。
OUT	权标未接收的通知	在等时 OUT 传送时，不能在间隔帧中正常接收 OUT 权标。

通过接收 SOF 或者内插 SOF 进行间隔计数，因此即使 SOF 破损也能保证等时性。能设定的帧间隔为 2^{IITV} 帧或者 $2^{IITV}\mu$ 帧。

(a) 选择功能控制器功能时的间隔计数器的初始化

在以下条件下，此模块对间隔计数器进行初始化。

- 上电复位
对 IITV 位进行初始化。
- 通过 ACLRM 进行的缓冲存储器初始化
不对 IITV 位而对计数值进行初始化。
- USB 总线复位

在对间隔计数器进行初始化后正常传送信息包，然后如果满足以下 1. 或者 2. 的条件，就开始进行间隔计数。

1. 在 PID 位为 BUF 的状态下，在对 IN 权标发送数据后接收 SOF。
2. 在 PID 位为 BUF 的状态下，在接收 OUT 权标的数据后接收 SOF。

在以下条件下，不对间隔计数器进行初始化。

1. 将 PID 位设定为 NAK 或者 STALL。
间隔定时器不停止计数，尝试在下一个间隔执行事务。
2. USB 总线复位或者 USB 挂起
不对 IITV 位进行初始化。如果接收到 SOF，就从接收前的值开始计数。

(b) 选择主机控制器功能时的间隔计数和传送控制

此模块根据 IITV 位的设定值控制权标的发行间隔。此模块以 2^{IITV} 帧 (μ) 为 1 个间隔对所选管道发行权标。此模块对于和高速 HUB 连接的全速 / 低速外围设备进行通信的管道，每 1ms 帧进行一次间隔计数。此模块从将 PID 位设定为 BUF 的下一个 (μ) 帧开始权标发行间隔的计数。

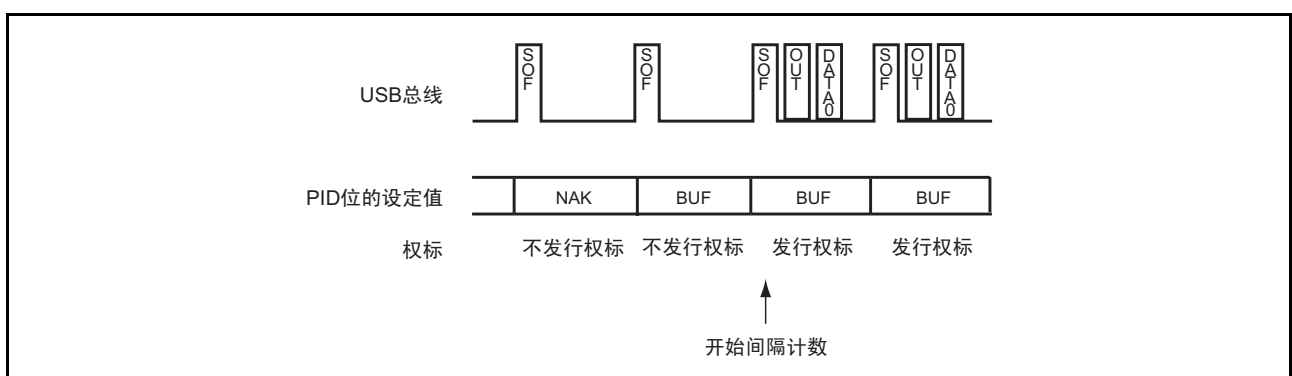


图 29.16 IITV 位为“0”时的权标发行情况

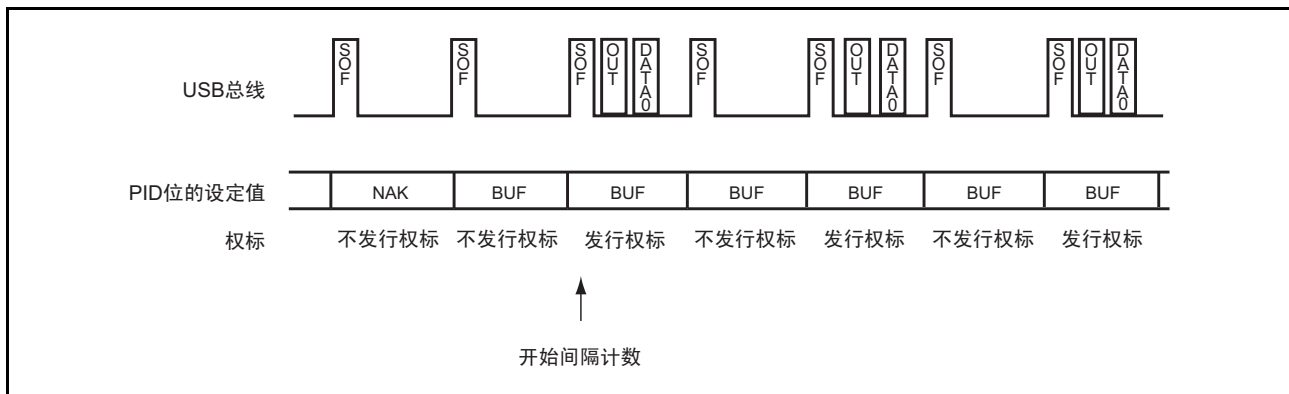


图 29.17 IITV 位为“1”时的权标发行情况

在所选管道的传送类型为等时传送时，此模块根据控制权标的发行间隔进行以下运行。即使满足 NRDY 中断产生条件，此模块也发行权标。

1. 所选管道为等时 IN 传送管道的情况

发行 IN 权标，在没有正常从外围设备接收到信息包时（不应答或者信息包错误等），产生 NRDY 中断。如果在因 FIFO 缓冲器满而使此模块不能接收数据的状态下（这是因为从 FIFO 缓冲器读数据的速度慢）到了发行 IN 权标的时间，此模块就将 OVRN 位置“1”，产生 NRDY 中断。

所选管道为等时 OUT 传送管道的情况

如果在 FIFO 缓冲器中没有能发送的数据的状态下（这是因为给 FIFO 缓冲器写数据的速度慢）到了发行 OUT 权标的时间，此模块就将 OVRN 位置“1”，产生 NRDY 中断，并且发送 Zero-Length 包。

权标发行间隔的复位条件是上电复位以及将 ACLRM 位置“1”。

(c) 选择功能控制器功能时的间隔计数和传送控制

1. 所选管道为等时 OUT 传送管道的情况

在 IITV 位设定的各间隔（ μ ）帧中没有接收到 DATA 包时，此控制器产生 NRDY 中断。

在因 DATA 包发生 CRC 错误等或者因 FIFO 缓冲器满而不能接收数据时，此模块也产生 NRDY 中断。

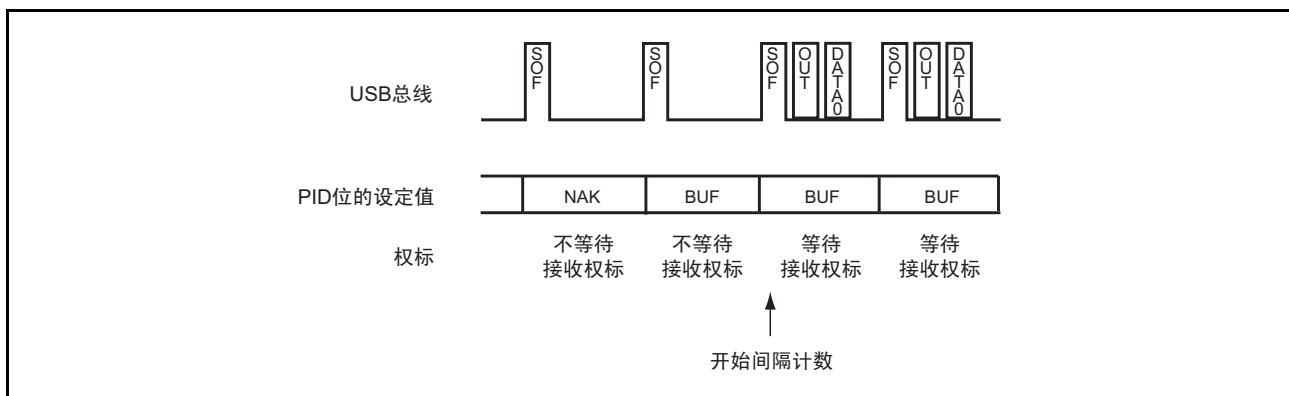
NRDY 中断的产生时序是接收到 SOF 包时。即使 SOF 包破损，也通过内插功能在应该接收 SOF 时产生中断。

但是，当 IITV 位不是“0”时，就在间隔计数开始后的各间隔接收 SOF 包时产生 NRDY 中断。

如果在启动间隔定时器后将 PID 位设定为 NAK，此模块即使接收到 SOF 包也不产生 NRDY 中断。

间隔计数的开始条件因 IITV 位的设定值而不同。

- 当 IITV 位为“0”时，从将所选管道的 PID 位改为 BUF 后的下一帧（ μ ）开始间隔计数。

图 29.18 IITV 位为“0”时的（ μ ）帧和是否等待接收权标的关系

- 当 IITV 位不为“0”时，从将所选管道的 PID 位改为 BUF 后正常接收到最初的 DATA 包时开始间隔计数。

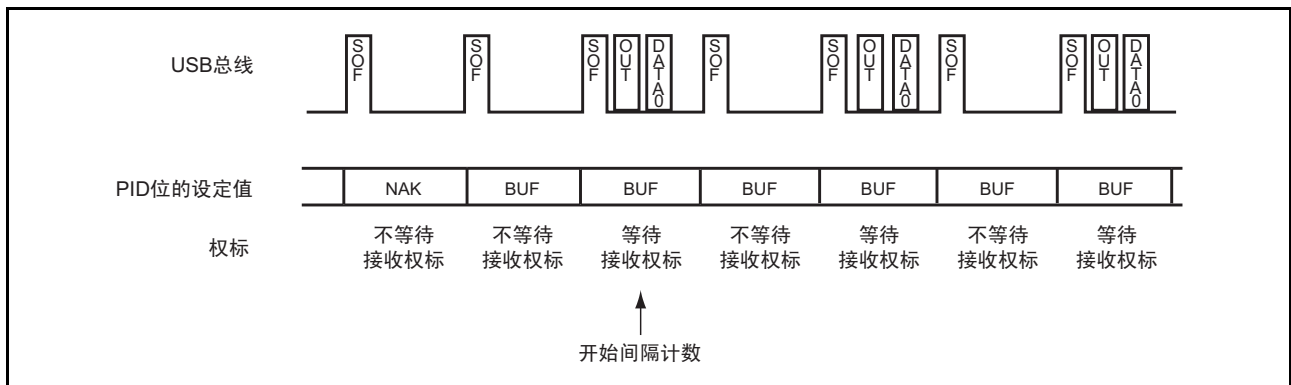


图 29.19 IITV 位为“1”时的 (μ) 帧和是否等待接收权标的关系

2. 所选管道为等时 IN 传送管道的情况

在 IFIS 位为“1”时使用。当 IFIS 位为“0”时，与 IITV 位的设定值无关，在对接收的权标进行应答后发送数据包。

在 IFIS 位为“1”并且 FIFO 缓冲器有能发送的数据的状态下，如果在 IITV 位设定的各间隔 (μ) 帧中没有接收到 IN 权标，此模块就清除 FIFO 缓冲器。

即使在因 IN 权标发生 CRC 错误等的总线错误而使此模块不能正常接收时，也清除 FIFO 缓冲器。

FIFO 缓冲器的清除时序是接收到 SOF 包时。即使 SOF 包破损，也通过内插功能在应该接收 SOF 时清除 FIFO 缓冲器。

间隔计数的开始条件因 IITV 位的设定值而不同（与 OUT 时相同）。

选择功能控制器功能时的间隔计数清除条件如下：

- 上电复位
- 当将 ACLRM 位置“1”时
- 当此模块检测到 USB 总线复位时

(4) 选择功能控制器功能时的等时传送发送数据的设置

当选择功能控制器功能时，在此模块的等时数据发送过程中给缓冲存储器写数据后，能在检测到 SOF 包的下一帧发送数据包，此功能称为等时传送的发送数据设置功能。能通过此功能特定开始发送的帧。

在缓冲存储器用作双缓冲器时，即使写完 2 个缓冲器面，可传送状态的缓冲存储器也只为先写完的 1 个面。因此，即使在同一帧接收到多个 IN 权标时，被发送的缓冲存储器也只为 1 个包的数据。

在接收到 IN 权标时，如果缓冲存储器为可发送状态，就传送数据并且进行正常的应答。如果缓冲存储器为不可发送状态，就发送 Zero-Length 包并且发生欠载错误。

此模块通过 IITV 位为“0”（每帧）时的等时传送发送数据设置功能进行的发送例子如图 29.20 所示。

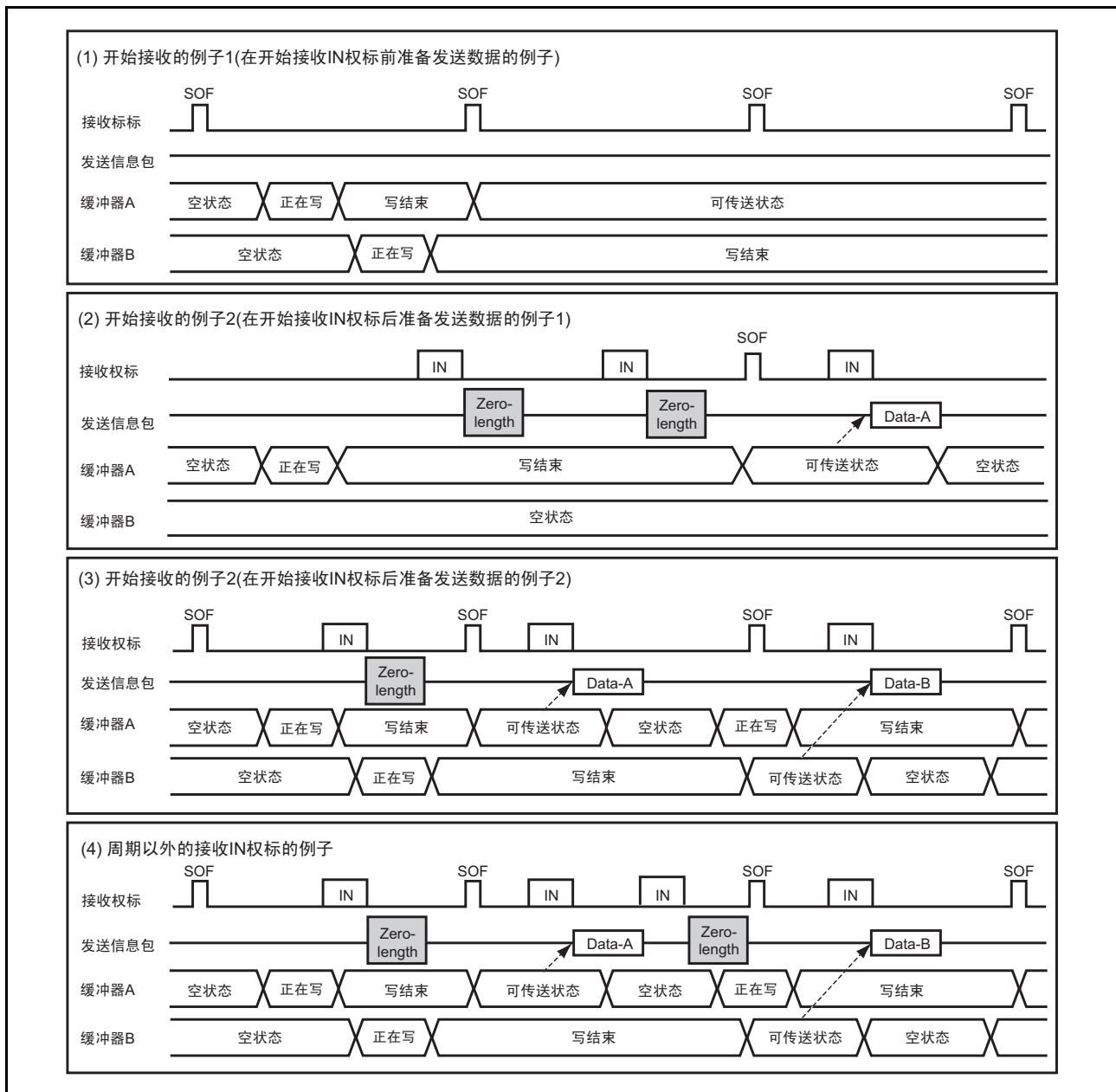


图 29.20 数据设置功能的运行例子

(5) 选择功能控制器功能时的等时传送发送缓冲器的转储清除

当选择功能控制器功能时，如果此模块在等时数据发送时的间隔帧中没有接收到 IN 权标而接收到下一帧的 SOF 或者 μ SOF 包，就作为 IN 权标破损进行处理，并且清除处于可发送状态的缓冲器，使该缓冲器变为可写状态。

此时，如果使用双缓冲器并且写完 2 个缓冲器面，放弃的缓冲存储器就被看作同一间隔帧发送的数据，并且将接收 SOF 或者 μ SOF 包时未放弃的缓冲存储器设定为可传送状态。

缓冲器转储清除功能的运行开始时序因 IITV 位的设定值而不同。

- IITV 位为 “0” 的情况
从管道有效的下一帧开始进行缓冲器转储清除。
- IITV 不为 “0” 的情况
从最初的正常事务后进行缓冲器转储清除。

此模块的缓冲器转储清除功能的运行例子如图 29.21 所示。但是，对于所设间隔以外的权标（间隔帧前的权标），根据数据设置状态，发送写数据或者发送作为欠载错误的 Zero-Length 包。

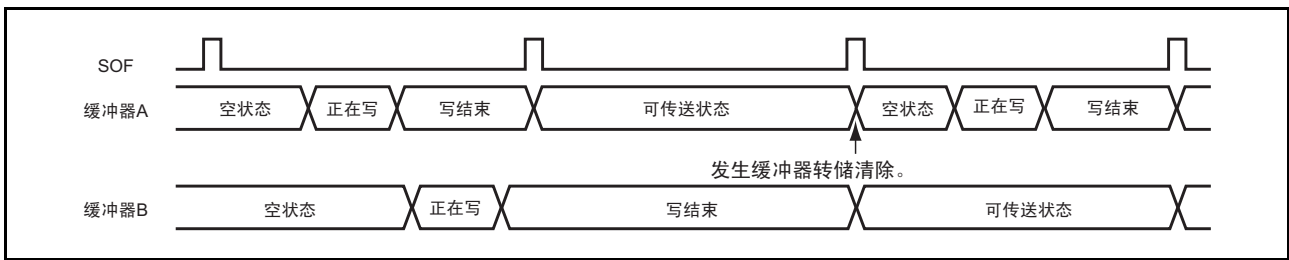


图 29.21 缓冲器转储清除功能的运行例子

此模块的间隔错误发生例子如图 29.22 所示。间隔错误有以下 5 种。在图中的①时序发生间隔错误并且进行缓冲器转储清除功能的运行。

如果在 IN 传送时发生间隔错误，缓冲器转储清除功能就运行；如果在 OUT 传送时发生间隔错误，就产生 NRDY 中断。

必须通过 OVRN 位判断接收包错误等 NRDY 中断和超限错误的区别。

图中阴影部分的权标是根据缓冲存储器的状态进行的应答。

1. IN 方向

- 如果缓冲器为可传送状态，就传送数据并且进行正常的应答。
- 如果缓冲器为不可传送状态，就发送 Zero-Length 包并且发生欠载错误。

2. OUT 方向

- 如果缓冲器为可接收状态，就接收数据并且进行正常的应答。
- 如果缓冲器为不可接收状态，就放弃数据并且发生超限错误。

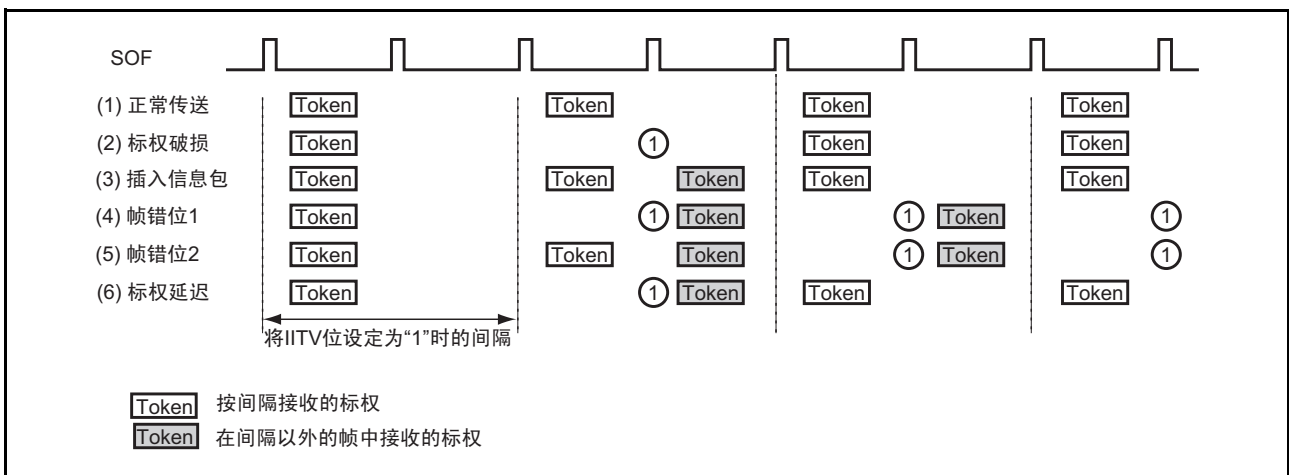


图 29.22 IITV 位为 “1” 时的间隔错误发生例子

29.4.9 SOF 内插功能

当选择功能控制器功能时，如果因 SOF 包的破损或者丢失而在 1ms（全速运行时）或者 125 μ s（高速运行时）间隔内无法接收到 SOF 包，此模块就内插 SOF。在 SYSCFG 寄存器的 USBE 位和 SCKE 位都为“1”并且接收 SOF 包时，开始内插 SOF。在以下条件下，对内插功能进行初始化：

- 上电复位
- USB 总线复位
- 检测到挂起

SOF 内插按以下规格进行：

- 帧间隔（125 μ s 或者 1ms）取决于复位握手信号协议的结果。
- 在接收到 SOF 包前不进行内插功能。
- 在接收到最初的 SOF 包后，通过内部时钟 48MHz 对 125 μ s 或者 1ms 进行计数并且进行内插。
- 在接收到第 2 次及以后的 SOF 包后，使用前一次接收的间隔进行内插。
- 在挂起时或者接收到 USB 总线复位时不进行内插。
（在高速运行的挂起转移时，在接收最后包后的 3ms 内继续进行内插）

此模块根据接收的 SOF 包进行以下功能的运行。如果 SOF 包丢失，就内插 SOF，因此能继续正常运行。

- 帧号和 μ 帧号的更新
- SOFR 中断和 μ SOF 的锁定
- 等时传送的间隔计数

如果在全速运行时丢失 SOF 包，就不更新 FRMNUM 寄存器的 FRNM 位；如果在高速运行时丢失 μ SOF 包，就不更新 UFRMNUM 寄存器的 UFRNM 位。但是，如果丢失 UFRNM 位为“000”的 μ SOF 包，就不更新 FRNM 位。此时，即使继续正常接收到 UFRNM 位不是“000”的 μ SOF 包，也不更新 FRNM 位。

29.4.10 管道的安排

(1) 事务发行条件

当选择主机控制器功能时，此模块在将 UACT 位置“1”后按表 29.30 所示的条件发行事务。

表 29.30 事务发行条件

事务	发行条件				
	DIR	PID	IITV0	缓冲器的状态	SUREQ
设置	—*1	—*1	—*1	—*1	设定 1
控制传送的数据阶段、状态阶段、批量传送	IN	BUF	无效	有接收区	—*1
	OUT	BUF	无效	有发送数据	—*1
中断传送	IN	BUF	有效	有接收区	—*1
	OUT	BUF	有效	有发送数据	—*1
等时传送	IN	BUF	有效	*2	—*1
	OUT	BUF	有效	*3	—*1

【注】 *1 表中的“—”表示是与权标的发行无关的条件。“有效”表示对于中断传送和等时传送，只在间隔计数器进行的传送帧中发行。“无效”表示与间隔计数器无关进行发行。

*2 不管有还是没有接收区，都发行事务。但是在没有接收区时，放弃接收数据。

*3 不管有还是没有发送数据，都发行事务。但是在没有发送数据时，发行 Zero-Length 包。

(2) 传送的安排

以下说明此模块帧内的传送安排方法。在发送 SOF 后，此模块按照以下所示的步骤进行传送。

1. 周期性传送的执行

按照管道 1→管道 2→管道 6→管道 7→管道 8→管道 9 的顺序查找管道，如果有能发行等时传送或者中断传送事务的管道，就发行事务。

2. 控制传送的设置事务

确认 DCP，如果能发行设置事务，就发行事务。

3. 批量、控制传送数据阶段和状态阶段的执行

按照 DCP→管道 1→管道 2→管道 3→管道 4→管道 5 的顺序查找管道，如果有能发行批量、控制传送数据阶段、控制传送状态阶段事务的管道，就执行事务。

在发行事务后，无论外围设备的应答是 ACK 还是 NAK，都移到下一个管道的事务。如果帧内有传送的时间，就重复执行 3。

(3) USB 通信的允许

能通过将 DVSTCTR 寄存器的 UACT 位置“1”，开始发送 SOF 或者 μ SOF，并且进入可发行事务的状态。

如果将 UACT 位置“0”，就停止发送 SOF 或者 μ SOF 而进入挂起状态。如果将 UACT 位从“1”改为“0”，就在发送下一个 SOF 或者 μ SOF 后停止发送。

29.5 使用时的注意事项

29.5.1 USB 收发器部电源

- 电源USB V_{CC} 、USB DV_{CC}^* 、USB UV_{CC}^* 必须和 V_{CC} 同电位。
- 电源USB APV_{CC} 、USB DPV_{CC}^* 必须和 PV_{CC} 同电位。
- 接地USB AV_{SS}^* 、USB DV_{SS}^* 、USB UV_{SS}^* 、USB APV_{SS}^* 、USB DPV_{SS}^* 必须和 V_{SS} 同电位。
- 模拟电源（USB AV_{CC} 、USB AV_{SS}^* 、USB APV_{CC} 、USB APV_{SS}^* ）必须和数字电源分离。

【注】 * SH7269（BGA）没有此引脚。

30. 数字视频解码器

30.1 特点

数字视频解码器由用于输入图像信号的 A/D 转换器、同步分离电路、Burst Controlled Oscillator (BCO)、2D 适应型 Y/C 分离电路、色度解码电路、数字钳位电路和输出调整电路构成。数字视频解码器的功能如表 30.1 所示。

表 30.1 数字视频解码器的功能

项目	功能
输入信号	图像信号 复合图像信号 (CVBS)
功能概要	<ul style="list-style-type: none"> • 用于输入图像信号的 A/D 转换器 VIN1 引脚和 VIN2 引脚的输入选择 内置同步端钳位。 内置 Programmable Gain Amp (PGA) (1.835dB ~ 8.023dB)。 内置 10 位精度流水线方式的 A/D 转换器。 • 同步分离 噪声消除 LPF、自动电平控制同步限幅器、水平 Auto Frequency Control (AFC)、 垂直递减计数、隔行扫描判定、Auto Gain Control (AGC) / 峰值限制器控制 • Burst Controlled Oscillator (BCO) 色度副载波播放、颜色体系判定 (※详细内容请参照表 30.3) • Y/C 分离 (※详细内容请参照表 30.2) 支持 NTSC 2D、PAL 2D 和 SECAM 1D。 • 色度解码 支持 NTSC、PAL 和 SECAM。 消色器、Auto Color Control (ACC)、TINT 校正、R-Y 轴校正 • 数字钳位 消隐脉冲钳位 (Y)、中心钳位 (Cb/Cr)、噪声检测 • 输出调整 对比度调整: 0 ~ 约 2[倍] 色彩调整 (Cb/Cr 独立): 0 ~ 约 2[倍]

表 30.2 Y/C 分离运行对应表

	YC 分离运行
NTSC-3.58	适应 2 维
NTSC-4.43	适应 2 维
PAL-M	适应 2 维
PAL-N	适应 2 维
PAL-4.43	适应 2 维
SECAM	1 维

表 30.3 颜色体系判定表

COLORSYS[1:0]	FSCMODE	FVMODE	判别结果
0: NTSC	0: 3.58MHz	don't care	NTSC-M
0: NTSC	1: 4.43MHz	don't care	NTSC-4.43
1: PAL	0: 3.58MHz	0: 50Hz	PAL-N
1: PAL	0: 3.58MHz	1: 60Hz	PAL-M
1: PAL	1: 4.43MHz	0: 50Hz	PAL-B、H、I、G、D
1: PAL	1: 4.43MHz	1: 60Hz	PAL-60
2: SECAM	—	—	SECAM
3: unknown	—	—	无法判别

30.2 框图

此模块的整体框图如下所示：

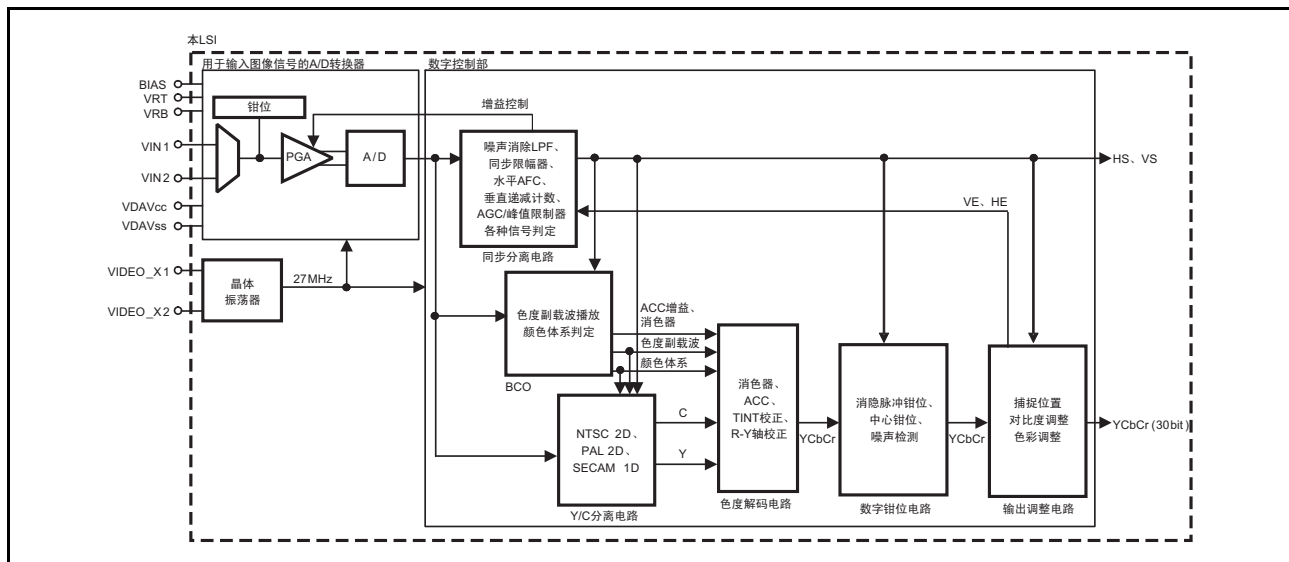


图 30.1 框图

30.3 输入 / 输出引脚

引脚结构如表 30.4 所示

表 30.4 引脚结构

区分	名称	引脚名	输入 / 输出	功能
信号	复合图像信号 输入	VIN1	输入	复合图像信号 (CVBS) 的输入引脚 1
		VIN2	输入	复合图像信号 (CVBS) 的输入引脚 2
时钟	晶体谐振器 / 外部时钟	VIDEO_X1	输入	连接用于数字视频解码器的晶体谐振器, VIDEO_X1 引脚还能输入外部时钟。
		VIDEO_X2	输出	
基准电压	TOP 基准电压	VRT	输出	用于输入图像信号的 A/D 转换器的 TOP 基准电压引脚必须通过 0.1 μ F 的电容器连接 VDAV _{SS} 。
	BOTTOM 基准电压	VRB	输出	用于输入图像信号的 A/D 转换器的 BOTTOM 基准电压引脚必须通过 0.1 μ F 的电容器连接 VDAV _{SS} 。
	基准电压	BIAS	输入	用于输入图像信号的 A/D 转换器的基准电压引脚必须通过 24k Ω ±1% 的电阻连接 VDAV _{SS} 。
电源	模拟电源	VDAV _{CC}	输入	用于输入图像信号的 A/D 转换器的电源引脚
	模拟接地	VDAV _{SS}	输入	用于输入图像信号的 A/D 转换器的接地引脚

30.4 寄存器说明

寄存器结构如表 30.5 所示:

表 30.5 寄存器结构

寄存器名	略称	R/W	地址	存取长度
ADC 控制寄存器 1	ADCCR1	R/W	H'FFFFA008	16
时序生成控制寄存器 (1)	TGCR1	R/W	H'FFFFA00E	16
时序生成控制寄存器 (2)	TGCR2	R/W	H'FFFFA010	16
时序生成控制寄存器 (3)	TGCR3	R/W	H'FFFFA012	16
同步分离控制寄存器 (1)	SYNSCR1	R/W	H'FFFFA01A	16
同步分离控制寄存器 (2)	SYNSCR2	R/W	H'FFFFA01C	16
同步分离控制寄存器 (3)	SYNSCR3	R/W	H'FFFFA01E	16
同步分离控制寄存器 (4)	SYNSCR4	R/W	H'FFFFA020	16
同步分离控制寄存器 (5)	SYNSCR5	R/W	H'FFFFA022	16
水平 AFC 控制寄存器 (1)	HAFCCR1	R/W	H'FFFFA024	16
水平 AFC 控制寄存器 (2)	HAFCCR2	R/W	H'FFFFA026	16
水平 AFC 控制寄存器 (3)	HAFCCR3	R/W	H'FFFFA028	16
垂直递减计数控制寄存器 (1)	VCDWCR1	R/W	H'FFFFA02A	16
数字钳位控制寄存器 (1)	DCPCR1	R/W	H'FFFFA030	16
数字钳位控制寄存器 (2)	DCPCR2	R/W	H'FFFFA032	16
数字钳位控制寄存器 (3)	DCPCR3	R/W	H'FFFFA034	16
数字钳位控制寄存器 (4)	DCPCR4	R/W	H'FFFFA036	16
数字钳位控制寄存器 (5)	DCPCR5	R/W	H'FFFFA038	16
数字钳位控制寄存器 (6)	DCPCR6	R/W	H'FFFFA03A	16

寄存器名	略称	R/W	地址	存取长度
数字钳位控制寄存器 (7)	DCPCR7	R/W	H'FFFFFFA03C	16
数字钳位控制寄存器 (8)	DCPCR8	R/W	H'FFFFFFA03E	16
噪声检测控制寄存器	NSDCR	R/W	H'FFFFFFA040	16
色同步锁定控制 / 色度解码控制寄存器	BTLCR	R/W	H'FFFFFFA042	16
色同步选通脉冲控制寄存器	BTGPCR	R/W	H'FFFFFFA044	16
ACC 控制寄存器 (1)	ACCCR1	R/W	H'FFFFFFA046	16
ACC 控制寄存器 (2)	ACCCR2	R/W	H'FFFFFFA048	16
ACC 控制寄存器 (3)	ACCCR3	R/W	H'FFFFFFA04A	16
TINT 控制寄存器	TINTCR	R/W	H'FFFFFFA04C	16
Y/C 延迟控制 / 色度解码控制寄存器	YDCR	R/W	H'FFFFFFA04E	16
AGC 控制寄存器 (1)	AGCCR1	R/W	H'FFFFFFA050	16
AGC 控制寄存器 (2)	AGCCR2	R/W	H'FFFFFFA052	16
峰值限制器的控制寄存器	PKLIMITCR	R/W	H'FFFFFFA054	16
超范围控制寄存器 (1)	RGORCR1	R/W	H'FFFFFFA056	16
超范围控制寄存器 (2)	RGORCR2	R/W	H'FFFFFFA058	16
超范围控制寄存器 (3)	RGORCR3	R/W	H'FFFFFFA05A	16
超范围控制寄存器 (4)	RGORCR4	R/W	H'FFFFFFA05C	16
超范围控制寄存器 (5)	RGORCR5	R/W	H'FFFFFFA05E	16
超范围控制寄存器 (6)	RGORCR6	R/W	H'FFFFFFA060	16
超范围控制寄存器 (7)	RGORCR7	R/W	H'FFFFFFA062	16
水平 AFC 相位比较器的反馈量调整寄存器	AFCPFCR	R/W	H'FFFFFFA07C	16
寄存器的更新控制寄存器	RUPDCR	R/W	H'FFFFFFA07E	16
同步分离状态 / 垂直周期读寄存器	VSYNCR	R	H'FFFFFFA080	16
水平周期读寄存器	HSYNCR	R	H'FFFFFFA082	16
数字钳位读寄存器 (1)	DCPSR1	R	H'FFFFFFA084	16
数字钳位读寄存器 (2)	DCPSR2	R	H'FFFFFFA086	16
噪声检测读寄存器	NSDSR	R	H'FFFFFFA08C	16
色度解码读寄存器 (1)	CROMASR1	R	H'FFFFFFA08E	16
色度解码读寄存器 (2)	CROMASR2	R	H'FFFFFFA090	16
同步分离读寄存器	SYNCSR	R	H'FFFFFFA092	16
AGC 控制读寄存器 (1)	AGCSR1	R	H'FFFFFFA094	16
AGC 控制读寄存器 (2)	AGCSR2	R	H'FFFFFFA096	16
Y/C 分离控制寄存器 (3)	YCSCR3	R/W	H'FFFFFFA104	16
Y/C 分离控制寄存器 (4)	YCSCR4	R/W	H'FFFFFFA106	16
Y/C 分离控制寄存器 (5)	YCSCR5	R/W	H'FFFFFFA108	16
Y/C 分离控制寄存器 (6)	YCSCR6	R/W	H'FFFFFFA10A	16
Y/C 分离控制寄存器 (7)	YCSCR7	R/W	H'FFFFFFA10C	16
Y/C 分离控制寄存器 (8)	YCSCR8	R/W	H'FFFFFFA10E	16
Y/C 分离控制寄存器 (9)	YCSCR9	R/W	H'FFFFFFA110	16
Y/C 分离控制寄存器 (11)	YCSCR11	R/W	H'FFFFFFA114	16
Y/C 分离控制寄存器 (12)	YCSCR12	R/W	H'FFFFFFA116	16
数字钳位控制寄存器 (9)	DCPCR9	R/W	H'FFFFFFA180	16

寄存器名	略称	R/W	地址	存取长度
Y/C 分离的色度信号滤波器 TAP 系数 (WA_F0) 寄存器	YCTWA_F0	R/W	H'FFFFA192	16
Y/C 分离的色度信号滤波器 TAP 系数 (WA_F1) 寄存器	YCTWA_F1	R/W	H'FFFFA194	16
Y/C 分离的色度信号滤波器 TAP 系数 (WA_F2) 寄存器	YCTWA_F2	R/W	H'FFFFA196	16
Y/C 分离的色度信号滤波器 TAP 系数 (WA_F3) 寄存器	YCTWA_F3	R/W	H'FFFFA198	16
Y/C 分离的色度信号滤波器 TAP 系数 (WA_F4) 寄存器	YCTWA_F4	R/W	H'FFFFA19A	16
Y/C 分离的色度信号滤波器 TAP 系数 (WA_F5) 寄存器	YCTWA_F5	R/W	H'FFFFA19C	16
Y/C 分离的色度信号滤波器 TAP 系数 (WA_F6) 寄存器	YCTWA_F6	R/W	H'FFFFA19E	16
Y/C 分离的色度信号滤波器 TAP 系数 (WA_F7) 寄存器	YCTWA_F7	R/W	H'FFFFA1A0	16
Y/C 分离的色度信号滤波器 TAP 系数 (WA_F8) 寄存器	YCTWA_F8	R/W	H'FFFFA1A2	16
Y/C 分离的色度信号滤波器 TAP 系数 (WB_F0) 寄存器	YCTWB_F0	R/W	H'FFFFA1A4	16
Y/C 分离的色度信号滤波器 TAP 系数 (WB_F1) 寄存器	YCTWB_F1	R/W	H'FFFFA1A6	16
Y/C 分离的色度信号滤波器 TAP 系数 (WB_F2) 寄存器	YCTWB_F2	R/W	H'FFFFA1A8	16
Y/C 分离的色度信号滤波器 TAP 系数 (WB_F3) 寄存器	YCTWB_F3	R/W	H'FFFFA1AA	16
Y/C 分离的色度信号滤波器 TAP 系数 (WB_F4) 寄存器	YCTWB_F4	R/W	H'FFFFA1AC	16
Y/C 分离的色度信号滤波器 TAP 系数 (WB_F5) 寄存器	YCTWB_F5	R/W	H'FFFFA1AE	16
Y/C 分离的色度信号滤波器 TAP 系数 (WB_F6) 寄存器	YCTWB_F6	R/W	H'FFFFA1B0	16
Y/C 分离的色度信号滤波器 TAP 系数 (WB_F7) 寄存器	YCTWB_F7	R/W	H'FFFFA1B2	16
Y/C 分离的色度信号滤波器 TAP 系数 (WB_F8) 寄存器	YCTWB_F8	R/W	H'FFFFA1B4	16
Y/C 分离的色度信号滤波器 TAP 系数 (NA_F0) 寄存器	YCTNA_F0	R/W	H'FFFFA1B6	16
Y/C 分离的色度信号滤波器 TAP 系数 (NA_F1) 寄存器	YCTNA_F1	R/W	H'FFFFA1B8	16
Y/C 分离的色度信号滤波器 TAP 系数 (NA_F2) 寄存器	YCTNA_F2	R/W	H'FFFFA1BA	16
Y/C 分离的色度信号滤波器 TAP 系数 (NA_F3) 寄存器	YCTNA_F3	R/W	H'FFFFA1BC	16
Y/C 分离的色度信号滤波器 TAP 系数 (NA_F4) 寄存器	YCTNA_F4	R/W	H'FFFFA1BE	16
Y/C 分离的色度信号滤波器 TAP 系数 (NA_F5) 寄存器	YCTNA_F5	R/W	H'FFFFA1C0	16
Y/C 分离的色度信号滤波器 TAP 系数 (NA_F6) 寄存器	YCTNA_F6	R/W	H'FFFFA1C2	16
Y/C 分离的色度信号滤波器 TAP 系数 (NA_F7) 寄存器	YCTNA_F7	R/W	H'FFFFA1C4	16
Y/C 分离的色度信号滤波器 TAP 系数 (NA_F8) 寄存器	YCTNA_F8	R/W	H'FFFFA1C6	16
Y/C 分离的色度信号滤波器 TAP 系数 (NB_F0) 寄存器	YCTNB_F0	R/W	H'FFFFA1C8	16
Y/C 分离的色度信号滤波器 TAP 系数 (NB_F1) 寄存器	YCTNB_F1	R/W	H'FFFFA1CA	16
Y/C 分离的色度信号滤波器 TAP 系数 (NB_F2) 寄存器	YCTNB_F2	R/W	H'FFFFA1CC	16
Y/C 分离的色度信号滤波器 TAP 系数 (NB_F3) 寄存器	YCTNB_F3	R/W	H'FFFFA1CE	16
Y/C 分离的色度信号滤波器 TAP 系数 (NB_F4) 寄存器	YCTNB_F4	R/W	H'FFFFA1D0	16
Y/C 分离的色度信号滤波器 TAP 系数 (NB_F5) 寄存器	YCTNB_F5	R/W	H'FFFFA1D2	16
Y/C 分离的色度信号滤波器 TAP 系数 (NB_F6) 寄存器	YCTNB_F6	R/W	H'FFFFA1D4	16
Y/C 分离的色度信号滤波器 TAP 系数 (NB_F7) 寄存器	YCTNB_F7	R/W	H'FFFFA1D6	16
Y/C 分离的色度信号滤波器 TAP 系数 (NB_F8) 寄存器	YCTNB_F8	R/W	H'FFFFA1D8	16
亮度 (Y) 信号增益调整控制寄存器	YGAINCR	R/W	H'FFFFA200	16
色差 (Cb) 信号增益调整控制寄存器	CBGAINCR	R/W	H'FFFFA202	16
色差 (Cr) 信号增益调整控制寄存器	CRGAINCR	R/W	H'FFFFA204	16
PGA 部寄存器更新控制	PGA_UPDATE	R/W	H'FFFFA280	16
PGA 控制寄存器	PGACR	R/W	H'FFFFA282	16
ADC 控制寄存器 2	ADCCR2	R/W	H'FFFFA284	16

30.4.1 ADC 控制寄存器 1 (ADCCR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	AGC MODE	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	AGCMODE	0	R/W	A/D 转换器的 AGC ON/OFF 控制 0: AGC ON 1: AGC OFF
7 ~ 0	—	全 0	R	保留位 读写值都为“0”。

(1) AGC 控制

通过 AGCMODE 位控制 AGC 的 ON/OFF。当 AGCMODE 位为“1”时，通过检测同步振幅和图像的峰值振幅以及控制 PGA 增益，进行 AGC 运行。

当 PGACR.PGA_GAIN_SEL 位为“1”时，能通过 PGACR.PGA_GAIN 值直接控制 PGA 的增益。此时，AGCMODE 位无效。

另外，当 AGCMODE 位为“0”时，禁止将 PGACR.PGA_GAIN_SEL 位置“0”。

30.4.2 时序生成控制寄存器 (1) (TGCR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
	—	—	—	—	—	—	—	SRCLEFT[8:0]									—	—
初始值:	0	0	0	0	0	0	0	1	0	0	1	1	1	1	0	0		
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W		

位	位名	初始值	R/W	说明
15 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8 ~ 0	SRCLEFT[8:0]	H'13C	R/W	输入图像信号捕捉区左端 以 27MHz 时钟周期为单位指定离水平同步基准的位置。

【注】 在全部寄存器的更新控制寄存器 (RUPDCR) 的 NEWSETTING 位为“1”并且垂直同步信号有效时，更新此寄存器。

(1) 时序生成 (水平开始) 控制

以 27MHz 时钟周期为单位，通过 SRCLEFT 位给要输出的图像信号设定水平允许信号的起始点离水平同步基准的位置。

30.4.3 时序生成控制寄存器 (2) (TGCR2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SRCTOP[5:0]						SRCHEIGHT[9:0]									
初始值:	0	1	0	1	0	0	0	0	1	1	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 10	SRCTOP[5:0]	H'14	R/W	输入图像信号捕捉区上端 以行为单位指定离垂直同步基准的位置。
9 ~ 0	SRCHEIGHT [9:0]	H'0E8	R/W	输入图像信号捕捉区高度 以行为单位指定垂直有效期间。

【注】 在全部寄存器的更新控制寄存器 (RUPDCR) 的NEWSETTING位为“1”并且垂直同步信号有效时,更新此寄存器。

(1) 时序 (垂直开始) 生成控制

以行为单位,通过 SRCTOP 位给要输出的图像信号设定垂直允许信号的起始点离垂直同步基准的位置。

(2) 时序生成 (垂直宽度) 控制

以行为单位,通过 SRCHEIGHT 位给要输出的图像信号指定垂直允许信号的高度。

30.4.4 时序生成控制寄存器 (3) (TGCR3)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SRCWIDTH[10:0]										
初始值:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	SRCWIDTH [10:0]	H'500	R/W	输入图像信号捕捉区宽度 以 27MHz 时钟周期为单位指定水平有效期间。

【注】 在全部寄存器的更新控制寄存器 (RUPDCR) 的NEWSETTING位为“1”并且垂直同步信号有效时,更新此寄存器。

(1) 时序生成 (水平宽度) 控制

以 27MHz 时钟周期为单位,通过 SRCWIDTH 位给要输出的图像信号设定水平允许信号的宽度。

NTSC (59.94Hz) 和 PAL/SECAM (50.00Hz) 的各设定的时序图如图 30.2、图 30.3、图 30.4、图 30.5 和图 30.6 所示。

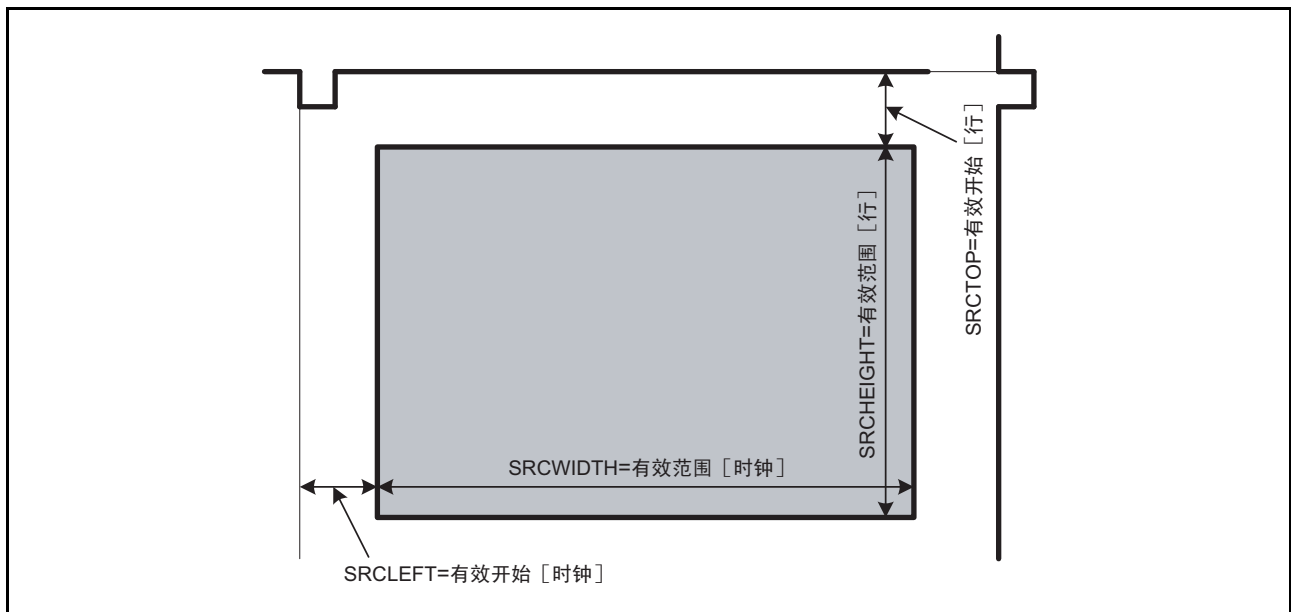


图 30.2 有效图像范围设定图

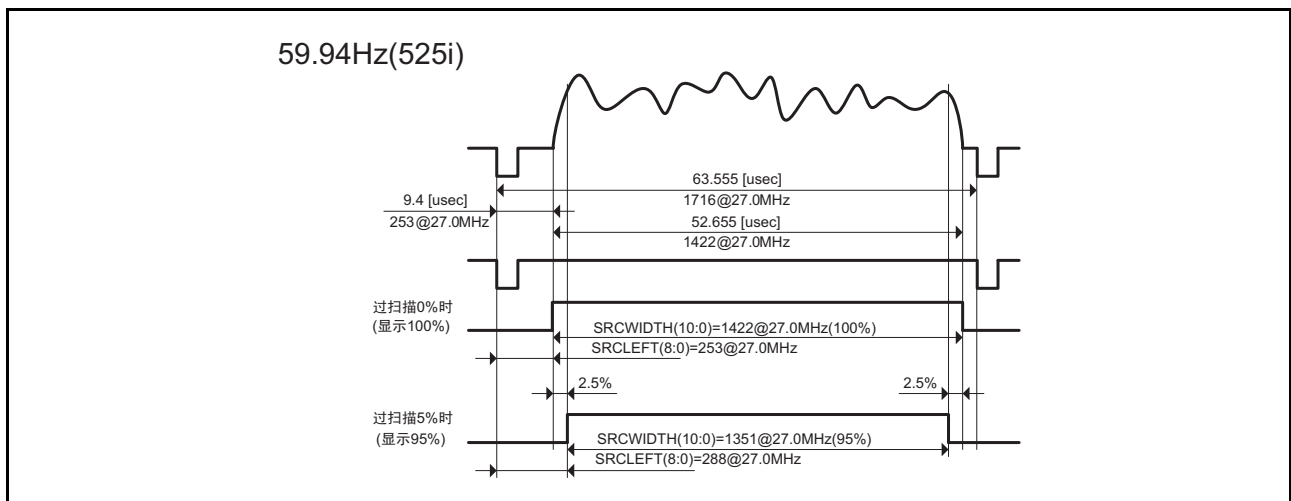


图 30.3 水平有效图像期间的设定例子 (59.94Hz (525i))

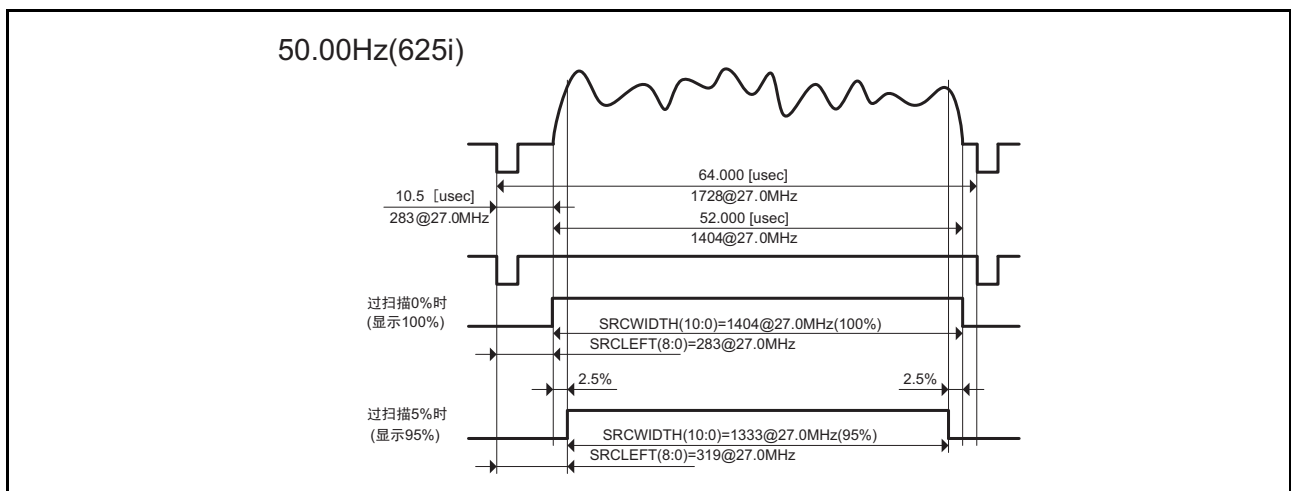


图 30.4 水平有效图像期间的设定例子 (50.00Hz (625i))

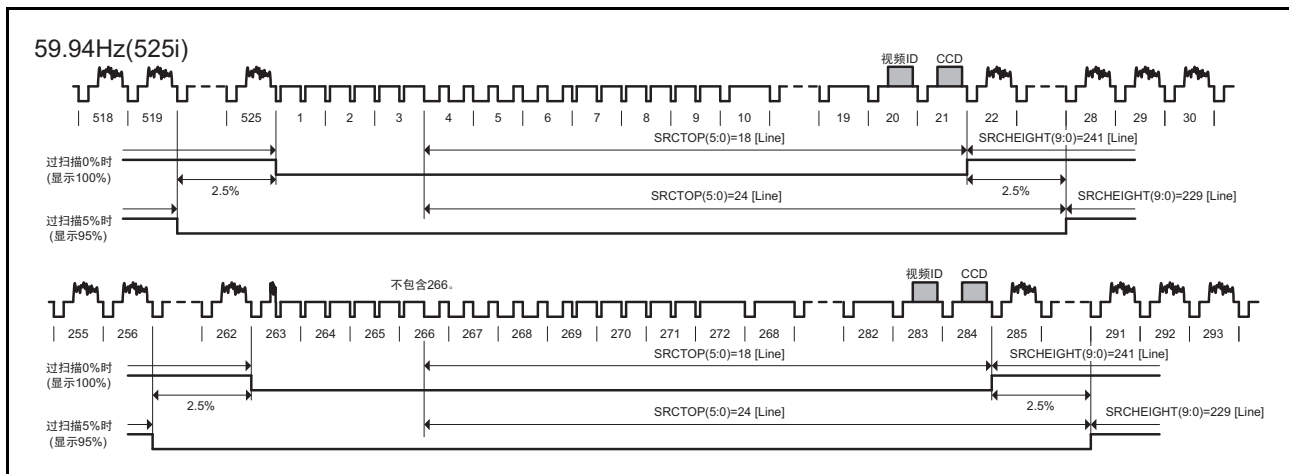


图 30.5 垂直有效图像期间的设定例子（59.94Hz（525i））

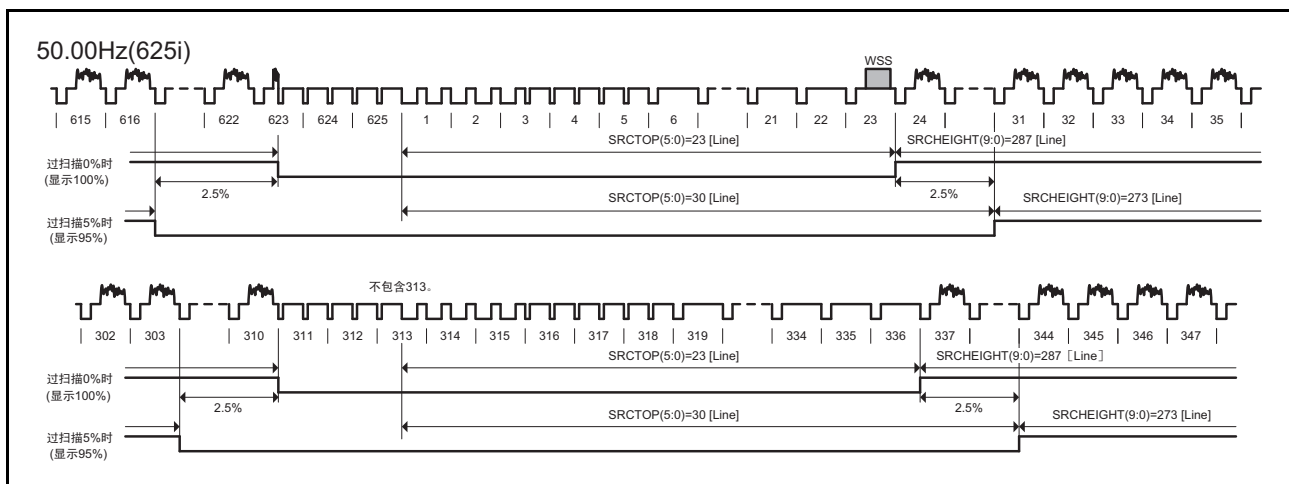


图 30.6 垂直有效图像期间的设定例子（50.00Hz（625i））

有效期间的宽度不能大于所需宽度。

只能在峰值限制器的有效期间等的此模块内部使用 TGCR1 ~ TGCR3 的设定。有关输入图像的显示尺寸，必须通过视频显示控制器 4 的缩放部的捕捉垂直尺寸寄存器（SCL0_DS2）和捕捉水平尺寸寄存器（SCL0_DS3）进行设定。

30.4.5 同步分离控制寄存器 (1) (SYNSCR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LPFVSYNC[2:0]			LPFHSYNC[2:0]			—	—	VELOCITYSHIFT_H[3:0]				SLICER MODE_H[1:0]		SLICER MODE_V[1:0]	
初始值:	0	1	1	0	1	1	0	0	0	0	0	0	1	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 13	LPFVSYNC[2:0]	011	R/W	垂直同步分离前段的低通滤波器的截止频率 0: 无 1: 0.94MHz 2: 0.67MHz 3: 0.54MHz 4: 0.47MHz 5: 0.34MHz 6: 0.27MHz 7: 0.23MHz
12 ~ 10	LPFHSYNC[2:0]	011	R/W	水平同步分离前段的低通滤波器的截止频率 0: 无 1: 2.15MHz 2: 1.88MHz 3: 1.34MHz 4: 1.07MHz 5: 0.94MHz 6: 0.67MHz 7: 0.54MHz
9、8	—	全 0	R	保留位 读写值都为“0”。
7 ~ 4	VELOCITY SHIFT_H[3:0]	0000	R/W	复合同步分离（用于水平同步信号）的基准电平运算速度调整 0: 等倍 1: 2 倍 2: 4 倍 3: 8 倍 4: 16 倍 5: 32 倍 6: 64 倍 7: 128 倍 其他: 256 倍 速度标准（等倍）↔ 高速（256 倍）
3、2	SLICER MODE_H[1:0]	10	R/W	复合同步分离电路（用于水平同步信号）的自动限幅电平设定 0: 通过 CSYNCSLICE_H 位进行手动设定 1: 同步深度的 25%（自动） 2: 同步深度的 50%（自动） 3: 同步深度的 75%（自动）

位	位名	初始值	R/W	说明
1、0	SLICER MODE_V[1:0]	10	R/W	复合同步分离电路（用于垂直同步信号）的自动限幅电平设定 0: 通过 CSYNCSLICE_V 位进行手动设定 1: 同步深度的 25%（自动） 2: 同步深度的 50%（自动） 3: 同步深度的 75%（自动）

(1) 垂直同步分离前段的低通滤波器控制

为了防止噪声引起的同步分离误动作，通过 LPFVSYNC 位给输入到同步分离电路的图像信号设定噪声消除低通滤波器。将低通滤波器的截止频率设定在复合同步信号成分不恶化（能检测）的范围内。

表 30.6 垂直同步分离前段的低通滤波器的截止频率

用于垂直同步分离		
LPFVSYNC[2:0]	t	fc[MHz]
1	0.109375	0.939647766
2	0.078125	0.671176976
3	0.0625	0.536941581
4	0.0546875	0.469823883
5	0.0390625	0.335588488
6	0.03125	0.26847079
7	0.0273438	0.234911942

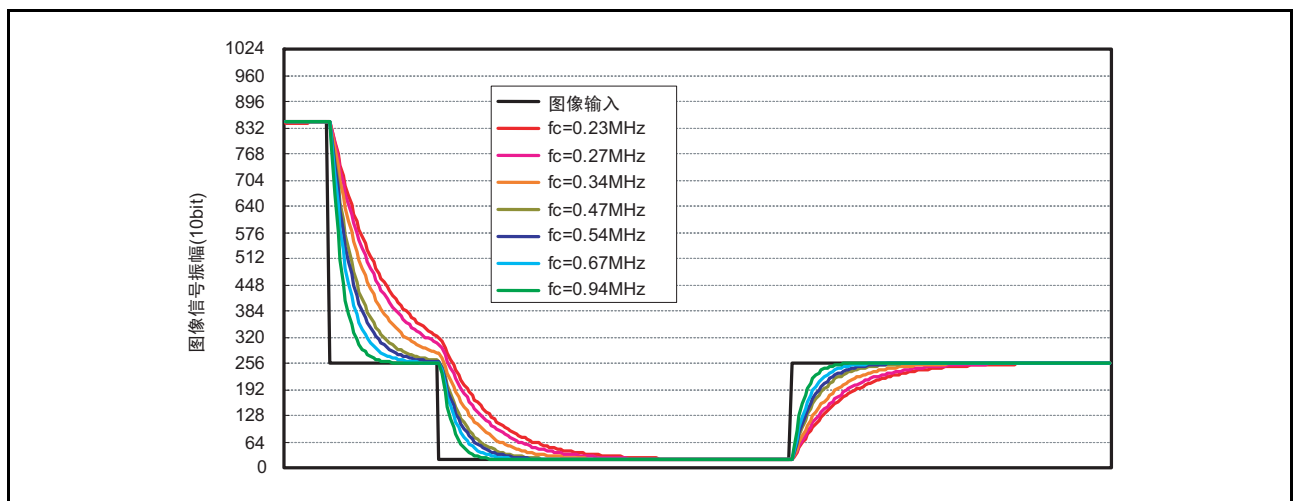


图 30.7 输入全白信号时的水平同步信号附近的低通滤波器输出波形（垂直：模式图）

(2) 水平同步分离前段的低通滤波器控制

为了防止噪声引起的同步分离误动作，通过 LPFHSYNC 位给输入到同步分离电路的图像信号设定噪声消除低通滤波器。将低通滤波器的截止频率设定在复合同步信号成分不恶化（能检测）的范围内。

表 30.7 水平同步分离前段的低通滤波器的截止频率

用于水平同步分离		
LPFHSYNC[2:0]	t	fc[MHz]
1	0.25	2.147766323
2	0.21875	1.879295533
3	0.15625	1.342353952
4	0.125	1.073883162
5	0.10938	0.939647766
6	0.07813	0.671176976
7	0.0625	0.536941581

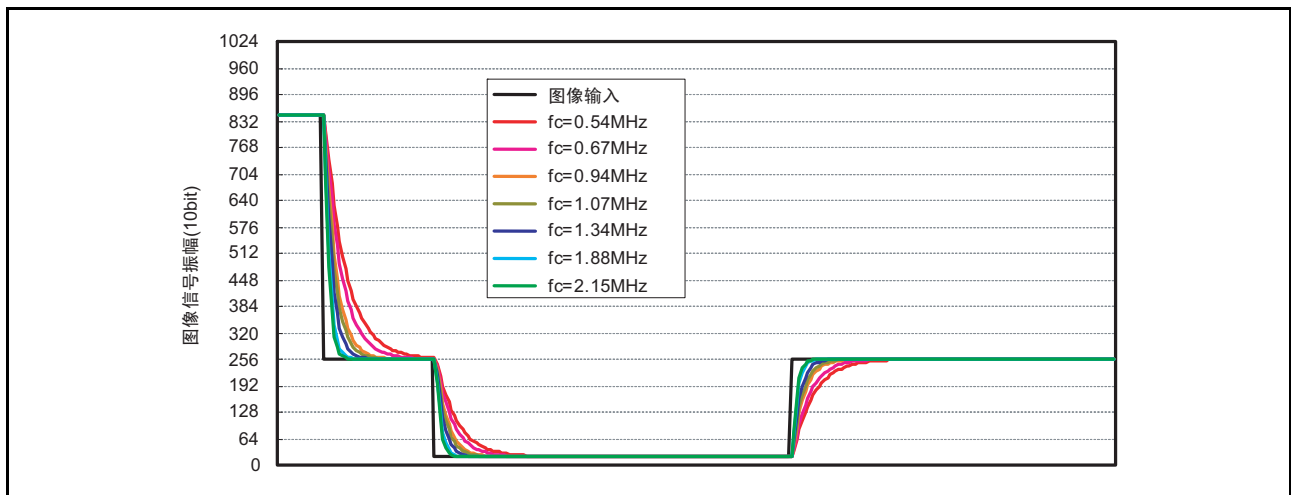


图 30.8 输入全白信号时的水平同步信号附近的低通滤波器输出波形（水平：模式图）

(3) 同步分离的基准电平运算速度控制

通过 VELOCITYSHIFT_H 位控制自动判定限幅电平时的电平判定速度。

当因同步下垂等而发生同步倾斜时，能通过此寄存器提高判定速度，改善同步倾斜。标准设定值为“0”。

(4) 水平同步限幅器控制

通过 SLICERMODE_H 位控制复合同步信号从图像信号的分离。

复合同步分离的限幅电平能选择自动设定和手动设定。在自动设定时，根据后述的同步信号振幅检测结果自动设定限幅电平。

同步限幅器能独立控制水平同步和垂直同步。

(5) 垂直同步限幅器控制

通过 SLICERMODE_V 位控制复合同步信号从图像信号的分离。

复合同步分离的限幅电平能选择自动设定和手动设定。在自动设定时，根据后述的同步信号振幅检测结果自动设定限幅电平。

同步限幅器能独立控制水平同步和垂直同步。

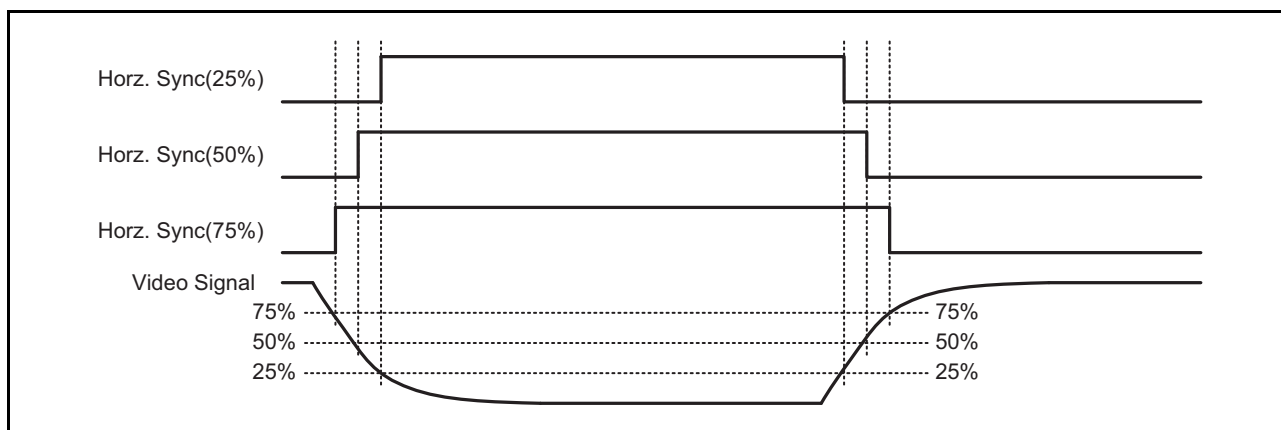


图 30.9 限幅电平的自动设定

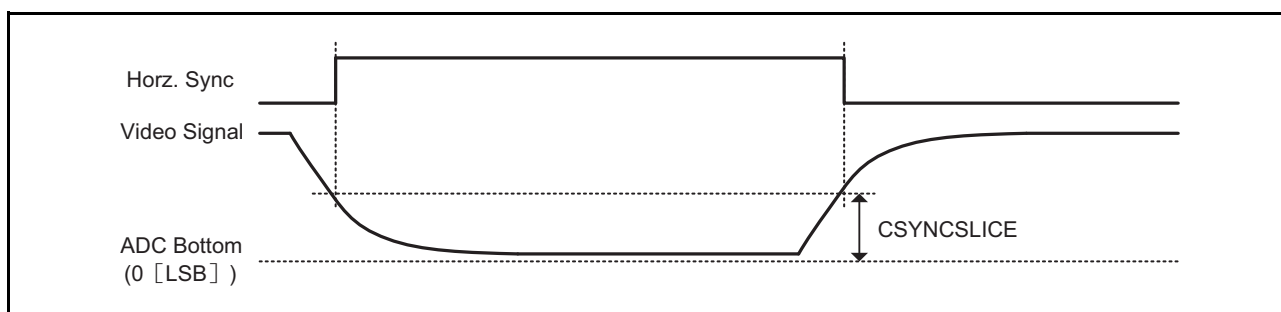


图 30.10 限幅电平的手动设定

30.4.6 同步分离控制寄存器 (2) (SYNSCR2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SYNCMAXDUTY_H[5:0]					SYNCMINDUTY_H[5:0]						
初始值:	0	0	0	0	0	0	1	1	1	1	0	0	1	0	1	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 12	—	全 0	R	保留位 读写值都为“0”。
11 ~ 6	SYNCMAX DUTY_H[5:0]	001111	R/W	水平周期和水平同步信号脉宽的最大比率 在自动设定限幅电平 (SLICERMODE_H≠0) 时有效。
5 ~ 0	SYNCMIN DUTY_H[5:0]	001010	R/W	水平周期和水平同步信号脉宽的最小比率 在自动设定限幅电平 (SLICERMODE_H≠0) 时有效。

(1) 水平同步分离的同步振幅检测控制

通过 SYNCMAXDUTY_H 位和 SYNCMINDUTY_H 位控制图像信号中包含的复合同步信号的同步信号振幅检测。

表 30.8 复合同步分离的自动限幅电平寄存器功能表

	水平 期间 [usec]	水平 同步宽度 [usec]	图像 有效期间 [usec]	水平 回扫期间 [usec]	SYNCMAXDUTY_H [5:0]	SYNCMINDUTY_H [5:0]
					推荐值	推荐值
525i/59.94Hz	63.56	4.70	52.66	10.90	15	10
625i/50Hz	64.00	4.70	52.00	12.00	15	10

30.4.7 同步分离控制寄存器 (3) (SYNSCR3)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	SSCLIPSEL[3:0]				CSYNCSLICE_H[9:0]									
初始值:	0	0	1	1	1	1	0	0	1	0	0	1	0	0	1	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13 ~ 10	SSCLIPSEL[3:0]	1111	R/W	限幅值的设定 限制要输入到低通滤波器的图像信号 (位值 = 限幅值: 振幅的 1/2 ~ 无限幅)。 0: 512 1: 546 2: 580 3: 614 4: 648 5: 682 6: 716 7: 750 8: 785 9: 819 10: 853 11: 887 12: 921 13: 955 14: 989 15: 1023
9 ~ 0	CSYNCSLICE_H [9:0]	0010010 010	R/W	复合同步信号分离 (用于水平同步信号) 的限幅电平 在手动设定限幅电平 (SLICERMODE_H=0) 时有效。 设定范围: 0 ~ 1023

(1) 同步分离的图像信号限幅控制

对于输入到同步分离电路的图像信号，为了防止图像信号的振幅依赖性，通过 SSCLIPSEL 位对图像高频成分的限幅电平进行控制。

将图像限幅电平设定在复合同步信号成分不恶化 (能检测) 的范围内。

(2) 水平同步分离的限幅电平控制

通过 CSYNCSLICE_H 位控制同步分离的限幅电平。此位只在 SLICERMODE_H 为“0”时有效。

30.4.8 同步分离控制寄存器 (4) (SYNSCR4)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SYNCMAXDUTY_V[5:0]					SYNCMINDUTY_V[5:0]						
初始值:	0	0	0	0	0	0	1	1	1	1	0	0	1	0	1	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 12	—	全 0	R	保留位 读写值都为“0”。
11 ~ 6	SYNCMAX DUTY_V[5:0]	001111	R/W	水平周期和水平同步信号脉宽的最大比率 在自动设定限幅电平 (SLICERMODE_V≠0) 时有效。
5 ~ 0	SYNCMIN DUTY_V[5:0]	001010	R/W	水平周期和水平同步信号脉宽的最小比率 在自动设定限幅电平 (SLICERMODE_V≠0) 时有效。

(1) 垂直同步分离的同步振幅检测控制

通过 SYNCMAXDUTY_V 位和 SYNCMINDUTY_V 位控制图像信号中包含的复合同步信号的同步信号振幅检测。

表 30.9 复合同步分离的自动限幅电平寄存器功能表

	水平 期间 [usec]	水平 同步宽度 [usec]	图像 有效期间 [usec]	水平 回扫期间 [usec]	SYNCMAXDUTY_V [5:0]	SYNCMINDUTY_V [5:0]
					推荐值	推荐值
525i/59.94Hz	63.56	4.70	52.66	10.90	15	9
625i/50Hz	64.00	4.70	52.00	12.00	15	9

30.4.9 同步分离控制寄存器 (5) (SYNSCR5)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VSYNC DELAY	VSYNC SLICE[4:0]					CSYNC SLICE_V[9:0]									
初始值:	0	0	1	0	1	1	0	0	1	0	0	1	0	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	VSYNCDELAY	0	R/W	使分离的垂直同步信号只延迟 1/4 水平周期。 1: 延迟 1/4fH 0: 未延迟 1/4fH 【注】* 如果在场判别结果不稳定时更改此值, 就可能改善场判别结果的稳定性。
14 ~ 10	VSYNC SLICE [4:0]	01011	R/W	垂直同步分离的阈值 值越大需要的脉冲越宽。
9 ~ 0	CSYNC SLICE_V[9:0]	0010010 010	R/W	复合同步信号分离 (用于垂直同步信号) 的限幅电平 在手动设定限幅电平 (SLICERMODE_V=0) 时有效。 设定范围: 0 ~ 1023

(1) 垂直同步分离控制

通过 VSYNCDELAY 位控制垂直同步信号和水平同步信号的相位。

如果将 VSYNCDELAY 置“1”, 就让垂直同步信号延迟 1/4fH, 从而可能改善场判别结果的稳定性。

(2) 垂直同步分离控制

通过 VSYNC SLICE 位控制从复合同步信号分离垂直同步信号的阈值。

根据各图像信号格式的锯齿脉冲信号宽度设定阈值。推荐的设定值如表 30.10 所示。

表 30.10 锯齿脉宽对应的阈值一览表 (参考)

	锯齿脉冲期间 [usec]	VSYNC SLICE[4:0]
525i/59.94Hz	27.08	15
625i/50Hz	27.30	15

(3) 垂直同步分离的限幅电平控制

通过 CSYNC SLICE_V 位控制同步分离的限幅电平。此位只在 SLICERMODE_V 为“0”时有效。

30.4.10 水平 AFC 控制寄存器 (1) (HAFCCR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HAFCGAIN[3:0]				—	HAFCFREERUN	HAFCTYP[9:0]									
初始值:	0	1	1	0	0	0	1	0	1	0	1	1	0	1	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 12	HAFCGAIN[3:0]	0110	R/W	水平 AFC 的环路增益设定 0 ~ 5: 值越小响应速度越慢 6: 标准值 7 ~ 15: 值越大响应速度越快
11	—	0	R	保留位 读写值都为“0”。
10	HAFCFREERUN	0	R/W	水平 AFC 的自激振荡模式 ON/OFF 0: OFF 1: ON
9 ~ 0	HAFCTYP[9:0]	1010110 100	R/W	水平 AFC 的中心振荡周期 以 27MHz 时钟周期为单位指定从 1024 个时钟开始的偏移量。

(1) 水平 AFC 的环路增益控制

通过 HAFCGAIN 位控制水平 AFC 的环路增益（响应速度）。

虽然值越大响应速度越快，但是对噪声的误判断率会变高。

此位的标准设定值为“6”。

(2) 水平 AFC 的自激振荡控制

通过 HAFCFREERUN 位控制水平 AFC 的自激振荡。

如果将 HAFCFREERUN 位置“1”，水平 AFC 就不依赖输入而进行自激振荡。通常必须将 HAFCFREERUN 位置“0”。

(3) 水平 AFC 的同步范围（水平）控制

通过 HAFCMIN 位、HAFCTYP 位和 HAFCMAX 位控制水平 AFC 的中心频率和同步范围。

控制水平 AFC 功能，稳定弱电场和 VTR 特殊播放等恶化信号的水平同步信号。

HAFCMIN[9:0] : 水平 AFC 的最小振荡周期

HAFCTYP[9:0] : 水平 AFC 的中心振荡周期

HAFCMAX[9:0] : 水平 AFC 的最大振荡周期

水平 AFC 的同步范围用以下计算式表示：

$$\text{HAFCMIN} < \text{HAFCTYP} < \text{HAFCMAX} \quad \dots\dots \textcircled{1}$$

因此，

HAFCMIN = HAFCTYP - 容许偏差

HAFCTYP = N × M - 1024

HAFCMAX = HAFCTYP + 容许偏差

M: 每个水平期间的时钟个数（27MHz 采样）

N: 倍速设定

2（倍速）：M < 1024 时，1（正常）：M ≥ 1024 时

如果满足计算式①，就锁定水平 AFC。

如果锁定水平 AFC，VSYNC SR 寄存器的 FHLOCK 位就为“1”，否则 FHLOCK 位就为“0”。

表 30.11 水平 AFC 同步范围设定表

信号 FORMAT	fH 水平 周期	M fH @27.0MHz	N 倍速 设定	HAFCMAX [9:0]	HAFCTYP [9:0]	HAFCMIN [9:0]	偏差		单位
525i	63.56 [usec]	1716 [clk]	1	771	692	618	79	-74	[clk]
				15.034	15.734	16.434	-0.700	0.700	[kHz]
625i	64.00 [usec]	1728 [clk]	1	785	704	629	81	-75	[clk]
				14.925	15.625	16.325	-0.700	0.700	[kHz]

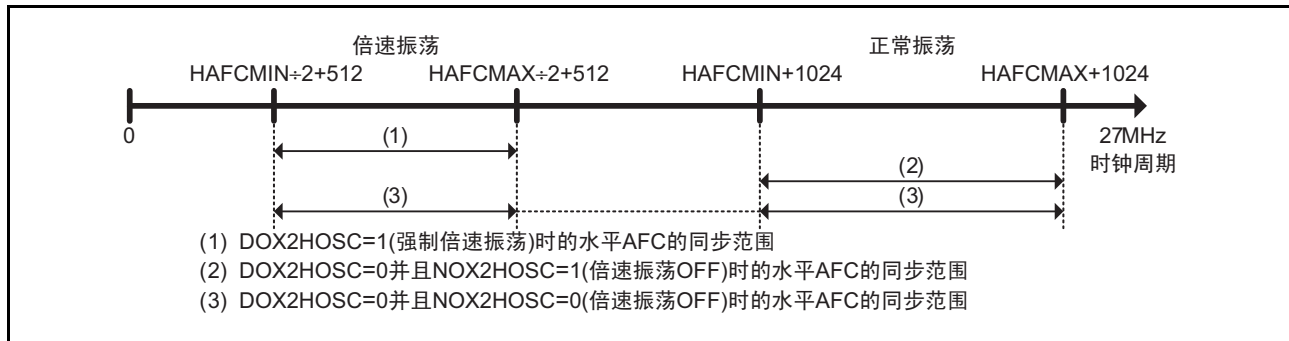


图 30.11 水平 AFC 的同步范围 (水平)

30.4.11 水平 AFC 控制寄存器 (2) (HAFCCR2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HAFSTART[3:0]				NOX2H OSC	DOX2H OSC	HAFCMAX[9:0]									
初始值:	0	0	0	0	0	0	1	0	1	1	1	0	0	1	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 12	HAFSTART[3:0]	0000	R/W	水平 AFC 的正常运行开始行 (=VBI 处理结束行) 从垂直同步信号之后的第 N 行开始相位比较。
11	NOX2HOSC	0	R/W	水平 AFC 的倍速判定禁止设定 0: 自动控制 1: 禁止倍速振荡
10	DOX2HOSC	0	R/W	水平 AFC 的强制倍速振荡设定 0: 自动控制 1: 强制倍速振荡
9 ~ 0	HAFCMAX[9:0]	101110 0110	R/W	水平 AFC 的最大振荡周期 以 27MHz 时钟周期为单位指定从 1024 个时钟开始的偏移量。

(1) 水平 AFC 的同步范围 (垂直) 控制

能通过 HAFSTART 位和 HAFEND 位指定水平 AFC 的运行范围。为了避免 VTR 磁头切换部的误动作, 通常从垂直同步信号的前几行开始到垂直同步信号为止停止水平 AFC 运行。

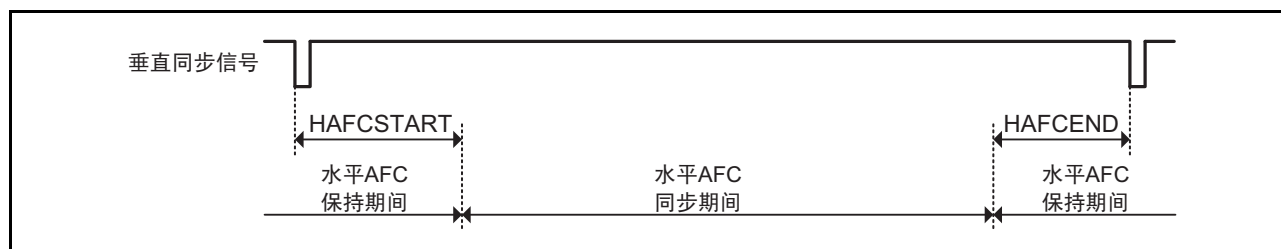


图 30.12 水平 AFC 的同步范围 (垂直)

(2) 水平 AFC 的倍速控制

通过 NOX2HOSC 位和 DOX2HOSC 位控制水平 AFC 的倍速判定。在 NTSC、PAL 和 SECAM 中, 必须将 DOX2HOSC 位置 “0”。

30.4.12 水平 AFC 控制寄存器 (3) (HAFCCR3)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	HAFCEMEND[3:0]				HAFCMODE[1:0]		HAFCMIN[9:0]										
初始值:	1	0	0	0	1	0	1	0	1	0	0	0	0	0	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 12	HAFCEMEND[3:0]	1000	R/W	水平 AFC 的正常运行结束行 (=VBI 处理开始行) 从垂直同步信号的前 N 行开始停止相位比较。
11、10	HAFCMODE[1:0]	10	R/W	水平 AFC 的 VBI 期间运行模式设定 [1] 低 S/N 时的环路增益控制设定 0: 固定环路增益 1: 自动控制环路增益 [0] VBI 期间的水平 AFC 控制设定 0: 停止 VBI 期间的相位比较 1: 降低 VBI 期间的环路增益
9 ~ 0	HAFCMIN[9:0]	101000 0010	R/W	水平 AFC 的最小振荡周期 以 27MHz 时钟周期为单位指定从 1024 个时钟开始的偏移量。

(1) 水平 AFC 的 VBI 期间运行控制

通过将 HAFCMODE[1] 位置 “1”，降低低 S/N (VSYNC SR.ISNOISY=1) 时的环路增益，防止噪声引起的误动作。

此位的推荐设定值为 “1”。

通过 HAFCMODE[0] 位控制 VBI 期间的水平 AFC 运行。

此位的推荐设定值为 “0”。

30.4.13 垂直递减计数控制寄存器 (1) (VCDWCR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VCDFREERUN	NOVCD50	NOVCD60	VCDDEFAULT [1:0]	VCDWINDOW[5:0]					VCDOFFSET[4:0]						
初始值:	0	0	0	0	0	0	1	0	1	0	0	0	1	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	VCDFREERUN	0	R/W	垂直递减计数的自激振荡模式 ON/OFF 0: OFF 1: ON
14	NOVCD50	0	R/W	垂直递减计数的 50Hz 振荡模式 OFF 0: 50Hz 振荡 ON 1: 50Hz 振荡 OFF
13	NOVCD60	0	R/W	垂直递减计数的 60Hz (59.94Hz) 振荡模式 OFF 0: 60Hz 振荡 ON 1: 60Hz 振荡 OFF
12、11	VCDDEFAULT [1:0]	00	R/W	垂直递减计数的中心振荡周期 0: 自动判定 1: 50.00Hz 2: 59.94Hz 3: 60.00Hz
10 ~ 5	VCDWINDOW [5:0]	010100	R/W	垂直递减计数的同步范围 (以 0.1ms 为单位进行设定)
4 ~ 0	VCDOFFSET [4:0]	01010	R/W	垂直递减计数的最小振荡周期 以 0.1ms 为单位设定从中心周期开始的偏移。

(1) 垂直递减计数的自激振荡控制

通过 VCDFREERUN 位控制垂直递减计数的自激振荡。

通过将 VCDFREERUN 位置 “1”，使垂直递减计数不依赖输入而进行自激振荡。通常必须将此位置 “0”。

(2) 垂直递减计数的 50Hz 振荡控制

通过 NOVCD50 位控制 50Hz 的振荡。

如果将 NOVCD50 位置 “1”，垂直递减计数就不锁定为 50Hz。

(3) 垂直递减计数的 60Hz 振荡控制

通过 NOVCD60 位控制 60Hz 的振荡。

如果将 NOVCD60 位置 “1”，垂直递减计数就不锁定为 60Hz。

(4) 垂直递减计数的中心频率控制

通过 VCDDEFAULT 位控制垂直递减计数的中心频率。

表 30.12 垂直递减计数运行模式表

VCDDEFAULT[1:0]	运行模式
0	自动判定
1	50.00Hz
2	59.94Hz
3	60.00Hz

(5) 垂直递减计数的同步范围控制

通过 VCDWINDOW 位和 VCDOFFSET 位控制垂直递减计数的同步范围。

各位的设定如图 30.13 所示。

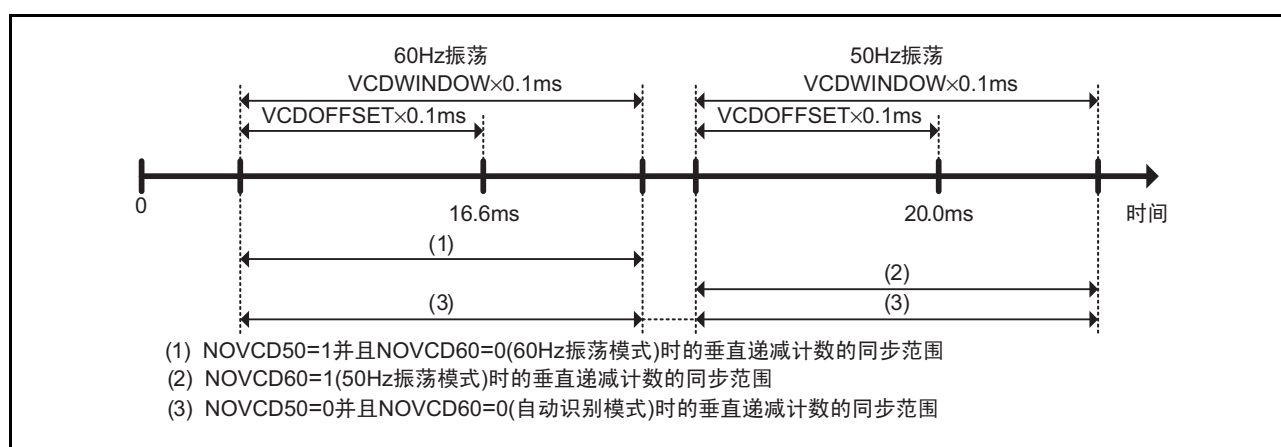


图 30.13 垂直递减计数的同步范围

30.4.14 数字钳位控制寄存器 (1) (DCPCR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCPMODE_Y	—	—	—	DCPCHECK	—	BLANKLEVEL_Y[9:0]									
初始值:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	DCPMODE_Y	1	R/W	钳位电平设定模式 (Y 信号) 0: 手动设定钳位电平 1: 自动设定钳位电平
14 ~ 12	—	全 0	R	保留位 读写值都为 “0”。
11	DCPCHECK	0	R/W	数字钳位脉冲的位置确认 只将钳位偏移电平设定值的偏移值加到钳位位置。
10	—	0	R	保留位 读写值都为 “0”。
9 ~ 0	BLANKLEVEL_Y [9:0]	000000 0000	R/W	钳位偏移电平 (Y 信号) (设定被减值) 以 1 个 LSB 为单位, 用 2 的补码进行设定。

(1) Y 信号的钳位运行控制

通过 DCPMODE_Y 位控制 Y 信号的钳位电平。

当 DCPMODE_Y 位为 “0” 时, 图像信号减去 BLANKLEVEL_Y 位的设定值。

$$Y \text{ 信号输出} = Y \text{ 信号输入} - \text{BLANKLEVEL_Y}$$

当 DCPMODE_Y 位为 “1” 时, 图像信号减去 (数字钳位脉冲位置的图像信号电平 (区分电平) 加上 BLANKLEVEL_Y 位的设定值)。

$$Y \text{ 信号输出} = Y \text{ 信号输入} - (\text{检测值} + \text{BLANKLEVEL_Y})$$

(2) 数字钳位脉冲位置确认控制

通过 DCPCHECK 位, 在画面上控制数字钳位脉冲的位置确认。

按以下步骤进行位置确认:

数字钳位位置确认位: 将 DCPCHECK 位置 “1”。

输入图像信号捕捉区左端设定位: 将 SRCLEFT 位置 “0”, 并且将视频显示控制器 4 的缩放部的捕捉水平寄存器 (SCL0_DS3).RES_HS[10:0] 位置 “0”。

将监视信号的钳位偏移电平设定为最小值 (BLANKLEVEL_Y 位为 -512, BLANKLEVEL_Cb 位和 BLANKLEVEL_Cr 位为 -32)。

通过 DCPPOS_Y 位 (或者 DCPPOS_C 位) 和 DCPWIDTH 位调整脉冲的位置和宽度。

30.4.15 数字钳位控制寄存器 (2) (DCPCR2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCPMODE_C	—	—	—	BLANKLEVEL_CB[5:0]					BLANKLEVEL_CR[5:0]						
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	DCPMODE_C	0	R/W	钳位电平设定模式 (Cb/Cr 信号) 0: 手动设定钳位电平 1: 自动设定钳位电平
14 ~ 12	—	全 0	R	保留位 读写值都为“0”。
11 ~ 6	BLANK LEVEL_CB[5:0]	000000	R/W	钳位偏移电平 (Cb 信号) (设定被减值) 以 1 个 LSB 为单位, 用 2 的补码进行设定。
5 ~ 0	BLANK LEVEL_CR[5:0]	000000	R/W	钳位偏移电平 (Cr 信号) (设定被减值) 以 1 个 LSB 为单位, 用 2 个补码进行设定。

(1) Cb/Cr 信号的钳位运行控制

通过 DCPMODE_C 位控制 Cb/Cr 信号的钳位电平。

当 DCPMODE_C 位为“0”时, 图像信号减去 BLANKLEVEL_CB 位 /BLANKLEVEL_CR 位的设定值。

$$\text{Cb 信号输出} = \text{Cb 信号输入} - \text{BLANKLEVEL_CB}$$

$$\text{Cr 信号输出} = \text{Cr 信号输入} - \text{BLANKLEVEL_CR}$$

当 DCPMODE_C 位为“1”时, 图像信号减去 (数字钳位脉冲位置的图像信号电平 (区分电平) 加上 BLANKLEVEL_CB 位 /BLANKLEVEL_CR 位的设定值)。

$$\text{Cb 信号输出} = \text{Cb 信号输入} - (\text{检测值} + \text{BLANKLEVEL_CB})$$

$$\text{Cr 信号输出} = \text{Cr 信号输入} - (\text{检测值} + \text{BLANKLEVEL_CR})$$

30.4.16 数字钳位控制寄存器 (3) (DCPCR3)

	位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		—	DCPRESPONSE[2:0]			—	—	—	—	—	—	—	—	—	—	—	—
初始值:		0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:		R	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	DCPRESPONSE [2:0]	101	R/W	数字钳位的响应速度 值越大响应速度越快, 但是容易受噪声的影响。
11 ~ 0	—	全 0	R	保留位 读写值都为“0”。

(1) 数字钳位的响应速度控制

通过 DCPRESPONSE 位控制数字钳位的响应速度。

DCPRESPONSE 位的值越大响应速度越快, 但是容易受噪声的影响。

DCPRESPONSE 位在 Y、Cb 和 Cr 中共用。

30.4.17 数字钳位控制寄存器 (4) (DCPCR4)

	位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		DCPSTART[5:0]						—	—	—	—	—	—	—	—	—	—
初始值:		0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:		R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 10	DCPSTART[5:0]	010000	R/W	数字钳位开始行 (以 1 行为单位) 从垂直同步信号之后的第 N 行开始钳位。
9 ~ 0	—	全 0	R	保留位 读写值都为“0”。

30.4.18 数字钳位控制寄存器 (5) (DCPCR5)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCPEND[5:0]						—	—	—	—	—	—	—	—	—	—
初始值:	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 10	DCPEND[5:0]	010000	R/W	数字钳位结束行 (以 1 行为单位) 从垂直同步信号的前 N 行开始停止钳位。
9 ~ 0	—	全 0	R	保留位 读写值都为“0”。

(1) 数字钳位控制 (垂直)

通过 DCPSTART 位和 DCPEND 位控制数字钳位垂直方向的运行。

各位的设定如图 30.14 所示。

DCPSTART 位和 DCPEND 位在 Y、Cb 和 Cr 中共用。

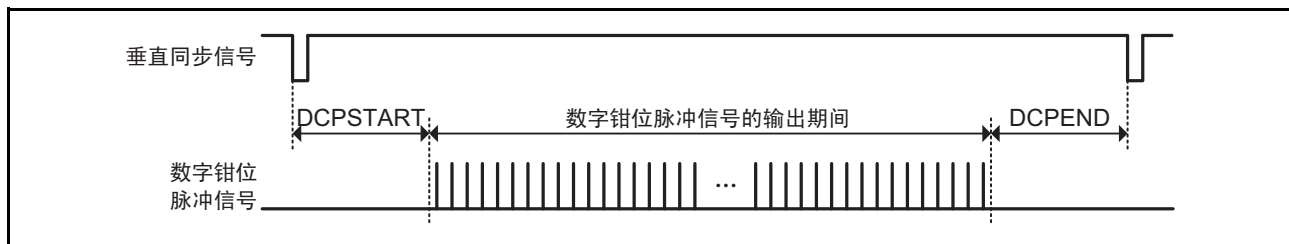


图 30.14 数字钳位时序 (垂直) 图

30.4.19 数字钳位控制寄存器 (6) (DCPCR6)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	DCPWIDTH[6:0]						—	—	—	—	—	—	—	—	—
初始值:	0	0	1	1	0	1	1	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 8	DCPWIDTH[6:0]	0110110	R/W	数字钳位脉冲的宽度 设定范围: 0 ~ 127, 以 27MHz 时钟周期为单位。
7 ~ 0	—	全 0	R	保留位 读写值都为“0”。

30.4.20 数字钳位控制寄存器 (7) (DCPCR7)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCPPOS_Y[7:0]								—	—	—	—	—	—	—	—
初始值:	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 8	DCPPOS_Y[7:0]	10100010	R/W	数字钳位脉冲的水平开始位置 (Y 信号) 设定范围: 0 ~ 255, 以 27MHz 时钟周期为单位。
7 ~ 0	—	全 0	R	保留位 读写值都为“0”。

30.4.21 数字钳位控制寄存器 (8) (DCPCR8)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCPPOS_C[7:0]								—	—	—	—	—	—	—	—
初始值:	0	0	0	1	1	0	1	1	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 8	DCPPOS_C[7:0]	00011011	R/W	数字钳位脉冲的水平开始位置 (Cb/Cr 信号) 设定范围: 0 ~ 255, 以 27MHz 时钟周期为单位。
7 ~ 0	—	全 0	R	保留位 读写值都为“0”。

(1) 数字钳位控制 (水平)

通过 DCPWIDTH 位、DCPPOS_Y 位和 DCPPOS_C 位控制数字钳位的水平方向。

各位的设定如图 30.15 所示。

DCPPOS_Y 位对应 Y 信号, DCPPOS_C 位对应 Cb、Cr 信号。DCPWIDTH 位在 Y、Cb 和 Cr 中共用。

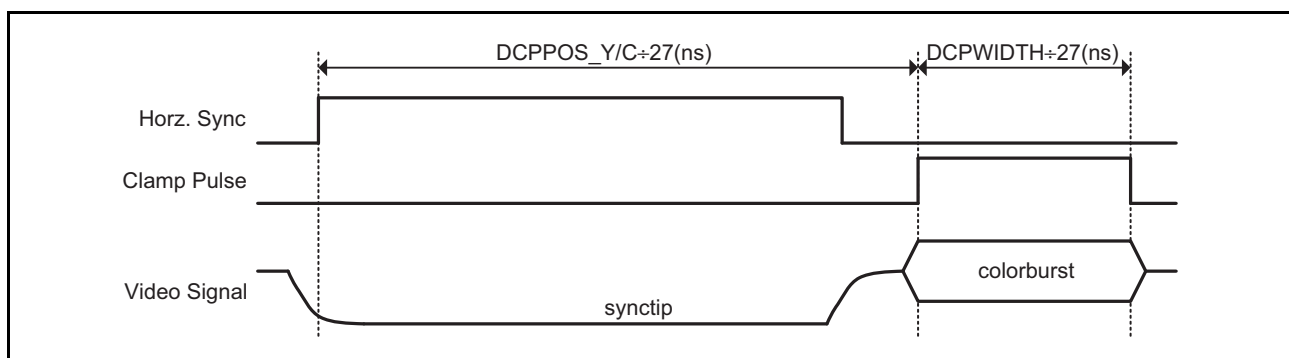


图 30.15 数字钳位时序 (水平) 图

30.4.22 噪声检测控制寄存器 (NSDCR)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	ACFINPUT [1:0]		—	—	—	ACFLAGTIME [4:0]				—	—	ACFFILTER [1:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13、12	ACFINPUT[1:0]	00	R/W	求自相关函数的图像信号 0: Y 信号 1: Cb 信号 2、3: Cr 信号
11 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8 ~ 4	ACFLAGTIME [4:0]	00000	R/W	计算自相关函数时的延迟时间 0 ~ 31 个时钟 @27MHz 时钟 如果将延迟时间设定为“0”，NSDSR.ACFSTRENGTH 位的值就几乎相当于噪声功率。
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	ACFFILTER[1:0]	00	R/W	自相关函数值的平滑参数 ACFFILTER 位值越小平均时间越长。 在 1 个场量 ~ 几秒之间变化。

(1) 噪声检测的输入控制

通过 ACFINPUT 位控制噪声检测的输入。

表 30.13 噪声检测的输入选择表

ACFINPUT	输入信号
0	Y 信号
1	Cb 信号
2、3	Cr 信号

(2) 噪声检测的自相关函数控制

通过 ACFLAGTIME 位控制噪声检测的自相关函数。

如果将延迟时间设定为“0”，NSDSR.ACFSTRENGTH 位的值就几乎相当于噪声功率。

(3) 噪声检测的平滑滤波器控制

通过 ACFFILTER 位控制噪声检测的输入平滑。

ACFFILTER 的值越小场累积量越多（检测噪声时需要时间）。

30.4.23 色同步锁定控制 / 色度解码控制寄存器 (BTLCR)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LOCKRANGE [1:0]		LOOPGAIN[1:0]		LOCKLIMIT[1:0]		BCOFR EERUN	—	DEFAULTSYS [1:0]		NONTS C358	NONTS C443	NOPAL M	NOPAL N	NOPAL 443	NOSEC AM
初始值:	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15、14	LOCKRANGE [1:0]	01	R/W	色同步锁定 PLL 的同步范围 0: ±400Hz 1: ±800Hz 2: ±1200Hz 3: ±1600Hz
13、12	LOOPGAIN[1:0]	01	R/W	色同步锁定 PLL 的环路增益 值越大响应速度越快，但是容易受噪声的影响。
11、10	LOCKLIMIT[1:0]	10	R/W	色同步锁定 PLL 重新搜索自激频率时的电平值 电平值越大越容易解除锁定，也越容易进行重新搜索。
9	BCOFR EERUN	0	R/W	色同步锁定 PLL 的自激振荡模式 ON/OFF 0: OFF 1: ON
8	—	0	R	保留位 读写值都为“0”。
7、6	DEFAULTSYS [1:0]	00	R/W	默认颜色体系 0: NTSC 1: PAL 2: SECAM 3: 未指定
5	NONTSC358	0	R/W	不判别 NTSC-M。 0: NTSC-M 判别 ON 1: NTSC-M 判别 OFF
4	NONTSC443	0	R/W	不判别 NTSC-4.43。 0: NTSC-4.43 判别 ON 1: NTSC-4.43 判别 OFF
3	NOPALM	0	R/W	不判别 PAL-M。 0: PAL-M 判别 ON 1: PAL-M 判别 OFF
2	NOPALN	0	R/W	不判别 PAL-N。 0: PAL-N 判别 ON 1: PAL-N 判别 OFF
1	NOPAL443	0	R/W	不判别 PAL-B、G、H、I、D。 0: PAL-B、G、H、I、D 判别 ON 1: PAL-B、G、H、I、D 判别 OFF
0	NOSECAM	0	R/W	不判别 SECAM。 0: SECAM 判别 ON 1: SECAM 判别 OFF

(1) 色同步锁定 PLL 的同步范围控制

通过 LOCKRANGE 位控制色同步锁定 PLL 的同步范围。

表 30.14 色同步锁定 PLL 的同步范围

LOCKRANGE	色同步锁定 PLL 的同步范围
0	±400Hz
1	±800Hz
2	±1200Hz
3	±1600Hz

(2) 色同步锁定 PLL 的环路增益控制

通过 LOOPGAIN 位控制色同步锁定 PLL 的环路增益。

值越大响应速度越快，但是容易受噪声的影响。

(3) 色同步锁定 PLL 的同步限制控制

通过 LOCKLIMIT 位控制色同步锁定 PLL 的同步限制。

LOCKLIMIT 位的值越大，越容易解除色同步锁定 PLL 自激频率的锁定，也越容易进行重新搜索。

(4) 色同步锁定 PLL 的环路增益控制

通过 BCOFREERUN 位控制色同步锁定 PLL 的自激振荡。

通过将 BCOFREERUN 位置“1”，使色同步锁定 PLL 不依赖输入而进行自激振荡。通常必须将此位置“0”。

(5) 色度解码无信号时的颜色体系控制

通过 DEFAULTSYS 位控制色度解码无信号时的颜色体系。

表 30.15 无信号时的颜色体系

DEFAULTSYS	无信号时的颜色体系
0	NTSC
1	PAL
2	SECAM
3	未指定

(6) 颜色体系判定控制

通过 NONTSC358 位、NONTSC443 位、NOPALM 位、NOPALN 位、NOPAL443 位和 NOSECAM 位控制颜色体系的判定。

颜色体系的判别能选择全自动控制、手动控制以及一部分颜色体系的自动控制（不判别指定以外的内容）。如果不属于任何颜色体系，就作为 DEFAULTSYS 位指定的颜色体系运行。

能分别设定各种类型的判别控制（ON/OFF）。能通过只识别 1 个颜色体系来固定颜色体系，颜色体系的判别方法如表 30.16 所示。

表 30.16 颜色体系判别控制表

	NOSECAM	NOPAL443	NOPALN	NOPALM	NONTSC443	NONTSC358
Auto	0	0	0	0	0	0
NTSC-3.58 (M)	1	1	1	1	1	0
NTSC-4.43	1	1	1	1	0	1
PAL-M	1	1	1	0	1	1
PAL-N	1	1	0	1	1	1
PAL-4.43	1	0	1	1	1	1
SECAM	0	1	1	1	1	1

在自动设定的情况下，将颜色体系的判别结果保存到寄存器。根据寄存器的值进行颜色体系判别的结果如表 30.17 和表 30.18 所示。

表 30.17 颜色体系判定表 (1)

COLORSYS[1:0]	FSCMODE	FVMODE	判别结果
0: NTSC	0: 3.58MHz	don't care	NTSC-M
0: NTSC	1: 4.43MHz	don't care	NTSC-4.43
1: PAL	0: 3.58MHz	0: 50Hz	PAL-N
1: PAL	0: 3.58MHz	1: 60Hz	PAL-M
1: PAL	1: 4.43MHz	0: 50Hz	PAL-B、H、I、G、D
1: PAL	1: 4.43MHz	1: 60Hz	PAL-60
2: SECAM	—	—	SECAM
3: unknown	—	—	无法判别

表 30.18 颜色体系判定表 (2)

	ISNTSC	ISPAL	ISSECAM
无法判断	0	0	0
NTSC	1	0	0
PAL	0	1	0
SECAM	0	0	1

30.4.24 色同步选通脉冲控制寄存器 (BTGPCR)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BGPCH ECK	BGPWIDTH[6:0]							BGPSTART[7:0]							
初始值:	0	0	1	0	0	1	0	0	1	0	0	0	0	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	BGPCHECK	0	R/W	色同步选通脉冲的位置确认 用白线表示色同步选通脉冲的前沿和后沿。
14 ~ 8	BGPWIDTH[6:0]	0100100	R/W	色同步选通脉冲的宽度 (推荐值: 88) 以 27MHz 时钟周期为单位指定从 64 个时钟宽度开始的偏移量。
7 ~ 0	BGPSTART[7:0]	10000 010	R/W	色同步选通脉冲的开始位置 (推荐值: 50) 以 27MHz 时钟周期为单位指定离水平同步基准的位置。

(1) 色同步选通脉冲控制

通过 BGPWIDTH 位和 BGPSTART 位控制色同步选通脉冲的时序。

指定从作为色同步锁定 PLL 基准信号的图像信号中抽取色同步的色同步选通脉冲位置。为了防止 VTR 中的色同步信号的插入位置发生偏差, 通常从水平同步信号的后半部分开始色同步选通脉冲并且以包含标准位置的形式进行设定。

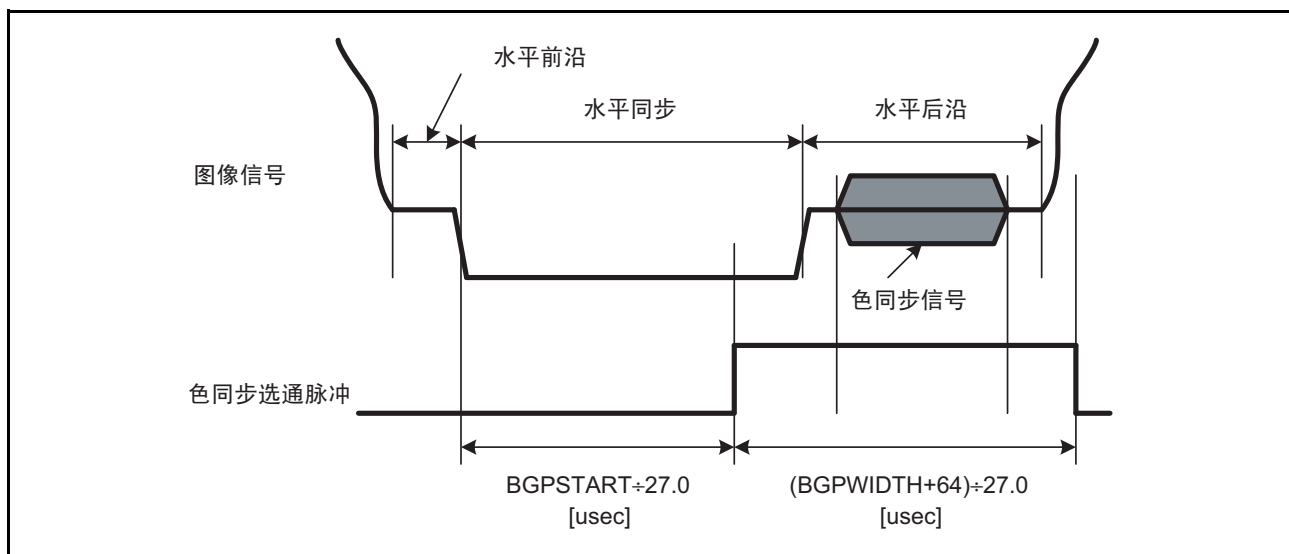


图 30.16 色同步选通脉冲的生成时序

(2) 色同步选通脉冲的位置确认

通过 BGPCHECK 位, 在画面上控制色同步选通脉冲的位置确认。

按以下步骤进行位置确认:

- 色同步选通的位置确认: 将 BGPCHECK 位置 “1”。
- 输入图像信号捕捉区左端设定: 将 SRCLEFT 位置 “0”, 并且将视频显示控制器 4 的缩放部的捕捉水平寄存器 (SCL0_DS3).RES_HS[10:0] 位置 “0”。
- 通过 BGPSTART 位和 BGPWIDTH 位调整脉冲的位置和宽度。

30.4.25 ACC 控制寄存器 (1) (ACCCR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	KILLEROFFSET[3:0]				ACC MODE	ACCMAXGAIN [1:0]			ACCLEVEL[8:0]							
初始值:	1	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 12	KILLEROFFSET [3:0]	1000	R/W	KILLERLEVEL+KILLEROFFSET 为消色器 OFF 时的电平。 此电平对应色同步信号的 Peak-to-Peak 振幅。
11	ACCMODE	0	R/W	ACC 运行模式 0: 自动设定增益 1: 手动设定增益
10、9	ACCMAXGAIN [1:0]	00	R/W	ACC 最大增益 在 ACCMODE 为 “0” (自动设定增益) 时有效。 0: 6 倍 1: 8 倍 2: 12 倍 3: 16 倍
8 ~ 0	ACCLEVEL[8:0]	100100 100	R/W	ACC 基准的色同步振幅 (推荐值: 236) 在 ACCMODE 为 “0” (自动设定增益) 时有效。 以 1 个 LSB 为单位设定 Peak-to-Peak 的振幅值。

(1) 消色器的偏移控制

通过 KILLEROFFSET 位控制消色器 OFF 时的滞后。

如果 KILLEROFFSET 位的值太大, 只要色同步振幅不够大, 消色器就不 OFF。如果值太小, 消色器就会因噪声的影响而重复 ON/OFF。

标准设定值为 “4” ~ “10”。

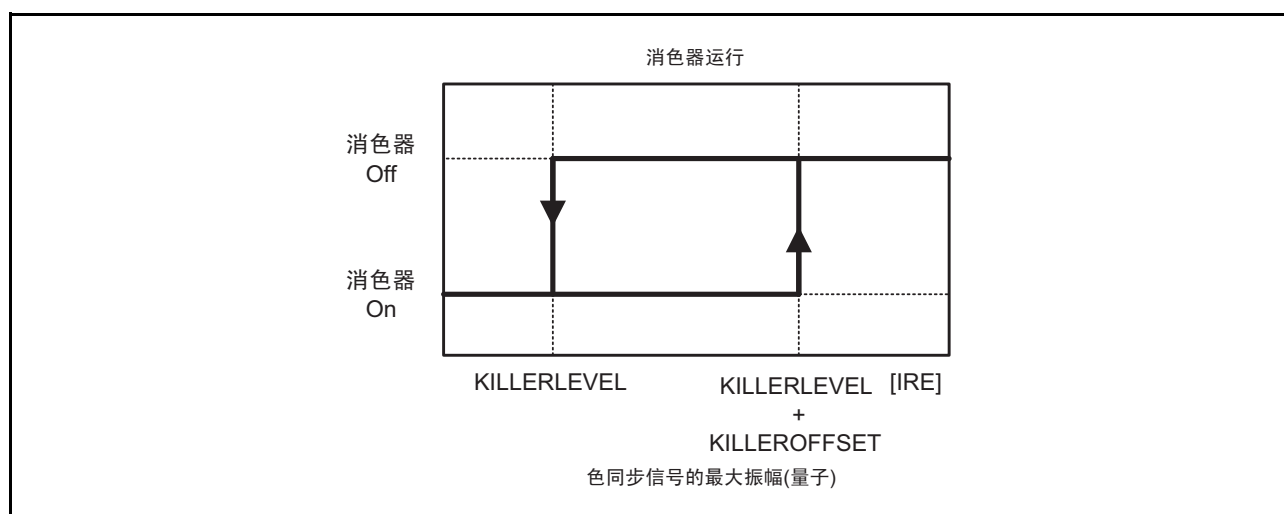


图 30.17 消色器运行

(2) ACC 运行控制

通过 ACCMODE 位控制 ACC 的运行。

表 30.19 ACC 的运行模式

ACCMODE	色彩增益调整
0	自动
1	手动

(3) ACC 最大增益控制

通过 ACCMAXGAIN 位控制 ACC 的最大增益。

此位只在 ACCMODE 位为“0”时有效。

表 30.20 ACC 的最大增益

ACCMAXGAIN	色彩增益的最大值
0	6[倍]
1	8[倍]
2	12[倍]
3	16[倍]

(4) ACC 电平控制

通过 ACCLEVEL 位控制色度信号增益校正后的色同步振幅。

此位只在 ACCMODE 位为“1”时有效。

ACC 进行增益调整，使输入的色度信号的色同步振幅和 ACCLEVEL 的电平值相同。

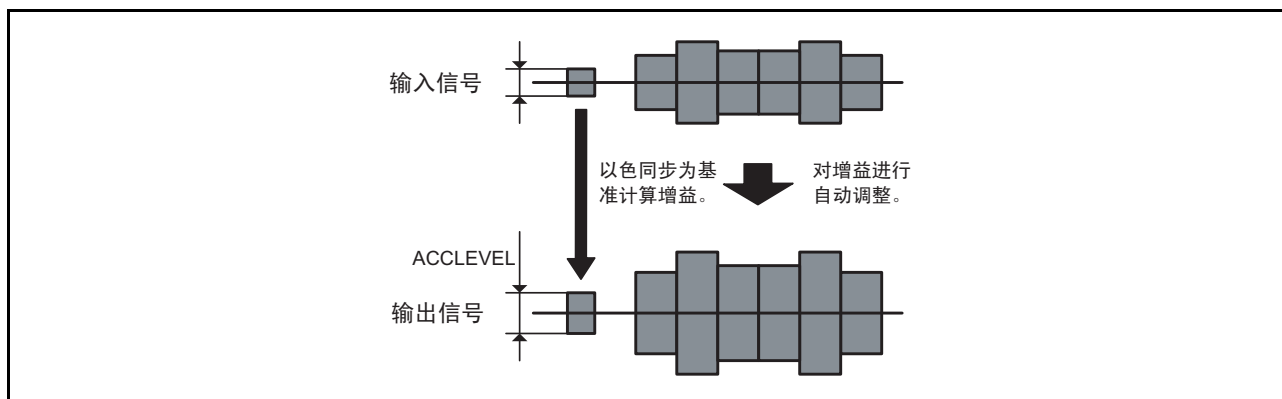


图 30.18 ACC 电平设定

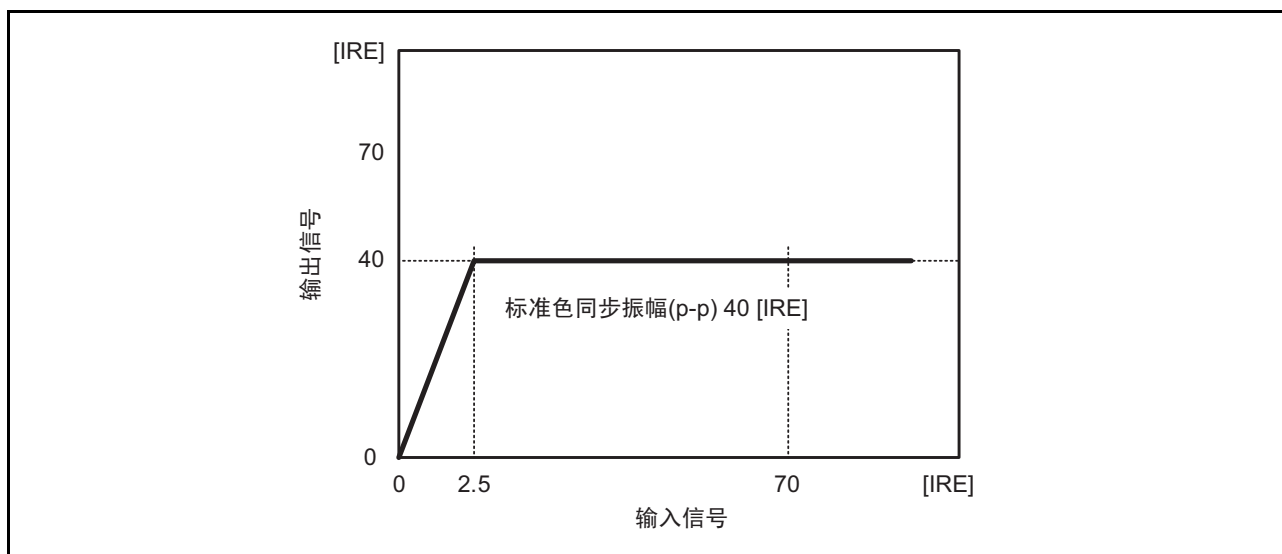


图 30.19 ACC 的输入 / 输出特性

表 30.21 ACC 特性表

输入色同步信号电平	输出色同步信号电平
$\geq -24.1[\text{dB}]$	基准振幅 (可变) \pm 容许误差 (可变)
$\leq -24.1[\text{dB}]$	与输入电平成比例减少。

30.4.26 ACC 控制寄存器 (2) (ACCCR2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CHROMASUB GAIN[1:0]		CHROMAMAINGAIN[8:0]								
初始值:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10、9	CHROMA SUBGAIN[1:0]	00	R/W	色度手动增益 (副) 在 ACCMODE 位为“1” (手动设定增益) 时有效。 0: 1 倍 1: 2 倍 2: 4 倍 3: 8 倍
8 ~ 0	CHROMA MAINGAIN[8:0]	100000 000	R/W	色度手动增益 (主) 在 ACCMODE 位为“1” (手动设定增益) 时有效。 256 相当于 1 倍。

(1) 色度增益调整 (手动) 控制

通过 CHROMASUBGAIN 位和 CHROMAMAINGAIN 位控制色度增益。

此位只在 ACCMODE 位为“1”时有效。

$$C \text{ 信号输出} = C \text{ 信号输入} \times (\text{CHROMASUBGAIN} + (\text{CHROMAMAINGAIN} \div 256))$$

30.4.27 ACC 控制寄存器 (3) (ACCCR3)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ACCRESPONSE [1:0]		ACCPRECIS[5:0]						KILLER MODE	KILLERLEVEL[5:0]					—	
初始值:	0	1	0	1	0	1	0	0	0	0	0	1	0	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

位	位名	初始值	R/W	说明
15、14	ACCRESPONSE [1:0]	01	R/W	ACC 的响应速度 值越大响应速度越快，但是容易受噪声的影响。
13 ~ 8	ACCPRECIS[5:0]	010100	R/W	ACC 增益调整精度 以 10 位精度的 1 个 LSB 为单位设定 ACC 调整后的色同步振幅的容许误差。
7	KILLERMODE	0	R/W	强制消色模式 ON/OFF 0: 自动判定 1: 强制消色器 ON
6 ~ 1	KILLERLEVEL [5:0]	001001	R/W	消色器的运行开始点 以 10 位精度的 1 个 LSB 为单位设定 Peak-to-Peak 振幅的 1/2。
0	—	0	R	保留位 读写值都为“0”。

(1) ACC 的响应速度控制

通过 ACCRESPONSE 位控制 ACC 的响应速度。

值越大响应速度越快，值越小响应速度越慢（值越大响应速度越快，但是容易受噪声影响）。

(2) ACC 的容许误差范围控制

通过 ACCPRECIS 位对 ACCLEVEL 位（目标值）输出信号的色同步信号振幅的容许误差进行控制。

在 ACCLEVEL 位为“236”并且 ACCPRECIS 位为“20”的情况下，在以下的范围内通过 ACC 进行增益调整
(236-20) < 输出信号的色同步振幅 < (236+20)

(3) 消色运行模式控制

通过 KILLERMODE 位控制消色运行模式。

当 KILLERMODE 位为“1”时，强制消色器为 ON。

(4) 消色电平控制

通过 KILLERLEVEL 位控制消色器 ON 时的电平。

详细内容请参照“30.4.25 ACC 控制寄存器 (1) (ACCCR1)”的“(1) 消色器的偏移控制”。

30.4.28 TINT 控制寄存器 (TINTCR)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TINTSUB[5:0]						TINTMAIN[9:0]									
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 10	TINTSUB[5:0]	000000	R/W	R-Y 解调轴的微调 (只对 NTSC/PAL 有效) 以 360/1024 度为单位, 用 2 的补码进行设定。
9 ~ 0	TINTMAIN[9:0]	0000000 000	R/W	色调调整量 (只对 NTSC/PAL 有效) 以 360/1024 度为单位, 用 2 的补码进行设定。

(1) R-Y 轴校正控制

通过 TINTSUB 位控制 ± 11.25 度 R-Y 轴的相位。

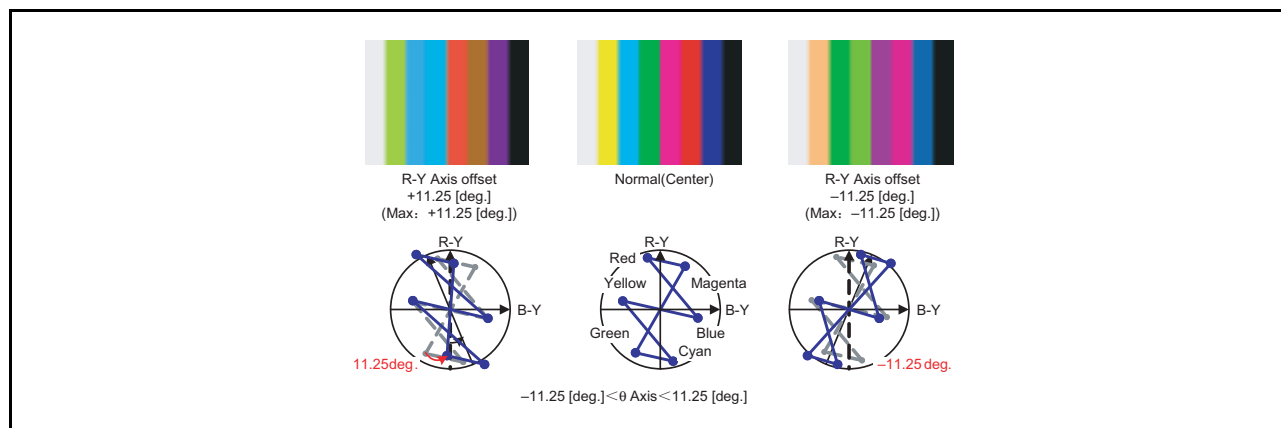


图 30.20 R-Y 轴校正例子

(2) 色调调整 (TINT) 校正控制

通过 TINTMAIN 位控制 $0 \sim 360$ 度解调轴的相位。

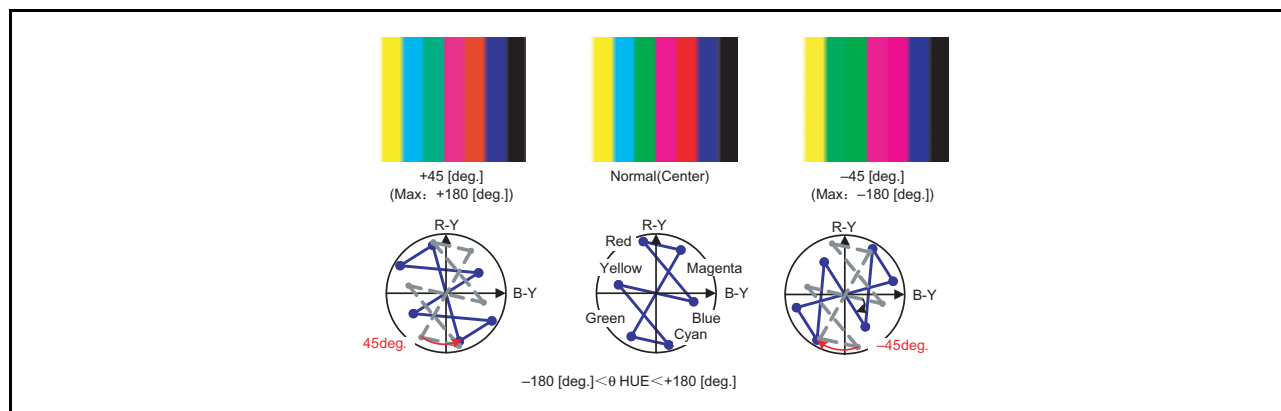


图 30.21 色调调整 (TINT) 校正例子

30.4.29 Y/C 延迟控制、色度解码控制寄存器 (YCDCR)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	LUMADELAY[4:0]				—	CHROM ALPF	DEMODMODE [1:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8 ~ 4	LUMADELAY [4:0]	00000	R/W	亮度信号的延迟调整 用 2 的补码设定 -16 个时钟 ~ +15 个时钟。
3	—	0	R	保留位 读写值都为“0”。
2	CHROMALPF	0	R/W	用于解调后色度的 LPF 0: 不使用 1: 使用
1、0	DEMODMODE [1:0]	10	R/W	解调前的行平均处理 0: 无 1: 禁止设定 2: 只限于 PAL 3: 禁止设定

(1) Y/C 延迟调整控制

通过 LUMADELAY 位控制 Y/C 的延迟。

表 30.22 Y/C 延迟调整控制

LUMADELAY	运行
31	Y 信号延迟 -1[clk]
:	:
16	Y 信号延迟 -16[clk]
0	无延迟
1	Y 信号延迟 +1[clk]
:	:
15	Y 信号延迟 +15[clk]

(2) 解调后的频带限制滤波器控制

通过 CHROMALPF 位对解调后的频带限制滤波器进行控制。

表 30.23 解调后的频带限制滤波器控制

CHROMALPF	运行
0	解调后的频带限制滤波器 OFF
1	解调后的频带限制滤波器 ON

(3) 色度解码运行控制

通过 DEMODMODE 位控制色度解调的运行模式。

表 30.24 色度解码运行模式表

DEMODMODE	输出色同步信号的电平
0	进行 1 行解调。
2	只有 PAL 才进行 2 行解调。
1、3	禁止设定。

30.4.30 AGC 控制寄存器 (1) (AGCCR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	DOREDUCE	NOREDUCE	AGCRESPONSE [2:0]			AGCLEVEL[8:0]								
初始值:	0	0	0	0	1	0	1	0	1	1	1	0	1	1	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13	DOREDUCE	0	R/W	VBI 期间的同步振幅检测的手动切换 0: 将同步振幅设定为 AGC 的基准 1: 将 AGC 增益设定为通常的 3/4 倍
12	NOREDUCE	0	R/W	VBI 期间的同步振幅检测的切换 0: 检测同步振幅 1: 不检测同步振幅
11 ~ 9	AGCRESPONSE [2:0]	101	R/W	AGC 的响应速度 值越大响应速度越快, 但是容易受噪声的影响。
8 ~ 0	AGCLEVEL[8:0]	011101 100	R/W	同步脉冲的基准振幅 设定范围: 0 ~ 511, 无符号 10 位

(1) VBI 期间的同步振幅检测控制

通过 DOREDUCE 位和 NOREDUCE 位控制 AGC 的 VBI 期间的同步信号振幅变动检测。

表 30.25 VBI 期间的同步振幅检测

DOREDUCE	VBI 期间的同步振幅检测
0	将同步振幅设定为 AGC 的基准。
1	将 AGC 增益设定为通常的 3/4 倍。

表 30.26 VBI 期间的同步振幅检测

NOREDUCE	VBI 期间的同步振幅检测
0	检测同步振幅。
1	不检测同步振幅。

(2) AGC 的响应速度控制

通过 AGCRESPONSE 位控制 AGC 的响应速度。

值越大响应速度越快, 值越小响应速度越慢 (值越大响应速度越快, 但是容易受噪声的影响)。

考虑因弱电场和 VTR 的特殊播放 (快进播放 / 倒带播放) 而导致误动作, 推荐值为“5”。

(3) AGC 的电平控制

通过 AGCLEVEL 位控制 AGC 的目标电平。

在通过 10 位 A/D 转换器对 NTSC 信号进行量化时，A/D 转换器全量程的同步信号振幅为：

$$1023[\text{LSB}] \times (40[\text{IRE}] \div 173[\text{IRE}]) = 236.53179[\text{LSB}]$$

各输入信号格式的 AGC 电平理想值如表 30.27 所示。

表 30.27 AGC 电平设定值（理想值）

输入信号的格式	AGCLEVEL[8:0]
NTSC	236
PAL/SECAM	248

表 30.28 AGC 特性表

输入同步信号的电平	输出同步信号的电平
$\geq 0[\text{dB}]$	与输入电平成比例增加。
$-8.52[\text{dB}] \sim 0[\text{dB}]$	基准振幅（可变） \pm 容许误差（可变）
$\leq -8.52[\text{dB}]$	与输入电平成比例减少。

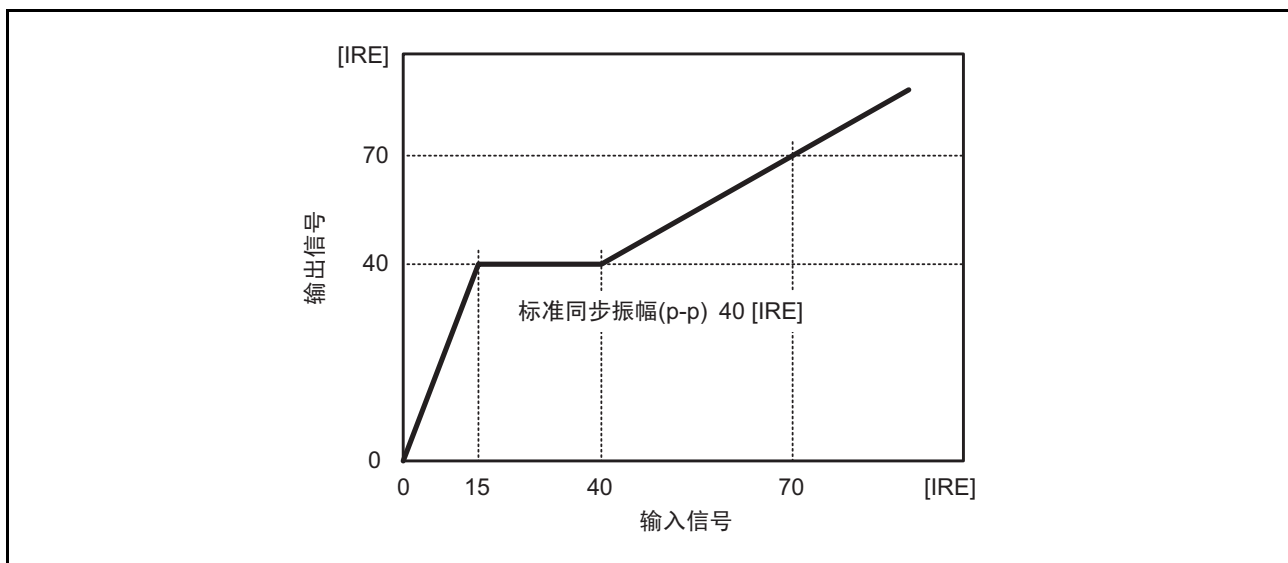


图 30.22 AGC 特性图（同步信号振幅基准）

30.4.31 AGC 控制寄存器 (2) (AGCCR2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	—	—	AGCPRECIS[5:0]						—	—	—	—	—	—	—	—	—
初始值:	1	1	0	0	1	0	1	0	0	1	0	0	0	0	0	0	
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	

位	位名	初始值	R/W	说明
15、14	—	全 1	R	保留位 读写值都为“1”。
13 ~ 8	AGCPRECIS[5:0]	001010	R/W	AGC 增益调整精度 以 10 位精度的 1 个 LSB 为单位设定 AGC 调整后的同步脉冲振幅的容许误差。
7	—	0	R	保留位 读写值都为“0”。
6	—	1	R	保留位 读写值都为“1”。
5 ~ 0	—	全 0	R/W	保留位 读写值都为“0”。

(1) AGC 的容许误差范围控制

通过 AGCPRECIS 位对 AGCLEVEL 位（目标值）输出信号的同步信号振幅的容许误差进行控制。

当 AGCLEVEL 位为“236”并且 AGCPRECIS 位为“10”时，在以下范围内固定 AGC 增益：

$$(236-10) < \text{输出信号的同步振幅} < (236+10)$$

此时，将 PGA 的增益设定为 2 个阶梯的范围。对于同步信号振幅稳定的图像，建议在强电场时将 PGA 的增益设定为 1 个阶梯的范围（AGCPRECIS 位为“4”）。在上述设定中，有可能因弱电场等引起摆动，所以推荐值为“10”。

30.4.32 峰值限制器的控制寄存器 (PKLIMITCR)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PEAKLEVEL [1:0]		PEAKATTACK [1:0]		PEAKRELEASE [1:0]		PEAKRATIO [1:0]		MAXPEAKSAMPLES [7:0]							
初始值:	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15、14	PEAKLEVEL[1:0]	00	R/W	由峰值限制器限制的峰值亮度值（图像信号电平） 0: 限制器 OFF 1: 1008 个 LSB 2: 992 个 LSB 3: 960 个 LSB 【注】 在 AGC OFF 的情况下，与 PEAKLEVEL 的值无关，峰值限制器不运行。
13、12	PEAKATTACK [1:0]	10	R/W	减少峰值限制器增益时的响应速度 值越大响应速度越快。
11、10	PEAKRELEASE [1:0]	00	R/W	增加峰值限制器增益时的响应速度 值越大响应速度越快。
9、8	PEAKRATIO[1:0]	00	R/W	峰值限制器的最大压缩率 0: 压缩到 50% 1: 压缩到 25% 2: 压缩到 12.5% 3: 压缩到 0%
7 ~ 0	MAXPEAK SAMPLES[7:0]	00000000	R/W	发生上溢的像素的容许最大数 以 1024 像素为单位进行指定。如果超过此值，峰值限制器就开始运行。

(1) 峰值限制器的电平控制

通过 PEAKLEVEL 位控制由峰值限制器限制的峰值亮度值。

对大于等于 PEAKLEVEL 位设定值的像素进行计数。如果存在大于等于 MAXPEAKSAMPLES 位设定值的像素个数，峰值限制器功能就对减少增益有效。

表 30.29 峰值限制器的电平控制

PEAKLEVEL	输出同步信号的电平
0	峰值限制器 OFF
1	峰值限制器以 1008 个 LSB 运行。
2	峰值限制器以 992 个 LSB 运行。
3	峰值限制器以 960 个 LSB 运行。

(2) 峰值限制器的响应速度控制

通过 PEAKATTACK 位对减少峰值限制器增益时的响应速度进行控制。

值越大响应速度越快。

(3) 峰值限制器的响应速度控制

通过 PEAKRELEASE 位对增加峰值限制器增益时的响应速度进行控制。
值越大响应速度越快。

(4) 峰值限制器的增益减少控制

通过 PEAKRATIO 位控制峰值限制器的最大压缩率。
通过峰值限制器功能对减少增益时的增益减少量（压缩率）进行控制。

表 30.30 峰值限制器的增益减少控制

PEAKRATIO	输出同步信号的电平
0	压缩到 50.0%
1	压缩到 25.0%
2	压缩到 12.5%
3	压缩到 0%

(5) 峰值限制器的判定控制

通过 MAXPEAKSAMPLES 位对发生上溢的像素的容许最大数进行控制。

在垂直有效期间，对大于等于 PEAKLEVEL 位设定值的像素进行计数。如果存在大于等于 MAXPEAKSAMPLES 位设定值的像素个数，峰值限制器功能就对减少增益有效。

容许最大数为 MAXPEAKSAMPLES 位的设定值乘 1024 后的值。

30.4.33 超范围控制寄存器 (1) (RGORCR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RADJ_O_LEVEL0[9:0]									
初始值:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	RADJ_O_ LEVEL0[9:0]	111111 1111	R/W	A/D 过阈值电平 (LEVEL0 和 LEVEL1 之间) 从 LEVEL0 (正常空间) 到 LEVEL3 (完全超范围)

(1) A/D 过阈值电平 (LEVEL0 和 LEVEL1 之间) 控制

通过 RADJ_O_LEVEL0 位控制 A/D 过阈值电平 (LEVEL0 和 LEVEL1 之间)。

阈值电平的对应如图 30.23 所示。

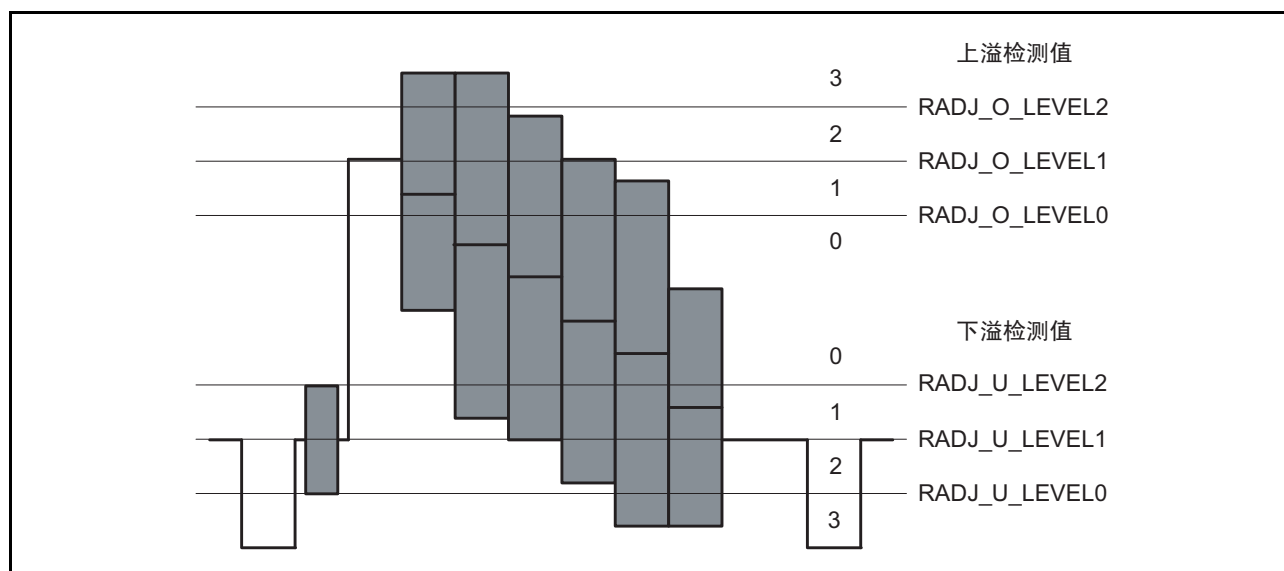


图 30.23 超范围控制的判定区域

30.4.34 超范围控制寄存器 (2) (RGORCR2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RADJ_U_LEVEL0[9:0]									
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	RADJ_U_ LEVEL0[9:0]	000000 0000	R/W	A/D 欠阈值电平 (LEVEL3 和 LEVEL2 之间) 从 LEVEL0 (正常空间) 到 LEVEL3 (完全欠范围)

(1) A/D 欠阈值电平 (LEVEL3 和 LEVEL2 之间) 控制

通过 RADJ_U_LEVEL0 位控制 A/D 欠阈值电平 (LEVEL3 和 LEVEL2 之间)。

阈值电平的对应请参照“30.4.33 超范围控制寄存器 (1) (RGORCR1)”。

30.4.35 超范围控制寄存器 (3) (RGORCR3)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RADJ_O_LEVEL1[9:0]									
初始值:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	RADJ_O_ LEVEL1[9:0]	111111 1111	R/W	A/D 过阈值电平 (LEVEL1 和 LEVEL2 之间) 从 LEVEL0 (正常空间) 到 LEVEL3 (完全超范围)

(1) A/D 过阈值电平 (LEVEL1 和 LEVEL2 之间) 控制

通过 RADJ_O_LEVEL1 位控制 A/D 过阈值电平 (LEVEL1 和 LEVEL2 之间)。

阈值电平的对应请参照“30.4.33 超范围控制寄存器 (1) (RGORCR1)”。

30.4.36 超范围控制寄存器 (4) (RGORCR4)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RADJ_U_LEVEL1[9:0]									
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	RADJ_U_ LEVEL1[9:0]	000000 0000	R/W	A/D 欠阈值电平 (LEVEL2 和 LEVEL1 之间) 从 LEVEL0 (正常空间) 到 LEVEL3 (完全欠范围)

(1) A/D 欠阈值电平 (LEVEL2 和 LEVEL1 之间) 控制

通过 RADJ_U_LEVEL1 位控制 A/D 欠阈值电平 (LEVEL2 和 LEVEL1 之间)。
阈值电平的对应请参照“30.4.33 超范围控制寄存器 (1) (RGORCR1)”。

30.4.37 超范围控制寄存器 (5) (RGORCR5)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RADJ_O_LEVEL2[9:0]									
初始值:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	RADJ_O_ LEVEL2[9:0]	111111 1111	R/W	A/D 过阈值电平 (LEVEL2 和 LEVEL3 之间) 从 LEVEL0 (正常空间) 到 LEVEL3 (完全超范围)

(1) A/D 过阈值电平 (LEVEL2 和 LEVEL3 之间) 控制

通过 RADJ_O_LEVEL2 位控制 A/D 过阈值电平 (LEVEL2 和 LEVEL3 之间)。
阈值电平的对应请参照“30.4.33 超范围控制寄存器 (1) (RGORCR1)”。

30.4.38 超范围控制寄存器 (6) (RGORCR6)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—						RADJ_U_LEVEL2[9:0]									
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	RADJ_U_ LEVEL2[9:0]	000000 0000	R/W	A/D 欠阈值电平 (LEVEL1 和 LEVEL0 之间) 从 LEVEL0 (正常空间) 和 LEVEL3 (完全欠范围) 之间

(1) A/D 欠阈值电平 (LEVEL1 和 LEVEL0 之间) 控制

通过 RADJ_U_LEVEL2 位控制 A/D 欠阈值电平 (LEVEL1 和 LEVEL0 之间)。

阈值电平的对应请参照 “30.4.33 超范围控制寄存器 (1) (RGORCR1)”。

30.4.39 超范围控制寄存器 (7) (RGORCR7)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TEST_MONI[2:0]			RADJ_MIX_K_FIX[2:0]			—	—	—	—	—	—	UCMP_SW	DCMP_SW	HWIDE_SW
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	TEST_MONI[2:0]	000	R/W	测试模式 0 ~ 3: 正常运行 4: LEVEL0 部分输出黑色信号 5: LEVEL1 部分输出黑色信号 6: LEVEL2 部分输出黑色信号 7: LEVEL3 部分输出黑色信号
11 ~ 9	RADJ_MIX_K_FIX[2:0]	000	R/W	超欠范围强制模式 0 ~ 3: 自动检测 4: 固定为 LEVEL0 (正常运行) 5: 固定为 LEVEL1 (基本正常运行) 6: 固定为 LEVEL2 (几乎超范围) 7: 固定为 LEVEL3 (完全超范围状态)
8 ~ 3	—	全 0	R	保留位 读写值都为“0”。
2	UCMP_SW	0	R/W	超范围判别允许 0: 不检测超范围 1: 检测超范围
1	DCMP_SW	0	R/W	欠范围判别允许 0: 不检测超范围 1: 检测超范围
0	HWIDE_SW	1	R/W	超欠范围电平的水平扩大 0: 无水平扩大 1: 有水平扩大

(1) 超范围测试控制

通过 TEST_MONI 位控制超范围测试。

(2) 超欠范围强制模式控制

通过 RADJ_MIX_K_FIX 位控制超欠范围的强制允许。

(3) 超范围判别允许控制

通过 UCMP_SW 位控制超范围的判别允许。

(4) 欠范围判别允许控制

通过 DCMP_SW 位控制欠范围的判别允许。

(5) 超欠范围电平的水平扩大控制

通过 HWIDE_SW 位控制超欠范围电平的水平扩大。

30.4.40 水平 AFC 相位比较器的反馈量调整寄存器 (AFPCFPCR)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PHDET_FIX	—	PHDET_DIV[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	PHDET_FIX	0	R/W	强制 LOWGAIN 控制 0: 使用 LOWGAIN 判定结果 1: 强制 (通过 PHDET_DIV 进行调整)
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PHDET_DIV[2:0]	101	R/W	同步信号的锁定稳定性降低时的相位比较结果的反馈量调整 0: 1/1 1: 1/2 2: 1/4 3: 1/8 4: 1/16 5: 1/32 6 ~ 7: 禁止设定

(1) 相位比较器的反馈量调整

在锁定稳定性降低时，调整相位比较结果的反馈量。

分母越大信号反应速度越慢。

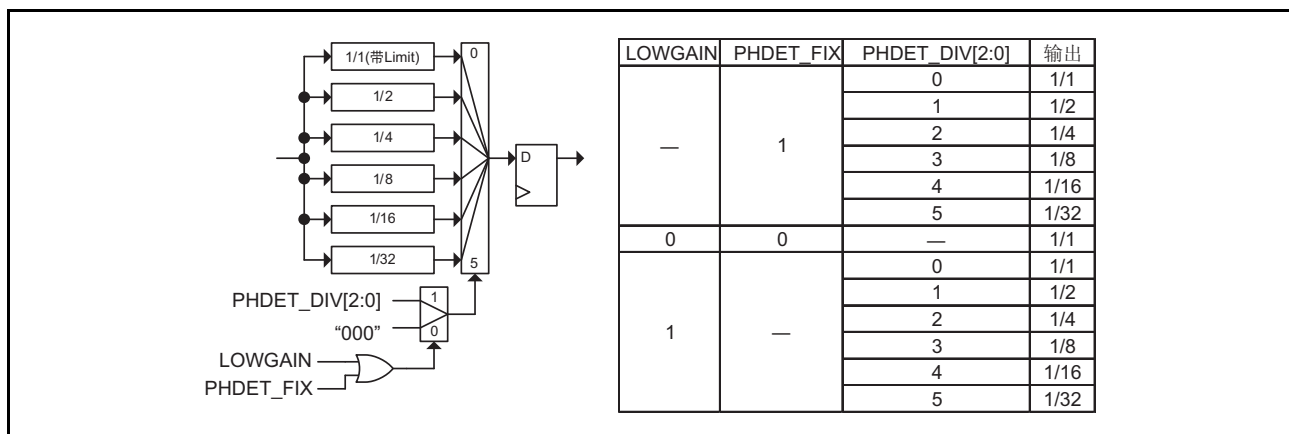


图 30.24 相位比较器的反馈量调整

30.4.41 寄存器的更新控制寄存器 (RUPDCR)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NEWSE TTING	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15	NEWSETTING	0	R/W	TGCR1 ~ 3 寄存器的 V 更新允许 1: 允许 0: 禁止
14 ~ 0	—	全 0	R	保留位 读写值都为“0”。

(1) TGCR1 ~ 3 寄存器的 V 更新允许控制

通过 NEWSETTING 位控制 TGCR1 ~ 3 寄存器的 V 更新允许。

30.4.42 同步分离状态、垂直周期读寄存器 (VSYNCSTR)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FHCOUNT[0]	FHLOCK	ISNOISY	FHMODE	NOSIGNAL	FVLOCK	FVMODE	INTERLACED	FVCOUNT[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15	FHCOUNT[0]	0	R	水平 AFC 的振荡周期 (bit0) 以 27MHz 时钟的 1/64 为单位。
14	FHLOCK	0	R	水平 AFC 的锁定判定 0: 未锁定 1: 锁定
13	ISNOISY	0	R	通过同步分离进行的低 S/N 信号判别结果 0: 不是低 S/N 信号 1: 是低 S/N 信号
12	FHMODE	0	R	倍速判定结果 0: 正常运行 (525i/625i 等) 1: 倍速运行 (525p/625p 等)
11	NOSIGNAL	0	R	无信号判别结果 0: 有垂直同步信号 1: 无垂直同步信号
10	FVLOCK	0	R	垂直递减计数的锁定判定 0: 未锁定 1: 锁定
9	FVMODE	0	R	垂直递减计数振荡模式 0: 50Hz 1: 60Hz
8	INTERLACED	0	R	隔行扫描判别结果 0: 不是隔行扫描 1: 是隔行扫描
7~0	FVCOUNT[7:0]	00000000	R	垂直周期测量结果 (以 0.1ms 为单位)

(1) 读水平 AFC 的振荡周期

通过 FHCOUNT 位读水平 AFC 振荡周期的 bit0。

(2) 读水平 AFC 的锁定判定

通过 FHLOCK 位读水平 AFC 的锁定判定。

(3) 读通过同步分离进行的低 S/N 信号判别结果

通过 ISNOISY 位读通过同步分离进行的低 S/N 信号判别结果。

(4) 读倍速判定结果

通过 FHMODE 位读倍速判定结果。

(5) 读无信号判别结果

通过 NOSIGNAL 位读无信号判别结果。

(6) 读垂直递减计数的锁定判定

通过 FVLOCK 位读垂直递减计数的锁定判定。

(7) 读隔行扫描判别结果

通过 INTERLACED 位读隔行扫描判别结果。

(8) 读垂直周期测量结果

通过 FVCOUNT 位读垂直周期测量结果。

30.4.43 水平周期读寄存器 (HSYNC CSR)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FHCCOUNT[16:1]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 0	FHCCOUNT[16:1]	H'0000	R	水平 AFC 的振荡周期 (bit16-bit1) 以 27MHz 时钟的 1/64 为单位。

(1) 读水平 AFC 的振荡周期

通过 FHCCOUNT 位读水平 AFC 振荡周期的高位。

30.4.44 数字钳位读寄存器 (1) (DCPSR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CLAMPLEVEL_CB[5:0]						CLAMPLEVEL_Y[9:0]									
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 10	CLAMP LEVEL_CB[5:0]	000000	R	数字钳位被减值 (Cb 信号) 从黑色信号基准电平开始的偏移 (以 1 个 LSB 为单位, 用 2 的补码 进行设定)
9 ~ 0	CLAMP LEVEL_Y[9:0]	000000 0000	R	数字钳位被减值 (Y 信号) 从黑色信号基准电平开始的偏移 (以 1 个 LSB 为单位, 用 2 的补码 进行设定)

(1) 读 Cb 信号的数字钳位被减值

通过 CLAMPLEVEL_CB 位读 Cb 信号的数字钳位被减值。

(2) 读 Y 信号的数字钳位被减值

通过 CLAMPLEVEL_Y 位读 Y 信号的数字钳位被减值。

30.4.45 数字钳位读寄存器 (2) (DCPSR2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CLAMPLEVEL_CR[5:0]						—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 10	CLAMP LEVEL_CR[5:0]	000000	R	数字钳位被减值 (Cr 信号) 从黑色信号基准电平开始的偏移 (以 1 个 LSB 为单位, 用 2 的补码 进行设定)
9 ~ 0	—	全 0	R	保留位 读写值都为“0”。

(1) 读 Cr 信号的数字钳位被减值

通过 CLAMPLEVEL_CR 位读 Cr 信号的数字钳位被减值。

30.4.46 噪声检测读寄存器 (NSDSR)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ACFSTRENGTH[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 0	ACFSTRENGTH [15:0]	H'0000	R	数字钳位脉冲位置（正常基准位置）中的噪声自相关强度 当 ACFLAGTIME 位为“0”时，此值几乎相当于基准位置中的噪声功率。 此时，平方根几乎相当于噪声振幅，对数几乎相当于 S/N（相对值）。

(1) 读数字钳位脉冲位置（正常基准位置）中的噪声自相关强度

通过 ACFSTRENGTH 位读数字钳位脉冲位置（正常基准位置）中的噪声自相关强度。

30.4.47 色度解码读寄存器 (1) (CROMASR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	COLORSYS [1:0]		FSC MODE	FSC LOCK	NO BURST	ACCSUBGAIN [1:0]		ACCMaingain[8:0]								
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15、14	COLORSYS[1:0]	00	R	颜色体系的判别结果 0: NTSC 1: PAL 2: SECAM 3: 无法判别
13	FSCMODE	0	R	色度副载波频率的判别结果 0: 3.58MHz 1: 4.43MHz
12	FSCLOCK	0	R	色同步锁定 PLL 的锁定判定 1: 锁定 0: 未锁定
11	NOBURST	0	R	色同步检测结果 0: 有色同步 1: 无色同步
10、9	ACCSUBGAIN [1:0]	00	R	当前 ACC 增益值 (副) 0: 1 倍 1: 2 倍 2: 4 倍 3: 8 倍
8 ~ 0	ACCMaingain [8:0]	0000000 00	R	当前 ACC 增益值 (主) 256 为 1 倍。

(1) 读颜色体系的判别结果

通过 COLORSYS 位读颜色体系的判别结果。

(2) 读色度副载波频率的判别结果

通过 FSCMODE 位读色彩副载波频率的判别结果。

(3) 读色同步锁定 PLL 的锁定判定

通过 FSCLOCK 位读色同步锁定 PLL 的锁定判定。

(4) 读色同步检测结果

通过 NOBURST 位读色同步检测结果。

(5) 读当前 ACC 增益值 (副)

通过 ACCSUBGAIN 位读当前 ACC 增益值 (副)。

(6) 读当前 ACC 增益值 (主)

通过 ACCMAINGAIN 位读当前 ACC 增益值 (主)。

30.4.48 色度解码读寄存器 (2) (CROMASR2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	ISSE CAM	ISPAL	ISNTSC	—	—	LOCKLEVEL[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12	ISSECAM	0	R	SECAM 检测器的检测结果 0: 不是 SECAM 1: 是 SECAM
11	ISPAL	0	R	PAL 检测器的检测结果 0: 不是 PAL 1: 是 PAL
10	ISNTSC	0	R	NTSC 检测器的检测结果 0: 不是 NTSC 1: 是 NTSC
9、8	—	全 0	R	保留位 读取值为不定值，写入值总是“0”。
7 ~ 0	LOCKLEVEL[7:0]	00000000	R	通过色同步锁定 PLL 进行的低 S/N 信号判别结果 值越大 S/N 越高。

(1) 读 SECAM 检测器的检测结果

通过 ISSECAM 位读 SECAM 检测器的检测结果。

(2) 读 PAL 检测器的检测结果

通过 ISPAL 位读 PAL 检测器的检测结果。

(3) 读 NTSC 检测器的检测结果

通过 ISNTSC 位读 NTSC 检测器的检测结果。

(4) 读通过色同步锁定 PLL 进行的低 S/N 信号判别结果

通过 LOCKLEVEL 位读通过色同步锁定 PLL 进行的低 S/N 信号判别结果。

30.4.49 同步分离读寄存器 (SYNCSSR)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	ISREDUCED	—	—	SYNCDEPTH[9:0]									
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12	ISREDUCED	0	R	VBI 期间的同步振幅检测结果 0: 同步振幅大于图像有效期间的振幅 1: 同步振幅等于图像有效期间的振幅
11、10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	SYNCDEPTH [9:0]	000000 0000	R	同步脉冲振幅的检测结果

(1) 读 VBI 期间的同步振幅检测结果

通过 ISREDUCED 位读 VBI 期间的同步振幅检测结果。

(2) 读同步脉冲振幅的检测结果

通过 SYNCDEPTH 位读同步脉冲振幅的检测结果。

30.4.50 AGC 控制读寄存器 (1) (AGCCSR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HIGHSAMPLES[7:0]								PEAKSAMPLES[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 8	HIGHSAMPLES [7:0]	00000000	R	超过峰值限制器限制的峰值亮度值的像素个数 以 1024 像素为单位。
7 ~ 0	PEAKSAMPLES [7:0]	00000000	R	发生上溢的像素个数 以 1024 像素为单位。

(1) 读超过峰值限制器限制的峰值亮度值的像素个数

通过 HIGHSAMPLES 位读超过峰值限制器限制的峰值亮度值的像素个数。

(2) 读发生上溢的像素个数

通过 PEAKSAMPLES 位读发生上溢的像素个数。

30.4.51 AGC 控制读寄存器 (2) (AGCCSR2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	AGCCON VERGE	AGCGAIN[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	AGCCONVERGE	0	R	AGC 的收敛判定结果 1: 收敛 0: 未收敛
7 ~ 0	AGCGAIN[7:0]	010000 00	R	当前 AGC 增益值 64 相当于 1 倍。

(1) 读 AGC 的收敛判定结果

通过 AGCCONVERGE 位读 AGC 的收敛判定结果。

(2) 读当前 AGC 增益值

通过 AGCGAIN 位读当前 AGC 增益值。

30.4.52 Y/C 分离控制寄存器 (3) (YCSCR3)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	K15[3:0]				K13[5:0]					K11[5:0]						
初始值:	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 12	K15[3:0]	0010	R/W	Y/C 分离 2 维滤波器的选择系数 如果增大此值, 水平 BPF 的适用范围就会变小。
11 ~ 6	K13[5:0]	001000	R/W	Y/C 分离 2 维滤波器的选择系数 如果增大此值, 水平 BPF 的适用范围就会变小。
5 ~ 0	K11[5:0]	000100	R/W	Y/C 分离 2 维滤波器的选择系数 如果增大此值, 水平 BPF 的适用范围就会变小。

(1) Y/C 分离 2 维滤波器的选择系数控制

通过 K15 位控制 Y/C 分离 2 维滤波器的选择系数。
详细内容请参照 “30.5.5(3) 水平和垂直的相关检测”。

(2) Y/C 分离 2 维滤波器的选择系数控制

通过 K13 位控制 Y/C 分离 2 维滤波器的选择系数。
详细内容请参照 “30.5.5(3) 水平和垂直的相关检测”。

(3) Y/C 分离 2 维滤波器的选择系数控制

通过 K11 位控制 Y/C 分离 2 维滤波器的选择系数。
详细内容请参照 “30.5.5(3) 水平和垂直的相关检测”。

30.4.53 Y/C 分离控制寄存器 (4) (YCSCR4)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	K16[3:0]				K14[5:0]						K12[5:0]					
初始值:	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 12	K16[3:0]	0011	R/W	Y/C 分离 2 维滤波器的选择系数 如果增大此值，水平 BPF 的适用范围就会变小。
11 ~ 6	K14[5:0]	010000	R/W	Y/C 分离 2 维滤波器的选择系数 如果增大此值，水平 BPF 的适用范围就会变小。
5 ~ 0	K12[5:0]	000001	R/W	Y/C 分离 2 维滤波器的选择系数 如果增大此值，水平 BPF 的适用范围就会变小。

(1) Y/C 分离 2 维滤波器的选择系数控制

通过 K16 位控制 Y/C 分离 2 维滤波器的选择系数。
详细内容请参照“30.5.5(3) 水平和垂直的相关检测”。

(2) Y/C 分离 2 维滤波器的选择系数控制

通过 K14 位控制 Y/C 分离 2 维滤波器的选择系数。
详细内容请参照“30.5.5(3) 水平和垂直的相关检测”。

(3) Y/C 分离 2 维滤波器的选择系数控制

通过 K12 位控制 Y/C 分离 2 维滤波器的选择系数。
详细内容请参照“30.5.5(3) 水平和垂直的相关检测”。

30.4.54 Y/C 分离控制寄存器 (5) (YCSCR5)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	K22A[7:0]							—	—	K21A[5:0]						
初始值:	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 8	K22A[7:0]	01000000	R/W	Y/C 分离 2 维滤波器的选择系数 如果增大此值, 垂直 BPF 的适用范围就会变小。
7、6	—	全 0	R	保留位 读写值都为“0”。
5 ~ 0	K21A[5:0]	000110	R/W	Y/C 分离 2 维滤波器的选择系数 如果增大此值, 垂直 BPF 的适用范围就会变小。

(1) Y/C 分离 2 维滤波器的选择系数控制

通过 K22A 位控制 Y/C 分离 2 维滤波器的选择系数。

详细内容请参照“30.5.5(3) 水平和垂直的相关检测”。

(2) Y/C 分离 2 维滤波器的选择系数控制

通过 K21A 位控制 Y/C 分离 2 维滤波器的选择系数。

详细内容请参照“30.5.5(3) 水平和垂直的相关检测”。

30.4.55 Y/C 分离控制寄存器 (6) (YCSCR6)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	K22B[7:0]							—	—	K21B[5:0]						
初始值:	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 8	K22B[7:0]	00010000	R/W	Y/C 分离 2 维滤波器的选择系数 如果增大此值, 垂直 BPF 的适用范围就会变小。
7、6	—	全 0	R	保留位 读写值都为“0”。
5 ~ 0	K21B[5:0]	000110	R/W	Y/C 分离 2 维滤波器的选择系数 如果增大此值, 垂直 BPF 的适用范围就会变小。

(1) Y/C 分离 2 维滤波器的选择系数控制

通过 K22B 位控制 Y/C 分离 2 维滤波器的选择系数。

详细内容请参照“30.5.5(3) 水平和垂直的相关检测”。

(2) Y/C 分离 2 维滤波器的选择系数控制

通过 K21B 位控制 Y/C 分离 2 维滤波器的选择系数。

详细内容请参照“30.5.5(3) 水平和垂直的相关检测”。

30.4.56 Y/C 分离控制寄存器 (7) (YCSCR7)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	K23B[3:0]				K23A[3:0]				—	—	—	K24[4:0]				
初始值:	0	1	1	0	0	0	1	1	0	0	1	0	0	1	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 12	K23B[3:0]	0110	R/W	Y/C 分离 2 维滤波器的选择系数 如果增大此值, 垂直 BPF 的适用范围就会变小。
11 ~ 8	K23A[3:0]	0011	R/W	Y/C 分离 2 维滤波器的选择系数 如果增大此值, 垂直 BPF 的适用范围就会变小。
7、6	—	全 0	R	保留位 读写值都为“0”。
5	—	1	R	保留位 读写值都为“1”。
4 ~ 0	K24[4:0]	00101	R/W	Y/C 分离 2 维滤波器的选择系数 如果增大此值, 垂直 BPF 的适用范围就会变大。

(1) Y/C 分离 2 维滤波器的选择系数控制

通过 K23B 位控制 Y/C 分离 2 维滤波器的选择系数。
详细内容请参照“30.5.5(3) 水平和垂直的相关检测”。

(2) Y/C 分离 2 维滤波器的选择系数控制

通过 K23A 位控制 Y/C 分离 2 维滤波器的选择系数。
详细内容请参照“30.5.5(3) 水平和垂直的相关检测”。

(3) Y/C 分离 2 维滤波器的选择系数控制

通过 K24 位控制 Y/C 分离 2 维滤波器的选择系数。
详细内容请参照“30.5.5(3) 水平和垂直的相关检测”。

30.4.57 Y/C 分离控制寄存器 (8) (YCSCR8)

在 2 维 Y/C 分离中, 适当地切换水平 BPF、垂直 BPF 和水平垂直 BPF。有关水平 BPF 和水平垂直 BPF, 能选择水平方向的特性。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HBPF_NARROW	HVBPF_NARROW	HBPF1_9TAP_ON	HVBPF1_9TAP_ON	HFIL_TAP_SEL	—	—	—	—	—	—	—	—	—	—	—
初始值:	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15	HBPF_NARROW	1	R/W	水平 BPF (后段) 选择 0: 旁路 1: 17TAP
14	HVBPF_NARROW	1	R/W	水平垂直 BPF (后段) 选择 0: 旁路 1: 17TAP
13	HBPF1_9TAP_ON	0	R/W	水平 BPF (前段) 选择 0: 17TAP 1: 9TAP
12	HVBPF1_9TAP_ON	0	R/W	水平垂直 (前段) 的 BPF 选择 0: 17TAP 1: 9TAP
11	HFIL_TAP_SEL	0	R/W	水平滤波器和水平垂直滤波器的频带切换信号 0: 17TAP 1: 9TAP
10 ~ 0	—	全 0	R	保留位 读写值都为 “0”。

(1) 水平 BPF 的选择控制

通过 HBPF_NARROW 位控制水平 BPF (后段) 的选择。
详细内容请参照 “30.5.5(2) 水平和垂直滤波器”。

(2) 水平垂直 BPF 的选择控制

通过 HVBPF_NARROW 位控制水平垂直 BPF (后段) 的选择。
详细内容请参照 “30.5.5(2) 水平和垂直滤波器”。

(3) 水平 BPF (宽带) 的选择控制

通过 HBPF1_9TAP_ON 位控制水平 BPF (前段) 的选择。
详细内容请参照 “30.5.5(2) 水平和垂直滤波器”。

(4) 水平垂直 BPF (宽带) 的选择控制

通过 HVBPF1_9TAP_ON 位控制水平垂直 BPF (前段) 的选择。
详细内容请参照 “30.5.5(2) 水平和垂直滤波器”。

(5) 水平 BPF 的频带切换控制

通过 HFIL_TAP_SEL 位控制混合水平 BPF 的频带切换。
详细内容请参照 “30.5.5(5) 水平和垂直的混合”。

30.4.58 Y/C 分离控制寄存器 (9) (YCSCR9)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DET2_ON	—	—	—	HSEL_MIX_Y[3:0]			VSEL_MIX_Y[3:0]			HVSEL_MIX_Y[3:0]					
初始值:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	DET2_ON	1	R/W	2 维滤波器的混合处理选择 0: 不混合通过相关检测滤波器的信号 1: 混合通过相关检测滤波器的信号
14 ~ 12	—	全 0	R	保留位 读写值都为“0”。
11 ~ 8	HSEL_MIX_Y [3:0]	0000	R/W	通过水平滤波器和水平滤波器（前段）的信号的混合比 0: 水平滤波器 100.0% 1: 水平滤波器 87.5%+ 水平滤波器（前段） 12.5% 2: 水平滤波器 75.0%+ 水平滤波器（前段） 25.0% 3: 水平滤波器 62.5%+ 水平滤波器（前段） 37.5% 4: 水平滤波器 50.0%+ 水平滤波器（前段） 50.0% 5: 水平滤波器 37.5%+ 水平滤波器（前段） 62.5% 6: 水平滤波器 25.0%+ 水平滤波器（前段） 75.0% 7: 水平滤波器 12.5%+ 水平滤波器（前段） 87.5% 8: 水平滤波器（前段） 100.0% 9 ~ 15: 禁止设定
7 ~ 4	VSEL_MIX_Y [3:0]	0000	R/W	通过垂直滤波器和水平垂直滤波器（前段）的信号的混合比 0: 垂直滤波器 100.0% 1: 垂直滤波器 87.5%+ 水平垂直滤波器（前段） 12.5% 2: 垂直滤波器 75.0%+ 水平垂直滤波器（前段） 25.0% 3: 垂直滤波器 62.5%+ 水平垂直滤波器（前段） 37.5% 4: 垂直滤波器 50.0%+ 水平垂直滤波器（前段） 50.0% 5: 垂直滤波器 37.5%+ 水平垂直滤波器（前段） 62.5% 6: 垂直滤波器 25.0%+ 水平垂直滤波器（前段） 75.0% 7: 垂直滤波器 12.5%+ 水平垂直滤波器（前段） 87.5% 8: 水平垂直滤波器（前段） 100.0% 9 ~ 15: 禁止设定
3 ~ 0	HVSEL_MIX_Y [3:0]	0	R/W	通过水平垂直滤波器和水平垂直滤波器（前段）的信号的混合比 0: 水平垂直滤波器 100.0% 1: 水平垂直滤波器 87.5%+ 水平垂直滤波器（前段） 12.5% 2: 水平垂直滤波器 75.0%+ 水平垂直滤波器（前段） 25.0% 3: 水平垂直滤波器 62.5%+ 水平垂直滤波器（前段） 37.5% 4: 水平垂直滤波器 50.0%+ 水平垂直滤波器（前段） 50.0% 5: 水平垂直滤波器 37.5%+ 水平垂直滤波器（前段） 62.5% 6: 水平垂直滤波器 25.0%+ 水平垂直滤波器（前段） 75.0% 7: 水平垂直滤波器 12.5%+ 水平垂直滤波器（前段） 87.5% 8: 水平垂直滤波器（前段） 100.0% 9 ~ 15: 禁止设定

(1) 2 维滤波器的混合处理选择控制

通过 DET2_ON 位处理 2 维滤波器的混合处理选择。
详细内容请参照“30.5.5(5) 水平和垂直的混合”。

(2) 通过水平滤波器和水平滤波器（前段）的信号的混合比控制

通过 HSEL_MIX_Y 位控制通过水平滤波器和水平滤波器（前段）的信号的混合比。
详细内容请参照“30.5.5(5) 水平和垂直的混合”。

(3) 通过垂直滤波器和水平垂直滤波器（前段）的信号的混合比控制

通过 VSEL_MIX_Y 位控制通过垂直滤波器和水平垂直滤波器（前段）的信号的混合比。
详细内容请参照“30.5.5(5) 水平和垂直的混合”。

(4) 通过水平垂直滤波器和水平垂直滤波器（前段）的信号的混合比控制

通过 HVSEL_MIX_Y 位控制通过水平垂直滤波器和水平垂直滤波器（前段）的信号的混合比。
详细内容请参照“30.5.5(5) 水平和垂直的混合”。

30.4.59 Y/C 分离控制寄存器（11）（YCSCR11）

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	V_Y_LEVEL[8:0]								
初始值:	1	1	0	1	1	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15、14	—	全 1	R	保留位 读写值都为“1”。
13	—	0	R	保留位 读写值都为“0”。
12、11	—	全 1	R	保留位 读写值都为“1”。
10、9	—	全 0	R	保留位 读写值都为“0”。
8 ~ 0	V_Y_LEVEL[8:0]	0000000 11	R/W	相关检测滤波器的垂直亮度检测电平 通过小于等于此值判断亮度。

(1) 相关检测滤波器的垂直亮度检测电平

通过 V_Y_LEVEL[8:0] 位控制相关检测滤波器的垂直亮度检测电平。
在使用此模块时，必须将这些位全部置“0”。

30.4.60 Y/C 分离控制寄存器 (12) (YCSCR12)

在 2 维 Y/C 分离运行中，能在切换水平 BPF、垂直 BPF 和水平垂直 BPF 后，通过级联水平 BPF 将水平方向的频带设定得更窄。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DET2_MIX_C[3:0]				DET2_MIX_Y[3:0]				—	—	—	—	FIL2_MODE_2D [1:0]	—	FIL2_NARROW_2D	
初始值:	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R	R/W

位	位名	初始值	R/W	说明
15 ~ 12	DET2_MIX_C [3:0]	0000	R/W	通过水平垂直适应滤波器的 C 信号和通过相关检测滤波器的信号的混合比（当 DET2_ON 位为“0”时，将这些位置“0”） 0: 水平垂直适应滤波器 100.0% 1: 水平垂直适应滤波器 87.5%+ 相关检测滤波器 12.5% 2: 水平垂直适应滤波器 75.0%+ 相关检测滤波器 25.0% 3: 水平垂直适应滤波器 62.5%+ 相关检测滤波器 37.5% 4: 水平垂直适应滤波器 50.0%+ 相关检测滤波器 50.0% 5: 水平垂直适应滤波器 37.5%+ 相关检测滤波器 62.5% 6: 水平垂直适应滤波器 25.0%+ 相关检测滤波器 75.0% 7: 水平垂直适应滤波器 12.5%+ 相关检测滤波器 87.5% 8: 相关检测滤波器 100.0% 9 ~ 15: 禁止设定
11 ~ 8	DET2_MIX_Y [3:0]	0110	R/W	通过水平垂直适应滤波器的用于生成 Y 信号的 C 信号和通过相关检测滤波器的信号的混合比（当 DET2_ON 位为“0”时，将这些位置“0”） 0: 水平垂直适应滤波器 100.0% 1: 水平垂直适应滤波器 87.5%+ 相关检测滤波器 12.5% 2: 水平垂直适应滤波器 75.0%+ 相关检测滤波器 25.0% 3: 水平垂直适应滤波器 62.5%+ 相关检测滤波器 37.5% 4: 水平垂直适应滤波器 50.0%+ 相关检测滤波器 50.0% 5: 水平垂直适应滤波器 37.5%+ 相关检测滤波器 62.5% 6: 水平垂直适应滤波器 25.0%+ 相关检测滤波器 75.0% 7: 水平垂直适应滤波器 12.5%+ 相关检测滤波器 87.5% 8: 相关检测滤波器 100.0% 9 ~ 15: 禁止设定
7 ~ 4	—	全 0	R	保留位 读写值都为“0”。
3、2	FIL2_MODE_2D [1:0]	01	R/W	2 维级联 /TAKE-OFF 滤波器的模式选择 0: 旁路 1: 选择级联滤波器 2: 选择 TAKE-OFF 滤波器 3: 禁止设定
1	—	0	R	保留位 读写值都为“0”。
0	FIL2_NARROW_2D	1	R/W	2 维级联滤波器的选择 0: 旁路 1: 17TAP

(1) 通过水平垂直适应滤波器的 C 信号和通过相关检测滤波器的信号的混合比控制

通过 DET2_MIX_C 位控制彩色信号（适应处理结果）和通过相关检测滤波器的信号的混合比。
详细内容请参照“30.5.5(6) 相关检测值的混合”。

(2) 通过水平垂直适应滤波器的用于生成 Y 信号的 C 信号和通过相关检测滤波器的信号的混合比控制

详细内容请参照“30.5.5(6) 相关检测值的混合”。

(3) 2 维级联 /TAKE-OFF 滤波器的模式选择控制

通过 FIL2_MODE_2D 位控制 2 维级联 /TAKE-OFF 滤波器的模式选择。
详细内容请参照“30.5.5(8) 级联滤波器”。

(4) 2 维级联滤波器的选择控制

通过 FIL2_NARROW_2D 位控制 2 维级联滤波器的选择。
详细内容请参照“30.5.5(8) 级联滤波器”。

30.4.61 数字钳位控制寄存器 (9) (DCPCR9)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	CLP_HOLD_ON_Y	CLP_HOLD_ON_CB	CLP_HOLD_ON_CR	—	—	—	—	—	—	—	—	—	—
初始值:	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 13	—	全 1	R	保留位 读写值都为“1”
12	CLP_HOLD_ON_Y	1	R/W	钳位数据的 HOLD 处理 ON/OFF (Y) 0: 无 HOLD 处理 1: 有 HOLD 处理
11	CLP_HOLD_ON_CB	1	R/W	钳位数据的 HOLD 处理 ON/OFF (Cb) 0: 无 HOLD 处理 1: 有 HOLD 处理
10	CLP_HOLD_ON_CR	1	R/W	钳位数据的 HOLD 处理 ON/OFF (Cr) 0: 无 HOLD 处理 1: 有 HOLD 处理
9 ~ 0	—	全 0	R	保留位 读写值都为“0”。

(1) 钳位数据的 HOLD 处理 ON/OFF (Y) 控制

通过 CLP_HOLD_ON_Y 位控制 Y 信号钳位数据的 HOLD 处理 ON/OFF。
在使用此模块时，必须将此位置“0”。

(2) 钳位数据的 HOLD 处理 ON/OFF (Cb) 控制

通过 CLP_HOLD_ON_CB 位控制 Cb 信号钳位数据的 HOLD 处理 ON/OFF。
在使用此模块时，必须将此位置“0”。

(3) 钳位数据的 HOLD 处理 ON/OFF (Cr) 控制

通过 CLP_HOLD_ON_CR 位 Cr 信号控制钳位数据的 HOLD 处理 ON/OFF。
在使用此模块时，必须将此位置“0”。

30.4.62 Y/C 分离的色度信号滤波器 TAP 系数 (WA_F0 ~ F8) 寄存器 (YCTWA_F0 ~ F8)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	FIL2_2D_WA_F0~F8[12:0]												
初始值:	0	0	0	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12 ~ 0	FIL2_2D_WA_F0 ~ F8[12:0]	*	R/W	2 维级联宽带 (3.58/4.43/SECAM-DR) /TAKE-OFF 滤波器 TAP 系数 0 ~ 8 [12]: 符号 [11:0]: 绝对值 * 初始值 FIL2_2D_WA_F0: H'0018 FIL2_2D_WA_F1: H'002C FIL2_2D_WA_F2: H'0014 FIL2_2D_WA_F3: H'1034 FIL2_2D_WA_F4: H'1080 FIL2_2D_WA_F5: H'1080 FIL2_2D_WA_F6: H'100C FIL2_2D_WA_F7: H'0084 FIL2_2D_WA_F8: H'00C8

(1) 2 维级联宽带 (3.58/4.43/SECAM-DR) /TAKE-OFF 滤波器的 TAP 系数 0 ~ 8 控制

通过 FIL2_2D_WA_F0 位控制 2 维级联宽带 (3.58/4.43/SECAM-DR) /TAKE-OFF 滤波器 TAP 系数 0 ~ 8。

传输函数为:

$$H(z) = \{F0(z^{-8}+z^8)+F1(z^{-7}+z^7)+F2(z^{-6}+z^6)+F3(z^{-5}+z^5)+F4(z^{-4}+z^4) \\ +F5(z^{-3}+z^3)+F6(z^{-2}+z^2)+F7(z^{-1}+z^1)+F8(z^0)\} \div 1024$$

对于系数值，将最高位设定为符号位，用绝对值设定有效值。

表 30.31 TAP 系数的设定值

最高位	最高位以外的位	设定值
0	0 ~ 4095	+0 ~ +4095
1	0 ~ 4095	-0 ~ -4095

各滤波器的推荐设定值请参照“30.5.5(8) 级联滤波器”。

30.4.63 Y/C 分离的色度滤信号波器 TAP 系数 (WB_F0 ~ F8) 寄存器 (YCTWB_F0 ~ F8)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	FIL2_2D_WB_F0~F8[12:0]												
初始值:	0	0	0	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12 ~ 0	FIL2_2D_WB_F0 ~ F8[12:0]	*	R/W	2 维级联宽带 (SECAM-DB) 滤波器 TAP 系数 0 ~ 8 [12]: 符号 [11:0]: 绝对值 * 初始值 FIL2_2D_WB_F0: H'100C FIL2_2D_WB_F1: H'0028 FIL2_2D_WB_F2: H'003C FIL2_2D_WB_F3: H'000C FIL2_2D_WB_F4: H'1068 FIL2_2D_WB_F5: H'109C FIL2_2D_WB_F6: H'1040 FIL2_2D_WB_F7: H'0078 FIL2_2D_WB_F8: H'00D0

(1) 2 维级联宽带 (SECAM-DB) 滤波器的 TAP 系数 0 ~ 8 控制

通过 FIL2_2D_WB_F0 位控制 2 维级联宽带 (SECAM-DB) 滤波器 TAP 系数 0 ~ 8。

传输函数为:

$$H(z) = \{F0(z^{-8}+z^8)+F1(z^{-7}+z^7)+F2(z^{-6}+z^6)+F3(z^{-5}+z^5)+F4(z^{-4}+z^4) \\ +F5(z^{-3}+z^3)+F6(z^{-2}+z^2)+F7(z^{-1}+z^1)+F8(z^0)\} \div 1024$$

对于系数值, 将最高位设定为符号位, 用绝对值设定有效值。

表 30.32 TAP 系数的设定值

最高位	最高位以外的位	设定值
0	0 ~ 4095	+0 ~ +4095
1	0 ~ 4095	-0 ~ -4095

各滤波器的推荐设定值请参照“30.5.5(8) 级联滤波器”。

30.4.64 Y/C 分离的色度信号滤波器 TAP 系数 (NA_F0 ~ F8) 寄存器 (YCTNA_F0 ~ F8)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	FIL2_2D_NA_F0~F8[12:0]												
初始值:	0	0	0	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12 ~ 0	FIL2_2D_NA_F0 ~ F8[12:0]	*	R/W	2 维级联窄带 (3.58/4.43/SECAM-DR) 滤波器 TAP 系数 0 ~ 8 [12]: 符号 [11:0]: 绝对值 * 初始值 FIL2_2D_NA_F0: H'0018 FIL2_2D_NA_F1: H'002C FIL2_2D_NA_F2: H'0014 FIL2_2D_NA_F3: H'1034 FIL2_2D_NA_F4: H'1080 FIL2_2D_NA_F5: H'1080 FIL2_2D_NA_F6: H'100C FIL2_2D_NA_F7: H'0084 FIL2_2D_NA_F8: H'00C8

(1) 2 维级联窄带 (3.58/4.43/SECAM-DR) 滤波器的 TAP 系数 0 ~ 8 控制

通过 FIL2_2D_NA_F0 位控制 2 维级联窄带 (3.58/4.43/SECAM-DR) 滤波器 TAP 系数 0 ~ 8。

传输函数为:

$$H(z) = \{F0(z^{-8}+z^8)+F1(z^{-7}+z^7)+F2(z^{-6}+z^6)+F3(z^{-5}+z^5)+F4(z^{-4}+z^4) \\ +F5(z^{-3}+z^3)+F6(z^{-2}+z^2)+F7(z^{-1}+z^1)+F8(z^0)\} \div 1024$$

对于系数值, 将最高位设定为符号位, 用绝对值设定有效值。

表 30.33 TAP 系数的设定值

最高位	最高位以外的位	设定值
0	0 ~ 4095	+0 ~ +4095
1	0 ~ 4095	-0 ~ -4095

各滤波器的推荐设定值请参照“30.5.5(8) 级联滤波器”。

30.4.65 Y/C 分离的色度信号滤波器 TAP 系数 (NB_F0 ~ F8) 寄存器 (YCTNB_F0 ~ F8)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	FIL2_2D_NB_F0~F8[12:0]												
初始值:	0	0	0	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12 ~ 0	FIL2_2D_NB_F0 ~ F8[12:0]	*	R/W	2 维级联窄带 (SECAM-DB) 滤波器 TAP 系数 0 ~ 8 [12]: 符号 [11:0]: 绝对值 * 初始值 FIL2_2D_NB_F0: H'1438 FIL2_2D_NB_F1: H'0AF0 FIL2_2D_NB_F2: H'1CEC FIL2_2D_NB_F3: H'065C FIL2_2D_NB_F4: H'05A4 FIL2_2D_NB_F5: H'1CEC FIL2_2D_NB_F6: H'085C FIL2_2D_NB_F7: H'0178 FIL2_2D_NB_F8: H'1568

(1) 2 维级联窄带 (SECAM-DB) 滤波器的 TAP 系数 0 ~ 8 控制

通过 FIL2_2D_NB_F0 位控制 2 维级联窄带 (SECAM-DB) 滤波器 TAP 系数 0 ~ 8。

传输函数为:

$$H(z) = \{F0(z^{-8}+z^8)+F1(z^{-7}+z^7)+F2(z^{-6}+z^6)+F3(z^{-5}+z^5)+F4(z^{-4}+z^4) \\ +F5(z^{-3}+z^3)+F6(z^{-2}+z^2)+F7(z^{-1}+z^1)+F8(z^0)\} \div 1024$$

对于系数值，将最高位设定为符号位，用绝对值设定有效值。

表 30.34 TAP 系数的设定值

最高位	最高位以外的位	设定值
0	0 ~ 4095	+0 ~ +4095
1	0 ~ 4095	-0 ~ -4095

各滤波器的推荐设定值请参照“30.5.5(8) 级联滤波器”

30.4.66 亮度 (Y) 信号增益调整控制寄存器 (YGAINCR)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	Y_GAIN2[9:0]									
初始值:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	Y_GAIN2[9:0]	100000 0000	R/W	Y 信号的增益系数 (0=0[倍], 512=1.0[倍], 1023≈2.0[倍])

(1) Y 信号的输出增益控制

通过 Y_GAIN2 位控制 Y 信号的输出增益。

$$Y \text{ 信号输出} = \text{解码后的 Y 信号} \times (Y_GAIN2 \div 512)$$

30.4.67 色差 (Cb) 信号增益调整控制寄存器 (CBGAINCR)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CB_GAIN2[9:0]									
初始值:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	CB_GAIN2[9:0]	100000 0000	R/W	Cb 信号的增益系数 (0=0[倍], 512=1.0[倍], 1023≈2.0[倍])

(1) Cb 信号的输出增益控制

通过 CB_GAIN2 位控制 Cb 信号的输出增益。

$$Cb \text{ 信号输出} = \text{解码后的 Cb 信号} \times (CB_GAIN2 \div 512)$$

30.4.68 色差 (Cr) 信号增益调整控制寄存器 (CRGAINCR)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CR_GAIN2[9:0]									
初始值:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	CR_GAIN2[9:0]	100000 0000	R/W	Cr 信号的增益系数 (0=0[倍], 512=1.0[倍], 1023≈2.0[倍])

(1) Cr 信号的输出增益控制

通过 CR_GAIN2 位控制 Cr 信号的输出增益。

$$\text{Cr 信号输出} = \text{解码后的 Cr 信号} \times (\text{CR_GAIN2} \div 512)$$

30.4.69 PGA 部寄存器更新 (PGA_UPDATE)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PGA_VEN
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
15 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	PGA_VEN	1	R/W	PGACR 寄存器的 V 更新允许 1: 允许 0: 禁止

(1) PGACR 寄存器的 V 更新允许

通过 PGA_VEN 位控制 PGACR 寄存器的 V 更新允许。

30.4.70 PGA 控制寄存器 (PGACR)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PGA_GAIN_SEL	PGA_GAIN[4:0]				—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13	PGA_GAIN_SEL	0	R/W	PGA 切换设定 0: 自动 (AGC) 1: 手动 (参照 PGA_GAIN)
12 ~ 8	PGA_GAIN[4:0]	01000	R/W	PGA 增益设定 0(0.8Vpp) ~ 31(1.6Vpp)
7 ~ 4	—	全 0	R	保留位 读写值都为“0”。
3	—	1	R	保留位 读写值都为“1”。
2 ~ 0	—	全 0	R	保留位 读写值都为“0”。

【注】 在垂直同步信号有效并且全部 PGA 部寄存器更新 (PGA_UPDATE) 的 PGA_VEN 位为“1”的状态下, 更新此寄存器。

(1) PGA 切换设定

当 PGA_GAIN_SEL 位为“0”并且 ADCCR1.AGCMODE 位为“1”时, 将 AGC 的控制值反映到 PGA 的增益设定。当 PGA_GAIN_SEL 位为“1”时, 将 PGA_GAIN 的值反映到 PGA 的增益设定。此时, ADCCR1.AGCMODE 位无效。

禁止将 ADCCR1.AGCMODE 位和 PGA_GAIN_SEL 位置“0”。

(2) PGA 增益设定

当 PGA_GAIN_SEL 位为“1”时, 将 PGA_GAIN 的值反映到 PGA 的增益设定。本 LSI 的 PGA 能设定 32 个级别。

30.4.71 ADC 控制寄存器 2 (ADCCR2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADC_VINSEL
初始值:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
15 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	—	1	R	保留位 读写值都为“1”。
7 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	ADC_VINSEL	0	R/W	输入引脚控制 0: 输入 VIN1 1: 输入 VIN2

(1) 输入引脚控制

通过 ADC_VINSEL 位选择复合图像信号的输入引脚。

30.5 运行说明

30.5.1 概要

此模块将复合图像信号（CVBS）进行解码并且分离为水平 / 垂直同步信号、亮度（Y）信号和色差（Cb/Cr）信号，支持 NTSC、PAL 和 SECAM 的颜色体系。此模块由用于输入图像信号的 A/D 转换器、同步分离电路、Burst Controlled Oscillator（BCO）、Y/C 分离电路、色度解码电路、数字钳位电路和输出调整电路构成，框图如图 30.25 所示。

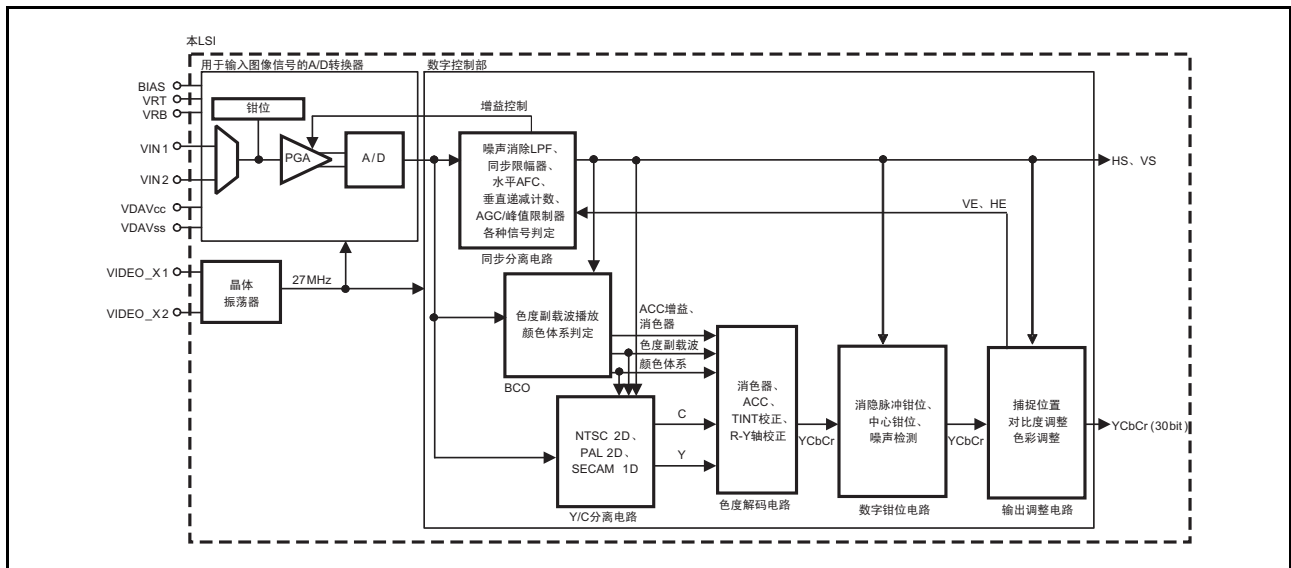


图 30.25 整体框图

(1) 用于输入图像信号的 A/D 转换器

在通过同步端钳位和 Programmable Gain Amp（PGA）处理复合图像信号（CVBS）后进行 A/D 转换。复合图像信号可选择 VIN1 引脚和 VIN2 引脚。

(2) 同步分离电路

从复合图像信号中抽出水平同步信号和垂直同步信号。具有通过检测同步信号的振幅来自动进行 PGA 增益调整的 Auto Gain Control（AGC）功能。

(3) Burst Controlled Oscillator（BCO）

从复合图像信号中抽出色同步信号，重新生成彩色解调所需的色度副载波信号，并且能通过从色同步信号得到的相位和频率信息，检测颜色体系。

(4) Y/C 分离电路

从 NTSC、PAL 和 SECAM 格式的复合图像信号中分离 Y 信号和 C 信号。NTSC 和 PAL 为适应 2 维的分离，SECAM 为 1 维分离。

(5) 色度解码

将通过“Y/C 分离电路”抽出的 C 信号解调为 Cb/Cr 信号。具有通过检测色同步信号的振幅来自动进行色彩增益调整的 Auto Color Control（ACC）功能以及对解调时的色调进行调整的 TINT/R-Y 轴校正等功能。

(6) 数字钳位电路

在任意的钳位位置对 Y 信号进行消隐脉冲钳位，对 Cb/Cr 信号进行中心钳位。具有通过自相关函数检测噪声量的功能。

(7) 输出调整电路

此电路设定捕捉位置并且进行对比度和色彩的调整。

30.5.2 用于输入图像信号的 A/D 转换器

在通过同步端钳位和 Programmable Gain Amp (PGA) 处理复合图像信号 (CVBS) 后进行 A/D 转换。用于输入图像信号的 A/D 转换器的框图如图 30.26 所示。

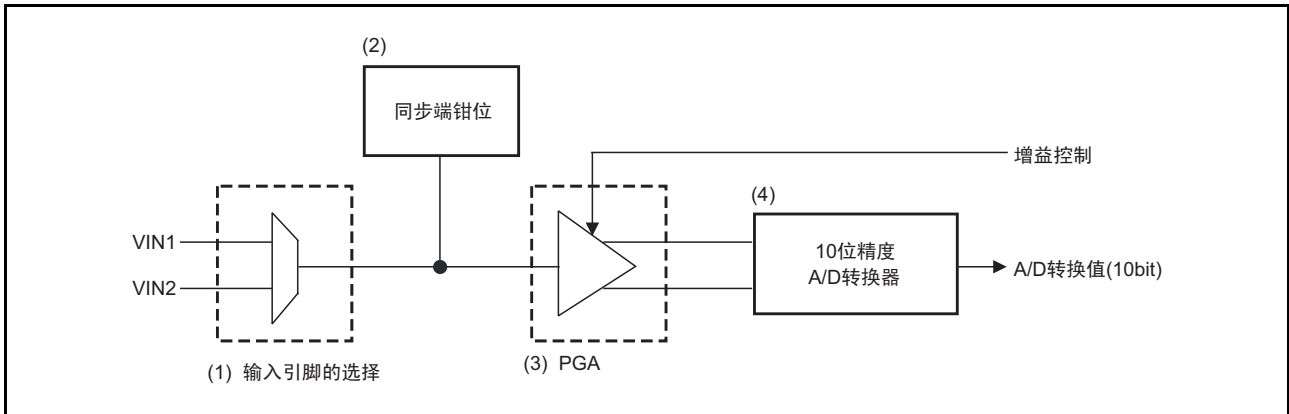


图 30.26 用于输入图像信号的 A/D 转换器的框图

将图像信号进行 A/D 转换前的图像波形如图 30.27 所示。

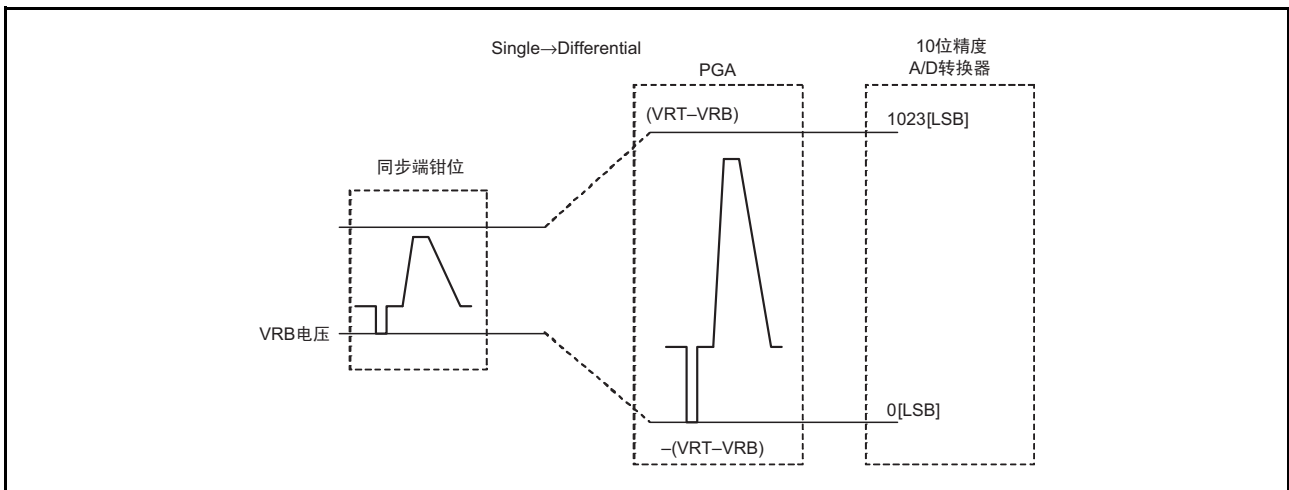


图 30.27 A/D 转换图像图

(1) 输入引脚选择

能通过 ADCCR2.ADC_VINSEL 位选择 VIN1 引脚和 VIN2 引脚为输入引脚。

(2) 同步端钳位

将同步端电平钳位为 VRB 电压 ($\approx 1.0V$)。

(3) Programmable Gain Amp (PGA)

调整增益, 使图像输入信号的电压 ($\approx 0.8V_{pp} \sim 1.6V_{pp}$) 成为 A/D 转换器的输入电压电平 ($\approx 2.0V_{pp}$)。增益能设定 32 个级别, 每增加 1 级, 就增大 TYP: 0.2[dB]。最小为 TYP: 1.835[dB], 最大为 TYP: 8.023[dB]。

PGA 的增益设定如表 30.35 所示。

表 30.35 PGA 的增益设定表 (dB 显示)

PGA 的增益设定	输入范围 [Vpp]	增益 (dB)
0	1.619	1.835
1	1.585	2.021
2	1.551	2.206
3	1.519	2.392
4	1.486	2.579
5	1.455	2.766
6	1.423	2.954
7	1.393	3.142
8	1.363	3.332
9	1.333	3.522
·	·	·
·	·	·
·	·	·
27	0.879	7.143
28	0.857	7.360
29	0.836	7.578
30	0.815	7.799
31	0.794	8.023

能通过将 PGACR.PGA_GAIN_SEL 位置 “1” 并且通过 PGACR.PGA_GAIN[4:0] 来设定 PGA 的增益。在 AGC 运行 (ADCCR1.AGCMODE 位为 “1”) 的情况下, 自动进行设定。

(4) 10 位精度的 A/D 转换器

将由 PGA 进行增益调整后的图像信号进行 A/D 转换。分辨率为 10 位精度, 采样频率为 VIDEO_X1/X2 输入时钟的 27MHz。

30.5.3 同步分离电路

从复合图像信号中抽出水平同步信号和垂直同步信号。具有通过检测同步信号的振幅来自动进行 PGA 增益调整的 Auto Gain Control (AGC) 功能。

同步分离电路的框图如图 30.28 所示。

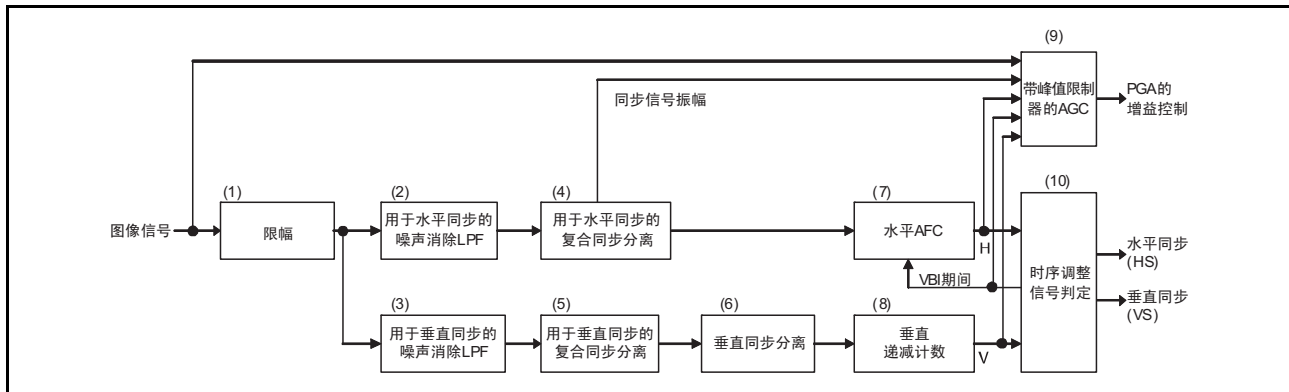


图 30.28 同步分离电路的框图

(1) 限幅

为了防止图像信号的振幅依赖性，对图像的高频成分进行限幅。能通过 `SYNSCR3.SSCLIPSEL[3:0]` 位设定限幅电平。

将图像限幅电平设定在复合同步信号成分不恶化（能检测）的范围内。

(2) 用于水平同步的噪声消除低通滤波器（LPF）

在分离复合同步信号前，消除叠加在图像信号中的噪声。能分别设定为用于水平同步和用于垂直同步的低通滤波器，并且能通过 `SYNSCR1.LPFHSYNC[2:0]` 位设定用于水平同步的低通滤波器截止频率。将截止频率设定在复合同步信号成分不恶化（能检测）的范围内。

(3) 用于垂直同步的噪声消除低通滤波器（LPF）

在分离复合同步信号前，消除叠加在图像信号中的噪声。能分别设定为用于水平同步和用于垂直同步的低通滤波器，并且能通过 `SYNSCR1.LPFVSYNC[2:0]` 位设定用于垂直同步的低通滤波器截止频率。将截止频率设定在复合同步信号成分不恶化（能检测）的范围内。

(4) 用于水平同步的复合同步分离

根据限幅电平，从图像信号分离复合同步信号。能分别设定用于水平同步和用于垂直同步的复合同步分离限幅电平，并且能通过 `SYNSCR1.SLICERMODE_H[1:0]` 位选择自动或者手动设定限幅电平。

如果选择自动设定，就根据 `SYNSCR1.SLICERMODE_H[1:0]` 位、`SYNSCR2.SYNCMAXDUTY_H[5:0]` 位和 `SYNSCR2.SYNCMINDUTY_H[5:0]` 位的设定，自动控制限幅电平。能通过 `SYNSCR1.VELOCITYSHIFT_H[3:0]` 位设定限幅电平的判定速度。

如果选择手动设定，`SYNSCR3.CSYNCSLICE_H[9:0]` 位的值就为限幅电平。

(5) 用于垂直同步的复合同步分离

根据限幅电平，从图像信号分离复合同步信号。能分别设定为用于水平同步和用于垂直同步的复合同步分离限幅电平，并且能通过 `SYNSCR1.SLICERMODE_V[1:0]` 位选择自动或者手动设定限幅电平。

如果选择自动设定，就根据 `SYNSCR1.SLICERMODE_V[1:0]` 位、`SYNSCR4.SYNCMAXDUTY_V[5:0]` 位和 `SYNSCR4.SYNCMINDUTY_V[5:0]` 位的设定，自动控制限幅电平。

如果选择手动设定，`SYNSCR5.CSYNCSLICE_V[9:0]` 位的值就为限幅电平。

(6) 垂直同步分离

从通过 (5) 分离的复合同步信号抽出垂直同步信号。能通过 `SYNSCR5.VSYNCSLICE[4:0]` 位设定用于分离垂直同步信号的阈值，根据各图像信号格式的锯齿脉冲信号宽度进行设定。

(7) 水平 Auto Frequency Control (AFC)

水平 Auto Frequency Control (AFC) 是用于从通过 (4) 分离的复合同步信号中抽出水平同步信号的数字 PLL。在消除假水平同步信号的同时内插欠缺的水平同步信号，生成稳定的水平同步信号。

能通过 `HAFCCR1.HAFCTYP[9:0]` 位、`HAFCCR2.HAFCMAX[9:0]` 位和 `HAFCCR3.HAFCCMIN[9:0]` 位设定水平 AFC 的中心频率和同步范围。当锁定水平 AFC 时，`VSYNCSR.FHLOCK` 位为“1”；当未锁定水平 AFC 时，`VSYNCSR.FHLOCK` 位为“0”。能通过 `{HSYNCSR.FHCOUNT[16:1]` 位、`VSYNCSR.FHCOUNT[0]` 位 } 确认水平 AFC 的振荡周期。

能通过 `HAFCCR1.HAFCCGAIN[3:0]` 位设定水平 AFC 的环路增益（响应速度）。响应速度越快锁定时间越短，但是由于噪声等原因，水平周期容易发生变动。能通过 `HAFCCR3.HAFCMODE[1]` 位、`AFCPFCR.PHDET_FIX` 位和 `PHDET_DIV[2:0]` 位降低低 S/N 时的环路增益，防止噪声引起的误动作。能通过 `VSYNCSR.ISNOISY` 位读到是或者不是低 S/N 状态。

能通过 `HAFCCR2.HAFCSTART[3:0]` 位、`HAFCCR3.HAFCEEND[3:0]` 位和 `HAFCCR3.HAFCMODE[0]` 位设定垂直回扫期间（VBI）的环路增益。通常用于避免 VTR 磁头切换部的误动作。

(8) 垂直递减计数

对于垂直递减计数，在从通过 (6) 分离的垂直同步信号中消除假同步信号的同时内插欠缺的垂直同步信号，生成稳定的垂直同步信号。

能通过 `VCDWCR1.VCDDEFAULT[1:0]` 位设定垂直递减计数的振荡周期。如果将 `VCDWCR1.VCDDEFAULT[1:0]` 位置“0”，就判别输入的垂直同步信号，然后自动设定振荡周期。将输入的垂直同步信号的判别结果反映到 `VSYNCSR.FVMODE` 位。如果置“1”，就为 50.00Hz 振荡模式。此时，为了避免意料之外的误动作，建议将 `VCDWCR1.NOVCD60` 位置“1”（60Hz 振荡 OFF）。如果置“2”或者“3”，就为 59.94/60.00Hz 振荡模式。此时，为了避免意料之外的误动作，建议将 `VCDWCR1.NOVCD50` 位置“1”（50Hz 振荡 OFF）。

能通过 `VCDWCR1.VCDWINDOW[5:0]` 位和 `VCDWCR1.VCDOFFSET[4:0]` 位设定垂直递减计数的同步范围。当锁定垂直递减计数时，`VSYNCSR.FVLOCK` 位为“1”；当未锁定垂直递减计数时，`VSYNCSR.FVLOCK` 位为“0”。

能通过 `VSYNCSR.FVCOUNT[7:0]` 位确认已输入的垂直同步信号周期。在检测不到垂直同步信号的输入时，将 `VSYNCSR.NOSIGNAL` 位置“1”。

(9) 带峰值限制器的 Auto Gain Control (AGC)

Auto Gain Control (AGC) 检测同步信号的振幅，并且根据目标值自动控制 PGA 的增益。AGC 在 `ADCCR1.AGCMODE` 位为“1”时运行。

- 根据同步振幅进行的增益控制

能通过 `AGCCR1.AGCLEVEL[8:0]` 位和 `AGCCR2.AGCPRECIS[5:0]` 位设定同步信号的振幅目标值。例如，当通过 10 位 A/D 转换器量化 NTSC 信号时，A/D 转换器全量程的同步信号振幅为 $1023[\text{LSB}] \times (40[\text{IRE}] \div 173[\text{IRE}]) = 236.53179[\text{LSB}]$

因此给 `AGCCR1.AGCLEVEL[8:0]` 位设定 236[LSB]。在目标值（`AGCCR1.AGCLEVEL[8:0]` 位） \pm `AGCCR2.AGCPRECIS[5:0]` 位的情况下，固定增益。能通过读 `AGCCR2.AGCCONVERGE` 位，判别增益是否稳定。能通过 `SYNCSR.SYNCDEPTH[9:0]` 位确认检测的同步信号振幅，并且能通过 `AGCCR1.AGCRESPONSE[2:0]` 位设定 AGC 的响应速度。如果加快响应速度，对输入信号的反应越快，但是容易受噪声的影响。

能通过 `AGCCR2.AGCGAIN[7:0]` 位确认当前设定的增益值，并且能通过以下表达式大概算出实际 PGA 的增益设定值：

PGA 的增益设定值 = $0.585776 \times (\text{AGCCR2.AGCGAIN}[7:0] - 49)$

例如，当 `AGCCR2.AGCGAIN[7:0]` 位为“64”（相当于 1 倍）时，

PGA 的增益设定值 = $0.585776 \times (64 - 49) \approx 8.78$

PGA的增益设定值为“8”或者“9”。PGA增益能设定0~31的值。

能通过 AGCCR1.DOREDUCE 位和 AGCCR1.NOREDUCE 位设定 VBI 期间的增益。能通过 SYNCSSR.ISREDUCED 位读 VBI 期间的同步振幅结果。

- 峰值限制器

峰值限制器在同步信号振幅值和图像信号振幅值的比发生偏差时运行。假设图像信号振幅和同步信号振幅的比小于预定的比，因为PGA的增益变小，所以增益调整后的图像信号也小于A/D转换器的全量程。相反，假设图像信号振幅和同步信号振幅的比大于预定的比，因为PGA的增益变高，所以增益调整后的图像信号也大于A/D转换器的全量程。

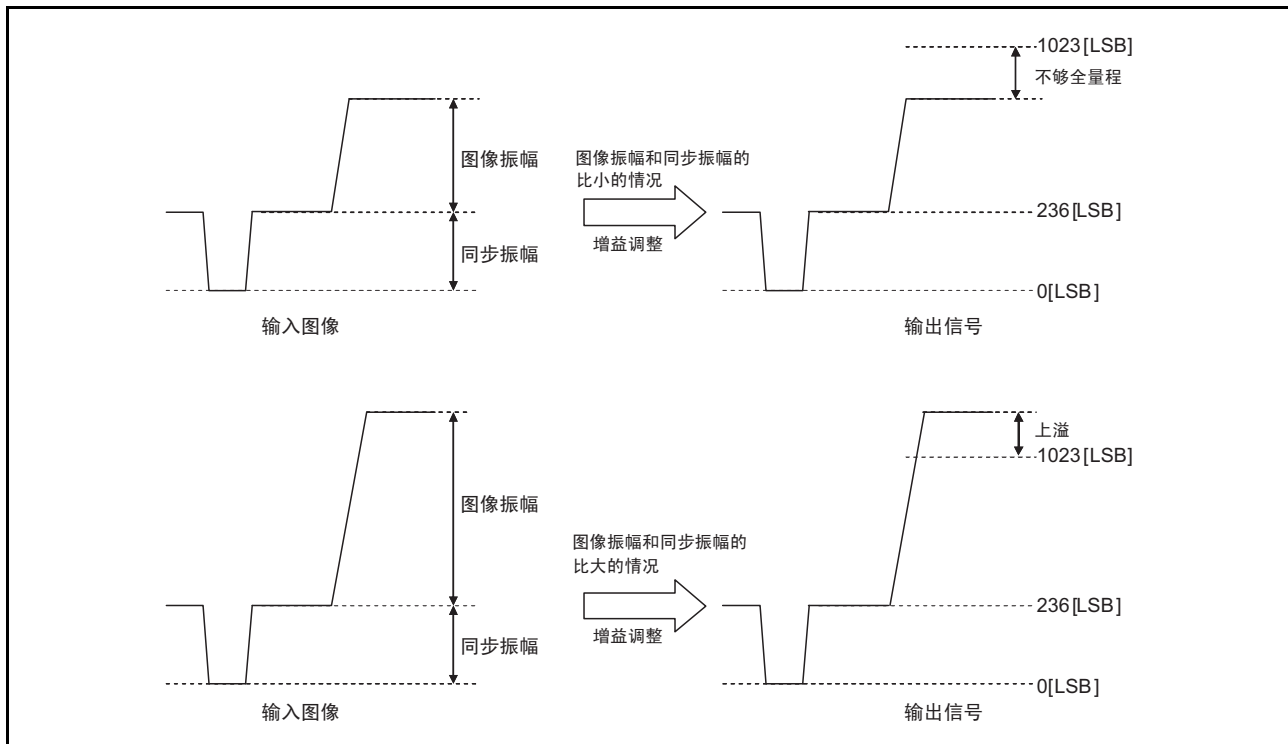


图 30.29 图像振幅和同步振幅的比发生偏差的情况

因此，对图像信号的峰值进行采样并且通过采样值调整PGA的增益。

能通过 PKLIMITCR.PEAKLEVEL[1:0] 位设定被限制的峰值。当采样到的图像信号峰值小于 PKLIMITCR.PEAKLEVEL[1:0] 位的设定值时，提高增益；当采样到的图像信号峰值大于 PKLIMITCR.PEAKLEVEL[1:0] 位的设定值并且超过 PKLIMITCR.MAXPEAKSAMPLES[7:0] 设定的容许最大数时，降低增益。

能通过 PKLIMITCR.PEAKATTACK[1:0] 位、PKLIMITCR.PEAKRELEASE[1:0] 位和 PKLIMITCR.PEAKRATIO[1:0] 位设定增益增加和减少时的响应速度和最大压缩率，并且能通过 AGCCSR1.HIGHSAMPLES[7:0] 位确认超过 PKLIMITCR.PEAKLEVEL[1:0] 位设定值的像素个数，通过 AGCCSR1.PEAKSAMPLES[7:0] 位确认发生上溢（超过 1023[LSB]）的像素个数。

- 手动设定

能通过将 PGACR.PGA_GAIN_SEL 位置“1”，手动设定PGA的增益设定值。设定值为 PGACR.PGA_GAIN[4:0] 位的值。当 PGACR.PGA_GAIN_SEL 位为“1”时，ADCCR1.AGCMODE 的设定无效；当 PGACR.PGA_GAIN_SEL 位为“0”（自动）时，禁止将 ADCCR1.AGCMODE 位置“0”（AGC OFF）。

(10) 时序调整和信号判定

调整通过 (7) 和 (8) 生成的水平同步信号和垂直同步信号的输出时序，能进行场判别并且通过 VSYNCSSR.INTERLACED 位确认隔行扫描或者逐行扫描。在场判别不稳定时，有时通过将 SYNCSSR5.VSYNCDELAY 位置“1”来加以改善。

30.5.4 Burst Controlled Oscillator (BCO)

从复合图像信号中抽出色同步信号，重新生成彩色解调所需的色度副载波信号，并且通过从色同步信号得到的相位和频率信息，检测颜色体系。

Burst Controlled Oscillator 的框图如图 30.30 所示。

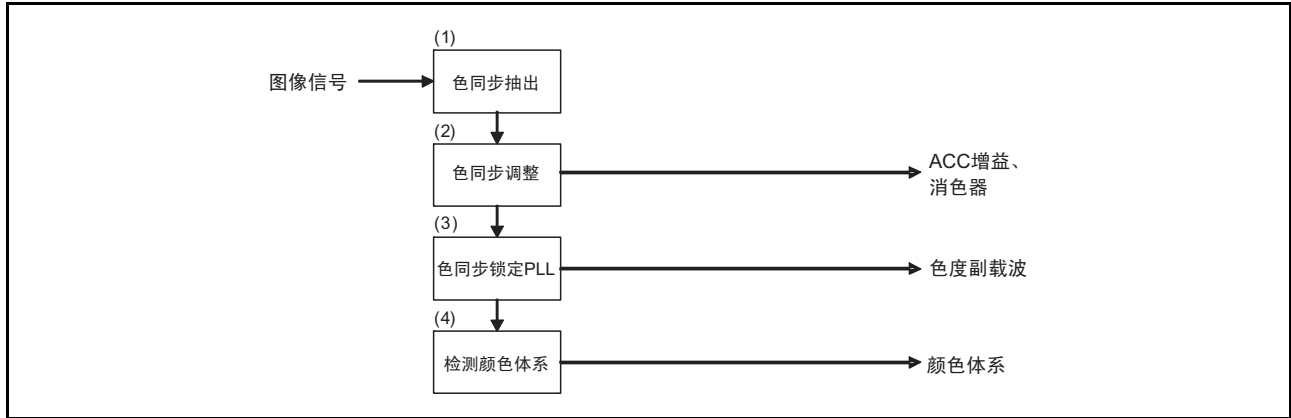


图 30.30 Burst Controlled Oscillator 的框图

(1) 色同步抽出

抽出色同步信号。能通过 BTGPCR.BGPWIDTH[6:0] 位和 BTGPCR.BGPSTART[7:0] 位对抽出色同步信号的位置进行调整，并且能通过 CROMASR1.NOBURST 位确认色同步信号的抽出结果。

(2) 色同步调整

对抽出的色同步信号振幅进行调整，详细内容请参照“30.5.6(1) Auto Color Control (ACC)”。根据输入的色同步信号振幅，输出消色器的 ON/OFF 信号，详细内容请参照“30.5.6(2) 消色器”。

(3) 色同步锁定 PLL

色同步锁定 PLL 是从调整后的色同步信号中重新生成色度副载波的数字 PLL。

能通过 BTLCR.LOCKRANGE[1:0] 位设定色同步锁定 PLL 的同步范围。当锁定时，CROMASR1.FSCLOCK 位为“1”；当未锁定时，CROMASR1.FSCLOCK 位为“0”。

能通过 BTLCR.LOOPGAIN[1:0] 位和 BTLCR.LOCKLIMIT[1:0] 位设定色同步锁定 PLL 的环路增益。如果加快响应速度和频率的搜索时间，锁定时间就会缩短，但是由于噪声等原因，容易发生变动或者容易解除锁定。

能通过 CROMASR2.LOCKLEVEL[7:0] 位确认色同步信号的 S/N。

(4) 颜色体系的判定

根据色同步锁定 PLL 的振荡频率和色同步信号的相位信息判定输入图像信号的颜色体系。能通过 BTLCR.NONTSC358 位、BTLCR.NONTSC443 位、BTLCR.NOPALM 位、BTLCR.NOPALN 位、BTLCR.NOPAL443 位和 BTLCR.NOSECAM 位设定颜色体系的判定，颜色体系的判别能选择全自动控制、手动控制以及只有部分颜色体系自动控制（不判别未指定的内容）。如果不属于任何颜色体系，就以 BTLCR.DEFAULTSYS[1:0] 位指定的颜色体系运行。

当检测到 NTSC、PAL 和 SECAM 信号时，分别从 CROMASR2.ISNTSC 位、CROMASR2.ISPAL 位和 CROMASR2.ISSECAM 位读“1”。能通过 CROMASR1.COLORSYS[1:0] 位确认当前正在运行的颜色体系。

能通过读 CROMASR1.FSCMODE 位确认色度副载波的频率。

30.5.5 Y/C 分离电路

从 NTSC、PAL 和 SECAM 格式的复合图像信号中分离 Y 信号和 C 信号。NTSC 和 PAL 为适应 2 维的分离，SECAM 为 1 维分离。Y/C 分离电路的框图如图 30.31 所示。

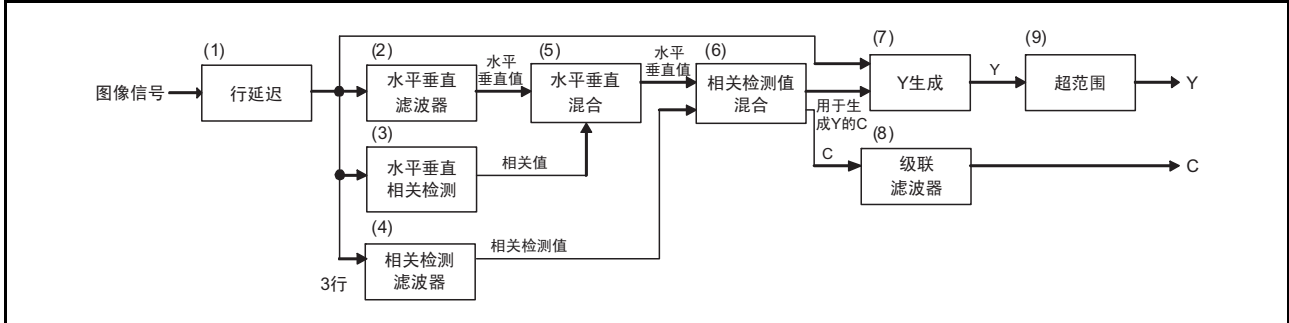


图 30.31 Y/C 分离电路的框图

各颜色格式的 Y/C 分离电路运行如表 30.36 所示。

表 30.36 Y/C 分离运行对应表

	Y/C 分离运行
NTSC-3.58	适应 2 维
NTSC-4.43	适应 2 维
PAL-M	适应 2 维
PAL-N	适应 2 维
PAL-4.43	适应 2 维
SECAM	1 维

(1) 行延迟

2 维 Y/C 分离需要 3 行数据（NTSC：邻接 1 行，PAL：隔 1 行）。因此，此块保持用于延迟图像信号的 3 行数据。

(2) 水平和垂直滤波器

在适应 2 维的 Y/C 分离中，根据上、下、左、右像素的相关性，适当地切换水平带通滤波器（BPF）、垂直带通滤波器（BPF）和水平垂直带通滤波器（BPF）。此块对输入信号进行水平 BPF、垂直 BPF 和水平垂直 BPF 的处理。在 1 维 Y/C 分离中，只使用水平 BPF。结构图如图 30.32 所示。

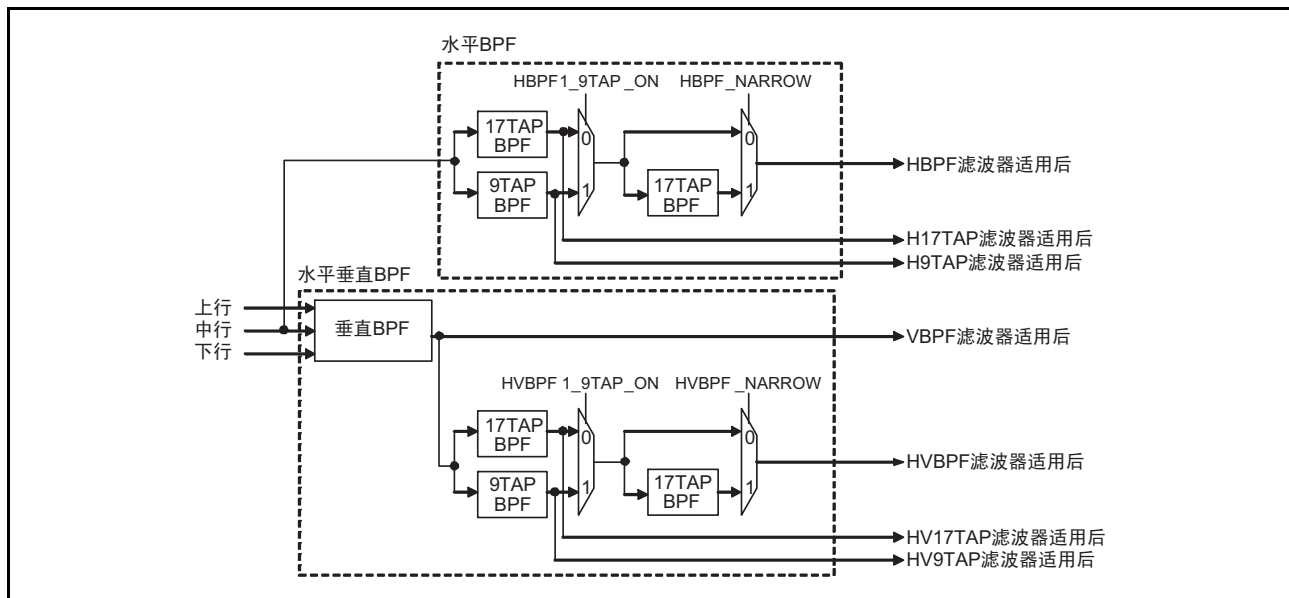


图 30.32 水平和垂直滤波器的结构图

水平方向的带通滤波器由 2 段构成。前段选择 9TAP 或者 17TAP，当 YCSCR8.HBPFF1_9TAP_ON 位和 HVBPF1_9TAP_ON 使用为“0”时，选择 17TAP 的 BPF；当 YCSCR8.HBPFF1_9TAP_ON 位和 HVBPF1_9TAP_ON 位为“1”时，选择 9TAP 的 BPF。后段选择旁路或者 17TAP，当 YCSCR8.HBPFF_NARROW 位和 HVBPF_NARROW 位为“0”时，选择旁路的 BPF；当 YCSCR8.HBPFF_NARROW 位和 HVBPF_NARROW 位为“1”时，选择 17TAP 的 BPF。

(3) 水平和垂直的相关检测

检测水平像素、垂直像素和水平垂直像素的相关值。检测到的相关值中混合的 Y/C 分离 2 维滤波器选择系数为水平滤波器、垂直滤波器和水平垂直滤波器选择的相关值。

Y/C 分离 2 维滤波器的选择系数如表 30.37 所示。

表 30.37 Y/C 分离 2 维滤波器的选择系数

分类	位名	说明	相关性
垂直 Y/C 分离的选择系数	YCSCR5.K21A[5:0]	如果增大此值，垂直 BPF 的适用范围就会变小。	与这些位有关。 如果水平方向的点状干扰显著，就必须减小此值（增大 K24）。 但是，如果将此值（增大 K24）减得太小，就会产生垂直方向的点状干扰。
	YCSCR5.K22A[7:0]	如果增大此值，垂直 BPF 的适用范围就会变小。	
	YCSCR7.K23A[3:0]	如果增大此值，垂直 BPF 的适用范围就会变小。	
	YCSCR7.K24[4:0]	如果增大此值，垂直 BPF 的适用范围就会变大。	
	YCSCR6.K21B[5:0]	如果增大此值，垂直 BPF 的适用范围就会变小。	与这些位有关。 如果水平方向的点状干扰明显，就必须减小此值。 但是，如果将此值减得太小，就会产生垂直方向的点状干扰。
	YCSCR6.K22B[7:0]	如果增大此值，垂直 BPF 的适用范围就会变小。	
	YCSCR7.K23B[3:0]	如果增大此值，垂直 BPF 的适用范围就会变小。	
水平 Y/C 分离选择系数	YCSCR3.K11[5:0]	如果增大此值，水平 BPF 的适用范围就会变小。	与这些位有关。 如果垂直方向的点状干扰明显，就必须增大此值。 但是，如果将此值减得太小，就会产生水平方向的点状干扰。
	YCSCR3.K13[5:0]	如果增大此值，水平 BPF 的适用范围就会变小。	
	YCSCR3.K15[3:0]	如果增大此值，水平 BPF 的适用范围就会变小。	
	YCSCR4.K12[5:0]	如果增大此值，水平 BPF 的适用范围就会变小。	与这些位有关。 如果垂直方向的点状干扰明显，就必须增大此值。 但是，如果将此值减得太小，就会产生水平方向的点状干扰。
	YCSCR4.K14[5:0]	如果增大此值，水平 BPF 的适用范围就会变小。	
	YCSCR4.K16[3:0]	如果增大此值，水平 BPF 的适用范围就会变小。	

(4) 相关检测滤波器

这是此模块独自的相关检测滤波器。重视行之间的相关性，尤其能减轻十字交叉点的点状干扰。对于水平滤波器和垂直滤波器中排除不了的点状干扰，能通过混合相关检测滤波器后的信号来减轻点状干扰。

(5) 水平和垂直的混合

对于通过水平滤波器、垂直滤波器和水平垂直滤波器的信号，分别混合水平滤波器和水平垂直滤波器（前段）适用后的信号，然后根据通过 (3) 算出的相关值，选择通过水平滤波器、垂直滤波器和水平垂直滤波器的信号。

结构图如图 30.33 所示。

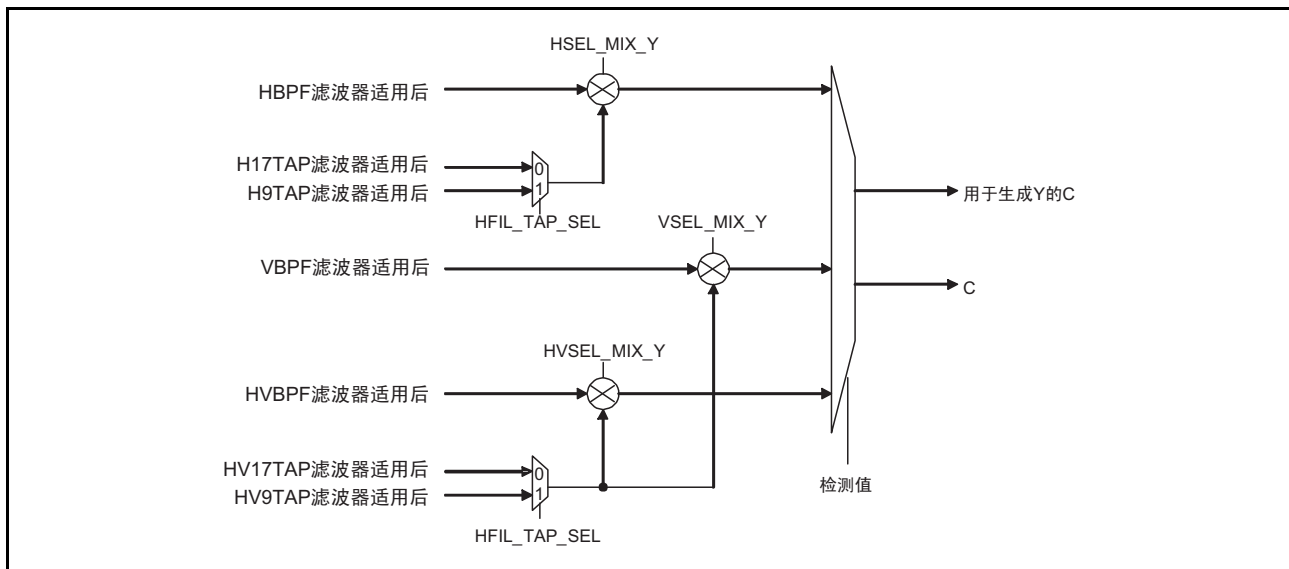


图 30.33 水平和垂直的混合结构图

能通过 YCSCR8.HFIL_TAP_SEL 位选择要混合的水平滤波器和水平垂直滤波器（前段）适用后的信号。当 YCSCR8.HFIL_TAP_SEL 位为“0”时，选择 17TAP 滤波器适用后的信号；当 YCSCR8.HFIL_TAP_SEL 位为“1”时，选择 9TAP 滤波器适用后的信号。

在通过水平滤波器的信号中混合上述水平滤波器（前段）适用后的信号，能通过 YCSCR9.HSEL_MIX_Y[3:0] 位设定混合比。同样，在通过垂直滤波器和水平垂直滤波器的信号中混合上述水平垂直滤波器（前段）适用后的信号，能通过 YCSCR9.VSEL_MIX_Y[3:0] 位和 YCSCR9.HVSEL_MIX_Y[3:0] 位设定混合比。

根据通过 (3) 水平和垂直的相关检测算出的相关值，选择水平滤波器、垂直滤波器和水平垂直滤波器适用后的信号。

(6) 相关检测值的混合

对于通过 (5) 生成的用于 Y 信号的 C 和 C 信号，混合 (4) 相关检测滤波器适用后的信号。
结构图如图 30.34 所示。

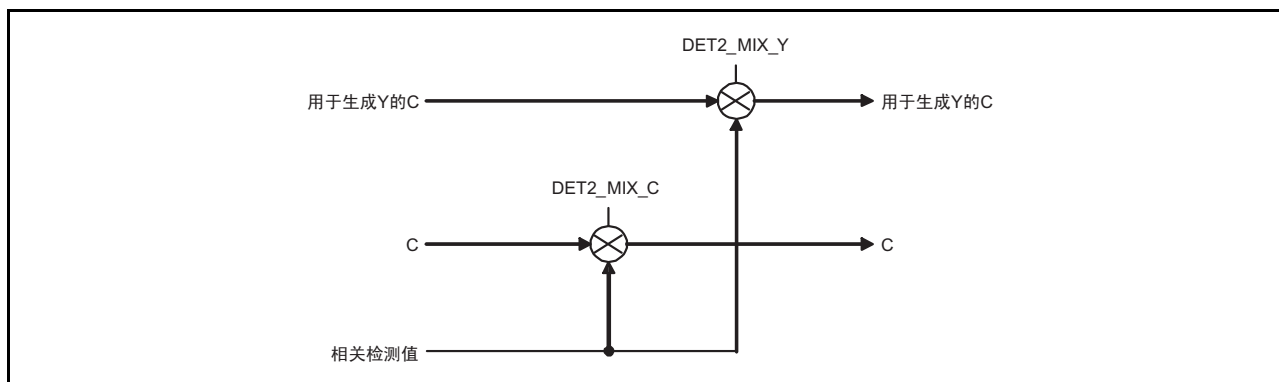


图 30.34 相关检测值混合结构图

当 YCSCR9.DET2_ON 位为“1”时，混合相关检测滤波器适用后的信号，能通过 YCSCR12.DET2_MIX_Y[3:0] 位设定用于生成 Y 信号的 C 信号和相关检测滤波器适用后的信号混合比。同样，能通过 YCSCR12.DET2_MIX_C [3:0] 位设定 C 信号和相关检测滤波器适用后的信号混合比。

当 YCSCR9.DET2_ON 位为“0”时，直接输出水平滤波器和垂直滤波器适用后的信号。

(7) Y 的生成

通过从图像信号中抽出用于生成 Y 信号的 C 信号，生成 Y 信号。

(8) 级联滤波器

C 信号通过用于进一步限制频带的级联滤波器或者 TAKE-OFF 滤波器。
结构图如图 30.35 所示。

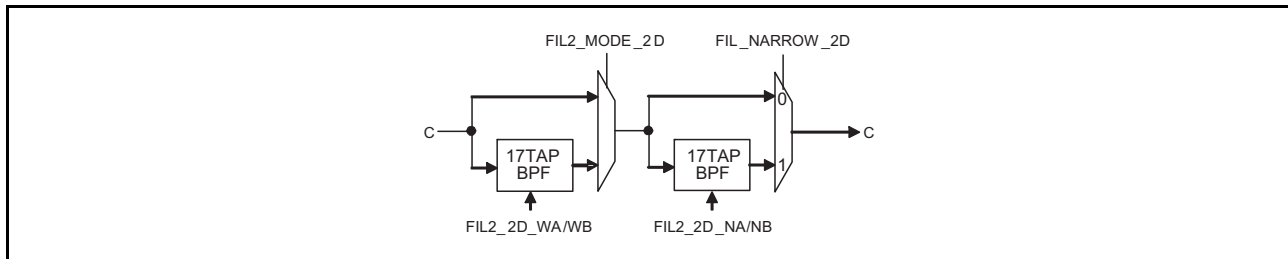


图 30.35 级联滤波器结构图

级联滤波器由 2 段构成。前段选择旁路或者 17TAP，能通过 YCSCR12.FIL2_MODE_2D[1:0] 位选择旁路、级联滤波器或者 TAKE-OFF 滤波器。同样，后段也能通过 YCSCR12.FIL2_NARROW_2D 位选择旁路或者 17TAP。

前段滤波器和后段滤波器都是通用滤波器，能通过 YCTWA_F0 ~ YCTWA_F8、YCTWB_F0 ~ YCTWB_F8、YCTNA_F0 ~ YCTNA_F8 和 YCTNB_F0 ~ YCTNB_F8 进行设定。

各滤波器的推荐设定值如表 30.38 ~ 表 30.40 所示。

表 30.38 2 维 Y/C 滤波器的寄存器设定表 (NTSC)

位名	NTSC					位名	NTSC				
	旁路	级联		TAKE-OFF			旁路	级联		TAKE-OFF	
		滤波器		滤波器				滤波器		滤波器	
		1 段	2 段	宽带	窄带			1 段	2 段	宽带	窄带
FIL2_MODE_2D	0	1		2		FIL2_MODE_2D	0	1		2	
FIL2_NARROW_2D	—	0	1	—	—	FIL2_NARROW_2D	—	0	1	—	—
FIL2_2D_WA_F0	—	24	24	0	0	FIL2_2D_NA_F0	—	—	24	—	—
FIL2_2D_WA_F1	—	44	44	0	-48	FIL2_2D_NA_F1	—	—	44	—	—
FIL2_2D_WA_F2	—	20	20	0	-20	FIL2_2D_NA_F2	—	—	20	—	—
FIL2_2D_WA_F3	—	-52	-52	-28	160	FIL2_2D_NA_F3	—	—	-52	—	—
FIL2_2D_WA_F4	—	-128	-128	96	232	FIL2_2D_NA_F4	—	—	-128	—	—
FIL2_2D_WA_F5	—	-128	-128	228	-116	FIL2_2D_NA_F5	—	—	-128	—	—
FIL2_2D_WA_F6	—	-12	-12	-916	-900	FIL2_2D_NA_F6	—	—	-12	—	—
FIL2_2D_WA_F7	—	132	132	-204	-4	FIL2_2D_NA_F7	—	—	132	—	—
FIL2_2D_WA_F8	—	200	200	1648	1392	FIL2_2D_NA_F8	—	—	200	—	—
FIL2_2D_WB_F0	—	—	—	—	—	FIL2_2D_NB_F0	—	—	—	—	—
FIL2_2D_WB_F1	—	—	—	—	—	FIL2_2D_NB_F1	—	—	—	—	—
FIL2_2D_WB_F2	—	—	—	—	—	FIL2_2D_NB_F2	—	—	—	—	—
FIL2_2D_WB_F3	—	—	—	—	—	FIL2_2D_NB_F3	—	—	—	—	—
FIL2_2D_WB_F4	—	—	—	—	—	FIL2_2D_NB_F4	—	—	—	—	—
FIL2_2D_WB_F5	—	—	—	—	—	FIL2_2D_NB_F5	—	—	—	—	—
FIL2_2D_WB_F6	—	—	—	—	—	FIL2_2D_NB_F6	—	—	—	—	—
FIL2_2D_WB_F7	—	—	—	—	—	FIL2_2D_NB_F7	—	—	—	—	—
FIL2_2D_WB_F8	—	—	—	—	—	FIL2_2D_NB_F8	—	—	—	—	—

表 30.39 2 维 Y/C 滤波器的寄存器设定表 (PAL)

位名	PAL					位名	PAL				
	旁路	级联		TAKE-OFF			旁路	级联		TAKE-OFF	
		滤波器		滤波器				滤波器		滤波器	
		1 段	2 段	宽带	窄带			1 段	2 段	宽带	窄带
FIL2_MODE_2D	0	1		2		FIL2_MODE_2D	0	1		2	
FIL2_NARROW_2D	—	0	1	—	—	FIL2_NARROW_2D	—	0	1	—	—
FIL2_2D_WA_F0	—	-20	-20	0	0	FIL2_2D_NA_F0	—	—	-20	—	—
FIL2_2D_WA_F1	—	24	24	0	0	FIL2_2D_NA_F1	—	—	24	—	—
FIL2_2D_WA_F2	—	64	64	0	-23	FIL2_2D_NA_F2	—	—	64	—	—
FIL2_2D_WA_F3	—	40	40	16	-46	FIL2_2D_NA_F3	—	—	40	—	—
FIL2_2D_WA_F4	—	-76	-76	59	145	FIL2_2D_NA_F4	—	—	-76	—	—
FIL2_2D_WA_F5	—	-164	-164	85	409	FIL2_2D_NA_F5	—	—	-164	—	—
FIL2_2D_WA_F6	—	-84	-84	-498	-918	FIL2_2D_NA_F6	—	—	-84	—	—
FIL2_2D_WA_F7	—	108	108	-101	-363	FIL2_2D_NA_F7	—	—	108	—	—
FIL2_2D_WA_F8	—	216	216	878	1592	FIL2_2D_NA_F8	—	—	216	—	—
FIL2_2D_WB_F0	—	—	—	—	—	FIL2_2D_NB_F0	—	—	—	—	—
FIL2_2D_WB_F1	—	—	—	—	—	FIL2_2D_NB_F1	—	—	—	—	—
FIL2_2D_WB_F2	—	—	—	—	—	FIL2_2D_NB_F2	—	—	—	—	—
FIL2_2D_WB_F3	—	—	—	—	—	FIL2_2D_NB_F3	—	—	—	—	—
FIL2_2D_WB_F4	—	—	—	—	—	FIL2_2D_NB_F4	—	—	—	—	—
FIL2_2D_WB_F5	—	—	—	—	—	FIL2_2D_NB_F5	—	—	—	—	—
FIL2_2D_WB_F6	—	—	—	—	—	FIL2_2D_NB_F6	—	—	—	—	—
FIL2_2D_WB_F7	—	—	—	—	—	FIL2_2D_NB_F7	—	—	—	—	—
FIL2_2D_WB_F8	—	—	—	—	—	FIL2_2D_NB_F8	—	—	—	—	—

表 30.40 2 维 Y/C 滤波器的寄存器设定表 (SECAM)

位名	SECAM				位名	SECAM			
	旁路	级联		TAKE-OFF		旁路	级联		TAKE-OFF
		滤波器		滤波器			滤波器		滤波器
		1 段	2 段				1 段	2 段	
FIL2_MODE_2D	0	1		2	FIL2_MODE_2D	0	1		2
FIL2_NARROW_2D	—	0	1	—	FIL2_NARROW_2D	—	0	1	—
FIL2_2D_WA_F0	—	-20	-20	0	FIL2_2D_NA_F0	—	—	-1008	—
FIL2_2D_WA_F1	—	24	24	-12	FIL2_2D_NA_F1	—	—	1976	—
FIL2_2D_WA_F2	—	64	64	-18	FIL2_2D_NA_F2	—	—	-2024	—
FIL2_2D_WA_F3	—	40	40	38	FIL2_2D_NA_F3	—	—	444	—
FIL2_2D_WA_F4	—	-76	-76	100	FIL2_2D_NA_F4	—	—	1868	—
FIL2_2D_WA_F5	—	-164	-164	88	FIL2_2D_NA_F5	—	—	-2864	—
FIL2_2D_WA_F6	—	-84	-84	-508	FIL2_2D_NA_F6	—	—	1352	—
FIL2_2D_WA_F7	—	108	108	-114	FIL2_2D_NA_F7	—	—	1376	—
FIL2_2D_WA_F8	—	216	216	852	FIL2_2D_NA_F8	—	—	-2240	—
FIL2_2D_WB_F0	—	-12	-12	—	FIL2_2D_NB_F0	—	—	-1080	—
FIL2_2D_WB_F1	—	40	40	—	FIL2_2D_NB_F1	—	—	2800	—
FIL2_2D_WB_F2	—	60	60	—	FIL2_2D_NB_F2	—	—	-3308	—
FIL2_2D_WB_F3	—	12	12	—	FIL2_2D_NB_F3	—	—	1628	—
FIL2_2D_WB_F4	—	-104	-104	—	FIL2_2D_NB_F4	—	—	1444	—
FIL2_2D_WB_F5	—	-156	-156	—	FIL2_2D_NB_F5	—	—	-3308	—
FIL2_2D_WB_F6	—	-64	-64	—	FIL2_2D_NB_F6	—	—	2140	—
FIL2_2D_WB_F7	—	120	120	—	FIL2_2D_NB_F7	—	—	376	—
FIL2_2D_WB_F8	—	208	208	—	FIL2_2D_NB_F8	—	—	-1384	—

(9) 超范围

如果图像信号色振幅的上部或者下部发生上溢或者下溢，就可能无法正常地进行 Y/C 分离，并且因产生点状干扰而出现竖线。为了减轻此现象，在发生上溢或者下溢时，在 Y 信号中自动插入低通滤波器（切断竖线的频率成分）。

通过将 RGORCR7.UCMP_SW 位置“1”，使超范围处理有效；通过将 RGORCR7.DCMP_SW 位置“1”，使欠范围处理有效。能通过 RGORCR1.RADJ_O_LEVEL0[9:0] 位、RGORCR3.RADJ_O_LEVEL1[9:0] 位和 RGORCR5.RADJ_O_LEVEL2[9:0] 位将超范围电平设定为 4 个级别。同样，通过 RGORCR2.RADJ_U_LEVEL0[9:0] 位、RGORCR4.RADJ_U_LEVEL1[9:0] 位和 RGORCR6.RADJ_O_LEVEL2[9:0] 位将欠范围电平设定为 4 个级别。根据超欠范围电平，适当更改要插入的滤波器。

通过将 HWIDE_SW 位置“1”，不仅检测正在处理检测电平的像素而且检测水平方向 5 个像素的最大值（最小值），判断超欠范围。

30.5.6 色度解码

色度解码将通过“Y/C 分离电路”抽出的 C 信号解调为 Cb/Cr 信号，具有通过检测色同步信号的振幅来自动调整色彩增益的 Auto Color Control (ACC) 功能以及对解调时的色调进行调整的 TINT/R-Y 轴校正等功能。

色度解码的框图如图 30.36 所示。

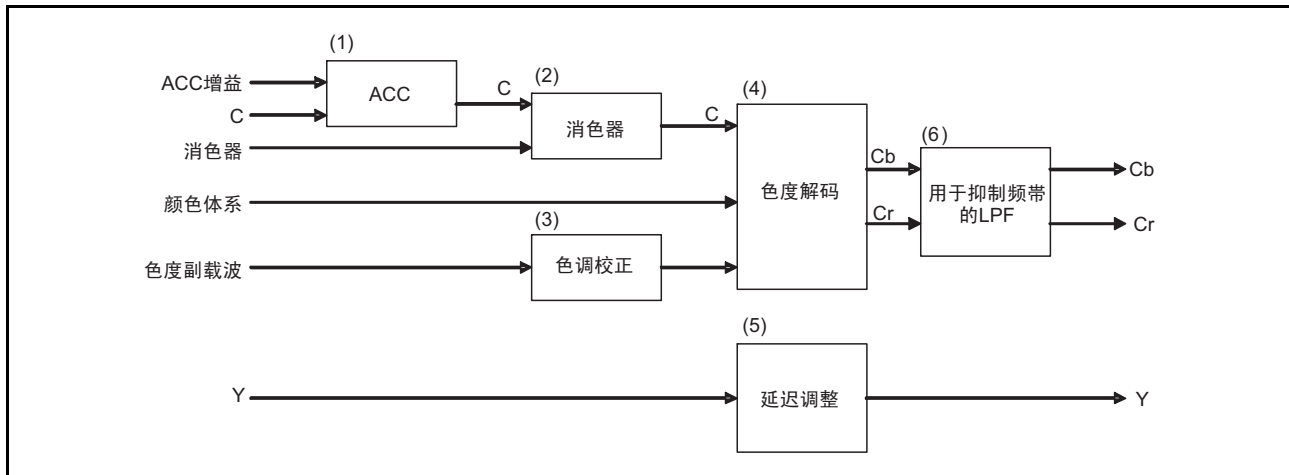


图 30.36 色度解码的框图

(1) Auto Color Control (ACC)

Auto Color Control (ACC) 检测色同步信号的振幅，并且根据目标值自动控制 C 信号的增益。ACC 在 ACCCR1.ACCMODE 位为“0”时运行。

能通过 ACCCR1.ACCLEVEL[8:0] 位设定色同步信号的振幅目标值。在目标值 (ACCCR1.ACCLEVEL[8:0] 位) \pm ACCCR3.ACCPRECIS[5:0] 位的情况下，固定增益。能通过 ACCCR1.ACCMAXGAIN[1:0] 限制最大增益，并且能通过 CROMASR1.ACCMAINGAIN[8:0] 位（主）和 CROMASR1.ACCSUBGAIN[1:0] 位（副）确认当前设定的增益值。

当 ACCCR1.ACCMODE 位为“1”时，能手动设定 C 信号的增益值。能通过 ACCCR2.CHROMAMAINGAIN [8:0] 位（主）和 ACCCR2.CHROMASUBGAIN[1:0] 位（副）设定 C 信号的增益值。

(2) 消色器

在因弱电场等而色同步信号的振幅较小时，消色器删除颜色信息。消色器的 ON/OFF 为滞后，在 ACCCR3.KILLERLEVEL[5:0] 位的情况下，消色器 ON；在 ACCCR3.KILLERLEVEL[5:0] 位 + ACCCR1.KILLEROFFSET [3:0] 位的情况下，消色器 OFF。也能通过 ACCCR3.KILLERMODE 位强制消色器 ON。

(3) 色调校正

能通过调整色度副载波的相位，调整色度解码后的 Cb/Cr 的色调。此调整只对 NTSC 和 PAL 有效。通过 TINTCR.TINTMAIN[9:0] 位控制解调轴的相位，通过 TINTCR.TINTSUB[5:0] 位控制 R-Y 轴的相位。

(4) 色度解码

从 C 信号解调 Cb/Cr 信号。能通过 YCDCR.DEMODMODE[1:0] 位进行解调前的行平均处理。通常必须将 YCDCR.DEMODMODE[1:0] 位置“2”（只有 PAL 才进行 2 行解调，NTSC 进行 1 行解调）。

(5) 延迟调整

通过延迟 Y 信号来调整 Y/C 信号的延迟。能通过 YCDCR.LUMADELAY[4:0] 位调整 -16 ~ 15 个时钟。

(6) 用于控制频带的 LPF

能通过启动低通滤波器，抑制色度解码后的 Cb/Cr 信号的频带。通过 YCDCR.CHROMALPF 位设定低通滤波器的 ON/OFF。

30.5.7 数字钳位电路

在任意的钳位位置，Y 信号进行消隐脉冲钳位处理，Cb/Cr 信号进行中心钳位处理。数字钳位电路具有通过自相关函数来检测噪声量的功能。

数字钳位的框图如图 30.37 所示。

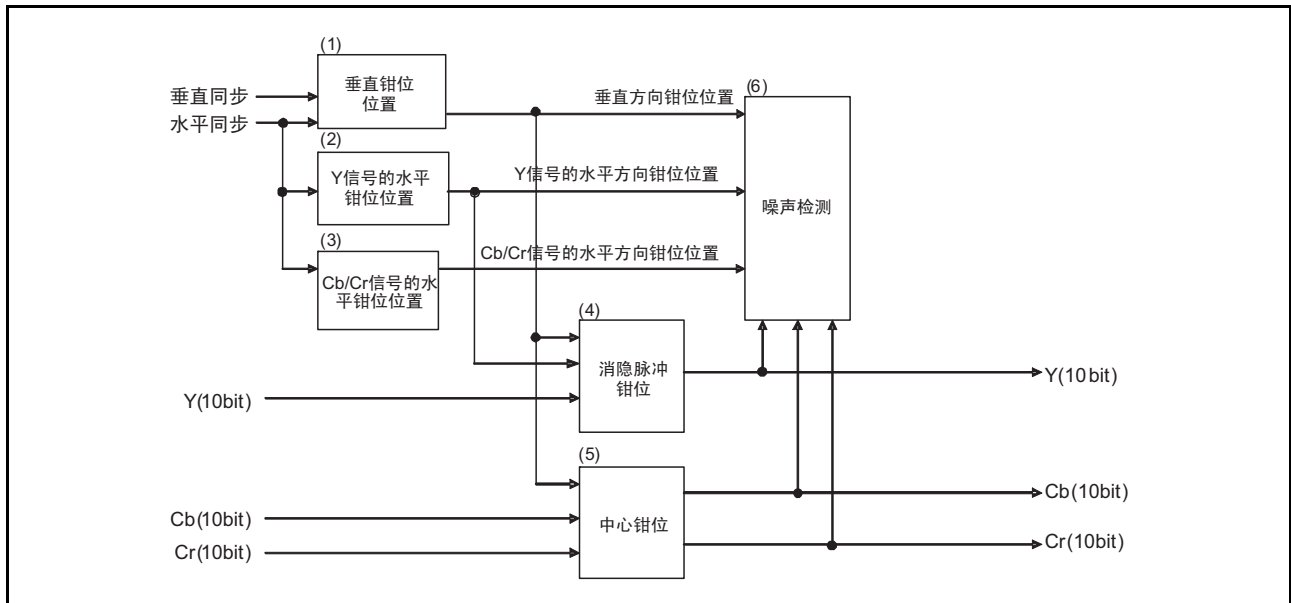


图 30.37 数字钳位的框图

(1) 垂直钳位位置

通过 DCPCR4.DCPSTART[5:0] 位和 DCPCR5.DCPEND[5:0] 位设定垂直方向的钳位位置。垂直方向的钳位位置设定为 Y、Cb 和 Cr 的共用设定。

(2) Y 信号的水平钳位位置

通过 DCPCR7.DCPPOS_Y[7:0] 位设定 Y 信号水平方向的钳位开始位置。通过 DCPCR6.DCPWIDTH[6:0] 位设定水平方向的钳位宽度。水平方向的钳位宽度为 Y、Cb 和 Cr 的共用设定。

(3) Cb/Cr 信号的水平钳位位置

通过 DCPCR8.DCPPOS_C[7:0] 位设定 Cb/Cr 信号水平方向的钳位开始位置。通过 DCPCR6.DCPWIDTH[6:0] 位设定水平方向的钳位宽度。水平方向的钳位宽度为 Y、Cb 和 Cr 的共用设定。

(4) 消隐脉冲钳位

通过消隐脉冲钳位，稳定 Y 信号的消隐脉冲电平。当 DCPCR1.DCPMODE_Y 位为“0”时，Y 信号减去 DCPCR1.BLANKLEVEL_Y[9:0] 位的设定值。

Y 信号输出 = Y 信号输入 - DCPCR1.BLANKLEVEL_Y[9:0]

当 DCPCR1.DCPMODE_Y 位为“1”时，Y 信号减去（在所设钳位位置检测到的 Y 信号电平加上 DCPCR1.BLANKLEVEL_Y[9:0] 后的值）。

Y 信号输出 = Y 信号输入 - (检测值 + DCPCR1.BLANKLEVEL_Y[9:0])

能通过 DCPSR1.CLAMPLEVEL_Y[9:0] 位读检测值。

能通过 DCPCR3.DCPRESPONSE[2:0] 位设定钳位的响应速度。此设定为 Y、Cb 和 Cr 的共用设定。

(5) 中心钳位

通过中心钳位，稳定 Cb/Cr 信号的中心电平。

当 DCPCR2.DCPMODE_C 位为“0”时，Cb/Cr 信号减去 DCPCR2.BLANKLEVEL_CB/CR[5:0] 位的设定值。

Cb 信号输出 = Cb 信号输入 - DCPCR2.BLANKLEVEL_CB[5:0]

Cr 信号输出 = Cr 信号输入 - DCPCR2.BLANKLEVEL_CR[5:0]

当 DCPMODE_C 位为“1”时，Cb/Cr 信号减去（在所设钳位位置检测到的 Cb/Cr 信号电平加上 DCPCR2.BLANKLEVEL_CB/CR[5:0] 后的值）。

Cb 信号输出 = Cb 信号输入 - (检测值 + DCPCR2.BLANKLEVEL_CB[5:0])

Cr 信号输出 = Cr 信号输入 - (检测值 + DCPCR2.BLANKLEVEL_CR[5:0])

能通过 DCPSR1.CLAMPLEVEL_CB[5:0] 位和 DCPSR2.CLAMPLEVEL_CR[5:0] 位读检测值。

能通过 DCPCR3.DCPRESPONSE[2:0] 位设定钳位的响应速度。此设定为 Y、Cb 和 Cr 的共用设定。

(6) 噪声检测

能通过自相关函数检测所设钳位位置的噪声量。通过 NSDCR.ACFINPUT[1:0] 位，从 Y、Cb 和 Cr 中选择自相关函数的信号。通过 NSDCR.ACFLAGTIME[4:0] 位设定计算自相关函数时的延迟时间，通过 NSDCR.ACFFILTER[1:0] 位设定自相关函数的场累积量。能通过 NSDSR.ACFSTRENGTH[15:0] 位读自相关函数（相关系数）。

30.5.8 输出调整电路

此电路能设定捕捉位置以及调整对比度和色彩。

输出调整电路的框图如图 30.38 所示。

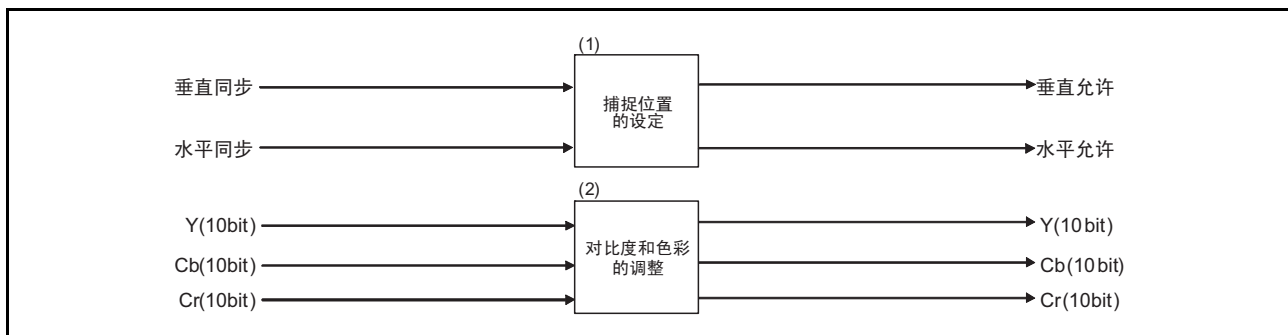


图 30.38 输出调整框图

(1) 捕捉位置的设定

设定输入图像信号的捕捉位置。能通过 TGCR1.SRCLEFT[8:0] 位、TGCR2.SRCTOP[5:0] 位、TGCR2.SRCHEIGHT[9:0] 位和 TGCR3.SRCWIDTH[10:0] 位设定捕捉位置。但是，此设定只能用于此模块内部。有关输入图像的显示尺寸，必须通过视频显示控制器 4 的缩放部的捕捉垂直尺寸寄存器（SCL0_DS2）和捕获水平尺寸寄存器（SCL0_DS3）进行设定。

(2) 对比度和色彩的调整

调整输出 Y 信号、Cb 信号和 Cr 信号的增益。能通过 YGAINCR.Y_GAIN2[9:0] 位调整对比度（调整 Y 信号的增益），能通过 CBGAINCR.CB_GAIN2[9:0] 位和 CRGAINCR.CR_GAIN2[9:0] 位调整色彩（调整 Cb/Cr 信号的增益）。

30.6 推荐设定

此模块的推荐设定值如表 30.41 和表 30.42 所示。

表 30.41 各颜色格式的共用推荐设定一览表

寄存器	位	初始值 (10 进制)	推荐值 (10 进制)	备注
ADCCR1	AGCMODE	0	1	AGC ON
SYNSCR1	LPFVSYNC	3	3	
	LPFHSYNC	3	5	
	VELOCITYSHIFT_H	0	2	
	SLICERMODE_H	2	2	自动限幅
	SLICERMODE_V	2	2	自动限幅
SYNSCR2	SYNCMAXDUTY_H	15	15	
	SYNCMINDUTY_H	10	10	
SYNSCR3	SSCLIPSEL	15	15	
	CSYNCSLICE_H	146	146	
SYNSCR4	SYNCMAXDUTY_V	15	15	
	SYNCMINDUTY_V	10	9	
SYNSCR5	VSYNCDELAY	0	0	
	VSYNCSLICE	11	10	
	CSYNCSLICE_V	146	146	
HAFCCR1	HAFCGAIN	6	12	
	HAFCFREERUN	0	0	
HAFCCR2	HAFSTART	0	0	
	NOX2HOSC	0	1	
	DOX2HOSC	0	0	
HAFCCR3	HAFEND	8	8	
	HAFCMODE	2	2	VBI 期间比较停止
VCDWCR1	VCDFREERUN	0	0	
DCPCR1	DCPMODE_Y	1	1	自动钳位
	DCPCHECK	0	0	
	BLANKLEVEL_Y	0	-40(984)	
DCPCR2	DCPMODE_C	0	0	
	BLANKLEVEL_CB	0	0	
	BLANKLEVEL_CR	0	0	
DCPCR3	DCPRESPONSE	5	0	
DCPCR4	DCPSTART	16	16	
DCPCR5	DCPEND	16	16	
DCPCR6	DCPWIDTH	54	27	
DCPCR7	DCPPOS_Y	162	162	
DCPCR8	DCPPOS_C	27	54	
NSDCR	ACFINPUT	0	0	
	ACFLAGTIME	0	0	
	ACFFILTER	0	3	

寄存器	位	初始值 (10 进制)	推荐值 (10 进制)	备注
BTLCR	LOCKRANGE	1	1	
	LOOPGAIN	1	3	
	LOCKLIMIT	2	1	
	BCOFREERUN	0	0	
BTGPCR	BGPCHECK	0	0	
	BGPWIDTH	36	54	
	BGPSTART	130	110	
ACCCR1	KILLEROFFSET	8	5	
	ACCMODE	0	0	ACC ON
	ACCMAXGAIN	0	0	
ACCCR2	CHROMASUBGAIN	0	0	
	CHROMAMAINGAIN	256	210	
ACCCR3	ACCRESPONSE	1	1	
	ACCPRECIS	20	8	
	KILLERMODE	0	0	
	KILLERLEVEL	9	4	
TINTCR	TINTSUB	0	0	
	TINTMAIN	0	0	
YCDCR	LUMADELAY	0	0	
	CHROMALPF	0	0	
	DEMODMODE	2	2	
AGCCR1	DOREDUCE	0	0	
	NOREDUCE	0	0	
	AGCRESPONSE	5	4	
AGCCR2	AGCPRECIS	10	10	
PKLIMITCR	PEAKLEVEL	0	2	峰值限制器 ON
	PEAKATTACK	2	2	
	PEAKRELEASE	0	3	
	PEAKRATIO	0	0	
	MAXPEAKSAMPLES	0	20	
RGORCR1	RADJ_O_LEVEL0	1023	992	
RGORCR2	RADJ_U_LEVEL0	0	64	
RGORCR3	RADJ_O_LEVEL1	1023	960	
RGORCR4	RADJ_U_LEVEL1	0	48	
RGORCR5	RADJ_O_LEVEL2	1023	928	
RGORCR6	RADJ_U_LEVEL2	0	32	
RGORCR7	TEST_MONI	0	0	
	RADJ_MIX_K_FIX	0	0	
	UCMP_SW	0	1	超范围检测
	DCMP_SW	0	1	欠范围检测
	HWDI_SW	1	1	

寄存器	位	初始值 (10 进制)	推荐值 (10 进制)	备注
AFCPFCR	PHDET_FIX	0	0	
	PHDET_DIV	5	5	
RUPDCR	NEWSETTING	0	1	
YCSCR8	HBPF_NARROW	1	0	
	HVBPF_NARROW	1	0	
	HBPF1_9TAP_ON	0	0	
	HVBPF1_9TAP_ON	0	0	
	HFIL_TAP_SEL	0	0	
YCSCR11	V_Y_LEVEL	3	0	
DCPCR9	CLP_HOLD_ON_Y	1	0	
	CLP_HOLD_ON_CB	1	0	
	CLP_HOLD_ON_CR	1	0	
YCTWA_F0 ~ F8	FIL2_2D_WA_F0 ~ F8	参照 30.5.5(8) 级联滤波器		
YCTWB_F0 ~ F8	FIL2_2D_WB_F0 ~ F8	参照 30.5.5(8) 级联滤波器		
YCTNA_F0 ~ F8	FIL2_2D_NA_F0 ~ F8	参照 30.5.5(8) 级联滤波器		
YCTNB_F0 ~ F8	FIL2_2D_NB_F0 ~ F8	参照 30.5.5(8) 级联滤波器		
YGAINCR	Y_GAIN2	512	816	
CBGAINCR	CB_GAIN2	512	663	
CRGAINCR	CR_GAIN2	512	663	
PGA_UPDATE	PGA_VEN	1	1	
PGACR	PGA_GAIN_SEL	0	0	
	PGA_GAIN	0	0	
ADCCR2	ADC_VINSEL	0	0	

表 30.42 各颜色格式的推荐设定一览表

寄存器	位	NTSC-3.58	NTSC-4.43	PAL-4.43	PAL-M	PAL-N	SECAM
捕捉位置设定							
TGCR1	SRCLEFT	256	256	256	256	256	256
TGCR2	SRCTOP	16	19	19	16	19	19
	SRCHEIGHT	241	288	288	241	288	288
TGCR3	SRCWIDTH	1428	1412	1412	1428	1412	1412
水平 AFC 设定							
HAFCCR1	HAFCTYP	692	704	704	692	704	704
HAFCCR2	HAFCMAX	792	785	785	792	785	785
HAFCCR3	HAFCMIN	592	630	630	592	630	630
垂直递减计数设定							
VCDWCR1	NOVCD50	1	0	0	1	0	0
	NOVCD60	0	1	1	0	1	1
	VCDDEFAULT	2	1	1	2	1	1
	VCDWINDOW	30	30	30	30	30	30
	VCDOFFSET	15	15	15	15	15	15
BCO 设定							

寄存器	位	NTSC-3.58	NTSC-4.43	PAL-4.43	PAL-M	PAL-N	SECAM
BTLCR	DEFAULTSYS	0	0	1	1	1	2
	NONTSC358	0	1	1	1	1	1
	NONTSC443	1	0	1	1	1	1
	NOPALM	1	1	1	0	1	1
	NOPALN	1	1	1	1	0	1
	NOPAL443	1	1	0	1	1	1
	NOSECAM	1	1	1	1	1	0
ACC 电平设定							
ACCCR1	ACCLEVEL	220	220	220	230	230	220
AGC 电平设定							
AGCCR1	AGCLEVEL	230	230	242	242	242	242
Y/C 分离设定							
YCSCR3	K15	2	2	2	2	2	2
	K13	8	8	8	8	8	8
	K11	4	4	3	3	3	4
YCSCR4	K16	3	3	4	4	4	3
	K14	16	16	63	63	63	16
	K12	8	8	2	2	2	1
YCSCR5	K22A	32	32	32	32	32	32
	K21A	6	6	10	10	10	10
YCSCR6	K22B	8	8	15	15	15	15
	K21B	6	6	10	10	10	6
YCSCR7	K23B	6	6	3	3	3	3
	K23A	3	3	3	3	3	3
	K24	5	5	8	8	8	8
YCSCR9	DET2_ON	1	1	0	0	0	1
	HSEL_MIX_Y	6	6	0	0	0	6
	VSEL_MIX_Y	6	6	0	0	0	6
	HVSEL_MIX_Y	0	0	0	0	0	0
YCSCR12	DET2_MIX_C	0	0	0	0	0	0
	DET2_MIX_Y	2	2	0	0	0	0
	FIL2_MODE_2D	1	1	0	0	0	1
	FIL2_NARROW_2D	1	1	1	1	1	1

30.7 连接例子

此模块的引脚连接例子如图 30.39 所示。

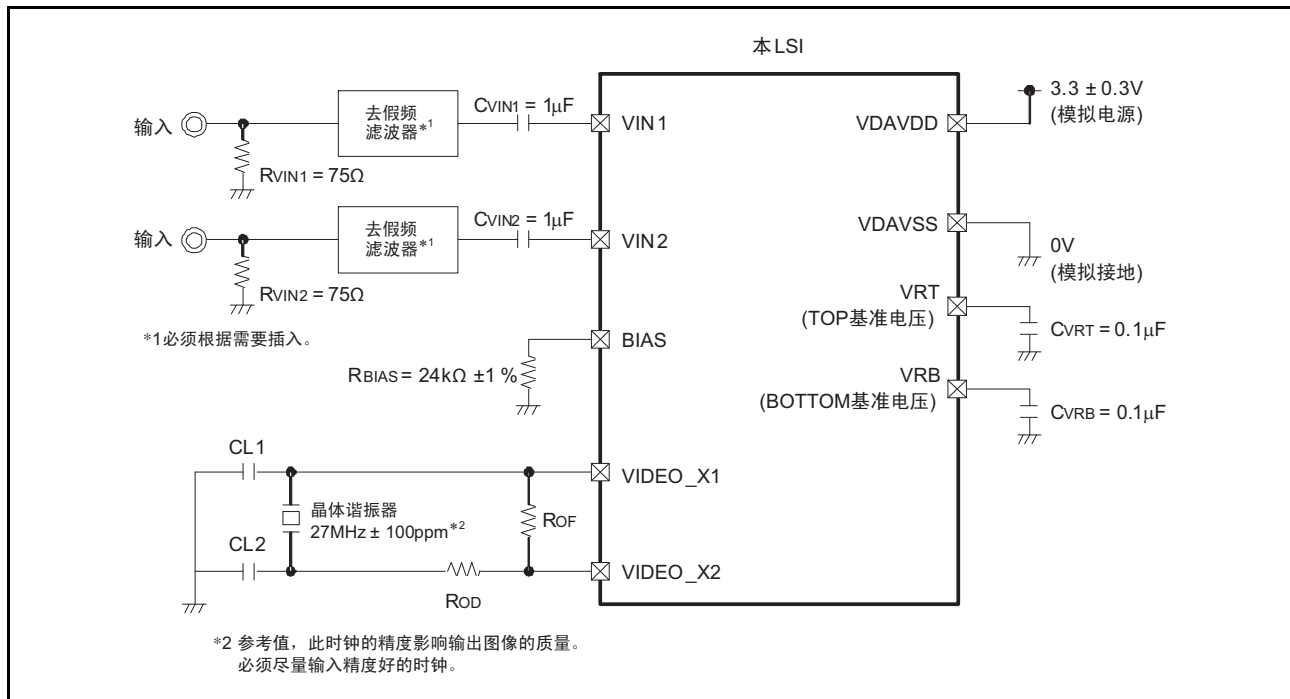


图 30.39 引脚连接例子

31. 视频显示控制器 4 (1) 概要

31.1 特点

视频显示控制器 4 由以下 6 个块构成。图像合成可选择 3 面图形或者图像 +2 面图形。

1. 输入控制部：输入图像的选择、同步调整、水平噪声降低以及通过色彩矩阵功能进行的亮度调整、增益调整、YCbCr 和 GBR 的相互转换
2. 缩放部：使用帧缓冲器的输入图像的扩大、缩小和旋转以及进行帧缓冲器所设场数的重复录像的功能
3. 图像质量改善部：黑色信号展宽、LTI/清晰度、由色彩矩阵功能进行的 YCbCr 和 GBR 之间的转换
4. 图像合成部：3 面图形或者图像 +2 面图形的合成功能
5. 输出控制部：亮度/对比度调整、灰度校正、抖动处理、输出格式转换、TFT-LCD 显示屏的控制信号输出功能
6. 系统控制部：中断控制、显示屏时钟控制、CLUT 表选择信号状态标志的输出功能

视频显示控制器 4 的主要功能如表 31.1 所示。

表 31.1 视频显示控制器 4 的功能

项目	功能
工作频率	输入图像时钟：27/54MHz（视频图像）、~ 66.67MHz（RGB 图像） 显示屏时钟：~ 66.67MHz（取决于显示屏规格）
输入图像规格	符合 ITU-R BT.656 规格的 8bit（27MHz）。 符合 ITU-R BT.601 规格的 8bit（27MHz、隔行扫描信号）。 符合 ITU-R BT.601 规格的 8bit（54MHz、逐行扫描信号）。 数字引脚输入：支持 YCbCr444、RGB888、RGB666、RGB565 图像 数字引脚输入尺寸： 能设定的输入图像尺寸*：水平 1024 像素、垂直 1024 行 【注】* 取决于和连接设备的 AC 特性。 输入图像尺寸例子： SVGA（800×600）、WVGA（800×480）、VGA（640×480）、WAVGA（480×240）、 QVGA 横向尺寸（320×240）、QVGA 纵向尺寸（240×320）
图像录像功能	使用 YCbCr422/RGB565/RGB888 格式，以 1/1、1/2、1/4、1/8 场频保存图像。 能保存的最大图像尺寸：和输入图像同倍
图像质量调整功能	对比度调整、亮度调整、水平噪声降低、黑色信号展宽、LTI/清晰度
图像缩放处理	垂直 1/8 ~ 8 倍、线性内插 / 保持内插 水平 1/8 ~ 8 倍、线性内插 / 保持内插 能通过调整初始相位进行 IP 转换。
图像旋转功能	0、90、180、270 度旋转和水平镜像 (但是，不能和失真校正引擎一起控制)

项目	功能
图形	图形面数: 3 面 (图形 (1)、(2)、(3)) 对应的图像格式: RGB565 逐行格式 (α : 无, R: 5bit, G: 6bit, B: 5bit 共计 16bit) RGB888 逐行格式 (α : 无, R: 8bit, G: 8bit, B: 8bit 共计 24bit) α RGB1555 逐行格式 (α : 1bit, R: 5bit, G: 5bit, B: 5bit 共计 16bit) α RGB4444 逐行格式 (α : 4bit, R: 4bit, G: 4bit, B: 4bit 共计 16bit) α RGB8888 逐行格式 (α : 8bit, R: 8bit, G: 8bit, B: 8bit 共计 32bit) CLUT8 逐行格式 (CLUT: 8bit) CLUT4 逐行格式 (CLUT: 4bit) CLUT1 逐行格式 (CLUT: 1bit) YCC422 逐行格式 (Y: 8bit, Cb/Cr: 8bit 共计 16bit) (只限于图形 (1)) 能读的最大图像尺寸: 水平 1024 像素、垂直 1024 行
图形功能	矩形区域 α 混合 在设定的区域中, 根据透明度 α 进行混合。 (有淡入 / 淡出功能) 色键 通过设定的 RGB 色彩和 CLUT 值并且根据透明度 α 进行混合。 像素单位 α 混合 在 α RGB1555、 α RGB4444、 α RGB8888 和 CLUT8/4/1 格式的图形 图像时, 根据透明度 α 进行混合。 在各点中, 上述功能的 α 值优先级为: 矩形区域 α 混合 > 色键 > 像素单位 α 混合
输出图像尺寸	能设定的最大输出图像尺寸 *: 水平 1999 像素、垂直 2035 行 【注】 * 取决于和显示屏的 AC 特性。 输入图像尺寸例子: SVGA (800×600)、WVGA (800×480)、 VGA (640×480)、WQVGA (480×240)、 QVGA 横向尺寸 (320×240)、 QVGA 纵向尺寸 (240×320)
输出图像格式	RGB888 逐行图像输出 (24bit 并行输出) RGB666 逐行图像输出 (18bit 并行输出) RGB565 逐行图像输出 (16bit 并行输出) RGB888 逐行图像输出 (8bit 串行输出)
显示屏输出调整功能	显示屏的亮度 / 对比度调整、RGB 灰度校正、抖动处理、输出格式转换
同步信号输出	输出 TFT-LCD 显示屏的控制信号。
中断输出	图像输入 / 输出的垂直同步信号 行中断输出 (能在任意行输出) 图像输入的垂直同步周期异常检测信号 场写结束信号 内部缓冲器的上溢 / 下溢检测信号

31.2 框图

此模块的整体框图如下所示，详细内容请参照各块的说明。

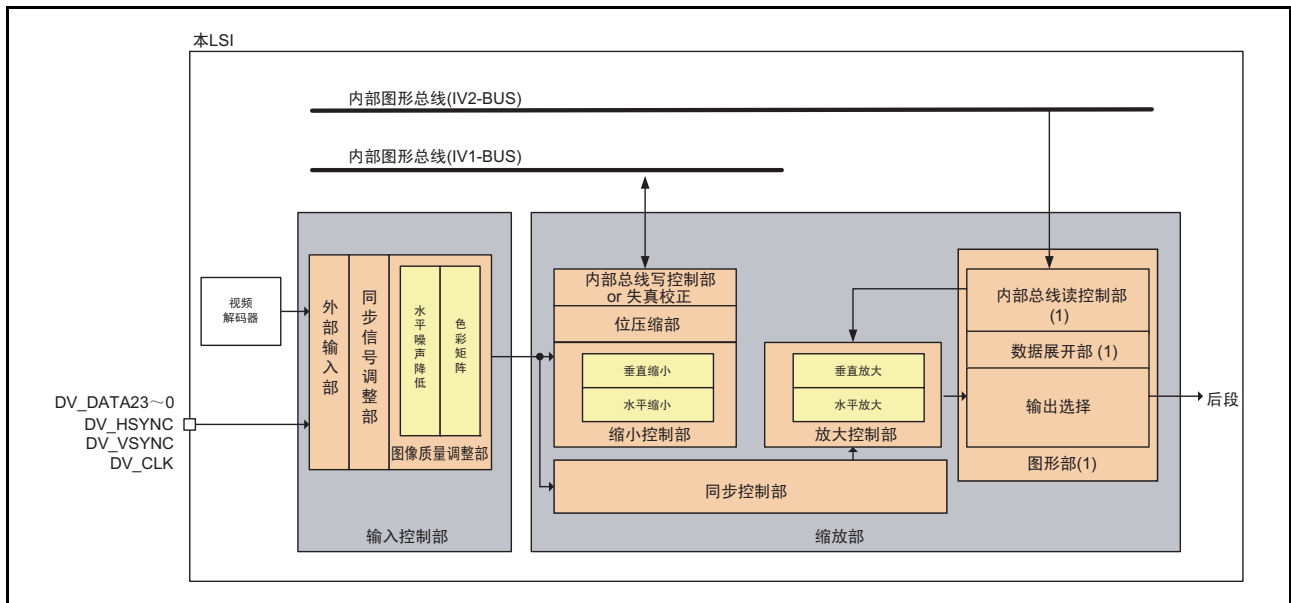


图 31.1 视频显示控制器 4 的前段框图

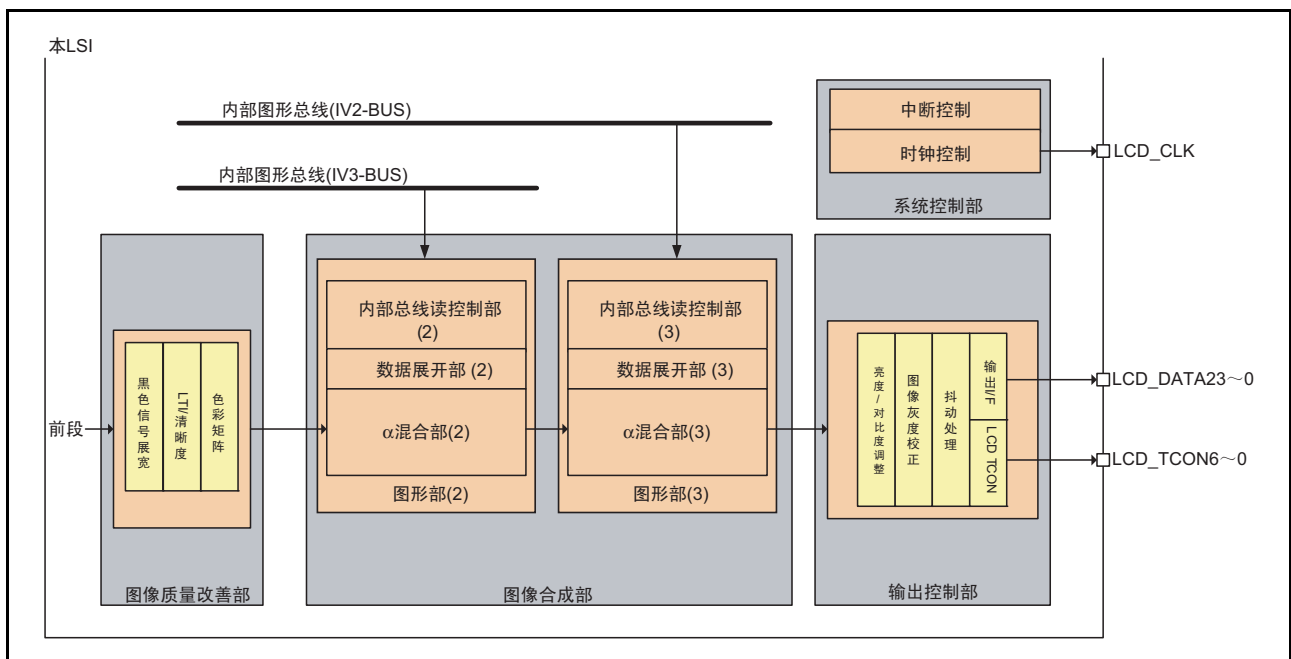


图 31.2 视频显示控制器 4 的后段框图

31.3 输入 / 输出引脚

引脚结构如表 31.2 所示

表 31.2 输入 / 输出引脚

引脚名	输入 / 输出	名称	说明
DV_CLK	输入	外部输入时钟	外部输入的时钟引脚
DV_VSYNC	输入	外部输入垂直同步	外部输入的垂直同步引脚
DV_HSYNC	输入	外部输入水平同步	外部输入的水平同步引脚
DV_DATA23~0	输入	外部输入图像数据	外部输入的图像数据引脚
LCD_CLK	输出	显示屏时钟	显示屏输出的时钟引脚
LCD_DATA23~0	输出	显示屏图像数据	显示屏输出的图像数据引脚
LCD_TCON6~0	输出	显示屏控制信号	显示屏输出的时序控制引脚
LCD_EXTCLK	输入	显示屏时钟源	显示屏时钟源的输入引脚

31.4 时钟

此模块在各功能中使用的时钟主要有图像时钟和像素时钟。

从输入控制部开始，经过缩放部的缩小控制部到将图像写到缓冲器（内部总线写控制）为止使用图像时钟。当输入控制部的 INP_SEL_CNT.INP_SEL 位为“0”（选择视频解码器输出）时，VIDEO_X1(27MHz) 时钟为图像时钟；当 INP_SEL_CNT.INP_SEL 位为“1”（选择外部输入引脚）时，DV_CLK 时钟为图像时钟。

从读缩放部的图形（内部总线读控制部）开始到输出控制部为止使用像素时钟。如果在输出控制部选择并行 RGB 输出，像素时钟频率就和显示屏时钟（LCD_CLK）相同。显示屏时钟能通过系统控制部的 SYSCNT_PANEL_CLK.PANEL_ICKSEL[1:0] 位选择图像时钟、LCD_EXTCLK 或者外围总线时钟 1（P1 ϕ ）。如果在输出控制部选择串行 RGB（3/4 倍速模式），像素时钟频率就为显示屏时钟（LCD_CLK）的 1/3 或者 1/4。

31.5 水平 / 垂直同步信号

在缩放部的同步控制部，通过缩放部的放大控制部生成后段逻辑电路中使用的水平 / 垂直同步信号。因为此水平 / 垂直同步信号是生成用于驱动 LCD 显示屏各种时序的 LCD TCON 基准信号，所以传递给 LCD 显示屏的控制信号（LCD_TCON6~0 引脚）也是以这些信号为基准。

输出的水平同步信号总是为自激周期，通过 SCL0_FRC4.RES_FH[10:0] 位设定水平周期。另外，通过缩放部的 SCL0_FRC3.RES_VS_SEL 位选择外部输入（视频解码器或者数字引脚输入）的垂直同步信号或者自激垂直同步信号作为要输出的垂直同步信号。

(1) 外部输入的垂直同步

在此模式中，生成以外部输入的垂直同步信号为基准输出的垂直同步信号。在将视频解码器或者数字引脚的输入图像显示到显示屏时，必须设定为此模式。在此模式中，输出的水平同步信号为自激周期，时序图如下所示：

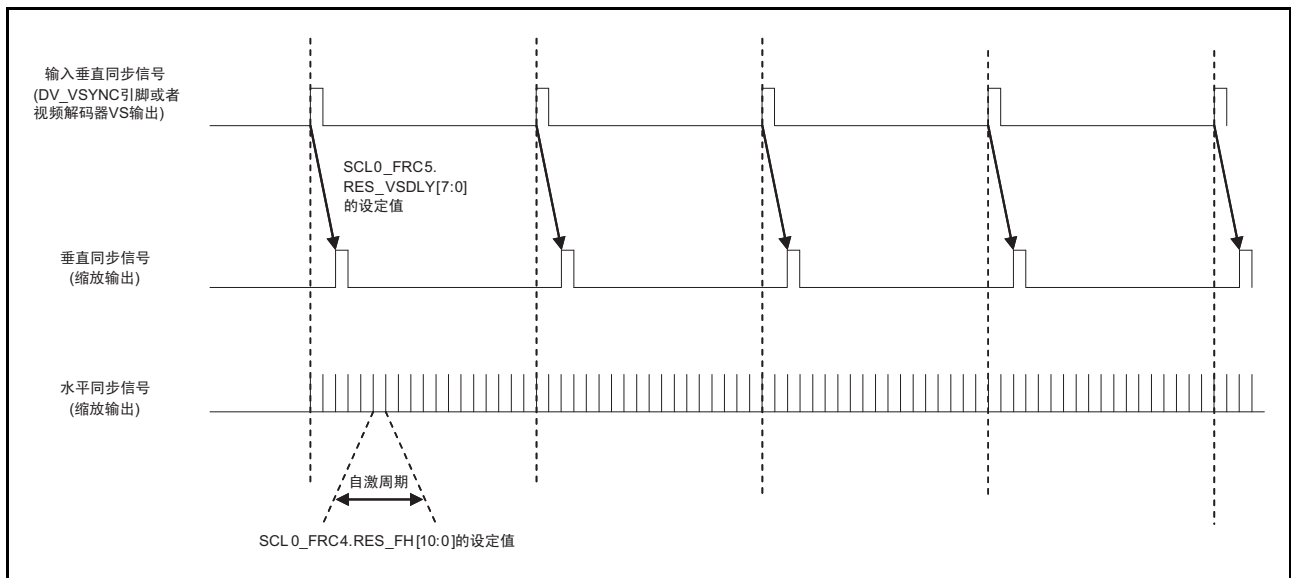


图 31.3 外部输入的垂直同步时序图

(2) 自激垂直同步

在此模式中，以像素时钟为基准生成垂直同步信号（自激）。通过 `SCL0_FRC4.RES_FV[10:0]` 位设定垂直周期，输出的水平同步信号也为自激周期，时序图如下所示：

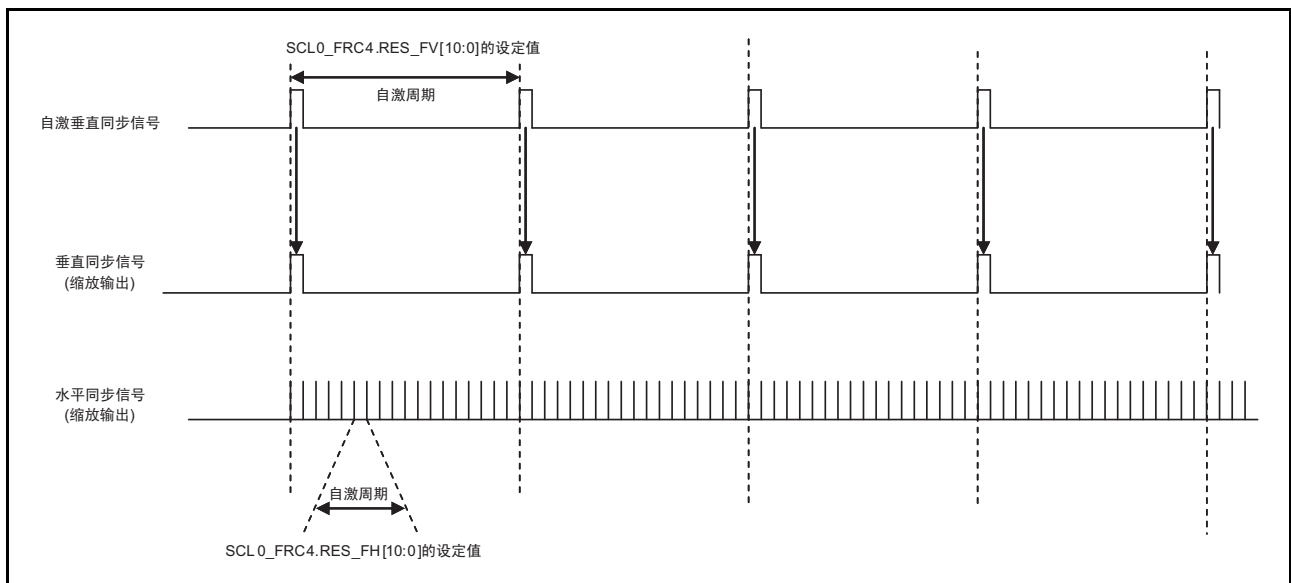


图 31.4 自激垂直同步时序图

(3) 同步信号选择一览表

同步信号选择的使用例子一览表如下所示:

表 31.3 同步信号选择一览表

外部图像输入	图形显示	垂直同步信号选择
无显示	无	不使用
无显示	有	自激垂直同步
有显示	无	外部输入的垂直同步
有显示	有	外部输入的垂直同步
录像	无	不使用
录像	有	自激垂直同步

(4) 切换垂直同步信号时的注意事项

在切换垂直同步信号的选择时，因为输出的垂直同步信号不连续，所以显示屏会出现乱显示的期间。因此，必须在根据需要进行静噪处理后的状态下切换垂直同步信号。必须根据显示屏的规格进行静噪处理。

32. 视频显示控制器 4 (2) 输入控制部

32.1 输入控制功能

32.1.1 功能概要

输入控制部进行视频解码器的输出、外部输入引脚的信号选择、同步调整、水平噪声降低以及通过色彩矩阵进行的亮度调整、增益调整、YCbCr 和 GBR 的相互转换。

输入控制部的功能框图如下所示：

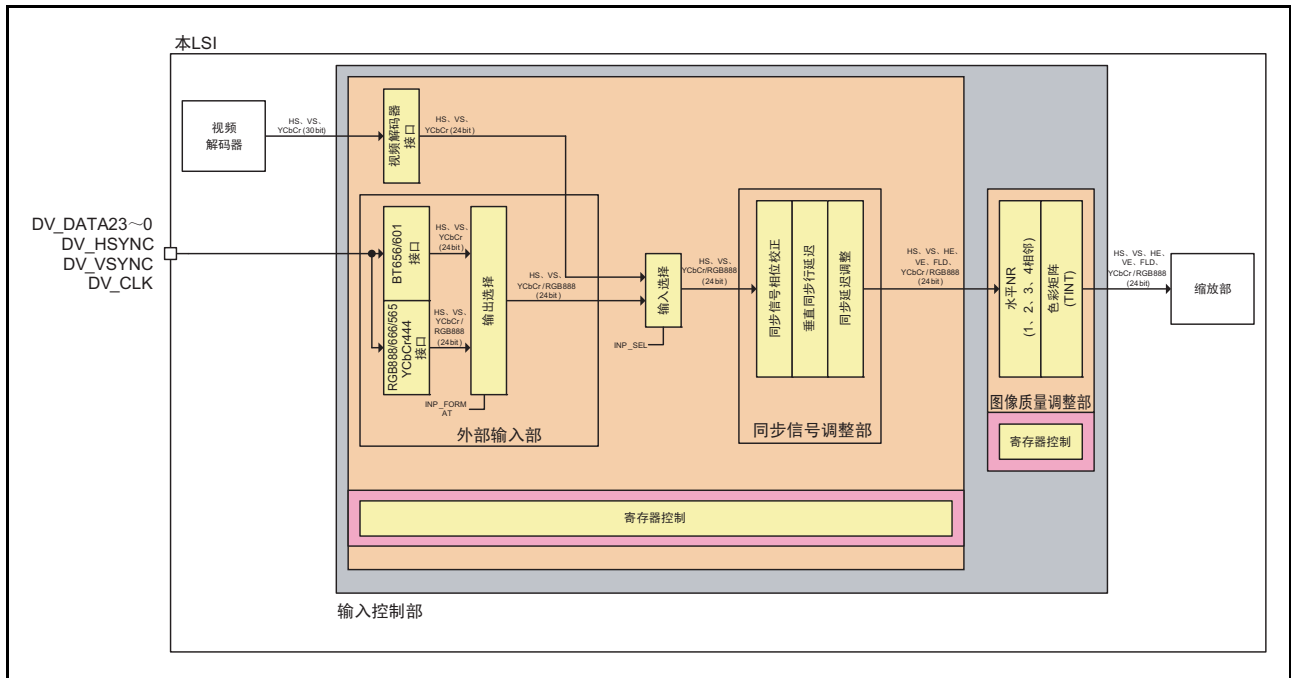


图 32.1 输入控制部的功能框图

32.1.2 外部输入部和同步信号调整部的寄存器更新控制

通过将更新控制寄存器置“1”，更新外部输入部和同步信号调整部的控制寄存器。

图像质量调整部的控制寄存器全部通过垂直同步信号对更新时序进行管理。

在将更新控制寄存器置“1”后，在垂直同步信号的上升沿反映各种寄存器。反映后，更新控制寄存器自动清“0”。

表 32.1 寄存器的更新控制

寄存器名	位名	初始值	说明
INP_UPDATE	INP_EXT_UPDATE	0	外部输入部的寄存器更新 0: 不更新寄存器 1: 更新寄存器
INP_UPDATE	INP_IMG_UPDATE	0	同步信号调整部的寄存器更新 0: 不更新寄存器 1: 更新寄存器
IMGCNT_UPDATE	IMGCNT_VEN	0	图像质量调整部的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器

32.1.3 输入选择

选择视频解码器的输出信号和外部输入引脚的信号。

表 32.2 输入选择

寄存器名	位名	初始值	说明
INP_SEL_CNT	INP_SEL	0	输入选择 0: 视频解码器的输出 1: 外部输入引脚

32.1.4 外部输入的图像信号控制

外部输入的图像信号支持 YCbCr444、RGB888、RGB666、RGB565、BT656 和 BT601 格式。

BT656 信号支持隔行扫描信号的 525Line/59.94Hz (27.0MHz) 和 625Line/50.00Hz (27.0MHz)。

BT601 信号支持数据行为 8 位的 525Line/59.94Hz (27.0MHz) 和 625Line/50.00Hz (27.0MHz) 的隔行扫描信号以及 525Line/59.94Hz (54.0MHz) 和 625Line/50.00Hz (54.0MHz) 的逐行扫描信号。

通过 INP_FORMAT[2:0] 位选择上述信号，通过 INP_ENDIAN_ON 位和 INP_SWAP_ON 位控制位序和 B/R 信号的交换。

表 32.3 外部输入的图像信号控制

寄存器名	位名	初始值	说明
INP_SEL_CNT	INP_FORMAT[2:0]	000	外部输入的格式选择 0: YCbCr444、RGB888 1: RGB666 2: RGB565 3: BT656 4: BT601 5 ~ 7: 禁止设定
INP_EXT_SYNC_CNT	INP_ENDIAN_ON	0	外部输入的位序变更 ON/OFF 控制 0: OFF 1: ON
INP_EXT_SYNC_CNT	INP_SWAP_ON	0	外部输入的 B/R 信号交换 ON/OFF 控制 0: OFF 1: ON

32.1.5 外部输入的时钟边沿选择

通过 INP_PXD_EDGE 位、INP_VS_EDGE 位和 INP_HS_EDGE 位分别选择图像信号、垂直同步信号和水平同步信号的接受时钟边沿。

表 32.4 外部输入的时钟边沿选择

寄存器名	位名	初始值	说明
INP_SEL_CNT	INP_PXD_EDGE	0	外部输入图像信号 DV_DATA23 ~ 0 的输入段捕捉时钟边沿选择 0: 上升沿 1: 下降沿
INP_SEL_CNT	INP_VS_EDGE	0	外部输入垂直同步信号 DV_VSYNC 的输入段捕捉时钟边沿选择 0: 上升沿 1: 下降沿
INP_SEL_CNT	INP_HS_EDGE	0	外部输入水平同步信号 DV_HSYNC 的输入段捕捉时钟边沿选择 0: 上升沿 1: 下降沿

外部输入的标准输入时序如图 32.2 所示。

能在 INP_PXD_EDGE 位、INP_VS_EDGE 位和 INP_HS_EDGE 位为“0”时，在时钟信号 DV_CLK 的上升沿接受各输入信号。

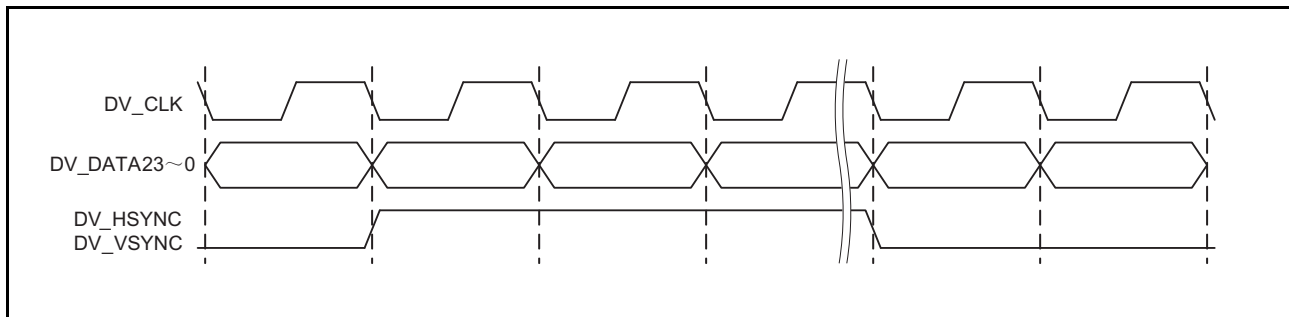


图 32.2 外部输入的标准输入时序 (时钟相位)

32.1.6 外部输入的同步信号反相控制

通过 INP_VS_INV 位和 INP_HS_INV 位对垂直同步信号和水平同步信号的极性进行反相控制。

表 32.5 同步信号的反相控制

寄存器名	位名	初始值	说明
INP_EXT_SYNC_CNT	INP_VS_INV	0	外部输入垂直同步信号 DV_VSYNC 的反相控制 0: 正相 (正极性) 1: 反相 (负极性)
INP_EXT_SYNC_CNT	INP_HS_INV	0	外部输入水平同步信号 DV_HSYNC 的反相控制 0: 正相 (正极性) 1: 反相 (负极性)

32.1.7 外部输入图像信号的位分配

各种格式的外部输入图像信号 DV_DATA 引脚的信号分配如下所示。

(1) YCbCr444/RGB888 输入

如表 32.6 所示，YCbCr444/RGB888 输入时的外部输入图像信号 DV_DATA 引脚分配给内部信号 Y/GOUT、Cb/BOUT 和 Cr/ROUT。

表 32.6 YCbCr444/RGB888 输入时的 DV_DATA 信号输入的位分配

INP_FORMAT[2:0]	0	0	0	0
INP_ENDIAN_ON	0	0	1	1
INP_SWAP_ON	0	1	0	1
DV_DATA23	Cr/ROUT[7]	Cb/BOUT[7]	Cr/ROUT[0]	Cb/BOUT[0]
DV_DATA22	Cr/ROUT[6]	Cb/BOUT[6]	Cr/ROUT[1]	Cb/BOUT[1]
DV_DATA21	Cr/ROUT[5]	Cb/BOUT[5]	Cr/ROUT[2]	Cb/BOUT[2]
DV_DATA20	Cr/ROUT[4]	Cb/BOUT[4]	Cr/ROUT[3]	Cb/BOUT[3]
DV_DATA19	Cr/ROUT[3]	Cb/BOUT[3]	Cr/ROUT[4]	Cb/BOUT[4]
DV_DATA18	Cr/ROUT[2]	Cb/BOUT[2]	Cr/ROUT[5]	Cb/BOUT[5]
DV_DATA17	Cr/ROUT[1]	Cb/BOUT[1]	Cr/ROUT[6]	Cb/BOUT[6]
DV_DATA16	Cr/ROUT[0]	Cb/BOUT[0]	Cr/ROUT[7]	Cb/BOUT[7]
DV_DATA15	Y/GOUT[7]	Y/GOUT[7]	Y/GOUT[0]	Y/GOUT[0]
DV_DATA14	Y/GOUT[6]	Y/GOUT[6]	Y/GOUT[1]	Y/GOUT[1]
DV_DATA13	Y/GOUT[5]	Y/GOUT[5]	Y/GOUT[2]	Y/GOUT[2]
DV_DATA12	Y/GOUT[4]	Y/GOUT[4]	Y/GOUT[3]	Y/GOUT[3]
DV_DATA11	Y/GOUT[3]	Y/GOUT[3]	Y/GOUT[4]	Y/GOUT[4]
DV_DATA10	Y/GOUT[2]	Y/GOUT[2]	Y/GOUT[5]	Y/GOUT[5]
DV_DATA9	Y/GOUT[1]	Y/GOUT[1]	Y/GOUT[6]	Y/GOUT[6]
DV_DATA8	Y/GOUT[0]	Y/GOUT[0]	Y/GOUT[7]	Y/GOUT[7]
DV_DATA7	Cb/BOUT[7]	Cr/ROUT[7]	Cb/BOUT[0]	Cr/ROUT[0]
DV_DATA6	Cb/BOUT[6]	Cr/ROUT[6]	Cb/BOUT[1]	Cr/ROUT[1]
DV_DATA5	Cb/BOUT[5]	Cr/ROUT[5]	Cb/BOUT[2]	Cr/ROUT[2]
DV_DATA4	Cb/BOUT[4]	Cr/ROUT[4]	Cb/BOUT[3]	Cr/ROUT[3]
DV_DATA3	Cb/BOUT[3]	Cr/ROUT[3]	Cb/BOUT[4]	Cr/ROUT[4]
DV_DATA2	Cb/BOUT[2]	Cr/ROUT[2]	Cb/BOUT[5]	Cr/ROUT[5]
DV_DATA1	Cb/BOUT[1]	Cr/ROUT[1]	Cb/BOUT[6]	Cr/ROUT[6]
DV_DATA0	Cb/BOUT[0]	Cr/ROUT[0]	Cb/BOUT[7]	Cr/ROUT[7]

(2) RGB666 输入

如表 32.7 所示，RGB666 输入时的外部输入图像信号 DV_DATA 引脚分配给内部信号 GOUT、BOUT 和 ROUT。

根据以下计算式，从 RGB666 接口（共 24 位）输出被分配的 GOUT、BOUT 和 ROUT 图像信号。

$$G[7:0]=GOUT[7:2]\times 255\div 63$$

$$B[7:0]=BOUT[7:2]\times 255\div 63$$

$$R[7:0]=ROUT[7:2]\times 255\div 63$$

表 32.7 RGB666 输入时的 DV_DATA 引脚输入的位分配

INP_FORMAT[2:0]	1	1	1	1
INP_ENDIAN_ON	0	0	1	1
INP_SWAP_ON	0	1	0	1
DV_DATA17	ROUT[7]	BOUT[7]	ROUT[2]	BOUT[2]
DV_DATA16	ROUT[6]	BOUT[6]	ROUT[3]	BOUT[3]
DV_DATA15	ROUT[5]	BOUT[5]	ROUT[4]	BOUT[4]
DV_DATA14	ROUT[4]	BOUT[4]	ROUT[5]	BOUT[5]
DV_DATA13	ROUT[3]	BOUT[3]	ROUT[6]	BOUT[6]
DV_DATA12	ROUT[2]	BOUT[2]	ROUT[7]	BOUT[7]
DV_DATA11	GOUT[7]	GOUT[7]	GOUT[2]	GOUT[2]
DV_DATA10	GOUT[6]	GOUT[6]	GOUT[3]	GOUT[3]
DV_DATA9	GOUT[5]	GOUT[5]	GOUT[4]	GOUT[4]
DV_DATA8	GOUT[4]	GOUT[4]	GOUT[5]	GOUT[5]
DV_DATA7	GOUT[3]	GOUT[3]	GOUT[6]	GOUT[6]
DV_DATA6	GOUT[2]	GOUT[2]	GOUT[7]	GOUT[7]
DV_DATA5	BOUT[7]	ROUT[7]	BOUT[2]	ROUT[2]
DV_DATA4	BOUT[6]	ROUT[6]	BOUT[3]	ROUT[3]
DV_DATA3	BOUT[5]	ROUT[5]	BOUT[4]	ROUT[4]
DV_DATA2	BOUT[4]	ROUT[4]	BOUT[5]	ROUT[5]
DV_DATA1	BOUT[3]	ROUT[3]	BOUT[6]	ROUT[6]
DV_DATA0	BOUT[2]	ROUT[2]	BOUT[7]	ROUT[7]

(3) RGB565 输入

如表 32.8 所示，RGB565 输入时的外部输入图像信号 DV_DATA 引脚分配给内部信号 GOUT、BOUT 和 ROUT。

根据以下计算式，从 RGB565 接口（共 24 位）输出被分配的 GOUT、BOUT 和 ROUT 图像信号。

$$G[7:0]=GOUT[7:2]\times 255\div 63$$

$$B[7:0]=BOUT[7:3]\times 255\div 31$$

$$R[7:0]=ROUT[7:3]\times 255\div 31$$

表 32.8 RGB565 输入时的 DV_DATA 引脚输入的位分配

INP_FORMAT[2:0]	2	2	2	2
INP_ENDIAN_ON	0	0	1	1
INP_SWAP_ON	0	1	0	1
DV_DATA15	ROUT[7]	BOUT[7]	ROUT[3]	BOUT[3]
DV_DATA14	ROUT[6]	BOUT[6]	ROUT[4]	BOUT[4]
DV_DATA13	ROUT[5]	BOUT[5]	ROUT[5]	BOUT[5]
DV_DATA12	ROUT[4]	BOUT[4]	ROUT[6]	BOUT[6]
DV_DATA11	ROUT[3]	BOUT[3]	ROUT[7]	BOUT[7]
DV_DATA10	GOUT[7]	GOUT[7]	GOUT[2]	GOUT[2]
DV_DATA9	GOUT[6]	GOUT[6]	GOUT[3]	GOUT[3]
DV_DATA8	GOUT[5]	GOUT[5]	GOUT[4]	GOUT[4]
DV_DATA7	GOUT[4]	GOUT[4]	GOUT[5]	GOUT[5]
DV_DATA6	GOUT[3]	GOUT[3]	GOUT[6]	GOUT[6]
DV_DATA5	GOUT[2]	GOUT[2]	GOUT[7]	GOUT[7]
DV_DATA4	BOUT[7]	ROUT[7]	BOUT[3]	ROUT[3]
DV_DATA3	BOUT[6]	ROUT[6]	BOUT[4]	ROUT[4]
DV_DATA2	BOUT[5]	ROUT[5]	BOUT[5]	ROUT[5]
DV_DATA1	BOUT[4]	ROUT[4]	BOUT[6]	ROUT[6]
DV_DATA0	BOUT[3]	ROUT[3]	BOUT[7]	ROUT[7]

(4) BT656/BT601 输入

如表 32.9 所示，BT656/BT601 输入时的外部输入图像信号 DV_DATA 引脚分配给内部信号 BTOUT。

从被分配的 BTOUT 信号生成 YCbCr 信号。

YCbCr 信号的生成请参照后述的 32.1.11。

表 32.9 BT656/BT601 输入时的 DV_DATA 引脚输入的位分配

INP_FORMAT[2:0]	3 ~ 4	3 ~ 4
INP_ENDIAN_ON	0	1
INP_SWAP_ON	0	0
DV_DATA7	BTOUT[7]	BTOUT[0]
DV_DATA6	BTOUT[6]	BTOUT[1]
DV_DATA5	BTOUT[5]	BTOUT[2]
DV_DATA4	BTOUT[4]	BTOUT[3]
DV_DATA3	BTOUT[3]	BTOUT[4]
DV_DATA2	BTOUT[2]	BTOUT[5]
DV_DATA1	BTOUT[1]	BTOUT[6]
DV_DATA0	BTOUT[0]	BTOUT[7]

32.1.8 BT601 的标准信号时序

BT601 格式的水平时序如图 32.3 和图 32.4 所示，垂直时序如图 32.5 和图 32.6 所示。

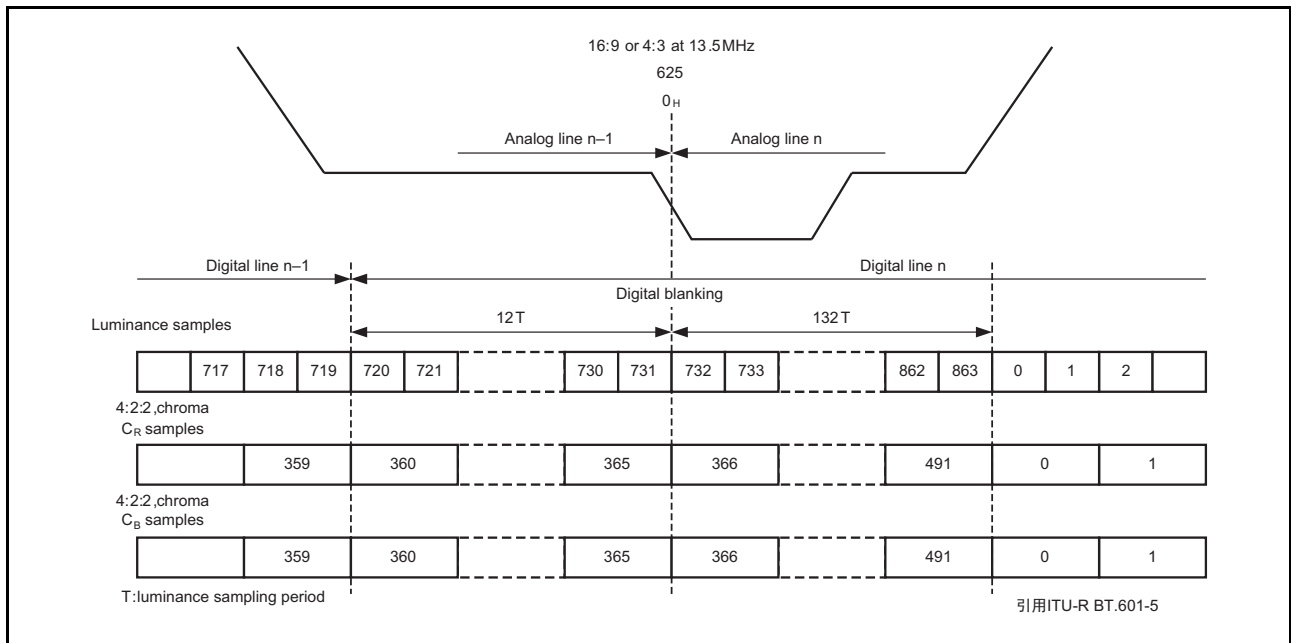


图 32.3 BT601 水平时序 (625 行 /50.00Hz)

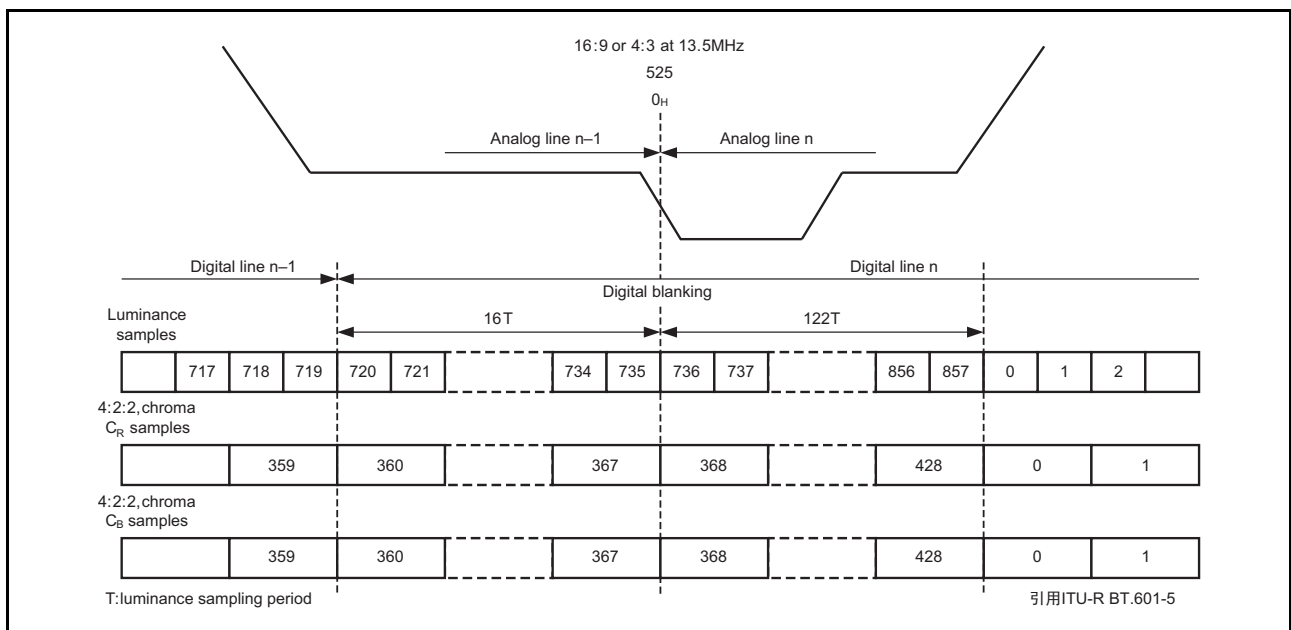


图 32.4 BT601 水平时序 (525 行 /59.94Hz)

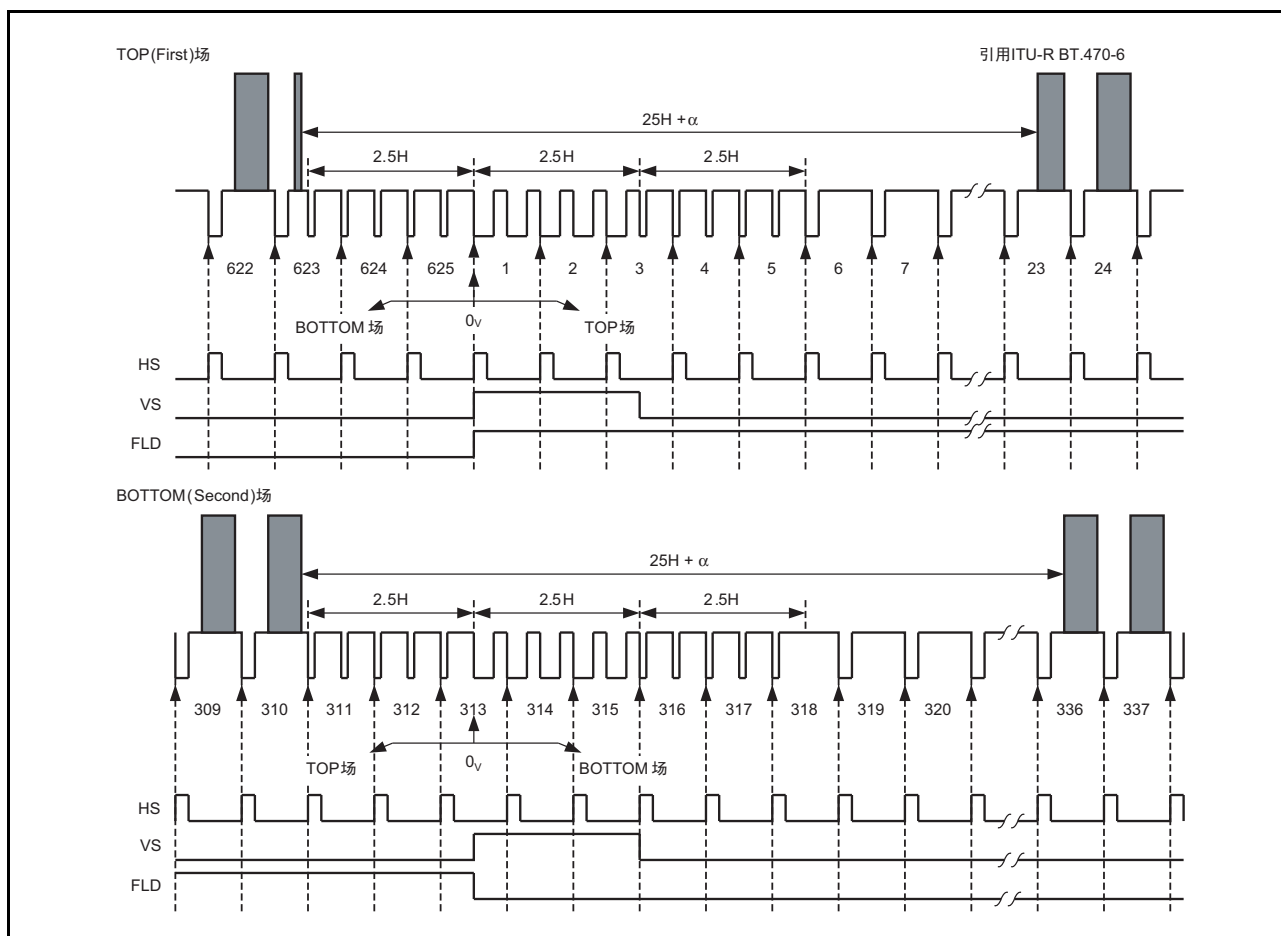


图 32.5 BT601 垂直时序 (625 行 / 50.00Hz)

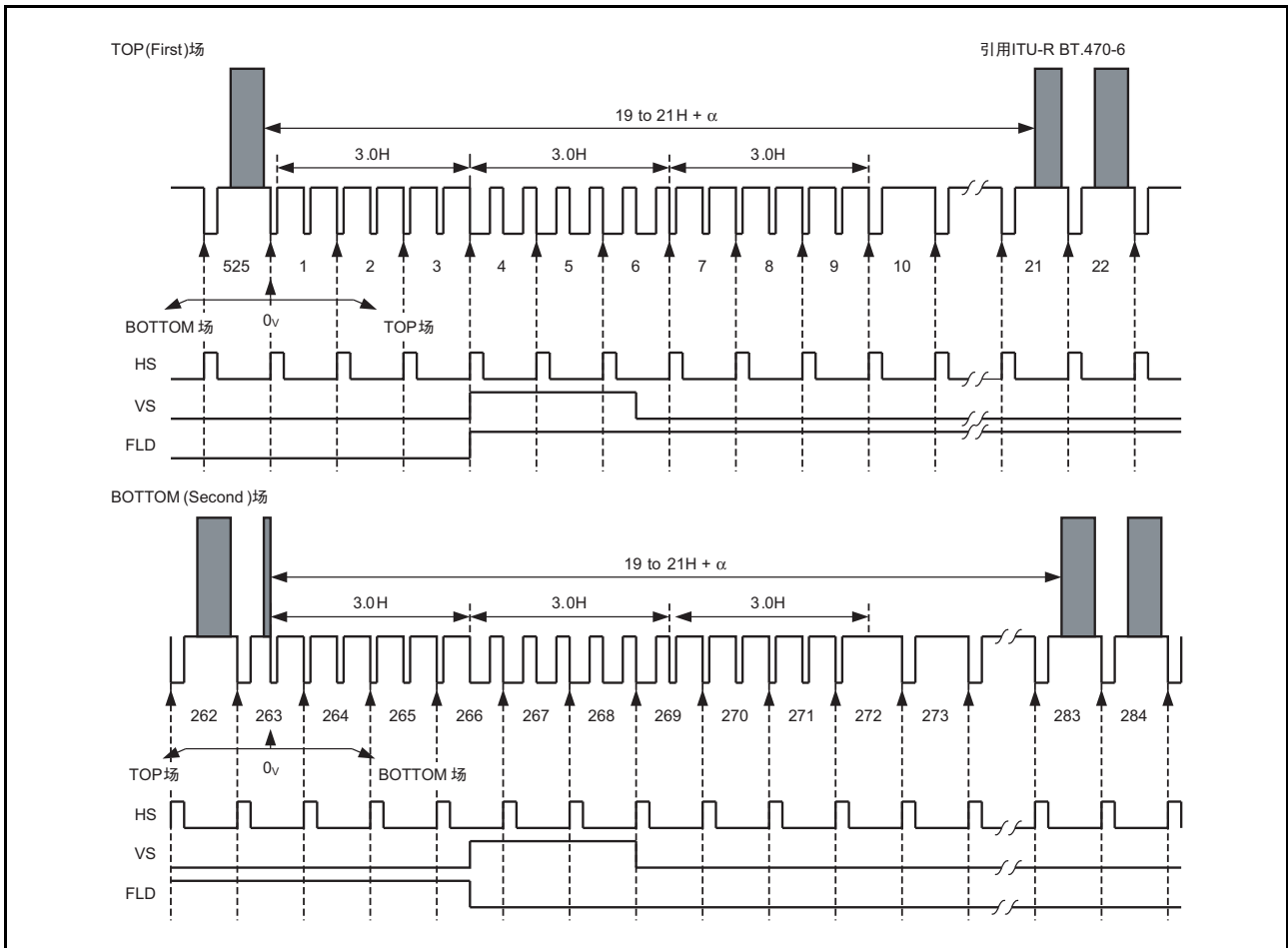


图 32.6 BT601 垂直时序 (525 行 / 59.94Hz)

32.1.9 BT656 的标准信号时序

BT656 格式的水平时序如图 32.7 和图 32.8 所示。

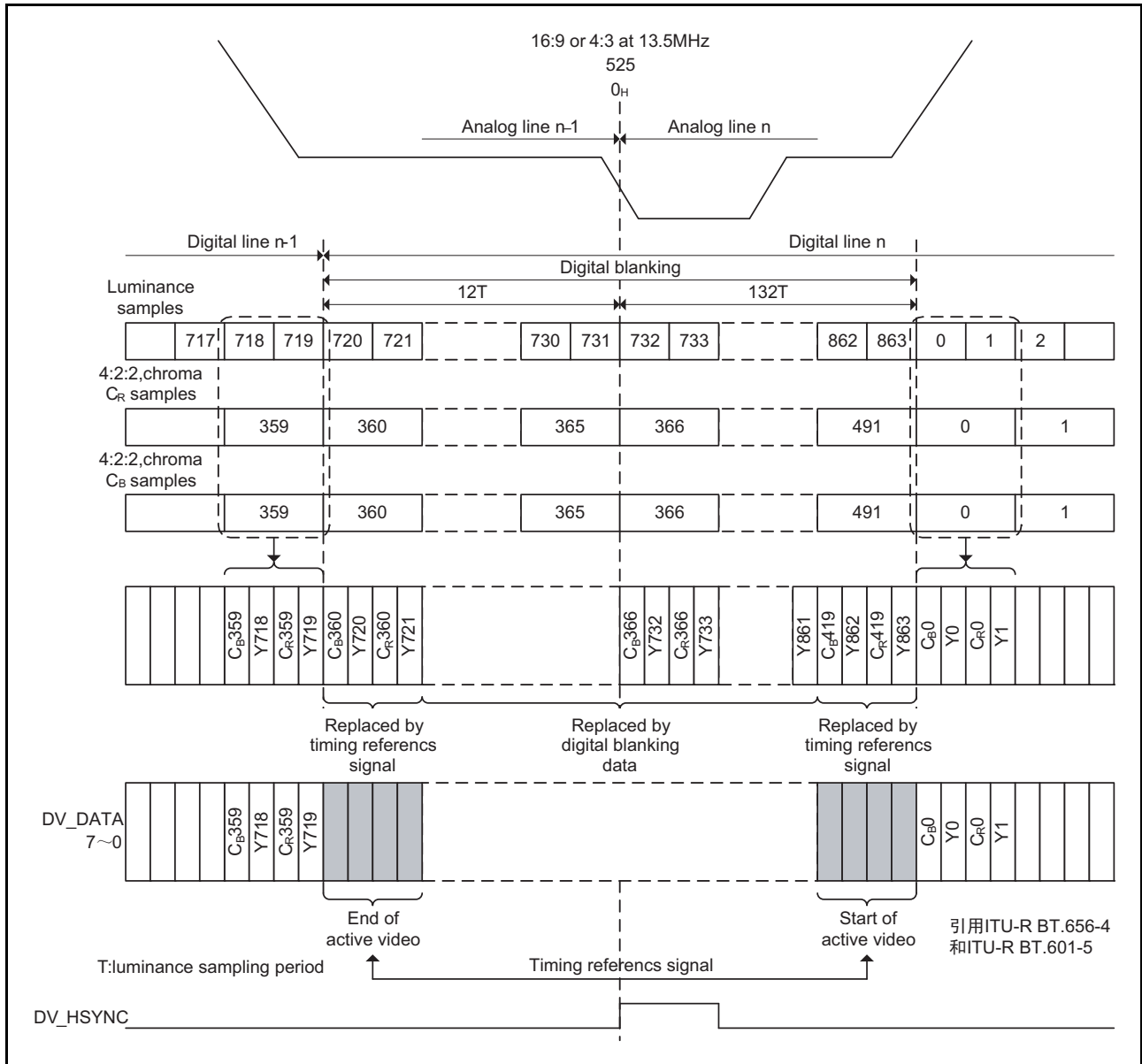


图 32.7 BT656 水平时序 (625 行 /50.00Hz)

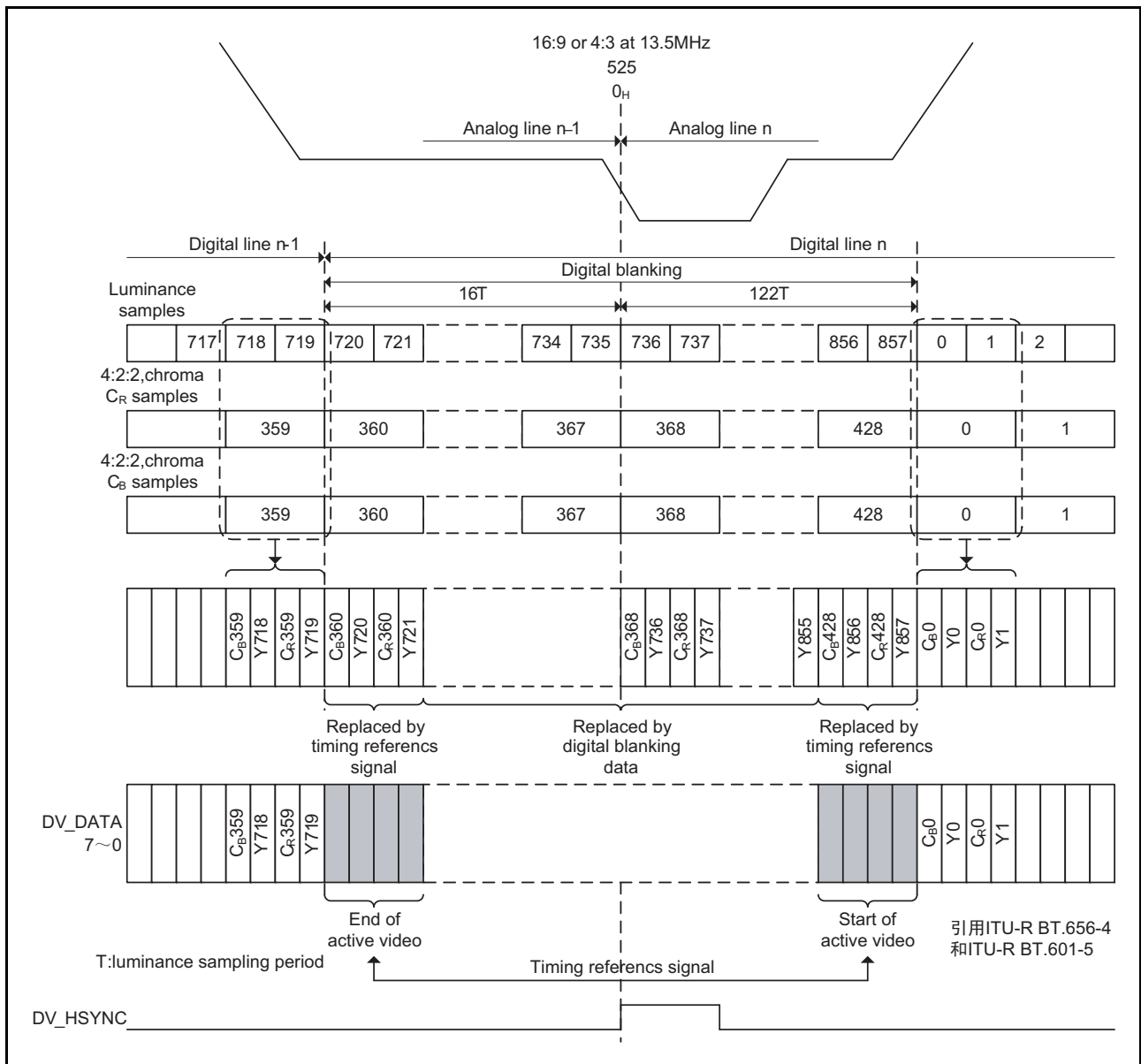


图 32.8 BT656 水平时序 (525 行 /59.94Hz)

32.1.10 BT656 的 SAV 码和 EAV 码

BT656 格式的 SAV 码和 EAV 码的插入时序如表 32.10 所示，位信息如表 32.11 和表 32.12 所示。此产品不参照表 32.12 所示的奇偶校验位 P3、P2、P1 和 P0。

表 32.10 SAV 码和 EAV 码的插入时序 (行)

		625	525
V-digital field blanking			
Field 1	Start (V=1)	Line 624	Line 1
	Finish (V=0)	Line 23	Line 20
Field 2	Start (V=1)	Line 311	Line 264
	Finish (V=0)	Line 336	Line 283
V-digital field blanking			
Field 1	F=0	Line 1	Line 4
Field 2	F=1	Line 313	Line 266

表 32.11 SAV 码和 EAV 码的位信息 (1)

Data bit number	1st word (FF)	2nd word (00)	3rd word (00)	4th word (XY)
7 (MSB)	1	0	0	1
6	1	0	0	F
5	1	0	0	V
4	1	0	0	H
3	1	0	0	P3
2	1	0	0	P2
1	1	0	0	P1
0	1	0	0	P0

F=0 during field 1

F=1 during field 2

V=0 elsewhere

V=1 during field blanking

H=0 is SAV

H=1 is EAV

表 32.12 SAV 码和 EAV 码的位信息 (2)

F	V	H	P3	P2	P1	P0
0	0	0	0	0	0	0
0	0	1	1	1	0	1
0	1	0	1	0	1	1
0	1	1	0	1	1	0
1	0	0	0	1	1	1
1	0	1	1	0	1	0
1	1	0	1	1	0	0
1	1	1	0	0	0	1

SAV 码和 EAV 码表如图 32.9 和图 32.10 所示。

		1H 期间																		
		EAV				H blank	SAV				有效区域									
		1	2	3	4		285	286	287	288	289	290	291	292	...	1725	1726	1727	1728	
Field1 (top)	1	FF	00	00	B6	Digital Blanking Data	FF	00	00	AB										
	:	FF	00	00	B6		FF	00	00	AB										
	22	FF	00	00	B6		FF	00	00	AB										
	23	FF	00	00	9D	有效像素数据区	FF	00	00	80	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719	
	:	FF	00	00	9D		FF	00	00	80	:									
	:	FF	00	00	9D		FF	00	00	80	:									
	:	FF	00	00	9D		FF	00	00	80	:									
	:	FF	00	00	9D		FF	00	00	80	:									
	:	FF	00	00	9D		FF	00	00	80	:									
	310	FF	00	00	9D		FF	00	00	80	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719	
311	FF	00	00	B6	Digital Blanking Data	FF	00	00	AB											
312	FF	00	00	B6		FF	00	00	AB											
Field2 (bottom)	313	FF	00	00	F1	Digital Blanking Data	FF	00	00	EC										
	:	FF	00	00	F1		FF	00	00	EC										
	335	FF	00	00	F1		FF	00	00	EC										
	336	FF	00	00	DA	有效像素数据区	FF	00	00	C7	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719	
	:	FF	00	00	DA		FF	00	00	C7	:									
	:	FF	00	00	DA		FF	00	00	C7	:									
	:	FF	00	00	DA		FF	00	00	C7	:									
	:	FF	00	00	DA		FF	00	00	C7	:									
	:	FF	00	00	DA		FF	00	00	C7	:									
	623	FF	00	00	DA		FF	00	00	C7	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719	
624	FF	00	00	F1	Digital Blanking Data	FF	00	00	EC											
625	FF	00	00	F1		FF	00	00	EC											

图 32.9 BT656 的 SAV 码和 EAV 码 (625 行 /50.00Hz)

		1H 期间																									
		EAV				H blank	SAV				有效区域																
		1	2	3	4		273	274	275	276	277	278	279	280	...	1713	1714	1715	1716								
Field2	1	FF	00	00	F1		FF	00	00	EC	Digital Blanking Data																
	2	FF	00	00	F1		FF	00	00	EC																	
	3	FF	00	00	F1		FF	00	00	EC																	
	4	FF	00	00	B6		FF	00	00	AB	Digital Blanking Data																
	:	FF	00	00	B6		FF	00	00	AB																	
	19	FF	00	00	B6		FF	00	00	AB																	
Field1 (top)	20	FF	00	00	9D		FF	00	00	80	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719								
	:	FF	00	00	9D		FF	00	00	80	:																
	:	FF	00	00	9D		FF	00	00	80	:																
	:	FF	00	00	9D		FF	00	00	80	:	有效像素数据区															
	:	FF	00	00	9D		FF	00	00	80	:																
	:	FF	00	00	9D		FF	00	00	80	:																
	263	FF	00	00	9D		FF	00	00	80	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719								
	264	FF	00	00	B6		FF	00	00	AB	Digital Blanking Data																
	265	FF	00	00	B6		FF	00	00	AB																	
	Field2 (bottom)	266	FF	00	00		F1		FF	00	00	EC	Digital Blanking Data														
:		FF	00	00	F1	FF	00		00	EC																	
282		FF	00	00	F1	FF	00		00	EC																	
283		FF	00	00	DA	FF	00		00	C7	Cb0	Y0									Cr0	Y1	...	Cb718	Y718	Cr718	Y719
:		FF	00	00	DA	FF	00		00	C7	:																
:		FF	00	00	DA	FF	00		00	C7	:	有效像素数据区															
:	FF	00	00	DA	FF	00	00	C7	:																		
:	FF	00	00	DA	FF	00	00	C7	:																		
525	FF	00	00	DA	FF	00	00	C7	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719										

图 32.10 BT656 的 SAV 码和 EAV 码 (525 行 /59.94Hz)

32.1.11 BT656/BT601 的设定

BT656 格式支持 525 行 59.94Hz 和 625 行 50.00Hz 的隔行扫描信号格式。

BT601 格式支持 525 行 59.94Hz 和 625 行 50.00Hz 的隔行扫描信号格式和逐行扫描信号格式。

对于 BT656 格式，525 行和 625 行的垂直同步信号时序不同。

通过 INP_F525_625 位设定运行模式。

表 32.13 BT656 运行模式的设定

寄存器名	位名	初始值	说明
INP_EXT_SYNC_CNT	INP_F525_625	0	外部输入系统的 BT656 输入时的行数设定 0: 525 行 1: 625 行

在输入 BT656/BT601 格式的隔行扫描信号时，通过 INP_FH50[9:0] 位设定垂直同步信号和水平同步信号的 1/2fH 相位时序。

INP_FH50[9:0] 位也用于垂直同步相位调整部，寄存器说明请参照表 32.17。

在输入 BT656 格式时，通过 INP_H_EDGE_SEL 位设定水平同步信号的基准。

表 32.14 BT656 水平同步信号的基准选择

寄存器名	位名	初始值	说明
INP_EXT_SYNC_CNT	INP_H_EDGE_SEL	0	外部输入系统的 BT656 水平同步信号的基准选择 0: EAV 基准 1: SAV 基准

在输入 BT656/BT601 格式时，从 DV_DATA 引脚输入，然后将被分配的內部信号 BTOUT[7:0] 展开为 24 位 YCbCr 信号。

通过 INP_H_POS[1:0] 位设定对水平同步信号基准的展开时序。

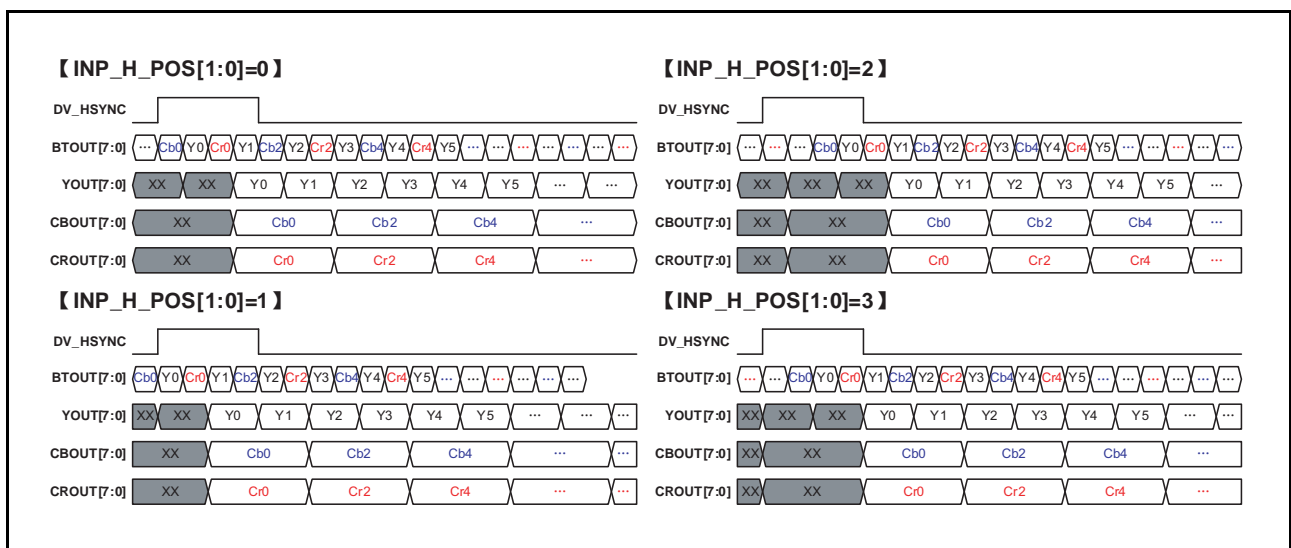


图 32.11 BT656/BT601 输入时的 YCbCr 数据展开

表 32.15 数据串的开始时序选择

寄存器名	位名	初始值	说明
INP_EXT_SYNC_CNT	INP_H_POS[1:0]	0	水平同步基准的 Y/Cb/Y/Cr 数据串开始时序设定 0: Cb/Y/Cr/Y 1: Y/Cr/Y/Cb 2: Cr/Y/Cb/Y 3: Y/Cb/Y/Cr

32.1.12 YCbCr444/RGB888/666/565 的外部输入时序

YCbCr444/RGB888/666/565 格式支持逐行扫描 YCbCr/RGB 信号。

同步信号的宽度 (H_SYNC 和 V_SYNC)、极性 (H_POL 和 V_POL)、有效期间的开始位置 (H_BP 和 V_BP) 和结束位置 (H_FP 和 V_FP) 以及图像宽度 (H_ACTIVE 和 V_ACTIVE) 的对应如表 32.16 所示。

表 32.16 YCbCr/RGB 信号的接受时序

项目	说明
外部输入时钟	外部输入时钟频率的最大值: 66.67[MHz]
垂直同步信号宽度 (V_SYNC)	垂直同步信号宽度的最小值: 至少 1[CLK]
垂直同步信号极性 (V_POL)	通过寄存器选择正极性和负极性。
垂直有效期间的开始位置 (V_BP)	从垂直同步基准到图像开始至少 4[行]
垂直有效期间的图像宽度 (V_ACTIVE)	垂直有效期间的最大值: 1024[行]
垂直有效期间的结束位置 (V_FP)	从图像结束到垂直同步基准为止至少 4[行]
水平同步信号宽度 (H_SYNC)	水平同步信号宽度的最小值: 至少 1[CLK]
水平同步信号极性 (H_POL)	通过寄存器选择正极性和负极性。
水平有效期间的开始位置 (H_BP)	从水平同步基准到图像开始至少 16[CLK]
水平有效期间的图像宽度 (H_ACTIVE)	水平有效期间的最大值: 1024[像素]
水平有效期间的结束位置 (H_FP)	从图像结束到水平同步基准为止至少 16[CLK]

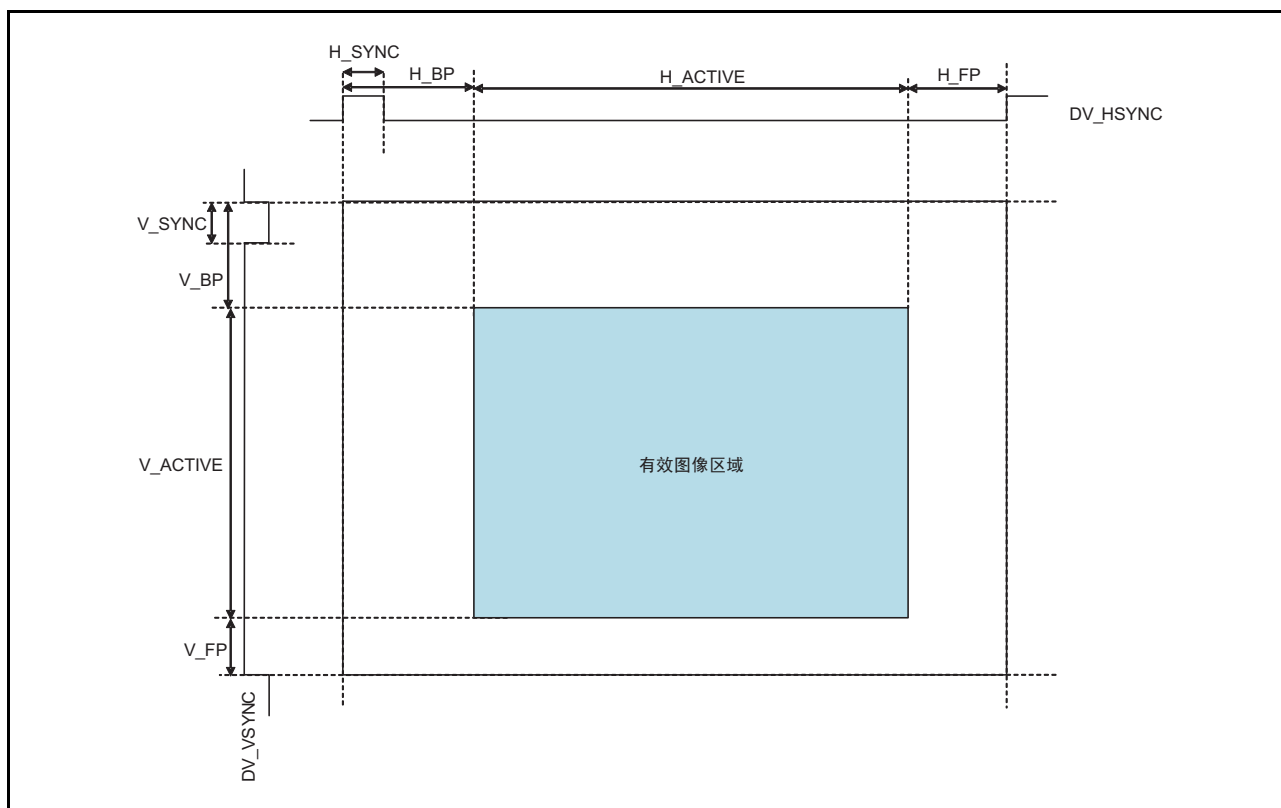


图 32.12 YCbCr/RGB 信号的接受时序图

32.1.13 场判别和垂直同步相位调整

在检测到输入信号的垂直同步信号和水平同步信号的相位后，进行隔行扫描信号的场判别。对于水平同步信号，如果在 ± 0.5 水平期间检测到垂直同步信号的基准，就判定为隔行扫描的 TOP 场；如果在 ± 0.5 水平期间外检测到垂直同步信号的基准，就判定为隔行扫描的 BOTTOM 场。

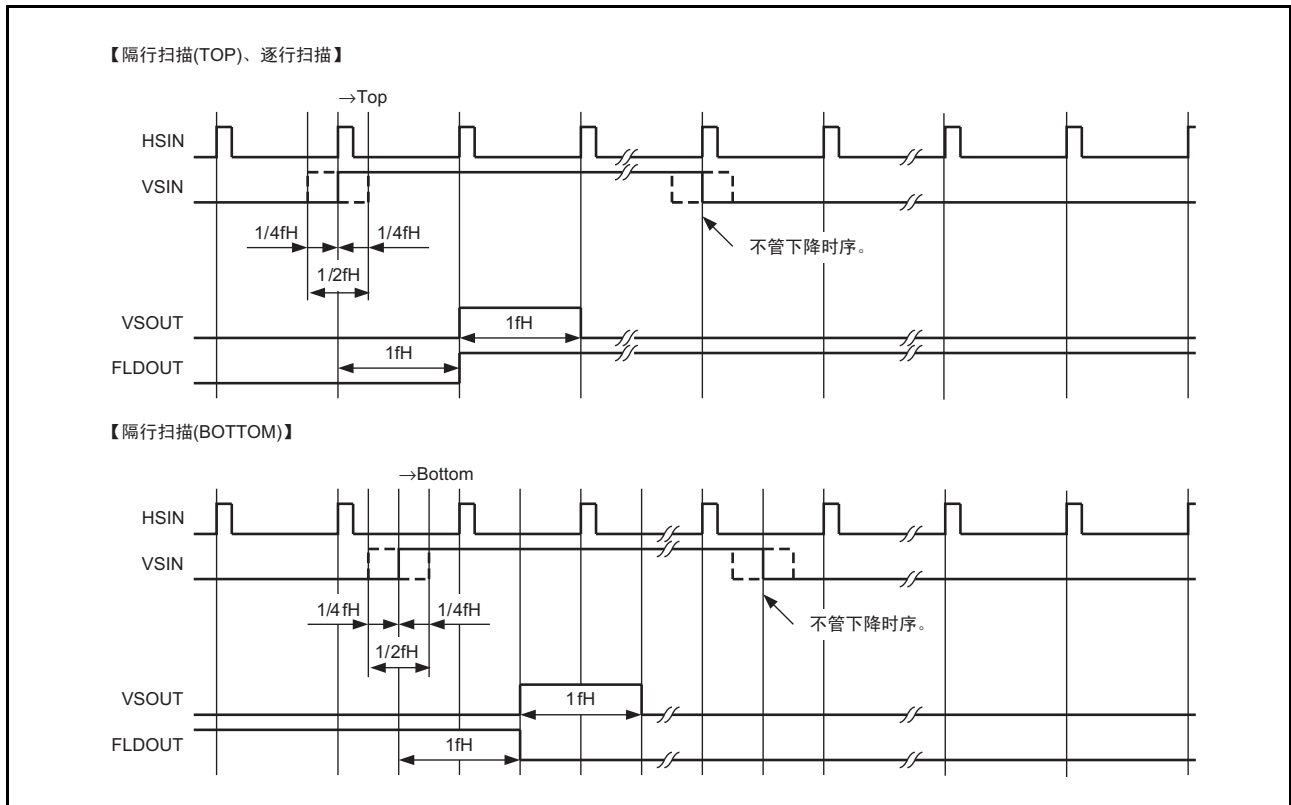


图 32.13 垂直同步相位调整

通过 INP_FH50[9:0] 位设定垂直同步信号的 $1/2fH$ 相位时序，通过 INP_FH25[9:0] 位设定垂直同步信号的 $1/4fH$ 相位时序。

表 32.17 垂直同步的相位时序设定

寄存器名	位名	初始值	说明
INP_VSYNC_PH_ADJ	INP_FH50[9:0]	858	垂直同步的 $1/2fH$ 相位时序设定 必须设定水平周期的 $1/2$ 个时钟周期。
INP_VSYNC_PH_ADJ	INP_FH25[9:0]	429	垂直同步的 $1/4fH$ 相位时序设定 必须设定水平周期的 $1/4$ 个时钟周期。

32.1.14 垂直同步信号的行延迟调整

垂直同步信号行延迟调整部能以行为单位延迟垂直同步信号和场判别信号。
如果在垂直前沿输入短的图像信号，就调整垂直前沿。

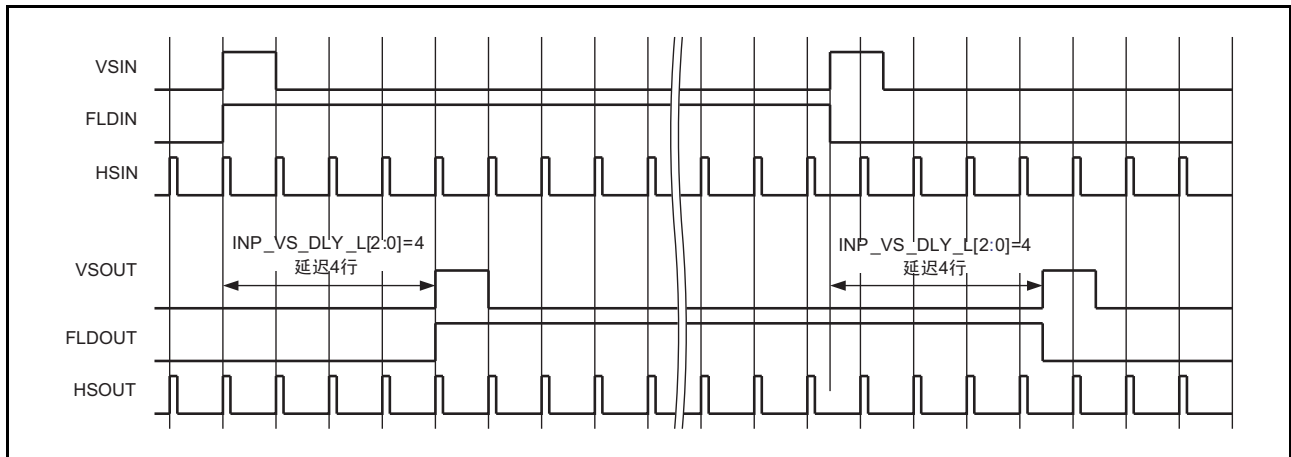


图 32.14 垂直同步信号的行延迟时序图

表 32.18 垂直同步信号的行延迟调整

寄存器名	位名	初始值	说明
INP_DLY_ADJ	INP_VS_DLY_L[2:0]	0	垂直同步信号和场判别信号的行延迟 延迟量: 0 ~ 7[行]

32.1.15 同步延迟调整

能以时钟为单位分别对垂直同步信号、水平同步信号和场判别信号进行延迟调整。
对输入的同步偏差引起的水平前沿欠缺余量进行调整。

表 32.19 同步信号的延迟调整

寄存器名	位名	初始值	说明
INP_DLY_ADJ	INP_VS_DLY[7:0]	0	垂直同步信号的延迟量 延迟量: 0 ~ 254[clk]
INP_DLY_ADJ	INP_HS_DLY[7:0]	0	水平同步信号的延迟量 延迟量: 0 ~ 254[clk]
INP_DLY_ADJ	INP_FLD_DLY[7:0]	0	场判别信号的延迟量 延迟量: 0 ~ 254[clk]

32.1.16 水平噪声的降低

参照水平像素来降低噪声。

通过噪声成分的频带 (TAP)、噪声电平 (阈值) 和噪声降低强度 (增益), 控制噪声的降低。

(1) 噪声成分的频带 (TAP) 设定

能通过 NR1D_Y_TAP[1:0] 位、NR1D_CB_TAP[1:0] 位和 NR1D_CR_TAP[1:0] 位分别选择 4 种噪声频带。

1 个相邻像素 (噪声降低 NR1D_Y/CB/CR_TAP=0) 的情况: $BPF_{(1)} = \frac{1}{4}(-1 \times Z_{(-1)}, 2 \times Z_{(0)}, -1 \times Z_{(+1)})$

2 个相邻像素 (噪声降低 NR1D_Y/CB/CR_TAP=1) 的情况: $BPF_{(2)} = \frac{1}{4}(-1 \times Z_{(-2)}, 2 \times Z_{(0)}, -1 \times Z_{(+2)})$

3 个相邻像素 (噪声降低 NR1D_Y/CB/CR_TAP=2) 的情况: $BPF_{(3)} = \frac{1}{4}(-1 \times Z_{(-3)}, 2 \times Z_{(0)}, -1 \times Z_{(+3)})$

4 个相邻像素 (噪声降低 NR1D_Y/CB/CR_TAP=3) 的情况: $BPF_{(4)} = \frac{1}{4}(-1 \times Z_{(-4)}, 2 \times Z_{(0)}, -1 \times Z_{(+4)})$

【注】 $Z_{(0)}$ 是噪声降低的对应像素, $Z_{(n)}$ 是从对应像素沿水平方向移动 n 后的像素。

(2) 噪声电平 (阈值) 的设定

将噪声检测量 (BPF 输出值) 的绝对值和 NR1D_Y_TH[6:0] 位、NR1D_CB_TH[6:0] 位、NR1D_CR_TH[6:0] 位进行比较, 当噪声检测量 > NR1D_Y/CB/CR_TH 时, 将噪声检测量的绝对值作为 NR1D_Y/CB/CR_TH (固定值)。

$ABS(BPF_{(n)}) \leq NR1D_Y/CB/CR_TH$ 时的噪声检测量绝对值: $NOISE_ABS=ABS(BPF_{(n)})$

$ABS(BPF_{(n)}) > NR1D_Y/CB/CR_TH$ 时的噪声检测量绝对值: $NOISE_ABS=NR1D_Y/CB/CR_TH$

(3) 噪声降低强度 (增益) 设定

通过 NR1D_Y_GAIN[1:0] 位、NR1D_CB_GAIN[1:0] 位和 NR1D_CR_GAIN[1:0] 位将噪声检测量绝对值乘增益, 然后对源信号进行反馈运算。

噪声检测量 ($BPF_{(n)}$) 为负 (-) 时的运算: $DOUT=DIN+NOISE_ABS \div 2^{(NR1D_Y/CB/CR_GAIN+1)}$

噪声检测量 ($BPF_{(n)}$) 为正 (+) 时的运算: $DOUT=DIN-NOISE_ABS \div 2^{(NR1D_Y/CB/CR_GAIN+1)}$

表 32.20 水平噪声的降低

寄存器名	位名	初始值	说明
IMGCNT_NR_CNT0	NR1D_MD	1	水平噪声降低的运行模式 0: G/B/R 模式 1: Y/Cb/Cr 模式
IMGCNT_NR_CNT0	NR1D_ON	0	噪声降低的 ON/OFF 控制 0: 噪声降低 OFF 1: 噪声降低 ON
IMGCNT_NR_CNT0	NR1D_Y_TAP[1:0]	0	Y/G 信号的 TAP 选择 0: 1 个相邻像素 1: 2 个相邻像素 2: 3 个相邻像素 3: 4 个相邻像素
IMGCNT_NR_CNT0	NR1D_Y_TH[6:0]	8	Y/G 信号的核化降噪最大值 (绝对值) 在噪声检测值 \leq NR1D_Y_TH 时实施核化降噪。 无符号: 0 ~ 127[LSB]
IMGCNT_NR_CNT0	NR1D_Y_GAIN[1:0]	3	Y/G 信号的噪声降低增益调整 0: 1/2 1: 1/4 2: 1/8 3: 1/16
IMGCNT_NR_CNT1	NR1D_CB_TAP[1:0]	0	Cb/B 信号的 TAP 选择 0: 1 个相邻像素 1: 2 个相邻像素 2: 3 个相邻像素 3: 4 个相邻像素
IMGCNT_NR_CNT1	NR1D_CB_TH[6:0]	8	Cb/B 信号的核化降噪最大值 (绝对值) 在噪声检测值 \leq NR1D_C_TH 时实施核化降噪。 无符号: 0 ~ 127[LSB]
IMGCNT_NR_CNT1	NR1D_CB_GAIN[1:0]	3	Cb/B 信号的噪声降低增益调整 0: 1/2 1: 1/4 2: 1/8 3: 1/16
IMGCNT_NR_CNT1	NR1D_CR_TAP[1:0]	0	Cr/R 信号的 TAP 选择 0: 1 个相邻像素 1: 2 个相邻像素 2: 3 个相邻像素 3: 4 个相邻像素
IMGCNT_NR_CNT1	NR1D_CR_TH[6:0]	8	Cr/R 信号的核化降噪最大值 (绝对值) 在噪声检测值 \leq NR1D_C_TH 时实施核化降噪。 无符号: 0 ~ 127[LSB]
IMGCNT_NR_CNT1	NR1D_CR_GAIN[1:0]	3	Cr/R 信号的噪声降低增益调整 0: 1/2 1: 1/4 2: 1/8 3: 1/16

32.1.17 色彩矩阵

色彩矩阵能进行各输入信号的偏移调整和 9 轴的增益调整。因此，能进行亮度调整、增益调整以及 YCbCr 和 GBR 的相互转换。

(1) GBR→GBR 转换

$$\begin{aligned} YGIN_A &= YGIN + IMGCNT_MTX_YG - 128 \\ CBBIN_A &= CBBIN + IMGCNT_MTX_B - 128 \\ CRRIN_A &= CRRIN + IMGCNT_MTX_R - 128 \\ YGOUT &= (IMGCNT_MTX_GG \times YGIN_A + IMGCNT_MTX_GB \times CBBIN_A + IMGCNT_MTX_GR \times CRRIN_A) \\ &\quad \div 256 \\ CBBOUT &= (IMGCNT_MTX_BG \times YGIN_A + IMGCNT_MTX_BB \times CBBIN_A + IMGCNT_MTX_BR \times CRRIN_A) \\ &\quad \div 256 \\ CRROUT &= (IMGCNT_MTX_RG \times YGIN_A + IMGCNT_MTX_RB \times CBBIN_A + IMGCNT_MTX_RR \times CRRIN_A) \\ &\quad \div 256 \end{aligned}$$

(2) GBR→YCbCr 转换

$$\begin{aligned} YGIN_A &= YGIN + IMGCNT_MTX_YG - 128 \\ CBBIN_A &= CBBIN + IMGCNT_MTX_B - 128 \\ CRRIN_A &= CRRIN + IMGCNT_MTX_R - 128 \\ YGOUT &= (IMGCNT_MTX_GG \times YGIN_A + IMGCNT_MTX_GB \times CBBIN_A + IMGCNT_MTX_GR \times CRRIN_A) \\ &\quad \div 256 \\ CBBOUT &= (IMGCNT_MTX_BG \times YGIN_A + IMGCNT_MTX_BB \times CBBIN_A + IMGCNT_MTX_BR \times CRRIN_A) \\ &\quad \div 256 + 128 \\ CRROUT &= (IMGCNT_MTX_RG \times YGIN_A + IMGCNT_MTX_RB \times CBBIN_A + IMGCNT_MTX_RR \times CRRIN_A) \\ &\quad \div 256 + 128 \end{aligned}$$

表 32.21 SMPTE 293M 的矩阵系数 (标准值)

	YGIN		CBBIN		CRRIN	
	系数	设定值	系数	设定值	系数	设定值
YGOUT	0.587	IMGCNT_MTX_GG =150	0.114	IMGCNT_MTX_GB =29	0.299	IMGCNT_MTX_GR =77
CBBOUT	-0.331	IMGCNT_MTX_BG =1963	0.500	IMGCNT_MTX_BB =128	-0.169	IMGCNT_MTX_BR =2005
CRROUT	-0.419	IMGCNT_MTX_RG =1941	-0.081	IMGCNT_MTX_RB =2027	0.500	IMGCNT_MTX_RR =128

(3) YCbCr→GBR 转换

$$YGIN_A = YGIN + IMGCNT_MTX_YG - 128$$

$$CBBIN_A = CBBIN - 128$$

$$CRRIN_A = CRRIN - 128$$

$$YGOUT = (IMGCNT_MTX_GG \times YGIN_A + IMGCNT_MTX_GB \times CBBIN_A + IMGCNT_MTX_GR \times CRRIN_A) \div 256$$

$$CBBOUT = (IMGCNT_MTX_BG \times YGIN_A + IMGCNT_MTX_BB \times CBBIN_A + IMGCNT_MTX_BR \times CRRIN_A) \div 256$$

$$CRROUT = (IMGCNT_MTX_RG \times YGIN_A + IMGCNT_MTX_RB \times CBBIN_A + IMGCNT_MTX_RR \times CRRIN_A) \div 256$$

表 32.22 SMPTE 293M 的矩阵系数 (标准值)

	YGIN		CBBIN		CRRIN	
	系数	设定值	系数	设定值	系数	设定值
YGOUT	1.000	IMGCNT_MTX_GG =256	-0.344	IMGCNT_MTX_GB =1960	-0.714	IMGCNT_MTX_GR =1865
CBBOUT	1.000	IMGCNT_MTX_BG =256	1.772	IMGCNT_MTX_BB =454	0.000	IMGCNT_MTX_BR =0
CRROUT	1.000	IMGCNT_MTX_RG =256	0.000	IMGCNT_MTX_RB =0	1.402	IMGCNT_MTX_RR =359

(4) YCbCr→YCbCr 转换

$$YGIN_A = YGIN + IMGCNT_MTX_YG - 128$$

$$CBBIN_A = CBBIN - 128$$

$$CRRIN_A = CRRIN - 128$$

$$YGOUT = (IMGCNT_MTX_GG \times YGIN_A + IMGCNT_MTX_GB \times CBBIN_A + IMGCNT_MTX_GR \times CRRIN_A) \div 256$$

$$CBBOUT = (IMGCNT_MTX_BG \times YGIN_A + IMGCNT_MTX_BB \times CBBIN_A + IMGCNT_MTX_BR \times CRRIN_A) \div 256 + 128$$

$$CRROUT = (IMGCNT_MTX_RG \times YGIN_A + IMGCNT_MTX_RB \times CBBIN_A + IMGCNT_MTX_RR \times CRRIN_A) \div 256 + 128$$

表 32.23 YCbCr→GBR 的转换

寄存器名	位名	初始值	说明
IMGCNT_MTX_MODE	IMGCNT_MTX_MD [1:0]	3	运行模式 0: GBR => GBR 1: GBR => YCbCr 2: YCbCr => GBR 3: YCbCr => YCbCr
IMGCNT_MTX_YG_ADJ0	IMGCNT_MTX_YG [7:0]	128	Y/G 信号的偏移 (DC) 调整 无符号 (0 (-128) ~ 128 (0) ~ 255 (+127) [LSB])
IMGCNT_MTX_CBB_ADJ0	IMGCNT_MTX_B [7:0]	128	B 信号的偏移 (DC) 调整 无符号 (0 (-128) ~ 128 (0) ~ 255 (+127) [LSB])
IMGCNT_MTX_CRR_ADJ0	IMGCNT_MTX_R [7:0]	128	R 信号的偏移 (DC) 调整 无符号 (0 (-128) ~ 128 (0) ~ 255 (+127) [LSB])
IMGCNT_MTX_YG_ADJ0	IMGCNT_MTX_GG [10:0]	256	Y/G 信号输出的 Y/G 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB] =1.0[倍])
IMGCNT_MTX_YG_ADJ1	IMGCNT_MTX_GB [10:0]	0	Y/G 信号输出的 Cb/B 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB] =1.0[倍])
IMGCNT_MTX_YG_ADJ1	IMGCNT_MTX_GR [10:0]	0	Y/G 信号输出的 Cr/R 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB] =1.0[倍])
IMGCNT_MTX_CBB_ADJ0	IMGCNT_MTX_BG [10:0]	0	Cb/B 信号输出的 Y/G 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB] =1.0[倍])
IMGCNT_MTX_CBB_ADJ1	IMGCNT_MTX_BB [10:0]	256	Cb/B 信号输出的 Cb/B 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB] =1.0[倍])
IMGCNT_MTX_CBB_ADJ1	IMGCNT_MTX_BR [10:0]	0	Cb/B 信号输出的 Cr/R 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB] =1.0[倍])
IMGCNT_MTX_CRR_ADJ0	IMGCNT_MTX_RG [10:0]	0	Cr/R 信号输出的 Y/G 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB] =1.0[倍])
IMGCNT_MTX_CRR_ADJ1	IMGCNT_MTX_RB [10:0]	0	Cr/R 信号输出的 Cb/B 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB] =1.0[倍])
IMGCNT_MTX_CRR_ADJ1	IMGCNT_MTX_RR [10:0]	256	Cr/R 信号输出的 Cr/R 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB] =1.0[倍])

32.2 寄存器说明

寄存器结构如表 32.24 和表 32.25 所示。

【说明寄存器时的符号】

初始值 : 上电复位后的寄存器值

— : 不定值

R/W : 可读写。能读到写入值。

R/WC0 : 可读写。如果写“0”，该位就被初始化，但是忽视写“1”。

R/WC1 : 可读写。如果写“1”，该位就被初始化，但是忽视写“0”。

R : 只能读。写入值总是“0”。

—/W : 只能写。读取值为不定值。

表 32.24 输入控制部的寄存器结构

名称	略称	R/W	初始值	地址	存取长度
外部输入部的寄存器更新控制寄存器	INP_UPDATE	R/WC1	H'0000 0000	H'FFFF 7400	32/16
输入选择控制寄存器	INP_SEL_CNT	R/W	H'0000 0000	H'FFFF 7404	32/16
外部输入同步信号控制寄存器	INP_EXT_SYNC_CNT	R/W	H'0000 0000	H'FFFF 7408	32/16
垂直同步信号相位调整寄存器	INP_VSYNC_PH_ADJ	R/W	H'035A 01AD	H'FFFF 740C	32/16
同步信号延迟调整寄存器	INP_DLY_ADJ	R/W	H'0000 0000	H'FFFF 7410	32/16

表 32.25 图像质量调整部的寄存器结构

名称	略称	R/W	初始值	地址	存取长度
图像质量调整部的寄存器更新控制寄存器	IMGCNT_UPDATE	R/WC1	H'0000 0000	H'FFFF 7480	32/16
NR 控制寄存器 0	IMGCNT_NR_CNT0	R/W	H'0010 0803	H'FFFF 7484	32/16
NR 控制寄存器 1	IMGCNT_NR_CNT1	R/W	H'0803 0803	H'FFFF 7488	32/16
图像质量调整部的矩阵模式寄存器	IMGCNT_MTX_MODE	R/W	H'0000 0003	H'FFFF 74A0	32/16
图像质量调整部的矩阵 YG 调整寄存器 0	IMGCNT_MTX_YG_ADJ0	R/W	H'0080 0100	H'FFFF 74A4	32/16
图像质量调整部的矩阵 YG 调整寄存器 1	IMGCNT_MTX_YG_ADJ1	R/W	H'0000 0000	H'FFFF 74A8	32/16
图像质量调整部的矩阵 CBB 调整寄存器 0	IMGCNT_MTX_CBB_ADJ0	R/W	H'0080 0000	H'FFFF 74AC	32/16
图像质量调整部的矩阵 CBB 调整寄存器 1	IMGCNT_MTX_CBB_ADJ1	R/W	H'0100 0000	H'FFFF 74B0	32/16
图像质量调整部的矩阵 CRR 调整寄存器 0	IMGCNT_MTX_CRR_ADJ0	R/W	H'0080 0000	H'FFFF 74B4	32/16
图像质量调整部的矩阵 CRR 调整寄存器 1	IMGCNT_MTX_CRR_ADJ1	R/W	H'0000 0100	H'FFFF 74B8	32/16

32.2.1 外部输入部的寄存器更新控制寄存器 (INP_UPDATE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	INP_EXT_UPDATE	—	—	—	INP_IMG_UPDATE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1

位	位名	初始值	R/W	说明
31 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	INP_EXT_UPDATE	0	R/WC1	外部输入部的寄存器更新 0: 不更新寄存器 1: 更新寄存器
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	INP_IMG_UPDATE	0	R/WC1	同步信号调整部的寄存器更新 0: 不更新寄存器 1: 更新寄存器

32.2.2 输入选择控制寄存器 (INP_SEL_CNT)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	INP_SEL	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	INP_FORMAT[2:0]	—	—	—	—	INP_PXD_EDGE	—	—	—	—	INP_VS_EDGE	—	—	—	INP_HS_EDGE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 21	—	全 0	R	保留位 读写值都为“0”。
20	INP_SEL	0	R/W	输入选择 0: 视频解码器的输出 1: 外部输入引脚
19 ~ 15	—	全 0	R	保留位 读写值都为“0”。
14 ~ 12	INP_FORMAT [2:0]	0	R/W	外部输入的格式选择 0: YCbCr444、RGB888 1: RGB666 2: RGB565 3: BT656 4: BT601 5 ~ 7: 禁止设定
11 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	INP_PXD_EDGE	0	R/W	外部输入图像信号 DV_DATA23~0 的输入段捕捉时钟边沿选择 0: 上升沿 1: 下降沿
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	INP_VS_EDGE	0	R/W	外部输入垂直同步信号 DV_VSYNC 的输入段捕捉时钟边沿选择 0: 上升沿 1: 下降沿
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	INP_HS_EDGE	0	R/W	外部输入水平同步信号 DV_HSYNC 的输入段捕捉时钟边沿选择 0: 上升沿 1: 下降沿

【注】 当外部输入部的寄存器更新控制寄存器 (INP_UPDATE) 的 INP_EXT_UPDATE 位为“1”时, 更新 INP_FORMAT 位、INP_PXD_EDGE 位、INP_VS_EDGE 位和 INP_HS_EDGE 位。如果设定 INP_SEL 位, 就更新此位。

32.2.3 外部输入同步信号控制寄存器 (INP_EXT_SYNC_CNT)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	INP_ENDIAN_ON	—	—	—	INP_SWAP_ON	—	—	—	INP_VS_INV	—	—	—	INP_HS_INV
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	INP_H_EDGE_SEL	—	—	—	INP_F525_625	—	—	—	INP_H_POS[1:0]
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 29	—	全 0	R	保留位 读写值都为“0”。
28	INP_ENDIAN_ON	0	R/W	外部输入的位序变更 ON/OFF 控制 0: OFF 1: ON
27 ~ 25	—	全 0	R	保留位 读写值都为“0”。
24	INP_SWAP_ON	0	R/W	外部输入的 B/R 信号交换 ON/OFF 控制 0: OFF 1: ON
23 ~ 21	—	全 0	R	保留位 读写值都为“0”。
20	INP_VS_INV	0	R/W	外部输入垂直同步信号 DV_VSYNC 的反相控制 0: 正相 (正极性) 1: 反相 (负极性)
19 ~ 17	—	全 0	R	保留位 读写值都为“0”。
16	INP_HS_INV	0	R/W	外部输入水平同步信号 DV_HSYNC 的反相控制 0: 正相 (正极性) 1: 反相 (反极性)
15 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	INP_H_EDGE_SEL	0	R/W	外部输入 BT656 水平同步信号的基准选择 0: EAV 基准 1: SAV 基准
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	INP_F525_625	0	R/W	外部输入的 BT656 输入时的行数设定 0: 525 行 1: 625 行
3、2	—	全 0	R	保留位 读写值都为“0”。

位	位名	初始值	R/W	说明
1、0	INP_H_POS[1:0]	0	R/W	外部输入的 BT656/601 输入时的水平同步基准的 Y/Cb/Y/Cr 数据串开始时序设定 0: Cb/Y/Cr/Y 1: Y/Cr/Y/Cb 2: Cr/Y/Cb/Y 3: Y/Cb/Y/Cr

【注】 在全部外部输入部的寄存器更新控制寄存器 (INP_UPDATE) 的 INP_EXT_UPDATE 位为 “1” 时更新此寄存器。

32.2.4 垂直同步信号相位调整寄存器 (INP_VSYNC_PH_ADJ)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16			
	—	—	—	—	—	—	INP_FH50[9:0]									—	—	—	—
初始值:	0	0	0	0	0	0	1	1	0	1	0	1	1	0	1	0			
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W			
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
	—	—	—	—	—	—	INP_FH25[9:0]									—	—	—	—
初始值:	0	0	0	0	0	0	0	1	1	0	1	0	1	1	0	1			
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W			

位	位名	初始值	R/W	说明
31 ~ 26	—	全 0	R	保留位 读写值都为 “0”。
25 ~ 16	INP_FH50[9:0]	858	R/W	垂直同步的 1/2fH 相位时序设定 必须设定水平周期的 1/2 个时钟周期。
15 ~ 10	—	全 0	R	保留位 读写值都为 “0”。
9 ~ 0	INP_FH25[9:0]	429	R/W	垂直同步的 1/4fH 相位时序设定 必须设定水平周期的 1/4 个时钟周期。

【注】 当外部输入部的寄存器更新控制寄存器 (INP_UPDATE) 的 INP_EXT_UPDATE 位和 INP_IMG_UPDATE 位都为 “1” 时, 更新 INP_FH50[9:0] 位; 当外部输入部的寄存器更新控制寄存器 (INP_UPDATE) 的 INP_IMG_UPDATE 位为 “1” 时, 更新 INP_FH25[9:0] 位。

32.2.5 同步信号延迟调整寄存器 (INP_DLY_ADJ)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	INP_VS_DLY_L[2:0]			INP_FLD_DLY[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INP_VS_DLY[7:0]								INP_HS_DLY[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 24	INP_VS_DLY_L [2:0]	0	R/W	垂直同步信号和场判别信号的行延迟量 延迟量: 0 ~ 7[行]
23 ~ 16	INP_FLD_DLY [7:0]	0	R/W	场判别信号的延迟量 延迟量: 0 ~ 254[clk]
15 ~ 8	INP_VS_DLY [7:0]	0	R/W	垂直同步信号的延迟量 延迟量: 0 ~ 254[clk]
7 ~ 0	INP_HS_DLY [7:0]	0	R/W	水平同步信号的延迟量 延迟量: 0 ~ 254[clk]

【注】 在全部外部输入部的寄存器更新控制寄存器 (INP_UPDATE) 的 INP_IMG_UPDATE 位为“1”时更新此寄存器。

32.2.6 图像质量调整部的寄存器更新控制寄存器 (IMGCNT_UPDATE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IMGCNT _VEN
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

位	位名	初始值	R/W	说明
31 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	IMGCNT_VEN	0	R/WC1	图像质量调整部的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器

32.2.7 NR 控制寄存器 0 (IMGCNT_NR_CNT0)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	NR1D_MD	—	—	—	NR1D_ON
初始值:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	NR1D_Y_TH[6:0]						—	—	NR1D_Y_TAP[1:0]			—	—	NR1D_Y_GAIN[1:0]	
初始值:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 21	—	全 0	R	保留位 读写值都为“0”。
20	NR1D_MD	1	R/W	水平噪声降低的运行模式 0: G/B/R 模式 1: Y/Cb/Cr 模式
19 ~ 17	—	全 0	R	保留位 读写值都为“0”。
16	NR1D_ON	0	R/W	噪声降低的 ON/OFF 控制 0: 噪声降低 OFF 1: 噪声降低 ON
15	—	0	R	保留位 读写值都为“0”。
14 ~ 8	NR1D_Y_TH[6:0]	8	R/W	Y 信号和 G 信号的核化降噪最大值 (绝对值) 在噪声检测值 ≤ NR1D_Y_TH 时实施核化降噪。 无符号: 0 ~ 127[LSB]
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	NR1D_Y_TAP [1:0]	0	R/W	Y 信号和 G 信号的 TAP 选择 0: 1 个相邻像素 1: 2 个相邻像素 2: 3 个相邻像素 3: 4 个相邻像素
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	NR1D_Y_GAIN [1:0]	3	R/W	Y 信号和 G 信号的噪声降低增益调整 0: 1/2 1: 1/4 2: 1/8 3: 1/16

【注】 在全部图像质量调整部的寄存器更新控制寄存器 (IMGCNT_UPDATE) 的 IMGCNT_VEN 位为“1”时更新此寄存器。

32.2.8 NR 控制寄存器 1 (IMGCNT_NR_CNT1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	NR1D_CB_TH[6:0]						—	—	NR1D_CB_TAP[1:0]		—	—	NR1D_CB_GAIN[1:0]		
初始值:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	NR1D_CR_TH[6:0]						—	—	NR1D_CR_TAP[1:0]		—	—	NR1D_CR_GAIN[1:0]		
初始值:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
31	—	0	R	保留位 读写值都为“0”。
30 ~ 24	NR1D_CB_TH [6:0]	8	R/W	Cb/B 信号的核化降噪最大值 (绝对值) 在噪声检测值 ≤ NR1D_CB_TH 时实施核化降噪。 无符号: 0 ~ 127[LSB]
23、22	—	全 0	R	保留位 读写值都为“0”。
21、20	NR1D_CB_TAP [1:0]	0	R/W	Cb/B 信号的 TAP 选择 0: 1 个相邻像素 1: 2 个相邻像素 2: 3 个相邻像素 3: 4 个相邻像素
19、18	—	全 0	R	保留位 读写值都为“0”。
17、16	NR1D_CB_GAIN [1:0]	3	R/W	Cb/B 信号的噪声降低增益调整 0: 1/2 1: 1/4 2: 1/8 3: 1/16
15	—	0	R	保留位 读写值都为“0”。
14 ~ 8	NR1D_CR_TH [6:0]	8	R/W	Cr/R 信号的核化降噪的最大值 (绝对值) 在噪声检测值 ≤ NR1D_CR_TH 时实施核化降噪。 无符号: 0 ~ 127[LSB]
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	NR1D_CR_TAP [1:0]	0	R/W	Cr/R 信号的 TAP 选择 0: 1 个相邻像素 1: 2 个相邻像素 2: 3 个相邻像素 3: 4 个相邻像素
3、2	—	全 0	R	保留位 读写值都为“0”。

位	位名	初始值	R/W	说明
1、0	NR1D_CR_GAIN [1:0]	3	R/W	Cr/R 信号的噪声降低增益调整 0: 1/2 1: 1/4 2: 1/8 3: 1/16

【注】 在全部图像质量调整部的寄存器更新控制寄存器 (IMGCNT_UPDATE) 的 IMGCNT_VEN 位为“1”时更新此寄存器。

32.2.9 图像质量调整部的矩阵模式寄存器 (IMGCNT_MTX_MODE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IMGCNT_MTX_MD[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明
31~2	—	全 0	R	保留位 读写值都为“0”。
1、0	IMGCNT_MTX_MD[1:0]	3	R/W	运行模式 0: GBR => GBR 1: GBR => YCbCr 2: YCbCr => GBR 3: YCbCr => YCbCr

【注】 在全部图像质量调整部的寄存器更新控制寄存器 (IMGCNT_UPDATE) 的 IMGCNT_VEN 位为“1”时更新此寄存器。

32.2.10 图像质量调整部的矩阵 YG 调整寄存器 0 (IMGCNT_MTX_YG_ADJ0)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	IMGCNT_MTX_YG[7:0]							
初始值:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	IMGCNT_MTX_GG[10:0]										
初始值:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23 ~ 16	IMGCNT_MTX_YG[7:0]	128	R/W	Y/G 信号的偏移 (DC) 调整 无符号 (0 (-128) ~ 128 (0) ~ 255 (+127) [LSB])
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	IMGCNT_MTX_GG[10:0]	256	R/W	Y/G 信号输出的 Y/G 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])

【注】 在全部图像质量调整部的寄存器更新控制寄存器 (IMGCNT_UPDATE) 的 IMGCNT_VEN 位为“1”时更新此寄存器。

32.2.11 图像质量调整部的矩阵 YG 调整寄存器 1 (IMGCNT_MTX_YG_ADJ1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	IMGCNT_MTX_GB[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	IMGCNT_MTX_GR[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	IMGCNT_MTX_GB[10:0]	0	R/W	Y/G 信号输出的 Cb/B 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	IMGCNT_MTX_GR[10:0]	0	R/W	Y/G 信号输出的 Cr/R 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])

【注】 在全部图像质量调整部的寄存器更新控制寄存器 (IMGCNT_UPDATE) 的 IMGCNT_VEN 位为“1”时更新此寄存器。

32.2.12 图像质量调整部的矩阵 CBB 调整寄存器 0 (IMGCNT_MTX_CBB_ADJ0)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	IMGCNT_MTX_B[7:0]							
初始值:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	IMGCNT_MTX_BG[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23 ~ 16	IMGCNT_MTX_B[7:0]	128	R/W	Cb/B 信号的偏移 (DC) 调整 无符号 (0 (-128) ~ 128 (0) ~ 255 (+127) [LSB])
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	IMGCNT_MTX_BG[10:0]	0	R/W	Cb/B 信号输出的 Y/G 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])

【注】 在全部图像质量调整部的寄存器更新控制寄存器 (IMGCNT_UPDATE) 的 IMGCNT_VEN 位为“1”时更新此寄存器。

32.2.13 图像质量调整部的矩阵 CBB 调整寄存器 1 (IMGCNT_MTX_CBB_ADJ1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	IMGCNT_MTX_BB[10:0]										
初始值:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	IMGCNT_MTX_BR[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	IMGCNT_MTX_BB[10:0]	256	R/W	Cb/B 信号输出的 Cb/B 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	IMGCNT_MTX_BR[10:0]	0	R/W	Cb/B 信号输出的 Cr/R 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])

【注】 在全部图像质量调整部的寄存器更新控制寄存器 (IMGCNT_UPDATE) 的 IMGCNT_VEN 位为“1”时更新此寄存器。

32.2.14 图像质量调整部的矩阵 CRR 调整寄存器 0 (IMGCNT_MTX_CRR_ADJ0)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	IMGCNT_MTX_R[7:0]							
初始值:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	IMGCNT_MTX_RG[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23 ~ 16	IMGCNT_MTX_R[7:0]	128	R/W	Cr/R 信号的偏移 (DC) 调整 无符号 (0 (-128) ~ 128 (0) ~ 255 (+127) [LSB])
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	IMGCNT_MTX_RG[10:0]	0	R/W	Cr/R 信号输出的 Y/G 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])

【注】 在全部图像质量调整部的寄存器更新控制寄存器 (IMGCNT_UPDATE) 的 IMGCNT_VEN 位为“1”时更新此寄存器。

32.2.15 图像质量调整部的矩阵 CRR 调整寄存器 1 (IMGCNT_MTX_CRR_ADJ1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	IMGCNT_MTX_RB[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	IMGCNT_MTX_RR[10:0]										
初始值:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	IMGCNT_MTX_RB[10:0]	0	R/W	Cr/R 信号输出的 Cb/B 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	IMGCNT_MTX_RR[10:0]	256	R/W	Cr/R 信号输出的 Cr/R 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])

【注】 在全部图像质量调整部的寄存器更新控制寄存器 (IMGCNT_UPDATE) 的 IMGCNT_VEN 位为“1”时更新此寄存器。

32.3 使用方法

32.3.1 输入格式的调整方法

各输入格式的设定例子如下所示：

表 32.26 视频解码器 (NTSC) 输入的设定例子

寄存器名	位名	说明	设定值
INP_SEL_CNT	INP_SEL	选择输入信号。	0
INP_SEL_CNT	INP_FORMAT[2:0]	选择外部输入格式。	不需要控制
INP_SEL_CNT	INP_PXD_EDGE	选择外部输入图像信号的捕捉时钟边沿。	不需要控制
INP_SEL_CNT	INP_VS_EDGE	选择外部输入垂直同步信号的捕捉时钟边沿。	不需要控制
INP_SEL_CNT	INP_HS_EDGE	选择外部输入水平同步信号的捕捉时钟边沿。	不需要控制
INP_EXT_SYNC_CNT	INP_ENDIAN_ON	更改外部输入的位序。	不需要控制
INP_EXT_SYNC_CNT	INP_SWAP_ON	对外部输入的 B/R 信号交换进行 ON/OFF 控制。	不需要控制
INP_EXT_SYNC_CNT	INP_HS_INV	对外部输入的水平同步信号进行反相控制。	不需要控制
INP_EXT_SYNC_CNT	INP_VS_INV	对外部输入的垂直同步信号进行反相控制。	不需要控制
INP_EXT_SYNC_CNT	INP_H_EDGE_SEL	选择 BT656 输入的水平同步基准。	不需要控制
INP_EXT_SYNC_CNT	INP_F525_625	设定 BT656 输入的行数。	不需要控制
INP_EXT_SYNC_CNT	INP_H_POS[1:0]	给 BT656/601 输入的水平同步设定数据开始时序。	不需要控制
INP_VSYNC_PH_ADJ	INP_FH50[9:0]	设定 1/2fH 相位 (以时钟为单位)。	858
INP_VSYNC_PH_ADJ	INP_FH25[9:0]	设定 1/4fH 相位 (以时钟为单位)。	429
INP_DLY_ADJ	INP_VS_DLY_L[2:0]	设定垂直同步信号和场判别信号的行延迟量。	0
INP_DLY_ADJ	INP_VS_DLY[7:0]	设定垂直同步信号的延迟量 (以时钟为单位)。	0
INP_DLY_ADJ	INP_HS_DLY[7:0]	设定水平同步信号的延迟量 (以时钟为单位)。	0
INP_DLY_ADJ	INP_FLD_DLY[7:0]	设定场判别信号的延迟量 (以时钟为单位)。	0

【注】 在设定寄存器后，需要通过一部分寄存器将外部输入部的寄存器更新控制寄存器 (INP_UPDATE) 的 INP_EXT_UPDATE 位和 INP_IMG_UPDATE 位置“1”。

表 32.27 外部输入 (BT656、525i) 的设定例子

寄存器名	位名	说明	设定值
INP_SEL_CNT	INP_SEL	选择输入信号。	1
INP_SEL_CNT	INP_FORMAT[2:0]	选择外部输入格式。	3
INP_SEL_CNT	INP_PXD_EDGE	选择外部输入图像信号的捕捉时钟边沿。	0
INP_SEL_CNT	INP_VS_EDGE	选择外部输入垂直同步信号的捕捉时钟边沿。	0
INP_SEL_CNT	INP_HS_EDGE	选择外部输入水平同步信号的捕捉时钟边沿。	0
INP_EXT_SYNC_CNT	INP_ENDIAN_ON	更改外部输入的位序。	0
INP_EXT_SYNC_CNT	INP_SWAP_ON	对外部输入的 B/R 信号交换进行 ON/OFF 控制。	0
INP_EXT_SYNC_CNT	INP_HS_INV	对外部输入的水平同步信号进行反相控制。	1
INP_EXT_SYNC_CNT	INP_VS_INV	对外部输入的垂直同步信号进行反相控制。	1
INP_EXT_SYNC_CNT	INP_H_EDGE_SEL	选择 BT656 输入的水平同步基准。	0
INP_EXT_SYNC_CNT	INP_F525_625	设定 BT656 输入的行数。	0
INP_EXT_SYNC_CNT	INP_H_POS[1:0]	给 BT656/601 输入的水平同步设定数据开始时序。	0
INP_VSYNC_PH_ADJ	INP_FH50[9:0]	设定 1/2fH 相位 (以时钟为单位)。	858
INP_VSYNC_PH_ADJ	INP_FH25[9:0]	设定 1/4fH 相位 (以时钟为单位)。	429
INP_DLY_ADJ	INP_VS_DLY_L[2:0]	设定垂直同步信号和场判别信号的行延迟量。	0
INP_DLY_ADJ	INP_VS_DLY[7:0]	设定垂直同步信号的延迟量 (以时钟为单位)。	0
INP_DLY_ADJ	INP_HS_DLY[7:0]	设定水平同步信号的延迟量 (以时钟为单位)。	0
INP_DLY_ADJ	INP_FLD_DLY[7:0]	设定场判别信号的延迟量 (以时钟为单位)。	0

【注】 在设定寄存器后, 需要通过一部分寄存器将外部输入部的寄存器更新控制寄存器 (INP_UPDATE) 的 INP_EXT_UPDATE 位和 INP_IMG_UPDATE 位置 “1”。

表 32.28 外部输入 (BT601、525i) 的设定例子

寄存器名	位名	说明	设定值
INP_SEL_CNT	INP_SEL	选择输入信号。	1
INP_SEL_CNT	INP_FORMAT[2:0]	选择外部输入格式。	4
INP_SEL_CNT	INP_PXD_EDGE	选择外部输入图像信号的捕捉时钟边沿。	0
INP_SEL_CNT	INP_VS_EDGE	选择外部输入垂直同步信号的捕捉时钟边沿。	0
INP_SEL_CNT	INP_HS_EDGE	选择外部输入水平同步信号的捕捉时钟边沿。	0
INP_EXT_SYNC_CNT	INP_ENDIAN_ON	更改外部输入的位序。	0
INP_EXT_SYNC_CNT	INP_SWAP_ON	对外部输入的 B/R 信号交换进行 ON/OFF 控制。	0
INP_EXT_SYNC_CNT	INP_HS_INV	对外部输入的水平同步信号进行反相控制。	1
INP_EXT_SYNC_CNT	INP_VS_INV	对外部输入的垂直同步信号进行反相控制。	1
INP_EXT_SYNC_CNT	INP_H_EDGE_SEL	选择 BT656 输入的水平同步基准。	0
INP_EXT_SYNC_CNT	INP_F525_625	设定 BT656 输入的行数。	0
INP_EXT_SYNC_CNT	INP_H_POS[1:0]	给 BT656/601 输入的水平同步设定数据开始时序。	0
INP_VSYNC_PH_ADJ	INP_FH50[9:0]	设定 1/2fH 相位 (以时钟为单位)。	858
INP_VSYNC_PH_ADJ	INP_FH25[9:0]	设定 1/4fH 相位 (以时钟为单位)。	429
INP_DLY_ADJ	INP_VS_DLY_L[2:0]	设定垂直同步信号和场判别信号的行延迟量。	0
INP_DLY_ADJ	INP_VS_DLY[7:0]	设定垂直同步信号的延迟量 (以时钟为单位)。	0
INP_DLY_ADJ	INP_HS_DLY[7:0]	设定水平同步信号的延迟量 (以时钟为单位)。	0
INP_DLY_ADJ	INP_FLD_DLY[7:0]	设定场判别信号的延迟量 (以时钟为单位)。	0

【注】 在设定寄存器后, 需要通过一部分寄存器将外部输入部的寄存器更新控制寄存器 (INP_UPDATE) 的 INP_EXT_UPDATE 位和 INP_IMG_UPDATE 位置 “1”。

32.3.2 色彩矩阵变换的使用方法

各种标准数据变换的设定例字如下所示：

表 32.29 色彩矩阵的变换

寄存器名	位名	GBR→ GBR	GBR→ YCbCr	YCbCr→ GBR	YCbCr→ YCbCr
IMGCNT_MTX_MODE	IMGCNT_MTX_MD[1:0]	0	1	2	3
IMGCNT_MTX_YG_ADJ0	IMGCNT_MTX_YG[7:0]	128	128	128	128
IMGCNT_MTX_YG_ADJ0	IMGCNT_MTX_GG[10:0]	256	150	256	256
IMGCNT_MTX_YG_ADJ1	IMGCNT_MTX_GB[10:0]	0	29	1960	0
IMGCNT_MTX_YG_ADJ1	IMGCNT_MTX_GR[10:0]	0	77	1865	0
IMGCNT_MTX_CBB_ADJ0	IMGCNT_MTX_B[7:0]	128	128	128	128
IMGCNT_MTX_CBB_ADJ0	IMGCNT_MTX_BG[10:0]	0	1963	256	0
IMGCNT_MTX_CBB_ADJ1	IMGCNT_MTX_BB[10:0]	256	128	454	256
IMGCNT_MTX_CBB_ADJ1	IMGCNT_MTX_BR[10:0]	0	2005	0	0
IMGCNT_MTX_CRR_ADJ0	IMGCNT_MTX_R[7:0]	128	128	128	128
IMGCNT_MTX_CRR_ADJ0	IMGCNT_MTX_RG[10:0]	0	1941	256	0
IMGCNT_MTX_CRR_ADJ1	IMGCNT_MTX_RB[10:0]	0	2027	0	0
IMGCNT_MTX_CRR_ADJ1	IMGCNT_MTX_RR[10:0]	256	128	359	256

【注】 在设定寄存器后，需要将图像质量调整部的寄存器更新控制寄存器（IMGCN_UPDATE）的 IMGCNT_VEN 位置“1”。

33. 视频显示控制器 4 (3) 缩放部

33.1 缩放功能

33.1.1 功能概要

缩放部对输入控制部的 YCbCr 和 RGB 输出信号进行同步信号的生成、图像的缩小、放大和旋转控制，还对帧缓冲器进行图像的录像。

缩放部的放大处理选择图形 (1) 的处理。

缩放部的功能框图如下图所示：

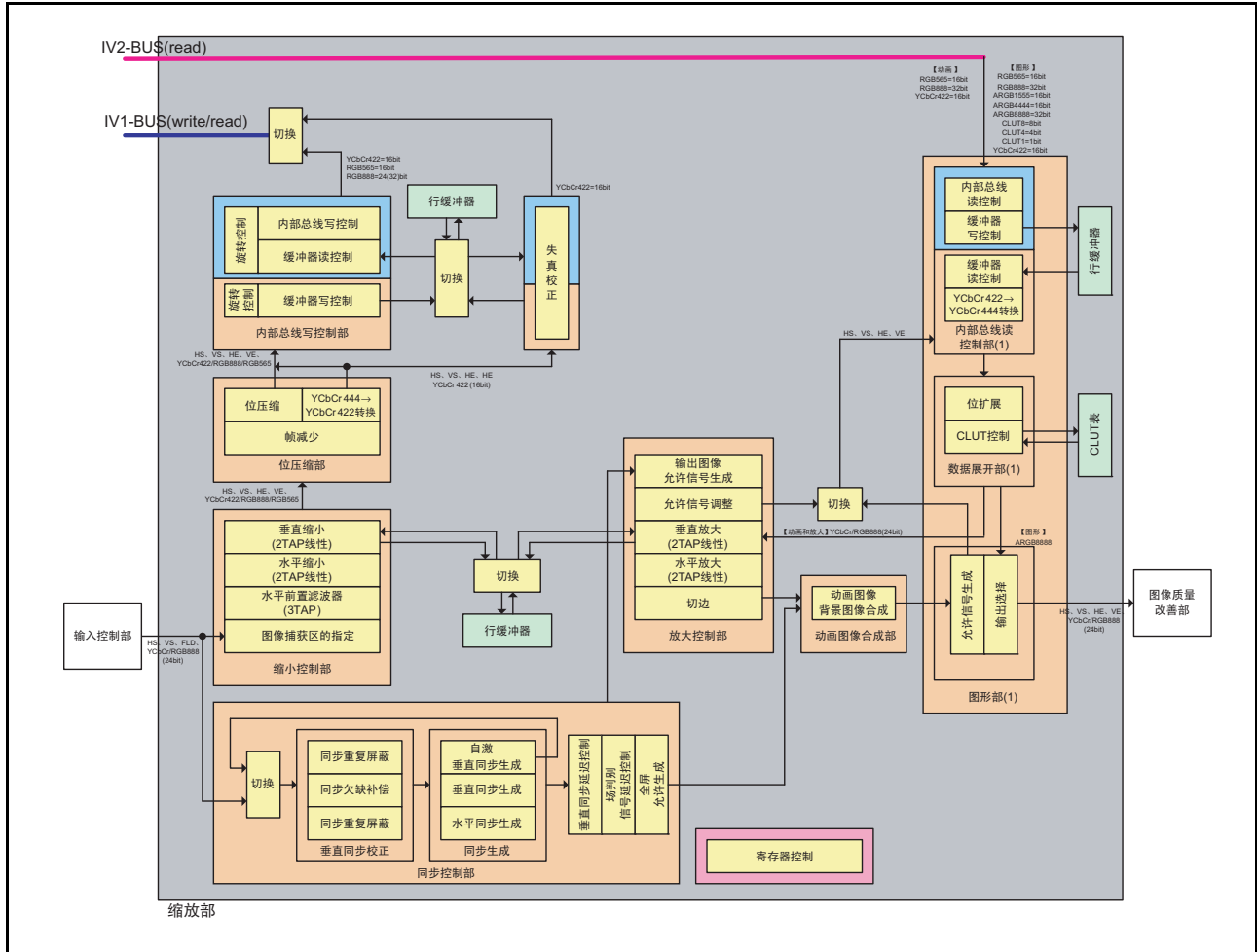


图 33.1 缩放部的功能框图

33.1.2 寄存器控制

(1) 寄存器的更新控制

通过垂直同步信号管理缩放部和图形部控制寄存器（同步控制部的部分寄存器除外）的更新时序。

在将更新控制寄存器置“1”后，在垂直同步信号的上升沿反映各种寄存器。反映后，更新控制寄存器自动清“0”。

表 33.1 寄存器的更新控制

寄存器名	位名	初始值	说明
SCL0_UPDATE	SCL0_UPDATE	0	同步控制的寄存器更新 0: 不更新寄存器 1: 更新寄存器
SCL0_UPDATE	SCL0_VEN_D	0	放大控制和帧缓冲器读控制的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器
SCL0_UPDATE	SCL0_VEN_C	0	缩小控制和帧缓冲器写控制的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器
SCL0_UPDATE	SCL0_VEN_B	0	同步控制和放大控制的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器
SCL0_UPDATE	SCL0_VEN_A	0	缩小控制的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器
SCL1_UPDATE	SCL1_VEN_B	0	帧缓冲器写控制的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器
SCL1_UPDATE	SCL1_VEN_A	0	帧缓冲器写控制的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器
GR1_VEN	GR1_P_VEN	0	图形显示的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器
GR1_VEN	GR1_IBUS_VEN	0	帧缓冲器读控制的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器

在输入垂直同步信号的上升沿更新由 SCL0_VEN_A 位、SCL0_VEN_C 位、SCL1_VEN_A 位和 SCL1_VEN_B 位控制的寄存器。

在输出垂直同步信号的上升沿更新由 SCL0_VEN_B 位、SCL0_VEN_D 位、GR1_P_VEN 位和 GR1_IBUS_VEN 位控制的寄存器。

33.1.3 同步控制

(1) 垂直同步信号的选择

选择缩放输出的垂直同步信号。

在显示外部输入信号时，输出外部输入的垂直同步信号。

如果没有外部输入信号，就输出自激垂直同步信号。

表 33.2 垂直同步信号的选择控制

寄存器名	位名	初始值	说明
SCL0_FRC3	RES_VS_SEL	1	输出的垂直同步信号的选择 0: 外部输入垂直同步信号 1: 内部生成的自激垂直同步信号

(2) 垂直同步信号的重复屏蔽

通过设定从接受输入垂直同步信号到开始接受下一个垂直同步信号的时序，禁止接受短于标准周期的垂直同步信号。

通过 RES_VMASK[15:0] 位设定垂直同步信号的接受屏蔽期间。

屏蔽期间 [usec]=RES_VMASK×128÷ 像素时钟 [MHz]

通过 RES_VMASK_ON 位控制此功能的有效或者无效。

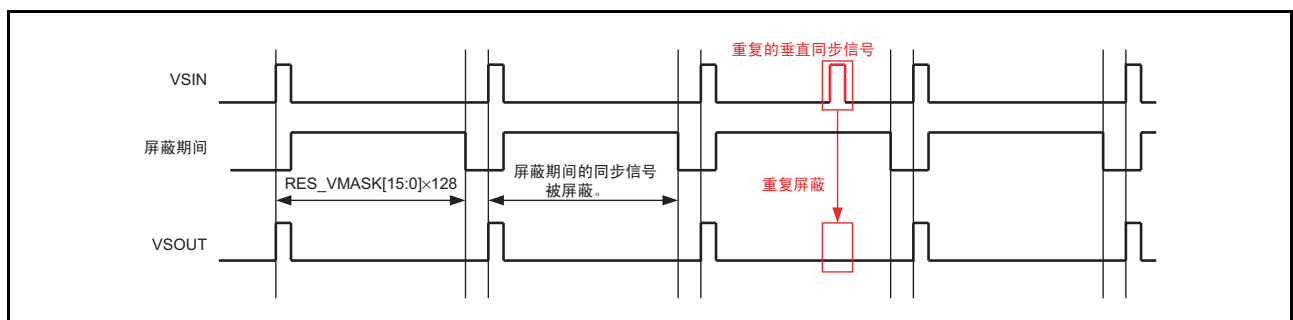


图 33.2 垂直同步信号的重复屏蔽时序图

表 33.3 垂直同步信号的重复屏蔽控制

寄存器名	位名	初始值	说明
SCL0_FRC1	RES_VMASK_ON	1	垂直同步信号的重复屏蔽控制 0: 重复屏蔽控制 OFF 1: 重复屏蔽控制 ON
SCL0_FRC1	RES_VMASK[15:0]	2800	垂直同步信号的重复屏蔽期间设定 以 128 倍像素时钟周期，设定垂直同步信号的同步重复屏蔽期间。 屏蔽期间 [usec]=RES_VMASK×128÷ 像素时钟 [MHz]

(3) 垂直同步信号的欠缺补偿

通过设定从接受输入垂直同步信号到输入下一个垂直同步信号为止的等待期间，禁止输入长于标准周期的垂直同步信号。

如果在等待期间内没有接受垂直同步信号，就插入内部生成的同步信号。

通过 RES_VLACK[15:0] 位设定接受垂直同步信号的等待期间。

等待期间 [usec]=RES_VLACK×128÷ 像素时钟 [MHz]

通过 RES_VLACK_ON 位控制此功能的有效或者无效。

如果在垂直同步信号的接受期间没有输入垂直同步信号，就将 RES_QVLACK 位置为“H”电平。

如果在垂直同步信号的接受期间至少连续4次检测到垂直同步信号，就将 RES_QVLOCK 位置为“H”电平。

即使将 RES_VMASK_ON 位和 RES_VLACK_ON 位都设定为 OFF，也能检测到 RES_QVLOCK 位。

但是，需要正确地设定 RES_VMASK 位和 RES_VLACK 位。

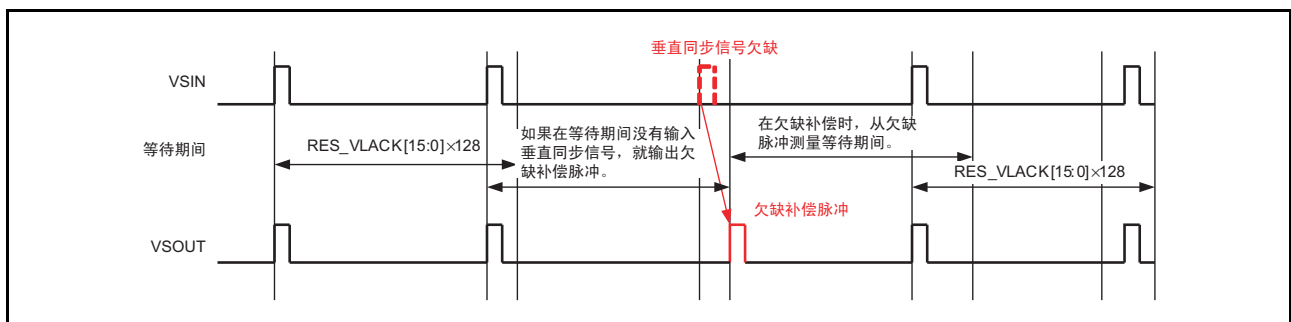


图 33.3 垂直同步信号的欠缺补偿

表 33.4 垂直同步信号的欠缺补偿控制

寄存器名	位名	初始值	说明
SCL0_FRC2	RES_VLACK_ON	1	垂直同步信号的欠缺补偿控制 0: 欠缺补偿控制 OFF 1: 欠缺补偿控制 ON
SCL0_FRC2	RES_VLACK[15:0]	3600	垂直同步信号的欠缺补偿期间设定 以 128 倍像素时钟周期，设定从垂直同步信号到输出同步欠缺补偿脉冲为止的等待期间。 等待期间 [usec]=RES_VLACK×128÷ 像素时钟 [MHz]
SCL0_FRC9	RES_QVLACK	—	垂直同步信号的欠缺检测标志 1: 输入垂直同步信号有欠缺 0: 输入垂直同步信号无欠缺
SCL0_FRC9	RES_QVLOCK	—	垂直同步信号的锁定检测标志 1: 输入垂直同步信号至少连续 4 个垂直期间没有重复或者欠缺 0: 输入垂直同步信号有重复或者欠缺

垂直同步信号的重复屏蔽和垂直同步信号的欠缺补偿为重复屏蔽 → 欠缺补偿 → 重复屏蔽的结构。

在垂直同步信号有欠缺并且在欠缺补偿脉冲后立即输入垂直同步信号等情况下，为了不重复输出垂直同步信号，在欠缺补偿后插入重复屏蔽处理。

后段垂直同步重复屏蔽处理的 ON/OFF 控制和垂直同步欠缺补偿的设定是共用的，屏蔽周期和前段垂直同步重复屏蔽处理的屏蔽期间的设定是共用的。

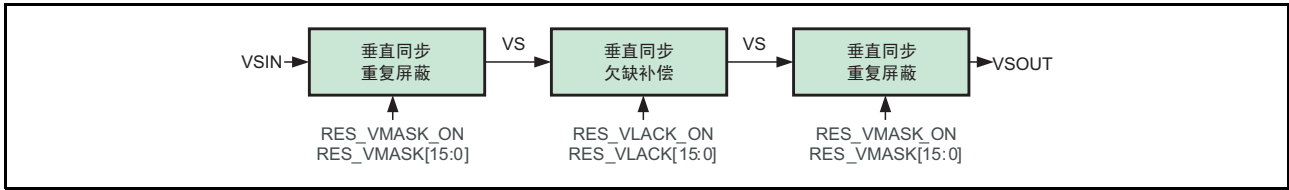


图 33.4 重复屏蔽和欠缺补偿结构

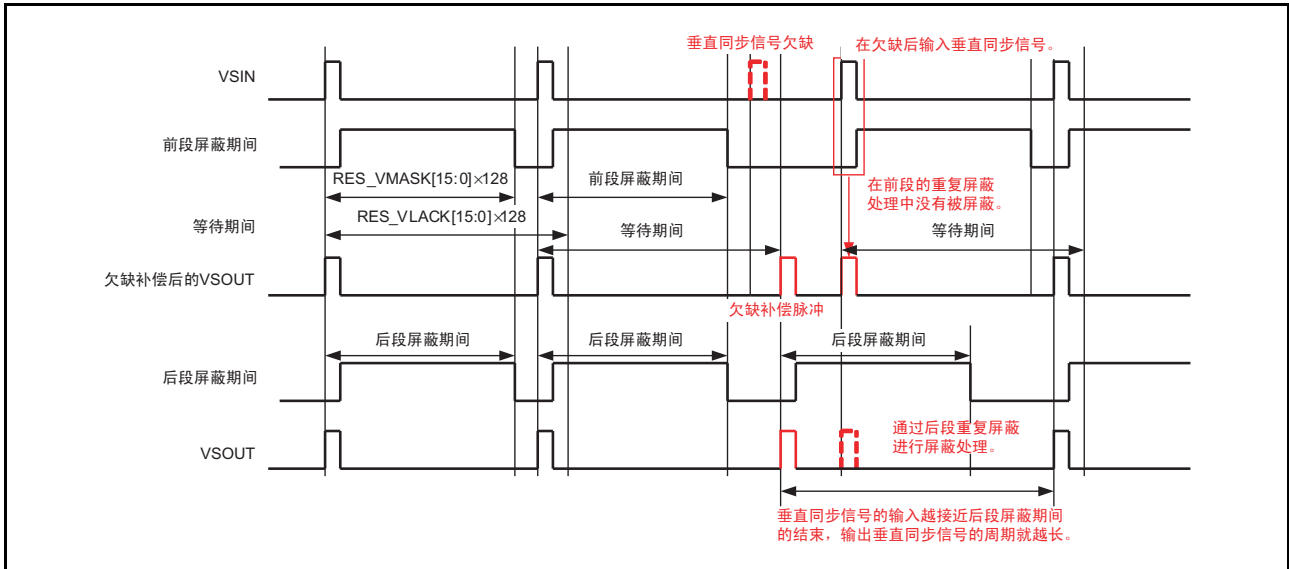


图 33.5 重复屏蔽和欠缺补偿时序图

(4) 自激周期

设定自激垂直同步信号和水平同步信号的周期。

水平同步信号周期 [usec] = (RES_FH + 1) ÷ 像素时钟 [MHz]

垂直同步信号周期 [usec] = 水平周期 [usec] × (RES_FV + 1)

表 33.5 自激周期控制

寄存器名	位名	初始值	说明
SCL0_FRC4	RES_FV[10:0]	524	自激垂直同步信号的周期设定 自激垂直同步信号周期 = (RES_FV + 1) × 水平周期 [usec]
SCL0_FRC4	RES_FH[10:0]	799	水平同步信号的周期设定 水平同步信号周期 [usec] = (RES_FH + 1) ÷ 像素时钟 [MHz]

要选择外部输入的垂直同步信号时，将 RES_VS_SEL 位置 “0”。此时，不输出内部生成的自激垂直同步信号。

对于水平同步信号，随时从缩放部输出由自激设定生成的水平同步信号。

(5) 垂直同步信号的延迟控制

控制缩放输出的垂直同步信号的延迟。
此延迟用于调整帧缓冲器读时序。

表 33.6 垂直同步信号的延迟控制

寄存器名	位名	初始值	说明
SCL0_FRC5	RES_VSDLY[7:0]	1	垂直同步信号的延迟控制 以输出水平周期为单位，调整垂直同步信号的延迟。 垂直同步信号延迟量 [usec]: $RES_VSDLY \times \text{输出水平周期 [usec]}$

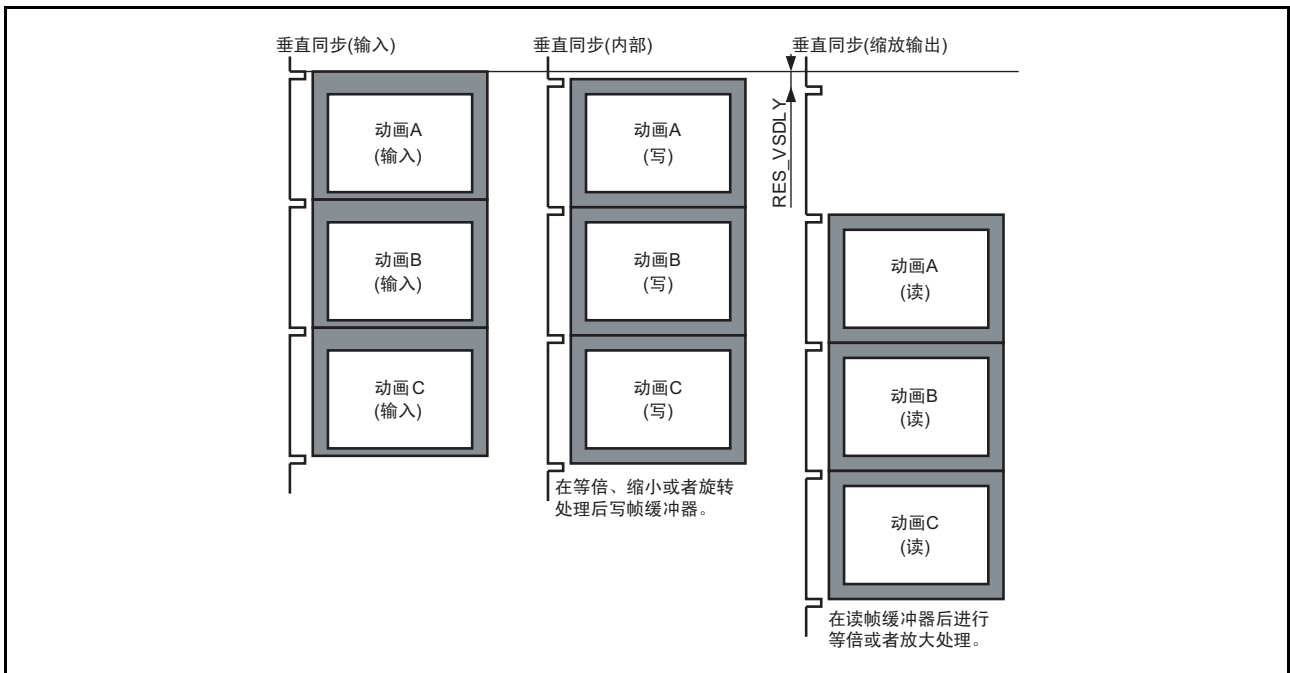


图 33.6 垂直同步信号的相位关系图 (使用 2 面帧缓冲器的情况)

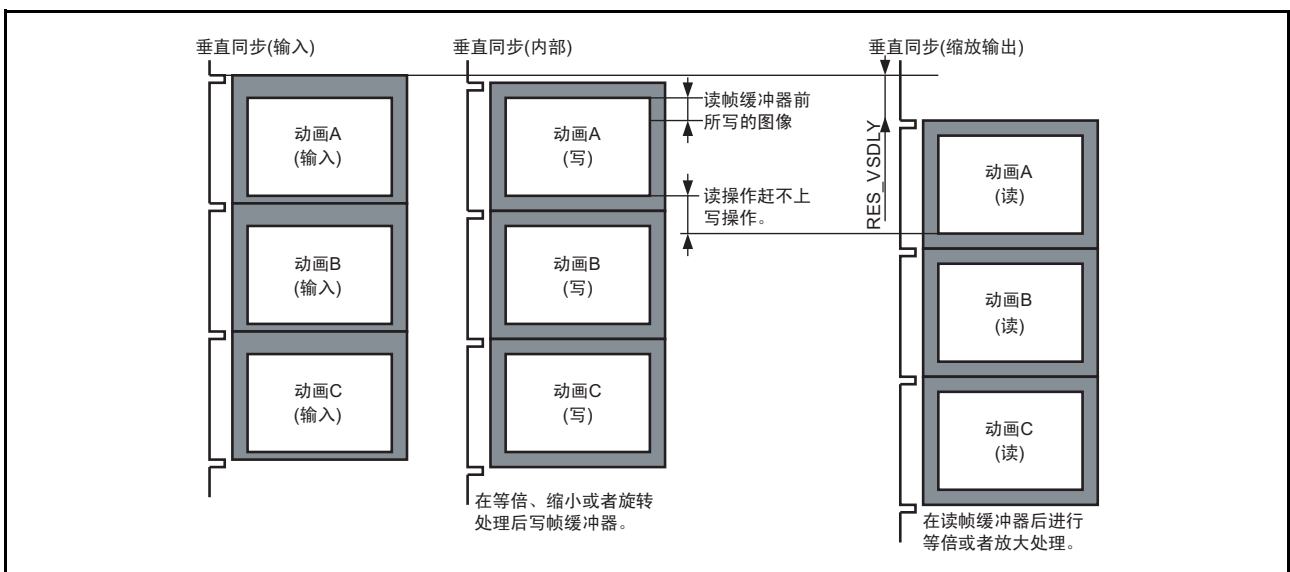


图 33.7 垂直同步信号的相位关系图 (使用 1 面帧缓冲器的情况)

33.1.4 视角大小的设定

(1) 图像捕捉区的设定

对进行缩小和放大处理的图像捕捉区进行设定。

以输入的水平同步信号和垂直同步信号为基准，用起始位置和宽度设定图像捕捉区。

表 33.7 图像捕捉区的控制

寄存器名	位名	初始值	说明
SCL0_DS2	RES_VS[10:0]	18	捕捉图像信号的垂直位置设定 (VSYNC+V 后沿 行数) 【注】 必须至少设定 4 行，使 RES_VS+RES_VW 不超过 2039 行。
SCL0_DS2	RES_VW[10:0]	240	捕捉图像信号的垂直宽度 (行数) 【注】 RES_VS+RES_VW 不能超过 2039 行。
SCL0_DS3	RES_HS[10:0]	244	捕捉图像信号的水平位置设定 (HSYNC+H 后沿 图像时钟个数) 【注】 必须至少设定 16 个时钟，使 RES_HS+RES_HW 不超过 2015 个时钟。
SCL0_DS3	RES_HW[10:0]	1440	捕捉图像信号的水平宽度 (图像时钟个数) 【注】 RES_HS+RES_HW 不能超过 2015 个时钟。

(2) 全屏幕允许信号的生成

设定缩放输出的全屏幕有效期间。

以缩放输出的水平同步信号和垂直同步信号为基准，用起始位置和宽度设定有效期间。

表 33.8 全屏幕的允许控制

寄存器名	位名	初始值	说明
SCL0_FRC6	RES_F_VS[10:0]	35	全屏幕垂直允许信号的起始位置设定 (VSYNC+V 后沿 行数) 【注】 必须至少设定 4 行，使 RES_F_VS+RES_F_VW 不超过 2039 行。
SCL0_FRC6	RES_F_VW[10:0]	480	全屏幕垂直允许信号的宽度设定 (行数) 【注】 RES_F_VS+RES_F_VW 不能超过 2039 行。
SCL0_FRC7	RES_F_HS[10:0]	144	全屏幕水平允许信号的起始位置设定 (HSYNC+H 后沿 像素时钟个数) 【注】 必须至少设定 16 个时钟，使 RES_F_HS+RES_F_HW 不超过 2015 个时钟。
SCL0_FRC7	RES_F_HW[10:0]	640	全屏幕水平允许信号的宽度设定 (像素时钟个数) 【注】 RES_F_HS+RES_F_HW 不能超过 2015 个时钟。

(3) 图像输出允许信号的生成

设定输出图像的有效期间。

以缩放输出的水平同步信号和垂直同步信号为基准，用起始位置和宽度设定有效期间。

表 33.9 图像输出的允许控制

寄存器名	位名	初始值	说明
SCL0_US2	RES_P_VS[10:0]	35	图像输出的垂直允许信号的起始位置设定 (VSYNC+V 后沿 行数) 【注】 设定至少设定 4 行，使 RES_P_VS+RES_P_VW 不超过 2039 行。
SCL0_US2	RES_P_VW[10:0]	480	图像输出的垂直允许信号宽度设定 (行数) 【注】 RES_P_VS+RES_P_VW 不能超过 2039 行。
SCL0_US3	RES_P_HS[10:0]	144	图像输出的水平允许信号起始位置设定 (HSYNC+H 后沿 像素时钟个数) 【注】 必须至少设定 16 个时钟，使 RES_P_HS+RES_P_HW 不超过 2015 个时钟。
SLC0_US3	RES_P_HW[10:0]	640	图像输出的水平允许信号宽度设定 (像素时钟个数) 【注】 RES_P_HS+RES_P_HW 不能超过 2015 个时钟。

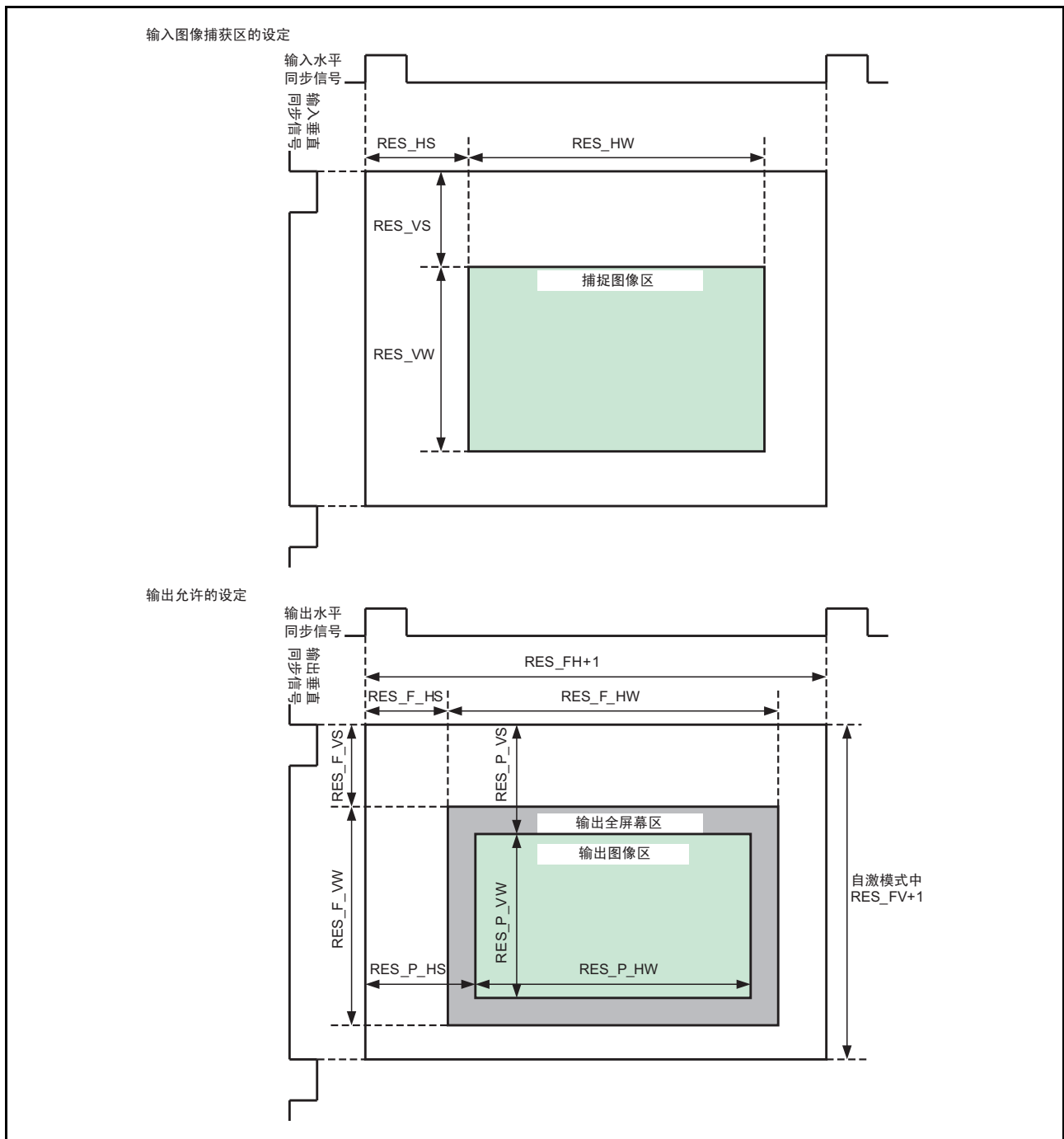


图 33.8 允许设定图

33.1.5 缩放设定

(1) 缩放处理块

缩小控制部对输入控制部的输入图像进行缩放处理。
 如果有旋转控制，就在按比例缩小后旋转，然后写到帧缓冲器。
 放大控制部在从帧缓冲器读到旋转后的图像后进行缩放处理。

表 33.10 旋转控制和缩放处理

旋转控制	水平缩放	垂直缩放	缩小控制部	放大控制部
正常	水平缩小	垂直缩小	水平缩小 / 垂直缩小	水平等倍 / 垂直等倍
	水平缩小	垂直放大	水平缩小 / 垂直等倍	水平等倍 / 垂直放大
	水平放大	垂直缩小	水平等倍 / 垂直缩小	水平放大 / 垂直等倍
	水平放大	垂直放大	水平等倍 / 垂直等倍	水平放大 / 垂直放大
水平镜像	水平缩小	垂直缩小	水平缩小 / 垂直缩小	水平等倍 / 垂直等倍
	水平缩小	垂直放大	水平缩小 / 垂直等倍	水平等倍 / 垂直放大
	水平放大	垂直缩小	水平等倍 / 垂直缩小	水平放大 / 垂直等倍
	水平放大	垂直放大	水平等倍 / 垂直等倍	水平放大 / 垂直放大
90 度旋转	(水平输入 → 垂直输出) 缩小	(垂直输入 → 水平输出) 缩小	水平缩小 / 垂直缩小	水平等倍 / 垂直等倍
	(水平输入 → 垂直输出) 缩小	(垂直输入 → 水平输出) 放大	水平缩小 / 垂直等倍	水平放大 / 垂直等倍
	(水平输入 → 垂直输出) 放大	(垂直输入 → 水平输出) 放大	水平等倍 / 垂直等倍	水平放大 / 垂直放大
180 度旋转	水平缩小	垂直缩小	水平缩小 / 垂直缩小	水平等倍 / 垂直等倍
	水平缩小	垂直放大	水平缩小 / 垂直等倍	水平等倍 / 垂直放大
	水平放大	垂直缩小	水平等倍 / 垂直缩小	水平放大 / 垂直等倍
	水平放大	垂直放大	水平等倍 / 垂直等倍	水平放大 / 垂直放大
270 度旋转	(水平输入 → 垂直输出) 缩小	(垂直输入 → 水平输出) 缩小	水平缩小 / 垂直缩小	水平等倍 / 垂直等倍
	(水平输入 → 垂直输出) 缩小	(垂直输入 → 水平输出) 放大	水平缩小 / 垂直等倍	水平放大 / 垂直等倍
	(水平输入 → 垂直输出) 放大	(垂直输入 → 水平输出) 放大	水平等倍 / 垂直等倍	水平放大 / 垂直放大

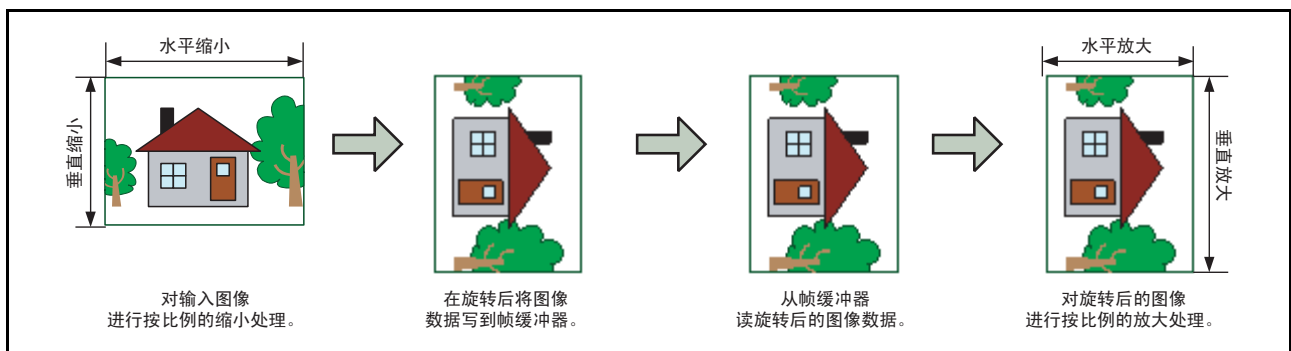


图 33.9 旋转控制和缩放处理

因为缩小控制部的垂直缩小和放大控制部的垂直放大为排他处理，所以不能同时使用。
 因此，不能在进行 90 度旋转或者 270 度旋转时进行以下的缩放处理。

表 33.11 不支持的缩放处理

旋转控制	水平缩放	垂直缩放	缩小控制部	放大控制部
90 度旋转 270 度旋转	(水平输入 → 垂直输出) 放大	(垂直输入 → 水平输出) 缩小	水平等倍 / 垂直缩小	水平等倍 / 垂直放大

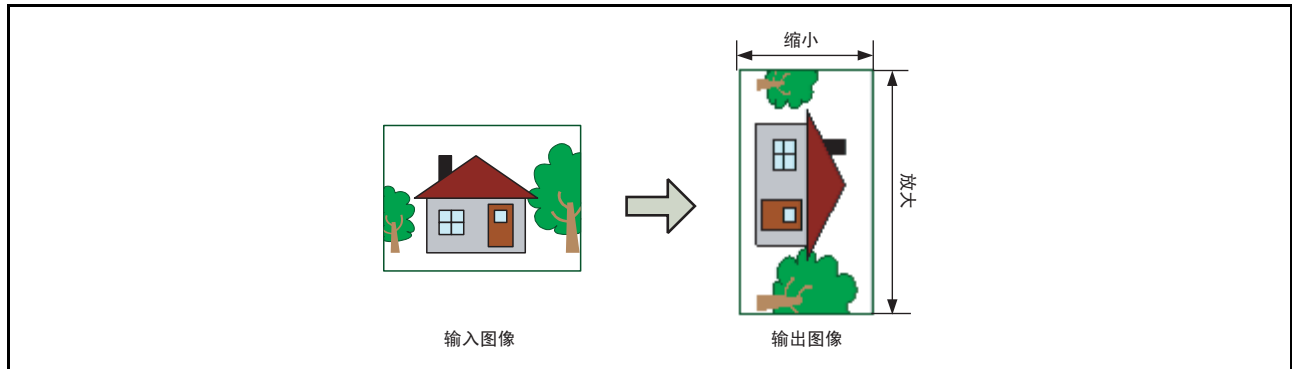


图 33.10 不支持的缩放处理

33.1.6 水平前置滤波器

对在水平缩小处理时用于抑制信号频带宽度的亮度 (Y) 信号和 RGB 信号进行水平前置滤波器的 ON/OFF 控制。输入格式遵循写运行模式寄存器 (SCL1_WR1) 的 RES_MD[1:0] 位的设定。

如果水平缩小率高并且折叠频率成分明显时, 就必须将水平前置滤波器置为 ON。

表 33.12 水平前置滤波器控制的设定值

输入格式	RES_PFIL_SEL	运行
YCbCr 输入	1	Y 信号滤波器 ON、Cb/Cr 信号滤波器 OFF
	0	滤波器 OFF
RGB 输入	1	RGB 信号滤波器 ON
	0	滤波器 OFF

表 33.13 水平前置滤波器控制

寄存器名	位名	初始值	说明
SCL0_DS4	RES_PFIL_SEL	0	亮度信号前置滤波器的模式选择 0: 前置滤波器 OFF 1: 前置滤波器 ON (1/4+1/2+1/4)

33.1.7 水平缩小处理

对水平像素进行 1/1 ~ 1/8[倍] 任意倍率的像素转换。

缩放滤波器能选择保持内插和线性内插。

(1) 1TAP 保持内插

当内插位置在输入像素 X_n 和 X_{n+1} 之间时, $X_{interpo}$ 内插值为

$$X_{interpo} = X_n$$

(2) 2TAP 线性内插

当内插位置在输入像素 X_n 和 X_{n+1} 之间时，通过内插位置 $phase$ 求 $X_{interpo}$ 内插值，计算式如下：

$$X_{interpo}=(X_n \times (4096 - phase) + X_{n+1} \times phase) / 4096$$

(3) 水平缩小率的计算

用以下计算式，通过缩小控制部的输入像素个数 RES_HW 和输出像素个数 RES_OUT_HW 求水平缩小率 RES_DS_H_RATIO（小数四舍五入）。

$$RES_DS_H_RATIO = \text{round}(RES_HW \div RES_OUT_HW \times 4096)$$

在水平等倍的情况下，必须给 RES_HW 位和 RES_OUT_HW 位设定相同的值并且将 RES_DS_H_RATIO 位置“4096”。

(4) 最后输入像素的欠缺对策

通过（最后输入 -1）像素和最后输入像素间的内插，生成画面右端的最后输出像素。根据水平缩小率，当最后输出像素的内插位置靠近（最后输入 -1）像素时，最后输入像素看上去可能有欠缺。

能用以下计算式，通过调整水平缩小率来减少最后输入像素欠缺的影响。

求调整前的水平缩小率 $RATIO_org$ ，算出调整值 σ 和缩小率 RES_DS_H_RATIO。

$$RATIO_org = \text{round}(RES_HW \div RES_OUT_HW \times 4096)$$

$$\sigma = (RATIO_org \times (RES_OUT_HW - 1) - (RES_HW - 1) \times 4096) \div (RES_OUT_HW - 1)$$

$$RES_DS_H_RATIO = \text{round}(RATIO_org - \sigma)$$

表 33.14 水平缩小控制

寄存器名	位名	初始值	说明
SCL0_DS1	RES_DS_H_ON	1	水平缩小 ON/OFF 设定 0: OFF 1: ON
SCL0_DS7	RES_OUT_HW[10:0]	640	缩小控制部输出的水平有效像素个数 (图像时钟个数)
SCL0_DS4	RES_DS_H_INTERPOTYP	1	水平内插方法选择 0: 保持内插 1: 线性内插
SCL0_DS4	RES_DS_H_RATIO[15:0]	9224	水平缩小率 [15:12]: 整数部 [11:0]: 小数部 $\text{round}(RES_HW \div RES_OUT_HW \times 4096)$ RES_DS_H_RATIO < 4096: 禁止设定 RES_DS_H_RATIO = 4096: 等倍 RES_DS_H_RATIO > 4096: 缩小

【注】 在设定时，必须以 4 个像素为单位调整 RES_OUT_HW 位并且满足 $RES_OUT_HW \leq RES_HW$ 的条件。

33.1.8 垂直缩小处理

对垂直行进行 1/1 ~ 1/8[倍] 任意倍率的像素转换。

缩放滤波器能选择保持内插和线性内插。

(1) 1TAP 保持内插

当内插位置在输入行 X_n 和 X_{n+1} 之间时， $X_{interpo}$ 内插值为

$$X_{interpo} = X_n$$

(2) 2TAP 线性内插

当内插位置在输入行 X_n 和 X_{n+1} 之间时，通过内插位置 $phase$ 求 $X_{interpo}$ 内插值，计算式如下：

$$X_{interpo}=(X_n \times (4096 - phase) + X_{n+1} \times phase) / 4096$$

(3) 垂直缩小率的计算

用以下计算式，通过缩小控制部的输入行数 RES_VW 和输出行数 RES_OUT_VW 求垂直缩小率 RES_V_RATIO (小数四舍五入)。

$$RES_V_RATIO = \text{round}(RES_VW \div RES_OUT_VW \times 4096)$$

在垂直等倍或者垂直放大的情况下，必须给 RES_VW 位和 RES_OUT_VW 位设定相同的值。

在垂直等倍的情况下，必须将 RES_V_RATIO 位置“4096”进行缩小处理。

(4) 最后输入行欠缺对策

通过 (最后输入 -1) 行和最后输入行间的内插，生成画面下端的最后输出行。根据垂直缩小率，当最后输出行的内插位置接近 (最后输入 -1) 行，最后输入行看上去可能有欠缺。

能用以下计算式，通过调整垂直缩小率来减少最后输入行欠缺的影响。

求调整前的垂直缩小率 $RATIO_org$ ，算出调整值 σ 和缩小率 RES_V_RATIO 。

$$RATIO_org = \text{round}(RES_VW \div RES_OUT_VW \times 4096)$$

$$\sigma = (RATIO_org \times (RES_OUT_VW - 1) - (RES_VW - 1) \times 4096) \div (RES_OUT_VW - 1)$$

$$RES_V_RATIO = \text{round}(RATIO_org - \sigma)$$

表 33.15 垂直缩小控制

寄存器名	位名	初始值	说明
SCL0_DS1	RES_DS_V_ON	1	垂直缩小 ON/OFF 设定 0: OFF 1: ON
SCL0_DS7	RES_OUT_VW[10:0]	240	缩小控制部输出的垂直有效行数 (行数) 将这些位的设定反映到写到帧缓冲器的行数。 当 SCL1_WR1.GR1_FLM_LOOP 位为“0” (帧写模式) 时，必须指定 1 帧行数。 当 GR1_FLM5.GR1_FLM_LOOP 位为“1” (行写模式) 时，必须指定要链状写的行数。
SCL0_DS5	RES_V_INTERPOTYP	1	垂直内插方法选择 0: 保持内插 1: 线性内插
SCL0_DS6	RES_V_RATIO[15:0]	2044	垂直放大率 / 缩小率 [15:12]: 整数部 [11:0]: 小数部 round($RES_VW \div RES_OUT_VW \times 4096$): 缩小时 round($RES_IN_VW \div RES_P_VW \times 4096$): 放大时 RES_V_RATIO < 4096: 放大 RES_V_RATIO = 4096: 等倍 RES_V_RATIO > 4096: 缩小

【注】 RES_V_RATIO 位和 $RES_V_INTERPOTYP$ 位在垂直缩小和垂直放大中为公共寄存器。

因为垂直缩小和垂直放大为排他处理，所以不能同时使用。

在设定时，必须以 4 行为单位调整 RES_OUT_VW 位并且满足 $RES_OUT_VW \leq RES_VW$ 的条件。

33.1.9 水平放大处理

对水平像素进行 1 ~ 8[倍] 任意倍率的像素转换。
缩放滤波器能选择保持内插和线性内插。

(1) 1TAP 保持内插

当内插位置在输入像素 X_n 和 X_{n+1} 之间时, $X_{interpo}$ 内插值为
 $X_{interpo}=X_n$

(2) 2TAP 线性内插

当内插位置在输入像素 X_n 和 X_{n+1} 之间时, 通过内插位置 $phase$ 求 $X_{interpo}$ 内插值, 计算式如下:
 $X_{interpo}=(X_n \times (4096 - phase) + X_{n+1} \times phase) / 4096$

(3) 水平放大率的计算

用以下计算式, 通过放大控制部的输入像素个数 RES_IN_HW 和输出像素个数 RES_P_HW 求水平放大率 $RES_US_H_RATIO$ (小数四舍五入)。

$$RES_US_H_RATIO = \text{round}(RES_IN_HW \div RES_P_HW \times 4096)$$

在水平等倍的情况下, 必须给 RES_IN_HW 位和 RES_P_HW 位设定相同的值并且将 $RES_US_H_RATIO$ 位置 “4096”。

(4) 折叠对策

通过最后输入像素和折叠 (最后输入 -1) 像素间的内插, 生成画面右端的最后输出像素。因此, 根据水平放大率, 折叠可能比较明显。

能用以下计算式, 通过调整水平放大率来减小折叠像素的影响。

求调整前的水平放大率 $RATIO_org$, 算出调整值 σ 和放大率 $RES_US_H_RATIO$ 。

$$RATIO_org = \text{round}(RES_IN_HW \div RES_P_HW \times 4096)$$

$$\sigma = (RATIO_org \times (RES_P_HW - 1) - (RES_IN_HW - 1) \times 4096) \div (RES_P_HW - 1)$$

$$RES_US_H_RATIO = \text{round}(RATIO_org - \sigma)$$

表 33.16 水平放大处理控制

寄存器名	位名	初始值	说明
SCL0_US1	RES_US_H_ON	1	水平放大 ON/OFF 设定 0: OFF 1: ON
SCL0_US4	RES_IN_HW[10:0]	640	放大控制部输入的水平有效像素个数 (像素时钟个数)
SCL0_US6	RES_US_H_INTERPOTYP	1	水平内插方法指示 0: 保持内插 1: 线性内插
SCL0_US5	RES_US_H_RATIO[15:0]	9224	水平放大率 [15:12]: 整数部 [11:0]: 小数部 $\text{round}(RES_IN_HW \div RES_P_HW \times 4096)$ RES_US_H_RATIO < 4096: 放大 RES_US_H_RATIO = 4096: 等倍 RES_US_H_RATIO > 4096: 禁止设定

33.1.10 垂直放大处理

对垂直行进行 1 ~ 8[倍] 任意倍率的像素转换。
缩放滤波器能选择保持内插和线性内插。

(1) 1TAP 保持内插

当内插位置在输入行 X_n 和 X_{n+1} 之间时, $X_{interpo}$ 内插值为
 $X_{interpo}=X_n$

(2) 2TAP 线性内插

当内插位置在输入行 X_n 和 X_{n+1} 之间时, 通过内插位置 phase 求 $X_{interpo}$ 内插值, 计算式如下:
 $X_{interpo}=(X_n \times (4096 - \text{phase}) + X_{n+1} \times \text{phase}) / 4096$

(3) 垂直放大率的计算

用以下计算式, 通过放大控制部的输入行数 RES_IN_VW 和输出行数 RES_P_VW 求垂直放大率 RES_V_RATIO (小数四舍五入)。

$$\text{RES_V_RATIO} = \text{round}(\text{RES_IN_VW} \div \text{RES_P_VW} \times 4096)$$

在垂直等倍或者垂直缩小的情况下, 必须给 RES_IN_VW 位和 RES_P_VW 位设定相同的值。

(4) 折叠对策

通过最后一行和折叠 (最后输入 -1) 行的内插, 生成画面下端的最后输出行波, 因此, 根据垂直放大率, 折叠可能比较明显。

能以下计算式, 通过调整垂直放大率来减小折叠行的影响。

求调整前的垂直放大率 RATIO_org , 算出调整值 σ 和放大率 RES_V_RATIO。

$$\text{RATIO_org} = \text{round}(\text{RES_IN_VW} \div \text{RES_P_VW} \times 4096)$$

$$\sigma = (\text{RATIO_org} \times (\text{RES_P_VW} - 1) - (\text{RES_IN_VW} - 1) \times 4096) \div (\text{RES_P_VW} - 1)$$

$$\text{RES_V_RATIO} = \text{round}(\text{RATIO_org} - \sigma)$$

表 33.17 垂直放大处理控制

寄存器名	位名	初始值	说明
SCL0_US1	RES_US_V_ON	1	垂直放大 ON/OFF 设定 0: OFF 1: ON
SCL0_US4	RES_IN_VW[10:0]	240	放大控制部输入的垂直有效行数 (行数)
SCL0_DS5	RES_V_INTERPOTYP	1	垂直内插方法选择 0: 保持内插 1: 线性内插
SCL0_DS6	RES_V_RATIO[15:0]	2044	垂直放大率 [15:12]: 整数部 [11:0]: 小数部 $\text{round}(\text{RES_VW} \div \text{RES_OUT_VW} \times 4096)$: 缩小时 $\text{round}(\text{RES_IN_VW} \div \text{RES_P_VW} \times 4096)$: 放大时 RES_V_RATIO < 4096: 放大 RES_V_RATIO = 4096: 等倍 RES_V_RATIO > 4096: 缩小

【注】 RES_V_RATIO 位和 RES_V_INTERPOTYP 位在垂直缩小和垂直放大中为公共寄存器。
因为垂直缩小和垂直放大为排他处理, 所以不能同时使用。

33.1.11 IP 转换

(1) 初始相位的控制

在输入隔行扫描信号时，通过个别调整 TOP 场和 BOTTOM 场的缩放初始相位，减轻场间的行偏移引起的行间闪烁。

各种运行时的设定请参照下表。

表 33.18 IP 转换时的缩放初始相位设定表（标准值）

旋转控制	水平缩放	垂直缩放	参照位（设定值）
正常	水平缩小	垂直缩小	RES_TOP_INIPHASE=2048
	水平缩小	垂直放大	RES_TOP_INIPHASE=2048
	水平放大	垂直缩小	RES_TOP_INIPHASE=2048
	水平放大	垂直放大	RES_TOP_INIPHASE=2048
水平镜像	水平缩小	垂直缩小	RES_TOP_INIPHASE=2048
	水平缩小	垂直放大	RES_TOP_INIPHASE=2048
	水平放大	垂直缩小	RES_TOP_INIPHASE=2048
	水平放大	垂直放大	RES_TOP_INIPHASE=2048
90 度旋转	（水平输入 → 垂直输出）缩小	（垂直输入 → 水平输出）缩小	RES_TOP_INIPHASE=2048
	（水平输入 → 垂直输出）缩小	（垂直输入 → 水平输出）放大	RES_TOP_INIPHASE=2048
	（水平输入 → 垂直输出）放大	（垂直输入 → 水平输出）放大	RES_US_HB_INIPHASE=2048
180 度旋转	水平缩小	垂直缩小	RES_TOP_INIPHASE=2048
	水平缩小	垂直放大	RES_BTM_INIPHASE=2048
	水平放大	垂直缩小	RES_TOP_INIPHASE=2048
	水平放大	垂直放大	RES_BTM_INIPHASE=2048
270 度旋转	（水平输入 → 垂直输出）缩小	（垂直输入 → 水平输出）缩小	RES_TOP_INIPHASE=2048
	（水平输入 → 垂直输出）缩小	（垂直输入 → 水平输出）放大	RES_TOP_INIPHASE=2048
	（水平输入 → 垂直输出）放大	（垂直输入 → 水平输出）放大	RES_US_HT_INIPHASE=2048

【注】 必须将表中没有设定值的初始相位控制寄存器置“0”。

在输入逐行扫描信号时，必须将初始相位控制寄存器置“0”。

表 33.19 缩放初始相位的控制

寄存器名	位名	初始值	说明
SCL0_DS5	RES_BTM_INIPHASE [11:0]	0	BOTTOM 场垂直内插开始相位值 0 ~ 4095 (0 ~ 约 1.0)
SCL0_DS5	RES_TOP_INIPHASE [11:0]	2048	TOP 场垂直内插开始相位值 0 ~ 4095 (0 ~ 约 1.0)
SCL0_US6	RES_US_HB_INIPHASE [11:0]	0	BOTTOM 场水平内插开始相位值 0 ~ 4095 (0 ~ 约 1.0)
SCL0_US6	RES_US_HT_INIPHASE [11:0]	0	TOP 场水平内插开始相位值 0 ~ 4095 (0 ~ 约 1.0)

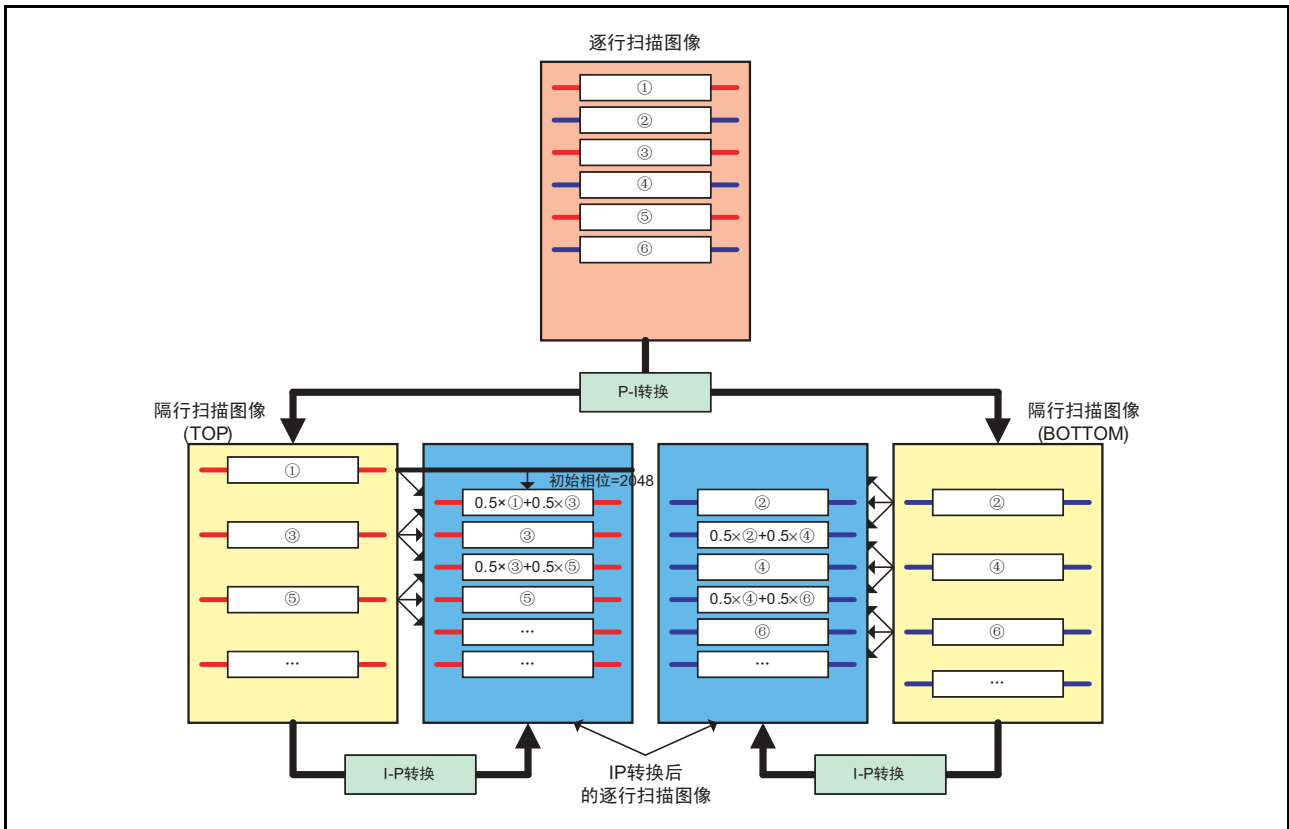


图 33.11 IP 转换处理概要图

(2) 场判别信号的控制

在输入隔行扫描信号时，通过垂直缩放对输出到放大控制部的场判别信号进行控制。

在输入逐行扫描信号或者通过缩小控制部进行垂直缩放时，因为输出到放大控制部的场判别信号是固定值，所以 RES_FLD_DLY_SEL 位的设定值可以是任意值。

表 33.20 场判别信号控制的设定值

输入信号	旋转控制	垂直处理	帧缓冲器	RES_FLD_DLY_SEL
逐行扫描	—	—	—	—
隔行扫描	正常	垂直缩小	—	—
		垂直放大	小于等于 1 面	0
	大于等于 2 面		1	
	180 度旋转	(水平输入 → 垂直输出) 缩小	—	—
(水平输入 → 垂直输出) 放大		小于等于 1 面	0	
		大于等于 2 面	1	

表 33.21 场判别信号的控制

寄存器名	位名	初始值	说明
SCL0_FRC5	RES_FLD_DLY_SEL	1	场判别信号的延迟控制 0: 无延迟 1: 1 个垂直期间的延迟

33.1.12 切边

对于缩放后的图像，在将 RES_V CUT 和 RES_H CUT 设定的上下左右端的图像剪切后输出。
能通过将 RES_DISP_ON 位置“1”，显示区域轮廓。

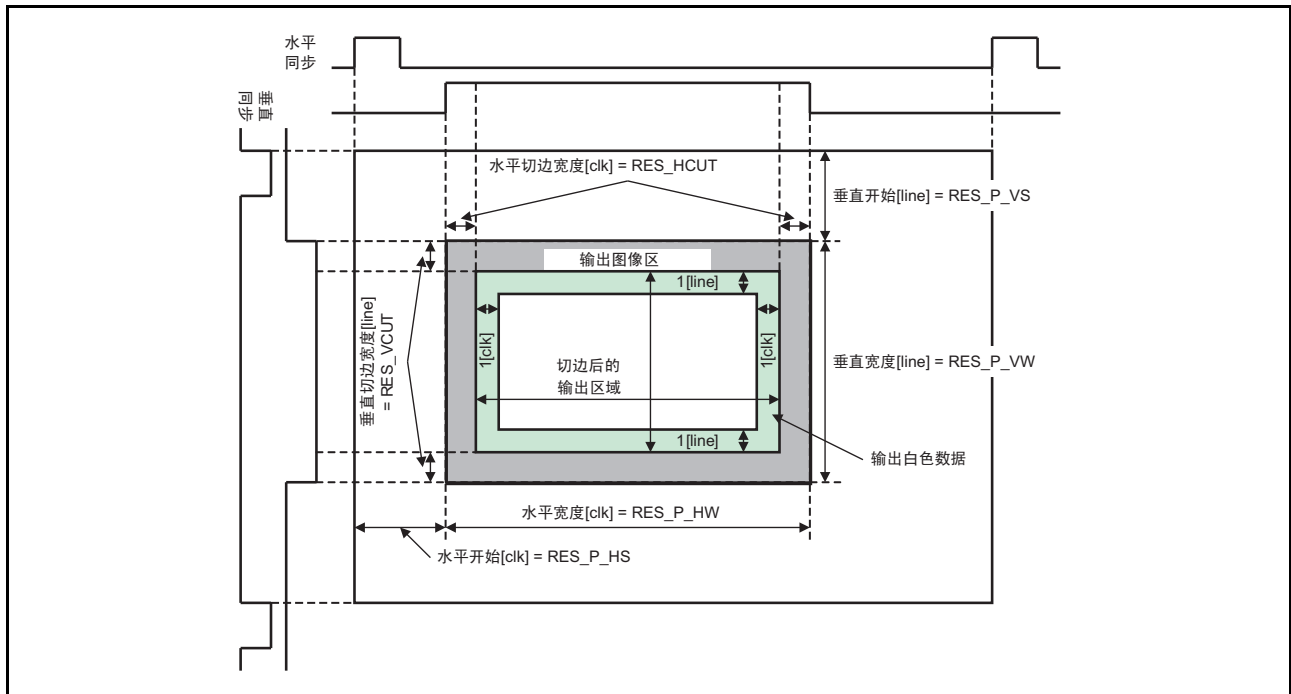


图 33.12 切边区域图（显示轮廓的情况）

表 33.22 切边控制

寄存器名	位名	初始值	说明
SCL0_US7	RES_H_CUT[7:0]	0	缩放后的图像水平方向左右剪切数 设定像素时钟个数。
SCL0_US7	RES_V_CUT[7:0]	0	缩放后的图像垂直方向上下剪切数 设定行数。
SCL0_US8	RES_DISP_ON	0	切边后的图像轮廓显示 ON/OFF 设定 0: 不显示轮廓 1: 显示轮廓

33.1.13 图像合成

对全屏幕有效期间重叠图像输出区进行输出。如果图像输出区小于全屏幕有效期间，就显示 RES_BK_COL_R 位、RES_BK_COL_G 位和 RES_BK_COL_B 位设定的背景色。

表 33.23 图像合成控制

寄存器名	位名	初始值	说明
SCL0_OVR1	RES_BK_COL_R[7:0]	128	背景显示色设定 R/Cr 信号 R: 8 位、无符号 (0 ~ 255[LSB]) Cr: 8 位、128 偏移二进制、无符号 (0 ~ 255[LSB])
SCL0_OVR1	RES_BK_COL_B[7:0]	128	背景显示色设定 B/Cb 信号 B: 无符号的 8 位 (0 ~ 255[LSB]) Cb: 8 位、128 偏移二进制、无符号 (0 ~ 255[LSB])
SCL0_OVR1	RES_BK_COL_G[7:0]	0	背景显示色设定 G/Y 信号 G/Y: 8 位、无符号 (0 ~ 255[LSB])

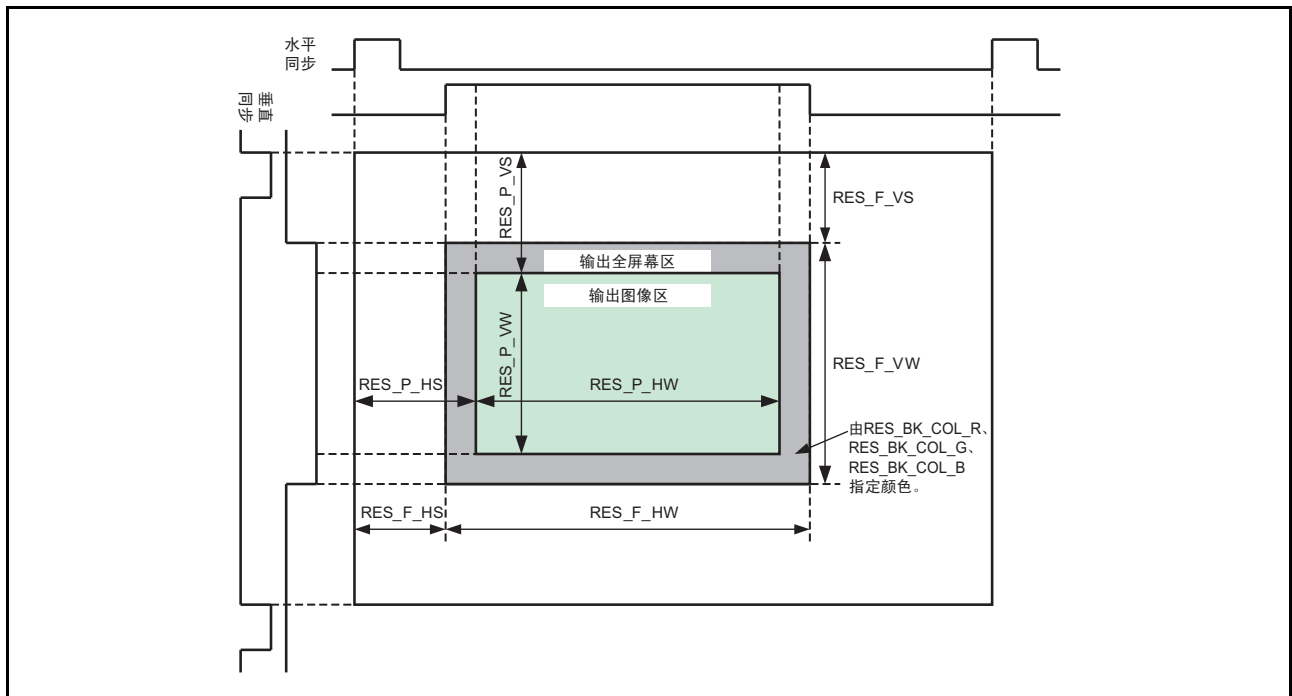


图 33.13 图像输出尺寸小于全屏幕时的区域图

33.1.14 帧缓冲器的写图像格式选择

选择帧缓冲器的写图像格式。

将 24 位 YCbCr 信号或者 24 位 RGB 信号输入到缩放部，通过 YCbCr422 (16 位)、RGB565 (16 位) 和 RGB888 (32 位) 写帧缓冲器。

RGB565 的位压缩处理可通过 RES_DTH_ON 位选择四舍五入或者 2×2 图案抖动。有关图案抖动的详细内容，请参照输出控制部的“36.1.7 抖动处理”。

在将 YCbCr 信号输入转换为 YCbCr422 后输出到失真校正引擎。

有关失真校正处理，请参照“38. 失真校正引擎”。

表 33.24 帧缓冲器的写操作模式设定表

RES_BITDEC_ON	RES_MD[1:0]	操作模式
0	2	RGB888 (正常、水平镜像)
1	1	RGB565 (正常、水平镜像、旋转)
0	0	YCbCr422 (正常、水平镜像、旋转)
1	0	YCbCr422 (失真校正)

表 33.25 图像格式的选择控制

寄存器名	位名	初始值	说明
SCL1_WR1	RES_MD[1:0]	0	帧缓冲器的写图像格式 0: YCbCr422 (16 位) 1: RGB565 (16 位) 2: RGB888 (24 (32) 位) 3: 禁止设定
SCL1_WR6	RES_BITDEC_ON	0	位压缩 ON/OFF 设定 0: OFF 1: ON
SCL1_WR6	RES_DTH_ON	0	抖动校正 ON/OFF 设定 0: OFF (四舍五入) 1: ON (2×2 图案抖动)

33.1.15 水平镜像和旋转处理

将缩小处理后的图像进行水平镜像和旋转处理，然后写到帧缓冲器。

水平镜像和旋转处理中的图像和处理模式的对应表如下表所示。

表 33.26 水平镜像和旋转处理的对应表

RES_DS_WR_MD[2:0]	写操作模式	YCbCr422	RGB565	RGB888
0	正常写	○	○	○
1	水平镜像写	○	○	○
2	90 度旋转写	○	○	×
3	180 度旋转写	○	○	×
4	270 度旋转写	○	○	×
5 ~ 7	禁止设定	—	—	—

表 33.27 水平镜像和旋转处理控制

寄存器名	位名	初始值	说明
SCL1_WR1	RES_DS_WR_MD[2:0]	0	帧缓冲器的写操作模式 0: 正常写 1: 水平镜像写 2: 90 度旋转写 3: 180 度旋转写 4: 270 度旋转写 5 ~ 7: 禁止设定

33.1.16 帧缓冲器的写处理

(1) 帧缓冲器的传送模式

保存图像和图形数据的帧缓冲器存取模式可选择 32 字节传送或者 128 字节传送。

表 33.28 帧缓冲器的传送模式

寄存器名	位名	初始值	说明
SCL1_WR1	RES_BST_MD	0	帧缓冲器写传送的突发长度 0: 32 字节传送 (突发长度为 4) 1: 128 字节传送 (突发长度为 16)

(2) 帧缓冲器的写控制

控制是否允许写帧缓冲器。

表 33.29 帧缓冲器的写控制

寄存器名	位名	初始值	说明
SCL1_WR5	RES_WENB	0	帧缓冲器的写允许 0: 禁止写 1: 允许写

(3) 帧缓冲器的写速率选择

根据输入信号的垂直周期, 帧缓冲器的写速率可选择 1/1、1/2、1/4 或者 1/8。
在写速率为 1/2、1/4 或者 1/8 时, 能选择写对象场。

表 33.30 帧缓冲器的写控制

寄存器名	位名	初始值	说明
SCL1_WR5	RES_FS_RATE[1:0]	0	写间隔 设定对于输入信号的写帧率。 0: 输入信号的 1/1 (RES_FLD_SEL 位的设定无效) 1: 输入信号的 1/2 2: 输入信号的 1/4 3: 输入信号的 1/8
SCL1_WR5	RES_FLD_SEL	0	写场选择 0: TOP 场 1: BOTTOM 场
SCL1_WR5	RES_INTER	1	场操作模式设定 0: 逐行扫描 1: 隔行扫描

(4) 帧缓冲器的写地址

通过基址、行偏移地址、帧偏移地址、1 行数据长度和 1 帧行数，设定帧缓冲器的地址。

以 32 字节为单位设定 RES_BASE[31:0] 位、RES_LN_OFF[14:0] 位和 RES_FLM_OFF[22:0] 位（低 5 位固定为“0”）。

因为在 128 字节传送时需要以 128 字节为单位进行设定，所以必须将地址控制寄存器的 [6:5] 固定为“0”。

1 行数据长度和 1 帧行数使用缩小控制部设定的寄存器值。

表 33.31 帧缓冲器的写地址控制

寄存器名	位名	初始值	说明
SCL1_WR2	RES_BASE[31:0]	0	帧缓冲器的基址 设定保存帧数据的帧缓冲器的起始地址。 在进行 32 字节传送时，必须将低 5 位固定为“0_0000”。 在进行 128 字节传送时，必须将低 7 位固定为“000_0000”。
SCL1_WR3	RES_LN_OFF[14:0]	2048	帧缓冲器的行偏移地址 设定计算行起始地址时的行偏移地址。 行 0: RES_BASE 行 1: RES_BASE+RES_LN_OFF×1 : 行 n: RES_BASE+RES_LN_OFF×n 在进行 32 字节传送时，必须将低 5 位固定为“0_0000”。 在进行 128 字节传送时，必须将低 7 位固定为“000_0000”。
SCL1_WR4	RES_FLM_OFF[22:0]	524288	帧缓冲器的帧偏移地址 设定计算各帧起始地址时的帧偏移地址。 缓冲器 0: RES_BASE 缓冲器 1: RES_BASE+RES_FLM_OFF×1 : 缓冲器 n: RES_BASE+RES_FLM_OFF×n 在进行 32 字节传送时，必须将低 5 位固定为“0_0000”。 在进行 128 字节传送时，必须将低 7 位固定为“000_0000”。

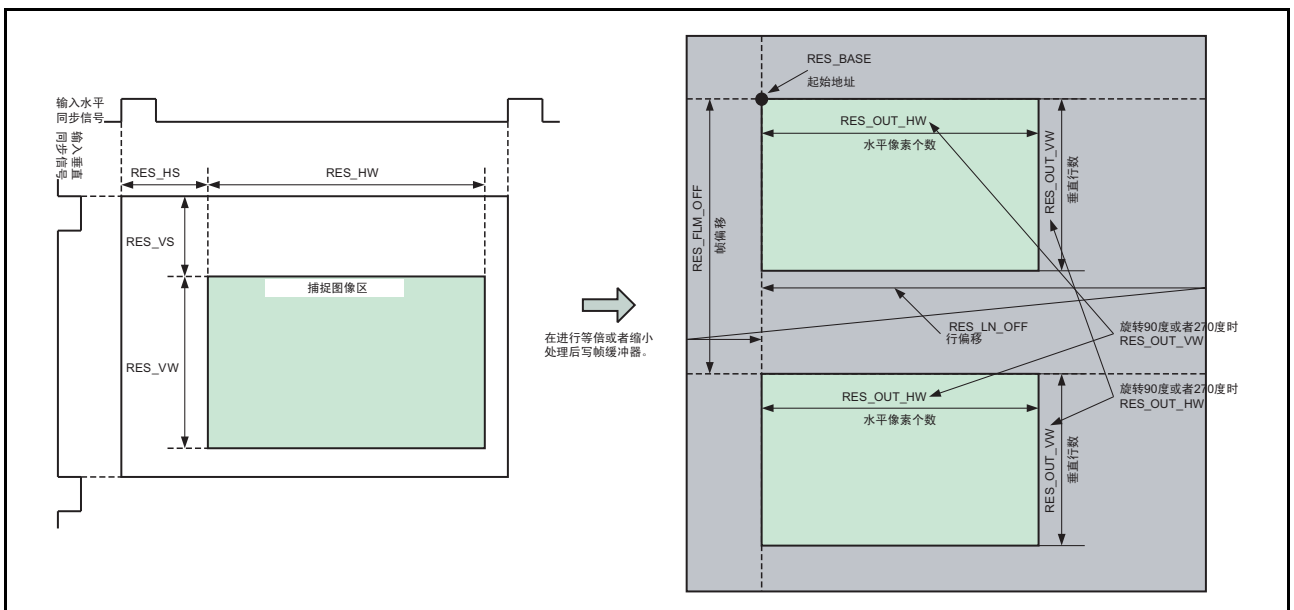


图 33.14 帧缓冲器的数据配置图

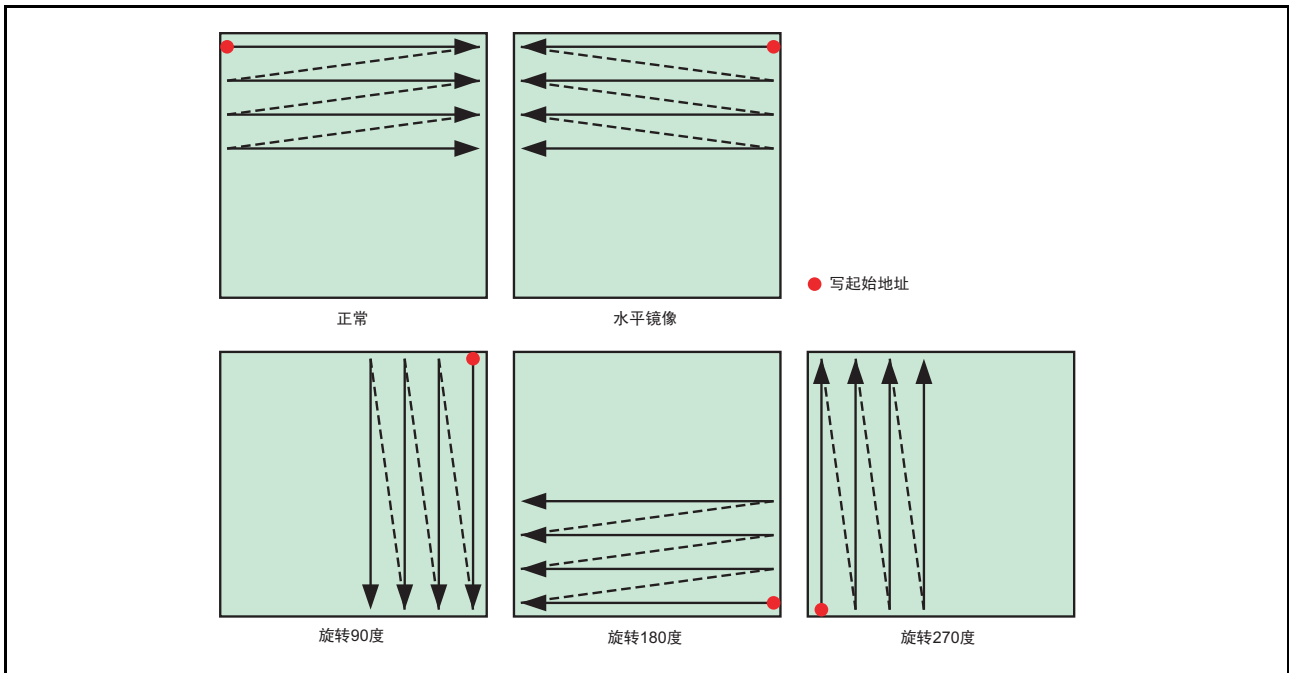


图 33.15 各种写模式的帧缓冲器数据配置图

(5) 帧缓冲器的管理

作为帧缓冲器，缩放部对应多帧。

根据 RES_FLM_NUM 位设定的帧数进行循环写。

在进行旋转处理时，必须将 RES_FLM_NUM 位置“1”（使用 2 帧）。

如果通过行的设定将帧缓冲器用于链状，就必须将 RES_FLM_NUM 位置“0”（使用 1 帧）并且将 RES_LOOP 位置“1”。

表 33.32 帧缓冲器的写控制

寄存器名	位名	初始值	说明
SCL1_WR3	RES_FLM_NUM[9:0]	1	写帧缓冲器的帧数 使用 RES_FLM_NUM+1 的帧数。 正常显示 : 置“0”或者置“1”（1面或者2面） 水平镜像显示 : 置“0”或者置“1”（1面或者2面） 旋转显示 : 置“1”（2面） 录像 : 设定“保存的帧数-1”
SCL1_WR1	RES_LOOP	0	帧缓冲器的写模式选择 0: 帧写模式 1: 行写模式（链状读）
SCL1_WR7	RES_FLM_CNT[9:0]	—	正在存取的帧号

(6) 缓冲器上溢处理

如果在写帧缓冲器的处理中因总线流量等问题而无法写，就将上溢中断输出到中断控制。

表 33.33 缓冲器的上溢检测

寄存器名	位名	初始值	说明
SCL1_WR7	RES_OVERFLOW	—	行缓冲器的上溢检测 1: 行缓冲器发生上溢 0: 行缓冲器未发生上溢

(7) 帧缓冲器的写结束标志

如果将 1 帧数据全部写到帧缓冲器，就将帧缓冲器写结束中断输出到中断控制。

33.1.17 放大处理和图形 (1) 处理的选择

因为放大处理和图形 (1) 处理为排他处理，所以不能同时读帧缓冲器。

如果显示输入的图像信号或者放大显示图形，就从放大控制部读帧缓冲器的数据。

但是，只有 RGB565、RGB888 和 YCbCr422 格式的图形才能通过放大控制部进行放大显示。

如果不放大显示图形，就从图形 (1) 处理部读帧缓冲器的数据。

通过 RES_IBUS_SYNC_SEL 位选择帧缓冲器的读同步信号寄存器和帧缓冲器的读大小设定寄存器。

表 33.34 放大处理和图形 (1) 处理的选择

缩放显示输出	RES_IBUS_SYNC_SEL	帧缓冲器的读同步信号	帧缓冲器的读大小设定	显示允许设定
输入图像信号显示 图形放大显示	0	放大控制部输出	RES_IN_VW RES_IN_HW	RES_P_VS RES_P_VW RES_P_HS RES_P_HW
图形显示	1	图形 (1) 输出	GR1_FLM_LNUM* GR1_HW*	GR1_GRC_VS GR1_GRC_VW GR1_GRC_HS GR1_GRC_HW

【注】 * 读大小为“寄存器设定值 +1”。

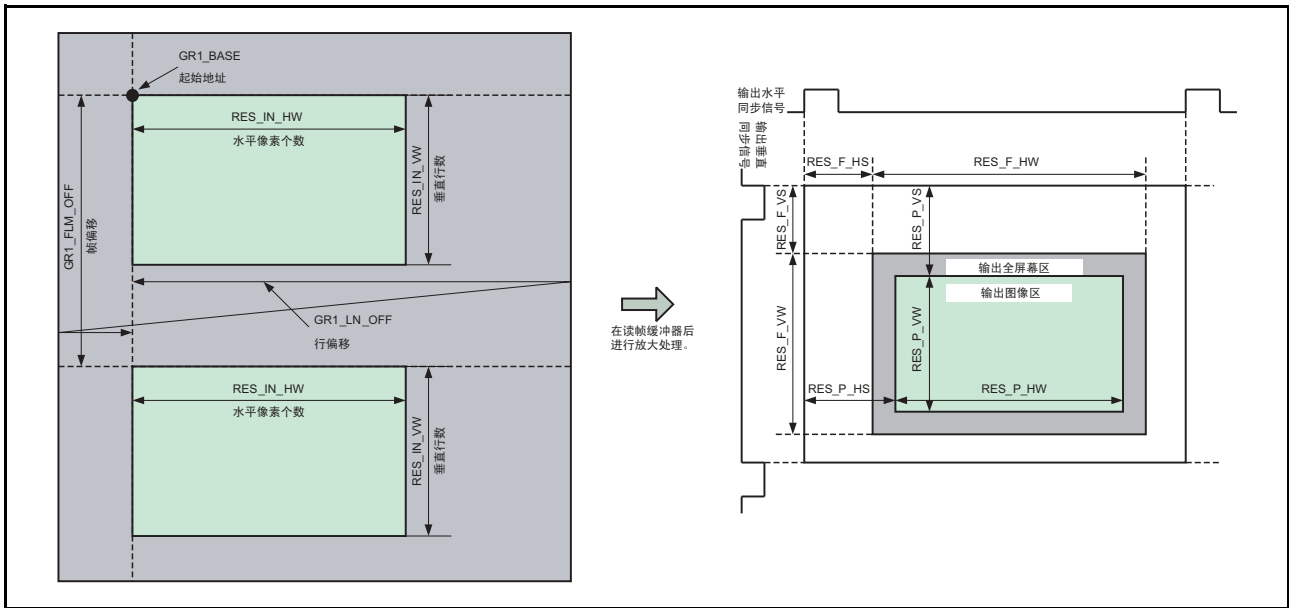


图 33.16 显示输入图像信号和放大显示图形时的区域设定

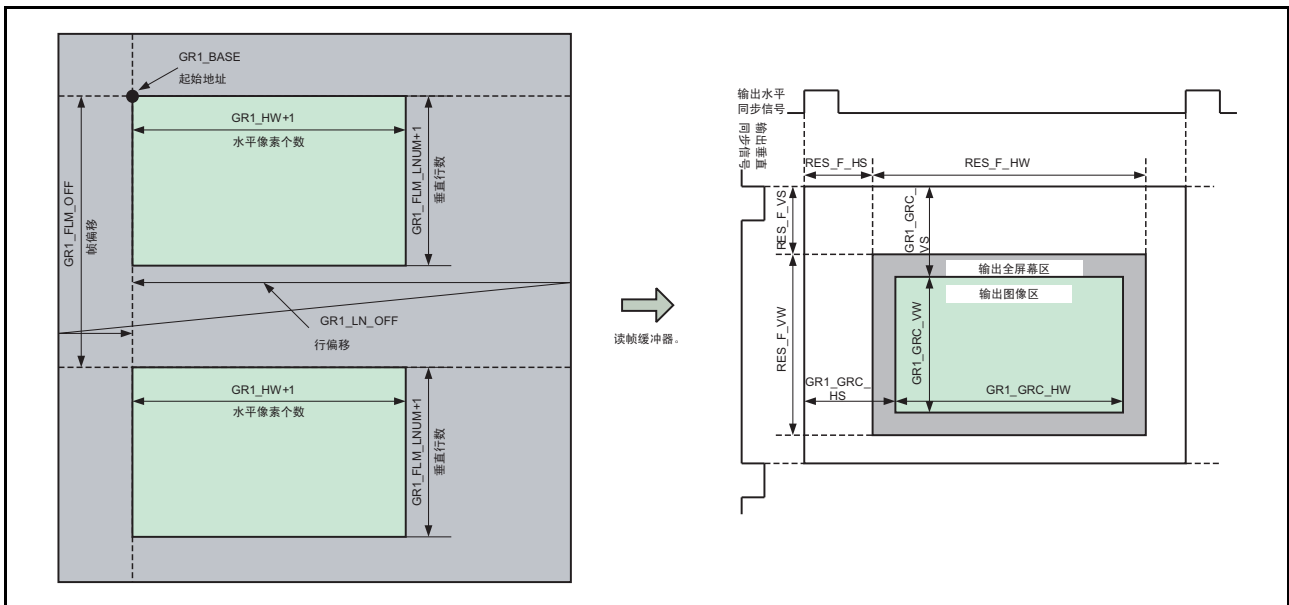


图 33.17 显示图形时的区域设定

表 33.35 放大处理和图形 (1) 处理的选择控制

寄存器名	位名	初始值	说明
SCL0_US8	RES_IBUS_SYNC_SEL	0	帧缓冲器读取部的同步信号选择 0: 放大控制部的同步信号 1: 图形部的同步信号

通过 GR1_DISP_SEL 位控制放大控制部显示（图像显示和图形放大显示）和图形显示的选择。
图形处理的详细内容请参照后述的图像合成部。

33.1.18 帧缓冲器的读处理

有关帧缓冲器的读处理和图形处理请参照后述的图像合成部。

33.2 寄存器说明

寄存器结构如表 33.36 所示。

【说明寄存器时的符号】

初始值 : 复位后的寄存器值

— : 不定值

R/W : 可读写。能读到写入值。

R/WC0 : 可读写。如果写“0”，该位就被初始化，但是忽视写“1”。

R/WC1 : 可读写。如果写“1”，该位就被初始化，但是忽视写“0”。

R : 只能读。写入值总是“0”。

—/W : 只能写。读取值为不定值。

表 33.36 缩放部的寄存器结构

名称	略称	R/W	初始值	地址	存取长度
SCL0 寄存器的更新控制寄存器	SCL0_UPDATE	R/WC1	H'0000 0000	H'FFFF 7500	32/16
屏蔽处理寄存器	SCL0_FRC1	R/W	H'0AF0 0001	H'FFFF 7504	32/16
欠缺补偿寄存器	SCL0_FRC2	R/W	H'0E10 0001	H'FFFF 7508	32/16
输出同步选择寄存器	SCL0_FRC3	R/W	H'0000 0001	H'FFFF 750C	32/16
自激周期寄存器	SCL0_FRC4	R/W	H'020C 031F	H'FFFF 7510	32/16
输出延迟控制寄存器	SCL0_FRC5	R/W	H'0000 0101	H'FFFF 7514	32/16
全屏幕垂直尺寸寄存器	SCL0_FRC6	R/W	H'0023 01E0	H'FFFF 7518	32/16
全屏幕水平尺寸寄存器	SCL0_FRC7	R/W	H'0090 0280	H'FFFF 751C	32/16
同步检测寄存器	SCL0_FRC9	R	H'0000 0000	H'FFFF 7524	32/16
缩小控制寄存器	SCL0_DS1	R/W	H'0000 0011	H'FFFF 752C	32/16
捕捉垂直尺寸寄存器	SCL0_DS2	R/W	H'0012 00F0	H'FFFF 7530	32/16
捕捉水平尺寸寄存器	SCL0_DS3	R/W	H'00F4 05A0	H'FFFF 7534	32/16
水平缩小寄存器	SCL0_DS4	R/W	H'1000 2408	H'FFFF 7538	32/16
垂直初始相位寄存器	SCL0_DS5	R/W	H'1800 0000	H'FFFF 753C	32/16
垂直缩放寄存器	SCL0_DS6	R/W	H'0000 07FC	H'FFFF 7540	32/16
缩小控制部输出尺寸寄存器	SCL0_DS7	R/W	H'00F0 0280	H'FFFF 7544	32/16
放大控制寄存器	SCL0_US1	R/W	H'0000 0011	H'FFFF 7548	32/16
输出图像垂直尺寸寄存器	SCL0_US2	R/W	H'0023 01E0	H'FFFF 754C	32/16
输出图像水平尺寸寄存器	SCL0_US3	R/W	H'0090 0280	H'FFFF 7550	32/16
放大控制部输入尺寸寄存器	SCL0_US4	R/W	H'00F0 0280	H'FFFF 7554	32/16
水平放大寄存器	SCL0_US5	R/W	H'0000 2408	H'FFFF 7558	32/16
水平放大初始相位寄存器	SCL0_US6	R/W	H'1000 0000	H'FFFF 755C	32/16
切边寄存器	SCL0_US7	R/W	H'0000 0000	H'FFFF 7560	32/16
帧缓冲器的读选择寄存器	SCL0_US8	R/W	H'0000 0000	H'FFFF 7564	32/16
背景色寄存器	SCL0_OVR1	R/W	H'0080 0080	H'FFFF 756C	32/16
SCL1 寄存器的更新控制寄存器	SCL1_UPDATE	R/WC1	H'0000 0000	H'FFFF 7580	32/16

名称	略称	R/W	初始值	地址	存取长度
写操作模式寄存器	SCL1_WR1	R/W	H'0000 0000	H'FFFF 7588	32/16
写地址寄存器 1	SCL1_WR2	R/W	H'0000 0000	H'FFFF 758C	32/16
写地址寄存器 2	SCL1_WR3	R/W	H'0800 0001	H'FFFF 7590	32/16
写地址寄存器 3	SCL1_WR4	R/W	H'0008 0000	H'FFFF 7594	32/16
帧减少寄存器	SCL1_WR5	R/W	H'0000 1000	H'FFFF 759C	32/16
位压缩寄存器	SCL1_WR6	R/W	H'0000 0000	H'FFFF 75A0	32/16
写检测寄存器	SCL1_WR7	R	H'0000 0000	H'FFFF 75A4	32/16
图形 (1) 寄存器的更新控制寄存器	GR1_UPDATE	R/WC1	H'0000 0000	H'FFFF 7600	32/16
帧缓冲器的读控制寄存器 (图形 (1))	GR1_FLM_RD	R/W	H'0000 0000	H'FFFF 7604	32/16
帧缓冲控制寄存器 1 (图形 (1))	GR1_FLM1	R/W	H'0000 0000	H'FFFF 7608	32/16
帧缓冲控制寄存器 2 (图形 (1))	GR1_FLM2	R/W	H'0000 0000	H'FFFF 760C	32/16
帧缓冲控制寄存器 3 (图形 (1))	GR1_FLM3	R/W	H'0800 0001	H'FFFF 7610	32/16
帧缓冲控制寄存器 4 (图形 (1))	GR1_FLM4	R/W	H'0008 0000	H'FFFF 7614	32/16
帧缓冲控制寄存器 5 (图形 (1))	GR1_FLM5	R/W	H'0000 03FF	H'FFFF 7618	32/16
帧缓冲控制寄存器 6 (图形 (1))	GR1_FLM6	R/W	H'8000 0000	H'FFFF 761C	32/16
α 混合控制寄存器 1 (图形 (1))	GR1_AB1	R/W	H'0000 0000	H'FFFF 7620	32/16
α 混合控制寄存器 2 (图形 (1))	GR1_AB2	R/W	H'0000 0000	H'FFFF 7624	32/16
α 混合控制寄存器 3 (图形 (1))	GR1_AB3	R/W	H'0000 0000	H'FFFF 7628	32/16
α 混合控制寄存器 7 (图形 (1))	GR1_AB7	R/W	H'00FF 0000	H'FFFF 7638	32/16
α 混合控制寄存器 8 (图形 (1))	GR1_AB8	R/W	H'0000 0000	H'FFFF 763C	32/16
α 混合控制寄存器 9 (图形 (1))	GR1_AB9	R/W	H'0000 0000	H'FFFF 7640	32/16
α 混合控制寄存器 10 (图形 (1))	GR1_AB10	R/W	H'0000 0000	H'FFFF 7644	32/16
α 混合控制寄存器 11 (图形 (1))	GR1_AB11	R/W	H'0000 0000	H'FFFF 7648	32/16
背景色控制寄存器 (图形 (1))	GR1_BASE	R/W	H'0000 8080	H'FFFF 764C	32/16
CLUT 表控制寄存器 (图形 (1))	GR1_CLUT	R/W	H'0000 0000	H'FFFF 7650	32/16

33.2.1 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	SCL0_VEN_D	SCL0_VEN_C	—	—	—	SCL0_UPDATE	—	—	—	SCL0_VEN_B	—	—	—	SCL0_VEN_A
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/WC1	R/WC1	R	R	R	R/WC1	R	R	R	R/WC1	R	R	R	R/WC1

位	位名	初始值	R/W	说明
31 ~ 14	—	全 0	R	保留位 读写值都为“0”。
13	SCL0_VEN_D	0	R/WC1	放大控制和帧缓冲器读控制的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器
12	SCL0_VEN_C	0	R/WC1	缩小控制和帧缓冲器写控制的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器
11 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	SCL0_UPDATE	0	R/WC1	同步控制的寄存器更新 0: 不更新寄存器 1: 更新寄存器
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	SCL0_VEN_B	0	R/WC1	同步控制和放大控制的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	SCL0_VEN_A	0	R/WC1	缩小控制的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器

33.2.2 屏蔽处理寄存器 (SCL0_FRC1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RES_VMASK [15:0]															
初始值:	0	0	0	0	1	0	1	0	1	1	1	1	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RES_VMASK_ON
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 16	RES_VMASK [15:0]	2800	R/W	垂直同步信号的重复屏蔽期间设定 以 128 倍像素时钟周期, 设定垂直同步信号的同步重复屏蔽期间。 屏蔽期间 [usec]=RES_VMASK×128÷ 像素时钟 [MHz]
15 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	RES_VMASK_ON	1	R/W	垂直同步信号的重复屏蔽控制 0: 重复屏蔽控制 OFF 1: 重复屏蔽控制 ON

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_UPDATE 位为“1”时更新此寄存器。

33.2.3 欠缺补偿寄存器 (SCL0_FRC2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RES_VLACK [15:0]															
初始值:	0	0	0	0	1	1	1	0	0	0	0	1	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RES_VLACK_ON
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 16	RES_VLACK [15:0]	3600	R/W	垂直同步信号的欠缺补偿期间设定 以 128 倍像素时钟周期, 设定从垂直同步信号输出同步欠缺补偿脉冲前的等待期间。 等待期间 [usec]=RES_VLACK×128÷ 像素时钟 [MHz]
15 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	RES_VLACK_ON	1	R/W	垂直同步信号的欠缺补偿控制 0: 欠缺补偿控制 OFF 1: 欠缺补偿控制 ON

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_UPDATE 位为“1”时更新此寄存器。

33.2.4 输出同步选择寄存器 (SCL0_FRC3)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RES_VS_SEL
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	RES_VS_SEL	1	R/W	输出垂直同步信号的选择 0: 外部输入垂直同步信号 1: 内部生成的自激垂直同步信号

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_UPDATE 位为“1”时更新此寄存器。

33.2.5 自激周期寄存器 (SCL0_FRC4)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	RES_FV[10:0]										
初始值:	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RES_FH[10:0]										
初始值:	0	0	0	0	0	0	1	1	0	0	0	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	RES_FV[10:0]	524	R/W	自激垂直同步信号的周期设定 自激垂直同步信号周期 = (RES_FV + 1) × 水平周期 [usec]
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	RES_FH[10:0]	799	R/W	水平同步信号的周期设定 水平同步信号周期 [usec] = (RES_FH + 1) ÷ 像素时钟 [MHz]

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_UPDATE 位为“1”时更新此寄存器。

33.2.6 输出延迟控制寄存器 (SCL0_FRC5)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	RES_FLD_DLY_SEL	RES_VSDLY[7:0]							
初始值:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	RES_FLD_DLY_SEL	1	R/W	场判别信号的延迟控制 0: 无延迟 1: 1 个垂直期间的延迟
7 ~ 0	RES_VSDLY[7:0]	1	R/W	垂直同步信号的延迟控制 以输出水平周期为单位, 调整垂直同步信号的延迟。 垂直同步信号延迟量 [usec]: RES_VSDLY× 输出水平周期 [usec]

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_VEN_B 位为“1”时更新此寄存器。

33.2.7 全屏幕垂直尺寸寄存器 (SCL0_FRC6)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	RES_F_VS[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RES_F_VW[10:0]										
初始值:	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	RES_F_VS[10:0]	35	R/W	全屏幕垂直允许信号的起始位置设定 (VSYNC+V 后沿 行数) 【注】 必须至少设定 4 行, 使 RES_F_VS+RES_F_VW 不超过 2039 行。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	RES_F_VW[10:0]	480	R/W	全屏幕垂直允许信号的宽度设定 (行数) 【注】 RES_F_VS+RES_F_VW 不能超过 2039 行。

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_VEN_B 位为“1”时更新此寄存器。

33.2.8 全屏幕水平尺寸寄存器 (SCL0_FRC7)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	RES_F_HS[10:0]										
初始值:	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RES_F_HW[10:0]										
初始值:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	RES_F_HS[10:0]	144	R/W	全屏水平允许信号的起始位置设定 (HSYNC+H 后沿 像素时钟个数) 【注】 必须至少设定 16 个时钟, 使 RES_F_HS+RES_F_HW 不超过 2015 个时钟。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	RES_F_HW[10:0]	640	R/W	全屏水平允许信号的宽度设定 (像素时钟个数) 【注】 RES_F_HS+RES_F_HW 不能超过 2015 个时钟。

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_VEN_B 位为“1”时更新此寄存器。

33.2.9 同步检测寄存器 (SCL0_FRC9)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	RES_QVLOCK	—	—	—	RES_QVLACK
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	RES_QVLOCK	0	R	垂直同步信号的锁定检测标志 1: 输入垂直同步信号至少连续 4 个垂直期间没有重复或者欠缺 0: 输入垂直同步信号有重复或者欠缺
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	RES_QVLACK	0	R	垂直同步信号的欠缺检测标志 1: 输入垂直同步信号有欠缺 0: 输入垂直同步信号无欠缺

33.2.10 缩小控制寄存器 (SCL0_DS1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	RES_DS_V_ON	—	—	—	RES_DS_H_ON
初始值:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	RES_DS_V_ON	1	R/W	垂直缩小 ON/OFF 设定 0: OFF 1: ON
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	RES_DS_H_ON	1	R/W	水平缩小 ON/OFF 设定 0: OFF 1: ON

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_VEN_A 位为“1”时更新此寄存器。

33.2.11 捕捉垂直尺寸寄存器 (SCL0_DS2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	RES_VS[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RES_VW[10:0]										
初始值:	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	RES_VS[10:0]	18	R/W	捕捉图像信号的垂直位置设定 (VSYNC+V 后沿 行数) 【注】 必须至少设定 4 行, 使 RES_VS+RES_VW 不超过 2039 行。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	RES_VW[10:0]	240	R/W	捕捉图像信号的垂直宽度 (行数) 【注】 RES_VS+RES_VW 不能超过 2039 行。

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_VEN_A 位为“1”时更新此寄存器。

33.2.12 捕捉水平尺寸寄存器 (SCL0_DS3)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	RES_HS[10:0]										
初始值:	0	0	0	0	0	0	0	0	1	1	1	1	0	1	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RES_HW[10:0]										
初始值:	0	0	0	0	0	1	0	1	1	0	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	RES_HS[10:0]	244	R/W	捕捉图像信号的水平位置设定 (HSYNC+H 后沿 图像时钟个数) 【注】 必须至少设定 16 个时钟, 使 RES_HS+RES_HW 不超过 2015 个时钟。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	RES_HW[10:0]	1440	R/W	捕捉图像信号的水平宽度 (图像时钟个数) 【注】 RES_HS+RES_HW 不能超过 2015 个时钟。

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_VEN_A 位为“1”时更新此寄存器。

33.2.13 水平缩小寄存器 (SCL0_DS4)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	RES_PFIL_SEL	RES_DS_H_INTERPOTYP	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RES_DS_H_RATIO[15:0]															
初始值:	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31、30	—	全 0	R	保留位 读写值都为“0”。
29	RES_PFIL_SEL	0	R/W	亮度信号前置滤波器的模式选择 0: 前置滤波器 OFF 1: 前置滤波器 ON (1/4+1/2+1/4)
28	RES_DS_H_INTERPOTYP	1	R/W	水平内插方法选择 0: 保持内插 1: 线性内插
27 ~ 16	—	全 0	R/W	保留位 读写值都为“0”。
15 ~ 0	RES_DS_H_RATIO[15:0]	9224	R/W	水平缩小率 [15:12]: 整数部 [11:0]: 小数部 round (RES_HW÷RES_OUT_HW×4096) RES_DS_H_RATIO<4096: 禁止设定 RES_DS_H_RATIO=4096: 等倍 RES_DS_H_RATIO>4096: 缩小

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_VEN_A 位为“1”时更新此寄存器。

33.2.14 垂直初始相位寄存器 (SCL0_DS5)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	RES_V INTERPOT YP	RES_TOP_INIPHASE [11:0]											
初始值:	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	RES_BTM_INIPHASE [11:0]											
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 29	—	全 0	R	保留位 读写值都为“0”。
28	RES_V_ INTERPOTYP	1	R/W	垂直内插方法选择 0: 保持内插 1: 线性内插
27 ~ 16	RES_TOP_ INIPHASE[11:0]	2048	R/W	TOP 场的垂直内插开始相位值 0 ~ 4095 (0 ~ 约 1.0)
15 ~ 12	—	全 0	R	保留位 读写值都为“0”。
11 ~ 0	RES_BTM_ INIPHASE[11:0]	0	R/W	BOTTOM 场的垂直内插开始相位值 0 ~ 4095 (0 ~ 约 1.0)

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_VEN_A 位或者 SCL0_VEN_B 位为“1”时更新此寄存器。

33.2.15 垂直缩放寄存器 (SCL0_DS6)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RES_V_RATIO[15:0]															
初始值:	0	0	0	0	0	1	1	1	1	1	1	1	1	1	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 16	—	全 0	R	保留位 读写值都为“0”。
15 ~ 0	RES_V_RATIO [15:0]	2044	R/W	垂直放大 / 缩小率 [15:12]: 整数部 [11:0]: 小数部 round($RES_VW \div RES_OUT_VW \times 4096$): 缩小时 iround($RES_IN_VW \div RES_P_VW \times 4096$): 放大时 RES_V_RATIO < 4096: 放大 RES_V_RATIO = 4096: 等倍 RES_V_RATIO > 4096: 缩小

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_VEN_A 位或者 SCL0_VEN_B 位为“1”时更新此寄存器。

33.2.16 缩小控制部输出尺寸寄存器 (SCL0_DS7)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	RES_OUT_VW[10:0]										
初始值:	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RES_OUT_HW[10:0]										
初始值:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	RES_OUT_VW [10:0]	240	R/W	缩小控制部输出的垂直有效行数 (行数) 将这些位的设定反映到写到帧缓冲器的行数。 当 SCL1_WR1.GR1_FLM_LOOP 位为“0” (帧写模式) 时, 必须指定 1 帧行数。 当 GR1_FLM5.GR1_FLM_LOOP 位为“1” (行写模式) 时, 必须指定要链状写的行数。 【注】 在设定时, 必须以 4 行为单位调整 RES_OUT_VW 并且满足 RES_OUT_VW ≤ RES_VW 的条件。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	RES_OUT_HW [10:0]	640	R/W	缩小控制部输出的水平有效像素个数 (图像时钟个数) 【注】 在设定时, 必须以 4 个像素为单位调整 RES_OUT_HW 位并且满足 RES_OUT_HW ≤ RES_HW 的条件。

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_VEN_A 位或者 SCL0_VEN_C 位为“1”时更新此寄存器。

33.2.17 放大控制寄存器 (SCL0_US1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	RES_US_V_ON	—	—	—	RES_US_H_ON
初始值:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	RES_US_V_ON	1	R/W	垂直放大 ON/OFF 设定 0: OFF 1: ON
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	RES_US_H_ON	1	R/W	水平放大 ON/OFF 设定 0: OFF 1: ON

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_VEN_B 位为“1”时更新此寄存器。

33.2.18 输出图像垂直尺寸寄存器 (SCL0_US2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	RES_P_VS[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RES_P_VW[10:0]										
初始值:	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	RES_P_VS[10:0]	35	R/W	图像输出的垂直允许信号起始位置设定 (VSYNC+V 后沿 行数) 【注】 必须至少设定 4 行, 使 RES_P_VS+RES_P_VW 不超过 2039 行。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	RES_P_VW[10:0]	480	R/W	图像输出的垂直允许信号宽度设定 (行数) 【注】 RES_P_VS+RES_P_VW 不能超过 2039 行。

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_VEN_B 位为“1”时更新此寄存器。

33.2.19 输出图像水平尺寸寄存器 (SCL0_US3)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	RES_P_HS[10:0]										
初始值:	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RES_P_HW[10:0]										
初始值:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	RES_P_HS[10:0]	144	R/W	图像输出的水平允许信号起始位置设定 (HSYNC+H 后沿 像素时钟个数) 【注】 必须至少设定 16 个时钟, 使 RES_P_HS+RES_P_HW 不超过 2015 个时钟。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	RES_P_HW[10:0]	640	R/W	图像输出的水平允许信号宽度设定 (像素时钟个数) 【注】 RES_P_HS+RES_P_HW 不能超过 2015 个时钟。

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_VEN_B 位为“1”时更新此寄存器。

33.2.20 放大控制部输入尺寸寄存器 (SCL0_US4)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	RES_IN_VW[10:0]										
初始值:	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RES_IN_HW[10:0]										
初始值:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	RES_IN_VW [10:0]	240	R/W	放大控制部输入的垂直有效行数 (行数)
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	RES_IN_HW [10:0]	480	R/W	放大控制部输入的水平有效像素个数 (像素时钟个数)

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_VEN_B 位或者 SCL0_VEN_D 位为“1”时更新此寄存器。

33.2.21 水平放大寄存器 (SCL0_US5)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RES_US_H_RATIO[15:0]															
初始值:	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 16	—	全 0	R	保留位 读写值都为“0”。
15 ~ 0	RES_US_H_ RATIO[15:0]	9224	R/W	水平放大率 [15:12]: 整数部 [11:0]: 小数部 $\text{round}(\text{RES_IN_HW} \div \text{RES_P_HW} \times 4096)$ RES_US_H_RATIO < 4096: 放大 RES_US_H_RATIO = 4096: 等倍 RES_US_H_RATIO > 4096: 禁止设定

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_VEN_B 位为“1”时更新此寄存器。

33.2.22 水平放大初始相位寄存器 (SCL0_US6)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	RES_US_H_INTERP_OTYP	RES_US_HT_INIPHASE[11:0]											
初始值:	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	RES_US_HB_INIPHASE[11:0]											
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 29	—	全 0	R	保留位 读写值都为“0”。
28	RES_US_H_INTERPOTYP	1	R/W	水平内插方法指示 0: 保持内插 1: 线性内插
27 ~ 16	RES_US_HT_INIPHASE[11:0]	0	R/W	TOP 场的水平内插开始相位值 0 ~ 4095 (0 ~ 约 1.0)
15 ~ 12	—	全 0	R	保留位 读写值都为“0”。
11 ~ 0	RES_US_HB_INIPHASE[11:0]	0	R/W	BOTTOM 场的水平内插开始相位值 0 ~ 4095 (0 ~ 约 1.0)

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_VEN_B 位为“1”时更新此寄存器。

33.2.23 切边寄存器 (SCL0_US7)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RES_HCUT[7:0]							RES_V CUT[7:0]								
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 16	—	全 0	R	保留位 读写值都为“0”。
15 ~ 8	RES_HCUT[7:0]	0	R/W	缩放后的图像水平方向左右剪切数 设定像素时钟个数。
7 ~ 0	RES_V CUT[7:0]	0	R/W	缩放后的图像的垂直方向上下剪切数 设定行数。

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_VEN_B 位为“1”时更新此寄存器。

33.2.24 帧缓冲器的读选择寄存器 (SCL0_US8)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	RES_IBUS_SYNC_SEL	—	—	—	RES_DISP_ON
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	RES_IBUS_SYNC_SEL	0	R/W	帧缓冲器读取部的同步信号选择 0: 放大控制部同步信号 1: 图形部同步信号
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	RES_DISP_ON	0	R/W	微调后的图像轮廓显示 ON/OFF 设定 0: 不显示轮廓 1: 显示轮廓

【注】 在SCL0寄存器的更新控制寄存器 (SCL0_UPDATE) 的SCL0_VEN_D位为“1”时更新RES_IBUS_SYNC_SEL位。
在SCL0寄存器的更新控制寄存器 (SCL0_UPDATE) 的SCL0_VEN_B位为“1”时更新RES_DISP_ON位。

33.2.25 背景色寄存器 (SCL0_OVR1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	RES_BK_COL_R[7:0]							
初始值:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RES_BK_COL_G[7:0]								RES_BK_COL_B[7:0]							
初始值:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23 ~ 16	RES_BK_ COL_R[7:0]	128	R/W	背景显示色设定 R/Cr 信号 R: 8 位、无符号 (0 ~ 255[LSB]) Cr: 8 位、128 偏移二进制、无符号 (0 ~ 255[LSB])
15 ~ 8	RES_BK_ COL_G[7:0]	0	R/W	背景显示色设定 G/Y 信号 G/Y: 8 位、无符号 (0 ~ 255[LSB])
7 ~ 0	RES_BK_ COL_B[7:0]	128	R/W	背景显示色设定 B/Cb 信号 B: 8 位、无符号 (0 ~ 255[LSB]) Cb: 8 位、128 偏移二进制、无符号 (0 ~ 255[LSB])

【注】 在全部 SCL0 寄存器的更新控制寄存器 (SCL0_UPDATE) 的 SCL0_VEN_B 位为“1”时更新此寄存器。

33.2.26 SCL1 寄存器的更新控制寄存器 (SCL1_UPDATE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SCL1_VEN_B	—	—	—	SCL1_VEN_A
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1

位	位名	初始值	R/W	说明
31 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	SCL1_VEN_B	0	R/WC1	帧缓冲器写控制的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	SCL1_VEN_A	0	R/WC1	帧缓冲器写控制的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器

33.2.27 写操作模式寄存器 (SCL1_WR1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RES_DS_WR_MD[2:0]		RES_MD[1:0]		RES_LOOP	RES_BST_MD	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 7	—	全 0	R	保留位 读写值都为“0”。
6 ~ 4	RES_DS_WR_MD[2:0]	0	R/W	帧缓冲器的写操作模式 0: 正常写 1: 水平镜像写 2: 90 度旋转写 3: 180 度旋转写 4: 270 度旋转写 5 ~ 7: 禁止设定
3 ~ 2	RES_MD[1:0]	0	R/W	帧缓冲器的写图像格式 0: YCbCr422 (16 位) 1: RGB565 (16 位) 2: RGB888 (24 (32) 位) 3: 禁止设定
1	RES_LOOP	0	R/W	帧缓冲器的写模式选择 0: 帧写模式 1: 行写模式 (链状读)
0	RES_BST_MD	0	R/W	帧缓冲器写传送的突发长度 0: 32 字节传送 (突发长度为 4) 1: 128 字节传送 (突发长度为 16)

【注】 在 SCL1 寄存器的更新控制寄存器 (SCL1_UPDATE) 的 SCL1_VEN_B 位为“1”时更新 RES_LOOP 位和 RES_BST_MD 位。

在 SCL1 寄存器的更新控制寄存器 (SCL1_UPDATE) 的 SCL1_VEN_A 位或者 SCL1_VEN_B 位为“1”时更新 RES_DS_WR_MD 位和 RES_MD 位。

33.2.28 写地址寄存器 1 (SCL1_WR2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RES_BASE[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RES_BASE[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 0	RES_BASE[31:0]	0	R/W	帧缓冲器的基址 设定保存帧数据的帧缓冲器的起始地址。 在进行 32 字节传送时, 必须将低 5 位固定为 “0_0000”。 在进行 128 字节传送时, 必须将低 7 位固定为 “000_0000”。

【注】 在全部 SCL1 寄存器的更新控制寄存器 (SCL1_UPDATE) 的 SCL1_VEN_B 位为 “1” 时更新此寄存器。

33.2.29 写地址寄存器 2 (SCL1_WR3)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	RES_LN_OFF[14:0]														
初始值:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RES_FLM_NUM[9:0]									
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31	—	0	R	保留位 读写值都为“0”。
30 ~ 16	RES_LN_OFF [14:0]	2048	R/W	帧缓冲器的行偏移地址 设定计算行起始地址时的行偏移地址。 行 0: RES_BASE 行 1: RES_BASE+RES_LN_OFF×1 : 行 n: RES_BASE+RES_LN_OFF×n 在进行 32 字节传送时, 必须将低 5 位固定为“0_0000”。 在进行 128 字节传送时, 必须将低 7 位固定为“000_0000”。
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	RES_FLM_NUM [9:0]	1	R/W	写帧缓冲器的帧数 使用 RES_FLM_NUM+1 的帧数。 正常显示 : 置“0”或者置“1”(1 面 or 2 面) 水平镜像显示 : 置“0”或者置“1”(1 面 or 2 面) 旋转显示 : 置“1”(2 面) 录像 : 设定“保存的帧数-1”

【注】 在全部 SCL1 寄存器的更新控制寄存器 (SCL1_UPDATE) 的 SCL1_VEN_B 位为“1”时更新此寄存器。

33.2.30 写地址寄存器 3 (SCL1_WR4)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	RES_FLM_OFF[22:16]						
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RES_FLM_OFF[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 23	—	全 0	R	保留位 读写值都为“0”。
22 ~ 0	RES_FLM_OFF [22:0]	524288	R/W	帧缓冲器的帧偏移地址 设定计算各帧起始地址时的帧偏移地址。 缓冲器 0: RES_BASE 缓冲器 1: RES_BASE+RES_FLM_OFF×1 : 缓冲器 n: RES_BASE+RES_FLM_OFF×n 在进行 32 字节传送时, 必须将低 5 位固定为“0_0000”。 在进行 128 字节传送时, 必须将低 7 位固定为“000_0000”。

【注】 在全部 SCL1 寄存器的更新控制寄存器 (SCL1_UPDATE) 的 SCL1_VEN_B 位为“1”时更新此寄存器。

33.2.31 帧减少寄存器 (SCL1_WR5)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	RES INTER	—	—	RES_FS RATE[1:0]	—	—	—	RES FLD_SEL	—	—	—	—	RES WENB
初始值:	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12	RES_INTER	1	R/W	场操作模式设定 0: 逐行扫描 1: 隔行扫描
11、10	—	全 0	R	保留位 读写值都为“0”。
9、8	RES_FS_RATE [1:0]	0	R/W	写间隔 设定对于输入信号的写帧率。 0: 输入信号的 1/1 (RES_FLD_SEL 位的设定无效) 1: 输入信号的 1/2 2: 输入信号的 1/4 3: 输入信号的 1/8
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	RES_FLD_SEL	0	R/W	写场选择 0: TOP 场 1: BOTTOM 场
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	RES_WENB	0	R/W	帧缓冲器的写允许 0: 禁止写 1: 允许写

【注】 在全部 SCL1 寄存器的更新控制寄存器 (SCL1_UPDATE) 的 SCL1_VEN_A 位为“1”时更新此寄存器。

33.2.32 位压缩寄存器 (SCL1_WR6)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	RES_DTH_ON	—	—	—	RES_BITDEC_ON
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	RES_DTH_ON	0	R/W	抖动校正 ON/OFF 设定 0: OFF (四舍五入) 1: ON (2×2 图案抖动)
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	RES_BITDEC_ON	0	R/W	位压缩 ON/OFF 设定 0: OFF 1: ON

【注】 在全部 SCL1 寄存器的更新控制寄存器 (SCL1_UPDATE) 的 SCL1_VEN_A 位为“1”时更新此寄存器。

33.2.33 写检测寄存器 (SCL1_WR7)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RES_OVER_FLOW	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	—	—	—	—	—	—	RES_FLM_CNT[9:0]									—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	

位	位名	初始值	R/W	说明
31 ~ 17	—	全 0	R	保留位 读写值都为“0”。
16	RES_OVERFLOW	0	R	行缓冲器的上溢检测 1: 行缓冲器发生上溢 0: 行缓冲器未发生上溢
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	RES_FLM_CNT [9:0]	0	R	正在存取的帧号

33.2.34 图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	GR1_P_VEN	—	—	—	GR1_IBUS_VEN
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1

位	位名	初始值	R/W	说明
31 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	GR1_P_VEN	0	R/WC1	图形显示的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GR1_IBUS_VEN	0	R/WC1	帧缓冲器读控制的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器

33.2.35 帧缓冲器的读控制寄存器 (图形 (1)) (GR1_FLM_RD)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR1_R_ENB
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GR1_R_ENB	0	R/W	帧缓冲器的读允许 0: 禁止读 1: 允许写

【注】 在全部图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE) 的 GR1_IBUS_VEN 位为“1”时更新此寄存器。

33.2.36 帧缓冲控制寄存器 1 (图形 (1)) (GR1_FLM1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR1_LN_OFF_DIR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR1_FLM_SEL[1:0]	—	—	—	GR1_IMR_FLM_INV	—	—	—	—	GR1_BST_MD
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 17	—	全 0	R	保留位 读写值都为“0”。
16	GR1_LN_OFF_DIR	0	R/W	帧缓冲器行偏移地址的方向设定 0: 递增行偏移地址 1: 递减行偏移地址
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9、8	GR1_FLM_SEL [1:0]	0	R/W	帧缓冲器地址设定信号的选择 0: 协同缩小处理 1: 选择 GR1_FLM_NUM 2: 协同失真校正处理 3: 禁止设定
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	GR1_IMR_FLM_INV	0	R/W	失真校正帧缓冲器号设定 0: 不交换读帧缓冲器号 1: 交换读帧缓冲器号
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GR1_BST_MD	0	R/W	帧缓冲器的突发传送模式 0: 32 字节 1: 128 字节

【注】 在图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE) 的 GR1_IBUS_VEN 位为“1”时更新 GR1_LN_OFF_DIR 位、GR1_FLM_SEL 位和 GR1_IMR_FLM_INV 位。

在图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE) 的 GR1_IBUS_VEN 位或者 GR1_P_VEN 位为“1”时更新 GR1_BST_MD 位。

33.2.37 帧缓冲控制寄存器 2 (图形 (1)) (GR1_FLM2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR1_BASE[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR1_BASE[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 0	GR1_BASE[31:0]	0	R/W	帧缓冲器的基址 设定保存帧数据的帧缓冲器的起始地址。 在进行 32 字节传送时, 参照 GR_BASE[4:3] 位; 在进行 128 字节传送时, 参照 GR_BASE[6:3] 位, 跳读起始行的数据。 必须将低 3 位固定为 “000”。

【注】 在全部图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE) 的 GR1_IBUS_VEN 位或者 GR1_P_VEN 位为 “1” 时更新此寄存器。

33.2.38 帧缓冲控制寄存器 3 (图形 (1)) (GR1_FLM3)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16																
	<table border="1" style="width:100%; border-collapse: collapse;"> <tr> <td style="width: 5%; text-align: center;">—</td> <td colspan="15" style="text-align: center;">GR1_LN_OFF[14:0]</td> </tr> </table>																—	GR1_LN_OFF[14:0]														
—	GR1_LN_OFF[14:0]																															
初始值:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0																
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W																
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																
	<table border="1" style="width:100%; border-collapse: collapse;"> <tr> <td style="width: 5%; text-align: center;">—</td><td style="width: 5%; text-align: center;">—</td><td style="width: 5%; text-align: center;">—</td><td style="width: 5%; text-align: center;">—</td><td style="width: 5%; text-align: center;">—</td><td style="width: 5%; text-align: center;">—</td><td colspan="10" style="text-align: center;">GR1_FLM_NUM[9:0]</td> </tr> </table>																—	—	—	—	—	—	GR1_FLM_NUM[9:0]									
—	—	—	—	—	—	GR1_FLM_NUM[9:0]																										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1																
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W																

位	位名	初始值	R/W	说明
31	—	0	R	保留位 读写值都为“0”。
30 ~ 26	GR1_LN_OFF [14:0]	2048	R/W	帧缓冲器的行偏移地址 设定计算行起始地址时的行偏移地址。 行 0: GR1_BASE 行 1: GR1_BASE+GR1_LN_OFF×1 : 行 n: GR1_BASE+GR1_LN_OFF×n 在进行 32 字节传送时, 必须将低 5 位固定为“0_0000”。 在进行 128 字节传送时, 必须将低 7 位固定为“000_0000”。
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	GR1_FLM_NUM [9:0]	1	R/W	帧缓冲器的帧号 在 GR_FLM_SEL 位为“1”时手动设定帧号。

【注】 在全部图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE) 的 GR1_IBUS_VEN 位为“1”时更新此寄存器。

33.2.39 帧缓冲控制寄存器 4 (图形 (1)) (GR1_FLM4)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	GR1_FLM_OFF[22:16]						
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR1_FLM_OFF[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 23	—	全 0	R	保留位 读写值都为“0”。
22 ~ 0	GR1_FLM_OFF [22:0]	524288	R/W	帧缓冲器的帧偏移地址 在使用多面帧缓冲器时, 设定计算各帧起始地址时的帧偏移地址。 缓冲器 0: GR1_BASE 缓冲器 1: GR1_BASE+GR1_FLM_OFF×1 : 缓冲器 n: GR1_BASE+GR1_FLM_OFF×n 在进行 32 字节传送时, 必须将低 5 位固定为“0_0000”。 在进行 128 字节传送时, 必须将低 7 位固定为“000_0000”。

【注】 在全部图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE) 的 GR1_IBUS_VEN 位为“1”时更新此寄存器。

33.2.40 帧缓冲控制寄存器 5 (图形 (1)) (GR1_FLM5)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	GR1_FLM_LNUM[9:0]									
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR1_FLM_LOOP[9:0]									
初始值:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 26	—	全 0	R	保留位 读写值都为“0”。
25 ~ 16	GR1_FLM_LNUM [9:0]	0	R/W	1 帧的行数设定 行数为 (GR1_FLM_LNUM+1) 行。
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	GR1_FLM_LOOP [9:0]	1023	R/W	链状读地址时的行数 行数为 (GR1_FLM_LOOP+1) 行。

【注】 在全部图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE) 的 GR1_IBUS_VEN 位为“1”时更新此寄存器。

33.2.41 帧缓冲控制寄存器 6 (图形 (1)) (GR1_FLM6)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR1_FORMAT [3:0]				—	—	GR1_HW [9:0]									
初始值:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR1_YCC_SWAP [2:0]		GR1_ENDIAN_ON	—	—	—	GR1_CNV444_MD	—	—	GR1_STA_POS [5:0]						
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 28	GR1_FORMAT [3:0]	8	R/W	帧缓冲器读信号的格式设定 0: RGB565 1: RGB888 2: αRGB1555 3: αRGB4444 4: αRGB8888 5: CLUT8 6: CLUT4 7: CLUT1 8: YCbCr422 9 ~ 15: 禁止设定
27、26	—	全 0	R	保留位 读写值都为“0”。
25 ~ 16	GR1_HW [9:0]	0	R/W	水平有效期间的宽度设定 宽度为 (GR1_HW+1) 个像素。 【注】 必须设定大于等于 2 的值。
15 ~ 13	GR1_YCC_SWAP [2:0]	0	R/W	YCbCr422 格式时的缓冲器读数据的交换控制 这些位在 GR1_ENDIAN_ON 位为“1”时有效。 0: Cb/Y0/Cr/Y1 1: Y0/Cb/Y1/Cr 2: Cr/Y0/Cb/Y1 3: Y0/Cr/Y1/Cb 4: Y1/Cr/Y0/Cb 5: Cr/Y/Cb/Y0 6: Y1/Cb/Y0/Cr 7: Cb/Y1/Cr/Y0
12	GR1_ENDIAN_ON	0	R/W	缓冲器读数据的字节序控制 ON/OFF 设定 0: OFF 1: ON
11 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	GR1_CNV444_MD	0	R/W	YCC422→YCbCr444 转换时的内插模式设定 0: 保持内插 1: 平均值内插

位	位名	初始值	R/W	说明
7、6	—	全 0	R	保留位 读写值都为“0”。
5 ~ 0	GR1_STA_POS[5:0]	0	R/W	数据的跳读量 从行首开始跳读 GR1_STA_POS 个数据。

【注】 在图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE) 的 GR1_P_VEN 位为“1”时更新 GR1_YCC_SWAP 位、GR1_ENDIAN_ON 位、GR1_CNV444 位和 GR1_STA_POS 位。
在图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE) 的 GR1_IBUS_VEN 位或者 GR1_P_VEN 位为“1”时更新 GR1_FORMAT 位和 GR1_HW 位。

33.2.42 α 混合控制寄存器 1 (图形 (1)) (GR1_AB1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	GR1_GRC_DISP_ON	—	—	GR1_DISP_SEL[1:0]	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	GR1_GRC_DISP_ON	0	R/W	图形图像区的轮廓显示 ON/OFF 设定 0: 不显示轮廓 1: 显示轮廓
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	GR1_DISP_SEL[1:0]	0	R/W	图形显示设定 0: 显示背景色 (寄存器 GR1_BASE) 1: 显示下层图形 在显示图像或者放大显示图形时, 必须选择显示背景色或者下层图形。 2: 显示当前图形 在显示图形时, 必须选择显示当前图形。 3: 混合显示下层图形和当前图形 * 【注】 * 在色键处理时进行设定。在色键处理时, 为了只显示当前图形, 必须将用于替换色键对象像素的 α 值和非色键对象像素的 α 值设定为“255”。

【注】 在全部图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE) 的 GR1_P_VEN 位为“1”时更新此寄存器。

33.2.43 α 混合控制寄存器 2 (图形 (1)) (GR1_AB2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR1_GRC_VS[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR1_GRC_VW[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	GR1_GRC_VS [10:0]	0	R/W	图形图像区的垂直起始位置设定 【注】 必须至少设定 4 行, 使 GR1_GRC_VS+GR1_GRC_VW 不超过 2039 行。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	GR1_GRC_VW [10:0]	0	R/W	图形图像区的垂直宽度设定

【注】 在全部图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE) 的 GR1_P_VEN 位为“1”时更新此寄存器。

33.2.44 α 混合控制寄存器 3 (图形 (1)) (GR1_AB3)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR1_GRC_HS[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR1_GRC_HW[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	GR1_GRC_HS [10:0]	0	R/W	图形图像区的水平起始位置设定 【注】 必须至少设定 16 个时钟，使 GR1_GRC_HS+GR1_GRC_HW 不超过 2015 个时钟。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	GR1_GRC_HW [10:0]	0	R/W	图形图像区的水平宽度设定 【注】 在显示水平宽度 1 个或者 2 个像素时，必须将 GR1_HW 位置“2”，并且将 GR1_GRC_HW 位置“1”（1 个像素）或者“2”（2 个像素）。

【注】 在全部图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE) 的 GR1_P_VEN 位为“1”时更新此寄存器。

33.2.45 α 混合控制寄存器 7 (图形 (1)) (GR1_AB7)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR1_CK_ON
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23 ~ 16	—	全 1	R	保留位 读写值都为“1”。
15 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GR1_CK_ON	0	R/W	CLUT 参照 /RGB 参照色键处理 ON/OFF 设定 0: OFF 1: ON

【注】 在全部图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE) 的 GR1_P_VEN 位为“1”时更新此寄存器。

33.2.46 α 混合控制寄存器 8 (图形 (1)) (GR1_AB8)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR1_CK_KCLUT[7:0]								GR1_CK_KG[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR1_CK_KB[7:0]								GR1_CK_KR[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GR1_CK_KCLUT[7:0]	0	R/W	CLUT 参照色键处理对象 CLUT 信号 CLUT: 8 位、无符号 (0 ~ 255[LSB])
23 ~ 16	GR1_CK_KG[7:0]	0	R/W	RGB 参照色键处理对象 G 信号 G: 8 位、无符号 (0 ~ 255[LSB])
15 ~ 8	GR1_CK_KB[7:0]	0	R/W	RGB 参照色键处理对象 B 信号 B: 8 位、无符号 (0 ~ 255[LSB])
7 ~ 0	GR1_CK_KR[7:0]	0	R/W	RGB 参照色键处理对象 R 信号 R: 8 位、无符号 (0 ~ 255[LSB])

【注】 在全部图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE) 的 GR1_P_VEN 位为“1”时更新此寄存器。

33.2.47 α 混合控制寄存器 9 (图形 (1)) (GR1_AB9)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR1_CK_A[7:0]								GR1_CK_G[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR1_CK_B[7:0]								GR1_CK_R[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GR1_CK_A[7:0]	0	R/W	RGB 参照色键处理替换后的 α 信号 α : 8 位、无符号 (0 ~ 255[LSB]) 【注】 为了只显示当前图形, 必须设定为 “255”。
23 ~ 16	GR1_CK_G[7:0]	0	R/W	RGB 参照色键处理替换后的 G 信号 G: 8 位、无符号 (0 ~ 255[LSB])
15 ~ 8	GR1_CK_B[7:0]	0	R/W	RGB 参照色键处理替换后的 B 信号 B: 8 位、无符号 (0 ~ 255[LSB])
7 ~ 0	GR1_CK_R[7:0]	0	R/W	RGB 参照色键处理替换后的 R 信号 R: 8 位、无符号 (0 ~ 255[LSB])

【注】 在全部图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE) 的 GR1_P_VEN 位为 “1” 时更新此寄存器。

33.2.48 α 混合控制寄存器 10 (图形 (1)) (GR1_AB10)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR1_A0[7:0]								GR1_G0[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR1_B0[7:0]								GR1_R0[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GR1_A0[7:0]	0	R/W	CLUT1 的 $\alpha 0$ 信号 在 CLUT1 格式并且 CLUT1 为 “0” 的情况下, 替换为 α 信号。 在 α RGB1555 格式并且 α 为 “0” 的情况下, 替换为 α 信号。 【注】 为了只显示当前图形, 必须设定为 “255”。
23 ~ 16	GR1_G0[7:0]	0	R/W	CLUT1 的 G0 信号 在 CLUT1 格式并且 CLUT1 为 “0” 的情况下, 替换为 G 信号。
15 ~ 8	GR1_B0[7:0]	0	R/W	CLUT1 的 B0 信号 在 CLUT1 格式并且 CLUT1 为 “0” 的情况下, 替换为 B 信号。
7 ~ 0	GR1_R0[7:0]	0	R/W	CLUT1 的 R0 信号 在 CLUT1 格式并且 CLUT1 为 “0” 的情况下, 替换为 R 信号。

【注】 在全部图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE) 的 GR1_P_VEN 位为 “1” 时更新此寄存器。

33.2.49 α 混合控制寄存器 11 (图形 (1)) (GR1_AB11)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR1_A1[7:0]								GR1_G1[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR1_B1[7:0]								GR1_R1[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始置	R/W	说明
31 ~ 24	GR1_A1[7:0]	0	R/W	CLUT1 的 α 1 信号 在 CLUT1 格式并且 CLUT1 为 “1” 的情况下, 替换为 α 信号。 在 α RGB1555 格式并且 α 为 “1” 的情况下, 替换为 α 信号。 【注】 为了只显示当前图形, 必须设定为 “255”。
23 ~ 16	GR1_G1[7:0]	0	R/W	CLUT1 的 G1 信号 在 CLUT1 格式并且 CLUT1 为 “1” 的情况下, 替换为 G 信号。
15 ~ 8	GR1_B1[7:0]	0	R/W	CLUT1 的 B1 信号 在 CLUT1 格式并且 CLUT1 为 “1” 的情况下, 替换为 B 信号。
7 ~ 0	GR1_R1[7:0]	0	R/W	CLUT1 的 R1 信号 在 CLUT1 格式并且 CLUT1 为 “1” 的情况下, 替换为 R 信号。

【注】 在全部图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE) 的 GR1_P_VEN 位为 “1” 时更新此寄存器。

33.2.50 背景色控制寄存器 (图形 (1)) (GR1_BASE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR1_BASE_G[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR1_BASE_B[7:0]								GR1_BASE_R[7:0]							
初始值:	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为 “0”。
23 ~ 16	GR1_BASE_G [7:0]	0	R/W	背景色 G 信号 G: 8 位、无符号 (0 ~ 255[LSB])
15 ~ 8	GR1_BASE_B [7:0]	128	R/W	背景色 B 信号 B: 8 位、无符号 (0 ~ 255[LSB])
7 ~ 0	GR1_BASE_R [7:0]	128	R/W	背景色 R 信号 R: 8 位、无符号 (0 ~ 255[LSB])

【注】 在全部图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE) 的 GR1_P_VEN 位为 “1” 时更新此寄存器。

33.2.51 CLUT 表控制寄存器 (图形 (1)) (GR1_CLUT)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR1_CLT_SEL
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 17	—	全 0	R	保留位 读写值都为“0”。
16	GR1_CLT_SEL	0	R/W	CLUT 表的选择信号 0: 选择 CLUT 表 0 参照 CLUT 表 0 并且展开为 α RGB8888。 CPU 能读写 CLUT 表 1。 1: 选择 CLUT 表 1 参照 CLUT 表 1 并且展开为 α RGB8888。 CPU 能读写 CLUT 表 0。
15 ~ 0	—	全 0	R	保留位 读写值都为“0”。

【注】 在全部图形 (1) 寄存器的更新控制寄存器 (GR1_UPDATE) 的 GR1_P_VEN 位为“1”时更新此寄存器。

33.3 使用方法

33.3.1 显示输入图像时的缩放设定例子

(1) 输入 / 输出视角

输入 / 输出视角信号的设定例子如表 33.37 所示。
过扫描率为 100%。

表 33.37 显示输入图像时的输入 / 输出视角

输入信号	输出信号	信号格式	旋转	缓冲器面	缩放滤波器
1440×240	240×320	YCbCr	正常	2 面	2TAP 线性

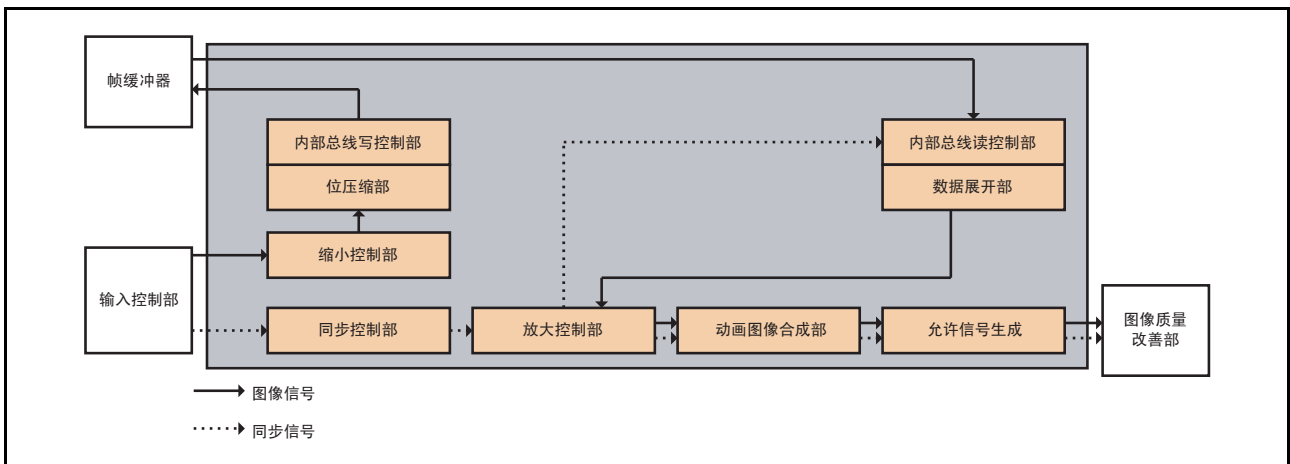


图 33.18 显示输入图像时的信号路径

(2) 水平缩放（水平缩小，缩放滤波器为 2TAP 线性）

通过折叠对策计算缩放率的方法如下：

$$RATIO_org = \text{round}(1440 \div 120 \times 4096) = 49152$$

$$\sigma = (49152 \times (120 - 1) - (1440 - 1) \times 4096) \div (120 - 1) = -378.62$$

$$\text{水平缩放率} = \text{round}(49152 - (-378.62)) = 49531$$

(3) 水平缩放（水平放大，缩放滤波器为 2TAP 线性）

通过折叠对策计算缩放率的方法如下：

$$RATIO_org = \text{round}(120 \div 240 \times 4096) = 2048$$

$$\sigma = (2048 \times (240 - 1) - (120 - 1) \times 4096) \div (240 - 1) = 8.56$$

$$\text{水平缩放率} = \text{round}(2048 - (8.56)) = 2039$$

(4) 垂直缩放（垂直放大，缩放滤波器为 2TAP 线性）

通过折叠对策计算缩放率的方法如下：

$$RATIO_org = \text{round}(240 \div 320 \times 4096) = 3072$$

$$\sigma = (3072 \times (32 - 1) - (240 - 1) \times 4096) \div (320 - 1) = 3.21$$

$$\text{垂直缩放率} = \text{round}(3072 - (3.21)) = 3069$$

(5) 帧缓冲器存取区的设定

因为在按比例缩小后将图像数据写到帧缓冲器，所以写尺寸为 640×240。

帧缓冲器的行偏移量至少需要 640 个像素，帧偏移至少需要行偏移量 ×240 的缓冲区。

将帧缓冲器的工作区设定为 1024×256。

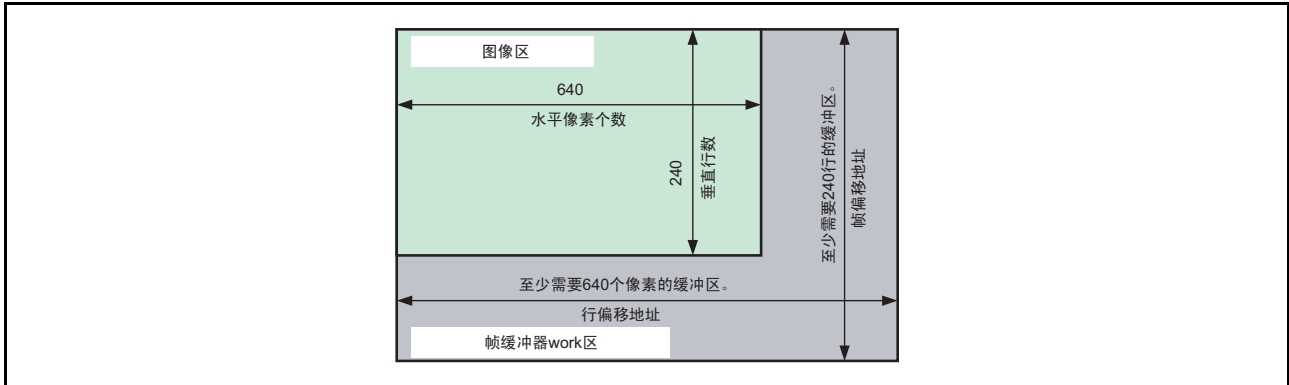


图 33.19 帧缓冲器存取区的设定

因为以 64 位为单位存取帧缓冲器，所以以 4 个像素为单位存取 YCC422 (16 位)。

行偏移地址的设定如下：

$RES_LN_OFF[14:0]=1024 \times 2=2048$

$GR1_LN_OFF[14:0]=1024 \times 2=2048$

帧偏移地址的设定如下：

$RES_FLM_OFF[22:0]=RES_LN_OFF[14:0] \times 256=524288$

$GR1_FLM_OFF[22:0]=GR1_LN_OFF[14:0] \times 256=524288$

(6) 寄存器的设定例子

表 33.38 输入 525i 图像和输出 VGA 尺寸图像时的寄存器设定例子

寄存器名	位名	设定值	备注
同步控制			
SCL0_FRC3	RES_VS_SEL	0	选择外部垂直同步。
SCL0_FRC4	RES_FH[10:0]	799	输出信号的水平周期宽度 (周期宽度 = 设定值 +1)
视角大小设定			
SCL0_DS2	RES_VS[10:0]	15	输入信号的捕捉垂直起始位置
SCL0_DS2	RES_VW[10:0]	240	输入信号的捕捉垂直宽度
SCL0_DS3	RES_HS[10:0]	244	输入信号的捕捉水平起始位置
SCL0_DS3	RES_HW[10:0]	1440	输入信号的捕捉水平宽度
SCL0_FRC6	RES_F_VS[10:0]	35	全屏幕垂直有效起始位置
SCL0_FRC6	RES_F_VW[10:0]	480	全屏幕垂直有效宽度
SCL0_FRC7	RES_F_HS[10:0]	144	全屏幕水平有效起始位置
SCL0_FRC7	RES_F_HW[10:0]	640	全屏幕水平有效宽度
SCL0_US2	RES_P_VS[10:0]	35	图像输出的垂直有效起始位置
SCL0_US2	RES_P_VW[10:0]	480	图像输出的垂直有效宽度
SCL0_US3	RES_P_HS[10:0]	144	图像输出的水平有效起始位置

寄存器名	位名	设定值	备注
SCL0_US3	RES_P_HW[10:0]	640	图像输出的水平有效宽度
缩放设定			
SCL0_DS4	RES_DS_H_RATIO[15:0]	9224	因为 RES_DS_H_RATIO \geq 4096, 所以进行水平缩小处理。
SCL0_DS1	RES_DS_H_ON	1	水平缩小 ON
SCL0_US1	RES_US_H_ON	0	水平放大 OFF
SCL0_US5	RES_US_H_RATIO[15:0]	4096	因为 RES_US_H_RATIO \geq 4096, 所以水平放大 OFF。
SCL0_DS1	RES_DS_V_ON	0	垂直缩小 OFF
SCL0_US1	RES_US_V_ON	1	垂直放大 ON
SCL0_DS6	RES_V_RATIO[15:0]	2044	因为 RES_V_RATIO $<$ 4096, 所以进行垂直放大处理。
SCL0_DS7	RES_OUT_VW[10:0]	240	因为垂直缩小 OFF, 所以是输入有效垂直宽度。
SCL0_DS7	RES_OUT_HW[10:0]	640	水平缩小后的水平图像尺寸
SCL0_US4	RES_IN_VW[10:0]	240	帧缓冲器读的垂直宽度
SCL0_US4	RES_IN_HW[10:0]	640	帧缓冲器读的水平宽度
IP 转换设定			
SCL0_DS5	RES_TOP_INIPHASE[11:0]	2048	将 TOP 场调整为 0.5 行相位。
SCL0_DS5	RES_BTM_INIPHASE[11:0]	0	BOTTOM 场不进行相位调整。
SCL0_FRC5	RES_FLD_DLY_SEL	1	使用 2 面帧缓冲器并且通过垂直放大处理进行 IP 转换。
帧缓冲器的写设定			
SCL1_WR1	RES_DS_WR_MD[2:0]	0	旋转控制为正常写。
SCL1_WR1	RES_MD[1:0]	0	帧缓冲器的写格式为 YCC422 (16 位)。
SCL1_WR2	RES_BASE[31:0]	0	帧缓冲器的写起始地址 (在设定例子中为“0”)
SCL1_WR3	RES_LN_OFF[14:0]	2048	帧缓冲器的写行偏移
SCL1_WR3	RES_FLM_NUM[9:0]	1	使用 2 面帧缓冲器。
SCL1_WR4	RES_FLN_OFF[22:0]	524288	帧缓冲器的写帧偏移
SCL1_WR5	RES_WENB	1	允许写帧缓冲器。
帧缓冲器的读设定			
GR1_FLM1	GR1_FLM_SEL[1:0]	0	选择帧缓冲器写输出的帧号。
GR1_FLM2	GR1_BASE[31:0]	0	取决于帧缓冲器的写设定。
GR1_FLM3	GR1_LN_OFF[14:0]	2048	取决于帧缓冲器的写设定。
GR1_FLM4	GR1_FLM_OFF[22:0]	524288	取决于帧缓冲器的写设定。
GR1_FLM6	GR1_FORMAT[3:0]	8	帧缓冲器的读格式为 YCC422。
GR1_FLM_RD	GR1_R_ENB	1	允许读帧缓冲器。
GR1_FLM6	GR1_CNV444_MD	1	YCC422→YCbCr444 转换时的平均值内插
放大处理选择			
SCL0_US8	RES_IBUS_SYNC_SEL	0	显示图像信号的放大处理输出。
GR1_AB1	GR1_DISP_SEL[1:0]	1	选择缩放显示。

33.3.2 显示图形时的缩放设定例子

(1) 图形视角

输入 / 输出视角信号的设定例子如表 33.39 所示。

表 33.39 显示图形时的输入 / 输出视角

图形大小	输出信号	图形信号格式
640×480	640×480	RGB888

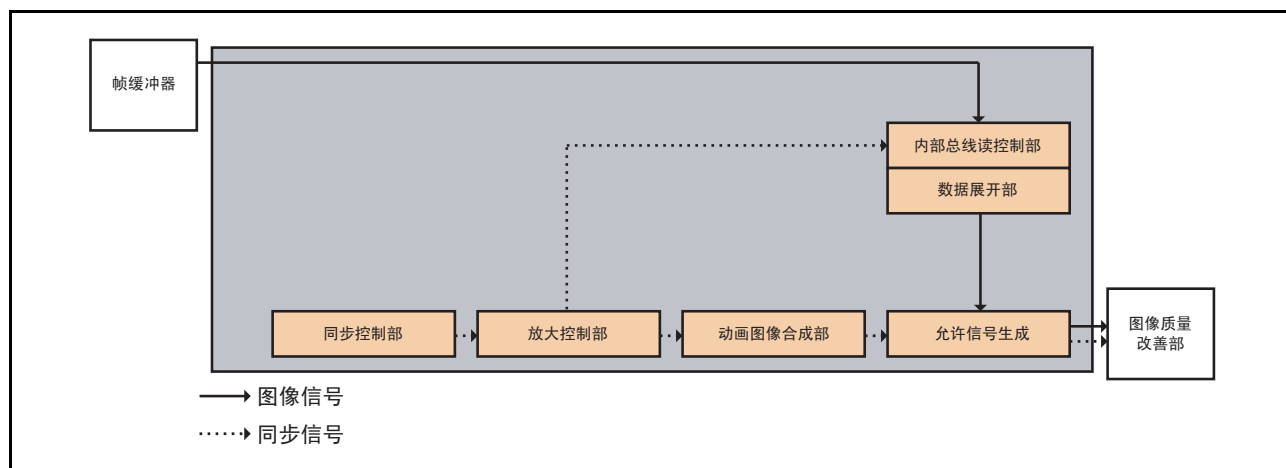


图 33.20 显示图形时的信号路径

(2) 帧缓冲器存取区的设定

保存图形数据的帧缓冲器至少需要展开为 640×480 的区域。

将帧缓冲器的图形展开区设定为 640×480。

因为以 64 位为单位存取帧缓冲器，所以以 2 个像素为单位存取 RGB888 (32 位)。

行偏移地址的设定如下：

$GR1_LN_OFF[14:0]=640 \times 4=2560$

帧偏移地址的设定如下：

$GR1_FLM_OFF[22:0]=GR1_LN_OFF[14:0] \times 480=1228800$

(3) 寄存器的设定例子

表 33.40 显示图形时的寄存器设定例子

寄存器名	位名	设定值	备注
同步控制			
SCL0_FRC3	RES_VS_SEL	1	选择自激垂直同步（如果有输入信号，也能选择外部同步）。
SCL0_FRC4	RES_FV[10:0]	524	输出信号的垂直周期宽度（周期宽度 = 设定值 + 1）
SCL0_FRC4	RES_FH[10:0]	799	输出信号的水平周期宽度（周期宽度 = 设定值 + 1）
视角大小设定			
SCL0_FRC6	RES_F_VS[10:0]	35	全屏幕垂直有效起始位置
SCL0_FRC6	RES_F_VW[10:0]	480	全屏幕垂直有效宽度
SCL0_FRC7	RES_F_HS[10:0]	144	全屏幕水平有效起始位置
SCL0_FRC7	RES_F_HW[10:0]	640	全屏幕水平有效宽度
GR1_AB2	GR1_GRC_VS[10:0]	35	图形输出的垂直有效起始位置
GR1_AB2	GR1_GRC_VW[10:0]	480	图形输出的垂直有效宽度
GR1_AB3	GR1_GRC_HS[10:0]	144	图形输出的水平有效起始位置
GR1_AB3	GR1_GRC_HW[10:0]	640	图形输出的水平有效宽度
帧缓冲器的读设定			
GR1_FLM1	GR1_FLM_SEL[1:0]	1	通过设定寄存器来选择帧号。
GR1_FLM3	GR1_FLM_NUM[9:0]	0	设定帧缓冲器的帧号（在设定例子中为“0”）。
GR1_FLM5	GR1_FLM_LNUM[9:0]	479	设定图形的行数（行数 = 设定值 + 1）。
GR1_FLM6	GR1_HW[9:0]	639	设定图形的水平有效宽度（有效宽度 = 设定值 + 1）。
GR1_FLM2	GR1_BASE[31:0]	0	取决于图形展开设定（在设定例子中为“0”）。
GR1_FLM3	GR1_LN_OFF[14:0]	2560	取决于图形展开设定。
GR1_FLM4	GR1_FLM_OFF[22:0]	1228800	取决于图形展开设定。
GR1_FLM6	GR1_FORMAT[3:0]	1	帧缓冲器读格式为 RGB888。
GR1_FLM_RD	GR1_R_ENB	1	允许读帧缓冲器。
放大处理选择			
SCL0_US8	RES_IBUS_SYNC_SEL	1	显示图形输出。
GR1_AB1	GR1_DISP_SEL[1:0]	2	选择图形显示。

33.3.3 放大显示图形时的缩放设定例子

(1) 输入 / 输出视角

输入 / 输出视角信号的设定例子如表 33.41 所示。

表 33.41 放大显示图形时的输入 / 输出视角

图形大小	输出信号	图形信号格式
640×480	800×600	RGB565

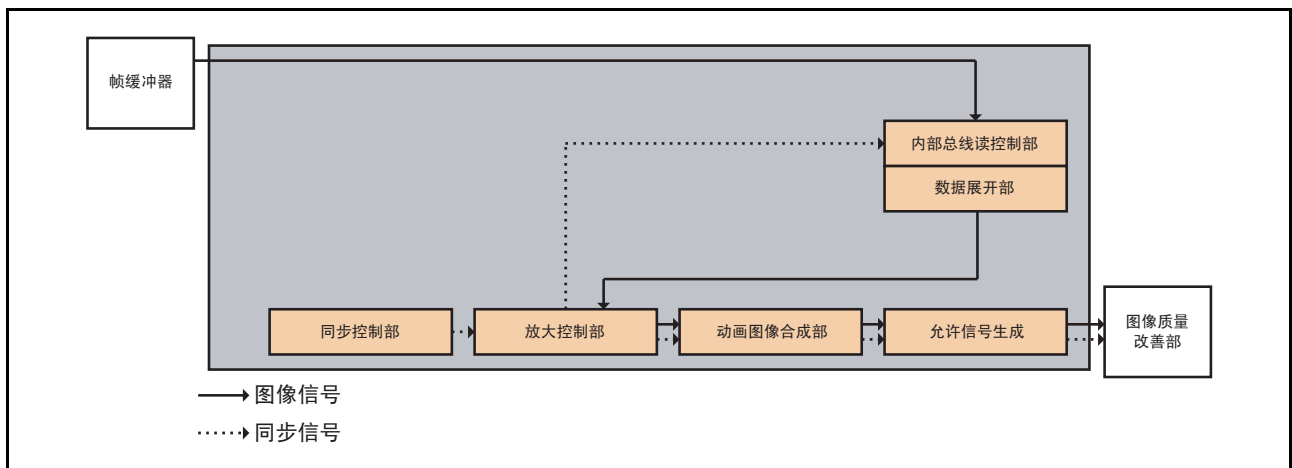


图 33.21 放大显示图形时的信号路径

(2) 水平缩放（水平放大，缩放滤波器为 2TAP 线性）

通过折叠对策计算缩放率的方法如下：

$$\text{RATIO_org} = \text{round}(640 \div 800 \times 4096) = 3277$$

$$\sigma = (3277 \times (800 - 1) - (640 - 1) \times 4096) \div (800 - 1) = 1.23$$

$$\text{水平缩放率} = \text{round}(3277 - (1.23)) = 3276$$

(3) 垂直缩放（垂直放大，缩放滤波器为 2TAP 线性）

通过折叠对策计算缩放率的方法如下：

$$\text{RATIO_org} = \text{round}(480 \div 600 \times 4096) = 3277$$

$$\sigma = (3277 \times (600 - 1) - (480 - 1) \times 4096) \div (600 - 1) = 1.57$$

$$\text{垂直缩放率} = \text{round}(3277 - (1.57)) = 3275$$

(4) 帧缓冲器存取区的设定

保存图形数据的帧缓冲器至少需要展开为 640×480 的区域。

将帧缓冲器的图形展开区域设定为 640×480。

因为以 64 位为单位存取帧缓冲器，所以以 4 个像素为单位存取 RGB565（16 位）。

行偏移地址的设定如下：

$$\text{GR1_LN_OFF}[14:0] = 640 \times 2 = 1280$$

帧偏移地址的设定如下：

$$\text{GR1_FLM_OFF}[22:0] = \text{GR1_LN_OFF}[14:0] \times 480 = 614400$$

(5) 寄存器的设定例子

表 33.42 放大显示图形时的寄存器设定例子

寄存器名	位名	设定值	备注
同步控制			
SCL0_FRC3	RES_VS_SEL	1	选择自激垂直同步 (如果有输入信号, 也能选择外部同步)
SCL0_FRC4	RES_FV[10:0]	668	输出信号的垂直周期宽度 (周期宽度 = 设定值 +1)
SCL0_FRC4	RES_FH[10:0]	1040	输出信号的水平周期宽度 (周期宽度 = 设定值 +1)
视角大小设定			
SCL0_FRC6	RES_F_VS[10:0]	27	全屏幕垂直有效起始位置
SCL0_FRC6	RES_F_VW[10:0]	600	全屏幕垂直有效宽度
SCL0_FRC7	RES_F_HS[10:0]	216	全屏幕水平有效起始位置
SCL0_FRC7	RES_F_HW[10:0]	800	全屏幕水平有效宽度
SCL0_US2	RES_P_VS[10:0]	27	图像输出的垂直有效起始位置
SCL0_US2	RES_P_VW[10:0]	600	图像输出的垂直有效宽度
SCL0_US3	RES_P_HS[10:0]	216	图像输出的水平有效起始位置
SCL0_US3	RES_P_HW[10:0]	800	图像输出的水平有效宽度
缩放设定			
SCL0_US5	RES_US_H_RATIO[15:0]	3276	因为 RES_US_H_RATIO < 4096, 所以进行水平放大处理。
SCL0_DS6	RES_V_RATIO[15:0]	3275	因为 RES_V_RATIO < 4096, 所以进行垂直放大处理。
SCL0_US1	RES_US_H_ON	1	水平放大 ON
SCL0_US1	RES_US_V_ON	1	垂直放大 ON
SCL0_US4	RES_IN_VW[10:0]	480	帧缓冲器读的垂直宽度
SCL0_US4	RES_IN_HW[10:0]	640	帧缓冲器读的水平宽度
帧缓冲器的读设定			
GR1_FLM1	GR1_FLM_SEL[1:0]	1	通过设定寄存器来选择帧号。
GR1_FLM3	GR1_FLM_NUM[9:0]	0	设定帧缓冲器的帧号 (在设定例子中为“0”)。
GR1_FLM2	GR1_BASE[31:0]	0	取决于图形展开设定 (在设定例子中为“0”)。
GR1_FLM3	GR1_LN_OFF[14:0]	1280	取决于图形展开设定。
GR1_FLM4	GR1_FLM_OFF[22:0]	614400	取决于图形展开设定。
GR1_FLM6	GR1_FORMAT[3:0]	0	帧缓冲器读格式为 RGB565。
GR1_FLM_RD	GR1_R_ENB	1	允许读帧缓冲器。
放大处理选择			
SCL0_US8	RES_IBUS_SYNC_SEL	0	显示图像信号的放大处理输出。
GR1_AB1	GR1_DISP_SEL[1:0]	1	选择缩放显示。

34. 视频显示控制器 4 (4) 图像质量改善部

34.1 图像质量改善功能

34.1.1 功能概要

图像质量改善部对缩放后的 YCbCr 信号进行黑色信号展宽、LTI/ 清晰度处理以及通过色彩矩阵进行 YCbCr 到 GBR 的转换。

在输入 RGB 信号时，不改善图像质量。
 图像质量改善部的功能框图如下图所示。

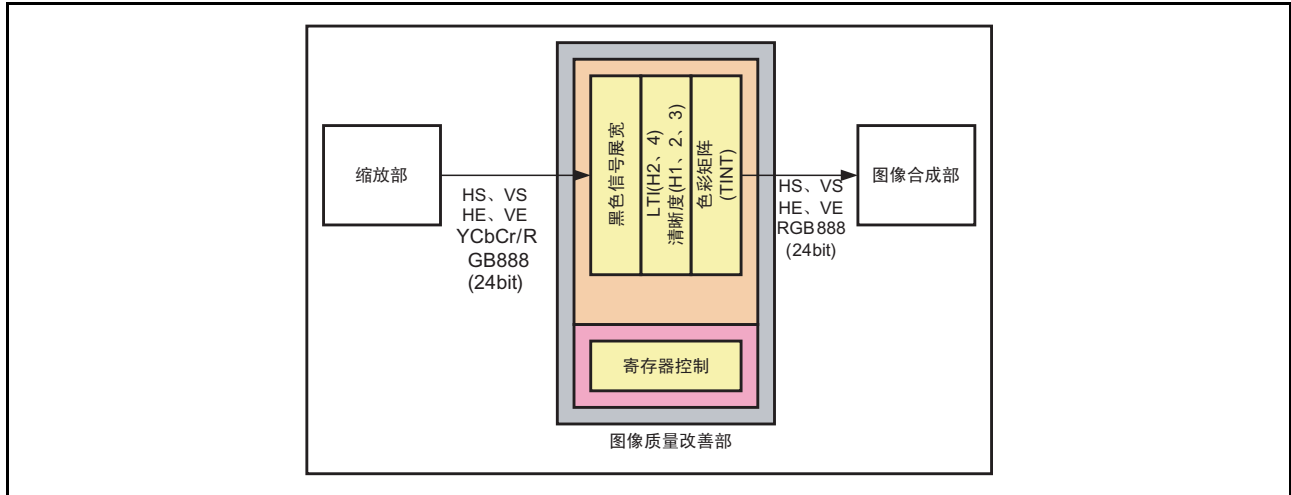


图 34.1 图像质量改善部的功能框图

34.1.2 寄存器的更新控制

图像质量改善部的控制寄存器全部通过垂直同步信号管理更新时序。

在将更新控制寄存器置“1”后，在垂直同步信号的上升沿反映各种寄存器。反映后，更新控制寄存器自动清“0”。

表 34.1 寄存器的更新控制

寄存器名	位名	初始值	说明
ADJ_UPDATE	ADJ_VEN	0	图像质量改善部的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器

34.1.3 黑色信号展宽

黑色信号展宽是指对 YCbCr 格式的输入图像信号进行 Y 信号黑色信号展宽校正。
 通过调整时间常数、深度（增益）和起始点，对 Y 信号进行校正。
 黑色信号展宽校正图如图 34.2 所示。

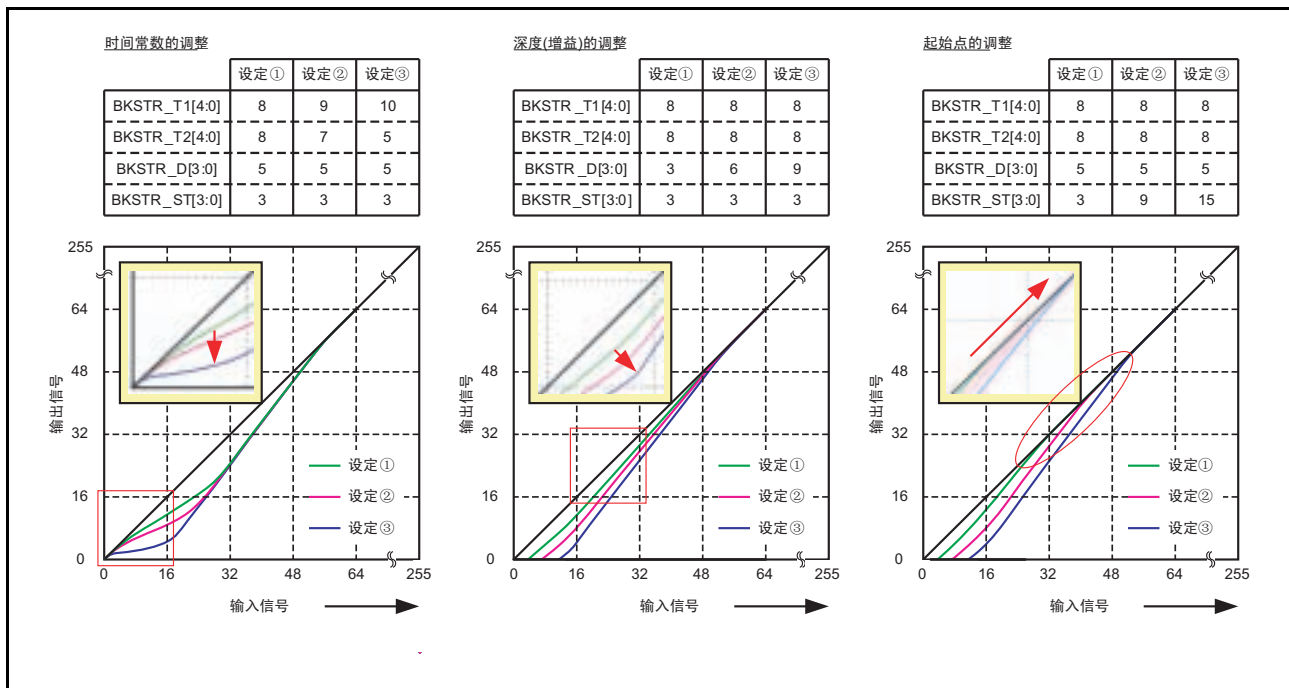


图 34.2 黑色信号展宽校正图 (设定例子)

表 34.2 黑色信号展宽的控制

寄存器名	位名	初始值	说明
ADJ_BKSTR_SET	BKSTR_ON	0	黑色信号展宽的 ON/OFF 控制 0: 黑色信号展宽 OFF 1: 黑色信号展宽 ON
ADJ_BKSTR_SET	BKSTR_ST[3:0]	0	黑色信号展宽的起始点指定 0 (低) ~ 15 (高)
ADJ_BKSTR_SET	BKSTR_T1[4:0]	0	黑色信号展宽的时间常数 (T1) 0 (小) ~ 31 (大)
ADJ_BKSTR_SET	BKSTR_T2[4:0]	0	黑色信号展宽的时间常数 (T2) 0 (小) ~ 30 (大)、31: 禁止设定
ADJ_BKSTR_SET	BKSTR_D[3:0]	0	黑色信号展宽的深度 0 (浅) ~ 15 (深)

34.1.4 增强器

通过使用增强器，对缩放后的 Y 信号输入进行水平方向的瞬态改善（LTI）和清晰度处理。

(1) 增强器的区域指定

以水平同步信号和垂直同步信号的上升沿为基准指定增强器的运行范围。ENH_HS 至少需要 4 个时钟，ENH_VS 至少需要 2 行。增强器的区域设定图如图 34.3 所示。

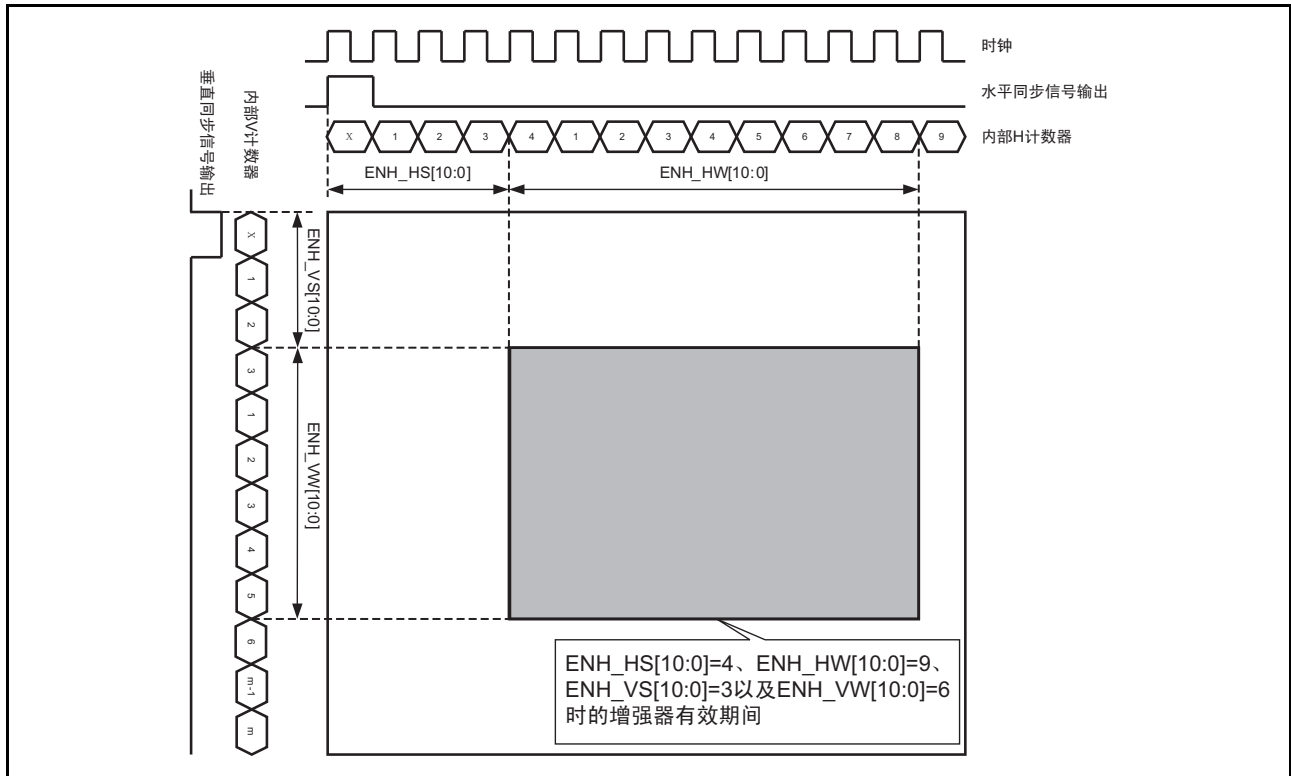


图 34.3 增强器有效期间

能通过将 ENH_DISP_ON 位置“1”，显示增强器有效区域的轮廓。

表 34.3 增强器的区域控制

寄存器名	位名	初始值	说明
ADJ_ENH_TIM1	ENH_MD	1	运行模式 0: RGB 模式 1: YCbCr 模式
ADJ_ENH_TIM2	ENH_VS[10:0]	0	增强器有效区域的垂直有效图像区的起始位置设定 【注】 必须至少设定 2 行。
ADJ_ENH_TIM2	ENH_VW[10:0]	0	增强器有效区域的垂直有效图像区的宽度设定
ADJ_ENH_TIM3	ENH_HS[10:0]	0	增强器有效区域的水平有效图像区的起始位置设定 【注】 必须至少设定 4 个时钟。
ADJ_ENH_TIM3	ENH_HW[10:0]	0	增强器有效区域的水平有效图像区的宽度设定
ADJ_ENH_TIM1	ENH_DISP_ON	0	增强器有效区域的轮廓显示 ON/OFF 设定 0: 不显示轮廓 1: 显示轮廓

(2) LTI (Luminance Transient Improvement)

LTI 是指对 Y 信号输入进行水平方向的瞬态改善。
消隐的瞬态改善变为 OFF 状态。

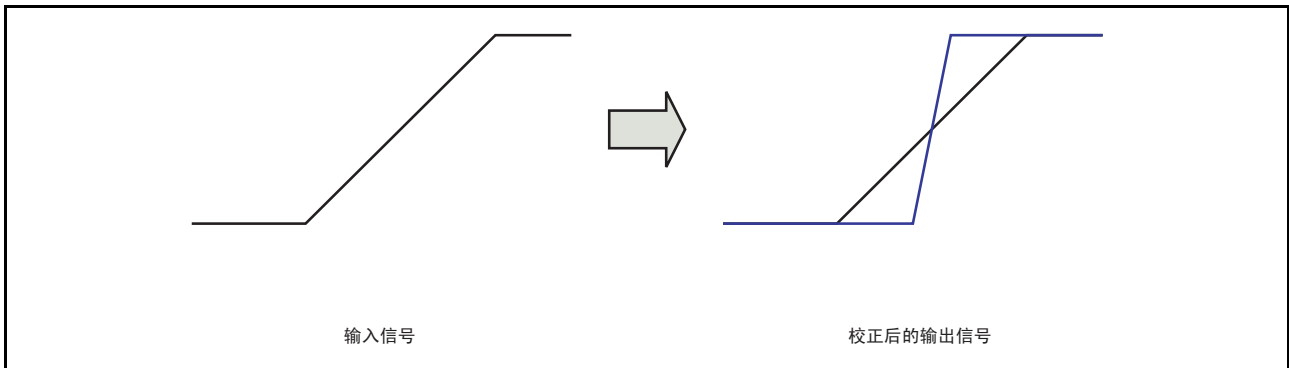


图 34.4 LTI 校正图

在检测到图像边沿后，LTI 能独立控制水平方向的 2 个频带。
LTI 在检测到图像边沿后插入中值滤波器。
LTI (H4) 能选择中值滤波器的参照像素。
但是，在正常运行中参照边沿检测时的 1/2 的 TAP 数据 (第 2 个相邻像素)。

表 34.4 LTI 参照像素表

LTI 频带	边沿检测时的参照像素	LPF 的有无	中值滤波器的参照像素
水平 LTI (H2)	参照第 2 个相邻像素。	无 LPF 或者 LPF (1、2、1)	参照第 1 个相邻像素。
水平 LTI (H4)	参照第 4 个相邻像素。	LPF (1、2、1)	参照第 1 个或者第 2 个相邻像素。

LTI 能对边沿检测结果实施核化降噪。
从边沿检测结果减去寄存器设定的核化降噪，并且对减法结果的核化降噪输出进行 LTI 校正。

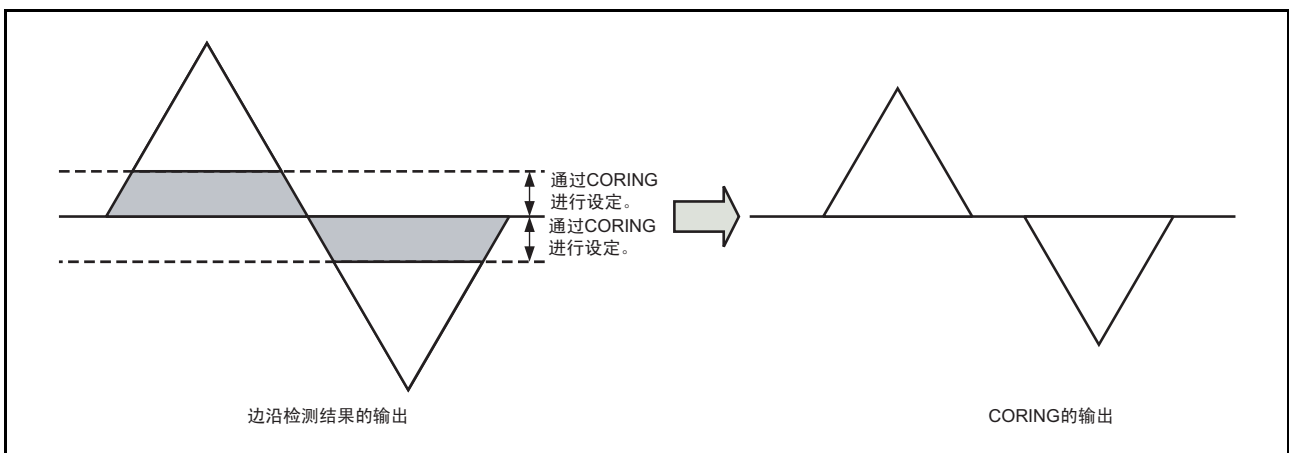


图 34.5 LTI 核化降噪图

表 34.5 LTI 控制

寄存器名	位名	初始值	说明
ADJ_ENH_LTI1	LTI_H_ON	0	LTI 的 ON/OFF 控制 0: LTI OFF 1: LTI ON
ADJ_ENH_LTI1	LTI_H2_INC_ZERO[7:0]	10	中值滤波器的 LTI 校正阈值设定 在 $ 右 TAP 值 - 中心 TAP 值 < LTI_H2_INC_ZERO$ 或者 $ 左 TAP 值 - 中心 TAP 值 < LTI_H2_INC_ZERO$ 的情况下, 禁止 LTI 校正。
ADJ_ENH_LTI1	LTI_H2_LPF_SEL	0	用于在 H2 边沿检测前消除折叠的 LPF 选择 0: 无 LPF 1: 有 LPF
ADJ_ENH_LTI1	LTI_H2_GAIN[7:0]	0	LTI 边沿振幅值的增益设定 0 (0 倍) ~ 64 (+1 倍) ~ 255 (+ 约 4 倍)
ADJ_ENH_LTI1	LTI_H2_CORE[7:0]	0	LTI 的核化降噪 (核化降噪量最大为 “255”) 对小于等于 LTI_H2_CORE 的边沿振幅值实施核化降噪 (如 果将这些位置 “128”, 核化降噪量就为 “128”)。
ADJ_ENH_LTI2	LTI_H4_INC_ZERO[7:0]	10	中值滤波器的 LTI 校正阈值设定 在 $ 右 TAP 值 - 中心 TAP 值 < LTI_H4_INC_ZERO$ 或者 $ 左 TAP 值 - 中心 TAP 值 < LTI_H4_INC_ZERO$ 的情况下, 禁止 LTI 校正。
ADJ_ENH_LTI2	LTI_H4_MEDIAN_TAP_SEL	0	中值滤波器的参照像素选择 0: 参照第 2 个相邻像素 1: 参照第 1 个相邻像素
ADJ_ENH_LTI2	LTI_H4_GAIN[7:0]	0	LTI 边沿振幅值的增益设定 0 (0 倍) ~ 64 (+1 倍) ~ 255 (+ 约 4 倍)
ADJ_ENH_LTI2	LTI_H4_CORE[7:0]	0	LTI 的核化降噪 (核化降噪量最大为 “255 “) 对小于等于 LTI_H4_CORE 的边沿振幅值实施核化降噪 (如 果将这些位置 “128”, 核化降噪量就为 “128”)。

(3) 清晰度处理

清晰度处理是指增强 Y 信号输入的水平方向的边缘。通过加入过冲和下冲, 增强源信号的边缘。另外, 消隐的边缘增强变为 OFF 状态。

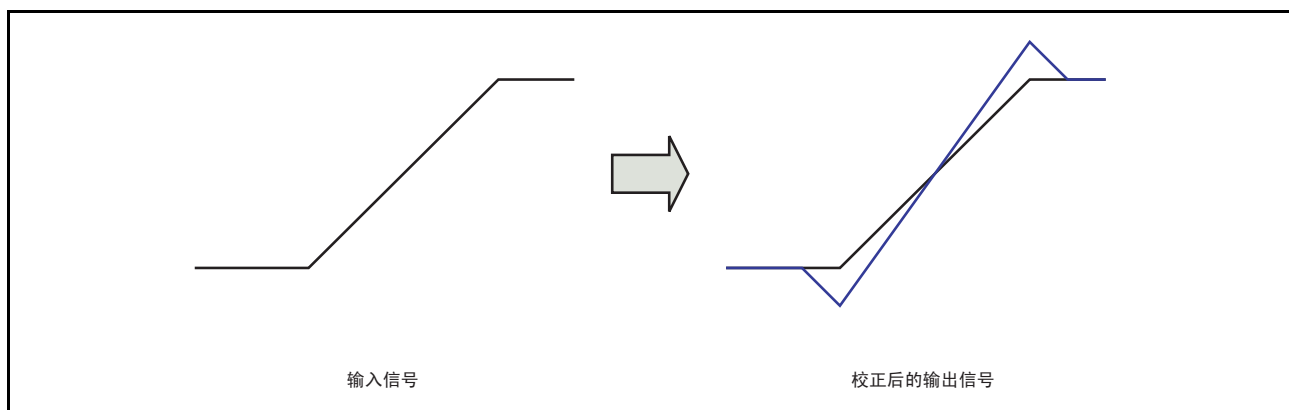


图 34.6 清晰度校正图

清晰度处理能在检测到图像边沿后独立控制水平方向的 3 个频带。

在水平方向的清晰度处理中，为了防止边沿检测时的折叠而插入 3TAP LPF，并且能通过设定寄存器来选择 LPF 的 ON/OFF。

表 34.6 清晰度的参照像素表

清晰度频带	边沿检测时的参照像素	LPF 的有无
水平清晰度 (H1)	参照第 1 个相邻像素。	无 LPF
水平清晰度 (H2)	参照第 2 个相邻像素。	无 LPF 或者 LPF (1、2、1)
水平清晰度 (H3)	参照第 3 个相邻像素。	LPF (1、2、1)

能根据 SHP_CORE 的值对增强边缘的边沿振幅进行调整。

当图像的边沿检测结果大于 SHP_CORE 时，为边缘增强。

在边缘增强中，将 (边沿振幅值 - SHP_CORE) 乘清晰度增益后，输出校正值。

当图像的边沿检测结果小于 SHP_CORE 时，清晰度处理为 OFF 状态。

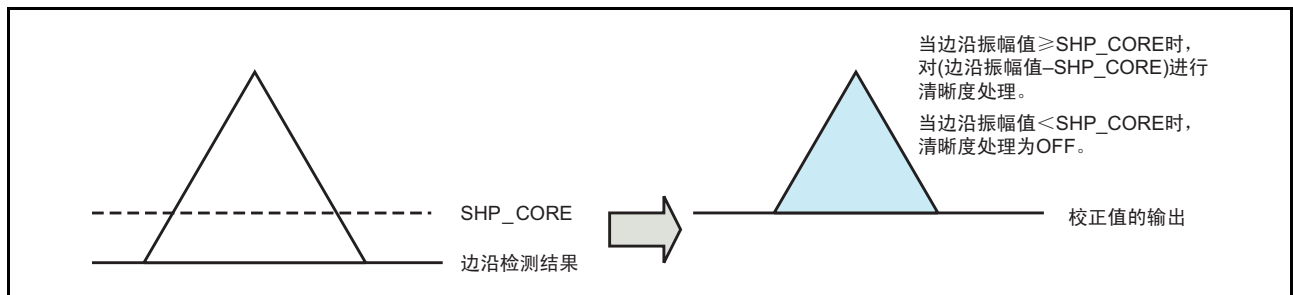


图 34.7 清晰度特性图

表 34.7 清晰度控制

寄存器名	位名	初始值	说明
ADJ_ENH_SHP1	SHP_H_ON	0	清晰度的 ON/OFF 控制 0: 水平清晰度 OFF 1: 水平清晰度 ON
ADJ_ENH_SHP3	SHP_H2_LPF_SEL	0	用于在 H2 边沿检测前消除折叠的 LPF 选择 0: 无 LPF 1: 有 LPF
ADJ_ENH_SHP2	SHP_H1_CLIP_O[7:0]	0	清晰度的校正值限幅 (过冲侧) 通过 SHP_H1_CLIP_O 位对校正值进行限幅处理。
ADJ_ENH_SHP2	SHP_H1_CLIP_U[7:0]	0	清晰度的校正值限幅 (下冲侧) 通过 SHP_H1_CLIP_U 位对校正值进行限幅处理
ADJ_ENH_SHP2	SHP_H1_GAIN_O[7:0]	0	清晰度边沿振幅值的增益设定 (过冲侧) 0 (0 倍) ~ 64 (+1 倍) ~ 255 (+ 约 4 倍) 清晰度校正值 = $SHP_H1_GAIN_O \times (\text{边沿振幅值} - SHP_H1_CORE)$
ADJ_ENH_SHP2	SHP_H1_GAIN_U[7:0]	0	清晰度边沿振幅值的增益设定 (下冲侧) 0 (0 倍) ~ 64 (+1 倍) ~ 255 (+ 约 4 倍) 清晰度校正值 = $SHP_H1_GAIN_U \times (\text{边沿振幅值} - SHP_H1_CORE)$

寄存器名	位名	初始值	说明
ADJ_ENH_SHP1	SHP_H1_CORE[6:0]	0	清晰度的有效区域指定 边沿振幅值 \geq SHP_H1_CORE: 进行清晰度处理。 边沿振幅值 $<$ SHP_H1_CORE: 不进行清晰度处理。 当边沿检测值大于等于 128 时, 固定进行清晰度处理。
ADJ_ENH_SHP4	SHP_H2_CLIP_O[7:0]	0	清晰度的校正限值幅 (过冲侧) 通过 SHP_H2_CLIP_O 位对校正值进行限幅处理。
ADJ_ENH_SHP4	SHP_H2_CLIP_U[7:0]	0	清晰度的校正限值幅 (下冲侧) 通过 SHP_H2_CLIP_U 位对校正值进行限幅处理。
ADJ_ENH_SHP4	SHP_H2_GAIN_O[7:0]	0	清晰度边沿振幅值的增益设定 (过冲侧) 0 (0 倍) \sim 64 (+1 倍) \sim 255 (+ 约 4 倍) 清晰度校正值 = $SHP_H2_GAIN_O \times (\text{边沿振幅值} - SHP_H2_CORE)$
ADJ_ENH_SHP4	SHP_H2_GAIN_U[7:0]	0	清晰度边沿振幅值的增益设定 (下冲侧) 0 (0 倍) \sim 64 (+1 倍) \sim 255 (+ 约 4 倍) 清晰度校正值 = $SHP_H2_GAIN_U \times (\text{边沿振幅值} - SHP_H2_CORE)$
ADJ_ENH_SHP3	SHP_H2_CORE[6:0]	0	清晰度的有效区域指定 边沿振幅值 \geq SHP_H2_CORE: 进行清晰度处理。 边沿振幅值 $<$ SHP_H2_CORE: 不进行清晰度处理。 当边沿检测值大于等于 128 时, 固定进行清晰度处理。
ADJ_ENH_SHP6	SHP_H3_CLIP_O[7:0]	0	清晰度的校正限值幅 (过冲侧) 通过 SHP_H3_CLIP_O 位对校正值进行限幅处理。
ADJ_ENH_SHP6	SHP_H3_CLIP_U[7:0]	0	清晰度的校正限值幅 (下冲侧) 通过 SHP_H3_CLIP_U 位对校正值进行限幅处理。
ADJ_ENH_SHP6	SHP_H3_GAIN_O[7:0]	0	清晰度边沿振幅值的增益设定 (过冲侧) 0 (0 倍) \sim 64 (+1 倍) \sim 255 (+ 约 4 倍) 清晰度校正值 = $SHP_H3_GAIN_O \times (\text{边沿振幅值} - SHP_H3_CORE)$
ADJ_ENH_SHP6	SHP_H3_GAIN_U[7:0]	0	清晰度边沿振幅值的增益设定 (下冲侧) 0 (0 倍) \sim 64 (+1 倍) \sim 255 (+ 约 4 倍) 清晰度校正值 = $SHP_H3_GAIN_U \times (\text{边沿振幅值} - SHP_H3_CORE)$
ADJ_ENH_SHP5	SHP_H3_CORE[6:0]	0	清晰度的有效区域指定 边沿振幅值 \geq SHP_H3_CORE: 进行清晰度处理。 边沿振幅值 $<$ SHP_H3_CORE: 不进行清晰度处理。 当边沿检测值大于等于 128 时, 固定进行清晰度处理。

34.1.5 色彩矩阵

色彩矩阵能调整各输入信号的偏移和 9 轴增益，因此能进行 YCbCr 到 GBR 的转换。

(1) GBR→GBR 的转换

$$YGIN_A=YGIN+ADJ_MTX_YG-128$$

$$CBBIN_A=CBBIN+ADJ_MTX_B-128$$

$$CRRIN_A=CRRIN+ADJ_MTX_R-128$$

$$YGOUT=(ADJ_MTX_GG \times YGIN_A+ADJ_MTX_GB \times CBBIN_A+ADJ_MTX_GR \times CRRIN_A) \div 256$$

$$CBBOUT=(ADJ_MTX_BG \times YGIN_A+ADJ_MTX_BB \times CBBIN_A+ADJ_MTX_BR \times CRRIN_A) \div 256$$

$$CRROUT=(ADJ_MTX_RG \times YGIN_A+ADJ_MTX_RB \times CBBIN_A+ADJ_MTX_RR \times CRRIN_A) \div 256$$

(2) YCbCr→GBR 的转换

$$YGIN_A=YGIN+ADJ_MTX_YG-128$$

$$CBBIN_A=CBBIN-128$$

$$CRRIN_A=CRRIN-128$$

$$YGOUT=(ADJ_MTX_GG \times YGIN_A+ADJ_MTX_GB \times CBBIN_A+ADJ_MTX_GR \times CRRIN_A) \div 256$$

$$CBBOUT=(ADJ_MTX_BG \times YGIN_A+ADJ_MTX_BB \times CBBIN_A+ADJ_MTX_BR \times CRRIN_A) \div 256$$

$$CRROUT=(ADJ_MTX_RG \times YGIN_A+ADJ_MTX_RB \times CBBIN_A+ADJ_MTX_RR \times CRRIN_A) \div 256$$

表 34.8 SMPTE 293M 的矩阵系数 (标准值)

	YGIN		CBBIN		CRRIN	
	系数	位设定值	系数	位设定值	系数	位设定值
YGOUT	1.000	ADJ_MTX_GG =256	-0.344	ADJ_MTX_GB =1960	-0.714	ADJ_MTX_GR =1865
CBBOUT	1.000	ADJ_MTX_BG =256	1.772	ADJ_MTX_BB =454	0.000	ADJ_MTX_BR =0
CRROUT	1.000	ADJ_MTX_RG =256	0.000	ADJ_MTX_RB =0	1.402	ADJ_MTX_RR =359

表 34.9 色彩矩阵控制

寄存器名	位名	初始值	说明
ADJ_MTX_MODE	ADJ_MTX_MD[1:0]	2	运行模式 0: GBR→GBR 1: 禁止设定 2: YCbCr→GBR 3: 禁止设定
ADJ_MTX_YG_ADJ0	ADJ_MTX_YG[7:0]	128	Y/G 信号的偏移 (DC) 调整 无符号 (0 (-128) ~ 128 (0) ~ 255 (+127) [LSB])
ADJ_MTX_CBB_ADJ0	ADJ_MTX_B[7:0]	128	B 信号的偏移 (DC) 调整 无符号 (0 (-128) ~ 128 (0) ~ 255 (+127) [LSB])
ADJ_MTX_CRR_ADJ0	ADJ_MTX_R[7:0]	128	R 信号的偏移 (DC) 调整 无符号 (0 (-128) ~ 128 (0) ~ 255 (+127) [LSB])
ADJ_MTX_YG_ADJ0	ADJ_MTX_GG[10:0]	256	G 信号输出的 Y/G 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])
ADJ_MTX_YG_ADJ1	ADJ_MTX_GB[10:0]	1960	G 信号输出的 Cb/B 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])
ADJ_MTX_YG_ADJ1	ADJ_MTX_GR[10:0]	1865	G 信号输出的 Cr/R 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])
ADJ_MTX_CBB_ADJ9	ADJ_MTX_BG[10:0]	256	B 信号输出的 Y/G 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])
ADJ_MTX_CBB_ADJ1	ADJ_MTX_BB[10:0]	454	B 信号输出的 Cb/B 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])
ADJ_MTX_CBB_ADJ1	ADJ_MTX_BR[10:0]	0	B 信号输出的 Cr/R 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])
ADJ_MTX_CRR_ADJ0	ADJ_MTX_RG[10:0]	256	R 信号输出 Y/G 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])
ADJ_MTX_CRR_ADJ1	ADJ_MTX_RB[10:0]	0	R 信号输出的 Cb/B 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])
ADJ_MTX_CRR_ADJ1	ADJ_MTX_RR[10:0]	359	R 信号输出的 Cr/R 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])

34.2 寄存器说明

寄存器结构如表 34.10 所示。

【寄存器的符号说明】

初始值：复位后的寄存器值

— : 不定值

R/W : 可读写。能读到写入值。

R/WC0 : 可读写。如果写“0”，该位就被初始化，但是忽视写“1”。

R/WC1 : 可读写。如果写“1”，该位就被初始化，但是忽视写“0”。

R : 只能读。写入值总是“0”。

—/W : 只能写。读取值为不定值。

表 34.10 图像质量改善部的寄存器结构

名称	略称	R/W	初始值	地址	存取长度
图像质量改善部的寄存器更新控制寄存器	ADJ_UPDATE	R/WC1	H'0000 0000	H'FFFF 7680	32/16
黑色信号展宽部寄存器	ADJ_BKSTR_SET	R/W	H'0000 0000	H'FFFF 7684	32/16
增强器的时序调整寄存器 1	ADJ_ENH_TIM1	R/W	H'0000 0010	H'FFFF 7688	32/16
增强器的时序调整寄存器 2	ADJ_ENH_TIM2	R/W	H'0023 01E0	H'FFFF 768C	32/16
增强器的时序调整寄存器 3	ADJ_ENH_TIM3	R/W	H'0091 0280	H'FFFF 7690	32/16
增强器的清晰度寄存器 1	ADJ_ENH_SHP1	R/W	H'0000 0000	H'FFFF 7694	32/16
增强器的清晰度寄存器 2	ADJ_ENH_SHP2	R/W	H'0000 0000	H'FFFF 7698	32/16
增强器的清晰度寄存器 3	ADJ_ENH_SHP3	R/W	H'0000 0000	H'FFFF 769C	32/16
增强器的清晰度寄存器 4	ADJ_ENH_SHP4	R/W	H'0000 0000	H'FFFF 76A0	32/16
增强器的清晰度寄存器 5	ADJ_ENH_SHP5	R/W	H'0000 0000	H'FFFF 76A4	32/16
增强器的清晰度寄存器 6	ADJ_ENH_SHP6	R/W	H'0000 0000	H'FFFF 76A8	32/16
增强器的 LTI 寄存器 1	ADJ_ENH_LTI1	R/W	H'000A 0000	H'FFFF 76AC	32/16
增强器的 LTI 寄存器 2	ADJ_ENH_LTI2	R/W	H'000A 0000	H'FFFF 76B0	32/16
图像质量改善部的矩阵模式寄存器	ADJ_MTX_MODE	R/W	H'0000 0002	H'FFFF 76B4	32/16
图像质量改善部的矩阵 YG 调整寄存器 0	ADJ_MTX_YG_ADJ0	R/W	H'0080 0100	H'FFFF 76B8	32/16
图像质量改善部的矩阵 YG 调整寄存器 1	ADJ_MTX_YG_ADJ1	R/W	H'07A8 0749	H'FFFF 76BC	32/16
图像质量改善部的矩阵 CBB 调整寄存器 0	ADJ_MTX_CBB_ADJ0	R/W	H'0080 0100	H'FFFF 76C0	32/16
图像质量改善部的矩阵 CBB 调整寄存器 1	ADJ_MTX_CBB_ADJ1	R/W	H'01C6 0000	H'FFFF 76C4	32/16
图像质量改善部的矩阵 CRR 调整寄存器 0	ADJ_MTX_CRR_ADJ0	R/W	H'0080 0100	H'FFFF 76C8	32/16
图像质量改善部的矩阵 CRR 调整寄存器 1	ADJ_MTX_CRR_ADJ1	R/W	H'0000 0167	H'FFFF 76CC	32/16

34.2.1 图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADJ VEN
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

位	位名	初始值	R/W	说明
31 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	ADJ_VEN	0	R/WC1	图像质量改善部的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器

34.2.2 黑色信号展宽部寄存器 (ADJ_BKSTR_SET)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	BKSTR_ON	BKSTR_ST[3:0]			BKSTR_D[3:0]				
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	BKSTR_T1[4:0]				—	—	—	BKSTR_T2[4:0]					
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 25	—	全 0	R	保留位 读写值都为“0”。
24	BKSTR_ON	0	R/W	黑色信号展宽的 ON/OFF 控制 0: 黑色信号展宽 OFF 1: 黑色信号展宽 ON
23 ~ 20	BKSTR_ST[3:0]	0	R/W	黑色信号展宽的起始点指定 设定值: 0 (低) ~ 15 (高)
19 ~ 16	BKSTR_D[3:0]	0	R/W	黑色信号展宽的深度 设定值: 0 (浅) ~ 15 (深)
15 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12 ~ 8	BKSTR_T1[4:0]	0	R/W	黑色信号展宽的时间常数 (T1) 设定值: 0 (小) ~ 31 (大)
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4 ~ 0	BKSTR_T2[4:0]	0	R/W	黑色信号展宽的时间常数 (T2) 设定值: 0 (小) ~ 31 (大)

【注】 在全部图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE) 的 ADJ_VEN 为“1”时更新此寄存器。

34.2.3 增强器的时序调整寄存器 1 (ADJ_ENH_TIM1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	ENH_MD	—	—	—	ENH_DISP_ON
初始值:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	ENH_MD	1	R/W	运行模式 0: RGB 模式 1: YCbCr 模式
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	ENH_DISP_ON	0	R/W	增强有效区域的轮廓显示 ON/OFF 设定 0: 不显示轮廓 1: 显示轮廓

【注】 在全部图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE) 的 ADJ_VEN 为“1”时更新此寄存器。

34.2.4 增强器的时序调整寄存器 2 (ADJ_ENH_TIM2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ENH_VS[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ENH_VW[10:0]										
初始值:	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	ENH_VS[10:0]	35	R/W	增强有效区域的垂直有效图像区的起始位置设定 【注】 必须至少设定 2 行。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	ENH_VW[10:0]	480	R/W	增强有效区域的垂直有效图像区的宽度设定

【注】 在全部图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE) 的 ADJ_VEN 为“1”时更新此寄存器。

34.2.5 增强器的时序调整寄存器 3 (ADJ_ENH_TIM3)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ENH_HS[10:0]										
初始值:	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ENH_HW[10:0]										
初始值:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	ENH_HS[10:0]	145	R/W	增强有效区域的水平有效图像区的起始位置设定 【注】 必须至少设定 4 行。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	ENH_HW[10:0]	640	R/W	增强有效区域的水平有效图像区的宽度设定

【注】 在全部图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE) 的 ADJ_VEN 为“1”时更新此寄存器。

34.2.6 增强器的清晰度寄存器 1 (ADJ_ENH_SHP1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SHP_H_ON
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	SHP_H1_CORE[6:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 17	—	全 0	R	保留位 读写值都为“0”。
16	SHP_H_ON	0	R/W	清晰度的 ON/OFF 控制 0: 水平清晰度 OFF 1: 水平清晰度 ON
15 ~ 7	—	全 0	R	保留位 读写值都为“0”。
6 ~ 0	SHP_H1_CORE [6:0]	0	R/W	清晰度的有效区域指定 边沿振幅值 ≥ SHP_H1_CORE: 进行清晰度处理。 边沿振幅值 < SHP_H1_CORE: 不进行清晰度处理。 当边沿检测值大于等于 128 时, 固定进行清晰度处理。

【注】 在全部图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE) 的 ADJ_VEN 为“1”时更新此寄存器。

34.2.7 增强器的清晰度寄存器 2 (ADJ_ENH_SHP2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SHP_H1_CLIP_O[7:0]								SHP_H1_CLIP_U[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SHP_H1_GAIN_O[7:0]								SHP_H1_GAIN_U[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	SHP_H1_CLIP_O [7:0]	0	R/W	清晰度的校正限幅 (过冲侧) 通过 SHP_H1_CLIP_O 位对校正值进行限幅处理。
23 ~ 16	SHP_H1_CLIP_U [7:0]	0	R/W	清晰度的校正限幅 (下冲侧) 通过 SHP_H1_CLIP_U 位对校正值进行限幅处理。
15 ~ 8	SHP_H1_GAIN_O [7:0]	0	R/W	清晰度边沿振幅值的增益设定 (过冲侧) 0 (0 倍) ~ 64 (+1 倍) ~ 255 (+ 约 4 倍) 清晰度校正值 = SHP_H1_GAIN_O × (边沿振幅值 - SHP_H1_CORE)
7 ~ 0	SHP_H1_GAIN_U [7:0]	0	R/W	清晰度边沿振幅值的增益设定 (下冲侧) 0 (0 倍) ~ 64 (+1 倍) ~ 255 (+ 约 4 倍) 清晰度校正值 = SHP_H1_GAIN_U × (边沿振幅值 - SHP_H1_CORE)

【注】 在全部图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE) 的 ADJ_VEN 为 “1” 时更新此寄存器。

34.2.8 增强器的清晰度寄存器 3 (ADJ_ENH_SHP3)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SHP_H2_LPF_SEL
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	SHP_H2_CORE[6:0]						
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 17	—	全 0	R	保留位 读写值都为“0”。
16	SHP_H2_LPF_SEL	0	R/W	用于在 H2 边沿检测前消除折叠的 LPF 选择 0: 无 LPF 1: 有 LPF
15 ~ 7	—	全 0	R	保留位 读写值都为“0”。
6 ~ 0	SHP_H2_CORE [6:0]	0	R/W	清晰度的有效区域指定 边沿振幅值 \geq SHP_H2_CORE: 进行清晰度处理。 边沿振幅值 $<$ SHP_H2_CORE: 不进行清晰度处理。 当边沿检测值大于等于 128 时, 固定进行清晰度处理。

【注】 在全部图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE) 的 ADJ_VEN 为“1”时更新此寄存器。

34.2.9 增强器的清晰度寄存器 4 (ADJ_ENH_SHP4)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SHP_H2_CLIP_O[7:0]								SHP_H2_CLIP_U[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SHP_H2_GAIN_O[7:0]								SHP_H2_GAIN_U[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	SHP_H2_CLIP_O [7:0]	0	R/W	清晰度的校正限幅（过冲侧） 通过 SHP_H2_CLIP_O 位对校正限幅值进行限幅处理。
23 ~ 16	SHP_H2_CLIP_U [7:0]	0	R/W	清晰度的校正限幅（下冲侧） 通过 SHP_H2_CLIP_U 位对校正限幅值进行限幅处理。
15 ~ 8	SHP_H2_GAIN_O [7:0]	0	R/W	清晰度边沿振幅值的增益设定（过冲侧） 0（0倍）~ 64（+1倍）~ 255（+约4倍） 清晰度校正 = $SHP_H2_GAIN_O \times (\text{边沿振幅值} - SHP_H2_CORE)$
7 ~ 0	SHP_H2_GAIN_U [7:0]	0	R/W	清晰度边沿振幅值的增益设定（下冲侧） 0（0倍）~ 64（+1倍）~ 255（+约4倍） 清晰度校正 = $SHP_H2_GAIN_U \times (\text{边沿振幅值} - SHP_H2_CORE)$

【注】 在全部图像质量改善部的寄存器更新控制寄存器（ADJ_UPDATE）的 ADJ_VEN 为“1”时更新此寄存器。

34.2.10 增强器的清晰度寄存器 5 (ADJ_ENH_SHP5)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	SHP_H3_CORE[6:0]						
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 7	—	全 0	R	保留位 读写值都为“0”。
6 ~ 0	SHP_H3_CORE [6:0]	0	R/W	清晰度的有效区域指定 边沿振幅值 \geq SHP_H3_CORE: 进行清晰度处理。 边沿振幅值 $<$ SHP_H3_CORE: 不进行清晰度处理。 当边沿检测值大于等于 128 时, 固定进行清晰度处理。

【注】 在全部图像质量改善部的寄存器更新控制寄存器（ADJ_UPDATE）的 ADJ_VEN 为“1”时更新此寄存器。

34.2.11 增强器的清晰度寄存器 6 (ADJ_ENH_SHP6)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SHP_H3_CLIP_O[7:0]								SHP_H3_CLIP_U[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SHP_H3_GAIN_O[7:0]								SHP_H3_GAIN_U[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	SHP_H3_CLIP_O [7:0]	0	R/W	清晰度的校正限幅 (过冲侧) 通过 SHP_H3_CLIP_O 位对校正限幅值进行限幅处理。
23 ~ 16	SHP_H3_CLIP_U [7:0]	0	R/W	清晰度的校正限幅 (下冲侧) 通过 SHP_H3_CLIP_U 位对校正限幅值进行限幅处理。
15 ~ 8	SHP_H3_GAIN_O [7:0]	0	R/W	清晰度边沿振幅值的增益设定 (过冲侧) 0 (0 倍) ~ 64 (+1 倍) ~ 255 (+ 约 4 倍) 清晰度校正 = $SHP_H3_GAIN_O \times (\text{边沿振幅值} - SHP_H3_CORE)$
7 ~ 0	SHP_H3_GAIN_U [7:0]	0	R/W	清晰度边沿振幅值的增益设定 (下冲侧) 0 (0 倍) ~ 64 (+1 倍) ~ 255 (+ 约 4 倍) 清晰度校正 = $SHP_H3_GAIN_U \times (\text{边沿振幅值} - SHP_H3_CORE)$

【注】 在全部图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE) 的 ADJ_VEN 为 “1” 时更新此寄存器。

34.2.12 增强器的 LTI 寄存器 1 (ADJ_ENH_LTI1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LTI_H_ON	—	—	—	—	—	—	LTI_H2_LPF_SEL	LTI_H2_INC_ZERO[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
R/W:	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LTI_H2_GAIN[7:0]								LTI_H2_CORE[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31	LTI_H_ON	0	R/W	LTI 的 ON/OFF 控制 0: LTIOFF 1: LTION
30 ~ 25	—	全 0	R	保留位 读写值都为“0”。
24	LTI_H2_LPF_SEL	0	R/W	用于在 H2 边沿检测前消除折叠的 LPF 选择 0: 无 LPF 1: 有 LPF
23 ~ 16	LTI_H2_INC_ZERO[7:0]	10	R/W	中值滤波器的 LTI 校正阈值设定 在 右 TAP 值 - 中心 TAP 值 < LTI_H2_INC_ZERO 或者 左 TAP 值 - 中心 TAP 值 < LTI_H2_INC_ZERO 的情况下, 禁止 LTI 校正。
15 ~ 8	LTI_H2_GAIN[7:0]	0	R/W	LTI 边沿振幅值的增益设定 0 (0 倍) ~ 64 (+1 倍) ~ 255 (+约 4 倍)
7 ~ 0	LTI_H2_CORE[7:0]	0	R/W	LTI 的核化降噪 (核化降噪量最大为“255”) 对小于等于 LTI_H2_CORE 的边沿振幅值进行核化降噪 (如果将这些位置“128”, 核化降噪量就为“128”)

【注】 在全部图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE) 的 ADJ_VEN 为“1”时更新此寄存器。

34.2.13 增强器的 LTI 寄存器 2（ADJ_ENH_LTI2）

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	LTI_H4_MEDIAN_TAP_SEL	LTI_H4_INC_ZERO[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LTI_H4_GAIN[7:0]								LTI_H4_CORE[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 25	—	全 0	R	保留位 读写值都为“0”。
24	LTI_H4_MEDIAN_TAP_SEL	0	R/W	中值滤波器的参照像素选择 0: 参照第 2 个相邻像素 1: 参照第 1 个相邻像素
23 ~ 16	LTI_H4_INC_ZERO[7:0]	10	R/W	中值滤波器的 LTI 校正阈值设定 在 右 TAP 值 - 中心 TAP 值 < LTI_H4_INC_ZERO 或者 左 TAP 值 - 中心 TAP 值 < LTI_H4_INC_ZERO 的情况下, 禁止 LTI 校正。
15 ~ 8	LTI_H4_GAIN[7:0]	0	R/W	LTI 边沿振幅值的增益设定 0 (0 倍) ~ 64 (+1 倍) ~ 255 (+ 约 4 倍)
7 ~ 0	LTI_H4_CORE[7:0]	0	R/W	LTI 的核化降噪 (核化降噪量最大为“255”) 对小于等于 LTI_H4_CORE 的边沿振幅值进行核化降噪 (如果将这些位置“128”, 核化降噪量就为“128”)

【注】 在全部图像质量改善部的寄存器更新控制寄存器（ADJ_UPDATE）的 ADJ_VEN 为“1”时更新此寄存器。

34.2.14 图像质量改善部的矩阵模式寄存器 (ADJ_MTX_MODE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADJ_MTX_MD[1:0]
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 2	—	全 0	R	保留位 读写值都为“0”。
1、0	ADJ_MTX_MD [1:0]	2	R/W	运行模式 0: GBR=>GBR 1: 禁止设定 2: YCbCr=>GBR 3: 禁止设定

【注】 在全部图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE) 的 ADJ_VEN 为“1”时更新此寄存器。

34.2.15 图像质量改善部的矩阵 YG 调整寄存器 0 (ADJ_MTX_YG_ADJ0)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	ADJ_MTX_YG[7:0]							
初始值:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ_MTX_GG[10:0]										
初始值:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23 ~ 16	ADJ_MTX_YG [7:0]	128	R/W	Y/G 信号的偏移 (DC) 调整 无符号 (0 (-128) ~ 128 (0) ~ 255 (+127) [LSB])
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	ADJ_MTX_GG [10:0]	256	R/W	G 信号输出的 Y/G 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])

【注】 在全部图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE) 的 ADJ_VEN 为“1”时更新此寄存器。

34.2.16 图像质量改善部的矩阵 YG 调整寄存器 1 (ADJ_MTX_YG_ADJ1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ADJ_MTX_GB[10:0]										
初始值:	0	0	0	0	0	1	1	1	1	0	1	0	1	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ_MTX_GR[10:0]										
初始值:	0	0	0	0	0	1	1	1	0	1	0	0	1	0	0	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	ADJ_MTX_GB [10:0]	1960	R/W	G 信号输出的 Cb/B 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	ADJ_MTX_GR [10:0]	1865	R/W	G 信号输出的 Cr/R 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])

【注】 在全部图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE) 的 ADJ_VEN 为“1”时更新此寄存器。

34.2.17 图像质量改善部的矩阵 CBB 调整寄存器 0 (ADJ_MTX_CBB_ADJ0)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	ADJ_MTX_B[7:0]							
初始值:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ_MTX_BG[10:0]										
初始值:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23 ~ 16	ADJ_MTX_B [7:0]	128	R/W	B 信号的偏移 (DC) 调整 无符号 (0 (-128) ~ 128 (0) ~ 255 (+127) [LSB])
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	ADJ_MTX_BG [10:0]	256	R/W	B 信号输出的 Y/G 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])

【注】 在全部图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE) 的 ADJ_VEN 为“1”时更新此寄存器。

34.2.18 图像质量改善部的矩阵 CBB 调整寄存器 1 (ADJ_MTX_CBB_ADJ1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ADJ_MTX_BB[10:0]										
初始值:	0	0	0	0	0	0	0	1	1	1	0	0	0	1	1	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ_MTX_BR[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	ADJ_MTX_BB [10:0]	454	R/W	B 信号输出的 Cb/B 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	ADJ_MTX_BR [10:0]	0	R/W	B 信号输出的 Cr/R 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])

【注】 在全部图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE) 的 ADJ_VEN 为“1”时更新此寄存器。

34.2.19 图像质量改善部的矩阵 CRR 调整寄存器 0 (ADJ_MTX_CRR_ADJ0)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	ADJ_MTX_R[7:0]							
初始值:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ_MTX_RG[10:0]										
初始值:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23 ~ 16	ADJ_MTX_R [7:0]	128	R/W	R 信号的偏移 (DC) 调整 无符号 (0 (-128) ~ 128 (0) ~ 255 (+127) [LSB])
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	ADJ_MTX_RG [10:0]	256	R/W	R 信号输出的 Y/G 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])

【注】 在全部图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE) 的 ADJ_VEN 为“1”时更新此寄存器。

34.2.20 图像质量改善部的矩阵 CRR 调整寄存器 1 (ADJ_MTX_CRR_ADJ1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ADJ_MTX_RB[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ_MTX_RR[10:0]										
初始值:	0	0	0	0	0	0	0	1	0	1	1	0	0	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	ADJ_MTX_RB [10:0]	0	R/W	R 信号输出的 Cb/B 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	ADJ_MTX_RR [10:0]	359	R/W	R 信号输出的 Cr/R 信号增益调整 带符号 (2 的补码) (-1024 ~ +1023[LSB], 256[LSB]=1.0[倍])

【注】 在全部图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE) 的 ADJ_VEN 为“1”时更新此寄存器。

34.3 使用方法

34.3.1 黑色信号展宽的使用方法

能通过设定黑色信号展宽的深度 (BKSTR_D[3:0]) 和起始点 (BKSTR_ST[3:0])，调整黑色信号展宽程度；能通过设定时间常数 (BKSTR_T1[4:0] 和 BKSTR_T2[4:0])，调整黑色信号展宽时间轴的变化量；能通过设定时间常数，抑制景物交换时的急剧变化。

表 34.11 黑色信号展宽的设定寄存器

寄存器名	位名	设定值
ADJ_BKSTR_SET	BKSTR_ON	当黑色信号展宽 ON 时，为“1”。
ADJ_BKSTR_SET	BKSTR_D[3:0]	设定黑色信号展宽的深度。值越大，展宽深度越深。
ADJ_BKSTR_SET	BKSTR_ST[3:0]	设定黑色信号展宽的起始点。值越大，展宽区域越多。
ADJ_BKSTR_SET	BKSTR_T1[4:0]	设定黑色信号展宽的正向时间常数。值越大，变化越慢。
ADJ_BKSTR_SET	BKSTR_T2[4:0]	设定黑色信号展宽的反向时间常数。值越大，变化越慢。

【注】需要在设定寄存器后将图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE) 的 ADJ_VEN 置“1”。

34.3.2 增强器的 LTI 处理

LTI 调整如图 34.8 所示。

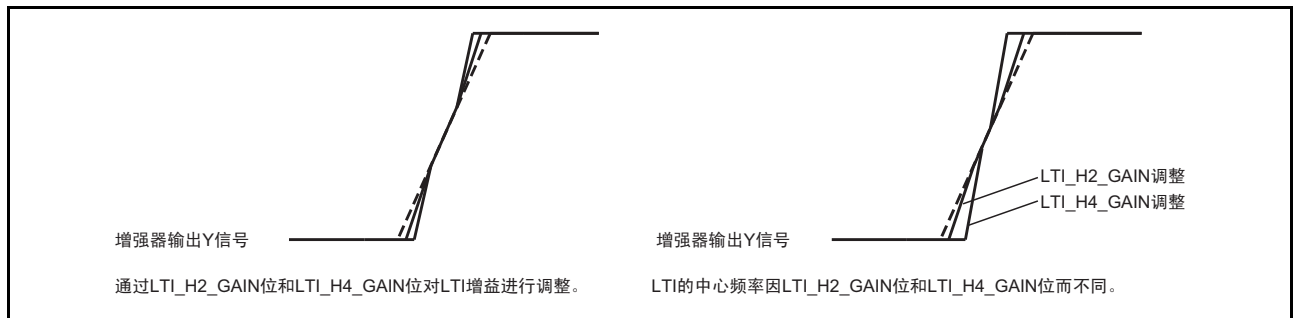


图 34.8 LTI 调整例子

34.3.3 增强器的清晰度处理

清晰度调整例子如图 34.9 所示。

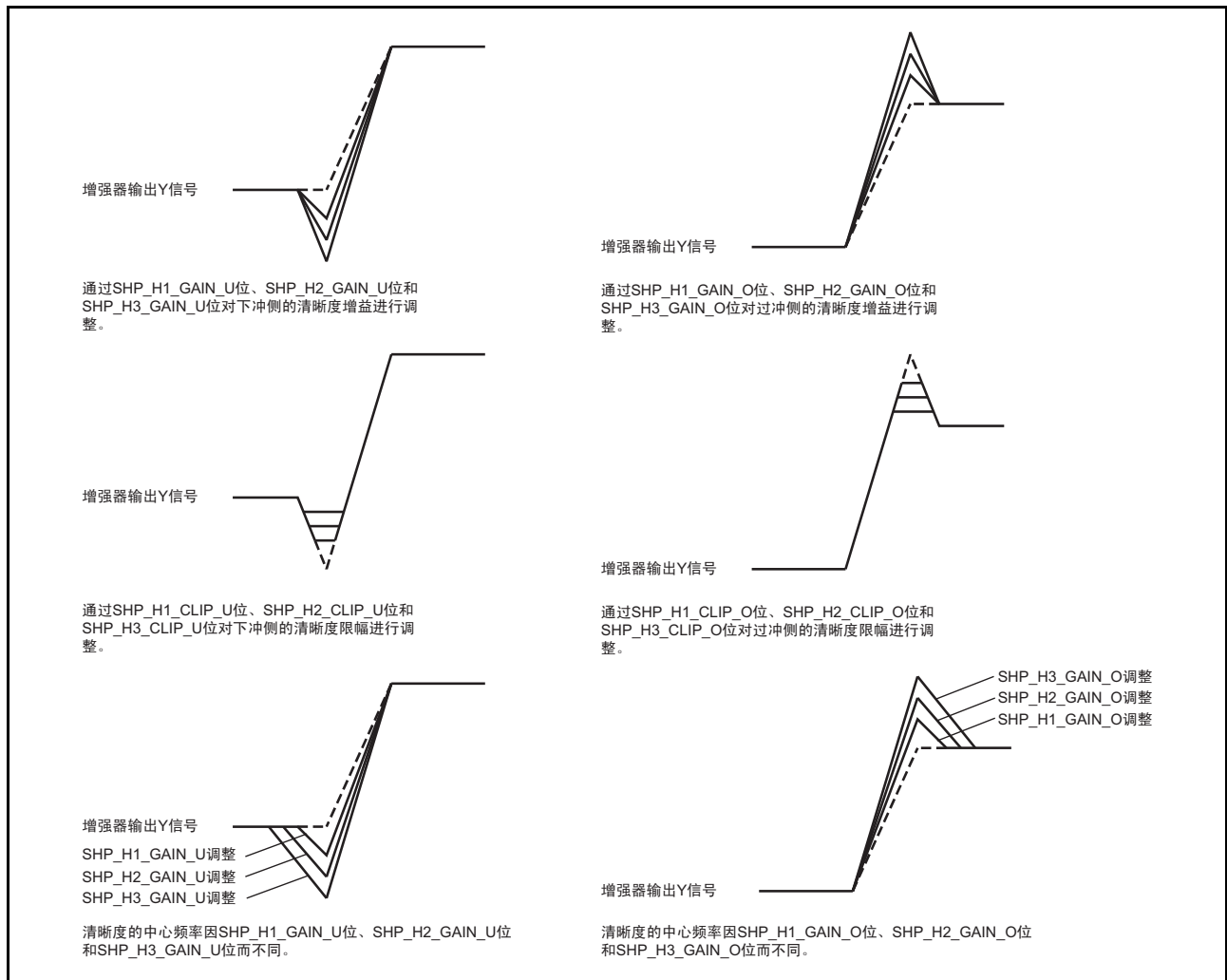


图 34.9 清晰度调整例子

34.3.4 色彩矩阵的数据转换设定方法

因为假设将 GBR 信号输入到图像质量改善部的后段电路，所以需要以 GBR 格式从色彩矩阵电路输出此信号。

GBR 转换的设定例子如表 34.12 所示。

表 34.12 矩阵转换的推荐设定值

寄存器名	位名	GBR→GBR 转换	YCbCr→GBR 转换
		推荐值	推荐值
ADJ_MTX_MODE	ADJ_MTX_MD[1:0]	0	2
ADJ_MTX_YG_ADJ0	ADJ_MTX_YG[7:0]	128	128
ADJ_MTX_CBB_ADJ0	ADJ_MTX_B[7:0]	128	128
ADJ_MTX_CRR_ADJ0	ADJ_MTX_R[7:0]	128	128
ADJ_MTX_YG_ADJ0	ADJ_MTX_GG[10:0]	256	256
ADJ_MTX_YG_ADJ1	ADJ_MTX_GB[10:0]	0	1960
ADJ_MTX_YG_ADJ1	ADJ_MTX_GR[10:0]	0	1865
ADJ_MTX_CBB_ADJ0	ADJ_MTX_BG[10:0]	0	256
ADJ_MTX_CBB_ADJ1	ADJ_MTX_BB[10:0]	256	454
ADJ_MTX_CBB_ADJ1	ADJ_MTX_BR[10:0]	0	0
ADJ_MTX_CRR_ADJ0	ADJ_MTX_RG[10:0]	0	256
ADJ_MTX_CRR_ADJ1	ADJ_MTX_RB[10:0]	0	0
ADJ_MTX_CRR_ADJ1	ADJ_MTX_RR[10:0]	256	359

【注】 需要在设定寄存器后将图像质量改善部的寄存器更新控制寄存器 (ADJ_UPDATE) 的 ADJ_VEN 置“1”。

35. 视频显示控制器 4 (5) 图像合成部

35.1 图像合成功能

35.1.1 功能概要

图像合成部读帧缓冲器的图形数据进行显示。

合成可选择 3 面图形或者图像 +2 面图形。

图形数据对应 RGB565、RGB888、αRGB1555、αRGB4444、αRGB8888、CLUT8、CLUT4、CLUT1 和 YCbCr422 (图形 (1))。

图像数据对应 RGB565、RGB888 和 YCbCr422。

各图形面能显示背景色、下层图形、当前图形以及下层图形和当前图形的混合处理 (图形 (2)、图形 (3))。

图像合成部的功能框图如下图所示。

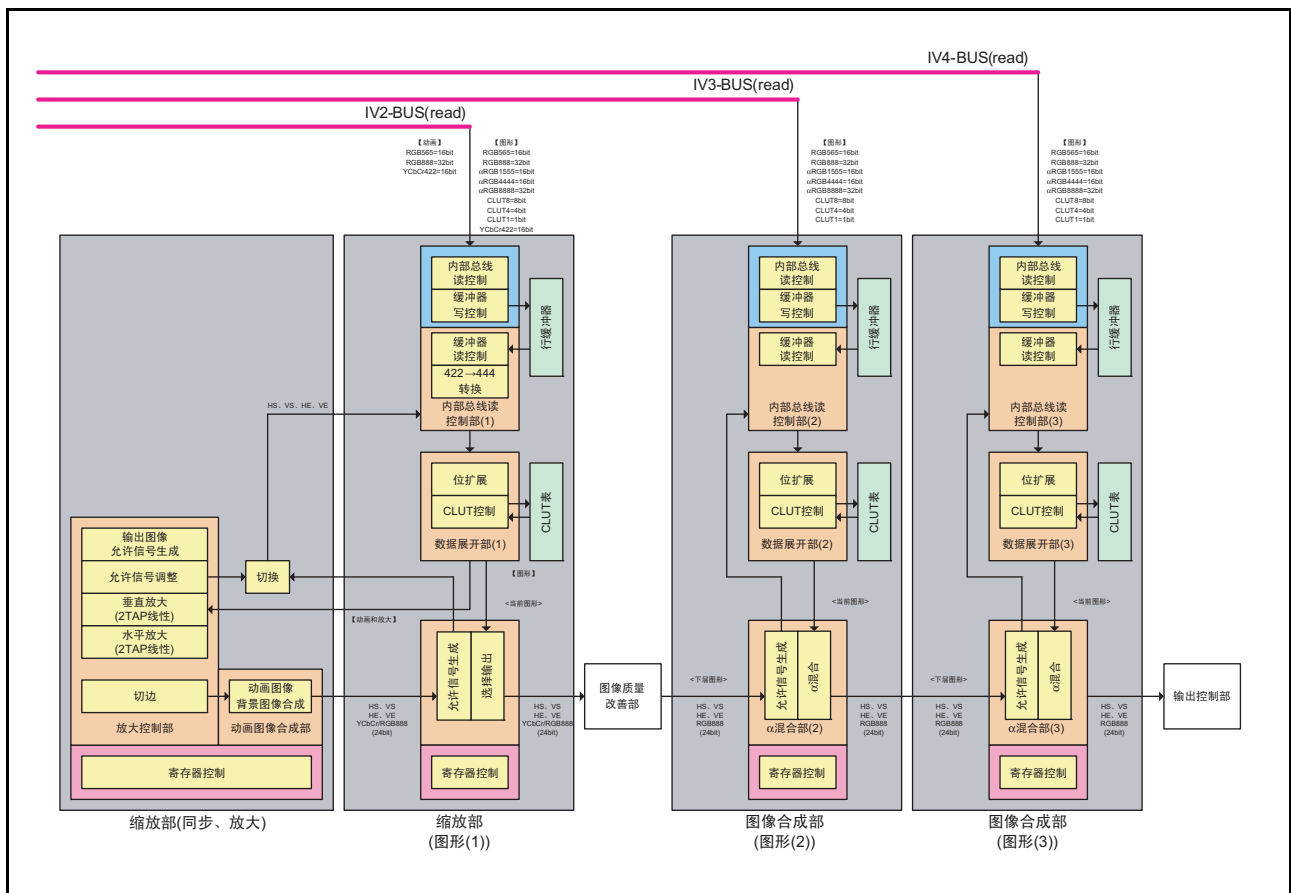


图 35.1 图像合成部的功能框图

35.1.2 图形数据读控制

图形数据读控制有缩放部的图形 (1) 显示、图像合成部的图形 (2) 显示和图形 (3) 显示共 3 个系统。

各部的寄存器和位名为 GR1_xxxx、GR2_xxxx 和 GR3_xxxx，但是在此规格书中，将寄存器和位名记为“GR_xxxx”。

(1) 寄存器的更新控制

图形显示控制寄存器和帧缓冲器读控制寄存器全部通过垂直同步信号管理更新时序。

在将更新控制寄存器置“1”后，在垂直同步信号的上升沿反映各种寄存器。反映后，更新控制寄存器自动清“0”。

表 35.1 寄存器的更新控制

寄存器名	位名	初始值	说明
GR_UPDATE	GR_P_VEN	0	图形显示的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器
GR_UPDATE	GR_IBUS_VEN	0	读帧缓冲器的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器

(2) 帧缓冲器的突发传送模式

保存图像和图形数据的帧缓冲器的存取可选择 32 字节传送模式或者 128 字节传送模式。

表 35.2 帧缓冲器的突发传送模式

寄存器名	位名	初始值	说明
GR_FLM1	GR_BST_MD	0	帧缓冲器的突发传送模式 0: 32 字节 1: 128 字节

(3) 帧缓冲器控制模式

帧缓冲器的读支持多个面。

对于图形，将 GR_FLM_SEL[1:0] 位置“1”并且通过 GR_FLM_NUM[9:0] 位设定显示面的帧号；对于图像，根据写时的处理，通过 GR_FLM_SEL[1:0] 位选择模式。图像时的使用帧数为写处理部设定的值。

表 35.3 帧缓冲器控制模式

寄存器名	位名	初始值	说明
GR_FLM1	GR_FLM_SEL[1:0]	0	帧缓冲器地址设定信号的选择 0: 协同缩小处理或者选择帧 0*1 1: 选择寄存器 GR_FLM_NUM 2: 协同失真校正处理或者选择帧 0*2 3: 禁止设定
GR_FLM3	GR_FLM_NUM[9:0]	0	帧缓冲器的帧号 在 GR_FLM_SEL 位为“1”时手动设定帧号。

【注】 *1 图形 (1) 协同缩小处理，图形 (2) 和图形 (3) 选择帧 0。

*2 图形 (1) 协同失真校正处理，图形 (2) 和图形 (3) 选择帧 0。

(4) 帧缓冲器的读控制

控制允许或者禁止读帧缓冲器。

表 35.4 帧缓冲器的读控制

寄存器名	位名	初始值	说明
GR_FLM_RD	GR_R_ENB	0	帧缓冲器的读允许 0: 禁止读 1: 允许读

(5) 失真校正帧缓冲器的控制

失真校正帧缓冲器使用帧 0 和帧 1 共 2 面，并且设定要从失真校正引擎部读的帧号。
能通过 GR_IMR_FLM_INV 位替换要读的帧缓冲器号（帧 0 和帧 1）。
此位只在 GR_FLM_SEL 位为“2”时有效。

表 35.5 失真校正帧缓冲器的控制

寄存器名	位名	初始值	说明
GR1_FLM1	GR_IMR_FLM_INV	0	失真校正帧缓冲器号的设定* 0: 不替换读帧缓冲器号 1: 替换读帧缓冲器号

【注】 * 只有图形 (1) 有此位。

(6) 帧缓冲器大小

设定要读的帧缓冲器大小。
通过 GR_HW[9:0] 位设定水平像素个数，通过 GR_FLM_LNUM[9:0] 位设定垂直行数。

表 35.6 帧缓冲器大小

寄存器名	位名	初始值	说明
GR_FLM6	GR_HW[9:0]	0	水平有效期间的宽度设定 宽度为 (GR_HW+1) 个像素。 【注】 必须设定大于等于 2 的值。
GR_FLM5	GR_FLM_LNUM[9:0]	0	1 帧的行数设定 行数为 (GR_FLM_LNUM+1) 行。

(7) 帧缓冲器的地址计算

通过 $GR_BASE[31:0]$ 位、 $GR_LN_OFF[14:0]$ 位、 $GR_FLM_OFF[22:0]$ 位和显示面的帧号计算帧缓冲器的数据区。

以 32 字节或者 128 字节为单位，设定 $GR_LN_OFF[14:0]$ 位和 $GR_FLM_OFF[22:0]$ 位（低 5 位或者低 7 位固定为“0”）。

为了设定显示数据的起始位置，以 64 位为单位设定 $GR_BASE[31:0]$ 位（固定低 3 位）。

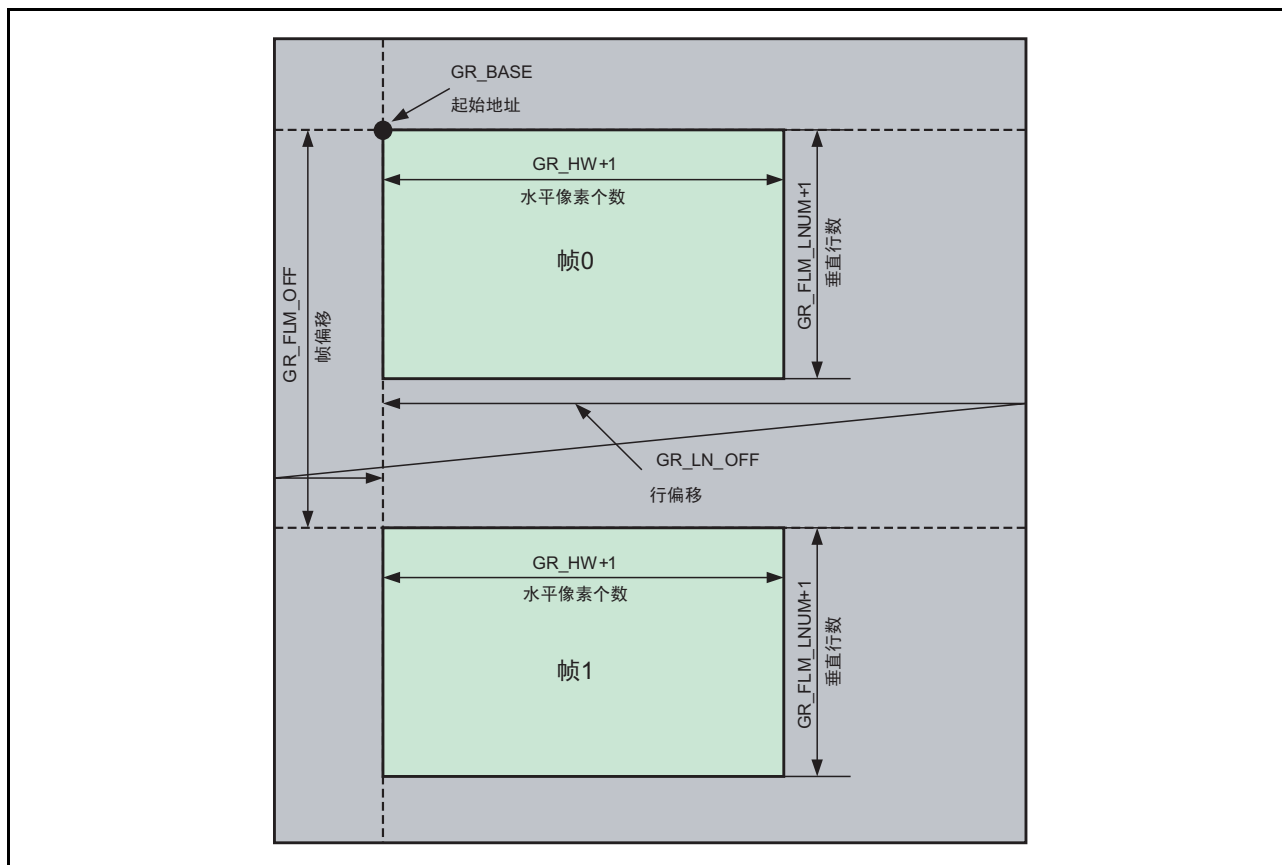


图 35.2 帧缓冲器的数据配置图

表 35.7 帧缓冲器的地址计算

寄存器名	位名	初始值	说明
GR_FLM2	GR_BASE[31:0]	0	帧缓冲器的基址 设定保存帧数据的帧缓冲器的起始地址。 在进行 32 字节突发传送时参照 GR_BASE[4:3] 位，在进行 128 字节突发传送时参照 GR_BASE[6:3] 位，并且跳读起始行数据。 必须将低 3 位固定为“000”。
GR_FLM3	GR_LN_OFF[14:0]	0	帧缓冲器的行偏移地址 设定计算行起始地址时的行偏移地址。 行 0: GR_BASE 行 1: GR_BASE+GR_LN_OFF×1 : 行 n: GR_BASE+GR_LN_OFF×n 在进行 32 字节传送时，必须将低 5 位固定为“0_0000”。 在进行 128 字节传送时，必须将低 7 位固定为“000_0000”。
GR_FLM4	GR_FLM_OFF[22:0]	0	帧缓冲器的帧偏移地址 在使用多面帧缓冲器时，设定计算各帧起始地址时的帧偏移地址。 缓冲器 0: GR_BASE 缓冲器 1: GR_BASE+GR_FLM_OFF×1 : 缓冲器 n: GR_BASE+GR_FLM_OFF×n 在进行 32 字节传送时，必须将低 5 位固定为“0_0000”。 在进行 128 字节传送时，必须将低 7 位固定为“000_0000”。

(8) 小于等于 1 面的帧缓冲器大小设定

能以行为单位设定帧缓冲器大小。

如果 GR_FLM_LOOP[9:0] 位设定的行数小于 GR_FLM_LNUM[9:0] 位，一旦读完 (GR_FLM_LOOP[9:0]+1) 设定的行数，就从帧缓冲器起始位置重新读数据。

表 35.8 小于等于 1 面的帧缓冲器大小指定

寄存器名	位名	初始值	说明
GR_FLM5	GR_FLM_LOOP[9:0]	1023	链状读地址时的行数 行数为 (GR_FLM_LOOP+1) 行。

(9) 帧缓冲器的行偏移控制

设定帧缓冲器的行偏移地址方向。

表 35.9 帧缓冲器的行偏移地址方向控制

寄存器名	位名	初始值	说明
GR_FLM1	GR_LN_OFF_DIR	0	帧缓冲器的行偏移地址方向设定 0: 递增行偏移地址 1: 递减行偏移地址

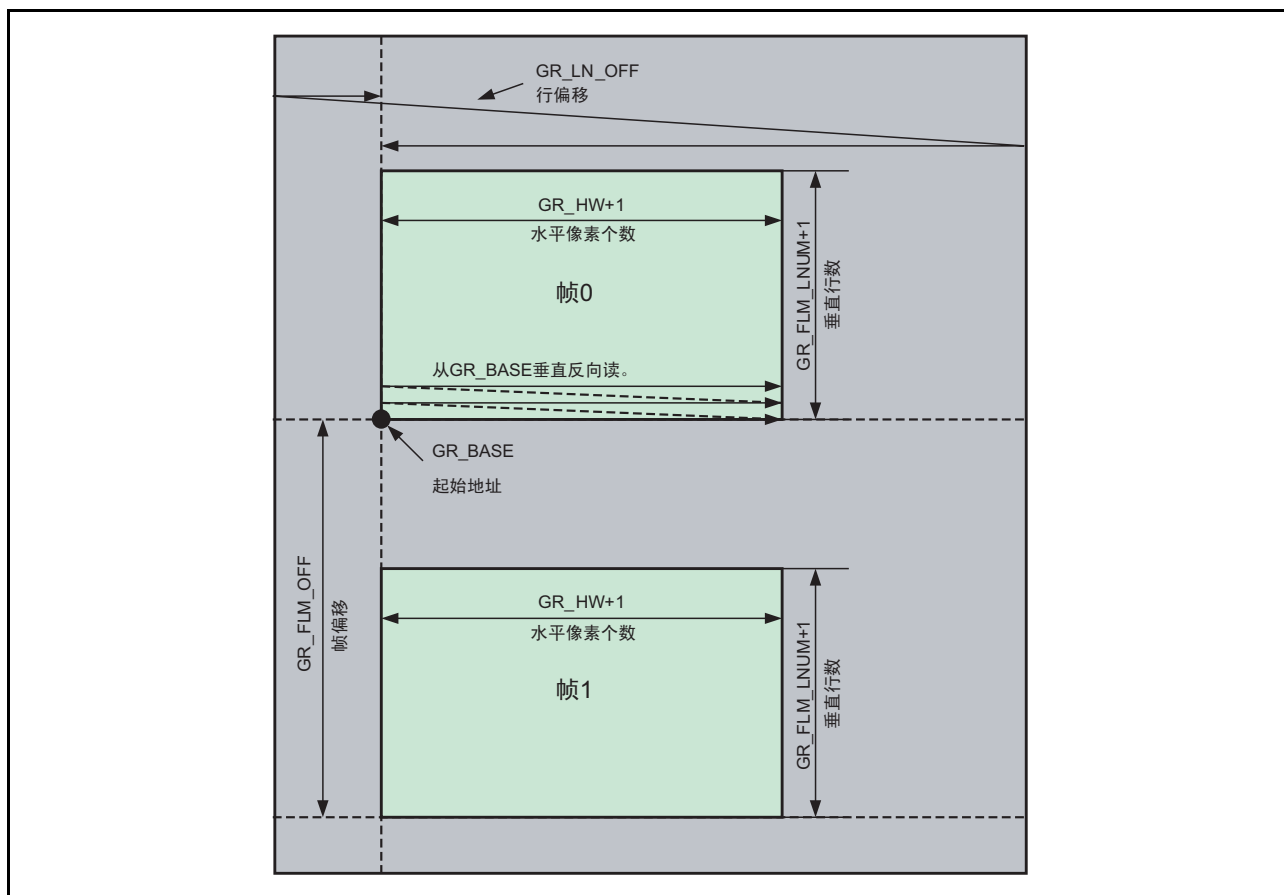


图 35.3 行偏移 递减控制时的数据配置图

(10) 帧缓冲器读信号的格式选择

图形 (1)、(2)、(3) 对应 RGB565、RGB888、 α RGB1555、 α RGB4444、 α RGB8888、CLUT8、CLUT4 和 CLUT1 格式，图形 (1) 还对应 YCbCr422。

通过 GR_FORMAT[3:0] 位设定格式。

表 35.10 帧缓冲器读信号的格式选择

寄存器名	位名	初始值	说明
GR_FLM6	GR_FORMAT[3:0]	0	帧缓冲器读信号的格式设定 0: RGB565 1: RGB888 2: α RGB1555 3: α RGB4444 4: α RGB8888 5: CLUT8 6: CLUT4 7: CLUT1 8: YCbCr422 或者禁止设定 * 9 ~ 15: 禁止设定

【注】 * 图形 (1) 能设定 YCbCr422，图形 (2) 和图形 (3) 禁止设定。

(11) 字节序控制

帧缓冲器以 64 位为单位管理 1 个数据，能通过 GR_ENDIAN_ON 位控制要读的数据字节序。对于 YCbCr422 格式，在 GR_ENDIAN_ON 位为“1”时，通过 GR_YCC_SWAP[2:0] 位控制数据配置。

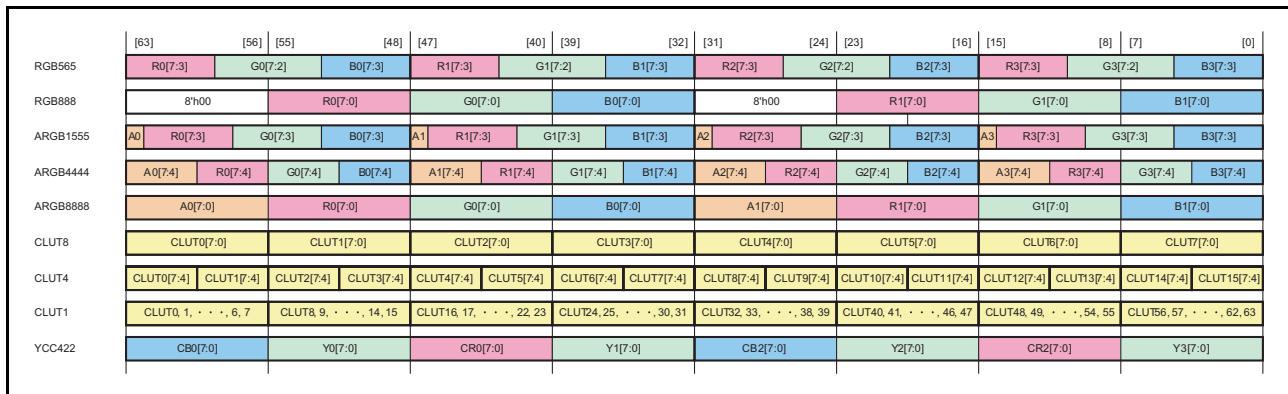


图 35.4 无字节序控制时的数据配置

	[63]	[56]	[55]	[48]	[47]	[40]	[39]	[32]	[31]	[24]	[23]	[16]	[15]	[8]	[7]	[0]								
RGB565	G0[4:2]	B0[7:3]	R0[7:3]	G0[7:5]	G1[4:2]	B1[7:3]	R1[7:3]	G1[7:5]	G2[4:2]	B2[7:3]	R2[7:3]	G2[7:5]	G3[4:2]	B3[7:3]	R3[7:3]	G3[7:5]								
RGB888	B0[7:0]		G0[7:0]		R0[7:0]		8'h00		B1[7:0]		G1[7:0]		R1[7:0]		8'h00									
ARGB1555	G0[5:3]	B0[7:3]	A0	R0[7:3]	G0[7:6]	G1[5:3]	B1[7:3]	A1	R1[7:3]	G1[7:6]	G2[5:3]	B2[7:3]	A2	R2[7:3]	G2[7:6]	G3[5:3]	B3[7:3]	A3	R3[7:3]	G3[7:6]				
ARGB4444	G0[7:4]	B0[7:4]	A0[7:4]	R0[7:4]	G1[7:4]	B1[7:4]	A1[7:4]	R1[7:4]	G2[7:4]	B2[7:4]	A2[7:4]	R2[7:4]	G3[7:4]	B3[7:4]	A3[7:4]	R3[7:4]								
ARGB8888	B0[7:0]		G0[7:0]		R0[7:0]		A0[7:0]		B1[7:0]		G1[7:0]		R1[7:0]		A1[7:0]									
CLUT8	CLUT0[7:0]			CLUT1[7:0]			CLUT2[7:0]			CLUT3[7:0]			CLUT4[7:0]			CLUT5[7:0]			CLUT6[7:0]			CLUT7[7:0]		
CLUT4	CLUT0[7:4]	CLUT1[7:4]	CLUT2[7:4]	CLUT3[7:4]	CLUT4[7:4]	CLUT5[7:4]	CLUT6[7:4]	CLUT7[7:4]	CLUT8[7:4]	CLUT9[7:4]	CLUT10[7:4]	CLUT11[7:4]	CLUT12[7:4]	CLUT13[7:4]	CLUT14[7:4]	CLUT15[7:4]								
CLUT1	CLUT0, 1, . . . , 6, 7			CLUT8, 9, . . . , 14, 15			CLUT16, 17, . . . , 22, 23			CLUT24, 25, . . . , 30, 31			CLUT32, 33, . . . , 38, 39			CLUT40, 41, . . . , 46, 47			CLUT48, 49, . . . , 54, 55			CLUT56, 57, . . . , 62, 63		

图 35.5 字节序控制时的数据配置

	[63]	[56]	[55]	[48]	[47]	[40]	[39]	[32]	[31]	[24]	[23]	[16]	[15]	[8]	[7]	[0]
YCC_SWAP=0	CB0[7:0]		Y0[7:0]		CR0[7:0]		Y1[7:0]		CB2[7:0]		Y2[7:0]		CR2[7:0]		Y3[7:0]	
YCC_SWAP=1	Y0[7:0]		CB0[7:0]		Y1[7:0]		CR0[7:0]		Y2[7:0]		CB2[7:0]		Y3[7:0]		CR2[7:0]	
YCC_SWAP=2	CR0[7:0]		Y0[7:0]		CB0[7:0]		Y1[7:0]		CR2[7:0]		Y2[7:0]		CB2[7:0]		Y3[7:0]	
YCC_SWAP=3	Y0[7:0]		CR0[7:0]		Y1[7:0]		CB0[7:0]		Y2[7:0]		CR2[7:0]		Y3[7:0]		CB2[7:0]	
YCC_SWAP=4	Y1[7:0]		CR0[7:0]		Y0[7:0]		CB0[7:0]		Y3[7:0]		CR2[7:0]		Y2[7:0]		CB2[7:0]	
YCC_SWAP=5	CR0[7:0]		Y1[7:0]		CB0[7:0]		Y0[7:0]		CR2[7:0]		Y3[7:0]		CB2[7:0]		Y2[7:0]	
YCC_SWAP=6	Y1[7:0]		CB0[7:0]		Y0[7:0]		CR0[7:0]		Y3[7:0]		CB2[7:0]		Y2[7:0]		CR2[7:0]	
YCC_SWAP=7	CB0[7:0]		Y1[7:0]		CR0[7:0]		Y0[7:0]		CB2[7:0]		Y3[7:0]		CR2[7:0]		Y2[7:0]	

图 35.6 YCbCr422 的数据交换控制时的数据配置

表 35.11 字节序控制

寄存器名	位名	初始值	说明
GR_FLM6	GR_ENDIAN_ON	0	缓冲器读数据的字节序控制 ON/OFF 设定 0: OFF 1: ON
GR1_FLM6	GR1_YCC_SWAP[2:0]	0	YCbCr422 格式时的缓冲器读数据的交换控制在 GR1_ENDIAN_ON 位为“1”时有效*。 0: CbY0/CrY1 1: Y0/Cb/Y1/Cr 2: Cr/Y0/Cb/Y1 3: Y0/Cr/Y1/Cb 4: Y1/Cr/Y0/Cb 5: Cr/Y1/Cb/Y0 6: Y1/Cb/Y0/Cr 7: Cb/Y1/Cr/Y0

【注】 * 只有图形 (1) 有这些位。

(12) 读数据的显示起始像素设定

如果在将帧缓冲器的数据进行水平方向偏移后进行显示，就通过 GR_BASE[31:0] 位和 GR_STA_POS[5:0] 位设定显示起始像素。各格式的 GR_BASE[31:0] 位和 GR_STA_POS[5:0] 位值的计算式不同。如果将显示起始的水平偏移设定为 H_OFF，就能通过下表进行计算。

表 35.12 各格式的显示起始像素位置计算式

图像 / 图形的格式	1 个像素的位数	计算式 *1
RGB888 αRGB8888 YCbCr422*2	32	GR_BASE[31:3] = int (H_OFF÷2) GR_STA_POS[5:0] = mod (H_OFF÷2)
RGB565 αRGB1555 αRGB4444	16	GR_BASE[31:3] = int (H_OFF÷4) GR_STA_POS[5:0] = mod (H_OFF÷4)
CLUT8	8	GR_BASE[31:3] = int (H_OFF÷8) GR_STA_POS[5:0] = mod (H_OFF÷8)
CLUT4	4	GR_BASE[31:3] = int (H_OFF÷16) GR_STA_POS[5:0] = mod (H_OFF÷16)
CLUT1	1	GR_BASE[31:3] = int (H_OFF÷64) GR_STA_POS[5:0] = mod (H_OFF÷64)

【注】 *1 int() 函数输出商， mod() 函数输出余数。

*2 图形 (2) 和图形 (3) 不对应 YCbCr422 格式。因为 YCbCr422 格式由 Cb/Y0/Cr/Y1 的 32 位像素构成，所以起始位置的控制是以 32 位为单位的调整。

表 35.13 读数据的显示起始像素设定

寄存器名	位名	初始值	说明
GR_FLM6	GR_STA_POS[5:0]	0	数据的跳读量 从行首跳读 GR_STA_POS 个数据。
GR_FLM2	GR_BASE[31:0]	0	帧缓冲器的基址 设定保存帧数据的帧缓冲器的起始地址。 在进行 32 字节突发传送时参照 GR_BASE[4:3] 位，在进行 128 字节突发传送时参照 GR_BASE[6:3] 位，并且跳读起始行数据。 必须将低 3 位固定为“000”。

(13) YCbCr422→YCbCr444 的转换

将图形 (1) 的 YCbCr422 格式转换为 YCbCr444 格式。
图形 (2) 和图形 (3) 没有此功能。

表 35.14 YCbCr422→YCbCr444 的转换

寄存器名	位名	初始值	说明
GR1_FLM6	GR1_CNV444_MD	0	YCbCr422→YCbCr444 转换时的内插模式设定 * 0: 保持内插 1: 平均值内插

【注】 * 因为图形 (2) 和图形 (3) 不对应 YCbCr422 格式，所以没有此位。

(14) 位扩展

将 RGB565、RGB888、αRGB1555 和 αRGB4444 格式转换为 αRGB8888 格式。

RGB565→αRGB8888 的格式转换

转换后 α[7:0]=255 (固定)

转换后 R[7:0]=R[4:0]×263÷32 (小数四舍五入) #R[4:0]×255÷31 的近似表达式

转换后 G[7:0]=G[5:0]×259÷64 (小数四舍五入) #G[5:0]×255÷63 的近似表达式

转换后 B[7:0]=B[4:0]×263÷32 (小数四舍五入) #B[4:0]×255÷31 的近似表达式

RGB888→αRGB8888 的格式转换

转换后 α[7:0]=255 (固定)

αRGB1555→αRGB8888 格式转换

转换后 α[7:0]= (当 α 输入为 “1” 时，参照 GR_A1 参照；当 α 输入为 “0” 时，参照 GR_A0)

转换后 R[7:0]=R[4:0]×263÷32 (小数四舍五入) #R[4:0]×255÷31 的近似表达式

转换后 G[7:0]=G[4:0]×263÷32 (小数四舍五入) #G[4:0]×255÷31 的近似表达式

转换后 B[7:0]=B[4:0]×263÷32 (小数四舍五入) #B[4:0]×255÷31 的近似表达式

αRGB4444→αRGB8888 格式转换

转换后 α[7:0]=α[3:0]×17

转换后 R[7:0]=R[3:0]×17

转换后 G[7:0]=G[3:0]×17

转换后 B[7:0]=B[3:0]×17

(15) 缓冲器下溢处理

如果在帧缓冲器的读处理中因总线流通量等而无法读时，就输出下溢中断信号。

35.1.3 图形区的设定

以水平同步信号的上升沿为基准，通过 GR_GRC_HS[10:0] 位、GR_GRC_HW[10:0] 位、GR_GRC_VS[10:0] 位和 GR_GRC_VW[10:0] 位设定图形的显示区。

图形的显示区如图 35.7 所示。

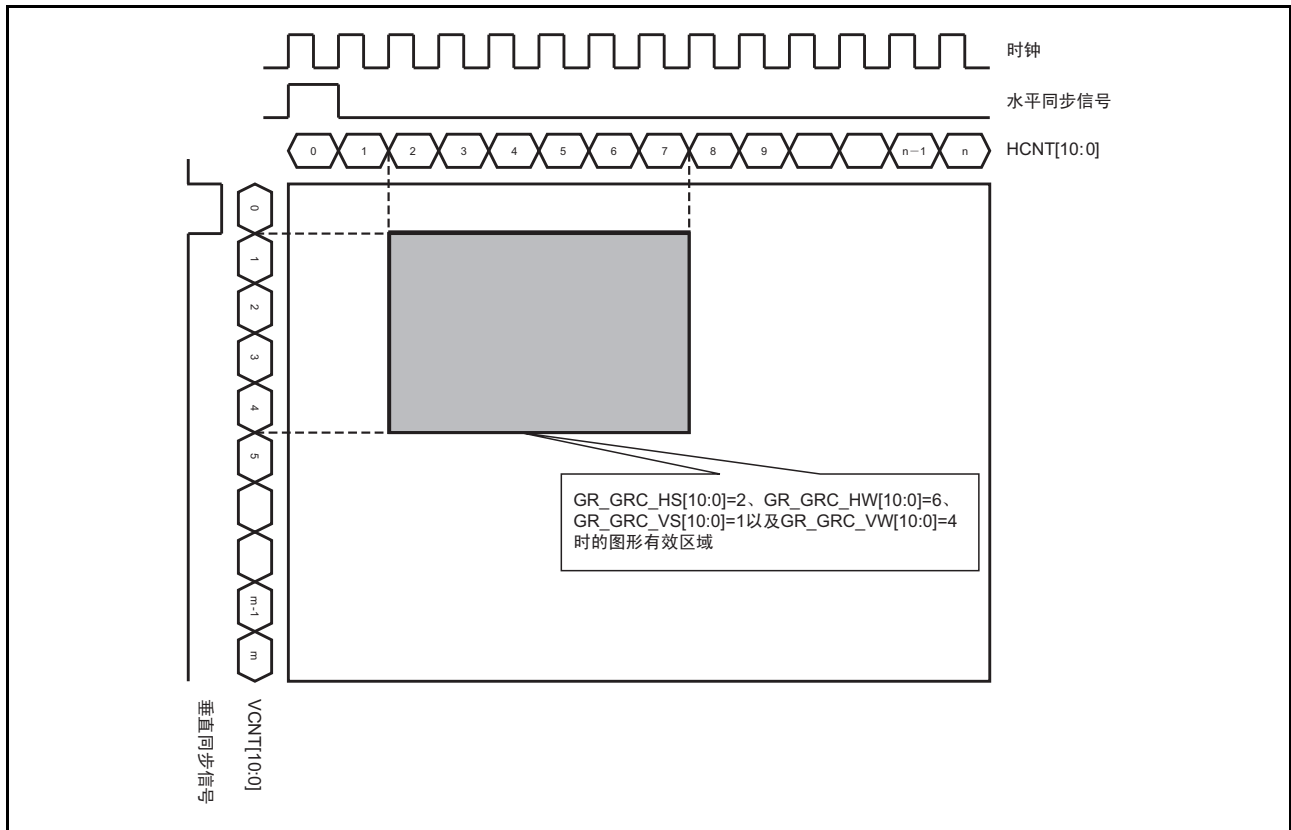


图 35.7 图形的显示区

能通过将 GR_GRC_DISP_ON 位置“1”，显示图形区的轮廓。

表 35.15 图形的显示区设定

寄存器名	位名	初始值	说明
GR_AB3	GR_GRC_HS[10:0]	0	图形图像区的水平起始位置设定 【注】 必须至少设定 16 个时钟，使 GR_GRC_HS+GR_GRC_HW 不超过 2015 个时钟。
GR_AB3	GR_GRC_HW[10:0]	0	图形图像区的水平宽度设定 【注】 在显示水平宽度为“1”或者“2”的像素时，必须将 GR_HW 位置“2”，将 GR_GRC_HW 位置“1”（1 个像素）或者“2”（2 个像素）。
GR_AB2	GR_GRC_VS[10:0]	0	图形图像区的垂直起始位置设定 【注】 必须至少设定 4 行，使 GR_GRC_VS+GR_GRC_VW 不超过 2039 行。
GR_AB2	GR_GRC_VW[10:0]	0	图形图像区的垂直宽度设定
GR_AB1	GR_GRC_DISP_ON	0	图形图像区的轮廓显示 ON/OFF 设定 0: 不显示轮廓 1: 显示轮廓

35.1.4 指定行中断的生成

在为 GR_LINE[10:0] 位设定的行时产生中断信号。

表 35.16 指定行中断

寄存器名	位名	初始值	说明
GR3_CLUT_INT	GR3_LINE[10:0]	0	行中断设定 * 在行数和寄存器 GR3_LINE 位的值相同时，输出中断信号。 即使不使用图形 (3) 的功能，此功能也起作用。

【注】 * 只有图形 (3) 有此功能，图形 (1) 和图形 (2) 没有这些位。

35.1.5 帧缓冲器读信号格式和 α 混合的对应

通过 GR_FORMAT[3:0] 位选择图形数据的帧缓冲器读信号格式。

各信号格式和 α 混合的对应表如表 35.17 所示。图形 (2) 和图形 (3) 有各种混合功能。 α 值的优先级为矩形区域 α 混合 > 色键 > 以像素为单位的 α 混合。

表 35.17 读信号格式和 α 混合的对应表

GR_FORMAT [3:0]	信号格式	矩形区域 α 混合	RGB 参照 色键	CLUT 参照 色键	以像素为单 位的 α 混合
0	RGB565	○	○ *1	×	×
1	RGB888	○	○	×	×
2	α RGB1555	○	○ *1*3	×	○ *3
3	α RGB4444	○	○ *1	×	○
4	α RGB8888	○	○	×	○
5	CLUT8	○	×	○	○
6	CLUT4	○	×	○	○
7	CLUT1	○ *4	×	○ *4	○ *4
8	YCbCr422	×	×	×	×

【注】 *1 当帧缓冲器读信号格式不是 8 位时，通过运算将 RGB 参照色键换算为 8 位（参照“35.1.2(14) 位扩展”）。

*2 因为将 α 值作为“255”处理，所以固定显示当前图形。

*3 帧缓冲器读数据的 α 值为 1 位。通过这个 1 位信号选择保存 8 位 α 值的 2 组寄存器的其中 1 组。

*4 帧缓冲器信号的 CLUT 值为 1 位。通过这个 1 位信号选择保存各 8 位 α /G/B/R 值的 2 组寄存器的其中 1 组。不参照 CLUT 表。

*5 图形 (1) 能显示 YCbCr422，但是不能进行各种混合处理和色键处理。

35.1.6 显示选择

通过 GR_DISP_SEL[1:0] 位选择背景色、下层图形、当前图形以及下层图形和当前图形的混合共 4 种显示。在选择混合显示时，能选择矩形区域 α 混合处理、RGB 参照色键处理、CLUT 参照色键处理以及以像素为单位的 α 混合处理。

显示选择一览表如表 35.18 所示。

表 35.18 显示选择一览表

GR_DISP_SEL [1:0]	GR_ARC_ON	GR_CK_ON	图形区内的处理	图形区外的处理
0	X	X	背景色	背景色
1	X	X	下层图形	下层图形
2	X	X	当前图形	背景色
3	1	X	矩形区域 α 混合	下层图形
3	0	1	RGB 参照色键 或者 CLUT 参照色键	下层图形
3	0	0	以像素为单位的 α 混合	下层图形

【注】 * 只有图形 (2) 和图形 (3) 有各种混合功能。

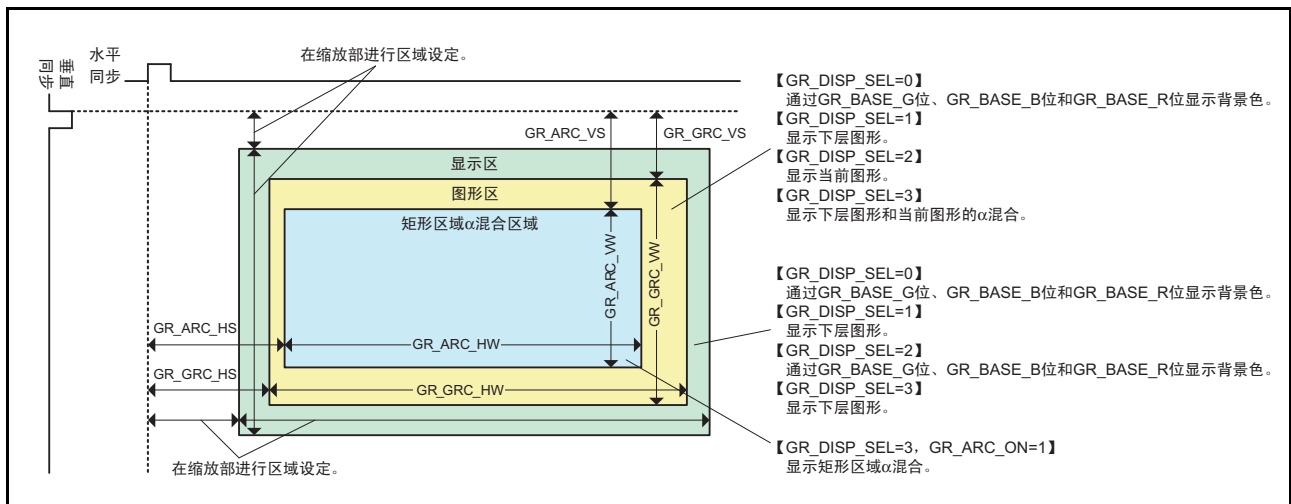


图 35.8 图形显示选择图

GR_DISP_SEL 位为“3”时的图形面如图 35.9 所示。

下层图形和当前图形的对应请参照图 35.1。

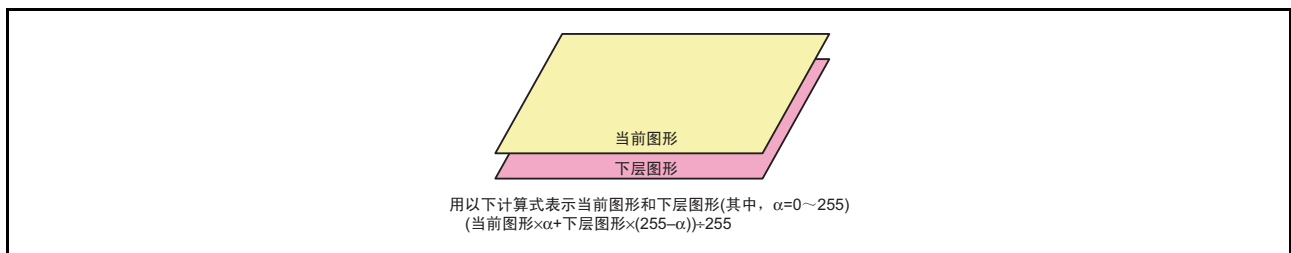


图 35.9 GR_DISP_SEL 位为“3”时的图形面

表 35.19 α 混合设定

寄存器名	位名	初始值	说明
GR_AB1	GR_DISP_SEL[1:0]	0	图形显示设定 0: 显示背景色 1: 显示下层图形 2: 显示当前图形 3: 混合显示下层图形和当前图形 *1
GR_AB1	GR_ARC_ON	0	矩形区域 α 混合处理的 ON/OFF 设定 *2 0: OFF 1: ON
GR_AB7	GR_CK_ON	0	CLUT 参照 /RGB 参照色键处理的 ON/OFF 设定 0: OFF 1: ON

【注】 *1 在图形 (1) 中只能进行色键处理。在进行色键处理时，为了只显示当前图形，必须将替换色键对象像素的 α 值和非色键对象像素的 α 值设定为“255”。

*2 只有图形 (2) 和图形 (3) 有此功能，图形 (1) 没有此位。

35.1.7 背景色显示处理

显示 GR_BASE_G[7:0] 位、GR_BASE_B[7:0] 位和 GR_BASE_R[7:0] 位设定的颜色。

G 输出 = GR_BASE_G

B 输出 = GR_BASE_B

R 输出 = GR_BASE_R

表 35.20 背景色设定

寄存器名	位名	初始值	说明
GR_BASE	GR_BASE_G[7:0]	0	背景色 G 信号 G: 8 位、无符号 (0 ~ 255[LSB])
GR_BASE	GR_BASE_B[7:0]	0	背景色 B 信号 B: 8 位、无符号 (0 ~ 255[LSB])
GR_BASE	GR_BASE_R[7:0]	0	背景色 R 信号 R: 8 位、无符号 (0 ~ 255[LSB])

35.1.8 下层图形显示处理

显示下层图形。

G 输出 = 下层图形的 G 输入

B 输出 = 下层图形的 B 输入

R 输出 = 下层图形的 R 输入

35.1.9 当前图形显示处理

显示当前图形。

G 输出 = 当前图形的 G 输入

B 输出 = 当前图形的 B 输入

R 输出 = 当前图形的 R 输入

35.1.10 矩形区域 α 混合显示处理

以水平同步信号和垂直同步信号的上升沿为基准，通过 GR_ARC_HS[10:0] 位、GR_ARC_HW[10:0] 位、GR_ARC_VS[10:0] 位和 GR_ARC_VW[10:0] 位设定矩形区域 α 混合处理的矩形区域。图形 (1) 无此功能。

矩形区域 α 混合处理的区域设定如图 35.10 所示。

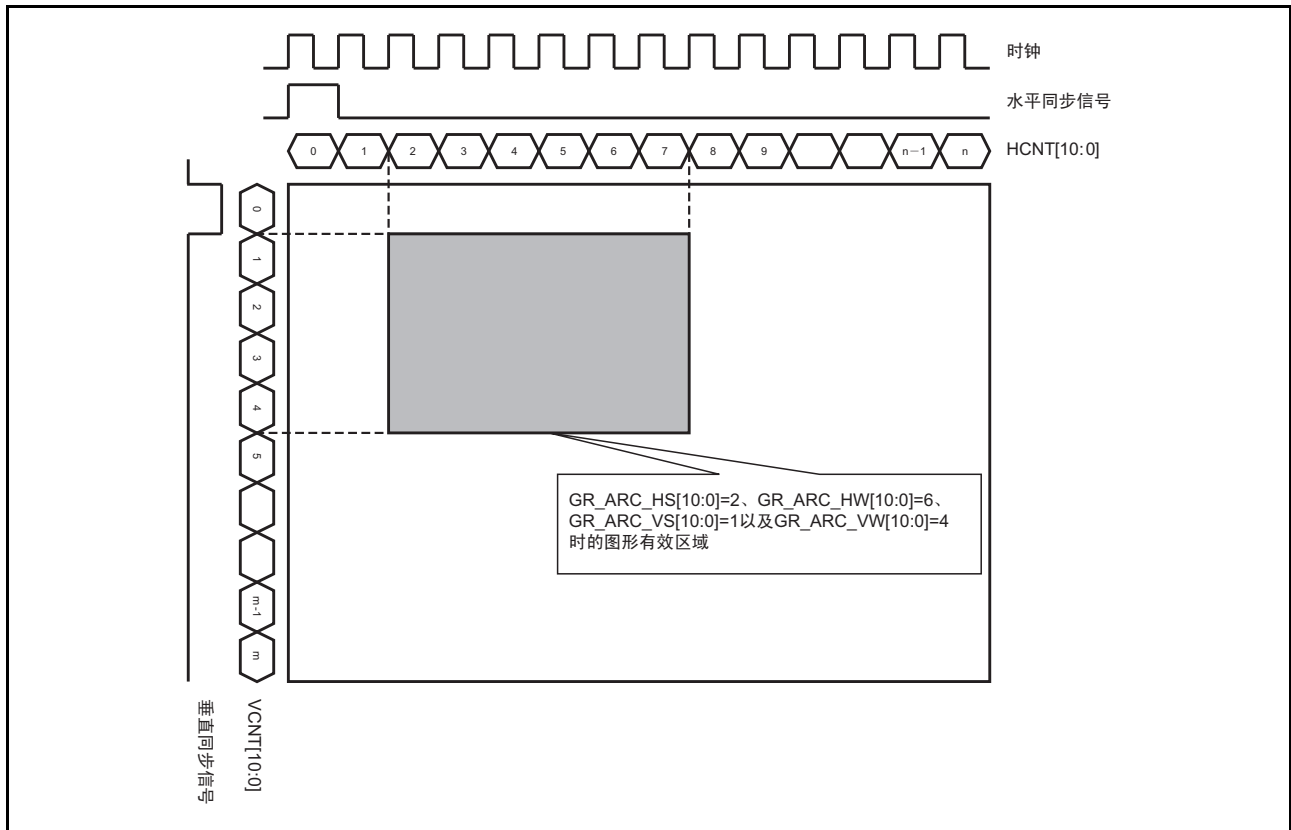


图 35.10 矩形区域 α 混合处理的区域设定

能通过将 GR_ARC_DISP_ON 位置 “1”，显示图形区的轮廓。

表 35.21 矩形区域 α 混合处理的区域设定

寄存器名	位名	初始值	说明
GR_AB5	GR_ARC_HS[10:0]	0	矩形区域 α 混合处理的有效图像区水平起始位置设定
GR_AB5	GR_ARC_HW[10:0]	0	矩形区域 α 混合处理的有效图像区水平宽度设定
GR_AB4	GR_ARC_VS[10:0]	0	矩形区域 α 混合处理的有效图像区垂直起始位置设定
GR_AB4	GR_ARC_VW[10:0]	0	矩形区域 α 混合处理的有效图像区垂直宽度设定
GR_AB1	GR_ARC_DISP_ON	0	矩形区域 α 混合处理的图像区轮廓显示 ON/OFF 设定 0: 不显示轮廓 1: 显示轮廓

矩形区域 α 混合处理通过 GR_ARC_DEF[7:0] 位、GR_ARC_MODE 位、GR_ARC_COEF[7:0] 位和 GR_ARC_RATE[7:0] 位设定淡入 / 淡出系数，进行当前图形的淡入和淡出。

首先，将 GR_ARC_DEF[7:0] 位赋值到 α 值。

每当垂直同步信号的上升次数和 GR_ARC_RATE[7:0] 位的值相同时， α 值加上或者减去 GR_ARC_COEF[7:0] 位的值。

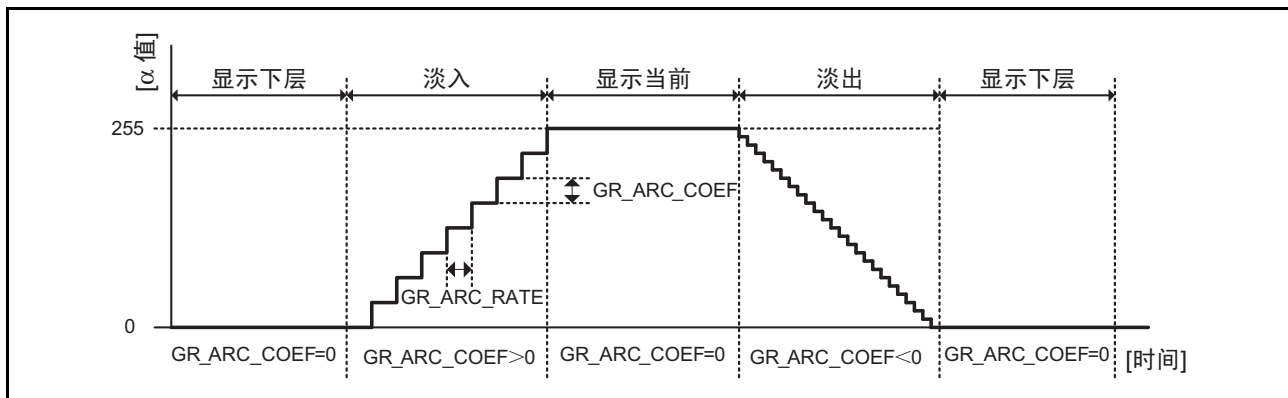


图 35.11 淡入和淡出

表 35.22 矩形区域 α 混合处理的设定

寄存器名	位名	初始值	说明
GR_AB7	GR_ARC_DEF[7:0]	0	矩形区域 α 混合处理的 α 初始值 【注】 不能在加法运算或者减法运算处理过程中（GR_ARC_ST 位为“1”）更改 α 初始值。要在上述状态下进行更改时，需要暂时将矩形区域 α 混合处理设定为 OFF（GR_ARC_ON 位为“0”）。
GR_AB6	GR_ARC_MODE	0	矩形区域 α 处理模式 0: 加法运算 1: 减法运算
GR_AB6	GR_ARC_COEF[7:0]	0	矩形区域 α 混合处理的 α 系数（0 ~ 255） [7:0]: 变化量（用绝对值表示）
GR_AB6	GR_ARC_RATE[7:0]	0	矩形区域 α 混合处理的帧率
GR_MON	GR_ARC_ST	—	矩形区域 α 混合处理的状态标志 0: 加法运算或者减法运算处理结束 （ α 值为“0”或者“255”的状态） 1: 正在进行加法运算或者减法运算处理

将下述表达式设定的值用于后述的 α 混合运算。

α 值 = 淡入 / 淡出系数

G 值 = 当前图形的 G 输入

B 值 = 当前图形的 B 输入

R 值 = 当前图形的 R 输入

35.1.11 RGB 参照色键显示处理

对满足下述 3 个表达式的像素进行 RGB 参照色键处理。

当前图形的 G 输入 = GR_CK_KG

当前图形的 B 输入 = GR_CK_KB

当前图形的 R 输入 = GR_CK_KR

在 RGB 参照色键处理中，将下述表达式设定的值用于后述的 α 混合运算式。

【 α RGB1555 格式的情况 (GR_FORMAT[3:0] 位为 “2”)】

α 值 = (在当前图形的 α 输入是 “1” 时，为 GR_A1；在当前图形的 α 输入是 “0” 时，为 GR_A0。)

G 值 = GR_CK_G

B 值 = GR_CK_B

R 值 = GR_CK_R

【其他格式的情况 (GR_FORMAT[3:0]=0、1、3、4)】

α 值 = GR_CK_A

G 值 = GR_CK_G

B 值 = GR_CK_B

R 值 = GR_CK_R

对不进行 RGB 参照色键处理的像素，将下述表达式设定的值用于后述的 α 混合运算式。

α 值 = 当前图形的 α 输入

G 值 = 当前图形的 G 输入

B 值 = 当前图形的 B 输入

R 值 = 当前图形的 R 输入

表 35.23 RGB 参照色键处理的设定

寄存器名	位名	初始值	说明
GR_AB8	GR_CK_KG[7:0]	0	RGB 参照色键处理对象 G 信号 G: 8 位、无符号 (0 ~ 255[LSB])
GR_AB8	GR_CK_KB[7:0]	0	RGB 参照色键处理对象 B 信号 B: 8 位、无符号 (0 ~ 255[LSB])
GR_AB8	GR_CK_KR[7:0]	0	RGB 参照色键处理对象 R 信号 R: 8 位、无符号 (0 ~ 255[LSB])
GR_AB9	GR_CK_A[7:0]	0	RGB 参照色键处理替换后的 α 信号 * α : 8 位、无符号 (0 ~ 255[LSB])
GR_AB9	GR_CK_G[7:0]	0	RGB 参照色键处理替换后的 G 信号 G: 8 位、无符号 (0 ~ 255[LSB])
GR_AB9	GR_CK_B[7:0]	0	RGB 参照色键处理替换后的 B 信号 B: 8 位、无符号 (0 ~ 255[LSB])
GR_AB9	GR_CK_R[7:0]	0	RGB 参照色键处理替换后的 R 信号 R: 8 位、无符号 (0 ~ 255[LSB])

【注】 * 在图形 (1) 中使用此功能时，必须将这些位置 “255”。

35.1.12 CLUT 参照色键显示处理

对满足下述表达式的像素进行 CLUT 参照色键处理。

当前图形的 CLUT 输入 = GR_CK_KCLUT

在 CLUT 参照色键处理中，将下述表达式设定的值用于后述的 α 混合运算式。

【CLUT1 格式的情况 (GR_FORMAT[3:0] 位为 “7”)】

α 值 = (在当前图形的 CLUT 输入是 “1” 时，为 GR_A1；在当前图形的 CLUT 输入是 “0” 时，为 GR_A0。)

G 值 = (在当前图形的 CLUT 输入是 “1” 时，为 GR_G1；在当前图形的 CLUT 输入是 “0” 时，为 GR_G0。)

B 值 = (在当前图形的 CLUT 输入是 “1” 时，为 GR_B1；在当前图形的 CLUT 输入是 “0” 时，为 GR_B0。)

R 值 = (在当前图形的 CLUT 输入是 “1” 时，为 GR_R1；在当前图形的 CLUT 输入是 “0” 时，为 GR_R0。)

【其他格式的情况 (GR_FORMAT[3:0]=5、6)】

α 值 = GR_CK_A

G 值 = GR_CK_G

B 值 = GR_CK_B

R 值 = GR_CK_R

对不进行 CLUT 参照色键处理的像素，将下述表达式设定的值用于后述的 α 混合运算式。

α 值 = 当前图形的 α 输入

G 值 = 当前图形的 G 输入

B 值 = 当前图形的 B 输入

R 值 = 当前图形的 R 输入

表 35.24 CLUT 参照色键处理的设定

寄存器名	位名	初始值	说明
GR_AB8	GR_CK_KCLUT[7:0]	0	CLUT 参照色键处理对象 CLUT 信号 CLUT: 8 位、无符号 (0 ~ 255[LSB])
GR_AB10	GR_A0[7:0]	0	CLUT1 的 α 0 信号 * 在 CLUT1 格式并且 CLUT1 为 “0” 时替换为 α 信号。 在 α RGB1555 格式并且 α 为 “0” 时替换为 α 信号。
GR_AB10	GR_G0[7:0]	0	CLUT1 的 G0 信号 在 CLUT1 格式并且 CLUT1 为 “0” 时替换为 G 信号。
GR_AB10	GR_B0[7:0]	0	CLUT1 的 B0 信号 在 CLUT1 格式并且 CLUT1 为 “0” 时替换为 B 信号。
GR_AB10	GR_R0[7:0]	0	CLUT1 的 R0 信号 在 CLUT1 格式并且 CLUT1 为 “0” 时替换为 R 信号。
GR_AB11	GR_A1[7:0]	0	CLUT1 的 α 1 信号 * 在 CLUT1 格式并且 CLUT1 为 “1” 时替换为 α 信号。 在 α RGB1555 格式并且 α 为 “1” 时替换为 α 信号。
GR_AB11	GR_G1[7:0]	0	CLUT1 的 G1 信号 在 CLUT1 格式并且 CLUT1 为 “1” 时替换为 G 信号。
GR_AB11	GR_B1[7:0]	0	CLUT1 的 B1 信号 在 CLUT1 格式并且 CLUT1 为 “1” 时替换为 B 信号。
GR_AB11	GR_R1[7:0]	0	CLUT1 的 R1 信号 在 CLUT1 格式并且 CLUT1 为 “1” 时替换为 R 信号。

【注】 * 在图形 (1) 中使用此功能时，必须将这些位置 “255”。

35.1.13 以像素为单位的 α 混合显示处理

在以像素为单位的 α 混合处理中，将下述表达式设定的值用于后述的 α 混合运算式。

α 值 = 当前图形的 α 输入

G 值 = 当前图形的 G 输入

B 值 = 当前图形的 B 输入

R 值 = 当前图形的 R 输入

35.1.14 α 混合运算式

使用信号 α 值进行的 2 个系统输入信号的 α 混合处理如下所示：

G 输出 = (G 值 \times α 值 + 下层图形的 G 输入 \times (255 - α 值)) \div 255

B 输出 = (B 值 \times α 值 + 下层图形的 B 输入 \times (255 - α 值)) \div 255

R 输出 = (R 值 \times α 值 + 下层图形的 R 输入 \times (255 - α 值)) \div 255

35.1.15 CLUT 表

在 CLUT8/CLUT4 的情况下，参照 CLUT 表并且展开为 α RGB8888；在 CLUT1 的情况下，参照寄存器并且展开为 α RGB8888。

CLUT 表的数据配置如图 35.12 所示。

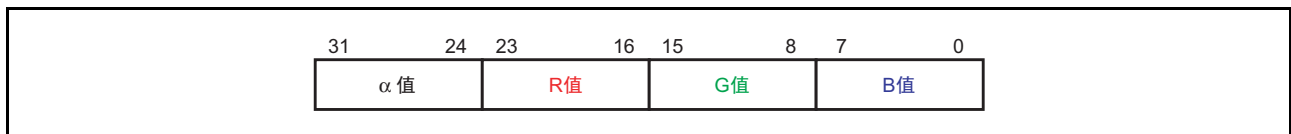


图 35.12 CLUT 表的数据配置

CLUT 表分配在下述地址：

图形 (1) 的 CLUT 表：H'FFFF6000 ~ H'FFFF63FF（在 CLUT4 的情况下，H'FFFF6000 ~ H'FFFF603F 有效）

图形 (2) 的 CLUT 表：H'FFFF6400 ~ H'FFFF67FF（在 CLUT4 的情况下，H'FFFF6400 ~ H'FFFF643F 有效）

图形 (3) 的 CLUT 表：H'FFFF6800 ~ H'FFFF6BFF（在 CLUT4 的情况下，H'FFFF6800 ~ H'FFFF683F 有效）

对于同一个地址，CLUT 表有 2 个面（CLUT 表 0 和 CLUT 表 1）。通过 GR_CLT_SEL 位选择并使用 CLUT 表。因此，即使在此模块参照 CLUT 表时，也能改写反面的 CLUT 表。

表 35.25 CLUT 表的选择设定

寄存器名	位名	初始值	说明
GR_CLUT	GR_CLT_SEL	0	CLUT 表选择信号 0: 选择 CLUT 表 0 参照 CLUT 表 0 并且展开为 α RGB8888。 CPU 能读写 CLUT 表 1。 1: 选择 CLUT 表 1 参照 CLUT 表 1 并且展开为 α RGB8888。 CPU 能读写 CLUT 表 0。

35.2 寄存器说明

寄存器结构如表 35.26 ~ 表 35.28 所示。

【说明寄存器时的符号】

初始值 : 复位后的寄存器值

— : 不定值

R/W : 可读写。能读到写入值。

R/WC0 : 可读写。如果写“0”，该位就被初始化，但是忽视写“1”。

R/WC1 : 可读写。如果写“1”，该位就被初始化，但是忽视写“0”。

R : 只能读。写入值总是“0”。

—/W : 只能写。读取值为不定值。

图形(2)的寄存器结构如表35.26所示，图形(3)的寄存器结构如表35.27所示，CLUT表结构如表35.28所示。缩放部记载了图形(1)的寄存器结构。

表 35.26 图像合成部 (图形 (2)) 寄存器结构

名称	略称	R/W	初始值	地址	存取长度
图形 (2) 的寄存器更新控制寄存器	GR2_UPDATE	R/WC1	H'0000 0000	H'FFFF 7700	32/16
帧缓冲器的读控制寄存器 (图形 (2))	GR2_FLM_RD	R/W	H'0000 0000	H'FFFF 7704	32/16
帧缓冲控制寄存器 1 (图形 (2))	GR2_FLM1	R/W	H'0000 0000	H'FFFF 7708	32/16
帧缓冲控制寄存器 2 (图形 (2))	GR2_FLM2	R/W	H'0000 0000	H'FFFF 770C	32/16
帧缓冲控制寄存器 3 (图形 (2))	GR2_FLM3	R/W	H'0000 0000	H'FFFF 7710	32/16
帧缓冲控制寄存器 4 (图形 (2))	GR2_FLM4	R/W	H'0000 0000	H'FFFF 7714	32/16
帧缓冲控制寄存器 5 (图形 (2))	GR2_FLM5	R/W	H'0000 03FF	H'FFFF 7718	32/16
帧缓冲控制寄存器 6 (图形 (2))	GR2_FLM6	R/W	H'0000 0000	H'FFFF 771C	32/16
α 混合控制寄存器 1 (图形 (2))	GR2_AB1	R/W	H'0000 0000	H'FFFF 7720	32/16
α 混合控制寄存器 2 (图形 (2))	GR2_AB2	R/W	H'0000 0000	H'FFFF 7724	32/16
α 混合控制寄存器 3 (图形 (2))	GR2_AB3	R/W	H'0000 0000	H'FFFF 7728	32/16
α 混合控制寄存器 4 (图形 (2))	GR2_AB4	R/W	H'0000 0000	H'FFFF 772C	32/16
α 混合控制寄存器 5 (图形 (2))	GR2_AB5	R/W	H'0000 0000	H'FFFF 7730	32/16
α 混合控制寄存器 6 (图形 (2))	GR2_AB6	R/W	H'0000 0000	H'FFFF 7734	32/16
α 混合控制寄存器 7 (图形 (2))	GR2_AB7	R/W	H'00FF 0000	H'FFFF 7738	32/16
α 混合控制寄存器 8 (图形 (2))	GR2_AB8	R/W	H'0000 0000	H'FFFF 773C	32/16
α 混合控制寄存器 9 (图形 (2))	GR2_AB9	R/W	H'0000 0000	H'FFFF 7740	32/16
α 混合控制寄存器 10 (图形 (2))	GR2_AB10	R/W	H'0000 0000	H'FFFF 7744	32/16
α 混合控制寄存器 11 (图形 (2))	GR2_AB11	R/W	H'0000 0000	H'FFFF 7748	32/16
背景色控制寄存器 (图形 (2))	GR2_BASE	R/W	H'0000 0000	H'FFFF 774C	32/16
CLUT 表控制寄存器 (图形 (2))	GR2_CLUT	R/W	H'0000 0000	H'FFFF 7750	32/16
状态监视寄存器 (图形 (2))	GR2_MON	R	H'0000 0000	H'FFFF 7754	32/16

表 35.27 图像合成部 (图形 (3)) 寄存器结构

名称	略称	R/W	初始值	地址	存取长度
图形 (3) 的寄存器更新控制寄存器	GR3_UPDATE	R/WC1	H'0000 0000	H'FFFF 7780	32/16
帧缓冲器的读控制寄存器 (图形 (3))	GR3_FLM_RD	R/W	H'0000 0000	H'FFFF 7784	32/16
帧缓冲控制寄存器 1 (图形 (3))	GR3_FLM1	R/W	H'0000 0000	H'FFFF 7788	32/16
帧缓冲控制寄存器 2 (图形 (3))	GR3_FLM2	R/W	H'0000 0000	H'FFFF 778C	32/16
帧缓冲控制寄存器 3 (图形 (3))	GR3_FLM3	R/W	H'0000 0000	H'FFFF 7790	32/16
帧缓冲控制寄存器 4 (图形 (3))	GR3_FLM4	R/W	H'0000 0000	H'FFFF 7794	32/16
帧缓冲控制寄存器 5 (图形 (3))	GR3_FLM5	R/W	H'0000 03FF	H'FFFF 7798	32/16
帧缓冲控制寄存器 6 (图形 (3))	GR3_FLM6	R/W	H'0000 0000	H'FFFF 779C	32/16
α 混合控制寄存器 1 (图形 (3))	GR3_AB1	R/W	H'0000 0000	H'FFFF 77A0	32/16
α 混合控制寄存器 2 (图形 (3))	GR3_AB2	R/W	H'0000 0000	H'FFFF 77A4	32/16
α 混合控制寄存器 3 (图形 (3))	GR3_AB3	R/W	H'0000 0000	H'FFFF 77A8	32/16
α 混合控制寄存器 4 (图形 (3))	GR3_AB4	R/W	H'0000 0000	H'FFFF 77AC	32/16
α 混合控制寄存器 5 (图形 (3))	GR3_AB5	R/W	H'0000 0000	H'FFFF 77B0	32/16
α 混合控制寄存器 6 (图形 (3))	GR3_AB6	R/W	H'0000 0000	H'FFFF 77B4	32/16
α 混合控制寄存器 7 (图形 (3))	GR3_AB7	R/W	H'00FF 0000	H'FFFF 77B8	32/16
α 混合控制寄存器 8 (图形 (3))	GR3_AB8	R/W	H'0000 0000	H'FFFF 77BC	32/16
α 混合控制寄存器 9 (图形 (3))	GR3_AB9	R/W	H'0000 0000	H'FFFF 77C0	32/16
α 混合控制寄存器 10 (图形 (3))	GR3_AB10	R/W	H'0000 0000	H'FFFF 77C4	32/16
α 混合控制寄存器 11 (图形 (3))	GR3_AB11	R/W	H'0000 0000	H'FFFF 77C8	32/16
背景色控制寄存器 (图形 (3))	GR3_BASE	R/W	H'0000 0000	H'FFFF 77CC	32/16
CLUT 表中断控制寄存器 (图形 (3))	GR3_CLUT_INT	R/W	H'0000 0000	H'FFFF 77D0	32/16
状态监视寄存器 (图形 (3))	GR3_MON	R	H'0000 0000	H'FFFF 77D4	32/16

表 35.28 CLUT 表结构

名称	略称	R/W	初始值	地址	存取长度
图形 (1) 的 CLUT 表	GR1_CLUT	R/W	—	H'FFFF 6000 ~ H'FFFF 63FF	32
图形 (2) 的 CLUT 表	GR2_CLUT	R/W	—	H'FFFF 6400 ~ H'FFFF 67FF	32
图形 (3) 的 CLUT 表	GR3_CLUT	R/W	—	H'FFFF 6800 ~ H'FFFF 6BFF	32

35.2.1 图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	GR2_P_VEN	—	—	—	GR2_IBUS_VEN
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	RWC1	R	R	R	RWC1

位	位名	初始值	R/W	说明
31 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	GR2_P_VEN	0	R/WC1	图形显示的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GR2_IBUS_VEN	0	R/WC1	读帧缓冲器的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器

35.2.2 帧缓冲器的读控制寄存器 (图形 (2)) (GR2_FLM_RD)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR2_R_ENB
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GR2_R_ENB	0	R/W	帧缓冲器的读允许 0: 禁止读 1: 允许读

【注】 在全部图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_IBUS_VEN 位为“1”时更新此寄存器。

35.2.3 帧缓冲控制寄存器 1 (图形 (2)) (GR2_FLM1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR2_LN_OFF_DIR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR2_FLM_SEL[1:0]	—	—	—	—	—	—	—	—	GR2_BST_MD
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 17	—	全 0	R	保留位 读写值都为“0”。
16	GR2_LN_OFF_DIR	0	R/W	帧缓冲器的行偏移地址方向设定 0: 递增行偏移地址 1: 递减行偏移地址
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9、8	GR2_FLM_SEL [1:0]	0	R/W	帧缓冲器地址设定信号的选择 0: 选择帧 0 1: 选择 GR2_FLM_NUM 2: 选择帧 0 3: 禁止设定
7 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GR2_BST_MD	0	R/W	帧缓冲器的突发传送模式 0: 32 字节 1: 128 字节

【注】 在图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_IBUS_VEN 位为“1”时更新 GR2_LN_OFF_DIR 位和 GR2_FLM_SEL 位。

在 GR2_IBUS_VEN 位为“1”并且图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_P_VEN 位为“1”时更新 GR2_BST_MD。

35.2.4 帧缓冲控制寄存器 2 (图形 (2)) (GR2_FLM2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR2_BASE[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_BASE[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 0	GR2_BASE[31:0]	0	R/W	帧缓冲器的基址 设定保存帧数据的帧缓冲器的起始地址。 在进行 32 字节突发传送时参照 GR_BASE[4:3] 位, 在进行 128 字节突发传送时参照 GR_BASE[6:3] 位, 并且跳读起始行数据。 必须将低 3 位固定为“000”。

【注】 在全部图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_IBUS_VEN 位和 GR2_P_VEN 位都为“1”时更新此寄存器。

35.2.5 帧缓冲控制寄存器 3 (图形 (2)) (GR2_FLM3)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	GR2_LN_OFF[14:0]														
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR2_FLM_NUM[9:0]									
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31	—	0	R	保留位 读写值都为“0”。
30 ~ 16	GR2_LN_OFF [14:0]	0	R/W	帧缓冲器的行偏移地址 设定计算行起始地址时的行偏移地址。 行 0: GR2_BASE 行 1: GR2_BASE+GR2_LN_OFF×1 : 行 n: GR2_BASE+GR2_LN_OFF×n 在进行 32 字节传送时, 必须将低 5 位固定为“0_0000”。 在进行 128 字节传送时, 必须将低 7 位固定为“000_0000”。
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	GR2_FLM_NUM [9:0]	0	R/W	帧缓冲器的帧号 在 GR2_FLM_SEL 位为“1”时手动设定帧号。

【注】 在全部图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_IBUS_VEN 位为“1”时更新此寄存器。

35.2.6 帧缓冲控制寄存器 4 (图形 (2)) (GR2_FLM4)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	GR2_FLM_OFF[22:16]						
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_FLM_OFF[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 23	—	全 0	R	保留位 读写值都为“0”。
22 ~ 0	GR2_FLM_OFF [22:0]	0	R/W	帧缓冲器的帧偏移地址 在使用多面帧缓冲器时, 设定计算各帧起始地址时的帧偏移地址。 缓冲器 0: GR2_BASE 缓冲器 1: GR2_BASE+GR2_FLM_OFF×1 : 缓冲器 n: GR2_BASE+GR2_FLM_OFF×n 在进行 32 字节传送时, 必须将低 5 位固定为“0_0000”。 在进行 128 字节传送时, 必须将低 7 位固定为“000_0000”。

【注】 在全部图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_IBUS_VEN 位为“1”时更新此寄存器。

35.2.7 帧缓冲控制寄存器 5 (图形 (2)) (GR2_FLM5)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	GR2_FLM_LNUM[9:0]									
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR2_FLM_LOOP[9:0]									
初始值:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 26	—	全 0	R	保留位 读写值都为“0”。
25 ~ 16	GR2_FLM_LNUM [9:0]	0	R/W	1 帧的行数设定 行数为 (GR2_FLM_LNUM+1) 行。
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	GR2_FLM_LOOP [9:0]	1023	R/W	链状读地址时的行数 行数为 (GR2_FLM_LOOP+1) 行。

【注】 在全部图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_IBUS_VEN 位为“1”时更新此寄存器。

35.2.8 帧缓冲控制寄存器 6 (图形 (2)) (GR2_FLM6)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR2_FORMAT [3:0]				—	—	GR2_HW[9:0]									
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	GR2_ENDIAN_ON	—	—	—	—	—	—	GR2_STA_POS[5:0]					
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 28	GR2_FORMAT [3:0]	0	R/W	帧缓冲器读信号的格式设定 0: RGB565 1: RGB888 2: αRGB1555 3: αRGB4444 4: αRGB8888 5: CLUT8 6: CLUT4 7: CLUT1 8 ~ 15: 禁止设定
27、26	—	全 0	R	保留位 读写值都为“0”。
25 ~ 16	GR2_HW[9:0]	0	R/W	水平有效期间的宽度设定 宽度为 (GR2_HW+1) 个像素。 【注】 必须设定大于等于 2 的值。
15 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12	GR2_ENDIAN_ON	0	R/W	缓冲器读数据的字节序控制 ON/OFF 设定 0: OFF 1: ON
11 ~ 6	—	全 0	R	保留位 读写值都为“0”。
5 ~ 0	GR2_STA_POS [5:0]	0	R/W	数据的跳读量 从行首开始跳读 GR2_STA_POS 个数据。

【注】 在图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_P_VEN 位为“1”时更新 GR2_ENDIAN_ON 位和 GR2_STA_POS 位。

在图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_IBUS_VEN 位和 GR2_P_VEN 位都为“1”时更新 GR2_FORMAT 位和 GR2_HW 位。

35.2.9 α 混合控制寄存器 1 (图形 (2)) (GR2_AB1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	GR2_ARC_ON	—	—	—	GR2_ARC_DISP_ON	—	—	—	GR2_GRC_DISP_ON	—	—	GR2_DISP_SEL[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12	GR2_ARC_ON	0	R/W	矩形区域 α 混合处理的 ON/OFF 设定 0: OFF 1: ON
11 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	GR2_ARC_DISP_ON	0	R/W	矩形区域 α 混合处理的图像区轮廓显示 ON/OFF 设定 0: 不显示轮廓 1: 显示轮廓
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	GR2_GRC_DISP_ON	0	R/W	图形图像区的轮廓显示 ON/OFF 设定 0: 不显示轮廓 1: 显示轮廓
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	GR2_DISP_SEL [1:0]	0	R/W	图形显示设定 0: 显示背景色 1: 显示下层图形 2: 显示当前图形 3: 混合显示下层图形和当前图形

【注】 在全部图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_P_VEN 位为“1”时更新此寄存器。

35.2.10 α 混合控制寄存器 2 (图形 (2)) (GR2_AB2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR2_GRC_VS[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR2_GRC_VW[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	GR2_GRC_VS [10:0]	0	R/W	图形图像区的垂直起始位置设定 【注】 必须至少设定 4 行, 使 GR2_GRC_VS+GR2_GRC_VW 不超过 2039 行。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	GR2_GRC_VW [10:0]	0	R/W	图形图像区的垂直宽度设定

【注】 在全部图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_P_VEN 位为“1”时更新此寄存器。

35.2.11 α 混合控制寄存器 3 (图形 (2)) (GR2_AB3)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR2_GRC_HS[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR2_GRC_HW[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	GR2_GRC_HS [10:0]	0	R/W	图形图像区的水平起始位置设定 【注】 必须至少设定 16 个时钟, 使 GR2_GRC_HS+GR2_GRC_HW 不超过 2015 个时钟。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	GR2_GRC_HW [10:0]	0	R/W	图形图像区的水平宽度设定 【注】 在显示水平宽度为“1”或者“2”的像素时, 必须将 GR2_HW 位置“2”, 将 GR2_GRC_HW 位置“1”(1 个像素)或者“2”(2 个像素)。

【注】 在全部图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_P_VEN 位为“1”时更新此寄存器。

35.2.12 α 混合控制寄存器 4 (图形 (2)) (GR2_AB4)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR2_ARC_VS[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR2_ARC_VW[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	GR2_ARC_VS [10:0]	0	R/W	矩形区域 α 混合处理的有效图像区垂直起始位置设定
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	GR2_ARC_VW [10:0]	0	R/W	矩形区域 α 混合处理的有效图像区垂直宽度设定

【注】 在全部图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_P_VEN 位为“1”时更新此寄存器。

35.2.13 α 混合控制寄存器 5 (图形 (2)) (GR2_AB5)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR2_ARC_HS[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR2_ARC_HW[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	GR2_ARC_HS [10:0]	0	R/W	矩形区域 α 混合处理的有效图像区水平起始位置设定
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	GR2_ARC_HW [10:0]	0	R/W	矩形区域 α 混合处理的有效图像区水平宽度设定

【注】 在全部图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_P_VEN 位为“1”时更新此寄存器。

35.2.14 α 混合控制寄存器 6 (图形 (2)) (GR2_AB6)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	GR2_ARC_MODE	GR2_ARC_COEF[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	GR2_ARC_RATE[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 25	—	全 0	R	保留位 读写值都为“0”。
24	GR2_ARC_MODE	0	R/W	矩形区域 α 混合处理模式 0: 加法运算 1: 减法运算
23 ~ 16	GR2_ARC_COEF[7:0]	0	R/W	矩形区域 α 混合处理的 α 系数 (0 ~ 255) [7:0]: 变化量 (用绝对值表示)
15 ~ 8	—	全 0	R	保留位 读写值都为“0”。
7 ~ 0	GR2_ARC_RATE[7:0]	0	R/W	矩形区域 α 混合处理的帧率

【注】 在全部图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_P_VEN 位为“1”时更新此寄存器。

35.2.15 α 混合控制寄存器 7 (图形 (2)) (GR2_AB7)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR2_ARC_DEF[7:0]							
初始值:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR2_CK_ON
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23 ~ 16	GR2_ARC_DEF [7:0]	255	R/W	矩形区域 α 混合处理的 α 初始值 【注】 不能在加法运算或者减法运算处理过程中 (GR2_ARC_ST 位为“1”) 更改 α 初始值。要在上述状态下进行更改时, 需要暂时将矩形区域 α 混合处理设定为 OFF (GR_ARC2_ON 位为“0”)。
15 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GR2_CK_ON	0	R/W	CLUT 参照 /RGB 参照色键处理的 ON/OFF 设定 0: OFF 1: ON

【注】 在全部图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_P_VEN 位为“1”时更新此寄存器。

35.2.16 α 混合控制寄存器 8 (图形 (2)) (GR2_AB8)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR2_CK_KCLUT[7:0]								GR2_CK_KG[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_CK_KB[7:0]								GR2_CK_KR[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GR2_CK_KCLUT [7:0]	0	R/W	CLUT 参照色键处理对象 CLUT 信号 CLUT: 8 位、无符号 (0 ~ 255[LSB])
23 ~ 16	GR2_CK_KG[7:0]	0	R/W	RGB 参照色键处理对象 G 信号 G: 8 位、无符号 (0 ~ 255[LSB])
15 ~ 8	GR2_CK_KB[7:0]	0	R/W	RGB 参照色键处理对象 B 信号 B: 8 位、无符号 (0 ~ 255[LSB])
7 ~ 0	GR2_CK_KR[7:0]	0	R/W	RGB 参照色键处理对象 R 信号 R: 8 位、无符号 (0 ~ 255[LSB])

【注】 在全部图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_P_VEN 位为 “1” 时更新此寄存器。

35.2.17 α 混合控制寄存器 9 (图形 (2)) (GR2_AB9)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR2_CK_A[7:0]								GR2_CK_G[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_CK_B[7:0]								GR2_CK_R[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GR2_CK_A[7:0]	0	R/W	RGB/CLUT 参照色键处理替换后的 α 信号 α : 8 位、无符号 (0 ~ 255[LSB])
23 ~ 16	GR2_CK_G[7:0]	0	R/W	RGB/CLUT 参照色键处理替换后的 G 信号 G: 8 位、无符号 (0 ~ 255[LSB])
15 ~ 8	GR2_CK_B[7:0]	0	R/W	RGB/CLUT 参照色键处理替换后的 B 信号 B: 8 位、无符号 (0 ~ 255[LSB])
7 ~ 0	GR2_CK_R[7:0]	0	R/W	RGB/CLUT 参照色键处理替换后的 R 信号 R: 8 位、无符号 (0 ~ 255[LSB])

【注】 在全部图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_P_VEN 位为 “1” 时更新此寄存器。

35.2.18 α 混合控制寄存器 10 (图形 (2)) (GR2_AB10)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR2_A0[7:0]								GR2_G0[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_B0[7:0]								GR2_R0[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GR2_A0[7:0]	0	R/W	CLUT1 的 α 0 信号 在 CLUT1 格式并且 CLUT1 为 “0” 时替换为 α 信号。 在 α RGB1555 格式并且 α 为 “0” 时替换为 α 信号。
23 ~ 16	GR2_G0[7:0]	0	R/W	CLUT1 的 G0 信号 在 CLUT1 格式并且 CLUT1 为 “0” 时替换为 G 信号。
15 ~ 8	GR2_B0[7:0]	0	R/W	CLUT1 的 B0 信号 在 CLUT1 格式并且 CLUT1 为 “0” 时替换为 B 信号。
7 ~ 0	GR2_R0[7:0]	0	R/W	CLUT1 的 R0 信号 在 CLUT1 格式并且 CLUT1 为 “0” 时替换为 R 信号。

【注】 在全部图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_P_VEN 位为 “1” 时更新此寄存器。

35.2.19 α 混合控制寄存器 11 (图形 (2)) (GR2_AB11)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR2_A1[7:0]								GR2_G1[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_B1[7:0]								GR2_R1[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GR2_A1[7:0]	0	R/W	CLUT1 的 α 1 信号 在 CLUT1 格式并且 CLUT1 为 “1” 时替换为 α 信号。 在 α RGB1555 格式并且 α 为 “1” 时替换为 α 信号。
23 ~ 16	GR2_G1[7:0]	0	R/W	CLUT1 的 G1 信号 在 CLUT1 格式并且 CLUT1 为 “1” 时替换为 G 信号。
15 ~ 8	GR2_B1[7:0]	0	R/W	CLUT1 的 B1 信号 在 CLUT1 格式并且 CLUT1 为 “1” 时替换为 B 信号。
7 ~ 0	GR2_R1[7:0]	0	R/W	CLUT1 的 R1 信号 在 CLUT1 格式并且 CLUT1 为 “1” 时替换为 R 信号。

【注】 在全部图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_P_VEN 位为 “1” 时更新此寄存器。

35.2.20 背景色控制寄存器 (图形 (2)) (GR2_BASE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR2_BASE_G[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_BASE_B[7:0]								GR2_BASE_R[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23 ~ 16	GR2_BASE_G [7:0]	0	R/W	背景色 G 信号 G: 8 位、无符号 (0 ~ 255[LSB])
15 ~ 8	GR2_BASE_B [7:0]	0	R/W	背景色 B 信号 B: 8 位、无符号 (0 ~ 255[LSB])
7 ~ 0	GR2_BASE_R [7:0]	0	R/W	背景色 R 信号 R: 8 位、无符号 (0 ~ 255[LSB])

【注】 在全部图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_P_VEN 位为“1”时更新此寄存器。

35.2.21 CLUT 表控制寄存器 (图形 (2)) (GR2_CLUT)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR2_CLT_SEL
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 17	—	全 0	R	保留位 读写值都为“0”。
16	GR2_CLT_SEL	0	R/W	CLUT 表的选择信号 0: 选择 CLUT 表 0 参照 CLUT 表 0 并且展开为 α RGB8888。 CPU 能读写 CLUT 表 1。 1: 选择 CLUT 表 1 参照 CLUT 表 1 并且展开为 α RGB8888。 CPU 能读写 CLUT 表 0。
15 ~ 0	—	全 0	R	保留位 读写值都为“0”。

【注】 在全部图形 (2) 的寄存器更新控制寄存器 (GR2_UPDATE) 的 GR2_P_VEN 位为“1”时更新此寄存器。

35.2.22 状态监视寄存器 (图形 (2)) (GR2_MON)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR2_ARC_ST
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GR2_ARC_ST	0	R	矩形区域 α 混合处理的状态标志 0: 加法运算或者减法运算处理结束 (α 值为“0”或者“255”的状态) 1: 正在进行加法运算或者减法运算

35.2.23 图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	GR3_P_VEN	—	—	—	GR3_IBUS_VEN
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	RWC1	R	R	R	RWC1

位	位名	初始值	R/W	说明
31 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	GR3_P_VEN	0	R/WC1	图形显示的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GR3_IBUS_VEN	0	R/WC1	帧缓冲器读的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器

35.2.24 帧缓冲器的读控制寄存器 (图形 (3)) (GR3_FLM_RD)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR3_R_ENB
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GR3_R_ENB	0	R/W	帧缓冲器的读允许 0: 禁止读 1: 允许读

【注】 在全部图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_IBUS_VEN 位为“1”时更新此寄存器。

35.2.25 帧缓冲控制寄存器 1 (图形 (3)) (GR3_FLM1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR3_LN_OFF_DIR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR3_FLM_SEL[1:0]	—	—	—	—	—	—	—	—	GR3_BST_MD
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 17	—	全 0	R	保留位 读写值都为“0”。
16	GR3_LN_OFF_DIR	0	R/W	帧缓冲器的行偏移地址方向设定 0: 递增行偏移地址 1: 递减行偏移地址
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9、8	GR3_FLM_SEL [1:0]	0	R/W	帧缓冲器地址设定信号的选择 0: 选择帧 0 1: 选择 GR3_FLM_NUM 2: 选择帧 0 3: 禁止设定
7 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GR3_BST_MD	0	R/W	帧缓冲器的突发传送模式 0: 32 字节 1: 128 字节

【注】 在图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_IBUS_VEN 位为“1”时更新 GR3_LN_OFF_DIR 位和 GR3_FLM_SEL 位。

在图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_IBUS_VEN 位和 GR3_P_VEN 位都为“1”时更新 GR3_BST_MD 位。

35.2.26 帧缓冲控制寄存器 2 (图形 (3)) (GR3_FLM2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR3_BASE[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_BASE[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 0	GR3_BASE[31:0]	0	R/W	帧缓冲器的基址 设定保存帧数据的帧缓冲器的起始地址。 在进行 32 字节突发传送时参照 GR_BASE[4:3] 位, 在进行 128 字节突发传送时参照 GR_BASE[6:3] 位, 并且跳读起始行数据。 必须将低 3 位固定为“000”。

【注】 在全部图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_IBUS_VEN 位和 GR3_P_VEN 位都为“1”时更新此寄存器。

35.2.27 帧缓冲控制寄存器 3 (图形 (3)) (GR3_FLM3)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	GR3_LN_OFF[14:0]														
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR3_FLM_NUM[9:0]									
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31	—	0	R	保留位 读写值都为“0”。
30 ~ 16	GR3_LN_OFF [14:0]	0	R/W	帧缓冲器的行偏移地址 设定计算行起始地址时的行偏移地址。 行 0: GR3_BASE 行 1: GR3_BASE+GR3_LN_OFF×1 : 行 n: GR3_BASE+GR3_LN_OFF×n 在进行 32 字节传送时, 必须将低 5 位固定为“0_0000”。 在进行 128 字节传送时, 必须将低 7 位固定为“000_0000”。
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	GR3_FLM_NUM [9:0]	0	R/W	帧缓冲器的帧号 在 GR_FLM_SEL 位为“1”时手动设定帧号。

【注】 在全部图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_IBUS_VEN 位为“1”时更新此寄存器。

35.2.28 帧缓冲控制寄存器 4 (图形 (3)) (GR3_FLM4)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	GR3_FLM_OFF[22:16]						
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_FLM_OFF[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 23	—	全 0	R	保留位 读写值都为“0”。
22 ~ 0	GR3_FLM_OFF [22:0]	0	R/W	帧缓冲器的帧偏移地址 在使用多面帧缓冲器时, 设定计算各帧起始地址时的帧偏移地址。 缓冲器 0: GR3_BASE 缓冲器 1: GR3_BASE+GR3_FLM_OFF×1 : 缓冲器 n: GR3_BASE+GR3_FLM_OFF×n 在进行 32 字节传送时, 必须将低 5 位固定为“0_0000”。 在进行 128 字节传送时, 必须将低 7 位固定为“000_0000”。

【注】 在全部图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_IBUS_VEN 位为“1”时更新此寄存器。

35.2.29 帧缓冲控制寄存器 5 (图形 (3)) (GR3_FLM5)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	GR3_FLM_LNUM[9:0]									
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR3_FLM_LOOP[9:0]									
初始值:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 26	—	全 0	R	保留位 读写值都为“0”。
25 ~ 16	GR3_FLM_LNUM [9:0]	0	R/W	1 帧的行数设定 行数为 (GR3_FLM_LNUM+1) 行。
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	GR3_FLM_LOOP [9:0]	1023	R/W	链状读地址时的行数 行数为 (GR3_FLM_LOOP+1) 行。

【注】 在全部图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_IBUS_VEN 位为“1”时更新此寄存器。

35.2.30 帧缓冲控制寄存器 6 (图形 (3)) (GR3_FLM6)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR3_FORMAT[3:0]				—	—	GR3_HW[9:0]									
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	GR3_ENDIAN_ON	—	—	—	—	—	—	GR3_STA_POS[5:0]					
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 28	GR3_FORMAT [3:0]	0	R/W	帧缓冲器读信号的格式设定 0: RGB565 1: RGB888 2: αRGB1555 3: αRGB4444 4: αRGB8888 5: CLUT8 6: CLUT4 7: CLUT1 8 ~ 15: 禁止设定
27、26	—	全 0	R	保留位 读写值都为“0”。
25 ~ 16	GR3_HW[9:0]	0	R/W	水平有效期间的宽度设定 宽度为 (GR3_HW+1) 个像素。 【注】必须设定大于等于 2 的值。
15 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12	GR3_ENDIAN_ON	0	R/W	缓冲器读数据的字节序控制 ON/OFF 设定 0: OFF 1: ON
11 ~ 6	—	全 0	R	保留位 读写值都为“0”。
5 ~ 0	GR3_STA_POS [5:0]	0	R/W	数据的跳读量 从行首跳读 GR3_STA_POS 个数据。

【注】在图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_P_VEN 位为“1”时更新 GR3_ENDIAN_ON 位和 GR3_STA_POS 位。

在图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_IBUS_VEN 位和 GR3_P_VEN 位都为“1”时更新 GR3_FORMAT 位和 GR3_HW 位。

35.2.31 α 混合控制寄存器 1 (图形 (3)) (GR3_AB1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	GR3 ARC_ON	—	—	—	GR3 ARC DISP_ON	—	—	—	GR3 GRC DISP_ON	—	—	GR3_DISP_SEL[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12	GR3_ARC_ON	0	R/W	矩形区域 α 混合处理的 ON/OFF 设定 0: OFF 1: ON
11 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	GR3_ARC_DISP_ON	0	R/W	矩形区域 α 混合处理的图像区轮廓显示 ON/OFF 设定 0: 不显示轮廓 1: 显示轮廓
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	GR3_GRC_DISP_ON	0	R/W	图形图像区的轮廓显示 ON/OFF 设定 0: 不显示轮廓 1: 显示轮廓
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	GR3_DISP_SEL [1:0]	0	R/W	图形显示设定 0: 显示背景色 1: 显示下层图形 2: 显示当前图形 3: 混合显示下层图形和当前图形

【注】 在全部图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_P_VEN 位为“1”时更新此寄存器。

35.2.32 α 混合控制寄存器 2 (图形 (3)) (GR3_AB2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR3_GRC_VS[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR3_GRC_VW[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	GR3_GRC_VS [10:0]	0	R/W	图形图像区的垂直起始位置设定 【注】 必须至少设定 4 行, 使 GR3_GRC_VS+GR3_GRC_VW 不超过 2039 行。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	GR3_GRC_VW [10:0]	0	R/W	图形图像区的垂直宽度设定

【注】 在全部图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_P_VEN 位为“1”时更新此寄存器。

35.2.33 α 混合控制寄存器 3 (图形 (3)) (GR3_AB3)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR3_GRC_HS[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR3_GRC_HW[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	GR3_GRC_HS [10:0]	0	R/W	图形图像区的水平起始位置设定 【注】 必须至少设定 16 个时钟, 使 GR3_GRC_HS+GR3_GRC_HW 不超过 2015 个时钟。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	GR3_GRC_HW [10:0]	0	R/W	图形图像区的水平宽度设定 【注】 在显示水平宽度为“1”或者“2”的像素时, 必须将 GR3_HW 位置“2”, 将 GR3_GRC_HW 位置“1”(1 个像素)或者“2”(2 个像素)。

【注】 在全部图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_P_VEN 位为“1”时更新此寄存器。

35.2.34 α 混合控制寄存器 4 (图形 (3)) (GR3_AB4)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR3_ARC_VS[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR3_ARC_VW[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	GR3_ARC_VS [10:0]	0	R/W	矩形区域 α 混合处理的有效图像区垂直起始位置设定
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	GR3_ARC_VW [10:0]	0	R/W	矩形区域 α 混合处理的有效图像区垂直宽度设定

【注】 在全部图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_P_VEN 位为“1”时更新此寄存器。

35.2.35 α 混合控制寄存器 5 (图形 (3)) (GR3_AB5)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR3_ARC_HS[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR3_ARC_HW[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	GR3_ARC_HS [10:0]	0	R/W	矩形区域 α 混合处理的有效图像区水平起始位置设定
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	GR3_ARC_HW [10:0]	0	R/W	矩形区域 α 混合处理的有效图像区水平宽度设定

【注】 在全部图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_P_VEN 位为“1”时更新此寄存器。

35.2.36 α 混合控制寄存器 6 (图形 (3)) (GR3_AB6)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	GR3_ARC_MODE	GR3_ARC_COEF[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	GR3_ARC_RATE[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 25	—	全 0	R	保留位 读写值都为“0”。
24	GR3_ARC_MODE	0	R/W	矩形区域 α 混合处理模式 0: 加法运算 1: 减法运算
23 ~ 16	GR3_ARC_COEF[7:0]	0	R/W	矩形区域 α 混合处理的 α 系数 (0 ~ 255) [7:0] 位: 变化量 (用绝对值表示)
15 ~ 8	—	全 0	R	保留位 读写值都为“0”。
7 ~ 0	GR3_ARC_RATE[7:0]	0	R/W	矩形区域 α 混合处理的帧率

【注】 在全部图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_P_VEN 位为“1”时更新此寄存器。

35.2.37 α 混合控制寄存器 7 (图形 (3)) (GR3_AB7)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR3_ARC_DEF[7:0]							
初始值:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR3_CK_ON
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23 ~ 16	GR3_ARC_DEF [7:0]	255	R/W	矩形区域 α 混合处理的 α 初始值 【注】不能在加法运算或者减法运算处理过程中 (GR3_ARC_ST 位为“1”) 更改 α 初始值。要在上述状态下进行更改时, 需要暂时将矩形区域 α 混合处理设定为 OFF (GR3_ARC_ON 位为“0”)。
15 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GR3_CK_ON	0	R/W	CLUT 参照 /RGB 参照色键处理的 ON/OFF 设定 0: OFF 1: ON

【注】 在全部图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_P_VEN 位为“1”时更新此寄存器。

35.2.38 α 混合控制寄存器 8 (图形 (3)) (GR3_AB8)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR3_CK_KCLUT[7:0]								GR3_CK_KG[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_CK_KB[7:0]								GR3_CK_KR[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GR3_CK_KCLUT [7:0]	0	R/W	CLUT 参照色键处理对象 CLUT 信号 CLUT: 8 位、无符号 (0 ~ 255[LSB])
23 ~ 16	GR3_CK_KG [7:0]	0	R/W	RGB 参照色键处理对象 G 信号 G: 8 位、无符号 (0 ~ 255[LSB])
15 ~ 8	GR3_CK_KB [7:0]	0	R/W	RGB 参照色键处理对象 B 信号 B: 8 位、无符号 (0 ~ 255[LSB])
7 ~ 0	GR3_CK_KR [7:0]	0	R/W	RGB 参照色键处理对象 R 信号 R: 8 位、无符号 (0 ~ 255[LSB])

【注】 在全部图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_P_VEN 位为 “1” 时更新此寄存器。

35.2.39 α 混合控制寄存器 9 (图形 (3)) (GR3_AB9)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR3_CK_A[7:0]								GR3_CK_G[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_CK_B[7:0]								GR3_CK_R[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GR3_CK_A[7:0]	0	R/W	RGB/CLUT 参照色键处理替换后的 α 信号 α : 8 位、无符号 (0 ~ 255[LSB])
23 ~ 16	GR3_CK_G[7:0]	0	R/W	RGB/CLUT 参照色键处理替换后的 G 信号 G: 8 位、无符号 (0 ~ 255[LSB])
15 ~ 8	GR3_CK_B[7:0]	0	R/W	RGB/CLUT 参照色键处理替换后的 B 信号 B: 8 位、无符号 (0 ~ 255[LSB])
7 ~ 0	GR3_CK_R[7:0]	0	R/W	RGB/CLUT 参照色键处理替换后的 R 信号 R: 8 位、无符号 (0 ~ 255[LSB])

【注】 在全部图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_P_VEN 位为 “1” 时更新此寄存器。

35.2.40 α 混合控制寄存器 10 (图形 (3)) (GR3_AB10)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR3_A0[7:0]								GR3_G0[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_B0[7:0]								GR3_R0[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GR3_A0[7:0]	0	R/W	CLUT1 的 α 0 信号 在 CLUT1 格式并且 CLUT1 为 “0” 时替换为 α 信号。 在 α RGB1555 格式并且 α 为 “0” 时替换为 α 信号。
23 ~ 16	GR3_G0[7:0]	0	R/W	CLUT1 的 G0 信号 在 CLUT1 格式并且 CLUT1 为 “0” 时替换为 G 信号。
15 ~ 8	GR3_B0[7:0]	0	R/W	CLUT1 的 B0 信号 在 CLUT1 格式并且 CLUT1 为 “0” 时替换为 B 信号。
7 ~ 0	GR3_R0[7:0]	0	R/W	CLUT1 的 R0 信号 在 CLUT1 格式并且 CLUT1 为 “0” 时替换为 R 信号。

【注】 在全部图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_P_VEN 位为 “1” 时更新此寄存器。

35.2.41 α 混合控制寄存器 11 (图形 (3)) (GR3_AB11)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR3_A1[7:0]								GR3_G1[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_B1[7:0]								GR3_R1[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GR3_A1[7:0]	0	R/W	CLUT1 的 α 1 信号 在 CLUT1 格式并且 CLUT1 为 “1” 时替换为 α 信号。 在 α RGB1555 格式并且 α 为 “1” 时替换为 α 信号。
23 ~ 16	GR3_G1[7:0]	0	R/W	CLUT1 的 G1 信号 在 CLUT1 格式并且 CLUT1 为 “1” 时替换为 G 信号。
15 ~ 8	GR3_B1[7:0]	0	R/W	CLUT1 的 B1 信号 在 CLUT1 格式并且 CLUT1 为 “1” 时替换为 B 信号。
7 ~ 0	GR3_R1[7:0]	0	R/W	CLUT1 的 R1 信号 在 CLUT1 格式并且 CLUT1 为 “1” 时替换为 R 信号。

【注】 在全部图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_P_VEN 位为 “1” 时更新此寄存器。

35.2.42 背景色控制寄存器 (图形 (3)) (GR3_BASE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR3_BASE_G[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_BASE_B[7:0]								GR3_BASE_R[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23 ~ 16	GR3_BASE_G [7:0]	0	R/W	背景色 G 信号 G: 8 位、无符号 (0 ~ 255[LSB])
15 ~ 8	GR3_BASE_B [7:0]	0	R/W	背景色 B 信号 B: 8 位、无符号 (0 ~ 255[LSB])
7 ~ 0	GR3_BASE_R [7:0]	0	R/W	背景色 R 信号 R: 8 位、无符号 (0 ~ 255[LSB])

【注】 在全部图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_P_VEN 位为“1”时更新此寄存器。

35.2.43 CLUT 表中断控制寄存器 (图形 (3)) (GR3_CLUT_INT)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR3 CLT_SEL
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR3_LINE[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 17	—	全 0	R	保留位 读写值都为“0”。
16	GR3_CLT_SEL	0	R/W	CLUT 的表选择信号 0: 选择 CLUT 表 0 参照 CLUT 表 0 并且展开为 αRGB8888。 CPU 能读写 CLUT 表 1。 1: 选择 CLUT 表 1 参照 CLUT 表 1 并且展开为 αRGB8888。 CPU 能读写 CLUT 表 0。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	GR3_LINE[10:0]	0	R/W	行中断设定 在行数和 GR3_LINE 值相同时输出中断信号。即使不使用图形 (3) 的功能, 此功能也起作用。

【注】 在全部图形 (3) 的寄存器更新控制寄存器 (GR3_UPDATE) 的 GR3_P_VEN 位为“1”时更新此寄存器。

35.2.44 状态监视寄存器 (图形 (3)) (GR3_MON)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR3 ARC_ST
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 16	—	全 0	R	保留位 读写值都为“0”。
15 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GR3_ARC_ST	0	R	矩形区域 α 混合处理的状态标志 0: 加法运算或者减法运算处理结束 (α 值为“0”或者“255”的状态) 1: 正在进行加法运算或者减法运算处理

35.3 使用方法

35.3.1 静噪

GR1_DISP_SEL[1:0] 位、GR2_DISP_SEL[1:0] 位和 GR3_DISP_SEL[1:0] 位的初始值全部为“0”。因此，在初始状态下，无论是图形 (1)、(2)、(3) 的全部图形区内还是区外都显示背景色。因为背景色的初始值为黑色，所以初始状态显示黑色静噪。

35.3.2 矩形区域 α 混合

如果在淡入或者淡出过程中更改 GR_ARC_MODE 位、GR_ARC_COEF[7:0] 位和 GR_ARC_RATE[7:0] 位，就能在中途更改 α 系数和帧率。

36. 视频显示控制器 4 (6) 输出控制部

36.1 输出控制功能

36.1.1 功能概要

输出控制部对图像合成部的 RGB 输出信号进行亮度调整、对比度调整、RGB 独立的灰度校正、抖动处理和输出格式转换，还生成用于驱动 LCD 显示屏的各种时序信号。

输出控制部的功能框图如下图所示。

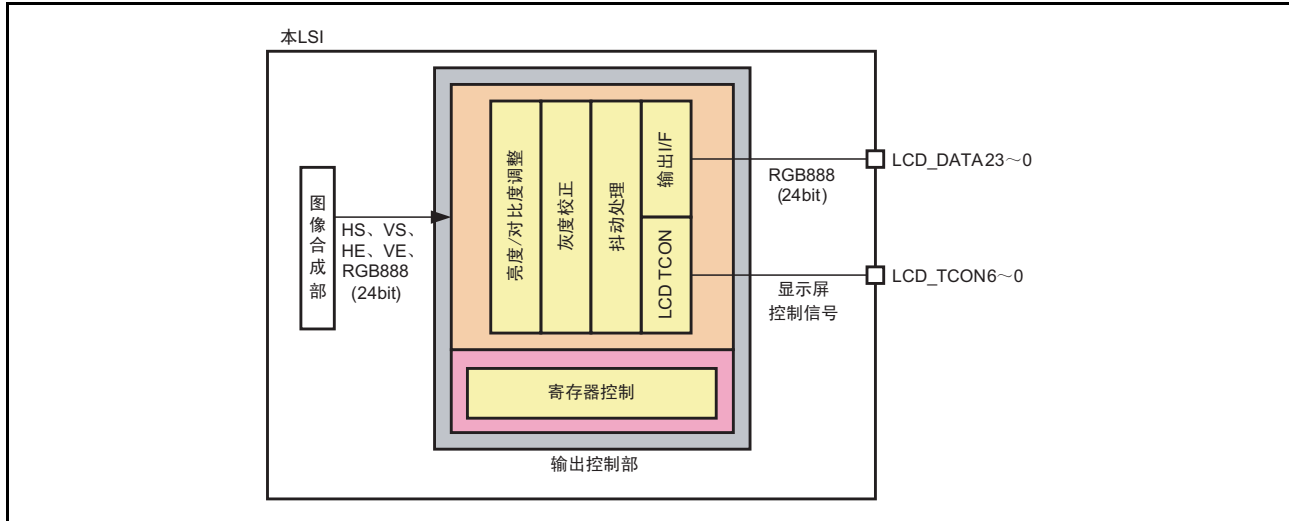


图 36.1 输出控制部的功能框图

36.1.2 寄存器的更新控制

输出控制部的控制寄存器全部通过垂直同步信号管理更新时序。

在将更新控制寄存器置“1”后，在垂直同步信号的上升沿反映各种寄存器。反映后，更新控制寄存器自动清“0”。

表 36.1 寄存器的更新控制

寄存器名	位名	初始值	说明
OUT_UPDATE	OUTCNT_VEN	0	亮度 / 对比度调整、抖动处理和输出接口的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器
GAM_G_UPDATE	GAM_G_VEN	0	灰度校正 (G) 的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器
GAM_B_UPDATE	GAM_B_VEN	0	灰度校正 (B) 的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器
GAM_R_UPDATE	GAM_R_VEN	0	灰度校正 (R) 的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器
TCON_UPDATE	TCON_VEN	0	LCD TCON 的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器

36.1.3 路径选择

能通过设定寄存器来切换亮度 / 对比度调整和灰度校正的处理顺序。

表 36.2 路径选择

寄存器名	位名	初始值	说明
OUT_CLK_PHASE	OUTCNT_FRONT_GAM	0	校正电路的顺序控制 0: 亮度 → 对比度 → 灰度校正 1: 灰度校正 → 亮度 → 对比度

36.1.4 亮度调整

对图像合成部的 RGB 信号进行 RGB 独立的亮度 (DC) 调整。

(为了使亮度调整后的 BRT_R/G/BOUT 不发生上溢或者下溢而准备了多个位。在对比度运算时进行上溢和下溢处理。)

(1) 亮度 (DC) 调整的计算式

$$\text{BRT_GOUT} = \text{GIN} + \text{PBRT_G} - 512$$

$$\text{BRT_BOU} = \text{BIN} + \text{PBRT_B} - 512$$

$$\text{BRT_ROU} = \text{RIN} + \text{PBRT_R} - 512$$

表 36.3 亮度 (DC) 调整

寄存器名	位名	初始值	说明
OUT_BRIGHT1	PBRT_G[9:0]	512	G 信号的亮度 (DC) 调整 无符号 (0 (-512) ~ 512 (0) ~ 1023 (+511) [LSB], 带偏移的 512[LSB])
OUT_BRIGHT2	PBRT_B[9:0]	512	B 信号的亮度 (DC) 调整 无符号 (0 (-512) ~ 512 (0) ~ 1023 (+511) [LSB], 带偏移的 512[LSB])
OUT_BRIGHT2	PBRT_R[9:0]	512	R 信号的亮度 (DC) 调整 无符号 (0 (-512) ~ 512 (0) ~ 1023 (+511) [LSB], 带偏移的 512[LSB])

36.1.5 对比度调整

对亮度运算结果的 RGB 信号进行对比度运算。

(如果发生上溢或者下溢, 就限幅为最大值或者最小值。)

(1) 对比度 (增益) 调整的计算式

$$GOUT = BRT_GOUT \times CONT_G \div 128$$

$$BOUT = BRT_BOUT \times CONT_B \div 128$$

$$ROUT = BRT_ROUT \times CONT_R \div 128$$

表 36.4 对比度 (增益) 调整

寄存器名	位名	初始值	说明
OUT_CONTRAST	CONT_G[7:0]	128	G 信号的对比度 (增益) 调整 0/128 ~ 255/128 (约 2 倍)
OUT_CONTRAST	CONT_B[7:0]	128	B 信号的对比度 (增益) 调整 0/128 ~ 255/128 (约 2 倍)
OUT_CONTRAST	CONT_R[7:0]	128	R 信号的对比度 (增益) 调整 0/128 ~ 255/128 (约 2 倍)

36.1.6 灰度校正

通过将 256 灰度等级的输入信号分成 32 级并且控制各区域的增益, 进行灰度校正。能将各区域的增益系数设定为 0 ~ 约 2.0[倍]。

(1) 各区域的灰度校正计算式

$$DOUT = ((DIN - TH_{(n)}) \times GAIN_{(n)} + OFFSET_{(n)}) \div 256$$

DIN : 输入信号 (8 位)
 DOUT : 输出信号 (10 位)
 TH_(n) : 阈值 (8 位)
 OFFSET_(n) : 偏移值 (19 位)
 GAIN_(n) : 增益系数 (11 位)

(2) 各区域的偏移计算式

$$OFFSET_{(n)} = OFFSET_{(n-1)} + DEF_O_{(n)} \quad \text{其中, 在 } n \text{ 为 "0" 时 } OFFSET_{(0)} = 0$$

$$DEF_O_{(n)} = (TH_{(n)} - TH_{(n-1)}) \times GAIN_{(n-1)} \quad \text{其中, 在 } n \text{ 为 "0" 时 } DEF_O_{(0)} = 0$$

OFFSET_(n) : 当前区域的偏移值 (19 位)
 OFFSET_(n-1) : 前一个区域的偏移值 (19 位)
 DEF_O_(n) : 当前区域和前一个区域的偏移值的差分 (19 位)
 TH_(n) : 当前区域的阈值 (8 位)
 TH_(n-1) : 前一个区域的阈值 (8 位)
 GAIN_(n-1) : 前一个区域的增益系数 (11 位)

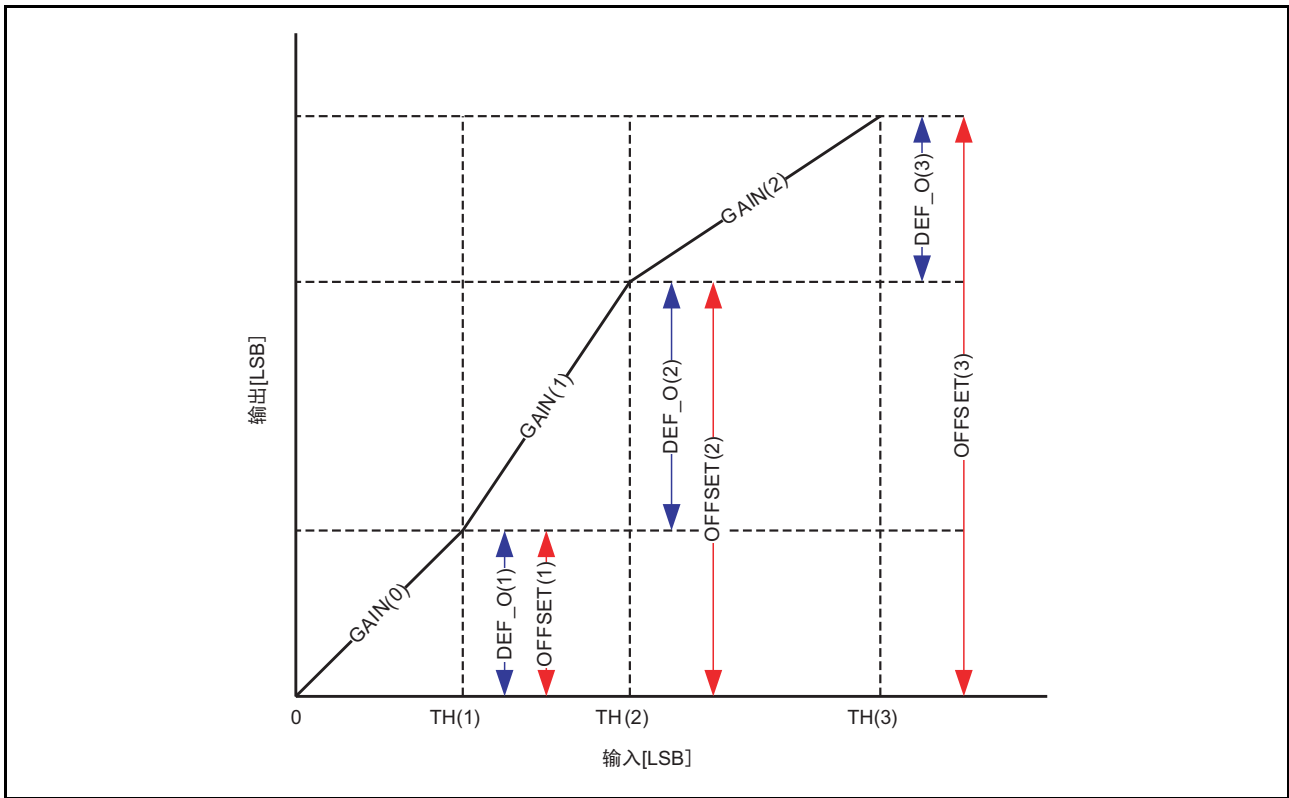


图 36.2 偏移计算式的对应图

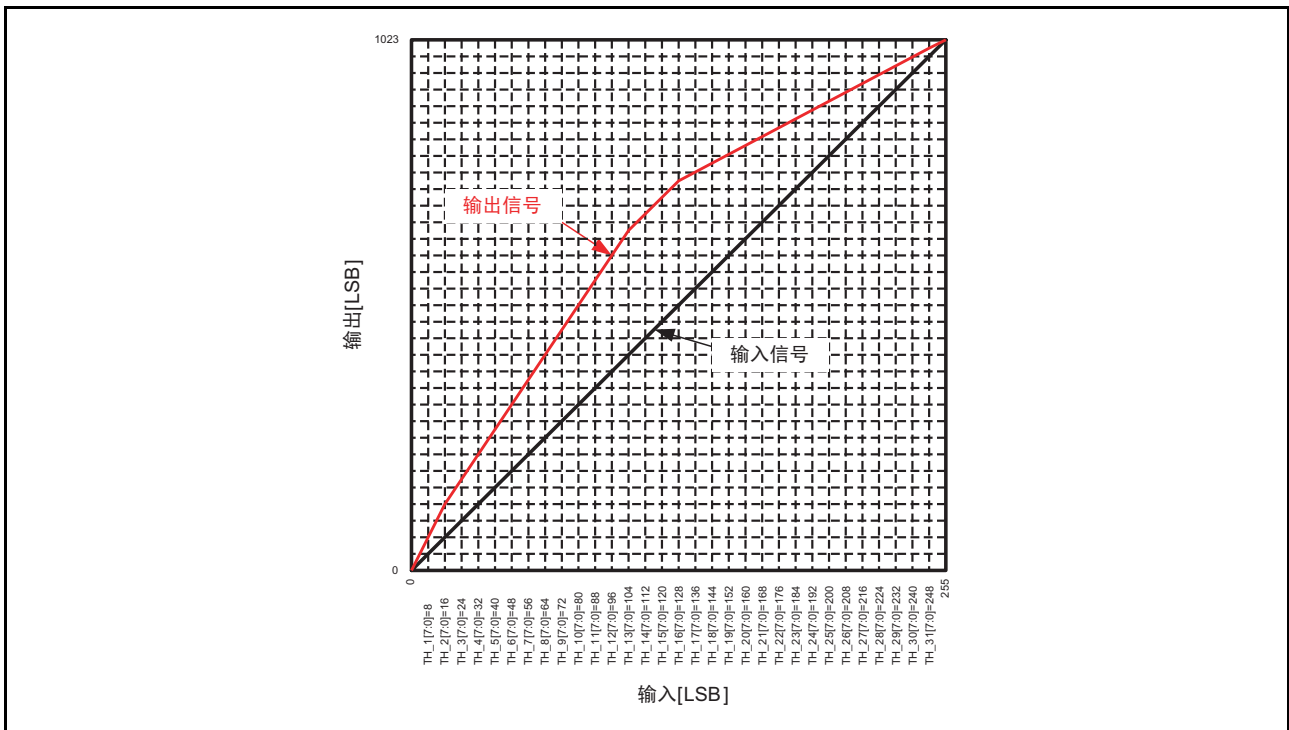


图 36.3 灰度校正的输入 / 输出特性 (例)

表 36.5 灰度校正

寄存器名	位名	初始值	说明
GAM_SW	GAM_ON	0	灰度校正的 ON/OFF 控制 0: OFF 1: ON
GAM_G_AREA1 ~ 8	GAM_G_TH_01 ~ 31 [7:0]	*	G 信号区域 1 ~ 31 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值*1 ≤ 当前区域的阈值 < 下一个区域的阈值*2 【注】 *1 GAM_G_TH_01 位为 “0”。 *2 GAM_G_TH_31 位为 “255”。 * 初始值 GAM_G_TH_01: 8, GAM_G_TH_02: 16, GAM_G_TH_03: 24, GAM_G_TH_04: 32, GAM_G_TH_05: 40, GAM_G_TH_06: 48, GAM_G_TH_07: 56, GAM_G_TH_08: 64, GAM_G_TH_09: 72, GAM_G_TH_10: 80, GAM_G_TH_11: 88, GAM_G_TH_12: 96, GAM_G_TH_13: 104, GAM_G_TH_14: 112, GAM_G_TH_15: 120, GAM_G_TH_16: 128, GAM_G_TH_17: 136, GAM_G_TH_18: 144, GAM_G_TH_19: 152, GAM_G_TH_20: 160, GAM_G_TH_21: 168, GAM_G_TH_22: 176, GAM_G_TH_23: 184, GAM_G_TH_24: 192, GAM_G_TH_25: 200, GAM_G_TH_26: 208, GAM_G_TH_27: 216, GAM_G_TH_28: 224, GAM_G_TH_29: 232, GAM_G_TH_30: 240, GAM_G_TH_31: 248
GAM_G_LUT1 ~ 16	GAM_G_GAIN_00 ~ 31 [10:0]	1024	G 信号区域 0 ~ 31 的增益调整 无符号 (0 ~ 2047[LSB], 1024[LSB]=1.0[倍])
GAM_B_AREA1 ~ 8	GAM_B_TH_01 ~ 31 [7:0]	*	B 信号区域 1 ~ 31 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值*1 ≤ 当前区域的阈值 < 下一个区域的阈值*2 【注】 *1 GAM_B_TH_01 位为 “0”。 *2 GAM_B_TH_31 位为 “255”。 * 初始值 GAM_B_TH_01: 8, GAM_B_TH_02: 16, GAM_B_TH_03: 24, GAM_B_TH_04: 32, GAM_B_TH_05: 40, GAM_B_TH_06: 48, GAM_B_TH_07: 56, GAM_B_TH_08: 64, GAM_B_TH_09: 72, GAM_B_TH_10: 80, GAM_B_TH_11: 88, GAM_B_TH_12: 96, GAM_B_TH_13: 104, GAM_B_TH_14: 112, GAM_B_TH_15: 120, GAM_B_TH_16: 128, GAM_B_TH_17: 136, GAM_B_TH_18: 144, GAM_B_TH_19: 152, GAM_B_TH_20: 160, GAM_B_TH_21: 168, GAM_B_TH_22: 176, GAM_B_TH_23: 184, GAM_B_TH_24: 192, GAM_B_TH_25: 200, GAM_B_TH_26: 208, GAM_B_TH_27: 216, GAM_B_TH_28: 224, GAM_B_TH_29: 232, GAM_B_TH_30: 240, GAM_B_TH_31: 248

寄存器名	位名	初始值	说明
GAM_B_LUT1 ~ 16	GAM_B_GAIN_00 ~ 31 [10:0]	1024	B 信号区域 0 ~ 31 的增益调整 无符号 (0 ~ 2047[LSB], 1024[LSB]=1.0[倍])
GAM_R_AREA1 ~ 8	GAM_R_TH_01 ~ 31 [7:0]	*	R 信号区域 1 ~ 31 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值*1 ≤ 当前区域的阈值 < 下一个区域的阈值*2 【注】 *1 GAM_R_TH_01 位为 “0”。 *2 GAM_R_TH_31 位为 “255”。 * 初始值 GAM_R_TH_01: 8, GAM_R_TH_02: 16, GAM_R_TH_03: 24, GAM_R_TH_04: 32, GAM_R_TH_05: 40, GAM_R_TH_06: 48, GAM_R_TH_07: 56, GAM_R_TH_08: 64, GAM_R_TH_09: 72, GAM_R_TH_10: 80, GAM_R_TH_11: 88, GAM_R_TH_12: 96, GAM_R_TH_13: 104, GAM_R_TH_14: 112, GAM_R_TH_15: 120, GAM_R_TH_16: 128, GAM_R_TH_17: 136, GAM_R_TH_18: 144, GAM_R_TH_19: 152, GAM_R_TH_20: 160, GAM_R_TH_21: 168, GAM_R_TH_22: 176, GAM_R_TH_23: 184, GAM_R_TH_24: 192, GAM_R_TH_25: 200, GAM_R_TH_26: 208, GAM_R_TH_27: 216, GAM_R_TH_28: 224, GAM_R_TH_29: 232, GAM_R_TH_30: 240, GAM_R_TH_31: 248
GAM_R_LUT1 ~ 16	GAM_R_GAIN_00 ~ 31 [10:0]	1024	R 信号区域 0 ~ 31 的增益调整 无符号 (0 ~ 2047[LSB], 1024[LSB]=1.0[倍])

36.1.7 抖动处理

抖动处理是将亮度 / 对比度调整或者灰度校正输出的 10 位 RGB 信号压缩为 8 位、6 位或者 5 位。抖动处理的运行模式可选择舍去模式、四舍五入模式、2×2 图案抖动模式和随机图案抖动模式共 4 种模式。

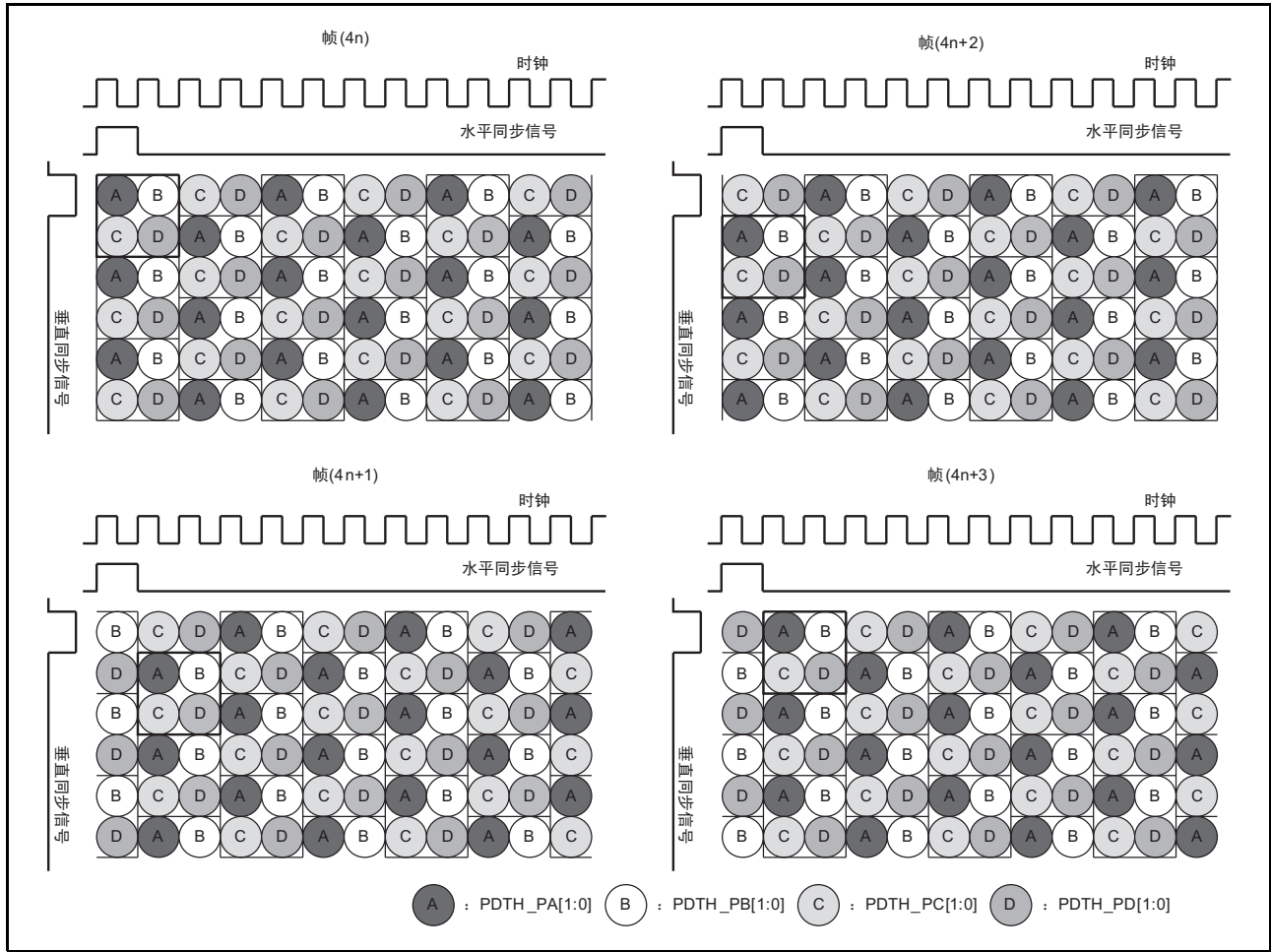


图 36.4 2×2 图案抖动的运行规格图

转换式如下所示:

- 舍去模式
 1. 10位→8位
输出RGB数据[7:0]=输入RGB数据[9:0]÷4 (舍去小数)
 2. 10位→6位
输出RGB数据[7:2]=输入RGB数据[9:0]÷16 (舍去小数)
 3. 10位→5位
输出RGB数据[7:3]=输入RGB数据[9:0]÷32 (舍去小数)
- 四舍五入模式
 1. 10位→8位
输出RGB数据[7:0]=输入RGB数据[9:0]÷4 (第一位小数四舍五入)
 2. 10位→6位
输出RGB数据[7:2]=输入RGB数据[9:0]÷16 (第一位小数四舍五入)
 3. 10位→5位
输出RGB数据[7:3]=输入RGB数据[9:0]÷32 (将第一位小数四舍五入)
- 2×2 图案抖动模式和随机图案抖动模式
 1. 10位→8位
输出RGB数据[7:0]=输入RGB数据[9:0]÷4+第一位小数加上图形值 (加法运算后舍去第一位小数)
 2. 10位→6位
输出RGB数据[7:2]=输入RGB数据[9:0]÷16+第一位小数加上图形值 (加法运算后舍去第一位小数)
 3. 10位→5位
输出RGB数据[7:3]=输入RGB数据[9:0]÷32+第一位小数加上图形值 (加法运算后舍去第一位小数)

表 36.6 显示屏抖动校正

寄存器名	位名	初始值	说明
OUT_PDTHA	PDTH_SEL[1:0]	0	显示屏抖动的运行模式 0: 舍去 1: 四舍五入 2: 2×2 图案抖动 3: 随机图案抖动
OUT_PDTHA	PDTH_FORMAT[1:0]	0	显示屏抖动的输出格式选择 0: RGB888 1: RGB666 2: RGB565 3: 禁止设定
OUT_PDTHA	PDTH_PA[1:0]	3	2×2 图案抖动的图案值 (A) 无符号 (0 ~ 3[LSB])
OUT_PDTHA	PDTH_PB[1:0]	0	2×2 图案抖动的图案值 (B) 无符号 (0 ~ 3[LSB])
OUT_PDTHA	PDTH_PC[1:0]	2	2×2 图案抖动的图案值 (C) 无符号 (0 ~ 3[LSB])
OUT_PDTHA	PDTH_PD[1:0]	1	2×2 图案抖动的图案值 (D) 无符号 (0 ~ 3[LSB])

36.1.8 输出格式的转换

输出格式的转换是将抖动处理后的 RGB 信号作为 LCD 输出信号转换为并行 RGB888、RGB666、RGB565 格式或者串行 RGB 格式

另外，能选择转换后的数据的 LCD 输出引脚分配。

(1) RGB888 输出时的 LCD 信号的位分配

RGB888 输出时分配到 LCD 信号输出的 RGB 信号输入如表 36.7 所示。

R/G/BIN[7:0] 是抖动处理后的 RGB 内部信号。

表 36.7 RGB888 输出时的 RGB 信号输入的位分配

OUT_FORMAT	0	0	0	0
OUT_ENDIAN_ON	0	0	1	1
OUT_SWAP_ON	0	1	0	1
LCD_DATA23	RIN[7]	BIN[7]	RIN[0]	BIN[0]
LCD_DATA22	RIN[6]	BIN[6]	RIN[1]	BIN[1]
LCD_DATA21	RIN[5]	BIN[5]	RIN[2]	BIN[2]
LCD_DATA20	RIN[4]	BIN[4]	RIN[3]	BIN[3]
LCD_DATA19	RIN[3]	BIN[3]	RIN[4]	BIN[4]
LCD_DATA18	RIN[2]	BIN[2]	RIN[5]	BIN[5]
LCD_DATA17	RIN[1]	BIN[1]	RIN[6]	BIN[6]
LCD_DATA16	RIN[0]	BIN[0]	RIN[7]	BIN[7]
LCD_DATA15	GIN[7]	GIN[7]	GIN[0]	GIN[0]
LCD_DATA14	GIN[6]	GIN[6]	GIN[1]	GIN[1]
LCD_DATA13	GIN[5]	GIN[5]	GIN[2]	GIN[2]
LCD_DATA12	GIN[4]	GIN[4]	GIN[3]	GIN[3]
LCD_DATA11	GIN[3]	GIN[3]	GIN[4]	GIN[4]
LCD_DATA10	GIN[2]	GIN[2]	GIN[5]	GIN[5]
LCD_DATA9	GIN[1]	GIN[1]	GIN[6]	GIN[6]
LCD_DATA8	GIN[0]	GIN[0]	GIN[7]	GIN[7]
LCD_DATA7	BIN[7]	RIN[7]	BIN[0]	RIN[0]
LCD_DATA6	BIN[6]	RIN[6]	BIN[1]	RIN[1]
LCD_DATA5	BIN[5]	RIN[5]	BIN[2]	RIN[2]
LCD_DATA4	BIN[4]	RIN[4]	BIN[3]	RIN[3]
LCD_DATA3	BIN[3]	RIN[3]	BIN[4]	RIN[4]
LCD_DATA2	BIN[2]	RIN[2]	BIN[5]	RIN[5]
LCD_DATA1	BIN[1]	RIN[1]	BIN[6]	RIN[6]
LCD_DATA0	BIN[0]	RIN[0]	BIN[7]	RIN[7]

(2) RGB666 输出时的 LCD 信号的位分配

RGB666 输出时分配到 LCD 信号输出的 RGB 信号输入如表 36.8 所示。

R/G/BIN[7:0] 是抖动处理后的 RGB 内部信号。

表 36.8 RGB666 输出时的 RGB 信号输入的位分配

OUT_FORMAT	1	1	1	1
OUT_ENDIAN_ON	0	0	1	1
OUT_SWAP_ON	0	1	0	1
LCD_DATA23	固定为“0”	固定为“0”	固定为“0”	固定为“0”
LCD_DATA22	固定为“0”	固定为“0”	固定为“0”	固定为“0”
LCD_DATA21	固定为“0”	固定为“0”	固定为“0”	固定为“0”
LCD_DATA20	固定为“0”	固定为“0”	固定为“0”	固定为“0”
LCD_DATA19	固定为“0”	固定为“0”	固定为“0”	固定为“0”
LCD_DATA18	固定为“0”	固定为“0”	固定为“0”	固定为“0”
LCD_DATA17	RIN[7]	BIN[7]	RIN[2]	BIN[2]
LCD_DATA16	RIN[6]	BIN[6]	RIN[3]	BIN[3]
LCD_DATA15	RIN[5]	BIN[5]	RIN[4]	BIN[4]
LCD_DATA14	RIN[4]	BIN[4]	RIN[5]	BIN[5]
LCD_DATA13	RIN[3]	BIN[3]	RIN[6]	BIN[6]
LCD_DATA12	RIN[2]	BIN[2]	RIN[7]	BIN[7]
LCD_DATA11	GIN[7]	GIN[7]	GIN[2]	GIN[2]
LCD_DATA10	GIN[6]	GIN[6]	GIN[3]	GIN[3]
LCD_DATA9	GIN[5]	GIN[5]	GIN[4]	GIN[4]
LCD_DATA8	GIN[4]	GIN[4]	GIN[5]	GIN[5]
LCD_DATA7	GIN[3]	GIN[3]	GIN[6]	GIN[6]
LCD_DATA6	GIN[2]	GIN[2]	GIN[7]	GIN[7]
LCD_DATA5	BIN[7]	RIN[7]	BIN[2]	RIN[2]
LCD_DATA4	BIN[6]	RIN[6]	BIN[3]	RIN[3]
LCD_DATA3	BIN[5]	RIN[5]	BIN[4]	RIN[4]
LCD_DATA2	BIN[4]	RIN[4]	BIN[5]	RIN[5]
LCD_DATA1	BIN[3]	RIN[3]	BIN[6]	RIN[6]
LCD_DATA0	BIN[2]	RIN[2]	BIN[7]	RIN[7]

(3) RGB565 输出时的 LCD 信号的位分配

RGB565 输出时分配到 LCD 信号输出的 RGB 信号输入如表 36.9 所示。

R/G/BIN[7:0] 是抖动处理后的 RGB 内部信号。

表 36.9 RGB565 输出时的 RGB 信号输入的位分配

OUT_FORMAT	2	2	2	2
OUT_ENDIAN_ON	0	0	1	1
OUT_SWAP_ON	0	1	0	1
LCD_DATA23	固定为“0”	固定为“0”	固定为“0”	固定为“0”
LCD_DATA22	固定为“0”	固定为“0”	固定为“0”	固定为“0”
LCD_DATA21	固定为“0”	固定为“0”	固定为“0”	固定为“0”
LCD_DATA20	固定为“0”	固定为“0”	固定为“0”	固定为“0”
LCD_DATA19	固定为“0”	固定为“0”	固定为“0”	固定为“0”
LCD_DATA18	固定为“0”	固定为“0”	固定为“0”	固定为“0”
LCD_DATA17	固定为“0”	固定为“0”	固定为“0”	固定为“0”
LCD_DATA16	固定为“0”	固定为“0”	固定为“0”	固定为“0”
LCD_DATA15	RIN[7]	BIN[7]	RIN[3]	BIN[3]
LCD_DATA14	RIN[6]	BIN[6]	RIN[4]	BIN[4]
LCD_DATA13	RIN[5]	BIN[5]	RIN[5]	BIN[5]
LCD_DATA12	RIN[4]	BIN[4]	RIN[6]	BIN[6]
LCD_DATA11	RIN[3]	BIN[3]	RIN[7]	BIN[7]
LCD_DATA10	GIN[7]	GIN[7]	GIN[2]	GIN[2]
LCD_DATA9	GIN[6]	GIN[6]	GIN[3]	GIN[3]
LCD_DATA8	GIN[5]	GIN[5]	GIN[4]	GIN[4]
LCD_DATA7	GIN[4]	GIN[4]	GIN[5]	GIN[5]
LCD_DATA6	GIN[3]	GIN[3]	GIN[6]	GIN[6]
LCD_DATA5	GIN[2]	GIN[2]	GIN[7]	GIN[7]
LCD_DATA4	BIN[7]	RIN[7]	BIN[3]	RIN[3]
LCD_DATA3	BIN[6]	RIN[6]	BIN[4]	RIN[4]
LCD_DATA2	BIN[5]	RIN[5]	BIN[5]	RIN[5]
LCD_DATA1	BIN[4]	RIN[4]	BIN[6]	RIN[6]
LCD_DATA0	BIN[3]	RIN[3]	BIN[7]	RIN[7]

(4) 串行 RGB 输出时的 LCD 信号的位分配

在串行 RGB 输出时，将表 36.10 所示的 RGB 信号输入分配到 rgb 内部信号，在将该信号进行并行到串行的转换后输出到 LCD 信号。R/G/BIN[7:0] 是抖动处理后的 RGB 内部信号。

将内部信号 r[7:0]、g[7:0]、b[7:0] 串行输出到 LCD_DATA7 ~ 0。

表 36.10 串行 RGB 输出时的 RGB 信号输入的位分配

OUT_FORMAT	3	3	3	3
OUT_ENDIAN_ON	0	0	1	1
OUT_SWAP_ON	0	1	0	1
r[7]	RIN[7]	BIN[7]	RIN[0]	BIN[0]
r[6]	RIN[6]	BIN[6]	RIN[1]	BIN[1]
r[5]	RIN[5]	BIN[5]	RIN[2]	BIN[2]
r[4]	RIN[4]	BIN[4]	RIN[3]	BIN[3]
r[3]	RIN[3]	BIN[3]	RIN[4]	BIN[4]
r[2]	RIN[2]	BIN[2]	RIN[5]	BIN[5]
r[1]	RIN[1]	BIN[1]	RIN[6]	BIN[6]
r[0]	RIN[0]	BIN[0]	RIN[7]	BIN[7]
g[7]	GIN[7]	GIN[7]	GIN[0]	GIN[0]
g[6]	GIN[6]	GIN[6]	GIN[1]	GIN[1]
g[5]	GIN[5]	GIN[5]	GIN[2]	GIN[2]
g[4]	GIN[4]	GIN[4]	GIN[3]	GIN[3]
g[3]	GIN[3]	GIN[3]	GIN[4]	GIN[4]
g[2]	GIN[2]	GIN[2]	GIN[5]	GIN[5]
g[1]	GIN[1]	GIN[1]	GIN[6]	GIN[6]
g[0]	GIN[0]	GIN[0]	GIN[7]	GIN[7]
b[7]	BIN[7]	RIN[7]	BIN[0]	RIN[0]
b[6]	BIN[6]	RIN[6]	BIN[1]	RIN[1]
b[5]	BIN[5]	RIN[5]	BIN[2]	RIN[2]
b[4]	BIN[4]	RIN[4]	BIN[3]	RIN[3]
b[3]	BIN[3]	RIN[3]	BIN[4]	RIN[4]
b[2]	BIN[2]	RIN[2]	BIN[5]	RIN[5]
b[1]	BIN[1]	RIN[1]	BIN[6]	RIN[6]
b[0]	BIN[0]	RIN[0]	BIN[7]	RIN[7]

(5) 并行到串行的转换

如表 36.11 所示，能通过控制倍速模式和选择扫描方向进行 4 种并行到串行的转换（表中的 n 是自然数）。

表 36.11 串行 RGB 输出的规格

OUT_FRQ_SEL	1	1	2	2
OUT_DIR_SEL	0	1	0	1
行 (2n-1)	重复 (r→g→b)	重复 (b→g→r)	重复 (r→g→b→X)	重复 (X→b→g→r)
行 2n	重复 (g→b→r)	重复 (r→b→g)	重复 (r→g→b→X)	重复 (X→b→g→r)

3 倍速和 4 倍速的并行到串行的转换时序图分别如图 36.5 和图 36.6 所示。

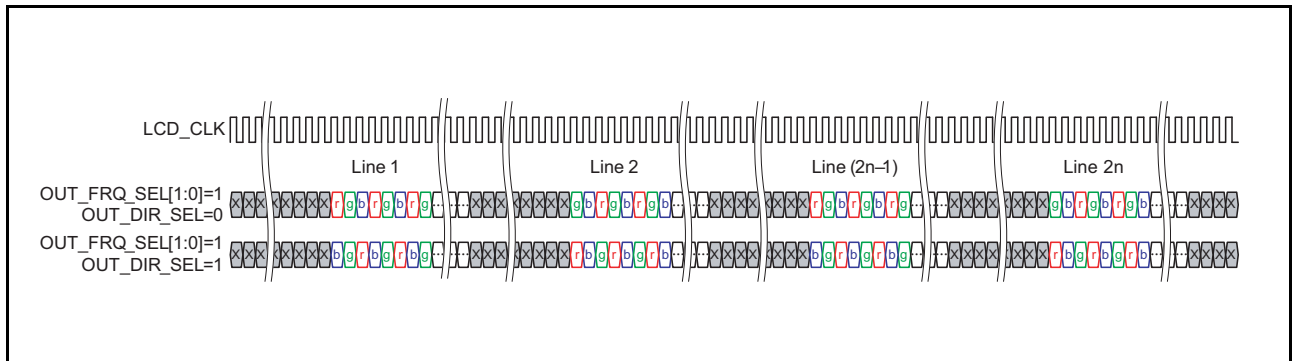


图 36.5 3 倍速并行到串行的转换时序图

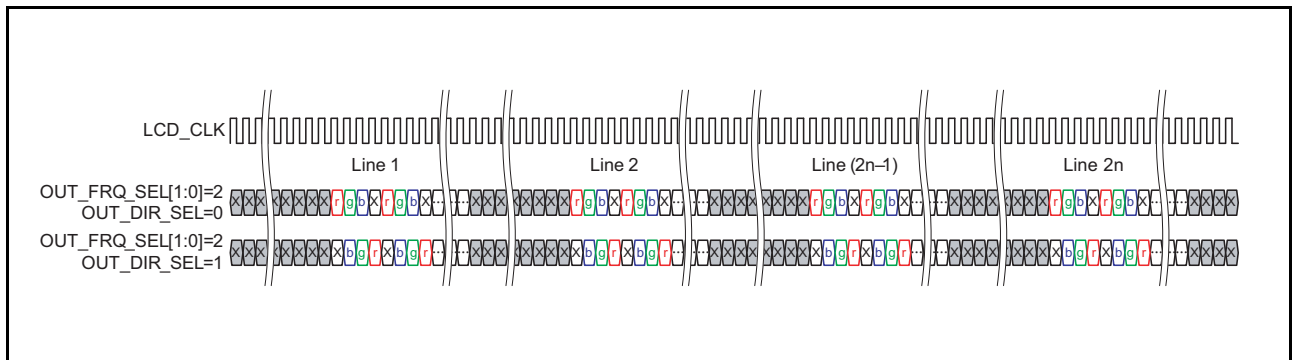


图 36.6 4 倍速并行到串行的转换时序图

能通过寄存器 OUT_PHASE[1:0] 调整相对于串行输出时的 HE 信号的相位时序。

串行 RGB 输出 (3 倍速模式) 的时钟相位时序图如图 36.7 所示。

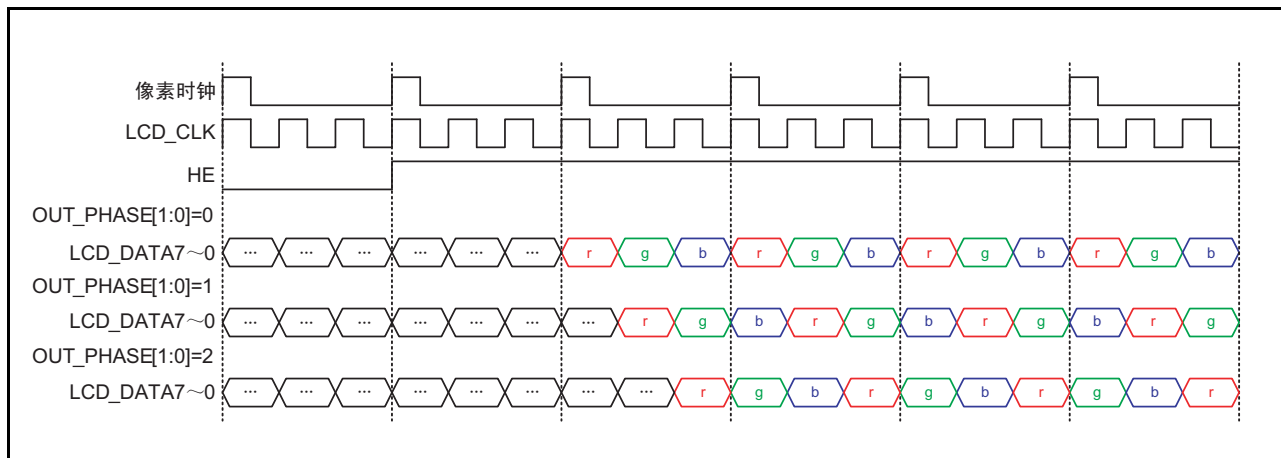


图 36.7 串行 RGB (3 倍速模式) 的时钟相位时序图

串行 RGB 输出 (4 倍速模式) 的时钟相位时序图如图 36.8 所示。

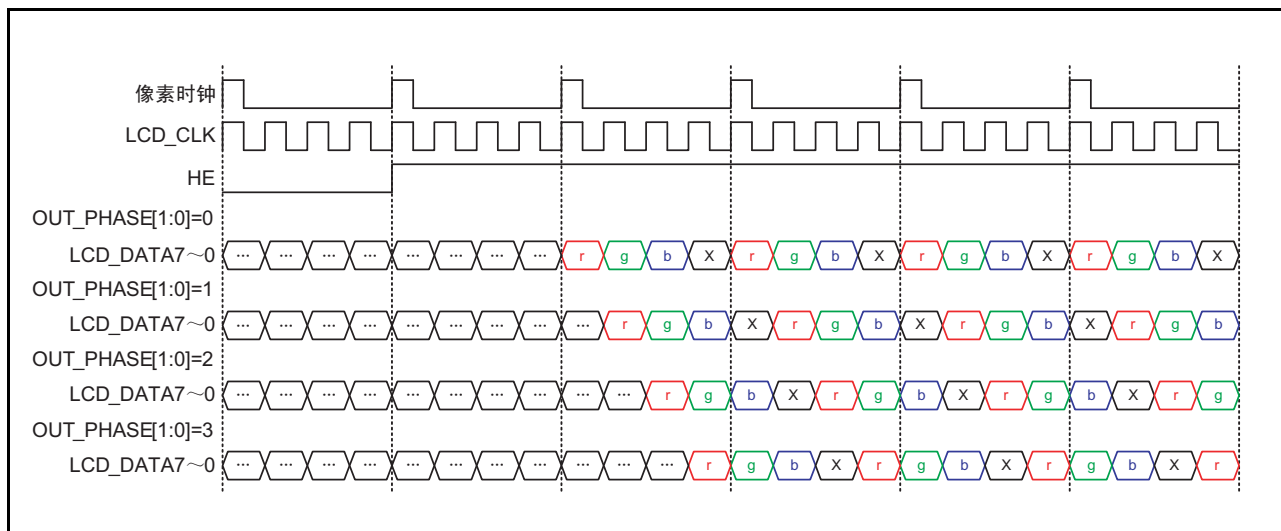


图 36.8 串行 RGB (4 倍速模式) 的时钟相位时序图

表 36.12 输出格式的转换

寄存器名	位名	初始值	说明
OUT_SET	OUT_FORMAT[1:0]	0	输出格式选择 0: RGB888 1: RGB666 2: RGB565 3: 串行 RGB
OUT_SET	OUT_ENDIAN_ON	0	位序变更 ON/OFF 控制 0: OFF 1: ON
OUT_SET	OUT_SWAP_ON	0	B/R 信号交换 ON/OFF 控制 0: OFF 1: ON
OUT_SET	OUT_FRQ_SEL[1:0]	0	时钟频率控制 0: 1 倍速 (并行 RGB) 1: 3 倍速 (串行 RGB) 2: 4 倍速 (串行 RGB) 3: 禁止设定
OUT_SET	OUT_DIR_SEL	0	扫描方向选择 0: 正向扫描 1: 反向扫描
OUT_SET	OUT_PHASE[1:0]	0	串行 RGB 输出时的时钟相位调整 3 倍速的情况 0: 0[clk] 1: 1[clk] 2: 2[clk] 3: 禁止设定 4 倍速的情况 0: 0[clk] 1: 1[clk] 2: 2[clk] 3: 3[clk]

36.1.9 LCD TCON

生成用于驱动 LCD 显示屏的各种时序信号。

能生成的时序信号是 2 个垂直显示屏驱动信号、5 个水平显示屏驱动信号、1 个垂直和水平显示屏驱动信号的合成信号。能生成的时序信号一览表如表 36.13 所示。

表 36.13 LCD TCON 生成信号

信号名	分类	说明
STVA/VS	垂直	<ul style="list-style-type: none"> 栅极启动信号 能控制脉宽、脉冲位置和脉冲极性。 垂直同步信号 能控制同步信号宽度、同步信号位置和同步信号极性。
STVB/VE	垂直	<ul style="list-style-type: none"> 栅极启动信号 能控制脉宽、脉冲位置和脉冲极性。 垂直允许信号 能控制同步信号宽度、同步信号位置和同步信号极性。
STH/SP/HS	水平	<ul style="list-style-type: none"> 源极启动信号 能控制脉宽、脉冲位置和脉冲极性。 水平同步信号 能控制同步信号宽度、同步信号位置和同步信号极性。
STB/LP/HE	水平	<ul style="list-style-type: none"> 源极选通信号 能控制脉宽、脉冲位置和脉冲极性 水平允许信号 能控制允许宽度、允许位置和允许极性。
CPV/GCK	水平	<ul style="list-style-type: none"> 栅极时钟信号 能控制脉宽、脉冲位置和脉冲极性。
POLA	水平	<ul style="list-style-type: none"> VCOM 电压极性控制信号 能控制极性反转位置和极性反转操作 (1×1、1×2、2×2)。
POLB	水平	<ul style="list-style-type: none"> VCOM 电压极性控制信号 能控制极性反转位置和极性反转操作 (1×1、1×2、2×2)。
DE	水平和垂直	<ul style="list-style-type: none"> 数据允许信号 能控制允许宽度、允许位置和允许极性。

(1) 水平基准偏移控制

以水平同步信号的上升沿为基准生成延迟了 $TCON_OFFSET[10:0]$ 个时钟的基准信号。如果需要跨过水平同步信号而生成信号，就以此偏移的基准信号为基准生成信号。

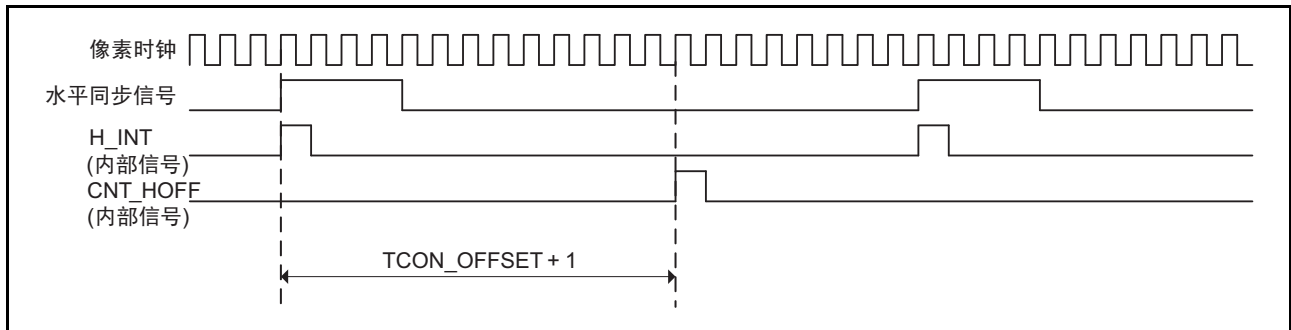


图 36.9 带偏移的水平基准 (H_OFF) 信号的生成

表 36.14 水平基准信号的选择表

寄存器名	位名	初始值	说明
TCON_TIM	TCON_OFFSET[10:0]	0	带偏移的水平同步信号的时序设定 设定从水平同步信号上升沿开始的时钟个数。
TCON_TIM_STH2	TCON_STH_HS_SEL	0	STH 信号的运行基准选择 0: 水平同步信号基准 1: 偏移后的水平同步信号基准
TCON_TIM_STB2	TCON_STB_HS_SEL	0	STB 信号的运行基准选择 0: 水平同步信号基准 1: 偏移后的水平同步信号基准
TCON_TIM_CPV2	TCON_CPV_HS_SEL	0	CPV 信号的运行基准选择 0: 水平同步信号基准 1: 偏移后的水平同步信号基准
TCON_TIM_POLA2	TCON_POLA_HS_SEL	0	POLA 信号的运行基准选择 0: 水平同步信号基准 1: 偏移后的水平同步信号基准
TCON_TIM_POLB2	TCON_POLB_HS_SEL	0	POLB 信号的运行基准选择 0: 水平同步信号基准 1: 偏移后的水平同步信号基准

【注】 在反向模式中生成 POLA 信号和 POLB 信号时，必须将 TCON_POLA_HS_SEL 位和 TCON_POLB_HS_SEL 位都置“0”。

(2) 水平显示屏驱动信号的生成 (A)

在生成水平同步显示屏驱动信号时, 通过设定第 1 个变化时序的 TCON_xxxx_HS[10:0] 位和设定第 2 个变化时序的 TCON_xxxx_HW[10:0] 位, 生成水平周期内变化 2 次的时序信号。

内部计数器的运行如下:

1. 以水平同步信号的上升沿为基准进行计数器的复位。
2. 以显示屏时钟的上升沿为基准进行递增计数。

也能通过将设定第 2 个变化时序的 TCON_xxxx_HW[10:0] 位置 “0”, 输出固定值 “0”。

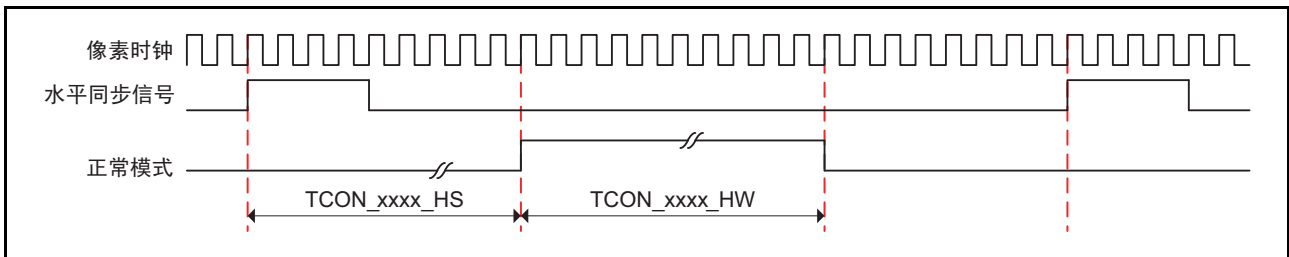


图 36.10 水平显示屏驱动信号 (正常模式)

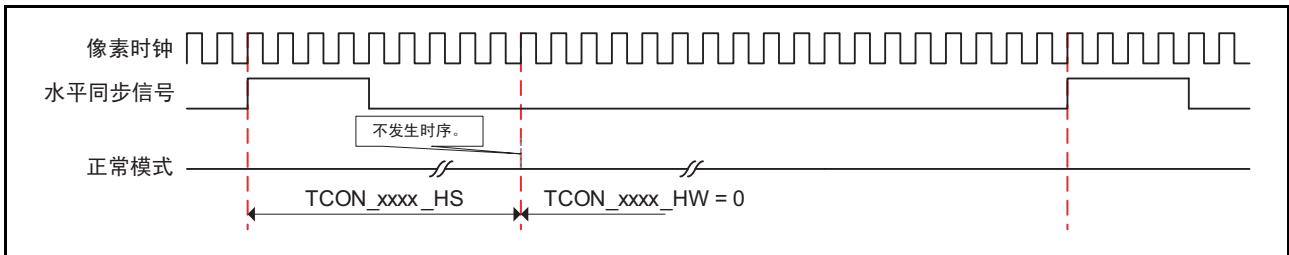


图 36.11 水平显示屏驱动信号 (正常模式, TCON_xxxx_HW 位为 “0”)

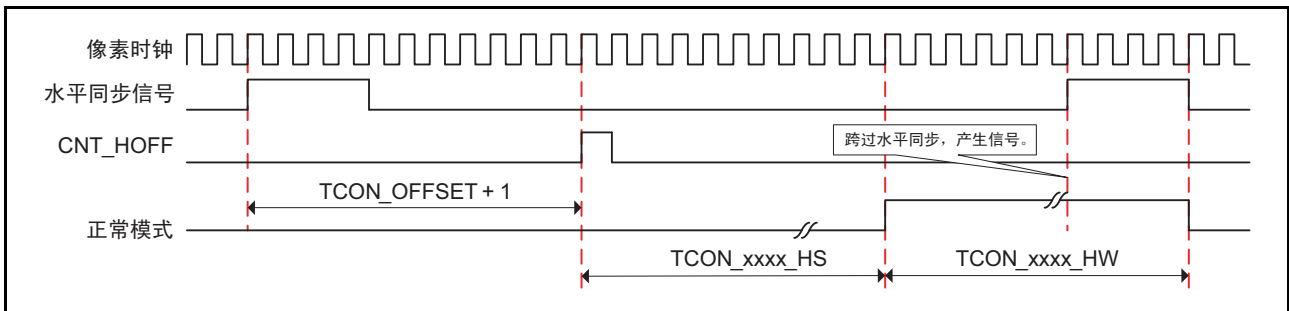


图 36.12 水平显示屏驱动信号 (正常模式, 使用带偏移的水平基准)

表 36.15 水平显示屏驱动信号 (A) 的生成设定

寄存器名	位名	初始值	说明
TCON_TIM_STH1	TCON_STH_HS[10:0]	0	设定 STH 信号的脉冲起始位置 (第 1 个变化时序)。在从水平同步信号的上升沿经过 TCON_STH_HS 后开始输出脉冲 (时钟周期)。
TCON_TIM_STH1	TCON_STH_HW[10:0]	96	设定 STH 信号的脉宽 (第 2 个变化时序)。输出 TCON_STH_HW 期间的脉冲 (时钟周期)。
TCON_TIM_STB1	TCON_STB_HS[10:0]	144	设定 STB 信号的脉冲起始位置 (第 1 个变化时序)。在从水平同步信号的上升沿经过 TCON_STB_HS 后开始输出脉冲 (时钟周期)。
TCON_TIM_STB1	TCON_STB_HW[10:0]	640	设定 STB 信号的脉宽 (第 2 个变化时序)。输出 TCON_STB_HW 期间的脉冲 (时钟周期)。
TCON_TIM_CPV1	TCON_CPV_HS[10:0]	0	设定 CPV 信号的脉冲起始位置 (第 1 个变化时序)。在从水平同步信号的上升沿经过 TCON_CPV_HS 后开始输出脉冲 (时钟周期)。
TCON_TIM_CPV1	TCON_CPV_HW[10:0]	0	设定 CPV 信号的脉宽 (第 2 个变化时序)。输出 TCON_CPV_HW 期间的脉冲 (时钟周期)。

(3) 水平显示屏驱动信号的生成 (B)

除 (2) 的正常模式运行以外, 还能以垂直同步信号的上升沿为基准开始运行, 通过设定第 1 个变化时序的寄存器 TCON_xxxx_HS[10:0] 的时序, 在每个水平周期生成极性相反的信号。

在反向模式中, 与垂直的行数 (偶数、奇数) 无关, 总是在每个垂直周期生成极性相反的信号。作为极性的反转操作, 可选择以下 3 种模式。

表 36.16 水平显示屏驱动信号的生成模式表

寄存器名	位名	初始值	说明
TCON_TIM_POLA2	TCON_POLA_MD[1:0]	1	POLA 信号的生成模式选择 0: 正常模式 在水平周期生成变化 2 次的信号。 1: 1×1 反向模式 在每个水平周期生成极性相反的信号。 2: 1×2 反向模式 在第 1 个水平期间将极性反转, 然后在每 2 个水平周期生成极性相反的信号。 3: 2×2 反向模式 在每 2 个水平周期生成极性相反的信号。
TCON_TIM_POLB2	TCON_POLB_MD[1:0]	1	POLB 信号的生成模式选择 0: 正常模式 在水平周期生成变化 2 次的信号。 1: 1×1 反向模式 在每个水平周期生成极性相反的信号。 2: 1×2 反向模式 在第 1 个水平期间将极性反转, 然后在每 2 个水平周期生成极性相反的信号。 3: 2×2 反向模式 在每 2 个水平周期生成极性相反的信号。

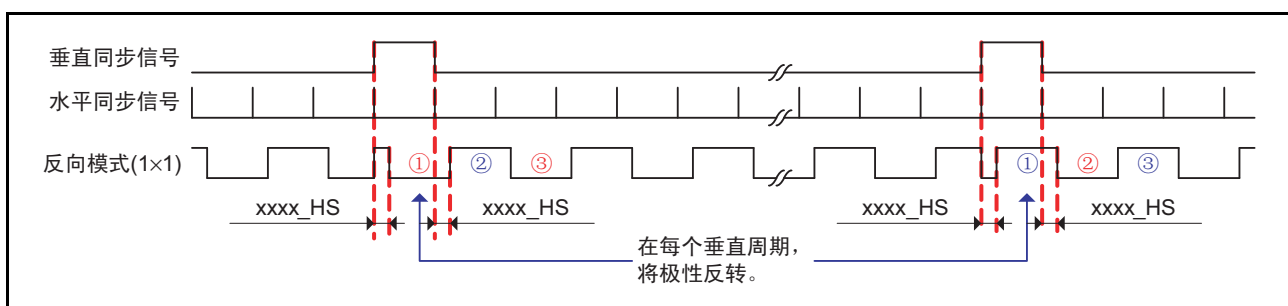


图 36.13 水平显示屏驱动信号 (反向模式、1×1)

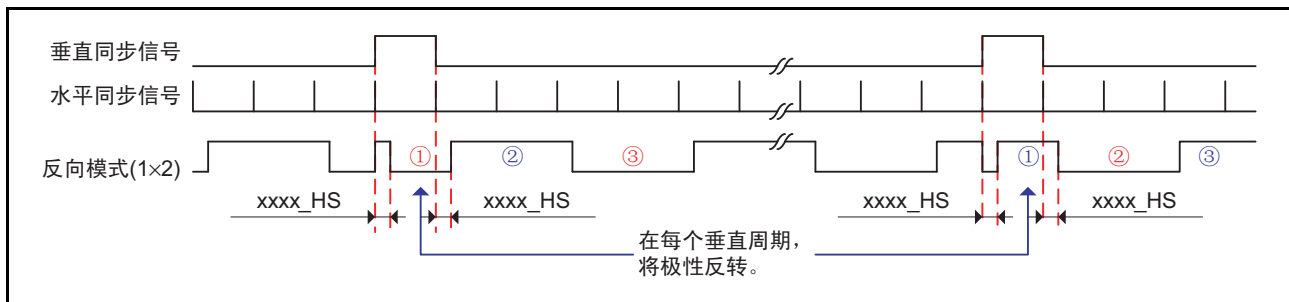


图 36.14 水平显示屏驱动信号 (反向模式、1×2)

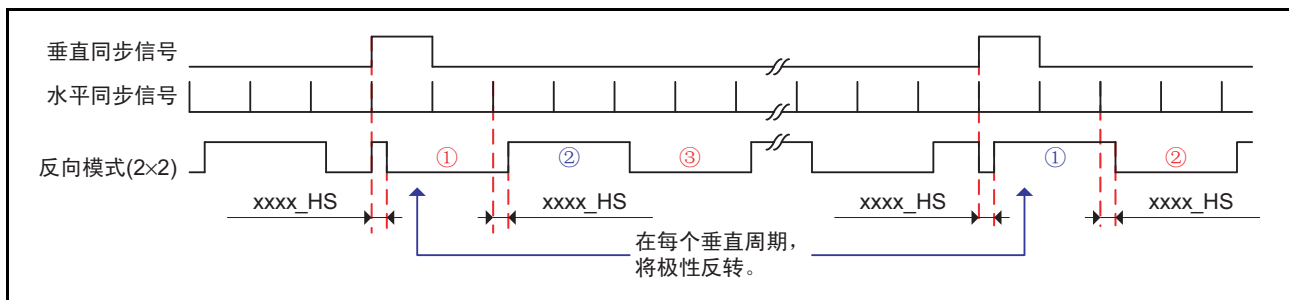


图 36.15 水平显示屏驱动信号 (反向模式、2×2)

表 36.17 水平显示屏驱动信号 (B) 的生成设定

寄存器名	位名	初始值	说明
TCON_TIM_POLA1	TCON_POLA_HS[10:0]	0	设定 POLA 信号的脉冲起始位置 (第 1 个变化时序)。在从水平同步信号的上升沿经过 TCON_POLA_HS 后开始输出脉冲 (时钟周期)。【注】 在设定 1×1、1×2、2×2 反向模式时, 必须设定大于等于 1 的值。
TCON_TIM_POLA1	TCON_POLA_HW[10:0]	0	设定 POLA 信号的脉宽 (第 2 个变化时序)。输出 TCON_POLA_HW 期间的脉冲 (时钟周期)。
TCON_TIM_POLB1	TCON_POLB_HS[10:0]	0	设定 POLB 信号的脉冲起始位置 (第 1 个变化时序)。在从水平同步信号的上升沿经过 TCON_POLB_HS 后开始输出脉冲 (时钟周期)。【注】 在设定 1×1、1×2、2×2 反向模式时, 必须设定大于等于 1 的值。
TCON_TIM_POLB1	TCON_POLB_HW[10:0]	0	设定 POLB 信号的脉宽 (第 2 个变化时序)。输出 TCON_POLB_HW 期间的脉冲 (时钟周期)。

(4) 垂直显示屏驱动信号的生成

垂直同步显示屏驱动信号的生成进行以下运行：

1. 在垂直同步信号的上升沿进行初始化。
2. 通过内部计数器、设定第1个变化时序的 TCON_xxxx_VS[10:0] 以及设定第2个变化时序的 TCON_xxxx_VW[10:0]，在垂直周期生成变化2次的时序信号。

内部计数器在以下情况下进行递增计数：

1. 内部计数器以水平同步信号的上升沿为基准。
2. 以水平同步信号的上升沿为基准并且通过 TCON_HALF[10:0] 的设定进行了时钟延迟的基准（通常设定 $1/2fH$ ）。

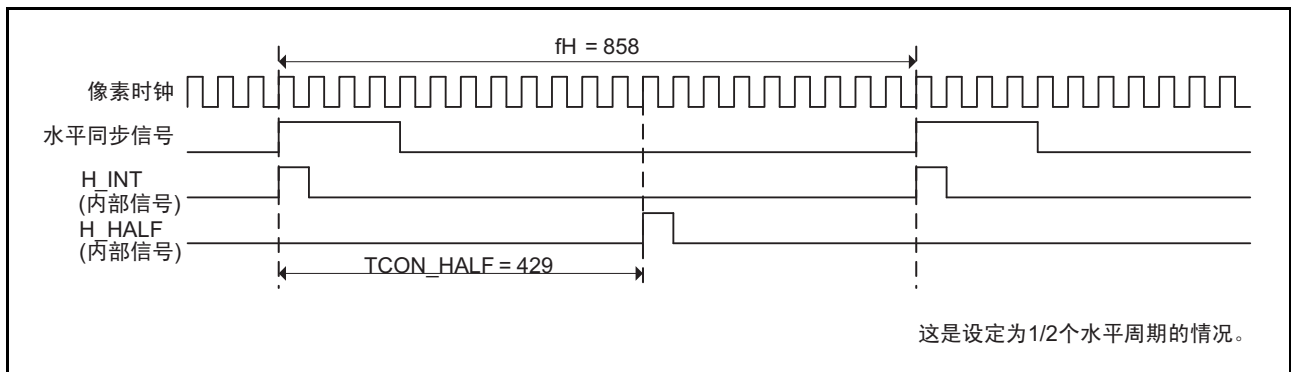


图 36.16 1/2 脉冲 (H_HALF) 信号的生成

表 36.18 1/2 脉冲 (H_HALF) 信号的生成设定

寄存器名	位名	初始值	说明
TCON_TIM	TCON_HALF[10:0]	400	1/2fH 时序设定 将从水平同步信号的上升沿开始的时钟个数指定为垂直计数器的计数运行时序。

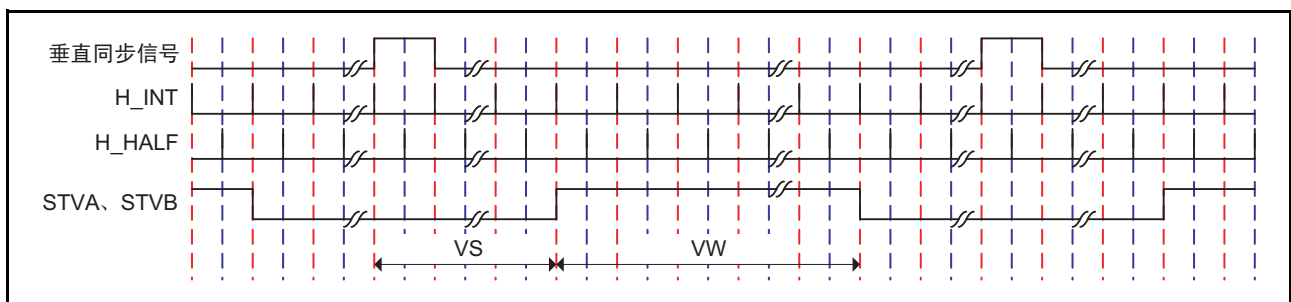


图 36.17 垂直显示屏驱动信号 (H_INT 标准运行)

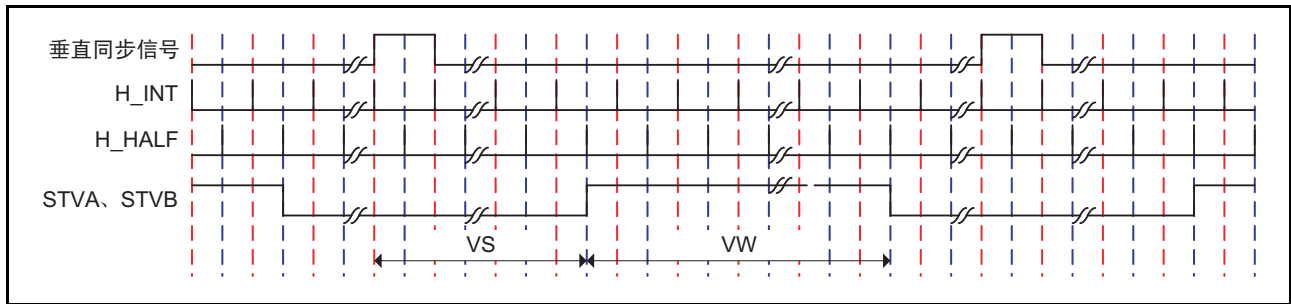


图 36.18 垂直显示屏驱动信号 (H_HALF 标准运行)

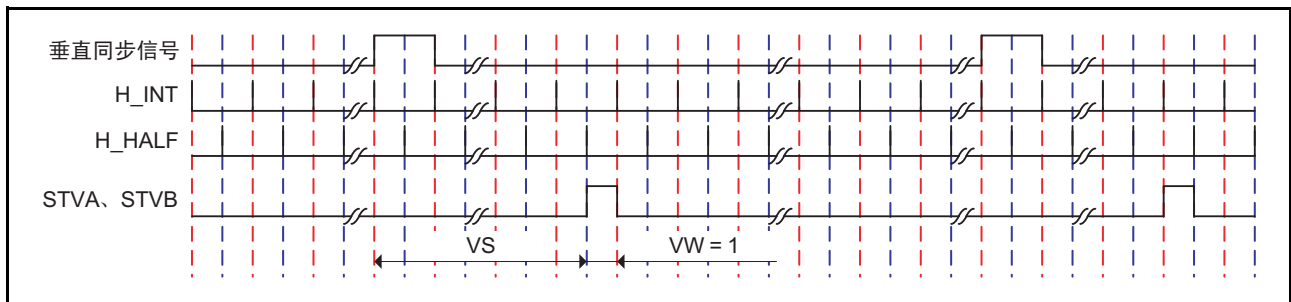


图 36.19 垂直显示屏驱动信号 (H_INT、H_HALF 标准运行)

表 36.19 垂直显示屏驱动信号的生成

寄存器名	位名	初始值	说明
TCON_TIM_STVA1	TCON_STVA_VS[10:0]	0	设定 STVA 信号的脉冲起始位置 (第 1 个变化时序)。在从垂直同步信号的上升沿经过 TCON_STVA_VS 后开始输出脉冲 (1/2fH 周期)。
TCON_TIM_STVA1	TCON_STVA_VW[10:0]	4	设定 STVA 信号的脉宽 (第 2 个变化时序)。输出 TCON_STVA_VW 期间的脉冲 (1/2fH 周期)。
TCON_TIM_STVB1	TCON_STVB_VS[10:0]	70	设定 STVB 信号的脉冲起始位置 (第 1 个变化时序)。在从垂直同步信号的上升沿经过 TCON_STVB_VS 后开始输出脉冲 (1/2fH 周期)。
TCON_TIM_STVB1	TCON_STVB_VW[10:0]	960	设定 STVB 信号的脉宽 (第 2 个变化时序)。输出 TCON_STVB_VW 期间的脉冲 (1/2fH 周期)。

(5) DE 时序信号的生成

在合成水平显示屏驱动 (HE) 信号和垂直显示屏驱动 (VE) 信号 (逻辑与) 后生成表示图像信号有效期间的数据允许 (DE) 信号。

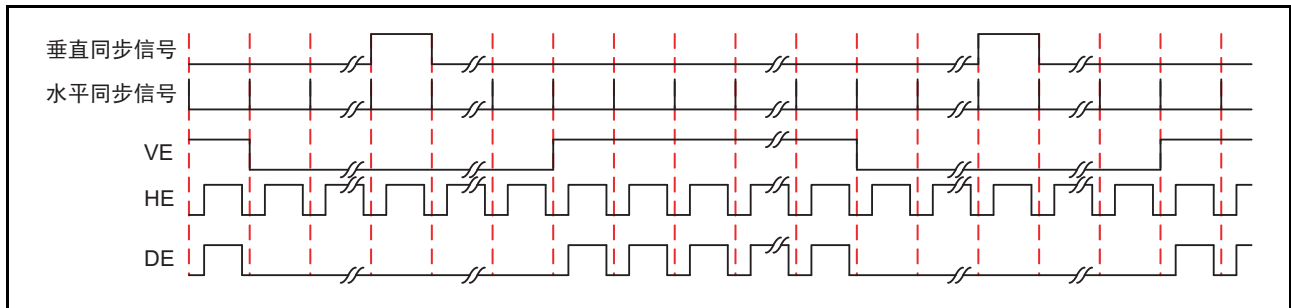


图 36.20 数据允许信号的生成

(6) 极性反转

能将通过信号生成电路生成的各信号的极性进行反转。

表 36.20 显示屏驱动信号的极性反转控制

寄存器名	位名	初始值	说明
TCON_TIM_STVA2	TCON_STVA_INV	1	STVA 信号的极性反转控制 0: 不反转 1: 反转
TCON_TIM_STVB2	TCON_STVB_INV	0	STVB 信号的极性反转控制 0: 不反转 1: 反转
TCON_TIM_STH2	TCON_STH_INV	1	STH 信号的极性反转控制 0: 不反转 1: 反转
TCON_TIM_STB2	TCON_STB_INV	0	STB 信号的极性反转控制 0: 不反转 1: 反转
TCON_TIM_CPV2	TCON_CPV_INV	0	CPV 信号的极性反转控制 0: 不反转 1: 反转
TCON_TIM_POLA2	TCON_POLA_INV	0	POLA 信号的极性反转控制 0: 不反转 1: 反转
TCON_TIM_POLB2	TCON_POLB_INV	0	POLB 信号的极性反转控制 0: 不反转 1: 反转
TCON_TIM_DE	TCON_DE_INV	0	DE 信号的极性反转控制 0: 不反转 1: 反转

(7) 输出选择

选择被极性反转控制的各信号的输出引脚分配。

表 36.21 显示屏驱动信号的输出选择

寄存器名	位名	初始值	说明
TCON_TIM_STVA2	TCON_STVA_SEL[2:0]	0	LCD_TCON0 引脚的输出信号选择 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE
TCON_TIM_STVB2	TCON_STVB_SEL[2:0]	1	LCD_TCON1 引脚的输出信号选择 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE
TCON_TIM_STH2	TCON_STH_SEL[2:0]	2	LCD_TCON2 引脚的输出信号选择 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE
TCON_TIM_STB2	TCON_STB_SEL[2:0]	7	LCD_TCON3 引脚的输出信号选择 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

寄存器名	位名	初始值	说明
TCON_TIM_CPV2	TCON_CPV_SEL[2:0]	4	LCD_TCON4 引脚的输出信号选择 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE
TCON_TIM_POLA2	TCON_POLA_SEL[2:0]	5	LCD_TCON5 引脚的输出信号选择 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE
TCON_TIM_POLB2	TCON_POLB_SEL[2:0]	6	LCD_TCON6 引脚的输出信号选择 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

(8) 输出相位的选择

对于图像输出信号和各种时序输出信号，能以 LCD_CLK（显示屏时钟）为基准个别选择输出相位。

表 36.22 显示屏输出信号的相位选择

寄存器名	位名	初始值	说明
OUT_CLK_PHASE	OUTCNT_LCD_EDGE	0	LCD_DATA23 ~ 0 引脚的输出相位控制 0: 在 LCD_CLK 引脚的上升沿输出 1: 在 LCD_CLK 引脚的下降沿输出
OUT_CLK_PHASE	OUTCNT_STVA_EDGE	0	LCD_TCON0 引脚的输出相位控制 0: 在 LCD_CLK 引脚的上升沿输出 1: 在 LCD_CLK 引脚的下降沿输出
OUT_CLK_PHASE	OUTCNT_STVB_EDGE	0	LCD_TCON1 引脚的输出相位控制 0: 在 LCD_CLK 引脚的上升沿输出 1: 在 LCD_CLK 引脚的下降沿输出
OUT_CLK_PHASE	OUTCNT_STH_EDGE	0	LCD_TCON2 引脚的输出相位控制 0: 在 LCD_CLK 引脚的上升沿输出 1: 在 LCD_CLK 引脚的下降沿输出
OUT_CLK_PHASE	OUTCNT_STB_EDGE	0	LCD_TCON3 引脚的输出相位控制 0: 在 LCD_CLK 引脚的上升沿输出 1: 在 LCD_CLK 引脚的下降沿输出
OUT_CLK_PHASE	OUTCNT_CPV_EDGE	0	LCD_TCON4 引脚的输出相位控制 0: 在 LCD_CLK 引脚的上升沿输出 1: 在 LCD_CLK 引脚的下降沿输出
OUT_CLK_PHASE	OUTCNT_POLA_EDGE	0	LCD_TCON5 引脚的输出相位控制 0: 在 LCD_CLK 引脚的上升沿输出 1: 在 LCD_CLK 引脚的下降沿输出
OUT_CLK_PHASE	OUTCNT_POLB_EDGE	0	LCD_TCON6 引脚的输出相位控制 0: 在 LCD_CLK 引脚的上升沿输出 1: 在 LCD_CLK 引脚的下降沿输出

36.2 寄存器说明

寄存器结构如表 36.23 所示。

【说明寄存器时的符号】

初始值 : 复位后的寄存器值

— : 不定值

R/W : 可读写。能读到写入值。

R/WC0 : 可读写。如果写“0”，该位就被初始化，但是忽视写“1”。

R/WC1 : 可读写。如果写“1”，该位就被初始化，但是忽视写“0”。

R : 只能读。写入值总是“0”。

—/W : 只能写。读取值为不定值。

表 36.23 灰度校正部的寄存器结构

名称	略称	R/W	初始值	地址	存取长度
灰度校正部的寄存器更新控制寄存器 G	GAM_G_UPDATE	R/WC1	H'0000 0000	H'FFFF 7800	32/16
灰度校正部的功能切换寄存器	GAM_SW	R/W	H'0000 0000	H'FFFF 7804	32/16
灰度校正部的表设定寄存器 G1	GAM_G_LUT1	R/W	H'0400 0400	H'FFFF 7808	32/16
灰度校正部的表设定寄存器 G2	GAM_G_LUT2	R/W	H'0400 0400	H'FFFF 780C	32/16
灰度校正部的表设定寄存器 G3	GAM_G_LUT3	R/W	H'0400 0400	H'FFFF 7810	32/16
灰度校正部的表设定寄存器 G4	GAM_G_LUT4	R/W	H'0400 0400	H'FFFF 7814	32/16
灰度校正部的表设定寄存器 G5	GAM_G_LUT5	R/W	H'0400 0400	H'FFFF 7818	32/16
灰度校正部的表设定寄存器 G6	GAM_G_LUT6	R/W	H'0400 0400	H'FFFF 781C	32/16
灰度校正部的表设定寄存器 G7	GAM_G_LUT7	R/W	H'0400 0400	H'FFFF 7820	32/16
灰度校正部的表设定寄存器 G8	GAM_G_LUT8	R/W	H'0400 0400	H'FFFF 7824	32/16
灰度校正部的表设定寄存器 G9	GAM_G_LUT9	R/W	H'0400 0400	H'FFFF 7828	32/16
灰度校正部的表设定寄存器 G10	GAM_G_LUT10	R/W	H'0400 0400	H'FFFF 782C	32/16
灰度校正部的表设定寄存器 G11	GAM_G_LUT11	R/W	H'0400 0400	H'FFFF 7830	32/16
灰度校正部的表设定寄存器 G12	GAM_G_LUT12	R/W	H'0400 0400	H'FFFF 7834	32/16
灰度校正部的表设定寄存器 G13	GAM_G_LUT13	R/W	H'0400 0400	H'FFFF 7838	32/16
灰度校正部的表设定寄存器 G14	GAM_G_LUT14	R/W	H'0400 0400	H'FFFF 783C	32/16
灰度校正部的表设定寄存器 G15	GAM_G_LUT15	R/W	H'0400 0400	H'FFFF 7840	32/16
灰度校正部的表设定寄存器 G16	GAM_G_LUT16	R/W	H'0400 0400	H'FFFF 7844	32/16
灰度校正部的区域设定寄存器 G1	GAM_G_AREA1	R/W	H'0008 1018	H'FFFF 7848	32/16
灰度校正部的区域设定寄存器 G2	GAM_G_AREA2	R/W	H'2028 3038	H'FFFF 784C	32/16
灰度校正部的区域设定寄存器 G3	GAM_G_AREA3	R/W	H'4048 5058	H'FFFF 7850	32/16
灰度校正部的区域设定寄存器 G4	GAM_G_AREA4	R/W	H'6068 7078	H'FFFF 7854	32/16
灰度校正部的区域设定寄存器 G5	GAM_G_AREA5	R/W	H'8088 9098	H'FFFF 7858	32/16
灰度校正部的区域设定寄存器 G6	GAM_G_AREA6	R/W	H'A0A8 B0B8	H'FFFF 785C	32/16
灰度校正部的区域设定寄存器 G7	GAM_G_AREA7	R/W	H'C0C8 D0D8	H'FFFF 7860	32/16
灰度校正部的区域设定寄存器 G8	GAM_G_AREA8	R/W	H'E0E8 F0F8	H'FFFF 7864	32/16
灰度校正部的寄存器更新控制寄存器 B	GAM_B_UPDATE	R/WC1	H'0000 0000	H'FFFF 7880	32/16
灰度校正部的表设定寄存器 B1	GAM_B_LUT1	R/W	H'0400 0400	H'FFFF 7888	32/16
灰度校正部的表设定寄存器 B2	GAM_B_LUT2	R/W	H'0400 0400	H'FFFF 788C	32/16
灰度校正部的表设定寄存器 B3	GAM_B_LUT3	R/W	H'0400 0400	H'FFFF 7890	32/16

名称	略称	R/W	初始值	地址	存取长度
灰度校正部的表设定寄存器 B4	GAM_B_LUT4	R/W	H'0400 0400	H'FFFF 7894	32/16
灰度校正部的表设定寄存器 B5	GAM_B_LUT5	R/W	H'0400 0400	H'FFFF 7898	32/16
灰度校正部的表设定寄存器 B6	GAM_B_LUT6	R/W	H'0400 0400	H'FFFF 789C	32/16
灰度校正部的表设定寄存器 B7	GAM_B_LUT7	R/W	H'0400 0400	H'FFFF 78A0	32/16
灰度校正部的表设定寄存器 B8	GAM_B_LUT8	R/W	H'0400 0400	H'FFFF 78A4	32/16
灰度校正部的表设定寄存器 B9	GAM_B_LUT9	R/W	H'0400 0400	H'FFFF 78A8	32/16
灰度校正部的表设定寄存器 B10	GAM_B_LUT10	R/W	H'0400 0400	H'FFFF 78AC	32/16
灰度校正部的表设定寄存器 B11	GAM_B_LUT11	R/W	H'0400 0400	H'FFFF 78B0	32/16
灰度校正部的表设定寄存器 B12	GAM_B_LUT12	R/W	H'0400 0400	H'FFFF 78B4	32/16
灰度校正部的表设定寄存器 B13	GAM_B_LUT13	R/W	H'0400 0400	H'FFFF 78B8	32/16
灰度校正部的表设定寄存器 B14	GAM_B_LUT14	R/W	H'0400 0400	H'FFFF 78BC	32/16
灰度校正部的表设定寄存器 B15	GAM_B_LUT15	R/W	H'0400 0400	H'FFFF 78C0	32/16
灰度校正部的表设定寄存器 B16	GAM_B_LUT16	R/W	H'0400 0400	H'FFFF 78C4	32/16
灰度校正部的区域设定寄存器 B1	GAM_B_AREA1	R/W	H'0008 1018	H'FFFF 78C8	32/16
灰度校正部的区域设定寄存器 B2	GAM_B_AREA2	R/W	H'2028 3038	H'FFFF 78CC	32/16
灰度校正部的区域设定寄存器 B3	GAM_B_AREA3	R/W	H'4048 5058	H'FFFF 78D0	32/16
灰度校正部的区域设定寄存器 B4	GAM_B_AREA4	R/W	H'6068 7078	H'FFFF 78D4	32/16
灰度校正部的区域设定寄存器 B5	GAM_B_AREA5	R/W	H'8088 9098	H'FFFF 78D8	32/16
灰度校正部的区域设定寄存器 B6	GAM_B_AREA6	R/W	H'A0A8 B0B8	H'FFFF 78DC	32/16
灰度校正部的区域设定寄存器 B7	GAM_B_AREA7	R/W	H'C0C8 D0D8	H'FFFF 78E0	32/16
灰度校正部的区域设定寄存器 B8	GAM_B_AREA8	R/W	H'E0E8 F0F8	H'FFFF 78E4	32/16
灰度校正部的寄存器更新控制寄存器 R	GAM_R_UPDATE	R/WC1	H'0000 0000	H'FFFF 7900	32/16
灰度校正部的表设定寄存器 R1	GAM_R_LUT1	R/W	H'0400 0400	H'FFFF 7908	32/16
灰度校正部的表设定寄存器 R2	GAM_R_LUT2	R/W	H'0400 0400	H'FFFF 790C	32/16
灰度校正部的表设定寄存器 R3	GAM_R_LUT3	R/W	H'0400 0400	H'FFFF 7910	32/16
灰度校正部的表设定寄存器 R4	GAM_R_LUT4	R/W	H'0400 0400	H'FFFF 7914	32/16
灰度校正部的表设定寄存器 R5	GAM_R_LUT5	R/W	H'0400 0400	H'FFFF 7918	32/16
灰度校正部的表设定寄存器 R6	GAM_R_LUT6	R/W	H'0400 0400	H'FFFF 791C	32/16
灰度校正部的表设定寄存器 R7	GAM_R_LUT7	R/W	H'0400 0400	H'FFFF 7920	32/16
灰度校正部的表设定寄存器 R8	GAM_R_LUT8	R/W	H'0400 0400	H'FFFF 7924	32/16
灰度校正部的表设定寄存器 R9	GAM_R_LUT9	R/W	H'0400 0400	H'FFFF 7928	32/16
灰度校正部的表设定寄存器 R10	GAM_R_LUT10	R/W	H'0400 0400	H'FFFF 792C	32/16
灰度校正部的表设定寄存器 R11	GAM_R_LUT11	R/W	H'0400 0400	H'FFFF 7930	32/16
灰度校正部的表设定寄存器 R12	GAM_R_LUT12	R/W	H'0400 0400	H'FFFF 7934	32/16
灰度校正部的表设定寄存器 R13	GAM_R_LUT13	R/W	H'0400 0400	H'FFFF 7938	32/16
灰度校正部的表设定寄存器 R14	GAM_R_LUT14	R/W	H'0400 0400	H'FFFF 793C	32/16
灰度校正部的表设定寄存器 R15	GAM_R_LUT15	R/W	H'0400 0400	H'FFFF 7940	32/16
灰度校正部的表设定寄存器 R16	GAM_R_LUT16	R/W	H'0400 0400	H'FFFF 7944	32/16
灰度校正部的区域设定寄存器 R1	GAM_R_AREA1	R/W	H'0008 1018	H'FFFF 7948	32/16
灰度校正部的区域设定寄存器 R2	GAM_R_AREA2	R/W	H'2028 3038	H'FFFF 794C	32/16
灰度校正部的区域设定寄存器 R3	GAM_R_AREA3	R/W	H'4048 5058	H'FFFF 7950	32/16
灰度校正部的区域设定寄存器 R4	GAM_R_AREA4	R/W	H'6068 7078	H'FFFF 7954	32/16

名称	略称	R/W	初始值	地址	存取长度
灰度校正部的区域设定寄存器 R5	GAM_R_AREA5	R/W	H'8088 9098	H'FFFF 7958	32/16
灰度校正部的区域设定寄存器 R6	GAM_R_AREA6	R/W	H'A0A8 B0B8	H'FFFF 795C	32/16
灰度校正部的区域设定寄存器 R7	GAM_R_AREA7	R/W	H'C0C8 D0D8	H'FFFF 7960	32/16
灰度校正部的区域设定寄存器 R8	GAM_R_AREA8	R/W	H'E0E8 F0F8	H'FFFF 7964	32/16

表 36.24 TCON 部的寄存器结构

名称	略称	R/W	初始值	地址	存取长度
TCON 寄存器的更新控制寄存器	TCON_UPDATE	R/WC1	H'0000 0000	H'FFFF 7980	32/16
TCON 基准时序设定寄存器	TCON_TIM	R/W	H'0190 0000	H'FFFF 7984	32/16
TCON 垂直时序设定寄存器 A1	TCON_TIM_STVA1	R/W	H'0000 0004	H'FFFF 7988	32/16
TCON 垂直时序设定寄存器 A2	TCON_TIM_STVA2	R/W	H'0000 0010	H'FFFF 798C	32/16
TCON 垂直时序设定寄存器 B1	TCON_TIM_STVB1	R/W	H'0046 03C0	H'FFFF 7990	32/16
TCON 垂直时序设定寄存器 B2	TCON_TIM_STVB2	R/W	H'0000 0001	H'FFFF 7994	32/16
TCON 水平时序设定寄存器 STH1	TCON_TIM_STH1	R/W	H'0000 0060	H'FFFF 7998	32/16
TCON 水平时序设定寄存器 STH2	TCON_TIM_STH2	R/W	H'0000 0012	H'FFFF 799C	32/16
TCON 水平时序设定寄存器 STB1	TCON_TIM_STB1	R/W	H'0090 0280	H'FFFF 79A0	32/16
TCON 水平时序设定寄存器 STB2	TCON_TIM_STB2	R/W	H'0000 0007	H'FFFF 79A4	32/16
TCON 水平时序设定寄存器 CPV1	TCON_TIM_CPV1	R/W	H'0000 0000	H'FFFF 79A8	32/16
TCON 水平时序设定寄存器 CPV2	TCON_TIM_CPV2	R/W	H'0000 0004	H'FFFF 79AC	32/16
TCON 水平时序设定寄存器 POLA1	TCON_TIM_POLA1	R/W	H'0000 0000	H'FFFF 79B0	32/16
TCON 水平时序设定寄存器 POLA2	TCON_TIM_POLA2	R/W	H'0000 1005	H'FFFF 79B4	32/16
TCON 水平时序设定寄存器 POLB1	TCON_TIM_POLB1	R/W	H'0000 0000	H'FFFF 79B8	32/16
TCON 水平时序设定寄存器 POLB2	TCON_TIM_POLB2	R/W	H'0000 1006	H'FFFF 79BC	32/16
TCON 数据允许极性设定寄存器	TCON_TIM_DE	R/W	H'0000 0000	H'FFFF 79C0	32/16

表 36.25 输出控制部的寄存器结构

名称	略称	R/W	初始值	地址	存取长度
输出控制部的寄存器更新控制寄存器	OUT_UPDATE	R/WC1	H'0000 0000	H'FFFF 7A00	32/16
输出接口寄存器	OUT_SET	R/W	H'001F 0000	H'FFFF 7A04	32/16
亮度 (DC) 校正寄存器 1	OUT_BRIGHT1	R/W	H'0000 0200	H'FFFF 7A08	32/16
亮度 (DC) 校正寄存器 2	OUT_BRIGHT2	R/W	H'0200 0200	H'FFFF 7A0C	32/16
对比度 (增益) 校正寄存器	OUT_CONTRAST	R/W	H'0080 8080	H'FFFF 7A10	32/16
显示屏抖动寄存器	OUT_PDTHA	R/W	H'0000 3021	H'FFFF 7A14	32/16
输出相位控制寄存器	OUT_CLK_PHASE	R/W	H'0000 0000	H'FFFF 7A24	32/16

36.2.1 灰度校正部的寄存器更新控制寄存器 G (GAM_G_UPDATE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GAM_G_VEN
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

位	位名	初始值	R/W	说明
31 ~ 16	—	全 0	R	保留位 读写值都为“0”。
15 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GAM_G_VEN	0	R/WC1	灰度校正 (G) 的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器

36.2.2 灰度校正部的功能切换寄存器 (GAM_SW)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GAM_ON
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 16	—	全 0	R	保留位 读写值都为“0”。
15 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GAM_ON	0	R/W	灰度校正的 ON/OFF 控制 0: OFF 1: ON

【注】 在全部灰度校正部的寄存器更新控制寄存器 G (GAM_G_UPDATE) 的 GAM_G_VEN 位为“1”时更新此寄存器。

36.2.3 灰度校正部的表设定寄存器 G1 ~ 16 (GAM_G_LUT1 ~ 16)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GAM_G_GAIN_xx[10:0]										
初始值:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GAM_G_GAIN_yy[10:0]										
初始值:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	*	1024	R/W	GAM_G_LUT1: G 信号区域 0 的增益调整 GAM_G_LUT2: G 信号区域 2 的增益调整 GAM_G_LUT3: G 信号区域 4 的增益调整 GAM_G_LUT4: G 信号区域 6 的增益调整 GAM_G_LUT5: G 信号区域 8 的增益调整 GAM_G_LUT6: G 信号区域 10 的增益调整 GAM_G_LUT7: G 信号区域 12 的增益调整 GAM_G_LUT8: G 信号区域 14 的增益调整 GAM_G_LUT9: G 信号区域 16 的增益调整 GAM_G_LUT10: G 信号区域 18 的增益调整 GAM_G_LUT11: G 信号区域 20 的增益调整 GAM_G_LUT12: G 信号区域 22 的增益调整 GAM_G_LUT13: G 信号区域 24 的增益调整 GAM_G_LUT14: G 信号区域 26 的增益调整 GAM_G_LUT15: G 信号区域 28 的增益调整 GAM_G_LUT16: G 信号区域 30 的增益调整 无符号 (0 ~ 2047[LSB], 1024[LSB]=1.0[倍]) *: 位名 GAM_G_LUT1: GAM_G_GAIN_00[10:0] GAM_G_LUT2: GAM_G_GAIN_02[10:0] GAM_G_LUT3: GAM_G_GAIN_04[10:0] GAM_G_LUT4: GAM_G_GAIN_06[10:0] GAM_G_LUT5: GAM_G_GAIN_08[10:0] GAM_G_LUT6: GAM_G_GAIN_10[10:0] GAM_G_LUT7: GAM_G_GAIN_12[10:0] GAM_G_LUT8: GAM_G_GAIN_14[10:0] GAM_G_LUT9: GAM_G_GAIN_16[10:0] GAM_G_LUT10: GAM_G_GAIN_18[10:0] GAM_G_LUT11: GAM_G_GAIN_20[10:0] GAM_G_LUT12: GAM_G_GAIN_22[10:0] GAM_G_LUT13: GAM_G_GAIN_24[10:0] GAM_G_LUT14: GAM_G_GAIN_26[10:0] GAM_G_LUT15: GAM_G_GAIN_28[10:0] GAM_G_LUT16: GAM_G_GAIN_30[10:0]

位	位名	初始值	R/W	说明
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	*	1024	R/W	GAM_G_LUT1: G 信号区域 1 的增益调整 GAM_G_LUT2: G 信号区域 3 的增益调整 GAM_G_LUT3: G 信号区域 5 的增益调整 GAM_G_LUT4: G 信号区域 7 的增益调整 GAM_G_LUT5: G 信号区域 9 的增益调整 GAM_G_LUT6: G 信号区域 11 的增益调整 GAM_G_LUT7: G 信号区域 13 的增益调整 GAM_G_LUT8: G 信号区域 15 的增益调整 GAM_G_LUT9: G 信号区域 17 的增益调整 GAM_G_LUT10: G 信号区域 19 的增益调整 GAM_G_LUT11: G 信号区域 21 的增益调整 GAM_G_LUT12: G 信号区域 23 的增益调整 GAM_G_LUT13: G 信号区域 25 的增益调整 GAM_G_LUT14: G 信号区域 27 的增益调整 GAM_G_LUT15: G 信号区域 29 的增益调整 GAM_G_LUT16: G 信号区域 31 的增益调整 无符号 (0 ~ 2047[LSB], 1024[LSB]=1.0[倍]) *: 位名 GAM_G_LUT1: GAM_G_GAIN_01[10:0] GAM_G_LUT2: GAM_G_GAIN_03[10:0] GAM_G_LUT3: GAM_G_GAIN_05[10:0] GAM_G_LUT4: GAM_G_GAIN_07[10:0] GAM_G_LUT5: GAM_G_GAIN_09[10:0] GAM_G_LUT6: GAM_G_GAIN_11[10:0] GAM_G_LUT7: GAM_G_GAIN_13[10:0] GAM_G_LUT8: GAM_G_GAIN_15[10:0] GAM_G_LUT9: GAM_G_GAIN_17[10:0] GAM_G_LUT10: GAM_G_GAIN_19[10:0] GAM_G_LUT11: GAM_G_GAIN_21[10:0] GAM_G_LUT12: GAM_G_GAIN_23[10:0] GAM_G_LUT13: GAM_G_GAIN_25[10:0] GAM_G_LUT14: GAM_G_GAIN_27[10:0] GAM_G_LUT15: GAM_G_GAIN_29[10:0] GAM_G_LUT16: GAM_G_GAIN_31[10:0]

【注】 在全部灰度校正部的寄存器更新控制寄存器 G (GAM_G_UPDATE) 的 GAM_G_VEN 位为“1”时更新此寄存器。

36.2.4 灰度校正部的区域设定寄存器 G1 (GAM_G_AREA1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GAM_G_TH_01[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_G_TH_02[7:0]								GAM_G_TH_03[7:0]							
初始值:	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23 ~ 16	GAM_G_TH_01 [7:0]	8	R/W	G 信号区域 1 的开始阈值 无符号 (0 ~ 255[LSB]) 0 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_G_TH_02 [7:0]	16	R/W	G 信号区域 2 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_G_TH_03 [7:0]	24	R/W	G 信号区域 3 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 G (GAM_G_UPDATE) 的 GAM_G_VEN 位为“1”时更新此寄存器。

36.2.5 灰度校正部的区域设定寄存器 G2 (GAM_G_AREA2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_G_TH_04[7:0]								GAM_G_TH_05[7:0]							
初始值:	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_G_TH_06[7:0]								GAM_G_TH_07[7:0]							
初始值:	0	0	1	1	0	0	0	0	0	0	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_G_TH_04 [7:0]	32	R/W	G 信号区域 4 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_G_TH_05 [7:0]	40	R/W	G 信号区域 5 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_G_TH_06 [7:0]	48	R/W	G 信号区域 6 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_G_TH_07 [7:0]	56	R/W	G 信号区域 7 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 G (GAM_G_UPDATE) 的 GAM_G_VEN 位为“1”时更新此寄存器。

36.2.6 灰度校正部的区域设定寄存器 G3 (GAM_G_AREA3)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_G_TH_08[7:0]								GAM_G_TH_09[7:0]							
初始值:	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_G_TH_10[7:0]								GAM_G_TH_11[7:0]							
初始值:	0	1	0	1	0	0	0	0	0	1	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_G_TH_08 [7:0]	64	R/W	G 信号区域 8 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_G_TH_09 [7:0]	72	R/W	G 信号区域 9 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_G_TH_10 [7:0]	80	R/W	G 信号区域 10 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_G_TH_11 [7:0]	88	R/W	G 信号区域 11 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 G (GAM_G_UPDATE) 的 GAM_G_VEN 位为“1”时更新此寄存器。

36.2.7 灰度校正部的区域设定寄存器 G4 (GAM_G_AREA4)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_G_TH_12[7:0]								GAM_G_TH_13[7:0]							
初始值:	0	1	1	0	0	0	0	0	0	1	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_G_TH_14[7:0]								GAM_G_TH_15[7:0]							
初始值:	0	1	1	1	0	0	0	0	0	1	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_G_TH_12 [7:0]	96	R/W	G 信号区域 12 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_G_TH_13 [7:0]	104	R/W	G 信号区域 13 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_G_TH_14 [7:0]	112	R/W	G 信号区域 14 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_G_TH_15 [7:0]	120	R/W	G 信号区域 15 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部寄存器更新控制寄存器 G (GAM_G_UPDATE) 的 GAM_G_VEN 位为 “1” 时更新此寄存器。

36.2.8 灰度校正部的区域设定寄存器 G5 (GAM_G_AREA5)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_G_TH_16[7:0]								GAM_G_TH_17[7:0]							
初始值:	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_G_TH_18[7:0]								GAM_G_TH_19[7:0]							
初始值:	1	0	0	1	0	0	0	0	1	0	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_G_TH_16 [7:0]	128	R/W	G 信号区域 16 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_G_TH_17 [7:0]	136	R/W	G 信号区域 17 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_G_TH_18 [7:0]	144	R/W	G 信号区域 18 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_G_TH_19 [7:0]	152	R/W	G 信号区域 19 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 G (GAM_G_UPDATE) 的 GAM_G_VEN 位为“1”时更新此寄存器。

36.2.9 灰度校正部的区域设定寄存器 G6 (GAM_G_AREA6)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_G_TH_20[7:0]								GAM_G_TH_21[7:0]							
初始值:	1	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_G_TH_22[7:0]								GAM_G_TH_23[7:0]							
初始值:	1	0	1	1	0	0	0	0	1	0	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_G_TH_20 [7:0]	160	R/W	G 信号区域 20 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_G_TH_21 [7:0]	168	R/W	G 信号区域 21 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_G_TH_22 [7:0]	176	R/W	G 信号区域 22 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_G_TH_23 [7:0]	184	R/W	G 信号区域 23 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 G (GAM_G_UPDATE) 的 GAM_G_VEN 位为“1”时更新此寄存器。

36.2.10 灰度校正部的区域设定寄存器 G7 (GAM_G_AREA7)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_G_TH_24[7:0]								GAM_G_TH_25[7:0]							
初始值:	1	1	0	0	0	0	0	0	1	1	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_G_TH_26[7:0]								GAM_G_TH_27[7:0]							
初始值:	1	1	0	1	0	0	0	0	1	1	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_G_TH_24 [7:0]	192	R/W	G 信号区域 24 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_G_TH_25 [7:0]	200	R/W	G 信号区域 25 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_G_TH_26 [7:0]	208	R/W	G 信号区域 26 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_G_TH_27 [7:0]	216	R/W	G 信号区域 27 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 G (GAM_G_UPDATE) 的 GAM_G_VEN 位为“1”时更新此寄存器。

36.2.11 灰度校正部的区域设定寄存器 G8 (GAM_G_AREA8)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_G_TH_28[7:0]							GAM_G_TH_29[7:0]								
初始值:	1	1	1	0	0	0	0	0	1	1	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_G_TH_30[7:0]							GAM_G_TH_31[7:0]								
初始值:	1	1	1	1	0	0	0	0	1	1	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_G_TH_28 [7:0]	224	R/W	G 信号区域 28 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_G_TH_29 [7:0]	232	R/W	G 信号区域 29 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_G_TH_30 [7:0]	240	R/W	G 信号区域 30 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_G_TH_31 [7:0]	248	R/W	G 信号区域 31 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 ≤ 255

【注】 在全部灰度校正部的寄存器更新控制寄存器 G (GAM_G_UPDATE) 的 GAM_G_VEN 位为“1”时更新此寄存器。

36.2.12 灰度校正部的寄存器更新控制寄存器 B (GAM_B_UPDATE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GAM_B_VEN
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

位	位名	初始值	R/W	说明
31 ~ 16	—	全 0	R	保留位 读写值都为“0”。
15 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GAM_B_VEN	0	R/WC1	灰度校正 (B) 的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器

36.2.13 灰度校正部的表设定寄存器 B1 ~ 16 (GAM_B_LUT1 ~ 16)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GAM_B_GAIN_xx[10:0]										
初始值:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GAM_B_GAIN_yy[10:0]										
初始值:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。

位	位名	初始值	R/W	说明
26 ~ 16	*	1024	R/W	GAM_B_LUT1: B 信号区域 0 的增益调整 GAM_B_LUT2: B 信号区域 2 的增益调整 GAM_B_LUT3: B 信号区域 4 的增益调整 GAM_B_LUT4: B 信号区域 6 的增益调整 GAM_B_LUT5: B 信号区域 8 的增益调整 GAM_B_LUT6: B 信号区域 10 的增益调整 GAM_B_LUT7: B 信号区域 12 的增益调整 GAM_B_LUT8: B 信号区域 14 的增益调整 GAM_B_LUT9: B 信号区域 16 的增益调整 GAM_B_LUT10: B 信号区域 18 的增益调整 GAM_B_LUT11: B 信号区域 20 的增益调整 GAM_B_LUT12: B 信号区域 22 的增益调整 GAM_B_LUT13: B 信号区域 24 的增益调整 GAM_B_LUT14: B 信号区域 26 的增益调整 GAM_B_LUT15: B 信号区域 28 的增益调整 GAM_B_LUT16: B 信号区域 30 的增益调整 无符号 (0 ~ 2047[LSB], 1024[LSB]=1.0[倍]) *: 位名 GAM_B_LUT1: GAM_B_GAIN_00[10:0] GAM_B_LUT2: GAM_B_GAIN_02[10:0] GAM_B_LUT3: GAM_B_GAIN_04[10:0] GAM_B_LUT4: GAM_B_GAIN_06[10:0] GAM_B_LUT5: GAM_B_GAIN_08[10:0] GAM_B_LUT6: GAM_B_GAIN_10[10:0] GAM_B_LUT7: GAM_B_GAIN_12[10:0] GAM_B_LUT8: GAM_B_GAIN_14[10:0] GAM_B_LUT9: GAM_B_GAIN_16[10:0] GAM_B_LUT10: GAM_B_GAIN_18[10:0] GAM_B_LUT11: GAM_B_GAIN_20[10:0] GAM_B_LUT12: GAM_B_GAIN_22[10:0] GAM_B_LUT13: GAM_B_GAIN_24[10:0] GAM_B_LUT14: GAM_B_GAIN_26[10:0] GAM_B_LUT15: GAM_B_GAIN_28[10:0] GAM_B_LUT16: GAM_B_GAIN_30[10:0]
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。

位	位名	初始值	R/W	说明
10 ~ 0	*	1024	R/W	GAM_B_LUT1: B 信号区域 1 的增益调整 GAM_B_LUT2: B 信号区域 3 的增益调整 GAM_B_LUT3: B 信号区域 5 的增益调整 GAM_B_LUT4: B 信号区域 7 的增益调整 GAM_B_LUT5: B 信号区域 9 的增益调整 GAM_B_LUT6: B 信号区域 11 的增益调整 GAM_B_LUT7: B 信号区域 13 的增益调整 GAM_B_LUT8: B 信号区域 15 的增益调整 GAM_B_LUT9: B 信号区域 17 的增益调整 GAM_B_LUT10: B 信号区域 19 的增益调整 GAM_B_LUT11: B 信号区域 21 的增益调整 GAM_B_LUT12: B 信号区域 23 的增益调整 GAM_B_LUT13: B 信号区域 25 的增益调整 GAM_B_LUT14: B 信号区域 27 的增益调整 GAM_B_LUT15: B 信号区域 29 的增益调整 GAM_B_LUT16: B 信号区域 31 的增益调整 无符号 (0 ~ 2047[LSB], 1024[LSB]=1.0[倍]) *: 位名 GAM_B_LUT1: GAM_B_GAIN_01[10:0] GAM_B_LUT2: GAM_B_GAIN_03[10:0] GAM_B_LUT3: GAM_B_GAIN_05[10:0] GAM_B_LUT4: GAM_B_GAIN_07[10:0] GAM_B_LUT5: GAM_B_GAIN_09[10:0] GAM_B_LUT6: GAM_B_GAIN_11[10:0] GAM_B_LUT7: GAM_B_GAIN_13[10:0] GAM_B_LUT8: GAM_B_GAIN_15[10:0] GAM_B_LUT9: GAM_B_GAIN_17[10:0] GAM_B_LUT10: GAM_B_GAIN_19[10:0] GAM_B_LUT11: GAM_B_GAIN_21[10:0] GAM_B_LUT12: GAM_B_GAIN_23[10:0] GAM_B_LUT13: GAM_B_GAIN_25[10:0] GAM_B_LUT14: GAM_B_GAIN_27[10:0] GAM_B_LUT15: GAM_B_GAIN_29[10:0] GAM_B_LUT16: GAM_B_GAIN_31[10:0]

【注】 在全部灰度校正部的寄存器更新控制寄存器 B (GAM_B_UPDATE) 的 GAM_B_VEN 位为“1”时更新此寄存器。

36.2.14 灰度校正部的区域设定寄存器 B1 (GAM_B_AREA1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GAM_B_TH_01[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_02[7:0]								GAM_B_TH_03[7:0]							
初始值:	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23 ~ 16	GAM_B_TH_01 [7:0]	8	R/W	B 信号区域 1 的开始阈值 无符号 (0 ~ 255[LSB]) 0 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_B_TH_02 [7:0]	16	R/W	B 信号区域 2 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_B_TH_03 [7:0]	24	R/W	B 信号区域 3 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 B (GAM_B_UPDATE) 的 GAM_B_VEN 位为“1”时更新此寄存器。

36.2.15 灰度校正部的区域设定寄存器 B2 (GAM_B_AREA2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_04[7:0]								GAM_B_TH_05[7:0]							
初始值:	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_06[7:0]								GAM_B_TH_07[7:0]							
初始值:	0	0	1	1	0	0	0	0	0	0	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_B_TH_04 [7:0]	32	R/W	B 信号区域 4 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_B_TH_05 [7:0]	40	R/W	B 信号区域 5 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_B_TH_06 [7:0]	48	R/W	B 信号区域 6 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_B_TH_07 [7:0]	56	R/W	B 信号区域 7 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 B (GAM_B_UPDATE) 的 GAM_B_VEN 位为 “1” 时更新此寄存器。

36.2.16 灰度校正部的区域设定寄存器 B3 (GAM_B_AREA3)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_08[7:0]								GAM_B_TH_09[7:0]							
初始值:	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_10[7:0]								GAM_B_TH_11[7:0]							
初始值:	0	1	0	1	0	0	0	0	0	1	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_B_TH_08 [7:0]	64	R/W	B 信号区域 8 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_B_TH_09 [7:0]	72	R/W	B 信号区域 9 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_B_TH_10 [7:0]	80	R/W	B 信号区域 10 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_B_TH_11 [7:0]	88	R/W	B 信号区域 11 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 B (GAM_B_UPDATE) 的 GAM_B_VEN 位为 “1” 时更新此寄存器。

36.2.17 灰度校正部的区域设定寄存器 B4 (GAM_B_AREA4)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_12[7:0]							GAM_B_TH_13[7:0]								
初始值:	0	1	1	0	0	0	0	0	0	1	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_14[7:0]							GAM_B_TH_15[7:0]								
初始值:	0	1	1	1	0	0	0	0	0	1	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_B_TH_12 [7:0]	96	R/W	B 信号区域 12 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_B_TH_13 [7:0]	104	R/W	B 信号区域 13 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_B_TH_14 [7:0]	112	R/W	B 信号区域 14 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_B_TH_15 [7:0]	120	R/W	B 信号区域 15 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 B (GAM_B_UPDATE) 的 GAM_B_VEN 位为 “1” 时更新此寄存器。

36.2.18 灰度校正部的区域设定寄存器 B5 (GAM_B_AREA5)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_16[7:0]								GAM_B_TH_17[7:0]							
初始值:	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_18[7:0]								GAM_B_TH_19[7:0]							
初始值:	1	0	0	1	0	0	0	0	1	0	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_B_TH_16 [7:0]	128	R/W	B 信号区域 16 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_B_TH_17 [7:0]	136	R/W	B 信号区域 17 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_B_TH_18 [7:0]	144	R/W	B 信号区域 18 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_B_TH_19 [7:0]	152	R/W	B 信号区域 19 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 B (GAM_B_UPDATE) 的 GAM_B_VEN 位为 “1” 时更新此寄存器。

36.2.19 灰度校正部的区域设定寄存器 B6 (GAM_B_AREA6)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_20[7:0]								GAM_B_TH_21[7:0]							
初始值:	1	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_22[7:0]								GAM_B_TH_23[7:0]							
初始值:	1	0	1	1	0	0	0	0	1	0	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_B_TH_20 [7:0]	160	R/W	B 信号区域 20 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_B_TH_21 [7:0]	168	R/W	B 信号区域 21 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_B_TH_22 [7:0]	176	R/W	B 信号区域 22 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_B_TH_23 [7:0]	184	R/W	B 信号区域 23 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 B (GAM_B_UPDATE) 的 GAM_B_VEN 位为 “1” 时更新此寄存器。

36.2.20 灰度校正部的区域设定寄存器 B7 (GAM_B_AREA7)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_24[7:0]								GAM_B_TH_25[7:0]							
初始值:	1	1	0	0	0	0	0	0	1	1	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_26[7:0]								GAM_B_TH_27[7:0]							
初始值:	1	1	0	1	0	0	0	0	1	1	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_B_TH_24 [7:0]	192	R/W	B 信号区域 24 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_B_TH_25 [7:0]	200	R/W	B 信号区域 25 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_B_TH_26 [7:0]	208	R/W	B 信号区域 26 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_B_TH_27 [7:0]	216	R/W	B 信号区域 27 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 B (GAM_B_UPDATE) 的 GAM_B_VEN 位为 “1” 时更新此寄存器。

36.2.21 灰度校正部的区域设定寄存器 B8 (GAM_B_AREA8)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_28[7:0]								GAM_B_TH_29[7:0]							
初始值:	1	1	1	0	0	0	0	0	1	1	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_30[7:0]								GAM_B_TH_31[7:0]							
初始值:	1	1	1	1	0	0	0	0	1	1	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_B_TH_28 [7:0]	224	R/W	B 信号区域 28 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_B_TH_29 [7:0]	232	R/W	B 信号区域 29 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_B_TH_30 [7:0]	240	R/W	B 信号区域 30 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_B_TH_31 [7:0]	248	R/W	B 信号区域 31 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 ≤ 255

【注】 在全部灰度校正部的寄存器更新控制寄存器 B (GAM_B_UPDATE) 的 GAM_B_VEN 位为 “1” 时更新此寄存器。

36.2.22 灰度校正部的寄存器更新控制寄存器 R (GAM_R_UPDATE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GAM_R_VEN
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

位	位名	初始值	R/W	说明
31 ~ 16	—	全 0	R	保留位 读写值都为“0”。
15 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GAM_R_VEN	0	R/WC1	灰度校正 (R) 的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器

36.2.23 灰度校正部的表设定寄存器 R1 ~ 16 (GAM_R_LUT1 ~ 16)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GAM_R_GAIN_xx[10:0]										
初始值:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GAM_R_GAIN_yy[10:0]										
初始值:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。

位	位名	初始值	R/W	说明
26 ~ 16	*	1024	R/W	GAM_R_LUT1: R 信号区域 0 的增益调整 GAM_R_LUT2: R 信号区域 2 的增益调整 GAM_R_LUT3: R 信号区域 4 的增益调整 GAM_R_LUT4: R 信号区域 6 的增益调整 GAM_R_LUT5: R 信号区域 8 的增益调整 GAM_R_LUT6: R 信号区域 10 的增益调整 GAM_R_LUT7: R 信号区域 12 的增益调整 GAM_R_LUT8: R 信号区域 14 的增益调整 GAM_R_LUT9: R 信号区域 16 的增益调整 GAM_R_LUT10: R 信号区域 18 的增益调整 GAM_R_LUT11: R 信号区域 20 的增益调整 GAM_R_LUT12: R 信号区域 22 的增益调整 GAM_R_LUT13: R 信号区域 24 的增益调整 GAM_R_LUT14: R 信号区域 26 的增益调整 GAM_R_LUT15: R 信号区域 28 的增益调整 GAM_R_LUT16: R 信号区域 30 的增益调整 无符号 (0 ~ 2047[LSB], 1024[LSB]=1.0[倍]) *: 位名 GAM_R_LUT1: GAM_R_GAIN_00[10:0] GAM_R_LUT2: GAM_R_GAIN_02[10:0] GAM_R_LUT3: GAM_R_GAIN_04[10:0] GAM_R_LUT4: GAM_R_GAIN_06[10:0] GAM_R_LUT5: GAM_R_GAIN_08[10:0] GAM_R_LUT6: GAM_R_GAIN_10[10:0] GAM_R_LUT7: GAM_R_GAIN_12[10:0] GAM_R_LUT8: GAM_R_GAIN_14[10:0] GAM_R_LUT9: GAM_R_GAIN_16[10:0] GAM_R_LUT10: GAM_R_GAIN_18[10:0] GAM_R_LUT11: GAM_R_GAIN_20[10:0] GAM_R_LUT12: GAM_R_GAIN_22[10:0] GAM_R_LUT13: GAM_R_GAIN_24[10:0] GAM_R_LUT14: GAM_R_GAIN_26[10:0] GAM_R_LUT15: GAM_R_GAIN_28[10:0] GAM_R_LUT16: GAM_R_GAIN_30[10:0]
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。

位	位名	初始值	R/W	说明
10 ~ 0	*	1024	R/W	GAM_R_LUT1: R 信号区域 1 的增益调整 GAM_R_LUT2: R 信号区域 3 的增益调整 GAM_R_LUT3: R 信号区域 5 的增益调整 GAM_R_LUT4: R 信号区域 7 的增益调整 GAM_R_LUT5: R 信号区域 9 的增益调整 GAM_R_LUT6: R 信号区域 11 的增益调整 GAM_R_LUT7: R 信号区域 13 的增益调整 GAM_R_LUT8: R 信号区域 15 的增益调整 GAM_R_LUT9: R 信号区域 17 的增益调整 GAM_R_LUT10: R 信号区域 19 的增益调整 GAM_R_LUT11: R 信号区域 21 的增益调整 GAM_R_LUT12: R 信号区域 23 的增益调整 GAM_R_LUT13: R 信号区域 25 的增益调整 GAM_R_LUT14: R 信号区域 27 的增益调整 GAM_R_LUT15: R 信号区域 29 的增益调整 GAM_R_LUT16: R 信号区域 31 的增益调整 无符号 (0 ~ 2047[LSB], 1024[LSB]=1.0[倍]) *: 位名 GAM_R_LUT1: GAM_R_GAIN_01[10:0] GAM_R_LUT2: GAM_R_GAIN_03[10:0] GAM_R_LUT3: GAM_R_GAIN_05[10:0] GAM_R_LUT4: GAM_R_GAIN_07[10:0] GAM_R_LUT5: GAM_R_GAIN_09[10:0] GAM_R_LUT6: GAM_R_GAIN_11[10:0] GAM_R_LUT7: GAM_R_GAIN_13[10:0] GAM_R_LUT8: GAM_R_GAIN_15[10:0] GAM_R_LUT9: GAM_R_GAIN_17[10:0] GAM_R_LUT10: GAM_R_GAIN_19[10:0] GAM_R_LUT11: GAM_R_GAIN_21[10:0] GAM_R_LUT12: GAM_R_GAIN_23[10:0] GAM_R_LUT13: GAM_R_GAIN_25[10:0] GAM_R_LUT14: GAM_R_GAIN_27[10:0] GAM_R_LUT15: GAM_R_GAIN_29[10:0] GAM_R_LUT16: GAM_R_GAIN_31[10:0]

【注】 在全部灰度校正部寄存器更新控制寄存器 R (GAM_R_UPDATE) 的 GAM_R_VEN 位为 “1” 时更新此寄存器。

36.2.24 灰度校正部的区域设定寄存器 R1 (GAM_R_AREA1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GAM_R_TH_01[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_02[7:0]								GAM_R_TH_03[7:0]							
初始值:	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23 ~ 16	GAM_R_TH_01 [7:0]	8	R/W	R 信号区域 1 的开始阈值 无符号 (0 ~ 255[LSB]) 0 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_R_TH_02 [7:0]	16	R/W	R 信号区域 2 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_R_TH_03 [7:0]	24	R/W	R 信号区域 3 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 R (GAM_R_UPDATE) 的 GAM_R_VEN 位为“1”时更新此寄存器。

36.2.25 灰度校正部的区域设定寄存器 R2 (GAM_R_AREA2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_R_TH_04[7:0]								GAM_R_TH_05[7:0]							
初始值:	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_06[7:0]								GAM_R_TH_07[7:0]							
初始值:	0	0	1	1	0	0	0	0	0	0	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_R_TH_04 [7:0]	32	R/W	R 信号区域 4 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_R_TH_05 [7:0]	40	R/W	R 信号区域 5 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_R_TH_06 [7:0]	48	R/W	R 信号区域 6 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_R_TH_07 [7:0]	56	R/W	R 信号区域 7 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 R (GAM_R_UPDATE) 的 GAM_R_VEN 位为“1”时更新此寄存器。

36.2.26 灰度校正部的区域设定寄存器 R3 (GAM_R_AREA3)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_R_TH_08[7:0]								GAM_R_TH_09[7:0]							
初始值:	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_10[7:0]								GAM_R_TH_11[7:0]							
初始值:	0	1	0	1	0	0	0	0	0	1	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_R_TH_08 [7:0]	64	R/W	R 信号区域 8 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_R_TH_09 [7:0]	72	R/W	R 信号区域 9 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_R_TH_10 [7:0]	80	R/W	R 信号区域 10 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_R_TH_11 [7:0]	88	R/W	R 信号区域 11 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 R (GAM_R_UPDATE) 的 GAM_R_VEN 位为“1”时更新此寄存器。

36.2.27 灰度校正部的区域设定寄存器 R4 (GAM_R_AREA4)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_R_TH_12[7:0]								GAM_R_TH_13[7:0]							
初始值:	0	1	1	0	0	0	0	0	0	1	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_14[7:0]								GAM_R_TH_15[7:0]							
初始值:	0	1	1	1	0	0	0	0	0	1	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_R_TH_12 [7:0]	96	R/W	R 信号区域 12 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_R_TH_13 [7:0]	104	R/W	R 信号区域 13 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_R_TH_14 [7:0]	112	R/W	R 信号区域 14 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_R_TH_15 [7:0]	120	R/W	R 信号区域 15 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 R (GAM_R_UPDATE) 的 GAM_R_VEN 位为“1”时更新此寄存器。

36.2.28 灰度校正部的区域设定寄存器 R5 (GAM_R_AREA5)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_R_TH_16[7:0]								GAM_R_TH_17[7:0]							
初始值:	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_18[7:0]								GAM_R_TH_19[7:0]							
初始值:	1	0	0	1	0	0	0	0	1	0	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_R_TH_16 [7:0]	128	R/W	R 信号区域 16 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_R_TH_17 [7:0]	136	R/W	R 信号区域 17 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_R_TH_18 [7:0]	144	R/W	R 信号区域 18 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_R_TH_19 [7:0]	152	R/W	R 信号区域 19 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 R (GAM_R_UPDATE) 的 GAM_R_VEN 位为“1”时更新此寄存器。

36.2.29 灰度校正部的区域设定寄存器 R6 (GAM_R_AREA6)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_R_TH_20[7:0]								GAM_R_TH_21[7:0]							
初始值:	1	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_22[7:0]								GAM_R_TH_23[7:0]							
初始值:	1	0	1	1	0	0	0	0	1	0	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_R_TH_20 [7:0]	160	R/W	R 信号区域 20 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_R_TH_21 [7:0]	168	R/W	R 信号区域 21 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_R_TH_22 [7:0]	176	R/W	R 信号区域 22 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_R_TH_23 [7:0]	184	R/W	R 信号区域 23 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 R (GAM_R_UPDATE) 的 GAM_R_VEN 位为“1”时更新此寄存器。

36.2.30 灰度校正部的区域设定寄存器 R7 (GAM_R_AREA7)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_R_TH_24[7:0]								GAM_R_TH_25[7:0]							
初始值:	1	1	0	0	0	0	0	0	1	1	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_26[7:0]								GAM_R_TH_27[7:0]							
初始值:	1	1	0	1	0	0	0	0	1	1	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_R_TH_24 [7:0]	192	R/W	R 信号区域 24 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_R_TH_25 [7:0]	200	R/W	R 信号区域 25 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_R_TH_26 [7:0]	208	R/W	R 信号区域 26 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_R_TH_27 [7:0]	216	R/W	R 信号区域 27 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值

【注】 在全部灰度校正部的寄存器更新控制寄存器 R (GAM_R_UPDATE) 的 GAM_R_VEN 位为“1”时更新此寄存器。

36.2.31 灰度校正部的区域设定寄存器 R8 (GAM_R_AREA8)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_R_TH_28[7:0]								GAM_R_TH_29[7:0]							
初始值:	1	1	1	0	0	0	0	0	1	1	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_30[7:0]								GAM_R_TH_31[7:0]							
初始值:	1	1	1	1	0	0	0	0	1	1	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	GAM_R_TH_28 [7:0]	224	R/W	R 信号区域 28 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
23 ~ 16	GAM_R_TH_29 [7:0]	232	R/W	R 信号区域 29 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
15 ~ 8	GAM_R_TH_30 [7:0]	240	R/W	R 信号区域 30 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 < 下一个区域的阈值
7 ~ 0	GAM_R_TH_31 [7:0]	248	R/W	R 信号区域 31 的开始阈值 无符号 (0 ~ 255[LSB]) 前一个区域的阈值 ≤ 当前区域的阈值 ≤ 255

【注】 在全部灰度校正部的寄存器更新控制寄存器 R (GAM_R_UPDATE) 的 GAM_R_VEN 位为 “1” 时更新此寄存器。

36.2.32 TCON 寄存器的更新控制寄存器 (TCON_UPDATE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TCON_VEN
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

位	位名	初始值	R/W	说明
31 ~ 16	—	全 0	R	保留位 读写值都为“0”。
15 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	TCON_VEN	0	R/WC1	LCD TCON 的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器

36.2.33 TCON 基准时序设定寄存器 (TCON_TIM)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_HALF[10:0]										
初始值:	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_OFFSET[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	TCON_HALF [10:0]	400	R/W	1/2fH 时序设定 将从水平同步信号的上升沿开始的时钟个数指定为垂直计数器计数运行时序。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	TCON_OFFSET [10:0]	0	R/W	带偏移的水平同步信号的时序设定 指定从水平同步信号的上升沿开始的时钟个数。

【注】 在全部 TCON 寄存器的更新控制寄存器 (TCON_UPDATE) 的 TCON_VEN 位为“1”时更新此寄存器。

36.2.34 TCON 垂直时序设定寄存器 A1 (TCON_TIM_STVA1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_STVA_VS[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_STVA_VW[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	TCON_STVA_VS[10:0]	0	R/W	设定 STVA 信号的脉冲起始位置 (第 1 个变化时序)。在从垂直同步信号的上升沿经过 TCON_STVA_VS 后开始输出脉冲 (1/2fH 周期)。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	TCON_STVA_VW[10:0]	4	R/W	设定 STVA 信号的脉宽 (第 2 个变化时序)。输出 TCON_STVA_VW 期间的脉冲 (1/2fH 周期)。

【注】 在全部 TCON 寄存器的更新控制寄存器 (TCON_UPDATE) 的 TCON_VEN 位为“1”时更新此寄存器。

36.2.35 TCON 垂直时序设定寄存器 A2 (TCON_TIM_STVA2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	TCON_STVA_INV	—	TCON_STVA_SEL[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 16	—	全 0	R	保留位 读写值都为“0”。
15 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	TCON_STVA_INV	1	R/W	STVA 信号的极性反转控制 0: 不反转 1: 反转
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	TCON_STVA_SEL[2:0]	0	R/W	LCD_TCON0 引脚的输出信号选择 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

【注】 在全部 TCON 寄存器的更新控制寄存器 (TCON_UPDATE) 的 TCON_VEN 位为“1”时更新此寄存器。

36.2.36 TCON 垂直时序设定寄存器 B1 (TCON_TIM_STVB1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	—	—	—	—	—	TCON_STVB_VS[10:0]											
初始值:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	—	—	—	—	—	TCON_STVB_VW[10:0]											
初始值:	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	TCON_STVB_VS[10:0]	70	R/W	设定 STVB 信号的脉冲起始位置 (第 1 个变化时序)。在从垂直同步信号的上升沿经过 TCON_STVB_VS 后开始输出脉冲 (1/2fH 周期)。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	TCON_STVB_VW[10:0]	960	R/W	设定 STVB 信号的脉宽 (第 2 个变化时序)。输出 TCON_STVB_VW 期间的脉冲 (1/2fH 周期)。

【注】 在全部 TCON 寄存器的更新控制寄存器 (TCON_UPDATE) 的 TCON_VEN 位为“1”时更新此寄存器。

36.2.37 TCON 垂直时序设定寄存器 B2 (TCON_TIM_STVB2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	TCON_STVB_INV	—	TCON_STVB_SEL[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 16	—	全 0	R	保留位 读写值都为“0”。
15 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	TCON_STVB_INV	0	R/W	STVB 信号的极性反转控制 0: 不反转 1: 反转
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	TCON_STVB_SEL[2:0]	1	R/W	LCD_TCON1 引脚的输出信号选择 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

【注】 在全部 TCON 寄存器更新控制寄存器 (TCON_UPDATE) 的 TCON_VEN 位为“1”时更新此寄存器。

36.2.38 TCON 水平时序设定寄存器 STH1 (TCON_TIM_STH1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_STH_HS[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_STH_HW[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	TCON_STH_HS [10:0]	0	R/W	设定 STH 信号的脉冲起始位置 (第 1 个变化时序)。 在从水平同步信号的上升沿经过 TCON_STH_HS 后开始输出脉冲 (时钟周期)。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	TCON_STH_HW [10:0]	96	R/W	设定 STH 信号的脉宽 (第 2 个变化时序)。 输出 TCON_STH_HW 期间的脉冲 (时钟周期)。

【注】 在全部 TCON 寄存器的更新控制寄存器 (TCON_UPDATE) 的 TCON_VEN 位为“1”时更新此寄存器。

36.2.39 TCON 水平时序设定寄存器 STH2 (TCON_TIM_STH2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TCON_STH_HS_SEL	—	—	—	TCON_STH_INV	—	TCON_STH_SEL[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	TCON_STH_HS_SEL	0	R/W	STH 信号的运行基准选择 0: 水平同步信号基准 1: 偏移后的水平同步信号基准
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	TCON_STH_INV	1	R/W	STH 信号的极性反转控制 0: 不反转 1: 反转
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	TCON_STH_SEL [2:0]	2	R/W	LCD_TCON2 引脚的输出信号选择 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

【注】 在全部 TCON 寄存器的更新控制寄存器 (TCON_UPDATE) 的 TCON_VEN 位为“1”时更新此寄存器。

36.2.40 TCON 水平时序设定寄存器 STB1 (TCON_TIM_STB1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_STB_HS[10:0]										
初始值:	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_STB_HW[10:0]										
初始值:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	TCON_STB_HS [10:0]	144	R/W	设定 STB 信号的脉冲起始位置 (第 1 个变化时序)。 在从水平同步信号的上升沿经过 TCON_STB_HS 后开始输出脉冲 (时钟周期)。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	TCON_STB_HW [10:0]	640	R/W	设定 STB 信号的脉宽 (第 2 个变化时序)。 输出 TCON_STB_HW 期间的脉冲 (时钟周期)。

【注】 在全部 TCON 寄存器的更新控制寄存器 (TCON_UPDATE) 的 TCON_VEN 位为“1”时更新此寄存器。

36.2.41 TCON 水平时序设定寄存器 STB2 (TCON_TIM_STB2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TCON_STB_HS_SEL	—	—	—	TCON_STB_INV	—	TCON_STB_SEL[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	TCON_STB_HS_SEL	0	R/W	STB 信号的运行基准选择 0: 水平同步信号基准 1: 偏移后的水平同步信号基准
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	TCON_STB_INV	0	R/W	STB 信号的极性反转控制 0: 不反转 1: 反转
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	TCON_STB_SEL [2:0]	7	R/W	LCD_TCON3 引脚的输出信号选择 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

【注】 在全部 TCON 寄存器的更新控制寄存器 (TCON_UPDATE) 的 TCON_VEN 位为“1”时更新此寄存器。

36.2.42 TCON 水平时序设定寄存器 CPV1 (TCON_TIM_CPV1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_CPV_HS[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_CPV_HW[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	TCON_CPV_HS [10:0]	0	R/W	设定 CPV 信号的脉冲起始位置 (第 1 个变化时序)。 在从水平同步信号的上升沿经过 TCON_CPV_HS 后开始输出脉冲 (时钟周期)。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	TCON_CPV_HW [10:0]	0	R/W	设定 CPV 信号的脉宽 (第 2 个变化时序)。 输出 TCON_CPV_HW 期间的脉冲 (时钟周期)。

【注】 在全部 TCON 寄存器的更新控制寄存器 (TCON_UPDATE) 的 TCON_VEN 位为“1”时更新此寄存器。

36.2.43 TCON 水平时序设定寄存器 CPV2 (TCON_TIM_CPV2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TCON_CPV_HS_SEL	—	—	—	TCON_CPV_INV	—	TCON_CPV_SEL[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	TCON_CPV_HS_SEL	0	R/W	CPV 信号的运行基准选择 0: 水平同步信号基准 1: 偏移后的水平同步信号基准
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	TCON_CPV_INV	0	R/W	CPV 信号的极性反转控制 0: 不反转 1: 反转
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	TCON_CPV_SEL[2:0]	4	R/W	LCD_TCON4 引脚的输出信号选择 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

【注】 在全部 TCON 寄存器更新控制寄存器 (TCON_UPDATE) 的 TCON_VEN 位为“1”时更新此寄存器。

36.2.44 TCON 水平时序设定寄存器 POLA1 (TCON_TIM_POLA1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_POLA_HS[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_POLA_HW[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	TCON_POLA_HS[10:0]	0	R/W	设定 POLA 信号的脉冲起始位置 (第 1 个变化时序)。 在从水平同步信号的上升沿经过 TCON_POLA_HS 后开始输出脉冲 (时钟周期)。 【注】 在设定 1×1、1×2 和 2×2 反向模式时, 必须设定大于等于“1”的值。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	TCON_POLA_HW[10:0]	0	R/W	设定 POLA 信号的脉宽 (第 2 个变化时序)。 输出 TCON_POLA_HW 期间的脉冲 (时钟周期)。

【注】 在全部 TCON 寄存器的更新控制寄存器 (TCON_UPDATE) 的 TCON_VEN 位为“1”时更新此寄存器。

36.2.45 TCON 水平时序设定寄存器 POLA2 (TCON_TIM_POLA2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TCON_POLA_MD[1:0]	—	—	—	TCON_POLA_HS_SEL	—	—	—	TCON_POLA_INV	—	TCON_POLA_SEL[2:0]			
初始值:	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	1
R/W:	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 14	—	全 0	R	保留位 读写值都为“0”。
13、12	TCON_POLA_MD[1:0]	1	R/W	POLA 信号的生成模式选择 0: 正常模式 在水平周期生成变化 2 次的信号。 1: 1×1 反向模式 在每个水平周期生成极性相反的信号。 2: 1×2 反向模式 在第 1 个水平期间将极性反转, 然后在每 2 个水平周期生成极性相反的信号。 3: 2×2 反向模式 在每 2 个水平周期生成极性相反的信号。
11 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	TCON_POLA_HS_SEL	0	R/W	POLA 信号的运行基准选择 0: 水平同步信号基准 1: 偏移后的水平同步信号基准
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	TCON_POLA_INV	0	R/W	POLA 信号的极性反转控制 0: 不反转 1: 反转
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	TCON_POLA_SEL[2:0]	5	R/W	LCD_TCON5 引脚的输出信号选择 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

【注】 在全部 TCON 寄存器的更新控制寄存器 (TCON_UPDATE) 的 TCON_VEN 位为“1”时更新此寄存器。

36.2.46 TCON 水平时序设定寄存器 POLB1 (TCON_TIM_POLB1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_POLB_HS[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_POLB_HW[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26 ~ 16	TCON_POLB_HS[10:0]	0	R/W	设定 POLB 信号的脉冲起始位置 (第 1 个变化时序)。 在从水平同步信号的上升沿经过 TCON_POLB_HS 后开始输出脉冲 (时钟周期)。
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	TCON_POLB_HW[10:0]	0	R/W	设定 POLB 信号的脉宽 (第 2 个变化时序)。 输出 TCON_POLB_HW 期间的脉冲 (时钟周期)。

【注】 在全部 TCON 寄存器的更新控制寄存器 (TCON_UPDATE) 的 TCON_VEN 位为“1”时更新此寄存器。

36.2.47 TCON 水平时序设定寄存器 POLB2 (TCON_TIM_POLB2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TCON_POLB_MD[1:0]	—	—	—	TCON_POLB_HS_SEL	—	—	—	TCON_POLB_INV	—	TCON_POLB_SEL[2:0]			
初始值:	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	0
R/W:	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 14	—	全 0	R	保留位 读写值都为“0”。
13、12	TCON_POLB_MD[1:0]	1	R/W	POLB 信号的生成模式选择 0: 正常模式 在水平周期生成变化 2 次的信号。 1: 1×1 反向模式 在每个水平周期生成极性相反的信号。 2: 1×2 反向模式 在第 1 个水平期间将极性反转, 然后在每 2 个水平周期生成极性相反的信号。 3: 2×2 反向模式 在每 2 个水平周期生成极性相反的信号。
11 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	TCON_POLB_HS_SEL	0	R/W	POLB 信号的运行基准选择 0: 水平同步信号基准 1: 偏移后的水平同步信号基准
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	TCON_POLB_INV	0	R/W	POLB 信号的极性反转控制 0: 不反转 1: 反转
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	TCON_POLB_SEL[2:0]	6	R/W	LCD_TCON6 引脚的输出信号选择 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

【注】 在全部 TCON 寄存器的更新控制寄存器 (TCON_UPDATE) 的 TCON_VEN 位为“1”时更新此寄存器。

36.2.48 TCON 数据允许极性设定寄存器 (TCON_TIM_DE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TCON_DE_INV
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	TCON_DE_INV	0	R/W	DE 信号的极性反转控制 0: 不反转 1: 反转

【注】 在全部 TCON 寄存器的更新控制寄存器 (TCON_UPDATE) 的 TCON_VEN 位为“1”时更新此寄存器。

36.2.49 输出控制部的寄存器更新控制寄存器 (OUT_UPDATE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OUTCNT_VEN
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

位	位名	初始值	R/W	说明
31 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	OUTCNT_VEN	0	R/WC1	亮度 / 对比度、抖动处理、输出接口的寄存器更新 0: 不更新寄存器 1: 在垂直同步信号的上升沿更新寄存器

36.2.50 输出接口寄存器 (OUT_SET)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	OUT_ENDIAN_ON	—	—	—	OUT_SWAP_ON	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	OUT_FORMAT[1:0]	—	—	OUT_FRQ_SEL[1:0]	—	—	—	—	—	OUT_DIR_SEL	—	—	—	OUT_PHASE[1:0]
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 29	—	全 0	R	保留位 读写值都为“0”。
28	OUT_ENDIAN_ON	0	R/W	位序变更 ON/OFF 控制 0: OFF 1: ON
27 ~ 25	—	全 0	R	保留位 读写值都为“0”。
24	OUT_SWAP_ON	0	R/W	B/R 信号交换的 ON/OFF 控制 0: OFF 1: ON
23 ~ 21	—	全 0	R	保留位 读写值都为“0”。
20 ~ 16	—	全 1	R	保留位 读写值都为“1”。
15、14	—	全 0	R	保留位 读写值都为“0”。
13、12	OUT_FORMAT [1:0]	0	R/W	输出格式选择 0: RGB888 1: RGB666 2: RGB565 3: 串行 RGB
11、10	—	全 0	R	保留位 读写值都为“0”。
9、8	OUT_FRQ_SEL [1:0]	0	R/W	时钟频率控制 0: 1 倍速 (并行 RGB) 1: 3 倍速 (串行 RGB) 2: 4 倍速 (串行 RGB) 3: 禁止设定
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	OUT_DIR_SEL	0	R/W	扫描方向选择 0: 正向扫描 1: 反向扫描

位	位名	初始值	R/W	说明
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	OUT_PHASE[1:0]	0	R/W	串行 RGB 输出时的时钟相位调整 3 倍速的情况 0: 0[clk] 1: 1[clk] 2: 2[clk] 3: 禁止设定 4 倍速的情况 0: 0[clk] 1: 1[clk] 2: 2[clk] 3: 3[clk]

【注】 在全部输出控制部的寄存器更新控制寄存器 (OUT_UPDATE) 的 OUTCNT_VEN 位为“1”时更新此寄存器。

36.2.51 亮度 (DC) 校正寄存器 1 (OUT_BRIGHT1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PBRT_G[9:0]									
初始值:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	PBRT_G[9:0]	512	R/W	G 信号的亮度 (DC) 调整 无符号 (0 (-512) ~ 512 (0) ~ 1023 (+511) [LSB], 带偏移的 512[LSB])

【注】 在全部输出控制部的寄存器更新控制寄存器 (OUT_UPDATE) 的 OUTCNT_VEN 位为“1”时更新此寄存器。

36.2.52 亮度 (DC) 校正寄存器 2 (OUT_BRIGHT2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	PBRT_B[9:0]									
初始值:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PBRT_R[9:0]									
初始值:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 26	—	全 0	R	保留位 读写值都为“0”。
25 ~ 16	PBRT_B[9:0]	512	R/W	B 信号的亮度 (DC) 调整 无符号 (0 (-512) ~ 512 (0) ~ 1023 (+511) [LSB], 带偏移的 512[LSB])
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9 ~ 0	PBRT_R[9:0]	512	R/W	R 信号的亮度 (DC) 调整 无符号 (0 (-512) ~ 512 (0) ~ 1023 (+511) [LSB], 带偏移的 512[LSB])

【注】 在全部输出控制部的寄存器更新控制寄存器 (OUT_UPDATE) 的 OUTCNT_VEN 位为“1”时更新此寄存器。

36.2.53 对比度 (增益) 校正寄存器 (OUT_CONTRAST)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CONT_G[7:0]							
初始值:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CONT_B[7:0]							CONT_R[7:0]								
初始值:	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 24	—	全 0	R	保留位 读写值都为“0”。
23 ~ 16	CONT_G[7:0]	128	R/W	G 信号的对比度 (增益) 调整 0/128 ~ 255/128 (约 2 倍)
15 ~ 8	CONT_B[7:0]	128	R/W	B 信号的对比度 (增益) 调整 0/128 ~ 255/128 (约 2 倍)
7 ~ 0	CONT_R[7:0]	128	R/W	R 信号的对比度 (增益) 调整 0/128 ~ 255/128 (约 2 倍)

【注】 在全部输出控制部的寄存器更新控制寄存器 (OUT_UPDATE) 的 OUTCNT_VEN 位为“1”时更新此寄存器。

36.2.54 显示屏抖动寄存器（OUT_PDTHA）

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	PDTH_SEL[1:0]	—	—	—	PDTH_FORMAT [1:0]
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	PDTH_PA[1:0]	—	—	—	PDTH_PB[1:0]	—	—	—	PDTH_PC[1:0]	—	—	—	PDTH_PD[1:0]
初始值:	0	0	1	1	0	0	0	0	0	0	1	0	0	0	0	1
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 22	—	全 0	R	保留位 读写值都为“0”。
21、20	PDTH_SEL[1:0]	0	R/W	显示屏抖动运行模式 0: 舍去 1: 四舍五入 2: 2×2 图案抖动 3: 随机图案抖动
19、18	—	全 0	R	保留位 读写值都为“0”。
17、16	PDTH_FORMAT [1:0]	0	R/W	显示屏抖动输出格式选择 0: RGB888 1: RGB666 2: RGB565 3: 禁止设定
15、14	—	全 0	R	保留位 读写值都为“0”。
13、12	PDTH_PA[1:0]	3	R/W	2×2 图案抖动的图案值（A） 无符号（0 ~ 3[LSB]）
11、10	—	全 0	R	保留位 读写值都为“0”。
9、8	PDTH_PB[1:0]	0	R/W	2×2 图案抖动的图案值（B） 无符号（0 ~ 3[LSB]）
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	PDTH_PC[1:0]	2	R/W	2×2 图案抖动的图案值（C） 无符号（0 ~ 3[LSB]）
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	PDTH_PD[1:0]	1	R/W	2×2 图案抖动的图案值（D） 无符号（0 ~ 3[LSB]）

【注】 在全部输出控制部的寄存器更新控制寄存器（OUT_UPDATE）的 OUTCNT_VEN 位为“1”时更新此寄存器。

36.2.55 输出相位控制寄存器 (OUT_CLK_PHASE)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	OUTCNT_FRONT_GAM	—	—	—	OUTCNT_LCD_EDGE	—	OUTCNT_STVA_EDGE	OUTCNT_STVB_EDGE	OUTCNT_STH_EDGE	OUTCNT_STB_EDGE	OUTCNT_CPV_EDGE	OUTCNT_POLA_EDGE	OUTCNT_POLB_EDGE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12	OUTCNT_FRONT_GAM	0	R/W	校正电路的顺序控制 0: 亮度 → 对比度 → 灰度校正 1: 灰度校正 → 亮度 → 对比度
11 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	OUTCNT_LCD_EDGE	0	R/W	LCD_DATA23 ~ 0 引脚的输出相位控制 0: 在 LCD_CLK 引脚的上升沿输出 1: 在 LCD_CLK 引脚的下降沿输出
7	—	0	R	保留位 读写值都为“0”。
6	OUTCNT_STVA_EDGE	0	R/W	LCD_TCON0 引脚的输出相位控制 0: 在 LCD_CLK 引脚的上升沿输出 1: 在 LCD_CLK 引脚的下降沿输出
5	OUTCNT_STVB_EDGE	0	R/W	LCD_TCON1 引脚的输出相位控制 0: 在 LCD_CLK 引脚的上升沿输出 1: 在 LCD_CLK 引脚的下降沿输出
4	OUTCNT_STH_EDGE	0	R/W	LCD_TCON2 引脚的输出相位控制 0: 在 LCD_CLK 引脚的上升沿输出 1: 在 LCD_CLK 引脚的下降沿输出
3	OUTCNT_STB_EDGE	0	R/W	LCD_TCON3 引脚的输出相位控制 0: 在 LCD_CLK 引脚的上升沿输出 1: 在 LCD_CLK 引脚的下降沿输出
2	OUTCNT_CPV_EDGE	0	R/W	LCD_TCON4 引脚的输出相位控制 0: 在 LCD_CLK 引脚的上升沿输出 1: 在 LCD_CLK 引脚的下降沿输出
1	OUTCNT_POLA_EDGE	0	R/W	LCD_TCON5 引脚的输出相位控制 0: 在 LCD_CLK 引脚的上升沿输出 1: 在 LCD_CLK 引脚的下降沿输出
0	OUTCNT_POLB_EDGE	0	R/W	LCD_TCON6 引脚的输出相位控制 0: 在 LCD_CLK 引脚的上升沿输出 1: 在 LCD_CLK 引脚的下降沿输出

【注】 在全部输出控制部的寄存器更新控制寄存器 (OUT_UPDATE) 的 OUTCNT_VEN 位为“1”时更新此寄存器

36.3 使用方法

36.3.1 灰度校正的调整方法

测量连接各显示屏的 G/B/R 特性，设定适合显示屏的灰度校正。
因为灰度校正的调整取决于显示屏的特性，所以没有推荐设定值。

36.3.2 抖动的使用方法

在显示画面出现假轮廓时使用此方法。

表 36.26 抖动的设定

位名	设定值
PDTH_FORMAT[1:0]	设定格式。 RGB888: 0 RGB666: 1 RGB565: 2
PDTH_SEL[1:0]	使用 2×2 图案抖动: 2
PDTH_PA[1:0]	通常用作初始值: 3
PDTH_PB[1:0]	通常用作初始值: 0
PDTH_PC[1:0]	通常用作初始值: 2
PDTH_PD[1:0]	通常用作初始值: 1

36.3.3 输出格式的调整方法

典型输出格式的设定例子如下所示。

各输出格式的同步设定需要和缩放后的输出设定相同。

表 36.27 同步信号的设定例子

寄存器名	位名	VGA	SVGA	说明
TCON_TIM	TCON_HALF[10:0]	400	528	以时钟为单位设定 1H 期间的 1/2 值。
垂直同步信号				
TCON_TIM_STVA1	TCON_STVA_VS[10:0]	0	0	设定从内部垂直同步信号上升沿开始的脉冲生成起始位置。 将 1/2H 期间的值设定为“1”。
TCON_TIM_STVA1	TCON_STVA_VW[10:0]	4	8	设定从上述脉冲生成起始位置开始变化的指针。 将 1/2H 期间的值设定为“1”。
TCON_TIM_STVA2	TCON_STVA_INV	0	0	设定上述脉冲的输出极性。在反相输出的情况下，设定为“1”。
TCON_TIM_STVA2	TCON_STVA_SEL[2:0]	0	0	在选择 STVA 输出的情况下，设定为“0”。
垂直允许信号				
TCON_TIM_STVB1	TCON_STVB_VS[10:0]	68	44	设定从内部垂直同步信号的上升沿开始的生成脉冲起始位置。 将 1/2H 期间的值设定为“1”。
TCON_TIM_STVB1	TCON_STVB_VW[10:0]	960	1200	设定从上述脉冲生成起始位置开始变化的指针。 将 1/2H 期间的值设定为“1”。
TCON_TIM_STVB2	TCON_STVB_INV	0	0	设定上述脉冲的输出极性。在反相输出的情况下，设定为“1”。
TCON_TIM_STVB2	TCON_STVB_SEL[2:0]	1	1	在选择 STVB 输出的情况下，设定为“1”。
水平同步信号				
TCON_TIM_STH1	TCON_STH_HS[10:0]	0	0	设定从内部水平同步信号的上升沿开始的脉冲生成起始位置。
TCON_TIM_STH1	TCON_STH_HW[10:0]	96	128	设定从上述脉冲生成起始位置开始变化的指针。
TCON_TIM_STH2	TCON_STH_INV	0	0	设定上述脉冲的输出极性。在反相输出的情况下，设定为“1”。
TCON_TIM_STH2	TCON_STH_SEL[2:0]	2	2	在选择 STH 输出的情况下，设定为“2”。
水平允许信号				
TCON_TIM_STB1	TCON_STB_HS[10:0]	128	192	设定从内部水平同步信号的上升沿开始的脉冲生成起始位置。
TCON_TIM_STB1	TCON_STB_HW[10:0]	640	800	设定从上述脉冲生成起始位置开始变化的指针。
TCON_TIM_STB2	TCON_STB_INV	0	0	设定上述脉冲的输出极性。在反相输出的情况下，设定为“1”。
TCON_TIM_STB2	TCON_STB_SEL[2:0]	3	3	在选择 STB 输出的情况下，设定为“3”。

表 36.28 数据的设定例子

寄存器名	位名	RGB888	串行 RGB (3 倍速)	说明
OUT_SET	OUT_ENDIAN_ON	0	0	在更改位序时, 设定为“1”。
OUT_SET	OUT_SWAP_ON	0	0	在进行 B/R 替换时, 设定为“1”。
OUT_SET	OUT_PIXEL_INV_ON	0	0	在使用减轻同时变化数的功能时, 设定为“1”。
OUT_SET	OUT_SUM_MOVE[4:0]	31	31	此寄存器在 OUT_PIXEL_INV_ON 为“1”时有效。设定同时变化数的阈值。
OUT_SET	OUT_FORMAT[1:0]	0	3	输出格式设定 RGB888: 0 RGB666: 1 RGB565: 2 串行 RGB: 3
OUT_SET	OUT_FRQ_SEL[1:0]	0	1	输出时钟控制 RGB888、RGB666、RGB565: 0 串行 RGB3 倍速输出: 1 串行 RGB4 倍速输出: 2
OUT_SET	OUT_DIR_SEL	0	0	在颠倒串行 RGB 输出顺序时, 设定为“1”。
OUT_SET	OUT_PHASE[1:0]	0	0	在串行 RGB 的输出相位发生偏差时进行设定。 无延迟: 0 延迟 1 个时钟: 1 延迟 2 个时钟: 2 延迟 3 个时钟: 3 (只支持 4 倍速)

37. 视频显示控制器 4 (7) 系统控制部

37.1 系统控制功能

37.1.1 功能概要

系统控制部有中断控制、显示屏时钟控制和 CLUT 表的读选择信号状态标志输出功能。

37.1.2 中断控制

如表 37.1 所示，中断信号有 9 个从缩放部和图像合成部输出的信号。通过中断控制功能控制是否将这些信号输出到外部。

在接受中断信号时，将 INT_STA* 位置“1”。但是，即使在写“1”后读 INT_STA* 位，在接受中断信号前读取值也为“0”。一旦接受了中断信号，就从 INT_STA* 位读“1”。

在清除已接受的中断信号时，将 INT_STA* 位置“0”。

如果在清除中断信号后再次接受中断信号，就将 INT_STA* 位置“1”。

表 37.1 中断信号

信号名	功能
INT0	缩放前的垂直同步信号
INT1	缩放后的垂直同步信号
INT2	缩放的垂直同步信号的欠缺信号
INT3	显示屏输出的指定行信号
INT4	录像功能的场结束信号
INT5	帧缓冲器写上溢信号
INT6	图形 (1) 的帧缓冲器读下溢信号
INT7	图形 (2) 的帧缓冲器读下溢信号
INT8	图形 (3) 的帧缓冲器读下溢信号

表 37.2 中断的清除 / 保持设定

寄存器名	位名	初始值	说明
SYSCNT_INT2	INT_STA0	0	INT0 中断的清除 / 保持设定 0(W): 清除中断状态 1(W): 开始接受中断 0(R): 没有中断 1(R): 有中断
SYSCNT_INT2	INT_STA1	0	INT1 中断的清除 / 保持设定 0(W): 清除中断状态 1(W): 开始接受中断 0(R): 没有中断 1(R): 有中断
SYSCNT_INT2	INT_STA2	0	INT2 中断的清除 / 保持设定 0(W): 清除中断状态 1(W): 开始接受中断 0(R): 没有中断 1(R): 有中断
SYSCNT_INT2	INT_STA3	0	INT3 中断清除 / 保持设定 0(W): 清除中断状态 1(W): 开始接受中断 0(R): 没有中断 1(R): 有中断
SYSCNT_INT2	INT_STA4	0	INT4 中断清除 / 保持设定 0(W): 清除中断状态 1(W): 开始接受中断 0(R): 没有中断 1(R): 有中断
SYSCNT_INT2	INT_STA5	0	INT5 中断的清除 / 保持设定 0(W): 清除中断状态 1(W): 开始接受中断 0(R): 没有中断 1(R): 有中断
SYSCNT_INT2	INT_STA6	0	INT6 中断的清除 / 保持设定 0(W): 清除中断状态 1(W): 开始接受中断 0(R): 没有中断 1(R): 有中断
SYSCNT_INT2	INT_STA7	0	INT7 中断的清除 / 保持设定 0(W): 清除中断状态 1(W): 开始接受中断 0(R): 没有中断 1(R): 有中断
SYSCNT_INT1	INT_STA8	0	INT8 中断的清除 / 保持设定 0(W): 清除中断状态 1(W): 开始接受中断 0(R): 没有中断 1(R): 有中断

表 37.3 中断输出的 ON/OFF 设定

寄存器名	位名	初始值	说明
SYSCNT_INT4	INT_OUT0_ON	0	INT0 中断输出的 ON/OFF 设定 0: OFF 1: ON
SYSCNT_INT4	INT_OUT1_ON	0	INT1 中断输出的 ON/OFF 设定 0: OFF 1: ON
SYSCNT_INT4	INT_OUT2_ON	0	INT2 中断输出的 ON/OFF 设定 0: OFF 1: ON
SYSCNT_INT4	INT_OUT3_ON	0	INT3 中断输出的 ON/OFF 设定 0: OFF 1: ON
SYSCNT_INT4	INT_OUT4_ON	0	INT4 中断输出的 ON/OFF 设定 0: OFF 1: ON
SYSCNT_INT4	INT_OUT5_ON	0	INT5 中断输出的 ON/OFF 设定 0: OFF 1: ON
SYSCNT_INT4	INT_OUT6_ON	0	INT6 中断输出的 ON/OFF 设定 0: OFF 1: ON
SYSCNT_INT4	INT_OUT7_ON	0	INT7 中断输出的 ON/OFF 设定 0: OFF 1: ON
SYSCNT_INT3	INT_OUT8_ON	0	INT8 中断输出的 ON/OFF 设定 0: OFF 1: ON

37.1.3 显示屏时钟控制

此模块可选择图像时钟、外部时钟或者外围时钟 1 作为显示屏时钟的供给源，而且还内置 1/1 ~ 1/32 的分频器。通过表 37.4 所示的寄存器控制显示屏时钟。

表 37.4 显示屏时钟控制

寄存器名	位名	初始值	说明
SYSCNT_PANEL_CLK	PANEL_ICKSEL[1:0]	0	显示屏时钟的供给源选择 0: 选择图像时钟 (当 INP_SEL 位为“0”时, 选择 VIDEO_X1; 当 INP_SEL 位为“1”时, 选择 DV_CLK。) 1: 选择外部时钟 (LCD_EXTCLK) 2: 选择外围总线时钟 1 (P1φ) 3: 禁止设定
SYSCNT_PANEL_CLK	PANEL_ICKEN	0	显示屏时钟的运行允许设定 0: 禁止显示屏时钟运行块的运行 1: 允许显示屏时钟运行块的运行 【注】 必须在将此位置“0”后更改 PANEL_ICKSEL 位和 PANEL_DCDR 位。
SYSCNT_PANEL_CLK	PANEL_DCDR[5:0]	1	时钟分频比设定 设定的详细内容请参照表 37.5。 【注】 禁止表 37.5 以外的设定。

表 37.5 输入 / 输出时钟频率和分频比

DCDR[5:0]	时钟分频比	输入 / 输出时钟频率 (MHz)		
		27.00	54.00	66.67*2
000001*1	1/1	27.00	54.00	66.67
000010	1/2	13.50	27.00	33.33
000011	1/3	9.00	18.00	22.22
000100	1/4	6.75	13.50	16.67
000101	1/5	5.40	10.80	13.33
000110	1/6	4.50	9.00	11.11
000111	1/7	3.86	7.71	9.52
001000	1/8	3.38	6.75	8.33
001001	1/9	3.00	6.00	7.41
001100	1/12	2.25	4.50	5.56
010000	1/16	1.69	3.38	4.17
011000	1/24	1.13	2.25	2.78
100000	1/32	0.84	1.69	2.08

【注】 *1 如果将显示屏时钟的供给源设定为外围时钟 1 (P1φ), 就禁止设定。

*2 这是将显示屏时钟的供给源设定为外部时钟 (LCD_EXTCLK) 或者外围时钟 1 (P1φ) 的情况。

37.1.4 CLUT 表的读选择信号状态标志

如表 37.6 所示，能读取 CLUT 表的读选择信号状态标志。

表 37.6 CLUT 表的读选择信号状态标志

寄存器名	位名	初始值	说明
SYSCNT_CLUT	GR1_CLT_SEL_ST	—	图形 (1) 的 CLUT 表的读选择信号状态标志 0: 读 CLUT 表 0 1: 读 CLUT 表 1
SYSCNT_CLUT	GR2_CLT_SEL_ST	—	图形 (2) 的 CLUT 表的读选择信号状态标志 0: 读 CLUT 表 0 1: 读 CLUT 表 1
SYSCNT_CLUT	GR3_CLT_SEL_ST	—	图形 (3) 的 CLUT 表的读选择信号状态标志 0: 读 CLUT 表 0 1: 读 CLUT 表 1

37.2 寄存器说明

以下寄存器组分配在 SH 寄存器映像空间。

【说明寄存器时的符号】

初始值 : 复位后的寄存器值

— : 不定值

R/W : 可读写。能读到写入值。

R/WC0 : 可读写。如果写“0”，该位就被初始化，但是忽视写“1”。

R/WC1 : 可读写。如果写“1”，该位就被初始化，但是忽视写“0”。

R : 只能读。写入值总是“0”。

—/W : 只能写。读取值为不定值。

表 37.7 系统控制部的寄存器结构

名称	略称	R/W	初始值	地址	存取长度
中断控制寄存器 1	SYSCNT_INT1	R/W	H'0000 0000	H'FFFF 7A80	32/16
中断控制寄存器 2	SYSCNT_INT2	R/W	H'0000 0000	H'FFFF 7A84	32/16
中断控制寄存器 3	SYSCNT_INT3	R/W	H'0000 0000	H'FFFF 7A88	32/16
中断控制寄存器 4	SYSCNT_INT4	R/W	H'0000 0000	H'FFFF 7A8C	32/16
显示屏时钟控制寄存器	SYSCNT_PANEL_CLK	R/W	H'0001	H'FFFF 7A90	16
CLUT 表的读选择信号状态标志寄存器	SYSCNT_CLUT	R	H'0000	H'FFFF 7A92	16

37.2.1 中断控制寄存器 1 (SYSCNT_INT1)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	INT STA8
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	INT_STA8	0	R/W	INT8 中断的清除 / 保持设定 0(W): 清除中断状态 1(W): 开始接受中断 0(R): 没有中断 1(R): 有中断

37.2.2 中断控制寄存器 2 (SYSCNT_INT2)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	INT_STA7	—	—	—	INT_STA6	—	—	—	INT_STA5	—	—	—	INT_STA4
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	INT_STA3	—	—	—	INT_STA2	—	—	—	INT_STA1	—	—	—	INT_STA0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 29	—	全 0	R	保留位 读写值都为“0”。
28	INT_STA7	0	R/W	INT7 中断的清除 / 保持设定 0(W): 清除中断状态 1(W): 开始接受中断 0(R): 没有中断 1(R): 有中断
27 ~ 25	—	全 0	R	保留位 读写值都为“0”。
24	INT_STA6	0	R/W	INT6 中断的清除 / 保持设定 0(W): 清除中断状态 1(W): 开始接受中断 0(R): 没有中断 1(R): 有中断
23 ~ 21	—	全 0	R	保留位 读写值都为“0”。
20	INT_STA5	0	R/W	INT5 中断的清除 / 保持设定 0(W): 清除中断状态 1(W): 开始接受中断 0(R): 没有中断 1(R): 有中断
19 ~ 17	—	全 0	R	保留位 读写值都为“0”。
16	INT_STA4	0	R/W	INT4 中断的清除 / 保持设定 0(W): 清除中断状态 1(W): 开始接受中断 0(R): 没有中断 1(R): 有中断
15 ~ 13	—	全 0	R	保留位 读写值都为“0”。

位	位名	初始值	R/W	说明
12	INT_STA3	0	R/W	INT3 中断的清除 / 保持设定 0(W): 清除中断状态 1(W): 开始接受中断 0(R): 没有中断 1(R): 有中断
11 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	INT_STA2	0	R/W	INT2 中断清除 / 保持设定 0(W): 清除中断状态 1(W): 开始接受中断 0(R): 没有中断 1(R): 有中断
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	INT_STA1	0	R/W	INT1 中断的清除 / 保持设定 0(W): 清除中断状态 1(W): 开始接受中断 0(R): 没有中断 1(R): 有中断
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	INT_STA0	0	R/W	INT0 中断清除 / 保持设定 0(W): 清除中断状态 1(W): 开始接受中断 0(R): 没有中断 1(R): 有中断

37.2.3 中断控制寄存器 3 (SYSCNT_INT3)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	INT_OUT8_ON
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	INT_OUT8_ON	0	R/W	INT8 中断输出的 ON/OFF 设定 0: OFF 1: ON

37.2.4 中断控制寄存器 4 (SYSCNT_INT4)

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	INT_OUT7_ON	—	—	—	INT_OUT6_ON	—	—	—	INT_OUT5_ON	—	—	—	INT_OUT4_ON
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	INT_OUT3_ON	—	—	—	INT_OUT2_ON	—	—	—	INT_OUT1_ON	—	—	—	INT_OUT0_ON
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 29	—	全 0	R	保留位 读写值都为“0”。
28	INT_OUT7_ON	0	R/W	INT7 中断输出的 ON/OFF 设定 0: OFF 1: ON
27 ~ 25	—	全 0	R	保留位 读写值都为“0”。
24	INT_OUT6_ON	0	R/W	INT6 中断输出的 ON/OFF 设定 0: OFF 1: ON
23 ~ 21	—	全 0	R	保留位 读写值都为“0”。
20	INT_OUT5_ON	0	R/W	INT5 中断输出的 ON/OFF 设定 0: OFF 1: ON
19 ~ 17	—	全 0	R	保留位 读写值都为“0”。
16	INT_OUT4_ON	0	R/W	INT4 中断输出的 ON/OFF 设定 0: OFF 1: ON
15 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12	INT_OUT3_ON	0	R/W	INT3 中断输出的 ON/OFF 设定 0: OFF 1: ON
11 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	INT_OUT2_ON	0	R/W	INT2 中断输出的 ON/OFF 设定 0: OFF 1: ON
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。

位	位名	初始值	R/W	说明
4	INT_OUT1_ON	0	R/W	INT1 中断输出的 ON/OFF 设定 0: OFF 1: ON
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	INT_OUT0_ON	0	R/W	INT0 中断输出的 ON/OFF 设定 0: OFF 1: ON

37.2.5 显示屏时钟控制寄存器 (SYSCNT_PANEL_CLK)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PANEL_ICKSEL[1:0]	—	—	—	PANEL_ICKEN	—	—	PANEL_DCDR[5:0]						
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R/W	R/W	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13、12	PANEL_ICKSEL [1:0]	0	R/W	显示屏时钟的供给源选择 0: 选择图像时钟 (当 INP_SEL 位为“0”时, 选择 VIDEO_X1 ; 当 INP_SEL 位为“1”时, 选择 DV_CLK。) 1: 选择外部时钟 (LCD_EXTCLK) 2: 选择外围总线时钟 1 (P1 ϕ) 3: 禁止设定
11 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	PANEL_ICKEN	0	R/W	显示屏时钟的运行允许设定 0: 禁止显示屏时钟运行块的运行 1: 允许显示屏时钟运行块的运行 【注】 必须在将此位置“0”后更改 PANEL_ICKSEL 位和 PANEL_DCDR 位。
7、6	—	全 0	R	保留位 读写值都为“0”。
5 ~ 0	PANEL_DCDR [5:0]	1	R/W	时钟频比设定 设定的详细内容请参照表 37.5。 【注】 禁止表 37.5 以外的设定。

37.2.6 CLUT 表的读选择信号状态标志寄存器 (SYSCNT_CLUT)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	GR3_CLT _SEL_ST	—	—	—	GR2_CLT _SEL_ST	—	—	—	GR1_CLT _SEL_ST
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	GR3_CLT_SEL_ST	—	R	图形 (3) 的 CLUT 表的读选择信号状态标志 0: 读 CLUT 表 0 1: 读 CLUT 表 1
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	GR2_CLT_SEL_ST	—	R	图形 (2) 的 CLUT 表的读选择信号状态标志 0: 读 CLUT 表 0 1: 读 CLUT 表 1
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GR1_CLT_SEL_ST	—	R	图形 (1) 的 CLUT 表的读选择信号状态标志 0: 读 CLUT 表 0 1: 读 CLUT 表 1

38. 失真校正引擎

本章在缔结保密合同后予以公开。
详细内容请向本公司的销售员确认。

39. 显示输出比较单元 (DISCOM)

显示输出比较单元能检查从图形显示模块 * 输出的数据是否为期待的图形数据。此检查是预先算出所期待图形数据的 CRC 码，然后将其和从图形显示模块输出的数据 CRC 码进行比较。

【注】 * 在本 LSI 中，图形显示模块是指视频显示控制器 4。

39.1 特点

1. 图形显示模块的图形面比较
能从图形显示模块的图形面中选择 1 面进行 CRC 码比较。
2. α 混合后的数据比较
能和图形显示模块的 α 混合后的图形数据进行 CRC 码比较。
3. 矩形区域的指定
能和从图形显示模块的输出图形数据（图形面或者 α 混合后）中指定的矩形区域进行 CRC 码比较。
4. 像素格式
像素格式可选择 32bit/pixel 或者 16bit/pixel。
在 32bit/pixel 时为 ARGB8888/RGB888 格式，在 16bit/pixel 时为 RGB565 格式。

39.2 框图

此模块的整体框图如下所示。

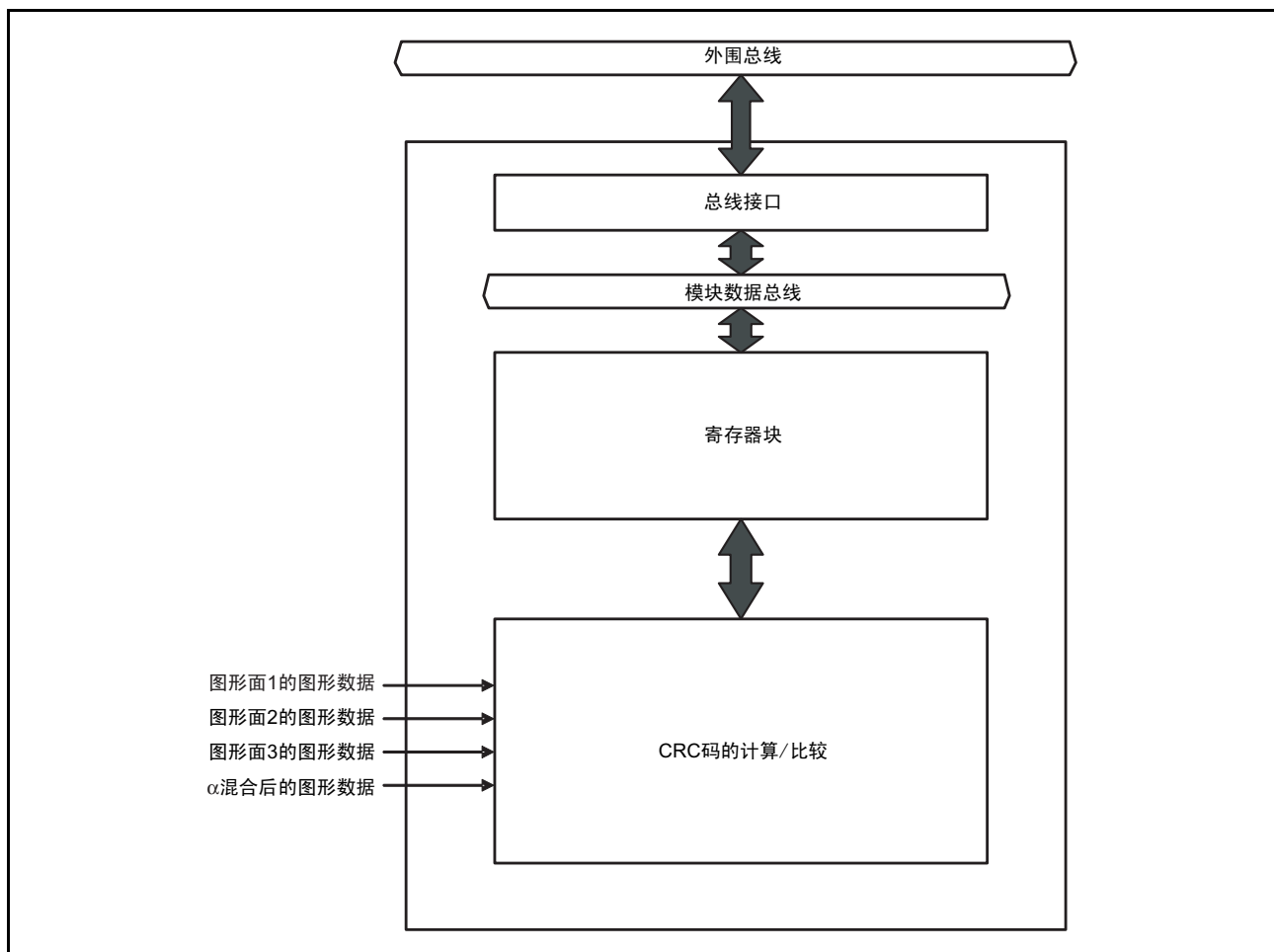


图 39.1 框图

39.3 寄存器说明

寄存器结构如下所示。

表 39.1 寄存器结构

寄存器名	略称	R/W	地址	存取长度
控制寄存器	DOCMCR	R/W	H'FFFA800	32
状态寄存器	DOCMSTR	R	H'FFFA804	32
状态清除寄存器	DOCMCLSTR	R/W	H'FFFA808	32
中断允许寄存器	DOCMENR	R/W	H'FFFA80C	32
工作参数设定寄存器	DOCMPMR	R/W	H'FFFA814	32
CRC 码期待值寄存器	DOCMCCR	R/W	H'FFFA818	32
CRC 码计算值寄存器	DOCMCCR	R	H'FFFA81C	32
水平方向起始位置设定寄存器	DOCMSPXR	R/W	H'FFFA820	32
垂直方向起始位置设定寄存器	DOCMSPYR	R/W	H'FFFA824	32
水平方向尺寸寄存器	DOCMSZXR	R/W	H'FFFA828	32
垂直方向尺寸寄存器	DOCMSZYR	R/W	H'FFFA82C	32
CRC 码初始值寄存器	DOCMCIR	R/W	H'FFFA830	32

39.3.1 控制寄存器 (DOCMCR)

此寄存器是对 CRC 码比较进行 ON/OFF 控制的寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CMPRU
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CMPR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 17	—	全 0	R	保留位 读写值都为“0”。
16	CMPRU	0	R	显示输出比较执行的更新值 此值反映 CMPR 位的内部更新值。 有关 DOCMCR 的 CMPR 位以及 DOCMCLSTR 和 DOCMIENR 除外的寄存器更新, 必须在确认此位为“0”后进行更新, 详细内容请参照“39.4.8 寄存器的更新时序”。
15 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	CMPR	0	R/W	显示输出的比较执行 进行显示输出的比较。 在检测到图形数据有效期间的开始后, 将此位装入到内部, 详细内容请参照“39.4.8 寄存器的更新时序”。 0: 停止显示输出比较 1: 进行显示输出比较

39.3.2 状态寄存器 (DOCMSTR)

此寄存器是返回 CRC 码比较结果的寄存器。在检测到图形数据有效期间的结束后，将比较结果反映到寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CMPST
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31、30	—	全 0	R	保留位 读取值为不定值，写入值总是“0”。
29 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	CMPST	0	R	显示输出比较状态 0: 在显示输出比较中 CRC 码相同 1: 在显示输出比较中 CRC 码不同

39.3.3 状态清除寄存器 (DOCMCLSTR)

能通过给此寄存器的 CMPCLST 位写“1”，清除 DOCMSTR 的 CMPST 位。

但是，从给 CMPCLST 位写“1”到清除 DOCMSTR 的 CMPST 位为止需要一定的时间。

因此，必须在给 CMPCLST 位写“1”后确认 DOCMSTR 的 CMPST 位已被清除。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CMP CLST
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	CMPCLST	0	R/W	显示输出比较状态的清除 通过给此位写“1”，将 DOCMSTR 的 CMPST 位清“0”。 读取值总是“0”。

39.3.4 中断允许寄存器 (DOCMIENR)

此寄存器允许 DOCMSTR 对应的各状态位的中断。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CMPIEN
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	CMPIEN	0	R/W	显示输出比较差异检测中断的允许 当 DOCMSTR 的 CMPST 位为“1”时，允许或者禁止显示输出比较差异检测中断 (CMPI)。 0: 禁止显示输出比较差异检测中断 (CMPI) 1: 允许显示输出比较差异检测中断 (CMPI)

39.3.5 工作参数设定寄存器 (DOCMPMR)

能通过此寄存器选择图形数据以及设定像素格式。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CMPBT
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMPDFA[7:0]							CMPDAUF	—	—	—	CMPSELP[3:0]				
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 17	—	全 0	R	保留位 读写值都为“0”。
16	CMPBT	0	R/W	像素格式的数据宽度 指定像素格式的数据宽度。 0: 32bit/pixel (ARGB8888/RGB888 格式) 1: 16bit/pixel (RGB565 格式)
15 ~ 8	CMPDFA [7:0]	H'00	R/W	显示输出比较的默认 α 值 设定默认 α 值。这些位在 CMPDAUF 位为“1”时有效。
7	CMPDAUF	0	R/W	显示输出比较的默认 α 值使用 使用默认 α 值。 0: 不使用默认 α 值 1: 使用默认 α 值 【注】 此位只在选择 RGB888 格式时有效。在选择 ARGB8888/RGB565 格式时, 必须总是将此位置“0”。 在选择上述像素格式时, 此位为“0”时的 α 值是从图形显示模块输出的数据 α 值。
6 ~ 4	—	全 0	R	保留位 读写值都为“0”。
3 ~ 0	CMPSELP [3:0]	0000	R/W	显示输出比较的选择面 指定要进行 CRC 码比较的图形数据。 0: 不选择 1 ~ 3: 图形面 1 ~ 3 的图形数据 9: α 混合后的图形数据 上述以外: 禁止设定

39.3.6 CRC 码期待值寄存器 (DOCMECRCR)

给此寄存器指定所期待的图形数据的 CRC 码。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMPECRC[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMPECRC[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 0	CMPECRC [31:0]	H'00000 000	R/W	显示输出比较的 CRC 码期待值 所选图形数据或者矩形区域的 CRC 码期待值

39.3.7 CRC 码计算值寄存器 (DOCMCCRCR)

能从此寄存器读所选图形面或者矩形区域的 CRC 码计算结果。在检测到图形数据有效期间的结束后，将计算结果反映到寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMPCCRC[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMPCCRC[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 0	CMPCCRC [31:0]	H'00000 000	R	显示输出比较的 CRC 码计算值 所选图形数据或者矩形区域的 CRC 码计算值

39.3.8 水平方向起始位置设定寄存器 (DOCMSPXR)

给此寄存器指定 CRC 码计算对象矩形区域的水平方向起始位置。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CMPSPX[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	CMPSPX [10:0]	H'000	R/W	显示输出比较的水平方向起始位置 指定 CRC 码计算对象矩形区域的水平方向起始位置。 必须设定不超过图形数据水平方向尺寸的值。

39.3.9 垂直方向起始位置设定寄存器 (DOCMSPYR)

给此寄存器指定 CRC 码计算对象矩形区域的垂直方向起始位置。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CMPSPY[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	CMPSPY [10:0]	H'000	R/W	显示输出比较的垂直方向起始位置 指定 CRC 码计算对象矩形区域的垂直方向起始位置。 必须设定不超过图形数据垂直方向尺寸的值。

39.3.10 水平方向尺寸寄存器 (DOCMSZXR)

给此寄存器指定 CRC 码计算对象矩形区域的水平方向尺寸。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CMPSZX[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	CMPSZX [10:0]	H'000	R/W	显示输出比较的水平方向尺寸 指定 CRC 码计算对象矩形区域的水平方向尺寸。 设定值必须满足： 图形数据的水平方向尺寸 ≥ 水平方向起始位置 (CMPSPX) + 水平方向尺寸 (CMPSZX)

39.3.11 垂直方向尺寸寄存器 (DOCMSZYR)

给此寄存器指定 CRC 码计算对象矩形区域的垂直方向尺寸。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CMPSZY[10:0]										
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 0	CMPSZY [10:0]	H'000	R/W	显示输出比较的垂直方向尺寸 指定 CRC 码计算对象矩形区域的垂直方向尺寸。 设定值必须满足： 图形数据的垂直方向尺寸 ≥ 垂直方向起始位置 (CMPSPY) + 垂直方向尺寸 (CMPSZY)

39.3.12 显示输出比较的 CRC 初始值寄存器 (DOCMCRCIR)

给此寄存器指定 CRC 码的初始值。

	位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
		CRCINI[31:16]															
初始值:		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
	位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		CRCINI[15:0]															
初始值:		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 0	CRCINI[31:0]	H'FFFFFF FFF	R/W	显示输出比较的 CRC 初始值 指定所选图形数据或者矩形区域的 CRC 码初始值。

39.4 运行说明

39.4.1 运行概要

此模块具有计算任意图形数据的矩形区域 CRC 码的功能。能通过和预先计算得到的 CRC 码期待值进行比较, 检测是否能得到和期待值相同的显示输出。

此模块的主要特点如下:

- 可选择图形面 1 ~ 3。
- 可选择 α 混合后的图形数据。
- 能指定所选图形数据的任意矩形区域。
- 像素格式可选择 32bit/pixel (ARGB8888/RGB888) 或者 16bit/pixel (RGB565)。
- 在比较结果不同时发生中断。

39.4.2 系统结构

此模块的结构如下图所示。

在收到从图形显示模块输出的图形数据后计算 CRC 码, 然后和预先计算得到的 CRC 码期待值进行比较。图形数据能选择图形面 1 ~ 3 和 α 混合后的图形数据。

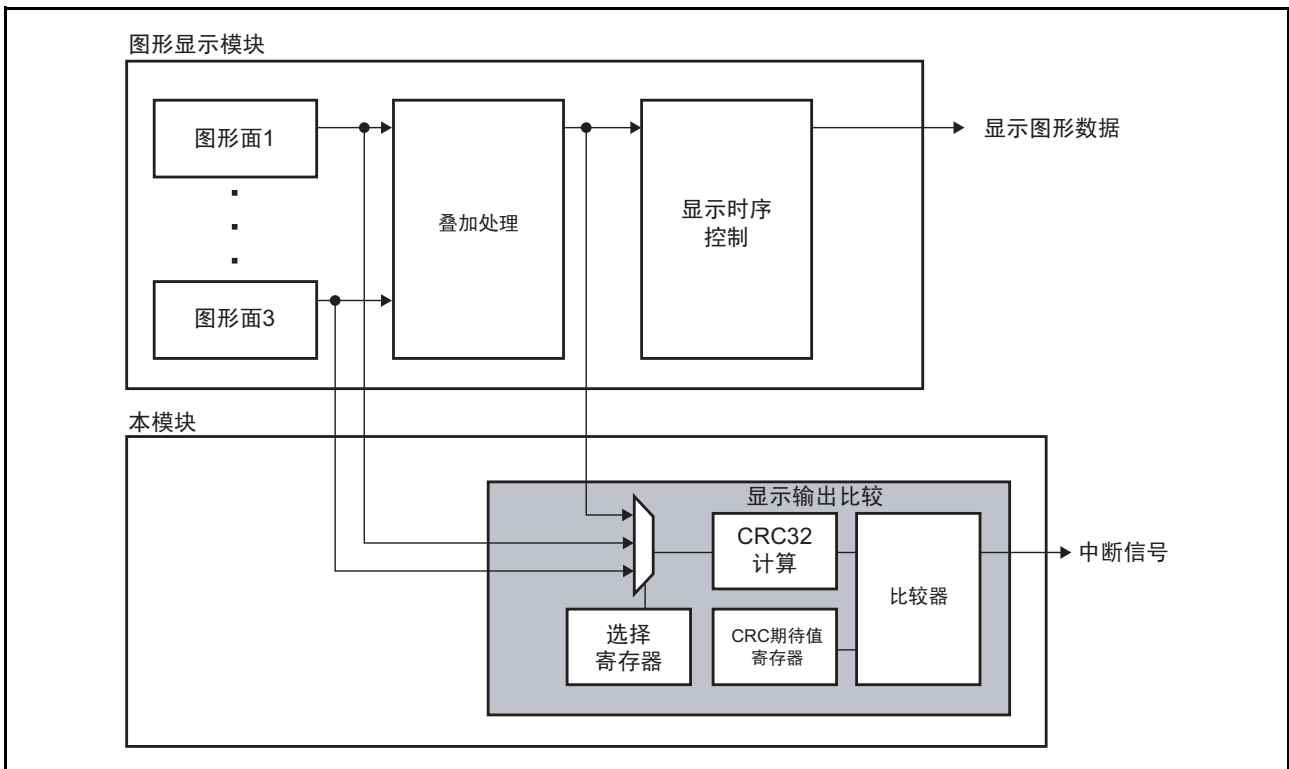


图 39.2 系统结构

39.4.3 CRC 计算方法

在显示输出比较单元中，用以下 CRC 多项式生成 32 位 CRC 码 (IEEE802.3)。

$$x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$$

以像素为单位计算 CRC (从 LSB 开始按顺序计算)。

即，在 32bit/pixel 的情况下，以 32 位为单位计算 CRC；在 16bit/pixel 的情况下，以 16 位为单位进行计算。作为计算 CRC 的数据顺序，在输出图形数据时 (左上 → 右下) 输入并且计算像素数据。

39.4.4 CRC 码生成图形数据的选择

对于计算 CRC 码的图形数据，能通过设定 DOCMPMR 的 CMPSELP[3:0] 位，选择图形面 1 ~ 3 或者 α 混合后的图形数据。

39.4.5 像素格式

(a) 像素格式的指定

通过 DOCMPMR 指定像素格式，像素格式一览表如下所示：

表 39.2 设定的像素格式

DOCMPMR 的设定位		CMDBT	CMPDFA	CMPDAUF[7:0]
32bit/pixel	ARGB8888	0	—	—
	RGB888		任意	任意
16bit/pixel	RGB565	1	—	—

(b) 对应像素格式的数据配置

各像素格式的数据配置如下所示：

- ARGB8888 (32bit/pixel)

b31	b24 b23	b16 b15	b8 b7	b0
α 8bit	Red 8bit	Green 8bit	Blue 8bit	

- RGB888 (32bit/pixel)

b31	b24 b23	b16 b15	b8 b7	b0
*	Red 8bit	Green 8bit	Blue 8bit	

【注】 * 当 CMPDFA 是 “0” 时，为从图形显示模块输出的 α 值。
当 CMPDFA 是 “1” 时，为 CMPDAUF[7:0] 指定的 α 值。

- RGB565 (16bit/pixel)

b15	b11 b10	b5 b4	b0
Red 5bit	Green 6bit	Blue 5bit	

39.4.6 矩形区域的设定

能通过寄存器设定所选图形数据的 CRC 码计算对象矩形区域的起始位置和尺寸。CRC 码计算对象矩形区域的图和寄存器设定如下所示。

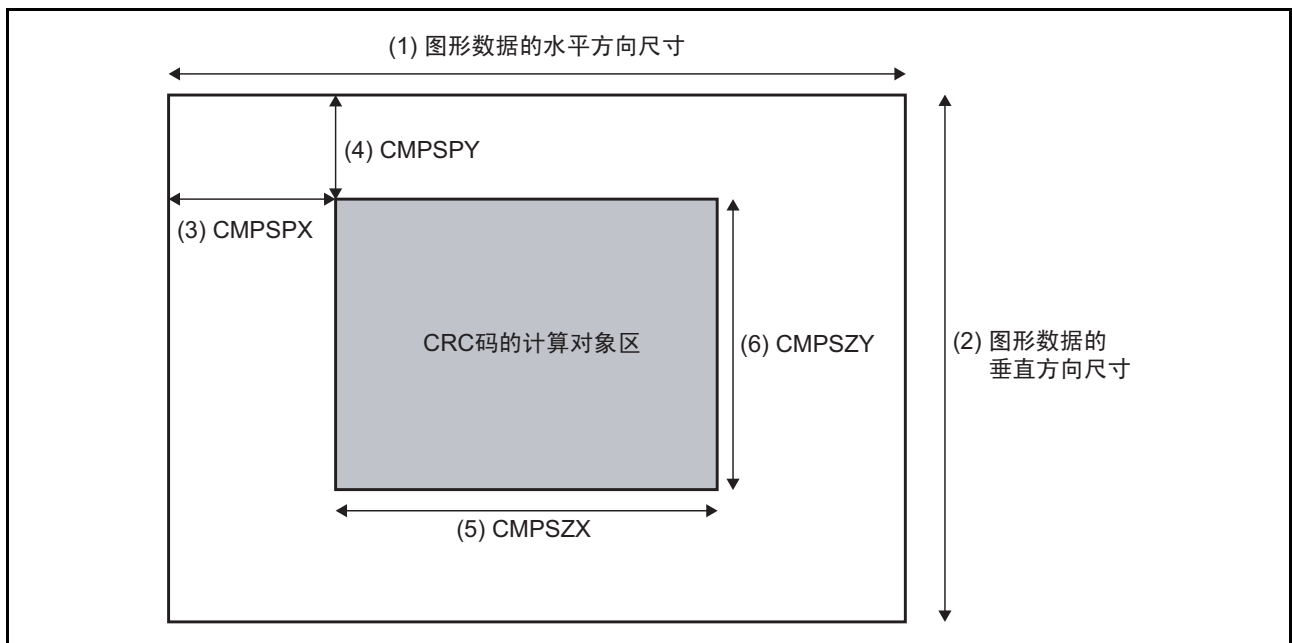


图 39.3 CRC 码计算对象矩形区域

表 39.3 CRC 码计算对象的设定

No	图中的名称	设定的寄存器名	说明
(1)	图形数据的水平尺寸		图形数据的水平尺寸 必须通过图形显示模块进行设定。
(2)	图形数据的垂直尺寸		图形数据的垂直尺寸 必须通过图形显示模块进行设定。
(3)	CMPSPX (水平起始位置)	DOCMSPXR	以图形数据的左上点为原点, 以像素为单位设定 CRC 码计算对象的水平位置。
(4)	CMPSPY (垂直起始位置)	DOCMSPYR	以图形数据的左上点为原点, 以行为单位设定 CRC 码计算对象的垂直位置。
(5)	CMPSZX (水平尺寸)	DOCMSZXR	以像素为单位设定 CRC 码计算对象的水平尺寸。 设定值必须满足: 图形数据的水平尺寸 \geq CMPSPX+CMPSZX
(6)	CMPSZY (垂直尺寸)	DOCMSZYR	以像素为单位设定 CRC 码计算对象的垂直尺寸。 设定值必须满足: 图形数据的垂直尺寸 \geq CMPSPY+CMPSZY

39.4.7 CRC 计算期间和比较时序

CRC 计算期间和期待值的比较时序图如下所示:

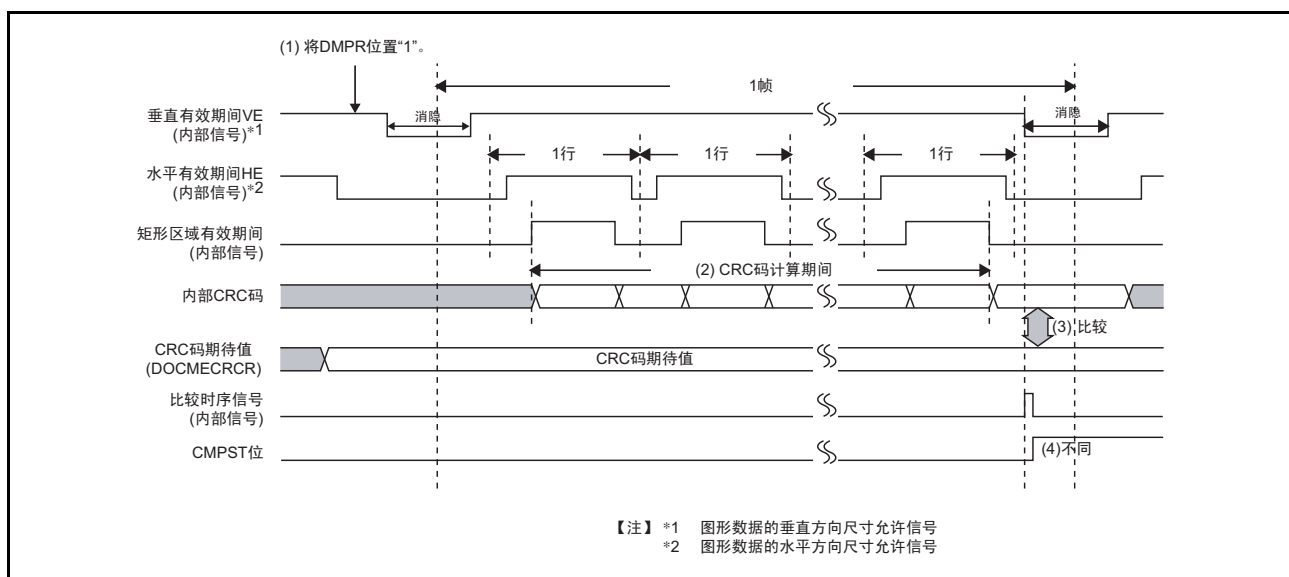


图 39.4 CRC 计算期间和比较时序例子

【运行说明】

- (1) 在将 DOCMSTR 的 DMPR 位置“1”后, 从下一帧开始运行。有关寄存器的更新时序, 请参照“39.4.8 寄存器的更新时序”。
- (2) 在设定的矩形区域中进行 CRC 码计算。
- (3) 在图形数据有效期间结束时, 将 CRC 码计算结果和 CRC 码期待值 (DOCMECR) 进行比较。
- (4) 如果比较结果不同, 就将 DOCMSTR 的 CMPST 位置位。

39.4.8 寄存器的更新时序

(1) 将寄存器的设定值装入到内部的时序

在写寄存器后，立即将 DOCMCR 的 CMPR 位除外的寄存器装入到内部。因此，如果在 DOCMCR 的 CMPRU 位为“1”时更新寄存器，就会得到意料不到的 CRC 码计算结果。对于影响 CRC 码计算的寄存器 (DOCMCLSTR 和 DOCMIENR 除外)，必须在确认 DOCMCR 的 CMPRU 位为“0”后进行更改。

在检测到图形数据有效期间的开始后，将 DOCMCR 的 CMPR 位装入到内部。因此，即使在帧的中途改写寄存器，也不影响该帧的 CRC 码计算。

(2) 将内部状态反映到寄存器的时序

在图形数据有效期间结束时，将内部状态反映到 DOCMSTR 和 DOCMCCRCR 寄存器。在图形数据有效期间开始时，将 CMPR 的内部状态反映到寄存器 DOCMCR 的 CMPRU 位。

寄存器的更新时序图如下所示：

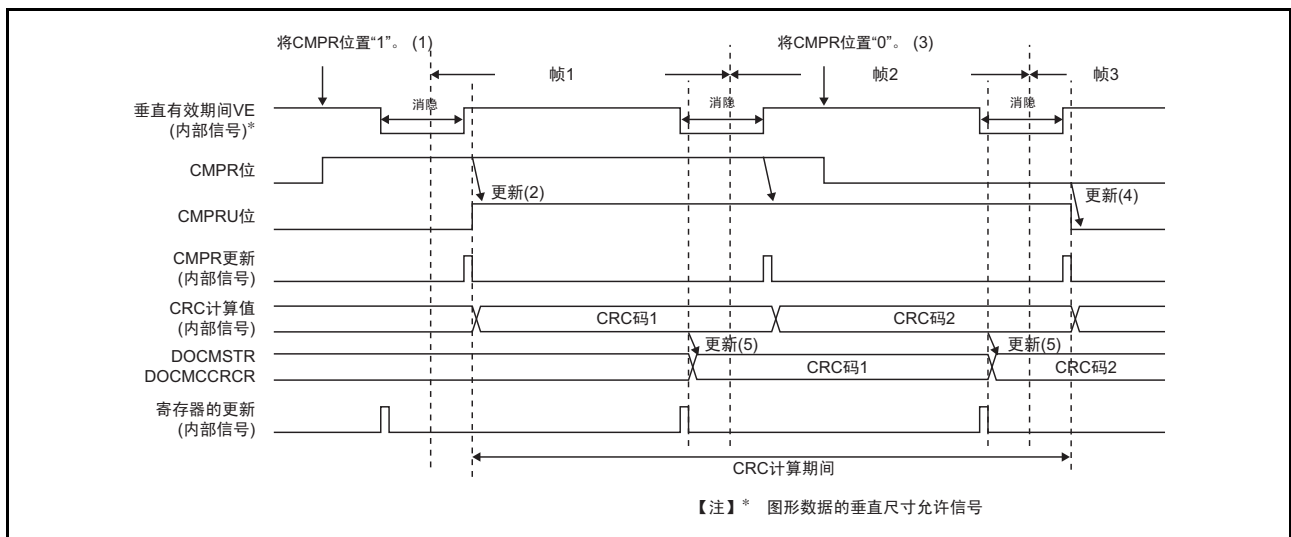


图 39.5 寄存器的更新时序图

【运行说明】

- (1) 将 CMPR 位置“1”，但是不立即将此位的设定值装入到内部。
- (2) 在检测到图形数据有效期间的开始后，先将 CMPR 位的设定值装入到内部，然后进行 CRC 计算。
- (3) 为了停止 CRC 比较，将 CMPR 位置“0”。和 (1) 一样不立即装入到内部。
- (4) 在检测到图形数据有效期间的开始后，先将 CMPR 位的设定值装入到内部，然后停止 CRC 计算。
- (5) 在图形数据有效期间结束时，将内部状态反映到寄存器。

39.4.9 运行流程

(a) 显示输出比较的开始流程

显示输出比较的开始流程例子如下所示：

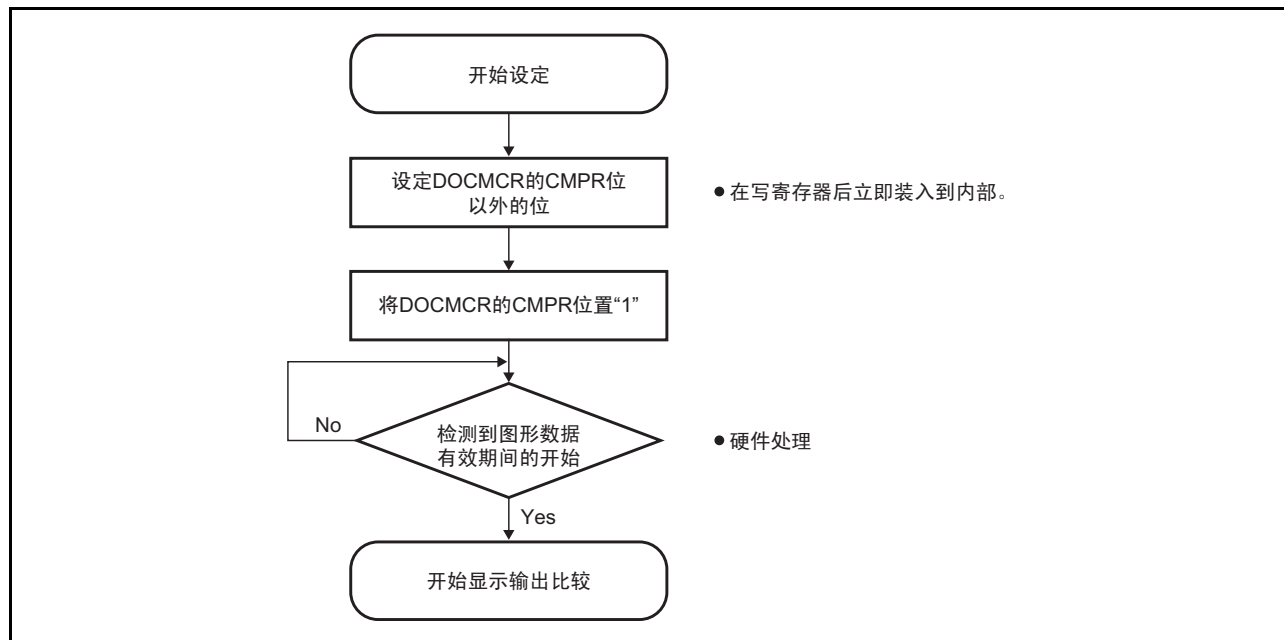


图 39.6 显示输出比较的开始流程例子

(b) 寄存器的设定变更流程

显示输出比较中的寄存器设定变更流程例子如下所示：

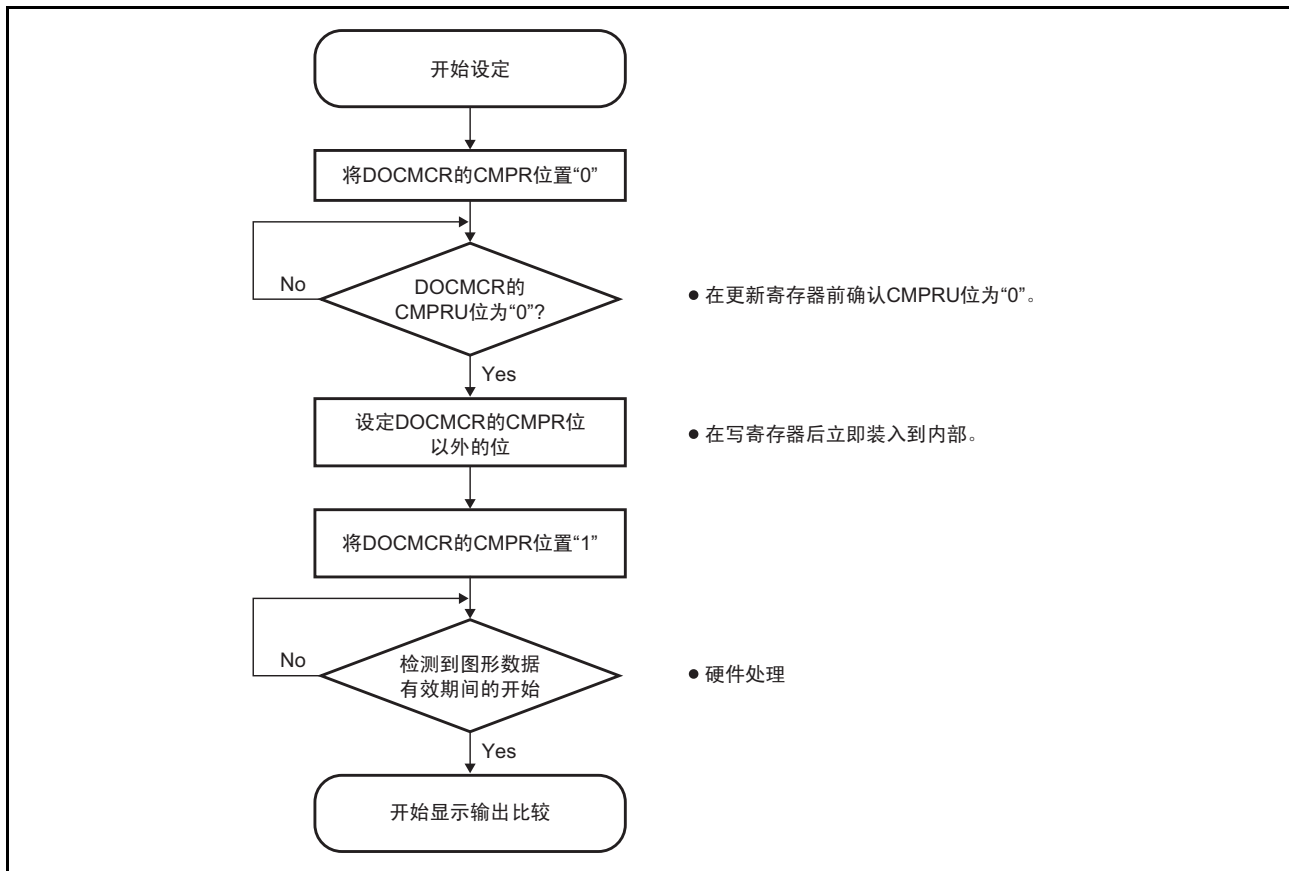


图 39.7 寄存器的设定变更流程例子

39.5 中断

中断源有“显示输出比较差异检测中断”。

如果将 DOCMENR 的 CMPPIEN 位置“1”，就在 DOCMSTR 的 CMPST 位被置“1”时产生中断请求。

必须在中断异常服务程序中清除 CMPST 位。从清除 CMPST 位到实际解除 CPU 中断源为止需要一定的时间。因此，为了避免再次误接收应该清除的中断源，必须在确认 2 次 CMPST 位已被清除后，执行 RTE 指令。

39.6 使用时的注意事项

39.6.1 CRC 期待值

必须通过软件，从要使用的图形数据算出选择图形面 1 ~ 3 时的 CRC 码期待值 (DOCMECR CR)。在选择 α 混合后的图形数据时，因计算误差而通过软件叠加图形数据的结果与通过图形显示模块叠加的结果有可能不同。因此在调试时，必须选择 α 混合后的图形数据，并且将读到的 CRC 码计算值 (DOCMCCR CR) 用作 CRC 码期待值。

39.6.2 放大控制功能

如果通过视频显示控制器 4 以及缩放处理的设定进行放大处理，此模块就不能生成 CRC 值。因此，在进行放大处理时，必须停止此模块。

40. OpenVG 瑞萨图形处理器

详细内容请向本公司的销售员确认。

41. JPEG 编解码器单元 (JCU)

JPEG 编解码器单元内置符合 JPEG 基准的压缩解压方式的 JPEG 编解码器，能高速处理图像数据的压缩和 JPEG 数据的解码。

41.1 特点

JPEG 编解码器单元有以下特点：

- 对应规格：JPEG 基准
本章记载的范围符合 JPEG 基准，但是不支持以下的基本特性：
 - 不支持 2 个成分的扫描。
 - 不支持多个成分的逐行扫描。
 - 运算精度：符合 JPEG Part2 和 ISO-IEC10918-2 的标准。
- 图像输入/输出方式：块隔行扫描方式
- 像素格式
 - 压缩：YCbCr422(H=2:1:1, V=1:1:1)
 - 解压：YCbCr422(H=2:1:1, V=1:1:1),
YCbCr420(H=2:1:1, V=2:1:1)
但是，输出到缓冲器的像素格式为 YCbCr422、ARGB8888 和 RGB565。
- 量化表：内置 4 张表。
- 霍夫曼表：内置 4 张表（2 张 AC 系数表，2 张 DC 系数表）。
- 对象标记：SOI (Start Of Image)、SOF0 (Start Of Frame Type 0)、SOS (Start Of Scan)、DQT (Define Quantization Tables)、DHT (Define Huffman Tables)、DRI (Define Restart Interval)、RSTm (Restart marks)、EOI (End Of Image)
- 图像数据速率：最大 133.34MB/s（以 66.67MHz 运行时）
- 在输入或者输出图像数据或者符号数据的情况下，每次进行设定行数或者数据量的传送都使用能暂时停止传送的模式来减小缓冲器容量。
- 处理单位：能以 8 字节为单位设定地址边界。
- 能处理的图像尺寸：能用 MCU (Minimum Coded Unit) 单位（在 YCbCr422 时为 16 像素×8 行，在 YCbCr420 时为 16 像素×16 行）整除的尺寸。

【注】 不能对不支持的像素格式和图像尺寸的图像进行压缩和解压处理。

框图如图 41.1 所示。

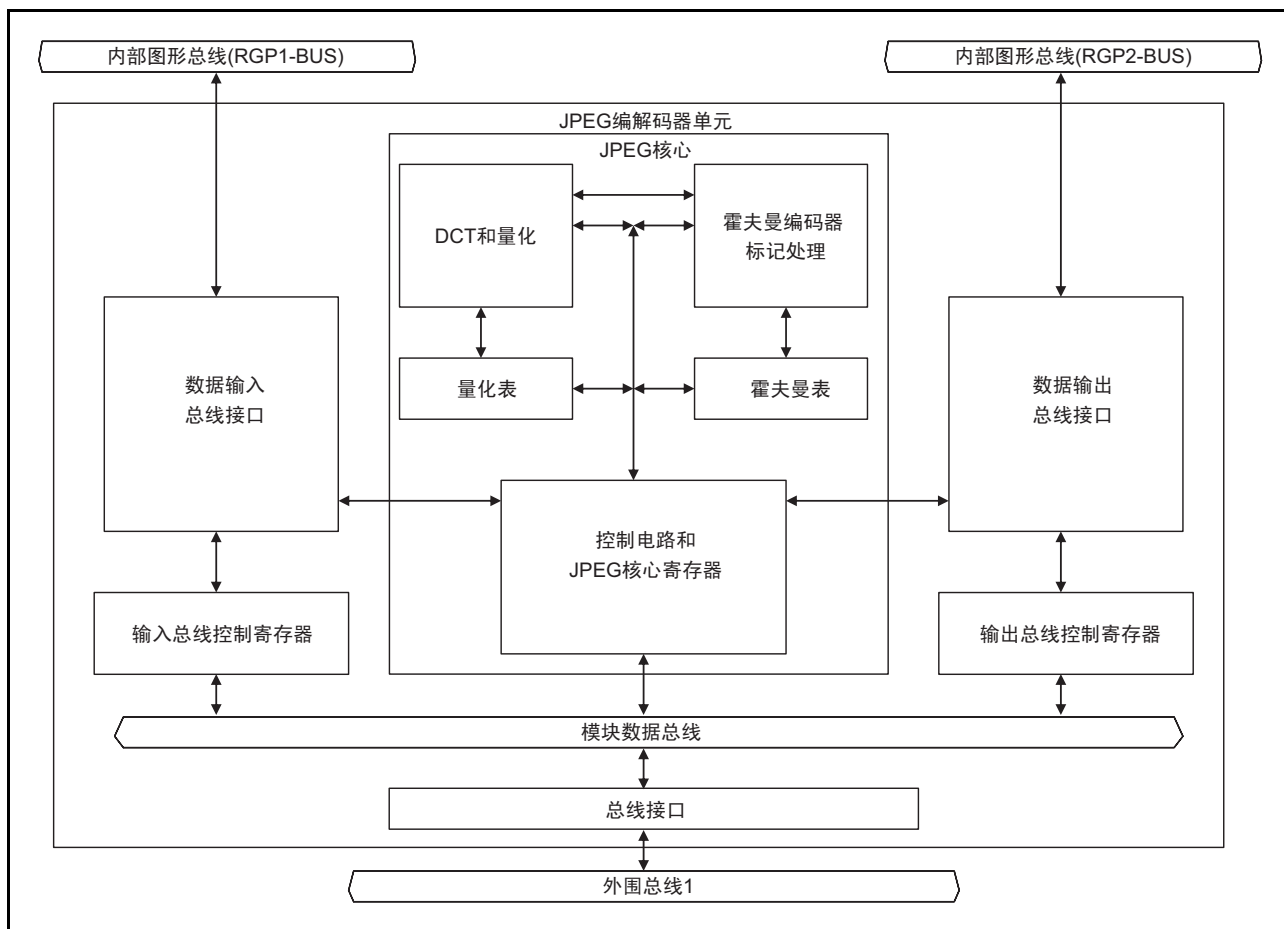


图 41.1 框图

41.2 寄存器说明

寄存器结构如表 41.1 所示。

表 41.1 寄存器结构

寄存器名称	略称	R/W	地址	存取长度
JPEG 码模式寄存器	JCMOD	R/W	H'E801 7000	8
JPEG 码命令寄存器	JCCMD	R/W	H'E801 7001	8
JPEG 码量化表号寄存器	JCQTN	R/W	H'E801 7003	8
JPEG 码霍夫曼表号寄存器	JCHTN	R/W	H'E801 7004	8
JPEG 码 DRI 高位寄存器	JCDRIU	R/W	H'E801 7005	8
JPEG 码 DRI 低位寄存器	JCDRID	R/W	H'E801 7006	8
JPEG 码垂直方向尺寸高位寄存器	JCVSZU	R/W	H'E801 7007	8
JPEG 码垂直方向尺寸低位寄存器	JCVSZD	R/W	H'E801 7008	8
JPEG 码水平方向尺寸高位寄存器	JCHSZU	R/W	H'E801 7009	8
JPEG 码水平方向尺寸低位寄存器	JCHSZD	R/W	H'E801 700A	8
JPEG 码数据计数高位寄存器	JCDTCU	R	H'E801 700B	8
JPEG 码数据计数中位寄存器	JCDTCM	R	H'E801 700C	8
JPEG 码数据计数低位寄存器	JCDTCD	R	H'E801 700D	8
JPEG 中断允许寄存器 0	JINTE0	R/W	H'E801 700E	8
JPEG 中断状态寄存器 0	JINTS0	R/W	H'E801 700F	8
JPEG 码解码错误寄存器	JCDERR	R/W	H'E801 7010	8
JPEG 码重新启动寄存器	JCRST	R	H'E801 7011	8
JPEG 接口压缩控制寄存器	JIFECNT	R/W	H'E801 7040	32
JPEG 接口压缩源地址寄存器	JIFESA	R/W	H'E801 7044	32
JPEG 接口压缩行偏移寄存器	JIFESOFST	R/W	H'E801 7048	32
JPEG 接口压缩目标地址寄存器	JIFEDA	R/W	H'E801 704C	32
JPEG 接口压缩源行计数寄存器	JIFESLC	R/W	H'E801 7050	32
JPEG 接口压缩目标寄存器	JIFEDDC	R/W	H'E801 7054	32
JPEG 接口解压控制寄存器	JIFDCNT	R/W	H'E801 7058	32
JPEG 接口解压源地址寄存器	JIFDSA	R/W	H'E801 705C	32
JPEG 接口解压目标偏移寄存器	JIFDDOFST	R/W	H'E801 7060	32
JPEG 接口解压目标地址寄存器	JIFDDA	R/W	H'E801 7064	32
JPEG 接口解压源计数寄存器	JIFSDC	R/W	H'E801 7068	32
JPEG 接口解压目标行计数寄存器	JIFDDL	R/W	H'E801 706C	32
JPEG 接口解压 α 设定寄存器	JIFDADT	R/W	H'E801 7070	32
JPEG 中断允许寄存器 1	JINTE1	R/W	H'E801 708C	32
JPEG 中断状态寄存器 1	JINTS1	R/W	H'E801 7090	32
JPEG 码量化表 0 寄存器	JCQTBL0	R/W	H'E801 7100 ~ H'E801 713F	8
JPEG 码量化表 1 寄存器	JCQTBL1	R/W	H'E801 7140 ~ H'E801 717F	8
JPEG 码量化表 2 寄存器	JCQTBL2	R/W	H'E801 7180 ~ H'E801 71BF	8

寄存器名称	略称	R/W	地址	存取长度
JPEG 码量化表 3 寄存器	JCQTBL3	R/W	H'E801 71C0 ~ H'E801 71FF	8
JPEG 码霍夫曼表 DC0 寄存器	JCHTBD0	W	H'E801 7200 ~ H'E801 721B	8
JPEG 码霍夫曼表 AC0 寄存器	JCHTBA0	W	H'E801 7220 ~ H'E801 72D1	8
JPEG 码霍夫曼表 DC1 寄存器	JCHTBD1	W	H'E801 7300 ~ H'E801 731B	8
JPEG 码霍夫曼表 AC1 寄存器	JCHTBA1	W	H'E801 7320 ~ H'E801 73D1	8

【注】 有关 JPEG 码量化表和 JPEG 码霍夫曼表的设定, 请参照“41.3.1 (4) 表的设定”。

41.2.1 JPEG 码模式寄存器 (JCMOD)

JCMOD 是在运行前设定各种模式的寄存器。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	DSP	REDU[2:0]		
初始值:	0	0	0	0	0	0	0	0
R/W(压缩):	R	R	R	R	R/W	R/W	R/W	R/W
R/W(解压):	R	R	R	R	R/W	R	R	R

位	位名	初始值	R/W		说明
			压缩	解压	
7 ~ 4	—	全 0	R		保留位 读写值都为“0”。
3	DSP	0	R/W		压缩 / 解压设定 0: 压缩处理 1: 解压处理 【注】 如果要更改压缩处理和解压处理, 就必须通过低功耗模式的软件复位控制寄存器 2 (SWRSTCR2) 的 JCUSRST 位, 使此模块进入复位状态。
2 ~ 0	REDU[2:0]	000	R/W	R	像素格式设定 [压缩时] 001: YCbCr422 上述以外: 禁止设定 [解压时] 001: YCbCr422 010: YCbCr420 上述以外: 因为不能正确地进行处理, 所以必须作为错误进行处理。

41.2.2 JPEG 码命令寄存器 (JCCMD)

JCCMD 是设定各种命令的寄存器。在设定命令后，不需要通过写“0”进行清除。
禁止同时设定各种命令。

位:	7	6	5	4	3	2	1	0
	BRST	—	—	—	—	JEND	JRST	JSRT
初始值:	0	0	0	0	0	0	0	0
R/W(压缩):	R*/W	R	R	R	R	R*/W	无效	R*/W
R/W(解压):	R*/W	R	R	R	R	R*/W	R*/W	R*/W

【注】* 读取值为不定值。

位	位名	初始值	R/W		说明
			压缩	解压	
7	BRST	0	R*/W		总线复位 通过将此位置“1”，对内部电路进行复位。 不能在运行过程中（从设定 JPEG 核心处理开始命令到写完最后的输出符号/图像数据为止）将此位置“1”。 有关总线复位处理，请参照“41.5 总线复位处理”。
6~3	—	全0	R		保留位 读写值都为“0”。
2	JEND	0	R*/W		中断请求解除命令 此位只对 JINTS0 的 INS6 位、INS5 位和 INS3 位的中断源有效。 要解除中断请求时，必须将此位置“1”。
1	JRST	0	无效	R*/W	JPEG 核心处理停止解除命令 在解除由图像尺寸和像素格式的读请求（JINTE0 的 INT3 位的设定）产生的处理停止状态时，必须将此位置“1”。
0	JSRT	0	R*/W		JPEG 核心处理开始命令 在开始处理时，必须将此位置“1”。不能在运行过程中重新写“1”。

【注】* 读取值为不定值。

41.2.3 JPEG 码量化表号寄存器 (JCQTN)

JCQTN 是在压缩处理前设定量化表号的寄存器。

- 当量化表 No.0 JCQTBL0 用于第一颜色成分时，给 QT1 设定 “B'00”。
- 当量化表 No.1 JCQTBL1 用于第一颜色成分时，给 QT1 设定 “B'01”。
- 当量化表 No.2 JCQTBL2 用于第一颜色成分时，给 QT1 设定 “B'10”。
- 当量化表 No.3 JCQTBL3 用于第一颜色成分时，给 QT1 设定 “B'11”。

位:	7	6	5	4	3	2	1	0
	—	—	QT3[1:0]	QT2[1:0]	QT1[1:0]			
初始值:	0	0	0	0	0	0	0	0
R/W(压缩):	R	R	R/W	R/W	R/W	R/W	R/W	R/W
R/W(解压):	R	R	R	R	R	R	R	R

位	位名	初始值	R/W		说明
			压缩	解压	
7、6	—	全 0	R		保留位 读写值都为 “0”。
5、4	QT3[1:0]	00	R/W	R	第三颜色成分的量化表号
3、2	QT2[1:0]	00	R/W	R	第二颜色成分的量化表号
1、0	QT1[1:0]	00	R/W	R	第一颜色成分的量化表号

41.2.4 JPEG 码霍夫曼表号寄存器 (JCHTN)

JCHTN 是在压缩处理前设定霍夫曼表号 (AC/DC) 的寄存器。

- 当 DC/AC 霍夫曼表 No.0 (JCHTBD0 和 JCHTBA0) 用于第一颜色成分时，给 HTA1 位设定 “B'0” 并且给 HTD1 位设定 “B'0”。
- 当 DC/AC 霍夫曼表 No.1 (JCHTBD1、JCHTBA1) 用于第一颜色成分时，给 HTA1 位设定 “B'1” 并且给 HTD1 位设定 “B'1”。

位:	7	6	5	4	3	2	1	0
	—	—	HTA3	HTD3	HTA2	HTD2	HTA1	HTD1
初始值:	0	0	0	0	0	0	0	0
R/W(压缩):	R	R	R/W	R/W	R/W	R/W	R/W	R/W
R/W(解压):	R	R	R	R	R	R	R	R

位	位名	初始值	R/W		说明
			压缩	解压	
7、6	—	全 0	R		保留位 读写值都为 “0”。
5	HTA3	0	R/W	R	第三颜色成分的霍夫曼表号 (AC)
4	HTD3	0	R/W	R	第三颜色成分的霍夫曼表号 (DC)
3	HTA2	0	R/W	R	第二颜色成分的霍夫曼表号 (AC)
2	HTD2	0	R/W	R	第二颜色成分的霍夫曼表号 (DC)
1	HTA1	0	R/W	R	第一颜色成分的霍夫曼表号 (AC)
0	HTD1	0	R/W	R	第一颜色成分的霍夫曼表号 (DC)

41.2.5 JPEG 码 DRI 高位寄存器 (JCDRIU)

JCDRIU 设定要插入 RST 标记的 MCU 数的高位字节。

位:	7	6	5	4	3	2	1	0
	DRIU[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W(压缩):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W(解压):	无效	无效	无效	无效	无效	无效	无效	无效

位	位名	初始值	R/W		说明
			压缩	解压	
7 ~ 0	DRIU[7:0]	H'00	R/W	无效	要插入 RST 标记的 MCU 数的高位字节 如果将高位和低位都置“H'00”，就不插入 DRI 标记和 RST 标记。

41.2.6 JPEG 码 DRI 低位寄存器 (JCDRID)

JCDRID 设定要插入 RST 标记的 MCU 数的低位字节。

位:	7	6	5	4	3	2	1	0
	DRID[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W(压缩):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W(解压):	无效	无效	无效	无效	无效	无效	无效	无效

位	位名	初始值	R/W		说明
			压缩	解压	
7 ~ 0	DRID[7:0]	H'00	R/W	无效	要插入 RST 标记的 MCU 数的低位字节 如果将高位和低位都置“H'00”，就不插入 DRI 标记和 RST 标记。

41.2.7 JPEG 码垂直方向尺寸高位寄存器 (JCVSZU)

JCVSZU 是设定垂直方向图像尺寸值的高位字节的寄存器。

位:	7	6	5	4	3	2	1	0
	VSZU[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W(压缩):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W(解压):	R	R	R	R	R	R	R	R

位	位名	初始值	R/W		说明
			压缩	解压	
7 ~ 0	VSZU[7:0]	H'00	R/W	R	垂直方向图像尺寸值的高位字节 在解压时，这些位被设定为从 JPEG 符号数据下载的值。

41.2.8 JPEG 码垂直方向尺寸低位寄存器 (JCVSZD)

JCVSZD 是设定垂直方向图像尺寸值的低位字节的寄存器。

位:	7	6	5	4	3	2	1	0
	VSZD[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W(压缩):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W(解压):	R	R	R	R	R	R	R	R

位	位名	初始值	R/W		说明
			压缩	解压	
7 ~ 0	VSZD[7:0]	H'00	R/W	R	垂直方向图像尺寸值的低位字节 在解压时, 这些位被设定为从 JPEG 符号数据下载的值。

41.2.9 JPEG 码水平方向尺寸高位寄存器 (JCHSZU)

JCHSZU 是设定水平方向图像尺寸值的高位字节的寄存器。

位:	7	6	5	4	3	2	1	0
	HSZU[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W(压缩):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W(解压):	R	R	R	R	R	R	R	R

位	位名	初始值	R/W		说明
			压缩	解压	
7 ~ 0	HSZU[7:0]	H'00	R/W	R	水平方向图像尺寸值的高位字节 在解压时, 这些位被设定为从 JPEG 符号数据下载的值。

41.2.10 JPEG 码水平方向尺寸低位寄存器 (JCHSZD)

JCHSZD 是设定水平方向图像尺寸值的低位字节的寄存器。

位:	7	6	5	4	3	2	1	0
	HSZD[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W(压缩):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W(解压):	R	R	R	R	R	R	R	R

位	位名	初始值	R/W		说明
			压缩	解压	
7 ~ 0	HSZD[7:0]	H'00	R/W	R	水平方向图像尺寸值的低位字节 在解压时, 这些位被设定为从 JPEG 符号数据下载的值。

41.2.11 JPEG 码数据计数高位寄存器 (JCDCU)

JCDCU 是设定压缩数据量计数值的高位字节的寄存器。在开始处理时，对此寄存器的值进行复位。

位:	7	6	5	4	3	2	1	0
	DCU[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W(压缩):	R	R	R	R	R	R	R	R
R/W(解压):	无效	无效	无效	无效	无效	无效	无效	无效

位	位名	初始值	R/W		说明
			压缩	解压	
7 ~ 0	DCU[7:0]	H'00	R	无效	压缩数据量计数值的高位字节

41.2.12 JPEG 码数据计数中位寄存器 (JCDCM)

JCDCM 是设定压缩数据量计数值的中位字节的寄存器。在开始处理时，对此寄存器的值进行复位。

位:	7	6	5	4	3	2	1	0
	DCM[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W(压缩):	R	R	R	R	R	R	R	R
R/W(解压):	无效	无效	无效	无效	无效	无效	无效	无效

位	位名	初始值	R/W		说明
			压缩	解压	
7 ~ 0	DCM[7:0]	H'00	R	无效	压缩数据量计数值的中位字节

41.2.13 JPEG 码数据计数低位寄存器 (JCDCD)

JCDCD 是设定压缩数据量计数值的低位字节的寄存器。在开始处理时，对此寄存器的值进行复位。

位:	7	6	5	4	3	2	1	0
	DCD[7:0]							
初始值:	0	0	0	0	0	0	0	0
R/W(压缩):	R	R	R	R	R	R	R	R
R/W(解压):	无效	无效	无效	无效	无效	无效	无效	无效

位	位名	初始值	R/W		说明
			压缩	解压	
7 ~ 0	DCD[7:0]	H'00	R	无效	压缩数据量计数值的低位字节

41.2.14 JPEG 中断允许寄存器 0 (JINTE0)

JINTE0 是设定中断允许的寄存器。

如果将 INT7 ~ INT5 位置“B'1”，就在发生压缩数据错误时 JINTS0 的 INS5 位 (错误状态位) 变为“B'1”。JCDERR 的 ERR 位表示详细的错误代码。

位:	7	6	5	4	3	2	1	0
	INT7	INT6	INT5	—	INT3	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W(压缩):	无效	无效	无效	R	无效	R	R	R
R/W(解压):	R/W	R/W	R/W	R	R/W	R	R	R

位	位名	初始值	R/W		说明
			压缩	解压	
7	INT7	0	无效	R/W	在解压时, 设定在霍夫曼符号化段内的重新启动间隔期间的数据量发生异常时是否产生中断。 如果不设定, 就不返回错误码。
6	INT6	0	无效	R/W	在解压时, 设定在霍夫曼符号化段内的总数据量发生异常时是否产生中断。 如果不设定, 就不返回错误码。
5	INT5	0	无效	R/W	在解压时, 设定在霍夫曼符号化段内的最后 MCU 数据量发生异常时是否产生中断。 如果不设定, 就不返回错误码。
4	—	0	R		保留位 读写值都为“0”。
3	INT3	0	无效	R/W	设定在能读压缩数据的解析结果、图像尺寸和减少的设定值时是否产生中断。
2~0	—	0	R		保留位 读写值都为“0”。

41.2.15 JPEG 中断状态寄存器 0 (JINTS0)

JINTS0 是判断中断发生源的寄存器。

对于此寄存器的中断源，必须在清除状态的同时，通过 JCCMD 解除中断。

位:	7	6	5	4	3	2	1	0
	—	INS6	INS5	—	INS3	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W(压缩):	R	R/W*	无效	R	无效	R	R	R
R/W(解压):	R	R/W*	R/W*	R	R/W*	R	R	R

【注】* 必须通过写“0”进行清除。
不能写“1”。

位	位名	初始值	R/W		说明
			压缩	解压	
7	—	0	R		保留位 读写值都为“0”。
6	INS6	0	R/W*		如果正常结束，就将此位置“1”。
5	INS5	0	无效	R/W*	如果发生压缩数据错误，就将此位置“1”。
4	—	0	R		保留位 读写值都为“0”。
3	INS3	0	无效	R/W*	在能读图像尺寸和像素格式时，此位被置“1”。在发生中断时，此模块为处理停止状态。JCRST 表示处理停止状态。通过设定处理停止解除命令 (JCCMD 的 JRST 位)，重新开始处理。
2~0	—	全 0	R		保留位 读写值都为“0”。

【注】* 必须通过写“0”进行清除，不能写“1”。

41.2.16 JPEG 码解码错误寄存器 (JCDERR)

在解压时，JCDERR 通过代码判断压缩数据解析中发生错误时的错误种类。

在开始处理时，对此寄存器的值进行复位。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	ERR[3:0]			
初始值:	0	0	0	0	1	0	1	0
R/W(压缩):	R	R	R	R	无效	无效	无效	无效
R/W(解压):	R	R	R	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W		说明
			压缩	解压	
7~4	—	全 0	R		保留位 读取值总是“0”。
3~0	ERR[3:0]	1010	无效	R/W	错误码 (参照表 41.3 和表 41.4)

41.2.17 JPEG 码重新启动寄存器 (JCRST)

JCRST 表示由图像尺寸和像素格式的读请求 (JINTE0 的 INT3 位的设定) 产生的处理停止状态。
通过设定处理停止解除命令 JCCMD 的 JRST 位, 重新开始处理。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	RST
初始值:	0	0	0	0	0	0	0	0
R/W(压缩):	R	R	R	R	R	R	R	无效
R/W(解压):	R	R	R	R	R	R	R	R

位	位名	初始值	R/W		说明
			压缩	解压	
7 ~ 1	—	全 0	R		保留位 读取值总是“0”。
0	RST	0	无效	R	运行状态 0: 下述以外的状态 1: JINTE0 的中断源产生的挂起状态

41.2.18 JPEG 接口压缩控制寄存器 (JIFECNT)

JIFECNT 是控制压缩处理的寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(解压):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	JOUT RINI	JOUT RCMD	JOUT C	—	JOUTSWAP[2:0]			—	DIN RINI	DIN RCMD	DIN LC	—	DINSWAP[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W
R/W(解压):	R	无效	无效	无效	R	无效	无效	无效	R	无效	无效	无效	R	无效	无效	无效

位	位名	初始值	R/W		说明
			压缩	解压	
31 ~ 15	—	全 0	R		保留位 读写值都为“0”。
14	JOUTRINI	0	R/W	无效	重新开始输出符号数据时的地址初始化设定 此位只在输出符号数据停止计数 ON 时有效。 必须在给数据重新开始命令写“1”前进行设定。 0: 在重新开始输出符号数据时, 不对传送地址进行初始化。 1: 在重新开始输出符号数据时, 对传送地址进行初始化。
13	JOUTRCMD	0	R/W	无效	输出符号数据重新开始命令 此位只在输出符号数据停止计数 ON 时有效。 通过将此位置“1”, 重新开始写输出符号数据。 读取值总是“0”。
12	JOUTC	0	R/W	无效	输出符号数据的停止计数模式设定 0: 输出符号数据停止计数 OFF 1: 输出符号数据停止计数 ON

位	位名	初始值	R/W		说明
			压缩	解压	
11	—	0	R		保留位 读写值都为“0”。
10 ~ 8	JOUTSWAP [2:0]	000	R/W	无效	字节 / 字 / 长字的交换设定 压缩时的输出符号数据为交换对象。 000: (1) (2) (3) (4) (5) (6) (7) (8) 001: (2) (1) (4) (3) (6) (5) (8) (7) [字节交换] 010: (3) (4) (1) (2) (7) (8) (5) (6) [字交换] 011: (4) (3) (2) (1) (8) (7) (6) (5) [字 - 字节交换] 100: (5) (6) (7) (8) (1) (2) (3) (4) [长字交换] 101: (6) (5) (8) (7) (2) (1) (4) (3) [长字 - 字节交换] 110: (7) (8) (5) (6) (3) (4) (1) (2) [长字 - 字交换] 111: (8) (7) (6) (5) (4) (3) (2) (1) [长字 - 字 - 字节交换]
7	—	0	R		保留位 读写值都为“0”。
6	DINRINI	0	R/W	无效	重新开始输入图像数据行时的地址初始化设定 此位只在输入图像数据行停止计数 ON 时有效。 必须在给数据行重新开始命令写“1”前进行设定。 0: 在重新开始输入图像数据行时, 不对传送地址进行初始化。 1: 在重新开始输入图像数据行时, 对传送地址进行初始化。
5	DINRCMD	0	R/W	无效	输入图像数据行重新开始命令 此位只在输入图像数据行停止计数 ON 时有效。 通过将此位置“1”, 重新开始读输入图像数据。 读取值总是“0”。
4	DINLC	0	R/W	无效	输入图像数据行的停止计数模式设定 0: 输入图像数据行停止计数 OFF 1: 输入图像数据行停止计数 ON
3	—	0	R		保留位 读写值都为“0”。
2 ~ 0	DINSWAP [2:0]	000	R/W	无效	字节 / 字 / 长字的交换设定 压缩时的输入图像数据为交换对象。 000: (1) (2) (3) (4) (5) (6) (7) (8) 001: (2) (1) (4) (3) (6) (5) (8) (7) [字节交换] 010: (3) (4) (1) (2) (7) (8) (5) (6) [字交换] 011: (4) (3) (2) (1) (8) (7) (6) (5) [字 - 字节交换] 100: (5) (6) (7) (8) (1) (2) (3) (4) [长字交换] 101: (6) (5) (8) (7) (2) (1) (4) (3) [长字 - 字节交换] 110: (7) (8) (5) (6) (3) (4) (1) (2) [长字 - 字交换] 111: (8) (7) (6) (5) (4) (3) (2) (1) [长字 - 字 - 字节交换]

41.2.19 JPEG 接口压缩源地址寄存器 (JIFESA)

JIFESA 设定输入图像数据源地址的寄存器。必须以 8 字节为单位设定此寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ESA[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W(解压):	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ESA[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R
R/W(解压):	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效

位	位名	初始值	R/W		说明
			压缩	解压	
31 ~ 3	ESA[31:3]	H'0000	R/W	无效	输入图像数据的源地址设定 (以 8 字节为单位) 必须将低 3 位置 "0"。
2 ~ 0	ESA[2:0]	0000	R		

41.2.20 JPEG 接口压缩行偏移寄存器 (JIFESOFST)

JIFESOFST 是设定输入图像数据行偏移的寄存器 (参照 "41.3.4 图像数据的保存")。
必须以 8 字节为单位设定此寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(解压):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	ESMW[14:0]														
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R
R/W(解压):	R	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效

位	位名	初始值	R/W		说明
			压缩	解压	
31 ~ 15	—	全 0	R		保留位 读写值都为 "0"。
14 ~ 3	ESMW[14:3]	H'0000	R/W	无效	输入图像数据的行偏移设定 (以 8 字节为单位) 必须将低 3 位置 "0"。
2 ~ 0	ESMW[2:0]		R		

41.2.21 JPEG 接口压缩目标地址寄存器 (JIFEDA)

JIFEDA 是设定输出符号数据目标地址的寄存器。

必须以 8 字节为单位设定此寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EDA[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W(解压):	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EDA[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R
R/W(解压):	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效

位	位名	初始值	R/W		说明
			压缩	解压	
31 ~ 3	EDA[31:3]	H'0000	R/W	无效	输出符号数据的目标地址设定 (以 8 字节为单位) 必须将低 3 位置 “0”。
2 ~ 0	EDA[2:0]	0000	R		

41.2.22 JPEG 接口压缩源行计数寄存器 (JIFESLC)

JIFESLC 设定输入图像数据行停止计数 ON (JIFECNT 的 DINLC 位为 “1”) 时的输入图像数据行数。必须以 8 行为单位设定此寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(压缩):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(解压):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LINES[15:0]															
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(压缩):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R
R/W(解压):	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效

位	位名	初始值	R/W		说明
			压缩	解压	
31 ~ 16	—	H'FFF8	R		保留位 读取值为不定值, 写入值总是 “0”。
15 ~ 3	LINES[15:3]	H'FFF8	R/W	无效	必须设定输入图像数据的读行数 (以 8 行为单位)。 必须将低 3 位置 “0”。
2 ~ 0	LINES[2:0]		R		

41.2.23 JPEG 接口压缩目标计数寄存器 (JIFEDDC)

JIFEDDC 设定输出符号数据停止计数 ON (JIFECNT 的 JOUTC 位为“1”)时的输出符号数据量。必须以 8 字节为单位设定此寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(压缩):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(解压):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	JDATAS[15:0]															
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(压缩):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R
R/W(解压):	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效

位	位名	初始值	R/W		说明
			压缩	解压	
31 ~ 16	—	H'FFF8	R		保留位 读取值为不定值，写入值总是“0”。
15 ~ 3 2 ~ 0	JDATAS[15:3] JDATAS[2:0]	H'FFF8	R/W R	无效	必须设定输出符号数据的写数据量（以 8 字节为单位）。 必须将低 3 位置“0”。

41.2.24 JPEG 接口解压控制寄存器 (JIFDCNT)

JIFDCNT 是控制解压处理的寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	VINTER[1:0]	HINTER[1:0]	OPF[1:0]	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	R	R	无效	无效	无效	无效	无效	无效	R	R	R	R	R	R	R	R
R/W(解压):	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	JINRINI	JINRCMD	JINC	—	JINSWAP[2:0]	—	DOUTRINI	DOUTRCMD	DOUTLC	—	DOUTSWAP[2:0]	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	R	无效	无效	无效	R	无效	无效	无效	R	无效	无效	无效	R	无效	无效	无效
R/W(解压):	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W		说明
			压缩	解压	
31、30	—	全 0	R		保留位 读写值都为“0”。
29、28	VINTER [1:0]	00	无效	R/W	垂直方向减少设定 减少输出图像数据的垂直方向。 00: 未减少 01: 减少到 1/2 10: 减少到 1/4 11: 减少到 1/8
27、26	HINTER [1:0]	00	无效	R/W	水平方向的减少设定 减少输出图像数据的水平方向。 00: 未减少 01: 减少到 1/2 10: 减少到 1/4 11: 减少到 1/8
25、24	OPF[1:0]	00	无效	R/W	指定输出图像数据的像素格式。 00: YCbCr422 01: ARGB8888 10: RGB565 11: 禁止设定
23 ~ 15	—	全 0	R		保留位 读写值都为“0”。
14	JINRINI	0	无效	R/W	重新开始输入符号数据时的地址初始化设定 此位只在输入符号数据停止计数 ON 时有效。 必须在给数据重新开始命令写“1”前进行设定。 0: 在重新开始输入符号数据时, 不对传送地址进行初始化。 1: 在重新开始输入符号数据时, 对传送地址进行初始化。
13	JINRCMD	0	无效	R/W	输入符号数据重新开始命令 此位只在输入符号数据停止计数 ON 时有效。 通过给此位置“1”, 重新开始读输入符号数据。 读取值总是“0”。
12	JINC	0	无效	R/W	输入符号数据的停止计数设定 0: 输入符号数据停止计数 OFF 1: 输入符号数据停止计数 ON

位	位名	初始值	R/W		说明
			压缩	解压	
11	—	0	R		保留位 读写值都为“0”。
10 ~ 8	JINSWAP [2:0]	000	无效	R/W	字节 / 字 / 长字的交换设定 解压时的输入符号数据为交换对象。 000: (1) (2) (3) (4) (5) (6) (7) (8) 001: (2) (1) (4) (3) (6) (5) (8) (7) [字节交换] 010: (3) (4) (1) (2) (7) (8) (5) (6) [字交换] 011: (4) (3) (2) (1) (8) (7) (6) (5) [字 - 字节交换] 100: (5) (6) (7) (8) (1) (2) (3) (4) [长字交换] 101: (6) (5) (8) (7) (2) (1) (4) (3) [长字 - 字节交换] 110: (7) (8) (5) (6) (3) (4) (1) (2) [长字 - 字交换] 111: (8) (7) (6) (5) (4) (3) (2) (1) [长字 - 字 - 字节交换]
7	—	0	R		保留位 读写值都为“0”。
6	DOUINTRINI	0	无效	R/W	重新开始输出图像数据行时的地址初始化设定 此位只在输出图像数据行停止计数 ON 时有效。 必须在给数据行重新开始命令写“1”前进行设定。 0: 在重新开始输出图像数据行时, 不对传送地址进行初始化。 1: 在重新开始输出图像数据行时, 对传送地址进行初始化。
5	DOUTRCMD	0	无效	R/W	输出图像数据行重新开始命令 此位只在输出图像数据行停止计数 ON 时有效。 通过给此位置“1”, 重新开始写图像数据。 读取值总是“0”。
4	DOUOLC	0	无效	R/W	输出图像数据行的停止计数设定 0: 输出图像数据行停止计数 OFF 1: 输出图像数据行停止计数 ON
3	—	0	R		保留位 读写值都为“0”。
2 ~ 0	DOUOSWAP [2:0]	000	无效	R/W	字节 / 字 / 长字的交换设定 解压时的输出图像数据为交换对象。 000: (1) (2) (3) (4) (5) (6) (7) (8) 001: (2) (1) (4) (3) (6) (5) (8) (7) [字节交换] 010: (3) (4) (1) (2) (7) (8) (5) (6) [字交换] 011: (4) (3) (2) (1) (8) (7) (6) (5) [字 - 字节交换] 100: (5) (6) (7) (8) (1) (2) (3) (4) [长字交换] 101: (6) (5) (8) (7) (2) (1) (4) (3) [长字 - 字节交换] 110: (7) (8) (5) (6) (3) (4) (1) (2) [长字 - 字交换] 111: (8) (7) (6) (5) (4) (3) (2) (1) [长字 - 字 - 字节交换]

41.2.25 JPEG 接口解压源地址寄存器 (JIFDSA)

JIFDSA 设定输入符号数据的源地址。必须以 8 字节为单位设定此寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSA[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效
R/W(解压):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSA[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效
R/W(解压):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

位	位名	初始值	R/W		说明
			压缩	解压	
31 ~ 3	DSA[31:3]	H'0000	无效	R/W	输入符号数据的源地址设定 (以 8 字节为单位)
2 ~ 0	DSA[2:0]	0000		R	必须将低 3 位置 "0"。

41.2.26 JPEG 接口解压行偏移寄存器 (JIFDDOFST)

JIFDDOFST 是设定输出图像数据行偏移的寄存器 (参照 "41.3.4 图像数据的保存")。

必须以 8 字节为单位设定此寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(解压):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	DDMW[14:0]														
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	R	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效
R/W(解压):	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

位	位名	初始值	R/W		说明
			压缩	解压	
31 ~ 15	—	全 0		R	保留位 读写值都为 "0"。
14 ~ 3	DDMW[14:3]	H'0000	无效	R/W	输出图像数据的行偏移设定 (以 8 字节为单位)
2 ~ 0	DDMW[2:0]			R	必须将低 3 位置 "0"。

41.2.27 JPEG 接口解压目标地址寄存器 (JIFDDA)

JIFDDA 是设定输出图像数据目标地址的寄存器。必须以 8 字节为单位设定此寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DDA[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效
R/W(解压):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DDA[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效
R/W(解压):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

位	位名	初始值	R/W		说明
			压缩	解压	
31 ~ 3	DDA[31:3]	H'0000	无效	R/W	输出图像数据的目标地址设定 (以 8 字节为单位) 必须将低 3 位置 "0"。
2 ~ 0	DDA[2:0]	0000		R	

41.2.28 JPEG 接口解压源数据计数寄存器 (JIFDSDC)

JIFDSDC 设定输入符号数据停止计数 ON (JIFDCNT 的 JINC 位为 "1") 时的输入符号数据量。必须以 8 字节为单位设定此寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(压缩):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(解压):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	JDATAS[15:0]															
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(压缩):	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效
R/W(解压):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

位	位名	初始值	R/W		说明
			压缩	解压	
31 ~ 16	—	H'FFF8	R		保留位 读取值为不定值, 写入值总是 "0"。
15 ~ 3	JDATAS[15:3]	H'FFF8	无效	R/W	必须设定输入符号数据的读数据量 (以 8 字节为单位)。 必须将低 3 位置 "0"。
2 ~ 0	JDATAS[2:0]			R	

41.2.29 JPEG 接口解压目标行计数寄存器 (JIFDDL)C)

JIFDDL 设定输出图像数据行停止计数 ON (JIFDCNT 的 DOUTLC 位为“1”)时的输出图像数据行数。必须以 8 行为单位设定此寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(压缩):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(解压):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LINES[15:0]															
初始值:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(压缩):	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效	无效
R/W(解压):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

位	位名	初始值	R/W		说明
			压缩	解压	
31 ~ 16	—	H'FFF8	R		保留位 读取值为不定值，写入值总是“0”。
15 ~ 3 2 ~ 0	LINES[15:3] LINES[2:0]	H'FFF8	无效	R/W R	必须设定输出图像数据的写行数 (8 行单位)。 必须将低 3 位置“0”。

41.2.30 JPEG 接口解压 α 设定寄存器 (JIFDADT)

JIFDADT 设定以 ARGB8888 格式输出时的 α 值。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(解压):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ALPHA[7:0]							
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	R	R	R	R	R	R	R	R	无效	无效	无效	无效	无效	无效	无效	无效
R/W(解压):	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W		说明
			压缩	解压	
31 ~ 8	—	全 0	R		保留位 读写值都为“0”。
7 ~ 0	ALPHA[7:0]	H'00	无效	R/W	设定以 ARGB8888 格式输出时的 α 值。

41.2.31 JPEG 中断允许寄存器 1 (JINTE1)

JINTE1 是设定中断允许的寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(解压):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CBTEN	DINLEN	JOUTEN	—	DBTEN	JINEN	DOUTLEN
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	无效	无效	无效
R/W(解压):	R	R	R	R	R	R	R	R	R	无效	无效	无效	R	R/W	R/W	R/W

位	位名	初始值	R/W		说明
			压缩	解压	
31 ~ 7	—	全 0	R		保留位 读写值都为“0”。
6	CBTEN	0	R/W	无效	允许或者禁止在 JINTS1 的 CBTF 位被置“1”时产生数据传送处理中断请求 (JDTI)。 0: 禁止中断请求 1: 允许中断请求
5	DINLEN	0	R/W	无效	允许或者禁止在 JINTS1 的 DINLF 位被置“1”时产生数据传送处理中断请求 (JDTI)。 0: 禁止中断请求 1: 允许中断请求
4	JOUTEN	0	R/W	无效	允许或者禁止在 JINTS1 的 JOUTF 位被置“1”时产生数据传送处理中断请求 (JDTI)。 0: 禁止中断请求 1: 允许中断请求
3	—	0	R		保留位 读写值都为“0”。
2	DBTEN	0	无效	R/W	允许或者禁止在 JINTS1 的 DBTF 位被置“1”时产生数据传送处理中断请求 (JDTI)。 0: 禁止中断请求 1: 允许中断请求
1	JINEN	0	无效	R/W	允许或者禁止在 JINTS1 的 JINF 位被置“1”时产生数据传送处理中断请求 (JDTI)。 0: 禁止中断请求 1: 允许中断请求
0	DOUTLEN	0	无效	R/W	允许或者禁止在 JINTS1 的 DOUTLF 位被置“1”时产生数据传送处理中断请求 (JDTI)。 0: 禁止中断请求 1: 允许中断请求

41.2.32 JPEG 中断状态寄存器 1 (JINTS1)

JINTS1 是判断中断发生源的寄存器。

必须通过写“0”来清除此寄存器的中断源。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(解压):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CBTF	DINLF	JOUTF	—	DBTF	JINF	DOU TLF
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(压缩):	R	R	R	R	R	R	R	R	R	R/W*	R/W*	R/W*	R	无效	无效	无效
R/W(解压):	R	R	R	R	R	R	R	R	R	无效	无效	无效	R	R/W*	R/W*	R/W*

【注】* 只能通过给读取值为“1”的位写“0”进行清除。
必须给读取值为“0”的位写“1”。

位	位名	初始值	R/W		说明
			压缩	解压	
31 ~ 7	—	全 0	R		保留位 读写值都为“0”。
6	CBTF	0	R/W*	无效	如果在压缩时写最后的输出符号数据, 就将此位置“1”。
5	DINLF	0	R/W*	无效	如果在压缩时读 JIFESLC 所示行数的输入图像数据, 就将此位置“1”。 此位只在 JIFECNT 的 DINLC 位为“1”时有效。
4	JOUTF	0	R/W*	无效	如果在压缩时写 JIFEDDC 所示数据量的输出符号数据, 就将此位置“1”。 此位只在 JIFECNT 的 JOUTC 位为“1”时有效。
3	—	0	R		保留位 读写值都为“0”。
2	DBTF	0	无效	R/W*	如果在解压时写最后的输出图像数据, 就将此位置“1”。
1	JINF	0	无效	R/W*	如果在解压时读 JIFSDC 所示数据量的输入符号数据, 就将此位置“1”。 此位只在 JIFDCNT 的 JINC 位为“1”时有效。
0	DOU TLF	0	无效	R/W*	如果在解压时写 JIFDDLC 所示行数的输出图像数据, 就将此位置“1”。 此位只在 JIFDCNT 的 DOU TLF 位为“1”时有效。

【注】* 只能通过给读取值为“1”的位写“0”进行清除。
必须给读取值为“0”的位写“1”。

41.3 运行说明

41.3.1 压缩

(1) 处理概要

压缩处理的运行流程如下。

1. 启动JPEG核心。

输出标记（能在标记输出结束后输入图像数据）。

SOI~SOS标记的建立需要大约30000个周期。

2. 以MCU为单位将图像数据从外部缓冲器传送到此模块。

在设定输入图像数据行停止计数ON的情况下，每次读完JIFESLC设定行数的图像数据时就停止读操作。通过将JIFECNT的DINRCMD位置“1”，重新开始读操作。

当JIFECNT的DINRINI位为“0”时，从上次传送地址的后续地址重新开始读目标地址；当DINRINI位为“1”时，从JIFESA设定的地址重新开始读目标地址。

在1个画面的图像数据传送结束时也停止读操作。

在设定输入图像数据行停止计数OFF的情况下，一直读到1个画面的图像数据结束为止。

3. 将图像数据输入到JPEG核心部。

在JPEG核心部，以MCU为单位随时进行处理。

4. 将符号数据从此模块传送到外部缓冲器。

在设定输出符号数据停止计数ON的情况下，每次写完JIFEDDC设定数据量的符号数据时就停止写操作。通过将JIFECNT的JOUTRCMD位置“1”，重新开始写操作。

当JIFECNT的JOUTRINI位为“0”时，从上次传送地址的后续地址重新开始写目标地址；当JOUTRINI位为“1”时，从JIFEDA设定的地址重新开始写目标地址。

在1个画面的符号数据传送结束时也停止写操作。

在设定输出符号数据停止计数OFF的情况下，一直写到1个画面的符号数据结束为止。

5. 在1个画面的数据处理结束时结束压缩。

(2) 流程图 (压缩)

(a) 初始设定

在设定 JPEG 核心和输入 / 输出缓冲器以及给外部缓冲器准备图像数据后, 将 JCCMD 的 JSRT 位置 “1” 并且启动此模块。在启动后, 生成并且输出 JPEG 标记头 (SOI ~ SOS), 但是标记头的生成需要大约 30000 个周期。

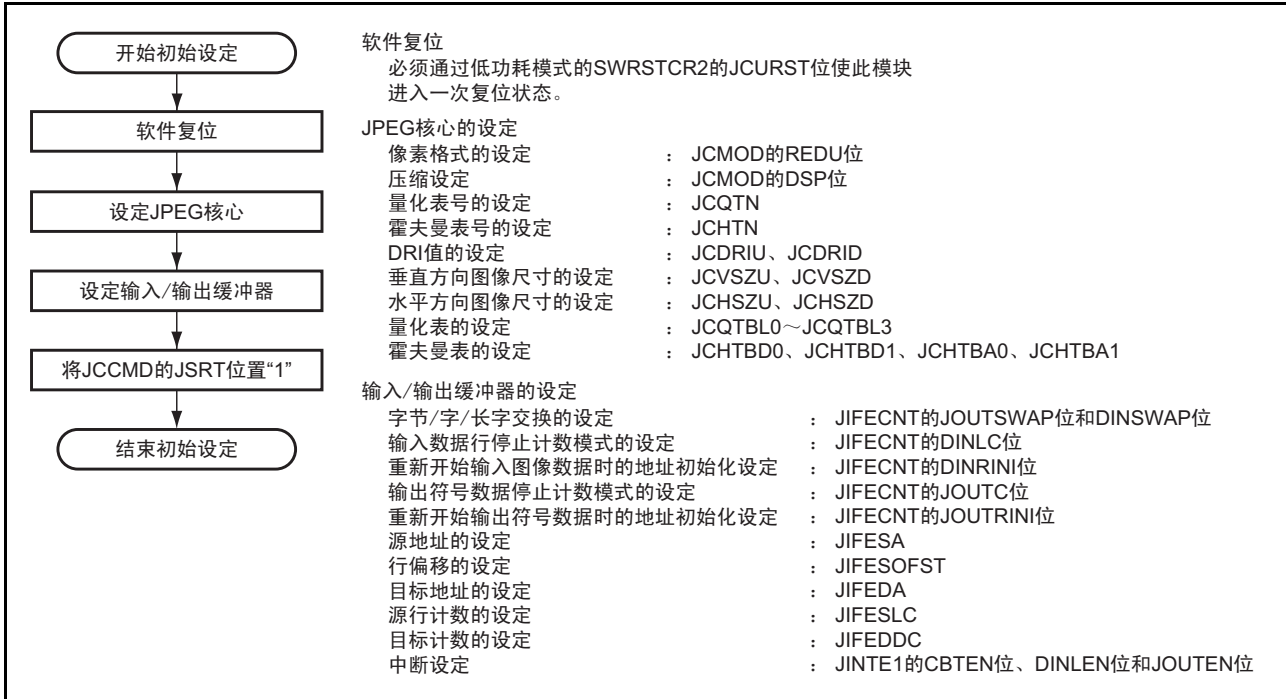


图 41.2 压缩的初始设定流程

(b) 压缩处理

压缩处理时的流程如下所示。

- 如果在 JPEG 核心部压缩处理结束, 就将中断状态 JINTS0 的 INS6 位置 “1”。因为还有未传送的符号数据, 所以此模块继续进行处理。如果最后符号数据的传送结束, 将 JINTS1 的 CBTF 位置 “1”。通过给 INTS6 位写 “0” 来清除中断源。但是, 即使给 INTS6 位写 “0”, 也无法清除被中断源置为有效的中断请求。必须通过设定中断请求解除命令 (将 JCCMD 的 JEND 位置 “1”), 清除中断请求。
- 一旦压缩处理结束并且全部符号数据的传送结束, 就将中断状态 JINTS1 的 CBTF 标志置 “1”。如果 JINTE1 的 CBTEN 位已被置 “1”, 就产生中断。通过给 CBTF 位写 “0” 来清除中断源。
- 在设定输入图像数据行停止计数 ON 的情况下, 一旦读完 JIFESLC 设定行数的图像数据, 就将 JINTS1 的 DINLF 标志置 “1” 并且停止读操作。如果 JINTE1 的 DINLEN 位已被置 “1”, 就发生中断。通过给 DINLEN 位写 “0” 来清除中断源。
通过将 JIFECNT 的 DINRCMD 位置 “1”, 重新开始读操作。
当 JIFECNT 的 DINRINI 位为 “0” 时, 从上次传送地址的后续地址重新开始读目标地址; 当 DINRINI 位为 “1” 时, 从 JIFESA 设定的地址重新开始读目标地址。
- 在设定输出符号数据停止计数 ON 时的情况下, 一旦写完 JIFEDDC 设定数据量的符号数据, 就将 JINT1 的 JOUTF 标志置 “1” 并且停止写操作。如果 JINTE1 的 JOUTEN 位已被置 “1”, 就产生中断。通过给 JOUTF 位写 “0” 来清除中断源。
通过将 JIFECNT 的 JOUTRCMD 位置 “1”, 重新开始写操作。

当 JIFECNT 的 JOUTRINI 位为 “0” 时，从上次传送地址的后续地址重新开始写目标地址；当 JOUTRINI 位为 “1” 时，从 JIFEDA 设定的地址重新开始写目标地址。

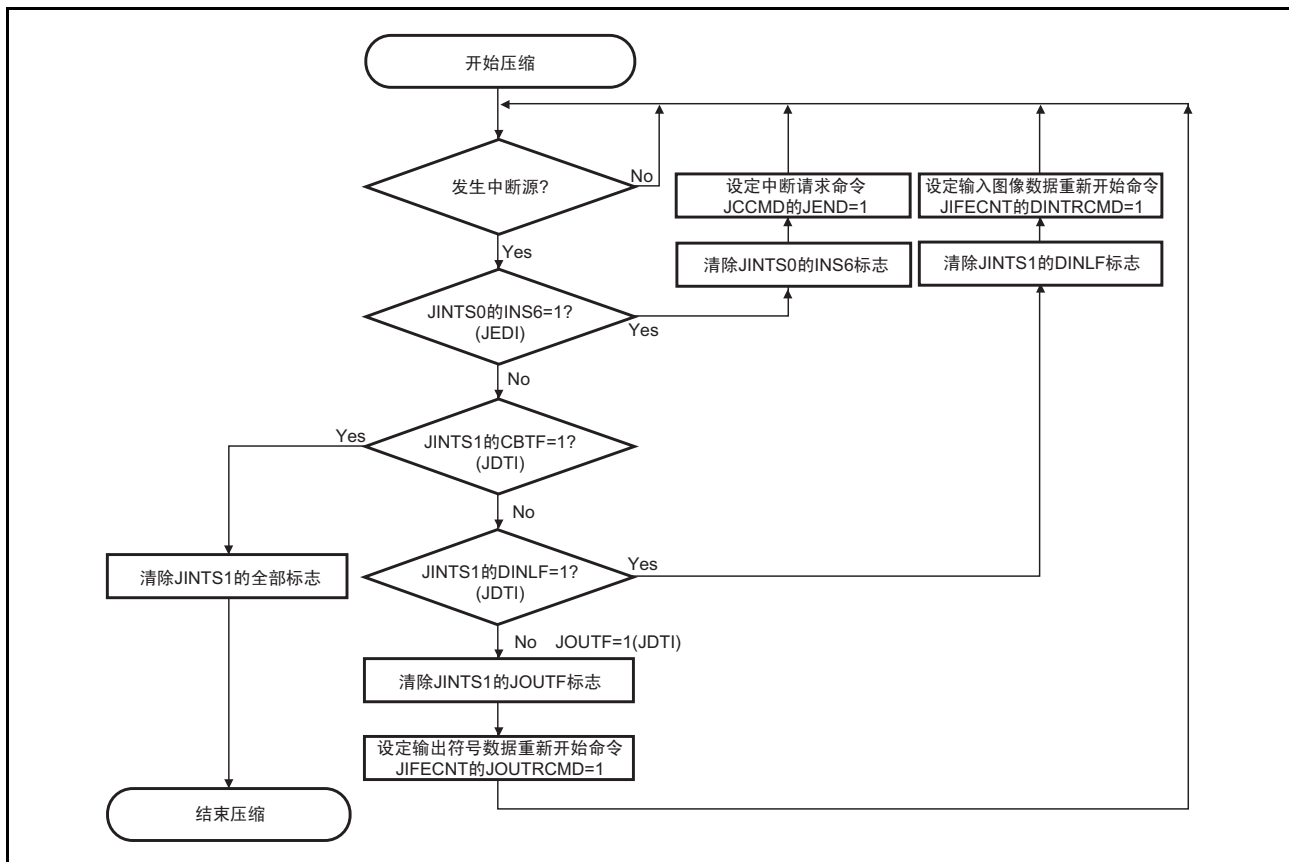


图 41.3 压缩的处理流程

(3) JPEG 符号数据格式

压缩时输出的数据流结构如图 41.4 所示, JCDTCU、JCDTCM 和 JCDTCD 表示 SOI ~ EOI 的符号数据量。如果 JCDRIU 和 JCDRID 都被置“H'0000 0000”, 就不输出以下标记:

- DRI 标记
- RST 标记 (压缩图像数据内)

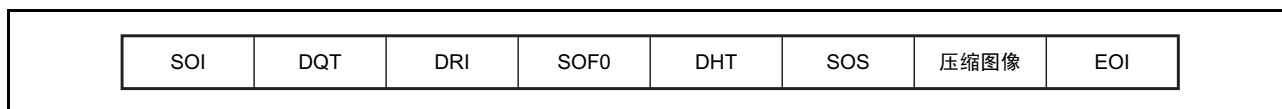


图 41.4 JPEG 符号数据格式

DQT : 不输出未使用的表。

DHT : 输出顺序为 DC0、AC0、DC1、AC1。不输出未使用的表。

SOF0 : 成分标识符为 C1= 第一颜色成分, C2= 第二颜色成分, C3= 第三颜色成分。

SOS : 扫描成分选择为 CS1= 第一颜色成分, CS2= 第二颜色成分, CS3= 第三颜色成分。

各标头容量 (参考)

SOI : 2 字节 (FFD8)

DQT : 在使用 2 张量化表时, 为 134 字节; 在使用 3 张量化表时, 为 199 字节 (增减 1 个为 ±65 字节)

DRI : 6 字节

SOF0 : 19 字节 (4:2:2)

DHT : 420 字节 (使用 2 张表时)

SOS : 14 字节 (4:2:2)

EOI : 2 字节 (FFD9)

(4) 表的设定

(a) 量化表的设定

对应 8×8 块的地址顺序和寄存器地址顺序, 禁止存取此模块正在处理的量化表。

表 41.2 量化表

00	01	02	03	04	05	06	07
08	09	0A	0B	0C	0D	0E	0F
10	11	12	13	14	15	16	17
18	19	1A	1B	1C	1D	1E	1F
20	21	22	23	24	25	26	27
28	29	2A	2B	2C	2D	2E	2F
30	31	32	33	34	35	36	37
38	39	3A	3B	3C	3D	3E	3F

JCQTBL0 (H'E801 7100) =H'00

JCQTBL0 (H'E801 7101) =H'01

JCQTBL0 (H'E801 7102) =H'02

JCQTBL0 (H'E801 7103) =H'03

:

JCQTBL0 (H'E801 713F) =H'3F

(b) 霍夫曼表的设定

JPEG 推荐的 ITU-T T81 附件 K.3.3 中记载的霍夫曼表设定方法

为了在压缩处理时对全部群号都能生成霍夫曼符号，必须按符号数进行以下设定：

- DC 霍夫曼表：各符号长度的符号数为 12 个，发生频率顺序的群号为 12 个。
- AC 霍夫曼表：各符号长度的符号数为 162 个，发生频率顺序的零行程长度/群号为 162 个。

禁止存取此模块正在处理的霍夫曼表，尤其禁止读存取。

- 表 K.3/T81

JCHTBD0 (H'E801 7200) =H'00
 JCHTBD0 (H'E801 7201) =H'01
 JCHTBD0 (H'E801 7202) =H'05
 JCHTBD0 (H'E801 7203) =H'01
 :
 JCHTBD0 (H'E801 721B) =H'0B

- 表 K.4/T81

JCHTBD1 (H'E801 7300) =H'00
 JCHTBD1 (H'E801 7301) =H'03
 JCHTBD1 (H'E801 7302) =H'01
 JCHTBD1 (H'E801 7303) =H'01
 :
 JCHTBD1 (H'E801 731B) =H'0B

- 表 K.5/T81

JCHTBA0 (H'E801 7220) =H'00
 JCHTBA0 (H'E801 7221) =H'02
 JCHTBA0 (H'E801 7222) =H'01
 JCHTBA0 (H'E801 7223) =H'03
 :
 JCHTBA0 (H'E801 72D1) =H'FA

- 表 K.6/T81

JCHTBA1 (H'E801 7320) =H'00
 JCHTBA1 (H'E801 7321) =H'02
 JCHTBA1 (H'E801 7322) =H'01
 JCHTBA1 (H'E801 7323) =H'02
 :
 JCHTBA1 (H'E801 73D1) =H'FA

(5) 输入像素格式

此模块能输入用 YCbCr422 格式制成的图像。

能通过 JIFCNT 的 DINSWAP 位, 更改 YCbCr422 格式的数据配置, 如下所示。

- 当 JOUTSWAP 位为 “000” 时

b63	b56	b55	b48	b47	b40	b39	b32	b31	b24	b23	b16	b15	b8	b7	b0
Y0 8bit	Cb0 8bit	Y1 8bit	Cr0 8bit	Y2 8bit	Cb1 8bit	Y3 8bit	Cr1 8bit								

- 当 JOUTSWAP 位为 “001” 时

b63	b56	b55	b48	b47	b40	b39	b32	b31	b24	b23	b16	b15	b8	b7	b0
Cb0 8bit	Y0 8bit	Cr0 8bit	Y1 8bit	Cb1 8bit	Y2 8bit	Cr1 8bit	Y3 8bit								

- 当 JOUTSWAP 位为 “010” 时

b63	b56	b55	b48	b47	b40	b39	b32	b31	b24	b23	b16	b15	b8	b7	b0
Y1 8bit	Cr0 8bit	Y0 8bit	Cb0 8bit	Y3 8bit	Cr1 8bit	Y2 8bit	Cb1 8bit								

- 当 JOUTSWAP 位为 “100” 时

b63	b56	b55	b48	b47	b40	b39	b32	b31	b24	b23	b16	b15	b8	b7	b0
Y2 8bit	Cb1 8bit	Y3 8bit	Cr1 8bit	Y0 8bit	Cb0 8bit	Y1 8bit	Cr0 8bit								

- 当 JOUTSWAP 位为 “101” 时

b63	b56	b55	b48	b47	b40	b39	b32	b31	b24	b23	b16	b15	b8	b7	b0
Cb1 8bit	Y2 8bit	Cr1 8bit	Y3 8bit	Cb0 8bit	Y0 8bit	Cr0 8bit	Y1 8bit								

- 当 JOUTSWAP 位为 “110” 时

b63	b56	b55	b48	b47	b40	b39	b32	b31	b24	b23	b16	b15	b8	b7	b0
Y3 8bit	Cr1 8bit	Y2 8bit	Cb1 8bit	Y1 8bit	Cr0 8bit	Y0 8bit	Cb0 8bit								

- 当 JOUTSWAP 位为 “111” 时

b63	b56	b55	b48	b47	b40	b39	b32	b31	b24	b23	b16	b15	b8	b7	b0
Cr1 8bit	Y3 8bit	Cb1 8bit	Y2 8bit	Cr0 8bit	Y1 8bit	Cb0 8bit	Y0 8bit								

(6) 输出符号数据

在压缩时输出符号数据。此模块以 16bit 为单位处理输出符号数据。

如果符号数据为奇数符号长度 (尾数), 最后的符号就输出 “H'D9FF”。

能通过 JIFCNT 的 JOUTSWAP 位更改输出符号数据的数据配置。

41.3.2 解压

(1) 处理概要

解压处理的运行流程如下所示。

1. 启动 JPEG 核心。
2. 将符号数据从外部缓冲器传送到此模块。
在设定输入符号数据停止计数 ON 的情况下，每次读完 JIFSDC 设定数据量的符号数据时就停止读操作。通过将 JIFDCNT 的 JINRCMD 位置 “1”，重新开始读操作。
当 JIFDCNT 的 JINRINI 位为 “0” 时，从上次传送地址的后续地址重新开始读目标地址；当 JINRINI 位为 “1” 时，从 JIFDSA 设定的地址重新开始读目标地址。在检测到符号结束时也停止读操作。
在设定输入符号数据停止计数 OFF 的情况下，一直读到检测到符号结束为止。
只要未检测到符号结束，此模块就一直读符号数据，因此有可能读到大于等于符号数据长度的数据。
3. 将符号数据输入到 JPEG 核心部。
在 JPEG 核心部，以 MCU 为单位随时进行处理。
4. 以 MCU 为单位将图像数据从此模块传送到外部缓冲器。
在设定输出图像数据行停止计数 ON 的情况下，每次写完 JIFDDL 设定行数的图像数据时就停止写操作。通过将 JIFDCNT 的 DOUTRCMD 位置 “1”，重新开始写操作。
当 JIFDCNT 的 DOUTRINI 位为 “0” 时，从上次传送地址的后续地址重新开始写目标地址；当 DOUTRINI 位为 “1” 时，从 JIFDDA 设定的地址重新开始写目标地址。
在 1 个画面的图像数据传送结束时也停止写操作。
在设定输出图像数据行停止计数 OFF 的情况下，一直写到 1 个画面的图像数据结束为止。
5. 在 1 个画面的数据处理结束时结束解压。

(a) 初始设定

- JINTE0 的 INT3 位为 “0” 的情况
在设定 JPEG 核心和输入/输出缓冲器以及给外部缓冲器准备符号数据后，将 JCCMD 的 JSRT 位置 “1” 并且启动此模块。
- JINTE0 的 INT3 位为 “1” 的情况
在设定 JPEG 核心和输入缓冲器以及给外部缓冲器准备符号数据后，将 JCCMD 的 JSRT 位置 “1” 并且启动此模块。
如果在将符号数据解压后能读图像尺寸和像素格式，就将 JINTS0 的 INS3 置位，暂停解压处理。
在读图像尺寸和像素格式后，设定输出缓冲器。
在进行中断处理后，通过将 JCCMD 的 JRST 位置 “1”，重新开始解压处理。

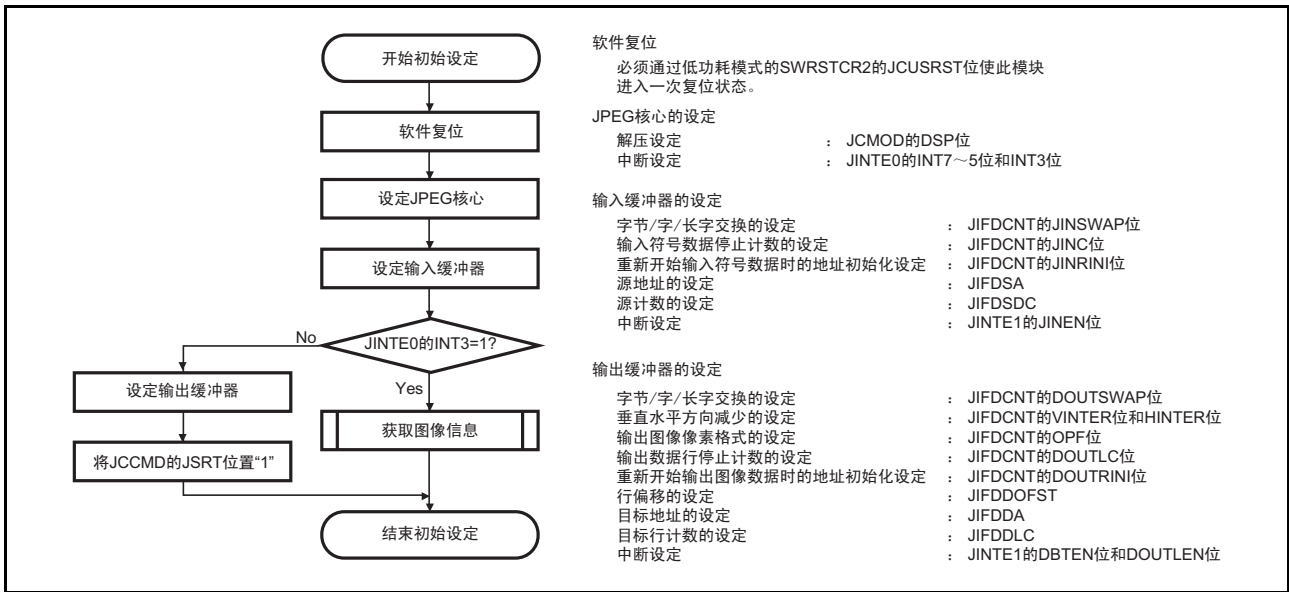


图 41.5 解压的初始设定流程

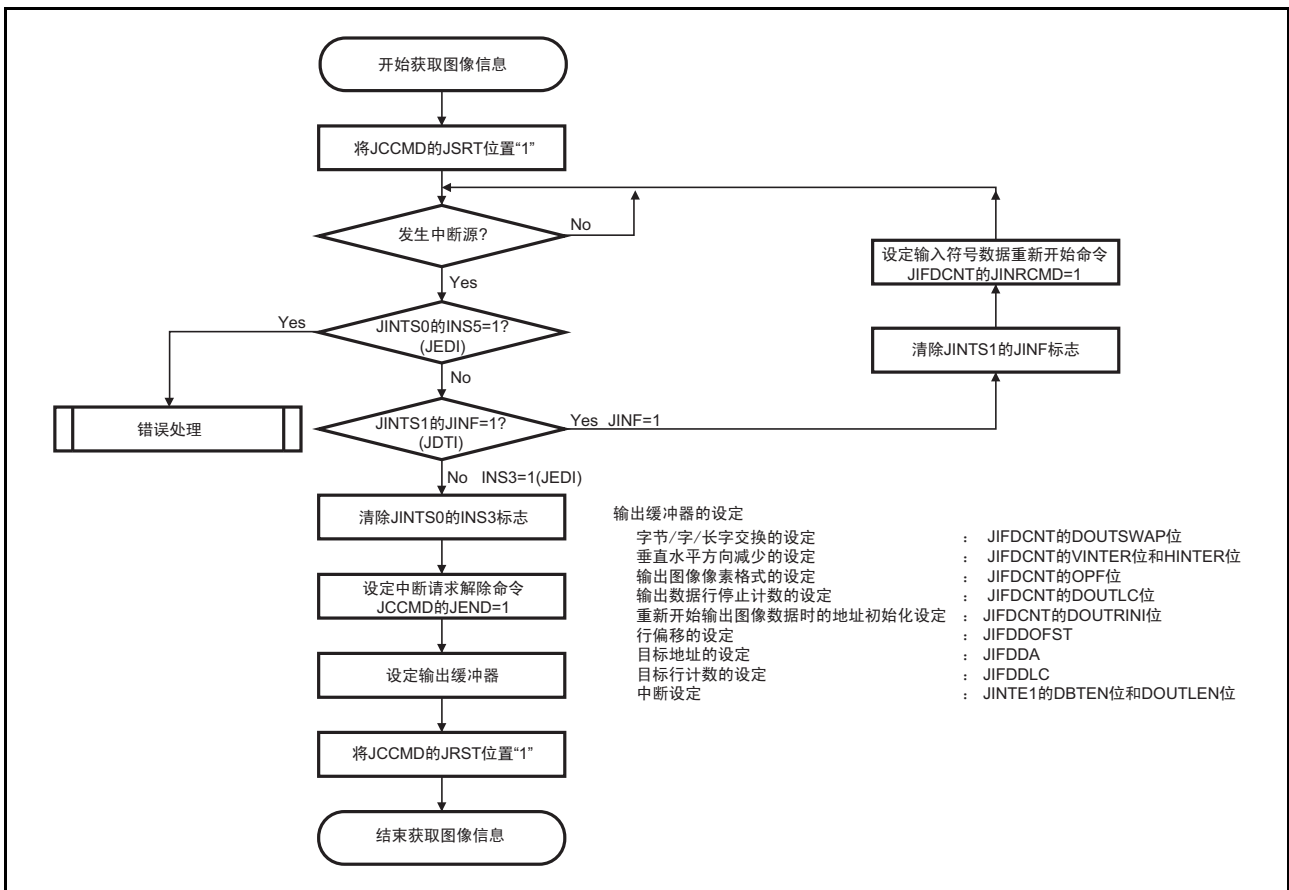


图 41.6 图像信息的捕捉流程

(b) 解压处理

解压处理时的流程如下所示。

- 如果在 JPEG 核心部解压处理结束，就将中断状态 JINTS0 的 INS6 位置 “1”。因为还有未传送的图像数据，所以此模块继续进行处理。如果最后图像数据的传送结束，就将 JINTS1 的 DBTF 位置 “1”。通过给 INTS6 位写 “0” 来清除中断源。但是，即使给 INTS6 位写 “0”，也无法清除被中断源置为有效的中断请求。必须通过设定中断请求解除命令（将 JCCMD 的 JEND 位置 “1”），清除中断请求。
- 一旦解压处理结束并且全部图像数据的传送结束，就将中断状态 JINTS1 的 DBTF 标志置 “1”。如果 JINTE1 的 DBTEN 位已被置 “1”，就产生中断。通过给 DBTF 位写 “0” 来清除中断源。
- 在设定输入符号数据停止计数 ON 的情况下，一旦读完 JIFSDC 设定数据量的符号数据，就将 JINTS1 的 JINF 标志置 “1” 并且停止读操作。如果 JINTE1 的 JINEN 位已被置 “1”，就产生中断。通过给 JINF 位写 “0” 来清除中断源。

通过将 JIFDCNT 的 JINRCMD 位置 “1”，重新开始读操作。

当 JIFDCNT 的 JINRINI 位为 “0” 时，从上次传送地址的后续地址重新开始读目标地址；当 JINRINI 位为 “1” 时，从 JIFDSA 设定的地址重新开始读目标地址。

- 在设定输出图像数据停止计数 ON 的情况下，一旦写完 JIFDDL 设定行数的图像数据，就将 JINT1 的 DOUTLF 标志置 “1” 并且停止写操作。如果将 JINTE1 的 DOUTLEN 位已被置 “1”，就产生中断。通过给 DOUTLF 位写 “0” 来清除中断源。

通过将 JIFDCNT 的 DOUTRCMD 位置 “1”，重新开始写操作。

当 JIFDCNT 的 DOUTRINI 位为 “0” 时，从上次传送地址的后续地址重新开始写目标地址；当 DOUTRINI 位为 “1” 时，从 JIFDDA 设定的地址重新开始写目标地址。

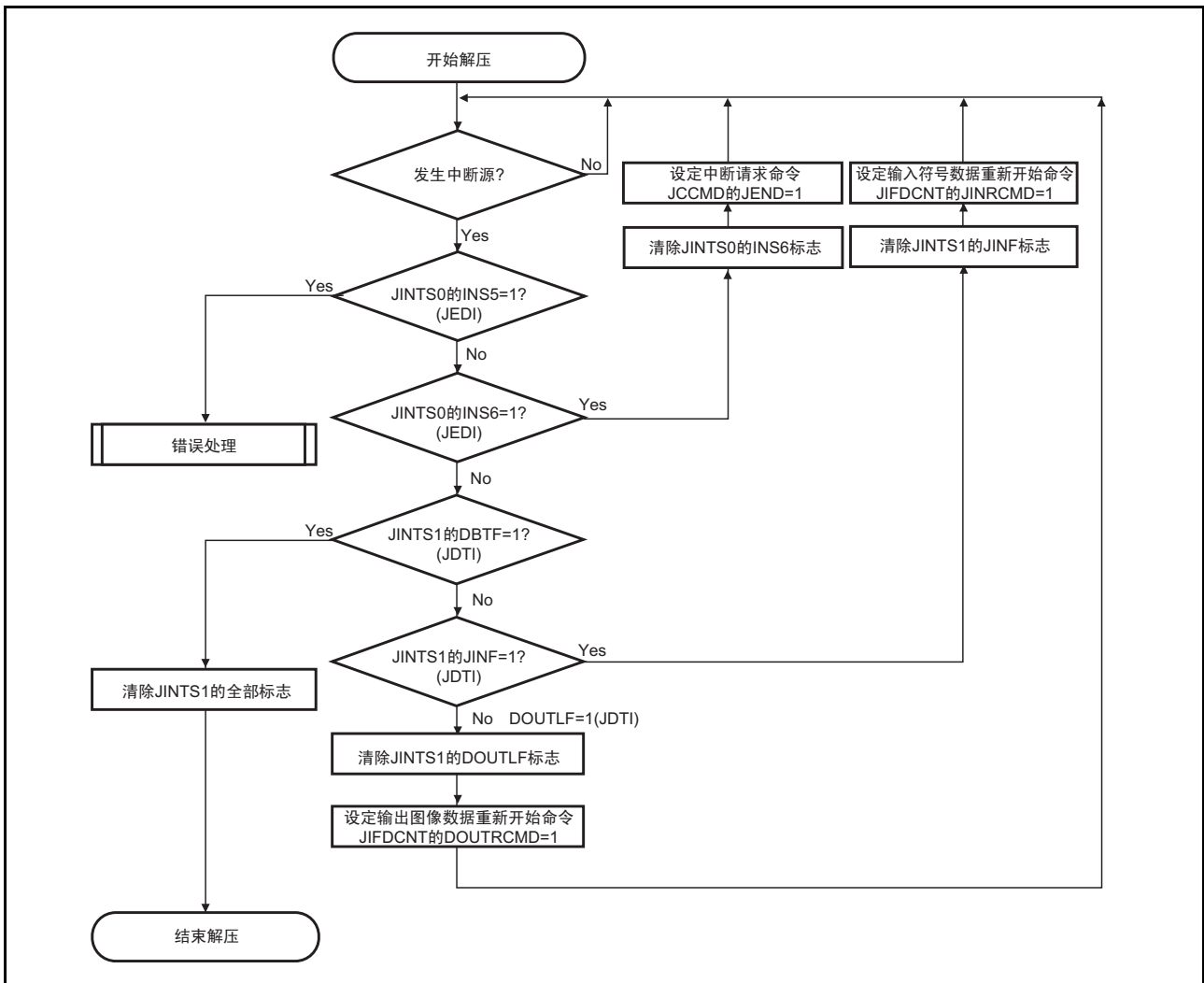


图 41.7 解压的处理流程

(c) 错误处理

如果中断状态 JINTS0 的 INS5 位为“1”，输入的 JPEG 符号数据就有错误并且此模块已经结束解压处理。必须通过读 JCDERR 的 ERR 位，判断错误源。对于被此中断源置为有效的中断信号，不能通过清除中断状态将其置为无效。必须通过设定中断信号解除命令（将 JCCMD 的 JEND 位置“1”），清除中断请求。

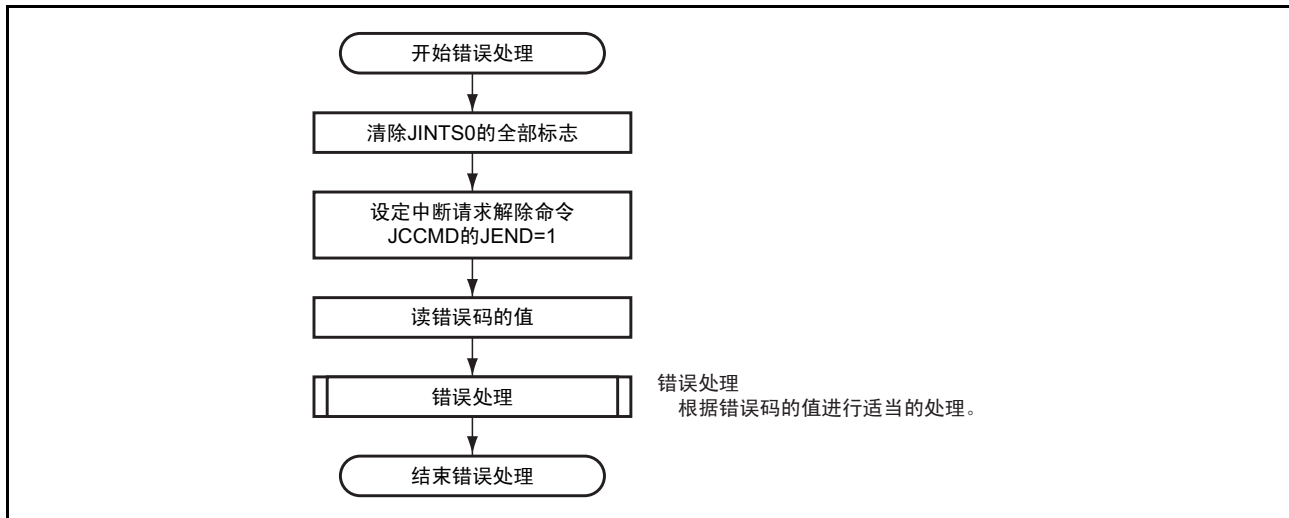


图 41.8 错误处理流程

(2) 输入 JPEG 符号数据

解压处理时的处理对象标记为 SOI、SOF0、SOS、DQT、DHT、DRI、RSTm、EOI。除以下记载的错误对象标记以外，即使检测到其他标记，也进行跳读。

能通过 JIFDCNT 的 JINSWAP 位更改输入符号数据的数据配置。

(3) JPEG 解压错误

(a) 错误标记

在解压处理时，如果在压缩数据解析中发生标记错误，就通过代码判断错误种类，将表 41.3 所示的代码值设定到 JCDERR 的 ERR 位。如果检测到错误，此模块就产生中断信号并且结束处理。在开始下一帧的处理和总线复位时，保存的代码值为默认值（B'1010）。

表 41.3 解压的错误码

代码值 (CODE)	错误内容
B'0000	正常
B'0001	未检测到 SOI。在检测到 EOI 前未检测到 SOI。
B'0010	检测到 SOF1 ~ SOFF。
B'0011	检测到对象以外的像素格式。
B'0100	SOF 精度异常。检测到不是“8”。
B'0101	DQT 精度异常。检测到不是“0”。
B'0110	构件异常 1。检测到 SOF0 标头构件数不是“1”、“3”或者“4”。
B'0111	构件异常 2。SOF0 标头构件数和 SOS 的构件数不同。
B'1000	在检测到 SOS 时，未检测到 SOF0、DQT 和 DHT。
B'1001	未检测到 SOS。在检测到 EOI 前未检测到 SOS。
B'1010	未检测到 EOI (默认)。
B'1011	检测到重新启动间隔数据量错误。
B'1100	检测到图像尺寸错误。
B'1101	检测到最后 MCU 数据量错误。
B'1110	检测到块数据量错误。

(b) 霍夫曼符号化段错误

在解压处理时，如果在压缩数据解析中因霍夫曼符号化段的位取反或者数据丢失引起错误而使解码数据量发生增减，就判断错误种类并且将错误码设定到 JCDERR 的 ERR 位。段错误码如表 41.4 所示。只有在将 JINTE0 的 INT7 ~ INT5 的对应位置“1”时，此模块才设定错误码，产生中断信号并且结束处理。在开始下一帧的处理和总线复位时，保存的代码值为默认值 (B'1010)。

因为检测解码数据量的增减，如果在霍夫曼符号化段内有错误而解码数据量不发生增减，就检测不到此错误。

【例】 像素格式的设定为 YCbCr422、DRI=2、X=80Pixel、Y=8Pixel 时的霍夫曼符号化段的数据量

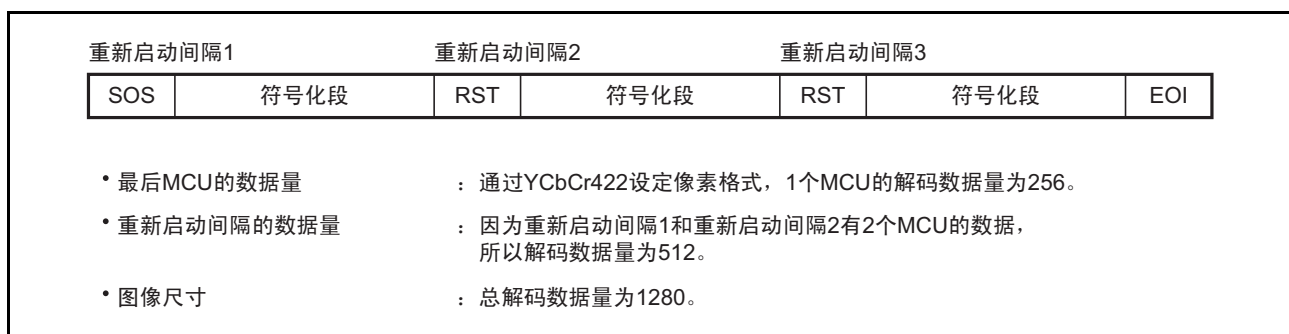


图 41.9 霍夫曼符号化段

表 41.4 段错误码

代码值 (CODE)	错误内容
B'0000	正常
B'1011	重新启动间隔数据量错误 将 DRI 标记段规定的的数据量和各间隔的数据量进行比较, 如果检测到不同的间隔, 就设定解压错误 CODE[1011]。短于重新启动间隔的最后间隔不是比较对象。 如果没有 DRI 标记段或者规定值为“00”, 即使有 RSTm 标记, 也检测不到错误。RSTm 标记模 8 的 m (m=0 ~ 7) 也不是错误检测对象。 如果将 JINTE0 的 INT7 位置“0”, 就检测不到此错误。
B'1100	图像尺寸错误 将根据帧参数规定的行数以及每行的样本数算出的图像数据量和 SOS ~ EOI 的总数据量 (像素单位) 进行比较, 如果不同, 就设定为解压错误 CODE[1100]。如果将 JINTE0 的 INT6 位置“0”, 就检测不到此错误。因为图像数据量以 MCU 为单位, 所以算出时的行数和每行样本数必须以 MCU 为单位进行四舍五入的尾数处理。
B'1101	最后 MCU 数据量错误 检查 EOI 检测时的 MCU 数据量是否以 MCU 为单位并且检测有无尾数。如果同时发生 CODE[1100], 就优先 CODE[1100]。 如果将 JINTE0 的 INT5 位置“0”, 就检测不到此错误。
B'1110	块数据量错误 检查 1 块是否以 8x8 位为单位并且检测有无尾数。如果将 JINTE0 的 INT7 ~ INT5 位全部置“0”, 就检测不到此错误。

41.3.3 解压时的输出像素格式

此模块能解压用 YCbCr422 和 YCbCr420 格式建立的 JPEG 符号数据。但是, 输出图像的像素格式为 YCbCr422、ARGB8888、RGB565。将解压的数据转换为输出像素格式的流程如下所示。

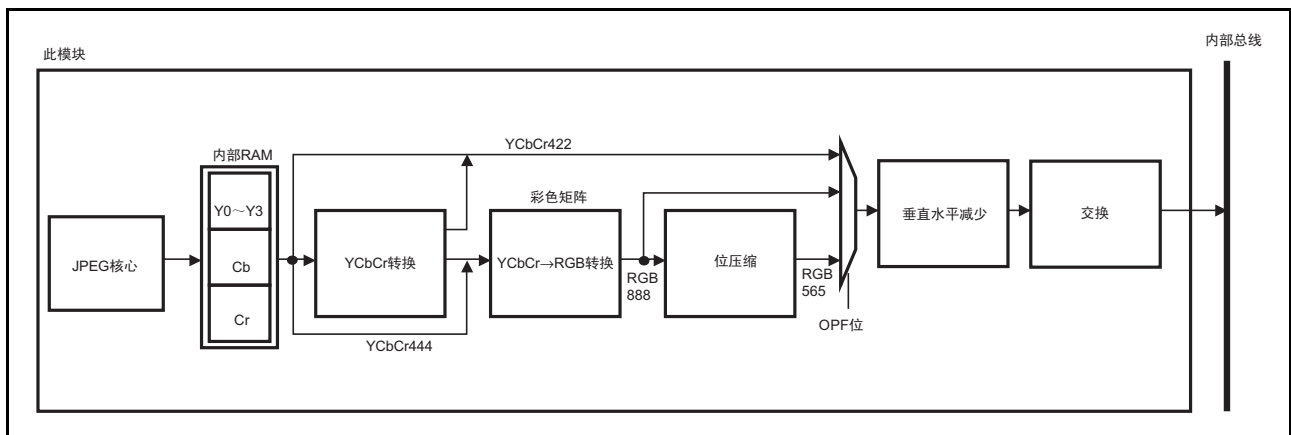


图 41.10 解压时的输出像素格式转换框图

(1) 内部 RAM

以 MCU 为单位，将通过 JPEG 核心解码的数据保存到此模块的内部 RAM。

(2) YCbCr 转换

在输出 ARGB8888 或者 RGB565 格式时，将 YCbCr422 或者 YCbCr420 格式的数据转换为 YCbCr444 格式。
在输出 YCbCr422 格式时，将 YCbCr420 格式的数据转换为 YCbCr422 格式。
通过单纯的内插进行此转换。

(3) YCbCr→RGB 的转换

将 YCbCr444 格式的数据转换为 RGB888 格式。

计算式如下：

$$R=1.000Y+1.402Cr$$

$$G=1.000Y-0.344Cb-0.714Cr$$

$$B=1.000Y+1.772Cb$$

(4) 位压缩

将 RGB888 的数据压缩为 RGB565 的数据。删除 Red 和 Blue 的低 3 位以及 Green 的低 2 位。

(5) 输出像素格式的选择

通过 JIFDCNT 的 OPF 位选择要输出的像素格式。

像素格式的数据配置（将 JIFDCNT 的 DOUTSWAP 位置“000”的情况）如下。

- YCbCr422(32bit/pixel)

b31	b24 b23	b16 b15	b8 b7	b0
Y0 8bit	Cb 8bit	Y1 8bit	Cr 8bit	

- ARGB8888(32bit/pixel)

b31	b24 b23	b16 b15	b8 b7	b0
*	Red 8bit	Green 8bit	Blue 8bit	

【注】 * JIFDADT 的 ALPHA[7:0] 指定的值

- RGB565(16bit/pixel)

b15	b11 b10	b5 B4	b0
Red 5bit	Green 6bit	Blue 5bit	

(6) 垂直 / 水平减少

能通过 JIFDCNT 的 VINTER 位和 HINTER 位，按水平方向或者垂直方向减少输出数据。

减少行如图 41.11 ~ 图 41.13 所示。

在输出 ARGB8888 或者 RGB565 格式时，用 1 像素 / 1 格表示。

在输出 YCbCr422 格式时，用 Y0Cb0Y1Cr0/1 格表示。

因为以 MCU 为单位进行处理，所以水平 / 垂直的块数因解压的像素格式而不同。

n 和 m 的值分别如下。

[水平方向]

表 41.5 水平块数

压缩格式	输出格式	n
YCbCr422	YCbCr422	1
YCbCr422	ARGB8888、RGB565	2
YCbCr420	YCbCr422	1
YCbCr420	ARGB8888、RGB565	2

[垂直方向]

表 41.6 垂直块数

压缩格式	输出格式	m
YCbCr422	YCbCr422	1
YCbCr422	ARGB8888、RGB565	1
YCbCr420	YCbCr422	2
YCbCr420	ARGB8888、RGB565	2

- 减少到1/2
减少偶数行。

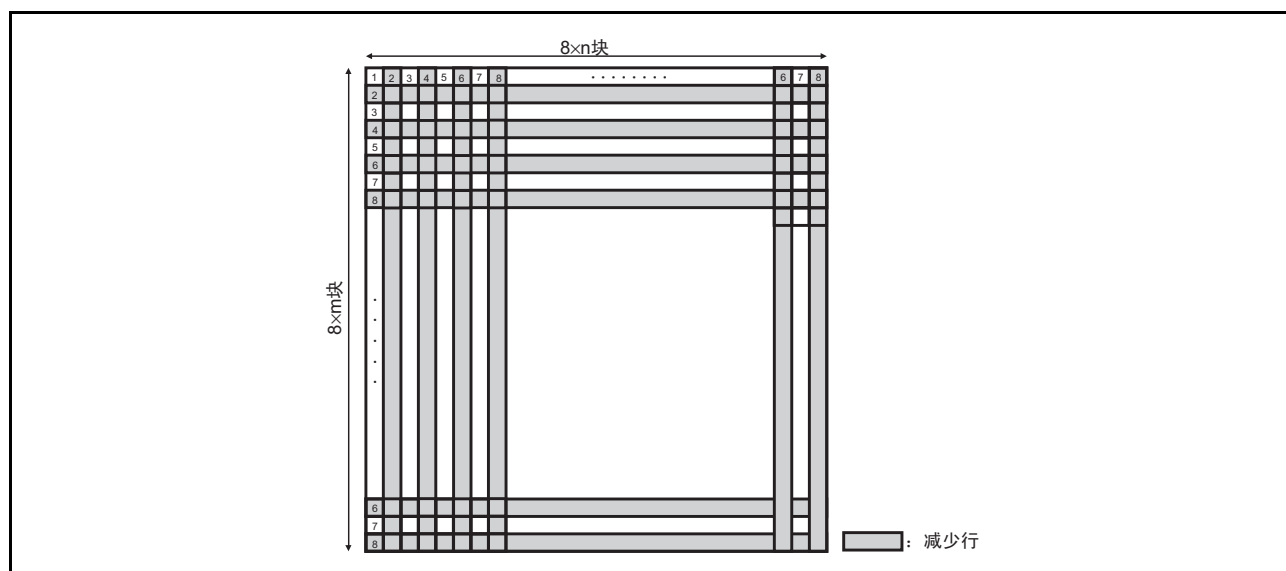


图 41.11 选择减少到 1/2 时的 MCU 图

- 减少到 1/4
减少第 2、3、4 行。

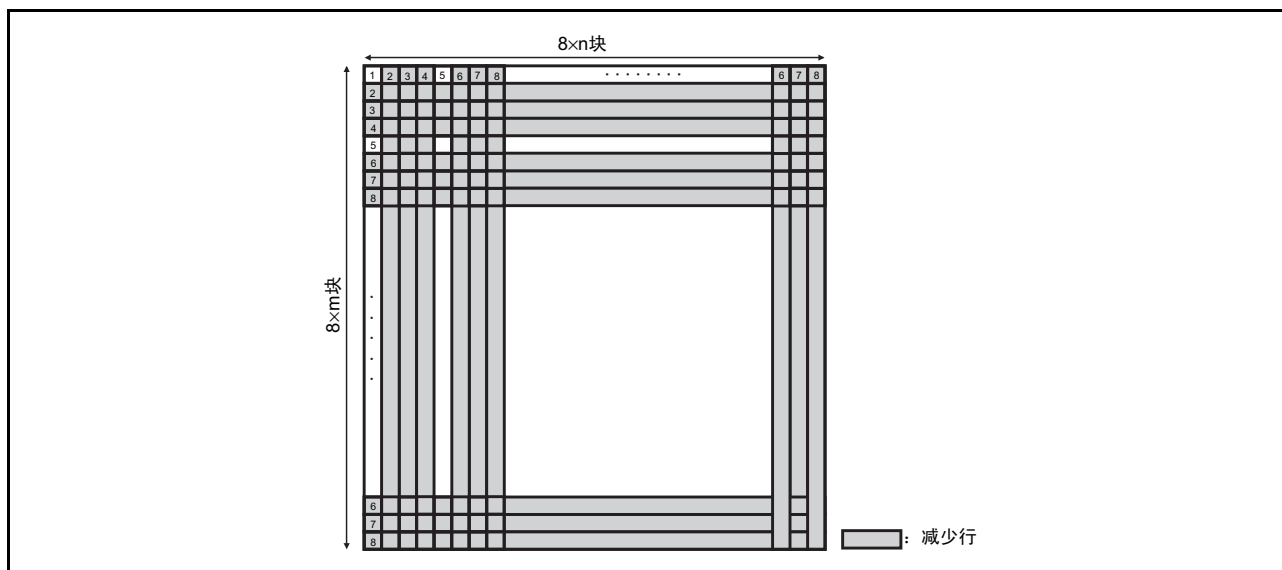


图 41.12 选择减少到 1/4 时的 MCU 图

- 减少到 1/8
减少第 2、3、4、5、6、7、8 行。

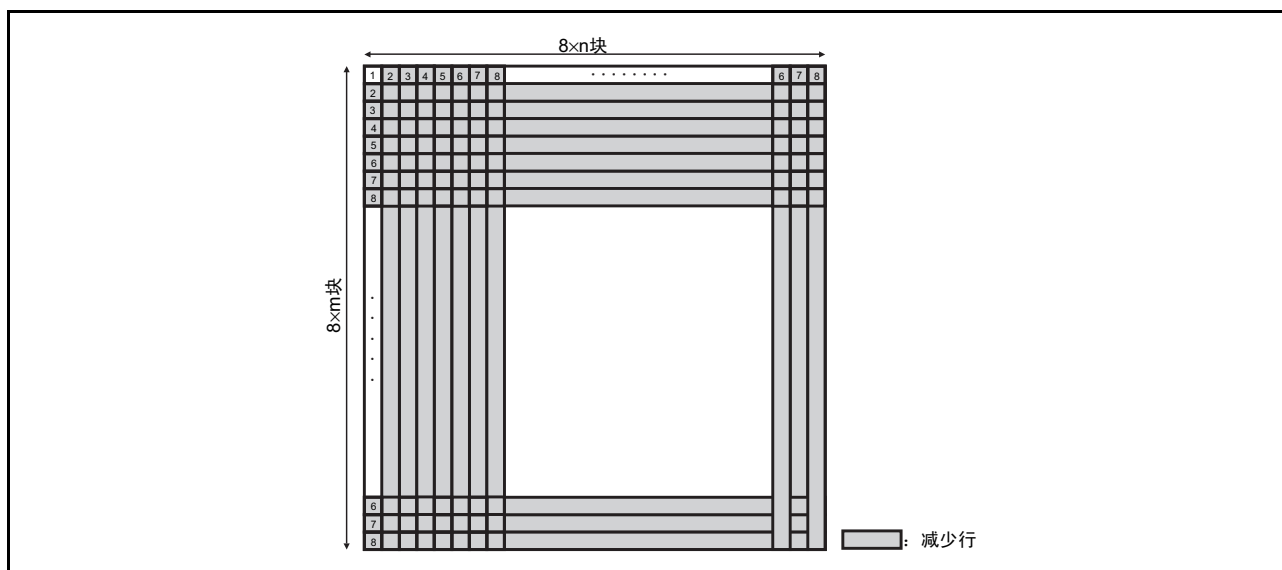


图 41.13 选择减少到 1/8 时的 MCU 图

(7) 交换

能通过 JIFDCNT 的 DOUTSWAP 位更改数据配置。

41.3.4 图像数据的保存

将图像数据保存到缓冲器的保存图如下所示：

起始地址 压缩：JIFESA、解压：JIFDDA

水平方向尺寸 压缩、解压：JCHSZU、JCHSZD

垂直方向尺寸 压缩、解压：JCVSZU、JCVSZD

偏移 压缩：JIFESOFST、解压 JIFDDOFST

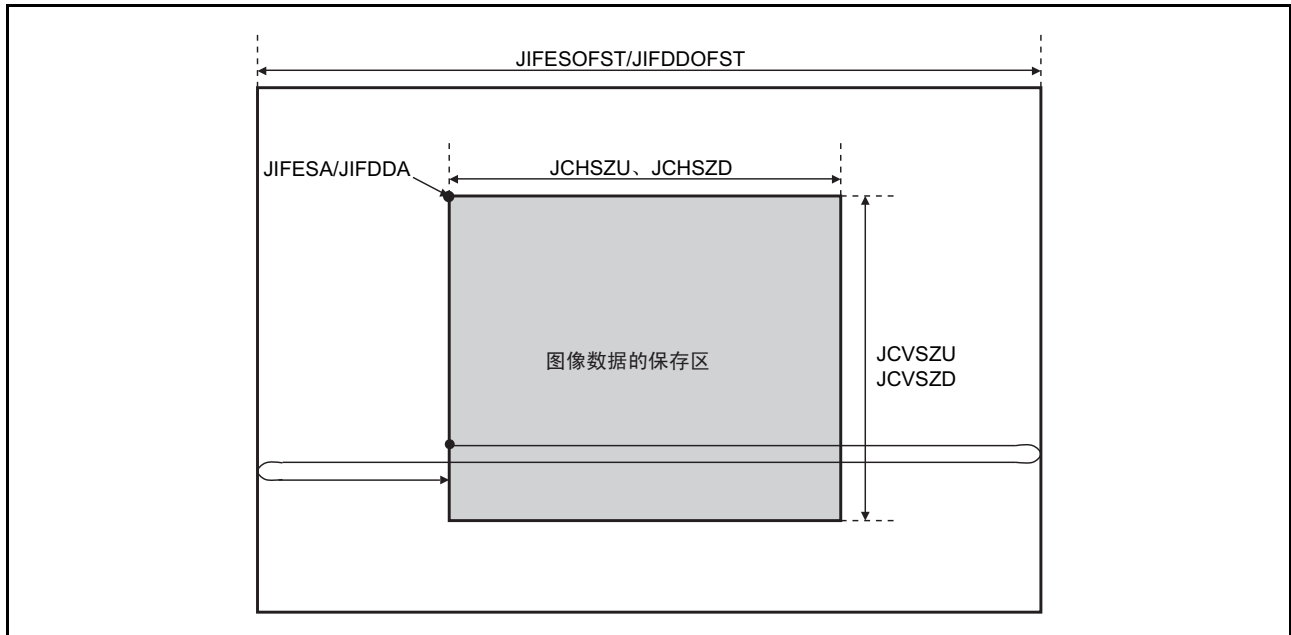


图 41.14 图像数据的保存图

41.4 中断

此模块有压缩解压处理中断请求 (JEDI) 和数据传送中断请求 (JDTI) 共 2 种中断请求。压缩解压处理中断请求是与压缩 / 解压处理有关的中断源，数据传送中断请求是与数据传送有关的中断源。中断请求的解除方法因中断源而不同。

41.4.1 压缩解压处理中断请求 (JEDI)

中断状态 JINTS0 的标志是压缩解压处理产生的中断源。对于被此中断源置为有效的中断请求，不能通过清除中断状态将其置为无效。必须通过设定中断信号解除命令 (将 JCCMD 的 JEND 位置 “1”)，清除中断请求。如果将中断状态 JINTS0 的标志置 “1”，就将压缩解压处理中断请求输出到中断控制器。

(1) 压缩

- JPEG 压缩处理结束

当中断状态 JINTS0 的 INS6 位为 “1” 时，表示 JPEG 压缩处理已经结束。一旦符号数据传送结束，压缩处理就结束。

(2) 解压

- JPEG 解压处理结束
当中断状态 JINTS0 的 INS6 位为 “1” 时，表示 JPEG 解压处理已经结束。一旦图像数据传送结束，解压处理就结束。
- JPEG 解压错误的发生
当中断状态 JINTS0 的 INS5 位为 “1” 时，表示输入的 JPEG 符号数据有错误并且解压处理已经结束。必须通过读错误码 (JCDERR 的 ERR 位)，判断错误源。如果中断设定 JINTE0 的 INT7、INT6、INT5 位已被置 “1”，就产生此中断。
- 图像尺寸/像素格式信息的读请求
当中断状态 JINTS0 的 INS3 位为 “1” 时，表示输入了 JPEG 符号数据并且能读像素格式和图像尺寸信息。因为 JPEG 解压处理为暂停状态，所以必须在存取各寄存器后通过设定处理停止解除命令，重新开始 JPEG 解压处理。如果中断设定 JINTE0 的 INT3 位已被置 “1”，就产生此中断。

41.4.2 数据传送处理中断请求 (JDTI)

中断状态 JINTS1 的标志是与图像 / 符号数据传送有关的中断源。对于被此中断源置为有效的中断请求，不能通过清除将其置为无效。

(1) 压缩

- 在读完输入图像设定行时产生
当中断状态 JINTS1 的 DINLF 位为 “1” 时，表示 JIFESLC 设定行数的图像数据已经传送结束。必须给外部缓冲器准备下一个图像数据，并且从外部缓冲器重新开始传送处理。如果中断设定 JINTE1 的 DINLEN 位已被置 “1”，就输出数据传送处理中断请求。
- 在写完输出符号设定数据时产生
当中断状态 JINTS1 的 JOUTF 位为 “1” 时，表示 JIFEDDC 设定符号数据量的传送已经结束。必须给外部缓冲器准备下一个符号数据空区域，并且重新开始传送处理。如果中断设定 JINTE1 的 JOUTEN 位已被置 “1”，就输出数据传送处理中断请求。
- 在全部处理结束时产生
当中断状态 JINTS1 的 CBTF 位为 “1” 时，表示压缩处理和全部符号数据的传送已经结束。如果将中断设定 JINTE1 的 CBTEN 位已被置 “1”，就输出数据传送处理中断请求。

(2) 解压

- 在写完输出图像设定行时产生
当中断状态 JINTS1 的 DOUTLF 位为 “1” 时，表示 JIFDDLC 设定行数的图像数据已经传送结束。必须给外部缓冲器准备下一个图像数据空区域，并且重新开始传送处理。如果中断设定 JINTE1 的 DOUTLEN 位已被置 “1”，就输出数据传送处理中断请求。
- 在读完输入符号设定数据时产生
当中断状态 JINTS1 的 JINF 位为 “1” 时，表示 JIFSDC 设定符号数据量的传送已经结束。必须给外部缓冲器准备下一个符号数据，并且从外部缓冲器重新开始传送处理。如果中断设定 JINTE1 的 JINEN 位已被置 “1”，就输出数据传送处理中断请求。
- 在全部处理结束时产生
当中断状态 JINTS1 的 DBTF 位为 “1” 时，表示解压处理和全部图像数据的传送已经结束。如果中断设定 JINTE1 的 DBTEN 位已被置 “1”，就输出数据传送处理中断请求。

41.5 总线复位处理

通过发行总线复位命令（将 JCCMD 的 BRST 位置“1”）进行总线复位，不能在运行过程中发行总线复位命令。

通过总线复位对以下寄存器进行初始化：

- JPEG 码数据计数高位寄存器 (JCDTCU)
- JPEG 码数据计数中位寄存器 (JCDTCM)
- JPEG 码数据计数低位寄存器 (JCDTCD)
- JPEG 中断状态寄存器0 (JINTS0)
- JPEG 码解码错误寄存器 (JCDERR)
- JPEG 码重新启动寄存器 (JCRST)

41.6 使用时的注意事项

41.6.1 像素格式 YCbCr

此模块在 -128 ~ 127 范围内处理像素格式 YCbCr422 和 YCbCr420 的 CbCr 输入 / 输出值，而显示控制器 4 在 0 ~ 255 范围内处理 CbCr 输入 / 输出值。因此，如果要在该模块和视频显示控制器 4 之间传递像素格式 YCbCr 的数据，就必须通过减 128 或者加 128 来调整 CbCr 值。

42. 采样率转换器

采样率转换器是转换 WMA/MP3/AAC 等各种解码器生成的数据采样率的模块。

42.1 特点

- 数据格式：16 位（立体声/单声道）
- 采样率
输入：可选择 8kHz、11.025kHz、12kHz、16kHz、22.05kHz、24kHz、32kHz、44.1kHz 或者 48kHz。
输出：可选择 8kHz*、16kHz*、32kHz、44.1kHz 或者 48kHz（*：只在选择 44.1kHz 的采样率时）。
- 处理性能：1 个样品的输出间隔最大约为 14 μ s（P0 ϕ =33MHz）
- SNR： ≥ 80 db
- 中断源：5 种
输入数据 FIFO 空、输出数据 FIFO 满、输出数据 FIFO 重写、输出数据 FIFO 下溢、转换处理结束
- DMA 传送源：2 种
输入数据 FIFO 空、输出数据 FIFO 满
- 模块待机模式
在不使用时停止给此模块提供时钟，能降低功耗。

框图如图 42.1 所示。

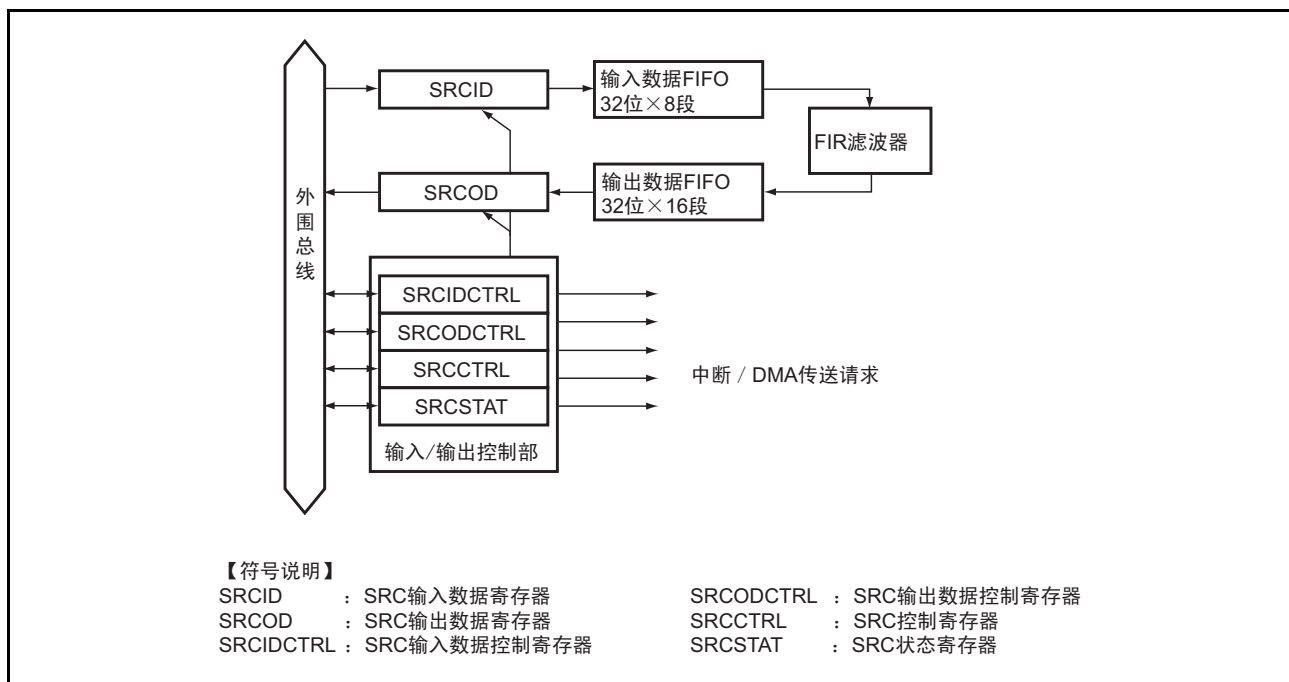


图 42.1 框图

42.2 寄存器说明

寄存器结构如表 42.1 所示。

表 42.1 寄存器结构

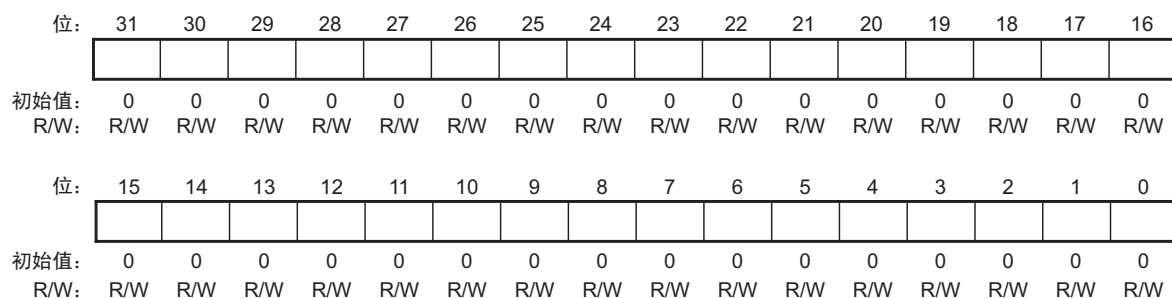
通道	寄存器名称	略称	R/W	初始值	地址	存取长度
0	输入数据寄存器 _0	SRCID_0	R/W	H'00000000	H'FFFE7000	16、32
	输出数据寄存器 _0	SRCOD_0	R	H'00000000	H'FFFE7004	16、32
	输入数据控制寄存器 _0	SRCIDCTRL_0	R/W	H'0000	H'FFFE7008	16
	输出数据控制寄存器 _0	SRCODCTRL_0	R/W	H'0000	H'FFFE700A	16
	控制寄存器 _0	SRCCTRL_0	R/W	H'0000	H'FFFE700C	16
	状态寄存器 _0	SRCSTAT_0	R/(W)*	H'0002	H'FFFE700E	16
1	输入数据寄存器 _1	SRCID_1	R/W	H'00000000	H'FFFE7800	16、32
	输出数据寄存器 _1	SRCOD_1	R	H'00000000	H'FFFE7804	16、32
	输入数据控制寄存器 _1	SRCIDCTRL_1	R/W	H'0000	H'FFFE7808	16
	输出数据控制寄存器 _1	SRCODCTRL_1	R/W	H'0000	H'FFFE780A	16
	控制寄存器 _1	SRCCTRL_1	R/W	H'0000	H'FFFE780C	16
	状态寄存器 _1	SRCSTAT_1	R/(W)*	H'0002	H'FFFE780E	16
2	输入数据寄存器 _2	SRCID_2	R/W	H'00000000	H'FFFE8000	16、32
	输出数据寄存器 _2	SRCOD_2	R	H'00000000	H'FFFE8004	16、32
	输入数据控制寄存器 _2	SRCIDCTRL_2	R/W	H'0000	H'FFFE8008	16
	输出数据控制寄存器 _2	SRCODCTRL_2	R/W	H'0000	H'FFFE800A	16
	控制寄存器 _2	SRCCTRL_2	R/W	H'0000	H'FFFE800C	16
	状态寄存器 _2	SRCSTAT_2	R/(W)*	H'0002	H'FFFE800E	16

【注】 * bit15 ~ 6 和 bit4 是只读位。bit5 和 bit3 只能在读“1”后写“0”。

42.2.1 输入数据寄存器 (SRCID)

SRCID 是 32 位可读写寄存器，用于在转换采样率前输入数据，位的读取值都为“0”。将写到 SRCID 的数据保存到 8 段的输入数据 FIFO 中。在输入数据 FIFO 的数据量为 8 时，SRCID 的写操作无效。

在立体声数据的情况下，bit31 ~ 16 保存 Lch 的数据，bit15 ~ 0 保存 Rch 的数据。在单声道数据的情况下，bit31 ~ 16 保存的数据有效，bit15 ~ 0 保存的数据无效。



转换处理对象的数据调整因 SRCIDCTRL 的 IED 位的设定值而不同。SRCIDCTRL 的 IED 位的设定值和数据调整的关系如表 42.2 所示。

表 42.2 转换数据的调整

IED	Lch[15:8]	Lch[7:0]	Rch[15:8]	Rch[7:0]
0	SRCID[31:24]	SRCID[23:16]	SRCID[15:8]	SRCID[7:0]
1	SRCID[23:16]	SRCID[31:24]	SRCID[7:0]	SRCID[15:8]

42.2.2 输出数据寄存器 (SRCOD)

SRCOD 是 32 位可读寄存器，用于输出采样率转换后的数据。能从 SRCOD 读保存在 16 段输出数据 FIFO 中的数据。在开始转换处理后输出数据 FIFO 的数据量为“0”时，读取和前次相同的值。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

SRCOD 的内容调整因 SRCODCTRL 的 OCH 位和 OED 位的设定值而不同。SRCODCTRL 的 OCH 位和 OED 位的设定值和保存在 SRCOD 中的数据调整的关系如表 42.3 所示。

表 42.3 SRCOD 的调整

OCH	OED	SRCOD[31:24]	SRCOD[23:16]	SRCOD[15:8]	SRCOD[7:0]
0	0	Lch[15:8]	Lch[7:0]	Rch[15:8]*2	Rch[7:0]*2
	1	Lch[7:0]	Lch[15:8]	Rch[7:0]*2	Rch[15:8]*2
1*1	0	Rch[15:8]	Rch[7:0]	Lch[15:8]	Lch[7:0]
	1	Rch[7:0]	Rch[15:8]	Lch[7:0]	Lch[15:8]

【注】 *1 当处理单声道数据时，不能设定。

*2 当处理单声道数据时，为无效数据。

42.2.3 输入数据控制寄存器 (SRCIDCTRL)

SRCIDCTRL 是 16 位可读写寄存器，设定输入数据字节序格式，允许或者禁止中断请求以及设定触发数据的个数。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	IED	IEN	—	—	—	—	—	—	IFTRG[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

位	位名称	初始值	R/W	说明
15 ~ 10	—	全 0	R	保留位 读写值都为“0”。
9	IED	0	R/W	输入数据字节序指定 指定输入数据的字节序格式。 0: 大端法 1: 小端法
8	IEN	0	R/W	输入数据空中断允许 在输入 FIFO 的数据量小于等于 IFTRG[1:0] 位设定的触发个数并且状态寄存器 (SRCSTAT) 的 IINT 位被置“1”时，允许或者禁止产生输入数据空中断请求。 0: 禁止输入数据空中断请求 1: 允许输入数据空中断请求
7 ~ 2	—	全 0	R	保留位 读写值都为“0”。
1、0	IFTRG[1:0]	00	R/W	输入 FIFO 数据触发个数 指定状态寄存器 (SRCSTAT) 的 IINT 位的置位条件。当保存在输入 FIFO 中的输入数据量小于等于以下所设触发个数时，将 IINT 位置“1”。 00: 0 01: 2 10: 4 11: 6

42.2.4 输出数据控制寄存器 (SRCODCTRL)

SRCODCTRL 是 16 位可读写寄存器，设定输出数据的通道交换和字节序格式，允许或者禁止中断请求以及设定触发数据的个数。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	OCH	OED	OEN	—	—	—	—	—	—	OFTRG[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

位	位名称	初始值	R/W	说明
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10	OCH	0	R/W	输出数据通道交换 指定输出数据寄存器 (SRCOD) 的通道交换。在转换单声道数据时，不能置“1”。 0: 不交换通道 (和输入数据的顺序相同) 1: 交换通道 (和输入数据的顺序相反)
9	OED	0	R/W	输出数据字节序指定 指定输出数据的字节序格式。 0: 大端法 1: 小端法
8	OEN	0	R/W	输出数据满中断允许 在输出 FIFO 的数据量大于等于 OFTRG[1:0] 位设定的触发个数并且状态寄存器 (SRCSTAT) 的 OINT 位被置位时，允许或者禁止产生输出数据满中断请求。 0: 禁止输出数据满中断请求 1: 允许输出数据满中断请求
7 ~ 2	—	全 0	R	保留位 读写值都为“0”。
1、0	OFTRG[1:0]	00	R/W	输出 FIFO 数据触发个数 指定状态寄存器 (SRCSTAT) 的 OINT 位的置位条件。当保存在输出 FIFO 中的输出数据量大于等于以下所设触发个数时，将 OINT 位置“1”。 00: 1 01: 4 10: 8 11: 12

42.2.5 控制寄存器 (SRCCTRL)

SRCCTRL 是 16 位可读写寄存器，允许或者禁止模块运行和中断请求，设定转储清除处理、内部工作存储器的清除处理和输入 / 输出采样率。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	CEEN	SRCEN	UDEN	OVEN	FL	CL	IFS[3:0]			—	OFS[2:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名称	初始值	R/W	说明												
15、14	—	全 0	R	保留位 读写值都为“0”。												
13	CEEN	0	R/W	转换处理结束中断允许 在结束转储清除处理后，当读到全部输出数据并且状态寄存器 (SRCSTAT) 的 CEF 位被置“1”时，允许或者禁止产生转换处理结束中断。 0: 禁止产生转换处理结束中断 1: 允许产生转换处理结束中断												
12	SRCEN	0	R/W	模块允许 设定允许或者禁止此模块的运行。如果在 SRCEN 位为“0”时写“1”，就清除内部工作存储器。 0: 禁止此模块的运行 1: 允许此模块的运行 【注】 在 SRCEN 位为“1”时，不能改变以下位的设定值。 <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>寄存器</th> <th>位</th> <th>位名</th> </tr> </thead> <tbody> <tr> <td>SRCIDCTRL</td> <td>9</td> <td>IED</td> </tr> <tr> <td>SRCODCTRL</td> <td>10、9</td> <td>OCH、OED</td> </tr> <tr> <td>SRCCTRL</td> <td>7~4、2~0</td> <td>IFS[3:0]、OFS[2:0]</td> </tr> </tbody> </table>	寄存器	位	位名	SRCIDCTRL	9	IED	SRCODCTRL	10、9	OCH、OED	SRCCTRL	7~4、2~0	IFS[3:0]、OFS[2:0]
寄存器	位	位名														
SRCIDCTRL	9	IED														
SRCODCTRL	10、9	OCH、OED														
SRCCTRL	7~4、2~0	IFS[3:0]、OFS[2:0]														
11	UDEN	0	R/W	输出 FIFO 下溢中断允许 在输出 FIFO 的数据量为“0”的状态下读输出数据 FIFO 并且状态寄存器 (SRCSTAT) 的 UDF 位被置“1”时，允许或者禁止产生输出 FIFO 下溢中断请求。 0: 禁止产生输出 FIFO 下溢中断请求 1: 允许产生输出 FIFO 下溢中断请求												
10	OVEN	0	R/W	输出 FIFO 重写中断允许 在输出 FIFO 满的状态下结束下一个数据的转换处理并且状态寄存器 (SRCSTAT) 的 OVF 位被置“1”时，允许或者禁止产生输出 FIFO 重写中断请求。 如果在 OVEN 位为“1”时发生输出 FIFO 重写中断，就在通过 CPU 存取 SRCSTAT 来清除 OVF 位前停止转换处理，并且停止将转换结果写到数据输出 FIFO。 如果在 OVEN 位为“0”时输出 FIFO 变为空，就能自动清除 OVF 位并且继续转换处理。 0: 禁止输出 FIFO 重写中断请求 1: 允许输出 FIFO 重写中断请求												

位	位名称	初始值	R/W	说明
9	FL	0	R/W	内部工作存储器转储清除 如果给此位写“1”，就开始对保存在输入 FIFO、输入缓冲存储器和中间缓冲存储器中的全部数据进行采样率的转换（转储清除处理）。此位的读取值总是“0”。 当 SRCEN 位为“0”时，即使写“1”也不进行转储清除处理。如果在输入的数据量小于表 42.6 所示值的状态下给 FL 位写“1”，就得不到有效的输出数据，因此不进行转储清除处理而清除内部工作存储器。
8	CL	0	R/W	内部工作存储器清除 如果给此位写“1”，就清除输入 FIFO、输出 FIFO、输入缓冲存储器、中间缓冲存储器和累加器。此位的读取值总是“0”。即使在 SRCEN 位为“0”时，只要给此位写“1”也进行清除。
7~4	IFS[3:0]	0000	R/W	输入采样率 设定输入采样率。 0000: 8.0kHz 0001: 11.025kHz 0010: 12.0kHz 0011: 禁止设定 0100: 16.0kHz 0101: 22.05kHz 0110: 24.0kHz 0111: 禁止设定 1000: 32.0kHz 1001: 44.1kHz 1010: 48.0kHz 1011: 禁止设定 1100: 禁止设定 1101: 禁止设定 1110: 禁止设定 1111: 禁止设定
3	—	全 0	R	保留位 读写值都为“0”。
2~0	OFS[2:0]	0	R/W	输出采样率 设定输出采样率。 000: 44.1kHz 001: 48.0kHz 010: 32.0kHz 011: 禁止设定 100: 8.0kHz* 101: 16.0kHz* 110: 禁止设定 111: 禁止设定 【注】 * OFS=(100、101) 只在将 IFS 设定为“1001”时有效。

在转储清除处理结束后，用以下计算式求作为转换结果得到的输出数据：

$$\text{输出采样数} = \left[(\text{输入数据数} \times n - 1) \times \frac{\text{输出采样率}}{\text{输入采样率} \times n} \right] + 1$$

表 42.4 计算式中的 n 值

OFS 的设定值 (输出采样率 [kHz])	IFS 的设定值 (输入采样率 [kHz])								
	0000 (8.0)	0001 (11.025)	0010 (12.0)	0100 (16.0)	0101 (22.05)	0110 (24.0)	1000 (32.0)	1001 (44.1)	1010 (48.0)
000(44.1)	6	4	4	3	2	2	3	—	1
001(48.0)	6	4	4	3	2	2	3	1	—
010(32.0)	4	8	4	2	4	2	—	2	1
100(8.0)	—	—	—	—	—	—	—	1	—
101(16.0)	—	—	—	—	—	—	—	1	—

因为在输入规定个数的数据前不开始转换处理，所以得不到输出数据。为了获取最初的输出数据所需的输入数据量因 IFS 位和 OFS 位的设定值而不同。IFS 位和 OFS 位的设定值以及需要初始输入的数据量的关系如表 42.5 和表 42.6 所示。

表 42.5 采样率的设定和需要初始输入的数据量

OFS 的设定值 (输出采样率 [kHz])	IFS 的设定值 (输入采样率 [kHz])								
	0000 (8.0)	0001 (11.025)	0010 (12.0)	0100 (16.0)	0101 (22.05)	0110 (24.0)	1000 (32.0)	1001 (44.1)	1010 (48.0)
000(44.1)	38	40	40	43	48	48	43	—	63
001(48.0)	38	40	40	43	48	48	43	32	—
010(32.0)	40	37	40	48	40	48	—	48	63
100(8.0)	—	—	—	—	—	—	—	63	—
101(16.0)	—	—	—	—	—	—	—	63	—

表 42.6 采样率的设定和转储清除所需的输入数据量

OFS 的设定值 (输出采样率 [kHz])	IFS 的设定值 (输入采样率 [kHz])								
	0000 (8.0)	0001 (11.025)	0010 (12.0)	0100 (16.0)	0101 (22.05)	0110 (24.0)	1000 (32.0)	1001 (44.1)	1010 (48.0)
000(44.1)	27	24	24	22	16	16	22	—	1
001(48.0)	27	24	24	22	16	16	22	32	—
010(32.0)	24	29	24	16	24	16	—	16	1
100(8.0)	—	—	—	—	—	—	—	1	—
101(16.0)	—	—	—	—	—	—	—	1	—

42.2.6 状态寄存器 (SRCSTAT)

SRCSTAT 是 16 位可读写寄存器，表示输出 FIFO 和输入 FIFO 的数据量、各中断源的发生状态以及转储清除处理的执行状态。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	OFDN[4:0]				IFDN[3:0]				—	CEF	FLF	UDF	OVF	IINT	OINT		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	
R/W:	R	R	R	R	R	R	R	R	R	R	R/(W)*	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

【注】 * 只能在读“1”后写“0”。

位	位名称	初始值	R/W	说明
15 ~ 11	OFDN[4:0]	00000	R	输出 FIFO 数据计数 表示保存在输出 FIFO 中的数据量。
10 ~ 7	IFDN[3:0]	0000	R	输入 FIFO 数据计数 表示保存在输入 FIFO 中的数据量。
6	—	0	R	保留位 读写值都为“0”。
5	CEF	0	R/(W)*	转换处理结束标志 表示在转储清除处理结束后读到全部的输出数据。 [清除条件] • 在 CEF 位为“1”的状态下读 CEF 位后写“0”时 • 当给 SRCCTRL 的 CL 位写“1”时 • 在 SRCCTRL 的 SRCEN 位为“0”的状态下给 SRCEN 位写“1”时 [置位条件] • 在转储清除处理后输出数据 FIFO 的数据量变为“0”时
4	FLF	0	R	转储清除处理状态标志 表示正在进行转储清除处理。 [清除条件] • 当转储清除处理结束时 • 当给 SRCCTRL 的 CL 位写“1”时 • 在 SRCCTRL 的 SRCEN 位为“0”的状态下给 SRCEN 位写“1”时 [置位条件] • 当给 SRCCTRL 的 FL 位写“1”时
3	UDF	0	R/(W)*	输出 FIFO 下溢中断请求标志 表示在输出 FIFO 的数据量为“0”时读输出数据 FIFO。 [清除条件] • 在 UDF 位为“1”的状态下读 UDF 位后写“0”时 • 当给 SRCCTRL 的 CL 位写“1”时 • 在 SRCCTRL 的 SRCEN 位为“0”的状态下给 SRCEN 位写“1”时 [置位条件] • 在输出 FIFO 的数据量为“0”的状态下读输出数据 FIFO 时

位	位名称	初始值	R/W	说明
2	OVF	0	R/(W)*	<p>输出 FIFO 重写中断请求标志</p> <p>表示在输出 FIFO 满时结束下一个数据的转换处理。在清除 OVF 标志前停止转换处理。</p> <p>[清除条件]</p> <ul style="list-style-type: none"> 在 SRCCTRL 的 OVEN 位为 “1” 并且 OVF 位为 “1” 的状态下读 OVF 位后写 “0” 时 在 SRCCTRL 的 OVEN 位为 “0” 时读 SRCOD，使输出 FIFO 的数据量减少时 当给 SRCCTRL 的 CL 位写 “1” 时 在 SRCCTRL 的 SRCEN 位为 “0” 的状态下给 SRCEN 位写 “1” 时 <p>[置位条件]</p> <ul style="list-style-type: none"> 在输出 FIFO 满的状态下结束下一个数据的转换处理时
1	IINT	1	R/(W)*	<p>输入 FIFO 空中断请求标志</p> <p>表示保存在输入 FIFO 中的数据量小于等于 SRC 输入数据控制寄存器 (SRCIDCTRL) 的 IFTRG[1:0] 位设定的触发个数。</p> <p>[清除条件]</p> <ul style="list-style-type: none"> 在 IINT 位为 “1” 的状态下读 IINT 位后写 “0” 时 通过 DMA 传送，使输入 FIFO 的数据量超过设定的触发个数时 <p>[置位条件]</p> <ul style="list-style-type: none"> 当保存在输入 FIFO 中的数据量小于等于设定的触发个数时 当给 SRCCTRL 的 CL 位写 “1” 时 在 SRCCTRL 的 SRCEN 位为 “0” 的状态下给 SRCEN 位写 “1” 时
0	OINT	0	R/(W)*	<p>输出 FIFO 满中断请求标志</p> <p>此位表示保存在输出 FIFO 中的数据量大于等于 SRC 输出数据控制寄存器 (SRCODCTRL) 的 OFTRG[1:0] 位设定的触发个数。</p> <p>[清除条件]</p> <ul style="list-style-type: none"> 在 OINT 位为 “1” 的状态下读 OINT 位后写 “0” 时 通过 DMA 传送，使输出 FIFO 的数据量小于设定的触发个数时 当给 SRCCTRL 的 CL 位写 “1” 时 在 SRCCTRL 的 SRCEN 位为 “0” 的状态下给 SRCEN 位写 “1” 时 <p>[置位条件]</p> <ul style="list-style-type: none"> 当保存在输出 FIFO 中的数据量大于等于设定的触发个数时

【注】 * 只能在读 “1” 后写 “0”。

42.3 运行说明

42.3.1 初始设定

初始设定的步骤如图 42.2 所示。

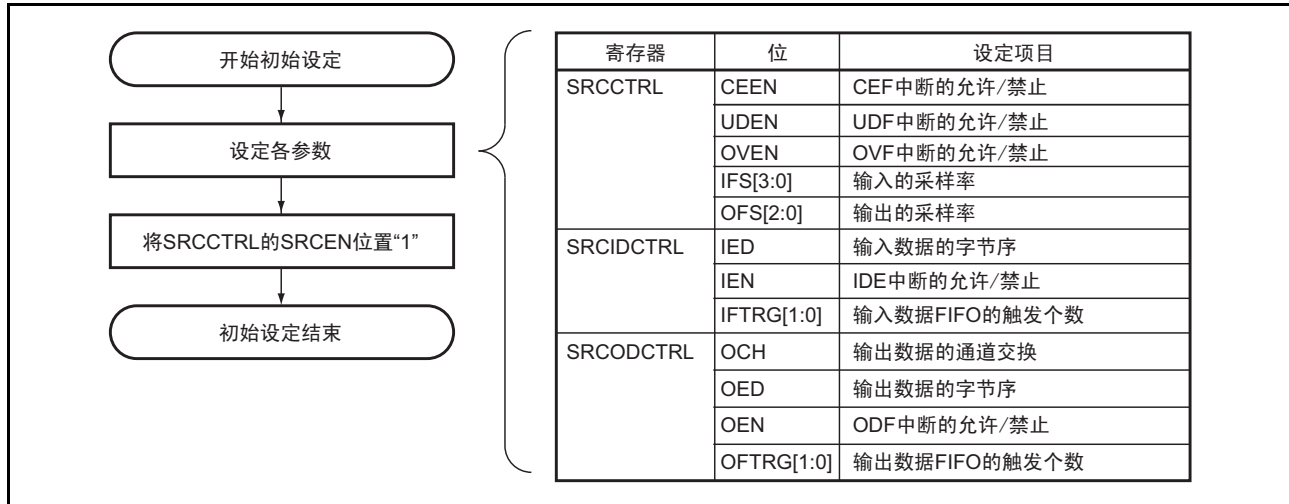


图 42.2 初始设定的步骤例子

42.3.2 数据输入

数据输入的步骤如图 42.3 所示。

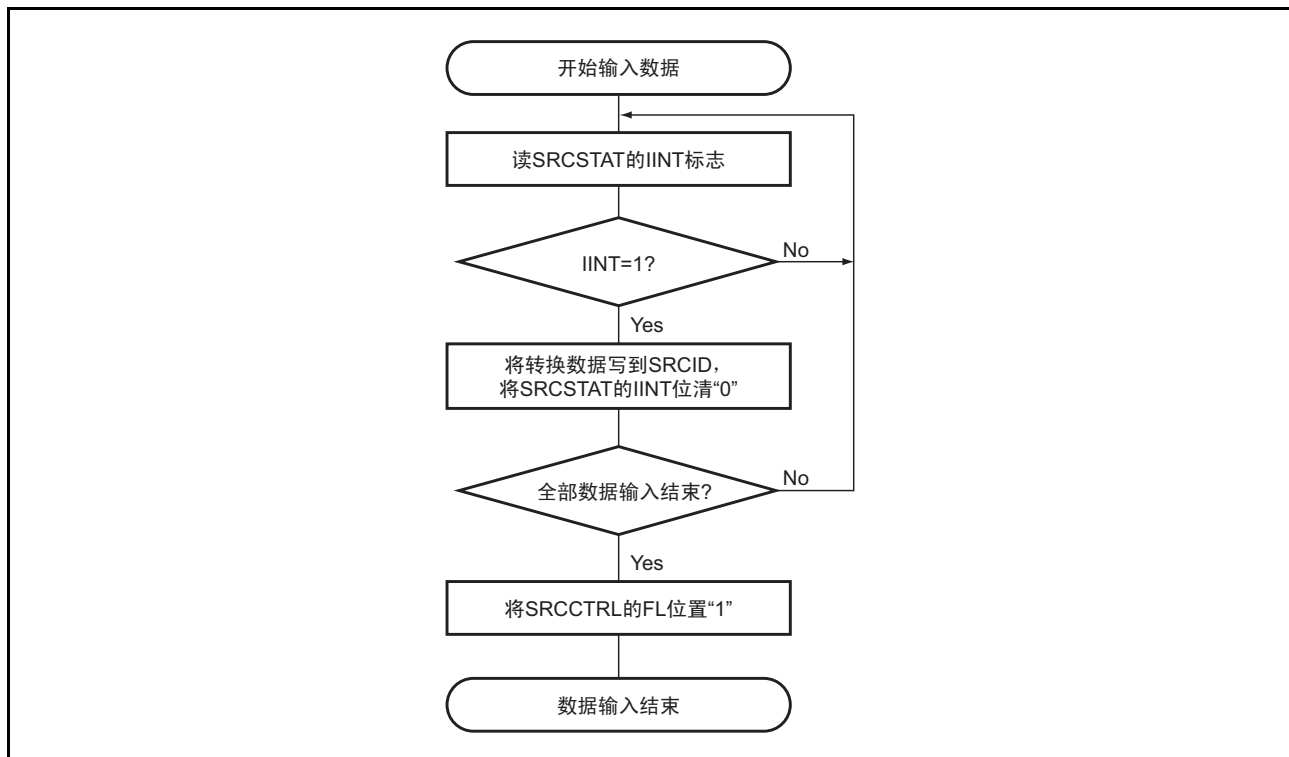


图 42.3 数据输入的步骤例子

(1) 向 CPU 请求中断的情况

1. 将SRCIDCTRL的IEN位置“1”。
2. 如果SRCSTAT的IINT位被置“1”，就产生IDE中断请求。通过中断处理程序，在SRCSTAT的IINT位变为“1”后将数据写到SRCID，并且给SRCSTAT的IINT位写“0”。然后，从中断处理程序返回。
3. 重复执行2.，一旦全部数据的输入结束，就给SRCCTRL的FL位写“1”。

(2) 通过中断启动直接存储器存取控制器的情况

1. 将直接存储器存取控制器的任意一个通道分配到此模块的IDEI。
2. 将SRCIDCTRL的IEN位置“1”。
3. 如果SRCSTAT的IINT位被置“1”，就产生IDE中断请求，启动直接存储器存取控制器。通过DMA传送，将数据写到SRCID。如果输入数据FIFO的数据量大于SRCIDCTRL的IFTRG[1:0]位设定的触发个数，就清除SRCSTAT的IINT位。
4. 重复执行3.，一旦全部数据的输入结束，就给SRCCTRL的FL位写“1”。

(3) 通过串行音频接口的中断启动直接存储器存取控制器并且从串行音频接口传送输入数据的情况

1. 将直接存储器存取控制器任意一个通道的DMA传送请求源分配到串行音频接口，将传送源设定为串行音频接口的SSIFRDR而将传送目标设定为此模块的SRCID，然后设定串行音频接口，使其进入能接受的状态。
2. 如果SSIFSR的RDF位被置“1”，串行音频接口就产生中断请求，启动直接存储器存取控制器。通过DMA传送，将从SSIFRDR读到的数据写到SRCID。
3. 重复执行2.，一旦全部数据的输入结束，就给SRCCTRL的FL位写“1”。

42.3.3 数据输出

数据输出的步骤如图 42.4 所示。

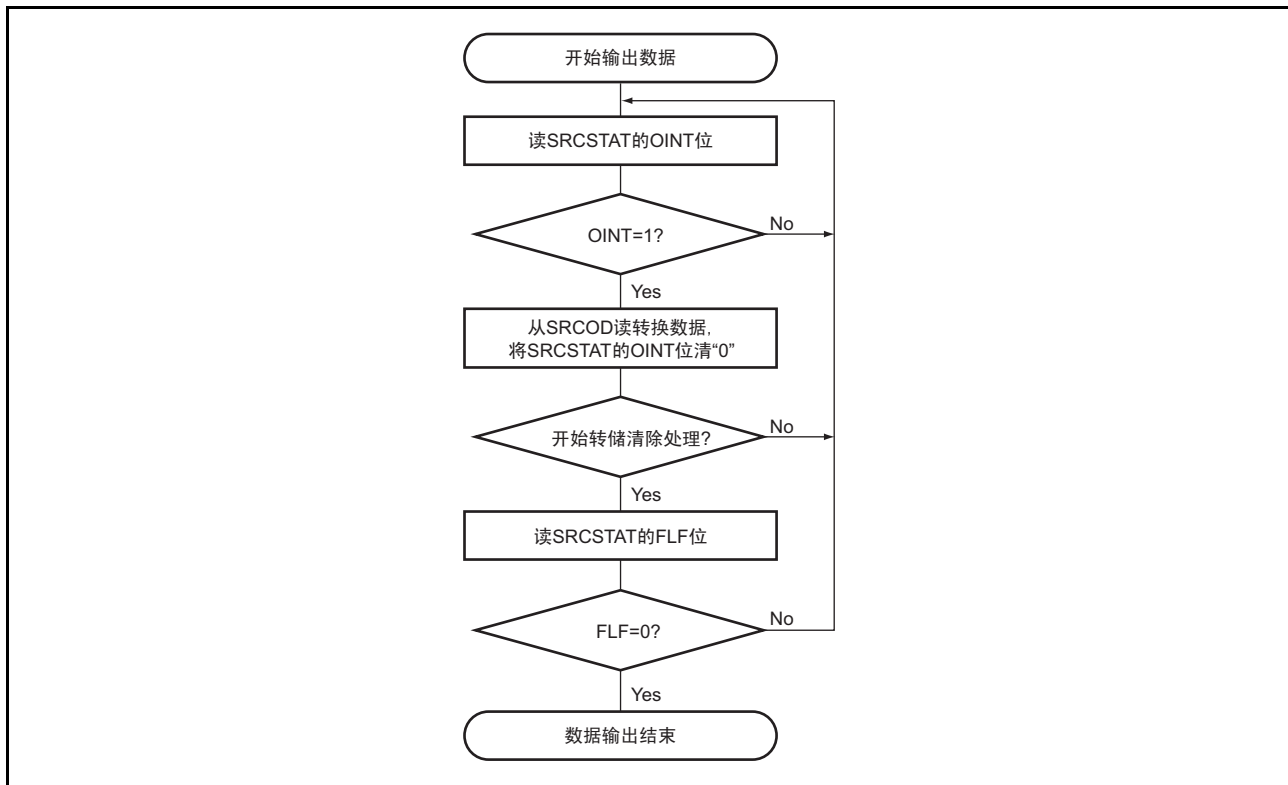


图 42.4 数据输出的步骤例子

(1) 向 CPU 请求中断的情况

1. 将SRCODCTRL的OEN位置“1”。
2. 如果SRCSTAT的OINT位被置“1”，就产生ODF中断请求。通过中断处理程序，在SRCSTAT的OINT位变为“1”后从SRCOD读数据，并且给SRCSTAT的OINT位写“0”。然后，从中断处理程序返回。
3. 在开始转储清除处理后重复执行2.，直到SRCSTAT的CEF位变为“1”为止。

(2) 通过中断启动直接存储器存取控制器的情况

1. 将直接存储器存取控制器的任意一个通道分配到此模块的ODFI。
2. 将SRCODCTRL的OEN位置“1”。
3. 如果SRCSTAT的OINT位被置“1”，就产生ODF中断请求，启动直接存储器存取控制器。通过DMA传送，从SRCOD读数据。如果输出数据FIFO的数据量小于SRCODCTRL的OFTRG[1:0]位设定的触发个数，就清除SRCSTAT的OINT位。
4. 在开始转储清除处理后重复执行3.，直到SRCSTAT的FLF位变为“0”为止。

(3) 通过串行音频接口的中断启动直接存储器存取控制器并且将输出数据传送到串行音频接口的情况

1. 将SRCCTRL的OVEN位置“0”，禁止发生OVF中断请求。
2. 将直接存储器存取控制器任意一个通道的DMA传送请求源分配到串行音频接口，将传送源设定为此模块的SRCOD而将传送目标设定为串行音频接口的SSIFTDR，然后设定串行音频接口，使其进入能发送的状态。
3. 如果SSIFSR的TDE位被置“1”，串行音频接口就产生中断请求，启动直接存储器存取控制器。通过DMA传送，将从SRCOD读到的数据写到SSIFTDR。
4. 在开始转储清除处理后重复执行3.，直到SRCSTAT的CEF位变为“1”为止。

42.4 中断

此模块的中断源有输入 FIFO 数据空 (IDEI)、输出 FIFO 数据满 (ODFI)、输出 FIFO 重写 (OVF)、输出 FIFO 下溢 (UDF) 和转换处理结束 (CEF) 共 5 种。中断的种类和发生条件如表 42.7 所示。

表 42.7 中断请求的种类和发生条件

中断请求	略称	发生条件	直接存储器存取控制器的启动
输入数据 FIFO 空	IDEI	$(IINT=1) \cdot (IEN=1) \cdot (SRCEN=1)$	能
输出数据 FIFO 满	ODFI	$(OINT=1) \cdot (OEN=1) \cdot (SRCEN=1)$	能
输出数据 FIFO 重写	OVF	$(OVF=1) \cdot (OVEN=1) \cdot (SRCEN=1)$	不能
输出数据 FIFO 下溢	UDF	$(UDF=1) \cdot (UDEN=1) \cdot (SRCEN=1)$	不能
转换处理结束	CEF	$(CEF=1) \cdot (CEEN=1) \cdot (SRCEN=1)$	不能

如果中断发生条件成立，CPU 就进行中断异常处理。必须在中断异常处理程序中清除中断源标志。

通过设定直接存储器存取控制器，IDEI 中断和 ODFI 中断能启动直接存储器存取控制器。如果启动直接存储器存取控制器，此模块就不向 CPU 请求中断。通过 DMA 传送，将数据写到 SRCID。如果输入数据 FIFO 的数据量大于设定的触发个数，就清除 IINT。同样，从 SRCOD 读数据，如果输出数据 FIFO 的数据量小于设定的触发个数，就清除 OINT。

42.5 使用时的注意事项

42.5.1 存取寄存器时的注意点

以下所示的从 SRCCTRL 的写操作开始到反映到 SRCSTAT 为止的时间，需要 3 个外围时钟 0 (P0φ) 周期。

- 在给 SRCCTRL 的 FL 位写 “1” 后到 SRCSTAT 的 FLF 位被置位为止。
- 在给 SRCCTRL 的 CL 位写 “1” 后到 SRCSTAT 的各位被初始化为止。
- 在 SRCCTRL 的 SRCEN 位为 “0” 的状态下给 SRCEN 位写 “1” 后到 SRCSTAT 的各位被初始化为止。

另外，因为 CPU 不等待寄存器写操作的结束而执行后续指令，所以写 SRCCTRL 的指令后的第一条指令读不到 SRCSTAT 变更后的状态。要确认 SRCSTAT 变更后的状态时，必须在写 SRCCTRL 的指令之后虚读 SRCCTRL 或者 SRCSTAT。

42.5.2 有关转储清除的注意点

如果给 SRC 控制寄存器 (SRCCTRL) 的 FL 位写 “1”，此模块就在此时为止输入的数据之后边附加零数据边继续进行转换处理。必须在输入完作为音频数据结束的数据并且没有后续数据时进行转储清除处理。

在转储清除处理后要再次进行转换处理时，必须通过以下处理清除内部工作存储器：

- 给 SRCCTRL 的 CL 位写 “1”。
- 在给 SRCCTRL 的 SRCEN 位写 “0” 后写 “1”。

43. 音频发生器

本 LSI 内置 4 个通道的音频发生器。

43.1 特点

- 能通过 8 位 PWM 输出来调整音量。
- 可选择运行时钟。
可选择 4 种运行时钟 ($P0\phi/2 \sim P0\phi/16$)。
- 能以不超过 1% 的输出精度设定 100Hz ~ 20kHz 的频率。
- 可选择输出的停止方法
- 可选择自动衰减功能。
- 中断源：1 种
能请求衰减结束中断。
- 能设定模块停止模式。

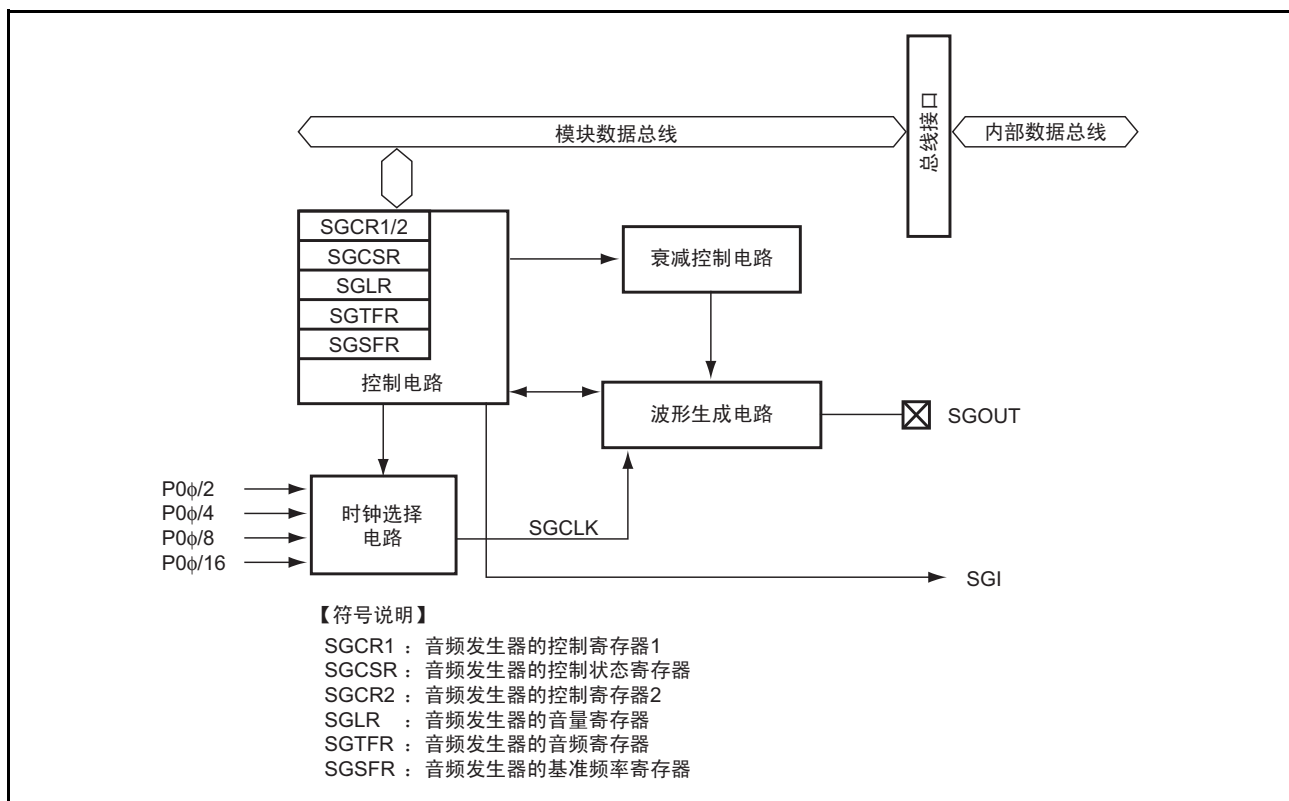


图 43.1 框图

43.2 输入 / 输出引脚

音频发生器的引脚结构如表 43.1 所示。

表 43.1 引脚结构

名称	符号	输入 / 输出	功能
音频发生器的输出引脚 0	SGOUT_0	输出	通道 0 的音频发生器输出
音频发生器的输出引脚 1	SGOUT_1	输出	通道 1 的音频发生器输出
音频发生器的输出引脚 2	SGOUT_2	输出	通道 2 的音频发生器输出
音频发生器的输出引脚 3	SGOUT_3	输出	通道 3 的音频发生器输出

43.3 寄存器说明

寄存器结构如表 43.2 所示，在本章节中省略了各通道的区别。

表 43.2 寄存器结构

通道	寄存器名	略称	R/W	初始值	地址	存取长度
0	音频发生器的控制寄存器 1_0	SGCR1_0	R/W	H'00	H'FFFEC800	8、16
	音频发生器的控制状态寄存器 _0	SGCSR_0	R/W	H'00	H'FFFEC801	8、16
	音频发生器的控制寄存器 2_0	SGCR2_0	R/W	H'00	H'FFFEC802	8、16
	音频发生器的音量寄存器 _0	SGLR_0	R/W	H'00	H'FFFEC803	8、16
	音频发生器的音频寄存器 _0	SGTFR_0	R/W	H'00	H'FFFEC804	8、16
	音频发生器的基准频率寄存器 _0	SGSFR_0	R/W	H'00	H'FFFEC805	8、16
1	音频发生器的控制寄存器 1_1	SGCR1_1	R/W	H'00	H'FFFECA00	8、16
	音频发生器的控制状态寄存器 _1	SGCSR_1	R/W	H'00	H'FFFECA01	8、16
	音频发生器的控制寄存器 2_1	SGCR2_1	R/W	H'00	H'FFFECA02	8、16
	音频发生器的音量寄存器 _1	SGLR_1	R/W	H'00	H'FFFECA03	8、16
	音频发生器的音频寄存器 _1	SGTFR_1	R/W	H'00	H'FFFECA04	8、16
	音频发生器的基准频率寄存器 _1	SGSFR_1	R/W	H'00	H'FFFECA05	8、16
2	音频发生器的控制寄存器 1_2	SGCR1_2	R/W	H'00	H'FFFEC00	8、16
	音频发生器的控制状态寄存器 _2	SGCSR_2	R/W	H'00	H'FFFEC01	8、16
	音频发生器的控制寄存器 2_2	SGCR2_2	R/W	H'00	H'FFFEC02	8、16
	音频发生器的音量寄存器 _2	SGLR_2	R/W	H'00	H'FFFEC03	8、16
	音频发生器的音频寄存器 _2	SGTFR_2	R/W	H'00	H'FFFEC04	8、16
	音频发生器的基准频率寄存器 _2	SGSFR_2	R/W	H'00	H'FFFEC05	8、16
3	音频发生器的控制寄存器 1_3	SGCR1_3	R/W	H'00	H'FFFEC00	8、16
	音频发生器的控制状态寄存器 _3	SGCSR_3	R/W	H'00	H'FFFEC01	8、16
	音频发生器的控制寄存器 2_3	SGCR2_3	R/W	H'00	H'FFFEC02	8、16
	音频发生器的音量寄存器 _3	SGLR_3	R/W	H'00	H'FFFEC03	8、16
	音频发生器的音频寄存器 _3	SGTFR_3	R/W	H'00	H'FFFEC04	8、16
	音频发生器的基准频率寄存器 _3	SGSFR_3	R/W	H'00	H'FFFEC05	8、16

43.3.1 音频发生器的控制寄存器 1 (SGCR1)

SGCR1 控制此模块的运行。

位	7	6	5	4	3	2	1	0
位名	SGST	STPM	—	SGCK[1]	SGCK[0]	DPF[2]	DPF[1]	DPF[0]
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	SGST	0	R/W	运行开始 设定允许或者禁止运行。 0: 停止 1: 允许 但是, 在 SGST 位为 “1” 时, 停止方法也会因 STPM 的状态而不同。
6	STPM	0	R/W	停止方法选择 选择运行的停止方法。 0: 在 SGST 位为 “0” 时停止 1: 当衰减功能 on 时, 在 SGST 位为 “0” 并且 SGDEF 位为 “1” 时停止。 当衰减功能 off 时, 在 SGST 位为 “0” 并且 SGEND 位为 “1” 时停止。
5	—	0	R	保留位 读写值都为 “0”
4 3	SGCK[1:0]	H'0	R/W R/W	时钟选择 选择运行时钟 (SGCLK)。 00: P0φ/2 01: P0φ/4 10: P0φ/8 11: P0φ/16
2 1 0	DPF[2:0]	H'0	R/W R/W R/W	衰减功能选择 选择衰减功能的 on/off 以及衰减周期。 000: 衰减功能 off 001: 以 TONE 频率衰减 010: 以 TONE 频率 /2 衰减 011: 以 TONE 频率 /4 衰减 100: 以 TONE 频率 /8 衰减 101: 以 TONE 频率 /16 衰减 110: 以 TONE 频率 /32 衰减 111: 禁止设定

43.3.2 音频发生器的控制状态寄存器 (SGCSR)

SGCSR 是状态寄存器。

位	7	6	5	4	3	2	1	0
位名	SGIE	SGDEF	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/(W)*	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7	SGIE	0	R/W	中断允许 允许或者禁止衰减结束中断请求。 0: 禁止中断请求 1: 允许中断请求
6	SGDEF	0	R/(W)*	衰减结束标志 [置位条件] • 当衰减结束时 [清除条件] • 在读“1”后写“0”时 • 当写 SGLR 位时
5~0	—	全 0	R	保留位 读写值都为“0”。

【注】 * 为了清除标志，只能写“0”。

43.3.3 音频发生器的控制寄存器 2 (SGCR2)

SGCR2 设定此模块的停止。

位	7	6	5	4	3	2	1	0
位名	SGEND	TCHG	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7	SGEND	0	R/W	停止位 在衰减功能 off 并且 STPM 位为“1”时，控制此模块的运行。 0: 继续运行 1: 停止运行 当 STPM 位为“0”时，与此位的状态无关，通过 SGST 位控制运行。
6	TCHG	0	R/W	TONE 更改保护 设定允许或者禁止写 TONE 位和 SFS 位。 能在 TCHG 位为“1”的状态下写 TONE 位和 SFS 位。 0: 禁止写 TONE 位和 SFS 位 1: 允许写 TONE 位和 SFS 位
5~0	—	全 0	R	保留位 读写值都为“0”。

43.3.4 音频发生器的音量寄存器 (SGLR)

SGLR 设定 SGOUT 的占空比。

位	7	6	5	4	3	2	1	0
位名	LD[7]	LD[6]	LD[5]	LD[4]	LD[3]	LD[2]	LD[1]	LD[0]
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 0	LD[7:0]	H'00	R/W	音量数据 保存输出脉冲的占空比数据。

43.3.5 音频发生器的音频寄存器 (SGTFR)

SGTFR 设定 TONE 频率。

位	7	6	5	4	3	2	1	0
位名	—	TONE[6]	TONE[5]	TONE[4]	TONE[3]	TONE[2]	TONE[1]	TONE[0]
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

【注】 * 只能在TCHG位为“1”时写此位。

位	位名	初始值	R/W	说明
7	—	0	R	保留位 读写值都为“0”。
6 ~ 0	TONE[6:0]	H'00	R/(W)*	TONE 频率设定 基于 SFS 位设定的基准频率，设定 TONE 频率。禁止设定“H'00”。

【注】 * 只能在 TCHG 位为“1”时写此位。

43.3.6 音频发生器的基准频率寄存器 (SGSFR)

SGSFR 设定基准频率。

位	7	6	5	4	3	2	1	0
位名	SFS[7]	SFS[6]	SFS[5]	SFS[4]	SFS[3]	SFS[2]	SFS[1]	SFS[0]
初始值:	0	0	0	0	0	0	0	0
R/W:	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

【注】 * 只能在TCHG位为“1”时写此位。

位	位名	初始值	R/W	说明
7 ~ 0	SFS[7:0]	H'00	R/(W)*	基准频率设定 基于 SGCR1 的 SGCK 位选择的运行时钟 (SGCLK)，设定基准频率。禁止设定“H'00”。

【注】 * 只能在 TCHG 位为“1”时写此位。

43.4 运行说明

43.4.1 基本运行

(1) 初始设定

必须在确认此模块已经停止后设定各寄存器。

通过 SGCR1 的 STPM 位选择停止方法，通过 SGCK 位选择运行时钟，并且通过 DPF2 ~ 0 位选择衰减功能的 on/off。如果衰减功能 on，就选择衰减周期。通过 SGCSR 的 SGIE 位设定中断请求。

(2) 运行开始

将 SGCR1 的 SGST 位置“1”并且将 SGCR2 的 SGEND 位清“0”。在通过将 SGCR2 的 TCHG 位置“1”解除 SGSFR 和 SGTFR 的写保护后，通过 SGSFR 的 SFS7 ~ 0 位设定基准频率，并且通过 SGTFR 的 TONE6 ~ 0 位设定 TONE 频率。通过 SGLR 设定音量。如果写完 SGCR2、SGLR、SGTFR 和 SGSFR 寄存器，此模块就开始运行。

(3) 运行停止

通过 SGCR1 的 STPM 位设定此模块的运行停止方法。

当衰减功能 off 并且 STPM 位为“0”时，与 SGEND 位无关，通过 SGST 位停止运行。

当 STPM 位为“1”时，在 SGST 位为“0”并且 SGEND 位为“0”时停止运行。

当衰减功能 on 并且 STPM 位为“0”时，即使不结束自动衰减（不将 SGDEF 位置“1”），也能通过将 SGST 位清“0”来停止运行。当 STPM 位为“1”时，即使将 SGST 位清“0”也不停止运行，而通过结束自动衰减（将 SGDEF 位置“1”）来停止运行。

停止方法一览表和停止方法分别如表 43.3 和图 43.2 所示。

表 43.3 停止方法一览表

衰减功能 off 时					衰减功能 on 时				
STPM	SGST	SGEND	SGDEF	运行	STPM	SGST	SGEND	SGDEF	运行
0	0	x	x	停止	0	0	x	x	停止
0	1	x	x	输出	0	1	x	x	输出
1	0	0	x	保持*	1	0	x	0	保持*
1	0	1	x	停止	1	0	x	1	停止
1	1	0	x	输出	1	1	x	0	输出
1	1	1	x	输出	1	1	x	1	输出

【符号说明】

x: Don't care

【注】* 保持以前的状态

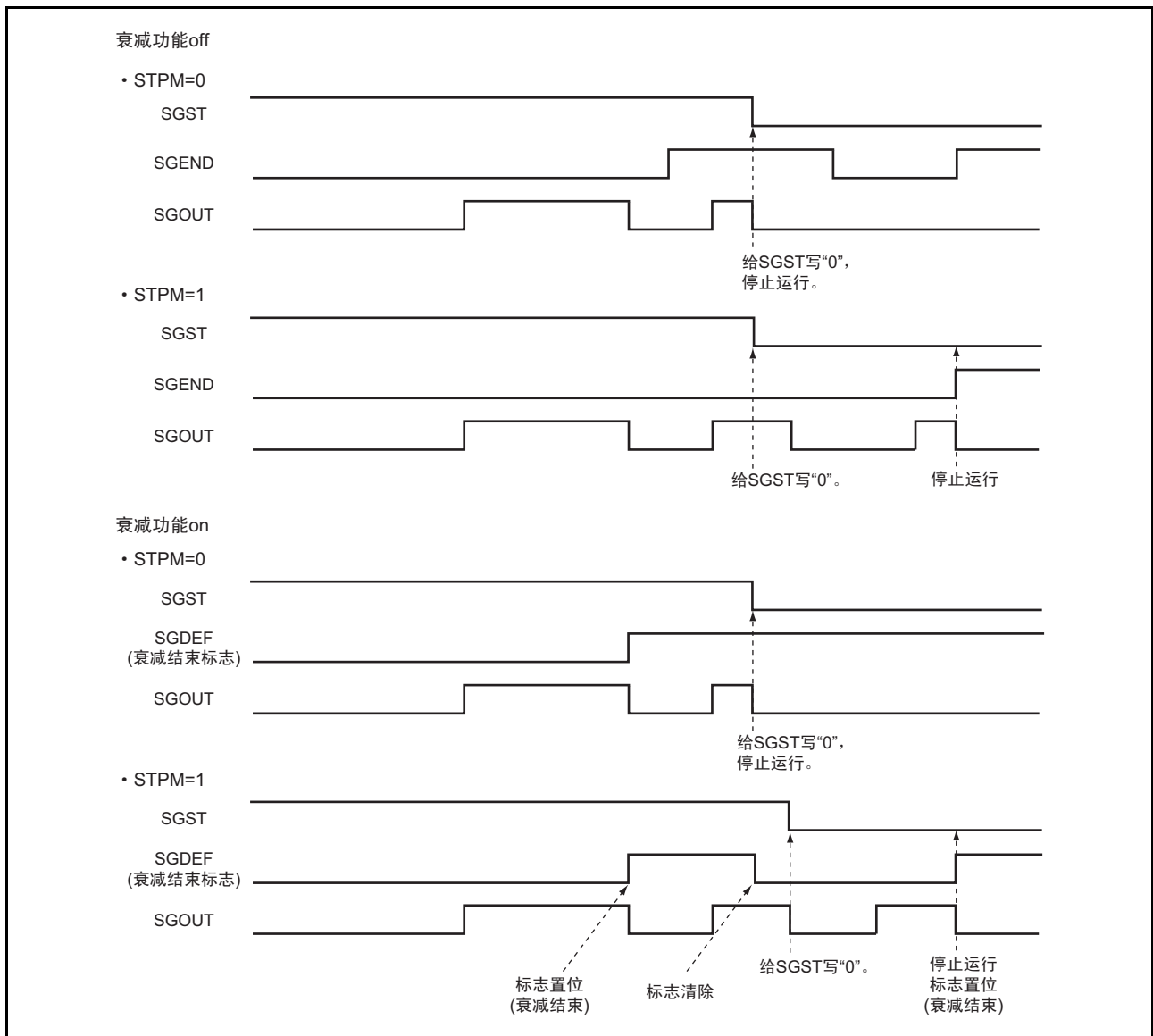


图 43.2 停止方法

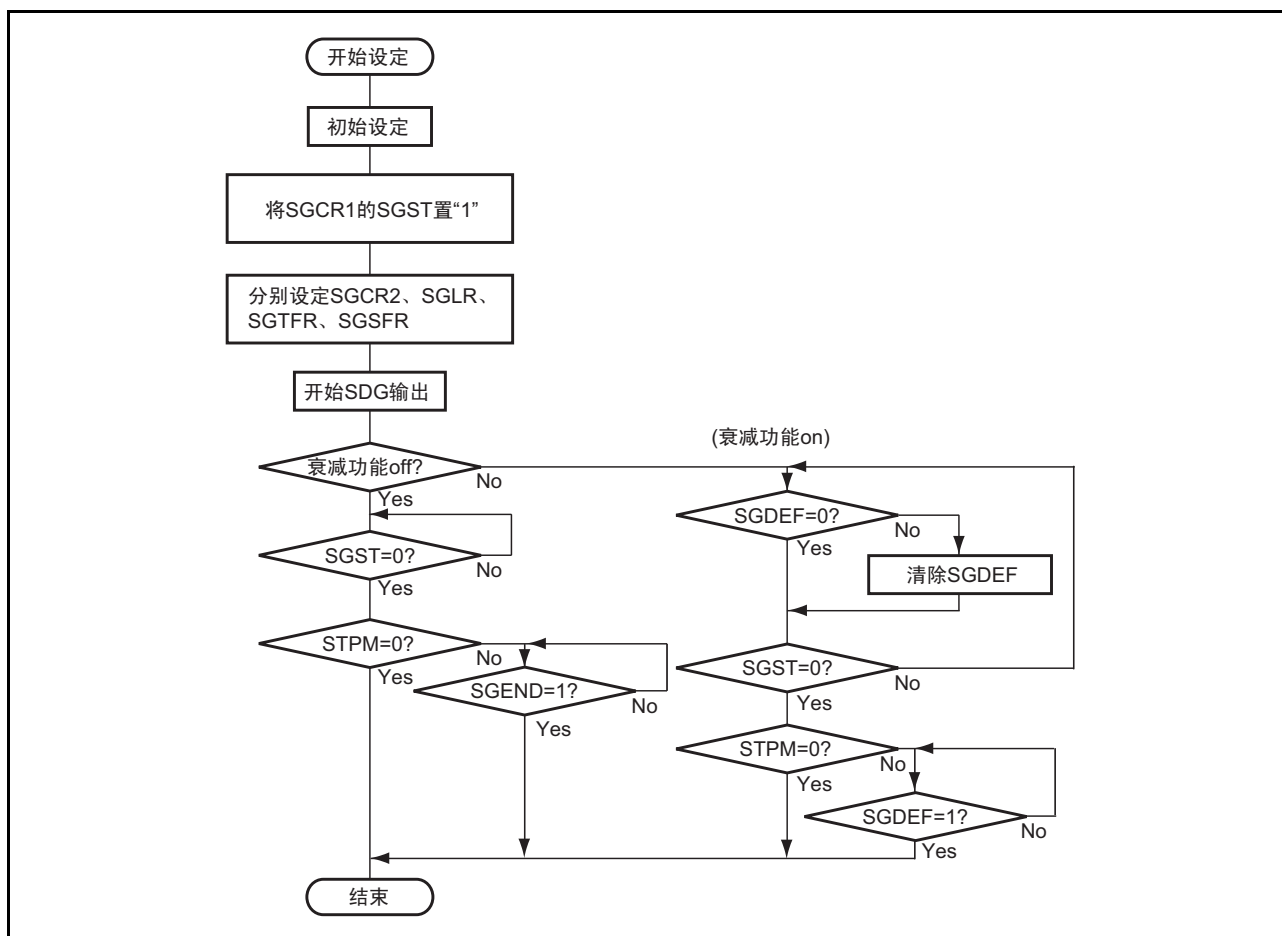


图 43.3 运行流程图

43.4.2 TONE 频率的设定

如表 43.4 所示，此模块以不超过 1% 的精度输出 100 ~ 20kHz 的 TONE 频率。
用以下计算式求 TONE 频率：

$$\begin{aligned} \text{基准频率 [Hz]} &= \text{SGCLK(Hz)} / \text{SFS} \\ \text{TONE 频率 [Hz]} &= \text{基准频率 (Hz)} / (2 \times \text{TONE}) \\ &= \text{SGCLK(Hz)} / (2 \times \text{SFS} \times \text{TONE}) \end{aligned}$$

用以下计算式求 SGTFR 的 TONE 位和 SGSFR 的 SFS 位设定值。

$$\begin{aligned} \text{SFS} &= \text{SGCLK[Hz]} / \text{基准频率 [Hz]} \\ [0 < \text{SFS} \leq 255] \\ \text{TONE} &= \text{基准频率 [Hz]} / (2 \times \text{TONE 频率 [Hz]}) \\ [0 < \text{TONE} \leq 127] \end{aligned}$$

TONE 频率和输出误差的关系例子如表 43.4 所示。

表 43.4 TONE 频率和输出误差的关系

TONE 频率	SFS[7:0]	TONE[6:0]	误差 [%]
220.00	F7	2E	0.01
329.63	ED	20	0.003
440.00	F7	17	0.01
659.26	ED	10	0.003
880.00	8E	14	0.03
1318.50	ED	8	0.04
1760.00	8E	A	0.03
2637.00	ED	4	0.004
3520.00	8E	5	0.03
5274.00	ED	2	0.002
7040.00	47	5	0.03

【注】 SGCLK=5MHz

如果 P0φ 的周期发生变化，TONE 频率也会发生变化，所以在更改时必须注意。

43.4.3 自动衰减功能

在使用自动衰减功能时，音量数据（LD）为 SGOUT 的初始占空比。

在 SGR1 的 DPF 位设定的衰减周期，每次以 1/32 衰减率来减少 SGOUT 的占空比。

用以下计算式求衰减特性：

$$LD_n = \text{int}(LD_0 \times (1 - 1/32)^n)$$

LD: SGOUT 占空比（初始数据为 SGLR）

n: 衰减周期数

衰减特性图如图 43.4 所示。

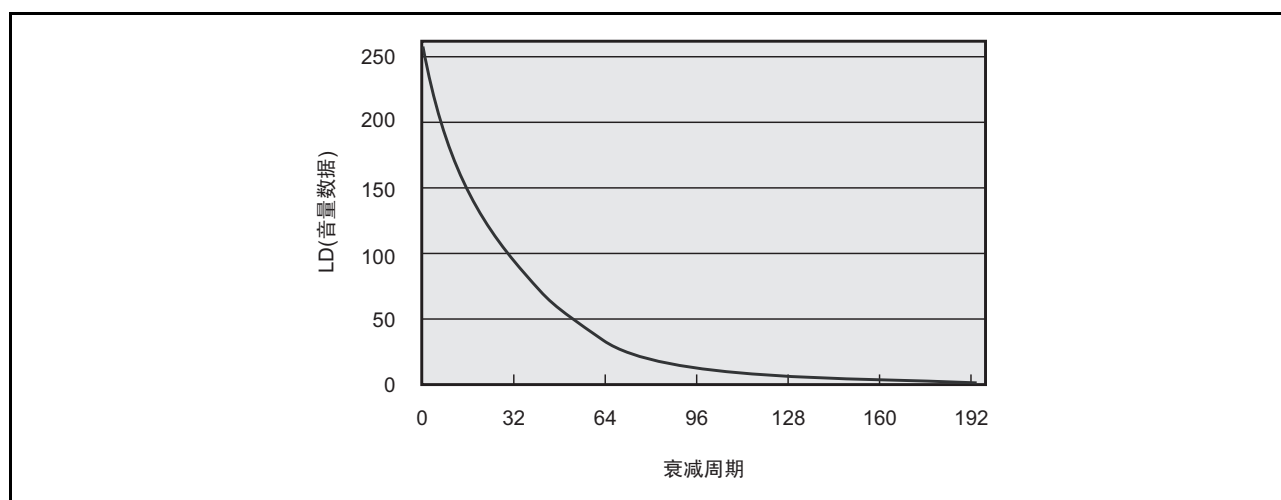


图 43.4 衰减特性

43.4.4 输出波形

如图 43.5 所示，此模块的输出波形是内部 8 位 PWM 的脉冲输出和 TONE 频率的合成输出。通过 SGLR 设定内部 8 位 PWM 的脉冲输出占空比。

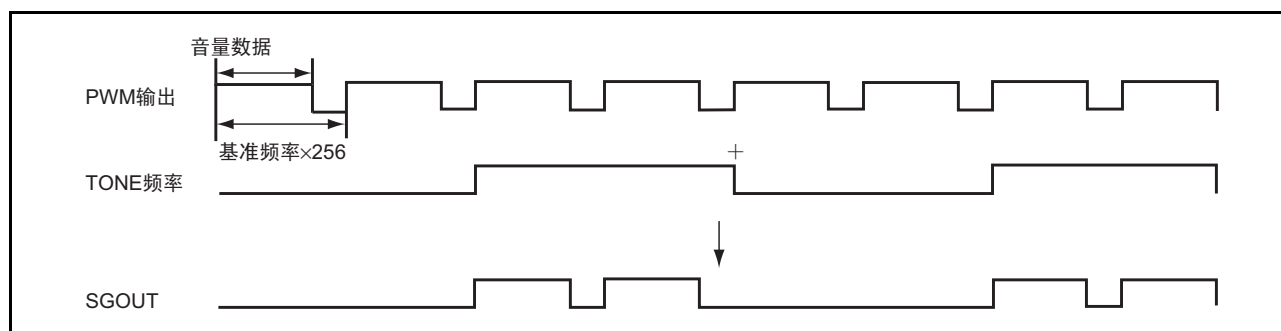


图 43.5 输出波形

43.5 中断源

当衰减功能 on 时，一旦自动衰减结束（内部 PWM 的占空比为“0”），就将 SGCSR 的 SGDEF 位置位。此时，如果 SGCSR 的 SGIE 位已被置“1”，就请求中断。当 STPM 位为“0”时，只在第 1 次衰减结束时将 SGDEF 位置位。如果在自动衰减过程中清除 SGDEF 位，就在下次衰减结束时将 SGDEF 位置位。

表 43.5 中断源

名称	中断源	中断标志
SGDEI	衰减结束	SGDEF

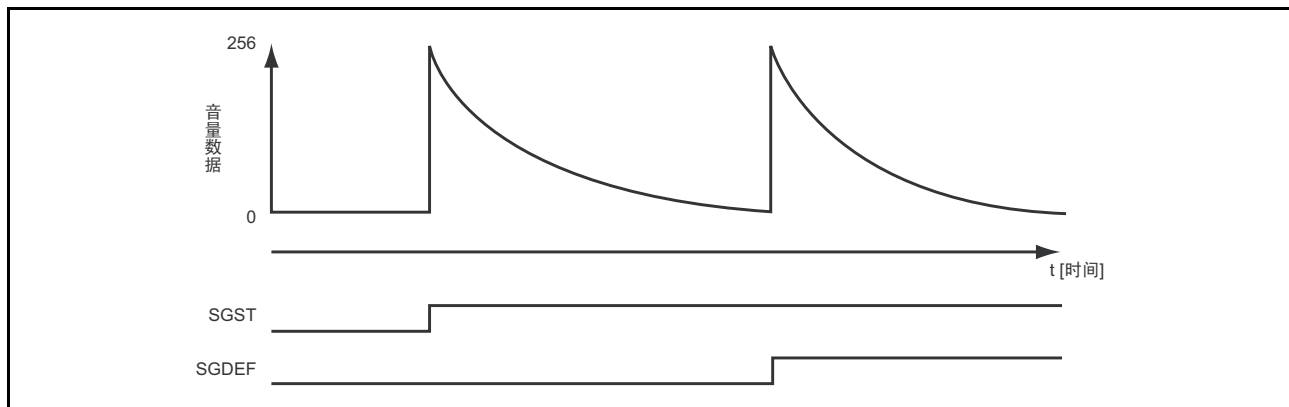


图 43.6 衰减结束标志的置位时序

43.6 使用时的注意事项

43.6.1 模块停止模式的设定

能通过模块停止控制寄存器设定允许或者停止此模块的运行，初始值为停止此模块的运行。通过解除模块停止模式，使寄存器变为可存取的状态，详细内容请参照“49. 低功耗模式”。

44. SD 主机接口

本章在缔结保密合同后予以公开。
详细内容请向本公司的销售员确认。

特点请参照“1. 概要”。
电特性请参照“52. 电特性”。

45. MMC 主机接口

MMC 主机接口是对应 JEDEC STANDARD JESD84-A44 的主机控制器，能连接有 MMC 接口的各种设备。

45.1 特点

- 对应 1/4/8 位的 MMC bus。
- 只对应 Single Data Rate。
- MMC 时钟频率 = P1φ 频率 / 2ⁿ (n=1……10)
- 数据缓冲器：512 字节 × 2
- 中断请求：3 个（正常运行、错误/超时、卡检测）
- DMA 传送请求：写缓冲器的 DMA 传送请求和读缓冲器的 DMA 传送请求
- 卡检测功能

框图如图 45.1 所示。

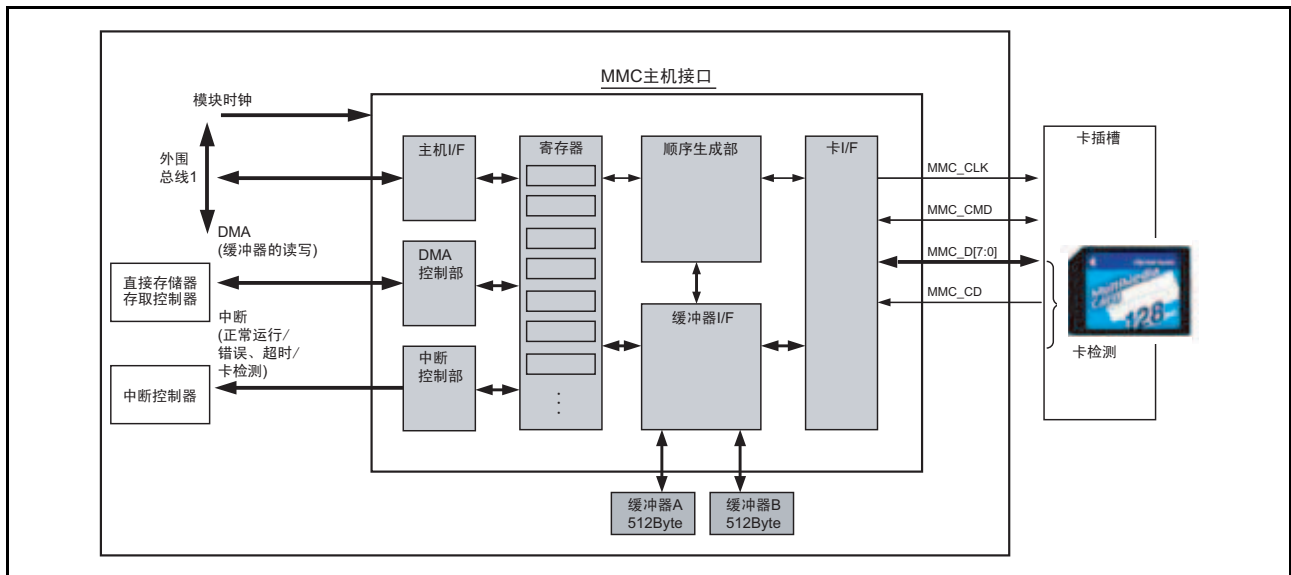


图 45.1 MMC 主机接口的框图

45.2 输入 / 输出引脚

此模块的引脚结构如表 45.1 所示。

表 45.1 引脚结构

引脚名	输入 / 输出	功能
MMC_CLK	输出	MMC 时钟
MMC_CMD	输入 / 输出	命令 / 应答
MMC_D[7:0]	输入 / 输出	发送数据 / 接收数据
MMC_CD	输入	卡检测 *

【注】 * 必须在考虑要使用的卡插槽规格后进行连接。

45.3 寄存器说明

此模块的寄存器结构如表 45.2 所示。

表 45.2 寄存器结构

寄存器名称	略称	R/W	地址	存取长度
命令设定寄存器	CE_CMD_SET	R/W	H'E8030800	16/32
自变量寄存器	CE_ARG	R/W	H'E8030808	16/32
自动 CMD12 的自变量寄存器	CE_ARG_CMD12	R/W	H'E803080C	16/32
命令控制寄存器	CE_CMD_CTRL	R/W	H'E8030810	16/32
传送块设定寄存器	CE_BLOCK_SET	R/W	H'E8030814	16/32
时钟控制寄存器	CE_CLK_CTRL	R/W	H'E8030818	16/32
缓冲器存取设定寄存器	CE_BUF_ACC	R/W	H'E803081C	16/32
应答寄存器 3	CE_RESP3	R	H'E8030820	16/32
应答寄存器 2	CE_RESP2	R	H'E8030824	16/32
应答寄存器 1	CE_RESP1	R	H'E8030828	16/32
应答寄存器 0	CE_RESP0	R	H'E803082C	16/32
自动 CMD12 的应答寄存器	CE_RESP_CMD12	R	H'E8030830	16/32
数据寄存器	CE_DATA	R/W	H'E8030834	16*/32
中断标志寄存器	CE_INT	R/W	H'E8030840	16/32
中断允许寄存器	CE_INT_EN	R/W	H'E8030844	16/32
状态寄存器 1	CE_HOST_STS1	R/W	H'E8030848	16/32
状态寄存器 2	CE_HOST_STS2	R/W	H'E803084C	16/32
DMA 模式设定寄存器	CE_DMA_MODE	R/W	H'E803085C	16/32
卡检测 / 端口控制寄存器	CE_DETECT	R/W	H'E8030870	16/32
特殊模式设定寄存器	CE_ADD_MODE	R/W	H'E8030874	16/32
版本寄存器	CE_VERSION	R/W	H'E803087C	16/32

【注】 不能存取上述以外的寄存器。

* 在进行 16 位存取时，只能存取 H'E8030834。

45.3.1 命令设定寄存器 (CE_CMD_SET)

CE_CMD_SET 是设定命令顺序的寄存器。

如果设定 bit31 ~ 16, 就开始执行命令顺序。不能在命令顺序执行过程中 (CE_HOST_STS1 的 CMDSEQ 位为 “1” 时) 写 CE_CMD_SET。必须根据 “45.7.12 CE_CMD_SET 的设定值” 设定 CE_CMD_SET。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CMD[5:0]					RTYP[1:0]	RBSY	—	WDAT	DWEN	CMLTE	CMD12 EN		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RIDXC[1:0]	RCRC7C[1:0]	—	CRC 16C	—	CRC STE	TBIT	OPDM	—	—	SBIT	—	DATW[1:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R	R/W	R/W	R/W	R	R	R/W	R	R/W	R/W

位	位名	初始值	R/W	说明
31、30	—	全 0	R	保留位 读写值都为 “0”。
29 ~ 24	CMD[5:0]	H'00	R/W	命令索引 设定命令索引 ([45:40])。 【注】 如果设定命令索引, 就开始执行命令顺序。
23、22	RTYP[1:0]	00	R/W	应答类型 00: 不应答 01: 6 字节应答 (R1、R1b、R3、R4、R5) 10: 17 字节应答 (R2) 11: 禁止设定
21	RBSY	0	R/W	应答忙 / 不忙 选择在接收应答时应答是否忙。 0: 应答不忙 1: 应答忙 (R1b)
20	—	0	R	保留位 读写值都为 “0”。
19	WDAT	0	R/W	数据的有无 0: 无数据 1: 有数据
18	DWEN	0	R/W	读写 (在有数据时有效) 0: 读卡 1: 写卡
17	CMLTE	0	R/W	单块传送 / 多块传送选择 (在有数据时有效) 0: 单块传送 1: 多块传送
16	CMD12EN	0	R/W	CMD12 自动发行设定 (在多块传送时有效) 0: 不自动发行 CMD12 1: 自动发行 CMD12 (= 自动 CMD12) 有关 CMD12 的自动发行, 请参照 “45.6.4 CMD12 的自动发行”。 【注】 必须将传送块大小设定为 512 字节, 并且将 RBSY 位置 “0”。

位	位名	初始值	R/W	说明
15、14	RIDXC[1:0]	00	R/W	应答索引检查 设定 6 字节应答的 [45:40] 位或者 17 字节应答的 [133:128] 位的检查内容。 00: 检查索引 (确认是否和命令索引相同) 01: 检查 check bits (确认这些位是否都为“1”) 10: 未检查 11: 禁止设定
13、12	RCRC7C [1:0]	00	R/W	应答 CRC7 检查 设定 6 字节应答或者 17 字节应答的 [7:1] 位的检查内容。 00: 检查 CRC7 (必须将应答类型设定为“01”) 01: 检查 check bits (必须将应答类型设定为“01”) 10: 检查 internal CRC7 (R2 专用) (必须将应答类型设定为“10”) 11: 未检查
11	—	0	R	保留位 读写值都为“0”。
10	CRC16C	0	R/W	接收时的 CRC16 检查 (在有数据并且读操作时有效) 0: 检查 CRC16 1: 不检查 CRC16 (用于 CMD14 时)
9	—	0	R	保留位 读写值都为“0”。
8	CRCSTE	0	R/W	CRC status 接收 (在有数据并且写操作时有效) 0: 接收 CRC status 1: 不接收 CRC status (用于 CMD19 时)
7	TBIT	0	R/W	发送位设定 0: 将发送位 ([46]) 置“1” 1: 将发送位 ([46]) 置“0”
6	OPDM	0	R/W	漏极开路输出模式设定 0: 正常输出 1: 漏极开路输出 【注】 只对 MMC_CMD 线有效。
5、4	—	全 0	R	保留位 读写值都为“0”。
3	SBIT	0	R/W	读数据的起始位检测设定 (在有数据并且读操作时有效) 0: 在通过设定 DATW 将有效的 MMC_D 全部置“0”时检测起始位 1: 在 MMC_D[0] 位为“0”时检测起始位
2	—	0	R	保留位 读写值都为“0”。
1、0	DATW[1:0]	00	R/W	数据总线宽度设定 (在有数据时有效) 00: 1 位 01: 4 位 10: 8 位 11: 禁止设定

45.3.2 自变量寄存器 (CE_ARG)

CE_ARG 是设定发送命令自变量的寄存器。必须在开始执行命令顺序前设定 CE_ARG。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ARG[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ARG[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 0	ARG[31:0]	H'0000 0000	R/W	设定命令的 [39:8] 位。 【注】 必须通过 CE_ARG_CMD12 设定自动 CMD12 的自变量。

45.3.3 自动 CMD12 的自变量寄存器 (CE_ARG_CMD12)

CE_ARG_CMD12 是设定自动 CMD12 自变量的寄存器。在多块传送时自动发行 CMD12 的情况下，此寄存器有效。

有关自动 CMD12，请参照“45.6.4 CMD12 的自动发行”。必须在开始执行命令顺序前设定 CE_ARG_CMD12。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	C12ARG[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	C12ARG[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 0	C12ARG [31:0]	H'0000 0000	R/W	设定自动 CMD12 的 [39:8] 位。

45.3.4 命令控制寄存器 (CE_CMD_CTRL)

CE_CMD_CTRL 是强制结束命令顺序的寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BREAK
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	BREAK	0	R/W	命令顺序强制结束 要中止执行命令顺序时，必须在 BREAK 位为“0”的状态下写“1”后写“0”。此后，必须在确认 CE_HOST_STS1 的 CMDSEQ 位为“0”后进行软件复位。 【注】 一旦进行软件复位，寄存器的值将返回初始值，因此需要重新设定寄存器。

45.3.5 传送块设定寄存器 (CE_BLOCK_SET)

CE_BLOCK_SET 是设定传送数据块大小和块数的寄存器。必须在开始执行命令顺序前设定 CE_BLOCK_SET。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BLKCNT[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BLKSIZ[15:0]															
初始值:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 16	BLKCNT [15:0]	H'0000	R/W	传送块数 【注】 在多块传送时，这些位有效。
15 ~ 0	BLKSIZ [15:0]	H'0200	R/W	传送块大小 【注】 必须将传送块大小设定为： • 在单块传送时 1 ~ 512 字节。 • 在多块传送时 512 字节。

45.3.6 时钟控制寄存器 (CE_CLK_CTRL)

CE_CLK_CTRL 是控制 MMC 时钟以及设定超时值的寄存器。不能在命令顺序执行过程中重新进行设定。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	CLKEN	—	—	—	—	CLKDIV[3:0]			
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	SRSPTO[1:0]		SRBSYTO[3:0]			SRWDTO[3:0]			—	—	—	—		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 25	—	全 0	R	保留位 读写值都为“0”。
24	CLKEN	0	R/W	MMC 时钟输出控制 0: 不输出 MMC 时钟 (固定为“L”电平) 1: 输出 MMC 时钟
23 ~ 20	—	全 0	R	保留位 读写值都为“0”。
19 ~ 16	CLKDIV[3:0]	0000	R/W	MMC 时钟频率设定 0000: $P1\phi/2^1$ 0001: $P1\phi/2^2$: 0111: $P1\phi/2^8$ 1000: $P1\phi/2^9$ 1001: $P1\phi/2^{10}$ 1010 ~ 1111: 禁止设定
15、14	—	全 0	R	保留位 读写值都为“0”。
13、12	SRSPTO [1:0]	00	R/W	应答超时设定 设定 CE_INT 的 RSPTO 期间。 00: 64×MMC 时钟周期 01: 128×MMC 时钟周期 10: 256×MMC 时钟周期 11: 禁止设定
11 ~ 8	SRBSYTO [3:0]	0000	R/W	应答忙超时设定 设定 CE_INT 的 RBSYTO 期间。 0000: 2^{14} ×MMC 时钟周期 0001: 2^{15} ×MMC 时钟周期 : 1110: 2^{28} ×MMC 时钟周期 1111: 2^{29} ×MMC 时钟周期

位	位名	初始值	R/W	说明
7 ~ 4	SRWDTO [3:0]	0000	R/W	写数据超时 / 读数据超时设定 设定 CE_INT 的 WDATTO 期间和 RDATTO 期间。 0000: 2 ¹⁴ ×MMC 时钟周期 0001: 2 ¹⁵ ×MMC 时钟周期 : 1110: 2 ²⁸ ×MMC 时钟周期 1111: 2 ²⁹ ×MMC 时钟周期
3 ~ 0	—	0000	R	保留位 读写值都为“0”。

45.3.7 缓冲器存取设定寄存器 (CE_BUF_ACC)

CE_BUF_ACC 设定数据寄存器的存取方法以及 DMA 的传送方法。

不能在命令顺序执行过程中重新进行设定。有关缓冲器说明，请参照“45.6.3 缓冲器结构和缓冲器的存取”。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	DMAW EN	DMAR EN	—	—	—	—	—	—	BUSW	ATYP
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 26	—	全 0	R	保留位 读写值都为“0”。
25	DMAWEN	0	R/W	写缓冲器的 DMA 传送请求允许 0: 禁止写缓冲器的 DMA 传送请求 1: 允许写缓冲器的 DMA 传送请求
24	DMAREN	0	R/W	读缓冲器的 DMA 传送请求允许 0: 禁止读缓冲器的 DMA 传送请求 1: 允许读缓冲器的 DMA 传送请求
23 ~ 18	—	全 0	R	保留位 读写值都为“0”。
17	BUSW	0	R/W	数据寄存器的存取宽度选择 0: 以 32 位存取 CE_DATA 1: 以 16 位存取 CE_DATA
16	ATYP	0	R/W	缓冲器的存取选择 0: 不以字节为单位进行交换 1: 以字节为单位进行交换 【注】 有关缓冲器的存取，请参照“45.6.3 缓冲器结构和缓冲器的存取”。
15 ~ 0	—	全 0	R	保留位 读写值都为“0”。

45.3.8 应答寄存器 3 ~ 0 (CE_RESP3 ~ 0)

CE_RESP3 ~ 0 是保存接收应答值的寄存器。

有关应答值的格式，请参照“45.6.1 命令 / 应答的格式”。

• CE_RESP3

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RSP[127:112]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSP[111:96]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 0	RSP[127:96]	H'0000 0000	R	保存 17 字节应答的 [127:96] 位。

• CE_RESP2

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RSP[95:80]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSP[79:64]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 0	RSP[95:64]	H'0000 0000	R	保存 17 字节应答的 [95:64] 位。

• CE_RESP1

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RSP[63:48]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSP[47:32]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 0	RSP[63:32]	H'0000 0000	R	保存 17 字节应答的 [63:32] 位。

- CE_RESP0

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RSP[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSP[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 0	RSP[31:0]	H'0000 0000	R	保存 6 字节应答的 [39:8] 位或者 17 字节应答的 [31:0] 位。 【注】 将对自动 CMD12 的应答保存到 CE_RESP_CMD12。

45.3.9 自动 CMD12 的应答寄存器 (CE_RESP_CMD12)

在自动发行 CMD12 时，CE_RESP_CMD12 保存对 CMD12 的应答值。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RSP12[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSP12[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 0	RSP12[31:0]	H'0000 0000	R	保存对自动 CMD12 的应答 [39:8] 位。

45.3.10 数据寄存器 (CE_DATA)

CE_DATA 是存取缓冲器的寄存器。

在进行 16 位存取时，只能存取 DATA[31:16] 位。有关写数据和读数据的格式，请参照“45.6.2 数据块的格式”。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA[31:16]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 0	DATA[31:0]	H'0000 0000	R/W	缓冲器的写 / 读数据 [31:0] 位

45.3.11 中断标志寄存器 (CE_INT)

CE_INT 是表示命令顺序执行过程中的各种状态的寄存器。如果满足置位条件，就将各位置“1”。在清除标志时，只能将要清除的位置“0”而必须将其他位置“1”。

有关发生错误或者超时时的运行，请参照“45.6.5 发生错误或者超时时的此模块的处理”。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	CMD12 DRE	CMD12 RBE	CMD12 CRE	DTRAN E	BUFR E	BUFW EN	BUFR EN	—	—	RBSY E	CRSP E
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMD VIO	BUF VIO	—	—	WDAT ERR	RDAT ERR	RIDX ERR	RSP ERR	—	—	—	CRCS TO	WDAT TO	RDAT TO	RBSY TO	RSP TO
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26	CMD12DRE	0	R/W*	自动 CMD12 & 缓冲器读结束 置位条件: 自动 CMD12 应答忙并且读完缓冲器时 清除条件: 写“0” 【注】 因为在 CMD12DRE 被置位时也将 CMD12RBE、CMD12CRE 和 BUFRE 位置位，所以必须清除这些位。
25	CMD12RBE	0	R/W*	自动 CMD12 应答忙结束 置位条件: 自动 CMD12 的应答接收和应答忙结束时 清除条件: 写“0”。 【注】 因为在 CMD12RBE 位被置位时也将 CMD12CRE 位置位，所以必须清除 CMD12CRE 位。在进行多块写的情况下 CMD12RBE 位被置位时也将 DTRANE 位置位，所以必须清除 DTRANE 位。
24	CMD12CRE	0	R/W*	自动 CMD12 命令应答结束 置位条件: 接收到自动 CMD12 的应答时 清除条件: 写“0”
23	DTRANE	0	R/W*	数据发送结束 置位条件: 发送全部块的全部数据时 • 设定为接收 CRC status 的情况 CRC status 后的忙 (数据忙) 结束时 • 设定为不接收 CRC status 的情况 发送完数据时 清除条件: 写“0”
22	BUFRE	0	R/W*	缓冲器读结束 置位条件: 接收全部块的全部数据并且已经从缓冲器读完数据时 清除条件: 写“0”
21	BUFWEN	0	R/W*	缓冲器可写 置位条件: 在缓冲器为空状态并且处于能写的状态时 清除条件: 写“0” 【注】 在允许写缓冲器的 DMA 传送请求时，不将此位置位。

位	位名	初始值	R/W	说明
20	BUFREN	0	R/W*	缓冲器可读 置位条件：将传送块大小的数据保存到缓冲器并且处于能读的状态时 清除条件：写“0” 【注】 在允许读缓冲器的 DMA 传送请求时，不将此位置位。
19、18	—	全 0	R	保留位 读写值都为“0”。
17	RBSYE	0	R/W*	应答忙结束 置位条件：接收完应答和应答忙时 清除条件：写“0” 【注】 因为在 RBSYE 位被置位时也将 CRSPE 位置位，所以必须清除 CRSPE 位。将自动 CMD12 的应答和应答忙反映到 CMD12RBE 位。
16	CRSPE	0	R/W*	命令应答结束 置位条件：发送完命令或者接收完应答时 • 设定为不应答的情况 发送完命令时 • 设定为 6 字节应答或者 17 字节应答的情况 接收完应答时 清除条件：写“0” 【注】 将自动 CMD12 的应答反映到 CMD12CRE。
15	CMDVIO	0	R/W*	命令发行错误 置位条件：CE_CMD_SET、CE_BLOCK_SET 的设定非法时 • 正在执行命令顺序的情况： 写 CE_CMD_SET 的 CMD[5:0] 位时 （不自动停止执行命令顺序） • 开始执行命令顺序时的情况： 在寄存器的设定为以下任意组合的状态下写 CE_CMD_SET 的 CMD[5:0] 位时 - 不应答+应答忙 - 不应答+有数据 - 无数据+自动发行 CMD12 - 有数据+单块+自动发行 CMD12 - 有数据+应答忙+自动发行 CMD12 - 有数据+传送块大小=0 - 有数据+传送块大小≥513 - 有数据+多块传送+传送块数=0 清除条件：写“0”
14	BUFVIO	0	R/W*	缓冲器存取错误 置位条件：非法存取缓冲器时 • 存取多于 CE_BLOCK_SET 的 BLKSIZ[15:0] 位设定块大小的 CE_DATA 时 • 正在读卡的数据的情况 在 BUFREN 位未被置位（在使用 DMA 时没有发出读缓冲器的 DMA 传送请求）的情况下存取 CE_DATA 时 • 正在给卡写数据的情况： 在 BUFVIO 位未被置位（在使用 DMA 时没有发出写缓冲器的 DMA 传送请求）的情况下存取 CE_DATA 时 清除条件：写“0” 【注】 如果 BUFVIO 位被置位，就不自动停止执行命令顺序。

位	位名	初始值	R/W	说明
13、12	—	全 0	R	保留位 读写值都为“0”。
11	WDATERR	0	R/W*	写数据错误 置位条件： <ul style="list-style-type: none"> • 当写数据有错误时 • 当 CRC status 的状态有错误时 • 当 CRC status 的结束位有错误时 清除条件：写“0” 【注】 如果 WDATERR 位被置位，就自动停止执行命令顺序。
10	RDATERR	0	R/W*	读数据错误 置位条件：当读数据有错误时 <ul style="list-style-type: none"> • 当读数据的 CRC16 有错误时 • 当读数据的结束位有错误时 清除条件：写“0” 【注】 如果 RDATERR 位被置位，就自动停止执行命令顺序。
9	RIDXERR	0	R/W*	应答索引错误 置位条件：当应答的索引值有错误时 <ul style="list-style-type: none"> • 当 6 字节应答（包含自动 CMD12）的 [45:40] 位或者 17 字节应答的 [133:128] 位有错误时 （将检查内容设定到 CE_CMD_SET 的 RIDXC 位） 清除条件：写“0” 【注】 如果 RIDXERR 位被置位，就自动停止执行命令顺序。。
8	RSPERR	0	R/W*	应答错误 置位条件：当应答的应答值有错误时 <ul style="list-style-type: none"> • 当应答的发送位为“H”电平时 • 当应答的结束位有错误时 • 当 6 字节应答（包含自动 CMD12）或者 17 字节应答的 [7:1] 位有错误时 （将检查内容设定到 CE_CMD_SET 的 RCRC7C 位） 清除条件：写“0” 【注】 如果 RSPERR 位被置位，就自动停止执行命令顺序。
7~5	—	全 0	R	保留位 读写值都为“0”。
4	CRCSTO	0	R/W*	CRC status 超时 置位条件：当不能接收 CRC status 时 清除条件：写“0” 【注】 即使 CRCSTO 位被置位，也不停止执行命令顺序。
3	WDATTO	0	R/W*	写数据超时 置位条件：在接收 CRC status 后经过 CE_CLK_CTRL 的 SRWDTO 期间还持续忙状态时 清除条件：写“0” 【注】 即使 WDATTO 位被置位，也不停止执行命令顺序。

位	位名	初始值	R/W	说明
2	RDATTO	0	R/W*	读数据超时 置位条件： - 在发送读命令后经过CE_CLK_CTRL的SRWDTO期间还接收不到读数据时 - 在接收读数据后经过CE_CLK_CTRL的SRWDTO期间还接收不到读数据时 清除条件：写“0” 【注】 即使 RDATTO 位被置位，也不停止执行命令顺序。
1	RBSYTO	0	R/W*	应答忙超时 置位条件：在发送命令（包含自动 CMD12）后经过 CE_CLK_CTRL 的 SRBSYTO 期间还持续忙状态时 清除条件：写“0” 【注】 即使 RBSYTO 位被置位，也不停止执行命令顺序。
0	RSPTO	0	R/W*	应答超时 置位条件：在发送命令（包含自动 CMD12）后经过 CE_CLK_CTRL 的 SRSPTO 期间还接收不到应答时 清除条件：写“0” 【注】 即使 RSPTO 位被置位，也不停止行命令顺序。

【注】 * 只能写“0”，即使写“1”也不反映到此位。

45.3.12 中断允许寄存器 (CE_INT_EN)

CE_INT_EN 是控制 CE_INT 中断输出的寄存器。如果被置“1”的位对应的 CE_INT 标志为“1”，就输出中断。有关中断请求的详细内容，请参照“45.4 中断请求的说明”。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	MCMD 12DRE	MCMD 12RBE	MCMD 12CRE	MDT RANE	MBUF RE	MBUF WEN	MBUF REN	—	—	MRBSY E	MCRSP E
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MCMD VIO	MBUF VIO	—	—	MWDAT ERR	MRDAT ERR	MRIDX ERR	MRSP ERR	—	—	—	MCRC STO	MWDA TTO	MRDA TTO	MRBS YTO	MRSP TO
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
31 ~ 27	—	全 0	R	保留位 读写值都为“0”。
26	MCMD12DRE	0	R/W	CMD12DRE 中断允许 0: 禁止由 CMD12DRE 标志引起的中断输出 1: 允许由 CMD12DRE 标志引起的中断输出
25	MCMD12RBE	0	R/W	CMD12RBE 中断允许 0: 禁止由 CMD12RBE 标志引起的中断输出 1: 允许由 CMD12RBE 标志引起的中断输出
24	MCMD12CRE	0	R/W	CMD12CRE 中断允许 0: 禁止由 CMD12CRE 标志引起的中断输出 1: 允许由 CMD12CRE 标志引起的中断输出
23	MDTRANE	0	R/W	DTRANE 中断允许 0: 禁止由 DTRANE 标志引起的中断输出 1: 允许由 DTRANE 标志引起的中断输出
22	MBUFRE	0	R/W	BUFRE 中断允许 0: 禁止由 BUFRE 标志引起的中断输出 1: 允许由 BUFRE 标志引起的中断输出
21	MBUFWEN	0	R/W	BUFWEN 中断允许 0: 禁止由 BUFWEN 标志引起的中断输出 1: 允许由 BUFWEN 标志引起的中断输出
20	MBUFREN	0	R/W	BUFREN 中断允许 0: 禁止由 BUFREN 标志引起的中断输出 1: 允许由 BUFREN 标志引起的中断输出
19、18	—	全 0	R	保留位 读写值都为“0”。
17	MRBSYE	0	R/W	RBSYE 中断允许 0: 禁止由 RBSYE 标志引起的中断输出 1: 允许由 RBSYE 标志引起的中断输出
16	MCRSPE	0	R/W	CRSPE 中断允许 0: 禁止由 CRSPE 标志引起的中断输出 1: 允许由 CRSPE 标志引起的中断输出

位	位名	初始值	R/W	说明
15	MCMDVIO	0	R/W	CMDVIO 中断允许 0: 禁止由 CMDVIO 标志引起的中断输出 1: 允许由 CMDVIO 标志引起的中断输出
14	MBUFVIO	0	R/W	BUFVIO 中断允许 0: 禁止由 BUFVIO 标志引起的中断输出 1: 允许由 BUFVIO 标志引起的中断输出
13、12	—	全 0	R	保留位 读写值都为“0”。
11	MWDATERR	0	R/W	WDATERR 中断允许 0: 禁止由 WDATERR 标志引起的中断输出 1: 允许由 WDATERR 标志引起的中断输出
10	MRDATERR	0	R/W	RDATERR 中断允许 0: 禁止由 RDATERR 标志引起的中断输出 1: 允许由 RDATERR 标志引起的中断输出
9	MRIDXERR	0	R/W	RIDXERR 中断允许 0: 禁止由 RIDXERR 标志引起的中断输出 1: 允许由 RIDXERR 标志引起的中断输出
8	MRSPELL	0	R/W	RSPERR 中断允许 0: 禁止由 RSPERR 标志引起的中断输出 1: 允许由 RSPERR 标志引起的中断输出
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	MCRCSTO	0	R/W	CRCSTO 中断允许 0: 禁止由 CRCSTO 标志引起的中断输出 1: 允许由 CRCSTO 标志引起的中断输出
3	MWDATTO	0	R/W	WDATTO 中断允许 0: 禁止由 WDATTO 标志引起的中断输出 1: 允许由 WDATTO 标志引起的中断输出
2	MRDATTO	0	R/W	RDATTO 中断允许 0: 禁止由 RDATTO 标志引起的中断输出 1: 允许由 RDATTO 标志引起的中断输出
1	MRBSYTO	0	R/W	RBSYTO 中断允许 0: 禁止由 RBSYTO 标志引起的中断输出 1: 允许由 RBSYTO 标志引起的中断输出
0	MRSPTO	0	R/W	RSPTO 中断允许 0: 禁止由 RSPTO 标志引起的中断输出 1: 允许由 RSPTO 标志引起的中断输出

45.3.13 状态寄存器 1 (CE_HOST_STS1)

CE_HOST_STS1 表示传送完的块数、MMC_CMD 线和 MMC_D 线的状态、接收的应答索引以及命令顺序状态。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMD SEQ	CMD SIG	RSPIDX[5:0]						DATSIG[7:0]							
初始值:	0	—	0	0	0	0	0	0	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCVBLK[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31	CMDSEQ	0	R	命令顺序状态表示 0: 命令顺序为初始状态 1: 正在执行命令顺序
30	CMDSIG	不定值	R	MMC_CMD 状态表示 表示 MMC_CMD 线的状态。
29 ~ 24	RSPIDX[5:0]	H'00	R	应答索引表示 表示 6 字节应答的 [45:40] 位或者 17 字节应答的 [133:128] 位。
23 ~ 16	DATSIG [7:0]	不定值	R	MMC_D 状态表示 表示 MMC_D [7:0] 线的状态。 【注】 如果发生通信错误或者超时, 就可能保持 MMCDAT[0] 位为 “0” 的状态。
15 ~ 0	RCVBLK [15:0]	H'0000	R	传送结束块数表示 表示传送结束的块数。 当 CE_CMD_SET 的 DWEN 位为 “0” 时, 为从卡读到的块数。 当 CE_CMD_SET 的 DWEN 位为 “1” 时, 为写到卡的块数。

45.3.14 状态寄存器 2 (CE_HOST_STS2)

CE_HOST_STS2 是表示各种超时和各种错误状态寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRC STE	CRC 16E	AC12 CRCE	RSP CRC7E	CRC STEBE	RDATEBE	AC12REBE	RSP EBE	AC12 IDXE	RSP IDXE	—	—	—	CRCST[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	STRD ATTO	DATBS YTO	CRCST TO	AC12 BSYTO	RSPBS YTO	AC12 RSPPTO	STRS PTO	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31	CRCSTE	0	R	CRC status 错误 如果 CRC status 值有错误, 此位就变为“1”。
30	CRC16E	0	R	读数据 CRC16 错误 如果读数据的 CRC16 有错误, 此位就变为“1”。
29	AC12CRCE	0	R	自动 CMD12 应答 CRC7 错误 如果自动 CMD12 的应答 [7:1] 位有错误, 此位就变为“1”。 【注】 将检查内容设定到 CE_CMD_SET 的 RCRC7C 位。
28	RSPCRC7E	0	R	命令应答 CRC7 错误 (自动 CMD12 除外) 如果 6 字节应答或者 17 字节应答的 [7:1] 位有错误, 此位就变为“1”。 【注】 将检查内容设定到 CE_CMD_SET 的 RCRC7C 位。
27	CRCSTEBE	0	R	CRC status 结束位错误 如果 CRC status 的结束位有错误, 此位变为“1”。
26	RDATEBE	0	R	读数据的结束位错误 如果读数据的结束位有错误, 此位变为“1”。
25	AC12REBE	0	R	自动 CMD12 应答结束位错误 如果自动 CMD12 应答的结束位有错误, 此位变为“1”。
24	RSPEBE	0	R	命令应答结束位错误 (自动 CMD12 除外) 如果应答的结束位有错误, 此位变为“1”。
23	AC12IDXE	0	R	自动 CMD12 应答索引错误 如果自动 CMD12 的应答 [45:40] 位有错误, 此位变为“1”。 【注】 将检查内容设定到 CE_CMD_SET 的 RIDXC 位。
22	RSPIDXE	0	R	命令应答索引错误 (自动 CMD12 除外) 如果 6 字节应答的 [45:40] 位或者 17 字节应答的 [133:128] 位有错误, 此位变为“1”。 【注】 将检查内容设定到 CE_CMD_SET 的 RIDXC 位。
21 ~ 19	—	全 0	R	保留位 读写值都为“0”。
18 ~ 16	CRCST[2:0]	000	R	CRC status 表示 表示接收的 CRC status 状态值。
15	—	0	R	保留位 读写值都为“0”。

位	位名	初始值	R/W	说明
14	STRDATTO	0	R	读数据超时 <ul style="list-style-type: none"> 在发送读命令后，如果在经过 CE_CLK_CTRL 的 SRWDTO 期间后还接收不到读数据，此位就变为“1”。 在接收读数据后，如果在经过 CE_CLK_CTRL 的 SRWDTO 期间后还接收不到读数据，此位就变为“1”。
13	DATBSYTO	0	R	数据忙超时 在接收 CRC status 后，如果在经过 CE_CLK_CTRL 的 SRWDTO 期间后还持续忙状态，此位就变为“1”。
12	CRCSTTO	0	R	CRC status 超时 如果接收不到 CRC status，此位就变为“1”。
11	AC12BSYTO	0	R	自动 CMD12 应答忙超时 在发送自动 CMD12 后，如果在经过 CE_CLK_CTRL 的 SRBSYTO 期间后还持续忙状态，此位就变为“1”。
10	RSPBSYTO	0	R	应答忙超时 在发送命令（自动 CMD12 除外）后，如果在经过 CE_CLK_CTRL 的 SRBSYTO 期间后还持续忙状态，此位就变为“1”。
9	AC12RSPTO	0	R	自动 CMD12 应答超时 在发送自动 CMD12 后，如果在经过 CE_CLK_CTRL 的 SRSPTO 期间后还接收不到应答，此位就变为“1”。
8	STRSPTO	0	R	应答超时 在发送命令（自动 CMD12 除外）后，如果在经过 CE_CLK_CTRL 的 SRSPTO 期间后还接收不到应答，此位就变为“1”。
7 ~ 0	—	全 0	R	保留位 读写值都为“0”。

45.3.15 DMA 模式设定寄存器 (CE_DMA_MODE)

CE_DMA_MODE 是在要进行 DMA 传送时设定传送单位的寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DMA SEL
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
31 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	DMASEL	0	R/W	DMA 传送长度选择 选择进行读写 CE_DATA 的 DMA 传送时的传送单位。传送大小必须符合 DMA 通道控制寄存器的传送大小 (TS[1:0])。 0: 以字 (2 字节) 或者长字 (4 字节) 为单位 1: 以 16 字节 (长字 ×4) 为单位 【注】 如果在以 16 字节为单位的传送中因通信错误或者超时等而强制结束 DMA 传送, 就必须在给此位写“0”后重新进行设定。在使用软件复位时, 也必须在给此位写“0”后重新进行设定。

45.3.16 卡检测 / 端口控制寄存器 (CE_DETECT)

CE_DETECT 是控制卡检测的寄存器。有关卡检测引起的中断请求的详细内容，请参照“45.4 中断请求的说明”。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	CD SIG	CD RISE	CD FALL	—	—	—	—	—	—	MCD RISE	MCD FALL	—	—	—	—
初始值:	0	—	0	0	0	—	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W*	R/W*	R	R	R	R	R	R	R/W	R/W	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 15	—	全 0	R	保留位 读写值都为“0”。
14	CDSIG	不定值	R	MMC_CD 引脚状态表示 表示 MMC_CD 引脚的状态。
13	CDRISE	0	R/W*	MMC_CD 引脚的 rise 检测标志 置位条件: 当 MMC_CD 引脚从 Low 电平变为 High 电平时 清除条件: 写“0”
12	CDFALL	0	R/W*	MMC_CD 引脚的 fall 检测标志 置位条件: 当 MMC_CD 引脚从 High 电平变为 Low 电平时 清除条件: 写“0”
11	—	0	R	保留位 读写值都为“0”。
10	—	不定值	R	保留位 写入值总是“0”。
9 ~ 6	—	全 0	R	保留位 读写值都为“0”。
5	MCDRISE	0	R/W	CDRISE 中断允许 0: 禁止由 CDRISE 标志引起的中断输出 1: 允许由 CDRISE 标志引起的中断输出
4	MCDFALL	0	R/W	CDFALL 中断允许 0: 禁止由 CDFALL 标志引起的中断输出 1: 允许由 CDFALL 标志引起的中断输出
3 ~ 0	—	全 0	R	保留位 读写值都为“0”。

【注】* 只能写“0”，即使写“1”也不反映到此位。

45.3.17 特殊模式设定寄存器 (CE_ADD_MODE)

CE_ADD_MODE 是控制内部时钟的寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CLK MAIN	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31 ~ 20	—	全 0	R	保留位 读写值都为“0”。
19	CLKMAIN	0	R/W	内部时钟控制 0: 正常模式 1: 低功耗模式 (只限于卡检测)
18 ~ 0	—	全 0	R	保留位 读写值都为“0”。

45.3.18 版本寄存器 (CE_VERSION)

CE_VERSION 是表示此模块的版本值以及控制软件复位的寄存器。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SW RST	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VERSION[15:0]															
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
31	SWRST	0	R/W	软件复位 0: 解除软件复位 (正常运行) 1: 正在进行软件复位 如果将 SWRST 位置“1”，全部寄存器的值就都变为初始值 (SWRST 位不变为初始值)。
30 ~ 16	—	全 0	R	保留位 读写值都为“0”。
15 ~ 0	VERSION [15:0]	H'0003	R	版本表示 表示此模块的版本值。

45.4 中断请求的说明

此模块的中断请求规格如表 45.3 所示。此模块有正常运行中断、错误 / 超时中断、卡检测中断共 3 种中断请求。在中断标志位为“1”并且中断允许位为“1”时，将中断请求置为有效。

表 45.3 中断请求的规格

标志寄存器	位	屏蔽寄存器	位	中断请求
CE_INT	CMD12DRE	CE_INT_EN	MCMD12DRE	正常运行中断 (MMC2)
	CMD12RBE		MCMD12RBE	
	CMD12CRE		MCMD12CRE	
	DTRANE		MDTRANE	
	BUFRE		MBUFRE	
	BUFWEN		MBUFWEN	
	BUFREN		MBUFREN	
	RBSYE		MRBSYE	
	CRSPE		MCRSPE	
	CMDVIO		MCMDVIO	错误 / 超时中断 (MMC1)
	BUFVIO		MBUFVIO	
	WDATERR		MWDATERR	
	RDATERR		MRDATERR	
	RIDXERR		MRIDXERR	
	RSPERR		MRSPELL	
	CRCSTO		MCRCSO	
	WDATTO		MWDATTO	
	RDATTO		MRDATTO	
	RBSYTO		MRBSYTO	
RSPTO	MRSPTO			
CE_DETECT	CDRISE	CE_DETECT	MCDRISE	卡检测中断 (MMC0)
	CDFALL		MCDFALL	

45.5 DMA 规格

此模块有用于读缓冲器和写缓冲器的 2 个通道的 DMA 传送请求。

45.5.1 写缓冲器的 DMA 说明

在将 CE_BUF_ACC 的 DMAWEN 位置“1”并且缓冲器为空时，将写缓冲器的 DMA 传送请求置为有效。

如果将 BLKSIZ (CE_BLOCK_SET 设定的块大小) × BLKCNT (CE_BLOCK_SET 设定的传送块数) 的 DMA 传送请求置为有效并且结束最后块的传送，就将 DMA 传送请求置为无效。此时，必须注意：CE_INT 的 BUFWEN 位无效。

如果在 DMA 传送过程中发生错误或者强制结束 DMA 传送，就自动停止执行命令顺序，因此将 DMA 传送请求置为无效。

45.5.2 读缓冲器的 DMA 说明

在将 CE_BUF_ACC 的 DMAREN 位置“1”并且缓冲器保存了 CE_BLOCK_SET 设定传送块大小的数据时，将读缓冲器的 DMA 传送请求置为有效。

如果将 BLKSIZ (CE_BLOCK_SET 设定的块大小) × BLKCNT (CE_BLOCK_SET 设定的传送块数) 的 DMA 传送请求置为有效并且结束最后块的传送，就将 DMA 传送请求置为无效。此时，必须注意：CE_INT 的 BUFREN 无效。

如果在 DMA 传送过程中发生错误或者强制结束 DMA 传送，就自动停止执行命令顺序，因此将 DMA 传送请求置为无效。

45.6 运行说明

45.6.1 命令 / 应答的格式

发送的命令格式如图45.2所示。在此命令中反映CE_CMD_SET的CMD[5:0]位设定的命令索引以及CE_ARG的 ARG[31:0] 位设定的自变量。

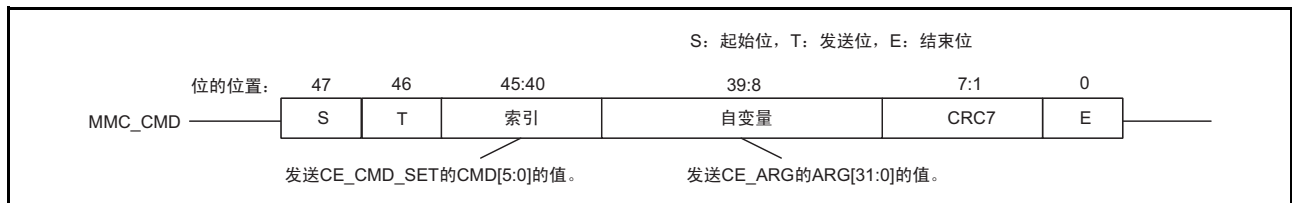


图 45.2 命令格式

接收 6 字节应答和 17 字节应答 (R2) 时的格式分别如图 45.3 和图 45.4 所示。将应答索引保存到 CE_HOST_STS1 的 RSPIDX[5:0] 位，并且将应答的状态值保存到 CE_RESP0 或者 CE_RESP3 ~ 0。

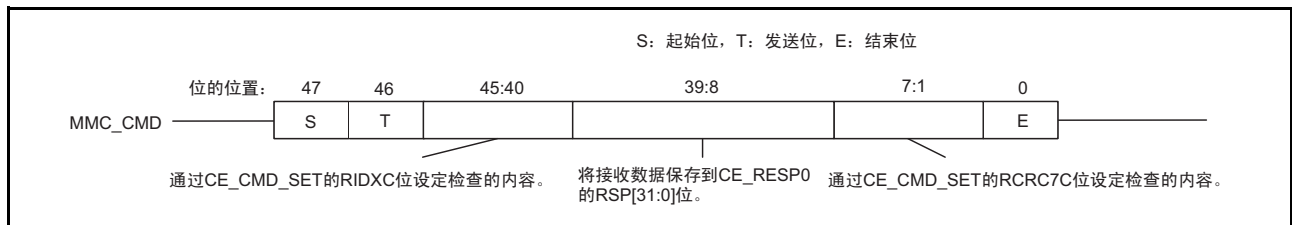


图 45.3 6 字节应答的格式

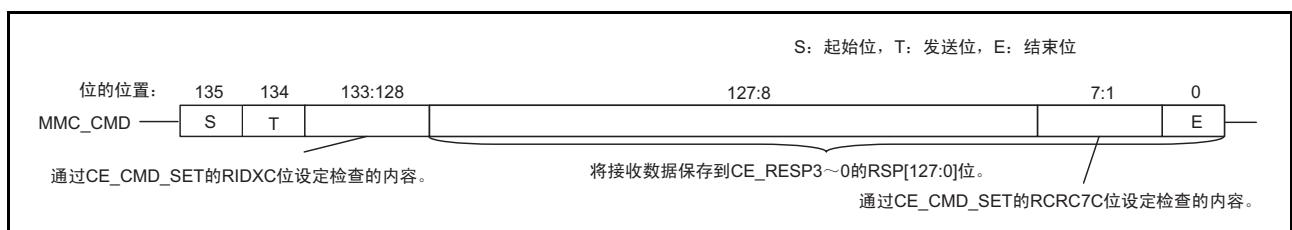


图 45.4 17 字节应答的格式 (R2)

45.6.2 数据块的格式

数据块的格式如图 45.5 所示。有关图中的 D0 ~ D3，请参照“45.6.3 缓冲器结构和缓冲器的存取”。在写卡时，发送保存在缓冲器的数据；在读卡的数据时，将接收的数据保存到缓冲器。

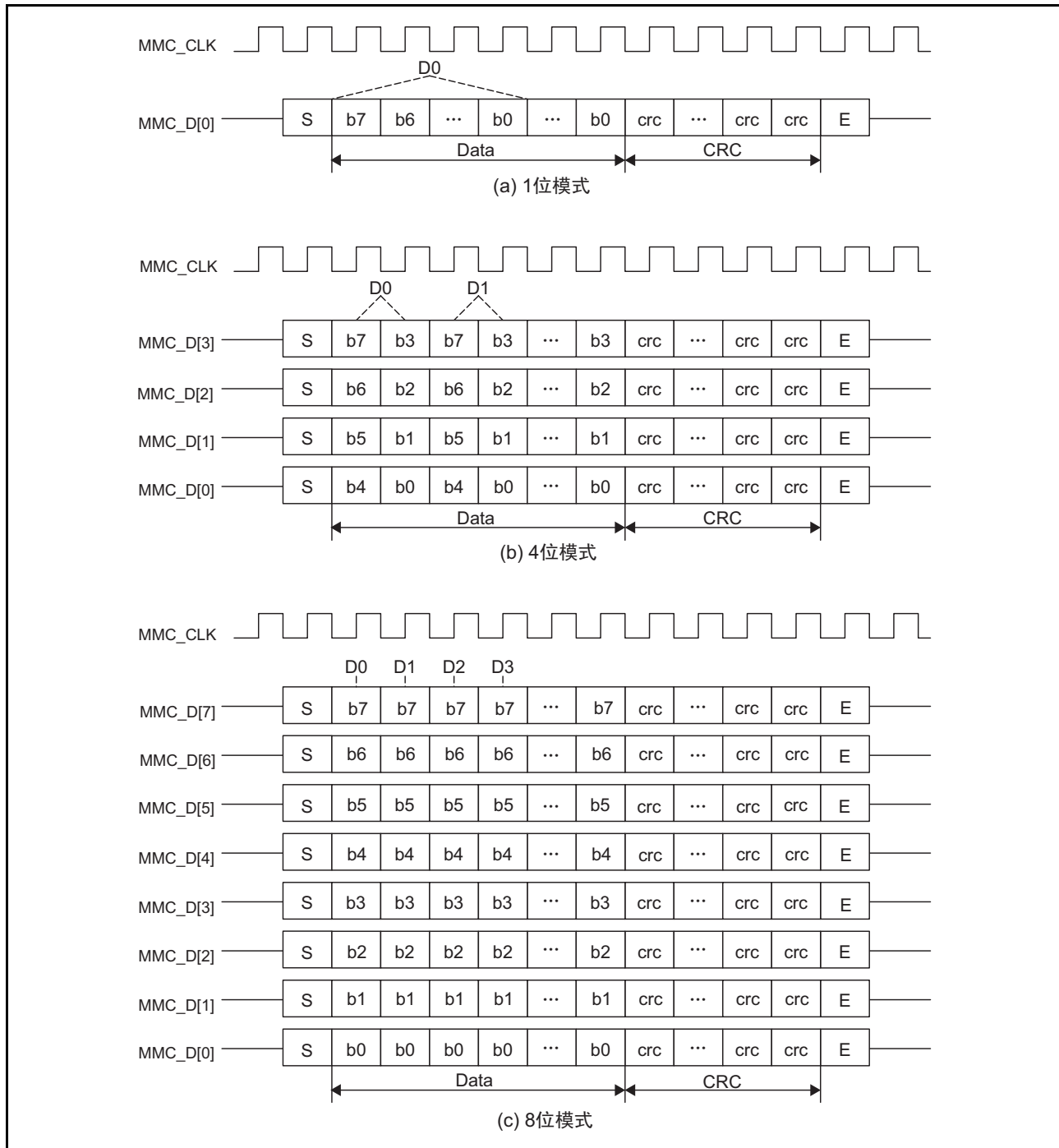


图 45.5 数据块的格式

45.6.3 缓冲器结构和缓冲器的存取

如图 45.6 所示，此模块有 2 个 512 字节的 RAM。因此，在进行多块写时，即使发送保存在缓冲器的 1 块数据 (=512 字节)，如果另一个缓冲器已满，也能继续发送下一个块的数据。在进行多块读时，即使将 1 块的数据 (=512 字节) 保存到缓冲器，如果另一个缓冲器为空，也能继续将下一个块的数据保存到缓冲器。

如果在进行多块读时两个缓冲器都不为空，就停止 MMC 时钟，暂停接收。然后，如果其中一个缓冲器变空，就开始提供 MMC 时钟，重新开始接收。

使用 CE_DATA 存取缓冲器。如果将传送块大小设定为 $4 \times n + 1$ 或者 $4 \times n + 3$ ，就必须在进行 16 位存取时存取 $4 \times n + 2$ 字节或者 $4 \times (n + 1)$ 字节的 CE_DATA，在进行 32 位存取时存取 $4 \times (n + 1)$ 字节的 CE_DATA ($n = 0, 1, 2, 3, \dots, 127$)。

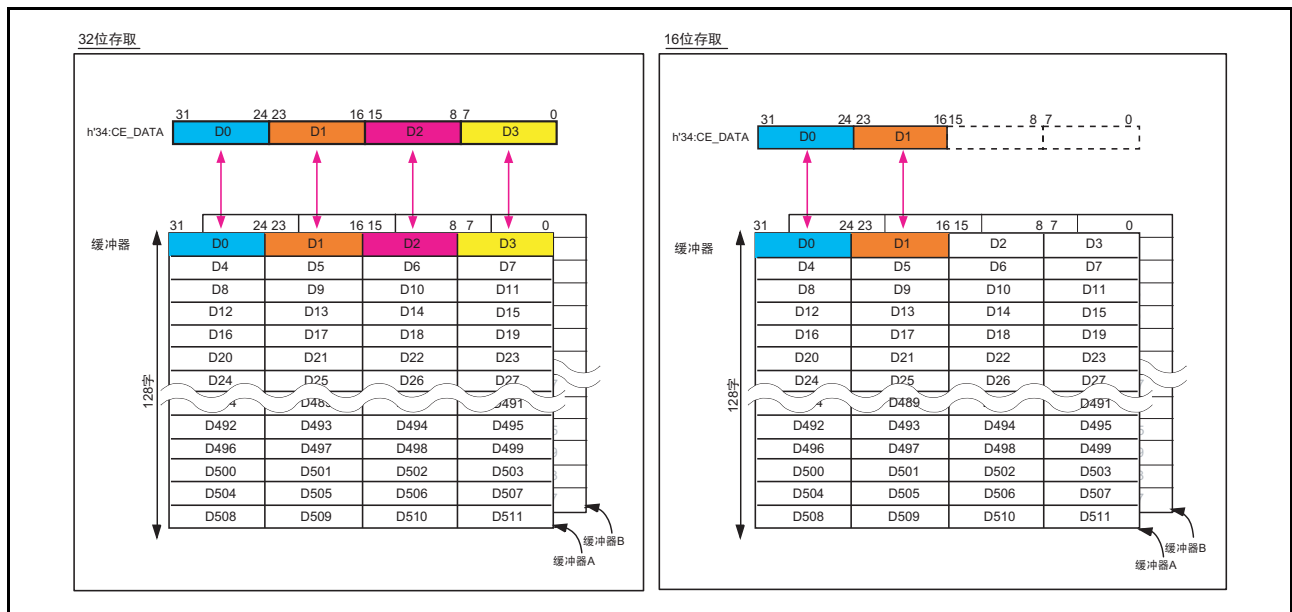


图 45.6 双缓冲器的结构

能通过 CE_BUF_ACC 的缓冲器存取选择功能，以字节为单位交换读写 CE_DATA 的数据进行缓冲器的存取。

32 位 / 16 位存取时的规格如图 45.7 所示。

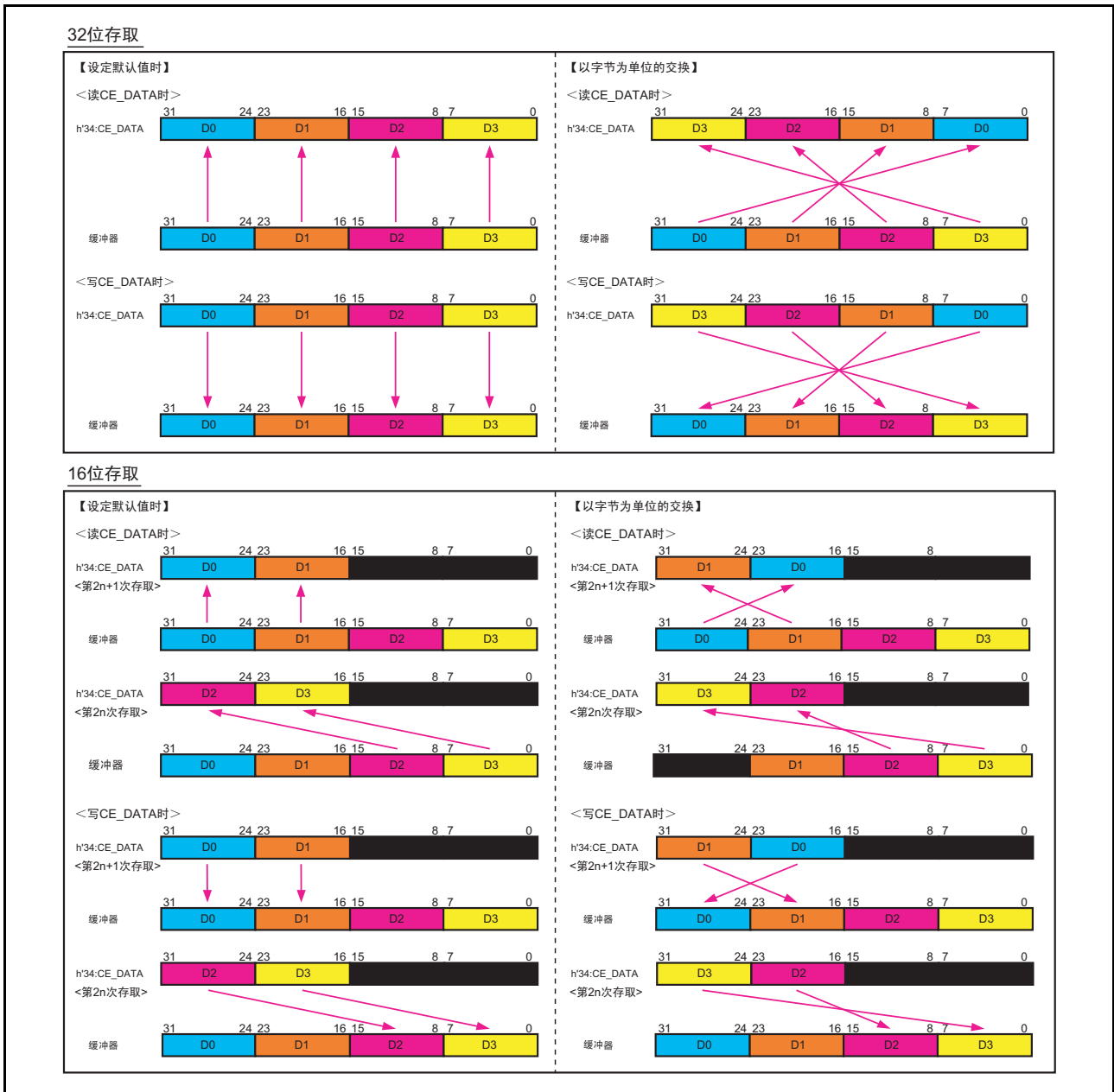


图 45.7 32 位 /16 位存取时的以字节为单位的交换规格 (n=0、1、2、……、255)

45.6.4 CMD12 的自动发行

如果将 CE_CMD_SET 的 CMD12EN 位置 “1” 并且进行多块传送，此模块就自动发行 CMD12。

多块读时的 CMD12 的自动发行时序如图 45.8 所示。在接收最后块的过程中，为了在发送数据结束位的前 2 位之前发送命令结束位，发行 CMD12。

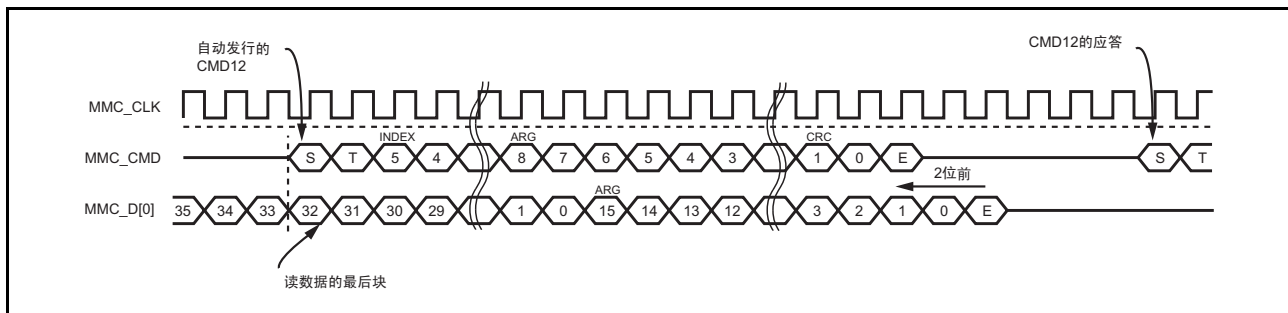


图 45.8 多块读时的 CMD12 的自动发行时序（1 位模式）

多块写时的 CMD12 的自动发行时序如图 45.9 所示。在发送最后块后的数据忙结束后发行 CMD12。

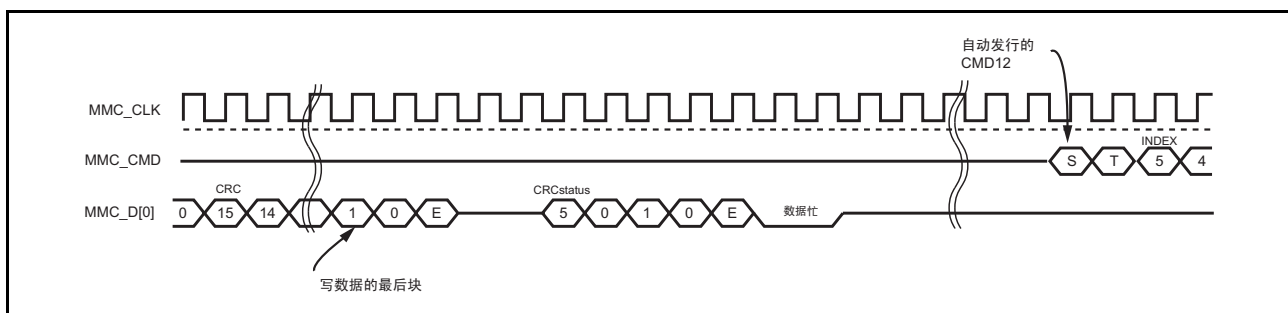


图 45.9 多块写时的 CMD12 的自动发行时序（1 位模式）

给 CE_ARG_CMD12 设定自动 CMD12 的自变量。将对 CMD12 的应答 [39:8] 位保存到 CE_RESP_CMD12。在应答接收时，接收忙的状态。

45.6.5 发生错误或者超时时的此模块的处理

当发生错误时，此模块可能不停止运行。此时，如果正在执行命令顺序（通过 CE_HOST_STS1 的 CMDSEQ 进行确认），就必须在执行强制结束后进行软件复位。不保证在发生错误时保存到缓冲器的发送数据和接收数据。

当发生超时，此模块不停止运行。如果要在发生超时时发行下一个命令，就必须在执行强制结束后进行软件复位，然后发行下一个命令。

有关强制结束，请参照“45.8 使用时的注意事项”。

45.7 设定例子

以下说明典型命令顺序的执行步骤。

45.7.1 范例说明

图中记载符号的范例如图 45.10 所示。

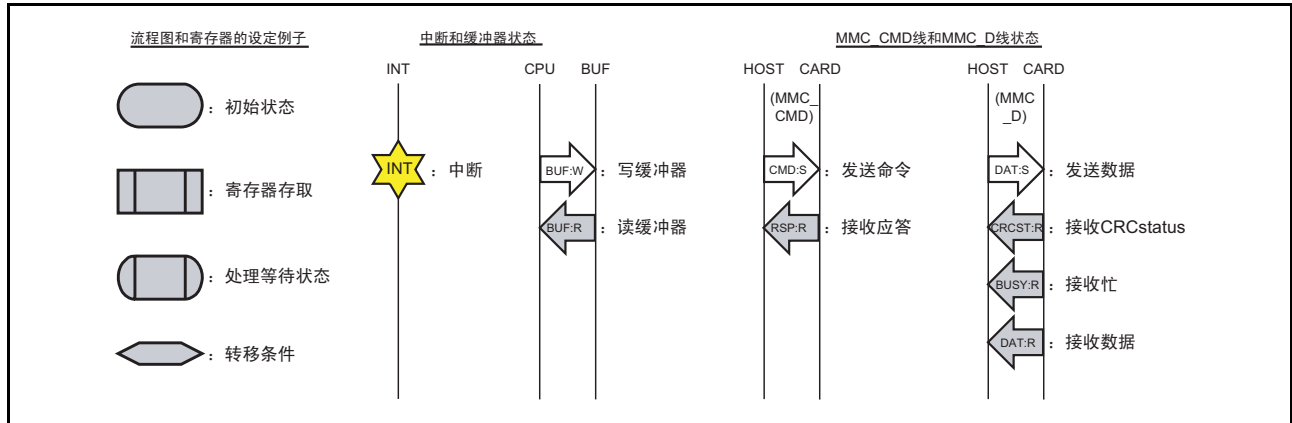


图 45.10 图中记载符号的范例

45.7.2 命令发送的设定例子

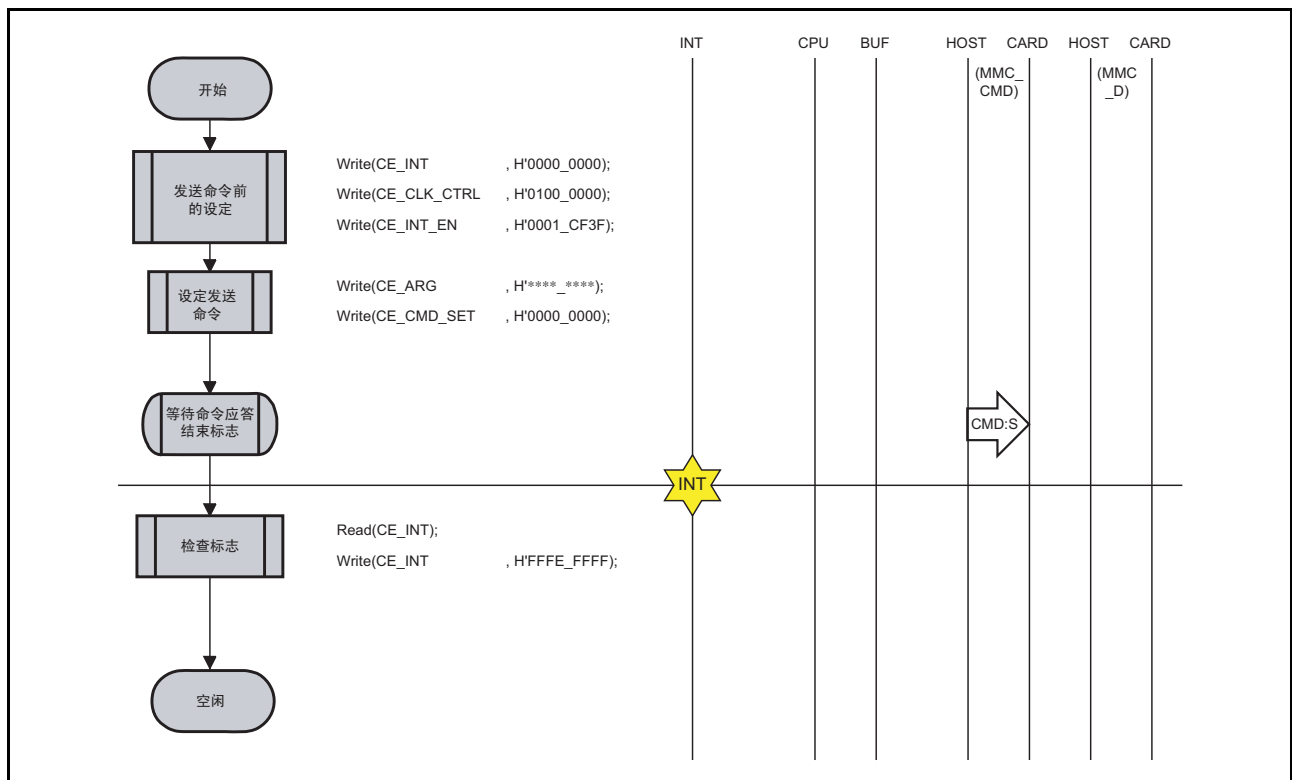


图 45.11 命令发送的设定例子 (CMD0)

45.7.3 命令发送 → 应答接收的设定例子

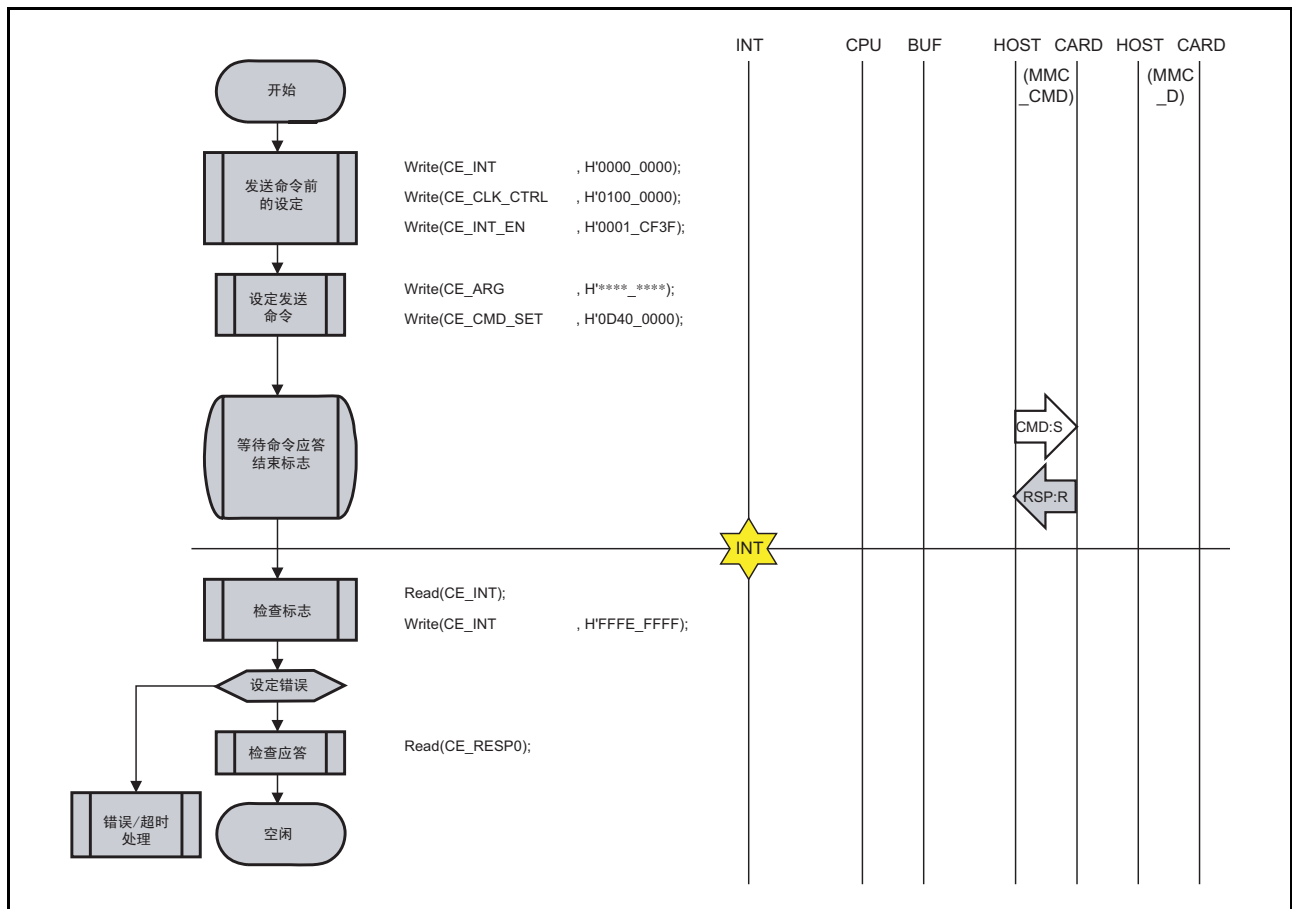


图 45.12 命令发送 → 应答接收的设定例子 (CMD3)

45.7.4 命令发送 → 应答接收（应答忙）的设定例子

- 忙的时间小于CE_CLK_CTRL的SRBSYTO设定值的情况

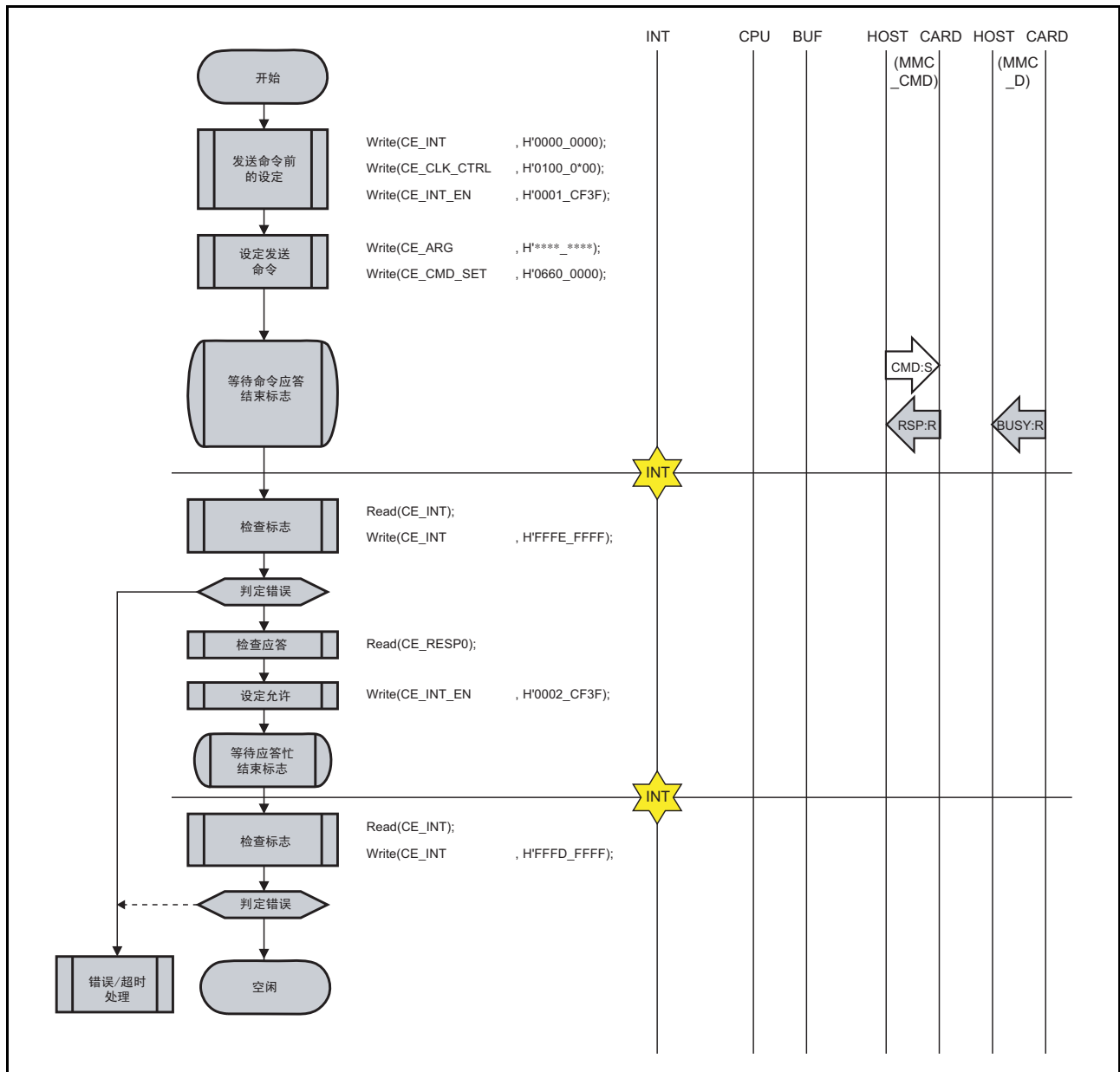


图 45.13 命令发送 → 应答接收（应答忙）的设定例子（CMD6）

- 忙的时间大于等于 CE_CLK_CTRL 的 SRBSYTO 设定值的情况

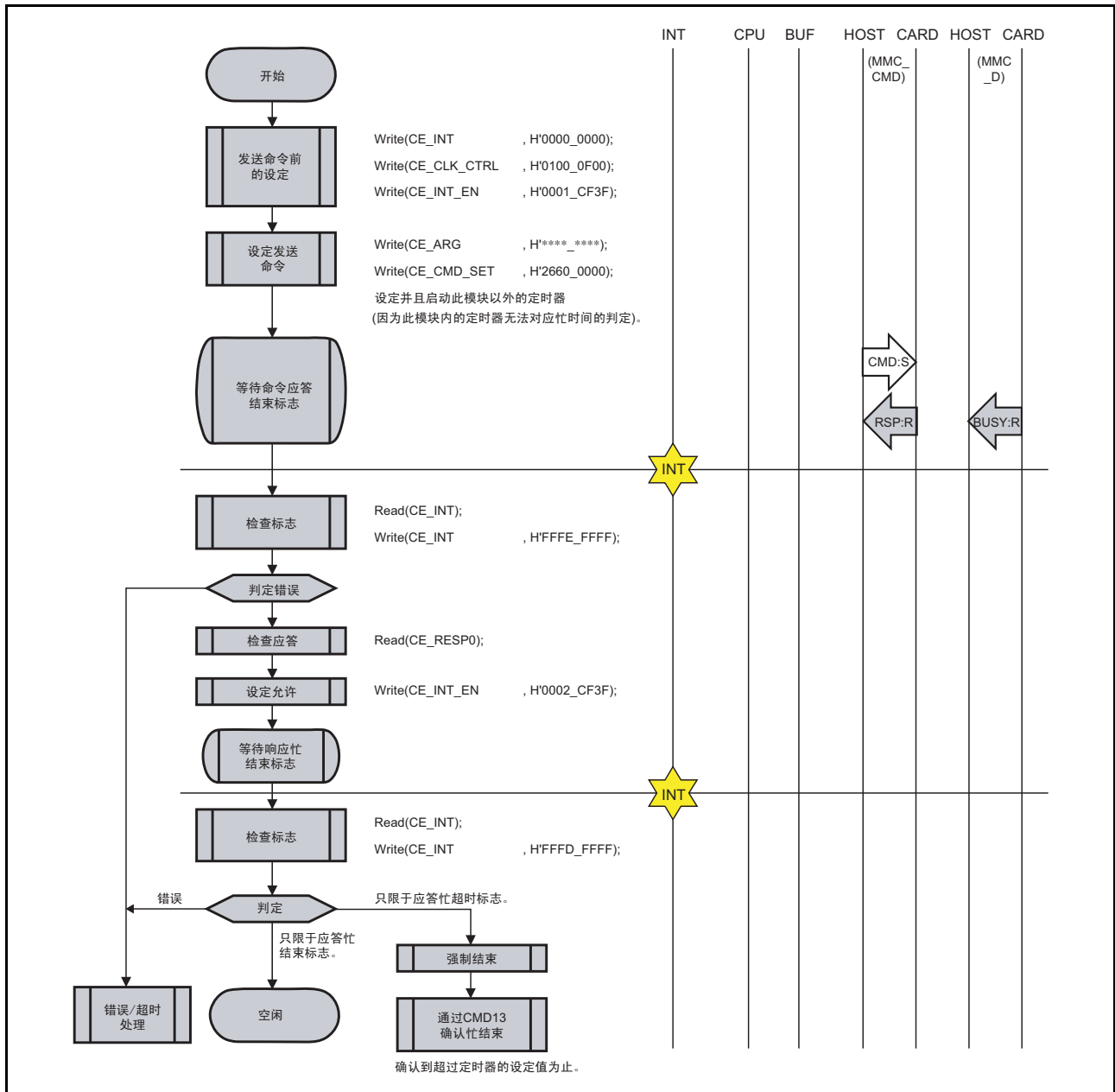


图 45.14 命令发送 → 应答接收（应答忙）的设定例子（CMD38）

45.7.5 单块读的设定例子

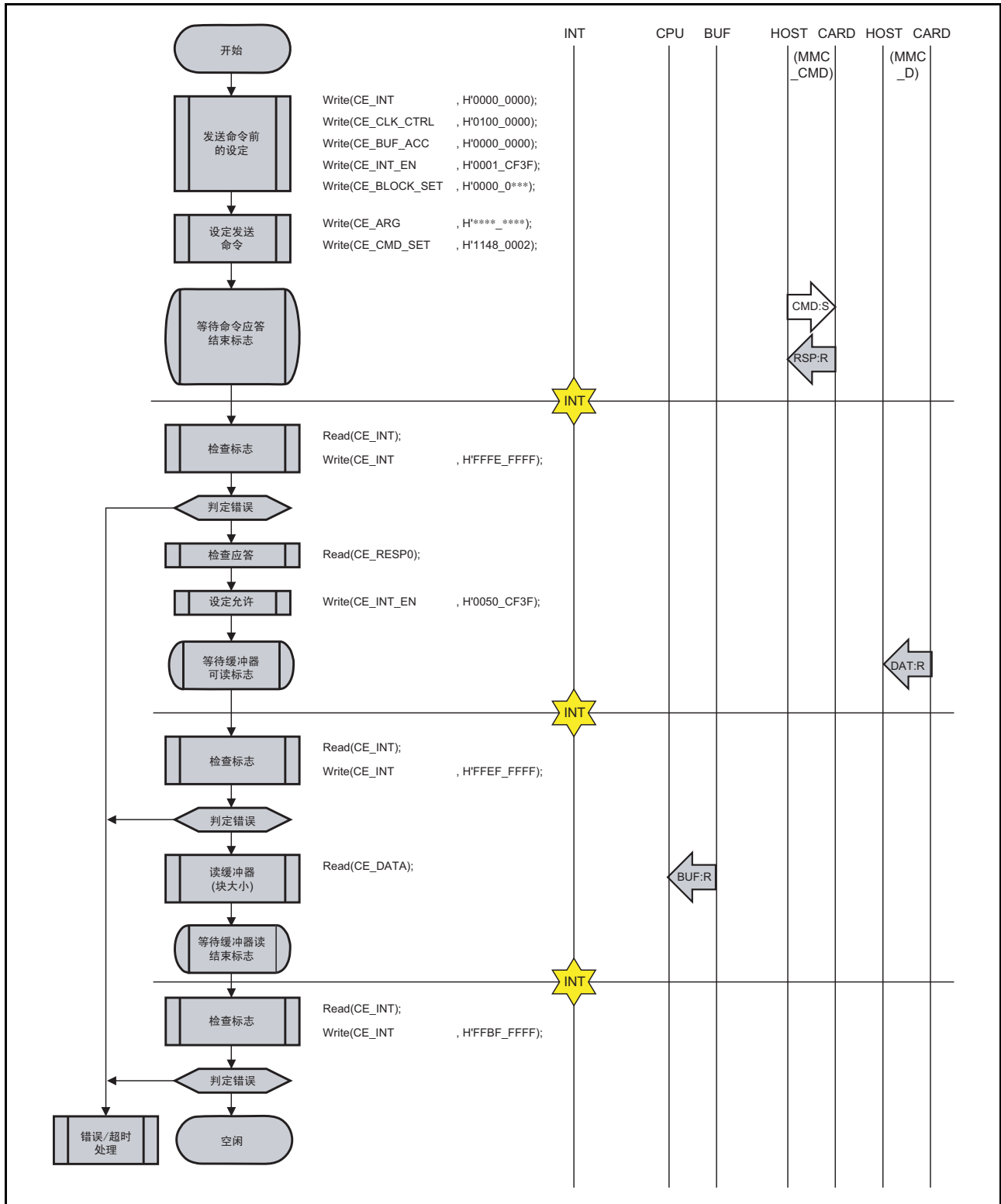


图 45.15 单块读的设定例子 (CMD17)

45.7.6 多块读的设定例子

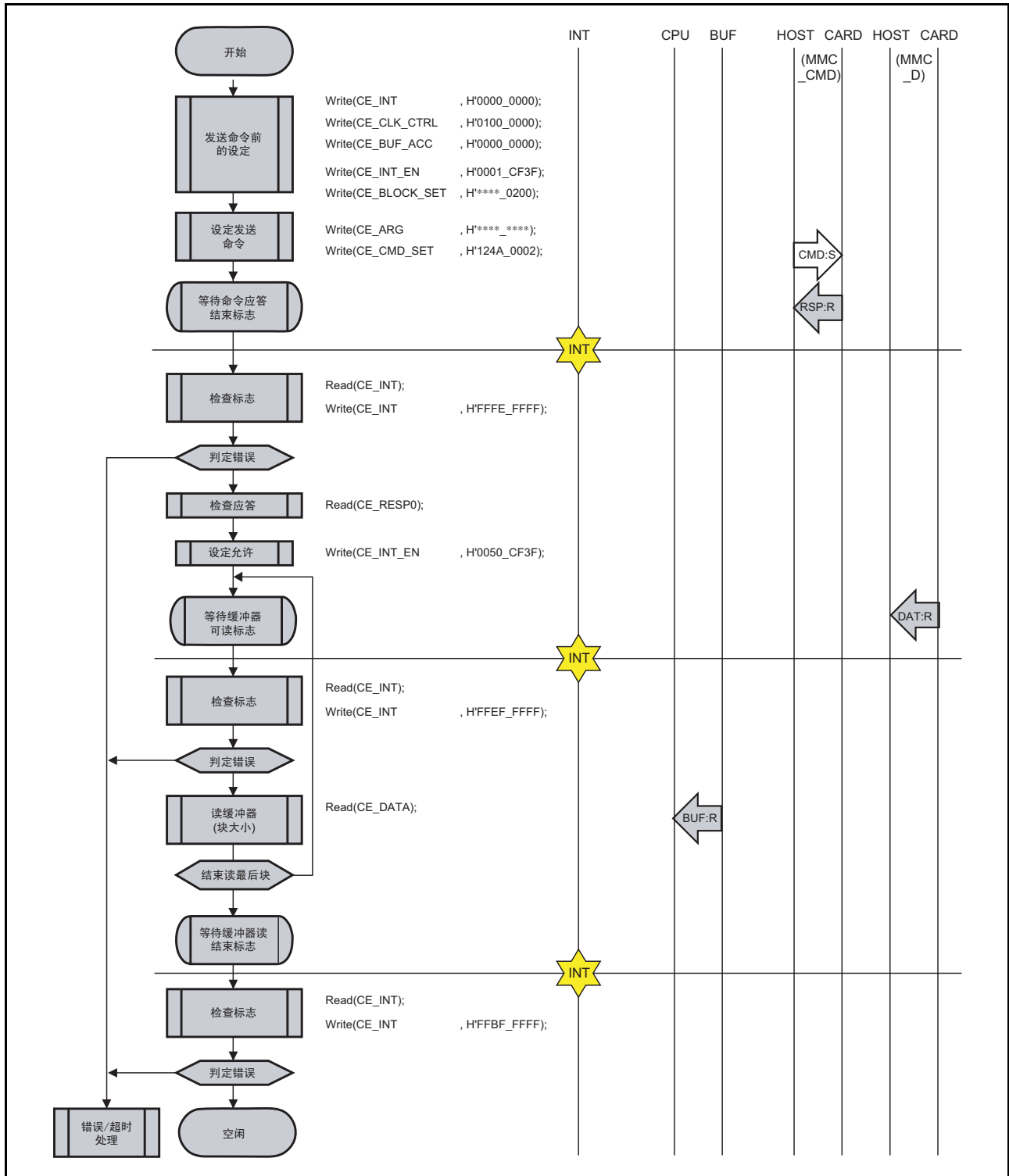


图 45.16 多块读的设定例子 (CMD18 Pre-defined)

45.7.7 多块读（有自动 CMD12）的设定例子

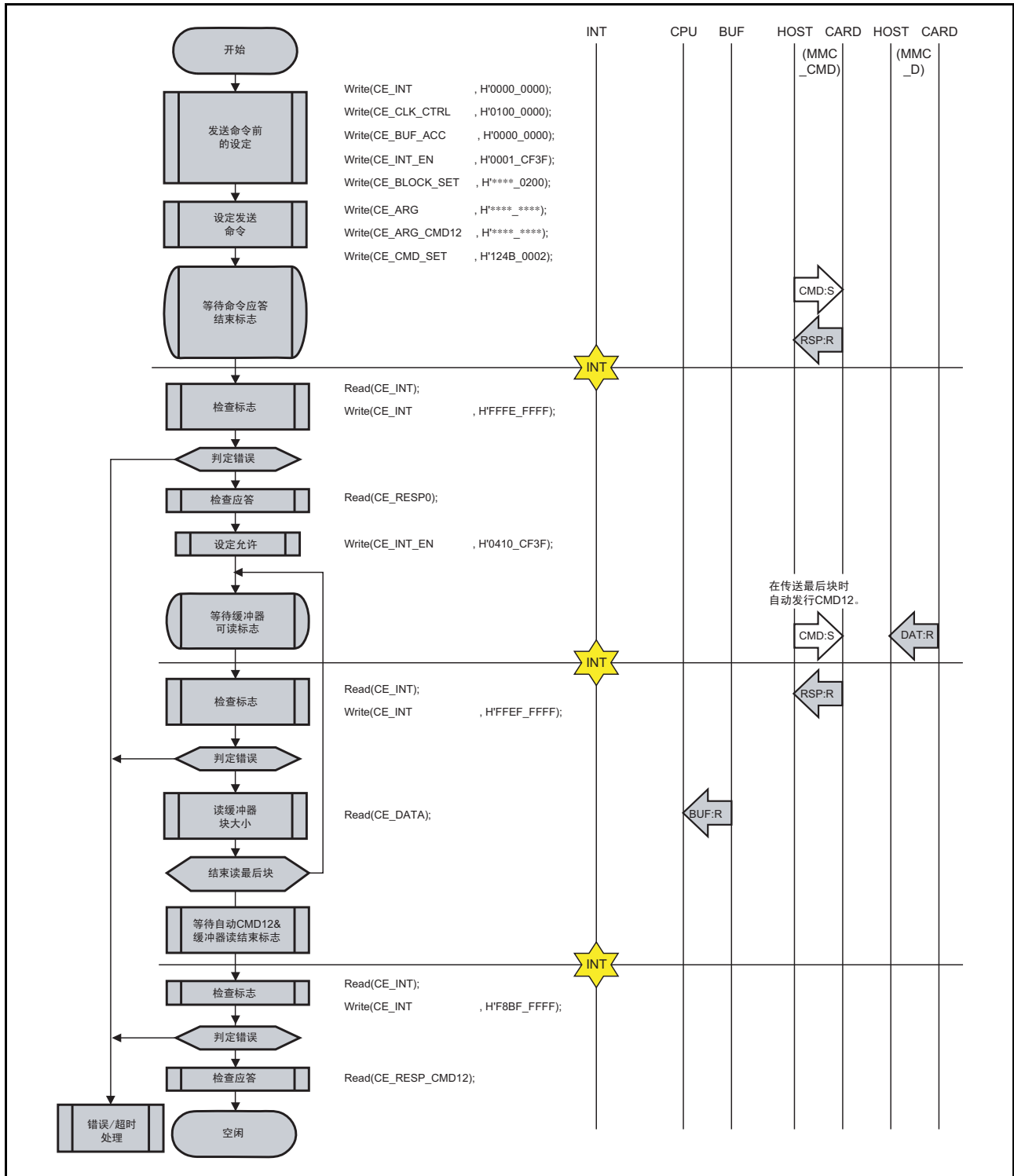


图 45.17 多块读（有自动 CMD12）的设定例子（CMD18 Open-ended）

45.7.8 单块写的设定例子

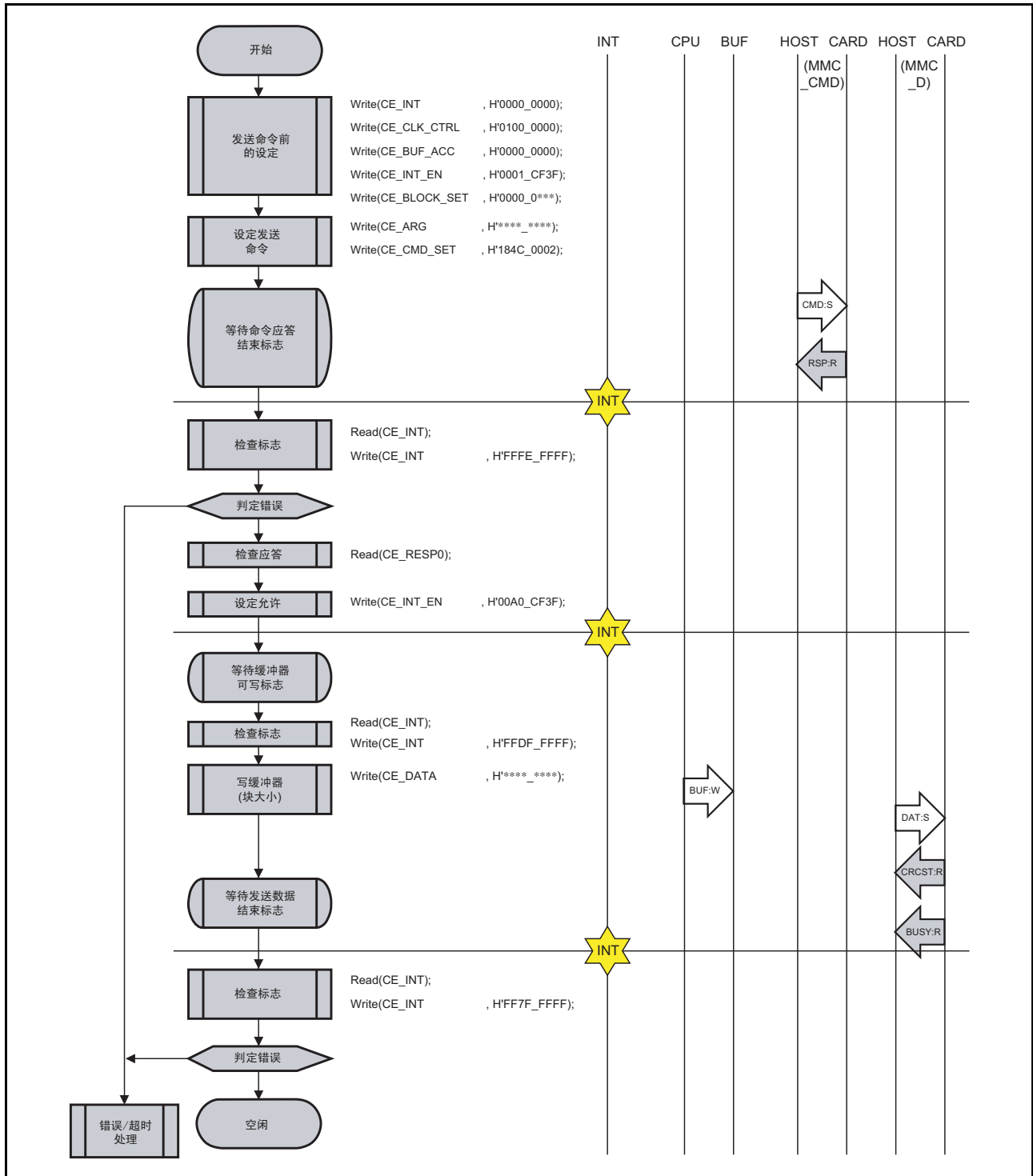


图 45.18 单块写的设定例子 (CMD24)

45.7.9 多块写的设定例子

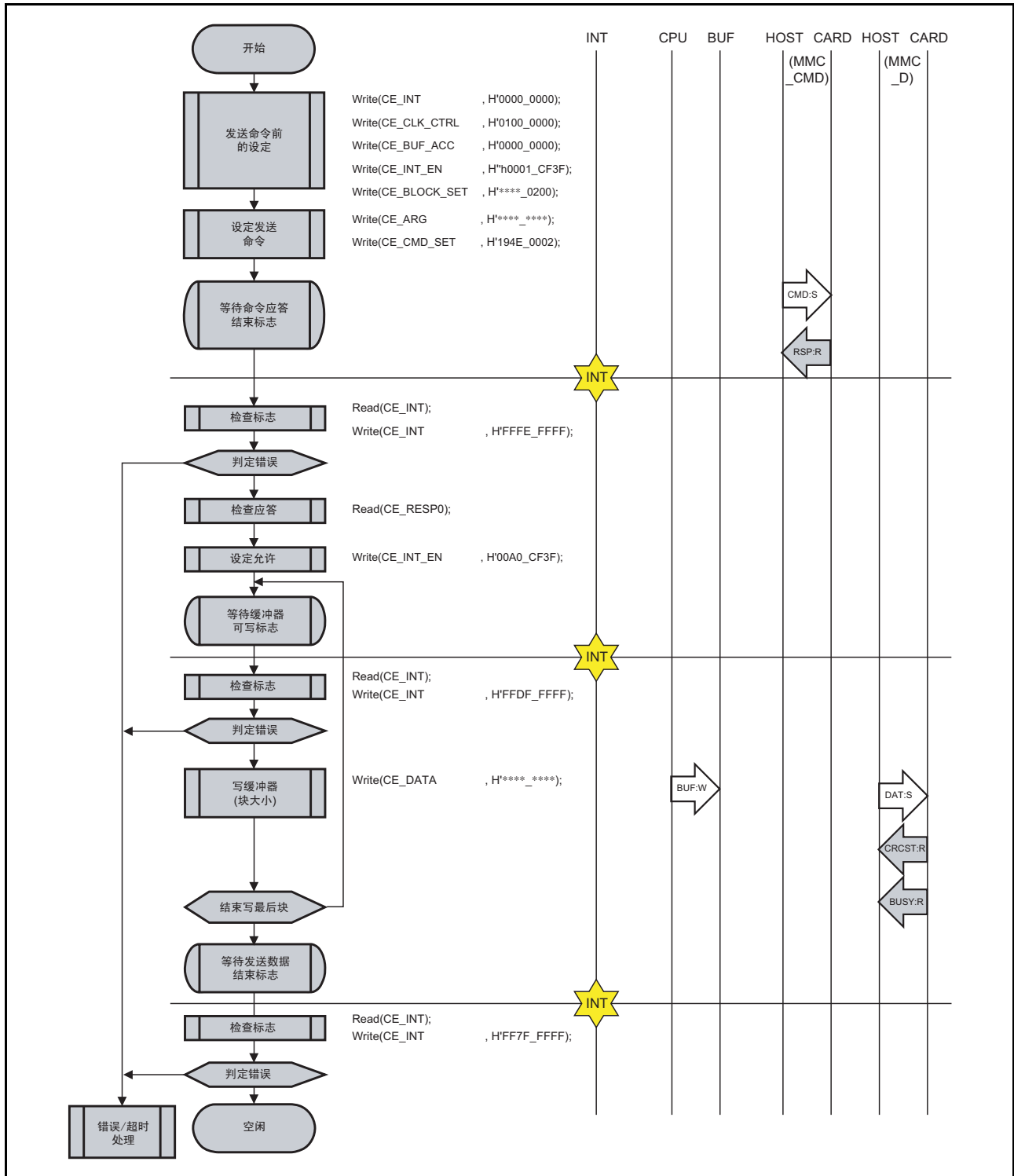


图 45.19 多块写的设定例子 (CMD25 Pre-defined)

45.7.10 多块写（有自动 CMD12）的设定例子

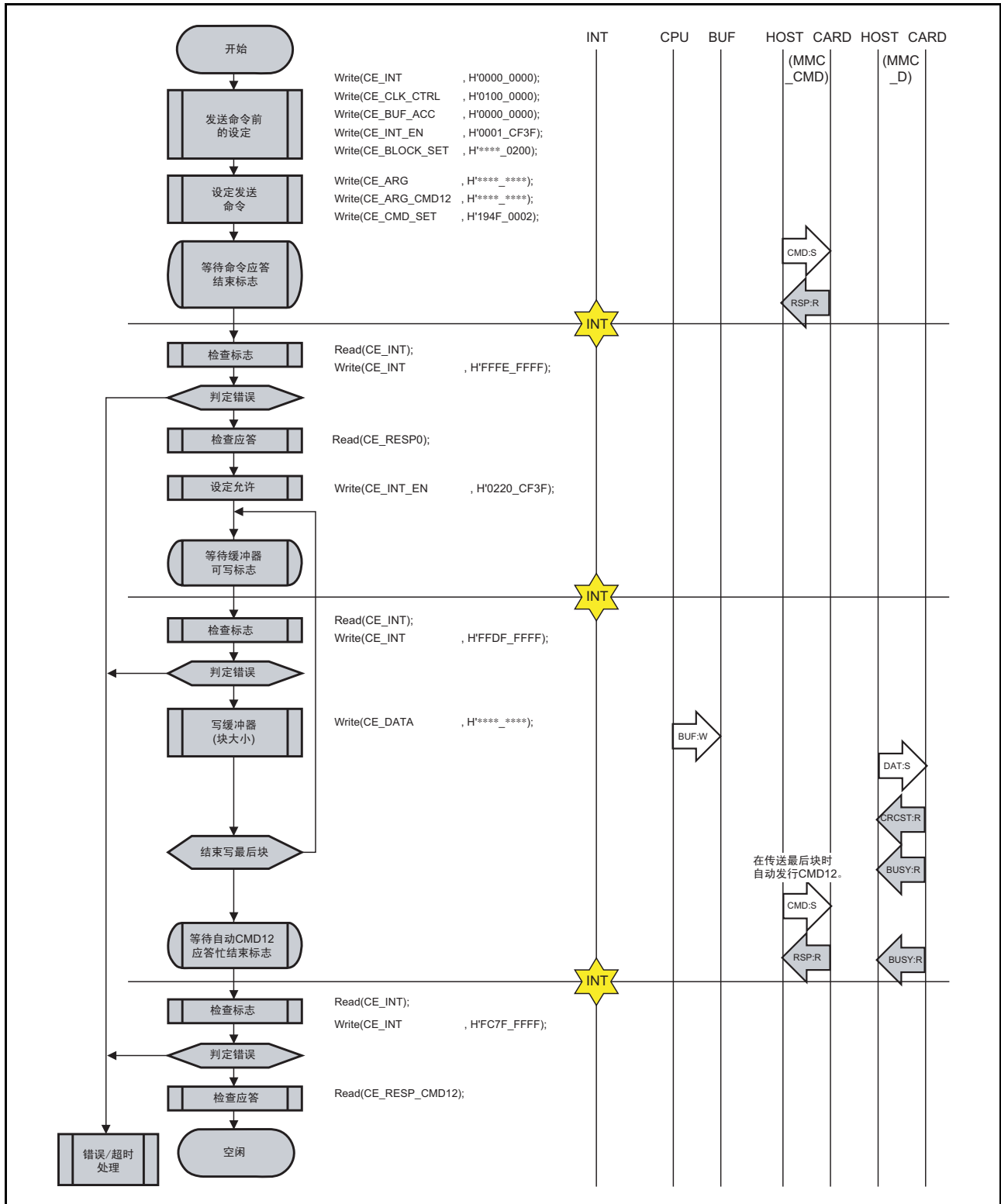


图 45.20 多块写（有自动 CMD12）的设定例子（CMD25 Open-ended）

45.7.11 强制结束的设置例子

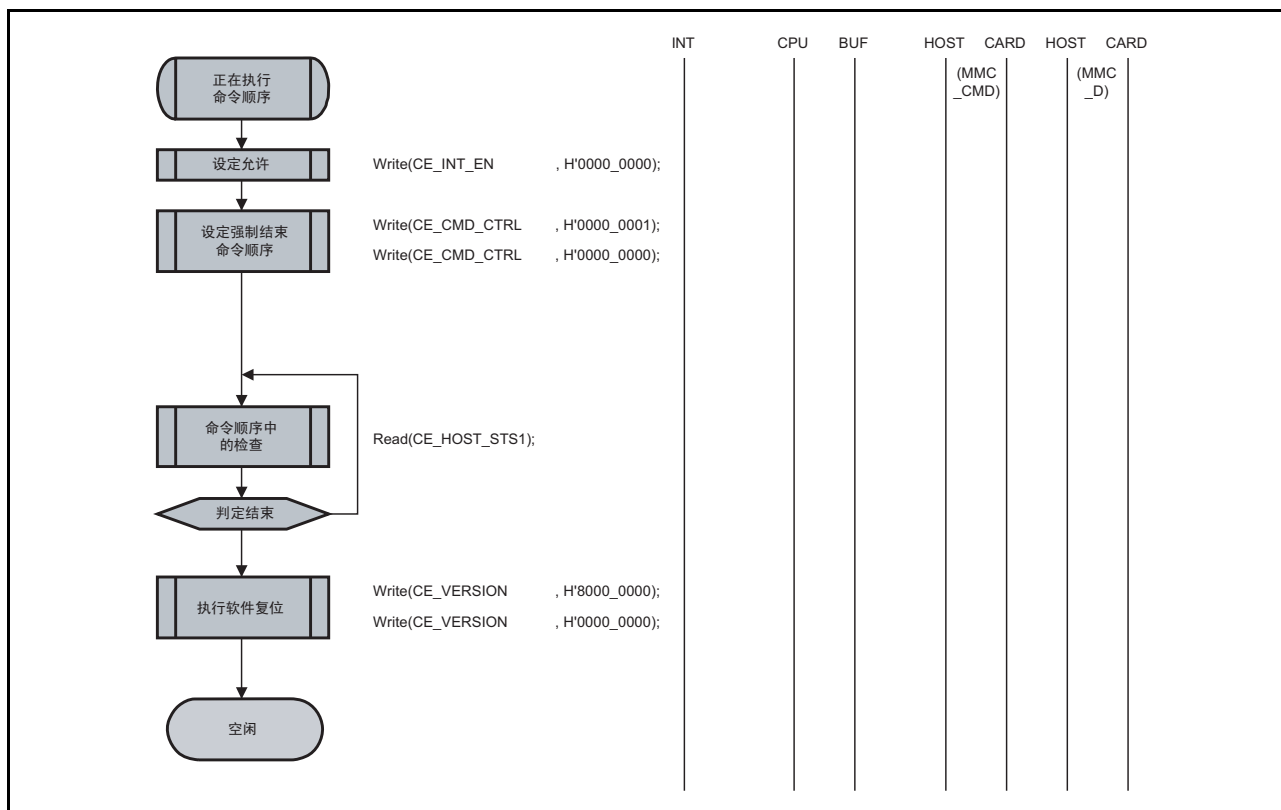


图 45.21 强制结束的设置例子

45.7.12 CE_CMD_SET 的设定值

发行命令时的设定值一览表如表 45.4 所示。

表 45.4 CE_CMD_SET 的设定值

Command	Response	CE_CMD_SET 的设定值																				备注			
		—	—	CMD[5:0]	RTYP[1:0]	RBSY	—	WDAT	DWEN	CMLTE	CMD12EN	RIDXC[1:0]	RCRC7C[1:0]	—	CRC16C	—	CRCSTE	TBIT	OPDM	—	—		SBIT	—	DATW[1:0]
CMD0	—	0	0	000000	00	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00		
CMD1	R3	0	0	000001	01	0	0	0	0	0	0	01	01	0	0	0	0	0	0	0	0	0	00		
CMD2	R2	0	0	000010	10	0	0	0	0	0	0	01	10	0	0	0	0	0	0	0	0	0	00		
CMD3	R1	0	0	000011	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00		
CMD4	—	0	0	000100	00	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00		
CMD5	R1b	0	0	000101	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00		
CMD6	R1b	0	0	000110	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00		
CMD7	R1	0	0	000111	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00		
	R1b	0	0	000111	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00		
CMD8	R1	0	0	001000	01	0	0	1	0	0	0	00	00	0	0	0	0	0	0	0	0	0	**	**	
CMD9	R2	0	0	001001	10	0	0	0	0	0	0	01	10	0	0	0	0	0	0	0	0	0	00		
CMD10	R2	0	0	001010	10	0	0	0	0	0	0	01	10	0	0	0	0	0	0	0	0	0	00		
CMD12	R1	0	0	001100	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00		
	R1b	0	0	001100	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00		
CMD13	R1	0	0	001101	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00		
CMD14	R1	0	0	001110	01	0	0	1	0	0	0	00	00	0	1	0	0	0	0	0	0	1	0	**	
CMD15	—	0	0	001111	00	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00		
CMD16	R1	0	0	010000	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00		
CMD17	R1	0	0	010001	01	0	0	1	0	0	0	00	00	0	0	0	0	0	0	0	0	0	*	**	
CMD18	R1	0	0	010010	01	0	0	1	0	1	0	00	00	0	0	0	0	0	0	0	0	0	*	**	Pre-defined
	R1	0	0	010010	01	0	0	1	0	1	1	00	00	0	0	0	0	0	0	0	0	0	*	**	Open-ended
CMD19	R1	0	0	010011	01	0	0	1	1	0	0	00	00	0	0	0	1	0	0	0	0	0	0	**	
CMD23	R1	0	0	010111	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00		
CMD24	R1	0	0	011000	01	0	0	1	1	0	0	00	00	0	0	0	0	0	0	0	0	0	*	**	
CMD25	R1	0	0	011001	01	0	0	1	1	1	0	00	00	0	0	0	0	0	0	0	0	0	*	**	Pre-defined
	R1	0	0	011001	01	0	0	1	1	1	1	00	00	0	0	0	0	0	0	0	0	0	*	**	Open-ended
CMD26	R1	0	0	011010	01	0	0	1	1	0	0	00	00	0	0	0	0	0	0	0	0	0	*	**	
CMD27	R1	0	0	011011	01	0	0	1	1	0	0	00	00	0	0	0	0	0	0	0	0	0	*	**	
CMD28	R1b	0	0	011100	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00		
CMD29	R1b	0	0	011101	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00		
CMD30	R1	0	0	011110	01	0	0	1	0	0	0	00	00	0	0	0	0	0	0	0	0	0	*	**	
CMD31	R1	0	0	011111	01	0	0	1	0	0	0	00	00	0	0	0	0	0	0	0	0	0	*	**	
CMD35	R1	0	0	100011	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00		
CMD36	R1	0	0	100100	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00		
CMD38	R1b	0	0	100110	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00		
CMD39	R4	0	0	100111	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00		

Command	Response	CE_CMD_SET 的设定值																				备注		
		—	—	CMD[5:0]	RTYP[1:0]	RBSY	—	WDAT	DWEN	CMLTE	CMD12EN	RIDXC[1:0]	RCRC7C[1:0]	—	CRC16C	—	CRCSTE	TBIT	OPDM	—	—		SBIT	—
CMD40	R5	0	0	101000	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00	Send CMD
	R5	0	0	101000	01	0	0	0	0	0	0	00	00	0	0	0	0	1	1	0	0	0	00	Send RSP
CMD42	R1	0	0	101010	01	0	0	1	1	0	0	00	00	0	0	0	0	0	0	0	0	0	**	
CMD55	R1	0	0	110111	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	00	
CMD56	R1	0	0	111000	01	0	0	1	0	0	0	00	00	0	0	0	0	0	0	0	0	*	**	Read
	R1	0	0	111000	01	0	0	1	1	0	0	00	00	0	0	0	0	0	0	0	0	0	*	**

【注】 此模块不对应 CMD11 位和 CMD20 位。

45.8 使用时的注意事项

45.8.1 卡检测

必须注意：用于此模块的卡检测功能的 CE_DETECT 的 CDRISE 位和 CDFALL 位没有抖动消除功能，必须通过软件进行抖动消除处理。

46. 马达控制 PWM 定时器

本LSI内置2个通道并且每个通道能进行最多8个脉冲输出的马达控制PWM (Pulse Width Modulator) 定时器。

46.1 特点

- 能进行最多 16 个脉冲输出。
内置 2 个通道并且每个通道有 8 个输出的 10 位 PWM。
有 10 位计数器 (PWCNT) 和周期寄存器 (PWCYR)。
能给每个脉冲输出设定占空比和输出极性。
- 每个周期能进行数据的自动传送。
4 个占空比寄存器 (PWDTRA) 各自有缓冲寄存器 (PWBFR)，每个周期自动进行数据传送。
- 能设定占空比。
能通过占空比寄存器的设定值设定 0% ~ 100% 的占空比。
- 能选择 5 种计数时钟。
能选择 5 种计数时钟 (P0φ、P0φ/2、P0φ/4、P0φ/8、P0φ/16)。
- 能通过内部 16 位总线进行高速存取。
- 中断源：2 种
能在周期寄存器比较匹配时分别向 2 个通道请求中断。
- 能进行寄存器数据的自动传送。
能通过启动直接存储器存取控制器，进行块传送和 1 个字数据的传送。
- 能设定模块停止模式。

框图如图 46.1 所示。

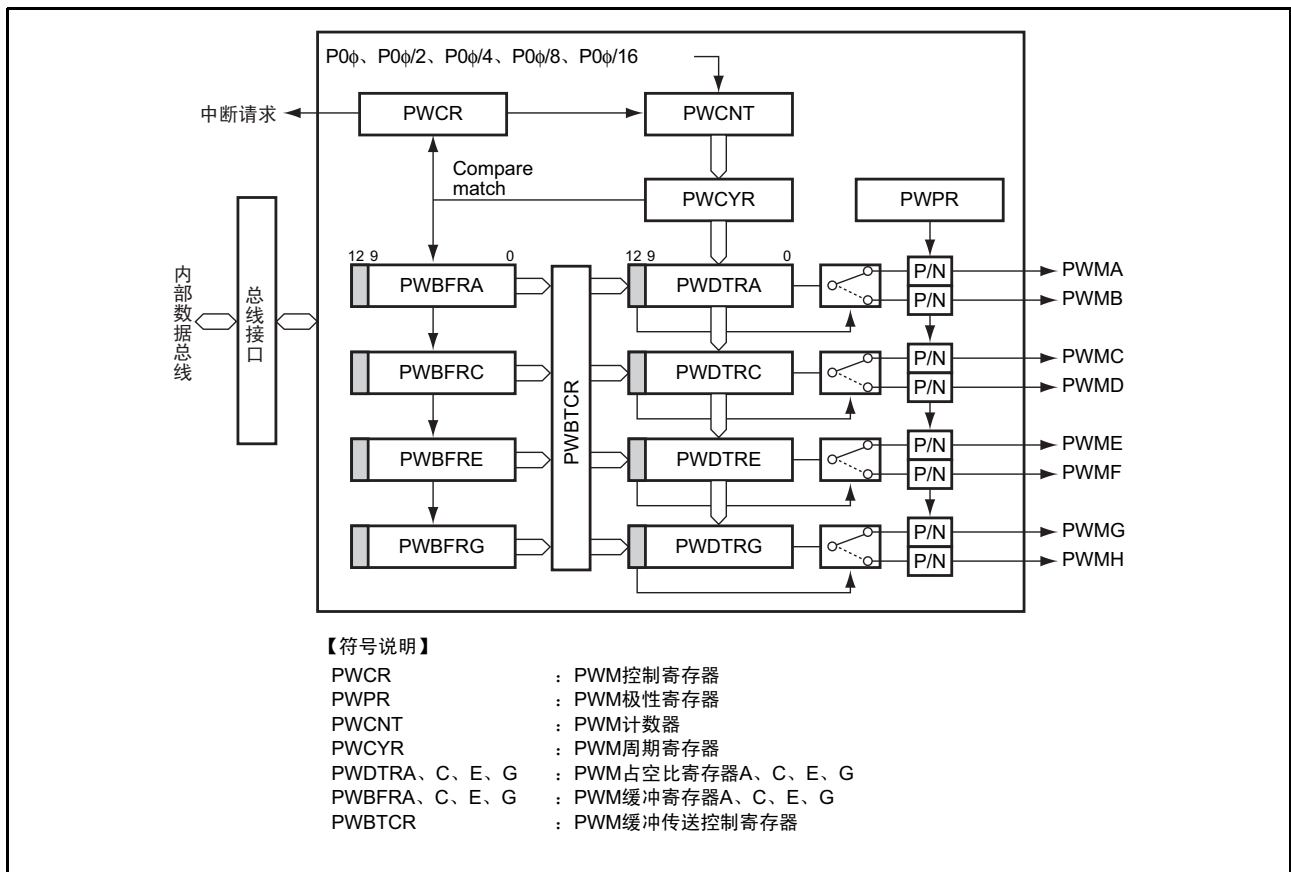


图 46.1 PWM 的框图

46.2 输入 / 输出引脚

引脚结构如表 46.1 所示。

表 46.1 引脚结构

通道	名称	符号	输入 / 输出	功能
1	PWM 输出引脚 1A	PWM1A	输出	通道 1A 的 PWM 输出
	PWM 输出引脚 1B	PWM1B	输出	通道 1B 的 PWM 输出
	PWM 输出引脚 1C	PWM1C	输出	通道 1C 的 PWM 输出
	PWM 输出引脚 1D	PWM1D	输出	通道 1D 的 PWM 输出
	PWM 输出引脚 1E	PWM1E	输出	通道 1E 的 PWM 输出
	PWM 输出引脚 1F	PWM1F	输出	通道 1F 的 PWM 输出
	PWM 输出引脚 1G	PWM1G	输出	通道 1G 的 PWM 输出
	PWM 输出引脚 1H	PWM1H	输出	通道 1H 的 PWM 输出
2	PWM 输出引脚 2A	PWM2A	输出	通道 2A 的 PWM 输出
	PWM 输出引脚 2B	PWM2B	输出	通道 2B 的 PWM 输出
	PWM 输出引脚 2C	PWM2C	输出	通道 2C 的 PWM 输出
	PWM 输出引脚 2D	PWM2D	输出	通道 2D 的 PWM 输出
	PWM 输出引脚 2E	PWM2E	输出	通道 2E 的 PWM 输出
	PWM 输出引脚 2F	PWM2F	输出	通道 2F 的 PWM 输出
	PWM 输出引脚 2G	PWM2G	输出	通道 2G 的 PWM 输出
	PWM 输出引脚 2H	PWM2H	输出	通道 2H 的 PWM 输出

46.3 寄存器说明

寄存器结构如表 46.2 所示。

表 46.2 寄存器结构

寄存器名称	略称	R/W	初始值	地址	存取长度
PWM 控制寄存器 _1	PWCR_1	R/W	H'00	H'FFFEF4E0	8、16
PWM 极性寄存器 _1	PWPR_1	R/W	H'00	H'FFFEF4E4	8、16
PWM 周期寄存器 _1	PWCYR_1	R/W	H'FFFF	H'FFFEF4E6	16
PWM 缓冲寄存器 _1A	PWBFR_1A	R/W	H'EC00	H'FFFEF4E8	16
PWM 缓冲寄存器 _1C	PWBFR_1C	R/W	H'EC00	H'FFFEF4EA	16
PWM 缓冲寄存器 _1E	PWBFR_1E	R/W	H'EC00	H'FFFEF4EC	16
PWM 缓冲寄存器 _1G	PWBFR_1G	R/W	H'EC00	H'FFFEF4EE	16
PWM 控制寄存器 _2	PWCR_2	R/W	H'00	H'FFFEF4F0	8、16
PWM 极性寄存器 _2	PWPR_2	R/W	H'00	H'FFFEF4F4	8、16
PWM 周期寄存器 _2	PWCYR_2	R/W	H'FFFF	H'FFFEF4F6	16
PWM 缓冲寄存器 _2A	PWBFR_2A	R/W	H'EC00	H'FFFEF4F8	16
PWM 缓冲寄存器 _2C	PWBFR_2C	R/W	H'EC00	H'FFFEF4FA	16
PWM 缓冲寄存器 _2E	PWBFR_2E	R/W	H'EC00	H'FFFEF4FC	16
PWM 缓冲寄存器 _2G	PWBFR_2G	R/W	H'EC00	H'FFFEF4FE	16
PWM 缓冲传送控制寄存器	PWBTCR	R/W	H'00	H'FFFEF406	8、16

46.3.1 PWM 控制寄存器_n (PWCR_n) (n=1、2)

PWCR_n 允许中断，启动或者停止计数器以及选择计数器的时钟，还有表示周期寄存器比较匹配的标志。

位:	7	6	5	4	3	2	1	0
	—	—	IE	CMF	CST	CKS2	CKS1	CKS0
初始值:	1	1	0	0	0	0	0	0
R/W:	—	—	R/W	R(W)*	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7、6	—	全 1	—	保留位 读取值总是“1”，写操作无效。
5	IE	0	R/W	中断允许 选择允许或者禁止对应通道的 PWCYR_n 发生比较匹配时的中断。 0: 禁止中断请求 1: 允许中断请求
4	CMF	0	R(W)*	比较匹配标志 表示对应通道的 PWCYR_n 发生比较匹配。 [置位条件] 当 PWCNT_n=(PWCYR_n-1) 时 [清除条件] • 在读“1”的状态后写“0”时 • 在通过比较匹配中断启动直接存储器存取控制器并且进行 DMA 传送时 (在使用中断并且通过 CPU 清除时，必须在写“0”后读此标志)
3	CST	0	R/W	计数器开始 选择启动或者停止对应通道的 PWCNT_n。 0: 停止 PWCNT_n 1: 启动 PWCNT_n
2	CKS2	0	R/W	时钟选择 选择对应通道的 PWCNT_n 的计数器时钟。 000: 通过内部时钟 P0φ 进行计数 001: 通过内部时钟 P0φ/2 进行计数 010: 通过内部时钟 P0φ/4 进行计数 011: 通过内部时钟 P0φ/8 进行计数 1xx: 通过内部时钟 P0φ/16 进行计数
1	CKS1	0	R/W	
0	CKS0	0	R/W	

【符号说明】 x: Don't care

【注】 * 为了清除标志，只能写“0”。

46.3.2 PWM 极性寄存器 $_n$ (PWPR $_n$) (n=1、2)

PWPR $_n$ 选择 PWM 输出的极性。

位:	7	6	5	4	3	2	1	0
	OPSnH	OPSnG	OPSnF	OPSnE	OPSnD	OPSnC	OPSnB	OPSnA
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	OPSnH	0	R/W	输出极性选择
6	OPSnG	0	R/W	选择各位对应的 PWM 输出的极性。
5	OPSnF	0	R/W	0: PWM 直接输出
4	OPSnE	0	R/W	1: PWM 反相输出
3	OPSnD	0	R/W	
2	OPSnC	0	R/W	
1	OPSnB	0	R/W	
0	OPSnA	0	R/W	

(n=1、2)

46.3.3 PWM 计数器 $_n$ (PWCNT $_n$) (n=1、2)

PWCNT $_n$ 是通过输入时钟进行递增计数的 10 位递增计数器。通过 PWCR $_n$ 的 CKS2 ~ CKS0 位选择要输出的时钟。CPU 不能直接存取 PWCNT $_n$ 。当 PWCR $_n$ 的 CST 位为 “0” 时，将 PWCNT $_n$ 初始化为 “HFC00”。

46.3.4 PWM 周期寄存器 $_n$ (PWCYR $_n$) (n=1、2)

PWCYR $_n$ 是设定 PWM 转换周期的 16 位可读写寄存器。

位:	15	14	13	12	11	10	9	8
	PWC Y15	PWC Y14	PWC Y13	PWC Y12	PWC Y11	PWC Y10	PWC Y9	PWC Y8
初始值:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
位:	7	6	5	4	3	2	1	0
	PWC Y7	PWC Y6	PWC Y5	PWC Y4	PWC Y3	PWC Y2	PWC Y1	PWC Y0
初始值:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

如果 PWCYR $_n$ 发生比较匹配，就清除 PWCNT $_n$ ，将数据从缓冲寄存器 (PWBFR $_n$) 传送到占空比寄存器 (PWDTR $_n$)。

必须在 PWCNT $_n$ 停止时写 PWCYR $_n$ ，而且不能给 PWCYR $_n$ 设定 “HFC00”。

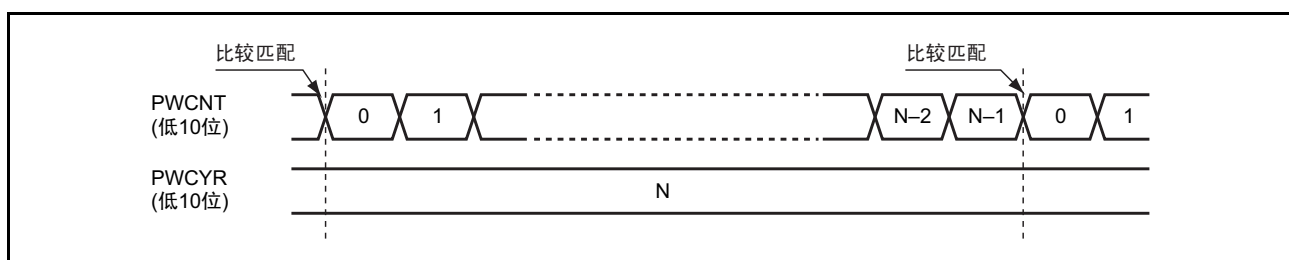


图 46.2 周期寄存器的比较匹配

46.3.5 PWM 占空比寄存器_nA、nC、nE、nG (PWDTR_nA、PWDTR_nC、PWDTR_nE、PWDTR_nG) (n=1、2)

PWDTR_n 由 4 个寄存器 (PWDTR_nA、PWDTR_nC、PWDTR_nE、PWDTR_nG) 构成。PWDTR_nA 对应 PWMnA 输出和 PWMnB 输出，PWDTR_nC 对应 PWMnC 输出和 PWMnD 输出，PWDTR_nE 对应 PWMnE 输出和 PWMnF 输出，PWDTR_nG 对应 PWMnG 输出和 PWMnH 输出。CPU 不能直接存取 PWDTR_n。如果 PWCYR_n 发生比较匹配，就将数据从缓冲寄存器 (PWBFR_n) 传送到占空比寄存器 (PWDTR_n)。当 CST 位为“0”时，将 PWDTR_n 初始化为“H'00”。

位:	15	14	13	12	11	10	9	8
	—	—	—	OTS	—	—	DT9	DT8
初始值:	—	—	—	0	—	—	0	0
R/W:	—	—	—	—	—	—	—	—
位:	7	6	5	4	3	2	1	0
	DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0
初始值:	0	0	0	0	0	0	0	0
R/W:	—	—	—	—	—	—	—	—

位	位名	初始值	R/W	说明
15 ~ 13	—	全 —	—	保留位
12	OTS	0	—	输出终端选择 选择进行 PWM 输出的引脚。未选择的引脚输出 Low 电平（当 PWPR_n 的对应位为“1”时，输出 High 电平），详细内容请参照表 46.3。
11、10	—	全 —	—	保留位
9	DT9	0	—	占空比 这是设定 PWM 输出占空比的数据。在 PWCYR_n 发生比较匹配时清除 PWCNT_n 后到 PWDTR_n 发生比较匹配的期间，输出 High 电平（当 PWPR_n 的对应位为“1”时，输出 Low 电平）。当全部位都为“0”时，没有输出 High 电平（当 PWPR_n 的对应位为“1”时，输出 Low 电平）的期间。
8	DT8	0	—	
7	DT7	0	—	
6	DT6	0	—	
5	DT5	0	—	
4	DT4	0	—	
3	DT3	0	—	
2	DT2	0	—	
1	DT1	0	—	
0	DT0	0	—	

表 46.3 OTS 位的输出选择

寄存器	bit12	说明
	OTS	
PWDTR_1A/ PWDTR_2A	0	选择 PWMnA 输出。
	1	选择 PWMnB 输出。
PWDTR_1C/ PWDTR_2C	0	选择 PWMnC 输出。
	1	选择 PWMnD 输出。
PWDTR_1E/ PWDTR_2E	0	选择 PWMnE 输出。
	1	选择 PWMnF 输出。
PWDTR_1G/ PWDTR_2G	0	选择 PWMnG 输出。
	1	选择 PWMnH 输出。

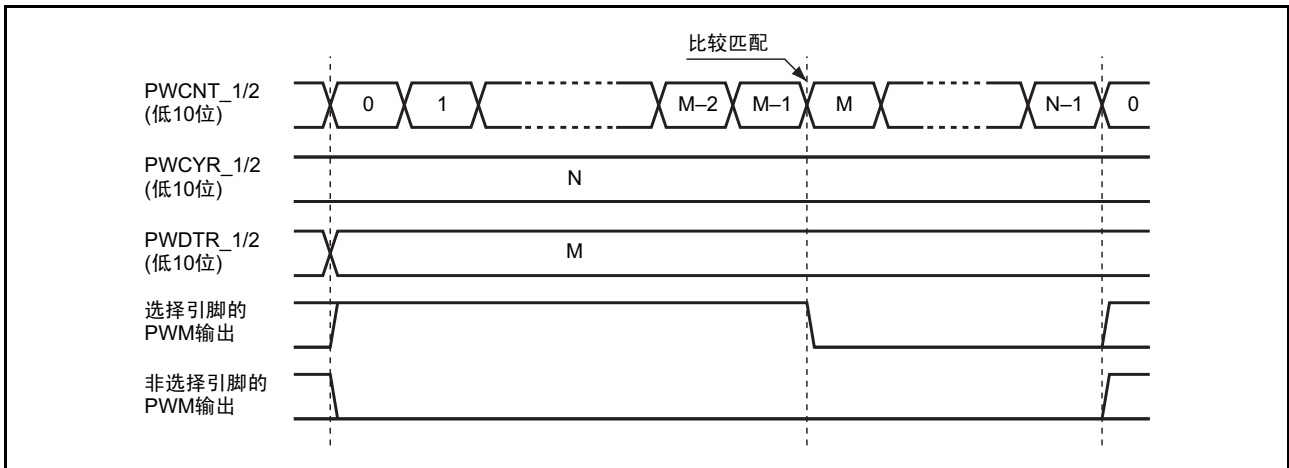


图 46.3 占空比寄存器的比较匹配 (PWPR_n 的 OPS 为 “0”)

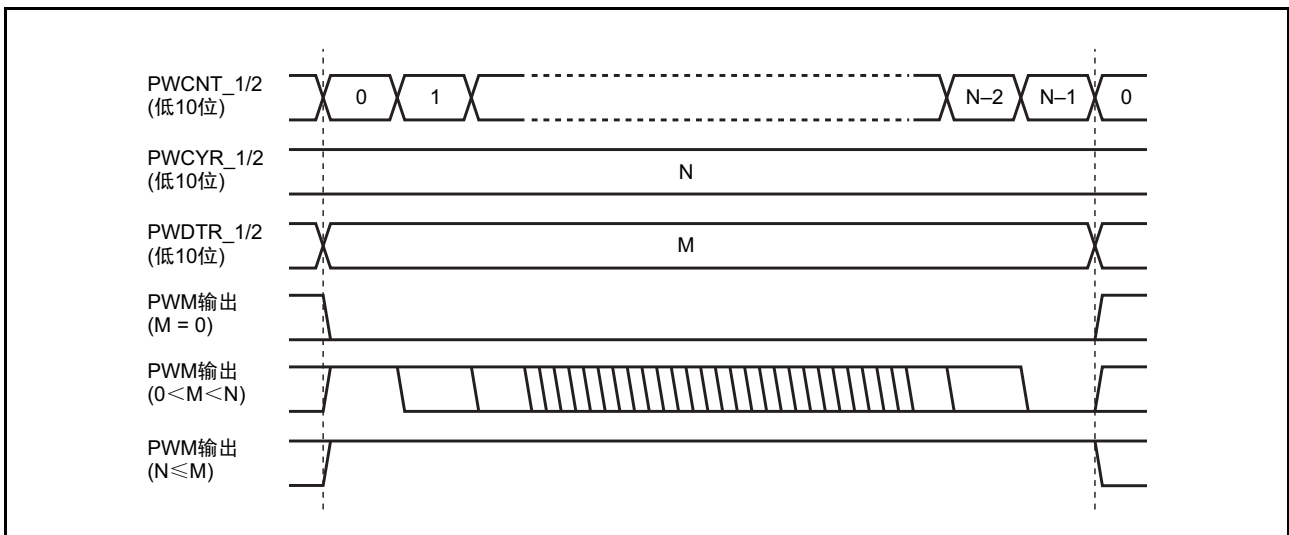


图 46.4 占空比寄存器的设定值引起的 PWM 输出的差异 (PWPR_n 的 OPS 位为 “0”)

46.3.6 PWM 缓冲寄存器_nA、_nC、_nE、_nG (PWBFR_nA、PWBFR_nC、PWBFR_nE、PWBFR_nG)

PWBFR_n 由 4 个寄存器 (PWBFR_nA、PWBFR_nC、PWBFR_nE、PWBFR_nG) 构成。如果 PWCYR 发生比较匹配, 就将数据从缓冲寄存器 (PWBFR_n) 传送到占空比寄存器 (PWDTR_n)。

位:	15	14	13	12	11	10	9	8
	—	—	—	OTS	—	—	DT9	DT8
初始值:	1	1	1	0	1	1	0	0
R/W:	R	R	R	R/W	R	R	R/W	R/W
位:	7	6	5	4	3	2	1	0
	DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 13	—	全 1	R	保留位 读取值总是 “1”，写操作无效。
12	OTS	0	R/W	输出终端选择 这是用于 PWDTR_n 的 bit12 的数据。
11、10	—	全 1	R	保留位 读取值总是 “1”，写操作无效。
9	DT9	0	R/W	占空比 这是用于 PWDTR_n 的 bit9 ~ 0 的数据。
8	DT8	0	R/W	
7	DT7	0	R/W	
6	DT6	0	R/W	
5	DT5	0	R/W	
4	DT4	0	R/W	
3	DT3	0	R/W	
2	DT2	0	R/W	
1	DT1	0	R/W	
0	DT0	0	R/W	

46.3.7 PWM 缓冲传送控制寄存器 (PWBTCR)

PWBTCR 禁止或者允许由 PWM 计数器和 PWM 周期寄存器的比较匹配引起的缓冲寄存器到占空比寄存器的数据传送。

位:	7	6	5	4	3	2	1	0
	BTC2G	BTC2E	BTC2C	BTC2A	BTC1G	BTC1E	BTC1C	BTC1A
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	BTC2G	0	R/W	0: 允许在 PWCNT_n 和 PWCYR_n 比较匹配时将数据从 PWBFR_n 传送到 PWDTR_n 1: 禁止在 PWCNT_n 和 PWCYR_n 比较匹配时将数据从 PWBFR_n 传送到 PWDTR_n
6	BTC2E	0	R/W	
5	BTC2C	0	R/W	
4	BTC2A	0	R/W	
3	BTC1G	0	R/W	
2	BTC1E	0	R/W	
1	BTC1C	0	R/W	
0	BTC1A	0	R/W	

46.4 和总线主控的接口

46.4.1 16 位数据寄存器

PWCYR_n 和 PWBFR_n 是 16 位寄存器。因为和总线主控之间的数据总线宽度为 16 位，所以能以 16 位为单位而不能以 8 位为单位读写这些寄存器。必须总是以 16 位为单位存取这些寄存器。

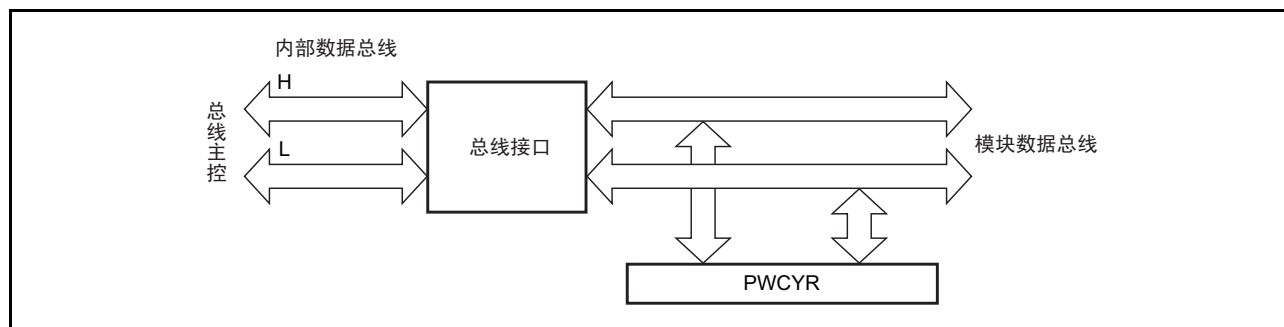


图 46.5 16 位寄存器的存取（总线主控 ↔ PWCYR_n（16 位））

46.4.2 8 位数据寄存器

PWCR_n、PWPR_n 和 PWBTCR 是 8 位寄存器，能以 8 位为单位读写这些寄存器。因为和总线主控之间的数据总线宽度为 16 位，所以能以 16 位为单位读写这些寄存器，但是低 8 位的读取值总是为“H'FF”。

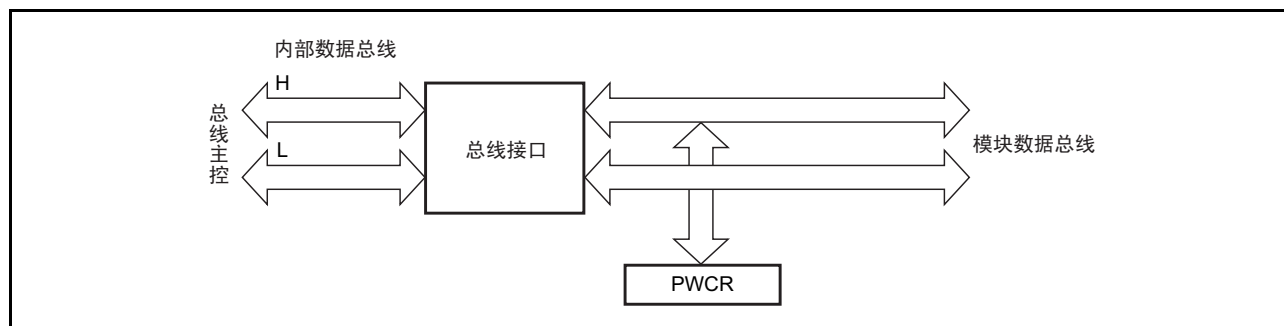


图 46.6 8 位寄存器的存取（总线主控 ↔ PWCR_n（高 8 位））

46.5 运行说明

46.5.1 PWM 的运行

从 PWM1A ~ 1H 引脚和 PWM2A ~ 2H 引脚输出图 46.7 所示的 PWM 波形。

(1) 初始设定

通过 PWPR_n 设定 PWM 输出的极性。通过 PWCR_n 的 CKS2 ~ CKS0 位选择要输入到 PWCNT_n 的时钟。

通过 PWCYR_n 设定 PWM 的转换周期。给 PWBFR_nA、PWBFR_nC、PWBFR_nE 和 PWBFR_nG 设定第 1 帧的数据。

(2) 启动

如果将 PWCR_n 的 CST 位置 “1”，PWCNT_n 就开始递增计数。如果 PWCNT_n 和 PWCYR_n 发生比较匹配，就将数据从缓冲寄存器传送到占空比寄存器并且将 PWCR_n 的 CMF 位置 “1”。此时，如果将 PWCR_n 的 IE 位置 “1”，就能产生中断请求或者启动直接存储器存取控制器。

(3) 波形输出

如果 PWCNT_n 和 PWCYR_n 发生比较匹配，PWDTR_nA、PWDTR_nC、PWDTR_nE 和 PWDTR_nG 的 OTS 位选择的 PWM 就输出 High 电平。OTS 位未选择的 PWM 输出 Low 电平。如果 PWCNT_n 和 PWDTR_nA、PWDTR_nC、PWDTR_nE、PWDTR_nG 发生比较匹配，就从对应的 PWM 输出 Low 电平。当 PWPR_n 的对应位为 “1” 时，进行反相输出。

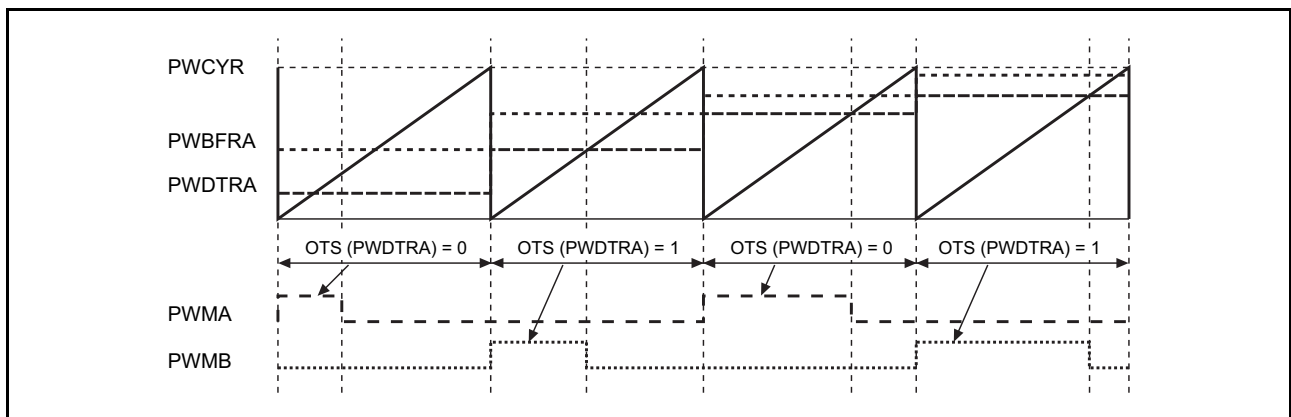


图 46.7 PWM 的运行

(4) 下一帧

PWCNT_n 和 PWCYR_n 发生比较匹配并且将数据从缓冲寄存器传送到占空比寄存器。PWCNT_n 被复位并且从 “H'000” 开始进行递增计数。如果将 PWCR_n 的 CMF 位和 IE 位置位，就能产生中断请求或者启动直接存储器存取控制器。

(5) 停止

如果将 PWCR_n 的 CST 位清 “0”，就对 PWCNT_n 进行复位并且停止。从各 PWM 输出 Low 电平（当 PWPR_n 的对应位为 “1” 时，输出 High 电平）。

46.5.2 缓冲器的传送控制

如果将 PWM 缓冲器传送控制寄存器的对应位置位，就禁止由比较匹配引起的缓冲器传送。即使在改写缓冲寄存器的过程中发生比较匹配，也不会引起输出变化。如果清除对应的位，就在以后发生比较匹配时进行缓冲器传送。

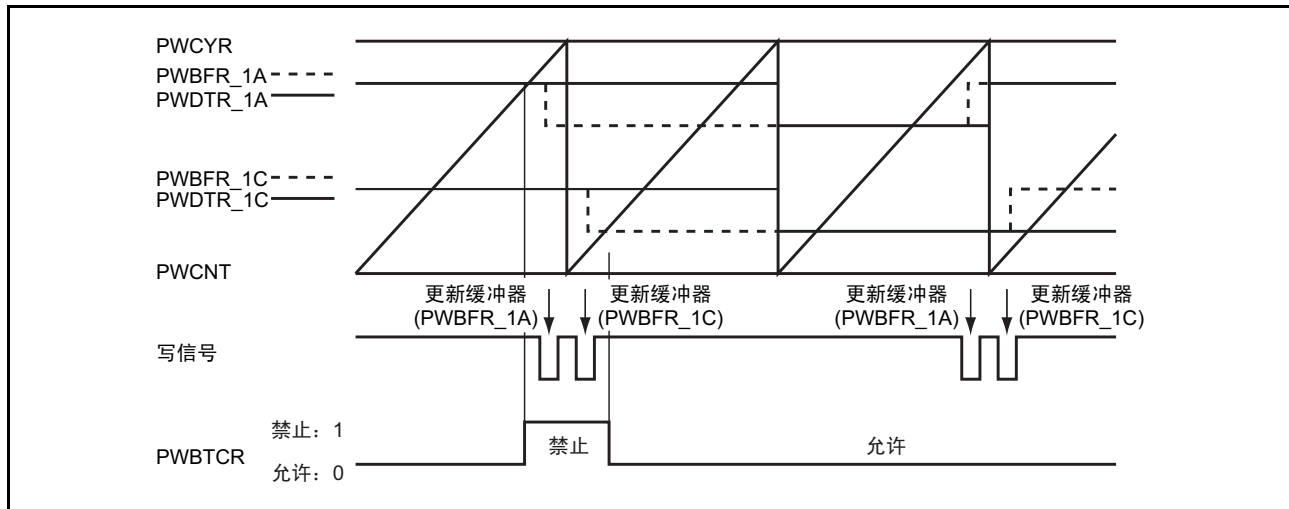


图 46.8 缓冲器传送的禁止运行说明

46.6 使用注意事项

46.6.1 缓冲寄存器的写和比较匹配的竞争

如果在周期寄存器比较匹配之后的状态下写 PWBFR_n，就同时改写缓冲寄存器和占空比寄存器。因为在周期寄存器发生比较匹配时变化的 PWM 输出在通过竞争改写占空比寄存器时不变，所以有可能输出意料之外的占空比。

必须在通过直接存储器存取控制器进行自动传送或者通过比较匹配中断进行异常处理或者检测到 PWCR_n 的 CMF（比较匹配标志）上升后到周期寄存器发生比较匹配前，结束缓冲寄存器的改写。

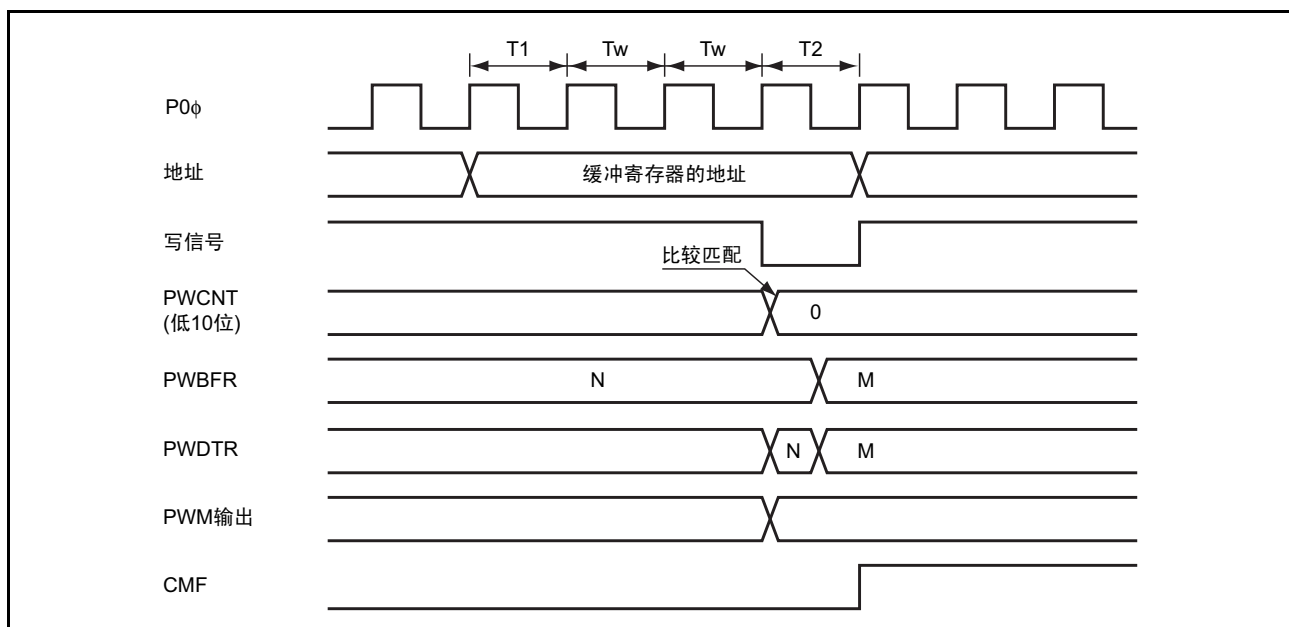


图 46.9 缓冲寄存器的写和比较匹配的竞争

47. 内部 RAM

本 LSI 内置能高速存取的高速内部 RAM、用于显示区和工作区的大容量内部 RAM（其中 128KB 和用于保持的内部 RAM 共享）以及在深度待机模式中也能保持数据的内部 RAM，能保存指令和数据。

高速内部 RAM 和大容量内部 RAM（包括用于保持的内部 RAM）能通过 RAM 允许和写允许，禁止存储器的运行和写操作。

用于保持的内部 RAM 分配在大容量内部 RAM 的页面 0，能按用于保持的内部 RAM 的页面选择在深度待机模式中是否保持数据。

47.1 特点

- 页面
高速内部 RAM 由 4 个页面构成，每个页面的容量为 16K 字节。
大容量内部 RAM 由 6 个页面构成。
用于保持的内部 RAM 由 4 个页面构成。页面 0 的容量为 16K 字节，页面 1 为 16K 字节，页面 2 为 32K 字节，页面 3 为 64K 字节。
- 存储器映像
内部 RAM 分配在表 47.1～表 47.3 的地址空间。

表 47.1 高速内部 RAM 的地址空间

页面	地址
页面 0	H'FFF80000 ~ H'FFF83FFF
页面 1	H'FFF84000 ~ H'FFF87FFF
页面 2	H'FFF88000 ~ H'FFF8BFFF
页面 3	H'FFF8C000 ~ H'FFF8FFFF

表 47.2 大容量内部 RAM 的地址空间

页面	高速缓存有效地址	高速缓存无效地址
页面 0 (256KB)	H'1C000000 ~ H'1C03FFFF	H'3C000000 ~ H'3C03FFFF
页面 1 (256KB)	H'1C040000 ~ H'1C07FFFF	H'3C040000 ~ H'3C07FFFF
页面 2 (512KB)	H'1C080000 ~ H'1C0FFFFFF	H'3C080000 ~ H'3C0FFFFFF
页面 3 (512KB)	H'1C100000 ~ H'1C17FFFF	H'3C100000 ~ H'3C17FFFF
页面 4 (512KB)	H'1C180000 ~ H'1C1FFFFFF	H'3C180000 ~ H'3C1FFFFFF
页面 5 (512KB)	H'1C200000 ~ H'1C27FFFF	H'3C200000 ~ H'3C27FFFF

表 47.3 用于保持的内部 RAM 的地址空间

页面	高速缓存有效地址	高速缓存无效地址
页面 0 (16KB)	H'1C000000 ~ H'1C003FFF	H'3C000000 ~ H'3C003FFF
页面 1 (16KB)	H'1C004000 ~ H'1C007FFF	H'3C004000 ~ H'3C007FFF
页面 2 (32KB)	H'1C008000 ~ H'1C00FFFF	H'3C008000 ~ H'3C00FFFF
页面 3 (64KB)	H'1C010000 ~ H'1C01FFFF	H'3C010000 ~ H'3C01FFFF

- 端口

高速内部RAM的各页面都有2个独立的读写端口，连接内部DMA总线（ID总线）、CPU取指令总线（F总线）和CPU存储器存取总线（M总线）（但是，F总线只连接读端口）。F总线和M总线用于CPU存取，ID总线用于DMAC存取。

大容量内部RAM的各页面都有1个读写端口，连接内部CPU总线（IC总线）、内部DMA总线（ID总线）和内部图形总线（IV1~4、RGP1~4）。因为用于保持的内部RAM包括在大容量内部RAM的页面0内，所以共享页面0的读写端口。
- 优先级

如果不同总线同时对高速内部RAM的相同页面产生存取请求，就根据优先级处理存取。优先级为ID总线>M总线>F总线。

如果不同总线同时对大容量内部RAM的相同页面产生存取请求，就根据优先级处理存取。优先级为（A）>RGP1总线>RGP2总线>RGP3总线>RGP4总线。（A）为循环运行，在IV1总线（读）、IV1总线（写）、IV2总线、IV3总线、IV4总线、IC总线和ID总线之间进行优先级的交换。
- 存取周期数

高速内部RAM：F总线和M总线的读存取周期数和写存取周期数都为1个 $I\phi$ 周期。

ID总线的存取周期取决于CPU时钟（ $I\phi$ ）和内部总线时钟（ $B\phi$ ）的时钟比。

ID总线的存取周期数如表47.4所示。

表 47.4 高速内部 RAM 的 ID 总线存取周期数

读 / 写	$I\phi:B\phi$ 的时钟比	存取周期数 ($B\phi$)
读	1:1	3
	2:1	2
	3:1	2
	4:1	2
	6:1	1
	8:1	1
写	1:1	2
	2:1	2
	3:1	2
	4:1	2
	6:1	1
	8:1	1

【注】 有关能设定的 $I\phi$ 和 $B\phi$ 的时钟比，请参照“5. 时钟脉冲振荡器”。

大容量内部RAM：全部总线的读存取周期数和写存取周期数都为1个 $B\phi$ 周期。

47.2 使用时的注意事项

47.2.1 页面竞争

如果不同总线同时对高速内部 RAM 或者大容量内部 RAM 的相同页面产生存取请求，就会发生页面竞争。虽然各存取能正常结束，但是此竞争会降低存储器的存取性能。因此，为了尽量避免竞争，建议使用软件对策。例如，如果各总线存取不同的页面，就不会发生竞争。

47.2.2 RAME 位和 RAMWE 位

对于高速内部 RAM，在禁止设定 RAME 位和 RAMWE 位时，必须在设定 RAME 位和 RAMWE 位前对各页面执行任意相同地址的读写，否则有可能无法将各页面的最后数据写到 RAM。

```
//对页面0
MOV.L #H'FFF80000, R0
MOV.L @R0, R1
MOV.L R1, @R0

//对页面1
MOV.L #H'FFF84000, R0
MOV.L @R0, R1
MOV.L R1, @R0

//对页面2
MOV.L #H'FFF88000, R0
MOV.L @R0, R1
MOV.L R1, @R0

//对页面3
MOV.L #H'FFF8C000, R0
MOV.L @R0, R1
MOV.L R1, @R0
```

图 47.1 执行例子

47.2.3 数据保持

在上电复位和深度待机模式以外的运行状态中，高速内部 RAM 和大容量内部 RAM（包括用于保持的内部 RAM）继续保持数据。在上电复位和深度待机模式中，进行以下运行：

(1) 上电复位

(a) 高速内部 RAM

通过将 RAME 位或者 RAMWE 位置为无效，即使进行上电复位也继续保持数据。
在 RAME 位和 RAMWE 位都有效的情况下，不能保持数据。

(b) 大容量内部 RAM（用于保持的内部 RAM 除外）

通过将 VRAME 或者 VRAMWE 位置为无效，即使进行上电复位也继续保持数据。
在 VRAME 位和 VRAMWE 位都有效的情况下，不能保持数据。

(c) 用于保持的内部 RAM

通过将 VRAME 位、VRAMWE 位或者 RRAMWE 位置为无效，即使进行上电复位也继续保持数据。但是，通过上电复位解除深度待机模式的情况除外。

在 VRAME 位、VRAMWE 位和 RRAMWE 位都有效的情况下，不能保持数据。

(2) 深度待机模式

(a) 高速内部 RAM、大容量内部 RAM（用于保持的内部 RAM 除外）

不能保持数据。

(b) 用于保持的内部 RAM

通过将 RRAMKP 位置为有效，即使在深度待机模式中也继续保持数据。

但是，在通过上电复位解除了深度待机模式的情况下，不能保持数据。

在通过中断或者用于解除的引脚变化解除了深度待机模式的情况下，虽然执行上电复位的异常处理，但是保持数据。

48. 通用输入 / 输出端口

本 LSI 有 A、B、C、D、E、F、G、H、J 共 9 组通用输入 / 输出端口。

各端口和外围模块的引脚多路复用。

通用输入 / 输出端口由选择多路复用引脚功能和输入 / 输出方向的寄存器、保存引脚数据的数据寄存器以及读引脚值的端口寄存器构成。

48.1 特点

- 通过设定控制寄存器来选择多路复用引脚的功能。
如果选择通用输入 / 输出功能或者多功能定时器脉冲单元 2 的 TIOC 输入 / 输出功能，就通过 IO 寄存器选择输入 / 输出方向。

表 48.1 通用输入 / 输出引脚个数

端口	SH7268	SH7269
A	2 个输入 / 输出引脚	
B	22 个输入 / 输出引脚	
C	9 个输入 / 输出引脚	
D	16 个输入 / 输出引脚	
E	4 个带漏极开路输出的输入引脚	8 个带漏极开路输出的输入引脚
F	23 个输入 / 输出引脚	
G	28 个输入 / 输出引脚	
H	6 个输入引脚	8 个输入引脚
J	—	32 个输入 / 输出引脚
总数	111 个 (101 个输入 / 输出引脚、 4 个带漏极开路输出的输入引脚和 6 个输入引脚)	149 个 (133 个输入 / 输出引脚、 8 个带漏极开路输出的输入引脚和 8 个输入引脚)

此 LSI 的多路复用引脚如表 48.2 ~ 表 48.10 所示。

阴影部分的寄存器和引脚功能只能用于 SH7269。

表 48.2 多路复用一览表 (端口 A)

端口	RES 引脚输入值	
	H	L
	功能 1	功能 2
A	PA1	MD_BOOT1
	PA0	MD_BOOT0

【注】在 $\overline{\text{RES}}$ 为“L”的状态下，端口 A 的功能 2 有效；在 $\overline{\text{RES}}$ 为“H”的状态下，总是为通用输入 / 输出功能。

表 48.3 多路复用一览表（端口 B）

设定的寄存器	模式位（PBnMD[2:0]）的设定值					
	000	001	010	011	100	110
	功能 1	功能 2	功能 3	功能 4	功能 5	功能 7
PBCR5	PB22	A22	CTx2	IETxD	$\overline{CS4}$	—
	PB21	A21	CRx2	IERxD	—	—
	PB20	A20	QMI_0/QIO1_0	MISO0	—	SPBMI_0/SPBIO1_0
PBCR4	PB19	A19	QMO_0/QIO0_0	MOSIO	—	SPBMO_0/SPBIO0_0
	PB18	A18	QSSL_0	SSL00	—	SPBSSL
	PB17	A17	QSPCLK_0	RSPCK0	—	SPBCLK
	PB16	A16	QIO3_0	—	—	SPBIO3_0
PBCR3	PB15	A15	QIO2_0	—	—	SPBIO2_0
	PB14	A14	QIO3_1	—	—	SPBIO3_1
	PB13	A13	QIO2_1	—	—	SPBIO2_1
	PB12	A12	TIOC3D	—	—	—
PBCR2	PB11	A11	TIOC3C	—	—	—
	PB10	A10	TIOC3B	—	—	—
	PB9	A9	TIOC3A	—	—	—
	PB8	A8	TIOC2B	—	—	—
PBCR1	PB7	A7	TIOC2A	—	—	—
	PB6	A6	TIOC1B	—	—	—
	PB5	A5	TIOC1A	—	—	—
	PB4	A4	TIOC0D	—	—	—
PBCR0	PB3	A3	TIOC0C	—	—	—
	PB2	A2	TIOC0B	—	—	—
	PB1	A1	TIOC0A	—	—	—

表 48.4 多路复用一览表（端口 C）

设定的寄存器	模式位（PCnMD[2:0]）的设定值					
	000	001	010	011	100	101
	功能 1	功能 2	功能 3	功能 4	功能 5	功能 6
PCCR2	PC8	$\overline{CS3}$	TxD7	CTx1	CTx0&CTx1	—
PCCR1	PC7	CKE	RxD7	CRx1	CRx0/CRx1	IRQ1
	PC6	\overline{CAS}	SCK7	CTx0	CTx0&CTx1&CTx2	—
	PC5	\overline{RAS}	—	CRx0	CRx0/CRx1/CRx2	IRQ0
	PC4	$\overline{WE1/DQMLU/WE}$	TxD6	—	—	—
PCCR0	PC3	$\overline{WE0/DQMLL}$	RxD6	—	—	—
	PC2	$\overline{RD/WR}$	SCK6	—	—	—
	PC1	\overline{RD}	—	—	—	—
	PC0	$\overline{CS0}$	MD_BOOT2	—	—	—

【注】在 \overline{RES} 为“L”的状态下，PC0 的功能 3 有效；在 \overline{RES} 为“H”的状态下，为功能 1 或者功能 2。

表 48.5 多路复用一览表（端口 D）

设定的寄存器	模式位 (PDnMD[1:0]) 的设定值		
	00	01	10
	功能 1	功能 2	功能 3
PDCR3	PD15	D15/NAF7	PWM2H
	PD14	D14/NAF6	PWM2G
	PD13	D13/NAF5	PWM2F
	PD12	D12/NAF4	PWM2E
PDCR2	PD11	D11/NAF3	PWM2D
	PD10	D10/NAF2	PWM2C
	PD9	D9/NAF1	PWM2B
	PD8	D8/NAF0	PWM2A
PDCR1	PD7	D7/ $\overline{\text{FWE}}$	PWM1H
	PD6	D6/FALE	PWM1G
	PD5	D5/FCLE	PWM1F
	PD4	D4/ $\overline{\text{FRE}}$	PWM1E
PDCR0	PD3	D3	PWM1D
	PD2	D2	PWM1C
	PD1	D1	PWM1B
	PD0	D0	PWM1A

【注】 功能 2 的总线状态控制器和 NAND 闪存控制器的功能进行自动切换。
（参照“10. 总线状态控制器”）。

表 48.6 多路复用一览表（端口 E）

设定的寄存器	模式位 (PEnMD[2:0]) 的设定值				
	000	001	010	011	100
	功能 1	功能 2	功能 3	功能 4	功能 5
PECR1	PE7	SDA3	RxD7	—	—
	PE6	SCL3	RxD6	—	—
	PE5	SDA2	RxD5	DV_HSYNC	—
	PE4	SCL2	RxD4	DV_VSYNC	—
PECR0	PE3	SDA1	TCLKD	$\overline{\text{ADTRG}}$	DV_HSYNC
	PE2	SCL1	TCLKC	$\overline{\text{IOIS16}}$	DV_VSYNC
	PE1	SDA0	TCLKB	AUDIO_CLK	DV_CLK
	PE0	SCL0	TCLKA	LCD_EXTCLK	—

表 48.7 多路复用一览表 (端口 F)

设定的寄存器	模式位 (PFnMD[2:0]) 的设定值						
	000	001	010	011	100	101	110
	功能 1	功能 2	功能 3	功能 4	功能 5	功能 6	功能 7
PFCR6	PF23	SD_D2_0	—	—	TxD3	MMC_D2	—
	PF22	SD_D3_0	—	—	RxD3	MMC_D3	—
	PF21	SD_CMD_0	—	—	SCK3	MMC_CMD	—
	PF20	SD_CLK_0	SSIDATA3	—	—	MMC_CLK	—
PFCR5	PF19	SD_D0_0	SSIWS3	—	IRQ7	MMC_D0	—
	PF18	SD_D1_0	SSISCK3	—	IRQ6	MMC_D1	—
	PF17	SD_WP_0	—	FRB	IRQ5	—	—
	PF16	SD_CD_0	—	$\overline{\text{FCE}}$	IRQ4	MMC_CD	—
PFCR4	PF15	A0	SSIDATA2	$\overline{\text{WDTOVF}}$	TxD2	$\overline{\text{UBCTR}}\overline{\text{G}}$	—
PFCR3	PF14	A25	SSIWS2	—	RxD2	—	—
	PF13	A24	SSISCK2	—	SCK2	—	—
	PF12	—	SSIDATA1	DV_DATA3	TxD1	MMC_D7	—
PFCR2	PF11	—	SSIWS1	DV_DATA2	RxD1	MMC_D6	—
	PF10	$\overline{\text{CS1}}$	SSISCK1	DV_DATA1	SCK1	MMC_D5	—
	PF9	$\overline{\text{BS}}$	—	DV_DATA0	SCK0	MMC_D4	$\overline{\text{RTS1}}$
	PF8	A23	—	—	TxD0	—	—
PFCR1	PF7	—	SSIRxD0	—	RxD0	SGOUT_3	$\overline{\text{CTS1}}$
	PF6	$\overline{\text{CE2A}}$	SSITxD0	—	—	SGOUT_2	—
	PF5	—	SSIWS0	—	—	SGOUT_1	—
	PF4	$\overline{\text{CS5/CE1A}}$	SSISCK0	—	—	SGOUT_0	—
PFCR0	PF3	$\overline{\text{CS2}}$	QMI_1/ QIO1_1	MISO1	TIOC4D	AUDIO_XOUT	SPBMI_1/ SPBIO1_1
	PF2	$\overline{\text{WAIT}}$	QMO_1/ QIO0_1	MOSI1	TIOC4C	TEND0	SPBMO_1/ SPBIO0_1
	PF1	$\overline{\text{BACK}}$	QSSL_1	SSL10	TIOC4B	DACK0	—
	PF0	$\overline{\text{BREQ}}$	QSPCLK_1	RSPCK1	TIOC4A	DREQ0	—

表 48.8 多路复用一览表（端口 G）

设定的寄存器	模式位 (PGnMD[2:0]) 的设定值				
	000	001	010	011	100
	功能 1	功能 2	功能 3	功能 4	功能 5
PGCR6	PG27	—	LCD_TCON2	LCD_EXTCLK	—
	PG26	—	LCD_TCON1	—	—
	PG25	—	LCD_TCON0	—	—
	PG24	—	LCD_CLK	—	—
PGCR5	PG23	—	LCD_DATA23	LCD_TCON6	TxD5
	PG22	—	LCD_DATA22	LCD_TCON5	RxD5
	PG21	DV_DATA7	LCD_DATA21	LCD_TCON4	TxD4
	PG20	DV_DATA6	LCD_DATA20	LCD_TCON3	RxD4
PGCR4	PG19	DV_DATA5	LCD_DATA19	SPDIF_OUT	SCK5
	PG18	DV_DATA4	LCD_DATA18	SPDIF_IN	SCK4
	PG17	$\overline{WE3/ICIOR}/\overline{AH}/\overline{DQM}U$	LCD_DATA17	—	—
	PG16	$\overline{WE2/ICIOR}/\overline{DQM}U$	LCD_DATA16	—	—
PGCR3	PG15	D31	LCD_DATA15	PINT7	—
	PG14	D30	LCD_DATA14	PINT6	—
	PG13	D29	LCD_DATA13	PINT5	—
	PG12	D28	LCD_DATA12	PINT4	—
PGCR2	PG11	D27	LCD_DATA11	PINT3	TIOC3D
	PG10	D26	LCD_DATA10	PINT2	TIOC3C
	PG9	D25	LCD_DATA9	PINT1	TIOC3B
	PG8	D24	LCD_DATA8	PINT0	TIOC3A
PGCR1	PG7	D23	LCD_DATA7	IRQ7	TIOC2B
	PG6	D22	LCD_DATA6	IRQ6	TIOC2A
	PG5	D21	LCD_DATA5	IRQ5	TIOC1B
	PG4	D20	LCD_DATA4	IRQ4	TIOC1A
PGCR0	PG3	D19	LCD_DATA3	IRQ3	TIOC0D
	PG2	D18	LCD_DATA2	IRQ2	TIOC0C
	PG1	D17	LCD_DATA1	IRQ1	TIOC0B
	PG0	D16	LCD_DATA0	IRQ0	TIOC0A

表 48.9 多路复用一览表（端口 H）

设定的寄存器	模式位（PHnMD[1:0]）的设定值			
	00	01	10	11
	功能 1	功能 2	功能 3	功能 4
PHCR1	PH7	AN7	PINT7	—
	PH6	AN6	PINT6	—
	PH5	AN5	PINT5	LCD_EXTCLK
	PH4	AN4	PINT4	—
PHCR0	PH3	AN3	PINT3	—
	PH2	AN2	PINT2	—
	PH1	AN1	PINT1	—
	PH0	AN0	PINT0	—

表 48.10 多路复用一览表（端口 J：只限于 SH7269）

设定的寄存器	模式位 (PJnMD[2:0]) 的设定值						
	000	001	010	011	100	101	110
	功能 1	功能 2	功能 3	功能 4	功能 5	功能 6	功能 7
PJCR7	PJ31	DV_CLK	—	—	—	—	—
	PJ30	—	SSIDATA5	—	TIOC2B	IETxD	—
	PJ29	—	SSIWS5	—	TIOC2A	IERxD	—
	PJ28	—	SSISCK5	—	TIOC1B	$\overline{\text{RTS}}_7$	—
PJCR6	PJ27	SGOUT_3	—	—	TIOC1A	$\overline{\text{CTS}}_7$	—
	PJ26	SGOUT_2	SSIDATA4	LCD_TCON5	—	TxD7	—
	PJ25	SGOUT_1	SSIWS4	LCD_TCON4	SPDIF_OUT	RxD7	—
	PJ24	SGOUT_0	SSISCK4	LCD_TCON3	SPDIF_IN	SCK7	—
PJCR5	PJ23	DV_DATA23	LCD_DATA23	LCD_TCON6	IRQ3	CTx1	CTx0&CTx1
	PJ22	DV_DATA22	LCD_DATA22	LCD_TCON5	IRQ2	CRx1	CRx0/CRx1
	PJ21	DV_DATA21	LCD_DATA21	LCD_TCON4	IRQ1	CTx2	CTx0&CTx1 &CTx2
	PJ20	DV_DATA20	LCD_DATA20	LCD_TCON3	IRQ0	CRx2	CRx0/CRx1/ CRx2
PJCR4	PJ19	DV_DATA19	LCD_DATA19	MISO0	TIOC0D	SIOFRxD	AUDIO_XOUT
	PJ18	DV_DATA18	LCD_DATA18	MOSI0	TIOC0C	SIOFTxD	—
	PJ17	DV_DATA17	LCD_DATA17	SSL00	TIOC0B	SIOFSYNC	—
	PJ16	DV_DATA16	LCD_DATA16	RSPCK0	TIOC0A	SIOFCK	—
PJCR3	PJ15	DV_DATA15	LCD_DATA15	PINT7	PWM2H	TxD7	—
	PJ14	DV_DATA14	LCD_DATA14	PINT6	PWM2G	TxD6	—
	PJ13	DV_DATA13	LCD_DATA13	PINT5	PWM2F	TxD5	—
	PJ12	DV_DATA12	LCD_DATA12	PINT4	PWM2E	SCK7	—
PJCR2	PJ11	DV_DATA11	LCD_DATA11	PINT3	PWM2D	SCK6	—
	PJ10	DV_DATA10	LCD_DATA10	PINT2	PWM2C	SCK5	—
	PJ9	DV_DATA9	LCD_DATA9	PINT1	PWM2B	$\overline{\text{RTS}}_5$	—
	PJ8	DV_DATA8	LCD_DATA8	PINT0	PWM2A	$\overline{\text{CTS}}_5$	—
PJCR1	PJ7	DV_DATA7	LCD_DATA7	SD_D2_1	PWM1H	—	—
	PJ6	DV_DATA6	LCD_DATA6	SD_D3_1	PWM1G	—	—
	PJ5	DV_DATA5	LCD_DATA5	SD_CMD_1	PWM1F	—	—
	PJ4	DV_DATA4	LCD_DATA4	SD_CLK_1	PWM1E	—	—
PJCR0	PJ3	DV_DATA3	LCD_DATA3	SD_D0_1	PWM1D	—	—
	PJ2	DV_DATA2	LCD_DATA2	SD_D1_1	PWM1C	—	—
	PJ1	DV_DATA1	LCD_DATA1	SD_WP_1	PWM1B	—	—
	PJ0	DV_DATA0	LCD_DATA0	SD_CD_1	PWM1A	—	—

48.2 寄存器说明

寄存器结构如表 48.11 所示。

表 48.11 寄存器结构

端口	寄存器名	略称	R/W	初始值	地址	存取长度
A	端口 A 的 IO 寄存器 0	PAIOR0	R/W	H'0000	H'FFFE3812	8、16*2
	端口 A 的数据寄存器 0	PADR0	R/W	H'0000	H'FFFE3816	8、16*2
	端口 A 的端口寄存器 0	PAPR0	R	H'xxxx	H'FFFE381A	8、16
B	端口 B 的控制寄存器 5	PBCR5	R/W	H'0000/H'0001*1	H'FFFE3824	8、16、32
	端口 B 的控制寄存器 4	PBCR4	R/W	H'0000/H'1111*1	H'FFFE3826	8、16
	端口 B 的控制寄存器 3	PBCR3	R/W	H'0000/H'1111*1	H'FFFE3828	8、16、32
	端口 B 的控制寄存器 2	PBCR2	R/W	H'0000/H'1111*1	H'FFFE382A	8、16
	端口 B 的控制寄存器 1	PBCR1	R/W	H'0000/H'1111*1	H'FFFE382C	8、16、32
	端口 B 的控制寄存器 0	PBCR0	R/W	H'0000/H'1110/H'1100*1	H'FFFE382E	8、16
	端口 B 的 IO 寄存器 1	PBIOR1	R/W	H'0000	H'FFFE3830	8、16、32
	端口 B 的 IO 寄存器 0	PBIOR0	R/W	H'0000	H'FFFE3832	8、16
	端口 B 的数据寄存器 1	PBDR1	R/W	H'0000	H'FFFE3834	8、16、32
	端口 B 的数据寄存器 0	PBDR0	R/W	H'0000	H'FFFE3836	8、16
	端口 B 的端口寄存器 1	PBPR1	R	H'xxxx	H'FFFE3838	8、16、32
	端口 B 的端口寄存器 0	PBPR0	R	H'xxxx	H'FFFE383A	8、16
	C	端口 C 的控制寄存器 2	PCCR2	R/W	H'0000	H'FFFE384A
端口 C 的控制寄存器 1		PCCR1	R/W	H'0000	H'FFFE384C	8、16、32
端口 C 的控制寄存器 0		PCCR0	R/W	H'0000/H'0011*1	H'FFFE384E	8、16
端口 C 的 IO 寄存器 0		PCIOR0	R/W	H'0000	H'FFFE3852	8、16
端口 C 的数据寄存器 0		PCDR0	R/W	H'0000	H'FFFE3856	8、16
端口 C 的端口寄存器 0		PCPR0	R	H'xxxx	H'FFFE385A	8、16
D	端口 D 的控制寄存器 3	PDCR3	R/W	H'0000/H'1111*1	H'FFFE3868	8、16、32
	端口 D 的控制寄存器 2	PDCR2	R/W	H'0000/H'1111*1	H'FFFE386A	8、16
	端口 D 的控制寄存器 1	PDCR1	R/W	H'0000/H'1111*1	H'FFFE386C	8、16、32
	端口 D 的控制寄存器 0	PDCR0	R/W	H'0000/H'1111*1	H'FFFE386E	8、16
	端口 D 的 IO 寄存器 0	PDIOR0	R/W	H'0000	H'FFFE3872	8、16
	端口 D 的数据寄存器 0	PDDR0	R/W	H'0000	H'FFFE3876	8、16
	端口 D 的端口寄存器 0	PDPR0	R	H'xxxx	H'FFFE387A	8、16
E	端口 E 的控制寄存器 1	PECR1	R/W	H'0000	H'FFFE388C	8、16、32
	端口 E 的控制寄存器 0	PECR0	R/W	H'0000	H'FFFE388E	8、16
	端口 E 的 IO 寄存器 0	PEIOR0	R/W	H'0000	H'FFFE3892	8、16
	端口 E 的数据寄存器 0	PEDR0	R/W	H'0000	H'FFFE3896	8、16
	端口 E 的端口寄存器 0	PEPR0	R	H'xxxx	H'FFFE389A	8、16
F	端口 F 的控制寄存器 6	PFCR6	R/W	H'0000	H'FFFE38A2	8、16
	端口 F 的控制寄存器 5	PFCR5	R/W	H'0000	H'FFFE38A4	8、16、32
	端口 F 的控制寄存器 4	PFCR4	R/W	H'0000	H'FFFE38A6	8*3、16
	端口 F 的控制寄存器 3	PFCR3	R/W	H'0000	H'FFFE38A8	8、16、32
	端口 F 的控制寄存器 2	PFCR2	R/W	H'0000	H'FFFE38AA	8、16

端口	寄存器名	略称	R/W	初始值	地址	存取长度
F	端口 F 的控制寄存器 1	PFCR1	R/W	H'0000	H'FFFE38AC	8、16、32
	端口 F 的控制寄存器 0	PFCR0	R/W	H'0000	H'FFFE38AE	8、16
	端口 F 的 IO 寄存器 1	PFIOR1	R/W	H'0000	H'FFFE38B0	8、16、32
	端口 F 的 IO 寄存器 0	PFIOR0	R/W	H'0000	H'FFFE38B2	8、16
	端口 F 的数据寄存器 1	PFDR1	R/W	H'0000	H'FFFE38B4	8、16、32
	端口 F 的数据寄存器 0	PFDR0	R/W	H'0000	H'FFFE38B6	8、16
	端口 F 的端口寄存器 1	PFPR1	R	H'xxxx	H'FFFE38B8	8、16、32
	端口 F 的端口寄存器 0	PFPR0	R	H'xxxx	H'FFFE38BA	8、16
G	端口 G 的控制寄存器 6	PGCR6	R/W	H'0000	H'FFFE38C2	8、16
	端口 G 的控制寄存器 5	PGCR5	R/W	H'0000	H'FFFE38C4	8、16、32
	端口 G 的控制寄存器 4	PGCR4	R/W	H'0000	H'FFFE38C6	8、16
	端口 G 的控制寄存器 3	PGCR3	R/W	H'0000/H'1111*1	H'FFFE38C8	8、16、32
	端口 G 的控制寄存器 2	PGCR2	R/W	H'0000/H'1111*1	H'FFFE38CA	8、16
	端口 G 的控制寄存器 1	PGCR1	R/W	H'0000/H'1111*1	H'FFFE38CC	8、16、32
	端口 G 的控制寄存器 0	PGCR0	R/W	H'0000/H'1111*1	H'FFFE38CE	8、16
	端口 G 的 IO 寄存器 1	PGIOR1	R/W	H'0000	H'FFFE38D0	8、16、32
	端口 G 的 IO 寄存器 0	PGIOR0	R/W	H'0000	H'FFFE38D2	8、16
	端口 G 的数据寄存器 1	PGDR1	R/W	H'0000	H'FFFE38D4	8、16、32
	端口 G 的数据寄存器 0	PGDR0	R/W	H'0000	H'FFFE38D6	8、16
	端口 G 的端口寄存器 1	PGPR1	R	H'xxxx	H'FFFE38D8	8、16、32
	端口 G 的端口寄存器 0	PGPR0	R	H'xxxx	H'FFFE38DA	8、16
	H	端口 H 的控制寄存器 1	PHCR1	R/W	H'0000	H'FFFE38EC
端口 H 的控制寄存器 0		PHCR0	R/W	H'0000	H'FFFE38EE	8、16
端口 H 的端口寄存器 0		PHPR0	R	H'xxxx	H'FFFE38FA	8、16
J	端口 J 的控制寄存器 7	PJCR7	R/W	H'0000	H'FFFE3900	8、16、32
	端口 J 的控制寄存器 6	PJCR6	R/W	H'0000	H'FFFE3902	8、16
	端口 J 的控制寄存器 5	PJCR5	R/W	H'0000	H'FFFE3904	8、16、32
	端口 J 的控制寄存器 4	PJCR4	R/W	H'0000	H'FFFE3906	8、16
	端口 J 的控制寄存器 3	PJCR3	R/W	H'0000	H'FFFE3908	8、16、32
	端口 J 的控制寄存器 2	PJCR2	R/W	H'0000	H'FFFE390A	8、16
	端口 J 的控制寄存器 1	PJCR1	R/W	H'0000	H'FFFE390C	8、16、32
	端口 J 的控制寄存器 0	PJCR0	R/W	H'0000	H'FFFE390E	8、16
	端口 J 的 IO 寄存器 1	PJIOR1	R/W	H'0000	H'FFFE3910	8、16、32
	端口 J 的 IO 寄存器 0	PJIOR0	R/W	H'0000	H'FFFE3912	8、16
	端口 J 的数据寄存器 1	PJDR1	R/W	H'0000	H'FFFE3914	8、16、32
	端口 J 的数据寄存器 0	PJDR0	R/W	H'0000	H'FFFE3916	8、16
	端口 J 的端口寄存器 1	PJPR1	R	H'xxxx	H'FFFE391C	8、16、32
	端口 J 的端口寄存器 0	PJPR0	R	H'xxxx	H'FFFE391A	8、16
—	串行音频接口 噪声消除器的控制寄存器	SNCR	R/W	H'0000	H'FFFE393E	8、16

【注】 *1 初始值因 LSI 的引导模式而不同。

*2 当进行 16 位存取时，能读而不能写。

*3 当进行 8 位存取时，能读而不能写。

48.2.1 端口 A 的 IO 寄存器 0 (PAIOR0)

PAIOR0 是 16 位可读写寄存器，选择端口 A 引脚的输入 / 输出方向。PA1IOR 位和 PA0IOR 位分别对应 PA1 引脚和 PA0 引脚。如果将 PAIOR0 的位置 “1”，对应的引脚就为输出；如果置 “0” 就为输入。

PAIOR0 的 bit15 ~ 9 和 bit7 ~ 1 是保留位，读写值都为 “0”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PA1 IOR	—	—	—	—	—	—	—	PA0 IOR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

48.2.2 端口 A 的数据寄存器 0 (PADR0)

PADR0 是 16 位可读写寄存器，保存端口 A 的数据。PA1DR 位和 PA0DR 位分别对应 PA1 引脚和 PA0 引脚。

在引脚功能为通用输出时，如果给 PADR0 写值，就从引脚输出此值；如果读 PADR0，就直接读寄存器的值，与引脚的状态无关。

在引脚功能为通用输入时，如果读 PADR0，就直接读引脚的状态而不读寄存器的值；如果给 PADR0 写值，就能将此值写到 PADR0，但是不影响引脚的状态。PADR0 的读写如表 48.12 所示。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PA1 DR	—	—	—	—	—	—	—	PA0 DR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
15 ~ 9	—	全 0	R	保留位 读写值都为 “0”。
8	PA1DR	0	R/W	参照表 48.12。
7 ~ 1	—	全 0	R	保留位 读写值都为 “0”。
0	PA0DR	0	R/W	参照表 48.12。

表 48.12 端口 A 的数据寄存器 0 (PADR0) 的读写

- PADR0 的 bit8 和 bit0

PAIOR0	引脚功能	读	写
0	通用输入	引脚的状态	能写 PADR0，但是不影响引脚的状态。
1	通用输出	PADR0 的值	从引脚输出写入值。

48.2.3 端口 A 的端口寄存器 0 (PAPR0)

PAPR0 是 16 位只读寄存器，PA1PR 位和 PA0PR 位分别对应 PA1 引脚和 PA0 引脚。PAPR0 能随时读引脚的值。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PA1 PR	PA0 PR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	PA1	PA0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 2	—	全 0	R	保留位 读写值都为“0”。
1	PA1PR	引脚的状态	R	如果读此位，就读引脚的状态。写操作无效。
0	PA0PR	引脚的状态	R	

48.2.4 端口 B 的控制寄存器 0 ~ 5 (PBCR0 ~ PBCR5)

PBCR0 ~ PBCR5 分别是 16 位可读写寄存器，选择端口 B 的多路复用引脚的功能。

(1) 端口 B 的控制寄存器 5 (PBCR5)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PB22MD[2:0]			—	—	PB21MD[1:0]		—	PB20MD[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0/1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 8	PB22MD[2:0]	000	R/W	PB22 模式 控制 PB22 引脚的功能。 000: PB22 001: A22 010: CTx2 011: IETxD 100: $\overline{\text{CS4}}$ 101: 禁止设定 110: 禁止设定 111: 禁止设定
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	PB21MD[1:0]	00	R/W	PB21 模式 控制 PB21 引脚的功能。 00: PB21 01: A21 10: CRx2 11: IERxD
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PB20MD[2:0]	000/001	R/W	PB20 模式 控制 PB20 引脚的功能。 • 引导模式 0、1 000: 禁止设定 001: A20 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 • 引导模式 2 ~ 5 000: PB20 (初始值) 001: A20 010: QMISO0/QIO10 011: MISO0 100: 禁止设定 101: 禁止设定 110: SPBMI_0/SPBIO1_0 111: 禁止设定

(2) 端口 B 的控制寄存器 4 (PBCR4)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PB19MD[2:0]			—	PB18MD[2:0]			—	PB17MD[2:0]			—	PB16MD[2:0]		
初始值:	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PB19MD[2:0]	000/001	R/W	PB19 模式 控制 PB19 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 000: 禁止设定 001: A19 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 000: PB19 (初始值) 001: A19 010: QMO_0/QIO0_0 011: MOSIO 100: 禁止设定 101: 禁止设定 110: SPBMO_0/SPBIO0_0 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PB18MD[2:0]	000/001	R/W	PB18 模式 控制 PB18 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 000: 禁止设定 001: A18 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 000: PB18 (初始值) 001: A18 010: QSSL_0 011: SSL00 100: 禁止设定 101: 禁止设定 110: SPBSSL 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PB17MD[2:0]	000/001	R/W	PB17 模式 控制 PB17 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 000: 禁止设定 001: A17 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 000: PB17 (初始值) 001: A17 010: QSPCLK_0 011: RSPCK0 100: 禁止设定 101: 禁止设定 110: SPBCLK 111: 禁止设定

位	位名	初始值	R/W	说明		
3	—	0	R	保留位 读写值都为“0”。		
2 ~ 0	PB16MD[2:0]	000/001	R/W	PB16 模式 控制 PB16 引脚的功能。 <table style="width: 100%; border: none;"> <tr> <td style="width: 50%; vertical-align: top;"> <ul style="list-style-type: none"> • 引导模式 0、1 000: 禁止设定 001: A16 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 </td> <td style="width: 50%; vertical-align: top;"> <ul style="list-style-type: none"> • 引导模式 2 ~ 5 000: PB16 (初始值) 001: A16 010: QIO3_0 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: SPBIO3_0 111: 禁止设定 </td> </tr> </table>	<ul style="list-style-type: none"> • 引导模式 0、1 000: 禁止设定 001: A16 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 	<ul style="list-style-type: none"> • 引导模式 2 ~ 5 000: PB16 (初始值) 001: A16 010: QIO3_0 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: SPBIO3_0 111: 禁止设定
<ul style="list-style-type: none"> • 引导模式 0、1 000: 禁止设定 001: A16 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 	<ul style="list-style-type: none"> • 引导模式 2 ~ 5 000: PB16 (初始值) 001: A16 010: QIO3_0 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: SPBIO3_0 111: 禁止设定 					

(3) 端口 B 的控制寄存器 3 (PBCR3)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PB15MD[2:0]			—	PB14MD[2:0]			—	PB13MD[2:0]			—	—	PB12MD[1:0]	
初始值:	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明		
15	—	0	R	保留位 读写值都为“0”。		
14 ~ 12	PB15MD[2:0]	000/001	R/W	PB15 模式 控制 PB15 引脚的功能。 <table style="width: 100%; border: none;"> <tr> <td style="width: 50%; vertical-align: top;"> <ul style="list-style-type: none"> • 引导模式 0、1 000: 禁止设定 001: A15 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 </td> <td style="width: 50%; vertical-align: top;"> <ul style="list-style-type: none"> • 引导模式 2 ~ 5 000: PB15 (初始值) 001: A15 010: QIO2_0 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: SPBIO2_0 111: 禁止设定 </td> </tr> </table>	<ul style="list-style-type: none"> • 引导模式 0、1 000: 禁止设定 001: A15 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 	<ul style="list-style-type: none"> • 引导模式 2 ~ 5 000: PB15 (初始值) 001: A15 010: QIO2_0 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: SPBIO2_0 111: 禁止设定
<ul style="list-style-type: none"> • 引导模式 0、1 000: 禁止设定 001: A15 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 	<ul style="list-style-type: none"> • 引导模式 2 ~ 5 000: PB15 (初始值) 001: A15 010: QIO2_0 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: SPBIO2_0 111: 禁止设定 					
11	—	0	R	保留位 读写值都为“0”。		

位	位名	初始值	R/W	说明
10 ~ 8	PB14MD[2:0]	000/001	R/W	PB14 模式 控制 PB14 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 000: 禁止设定 001: A14 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 000: PB14 (初始值) 001: A14 010: QIO3_1 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: SPBIO3_1 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PB13MD[2:0]	000/001	R/W	PB13 模式 控制 PB13 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 000: 禁止设定 001: A13 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 000: PB13 (初始值) 001: A13 010: QIO2_1 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: SPBIO2_1 111: 禁止设定
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	PB12MD[1:0]	00/01	R/W	PB12 模式 控制 PB12 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: A12 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PB12 (初始值) 01: A12 10: TIOC3D 11: 禁止设定

(4) 端口 B 的控制寄存器 2 (PBCR2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PB11MD[1:0]	—	—	PB10MD[1:0]	—	—	PB9MD[1:0]	—	—	PB8MD[1:0]	—	—	—	—
初始值:	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13、12	PB11MD[1:0]	00/01	R/W	PB11 模式 控制 PB11 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: A11 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PB11 (初始值) 01: A11 10: TIOC3C 11: 禁止设定
11、10	—	全 0	R	保留位 读写值都为“0”。
9、8	PB10MD[1:0]	00/01	R/W	PB10 模式 控制 PB10 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: A10 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PB10 (初始值) 01: A10 10: TIOC3B 11: 禁止设定
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	PB9MD[1:0]	00/01	R/W	PB9 模式 控制 PB9 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: A9 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PB9 (初始值) 01: A9 10: TIOC3A 11: 禁止设定
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	PB8MD[1:0]	00/01	R/W	PB8 模式 控制 PB8 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: A8 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PB8 (初始值) 01: A8 10: TIOC2B 11: 禁止设定

(5) 端口 B 的控制寄存器 1 (PBCR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PB7MD[1:0]	—	—	PB6MD[1:0]	—	—	PB5MD[1:0]	—	—	PB4MD[1:0]	—	—	—	—
初始值:	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13、12	PB7MD[1:0]	00/01	R/W	PB7 模式 控制 PB7 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: A7 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PB7 (初始值) 01: A7 10: TIOC2A 11: 禁止设定
11、10	—	全 0	R	保留位 读写值都为“0”。
9、8	PB6MD[1:0]	00/01	R/W	PB6 模式 控制 PB6 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: A6 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PB6 (初始值) 01: A6 10: TIOC1B 11: 禁止设定
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	PB5MD[1:0]	00/01	R/W	PB5 模式 控制 PB5 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: A5 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PB5 (初始值) 01: A5 10: TIOC1A 11: 禁止设定
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	PB4MD[1:0]	00/01	R/W	PB4 模式 控制 PB4 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: A4 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PB4 (初始值) 01: A4 10: TIOC0D 11: 禁止设定

(6) 端口 B 的控制寄存器 0 (PBCR0)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PB3MD[1:0]	—	—	PB2MD[1:0]	—	—	PB1MD[1:0]	—	—	—	—	—	—	—
初始值:	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R	R

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13、12	PB3MD[1:0]	00/01	R/W	PB3 模式 控制 PB3 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: A3 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PB3 (初始值) 01: A3 10: TIOC0C 11: 禁止设定
11、10	—	全 0	R	保留位 读写值都为“0”。
9、8	PB2MD[1:0]	00/01	R/W	PB2 模式 控制 PB2 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: A2 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PB2 (初始值) 01: A2 10: TIOC0B 11: 禁止设定
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	PB1MD[1:0]	00/01	R/W	PB1 模式 控制 PB1 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0 00: 禁止设定 01: A1 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 1 ~ 5 00: PB1 (初始值) 01: A1 10: TIOC0A 11: 禁止设定
3 ~ 0	—	全 0	R	保留位 读写值都为“0”。

48.2.5 端口 B 的 IO 寄存器 0、1 (PBIOR0、PBIOR1)

PBIOR1 和 PBIOR0 是 16 位可读写寄存器，选择端口 B 引脚的输入 / 输出方向。PB22IOR ~ PB1IOR 位分别对应 PB22 ~ PB1 引脚。在端口 B 的引脚功能为通用输入 / 输出 (PB22 ~ PB1) 或者多功能定时器脉冲单元 2 的 TIOC 输入 / 输出功能时，PBIOR1 和 PBIOR0 有效，否则无效。如果将 PBIOR1 和 PBIOR0 的位置“1”，对应的引脚就为输出；如果置“0”就为输入。

PBIOR1 的 bit15 ~ 7 和 PBIOR0 的 bit0 是保留位，读写值都为“0”。

(1) 端口 B 的 IO 寄存器 1 (PBIOR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	PB22 IOR	PB21 IOR	PB20 IOR	PB19 IOR	PB18 IOR	PB17 IOR	PB16 IOR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(2) 端口 B 的 IO 寄存器 0 (PBIOR0)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PB15 IOR	PB14 IOR	PB13 IOR	PB12 IOR	PB11 IOR	PB10 IOR	PB9 IOR	PB8 IOR	PB7 IOR	PB6 IOR	PB5 IOR	PB4 IOR	PB3 IOR	PB2 IOR	PB1 IOR	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

48.2.6 端口 B 的数据寄存器 0、1 (PBDR0、PBDR1)

PBDR1 和 PBDR0 是 16 位可读写寄存器，保存端口 B 的数据。PB22DR ~ PB1DR 位分别对应 PB22 ~ PB1 引脚。

在引脚功能为通用输出时，如果给 PBDR1 或者 PBDR0 写值，就从引脚输出此值；如果读 PBDR1 或者 PBDR0，就直接读寄存器的值，与引脚的状态无关。

在引脚功能为通用输入时，如果读 PBDR1 或者 PBDR0，就直接读引脚的状态而不读寄存器的值；如果给 PBDR1 或者 PBDR0 写值，就能将此值写到 PBDR1 或者 PBDR0，但是不影响引脚的状态。PBDR1 和 PBDR0 的读写如表 48.13 所示。

(1) 端口 B 的数据寄存器 1 (PBDR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	PB22 DR	PB21 DR	PB20 DR	PB19 DR	PB18 DR	PB17 DR	PB16 DR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 7	—	全 0	R	保留位 读写值都为“0”。
6	PB22DR	0	R/W	参照表 48.13。
5	PB21DR	0	R/W	
4	PB20DR	0	R/W	
3	PB19DR	0	R/W	
2	PB18DR	0	R/W	
1	PB17DR	0	R/W	
0	PB16DR	0	R/W	

(2) 端口 B 的数据寄存器 0 (PBDR0)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PB15 DR	PB14 DR	PB13 DR	PB12 DR	PB11 DR	PB10 DR	PB9 DR	PB8 DR	PB7 DR	PB6 DR	PB5 DR	PB4 DR	PB3 DR	PB2 DR	PB1 DR	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

位	位名	初始值	R/W	说明
15	PB15DR	0	R/W	参照表 48.13。
14	PB14DR	0	R/W	
13	PB13DR	0	R/W	
12	PB12DR	0	R/W	
11	PB11DR	0	R/W	
10	PB10DR	0	R/W	
9	PB9DR	0	R/W	
8	PB8DR	0	R/W	
7	PB7DR	0	R/W	
6	PB6DR	0	R/W	
5	PB5DR	0	R/W	
4	PB4DR	0	R/W	
3	PB3DR	0	R/W	
2	PB2DR	0	R/W	
1	PB1DR	0	R/W	
0	—	0	R	保留位 读写值都为“0”。

表 48.13 端口 B 的数据寄存器 1、0 (PBDR1、PBDR0) 的读写

- PBDR1 的 bit6 ~ 0 和 PBDR0 的 bit15 ~ 1

PBIOR1、0	引脚功能	读	写
0	通用输入	引脚的状态	能写 PBDR0 和 PBDR1，但是不影响引脚的状态。
	通用输入除外	引脚的状态	能写 PBDR0 和 PBDR1，但是不影响引脚的状态。
1	通用输出	PBDR0 和 PBDR1 的值	从引脚输出写入值。
	通用输出除外	PBDR0 和 PBDR1 的值	能写 PBDR0 和 PBDR1，但是不影响引脚的状态。

48.2.7 端口 B 的端口寄存器 0、1 (PBPR0、PBPR1)

PBPR1 和 PBPR0 分别是 16 位只读寄存器，PB22PR ~ PB1PR 位分别对应 PB22 ~ PB1 引脚。与 PBCR5 ~ PBCR0 的设定无关，PBPR1 和 PBPR0 能随时读引脚的值。

(1) 端口 B 的端口寄存器 1 (PBPR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	PB22 PR	PB21 PR	PB20 PR	PB19 PR	PB18 PR	PB17 PR	PB16 PR
初始值:	0	0	0	0	0	0	0	0	0	PB22	PB21	PB20	PB19	PB18	PB17	PB16
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 7	—	全 0	R	保留位 读写值都为“0”。
6	PB22PR	引脚的状态	R	如果读此位，就读引脚的状态。写操作无效。
5	PB21PR	引脚的状态	R	
4	PB20PR	引脚的状态	R	
3	PB19PR	引脚的状态	R	
2	PB18PR	引脚的状态	R	
1	PB17PR	引脚的状态	R	
0	PB16PR	引脚的状态	R	

(2) 端口 B 的端口寄存器 0 (PBPR0)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PB15 PR	PB14 PR	PB13 PR	PB12 PR	PB11 PR	PB10 PR	PB9 PR	PB8 PR	PB7 PR	PB6 PR	PB5 PR	PB4 PR	PB3 PR	PB2 PR	PB1 PR	—
初始值:	PB15	PB14	PB13	PB12	PB11	PB10	PB9	PB8	PB7	PB6	PB5	PB4	PB3	PB2	PB1	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15	PB15PR	引脚的状态	R	如果读此位，就读引脚的状态。写操作无效。
14	PB14PR	引脚的状态	R	
13	PB13PR	引脚的状态	R	
12	PB12PR	引脚的状态	R	
11	PB11PR	引脚的状态	R	
10	PB10PR	引脚的状态	R	
9	PB9PR	引脚的状态	R	
8	PB8PR	引脚的状态	R	
7	PB7PR	引脚的状态	R	
6	PB6PR	引脚的状态	R	
5	PB5PR	引脚的状态	R	
4	PB4PR	引脚的状态	R	
3	PB3PR	引脚的状态	R	
2	PB2PR	引脚的状态	R	
1	PB1PR	引脚的状态	R	
0	—	0	R	保留位 读写值都为“0”。

48.2.8 端口 C 的控制寄存器 0 ~ 2 (PCCR0 ~ PCCR2)

PCCR0 ~ PCCR2 分别是 16 位可读写寄存器，选择端口 C 的多路复用引脚的功能。

(1) 端口 C 的控制寄存器 2 (PCCR2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	PC8MD[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 3	—	全 0	R	保留位 读写值都为“0”。
2 ~ 0	PC8MD[2:0]	000	R/W	PC8 模式 控制 PC8 引脚的功能。 000: $\overline{PC8}$ 100: CTx0&CTx1 001: $\overline{CS3}$ 101: 禁止设定 010: Tx D7 110: 禁止设定 011: CTx1 111: 禁止设定

(2) 端口 C 的控制寄存器 1 (PCCR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PC7MD[2:0]			—	PC6MD[2:0]			—	PC5MD[2:0]			—	—	PC4MD[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PC7MD[2:0]	000	R/W	PC7 模式 控制 PC7 引脚的功能。 000: PC7 001: CKE 010: RxD7 011: CRx1 100: CRx0/CRx1 101: IRQ1 110: 禁止设定 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PC6MD[2:0]	000	R/W	PC6 模式 控制 PC6 引脚的功能。 000: PC6 001: $\overline{\text{CAS}}$ 010: SCK7 011: CTx0 100: CTx0&CTx1&CTx2 101: 禁止设定 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PC5MD[2:0]	000	R/W	PC5 模式 控制 PC5 引脚的功能。 000: PC5 001: RAS 010: 禁止设定 011: CRx0 100: CRx0/CRx1/CRx2 101: IRQ0 110: 禁止设定 111: 禁止设定
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	PC4MD[1:0]	00	R/W	PC4 模式 控制 PC4 引脚的功能。 00: PC4 01: $\overline{\text{WE1/DQMLU}}/\overline{\text{WE}}$ 10: TxD6 11: 禁止设定

(3) 端口 C 的控制寄存器 0 (PCCR0)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PC3MD[1:0]	—	—	PC2MD[1:0]	—	—	—	PC1MD	—	—	—	PC0MD		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0/1	0	0	0	0/1
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R/W

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13、12	PC3MD[1:0]	00	R/W	PC3 模式 控制 PC3 引脚的功能。 00: $\overline{\text{PC3}}$ 10: RxD6 01: $\overline{\text{WE0/DQMLL}}$ 11: 禁止设定
11、10	—	全 0	R	保留位 读写值都为“0”。
9、8	PC2MD[1:0]	00	R/W	PC2 模式 控制 PC2 引脚的功能。 00: $\overline{\text{PC2}}$ 10: SCK5 01: $\overline{\text{RD/WR}}$ 11: 禁止设定
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	PC1MD	0/1	R/W	PC1 模式 控制 PC1 引脚的功能。 • 引导模式 0、1 0: 禁止设定 1: $\overline{\text{RD}}$ (初始值) • 引导模式 2 ~ 5 0: $\overline{\text{PC1}}$ (初始值) 1: $\overline{\text{RD}}$
3 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	PC0MD	0/1	R/W	PC0 模式 控制 PC0 引脚的功能。 • 引导模式 0、1 0: 禁止设定 1: $\overline{\text{CS0}}$ (初始值) • 引导模式 2 ~ 5 0: $\overline{\text{PC0}}$ (初始值) 1: $\overline{\text{CS0}}$

48.2.9 端口 C 的 IO 寄存器 0 (PCIOR0)

PCIOR0 是 16 位可读写寄存器，选择端口 C 引脚的输入 / 输出方向。PC8IOR ~ PC0IOR 位分别对应 PC8 ~ PC0 引脚。PCIOR0 在端口 C 的引脚功能为通用输入 / 输出 (PC8 ~ PC0) 时有效，否则无效。如果将 PCIOR0 的位置 “1”，对应的引脚就为输出；如果置 “0” 就为输入。

PCIOR0 的 bit15 ~ 9 是保留位，读写值都为 “0”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PC8 IOR	PC7 IOR	PC6 IOR	PC5 IOR	PC4 IOR	PC3 IOR	PC2 IOR	PC1 IOR	PC0 IOR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

48.2.10 端口 C 的数据寄存器 0 (PCDR0)

PCDR0 是 16 位可读写寄存器，保存端口 C 的数据。PC8DR ~ PC0DR 位分别对应 PC8 ~ PC0 引脚。

在引脚功能为通用输出时，如果给 PCDR0 写值，就从引脚输出此值；如果读 PCDR0，就直接读寄存器的值，与引脚的状态无关。

在引脚功能为通用输入时，如果读 PCDR0，就直接读引脚的状态而不读寄存器的值；如果给 PCDR0 写值，就能将此值写到 PCDR0，但是不影响引脚的状态。PCDR0 的读写如表 48.14 所示。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PC8 DR	PC7 DR	PC6 DR	PC5 DR	PC4 DR	PC3 DR	PC2 DR	PC1 DR	PC0 DR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 9	—	全 0	R	保留位 读写值都为 “0”。
8	PC8DR	0	R/W	参照表 48.14。
7	PC7DR	0	R/W	
6	PC6DR	0	R/W	
5	PC5DR	0	R/W	
4	PC4DR	0	R/W	
3	PC3DR	0	R/W	
2	PC2DR	0	R/W	
1	PC1DR	0	R/W	
0	PC0DR	0	R/W	

表 48.14 端口 C 的数据寄存器 0 (PCDR0) 的读写

- PCDR0 的 bit8 ~ 0

PCIOR0	引脚功能	读	写
0	通用输入	引脚的状态	能写 PCDR0，但是不影响引脚的状态。
	通用输入除外	引脚的状态	能写 PCDR0，但是不影响引脚的状态。
1	通用输出	PCDR0 的值	从引脚输出写入值。
	通用输出除外	PCDR0 的值	能写 PCDR0，但是不影响引脚的状态。

48.2.11 端口 C 的端口寄存器 0 (PCPR0)

PCPR0 是 16 位只读寄存器，PC8PR ~ PC0PR 位分别对应 PC8 ~ PC0 引脚。与 PCCR0 ~ PCCR2 的设置无关，PCPR0 能随时读引脚的值。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PC8 PR	PC7 PR	PC6 PR	PC5 PR	PC4 PR	PC3 PR	PC2 PR	PC1 PR	PC0 PR
初始值:	0	0	0	0	0	0	0	PC8	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 9	—	全 0	R	保留位 读写值都为“0”。
8	PC8PR	引脚的状态	R	如果读此位，就读引脚的状态。写操作无效。
7	PC7PR	引脚的状态	R	
6	PC6PR	引脚的状态	R	
5	PC5PR	引脚的状态	R	
4	PC4PR	引脚的状态	R	
3	PC3PR	引脚的状态	R	
2	PC2PR	引脚的状态	R	
1	PC1PR	引脚的状态	R	
0	PC0PR	引脚的状态	R	

48.2.12 端口 D 的控制寄存器 0 ~ 3 (PDCR0 ~ PDCR3)

PDCR0 ~ PDCR3 分别是 16 位可读写寄存器，选择端口 D 的多路复用引脚的功能。

(1) 端口 D 的控制寄存器 3 (PDCR3)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PD15MD[1:0]	—	—	PD14MD[1:0]	—	—	PD13MD[1:0]	—	—	PD12MD[1:0]				
初始值:	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13、12	PD15MD[1:0]	00/01	R/W	PD15 模式 控制 PD15 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: D15/NAF7 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PD15 (初始值) 01: D15/NAF7 10: PWM2H 11: 禁止设定
11、10	—	全 0	R	保留位 读写值都为“0”。
9、8	PD14MD[1:0]	00/01	R/W	PD14 模式 控制 PD14 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: D14/NAF6 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PD14 (初始值) 01: D14/NAF6 10: PWM2G 11: 禁止设定
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	PD13MD[1:0]	00/01	R/W	PD13 模式 控制 PD13 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: D13/NAF5 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PD13 (初始值) 01: D13/NAF5 10: PWM2F 11: 禁止设定
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	PD12MD[1:0]	0/1	R/W	PD12 模式 控制 PD12 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: D12/NAF4 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PD12 (初始值) 01: D12/NAF4 10: PWM2E 11: 禁止设定

(2) 端口 D 的控制寄存器 2 (PDCR2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PD11MD[1:0]	—	—	PD10MD[1:0]	—	—	PD9MD[1:0]	—	—	PD8MD[1:0]	—	—	—	—
初始值:	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13、12	PD11MD[1:0]	00/01	R/W	PD11 模式 控制 PD11 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: D11/NAF3 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PD11 (初始值) 01: D11/NAF3 10: PWM2D 11: 禁止设定
11、10	—	全 0	R	保留位 读写值都为“0”。
9、8	PD10MD[1:0]	00/01	R/W	PD10 模式 控制 PD10 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: D10/NAF2 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PD10 (初始值) 01: D10/NAF2 10: PWM2C 11: 禁止设定
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	PD9MD[1:0]	00/01	R/W	PD9 模式 控制 PD9 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: D9/NAF1 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PD9 (初始值) 01: D9/NAF1 10: PWM2B 11: 禁止设定
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	PD8MD[1:0]	00/01	R/W	PD8 模式 控制 PD8 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: D8/NAF0 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PD8 (初始值) 01: D8/NAF0 10: PWM2A 11: 禁止设定

(3) 端口 D 的控制寄存器 1 (PDCR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PD7MD[1:0]	—	—	PD6MD[1:0]	—	—	PD5MD[1:0]	—	—	PD4MD[1:0]	—	—	—	—
初始值:	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13、12	PD7MD[1:0]	00/01	R/W	PD7 模式 控制 PD7 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: D7/FWE (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PD7 (初始值) 01: D7/FWE 10: PWM1H 11: 禁止设定
11、10	—	全 0	R	保留位 读写值都为“0”。
9、8	PD6MD[1:0]	00/01	R/W	PD6 模式 控制 PD6 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: D6/FALE (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PD6 (初始值) 01: D6/FALE 10: PWM1G 11: 禁止设定
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	PD5MD[1:0]	00/01	R/W	PD5 模式 控制 PD5 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: D5/FCLE (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PD5 (初始值) 01: D5/FCLE 10: PWM1F 11: 禁止设定
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	PD4MD[1:0]	00/01	R/W	PD4 模式 控制 PD4 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: D4/FRE (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PD4 (初始值) 01: D4/FRE 10: PWM1E 11: 禁止设定

(4) 端口 D 的控制寄存器 0 (PDCR0)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PD3MD[1:0]	—	—	—	PD2MD[1:0]	—	—	—	PD1MD[1:0]	—	—	—	—	PD0MD[1:0]
初始值:	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13、12	PD3MD[1:0]	00/01	R/W	PD3 模式 控制 PD3 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: D3 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PD3 (初始值) 01: D3 10: PWM1D 11: 禁止设定
11、10	—	全 0	R	保留位 读写值都为“0”。
9、8	PD2MD[1:0]	00/01	R/W	PD2 模式 控制 PD2 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: D2 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PD2 (初始值) 01: D2 10: PWM1C 11: 禁止设定
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	PD1MD[1:0]	00/01	R/W	PD1 模式 控制 PD1 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: D1 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PD1 (初始值) 01: D1 10: PWM1B 11: 禁止设定
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	PD0MD[1:0]	00/01	R/W	PD0 模式 控制 PD0 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 0、1 00: 禁止设定 01: D0 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 2 ~ 5 00: PD0 (初始值) 01: D0 10: PWM1A 11: 禁止设定

48.2.13 端口 D 的 IO 寄存器 0 (PDIOR0)

PDIOR0 是 16 位可读写寄存器，选择端口 D 引脚的输入 / 输出方向。PD15IOR ~ PD0IOR 位分别对应 PD15 ~ PD0 引脚。PDIOR0 在端口 D 的引脚功能为通用输入 / 输出 (PD15 ~ PD0) 时有效，否则无效。如果将 PDIOR0 的位置 “1”，对应的引脚就为输出；如果置 “0” 就为输入。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PD15 IOR	PD14 IOR	PD13 IOR	PD12 IOR	PD11 IOR	PD10 IOR	PD9 IOR	PD8 IOR	PD7 IOR	PD6 IOR	PD5 IOR	PD4 IOR	PD3 IOR	PD2 IOR	PD1 IOR	PD0 IOR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

48.2.14 端口 D 的数据寄存器 0 (PDDR0)

PDDR0 是 16 位可读写寄存器，保存端口 D 的数据。PD15DR ~ PD0DR 位分别对应 PD15 ~ PD0 引脚。

在引脚功能为通用输出时，如果给 PDDR0 写值，就从引脚输出此值；如果读 PDDR0，就直接读寄存器的值，与引脚的状态无关。

在引脚功能为通用输入时，如果读 PDDR0，就直接读引脚的状态而不读寄存器的值；如果给 PDDR0 写值，就能将此值写到 PDDR0，但是不影响引脚的状态。PDDR0 的读写如表 48.15 所示。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PD15 DR	PD14 DR	PD13 DR	PD12 DR	PD11 DR	PD10 DR	PD9 DR	PD8 DR	PD7 DR	PD6 DR	PD5 DR	PD4 DR	PD3 DR	PD2 DR	PD1 DR	PD0 DR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	PD15DR	0	R/W	参照表 48.15。
14	PD14DR	0	R/W	
13	PD13DR	0	R/W	
12	PD12DR	0	R/W	
11	PD11DR	0	R/W	
10	PD10DR	0	R/W	
9	PD9DR	0	R/W	
8	PD8DR	0	R/W	
7	PD7DR	0	R/W	
6	PD6DR	0	R/W	
5	PD5DR	0	R/W	
4	PD4DR	0	R/W	
3	PD3DR	0	R/W	
2	PD2DR	0	R/W	
1	PD1DR	0	R/W	
0	PD0DR	0	R/W	

表 48.15 端口 D 的数据寄存器 0 (PDDR0) 的读写

- PDDR0 的 bit15 ~ 0

PDIOR0	引脚功能	读	写
0	通用输入	引脚的状态	能写 PDDR0, 但是不影响引脚的状态。
	通用输入除外	引脚的状态	能写 PDDR0, 但是不影响引脚的状态。
1	通用输出	PDDR0 的值	从引脚输出写入值。
	通用输出除外	PDDR0 的值	能写 PDDR0, 但是不影响引脚的状态。

48.2.15 端口 D 的端口寄存器 0 (PDPR0)

PDPR0 是 16 位只读寄存器, PD15PR ~ PD0PR 位分别对应 PD15 ~ PD0 引脚。与 PDCR0 ~ PDCR3 的设置无关, PDPR0 能随时读引脚的值。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PD15 PR	PD14 PR	PD13 PR	PD12 PR	PD11 PR	PD10 PR	PD9 PR	PD8 PR	PD7 PR	PD6 PR	PD5 PR	PD4 PR	PD3 PR	PD2 PR	PD1 PR	PD0 PR
初始值:	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15	PD15PR	引脚的状态	R	如果读此位, 就读引脚的状态。写操作无效。
14	PD14PR	引脚的状态	R	
13	PD13PR	引脚的状态	R	
12	PD12PR	引脚的状态	R	
11	PD11PR	引脚的状态	R	
10	PD10PR	引脚的状态	R	
9	PD9PR	引脚的状态	R	
8	PD8PR	引脚的状态	R	
7	PD7PR	引脚的状态	R	
6	PD6PR	引脚的状态	R	
5	PD5PR	引脚的状态	R	
4	PD4PR	引脚的状态	R	
3	PD3PR	引脚的状态	R	
2	PD2PR	引脚的状态	R	
1	PD1PR	引脚的状态	R	
0	PD0PR	引脚的状态	R	

48.2.16 端口 E 的控制寄存器 0、1 (PECR0、PECR1)

PECR1 和 PECR0 分别是 16 位可读写寄存器，选择端口 E 的多路复用引脚的功能。

(1) 端口 E 的控制寄存器 1 (PECR1: 只限于 SH7269)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PE7MD[1:0]	—	—	PE6MD[1:0]	—	—	PE5MD[1:0]	—	—	PE4MD[1:0]	—	—	—	—
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13、12	PE7MD[1:0]	00	R/W	PE7 模式 控制 PE7 引脚的功能。 00: PE7 01: SDA3 10: RxD7 11: 禁止设定
11、10	—	全 0	R	保留位 读写值都为“0”。
9、8	PE6MD[1:0]	00	R/W	PE6 模式 控制 PE6 引脚的功能。 00: PE6 01: SCL3 10: RxD6 11: 禁止设定
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	PE5MD[1:0]	00	R/W	PE5 模式 控制 PE5 引脚的功能。 00: PE5 01: SDA2 10: RxD5 11: DV_HSYNC
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	PE4MD[1:0]	00	R/W	PE4 模式 控制 PE4 引脚的功能。 00: PE4 01: SCL2 10: RxD4 11: DV_VSYNC

(2) 端口 E 的控制寄存器 0 (PECR0)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PE3MD[2:0]			—	PE2MD[2:0]			—	PE1MD[2:0]			—	—	PE0MD[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PE3MD[2:0]	000	R/W	PE3 模式 控制 PE3 引脚的功能。 000: PE3 001: SDA1 010: TCLKD 011: $\overline{\text{ADTRG}}$ 100: DV_HSYNC 101: 禁止设定 110: 禁止设定 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PE2MD[2:0]	000	R/W	PE2 模式 控制 PE2 引脚的功能。 000: PE2 001: SCL1 010: $\overline{\text{TCLKD}}$ 011: IOIS16 100: DV_VSYNC 101: 禁止设定 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PE1MD[2:0]	000	R/W	PE1 模式 控制 PE1 引脚的功能。 000: PE1 001: SDA0 010: TCLKB 011: AUDIO_CLK 100: DV_CLK 101: 禁止设定 110: 禁止设定 111: 禁止设定
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	PE0MD[1:0]	00	R/W	PE0 模式 控制 PE0 引脚的功能。 00: PE0 01: SCL0 10: TCLKA 11: LCD_EXTCLK

48.2.17 端口 E 的 IO 寄存器 0 (PEIOR0)

PEIOR0 是 16 位可读写寄存器，选择端口 E 引脚的输入 / 输出方向。PE7IOR ~ PE0IOR 位分别对应 PE7 ~ PE0 引脚。PEIOR0 在端口 E 的引脚功能为通用输入 / 输出 (PE7 ~ PE0) 时有效，否则无效。如果将 PEIOR0 的位置 “1”，对应的引脚就为输出；如果置 “0” 就为输入。

PEIOR0 的 bit15 ~ 8 和 SH7268 的 PEIOR0 的 bit7 ~ 4 是保留位，读写值都为 “0”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PE7 IOR	PE6 IOR	PE5 IOR	PE4 IOR	PE3 IOR	PE2 IOR	PE1 IOR	PE0 IOR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

48.2.18 端口 E 的数据寄存器 0 (PEDR0)

PEDR0 是 16 位可读写寄存器，保存端口 E 的数据。PE7DR ~ PE0DR 位分别对应 PE7 ~ PE0 引脚。

端口 E 的 8 个引脚为漏极开路输出引脚。在引脚功能为通用输出时，如果给 PEDR0 写 “0”，就从引脚输出 Low 电平；如果写 “1”，引脚就变为高阻抗状态。另外，如果读 PEDR0，就直接读寄存器的值，与引脚的状态无关。

在引脚功能为通用输入时，如果读 PEDR0，就直接读引脚的状态而不读寄存器的值；如果给 PEDR0 写值，就能将此值写到 PEDR0，但是不影响引脚的状态。PEDR0 的读写如表 48.16 所示。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PE7 DR	PE6 DR	PE5 DR	PE4 DR	PE3 DR	PE2 DR	PE1 DR	PE0 DR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 8	—	全 0	R	保留位 读写值都为 “0”。
7	PE7DR	0	R/W	参照表 48.16。 【注】 在 SH7268 中，bit7 ~ 4 是保留位，读写值都为 “0”。
6	PE6DR	0	R/W	
5	PE5DR	0	R/W	
4	PE4DR	0	R/W	
3	PE3DR	0	R/W	
2	PE2DR	0	R/W	
1	PE1DR	0	R/W	
0	PE0DR	0	R/W	

表 48.16 端口 E 的数据寄存器 0 (PEDR0) 的读写

- PEDR0 的 bit7 ~ 0

PEIOR0	引脚功能	读	写
0	通用输入	引脚的状态	能写 PEDR0，但是不影响引脚的状态。
	通用输入除外	引脚的状态	能写 PEDR0，但是不影响引脚的状态。
1	通用输出	PEDR0 的值	当 PEXDR 位为 “0” 时，从引脚输出 Low 电平。 当 PEXDR 位为 “1” 时，引脚处于高阻抗状态。
	通用输出除外	PEDR0 的值	能写 PEDR0，但是不影响引脚的状态。

48.2.19 端口 E 的端口寄存器 0 (PEPR0)

PEPR0 是 16 位只读寄存器，PE7PR ~ PE0PR 位分别对应 PE7 ~ PE0 引脚。与 PECR0 和 PECR1 的设置无关，PEPR0 能随时读引脚的值。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PE7 PR	PE6 PR	PE5 PR	PE4 PR	PE3 PR	PE2 PR	PE1 PR	PE0 PR
初始值:	0	0	0	0	0	0	0	0	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 8	—	全 0	R	保留位 读写值都为“0”。
7	PE7PR	引脚的状态	R	如果读此位，就读引脚的状态。写操作无效。
6	PE6PR	引脚的状态	R	【注】 在 SH7268 中，bit7 ~ 4 是保留位，读取值为“0”。
5	PE5PR	引脚的状态	R	
4	PE4PR	引脚的状态	R	
3	PE3PR	引脚的状态	R	
2	PE2PR	引脚的状态	R	
1	PE1PR	引脚的状态	R	
0	PE0PR	引脚的状态	R	

48.2.20 端口 F 的控制寄存器 0 ~ 6 (PFCR0 ~ PFCR6)

PFCR0 ~ PFCR6 分别是 16 位可读写寄存器，选择端口 F 的多路复用引脚的功能。

(1) 端口 F 的控制寄存器 6 (PFCR6)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PF23MD[2:0]			—	PF22MD[2:0]			—	PF21MD[2:0]			—	PF20MD[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PF23MD[2:0]	000	R/W	PF23 模式 控制 PF23 引脚的功能。 000: PF23 001: SD_D2_0 010: 禁止设定 011: 禁止设定 100: TxD3 101: MMC_D2 110: 禁止设定 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PF22MD[2:0]	000	R/W	PF22 模式 控制 PF22 引脚的功能。 000: PF22 001: SD_D3_0 010: 禁止设定 011: 禁止设定 100: RxD3 101: MMC_D3 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PF21MD[2:0]	000	R/W	PF21 模式 控制 PF21 引脚的功能。 000: PF21 001: SD_CMD_0 010: 禁止设定 011: 禁止设定 100: SCK3 101: MMC_CMD 110: 禁止设定 111: 禁止设定
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PF20MD[2:0]	000	R/W	PF20 模式 控制 PF20 引脚的功能。 000: PF20 001: SD_CLK_0 010: SSIDATA3 011: 禁止设定 100: 禁止设定 101: MMC_CLK 110: 禁止设定 111: 禁止设定

(2) 端口 F 的控制寄存器 5 (PFCR5)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PF19MD[2:0]			—	PF18MD[2:0]			—	PF17MD[2:0]			—	PF16MD[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PF19MD[2:0]	000	R/W	PF19 模式 控制 PF19 引脚的功能。 000: PF19 001: SD_D0_0 010: SSIWS3 011: 禁止设定 100: IRQ7 101: MMC_D0 110: 禁止设定 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PF18MD[2:0]	000	R/W	PF18 模式 控制 PF18 引脚的功能。 000: PF18 001: SD_D1_0 010: SSISCK3 011: 禁止设定 100: IRQ6 101: MMC_D1 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PF17MD[2:0]	000	R/W	PF17 模式 控制 PF17 引脚的功能。 000: PF17 001: SD_WP_0 010: 禁止设定 011: FRB 100: IRQ5 101: 禁止设定 110: 禁止设定 111: 禁止设定
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PF16MD[2:0]	000	R/W	PF16 模式 控制 PF16 引脚的功能。 000: PF16 001: SD_CD_0 010: 禁止设定 011: FCE 100: IRQ4 101: MMC_CD 110: 禁止设定 111: 禁止设定

(3) 端口 F 的控制寄存器 4 (PFCR4)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	PF15MD[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

【注】 在写PFCR4时，必须将 bit15~8的值置“H'5A”，并且通过16位存取而不能通过8位存取写此寄存器。

位	位名	初始值	R/W	说明
15 ~ 8	—	全 0	R	保留位 读取值为“0”。 在写 PF15MD[2:0] 时，必须将写入值置“H'5A”。
7 ~ 3	—	全 0	R	保留位 读写值都为“0”。
2 ~ 0	PF15MD[2:0]	000*	R/W	PF15 模式 控制 PF15 引脚的功能。 000: PF15 001: A0 010: SSIDATA2 011: WDTOVF 100: TxD2 101: $\overline{\text{UBCTRG}}$ 110: 禁止设定 111: 禁止设定

【注】 * 在 WDT 上溢引起的上电复位时不进行初始化。

(4) 端口 F 的控制寄存器 3 (PFCR3)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PF14MD[2:0]			—	PF13MD[2:0]			—	PF12MD[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 11	—	全 0	R	保留位 读写值都为“0”。
10 ~ 8	PF14MD[2:0]	000	R/W	PF14 模式 控制 PF14 引脚的功能。 000: PF14 001: A25 010: SSIWS2 011: 禁止设定 100: RxD2 101: 禁止设定 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PF13MD[2:0]	000	R/W	PF13 模式 控制 PF13 引脚的功能。 000: PF13 001: A24 010: SSISCK2 011: 禁止设定 100: SCK2 101: 禁止设定 110: 禁止设定 111: 禁止设定
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PF12MD[2:0]	000	R/W	PF12 模式 控制 PF12 引脚的功能。 000: PF12 001: 禁止设定 010: SSIDATA1 011: DV_DATA3 100: TxD1 101: MMC_D7 110: 禁止设定 111: 禁止设定

(5) 端口 F 的控制寄存器 2 (PFCR2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PF11MD[2:0]			—	PF10MD[2:0]			—	PF9MD[2:0]			—	PF8MD[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PF11MD[2:0]	000	R/W	PF11 模式 控制 PF11 引脚的功能。 000: PF11 001: 禁止设定 010: SSIWS1 011: DV_DATA2 100: RxD1 101: MMC_D6 110: 禁止设定 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PF10MD[2:0]	000	R/W	PF10 模式 控制 PF10 引脚的功能。 000: PF10 001: CS1 010: SSISCK1 011: DV_DATA1 100: SCK1 101: MMC_D5 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PF9MD[2:0]	000	R/W	PF9 模式 控制 PF9 引脚的功能。 000: PF9 001: BS 010: 禁止设定 011: DV_DATA0 100: SCK0 101: MMC_D4 110: RTS1 111: 禁止设定
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PF8MD[2:0]	000	R/W	PF8 模式 控制 PF8 引脚的功能。 000: PF8 001: A23 010: 禁止设定 011: 禁止设定 100: TxD0 101: 禁止设定 110: 禁止设定 111: 禁止设定

(6) 端口 F 的控制寄存器 1 (PFCR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PF7MD[2:0]			—	PF6MD[2:0]			—	PF5MD[2:0]			—	PF4MD[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PF7MD[2:0]	000	R/W	PF7 模式 控制 PF7 引脚的功能。 000: PF7 001: 禁止设定 010: SSIRxD0 011: 禁止设定 100: RxD0 101: SGOUT_3 110: $\overline{\text{CTS1}}$ 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PF6MD[2:0]	000	R/W	PF6 模式 控制 PF6 引脚的功能。 000: PF6 001: $\overline{\text{CE2A}}$ 010: SSITxD0 011: 禁止设定 100: 禁止设定 101: SGOUT_2 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PF5MD[2:0]	000	R/W	PF5 模式 控制 PF5 引脚的功能。 000: PF5 001: 禁止设定 010: SSIWS0 011: 禁止设定 100: 禁止设定 101: SGOUT_1 110: 禁止设定 111: 禁止设定
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PF4MD[2:0]	000	R/W	PF4 模式 控制 PF4 引脚的功能。 000: PF4 001: $\overline{\text{CS5/CE1A}}$ 010: SSISCK0 011: 禁止设定 100: 禁止设定 101: SGOUT_0 110: 禁止设定 111: 禁止设定

(7) 端口 F 的控制寄存器 0 (PFCR0)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PF3MD[2:0]			—	PF2MD[2:0]			—	PF1MD[2:0]			—	PF0MD[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PF3MD[2:0]	000	R/W	PF3 模式 控制 PF3 引脚的功能。 000: $\overline{\text{PF3}}$ 100: TIOC4D 001: $\overline{\text{CS2}}$ 101: AUDIO_XOUT 010: QMI_1/QIO1_1 110: SPBMI_1/SPBIO1_1 011: MISO1 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PF2MD[2:0]	000	R/W	PF2 模式 控制 PF2 引脚的功能。 000: $\overline{\text{PF2}}$ 100: TIOC4C 001: $\overline{\text{WAIT}}$ 101: TEND0 010: QMO_1/QIO0_1 110: SPBMO_1/SPBIO0_1 011: MOSI1 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PF1MD[2:0]	000	R/W	PF1 模式 控制 PF1 引脚的功能。 000: $\overline{\text{PF1}}$ 100: TIOC4B 001: $\overline{\text{BACK}}$ 101: DACK0 010: QSSL_1 110: 禁止设定 011: SSL10 111: 禁止设定
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PF0MD[2:0]	000	R/W	PF0 模式 控制 PF0 引脚的功能。 000: $\overline{\text{PF0}}$ 100: TIOC4A 001: $\overline{\text{BREQ}}$ 101: DREQ0 010: QSPCLK_1 110: 禁止设定 011: RSPCK1 111: 禁止设定

48.2.21 端口 F 的 IO 寄存器 0、1 (PFIOR0、PFIOR1)

PFIOR1 和 PFIOR0 是 16 位可读写寄存器，选择端口 F 引脚的输入 / 输出方向。PF23IOR ~ PF0IOR 位分别对应 PF23 ~ PF0 引脚。在端口 F 的引脚功能为通用输入 / 输出 (PF23 ~ PF0) 或者多功能定时器脉冲单元 2 的 TIOC 输入 / 输出功能时，PFIOR1 和 PFIOR0 有效，否则无效。如果将 PFIOR1 和 PFIOR0 的位置“1”，对应的引脚就为输出；如果置“0”就为输入。

PFIOR1 的 bit15 ~ 8 是保留位，读写值都为“0”。

(1) 端口 F 的 IO 寄存器 1 (PFIOR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PF23 IOR	PF22 IOR	PF21 IOR	PF20 IOR	PF19 IOR	PF18 IOR	PF17 IOR	PF16 IOR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(2) 端口 F 的 IO 寄存器 0 (PFIOR0)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PF15 IOR	PF14 IOR	PF13 IOR	PF12 IOR	PF11 IOR	PF10 IOR	PF9 IOR	PF8 IOR	PF7 IOR	PF6 IOR	PF5 IOR	PF4 IOR	PF3 IOR	PF2 IOR	PF1 IOR	PF0 IOR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

48.2.22 端口 F 的数据寄存器 0、1 (PFDR0、PFDR1)

PFDR1 和 PFDR0 是 16 位可读写寄存器，保存端口 F 的数据。PF23DR ~ PF0DR 位分别对应 PF23 ~ PF0 引脚。

在引脚功能为通用输出时，如果给 PFDR1 或者 PFDR0 写值，就从引脚输出此值；如果读 PFDR1 或者 PFDR0，就直接读寄存器的值，与引脚的状态无关。

在引脚功能为通用输入时，如果读 PFDR1 或者 PFDR0，就直接读引脚的状态而不读寄存器的值。如果给 PFDR1 或者 PFDR0 写值，就能将此值写到 PFDR1 或者 PFDR0，但是不影响引脚的状态。PFDR1 和 PFDR0 的读写如表 48.17 所示。

(1) 端口 F 的数据寄存器 1 (PFDR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PF23 DR	PF22 DR	PF21 DR	PF20 DR	PF19 DR	PF18 DR	PF17 DR	PF16 DR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 8	—	全 0	R	保留位 读写值都为“0”。
7	PF23DR	0	R/W	参照表 48.17。
6	PF22DR	0	R/W	
5	PF21DR	0	R/W	
4	PF20DR	0	R/W	
3	PF19DR	0	R/W	
2	PF18DR	0	R/W	
1	PF17DR	0	R/W	
0	PF16DR	0	R/W	

(2) 端口 F 的数据寄存器 0 (PFDR0)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PF15 DR	PF14 DR	PF13 DR	PF12 DR	PF11 DR	PF10 DR	PF9 DR	PF8 DR	PF7 DR	PF6 DR	PF5 DR	PF4 DR	PF3 DR	PF2 DR	PF1 DR	PF0 DR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	PF15DR	0	R/W	参照表 48.17。
14	PF14DR	0	R/W	
13	PF13DR	0	R/W	
12	PF12DR	0	R/W	
11	PF11DR	0	R/W	
10	PF10DR	0	R/W	
9	PF9DR	0	R/W	
8	PF8DR	0	R/W	
7	PF7DR	0	R/W	
6	PF6DR	0	R/W	
5	PF5DR	0	R/W	
4	PF4DR	0	R/W	
3	PF3DR	0	R/W	
2	PF2DR	0	R/W	
1	PF1DR	0	R/W	
0	PF0DR	0	R/W	

表 48.17 端口 F 的数据寄存器 1、0 (PFDR1、PFDR0) 的读写

- PFDR1 的 bit23~0 和 PFDR0 的 bit15~0

PFIOR0	引脚功能	读	写
0	通用输入	引脚的状态	能写 PFDR1 和 PFDR0, 但是不影响引脚的状态。
	通用输入除外	引脚的状态	能写 PFDR1 和 PFDR0, 但是不影响引脚的状态。
1	通用输出	PFDR0 的值	从引脚输出写入值。
	通用输出除外	PFDR0 的值	能写 PFDR1 和 PFDR0, 但是不影响引脚的状态。

48.2.23 端口 F 的端口寄存器 0、1 (PFPR0、PFPR1)

PFPR1 和 PFPR0 是 16 位只读寄存器，PF23PR ~ PF0PR 位分别对应 PF23 ~ PF0 引脚。与 PFCR0 ~ PFCR5 的设定无关，PFPR1 和 PFPR0 能随时读引脚的值。

(1) 端口 F 的端口寄存器 1 (PFPR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PF23 PR	PF22 PR	PF21 PR	PF20 PR	PF19 PR	PF18 PR	PF17 PR	PF16 PR
初始值:	0	0	0	0	0	0	0	0	PF23	PF22	PF21	PF20	PF19	PF18	PF17	PF16
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 8	—	全 0	R	保留位 读写值都为“0”。
7	PF23PR	引脚的状态	R	如果读此位，就读引脚的状态。写操作无效。
6	PF22PR	引脚的状态	R	
5	PF21PR	引脚的状态	R	
4	PF20PR	引脚的状态	R	
3	PF19PR	引脚的状态	R	
2	PF18PR	引脚的状态	R	
1	PF17PR	引脚的状态	R	
0	PF16PR	引脚的状态	R	

(2) 端口 F 的端口寄存器 0 (PFPR0)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PF15 PR	PF14 PR	PF13 PR	PF12 PR	PF11 PR	PF10 PR	PF9 PR	PF8 PR	PF7 PR	PF6 PR	PF5 PR	PF4 PR	PF3 PR	PF2 PR	PF1 PR	PF0 PR
初始值:	PF15	PF14	PF13	PF12	PF11	PF10	PF9	PF8	PF7	PF6	PF5	PF4	PF3	PF2	PF1	PF0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15	PF15PR	引脚的状态	R	如果读此位，就读引脚的状态。写操作无效。
14	PF14PR	引脚的状态	R	
13	PF13PR	引脚的状态	R	
12	PF12PR	引脚的状态	R	
11	PF11PR	引脚的状态	R	
10	PF10PR	引脚的状态	R	
9	PF9PR	引脚的状态	R	
8	PF8PR	引脚的状态	R	
7	PF7PR	引脚的状态	R	
6	PF6PR	引脚的状态	R	
5	PF5PR	引脚的状态	R	
4	PF4PR	引脚的状态	R	
3	PF3PR	引脚的状态	R	
2	PF2PR	引脚的状态	R	
1	PF1PR	引脚的状态	R	
0	PF0PR	引脚的状态	R	

48.2.24 端口 G 的控制寄存器 0 ~ 6 (PGCR0 ~ PGCR6)

PGCR0 ~ PGCR6 分别是 16 位可读写寄存器，选择端口 G 的多路复用引脚的功能。

(1) 端口 G 的控制寄存器 6 (PGCR6)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PG27MD[1:0]	—	—	PG26MD[1:0]	—	—	PG25MD[1:0]	—	—	PG24MD[1:0]	—	—	PG24MD[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13、12	PG27MD[1:0]	00	R/W	PG27 模式 控制 PG27 引脚的功能。 00: PG27 01: 禁止设定 10: LCD_TCON2 11: LCD_EXTCLK
11、10	—	全 0	R	保留位 读写值都为“0”。
9、8	PG26MD[1:0]	00	R/W	PG26 模式 控制 PG26 引脚的功能。 00: PG26 01: 禁止设定 10: LCD_TCON1 11: 禁止设定
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	PG25MD[1:0]	00	R/W	PG25 模式 控制 PG25 引脚的功能。 00: PG25 01: 禁止设定 10: LCD_TCON0 11: 禁止设定
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	PG24MD[1:0]	00	R/W	PG24 模式 控制 PG24 引脚的功能。 00: PG24 01: 禁止设定 10: LCD_CLK 11: 禁止设定

(2) 端口 G 的控制寄存器 5 (PGCR5)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PG23MD[2:0]			—	PG22MD[2:0]			—	PG21MD[2:0]			—	PG20MD[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PG23MD[2:0]	000	R/W	PG23 模式 控制 PG23 引脚的功能。 000: PG23 001: 禁止设定 010: LCD_DATA23 011: LCD_TCON6 100: TxD5 101: 禁止设定 110: 禁止设定 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PG22MD[2:0]	000	R/W	PG22 模式 控制 PG22 引脚的功能。 000: PG22 001: 禁止设定 010: LCD_DATA22 011: LCD_TCON5 100: RxD5 101: 禁止设定 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PG21MD[2:0]	000	R/W	PG21 模式 控制 PG21 引脚的功能。 000: PG21 001: DV_DATA7 010: LCD_DATA21 011: LCD_TCON4 100: TxD4 101: 禁止设定 110: 禁止设定 111: 禁止设定
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PG20MD[2:0]	000	R/W	PG20 模式 控制 PG20 引脚的功能。 000: PG20 001: DV_DATA6 010: LCD_DATA20 011: LCD_TCON3 100: RxD4 101: 禁止设定 110: 禁止设定 111: 禁止设定

(3) 端口 G 的控制寄存器 4 (PGCR4)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PG19MD[2:0]			—	PG18MD[2:0]			—	—	PG17MD[1:0]		—	—	PG16MD[1:0]	
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PG19MD[2:0]	000	R/W	PG19 模式 控制 PG19 引脚的功能。 000: PG19 001: DV_DATA5 010: LCD_DATA19 011: SPDIF_OUT 100: SCK5 101: 禁止设定 110: 禁止设定 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PG18MD[2:0]	000	R/W	PG18 模式 控制 PG18 引脚的功能。 000: PG18 001: DV_DATA4 010: LCD_DATA18 011: SPDIF_IN 100: SCK4 101: 禁止设定 110: 禁止设定 111: 禁止设定
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	PG17MD[1:0]	00	R/W	PG17 模式 控制 PG17 引脚的功能。 00: PG17 01: WE3/ICIOWR/AH/DQMUU 10: LCD_DATA17 11: 禁止设定
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	PG16MD[1:0]	00	R/W	PG16 模式 控制 PG16 引脚的功能。 00: PG16 01: WE2/ICIORD/DQMUL 10: LCD_DATA16 11: 禁止设定

(4) 端口 G 的控制寄存器 3 (PGCR3)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PG15MD[1:0]	—	—	PG14MD[1:0]	—	—	PG13MD[1:0]	—	—	PG12MD[1:0]	—	—	—	—
初始值:	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1
R/W:	R	R	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13、12	PG15MD[1:0]	00/01	R/W	PG15 模式 控制 PG15 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 1 00: 禁止设定 01: D31 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 0、2 ~ 5 00: PG15 (初始值) 01: D31 10: LCD_DATA15 11: PINT7
11、10	—	全 0	R	保留位 读写值都为“0”。
9、8	PG14MD[1:0]	00/01	R/W	PG14 模式 控制 PG14 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 1 00: 禁止设定 01: D30 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 0、2 ~ 5 00: PG14 (初始值) 01: D30 10: LCD_DATA14 11: PINT6
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	PG13MD[1:0]	00/01	R/W	PG13 模式 控制 PG13 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 1 00: 禁止设定 01: D29 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 0、2 ~ 5 00: PG13 (初始值) 01: D29 10: LCD_DATA13 11: PINT5
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	PG12MD[1:0]	00/01	R/W	PG12 模式 控制 PG12 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 1 00: 禁止设定 01: D28 (初始值) 10: 禁止设定 11: 禁止设定 <ul style="list-style-type: none"> • 引导模式 0、2 ~ 5 00: PG12 (初始值) 01: D28 10: LCD_DATA12 11: PINT4

(5) 端口 G 的控制寄存器 2 (PGCR2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PG11MD[2:0]			—	PG10MD[2:0]			—	PG9MD[2:0]			—	PG8MD[2:0]		
初始值:	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PG11MD[2:0]	000/001	R/W	PG11 模式 控制 PG11 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 1 000: 禁止设定 001: D27 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 <ul style="list-style-type: none"> • 引导模式 0、2 ~ 5 000: PG11 (初始值) 001: D27 010: LCD_DATA11 011: PINT3 100: TIOC3D 101: 禁止设定 110: 禁止设定 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PG10MD[2:0]	000/001	R/W	PG10 模式 控制 PG10 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 1 000: 禁止设定 001: D26 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 <ul style="list-style-type: none"> • 引导模式 0、2 ~ 5 000: PG10 (初始值) 001: D26 010: LCD_DATA10 011: PINT2 100: TIOC3C 101: 禁止设定 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PG9MD[2:0]	000/001	R/W	PG9 模式 控制 PG9 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 1 000: 禁止设定 001: D25 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 <ul style="list-style-type: none"> • 引导模式 0、2 ~ 5 000: PG9 (初始值) 001: D25 010: LCD_DATA9 011: PINT1 100: TIOC3B 101: 禁止设定 110: 禁止设定 111: 禁止设定

位	位名	初始值	R/W	说明
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PG8MD[2:0]	000/001	R/W	PG8 模式 控制 PG8 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 1 000: 禁止设定 001: D24 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 <ul style="list-style-type: none"> • 引导模式 0、2 ~ 5 000: PG8 (初始值) 001: D24 010: LCD_DATA8 011: PINT0 100: TIOC3A 101: 禁止设定 110: 禁止设定 111: 禁止设定

(6) 端口 G 的控制寄存器 1 (PGCR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PG7MD[2:0]			—	PG6MD[2:0]			—	PG5MD[2:0]			—	PG4MD[2:0]		
初始值:	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PG7MD[2:0]	000/001	R/W	PG7 模式 控制 PG7 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 1 000: 禁止设定 001: D23 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 <ul style="list-style-type: none"> • 引导模式 0、2 ~ 5 000: PG7 (初始值) 001: D23 010: LCD_DATA7 011: IRQ7 100: TIOC2B 101: 禁止设定 110: 禁止设定 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。

位	位名	初始值	R/W	说明
10 ~ 8	PG6MD[2:0]	000/001	R/W	PG6 模式 控制 PG6 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 1 000: 禁止设定 001: D22 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 <ul style="list-style-type: none"> • 引导模式 0、2 ~ 5 000: PG6 (初始值) 001: D22 010: LCD_DATA6 011: IRQ6 100: TIOC2A 101: 禁止设定 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PG5MD[2:0]	000/001	R/W	PG5 模式 控制 PG5 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 1 000: 禁止设定 001: D21 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 <ul style="list-style-type: none"> • 引导模式 0、2 ~ 5 000: PG5 (初始值) 001: D21 010: LCD_DATA5 011: IRQ5 100: TIOC1A 101: 禁止设定 110: 禁止设定 111: 禁止设定
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PG4MD[2:0]	000/001	R/W	PG4 模式 控制 PG4 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 1 000: 禁止设定 001: D20 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 <ul style="list-style-type: none"> • 引导模式 0、2 ~ 5 000: PG4 (初始值) 001: D20 010: LCD_DATA4 011: IRQ4 100: TIOC1A 101: 禁止设定 110: 禁止设定 111: 禁止设定

(7) 端口 G 的控制寄存器 0 (PGCR0)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PG3MD[2:0]			—	PG2MD[2:0]			—	PG1MD[2:0]			—	PG0MD[2:0]		
初始值:	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1	0	0	0	0/1
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PG3MD[2:0]	000/001	R/W	PG3 模式 控制 PG3 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 1 000: 禁止设定 001: D19 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 <ul style="list-style-type: none"> • 引导模式 0、2 ~ 5 000: PG3 (初始值) 001: D19 010: LCD_DATA3 011: IRQ3 100: TIOC0D 101: 禁止设定 110: 禁止设定 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PG2MD[2:0]	000/001	R/W	PG2 模式 控制 PG2 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 1 000: 禁止设定 001: D18 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 <ul style="list-style-type: none"> • 引导模式 0、2 ~ 5 000: PG2 (初始值) 001: D18 010: LCD_DATA2 011: IRQ2 100: TIOC0C 101: 禁止设定 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PG1MD[2:0]	000/001	R/W	PG1 模式 控制 PG1 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 1 000: 禁止设定 001: D17 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 <ul style="list-style-type: none"> • 引导模式 0、2 ~ 5 000: PG1 (初始值) 001: D17 010: LCD_DATA1 011: IRQ1 100: TIOC0B 101: 禁止设定 110: 禁止设定 111: 禁止设定

位	位名	初始值	R/W	说明
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PG0MD[2:0]	000/001	R/W	PG0 模式 控制 PG0 引脚的功能。 <ul style="list-style-type: none"> • 引导模式 1 000: 禁止设定 001: D16 (初始值) 010: 禁止设定 011: 禁止设定 100: 禁止设定 101: 禁止设定 110: 禁止设定 111: 禁止设定 <ul style="list-style-type: none"> • 引导模式 0、2 ~ 5 000: PG0 (初始值) 001: D16 010: LCD_DATA0 011: IRQ0 100: TIOC0A 101: 禁止设定 110: 禁止设定 111: 禁止设定

48.2.25 端口 G 的 IO 寄存器 0、1 (PGIOR0、PGIOR1)

PGIOR1 和 PGIOR0 是 16 位可读写寄存器，选择端口 G 引脚的输入 / 输出方向。PG27IOR ~ PG0IOR 位分别对应 PG27 ~ PG0 引脚。在端口 G 的引脚功能为通用输入 / 输出 (PG27 ~ PG0) 或者多功能定时器脉冲单元 2 的 TIOC 输入 / 输出功能时，PGIOR1 和 PGIOR0 有效，否则无效。如果将 PGIOR1 和 PGIOR0 的位置“1”，对应的引脚就为输出；如果置“0”就为输入。

PGIOR1 的 bit15 ~ 12 是保留位，读写值都为“0”。

(1) 端口 G 的 IO 寄存器 1 (PGIOR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PG27 IOR	PG26 IOR	PG25 IOR	PG24 IOR	PG23 IOR	PG22 IOR	PG21 IOR	PG20 IOR	PG19 IOR	PG18 IOR	PG17 IOR	PG16 IOR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(2) 端口 G 的 IO 寄存器 0 (PGIOR0)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PG15 IOR	PG14 IOR	PG13 IOR	PG12 IOR	PG11 IOR	PG10 IOR	PG9 IOR	PG8 IOR	PG7 IOR	PG6 IOR	PG5 IOR	PG4 IOR	PG3 IOR	PG2 IOR	PG1 IOR	PG0 IOR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

48.2.26 端口 G 的数据寄存器 0、1 (PGDR0、PGDR1)

PGDR1和PGDR0是16位可读写寄存器，保存端口G的数据。PG27DR~PG0DR位分别对应PG27~PG0引脚。

在引脚功能为通用输出时，如果给 PGDR1 或者 PGDR0 写值，就从引脚输出此值；如果读 PGDR1 或者 PGDR0，就直接读寄存器的值，与引脚的状态无关。

在引脚功能为通用输入时，如果读 PGDR1 或者 PGDR0，就直接读引脚的状态而不读寄存器的值；如果给 PGDR1 或者 PGDR0 写值，就能将此值写到 PGDR1 或者 PGDR0，但是不影响引脚的状态。PGDR1 和 PGDR0 的读写如表 48.18 所示。

(1) 端口 G 的数据寄存器 1 (PGDR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PG27DR	PG26DR	PG25DR	PG24DR	PG23DR	PG22DR	PG21DR	PG20DR	PG19DR	PG18DR	PG17DR	PG16DR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 12	—	全 0	R	保留位 读写值都为“0”。
11	PG27DR	0	R/W	参照表 48.18。
10	PG26DR	0	R/W	
9	PG25DR	0	R/W	
8	PG24DR	0	R/W	
7	PG23DR	0	R/W	
6	PG22DR	0	R/W	
5	PG21DR	0	R/W	
4	PG20DR	0	R/W	
3	PG19DR	0	R/W	
2	PG18DR	0	R/W	
1	PG17DR	0	R/W	
0	PG16DR	0	R/W	

(2) 端口 G 的数据寄存器 0 (PGDR0)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PG15 DR	PG14 DR	PG13 DR	PG12 DR	PG11 DR	PG10 DR	PG9 DR	PG8 DR	PG7 DR	PG6 DR	PG5 DR	PG4 DR	PG3 DR	PG2 DR	PG1 DR	PG0 DR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	PG15DR	0	R/W	参照表 48.18。
14	PG14DR	0	R/W	
13	PG13DR	0	R/W	
12	PG12DR	0	R/W	
11	PG11DR	0	R/W	
10	PG10DR	0	R/W	
9	PG9DR	0	R/W	
8	PG8DR	0	R/W	
7	PG7DR	0	R/W	
6	PG6DR	0	R/W	
5	PG5DR	0	R/W	
4	PG4DR	0	R/W	
3	PG3DR	0	R/W	
2	PG2DR	0	R/W	
1	PG1DR	0	R/W	
0	PG0DR	0	R/W	

表 48.18 端口 G 的数据寄存器 1、0 (PGDR1、PGDR0) 的读写

- PGDR1 的 bit11 ~ 0 和 PGDR0 的 bit15 ~ 0

PGIOR1、0	引脚功能	读	写
0	通用输入	引脚的状态	能写 PGDR0 和 PGDR1，但是不影响引脚的状态。
	通用输入除外	引脚的状态	能写 PGDR0 和 PGDR1，但是不影响引脚的状态。
1	通用输出	PGDR0 和 PGDR1 的值	从引脚输出写入值。
	通用输出除外	PGDR0 和 PGDR1 的值	能写 PGDR0 和 PGDR1，但是不影响引脚的状态。

48.2.27 端口 G 的端口寄存器 0、1 (PGPR0、PGPR1)

PGPR1 和 PGPR0 分别是 16 位只读寄存器，PG27PR ~ PG0PR 位分别对应 PG27 ~ PG0 引脚。与 PGCR6 ~ PGCR0 的设定无关，PGPR1 和 PGPR0 能随时读引脚的值。

(1) 端口 G 的端口寄存器 1 (PGPR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PG27 PR	PG26 PR	PG25 PR	PG24 PR	PG23 PR	PG22 PR	PG21 PR	PG20 PR	PG19 PR	PG18 PR	PG17 PR	PG16 PR
初始值:	0	0	0	0	PG27	PG26	PG25	PG24	PG23	PG22	PG21	PG20	PG19	PG18	PG17	PG16
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 12	—	全 0	R	保留位 读写值都为“0”。
11	PG27PR	引脚的状态	R	如果读此位，就读引脚的状态。写操作无效。
10	PG26PR	引脚的状态	R	
9	PG25PR	引脚的状态	R	
8	PG24PR	引脚的状态	R	
7	PG23PR	引脚的状态	R	
6	PG22PR	引脚的状态	R	
5	PG21PR	引脚的状态	R	
4	PG20PR	引脚的状态	R	
3	PG19PR	引脚的状态	R	
2	PG18PR	引脚的状态	R	
1	PG17PR	引脚的状态	R	
0	PG16PR	引脚的状态	R	

(2) 端口 G 的端口寄存器 0 (PGPR0)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PG15 PR	PG14 PR	PG13 PR	PG12 PR	PG11 PR	PG10 PR	PG9 PR	PG8 PR	PG7 PR	PG6 PR	PG5 PR	PG4 PR	PG3 PR	PG2 PR	PG1 PR	PG0 PR
初始值:	PG15	PG14	PG13	PG12	PG11	PG10	PG9	PG8	PG7	PG6	PG5	PG4	PG3	PG2	PG1	PG0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15	PG15PR	引脚的状态	R	如果读此位, 就读引脚的状态。写操作无效。
14	PG14PR	引脚的状态	R	
13	PG13PR	引脚的状态	R	
12	PG12PR	引脚的状态	R	
11	PG11PR	引脚的状态	R	
10	PG10PR	引脚的状态	R	
9	PG9PR	引脚的状态	R	
8	PG8PR	引脚的状态	R	
7	PG7PR	引脚的状态	R	
6	PG6PR	引脚的状态	R	
5	PG5PR	引脚的状态	R	
4	PG4PR	引脚的状态	R	
3	PG3PR	引脚的状态	R	
2	PG2PR	引脚的状态	R	
1	PG1PR	引脚的状态	R	
0	PG0PR	引脚的状态	R	

48.2.28 端口 H 的控制寄存器 0、1 (PHCR0、PHCR1)

PHCR1 和 PHCR0 分别是 16 位可读写寄存器，选择端口 H 的多路复用引脚的功能。

(1) 端口 H 的控制寄存器 1 (PHCR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PH7MD[1:0]	—	—	—	PH6MD[1:0]	—	—	—	PH5MD[1:0]	—	—	—	—	PH4MD[1:0]
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13、12	PH7MD[1:0]	00	R/W	PH7 模式 控制 PH7 引脚的功能。 00: PH7 01: AN7 10: PINT7 11: 禁止设定 【注】在 SH7268 中，bit13 和 bit12 是保留位，读写值都为“0”。
11、10	—	全 0	R	保留位 读写值都为“0”。
9、8	PH6MD[1:0]	00	R/W	PH6 模式 控制 PH6 引脚的功能。 00: PH6 01: AN6 10: PINT6 11: 禁止设定 【注】在 SH7268 中，bit9 和 bit8 是保留位，读写值都为“0”。
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	PH5MD[1:0]	00	R/W	PH5 模式 控制 PH5 引脚的功能。 00: PH5 01: AN5 10: PINT5 11: LCD_EXTCLK
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	PH4MD[1:0]	00	R/W	PH4 模式 控制 PH4 引脚的功能。 00: PH4 01: AN4 10: PINT4 11: 禁止设定

(2) 端口 H 的控制寄存器 0 (PHCR0)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PH3MD[1:0]	—	—	—	PH2MD[1:0]	—	—	—	PH1MD[1:0]	—	—	—	—	PH0MD[1:0]
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

位	位名	初始值	R/W	说明
15、14	—	全 0	R	保留位 读写值都为“0”。
13、12	PH3MD[1:0]	00	R/W	PH3 模式 控制 PH3 引脚的功能。 00: PH3 01: AN3 10: PINT3 11: 禁止设定
11、10	—	全 0	R	保留位 读写值都为“0”。
9、8	PH2MD[1:0]	00	R/W	PH2 模式 控制 PH2 引脚的功能。 00: PH2 01: AN2 10: PINT2 11: 禁止设定
7、6	—	全 0	R	保留位 读写值都为“0”。
5、4	PH1MD[1:0]	00	R/W	PH1 模式 控制 PH1 引脚的功能。 00: PH1 01: AN1 10: PINT1 11: 禁止设定
3、2	—	全 0	R	保留位 读写值都为“0”。
1、0	PH0MD[1:0]	00	R/W	PH0 模式 控制 PH0 引脚的功能。 00: PH0 01: AN0 10: PINT0 11: 禁止设定

48.2.29 端口 H 的端口寄存器 0 (PHPR0)

PHPR0 是 16 位只读寄存器, PH7PR ~ PH0PR 位分别对应 PH7 ~ PH0 引脚。在选择通用输入功能时, PHPR0 能随时读引脚的值。在选择 A/D 转换器的功能时, 读取值为 “1”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PH7 PR	PH6 PR	PH5 PR	PH4 PR	PH3 PR	PH2 PR	PH1 PR	PH0 PR
初始值:	0	0	0	0	0	0	0	0	PH7	PH6	PH5	PH4	PH3	PH2	PH1	PH0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15 ~ 8	—	全 0	R	保留位 读写值都为 “0”。
7	PH7PR	引脚的状态	R	如果读此位, 就读引脚的状态。写操作无效。
6	PH6PR	引脚的状态	R	【注】 在 SH7268 中, bit7 和 bit6 是保留位, 读取值为 “1”。
5	PH5PR	引脚的状态	R	
4	PH4PR	引脚的状态	R	
3	PH3PR	引脚的状态	R	
2	PH2PR	引脚的状态	R	
1	PH1PR	引脚的状态	R	
0	PH0PR	引脚的状态	R	

48.2.30 端口 J 的控制寄存器 0 ~ 7 (PJCR0 ~ PJCR7: 只限于 SH7269)

PJCR7 ~ PJCR0 分别是 16 位可读写寄存器, 选择端口 J 的多路复用引脚的功能。

(1) 端口 J 的控制寄存器 7 (PJCR7)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	PJ31 MD	—	PJ30MD[2:0]			—	PJ29MD[2:0]			—	PJ28MD[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 13	—	全 0	R	保留位 读写值都为“0”。
12	PJ31MD	0	R/W	PJ31 模式 控制 PJ31 引脚的功能。 0: PJ31 1: DV_CLK
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PJ30MD[2:0]	000	R/W	PJ30 模式 控制 PJ30 引脚的功能。 000: PJ30 001: 禁止设定 010: SSIDATA5 011: 禁止设定 100: TIOC2B 101: IETxD 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PJ29MD[2:0]	000	R/W	PJ29 模式 控制 PJ29 引脚的功能。 000: PJ29 001: 禁止设定 010: SSIWS5 011: 禁止设定 100: TIOC2A 101: IERxD 110: 禁止设定 111: 禁止设定
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PJ28MD[2:0]	000	R/W	PJ28 模式 控制 PJ28 引脚的功能。 000: PJ28 001: 禁止设定 010: SSISCK5 011: 禁止设定 100: TIOC1B 101: $\overline{\text{RTS7}}$ 110: 禁止设定 111: 禁止设定

(2) 端口 J 的控制寄存器 6 (PJCR6)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PJ27MD[2:0]			—	PJ26MD[2:0]			—	PJ25MD[2:0]			—	PJ24MD[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PJ27MD[2:0]	000	R/W	PJ27 模式 控制 PJ27 引脚的功能。 000: PJ27 001: SGOUT_3 010: 禁止设定 011: 禁止设定 100: TIOC1A 101: CTS7 110: 禁止设定 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PJ26MD[2:0]	000	R/W	PJ26 模式 控制 PJ26 引脚的功能。 000: PJ26 001: SGOUT_2 010: SSIDATA4 011: LCD_TCON5 100: 禁止设定 101: TxD7 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PJ25MD[2:0]	000	R/W	PJ25 模式 控制 PJ25 引脚的功能。 000: PJ25 001: SGOUT_1 010: SSIWS4 011: LCD_TCON4 100: SPDIF_OUT 101: RxD7 110: 禁止设定 111: 禁止设定
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PJ24MD[2:0]	000	R/W	PJ24 模式 控制 PJ24 引脚的功能。 000: PJ24 001: SGOUT_0 010: SSISCK4 011: LCD_TCON3 100: SPDIF_IN 101: SCK7 110: 禁止设定 111: 禁止设定

(3) 端口 J 的控制寄存器 5 (PJCR5)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PJ23MD[2:0]			—	PJ22MD[2:0]			—	PJ21MD[2:0]			—	PJ20MD[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PJ23MD[2:0]	000	R/W	PJ23 模式 控制 PJ23 引脚的功能。 000: PJ23 001: DV_DATA23 010: LCD_DATA23 011: LCD_TCON6 100: IRQ3 101: CTx1 110: CTx0&CTx1 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PJ22MD[2:0]	000	R/W	PJ22 模式 控制 PJ22 引脚的功能。 000: PJ22 001: DV_DATA22 010: LCD_DATA22 011: LCD_TCON5 100: IRQ2 101: CRx1 110: CRx0/CRx1 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PJ21MD[2:0]	000	R/W	PJ21 模式 控制 PJ21 引脚的功能。 000: PJ21 001: DV_DATA21 010: LCD_DATA21 011: LCD_TCON4 100: IRQ1 101: CTx2 110: CTx0&CTx1&CTx2 111: 禁止设定
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PJ20MD[2:0]	000	R/W	PJ20 模式 控制 PJ20 引脚的功能。 000: PJ20 001: DV_DATA20 010: LCD_DATA20 011: LCD_TCON3 100: IRQ0 101: CRx2 110: CRx0/CRx1/CRx2 111: 禁止设定

(4) 端口 J 的控制寄存器 4 (PJCR4)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PJ19MD[2:0]			—	PJ18MD[2:0]			—	PJ17MD[2:0]			—	PJ16MD[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PJ19MD[2:0]	000	R/W	PJ19 模式 控制 PJ19 引脚的功能。 000: PJ19 001: DV_DATA19 010: LCD_DATA19 011: MIS00 100: TIOC0D 101: SIOFRxD 110: AUDIO_XOUT 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PJ18MD[2:0]	000	R/W	PJ18 模式 控制 PJ18 引脚的功能。 000: PJ18 001: DV_DATA18 010: LCD_DATA18 011: MOSI0 100: TIOC0C 101: SIOFTxD 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PJ17MD[2:0]	000	R/W	PJ17 模式 控制 PJ17 引脚的功能。 000: PJ17 001: DV_DATA17 010: LCD_DATA17 011: SSL00 100: TIOC0B 101: SIOFSYNC 110: 禁止设定 111: 禁止设定
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PJ16MD[2:0]	000	R/W	PJ16 模式 控制 PJ16 引脚的功能。 000: PJ16 001: DV_DATA16 010: LCD_DATA16 011: RSPCK0 100: TIOC0A 101: SIOFCK 110: 禁止设定 111: 禁止设定

(5) 端口 J 的控制寄存器 3 (PJCR3)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PJ15MD[2:0]			—	PJ14MD[2:0]			—	PJ13MD[2:0]			—	PJ12MD[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PJ15MD[2:0]	000	R/W	PJ15 模式 控制 PJ15 引脚的功能。 000: PJ15 001: DV_DATA15 010: LCD_DATA15 011: PINT7 100: PWM2H 101: TxD7 110: 禁止设定 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PJ14MD[2:0]	000	R/W	PJ14 模式 控制 PJ14 引脚的功能。 000: PJ14 001: DV_DATA14 010: LCD_DATA14 011: PINT6 100: PWM2G 101: TxD6 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PJ13MD[2:0]	000	R/W	PJ13 模式 控制 PJ13 引脚的功能。 000: PJ13 001: DV_DATA13 010: LCD_DATA13 011: PINT5 100: PWM2F 101: TxD5 110: 禁止设定 111: 禁止设定
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PJ12MD[1:0]	000	R/W	PJ12 模式 控制 PJ12 引脚的功能。 000: PJ12 001: DV_DATA12 010: LCD_DATA12 011: PINT4 100: PWM2E 101: SCK7 110: 禁止设定 111: 禁止设定

(6) 端口 J 的控制寄存器 2 (PJCR2)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PJ11MD[2:0]			—	PJ10MD[2:0]			—	PJ9MD[2:0]			—	PJ8MD[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PJ11MD[2:0]	000	R/W	PJ11 模式 控制 PJ11 引脚的功能。 000: PJ11 001: DV_DATA11 010: LCD_DATA11 011: PINT3 100: PWM2D 101: SCK6 110: 禁止设定 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PJ10MD[2:0]	000	R/W	PJ10 模式 控制 PJ10 引脚的功能。 000: PJ10 001: DV_DATA10 010: LCD_DATA10 011: PINT2 100: PWM2C 101: SCK5 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PJ9MD[2:0]	000	R/W	PJ9 模式 控制 PJ9 引脚的功能。 000: PJ9 001: DV_DATA9 010: LCD_DATA9 011: PINT1 100: PWM2B 101: $\overline{\text{RTS5}}$ 110: 禁止设定 111: 禁止设定
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PJ8MD[2:0]	000	R/W	PJ8 模式 控制 PJ8 引脚的功能。 000: PJ8 001: DV_DATA8 010: LCD_DATA8 011: PINT0 100: PWM2A 101: $\overline{\text{CTS5}}$ 110: 禁止设定 111: 禁止设定

(7) 端口 J 的控制寄存器 1 (PJCR1)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PJ7MD[2:0]			—	PJ6MD[2:0]			—	PJ5MD[2:0]			—	PJ4MD[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PJ7MD[2:0]	000	R/W	PJ7 模式 控制 PJ7 引脚的功能。 000: PJ7 001: DV_DATA7 010: LCD_DATA7 011: SD_D2_1 100: PWM1H 101: 禁止设定 110: 禁止设定 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PJ6MD[2:0]	000	R/W	PJ6 模式 控制 PJ6 引脚的功能。 000: PJ6 001: DV_DATA6 010: LCD_DATA6 011: SD_D3_1 100: PWM1G 101: 禁止设定 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PJ5MD[2:0]	000	R/W	PJ5 模式 控制 PJ5 引脚的功能。 000: PJ5 001: DV_DATA5 010: LCD_DATA5 011: SD_CMD_1 100: PWM1F 101: 禁止设定 110: 禁止设定 111: 禁止设定
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PJ4MD[2:0]	000	R/W	PJ4 模式 控制 PJ4 引脚的功能。 000: PJ4 001: DV_DATA4 010: LCD_DATA4 011: SD_CLK_1 100: PWM1E 101: 禁止设定 110: 禁止设定 111: 禁止设定

(8) 端口 J 的控制寄存器 0 (PJCR0)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PJ3MD[2:0]			—	PJ2MD[2:0]			—	PJ1MD[2:0]			—	PJ0MD[2:0]		
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14 ~ 12	PJ3MD[2:0]	000	R/W	PJ3 模式 控制 PJ3 引脚的功能。 000: PJ3 001: DV_DATA3 010: LCD_DATA3 011: SD_D0_1 100: PWM1D 101: 禁止设定 110: 禁止设定 111: 禁止设定
11	—	0	R	保留位 读写值都为“0”。
10 ~ 8	PJ2MD[2:0]	000	R/W	PJ2 模式 控制 PJ2 引脚的功能。 000: PJ2 001: DV_DATA2 010: LCD_DATA2 011: SD_D1_1 100: PWM1C 101: 禁止设定 110: 禁止设定 111: 禁止设定
7	—	0	R	保留位 读写值都为“0”。
6 ~ 4	PJ1MD[2:0]	000	R/W	PJ1 模式 控制 PJ1 引脚的功能。 000: PJ1 001: DV_DATA1 010: LCD_DATA1 011: SD_WP_1 100: PWM1B 101: 禁止设定 110: 禁止设定 111: 禁止设定
3	—	0	R	保留位 读写值都为“0”。
2 ~ 0	PJ0MD[2:0]	000	R/W	PJ0 模式 控制 PJ0 引脚的功能。 000: PJ0 001: DV_DATA0 010: LCD_DATA0 011: SD_CD_1 100: PWM1A 101: 禁止设定 110: 禁止设定 111: 禁止设定

48.2.31 端口 J 的 IO 寄存器 0、1（PJIOR0、PJIOR1：只限于 SH7269）

PJIOR1 和 PJIOR0 是 16 位可读写寄存器，选择端口 J 引脚的输入 / 输出方向。PJ31IOR ~ PJ0IOR 位分别对应 PJ31 ~ PJ0 引脚。PJIOR1 和 PJIOR0 在端口 J 的引脚功能为通用输入 / 输出（PJ31 ~ PJ0）时有效，否则无效。如果将 PJIOR1 和 PJIOR0 的位置“1”，对应的引脚就为输出；如果置“0”就为输入。

(1) 端口 J 的 IO 寄存器 1（PJIOR1）

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PJ31 IOR	PJ30 IOR	PJ29 IOR	PJ28 IOR	PJ27 IOR	PJ26 IOR	PJ25 IOR	PJ24 IOR	PJ23 IOR	PJ22 IOR	PJ21 IOR	PJ20 IOR	PJ19 IOR	PJ18 IOR	PJ17 IOR	PJ16 IOR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(2) 端口 J 的 IO 寄存器 0（PJIOR0）

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PJ15 IOR	PJ14 IOR	PJ13 IOR	PJ12 IOR	PJ11 IOR	PJ10 IOR	PJ9 IOR	PJ8 IOR	PJ7 IOR	PJ6 IOR	PJ5 IOR	PJ4 IOR	PJ3 IOR	PJ2 IOR	PJ1 IOR	PJ0 IOR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

48.2.32 端口 J 的数据寄存器 0、1（PJDR0、PJDR1：只限于 SH7289）

PJDR1 和 PJDR0 是 16 位可读写寄存器，保存端口 J 的数据。PJ31DR ~ PJ0DR 位分别对应 PJ31 ~ PJ0 引脚。

在引脚功能为通用输出时，如果给 PJDR1 或者 PJDR0 写值，就从引脚输出此值；如果读 PJDR1 或者 PJDR0，就直接读寄存器的值，与引脚的状态无关。

在引脚功能为通用输入时，如果读 PJDR1 或者 PJDR0，就直接读引脚的状态而不读寄存器的值。如果给 PJDR1 或者 PJDR0 写值，就能将此值写到 PJDR1 或者 PJDR0，但是不影响引脚的状态。PJDR1 和 PJDR0 的读写如表 48.19 所示。

(1) 端口 J 的数据寄存器 1（PJDR1）

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PJ31 DR	PJ30 DR	PJ29 DR	PJ28 DR	PJ27 DR	PJ26 DR	PJ25 DR	PJ24 DR	PJ23 DR	PJ22 DR	PJ21 DR	PJ20 DR	PJ19 DR	PJ18 DR	PJ17 DR	PJ16 DR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	PJ31DR	0	R/W	参照表 48.19。
14	PJ30DR	0	R/W	
13	PJ29DR	0	R/W	
12	PJ28DR	0	R/W	
11	PJ27DR	0	R/W	
10	PJ26DR	0	R/W	
9	PJ25DR	0	R/W	
8	PJ24DR	0	R/W	
7	PJ23DR	0	R/W	
6	PJ22DR	0	R/W	
5	PJ21DR	0	R/W	
4	PJ20DR	0	R/W	
3	PJ19DR	0	R/W	
2	PJ18DR	0	R/W	
1	PJ17DR	0	R/W	
0	PJ16DR	0	R/W	

(2) 端口 J 的数据寄存器 0 (PJDR0)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PJ15 DR	PJ14 DR	PJ13 DR	PJ12 DR	PJ11 DR	PJ10 DR	PJ9 DR	PJ8 DR	PJ7 DR	PJ6 DR	PJ5 DR	PJ4 DR	PJ3 DR	PJ2 DR	PJ1 DR	PJ0 DR
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	PJ15DR	0	R/W	参照表 48.19。
14	PJ14DR	0	R/W	
13	PJ13DR	0	R/W	
12	PJ12DR	0	R/W	
11	PJ11DR	0	R/W	
10	PJ10DR	0	R/W	
9	PJ9DR	0	R/W	
8	PJ8DR	0	R/W	
7	PJ7DR	0	R/W	
6	PJ6DR	0	R/W	
5	PJ5DR	0	R/W	
4	PJ4DR	0	R/W	
3	PJ3DR	0	R/W	
2	PJ2DR	0	R/W	
1	PJ1DR	0	R/W	
0	PJ0DR	0	R/W	

表 48.19 端口 J 的数据寄存器 1、0 (PJDR1、PJDR0) 的读写

• PJDR1 和 PJDR0

PJIOR1、0	引脚功能	读	写
0	通用输入	引脚的状态	能写 PJDR1 和 PJDR0，但是不影响引脚的状态。
	通用输入除外	引脚的状态	能写 PJDR1 和 PJDR0，但是不影响引脚的状态。
1	通用输出	PJDR1 和 PJDR0 的值	从引脚输出写入值。
	通用输出除外	PJDR1 和 PJDR0 的值	能写 PJDR1 和 PJDR0，但是不影响引脚的状态。

48.2.33 端口 J 的端口寄存器 0、1（PJPR0、PJPR1：只限于 SH7269）

PJPR1 和 PJPR0 是 16 位只读寄存器，PJ31PR ~ PJ0PR 位分别对应 PJ31 ~ PJ0 引脚。与 PJCR0 和 PJCR1 的设定无关，PJPR1 和 PJPR0 能随时读引脚的值。

(1) 端口 J 的端口寄存器 1（PJPR1）

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PJ31 PR	PJ30 PR	PJ29 PR	PJ28 PR	PJ27 PR	PJ26 PR	PJ25 PR	PJ24 PR	PJ23 PR	PJ22 PR	PJ21 PR	PJ20 PR	PJ19 PR	PJ18 PR	PJ17 PR	PJ16 PR
初始值:	PJ31	PJ30	PJ29	PJ28	PJ27	PJ26	PJ25	PJ24	PJ23	PJ22	PJ21	PJ20	PJ19	PJ18	PJ17	PJ16
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15	PJ31PR	引脚的状态	R	如果读此位，就读引脚的状态。写操作无效。
14	PJ30PR	引脚的状态	R	
13	PJ29PR	引脚的状态	R	
12	PJ28PR	引脚的状态	R	
11	PJ27PR	引脚的状态	R	
10	PJ26PR	引脚的状态	R	
9	PJ25PR	引脚的状态	R	
8	PJ24PR	引脚的状态	R	
7	PJ23PR	引脚的状态	R	
6	PJ22PR	引脚的状态	R	
5	PJ21PR	引脚的状态	R	
4	PJ20PR	引脚的状态	R	
3	PJ19PR	引脚的状态	R	
2	PJ18PR	引脚的状态	R	
1	PJ17PR	引脚的状态	R	
0	PJ16PR	引脚的状态	R	

(2) 端口 J 的端口寄存器 0 (PJPR0)

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PJ15 PR	PJ14 PR	PJ13 PR	PJ12 PR	PJ11 PR	PJ10 PR	PJ9 PR	PJ8 PR	PJ7 PR	PJ6 PR	PJ5 PR	PJ4 PR	PJ3 PR	PJ2 PR	PJ1 PR	PJ0 PR
初始值:	PJ15	PJ14	PJ13	PJ12	PJ11	PJ10	PJ9	PJ8	PJ7	PJ6	PJ5	PJ4	PJ3	PJ2	PJ1	PJ0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

位	位名	初始值	R/W	说明
15	PJ15PR	引脚的状态	R	如果读此位，就读引脚的状态。写操作无效。
14	PJ14PR	引脚的状态	R	
13	PJ13PR	引脚的状态	R	
12	PJ12PR	引脚的状态	R	
11	PJ11PR	引脚的状态	R	
10	PJ10PR	引脚的状态	R	
9	PJ9PR	引脚的状态	R	
8	PJ8PR	引脚的状态	R	
7	PJ7PR	引脚的状态	R	
6	PJ6PR	引脚的状态	R	
5	PJ5PR	引脚的状态	R	
4	PJ4PR	引脚的状态	R	
3	PJ3PR	引脚的状态	R	
2	PJ2PR	引脚的状态	R	
1	PJ1PR	引脚的状态	R	
0	PJ0PR	引脚的状态	R	

48.2.34 串行瑞萨接口噪声消除器的控制寄存器 (SNCR)

SNCR 是 16 位可读写寄存器，控制从芯片引脚到串行音频接口的输入线路中的噪声消除器。只能在对应串行音频接口的通道为从属模式的情况下设定各位。在主控模式中，必须使用初始值。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	SSI5 NCE	SSI4 NCE	SSI3 NCE	SSI2 NCE	SSI1 NCE	SSI0 NCE
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15 ~ 6	—	全 0	R	保留位 读写值都为“0”。
5	SSI5NCE	0	R/W	串行音频接口通道 5 的噪声消除器的允许 选择 SSISCK5、SSIWS5 和 SSIDATA5 的噪声消除器的有效或者无效。 0: 噪声消除器无效 1: 噪声消除器有效
4	SSI4NCE	0	R/W	串行音频接口通道 4 的噪声消除器的允许 选择 SSISCK4、SSIWS4 和 SSIDATA4 的噪声消除器的有效或者无效。 0: 噪声消除器无效 1: 噪声消除器有效
3	SSI3NCE	0	R/W	串行音频接口通道 3 的噪声消除器的允许 选择 SSISCK3、SSIWS3 和 SSIDATA3 的噪声消除器的有效或者无效。 0: 噪声消除器无效 1: 噪声消除器有效
2	SSI2NCE	0	R/W	串行音频接口通道 2 的噪声消除器的允许 选择 SSISCK2、SSIWS2 和 SSIDATA2 的噪声消除器的有效或者无效。 0: 噪声消除器无效 1: 噪声消除器有效
1	SSI1NCE	0	R/W	串行音频接口通道 1 的噪声消除器的允许 选择 SSISCK1、SSIWS1 和 SSIDATA1 的噪声消除器的有效或者无效。 0: 噪声消除器无效 1: 噪声消除器有效
0	SSI0NCE	0	R/W	串行音频接口通道 0 的噪声消除器的允许 选择 SSISCK0、SSIWS0 和 SSIRxD0 的噪声消除器的有效或者无效。 0: 噪声消除器无效 1: 噪声消除器有效

49. 低功耗模式

作为低功耗模式，本 LSI 支持睡眠模式、软件待机模式、深度待机模式、模块待机功能。在低功耗模式中，能通过停止 CPU、时钟、内部存储器、部分内部外围模块等的功能或者切断电源来降低功耗。通过复位或者中断等解除低功耗模式。

49.1 特点

49.1.1 低功耗模式的种类

低功耗模式有以下的模式和功能。

1. 睡眠模式
2. 软件待机模式
3. 深度待机模式
4. 模块待机功能

从程序执行状态转移到各模式的条件、各模式中的 CPU 和外围模块等的状态以及各模式的解除方法如表 49.1 所示。

表 49.1 低功耗模式的状态

低功耗模式	转移条件	状态*1									解除方法
		CPG	CPU	CPU 寄存器	高速内部 RAM 高速缓冲存储器	大容量内部 RAM (包含用于保持的内部RAM)	内部外围模块	实时时钟	电源	外部存储器	
睡眠模式	在 STBCR1 的 STBY 位为“0”的状态下执行 SLEEP 指令。	运行	停止	保持	运行	运行	运行	运行*2	运行	必须设定为自动刷新。	<ul style="list-style-type: none"> • 中断 • 手动复位 • 上电复位 • DMA地址错误
软件待机模式	在 STBCR1 的 STBY 位为“1”并且 DEEP 位为“0”的状态下执行 SLEEP 指令。	停止	停止	保持	停止 (保持内容*5*6)	停止 (保持内容*5*7)	停止	运行*2	运行	必须设定为自刷新。	<ul style="list-style-type: none"> • NMI中断 • IRQ中断 • 上电复位
深度待机模式	在 STBCR1 的 STBY 位和 DEEP 位为“1”的状态下执行 SLEEP 指令。	停止	停止	停止	停止 (不保持内容)	停止 (用于保持的内部RAM 保持内容*3)	停止	运行*2	停止	必须设定为自刷新。	<ul style="list-style-type: none"> • NMI中断*4 • 上电复位*4 • 实时时钟 • 闹钟中断*4 • 用于解除的引脚的变化*4
模块待机模式	将 STBCR2、STBCR3、STBCR4、STBCR5、STBCR6、STBCR7、STBCR8、STBCR9 和 STBCR10 的 MSTP 位置“1”。	运行	运行	保持	运行	运行	指定的模块 停止运行	停止	运行	必须设定为自动刷新。	<ul style="list-style-type: none"> • 将 MSTP 位清“0”。 • 上电复位 (但是，只限于用户调试接口和直接存储器存取控制器)

【注】 *1 引脚状态为保持状态或者高阻抗。详细内容请参照“53. 引脚状态和处理方法”的“53.1 引脚状态”。

*2 实时时钟在 RCR2 寄存器的 START 位为“1”时运行。详细内容请参照“15. 实时时钟”。如果通过上电复位解除深度待机模式，就不能保持运行状态。必须重新对实时时钟进行初始设定。

*3 如果将 RRAMKP 寄存器的 RRAMKP3 ~ RRAMKP0 位置“1”，就能在转移到深度待机模式时保持内部 RAM (用于保持) 的对象区的内容。但是，如果通过上电复位解除深度待机模式，就对保持的内容进行初始化。

*4 通过中断 (NMI、实时时钟闹钟中断)、复位 (上电复位) 以及用于解除的引脚 (PJ23 ~ PJ20、PG3 ~ PG2、PF19 ~ PF16、PC7、PC5) 的变化，解除深度待机模式。但是，在通过复位以外的方法解除深度待机模式时，也执行上电复位异常处理而不执行中断异常处理。PJ23 ~ PJ20 只能用于 SH7269。

*5 如果通过上电复位解除软件待机模式，就对保持的内容进行初始化。

*6 如果通过将 SYSCR1 寄存器的 RAME 位或者 SYSCR2 寄存器的 RAMWE 位设定为禁止并且通过上电复位解除软件待机模式，就能继续保持高速内部 RAM。

*7 如果通过将 SYSCR3 寄存器的 VRAME 位或者 SYSCR4 寄存器的 VRAMWE 位设定为禁止并且通过上电复位解除软件待机模式，就能继续保持大容量内部 RAM (包括用于保持的内部 RAM)。

49.2 寄存器说明

寄存器结构如表 49.2 所示。

表 49.2 寄存器结构

寄存器名	略称	R/W	初始值	地址	存取长度
待机控制寄存器 1	STBCR1	R/W	H'00	H'FFFE0014	8
待机控制寄存器 2	STBCR2	R/W	H'00	H'FFFE0018	8
待机控制寄存器 3	STBCR3	R/W	H'7E	H'FFFE0408	8
待机控制寄存器 4	STBCR4	R/W	H'FF	H'FFFE040C	8
待机控制寄存器 5	STBCR5	R/W	H'FF	H'FFFE0410	8
待机控制寄存器 6	STBCR6	R/W	H'FF	H'FFFE0414	8
待机控制寄存器 7	STBCR7	R/W	H'FF	H'FFFE0418	8
待机控制寄存器 8	STBCR8	R/W	H'FF	H'FFFE041C	8
待机控制寄存器 9	STBCR9	R/W	H'FF	H'FFFE0440	8
待机控制寄存器 10	STBCR10	R/W	H'FF	H'FFFE0444	8
软件复位控制寄存器 1	SWRSTCR1	R/W	H'00	H'FFFE0430	8
软件复位控制寄存器 2	SWRSTCR2	R/W	H'00	H'FFFE0434	8
系统控制寄存器 1	SYSCR1	R/W	H'FF	H'FFFE0400	8
系统控制寄存器 2	SYSCR2	R/W	H'FF	H'FFFE0404	8
系统控制寄存器 3	SYSCR3	R/W	H'FF	H'FFFE0420	8
系统控制寄存器 4	SYSCR4	R/W	H'FF	H'FFFE0424	8
系统控制寄存器 5	SYSCR5	R/W	H'00	H'FFFE0428	8
用于保持的内部 RAM 保持区指定寄存器	RRAMKP	R/W	H'00	H'FFFE6800	8
深度待机控制寄存器	DSCTR	R/W	H'00	H'FFFE6802	8
深度待机解除源选择寄存器	DSSSR	R/W	H'0000	H'FFFE6804	16
深度待机解除边沿选择寄存器	DSESR	R/W	H'0000	H'FFFE6806	16
深度待机解除源标志寄存器	DSFR	R/W	H'0000	H'FFFE6808	16
XTAL 晶体振荡器的增益控制寄存器	XTALCTR	R/W	H'00	H'FFFE6810	8

49.2.1 待机控制寄存器 1 (STBCR1)

STBCR1 是 8 位可读写寄存器，指定低功耗模式的状态。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	STBY	DEEP	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7	STBY	0	R/W	软件待机、深度待机 指定向软件待机模式或者深度待机模式的转移。 0x: 通过执行 SLEEP 指令转移到睡眠模式。 10: 通过执行 SLEEP 指令转移到软件待机模式。 11: 通过执行 SLEEP 指令转移到深度待机模式。
6	DEEP	0	R/W	
5 ~ 0	—	全 0	R	保留位 读写值都为“0”。

【符号说明】 x: Don't care

49.2.2 待机控制寄存器 2 (STBCR2)

STBCR2 是 8 位可读写寄存器，控制各模块的运行。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	MSTP 10	MSTP 9	MSTP 8	MSTP 7	—	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R

位	位名	初始值	R/W	说明
7	MSTP10	0	R/W	模块停止 10 如果将 MSTP10 位置“1”，就停止给用户调试接口提供时钟。 0: 用户调试接口运行 1: 停止给用户调试接口提供时钟
6	MSTP9	0	R/W	模块停止 9 如果将 MSTP9 位置“1”，就停止给用户中断控制器提供时钟。 0: 用户中断控制器运行 1: 停止给用户中断控制器提供时钟
5	MSTP8	0	R/W	模块停止 8 如果将 MSTP8 位置“1”，就停止给直接存储器存取控制器提供时钟。 0: 直接存储器存取控制器运行 1: 停止给直接存储器存取控制器提供时钟
4	MSTP7	0	R/W	模块停止 7 如果将 MSTP7 位置“1”，就停止给 FPU 提供时钟。不能在置“1”后写“0”来清除 MSTP7 位。即，将 MSTP7 位置“1”，在暂停给 FPU 提供时钟后，不能通过将 MSTP7 位清“0”来重新开始给 FPU 提供时钟。为了在停止给 FPU 提供时钟后重新给 FPU 提供时钟，必须对本 LSI 进行上电复位。 0: FPU 运行 1: 停止给 FPU 提供时钟
3~0	—	全 0	R	保留位 读写值都为“0”。

49.2.3 待机控制寄存器 3 (STBCR3)

STBCR3 是 8 位可读写寄存器，控制各模块的运行。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	HIZ	MSTP 36	MSTP 35	—	—	MSTP 32	—	MSTP 30
初始值:	0	1	1	1	1	1	1	0
R/W:	R/W	R/W	R/W	R	R	R/W	R	R/W

位	位名	初始值	R/W	说明
7	HIZ	0	R/W	端口高阻抗 在软件待机模式和深度待机模式中，选择是保持特定的输出引脚的状态还是置为高阻抗。有关控制哪个引脚，请参照“53. 引脚状态和处理方法”的“53.1 引脚状态”。 不能在看门狗定时器的 WTSCR 的 TME 位为“1”的状态下设定此位。要将输出引脚的状态置为高阻抗时，必须在 TME 位为“0”的状态下将 HIZ 位置位。 0: 在软件待机模式和深度待机模式中保持引脚状态。 1: 在软件待机模式和深度待机模式中将引脚状态置为高阻抗。
6	MSTP36	1	R/W	模块停止 36 如果将 MSTP36 位置“1”，就停止给 IEBus 控制器提供时钟。 0: IEBus 控制器运行 1: 停止给 IEBus 控制器提供时钟
5	MSTP35	1	R/W	模块停止 35 如果将 MSTP35 位置“1”，就停止给多功能定时器脉冲单元 2 提供时钟。 0: 多功能定时器脉冲单元 2 运行 1: 停止给多功能定时器脉冲单元 2 提供时钟
4、3	—	全 1	R	保留位 读写值都为“1”。
2	MSTP32	1	R/W	模块停止 32 如果将 MSTP32 位置“1”，就停止给 A/D 转换器提供时钟。 0: A/D 转换器运行 1: 停止给 A/D 转换器提供时钟
1	—	1	R	保留位 读写值都为“1”。
0	MSTP30	0	R/W	模块停止 30 如果将 MSTP30 位置“1”，就停止给实时时钟提供时钟。 0: 实时时钟运行 1: 停止给实时时钟提供时钟 【注】 如果要停止实时时钟，就必须设定以下实时时钟的寄存器： <ul style="list-style-type: none"> • 将控制寄存器 2 (RCR2) 的 RTCEN 位置“0”。 • 将控制寄存器 5 (RCR5) 的 RCKSEL 位置“0”。 必须在上述设定后将 MSTP30 位置“1”。

49.2.4 待机控制寄存器 4 (STBCR4)

STBCR4 是 8 位可读写寄存器，控制各模块的运行。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	MSTP 47	MSTP 46	MSTP 45	MSTP 44	MSTP 43	MSTP 42	MSTP 41	MSTP 40
初始值:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	MSTP47	1	R/W	模块停止 47 如果将 MSTP47 位置“1”，就停止给 FIFO 内置型串行通信单元的通道 0 提供时钟。 0: FIFO 内置型串行通信单元的通道 0 运行 1: 停止给 FIFO 内置型串行通信单元的通道 0 提供时钟
6	MSTP46	1	R/W	模块停止 46 如果将 MSTP46 位置“1”，就停止给 FIFO 内置型串行通信单元的通道 1 提供时钟。 0: FIFO 内置型串行通信单元的通道 1 运行 1: 停止给 FIFO 内置型串行通信单元的通道 1 提供时钟
5	MSTP45	1	R/W	模块停止 45 如果将 MSTP45 位置“1”，就停止给 FIFO 内置型串行通信单元的通道 2 提供时钟。 0: FIFO 内置型串行通信单元的通道 2 运行 1: 停止给 FIFO 内置型串行通信单元的通道 2 提供时钟
4	MSTP44	1	R/W	模块停止 44 如果将 MSTP44 位置“1”，就停止给 FIFO 内置型串行通信单元的通道 3 提供时钟。 0: FIFO 内置型串行通信单元的通道 3 运行 1: 停止给 FIFO 内置型串行通信单元的通道 3 提供时钟
3	MSTP43	1	R/W	模块停止 43 如果将 MSTP43 位置“1”，就停止给 FIFO 内置型串行通信单元的通道 4 提供时钟。 0: FIFO 内置型串行通信单元的通道 4 运行 1: 停止给 FIFO 内置型串行通信单元的通道 4 提供时钟
2	MSTP42	1	R/W	模块停止 42 如果将 MSTP42 位置“1”，就停止给 FIFO 内置型串行通信单元的通道 5 提供时钟。 0: FIFO 内置型串行通信单元的通道 5 运行 1: 停止给 FIFO 内置型串行通信单元的通道 5 提供时钟
1	MSTP41	1	R/W	模块停止 41 如果将 MSTP41 位置“1”，就停止给 FIFO 内置型串行通信单元的通道 6 提供时钟。 0: FIFO 内置型串行通信单元的通道 6 运行 1: 停止给 FIFO 内置型串行通信单元的通道 6 提供时钟
0	MSTP40	1	R/W	模块停止 40 如果将 MSTP40 位置“1”，就停止给 FIFO 内置型串行通信单元的通道 7 提供时钟。 0: FIFO 内置型串行通信单元的通道 7 运行 1: 停止给 FIFO 内置型串行通信单元的通道 7 提供时钟

49.2.5 待机控制寄存器 5 (STBCR5)

STBCR5 是 8 位可读写寄存器，控制各模块的运行。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	MSTP 57	MSTP 56	MSTP 55	MSTP 54	MSTP 53	MSTP 52	MSTP 51	MSTP 50
初始值:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	MSTP57	1	R/W	模块停止 57 如果将 MSTP57 位置“1”，就停止给 I ² C 总线接口 3 的通道 0 提供时钟。 0: I ² C 总线接口 3 的通道 0 运行 1: 停止给 I ² C 总线接口 3 的通道 0 提供时钟
6	MSTP56	1	R/W	模块停止 56 如果给 MSTP56 位置“1”，就停止给 I ² C 总线接口 3 的通道 1 提供时钟。 0: I ² C 总线接口 3 的通道 1 运行 1: 停止给 I ² C 总线接口 3 的通道 1 提供时钟
5	MSTP55	1	R/W	模块停止 55 如果将 MSTP55 位置“1”，就停止给 I ² C 总线接口 3 的通道 2 提供时钟。 0: I ² C 总线接口 3 的通道 2 运行 1: 停止给 I ² C 总线接口 3 的通道 2 提供时钟
4	MSTP54	1	R	模块停止 54 如果将 MSTP54 位置“1”，就停止给 I ² C 总线接口 3 的通道 3 提供时钟。 0: I ² C 总线接口 3 的通道 3 运行 1: 停止给 I ² C 总线接口 3 的通道 3 提供时钟
3	MSTP53	1	R/W	模块停止 53 如果将 MSTP53 位置“1”，就停止给控制器局域网的通道 0 提供时钟。 0: 控制器局域网的通道 0 运行 1: 停止给控制器局域网的通道 0 提供时钟
2	MSTP52	1	R/W	模块停止 52 如果将 MSTP52 位置“1”，就停止给控制器局域网的通道 1 提供时钟。 0: 控制器局域网的通道 1 运行 1: 停止给控制器局域网的通道 1 提供时钟
1	MSTP51	1	R/W	模块停止 51 如果将 MSTP51 位置“1”，就停止给瑞萨串行外围接口的通道 0 提供时钟。 0: 瑞萨串行外围接口的通道 0 运行 1: 停止给瑞萨串行外围接口的通道 0 提供时钟
0	MSTP50	1	R/W	模块停止 50 如果将 MSTP50 位置“1”，就停止给瑞萨串行外围接口的通道 1 提供时钟。 0: 瑞萨串行外围接口的通道 1 运行 1: 停止给瑞萨串行外围接口的通道 1 提供时钟

49.2.6 待机控制寄存器 6 (STBCR6)

STBCR6 是 8 位可读写寄存器，控制各模块的运行。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	MSTP 67	MSTP 66	MSTP 65	MSTP 64	MSTP 63	MSTP 62	MSTP 61	MSTP 60
初始值:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	MSTP67	1	R/W	模块停止 67 如果将 MSTP67 位置“1”，就停止给串行音频接口的通道 0 提供时钟。 0: 串行音频接口的通道 0 运行 1: 停止给串行音频接口的通道 0 提供时钟
6	MSTP66	1	R/W	模块停止 66 如果将 MSTP66 位置“1”，就停止给串行音频接口的通道 1 提供时钟。 0: 串行音频接口的通道 1 运行 1: 停止给串行音频接口的通道 1 提供时钟
5	MSTP65	1	R/W	模块停止 65 如果给 MSTP65 位置“1”，就停止给串行音频接口的通道 2 提供时钟。 0: 串行音频接口的通道 2 运行 1: 停止给串行音频接口的通道 2 提供时钟
4	MSTP64	1	R/W	模块停止 64 如果将 MSTP64 位置“1”，就停止给串行音频接口的通道 3 提供时钟。 0: 串行音频接口的通道 3 运行 1: 停止给串行音频接口的通道 3 提供时钟
3	MSTP63	1	R/W	模块停止 63 如果将 MSTP63 位置“1”，就停止给 CD-ROM 解码器提供时钟。 0: CD-ROM 解码器运行 1: 停止给 CD-ROM 解码器提供时钟
2	MSTP62	1	R/W	模块停止 62 如果给 MSTP62 位置“1”，就停止给采样率转换器的通道 0 提供时钟。 0: 采样率转换器的通道 0 运行 1: 停止给采样率转换器的通道 0 提供时钟
1	MSTP61	1	R/W	模块停止 61 如果将 MSTP61 位置“1”，就停止给采样率转换器的通道 1 提供时钟。 0: 采样率转换器的通道 1 运行 1: 停止给采样率转换器的通道 1 提供时钟
0	MSTP60	1	R/W	模块停止 60 如果将 MSTP60 位置“1”，就停止给 USB2.0 主机 / 功能模块提供时钟。 0: USB2.0 主机 / 功能模块运行 1: 停止给 USB2.0 主机 / 功能模块提供时钟

49.2.7 待机控制寄存器 7 (STBCR7)

STBCR7 是 8 位可读写寄存器，控制各模块的运行。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	MSTP 77	MSTP 76	MSTP 75	—	MSTP 73	MSTP 72	—	MSTP 70
初始值:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R	R/W	R/W	R	R/W

位	位名	初始值	R/W	说明
7	MSTP77	1	R/W	模块停止 77 如果将 MSTP77 位置“1”，就停止给带 FIFO 的时钟同步串行 I/O 提供时钟。 0: 带 FIFO 的时钟同步串行 I/O 运行 1: 停止给带 FIFO 的时钟同步串行 I/O 提供时钟
6	MSTP76	1	R/W	模块停止 76 如果将 MSTP76 位置“1”，就停止给瑞萨 SPDIF 接口提供时钟。 0: 瑞萨 SPDIF 接口运行 1: 停止给瑞萨 SPDIF 接口提供时钟
5	MSTP75	1	R/W	模块停止 75 如果将 MSTP75 位置“1”，就停止给 SPI 多 I/O 总线控制器提供时钟。 0: SPI 多 I/O 总线控制器运行 1: 停止给 SPI 多 I/O 总线控制器提供时钟
4	—	1	R	保留位 读写值都为“1”。
3	MSTP73	1	R/W	模块停止 73 如果将 MSTP73 位置“1”，就停止给视频显示控制器 4 提供时钟。 0: 视频显示控制器 4 运行 1: 停止给视频显示控制器 4 提供时钟
2	MSTP72	1	R/W	模块停止 72 如果将 MSTP72 位置“1”，就停止给比较匹配定时器提供时钟。 0: 比较匹配定时器运行 1: 停止给比较匹配定时器提供时钟
1	—	1	R	保留位 读写值都为“1”。
0	MSTP70	1	R/W	模块停止 70 如果将 MSTP70 位置“1”，就停止给 NAND 闪存控制器提供时钟。 0: NAND 闪存控制器运行 1: 停止给 NAND 闪存控制器提供时钟

49.2.8 待机控制寄存器 8 (STBCR8)

STBCR8 是 8 位可读写寄存器，控制各模块的运行。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	MSTP 87	MSTP 86	MSTP 85	MSTP 84	—	MST P82	MST P81	—
初始值:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R

位	位名	初始值	R/W	说明
7	MSTP87	1	R/W	模块停止 87 如果将 MSTP87 位置“1”，就停止给马达控制 PWM 定时器提供时钟。 0: 马达控制 PWM 定时器运行 1: 停止给马达控制 PWM 定时器提供时钟
6	MSTP86	1	R/W	模块停止 86 如果将 MSTP86 位置“1”，就停止给 MMC 主机接口提供时钟。 0: 运行 MMC 主机接口 1: 停止给 MMC 主机接口提供时钟
5	MSTP85	1	R/W	模块停止 85 如果将 MSTP85 位置“1”，就停止给失真校正引擎提供时钟。 0: 失真校正引擎运行 1: 停止给失真校正引擎提供时钟
4	MSTP84	1	R/W	模块停止 84 如果将 MSTP84 位置“1”，就停止给 OpenVG 瑞萨图形处理器提供时钟。 0: OpenVG 瑞萨图形处理器运行 1: 停止给 OpenVG 瑞萨图形处理器提供时钟
3	—	1	R	保留位 读写值都为“1”。
2	MSTP82	1	R/W	模块停止 82 如果将 MSTP82 位置“1”，就停止给瑞萨四路串行外围接口的通道 0 提供时钟。 0: 瑞萨四路串行外围接口的通道 0 运行 1: 停止给瑞萨四路串行外围接口的通道 0 提供时钟
1	MSTP81	1	R/W	模块停止 81 如果将 MSTP81 位置“1”，就停止给瑞萨四路串行外围接口的通道 1 提供时钟。 0: 瑞萨四路串行外围接口的通道 1 运行 1: 停止给瑞萨四路串行外围接口的通道 1 提供时钟
0	—	1	R	保留位 读写值都为“1”。

49.2.9 待机控制寄存器 9 (STBCR9)

STBCR9 是 8 位可读写寄存器，控制各模块的运行。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	MSTP 97	MSTP 96	MSTP 95	MSTP 94	MSTP 93	MSTP 92	MSTP 91	MSTP 90
初始值:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	MSTP97	1	R/W	模块停止 97 如果将 MSTP97 位置“1”，就停止给 SD 主机接口 00 提供时钟。 0: SD 主机接口 00 运行 1: 停止给 SD 主机接口 00 提供时钟
6	MSTP96	1	R/W	模块停止 96 如果将 MSTP96 位置“1”，就停止给 SD 主机接口 01 提供时钟。 0: SD 主机接口 01 运行 1: 停止给 SD 主机接口 01 提供时钟
5	MSTP95	1	R/W	模块停止 95 如果给 MSTP95 位置“1”，就停止给 SD 主机接口 10 提供时钟。 0: SD 主机接口 10 运行 1: 停止给 SD 主机接口 10 提供时钟
4	MSTP94	1	R/W	模块停止 94 如果将 MSTP94 位置“1”，就停止给 SD 主机接口 11 提供时钟。 0: SD 主机接口 11 运行 1: 停止给 SD 主机接口 11 提供时钟
3	MSTP93	1	R/W	模块停止 93 如果将 MSTP93 位置“1”，就停止给串行音频接口的通道 4 提供时钟。 0: 串行音频接口的通道 4 运行 1: 停止给串行音频接口的通道 4 提供时钟
2	MSTP92	1	R/W	模块停止 92 如果将 MSTP92 位置“1”，就停止给串行音频接口的通道 5 提供时钟。 0: 串行音频接口的通道 5 运行 1: 停止给串行音频接口的通道 5 提供时钟
1	MSTP91	1	R/W	模块停止 91 如果将 MSTP91 位置“1”，就停止给采样率转换器的通道 2 提供时钟。 0: 运行采样率转换器的通道 2 1: 停止给采样率转换器的通道 2 提供时钟
0	MSTP90	1	R/W	模块停止 90 如果将 MSTP90 位置“1”，就停止给控制器局域网的通道 2 提供时钟。 0: 控制器局域网的通道 2 运行 1: 停止给控制器局域网的通道 2 提供时钟

49.2.10 待机控制寄存器 10 (STBCR10)

STBCR10 是 8 位可读写寄存器，控制各模块的运行。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	MSTP 107	MSTP 106	MSTP 105	—	MSTP 103	MSTP 102	MSTP 101	MSTP 100
初始值:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	MSTP107	1	R/W	模块停止 107 如果将 MSTP107 位置“1”，就停止给数字视频解码器提供时钟。 0: 数字视频解码器运行 1: 停止给数字视频解码器提供时钟
6	MSTP106	1	R/W	模块停止 106 如果将 MSTP106 位置“1”，就停止给 JPEG 编解码器单元提供时钟。 0: JPEG 编解码器单元运行 1: 停止给 JPEG 编解码器单元提供时钟
5	MSTP105	1	R/W	模块停止 105 如果将 MSTP105 位置“1”，就停止给显示输出比较单元提供时钟。 0: 显示输出比较单元运行 1: 停止给显示输出比较单元提供时钟
4	—	1	R	保留位 读写值都为“1”。
3	MSTP103	1	R/W	模块停止 103 如果将 MSTP103 位置“1”，就停止给音频发生器的通道 0 提供时钟。 0: 音频发生器的通道 0 运行 1: 停止给音频发生器的通道 0 提供时钟
2	MSTP102	1	R/W	模块停止 102 如果将 MSTP102 位置“1”，就停止给音频发生器的通道 1 提供时钟。 0: 音频发生器的通道 1 运行 1: 停止给音频发生器的通道 1 提供时钟
1	MSTP101	1	R/W	模块停止 101 如果将 MSTP101 位置“1”，就停止给音频发生器的通道 2 提供时钟。 0: 音频发生器的通道 2 运行 1: 停止给音频发生器的通道 2 提供时钟
0	MSTP100	1	R/W	模块停止 100 如果将 MSTP100 位置“1”，就停止给音频发生器的通道 3 提供时钟。 0: 音频发生器的通道 3 运行 1: 停止给音频发生的器通道 3 提供时钟

49.2.11 软件复位控制寄存器 1 (SWRSTCR1)

SWRSTCR1 是 8 位可读写寄存器，进行串行音频接口和 IEBus 控制器的软件复位控制以及音频晶体谐振器的控制。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	AXT ALE	SSIF5 SRST	SSIF4 SRST	IEB SRST	SSIF3 SRST	SSIF2 SRST	SSIF1 SRST	SSIF0 SRST
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7	AXTALE	0	R/W	AUDIO_X1 时钟控制 控制 AUDIO_X1 引脚的功能。 0: 内部晶体振荡器振荡 / 允许外部时钟输入 1: 停止内部晶体振荡器 / 禁止外部时钟输入
6	SSIF5SRST	0	R/W	串行音频接口通道 5 的软件复位 通过软件控制串行音频接口通道 5 的复位。 0: 解除串行音频接口通道 5 的复位 1: 将串行音频接口通道 5 设定为复位状态
5	SSIF4SRST	0	R/W	串行音频接口通道 4 的软件复位 通过软件控制串行音频接口通道 4 的复位。 0: 解除串行音频接口通道 4 的复位 1: 将串行音频接口通道 4 设定为复位状态
4	IEBSRST	0	R/W	IEBus 控制器的软件复位 通过软件控制 IEBus 控制器的复位。 0: 解除 IEBus 控制器的复位 1: 将 IEBus 控制器设定为复位状态
3	SSIF3SRST	0	R/W	串行音频接口通道 3 的软件复位 通过软件控制串行音频接口通道 3 的复位。 0: 解除串行音频接口通道 3 的复位。 1: 将串行音频接口通道 3 设定为复位状态
2	SSIF2SRST	0	R/W	串行音频接口通道 2 的软件复位 通过软件控制串行音频接口通道 2 的复位。 0: 解除串行音频接口通道 2 的复位 1: 将串行音频接口通道 2 设定为复位状态
1	SSIF1SRST	0	R/W	串行音频接口通道 1 的软件复位 通过软件控制串行音频接口通道 1 的复位。 0: 解除串行音频接口通道 1 的复位 1: 将串行音频接口通道 1 设定为复位状态
0	SSIF0SRST	0	R/W	串行音频接口通道 0 的软件复位 通过软件控制串行音频接口通道 0 的复位。 0: 解除串行音频接口通道 0 的复位 1: 将串行音频接口通道 0 设定为复位状态

49.2.12 软件复位控制寄存器 2 (SWRSTCR2)

SWRSTCR2 是 8 位可读写寄存器，进行各模块的软件复位控制。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	—	—	—	JCU SRST	RGPV GSRST	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R	R	R

位	位名	初始值	R/W	说明
7 ~ 5	—	全 0	R	保留位 读写值都为“0”。
4	JCUSRST	0	R/W	JPEG 编解码器单元的软件复位 通过软件控制 JPEG 编解码器单元的复位。 0: 解除 JPEG 编解码器单元的复位 1: 将 JPEG 编解码器单元设定为复位状态
3	RGPVGSRST	0	R/W	OpenVG 瑞萨图形处理器的软件复位 通过软件控制 OpenVG 瑞萨图形处理器的复位。 0: 解除 OpenVG 瑞萨图形处理器的复位 1: 将 OpenVG 瑞萨图形处理器设定为复位状态
2 ~ 0	—	全 0	R	保留位 读写值都为“0”。

49.2.13 系统控制寄存器 1 (SYSCR1)

SYSCR1 是 8 位可读写寄存器，设定允许或者禁止高速内部 RAM 各页面的存取（读写）。

如果将 SYSCR1 的 RAME_n (n=0 ~ 3) 位置“1”，页面 n 的存取就有效；如果置“0”，就不能存取页面 n。此时，读页面 n 和取指令的读取值为不定值，并且忽视页面 n 的写操作。初始值为“1”。

如果要将 RAME_n 位置“0”，就必须在设定 RAME_n 位前读写页面 n 的任意同一地址。否则，最后写在页面 n 的数据有可能不被写到高速内部 RAM。

必须通过高速内部 RAM 空间以外的程序设定 SYSCR1。另外，必须紧接在写 SYSCR1 的指令之后配置读 SYSCR1 的指令。否则，就不能保证高速内部 RAM 的正常存取。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	RAME3	RAME2	RAME1	RAME0
初始值:	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 4	—	全 1	R	保留位 读写值都为“1”。
3	RAME3	1	R/W	RAM 允许 3（对象：高速内部 RAM 的页面 3*） 0: 页面 3 的存取无效 1: 页面 3 的存取有效
2	RAME2	1	R/W	RAM 允许 2（对象：高速内部 RAM 的页面 2*） 0: 页面 2 的存取无效 1: 页面 2 的存取有效
1	RAME1	1	R/W	RAM 允许 1（对象：高速内部 RAM 的页面 1*） 0: 页面 1 的存取无效 1: 页面 1 的存取有效
0	RAME0	1	R/W	RAM 允许 0（对象：高速内部 RAM 的页面 0*） 0: 页面 0 的存取无效 1: 页面 0 的存取有效

【注】 * 有关各页面的地址，请参照“47. 内部 RAM”。

49.2.14 系统控制寄存器 2 (SYSCR2)

SYSCR2 是 8 位可读写寄存器，设定允许或者禁止高速内部 RAM 各页面的写操作。

如果将 SYSCR2 的 RAMWE_n (n=0 ~ 3) 位置 “1”，页面 n 的写操作就有效；如果置 “0”，就忽视页面 n 的写操作。初始值为 “1”。

如果要将 RAMWE_n 位置 “0”，就必须在设定 RAMWE_n 位前读写页面 n 的任意地址。否则，最后写在页面 n 的数据有可能不被写到高速内部 RAM。

必须通过高速内部 RAM 空间以外的程序设定 SYSCR2。另外，必须紧接在写 SYSCR2 的指令之后配置读 SYSCR2 的指令。否则，就不能保证高速内部 RAM 的正常存取。

【注】 在写此寄存器时，请参照 “49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	RAM WE3	RAM WE2	RAM WE1	RAM WE0
初始值:	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 4	—	全 1	R	保留位 读写值都为 “1”。
3	RAMWE3	1	R/W	RAM 写允许 3 (对象: 高速内部 RAM 的页面 3*) 0: 页面 3 的写操作无效 1: 页面 3 的写操作有效
2	RAMWE2	1	R/W	RAM 写允许 2 (对象: 高速内部 RAM 的页面 2*) 0: 页面 2 的写操作无效 1: 页面 2 的写操作有效
1	RAMWE1	1	R/W	RAM 写允许 1 (对象: 高速内部 RAM 的页面 1*) 0: 页面 1 的写操作无效 1: 页面 1 的写操作有效
0	RAMWE0	1	R/W	RAM 写允许 0 (对象: 高速内部 RAM 的页面 0*) 0: 页面 0 的写操作无效 1: 页面 0 的写操作有效

【注】 * 有关各页面的地址，请参照 “47. 内部 RAM”。

49.2.15 系统控制寄存器 3 (SYSCR3)

SYSCR3 是 8 位可读写寄存器，设定允许或者禁止大容量内部 RAM 各页面的存取（读写）。

如果将 SYSCR3 的 VRAME_n（n=0～5）位置“1”，页面 n 的存取就有效；如果置“0”，就不能存取页面 n。此时，读页面 n 和取指令的读取值为不定值，并且忽视页面 n 的写操作。初始值为“1”。

必须通过大容量内部 RAM 空间以外的程序设定 SYSCR3。另外，必须紧接在写 SYSCR3 的指令之后配置读 SYSCR3 的指令。否则，就不能保证大容量内部 RAM 的正常存取。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	—	—	VRA ME5	VRA ME4	VRA ME3	VRA ME2	VRA ME1	VRA ME0
初始值:	1	1	1	1	1	1	1	1
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7、6	—	全 1	R	保留位 读写值都为“1”。
5	VRAME5	1	R/W	RAM 允许 5（对象：大容量内部 RAM 的页面 5*） 0：页面 5 的存取无效 1：页面 5 的存取有效
4	VRAME4	1	R/W	RAM 允许 4（对象：大容量内部 RAM 的页面 4*） 0：页面 4 的存取无效 1：页面 4 的存取有效
3	VRAME3	1	R/W	RAM 允许 3（对象：大容量内部 RAM 的页面 3*） 0：页面 3 的存取无效 1：页面 3 的存取有效
2	VRAME2	1	R/W	RAM 允许 2（对象：大容量内部 RAM 的页面 2*） 0：页面 2 的存取无效 1：页面 2 的存取有效
1	VRAME1	1	R/W	RAM 允许 1（对象：大容量内部 RAM 的页面 1*） 0：页面 1 的存取无效 1：页面 1 的存取有效
0	VRAME0	1	R/W	RAM 允许 0（对象：大容量内部 RAM 的页面 0*） 0：页面 0 的存取无效 1：页面 0 的存取有效

【注】 * 有关各页面的地址，请参照“47. 内部 RAM”。

49.2.16 系统控制寄存器 4 (SYSCR4)

SYSCR4 是 8 位可读写寄存器，设定允许或者禁止大容量内部 RAM 各页面的写操作。

如果将 SYSCR4 的 VRAMWEn (n=0 ~ 5) 位置 “1”，页面 n 的写操作就有效；如果置 “0”，就忽视页面 n 的写操作。初始值为 “1”。

必须通过大容量内部 RAM 空间以外的程序设定 SYSCR4。另外，必须紧接在写 SYSCR4 的指令之后配置读 SYSCR4 的指令。否则，就不能保证大容量内部 RAM 的正常存取。

【注】 在写此寄存器时，请参照 “49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	—	—	VRAM WE5	VRAM WE4	VRAM WE3	VRAM WE2	VRAM WE1	VRAM WE0
初始值:	1	1	1	1	1	1	1	1
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7、6	—	全 1	R	保留位 读写值都为 “1”。
5	VRAMWE5	1	R/W	RAM 写允许 5 (对象: 大容量内部 RAM 的页面 5*) 0: 页面 5 的写操作无效 1: 页面 5 的写操作有效
4	VRAMWE4	1	R/W	RAM 写允许 4 (对象: 大容量内部 RAM 的页面 4*) 0: 页面 4 的写操作无效 1: 页面 4 的写操作有效
3	VRAMWE3	1	R/W	RAM 写允许 3 (对象: 大容量内部 RAM 的页面 3*) 0: 页面 3 的写操作无效 1: 页面 3 的写操作有效
2	VRAMWE2	1	R/W	RAM 写允许 2 (对象: 大容量内部 RAM 的页面 2*) 0: 页面 2 的写操作无效 1: 页面 2 的写操作有效
1	VRAMWE1	1	R/W	RAM 写允许 1 (对象: 大容量内部 RAM 的页面 1*) 0: 页面 1 的写操作无效 1: 页面 1 的写操作有效
0	VRAMWE0	1	R/W	RAM 写允许 0 (对象: 大容量内部 RAM 的页面 0*) 0: 页面 0 的写操作无效 1: 页面 0 的写操作有效

【注】 * 有关各页面的地址，请参照 “47. 内部 RAM”。

49.2.17 系统控制寄存器 5 (SYSCR5)

SYSCR5 是 8 位可读写寄存器，设定允许或者禁止用于保持的内部 RAM 各页面的写操作。

如果将 SYSCR5 的 RRAMWEn (n=0 ~ 3) 位置 “1”，页面 n 的写操作就有效；如果置 “0”，就忽视页面 n 的写操作。初始值为 “0”。

必须通过用于保持的内部 RAM 空间以外的程序设定 SYSCR5。

【注】 在写此寄存器时，请参照 “49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	RRAM WE3	RRAM WE2	RRAM WE1	RRAM WE0
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 4	—	全 0	R	保留位 读写值都为 “0”。
3	RRAMWE3	0	R/W	RAM 写允许 3 (对象: 用于保持的内部 RAM 的页面 3*2) 0: 页面 3 的写操作无效 1: 页面 3 的写操作有效
2	RRAMWE2	0	R/W	RAM 写允许 2 (对象: 用于保持的内部 RAM 的页面 2*2) 0: 页面 2 的写操作无效 1: 页面 2 的写操作有效
1	RRAMWE1	0	R/W	RAM 写允许 1 (对象: 用于保持的内部 RAM 的页面 1*2) 0: 页面 1 的写操作无效 1: 页面 1 的写操作有效
0	RRAMWE0	0	R/W	RAM 写允许 0 (对象: 用于保持的内部 RAM 的页面 0*2) 0: 页面 0 的写操作无效 1: 页面 0 的写操作有效

【注】 *1 有关各页面的地址，请参照 “47. 内部 RAM”。

*2 如果已将 SYSCR3.VRAME0 置 “0” (大容量内部 RAM 的页面 0 的存取无效)，就不能存取 (读写) 用于保持的内部 RAM，与此位的设定无关。如果已将 SYSCR4.VRAMWE0 置 “0” (大容量内部 RAM 的页面 0 的写操作无效)，就不能写用于保持的内部 RAM，与此位的设定无关。

49.2.18 用于保持的内部 RAM 保持区指定寄存器 (RRAMKP)

RRAMKP 是 8 位可读写寄存器，在深度待机模式中设定是否保持用于保持的对象内部 RAM 区的内容。

如果将 RRAMKP3 ~ 0 位置“1”，就在深度待机模式中保持用于保持的对象内部 RAM 区的内容。如果清“0”，就在深度待机模式中不保持用于保持的对象内部 RAM 区的内容。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	RRAM KP3	RRAM KP2	RRAM KP1	RRAM KP0
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
7 ~ 4	—	全 0	R	保留位 读写值都为“0”。
3	RRAMKP3	0	R/W	用于保持的内部 RAM 保持区 3（对象：用于保持的内部 RAM 的页面 3*） 0：在深度待机模式中，不保持用于保持的内部 RAM。 1：在深度待机模式中，保持用于保持的内部 RAM。
2	RRAMKP2	0	R/W	用于保持的内部 RAM 保持区 2（对象：用于保持的内部 RAM 的页面 2*） 0：在深度待机模式中，不保持用于保持的内部 RAM。 1：在深度待机模式中，保持用于保持的内部 RAM。
1	RRAMKP1	0	R/W	用于保持的内部 RAM 保持区 1（对象：用于保持的内部 RAM 的页面 1*） 0：在深度待机模式中，不保持用于保持的内部 RAM。 1：在深度待机模式中，保持用于保持的内部 RAM。
0	RRAMKP0	0	R/W	用于保持的内部 RAM 保持区 0（对象：用于保持的内部 RAM 的页面 0*） 0：在深度待机模式中，不保持用于保持的内部 RAM。 1：在深度待机模式中，保持用于保持的内部 RAM。

【注】 * 有关各页面的地址，请参照“47. 内部 RAM”。

49.2.19 深度待机控制寄存器 (DSCTR)

DSCTR 是 8 位可读写寄存器，控制从深度待机模式返回时的外部存储器控制引脚的状态和启动方法。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	EBUS KEEPE	RAM BOOT	—	—	—	—	—	—
初始值:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R

位	位名	初始值	R/W	说明
7	EBUSKEEPE	0	R/W	外部存储器控制引脚状态的保持 0: 在从深度待机模式返回时，不保持外部存储器控制引脚的状态。 1: 在从深度待机模式返回时，保持外部存储器控制引脚的状态。
6	RAMBOOT	0	R/W	从深度待机返回后的启动方法选择 选择从深度待机返回后的启动方法。 0: 按照复位时设定的引导模式 1: 从用于保持的内部 RAM 读程序 程序计数器 (PC) : 地址 H'1C000000 堆栈指针 (SP) : 地址 H'1C000004
5 ~ 0	—	全 0	R	保留位 读写值都为“0”。

49.2.20 深度待机解除源选择寄存器 (DSSSR)

DSSSR 是 16 位可读写寄存器，由选择用哪种中断源来解除深度待机模式的位构成。可选择实时时钟的闹钟中断和用于解除的引脚（PJ23 ~ PJ20、PG3、PG2、PF19 ~ PF16、PC7、PC5）的变化。与通用输入 / 输出端口的功能设定无关，用于解除的引脚用作深度待机解除源。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PJ23	PJ22	PJ21	PJ20	PG3	PG2	NMI	—	RTCAR	PF19	PF18	PF17	PF16	PC7	PC5
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14	PJ23	0	R/W	通过 PJ23 变化的解除 0: 不通过 PJ23 的变化进行解除 1: 通过 PJ23 的变化进行解除 【注】 只有 SH7269 能使用此位。
13	PJ22	0	R/W	通过 PJ22 变化的解除 0: 不通过 PJ22 的变化进行解除 1: 通过 PJ22 的变化进行解除 【注】 只有 SH7269 能使用此位。
12	PJ21	0	R/W	通过 PJ21 变化的解除 0: 不通过 PJ21 的变化进行解除 1: 通过 PJ21 的变化进行解除 【注】 只有 SH7269 能使用此位。
11	PJ20	0	R/W	通过 PJ20 变化的解除 0: 不通过 PJ20 的变化进行解除 1: 通过 PJ20 的变化进行解除 【注】 只有 SH7269 能使用此位。
10	PG3	0	R/W	通过 PG3 变化的解除 0: 不通过 PG3 的变化进行解除 1: 通过 PG3 的变化进行解除
9	PG2	0	R/W	通过 PG2 变化的解除 0: 不通过 PG2 的变化进行解除 1: 通过 PG2 的变化进行解除
8	NMI	0	R/W	通过 NMI 变化的解除 0: 不通过 NMI 的变化进行解除 1: 通过 NMI 的变化进行解除
7	—	0	R	保留位 读写值都为“0”。
6	RTCAR	0	R/W	通过实时时钟的闹钟中断的解除 0: 不通过实时时钟的闹钟中断进行解除 1: 通过实时时钟的闹钟中断进行解除

位	位名	初始值	R/W	说明
5	PF19	0	R/W	通过 PF19 变化的解除 0: 不通过 PF19 的变化进行解除 1: 通过 PF19 的变化进行解除
4	PF18	0	R/W	通过 PF18 变化的解除 0: 不通过 PF18 的变化进行解除 1: 通过 PF18 的变化进行解除
3	PF17	0	R/W	通过 PF17 变化的解除 0: 不通过 PF17 的变化进行解除 1: 通过 PF17 的变化进行解除
2	PF16	0	R/W	通过 PF16 变化的解除 0: 不通过 PF16 的变化进行解除 1: 通过 PF16 的变化进行解除
1	PC7	0	R/W	通过 PC7 变化的解除 0: 不通过 PC7 的变化进行解除 1: 通过 PC7 的变化进行解除
0	PC5	0	R/W	通过 PC5 变化的解除 0: 不通过 PC5 的变化进行解除 1: 通过 PC5 的变化进行解除

49.2.21 深度待机解除边沿选择寄存器 (DSESR)

DSESR 是 16 位可读写寄存器，由选择引脚（通过 DSSSR 选择深度待机模式解除源的引脚）检测边沿的位构成。与中断控制器的设定无关，此寄存器的设定对解除深度待机模式有效。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PJ23E	PJ22E	PJ21E	PJ20E	PG3E	PG2E	NMIE	—	—	PF19E	PF18E	PF17E	PF16E	PC7E	PC5E
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

位	位名	初始值	R/W	说明
15	—	0	R	保留位 读写值都为“0”。
14	PJ23E	0	R/W	PJ23 边沿检测 0: 在 PJ23 的下降沿进行检测 1: 在 PJ23 的上升沿进行检测 【注】 只有 SH7269 能使用此位。
13	PJ22E	0	R/W	PJ22 边沿检测 0: 在 PJ22 的下降沿进行检测 1: 在 PJ22 的上升沿进行检测 【注】 只有 SH7269 能使用此位。
12	PJ21E	0	R/W	PJ21 边沿检测 0: 在 PJ21 的下降沿进行检测 1: 在 PJ21 的上升沿进行检测 【注】 只有 SH7269 能使用此位。

位	位名	初始值	R/W	说明
11	PJ20E	0	R/W	PJ20 边沿检测 0: 在 PJ20 的下降沿进行检测 1: 在 PJ20 的上升沿进行检测 【注】 只有 SH7269 能使用此位。
10	PG3E	0	R/W	PG3 边沿检测 0: 在 PG3 的下降沿进行检测 1: 在 PG3 的上升沿进行检测
9	PG2E	0	R/W	PG2 边沿检测 0: 在 PG2 的下降沿进行检测 1: 在 PG2 的上升沿进行检测
8	NMIE	0	R/W	NMI 边沿检测 0: 在 NMI 的下降沿进行检测 1: 在 NMI 的上升沿进行检测
7、6	—	全 0	R	保留位 读写值都为“0”。
5	PF19E	0	R/W	PF19 边沿检测 0: 在 PF19 的下降沿进行检测 1: 在 PF19 的上升沿进行检测
4	PF18E	0	R/W	PF18 边沿检测 0: 在 PF18 的下降沿进行检测 1: 在 PF18 的上升沿进行检测
3	PF17E	0	R/W	PF17 边沿检测 0: 在 PF17 的下降沿进行检测 1: 在 PF17 的上升沿进行检测
2	PF16E	0	R/W	PF16 边沿检测 0: 在 PF16 的下降沿进行检测 1: 在 PF16 的上升沿进行检测
1	PC7E	0	R/W	PC7 边沿检测 0: 在 PC7 的下降沿进行检测 1: 在 PC7 的上升沿进行检测
0	PC5E	0	R/W	PC5 边沿检测 0: 在 PC5 的下降沿进行检测 1: 在 PC5 的上升沿进行检测

49.2.22 深度待机解除源标志寄存器 (DSFR)

DSFR 是 16 位可读写寄存器，由确认深度待机模式是被哪种中断源解除的标志以及在解除深度待机模式后解除引脚状态的位构成。如果通过中断（NMI、实时时钟的闹钟中断）或者用于解除的引脚变化解除深度待机模式，就执行上电复位异常处理，但是此寄存器保持以前的值。如果通过上电复位解除深度待机模式，就将此寄存器初始化为“H'0000”。

在转移到深度待机模式之前，需要清除全部标志。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IO KEEP	PJ23F	PJ22F	PJ21F	PJ20F	PG3F	PG2F	NMIF	—	RTC ARF	PF19F	PF18F	PF17F	PF16F	PC7F	PC5F
初始值:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

【注】* 为了清除标志，只能在读“1”后写“0”。

位	位名	初始值	R/W	说明
15	IOKEEP	0	R/(W)*	引脚状态保持的解除 此位在解除深度待机模式时解除引脚状态的保持。 0: 不保持引脚状态 [清除条件] • 在读“1”后写“0” 1: 保持引脚状态 [置位条件] • 当转移到深度待机模式时
14	PJ23F	0	R/(W)*	PJ23 标志 0: PJ23 引脚无变化 1: PJ23 引脚有变化 【注】 只有 SH7269 能使用此位。
13	PJ22F	0	R/(W)*	PJ22 标志 0: PJ22 引脚无变化 1: PJ22 引脚有变化 【注】 只有 SH7269 能使用此位。
12	PJ21F	0	R/(W)*	PJ21 标志 0: PJ21 引脚无变化 1: PJ21 引脚有变化 【注】 只有 SH7269 能使用此位。
11	PJ20F	0	R/(W)*	PJ20 标志 0: PJ20 引脚无变化 1: PJ20 引脚有变化 【注】 只有 SH7269 能使用此位。
10	PG3F	0	R/(W)*	PG3 标志 0: PG3 引脚无变化 1: PG3 引脚有变化
9	PG2F	0	R/(W)*	PG2 标志 0: PG2 引脚无变化 1: PG2 引脚有变化
8	NMIF	0	R/(W)*	NMI 标志 0: NMI 引脚无中断 1: NMI 引脚有中断

位	位名	初始值	R/W	说明
7	—	0	R	保留位 读写值都为“0”。
6	RTCARF	0	R/(W)*	RTCAR 标志 0: 无实时时钟的闹钟中断 1: 有实时时钟的闹钟中断
5	PF19F	0	R/(W)*	PF19 标志 0: PF19 引脚无变化 1: PF19 引脚有变化
4	PF18F	0	R/(W)*	PF18 标志 0: PF18 引脚无变化 1: PF18 引脚有变化
3	PF17F	0	R/(W)*	PF17 标志 0: PF17 引脚无变化 1: PF17 引脚有变化
2	PF16F	0	R/(W)*	PF16 标志 0: PF16 引脚无变化 1: PF16 引脚有变化
1	PC7F	0	R/(W)*	PC7 标志 0: PC7 引脚无变化 1: PC7 引脚有变化
0	PC5F	0	R/(W)*	PC5 标志 0: PC5 引脚无变化 1: PC5 引脚有变化

【注】 * 为了清除标志，只能在读“1”后写“0”。

49.2.23 XTAL 晶体振荡器的增益控制寄存器 (XTALCTR)

XTALCTR 是 8 位可读写寄存器，控制 XTAL 晶体振荡器的增益。在实时时钟使用 XTAL 输入时，如果通过上电复位以外的方法解除软件待机模式或者深度待机模式，XTALCTR 就保持以前的值。在实时时钟未使用 XTAL 输入时，在软件待机模式、深度待机模式中将此寄存器初始化为“H'00”。如果通过上电复位进行解除，也将此寄存器初始化为“H'00”。

【注】 在写此寄存器时，请参照“49.4 使用时的注意事项”。

位:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	GAIN
初始值:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W

位	位名	初始值	R/W	说明
7 ~ 1	—	全 0	R	保留位 读写值都为“0”。
0	GAIN	0	R/W	XTAL 晶体振荡器的增益选择 0: 大增益振荡 1: 小增益振荡

49.3 运行说明

49.3.1 睡眠模式

(1) 向睡眠模式的转移

如果在 STBCR1 的 STBY 位为“0”的状态下执行 SLEEP 指令，就从程序执行状态转移到睡眠模式。在执行 SLEEP 指令后 CPU 停止运行，但是保持 CPU 的寄存器内容。内部外围模块继续运行，继续从 CKIO 引脚输出时钟。

(2) 睡眠模式的解除

通过中断（NMI、IRQ、内部外围）、DMA 地址错误和复位（手动复位、上电复位）解除睡眠模式。

- 通过中断的解除

如果发生 NMI、IRQ 和内部外围各中断，就解除睡眠模式，执行中断异常处理。如果发生的中断优先级小于等于 CPU 的状态寄存器（SR）设定的中断屏蔽级并且此模块禁止内部外围模块引起的中断，就不接受中断请求，也不解除睡眠模式。

- 通过 DMA 地址错误的解除

如果发生 DMA 地址错误，就解除睡眠模式，执行 DMA 地址错误异常处理。

- 通过复位的解除

通过上电复位或者手动复位，解除睡眠模式。

49.3.2 软件待机模式

(1) 向软件待机模式的转移

如果在 STBCR1 的 STBY 位为“1”并且 DEEP 位为“0”的状态下执行 SLEEP 指令，就从程序执行状态转移到软件待机模式。在软件待机模式中，不仅 CPU 停止运行而且时钟和内部外围模块也停止，并且还停止从 CKIO 引脚输出时钟。

保持 CPU 和高速缓存的寄存器内容，内部外围模块的部分寄存器被初始化。有关软件待机模式中的外围模块的寄存器状态，请参照“51.3 各运行模式中的寄存器状态一览表”。

CPU 用 1 个周期写完 STBCR1 并且执行下一条指令。但是，实际上写操作至少需要 1 个周期。因此，为了将 CPU 写到 STBCR1 的值确实地反映到 SLEEP 指令，必须在读 STBCR1 后执行 SLEEP 指令。

转移到软件待机模式的步骤如下所示：

1. 将看门狗定时器的控制寄存器（WTCSR）的 TME 位置“0”，停止看门狗定时器。
2. 将看门狗定时器的计数器（WTCNT）置“0”，并且使 WTCSR 寄存器的 CKS[2:0] 位的设定值大于等于振荡稳定时间。
3. 在将 STBCR1 寄存器的 STBY 位置“1”并且将 DEEP 位置“0”后，读 STBCR 寄存器，然后执行 SLEEP 指令。

(2) 软件待机模式的解除

通过中断（NMI、IRQ）和复位（上电复位）解除软件待机模式，从 CKIO 引脚开始输出时钟。

• 通过中断的解除

如果检测到NMI引脚的下降沿或者上升沿（通过中断控制器的中断控制寄存器0（ICR0）的NMI边沿选择位（NMIE）选择）、IRQ引脚（IRQ7～IRQ0）的下降沿或者上升沿（通过中断控制器的中断控制寄存器1（ICR1）的IRQn检测选择位（IRQn1S～IRQn0S）选择），时钟就开始振荡。只对振荡稳定时间进行计数的振荡稳定计数器（看门狗定时器）提供此时钟。

如果在转移到软件待机模式前超过预先设定到看门狗定时器的控制/状态寄存器（WTCSR）的时钟选择位（CKS[2:0]）的时间，看门狗定时器就发生上溢。根据此上溢的发生判断为时钟已稳定，给整个LSI提供时钟。因此，解除软件待机模式，执行NMI中断异常处理（在IRQ的情况下执行IRQ中断异常处理）。如果发生的中断优先级小于等于CPU的状态寄存器（SR）设定的中断屏蔽级，就不接受中断请求，也不解除软件待机模式。

要通过NMI中断或者IRQ中断解除软件待机模式时，必须设定CKS[2:0]位，使看门狗定时器的上溢周期大于等于振荡稳定时间。

在从检测到中断到解除软件待机模式为止的期间，CKIO引脚的时钟输出的相位可能不稳定。

• 通过复位的解除

如果将RES引脚置为低电平，就解除软件待机模式并且转移到上电复位状态。然后，如果将RES引脚置为高电平，就开始上电复位异常处理。

必须将RES引脚的低电平保持到时钟的振荡稳定为止。

将内部的时钟继续输出到CKIO引脚。

(3) 转移到软件待机模式时的注意事项

通过中断（NMI、IRQ）或复位（手动复位、上电复位）解除软件待机模式，如果SLEEP指令和NMI、IRQ以外的中断同时发生，就可能在接受中断后解除软件待机模式。

在转移到软件待机模式时，必须先设定为不能发生中断，然后执行SLEEP指令。

(4) 解除软件待机模式时的注意事项

在解除软件待机模式后的振荡稳定时间内，CKIO引脚输出不稳定的时钟。为了防止由此引起的误动作，必须更改FRQCR寄存器的bit13和bit12。

49.3.3 软件待机模式的应用例子

对在 NMI 信号的下降沿转移到软件待机模式并且在 NMI 信号的上升沿进行解除的例子进行说明。此例的时序如图 49.1 所示。

在将中断控制寄存器 0 (ICR0) 的 NMI 边沿选择位 (NMIE) 置“0” (下降沿检测) 的状态下, 如果将 NMI 引脚从高电平变为低电平, 就接受 NMI 中断。如果通过 NMI 异常服务程序将 NMIE 位置“1” (上升沿检测) 以及在 STBCR1 的 STBY 位为“1”并且 DEEP 位为“0”的状态下执行 SLEEP 指令, 就转移到软件待机模式。然后, 如果将 NMI 引脚从低电平变为高电平, 就解除软件待机模式。

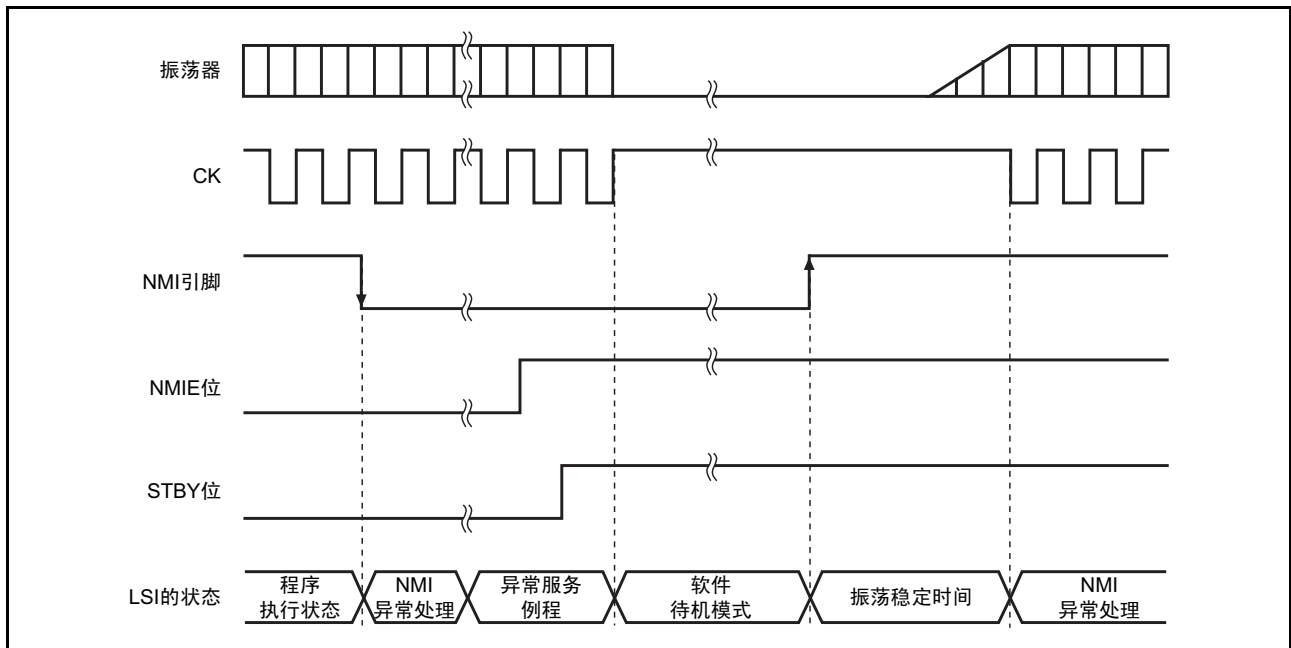


图 49.1 软件待机模式中的 NMI 时序 (应用例子)

49.3.4 深度待机模式

(1) 向深度待机模式的转移

如果在 STBCR1 的 STBY 位和 DEEP 位为“1”的状态下执行 SLEEP 指令，就从程序执行状态转移到深度待机模式。在深度待机模式中，不仅 CPU、时钟和内部外围模块停止运行，而且还切断电源（RRAMKP 寄存器的 RRAMKP3 ~ RRAMKP0 位所设用于保持的内部 RAM 区和实时时钟除外），能大幅度降低功耗。因此，不保持 CPU 和高速缓存的寄存器内容，也不保持内部外围模块的寄存器的相关值。但是，能将引脚的状态保持在转移到深度待机模式前的值。

CPU 用 1 个周期写完 DSFR 并且执行下一条指令。但是，实际上写操作至少需要 1 个周期。因此，为了将 CPU 写到 DSFR 的值确实地反映到 SLEEP 指令，必须在读 DSFR 后执行 SLEEP 指令。

转移到深度待机模式的步骤如下所示，该流程如图 49.2 所示。

1. 对需要保持的用于保持的内部 RAM 区，设定 RRAMKP 寄存器的 RRAMKP3 ~ RRAMKP0 位。将要保持的程序传送到设定为用于保持的内部 RAM 区。
2. 通过 DSCTR 寄存器的 EBUSKEEPE 位和 RAMBOOT 位，设定从深度待机模式返回时的启动方法以及外部存储器控制引脚的状态保持。
3. 在通过中断解除深度待机模式时，设定用哪个引脚或者哪种条件进行解除的 DSSSR 对应位。此时，通过 DSESR 的对应位，设定要解除的引脚的输入信号检测模式。
4. 对用于保持的内部 RAM 各页面，读写任意的同一地址。否则，最后的写数据有可能不被写到用于保持的内部 RAM。之后，如果要写用于保持的内部 RAM，就必须在最后写完用于保持的内部 RAM 后，执行此处理。
5. 将 STBCR1 寄存器的 STBY 位和 DEEP 位置“1”。
6. 在清除 DSFR 寄存器的标志后读 DSFR 寄存器，然后执行 SLEEP 指令。

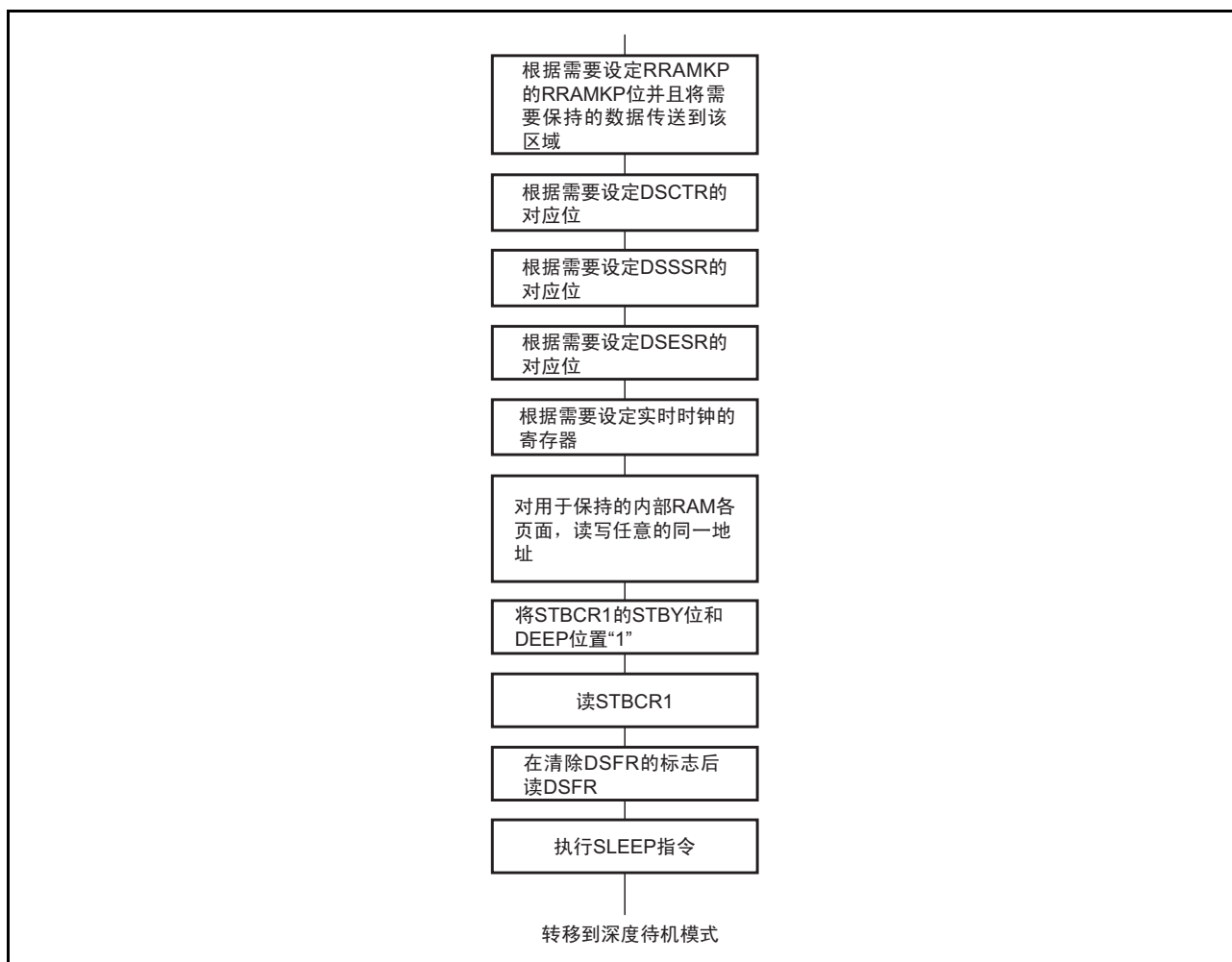


图 49.2 深度待机模式的转移流程

(2) 深度待机模式的解除

通过中断（NMI、实时时钟的闹钟中断）、用于解除的引脚变化和复位（上电复位）解除深度待机模式。与中断优先级和 CPU 的状态寄存器（SR）无关，实时时钟的闹钟中断为深度待机解除源。在通过复位以外的中断源进行解除时，也执行上电复位异常处理而不执行中断异常处理。深度待机模式的解除流程如图 49.3 所示。

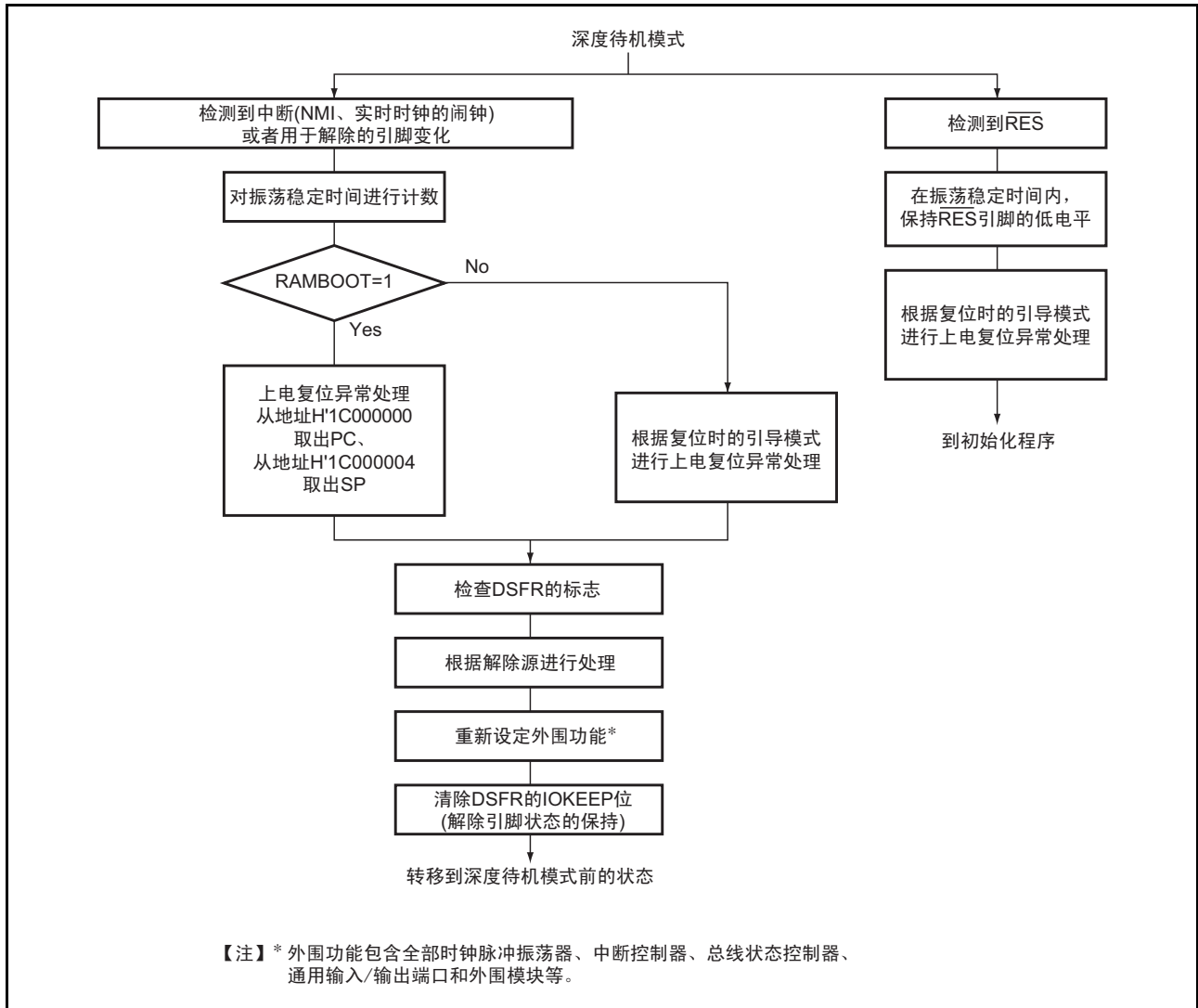


图 49.3 深度待机模式的解除流程

- 不通过复位的解除

如果检测到NMI引脚的下降沿或者上升沿（通过DSESR的对应位进行选择）、用于解除的引脚的下降沿或者上升沿（通过DSESR的对应位进行选择），或者发生实时时钟的闹钟中断（设定方法的详细内容请参照“15.4.4 闹钟功能”），就在经过电源稳定等待时间后时钟开始振荡。在经过振荡稳定时间后解除深度待机模式，执行上电复位异常处理。

在从检测到解除源到解除深度待机模式为止的期间，CKIO引脚的时钟输出的相位有可能不稳定。

NMI引脚、用于解除的引脚变化以及实时时钟的闹钟中断的检测从设定了DSSSR的对应位时开始有效。保持被检测到的解除源，但是在解除深度待机模式后被反映到DSFR。另外，通过CPU接受中断请求，将保持的解除源全部清除。如果在检测到解除源后转移到深度待机模式，就在转移后立即解除深度待机模式。

- 通过复位的解除

如果将RES引脚置为低电平，就解除深度待机模式，转移到上电复位状态。此后，如果将RES引脚置为高电平，就开始上电复位异常处理。如果将RES引脚置为低电平，就开始将内部的时钟输出到CKIO引脚。

必须将RES引脚的低电平保持到时钟的振荡稳定为止。

(3) 解除深度待机模式后的运行

能通过设定DSCTR寄存器的RAMBOOT位，选择是启动外部存储器还是启动用于保持的内部RAM。通过设定EBUSKEEPE位，能在解除深度待机模式后保持外部存储器控制引脚的状态。通过各位解除深度待机模式后的引脚状态如表49.3所示，外部存储器的控制引脚一览表如表49.4所示。

表 49.3 通过设定 DSCTR 寄存器解除深度待机后的引脚状态和启动方法

EBUSKEEPE 位	RAMBOOT 位	启动方法	解除深度待机后的引脚状态
0	0	外部存储器	不保持外部存储器的控制引脚状态。 在清除 IOKEEP 位时，其他引脚解除引脚状态的保持。
0	1	用于保持的内部 RAM	不保持外部存储器的控制引脚状态。 在解除深度待机后，外部存储器控制引脚解除引脚状态的保持。 在清除 IOKEEP 位时，其他引脚解除引脚状态的保持。
1	0	—	禁止设定
1	1	用于保存的内部 RAM	保持外部存储器的控制引脚状态。 在清除 IOKEEP 位时，外部存储器控制引脚和其他引脚都解除引脚状态的保持。

表 49.4 外部存储器的控制引脚一览表

引导模式 0 (CS0 空间: 16 位总线宽度)	引导模式 1 (CS0 空间: 32 位总线宽度)	引导模式 2 (NAND 闪存)	引导模式 3 (串行闪存)	引导模式 4 (SD 主机接口)	引导模式 5 (MMC 主机接口)
A[20:1] D[15:0] CS0、RD、CKIO	A[20:2] D[31:0] CS0、RD、CKIO	NAF[7:0] FRE、FCLE、 FALE、FWE、 FCE、FRB	RSPCK0、SSL00、 MOSI0、MISO0 (只限于 PB17 ~ 20)	SD_CLK0、 SD_CMD0、 SD_D[3:0]0	MMC_CLK、 MMC_CMD、 MMC_D[3:0]

如果通过中断（NMI、实时时钟的闹钟）和用于解除的引脚变化来解除深度待机模式，就能通过深度待机解除源标志寄存器（DSFR）确认用哪种解除源进行解除。

在转移到深度待机模式时引脚保持之前的状态，但是在外部存储器启动模式中，在解除深度待机模式后解除外部存储器控制引脚的状态保持，并且能取程序。其他引脚在解除深度待机模式后也继续保持引脚的状态，直到在读到 DSFR 寄存器的 IOKEEP 位的“1”后写“0”为止。在用于保持的内部 RAM 启动模式中，外部存储器控制引脚和其他引脚都在解除深度待机模式后继续保持引脚的状态，直到在读到 DSFR 寄存器的 IOKEEP 位的“1”后写“0”为止。要设定为转移到深度待机模式前的状态时，需要重新设定外围功能。外围功能包括全部时钟脉冲振荡器、中断控制器、通用输入 / 输出端口和外围模块等。在重新设定后，通过在读 IOKEEP 位的“1”后写“0”，解除引脚的状态保持，变为转移到深度待机模式前的状态。

(4) 转移到深度待机模式时的注意事项

如果设定并且输入多个解除源，就将多个解除源标志置位。

49.3.5 模块待机功能

(1) 向模块待机功能的转移

能通过将待机控制寄存器的各 MSTP 位置“1”，停止给各自对应的内部外围模块提供时钟。能通过使用此功能降低程序执行的状态和睡眠模式中的功耗。在转移前，必须在将此模块设定为禁止状态后设定为模块待机状态。另外，不能对模块待机状态的模块进行寄存器存取等。

寄存器的状态请参照“51.3 各运行模式中的寄存器状态一览表”。

(2) 模块待机功能的解除

通过将各 MSTP 位清“0”或者上电复位（只限于实时时钟、用户调试接口、直接存储器存取控制器）解除模块待机功能。在通过将各 MSTP 位清“0”解除模块待机功能时，必须在读该位后确认已被清“0”。

49.3.6 XTAL 晶体振荡器的增益调整功能

能通过更改 XTAL 晶体振荡器增益控制寄存器的 GAIN 位，更改振荡器的增益。在更改增益时需要 PLL 稳定时间。通过内部看门狗定时器对稳定时间进行计数。

1. 在初始状态下，振荡器的增益为大增益。
2. 为了达到指定的振荡稳定时间，给看门狗定时器设定值，并且停止看门狗定时器的运行。需要进行以下的设定：

WTCSR.TME=0: 看门狗定时器的停止

WTCSR.CKS[2:0]: 看门狗定时器计数时钟的分频率

WTCNT: 计数器的初始值

（看门狗定时器的递增计数器通过设定后的时钟进行递增计数。）

3. 将 GAIN 设定为目的值。
4. 本 LSI 内部暂时停止运行而开始看门狗定时器的递增计数。只给看门狗定时器提供时钟，停止其他的内部时钟，并且继续将不稳定的时钟输出到 CKIO 引脚。为了防止由此引起的误动作，必须更改 FRQCR 寄存器的 CKOEN2 位。此状态和软件待机模式的状态一样，通过模块对寄存器进行初始化，详细内容请参照“51.3 各运行模式中的寄存器状态一览表”。
5. 开始提供由看门狗定时器的计数上溢设定的时钟，本 LSI 重新开始运行。看门狗定时器在上溢后停止运行。

49.4 使用时的注意事项

49.4.1 设定寄存器时的注意

在写低功耗模式的相关寄存器时，必须注意以下事项：

在 CPU 写低功耗模式的相关寄存器时，CPU 在执行写指令后不等待实际的寄存器写操作结束而执行后续指令。

如果在执行后续指令时想反映写寄存器后的变更内容，就必须在写寄存器的指令和后续指令之间虚读相同的寄存器。

49.4.2 未使用实时时钟时的注意

在不使用实时时钟的情况下，必须在设定实时时钟的寄存器后将 STBCR3 寄存器的 MSTP30 位置“1”。详细内容请参照“49.2.3 待机控制寄存器 3 (STBCR3)”。

50. 用户调试接口

为了支持边界扫描功能和仿真器，本 LSI 内置了用户调试接口。

50.1 特点

用户调试接口是对应 JTAG（Joint Test Action Group, IEEE Std.1149.1 and IEEE Standard Test Access Port and Boundary-Scan Architecture）的串行输入 / 输出接口。

此模块内置边界扫描 TAP 控制器和控制用户调试接口中断功能等的仿真 TAP 控制器。包括在接通电源时，通过将 $\overline{\text{TRST}}$ 引脚置为有效，选择边界扫描 TAP 控制器；通过输入仿真 TAP 控制器切换命令，选择仿真 TAP 控制器。要从仿真 TAP 控制器切换到边界扫描 TAP 控制器时，需要将 $\overline{\text{TRST}}$ 引脚置为有效。

在 ASE 模式中选择仿真 TAP 控制器。有关连接仿真器的方法，请参照仿真器的手册。

框图如图 50.1 所示。

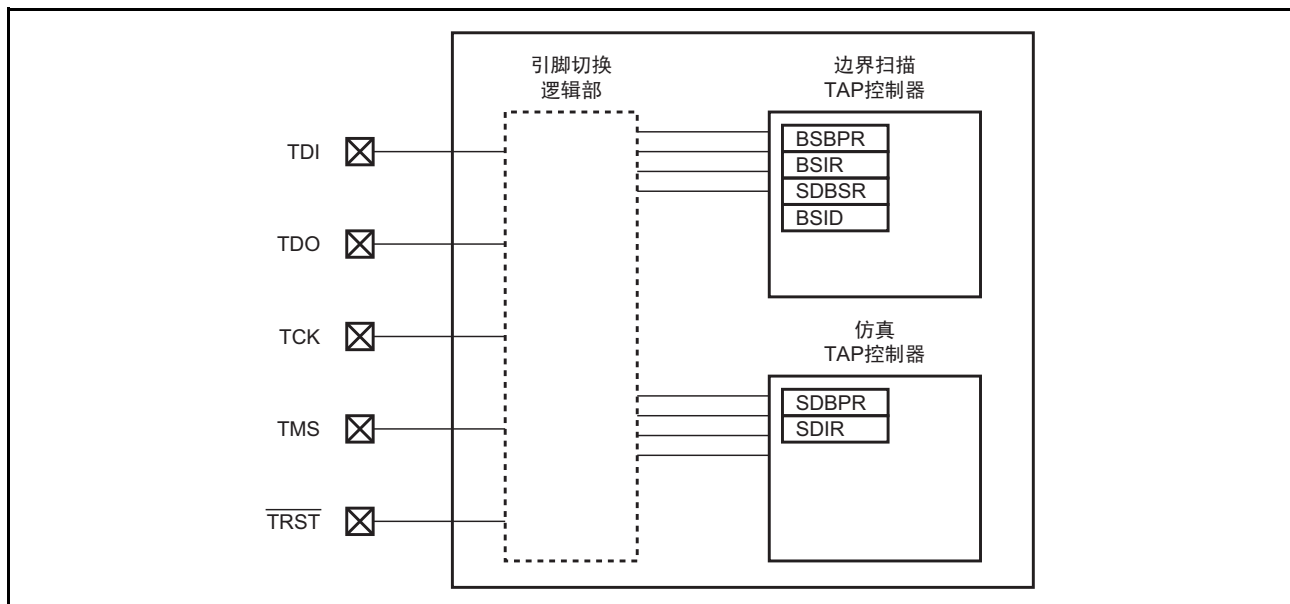


图 50.1 框图

50.2 输入 / 输出引脚

表 50.1 引脚结构

名称	引脚名	输入 / 输出	功能
串行数据输入 / 输出时钟引脚	TCK	输入	与此时钟同步，将数据从数据输入引脚（TDI）串行输入到此模块，并且从数据输出引脚（TDO）输出数据。
模式选择输入引脚	TMS	输入	与 TCK 同步，通过改变此信号决定 TAP 控制电路的状态。协议符合 JTAG 规格（IEEE Std.1149.1）。
复位输入引脚	$\overline{\text{TRST}}$	输入	不与 TCK 同步接受输入，用低电平对此模块进行复位。与是否使用此模块功能无关，必须在接通电源时将 $\overline{\text{TRST}}$ 置为一定期间的低电平。有关复位结构的详细内容，请参照“50.5.2 复位结构”。
串行数据输入引脚	TDI	输入	与 TCK 同步，通过改变此引脚信号，将数据发送到此模块。
串行数据输出引脚	TDO	输出	与 TCK 同步，通过读此引脚，从此模块读数据。数据输出时序的初始值是与下降沿同步，但是能通过给 SDIR 输入“TDO 变化时序切换”命令更改为与上升沿同步。详细内容请参照“50.5.3 TDO 输出时序”。
ASE 模式选择引脚	$\overline{\text{ASEMD}}^*$	输入	在 RES 引脚有效期间，如果给 $\overline{\text{ASEMD}}$ 引脚输入低电平，就为 ASE 模式；如果输入高电平，就为产品芯片模式。在 ASE 模式中，能使用仿真器专用的功能。 $\overline{\text{ASEMD}}$ 引脚的输入电平必须在 RES 引脚无效后至少保持 1 个周期。

【注】 * 在不使用仿真器时，必须固定为高电平。

50.3 边界扫描 TAP 控制器的寄存器说明

边界扫描 TAP 控制器有以下寄存器。

表 50.2 边界扫描 TAP 控制器的寄存器结构

寄存器名	略称	R/W	初始值	地址	存取长度
旁路寄存器	BSBPR	—	—	—	—
指令寄存器	BSIR	—	H'4	—	—
边界扫描寄存器	SDBSR	—	—	—	—
ID 寄存器	BSID	—	H'080C6447	—	—

50.3.1 旁路寄存器（BSBPR）

BSBPR 是 CPU 不能存取的 1 位寄存器。如果将 BSIR 设定为 BYPASS 模式，就将 BSBPR 连接在 TDI 引脚和 TDO 引脚之间。初始值为不定值。

50.3.2 指令寄存器 (BSIR)

BSIR是4位寄存器，在TRST有效时或者在TAP的Test-Logic-Reset状态下被初始化。CPU不能存取此寄存器。

位	位名	初始值	R/W	说明
3 ~ 0	TI[3:0]	0100	—	测试指令 通过 TDI 的串行输入，将此模块的指令传送到 BSIR。命令请参照表 50.3。

表 50.3 边界扫描 TAP 控制器支持的命令

bit3 ~ 0				说明
TI3	TI2	TI1	TI0	
0	0	0	0	EXTEST
0	0	0	1	SAMPLE/PRELOAD
0	0	1	1	仿真 TAP 控制器切换命令
0	1	0	0	IDCODE (初始值)
0	1	1	0	CLAMP
0	1	1	1	HIGHZ
1	1	1	1	BYPASS
上述以外				保留

50.3.3 边界扫描寄存器 (SDBSR)

SDBSR 是配置在 PAD 上的移位寄存器，用于控制本 LSI 的输入 / 输出引脚。CPU 不能存取此寄存器，初始值为不定值。

能使用 EXTEST、SAMPLE/PRELOAD、CLAMP、HIGHZ 命令，进行符合 JTAG 规格的边界扫描测试。本 LSI 的引脚和边界扫描寄存器的对应关系如表 50.4 所示。

表 50.4 本 LSI 的引脚和边界扫描寄存器的对应关系

SH7268 位号	SH7269 位号	引脚名 *1	类型	SH7268 位号	SH7269 位号	引脚名 *1	类型	SH7268 位号	SH7269 位号	引脚名 *1	类型
From TDI				411	411	PG5	OUTPUT	395	395	PG8	CONTROL
426	426	PG0	OUTPUT	410	410	PG5	CONTROL	394	394	PG8	INPUT
425	425	PG0	CONTROL	409	409	PG5	INPUT	—	393	PJ2	OUTPUT
424	424	PG0	INPUT	408	408	PG6	OUTPUT	—	392	PJ2	CONTROL
423	423	PG1	OUTPUT	407	407	PG6	CONTROL	—	391	PJ2	INPUT
422	422	PG1	CONTROL	406	406	PG6	INPUT	—	390	PJ3	OUTPUT
421	421	PG1	INPUT	405	405	PG7	OUTPUT	—	389	PJ3	CONTROL
420	420	PG2	OUTPUT	404	404	PG7	CONTROL	—	388	PJ3	INPUT
419	419	PG2	CONTROL	403	403	PG7	INPUT	—	387	PJ4	OUTPUT
418	418	PG2	INPUT	—	402	PJ0	OUTPUT	—	386	PJ4	CONTROL
417	417	PG3	OUTPUT	—	401	PJ0	CONTROL	—	385	PJ4	INPUT
416	416	PG3	CONTROL	—	400	PJ0	INPUT	384	384	PG9	OUTPUT
415	415	PG3	INPUT	—	399	PJ1	OUTPUT	383	383	PG9	CONTROL
414	414	PG4	OUTPUT	—	398	PJ1	CONTROL	382	382	PG9	INPUT
413	413	PG4	CONTROL	—	397	PJ1	INPUT	381	381	PG10	OUTPUT
412	412	PG4	INPUT	396	396	PG8	OUTPUT	380	380	PG10	CONTROL

SH7268 位号	SH7269 位号	引脚名 *1	类型	SH7268 位号	SH7269 位号	引脚名 *1	类型	SH7268 位号	SH7269 位号	引脚名 *1	类型
379	379	PG10	INPUT	333	333	PG21	OUTPUT	287	287	PF8	CONTROL
378	378	PG11	OUTPUT	332	332	PG21	CONTROL	286	286	PF8	INPUT
377	377	PG11	CONTROL	331	331	PG21	INPUT	285	285	PF9	OUTPUT
376	376	PG11	INPUT	330	330	PG22	OUTPUT	284	284	PF9	CONTROL
375	375	PG12	OUTPUT	329	329	PG22	CONTROL	283	283	PF9	INPUT
374	374	PG12	CONTROL	328	328	PG22	INPUT	282	282	PF10	OUTPUT
373	373	PG12	INPUT	327	327	PG23	OUTPUT	281	281	PF10	CONTROL
372	372	PG13	OUTPUT	326	326	PG23	CONTROL	280	280	PF10	INPUT
371	371	PG13	CONTROL	325	325	PG23	INPUT	279	279	PF11	OUTPUT
370	370	PG13	INPUT	324	324	PG24	OUTPUT	278	278	PF11	CONTROL
369	369	PG14	OUTPUT	323	323	PG24	CONTROL	277	277	PF11	INPUT
368	368	PG14	CONTROL	322	322	PG24	INPUT	276	276	PF12	OUTPUT
367	367	PG14	INPUT	321	321	PG25	OUTPUT	275	275	PF12	CONTROL
366	366	PG15	OUTPUT	320	320	PG25	CONTROL	274	274	PF12	INPUT
365	365	PG15	CONTROL	319	319	PG25	INPUT	273	273	PF13	OUTPUT
364	364	PG15	INPUT	318	318	PG26	OUTPUT	272	272	PF13	CONTROL
363	363	PG16	OUTPUT	317	317	PG26	CONTROL	271	271	PF13	INPUT
362	362	PG16	CONTROL	316	316	PG26	INPUT	270	270	PF14	OUTPUT
361	361	PG16	INPUT	315	315	PG27	OUTPUT	269	269	PF14	CONTROL
—	360	PJ5	OUTPUT	314	314	PG27	CONTROL	268	268	PF14	INPUT
—	359	PJ5	CONTROL	313	313	PG27	INPUT	267	267	PF15	OUTPUT
—	358	PJ5	INPUT	312	312	PF0	OUTPUT	266	266	PF15	CONTROL
—	357	PJ6	OUTPUT	311	311	PF0	CONTROL	265	265	PF15	INPUT
—	356	PJ6	CONTROL	310	310	PF0	INPUT	—	264	PJ10	OUTPUT
—	355	PJ6	INPUT	309	309	PF1	OUTPUT	—	263	PJ10	CONTROL
354	354	PG17	OUTPUT	308	308	PF1	CONTROL	—	262	PJ10	INPUT
353	353	PG17	CONTROL	307	307	PF1	INPUT	261	261	PF16	OUTPUT
352	352	PG17	INPUT	306	306	PF2	OUTPUT	260	260	PF16	CONTROL
—	351	PJ7	OUTPUT	305	305	PF2	CONTROL	259	259	PF16	INPUT
—	350	PJ7	CONTROL	304	304	PF2	INPUT	258	258	PF17	OUTPUT
—	349	PJ7	INPUT	303	303	PF3	OUTPUT	257	257	PF17	CONTROL
—	348	PJ8	OUTPUT	302	302	PF3	CONTROL	256	256	PF17	INPUT
—	347	PJ8	CONTROL	301	301	PF3	INPUT	255	255	PF18	OUTPUT
—	346	PJ8	INPUT	300	300	PF4	OUTPUT	254	254	PF18	CONTROL
—	345	PJ9	OUTPUT	299	299	PF4	CONTROL	253	253	PF18	INPUT
—	344	PJ9	CONTROL	298	298	PF4	INPUT	—	252	PJ11	OUTPUT
—	343	PJ9	INPUT	297	297	PF5	OUTPUT	—	251	PJ11	CONTROL
342	342	PG18	OUTPUT	296	296	PF5	CONTROL	—	250	PJ11	INPUT
341	341	PG18	CONTROL	295	295	PF5	INPUT	—	249	PJ12	OUTPUT
340	340	PG18	INPUT	294	294	PF6	OUTPUT	—	248	PJ12	CONTROL
339	339	PG19	OUTPUT	293	293	PF6	CONTROL	—	247	PJ12	INPUT
338	338	PG19	CONTROL	292	292	PF6	INPUT	—	246	PJ13	OUTPUT
337	337	PG19	INPUT	291	291	PF7	OUTPUT	—	245	PJ13	CONTROL
336	336	PG20	OUTPUT	290	290	PF7	CONTROL	—	244	PJ13	INPUT
335	335	PG20	CONTROL	289	289	PF7	INPUT	243	243	PF19	OUTPUT
334	334	PG20	INPUT	288	288	PF8	OUTPUT	242	242	PF19	CONTROL

SH7268 位号	SH7269 位号	引脚名 *1	类型	SH7268 位号	SH7269 位号	引脚名 *1	类型	SH7268 位号	SH7269 位号	引脚名 *1	类型
241	241	PF19	INPUT	195	195	PD7	OUTPUT	149	149	PC7	CONTROL
240	240	PF20	OUTPUT	194	194	PD7	CONTROL	148	148	PC7	INPUT
239	239	PF20	CONTROL	193	193	PD7	INPUT	147	147	PC8	OUTPUT
238	238	PF20	INPUT	192	192	PD8	OUTPUT	146	146	PC8	CONTROL
237	237	PF21	OUTPUT	191	191	PD8	CONTROL	145	145	PC8	INPUT
236	236	PF21	CONTROL	190	190	PD8	INPUT	144	144	PB1	OUTPUT
235	235	PF21	INPUT	189	189	PD9	OUTPUT	143	143	PB1	CONTROL
234	234	PF22	OUTPUT	188	188	PD9	CONTROL	142	142	PB1	INPUT
233	233	PF22	CONTROL	187	187	PD9	INPUT	141	141	PB2	OUTPUT
232	232	PF22	INPUT	186	186	PD10	OUTPUT	140	140	PB2	CONTROL
231	231	PF23	OUTPUT	185	185	PD10	CONTROL	139	139	PB2	INPUT
230	230	PF23	CONTROL	184	184	PD10	INPUT	138	138	PB3	OUTPUT
229	229	PF23	INPUT	183	183	PD11	OUTPUT	137	137	PB3	CONTROL
228	228	PD0	OUTPUT	182	182	PD11	CONTROL	136	136	PB3	INPUT
227	227	PD0	CONTROL	181	181	PD11	INPUT	—	135	PJ14	OUTPUT
226	226	PD0	INPUT	180	180	PD12	OUTPUT	—	134	PJ14	CONTROL
—	225	PJ24	OUTPUT	179	179	PD12	CONTROL	—	133	PJ14	INPUT
—	224	PJ24	CONTROL	178	178	PD12	INPUT	—	132	PJ15	OUTPUT
—	223	PJ24	INPUT	177	177	PD13	OUTPUT	—	131	PJ15	CONTROL
222	222	PD1	OUTPUT	176	176	PD13	CONTROL	—	130	PJ15	INPUT
221	221	PD1	CONTROL	175	175	PD13	INPUT	129	129	PB4	OUTPUT
220	220	PD1	INPUT	174	174	PD14	OUTPUT	128	128	PB4	CONTROL
219	219	PD2	OUTPUT	173	173	PD14	CONTROL	127	127	PB4	INPUT
218	218	PD2	CONTROL	172	172	PD14	INPUT	—	126	PJ16	OUTPUT
217	217	PD2	INPUT	171	171	PD15	OUTPUT	—	125	PJ16	CONTROL
216	216	PD3	OUTPUT	170	170	PD15	CONTROL	—	124	PJ16	INPUT
215	215	PD3	CONTROL	169	169	PD15	INPUT	—	123	PJ17	OUTPUT
214	214	PD3	INPUT	168	168	PC1	OUTPUT	—	122	PJ17	CONTROL
—	213	PJ25	OUTPUT	167	167	PC1	CONTROL	—	121	PJ17	INPUT
—	212	PJ25	CONTROL	166	166	PC1	INPUT	—	120	PJ18	OUTPUT
—	211	PJ25	INPUT	165	165	PC2	OUTPUT	—	119	PJ18	CONTROL
—	210	PJ26	OUTPUT	164	164	PC2	CONTROL	—	118	PJ18	INPUT
—	209	PJ26	CONTROL	163	163	PC2	INPUT	117	117	PB5	OUTPUT
—	208	PJ26	INPUT	162	162	PC3	OUTPUT	116	116	PB5	CONTROL
—	207	PJ27	OUTPUT	161	161	PC3	CONTROL	115	115	PB5	INPUT
—	206	PJ27	CONTROL	160	160	PC3	INPUT	114	114	PB6	OUTPUT
—	205	PJ27	INPUT	159	159	PC4	OUTPUT	113	113	PB6	CONTROL
204	204	PD4	OUTPUT	158	158	PC4	CONTROL	112	112	PB6	INPUT
203	203	PD4	CONTROL	157	157	PC4	INPUT	111	111	PB7	OUTPUT
202	202	PD4	INPUT	156	156	PC5	OUTPUT	110	110	PB7	CONTROL
201	201	PD5	OUTPUT	155	155	PC5	CONTROL	109	109	PB7	INPUT
200	200	PD5	CONTROL	154	154	PC5	INPUT	108	108	PB8	OUTPUT
199	199	PD5	INPUT	153	153	PC6	OUTPUT	107	107	PB8	CONTROL
198	198	PD6	OUTPUT	152	152	PC6	CONTROL	106	106	PB8	INPUT
197	197	PD6	CONTROL	151	151	PC6	INPUT	105	105	PB9	OUTPUT
196	196	PD6	INPUT	150	150	PC7	OUTPUT	104	104	PB9	CONTROL

SH7268 位号	SH7269 位号	引脚名 *1	类型	SH7268 位号	SH7269 位号	引脚名 *1	类型	SH7268 位号	SH7269 位号	引脚名 *1	类型
103	103	PB9	INPUT	68	68	PB16	CONTROL	—	33	PJ30	OUTPUT
102	102	PB10	OUTPUT	67	67	PB16	INPUT	—	32	PJ30	CONTROL
101	101	PB10	CONTROL	66	66	PB17	OUTPUT	—	31	PJ30	INPUT
100	100	PB10	INPUT	65	65	PB17	CONTROL	—	30	PJ31	OUTPUT
99	99	PB11	OUTPUT	64	64	PB17	INPUT	—	29	PJ31	CONTROL
98	98	PB11	CONTROL	63	63	PB18	OUTPUT	—	28	PJ31	INPUT
97	97	PB11	INPUT	62	62	PB18	CONTROL	27	27	PE0	OUTPUT*2
96	96	PB12	OUTPUT	61	61	PB18	INPUT	26	26	PE1	OUTPUT*2
95	95	PB12	CONTROL	60	60	PB19	OUTPUT	25	25	PE0	INPUT
94	94	PB12	INPUT	59	59	PB19	CONTROL	24	24	PE1	INPUT
—	93	PJ19	OUTPUT	58	58	PB19	INPUT	23	23	PE2	OUTPUT*2
—	92	PJ19	CONTROL	57	57	PB20	OUTPUT	22	22	PE3	OUTPUT*2
—	91	PJ19	INPUT	56	56	PB20	CONTROL	21	21	PE2	INPUT
—	90	PJ20	OUTPUT	55	55	PB20	INPUT	20	20	PE3	INPUT
—	89	PJ20	CONTROL	54	54	PB21	OUTPUT	—	19	PE4	OUTPUT*2
—	88	PJ20	INPUT	53	53	PB21	CONTROL	—	18	PE5	OUTPUT*2
87	87	PB13	OUTPUT	52	52	PB21	INPUT	—	17	PE4	INPUT
86	86	PB13	CONTROL	51	51	PB22	OUTPUT	—	16	PE5	INPUT
85	85	PB13	INPUT	50	50	PB22	CONTROL	—	15	PE6	OUTPUT*2
—	84	PJ21	OUTPUT	49	49	PB22	INPUT	—	14	PE7	OUTPUT*2
—	83	PJ21	CONTROL	48	48	PC0	OUTPUT	—	13	PE6	INPUT
—	82	PJ21	INPUT	47	47	PC0	CONTROL	—	12	PE7	INPUT
—	81	PJ22	OUTPUT	46	46	PC0	INPUT	11	11	NMI	INPUT
—	80	PJ22	CONTROL	45	45	PA0	OUTPUT	10	10	PH0	INPUT
—	79	PJ22	INPUT	44	44	PA0	CONTROL	9	9	PH1	INPUT
—	78	PJ23	OUTPUT	43	43	PA0	INPUT	8	8	PH2	INPUT
—	77	PJ23	CONTROL	42	42	PA1	OUTPUT	7	7	PH3	INPUT
—	76	PJ23	INPUT	41	41	PA1	CONTROL	6	6	PH4	INPUT
75	75	PB14	OUTPUT	40	40	PA1	INPUT	5	5	PH5	INPUT
74	74	PB14	CONTROL	—	39	PJ28	OUTPUT	—	4	PH6	INPUT
73	73	PB14	INPUT	—	38	PJ28	CONTROL	—	3	PH7	INPUT
72	72	PB15	OUTPUT	—	37	PJ28	INPUT	2	2	ASEBRKAKN /ASEBRK	OUTPUT
71	71	PB15	CONTROL	—	36	PJ29	OUTPUT	1	1	ASEBRKAKN /ASEBRK	CONTROL
70	70	PB15	INPUT	—	35	PJ29	CONTROL	0	0	ASEBRKAKN /ASEBRK	INPUT
69	69	PB16	OUTPUT	—	34	PJ29	INPUT	To TDO			

【注】 *1 这是“1.6 引脚一览表”的功能 1 的引脚功能名。

*2 这是漏极开路引脚。在置为低电平时输出低电平，在置为高电平时为 Hi-Z。

*3 CONTROL 为低电平有效。在将 CONTROL 置为低电平时，将对应引脚输出到 OUT 值。

50.3.4 ID 寄存器 (BSID)

BSID是CPU不能存取的32位寄存器。当给BSIR设定了IDCODE的命令时，能从引脚读BSID，不能写BSID。

位:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DID[31:16]															
初始值:	0	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0
R/W:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DID[15:0]															
初始值:	0	1	1	0	0	1	0	0	0	1	0	0	0	1	1	1
R/W:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

位	位名	初始值	R/W	说明
31 ~ 0	DID[31:0]	H'080C6447	—	设备 这是 JTAG 规定的 ID 寄存器。在本 LSI 中为 “H'080C6447”。但是，高 4 位有可能因芯片的版本而被更改。

50.4 仿真 TAP 控制器的寄存器说明

为了使用仿真 TAP 控制器，需要将仿真 TAP 控制器的切换命令输入到边界扫描 TAP 控制器的 BSIR 寄存器。仿真 TAP 控制器有以下寄存器。

表 50.5 仿真 TAP 控制器的寄存器结构

寄存器名	略称	R/W	初始值	地址	存取长度
旁路寄存器	SDBPR	—	—	—	—
指令寄存器	SDIR	R	H'EFFD	H'FFFE2000	16

50.4.1 旁路寄存器 (SDBPR)

SDBPR 是 CPU 不能存取的 1 位寄存器。如果将 SDIR 设定为 BYPASS 模式，就将 SDBPR 连接在 TDI 引脚和 TDO 引脚之间。初始值为不定值。

50.4.2 指令寄存器 (SDIR)

SDIR 是 16 位只读寄存器，在 $\overline{\text{TRST}}$ 有效时或者在 TAP 的 Test-Logic-Reset 状态下被初始化。与 CPU 的模式无关，此模块能写此寄存器。不保证给此寄存器设定了保留命令时的运行。初始值为“H'EFFD”。

位:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TI[7:0]								—	—	—	—	—	—	—	—
初始值:	1*	1*	1*	0*	1*	1*	1*	1*	1	1	1	1	1	1	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

【注】* TI[7:0]的初始值是保留值。在设定命令时，不能设定保留值。

位	位名	初始值	R/W	说明
15 ~ 8	TI[7:0]	11101111 *	R	测试指令 通过 TDI 的串行输入，将此模块的指令传送到 SDIR。 命令请参照表 50.6。
7 ~ 2	—	全 1	R	保留位 读取值总是“1”。
1	—	0	R	保留位 读取值总是“0”。
0	—	1	R	保留位 读取值总是“1”。

表 50.6 仿真 TAP 控制器支持的命令

位 15 ~ 8								说明
TI7	TI6	TI5	TI4	TI3	TI2	TI1	TI0	
0	1	1	0	—	—	—	—	用户调试接口复位无效
0	1	1	1	—	—	—	—	用户调试接口复位有效
1	0	0	1	1	1	0	0	TDO 变化时序切换
1	0	1	1	—	—	—	—	用户调试接口中断
1	1	1	1	—	—	—	—	BYPASS
上述以外								保留

50.5 运行说明

50.5.1 TAP 控制器

TAP 控制器的内部状态如图 50.2 所示，符合 JTAG 规定的状态转移。

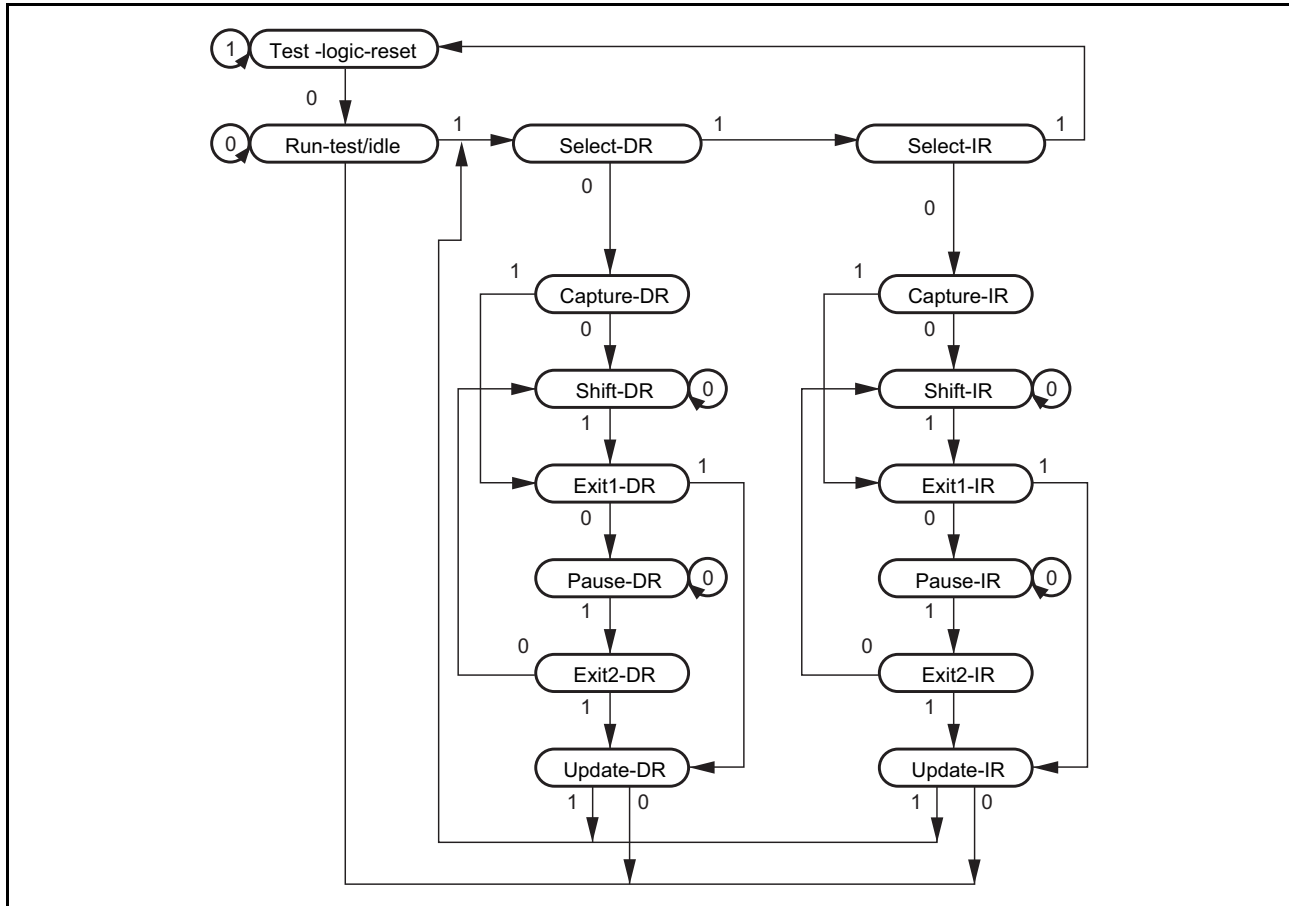


图 50.2 TAP 控制器的状态转移图

【注】 转移条件为 TCK 的上升沿的 TMS 值。在 TCK 的上升沿对 TDI 值进行采样并且在 TCK 的下降沿进行移位。有关 TDO 值的变化时序，请参照“50.5.3 TDO 输出时序”。TDO 在 Shift-DR 和 Shift-IR 以外的状态下为高阻抗状态。在 $\overline{\text{TRST}}$ 有效时，不与 TCK 同步而转移到 Test-Logic-Reset 状态。

50.5.2 复位结构

表 50.7 复位结构

$\overline{\text{ASEMD}}^{*1}$	$\overline{\text{RES}}$	$\overline{\text{TRST}}$	芯片状态
H	L	L	上电复位和此模块的复位
		H	上电复位
	H	L	只对此模块进行复位。
		H	正常运行
L	L	L	复位保持 *2
		H	上电复位
	H	L	只对此模块进行复位。
		H	正常运行

【注】 *1 选择产品芯片模式和 ASE 模式的设定。

当 $\overline{\text{ASEMD}}$ 为高电平时，为产品芯片模式。

当 $\overline{\text{ASEMD}}$ 为低电平时，为 ASE 模式。

*2 在 ASE 模式中 $\overline{\text{RES}}$ 无效时，如果 $\overline{\text{TRST}}$ 引脚为低电平，就为复位保持状态。在此状态下，不启动 CPU。此后，如果将 $\overline{\text{TRST}}$ 置为高电平，此模块的运行就为有效，但是不启动 CPU。通过上电复位解除复位保持状态。

50.5.3 TDO 输出时序

选择仿真 TAP 控制器时的 TDO 变化时序的初始值是与 TCK 的下降沿同步输出。但是，通过从引脚给 SDIR 设定 TDO 变化时序切换命令并且通过 Update-IR 状态，使 TDO 的变化时序与 TCK 的上升沿同步。此命令不影响边界扫描 TAP 控制器的 TDO 输出时序。

在设定 TDO 变化时序切换命令后，如果要使 TDO 的变化时序与 TCK 的下降沿同步输出，就需要在进行上电复位的同时将 $\overline{\text{TRST}}$ 引脚置为有效。在由 $\overline{\text{RES}}$ 引脚进行上电复位时，即使在 $\overline{\text{RES}}$ 引脚无效后芯片内部的同步复位还会工作一定的时间。因此，如果在 $\overline{\text{RES}}$ 引脚无效后立即将 $\overline{\text{TRST}}$ 引脚置为有效，就清除 TDO 变化时序切换命令，并且使 TDO 的变化时序与 TCK 下降沿同步输出。为了防止此现象， $\overline{\text{RES}}$ 引脚和 $\overline{\text{TRST}}$ 引脚之间的信号变化必须至少有 20tcyc 的间隔。

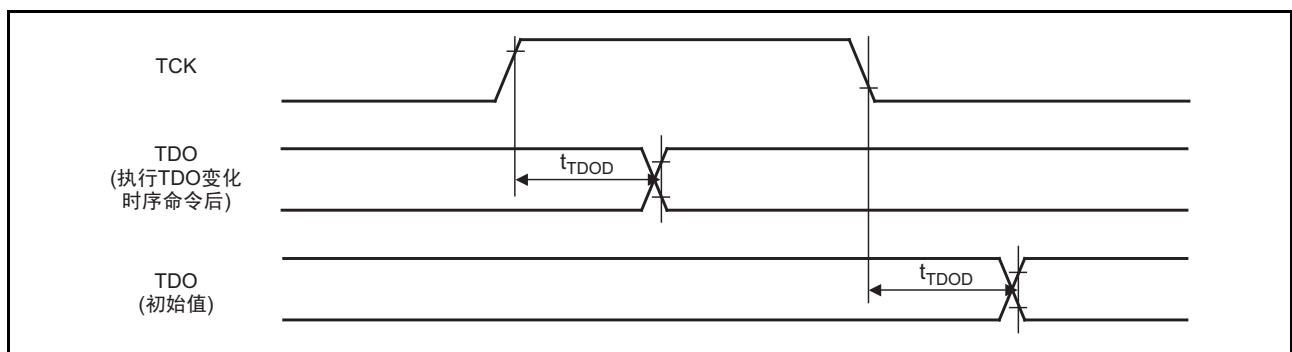


图 50.3 数据传送时序

50.5.4 用户调试接口复位

通过给 **SDIR** 设定用户调试接口复位有效命令，进行用户调试接口复位。用户调试接口复位是和上电复位一样的复位。通过设定用户调试接口复位无效命令，解除用户调试接口复位。用户调试接口复位有效命令和用户调试接口复位无效命令之间需要的时间，和上电复位时将 $\overline{\text{RES}}$ 引脚保持为低电平的时间相同。

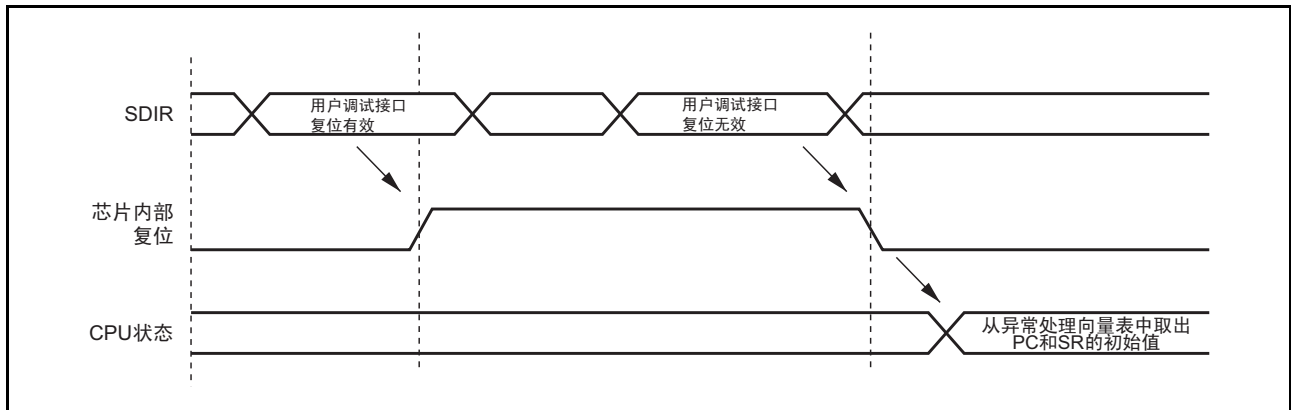


图 50.4 用户调试接口复位

50.5.5 用户调试接口中断

用户调试接口中断功能通过给 **SDIR** 设定此模块的命令来产生中断。用户调试接口中断为一般异常 / 中断运行，从异常处理向量表中取出异常服务程序的起始地址，跳转到该地址并执行。此中断请求的优先级固定为 15。

即使在睡眠模式中也接受用户调试接口中断，但是在软件待机模式中不接受用户调试接口中断。

50.6 边界扫描

通过从此模块给 **BSIR** 设定命令，将引脚设定为 JTAG 规定的边界扫描模式。

50.6.1 支持的命令

本 LSI 支持 JTAG 规格定义的 3 个必要命令 (**BYPASS**、**SAMPLE/PRELOAD**、**EXTEST**) 和 3 个选项命令 (**IDCODE**、**CLAMP**、**HIGHZ**)。

(1) **BYPASS**

BYPASS 指令是运行旁路寄存器所必须的标准指令，用于缩短移位路径并且将印刷电路板上的其他 LSI 的串行数据进行快速传送。在执行此指令的过程中，测试电路不会对系统电路产生任何影响。

(2) **SAMPLE/PRELOAD**

SAMPLE/PRELOAD 指令将值从本 LSI 的内部电路输入到边界扫描寄存器，从扫描路径输出或者将数据装入到扫描路径。在执行此指令的过程中，本 LSI 的输入引脚的值直接被传送到内部电路，内部电路的值从输出引脚直接输出到外部。在执行此指令的过程中，本 LSI 的系统电路不受任何影响。

在 **SAMPLE** 执行过程中，将从输入引脚传送到内部电路的值或者从内部电路传送到输出引脚的值的瞬态捕到边界扫描寄存器，并且从扫描路径读取。与 **Capture-DR** 状态的 **TCK** 上升沿同步捕获瞬态，并且不防碍本 LSI 的正常运行。

在 **PRELOAD** 执行过程中，在 **EXTEST** 指令前将初始值从扫描路径设定到边界扫描寄存器的并行输出锁存器。如果 **PRELOAD** 没有执行，就在执行 **EXTEST** 指令时，在第一个扫描顺序结束（传送给输出锁存器）前，从输出引脚输出不定值（随时通过 **EXTEST** 指令将并行输出锁存器的信号输出到输出引脚）。

(3) EXTEST

在将本 LSI 安装到印刷电路板时，此指令用于测试外部电路。在执行此指令时，输出引脚用于将测试数据（已经用 SAMPLE/PRELOAD 指令设定）从边界扫描寄存器输出到印刷电路板，输入引脚用于将测试结果从印刷电路板取到边界扫描寄存器。在使用 N 次 EXTEST 指令进行测试的情况下，在扫出第 (N-1) 次时扫入第 N 次测试数据。

在此指令的 Capture-DR 状态下，输出引脚的边界扫描寄存器装入的数据不用于外部电路的测试（在移位中进行替换）。

(4) IDCODE

能通过从引脚给 SDIR 设定命令，将引脚设定为 JTAG 规定的 IDCODE 模式。在对此模块进行初始化时（TRST 有效，或者将 TAP 置为 Test-Logic-Reset 状态），为 IDCODE 模式。

(5) CLAMP 和 HIGHZ

能通过从引脚给 SDIR 设定命令，将引脚设定为 JTAG 规定的 CLAMP 模式和 HIGHZ 模式。

50.6.2 注意事项

1. 时钟相关信号（EXTAL、XTAL、CKIO、AUDIO_X1、AUDIO_X2、USB_X1、USB_X2、RTC_X1、RTC_X2）不是边界扫描对象。
2. 复位相关符号（RES）不是边界扫描对象。
3. 此模块相关符号（TCK、TDI、TDO、TMS、 $\overline{\text{TRST}}$ 、 $\overline{\text{ASEMD}}$ ）不是边界扫描对象。
4. USB 相关信号（DP、DM、VBUS、REFRIN）不是边界扫描对象。
5. 要执行边界扫描时，必须在 RES 引脚有效期间给 $\overline{\text{ASEMD}}$ 引脚输入高电平，并且在产品芯片模式中执行边界扫描。在边界扫描过程中，必须将 $\overline{\text{ASEMD}}$ 引脚固定为高电平。

50.7 使用时的注意事项

1. 此模块命令一旦被设定，只要不再设定其他命令就不会被更改。在连续设定相同命令时，需要先设定不影响芯片运行的命令（BYPASS 模式等），然后再次设定此命令。
2. 在软件待机模式和此模块的模块待机状态下，不能使用此模块的全部功能。为了在待机模式的前后保持 TAP 的状态，需要在待机模式转移时预先将 TCK 设为高电平。
3. 与是否使用此模块无关，在接通电源以及通过 RES 引脚的有效来解除深度待机时，必须将 $\overline{\text{TRST}}$ 引脚置为低电平，对此模块进行初始化。
4. 如果设定 TDO 变化时序切换命令并且在 RES 引脚无效后立即将 $\overline{\text{TRST}}$ 引脚置为有效，就有可能清除 TDO 变化时序切换命令。
为了防止此现象，如果设定了 TDO 变化时序切换命令，RES 引脚和 $\overline{\text{TRST}}$ 引脚之间的信号变化就必须至少有 20tcyc 的间隔。详细内容请参照“50.5.3 TDO 输出时序”。
5. 从 $\overline{\text{TRST}}$ 引脚无效到操作 TAP 控制器的期间，必须至少有 200ns 的间隔。
6. 从 $\overline{\text{TRST}}$ 引脚无效后的 200ns 期间，必须将 TMS 引脚固定为高电平。

51. 寄存器一览

有关寄存器一览表中本 LSI 的内部 I/O 寄存器的信息，说明如下。

(1) 寄存器地址一览表（按功能模块和手册章节号的顺序）

- 按功能模块和手册章节号的顺序记载寄存器。
- 不能存取寄存器地址一览表中未记载的保留地址。
- 在地址为 16 位或者 32 位的情况下，以大端法为前提，记载 MSB 侧的地址。
- 在存取长度部分用 “*” 表示读写时存取长度不同的寄存器。
详细内容请参照相关章节的寄存器说明。

(2) 寄存器位一览表

- 按照“寄存器地址一览表（按功能模块和手册章节号的顺序）”的顺序，记载各寄存器的位结构。
- 在位名部分用 “—” 表示保留位。
- 表中位名部分的空白栏表示将该寄存器全部分配给计数器或者数据。

(3) 各运行模式中的寄存器状态一览表

- 按照“寄存器地址一览表（按功能模块和手册章节号的顺序）”的顺序，记载寄存器状态。
- 各位初始化状态请参照相关章节的寄存器说明。
- 表示基本运行模式中的寄存器状态。在有内部外围模块固有的复位时，请参照内部外围模块的章节。

(4) 写内部外围模块的寄存器时的注意事项

在存取内部外围模块的寄存器时，从内部总线至少需要 2 个周期的外围模块时钟（Pφ）。在从 CPU 写内部外围寄存器时，CPU 不等到写完寄存器就执行后续指令。

例如，以为了降低功耗而转移到软件待机模式的情况为例进行说明。为了进行转移，需要在将 STBCR1 寄存器的 STBY 位置 “1” 后执行 SLEEP 指令，但是必须在执行 SLEEP 指令前虚读 STBCR1 寄存器。如果不虚读此寄存器，CPU 就在 STBY1 位被置 “1” 前执行 SLEEP 指令，因此不转移到软件待机模式而转移到睡眠模式。为了等待写 STBY 位，需要虚读 STBCR 寄存器。

如此例所示，要在执行后续指令时反映内部外围寄存器所产生的变更时，必须在执行寄存器的写指令后虚读相同的寄存器，然后再执行后续指令。

51.1 寄存器地址一览表（按功能模块和手册章节号的顺序）

模块名	寄存器名	略称	位数	地址	存取长度
时钟脉冲振荡器	频率控制寄存器	FRQCR	16	H'FFFE0010	16
中断控制器	中断控制寄存器 0	ICR0	16	H'FFFE0800	16、32
	中断控制寄存器 1	ICR1	16	H'FFFE0802	16、32
	中断控制寄存器 2	ICR2	16	H'FFFE0804	16、32
	IRQ 中断请求寄存器	IRQRR	16	H'FFFE0806	16、32
	PINT 中断允许寄存器	PINTER	16	H'FFFE0808	16、32
	PINT 中断请求寄存器	PIRR	16	H'FFFE080A	16、32
	存储体控制寄存器	IBCR	16	H'FFFE080C	16、32
	存储体号寄存器	IBNR	16	H'FFFE080E	16、32
	中断优先级设定寄存器 01	IPR01	16	H'FFFE0818	16、32
	中断优先级设定寄存器 02	IPR02	16	H'FFFE081A	16、32
	中断优先级设定寄存器 05	IPR05	16	H'FFFE0820	16、32
	中断优先级设定寄存器 06	IPR06	16	H'FFFE0C00	16、32
	中断优先级设定寄存器 07	IPR07	16	H'FFFE0C02	16、32
	中断优先级设定寄存器 08	IPR08	16	H'FFFE0C04	16、32
	中断优先级设定寄存器 09	IPR09	16	H'FFFE0C06	16、32
	中断优先级设定寄存器 10	IPR10	16	H'FFFE0C08	16、32
	中断优先级设定寄存器 11	IPR11	16	H'FFFE0C0A	16、32
	中断优先级设定寄存器 12	IPR12	16	H'FFFE0C0C	16、32
	中断优先级设定寄存器 13	IPR13	16	H'FFFE0C0E	16、32
	中断优先级设定寄存器 14	IPR14	16	H'FFFE0C10	16、32
	中断优先级设定寄存器 15	IPR15	16	H'FFFE0C12	16、32
	中断优先级设定寄存器 16	IPR16	16	H'FFFE0C14	16、32
	中断优先级设定寄存器 17	IPR17	16	H'FFFE0C16	16、32
	中断优先级设定寄存器 18	IPR18	16	H'FFFE0C18	16、32
	中断优先级设定寄存器 19	IPR19	16	H'FFFE0C1A	16、32
	中断优先级设定寄存器 20	IPR20	16	H'FFFE0C1C	16、32
中断优先级设定寄存器 21	IPR21	16	H'FFFE0C1E	16、32	
中断优先级设定寄存器 22	IPR22	16	H'FFFE0C20	16、32	
中断优先级设定寄存器 23	IPR23	16	H'FFFE0C22	16、32	
中断优先级设定寄存器 24	IPR24	16	H'FFFE0C24	16、32	
中断优先级设定寄存器 25	IPR25	16	H'FFFE0C26	16、32	
中断优先级设定寄存器 26	IPR26	16	H'FFFE0C28	16、32	
用户断点控制器	断点地址寄存器 _0	BAR_0	32	H'FFFC0400	32
	断点地址屏蔽寄存器 _0	BAMR_0	32	H'FFFC0404	32
	断点数据寄存器 _0	BDR_0	32	H'FFFC0408	32
	断点数据屏蔽寄存器 _0	BDMR_0	32	H'FFFC040C	32
	断点地址寄存器 _1	BAR_1	32	H'FFFC0410	32
	断点地址屏蔽寄存器 _1	BAMR_1	32	H'FFFC0414	32
	断点数据寄存器 _1	BDR_1	32	H'FFFC0418	32

模块名	寄存器名	略称	位数	地址	存取长度
用户断点控制器	断点数据屏蔽寄存器 _1	BDMR_1	32	H'FFFC041C	32
	断点总线周期寄存器 _0	BBR_0	16	H'FFFC04A0	16
	断点总线周期寄存器 _1	BBR_1	16	H'FFFC04B0	16
	断点控制寄存器	BRCR	32	H'FFFC04C0	32
高速缓存	高速缓存控制寄存器 1	CCR1	32	H'FFFC1000	32
	高速缓存控制寄存器 2	CCR2	32	H'FFFC1004	32
总线状态控制器	公共控制寄存器	CMNCR	32	H'FFFC0000	32
	CS0 空间总线控制寄存器	CS0BCR	32	H'FFFC0004	32
	CS1 空间总线控制寄存器	CS1BCR	32	H'FFFC0008	32
	CS2 空间总线控制寄存器	CS2BCR	32	H'FFFC000C	32
	CS3 空间总线控制寄存器	CS3BCR	32	H'FFFC0010	32
	CS4 空间总线控制寄存器	CS4BCR	32	H'FFFC0014	32
	CS5 空间总线控制寄存器	CS5BCR	32	H'FFFC0018	32
	CS0 空间等待控制寄存器	CS0WCR	32	H'FFFC0028	32
	CS1 空间等待控制寄存器	CS1WCR	32	H'FFFC002C	32
	CS2 空间等待控制寄存器	CS2WCR	32	H'FFFC0030	32
	CS3 空间等待控制寄存器	CS3WCR	32	H'FFFC0034	32
	CS4 空间等待控制寄存器	CS4WCR	32	H'FFFC0038	32
	CS5 空间等待控制寄存器	CS5WCR	32	H'FFFC003C	32
	SDRAM 控制寄存器	SDCR	32	H'FFFC004C	32
	刷新定时器的控制 / 状态寄存器	RTCSR	16	H'FFFC0050	32
	刷新定时器的计数器	RTCNT	16	H'FFFC0054	32
	刷新时间常数寄存器	RTCOR	16	H'FFFC0058	32
	直接存储器 存取控制器	DMA 源地址寄存器 _0	SAR0	32	H'FFFE1000
DMA 目标地址寄存器 _0		DAR0	32	H'FFFE1004	16、32
DMA 传送计数寄存器 _0		DMATCR0	32	H'FFFE1008	16、32
DMA 通道控制寄存器 _0		RSAR0	32	H'FFFE1100	16、32
DMA 重加载源地址寄存器 _0		RDAR0	32	H'FFFE1104	16、32
DMA 重加载目标地址寄存器 _0		RDMATCR0	32	H'FFFE1108	16、32
DMA 重加载传送计数寄存器 _0		CHCR0	32	H'FFFE100C	8、16、32
DMA 源地址寄存器 _1		SAR1	32	H'FFFE1010	16、32
DMA 目标地址寄存器 _1		DAR1	32	H'FFFE1014	16、32
DMA 传送计数寄存器 _1		DMATCR1	32	H'FFFE1018	16、32
DMA 通道控制寄存器 _1		CHCR1	32	H'FFFE101C	8、16、32
DMA 重加载源地址寄存器 _1		RSAR1	32	H'FFFE1110	16、32
DMA 重加载目标地址寄存器 _1		RDAR1	32	H'FFFE1114	16、32
DMA 重加载传送计数寄存器 _1		RDMATCR1	32	H'FFFE1118	16、32
DMA 源地址寄存器 _2		SAR2	32	H'FFFE1020	16、32
DMA 目标地址寄存器 _2		DAR2	32	H'FFFE1024	16、32
DMA 传送计数寄存器 _2		DMATCR2	32	H'FFFE1028	16、32
DMA 通道控制寄存器 _2		CHCR2	32	H'FFFE102C	8、16、32
DMA 重加载源地址寄存器 _2	RSAR2	32	H'FFFE1120	16、32	

模块名	寄存器名	略称	位数	地址	存取长度
直接存储器 存取控制器	DMA 重加载目标地址寄存器 _2	RDAR2	32	H'FFFE1124	16、32
	DMA 重加载传送计数寄存器 _2	RDMATCR2	32	H'FFFE1128	16、32
	DMA 源地址寄存器 _3	SAR3	32	H'FFFE1030	16、32
	DMA 目标地址寄存器 _3	DAR3	32	H'FFFE1034	16、32
	DMA 传送计数寄存器 _3	DMATCR3	32	H'FFFE1038	16、32
	DMA 通道控制寄存器 _3	CHCR3	32	H'FFFE103C	8、16、32
	DMA 重加载源地址寄存器 _3	RSAR3	32	H'FFFE1130	16、32
	DMA 重加载目标地址寄存器 _3	RDAR3	32	H'FFFE1134	16、32
	DMA 重加载传送计数寄存器 _3	RDMATCR3	32	H'FFFE1138	16、32
	DMA 源地址寄存器 _4	SAR4	32	H'FFFE1040	16、32
	DMA 目标地址寄存器 _4	DAR4	32	H'FFFE1044	16、32
	DMA 传送计数寄存器 _4	DMATCR4	32	H'FFFE1048	16、32
	DMA 通道控制寄存器 _4	CHCR4	32	H'FFFE104C	8、16、32
	DMA 重加载源地址寄存器 _4	RSAR4	32	H'FFFE1140	16、32
	DMA 重加载目标地址寄存器 _4	RDAR4	32	H'FFFE1144	16、32
	DMA 重加载传送计数寄存器 _4	RDMATCR4	32	H'FFFE1148	16、32
	DMA 源地址寄存器 _5	SAR5	32	H'FFFE1050	16、32
	DMA 目标地址寄存器 _5	DAR5	32	H'FFFE1054	16、32
	DMA 传送计数寄存器 _5	DMATCR5	32	H'FFFE1058	16、32
	DMA 通道控制寄存器 _5	CHCR5	32	H'FFFE105C	8、16、32
	DMA 重加载源地址寄存器 _5	RSAR5	32	H'FFFE1150	16、32
	DMA 重加载目标地址寄存器 _5	RDAR5	32	H'FFFE1154	16、32
	DMA 重加载传送计数寄存器 _5	RDMATCR5	32	H'FFFE1158	16、32
	DMA 源地址寄存器 _6	SAR6	32	H'FFFE1060	16、32
	DMA 目标地址寄存器 _6	DAR6	32	H'FFFE1064	16、32
	DMA 传送计数寄存器 _6	DMATCR6	32	H'FFFE1068	16、32
	DMA 通道控制寄存器 _6	CHCR6	32	H'FFFE106C	8、16、32
	DMA 重加载源地址寄存器 _6	RSAR6	32	H'FFFE1160	16、32
	DMA 重加载目标地址寄存器 _6	RDAR6	32	H'FFFE1164	16、32
	DMA 重加载传送计数寄存器 _6	RDMATCR6	32	H'FFFE1168	16、32
	DMA 源地址寄存器 _7	SAR7	32	H'FFFE1070	16、32
	DMA 目标地址寄存器 _7	DAR7	32	H'FFFE1074	16、32
	DMA 传送计数寄存器 _7	DMATCR7	32	H'FFFE1078	16、32
	DMA 通道控制寄存器 _7	CHCR7	32	H'FFFE107C	8、16、32
	DMA 重加载源地址寄存器 _7	RSAR7	32	H'FFFE1170	16、32
	DMA 重加载目标地址寄存器 _7	RDAR7	32	H'FFFE1174	16、32
	DMA 重加载传送计数寄存器 _7	RDMATCR7	32	H'FFFE1178	16、32
	DMA 源地址寄存器 _8	SAR8	32	H'FFFE1080	16、32
	DMA 目标地址寄存器 _8	DAR8	32	H'FFFE1084	16、32
	DMA 传送计数寄存器 _8	DMATCR8	32	H'FFFE1088	16、32
DMA 通道控制寄存器 _8	RSAR8	32	H'FFFE1180	16、32	
DMA 重加载源地址寄存器 _8	RDAR8	32	H'FFFE1184	16、32	

模块名	寄存器名	略称	位数	地址	存取长度
直接存储器 存取控制器	DMA 重加载目标地址寄存器 _8	RDMATCR8	32	H'FFFE1188	16、32
	DMA 重加载传送计数寄存器 _8	CHCR8	32	H'FFFE108C	8、16、32
	DMA 源地址寄存器 _9	SAR9	32	H'FFFE1090	16、32
	DMA 目标地址寄存器 _9	DAR9	32	H'FFFE1094	16、32
	DMA 传送计数寄存器 _9	DMATCR9	32	H'FFFE1098	16、32
	DMA 通道控制寄存器 _9	CHCR9	32	H'FFFE109C	8、16、32
	DMA 重加载源地址寄存器 _9	RSAR9	32	H'FFFE1190	16、32
	DMA 重加载目标地址寄存器 _9	RDAR9	32	H'FFFE1194	16、32
	DMA 重加载传送计数寄存器 _9	RDMATCR9	32	H'FFFE1198	16、32
	DMA 源地址寄存器 _10	SAR10	32	H'FFFE10A0	16、32
	DMA 目标地址寄存器 _10	DAR10	32	H'FFFE10A4	16、32
	DMA 传送计数寄存器 _10	DMATCR10	32	H'FFFE10A8	16、32
	DMA 通道控制寄存器 _10	CHCR10	32	H'FFFE10AC	8、16、32
	DMA 重加载源地址寄存器 _10	RSAR10	32	H'FFFE11A0	16、32
	DMA 重加载目标地址寄存器 _10	RDAR10	32	H'FFFE11A4	16、32
	DMA 重加载传送计数寄存器 _10	RDMATCR10	32	H'FFFE11A8	16、32
	DMA 源地址寄存器 _11	SAR11	32	H'FFFE10B0	16、32
	DMA 目标地址寄存器 _11	DAR11	32	H'FFFE10B4	16、32
	DMA 传送计数寄存器 _11	DMATCR11	32	H'FFFE10B8	16、32
	DMA 通道控制寄存器 _11	CHCR11	32	H'FFFE10BC	8、16、32
	DMA 重加载源地址寄存器 _11	RSAR11	32	H'FFFE11B0	16、32
	DMA 重加载目标地址寄存器 _11	RDAR11	32	H'FFFE11B4	16、32
	DMA 重加载传送计数寄存器 _11	RDMATCR11	32	H'FFFE11B8	16、32
	DMA 源地址寄存器 _12	SAR12	32	H'FFFE10C0	16、32
	DMA 目标地址寄存器 _12	DAR12	32	H'FFFE10C4	16、32
	DMA 传送计数寄存器 _12	DMATCR12	32	H'FFFE10C8	16、32
	DMA 通道控制寄存器 _12	CHCR12	32	H'FFFE10CC	8、16、32
	DMA 重加载源地址寄存器 _12	RSAR12	32	H'FFFE11C0	16、32
	DMA 重加载目标地址寄存器 _12	RDAR12	32	H'FFFE11C4	16、32
	DMA 重加载传送计数寄存器 _12	RDMATCR12	32	H'FFFE11C8	16、32
	DMA 源地址寄存器 _13	SAR13	32	H'FFFE10D0	16、32
	DMA 目标地址寄存器 _13	DAR13	32	H'FFFE10D4	16、32
	DMA 传送计数寄存器 _13	DMATCR13	32	H'FFFE10D8	16、32
	DMA 通道控制寄存器 _13	CHCR13	32	H'FFFE10DC	8、16、32
	DMA 重加载源地址寄存器 _13	RSAR13	32	H'FFFE11D0	16、32
	DMA 重加载目标地址寄存器 _13	RDAR13	32	H'FFFE11D4	16、32
	DMA 重加载传送计数寄存器 _13	RDMATCR13	32	H'FFFE11D8	16、32
	DMA 源地址寄存器 _14	SAR14	32	H'FFFE10E0	16、32
	DMA 目标地址寄存器 _14	DAR14	32	H'FFFE10E4	16、32
	DMA 传送计数寄存器 _14	DMATCR14	32	H'FFFE10E8	16、32
DMA 通道控制寄存器 _14	CHCR14	32	H'FFFE10EC	8、16、32	
DMA 重加载源地址寄存器 _14	RSAR14	32	H'FFFE11E0	16、32	

模块名	寄存器名	略称	位数	地址	存取长度
直接存储器 存取控制器	DMA 重加载目标地址寄存器 _14	RDAR14	32	H'FFFE11E4	16、32
	DMA 重加载传送计数寄存器 _14	RDMATCR14	32	H'FFFE11E8	16、32
	DMA 源地址寄存器 _15	SAR15	32	H'FFFE10F0	16、32
	DMA 目标地址寄存器 _15	DAR15	32	H'FFFE10F4	16、32
	DMA 传送计数寄存器 _15	DMATCR15	32	H'FFFE10F8	16、32
	DMA 通道控制寄存器 _15	CHCR15	32	H'FFFE10FC	8、16、32
	DMA 重加载源地址寄存器 _15	RSAR15	32	H'FFFE11F0	16、32
	DMA 重加载目标地址寄存器 _15	RDAR15	32	H'FFFE11F4	16、32
	DMA 重加载传送计数寄存器 _15	RDMATCR15	32	H'FFFE11F8	16、32
	DMA 操作寄存器	DMAOR	16	H'FFFE1200	8、16
	DMA 扩展资源选择器 0	DMARS0	16	H'FFFE1300	16
	DMA 扩展资源选择器 1	DMARS1	16	H'FFFE1304	16
	DMA 扩展资源选择器 2	DMARS2	16	H'FFFE1308	16
	DMA 扩展资源选择器 3	DMARS3	16	H'FFFE130C	16
	DMA 扩展资源选择器 4	DMARS4	16	H'FFFE1310	16
	DMA 扩展资源选择器 5	DMARS5	16	H'FFFE1314	16
	DMA 扩展资源选择器 6	DMARS6	16	H'FFFE1318	16
	DMA 扩展资源选择器 7	DMARS7	16	H'FFFE131C	16
多功能定时器 脉冲单元 2	定时器的控制寄存器 _0	TCR_0	8	H'FFFE4300	8
	定时器的模式寄存器 _0	TMDR_0	8	H'FFFE4301	8
	定时器 I/O 的控制寄存器 H_0	TIORH_0	8	H'FFFE4302	8
	定时器 I/O 的控制寄存器 L_0	TIORL_0	8	H'FFFE4303	8
	定时器的中断允许寄存器 _0	TIER_0	8	H'FFFE4304	8
	定时器的状态寄存器 _0	TSR_0	8	H'FFFE4305	8
	定时器的计数器 _0	TCNT_0	16	H'FFFE4306	16
	定时器的通用寄存器 A_0	TGRA_0	16	H'FFFE4308	16
	定时器的通用寄存器 B_0	TGRB_0	16	H'FFFE430A	16
	定时器的通用寄存器 C_0	TGRC_0	16	H'FFFE430C	16
	定时器的通用寄存器 D_0	TGRD_0	16	H'FFFE430E	16
	定时器的通用寄存器 E_0	TGRE_0	16	H'FFFE4320	16
	定时器的通用寄存器 F_0	TGRF_0	16	H'FFFE4322	16
	定时器的中断允许寄存器 2_0	TIER2_0	8	H'FFFE4324	8
	定时器的状态寄存器 2_0	TSR2_0	8	H'FFFE4325	8
	定时器的缓冲运行传送模式寄存器 _0	TBTM_0	8	H'FFFE4326	8
	定时器的控制寄存器 _1	TCR_1	8	H'FFFE4380	8
	定时器的模式寄存器 _1	TMDR_1	8	H'FFFE4381	8
	定时器 I/O 的控制寄存器 _1	TIOR_1	8	H'FFFE4382	8
	定时器的中断允许寄存器 _1	TIER_1	8	H'FFFE4384	8
	定时器的状态寄存器 _1	TSR_1	8	H'FFFE4385	8
	定时器的计数器 _1	TCNT_1	16	H'FFFE4386	16
	定时器的通用寄存器 A_1	TGRA_1	16	H'FFFE4388	16
定时器的通用寄存器 B_1	TGRB_1	16	H'FFFE438A	16	

模块名	寄存器名	略称	位数	地址	存取长度
多功能定时器 脉冲单元 2	定时器的输入捕捉寄存器	TICCR	8	H'FFFE4390	8
	定时器的控制寄存器 _2	TCR_2	8	H'FFFE4000	8
	定时器的模式寄存器 _2	TMDR_2	8	H'FFFE4001	8
	定时器 I/O 的控制寄存器 _2	TIOR_2	8	H'FFFE4002	8
	定时器的中断允许寄存器 _2	TIER_2	8	H'FFFE4004	8
	定时器的状态寄存器 _2	TSR_2	8	H'FFFE4005	8
	定时器的计数器 _2	TCNT_2	16	H'FFFE4006	16
	定时器的通用寄存器 A_2	TGRA_2	16	H'FFFE4008	16
	定时器的通用寄存器 B_2	TGRB_2	16	H'FFFE400A	16
	定时器的控制寄存器 _3	TCR_3	8	H'FFFE4200	8
	定时器的模式寄存器 _3	TMDR_3	8	H'FFFE4202	8
	定时器 I/O 的控制寄存器 H_3	TIORH_3	8	H'FFFE4204	8
	定时器 I/O 的控制寄存器 L_3	TIORL_3	8	H'FFFE4205	8
	定时器的中断允许寄存器 _3	TIER_3	8	H'FFFE4208	8
	定时器的状态寄存器 _3	TSR_3	8	H'FFFE422C	8
	定时器的计数器 _3	TCNT_3	16	H'FFFE4210	16
	定时器的通用寄存器 A_3	TGRA_3	16	H'FFFE4218	16
	定时器的通用寄存器 B_3	TGRB_3	16	H'FFFE421A	16
	定时器的通用寄存器 C_3	TGRC_3	16	H'FFFE4224	16
	定时器的通用寄存器 D_3	TGRD_3	16	H'FFFE4226	16
	定时器的缓冲运行传送模式寄存器 _3	TBTM_3	8	H'FFFE4238	8
	定时器的控制寄存器 _4	TCR_4	8	H'FFFE4201	8
	定时器的模式寄存器 _4	TMDR_4	8	H'FFFE4203	8
	定时器 I/O 的控制寄存器 H_4	TIORH_4	8	H'FFFE4206	8
	定时器 I/O 的控制寄存器 L_4	TIORL_4	8	H'FFFE4207	8
	定时器的中断允许寄存器 _4	TIER_4	8	H'FFFE4209	8
	定时器的状态寄存器 _4	TSR_4	8	H'FFFE422D	8
	定时器的计数器 _4	TCNT_4	16	H'FFFE4212	16
	定时器的通用寄存器 A_4	TGRA_4	16	H'FFFE421C	16
	定时器的通用寄存器 B_4	TGRB_4	16	H'FFFE421E	16
	定时器的通用寄存器 C_4	TGRC_4	16	H'FFFE4228	16
	定时器的通用寄存器 D_4	TGRD_4	16	H'FFFE422A	16
	定时器的缓冲运行传送模式寄存器 _4	TBTM_4	8	H'FFFE4239	8
	定时器的 A/D 转换开始请求控制寄存器	TADCR	16	H'FFFE4240	16
	定时器的 A/D 转换开始请求周期设定寄存器 A_4	TADCORA_4	16	H'FFFE4244	16
	定时器的 A/D 转换开始请求周期设定寄存器 B_4	TADCORB_4	16	H'FFFE4246	16
	定时器的 A/D 转换开始请求周期设定缓冲寄存器 A_4	TADCOBRA_4	16	H'FFFE4248	16
	定时器的 A/D 转换开始请求周期设定缓冲寄存器 B_4	TADCOBRB_4	16	H'FFFE424A	16
	定时器的启动寄存器	TSTR	8	H'FFFE4280	8

模块名	寄存器名	略称	位数	地址	存取长度
多功能定时器 脉冲单元 2	定时器的同步寄存器	TSYR	8	H'FFFE4281	8
	定时器的读写允许寄存器	TRWER	8	H'FFFE4284	8
	定时器的输出主控允许寄存器	TOER	8	H'FFFE420A	8
	定时器的输出主控控制寄存器 1	TOCR1	8	H'FFFE420E	8
	定时器的输出主控控制寄存器 2	TOCR2	8	H'FFFE420F	8
	定时器的门控寄存器	TGCR	8	H'FFFE420D	8
	定时器的周期数据寄存器	TCDR	16	H'FFFE4214	16
	定时器的空载时间数据寄存器	TDDR	16	H'FFFE4216	16
	定时器的副计数器	TCNTS	16	H'FFFE4220	16
	定时器的周期缓冲寄存器	TCBR	16	H'FFFE4222	16
	定时器的中断减少设定寄存器	TITCR	8	H'FFFE4230	8
	定时器的中断减少次数计数器	TITCNT	8	H'FFFE4231	8
	定时器的缓冲传送设定寄存器	TBTER	8	H'FFFE4232	8
	定时器的空载时间允许寄存器	TDER	8	H'FFFE4234	8
	定时器的波形控制寄存器	TWCR	8	H'FFFE4260	8
定时器的输出电平缓冲寄存器	TOLBR	8	H'FFFE4236	8	
比较匹配定时器	比较匹配定时器的启动寄存器	CMSTR	16	H'FFFE0000	16
	比较匹配定时器的控制 / 状态寄存器 _0	CMCSR_0	16	H'FFFE0002	16
	比较匹配计数器 _0	CMCNT_0	16	H'FFFE0004	8、16
	比较匹配常数寄存器 _0	CMCOR_0	16	H'FFFE0006	8、16
	比较匹配定时器的控制 / 状态寄存器 _1	CMCSR_1	16	H'FFFE0008	16
	比较匹配计数器 _1	CMCNT_1	16	H'FFFE000A	8、16
	比较匹配常数寄存器 _1	CMCOR_1	16	H'FFFE000C	8、16
看门狗定时器	看门狗定时器的计数器	WTCNT	8	H'FFFE0002	8、16
	看门狗定时器的控制 / 状态寄存器	WTCSR	8	H'FFFE0000	8、16
	看门狗定时器的复位控制 / 状态寄存器	WRCSR	8	H'FFFE0004	8、16
实时时钟	64Hz 计数器	R64CNT	8	H'FFFE6000	8
	秒计数器	RSECCNT	8	H'FFFE6002	8
	分钟计数器	RMINCNT	8	H'FFFE6004	8
	小时计数器	RHRCNT	8	H'FFFE6006	8
	星期计数器	RWKCNT	8	H'FFFE6008	8
	日计数器	RDAYCNT	8	H'FFFE600A	8
	月计数器	RMONCNT	8	H'FFFE600C	8
	年计数器	RYRCNT	16	H'FFFE600E	16
	秒闹钟寄存器	RSECAR	8	H'FFFE6010	8
	分钟闹钟寄存器	RMINAR	8	H'FFFE6012	8
	小时闹钟寄存器	RHRAR	8	H'FFFE6014	8
	星期闹钟寄存器	RWKAR	8	H'FFFE6016	8
	日闹钟寄存器	RDAYAR	8	H'FFFE6018	8
	月闹钟寄存器	RMONAR	8	H'FFFE601A	8
	年闹钟寄存器	RYRAR	16	H'FFFE6020	16
	控制寄存器 1	RCR1	8	H'FFFE601C	8

模块名	寄存器名	略称	位数	地址	存取长度
实时时钟	控制寄存器 2	RCR2	8	H'FFFE601E	8
	控制寄存器 3	RCR3	8	H'FFFE6024	8
	控制寄存器 5	RCR5	8	H'FFFE6026	8
	频率寄存器 H	RFRH	16	H'FFFE602A	16
	频率寄存器 L	RFRL	16	H'FFFE602C	16
FIFO 内置型串行通信单元	串行模式寄存器 _0	SCSMR_0	16	H'E8007000	16
	位速率寄存器 _0	SCBRR_0	8	H'E8007004	8
	串行控制寄存器 _0	SCSCR_0	16	H'E8007008	16
	发送 FIFO 数据寄存器 _0	SCFTDR_0	8	H'E800700C	8
	串行状态寄存器 _0	SCFSR_0	16	H'E8007010	16
	接收 FIFO 数据寄存器 _0	SCFRDR_0	8	H'E8007014	8
	FIFO 控制寄存器 _0	SCFCR_0	16	H'E8007018	16
	FIFO 数据计数置位寄存器 _0	SCFDR_0	16	H'E800701C	16
	串行端口寄存器 _0	SCSPTR_0	16	H'E8007020	16
	线路状态寄存器 _0	SCLSR_0	16	H'E8007024	16
	串行扩展模式寄存器 _0	SCEMR_0	16	H'E8007028	16
	串行模式寄存器 _1	SCSMR_1	16	H'E8007800	16
	位速率寄存器 _1	SCBRR_1	8	H'E8007804	8
	串行控制寄存器 _1	SCSCR_1	16	H'E8007808	16
	发送 FIFO 数据寄存器 _1	SCFTDR_1	8	H'E800780C	8
	串行状态寄存器 _1	SCFSR_1	16	H'E8007810	16
	接收 FIFO 数据寄存器 _1	SCFRDR_1	8	H'E8007814	8
	FIFO 控制寄存器 _1	SCFCR_1	16	H'E8007818	16
	FIFO 数据计数置位寄存器 _1	SCFDR_1	16	H'E800781C	16
	串行端口寄存器 _1	SCSPTR_1	16	H'E8007820	16
	线路状态寄存器 _1	SCLSR_1	16	H'E8007824	16
	串行扩展模式寄存器 _1	SCEMR_1	16	H'E8007828	16
	串行模式寄存器 _2	SCSMR_2	16	H'E8008000	16
	位速率寄存器 _2	SCBRR_2	8	H'E8008004	8
	串行控制寄存器 _2	SCSCR_2	16	H'E8008008	16
	发送 FIFO 数据寄存器 _2	SCFTDR_2	8	H'E800800C	8
	串行状态寄存器 _2	SCFSR_2	16	H'E8008010	16
	接收 FIFO 数据寄存器 _2	SCFRDR_2	8	H'E8008014	8
	FIFO 控制寄存器 _2	SCFCR_2	16	H'E8008018	16
	FIFO 数据计数置位寄存器 _2	SCFDR_2	16	H'E800801C	16
	串行端口寄存器 _2	SCSPTR_2	16	H'E8008020	16
	线路状态寄存器 _2	SCLSR_2	16	H'E8008024	16
	串行扩展模式寄存器 _2	SCEMR_2	16	H'E8008028	16
串行模式寄存器 _3	SCSMR_3	16	H'E8008800	16	
位速率寄存器 _3	SCBRR_3	8	H'E8008804	8	
串行控制寄存器 _3	SCSCR_3	16	H'E8008808	16	
发送 FIFO 数据寄存器 _3	SCFTDR_3	8	H'E800880C	8	

模块名	寄存器名	略称	位数	地址	存取长度
FIFO 内置型串行通信单元	串行状态寄存器_3	SCFSR_3	16	H'E8008810	16
	接收 FIFO 数据寄存器_3	SCFRDR_3	8	H'E8008814	8
	FIFO 控制寄存器_3	SCFCR_3	16	H'E8008818	16
	FIFO 数据计数置位寄存器_3	SCFDR_3	16	H'E800881C	16
	串行端口寄存器_3	SCSPTR_3	16	H'E8008820	16
	线路状态寄存器_3	SCLSR_3	16	H'E8008824	16
	串行扩展模式寄存器_3	SCEMR_3	16	H'E8008828	16
	串行模式寄存器_4	SCSMR_4	16	H'E8009000	16
	位速率寄存器_4	SCBRR_4	8	H'E8009004	8
	串行控制寄存器_4	SCSCR_4	16	H'E8009008	16
	发送 FIFO 数据寄存器_4	SCFTDR_4	8	H'E800900C	8
	串行状态寄存器_4	SCFSR_4	16	H'E8009010	16
	接收 FIFO 数据寄存器_4	SCFRDR_4	8	H'E8009014	8
	FIFO 控制寄存器_4	SCFCR_4	16	H'E8009018	16
	FIFO 数据计数置位寄存器_4	SCFDR_4	16	H'E800901C	16
	串行端口寄存器_4	SCSPTR_4	16	H'E8009020	16
	线路状态寄存器_4	SCLSR_4	16	H'E8009024	16
	串行扩展模式寄存器_4	SCEMR_4	16	H'E8009028	16
	串行模式寄存器_5	SCSMR_5	16	H'E8009800	16
	位速率寄存器_5	SCBRR_5	8	H'E8009804	8
	串行控制寄存器_5	SCSCR_5	16	H'E8009808	16
	发送 FIFO 数据寄存器_5	SCFTDR_5	8	H'E800980C	8
	串行状态寄存器_5	SCFSR_5	16	H'E8009810	16
	接收 FIFO 数据寄存器_5	SCFRDR_5	8	H'E8009814	8
	FIFO 控制寄存器_5	SCFCR_5	16	H'E8009818	16
	FIFO 数据计数置位寄存器_5	SCFDR_5	16	H'E800981C	16
	串行端口寄存器_5	SCSPTR_5	16	H'E8009820	16
	线路状态寄存器_5	SCLSR_5	16	H'E8009824	16
	串行扩展模式寄存器_5	SCEMR_5	16	H'E8009828	16
	串行模式寄存器_6	SCSMR_6	16	H'E800A000	16
	位速率寄存器_6	SCBRR_6	8	H'E800A004	8
	串行控制寄存器_6	SCSCR_6	16	H'E800A008	16
	发送 FIFO 数据寄存器_6	SCFTDR_6	8	H'E800A00C	8
	串行状态寄存器_6	SCFSR_6	16	H'E800A010	16
	接收 FIFO 数据寄存器_6	SCFRDR_6	8	H'E800A014	8
FIFO 控制寄存器_6	SCFCR_6	16	H'E800A018	16	
FIFO 数据计数置位寄存器_6	SCFDR_6	16	H'E800A01C	16	
串行端口寄存器_6	SCSPTR_6	16	H'E800A020	16	
线路状态寄存器_6	SCLSR_6	16	H'E800A024	16	
串行扩展模式寄存器_6	SCEMR_6	16	H'E800A028	16	
串行模式寄存器_7	SCSMR_7	16	H'E800A800	16	
位速率寄存器_7	SCBRR_7	8	H'E800A804	8	

模块名	寄存器名	略称	位数	地址	存取长度
FIFO 内置型串行通信单元	串行控制寄存器 _7	SCSCR_7	16	H'E800A808	16
	发送 FIFO 数据寄存器 _7	SCFTDR_7	8	H'E800A80C	8
	串行状态寄存器 _7	SCFSR_7	16	H'E800A810	16
	接收 FIFO 数据寄存器 _7	SCFRDR_7	8	H'E800A814	8
	FIFO 控制寄存器 _7	SCFCR_7	16	H'E800A818	16
	FIFO 数据计数置位寄存器 _7	SCFDR_7	16	H'E800A81C	16
	串行端口寄存器 _7	SCSPTR_7	16	H'E800A820	16
	线路状态寄存器 _7	SCLSR_7	16	H'E800A824	16
	串行扩展模式寄存器 _7	SCEMR_7	16	H'E800A828	16
瑞萨串行外围接口	控制寄存器 0	SPCR0	8	H'E800E000	8、16
	从属选择性寄存器 0	SSLP0	8	H'E800E001	8、16
	引脚控制寄存器 0	SPPCR0	8	H'E800E002	8、16
	状态寄存器 0	SPSR0	8	H'E800E003	8、16
	数据寄存器 0	SPDR0	32	H'E800E004	8、16、32
	顺序控制寄存器 0	SPSCR0	8	H'E800E008	8、16
	顺序状态寄存器 0	SPSSR0	8	H'E800E009	8、16
	位速率寄存器 0	SPBR0	8	H'E800E00A	8、16
	数据控制寄存器 0	SPDCR0	8	H'E800E00B	8、16
	时钟延迟寄存器 0	SPCKD0	8	H'E800E00C	8、16
	从属选择无效延迟寄存器 0	SSLND0	8	H'E800E00D	8、16
	下次存取延迟寄存器 0	SPND0	8	H'E800E00E	8
	命令寄存器 00	SPCMD00	16	H'E800E010	16
	命令寄存器 01	SPCMD01	16	H'E800E012	16
	命令寄存器 02	SPCMD02	16	H'E800E014	16
	命令寄存器 03	SPCMD03	16	H'E800E016	16
	缓冲控制寄存器 0	SPBFCR0	8	H'E800E020	8、16
	缓冲数据计数置位寄存器 0	SPBFDR0	16	H'E800E022	16
	控制寄存器 1	SPCR1	8	H'E800E800	8、16
	从属选择性寄存器 1	SSLP1	8	H'E800E801	8、16
	引脚控制寄存器 1	SPPCR1	8	H'E800E802	8、16
	状态寄存器 1	SPSR1	8	H'E800E803	8、16
	数据寄存器 1	SPDR1	32	H'E800E804	8、16、32
	顺序控制寄存器 1	SPSCR1	8	H'E800E808	8、16
	顺序状态寄存器 1	SPSSR1	8	H'E800E809	8、16
	位速率寄存器 1	SPBR1	8	H'E800E80A	8、16
	数据控制寄存器 1	SPDCR1	8	H'E800E80B	8、16
	时钟延迟寄存器 1	SPCKD1	8	H'E800E80C	8、16
	从属选择无效延迟寄存器 1	SSLND1	8	H'E800E80D	8、16
	下次存取延迟寄存器 1	SPND1	8	H'E800E80E	8
	命令寄存器 10	SPCMD10	16	H'E800E810	16
	命令寄存器 11	SPCMD11	16	H'E800E812	16
命令寄存器 12	SPCMD12	16	H'E800E814	16	

模块名	寄存器名	略称	位数	地址	存取长度
瑞萨串行外围接口	命令寄存器 13	SPCMD13	16	H'E800E816	16
	缓冲控制寄存器 1	SPBFCR1	8	H'E800E820	8、16
	缓冲数据计数置位寄存器 1	SPBFDR1	16	H'E800E822	16
瑞萨四路串行外围接口	控制寄存器 _0	SPCR_0	8	H'E8033800	8、16、32
	从属选择极性寄存器 _0	SSLP_0	8	H'E8033801	8、16、32
	引脚控制寄存器 _0	SPPCR_0	8	H'E8033802	8、16、32
	状态寄存器 _0	SPSR_0	8	H'E8033803	8、16、32
	数据寄存器 _0	SPDR_0	32	H'E8033804	8、16、32
	顺序控制寄存器 _0	SPSCR_0	8	H'E8033808	8、16、32
	顺序状态寄存器 _0	SPSSR_0	8	H'E8033809	8、16、32
	位速率寄存器 _0	SPBR_0	8	H'E803380A	8、16、32
	数据控制寄存器 _0	SPDCR_0	8	H'E803380B	8、16、32
	时钟延迟寄存器 _0	SPCKD_0	8	H'E803380C	8、16、32
	从属选择无效延迟寄存器 _0	SSLND_0	8	H'E803380D	8、16、32
	下次存取延迟寄存器 _0	SPND_0	8	H'E803380E	8、16、32
	命令寄存器 0_0	SPCMD0_0	16	H'E8033810	8、16
	命令寄存器 1_0	SPCMD1_0	16	H'E8033812	8、16
	命令寄存器 2_0	SPCMD2_0	16	H'E8033814	8、16
	命令寄存器 3_0	SPCMD3_0	16	H'E8033816	8、16
	缓冲控制寄存器 _0	SPBFCR_0	8	H'E8033818	8、16、32
	缓冲数据计数置位寄存器 _0	SPBDCR_0	16	H'E803381A	8、16
	数据长度倍数设定寄存器 0_0	SPBMUL0_0	32	H'E803381C	32
	数据长度倍数设定寄存器 1_0	SPBMUL1_0	32	H'E8033820	32
	数据长度倍数设定寄存器 2_0	SPBMUL2_0	32	H'E8033824	32
	数据长度倍数设定寄存器 3_0	SPBMUL3_0	32	H'E8033828	32
	控制寄存器 _1	SPCR_1	8	H'E8034000	8、16、32
	从属选择极性寄存器 _1	SSLP_1	8	H'E8034001	8、16、32
	引脚控制寄存器 _1	SPPCR_1	8	H'E8034002	8、16、32
	状态寄存器 _1	SPSR_1	8	H'E8034003	8、16、32
	数据寄存器 _1	SPDR_1	32	H'E8034004	8、16、32
	顺序控制寄存器 _1	SPSCR_1	8	H'E8034008	8、16、32
	顺序状态寄存器 _1	SPSSR_1	8	H'E8034009	8、16、32
	位速率寄存器 _1	SPBR_1	8	H'E803400A	8、16、32
	数据控制寄存器 _1	SPDCR_1	8	H'E803400B	8、16、32
	时钟延迟寄存器 _1	SPCKD_1	8	H'E803400C	8、16、32
	从属选择无效延迟寄存器 _1	SSLND_1	8	H'E803400D	8、16、32
	下次存取延迟寄存器 _1	SPND_1	8	H'E803400E	8、16、32
	命令寄存器 0_1	SPCMD0_1	16	H'E8034010	8、16
	命令寄存器 1_1	SPCMD1_1	16	H'E8034012	8、16
	命令寄存器 2_1	SPCMD2_1	16	H'E8034014	8、16
	命令寄存器 3_1	SPCMD3_1	16	H'E8034016	8、16
	缓冲控制寄存器 _1	SPBFCR_1	8	H'E8034018	8、16、32

模块名	寄存器名	略称	位数	地址	存取长度
瑞萨四路串行 外围接口	缓冲数据计数置位寄存器_1	SPBDCR_1	16	H'E803401A	8、16
	数据长度倍数设定寄存器 0_1	SPBMUL0_1	32	H'E803401C	32
	数据长度倍数设定寄存器 1_1	SPBMUL1_1	32	H'E8034020	32
	数据长度倍数设定寄存器 2_1	SPBMUL2_1	32	H'E8034024	32
	数据长度倍数设定寄存器 3_1	SPBMUL3_1	32	H'E8034028	32
SPI 多 I/O 总线 控制器	公共控制寄存器	CMNCR	32	H'FFFC1C00	32
	SSL 延迟寄存器	SSLDR	32	H'FFFC1C04	32
	位速率设定寄存器	SPBCR	32	H'FFFC1C08	32
	数据读控制寄存器	DRCR	32	H'FFFC1C0C	32
	数据读命令设定寄存器	DRCMR	32	H'FFFC1C10	32
	数据读扩展地址设定寄存器	DREAR	32	H'FFFC1C14	32
	数据读选项设定寄存器	DROPR	32	H'FFFC1C18	32
	数据读允许设定寄存器	DRENDR	32	H'FFFC1C1C	32
	SPI 模式控制寄存器	SMCR	32	H'FFFC1C20	32
	SPI 模式命令设定寄存器	SMCMR	32	H'FFFC1C24	32
	SPI 模式地址设定寄存器	SMADR	32	H'FFFC1C28	32
	SPI 模式选项设定寄存器	SMOPR	32	H'FFFC1C2C	32
	SPI 模式允许设定寄存器	SMENR	32	H'FFFC1C30	32
	SPI 模式读数据寄存器 0	SMRDR0	32	H'FFFC1C38	8、16、32
	SPI 模式读数据寄存器 1	SMRDR1	32	H'FFFC1C3C	8、16、32
	SPI 模式写数据寄存器 0	SMWDR0	32	H'FFFC1C40	8、16、32
	SPI 模式写数据寄存器 1	SMWDR1	32	H'FFFC1C44	8、16、32
	公共状态寄存器	CMNSR	32	H'FFFC1C48	32
	I ² C 总线接口 3	I ² C 总线控制寄存器 1_0	ICCR1_0	8	H'FFFEE000
I ² C 总线控制寄存器 2_0		ICCR2_0	8	H'FFFEE001	8
I ² C 总线模式寄存器_0		ICMR_0	8	H'FFFEE002	8
I ² C 总线中断允许寄存器_0		ICIER_0	8	H'FFFEE003	8
I ² C 总线状态寄存器_0		ICSR_0	8	H'FFFEE004	8
从属地址寄存器_0		SAR_0	8	H'FFFEE005	8
I ² C 总线发送数据寄存器_0		ICDRT_0	8	H'FFFEE006	8
I ² C 总线接收数据寄存器_0		ICDRR_0	8	H'FFFEE007	8
NF2CYC 寄存器_0		NF2CYC_0	8	H'FFFEE008	8
I ² C 总线控制寄存器 1_1		ICCR1_1	8	H'FFFEE400	8
I ² C 总线控制寄存器 2_1		ICCR2_1	8	H'FFFEE401	8
I ² C 总线模式寄存器_1		ICMR_1	8	H'FFFEE402	8
I ² C 总线中断允许寄存器_1		ICIER_1	8	H'FFFEE403	8
I ² C 总线状态寄存器_1		ICSR_1	8	H'FFFEE404	8
从属地址寄存器_1		SAR_1	8	H'FFFEE405	8
I ² C 总线发送数据寄存器_1		ICDRT_1	8	H'FFFEE406	8
I ² C 总线接收数据寄存器_1		ICDRR_1	8	H'FFFEE407	8
NF2CYC 寄存器_1		NF2CYC_1	8	H'FFFEE408	8
I ² C 总线控制寄存器 1_2		ICCR1_2	8	H'FFFEE800	8

模块名	寄存器名	略称	位数	地址	存取长度
I2C 总线接口 3	I2C 总线控制寄存器_2_2	ICCR2_2	8	H'FFFEE801	8
	I2C 总线模式寄存器_2	ICMR_2	8	H'FFFEE802	8
	I2C 总线中断允许寄存器_2	ICIER_2	8	H'FFFEE803	8
	I2C 总线状态寄存器_2	ICSR_2	8	H'FFFEE804	8
	从属地址寄存器_2	SAR_2	8	H'FFFEE805	8
	I2C 总线发送数据寄存器_2	ICDRT_2	8	H'FFFEE806	8
	I2C 总线接收数据寄存器_2	ICDRR_2	8	H'FFFEE807	8
	NF2CYC 寄存器_2	NF2CYC_2	8	H'FFFEE808	8
	I2C 总线控制寄存器_1_3	ICCR1_3	8	H'FFFEEC00	8
	I2C 总线控制寄存器_2_3	ICCR2_3	8	H'FFFEEC01	8
	I2C 总线模式寄存器_3	ICMR_3	8	H'FFFEEC02	8
	I2C 总线中断允许寄存器_3	ICIER_3	8	H'FFFEEC03	8
	I2C 总线状态寄存器_3	ICSR_3	8	H'FFFEEC04	8
	从属地址寄存器_3	SAR_3	8	H'FFFEEC05	8
	I2C 总线发送数据寄存器_3	ICDRT_3	8	H'FFFEEC06	8
	I2C 总线接收数据寄存器_3	ICDRR_3	8	H'FFFEEC07	8
	NF2CYC 寄存器_3	NF2CYC_3	8	H'FFFEEC08	8
串行音频接口	控制寄存器_0	SSICR_0	32	H'FFFF0000	8、16、32
	状态寄存器_0	SSISR_0	32	H'FFFF0004	8、16、32
	FIFO 控制寄存器_0	SSIFCR_0	32	H'FFFF0010	8、16、32
	FIFO 状态寄存器_0	SSIFSR_0	32	H'FFFF0014	8、16、32
	发送 FIFO 数据寄存器_0	SSIFTDR_0	32	H'FFFF0018	32
	接收 FIFO 数据寄存器_0	SSIFRDR_0	32	H'FFFF001C	32
	TDM 模式寄存器_0	SSITDMR_0	32	H'FFFF0020	8、16、32
	控制寄存器_1	SSICR_1	32	H'FFFF0800	8、16、32
	状态寄存器_1	SSISR_1	32	H'FFFF0804	8、16、32
	FIFO 控制寄存器_1	SSIFCR_1	32	H'FFFF0810	8、16、32
	FIFO 状态寄存器_1	SSIFSR_1	32	H'FFFF0814	8、16、32
	发送 FIFO 数据寄存器_1	SSIFTDR_1	32	H'FFFF0818	32
	接收 FIFO 数据寄存器_1	SSIFRDR_1	32	H'FFFF081C	32
	TDM 模式寄存器_1	SSITDMR_1	32	H'FFFF0820	8、16、32
	控制寄存器_2	SSICR_2	32	H'FFFF1000	8、16、32
	状态寄存器_2	SSISR_2	32	H'FFFF1004	8、16、32
	FIFO 控制寄存器_2	SSIFCR_2	32	H'FFFF1010	8、16、32
	FIFO 状态寄存器_2	SSIFSR_2	32	H'FFFF1014	8、16、32
	发送 FIFO 数据寄存器_2	SSIFTDR_2	32	H'FFFF1018	32
	接收 FIFO 数据寄存器_2	SSIFRDR_2	32	H'FFFF101C	32
	TDM 模式寄存器_2	SSITDMR_2	32	H'FFFF1020	8、16、32
	控制寄存器_3	SSICR_3	32	H'FFFF1800	8、16、32
	状态寄存器_3	SSISR_3	32	H'FFFF1804	8、16、32
	FIFO 控制寄存器_3	SSIFCR_3	32	H'FFFF1810	8、16、32
FIFO 状态寄存器_3	SSIFSR_3	32	H'FFFF1814	8、16、32	

模块名	寄存器名	略称	位数	地址	存取长度
串行音频接口	发送 FIFO 数据寄存器 _3	SSIFTDR_3	32	H'FFFF1818	32
	接收 FIFO 数据寄存器 _3	SSIFRDR_3	32	H'FFFF181C	32
	TDM 模式寄存器 _3	SSITDMR_3	32	H'FFFF1820	8、16、32
	控制寄存器 _4	SSICR_4	32	H'FFFF2000	8、16、32
	状态寄存器 _4	SSISR_4	32	H'FFFF2004	8、16、32
	FIFO 控制寄存器 _4	SSIFCR_4	32	H'FFFF2010	8、16、32
	FIFO 状态寄存器 _4	SSIFSR_4	32	H'FFFF2014	8、16、32
	发送 FIFO 数据寄存器 _4	SSIFTDR_4	32	H'FFFF2018	32
	接收 FIFO 数据寄存器 _4	SSIFRDR_4	32	H'FFFF201C	32
	TDM 模式寄存器 _4	SSITDMR_4	32	H'FFFF2020	8、16、32
	控制寄存器 _5	SSICR_5	32	H'FFFF2800	8、16、32
	状态寄存器 _5	SSISR_5	32	H'FFFF2804	8、16、32
	FIFO 控制寄存器 _5	SSIFCR_5	32	H'FFFF2810	8、16、32
	FIFO 状态寄存器 _5	SSIFSR_5	32	H'FFFF2814	8、16、32
	发送 FIFO 数据寄存器 _5	SSIFTDR_5	32	H'FFFF2818	32
	接收 FIFO 数据寄存器 _5	SSIFRDR_5	32	H'FFFF281C	32
	TDM 模式寄存器 _5	SSITDMR_5	32	H'FFFF2820	8、16、32
带 FIFO 的时钟 同步串行 I/O	模式寄存器	SIMDR	16	H'FFFF4800	16
	时钟选择寄存器	SISCR	16	H'FFFF4802	16
	发送数据分配寄存器	SITDAR	16	H'FFFF4804	16
	接收数据分配寄存器	SIRDAR	16	H'FFFF4806	16
	控制寄存器	SICTR	16	H'FFFF480C	16
	FIFO 控制寄存器	SIFCTR	16	H'FFFF4810	16
	状态寄存器	SISTR	16	H'FFFF4814	16
	中断允许寄存器	SIIER	16	H'FFFF4816	16
	发送数据寄存器	SITDR	32	H'FFFF4820	8、16、32
	接收数据寄存器	SIRDAR	32	H'FFFF4824	8、16、32
控制器局域网	主控控制寄存器 _0	MCR_0	16	H'FFFE5000	16
	通用状态寄存器 _0	GSR_0	16	H'FFFE5002	16
	位配置寄存器 1_0	BCR1_0	16	H'FFFE5004	16
	位配置寄存器 0_0	BCR0_0	16	H'FFFE5006	16
	中断请求寄存器 _0	IRR_0	16	H'FFFE5008	16
	中断屏蔽寄存器 _0	IMR_0	16	H'FFFE500A	16
	发送 / 接收错误计数器 _0	TEC_REC_0	16	H'FFFE500C	8、16
	发送等待寄存器 1_0	TXPR1_0	16	H'FFFE5020	32
	发送等待寄存器 0_0	TXPR0_0	16	H'FFFE5022	16
	发送取消寄存器 1_0	TXCR1_0	16	H'FFFE5028	16
	发送取消寄存器 0_0	TXCR0_0	16	H'FFFE502A	16
	发送应答寄存器 1_0	TXACK1_0	16	H'FFFE5030	16
	发送应答寄存器 0_0	TXACK0_0	16	H'FFFE5032	16
	中止应答寄存器 1_0	ABACK1_0	16	H'FFFE5038	16
	中止应答寄存器 0_0	ABACK0_0	16	H'FFFE503A	16

模块名	寄存器名	略称	位数	地址	存取长度
控制器局域网	数据帧接收结束寄存器 1_0	RXPR1_0	16	H'FFFE5040	16
	数据帧接收结束寄存器 0_0	RXPR0_0	16	H'FFFE5042	16
	远程帧接收结束寄存器 1_0	RFPR1_0	16	H'FFFE5048	16
	远程帧接收结束寄存器 0_0	RFPR0_0	16	H'FFFE504A	16
	信箱中断屏蔽寄存器 1_0	MBIMR1_0	16	H'FFFE5050	16
	信箱中断屏蔽寄存器 0_0	MBIMR0_0	16	H'FFFE5052	16
	未读信息状态寄存器 1_0	UMSR1_0	16	H'FFFE5058	16
	未读信息状态寄存器 0_0	UMSR0_0	16	H'FFFE505A	16
	定时器的触发控制寄存器 0_0	TTCR0_0	16	H'FFFE5080	16
	最大周期/Tx_Enable_Window 寄存器_0	CMAX_TEW_0	16	H'FFFE5084	16
	基准触发偏移寄存器_0	RFTR0FF_0	16	H'FFFE5086	16
	定时器的状态寄存器_0	TSR_0	16	H'FFFE5088	16
	周期计数寄存器_0	CCR_0	16	H'FFFE508A	16
	定时器的计数寄存器_0	TCNTR_0	16	H'FFFE508C	16
	周期时间寄存器_0	CYCTR_0	16	H'FFFE5090	16
	基准标记寄存器_0	RFMK_0	16	H'FFFE5094	16
	时间比较匹配寄存器 0_0	TCMR0_0	16	H'FFFE5098	16
	时间比较匹配寄存器 1_0	TCMR1_0	16	H'FFFE509C	16
	时间比较匹配寄存器 2_0	TCMR2_0	16	H'FFFE50A0	16
	发送触发时间选择寄存器_0	TTTSEL_0	16	H'FFFE50A4	16
	信箱 n 控制 0H_0 (n=0 ~ 31)	MBn_CONTROL0_H_0 (n=0 ~ 31)	16	H'FFFE5100 +n×32	16、32
	信箱 n 控制 0L_0 (n=0 ~ 31)	MBn_CONTROL0_L_0 (n=0 ~ 31)	16	H'FFFE5102 +n×32	16
	信箱 n 本地接收滤波器屏蔽 0_0 (n=0 ~ 31)	MBn_LAFM0_0 (n=0 ~ 31)	16	H'FFFE5104 +n×32	16、32
	信箱 n 本地接收滤波器屏蔽 1_0 (n=0 ~ 31)	MBn_LAFM1_0 (n=0 ~ 31)	16	H'FFFE5106 +n×32	16
	信箱 n 数据 01_0 (n=0 ~ 31)	MBn_DATA_01_0 (n=0 ~ 31)	16	H'FFFE5108 +n×32	8、16、32
	信箱 n 数据 23_0 (n=0 ~ 31)	MBn_DATA_23_0 (n=0 ~ 31)	16	H'FFFE510A +n×32	8、16
	信箱 n 数据 45_0 (n=0 ~ 31)	MBn_DATA_45_0 (n=0 ~ 31)	16	H'FFFE510C +n×32	8、16、32
	信箱 n 数据 67_0 (n=0 ~ 31)	MBn_DATA_67_0 (n=0 ~ 31)	16	H'FFFE510E +n×32	8、16
	信箱 n 控制 1_0 (n=0 ~ 31)	MBn_CONTROL1_0 (n=0 ~ 31)	16	H'FFFE5110 +n×32	8、16
	信箱 n 时戳_0 (n=0 ~ 15、30、31)	MBn_TIMESTAMP_0 (n=0 ~ 15、30、31)	16	H'FFFE5112 +n×32	16
	信箱 n 触发时间_0 (n=24 ~ 30)	MBn_TTT_0 (n=24 ~ 30)	16	H'FFFE5114 +n×32	16
	信箱 nTT 控制_0 (n=24 ~ 29)	MBn_TTCONTROL_0 (n=24 ~ 29)	16	H'FFFE5116 +n×32	16

模块名	寄存器名	略称	位数	地址	存取长度
控制器局域网	主控控制寄存器_1	MCR_1	16	H'FFFE5800	16
	通用状态寄存器_1	GSR_1	16	H'FFFE5802	16
	位配置寄存器 1_1	BCR1_1	16	H'FFFE5804	16
	位配置寄存器 0_1	BCR0_1	16	H'FFFE5806	16
	中断请求寄存器_1	IRR_1	16	H'FFFE5808	16
	中断屏蔽寄存器_1	IMR_1	16	H'FFFE580A	16
	发送 / 接收错误计数器_1	TEC_REC_1	16	H'FFFE580C	8、16
	发送等待寄存器 1_1	TXPR1_1	16	H'FFFE5820	32
	发送等待寄存器 0_1	TXPR0_1	16	H'FFFE5822	16
	发送取消寄存器 1_1	TXCR1_1	16	H'FFFE5828	16
	发送取消寄存器 0_1	TXCR0_1	16	H'FFFE582A	16
	发送应答寄存器 1_1	TXACK1_1	16	H'FFFE5830	16
	发送应答寄存器 0_1	TXACK0_1	16	H'FFFE5832	16
	中止应答寄存器 1_1	ABACK1_1	16	H'FFFE5838	16
	中止应答寄存器 0_1	ABACK0_1	16	H'FFFE583A	16
	数据帧接收结束寄存器 1_1	RXPR1_1	16	H'FFFE5840	16
	数据帧接收结束寄存器 0_1	RXPR0_1	16	H'FFFE5842	16
	远程帧接收结束寄存器 1_1	RFPR1_1	16	H'FFFE5848	16
	远程帧接收结束寄存器 0_1	RFPR0_1	16	H'FFFE584A	16
	信箱中断屏蔽寄存器 1_1	MBIMR1_1	16	H'FFFE5850	16
	信箱中断屏蔽寄存器 0_1	MBIMR0_1	16	H'FFFE5852	16
	未读信息状态寄存器 1_1	UMSR1_1	16	H'FFFE5858	16
	未读信息状态寄存器 0_1	UMSR0_1	16	H'FFFE585A	16
	定时器的触发控制寄存器 0_1	TTCR0_1	16	H'FFFE5880	16
	最大周期 /Tx_Enable_Window 寄存器_1	CMAX_TEW_1	16	H'FFFE5884	16
	基准触发偏移寄存器_1	RFTR0FF_1	16	H'FFFE5886	16
	定时器的状态寄存器_1	TSR_1	16	H'FFFE5888	16
	周期计数寄存器_1	CCR_1	16	H'FFFE588A	16
	定时器的计数寄存器_1	TCNTR_1	16	H'FFFE588C	16
	周期时间寄存器_1	CYCTR_1	16	H'FFFE5890	16
	基准标记寄存器_1	RFMK_1	16	H'FFFE5894	16
	时间比较匹配寄存器 0_1	TCMR0_1	16	H'FFFE5898	16
	时间比较匹配寄存器 1_1	TCMR1_1	16	H'FFFE589C	16
	时间比较匹配寄存器 2_1	TCMR2_1	16	H'FFFE58A0	16
	发送触发时间选择寄存器_1	TTTSEL_1	16	H'FFFE58A4	16
	信箱 n 控制 0H_1 (n=0 ~ 31)	MBn_CONTROLO_H_1 (n=0 ~ 31)	16	H'FFFE5900 +n×32	16、32
	信箱 n 控制 0L_1 (n=0 ~ 31)	MBn_CONTROLO_L_1 (n=0 ~ 31)	16	H'FFFE5902 +n×32	16
	信箱 n 本地接收滤波器屏蔽 0_1 (n=0 ~ 31)	MBn_LAFM0_1 (n=0 ~ 31)	16	H'FFFE5904 +n×32	16、32
	信箱 n 本地接收滤波器屏蔽 1_1 (n=0 ~ 31)	MBn_LAFM1_1 (n=0 ~ 31)	16	H'FFFE5906 +n×32	16

模块名	寄存器名	略称	位数	地址	存取长度
控制器局域网	信箱 n 数据 01_1 (n=0 ~ 31)	MBn_DATA_01_1 (n=0 ~ 31)	16	H'FFFE5908 +n×32	8、16、32
	信箱 n 数据 23_1 (n=0 ~ 31)	MBn_DATA_23_1 (n=0 ~ 31)	16	H'FFFE590A +n×32	8、16
	信箱 n 数据 45_1 (n=0 ~ 31)	MBn_DATA_45_1 (n=0 ~ 31)	16	H'FFFE590C +n×32	8、16、32
	信箱 n 数据 67_1 (n=0 ~ 31)	MBn_DATA_67_1 (n=0 ~ 31)	16	H'FFFE590E +n×32	8、16
	信箱 n 控制 1_1 (n=0 ~ 31)	MBn_CONTROL1_1 (n=0 ~ 31)	16	H'FFFE5910 +n×32	8、16
	信箱 n 时戳_1 (n=0 ~ 15、30、31)	MBn_TIMESTAMP_1 (n=0 ~ 15、30、31)	16	H'FFFE5912 +n×32	16
	信箱 n 触发时间_1 (n=24 ~ 30)	MBn_TTT_1 (n=24 ~ 30)	16	H'FFFE5914 +n×32	16
	信箱 nTT 控制_1 (n=24 ~ 29)	MBn_TTCONTROL_1 (n=24 ~ 29)	16	H'FFFE5916 +n×32	16
	主控控制寄存器_2	MCR_2	16	H'FFFD800	16
	通用状态寄存器_2	GSR_2	16	H'FFFD802	16
	位配置寄存器 1_2	BCR1_2	16	H'FFFD804	16
	位配置寄存器 0_2	BCR0_2	16	H'FFFD806	16
	中断请求寄存器_2	IRR_2	16	H'FFFD808	16
	中断屏蔽寄存器_2	IMR_2	16	H'FFFD80A	16
	发送 / 接收错误计数器_2	TEC_REC_2	16	H'FFFD80C	8、16
	发送等待寄存器 1_2	TXPR1_2	16	H'FFFD820	32
	发送等待寄存器 0_2	TXPR0_2	16	H'FFFD822	16
	发送取消寄存器 1_2	TXCR1_2	16	H'FFFD828	16
	发送取消寄存器 0_2	TXCR0_2	16	H'FFFD82A	16
	发送应答寄存器 1_2	TXACK1_2	16	H'FFFD830	16
	发送应答寄存器 0_2	TXACK0_2	16	H'FFFD832	16
	中止应答寄存器 1_2	ABACK1_2	16	H'FFFD838	16
	中止应答寄存器 0_2	ABACK0_2	16	H'FFFD83A	16
	数据帧接收结束寄存器 1_2	RXPR1_2	16	H'FFFD840	16
	数据帧接收结束寄存器 0_2	RXPR0_2	16	H'FFFD842	16
	远程帧接收结束寄存器 1_2	RFPR1_2	16	H'FFFD848	16
	远程帧接收结束寄存器 0_2	RFPR0_2	16	H'FFFD84A	16
	信箱中断屏蔽寄存器 1_2	MBIMR1_2	16	H'FFFD850	16
	信箱中断屏蔽寄存器 0_2	MBIMR0_2	16	H'FFFD852	16
	未读信息状态寄存器 1_2	UMSR1_2	16	H'FFFD858	16
	未读信息状态寄存器 0_2	UMSR0_2	16	H'FFFD85A	16
	定时器的触发控制寄存器 0_2	TTCR0_2	16	H'FFFD880	16
	最大周期 /Tx_Enable_Window 寄存器_2	CMAX_TEW_2	16	H'FFFD884	16
	基准触发偏移寄存器_2	RFTROFF_2	16	H'FFFD886	16
	定时器状态寄存器_2	TSR_2	16	H'FFFD888	16
	周期计数寄存器_2	CCR_2	16	H'FFFD88A	16

模块名	寄存器名	略称	位数	地址	存取长度
控制器局域网	定时器的计数寄存器_2	TCNTR_2	16	H'FFFED88C	16
	周期时间寄存器_2	CYCTR_2	16	H'FFFED890	16
	基准标记寄存器_2	RFMK_2	16	H'FFFED894	16
	时间比较匹配寄存器 0_2	TCMR0_2	16	H'FFFED898	16
	时间比较匹配寄存器 1_2	TCMR1_2	16	H'FFFED89C	16
	时间比较匹配寄存器 2_2	TCMR2_2	16	H'FFFED8A0	16
	发送触发时间选择寄存器_2	TTTSEL_2	16	H'FFFED8A4	16
	信箱 n 控制 0H_2 (n=0 ~ 31)	MBn_CONTROL0_H_2 (n=0 ~ 31)	16	H'FFFED900 +n×32	16、32
	信箱 n 控制 0L_2 (n=0 ~ 31)	MBn_CONTROL0_L_2 (n=0 ~ 31)	16	H'FFFED902 +n×32	16
	信箱 n 本地接收滤波器屏蔽 0_2 (n=0 ~ 31)	MBn_LAFM0_2 (n=0 ~ 31)	16	H'FFFED904 +n×32	16、32
	信箱 n 本地接收滤波器屏蔽 1_2 (n=0 ~ 31)	MBn_LAFM1_2 (n=0 ~ 31)	16	H'FFFED906 +n×32	16
	信箱 n 数据 01_2 (n=0 ~ 31)	MBn_DATA_01_2 (n=0 ~ 31)	16	H'FFFED908 +n×32	8、16、32
	信箱 n 数据 23_2 (n=0 ~ 31)	MBn_DATA_23_2 (n=0 ~ 31)	16	H'FFFED90A +n×32	8、16
	信箱 n 数据 45_2 (n=0 ~ 31)	MBn_DATA_45_2 (n=0 ~ 31)	16	H'FFFED90C +n×32	8、16、32
	信箱 n 数据 67_2 (n=0 ~ 31)	MBn_DATA_67_2 (n=0 ~ 31)	16	H'FFFED90E +n×32	8、16
	信箱 n 控制 1_2 (n=0 ~ 31)	MBn_CONTROL1_2 (n=0 ~ 31)	16	H'FFFED910 +n×32	8、16
	信箱 n 时戳_2 (n=0 ~ 15、30、31)	MBn_TIMESTAMP_2 (n=0 ~ 15、30、31)	16	H'FFFED912 +n×32	16
	信箱 n 触发时间_2 (n=24 ~ 30)	MBn_TTT_2 (n=24 ~ 30)	16	H'FFFED914 +n×32	16
信箱 nTT 控制_2 (n=24 ~ 29)	MBn_TTCONTROL_2 (n=24 ~ 29)	16	H'FFFED916 +n×32	16	
IEBus 控制器	IEBus 控制寄存器	IECTR	8	H'FFFEF000	8
	IEBus 命令寄存器	IECMR	8	H'FFFEF001	8
	IEBus 主控控制寄存器	IEMCR	8	H'FFFEF002	8
	IEBus 本站地址寄存器 1	IEAR1	8	H'FFFEF003	8
	IEBus 本站地址寄存器 2	IEAR2	8	H'FFFEF004	8
	IEBus 从属地址设定寄存器 1	IESA1	8	H'FFFEF005	8
	IEBus 从属地址设定寄存器 2	IESA2	8	H'FFFEF006	8
	IEBus 发送信息长度寄存器	IETBFL	8	H'FFFEF007	8
	IEBus 接收主控地址寄存器 1	IEMA1	8	H'FFFEF009	8
	IEBus 接收主控地址寄存器 2	IEMA2	8	H'FFFEF00A	8
	IEBus 接收控制字段寄存器	IERCTL	8	H'FFFEF00B	8
	IEBus 接收信息长度寄存器	IERBFL	8	H'FFFEF00C	8
	IEBus 锁定地址寄存器 1	IELA1	8	H'FFFEF00E	8

模块名	寄存器名	略称	位数	地址	存取长度
IEBus 控制器	IEBus 锁定地址寄存器 2	IELA2	8	H'FFFEF00F	8
	IEBus 通用标志寄存器	IEFLG	8	H'FFFEF010	8
	IEBus 发送状态寄存器	IETSR	8	H'FFFEF011	8
	IEBus 发送中断允许寄存器	IEIET	8	H'FFFEF012	8
	IEBus 接收状态寄存器	IERSR	8	H'FFFEF014	8
	IEBus 接收中断允许寄存器	IEIER	8	H'FFFEF015	8
	IEBus 时钟选择寄存器	IECKSR	8	H'FFFEF018	8
	IEBus 发送数据缓冲器 001 ~ 128	IETB001 ~ IETB128	8	H'FFFEF100 ~ H'FFFEF17F	8
	IEBus 接收数据缓冲器 001 ~ 128	IERB001 ~ IERB128	8	H'FFFEF200 ~ H'FFFEF27F	8
瑞萨 SPDIF 接口	发送模块通道 1 的音频寄存器	TLCA	32	H'E8012000	32
	发送模块通道 2 的音频寄存器	TRCA	32	H'E8012004	32
	发送模块通道 1 的状态寄存器	TLCS	32	H'E8012008	32
	发送模块通道 2 的状态寄存器	TRCS	32	H'E801200C	32
	发送用户数据寄存器	TUI	32	H'E8012010	32
	接收模块通道 1 的音频寄存器	RLCA	32	H'E8012014	32
	接收模块通道 2 的音频寄存器	RRCA	32	H'E8012018	32
	接收模块通道 1 的状态寄存器	RLCS	32	H'E801201C	32
	接收模块通道 2 的状态寄存器	RRCS	32	H'E8012020	32
	接收用户数据寄存器	RUI	32	H'E8012024	32
	控制寄存器	CTRL	32	H'E8012028	32
	状态寄存器	STAT	32	H'E801202C	32
	发送模块 DMA 的音频数据寄存器	TDAD	32	H'E8012030	32
	接收模块 DMA 的音频数据寄存器	RDAD	32	H'E8012034	32
CD-ROM 解码器	允许控制寄存器	CROMEN	8	H'E8005000	8
	同步码的同步控制寄存器	CROMSY0	8	H'E8005001	8
	解码模式控制寄存器	CROMCTL0	8	H'E8005002	8
	EDC/ECC 校验控制寄存器	CROMCTL1	8	H'E8005003	8
	解码处理自动停止控制寄存器	CROMCTL3	8	H'E8005005	8
	解码选项设定控制寄存器	CROMCTL4	8	H'E8005006	8
	HEAD20 ~ 22 表示控制寄存器	CROMCTL5	8	H'E8005007	8
	同步码状态寄存器	CROMST0	8	H'E8005008	8
	ECC 后的标头错误状态寄存器	CROMST1	8	H'E8005009	8
	ECC 后的副标头错误状态寄存器	CROMST3	8	H'E800500B	8
	标头 / 副标头数据有效性判定状态寄存器	CROMST4	8	H'E800500C	8
	模式判定结果和链路扇区检测状态寄存器	CROMST5	8	H'E800500D	8
	ECC/EDC 错误状态寄存器	CROMST6	8	H'E800500E	8
	缓冲状态寄存器	CBUFST0	8	H'E8005014	8
	解码中止源状态寄存器	CBUFST1	8	H'E8005015	8
	缓冲上溢状态寄存器	CBUFST2	8	H'E8005016	8
	ECC 校正前标头部 -MINUTES 数据寄存器	HEAD00	8	H'E8005018	8
	ECC 校正前标头部 -SECONDS 数据寄存器	HEAD01	8	H'E8005019	8

模块名	寄存器名	略称	位数	地址	存取长度
CD-ROM 解码器	ECC 校正前标头部 -FRAMES (1/75 秒) 数据寄存器	HEAD02	8	H'E800501A	8
	ECC 校正前标头部 -MODE 数据寄存器	HEAD03	8	H'E800501B	8
	ECC 校正前副标头部 - 文件号 (BYTE-16) 数据寄存器	SHEAD00	8	H'E800501C	8
	ECC 校正前副标头部 - 通道号 (BYTE-17) 数据寄存器	SHEAD01	8	H'E800501D	8
	ECC 校正前副标头部 - 子模式 (BYTE-18) 数据寄存器	SHEAD02	8	H'E800501E	8
	ECC 校正前副标头部 - 数据类型 (BYTE-19) 数据寄存器	SHEAD03	8	H'E800501F	8
	ECC 校正前副标头部 - 文件号 (BYTE-20) 数据寄存器	SHEAD04	8	H'E8005020	8
	ECC 校正前副标头部 - 通道号 (BYTE-21) 数据寄存器	SHEAD05	8	H'E8005021	8
	ECC 校正前副标头部 - 子模式 (BYTE-22) 数据寄存器	SHEAD06	8	H'E8005022	8
	ECC 校正前副标头部 - 数据类型 (BYTE-23) 数据寄存器	SHEAD07	8	H'E8005023	8
	ECC 校正后标头部 -MINUTES 数据寄存器	HEAD20	8	H'E8005024	8
	ECC 校正后标头部 -SECONDS 数据寄存器	HEAD21	8	H'E8005025	8
	ECC 校正后标头部 -FRAMES (1/75 秒) 数据寄存器	HEAD22	8	H'E8005026	8
	ECC 校正后标头部 -MODE 数据寄存器	HEAD23	8	H'E8005027	8
	ECC 校正后副标头部 - 文件号 (BYTE-16) 数据寄存器	SHEAD20	8	H'E8005028	8
	ECC 校正后副标头部 - 通道号 (BYTE-17) 数据寄存器	SHEAD21	8	H'E8005029	8
	ECC 校正后副标头部 - 子模式 (BYTE-18) 数据寄存器	SHEAD22	8	H'E800502A	8
	ECC 校正后副标头部 - 数据类型 (BYTE-19) 数据寄存器	SHEAD23	8	H'E800502B	8
	ECC 校正后副标头部 - 文件号 (BYTE-20) 数据寄存器	SHEAD24	8	H'E800502C	8
	ECC 校正后副标头部 - 通道号 (BYTE-21) 数据寄存器	SHEAD25	8	H'E800502D	8
	ECC 校正后副标头部 - 子模式 (BYTE-22) 数据寄存器	SHEAD26	8	H'E800502E	8
	ECC 校正后副标头部 - 数据类 (BYTE-23) 数据寄存器	SHEAD27	8	H'E800502F	8
	自动缓冲设定控制寄存器	CBUFCTL0	8	H'E8005040	8
	自动缓冲起始扇区设定 -MINUTES 控制 寄存器	CBUFCTL1	8	H'E8005041	8
	自动缓冲起始扇区设定 -SECONDS 控制 寄存器	CBUFCTL2	8	H'E8005042	8

模块名	寄存器名	略称	位数	地址	存取长度
CD-ROM 解码器	自动缓冲起始扇区设定 -FRAMES 控制寄存器	CBUFCTL3	8	H'E8005043	8
	ISY 中断源屏蔽控制寄存器	CROMSTOM	8	H'E8005045	8
	CD-ROM 解码器模块复位控制寄存器	ROMDECRST	8	H'E8005100	8
	CD-ROM 解码器模块复位状态寄存器	RSTSTAT	8	H'E8005101	8
	串行音频接口数据控制寄存器	SSI	8	H'E8005102	8
	中断标志寄存器	INTHOLD	8	H'E8005108	8
	中断源屏蔽控制寄存器	INHINT	8	H'E8005109	8
	CD-ROM 解码器流数据输入寄存器 0	STRMDIN0	16	H'E8005200	16、32*
	CD-ROM 解码器流数据输入寄存器 2	STRMDIN2	16	H'E8005202	16
	CD-ROM 解码器流数据输出寄存器	STRMDOUT0	16	H'E8005204	16、32
A/D 转换器	A/D 数据寄存器 A	ADDRA	16	H'E8005800	16
	A/D 数据寄存器 B	ADDRB	16	H'E8005802	16
	A/D 数据寄存器 C	ADDRC	16	H'E8005804	16
	A/D 数据寄存器 D	ADDRD	16	H'E8005806	16
	A/D 数据寄存器 E	ADDRE	16	H'E8005808	16
	A/D 数据寄存器 F	ADDRF	16	H'E800580A	16
	A/D 数据寄存器 G	ADDRG	16	H'E800580C	16
	A/D 数据寄存器 H	ADDRH	16	H'E800580E	16
	A/D 控制 / 状态寄存器	ADCSR	16	H'E8005820	16
NAND 闪存 控制器	公共控制寄存器	FLCMNCR	32	H'FFFF4000	32
	命令控制寄存器	FLCMDCR	32	H'FFFF4004	32
	命令码寄存器	FLCMCDR	32	H'FFFF4008	32
	地址寄存器	FLADR	32	H'FFFF400C	32
	地址寄存器 2	FLADR2	32	H'FFFF403C	32
	数据寄存器	FLDATAR	32	H'FFFF4010	32
	数据计数寄存器	FLDTCNTR	32	H'FFFF4014	32
	中断 DMA 控制寄存器	FLINTDMACR	32	H'FFFF4018	32
	就绪 / 忙超时设定寄存器	FLBSYTMR	32	H'FFFF401C	32
	就绪 / 忙超时计数器	FLBSYCNT	32	H'FFFF4020	32
	数据 FIFO 寄存器	FLDTFIFO	32	H'FFFF4050	32
	管理码 FIFO 寄存器	FLECFIFO	32	H'FFFF4060	32
	传送控制寄存器	FLTRCR	8	H'FFFF402C	8
	总线占有期间设定寄存器	FLHOLDCR	32	H'FFFF4038	32
	USB2.0 主机 / 功能模块	系统配置控制寄存器	SYSCFG	16	H'E8010000
CPU 总线等待设定寄存器		BUSWAIT	16	H'E8010002	16
系统配置状态寄存器		SYSSTS	16	H'E8010004	16
设备状态控制寄存器		DVSTCTR	16	H'E8010008	16
测试模式寄存器		TESTMODE	16	H'E801000C	16
DMA0-FIFO 总线配置寄存器		D0FBCFG	16	H'E8010010	16
DMA1-FIFO 总线配置寄存器		D1FBCFG	16	H'E8010012	16
CFIFO 端口寄存器		CFIFO	32	H'E8010014	8、16、32

模块名	寄存器名	略称	位数	地址	存取长度
USB2.0 主机 / 功能模块	D0FIFO 端口寄存器	D0FIFO	32	H'E8010018	8、16、32
	D1FIFO 端口寄存器	D1FIFO	32	H'E801001C	8、16、32
	CFIFO 端口选择寄存器	CFIFOSEL	16	H'E8010020	16
	CFIFO 端口控制寄存器	CFIFOCTR	16	H'E8010022	16
	D0FIFO 端口选择寄存器	D0FIFOSEL	16	H'E8010028	16
	D0FIFO 端口控制寄存器	D0FIFOCTR	16	H'E801002A	16
	D1FIFO 端口选择寄存器	D1FIFOSEL	16	H'E801002C	16
	D1FIFO 端口控制寄存器	D1FIFOCTR	16	H'E801002E	16
	中断允许寄存器 0	INTENB0	16	H'E8010030	16
	中断允许寄存器 1	INTENB1	16	H'E8010032	16
	BRDY 中断允许寄存器	BRDYENB	16	H'E8010036	16
	NRDY 中断允许寄存器	NRDYENB	16	H'E8010038	16
	BEMP 中断允许寄存器	BEMPENB	16	H'E801003A	16
	SOF 输出配置寄存器	SOFCFG	16	H'E801003C	16
	中断状态寄存器 0	INTSTS0	16	H'E8010040	16
	中断状态寄存器 1	INTSTS1	16	H'E8010042	16
	BRDY 中断状态寄存器	BRDYSTS	16	H'E8010046	16
	NRDY 中断状态寄存器	NRDYSTS	16	H'E8010048	16
	BEMP 中断状态寄存器	BEMPSTS	16	H'E801004A	16
	帧号寄存器	FRMNUM	16	H'E801004C	16
	μ 帧号寄存器	UFRMNUM	16	H'E801004E	16
	USB 地址寄存器	USBADDR	16	H'E8010050	16
	USB 请求类型寄存器	USBREQ	16	H'E8010054	16
	USB 请求值寄存器	USBVAL	16	H'E8010056	16
	USB 请求变址寄存器	USBINDX	16	H'E8010058	16
	USB 请求长度寄存器	USBLENG	16	H'E801005A	16
	DCP 配置寄存器	DCPCFG	16	H'E801005C	16
	DCP 最大信息包长度寄存器	DCPMAXP	16	H'E801005E	16
	DCP 控制寄存器	DCPCTR	16	H'E8010060	16
	管道窗口选择寄存器	PIPESEL	16	H'E8010064	16
	管道配置寄存器	PIPECFG	16	H'E8010068	16
	管道缓冲指定寄存器	PIPEBUF	16	H'E801006A	16
	管道最大信息包长度寄存器	PIPEMAXP	16	H'E801006C	16
	管道周期控制寄存器	PIPEPERI	16	H'E801006E	16
	管道 1 的控制寄存器	PIPE1CTR	16	H'E8010070	16
	管道 2 的控制寄存器	PIPE2CTR	16	H'E8010072	16
	管道 3 的控制寄存器	PIPE3CTR	16	H'E8010074	16
	管道 4 的控制寄存器	PIPE4CTR	16	H'E8010076	16
	管道 5 的控制寄存器	PIPE5CTR	16	H'E8010078	16
	管道 6 的控制寄存器	PIPE6CTR	16	H'E801007A	16
管道 7 的控制寄存器	PIPE7CTR	16	H'E801007C	16	
管道 8 的控制寄存器	PIPE8CTR	16	H'E801007E	16	

模块名	寄存器名	略称	位数	地址	存取长度
USB2.0 主机 / 功能模块	管道 9 的控制寄存器	PIPE9CTR	16	H'E8010080	16
	管道 1 的事务计数允许寄存器	PIPE1TRE	16	H'E8010090	16
	管道 1 的事务计数寄存器	PIPE1TRN	16	H'E8010092	16
	管道 2 的事务计数允许寄存器	PIPE2TRE	16	H'E8010094	16
	管道 2 的事务计数寄存器	PIPE2TRN	16	H'E8010096	16
	管道 3 的事务计数允许寄存器	PIPE3TRE	16	H'E8010098	16
	管道 3 的事务计数寄存器	PIPE3TRN	16	H'E801009A	16
	管道 4 的事务计数允许寄存器	PIPE4TRE	16	H'E801009C	16
	管道 4 的事务计数寄存器	PIPE4TRN	16	H'E801009E	16
	管道 5 的事务计数允许寄存器	PIPE5TRE	16	H'E80100A0	16
	管道 5 的事务计数寄存器	PIPE5TRN	16	H'E80100A2	16
	设备地址 0 的配置寄存器	DEVADD0	16	H'E80100D0	16
	设备地址 1 的配置寄存器	DEVADD1	16	H'E80100D2	16
	设备地址 2 的配置寄存器	DEVADD2	16	H'E80100D4	16
	设备地址 3 的配置寄存器	DEVADD3	16	H'E80100D6	16
	设备地址 4 的配置寄存器	DEVADD4	16	H'E80100D8	16
	设备地址 5 的配置寄存器	DEVADD5	16	H'E80100DA	16
	设备地址 6 的配置寄存器	DEVADD6	16	H'E80100DC	16
	设备地址 7 的配置寄存器	DEVADD7	16	H'E80100DE	16
	设备地址 8 的配置寄存器	DEVADD8	16	H'E80100E0	16
设备地址 9 的配置寄存器	DEVADD9	16	H'E80100E2	16	
设备地址 A 的配置寄存器	DEVADDA	16	H'E80100E4	16	
数字视频解码器	ADC 控制寄存器 1	ADCCR1	16	H'FFFFFFA008	16
	时序生成控制寄存器 (1)	TGCR1	16	H'FFFFFFA00E	16
	时序生成控制寄存器 (2)	TGCR2	16	H'FFFFFFA010	16
	时序生成控制寄存器 (3)	TGCR3	16	H'FFFFFFA012	16
	同步分离控制寄存器 (1)	SYNSCR1	16	H'FFFFFFA01A	16
	同步分离控制寄存器 (2)	SYNSCR2	16	H'FFFFFFA01C	16
	同步分离控制寄存器 (3)	SYNSCR3	16	H'FFFFFFA01E	16
	同步分离控制寄存器 (4)	SYNSCR4	16	H'FFFFFFA020	16
	同步分离控制寄存器 (5)	SYNSCR5	16	H'FFFFFFA022	16
	水平 AFC 控制寄存器 (1)	HAFCCR1	16	H'FFFFFFA024	16
	水平 AFC 控制寄存器 (2)	HAFCCR2	16	H'FFFFFFA026	16
	水平 AFC 控制寄存器 (3)	HAFCCR3	16	H'FFFFFFA028	16
	垂直递减计数控制寄存器 (1)	VCDWCR1	16	H'FFFFFFA02A	16
	数字钳位控制寄存器 (1)	DCPCR1	16	H'FFFFFFA030	16
	数字钳位控制寄存器 (2)	DCPCR2	16	H'FFFFFFA032	16
	数字钳位控制寄存器 (3)	DCPCR3	16	H'FFFFFFA034	16
	数字钳位控制寄存器 (4)	DCPCR4	16	H'FFFFFFA036	16
	数字钳位控制寄存器 (5)	DCPCR5	16	H'FFFFFFA038	16
	数字钳位控制寄存器 (6)	DCPCR6	16	H'FFFFFFA03A	16
	数字钳位控制寄存器 (7)	DCPCR7	16	H'FFFFFFA03C	16

模块名	寄存器名	略称	位数	地址	存取长度
数字视频解码器	数字钳位控制寄存器 (8)	DCPCR8	16	H'FFFA03E	16
	噪声检测控制寄存器	NSDCR	16	H'FFFA040	16
	色同步锁定控制 / 色度解码控制寄存器	BTLCR	16	H'FFFA042	16
	色同步选通脉冲控制寄存器	BTGPCR	16	H'FFFA044	16
	ACC 控制寄存器 (1)	ACCCR1	16	H'FFFA046	16
	ACC 控制寄存器 (2)	ACCCR2	16	H'FFFA048	16
	ACC 控制寄存器 (3)	ACCCR3	16	H'FFFA04A	16
	TINT 控制寄存器	TINTCR	16	H'FFFA04C	16
	Y/C 延迟控制 / 色度解码控制寄存器	YCDCR	16	H'FFFA04E	16
	AGC 控制寄存器 (1)	AGCCR1	16	H'FFFA050	16
	AGC 控制寄存器 (2)	AGCCR2	16	H'FFFA052	16
	峰值限制器的控制寄存器	PKLIMITCR	16	H'FFFA054	16
	超范围控制寄存器 (1)	RGORCR1	16	H'FFFA056	16
	超范围控制寄存器 (2)	RGORCR2	16	H'FFFA058	16
	超范围控制寄存器 (3)	RGORCR3	16	H'FFFA05A	16
	超范围控制寄存器 (4)	RGORCR4	16	H'FFFA05C	16
	超范围控制寄存器 (5)	RGORCR5	16	H'FFFA05E	16
	超范围控制寄存器 (6)	RGORCR6	16	H'FFFA060	16
	超范围控制寄存器 (7)	RGORCR7	16	H'FFFA062	16
	水平 AFC 相位比较器的反馈量调整寄存器	AFCPFCR	16	H'FFFA07C	16
	寄存器的更新控制寄存器	RUPDCR	16	H'FFFA07E	16
	同步分离状态 / 垂直周期读寄存器	VSYNCSR	16	H'FFFA080	16
	水平周期读寄存器	HSYNCSR	16	H'FFFA082	16
	数字钳位读寄存器 (1)	DCPSR1	16	H'FFFA084	16
	数字钳位读寄存器 (2)	DCPSR2	16	H'FFFA086	16
	噪声检测读寄存器	NSDSR	16	H'FFFA08C	16
	色度解码读寄存器 (1)	CROMASR1	16	H'FFFA08E	16
	色度解码读寄存器 (2)	CROMASR2	16	H'FFFA090	16
	同步分离读寄存器	SYNCSR	16	H'FFFA092	16
	AGC 控制读寄存器 (1)	AGCSR1	16	H'FFFA094	16
	AGC 控制读寄存器 (2)	AGCSR2	16	H'FFFA096	16
	Y/C 分离控制寄存器 (3)	YCSCR3	16	H'FFFA104	16
	Y/C 分离控制寄存器 (4)	YCSCR4	16	H'FFFA106	16
	Y/C 分离控制寄存器 (5)	YCSCR5	16	H'FFFA108	16
	Y/C 分离控制寄存器 (6)	YCSCR6	16	H'FFFA10A	16
	Y/C 分离控制寄存器 (7)	YCSCR7	16	H'FFFA10C	16
	Y/C 分离控制寄存器 (8)	YCSCR8	16	H'FFFA10E	16
	Y/C 分离控制寄存器 (9)	YCSCR9	16	H'FFFA110	16
	Y/C 分离控制寄存器 (11)	YCSCR11	16	H'FFFA114	16
	Y/C 分离控制寄存器 (12)	YCSCR12	16	H'FFFA116	16
数字钳位控制寄存器 (9)	DCPCR9	16	H'FFFA180	16	

模块名	寄存器名	略称	位数	地址	存取长度
数字视频解码器	Y/C 分离的色度信号滤波器 TAP 系数 (WA_F0) 寄存器	YCTWA_F0	16	H'FFFFFFA192	16
	Y/C 分离的色度信号滤波器 TAP 系数 (WA_F1) 寄存器	YCTWA_F1	16	H'FFFFFFA194	16
	Y/C 分离的色度信号滤波器 TAP 系数 (WA_F2) 寄存器	YCTWA_F2	16	H'FFFFFFA196	16
	Y/C 分离的色度信号滤波器 TAP 系数 (WA_F3) 寄存器	YCTWA_F3	16	H'FFFFFFA198	16
	Y/C 分离的色度信号滤波器 TAP 系数 (WA_F4) 寄存器	YCTWA_F4	16	H'FFFFFFA19A	16
	Y/C 分离的色度信号滤波器 TAP 系数 (WA_F5) 寄存器	YCTWA_F5	16	H'FFFFFFA19C	16
	Y/C 分离的色度信号滤波器 TAP 系数 (WA_F6) 寄存器	YCTWA_F6	16	H'FFFFFFA19E	16
	Y/C 分离的色度信号滤波器 TAP 系数 (WA_F7) 寄存器	YCTWA_F7	16	H'FFFFFFA1A0	16
	Y/C 分离的色度信号滤波器 TAP 系数 (WA_F8) 寄存器	YCTWA_F8	16	H'FFFFFFA1A2	16
	Y/C 分离的色度信号滤波器 TAP 系数 (WB_F0) 寄存器	YCTWB_F0	16	H'FFFFFFA1A4	16
	Y/C 分离的色度信号滤波器 TAP 系数 (WB_F1) 寄存器	YCTWB_F1	16	H'FFFFFFA1A6	16
	Y/C 分离的色度信号滤波器 TAP 系数 (WB_F2) 寄存器	YCTWB_F2	16	H'FFFFFFA1A8	16
	Y/C 分离的色度信号滤波器 TAP 系数 (WB_F3) 寄存器	YCTWB_F3	16	H'FFFFFFA1AA	16
	Y/C 分离的色度信号滤波器 TAP 系数 (WB_F4) 寄存器	YCTWB_F4	16	H'FFFFFFA1AC	16
	Y/C 分离的色度信号滤波器 TAP 系数 (WB_F5) 寄存器	YCTWB_F5	16	H'FFFFFFA1AE	16
	Y/C 分离的色度信号滤波器 TAP 系数 (WB_F6) 寄存器	YCTWB_F6	16	H'FFFFFFA1B0	16
	Y/C 分离的色度信号滤波器 TAP 系数 (WB_F7) 寄存器	YCTWB_F7	16	H'FFFFFFA1B2	16
	Y/C 分离的色度信号滤波器 TAP 系数 (WB_F8) 寄存器	YCTWB_F8	16	H'FFFFFFA1B4	16
	Y/C 分离的色度信号滤波器 TAP 系数 (NA_F0) 寄存器	YCTNA_F0	16	H'FFFFFFA1B6	16
	Y/C 分离的色度信号滤波器 TAP 系数 (NA_F1) 寄存器	YCTNA_F1	16	H'FFFFFFA1B8	16
Y/C 分离的色度信号滤波器 TAP 系数 (NA_F2) 寄存器	YCTNA_F2	16	H'FFFFFFA1BA	16	
Y/C 分离的色度信号滤波器 TAP 系数 (NA_F3) 寄存器	YCTNA_F3	16	H'FFFFFFA1BC	16	

模块名	寄存器名	略称	位数	地址	存取长度
数字视频解码器	Y/C 分离的色度信号滤波器 TAP 系数 (NA_F4) 寄存器	YCTNA_F4	16	H'FFFA1BE	16
	Y/C 分离的色度信号滤波器 TAP 系数 (NA_F5) 寄存器	YCTNA_F5	16	H'FFFA1C0	16
	Y/C 分离的色度信号滤波器 TAP 系数 (NA_F6) 寄存器	YCTNA_F6	16	H'FFFA1C2	16
	Y/C 分离的色度信号滤波器 TAP 系数 (NA_F7) 寄存器	YCTNA_F7	16	H'FFFA1C4	16
	Y/C 分离的色度信号滤波器 TAP 系数 (NA_F8) 寄存器	YCTNA_F8	16	H'FFFA1C6	16
	Y/C 分离的色度信号滤波器 TAP 系数 (NB_F0) 寄存器	YCTNB_F0	16	H'FFFA1C8	16
	Y/C 分离的色度信号滤波器 TAP 系数 (NB_F1) 寄存器	YCTNB_F1	16	H'FFFA1CA	16
	Y/C 分离的色度信号滤波器 TAP 系数 (NB_F2) 寄存器	YCTNB_F2	16	H'FFFA1CC	16
	Y/C 分离的色度信号滤波器 TAP 系数 (NB_F3) 寄存器	YCTNB_F3	16	H'FFFA1CE	16
	Y/C 分离的色度信号滤波器 TAP 系数 (NB_F4) 寄存器	YCTNB_F4	16	H'FFFA1D0	16
	Y/C 分离的色度信号滤波器 TAP 系数 (NB_F5) 寄存器	YCTNB_F5	16	H'FFFA1D2	16
	Y/C 分离的色度信号滤波器 TAP 系数 (NB_F6) 寄存器	YCTNB_F6	16	H'FFFA1D4	16
	Y/C 分离的色度信号滤波器 TAP 系数 (NB_F7) 寄存器	YCTNB_F7	16	H'FFFA1D6	16
	Y/C 分离的色度信号滤波器 TAP 系数 (NB_F8) 寄存器	YCTNB_F8	16	H'FFFA1D8	16
	亮度 (Y) 信号增益调整控制寄存器	YGAINCR	16	H'FFFA200	16
	色差 (CB) 信号增益调整控制寄存器	CBGAINCR	16	H'FFFA202	16
	色差 (CR) 信号增益调整控制寄存器	CRGAINCR	16	H'FFFA204	16
	PGA 部寄存器更新控制	PGA_UPDATE	16	H'FFFA280	16
	PGA 控制寄存器	PGACR	16	H'FFFA282	16
	ADC 控制寄存器 2	ADCCR2	16	H'FFFA284	16
视频显示 控制器 4	外部输入部的寄存器更新控制寄存器	INP_UPDATE	32	H'FFFF7400	16、32
	输入选择控制寄存器	INP_SEL_CNT	32	H'FFFF7404	16、32
	外部输入同步信号控制寄存器	INP_EXT_SYNC_CNT	32	H'FFFF7408	16、32
	垂直同步信号相位调整寄存器	INP_VSYNC_PH_ADJ	32	H'FFFF740C	16、32
	同步信号延迟调整寄存器	INP_DLY_ADJ	32	H'FFFF7410	16、32
	图像质量调整部的寄存器更新控制寄存器	IMGCNT_UPDATE	32	H'FFFF7480	16、32
	NR 控制寄存器 0	IMGCNT_NR_CNT0	32	H'FFFF7484	16、32
	NR 控制寄存器 1	IMGCNT_NR_CNT1	32	H'FFFF7488	16、32
图像质量调整部的矩阵模式寄存器	IMGCNT_MTX_MODE	32	H'FFFF74A0	16、32	

模块名	寄存器名	略称	位数	地址	存取长度
视频显示 控制器 4	图像质量调整部的矩阵 YG 调整寄存器 0	IMGCNT_MTX_YG_ADJ0	32	H'FFFF74A4	16、32
	图像质量调整部的矩阵 YG 调整寄存器 1	IMGCNT_MTX_YG_ADJ1	32	H'FFFF74A8	16、32
	图像质量调整部的矩阵 CBB 调整寄存器 0	IMGCNT_MTX_CBB_ADJ0	32	H'FFFF74AC	16、32
	图像质量调整部的矩阵 CBB 调整寄存器 1	IMGCNT_MTX_CBB_ADJ1	32	H'FFFF74B0	16、32
	图像质量调整部的矩阵 CRR 调整寄存器 0	IMGCNT_MTX_CRR_ADJ0	32	H'FFFF74B4	16、32
	图像质量调整部的矩阵 CRR 调整寄存器 1	IMGCNT_MTX_CRR_ADJ1	32	H'FFFF74B8	16、32
	SCL0 寄存器的更新控制寄存器	SCL0_UPDATE	32	H'FFFF7500	16、32
	屏蔽处理寄存器	SCL0_FRC1	32	H'FFFF7504	16、32
	欠缺补偿寄存器	SCL0_FRC2	32	H'FFFF7508	16、32
	输出同步选择寄存器	SCL0_FRC3	32	H'FFFF750C	16、32
	自激周期寄存器	SCL0_FRC4	32	H'FFFF7510	16、32
	输出延迟控制寄存器	SCL0_FRC5	32	H'FFFF7514	16、32
	全屏垂直尺寸寄存器	SCL0_FRC6	32	H'FFFF7518	16、32
	全屏水平尺寸寄存器	SCL0_FRC7	32	H'FFFF751C	16、32
	同步检测寄存器	SCL0_FRC9	32	H'FFFF7524	16、32
	缩小控制寄存器	SCL0_DS1	32	H'FFFF752C	16、32
	捕捉垂直尺寸寄存器	SCL0_DS2	32	H'FFFF7530	16、32
	捕捉水平尺寸寄存器	SCL0_DS3	32	H'FFFF7534	16、32
	水平缩小寄存器	SCL0_DS4	32	H'FFFF7538	16、32
	垂直初始相位寄存器	SCL0_DS5	32	H'FFFF753C	16、32
	垂直缩放寄存器	SCL0_DS6	32	H'FFFF7540	16、32
	缩小控制部输出尺寸寄存器	SCL0_DS7	32	H'FFFF7544	16、32
	放大控制寄存器	SCL0_US1	32	H'FFFF7548	16、32
	输出图像垂直尺寸寄存器	SCL0_US2	32	H'FFFF754C	16、32
	输出图像水平尺寸寄存器	SCL0_US3	32	H'FFFF7550	16、32
	放大控制部输入尺寸寄存器	SCL0_US4	32	H'FFFF7554	16、32
	水平放大寄存器	SCL0_US5	32	H'FFFF7558	16、32
	水平放大初始相位寄存器	SCL0_US6	32	H'FFFF755C	16、32
	切边寄存器	SCL0_US7	32	H'FFFF7560	16、32
	帧缓冲器的读选择寄存器	SCL0_US8	32	H'FFFF7564	16、32
	背景色寄存器	SCL0_OVR1	32	H'FFFF756C	16、32
	SCL1 寄存器的更新控制寄存器	SCL1_UPDATE	32	H'FFFF7580	16、32
	写操作模式寄存器	SCL1_WR1	32	H'FFFF7588	16、32
	写地址寄存器 1	SCL1_WR2	32	H'FFFF758C	16、32
	写地址寄存器 2	SCL1_WR3	32	H'FFFF7590	16、32
	写地址寄存器 3	SCL1_WR4	32	H'FFFF7594	16、32
帧减少寄存器	SCL1_WR5	32	H'FFFF759C	16、32	

模块名	寄存器名	略称	位数	地址	存取长度
视频显示 控制器 4	位压缩寄存器	SCL1_WR6	32	H'FFFF75A0	16、32
	写检测寄存器	SCL1_WR7	32	H'FFFF75A4	16、32
	图形 (1) 寄存器的更新控制寄存器	GR1_UPDATE	32	H'FFFF7600	16、32
	帧缓冲器的读控制寄存器 (图形 (1))	GR1_FLM_RD	32	H'FFFF7604	16、32
	帧缓冲控制寄存器 1 (图形 (1))	GR1_FLM1	32	H'FFFF7608	16、32
	帧缓冲控制寄存器 2 (图形 (1))	GR1_FLM2	32	H'FFFF760C	16、32
	帧缓冲控制寄存器 3 (图形 (1))	GR1_FLM3	32	H'FFFF7610	16、32
	帧缓冲控制寄存器 4 (图形 (1))	GR1_FLM4	32	H'FFFF7614	16、32
	帧缓冲控制寄存器 5 (图形 (1))	GR1_FLM5	32	H'FFFF7618	16、32
	帧缓冲控制寄存器 6 (图形 (1))	GR1_FLM6	32	H'FFFF761C	16、32
	α 混合控制寄存器 1 (图形 (1))	GR1_AB1	32	H'FFFF7620	16、32
	α 混合控制寄存器 2 (图形 (1))	GR1_AB2	32	H'FFFF7624	16、32
	α 混合控制寄存器 3 (图形 (1))	GR1_AB3	32	H'FFFF7628	16、32
	α 混合控制寄存器 7 (图形 (1))	GR1_AB7	32	H'FFFF7638	16、32
	α 混合控制寄存器 8 (图形 (1))	GR1_AB8	32	H'FFFF763C	16、32
	α 混合控制寄存器 9 (图形 (1))	GR1_AB9	32	H'FFFF7640	16、32
	α 混合控制寄存器 10 (图形 (1))	GR1_AB10	32	H'FFFF7644	16、32
	α 混合控制寄存器 11 (图形 (1))	GR1_AB11	32	H'FFFF7648	16、32
	背景色控制寄存器 (图形 (1))	GR1_BASE	32	H'FFFF764C	16、32
	CLUT 表控制寄存器 (图形 (1))	GR1_CLUT	32	H'FFFF7650	16、32
	图像质量改善部的寄存器更新控制寄存器	ADJ_UPDATE	32	H'FFFF7680	16、32
	黑色信号展宽部寄存器	ADJ_BKSTR_SET	32	H'FFFF7684	16、32
	增强器的时序调整寄存器 1	ADJ_ENH_TIM1	32	H'FFFF7688	16、32
	增强器的时序调整寄存器 2	ADJ_ENH_TIM2	32	H'FFFF768C	16、32
	增强器的时序调整寄存器 3	ADJ_ENH_TIM3	32	H'FFFF7690	16、32
	增强器的清晰度寄存器 1	ADJ_ENH_SHP1	32	H'FFFF7694	16、32
	增强器的清晰度寄存器 2	ADJ_ENH_SHP2	32	H'FFFF7698	16、32
	增强器的清晰度寄存器 3	ADJ_ENH_SHP3	32	H'FFFF769C	16、32
	增强器的清晰度寄存器 4	ADJ_ENH_SHP4	32	H'FFFF76A0	16、32
	增强器的清晰度寄存器 5	ADJ_ENH_SHP5	32	H'FFFF76A4	16、32
	增强器的清晰度寄存器 6	ADJ_ENH_SHP6	32	H'FFFF76A8	16、32
	增强器的 LTI 寄存器 1	ADJ_ENH_LTI1	32	H'FFFF76AC	16、32
	增强器的 LTI 寄存器 2	ADJ_ENH_LTI2	32	H'FFFF76B0	16、32
	图像质量改善部的矩阵模式寄存器	ADJ_MTX_MODE	32	H'FFFF76B4	16、32
	图像质量改善部的矩阵 YG 调整寄存器 0	ADJ_MTX_YG_ADJ0	32	H'FFFF76B8	16、32
	图像质量改善部的矩阵 YG 调整寄存器 1	ADJ_MTX_YG_ADJ1	32	H'FFFF76BC	16、32
	图像质量改善部的矩阵 CBB 调整寄存器 0	ADJ_MTX_CBB_ADJ0	32	H'FFFF76C0	16、32
	图像质量改善部的矩阵 CBB 调整寄存器 1	ADJ_MTX_CBB_ADJ1	32	H'FFFF76C4	16、32
	图像质量改善部的矩阵 CRR 调整寄存器 0	ADJ_MTX_CRR_ADJ0	32	H'FFFF76C8	16、32
	图像质量改善部的矩阵 CRR 调整寄存器 1	ADJ_MTX_CRR_ADJ1	32	H'FFFF76CC	16、32
图形 (2) 的寄存器更新控制寄存器	GR2_UPDATE	32	H'FFFF7700	16、32	
帧缓冲器的读控制寄存器 (图形 (2))	GR2_FLM_RD	32	H'FFFF7704	16、32	

模块名	寄存器名	略称	位数	地址	存取长度
视频显示 控制器 4	帧缓冲器的控制寄存器 1 (图形 (2))	GR2_FLM1	32	H'FFFF7708	16、32
	帧缓冲器的控制寄存器 2 (图形 (2))	GR2_FLM2	32	H'FFFF770C	16、32
	帧缓冲器的控制寄存器 3 (图形 (2))	GR2_FLM3	32	H'FFFF7710	16、32
	帧缓冲器的控制寄存器 4 (图形 (2))	GR2_FLM4	32	H'FFFF7714	16、32
	帧缓冲器的控制寄存器 5 (图形 (2))	GR2_FLM5	32	H'FFFF7718	16、32
	帧缓冲器的控制寄存器 6 (图形 (2))	GR2_FLM6	32	H'FFFF771C	16、32
	α 混合控制寄存器 1 (图形 (2))	GR2_AB1	32	H'FFFF7720	16、32
	α 混合控制寄存器 2 (图形 (2))	GR2_AB2	32	H'FFFF7724	16、32
	α 混合控制寄存器 3 (图形 (2))	GR2_AB3	32	H'FFFF7728	16、32
	α 混合控制寄存器 4 (图形 (2))	GR2_AB4	32	H'FFFF772C	16、32
	α 混合控制寄存器 5 (图形 (2))	GR2_AB5	32	H'FFFF7730	16、32
	α 混合控制寄存器 6 (图形 (2))	GR2_AB6	32	H'FFFF7734	16、32
	α 混合控制寄存器 7 (图形 (2))	GR2_AB7	32	H'FFFF7738	16、32
	α 混合控制寄存器 8 (图形 (2))	GR2_AB8	32	H'FFFF773C	16、32
	α 混合控制寄存器 9 (图形 (2))	GR2_AB9	32	H'FFFF7740	16、32
	α 混合控制寄存器 10 (图形 (2))	GR2_AB10	32	H'FFFF7744	16、32
	α 混合控制寄存器 11 (图形 (2))	GR2_AB11	32	H'FFFF7748	16、32
	背景色控制寄存器 (图形 (2))	GR2_BASE	32	H'FFFF774C	16、32
	CLUT 表控制寄存器 (图形 (2))	GR2_CLUT	32	H'FFFF7750	16、32
	状态监视寄存器 (图形 (2))	GR2_MON	32	H'FFFF7754	16、32
	图形 (3) 的寄存器更新控制寄存器	GR3_UPDATE	32	H'FFFF7780	16、32
	帧缓冲控制寄存器 (图形 (3))	GR3_FLM_RD	32	H'FFFF7784	16、32
	帧缓冲控制寄存器 1 (图形 (3))	GR3_FLM1	32	H'FFFF7788	16、32
	帧缓冲控制寄存器 2 (图形 (3))	GR3_FLM2	32	H'FFFF778C	16、32
	帧缓冲控制寄存器 3 (图形 (3))	GR3_FLM3	32	H'FFFF7790	16、32
	帧缓冲控制寄存器 4 (图形 (3))	GR3_FLM4	32	H'FFFF7794	16、32
	帧缓冲控制寄存器 5 (图形 (3))	GR3_FLM5	32	H'FFFF7798	16、32
	帧缓冲控制寄存器 6 (图形 (3))	GR3_FLM6	32	H'FFFF779C	16、32
	α 混合控制寄存器 1 (图形 (3))	GR3_AB1	32	H'FFFF77A0	16、32
	α 混合控制寄存器 2 (图形 (3))	GR3_AB2	32	H'FFFF77A4	16、32
	α 混合控制寄存器 3 (图形 (3))	GR3_AB3	32	H'FFFF77A8	16、32
	α 混合控制寄存器 4 (图形 (3))	GR3_AB4	32	H'FFFF77AC	16、32
	α 混合控制寄存器 5 (图形 (3))	GR3_AB5	32	H'FFFF77B0	16、32
	α 混合控制寄存器 6 (图形 (3))	GR3_AB6	32	H'FFFF77B4	16、32
	α 混合控制寄存器 7 (图形 (3))	GR3_AB7	32	H'FFFF77B8	16、32
	α 混合控制寄存器 8 (图形 (3))	GR3_AB8	32	H'FFFF77BC	16、32
	α 混合控制寄存器 9 (图形 (3))	GR3_AB9	32	H'FFFF77C0	16、32
	α 混合控制寄存器 10 (图形 (3))	GR3_AB10	32	H'FFFF77C4	16、32
	α 混合控制寄存器 11 (图形 (3))	GR3_AB11	32	H'FFFF77C8	16、32
	背景色控制寄存器 (图形 (3))	GR3_BASE	32	H'FFFF77CC	16、32
	CLUT 表中断控制寄存器 (图形 (3))	GR3_CLUT_INT	32	H'FFFF77D0	16、32
	状态监视寄存器 (图形 (3))	GR3_MON	32	H'FFFF77D4	16、32

模块名	寄存器名	略称	位数	地址	存取长度
视频显示 控制器 4	灰度校正部的寄存器更新控制寄存器 G	GAM_G_UPDATE	32	H'FFFF7800	16、32
	灰度校正部的功能切换寄存器	GAM_SW	32	H'FFFF7804	16、32
	灰度校正部的表设定寄存器 G1	GAM_G_LUT1	32	H'FFFF7808	16、32
	灰度校正部的表设定寄存器 G2	GAM_G_LUT2	32	H'FFFF780C	16、32
	灰度校正部的表设定寄存器 G3	GAM_G_LUT3	32	H'FFFF7810	16、32
	灰度校正部的表设定寄存器 G4	GAM_G_LUT4	32	H'FFFF7814	16、32
	灰度校正部的表设定寄存器 G5	GAM_G_LUT5	32	H'FFFF7818	16、32
	灰度校正部的表设定寄存器 G6	GAM_G_LUT6	32	H'FFFF781C	16、32
	灰度校正部的表设定寄存器 G7	GAM_G_LUT7	32	H'FFFF7820	16、32
	灰度校正部的表设定寄存器 G8	GAM_G_LUT8	32	H'FFFF7824	16、32
	灰度校正部的表设定寄存器 G9	GAM_G_LUT9	32	H'FFFF7828	16、32
	灰度校正部的表设定寄存器 G10	GAM_G_LUT10	32	H'FFFF782C	16、32
	灰度校正部的表设定寄存器 G11	GAM_G_LUT11	32	H'FFFF7830	16、32
	灰度校正部的表设定寄存器 G12	GAM_G_LUT12	32	H'FFFF7834	16、32
	灰度校正部的表设定寄存器 G13	GAM_G_LUT13	32	H'FFFF7838	16、32
	灰度校正部的表设定寄存器 G14	GAM_G_LUT14	32	H'FFFF783C	16、32
	灰度校正部的表设定寄存器 G15	GAM_G_LUT15	32	H'FFFF7840	16、32
	灰度校正部的表设定寄存器 G16	GAM_G_LUT16	32	H'FFFF7844	16、32
	灰度校正部的区域设定寄存器 G1	GAM_G_AREA1	32	H'FFFF7848	16、32
	灰度校正部的区域设定寄存器 G2	GAM_G_AREA2	32	H'FFFF784C	16、32
	灰度校正部的区域设定寄存器 G3	GAM_G_AREA3	32	H'FFFF7850	16、32
	灰度校正部的区域设定寄存器 G4	GAM_G_AREA4	32	H'FFFF7854	16、32
	灰度校正部的区域设定寄存器 G5	GAM_G_AREA5	32	H'FFFF7858	16、32
	灰度校正部的区域设定寄存器 G6	GAM_G_AREA6	32	H'FFFF785C	16、32
	灰度校正部的区域设定寄存器 G7	GAM_G_AREA7	32	H'FFFF7860	16、32
	灰度校正部的区域设定寄存器 G8	GAM_G_AREA8	32	H'FFFF7864	16、32
	灰度校正部的寄存器更新控制寄存器 B	GAM_B_UPDATE	32	H'FFFF7880	16、32
	灰度校正部的表设定寄存器 B1	GAM_B_LUT1	32	H'FFFF7888	16、32
	灰度校正部的表设定寄存器 B2	GAM_B_LUT2	32	H'FFFF788C	16、32
	灰度校正部的表设定寄存器 B3	GAM_B_LUT3	32	H'FFFF7890	16、32
	灰度校正部的表设定寄存器 B4	GAM_B_LUT4	32	H'FFFF7894	16、32
	灰度校正部的表设定寄存器 B5	GAM_B_LUT5	32	H'FFFF7898	16、32
	灰度校正部的表设定寄存器 B6	GAM_B_LUT6	32	H'FFFF789C	16、32
	灰度校正部的表设定寄存器 B7	GAM_B_LUT7	32	H'FFFF78A0	16、32
	灰度校正部的表设定寄存器 B8	GAM_B_LUT8	32	H'FFFF78A4	16、32
	灰度校正部的表设定寄存器 B9	GAM_B_LUT9	32	H'FFFF78A8	16、32
	灰度校正部的表设定寄存器 B10	GAM_B_LUT10	32	H'FFFF78AC	16、32
	灰度校正部的表设定寄存器 B11	GAM_B_LUT11	32	H'FFFF78B0	16、32
	灰度校正部的表设定寄存器 B12	GAM_B_LUT12	32	H'FFFF78B4	16、32
	灰度校正部的表设定寄存器 B13	GAM_B_LUT13	32	H'FFFF78B8	16、32
灰度校正部的表设定寄存器 B14	GAM_B_LUT14	32	H'FFFF78BC	16、32	
灰度校正部的表设定寄存器 B15	GAM_B_LUT15	32	H'FFFF78C0	16、32	

模块名	寄存器名	略称	位数	地址	存取长度
视频显示 控制器 4	灰度校正部的表设定寄存器 B16	GAM_B_LUT16	32	H'FFFF78C4	16、32
	灰度校正部的区域设定寄存器 B1	GAM_B_AREA1	32	H'FFFF78C8	16、32
	灰度校正部的区域设定寄存器 B2	GAM_B_AREA2	32	H'FFFF78CC	16、32
	灰度校正部的区域设定寄存器 B3	GAM_B_AREA3	32	H'FFFF78D0	16、32
	灰度校正部的区域设定寄存器 B4	GAM_B_AREA4	32	H'FFFF78D4	16、32
	灰度校正部的区域设定寄存器 B5	GAM_B_AREA5	32	H'FFFF78D8	16、32
	灰度校正部的区域设定寄存器 B6	GAM_B_AREA6	32	H'FFFF78DC	16、32
	灰度校正部的区域设定寄存器 B7	GAM_B_AREA7	32	H'FFFF78E0	16、32
	灰度校正部的区域设定寄存器 B8	GAM_B_AREA8	32	H'FFFF78E4	16、32
	灰度校正部的寄存器更新控制寄存器 R	GAM_R_UPDATE	32	H'FFFF7900	16、32
	灰度校正部的表设定寄存器 R1	GAM_R_LUT1	32	H'FFFF7908	16、32
	灰度校正部的表设定寄存器 R2	GAM_R_LUT2	32	H'FFFF790C	16、32
	灰度校正部的表设定寄存器 R3	GAM_R_LUT3	32	H'FFFF7910	16、32
	灰度校正部的表设定寄存器 R4	GAM_R_LUT4	32	H'FFFF7914	16、32
	灰度校正部的表设定寄存器 R5	GAM_R_LUT5	32	H'FFFF7918	16、32
	灰度校正部的表设定寄存器 R6	GAM_R_LUT6	32	H'FFFF791C	16、32
	灰度校正部的表设定寄存器 R7	GAM_R_LUT7	32	H'FFFF7920	16、32
	灰度校正部的表设定寄存器 R8	GAM_R_LUT8	32	H'FFFF7924	16、32
	灰度校正部的表设定寄存器 R9	GAM_R_LUT9	32	H'FFFF7928	16、32
	灰度校正部的表设定寄存器 R10	GAM_R_LUT10	32	H'FFFF792C	16、32
	灰度校正部的表设定寄存器 R11	GAM_R_LUT11	32	H'FFFF7930	16、32
	灰度校正部的表设定寄存器 R12	GAM_R_LUT12	32	H'FFFF7934	16、32
	灰度校正部的表设定寄存器 R13	GAM_R_LUT13	32	H'FFFF7938	16、32
	灰度校正部的表设定寄存器 R14	GAM_R_LUT14	32	H'FFFF793C	16、32
	灰度校正部的表设定寄存器 R15	GAM_R_LUT15	32	H'FFFF7940	16、32
	灰度校正部的表设定寄存器 R16	GAM_R_LUT16	32	H'FFFF7944	16、32
	灰度校正部的区域设定寄存器 R1	GAM_R_AREA1	32	H'FFFF7948	16、32
	灰度校正部的区域设定寄存器 R2	GAM_R_AREA2	32	H'FFFF794C	16、32
	灰度校正部的区域设定寄存器 R3	GAM_R_AREA3	32	H'FFFF7950	16、32
	灰度校正部的区域设定寄存器 R4	GAM_R_AREA4	32	H'FFFF7954	16、32
	灰度校正部的区域设定寄存器 R5	GAM_R_AREA5	32	H'FFFF7958	16、32
	灰度校正部的区域设定寄存器 R6	GAM_R_AREA6	32	H'FFFF795C	16、32
	灰度校正部的区域设定寄存器 R7	GAM_R_AREA7	32	H'FFFF7960	16、32
	灰度校正部的区域设定寄存器 R8	GAM_R_AREA8	32	H'FFFF7964	16、32
	TCON 寄存器的更新控制寄存器	TCON_UPDATE	32	H'FFFF7980	16、32
	TCON 基准时序设定寄存器	TCON_TIM	32	H'FFFF7984	16、32
	TCON 垂直时序设定寄存器 A1	TCON_TIM_STVA1	32	H'FFFF7988	16、32
	TCON 垂直时序设定寄存器 A2	TCON_TIM_STVA2	32	H'FFFF798C	16、32
	TCON 垂直时序设定寄存器 B1	TCON_TIM_STVB1	32	H'FFFF7990	16、32
	TCON 垂直时序设定寄存器 B2	TCON_TIM_STVB2	32	H'FFFF7994	16、32
TCON 水平时序设定寄存器 STH1	TCON_TIM_STH1	32	H'FFFF7998	16、32	
TCON 水平时序设定寄存器 STH2	TCON_TIM_STH2	32	H'FFFF799C	16、32	

模块名	寄存器名	略称	位数	地址	存取长度
视频显示 控制器 4	TCON 水平时序设定寄存器 STB1	TCON_TIM_STB1	32	H'FFFF79A0	16、32
	TCON 水平时序设定寄存器 STB2	TCON_TIM_STB2	32	H'FFFF79A4	16、32
	TCON 水平时序设定寄存器 CPV1	TCON_TIM_CPV1	32	H'FFFF79A8	16、32
	TCON 水平时序设定寄存器 CPV2	TCON_TIM_CPV2	32	H'FFFF79AC	16、32
	TCON 水平时序设定寄存器 POLA1	TCON_TIM_POLA1	32	H'FFFF79B0	16、32
	TCON 水平时序设定寄存器 POLA2	TCON_TIM_POLA2	32	H'FFFF79B4	16、32
	TCON 水平时序设定寄存器 POLB1	TCON_TIM_POLB1	32	H'FFFF79B8	16、32
	TCON 水平时序设定寄存器 POLB2	TCON_TIM_POLB2	32	H'FFFF79BC	16、32
	TCON 数据允许极性设定寄存器	TCON_TIM_DE	32	H'FFFF79C0	16、32
	输出控制部的寄存器更新控制寄存器	OUT_UPDATE	32	H'FFFF7A00	16、32
	输出接口寄存器	OUT_SET	32	H'FFFF7A04	16、32
	亮度 (DC) 校正寄存器 1	OUT_BRIGHT1	32	H'FFFF7A08	16、32
	亮度 (DC) 校正寄存器 2	OUT_BRIGHT2	32	H'FFFF7A0C	16、32
	对比度 (增益) 校正寄存器	OUT_CONTRAST	32	H'FFFF7A10	16、32
	显示屏抖动寄存器	OUT_PDTHA	32	H'FFFF7A14	16、32
	输出相位控制寄存器	OUT_CLK_PHASE	32	H'FFFF7A24	16、32
	中断控制寄存器 1	SYSCNT_INT1	32	H'FFFF7A80	16、32
	中断控制寄存器 2	SYSCNT_INT2	32	H'FFFF7A84	16、32
	中断控制寄存器 3	SYSCNT_INT3	32	H'FFFF7A88	16、32
	中断控制寄存器 4	SYSCNT_INT4	32	H'FFFF7A8C	16、32
	显示屏时钟控制寄存器	SYSCNT_PANEL_CLK	16	H'FFFF7A90	16
	CLUT 表的读选择信号状态标志寄存器	SYSCNT_CLUT	16	H'FFFF7A92	16
	失真校正引擎	控制寄存器	CR	32	H'FFFF3008
状态寄存器		SR	32	H'FFFF300C	32
状态清除寄存器		SRCR	32	H'FFFF3010	32
中断控制寄存器		ICR	32	H'FFFF3014	32
中断屏蔽寄存器		IMR	32	H'FFFF3018	32
DL 状态寄存器		DLPR	32	H'FFFF3020	32
DL 起始地址寄存器		DLSAR	32	H'FFFF3030	32
目标起始地址寄存器		DSAR	32	H'FFFF3034	32
目标跨距寄存器		DSTR	32	H'FFFF303C	32
目标起始地址寄存器 2		DSAR2	32	H'FFFF3048	32
三角形模式寄存器		TRIMR	32	H'FFFF3060	32
三角形置位寄存器		TRIMSR	32	H'FFFF3064	32
三角形清除寄存器		TRIMCR	32	H'FFFF3068	32
三角形颜色寄存器		TRICR	32	H'FFFF306C	32
源 / 目标坐标小数点寄存器		UVDPOR	32	H'FFFF3070	32
源宽度 W 寄存器		SUSR	32	H'FFFF3074	32
源宽度 H 寄存器		SVSR	32	H'FFFF3078	32
MIN 限幅 X 寄存器		XMINR	32	H'FFFF3080	32
MIN 限幅 Y 寄存器		YMINR	32	H'FFFF3084	32
MAX 限幅 X 寄存器		XMAXR	32	H'FFFF3088	32

模块名	寄存器名	略称	位数	地址	存取长度
失真校正引擎	MAX 限幅 Y 寄存器	YMAXR	32	H'FFFF308C	32
	网格生成 X 大小寄存器	AMXSR	32	H'FFFF3090	32
	网格生成 Y 大小寄存器	AMYSR	32	H'FFFF3094	32
	网格生成 X 启动寄存器	AMXOR	32	H'FFFF3098	32
	网格生成 Y 启动寄存器	AMYOR	32	H'FFFF309C	32
	存储器存取控制寄存器 1	MACR1	32	H'FFFF30A0	32
	开始行指定寄存器	LSPR	32	H'FFFF3A00	32
	结束行指定寄存器	LEPR	32	H'FFFF3A04	32
	网格大小寄存器	LMSR	32	H'FFFF3A08	32
显示输出比较单元	控制寄存器	DOCMCR	32	H'FFFA800	32
	状态寄存器	DOCMSTR	32	H'FFFA804	32
	状态清除寄存器	DOCMCLSTR	32	H'FFFA808	32
	中断允许寄存器	DOCMENR	32	H'FFFA80C	32
	工作参数设定寄存器	DOCMPMR	32	H'FFFA814	32
	CRC 码期待值寄存器	DOCMECRCR	32	H'FFFA818	32
	CRC 码计算值寄存器	DOCMCCRCR	32	H'FFFA81C	32
	水平方向起始位置设定寄存器	DOCMSPXR	32	H'FFFA820	32
	垂直方向起始位置设定寄存器	DOCMSPYR	32	H'FFFA824	32
	水平方向尺寸寄存器	DOCMSZXR	32	H'FFFA828	32
	垂直方向尺寸寄存器	DOCMSZYR	32	H'FFFA82C	32
	CRC 码初始值寄存器	DOCMCRCIR	32	H'FFFA830	32
JPEG 编解码器单元	JPEG 码模式寄存器	JCMOD	8	H'E8017000	8
	JPEG 码命令寄存器	JCCMD	8	H'E8017001	8
	JPEG 码量化表号寄存器	JCQTN	8	H'E8017003	8
	JPEG 码霍夫曼表号寄存器	JCHTN	8	H'E8017004	8
	JPEG 码 DRI 高位寄存器	JCDRIU	8	H'E8017005	8
	JPEG 码 DRI 低位寄存器	JCDRID	8	H'E8017006	8
	JPEG 码垂直方向尺寸高位寄存器	JCVSZU	8	H'E8017007	8
	JPEG 码垂直方向尺寸低位寄存器	JCVSZD	8	H'E8017008	8
	JPEG 码水平方向尺寸高位寄存器	JCHSZU	8	H'E8017009	8
	JPEG 码水平方向尺寸低位寄存器	JCHSZD	8	H'E801700A	8
	JPEG 码数据计数高位寄存器	JCDTCU	8	H'E801700B	8
	JPEG 码数据计数中位寄存器	JCDTCM	8	H'E801700C	8
	JPEG 码数据计数低位寄存器	JCDTCD	8	H'E801700D	8
	JPEG 中断允许寄存器 0	JINTE0	8	H'E801700E	8
	JPEG 中断状态寄存器 0	JINTS0	8	H'E801700F	8
	JPEG 码解码错误寄存器	JCDERR	8	H'E8017010	8
	JPEG 码重新启动寄存器	JCRST	8	H'E8017011	8
	JPEG 接口压缩控制寄存器	JIFECNT	32	H'E8017040	32
	JPEG 接口压缩源地址寄存器	JIFESA	32	H'E8017044	32
	JPEG 接口压缩行偏移寄存器	JIFESOFST	32	H'E8017048	32
JPEG 接口压缩目标地址寄存器	JIFEDA	32	H'E801704C	32	

模块名	寄存器名	略称	位数	地址	存取长度
JPEG 编解码器 单元	JPEG 接口压缩源行计数寄存器	JIFESLC	32	H'E8017050	32
	JPEG 接口压缩目标寄存器	JIFEDDC	32	H'E8017054	32
	JPEG 接口解压控制寄存器	JIFDCNT	32	H'E8017058	32
	JPEG 接口解压源地址寄存器	JIFDSA	32	H'E801705C	32
	JPEG 接口解压目标偏移寄存器	JIFDDOFST	32	H'E8017060	32
	JPEG 接口解压目标地址寄存器	JIFDDA	32	H'E8017064	32
	JPEG 接口解压源计数寄存器	JIFSDC	32	H'E8017068	32
	JPEG 接口解压目标行计数寄存器	JIFDDL	32	H'E801706C	32
	JPEG 接口解压 α 设定寄存器	JIFDADT	32	H'E8017070	32
	JPEG 中断允许寄存器 1	JINTE1	32	H'E801708C	32
	JPEG 中断状态寄存器 1	JINTS1	32	H'E8017090	32
	JPEG 码量化表 0 寄存器	JCQTBL0	512	H'E8017100 ~ H'E801713F	8
	JPEG 码量化表 1 寄存器	JCQTBL1	512	H'E8017140 ~ H'E801717F	8
	JPEG 码量化表 2 寄存器	JCQTBL2	512	H'E8017180 ~ H'E80171BF	8
	JPEG 码量化表 3 寄存器	JCQTBL3	512	H'E80171C0 ~ H'E80171FF	8
	JPEG 码霍夫曼表 DC0 寄存器	JCHTBD0	224	H'E8017200 ~ H'E801721B	8
	JPEG 码霍夫曼表 AC0 寄存器	JCHTBA0	1416	H'E8017220 ~ H'E80172D1	8
	JPEG 码霍夫曼表 DC1 寄存器	JCHTBD1	224	H'E8017300 ~ H'E801731B	8
	JPEG 码霍夫曼表 AC1 寄存器	JCHTBA1	1416	H'E8017320 ~ H'E80173D1	8
采样率转换器	输入数据寄存器 0	SRCID_0	32	H'FFFE7000	16、32
	输出数据寄存器 0	SRCOD_0	32	H'FFFE7004	16、32
	输入数据控制寄存器 0	SRCIDCTRL_0	16	H'FFFE7008	16
	输出数据控制寄存器 0	SRCODCTRL_0	16	H'FFFE700A	16
	控制寄存器 0	SRCCTRL_0	16	H'FFFE700C	16
	状态寄存器 0	SRCSTAT_0	16	H'FFFE700E	16
	输入数据寄存器 1	SRCID_1	32	H'FFFE7800	16、32
	输出数据寄存器 1	SRCOD_1	32	H'FFFE7804	16、32
	输入数据控制寄存器 1	SRCIDCTRL_1	16	H'FFFE7808	16
	输出数据控制寄存器 1	SRCODCTRL_1	16	H'FFFE780A	16
	控制寄存器 1	SRCCTRL_1	16	H'FFFE780C	16
	状态寄存器 1	SRCSTAT_1	16	H'FFFE780E	16
	输入数据寄存器 2	SRCID_2	32	H'FFFE8000	16、32
	输出数据寄存器 2	SRCOD_2	32	H'FFFE8004	16、32
	输入数据控制寄存器 2	SRCIDCTRL_2	16	H'FFFE8008	16
	输出数据控制寄存器 2	SRCODCTRL_2	16	H'FFFE800A	16
	控制寄存器 2	SRCCTRL_2	16	H'FFFE800C	16

模块名	寄存器名	略称	位数	地址	存取长度
采样率转换器	状态寄存器 2	SRCSTAT_2	16	H'FFFE800E	16
音频发生器	音频发生器的控制寄存器 1_0	SGCR1_0	8	H'FFFE800	8、16
	音频发生器的控制状态寄存器 _0	SGCSR_0	8	H'FFFE801	8、16
	音频发生器的控制寄存器 2_0	SGCR2_0	8	H'FFFE802	8、16
	音频发生器的音量寄存器 _0	SGLR_0	8	H'FFFE803	8、16
	音频发生器的音频寄存器 _0	SGTFR_0	8	H'FFFE804	8、16
	音频发生器的基准频率寄存器 _0	SGSFR_0	8	H'FFFE805	8、16
	音频发生器的控制寄存器 1_1	SGCR1_1	8	H'FFFECA00	8、16
	音频发生器的控制状态寄存器 _1	SGCSR_1	8	H'FFFECA01	8、16
	音频发生器的控制寄存器 2_1	SGCR2_1	8	H'FFFECA02	8、16
	音频发生器的音量寄存器 _1	SGLR_1	8	H'FFFECA03	8、16
	音频发生器的音频寄存器 _1	SGTFR_1	8	H'FFFECA04	8、16
	音频发生器的基准频率寄存器 _1	SGSFR_1	8	H'FFFECA05	8、16
	音频发生器的控制寄存器 1_2	SGCR1_2	8	H'FFFECC00	8、16
	音频发生器的控制状态寄存器 _2	SGCSR_2	8	H'FFFECC01	8、16
	音频发生器的控制寄存器 2_2	SGCR2_2	8	H'FFFECC02	8、16
	音频发生器的音量寄存器 _2	SGLR_2	8	H'FFFECC03	8、16
	音频发生器的音频寄存器 _2	SGTFR_2	8	H'FFFECC04	8、16
	音频发生器的基准频率寄存器 _2	SGSFR_2	8	H'FFFECC05	8、16
	音频发生器的控制寄存器 1_3	SGCR1_3	8	H'FFFECE00	8、16
	音频发生器的控制状态寄存器 _3	SGCSR_3	8	H'FFFECE01	8、16
	音频发生器的控制寄存器 2_3	SGCR2_3	8	H'FFFECE02	8、16
	音频发生器的音量寄存器 _3	SGLR_3	8	H'FFFECE03	8、16
	音频发生器的音频寄存器 _3	SGTFR_3	8	H'FFFECE04	8、16
	音频发生器的基准频率寄存器 _3	SGSFR_3	8	H'FFFECE05	8、16
MMC 主机接口	命令设定寄存器	CE_CMD_SET	32	H'E8030800	16、32
	自变量寄存器	CE_ARG	32	H'E8030808	16、32
	自动 CMD12 的自变量寄存器	CE_ARG_CMD12	32	H'E803080C	16、32
	命令控制寄存器	CE_CMD_CTRL	32	H'E8030810	16、32
	传送块设定寄存器	CE_BLOCK_SET	32	H'E8030814	16、32
	时钟控制寄存器	CE_CLK_CTRL	32	H'E8030818	16、32
	缓冲器存取设定寄存器	CE_BUF_ACC	32	H'E803081C	16、32
	响应寄存器 3	CE_RESP3	32	H'E8030820	16、32
	响应寄存器 2	CE_RESP2	32	H'E8030824	16、32
	响应寄存器 1	CE_RESP1	32	H'E8030828	16、32
	响应寄存器 0	CE_RESP0	32	H'E803082C	16、32
	自动 CMD12 的响应寄存器	CE_RESP_CMD12	32	H'E8030830	16、32
	数据寄存器	CE_DATA	32	H'E8030834	16、32
	中断标志寄存器	CE_INT	32	H'E8030840	16、32
	中断允许寄存器	CE_INT_EN	32	H'E8030844	16、32
	状态寄存器 1	CE_HOST_STS1	32	H'E8030848	16、32
状态寄存器 2	CE_HOST_STS2	32	H'E803084C	16、32	

模块名	寄存器名	略称	位数	地址	存取长度
MMC 主机接口	DMA 模式设定寄存器	CE_DMA_MODE	32	H'E803085C	16、32
	卡检测 / 端口控制寄存器	CE_DETECT	32	H'E8030870	16、32
	特殊模式设定寄存器	CE_ADD_MODE	32	H'E8030874	16、32
	版本寄存器	CE_VERSION	32	H'E803087C	16、32
马达控制 PWM 定时器	PWM 控制寄存器 _1	PWCR_1	8	H'FFFEF4E0	8、16
	PWM 极性寄存器 _1	PWPR_1	8	H'FFFEF4E4	8、16
	PWM 周期寄存器 _1	PWCYR_1	16	H'FFFEF4E6	16
	PWM 缓冲寄存器 _1A	PWBFR_1A	16	H'FFFEF4E8	16
	PWM 缓冲寄存器 _1C	PWBFR_1C	16	H'FFFEF4EA	16
	PWM 缓冲寄存器 _1E	PWBFR_1E	16	H'FFFEF4EC	16
	PWM 缓冲寄存器 _1G	PWBFR_1G	16	H'FFFEF4EE	16
	PWM 控制寄存器 _2	PWCR_2	8	H'FFFEF4F0	8、16
	PWM 极性寄存器 _2	PWPR_2	8	H'FFFEF4F4	8、16
	PWM 周期寄存器 _2	PWCYR_2	16	H'FFFEF4F6	16
	PWM 缓冲寄存器 _2A	PWBFR_2A	16	H'FFFEF4F8	16
	PWM 缓冲寄存器 _2C	PWBFR_2C	16	H'FFFEF4FA	16
	PWM 缓冲寄存器 _2E	PWBFR_2E	16	H'FFFEF4FC	16
	PWM 缓冲寄存器 _2G	PWBFR_2G	16	H'FFFEF4FE	16
	PWM 缓冲传送控制寄存器	PWBTCR	8	H'FFFEF406	8、16
通用输入 / 输出 端口	端口 A 的 IO 寄存器 0	PAIOR0	16	H'FFFE3812	8、16*
	端口 A 的数据寄存器 0	PADR0	16	H'FFFE3816	8、16*
	端口 A 的端口寄存器 0	PAPR0	16	H'FFFE381A	8、16
	端口 B 的控制寄存器 5	PBCR5	16	H'FFFE3824	8、16、32
	端口 B 的控制寄存器 4	PBCR4	16	H'FFFE3826	8、16
	端口 B 的控制寄存器 3	PBCR3	16	H'FFFE3828	8、16、32
	端口 B 的控制寄存器 2	PBCR2	16	H'FFFE382A	8、16
	端口 B 的控制寄存器 1	PBCR1	16	H'FFFE382C	8、16、32
	端口 B 的控制寄存器 0	PBCR0	16	H'FFFE382E	8、16
	端口 B 的 IO 寄存器 1	PBIOR1	16	H'FFFE3830	8、16、32
	端口 B 的 IO 寄存器 0	PBIOR0	16	H'FFFE3832	8、16
	端口 B 的数据寄存器 1	PBDR1	16	H'FFFE3834	8、16、32
	端口 B 的数据寄存器 0	PBDR0	16	H'FFFE3836	8、16
	端口 B 的端口寄存器 1	PBPR1	16	H'FFFE3838	8、16、32
	端口 B 的端口寄存器 0	PBPR0	16	H'FFFE383A	8、16
	端口 C 的控制寄存器 2	PCCR2	16	H'FFFE384A	8、16
	端口 C 的控制寄存器 1	PCCR1	16	H'FFFE384C	8、16、32
	端口 C 的控制寄存器 0	PCCR0	16	H'FFFE384E	8、16
	端口 C 的 IO 寄存器 0	PCIOR0	16	H'FFFE3852	8、16
	端口 C 的数据寄存器 0	PCDR0	16	H'FFFE3856	8、16
	端口 C 的端口寄存器 0	PCPR0	16	H'FFFE385A	8、16
	端口 D 的控制寄存器 3	PDCR3	16	H'FFFE3868	8、16、32
	端口 D 的控制寄存器 2	PDCR2	16	H'FFFE386A	8、16

模块名	寄存器名	略称	位数	地址	存取长度
通用输入 / 输出 端口	端口 D 的控制寄存器 1	PDCR1	16	H'FFFE386C	8、16、32
	端口 D 的控制寄存器 0	PDCR0	16	H'FFFE386E	8、16
	端口 D 的 IO 寄存器 0	PDIOR0	16	H'FFFE3872	8、16
	端口 D 的数据寄存器 0	PDDR0	16	H'FFFE3876	8、16
	端口 D 的端口寄存器 0	PDPR0	16	H'FFFE387A	8、16
	端口 E 的控制寄存器 1	PECR1	16	H'FFFE388C	8、16、32
	端口 E 的控制寄存器 0	PECR0	16	H'FFFE388E	8、16
	端口 E 的 IO 寄存器 0	PEIOR0	16	H'FFFE3892	8、16
	端口 E 的数据寄存器 0	PEDR0	16	H'FFFE3896	8、16
	端口 E 的端口寄存器 0	PEPR0	16	H'FFFE389A	8、16
	端口 F 的控制寄存器 6	PFGR6	16	H'FFFE38A2	8、16
	端口 F 的控制寄存器 5	PFGR5	16	H'FFFE38A4	8、16、32
	端口 F 的控制寄存器 4	PFGR4	16	H'FFFE38A6	8、16*
	端口 F 的控制寄存器 3	PFGR3	16	H'FFFE38A8	8、16、32
	端口 F 的控制寄存器 2	PFGR2	16	H'FFFE38AA	8、16
	端口 F 的控制寄存器 1	PFGR1	16	H'FFFE38AC	8、16、32
	端口 F 的控制寄存器 0	PFGR0	16	H'FFFE38AE	8、16
	端口 F 的 IO 寄存器 1	PFIOR1	16	H'FFFE38B0	8、16、32
	端口 F 的 IO 寄存器 0	PFIOR0	16	H'FFFE38B2	8、16
	端口 F 的数据寄存器 1	PFDR1	16	H'FFFE38B4	8、16、32
	端口 F 的数据寄存器 0	PFDR0	16	H'FFFE38B6	8、16
	端口 F 的端口寄存器 1	PFPR1	16	H'FFFE38B8	8、16、32
	端口 F 的端口寄存器 0	PFPR0	16	H'FFFE38BA	8、16
	端口 G 的控制寄存器 6	PGCR6	16	H'FFFE38C2	8、16
	端口 G 的控制寄存器 5	PGCR5	16	H'FFFE38C4	8、16、32
	端口 G 的控制寄存器 4	PGCR4	16	H'FFFE38C6	8、16
	端口 G 的控制寄存器 3	PGCR3	16	H'FFFE38C8	8、16、32
	端口 G 的控制寄存器 2	PGCR2	16	H'FFFE38CA	8、16
	端口 G 的控制寄存器 1	PGCR1	16	H'FFFE38CC	8、16、32
	端口 G 的控制寄存器 0	PGCR0	16	H'FFFE38CE	8、16
	端口 G 的 IO 寄存器 1	PGIOR1	16	H'FFFE38D0	8、16、32
	端口 G 的 IO 寄存器 0	PGIOR0	16	H'FFFE38D2	8、16
	端口 G 的数据寄存器 1	PGDR1	16	H'FFFE38D4	8、16、32
	端口 G 的数据寄存器 0	PGDR0	16	H'FFFE38D6	8、16
	端口 G 的端口寄存器 1	PGPR1	16	H'FFFE38D8	8、16、32
	端口 G 的端口寄存器 0	PGPR0	16	H'FFFE38DA	8、16
	端口 H 的控制寄存器 1	PHCR1	16	H'FFFE38EC	8、16、32
	端口 H 的控制寄存器 0	PHCR0	16	H'FFFE38EE	8、16
	端口 H 的端口寄存器 0	PHPR0	16	H'FFFE38FA	8、16
	端口 J 的控制寄存器 7	PJCR7	16	H'FFFE3900	8、16、32
	端口 J 的控制寄存器 6	PJCR6	16	H'FFFE3902	8、16
	端口 J 的控制寄存器 5	PJCR5	16	H'FFFE3904	8、16、32

模块名	寄存器名	略称	位数	地址	存取长度
通用输入 / 输出 端口	端口 J 的控制寄存器 4	PJCR4	16	H'FFFE3906	8、16
	端口 J 的控制寄存器 3	PJCR3	16	H'FFFE3908	8、16、32
	端口 J 的控制寄存器 2	PJCR2	16	H'FFFE390A	8、16
	端口 J 的控制寄存器 1	PJCR1	16	H'FFFE390C	8、16、32
	端口 J 的控制寄存器 0	PJCR0	16	H'FFFE390E	8、16
	端口 J 的 IO 寄存器 1	PJIOR1	16	H'FFFE3910	8、16、32
	端口 J 的 IO 寄存器 0	PJIOR0	16	H'FFFE3912	8、16
	端口 J 的数据寄存器 1	PJDR1	16	H'FFFE3914	8、16、32
	端口 J 的数据寄存器 0	PJDR0	16	H'FFFE3916	8、16
	端口 J 的端口寄存器 1	PJPR1	16	H'FFFE3918	8、16、32
	端口 J 的端口寄存器 0	PJPR0	16	H'FFFE391A	8、16
	串行音频接口噪声消除器的控制寄存器	SNCR	16	H'FFFE393E	8、16
	低功耗模式	待机控制寄存器 1	STBCR1	8	H'FFFE0014
待机控制寄存器 2		STBCR2	8	H'FFFE0018	8
待机控制寄存器 3		STBCR3	8	H'FFFE0408	8
待机控制寄存器 4		STBCR4	8	H'FFFE040C	8
待机控制寄存器 5		STBCR5	8	H'FFFE0410	8
待机控制寄存器 6		STBCR6	8	H'FFFE0414	8
待机控制寄存器 7		STBCR7	8	H'FFFE0418	8
待机控制寄存器 8		STBCR8	8	H'FFFE041C	8
待机控制寄存器 9		STBCR9	8	H'FFFE0440	8
待机控制寄存器 10		STBCR10	8	H'FFFE0444	8
软件复位控制寄存器 1		SWRSTCR1	8	H'FFFE0430	8
软件复位控制寄存器 2		SWRSTCR2	8	H'FFFE0434	8
系统控制寄存器 1		SYSCR1	8	H'FFFE0400	8
系统控制寄存器 2		SYSCR2	8	H'FFFE0404	8
系统控制寄存器 3		SYSCR3	8	H'FFFE0420	8
系统控制寄存器 4		SYSCR4	8	H'FFFE0424	8
系统控制寄存器 5		SYSCR5	8	H'FFFE0428	8
内部 RAM 的保持区指定寄存器		RRAMKP	8	H'FFFE6800	8
深度待机控制寄存器		DSCTR	8	H'FFFE6802	8
深度待机解除源选择寄存器		DSSSR	16	H'FFFE6804	16
深度待机解除边沿选择寄存器		DSESR	16	H'FFFE6806	16
深度待机解除源标志寄存器	DSFR	16	H'FFFE6808	16	
XTAL 晶体振荡器的增益控制寄存器	XTALCTR	8	H'FFFE6810	8	
用户调试接口	指令寄存器	SDIR	16	H'FFFE2000	16

51.2 寄存器位一览表

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
时钟脉冲 振荡器	FRQCR	—	CKOEN2	CKOEN[1]	CKOEN[0]	—	—	IFC[1]	IFC[0]
		—	—	BFC[1]	BFC[0]	—	—	—	—
中断控制器	ICR0	NMIL	—	—	—	—	—	—	NMIE
		—	—	—	—	—	—	—	—
	ICR1	IRQ71S	IRQ70S	IRQ61S	IRQ60S	IRQ51S	IRQ50S	IRQ41S	IRQ40S
		IRQ31S	IRQ30S	IRQ21S	IRQ20S	IRQ11S	IRQ10S	IRQ01S	IRQ00S
	ICR2	—	—	—	—	—	—	—	—
		PINT7S	PINT6S	PINT5S	PINT4S	PINT3S	PINT2S	PINT1S	PINT0S
	IRQRR	—	—	—	—	—	—	—	—
		IRQ7F	IRQ6F	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F
	PINTER	—	—	—	—	—	—	—	—
		PINT7E	PINT6E	PINT5E	PINT4E	PINT3E	PINT2E	PINT1E	PINT0E
	PIRR	—	—	—	—	—	—	—	—
		PINT7R	PINT6R	PINT5R	PINT4R	PINT3R	PINT2R	PINT1R	PINT0R
	IBCR	E15	E14	E13	E12	E11	E10	E9	E8
		E7	E6	E5	E4	E3	E2	E1	—
	IBNR	BE[1]	BE[0]	BOVE	—	—	—	—	—
		—	—	—	—	BN[3]	BN[2]	BN[1]	BN[0]
	IPR01								
	IPR02								
	IPR05								
	IPR06								
	IPR07								
	IPR08								
	IPR09								
	IPR10								
	IPR11								
	IPR12								
	IPR13								
	IPR14								
IPR15									

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
中断控制器	IPR16									
	IPR17									
	IPR18									
	IPR19									
	IPR20									
	IPR21									
	IPR22									
	IPR23									
	IPR24									
	IPR25									
	IPR26									
	高速缓存	CCR1	—	—	—	—	—	—	—	—
			—	—	—	—	—	—	—	—
—			—	—	—	ICF	—	—	ICE	
—			—	—	—	OCF	—	WT	OCE	
CCR2		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	LE	
		—	—	—	—	—	—	W3LOAD	W3LOCK	
		—	—	—	—	—	—	W2LOAD	W2LOCK	
用户断点 控制器	BAR_0	BA31	BA30	BA29	BA28	BA27	BA26	BA25	BA24	
		BA23	BA22	BA21	BA20	BA19	BA18	BA17	BA16	
		BA15	BA14	BA13	BA12	BA11	BA10	BA9	BA8	
		BA7	BA6	BA5	BA4	BA3	BA2	BA1	BA0	
	BAMR_0	BAM31	BAM30	BAM29	BAM28	BAM27	BAM26	BAM25	BAM24	
		BAM23	BAM22	BAM21	BAM20	BAM19	BAM18	BAM17	BAM16	
		BAM15	BAM14	BAM13	BAM12	BAM11	BAM10	BAM9	BAM8	
		BAM7	BAM6	BAM5	BAM4	BAM3	BAM2	BAM1	BAM0	
	BBR_0	—	—	UBID	DBE	—	—	CP[1]	CP[0]	
		CD[1]	CD[0]	ID[1]	ID[0]	RW[1]	RW[0]	SZ[1]	SZ[0]	
	BDR_0	BD31	BD30	BD29	BD28	BD27	BD26	BD25	BD24	
		BD23	BD22	BD21	BD20	BD19	BD18	BD17	BD16	
		BD15	BD14	BD13	BD12	BD11	BD10	BD9	BD8	
		BD7	BD6	BD5	BD4	BD3	BD2	BD1	BD0	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
用户断点 控制器	BDMR_0	BDM31	BDM30	BDM29	BDM28	BDM27	BDM26	BDM25	BDM24	
		BDM23	BDM22	BDM21	BDM20	BDM19	BDM18	BDM17	BDM16	
		BDM15	BDM14	BDM13	BDM12	BDM11	BDM10	BDM9	BDM8	
		BDM7	BDM6	BDM5	BDM4	BDM3	BDM2	BDM1	BDM0	
	BAR_1	BA31	BA30	BA29	BA28	BA27	BA26	BA25	BA24	
		BA23	BA22	BA21	BA20	BA19	BA18	BA17	BA16	
		BA15	BA14	BA13	BA12	BA11	BA10	BA9	BA8	
		BA7	BA6	BA5	BA4	BA3	BA2	BA1	BA0	
	BAMR_1	BAM31	BAM30	BAM29	BAM28	BAM27	BAM26	BAM25	BAM24	
		BAM23	BAM22	BAM21	BAM20	BAM19	BAM18	BAM17	BAM16	
		BAM15	BAM14	BAM13	BAM12	BAM11	BAM10	BAM9	BAM8	
		BAM7	BAM6	BAM5	BAM4	BAM3	BAM2	BAM1	BAM0	
	BBR_1	—	—	UBID	DBE	—	—	CP[1]	CP[0]	
		CD[1]	CD[0]	ID[1]	ID[0]	RW[1]	RW[0]	SZ[1]	SZ[0]	
	BDR_1	BD31	BD30	BD29	BD28	BD27	BD26	BD25	BD24	
		BD23	BD22	BD21	BD20	BD19	BD18	BD17	BD16	
		BD15	BD14	BD13	BD12	BD11	BD10	BD9	BD8	
		BD7	BD6	BD5	BD4	BD3	BD2	BD1	BD0	
	BDMR_1	BDM31	BDM30	BDM29	BDM28	BDM27	BDM26	BDM25	BDM24	
		BDM23	BDM22	BDM21	BDM20	BDM19	BDM18	BDM17	BDM16	
		BDM15	BDM14	BDM13	BDM12	BDM11	BDM10	BDM9	BDM8	
		BDM7	BDM6	BDM5	BDM4	BDM3	BDM2	BDM1	BDM0	
	BRCCR	—	—	—	—	—	—	—	—	
		—	—	—	—	UTOD1	UTOD0	CKS[1]	CKS[0]	
		SCMFC0	SCMFC1	SCMFD0	SCMFD1	—	—	—	—	
		—	PCB1	PCB0	—	—	—	—	—	
	总线状态 控制器	CMNCR	—	—	—	—	—	—	—	—
			—	—	—	—	—	—	—	—
			—	—	—	—	BLOCK	DPRTY[1]	DPRTY[0]	DMAIW[2]
			DMAIW[1]	DMAIW[0]	DMAIWA	—	—	—	HIZMEM	HIZCNT
		CS0BCR	—	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]
			IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]
			—	TYPE[2]	TYPE[1]	TYPE[0]	ENDIAN	BSZ[1]	BSZ[0]	—
			—	—	—	—	—	—	—	—
		CS1BCR	—	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]
			IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]
			—	TYPE[2]	TYPE[1]	TYPE[0]	ENDIAN	BSZ[1]	BSZ[0]	—
			—	—	—	—	—	—	—	—
		CS2BCR	—	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]
			IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]
—			TYPE[2]	TYPE[1]	TYPE[0]	ENDIAN	BSZ[1]	BSZ[0]	—	
—			—	—	—	—	—	—	—	
CS3BCR		—	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]	
		IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]	
		—	TYPE[2]	TYPE[1]	TYPE[0]	ENDIAN	BSZ[1]	BSZ[0]	—	
		—	—	—	—	—	—	—	—	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
总线状态 控制器	CS4BCR	—	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]
		IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]
		—	TYPE[2]	TYPE[1]	TYPE[0]	ENDIAN	BSZ[1]	BSZ[0]	—
		—	—	—	—	—	—	—	—
	CS5BCR	—	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]
		IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]
		—	TYPE[2]	TYPE[1]	TYPE[0]	ENDIAN	BSZ[1]	BSZ[0]	—
		—	—	—	—	—	—	—	—
	CS0WCR	—	—	—	—	—	—	—	—
		—	—	—	BAS	—	—	—	—
		—	—	—	SW[1]	SW[0]	WR[3]	WR[2]	WR[1]
		WR[0]	WM	—	—	—	—	HW[1]	HW[0]
	CS0WCR	—	—	—	—	—	—	—	—
		—	—	BST[1]	BST[0]	—	—	BW[1]	BW[0]
		—	—	—	—	—	WR[3]	WR[2]	WR[1]
		WR[0]	WM	—	—	—	—	—	—
	CS0WCR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	BW[1]	BW[0]
		—	—	—	—	—	WR[3]	WR[2]	WR[1]
		WR[0]	WM	—	—	—	—	—	—
	CS1WCR	—	—	—	—	—	—	—	—
		—	—	—	BAS	—	WW[2]	WW[1]	WW[0]
		—	—	—	SW[1]	SW[0]	WR[3]	WR[2]	WR[1]
		WR[0]	WM	—	—	—	—	HW[1]	HW[0]
	CS2WCR	—	—	—	—	—	—	—	—
		—	—	—	BAS	—	—	—	—
		—	—	—	—	—	WR[3]	WR[2]	WR[1]
		WR[0]	WM	—	—	—	—	—	—
	CS2WCR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	A2CL1
		A2CL0	—	—	—	—	—	—	—
	CS3WCR	—	—	—	—	—	—	—	—
		—	—	—	BAS	—	—	—	—
		—	—	—	—	—	WR[3]	WR[2]	WR[1]
		WR[0]	WM	—	—	—	—	—	—
	CS3WCR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	WTRP[1]	WTRP[0]	—	WTRCD[1]	WTRCD[0]	—	A3CL1
		A3CL0	—	—	TRWL[1]	TRWL[0]	—	WTRC[1]	WTRC[0]
CS4WCR	—	—	—	—	—	—	—	—	
	—	—	—	BAS	—	WW[2]	WW[1]	WW[0]	
	—	—	—	SW[1]	SW[0]	WR[3]	WR[2]	WR[1]	
	WR[0]	WM	—	—	—	—	HW[1]	HW[0]	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
总线状态 控制器	CS4WCR	—	—	—	—	—	—	—	—
		—	—	BST[1]	BST[0]	—	—	BW[1]	BW[0]
		—	—	—	SW[1]	SW[0]	WR[3]	WR[2]	WR[1]
		W[0]	WM	—	—	—	—	HW[1]	HW[0]
	CS5WCR	—	—	—	—	—	—	—	—
		—	—	SZSEL	MPXW/BAS	—	WW[2]	WW[1]	WW[0]
		—	—	—	SW[1]	SW[0]	WR[3]	WR[2]	WR[1]
		WR[0]	WM	—	—	—	—	HW[1]	HW[0]
	CS5WCR	—	—	—	—	—	—	—	—
		—	—	SA[1]	SA[0]	—	—	—	—
		—	TED[3]	TED[2]	TED[1]	TED[0]	PCW[3]	PCW[2]	PCW[1]
		PCW[0]	WM	—	—	TEH[3]	TEH[2]	TEH[1]	TEH[0]
	SDCR	—	—	—	—	—	—	—	—
		—	—	—	A2ROW[1]	A2ROW[0]	—	A2COL[1]	A2COL[0]
		—	—	DEEP	—	RFSH	RMODE	PDOWN	BACTV
		—	—	—	A3ROW[1]	A3ROW[0]	—	A3COL[1]	A3COL[0]
	RTCSR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		CMF	CMIE	CKS[2]	CKS[1]	CKS[0]	RRC[2]	RRC[1]	RRC[0]
	RTCNT	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
	RTCOR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
直接存储器 存取控制器	SAR0	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
	DAR0	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
	DMATCR0	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
	CHCR0	TC	—	RLDSAR	RLDDAR	—	DAF	SAF	—
		DO	TL	—	TEMASK	HE	HIE	AM	AL
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		DL	DS	TB	TS[1]	TS[0]	IE	TE	DE

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
直接存储器 存取控制器	RSAR0								
	RDAR0								
	RDMATCR0	—	—	—	—	—	—	—	—
	SAR1								
	DAR1								
	DMATCR1	—	—	—	—	—	—	—	—
	CHCR1	TC	—	RLDSAR	RLDDAR	—	DAF	SAF	—
		DO	TL	—	TEMASK	HE	HIE	AM	AL
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		DL	DS	TB	TS[1]	TS[0]	IE	TE	DE
	RSAR1								
	RDAR1								
	RDMATCR1	—	—	—	—	—	—	—	—
	SAR2								

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
直接存储器 存取控制器	DAR2								
	DMATCR2	—	—	—	—	—	—	—	—
	CHCR2	TC	—	RLDSAR	RLDDAR	—	DAF	SAF	—
		—	—	—	TEMASK	HE	HIE	—	—
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		—	—	TB	TS[1]	TS[0]	IE	TE	DE
	RSAR2								
	RDAR2								
	RDMATCR2	—	—	—	—	—	—	—	—
	SAR3								
	DAR3								
	DMATCR3	—	—	—	—	—	—	—	—
	CHCR3	TC	—	RLDSAR	RLDDAR	—	DAF	SAF	—
		—	—	—	TEMASK	HE	HIE	—	—
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		—	—	TB	TS[1]	TS[0]	IE	TE	DE
	RSAR3								

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
直接存储器 存取控制器	RDAR3								
	RDMATCR3	—	—	—	—	—	—	—	—
	SAR4								
	DAR4								
	DMATCR4	—	—	—	—	—	—	—	—
	CHCR4	TC	—	RLDSAR	RLDDAR	—	DAF	SAF	—
		—	—	—	TEMASK	HE	HIE	—	—
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		—	—	TB	TS[1]	TS[0]	IE	TE	DE
	RSAR4								
	RDAR4								
	RDMATCR4	—	—	—	—	—	—	—	—
	SAR5								
	DAR5								

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
直接存储器 存取控制器	DMATCR5	—	—	—	—	—	—	—	—	
	CHCR5	TC	—	—	RLDSAR	RLDDAR	—	DAF	SAF	—
		—	—	—	TEMASK	HE	HIE	—	—	—
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]	—
		—	—	TB	TS[1]	TS[0]	IE	TE	DE	—
	RSAR5									
	RDAR5									
	RDMATCR5	—	—	—	—	—	—	—	—	—
	SAR6									
	DAR6									
	DMATCR6	—	—	—	—	—	—	—	—	—
	CHCR6	TC	—	—	RLDSAR	RLDDAR	—	DAF	SAF	—
		—	—	—	TEMASK	HE	HIE	—	—	—
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]	—
		—	—	TB	TS[1]	TS[0]	IE	TE	DE	—
	RSAR6									
	RDAR6									

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
直接存储器 存取控制器	RDMATCR6	—	—	—	—	—	—	—	—
	SAR7								
	DAR7								
	DMATCR7	—	—	—	—	—	—	—	—
	CHCR7	TC	—	RLDSAR	RLDDAR	—	DAF	SAF	—
		—	—	—	TEMASK	HE	HIE	—	—
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		—	—	TB	TS[1]	TS[0]	IE	TE	DE
	RSAR7								
	RDAR7								
	RDMATCR7	—	—	—	—	—	—	—	—
	SAR8								
	DAR8								
	DMATCR8	—	—	—	—	—	—	—	—

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
直接存储器 存取控制器	CHCR8	TC	—	RLDSAR	RLDDAR	—	DAF	SAF	—	
		—	—	—	TEMASK	HE	HIE	—	—	
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]	
		—	—	TB	TS[1]	TS[0]	IE	TE	DE	
	RSAR8									
	RDAR8									
	RDMATCR8	—	—	—	—	—	—	—	—	—
	SAR9									
	DAR9									
	DMATCR9	—	—	—	—	—	—	—	—	—
	CHCR9	TC	—	RLDSAR	RLDDAR	—	DAF	SAF	—	
		—	—	—	TEMASK	HE	HIE	—	—	
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]	
		—	—	TB	TS[1]	TS[0]	IE	TE	DE	
	RSAR9									
	RDAR9									
	RDMATCR9	—	—	—	—	—	—	—	—	—

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
直接存储器 存取控制器	SAR10								
	DAR10								
	DMATCR10	—	—	—	—	—	—	—	—
	CHCR10	TC	—	RLDSAR	RLDDAR	—	DAF	SAF	—
		—	—	—	TEMASK	HE	HIE	—	—
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		—	—	TB	TS[1]	TS[0]	IE	TE	DE
	RSAR10								
	RDAR10								
	RDMATCR10	—	—	—	—	—	—	—	—
	SAR11								
	DAR11								
	DMATCR11	—	—	—	—	—	—	—	—
	CHCR11	TC	—	RLDSAR	RLDDAR	—	DAF	SAF	—
		—	—	—	TEMASK	HE	HIE	—	—
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		—	—	TB	TS[1]	TS[0]	IE	TE	DE

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
直接存储器 存取控制器	RSAR11								
	RDAR11								
	RDMATCR11	—	—	—	—	—	—	—	—
	SAR12								
	DAR12								
	DMATCR12	—	—	—	—	—	—	—	—
	CHCR12	TC	—	RLDSAR	RLDDAR	—	DAF	SAF	—
		—	—	—	TEMASK	HE	HIE	—	—
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		—	—	TB	TS[1]	TS[0]	IE	TE	DE
	RSAR12								
	RDAR12								
	RDMATCR12	—	—	—	—	—	—	—	—
	SAR13								

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
直接存储器 存取控制器	DAR13								
	DMATCR13	—	—	—	—	—	—	—	—
	CHCR13	TC	—	RLDSAR	RLDDAR	—	DAF	SAF	—
		—	—	—	TEMASK	HE	HIE	—	—
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		—	—	TB	TS[1]	TS[0]	IE	TE	DE
	RSAR13								
	RDAR13								
	RDMATCR13	—	—	—	—	—	—	—	—
	SAR14								
	DAR14								
	DMATCR14	—	—	—	—	—	—	—	—
	CHCR14	TC	—	RLDSAR	RLDDAR	—	DAF	SAF	—
		—	—	—	TEMASK	HE	HIE	—	—
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		—	—	TB	TS[1]	TS[0]	IE	TE	DE
	RSAR14								

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
直接存储器 存取控制器	RDAR14								
	RDMATCR14	—	—	—	—	—	—	—	—
	SAR15								
	DAR15								
	DMATCR15	—	—	—	—	—	—	—	—
	CHCR15	TC	—	RLDSAR	RLDDAR	—	DAF	SAF	—
		—	—	—	TEMASK	HE	HIE	—	—
		DM[1]	DM[0]	SM[1]	SM[0]	RS[3]	RS[2]	RS[1]	RS[0]
		—	—	TB	TS[1]	TS[0]	IE	TE	DE
	RSAR15								
	RDAR15								
	RDMATCR15	—	—	—	—	—	—	—	—
	DMAOR	—	—	CMS[1]	CMS[0]	—	—	PR[1]	PR[0]
		—	—	—	—	—	AE	NMIF	DME
	DMARS0	CH1MID[5]	CH1MID[4]	CH1MID[3]	CH1MID[2]	CH1MID[1]	CH1MID[0]	CH1RID[1]	CH1RID[0]
		CH0MID[5]	CH0MID[4]	CH0MID[3]	CH0MID[2]	CH0MID[1]	CH0MID[0]	CH0RID[1]	CH0RID[0]
	DMARS1	CH3MID[5]	CH3MID[4]	CH3MID[3]	CH3MID[2]	CH3MID[1]	CH3MID[0]	CH3RID[1]	CH3RID[0]
		CH2MID[5]	CH2MID[4]	CH2MID[3]	CH2MID[2]	CH2MID[1]	CH2MID[0]	CH2RID[1]	CH2RID[0]
	DMARS2	CH5MID[5]	CH5MID[4]	CH5MID[3]	CH5MID[2]	CH5MID[1]	CH5MID[0]	CH5RID[1]	CH5RID[0]
		CH4MID[5]	CH4MID[4]	CH4MID[3]	CH4MID[2]	CH4MID[1]	CH4MID[0]	CH4RID[1]	CH4RID[0]
	DMARS3	CH7MID[5]	CH7MID[4]	CH7MID[3]	CH7MID[2]	CH7MID[1]	CH7MID[0]	CH7RID[1]	CH7RID[0]
		CH6MID[5]	CH6MID[4]	CH6MID[3]	CH6MID[2]	CH6MID[1]	CH6MID[0]	CH6RID[1]	CH6RID[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
直接存储器 存取控制器	DMARS4	CH9MID[5]	CH9MID[4]	CH9MID[3]	CH9MID[2]	CH9MID[1]	CH9MID[0]	CH9RID[1]	CH9RID[0]	
		CH8MID[5]	CH8MID[4]	CH8MID[3]	CH8MID[2]	CH8MID[1]	CH8MID[0]	CH8RID[1]	CH8RID[0]	
	DMARS5	CH11MID[5]	CH11MID[4]	CH11MID[3]	CH11MID[2]	CH11MID[1]	CH11MID[0]	CH11RID[1]	CH11RID[0]	
		CH10MID[5]	CH10MID[4]	CH10MID[3]	CH10MID[2]	CH10MID[1]	CH10MID[0]	CH10RID[1]	CH10RID[0]	
	DMARS6	CH13MID[5]	CH13MID[4]	CH13MID[3]	CH13MID[2]	CH13MID[1]	CH13MID[0]	CH13RID[1]	CH13RID[0]	
		CH12MID[5]	CH12MID[4]	CH12MID[3]	CH12MID[2]	CH12MID[1]	CH12MID[0]	CH12RID[1]	CH12RID[0]	
	DMARS7	CH15MID[5]	CH15MID[4]	CH15MID[3]	CH15MID[2]	CH15MID[1]	CH15MID[0]	CH15RID[1]	CH15RID[0]	
		CH14MID[5]	CH14MID[4]	CH14MID[3]	CH14MID[2]	CH14MID[1]	CH14MID[0]	CH14RID[1]	CH14RID[0]	
	多功能定时 器脉冲单元 2	TCR_0	CCLR[2]	CCLR[1]	CCLR[0]	CKEG[1]	CKEG[0]	TPSC[2]	TPSC[1]	TPSC[0]
		TMDR_0	—	BFE	BFB	BFA	MD[3]	MD[2]	MD[1]	MD[0]
TIORH_0		IOB[3]	IOB[2]	IOB[1]	IOB[0]	IOA[3]	IOA[2]	IOA[1]	IOA[0]	
TIORL_0		IOD[3]	IOD[2]	IOD[1]	IOD[0]	IOC[3]	IOC[2]	IOC[1]	IOC[0]	
TIER_0		TTGE	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA	
TSR_0		TCFD	—	—	TCFV	TGFD	TGFC	TGFB	TGFA	
TCNT_0										
TGRA_0										
TGRB_0										
TGRC_0										
TGRD_0										
TGRE_0										
TGRF_0										
TIER2_0		TTGE2	—	—	—	—	—	TGIEF	TGIEE	
TSR2_0		—	—	—	—	—	—	TGFF	TGFE	
TBTM_0		—	—	—	—	—	TTSE	TTSB	T TSA	
TCR_1		—	CCLR[1]	CCLR[0]	CKEG[1]	CKEG[0]	TPSC[2]	TPSC[1]	TPSC[0]	
TMDR_1		—	—	—	—	MD[3]	MD[2]	MD[1]	MD[0]	
TIOR_1		IOB[3]	IOB[2]	IOB[1]	IOB[0]	IOA[3]	IOA[2]	IOA[1]	IOA[0]	
TIER_1		TTGE	—	TCIEU	TCIEV	—	—	TGIEB	TGIEA	
TSR_1		TCFD	—	TCFU	TCFV	TGFD	TGFC	TGFB	TGFA	
TCNT_1										
TGRA_1										
TGRB_1										
TICCR		—	—	—	—	I2BE	I2AE	I1BE	I1AE	
TCR_2		—	CCLR[1]	CCLR[0]	CKEG[1]	CKEG[0]	TPSC[2]	TPSC[1]	TPSC[0]	
TMDR_2		—	—	—	—	MD[3]	MD[2]	MD[1]	MD[0]	
TIOR_2		IOB[3]	IOB[2]	IOB[1]	IOB[0]	IOA[3]	IOA[2]	IOA[1]	IOA[0]	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
多功能定时器脉冲单元 2	TIER_2	TTGE	—	TCIEU	TCIEV	—	—	TGIEB	TGIEA	
	TSR_2	TCFD	—	TCFU	TCFV	TGFD	TGFC	TGFB	TGFA	
	TCNT_2									
	TGRA_2									
	TGRB_2									
	TCR_3	CCLR[2]	CCLR[1]	CCLR[0]	CKEG[1]	CKEG[0]	TPSC[2]	TPSC[1]	TPSC[0]	
	TMDR_3	—	—	BFB	BFA	MD[3]	MD[2]	MD[1]	MD[0]	
	TIORH_3	IOB[3]	IOB[2]	IOB[1]	IOB[0]	IOA[3]	IOA[2]	IOA[1]	IOA[0]	
	TIORL_3	IOD[3]	IOD[2]	IOD[1]	IOD[0]	IOC[3]	IOC[2]	IOC[1]	IOC[0]	
	TIER_3	TTGE	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA	
	TSR_3	TCFD	—	—	TCFV	TGFD	TGFC	TGFB	TGFA	
	TCNT_3									
	TGRA_3									
	TGRB_3									
	TGRC_3									
	TGRD_3									
	TBTM_3	—	—	—	—	—	—	—	TTSB	TTSA
	TCR_4	CCLR[2]	CCLR[1]	CCLR[0]	CKEG[1]	CKEG[0]	TPSC[2]	TPSC[1]	TPSC[0]	
	TMDR_4	—	—	BFB	BFA	MD[3]	MD[2]	MD[1]	MD[0]	
	TIORH_4	IOB[3]	IOB[2]	IOB[1]	IOB[0]	IOA[3]	IOA[2]	IOA[1]	IOA[0]	
	TIORL_4	IOD[3]	IOD[2]	IOD[1]	IOD[0]	IOC[3]	IOC[2]	IOC[1]	IOC[0]	
	TIER_4	TTGE	TTGE2	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA	
	TSR_4	TCFD	—	—	TCFV	TGFD	TGFC	TGFB	TGFA	
	TCNT_4									
	TGRA_4									
	TGRB_4									
	TGRC_4									
	TGRD_4									
	TBTM_4	—	—	—	—	—	—	—	TTSB	TTSA
	TADCR	BF[1]	BF[0]	—	—	—	—	—	—	—
		UT4AE	DT4AE	UT4BE	DT4BE	ITA3AE	ITA4VE	ITB3AE	ITB4VE	
	TADCORA_4									

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
多功能定时器脉冲单元 2	TADCORB_4								
	TADCOBRA_4								
	TADCOBRB_4								
	TSTR	CST4	CST3	—	—	—	CST2	CST1	CST0
	TSYR	SYNC4	SYNC3	—	—	—	SYNC2	SYNC1	SYNC0
	TRWER	—	—	—	—	—	—	—	RWE
	TOER	—	—	OE4D	OE4C	OE3D	OE4B	OE4A	OE3B
	TOCR1	—	PSYE	—	—	TOCL	TOCS	OLSN	OLSP
	TOCR2	BF[1]	BF[0]	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
	TGCR	—	BDC	N	P	FB	WF	VF	UF
	TCDR								
	TDDR								
	TCNTS								
	TCBR								
	TITCR	T3AEN	3ACOR[2]	3ACOR[1]	3ACOR[0]	T4VEN	4VCOR[2]	4VCOR[1]	4VCOR[0]
	TITCNT	—	3ACNT[2]	3ACNT[1]	3ACNT[0]	—	4VCNT[2]	4VCNT[1]	4VCNT[0]
	TBTER	—	—	—	—	—	—	BTE[1]	BTE[0]
	TDER	—	—	—	—	—	—	—	TDER
	TWCR	CCE	—	—	—	—	—	—	WRE
	TOLBR	—	—	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
比较匹配定时器	CMSTR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	STR1	STR0
	CMCSR_0	—	—	—	—	—	—	—	—
		CMF	CMIE	—	—	—	—	CKS[1]	CKS[0]
	CMCNT_0								
	CMCOR_0								
	CMCSR_1	—	—	—	—	—	—	—	—
		CMF	CMIE	—	—	—	—	CKS[1]	CKS[0]
CMCNT_1									
CMCOR_1									
看门狗定时器	WTCNT								
	WTCSR	IOVF	WT/IT	TME	—	—	CKS[2]	CKS[1]	CKS[0]
	WRCSR	WOVF	RSTE	RSTS	—	—	—	—	—
实时时钟	R64CNT	—	1Hz	2Hz	4Hz	8Hz	16Hz	32Hz	64Hz
	RSECCNT	—	10 秒 [2]	10 秒 [1]	10 秒 [0]	1 秒 [3]	1 秒 [2]	1 秒 [1]	1 秒 [0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
实时时钟	RMINCNT	—	10 分钟 [2]	10 分钟 [1]	10 分钟 [0]	1 分钟 [3]	1 分钟 [2]	1 分钟 [1]	1 分钟 [0]
	RHRCNT	—	—	10 小时 [1]	10 小时 [0]	1 小时 [3]	1 小时 [2]	1 小时 [1]	1 小时 [0]
	RWKCNT	—	—	—	—	—	星期 [2]	星期 [1]	星期 [0]
	RDAYCNT	—	—	10 日 [1]	10 日 [0]	1 日 [3]	1 日 [2]	1 日 [1]	1 日 [0]
	RMONCNT	—	—	—	10 月	1 月 [3]	1 月 [2]	1 月 [1]	1 月 [0]
	RYRCNT	1000 年 [3]	1000 年 [2]	1000 年 [1]	1000 年 [0]	100 年 [3]	100 年 [2]	100 年 [1]	100 年 [0]
		10 年 [3]	10 年 [2]	10 年 [1]	10 年 [0]	1 年 [3]	1 年 [2]	1 年 [1]	1 年 [0]
	RSECAR	ENB	10 秒 [2]	10 秒 [1]	10 秒 [0]	1 秒 [3]	1 秒 [2]	1 秒 [1]	1 秒 [0]
	RMINAR	ENB	10 分钟 [2]	10 分钟 [1]	10 分钟 [0]	1 分钟 [3]	1 分钟 [2]	1 分钟 [1]	1 分钟 [0]
	RHRAR	ENB	—	10 小时 [1]	10 小时 [0]	1 小时 [3]	1 小时 [2]	1 小时 [1]	1 小时 [0]
	RWKAR	ENB	—	—	—	—	星期 [2]	星期 [1]	星期 [0]
	RDAYAR	ENB	—	10 日 [1]	10 日 [0]	1 日 [3]	1 日 [2]	1 日 [1]	1 日 [0]
	RMONAR	ENB	—	—	10 月	1 月 [3]	1 月 [2]	1 月 [1]	1 月 [0]
	RYRAR	1000 年 [3]	1000 年 [2]	1000 年 [1]	1000 年 [0]	100 年 [3]	100 年 [2]	100 年 [1]	100 年 [0]
		10 年 [3]	10 年 [2]	10 年 [1]	10 年 [0]	1 年 [3]	1 年 [2]	1 年 [1]	1 年 [0]
	RCR1	CF	—	—	CIE	AIE	—	—	AF
	RCR2	PEF	PES[2]	PES[1]	PES[0]	RTCEN	ADJ	RESET	START
	RCR3	ENB	—	—	—	—	—	—	—
	RCR5	—	—	—	—	—	—	—	RCKSEL
	RFRH	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	RFC[16]
	RFRL	RFC[15]	RFC[14]	RFC[13]	RFC[12]	RFC[11]	RFC[10]	RFC[9]	RFC[8]
		RFC[7]	RFC[6]	RFC[5]	RFC[4]	RFC[3]	RFC[2]	RFC[1]	RFC[0]
	FIFO 内置型 串行通信单 元	SCSMR_0	—	—	—	—	—	—	—
			C/A	CHR	PE	O/E	STOP	—	CKS[1]
		SCBRR_0							
SCSCR_0		—	—	—	—	—	—	—	—
		TIE	RIE	TE	RE	REIE	—	CKE[1]	CKE[0]
SCFTDR_0									
SCFCSR_0		PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]
		ER	TEND	TDFE	BRK	FER	PER	RDF	DR
SCFRDR_0									
SCFCR_0		—	—	—	—	—	RSTRG[2]	RSTRG[1]	RSTRG[0]
		RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP
SCFDR_0		—	—	—	T[4]	T[3]	T[2]	T[1]	T[0]
		—	—	—	R[4]	R[3]	R[2]	R[1]	R[0]
SCSPTR_0		—	—	—	—	—	—	—	—
		—	—	—	—	SCKIO	SCKDT	SPB2IO	SPB2DT
SCLSR_0		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	ORER
SCEMR_0		—	—	—	—	—	—	—	—
		BGDM	—	—	—	—	—	—	ABCS
SCSMR_1		—	—	—	—	—	—	—	—
		C/A	CHR	PE	O/E	STOP	—	CKS[1]	CKS[0]
SCBRR_1									

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
FIFO 内置型 串行通信单 元	SCSCR_1	—	—	—	—	—	—	—	—
		TIE	RIE	TE	RE	REIE	—	CKE[1]	CKE[0]
	SCFTDR_1								
	SCFSR_1	PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]
		ER	TEND	TDFE	BRK	FER	PER	RDF	DR
	SCFRDR_1								
	SCFCR_1	—	—	—	—	—	RSTRG[2]	RSTRG[1]	RSTRG[0]
		RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP
	SCFDR_1	—	—	—	T[4]	T[3]	T[2]	T[1]	T[0]
		—	—	—	R[4]	R[3]	R[2]	R[1]	R[0]
	SCSPTR_1	—	—	—	—	—	—	—	—
		RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPB2IO	SPB2DT
	SCLSR_1	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	ORER
	SCEMR_1	—	—	—	—	—	—	—	—
		BGDM	—	—	—	—	—	—	ABCS
	SCSMR_2	—	—	—	—	—	—	—	—
		C/A	CHR	PE	O/E	STOP	—	CKS[1]	CKS[0]
	SCBRR_2								
	SCSCR_2	—	—	—	—	—	—	—	—
		TIE	RIE	TE	RE	REIE	—	CKE[1]	CKE[0]
	SCFTDR_2								
	SCFSR_2	PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]
		ER	TEND	TDFE	BRK	FER	PER	RDF	DR
	SCFRDR_2								
	SCFCR_2	—	—	—	—	—	RSTRG[2]	RSTRG[1]	RSTRG[0]
		RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP
	SCFDR_2	—	—	—	T[4]	T[3]	T[2]	T[1]	T[0]
		—	—	—	R[4]	R[3]	R[2]	R[1]	R[0]
	SCSPTR_2	—	—	—	—	—	—	—	—
		—	—	—	—	SCKIO	SCKDT	SPB2IO	SPB2DT
	SCLSR_2	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	ORER
	SCEMR_2	—	—	—	—	—	—	—	—
		BGDM	—	—	—	—	—	—	ABCS
	SCSMR_3	—	—	—	—	—	—	—	—
		C/A	CHR	PE	O/E	STOP	—	CKS[1]	CKS[0]
	SCBRR_3								
	SCSCR_3	—	—	—	—	—	—	—	—
		TIE	RIE	TE	RE	REIE	—	CKE[1]	CKE[0]
	SCFTDR_3								
	SCFSR_3	PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]
ER		TEND	TDFE	BRK	FER	PER	RDF	DR	
SCFRDR_3									
SCFCR_3	—	—	—	—	—	RSTRG[2]	RSTRG[1]	RSTRG[0]	
	RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
FIFO 内置型 串行通信单 元	SCFDR_3	—	—	—	T[4]	T[3]	T[2]	T[1]	T[0]
		—	—	—	R[4]	R[3]	R[2]	R[1]	R[0]
	SCSPTR_3	—	—	—	—	—	—	—	—
		—	—	—	—	SCKIO	SCKDT	SPB2IO	SPB2DT
	SCLSR_3	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	ORER
	SCEMR_3	—	—	—	—	—	—	—	—
		BGDM	—	—	—	—	—	—	ABCS
	SCSMR_4	—	—	—	—	—	—	—	—
		C/A	CHR	PE	O \bar{E}	STOP	—	CKS[1]	CKS[0]
	SCBRR_4								
	SCSCR_4	—	—	—	—	—	—	—	—
		TIE	RIE	TE	RE	REIE	—	CKE[1]	CKE[0]
	SCFTDR_4								
	SCFSR_4	PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]
		ER	TEND	TDFE	BRK	FER	PER	RDF	DR
	SCFRDR_4								
	SCFCR_4	—	—	—	—	—	RSTRG[2]	RSTRG[1]	RSTRG[0]
		RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP
	SCFDR_4	—	—	—	T[4]	T[3]	T[2]	T[1]	T[0]
		—	—	—	R[4]	R[3]	R[2]	R[1]	R[0]
	SCSPTR_4	—	—	—	—	—	—	—	—
		—	—	—	—	SCKIO	SCKDT	SPB2IO	SPB2DT
	SCLSR_4	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	ORER
	SCEMR_4	—	—	—	—	—	—	—	—
		BGDM	—	—	—	—	—	—	ABCS
	SCSMR_5	—	—	—	—	—	—	—	—
		C/A	CHR	PE	O/E	STOP	—	CKS[1]	CKS[0]
	SCBRR_5								
	SCSCR_5	—	—	—	—	—	—	—	—
		TIE	RIE	TE	RE	REIE	—	CKE[1]	CKE[0]
	SCFTDR_5								
	SCFSR_5	PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]
		ER	TEND	TDFE	BRK	FER	PER	RDF	DR
	SCFRDR_5								
	SCFCR_5	—	—	—	—	—	RSTRG[2]	RSTRG[1]	RSTRG[0]
		RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP
	SCFDR_5	—	—	—	T[4]	T[3]	T[2]	T[1]	T[0]
		—	—	—	R[4]	R[3]	R[2]	R[1]	R[0]
	SCSPTR_5	—	—	—	—	—	—	—	—
		RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPB2IO	SPB2DT
SCLSR_5	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	ORER	
SCEMR_5	—	—	—	—	—	—	—	—	
	BGDM	—	—	—	—	—	—	ABCS	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
FIFO 内置型 串行通信单 元	SCSMR_6	—	—	—	—	—	—	—	—
		C/A	CHR	PE	O/E	STOP	—	CKS[1]	CKS[0]
	SCBRR_6								
	SCSCR_6	—	—	—	—	—	—	—	—
		TIE	RIE	TE	RE	REIE	—	CKE[1]	CKE[0]
	SCFTDR_6								
	SCFSR_6	PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]
		ER	TEND	TDFE	BRK	FER	PER	RDF	DR
	SCFRDR_6								
	SCFCR_6	—	—	—	—	—	RSTRG[2]	RSTRG[1]	RSTRG[0]
		RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP
	SCFDR_6	—	—	—	T[4]	T[3]	T[2]	T[1]	T[0]
		—	—	—	R[4]	R[3]	R[2]	R[1]	R[0]
	SCSPTR_6	—	—	—	—	—	—	—	—
		—	—	—	—	SCKIO	SCKDT	SPB2IO	SPB2DT
	SCLSR_6	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	ORER
	SCEMR_6	—	—	—	—	—	—	—	—
		BGDM	—	—	—	—	—	—	ABCS
	SCSMR_7	—	—	—	—	—	—	—	—
		C/A	CHR	PE	O/E	STOP	—	CKS[1]	CKS[0]
	SCBRR_7								
	SCSCR_7	—	—	—	—	—	—	—	—
		TIE	RIE	TE	RE	REIE	—	CKE[1]	CKE[0]
	SCFTDR_7								
	SCFSR_7	PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]
		ER	TEND	TDFE	BRK	FER	PER	RDF	DR
	SCFRDR_7								
	SCFCR_7	—	—	—	—	—	RSTRG[2]	RSTRG[1]	RSTRG[0]
		RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP
	SCFDR_7	—	—	—	T[4]	T[3]	T[2]	T[1]	T[0]
		—	—	—	R[4]	R[3]	R[2]	R[1]	R[0]
SCSPTR_7	—	—	—	—	—	—	—	—	
	RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPB2IO	SPB2DT	
SCLSR_7	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	ORER	
SCEMR_7	—	—	—	—	—	—	—	—	
	BGDM	—	—	—	—	—	—	ABCS	
瑞萨串行 外围接口	SPCR0	SPRIE	SPE	SPTIE	SPEIE	MSTR	MODFEN	—	—
	SSLP0	—	—	—	—	—	—	—	SSL0P
	SPPCR0	—	—	MOIFE	MOIFV	—	—	—	SPLP
	SPSR0	SPRF	TEND	SPTEF	—	—	MODF	—	OVRF
	SPDR0	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24
		SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16
		SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8
SPD7		SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
瑞萨串行 外围接口	SPSCR0	—	—	—	—	—	—	SPSLN1	SPSLN0
	SPSSR0	—	—	—	—	—	—	SPCP1	SPCP0
	SPBR0	SPR7	SPR6	SPR5	SPR4	SPR3	SPR2	SPR1	SPR0
	SPDCR0	TXDMY	SPLW1	SPLW0	—	—	—	—	—
	SPCKD0	—	—	—	—	—	SCKDL2	SCKDL1	SCKDL0
	SSLND0	—	—	—	—	—	SLNDL2	SLNDL1	SLNDL0
	SPND0	—	—	—	—	—	SPNDL2	SPNDL1	SPNDL0
	SPCMD00	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	—	—	—	BRDV1	BRDV0	CPOL	CPHA
	SPCMD01	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	—	—	—	BRDV1	BRDV0	CPOL	CPHA
	SPCMD02	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	—	—	—	BRDV1	BRDV0	CPOL	CPHA
	SPCMD03	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	—	—	—	BRDV1	BRDV0	CPOL	CPHA
	SPBFCR0	TXRST	RXRST	TXTRG[1]	TXTRG[0]	—	RXTRG[2]	RXTRG[1]	RXTRG[0]
	SPBFDR0	—	—	—	—	T[3]	T[2]	T[1]	T[0]
		—	—	R[5]	R[4]	R[3]	R[2]	R[1]	R[0]
	SPCR1	SPRIE	SPE	SPTIE	SPEIE	MSTR	MODFEN	—	—
	SSLP1	—	—	—	—	—	—	—	SSLOP
	SPPCR1	—	—	MOIFE	MOIFV	—	—	—	SPLP
	SPSR1	SPRF	TEND	SPTEF	—	—	MODF	—	OVRF
	SPDR1	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24
		SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16
		SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8
		SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
	SPSCR1	—	—	—	—	—	—	SPSLN1	SPSLN0
	SPSSR1	—	—	—	—	—	—	SPCP1	SPCP0
	SPBR1	SPR7	SPR6	SPR5	SPR4	SPR3	SPR2	SPR1	SPR0
	SPDCR1	TXDMY	SPLW1	SPLW0	—	—	—	—	—
	SPCKD1	—	—	—	—	—	SCKDL2	SCKDL1	SCKDL0
	SSLND1	—	—	—	—	—	SLNDL2	SLNDL1	SLNDL0
	SPND1	—	—	—	—	—	SPNDL2	SPNDL1	SPNDL0
	SPCMD10	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	—	—	—	BRDV1	BRDV0	CPOL	CPHA
	SPCMD11	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	—	—	—	BRDV1	BRDV0	CPOL	CPHA
	SPCMD12	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	—	—	—	BRDV1	BRDV0	CPOL	CPHA
	SPCMD13	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	—	—	—	BRDV1	BRDV0	CPOL	CPHA
	SPBFCR1	TXRST	RXRST	TXTRG[1]	TXTRG[0]	—	RXTRG[2]	RXTRG[1]	RXTRG[0]
	SPBFDR1	—	—	—	—	T[3]	T[2]	T[1]	T[0]
		—	—	R[5]	R[4]	R[3]	R[2]	R[1]	R[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
瑞萨四路 串行外围 接口	SPCR_0	SPRIE	SPE	SPTIE	—	—	—	—	—
	SSLP_0	—	—	—	—	—	—	—	SSLP
	SPPCR_0	—	—	MOIFE	MOIFV	—	IO3FV	IO2FV	SPLP
	SPSR_0	SPRFF	TEND	SPTEF	—	—	—	—	—
	SPDR_0	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24
		SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16
		SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8
		SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
	SPSCR_0	—	—	—	—	—	—	SPSC1	SPSC0
	SPSSR_0	—	—	—	—	—	—	SPSS1	SPSS0
	SPBR_0	SPBR7	SPBR6	SPBR5	SPBR4	SPBR3	SPBR2	SPBR1	SPBR0
	SPDCR_0	TXDMY	—	—	—	—	—	—	—
	SPCKD_0	—	—	—	—	—	SCKDL2	SCKDL1	SCKDL0
	SSLND_0	—	—	—	—	—	SLNDL2	SLNDL1	SLNDL0
	SPND_0	—	—	—	—	—	SPNDL2	SPNDL1	SPNDL0
	SPCMD0_0	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	SPIMOD1	SPIMOD0	SPRW	BRDV1	BRDV0	CPOL	CPHA
	SPCMD1_0	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	SPIMOD1	SPIMOD0	SPRW	BRDV1	BRDV0	CPOL	CPHA
	SPCMD2_0	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	SPIMOD1	SPIMOD0	SPRW	BRDV1	BRDV0	CPOL	CPHA
	SPCMD3_0	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	SPIMOD1	SPIMOD0	SPRW	BRDV1	BRDV0	CPOL	CPHA
	SPBFCR_0	TXRST	RXRST	TXTRG1	TXTRG0	—	RXTRG2	RXTRG1	RXTRG0
	SPBDCR_0	—	—	TXBC5	TXBC4	TXBC3	TXBC2	TXBC1	TXBC0
		—	—	RXBC5	RXBC4	RXBC3	RXBC2	RXBC1	RXBC0
	SPBMUL0_0	SPBMUL[31]	SPBMUL[30]	SPBMUL[29]	SPBMUL[28]	SPBMUL[27]	SPBMUL[26]	SPBMUL[25]	SPBMUL[24]
		SPBMUL[23]	SPBMUL[22]	SPBMUL[21]	SPBMUL[20]	SPBMUL[19]	SPBMUL[18]	SPBMUL[17]	SPBMUL[16]
		SPBMUL[15]	SPBMUL[14]	SPBMUL[13]	SPBMUL[12]	SPBMUL[11]	SPBMUL[10]	SPBMUL[9]	SPBMUL[8]
		SPBMUL[7]	SPBMUL[6]	SPBMUL[5]	SPBMUL[4]	SPBMUL[3]	SPBMUL[2]	SPBMUL[1]	SPBMUL[0]
	SPBMUL1_0	SPBMUL[31]	SPBMUL[30]	SPBMUL[29]	SPBMUL[28]	SPBMUL[27]	SPBMUL[26]	SPBMUL[25]	SPBMUL[24]
		SPBMUL[23]	SPBMUL[22]	SPBMUL[21]	SPBMUL[20]	SPBMUL[19]	SPBMUL[18]	SPBMUL[17]	SPBMUL[16]
		SPBMUL[15]	SPBMUL[14]	SPBMUL[13]	SPBMUL[12]	SPBMUL[11]	SPBMUL[10]	SPBMUL[9]	SPBMUL[8]
		SPBMUL[7]	SPBMUL[6]	SPBMUL[5]	SPBMUL[4]	SPBMUL[3]	SPBMUL[2]	SPBMUL[1]	SPBMUL[0]
	SPBMUL2_0	SPBMUL[31]	SPBMUL[30]	SPBMUL[29]	SPBMUL[28]	SPBMUL[27]	SPBMUL[26]	SPBMUL[25]	SPBMUL[24]
		SPBMUL[23]	SPBMUL[22]	SPBMUL[21]	SPBMUL[20]	SPBMUL[19]	SPBMUL[18]	SPBMUL[17]	SPBMUL[16]
		SPBMUL[15]	SPBMUL[14]	SPBMUL[13]	SPBMUL[12]	SPBMUL[11]	SPBMUL[10]	SPBMUL[9]	SPBMUL[8]
		SPBMUL[7]	SPBMUL[6]	SPBMUL[5]	SPBMUL[4]	SPBMUL[3]	SPBMUL[2]	SPBMUL[1]	SPBMUL[0]
	SPBMUL3_0	SPBMUL[31]	SPBMUL[30]	SPBMUL[29]	SPBMUL[28]	SPBMUL[27]	SPBMUL[26]	SPBMUL[25]	SPBMUL[24]
		SPBMUL[23]	SPBMUL[22]	SPBMUL[21]	SPBMUL[20]	SPBMUL[19]	SPBMUL[18]	SPBMUL[17]	SPBMUL[16]
SPBMUL[15]		SPBMUL[14]	SPBMUL[13]	SPBMUL[12]	SPBMUL[11]	SPBMUL[10]	SPBMUL[9]	SPBMUL[8]	
SPBMUL[7]		SPBMUL[6]	SPBMUL[5]	SPBMUL[4]	SPBMUL[3]	SPBMUL[2]	SPBMUL[1]	SPBMUL[0]	
SPCR_1	SPRIE	SPE	SPTIE	—	—	—	—	—	
SSLP_1	—	—	—	—	—	—	—	SSLP	
SPPCR_1	—	—	MOIFE	MOIFV	—	IO3FV	IO2FV	SPLP	
SPSR_1	SPRFF	TEND	SPTEF	—	—	—	—	—	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
瑞萨四路 串行外围 接口	SPDR_1	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24	
		SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16	
		SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8	
		SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0	
	SPSCR_1	—	—	—	—	—	—	SPSC1	SPSC0	
	SPSSR_1	—	—	—	—	—	—	SPSS1	SPSS0	
	SPBR_1	SPBR7	SPBR6	SPBR5	SPBR4	SPBR3	SPBR2	SPBR1	SPBR0	
	SPDCR_1	TXDMY	—	—	—	—	—	—	—	
	SPCKD_1	—	—	—	—	—	SCKDL2	SCKDL1	SCKDL0	
	SSLND_1	—	—	—	—	—	SLNDL2	SLNDL1	SLNDL0	
	SPND_1	—	—	—	—	—	SPNDL2	SPNDL1	SPNDL0	
	SPCMD0_1	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0	
		SSLKP	SPIMOD1	SPIMOD0	SPRW	BRDV1	BRDV0	CPOL	CPHA	
	SPCMD1_1	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0	
		SSLKP	SPIMOD1	SPIMOD0	SPRW	BRDV1	BRDV0	CPOL	CPHA	
	SPCMD2_1	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0	
		SSLKP	SPIMOD1	SPIMOD0	SPRW	BRDV1	BRDV0	CPOL	CPHA	
	SPCMD3_1	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0	
		SSLKP	SPIMOD1	SPIMOD0	SPRW	BRDV1	BRDV0	CPOL	CPHA	
	SPBFCR_1	TXRST	RXRST	TXTRG1	TXTRG0	—	RXTRG2	RXTRG1	RXTRG0	
	SPBDCR_1	—	—	TXBC5	TXBC4	TXBC3	TXBC2	TXBC1	TXBC0	
		—	—	RXBC5	RXBC4	RXBC3	RXBC2	RXBC1	RXBC0	
	SPBMUL0_1	SPBMUL[31]	SPBMUL[30]	SPBMUL[29]	SPBMUL[28]	SPBMUL[27]	SPBMUL[26]	SPBMUL[25]	SPBMUL[24]	
		SPBMUL[23]	SPBMUL[22]	SPBMUL[21]	SPBMUL[20]	SPBMUL[19]	SPBMUL[18]	SPBMUL[17]	SPBMUL[16]	
		SPBMUL[15]	SPBMUL[14]	SPBMUL[13]	SPBMUL[12]	SPBMUL[11]	SPBMUL[10]	SPBMUL[9]	SPBMUL[8]	
		SPBMUL[7]	SPBMUL[6]	SPBMUL[5]	SPBMUL[4]	SPBMUL[3]	SPBMUL[2]	SPBMUL[1]	SPBMUL[0]	
	SPBMUL1_1	SPBMUL[31]	SPBMUL[30]	SPBMUL[29]	SPBMUL[28]	SPBMUL[27]	SPBMUL[26]	SPBMUL[25]	SPBMUL[24]	
		SPBMUL[23]	SPBMUL[22]	SPBMUL[21]	SPBMUL[20]	SPBMUL[19]	SPBMUL[18]	SPBMUL[17]	SPBMUL[16]	
		SPBMUL[15]	SPBMUL[14]	SPBMUL[13]	SPBMUL[12]	SPBMUL[11]	SPBMUL[10]	SPBMUL[9]	SPBMUL[8]	
		SPBMUL[7]	SPBMUL[6]	SPBMUL[5]	SPBMUL[4]	SPBMUL[3]	SPBMUL[2]	SPBMUL[1]	SPBMUL[0]	
	SPBMUL2_1	SPBMUL[31]	SPBMUL[30]	SPBMUL[29]	SPBMUL[28]	SPBMUL[27]	SPBMUL[26]	SPBMUL[25]	SPBMUL[24]	
		SPBMUL[23]	SPBMUL[22]	SPBMUL[21]	SPBMUL[20]	SPBMUL[19]	SPBMUL[18]	SPBMUL[17]	SPBMUL[16]	
		SPBMUL[15]	SPBMUL[14]	SPBMUL[13]	SPBMUL[12]	SPBMUL[11]	SPBMUL[10]	SPBMUL[9]	SPBMUL[8]	
		SPBMUL[7]	SPBMUL[6]	SPBMUL[5]	SPBMUL[4]	SPBMUL[3]	SPBMUL[2]	SPBMUL[1]	SPBMUL[0]	
	SPBMUL3_1	SPBMUL[31]	SPBMUL[30]	SPBMUL[29]	SPBMUL[28]	SPBMUL[27]	SPBMUL[26]	SPBMUL[25]	SPBMUL[24]	
		SPBMUL[23]	SPBMUL[22]	SPBMUL[21]	SPBMUL[20]	SPBMUL[19]	SPBMUL[18]	SPBMUL[17]	SPBMUL[16]	
		SPBMUL[15]	SPBMUL[14]	SPBMUL[13]	SPBMUL[12]	SPBMUL[11]	SPBMUL[10]	SPBMUL[9]	SPBMUL[8]	
		SPBMUL[7]	SPBMUL[6]	SPBMUL[5]	SPBMUL[4]	SPBMUL[3]	SPBMUL[2]	SPBMUL[1]	SPBMUL[0]	
	SPI 多 I/O 总线控制器	CMNCR	MD	—	—	—	—	—	—	
			MOIO3[1]	MOIO3[0]	MOIO2[1]	MOIO2[0]	MOIO1[1]	MOIO1[0]	MOIO0[1]	MOIO0[0]
			IO3FV[1]	IO3FV[0]	IO2FV[1]	IO2FV[0]	—	—	IO0FV[1]	IO0FV[0]
			—	CPHAT	CPHAR	SSLP	CPOL	—	BSZ[1]	BSZ[0]
		SSLDR	—	—	—	—	—	—	—	—
			—	—	—	—	—	SPNDL[2]	SPNDL[1]	SPNDL[0]
—			—	—	—	—	SLNDL[2]	SLNDL[1]	SLNDL[0]	
—			—	—	—	—	SCKDL[2]	SCKDL[1]	SCKDL[0]	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
SPI 多 I/O 总线控制器	SPBCR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		SPBR[7]	SPBR[6]	SPBR[5]	SPBR[4]	SPBR[3]	SPBR[2]	SPBR[1]	SPBR[0]
		—	—	—	—	—	—	BRDV[1]	BRDV[0]
	DRCR	—	—	—	—	—	—	—	—
		—	—	—	—	RBURST[3]	RBURST[2]	RBURST[1]	RBURST[0]
		—	—	—	—	—	—	RCF	RBE
		—	—	—	—	—	—	—	SSLE
	DRCMR	—	—	—	—	—	—	—	—
		CMD[7]	CMD[6]	CMD[5]	CMD[4]	CMD[3]	CMD[2]	CMD[1]	CMD[0]
		—	—	—	—	—	—	—	—
	DREAR	OCMD[7]	OCMD[6]	OCMD[5]	OCMD[4]	OCMD[3]	OCMD[2]	OCMD[1]	OCMD[0]
		—	—	—	—	—	—	—	—
		EAV[7]	EAV[6]	EAV[5]	EAV[4]	EAV[3]	EAV[2]	EAV[1]	EAV[0]
		—	—	—	—	—	—	—	—
	DROPR	—	—	—	—	—	EAC[2]	EAC[1]	EAC[0]
		OPD3[7]	OPD3[6]	OPD3[5]	OPD3[4]	OPD3[3]	OPD3[2]	OPD3[1]	OPD3[0]
		OPD2[7]	OPD2[6]	OPD2[5]	OPD2[4]	OPD2[3]	OPD2[2]	OPD2[1]	OPD2[0]
		OPD1[7]	OPD1[6]	OPD1[5]	OPD1[4]	OPD1[3]	OPD1[2]	OPD1[1]	OPD1[0]
	DRENDR	OPD0[7]	OPD0[6]	OPD0[5]	OPD0[4]	OPD0[3]	OPD0[2]	OPD0[1]	OPD0[0]
		CDB[1]	CDB[0]	OCDB[1]	OCDB[0]	—	—	ADB[1]	ADB[0]
		—	—	OPDB[1]	OPDB[0]	—	—	DRDB[1]	DRDB[0]
		—	CDE	—	OCDE	ADE[3]	ADE[2]	ADE[1]	ADE[0]
	SMCR	OPDE[3]	OPDE[2]	OPDE[1]	OPDE[0]	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
	SMCMR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		CMD[7]	CMD[6]	CMD[5]	CMD[4]	CMD[3]	CMD[2]	CMD[1]	CMD[0]
		—	—	—	—	—	—	—	—
	SMADR	OCMD[7]	OCMD[6]	OCMD[5]	OCMD[4]	OCMD[3]	OCMD[2]	OCMD[1]	OCMD[0]
		ADR[31]	ADR[30]	ADR[29]	ADR[28]	ADR[27]	ADR[26]	ADR[25]	ADR[24]
		ADR[23]	ADR[22]	ADR[21]	ADR[20]	ADR[19]	ADR[18]	ADR[17]	ADR[16]
		ADR[15]	ADR[14]	ADR[13]	ADR[12]	ADR[11]	ADR[10]	ADR[9]	ADR[8]
	SMOPR	ADR[7]	ADR[6]	ADR[5]	ADR[4]	ADR[3]	ADR[2]	ADR[1]	ADR[0]
		OPD3[7]	OPD3[6]	OPD3[5]	OPD3[4]	OPD3[3]	OPD3[2]	OPD3[1]	OPD3[0]
		OPD2[7]	OPD2[6]	OPD2[5]	OPD2[4]	OPD2[3]	OPD2[2]	OPD2[1]	OPD2[0]
		OPD1[7]	OPD1[6]	OPD1[5]	OPD1[4]	OPD1[3]	OPD1[2]	OPD1[1]	OPD1[0]
	SMENR	OPD0[7]	OPD0[6]	OPD0[5]	OPD0[4]	OPD0[3]	OPD0[2]	OPD0[1]	OPD0[0]
		CDB[1]	CDB[0]	OCDB[1]	OCDB[0]	—	—	ADB[1]	ADB[0]
		—	—	OPDB[1]	OPDB[0]	—	—	SPIDB[1]	SPIDB[0]
		—	CDE	—	OCDE	ADE[3]	ADE[2]	ADE[1]	ADE[0]
		OPDE[3]	OPDE[2]	OPDE[1]	OPDE[0]	SPIDE[3]	SPIDE[2]	SPIDE[1]	SPIDE[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
SPI 多 I/O 总线控制器	SMRDR0	RDATA0[31]	RDATA0[30]	RDATA0[29]	RDATA0[28]	RDATA0[27]	RDATA0[26]	RDATA0[25]	RDATA0[24]	
		RDATA0[23]	RDATA0[22]	RDATA0[21]	RDATA0[20]	RDATA0[19]	RDATA0[18]	RDATA0[17]	RDATA0[16]	
		RDATA0[15]	RDATA0[14]	RDATA0[13]	RDATA0[12]	RDATA0[11]	RDATA0[10]	RDATA0[9]	RDATA0[8]	
		RDATA0[7]	RDATA0[6]	RDATA0[5]	RDATA0[4]	RDATA0[3]	RDATA0[2]	RDATA0[1]	RDATA0[0]	
	SMRDR1	RDATA1[31]	RDATA1[30]	RDATA1[29]	RDATA1[28]	RDATA1[27]	RDATA1[26]	RDATA1[25]	RDATA1[24]	
		RDATA1[23]	RDATA1[22]	RDATA1[21]	RDATA1[20]	RDATA1[19]	RDATA1[18]	RDATA1[17]	RDATA1[16]	
		RDATA1[15]	RDATA1[14]	RDATA1[13]	RDATA1[12]	RDATA1[11]	RDATA1[10]	RDATA1[9]	RDATA1[8]	
		RDATA1[7]	RDATA1[6]	RDATA1[5]	RDATA1[4]	RDATA1[3]	RDATA1[2]	RDATA1[1]	RDATA1[0]	
	SMWDR0	WDATA0[31]	WDATA0[30]	WDATA0[29]	WDATA0[28]	WDATA0[27]	WDATA0[26]	WDATA0[25]	WDATA0[24]	
		WDATA0[23]	WDATA0[22]	WDATA0[21]	WDATA0[20]	WDATA0[19]	WDATA0[18]	WDATA0[17]	WDATA0[16]	
		WDATA0[15]	WDATA0[14]	WDATA0[13]	WDATA0[12]	WDATA0[11]	WDATA0[10]	WDATA0[9]	WDATA0[8]	
		WDATA0[7]	WDATA0[6]	WDATA0[5]	WDATA0[4]	WDATA0[3]	WDATA0[2]	WDATA0[1]	WDATA0[0]	
	SMWDR1	WDATA1[31]	WDATA1[30]	WDATA1[29]	WDATA1[28]	WDATA1[27]	WDATA1[26]	WDATA1[25]	WDATA1[24]	
		WDATA1[23]	WDATA1[22]	WDATA1[21]	WDATA1[20]	WDATA1[19]	WDATA1[18]	WDATA1[17]	WDATA1[16]	
		WDATA1[15]	WDATA1[14]	WDATA1[13]	WDATA1[12]	WDATA1[11]	WDATA1[10]	WDATA1[9]	WDATA1[8]	
		WDATA1[7]	WDATA1[6]	WDATA1[5]	WDATA1[4]	WDATA1[3]	WDATA1[2]	WDATA1[1]	WDATA1[0]	
	CMNSR	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	SSLF	TEND	
	I ² C 总线 接口 3	ICCR1_0	ICE	RCVD	MST	TRS	CKS[3]	CKS[2]	CKS[1]	CKS[0]
		ICCR2_0	BBSY	SCP	SDAO	SDAOP	SCLO	—	IICRST	—
		ICMR_0	MLS	—	—	—	BCWP	BC[2]	BC[1]	BC[0]
		ICIER_0	TIE	TEIE	RIE	NAKIE	STIE	ACKE	ACKBR	ACKBT
		ICSR_0	TDRE	TEND	RDRF	NACKF	STOP	AL/OVE	AAS	ADZ
		SAR_0	SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA[0]	FS
		ICDRT_0								
		ICDRR_0								
NF2CYC_0		—	—	—	—	—	—	PRS	NF2CYC	
ICCR1_1		ICE	RCVD	MST	TRS	CKS[3]	CKS[2]	CKS[1]	CKS[0]	
ICCR2_1		BBSY	SCP	SDAO	SDAOP	SCLO	—	IICRST	—	
ICMR_1		MLS	—	—	—	BCWP	BC[2]	BC[1]	BC[0]	
ICIER_1		TIE	TEIE	RIE	NAKIE	STIE	ACKE	ACKBR	ACKBT	
ICSR_1		TDRE	TEND	RDRF	NACKF	STOP	AL/OVE	AAS	ADZ	
SAR_1		SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA[0]	FS	
ICDRT_1										
ICDRR_1										
NF2CYC_1		—	—	—	—	—	—	PRS	NF2CYC	
ICCR1_2		ICE	RCVD	MST	TRS	CKS[3]	CKS[2]	CKS[1]	CKS[0]	
ICCR2_2		BBSY	SCP	SDAO	SDAOP	SCLO	—	IICRST	—	
ICMR_2		MLS	—	—	—	BCWP	BC[2]	BC[1]	BC[0]	
ICIER_2		TIE	TEIE	RIE	NAKIE	STIE	ACKE	ACKBR	ACKBT	
ICSR_2		TDRE	TEND	RDRF	NACKF	STOP	AL/OVE	AAS	ADZ	
SAR_2		SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA[0]	FS	
ICDRT_2										
ICDRR_2										

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
I ² C 总线 接口 3	NF2CYC_2	—	—	—	—	—	—	PRS	NF2CYC
	ICCR1_3	ICE	RCVD	MST	TRS	CKS[3]	CKS[2]	CKS[1]	CKS[0]
	ICCR2_3	BBSY	SCP	SDAO	SDAOP	SCLO	—	IICRST	—
	ICMR_3	MLS	—	—	—	BCWP	BC[2]	BC[1]	BC[0]
	ICIER_3	TIE	TEIE	RIE	NAKIE	STIE	ACKE	ACKBR	ACKBT
	ICSR_3	TDRE	TEND	RDRF	NACKF	STOP	AL/OVE	AAS	ADZ
	SAR_3	SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA[0]	FS
	ICDRT_3								
	ICDRR_3								
	NF2CYC_3	—	—	—	—	—	—	PRS	NF2CYC
串行音频 接口	SSICR_0	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IEN	—
		CHNL[1]	CHNL[0]	DWL[2]	DWL[1]	DWL[0]	SWL[2]	SWL[1]	SWL[0]
		SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL
		CKDV[3]	CKDV[2]	CKDV[1]	CKDV[0]	MUEN	—	TEN	REN
	SSISR_0	—	—	TUIRQ	TOIRQ	RUIRQ	ROIHQ	IIRQ	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	TCHNO[1]	TCHNO[0]	TSWNO	RCHNO[1]	RCHNO[0]	RSWNO	IDST
	SSIFCR_0	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		TTRG[1]	TTRG[0]	RTRG[1]	RTRG[0]	TIE	RIE	TFRST	RFRST
	SSIFSR_0	—	—	—	—	TDC[3]	TDC[2]	TDC[1]	TDC[0]
		—	—	—	—	—	—	—	TDE
		—	—	—	—	RDC[3]	RDC[2]	RDC[1]	RDC[0]
		—	—	—	—	—	—	—	RDF
	SSIFTDR_0								
	SSIFRDR_0								
	SSITDMR_0	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	CONT
		—	—	—	—	—	—	—	TDM
	SSICR_1	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IEN	—
		CHNL[1]	CHNL[0]	DWL[2]	DWL[1]	DWL[0]	SWL[2]	SWL[1]	SWL[0]
		SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL
		CKDV[3]	CKDV[2]	CKDV[1]	CKDV[0]	MUEN	—	TEN	REN
	SSISR_1	—	—	TUIRQ	TOIRQ	RUIRQ	ROIHQ	IIRQ	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	TCHNO[1]	TCHNO[0]	TSWNO	RCHNO[1]	RCHNO[0]	RSWNO	IDST

模块名	寄存器名称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
串行音频 接口	SSIFCR_1	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		TTRG[1]	TTRG[0]	RTRG[1]	RTRG[0]	TIE	RIE	TFRST	RFRST
	SSIFSR_1	—	—	—	—	TDC[3]	TDC[2]	TDC[1]	TDC[0]
		—	—	—	—	—	—	—	TDE
		—	—	—	—	RDC[3]	RDC[2]	RDC[1]	RDC[0]
		—	—	—	—	—	—	—	RDF
	SSIFTDR_1								
	SSIFRDR_1								
	SSITDMR_1	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	CONT
		—	—	—	—	—	—	—	TDM
	SSICR_2	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IEN	—
		CHNL[1]	CHNL[0]	DWL[2]	DWL[1]	DWL[0]	SWL[2]	SWL[1]	SWL[0]
		SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL
		CKDV[3]	CKDV[2]	CKDV[1]	CKDV[0]	MUEN	—	TEN	REN
	SSISR_2	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	TCHNO[1]	TCHNO[0]	TSWNO	RCHNO[1]	RCHNO[0]	RSWNO	IDST
	SSIFCR_2	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		TTRG[1]	TTRG[0]	RTRG[1]	RTRG[0]	TIE	RIE	TFRST	RFRST
	SSIFSR_2	—	—	—	—	TDC[3]	TDC[2]	TDC[1]	TDC[0]
		—	—	—	—	—	—	—	TDE
		—	—	—	—	RDC[3]	RDC[2]	RDC[1]	RDC[0]
		—	—	—	—	—	—	—	RDF
	SSIFTDR_2								
	SSIFRDR_2								

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
串行音频 接口	SSITDMR_2	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	CONT
		—	—	—	—	—	—	—	TDM
	SSICR_3	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IEN	—
		CHNL[1]	CHNL[0]	DWL[2]	DWL[1]	DWL[0]	SWL[2]	SWL[1]	SWL[0]
		SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL
		CKDV[3]	CKDV[2]	CKDV[1]	CKDV[0]	MUEN	—	TEN	REN
	SSISR_3	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	TCHNO[1]	TCHNO[0]	TSWNO	RCHNO[1]	RCHNO[0]	RSWNO	IDST
	SSIFCR_3	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		TTRG[1]	TTRG[0]	RTRG[1]	RTRG[0]	TIE	RIE	TFRST	RFRST
	SSIFSR_3	—	—	—	—	TDC[3]	TDC[2]	TDC[1]	TDC[0]
		—	—	—	—	—	—	—	TDE
		—	—	—	—	RDC[3]	RDC[2]	RDC[1]	RDC[0]
		—	—	—	—	—	—	—	RDF
	SSIFTDR_3	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
	SSIFRDR_3	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
	SSITDMR_3	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	CONT
		—	—	—	—	—	—	—	TDM
	SSICR_4	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IEN	—
		CHNL[1]	CHNL[0]	DWL[2]	DWL[1]	DWL[0]	SWL[2]	SWL[1]	SWL[0]
		SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL
		CKDV[3]	CKDV[2]	CKDV[1]	CKDV[0]	MUEN	—	TEN	REN
	SSISR_4	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	TCHNO[1]	TCHNO[0]	TSWNO	RCHNO[1]	RCHNO[0]	RSWNO	IDST
	SSIFCR_4	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		TTRG[1]	TTRG[0]	RTRG[1]	RTRG[0]	TIE	RIE	TFRST	RFRST

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
串行音频 接口	SSIFSR_4	—	—	—	—	TDC[3]	TDC[2]	TDC[1]	TDC[0]	
		—	—	—	—	—	—	—	TDE	
		—	—	—	—	RDC[3]	RDC[2]	RDC[1]	RDC[0]	
		—	—	—	—	—	—	—	RDF	
	SSIFTDR_4									
	SSIFRDR_4									
	SSITDMR_4	—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	CONT
		—	—	—	—	—	—	—	—	TDM
	SSICR_5	—	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	—	—
		CHNL[1]	CHNL[0]	DWL[2]	DWL[1]	DWL[0]	SWL[2]	SWL[1]	SWL[0]	—
		SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL	—
		CKDV[3]	CKDV[2]	CKDV[1]	CKDV[0]	MUEN	—	TEN	REN	—
	SSISR_5	—	—	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	—	—
		—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		—	TCHNO[1]	TCHNO[0]	TSWNO	RCHNO[1]	RCHNO[0]	RSWNO	IDST	—
	SSIFCR_5	—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		TTRG[1]	TTRG[0]	RTRG[1]	RTRG[0]	TIE	RIE	TFRST	RFRST	—
	SSIFSR_5	—	—	—	—	—	TDC[3]	TDC[2]	TDC[1]	TDC[0]
		—	—	—	—	—	—	—	—	TDE
		—	—	—	—	—	RDC[3]	RDC[2]	RDC[1]	RDC[0]
		—	—	—	—	—	—	—	—	RDF
	SSIFTDR_5									
	SSIFRDR_5									
	SSITDMR_5	—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	CONT
		—	—	—	—	—	—	—	—	TDM

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
带 FIFO 的 时钟同步 串行 I/O	SIMDR	TRMD[1]	TRMD[0]	SYNCAT	REDG	FL[3]	FL[2]	FL[1]	FL[0]
		TXDIZ	—	SYNCAC	SYNCDL	—	—	—	—
	SISCR	MSEL	—	—	BRPS[4]	BRPS[3]	BRPS[2]	BRPS[1]	BRPS[0]
		—	—	—	—	—	BRDV[2]	BRDV[1]	BRDV[0]
	SITDAR	TDLE	—	—	—	TDLA[3]	TDLA[2]	TDLA[1]	TDLA[0]
		TDRE	TLREP	—	—	TDRA[3]	TDRA[2]	TDRA[1]	TDRA[0]
	SIRDAR	RDLE	—	—	—	RDLA[3]	RDLA[2]	RDLA[1]	RDLA[0]
		RDRE	—	—	—	RDRA[3]	RDRA[2]	RDRA[1]	RDRA[0]
	SICTR	SCKE	FSE	—	—	—	—	TXE	RXE
		—	—	—	—	—	—	TXRST	RXRST
	SIFCTR	TFWM[2]	TFWM[1]	TFWM[0]	TFUA[4]	TFUA[3]	TFUA[2]	TFUA[1]	TFUA[0]
		RFWM[2]	RFWM[1]	RFWM[0]	RFUA[4]	RFUA[3]	RFUA[2]	RFUA[1]	RFUA[0]
	SISTR	—	—	TFEMP	TDREQ	—	—	RFFUL	RDREQ
		—	—	—	FSERR	TFOVF	TFUDF	RFUDF	RFOVF
	SIIER	TDMAE	—	TFEMPE	TDREQE	RDMAE	—	RFFULE	RDREQE
		—	—	—	FSERRE	TFOVFE	TFUDFE	RFUDFE	RFOVFE
	SITDR	SITDL[15]	SITDL[14]	SITDL[13]	SITDL[12]	SITDL[11]	SITDL[10]	SITDL[9]	SITDL[8]
		SITDL[7]	SITDL[6]	SITDL[5]	SITDL[4]	SITDL[3]	SITDL[2]	SITDL[1]	SITDL[0]
		SITDR[15]	SITDR[14]	SITDR[13]	SITDR[12]	SITDR[11]	SITDR[10]	SITDR[9]	SITDR[8]
		SITDR[7]	SITDR[6]	SITDR[5]	SITDR[4]	SITDR[3]	SITDR[2]	SITDR[1]	SITDR[0]
	SIRDR	SIRDL[15]	SIRDL[14]	SIRDL[13]	SIRDL[12]	SIRDL[11]	SIRDL[10]	SIRDL[9]	SIRDL[8]
		SIRDL[7]	SIRDL[6]	SIRDL[5]	SIRDL[4]	SIRDL[3]	SIRDL[2]	SIRDL[1]	SIRDL[0]
		SIRDR[15]	SIRDR[14]	SIRDR[13]	SIRDR[12]	SIRDR[11]	SIRDR[10]	SIRDR[9]	SIRDR[8]
		SIRDR[7]	SIRDR[6]	SIRDR[5]	SIRDR[4]	SIRDR[3]	SIRDR[2]	SIRDR[1]	SIRDR[0]
	MCR_0	MCR15	MCR14	—	—	—	TST[2]	TST[1]	TST[0]
		MCR7	MCR6	MCR5	—	—	MCR2	MCR1	MCR0
	GSR_0	—	—	—	—	—	—	—	—
		—	—	GSR5	GSR4	GSR3	GSR2	GSR1	GSR0
	BCR1_0	TSG1[3]	TSG1[2]	TSG1[1]	TSG1[0]	—	TSG2[2]	TSG2[1]	TSG2[0]
		—	—	SJW[1]	SJW[0]	—	—	—	BSP
	BCR0_0	—	—	—	—	—	—	—	—
		BRP[7]	BRP[6]	BRP[5]	BRP[4]	BRP[3]	BRP[2]	BRP[1]	BRP[0]
IRR_0	IRR15	IRR14	IRR13	IRR12	IRR11	IRR10	IRR9	IRR8	
	IRR7	IRR6	IRR5	IRR4	IRR3	IRR2	IRR1	IRR0	
IMR_0	IMR15	IMR14	IMR13	IMR12	IMR11	IMR10	IMR9	IMR8	
	IMR7	IMR6	IMR5	IMR4	IMR3	IMR2	IMR1	IMR0	
TEC_REC_0	TEC[7]	TEC[6]	TEC[5]	TEC[4]	TEC[3]	TEC[2]	TEC[1]	TEC[0]	
	REC[7]	REC[6]	REC[5]	REC[4]	REC[3]	REC[2]	REC[1]	REC[0]	
TXPR1_0	TXPR1[15]	TXPR1[14]	TXPR1[13]	TXPR1[12]	TXPR1[11]	TXPR1[10]	TXPR1[9]	TXPR1[8]	
	TXPR1[7]	TXPR1[6]	TXPR1[5]	TXPR1[4]	TXPR1[3]	TXPR1[2]	TXPR1[1]	TXPR1[0]	
TXPR0_0	TXPR0[15]	TXPR0[14]	TXPR0[13]	TXPR0[12]	TXPR0[11]	TXPR0[10]	TXPR0[9]	TXPR0[8]	
	TXPR0[7]	TXPR0[6]	TXPR0[5]	TXPR0[4]	TXPR0[3]	TXPR0[2]	TXPR0[1]	—	
TXCR1_0	TXCR1[15]	TXCR1[14]	TXCR1[13]	TXCR1[12]	TXCR1[11]	TXCR1[10]	TXCR1[9]	TXCR1[8]	
	TXCR1[7]	TXCR1[6]	TXCR1[5]	TXCR1[4]	TXCR1[3]	TXCR1[2]	TXCR1[1]	TXCR1[0]	
TXCR0_0	TXCR0[15]	TXCR0[14]	TXCR0[13]	TXCR0[12]	TXCR0[11]	TXCR0[10]	TXCR0[9]	TXCR0[8]	
	TXCR0[7]	TXCR0[6]	TXCR0[5]	TXCR0[4]	TXCR0[3]	TXCR0[2]	TXCR0[1]	—	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
控制器 局域网	TXACK1_0	TXACK1[15]	TXACK1[14]	TXACK1[13]	TXACK1[12]	TXACK1[11]	TXACK1[10]	TXACK1[9]	TXACK1[8]
		TXACK1[7]	TXACK1[6]	TXACK1[5]	TXACK1[4]	TXACK1[3]	TXACK1[2]	TXACK1[1]	TXACK1[0]
	TXACK0_0	TXACK0[15]	TXACK0[14]	TXACK0[13]	TXACK0[12]	TXACK0[11]	TXACK0[10]	TXACK0[9]	TXACK0[8]
		TXACK0[7]	TXACK0[6]	TXACK0[5]	TXACK0[4]	TXACK0[3]	TXACK0[2]	TXACK0[1]	—
	ABACK1_0	ABACK1[15]	ABACK1[14]	ABACK1[13]	ABACK1[12]	ABACK1[11]	ABACK1[10]	ABACK1[9]	ABACK1[8]
		ABACK1[7]	ABACK1[6]	ABACK1[5]	ABACK1[4]	ABACK1[3]	ABACK1[2]	ABACK1[1]	ABACK1[0]
	ABACK0_0	ABACK0[15]	ABACK0[14]	ABACK0[13]	ABACK0[12]	ABACK0[11]	ABACK0[10]	ABACK0[9]	ABACK0[8]
		ABACK0[7]	ABACK0[6]	ABACK0[5]	ABACK0[4]	ABACK0[3]	ABACK0[2]	ABACK0[1]	—
	RXPR1_0	RXPR1[15]	RXPR1[14]	RXPR1[13]	RXPR1[12]	RXPR1[11]	RXPR1[10]	RXPR1[9]	RXPR1[8]
		RXPR1[7]	RXPR1[6]	RXPR1[5]	RXPR1[4]	RXPR1[3]	RXPR1[2]	RXPR1[1]	RXPR1[0]
	RXPR0_0	RXPR0[15]	RXPR0[14]	RXPR0[13]	RXPR0[12]	RXPR0[11]	RXPR0[10]	RXPR0[9]	RXPR0[8]
		RXPR0[7]	RXPR0[6]	RXPR0[5]	RXPR0[4]	RXPR0[3]	RXPR0[2]	RXPR0[1]	RXPR0[0]
	RFPR1_0	RFPR1[15]	RFPR1[14]	RFPR1[13]	RFPR1[12]	RFPR1[11]	RFPR1[10]	RFPR1[9]	RFPR1[8]
		RFPR1[7]	RFPR1[6]	RFPR1[5]	RFPR1[4]	RFPR1[3]	RFPR1[2]	RFPR1[1]	RFPR1[0]
	RFPR0_0	RFPR0[15]	RFPR0[14]	RFPR0[13]	RFPR0[12]	RFPR0[11]	RFPR0[10]	RFPR0[9]	RFPR0[8]
		RFPR0[7]	RFPR0[6]	RFPR0[5]	RFPR0[4]	RFPR0[3]	RFPR0[2]	RFPR0[1]	RFPR0[0]
	MBIMR1_0	MBIMR1[15]	MBIMR1[14]	MBIMR1[13]	MBIMR1[12]	MBIMR1[11]	MBIMR1[10]	MBIMR1[9]	MBIMR1[8]
		MBIMR1[7]	MBIMR1[6]	MBIMR1[5]	MBIMR1[4]	MBIMR1[3]	MBIMR1[2]	MBIMR1[1]	MBIMR1[0]
	MBIMR0_0	MBIMR0[15]	MBIMR0[14]	MBIMR0[13]	MBIMR0[12]	MBIMR0[11]	MBIMR0[10]	MBIMR0[9]	MBIMR0[8]
		MBIMR0[7]	MBIMR0[6]	MBIMR0[5]	MBIMR0[4]	MBIMR0[3]	MBIMR0[2]	MBIMR0[1]	MBIMR0[0]
	UMSR1_0	UMSR1[15]	UMSR1[14]	UMSR1[13]	UMSR1[12]	UMSR1[11]	UMSR1[10]	UMSR1[9]	UMSR1[8]
		UMSR1[7]	UMSR1[6]	UMSR1[5]	UMSR1[4]	UMSR1[3]	UMSR1[2]	UMSR1[1]	UMSR1[0]
	UMSR0_0	UMSR0[15]	UMSR0[14]	UMSR0[13]	UMSR0[12]	UMSR0[11]	UMSR0[10]	UMSR0[9]	UMSR0[8]
		UMSR0[7]	UMSR0[6]	UMSR0[5]	UMSR0[4]	UMSR0[3]	UMSR0[2]	UMSR0[1]	UMSR0[0]
	TCR0_0	TCR15	TCR14	TCR13	TCR12	TCR11	TCR10	—	—
		—	TCR6	TPSC5	TPSC4	TPSC3	TPSC2	TPSC1	TPSC0
	CMAX_TEW_0	—	—	—	—	—	CMAX[2]	CMAX[1]	CMAX[0]
		—	—	—	—	TEW[3]	TEW[2]	TEW[1]	TEW[0]
	RFTROFF_0	RFTROFF[7]	RFTROFF[6]	RFTROFF[5]	RFTROFF[4]	RFTROFF[3]	RFTROFF[2]	RFTROFF[1]	RFTROFF[0]
		—	—	—	—	—	—	—	—
	TSR_0	—	—	—	—	—	—	—	—
		—	—	—	TSR4	TSR3	TSR2	TSR1	TSR0
	CCR_0	—	—	—	—	—	—	—	—
		—	—	CCR[5]	CCR[4]	CCR[3]	CCR[2]	CCR[1]	CCR[0]
	TCNTR_0	TCNTR[15]	TCNTR[14]	TCNTR[13]	TCNTR[12]	TCNTR[11]	TCNTR[10]	TCNTR[9]	TCNTR[8]
		TCNTR[7]	TCNTR[6]	TCNTR[5]	TCNTR[4]	TCNTR[3]	TCNTR[2]	TCNTR[1]	TCNTR[0]
	CYCTR_0	CYCTR[15]	CYCTR[14]	CYCTR[13]	CYCTR[12]	CYCTR[11]	CYCTR[10]	CYCTR[9]	CYCTR[8]
		CYCTR[7]	CYCTR[6]	CYCTR[5]	CYCTR[4]	CYCTR[3]	CYCTR[2]	CYCTR[1]	CYCTR[0]
	RFMK_0	RFMK[15]	RFMK[14]	RFMK[13]	RFMK[12]	RFMK[11]	RFMK[10]	RFMK[9]	RFMK[8]
		RFMK[7]	RFMK[6]	RFMK[5]	RFMK[4]	RFMK[3]	RFMK[2]	RFMK[1]	RFMK[0]
TCMR0_0	TCMR0[15]	TCMR0[14]	TCMR0[13]	TCMR0[12]	TCMR0[11]	TCMR0[10]	TCMR0[9]	TCMR0[8]	
	TCMR0[7]	TCMR0[6]	TCMR0[5]	TCMR0[4]	TCMR0[3]	TCMR0[2]	TCMR0[1]	TCMR0[0]	
TCMR1_0	TCMR1[15]	TCMR1[14]	TCMR1[13]	TCMR1[12]	TCMR1[11]	TCMR1[10]	TCMR1[9]	TCMR1[8]	
	TCMR1[7]	TCMR1[6]	TCMR1[5]	TCMR1[4]	TCMR1[3]	TCMR1[2]	TCMR1[1]	TCMR1[0]	
TCMR2_0	TCMR2[15]	TCMR2[14]	TCMR2[13]	TCMR2[12]	TCMR2[11]	TCMR2[10]	TCMR2[9]	TCMR2[8]	
	TCMR2[7]	TCMR2[6]	TCMR2[5]	TCMR2[4]	TCMR2[3]	TCMR2[2]	TCMR2[1]	TCMR2[0]	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
控制器 局域网	TTTSEL_0	—	TTTSEL[14]	TTTSEL[13]	TTTSEL[12]	TTTSEL[11]	TTTSEL[10]	TTTSEL[9]	TTTSEL[8]
		—	—	—	—	—	—	—	—
	MBn_CONT ROL0_H_0 (n=0 ~ 31)*1	—	STDID[10]	STDID[9]	STDID[8]	STDID[7]	STDID[6]	STDID[5]	STDID[4]
		STDID[3]	STDID[2]	STDID[1]	STDID[0]	RTR	IDE	EXTID[17]	EXTID[16]
	MBn_CONT ROL0_H_0 (n=0 ~ 31)*2	IDE	RTR	—	STDID[10]	STDID[9]	STDID[8]	STDID[7]	STDID[6]
		STDID[5]	STDID[4]	STDID[3]	STDID[2]	STDID[1]	STDID[0]	EXTID[17]	EXTID[16]
	MBn_CONT ROL0_L_0 (n=0 ~ 31)	EXTID[15]	EXTID[14]	EXTID[13]	EXTID[12]	EXTID[11]	EXTID[10]	EXTID[9]	EXTID[8]
		EXTID[7]	EXTID[6]	EXTID[5]	EXTID[4]	EXTID[3]	EXTID[2]	EXTID[1]	EXTID[0]
	MBn_ LAFM0_0 (n=0 ~ 31)*1	—	STDID_ LAFM[10]	STDID_ LAFM[9]	STDID_ LAFM[8]	STDID_ LAFM[7]	STDID_ LAFM[6]	STDID_ LAFM[5]	STDID_ LAFM[4]
		STDID_ LAFM[3]	STDID_ LAFM[2]	STDID_ LAFM[1]	STDID_ LAFM[0]	—	IDE	EXTID_ LAFM[17]	EXTID_ LAFM[16]
	MBn_ LAFM0_0 (n=0 ~ 31)*2	IDE	—	—	STDID_ LAFM[10]	STDID_ LAFM[9]	STDID_ LAFM[8]	STDID_ LAFM[7]	STDID_ LAFM[6]
		STDID_ LAFM[5]	STDID_ LAFM[4]	STDID_ LAFM[3]	STDID_ LAFM[2]	STDID_ LAFM[1]	STDID_ LAFM[0]	EXTID_ LAFM[17]	EXTID_ LAFM[16]
	MBn_ LAFM1_0 (n=0 ~ 31)	EXTID_ LAFM[15]	EXTID_ LAFM[14]	EXTID_ LAFM[13]	EXTID_ LAFM[12]	EXTID_ LAFM[11]	EXTID_ LAFM[10]	EXTID_ LAFM[9]	EXTID_ LAFM[8]
		EXTID_ LAFM[7]	EXTID_ LAFM[6]	EXTID_ LAFM[5]	EXTID_ LAFM[4]	EXTID_ LAFM[3]	EXTID_ LAFM[2]	EXTID_ LAFM[1]	EXTID_ LAFM[0]
	MBn_DATA _01_0 (n=0 ~ 31)	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0
		MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1
	MBn_DATA _23_0 (n=0 ~ 31)	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2
		MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3
	MBn_DATA _45_0 (n=0 ~ 31)	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4
		MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5
	MBn_DATA _67_0 (n=0 ~ 31)	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6
		MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7
	MBn_CONT ROL1_0 (n=0)	—	—	NMC	—	—	MBC[2]	MBC[1]	MBC[0]
		—	—	—	—	DLC[3]	DLC[2]	DLC[1]	DLC[0]
	MBn_CONT ROL1_0 (n=1 ~ 31)	—	—	NMC	ATX	DART	MBC[2]	MBC[1]	MBC[0]
		—	—	—	—	DLC[3]	DLC[2]	DLC[1]	DLC[0]
	MBn_TIME STAMP_0 (n=0 ~ 15、 30、31)	TS15	TS14	TS13	TS12	TS11	TS10	TS9	TS8
		TS7	TS6	TS5	TS4	TS3	TS2	TS1	TS0
	MBn_TTT_0 (n=24 ~ 30)	TTT15	TTT14	TTT13	TTT12	TTT11	TTT10	TTT9	TTT8
		TTT7	TTT6	TTT5	TTT4	TTT3	TTT2	TTT1	TTT0
	MBn_TTCO NTROL_0 (n=24 ~ 29)	TTW[1]	TTW[0]	OFFSET[5]	OFFSET[4]	OFFSET[3]	OFFSET[2]	OFFSET[1]	OFFSET[0]
		—	—	—	—	—	REP_ FACTOR[2]	REP_ FACTOR[1]	REP_ FACTOR[0]
	MCR_1	MCR15	MCR14	—	—	—	TST[2]	TST[1]	TST[0]
	MCR7	MCR6	MCR5	—	—	MCR2	MCR1	MCR0	
GSR_1	—	—	—	—	—	—	—	—	
	—	—	GSR5	GSR4	GSR3	GSR2	GSR1	GSR0	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
控制器 局域网	BCR1_1	TSG1[3]	TSG1[2]	TSG1[1]	TSG1[0]	—	TSG2[2]	TSG2[1]	TSG2[0]
		—	—	SJW[1]	SJW[0]	—	—	—	BSP
	BCR0_1	—	—	—	—	—	—	—	—
		BRP[7]	BRP[6]	BRP[5]	BRP[4]	BRP[3]	BRP[2]	BRP[1]	BRP[0]
	IRR_1	IRR15	IRR14	IRR13	IRR12	IRR11	IRR10	IRR9	IRR8
		IRR7	IRR6	IRR5	IRR4	IRR3	IRR2	IRR1	IRR0
	IMR_1	IMR15	IMR14	IMR13	IMR12	IMR11	IMR10	IMR9	IMR8
		IMR7	IMR6	IMR5	IMR4	IMR3	IMR2	IMR1	IMR0
	TEC_REC_1	TEC[7]	TEC[6]	TEC[5]	TEC[4]	TEC[3]	TEC[2]	TEC[1]	TEC[0]
		REC[7]	REC[6]	REC[5]	REC[4]	REC[3]	REC[2]	REC[1]	REC[0]
	TXPR1_1	TXPR1[15]	TXPR1[14]	TXPR1[13]	TXPR1[12]	TXPR1[11]	TXPR1[10]	TXPR1[9]	TXPR1[8]
		TXPR1[7]	TXPR1[6]	TXPR1[5]	TXPR1[4]	TXPR1[3]	TXPR1[2]	TXPR1[1]	TXPR1[0]
	TXPR0_1	TXPR0[15]	TXPR0[14]	TXPR0[13]	TXPR0[12]	TXPR0[11]	TXPR0[10]	TXPR0[9]	TXPR0[8]
		TXPR0[7]	TXPR0[6]	TXPR0[5]	TXPR0[4]	TXPR0[3]	TXPR0[2]	TXPR0[1]	—
	TXCR1_1	TXCR1[15]	TXCR1[14]	TXCR1[13]	TXCR1[12]	TXCR1[11]	TXCR1[10]	TXCR1[9]	TXCR1[8]
		TXCR1[7]	TXCR1[6]	TXCR1[5]	TXCR1[4]	TXCR1[3]	TXCR1[2]	TXCR1[1]	TXCR1[0]
	TXCR0_1	TXCR0[15]	TXCR0[14]	TXCR0[13]	TXCR0[12]	TXCR0[11]	TXCR0[10]	TXCR0[9]	TXCR0[8]
		TXCR0[7]	TXCR0[6]	TXCR0[5]	TXCR0[4]	TXCR0[3]	TXCR0[2]	TXCR0[1]	—
	TXACK1_1	TXACK1[15]	TXACK1[14]	TXACK1[13]	TXACK1[12]	TXACK1[11]	TXACK1[10]	TXACK1[9]	TXACK1[8]
		TXACK1[7]	TXACK1[6]	TXACK1[5]	TXACK1[4]	TXACK1[3]	TXACK1[2]	TXACK1[1]	TXACK1[0]
	TXACK0_1	TXACK0[15]	TXACK0[14]	TXACK0[13]	TXACK0[12]	TXACK0[11]	TXACK0[10]	TXACK0[9]	TXACK0[8]
		TXACK0[7]	TXACK0[6]	TXACK0[5]	TXACK0[4]	TXACK0[3]	TXACK0[2]	TXACK0[1]	—
	ABACK1_1	ABACK1[15]	ABACK1[14]	ABACK1[13]	ABACK1[12]	ABACK1[11]	ABACK1[10]	ABACK1[9]	ABACK1[8]
		ABACK1[7]	ABACK1[6]	ABACK1[5]	ABACK1[4]	ABACK1[3]	ABACK1[2]	ABACK1[1]	ABACK1[0]
	ABACK0_1	ABACK0[15]	ABACK0[14]	ABACK0[13]	ABACK0[12]	ABACK0[11]	ABACK0[10]	ABACK0[9]	ABACK0[8]
		ABACK0[7]	ABACK0[6]	ABACK0[5]	ABACK0[4]	ABACK0[3]	ABACK0[2]	ABACK0[1]	—
	RXPR1_1	RXPR1[15]	RXPR1[14]	RXPR1[13]	RXPR1[12]	RXPR1[11]	RXPR1[10]	RXPR1[9]	RXPR1[8]
		RXPR1[7]	RXPR1[6]	RXPR1[5]	RXPR1[4]	RXPR1[3]	RXPR1[2]	RXPR1[1]	RXPR1[0]
	RXPR0_1	RXPR0[15]	RXPR0[14]	RXPR0[13]	RXPR0[12]	RXPR0[11]	RXPR0[10]	RXPR0[9]	RXPR0[8]
		RXPR0[7]	RXPR0[6]	RXPR0[5]	RXPR0[4]	RXPR0[3]	RXPR0[2]	RXPR0[1]	RXPR0[0]
	RFPR1_1	RFPR1[15]	RFPR1[14]	RFPR1[13]	RFPR1[12]	RFPR1[11]	RFPR1[10]	RFPR1[9]	RFPR1[8]
		RFPR1[7]	RFPR1[6]	RFPR1[5]	RFPR1[4]	RFPR1[3]	RFPR1[2]	RFPR1[1]	RFPR1[0]
	RFPR0_1	RFPR0[15]	RFPR0[14]	RFPR0[13]	RFPR0[12]	RFPR0[11]	RFPR0[10]	RFPR0[9]	RFPR0[8]
		RFPR0[7]	RFPR0[6]	RFPR0[5]	RFPR0[4]	RFPR0[3]	RFPR0[2]	RFPR0[1]	RFPR0[0]
	MBIMR1_1	MBIMR1[15]	MBIMR1[14]	MBIMR1[13]	MBIMR1[12]	MBIMR1[11]	MBIMR1[10]	MBIMR1[9]	MBIMR1[8]
		MBIMR1[7]	MBIMR1[6]	MBIMR1[5]	MBIMR1[4]	MBIMR1[3]	MBIMR1[2]	MBIMR1[1]	MBIMR1[0]
	MBIMR0_1	MBIMR0[15]	MBIMR0[14]	MBIMR0[13]	MBIMR0[12]	MBIMR0[11]	MBIMR0[10]	MBIMR0[9]	MBIMR0[8]
		MBIMR0[7]	MBIMR0[6]	MBIMR0[5]	MBIMR0[4]	MBIMR0[3]	MBIMR0[2]	MBIMR0[1]	MBIMR0[0]
	UMSR1_1	UMSR1[15]	UMSR1[14]	UMSR1[13]	UMSR1[12]	UMSR1[11]	UMSR1[10]	UMSR1[9]	UMSR1[8]
		UMSR1[7]	UMSR1[6]	UMSR1[5]	UMSR1[4]	UMSR1[3]	UMSR1[2]	UMSR1[1]	UMSR1[0]
	UMSR0_1	UMSR0[15]	UMSR0[14]	UMSR0[13]	UMSR0[12]	UMSR0[11]	UMSR0[10]	UMSR0[9]	UMSR0[8]
		UMSR0[7]	UMSR0[6]	UMSR0[5]	UMSR0[4]	UMSR0[3]	UMSR0[2]	UMSR0[1]	UMSR0[0]
TTCR0_1	TCR15	TCR14	TCR13	TCR12	TCR11	TCR10	—	—	
	—	TCR6	TPSC5	TPSC4	TPSC3	TPSC2	TPSC1	TPSC0	
CMAX_ TEW_1	—	—	—	—	—	CMAX[2]	CMAX[1]	CMAX[0]	
	—	—	—	—	TEW[3]	TEW[2]	TEW[1]	TEW[0]	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
控制器 局域网	RFTROFF_1	RFTROFF[7]	RFTROFF[6]	RFTROFF[5]	RFTROFF[4]	RFTROFF[3]	RFTROFF[2]	RFTROFF[1]	RFTROFF[0]
		—	—	—	—	—	—	—	—
	TSR_1	—	—	—	—	—	—	—	—
		—	—	—	TSR4	TSR3	TSR2	TSR1	TSR0
	CCR_1	—	—	—	—	—	—	—	—
		—	—	CCR[5]	CCR[4]	CCR[3]	CCR[2]	CCR[1]	CCR[0]
	TCNTR_1	TCNTR[15]	TCNTR[14]	TCNTR[13]	TCNTR[12]	TCNTR[11]	TCNTR[10]	TCNTR[9]	TCNTR[8]
		TCNTR[7]	TCNTR[6]	TCNTR[5]	TCNTR[4]	TCNTR[3]	TCNTR[2]	TCNTR[1]	TCNTR[0]
	CYCTR_1	CYCTR[15]	CYCTR[14]	CYCTR[13]	CYCTR[12]	CYCTR[11]	CYCTR[10]	CYCTR[9]	CYCTR[8]
		CYCTR[7]	CYCTR[6]	CYCTR[5]	CYCTR[4]	CYCTR[3]	CYCTR[2]	CYCTR[1]	CYCTR[0]
	RFMK_1	RFMK[15]	RFMK[14]	RFMK[13]	RFMK[12]	RFMK[11]	RFMK[10]	RFMK[9]	RFMK[8]
		RFMK[7]	RFMK[6]	RFMK[5]	RFMK[4]	RFMK[3]	RFMK[2]	RFMK[1]	RFMK[0]
	TCMR0_1	TCMR0[15]	TCMR0[14]	TCMR0[13]	TCMR0[12]	TCMR0[11]	TCMR0[10]	TCMR0[9]	TCMR0[8]
		TCMR0[7]	TCMR0[6]	TCMR0[5]	TCMR0[4]	TCMR0[3]	TCMR0[2]	TCMR0[1]	TCMR0[0]
	TCMR1_1	TCMR1[15]	TCMR1[14]	TCMR1[13]	TCMR1[12]	TCMR1[11]	TCMR1[10]	TCMR1[9]	TCMR1[8]
		TCMR1[7]	TCMR1[6]	TCMR1[5]	TCMR1[4]	TCMR1[3]	TCMR1[2]	TCMR1[1]	TCMR1[0]
	TCMR2_1	TCMR2[15]	TCMR2[14]	TCMR2[13]	TCMR2[12]	TCMR2[11]	TCMR2[10]	TCMR2[9]	TCMR2[8]
		TCMR2[7]	TCMR2[6]	TCMR2[5]	TCMR2[4]	TCMR2[3]	TCMR2[2]	TCMR2[1]	TCMR2[0]
	TTTSEL_1	—	TTTSEL[14]	TTTSEL[13]	TTTSEL[12]	TTTSEL[11]	TTTSEL[10]	TTTSEL[9]	TTTSEL[8]
		—	—	—	—	—	—	—	—
	MBn_CONT ROLO_H_1 (n=0 ~ 31)*1	—	STDID[10]	STDID[9]	STDID[8]	STDID[7]	STDID[6]	STDID[5]	STDID[4]
		STDID[3]	STDID[2]	STDID[1]	STDID[0]	RTR	IDE	EXTID[17]	EXTID[16]
	MBn_CONT ROLO_H_1 (n=0 ~ 31)*2	IDE	RTR	—	STDID[10]	STDID[9]	STDID[8]	STDID[7]	STDID[6]
		STDID[5]	STDID[4]	STDID[3]	STDID[2]	STDID[1]	STDID[0]	EXTID[17]	EXTID[16]
	MBn_CONT ROLO_L_1 (n=0 ~ 31)	EXTID[15]	EXTID[14]	EXTID[13]	EXTID[12]	EXTID[11]	EXTID[10]	EXTID[9]	EXTID[8]
		EXTID[7]	EXTID[6]	EXTID[5]	EXTID[4]	EXTID[3]	EXTID[2]	EXTID[1]	EXTID[0]
	MBn_ LAFM0_1 (n=0 ~ 31)*1	—	STDID_ LAFM[10]	STDID_ LAFM[9]	STDID_ LAFM[8]	STDID_ LAFM[7]	STDID_ LAFM[6]	STDID_ LAFM[5]	STDID_ LAFM[4]
		STDID_ LAFM[3]	STDID_ LAFM[2]	STDID_ LAFM[1]	STDID_ LAFM[0]	—	IDE	EXTID_ LAFM[17]	EXTID_ LAFM[16]
	MBn_ LAFM0_1 (n=0 ~ 31)*2	IDE	—	—	STDID_ LAFM[10]	STDID_ LAFM[9]	STDID_ LAFM[8]	STDID_ LAFM[7]	STDID_ LAFM[6]
		STDID_ LAFM[5]	STDID_ LAFM[4]	STDID_ LAFM[3]	STDID_ LAFM[2]	STDID_ LAFM[1]	STDID_ LAFM[0]	EXTID_ LAFM[17]	EXTID_ LAFM[16]
	MBn_ LAFM1_1 (n=0 ~ 31)	EXTID_ LAFM[15]	EXTID_ LAFM[14]	EXTID_ LAFM[13]	EXTID_ LAFM[12]	EXTID_ LAFM[11]	EXTID_ LAFM[10]	EXTID_ LAFM[9]	EXTID_ LAFM[8]
		EXTID_ LAFM[7]	EXTID_ LAFM[6]	EXTID_ LAFM[5]	EXTID_ LAFM[4]	EXTID_ LAFM[3]	EXTID_ LAFM[2]	EXTID_ LAFM[1]	EXTID_ LAFM[0]
MBn_DATA _01_1 (n=0 ~ 31)	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	
	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	
MBn_DATA _23_1 (n=0 ~ 31)	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	
	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	
MBn_DATA _45_1 (n=0 ~ 31)	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	
	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	
MBn_DATA _67_1 (n=0 ~ 31)	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	
	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
控制器 局域网	MBn_CONT ROL1_1 (n=0)	—	—	NMC	—	—	MBC[2]	MBC[1]	MBC[0]
		—	—	—	—	DLC[3]	DLC[2]	DLC[1]	DLC[0]
	MBn_CONT ROL1_1 (n=1 ~ 31)	—	—	NMC	ATX	DART	MBC[2]	MBC[1]	MBC[0]
		—	—	—	—	DLC[3]	DLC[2]	DLC[1]	DLC[0]
	MBn_TIME STAMP_1 (n=0 ~ 15、 30、31)	TS15	TS14	TS13	TS12	TS11	TS10	TS9	TS8
		TS7	TS6	TS5	TS4	TS3	TS2	TS1	TS0
	MBn_TTT_1 (n=24 ~ 30)	TTT15	TTT14	TTT13	TTT12	TTT11	TTT10	TTT9	TTT8
		TTT7	TTT6	TTT5	TTT4	TTT3	TTT2	TTT1	TTT0
	MBn_TTCO NTROL_1 (n=24 ~ 29)	TTW[1]	TTW[0]	OFFSET[5]	OFFSET[4]	OFFSET[3]	OFFSET[2]	OFFSET[1]	OFFSET[0]
		—	—	—	—	—	REP_ FACTOR[2]	REP_ FACTOR[1]	REP_ FACTOR[0]
	MCR_2	MCR15	MCR14	—	—	—	TST[2]	TST[1]	TST[0]
		MCR7	MCR6	MCR5	—	—	MCR2	MCR1	MCR0
	GSR_2	—	—	—	—	—	—	—	—
		—	—	GSR5	GSR4	GSR3	GSR2	GSR1	GSR0
	BCR1_2	TSG1[3]	TSG1[2]	TSG1[1]	TSG1[0]	—	TSG2[2]	TSG2[1]	TSG2[0]
		—	—	SJW[1]	SJW[0]	—	—	—	BSP
	BCR0_2	—	—	—	—	—	—	—	—
		BRP[7]	BRP[6]	BRP[5]	BRP[4]	BRP[3]	BRP[2]	BRP[1]	BRP[0]
	IRR_2	IRR15	IRR14	IRR13	IRR12	IRR11	IRR10	IRR9	IRR8
		IRR7	IRR6	IRR5	IRR4	IRR3	IRR2	IRR1	IRR0
	IMR_2	IMR15	IMR14	IMR13	IMR12	IMR11	IMR10	IMR9	IMR8
		IMR7	IMR6	IMR5	IMR4	IMR3	IMR2	IMR1	IMR0
	TEC_REC_2	TEC[7]	TEC[6]	TEC[5]	TEC[4]	TEC[3]	TEC[2]	TEC[1]	TEC[0]
		REC[7]	REC[6]	REC[5]	REC[4]	REC[3]	REC[2]	REC[1]	REC[0]
	TXPR1_2	TXPR1[15]	TXPR1[14]	TXPR1[13]	TXPR1[12]	TXPR1[11]	TXPR1[10]	TXPR1[9]	TXPR1[8]
		TXPR1[7]	TXPR1[6]	TXPR1[5]	TXPR1[4]	TXPR1[3]	TXPR1[2]	TXPR1[1]	TXPR1[0]
	TXPR0_2	TXPR0[15]	TXPR0[14]	TXPR0[13]	TXPR0[12]	TXPR0[11]	TXPR0[10]	TXPR0[9]	TXPR0[8]
		TXPR0[7]	TXPR0[6]	TXPR0[5]	TXPR0[4]	TXPR0[3]	TXPR0[2]	TXPR0[1]	—
	TXCR1_2	TXCR1[15]	TXCR1[14]	TXCR1[13]	TXCR1[12]	TXCR1[11]	TXCR1[10]	TXCR1[9]	TXCR1[8]
		TXCR1[7]	TXCR1[6]	TXCR1[5]	TXCR1[4]	TXCR1[3]	TXCR1[2]	TXCR1[1]	TXCR1[0]
	TXCR0_2	TXCR0[15]	TXCR0[14]	TXCR0[13]	TXCR0[12]	TXCR0[11]	TXCR0[10]	TXCR0[9]	TXCR0[8]
		TXCR0[7]	TXCR0[6]	TXCR0[5]	TXCR0[4]	TXCR0[3]	TXCR0[2]	TXCR0[1]	—
	TXACK1_2	TXACK1[15]	TXACK1[14]	TXACK1[13]	TXACK1[12]	TXACK1[11]	TXACK1[10]	TXACK1[9]	TXACK1[8]
		TXACK1[7]	TXACK1[6]	TXACK1[5]	TXACK1[4]	TXACK1[3]	TXACK1[2]	TXACK1[1]	TXACK1[0]
	TXACK0_2	TXACK0[15]	TXACK0[14]	TXACK0[13]	TXACK0[12]	TXACK0[11]	TXACK0[10]	TXACK0[9]	TXACK0[8]
		TXACK0[7]	TXACK0[6]	TXACK0[5]	TXACK0[4]	TXACK0[3]	TXACK0[2]	TXACK0[1]	—
	ABACK1_2	ABACK1[15]	ABACK1[14]	ABACK1[13]	ABACK1[12]	ABACK1[11]	ABACK1[10]	ABACK1[9]	ABACK1[8]
		ABACK1[7]	ABACK1[6]	ABACK1[5]	ABACK1[4]	ABACK1[3]	ABACK1[2]	ABACK1[1]	ABACK1[0]
	ABACK0_2	ABACK0[15]	ABACK0[14]	ABACK0[13]	ABACK0[12]	ABACK0[11]	ABACK0[10]	ABACK0[9]	ABACK0[8]
		ABACK0[7]	ABACK0[6]	ABACK0[5]	ABACK0[4]	ABACK0[3]	ABACK0[2]	ABACK0[1]	—
RXPR1_2	RXPR1[15]	RXPR1[14]	RXPR1[13]	RXPR1[12]	RXPR1[11]	RXPR1[10]	RXPR1[9]	RXPR1[8]	
	RXPR1[7]	RXPR1[6]	RXPR1[5]	RXPR1[4]	RXPR1[3]	RXPR1[2]	RXPR1[1]	RXPR1[0]	
RXPR0_2	RXPR0[15]	RXPR0[14]	RXPR0[13]	RXPR0[12]	RXPR0[11]	RXPR0[10]	RXPR0[9]	RXPR0[8]	
	RXPR0[7]	RXPR0[6]	RXPR0[5]	RXPR0[4]	RXPR0[3]	RXPR0[2]	RXPR0[1]	RXPR0[0]	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
控制器 局域网	RFPR1_2	RFPR1[15]	RFPR1[14]	RFPR1[13]	RFPR1[12]	RFPR1[11]	RFPR1[10]	RFPR1[9]	RFPR1[8]
		RFPR1[7]	RFPR1[6]	RFPR1[5]	RFPR1[4]	RFPR1[3]	RFPR1[2]	RFPR1[1]	RFPR1[0]
	RFPR0_2	RFPR0[15]	RFPR0[14]	RFPR0[13]	RFPR0[12]	RFPR0[11]	RFPR0[10]	RFPR0[9]	RFPR0[8]
		RFPR0[7]	RFPR0[6]	RFPR0[5]	RFPR0[4]	RFPR0[3]	RFPR0[2]	RFPR0[1]	RFPR0[0]
	MBIMR1_2	MBIMR1[15]	MBIMR1[14]	MBIMR1[13]	MBIMR1[12]	MBIMR1[11]	MBIMR1[10]	MBIMR1[9]	MBIMR1[8]
		MBIMR1[7]	MBIMR1[6]	MBIMR1[5]	MBIMR1[4]	MBIMR1[3]	MBIMR1[2]	MBIMR1[1]	MBIMR1[0]
	MBIMR0_2	MBIMR0[15]	MBIMR0[14]	MBIMR0[13]	MBIMR0[12]	MBIMR0[11]	MBIMR0[10]	MBIMR0[9]	MBIMR0[8]
		MBIMR0[7]	MBIMR0[6]	MBIMR0[5]	MBIMR0[4]	MBIMR0[3]	MBIMR0[2]	MBIMR0[1]	MBIMR0[0]
	UMSR1_2	UMSR1[15]	UMSR1[14]	UMSR1[13]	UMSR1[12]	UMSR1[11]	UMSR1[10]	UMSR1[9]	UMSR1[8]
		UMSR1[7]	UMSR1[6]	UMSR1[5]	UMSR1[4]	UMSR1[3]	UMSR1[2]	UMSR1[1]	UMSR1[0]
	UMSR0_2	UMSR0[15]	UMSR0[14]	UMSR0[13]	UMSR0[12]	UMSR0[11]	UMSR0[10]	UMSR0[9]	UMSR0[8]
		UMSR0[7]	UMSR0[6]	UMSR0[5]	UMSR0[4]	UMSR0[3]	UMSR0[2]	UMSR0[1]	UMSR0[0]
	TTCR0_2	TCR15	TCR14	TCR13	TCR12	TCR11	TCR10	—	—
		—	TCR6	TPSC5	TPSC4	TPSC3	TPSC2	TPSC1	TPSC0
	CMAX_TEW_2	—	—	—	—	—	CMAX[2]	CMAX[1]	CMAX[0]
		—	—	—	—	TEW[3]	TEW[2]	TEW[1]	TEW[0]
	RFTROFF_2	RFTROFF[7]	RFTROFF[6]	RFTROFF[5]	RFTROFF[4]	RFTROFF[3]	RFTROFF[2]	RFTROFF[1]	RFTROFF[0]
		—	—	—	—	—	—	—	—
	TSR_2	—	—	—	—	—	—	—	—
		—	—	—	TSR4	TSR3	TSR2	TSR1	TSR0
	CCR_2	—	—	—	—	—	—	—	—
		—	—	CCR[5]	CCR[4]	CCR[3]	CCR[2]	CCR[1]	CCR[0]
	TCNTR_2	TCNTR[15]	TCNTR[14]	TCNTR[13]	TCNTR[12]	TCNTR[11]	TCNTR[10]	TCNTR[9]	TCNTR[8]
		TCNTR[7]	TCNTR[6]	TCNTR[5]	TCNTR[4]	TCNTR[3]	TCNTR[2]	TCNTR[1]	TCNTR[0]
	CYCTR_2	CYCTR[15]	CYCTR[14]	CYCTR[13]	CYCTR[12]	CYCTR[11]	CYCTR[10]	CYCTR[9]	CYCTR[8]
		CYCTR[7]	CYCTR[6]	CYCTR[5]	CYCTR[4]	CYCTR[3]	CYCTR[2]	CYCTR[1]	CYCTR[0]
	RFMK_2	RFMK[15]	RFMK[14]	RFMK[13]	RFMK[12]	RFMK[11]	RFMK[10]	RFMK[9]	RFMK[8]
		RFMK[7]	RFMK[6]	RFMK[5]	RFMK[4]	RFMK[3]	RFMK[2]	RFMK[1]	RFMK[0]
	TCMR0_2	TCMR0[15]	TCMR0[14]	TCMR0[13]	TCMR0[12]	TCMR0[11]	TCMR0[10]	TCMR0[9]	TCMR0[8]
		TCMR0[7]	TCMR0[6]	TCMR0[5]	TCMR0[4]	TCMR0[3]	TCMR0[2]	TCMR0[1]	TCMR0[0]
	TCMR1_2	TCMR1[15]	TCMR1[14]	TCMR1[13]	TCMR1[12]	TCMR1[11]	TCMR1[10]	TCMR1[9]	TCMR1[8]
		TCMR1[7]	TCMR1[6]	TCMR1[5]	TCMR1[4]	TCMR1[3]	TCMR1[2]	TCMR1[1]	TCMR1[0]
TCMR2_2	TCMR2[15]	TCMR2[14]	TCMR2[13]	TCMR2[12]	TCMR2[11]	TCMR2[10]	TCMR2[9]	TCMR2[8]	
	TCMR2[7]	TCMR2[6]	TCMR2[5]	TCMR2[4]	TCMR2[3]	TCMR2[2]	TCMR2[1]	TCMR2[0]	
TTTSEL_2	—	TTTSEL[14]	TTTSEL[13]	TTTSEL[12]	TTTSEL[11]	TTTSEL[10]	TTTSEL[9]	TTTSEL[8]	
	—	—	—	—	—	—	—	—	
MBn_CONT ROLO_H_2 (n=0 ~ 31)*1	—	STDID[10]	STDID[9]	STDID[8]	STDID[7]	STDID[6]	STDID[5]	STDID[4]	
	STDID[3]	STDID[2]	STDID[1]	STDID[0]	RTR	IDE	EXTID[17]	EXTID[16]	
MBn_CONT ROLO_H_2 (n=0 ~ 31)*2	IDE	RTR	—	STDID[10]	STDID[9]	STDID[8]	STDID[7]	STDID[6]	
	STDID[5]	STDID[4]	STDID[3]	STDID[2]	STDID[1]	STDID[0]	EXTID[17]	EXTID[16]	
MBn_CONT ROLO_L_2 (n=0 ~ 31)	EXTID[15]	EXTID[14]	EXTID[13]	EXTID[12]	EXTID[11]	EXTID[10]	EXTID[9]	EXTID[8]	
	EXTID[7]	EXTID[6]	EXTID[5]	EXTID[4]	EXTID[3]	EXTID[2]	EXTID[1]	EXTID[0]	
MBn_ LAFM0_2 (n=0 ~ 31)*1	—	STDID_ LAFM[10]	STDID_ LAFM[9]	STDID_ LAFM[8]	STDID_ LAFM[7]	STDID_ LAFM[6]	STDID_ LAFM[5]	STDID_ LAFM[4]	
	STDID_ LAFM[3]	STDID_ LAFM[2]	STDID_ LAFM[1]	STDID_ LAFM[0]	—	IDE	EXTID_ LAFM[17]	EXTID_ LAFM[16]	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
控制器 局域网	MBn_ LAFM0_2 (n=0 ~ 31)*2	IDE	—	—	STDID_ LAFM[10]	STDID_ LAFM[9]	STDID_ LAFM[8]	STDID_ LAFM[7]	STDID_ LAFM[6]
		STDID_ LAFM[5]	STDID_ LAFM[4]	STDID_ LAFM[3]	STDID_ LAFM[2]	STDID_ LAFM[1]	STDID_ LAFM[0]	EXTID_ LAFM[17]	EXTID_ LAFM[16]
	MBn_ LAFM1_2 (n=0 ~ 31)	EXTID_ LAFM[15]	EXTID_ LAFM[14]	EXTID_ LAFM[13]	EXTID_ LAFM[12]	EXTID_ LAFM[11]	EXTID_ LAFM[10]	EXTID_ LAFM[9]	EXTID_ LAFM[8]
		EXTID_ LAFM[7]	EXTID_ LAFM[6]	EXTID_ LAFM[5]	EXTID_ LAFM[4]	EXTID_ LAFM[3]	EXTID_ LAFM[2]	EXTID_ LAFM[1]	EXTID_ LAFM[0]
	MBn_DATA _01_2 (n=0 ~ 31)	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0	MSG_DATA0
		MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1	MSG_DATA1
	MBn_DATA _23_2 (n=0 ~ 31)	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2	MSG_DATA2
		MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3	MSG_DATA3
	MBn_DATA _45_2 (n=0 ~ 31)	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4	MSG_DATA4
		MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5	MSG_DATA5
	MBn_DATA _67_2 (n=0 ~ 31)	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6	MSG_DATA6
		MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7	MSG_DATA7
	MBn_CONT ROL1_2 (n=0)	—	—	NMC	—	—	MBC[2]	MBC[1]	MBC[0]
		—	—	—	—	DLC[3]	DLC[2]	DLC[1]	DLC[0]
	MBn_CONT ROL1_2 (n=1 ~ 31)	—	—	NMC	ATX	DART	MBC[2]	MBC[1]	MBC[0]
		—	—	—	—	DLC[3]	DLC[2]	DLC[1]	DLC[0]
	MBn_TIME STAMP_2 (n=0 ~ 15、 30、31)	TS15	TS14	TS13	TS12	TS11	TS10	TS9	TS8
		TS7	TS6	TS5	TS4	TS3	TS2	TS1	TS0
	MBn_TTT_2 (n=24 ~ 30)	TTT15	TTT14	TTT13	TTT12	TTT11	TTT10	TTT9	TTT8
		TTT7	TTT6	TTT5	TTT4	TTT3	TTT2	TTT1	TTT0
MBn_TTCO NTROL_2 (n=24 ~ 29)	TTW[1]	TTW[0]	OFFSET[5]	OFFSET[4]	OFFSET[3]	OFFSET[2]	OFFSET[1]	OFFSET[0]	
	—	—	—	—	—	REP_FACT OR[2]	REP_FACT OR[1]	REP_FACT OR[0]	
IEBus 控制器	IECTR	—	IOL	DEE	—	RE	—	—	
	IECMR	—	—	—	—	—	CMD[2]	CMD[1]	CMD[0]
	IEMCR	SS	RN[2]	RN[1]	RN[0]	CTL[3]	CTL[2]	CTL[1]	CTL[0]
	IEAR1	IARL4[3]	IARL4[2]	IARL4[1]	IARL4[0]	IMD[1]	IMD[0]	—	STE
	IEAR2	IARU8[7]	IARU8[6]	IARU8[5]	IARU8[4]	IARU8[3]	IARU8[2]	IARU8[1]	IARU8[0]
	IESA1	ISAL4[3]	ISAL4[2]	ISAL4[1]	ISAL4[0]	—	—	—	—
	IESA2	ISAU8[7]	ISAU8[6]	ISAU8[5]	ISAU8[4]	ISAU8[3]	ISAU8[2]	ISAU8[1]	ISAU8[0]
	IETBFL	IBFL[7]	IBFL[6]	IBFL[5]	IBFL[4]	IBFL[3]	IBFL[2]	IBFL[1]	IBFL[0]
	IEMA1	IMAL4[3]	IMAL4[2]	IMAL4[1]	IMAL4[0]	—	—	—	—
	IEMA2	IMAU8[7]	IMAU8[6]	IMAU8[5]	IMAU8[4]	IMAU8[3]	IMAU8[2]	IMAU8[1]	IMAU8[0]
	IERCTL	—	—	—	—	RCTL[3]	RCTL[2]	RCTL[1]	RCTL[0]
	IERBFL	RBFL[7]	RBFL[6]	RBFL[5]	RBFL[4]	RBFL[3]	RBFL[2]	RBFL[1]	RBFL[0]
	IELA1	ILAL8[7]	ILAL8[6]	ILAL8[5]	ILAL8[4]	ILAL8[3]	ILAL8[2]	ILAL8[1]	ILAL8[0]
	IELA2	—	—	—	—	ILAU4[3]	ILAU4[2]	ILAU4[1]	ILAU4[0]
	IEFLG	CMX	MRQ	SRQ	SRE	LCK	—	RSS	GG
	IETSR	—	TXS	TXF	—	TXEAL	TXETTME	TXERO	TXEACK
	IEIET	—	TXSE	TXFE	—	TXEAL	TXETTME	TXEROE	TXEACKE
IERSR	RXBSY	RXS	RXF	RXEDE	RXEOVE	RXERTME	RXEDLE	RXEPE	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
IEBus 控制器	IEIER	RXBSYE	RXSE	RXFE	RXEDEE	RXEVEE	RXERTMEE	RXEDLEE	RXEPEE
	IECKSR	—	—	—	CKS3	—	CKS[2]	CKS[1]	CKS[0]
	IETB001 ~ IETB128								
	IERB001 ~ IERB128								
瑞萨 SPDIF 接口	TLCA	—	—	—	—	—	—	—	—
	TRCA	—	—	—	—	—	—	—	—
	TLCS	—	—	CLAC[1]	CLAC[0]	FS[3]	FS[2]	FS[1]	FS[0]
		CHNO[3]	CHNO[2]	CHNO[1]	CHNO[0]	SRCNO[3]	SRCNO[2]	SRCNO[1]	SRCNO[0]
		CATCD[7]	CATCD[6]	CATCD[5]	CATCD[4]	CATCD[3]	CATCD[2]	CATCD[1]	CATCD[0]
		—	—	CTL[4]	CTL[3]	CTL[2]	CTL[1]	CTL[0]	—
	TRCS	—	—	CLAC[1]	CLAC[0]	FS[3]	FS[2]	FS[1]	FS[0]
		CHNO[3]	CHNO[2]	CHNO[1]	CHNO[0]	SRCNO[3]	SRCNO[2]	SRCNO[1]	SRCNO[0]
		CATCD[7]	CATCD[6]	CATCD[5]	CATCD[4]	CATCD[3]	CATCD[2]	CATCD[1]	CATCD[0]
		—	—	CTL[4]	CTL[3]	CTL[2]	CTL[1]	CTL[0]	—
	TUI								
	RLCA	—	—	—	—	—	—	—	—
	RRCA	—	—	—	—	—	—	—	—
	RLCS	—	—	CLAC[1]	CLAC[0]	FS[3]	FS[2]	FS[1]	FS[0]
		CHNO[3]	CHNO[2]	CHNO[1]	CHNO[0]	SRCNO[3]	SRCNO[2]	SRCNO[1]	SRCNO[0]
		CATCD[7]	CATCD[6]	CATCD[5]	CATCD[4]	CATCD[3]	CATCD[2]	CATCD[1]	CATCD[0]
		—	—	CTL[4]	CTL[3]	CTL[2]	CTL[1]	CTL[0]	—
	RRCS	—	—	CLAC[1]	CLAC[0]	FS[3]	FS[2]	FS[1]	FS[0]
		CHNO[3]	CHNO[2]	CHNO[1]	CHNO[0]	SRCNO[3]	SRCNO[2]	SRCNO[1]	SRCNO[0]
		CATCD[7]	CATCD[6]	CATCD[5]	CATCD[4]	CATCD[3]	CATCD[2]	CATCD[1]	CATCD[0]
		—	—	CTL[4]	CTL[3]	CTL[2]	CTL[1]	CTL[0]	—
	RUI								

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
瑞萨 SPDIF 接口	CTRL	—	—	—	CKS	—	PB	RASS[1]	RASS[0]	
		TASS[1]	TASS[0]	RDE	TDE	NCSI	AOS	RME	TME	
		REIE	TEIE	UBOI	UBUI	CREI	PAEI	PREI	CSEI	
		ABOI	ABUI	RUII	TUII	RCSI	RCBI	TCSI	TCBI	
	STAT	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	CMD	
		RIS	TIS	UBO	UBU	CE	PARE	PREE	CSE	
		ABO	ABU	RUIR	TUIR	CSRX	CBRX	CSTX	CBTX	
	TDAD	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
	RDAD	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
	CD-ROM 解码器	CROMEN	SUBC_EN	CROM_EN	CROM_STP	—	—	—	—	—
		CROMSY0	SY_AUT	SY_IEN	SY_DEN	—	—	—	—	—
CROMCTL0		MD_DESC	—	MD_AUTO	MD_AUTOS1	MD_AUTOS2	MD_SEC[2]	MD_SEC[1]	MD_SEC[0]	
CROMCTL1		M2F2EDC	MD_DEC[2]	MD_DEC[1]	MD_DEC[0]	—	—	MD_PQREP[1]	MD_PQREP[0]	
CROMCTL3		STP_ECC	STP_EDC	—	STP_MD	STP_MIN	—	—	—	
CROMCTL4		LINKOFF	LINK2	—	EROSEL	NO_ECC	—	—	—	
CROMCTL5		—	—	—	—	—	—	—	MSF_LBA_SEL	
CROMST0		—	—	ST_SYIL	ST_SYNO	ST_BLKs	ST_BLKL	ST_SECS	ST_SECL	
CROMST1		—	—	—	—	ER2_HEAD0	ER2_HEAD1	ER2_HEAD2	ER2_HEAD3	
CROMST3		ER2_SHEAD0	ER2_SHEAD1	ER2_SHEAD2	ER2_SHEAD3	ER2_SHEAD4	ER2_SHEAD5	ER2_SHEAD6	ER2_SHEAD7	
CROMST4		NG_MD	NG_MDCMP1	NG_MDCMP2	NG_MDCMP3	NG_MDCMP4	NG_MDDEF	NG_MDTIM1	NG_MDTIM2	
CROMST5		ST_AMD[2]	ST_AMD[1]	ST_AMD[0]	ST_MDx	LINK_ON	LINK_DET	LINK_SDET	LINK_OUT1	
CROMST6		ST_ERR	—	ST_ECCABT	ST_ECCNG	ST_ECCP	ST_ECCQ	ST_EDC1	ST_EDC2	
CBUFST0		BUF_REF	BUF_ACT	—	—	—	—	—	—	
CBUFST1		BUF_ECC	BUF_EDC	—	BUF_MD	BUF_MIN	—	—	—	
CBUFST2		BUF_NG	—	—	—	—	—	—	—	
HEAD00		HEAD00[7]	HEAD00[6]	HEAD00[5]	HEAD00[4]	HEAD00[3]	HEAD00[2]	HEAD00[1]	HEAD00[0]	
HEAD01		HEAD01[7]	HEAD01[6]	HEAD01[5]	HEAD01[4]	HEAD01[3]	HEAD01[2]	HEAD01[1]	HEAD01[0]	
HEAD02		HEAD02[7]	HEAD02[6]	HEAD02[5]	HEAD02[4]	HEAD02[3]	HEAD02[2]	HEAD02[1]	HEAD02[0]	
HEAD03		HEAD03[7]	HEAD03[6]	HEAD03[5]	HEAD03[4]	HEAD03[3]	HEAD03[2]	HEAD03[1]	HEAD03[0]	
SHEAD00		SHEAD00[7]	SHEAD00[6]	SHEAD00[5]	SHEAD00[4]	SHEAD00[3]	SHEAD00[2]	SHEAD00[1]	SHEAD00[0]	
SHEAD01		SHEAD01[7]	SHEAD01[6]	SHEAD01[5]	SHEAD01[4]	SHEAD01[3]	SHEAD01[2]	SHEAD01[1]	SHEAD01[0]	
SHEAD02		SHEAD02[7]	SHEAD02[6]	SHEAD02[5]	SHEAD02[4]	SHEAD02[3]	SHEAD02[2]	SHEAD02[1]	SHEAD02[0]	
SHEAD03		SHEAD03[7]	SHEAD03[6]	SHEAD03[5]	SHEAD03[4]	SHEAD03[3]	SHEAD03[2]	SHEAD03[1]	SHEAD03[0]	
SHEAD04		SHEAD04[7]	SHEAD04[6]	SHEAD04[5]	SHEAD04[4]	SHEAD04[3]	SHEAD04[2]	SHEAD04[1]	SHEAD04[0]	
SHEAD05		SHEAD05[7]	SHEAD05[6]	SHEAD05[5]	SHEAD05[4]	SHEAD05[3]	SHEAD05[2]	SHEAD05[1]	SHEAD05[0]	
SHEAD06		SHEAD06[7]	SHEAD06[6]	SHEAD06[5]	SHEAD06[4]	SHEAD06[3]	SHEAD06[2]	SHEAD06[1]	SHEAD06[0]	
SHEAD07		SHEAD07[7]	SHEAD07[6]	SHEAD07[5]	SHEAD07[4]	SHEAD07[3]	SHEAD07[2]	SHEAD07[1]	SHEAD07[0]	
HEAD20		HEAD20[7]	HEAD20[6]	HEAD20[5]	HEAD20[4]	HEAD20[3]	HEAD20[2]	HEAD20[1]	HEAD20[0]	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
CD-ROM 解码器	HEAD21	HEAD21[7]	HEAD21[6]	HEAD21[5]	HEAD21[4]	HEAD21[3]	HEAD21[2]	HEAD21[1]	HEAD21[0]	
	HEAD22	HEAD22[7]	HEAD22[6]	HEAD22[5]	HEAD22[4]	HEAD22[3]	HEAD22[2]	HEAD22[1]	HEAD22[0]	
	HEAD23	HEAD23[7]	HEAD23[6]	HEAD23[5]	HEAD23[4]	HEAD23[3]	HEAD23[2]	HEAD23[1]	HEAD23[0]	
	SHEAD20	SHEAD20[7]	SHEAD20[6]	SHEAD20[5]	SHEAD20[4]	SHEAD20[3]	SHEAD20[2]	SHEAD20[1]	SHEAD20[0]	
	SHEAD21	SHEAD21[7]	SHEAD21[6]	SHEAD21[5]	SHEAD21[4]	SHEAD21[3]	SHEAD21[2]	SHEAD21[1]	SHEAD21[0]	
	SHEAD22	SHEAD22[7]	SHEAD22[6]	SHEAD22[5]	SHEAD22[4]	SHEAD22[3]	SHEAD22[2]	SHEAD22[1]	SHEAD22[0]	
	SHEAD23	SHEAD23[7]	SHEAD23[6]	SHEAD23[5]	SHEAD23[4]	SHEAD23[3]	SHEAD23[2]	SHEAD23[1]	SHEAD23[0]	
	SHEAD24	SHEAD24[7]	SHEAD24[6]	SHEAD24[5]	SHEAD24[4]	SHEAD24[3]	SHEAD24[2]	SHEAD24[1]	SHEAD24[0]	
	SHEAD25	SHEAD25[7]	SHEAD25[6]	SHEAD25[5]	SHEAD25[4]	SHEAD25[3]	SHEAD25[2]	SHEAD25[1]	SHEAD25[0]	
	SHEAD26	SHEAD26[7]	SHEAD26[6]	SHEAD26[5]	SHEAD26[4]	SHEAD26[3]	SHEAD26[2]	SHEAD26[1]	SHEAD26[0]	
	SHEAD27	SHEAD27[7]	SHEAD27[6]	SHEAD27[5]	SHEAD27[4]	SHEAD27[3]	SHEAD27[2]	SHEAD27[1]	SHEAD27[0]	
	CBUFCTL0	CBUF_AUT	CBUF_EN	CBUF_LINK	CBUF_MD[1]	CBUF_MD[0]	CBUF_TS	CBUF_Q	—	—
	CBUFCTL1	BS_MIN[7]	BS_MIN[6]	BS_MIN[5]	BS_MIN[4]	BS_MIN[3]	BS_MIN[2]	BS_MIN[1]	BS_MIN[0]	—
	CBUFCTL2	BS_SEC[7]	BS_SEC[6]	BS_SEC[5]	BS_SEC[4]	BS_SEC[3]	BS_SEC[2]	BS_SEC[1]	BS_SEC[0]	—
	CBUFCTL3	BS_FRM[7]	BS_FRM[6]	BS_FRM[5]	BS_FRM[4]	BS_FRM[3]	BS_FRM[2]	BS_FRM[1]	BS_FRM[0]	—
	CROMST0M	—	—	ST_SYILM	ST_SYNOM	ST_BLKSM	ST_BLKLM	ST_SECSM	ST_SECLM	—
	ROMDECRST	LOGICRST	RAMRST	—	—	—	—	—	—	—
	RSTSTAT	RAMCLRST	—	—	—	—	—	—	—	—
	SSI	BYTEND	BITEND	BUFEND0[1]	BUFEND0[0]	BUFEND1[1]	BUFEND1[0]	—	—	—
	INTHOLD	ISEC	ITARG	ISY	IERR	IBUF	IREADY	—	—	—
	INHINT	INHISEC	INHITARG	INHISY	INHIERR	INHIBUF	INHIREADY	PRE INHREQDM	PRE INHIREADY	—
	STRMDIN0	STRMDIN[31]	STRMDIN[30]	STRMDIN[29]	STRMDIN[28]	STRMDIN[27]	STRMDIN[26]	STRMDIN[25]	STRMDIN[24]	—
		STRMDIN[23]	STRMDIN[22]	STRMDIN[21]	STRMDIN[20]	STRMDIN[19]	STRMDIN[18]	STRMDIN[17]	STRMDIN[16]	—
	STRMDIN2	STRMDIN[15]	STRMDIN[14]	STRMDIN[13]	STRMDIN[12]	STRMDIN[11]	STRMDIN[10]	STRMDIN[9]	STRMDIN[8]	—
		STRMDIN[7]	STRMDIN[6]	STRMDIN[5]	STRMDIN[4]	STRMDIN[3]	STRMDIN[2]	STRMDIN[1]	STRMDIN[0]	—
	STRMDOUT0	STRMDOUT [15]	STRMDOUT [14]	STRMDOUT [13]	STRMDOUT [12]	STRMDOUT [11]	STRMDOUT [10]	STRMDOUT [9]	STRMDOUT [8]	—
		STRMDOUT [7]	STRMDOUT [6]	STRMDOUT [5]	STRMDOUT [4]	STRMDOUT [3]	STRMDOUT [2]	STRMDOUT [1]	STRMDOUT [0]	—
A/D 转换器	ADDRA	—	—	—	—	—	—	—	—	
	ADDRB	—	—	—	—	—	—	—	—	
	ADDRC	—	—	—	—	—	—	—	—	
	ADDRD	—	—	—	—	—	—	—	—	
	ADDRE	—	—	—	—	—	—	—	—	
	ADDRF	—	—	—	—	—	—	—	—	
	ADDRG	—	—	—	—	—	—	—	—	
	ADDRH	—	—	—	—	—	—	—	—	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
A/D 转换器	ADCSR	ADF	ADIE	ADST	TRGS[3]	TRGS[2]	TRGS[1]	TRGS[0]	CKS[2]	
		CKS[1]	CKS[0]	MDS[2]	MDS[1]	MDS[0]	CH[2]	CH[1]	CH[0]	
NAND 闪存 控制器	FLCMNCR	—	—	—	—	—	—	—	—	
		—	—	BUSYON	—	—	SNAND	QTSEL	—	
		—	—	—	—	ACM[1]	ACM[0]	NANDWF	—	
		—	—	—	—	CE	—	—	—	
	FLCMDCR	ADRCNT2	SCTCNT[19]	SCTCNT[18]	SCTCNT[17]	SCTCNT[16]	ADRMD	CDSRC	DOSR	—
		—	—	SELRW	DOADR	ADRCNT[1]	ADRCNT[0]	DOCMD2	DOCMD1	—
		SCTCNT[15]	SCTCNT[14]	SCTCNT[13]	SCTCNT[12]	SCTCNT[11]	SCTCNT[10]	SCTCNT[9]	SCTCNT[8]	—
		SCTCNT[7]	SCTCNT[6]	SCTCNT[5]	SCTCNT[4]	SCTCNT[3]	SCTCNT[2]	SCTCNT[1]	SCTCNT[0]	—
	FLCMCDR	—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		CMD2[7]	CMD2[6]	CMD2[5]	CMD2[4]	CMD2[3]	CMD2[2]	CMD2[1]	CMD2[0]	—
		CMD1[7]	CMD1[6]	CMD1[5]	CMD1[4]	CMD1[3]	CMD1[2]	CMD1[1]	CMD1[0]	—
	FLADR*3	ADR4[7]	ADR4[6]	ADR4[5]	ADR4[4]	ADR4[3]	ADR4[2]	ADR4[1]	ADR4[0]	—
		ADR3[7]	ADR3[6]	ADR3[5]	ADR3[4]	ADR3[3]	ADR3[2]	ADR3[1]	ADR3[0]	—
		ADR2[7]	ADR2[6]	ADR2[5]	ADR2[4]	ADR2[3]	ADR2[2]	ADR2[1]	ADR2[0]	—
		ADR1[7]	ADR1[6]	ADR1[5]	ADR1[4]	ADR1[3]	ADR1[2]	ADR1[1]	ADR1[0]	—
	FLADR1*4	—	—	—	—	—	—	ADR[25]	ADR[24]	—
		ADR[23]	ADR[22]	ADR[21]	ADR[20]	ADR[19]	ADR[18]	ADR[17]	ADR[16]	—
		ADR[15]	ADR[14]	ADR[13]	ADR[12]	ADR[11]	ADR[10]	ADR[9]	ADR[8]	—
		ADR[7]	ADR[6]	ADR[5]	ADR[4]	ADR[3]	ADR[2]	ADR[1]	ADR[0]	—
	FLADR2	—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		ADR5[7]	ADR5[6]	ADR5[5]	ADR5[4]	ADR5[3]	ADR5[2]	ADR5[1]	ADR5[0]	—
	FLDTCNTR	ECFLW[7]	ECFLW[6]	ECFLW[5]	ECFLW[4]	ECFLW[3]	ECFLW[2]	ECFLW[1]	ECFLW[0]	—
		DTFLW[7]	DTFLW[6]	DTFLW[5]	DTFLW[4]	DTFLW[3]	DTFLW[2]	DTFLW[1]	DTFLW[0]	—
		—	—	—	—	DTCNT[11]	DTCNT[10]	DTCNT[9]	DTCNT[8]	—
		DTCNT[7]	DTCNT[6]	DTCNT[5]	DTCNT[4]	DTCNT[3]	DTCNT[2]	DTCNT[1]	DTCNT[0]	—
	FLDATAR	DT4[7]	DT4[6]	DT4[5]	DT4[4]	DT4[3]	DT4[2]	DT4[1]	DT4[0]	—
		DT3[7]	DT3[6]	DT3[5]	DT3[4]	DT3[3]	DT3[2]	DT3[1]	DT3[0]	—
		DT2[7]	DT2[6]	DT2[5]	DT2[4]	DT2[3]	DT2[2]	DT2[1]	DT2[0]	—
		DT1[7]	DT1[6]	DT1[5]	DT1[4]	DT1[3]	DT1[2]	DT1[1]	DT1[0]	—
FLINTMACR	—	—	—	—	—	—	—	—	—	
	—	—	FIFOTRG[1]	FIFOTRG[0]	AC1CLR	AC0CLR	DREQ1EN	DREQ0EN	—	
	—	—	—	—	—	—	—	—	STERB	
	BTOERB	TRREQF1	TRREQF0	STERINTE	RBERINTE	TEINTE	TRINTE1	TRINTE0	—	
FLBSYTMR	—	—	—	—	—	—	—	—	—	
	—	—	—	—	RBTMOUT [19]	RBTMOUT [18]	RBTMOUT [17]	RBTMOUT [16]	—	
	RBTMOUT [15]	RBTMOUT [14]	RBTMOUT [13]	RBTMOUT [12]	RBTMOUT [11]	RBTMOUT [10]	RBTMOUT [9]	RBTMOUT [8]	—	
	RBTMOUT [7]	RBTMOUT [6]	RBTMOUT [5]	RBTMOUT [4]	RBTMOUT [3]	RBTMOUT [2]	RBTMOUT [1]	RBTMOUT [0]	—	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
NAND 闪存 控制器	FLBSYCNT	STAT[7]	STAT[6]	STAT[5]	STAT[4]	STAT[3]	STAT[2]	STAT[1]	STAT[0]	
		—	—	—	—	RBTIMCNT [19]	RBTIMCNT [18]	RBTIMCNT [17]	RBTIMCNT [16]	
		RBTIMCNT [15]	RBTIMCNT [14]	RBTIMCNT [13]	RBTIMCNT [12]	RBTIMCNT [11]	RBTIMCNT [10]	RBTIMCNT [9]	RBTIMCNT [8]	
		RBTIMCNT [7]	RBTIMCNT [6]	RBTIMCNT [5]	RBTIMCNT [4]	RBTIMCNT [3]	RBTIMCNT [2]	RBTIMCNT [1]	RBTIMCNT [0]	
	FLDTFIFO	DTFO[31]	DTFO[30]	DTFO[29]	DTFO[28]	DTFO[27]	DTFO[26]	DTFO[25]	DTFO[24]	
		DTFO[23]	DTFO[22]	DTFO[21]	DTFO[20]	DTFO[19]	DTFO[18]	DTFO[17]	DTFO[16]	
		DTFO[15]	DTFO[14]	DTFO[13]	DTFO[12]	DTFO[11]	DTFO[10]	DTFO[9]	DTFO[8]	
		DTFO[7]	DTFO[6]	DTFO[5]	DTFO[4]	DTFO[3]	DTFO[2]	DTFO[1]	DTFO[0]	
	FLECFIFO	ECFO[31]	ECFO[30]	ECFO[29]	ECFO[28]	ECFO[27]	ECFO[26]	ECFO[25]	ECFO[24]	
		ECFO[23]	ECFO[22]	ECFO[21]	ECFO[20]	ECFO[19]	ECFO[18]	ECFO[17]	ECFO[16]	
		ECFO[15]	ECFO[14]	ECFO[13]	ECFO[12]	ECFO[11]	ECFO[10]	ECFO[9]	ECFO[8]	
		ECFO[7]	ECFO[6]	ECFO[5]	ECFO[4]	ECFO[3]	ECFO[2]	ECFO[1]	ECFO[0]	
	FLTRCR	—	—	—	—	—	TRSTAT	TREND	TRSTRT	
	FLHOLDRCR	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	HOLDEN	
	USB2.0 主机 / 功能 模块	SYSCFG	—	—	—	—	—	SCKE	—	—
			HSE	DCFM	DRPD	DPRPU	UCKFSEL	UCKPSEL	UPLLE	USBE
		BUSWAIT	—	—	—	—	—	—	—	—
—			—	—	—	BWAIT[3]	BWAIT[2]	BWAIT[1]	BWAIT[0]	
SYSSTS		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	LNST[1]	LNST[0]	
DVSTCTR		—	—	—	—	—	—	—	WKUP	
		RWUPE	USBRST	RESUME	UACT	—	RHST[2]	RHST[1]	RHST[0]	
TESTMODE		—	—	—	—	—	—	—	—	
		—	—	—	—	UTST[3]	UTST[2]	UTST[1]	UTST[0]	
D0FBCFG		—	—	—	—	—	—	—	—	
		—	—	—	TENDE	—	—	—	—	
D1FBCFG		—	—	—	—	—	—	—	—	
		—	—	—	TENDE	—	—	—	—	
CFIFO		FIFOPORT [31]	FIFOPORT [30]	FIFOPORT [29]	FIFOPORT [28]	FIFOPORT [27]	FIFOPORT [26]	FIFOPORT [25]	FIFOPORT [24]	
		FIFOPORT [23]	FIFOPORT [22]	FIFOPORT [21]	FIFOPORT [20]	FIFOPORT [19]	FIFOPORT [18]	FIFOPORT [17]	FIFOPORT [16]	
		FIFOPORT [15]	FIFOPORT [14]	FIFOPORT [13]	FIFOPORT [12]	FIFOPORT [11]	FIFOPORT [10]	FIFOPORT [9]	FIFOPORT [8]	
		FIFOPORT [7]	FIFOPORT [6]	FIFOPORT [5]	FIFOPORT [4]	FIFOPORT [3]	FIFOPORT [2]	FIFOPORT [1]	FIFOPORT [0]	
D0FIFO		FIFOPORT [31]	FIFOPORT [30]	FIFOPORT [29]	FIFOPORT [28]	FIFOPORT [27]	FIFOPORT [26]	FIFOPORT [25]	FIFOPORT [24]	
		FIFOPORT [23]	FIFOPORT [22]	FIFOPORT [21]	FIFOPORT [20]	FIFOPORT [19]	FIFOPORT [18]	FIFOPORT [17]	FIFOPORT [16]	
	FIFOPORT [15]	FIFOPORT [14]	FIFOPORT [13]	FIFOPORT [12]	FIFOPORT [11]	FIFOPORT [10]	FIFOPORT [9]	FIFOPORT [8]		
	FIFOPORT [7]	FIFOPORT [6]	FIFOPORT [5]	FIFOPORT [4]	FIFOPORT [3]	FIFOPORT [2]	FIFOPORT [1]	FIFOPORT [0]		

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
USB2.0 主机 / 功能 模块	D1FIFO	FIFOPORT [31]	FIFOPORT [30]	FIFOPORT [29]	FIFOPORT [28]	FIFOPORT [27]	FIFOPORT [26]	FIFOPORT [25]	FIFOPORT [24]
		FIFOPORT [23]	FIFOPORT [22]	FIFOPORT [21]	FIFOPORT [20]	FIFOPORT [19]	FIFOPORT [18]	FIFOPORT [17]	FIFOPORT [16]
		FIFOPORT [15]	FIFOPORT [14]	FIFOPORT [13]	FIFOPORT [12]	FIFOPORT [11]	FIFOPORT [10]	FIFOPORT [9]	FIFOPORT [8]
		FIFOPORT [7]	FIFOPORT [6]	FIFOPORT [5]	FIFOPORT [4]	FIFOPORT [3]	FIFOPORT [2]	FIFOPORT [1]	FIFOPORT [0]
CFIFOSEL	RCNT	REW	—	—	MBW[1]	MBW[0]	—	BIGEND	
	—	—	ISEL	—	CURPIPE[3]	CURPIPE[2]	CURPIPE[1]	CURPIPE[0]	
CFIFOCTR	BVAL	BCLR	FRDY	—	DTLN[11]	DTLN[10]	DTLN[9]	DTLN[8]	
	DTLN[7]	DTLN[6]	DTLN[5]	DTLN[4]	DTLN[3]	DTLN[2]	DTLN[1]	DTLN[0]	
D0FIFOSEL	RCNT	REW	DCLRM	DREQE	MBW[1]	MBW[0]	—	BIGEND	
	—	—	—	—	CURPIPE[3]	CURPIPE[2]	CURPIPE[1]	CURPIPE[0]	
D0FIFOCTR	BVAL	BCLR	FRDY	—	DTLN[11]	DTLN[10]	DTLN[9]	DTLN[8]	
	DTLN[7]	DTLN[6]	DTLN[5]	DTLN[4]	DTLN[3]	DTLN[2]	DTLN[1]	DTLN[0]	
D1FIFOSEL	RCNT	REW	DCLRM	DREQE	MBW[1]	MBW[0]	—	BIGEND	
	—	—	—	—	CURPIPE[3]	CURPIPE[2]	CURPIPE[1]	CURPIPE[0]	
D1FIFOCTR	BVAL	BCLR	FRDY	—	DTLN[11]	DTLN[10]	DTLN[9]	DTLN[8]	
	DTLN[7]	DTLN[6]	DTLN[5]	DTLN[4]	DTLN[3]	DTLN[2]	DTLN[1]	DTLN[0]	
INTENB0	VBSE	RSME	SOFE	DVSE	CTRE	BEMPE	NRDYE	BRDYE	
	—	—	—	—	—	—	—	—	
INTENB1	—	BCHGE	—	DTCHE	ATTCHE	—	—	—	
	—	EOFERRE	SIGNE	SACKE	—	—	—	—	
BRDYENB	—	—	—	—	—	—	PIPE9BRDYE	PIPE8BRDYE	
	PIPE7BRDYE	PIPE6BRDYE	PIPE5BRDYE	PIPE4BRDYE	PIPE3BRDYE	PIPE2BRDYE	PIPE1BRDYE	PIPE0BRDYE	
NRDYENB	—	—	—	—	—	—	PIPE9NRDYE	PIPE8NRDYE	
	PIPE7NRDYE	PIPE6NRDYE	PIPE5NRDYE	PIPE4NRDYE	PIPE3NRDYE	PIPE2NRDYE	PIPE1NRDYE	PIPE0NRDYE	
BEMPENB	—	—	—	—	—	—	PIPE9BEMPE	PIPE8BEMPE	
	PIPE7BEMPE	PIPE6BEMPE	PIPE5BEMPE	PIPE4BEMPE	PIPE3BEMPE	PIPE2BEMPE	PIPE1BEMPE	PIPE0BEMPE	
SOFCFG	—	—	—	—	—	—	—	TRNENSEL	
	—	BRDYM	—	—	—	—	—	—	
INTSTS0	VBINT	RESM	SOFR	DVST	CTRT	BEMP	NRDY	BRDY	
	VBSTS	DVSQ[2]	DVSQ[1]	DVSQ[0]	VALID	CTSQ[2]	CTSQ[1]	CTSQ[0]	
INTSTS1	—	BCHG	—	DTCH	ATTCH	—	—	—	
	—	EOFERR	SIGN	SACK	—	—	—	—	
BRDYSTS	—	—	—	—	—	—	PIPE9BRDY	PIPE8BRDY	
	PIPE7BRDY	PIPE6BRDY	PIPE5BRDY	PIPE4BRDY	PIPE3BRDY	PIPE2BRDY	PIPE1BRDY	PIPE0BRDY	
NRDYSTS	—	—	—	—	—	—	PIPE9NRDY	PIPE8NRDY	
	PIPE7NRDY	PIPE6NRDY	PIPE5NRDY	PIPE4NRDY	PIPE3NRDY	PIPE2NRDY	PIPE1NRDY	PIPE0NRDY	
BEMPSTS	—	—	—	—	—	—	PIPE9BEMP	PIPE8BEMP	
	PIPE7BEMP	PIPE6BEMP	PIPE5BEMP	PIPE4BEMP	PIPE3BEMP	PIPE2BEMP	PIPE1BEMP	PIPE0BEMP	
FRMNUM	OVRN	CRCE	—	—	—	FRNM[10]	FRNM[9]	FRNM[8]	
	FRNM[7]	FRNM[6]	FRNM[5]	FRNM[4]	FRNM[3]	FRNM[2]	FRNM[1]	FRNM[0]	
UFRMNUM	—	—	—	—	—	—	—	—	
	—	—	—	—	—	UFRNM[2]	UFRNM[1]	UFRNM[0]	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
USB2.0 主机 / 功能 模块	USBADDR	—	—	—	—	—	—	—	—
		—	USBADDR[6]	USBADDR[5]	USBADDR[4]	USBADDR[3]	USBADDR[2]	USBADDR[1]	USBADDR[0]
	USBREQ	BREQUEST[7]	BREQUEST[6]	BREQUEST[5]	BREQUEST[4]	BREQUEST[3]	BREQUEST[2]	BREQUEST[1]	BREQUEST[0]
		BMREQUEST TYPE[7]	BMREQUEST TYPE[6]	BMREQUEST TYPE[5]	BMREQUEST TYPE[4]	BMREQUEST TYPE[3]	BMREQUEST TYPE[2]	BMREQUEST TYPE[1]	BMREQUEST TYPE[0]
	USBVAL	WVALUE[15]	WVALUE[14]	WVALUE[13]	WVALUE[12]	WVALUE[11]	WVALUE[10]	WVALUE[9]	WVALUE[8]
		WVALUE[7]	WVALUE[6]	WVALUE[5]	WVALUE[4]	WVALUE[3]	WVALUE[2]	WVALUE[1]	WVALUE[0]
	USBINDEX	WINDEX[15]	WINDEX[14]	WINDEX[13]	WINDEX[12]	WINDEX[11]	WINDEX[10]	WINDEX[9]	WINDEX[8]
		WINDEX[7]	WINDEX[6]	WINDEX[5]	WINDEX[4]	WINDEX[3]	WINDEX[2]	WINDEX[1]	WINDEX[0]
	USBLENG	WLENGTH[15]	WLENGTH[14]	WLENGTH[13]	WLENGTH[12]	WLENGTH[11]	WLENGTH[10]	WLENGTH[9]	WLENGTH[8]
		WLENGTH[7]	WLENGTH[6]	WLENGTH[5]	WLENGTH[4]	WLENGTH[3]	WLENGTH[2]	WLENGTH[1]	WLENGTH[0]
	DCPCFG	—	—	—	—	—	—	—	—
		—	—	—	DIR	—	—	—	—
	DCPMAXP	DEVSEL[3]	DEVSEL[2]	DEVSEL[1]	DEVSEL[0]	—	—	—	—
		—	MXPS[6]	MXPS[5]	MXPS[4]	MXPS[3]	MXPS[2]	MXPS[1]	MXPS[0]
	DCPCTR	BSTS	SUREQ	CSCLR	CSSTS	SUREQCLR	—	—	SQCLR
		SQSET	SQMON	PBUSY	PINGE	—	CCPL	PID[1]	PID[0]
	PIPESEL	—	—	—	—	—	—	—	—
		—	—	—	—	PIPESEL[3]	PIPESEL[2]	PIPESEL[1]	PIPESEL[0]
	PIPECFG	TYPE[1]	TYPE[0]	—	—	—	BFRE	DBLB	CNTMD
		SHTNAK	—	—	DIR	EPNUM[3]	EPNUM[2]	EPNUM[1]	EPNUM[0]
	PIPEBUF	—	BUFSIZE[4]	BUFSIZE[3]	BUFSIZE[2]	BUFSIZE[1]	BUFSIZE[0]	—	—
		—	BUFNMB[6]	BUFNMB[5]	BUFNMB[4]	BUFNMB[3]	BUFNMB[2]	BUFNMB[1]	BUFNMB[0]
	PIPEMAXP	DEVSEL[3]	DEVSEL[2]	DEVSEL[1]	DEVSEL[0]	—	MXPS[10]	MXPS[9]	MXPS[8]
		MXPS[7]	MXPS[6]	MXPS[5]	MXPS[4]	MXPS[3]	MXPS[2]	MXPS[1]	MXPS[0]
	PIPEPERI	—	—	—	IFIS	—	—	—	—
		—	—	—	—	—	IITV[2]	IITV[1]	IITV[0]
	PIPE1CTR	BSTS	INBUFM	CSCLR	CSSTS	—	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	—	—	—	PID[1]	PID[0]
	PIPE2CTR	BSTS	INBUFM	CSCLR	CSSTS	—	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	—	—	—	PID[1]	PID[0]
	PIPE3CTR	BSTS	INBUFM	CSCLR	CSSTS	—	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	—	—	—	PID[1]	PID[0]
	PIPE4CTR	BSTS	INBUFM	CSCLR	CSSTS	—	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	—	—	—	PID[1]	PID[0]
	PIPE5CTR	BSTS	INBUFM	CSCLR	CSSTS	—	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	—	—	—	PID[1]	PID[0]
	PIPE6CTR	BSTS	—	CSCLR	CSSTS	—	—	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	—	—	—	PID[1]	PID[0]
	PIPE7CTR	BSTS	—	CSCLR	CSSTS	—	—	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	—	—	—	PID[1]	PID[0]
	PIPE8CTR	BSTS	—	CSCLR	CSSTS	—	—	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	—	—	—	PID[1]	PID[0]
	PIPE9CTR	BSTS	—	CSCLR	CSSTS	—	—	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	—	—	—	PID[1]	PID[0]
	PIPE1TRE	—	—	—	—	—	—	TRENB	TRCLR
		—	—	—	—	—	—	—	—

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
USB2.0 主机 / 功能 模块	PIPE1TRN	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]
	PIPE2TRE	—	—	—	—	—	—	TRENB	TRCLR
		—	—	—	—	—	—	—	—
	PIPE2TRN	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]
	PIPE3TRE	—	—	—	—	—	—	TRENB	TRCLR
		—	—	—	—	—	—	—	—
	PIPE3TRN	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]
	PIPE4TRE	—	—	—	—	—	—	TRENB	TRCLR
		—	—	—	—	—	—	—	—
	PIPE4TRN	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]
	PIPE5TRE	—	—	—	—	—	—	TRENB	TRCLR
		—	—	—	—	—	—	—	—
	PIPE5TRN	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]
	DEVADD0	—	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]
		USBSPD[1]	USBSPD[0]	—	—	—	—	—	—
	DEVADD1	—	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]
		USBSPD[1]	USBSPD[0]	—	—	—	—	—	—
	DEVADD2	—	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]
		USBSPD[1]	USBSPD[0]	—	—	—	—	—	—
	DEVADD3	—	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]
		USBSPD[1]	USBSPD[0]	—	—	—	—	—	—
	DEVADD4	—	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]
		USBSPD[1]	USBSPD[0]	—	—	—	—	—	—
	DEVADD5	—	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]
		USBSPD[1]	USBSPD[0]	—	—	—	—	—	—
DEVADD6	—	HPPHUB[3]	HPPHUB[2]	HPPHUB[1]	HPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]	
	USBSPD[1]	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	—	—	—	
DEVADD7	—	HPPHUB[3]	HPPHUB[2]	HPPHUB[1]	HPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]	
	USBSPD[1]	USBSPD[0]	—	—	—	—	—	—	
DEVADD8	—	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]	
	USBSPD[1]	USBSPD[0]	—	—	—	—	—	—	
DEVADD9	—	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]	
	USBSPD[1]	USBSPD[0]	—	—	—	—	—	—	
DEVADDA	—	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]	
	USBSPD[1]	USBSPD[0]	—	—	—	—	—	—	
数字视频 解码器	ADCCR1	—	—	—	—	—	—	—	AGCMODE
		—	—	—	—	—	—	—	
	TGCR1	—	—	—	—	—	—	—	SRCLEFT[8]
SRCLEFT[7]		SRCLEFT[6]	SRCLEFT[5]	SRCLEFT[4]	SRCLEFT[3]	SRCLEFT[2]	SRCLEFT[1]	SRCLEFT[0]	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
数字视频 解码器	TGCR2	SRCTOP[5]	SRCTOP[4]	SRCTOP[3]	SRCTOP[2]	SRCTOP[1]	SRCTOP[0]	SRHEIGHT [9]	SRHEIGHT [8]	
		SRHEIGHT [7]	SRHEIGHT [6]	SRHEIGHT [5]	SRHEIGHT [4]	SRHEIGHT [3]	SRHEIGHT [2]	SRHEIGHT [1]	SRHEIGHT [0]	
	TGCR3	—	—	—	—	—	—	SRCWIDTH [10]	SRCWIDTH [9]	SRCWIDTH [8]
		SRCWIDTH [7]	SRCWIDTH [6]	SRCWIDTH [5]	SRCWIDTH [4]	SRCWIDTH [3]	SRCWIDTH [2]	SRCWIDTH [1]	SRCWIDTH [0]	
	SYNSCR1	LPFVSYNC[2]	LPFVSYNC[1]	LPFVSYNC[0]	LPFHSYNC[2]	LPFHSYNC[1]	LPFHSYNC[0]	—	—	—
		VELOCITY SHIFT_H[3]	VELOCITY SHIFT_H[2]	VELOCITY SHIFT_H[1]	VELOCITY SHIFT_H[0]	SLICER MODE_H[1]	SLICER MODE_H[0]	SLICER MODE_V[1]	SLICER MODE_V[0]	
	SYNSCR2	—	—	—	—	—	—	—	—	—
		SYNMAX DUTY_H[1]	SYNMAX DUTY_H[0]	SYNMIN DUTY_H[5]	SYNMIN DUTY_H[4]	SYNMIN DUTY_H[3]	SYNMIN DUTY_H[2]	SYNMIN DUTY_H[1]	SYNMIN DUTY_H[0]	
	SYNSCR3	—	—	—	—	—	—	—	—	—
		SSCLIP SEL[3]	SSCLIP SEL[2]	SSCLIP SEL[1]	SSCLIP SEL[0]	CSYNC SLICE_H[9]	CSYNC SLICE_H[8]	CSYNC SLICE_H[7]	CSYNC SLICE_H[6]	CSYNC SLICE_H[5]
	SYNSCR4	—	—	—	—	—	—	—	—	—
		SYNMAX DUTY_V[1]	SYNMAX DUTY_V[0]	SYNMIN DUTY_V[5]	SYNMIN DUTY_V[4]	SYNMIN DUTY_V[3]	SYNMIN DUTY_V[2]	SYNMIN DUTY_V[1]	SYNMIN DUTY_V[0]	
	SYNSCR5	—	—	—	—	—	—	—	—	—
		VSYNC DELAY	VSYNC SLICE[4]	VSYNC SLICE[3]	VSYNC SLICE[2]	VSYNC SLICE[1]	VSYNC SLICE[0]	CSYNC SLICE_V[9]	CSYNC SLICE_V[8]	
	HAFCCR1	—	—	—	—	—	—	—	—	—
		HAFCGAIN [3]	HAFCGAIN [2]	HAFCGAIN [1]	HAFCGAIN [0]	—	HAFCFREE RUN	HAFCTYP [9]	HAFCTYP [8]	
	HAFCCR2	—	—	—	—	—	—	—	—	—
		HAFCTYP[7]	HAFCTYP[6]	HAFCTYP[5]	HAFCTYP[4]	HAFCTYP[3]	HAFCTYP[2]	HAFCTYP[1]	HAFCTYP[0]	
	HAFCCR3	—	—	—	—	—	—	—	—	—
		HAFSTAR T[3]	HAFSTAR T[2]	HAFSTAR T[1]	HAFSTAR T[0]	NOX2HOSC	DOX2HOSC	HAFCMAX [9]	HAFCMAX [8]	
	VCDWCR1	—	—	—	—	—	—	—	—	—
		HAFCMAX [7]	HAFCMAX [6]	HAFCMAX [5]	HAFCMAX [4]	HAFCMAX [3]	HAFCMAX [2]	HAFCMAX [1]	HAFCMAX [0]	
	DCPCR1	—	—	—	—	—	—	—	—	—
		HAFCEMID[3]	HAFCEMID[2]	HAFCEMID[1]	HAFCEMID[0]	HAFCMODE[1]	HAFCMODE[0]	HAFCMIN[9]	HAFCMIN[8]	
	DCPCR2	—	—	—	—	—	—	—	—	—
		HAFCEMID[7]	HAFCEMID[6]	HAFCEMID[5]	HAFCEMID[4]	HAFCEMID[3]	HAFCEMID[2]	HAFCEMID[1]	HAFCEMID[0]	
	DCPCR3	—	—	—	—	—	—	—	—	—
		VCDFREERUN	NOVCD50	NOVCD60	VCD DEFAULT[1]	VCD DEFAULT[0]	VCD WINDOW[5]	VCD WINDOW[4]	VCD WINDOW[3]	
	DCPCR4	—	—	—	—	—	—	—	—	—
		VCD WINDOW[2]	VCD WINDOW[1]	VCD WINDOW[0]	VCD OFFSET[4]	VCD OFFSET[3]	VCD OFFSET[2]	VCD OFFSET[1]	VCD OFFSET[0]	
	DCPCR1	—	—	—	—	—	—	—	—	—
		DCPMODE_Y	—	—	—	DCPCHECK	—	BLANK LEVEL_Y[9]	BLANK LEVEL_Y[8]	
DCPCR2	—	—	—	—	—	—	—	—	—	
	BLANK LEVEL_Y[7]	BLANK LEVEL_Y[6]	BLANK LEVEL_Y[5]	BLANK LEVEL_Y[4]	BLANK LEVEL_Y[3]	BLANK LEVEL_Y[2]	BLANK LEVEL_Y[1]	BLANK LEVEL_Y[0]		
DCPCR3	—	—	—	—	—	—	—	—	—	
	DCPMODE_C	—	—	—	BLANK LEVEL_CB[5]	BLANK LEVEL_CB[4]	BLANK LEVEL_CB[3]	BLANK LEVEL_CB[2]		
DCPCR4	—	—	—	—	—	—	—	—	—	
	BLANK LEVEL_CB[1]	BLANK LEVEL_CB[0]	BLANK LEVEL_CR[5]	BLANK LEVEL_CR[4]	BLANK LEVEL_CR[3]	BLANK LEVEL_CR[2]	BLANK LEVEL_CR[1]	BLANK LEVEL_CR[0]		
DCPCR3	—	—	—	—	—	—	—	—	—	
	—	DCP RESPONSE[2]	DCP RESPONSE[1]	DCP RESPONSE[0]	—	—	—	—		
DCPCR4	—	—	—	—	—	—	—	—	—	
	DCPSTART[5]	DCPSTART[4]	DCPSTART[3]	DCPSTART[2]	DCPSTART[1]	DCPSTART[0]	—	—		

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
数字视频 解码器	DCPCR5	DCPEND[5]	DCPEND[4]	DCPEND[3]	DCPEND[2]	DCPEND[1]	DCPEND[0]	—	—
		—	—	—	—	—	—	—	—
	DCPCR6	—	DCPWIDTH[6]	DCPWIDTH[5]	DCPWIDTH[4]	DCPWIDTH[3]	DCPWIDTH[2]	DCPWIDTH[1]	DCPWIDTH[0]
		—	—	—	—	—	—	—	—
	DCPCR7	DCPPOS_Y[7]	DCPPOS_Y[6]	DCPPOS_Y[5]	DCPPOS_Y[4]	DCPPOS_Y[3]	DCPPOS_Y[2]	DCPPOS_Y[1]	DCPPOS_Y[0]
		—	—	—	—	—	—	—	—
	DCPCR8	DCPPOS_C[7]	DCPPOS_C[6]	DCPPOS_C[5]	DCPPOS_C[4]	DCPPOS_C[3]	DCPPOS_C[2]	DCPPOS_C[1]	DCPPOS_C[0]
		—	—	—	—	—	—	—	—
	NSDCR	—	—	ACFINPUT[1]	ACFINPUT[0]	—	—	—	ACFLAG TIME[4]
		ACFLAG TIME[3]	ACFLAG TIME[2]	ACFLAG TIME[1]	ACFLAG TIME[0]	—	—	ACF FILTER[1]	ACF FILTER[0]
	BTLCR	LOCK RANGE[1]	LOCK RANGE[0]	LOOP GAIN[1]	LOOP GAIN[0]	LOCK LIMIT[1]	LOCK LIMIT[0]	BCO FREERUN	—
		DEFAULT SYS[1]	DEFAULT SYS[0]	NONTSC358	NONTSC443	NOPALM	NOPALN	NOPAL443	NOSECAM
	BTGPCR	BGPCHECK	BGPWIDTH[6]	BGPWIDTH[5]	BGPWIDTH[4]	BGPWIDTH[3]	BGPWIDTH[2]	BGPWIDTH[1]	BGPWIDTH[0]
		BGPSTART[7]	BGPSTART[6]	BGPSTART[5]	BGPSTART[4]	BGPSTART[3]	BGPSTART[2]	BGPSTART[1]	BGPSTART[0]
	ACCCR1	KILLER OFFSET[3]	KILLER OFFSET[2]	KILLER OFFSET[1]	KILLER OFFSET[0]	ACCMODE	ACC MAXGAIN[1]	ACC MAXGAIN[0]	ACC LEVEL[8]
		ACCLEVEL[7]	ACCLEVEL[6]	ACCLEVEL[5]	ACCLEVEL[4]	ACCLEVEL[3]	ACCLEVEL[2]	ACCLEVEL[1]	ACCLEVEL[0]
	ACCCR2	—	—	—	—	—	CHROMA SUBGAIN[1]	CHROMA SUBGAIN[0]	CHROMA MAINGAIN[8]
		CHROMA MAINGAIN[7]	CHROMA MAINGAIN[6]	CHROMA MAINGAIN[5]	CHROMA MAINGAIN[4]	CHROMA MAINGAIN[3]	CHROMA MAINGAIN[2]	CHROMA MAINGAIN[1]	CHROMA MAINGAIN[0]
	ACCCR3	ACC RESPONSE[1]	ACC RESPONSE[0]	ACC PRECIS[5]	ACC PRECIS[4]	ACC PRECIS[3]	ACC PRECIS[2]	ACC PRECIS[1]	ACC PRECIS[0]
		KILLER MODE	KILLER LEVEL[5]	KILLER LEVEL[4]	KILLER LEVEL[3]	KILLER LEVEL[2]	KILLER LEVEL[1]	KILLER LEVEL[0]	—
	TINTCR	TINTSUB[5]	TINTSUB[4]	TINTSUB[3]	TINTSUB[2]	TINTSUB[1]	TINTSUB[0]	TINTMAIN[9]	TINTMAIN[8]
		TINTMAIN[7]	TINTMAIN[6]	TINTMAIN[5]	TINTMAIN[4]	TINTMAIN[3]	TINTMAIN[2]	TINTMAIN[1]	TINTMAIN[0]
	YDCR	—	—	—	—	—	—	—	LUMA DELAY[4]
		LUMADELAY [3]	LUMADELAY [2]	LUMADELAY [1]	LUMADELAY [0]	—	CHROMALPF	DEMOD MODE[1]	DEMOD MODE[0]
	AGCCR1	—	—	DOREDUCE	NOREDUCE	AGC RESPONSE[2]	AGC RESPONSE[1]	AGC RESPONSE[0]	AGC LEVEL[8]
		AGC LEVEL[7]	AGC LEVEL[6]	AGC LEVEL[5]	AGC LEVEL[4]	AGC LEVEL[3]	AGC LEVEL[2]	AGC LEVEL[1]	AGC LEVEL[0]
	AGCCR2	—	—	AGC PRECIS[5]	AGC PRECIS[4]	AGC PRECIS[3]	AGC PRECIS[2]	AGC PRECIS[1]	AGC PRECIS[0]
		—	—	—	—	—	—	—	—
	PKLIMITCR	PEAK LEVEL[1]	PEAK LEVEL[0]	PEAK ATTACK[1]	PEAK ATTACK[0]	PEAK RELEASE[1]	PEAK RELEASE[0]	PEAK RATIO[1]	PEAK RATIO[0]
		MAXPEAK SAMPLES[7]	MAXPEAK SAMPLES[6]	MAXPEAK SAMPLES[5]	MAXPEAK SAMPLES[4]	MAXPEAK SAMPLES[3]	MAXPEAK SAMPLES[2]	MAXPEAK SAMPLES[1]	MAXPEAK SAMPLES[0]
	RGORCR1	—	—	—	—	—	—	RADJ_O_ LEVEL[9]	RADJ_O_ LEVEL[8]
		RADJ_O_ LEVEL[7]	RADJ_O_ LEVEL[6]	RADJ_O_ LEVEL[5]	RADJ_O_ LEVEL[4]	RADJ_O_ LEVEL[3]	RADJ_O_ LEVEL[2]	RADJ_O_ LEVEL[1]	RADJ_O_ LEVEL[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
数字视频 解码器	RGORCR2	—	—	—	—	—	—	RADJ_U_ LEVEL0[9]	RADJ_U_ LEVEL0[8]
		RADJ_U_ LEVEL0[7]	RADJ_U_ LEVEL0[6]	RADJ_U_ LEVEL0[5]	RADJ_U_ LEVEL0[4]	RADJ_U_ LEVEL0[3]	RADJ_U_ LEVEL0[2]	RADJ_U_ LEVEL0[1]	RADJ_U_ LEVEL0[0]
	RGORCR3	—	—	—	—	—	—	RADJ_O_ LEVEL1[9]	RADJ_O_ LEVEL1[8]
		RADJ_O_ LEVEL1[7]	RADJ_O_ LEVEL1[6]	RADJ_O_ LEVEL1[5]	RADJ_O_ LEVEL1[4]	RADJ_O_ LEVEL1[3]	RADJ_O_ LEVEL1[2]	RADJ_O_ LEVEL1[1]	RADJ_O_ LEVEL1[0]
	RGORCR4	—	—	—	—	—	—	RADJ_U_ LEVEL1[9]	RADJ_U_ LEVEL1[8]
		RADJ_U_ LEVEL1[7]	RADJ_U_ LEVEL1[6]	RADJ_U_ LEVEL1[5]	RADJ_U_ LEVEL1[4]	RADJ_U_ LEVEL1[3]	RADJ_U_ LEVEL1[2]	RADJ_U_ LEVEL1[1]	RADJ_U_ LEVEL1[0]
	RGORCR5	—	—	—	—	—	—	RADJ_O_ LEVEL2[9]	RADJ_O_ LEVEL2[8]
		RADJ_O_ LEVEL2[7]	RADJ_O_ LEVEL2[6]	RADJ_O_ LEVEL2[5]	RADJ_O_ LEVEL2[4]	RADJ_O_ LEVEL2[3]	RADJ_O_ LEVEL2[2]	RADJ_O_ LEVEL2[1]	RADJ_O_ LEVEL2[0]
	RGORCR6	—	—	—	—	—	—	RADJ_U_ LEVEL2[9]	RADJ_U_ LEVEL2[8]
		RADJ_U_ LEVEL2[7]	RADJ_U_ LEVEL2[6]	RADJ_U_ LEVEL2[5]	RADJ_U_ LEVEL2[4]	RADJ_U_ LEVEL2[3]	RADJ_U_ LEVEL2[2]	RADJ_U_ LEVEL2[1]	RADJ_U_ LEVEL2[0]
	RGORCR7	—	TEST_MONI [2]	TEST_MONI [1]	TEST_MONI [0]	RADJ_MIX_ K_FIX[2]	RADJ_MIX_ K_FIX[1]	RADJ_MIX_ K_FIX[0]	—
		—	—	—	—	—	UCMP_SW	DCMP_SW	HWDI_SW
	AFCPFCR	—	—	—	—	—	—	—	—
		—	—	—	PHDET_FIX	—	PHDET_ DIV[2]	PHDET_ DIV[1]	PHDET_ DIV[0]
	RUPDCR	NEWSETTING	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
	VSYNCSR	FHCOUNT_L	FHLOCK	ISNOISY	FHMODE	NOSIGNAL	FVLOCK	FVMODE	INTERLACED
		FVCOUNT[7]	FVCOUNT[6]	FVCOUNT[5]	FVCOUNT[4]	FVCOUNT[3]	FVCOUNT[2]	FVCOUNT[1]	FVCOUNT[0]
	HSYNCSR	FHCOUNT_ H[16]	FHCOUNT_ H[15]	FHCOUNT_ H[14]	FHCOUNT_ H[13]	FHCOUNT_ H[12]	FHCOUNT_ H[11]	FHCOUNT_ H[10]	FHCOUNT_ H[9]
		FHCOUNT_ H[8]	FHCOUNT_ H[7]	FHCOUNT_ H[6]	FHCOUNT_ H[5]	FHCOUNT_ H[4]	FHCOUNT_ H[3]	FHCOUNT_ H[2]	FHCOUNT_ H[1]
	DCPSR1	CLAMP LEVEL_CB[5]	CLAMP LEVEL_CB[4]	CLAMP LEVEL_CB[3]	CLAMP LEVEL_CB[2]	CLAMP LEVEL_CB[1]	CLAMP LEVEL_CB[0]	CLAMP LEVEL_Y[9]	CLAMP LEVEL_Y[8]
		CLAMP LEVEL_Y[7]	CLAMP LEVEL_Y[6]	CLAMP LEVEL_Y[5]	CLAMP LEVEL_Y[4]	CLAMP LEVEL_Y[3]	CLAMP LEVEL_Y[2]	CLAMP LEVEL_Y[1]	CLAMP LEVEL_Y[0]
	DCPSR2	CLAMP LEVEL_CR[5]	CLAMP LEVEL_CR[4]	CLAMP LEVEL_CR[3]	CLAMP LEVEL_CR[2]	CLAMP LEVEL_CR[1]	CLAMP LEVEL_CR[0]	—	—
		—	—	—	—	—	—	—	—
	NSDSR	ACFSTREN GTH[15]	ACFSTREN GTH[14]	ACFSTREN GTH[13]	ACFSTREN GTH[12]	ACFSTREN GTH[11]	ACFSTREN GTH[10]	ACFSTREN GTH[9]	ACFSTREN GTH[8]
		ACFSTREN GTH[7]	ACFSTREN GTH[6]	ACFSTREN GTH[5]	ACFSTREN GTH[4]	ACFSTREN GTH[3]	ACFSTREN GTH[2]	ACFSTREN GTH[1]	ACFSTREN GTH[0]
	CROMASR1	COLOR SYS[1]	COLOR SYS[0]	FSCMODE	FSCLOCK	NOBURST	ACCSUB GAIN[1]	ACCSUB GAIN[0]	ACCMAIN GAIN[8]
		ACCMAIN GAIN[7]	ACCMAIN GAIN[6]	ACCMAIN GAIN[5]	ACCMAIN GAIN[4]	ACCMAIN GAIN[3]	ACCMAIN GAIN[2]	ACCMAIN GAIN[1]	ACCMAIN GAIN[0]
	CROMASR2	—	—	—	ISSECAM	ISPAL	ISNTSC	—	—
		LOCK LEVEL[7]	LOCK LEVEL[6]	LOCK LEVEL[5]	LOCK LEVEL[4]	LOCK LEVEL[3]	LOCK LEVEL[2]	LOCK LEVEL[1]	LOCK LEVEL[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
数字视频 解码器	SYNCCSR	—	—	—	ISREDUCED	—	—	SYNCDEPTH [9]	SYNCDEPTH [8]
		SYNCDEPTH [7]	SYNCDEPTH [6]	SYNCDEPTH [5]	SYNCDEPTH [4]	SYNCDEPTH [3]	SYNCDEPTH [2]	SYNCDEPTH [1]	SYNCDEPTH [0]
	AGCCSR1	HIGH SAMPLES[7]	HIGH SAMPLES[6]	HIGH SAMPLES[5]	HIGH SAMPLES[4]	HIGH SAMPLES[3]	HIGH SAMPLES[2]	HIGH SAMPLES[1]	HIGH SAMPLES[0]
		PEAK SAMPLES[7]	PEAK SAMPLES[6]	PEAK SAMPLES[5]	PEAK SAMPLES[4]	PEAK SAMPLES[3]	PEAK SAMPLES[2]	PEAK SAMPLES[1]	PEAK SAMPLES[0]
	AGCCSR2	—	—	—	—	—	—	—	AGC CONVERGE
		AGCGAIN[7]	AGCGAIN[6]	AGCGAIN[5]	AGCGAIN[4]	AGCGAIN[3]	AGCGAIN[2]	AGCGAIN[1]	AGCGAIN[0]
	YCSCR3	K15[3]	K15[2]	K15[1]	K15[0]	K13[5]	K13[4]	K13[3]	K13[2]
		K13[1]	K13[0]	K11[5]	K11[4]	K11[3]	K11[2]	K11[1]	K11[0]
	YCSCR4	K16[3]	K16[2]	K16[1]	K16[0]	K14[5]	K14[4]	K14[3]	K14[2]
		K14[1]	K14[0]	K12[5]	K12[4]	K12[3]	K12[2]	K12[1]	K12[0]
	YCSCR5	K22A[7]	K22A[6]	K22A[5]	K22A[4]	K22A[3]	K22A[2]	K22A[1]	K22A[0]
		—	—	K21A[5]	K21A[4]	K21A[3]	K21A[2]	K21A[1]	K21A[0]
	YCSCR6	K22B[7]	K22B[6]	K22B[5]	K22B[4]	K22B[3]	K22B[2]	K22B[1]	K22B[0]
		—	—	K21B[5]	K21B[4]	K21B[3]	K21B[2]	K21B[1]	K21B[0]
	YCSCR7	K23B[3]	K23B[2]	K23B[1]	K23B[0]	K23A[3]	K23A[2]	K23A[1]	K23A[0]
		—	—	—	K24[4]	K24[3]	K24[2]	K24[1]	K24[0]
	YCSCR8	HBPF_ NARROW	HVBPF_ NARROW	HBPF1_ 9TAP_ON	HVBPF1_ 9TAP_ON	HFIL_ TAP_SEL	—	—	—
		—	—	—	—	—	—	—	—
	YCSCR9	DET2_ON	—	—	—	HSEL_MIX_ Y[3]	HSEL_MIX_ Y[2]	HSEL_MIX_ Y[1]	HSEL_MIX_ Y[0]
		VSEL_MIX_ Y[3]	VSEL_MIX_ Y[2]	VSEL_MIX_ Y[1]	VSEL_MIX_ Y[0]	HVSEL_MIX_ Y[3]	HVSEL_MIX_ Y[2]	HVSEL_MIX_ Y[1]	HVSEL_MIX_ Y[0]
	YCSCR11	—	—	—	—	—	—	—	V_Y_ LEVEL[8]
		V_Y_ LEVEL[7]	V_Y_ LEVEL[6]	V_Y_ LEVEL[5]	V_Y_ LEVEL[4]	V_Y_ LEVEL[3]	V_Y_ LEVEL[2]	V_Y_ LEVEL[1]	V_Y_ LEVEL[0]
	YCSCR12	DET2_MIX_ C[3]	DET2_MIX_ C[2]	DET2_MIX_ C[1]	DET2_MIX_ C[0]	DET2_MIX_ Y[3]	DET2_MIX_ Y[2]	DET2_MIX_ Y[1]	DET2_MIX_ Y[0]
		—	—	—	—	FIL2_MODE _2D[1]	FIL2_MODE _2D[0]	—	FIL2_ NARROW_2D
	DCPCR9	—	—	—	CLP_HOLD _ON_Y	CLP_HOLD _ON_CB	CLP_HOLD _ON_CR	—	—
		—	—	—	—	—	—	—	—
	YCTWA_F0	—	—	—	FIL2_2D_ WA_F0[12]	FIL2_2D_ WA_F0[11]	FIL2_2D_ WA_F0[10]	FIL2_2D_ WA_F0[9]	FIL2_2D_ WA_F0[8]
		FIL2_2D_ WA_F0[7]	FIL2_2D_ WA_F0[6]	FIL2_2D_ WA_F0[5]	FIL2_2D_ WA_F0[4]	FIL2_2D_ WA_F0[3]	FIL2_2D_ WA_F0[2]	FIL2_2D_ WA_F0[1]	FIL2_2D_ WA_F0[0]
	YCTWA_F1	—	—	—	FIL2_2D_ WA_F1[12]	FIL2_2D_ WA_F1[11]	FIL2_2D_ WA_F1[10]	FIL2_2D_ WA_F1[9]	FIL2_2D_ WA_F1[8]
		FIL2_2D_ WA_F1[7]	FIL2_2D_ WA_F1[6]	FIL2_2D_ WA_F1[5]	FIL2_2D_ WA_F1[4]	FIL2_2D_ WA_F1[3]	FIL2_2D_ WA_F1[2]	FIL2_2D_ WA_F1[1]	FIL2_2D_ WA_F1[0]
	YCTWA_F2	—	—	—	FIL2_2D_ WA_F2[12]	FIL2_2D_ WA_F2[11]	FIL2_2D_ WA_F2[10]	FIL2_2D_ WA_F2[9]	FIL2_2D_ WA_F2[8]
		FIL2_2D_ WA_F2[7]	FIL2_2D_ WA_F2[6]	FIL2_2D_ WA_F2[5]	FIL2_2D_ WA_F2[4]	FIL2_2D_ WA_F2[3]	FIL2_2D_ WA_F2[2]	FIL2_2D_ WA_F2[1]	FIL2_2D_ WA_F2[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
数字视频 解码器	YCTWA_F3	—	—	—	FIL2_2D_ WA_F3[12]	FIL2_2D_ WA_F3[11]	FIL2_2D_ WA_F3[10]	FIL2_2D_ WA_F3[9]	FIL2_2D_ WA_F3[8]
		FIL2_2D_ WA_F3[7]	FIL2_2D_ WA_F3[6]	FIL2_2D_ WA_F3[5]	FIL2_2D_ WA_F3[4]	FIL2_2D_ WA_F3[3]	FIL2_2D_ WA_F3[2]	FIL2_2D_ WA_F3[1]	FIL2_2D_ WA_F3[0]
	YCTWA_F4	—	—	—	FIL2_2D_ WA_F4[12]	FIL2_2D_ WA_F4[11]	FIL2_2D_ WA_F4[10]	FIL2_2D_ WA_F4[9]	FIL2_2D_ WA_F4[8]
		FIL2_2D_ WA_F4[7]	FIL2_2D_ WA_F4[6]	FIL2_2D_ WA_F4[5]	FIL2_2D_ WA_F4[4]	FIL2_2D_ WA_F4[3]	FIL2_2D_ WA_F4[2]	FIL2_2D_ WA_F4[1]	FIL2_2D_ WA_F4[0]
	YCTWA_F5	—	—	—	FIL2_2D_ WA_F5[12]	FIL2_2D_ WA_F5[11]	FIL2_2D_ WA_F5[10]	FIL2_2D_ WA_F5[9]	FIL2_2D_ WA_F5[8]
		FIL2_2D_ WA_F5[7]	FIL2_2D_ WA_F5[6]	FIL2_2D_ WA_F5[5]	FIL2_2D_ WA_F5[4]	FIL2_2D_ WA_F5[3]	FIL2_2D_ WA_F5[2]	FIL2_2D_ WA_F5[1]	FIL2_2D_ WA_F5[0]
	YCTWA_F6	—	—	—	FIL2_2D_ WA_F6[12]	FIL2_2D_ WA_F6[11]	FIL2_2D_ WA_F6[10]	FIL2_2D_ WA_F6[9]	FIL2_2D_ WA_F6[8]
		FIL2_2D_ WA_F6[7]	FIL2_2D_ WA_F6[6]	FIL2_2D_ WA_F6[5]	FIL2_2D_ WA_F6[4]	FIL2_2D_ WA_F6[3]	FIL2_2D_ WA_F6[2]	FIL2_2D_ WA_F6[1]	FIL2_2D_ WA_F6[0]
	YCTWA_F7	—	—	—	FIL2_2D_ WA_F7[12]	FIL2_2D_ WA_F7[11]	FIL2_2D_ WA_F7[10]	FIL2_2D_ WA_F7[9]	FIL2_2D_ WA_F7[8]
		FIL2_2D_ WA_F7[7]	FIL2_2D_ WA_F7[6]	FIL2_2D_ WA_F7[5]	FIL2_2D_ WA_F7[4]	FIL2_2D_ WA_F7[3]	FIL2_2D_ WA_F7[2]	FIL2_2D_ WA_F7[1]	FIL2_2D_ WA_F7[0]
	YCTWA_F8	—	—	—	FIL2_2D_ WA_F8[12]	FIL2_2D_ WA_F8[11]	FIL2_2D_ WA_F8[10]	FIL2_2D_ WA_F8[9]	FIL2_2D_ WA_F8[8]
		FIL2_2D_ WA_F8[7]	FIL2_2D_ WA_F8[6]	FIL2_2D_ WA_F8[5]	FIL2_2D_ WA_F8[4]	FIL2_2D_ WA_F8[3]	FIL2_2D_ WA_F8[2]	FIL2_2D_ WA_F8[1]	FIL2_2D_ WA_F8[0]
	YCTWB_F0	—	—	—	FIL2_2D_ WB_F0[12]	FIL2_2D_ WB_F0[11]	FIL2_2D_ WB_F0[10]	FIL2_2D_ WB_F0[9]	FIL2_2D_ WB_F0[8]
		FIL2_2D_ WB_F0[7]	FIL2_2D_ WB_F0[6]	FIL2_2D_ WB_F0[5]	FIL2_2D_ WB_F0[4]	FIL2_2D_ WB_F0[3]	FIL2_2D_ WB_F0[2]	FIL2_2D_ WB_F0[1]	FIL2_2D_ WB_F0[0]
	YCTWB_F1	—	—	—	FIL2_2D_ WB_F1[12]	FIL2_2D_ WB_F1[11]	FIL2_2D_ WB_F1[10]	FIL2_2D_ WB_F1[9]	FIL2_2D_ WB_F1[8]
		FIL2_2D_ WB_F1[7]	FIL2_2D_ WB_F1[6]	FIL2_2D_ WB_F1[5]	FIL2_2D_ WB_F1[4]	FIL2_2D_ WB_F1[3]	FIL2_2D_ WB_F1[2]	FIL2_2D_ WB_F1[1]	FIL2_2D_ WB_F1[0]
	YCTWB_F2	—	—	—	FIL2_2D_ WB_F2[12]	FIL2_2D_ WB_F2[11]	FIL2_2D_ WB_F2[10]	FIL2_2D_ WB_F2[9]	FIL2_2D_ WB_F2[8]
		FIL2_2D_ WB_F2[7]	FIL2_2D_ WB_F2[6]	FIL2_2D_ WB_F2[5]	FIL2_2D_ WB_F2[4]	FIL2_2D_ WB_F2[3]	FIL2_2D_ WB_F2[2]	FIL2_2D_ WB_F2[1]	FIL2_2D_ WB_F2[0]
	YCTWB_F3	—	—	—	FIL2_2D_ WB_F3[12]	FIL2_2D_ WB_F3[11]	FIL2_2D_ WB_F3[10]	FIL2_2D_ WB_F3[9]	FIL2_2D_ WB_F3[8]
		FIL2_2D_ WB_F3[7]	FIL2_2D_ WB_F3[6]	FIL2_2D_ WB_F3[5]	FIL2_2D_ WB_F3[4]	FIL2_2D_ WB_F3[3]	FIL2_2D_ WB_F3[2]	FIL2_2D_ WB_F3[1]	FIL2_2D_ WB_F3[0]
	YCTWB_F4	—	—	—	FIL2_2D_ WB_F4[12]	FIL2_2D_ WB_F4[11]	FIL2_2D_ WB_F4[10]	FIL2_2D_ WB_F4[9]	FIL2_2D_ WB_F4[8]
		FIL2_2D_ WB_F4[7]	FIL2_2D_ WB_F4[6]	FIL2_2D_ WB_F4[5]	FIL2_2D_ WB_F4[4]	FIL2_2D_ WB_F4[3]	FIL2_2D_ WB_F4[2]	FIL2_2D_ WB_F4[1]	FIL2_2D_ WB_F4[0]
	YCTWB_F5	—	—	—	FIL2_2D_ WB_F5[12]	FIL2_2D_ WB_F5[11]	FIL2_2D_ WB_F5[10]	FIL2_2D_ WB_F5[9]	FIL2_2D_ WB_F5[8]
		FIL2_2D_ WB_F5[7]	FIL2_2D_ WB_F5[6]	FIL2_2D_ WB_F5[5]	FIL2_2D_ WB_F5[4]	FIL2_2D_ WB_F5[3]	FIL2_2D_ WB_F5[2]	FIL2_2D_ WB_F5[1]	FIL2_2D_ WB_F5[0]
YCTWB_F6	—	—	—	FIL2_2D_ WB_F6[12]	FIL2_2D_ WB_F6[11]	FIL2_2D_ WB_F6[10]	FIL2_2D_ WB_F6[9]	FIL2_2D_ WB_F6[8]	
	FIL2_2D_ WB_F6[7]	FIL2_2D_ WB_F6[6]	FIL2_2D_ WB_F6[5]	FIL2_2D_ WB_F6[4]	FIL2_2D_ WB_F6[3]	FIL2_2D_ WB_F6[2]	FIL2_2D_ WB_F6[1]	FIL2_2D_ WB_F6[0]	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
数字视频 解码器	YCTWB_F7	—	—	—	FIL2_2D_ WB_F7[12]	FIL2_2D_ WB_F7[11]	FIL2_2D_ WB_F7[10]	FIL2_2D_ WB_F7[9]	FIL2_2D_ WB_F7[8]
		FIL2_2D_ WB_F7[7]	FIL2_2D_ WB_F7[6]	FIL2_2D_ WB_F7[5]	FIL2_2D_ WB_F7[4]	FIL2_2D_ WB_F7[3]	FIL2_2D_ WB_F7[2]	FIL2_2D_ WB_F7[1]	FIL2_2D_ WB_F7[0]
	YCTWB_F8	—	—	—	FIL2_2D_ WB_F8[12]	FIL2_2D_ WB_F8[11]	FIL2_2D_ WB_F8[10]	FIL2_2D_ WB_F8[9]	FIL2_2D_ WB_F8[8]
		FIL2_2D_ WB_F8[7]	FIL2_2D_ WB_F8[6]	FIL2_2D_ WB_F8[5]	FIL2_2D_ WB_F8[4]	FIL2_2D_ WB_F8[3]	FIL2_2D_ WB_F8[2]	FIL2_2D_ WB_F8[1]	FIL2_2D_ WB_F8[0]
	YCTNA_F0	—	—	—	FIL2_2D_NA_ F0[12]	FIL2_2D_NA_ F0[11]	FIL2_2D_NA_ F0[10]	FIL2_2D_NA_ F0[9]	FIL2_2D_NA_ F0[8]
		FIL2_2D_NA_ F0[7]	FIL2_2D_NA_ F0[6]	FIL2_2D_NA_ F0[5]	FIL2_2D_NA_ F0[4]	FIL2_2D_NA_ F0[3]	FIL2_2D_NA_ F0[2]	FIL2_2D_NA_ F0[1]	FIL2_2D_NA_ F0[0]
	YCTNA_F1	—	—	—	FIL2_2D_NA_ F1[12]	FIL2_2D_NA_ F1[11]	FIL2_2D_NA_ F1[10]	FIL2_2D_NA_ F1[9]	FIL2_2D_NA_ F1[8]
		FIL2_2D_NA_ F1[7]	FIL2_2D_NA_ F1[6]	FIL2_2D_NA_ F1[5]	FIL2_2D_NA_ F1[4]	FIL2_2D_NA_ F1[3]	FIL2_2D_NA_ F1[2]	FIL2_2D_NA_ F1[1]	FIL2_2D_NA_ F1[0]
	YCTNA_F2	—	—	—	FIL2_2D_NA_ F2[12]	FIL2_2D_NA_ F2[11]	FIL2_2D_NA_ F2[10]	FIL2_2D_NA_ F2[9]	FIL2_2D_NA_ F2[8]
		FIL2_2D_NA_ F2[7]	FIL2_2D_NA_ F2[6]	FIL2_2D_NA_ F2[5]	FIL2_2D_NA_ F2[4]	FIL2_2D_NA_ F2[3]	FIL2_2D_NA_ F2[2]	FIL2_2D_NA_ F2[1]	FIL2_2D_NA_ F2[0]
	YCTNA_F3	—	—	—	FIL2_2D_NA_ F3[12]	FIL2_2D_NA_ F3[11]	FIL2_2D_NA_ F3[10]	FIL2_2D_NA_ F3[9]	FIL2_2D_NA_ F3[8]
		FIL2_2D_NA_ F3[7]	FIL2_2D_NA_ F3[6]	FIL2_2D_NA_ F3[5]	FIL2_2D_NA_ F3[4]	FIL2_2D_NA_ F3[3]	FIL2_2D_NA_ F3[2]	FIL2_2D_NA_ F3[1]	FIL2_2D_NA_ F3[0]
	YCTNA_F4	—	—	—	FIL2_2D_NA_ F4[12]	FIL2_2D_NA_ F4[11]	FIL2_2D_NA_ F4[10]	FIL2_2D_NA_ F4[9]	FIL2_2D_NA_ F4[8]
		FIL2_2D_NA_ F4[7]	FIL2_2D_NA_ F4[6]	FIL2_2D_NA_ F4[5]	FIL2_2D_NA_ F4[4]	FIL2_2D_NA_ F4[3]	FIL2_2D_NA_ F4[2]	FIL2_2D_NA_ F4[1]	FIL2_2D_NA_ F4[0]
	YCTNA_F5	—	—	—	FIL2_2D_NA_ F5[12]	FIL2_2D_NA_ F5[11]	FIL2_2D_NA_ F5[10]	FIL2_2D_NA_ F5[9]	FIL2_2D_NA_ F5[8]
		FIL2_2D_NA_ F5[7]	FIL2_2D_NA_ F5[6]	FIL2_2D_NA_ F5[5]	FIL2_2D_NA_ F5[4]	FIL2_2D_NA_ F5[3]	FIL2_2D_NA_ F5[2]	FIL2_2D_NA_ F5[1]	FIL2_2D_NA_ F5[0]
	YCTNA_F6	—	—	—	FIL2_2D_NA_ F6[12]	FIL2_2D_NA_ F6[11]	FIL2_2D_NA_ F6[10]	FIL2_2D_NA_ F6[9]	FIL2_2D_NA_ F6[8]
		FIL2_2D_NA_ F6[7]	FIL2_2D_NA_ F6[6]	FIL2_2D_NA_ F6[5]	FIL2_2D_NA_ F6[4]	FIL2_2D_NA_ F6[3]	FIL2_2D_NA_ F6[2]	FIL2_2D_NA_ F6[1]	FIL2_2D_NA_ F6[0]
	YCTNA_F7	—	—	—	FIL2_2D_NA_ F7[12]	FIL2_2D_NA_ F7[11]	FIL2_2D_NA_ F7[10]	FIL2_2D_NA_ F7[9]	FIL2_2D_NA_ F7[8]
		FIL2_2D_NA_ F7[7]	FIL2_2D_NA_ F7[6]	FIL2_2D_NA_ F7[5]	FIL2_2D_NA_ F7[4]	FIL2_2D_NA_ F7[3]	FIL2_2D_NA_ F7[2]	FIL2_2D_NA_ F7[1]	FIL2_2D_NA_ F7[0]
	YCTNA_F8	—	—	—	FIL2_2D_NA_ F8[12]	FIL2_2D_NA_ F8[11]	FIL2_2D_NA_ F8[10]	FIL2_2D_NA_ F8[9]	FIL2_2D_NA_ F8[8]
		FIL2_2D_NA_ F8[7]	FIL2_2D_NA_ F8[6]	FIL2_2D_NA_ F8[5]	FIL2_2D_NA_ F8[4]	FIL2_2D_NA_ F8[3]	FIL2_2D_NA_ F8[2]	FIL2_2D_NA_ F8[1]	FIL2_2D_NA_ F8[0]
	YCTNB_F0	—	—	—	FIL2_2D_NB_ F0[12]	FIL2_2D_NB_ F0[11]	FIL2_2D_NB_ F0[10]	FIL2_2D_NB_ F0[9]	FIL2_2D_NB_ F0[8]
		FIL2_2D_NB_ F0[7]	FIL2_2D_NB_ F0[6]	FIL2_2D_NB_ F0[5]	FIL2_2D_NB_ F0[4]	FIL2_2D_NB_ F0[3]	FIL2_2D_NB_ F0[2]	FIL2_2D_NB_ F0[1]	FIL2_2D_NB_ F0[0]
YCTNB_F1	—	—	—	FIL2_2D_NB_ F1[12]	FIL2_2D_NB_ F1[11]	FIL2_2D_NB_ F1[10]	FIL2_2D_NB_ F1[9]	FIL2_2D_NB_ F1[8]	
	FIL2_2D_NB_ F1[7]	FIL2_2D_NB_ F1[6]	FIL2_2D_NB_ F1[5]	FIL2_2D_NB_ F1[4]	FIL2_2D_NB_ F1[3]	FIL2_2D_NB_ F1[2]	FIL2_2D_NB_ F1[1]	FIL2_2D_NB_ F1[0]	
YCTNB_F2	—	—	—	FIL2_2D_NB_ F2[12]	FIL2_2D_NB_ F2[11]	FIL2_2D_NB_ F2[10]	FIL2_2D_NB_ F2[9]	FIL2_2D_NB_ F2[8]	
	FIL2_2D_NB_ F2[7]	FIL2_2D_NB_ F2[6]	FIL2_2D_NB_ F2[5]	FIL2_2D_NB_ F2[4]	FIL2_2D_NB_ F2[3]	FIL2_2D_NB_ F2[2]	FIL2_2D_NB_ F2[1]	FIL2_2D_NB_ F2[0]	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
数字视频 解码器	YCTNB_F3	—	—	—	FIL2_2D_NB_F3[12]	FIL2_2D_NB_F3[11]	FIL2_2D_NB_F3[10]	FIL2_2D_NB_F3[9]	FIL2_2D_NB_F3[8]
		FIL2_2D_NB_F3[7]	FIL2_2D_NB_F3[6]	FIL2_2D_NB_F3[5]	FIL2_2D_NB_F3[4]	FIL2_2D_NB_F3[3]	FIL2_2D_NB_F3[2]	FIL2_2D_NB_F3[1]	FIL2_2D_NB_F3[0]
	YCTNB_F4	—	—	—	FIL2_2D_NB_F4[12]	FIL2_2D_NB_F4[11]	FIL2_2D_NB_F4[10]	FIL2_2D_NB_F4[9]	FIL2_2D_NB_F4[8]
		FIL2_2D_NB_F4[7]	FIL2_2D_NB_F4[6]	FIL2_2D_NB_F4[5]	FIL2_2D_NB_F4[4]	FIL2_2D_NB_F4[3]	FIL2_2D_NB_F4[2]	FIL2_2D_NB_F4[1]	FIL2_2D_NB_F4[0]
	YCTNB_F5	—	—	—	FIL2_2D_NB_F5[12]	FIL2_2D_NB_F5[11]	FIL2_2D_NB_F5[10]	FIL2_2D_NB_F5[9]	FIL2_2D_NB_F5[8]
		FIL2_2D_NB_F5[7]	FIL2_2D_NB_F5[6]	FIL2_2D_NB_F5[5]	FIL2_2D_NB_F5[4]	FIL2_2D_NB_F5[3]	FIL2_2D_NB_F5[2]	FIL2_2D_NB_F5[1]	FIL2_2D_NB_F5[0]
	YCTNB_F6	—	—	—	FIL2_2D_NB_F6[12]	FIL2_2D_NB_F6[11]	FIL2_2D_NB_F6[10]	FIL2_2D_NB_F6[9]	FIL2_2D_NB_F6[8]
		FIL2_2D_NB_F6[7]	FIL2_2D_NB_F6[6]	FIL2_2D_NB_F6[5]	FIL2_2D_NB_F6[4]	FIL2_2D_NB_F6[3]	FIL2_2D_NB_F6[2]	FIL2_2D_NB_F6[1]	FIL2_2D_NB_F6[0]
	YCTNB_F7	—	—	—	FIL2_2D_NB_F7[12]	FIL2_2D_NB_F7[11]	FIL2_2D_NB_F7[10]	FIL2_2D_NB_F7[9]	FIL2_2D_NB_F7[8]
		FIL2_2D_NB_F7[7]	FIL2_2D_NB_F7[6]	FIL2_2D_NB_F7[5]	FIL2_2D_NB_F7[4]	FIL2_2D_NB_F7[3]	FIL2_2D_NB_F7[2]	FIL2_2D_NB_F7[1]	FIL2_2D_NB_F7[0]
	YCTNB_F8	—	—	—	FIL2_2D_NB_F8[12]	FIL2_2D_NB_F8[11]	FIL2_2D_NB_F8[10]	FIL2_2D_NB_F8[9]	FIL2_2D_NB_F8[8]
		FIL2_2D_NB_F8[7]	FIL2_2D_NB_F8[6]	FIL2_2D_NB_F8[5]	FIL2_2D_NB_F8[4]	FIL2_2D_NB_F8[3]	FIL2_2D_NB_F8[2]	FIL2_2D_NB_F8[1]	FIL2_2D_NB_F8[0]
	YGAINCR	—	—	—	—	—	—	Y_GAIN2[9]	Y_GAIN2[8]
		Y_GAIN2[7]	Y_GAIN2[6]	Y_GAIN2[5]	Y_GAIN2[4]	Y_GAIN2[3]	Y_GAIN2[2]	Y_GAIN2[1]	Y_GAIN2[0]
	CBGAINCR	—	—	—	—	—	—	CB_GAIN2[9]	CB_GAIN2[8]
		CB_GAIN2[7]	CB_GAIN2[6]	CB_GAIN2[5]	CB_GAIN2[4]	CB_GAIN2[3]	CB_GAIN2[2]	CB_GAIN2[1]	CB_GAIN2[0]
	CRGAINCR	—	—	—	—	—	—	CR_GAIN2[9]	CR_GAIN2[8]
		CR_GAIN2[7]	CR_GAIN2[6]	CR_GAIN2[5]	CR_GAIN2[4]	CR_GAIN2[3]	CR_GAIN2[2]	CR_GAIN2[1]	CR_GAIN2[0]
	PGA_UPDATE	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	PGA_VEN
	PGACR	—	—	PGA_GAIN_SEL	PGA_GAIN[4]	PGA_GAIN[3]	PGA_GAIN[2]	PGA_GAIN[1]	PGA_GAIN[0]
		—	—	—	—	—	—	—	—
	ADCCR2	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	ADC_VINSEL
视频显示 控制器 4	INP_UPDATE	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	
		—	—	—	INP_EXT_UPDATE	—	—	—	INP_IMG_UPDATE
	INP_SEL_CNT	—	—	—	—	—	—	—	—
		—	—	—	INP_SEL	—	—	—	—
		—	INP_FORMAT[2]	INP_FORMAT[1]	INP_FORMAT[0]	—	—	—	INP_PXD_EDGE
		—	—	—	INP_VS_EDGE	—	—	—	INP_HS_EDGE

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	INP_EXT_ SYNC_CNT	—	—	—	INP_ ENDIAN_ON	—	—	—	INP_SWAP_ ON
		—	—	—	INP_VS_INV	—	—	—	INP_HS_INV
		—	—	—	—	—	—	—	INP_H_ EDGE_SEL
		—	—	—	INP_F525_625	—	—	INP_H_ POS[1]	INP_H_ POS[0]
	INP_VSYNC_ _PH_ADJ	—	—	—	—	—	—	INP_FH50[9]	INP_FH50[8]
		INP_FH50[7]	INP_FH50[6]	INP_FH50[5]	INP_FH50[4]	INP_FH50[3]	INP_FH50[2]	INP_FH50[1]	INP_FH50[0]
		—	—	—	—	—	—	INP_FH25[9]	INP_FH25[8]
	INP_DLY_ ADJ	INP_FH25[7]	INP_FH25[6]	INP_FH25[5]	INP_FH25[4]	INP_FH25[3]	INP_FH25[2]	INP_FH25[1]	INP_FH25[0]
		—	—	—	—	—	INP_VS_ DLY_L[2]	INP_VS_ DLY_L[1]	INP_VS_ DLY_L[0]
		INP_FLD_ DLY[7]	INP_FLD_ DLY[6]	INP_FLD_ DLY[5]	INP_FLD_ DLY[4]	INP_FLD_ DLY[3]	INP_FLD_ DLY[2]	INP_FLD_ DLY[1]	INP_FLD_ DLY[0]
		INP_VS_ DLY[7]	INP_VS_ DLY[6]	INP_VS_ DLY[5]	INP_VS_ DLY[4]	INP_VS_ DLY[3]	INP_VS_ DLY[2]	INP_VS_ DLY[1]	INP_VS_ DLY[0]
	IMGCNT_ UPDATE	INP_HS_ DLY[7]	INP_HS_ DLY[6]	INP_HS_ DLY[5]	INP_HS_ DLY[4]	INP_HS_ DLY[3]	INP_HS_ DLY[2]	INP_HS_ DLY[1]	INP_HS_ DLY[0]
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	IMGCNT_VEN
	IMGCNT_ NR_CNT0	—	—	—	—	—	—	—	—
		—	—	—	NR1D_MD	—	—	—	NR1D_ON
		—	NR1D_Y_ TH[6]	NR1D_Y_ TH[5]	NR1D_Y_ TH[4]	NR1D_Y_ TH[3]	NR1D_Y_ TH[2]	NR1D_Y_ TH[1]	NR1D_Y_ TH[0]
		—	—	NR1D_Y_ TAP[1]	NR1D_Y_ TAP[0]	—	—	NR1D_Y_ GAIN[1]	NR1D_Y_ GAIN[0]
	IMGCNT_ NR_CNT1	—	NR1D_CB_ TH[6]	NR1D_CB_ TH[5]	NR1D_CB_ TH[4]	NR1D_CB_ TH[3]	NR1D_CB_ TH[2]	NR1D_CB_ TH[1]	NR1D_CB_ TH[0]
		—	—	NR1D_CB_ TAP[1]	NR1D_CB_ TAP[0]	—	—	NR1D_CB_ GAIN[1]	NR1D_CB_ GAIN[0]
		—	NR1D_CR_ TH[6]	NR1D_CR_ TH[5]	NR1D_CR_ TH[4]	NR1D_CR_ TH[3]	NR1D_CR_ TH[2]	NR1D_CR_ TH[1]	NR1D_CR_ TH[0]
		—	—	NR1D_CR_ TAP[1]	NR1D_CR_ TAP[0]	—	—	NR1D_CR_ GAIN[1]	NR1D_CR_ GAIN[0]
	IMGCNT_ MTX_MODE	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	IMGCNT_ MTX_MD[1]	IMGCNT_ MTX_MD[0]
	IMGCNT_ MTX_YG_ ADJ0	—	—	—	—	—	—	—	—
		IMGCNT_ MTX_YG[7]	IMGCNT_ MTX_YG[6]	IMGCNT_ MTX_YG[5]	IMGCNT_ MTX_YG[4]	IMGCNT_ MTX_YG[3]	IMGCNT_ MTX_YG[2]	IMGCNT_ MTX_YG[1]	IMGCNT_ MTX_YG[0]
		—	—	—	—	—	IMGCNT_ MTX_GG[10]	IMGCNT_ MTX_GG[9]	IMGCNT_ MTX_GG[8]
		IMGCNT_ MTX_GG[7]	IMGCNT_ MTX_GG[6]	IMGCNT_ MTX_GG[5]	IMGCNT_ MTX_GG[4]	IMGCNT_ MTX_GG[3]	IMGCNT_ MTX_GG[2]	IMGCNT_ MTX_GG[1]	IMGCNT_ MTX_GG[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	IMGCNT_ MTX_YG_ ADJ1	—	—	—	—	—	IMGCNT_ MTX_GB[10]	IMGCNT_ MTX_GB[9]	IMGCNT_ MTX_GB[8]
		IMGCNT_ MTX_GB[7]	IMGCNT_ MTX_GB[6]	IMGCNT_ MTX_GB[5]	IMGCNT_ MTX_GB[4]	IMGCNT_ MTX_GB[3]	IMGCNT_ MTX_GB[2]	IMGCNT_ MTX_GB[1]	IMGCNT_ MTX_GB[0]
		—	—	—	—	—	IMGCNT_ MTX_GR[10]	IMGCNT_ MTX_GR[9]	IMGCNT_ MTX_GR[8]
		IMGCNT_ MTX_GR[7]	IMGCNT_ MTX_GR[6]	IMGCNT_ MTX_GR[5]	IMGCNT_ MTX_GR[4]	IMGCNT_ MTX_GR[3]	IMGCNT_ MTX_GR[2]	IMGCNT_ MTX_GR[1]	IMGCNT_ MTX_GR[0]
	IMGCNT_ MTX_CBB_ ADJ0	—	—	—	—	—	—	—	—
		IMGCNT_ MTX_B[7]	IMGCNT_ MTX_B[6]	IMGCNT_ MTX_B[5]	IMGCNT_ MTX_B[4]	IMGCNT_ MTX_B[3]	IMGCNT_ MTX_B[2]	IMGCNT_ MTX_B[1]	IMGCNT_ MTX_B[0]
		—	—	—	—	—	IMGCNT_ MTX_BG[10]	IMGCNT_ MTX_BG[9]	IMGCNT_ MTX_BG[8]
	IMGCNT_ MTX_CBB_ ADJ1	IMGCNT_ MTX_BG[7]	IMGCNT_ MTX_BG[6]	IMGCNT_ MTX_BG[5]	IMGCNT_ MTX_BG[4]	IMGCNT_ MTX_BG[3]	IMGCNT_ MTX_BG[2]	IMGCNT_ MTX_BG[1]	IMGCNT_ MTX_BG[0]
		—	—	—	—	—	IMGCNT_ MTX_BB[10]	IMGCNT_ MTX_BB[9]	IMGCNT_ MTX_BB[8]
		IMGCNT_ MTX_BB[7]	IMGCNT_ MTX_BB[6]	IMGCNT_ MTX_BB[5]	IMGCNT_ MTX_BB[4]	IMGCNT_ MTX_BB[3]	IMGCNT_ MTX_BB[2]	IMGCNT_ MTX_BB[1]	IMGCNT_ MTX_BB[0]
	IMGCNT_ MTX_CBB_ ADJ0	—	—	—	—	—	—	—	—
		IMGCNT_ MTX_BR[7]	IMGCNT_ MTX_BR[6]	IMGCNT_ MTX_BR[5]	IMGCNT_ MTX_BR[4]	IMGCNT_ MTX_BR[3]	IMGCNT_ MTX_BR[2]	IMGCNT_ MTX_BR[1]	IMGCNT_ MTX_BR[0]
		—	—	—	—	—	IMGCNT_ MTX_BR[10]	IMGCNT_ MTX_BR[9]	IMGCNT_ MTX_BR[8]
	IMGCNT_ MTX_CRR_ ADJ1	IMGCNT_ MTX_R[7]	IMGCNT_ MTX_R[6]	IMGCNT_ MTX_R[5]	IMGCNT_ MTX_R[4]	IMGCNT_ MTX_R[3]	IMGCNT_ MTX_R[2]	IMGCNT_ MTX_R[1]	IMGCNT_ MTX_R[0]
		—	—	—	—	—	IMGCNT_ MTX_RG[10]	IMGCNT_ MTX_RG[9]	IMGCNT_ MTX_RG[8]
		IMGCNT_ MTX_RG[7]	IMGCNT_ MTX_RG[6]	IMGCNT_ MTX_RG[5]	IMGCNT_ MTX_RG[4]	IMGCNT_ MTX_RG[3]	IMGCNT_ MTX_RG[2]	IMGCNT_ MTX_RG[1]	IMGCNT_ MTX_RG[0]
	IMGCNT_ MTX_CRR_ ADJ0	—	—	—	—	—	—	—	—
		IMGCNT_ MTX_RB[7]	IMGCNT_ MTX_RB[6]	IMGCNT_ MTX_RB[5]	IMGCNT_ MTX_RB[4]	IMGCNT_ MTX_RB[3]	IMGCNT_ MTX_RB[2]	IMGCNT_ MTX_RB[1]	IMGCNT_ MTX_RB[0]
		—	—	—	—	—	IMGCNT_ MTX_RR[10]	IMGCNT_ MTX_RR[9]	IMGCNT_ MTX_RR[8]
	SCL0_ UPDATE	IMGCNT_ MTX_RR[7]	IMGCNT_ MTX_RR[6]	IMGCNT_ MTX_RR[5]	IMGCNT_ MTX_RR[4]	IMGCNT_ MTX_RR[3]	IMGCNT_ MTX_RR[2]	IMGCNT_ MTX_RR[1]	IMGCNT_ MTX_RR[0]
		—	—	—	—	—	—	—	—
		—	—	SCL0_VEN_D	SCL0_VEN_C	—	—	—	SCL0_UPDATE
	SCL0_FRC1	—	—	—	SCL0_VEN_B	—	—	—	SCL0_VEN_A
		RES_VMASK [15]	RES_VMASK [14]	RES_VMASK [13]	RES_VMASK [12]	RES_VMASK [11]	RES_VMASK [10]	RES_VMASK [9]	RES_VMASK [8]
		RES_VMASK [7]	RES_VMASK [6]	RES_VMASK [5]	RES_VMASK [4]	RES_VMASK [3]	RES_VMASK [2]	RES_VMASK [1]	RES_VMASK [0]
		—	—	—	—	—	—	—	—
									RES_ VMASK_ON

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
视频显示 控制器 4	SCL0_FRC2	RES_VLACK[15]	RES_VLACK[14]	RES_VLACK[13]	RES_VLACK[12]	RES_VLACK[11]	RES_VLACK[10]	RES_VLACK[9]	RES_VLACK[8]	
		RES_VLACK[7]	RES_VLACK[6]	RES_VLACK[5]	RES_VLACK[4]	RES_VLACK[3]	RES_VLACK[2]	RES_VLACK[1]	RES_VLACK[0]	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	RES_VLACK_ON	
	SCL0_FRC3	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	RES_VS_SEL	
	SCL0_FRC4	—	—	—	—	—	—	RES_FV[10]	RES_FV[9]	RES_FV[8]
		RES_FV[7]	RES_FV[6]	RES_FV[5]	RES_FV[4]	RES_FV[3]	RES_FV[2]	RES_FV[1]	RES_FV[0]	
		—	—	—	—	—	—	RES_FH[10]	RES_FH[9]	RES_FH[8]
		RES_FH[7]	RES_FH[6]	RES_FH[5]	RES_FH[4]	RES_FH[3]	RES_FH[2]	RES_FH[1]	RES_FH[0]	
	SCL0_FRC5	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	RES_FLD_DLY_SEL	
		RES_VSDLY[7]	RES_VSDLY[6]	RES_VSDLY[5]	RES_VSDLY[4]	RES_VSDLY[3]	RES_VSDLY[2]	RES_VSDLY[1]	RES_VSDLY[0]	
	SCL0_FRC6	—	—	—	—	—	—	RES_F_VS[10]	RES_F_VS[9]	RES_F_VS[8]
		RES_F_VS[7]	RES_F_VS[6]	RES_F_VS[5]	RES_F_VS[4]	RES_F_VS[3]	RES_F_VS[2]	RES_F_VS[1]	RES_F_VS[0]	
		—	—	—	—	—	—	RES_F_VW[10]	RES_F_VW[9]	RES_F_VW[8]
		RES_F_VW[7]	RES_F_VW[6]	RES_F_VW[5]	RES_F_VW[4]	RES_F_VW[3]	RES_F_VW[2]	RES_F_VW[1]	RES_F_VW[0]	
	SCL0_FRC7	—	—	—	—	—	—	RES_F_HS[10]	RES_F_HS[9]	RES_F_HS[8]
		RES_F_HS[7]	RES_F_HS[6]	RES_F_HS[5]	RES_F_HS[4]	RES_F_HS[3]	RES_F_HS[2]	RES_F_HS[1]	RES_F_HS[0]	
		—	—	—	—	—	—	RES_F_HW[10]	RES_F_HW[9]	RES_F_HW[8]
		RES_F_HW[7]	RES_F_HW[6]	RES_F_HW[5]	RES_F_HW[4]	RES_F_HW[3]	RES_F_HW[2]	RES_F_HW[1]	RES_F_HW[0]	
	SCL0_FRC9	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	RES_QVLOCK	—	—	—	RES_QVLACK	
	SCL0_DS1	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	RES_DS_V_ON	—	—	—	RES_DS_H_ON	
	SCL0_DS2	—	—	—	—	—	—	RES_VS[10]	RES_VS[9]	RES_VS[8]
		RES_VS[7]	RES_VS[6]	RES_VS[5]	RES_VS[4]	RES_VS[3]	RES_VS[2]	RES_VS[1]	RES_VS[0]	
		—	—	—	—	—	—	RES_VW[10]	RES_VW[9]	RES_VW[8]
		RES_VW[7]	RES_VW[6]	RES_VW[5]	RES_VW[4]	RES_VW[3]	RES_VW[2]	RES_VW[1]	RES_VW[0]	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	SCL0_DS3	—	—	—	—	—	RES_HS[10]	RES_HS[9]	RES_HS[8]
		RES_HS[7]	RES_HS[6]	RES_HS[5]	RES_HS[4]	RES_HS[3]	RES_HS[2]	RES_HS[1]	RES_HS[0]
		—	—	—	—	—	RES_HW[10]	RES_HW[9]	RES_HW[8]
		RES_HW[7]	RES_HW[6]	RES_HW[5]	RES_HW[4]	RES_HW[3]	RES_HW[2]	RES_HW[1]	RES_HW[0]
	SCL0_DS4	—	—	RES_PFIL_	RES_DS_H_	—	—	—	—
		—	—	SEL	INTERPOTYP	—	—	—	—
		RES_DS_H_	RES_DS_H_	RES_DS_H_	RES_DS_H_	RES_DS_H_	RES_DS_H_	RES_DS_H_	RES_DS_H_
		RATIO[15]	RATIO[14]	RATIO[13]	RATIO[12]	RATIO[11]	RATIO[10]	RATIO[9]	RATIO[8]
	RES_DS_H_	RES_DS_H_	RES_DS_H_	RES_DS_H_	RES_DS_H_	RES_DS_H_	RES_DS_H_	RES_DS_H_	
	RATIO[7]	RATIO[6]	RATIO[5]	RATIO[4]	RATIO[3]	RATIO[2]	RATIO[1]	RATIO[0]	
	SCL0_DS5	—	—	—	RES_V_	RES_TOP_	RES_TOP_	RES_TOP_	RES_TOP_
		—	—	—	INTERPOTYP	INIPHASE[11]	INIPHASE[10]	INIPHASE[9]	INIPHASE[8]
		RES_TOP_	RES_TOP_	RES_TOP_	RES_TOP_	RES_TOP_	RES_TOP_	RES_TOP_	RES_TOP_
		INIPHASE[7]	INIPHASE[6]	INIPHASE[5]	INIPHASE[4]	INIPHASE[3]	INIPHASE[2]	INIPHASE[1]	INIPHASE[0]
	—	—	—	—	RES_BTM_	RES_BTM_	RES_BTM_	RES_BTM_	
	RES_BTM_	RES_BTM_	RES_BTM_	RES_BTM_	RES_BTM_	RES_BTM_	RES_BTM_	RES_BTM_	
	INIPHASE[7]	INIPHASE[6]	INIPHASE[5]	INIPHASE[4]	INIPHASE[3]	INIPHASE[2]	INIPHASE[1]	INIPHASE[0]	
	SCL0_DS6	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		RES_V_	RES_V_	RES_V_	RES_V_	RES_V_	RES_V_	RES_V_	RES_V_
		RATIO[15]	RATIO[14]	RATIO[13]	RATIO[12]	RATIO[11]	RATIO[10]	RATIO[9]	RATIO[8]
	RES_V_	RES_V_	RES_V_	RES_V_	RES_V_	RES_V_	RES_V_	RES_V_	
	RATIO[7]	RATIO[6]	RATIO[5]	RATIO[4]	RATIO[3]	RATIO[2]	RATIO[1]	RATIO[0]	
	SCL0_DS7	—	—	—	—	—	RES_OUT_	RES_OUT_	RES_OUT_
—		—	—	—	—	VW[10]	VW[9]	VW[8]	
RES_OUT_		RES_OUT_	RES_OUT_	RES_OUT_	RES_OUT_	RES_OUT_	RES_OUT_	RES_OUT_	
VW[7]		VW[6]	VW[5]	VW[4]	VW[3]	VW[2]	VW[1]	VW[0]	
—	—	—	—	—	RES_OUT_	RES_OUT_	RES_OUT_		
RES_OUT_	RES_OUT_	RES_OUT_	RES_OUT_	RES_OUT_	RES_OUT_	RES_OUT_	RES_OUT_		
HW[7]	HW[6]	HW[5]	HW[4]	HW[3]	HW[2]	HW[1]	HW[0]		
SCL0_US1	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	
	—	—	—	RES_US_V_	—	—	—	RES_US_H_	
	—	—	—	ON	—	—	—	ON	
	SCL0_US2	—	—	—	—	—	RES_P_VS[10]	RES_P_VS[9]	RES_P_VS[8]
		RES_P_VS[7]	RES_P_VS[6]	RES_P_VS[5]	RES_P_VS[4]	RES_P_VS[3]	RES_P_VS[2]	RES_P_VS[1]	RES_P_VS[0]
		—	—	—	—	—	RES_P_VW[10]	RES_P_VW[9]	RES_P_VW[8]
		RES_P_VW[7]	RES_P_VW[6]	RES_P_VW[5]	RES_P_VW[4]	RES_P_VW[3]	RES_P_VW[2]	RES_P_VW[1]	RES_P_VW[0]
	SCL0_US3	—	—	—	—	—	RES_P_HS[10]	RES_P_HS[9]	RES_P_HS[8]
		RES_P_HS[7]	RES_P_HS[6]	RES_P_HS[5]	RES_P_HS[4]	RES_P_HS[3]	RES_P_HS[2]	RES_P_HS[1]	RES_P_HS[0]
		—	—	—	—	—	RES_P_	RES_P_	RES_P_
RES_P_HW[7]		RES_P_HW[6]	RES_P_HW[5]	RES_P_HW[4]	RES_P_HW[3]	RES_P_HW[2]	RES_P_HW[1]	RES_P_HW[0]	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	SCL0_US4	—	—	—	—	—	RES_IN_VW[10]	RES_IN_VW[9]	RES_IN_VW[8]
		RES_IN_VW[7]	RES_IN_VW[6]	RES_IN_VW[5]	RES_IN_VW[4]	RES_IN_VW[3]	RES_IN_VW[2]	RES_IN_VW[1]	RES_IN_VW[0]
		—	—	—	—	—	RES_IN_HW[10]	RES_IN_HW[9]	RES_IN_HW[8]
		RES_IN_HW[7]	RES_IN_HW[6]	RES_IN_HW[5]	RES_IN_HW[4]	RES_IN_HW[3]	RES_IN_HW[2]	RES_IN_HW[1]	RES_IN_HW[0]
	SCL0_US5	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		RES_US_H_RATIO[15]	RES_US_H_RATIO[14]	RES_US_H_RATIO[13]	RES_US_H_RATIO[12]	RES_US_H_RATIO[11]	RES_US_H_RATIO[10]	RES_US_H_RATIO[9]	RES_US_H_RATIO[8]
		RES_US_H_RATIO[7]	RES_US_H_RATIO[6]	RES_US_H_RATIO[5]	RES_US_H_RATIO[4]	RES_US_H_RATIO[3]	RES_US_H_RATIO[2]	RES_US_H_RATIO[1]	RES_US_H_RATIO[0]
	SCL0_US6	—	—	—	RES_US_H_INTERPOTYP	RES_US_HT_INIPHASE[11]	RES_US_HT_INIPHASE[10]	RES_US_HT_INIPHASE[9]	RES_US_HT_INIPHASE[8]
		RES_US_HT_INIPHASE[7]	RES_US_HT_INIPHASE[6]	RES_US_HT_INIPHASE[5]	RES_US_HT_INIPHASE[4]	RES_US_HT_INIPHASE[3]	RES_US_HT_INIPHASE[2]	RES_US_HT_INIPHASE[1]	RES_US_HT_INIPHASE[0]
		—	—	—	—	RES_US_HB_INIPHASE[11]	RES_US_HB_INIPHASE[10]	RES_US_HB_INIPHASE[9]	RES_US_HB_INIPHASE[8]
		RES_US_HB_INIPHASE[7]	RES_US_HB_INIPHASE[6]	RES_US_HB_INIPHASE[5]	RES_US_HB_INIPHASE[4]	RES_US_HB_INIPHASE[3]	RES_US_HB_INIPHASE[2]	RES_US_HB_INIPHASE[1]	RES_US_HB_INIPHASE[0]
	SCL0_US7	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		RES_HCURT [7]	RES_HCURT [6]	RES_HCURT [5]	RES_HCURT [4]	RES_HCURT [3]	RES_HCURT [2]	RES_HCURT [1]	RES_HCURT [0]
		RES_VCURT [7]	RES_VCURT [6]	RES_VCURT [5]	RES_VCURT [4]	RES_VCURT [3]	RES_VCURT [2]	RES_VCURT [1]	RES_VCURT [0]
	SCL0_US8	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	RES_IBUS_SYNC_SEL	—	—	—	RES_DISP_ON
	SCL0_OVR1	—	—	—	—	—	—	—	—
		RES_BK_COL_R[7]	RES_BK_COL_R[6]	RES_BK_COL_R[5]	RES_BK_COL_R[4]	RES_BK_COL_R[3]	RES_BK_COL_R[2]	RES_BK_COL_R[1]	RES_BK_COL_R[0]
		RES_BK_COL_G[7]	RES_BK_COL_G[6]	RES_BK_COL_G[5]	RES_BK_COL_G[4]	RES_BK_COL_G[3]	RES_BK_COL_G[2]	RES_BK_COL_G[1]	RES_BK_COL_G[0]
		RES_BK_COL_B[7]	RES_BK_COL_B[6]	RES_BK_COL_B[5]	RES_BK_COL_B[4]	RES_BK_COL_B[3]	RES_BK_COL_B[2]	RES_BK_COL_B[1]	RES_BK_COL_B[0]
	SCL1_UPDATE	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	SCL1_VEN_B	—	—	—	SCL1_VEN_A
	SCL1_WR1	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	RES_DS_WR_MD[2]	RES_DS_WR_MD[1]	RES_DS_WR_MD[0]	RES_MD[1]	RES_MD[0]	RES_LOOP	RES_BST_MD

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
视频显示 控制器 4	SCL1_WR2	RES_	RES_	RES_	RES_	RES_	RES_	RES_	RES_	
		BASE[31]	BASE[30]	BASE[29]	BASE[28]	BASE[27]	BASE[26]	BASE[25]	BASE[24]	
		RES_	RES_	RES_	RES_	RES_	RES_	RES_	RES_	
		BASE[23]	BASE[22]	BASE[21]	BASE[20]	BASE[19]	BASE[18]	BASE[17]	BASE[16]	
	RES_	RES_	RES_	RES_	RES_	RES_	RES_	RES_	RES_	
	BASE[15]	BASE[14]	BASE[13]	BASE[12]	BASE[11]	BASE[10]	BASE[9]	BASE[8]	BASE[8]	
	RES_	RES_	RES_	RES_	RES_	RES_	RES_	RES_	RES_	
	BASE[7]	BASE[6]	BASE[5]	BASE[4]	BASE[3]	BASE[2]	BASE[1]	BASE[0]	BASE[0]	
	SCL1_WR3	—	RES_LN_	RES_LN_	RES_LN_	RES_LN_	RES_LN_	RES_LN_	RES_LN_	RES_LN_
		—	OFF[14]	OFF[13]	OFF[12]	OFF[11]	OFF[10]	OFF[9]	OFF[8]	OFF[8]
		RES_LN_	RES_LN_	RES_LN_	RES_LN_	RES_LN_	RES_LN_	RES_LN_	RES_LN_	RES_LN_
		OFF[7]	OFF[6]	OFF[5]	OFF[4]	OFF[3]	OFF[2]	OFF[1]	OFF[0]	OFF[0]
	—	—	—	—	—	—	RES_FLM_	RES_FLM_	RES_FLM_	
	—	—	—	—	—	—	NUM[9]	NUM[8]	NUM[8]	
	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	
	NUM[7]	NUM[6]	NUM[5]	NUM[4]	NUM[3]	NUM[2]	NUM[1]	NUM[0]	NUM[0]	
	SCL1_WR4	—	—	—	—	—	—	—	—	—
		—	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_
		—	OFF[22]	OFF[21]	OFF[20]	OFF[19]	OFF[18]	OFF[17]	OFF[16]	OFF[16]
		RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_
	OFF[15]	OFF[14]	OFF[13]	OFF[12]	OFF[11]	OFF[10]	OFF[9]	OFF[8]	OFF[8]	
	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	
	OFF[7]	OFF[6]	OFF[5]	OFF[4]	OFF[3]	OFF[2]	OFF[1]	OFF[0]	OFF[0]	
	SCL1_WR5	—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		—	—	—	RES_INTER	—	—	RES_FS_	RES_FS_	RES_FS_
		—	—	—	—	—	—	RATE[1]	RATE[0]	RATE[0]
	—	—	—	RES_FLD_	—	—	—	—	RES_WENB	
	—	—	—	SEL	—	—	—	—	—	
	SCL1_WR6	—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		—	—	—	RES_DTH_	—	—	—	—	RES_
	—	—	—	ON	—	—	—	—	BITDEC_ON	
	SCL1_WR7	—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	RES_OVER
		—	—	—	—	—	—	RES_FLM_	RES_FLM_	RES_FLM_
		RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_	RES_FLM_
	CNT[7]	CNT[6]	CNT[5]	CNT[4]	CNT[3]	CNT[2]	CNT[1]	CNT[0]	CNT[0]	
	GR1_	—	—	—	—	—	—	—	—	—
		UPDATE	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		—	—	—	GR1_P_VEN	—	—	—	—	GR1_IBUS_
	—	—	—	—	—	—	—	—	VEN	
GR1_FLM_	—	—	—	—	—	—	—	—	—	
	RD	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	—	
	—	—	—	—	—	—	—	—	GR1_R_ENB	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	GR1_FLM1	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	GR1_LN_ OFF_DIR
		—	—	—	—	—	—	GR1_FLM_ SEL[1]	GR1_FLM_ SEL[0]
		—	—	—	GR1_IMR_ FLM_INV	—	—	—	GR1_BST_ _MD
	GR1_FLM2	GR1_ BASE[31]	GR1_ BASE[30]	GR1_ BASE[29]	GR1_ BASE[28]	GR1_ BASE[27]	GR1_ BASE[26]	GR1_ BASE[25]	GR1_ BASE[24]
		GR1_ BASE[23]	GR1_ BASE[22]	GR1_ BASE[21]	GR1_ BASE[20]	GR1_ BASE[19]	GR1_ BASE[18]	GR1_ BASE[17]	GR1_ BASE[16]
		GR1_ BASE[15]	GR1_ BASE[14]	GR1_ BASE[13]	GR1_ BASE[12]	GR1_ BASE[11]	GR1_ BASE[10]	GR1_ BASE[9]	GR1_ BASE[8]
		GR1_ BASE[7]	GR1_ BASE[6]	GR1_ BASE[5]	GR1_ BASE[4]	GR1_ BASE[3]	GR1_ BASE[2]	GR1_ BASE[1]	GR1_ BASE[0]
	GR1_FLM3	—	GR1_LN_ OFF[14]	GR1_LN_ OFF[13]	GR1_LN_ OFF[12]	GR1_LN_ OFF[11]	GR1_LN_ OFF[10]	GR1_LN_ OFF[9]	GR1_LN_ OFF[8]
		GR1_LN_ OFF[7]	GR1_LN_ OFF[6]	GR1_LN_ OFF[5]	GR1_LN_ OFF[4]	GR1_LN_ OFF[3]	GR1_LN_ OFF[2]	GR1_LN_ OFF[1]	GR1_LN_ OFF[0]
		—	—	—	—	—	—	GR1_FLM_ NUM[9]	GR1_FLM_ NUM[8]
		GR1_FLM_ NUM[7]	GR1_FLM_ NUM[6]	GR1_FLM_ NUM[5]	GR1_FLM_ NUM[4]	GR1_FLM_ NUM[3]	GR1_FLM_ NUM[2]	GR1_FLM_ NUM[1]	GR1_FLM_ NUM[0]
	GR1_FLM4	—	—	—	—	—	—	—	—
		—	GR1_FLM_ OFF[22]	GR1_FLM_ OFF[21]	GR1_FLM_ OFF[20]	GR1_FLM_ OFF[19]	GR1_FLM_ OFF[18]	GR1_FLM_ OFF[17]	GR1_FLM_ OFF[16]
		GR1_FLM_ OFF[15]	GR1_FLM_ OFF[14]	GR1_FLM_ OFF[13]	GR1_FLM_ OFF[12]	GR1_FLM_ OFF[11]	GR1_FLM_ OFF[10]	GR1_FLM_ OFF[9]	GR1_FLM_ OFF[8]
		GR1_FLM_ OFF[7]	GR1_FLM_ OFF[6]	GR1_FLM_ OFF[5]	GR1_FLM_ OFF[4]	GR1_FLM_ OFF[3]	GR1_FLM_ OFF[2]	GR1_FLM_ OFF[1]	GR1_FLM_ OFF[0]
	GR1_FLM5	—	—	—	—	—	—	—	GR1_FLM_ LNUM[9]
		GR1_FLM_ LNUM[7]	GR1_FLM_ LNUM[6]	GR1_FLM_ LNUM[5]	GR1_FLM_ LNUM[4]	GR1_FLM_ LNUM[3]	GR1_FLM_ LNUM[2]	GR1_FLM_ LNUM[1]	GR1_FLM_ LNUM[0]
		—	—	—	—	—	—	GR1_FLM_ LOOP[9]	GR1_FLM_ LOOP[8]
		GR1_FLM_ LOOP[7]	GR1_FLM_ LOOP[6]	GR1_FLM_ LOOP[5]	GR1_FLM_ LOOP[4]	GR1_FLM_ LOOP[3]	GR1_FLM_ LOOP[2]	GR1_FLM_ LOOP[1]	GR1_FLM_ LOOP[0]
	GR1_FLM6	GR1_ FORMAT[3]	GR1_ FORMAT[2]	GR1_ FORMAT[1]	GR1_ FORMAT[0]	—	—	GR1_HW[9]	GR1_HW[8]
		GR1_HW[7]	GR1_HW[6]	GR1_HW[5]	GR1_HW[4]	GR1_HW[3]	GR1_HW[2]	GR1_HW[1]	GR1_HW[0]
		GR1_YCC_ SWAP[2]	GR1_YCC_ SWAP[1]	GR1_YCC_ SWAP[0]	GR1_ ENDIAN_ON	—	—	—	GR1_ CNV444_MD
		—	—	GR1_STA_ POS[5]	GR1_STA_ POS[4]	GR1_STA_ POS[3]	GR1_STA_ POS[2]	GR1_STA_ POS[1]	GR1_STA_ POS[0]
	GR1_AB1	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	GR1_GRC_ DISP_ON	—	—	GR1_DISP_ SEL[1]	GR1_DISP_ SEL[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	GR1_AB2	—	—	—	—	—	GR1_GRC_ VS[10]	GR1_GRC_ VS[9]	GR1_GRC_ VS[8]
		GR1_GRC_ VS[7]	GR1_GRC_ VS[6]	GR1_GRC_ VS[5]	GR1_GRC_ VS[4]	GR1_GRC_ VS[3]	GR1_GRC_ VS[2]	GR1_GRC_ VS[1]	GR1_GRC_ VS[0]
		—	—	—	—	—	GR1_GRC_ VW[10]	GR1_GRC_ VW[9]	GR1_GRC_ VW[8]
		GR1_GRC_ VW[7]	GR1_GRC_ VW[6]	GR1_GRC_ VW[5]	GR1_GRC_ VW[4]	GR1_GRC_ VW[3]	GR1_GRC_ VW[2]	GR1_GRC_ VW[1]	GR1_GRC_ VW[0]
	GR1_AB3	—	—	—	—	—	GR1_GRC_ HS[10]	GR1_GRC_ HS[9]	GR1_GRC_ HS[8]
		GR1_GRC_ HS[7]	GR1_GRC_ HS[6]	GR1_GRC_ HS[5]	GR1_GRC_ HS[4]	GR1_GRC_ HS[3]	GR1_GRC_ HS[2]	GR1_GRC_ HS[1]	GR1_GRC_ HS[0]
		—	—	—	—	—	GR1_GRC_ HW[10]	GR1_GRC_ HW[9]	GR1_GRC_ HW[8]
		GR1_GRC_ HW[7]	GR1_GRC_ HW[6]	GR1_GRC_ HW[5]	GR1_GRC_ HW[4]	GR1_GRC_ HW[3]	GR1_GRC_ HW[2]	GR1_GRC_ HW[1]	GR1_GRC_ HW[0]
	GR1_AB7	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	GR1_CK_ON
	GR1_AB8	GR1_CK_ KCLUT[7]	GR1_CK_ KCLUT[6]	GR1_CK_ KCLUT[5]	GR1_CK_ KCLUT[4]	GR1_CK_ KCLUT[3]	GR1_CK_ KCLUT[2]	GR1_CK_ KCLUT[1]	GR1_CK_ KCLUT[0]
		GR1_CK_ KG[7]	GR1_CK_ KG[6]	GR1_CK_ KG[5]	GR1_CK_ KG[4]	GR1_CK_ KG[3]	GR1_CK_ KG[2]	GR1_CK_ KG[1]	GR1_CK_ KG[0]
		GR1_CK_ KB[7]	GR1_CK_ KB[6]	GR1_CK_ KB[5]	GR1_CK_ KB[4]	GR1_CK_ KB[3]	GR1_CK_ KB[2]	GR1_CK_ KB[1]	GR1_CK_ KB[0]
		GR1_CK_ KR[7]	GR1_CK_ KR[6]	GR1_CK_ KR[5]	GR1_CK_ KR[4]	GR1_CK_ KR[3]	GR1_CK_ KR[2]	GR1_CK_ KR[1]	GR1_CK_ KR[0]
	GR1_AB9	GR1_CK_ A[7]	GR1_CK_ A[6]	GR1_CK_ A[5]	GR1_CK_ A[4]	GR1_CK_ A[3]	GR1_CK_ A[2]	GR1_CK_ A[1]	GR1_CK_ A[0]
		GR1_CK_ G[7]	GR1_CK_ G[6]	GR1_CK_ G[5]	GR1_CK_ G[4]	GR1_CK_ G[3]	GR1_CK_ G[2]	GR1_CK_ G[1]	GR1_CK_ G[0]
		GR1_CK_ B[7]	GR1_CK_ B[6]	GR1_CK_ B[5]	GR1_CK_ B[4]	GR1_CK_ B[3]	GR1_CK_ B[2]	GR1_CK_ B[1]	GR1_CK_ B[0]
		GR1_CK_ R[7]	GR1_CK_ R[6]	GR1_CK_ R[5]	GR1_CK_ R[4]	GR1_CK_ R[3]	GR1_CK_ R[2]	GR1_CK_ R[1]	GR1_CK_ R[0]
	GR1_AB10	GR1_A0[7]	GR1_A0[6]	GR1_A0[5]	GR1_A0[4]	GR1_A0[3]	GR1_A0[2]	GR1_A0[1]	GR1_A0[0]
		GR1_G0[7]	GR1_G0[6]	GR1_G0[5]	GR1_G0[4]	GR1_G0[3]	GR1_G0[2]	GR1_G0[1]	GR1_G0[0]
		GR1_B0[7]	GR1_B0[6]	GR1_B0[5]	GR1_B0[4]	GR1_B0[3]	GR1_B0[2]	GR1_B0[1]	GR1_B0[0]
		GR1_R0[7]	GR1_R0[6]	GR1_R0[5]	GR1_R0[4]	GR1_R0[3]	GR1_R0[2]	GR1_R0[1]	GR1_R0[0]
	GR1_AB11	GR1_A1[7]	GR1_A1[6]	GR1_A1[5]	GR1_A1[4]	GR1_A1[3]	GR1_A1[2]	GR1_A1[1]	GR1_A1[0]
		GR1_G1[7]	GR1_G1[6]	GR1_G1[5]	GR1_G1[4]	GR1_G1[3]	GR1_G1[2]	GR1_G1[1]	GR1_G1[0]
		GR1_B1[7]	GR1_B1[6]	GR1_B1[5]	GR1_B1[4]	GR1_B1[3]	GR1_B1[2]	GR1_B1[1]	GR1_B1[0]
		GR1_R1[7]	GR1_R1[6]	GR1_R1[5]	GR1_R1[4]	GR1_R1[3]	GR1_R1[2]	GR1_R1[1]	GR1_R1[0]
	GR1_BASE	—	—	—	—	—	—	—	—
		GR1_BASE_ G[7]	GR1_BASE_ G[6]	GR1_BASE_ G[5]	GR1_BASE_ G[4]	GR1_BASE_ G[3]	GR1_BASE_ G[2]	GR1_BASE_ G[1]	GR1_BASE_ G[0]
		GR1_BASE_ B[7]	GR1_BASE_ B[6]	GR1_BASE_ B[5]	GR1_BASE_ B[4]	GR1_BASE_ B[3]	GR1_BASE_ B[2]	GR1_BASE_ B[1]	GR1_BASE_ B[0]
		GR1_BASE_ R[7]	GR1_BASE_ R[6]	GR1_BASE_ R[5]	GR1_BASE_ R[4]	GR1_BASE_ R[3]	GR1_BASE_ R[2]	GR1_BASE_ R[1]	GR1_BASE_ R[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	GR1_CLUT	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	GR1_CLT_SEL
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
	ADJ_UPDATE	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	ADJ_VEN
	ADJ_BKSTR_SET	—	—	—	—	—	—	—	BKSTR_ON
		BKSTR_ST[3]	BKSTR_ST[2]	BKSTR_ST[1]	BKSTR_ST[0]	BKSTR_D[3]	BKSTR_D[2]	BKSTR_D[1]	BKSTR_D[0]
		—	—	—	BKSTR_T1[4]	BKSTR_T1[3]	BKSTR_T1[2]	BKSTR_T1[1]	BKSTR_T1[0]
	ADJ_ENH_TIM1	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	ENH_MD	—	—	—	ENH_DISP_ON
	ADJ_ENH_TIM2	—	—	—	—	—	ENH_VS[10]	ENH_VS[9]	ENH_VS[8]
		ENH_VS[7]	ENH_VS[6]	ENH_VS[5]	ENH_VS[4]	ENH_VS[3]	ENH_VS[2]	ENH_VS[1]	ENH_VS[0]
		—	—	—	—	—	ENH_VW[10]	ENH_VW[9]	ENH_VW[8]
		ENH_VW[7]	ENH_VW[6]	ENH_VW[5]	ENH_VW[4]	ENH_VW[3]	ENH_VW[2]	ENH_VW[1]	ENH_VW[0]
	ADJ_ENH_TIM3	—	—	—	—	—	ENH_HS[10]	ENH_HS[9]	ENH_HS[8]
		ENH_HS[7]	ENH_HS[6]	ENH_HS[5]	ENH_HS[4]	ENH_HS[3]	ENH_HS[2]	ENH_HS[1]	ENH_HS[0]
		—	—	—	—	—	ENH_HW[10]	ENH_HW[9]	ENH_HW[8]
		ENH_HW[7]	ENH_HW[6]	ENH_HW[5]	ENH_HW[4]	ENH_HW[3]	ENH_HW[2]	ENH_HW[1]	ENH_HW[0]
	ADJ_ENH_SHP1	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	SHP_H_ON
		—	—	—	—	—	—	—	—
		—	SHP_H1_CORE[6]	SHP_H1_CORE[5]	SHP_H1_CORE[4]	SHP_H1_CORE[3]	SHP_H1_CORE[2]	SHP_H1_CORE[1]	SHP_H1_CORE[0]
	ADJ_ENH_SHP2	SHP_H1_CLIP_O[7]	SHP_H1_CLIP_O[6]	SHP_H1_CLIP_O[5]	SHP_H1_CLIP_O[4]	SHP_H1_CLIP_O[3]	SHP_H1_CLIP_O[2]	SHP_H1_CLIP_O[1]	SHP_H1_CLIP_O[0]
		SHP_H1_CLIP_U[7]	SHP_H1_CLIP_U[6]	SHP_H1_CLIP_U[5]	SHP_H1_CLIP_U[4]	SHP_H1_CLIP_U[3]	SHP_H1_CLIP_U[2]	SHP_H1_CLIP_U[1]	SHP_H1_CLIP_U[0]
		SHP_H1_GAIN_O[7]	SHP_H1_GAIN_O[6]	SHP_H1_GAIN_O[5]	SHP_H1_GAIN_O[4]	SHP_H1_GAIN_O[3]	SHP_H1_GAIN_O[2]	SHP_H1_GAIN_O[1]	SHP_H1_GAIN_O[0]
		SHP_H1_GAIN_U[7]	SHP_H1_GAIN_U[6]	SHP_H1_GAIN_U[5]	SHP_H1_GAIN_U[4]	SHP_H1_GAIN_U[3]	SHP_H1_GAIN_U[2]	SHP_H1_GAIN_U[1]	SHP_H1_GAIN_U[0]
	ADJ_ENH_SHP3	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	SHP_H2_LPF_SEL
		—	—	—	—	—	—	—	—
		—	SHP_H2_CORE[6]	SHP_H2_CORE[5]	SHP_H2_CORE[4]	SHP_H2_CORE[3]	SHP_H2_CORE[2]	SHP_H2_CORE[1]	SHP_H2_CORE[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
视频显示 控制器 4	ADJ_ENH_ SHP4	SHP_H2_ CLIP_O[7]	SHP_H2_ CLIP_O[6]	SHP_H2_ CLIP_O[5]	SHP_H2_ CLIP_O[4]	SHP_H2_ CLIP_O[3]	SHP_H2_ CLIP_O[2]	SHP_H2_ CLIP_O[1]	SHP_H2_ CLIP_O[0]	
		SHP_H2_ CLIP_U[7]	SHP_H2_ CLIP_U[6]	SHP_H2_ CLIP_U[5]	SHP_H2_ CLIP_U[4]	SHP_H2_ CLIP_U[3]	SHP_H2_ CLIP_U[2]	SHP_H2_ CLIP_U[1]	SHP_H2_ CLIP_U[0]	
		SHP_H2_ GAIN_O[7]	SHP_H2_ GAIN_O[6]	SHP_H2_ GAIN_O[5]	SHP_H2_ GAIN_O[4]	SHP_H2_ GAIN_O[3]	SHP_H2_ GAIN_O[2]	SHP_H2_ GAIN_O[1]	SHP_H2_ GAIN_O[0]	
		SHP_H2_ GAIN_U[7]	SHP_H2_ GAIN_U[6]	SHP_H2_ GAIN_U[5]	SHP_H2_ GAIN_U[4]	SHP_H2_ GAIN_U[3]	SHP_H2_ GAIN_U[2]	SHP_H2_ GAIN_U[1]	SHP_H2_ GAIN_U[0]	
	ADJ_ENH_ SHP5	—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		—	SHP_H3_ CORE[6]	SHP_H3_ CORE[5]	SHP_H3_ CORE[4]	SHP_H3_ CORE[3]	SHP_H3_ CORE[2]	SHP_H3_ CORE[1]	SHP_H3_ CORE[0]	—
	ADJ_ENH_ SHP6	SHP_H3_ CLIP_O[7]	SHP_H3_ CLIP_O[6]	SHP_H3_ CLIP_O[5]	SHP_H3_ CLIP_O[4]	SHP_H3_ CLIP_O[3]	SHP_H3_ CLIP_O[2]	SHP_H3_ CLIP_O[1]	SHP_H3_ CLIP_O[0]	—
		SHP_H3_ CLIP_U[7]	SHP_H3_ CLIP_U[6]	SHP_H3_ CLIP_U[5]	SHP_H3_ CLIP_U[4]	SHP_H3_ CLIP_U[3]	SHP_H3_ CLIP_U[2]	SHP_H3_ CLIP_U[1]	SHP_H3_ CLIP_U[0]	—
		SHP_H3_ GAIN_O[7]	SHP_H3_ GAIN_O[6]	SHP_H3_ GAIN_O[5]	SHP_H3_ GAIN_O[4]	SHP_H3_ GAIN_O[3]	SHP_H3_ GAIN_O[2]	SHP_H3_ GAIN_O[1]	SHP_H3_ GAIN_O[0]	—
		SHP_H3_ GAIN_U[7]	SHP_H3_ GAIN_U[6]	SHP_H3_ GAIN_U[5]	SHP_H3_ GAIN_U[4]	SHP_H3_ GAIN_U[3]	SHP_H3_ GAIN_U[2]	SHP_H3_ GAIN_U[1]	SHP_H3_ GAIN_U[0]	—
	ADJ_ENH_ LTI1	LTI_H_ON	—	—	—	—	—	—	—	LTI_H2_ LPF_SEL
		LTI_H2_INC ZERO[7]	LTI_H2_INC ZERO[6]	LTI_H2_INC ZERO[5]	LTI_H2_INC ZERO[4]	LTI_H2_INC ZERO[3]	LTI_H2_INC ZERO[2]	LTI_H2_INC ZERO[1]	LTI_H2_INC ZERO[0]	—
		LTI_H2_ GAIN[7]	LTI_H2_ GAIN[6]	LTI_H2_ GAIN[5]	LTI_H2_ GAIN[4]	LTI_H2_ GAIN[3]	LTI_H2_ GAIN[2]	LTI_H2_ GAIN[1]	LTI_H2_ GAIN[0]	—
		LTI_H2_ CORE[7]	LTI_H2_ CORE[6]	LTI_H2_ CORE[5]	LTI_H2_ CORE[4]	LTI_H2_ CORE[3]	LTI_H2_ CORE[2]	LTI_H2_ CORE[1]	LTI_H2_ CORE[0]	—
	ADJ_ENH_ LTI2	—	—	—	—	—	—	—	—	LTI_H4_MEDI AN_TAP_SEL
		LTI_H4_INC ZERO[7]	LTI_H4_INC ZERO[6]	LTI_H4_INC ZERO[5]	LTI_H4_INC ZERO[4]	LTI_H4_INC ZERO[3]	LTI_H4_INC ZERO[2]	LTI_H4_INC ZERO[1]	LTI_H4_INC ZERO[0]	—
		LTI_H4_ GAIN[7]	LTI_H4_ GAIN[6]	LTI_H4_ GAIN[5]	LTI_H4_ GAIN[4]	LTI_H4_ GAIN[3]	LTI_H4_ GAIN[2]	LTI_H4_ GAIN[1]	LTI_H4_ GAIN[0]	—
		LTI_H4_ CORE[7]	LTI_H4_ CORE[6]	LTI_H4_ CORE[5]	LTI_H4_ CORE[4]	LTI_H4_ CORE[3]	LTI_H4_ CORE[2]	LTI_H4_ CORE[1]	LTI_H4_ CORE[0]	—
	ADJ_MTX_ MODE	—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	ADJ_MTX_ MD[1]	ADJ_MTX_ MD[0]	—
	ADJ_MTX_ YG_ADJ0	—	—	—	—	—	—	—	—	—
		ADJ_MTX_ YG[7]	ADJ_MTX_ YG[6]	ADJ_MTX_ YG[5]	ADJ_MTX_ YG[4]	ADJ_MTX_ YG[3]	ADJ_MTX_ YG[2]	ADJ_MTX_ YG[1]	ADJ_MTX_ YG[0]	—
		—	—	—	—	—	ADJ_MTX_ GG[10]	ADJ_MTX_ GG[9]	ADJ_MTX_ GG[8]	—
		ADJ_MTX_ GG[7]	ADJ_MTX_ GG[6]	ADJ_MTX_ GG[5]	ADJ_MTX_ GG[4]	ADJ_MTX_ GG[3]	ADJ_MTX_ GG[2]	ADJ_MTX_ GG[1]	ADJ_MTX_ GG[0]	—

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
视频显示 控制器 4	ADJ_MTX_ YG_ADJ1	—	—	—	—	—	ADJ_MTX_ GB[10]	ADJ_MTX_ GB[9]	ADJ_MTX_ GB[8]	
		ADJ_MTX_ GB[7]	ADJ_MTX_ GB[6]	ADJ_MTX_ GB[5]	ADJ_MTX_ GB[4]	ADJ_MTX_ GB[3]	ADJ_MTX_ GB[2]	ADJ_MTX_ GB[1]	ADJ_MTX_ GB[0]	
		—	—	—	—	—	ADJ_MTX_ GR[10]	ADJ_MTX_ GR[9]	ADJ_MTX_ GR[8]	
		ADJ_MTX_ GR[7]	ADJ_MTX_ GR[6]	ADJ_MTX_ GR[5]	ADJ_MTX_ GR[4]	ADJ_MTX_ GR[3]	ADJ_MTX_ GR[2]	ADJ_MTX_ GR[1]	ADJ_MTX_ GR[0]	
	ADJ_MTX_ CBB_ADJ0	—	—	—	—	—	—	—	—	
		ADJ_MTX_ B[7]	ADJ_MTX_ B[6]	ADJ_MTX_ B[5]	ADJ_MTX_ B[4]	ADJ_MTX_ B[3]	ADJ_MTX_ B[2]	ADJ_MTX_ B[1]	ADJ_MTX_ B[0]	
		—	—	—	—	—	ADJ_MTX_ BG[10]	ADJ_MTX_ BG[9]	ADJ_MTX_ BG[8]	
		ADJ_MTX_ BG[7]	ADJ_MTX_ BG[6]	ADJ_MTX_ BG[5]	ADJ_MTX_ BG[4]	ADJ_MTX_ BG[3]	ADJ_MTX_ BG[2]	ADJ_MTX_ BG[1]	ADJ_MTX_ BG[0]	
	ADJ_MTX_ CBB_ADJ1	—	—	—	—	—	—	ADJ_MTX_ BB[10]	ADJ_MTX_ BB[9]	ADJ_MTX_ BB[8]
		ADJ_MTX_ BB[7]	ADJ_MTX_ BB[6]	ADJ_MTX_ BB[5]	ADJ_MTX_ BB[4]	ADJ_MTX_ BB[3]	ADJ_MTX_ BB[2]	ADJ_MTX_ BB[1]	ADJ_MTX_ BB[0]	
		—	—	—	—	—	ADJ_MTX_ BR[10]	ADJ_MTX_ BR[9]	ADJ_MTX_ BR[8]	
		ADJ_MTX_ BR[7]	ADJ_MTX_ BR[6]	ADJ_MTX_ BR[5]	ADJ_MTX_ BR[4]	ADJ_MTX_ BR[3]	ADJ_MTX_ BR[2]	ADJ_MTX_ BR[1]	ADJ_MTX_ BR[0]	
	ADJ_MTX_ CRR_ADJ0	—	—	—	—	—	—	—	—	
		ADJ_MTX_ R[7]	ADJ_MTX_ R[6]	ADJ_MTX_ R[5]	ADJ_MTX_ R[4]	ADJ_MTX_ R[3]	ADJ_MTX_ R[2]	ADJ_MTX_ R[1]	ADJ_MTX_ R[0]	
		—	—	—	—	—	ADJ_MTX_ RG[10]	ADJ_MTX_ RG[9]	ADJ_MTX_ RG[8]	
		ADJ_MTX_ RG[7]	ADJ_MTX_ RG[6]	ADJ_MTX_ RG[5]	ADJ_MTX_ RG[4]	ADJ_MTX_ RG[3]	ADJ_MTX_ RG[2]	ADJ_MTX_ RG[1]	ADJ_MTX_ RG[0]	
	ADJ_MTX_ CRR_ADJ1	—	—	—	—	—	—	ADJ_MTX_ RB[10]	ADJ_MTX_ RB[9]	ADJ_MTX_ RB[8]
		ADJ_MTX_ RB[7]	ADJ_MTX_ RB[6]	ADJ_MTX_ RB[5]	ADJ_MTX_ RB[4]	ADJ_MTX_ RB[3]	ADJ_MTX_ RB[2]	ADJ_MTX_ RB[1]	ADJ_MTX_ RB[0]	
		—	—	—	—	—	ADJ_MTX_ RR[10]	ADJ_MTX_ RR[9]	ADJ_MTX_ RR[8]	
		ADJ_MTX_ RR[7]	ADJ_MTX_ RR[6]	ADJ_MTX_ RR[5]	ADJ_MTX_ RR[4]	ADJ_MTX_ RR[3]	ADJ_MTX_ RR[2]	ADJ_MTX_ RR[1]	ADJ_MTX_ RR[0]	
	GR2_ UPDATE	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	GR2_P_VEN	—	—	—	GR2_IBUS_ VEN	
	GR2_FLM_ RD	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	GR2_R_ENB	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	GR2_FLM1	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	GR2_LN_ OFF_DIR
		—	—	—	—	—	—	GR2_FLM_ SEL[1]	GR2_FLM_ SEL[0]
		—	—	—	—	—	—	—	GR2_BST_ MD
	GR2_FLM2	GR2_ BASE[31]	GR2_ BASE[30]	GR2_ BASE[29]	GR2_ BASE[28]	GR2_ BASE[27]	GR2_ BASE[26]	GR2_ BASE[25]	GR2_ BASE[24]
		GR2_ BASE[23]	GR2_ BASE[22]	GR2_ BASE[21]	GR2_ BASE[20]	GR2_ BASE[19]	GR2_ BASE[18]	GR2_ BASE[17]	GR2_ BASE[16]
		GR2_ BASE[15]	GR2_ BASE[14]	GR2_ BASE[13]	GR2_ BASE[12]	GR2_ BASE[11]	GR2_ BASE[10]	GR2_ BASE[9]	GR2_ BASE[8]
		GR2_ BASE[7]	GR2_ BASE[6]	GR2_ BASE[5]	GR2_ BASE[4]	GR2_ BASE[3]	GR2_ BASE[2]	GR2_ BASE[1]	GR2_ BASE[0]
	GR2_FLM3	—	GR2_LN_ OFF[14]	GR2_LN_ OFF[13]	GR2_LN_ OFF[12]	GR2_LN_ OFF[11]	GR2_LN_ OFF[10]	GR2_LN_ OFF[9]	GR2_LN_ OFF[8]
		GR2_LN_ OFF[7]	GR2_LN_ OFF[6]	GR2_LN_ OFF[5]	GR2_LN_ OFF[4]	GR2_LN_ OFF[3]	GR2_LN_ OFF[2]	GR2_LN_ OFF[1]	GR2_LN_ OFF[0]
		—	—	—	—	—	—	GR2_FLM_ NUM[9]	GR2_FLM_ NUM[8]
		GR2_FLM_ NUM[7]	GR2_FLM_ NUM[6]	GR2_FLM_ NUM[5]	GR2_FLM_ NUM[4]	GR2_FLM_ NUM[3]	GR2_FLM_ NUM[2]	GR2_FLM_ NUM[1]	GR2_FLM_ NUM[0]
	GR2_FLM4	—	—	—	—	—	—	—	—
		—	GR2_FLM_ OFF[22]	GR2_FLM_ OFF[21]	GR2_FLM_ OFF[20]	GR2_FLM_ OFF[19]	GR2_FLM_ OFF[18]	GR2_FLM_ OFF[17]	GR2_FLM_ OFF[16]
		GR2_FLM_ OFF[15]	GR2_FLM_ OFF[14]	GR2_FLM_ OFF[13]	GR2_FLM_ OFF[12]	GR2_FLM_ OFF[11]	GR2_FLM_ OFF[10]	GR2_FLM_ OFF[9]	GR2_FLM_ OFF[8]
		GR2_FLM_ OFF[7]	GR2_FLM_ OFF[6]	GR2_FLM_ OFF[5]	GR2_FLM_ OFF[4]	GR2_FLM_ OFF[3]	GR2_FLM_ OFF[2]	GR2_FLM_ OFF[1]	GR2_FLM_ OFF[0]
	GR2_FLM5	—	—	—	—	—	—	—	GR2_FLM_ LNUM[9]
		GR2_FLM_ LNUM[7]	GR2_FLM_ LNUM[6]	GR2_FLM_ LNUM[5]	GR2_FLM_ LNUM[4]	GR2_FLM_ LNUM[3]	GR2_FLM_ LNUM[2]	GR2_FLM_ LNUM[1]	GR2_FLM_ LNUM[0]
		—	—	—	—	—	—	GR2_FLM_ LOOP[9]	GR2_FLM_ LOOP[8]
		GR2_FLM_ LOOP[7]	GR2_FLM_ LOOP[6]	GR2_FLM_ LOOP[5]	GR2_FLM_ LOOP[4]	GR2_FLM_ LOOP[3]	GR2_FLM_ LOOP[2]	GR2_FLM_ LOOP[1]	GR2_FLM_ LOOP[0]
	GR2_FLM6	GR2_ FORMAT[3]	GR2_ FORMAT[2]	GR2_ FORMAT[1]	GR2_ FORMAT[0]	—	—	GR2_HW[9]	GR2_HW[8]
		GR2_HW[7]	GR2_HW[6]	GR2_HW[5]	GR2_HW[4]	GR2_HW[3]	GR2_HW[2]	GR2_HW[1]	GR2_HW[0]
		—	—	—	GR2_ ENDIAN_ON	—	—	—	—
		—	—	GR2_STA_ POS[5]	GR2_STA_ POS[4]	GR2_STA_ POS[3]	GR2_STA_ POS[2]	GR2_STA_ POS[1]	GR2_STA_ POS[0]
	GR2_AB1	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	GR2_ARC_ ON	—	—	—	GR2_ARC_ DISP_ON
		—	—	—	GR2_GRC_ DISP_ON	—	—	GR2_DISP_ SEL[1]	GR2_DISP_ SEL[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
视频显示 控制器 4	GR2_AB2	—	—	—	—	—	GR2_GRC_ VS[10]	GR2_GRC_ VS[9]	GR2_GRC_ VS[8]	
		GR2_GRC_ VS[7]	GR2_GRC_ VS[6]	GR2_GRC_ VS[5]	GR2_GRC_ VS[4]	GR2_GRC_ VS[3]	GR2_GRC_ VS[2]	GR2_GRC_ VS[1]	GR2_GRC_ VS[0]	
		—	—	—	—	—	GR2_GRC_ VW[10]	GR2_GRC_ VW[9]	GR2_GRC_ VW[8]	
		GR2_GRC_ VW[7]	GR2_GRC_ VW[6]	GR2_GRC_ VW[5]	GR2_GRC_ VW[4]	GR2_GRC_ VW[3]	GR2_GRC_ VW[2]	GR2_GRC_ VW[1]	GR2_GRC_ VW[0]	
	GR2_AB3	—	—	—	—	—	—	GR2_GRC_ HS[10]	GR2_GRC_ HS[9]	GR2_GRC_ HS[8]
		GR2_GRC_ HS[7]	GR2_GRC_ HS[6]	GR2_GRC_ HS[5]	GR2_GRC_ HS[4]	GR2_GRC_ HS[3]	GR2_GRC_ HS[2]	GR2_GRC_ HS[1]	GR2_GRC_ HS[0]	
		—	—	—	—	—	—	GR2_GRC_ HW[10]	GR2_GRC_ HW[9]	GR2_GRC_ HW[8]
		GR2_GRC_ HW[7]	GR2_GRC_ HW[6]	GR2_GRC_ HW[5]	GR2_GRC_ HW[4]	GR2_GRC_ HW[3]	GR2_GRC_ HW[2]	GR2_GRC_ HW[1]	GR2_GRC_ HW[0]	
	GR2_AB4	—	—	—	—	—	—	GR2_ARC_ VS[10]	GR2_ARC_ VS[9]	GR2_ARC_ VS[8]
		GR2_ARC_ VS[7]	GR2_ARC_ VS[6]	GR2_ARC_ VS[5]	GR2_ARC_ VS[4]	GR2_ARC_ VS[3]	GR2_ARC_ VS[2]	GR2_ARC_ VS[1]	GR2_ARC_ VS[0]	
		—	—	—	—	—	—	GR2_ARC_ VW[10]	GR2_ARC_ VW[9]	GR2_ARC_ VW[8]
		GR2_ARC_ VW[7]	GR2_ARC_ VW[6]	GR2_ARC_ VW[5]	GR2_ARC_ VW[4]	GR2_ARC_ VW[3]	GR2_ARC_ VW[2]	GR2_ARC_ VW[1]	GR2_ARC_ VW[0]	
	GR2_AB5	—	—	—	—	—	—	GR2_ARC_ HS[10]	GR2_ARC_ HS[9]	GR2_ARC_ HS[8]
		GR2_ARC_ HS[7]	GR2_ARC_ HS[6]	GR2_ARC_ HS[5]	GR2_ARC_ HS[4]	GR2_ARC_ HS[3]	GR2_ARC_ HS[2]	GR2_ARC_ HS[1]	GR2_ARC_ HS[0]	
		—	—	—	—	—	—	GR2_ARC_ HW[10]	GR2_ARC_ HW[9]	GR2_ARC_ HW[8]
		GR2_ARC_ HW[7]	GR2_ARC_ HW[6]	GR2_ARC_ HW[5]	GR2_ARC_ HW[4]	GR2_ARC_ HW[3]	GR2_ARC_ HW[2]	GR2_ARC_ HW[1]	GR2_ARC_ HW[0]	
	GR2_AB6	—	—	—	—	—	—	—	—	GR2_ARC_ MODE
		GR2_ARC_ COEF[7]	GR2_ARC_ COEF[6]	GR2_ARC_ COEF[5]	GR2_ARC_ COEF[4]	GR2_ARC_ COEF[3]	GR2_ARC_ COEF[2]	GR2_ARC_ COEF[1]	GR2_ARC_ COEF[0]	
		—	—	—	—	—	—	—	—	
		GR2_ARC_ RATE[7]	GR2_ARC_ RATE[6]	GR2_ARC_ RATE[5]	GR2_ARC_ RATE[4]	GR2_ARC_ RATE[3]	GR2_ARC_ RATE[2]	GR2_ARC_ RATE[1]	GR2_ARC_ RATE[0]	
	GR2_AB7	—	—	—	—	—	—	—	—	—
		GR2_ARC_ DEF[7]	GR2_ARC_ DEF[6]	GR2_ARC_ DEF[5]	GR2_ARC_ DEF[4]	GR2_ARC_ DEF[3]	GR2_ARC_ DEF[2]	GR2_ARC_ DEF[1]	GR2_ARC_ DEF[0]	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	GR2_CK_ON	
	GR2_AB8	GR2_CK_ KCLUT[7]	GR2_CK_ KCLUT[6]	GR2_CK_ KCLUT[5]	GR2_CK_ KCLUT[4]	GR2_CK_ KCLUT[3]	GR2_CK_ KCLUT[2]	GR2_CK_ KCLUT[1]	GR2_CK_ KCLUT[0]	
		GR2_CK_ KG[7]	GR2_CK_ KG[6]	GR2_CK_ KG[5]	GR2_CK_ KG[4]	GR2_CK_ KG[3]	GR2_CK_ KG[2]	GR2_CK_ KG[1]	GR2_CK_ KG[0]	
		GR2_CK_ KB[7]	GR2_CK_ KB[6]	GR2_CK_ KB[5]	GR2_CK_ KB[4]	GR2_CK_ KB[3]	GR2_CK_ KB[2]	GR2_CK_ KB[1]	GR2_CK_ KB[0]	
		GR2_CK_ KR[7]	GR2_CK_ KR[6]	GR2_CK_ KR[5]	GR2_CK_ KR[4]	GR2_CK_ KR[3]	GR2_CK_ KR[2]	GR2_CK_ KR[1]	GR2_CK_ KR[0]	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	GR2_AB9	GR2_CK_A[7]	GR2_CK_A[6]	GR2_CK_A[5]	GR2_CK_A[4]	GR2_CK_A[3]	GR2_CK_A[2]	GR2_CK_A[1]	GR2_CK_A[0]
		GR2_CK_G[7]	GR2_CK_G[6]	GR2_CK_G[5]	GR2_CK_G[4]	GR2_CK_G[3]	GR2_CK_G[2]	GR2_CK_G[1]	GR2_CK_G[0]
		GR2_CK_B[7]	GR2_CK_B[6]	GR2_CK_B[5]	GR2_CK_B[4]	GR2_CK_B[3]	GR2_CK_B[2]	GR2_CK_B[1]	GR2_CK_B[0]
		GR2_CK_R[7]	GR2_CK_R[6]	GR2_CK_R[5]	GR2_CK_R[4]	GR2_CK_R[3]	GR2_CK_R[2]	GR2_CK_R[1]	GR2_CK_R[0]
	GR2_AB10	GR2_A0[7]	GR2_A0[6]	GR2_A0[5]	GR2_A0[4]	GR2_A0[3]	GR2_A0[2]	GR2_A0[1]	GR2_A0[0]
		GR2_G0[7]	GR2_G0[6]	GR2_G0[5]	GR2_G0[4]	GR2_G0[3]	GR2_G0[2]	GR2_G0[1]	GR2_G0[0]
		GR2_B0[7]	GR2_B0[6]	GR2_B0[5]	GR2_B0[4]	GR2_B0[3]	GR2_B0[2]	GR2_B0[1]	GR2_B0[0]
		GR2_R0[7]	GR2_R0[6]	GR2_R0[5]	GR2_R0[4]	GR2_R0[3]	GR2_R0[2]	GR2_R0[1]	GR2_R0[0]
	GR2_AB11	GR2_A1[7]	GR2_A1[6]	GR2_A1[5]	GR2_A1[4]	GR2_A1[3]	GR2_A1[2]	GR2_A1[1]	GR2_A1[0]
		GR2_G1[7]	GR2_G1[6]	GR2_G1[5]	GR2_G1[4]	GR2_G1[3]	GR2_G1[2]	GR2_G1[1]	GR2_G1[0]
		GR2_B1[7]	GR2_B1[6]	GR2_B1[5]	GR2_B1[4]	GR2_B1[3]	GR2_B1[2]	GR2_B1[1]	GR2_B1[0]
	GR2_BASE	GR2_R1[7]	GR2_R1[6]	GR2_R1[5]	GR2_R1[4]	GR2_R1[3]	GR2_R1[2]	GR2_R1[1]	GR2_R1[0]
		—	—	—	—	—	—	—	—
		GR2_BASE_G[7]	GR2_BASE_G[6]	GR2_BASE_G[5]	GR2_BASE_G[4]	GR2_BASE_G[3]	GR2_BASE_G[2]	GR2_BASE_G[1]	GR2_BASE_G[0]
		GR2_BASE_B[7]	GR2_BASE_B[6]	GR2_BASE_B[5]	GR2_BASE_B[4]	GR2_BASE_B[3]	GR2_BASE_B[2]	GR2_BASE_B[1]	GR2_BASE_B[0]
	GR2_CLUT	GR2_BASE_R[7]	GR2_BASE_R[6]	GR2_BASE_R[5]	GR2_BASE_R[4]	GR2_BASE_R[3]	GR2_BASE_R[2]	GR2_BASE_R[1]	GR2_BASE_R[0]
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	GR2_CLT_SEL
		—	—	—	—	—	—	—	—
	GR2_MON	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	GR2_ARC_ST
	GR3_UPDATE	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	GR3_P_VEN	—	—	—	GR3_IBUS_VEN
	GR3_FLM_RD	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	GR3_R_ENB
	GR3_FLM1	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	GR3_LN_OFF_DIR
		—	—	—	—	—	—	GR3_FLM_SEL[1]	GR3_FLM_SEL[0]
		—	—	—	—	—	—	—	GR3_BST_MD

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	GR3_FLM2	GR3_BASE[31]	GR3_BASE[30]	GR3_BASE[29]	GR3_BASE[28]	GR3_BASE[27]	GR3_BASE[26]	GR3_BASE[25]	GR3_BASE[24]
		GR3_BASE[23]	GR3_BASE[22]	GR3_BASE[21]	GR3_BASE[20]	GR3_BASE[19]	GR3_BASE[18]	GR3_BASE[17]	GR3_BASE[16]
		GR3_BASE[15]	GR3_BASE[14]	GR3_BASE[13]	GR3_BASE[12]	GR3_BASE[11]	GR3_BASE[10]	GR3_BASE[9]	GR3_BASE[8]
		GR3_BASE[7]	GR3_BASE[6]	GR3_BASE[5]	GR3_BASE[4]	GR3_BASE[3]	GR3_BASE[2]	GR3_BASE[1]	GR3_BASE[0]
	GR3_FLM3	—	GR3_LN_OFF[14]	GR3_LN_OFF[13]	GR3_LN_OFF[12]	GR3_LN_OFF[11]	GR3_LN_OFF[10]	GR3_LN_OFF[9]	GR3_LN_OFF[8]
		GR3_LN_OFF[7]	GR3_LN_OFF[6]	GR3_LN_OFF[5]	GR3_LN_OFF[4]	GR3_LN_OFF[3]	GR3_LN_OFF[2]	GR3_LN_OFF[1]	GR3_LN_OFF[0]
		—	—	—	—	—	—	GR3_FLM_NUM[9]	GR3_FLM_NUM[8]
		GR3_FLM_NUM[7]	GR3_FLM_NUM[6]	GR3_FLM_NUM[5]	GR3_FLM_NUM[4]	GR3_FLM_NUM[3]	GR3_FLM_NUM[2]	GR3_FLM_NUM[1]	GR3_FLM_NUM[0]
	GR3_FLM4	—	—	—	—	—	—	—	—
		—	GR3_FLM_OFF[22]	GR3_FLM_OFF[21]	GR3_FLM_OFF[20]	GR3_FLM_OFF[19]	GR3_FLM_OFF[18]	GR3_FLM_OFF[17]	GR3_FLM_OFF[16]
		GR3_FLM_OFF[15]	GR3_FLM_OFF[14]	GR3_FLM_OFF[13]	GR3_FLM_OFF[12]	GR3_FLM_OFF[11]	GR3_FLM_OFF[10]	GR3_FLM_OFF[9]	GR3_FLM_OFF[8]
		GR3_FLM_OFF[7]	GR3_FLM_OFF[6]	GR3_FLM_OFF[5]	GR3_FLM_OFF[4]	GR3_FLM_OFF[3]	GR3_FLM_OFF[2]	GR3_FLM_OFF[1]	GR3_FLM_OFF[0]
	GR3_FLM5	—	—	—	—	—	—	GR3_FLM_LNUM[9]	GR3_FLM_LNUM[8]
		GR3_FLM_LNUM[7]	GR3_FLM_LNUM[6]	GR3_FLM_LNUM[5]	GR3_FLM_LNUM[4]	GR3_FLM_LNUM[3]	GR3_FLM_LNUM[2]	GR3_FLM_LNUM[1]	GR3_FLM_LNUM[0]
		—	—	—	—	—	—	GR3_FLM_LOOP[9]	GR3_FLM_LOOP[8]
		GR3_FLM_LOOP[7]	GR3_FLM_LOOP[6]	GR3_FLM_LOOP[5]	GR3_FLM_LOOP[4]	GR3_FLM_LOOP[3]	GR3_FLM_LOOP[2]	GR3_FLM_LOOP[1]	GR3_FLM_LOOP[0]
	GR3_FLM6	GR3_FORMAT[3]	GR3_FORMAT[2]	GR3_FORMAT[1]	GR3_FORMAT[0]	—	—	GR3_HW[9]	GR3_HW[8]
		GR3_HW[7]	GR3_HW[6]	GR3_HW[5]	GR3_HW[4]	GR3_HW[3]	GR3_HW[2]	GR3_HW[1]	GR3_HW[0]
		—	—	—	GR3_ENDIAN_ON	—	—	—	—
		—	—	GR3_STA_POS[5]	GR3_STA_POS[4]	GR3_STA_POS[3]	GR3_STA_POS[2]	GR3_STA_POS[1]	GR3_STA_POS[0]
	GR3_AB1	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	GR3_ARC_ON	—	—	—	GR3_ARC_DISP_ON
		—	—	—	GR3_GRC_DISP_ON	—	—	GR3_DISP_SEL[1]	GR3_DISP_SEL[0]
	GR3_AB2	—	—	—	—	—	GR3_GRC_VS[10]	GR3_GRC_VS[9]	GR3_GRC_VS[8]
		GR3_GRC_VS[7]	GR3_GRC_VS[6]	GR3_GRC_VS[5]	GR3_GRC_VS[4]	GR3_GRC_VS[3]	GR3_GRC_VS[2]	GR3_GRC_VS[1]	GR3_GRC_VS[0]
		—	—	—	—	—	GR3_GRC_VW[10]	GR3_GRC_VW[9]	GR3_GRC_VW[8]
		GR3_GRC_VW[7]	GR3_GRC_VW[6]	GR3_GRC_VW[5]	GR3_GRC_VW[4]	GR3_GRC_VW[3]	GR3_GRC_VW[2]	GR3_GRC_VW[1]	GR3_GRC_VW[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
视频显示 控制器 4	GR3_AB3	—	—	—	—	—	GR3_GRC_ HS[10]	GR3_GRC_ HS[9]	GR3_GRC_ HS[8]	
		GR3_GRC_ HS[7]	GR3_GRC_ HS[6]	GR3_GRC_ HS[5]	GR3_GRC_ HS[4]	GR3_GRC_ HS[3]	GR3_GRC_ HS[2]	GR3_GRC_ HS[1]	GR3_GRC_ HS[0]	
		—	—	—	—	—	GR3_GRC_ HW[10]	GR3_GRC_ HW[9]	GR3_GRC_ HW[8]	
		GR3_GRC_ HW[7]	GR3_GRC_ HW[6]	GR3_GRC_ HW[5]	GR3_GRC_ HW[4]	GR3_GRC_ HW[3]	GR3_GRC_ HW[2]	GR3_GRC_ HW[1]	GR3_GRC_ HW[0]	
	GR3_AB4	—	—	—	—	—	—	GR3_ARC_ VS[10]	GR3_ARC_ VS[9]	GR3_ARC_ VS[8]
		GR3_ARC_ VS[7]	GR3_ARC_ VS[6]	GR3_ARC_ VS[5]	GR3_ARC_ VS[4]	GR3_ARC_ VS[3]	GR3_ARC_ VS[2]	GR3_ARC_ VS[1]	GR3_ARC_ VS[0]	
		—	—	—	—	—	GR3_ARC_ VW[10]	GR3_ARC_ VW[9]	GR3_ARC_ VW[8]	
		GR3_ARC_ VW[7]	GR3_ARC_ VW[6]	GR3_ARC_ VW[5]	GR3_ARC_ VW[4]	GR3_ARC_ VW[3]	GR3_ARC_ VW[2]	GR3_ARC_ VW[1]	GR3_ARC_ VW[0]	
	GR3_AB5	—	—	—	—	—	—	GR3_ARC_ HS[10]	GR3_ARC_ HS[9]	GR3_ARC_ HS[8]
		GR3_ARC_ HS[7]	GR3_ARC_ HS[6]	GR3_ARC_ HS[5]	GR3_ARC_ HS[4]	GR3_ARC_ HS[3]	GR3_ARC_ HS[2]	GR3_ARC_ HS[1]	GR3_ARC_ HS[0]	
		—	—	—	—	—	GR3_ARC_ HW[10]	GR3_ARC_ HW[9]	GR3_ARC_ HW[8]	
		GR3_ARC_ HW[7]	GR3_ARC_ HW[6]	GR3_ARC_ HW[5]	GR3_ARC_ HW[4]	GR3_ARC_ HW[3]	GR3_ARC_ HW[2]	GR3_ARC_ HW[1]	GR3_ARC_ HW[0]	
	GR3_AB6	—	—	—	—	—	—	—	—	GR3_ARC_ MODE
		GR3_ARC_ COEF[7]	GR3_ARC_ COEF[6]	GR3_ARC_ COEF[5]	GR3_ARC_ COEF[4]	GR3_ARC_ COEF[3]	GR3_ARC_ COEF[2]	GR3_ARC_ COEF[1]	GR3_ARC_ COEF[0]	
		—	—	—	—	—	—	—	—	
		GR3_ARC_ RATE[7]	GR3_ARC_ RATE[6]	GR3_ARC_ RATE[5]	GR3_ARC_ RATE[4]	GR3_ARC_ RATE[3]	GR3_ARC_ RATE[2]	GR3_ARC_ RATE[1]	GR3_ARC_ RATE[0]	
	GR3_AB7	—	—	—	—	—	—	—	—	—
		GR3_ARC_ DEF[7]	GR3_ARC_ DEF[6]	GR3_ARC_ DEF[5]	GR3_ARC_ DEF[4]	GR3_ARC_ DEF[3]	GR3_ARC_ DEF[2]	GR3_ARC_ DEF[1]	GR3_ARC_ DEF[0]	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	GR3_CK_ON	
	GR3_AB8	GR3_CK_ KCLUT[7]	GR3_CK_ KCLUT[6]	GR3_CK_ KCLUT[5]	GR3_CK_ KCLUT[4]	GR3_CK_ KCLUT[3]	GR3_CK_ KCLUT[2]	GR3_CK_ KCLUT[1]	GR3_CK_ KCLUT[0]	
		GR3_CK_ KG[7]	GR3_CK_ KG[6]	GR3_CK_ KG[5]	GR3_CK_ KG[4]	GR3_CK_ KG[3]	GR3_CK_ KG[2]	GR3_CK_ KG[1]	GR3_CK_ KG[0]	
		GR3_CK_ KB[7]	GR3_CK_ KB[6]	GR3_CK_ KB[5]	GR3_CK_ KB[4]	GR3_CK_ KB[3]	GR3_CK_ KB[2]	GR3_CK_ KB[1]	GR3_CK_ KB[0]	
		GR3_CK_ KR[7]	GR3_CK_ KR[6]	GR3_CK_ KR[5]	GR3_CK_ KR[4]	GR3_CK_ KR[3]	GR3_CK_ KR[2]	GR3_CK_ KR[1]	GR3_CK_ KR[0]	
	GR3_AB9	GR3_CK_ A[7]	GR3_CK_ A[6]	GR3_CK_ A[5]	GR3_CK_ A[4]	GR3_CK_ A[3]	GR3_CK_ A[2]	GR3_CK_ A[1]	GR3_CK_ A[0]	
		GR3_CK_ G[7]	GR3_CK_ G[6]	GR3_CK_ G[5]	GR3_CK_ G[4]	GR3_CK_ G[3]	GR3_CK_ G[2]	GR3_CK_ G[1]	GR3_CK_ G[0]	
		GR3_CK_ B[7]	GR3_CK_ B[6]	GR3_CK_ B[5]	GR3_CK_ B[4]	GR3_CK_ B[3]	GR3_CK_ B[2]	GR3_CK_ B[1]	GR3_CK_ B[0]	
		GR3_CK_ R[7]	GR3_CK_ R[6]	GR3_CK_ R[5]	GR3_CK_ R[4]	GR3_CK_ R[3]	GR3_CK_ R[2]	GR3_CK_ R[1]	GR3_CK_ R[0]	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	GR3_AB10	GR3_A0[7]	GR3_A0[6]	GR3_A0[5]	GR3_A0[4]	GR3_A0[3]	GR3_A0[2]	GR3_A0[1]	GR3_A0[0]
		GR3_G0[7]	GR3_G0[6]	GR3_G0[5]	GR3_G0[4]	GR3_G0[3]	GR3_G0[2]	GR3_G0[1]	GR3_G0[0]
		GR3_B0[7]	GR3_B0[6]	GR3_B0[5]	GR3_B0[4]	GR3_B0[3]	GR3_B0[2]	GR3_B0[1]	GR3_B0[0]
		GR3_R0[7]	GR3_R0[6]	GR3_R0[5]	GR3_R0[4]	GR3_R0[3]	GR3_R0[2]	GR3_R0[1]	GR3_R0[0]
	GR3_AB11	GR3_A1[7]	GR3_A1[6]	GR3_A1[5]	GR3_A1[4]	GR3_A1[3]	GR3_A1[2]	GR3_A1[1]	GR3_A1[0]
		GR3_G1[7]	GR3_G1[6]	GR3_G1[5]	GR3_G1[4]	GR3_G1[3]	GR3_G1[2]	GR3_G1[1]	GR3_G1[0]
		GR3_B1[7]	GR3_B1[6]	GR3_B1[5]	GR3_B1[4]	GR3_B1[3]	GR3_B1[2]	GR3_B1[1]	GR3_B1[0]
		GR3_R1[7]	GR3_R1[6]	GR3_R1[5]	GR3_R1[4]	GR3_R1[3]	GR3_R1[2]	GR3_R1[1]	GR3_R1[0]
	GR3_BASE	—	—	—	—	—	—	—	—
		GR3_BASE_G[7]	GR3_BASE_G[6]	GR3_BASE_G[5]	GR3_BASE_G[4]	GR3_BASE_G[3]	GR3_BASE_G[2]	GR3_BASE_G[1]	GR3_BASE_G[0]
		GR3_BASE_B[7]	GR3_BASE_B[6]	GR3_BASE_B[5]	GR3_BASE_B[4]	GR3_BASE_B[3]	GR3_BASE_B[2]	GR3_BASE_B[1]	GR3_BASE_B[0]
		GR3_BASE_R[7]	GR3_BASE_R[6]	GR3_BASE_R[5]	GR3_BASE_R[4]	GR3_BASE_R[3]	GR3_BASE_R[2]	GR3_BASE_R[1]	GR3_BASE_R[0]
	GR3_CLUT_INT	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	GR3_CLT_SEL
		—	—	—	—	—	GR3_LINE[10]	GR3_LINE[9]	GR3_LINE[8]
		GR3_LINE[7]	GR3_LINE[6]	GR3_LINE[5]	GR3_LINE[4]	GR3_LINE[3]	GR3_LINE[2]	GR3_LINE[1]	GR3_LINE[0]
	GR3_MON	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	GR3_ARC_ST
		—	—	—	—	—	—	—	—
	GAM_G_UPDATE	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	GAM_G_VEN
	GAM_SW	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	GAM_ON
	GAM_G_LUT1	—	—	—	—	—	GAM_G_GAIN_00[10]	GAM_G_GAIN_00[9]	GAM_G_GAIN_00[8]
		GAM_G_GAIN_00[7]	GAM_G_GAIN_00[6]	GAM_G_GAIN_00[5]	GAM_G_GAIN_00[4]	GAM_G_GAIN_00[3]	GAM_G_GAIN_00[2]	GAM_G_GAIN_00[1]	GAM_G_GAIN_00[0]
		—	—	—	—	—	GAM_G_GAIN_01[10]	GAM_G_GAIN_01[9]	GAM_G_GAIN_01[8]
		GAM_G_GAIN_01[7]	GAM_G_GAIN_01[6]	GAM_G_GAIN_01[5]	GAM_G_GAIN_01[4]	GAM_G_GAIN_01[3]	GAM_G_GAIN_01[2]	GAM_G_GAIN_01[1]	GAM_G_GAIN_01[0]
	GAM_G_LUT2	—	—	—	—	—	GAM_G_GAIN_02[10]	GAM_G_GAIN_02[9]	GAM_G_GAIN_02[8]
		GAM_G_GAIN_02[7]	GAM_G_GAIN_02[6]	GAM_G_GAIN_02[5]	GAM_G_GAIN_02[4]	GAM_G_GAIN_02[3]	GAM_G_GAIN_02[2]	GAM_G_GAIN_02[1]	GAM_G_GAIN_02[0]
		—	—	—	—	—	GAM_G_GAIN_03[10]	GAM_G_GAIN_03[9]	GAM_G_GAIN_03[8]
		GAM_G_GAIN_03[7]	GAM_G_GAIN_03[6]	GAM_G_GAIN_03[5]	GAM_G_GAIN_03[4]	GAM_G_GAIN_03[3]	GAM_G_GAIN_03[2]	GAM_G_GAIN_03[1]	GAM_G_GAIN_03[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	GAM_G_ LUT3	—	—	—	—	—	GAM_G_ GAIN_04[10]	GAM_G_ GAIN_04[9]	GAM_G_ GAIN_04[8]
		GAM_G_ GAIN_04[7]	GAM_G_ GAIN_04[6]	GAM_G_ GAIN_04[5]	GAM_G_ GAIN_04[4]	GAM_G_ GAIN_04[3]	GAM_G_ GAIN_04[2]	GAM_G_ GAIN_04[1]	GAM_G_ GAIN_04[0]
		—	—	—	—	—	GAM_G_ GAIN_05[10]	GAM_G_ GAIN_05[9]	GAM_G_ GAIN_05[8]
		GAM_G_ GAIN_05[7]	GAM_G_ GAIN_05[6]	GAM_G_ GAIN_05[5]	GAM_G_ GAIN_05[4]	GAM_G_ GAIN_05[3]	GAM_G_ GAIN_05[2]	GAM_G_ GAIN_05[1]	GAM_G_ GAIN_05[0]
	GAM_G_ LUT4	—	—	—	—	—	GAM_G_ GAIN_06[10]	GAM_G_ GAIN_06[9]	GAM_G_ GAIN_06[8]
		GAM_G_ GAIN_06[7]	GAM_G_ GAIN_06[6]	GAM_G_ GAIN_06[5]	GAM_G_ GAIN_06[4]	GAM_G_ GAIN_06[3]	GAM_G_ GAIN_06[2]	GAM_G_ GAIN_06[1]	GAM_G_ GAIN_06[0]
		—	—	—	—	—	GAM_G_ GAIN_07[10]	GAM_G_ GAIN_07[9]	GAM_G_ GAIN_07[8]
		GAM_G_ GAIN_07[7]	GAM_G_ GAIN_07[6]	GAM_G_ GAIN_07[5]	GAM_G_ GAIN_07[4]	GAM_G_ GAIN_07[3]	GAM_G_ GAIN_07[2]	GAM_G_ GAIN_07[1]	GAM_G_ GAIN_07[0]
	GAM_G_ LUT5	—	—	—	—	—	GAM_G_ GAIN_08[10]	GAM_G_ GAIN_08[9]	GAM_G_ GAIN_08[8]
		GAM_G_ GAIN_08[7]	GAM_G_ GAIN_08[6]	GAM_G_ GAIN_08[5]	GAM_G_ GAIN_08[4]	GAM_G_ GAIN_08[3]	GAM_G_ GAIN_08[2]	GAM_G_ GAIN_08[1]	GAM_G_ GAIN_08[0]
		—	—	—	—	—	GAM_G_ GAIN_09[10]	GAM_G_ GAIN_09[9]	GAM_G_ GAIN_09[8]
		GAM_G_ GAIN_09[7]	GAM_G_ GAIN_09[6]	GAM_G_ GAIN_09[5]	GAM_G_ GAIN_09[4]	GAM_G_ GAIN_09[3]	GAM_G_ GAIN_09[2]	GAM_G_ GAIN_09[1]	GAM_G_ GAIN_09[0]
	GAM_G_ LUT6	—	—	—	—	—	GAM_G_ GAIN_10[10]	GAM_G_ GAIN_10[9]	GAM_G_ GAIN_10[8]
		GAM_G_ GAIN_10[7]	GAM_G_ GAIN_10[6]	GAM_G_ GAIN_10[5]	GAM_G_ GAIN_10[4]	GAM_G_ GAIN_10[3]	GAM_G_ GAIN_10[2]	GAM_G_ GAIN_10[1]	GAM_G_ GAIN_10[0]
		—	—	—	—	—	GAM_G_ GAIN_11[10]	GAM_G_ GAIN_11[9]	GAM_G_ GAIN_11[8]
		GAM_G_ GAIN_11[7]	GAM_G_ GAIN_11[6]	GAM_G_ GAIN_11[5]	GAM_G_ GAIN_11[4]	GAM_G_ GAIN_11[3]	GAM_G_ GAIN_11[2]	GAM_G_ GAIN_11[1]	GAM_G_ GAIN_11[0]
	GAM_G_ LUT7	—	—	—	—	—	GAM_G_ GAIN_12[10]	GAM_G_ GAIN_12[9]	GAM_G_ GAIN_12[8]
		GAM_G_ GAIN_12[7]	GAM_G_ GAIN_12[6]	GAM_G_ GAIN_12[5]	GAM_G_ GAIN_12[4]	GAM_G_ GAIN_12[3]	GAM_G_ GAIN_12[2]	GAM_G_ GAIN_12[1]	GAM_G_ GAIN_12[0]
		—	—	—	—	—	GAM_G_ GAIN_13[10]	GAM_G_ GAIN_13[9]	GAM_G_ GAIN_13[8]
		GAM_G_ GAIN_13[7]	GAM_G_ GAIN_13[6]	GAM_G_ GAIN_13[5]	GAM_G_ GAIN_13[4]	GAM_G_ GAIN_13[3]	GAM_G_ GAIN_13[2]	GAM_G_ GAIN_13[1]	GAM_G_ GAIN_13[0]
	GAM_G_ LUT8	—	—	—	—	—	GAM_G_ GAIN_14[10]	GAM_G_ GAIN_14[9]	GAM_G_ GAIN_14[8]
		GAM_G_ GAIN_14[7]	GAM_G_ GAIN_14[6]	GAM_G_ GAIN_14[5]	GAM_G_ GAIN_14[4]	GAM_G_ GAIN_14[3]	GAM_G_ GAIN_14[2]	GAM_G_ GAIN_14[1]	GAM_G_ GAIN_14[0]
		—	—	—	—	—	GAM_G_ GAIN_15[10]	GAM_G_ GAIN_15[9]	GAM_G_ GAIN_15[8]
		GAM_G_ GAIN_15[7]	GAM_G_ GAIN_15[6]	GAM_G_ GAIN_15[5]	GAM_G_ GAIN_15[4]	GAM_G_ GAIN_15[3]	GAM_G_ GAIN_15[2]	GAM_G_ GAIN_15[1]	GAM_G_ GAIN_15[0]
	GAM_G_ LUT9	—	—	—	—	—	GAM_G_ GAIN_16[10]	GAM_G_ GAIN_16[9]	GAM_G_ GAIN_16[8]
		GAM_G_ GAIN_16[7]	GAM_G_ GAIN_16[6]	GAM_G_ GAIN_16[5]	GAM_G_ GAIN_16[4]	GAM_G_ GAIN_16[3]	GAM_G_ GAIN_16[2]	GAM_G_ GAIN_16[1]	GAM_G_ GAIN_16[0]
		—	—	—	—	—	GAM_G_ GAIN_17[10]	GAM_G_ GAIN_17[9]	GAM_G_ GAIN_17[8]
		GAM_G_ GAIN_17[7]	GAM_G_ GAIN_17[6]	GAM_G_ GAIN_17[5]	GAM_G_ GAIN_17[4]	GAM_G_ GAIN_17[3]	GAM_G_ GAIN_17[2]	GAM_G_ GAIN_17[1]	GAM_G_ GAIN_17[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
视频显示 控制器 4	GAM_G_ LUT10	—	—	—	—	—	GAM_G_ GAIN_18[10]	GAM_G_ GAIN_18[9]	GAM_G_ GAIN_18[8]	
		GAM_G_ GAIN_18[7]	GAM_G_ GAIN_18[6]	GAM_G_ GAIN_18[5]	GAM_G_ GAIN_18[4]	GAM_G_ GAIN_18[3]	GAM_G_ GAIN_18[2]	GAM_G_ GAIN_18[1]	GAM_G_ GAIN_18[0]	
		—	—	—	—	—	GAM_G_ GAIN_19[10]	GAM_G_ GAIN_19[9]	GAM_G_ GAIN_19[8]	
		GAM_G_ GAIN_19[7]	GAM_G_ GAIN_19[6]	GAM_G_ GAIN_19[5]	GAM_G_ GAIN_19[4]	GAM_G_ GAIN_19[3]	GAM_G_ GAIN_19[2]	GAM_G_ GAIN_19[1]	GAM_G_ GAIN_19[0]	
	GAM_G_ LUT11	—	—	—	—	—	—	GAM_G_ GAIN_20[10]	GAM_G_ GAIN_20[9]	GAM_G_ GAIN_20[8]
		GAM_G_ GAIN_20[7]	GAM_G_ GAIN_20[6]	GAM_G_ GAIN_20[5]	GAM_G_ GAIN_20[4]	GAM_G_ GAIN_20[3]	GAM_G_ GAIN_20[2]	GAM_G_ GAIN_20[1]	GAM_G_ GAIN_20[0]	
		—	—	—	—	—	—	GAM_G_ GAIN_21[10]	GAM_G_ GAIN_21[9]	GAM_G_ GAIN_21[8]
		GAM_G_ GAIN_21[7]	GAM_G_ GAIN_21[6]	GAM_G_ GAIN_21[5]	GAM_G_ GAIN_21[4]	GAM_G_ GAIN_21[3]	GAM_G_ GAIN_21[2]	GAM_G_ GAIN_21[1]	GAM_G_ GAIN_21[0]	
	GAM_G_ LUT12	—	—	—	—	—	—	GAM_G_ GAIN_22[10]	GAM_G_ GAIN_22[9]	GAM_G_ GAIN_22[8]
		GAM_G_ GAIN_22[7]	GAM_G_ GAIN_22[6]	GAM_G_ GAIN_22[5]	GAM_G_ GAIN_22[4]	GAM_G_ GAIN_22[3]	GAM_G_ GAIN_22[2]	GAM_G_ GAIN_22[1]	GAM_G_ GAIN_22[0]	
		—	—	—	—	—	—	GAM_G_ GAIN_23[10]	GAM_G_ GAIN_23[9]	GAM_G_ GAIN_23[8]
		GAM_G_ GAIN_23[7]	GAM_G_ GAIN_23[6]	GAM_G_ GAIN_23[5]	GAM_G_ GAIN_23[4]	GAM_G_ GAIN_23[3]	GAM_G_ GAIN_23[2]	GAM_G_ GAIN_23[1]	GAM_G_ GAIN_23[0]	
	GAM_G_ LUT13	—	—	—	—	—	—	GAM_G_ GAIN_24[10]	GAM_G_ GAIN_24[9]	GAM_G_ GAIN_24[8]
		GAM_G_ GAIN_24[7]	GAM_G_ GAIN_24[6]	GAM_G_ GAIN_24[5]	GAM_G_ GAIN_24[4]	GAM_G_ GAIN_24[3]	GAM_G_ GAIN_24[2]	GAM_G_ GAIN_24[1]	GAM_G_ GAIN_24[0]	
		—	—	—	—	—	—	GAM_G_ GAIN_25[10]	GAM_G_ GAIN_25[9]	GAM_G_ GAIN_25[8]
		GAM_G_ GAIN_25[7]	GAM_G_ GAIN_25[6]	GAM_G_ GAIN_25[5]	GAM_G_ GAIN_25[4]	GAM_G_ GAIN_25[3]	GAM_G_ GAIN_25[2]	GAM_G_ GAIN_25[1]	GAM_G_ GAIN_25[0]	
	GAM_G_ LUT14	—	—	—	—	—	—	GAM_G_ GAIN_26[10]	GAM_G_ GAIN_26[9]	GAM_G_ GAIN_26[8]
		GAM_G_ GAIN_26[7]	GAM_G_ GAIN_26[6]	GAM_G_ GAIN_26[5]	GAM_G_ GAIN_26[4]	GAM_G_ GAIN_26[3]	GAM_G_ GAIN_26[2]	GAM_G_ GAIN_26[1]	GAM_G_ GAIN_26[0]	
		—	—	—	—	—	—	GAM_G_ GAIN_27[10]	GAM_G_ GAIN_27[9]	GAM_G_ GAIN_27[8]
		GAM_G_ GAIN_27[7]	GAM_G_ GAIN_27[6]	GAM_G_ GAIN_27[5]	GAM_G_ GAIN_27[4]	GAM_G_ GAIN_27[3]	GAM_G_ GAIN_27[2]	GAM_G_ GAIN_27[1]	GAM_G_ GAIN_27[0]	
	GAM_G_ LUT15	—	—	—	—	—	—	GAM_G_ GAIN_28[10]	GAM_G_ GAIN_28[9]	GAM_G_ GAIN_28[8]
		GAM_G_ GAIN_28[7]	GAM_G_ GAIN_28[6]	GAM_G_ GAIN_28[5]	GAM_G_ GAIN_28[4]	GAM_G_ GAIN_28[3]	GAM_G_ GAIN_28[2]	GAM_G_ GAIN_28[1]	GAM_G_ GAIN_28[0]	
		—	—	—	—	—	—	GAM_G_ GAIN_29[10]	GAM_G_ GAIN_29[9]	GAM_G_ GAIN_29[8]
		GAM_G_ GAIN_29[7]	GAM_G_ GAIN_29[6]	GAM_G_ GAIN_29[5]	GAM_G_ GAIN_29[4]	GAM_G_ GAIN_29[3]	GAM_G_ GAIN_29[2]	GAM_G_ GAIN_29[1]	GAM_G_ GAIN_29[0]	
	GAM_G_ LUT16	—	—	—	—	—	—	GAM_G_ GAIN_30[10]	GAM_G_ GAIN_30[9]	GAM_G_ GAIN_30[8]
		GAM_G_ GAIN_30[7]	GAM_G_ GAIN_30[6]	GAM_G_ GAIN_30[5]	GAM_G_ GAIN_30[4]	GAM_G_ GAIN_30[3]	GAM_G_ GAIN_30[2]	GAM_G_ GAIN_30[1]	GAM_G_ GAIN_30[0]	
		—	—	—	—	—	—	GAM_G_ GAIN_31[10]	GAM_G_ GAIN_31[9]	GAM_G_ GAIN_31[8]
		GAM_G_ GAIN_31[7]	GAM_G_ GAIN_31[6]	GAM_G_ GAIN_31[5]	GAM_G_ GAIN_31[4]	GAM_G_ GAIN_31[3]	GAM_G_ GAIN_31[2]	GAM_G_ GAIN_31[1]	GAM_G_ GAIN_31[0]	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	GAM_G_ AREA1	—	—	—	—	—	—	—	—
		GAM_G_TH _01[7]	GAM_G_TH _01[6]	GAM_G_TH _01[5]	GAM_G_TH _01[4]	GAM_G_TH _01[3]	GAM_G_TH _01[2]	GAM_G_TH _01[1]	GAM_G_TH _01[0]
		GAM_G_TH _02[7]	GAM_G_TH _02[6]	GAM_G_TH _02[5]	GAM_G_TH _02[4]	GAM_G_TH _02[3]	GAM_G_TH _02[2]	GAM_G_TH _02[1]	GAM_G_TH _02[0]
		GAM_G_TH _03[7]	GAM_G_TH _03[6]	GAM_G_TH _03[5]	GAM_G_TH _03[4]	GAM_G_TH _03[3]	GAM_G_TH _03[2]	GAM_G_TH _03[1]	GAM_G_TH _03[0]
	GAM_G_ AREA2	GAM_G_TH _04[7]	GAM_G_TH _04[6]	GAM_G_TH _04[5]	GAM_G_TH _04[4]	GAM_G_TH _04[3]	GAM_G_TH _04[2]	GAM_G_TH _04[1]	GAM_G_TH _04[0]
		GAM_G_TH _05[7]	GAM_G_TH _05[6]	GAM_G_TH _05[5]	GAM_G_TH _05[4]	GAM_G_TH _05[3]	GAM_G_TH _05[2]	GAM_G_TH _05[1]	GAM_G_TH _05[0]
		GAM_G_TH _06[7]	GAM_G_TH _06[6]	GAM_G_TH _06[5]	GAM_G_TH _06[4]	GAM_G_TH _06[3]	GAM_G_TH _06[2]	GAM_G_TH _06[1]	GAM_G_TH _06[0]
		GAM_G_TH _07[7]	GAM_G_TH _07[6]	GAM_G_TH _07[5]	GAM_G_TH _07[4]	GAM_G_TH _07[3]	GAM_G_TH _07[2]	GAM_G_TH _07[1]	GAM_G_TH _07[0]
	GAM_G_ AREA3	GAM_G_TH _08[7]	GAM_G_TH _08[6]	GAM_G_TH _08[5]	GAM_G_TH _08[4]	GAM_G_TH _08[3]	GAM_G_TH _08[2]	GAM_G_TH _08[1]	GAM_G_TH _08[0]
		GAM_G_TH _09[7]	GAM_G_TH _09[6]	GAM_G_TH _09[5]	GAM_G_TH _09[4]	GAM_G_TH _09[3]	GAM_G_TH _09[2]	GAM_G_TH _09[1]	GAM_G_TH _09[0]
		GAM_G_TH _10[7]	GAM_G_TH _10[6]	GAM_G_TH _10[5]	GAM_G_TH _10[4]	GAM_G_TH _10[3]	GAM_G_TH _10[2]	GAM_G_TH _10[1]	GAM_G_TH _10[0]
		GAM_G_TH _11[7]	GAM_G_TH _11[6]	GAM_G_TH _11[5]	GAM_G_TH _11[4]	GAM_G_TH _11[3]	GAM_G_TH _11[2]	GAM_G_TH _11[1]	GAM_G_TH _11[0]
	GAM_G_ AREA4	GAM_G_TH _12[7]	GAM_G_TH _12[6]	GAM_G_TH _12[5]	GAM_G_TH _12[4]	GAM_G_TH _12[3]	GAM_G_TH _12[2]	GAM_G_TH _12[1]	GAM_G_TH _12[0]
		GAM_G_TH _13[7]	GAM_G_TH _13[6]	GAM_G_TH _13[5]	GAM_G_TH _13[4]	GAM_G_TH _13[3]	GAM_G_TH _13[2]	GAM_G_TH _13[1]	GAM_G_TH _13[0]
		GAM_G_TH _14[7]	GAM_G_TH _14[6]	GAM_G_TH _14[5]	GAM_G_TH _14[4]	GAM_G_TH _14[3]	GAM_G_TH _14[2]	GAM_G_TH _14[1]	GAM_G_TH _14[0]
		GAM_G_TH _15[7]	GAM_G_TH _15[6]	GAM_G_TH _15[5]	GAM_G_TH _15[4]	GAM_G_TH _15[3]	GAM_G_TH _15[2]	GAM_G_TH _15[1]	GAM_G_TH _15[0]
	GAM_G_ AREA5	GAM_G_TH _16[7]	GAM_G_TH _16[6]	GAM_G_TH _16[5]	GAM_G_TH _16[4]	GAM_G_TH _16[3]	GAM_G_TH _16[2]	GAM_G_TH _16[1]	GAM_G_TH _16[0]
		GAM_G_TH _17[7]	GAM_G_TH _17[6]	GAM_G_TH _17[5]	GAM_G_TH _17[4]	GAM_G_TH _17[3]	GAM_G_TH _17[2]	GAM_G_TH _17[1]	GAM_G_TH _17[0]
		GAM_G_TH _18[7]	GAM_G_TH _18[6]	GAM_G_TH _18[5]	GAM_G_TH _18[4]	GAM_G_TH _18[3]	GAM_G_TH _18[2]	GAM_G_TH _18[1]	GAM_G_TH _18[0]
		GAM_G_TH _19[7]	GAM_G_TH _19[6]	GAM_G_TH _19[5]	GAM_G_TH _19[4]	GAM_G_TH _19[3]	GAM_G_TH _19[2]	GAM_G_TH _19[1]	GAM_G_TH _19[0]
	GAM_G_ AREA6	GAM_G_TH _20[7]	GAM_G_TH _20[6]	GAM_G_TH _20[5]	GAM_G_TH _20[4]	GAM_G_TH _20[3]	GAM_G_TH _20[2]	GAM_G_TH _20[1]	GAM_G_TH _20[0]
		GAM_G_TH _21[7]	GAM_G_TH _21[6]	GAM_G_TH _21[5]	GAM_G_TH _21[4]	GAM_G_TH _21[3]	GAM_G_TH _21[2]	GAM_G_TH _21[1]	GAM_G_TH _21[0]
		GAM_G_TH _22[7]	GAM_G_TH _22[6]	GAM_G_TH _22[5]	GAM_G_TH _22[4]	GAM_G_TH _22[3]	GAM_G_TH _22[2]	GAM_G_TH _22[1]	GAM_G_TH _22[0]
		GAM_G_TH _23[7]	GAM_G_TH _23[6]	GAM_G_TH _23[5]	GAM_G_TH _23[4]	GAM_G_TH _23[3]	GAM_G_TH _23[2]	GAM_G_TH _23[1]	GAM_G_TH _23[0]
	GAM_G_ AREA7	GAM_G_TH _24[7]	GAM_G_TH _24[6]	GAM_G_TH _24[5]	GAM_G_TH _24[4]	GAM_G_TH _24[3]	GAM_G_TH _24[2]	GAM_G_TH _24[1]	GAM_G_TH _24[0]
		GAM_G_TH _25[7]	GAM_G_TH _25[6]	GAM_G_TH _25[5]	GAM_G_TH _25[4]	GAM_G_TH _25[3]	GAM_G_TH _25[2]	GAM_G_TH _25[1]	GAM_G_TH _25[0]
		GAM_G_TH _26[7]	GAM_G_TH _26[6]	GAM_G_TH _26[5]	GAM_G_TH _26[4]	GAM_G_TH _26[3]	GAM_G_TH _26[2]	GAM_G_TH _26[1]	GAM_G_TH _26[0]
		GAM_G_TH _27[7]	GAM_G_TH _27[6]	GAM_G_TH _27[5]	GAM_G_TH _27[4]	GAM_G_TH _27[3]	GAM_G_TH _27[2]	GAM_G_TH _27[1]	GAM_G_TH _27[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
视频显示 控制器 4	GAM_G_ AREA8	GAM_G_TH _28[7]	GAM_G_TH _28[6]	GAM_G_TH _28[5]	GAM_G_TH _28[4]	GAM_G_TH _28[3]	GAM_G_TH _28[2]	GAM_G_TH _28[1]	GAM_G_TH _28[0]	
		GAM_G_TH _29[7]	GAM_G_TH _29[6]	GAM_G_TH _29[5]	GAM_G_TH _29[4]	GAM_G_TH _29[3]	GAM_G_TH _29[2]	GAM_G_TH _29[1]	GAM_G_TH _29[0]	
		GAM_G_TH _30[7]	GAM_G_TH _30[6]	GAM_G_TH _30[5]	GAM_G_TH _30[4]	GAM_G_TH _30[3]	GAM_G_TH _30[2]	GAM_G_TH _30[1]	GAM_G_TH _30[0]	
		GAM_G_TH _31[7]	GAM_G_TH _31[6]	GAM_G_TH _31[5]	GAM_G_TH _31[4]	GAM_G_TH _31[3]	GAM_G_TH _31[2]	GAM_G_TH _31[1]	GAM_G_TH _31[0]	
	GAM_B_ UPDATE	—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	GAM_B_VEN
	GAM_B_ LUT1	—	—	—	—	—	GAM_B_ GAIN_00[10]	GAM_B_ GAIN_00[9]	GAM_B_ GAIN_00[8]	GAM_B_ GAIN_00[7]
		GAM_B_ GAIN_00[7]	GAM_B_ GAIN_00[6]	GAM_B_ GAIN_00[5]	GAM_B_ GAIN_00[4]	GAM_B_ GAIN_00[3]	GAM_B_ GAIN_00[2]	GAM_B_ GAIN_00[1]	GAM_B_ GAIN_00[0]	GAM_B_ GAIN_00[0]
		—	—	—	—	—	GAM_B_ GAIN_01[10]	GAM_B_ GAIN_01[9]	GAM_B_ GAIN_01[8]	GAM_B_ GAIN_01[7]
		GAM_B_ GAIN_01[7]	GAM_B_ GAIN_01[6]	GAM_B_ GAIN_01[5]	GAM_B_ GAIN_01[4]	GAM_B_ GAIN_01[3]	GAM_B_ GAIN_01[2]	GAM_B_ GAIN_01[1]	GAM_B_ GAIN_01[0]	GAM_B_ GAIN_01[0]
	GAM_B_ LUT2	—	—	—	—	—	GAM_B_GA IN_02[10]	GAM_B_GA IN_02[9]	GAM_B_GA IN_02[8]	GAM_B_GA IN_02[7]
		GAM_B_GA IN_02[7]	GAM_B_GA IN_02[6]	GAM_B_GA IN_02[5]	GAM_B_GA IN_02[4]	GAM_B_GA IN_02[3]	GAM_B_GA IN_02[2]	GAM_B_GA IN_02[1]	GAM_B_GA IN_02[0]	GAM_B_GA IN_02[0]
		—	—	—	—	—	GAM_B_GA IN_03[10]	GAM_B_GA IN_03[9]	GAM_B_GA IN_03[8]	GAM_B_GA IN_03[7]
		GAM_B_GA IN_03[7]	GAM_B_GA IN_03[6]	GAM_B_GA IN_03[5]	GAM_B_GA IN_03[4]	GAM_B_GA IN_03[3]	GAM_B_GA IN_03[2]	GAM_B_GA IN_03[1]	GAM_B_GA IN_03[0]	GAM_B_GA IN_03[0]
	GAM_B_ LUT3	—	—	—	—	—	GAM_B_GA IN_04[10]	GAM_B_GA IN_04[9]	GAM_B_GA IN_04[8]	GAM_B_GA IN_04[7]
		GAM_B_GA IN_04[7]	GAM_B_GA IN_04[6]	GAM_B_GA IN_04[5]	GAM_B_GA IN_04[4]	GAM_B_GA IN_04[3]	GAM_B_GA IN_04[2]	GAM_B_GA IN_04[1]	GAM_B_GA IN_04[0]	GAM_B_GA IN_04[0]
		—	—	—	—	—	GAM_B_GA IN_05[10]	GAM_B_GA IN_05[9]	GAM_B_GA IN_05[8]	GAM_B_GA IN_05[7]
		GAM_B_GA IN_05[7]	GAM_B_GA IN_05[6]	GAM_B_GA IN_05[5]	GAM_B_GA IN_05[4]	GAM_B_GA IN_05[3]	GAM_B_GA IN_05[2]	GAM_B_GA IN_05[1]	GAM_B_GA IN_05[0]	GAM_B_GA IN_05[0]
	GAM_B_ LUT4	—	—	—	—	—	GAM_B_GA IN_06[10]	GAM_B_GA IN_06[9]	GAM_B_GA IN_06[8]	GAM_B_GA IN_06[7]
		GAM_B_GA IN_06[7]	GAM_B_GA IN_06[6]	GAM_B_GA IN_06[5]	GAM_B_GA IN_06[4]	GAM_B_GA IN_06[3]	GAM_B_GA IN_06[2]	GAM_B_GA IN_06[1]	GAM_B_GA IN_06[0]	GAM_B_GA IN_06[0]
		—	—	—	—	—	GAM_B_GA IN_07[10]	GAM_B_GA IN_07[9]	GAM_B_GA IN_07[8]	GAM_B_GA IN_07[7]
		GAM_B_GA IN_07[7]	GAM_B_GA IN_07[6]	GAM_B_GA IN_07[5]	GAM_B_GA IN_07[4]	GAM_B_GA IN_07[3]	GAM_B_GA IN_07[2]	GAM_B_GA IN_07[1]	GAM_B_GA IN_07[0]	GAM_B_GA IN_07[0]
	GAM_B_ LUT5	—	—	—	—	—	GAM_B_GA IN_08[10]	GAM_B_GA IN_08[9]	GAM_B_GA IN_08[8]	GAM_B_GA IN_08[7]
		GAM_B_GA IN_08[7]	GAM_B_GA IN_08[6]	GAM_B_GA IN_08[5]	GAM_B_GA IN_08[4]	GAM_B_GA IN_08[3]	GAM_B_GA IN_08[2]	GAM_B_GA IN_08[1]	GAM_B_GA IN_08[0]	GAM_B_GA IN_08[0]
		—	—	—	—	—	GAM_B_GA IN_09[10]	GAM_B_GA IN_09[9]	GAM_B_GA IN_09[8]	GAM_B_GA IN_09[7]
		GAM_B_GA IN_09[7]	GAM_B_GA IN_09[6]	GAM_B_GA IN_09[5]	GAM_B_GA IN_09[4]	GAM_B_GA IN_09[3]	GAM_B_GA IN_09[2]	GAM_B_GA IN_09[1]	GAM_B_GA IN_09[0]	GAM_B_GA IN_09[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	GAM_B_ LUT6	—	—	—	—	—	GAM_B_GA IN_10[10]	GAM_B_GA IN_10[9]	GAM_B_GA IN_10[8]
		GAM_B_GA IN_10[7]	GAM_B_GA IN_10[6]	GAM_B_GA IN_10[5]	GAM_B_GA IN_10[4]	GAM_B_GA IN_10[3]	GAM_B_GA IN_10[2]	GAM_B_GA IN_10[1]	GAM_B_GA IN_10[0]
		—	—	—	—	—	GAM_B_GA IN_11[10]	GAM_B_GA IN_11[9]	GAM_B_GA IN_11[8]
		GAM_B_GA IN_11[7]	GAM_B_GA IN_11[6]	GAM_B_GA IN_11[5]	GAM_B_GA IN_11[4]	GAM_B_GA IN_11[3]	GAM_B_GA IN_11[2]	GAM_B_GA IN_11[1]	GAM_B_GA IN_11[0]
	GAM_B_ LUT7	—	—	—	—	—	GAM_B_ GAIN_12[10]	GAM_B_ GAIN_12[9]	GAM_B_ GAIN_12[8]
		GAM_B_ GAIN_12[7]	GAM_B_ GAIN_12[6]	GAM_B_ GAIN_12[5]	GAM_B_ GAIN_12[4]	GAM_B_ GAIN_12[3]	GAM_B_ GAIN_12[2]	GAM_B_ GAIN_12[1]	GAM_B_ GAIN_12[0]
		—	—	—	—	—	GAM_B_ GAIN_13[10]	GAM_B_ GAIN_13[9]	GAM_B_ GAIN_13[8]
		GAM_B_ GAIN_13[7]	GAM_B_ GAIN_13[6]	GAM_B_ GAIN_13[5]	GAM_B_ GAIN_13[4]	GAM_B_ GAIN_13[3]	GAM_B_ GAIN_13[2]	GAM_B_ GAIN_13[1]	GAM_B_ GAIN_13[0]
	GAM_B_ LUT8	—	—	—	—	—	GAM_B_ GAIN_14[10]	GAM_B_ GAIN_14[9]	GAM_B_ GAIN_14[8]
		GAM_B_ GAIN_14[7]	GAM_B_ GAIN_14[6]	GAM_B_ GAIN_14[5]	GAM_B_ GAIN_14[4]	GAM_B_ GAIN_14[3]	GAM_B_ GAIN_14[2]	GAM_B_ GAIN_14[1]	GAM_B_ GAIN_14[0]
		—	—	—	—	—	GAM_B_ GAIN_15[10]	GAM_B_ GAIN_15[9]	GAM_B_ GAIN_15[8]
		GAM_B_ GAIN_15[7]	GAM_B_ GAIN_15[6]	GAM_B_ GAIN_15[5]	GAM_B_ GAIN_15[4]	GAM_B_ GAIN_15[3]	GAM_B_ GAIN_15[2]	GAM_B_ GAIN_15[1]	GAM_B_ GAIN_15[0]
	GAM_B_ LUT9	—	—	—	—	—	GAM_B_ GAIN_16[10]	GAM_B_ GAIN_16[9]	GAM_B_ GAIN_16[8]
		GAM_B_ GAIN_16[7]	GAM_B_ GAIN_16[6]	GAM_B_ GAIN_16[5]	GAM_B_ GAIN_16[4]	GAM_B_ GAIN_16[3]	GAM_B_G AIN_16[2]	GAM_B_ GAIN_16[1]	GAM_B_ GAIN_16[0]
		—	—	—	—	—	GAM_B_ GAIN_17[10]	GAM_B_ GAIN_17[9]	GAM_B_ GAIN_17[8]
		GAM_B_ GAIN_17[7]	GAM_B_ GAIN_17[6]	GAM_B_ GAIN_17[5]	GAM_B_ GAIN_17[4]	GAM_B_ GAIN_17[3]	GAM_B_ GAIN_17[2]	GAM_B_ GAIN_17[1]	GAM_B_ GAIN_17[0]
	GAM_B_ LUT10	—	—	—	—	—	GAM_B_ GAIN_18[10]	GAM_B_ GAIN_18[9]	GAM_B_ GAIN_18[8]
		GAM_B_GA IN_18[7]	GAM_B_GA IN_18[6]	GAM_B_GA IN_18[5]	GAM_B_GA IN_18[4]	GAM_B_GA IN_18[3]	GAM_B_ GAIN_18[2]	GAM_B_ GAIN_18[1]	GAM_B_ GAIN_18[0]
		—	—	—	—	—	GAM_B_ GAIN_19[10]	GAM_B_ GAIN_19[9]	GAM_B_ GAIN_19[8]
		GAM_B_ GAIN_19[7]	GAM_B_ GAIN_19[6]	GAM_B_ GAIN_19[5]	GAM_B_ GAIN_19[4]	GAM_B_ GAIN_19[3]	GAM_B_ GAIN_19[2]	GAM_B_ GAIN_19[1]	GAM_B_ GAIN_19[0]
	GAM_B_ LUT11	—	—	—	—	—	GAM_B_ GAIN_20[10]	GAM_B_ GAIN_20[9]	GAM_B_ GAIN_20[8]
		GAM_B_ GAIN_20[7]	GAM_B_ GAIN_20[6]	GAM_B_GA IN_20[5]	GAM_B_GA IN_20[4]	GAM_B_GA IN_20[3]	GAM_B_ GAIN_20[2]	GAM_B_ GAIN_20[1]	GAM_B_ GAIN_20[0]
		—	—	—	—	—	GAM_B_ GAIN_21[10]	GAM_B_ GAIN_21[9]	GAM_B_ GAIN_21[8]
		GAM_B_ GAIN_21[7]	GAM_B_ GAIN_21[6]	GAM_B_ GAIN_21[5]	GAM_B_ GAIN_21[4]	GAM_B_ GAIN_21[3]	GAM_B_ GAIN_21[2]	GAM_B_ GAIN_21[1]	GAM_B_ GAIN_21[0]
	GAM_B_ LUT12	—	—	—	—	—	GAM_B_ GAIN_22[10]	GAM_B_ GAIN_22[9]	GAM_B_ GAIN_22[8]
		GAM_B_ GAIN_22[7]	GAM_B_ GAIN_22[6]	GAM_B_ GAIN_22[5]	GAM_B_ GAIN_22[4]	GAM_B_ GAIN_22[3]	GAM_B_ GAIN_22[2]	GAM_B_ GAIN_22[1]	GAM_B_ GAIN_22[0]
		—	—	—	—	—	GAM_B_ GAIN_23[10]	GAM_B_ GAIN_23[9]	GAM_B_ GAIN_23[8]
		GAM_B_ GAIN_23[7]	GAM_B_ GAIN_23[6]	GAM_B_ GAIN_23[5]	GAM_B_ GAIN_23[4]	GAM_B_ GAIN_23[3]	GAM_B_ GAIN_23[2]	GAM_B_ GAIN_23[1]	GAM_B_ GAIN_23[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	GAM_B_ LUT13	—	—	—	—	—	GAM_B_ GAIN_24[10]	GAM_B_ GAIN_24[9]	GAM_B_ GAIN_24[8]
		GAM_B_ GAIN_24[7]	GAM_B_ GAIN_24[6]	GAM_B_ GAIN_24[5]	GAM_B_ GAIN_24[4]	GAM_B_ GAIN_24[3]	GAM_B_ GAIN_24[2]	GAM_B_ GAIN_24[1]	GAM_B_ GAIN_24[0]
		—	—	—	—	—	GAM_B_ GAIN_25[10]	GAM_B_ GAIN_25[9]	GAM_B_ GAIN_25[8]
		GAM_B_ GAIN_25[7]	GAM_B_ GAIN_25[6]	GAM_B_ GAIN_25[5]	GAM_B_ GAIN_25[4]	GAM_B_ GAIN_25[3]	GAM_B_ GAIN_25[2]	GAM_B_ GAIN_25[1]	GAM_B_ GAIN_25[0]
	GAM_B_ LUT14	—	—	—	—	—	GAM_B_ GAIN_26[10]	GAM_B_ GAIN_26[9]	GAM_B_ GAIN_26[8]
		GAM_B_ GAIN_26[7]	GAM_B_ GAIN_26[6]	GAM_B_ GAIN_26[5]	GAM_B_ GAIN_26[4]	GAM_B_ GAIN_26[3]	GAM_B_ GAIN_26[2]	GAM_B_ GAIN_26[1]	GAM_B_ GAIN_26[0]
		—	—	—	—	—	GAM_B_ GAIN_27[10]	GAM_B_ GAIN_27[9]	GAM_B_ GAIN_27[8]
		GAM_B_ GAIN_27[7]	GAM_B_ GAIN_27[6]	GAM_B_ GAIN_27[5]	GAM_B_ GAIN_27[4]	GAM_B_ GAIN_27[3]	GAM_B_ GAIN_27[2]	GAM_B_ GAIN_27[1]	GAM_B_ GAIN_27[0]
	GAM_B_ LUT15	—	—	—	—	—	GAM_B_ GAIN_28[10]	GAM_B_ GAIN_28[9]	GAM_B_ GAIN_28[8]
		GAM_B_ GAIN_28[7]	GAM_B_ GAIN_28[6]	GAM_B_ GAIN_28[5]	GAM_B_ GAIN_28[4]	GAM_B_ GAIN_28[3]	GAM_B_ GAIN_28[2]	GAM_B_ GAIN_28[1]	GAM_B_ GAIN_28[0]
		—	—	—	—	—	GAM_B_ GAIN_29[10]	GAM_B_ GAIN_29[9]	GAM_B_ GAIN_29[8]
		GAM_B_ GAIN_29[7]	GAM_B_ GAIN_29[6]	GAM_B_ GAIN_29[5]	GAM_B_ GAIN_29[4]	GAM_B_ GAIN_29[3]	GAM_B_GA IN_29[2]	GAM_B_ GAIN_29[1]	GAM_B_ GAIN_29[0]
	GAM_B_ LUT16	—	—	—	—	—	GAM_B_ GAIN_30[10]	GAM_B_ GAIN_30[9]	GAM_B_ GAIN_30[8]
		GAM_B_ GAIN_30[7]	GAM_B_ GAIN_30[6]	GAM_B_ GAIN_30[5]	GAM_B_ GAIN_30[4]	GAM_B_ GAIN_30[3]	GAM_B_ GAIN_30[2]	GAM_B_ GAIN_30[1]	GAM_B_ GAIN_30[0]
		—	—	—	—	—	GAM_B_ GAIN_31[10]	GAM_B_ GAIN_31[9]	GAM_B_ GAIN_31[8]
		GAM_B_ GAIN_31[7]	GAM_B_ GAIN_31[6]	GAM_B_ GAIN_31[5]	GAM_B_ GAIN_31[4]	GAM_B_ GAIN_31[3]	GAM_B_ GAIN_31[2]	GAM_B_ GAIN_31[1]	GAM_B_ GAIN_31[0]
	GAM_B_ AREA1	—	—	—	—	—	—	—	—
		GAM_B_TH _01[7]	GAM_B_TH _01[6]	GAM_B_TH _01[5]	GAM_B_TH _01[4]	GAM_B_TH _01[3]	GAM_B_TH _01[2]	GAM_B_TH _01[1]	GAM_B_TH _01[0]
		GAM_B_TH _02[7]	GAM_B_TH _02[6]	GAM_B_TH _02[5]	GAM_B_TH _02[4]	GAM_B_TH _02[3]	GAM_B_TH _02[2]	GAM_B_TH _02[1]	GAM_B_TH _02[0]
		GAM_B_TH _03[7]	GAM_B_TH _03[6]	GAM_B_TH _03[5]	GAM_B_TH _03[4]	GAM_B_TH _03[3]	GAM_B_TH _03[2]	GAM_B_TH _03[1]	GAM_B_TH _03[0]
	GAM_B_ AREA2	GAM_B_TH _04[7]	GAM_B_TH _04[6]	GAM_B_TH _04[5]	GAM_B_TH _04[4]	GAM_B_TH _04[3]	GAM_B_TH _04[2]	GAM_B_TH _04[1]	GAM_B_TH _04[0]
		GAM_B_TH _05[7]	GAM_B_TH _05[6]	GAM_B_TH _05[5]	GAM_B_TH _05[4]	GAM_B_TH _05[3]	GAM_B_TH _05[2]	GAM_B_TH _05[1]	GAM_B_TH _05[0]
		GAM_B_TH _06[7]	GAM_B_TH _06[6]	GAM_B_TH _06[5]	GAM_B_TH _06[4]	GAM_B_TH _06[3]	GAM_B_TH _06[2]	GAM_B_TH _06[1]	GAM_B_TH _06[0]
		GAM_B_TH _07[7]	GAM_B_TH _07[6]	GAM_B_TH _07[5]	GAM_B_TH _07[4]	GAM_B_TH _07[3]	GAM_B_TH _07[2]	GAM_B_TH _07[1]	GAM_B_TH _07[0]
	GAM_B_ AREA3	GAM_B_TH _08[7]	GAM_B_TH _08[6]	GAM_B_TH _08[5]	GAM_B_TH _08[4]	GAM_B_TH _08[3]	GAM_B_TH _08[2]	GAM_B_TH _08[1]	GAM_B_TH _08[0]
		GAM_B_TH _09[7]	GAM_B_TH _09[6]	GAM_B_TH _09[5]	GAM_B_TH _09[4]	GAM_B_TH _09[3]	GAM_B_TH _09[2]	GAM_B_TH _09[1]	GAM_B_TH _09[0]
		GAM_B_TH _10[7]	GAM_B_TH _10[6]	GAM_B_TH _10[5]	GAM_B_TH _10[4]	GAM_B_TH _10[3]	GAM_B_TH _10[2]	GAM_B_TH _10[1]	GAM_B_TH _10[0]
		GAM_B_TH _11[7]	GAM_B_TH _11[6]	GAM_B_TH _11[5]	GAM_B_TH _11[4]	GAM_B_TH _11[3]	GAM_B_TH _11[2]	GAM_B_TH _11[1]	GAM_B_TH _11[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	GAM_B_ AREA4	GAM_B_TH _12[7]	GAM_B_TH _12[6]	GAM_B_TH _12[5]	GAM_B_TH _12[4]	GAM_B_TH _12[3]	GAM_B_TH _12[2]	GAM_B_TH _12[1]	GAM_B_TH _12[0]
		GAM_B_TH _13[7]	GAM_B_TH _13[6]	GAM_B_TH _13[5]	GAM_B_TH _13[4]	GAM_B_TH _13[3]	GAM_B_TH _13[2]	GAM_B_TH _13[1]	GAM_B_TH _13[0]
		GAM_B_TH _14[7]	GAM_B_TH _14[6]	GAM_B_TH _14[5]	GAM_B_TH _14[4]	GAM_B_TH _14[3]	GAM_B_TH _14[2]	GAM_B_TH _14[1]	GAM_B_TH _14[0]
		GAM_B_TH _15[7]	GAM_B_TH _15[6]	GAM_B_TH _15[5]	GAM_B_TH _15[4]	GAM_B_TH _15[3]	GAM_B_TH _15[2]	GAM_B_TH _15[1]	GAM_B_TH _15[0]
	GAM_B_ AREA5	GAM_B_TH _16[7]	GAM_B_TH _16[6]	GAM_B_TH _16[5]	GAM_B_TH _16[4]	GAM_B_TH _16[3]	GAM_B_TH _16[2]	GAM_B_TH _16[1]	GAM_B_TH _16[0]
		GAM_B_TH _17[7]	GAM_B_TH _17[6]	GAM_B_TH _17[5]	GAM_B_TH _17[4]	GAM_B_TH _17[3]	GAM_B_TH _17[2]	GAM_B_TH _17[1]	GAM_B_TH _17[0]
		GAM_B_TH _18[7]	GAM_B_TH _18[6]	GAM_B_TH _18[5]	GAM_B_TH _18[4]	GAM_B_TH _18[3]	GAM_B_TH _18[2]	GAM_B_TH _18[1]	GAM_B_TH _18[0]
		GAM_B_TH _19[7]	GAM_B_TH _19[6]	GAM_B_TH _19[5]	GAM_B_TH _19[4]	GAM_B_TH _19[3]	GAM_B_TH _19[2]	GAM_B_TH _19[1]	GAM_B_TH _19[0]
	GAM_B_ AREA6	GAM_B_TH _20[7]	GAM_B_TH _20[6]	GAM_B_TH _20[5]	GAM_B_TH _20[4]	GAM_B_TH _20[3]	GAM_B_TH _20[2]	GAM_B_TH _20[1]	GAM_B_TH _20[0]
		GAM_B_TH _21[7]	GAM_B_TH _21[6]	GAM_B_TH _21[5]	GAM_B_TH _21[4]	GAM_B_TH _21[3]	GAM_B_TH _21[2]	GAM_B_TH _21[1]	GAM_B_TH _21[0]
		GAM_B_TH _22[7]	GAM_B_TH _22[6]	GAM_B_TH _22[5]	GAM_B_TH _22[4]	GAM_B_TH _22[3]	GAM_B_TH _22[2]	GAM_B_TH _22[1]	GAM_B_TH _22[0]
		GAM_B_TH _23[7]	GAM_B_TH _23[6]	GAM_B_TH _23[5]	GAM_B_TH _23[4]	GAM_B_TH _23[3]	GAM_B_TH _23[2]	GAM_B_TH _23[1]	GAM_B_TH _23[0]
	GAM_B_ AREA7	GAM_B_TH _24[7]	GAM_B_TH _24[6]	GAM_B_TH _24[5]	GAM_B_TH _24[4]	GAM_B_TH _24[3]	GAM_B_TH _24[2]	GAM_B_TH _24[1]	GAM_B_TH _24[0]
		GAM_B_TH _25[7]	GAM_B_TH _25[6]	GAM_B_TH _25[5]	GAM_B_TH _25[4]	GAM_B_TH _25[3]	GAM_B_TH _25[2]	GAM_B_TH _25[1]	GAM_B_TH _25[0]
		GAM_B_TH _26[7]	GAM_B_TH _26[6]	GAM_B_TH _26[5]	GAM_B_TH _26[4]	GAM_B_TH _26[3]	GAM_B_TH _26[2]	GAM_B_TH _26[1]	GAM_B_TH _26[0]
		GAM_B_TH _27[7]	GAM_B_TH _27[6]	GAM_B_TH _27[5]	GAM_B_TH _27[4]	GAM_B_TH _27[3]	GAM_B_TH _27[2]	GAM_B_TH _27[1]	GAM_B_TH _27[0]
	GAM_B_ AREA8	GAM_B_TH _28[7]	GAM_B_TH _28[6]	GAM_B_TH _28[5]	GAM_B_TH _28[4]	GAM_B_TH _28[3]	GAM_B_TH _28[2]	GAM_B_TH _28[1]	GAM_B_TH _28[0]
		GAM_B_TH _29[7]	GAM_B_TH _29[6]	GAM_B_TH _29[5]	GAM_B_TH _29[4]	GAM_B_TH _29[3]	GAM_B_TH _29[2]	GAM_B_TH _29[1]	GAM_B_TH _29[0]
		GAM_B_TH _30[7]	GAM_B_TH _30[6]	GAM_B_TH _30[5]	GAM_B_TH _30[4]	GAM_B_TH _30[3]	GAM_B_TH _30[2]	GAM_B_TH _30[1]	GAM_B_TH _30[0]
		GAM_B_TH _31[7]	GAM_B_TH _31[6]	GAM_B_TH _31[5]	GAM_B_TH _31[4]	GAM_B_TH _31[3]	GAM_B_TH _31[2]	GAM_B_TH _31[1]	GAM_B_TH _31[0]
	GAM_R_ UPDATE	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	GAM_R_VE N
	GAM_R_ LUT1	—	—	—	—	—	GAM_R_ GAIN_00[10]	GAM_R_ GAIN_00[9]	GAM_R_ GAIN_00[8]
		GAM_R_ GAIN_00[7]	GAM_R_ GAIN_00[6]	GAM_R_ GAIN_00[5]	GAM_R_ GAIN_00[4]	GAM_R_ GAIN_00[3]	GAM_R_ GAIN_00[2]	GAM_R_ GAIN_00[1]	GAM_R_ GAIN_00[0]
		—	—	—	—	—	GAM_R_ GAIN_01[10]	GAM_R_ GAIN_01[9]	GAM_R_ GAIN_01[8]
		GAM_R_ GAIN_01[7]	GAM_R_ GAIN_01[6]	GAM_R_ GAIN_01[5]	GAM_R_ GAIN_01[4]	GAM_R_ GAIN_01[3]	GAM_R_ GAIN_01[2]	GAM_R_ GAIN_01[1]	GAM_R_ GAIN_01[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	GAM_R_ LUT2	—	—	—	—	—	GAM_R_ GAIN_02[10]	GAM_R_ GAIN_02[9]	GAM_R_ GAIN_02[8]
		GAM_R_ GAIN_02[7]	GAM_R_ GAIN_02[6]	GAM_R_ GAIN_02[5]	GAM_R_ GAIN_02[4]	GAM_R_ GAIN_02[3]	GAM_R_ GAIN_02[2]	GAM_R_ GAIN_02[1]	GAM_R_ GAIN_02[0]
		—	—	—	—	—	GAM_R_ GAIN_03[10]	GAM_R_ GAIN_03[9]	GAM_R_ GAIN_03[8]
		GAM_R_ GAIN_03[7]	GAM_R_ GAIN_03[6]	GAM_R_ GAIN_03[5]	GAM_R_ GAIN_03[4]	GAM_R_ GAIN_03[3]	GAM_R_ GAIN_03[2]	GAM_R_ GAIN_03[1]	GAM_R_ GAIN_03[0]
	GAM_R_ LUT3	—	—	—	—	—	GAM_R_ GAIN_04[10]	GAM_R_ GAIN_04[9]	GAM_R_ GAIN_04[8]
		GAM_R_ GAIN_04[7]	GAM_R_ GAIN_04[6]	GAM_R_ GAIN_04[5]	GAM_R_ GAIN_04[4]	GAM_R_ GAIN_04[3]	GAM_R_ GAIN_04[2]	GAM_R_ GAIN_04[1]	GAM_R_ GAIN_04[0]
		—	—	—	—	—	GAM_R_ GAIN_05[10]	GAM_R_ GAIN_05[9]	GAM_R_ GAIN_05[8]
		GAM_R_ GAIN_05[7]	GAM_R_ GAIN_05[6]	GAM_R_ GAIN_05[5]	GAM_R_ GAIN_05[4]	GAM_R_ GAIN_05[3]	GAM_R_ GAIN_05[2]	GAM_R_ GAIN_05[1]	GAM_R_ GAIN_05[0]
	GAM_R_ LUT4	—	—	—	—	—	GAM_R_ GAIN_06[10]	GAM_R_ GAIN_06[9]	GAM_R_ GAIN_06[8]
		GAM_R_ GAIN_06[7]	GAM_R_ GAIN_06[6]	GAM_R_ GAIN_06[5]	GAM_R_ GAIN_06[4]	GAM_R_ GAIN_06[3]	GAM_R_ GAIN_06[2]	GAM_R_ GAIN_06[1]	GAM_R_ GAIN_06[0]
		—	—	—	—	—	GAM_R_ GAIN_07[10]	GAM_R_ GAIN_07[9]	GAM_R_ GAIN_07[8]
		GAM_R_ GAIN_07[7]	GAM_R_ GAIN_07[6]	GAM_R_ GAIN_07[5]	GAM_R_ GAIN_07[4]	GAM_R_ GAIN_07[3]	GAM_R_ GAIN_07[2]	GAM_R_ GAIN_07[1]	GAM_R_ GAIN_07[0]
	GAM_R_ LUT5	—	—	—	—	—	GAM_R_ GAIN_08[10]	GAM_R_ GAIN_08[9]	GAM_R_ GAIN_08[8]
		GAM_R_ GAIN_08[7]	GAM_R_ GAIN_08[6]	GAM_R_ GAIN_08[5]	GAM_R_ GAIN_08[4]	GAM_R_ GAIN_08[3]	GAM_R_ GAIN_08[2]	GAM_R_ GAIN_08[1]	GAM_R_ GAIN_08[0]
		—	—	—	—	—	GAM_R_ GAIN_09[10]	GAM_R_ GAIN_09[9]	GAM_R_ GAIN_09[8]
		GAM_R_ GAIN_09[7]	GAM_R_ GAIN_09[6]	GAM_R_ GAIN_09[5]	GAM_R_ GAIN_09[4]	GAM_R_ GAIN_09[3]	GAM_R_ GAIN_09[2]	GAM_R_ GAIN_09[1]	GAM_R_ GAIN_09[0]
	GAM_R_ LUT6	—	—	—	—	—	GAM_R_ GAIN_10[10]	GAM_R_ GAIN_10[9]	GAM_R_ GAIN_10[8]
		GAM_R_ GAIN_10[7]	GAM_R_ GAIN_10[6]	GAM_R_ GAIN_10[5]	GAM_R_ GAIN_10[4]	GAM_R_ GAIN_10[3]	GAM_R_ GAIN_10[2]	GAM_R_ GAIN_10[1]	GAM_R_ GAIN_10[0]
		—	—	—	—	—	GAM_R_ GAIN_11[10]	GAM_R_ GAIN_11[9]	GAM_R_ GAIN_11[8]
		GAM_R_ GAIN_11[7]	GAM_R_ GAIN_11[6]	GAM_R_ GAIN_11[5]	GAM_R_ GAIN_11[4]	GAM_R_ GAIN_11[3]	GAM_R_ GAIN_11[2]	GAM_R_ GAIN_11[1]	GAM_R_ GAIN_11[0]
	GAM_R_ LUT7	—	—	—	—	—	GAM_R_ GAIN_12[10]	GAM_R_ GAIN_12[9]	GAM_R_ GAIN_12[8]
		GAM_R_ GAIN_12[7]	GAM_R_ GAIN_12[6]	GAM_R_ GAIN_12[5]	GAM_R_ GAIN_12[4]	GAM_R_ GAIN_12[3]	GAM_R_ GAIN_12[2]	GAM_R_ GAIN_12[1]	GAM_R_ GAIN_12[0]
		—	—	—	—	—	GAM_R_ GAIN_13[10]	GAM_R_ GAIN_13[9]	GAM_R_ GAIN_13[8]
		GAM_R_ GAIN_13[7]	GAM_R_ GAIN_13[6]	GAM_R_ GAIN_13[5]	GAM_R_ GAIN_13[4]	GAM_R_ GAIN_13[3]	GAM_R_ GAIN_13[2]	GAM_R_ GAIN_13[1]	GAM_R_ GAIN_13[0]
	GAM_R_ LUT8	—	—	—	—	—	GAM_R_ GAIN_14[10]	GAM_R_ GAIN_14[9]	GAM_R_ GAIN_14[8]
		GAM_R_ GAIN_14[7]	GAM_R_ GAIN_14[6]	GAM_R_ GAIN_14[5]	GAM_R_ GAIN_14[4]	GAM_R_ GAIN_14[3]	GAM_R_ GAIN_14[2]	GAM_R_ GAIN_14[1]	GAM_R_ GAIN_14[0]
		—	—	—	—	—	GAM_R_ GAIN_15[10]	GAM_R_ GAIN_15[9]	GAM_R_ GAIN_15[8]
		GAM_R_ GAIN_15[7]	GAM_R_ GAIN_15[6]	GAM_R_ GAIN_15[5]	GAM_R_ GAIN_15[4]	GAM_R_ GAIN_15[3]	GAM_R_ GAIN_15[2]	GAM_R_ GAIN_15[1]	GAM_R_ GAIN_15[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	GAM_R_ LUT9	—	—	—	—	—	GAM_R_ GAIN_16[10]	GAM_R_ GAIN_16[9]	GAM_R_ GAIN_16[8]
		GAM_R_ GAIN_16[7]	GAM_R_ GAIN_16[6]	GAM_R_ GAIN_16[5]	GAM_R_ GAIN_16[4]	GAM_R_ GAIN_16[3]	GAM_R_ GAIN_16[2]	GAM_R_ GAIN_16[1]	GAM_R_ GAIN_16[0]
		—	—	—	—	—	GAM_R_ GAIN_17[10]	GAM_R_ GAIN_17[9]	GAM_R_ GAIN_17[8]
		GAM_R_ GAIN_17[7]	GAM_R_ GAIN_17[6]	GAM_R_ GAIN_17[5]	GAM_R_ GAIN_17[4]	GAM_R_ GAIN_17[3]	GAM_R_ GAIN_17[2]	GAM_R_ GAIN_17[1]	GAM_R_ GAIN_17[0]
	GAM_R_ LUT10	—	—	—	—	—	GAM_R_ GAIN_18[10]	GAM_R_ GAIN_18[9]	GAM_R_ GAIN_18[8]
		GAM_R_ GAIN_18[7]	GAM_R_ GAIN_18[6]	GAM_R_ GAIN_18[5]	GAM_R_ GAIN_18[4]	GAM_R_ GAIN_18[3]	GAM_R_ GAIN_18[2]	GAM_R_ GAIN_18[1]	GAM_R_ GAIN_18[0]
		—	—	—	—	—	GAM_R_ GAIN_19[10]	GAM_R_ GAIN_19[9]	GAM_R_ GAIN_19[8]
		GAM_R_ GAIN_19[7]	GAM_R_ GAIN_19[6]	GAM_R_ GAIN_19[5]	GAM_R_ GAIN_19[4]	GAM_R_ GAIN_19[3]	GAM_R_ GAIN_19[2]	GAM_R_ GAIN_19[1]	GAM_R_ GAIN_19[0]
	GAM_R_ LUT11	—	—	—	—	—	GAM_R_ GAIN_20[10]	GAM_R_ GAIN_20[9]	GAM_R_ GAIN_20[8]
		GAM_R_ GAIN_20[7]	GAM_R_ GAIN_20[6]	GAM_R_ GAIN_20[5]	GAM_R_ GAIN_20[4]	GAM_R_ GAIN_20[3]	GAM_R_ GAIN_20[2]	GAM_R_ GAIN_20[1]	GAM_R_ GAIN_20[0]
		—	—	—	—	—	GAM_R_ GAIN_21[10]	GAM_R_ GAIN_21[9]	GAM_R_ GAIN_21[8]
		GAM_R_ GAIN_21[7]	GAM_R_ GAIN_21[6]	GAM_R_ GAIN_21[5]	GAM_R_ GAIN_21[4]	GAM_R_ GAIN_21[3]	GAM_R_ GAIN_21[2]	GAM_R_ GAIN_21[1]	GAM_R_ GAIN_21[0]
	GAM_R_ LUT12	—	—	—	—	—	GAM_R_ GAIN_22[10]	GAM_R_ GAIN_22[9]	GAM_R_ GAIN_22[8]
		GAM_R_ GAIN_22[7]	GAM_R_ GAIN_22[6]	GAM_R_ GAIN_22[5]	GAM_R_ GAIN_22[4]	GAM_R_ GAIN_22[3]	GAM_R_ GAIN_22[2]	GAM_R_ GAIN_22[1]	GAM_R_ GAIN_22[0]
		—	—	—	—	—	GAM_R_ GAIN_23[10]	GAM_R_ GAIN_23[9]	GAM_R_ GAIN_23[8]
		GAM_R_ GAIN_23[7]	GAM_R_ GAIN_23[6]	GAM_R_ GAIN_23[5]	GAM_R_ GAIN_23[4]	GAM_R_ GAIN_23[3]	GAM_R_ GAIN_23[2]	GAM_R_ GAIN_23[1]	GAM_R_ GAIN_23[0]
	GAM_R_ LUT13	—	—	—	—	—	GAM_R_ GAIN_24[10]	GAM_R_ GAIN_24[9]	GAM_R_ GAIN_24[8]
		GAM_R_ GAIN_24[7]	GAM_R_ GAIN_24[6]	GAM_R_ GAIN_24[5]	GAM_R_ GAIN_24[4]	GAM_R_ GAIN_24[3]	GAM_R_ GAIN_24[2]	GAM_R_ GAIN_24[1]	GAM_R_ GAIN_24[0]
		—	—	—	—	—	GAM_R_ GAIN_25[10]	GAM_R_ GAIN_25[9]	GAM_R_ GAIN_25[8]
		GAM_R_ GAIN_25[7]	GAM_R_ GAIN_25[6]	GAM_R_ GAIN_25[5]	GAM_R_ GAIN_25[4]	GAM_R_ GAIN_25[3]	GAM_R_ GAIN_25[2]	GAM_R_ GAIN_25[1]	GAM_R_ GAIN_25[0]
	GAM_R_ LUT14	—	—	—	—	—	GAM_R_ GAIN_26[10]	GAM_R_ GAIN_26[9]	GAM_R_ GAIN_26[8]
		GAM_R_ GAIN_26[7]	GAM_R_ GAIN_26[6]	GAM_R_ GAIN_26[5]	GAM_R_ GAIN_26[4]	GAM_R_ GAIN_26[3]	GAM_R_ GAIN_26[2]	GAM_R_ GAIN_26[1]	GAM_R_ GAIN_26[0]
		—	—	—	—	—	GAM_R_ GAIN_27[10]	GAM_R_ GAIN_27[9]	GAM_R_ GAIN_27[8]
		GAM_R_ GAIN_27[7]	GAM_R_ GAIN_27[6]	GAM_R_ GAIN_27[5]	GAM_R_ GAIN_27[4]	GAM_R_ GAIN_27[3]	GAM_R_ GAIN_27[2]	GAM_R_ GAIN_27[1]	GAM_R_ GAIN_27[0]
	GAM_R_ LUT15	—	—	—	—	—	GAM_R_ GAIN_28[10]	GAM_R_ GAIN_28[9]	GAM_R_ GAIN_28[8]
		GAM_R_ GAIN_28[7]	GAM_R_ GAIN_28[6]	GAM_R_ GAIN_28[5]	GAM_R_ GAIN_28[4]	GAM_R_ GAIN_28[3]	GAM_R_ GAIN_28[2]	GAM_R_ GAIN_28[1]	GAM_R_ GAIN_28[0]
		—	—	—	—	—	GAM_R_ GAIN_29[10]	GAM_R_ GAIN_29[9]	GAM_R_ GAIN_29[8]
		GAM_R_ GAIN_29[7]	GAM_R_ GAIN_29[6]	GAM_R_ GAIN_29[5]	GAM_R_ GAIN_29[4]	GAM_R_ GAIN_29[3]	GAM_R_ GAIN_29[2]	GAM_R_ GAIN_29[1]	GAM_R_ GAIN_29[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	GAM_R_ LUT16	—	—	—	—	—	GAM_R_ GAIN_30[10]	GAM_R_ GAIN_30[9]	GAM_R_ GAIN_30[8]
		GAM_R_ GAIN_30[7]	GAM_R_ GAIN_30[6]	GAM_R_ GAIN_30[5]	GAM_R_ GAIN_30[4]	GAM_R_ GAIN_30[3]	GAM_R_ GAIN_30[2]	GAM_R_ GAIN_30[1]	GAM_R_ GAIN_30[0]
		—	—	—	—	—	GAM_R_ GAIN_31[10]	GAM_R_ GAIN_31[9]	GAM_R_ GAIN_31[8]
		GAM_R_ GAIN_31[7]	GAM_R_ GAIN_31[6]	GAM_R_ GAIN_31[5]	GAM_R_ GAIN_31[4]	GAM_R_ GAIN_31[3]	GAM_R_ GAIN_31[2]	GAM_R_ GAIN_31[1]	GAM_R_ GAIN_31[0]
	GAM_R_ AREA1	—	—	—	—	—	—	—	—
		GAM_R_TH _01[7]	GAM_R_TH _01[6]	GAM_R_TH _01[5]	GAM_R_TH _01[4]	GAM_R_TH _01[3]	GAM_R_TH _01[2]	GAM_R_TH _01[1]	GAM_R_TH _01[0]
		GAM_R_TH _02[7]	GAM_R_TH _02[6]	GAM_R_TH _02[5]	GAM_R_TH _02[4]	GAM_R_TH _02[3]	GAM_R_TH _02[2]	GAM_R_TH _02[1]	GAM_R_TH _02[0]
		GAM_R_TH _03[7]	GAM_R_TH _03[6]	GAM_R_TH _03[5]	GAM_R_TH _03[4]	GAM_R_TH _03[3]	GAM_R_TH _03[2]	GAM_R_TH _03[1]	GAM_R_TH _03[0]
	GAM_R_ AREA2	GAM_R_TH _04[7]	GAM_R_TH _04[6]	GAM_R_TH _04[5]	GAM_R_TH _04[4]	GAM_R_TH _04[3]	GAM_R_TH _04[2]	GAM_R_TH _04[1]	GAM_R_TH _04[0]
		GAM_R_TH _05[7]	GAM_R_TH _05[6]	GAM_R_TH _05[5]	GAM_R_TH _05[4]	GAM_R_TH _05[3]	GAM_R_TH _05[2]	GAM_R_TH _05[1]	GAM_R_TH _05[0]
		GAM_R_TH _06[7]	GAM_R_TH _06[6]	GAM_R_TH _06[5]	GAM_R_TH _06[4]	GAM_R_TH _06[3]	GAM_R_TH _06[2]	GAM_R_TH _06[1]	GAM_R_TH _06[0]
		GAM_R_TH _07[7]	GAM_R_TH _07[6]	GAM_R_TH _07[5]	GAM_R_TH _07[4]	GAM_R_TH _07[3]	GAM_R_TH _07[2]	GAM_R_TH _07[1]	GAM_R_TH _07[0]
	GAM_R_ AREA3	GAM_R_TH _08[7]	GAM_R_TH _08[6]	GAM_R_TH _08[5]	GAM_R_TH _08[4]	GAM_R_TH _08[3]	GAM_R_TH _08[2]	GAM_R_TH _08[1]	GAM_R_TH _08[0]
		GAM_R_TH _09[7]	GAM_R_TH _09[6]	GAM_R_TH _09[5]	GAM_R_TH _09[4]	GAM_R_TH _09[3]	GAM_R_TH _09[2]	GAM_R_TH _09[1]	GAM_R_TH _09[0]
		GAM_R_TH _10[7]	GAM_R_TH _10[6]	GAM_R_TH _10[5]	GAM_R_TH _10[4]	GAM_R_TH _10[3]	GAM_R_TH _10[2]	GAM_R_TH _10[1]	GAM_R_TH _10[0]
		GAM_R_TH _11[7]	GAM_R_TH _11[6]	GAM_R_TH _11[5]	GAM_R_TH _11[4]	GAM_R_TH _11[3]	GAM_R_TH _11[2]	GAM_R_TH _11[1]	GAM_R_TH _11[0]
	GAM_R_ AREA4	GAM_R_TH _12[7]	GAM_R_TH _12[6]	GAM_R_TH _12[5]	GAM_R_TH _12[4]	GAM_R_TH _12[3]	GAM_R_TH _12[2]	GAM_R_TH _12[1]	GAM_R_TH _12[0]
		GAM_R_TH _13[7]	GAM_R_TH _13[6]	GAM_R_TH _13[5]	GAM_R_TH _13[4]	GAM_R_TH _13[3]	GAM_R_TH _13[2]	GAM_R_TH _13[1]	GAM_R_TH _13[0]
		GAM_R_TH _14[7]	GAM_R_TH _14[6]	GAM_R_TH _14[5]	GAM_R_TH _14[4]	GAM_R_TH _14[3]	GAM_R_TH _14[2]	GAM_R_TH _14[1]	GAM_R_TH _14[0]
		GAM_R_TH _15[7]	GAM_R_TH _15[6]	GAM_R_TH _15[5]	GAM_R_TH _15[4]	GAM_R_TH _15[3]	GAM_R_TH _15[2]	GAM_R_TH _15[1]	GAM_R_TH _15[0]
	GAM_R_ AREA5	GAM_R_TH _16[7]	GAM_R_TH _16[6]	GAM_R_TH _16[5]	GAM_R_TH _16[4]	GAM_R_TH _16[3]	GAM_R_TH _16[2]	GAM_R_TH _16[1]	GAM_R_TH _16[0]
		GAM_R_TH _17[7]	GAM_R_TH _17[6]	GAM_R_TH _17[5]	GAM_R_TH _17[4]	GAM_R_TH _17[3]	GAM_R_TH _17[2]	GAM_R_TH _17[1]	GAM_R_TH _17[0]
		GAM_R_TH _18[7]	GAM_R_TH _18[6]	GAM_R_TH _18[5]	GAM_R_TH _18[4]	GAM_R_TH _18[3]	GAM_R_TH _18[2]	GAM_R_TH _18[1]	GAM_R_TH _18[0]
		GAM_R_TH _19[7]	GAM_R_TH _19[6]	GAM_R_TH _19[5]	GAM_R_TH _19[4]	GAM_R_TH _19[3]	GAM_R_TH _19[2]	GAM_R_TH _19[1]	GAM_R_TH _19[0]
	GAM_R_ AREA6	GAM_R_TH _20[7]	GAM_R_TH _20[6]	GAM_R_TH _20[5]	GAM_R_TH _20[4]	GAM_R_TH _20[3]	GAM_R_TH _20[2]	GAM_R_TH _20[1]	GAM_R_TH _20[0]
		GAM_R_TH _21[7]	GAM_R_TH _21[6]	GAM_R_TH _21[5]	GAM_R_TH _21[4]	GAM_R_TH _21[3]	GAM_R_TH _21[2]	GAM_R_TH _21[1]	GAM_R_TH _21[0]
		GAM_R_TH _22[7]	GAM_R_TH _22[6]	GAM_R_TH _22[5]	GAM_R_TH _22[4]	GAM_R_TH _22[3]	GAM_R_TH _22[2]	GAM_R_TH _22[1]	GAM_R_TH _22[0]
		GAM_R_TH _23[7]	GAM_R_TH _23[6]	GAM_R_TH _23[5]	GAM_R_TH _23[4]	GAM_R_TH _23[3]	GAM_R_TH _23[2]	GAM_R_TH _23[1]	GAM_R_TH _23[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	GAM_R_ AREA7	GAM_R_TH _24[7]	GAM_R_TH _24[6]	GAM_R_TH _24[5]	GAM_R_TH _24[4]	GAM_R_TH _24[3]	GAM_R_TH _24[2]	GAM_R_TH _24[1]	GAM_R_TH _24[0]
		GAM_R_TH _25[7]	GAM_R_TH _25[6]	GAM_R_TH _25[5]	GAM_R_TH _25[4]	GAM_R_TH _25[3]	GAM_R_TH _25[2]	GAM_R_TH _25[1]	GAM_R_TH _25[0]
		GAM_R_TH _26[7]	GAM_R_TH _26[6]	GAM_R_TH _26[5]	GAM_R_TH _26[4]	GAM_R_TH _26[3]	GAM_R_TH _26[2]	GAM_R_TH _26[1]	GAM_R_TH _26[0]
		GAM_R_TH _27[7]	GAM_R_TH _27[6]	GAM_R_TH _27[5]	GAM_R_TH _27[4]	GAM_R_TH _27[3]	GAM_R_TH _27[2]	GAM_R_TH _27[1]	GAM_R_TH _27[0]
	GAM_R_ AREA8	GAM_R_TH _28[7]	GAM_R_TH _28[6]	GAM_R_TH _28[5]	GAM_R_TH _28[4]	GAM_R_TH _28[3]	GAM_R_TH _28[2]	GAM_R_TH _28[1]	GAM_R_TH _28[0]
		GAM_R_TH _29[7]	GAM_R_TH _29[6]	GAM_R_TH _29[5]	GAM_R_TH _29[4]	GAM_R_TH _29[3]	GAM_R_TH _29[2]	GAM_R_TH _29[1]	GAM_R_TH _29[0]
		GAM_R_TH _30[7]	GAM_R_TH _30[6]	GAM_R_TH _30[5]	GAM_R_TH _30[4]	GAM_R_TH _30[3]	GAM_R_TH _30[2]	GAM_R_TH _30[1]	GAM_R_TH _30[0]
		GAM_R_TH _31[7]	GAM_R_TH _31[6]	GAM_R_TH _31[5]	GAM_R_TH _31[4]	GAM_R_TH _31[3]	GAM_R_TH _31[2]	GAM_R_TH _31[1]	GAM_R_TH _31[0]
	TCON_ UPDATE	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	TCON_VEN
	TCON_TIM	—	—	—	—	—	TCON_ HALF[10]	TCON_ HALF[9]	TCON_ HALF[8]
		TCON_ HALF[7]	TCON_ HALF[6]	TCON_ HALF[5]	TCON_ HALF[4]	TCON_ HALF[3]	TCON_ HALF[2]	TCON_ HALF[1]	TCON_ HALF[0]
		—	—	—	—	—	TCON_OFF SET[10]	TCON_OFF SET[9]	TCON_OFF SET[8]
		TCON_ OFFSET[7]	TCON_ OFFSET[6]	TCON_ OFFSET[5]	TCON_ OFFSET[4]	TCON_ OFFSET[3]	TCON_ OFFSET[2]	TCON_ OFFSET[1]	TCON_ OFFSET[0]
	TCON_TIM _STVA1	—	—	—	—	—	TCON_STV A_VS[10]	TCON_STV A_VS[9]	TCON_STV A_VS[8]
		TCON_STV A_VS[7]	TCON_STV A_VS[6]	TCON_STV A_VS[5]	TCON_STV A_VS[4]	TCON_STV A_VS[3]	TCON_STV A_VS[2]	TCON_STV A_VS[1]	TCON_STV A_VS[0]
		—	—	—	—	—	TCON_STV A_VW[10]	TCON_STV A_VW[9]	TCON_STV A_VW[8]
		TCON_STV A_VW[7]	TCON_STV A_VW[6]	TCON_STV A_VW[5]	TCON_STV A_VW[4]	TCON_STV A_VW[3]	TCON_STV A_VW[2]	TCON_STV A_VW[1]	TCON_STV A_VW[0]
	TCON_TIM _STVA2	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	TCON_STV A_INV	—	TCON_STV A_SEL[2]	TCON_STV A_SEL[1]	TCON_STV A_SEL[0]
	TCON_TIM _STVB1	—	—	—	—	—	TCON_STV B_VS[10]	TCON_STV B_VS[9]	TCON_STV B_VS[8]
		TCON_STV B_VS[7]	TCON_STV B_VS[6]	TCON_STV B_VS[5]	TCON_STV B_VS[4]	TCON_STV B_VS[3]	TCON_STV B_VS[2]	TCON_STV B_VS[1]	TCON_STV B_VS[0]
		—	—	—	—	—	TCON_STV B_VW[10]	TCON_STV B_VW[9]	TCON_STV B_VW[8]
		TCON_STV B_VW[7]	TCON_STV B_VW[6]	TCON_STV B_VW[5]	TCON_STV B_VW[4]	TCON_STV B_VW[3]	TCON_STV B_VW[2]	TCON_STV B_VW[1]	TCON_STV B_VW[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	TCON_TIM _STVB2	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	TCON_STVB _INV	—	TCON_STVB _SEL[2]	TCON_STVB _SEL[1]	TCON_STVB _SEL[0]
	TCON_TIM _STH1	—	—	—	—	—	TCON_STH _HS[10]	TCON_STH _HS[9]	TCON_STH _HS[8]
		TCON_STH _HS[7]	TCON_STH _HS[6]	TCON_STH _HS[5]	TCON_STH _HS[4]	TCON_STH _HS[3]	TCON_STH _HS[2]	TCON_STH _HS[1]	TCON_STH _HS[0]
		—	—	—	—	—	TCON_STH _HW[10]	TCON_STH _HW[9]	TCON_STH _HW[8]
		TCON_STH _HW[7]	TCON_STH _HW[6]	TCON_STH _HW[5]	TCON_STH _HW[4]	TCON_STH _HW[3]	TCON_STH _HW[2]	TCON_STH _HW[1]	TCON_STH _HW[0]
	TCON_TIM _STH2	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	TCON_STH _HS_SEL
		—	—	—	TCON_STH _INV	—	TCON_STH _SEL[2]	TCON_STH _SEL[1]	TCON_STH _SEL[0]
	TCON_TIM _STB1	—	—	—	—	—	TCON_STB _HS[10]	TCON_STB _HS[9]	TCON_STB _HS[8]
		TCON_STB _HS[7]	TCON_STB _HS[6]	TCON_STB _HS[5]	TCON_STB _HS[4]	TCON_STB _HS[3]	TCON_STB _HS[2]	TCON_STB _HS[1]	TCON_STB _HS[0]
		—	—	—	—	—	TCON_STB _HW[10]	TCON_STB _HW[9]	TCON_STB _HW[8]
		TCON_STB _HW[7]	TCON_STB _HW[6]	TCON_STB _HW[5]	TCON_STB _HW[4]	TCON_STB _HW[3]	TCON_STB _HW[2]	TCON_STB _HW[1]	TCON_STB _HW[0]
	TCON_TIM _STB2	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	TCON_STB _HS_SEL
		—	—	—	TCON_STB _INV	—	TCON_STB _SEL[2]	TCON_STB _SEL[1]	TCON_STB _SEL[0]
	TCON_TIM _CPV1	—	—	—	—	—	TCON_CPV _HS[10]	TCON_CPV _HS[9]	TCON_CPV _HS[8]
		TCON_CPV _HS[7]	TCON_CPV _HS[6]	TCON_CPV _HS[5]	TCON_CPV _HS[4]	TCON_CPV _HS[3]	TCON_CPV _HS[2]	TCON_CPV _HS[1]	TCON_CPV _HS[0]
		—	—	—	—	—	TCON_CPV _HW[10]	TCON_CPV _HW[9]	TCON_CPV _HW[8]
		TCON_CPV _HW[7]	TCON_CPV _HW[6]	TCON_CPV _HW[5]	TCON_CPV _HW[4]	TCON_CPV _HW[3]	TCON_CPV _HW[2]	TCON_CPV _HW[1]	TCON_CPV _HW[0]
	TCON_TIM _CPV2	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	TCON_CPV _HS_SEL
		—	—	—	TCON_CPV _INV	—	TCON_CPV _SEL[2]	TCON_CPV _SEL[1]	TCON_CPV _SEL[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
视频显示 控制器 4	TCON_TIM _POLA1	—	—	—	—	—	TCON_POL A_HS[10]	TCON_POL A_HS[9]	TCON_POL A_HS[8]	
		TCON_POL A_HS[7]	TCON_POL A_HS[6]	TCON_POL A_HS[5]	TCON_POL A_HS[4]	TCON_POL A_HS[3]	TCON_POL A_HS[2]	TCON_POL A_HS[1]	TCON_POL A_HS[0]	
		—	—	—	—	—	TCON_POL A_HW[10]	TCON_POL A_HW[9]	TCON_POL A_HW[8]	
		TCON_POL A_HW[7]	TCON_POL A_HW[6]	TCON_POL A_HW[5]	TCON_POL A_HW[4]	TCON_POL A_HW[3]	TCON_POL A_HW[2]	TCON_POL A_HW[1]	TCON_POL A_HW[0]	
	TCON_TIM _POLA2	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	TCON_POL A_MD[1]	TCON_POL A_MD[0]	—	—	—	TCON_POL A_HS_SEL	
		—	—	—	TCON_POL A_INV	—	TCON_POL A_SEL[2]	TCON_POL A_SEL[1]	TCON_POL A_SEL[0]	
	TCON_TIM _POLB1	—	—	—	—	—	—	TCON_POL B_HS[10]	TCON_POL B_HS[9]	TCON_POL B_HS[8]
		TCON_POL B_HS[7]	TCON_POL B_HS[6]	TCON_POL B_HS[5]	TCON_POL B_HS[4]	TCON_POL B_HS[3]	TCON_POL B_HS[2]	TCON_POL B_HS[1]	TCON_POL B_HS[0]	
		—	—	—	—	—	TCON_POL B_HW[10]	TCON_POL B_HW[9]	TCON_POL B_HW[8]	
		TCON_POL B_HW[7]	TCON_POL B_HW[6]	TCON_POL B_HW[5]	TCON_POL B_HW[4]	TCON_POL B_HW[3]	TCON_POL B_HW[2]	TCON_POL B_HW[1]	TCON_POL B_HW[0]	
	TCON_TIM _POLB2	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	TCON_POL B_MD[1]	TCON_POL B_MD[0]	—	—	—	TCON_POL B_HS_SEL	
		—	—	—	TCON_POL B_INV	—	TCON_POL B_SEL[2]	TCON_POL B_SEL[1]	TCON_POL B_SEL[0]	
	TCON_TIM _DE	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	TCON_DE_ INV	
	OUT_ UPDATE	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	OUTCNT_ VEN	
	OUT_SET	—	—	—	OUT_ ENDIAN_ON	—	—	—	OUT_ SWAP_ON	
		—	—	—	—	—	—	—	—	
		—	—	OUT_ FORMAT[1]	OUT_ FORMAT[0]	—	—	OUT_FRQ_ SEL[1]	OUT_FRQ_ SEL[0]	
		—	—	—	OUT_DIR_ SEL	—	—	OUT_ PHASE[1]	OUT_ PHASE[0]	
	OUT_ BRIGHT1	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	PBRT_G[9]	PBRT_G[8]	
		PBRT_G[7]	PBRT_G[6]	PBRT_G[5]	PBRT_G[4]	PBRT_G[3]	PBRT_G[2]	PBRT_G[1]	PBRT_G[0]	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
视频显示 控制器 4	OUT_ BRIGHT2	—	—	—	—	—	—	PBRT_B[9]	PBRT_B[8]
		PBRT_B[7]	PBRT_B[6]	PBRT_B[5]	PBRT_B[4]	PBRT_B[3]	PBRT_B[2]	PBRT_B[1]	PBRT_B[0]
		—	—	—	—	—	—	PBRT_R[9]	PBRT_R[8]
		PBRT_R[7]	PBRT_R[6]	PBRT_R[5]	PBRT_R[4]	PBRT_R[3]	PBRT_R[2]	PBRT_R[1]	PBRT_R[0]
	OUT_ CONTRAST	—	—	—	—	—	—	—	—
		CONT_G[7]	CONT_G[6]	CONT_G[5]	CONT_G[4]	CONT_G[3]	CONT_G[2]	CONT_G[1]	CONT_G[0]
		CONT_B[7]	CONT_B[6]	CONT_B[5]	CONT_B[4]	CONT_B[3]	CONT_B[2]	CONT_B[1]	CONT_B[0]
	OUT_ PDTHA	CONT_R[7]	CONT_R[6]	CONT_R[5]	CONT_R[4]	CONT_R[3]	CONT_R[2]	CONT_R[1]	CONT_R[0]
		—	—	—	—	—	—	—	—
		—	—	PDTH_SEL[1]	PDTH_SEL[0]	—	—	PDTH_ FORMAT[1]	PDTH_ FORMAT[0]
		—	—	PDTH_PA[1]	PDTH_PA[0]	—	—	PDTH_PB[1]	PDTH_PB[0]
	OUT_CLK_ PHASE	—	—	PDTH_PC[1]	PDTH_PC[0]	—	—	PDTH_PD[1]	PDTH_PD[0]
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
	SYSCNT_ INT1	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	INT_STA8
	SYSCNT_ INT2	—	—	—	INT_STA7	—	—	—	INT_STA6
		—	—	—	INT_STA5	—	—	—	INT_STA4
		—	—	—	INT_STA3	—	—	—	INT_STA2
		—	—	—	INT_STA1	—	—	—	INT_STA0
	SYSCNT_ INT3	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	INT_OUT8_ ON
	SYSCNT_ INT4	—	—	—	INT_OUT7_ ON	—	—	—	INT_OUT6_ ON
		—	—	—	INT_OUT5_ ON	—	—	—	INT_OUT4_ ON
		—	—	—	INT_OUT3_ ON	—	—	—	INT_OUT2_ ON
		—	—	—	INT_OUT1_ ON	—	—	—	INT_OUT0_ ON
	SYSCNT_ PANEL_ CLK	—	—	PANEL_ ICKSEL[1]	PANEL_ ICKSEL[0]	—	—	—	PANEL_ ICKEN
		—	—	PANEL_ DCDR[5]	PANEL_ DCDR[4]	PANEL_ DCDR[3]	PANEL_ DCDR[2]	PANEL_ DCDR[1]	PANEL_ DCDR[0]
	SYSCNT_ CLUT	—	—	—	—	—	—	—	GR3_CLT_ SEL_ST
		—	—	—	GR2_CLT_ SEL_ST	—	—	—	GR1_CLT_ SEL_ST

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
失真校正 引擎	CR	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		SWRST	—	—	—	—	—	—	—	—
		—	—	RESUME	STOP	—	—	ARS	RS	
	SR	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	DSA	—	STP	—	INT	IER	TRA	
	SRCR	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	STPCLR	—	INTCLR	IERCLR	TRACLR	
	ICR	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	STPENB	—	INTENB	IERENB	TRAENB	
	IMR	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	STM	—	INM	IEM	TRAM	
	DLPR	DLP[31]	DLP[30]	DLP[29]	DLP[28]	DLP[27]	DLP[26]	DLP[25]	DLP[24]	
		DLP[23]	DLP[22]	DLP[21]	DLP[20]	DLP[19]	DLP[18]	DLP[17]	DLP[16]	
		DLP[15]	DLP[14]	DLP[13]	DLP[12]	DLP[11]	DLP[10]	DLP[9]	DLP[8]	
		DLP[7]	DLP[6]	DLP[5]	DLP[4]	DLP[3]	DLP[2]	DLP[1]	DLP[0]	
	DLSAR	DLSA[31]	DLSA[30]	DLSA[29]	DLSA[28]	DLSA[27]	DLSA[26]	DLSA[25]	DLSA[24]	
		DLSA[23]	DLSA[22]	DLSA[21]	DLSA[20]	DLSA[19]	DLSA[18]	DLSA[17]	DLSA[16]	
		DLSA[15]	DLSA[14]	DLSA[13]	DLSA[12]	DLSA[11]	DLSA[10]	DLSA[9]	DLSA[8]	
		DLSA[7]	DLSA[6]	DLSA[5]	DLSA[4]	DLSA[3]	DLSA[2]	DLSA[1]	DLSA[0]	
	DSAR	DSAR[31]	DSAR[30]	DSAR[29]	DSAR[28]	DSAR[27]	DSAR[26]	DSAR[25]	DSAR[24]	
		DSAR[23]	DSAR[22]	DSAR[21]	DSAR[20]	DSAR[19]	DSAR[18]	DSAR[17]	DSAR[16]	
		DSAR[15]	DSAR[14]	DSAR[13]	DSAR[12]	DSAR[11]	DSAR[10]	DSAR[9]	DSAR[8]	
		DSAR[7]	DSAR[6]	DSAR[5]	DSAR[4]	DSAR[3]	DSAR[2]	DSAR[1]	DSAR[0]	
	DSTR	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	DSTR[12]	DSTR[11]	DSTR[10]	DSTR[9]	DSTR[8]	
		DSTR[7]	DSTR[6]	DSTR[5]	DSTR[4]	DSTR[3]	DSTR[2]	DSTR[1]	DSTR[0]	
	DSAR2	DSAR2[31]	DSAR2[30]	DSAR2[29]	DSAR2[28]	DSAR2[27]	DSAR2[26]	DSAR2[25]	DSAR2[24]	
		DSAR2[23]	DSAR2[22]	DSAR2[21]	DSAR2[20]	DSAR2[19]	DSAR2[18]	DSAR2[17]	DSAR2[16]	
		DSAR2[15]	DSAR2[14]	DSAR2[13]	DSAR2[12]	DSAR2[11]	DSAR2[10]	DSAR2[9]	DSAR2[8]	
		DSAR2[7]	DSAR2[6]	DSAR2[5]	DSAR2[4]	DSAR2[3]	DSAR2[2]	DSAR2[1]	DSAR2[0]	
	TRIMR	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	TCM	DUDVM	DXDYM	AUTOSG	AUTODG	BFE	TME	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
失真校正 引擎	TRIMSR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	TCMS	DUDVMS	DXDYMS	AUTOSGS	AUTODGS	BFES	TMES
	TRIMCR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	TCMC	DUDVMC	DXDYMC	AUTOSGC	AUTODGC	BFEC	TMEC
	TRICR	—	—	—	—	—	—	—	—
		TCV[7]	TCV[6]	TCV[5]	TCV[4]	TCV[3]	TCV[2]	TCV[1]	TCV[0]
		TCU[7]	TCU[6]	TCU[5]	TCU[4]	TCU[3]	TCU[2]	TCU[1]	TCU[0]
		TCY[7]	TCY[6]	TCY[5]	TCY[4]	TCY[3]	TCY[2]	TCY[1]	TCY[0]
	UVDPOR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	DDP
		—	—	—	—	—	UVDPO[2]	UVDPO[1]	UVDPO[0]
	SUSR	—	—	—	—	—	—	SUW[9]	SUW[8]
		SUW[7]	SUW[6]	SUW[5]	SUW[4]	SUW[3]	SUW[2]	SUW[1]	SUW[0]
		—	—	—	—	—	—	SVW[9]	SVW[8]
		SVW[7]	SVW[6]	SVW[5]	SVW[4]	SVW[3]	SVW[2]	SVW[1]	SVW[0]
	SVSR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	SVSR[9]	SVSR[8]
		SVSR[7]	SVSR[6]	SVSR[5]	SVSR[4]	SVSR[3]	SVSR[2]	SVSR[1]	SVSR[0]
	XMINR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	XMIN[11]	XMIN[10]	XMIN[9]	XMIN[8]
		XMIN[7]	XMIN[6]	XMIN[5]	XMIN[4]	XMIN[3]	XMIN[2]	XMIN[1]	XMIN[0]
	YMINR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	YMIN[11]	YMIN[10]	YMIN[9]	YMIN[8]
		YMIN[7]	YMIN[6]	YMIN[5]	YMIN[4]	YMIN[3]	YMIN[2]	YMIN[1]	YMIN[0]
	XMAXR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	XMAX[11]	XMAX[10]	XMAX[9]	XMAX[8]
		XMAX[7]	XMAX[6]	XMAX[5]	XMAX[4]	XMAX[3]	XMAX[2]	XMAX[1]	XMAX[0]
	YMAXR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	YMAX[11]	YMAX[10]	YMAX[9]	YMAX[8]
		YMAX[7]	YMAX[6]	YMAX[5]	YMAX[4]	YMAX[3]	YMAX[2]	YMAX[1]	YMAX[0]
	AMXSR	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	AMXS[9]	AMXS[8]
		AMXS[7]	AMXS[6]	AMXS[5]	AMXS[4]	AMXS[3]	AMXS[2]	AMXS[1]	AMXS[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0		
失真校正 引擎	AMYSR	—	—	—	—	—	—	—	—		
		—	—	—	—	—	—	—	—		
		—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	
	AMXOR	—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	
	AMYOR	—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	
	MACR1	—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	
		—	—	—	—	EMAM	—	—	LWSWAP	—	
		—	—	—	—	—	—	—	—	—	
	LSPR	—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	
	LEPR	—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	
	LMSR	—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	
	显示输出 比较单元	DOCMCR	—	—	—	—	—	—	—	—	
			—	—	—	—	—	—	—	—	
			—	—	—	—	—	—	—	—	
			—	—	—	—	—	—	—	—	
		DOCMSTR	—	—	—	—	—	—	—	—	—
			—	—	—	—	—	—	—	—	—
			—	—	—	—	—	—	—	—	—
			—	—	—	—	—	—	—	—	—
		DOCMCLSTR	—	—	—	—	—	—	—	—	—
			—	—	—	—	—	—	—	—	—
			—	—	—	—	—	—	—	—	—
			—	—	—	—	—	—	—	—	—
DOCMENR		—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	—	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
显示输出 比较单元	DOCMPMR	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	CMPBT	
		CMPDFA[7]	CMPDFA[6]	CMPDFA[5]	CMPDFA[4]	CMPDFA[3]	CMPDFA[2]	CMPDFA[1]	CMPDFA[0]	
		CMPDAUF	—	—	—	CMPSELP[3]	CMPSELP[2]	CMPSELP[1]	CMPSELP[0]	
	DOCMECRCR	CMPECRC [31]	CMPECRC [30]	CMPECRC [29]	CMPECRC [28]	CMPECRC [27]	CMPECRC [26]	CMPECRC [25]	CMPECRC [24]	
		CMPECRC [23]	CMPECRC [22]	CMPECRC [21]	CMPECRC [20]	CMPECRC [19]	CMPECRC [18]	CMPECRC [17]	CMPECRC [16]	
		CMPECRC [15]	CMPECRC [14]	CMPECRC [13]	CMPECRC [12]	CMPECRC [11]	CMPECRC [10]	CMPECRC [9]	CMPECRC [8]	
		CMPECRC [7]	CMPECRC [6]	CMPECRC [5]	CMPECRC [4]	CMPECRC [3]	CMPECRC [2]	CMPECRC [1]	CMPECRC [0]	
	DOCMCCRCR	CMPCCRC [31]	CMPCCRC [30]	CMPCCRC [29]	CMPCCRC [28]	CMPCCRC [27]	CMPCCRC [26]	CMPCCRC [25]	CMPCCRC [24]	
		CMPCCRC [23]	CMPCCRC [22]	CMPCCRC [21]	CMPCCRC [20]	CMPCCRC [19]	CMPCCRC [18]	CMPCCRC [17]	CMPCCRC [16]	
		CMPCCRC [15]	CMPCCRC [14]	CMPCCRC [13]	CMPCCRC [12]	CMPCCRC [11]	CMPCCRC [10]	CMPCCRC [9]	CMPCCRC [8]	
		CMPCCRC [7]	CMPCCRC [6]	CMPCCRC [5]	CMPCCRC [4]	CMPCCRC [3]	CMPCCRC [2]	CMPCCRC [1]	CMPCCRC [0]	
	DOCMSPIXR	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	CMPSPX [10]	CMPSPX[9]	CMPSPX[8]
		CMPSPX[7]	CMPSPX[6]	CMPSPX[5]	CMPSPX[4]	CMPSPX[3]	CMPSPX[2]	CMPSPX[1]	CMPSPX[0]	
	DOCMSPIYR	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	CMPSPY [10]	CMPSPY[9]	CMPSPY[8]
		CMPSPY[7]	CMPSPY[6]	CMPSPY[5]	CMPSPY[4]	CMPSPY[3]	CMPSPY[2]	CMPSPY[1]	CMPSPY[0]	
	DOCMSZXR	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	CMPSZX[10]	CMPSZX[9]	CMPSZX[8]
		CMPSZX[7]	CMPSZX[6]	CMPSZX[5]	CMPSZX[4]	CMPSZX[3]	CMPSZX[2]	CMPSZX[1]	CMPSZX[0]	
	DOCMSZYR	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	CMPSZY[10]	CMPSZY[9]	CMPSZY[8]
		CMPSZY[7]	CMPSZY[6]	CMPSZY[5]	CMPSZY[4]	CMPSZY[3]	CMPSZY[2]	CMPSZY[1]	CMPSZY[0]	
	DOCMCIRCIR	CRCINI[31]	CRCINI[30]	CRCINI[29]	CRCINI[28]	CRCINI[27]	CRCINI[26]	CRCINI[25]	CRCINI[24]	
		CRCINI[23]	CRCINI[22]	CRCINI[21]	CRCINI[20]	CRCINI[19]	CRCINI[18]	CRCINI[17]	CRCINI[16]	
		CRCINI[15]	CRCINI[14]	CRCINI[13]	CRCINI[12]	CRCINI[11]	CRCINI[10]	CRCINI[9]	CRCINI[8]	
		CRCINI[7]	CRCINI[6]	CRCINI[5]	CRCINI[4]	CRCINI[3]	CRCINI[2]	CRCINI[1]	CRCINI[0]	
	JPEG 编解 码器单元	JCMOD	—	—	—	—	DSP	REDU[2]	REDU[1]	REDU[0]
		JCCMD	BRST	—	—	—	—	JEND	JRST	JSRT
		JCQTN	—	—	QT3[1]	QT3[0]	QT2[1]	QT2[0]	QT1[1]	QT1[0]
		JCHTN	—	—	HTA3	HTD3	HTA2	HTD2	HTA1	HTD1
		JCDRIU	DRIU[7]	DRIU[6]	DRIU[5]	DRIU[4]	DRIU[3]	DRIU[2]	DRIU[1]	DRIU[0]
		JCDRID	DRID[7]	DRID[6]	DRID[5]	DRID[4]	DRID[3]	DRID[2]	DRID[1]	DRID[0]
		JCVSZU	VSZU[7]	VSZU[6]	VSZU[5]	VSZU[4]	VSZU[3]	VSZU[2]	VSZU[1]	VSZU[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
JPEG 编解码器单元	JCVSZD	VSZD[7]	VSZD[6]	VSZD[5]	VSZD[4]	VSZD[3]	VSZD[2]	VSZD[1]	VSZD[0]	
	JCHSZU	HSZU[7]	HSZU[6]	HSZU[5]	HSZU[4]	HSZU[3]	HSZU[2]	HSZU[1]	HSZU[0]	
	JCHSZD	HSZD[7]	HSZD[6]	HSZD[5]	HSZD[4]	HSZD[3]	HSZD[2]	HSZD[1]	HSZD[0]	
	JCDTCU	DCU[7]	DCU[6]	DCU[5]	DCU[4]	DCU[3]	DCU[2]	DCU[1]	DCU[0]	
	JCDTCM	DCM[7]	DCM[6]	DCM[5]	DCM[4]	DCM[3]	DCM[2]	DCM[1]	DCM[0]	
	JCDTCD	DCD[7]	DCD[6]	DCD[5]	DCD[4]	DCD[3]	DCD[2]	DCD[1]	DCD[0]	
	JINTE0	INT7	INT6	INT5	—	INT3	—	—	—	
	JINTS0	—	INS6	INS5	—	INS3	—	—	—	
	JCDERR	—	—	—	—	ERR[3]	ERR[2]	ERR[1]	ERR[0]	
	JCRST	—	—	—	—	—	—	—	RST	
	JIFECNT	—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		—	JOUTRINI	JOUTRCMD	JOUTC	—	JOUTSWAP [2]	JOUTSWAP [1]	JOUTSWAP [0]	JOUTSWAP [0]
		—	DINRINI	DINRCMD	DINLC	—	DINSWAP[2]	DINSWAP[1]	DINSWAP[0]	DINSWAP[0]
	JIFESA	ESA[31]	ESA[30]	ESA[29]	ESA[28]	ESA[27]	ESA[26]	ESA[25]	ESA[24]	ESA[24]
		ESA[23]	ESA[22]	ESA[21]	ESA[20]	ESA[19]	ESA[18]	ESA[17]	ESA[16]	ESA[16]
		ESA[15]	ESA[14]	ESA[13]	ESA[12]	ESA[11]	ESA[10]	ESA[9]	ESA[8]	ESA[8]
		ESA[7]	ESA[6]	ESA[5]	ESA[4]	ESA[3]	ESA[2]	ESA[1]	ESA[0]	ESA[0]
	JIFESOFST	—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		—	ESMW[14]	ESMW[13]	ESMW[12]	ESMW[11]	ESMW[10]	ESMW[9]	ESMW[8]	ESMW[8]
		ESMW[7]	ESMW[6]	ESMW[5]	ESMW[4]	ESMW[3]	ESMW[2]	ESMW[1]	ESMW[0]	ESMW[0]
	JIFEDA	EDA[31]	EDA[30]	EDA[29]	EDA[28]	EDA[27]	EDA[26]	EDA[25]	EDA[24]	EDA[24]
		EDA[23]	EDA[22]	EDA[21]	EDA[20]	EDA[19]	EDA[18]	EDA[17]	EDA[16]	EDA[16]
		EDA[15]	EDA[14]	EDA[13]	EDA[12]	EDA[11]	EDA[10]	EDA[9]	EDA[8]	EDA[8]
		EDA[7]	EDA[6]	EDA[5]	EDA[4]	EDA[3]	EDA[2]	EDA[1]	EDA[0]	EDA[0]
	JIFESLC	—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		LINES[15]	LINES[14]	LINES[13]	LINES[12]	LINES[11]	LINES[10]	LINES[9]	LINES[8]	LINES[8]
		LINES[7]	LINES[6]	LINES[5]	LINES[4]	LINES[3]	LINES[2]	LINES[1]	LINES[0]	LINES[0]
	JIFEDDC	—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—
		JDATAS[15]	JDATAS[14]	JDATAS[13]	JDATAS[12]	JDATAS[11]	JDATAS[10]	JDATAS[9]	JDATAS[8]	JDATAS[8]
		JDATAS[7]	JDATAS[6]	JDATAS[5]	JDATAS[4]	JDATAS[3]	JDATAS[2]	JDATAS[1]	JDATAS[0]	JDATAS[0]
	JIFDCNT	—	—	VINTER[1]	VINTER[0]	HINTER[1]	HINTER[0]	OPF[1]	OPF[0]	OPF[0]
		—	—	—	—	—	—	—	—	—
		—	JINRINI	JINRCMD	JINC	—	JINSWAP[2]	JINSWAP[1]	JINSWAP[0]	JINSWAP[0]
		—	DOURINI	DOURCMD	DOUTC	—	DOURSWAP [2]	DOURSWAP [1]	DOURSWAP [0]	DOURSWAP [0]
	JIFDSA	DSA[31]	DSA[30]	DSA[29]	DSA[28]	DSA[27]	DSA[26]	DSA[25]	DSA[24]	DSA[24]
		DSA[23]	DSA[22]	DSA[21]	DSA[20]	DSA[19]	DSA[18]	DSA[17]	DSA[16]	DSA[16]
		DSA[15]	DSA[14]	DSA[13]	DSA[12]	DSA[11]	DSA[10]	DSA[9]	DSA[8]	DSA[8]
		DSA[7]	DSA[6]	DSA[5]	DSA[4]	DSA[3]	DSA[2]	DSA[1]	DSA[0]	DSA[0]

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
JPEG 编解码器单元	JIFDDOFST	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	DDMW[14]	DDMW[13]	DDMW[12]	DDMW[11]	DDMW[10]	DDMW[9]	DDMW[8]
		DDMW[7]	DDMW[6]	DDMW[5]	DDMW[4]	DDMW[3]	DDMW[2]	DDMW[1]	DDMW[0]
	JIFDDA	DDA[31]	DDA[30]	DDA[29]	DDA[28]	DDA[27]	DDA[26]	DDA[25]	DDA[24]
		DDA[23]	DDA[22]	DDA[21]	DDA[20]	DDA[19]	DDA[18]	DDA[17]	DDA[16]
		DDA[15]	DDA[14]	DDA[13]	DDA[12]	DDA[11]	DDA[10]	DDA[9]	DDA[8]
		DDA[7]	DDA[6]	DDA[5]	DDA[4]	DDA[3]	DDA[2]	DDA[1]	DDA[0]
	JIFDSDC	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		JDATAS[15]	JDATAS[14]	JDATAS[13]	JDATAS[12]	JDATAS[11]	JDATAS[10]	JDATAS[9]	JDATAS[8]
		JDATAS[7]	JDATAS[6]	JDATAS[5]	JDATAS[4]	JDATAS[3]	JDATAS[2]	JDATAS[1]	JDATAS[0]
	JIFDDL	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		LINES[15]	LINES[14]	LINES[13]	LINES[12]	LINES[11]	LINES[10]	LINES[9]	LINES[8]
		LINES[7]	LINES[6]	LINES[5]	LINES[4]	LINES[3]	LINES[2]	LINES[1]	LINES[0]
	JIFDADT	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		ALPHA[7]	ALPHA[6]	ALPHA[5]	ALPHA[4]	ALPHA[3]	ALPHA[2]	ALPHA[1]	ALPHA[0]
	JINTE1	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	CBTEN	DINLEN	JOUTEN	—	DBTEN	JINEN	DOUTLEN
	JINTS1	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	CBTF	DINLF	JOUTF	—	DBTF	JINF	DOUTLF
	采样率转换器	SRCID_0							
SRCOD_0									
SRCIDCTRL_0		—	—	—	—	—	—	IED	IEN
		—	—	—	—	—	—	IFTRG[1]	IFTRG[0]
SRCODCTRL_0		—	—	—	—	—	OCH	OED	OEN
		—	—	—	—	—	—	OFTRG[1]	OFTRG[0]
SRCCTRL_0		—	—	CEEN	SRCEN	UDEN	OVEN	FL	CL
		IFS[3]	IFS[2]	IFS[1]	IFS[0]	—	OFS[2]	OFS[1]	OFS[0]
SRCSTAT_0		OFDN[4]	OFDN[3]	OFDN[2]	OFDN[1]	OFDN[0]	IFDN[3]	IFDN[2]	IFDN[1]
		IFDN[0]	—	CEF	FLF	UDF	OVF	IINT	OINT

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
采样率 转换器	SRCID_1									
	SRCOD_1									
	SRCIDCTRL_1	—	—	—	—	—	—	—	IED	IEN
		—	—	—	—	—	—	—	IFTRG[1]	IFTRG[0]
	SRCODCTRL_1	—	—	—	—	—	—	OCH	OED	OEN
		—	—	—	—	—	—	—	OFTRG[1]	OFTRG[0]
	SRCCTRL_1	—	—	CEEN	SRCEN	UDEN	OVEN	FL	CL	
		IFS[3]	IFS[2]	IFS[1]	IFS[0]	—	OFS[2]	OFS[1]	OFS[0]	
	SRCSTAT_1	OFDN[4]	OFDN[3]	OFDN[2]	OFDN[1]	OFDN[0]	IFDN[3]	IFDN[2]	IFDN[1]	
		IFDN[0]	—	CEF	FLF	UDF	OVF	IINT	OINT	
	SRCID_2									
	SRCOD_2									
	SRCIDCTRL_2	—	—	—	—	—	—	—	IED	IEN
		—	—	—	—	—	—	—	IFTRG[1]	IFTRG[0]
	SRCODCTRL_2	—	—	—	—	—	—	OCH	OED	OEN
		—	—	—	—	—	—	—	OFTRG[1]	OFTRG[0]
	SRCCTRL_2	—	—	CEEN	SRCEN	UDEN	OVEN	FL	CL	
		IFS[3]	IFS[2]	IFS[1]	IFS[0]	—	OFS[2]	OFS[1]	OFS[0]	
	SRCSTAT_2	OFDN[4]	OFDN[3]	OFDN[2]	OFDN[1]	OFDN[0]	IFDN[3]	IFDN[2]	IFDN[1]	
		IFDN[0]	—	CEF	FLF	UDF	OVF	IINT	OINT	
音频发生器	SGCR1_0	SGST	STPM	—	SGCK[1]	SGCK[0]	DPF[2]	DPF[1]	DPF[0]	
	SGCSR_0	SGIE	SGDEF	—	—	—	—	—	—	
	SGCR2_0	SGEND	TCHG	—	—	—	—	—	—	
	SGLR_0	LD[7]	LD[6]	LD[5]	LD[4]	LD[3]	LD[2]	LD[1]	LD[0]	
	SGTFR_0	—	TONE[6]	TONE[5]	TONE[4]	TONE[3]	TONE[2]	TONE[1]	TONE[0]	
	SGSFR_0	SFS[7]	SFS[6]	SFS[5]	SFS[4]	SFS[3]	SFS[2]	SFS[1]	SFS[0]	
	SGCR1_1	SGST	STPM	—	SGCK[1]	SGCK[0]	DPF[2]	DPF[1]	DPF[0]	
	SGCSR_1	SGIE	SGDEF	—	—	—	—	—	—	
	SGCR2_1	SGEND	TCHG	—	—	—	—	—	—	
	SGLR_1	LD[7]	LD[6]	LD[5]	LD[4]	LD[3]	LD[2]	LD[1]	LD[0]	
	SGTFR_1	—	TONE[6]	TONE[5]	TONE[4]	TONE[3]	TONE[2]	TONE[1]	TONE[0]	
	SGSFR_1	SFS[7]	SFS[6]	SFS[5]	SFS[4]	SFS[3]	SFS[2]	SFS[1]	SFS[0]	
	SGCR1_2	SGST	STPM	—	SGCK[1]	SGCK[0]	DPF[2]	DPF[1]	DPF[0]	
	SGCSR_2	SGIE	SGDEF	—	—	—	—	—	—	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
音频发生器	SGCR2_2	SGEND	TCHG	—	—	—	—	—	—
	SGLR_2	LD[7]	LD[6]	LD[5]	LD[4]	LD[3]	LD[2]	LD[1]	LD[0]
	SGTFR_2	—	TONE[6]	TONE[5]	TONE[4]	TONE[3]	TONE[2]	TONE[1]	TONE[0]
	SGSFR_2	SFS[7]	SFS[6]	SFS[5]	SFS[4]	SFS[3]	SFS[2]	SFS[1]	SFS[0]
	SGCR1_3	SGST	STPM	—	SGCK[1]	SGCK[0]	DPF[2]	DPF[1]	DPF[0]
	SGCSR_3	SGIE	SGDEF	—	—	—	—	—	—
	SGCR2_3	SGEND	TCHG	—	—	—	—	—	—
	SGLR_3	LD[7]	LD[6]	LD[5]	LD[4]	LD[3]	LD[2]	LD[1]	LD[0]
	SGTFR_3	—	TONE[6]	TONE[5]	TONE[4]	TONE[3]	TONE[2]	TONE[1]	TONE[0]
SGSFR_3	SFS[7]	SFS[6]	SFS[5]	SFS[4]	SFS[3]	SFS[2]	SFS[1]	SFS[0]	
MMC 主机 接口	CE_CMD_ SET	—	—	CMD[5]	CMD[4]	CMD[3]	CMD[2]	CMD[1]	CMD[0]
		RTYP[1]	RTYP[0]	RBSY	—	WDAT	DWEN	CMLTE	CMD12EN
		RIDXC[1]	RIDXC[0]	RCRC7C[1]	RCRC7C[0]	—	CRC16C	—	CRCSTE
		TBIT	OPDM	—	—	SBIT	—	DATW[1]	DATW[0]
	CE_ARG	ARG[31]	ARG[30]	ARG[29]	ARG[28]	ARG[27]	ARG[26]	ARG[25]	ARG[24]
		ARG[23]	ARG[22]	ARG[21]	ARG[20]	ARG[19]	ARG[18]	ARG[17]	ARG[16]
		ARG[15]	ARG[14]	ARG[13]	ARG[12]	ARG[11]	ARG[10]	ARG[9]	ARG[8]
		ARG[7]	ARG[6]	ARG[5]	ARG[4]	ARG[3]	ARG[2]	ARG[1]	ARG[0]
	CE_ARG_ CMD12	C12ARG[31]	C12ARG[30]	C12ARG[29]	C12ARG[28]	C12ARG[27]	C12ARG[26]	C12ARG[25]	C12ARG[24]
		C12ARG[23]	C12ARG[22]	C12ARG[21]	C12ARG[20]	C12ARG[19]	C12ARG[18]	C12ARG[17]	C12ARG[16]
		C12ARG[15]	C12ARG[14]	C12ARG[13]	C12ARG[12]	C12ARG[11]	C12ARG[10]	C12ARG[9]	C12ARG[8]
		C12ARG[7]	C12ARG[6]	C12ARG[5]	C12ARG[4]	C12ARG[3]	C12ARG[2]	C12ARG[1]	C12ARG[0]
	CE_CMD_ CTRL	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	BREAK
	CE_BLOCK_ SET	BLKCNT[15]	BLKCNT[14]	BLKCNT[13]	BLKCNT[12]	BLKCNT[11]	BLKCNT[10]	BLKCNT[9]	BLKCNT[8]
		BLKCNT[7]	BLKCNT[6]	BLKCNT[5]	BLKCNT[4]	BLKCNT[3]	BLKCNT[2]	BLKCNT[1]	BLKCNT[0]
		BLKSIZ[15]	BLKSIZ[14]	BLKSIZ[13]	BLKSIZ[12]	BLKSIZ[11]	BLKSIZ[10]	BLKSIZ[9]	BLKSIZ[8]
		BLKSIZ[7]	BLKSIZ[6]	BLKSIZ[5]	BLKSIZ[4]	BLKSIZ[3]	BLKSIZ[2]	BLKSIZ[1]	BLKSIZ[0]
	CE_CLK_ CTRL	—	—	—	—	—	—	—	CLKEN
		—	—	—	—	CLKDIV[3]	CLKDIV[2]	CLKDIV[1]	CLKDIV[0]
		—	—	SRSPTO[1]	SRSPTO[0]	SRBSYTO[3]	SRBSYTO[2]	SRBSYTO[1]	SRBSYTO[0]
		SRWDTO[3]	SRWDTO[2]	SRWDTO[1]	SRWDTO[0]	—	—	—	—
	CE_BUF_ ACC	—	—	—	—	—	—	DMAWEN	DMAREN
		—	—	—	—	—	—	BUSW	ATYP
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
CE_RESP3	RSP[127]	RSP[126]	RSP[125]	RSP[124]	RSP[123]	RSP[122]	RSP[121]	RSP[120]	
	RSP[119]	RSP[118]	RSP[117]	RSP[116]	RSP[115]	RSP[114]	RSP[113]	RSP[112]	
	RSP[111]	RSP[110]	RSP[109]	RSP[108]	RSP[107]	RSP[106]	RSP[105]	RSP[104]	
	RSP[103]	RSP[102]	RSP[101]	RSP[100]	RSP[99]	RSP[98]	RSP[97]	RSP[96]	
CE_RESP2	RSP[95]	RSP[94]	RSP[93]	RSP[92]	RSP[91]	RSP[90]	RSP[89]	RSP[88]	
	RSP[87]	RSP[86]	RSP[85]	RSP[84]	RSP[83]	RSP[82]	RSP[81]	RSP[80]	
	RSP[79]	RSP[78]	RSP[77]	RSP[76]	RSP[75]	RSP[74]	RSP[73]	RSP[72]	
	RSP[71]	RSP[70]	RSP[69]	RSP[68]	RSP[67]	RSP[66]	RSP[65]	RSP[64]	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
MMC 主机 接口	CE_RESP1	RSP[63]	RSP[62]	RSP[61]	RSP[60]	RSP[59]	RSP[58]	RSP[57]	RSP[56]
		RSP[55]	RSP[54]	RSP[53]	RSP[52]	RSP[51]	RSP[50]	RSP[49]	RSP[48]
		RSP[47]	RSP[46]	RSP[45]	RSP[44]	RSP[43]	RSP[42]	RSP[41]	RSP[40]
		RSP[39]	RSP[38]	RSP[37]	RSP[36]	RSP[35]	RSP[34]	RSP[33]	RSP[32]
	CE_RESP0	RSP[31]	RSP[30]	RSP[29]	RSP[28]	RSP[27]	RSP[26]	RSP[25]	RSP[24]
		RSP[23]	RSP[22]	RSP[21]	RSP[20]	RSP[19]	RSP[18]	RSP[17]	RSP[16]
		RSP[15]	RSP[14]	RSP[13]	RSP[12]	RSP[11]	RSP[10]	RSP[9]	RSP[8]
		RSP[7]	RSP[6]	RSP[5]	RSP[4]	RSP[3]	RSP[2]	RSP[1]	RSP[0]
	CE_RESP_ CMD12	RSP12[31]	RSP12[30]	RSP12[29]	RSP12[28]	RSP12[27]	RSP12[26]	RSP12[25]	RSP12[24]
		RSP12[23]	RSP12[22]	RSP12[21]	RSP12[20]	RSP12[19]	RSP12[18]	RSP12[17]	RSP12[16]
		RSP12[15]	RSP12[14]	RSP12[13]	RSP12[12]	RSP12[11]	RSP12[10]	RSP12[9]	RSP12[8]
		RSP12[7]	RSP12[6]	RSP12[5]	RSP12[4]	RSP12[3]	RSP12[2]	RSP12[1]	RSP12[0]
	CE_DATA	DATA[31]	DATA[30]	DATA[29]	DATA[28]	DATA[27]	DATA[26]	DATA[25]	DATA[24]
		DATA[23]	DATA[22]	DATA[21]	DATA[20]	DATA[19]	DATA[18]	DATA[17]	DATA[16]
		DATA[15]	DATA[14]	DATA[13]	DATA[12]	DATA[11]	DATA[10]	DATA[9]	DATA[8]
		DATA[7]	DATA[6]	DATA[5]	DATA[4]	DATA[3]	DATA[2]	DATA[1]	DATA[0]
	CE_INT	—	—	—	—	—	CMD12DRE	CMD12RBE	CMD12CRE
		DTRANE	BUFRE	BUFVEN	BUFREN	—	—	RBSYE	CRSPE
		CMDVIO	BUFVIO	—	—	WDATERR	RDATERR	RIDXERR	RSPERR
		—	—	—	CRCSTO	WDATTO	RDATTO	RBSYTO	RSPTO
	CE_INT_EN	—	—	—	—	—	MCMD12DRE	MCMD12RBE	MCMD12CRE
		MDTRANE	MBUFRE	MBUFVEN	MBUFREN	—	—	MRBSYE	MCRSPE
		MCMDVIO	MBUFVIO	—	—	MWDATERR	MRDATERR	MRIDXERR	MRSPEERR
		—	—	—	MCRCSTO	MWDATTO	MRDATTO	MRBSYTO	MRSPTO
	CE_HOST_ STS1	CMDSEQ	CMDSIG	RSPIDX[5]	RSPIDX[4]	RSPIDX[3]	RSPIDX[2]	RSPIDX[1]	RSPIDX[0]
		DATSIG[7]	DATSIG[6]	DATSIG[5]	DATSIG[4]	DATSIG[3]	DATSIG[2]	DATSIG[1]	DATSIG[0]
		RCVBLK[15]	RCVBLK[14]	RCVBLK[13]	RCVBLK[12]	RCVBLK[11]	RCVBLK[10]	RCVBLK[9]	RCVBLK[8]
		RCVBLK[7]	RCVBLK[6]	RCVBLK[5]	RCVBLK[4]	RCVBLK[3]	RCVBLK[2]	RCVBLK[1]	RCVBLK[0]
	CE_HOST_ STS2	CRCSTE	CRC16E	AC12CRCE	RSPCRC7E	CRCSTEBE	RDATEBE	AC12REBE	RSPEBE
		AC12IDXE	RSPIDXE	—	—	—	CRCST[2]	CRCST[1]	CRCST[0]
		—	STRDATTO	DATBSYTO	CRCSTTO	AC12BSYTO	RSPBSYTO	AC12RSPTO	STRSPTO
		—	—	—	—	—	—	—	—
	CE_DMA_ MODE	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	DMASEL
	CE_ DETECT	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—
		—	CDSIG	CDRISE	CDFALL	—	—	—	—
		—	—	MCDRISE	MCDFALL	—	—	—	—
	CE_ADD_ MODE	—	—	—	—	—	—	—	—
		—	—	—	—	CLKMAIN	—	—	—
		—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
MMC 主机 接口	CE_ VERSION	SWRST	—	—	—	—	—	—	—	
		—	—	—	—	—	—	—	—	
		VERSION[15]	VERSION[14]	VERSION[13]	VERSION[12]	VERSION[11]	VERSION[10]	VERSION[9]	VERSION[8]	
		VERSION[7]	VERSION[6]	VERSION[5]	VERSION[4]	VERSION[3]	VERSION[2]	VERSION[1]	VERSION[0]	
马达控制 PWM 定时 器	PWCR_1	—	—	IE	CMF	CST	CKS2	CKS1	CKS0	
	PWPR_1	OPS1H	OPS1G	OPS1F	OPS1E	OPS1D	OPS1C	OPS1B	OPS1A	
		PWCYR_1	PWCY15	PWCY14	PWCY13	PWCY12	PWCY11	PWCY10	PWCY9	PWCY8
	PWCY7		PWCY6	PWCY5	PWCY4	PWCY3	PWCY2	PWCY1	PWCY0	
	PWBFR_1A	—	—	—	OTS	—	—	DT9	DT8	
		DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0	
	PWBFR_1C	—	—	—	OTS	—	—	DT9	DT8	
		DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0	
	PWBFR_1E	—	—	—	OTS	—	—	DT9	DT8	
		DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0	
	PWBFR_1G	—	—	—	OTS	—	—	DT9	DT8	
		DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0	
	PWCR_2	—	—	IE	CMF	CST	CKS2	CKS1	CKS0	
	PWPR_2	OPS2H	OPS2G	OPS2F	OPS2E	OPS2D	OPS2C	OPS2B	OPS2A	
		PWCYR_2	PWCY15	PWCY14	PWCY13	PWCY12	PWCY11	PWCY10	PWCY9	PWCY8
	PWCY7		PWCY6	PWCY5	PWCY4	PWCY3	PWCY2	PWCY1	PWCY0	
	PWBFR_2A	—	—	—	OTS	—	—	DT9	DT8	
		DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0	
	PWBFR_2C	—	—	—	OTS	—	—	DT9	DT8	
		DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0	
	PWBFR_2E	—	—	—	OTS	—	—	DT9	DT8	
		DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0	
	PWBFR_2G	—	—	—	OTS	—	—	DT9	DT8	
		DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0	
	PWBTCR	BTC2G	BTC2E	BTC2C	BTC2A	BTC1G	BTC1E	BTC1C	BTC1A	
		—	—	—	—	—	—	—	—	
	通用输入 / 输出端口	PAIOR0	—	—	—	—	—	—	—	PA1IOR
			—	—	—	—	—	—	—	PA0IOR
		PADR0	—	—	—	—	—	—	—	PA1DR
			—	—	—	—	—	—	—	PA0DR
		PAPR0	—	—	—	—	—	—	—	—
			—	—	—	—	—	—	PA1PR	PA0PR
PBCR5		—	—	—	—	—	PB22MD2	PB22MD1	PB22MD0	
		—	—	PB21MD1	PB21MD0	—	PB20MD2	PB20MD1	PB20MD0	
PBCR4		—	PB19MD2	PB19MD1	PB19MD0	—	PB18MD2	PB18MD1	PB18MD0	
		—	PB17MD2	PB17MD1	PB17MD0	—	PB16MD2	PB16MD1	PB16MD0	
PBCR3		—	PB15MD2	PB15MD1	PB15MD0	—	PB14MD2	PB14MD1	PB14MD0	
		—	PB13MD2	PB13MD1	PB13MD0	—	—	PB12MD1	PB12MD0	
PBCR2		—	—	PB11MD1	PB11MD0	—	—	PB10MD1	PB10MD0	
		—	—	PB9MD1	PB9MD0	—	—	PB8MD1	PB8MD0	
PBCR1		—	—	PB7MD1	PB7MD0	—	—	PB6MD1	PB6MD0	
		—	—	PB5MD1	PB5MD0	—	—	PB4MD1	PB4MD0	

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0	
通用输入 / 输出端口	PBCR0	—	—	PB3MD1	PB3MD0	—	—	PB2MD1	PB2MD0	
		—	—	PB1MD1	PB1MD0	—	—	—	—	
	PBIOR1	—	—	—	—	—	—	—	—	
		—	PB22IOR	PB21IOR	PB20IOR	PB19IOR	PB18IOR	PB17IOR	PB16IOR	
	PBIOR0	PB15IOR	PB14IOR	PB13IOR	PB12IOR	PB11IOR	PB10IOR	PB9IOR	PB8IOR	
		PB7IOR	PB6IOR	PB5IOR	PB4IOR	PB3IOR	PB2IOR	PB1IOR	—	
	PBDR1	—	—	—	—	—	—	—	—	
		—	PB22DR	PB21DR	PB20DR	PB19DR	PB18DR	PB17DR	PB16DR	
	PBDR0	PB15DR	PB14DR	PB13DR	PB12DR	PB11DR	PB10DR	PB9DR	PB8DR	
		PB7DR	PB6DR	PB5DR	PB4DR	PB3DR	PB2DR	PB1DR	—	
	PBPR1	—	—	—	—	—	—	—	—	
		—	PB22PR	PB21PR	PB20PR	PB19PR	PB18PR	PB17PR	PB16PR	
	PBPR0	PB15PR	PB14PR	PB13PR	PB12PR	PB11PR	PB10PR	PB9PR	PB8PR	
		PB7PR	PB6PR	PB5PR	PB4PR	PB3PR	PB2PR	PB1PR	—	
	PCCR2	—	—	—	—	—	—	—	—	
		—	—	—	—	—	—	PC8MD2	PC8MD1	PC8MD0
	PCCR1	—	PC7MD2	PC7MD1	PC7MD0	—	—	PC6MD2	PC6MD1	PC6MD0
		—	PC5MD2	PC5MD1	PC5MD0	—	—	PC4MD1	PC4MD0	
	PCCR0	—	—	PC3MD1	PC3MD0	—	—	PC2MD1	PC2MD0	
		—	—	—	PC1MD0	—	—	—	PC0MD0	
	PCIOR0	—	—	—	—	—	—	—	PC8IOR	
		PC7IOR	PC6IOR	PC5IOR	PC4IOR	PC3IOR	PC2IOR	PC1IOR	PC0IOR	
	PCDR0	—	—	—	—	—	—	—	PC8DR	
		PC7DR	PC6DR	PC5DR	PC4DR	PC3DR	PC2DR	PC1DR	PC0DR	
	PCPR0	—	—	—	—	—	—	—	PC8PR	
		PC7PR	PC6PR	PC5PR	PC4PR	PC3PR	PC2PR	PC1PR	PC0PR	
	PDCR3	—	—	PD15MD1	PD15MD0	—	—	PD14MD1	PD14MD0	
		—	—	PD13MD1	PD13MD0	—	—	PD12MD1	PD12MD0	
	PDCR2	—	—	PD11MD1	PD11MD0	—	—	PD10MD1	PD10MD0	
		—	—	PD9MD1	PD9MD0	—	—	PD8MD1	PD8MD0	
	PDCR1	—	—	PD7MD1	PD7MD0	—	—	PD6MD1	PD6MD0	
		—	—	PD5MD1	PD5MD0	—	—	PD4MD1	PD4MD0	
	PDCR0	—	—	PD3MD1	PD3MD0	—	—	PD2MD1	PD2MD0	
		—	—	PD1MD1	PD1MD0	—	—	PD0MD1	PD0MD0	
	PDIOR0	PD15IOR	PD14IOR	PD13IOR	PD12IOR	PD11IOR	PD10IOR	PD9IOR	PD8IOR	
		PD7IOR	PD6IOR	PD5IOR	PD4IOR	PD3IOR	PD2IOR	PD1IOR	PD0IOR	
	PDDR0	PD15DR	PD14DR	PD13DR	PD12DR	PD11DR	PD10DR	PD9DR	PD8DR	
		PD7DR	PD6DR	PD5DR	PD4DR	PD3DR	PD2DR	PD1DR	PD0DR	
	PDPR0	PD15PR	PD14PR	PD13PR	PD12PR	PD11PR	PD10PR	PD9PR	PD8PR	
		PD7PR	PD6PR	PD5PR	PD4PR	PD3PR	PD2PR	PD1PR	PD0PR	
	PECR1	—	—	PE7MD1	PE7MD0	—	—	PE6MD1	PE6MD0	
		—	—	PE5MD1	PE5MD0	—	—	PE4MD1	PE4MD0	
PECR0	—	PE3MD2	PE3MD1	PE3MD0	—	PE2MD2	PE2MD1	PE2MD0		
	—	PE1MD2	PE1MD1	PE1MD0	—	—	PE0MD1	PE0MD0		
PEIOR0	—	—	—	—	—	—	—	—		
	PE7IOR	PE6IOR	PE5IOR	PE4IOR	PE3IOR	PE2IOR	PE1IOR	PE0IOR		

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
通用输入 / 输出端口	PEDR0	—	—	—	—	—	—	—	—
		PE7DR	PE6DR	PE5DR	PE4DR	PE3DR	PE2DR	PE1DR	PE0DR
	PEPR0	—	—	—	—	—	—	—	—
		PE7PR	PE6PR	PE5PR	PE4PR	PE3PR	PE2PR	PE1PR	PE0PR
	PFCR6	—	PF23MD2	PF23MD1	PF23MD0	—	PF22MD2	PF22MD1	PF22MD0
		—	PF21MD2	PF21MD1	PF21MD0	—	PF20MD2	PF20MD1	PF20MD0
	PFCR5	—	PF19MD2	PF19MD1	PF19MD0	—	PF18MD2	PF18MD1	PF18MD0
		—	PF17MD2	PF17MD1	PF17MD0	—	PF16MD2	PF16MD1	PF16MD0
	PFCR4	—	—	—	—	—	—	—	—
		—	—	—	—	—	PF15MD2	PF15MD1	PF15MD0
	PFCR3	—	—	—	—	—	PF14MD2	PF14MD1	PF14MD0
		—	PF13MD2	PF13MD1	PF13MD0	—	PF12MD2	PF12MD1	PF12MD0
	PFCR2	—	PF11MD2	PF11MD1	PF11MD0	—	PF10MD2	PF10MD1	PF10MD0
		—	PF9MD2	PF9MD1	PF9MD0	—	PF8MD2	PF8MD1	PF8MD0
	PFCR1	—	PF7MD2	PF7MD1	PF7MD0	—	PF6MD2	PF6MD1	PF6MD0
		—	PF5MD2	PF5MD1	PF5MD0	—	PF4MD2	PF4MD1	PF4MD0
	PFCR0	—	PF3MD2	PF3MD1	PF3MD0	—	PF2MD2	PF2MD1	PF2MD0
		—	PF1MD2	PF1MD1	PF1MD0	—	PF0MD2	PF0MD1	PF0MD0
	PFIOR1	—	—	—	—	—	—	—	—
		PF23IOR	PF22IOR	PF21IOR	PF20IOR	PF19IOR	PF18IOR	PF17IOR	PF16IOR
	PFIOR0	PF15IOR	PF14IOR	PF13IOR	PF12IOR	PF11IOR	PF10IOR	PF9IOR	PF8IOR
		PF7IOR	PF6IOR	PF5IOR	PF4IOR	PF3IOR	PF2IOR	PF1IOR	PF0IOR
	PFDR1	—	—	—	—	—	—	—	—
		PF23DR	PF22DR	PF21DR	PF20DR	PF19DR	PF18DR	PF17DR	PF16DR
	PFDR0	PF15DR	PF14DR	PF13DR	PF12DR	PF11DR	PF10DR	PF9DR	PF8DR
		PF7DR	PF6DR	PF5DR	PF4DR	PF3DR	PF2DR	PF1DR	PF0DR
	PFPR1	—	—	—	—	—	—	—	—
		PF23PR	PF22PR	PF21PR	PF20PR	PF19PR	PF18PR	PF17PR	PF16PR
	PFPR0	PF15PR	PF14PR	PF13PR	PF12PR	PF11PR	PF10PR	PF9PR	PF8PR
		PF7PR	PF6PR	PF5PR	PF4PR	PF3PR	PF2PR	PF1PR	PF0PR
	PGCR6	—	—	PG27MD1	PG27MD0	—	—	PG26MD1	PG26MD0
		—	—	PG25MD1	PG25MD0	—	—	PG24MD1	PG24MD0
	PGCR5	—	PG23MD2	PG23MD1	PG23MD0	—	PG22MD2	PG22MD1	PG22MD0
		—	PG21MD2	PG21MD1	PG21MD0	—	PG20MD2	PG20MD1	PG20MD0
	PGCR4	—	PG19MD2	PG19MD1	PG19MD0	—	PG18MD2	PG18MD1	PG18MD0
		—	—	PG17MD1	PG17MD0	—	—	PG16MD1	PG16MD0
	PGCR3	—	—	PG15MD1	PG15MD0	—	—	PG14MD1	PG14MD0
		—	—	PG13MD1	PG13MD0	—	—	PG12MD1	PG12MD0
	PGCR2	—	PG11MD2	PG11MD1	PG11MD0	—	PG10MD2	PG10MD1	PG10MD0
		—	PG9MD2	PG9MD1	PG9MD0	—	PG8MD2	PG8MD1	PG8MD0
	PGCR1	—	PG7MD2	PG7MD1	PG7MD0	—	PG6MD2	PG6MD1	PG6MD0
		—	PG5MD2	PG5MD1	PG5MD0	—	PG4MD2	PG4MD1	PG4MD0
	PGCR0	—	PG3MD2	PG3MD1	PG3MD0	—	PG2MD2	PG2MD1	PG2MD0
		—	PG1MD2	PG1MD1	PG1MD0	—	PG0MD2	PG0MD1	PG0MD0
	PGIOR1	—	—	—	—	PG27IOR	PG26IOR	PG25IOR	PG24IOR
		PG23IOR	PG22IOR	PG21IOR	PG20IOR	PG19IOR	PG18IOR	PG17IOR	PG16IOR

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
通用输入 / 输出端口	PGIOR0	PG15IOR	PG14IOR	PG13IOR	PG12IOR	PG11IOR	PG10IOR	PG9IOR	PG8IOR
		PG7IOR	PG6IOR	PG5IOR	PG4IOR	PG3IOR	PG2IOR	PG1IOR	PG0IOR
	PGDR1	—	—	—	—	PG27DR	PG26DR	PG25DR	PG24DR
		PG23DR	PG22DR	PG21DR	PG20DR	PG19DR	PG18DR	PG17DR	PG16DR
	PGDR0	PG15DR	PG14DR	PG13DR	PG12DR	PG11DR	PG10DR	PG9DR	PG8DR
		PG7DR	PG6DR	PG5DR	PG4DR	PG3DR	PG2DR	PG1DR	PG0DR
	PGPR1	—	—	—	—	PG27PR	PG26PR	PG25PR	PG24PR
		PG23PR	PG22PR	PG21PR	PG20PR	PG19PR	PG18PR	PG17PR	PG16PR
	PGPR0	PG15PR	PG14PR	PG13PR	PG12PR	PG11PR	PG10PR	PG9PR	PG8PR
		PG7PR	PG6PR	PG5PR	PG4PR	PG3PR	PG2PR	PG1PR	PG0PR
	PHCR1	—	—	PH7MD1	PH7MD0	—	—	PH6MD1	PH6MD0
		—	—	PH5MD1	PH5MD0	—	—	PH4MD1	PH4MD0
	PHCR0	—	—	PH3MD1	PH3MD0	—	—	PH2MD1	PH2MD0
		—	—	PH1MD1	PH1MD0	—	—	PH0MD1	PH0MD0
	PHPR0	—	—	—	—	—	—	—	—
		PH7PR	PH6PR	PH5PR	PH4PR	PH3PR	PH2PR	PH1PR	PH0PR
	PJCR7	—	—	—	PJ31MD	—	PJ30MD2	PJ30MD1	PJ30MD0
		—	PJ29MD2	PJ29MD1	PJ29MD0	—	PJ28MD2	PJ28MD1	PJ28MD0
	PJCR6	—	PJ27MD2	PJ27MD1	PJ27MD0	—	PJ26MD2	PJ26MD1	PJ26MD0
		—	PJ25MD2	PJ25MD1	PJ25MD0	—	PJ24MD2	PJ24MD1	PJ24MD0
	PJCR5	—	PJ23MD2	PJ23MD1	PJ23MD0	—	PJ22MD2	PJ22MD1	PJ22MD0
		—	PJ21MD2	PJ21MD1	PJ21MD0	—	PJ20MD2	PJ20MD1	PJ20MD0
	PJCR4	—	PJ19MD2	PJ19MD1	PJ19MD0	—	PJ18MD2	PJ18MD1	PJ18MD0
		—	PJ17MD2	PJ17MD1	PJ17MD0	—	PJ16MD2	PJ16MD1	PJ16MD0
	PJCR3	—	PJ15MD2	PJ15MD1	PJ15MD0	—	PJ14MD2	PJ14MD1	PJ14MD0
		—	PJ13MD2	PJ13MD1	PJ13MD0	—	PJ12MD2	PJ12MD1	PJ12MD0
	PJCR2	—	PJ11MD2	PJ11MD1	PJ11MD0	—	PJ10MD2	PJ10MD1	PJ10MD0
		—	PJ9MD2	PJ9MD1	PJ9MD0	—	PJ8MD2	PJ8MD1	PJ8MD0
	PJCR1	—	PJ7MD2	PJ7MD1	PJ7MD0	—	PJ6MD2	PJ6MD1	PJ6MD0
		—	PJ5MD2	PJ5MD1	PJ5MD0	—	PJ4MD2	PJ4MD1	PJ4MD0
	PJCR0	—	PJ3MD2	PJ3MD1	PJ3MD0	—	PJ2MD2	PJ2MD1	PJ2MD0
		—	PJ1MD2	PJ1MD1	PJ1MD0	—	PJ0MD2	PJ0MD1	PJ0MD0
	PJIOR1	PJ31IOR	PJ30IOR	PJ29IOR	PJ28IOR	PJ27IOR	PJ26IOR	PJ25IOR	PJ24IOR
		PJ23IOR	PJ22IOR	PJ21IOR	PJ20IOR	PJ19IOR	PJ18IOR	PJ17IOR	PJ16IOR
	PJIOR0	PJ15IOR	PJ14IOR	PJ13IOR	PJ12IOR	PJ11IOR	PJ10IOR	PJ9IOR	PJ8IOR
		PJ7IOR	PJ6IOR	PJ5IOR	PJ4IOR	PJ3IOR	PJ2IOR	PJ1IOR	PJ0IOR
	PJDR1	PJ31DR	PJ30DR	PJ29DR	PJ28DR	PJ27DR	PJ26DR	PJ25DR	PJ24DR
		PJ23DR	PJ22DR	PJ21DR	PJ20DR	PJ19DR	PJ18DR	PJ17DR	PJ16DR
	PJDR0	PJ15DR	PJ14DR	PJ13DR	PJ12DR	PJ11DR	PJ10DR	PJ9DR	PJ8DR
		PJ7DR	PJ6DR	PJ5DR	PJ4DR	PJ3DR	PJ2DR	PJ1DR	PJ0DR
	PJPR1	PJ31PR	PJ30PR	PJ29PR	PJ28PR	PJ27PR	PJ26PR	PJ25PR	PJ24PR
		PJ23PR	PJ22PR	PJ21PR	PJ20PR	PJ19PR	PJ18PR	PJ17PR	PJ16PR
	PJPR0	PJ15PR	PJ14PR	PJ13PR	PJ12PR	PJ11PR	PJ10PR	PJ9PR	PJ8PR
		PJ7PR	PJ6PR	PJ5PR	PJ4PR	PJ3PR	PJ2PR	PJ1PR	PJ0PR
	SNCR	—	—	—	—	—	—	—	—
		—	—	SSI5NCE	SSI4NCE	SSI3NCE	SSI2NCE	SSI1NCE	SSI0NCE

模块名	寄存器略称	bit 31/23/15/7	bit 30/22/14/6	bit 29/21/13/5	bit 28/20/12/4	bit 27/19/11/3	bit 26/18/10/2	bit 25/17/9/1	bit 24/16/8/0
低功耗模式	STBCR1	STBY	DEEP	—	—	—	—	—	—
	STBCR2	MSTP10	—	MSTP8	MSTP7	—	—	—	—
	STBCR3	HIZ	MSTP36	MSTP35	—	—	MSTP32	—	MSTP30
	STBCR4	MSTP47	MSTP46	MSTP45	MSTP44	MSTP43	MSTP42	MSTP41	MSTP40
	STBCR5	MSTP57	MSTP56	MSTP55	MSTP54	MSTP53	MSTP52	MSTP51	MSTP50
	STBCR6	MSTP67	MSTP66	MSTP65	MSTP64	MSTP63	MSTP62	MSTP61	MSTP60
	STBCR7	MSTP77	MSTP76	MSTP75	—	MSTP73	MSTP72	—	MSTP70
	STBCR8	MSTP87	MSTP86	MSTP85	MSTP84	—	MSTP82	MSTP81	—
	STBCR9	MSTP97	MSTP96	MSTP95	MSTP94	MSTP93	MSTP92	MSTP91	MSTP90
	STBCR10	MSTP107	MSTP106	MSTP105	—	MSTP103	MSTP102	MSTP101	MSTP100
	SWRSTCR1	AXTALE	SSIF5SRST	SSIF4SRST	IEBSRST	SSIF3SRST	SSIF2SRST	SSIF1SRST	SSIF0SRST
	SWRSTCR2	—	—	—	JCUSRST	RGPVGSRST	—	—	—
	SYSCR1	—	—	—	—	RAME3	RAME2	RAME1	RAME0
	SYSCR2	—	—	—	—	RAMWE3	RAMWE2	RAMWE1	RAMWE0
	SYSCR3	—	—	VRAME5	VRAME4	VRAME3	VRAME2	VRAME1	VRAME0
	SYSCR4	—	—	VRAMWE5	VRAMWE4	VRAMWE3	VRAMWE2	VRAMWE1	VRAMWE0
	SYSCR5	—	—	—	—	RRAMWE3	RRAMWE2	RRAMWE1	RRAMWE0
	RRAMKP	—	—	—	—	RRAMKP3	RRAMKP2	RRAMKP1	RRAMKP0
	DSCTR	EBUSKEEPE	RAMBOOT	—	—	—	—	—	—
	DSSSR	—	PJ23	PJ22	PJ21	PJ20	PG3	PG2	NMI
		—	RTCAR	PF19	PF18	PF17	PF16	PC7	PC5
	DSESR	—	PJ23E	PJ22E	PJ21E	PJ20E	PG3E	PG2E	NMIE
		—	—	PF19E	PF18E	PF17E	PF16E	PC7E	PC5E
	DSFR	IOKEEP	PJ23F	PJ22F	PJ21F	PJ20F	PG3F	PG2F	NMIF
		—	RTCARF	PF19F	PF18F	PF17F	PF16F	PC7F	PC5F
	XTALCTR	—	—	—	—	—	—	—	GAIN
	用户调试 接口	SDIR	T1[7]	T1[6]	T1[5]	T1[4]	T1[3]	T1[2]	T1[1]
—			—	—	—	—	—	—	—

- 【注】 *1 这是 MCR15 位为“0”的情况。
*2 这是 MCR15 位为“1”的情况。
*3 这是命令存取模式的情况。
*4 这是扇区存取模式的情况。

51.3 各运行模式中的寄存器状态一览表

模块	寄存器名	上电复位	手动复位	深度待机	软件待机	模块待机	睡眠
时钟脉冲振荡器	FRQCR	初始化 *1	保持	初始化	保持	—	保持
中断控制器	IBNR	初始化	保持 *2	初始化	保持	—	保持
	上述以外的全部寄存器	初始化	保持	初始化	保持	—	保持
用户断点控制器	全部寄存器	初始化	保持	初始化	保持	保持	保持
高速缓存	全部寄存器	初始化	保持	初始化	保持	—	保持
总线状态控制器	RTCSR	初始化	保持 *3	初始化	保持	—	保持 *3
	RTCNT	初始化	保持 *4	初始化	保持	—	保持 *4
	上述以外的全部寄存器	初始化	保持	初始化	保持	—	保持
直接存储器存取控制器	全部寄存器	初始化	保持	初始化	保持	保持	保持 *7
多功能定时器脉冲单元 2	全部寄存器	初始化	保持	初始化	初始化	初始化	保持
比较匹配定时器	全部寄存器	初始化	保持	初始化	初始化	保持	保持
看门狗定时器	WRCSR	初始化 *1	保持	初始化	保持	—	保持
	上述以外的全部寄存器	初始化	保持	初始化	保持	—	保持
实时时钟	R64CNT	保持 *4	保持 *4	保持 *4	保持 *4	保持	保持 *4
	RSECCNT	保持 *4	保持 *4	保持 *4	保持 *4	保持	保持 *4
	RMINCNT	保持 *4	保持 *4	保持 *4	保持 *4	保持	保持 *4
	RHRCNT	保持 *4	保持 *4	保持 *4	保持 *4	保持	保持 *4
	RWKCNT	保持 *4	保持 *4	保持 *4	保持 *4	保持	保持 *4
	RDAYCNT	保持 *4	保持 *4	保持 *4	保持 *4	保持	保持 *4
	RMONCNT	保持 *4	保持 *4	保持 *4	保持 *4	保持	保持 *4
	RYRCNT	保持 *4	保持 *4	保持 *4	保持 *4	保持	保持 *4
	RSECAR	保持	保持	保持	保持	保持	保持
	RMINAR	保持	保持	保持	保持	保持	保持
	RHRAR	保持	保持	保持	保持	保持	保持
	RWKAR	保持	保持	保持	保持	保持	保持
	RDAYAR	保持	保持	保持	保持	保持	保持
	RMONAR	保持	保持	保持	保持	保持	保持
	RYRAR	保持	保持	保持	保持	保持	保持
	RCR1	初始化	初始化	初始化	保持	保持	保持
	RCR2	初始化	初始化 *5	初始化	保持	保持	保持
	RCR3	初始化	保持	初始化	保持	保持	保持
	RCR5	保持	保持	保持	保持	保持	保持
	RFRH	保持	保持	保持	保持	保持	保持
RFRL	保持	保持	保持	保持	保持	保持	
FIFO 内置型串行通信接口	全部寄存器	初始化	保持	初始化	保持	保持	保持
瑞萨串行外围接口	全部寄存器	初始化	保持	初始化	保持	保持	保持
瑞萨四路串行外围接口	全部寄存器	初始化	保持	初始化	保持	保持	保持
SPI 多 I/O 总线控制器	全部寄存器	初始化	保持	初始化	保持	保持	保持
I ² C 总线接口 3	ICMR_0、1、2、3	初始化	保持	初始化	保持 *6	保持 *6	保持
	上述以外的全部寄存器	初始化	保持	初始化	保持	保持	保持

模块	寄存器名	上电复位	手动复位	深度待机	软件待机	模块待机	睡眠
串行音频接口	全部寄存器	初始化	保持	初始化	保持	保持	保持
带 FIFO 的时钟同步串行 I/O	全部寄存器	初始化	保持	初始化	保持	保持	保持
控制器局域网	全部寄存器	初始化	保持	初始化	保持	保持	保持
IEBus 控制器	全部寄存器	初始化	保持	初始化	保持	保持	保持
瑞萨 SPDIF 接口	全部寄存器	初始化	保持	初始化	保持	保持	保持
CD-ROM 解码器	全部寄存器	初始化	保持	初始化	保持	保持	保持
A/D 转换器	全部寄存器	初始化	保持	初始化	初始化	初始化	保持
NAND 闪存控制器	全部寄存器	初始化	保持	初始化	保持	保持	保持
USB2.0 主机 / 功能模块	全部寄存器	初始化	保持	初始化	保持	保持	保持
数字视频解码器	全部寄存器	初始化	保持	初始化	保持	保持	保持
视频显示控制器 4	全部寄存器	初始化	保持	初始化	保持	保持	保持
失真校正引擎	全部寄存器	初始化	保持	初始化	保持	保持	保持
显示输出比较单元	全部寄存器	初始化	保持	初始化	保持	保持	保持
JPEG 编解码器单元	全部寄存器	初始化	保持	初始化	保持	保持	保持
采样率转换器	全部寄存器	初始化	保持	初始化	保持	保持	保持
音频发生器	全部寄存器	初始化	保持	初始化	保持	保持	保持
MMC 主机接口	全部寄存器	初始化	保持	初始化	保持	保持	保持
马达控制 PWM 定时器	全部寄存器	初始化	保持	初始化	保持	保持	保持
通用输入 / 输出端口	全部寄存器	初始化	保持	初始化	保持	—	保持
低功耗模式	DSFR	初始化	保持	保持	保持	—	保持
	XTALCTR	初始化 *10	保持	保持 *9	保持 *9	—	保持
	上述以外的全部寄存器	初始化	保持	初始化	保持	—	保持
用户调试接口 *8	SDIR	保持	保持	初始化	保持	保持	保持

【注】 *1 在通过看门狗定时器进行的内部上电复位时保持状态。

*2 对 BN[3:0] 位进行初始化。

*3 继续处理标志。

*4 继续进行递增计数。

*5 保持 RTCEN 位和 START 位。

*6 对 BC[2:0] 位进行初始化。

*7 能继续传送。

*8 在 TRST 有效时或者在 TAP 的 Test-Logic-Reset 状态下进行初始化。

*9 在实时时钟不使用 EXTAL 时进行初始化。

*10 在通过看门狗定时器进行的内部上电复位或者通过用户调试接口进行的复位时保持状态。

52. 电特性

52.1 绝对最大额定值

表 52.1 绝对最大额定值

项目		符号	额定值	单位
电源电压 (I/O)		PV_{CC}	-0.3 ~ 4.6	V
电源电压 (内部)		V_{CC}	-0.3 ~ 1.7	V
PLL 电源电压		$PLL_{V_{CC}}$	-0.3 ~ 4.6	V
模拟电源电压		AV_{CC}	-0.3 ~ 4.6	V
模拟基准电压		AV_{ref}	-0.3 ~ $AV_{CC}+0.3$	V
USB 收发器的模拟电源电压 (I/O)		$USBAPV_{CC}$	-0.3 ~ 4.6	V
USB 收发器的数字电源电压 (I/O)		$USBDPV_{CC}$	-0.3 ~ 4.6	V
【注】 SH7269 (BGA) 没有此项。				
USB 收发器的模拟电源电压 (内部)		$USBAV_{CC}$	-0.3 ~ 1.7	V
USB 收发器的数字电源电压 (内部)		$USBDV_{CC}$	-0.3 ~ 1.7	V
【注】 SH7269 (BGA) 没有此项。				
USB480MHz 的电源 (内部)		$USBUV_{CC}$	-0.3 ~ 1.7	V
【注】 SH7269 (BGA) 没有此项。				
用于输入视频信号的 AD 转换器电源电压		$VDAV_{CC}$	-0.3 ~ 4.6	V
输入电压	VBUS	V_{in}	-0.3 ~ 5.5	V
	其他输入引脚	V_{in}	-0.3 ~ 3.3V 系电源 (PV_{CC} 、 $PLL_{V_{CC}}$ 、 AV_{CC} 、 $USBAPV_{CC}$ 、 $USBDPV_{CC}$ 、 $VDAV_{CC}$) +0.3	V
工作温度	标准温度范围的产品	T_{opr}	-20 ~ 85	°C
	大温度范围的产品		-40 ~ 85	
保存温度		T_{stg}	-55 ~ 125	°C

【使用时的注意事项】

如果在使用 LSI 时超过绝对最大额定值，就可能造成 LSI 的永久性破坏。

52.2 接通和切断电源的顺序

1.2V 系电源 (V_{CC} 、 $USBAV_{CC}$ 、 $USBDV_{CC}$ 、 $USBUV_{CC}$) 和 3.3V 系电源 (PV_{CC} 、 $PLL_{V_{CC}}$ 、 AV_{CC} 、 $USBAPV_{CC}$ 、 $USBDPV_{CC}$ 、 $VDAV_{CC}$) 的接通和切断顺序无论哪个在先都没有问题。

在接通电源时，必须将 \overline{TRST} 引脚和 \overline{RES} 引脚置为 Low 电平。否则，输出引脚和输入 / 输出引脚的输出不确定，可能会引起整个系统的误动作。

在切断电源时，也可能发生上述问题，必须将 \overline{TRST} 引脚和 \overline{RES} 引脚置为 Low 电平。

52.3 DC 特性

- “表52.2 DC特性(2)【消耗电流】SH7268/SH7269 (QFP)”和“表52.2 DC特性(3)【消耗电流】SH7269 (BGA)”以外的条件
 $V_{CC}=USBV_{CC}=USBV_{CC}=1.15\sim 1.35V$, $PV_{CC}=USBDPV_{CC}=3.0\sim 3.6V$, $PLL_{CC}=3.0\sim 3.6V$,
 $AV_{CC}=3.0\sim 3.6V$, $USBAV_{CC}=1.15\sim 1.35V$, $USBAPV_{CC}=3.0\sim 3.6V$, $VDAV_{CC}=3.0\sim 3.6V$,
 $V_{SS}=PLL_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=USBV_{SS}=VDAV_{SS}=0V$,
 $T_a=-20\sim 85^{\circ}C$ (标准温度范围)、 $-40\sim 85^{\circ}C$ (大温度范围)
- “表52.2 DC特性(2)【消耗电流】SH7268/SH7269 (QFP)”和“表52.2 DC特性(3)【消耗电流】SH7269 (BGA)”的条件
 $V_{CC}=USBV_{CC}=USBV_{CC}=1.25V$, $PV_{CC}=USBDPV_{CC}=3.3V$, $PLL_{CC}=3.3V$, $AV_{CC}=3.3V$,
 $USBAV_{CC}=1.25V$, $USBAPV_{CC}=3.3V$, $VDAV_{CC}=3.3V$,
 $V_{SS}=PLL_{SS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=USBV_{SS}=VDAV_{SS}=0V$,
 $AV_{ref}=3.3V$, $V_{BUS}=5.0V$
 $T_a=-20\sim 85^{\circ}C$ (标准温度范围)、 $-40\sim 85^{\circ}C$ (大温度范围)
 $I_{\phi}=266.67MHz$, $B_{\phi}=133.33MHz$, $P1_{\phi}=66.67MHz$, $P0_{\phi}=33.33MHz$

【注】 SH7269 (BGA) 没有 $USBDV_{CC}$ 、 $USBV_{CC}$ 、 $USBDPV_{CC}$ 、 PLL_{SS} 、 $USBDV_{SS}$ 、 $USBAV_{SS}$ 、 $USBV_{SS}$ 、 $USBDPV_{SS}$ 、 $USBAPV_{SS}$ 引脚。

表 52.2 DC 特性(1)【共用项目】

项目	符号	Min.	Typ.	Max.	单位	测量条件
电源电压	PV_{CC}	3.0	3.3	3.6	V	
	V_{CC}	1.15	1.25	1.35	V	
PLL 电源电压	PLL_{CC}	3.0	3.3	3.6	V	
模拟电源电压	AV_{CC}	3.0	3.3	3.6	V	
USB 电源电压 【注】 SH7269 (BGA) 没有 USBDPV _{CC} 、USBDV _{CC} 、 USBV _{CC} 引脚。	$USBAPV_{CC}$	3.0	3.3	3.6	V	
	$USBAV_{CC}$ $USBDV_{CC}$ $USBV_{CC}$	1.15	1.25	1.35	V	
用于输入视频信号的 AD 转换器电源电压	$VDAV_{CC}$	3.0	3.3	3.6	V	
输入泄漏电流	全部输入引脚	$ I_{in} $	—	—	1.0	μA $V_{in}=0.5\sim PV_{CC}-0.5V$
三态泄漏电流	全部输入 / 输出引脚和输出 引脚 (PE7 ~ PE0 除外) (OFF 状态)	$ I_{STI} $	—	—	1.0	μA $V_{in}=0.5\sim PV_{CC}-0.5V$
	PE7 ~ PE0		—	—	10	μA
引脚电容	USB2.0 主机 / 功能模块的 相关引脚 *	C_{in}	—	—	20	pF
	上述以外的全部输入 / 输出 引脚和输入引脚		—	—	10	pF

【注】 * DP、DM、VBUS 引脚

表 52.2 DC 特性 (2) 【消耗电流】 SH7268/SH7269 (QFP)

项目	电源种类	符号	Typ.	Max.	单位	测量条件	
正常运行时的消耗电流	V_{CC}	I_{CC}	220	285	mA		
	$PLL_{V_{CC}}$	$PLL_{I_{CC}}$	11	13	mA		
	PV_{CC}	PI_{CC}^*	92	—	mA		
	AV_{CC}	AI_{CC}	1	4	mA	A/D 转换中	
			1	3	μ A	A/D 待机中	
	AV_{ref}	AI_{ref}	1	4	mA	A/D 转换中、 A/D 待机中	
	$USBAV_{CC+}$ $USBDV_{CC+}$ $USBV_{CC}$	UI_{CC}	19	22	mA	USB 高速通信时	
	$USBAPV_{CC+}$ $USBDPV_{CC}$	UPI_{CC}	44	47	mA	USB 高速通信时	
	$VBUS$	VI_{CC}	8.5	10	μ A		
$VDAV_{CC}$	$VDAI_{CC}$	7.5	8	mA			
睡眠时的消耗电流	V_{CC}	I_{sleep}	170	240	mA		
	除上述以外, 和正常运行时相同。						
软件待机时的消耗电流	$T_a > 50^\circ\text{C}$	V_{CC+} $USBAV_{CC+}$ $USBDV_{CC+}$ $USBV_{CC}$	I_{sstby}	7	65	mA	
		PV_{CC+} $PLL_{V_{CC+}}$ $USBAPV_{CC+}$ $USBDPV_{CC+}$ $VDAV_{CC}$	PI_{sstby}	4.5	6.5	mA	
		除上述以外, 和正常运行时相同。					
	$T_a \leq 50^\circ\text{C}$	V_{CC+} $USBAV_{CC+}$ $USBDV_{CC+}$ $USBV_{CC}$	I_{sstby}	4	35	mA	
		PV_{CC+} $PLL_{V_{CC+}}$ $USBAPV_{CC+}$ $USBDPV_{CC+}$ $VDAV_{CC}$	PI_{sstby}	4.5	6.5	mA	
		除上述以外, 和正常运行时相同。					

项目		电源种类	符号	Typ.	Max.	单位	测量条件	
深度待机时的消耗电流	$T_a > 50^\circ\text{C}$	V_{CC+} USB A V_{CC+} USB D V_{CC+} USB U V_{CC}	I_{dstby}	6	27	μA	保持 RAM0KB 并且选择 RTC_X1 时	
				8	40	μA	保持 RAM16KB 并且选择 RTC_X1 时	
				10	53	μA	保持 RAM32KB 并且选择 RTC_X1 时	
				14	80	μA	保持 RAM64KB 并且选择 RTC_X1 时	
				22	132	μA	保持 RAM128KB 并且选择 RTC_X1 时	
		在选择 EXTAL 13MHz 时, 上述内容加上 Typ.5 μA 、Max.6 μA 。						
		PV_{CC+} PLL V_{CC+} AV_{CC+} AV_{ref+} USB A PV_{CC+} USB D PV_{CC+} $VDAV_{CC}$	$P I_{dstby}$	5.5	20	μA	RTC 不运行时	
				9.5	24	μA	选择 RTC_X1 时	
	1			—	mA	选择 EXTAL 13MHz 驱动小 *		
	$VBUS$	$V I_{CC}$	8.5	10	μA			
	$T_a \leq 50^\circ\text{C}$	V_{CC+} USB A V_{CC+} USB D V_{CC+} USB U V_{CC}	I_{dstby}	4	19	μA	保持 RAM0KB 并且选择 RTC_X1 时	
				5.5	29	μA	保持 RAM16KB 并且选择 RTC_X1 时	
				7	39	μA	保持 RAM32KB 并且选择 RTC_X1 时	
				10	58	μA	保持 RAM64KB 并且选择 RTC_X1 时	
				16	97	μA	保持 RAM128KB 并且选择 RTC_X1 时	
		在选择 EXTAL 13MHz 时, 上述内容加上 Typ.5 μA 、Max.6 μA 。						
PV_{CC+} PLL V_{CC+} AV_{CC+} AV_{ref+} USB A PV_{CC+} USB D PV_{CC+} $VDAV_{CC}$		$P I_{dstby}$	5	16	μA	RTC 不运行时		
			9	20	μA	选择 RTC_X1 时		
			1	—	mA	选择 EXTAL 13MHz 驱动小 *		
$VBUS$		$V I_{CC}$	8.5	10	μA			

【注】 * 这是参考值。因为实际工作电流很大程度取决于系统（IO 的负载引起的波形变形和交替频率等），所以必须进行系统的实际测量。

表 52.2 DC 特性 (3) 【消耗电流】SH7269 (BGA)

项目	电源种类	符号	Typ.	Max.	单位	测量条件	
正常运行时的消耗电流	V_{CC}	I_{CC}	220	285	mA	USB 不运行时	
			在 USB 高速通信时, 上述内容加上 Typ.12mA、Max.14mA。				
	$PLL_{V_{CC}}$	$PLL_{I_{CC}}$	11	13	mA		
	PV_{CC}	PI_{CC}^*	92	—	mA	USB 不运行时	
			在 USB 高速通信时, 上述内容加上 Typ.42mA。				
	AV_{CC}	AI_{CC}	1	4	mA	A/D 转换中	
			1	3	μ A	A/D 待机中	
	AV_{ref}	AI_{ref}	1	4	mA	A/D 转换中、 A/D 待机中	
	$USBAV_{CC}$	UI_{CC}	7	8	mA	USB 高速通信时	
	$USBAPV_{CC}$	UPI_{CC}	2	2.5	mA	USB 高速通信时	
$VBUS$	VI_{CC}	8.5	10	μ A			
$VDAV_{CC}$	$VDAI_{CC}$	7.5	8	mA			
睡眠时的消耗电流	V_{CC}	I_{sleep}	170	240	mA		
			在 USB 高速通信时, 上述内容加上 Typ.12mA、Max.14mA。				
除上述以外, 和正常运行时相同。							
软件待机时的消耗电流	$T_a > 50^\circ\text{C}$	$V_{CC}+$ $USBAV_{CC}$	I_{sstby}	7	65	mA	
		$PV_{CC}+$ $PLL_{V_{CC}}+$ $USBAPV_{CC}+$ $VDAV_{CC}$	PI_{sstby}	4.5	6.5	mA	
		除上述以外, 和正常运行时相同。					
	$T_a \leq 50^\circ\text{C}$	$V_{CC}+$ $USBAV_{CC}$	I_{sstby}	4	35	mA	
		$PV_{CC}+$ $PLL_{V_{CC}}+$ $USBAPV_{CC}+$ $VDAV_{CC}$	PI_{sstby}	4.5	6.5	mA	
		除上述以外, 和正常运行时相同。					

项目		电源种类	符号	Typ.	Max.	单位	测量条件	
深度待机时的消耗电流	$T_a > 50^\circ\text{C}$	V_{CC+} $USBV_{CC}$	I_{dstby}	6	27	μA	保持RAM0KB并且选择 RTC_X1 时	
				8	40	μA	保持 RAM16KB 并且选择 RTC_X1 时	
				10	53	μA	保持 RAM32KB 并且选择 RTC_X1 时	
				14	80	μA	保持 RAM64KB 并且选择 RTC_X1 时	
				22	132	μA	保持 RAM128KB 并且选择 RTC_X1 时	
		在选择 EXTAL 13MHz 时, 上述内容加上 Typ.5 μA 、Max.6 μA 。						
		PV_{CC+} $PLL_{V_{CC+}}$ AV_{CC+} AV_{ref+} $USBAPV_{CC+}$ $VDAV_{CC}$	PI_{dstby}	5.5	20	μA	RTC 不工作时	
				9.5	24	μA	选择 RTC_X1 时	
				1	—	mA	选择 EXTAL 13MHz 驱动小 *	
	$VBUS$	VI_{CC}	8.5	10	μA			
	$T_a \leq 50^\circ\text{C}$	V_{CC+} $USBV_{CC}$	I_{dstby}	4	19	μA	保持RAM0KB并且选择 RTC_X1 时	
				5.5	29	μA	保持 RAM16KB 并且选择 RTC_X1 时	
				7	39	μA	保持 RAM32KB 并且选择 RTC_X1 时	
				10	58	μA	保持 RAM64KB 并且选择 RTC_X1 时	
				16	97	μA	保持 RAM128KB 并且选择 RTC_X1 时	
		在选择 EXTAL 13MHz 时, 上述内容加上 Typ.5 μA 、Max.6 μA 。						
		PV_{CC+} $PLL_{V_{CC+}}$ AV_{CC+} AV_{ref+} $USBAPV_{CC+}$ $VDAV_{CC}$	PI_{dstby}	5	16	μA	RTC 不运行时	
				9	20	μA	选择 RTC_X1 时	
1				—	mA	选择 EXTAL 13MHz 驱动小 *		
$VBUS$	VI_{CC}	8.5	10	μA				

【注】 * 这是参考值。因为实际的工作电流很大程度取决于系统（IO 的负载引起的波形变形和交替频率等），所以必须进行系统的实际测量。

表 52.2 DC 特性 (4) 【I²C 总线接口、USB2.0 主机 / 功能模块的相关引脚除外】

项目	符号	Min.	Typ.	Max.	单位	测量条件	
输入高电平电压 (施密特引脚除外)	V_{IH}	2.2	—	$PV_{CC}+0.3$	V		
输入低电平电压 (施密特引脚除外)	V_{IL}	-0.3	—	0.8	V		
施密特触发输入特性	V_{T^+}	$PV_{CC}\times 0.75$	—	—	V		
	V_{T^-}	—	—	0.5	V		
	$V_{T^+}-V_{T^-}$	0.2	—	—	V		
输出高电平电压	V_{OH}	$PV_{CC}-0.5$	—	—	V	$I_{OH}=-2.0\text{mA}$	
输出低电平电压	V_{OL}	—	—	0.4	V	$I_{OL}=2.0\text{mA}$	
RAM 待机电压	软件待机模式 (高速内部 RAM 和 大容量内部 RAM)	V_{RAMS}	0.85	—	—	V	将 V_{CC} (= $PLL V_{CC}$) 作 为参数进行测量。
	深度待机模式 (只限于用于保持的 内部 RAM)	V_{RAMD}	1.15	—	—	V	

表 52.2 DC 特性 (5) 【I²C 总线接口 3 的相关引脚*】

项目	符号	Min.	Typ.	Max.	单位	测量条件
输入高电平电压	V_{IH}	$PV_{CC}\times 0.7$	—	$PV_{CC}+0.3$	V	
输入低电平电压	V_{IL}	-0.3	—	$PV_{CC}\times 0.3$	V	
施密特触发输入特性	$V_{IH}-V_{IL}$	$PV_{CC}\times 0.05$	—	—	V	
输出低电平电压	V_{OL}	—	—	0.4	V	$I_{OL}=3.0\text{mA}$

【注】 * PE7/SDA3/RxD7 ~ PE0/SCL0/TCLKA/LCD_EXTCLK 引脚 (漏极开路引脚)

表 52.2 DC 特性 (6) 【USB2.0 主机 / 功能模块的相关引脚*】

项目	符号	Min.	Typ.	Max.	单位	测量条件
基准电阻	R_{REF}	5.6k $\Omega\pm 1\%$				
输入高电平电压 (VBUS)	V_{IH}	4.02	—	5.25	V	
输入低电平电压 (VBUS)	V_{IL}	-0.3	—	0.5	V	
输入高电平电压 (USB_X1)	V_{IH}	$PV_{CC}-0.5$	—	$PV_{CC}+0.3$	V	
输入低电平电压 (USB_X1)	V_{IL}	-0.3	—	0.5	V	

【注】 * REFRIN、VBUS、USB_X1、USB_X2 引脚

表 52.2 DC 特性 (7) 【USB2.0 主机 / 功能模块的相关引脚* (低速 / 全速 / 高速共用项目)】

项目	符号	Min.	Typ.	Max.	单位	测量条件
DP 上拉电阻 (选择功能时)	R_{pu}	0.900	—	1.575	k Ω	空闲时
		1.425	—	3.090	k Ω	发送和接收时
DP、DM 下拉电阻 (选择主机功能时)	R_{pd}	14.25	—	24.80	k Ω	

【注】 * DP 引脚和 DM 引脚

表 52.2 DC 特性 (8) 【USB2.0 主机 / 功能模块的相关引脚 * (低速 / 全速时)】

项目	符号	Min.	Typ.	Max.	单位	测量条件
输入高电平电压	V_{IH}	2.0	—	—	V	
输入低电平电压	V_{IL}	—	—	0.8	V	
差动输入灵敏度	V_{DI}	0.2	—	—	V	$ (DP)-(DM) $
差动共模范围	V_{CM}	0.8	—	2.5	V	
输出高电平电压	V_{OH}	2.8	—	3.6	V	$I_{OH}=-200\mu A$
输出低电平电压	V_{OL}	0.0	—	0.3	V	$I_{OL}=2mA$
输出信号交叉电压	V_{CRS}	1.3	—	2.0	V	$C_L=50pF$ (全速时) $C_L=200 \sim 600pF$ (低速时)

【注】 * DP 引脚和 DM 引脚

表 52.2 DC 特性 (9) 【USB2.0 主机 / 功能模块的相关引脚 * (高速时)】

项目	符号	Min.	Typ.	Max.	单位	测量条件
静噪检测阈值电压 (差动电压)	V_{HSSQ}	100	—	150	mV	
共模电压范围	V_{HSCM}	-50	—	500	mV	
空闲状态	V_{HSOI}	-10.0	—	10.0	mV	
输出高电平电压	V_{HSOH}	360	—	440	mV	
输出低电平电压	V_{HSOL}	-10.0	—	10.0	mV	
Chirp J 输出电压 (差分)	V_{CHIRPJ}	700	—	1100	mV	
Chirp K 输出电压 (差分)	V_{CHIRPK}	-900	—	-500	mV	

【注】 * DP 引脚和 DM 引脚

表 52.3 输出容许电流值

项目	符号	Min.	Typ.	Max.	单位
输出低电平的容许电流 (每个引脚)	PE7 ~ PE0	I_{OL}	—	—	10
	其他输出引脚				2
输出低电平的容许电流 (总和)	ΣI_{OL}	—	—	150	mA
输出高电平的容许电流 (每个引脚)	$-I_{OH}$	—	—	2	mA
输出高电平的容许电流 (总和)	$\Sigma -I_{OH}$	—	—	150	mA

【使用时的注意事项】

为了确保 LSI 的可靠性, 输出电流值不能超过表 52.3 中的值。

52.4 AC 特性

本 LSI 的输入原则上是时钟同步输入，只要没有特殊要求，就必须遵守各输入信号的准备时间和保持时间。

条件: $V_{CC}=USB_{DV_{CC}}=USB_{UV_{CC}}=1.15 \sim 1.35V$, $PV_{CC}=USB_{DPV_{CC}}=3.0 \sim 3.6V$, $PLL_{V_{CC}}=3.0 \sim 3.6V$,
 $AV_{CC}=3.0 \sim 3.6V$, $USB_{AV_{CC}}=1.15 \sim 1.35V$, $USB_{APV_{CC}}=3.0 \sim 3.6V$, $VDA_{V_{CC}}=3.0 \sim 3.6V$,
 $V_{SS}=PLL_{V_{SS}}=AV_{SS}=USB_{DV_{SS}}=USB_{AV_{SS}}=USB_{DPV_{SS}}=USB_{APV_{SS}}=USB_{UV_{SS}}=VDA_{V_{SS}}=0V$,
 $T_a=-20 \sim 85^\circ C$ (标准温度范围)、 $-40 \sim 85^\circ C$ (大温度范围)

【注】SH7269 (BGA) 没有 $USB_{DV_{CC}}$ 、 $USB_{UV_{CC}}$ 、 $USB_{DPV_{CC}}$ 、 $PLL_{V_{SS}}$ 、 $USB_{DV_{SS}}$ 、 $USB_{AV_{SS}}$ 、 $USB_{UV_{SS}}$ 、 $USB_{DPV_{SS}}$ 、 $USB_{APV_{SS}}$ 引脚。

表 52.4 工作频率

项目		符号	Min.	Max.	单位	备注
工作频率	CPU 时钟 ($I\phi$)	f	200.00	266.67	MHz	
	内部总线时钟 ($B\phi$)		100.00	133.33	MHz	
	外围时钟 1 ($P1\phi$)		50.00	66.67	MHz	
	外围时钟 0 ($P0\phi$)		25.00	33.33	MHz	

52.4.1 时钟时序

表 52.5 时钟时序

项目	符号	Min.	Max.	单位	参照图
EXTAL 时钟输入频率 (给 USB2.0 主机 / 功能模块提供时钟时)	f_{EX}	12MHz \pm 100ppm			52.1
EXTAL 时钟输入频率 (不给 USB2.0 主机 / 功能模块提供时钟时)		10.00	13.33	MHz	
EXTAL 时钟输入的周期时间 (不给 USB2.0 主机 / 功能模块提供时钟时)	t_{EXcyc}	75.00	100.00	ns	
AUDIO_X1 时钟输入频率 (连接晶体谐振器时)	f_{EX}	10.00	50.00	MHz	
AUDIO_X1 时钟输入的周期时间 (连接晶体谐振器时)	t_{EXcyc}	20.00	100.00	ns	
AUDIO_X1、AUDIO_CLK 时钟输入频率 (输入外部时钟时)	f_{EX}	1.00	50.00	MHz	
AUDIO_X1、AUDIO_CLK 时钟输入的周期时间 (输入外部时钟时)	t_{EXcyc}	20.00	1000.00	ns	
USB_X1 时钟输入频率 (给 USB2.0 主机 / 功能模块提供 12MHz 时钟时)	f_{EX}	12MHz \pm 100ppm			
USB_X1 时钟输入频率 (给 USB2.0 主机 / 功能模块提供 48MHz 时钟并且使用高速传送时)		48MHz \pm 100ppm			

项目	符号	Min.	Max.	单位	参照图
USB_X1 时钟输入频率 (给 USB2.0 主机 / 功能模块提供 48MHz 时钟并且不使用高速传送而使用主机控制器功能时)	f_{EX}	48MHz±500ppm			52.1
USB_X1 时钟输入频率 (给 USB2.0 主机 / 功能模块提供 48MHz 时钟并且不使用高速传送和主机控制器功能时)		48MHz±2500ppm			
VIDEO_X1 时钟输入频率	f_{EX}	27MHz±100ppm*			
EXTAL、AUDIO_X1、AUDIO_CLK、USB_X1 时钟输入的低电平脉宽	t_{EXL}	0.4	0.6	t_{EXcyc}	
VIDEO_X1 时钟输入的低电平脉宽		0.45	0.55		
EXTAL、AUDIO_X1、AUDIO_CLK、USB_X1 时钟输入的高电平脉宽	t_{EXH}	0.4	0.6	t_{EXcyc}	
VIDEO_X1 时钟输入的高电平脉宽		0.45	0.55		
EXTAL、AUDIO_X1、AUDIO_CLK、USB_X1 时钟输入的上升时间	t_{EXr}	—	4	ns	
VIDEO_X1 时钟输入的上升时间		—	3		
EXTAL、AUDIO_X1、AUDIO_CLK、USB_X1 时钟输入的下降时间	t_{EXf}	—	4	ns	
VIDEO_X1 时钟输入的下降时间		—	3		
CKIO 时钟输出频率	f_{OP}	50.00	66.67	MHz	
CKIO 时钟输出的周期时间	t_{cyc}	15.00	20.00	ns	52.2(1)、 52.2(2)
CKIO 时钟输出的低电平脉宽 1	t_{CKOL1}	$t_{cyc}/2-t_{CKOr1}$	—	ns	52.2(1)
CKIO 时钟输出的高电平脉宽 1	t_{CKOH1}	$t_{cyc}/2-t_{CKOf1}$	—	ns	
CKIO 时钟输出的上升时间 1	t_{CKOr1}	—	3	ns	
CKIO 时钟输出的下降时间 1	t_{CKOf1}	—	3	ns	
CKIO 时钟输出的低电平脉宽 2	t_{CKOL2}	$t_{cyc}/2-t_{CKOr2}$	—	ns	52.2(2)
CKIO 时钟输出的高电平脉宽 2	t_{CKOH2}	$t_{cyc}/2-t_{CKOf2}$	—	ns	
CKIO 时钟输出的上升时间 2	t_{CKOr2}	—	2	ns	
CKIO 时钟输出的下降时间 2	t_{CKOf2}	—	2	ns	
上电振荡稳定时间	t_{OSC1}	10	—	ms	52.3
待机返回稳定时间 1	t_{OSC2}	10	—	ms	52.4
待机返回稳定时间 2	t_{OSC3}	10	—	ms	52.5
实时时钟振荡稳定时间	t_{ROSC}	3	—	s	52.6
模式保持时间	t_{MDH}	200	—	ns	52.3、52.4

【注】 * 这是参考值。此时钟精度影响数字视频解码器的输出图像质量，必须尽量输入精度高的时钟。

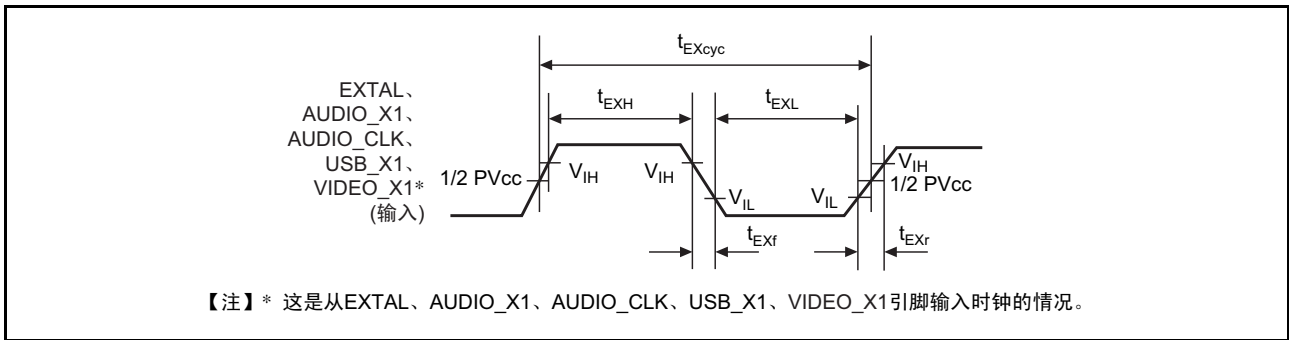


图 52.1 EXTAL、AUDIO_X1、AUDIO_CLK、USB_X1、VIDEO_X1 时钟输入时序

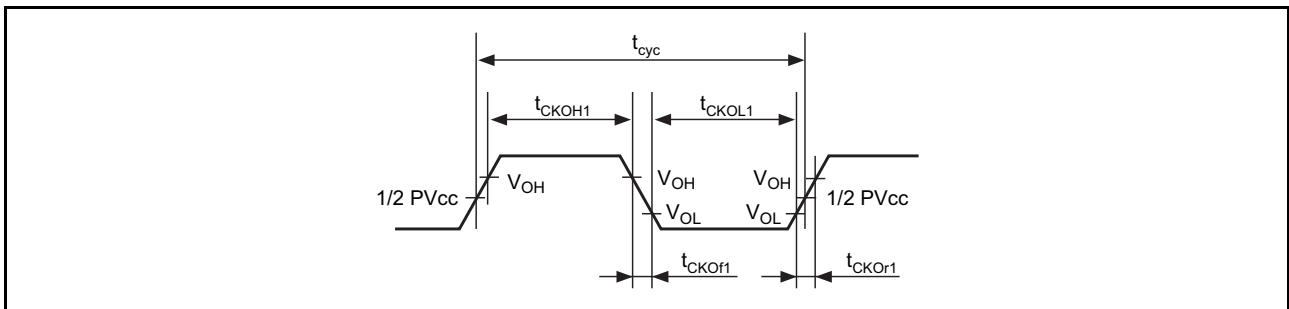


图 52.2(1) CKIO 时钟输出时序 1

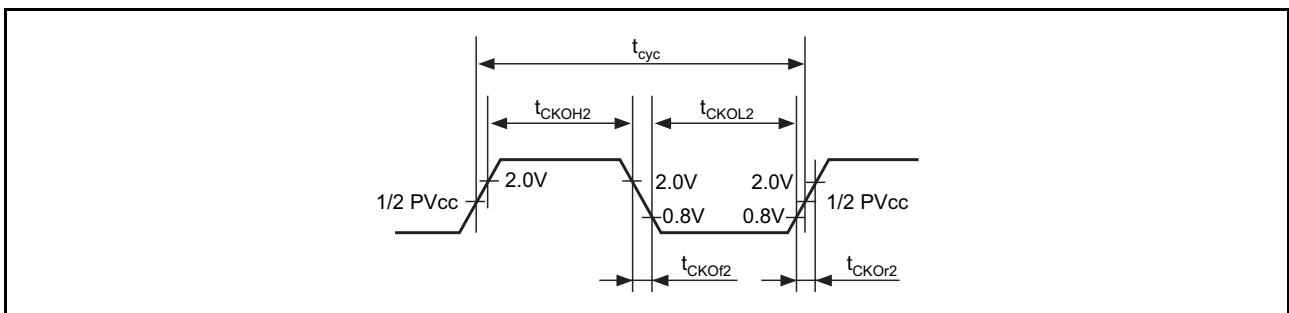


图 52.2(2) CKIO 时钟输出时序 2(2)

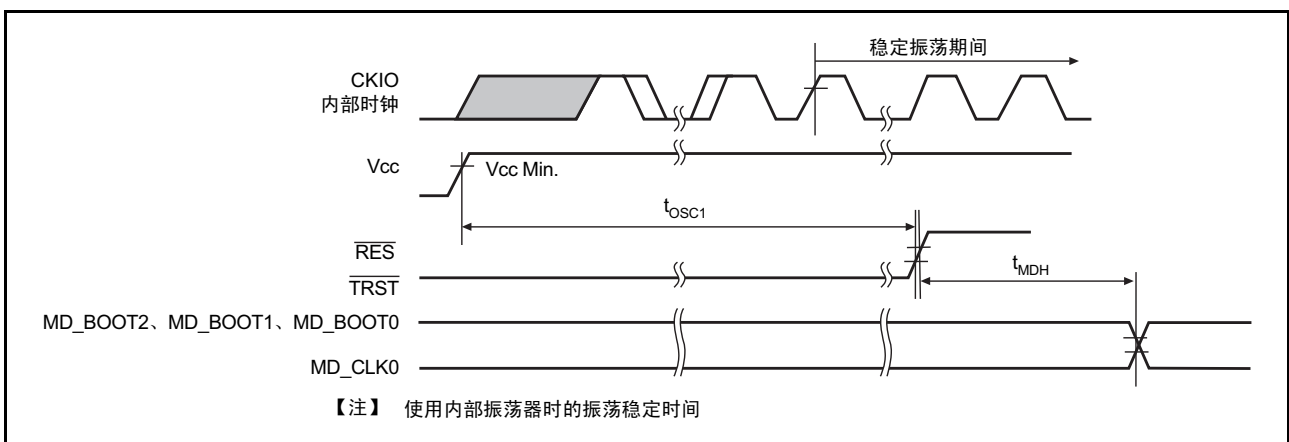


图 52.3 上电振荡稳定时间

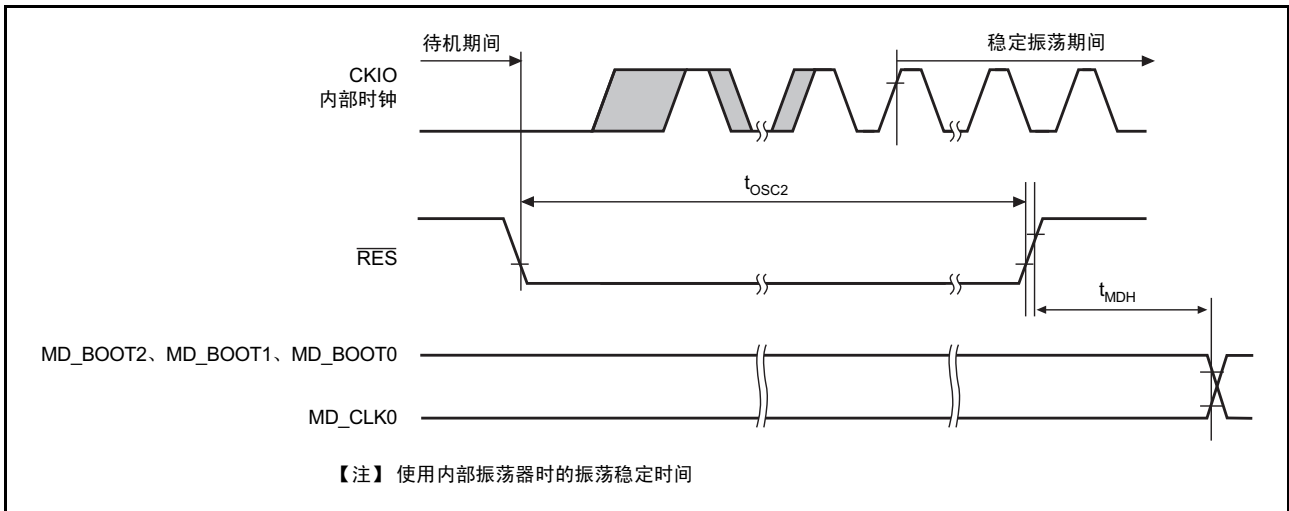


图 52.4 待机返回时的振荡稳定时间（通过复位返回）

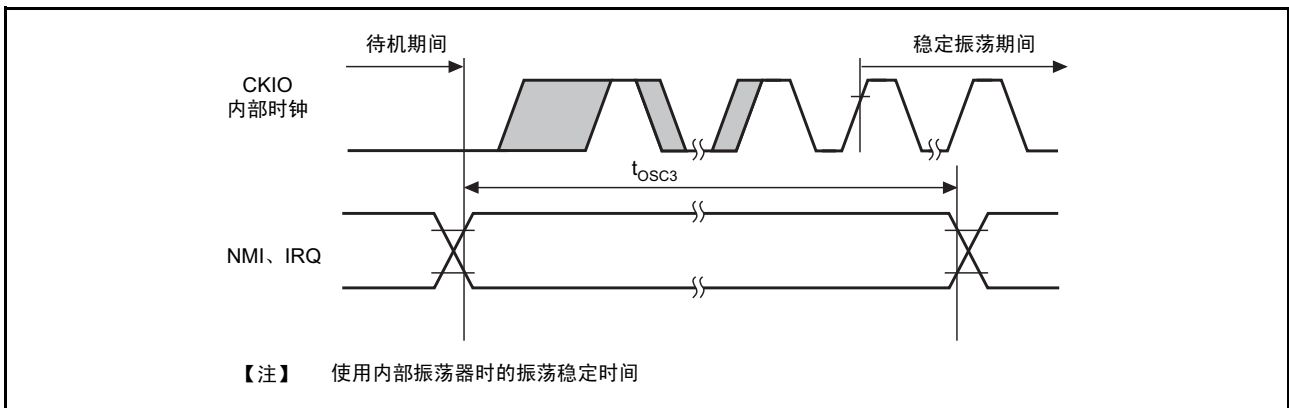


图 52.5 待机返回时的振荡稳定时间（通过 NMI、IRQ 返回）

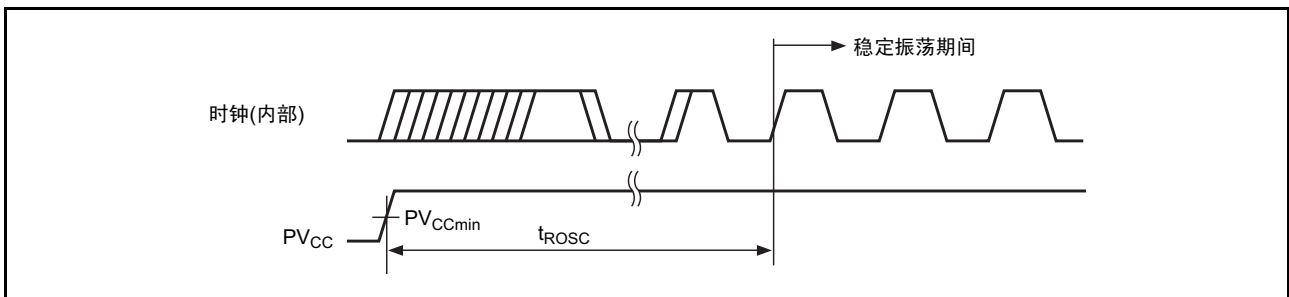


图 52.6 实时时钟振荡稳定时间

52.4.2 控制信号时序

表 52.6 控制信号时序

项目		符号	Min.	Max.	单位	参照图
RES 脉宽	解除待机模式时	t_{RESW}	10	—	ms	52.7(1)
	上述以外		20	—	t_{cyc}	
TRST 脉宽		t_{TRSW}	20	—	t_{cyc}	52.7(2)
NMI 脉宽		t_{NMIW}	20	—	t_{cyc}	
IRQ 脉宽		t_{IRQW}	20	—	t_{cyc}	
PINT 脉宽		t_{PINTW}	20	—	t_{cyc}	
BREQ 准备时间		t_{BREQS}	$1/2t_{cyc}+7$	—	ns	52.8
BREQ 保持时间		t_{BREQH}	$1/2t_{cyc}+2$	—	ns	
BACK 延迟时间		t_{BACKD}	—	$1/2t_{cyc}+13$	ns	
总线缓冲器 OFF 时间 1		t_{BOFF1}	—	15	ns	
总线缓冲器 OFF 时间 2		t_{BOFF2}	—	15	ns	
总线缓冲器 ON 时间 1		t_{BON1}	—	15	ns	
总线缓冲器 ON 时间 2		t_{BON2}	—	15	ns	
对总线缓冲器的 BACK 准备时间		t_{BACKS}	0	—	ns	

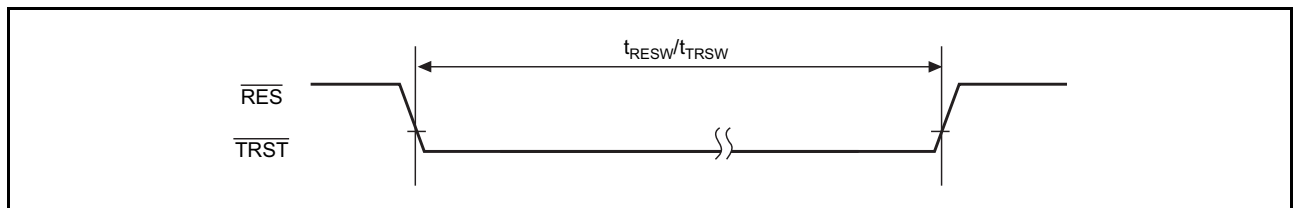


图 52.7(1) 复位的输入时序

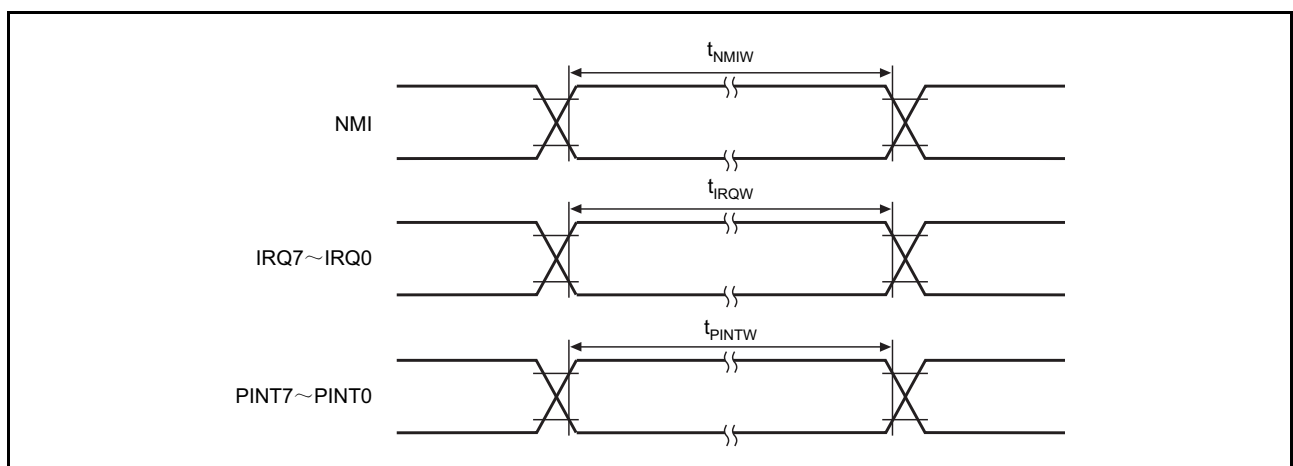


图 52.7(2) 中断信号的输入时序

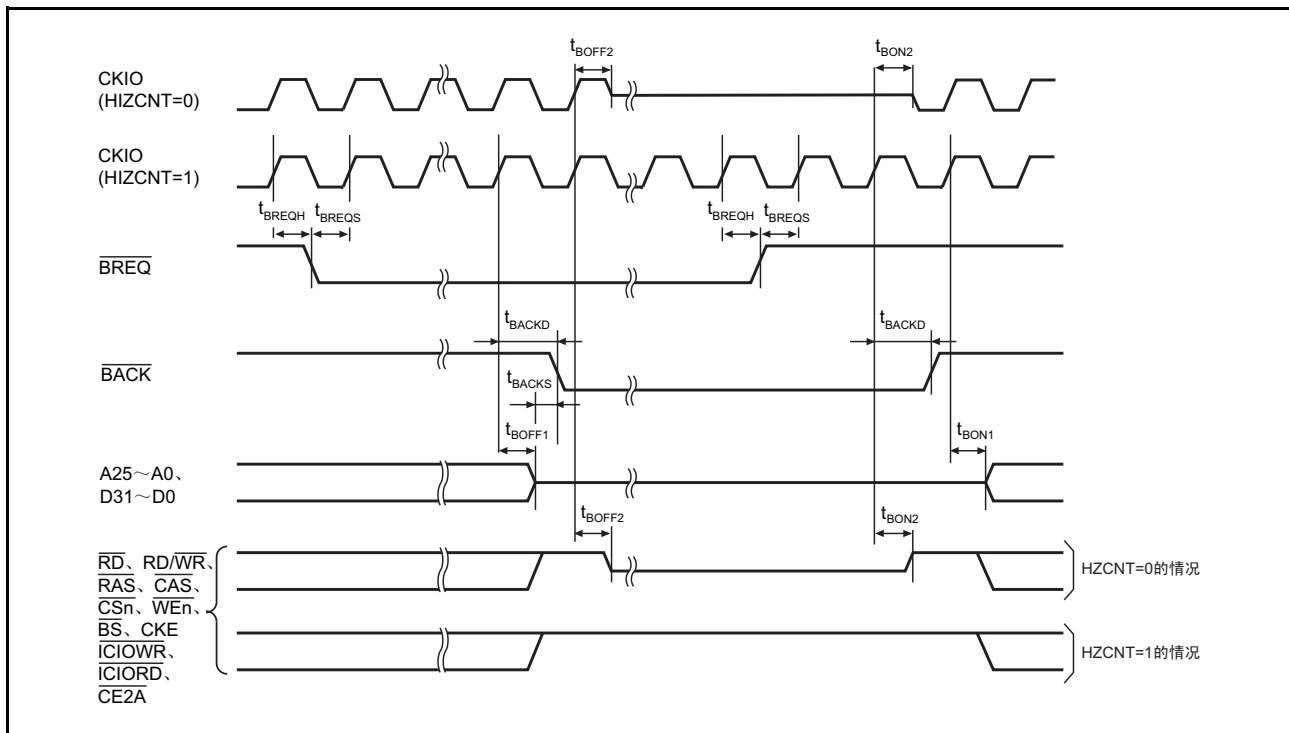


图 52.8 总线权的释放时序

52.4.3 总线时序

表 52.7 总线时序

项目	符号	CKIO=66.67MHz*1		单位	参照图
		Min.	Max.		
地址延迟时间 1	t_{AD1}	0/2*3	12	ns	52.9 ~ 52.33、52.34 ~ 52.37
地址延迟时间 2	t_{AD2}	$1/2t_{cyc}$	$1/2t_{cyc}+12$	ns	52.16
地址准备时间	t_{AS}	0	—	ns	52.9 ~ 52.12、52.16
芯片允许的准备时间	t_{CS}	0	—	ns	52.9 ~ 52.12、52.16
地址保持时间	t_{AH}	0	—	ns	52.9 ~ 52.12
BS 延迟时间	t_{BSD}	—	12	ns	52.9 ~ 52.30、 52.34 ~ 52.37
CS 延迟时间 1	t_{CSD1}	0/2*3	12	ns	52.9 ~ 52.33、52.34 ~ 52.37
读写延迟时间 1	t_{RWD1}	0/2*3	12	ns	52.9 ~ 52.33、52.34 ~ 52.37
读选通的延迟时间	t_{RSD}	$1/2t_{cyc}$	$1/2t_{cyc}+12$	ns	52.9 ~ 52.16、52.34、52.35
读数据的准备时间 1	t_{RDS1}	$1/2t_{cyc}+5$	—	ns	52.9 ~ 52.15、52.34 ~ 52.37
读数据的准备时间 2	t_{RDS2}	7	—	ns	52.17 ~ 52.20、 52.25 ~ 52.27
读数据的准备时间 3	t_{RDS3}	$1/2t_{cyc}+5$	—	ns	52.16
读数据的保持时间 1	t_{RDH1}	0	—	ns	52.9 ~ 52.15、52.34 ~ 52.37
读数据的保持时间 2	t_{RDH2}	2	—	ns	52.17 ~ 52.20、 52.25 ~ 52.27
读数据的保持时间 3	t_{RDH3}	0	—	ns	52.16
写允许的延迟时间 1	t_{WED1}	$1/2t_{cyc}$	$1/2t_{cyc}+12$	ns	52.9 ~ 52.14、52.34、52.35
写允许的延迟时间 2	t_{WED2}	—	12	ns	52.15
写数据的延迟时间 1	t_{WDD1}	—	12	ns	52.9 ~ 52.15、52.34 ~ 52.37
写数据的延迟时间 2	t_{WDD2}	—	12	ns	52.21 ~ 52.24、 52.28 ~ 52.30
写数据的保持时间 1	t_{WDH1}	1	—	ns	52.9 ~ 52.15、52.34 ~ 52.37
写数据的保持时间 2	t_{WDH2}	2	—	ns	52.21 ~ 52.24、 52.28 ~ 52.30
写数据的保持时间 4	t_{WDH4}	0	—	ns	52.9 ~ 52.13、52.34、52.36
WAIT 准备时间	t_{WTS}	$1/2t_{cyc}+4.5$	—	ns	52.10 ~ 52.16、52.35、52.37
WAIT 保持时间	t_{WTH}	$1/2t_{cyc}+3.5$	—	ns	52.10 ~ 52.16、52.35、52.37
IOIS16 准备时间	t_{IO16S}	$1/2t_{cyc}+4.5$	—	ns	52.37
IOIS16 保持时间	t_{IO16H}	$1/2t_{cyc}+3.5$	—	ns	52.37
RAS 延迟时间 1	t_{RASD1}	2	12	ns	52.17 ~ 52.33
CAS 延迟时间 1	t_{CASD1}	2	12	ns	52.17 ~ 52.33
DQM 延迟时间 1	t_{DQMD1}	2	12	ns	52.17 ~ 52.30
CKE 延迟时间 1	t_{CKED1}	2	12	ns	52.32
AH 延迟时间	t_{AHD}	$1/2t_{cyc}$	$1/2t_{cyc}+12$	ns	52.13
多路地址延迟时间	t_{MAD}	—	12	ns	52.13
多路地址保持时间	t_{MAH}	1	—	ns	52.13

项目	符号	CKIO=66.67MHz*1		单位	参照图
		Min.	Max.		
对 AH 的地址准备时间	t_{AVVH}	$1/2t_{cyc}-2$	—	ns	52.13
DACK、TEND 延迟时间	t_{DACD}	参照直接存储器存取控制器的时序。		ns	52.9 ~ 52.30、 52.34 ~ 52.37
ICIORD 延迟时间	t_{ICRSD}	—	$1/2t_{cyc}+12$	ns	52.36、52.37
ICIOWR 延迟时间	t_{ICWSD}	—	$1/2t_{cyc}+12$	ns	52.36、52.37

- 【注】 *1 必须根据所使用的系统结构，和等待数一起对 CKIO（外部总线时钟）的 f_{max} 进行探讨。
 *2 延迟时间、准备时间和保持时间的 $1/2t_{cyc}$ 表示从时钟上升开始的 $1/2$ 个周期，是时钟下降的基准。
 *3 这是使用 SDRAM 时的值。

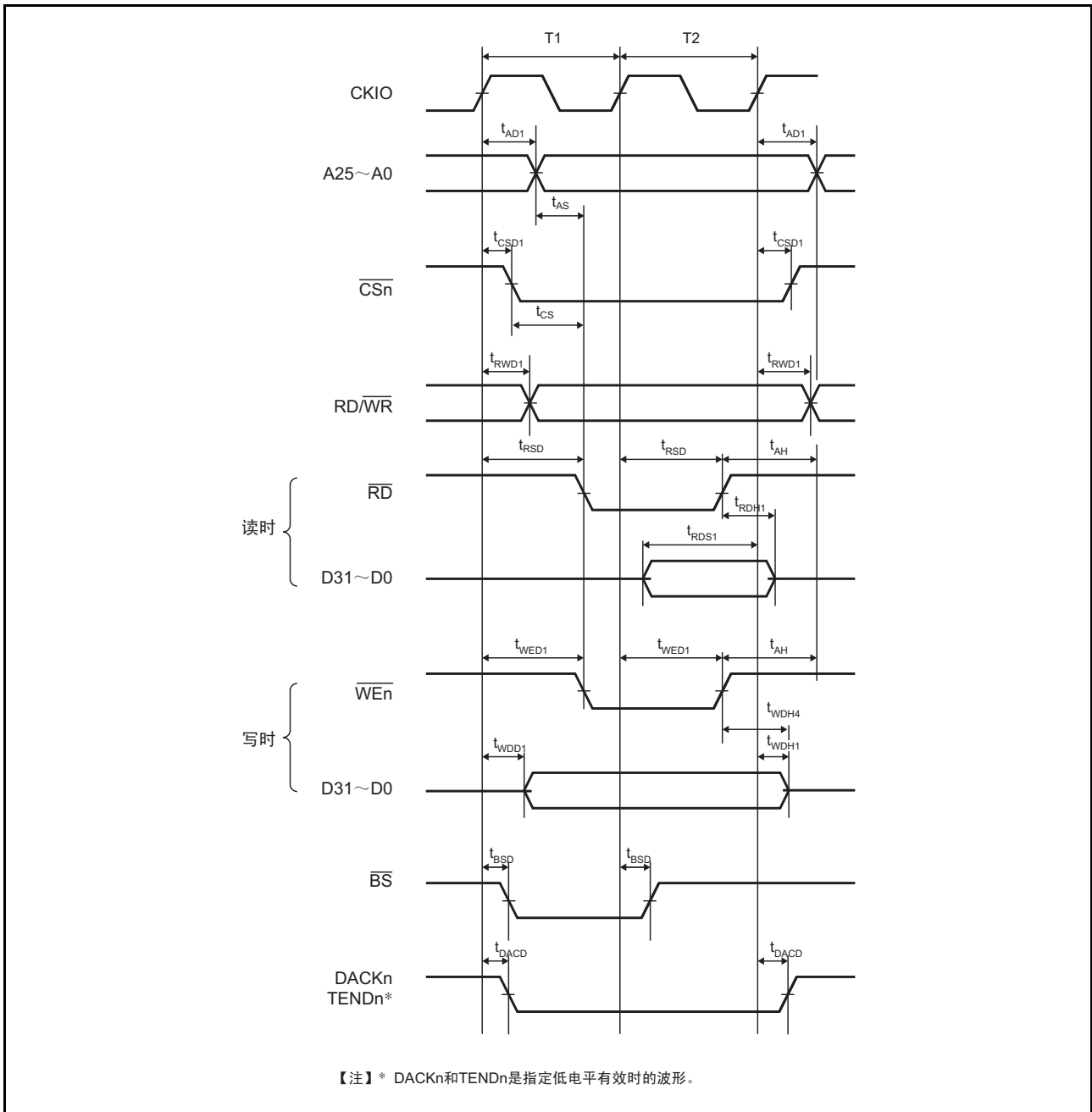


图 52.9 正规空间的基本总线周期（无等待）

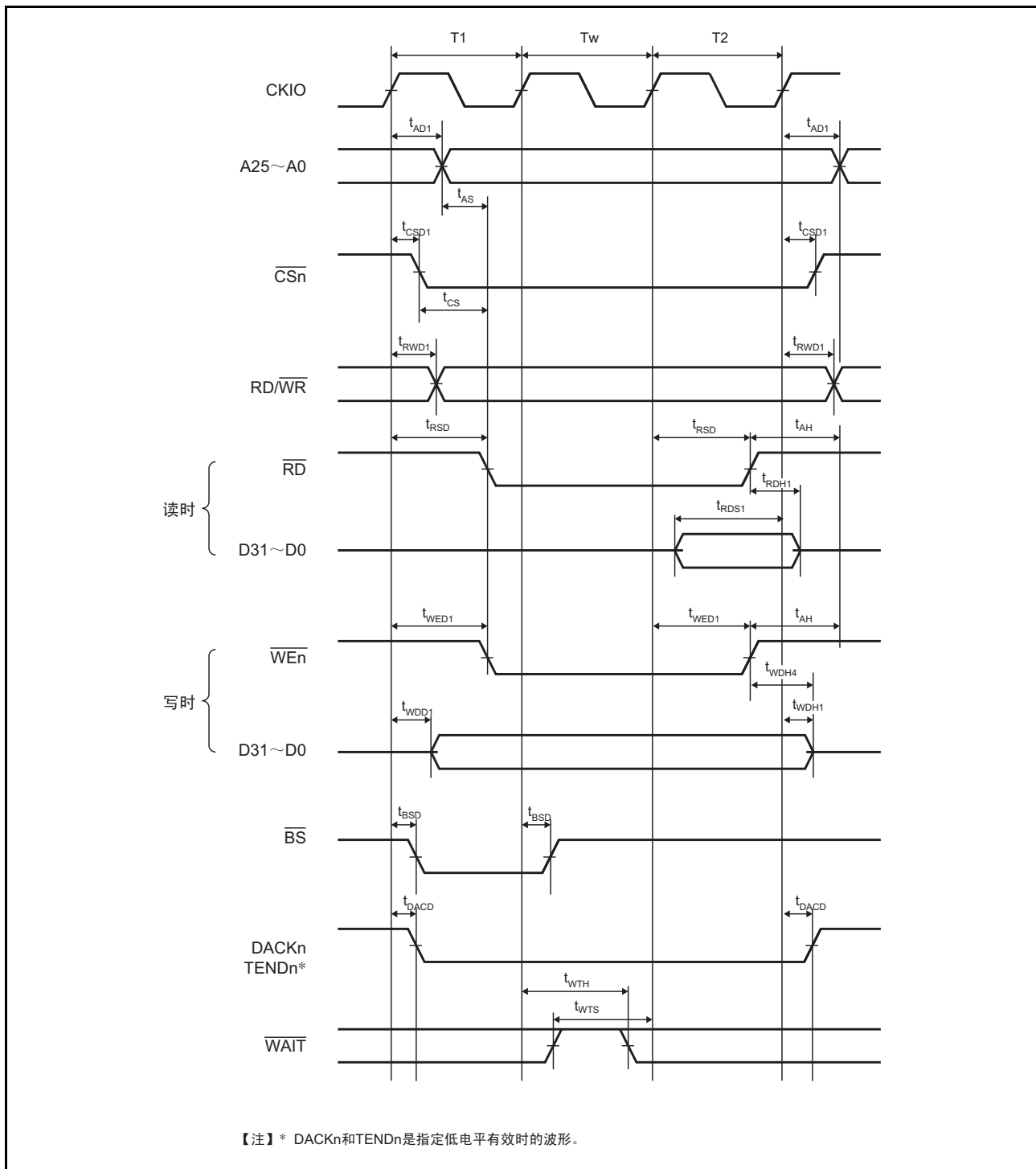


图 52.10 正规空间的基本总线周期（1 个软件等待）

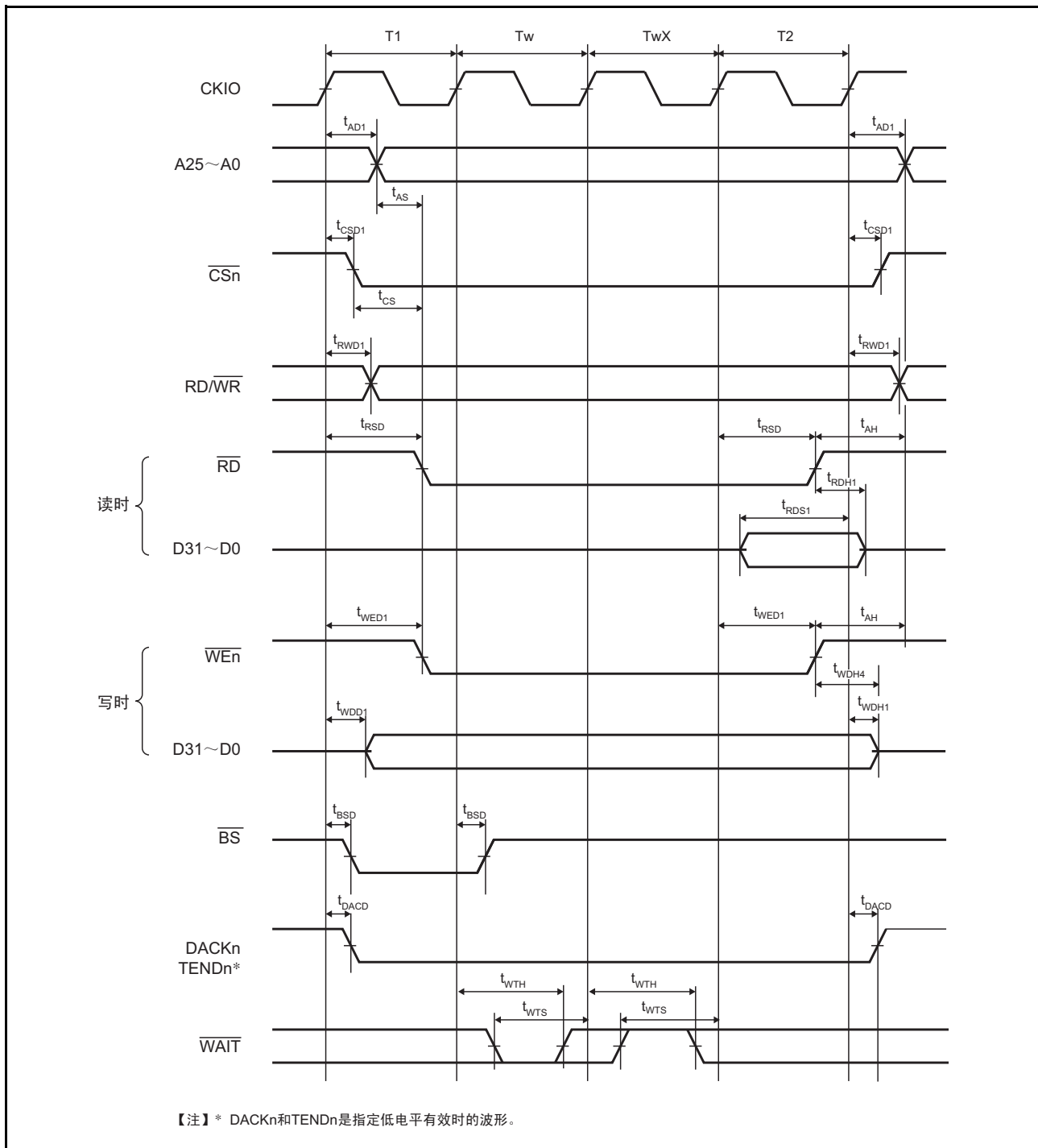


图 52.11 正规空间的基本总线周期（插入 1 个软件等待和 1 个外部等待）

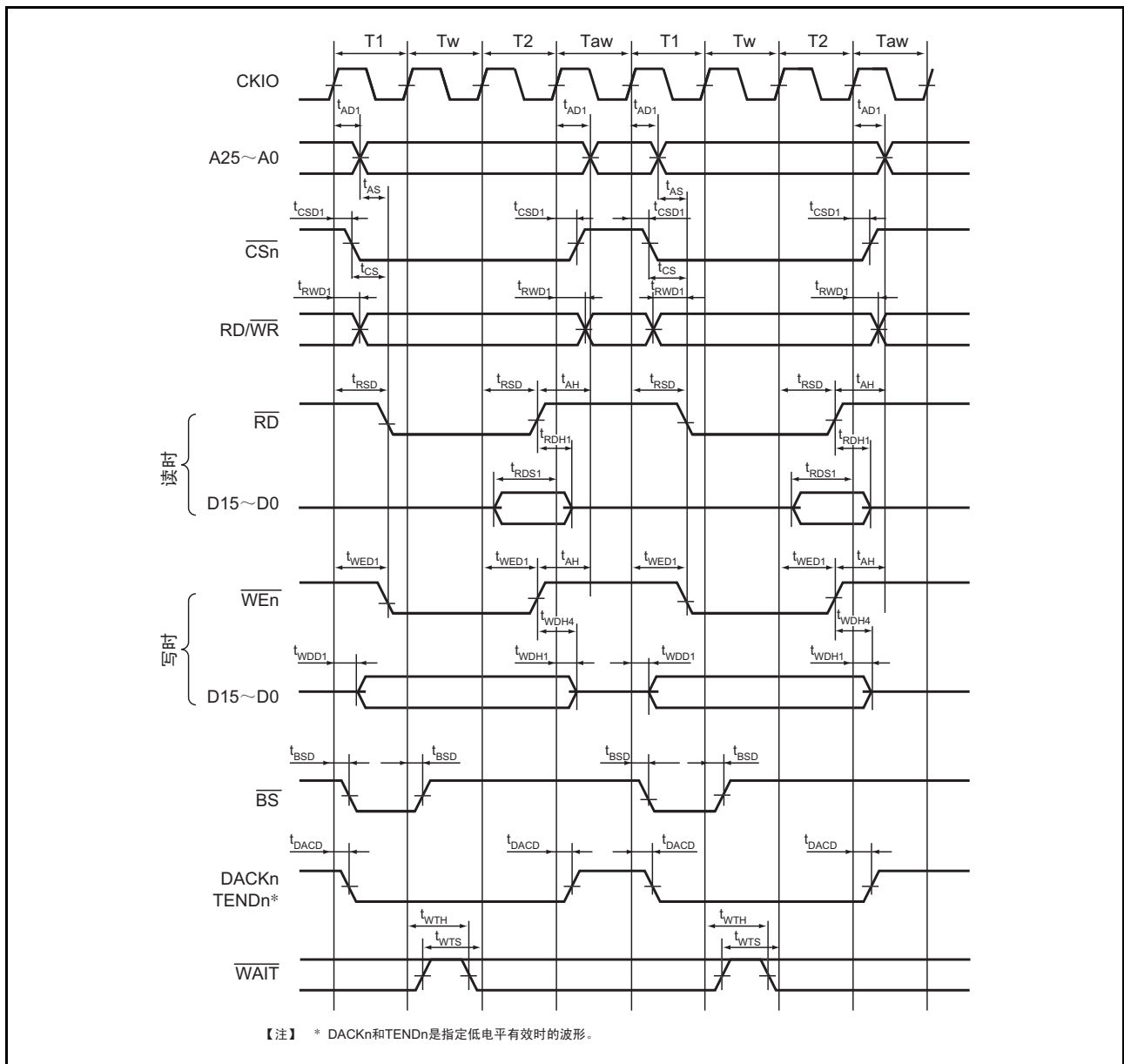


图 52.12 正规空间的基本总线周期（1 个软件等待，外部等待有效（WM 位为“0”），无空闲周期）

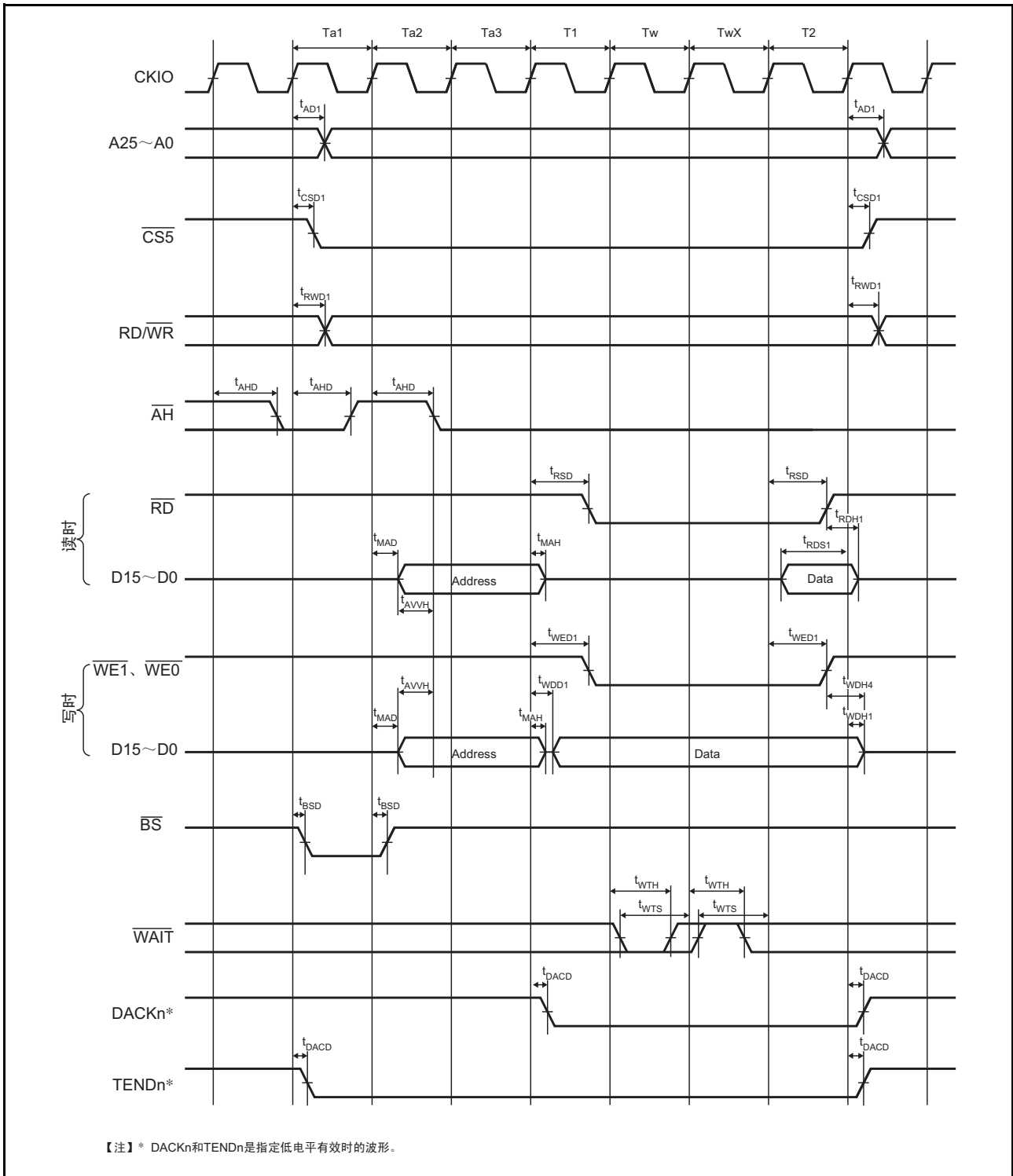


图 52.13 MPX-I/O 接口总线周期 (3 个地址周期, 插入 1 个软件等待和 1 个外部等待)

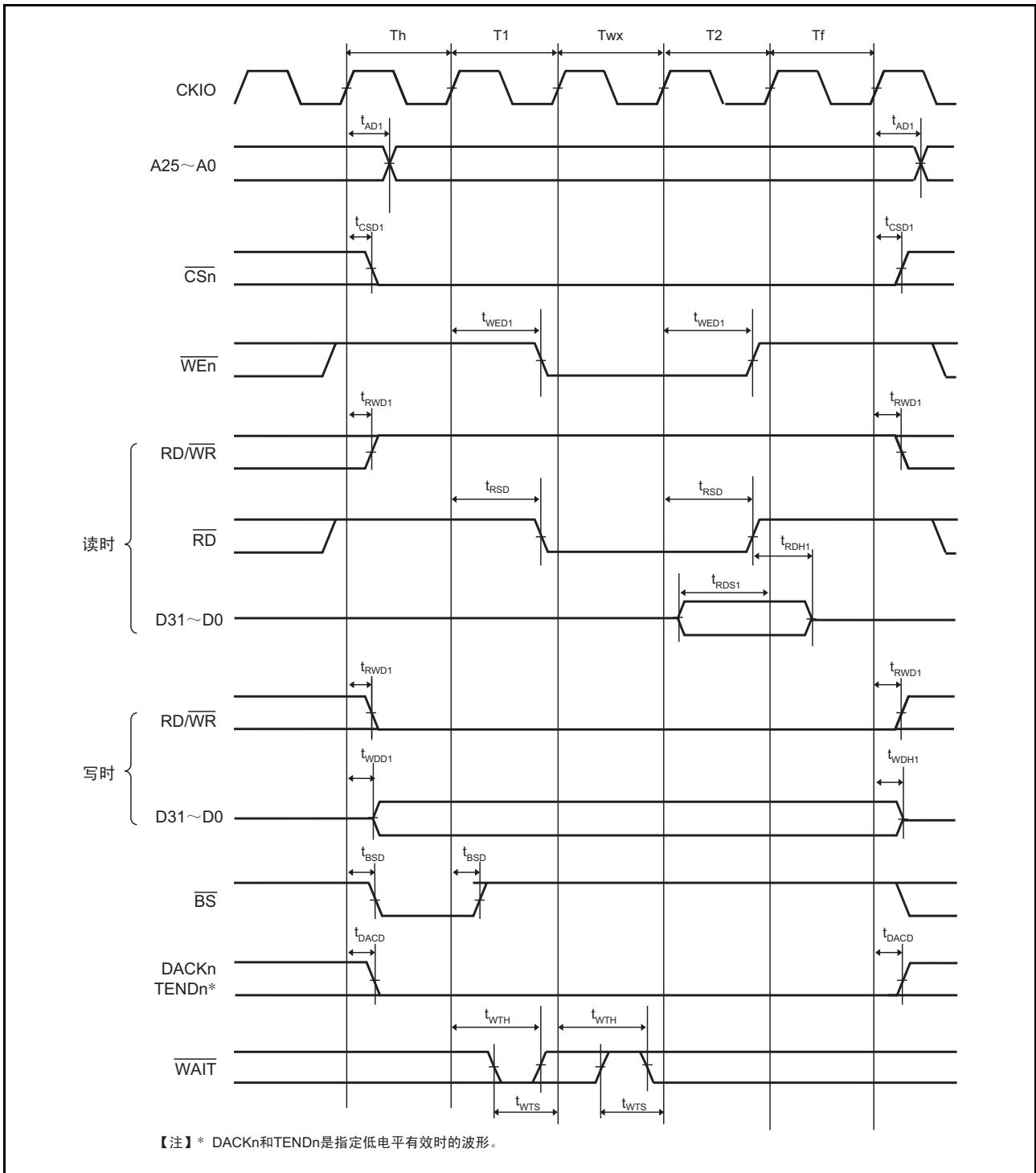


图 52.14 带字节选择的 SRAM 总线周期 (SW=1 个周期, HW=1 个周期, 插入 1 个异步外部等待, BAS=0 (写周期 UB/LB 控制))

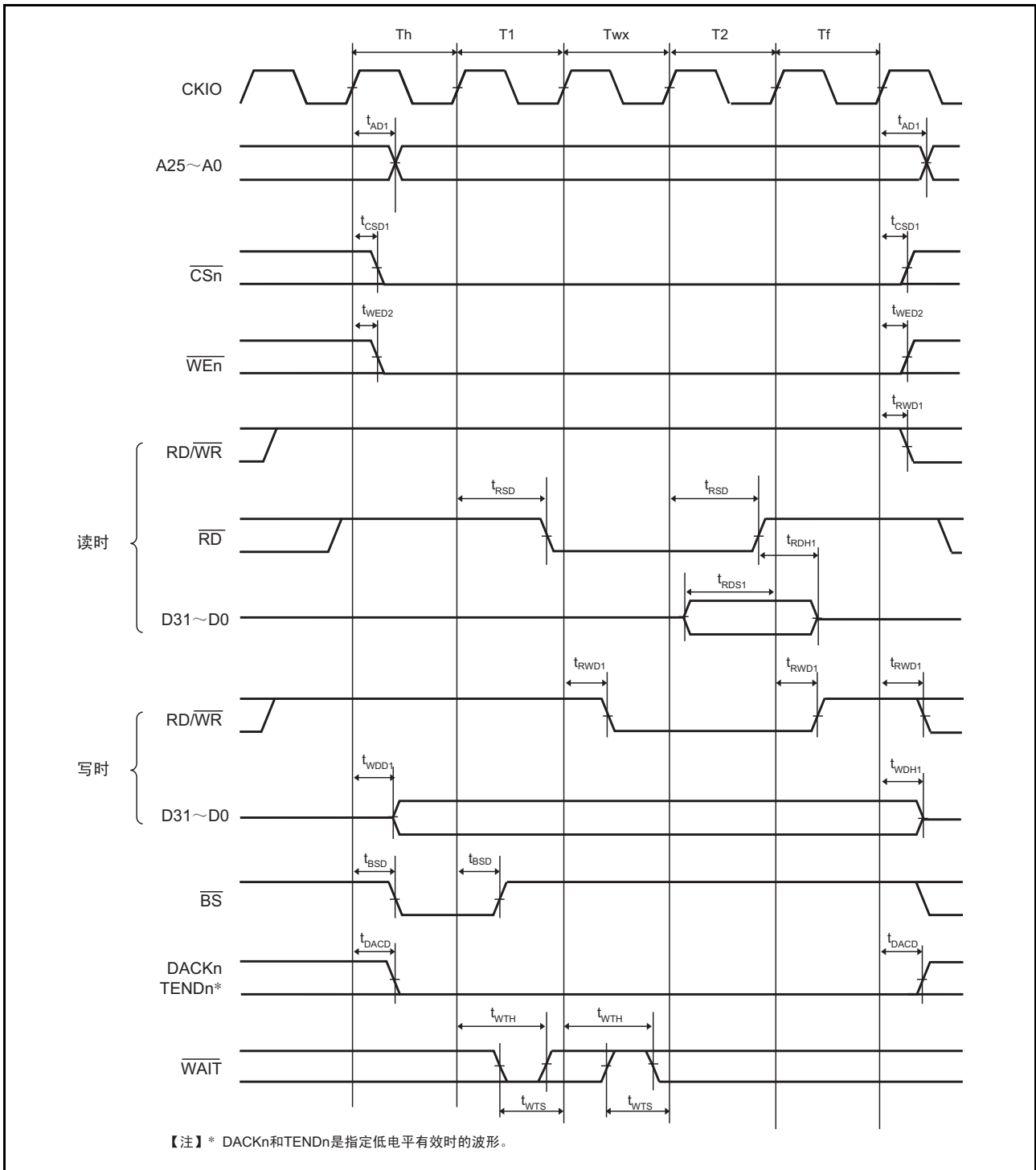


图 52.15 带字节选择的 SRAM 总线周期 (SW=1 个周期, HW=1 个周期, 插入 1 个异步外部等待, BAS=1 (写周期 WE 控制))

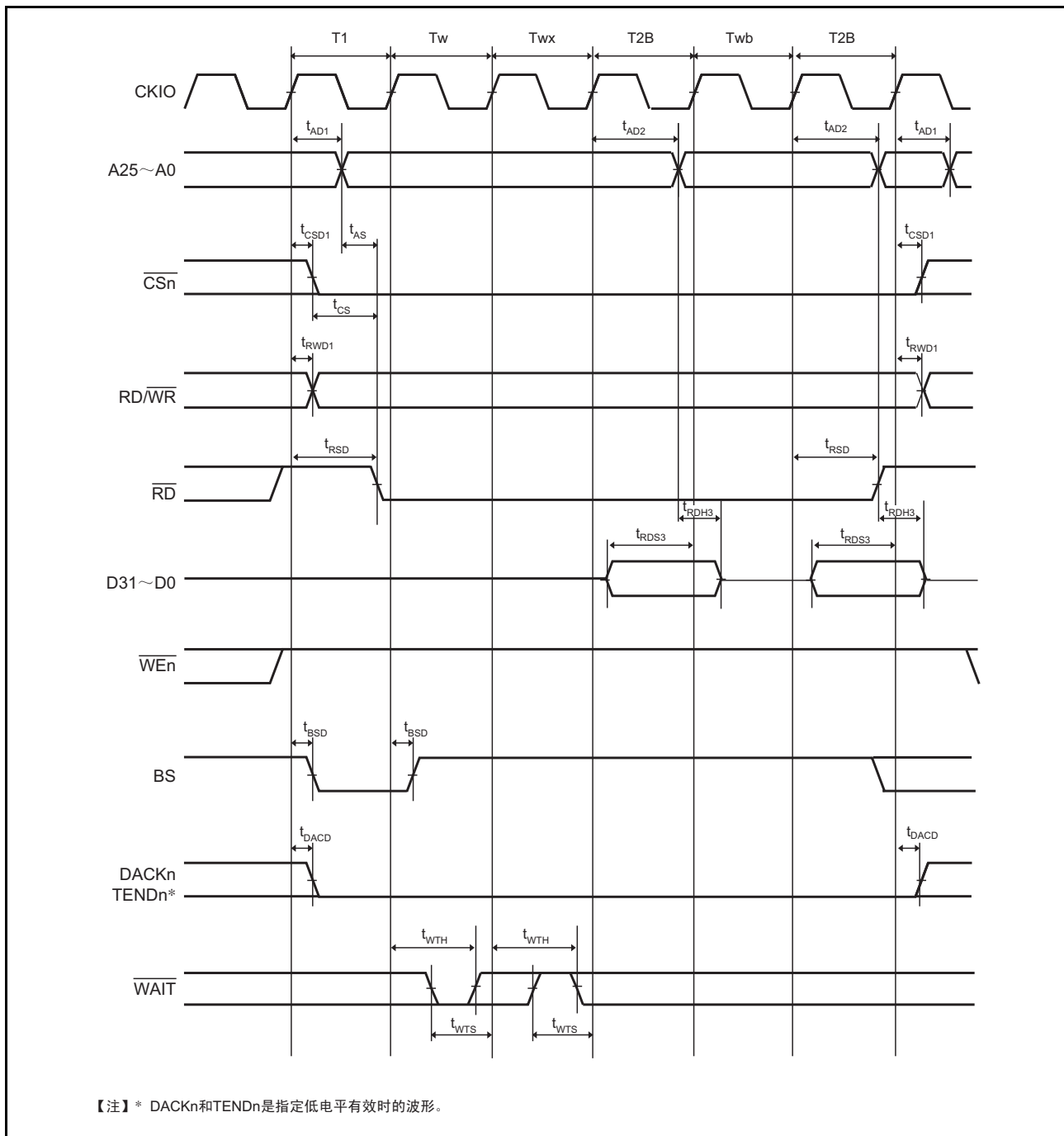


图 52.16 突发 ROM 读周期（插入 1 个软件等待和 1 个异步外部等待，突发等待 1、2 个突发）

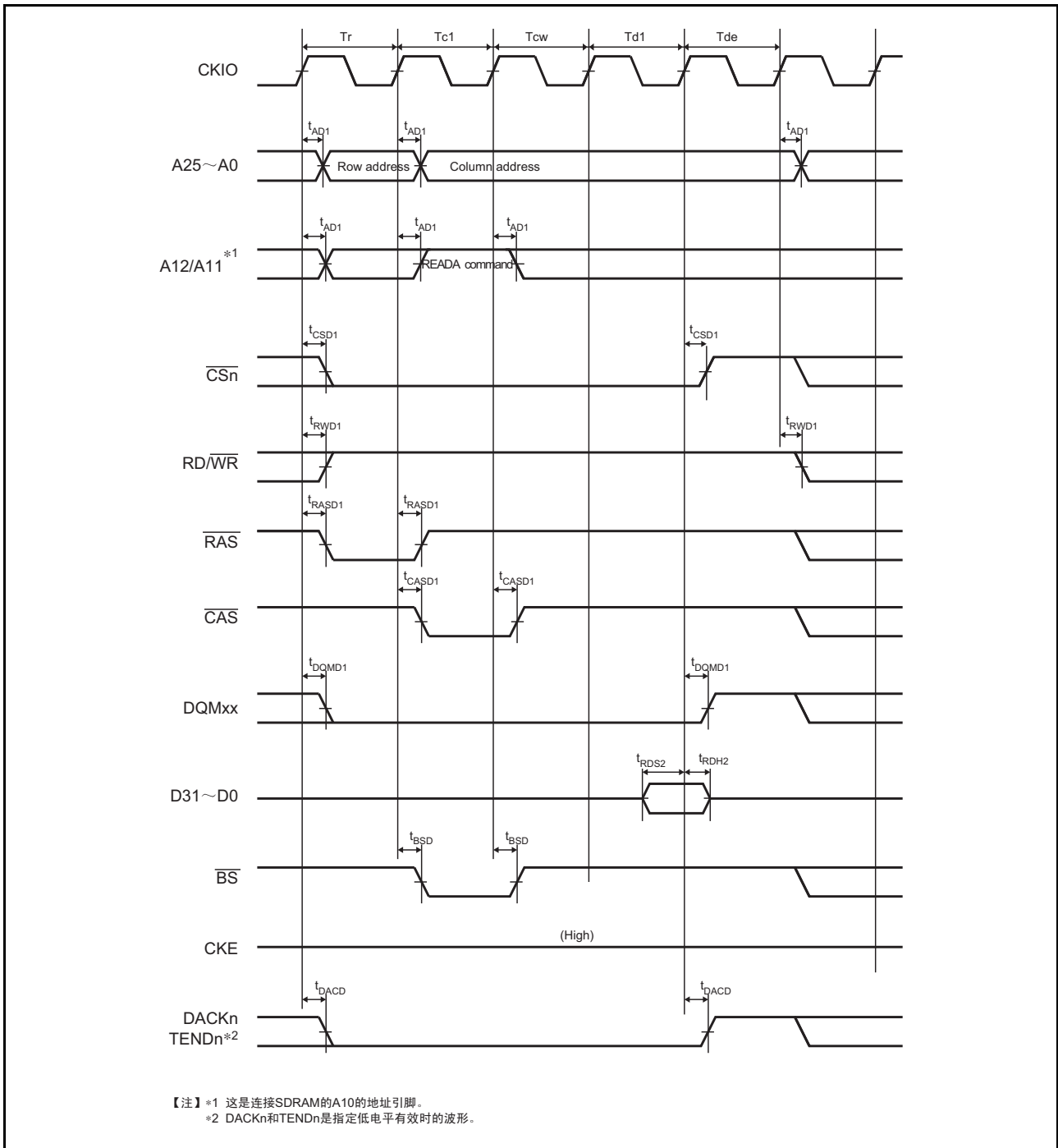


图 52.17 同步 DRAM 单次读的总线周期（有自动预充电，2 个 CAS 等待，WTRCD=0 个周期，WTRP=0 个周期）

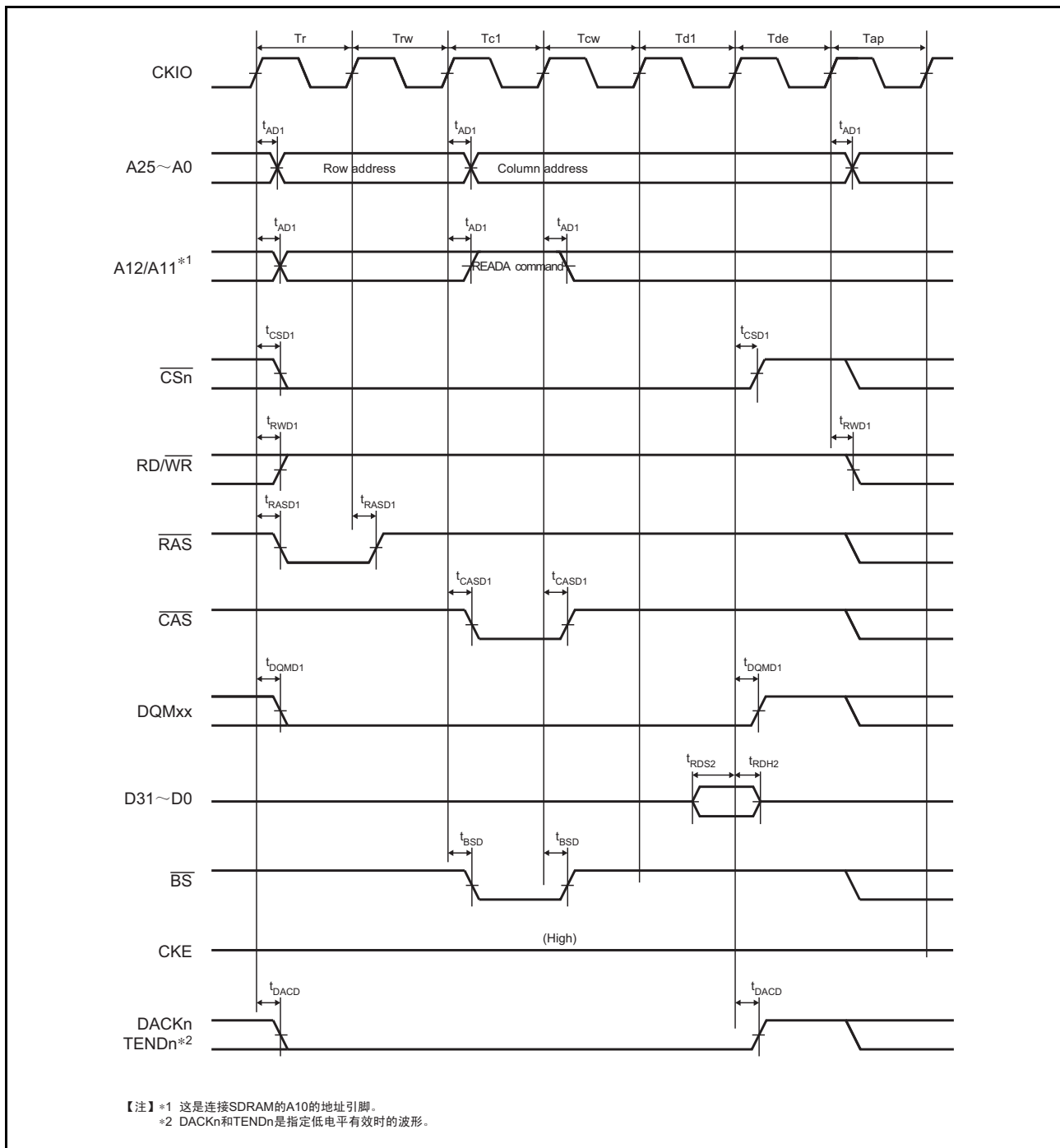


图 52.18 同步 DRAM 单次读的总线周期（有自动预充电，2 个 CAS 等待，WTRCD=1 个周期，WTRP=1 个周期）

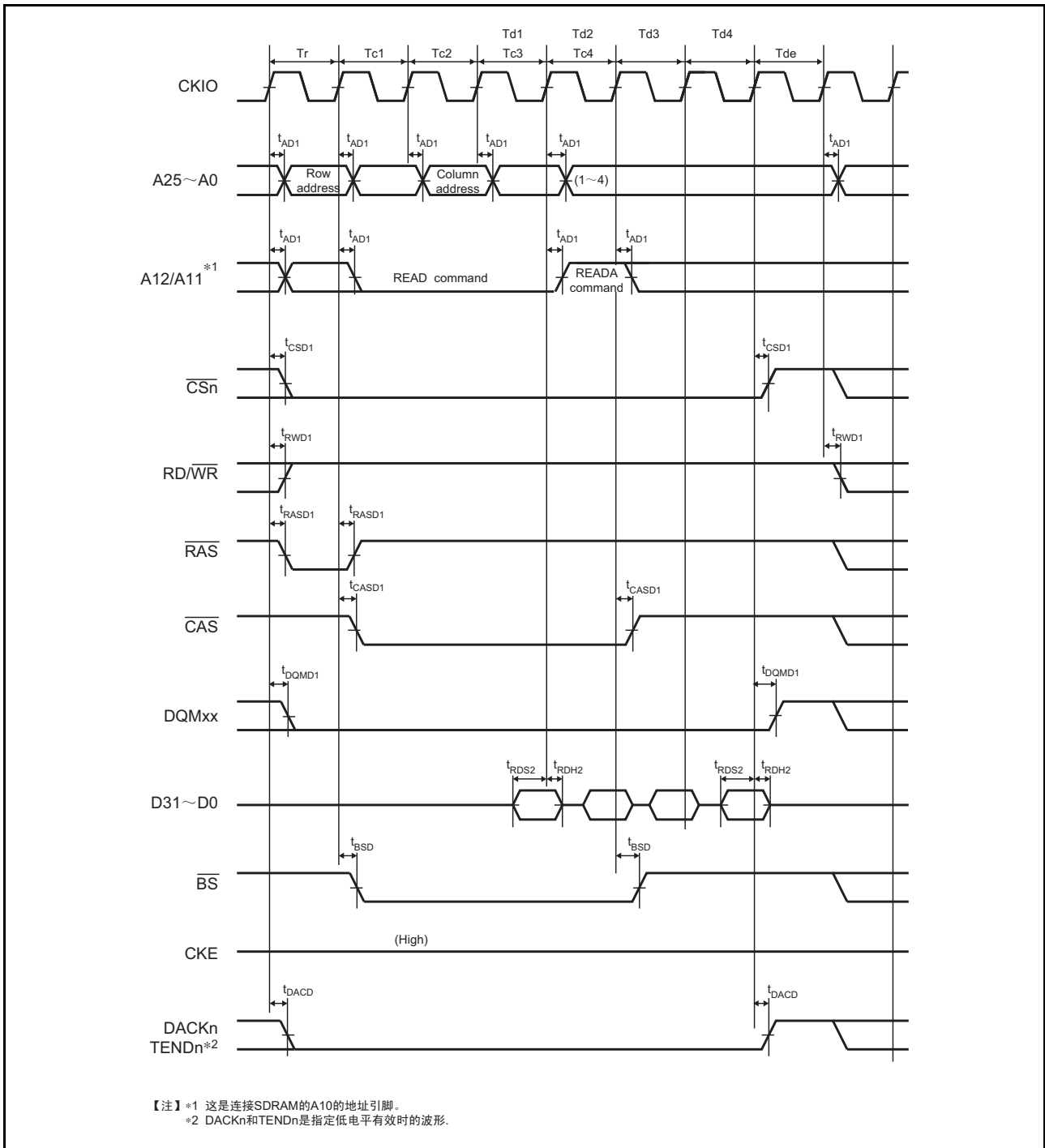


图 52.19 同步 DRAM 突发读的总线周期（4 个读周期）（有自动预充电，2 个 CAS 等待，WTRCD=0 个周期，WTRP=1 个周期）

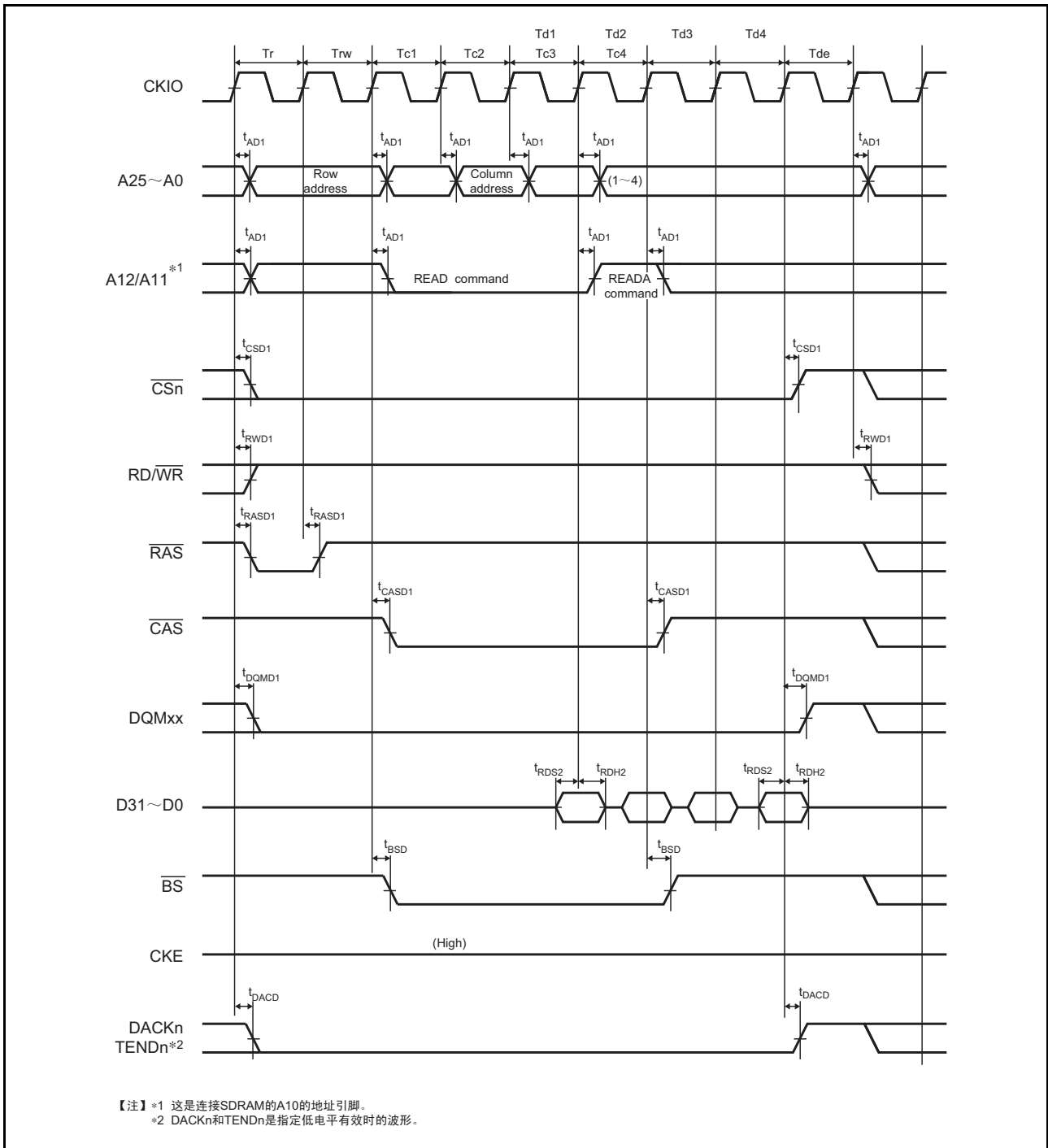


图 52.20 同步 DRAM 突发读的总线周期（4 个读周期）（有自动预充电，2 个 CAS 等待，WTRCD=1 个周期，WTRP=0 个周期）

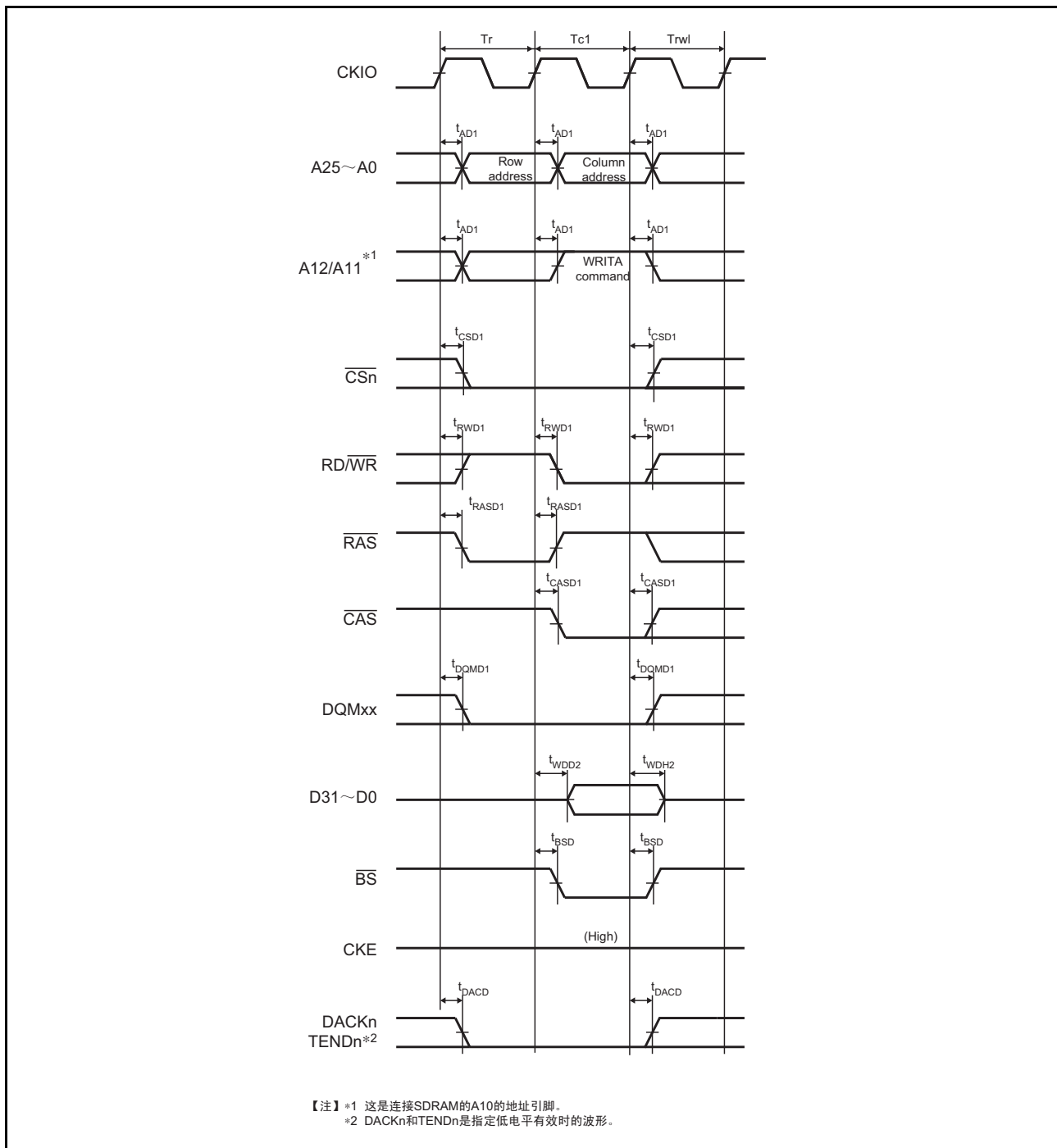


图 52.21 同步 DRAM 单次写的总线周期（有自动预充电，TRWL=1 个周期）

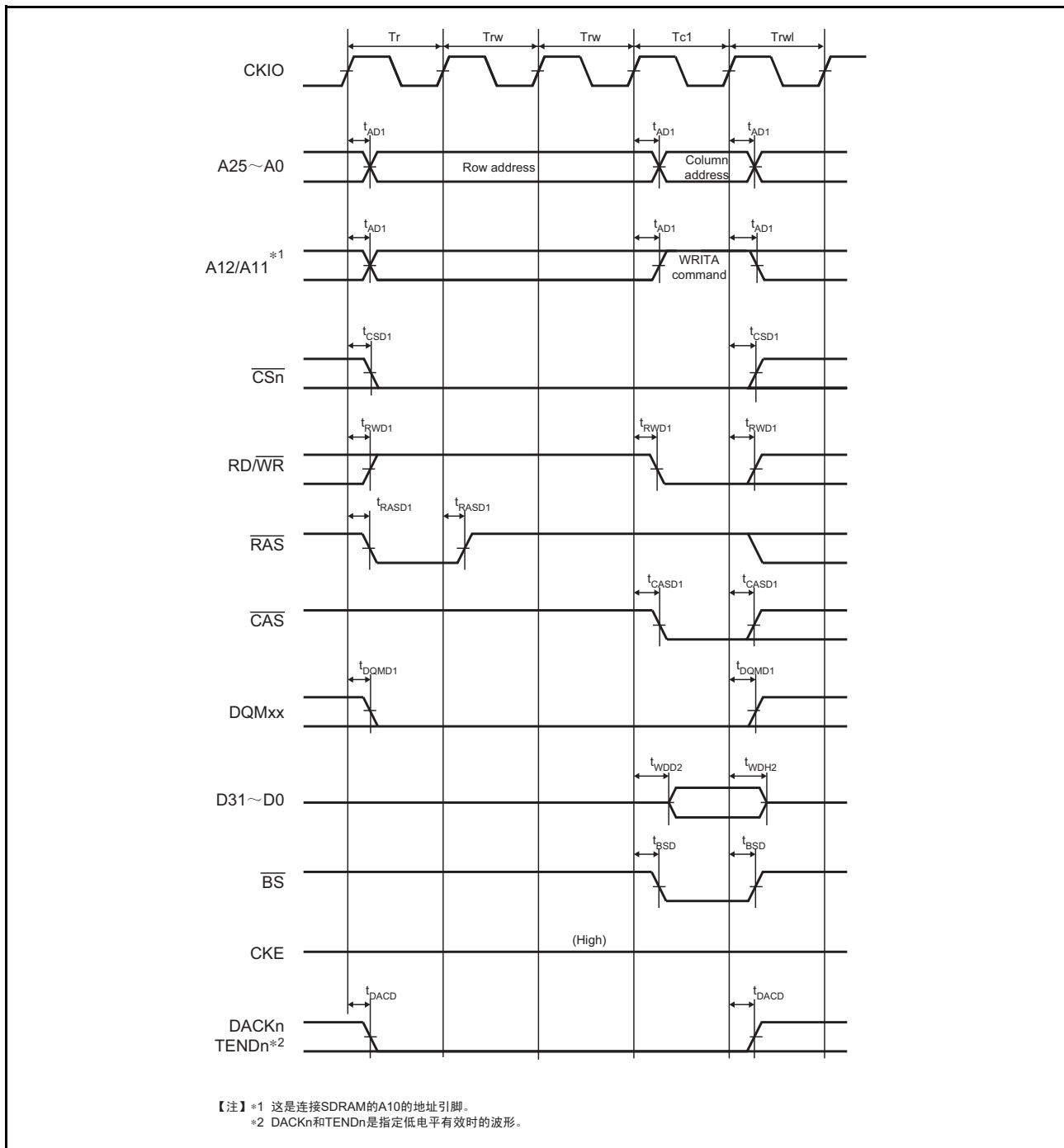


图 52.22 同步 DRAM 单次写的总线周期（有自动预充电，WTRCD=2 个周期，TRWL=1 个周期）

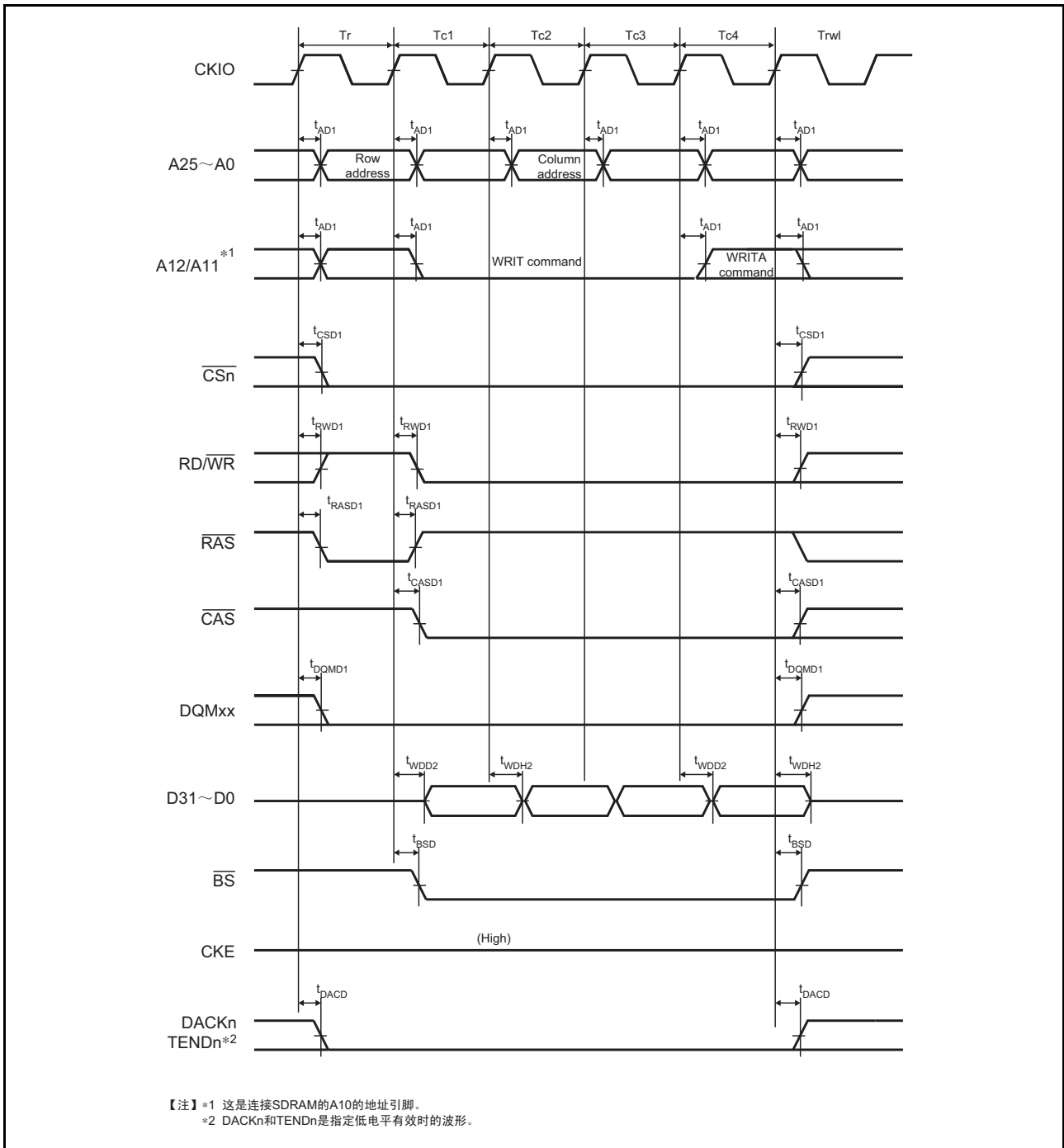


图 52.23 同步 DRAM 突发写的总线周期（4 个写周期）（有自动预充电，WTRCD=0 个周期，TRWL=1 个周期）

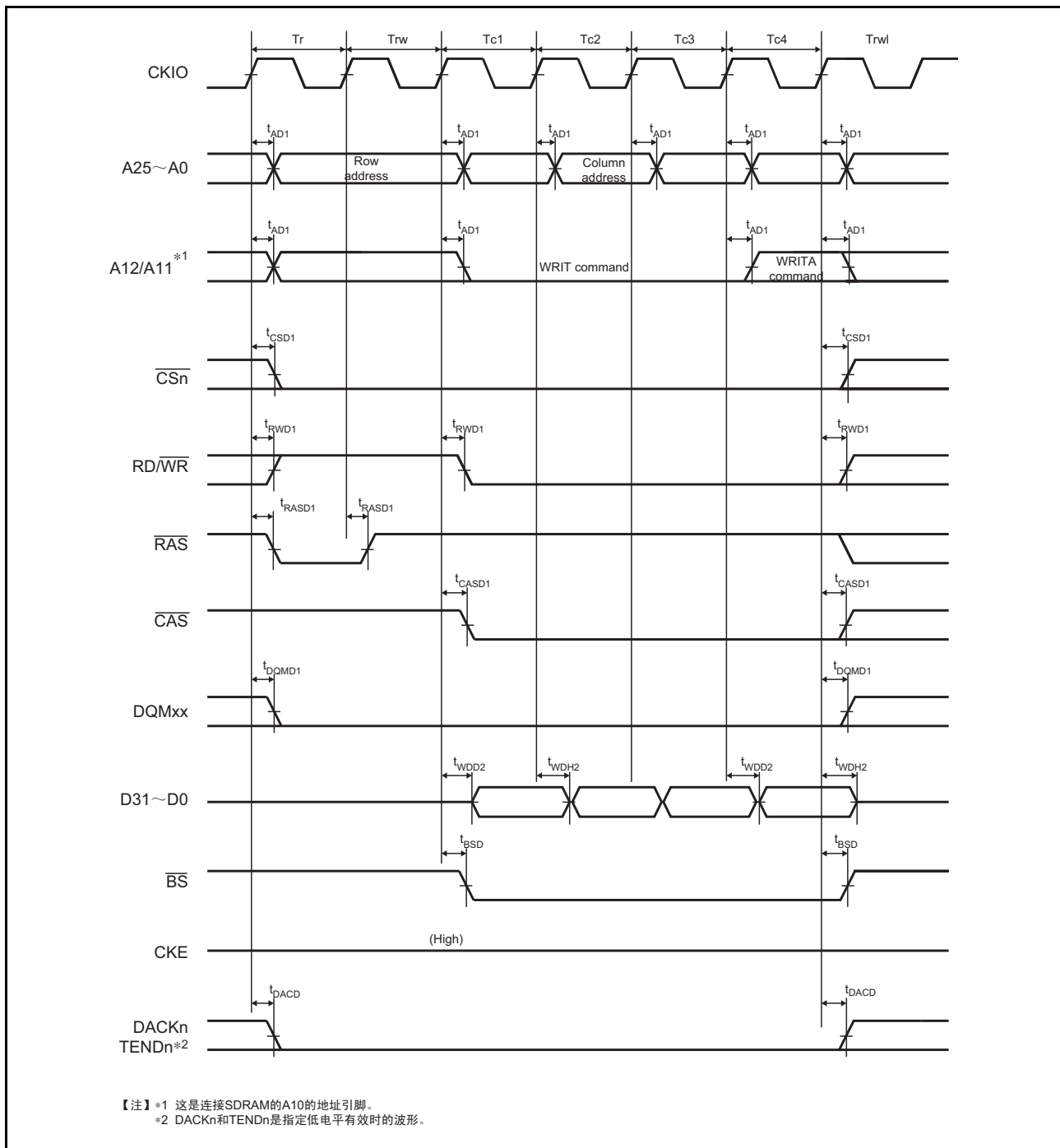


图 52.24 同步 DRAM 突发写的总线周期（4 个写周期）（有自动预充电，WTRCD=1 个周期，TRWL=1 个周期）

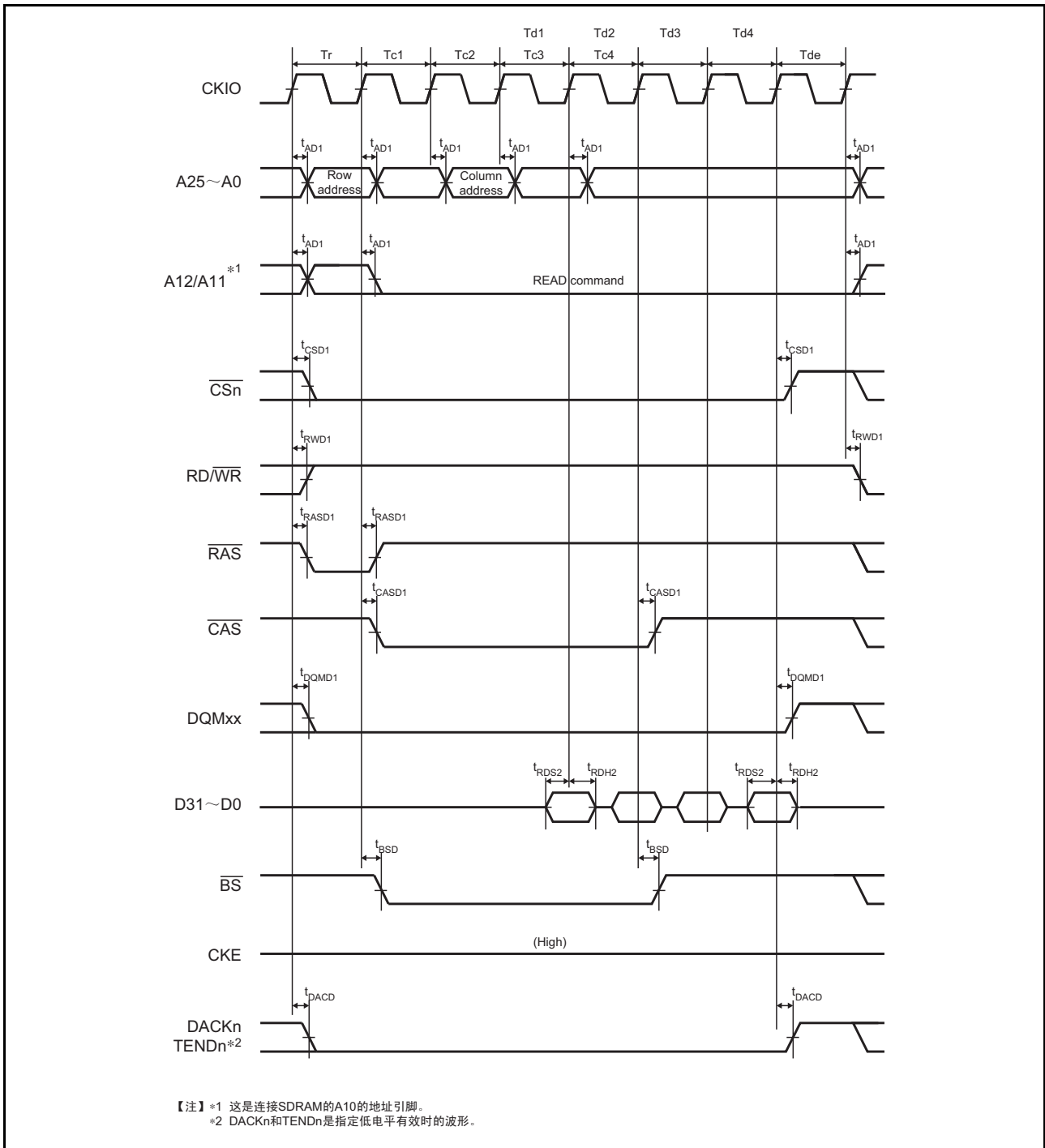


图 52.25 同步 DRAM 突发读的总线周期 (4 个读周期) (存储体激活模式: ACT+READ 命令, 2 个 CAS 等待, WTRCD=0 个周期)

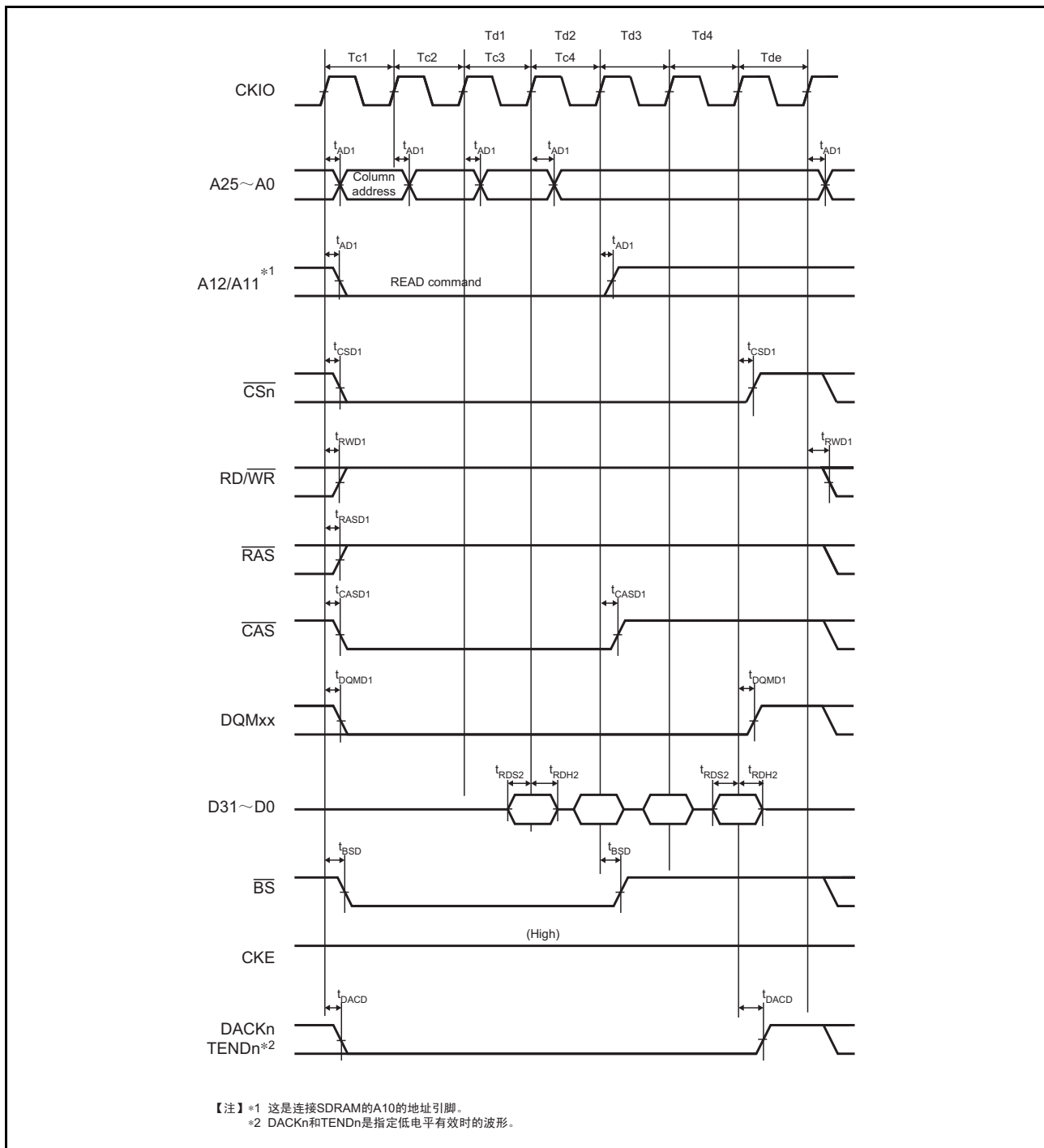


图 52.26 同步 DRAM 突发读的总线周期（4 个读周期）（存储体激活模式：READ 命令，同一行地址，2 个 CAS 等待，WTRCD=0 个周期）

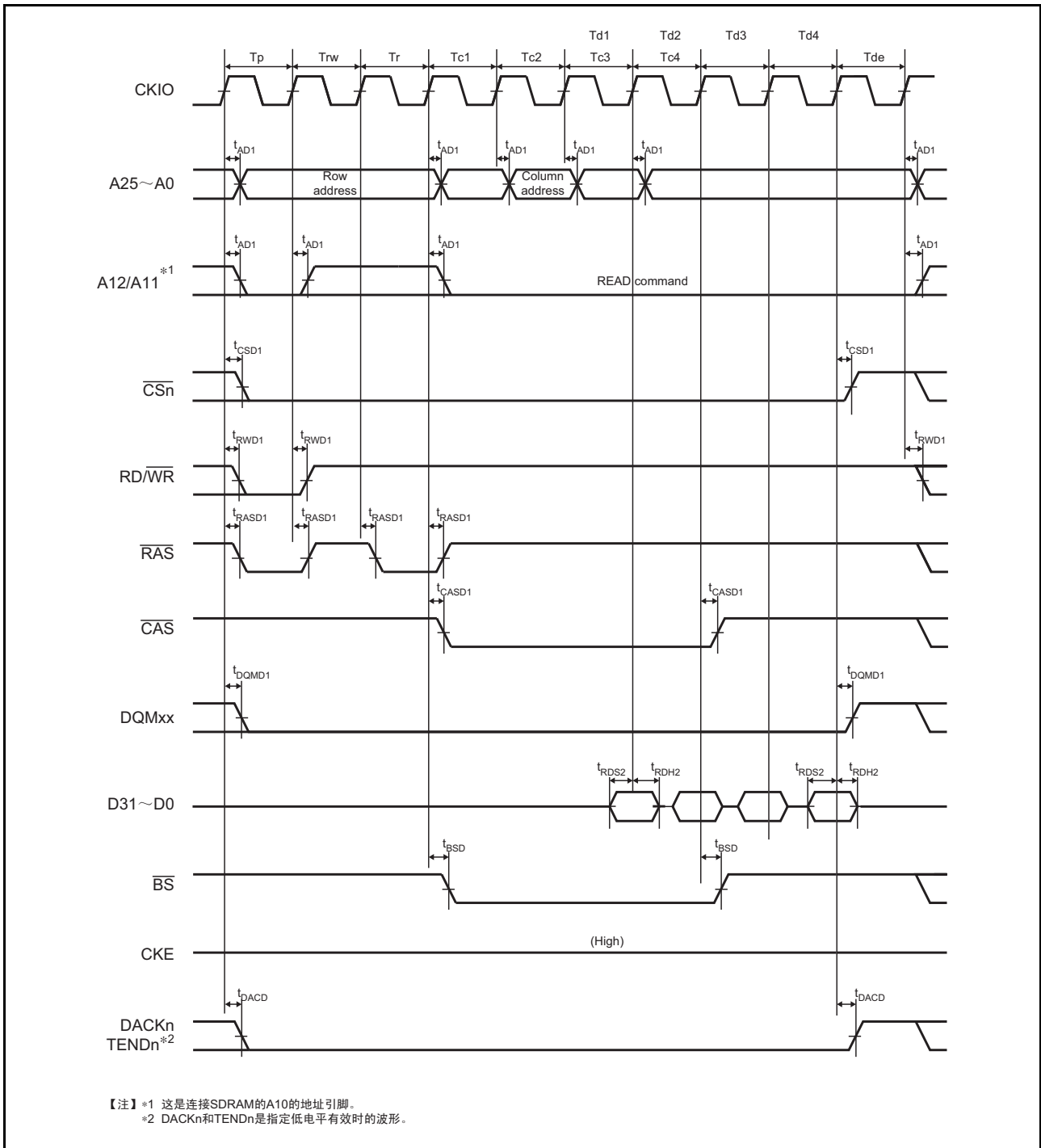


图 52.27 同步 DRAM 突发读的总线周期（4 个读周期）（存储体激活模式：PRE+ACT+READ 命令，不同行地址，2 个 CAS 等待，WTRCD=0 个周期）

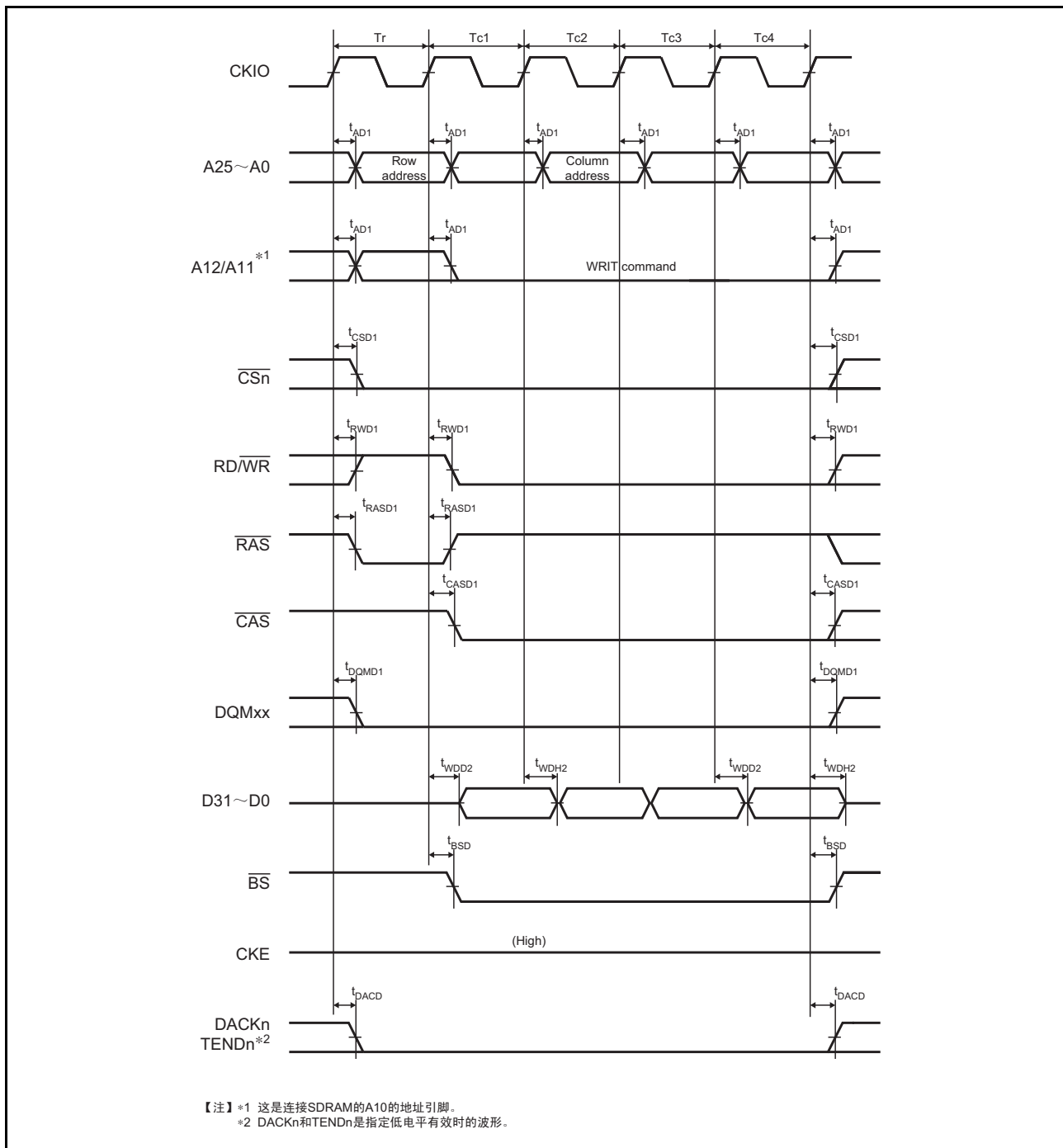


图 52.28 同步 DRAM 突发写的总线周期（4 个写周期）（存储体激活模式：ACT+WRITE 命令，WTRCD=0 个周期，TRWL=0 个周期）

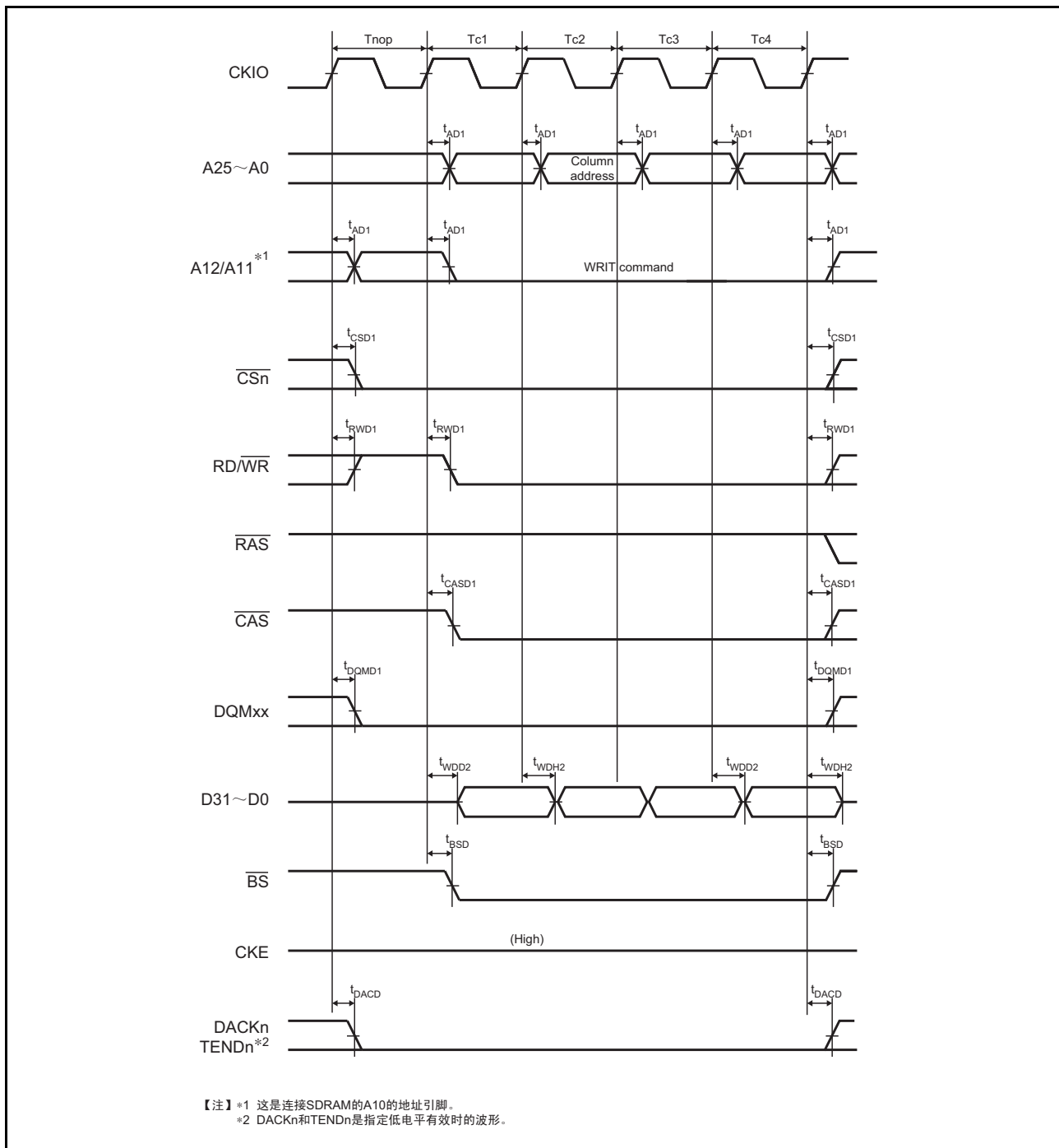


图 52.29 同步 DRAM 突发写的总线周期（4 个写周期）（存储体激活模式：WRITE 命令，同一行地址，WTRCD=0 个周期，TRWL=0 个周期）

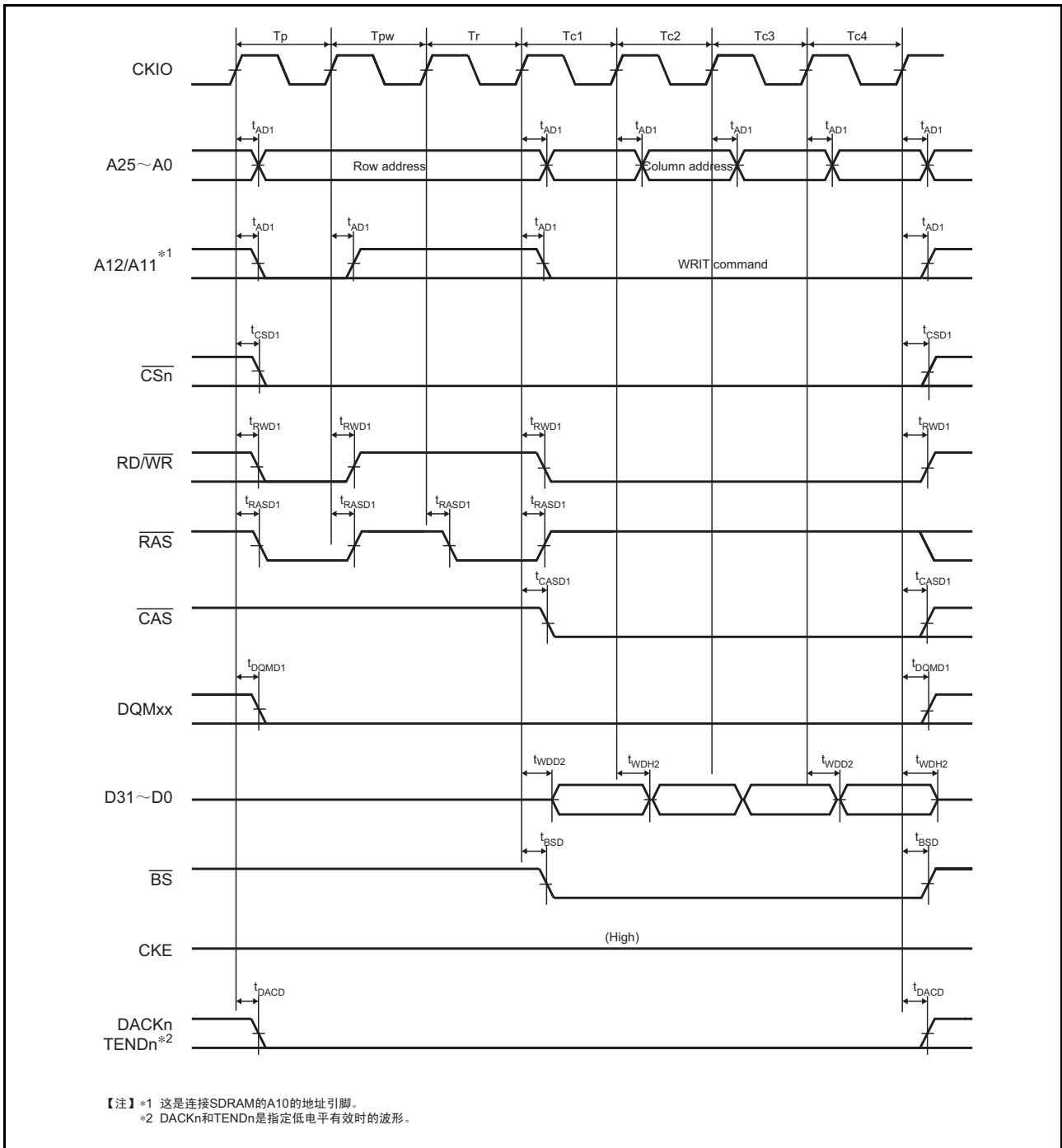


图 52.30 同步 DRAM 突发写的总线周期（4 个写周期）（存储体激活模式：PRE+ACT+WRITE 命令，不同行地址，WTRCD=0 个周期，TRWL=0 个周期）

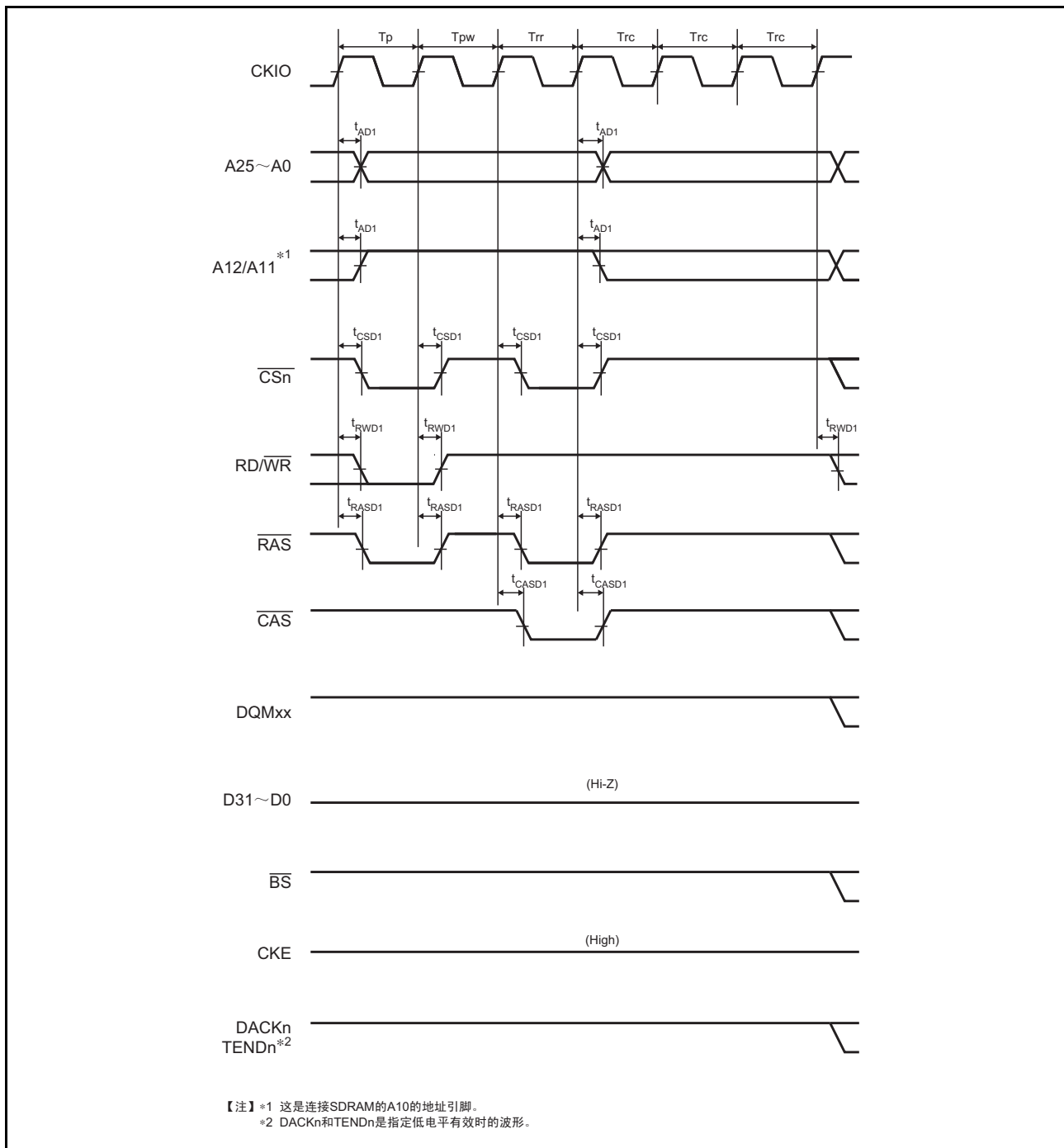


图 52.31 同步 DRAM 自动刷新时序 (WTRP=1 个周期, WTRC=3 个周期)

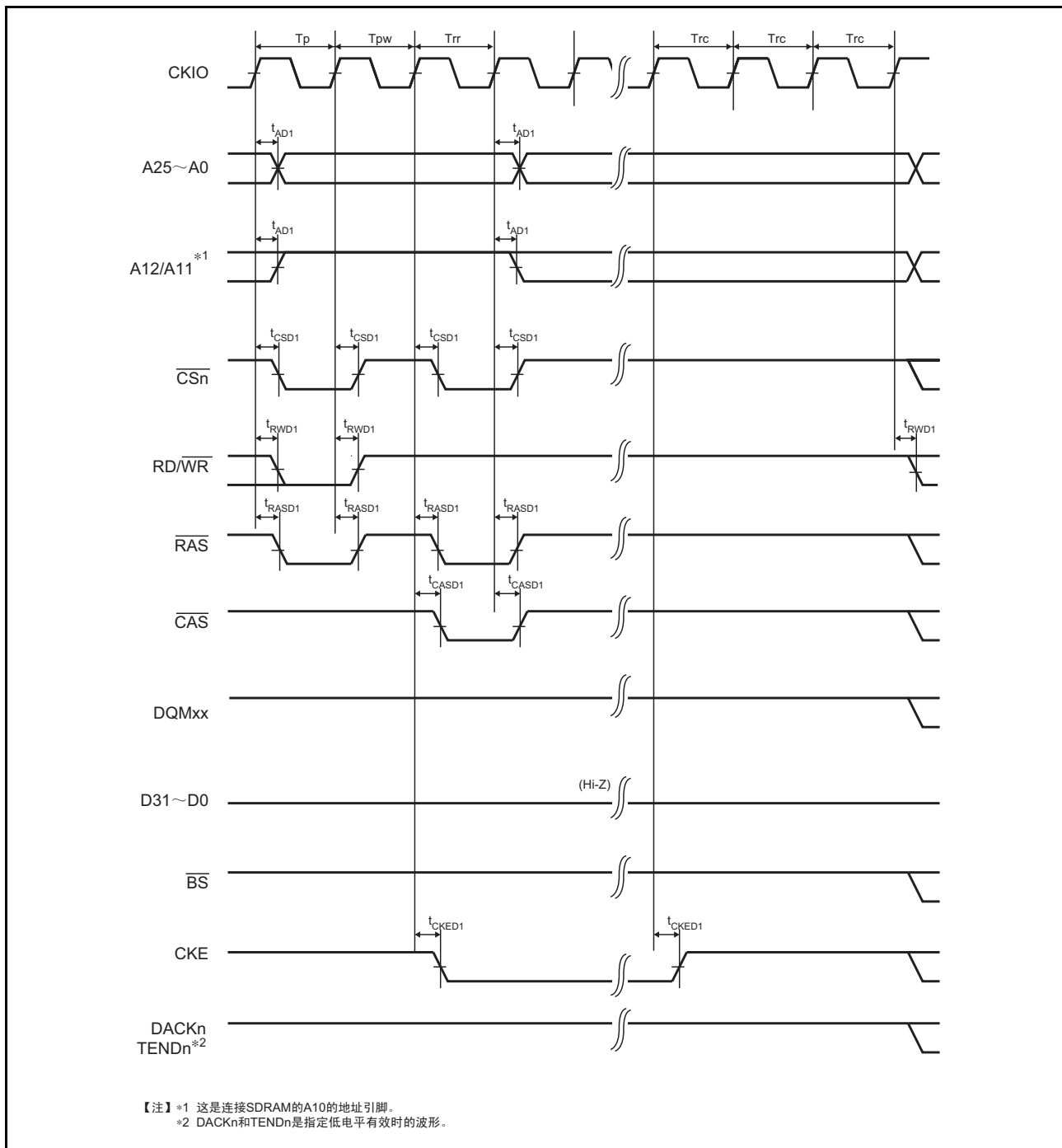


图 52.32 同步 DRAM 自刷新时序 (WTRP=1 个周期)

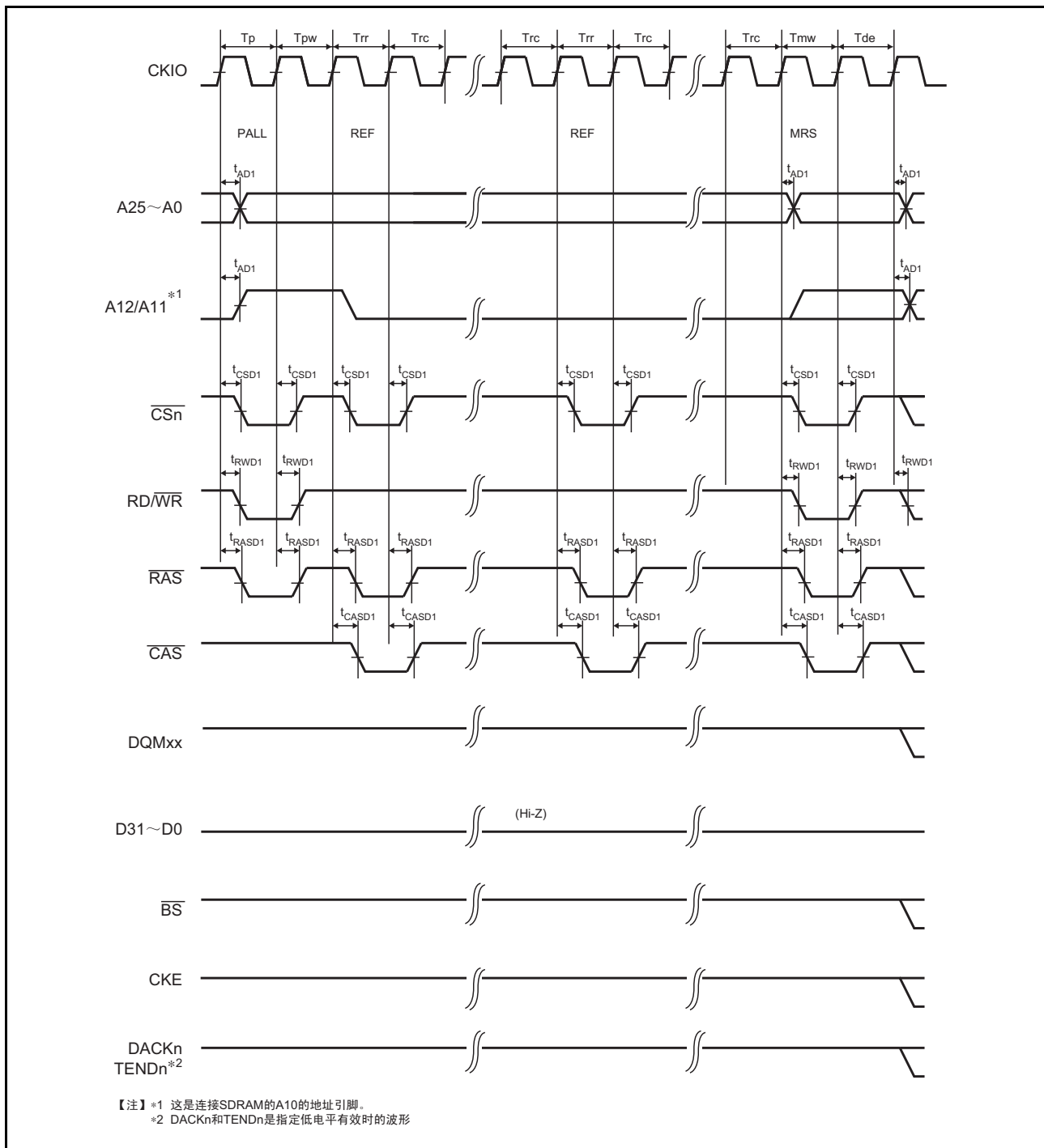


图 52.33 同步 DRAM 模式寄存器的写时序 (WTRP=1 个周期)

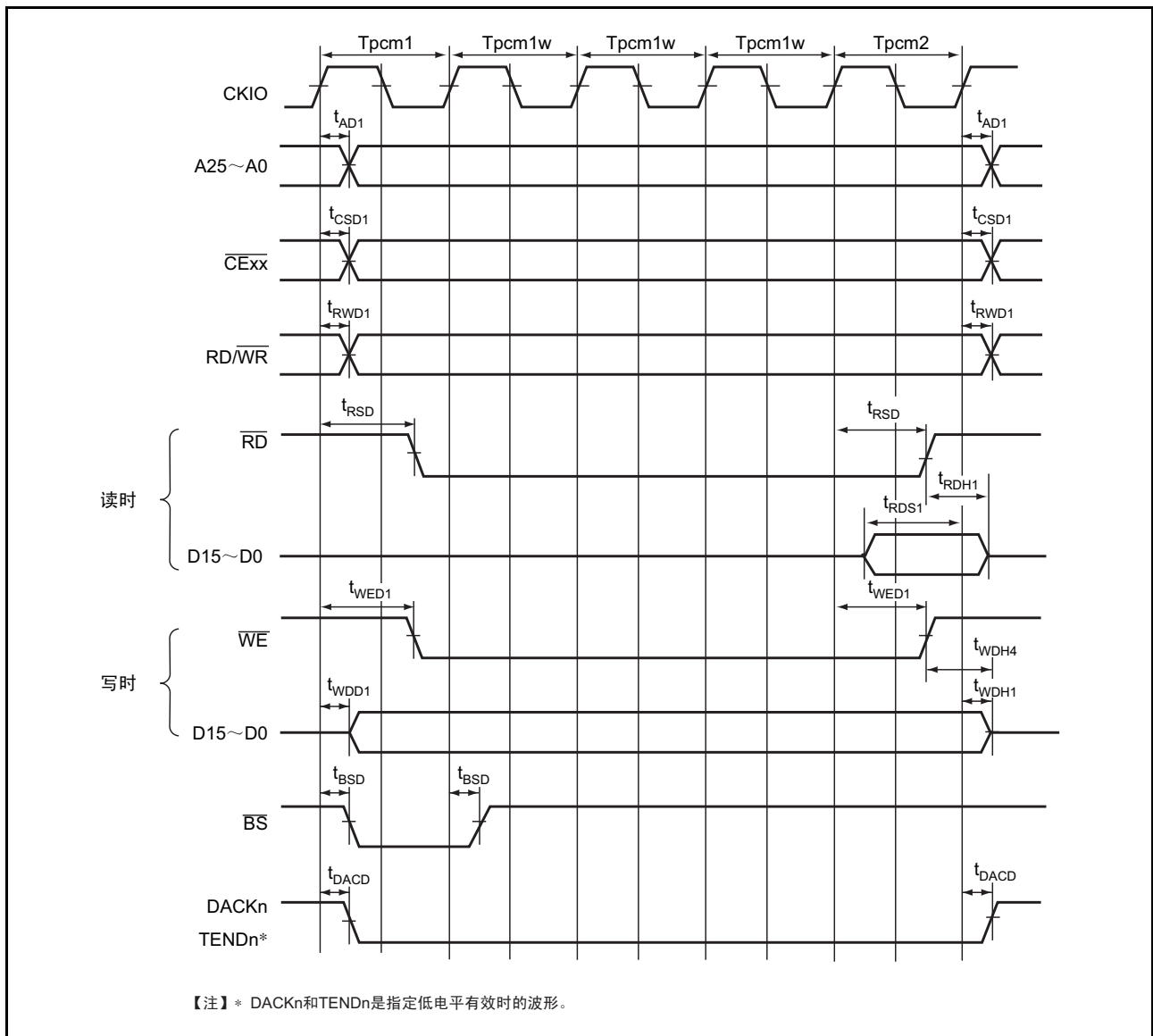


图 52.34 PCMCIA 存储器卡的总线周期 (TED=0 个周期, TEH=0 个周期, 无等待)

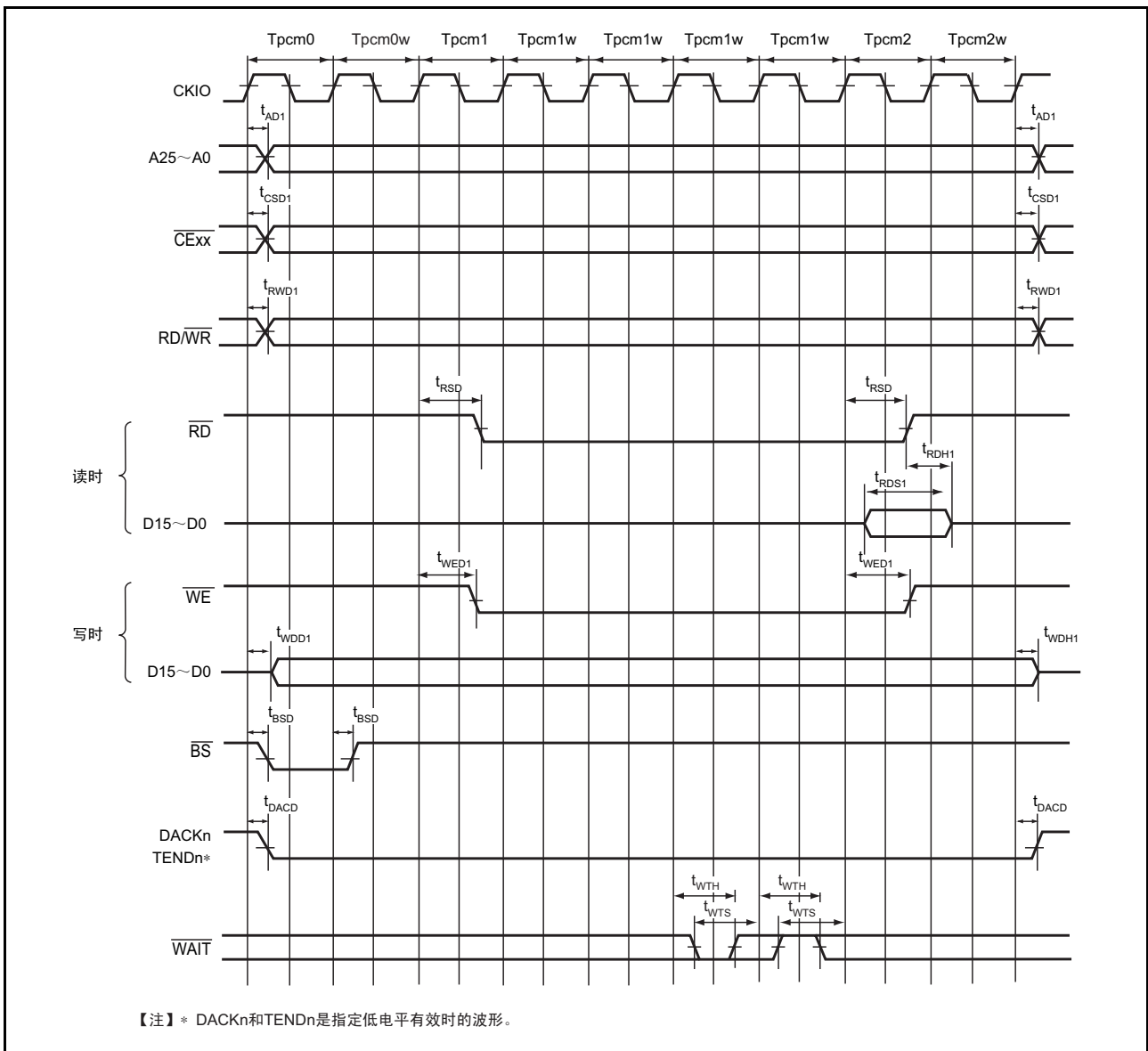


图 52.35 PCMCIA 存储器卡的总线周期 (TED=2 个周期, TEH=1 个周期, 0 个软件等待, 1 个硬件等待)

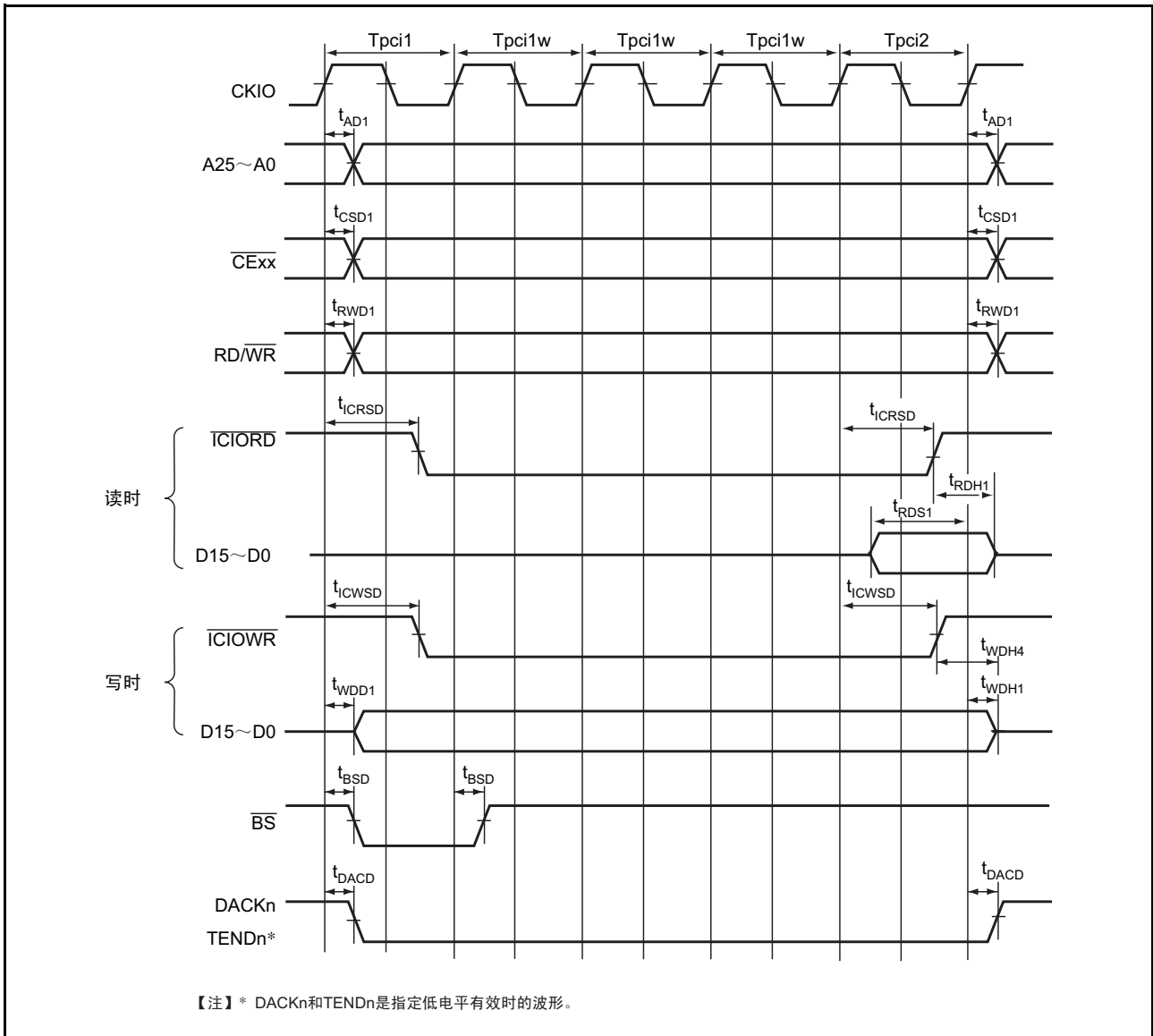


图 52.36 PCMCIA I/O 卡的总线周期 (TED=0 个周期, TEH=0 个周期, 无等待)

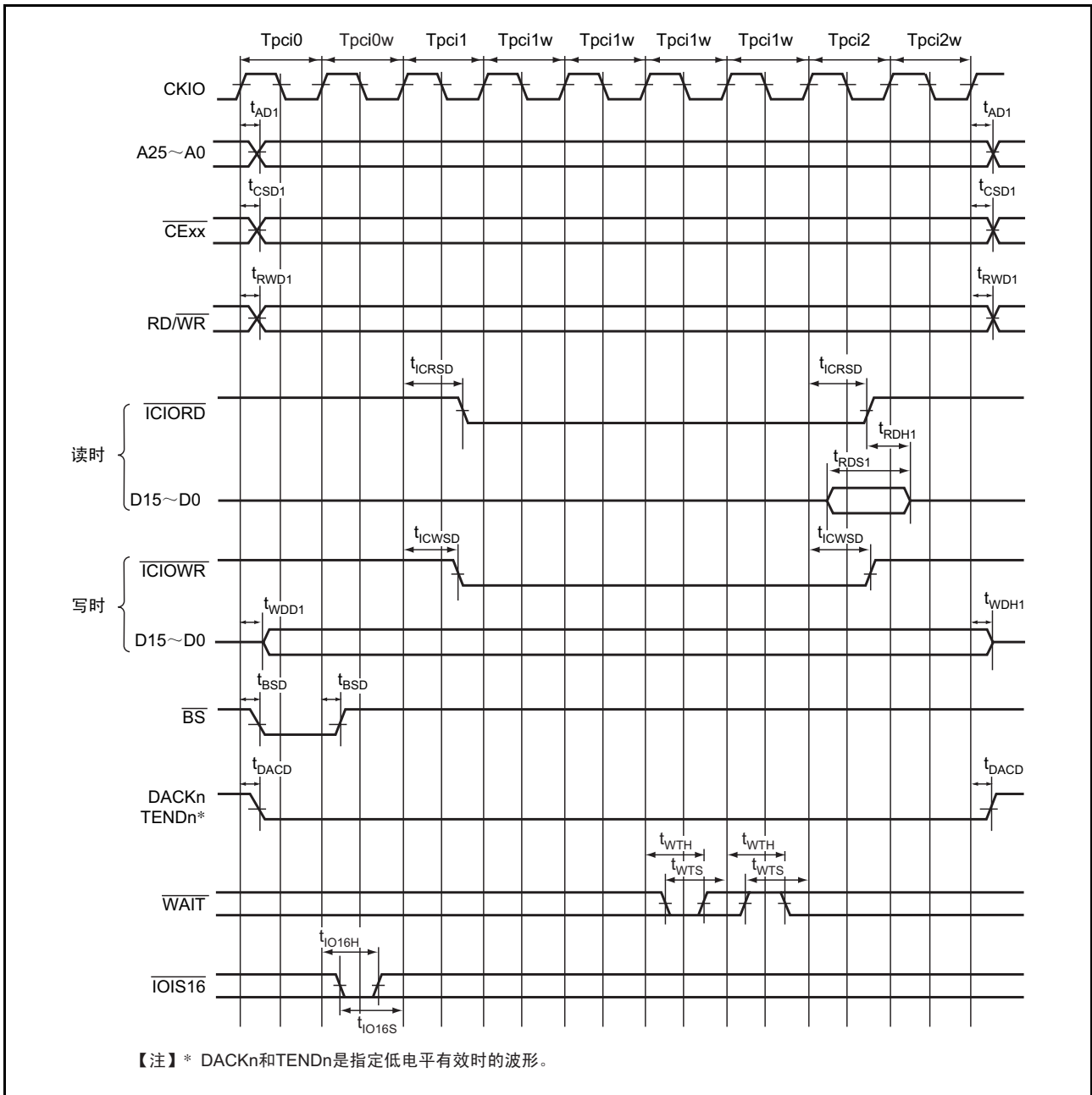


图 52.37 PCMCIA I/O 卡的总线周期 (TED=2 个周期, TEH=1 个周期, 0 个软件等待, 1 个硬件等待)

52.4.4 UBC 时序

表 52.8 UBC 时序

项目	符号	Min.	Max.	单位	参照图
UBCTRG 延迟时间	t_{UBCTGD}	—	14	ns	52.38

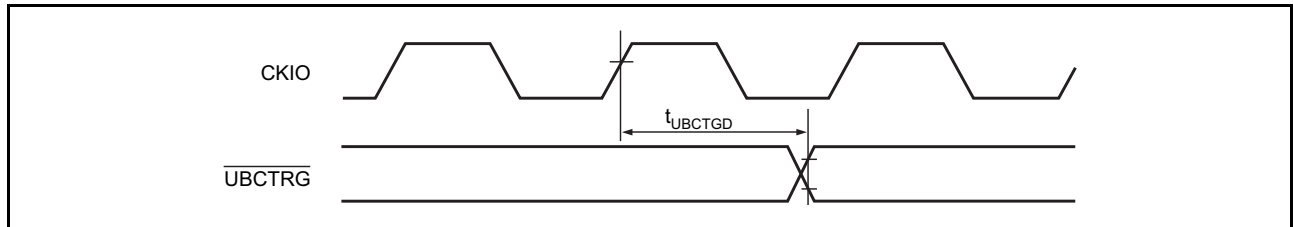


图 52.38 UBC 触发时序

52.4.5 直接存储器存取控制器的时序

表 52.9 直接存储器存取控制器的时序

项目	符号	Min.	Max.	单位	参照图
DREQ 准备时间	t_{DRQS}	5.5	—	ns	52.39
DREQ 保持时间	t_{DRQH}	2.5	—		
DACK、TEND 延迟时间	t_{DACD}	0	12		52.40

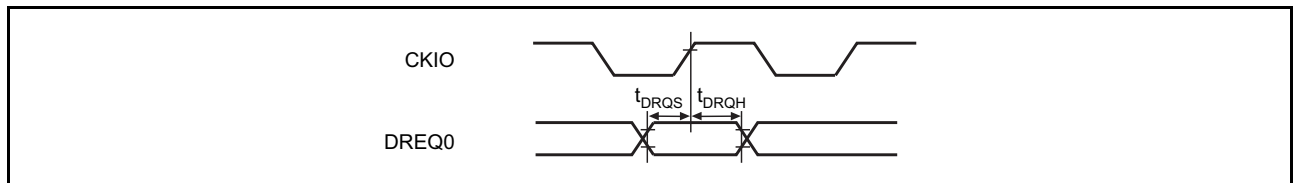


图 52.39 DREQ 输入时序

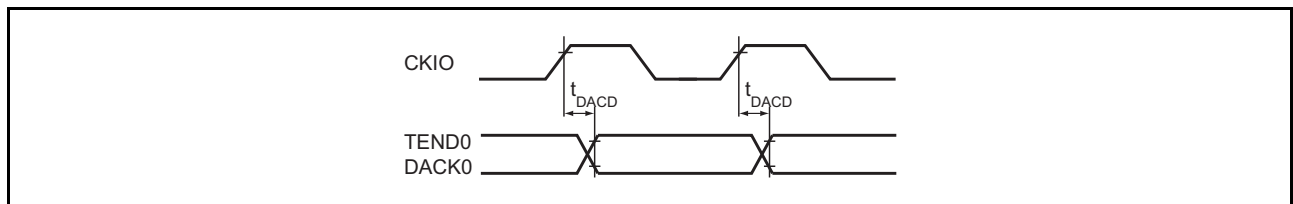


图 52.40 DACK、TEND 输出时序

52.4.6 多功能定时器脉冲单元 2 的时序

表 52.10 多功能定时器脉冲单元 2 的时序

项目	符号	Min.	Max.	单位	参照图
定时器时钟脉宽（单边沿的指定）	$t_{TCKWH/L}$	1.5	—	t_{p0cyc}	52.41
定时器时钟脉宽（双边沿的指定）	$t_{TCKWH/L}$	2.5	—	t_{p0cyc}	
定时器时钟脉宽（相位计数模式）	$t_{TCKWH/L}$	2.5	—	t_{p0cyc}	

【注】 t_{p0cyc} 表示外围时钟（P0 ϕ ）的周期。

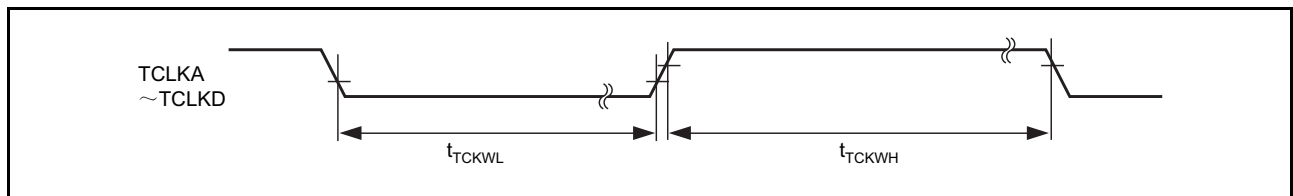


图 52.41 时钟输入时序

52.4.7 看门狗定时器的时序

表 52.11 看门狗定时器的时序

项目	符号	Min.	Max.	单位	参照图
WDTOVF 延迟时间	t_{WOVD}	—	100	ns	52.42

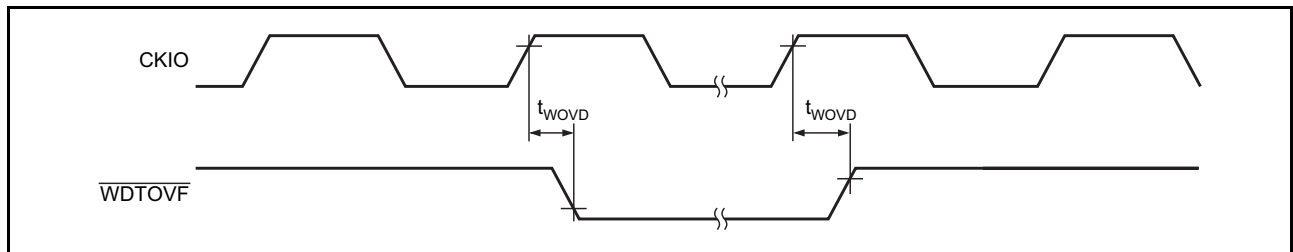


图 52.42 WDTOVF 输出时序

52.4.8 FIFO 内置型串行通信接口的时序

表 52.12 FIFO 内置型串行通信接口的时序

项目		符号	Min.	Max.	单位	参照图
输入时钟周期	时钟同步	t_{Scyc}	12	—	t_{p1cyc}	52.43
	异步		4	—	t_{p1cyc}	
输入时钟的上升时间		t_{SCKr}	—	1.5	t_{p1cyc}	
输入时钟的下降时间		t_{SCKf}	—	1.5	t_{p1cyc}	
输入时钟宽度		t_{SCKW}	0.4	0.6	t_{Scyc}	
发送数据的延迟时间（时钟同步）		t_{TXD}	—	$3t_{p1cyc}+15$	ns	52.44
接收数据的准备时间（时钟同步）		t_{RXS}	$4t_{p1cyc}+15$	—	ns	
接收数据的保持时间（时钟同步）		t_{RXH}	$1t_{p1cyc}+15$	—	ns	

【注】 t_{p1cyc} 表示 1 个外围时钟（P1 ϕ ）的周期。

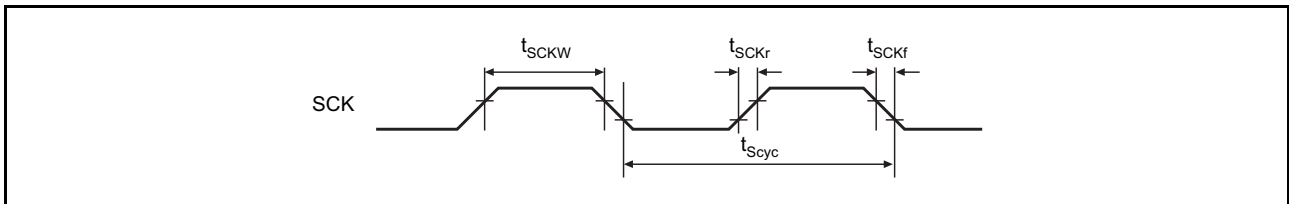


图 52.43 SCK 输入时钟时序

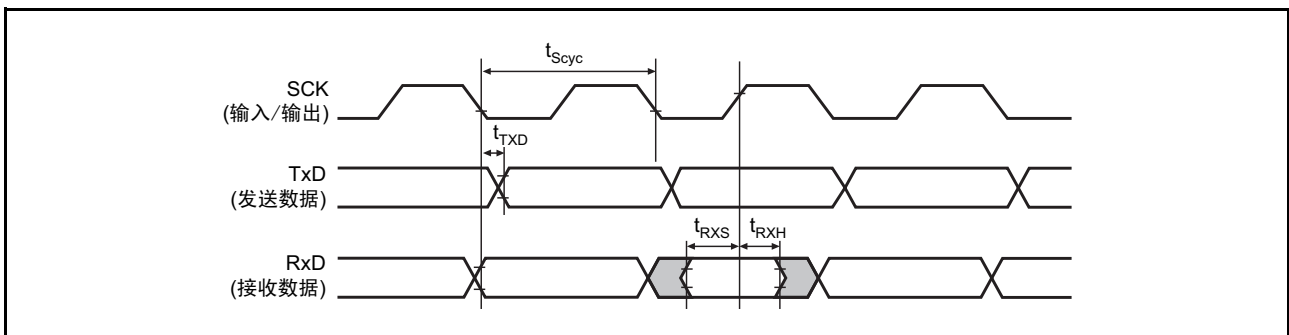


图 52.44 时钟同步式模式中的发送和接收数据的输入 / 输出时序

52.4.9 瑞萨串行外围接口的时序

表 52.13 瑞萨串行外围接口的时序

项目	符号	Min.	Max.	单位	参照图
RSPCK 时钟周期	主控	2	4096	t_{cyc}	52.45
	从属	8	4096		
RSPCK 时钟的高电平脉宽	主控	0.4	—	t_{SPCKWH}	t_{SPcyc}
	从属	0.4	—		
RSPCK 时钟的低电平脉宽	主控	0.4	—	t_{SPCKWL}	t_{SPcyc}
	从属	0.4	—		
数据输入的准备时间	主控	15	—	ns	52.46 ~ 52.49
	从属	0	—	t_{cyc}	
数据输入的保持时间	主控	0	—	ns	t_{cyc}
	从属	4	—		
SSL 准备时间	主控	1	8	t_{SPcyc}	t_{cyc}
	从属	4	—		
SSL 保持时间	主控	1	8	t_{SPcyc}	t_{cyc}
	从属	4	—		
数据输出的延迟时间	主控	—	21	ns	t_{cyc}
	从属	—	4		
数据输出的保持时间	主控	5	—	ns	t_{cyc}
	从属	3	—		
连续发送的延迟时间	主控	$1 \times t_{SPcyc} + 2 \times t_{cyc}$	$8 \times t_{SPcyc} + 2 \times t_{cyc}$	ns	t_{cyc}
	从属	$4 \times t_{cyc}$	—		
从属存取时间	t_{SA}	—	4	t_{cyc}	52.48、52.49
从属输出释放时间	t_{REL}	—	3	t_{cyc}	

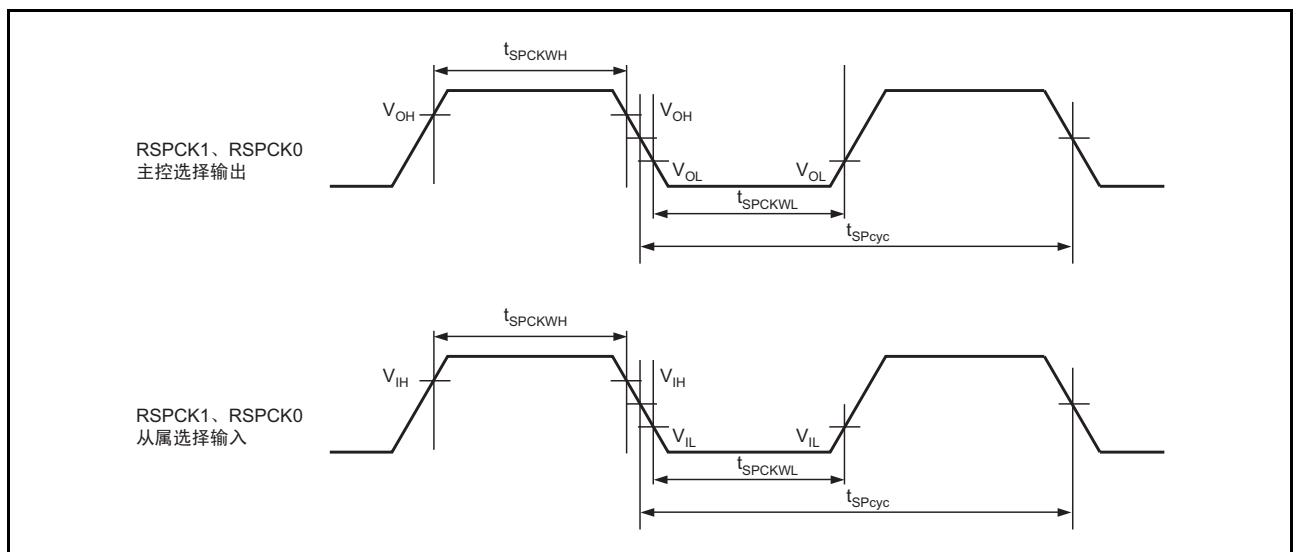


图 52.45 时钟时序

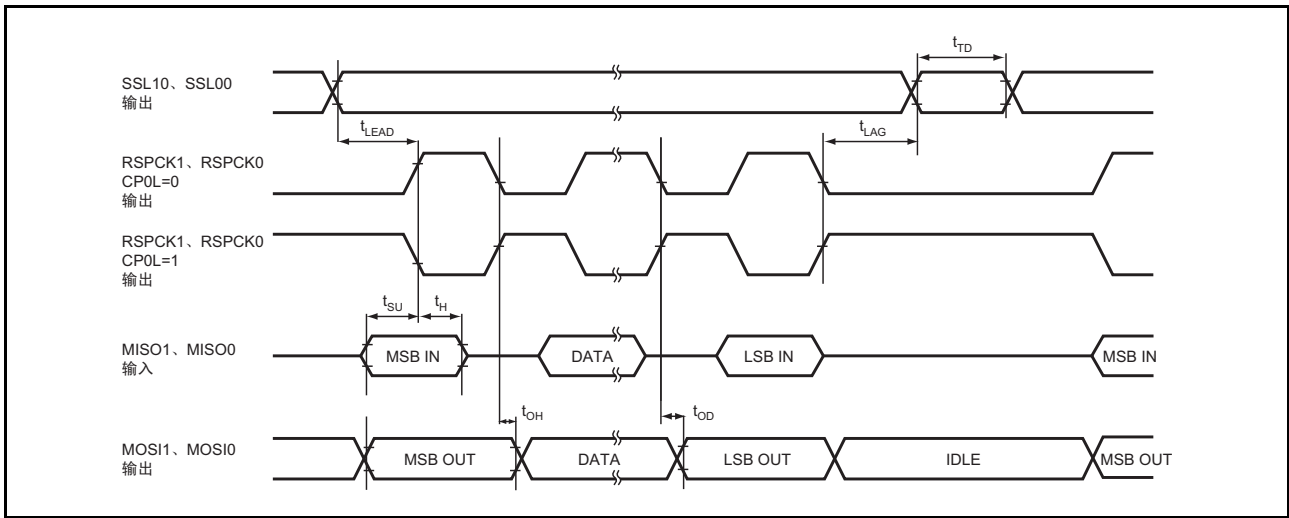


图 52.46 发送和接收时序（主控，CPHA 位为“0”）

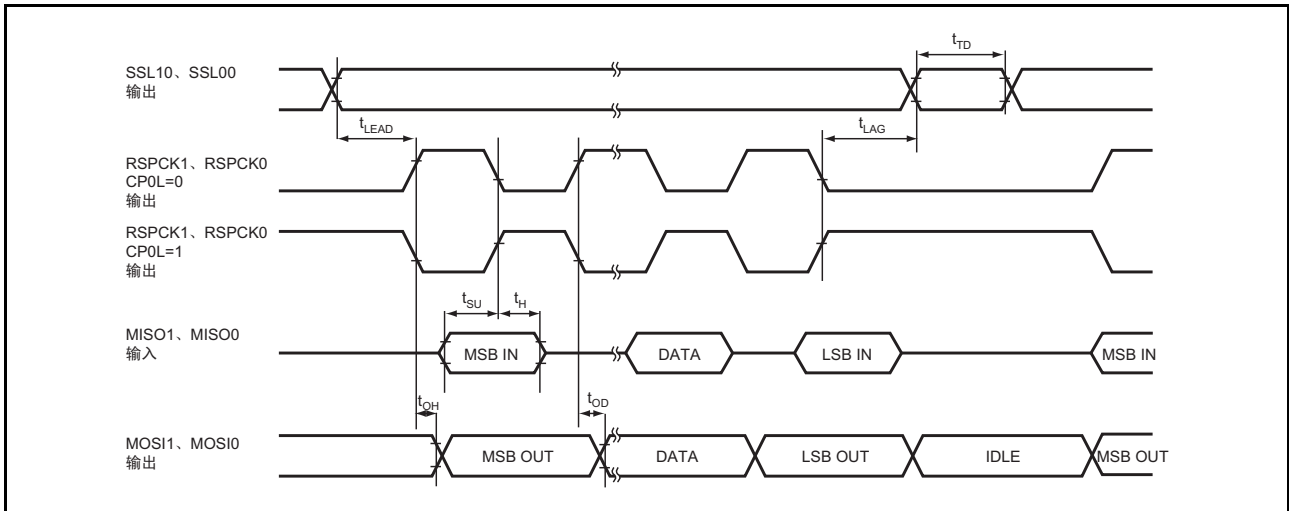


图 52.47 发送和接收时序（主控，CPHA 位为“1”）

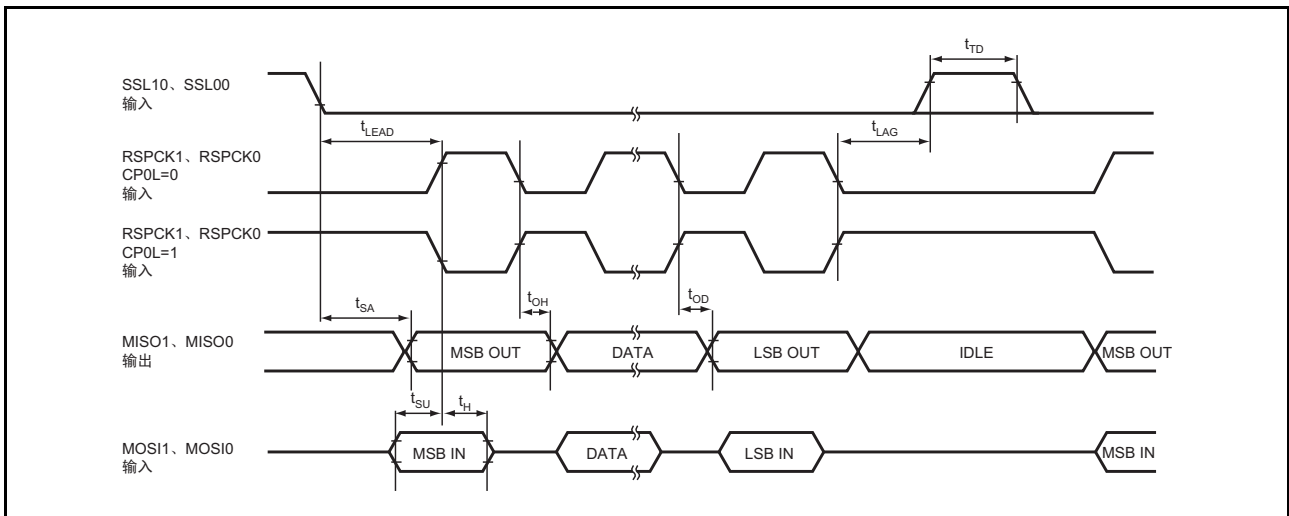


图 52.48 发送和接收时序（从属，CPHA 位为“0”）

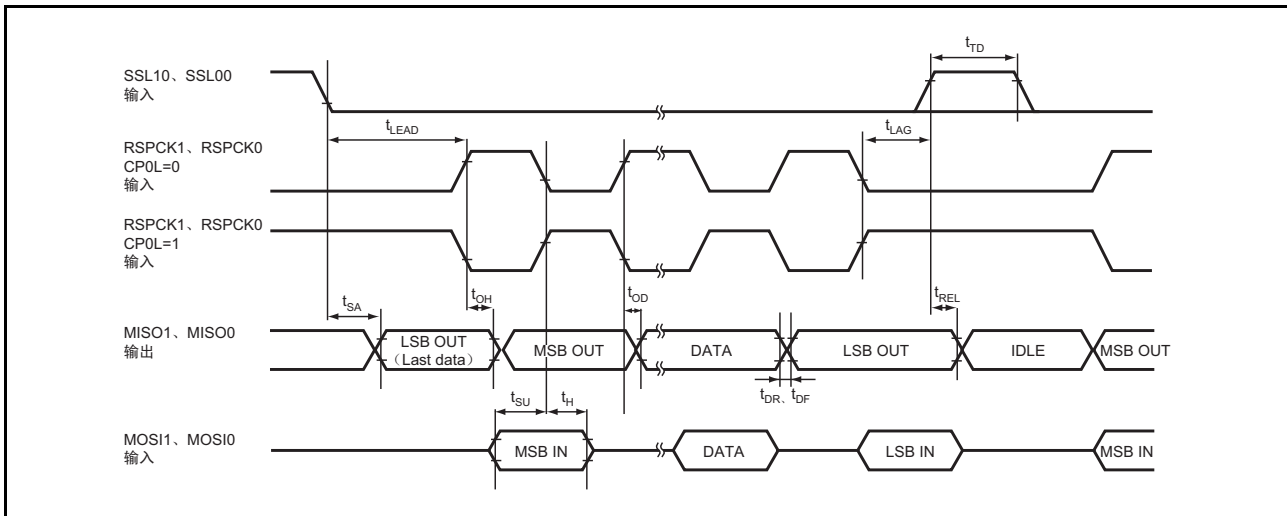


图 52.49 发送和接收时序（从属，CPHA 位为“1”）

52.4.10 瑞萨四路串行外围接口的时序

表 52.14 瑞萨四路串行外围接口的时序

项目	符号	Min.	Max.	单位	参照图
QSPCLK 时钟周期	t_{QScyc}	1	4080	t_{cyc}	52.50
数据输入的准备时间	t_{SU}	5.0	—	ns	52.51、 52.52
数据输入的保持时间	t_{H}	0.0	—	ns	
SSL 准备时间	t_{LEAD}	1.5	8.5	t_{QScyc}	
SSL 保持时间	t_{LAG}	1	8	t_{QScyc}	
数据输出的延迟时间	t_{OD}	—	10.0	ns	
数据输出的保持时间	t_{OH}	-5.0	—	ns	
连续传送的延迟时间	t_{TD}	1	8	t_{QScyc}	

【注】 t_{cyc} 表示 1 个总线时钟（P1 ϕ ）周期的时间。

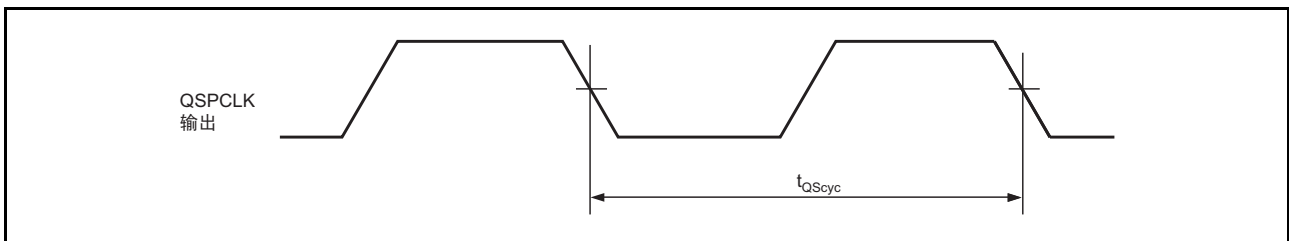


图 52.50 时钟时序

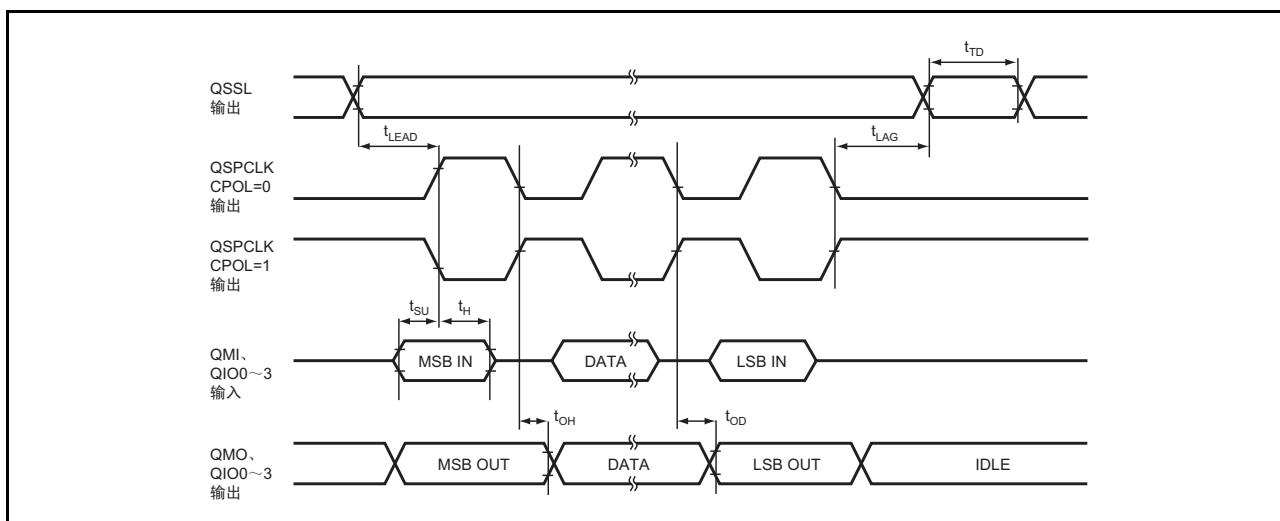


图 52.51 发送和接收时序 (CPHA 位为 “0”)

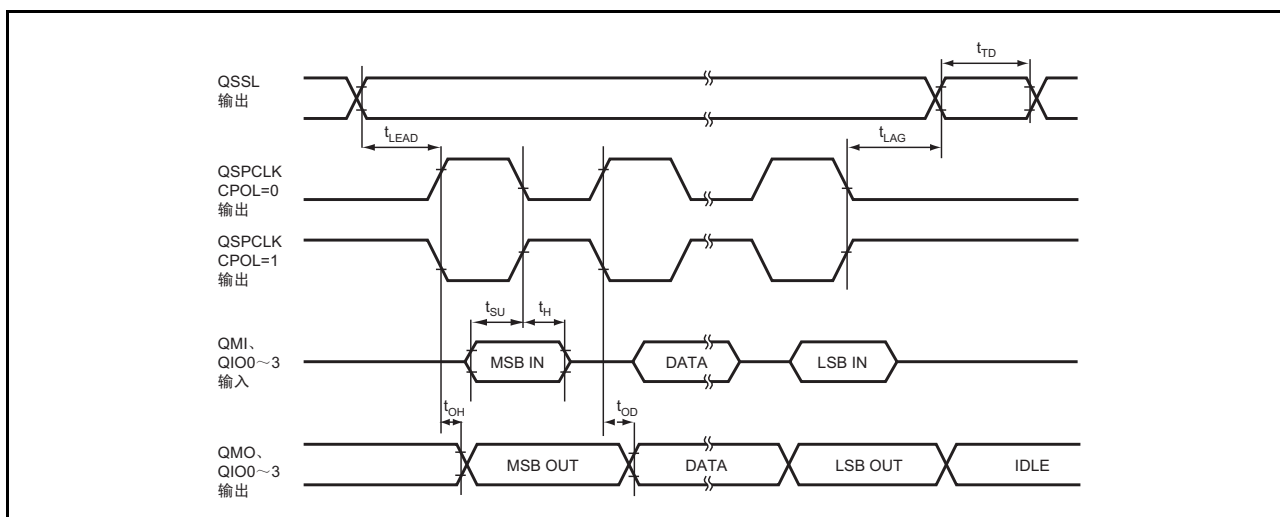


图 52.52 发送和接收时序 (CPHA 位为 “1”)

52.4.11 SPI 多 I/O 总线控制器的时序

表 52.15 SPI 多 I/O 总线控制器的时序

项目	符号	Min.	Max.	单位	参照图
SPBCLK 时钟周期	t_{SPBcyc}	2	2	t_{cyc}	52.53
数据输入的准备时间	t_{SU}	5.0	—	ns	52.54、 52.55
数据输入的保持时间	t_H	0.0	—	ns	
SSL 准备时间	t_{LEAD}	1	8	t_{SPBcyc}	
SSL 保持时间	t_{LAG}	1.5	1.5	t_{SPBcyc}	
连续传送的延迟时间	t_{TD}	1	1	t_{SPBcyc}	
数据输出的延迟时间	t_{OD}	—	4.0	ns	
数据输出的保持时间	t_{OH}	-2.0	—	ns	
数据输出缓冲器的 ON 时间	t_{BON}	—	4.0	ns	52.56、 52.57
数据输出缓冲器的 OFF 时间	t_{BOFF}	-9.0	2.0	ns	

【注】 t_{cyc} 表示 1 个总线时钟 (Bφ) 周期的时间。

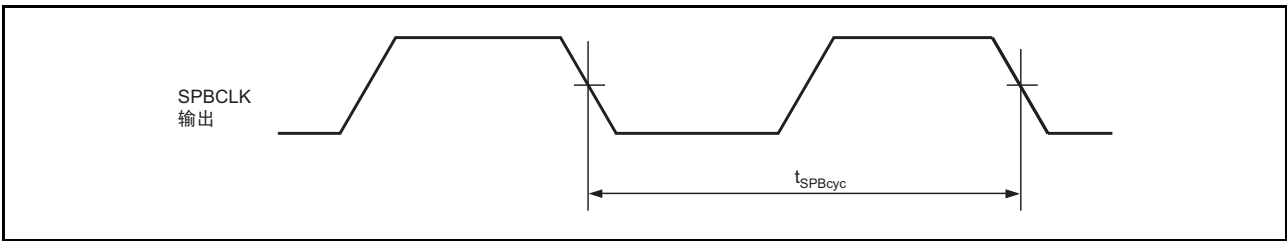


图 52.53 时钟时序

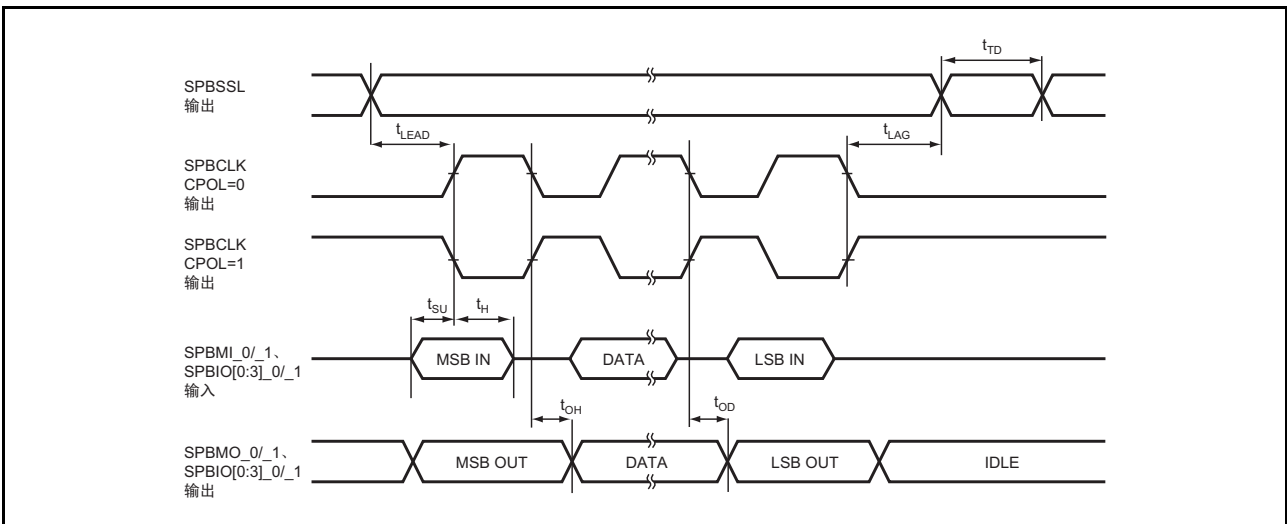


图 52.54 发送和接收时序 (CPHAT 位和 CPHAR 位都为 “0”)

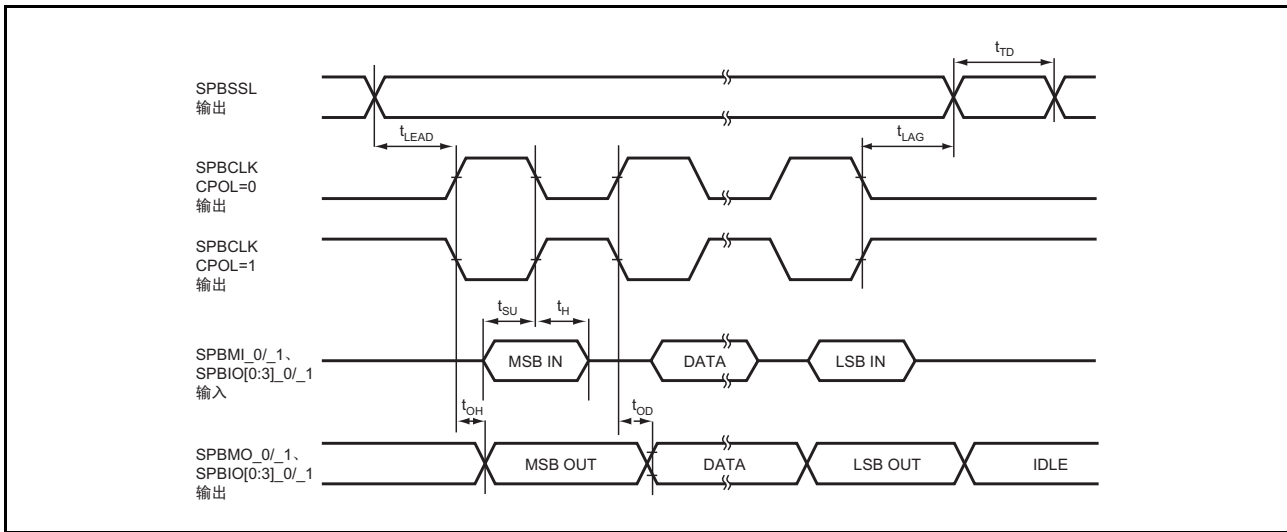


图 52.55 发送和接收时序 (CPHAT 位和 CPHAR 位都为 “1”)

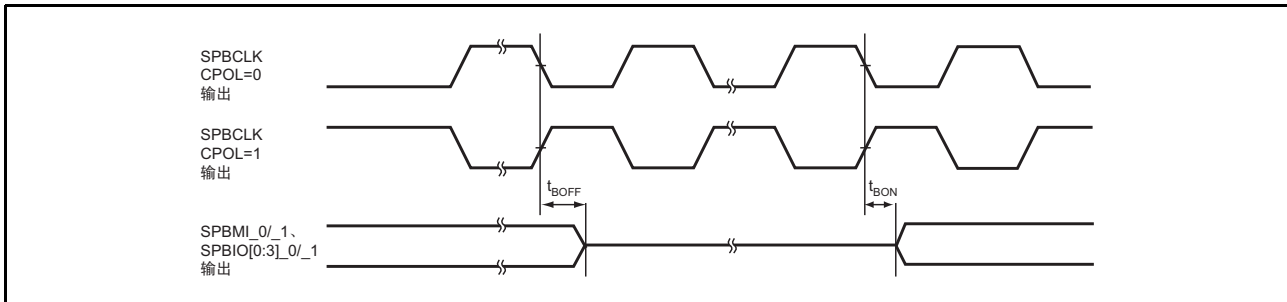


图 52.56 缓冲器 ON/OFF 的时序 (CPHAT 位和 CPHAR 位都为 “0”)

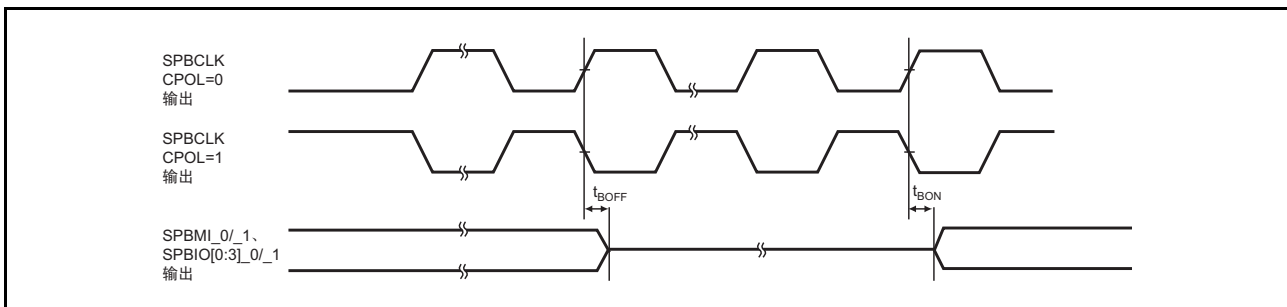


图 52.57 缓冲器 ON/OFF 的时序 (CPHAT 位和 CPHAR 位都为 “1”)

52.4.12 I²C 总线接口 3 的时序表 52.16 I²C 总线接口 3 的时序

项目	符号	Min.	Max.	单位	参照图
SCL 输入的周期时间	t_{SCL}	$12t_{p0cyc}^{*1}+600$	—	ns	52.58
SCL 输入的高电平脉宽	t_{SCLH}	$3t_{p0cyc}^{*1}+300$	—	ns	
SCL 输入的低电平脉宽	t_{SCLL}	$5t_{p0cyc}^{*1}+300$	—	ns	
SCL 输入和 SDA 输入的上升时间	t_{Sr}	—	300	ns	
SCL 输入和 SDA 输入的下降时间	t_{Sf}	—	300	ns	
SCL 输入和 SDA 输入的尖峰脉冲消除时间 *2	t_{SP}	—	1、2	t_{p0cyc}^{*1}	
SDA 输入总线自由时间	t_{BUF}	5	—	t_{p0cyc}^{*1}	
开始条件输入的保持时间	t_{STAH}	3	—	t_{p0cyc}^{*1}	
重新开始发送条件输入的准备时间	t_{STAS}	3	—	t_{p0cyc}^{*1}	
停止条件输入的准备时间	t_{STOS}	3	—	t_{p0cyc}^{*1}	
数据输入的准备时间	t_{SDAS}	$1t_{p0cyc}^{*1}+20$	—	ns	
数据输入保持时间	t_{SDAH}	0	—	ns	
SCL 和 SDA 的电容负载	Cb	0	400	pF	
SCL 输出和 SDA 输出的下降时间 *3	t_{Sf}	—	250	ns	

【注】 *1 t_{p0cyc} 表示外围时钟 (P0 ϕ) 的周期。

*2 取决于 NF2CYC 寄存器的值。

*3 表示 I/O 缓冲器的特性。

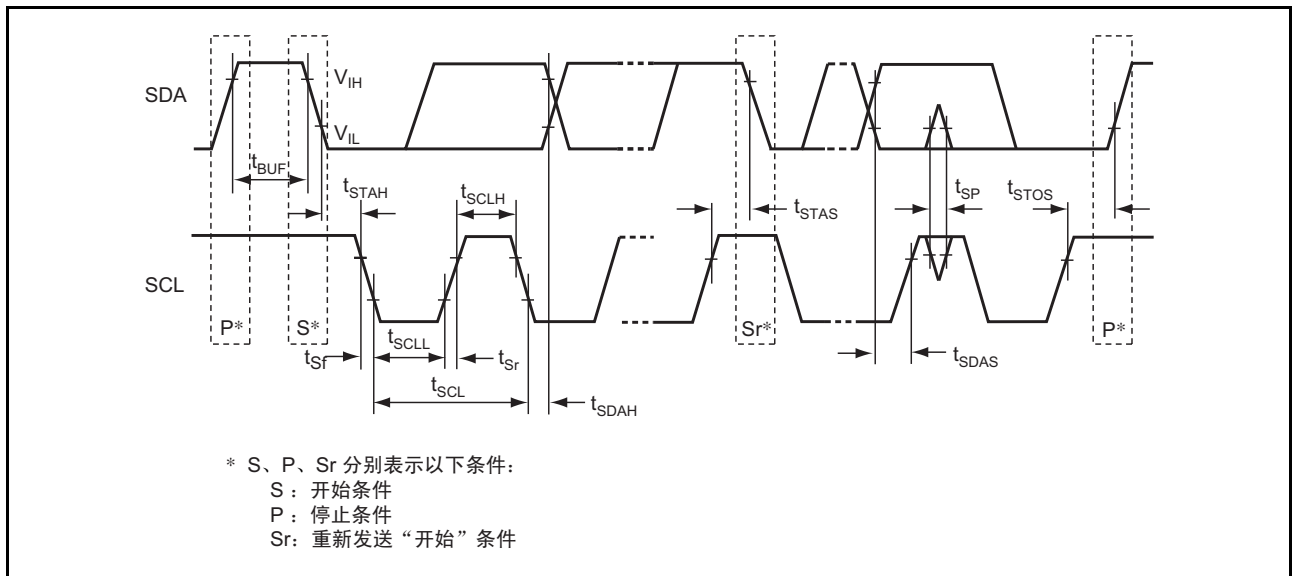


图 52.58 输入 / 输出时序

52.4.13 串行音频接口的时序

表 52.17 串行音频接口的时序

项目		符号	Min.	Max.	单位	备注	参照图
输出时钟周期		t_O	80	64000	ns	输出	52.59
输入时钟周期		t_I	80	64000	ns	输入	
时钟高电平		t_{HC}	32	—	ns	双向	
时钟低电平		t_{LC}	32	—	ns		
时钟上升时间		t_{RC}	—	25	ns	输出	
延迟	不使用噪声消除器时	t_{DTR}	-5	25	ns		52.60、 52.61
	使用噪声消除器时		10	45			
准备时间		t_{SR}	25	—	ns		
保持时间		t_{HTR}	5	—	ns		

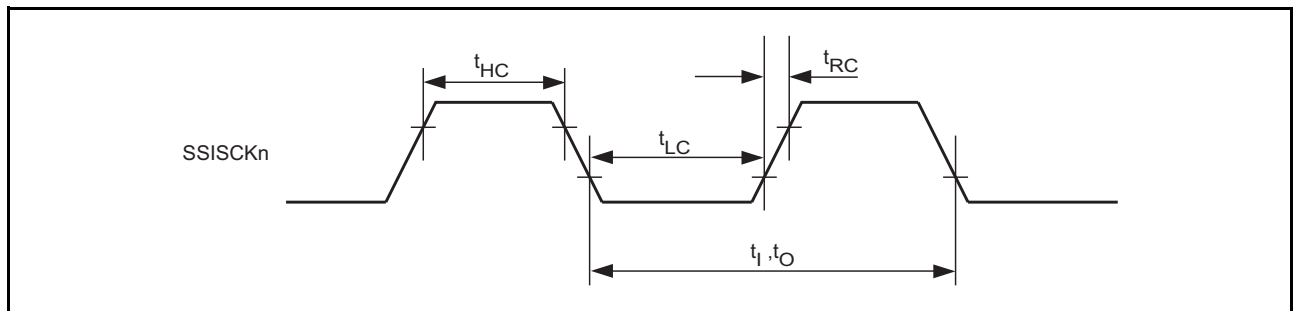


图 52.59 时钟输入 / 输出时序

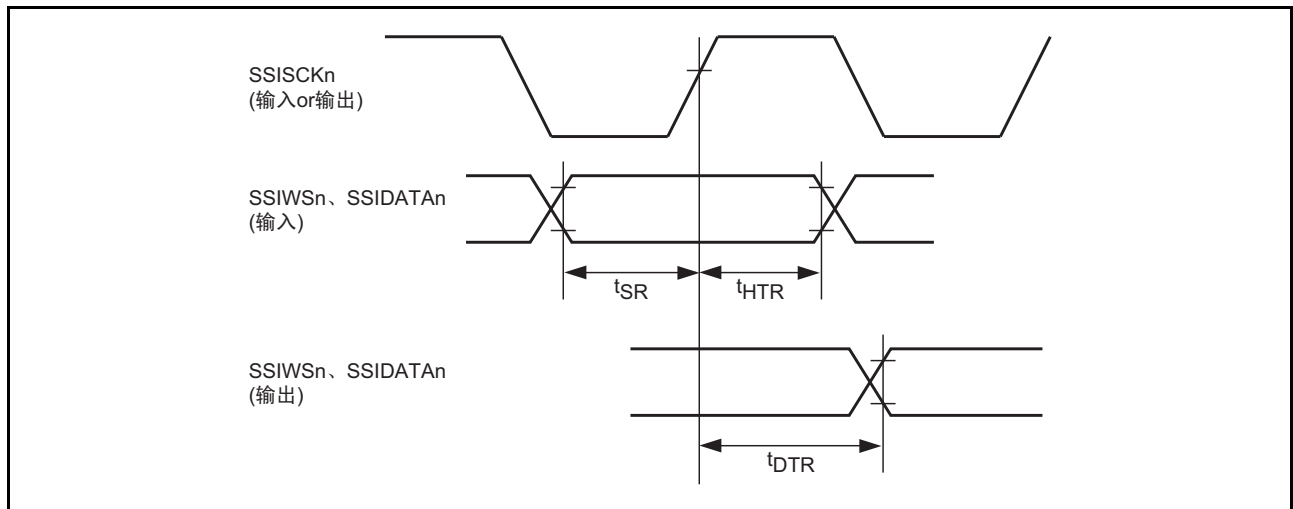


图 52.60 发送和接收时序 (SSISCKn 上升同步)

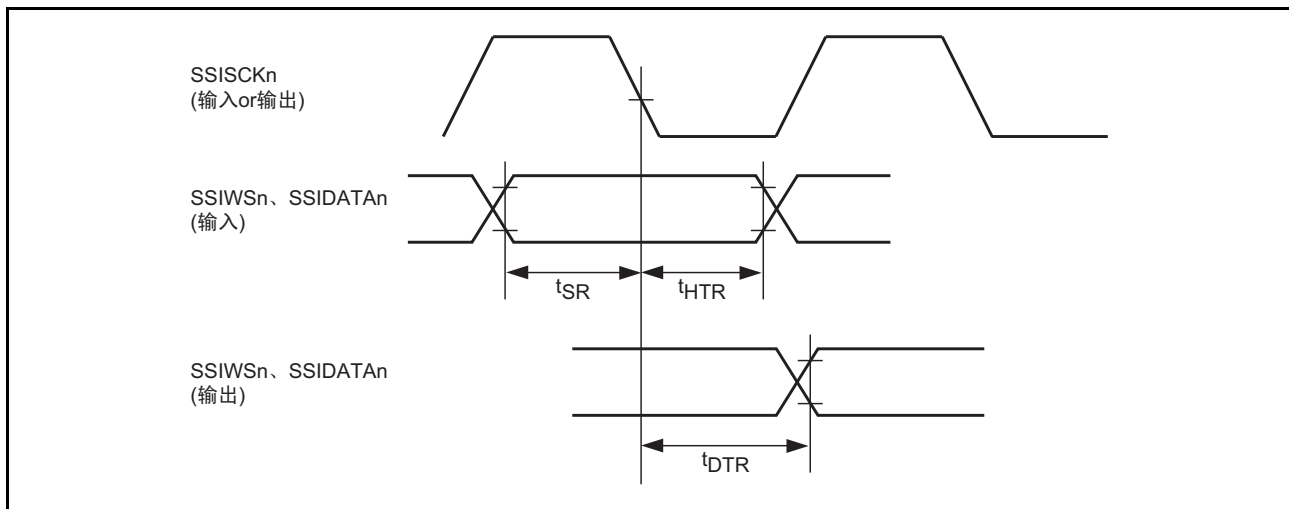


图 52.61 发送和接收时序（SSISCKn 下降同步）

52.4.14 带 FIFO 的时钟同步串行 I/O 的时序

表 52.18 带 FIFO 的时钟同步串行 I/O 的时序

项目	符号	Min.	Max.	单位	参照图
SCK_SIO 时钟输入 / 输出的周期时间	t_{S1cyc}	80	—	ns	52.62 ~ 52.64
SCK_SIO 输出的高电平宽度	t_{SWHO}	$0.4 \times t_{S1cyc}$	—		52.62、52.63
SCK_SIO 输出的低电平宽度	t_{SWLO}	$0.4 \times t_{S1cyc}$	—		
SIOFSYNC 输出的延迟时间	t_{FSD}	-5	20		
SCK_SIO 输入的高电平宽度	t_{SWHI}	$0.4 \times t_{S1cyc}$	—		52.64
SCK_SIO 输入的低电平宽度	t_{SWLI}	$0.4 \times t_{S1cyc}$	—		
SIOFSYNC 输入的准备时间	t_{FSS}	20	—		
SIOFSYNC 输入的保持时间	t_{FSH}	20	—		
TXD_SIO 输出的延迟时间	t_{STDD}	-5	20		52.62 ~ 52.64
RXD_SIO 输入的准备时间	t_{SRDS}	20	—		
RXD_SIO 输入的保持时间	t_{SRDH}	20	—		

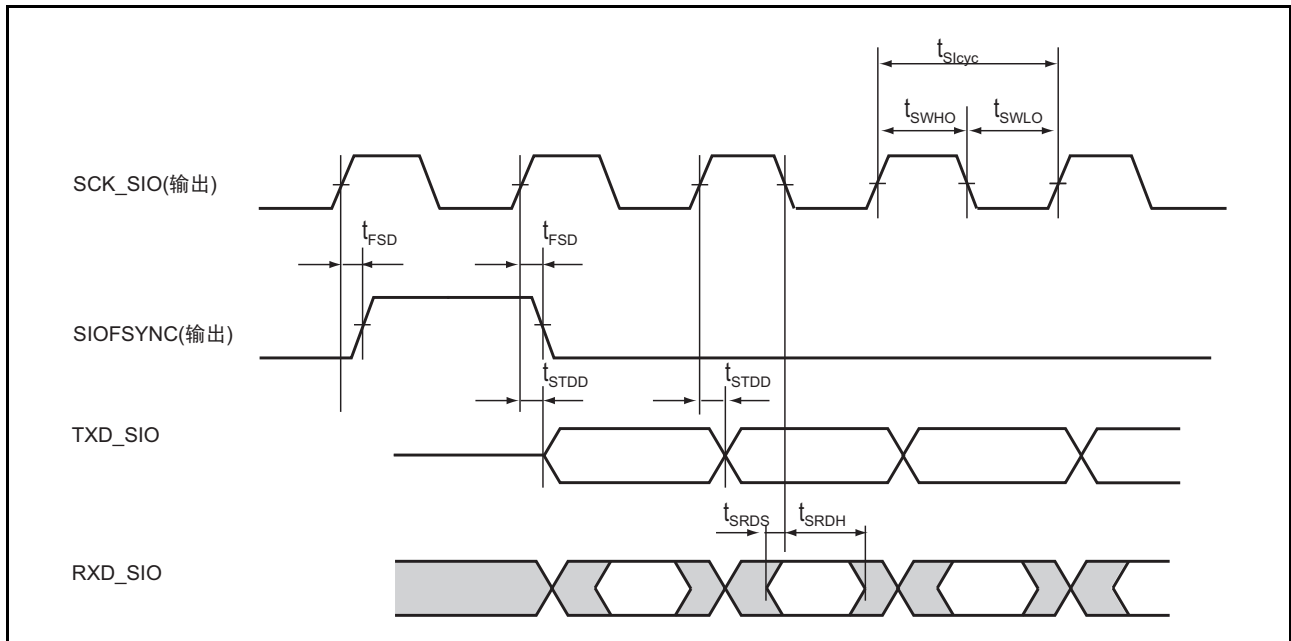


图 52.62 发送和接收时序（主控模式 1/ 下降沿采样的情况）

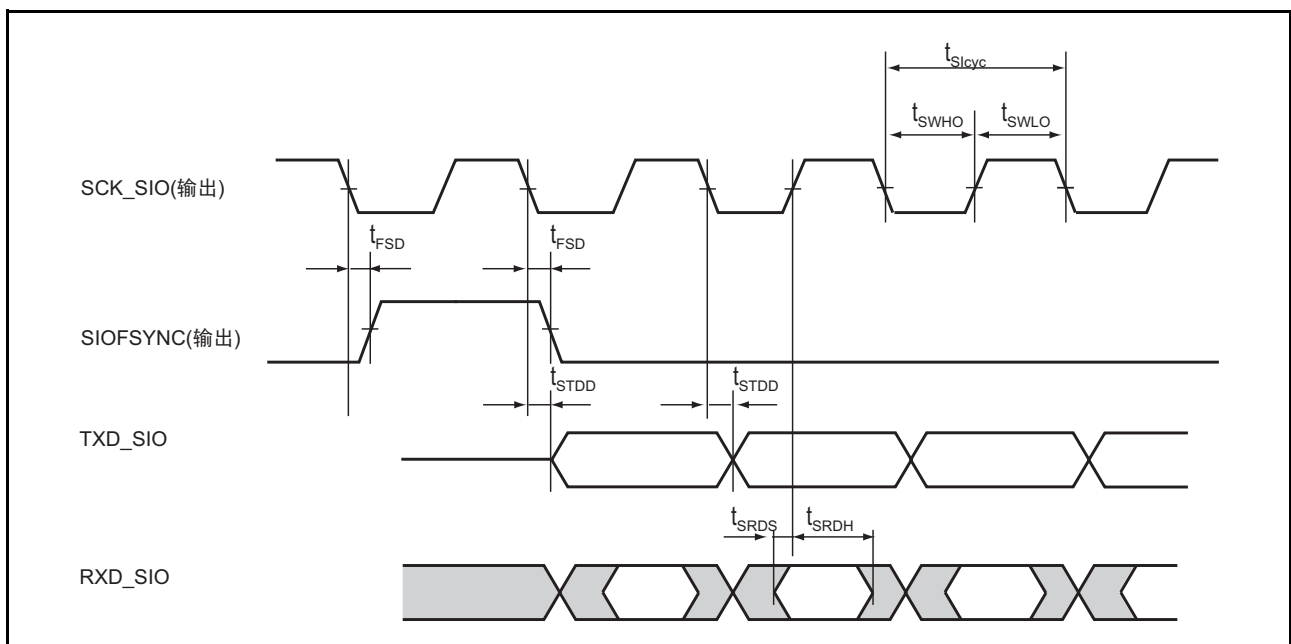


图 52.63 发送和接收时序（主控模式 1/ 上升沿采样的情况）

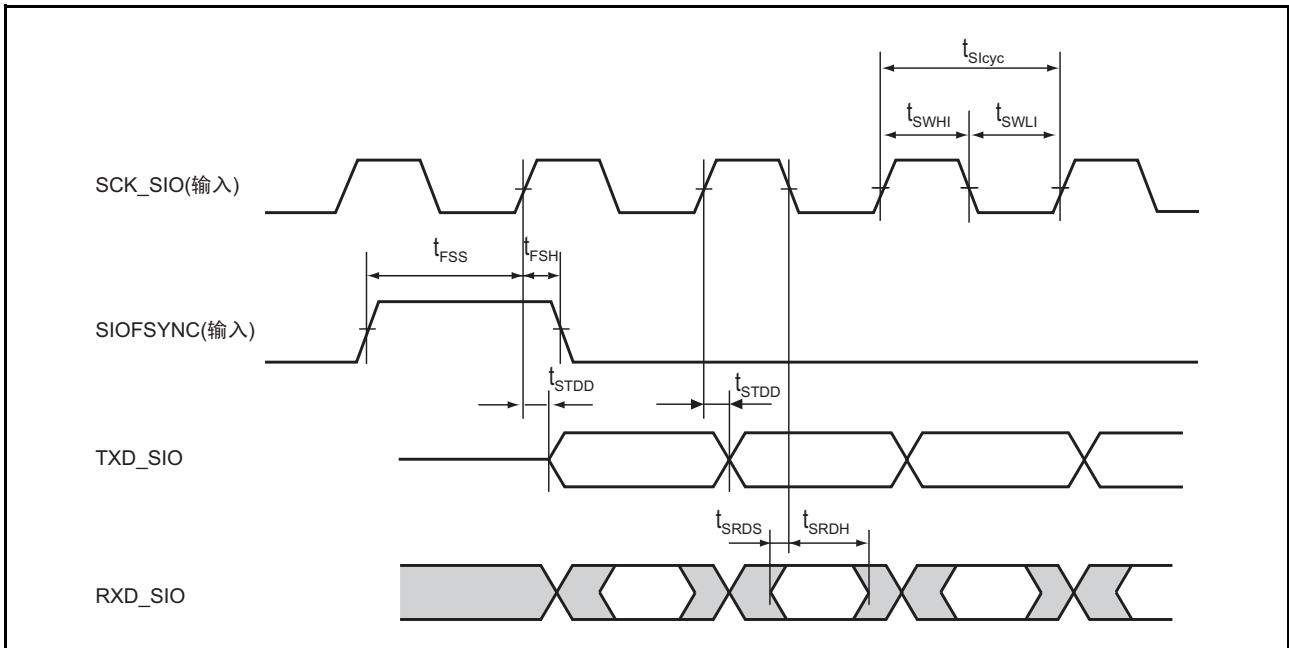


图 52.64 发送和接收时序（从属模式 1）

52.4.15 A/D 转换器的时序

表 52.19 A/D 转换器的时序

模块	项目	符号	Min.	Max.	单位	参照图
A/D 转换器	触发输入的准备时间	t_{TRGS}	17	—	ns	52.65

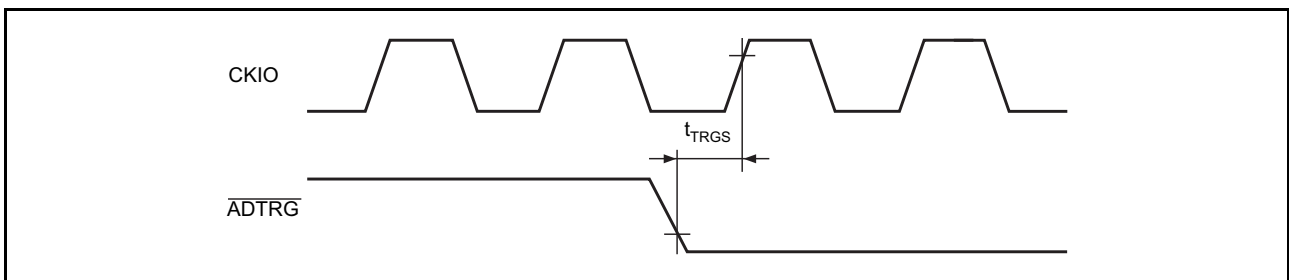


图 52.65 A/D 转换器的外部触发输入时序

52.4.16 NAND 闪存控制器的时序

表 52.20 NAND 闪存的接口的时序

项目	符号	Min.	Max.	单位	参照图
命令输出的准备时间	t_{NCDS}	$2 \times t_{\text{fcyc}} - 10$	—	ns	52.66、52.70
命令输出的保持时间	t_{NCDH}	$1.5 \times t_{\text{fcyc}} - 5$	—	ns	
数据输出的准备时间	t_{NDOS}	$0.5 \times t_{\text{wfcyc}} - 5$	—	ns	52.66、52.67、 52.69、52.70
数据输出的保持时间	t_{NDOH}	$0.5 \times t_{\text{wfcyc}} - 10$	—	ns	
命令 - 地址的转换时间 1	t_{NCAD1}	$1.5 \times t_{\text{fcyc}} - 10$	—	ns	52.66、52.67
命令 - 地址的转移时间 2	t_{NCAD2}	$2 \times t_{\text{fcyc}} - 10$	—	ns	52.67
$\overline{\text{FWE}}$ 周期时间	t_{NWC}	$t_{\text{wfcyc}} - 5$	—	ns	52.67、52.69
$\overline{\text{FEW}}$ 低电平脉宽	t_{NWP}	$0.5 \times t_{\text{wfcyc}} - 5$	—	ns	52.66、52.67、 52.69、52.70
$\overline{\text{FEW}}$ 高电平脉宽	t_{NWH}	$0.5 \times t_{\text{wfcyc}} - 5$	—	ns	
地址 - 就绪 / 忙转移时间	t_{NADRB}	—	$32 \times t_{\text{p0cyc}}$	ns	52.67、52.68
命令 - 就绪 / 忙转移时间	t_{NCDRB}	—	$10 \times t_{\text{p0cyc}}$	ns	52.67、52.68
就绪 / 忙 - 读数据的转移时间 1	t_{NRBDR1}	$1.5 \times t_{\text{fcyc}}$	—	ns	52.68
就绪 / 忙 - 读数据的转移时间 2	t_{NRBDR2}	$32 \times t_{\text{p0cyc}}$	—	ns	
$\overline{\text{FRE}}$ 周期时间	t_{NSCC}	$t_{\text{wfcyc}} - 5$	—	ns	
$\overline{\text{FRE}}$ 低电平脉宽	t_{NSP}	$0.5 \times t_{\text{wfcyc}} - 5$	—	ns	52.68、52.70
$\overline{\text{FRE}}$ 高电平脉宽	t_{NSPH}	$0.5 \times t_{\text{wfcyc}} - 5$	—	ns	52.68
读数据的准备时间	t_{NRDS}	16	—	ns	52.68、52.70
读数据的保持时间	t_{NRDH}	5	—	ns	52.68、52.70
编程数据的准备时间	t_{NDWS}	$32 \times t_{\text{p0cyc}}$	—	ns	52.69
命令 - 读状态的转移时间	t_{NCDSR}	$4 \times t_{\text{fcyc}}$	—	ns	52.70
命令输出 OFF- 读状态的转移时间	t_{NCDFSR}	$3.5 \times t_{\text{fcyc}}$	—	ns	
读状态的准备时间	t_{NSTS}	$2.5 \times t_{\text{fcyc}}$	—	ns	
$\overline{\text{FCE}}$ 输出的准备时间	t_{NCEs}	$8 \times t_{\text{p0cyc}}$	—	ns	52.66
$\overline{\text{FCE}}$ 输出的保持时间	t_{NCEH}	t_{p0cyc}	—	ns	52.69
$\overline{\text{FCE}}$ 输出的存取时间	t_{NCEA}	$6 \times t_{\text{p0cyc}}$	—	ns	52.68
$\overline{\text{FCE}}$ 输出的高电平保持时间	t_{NCEOH}	$2 \times t_{\text{p0cyc}}$	—	ns	

【注】 t_{fcyc} 为 1 个 FLCTL 时钟周期的时间。

当 NANDWF 位为“0”时， t_{wfcyc} 为 1 个 FLCTL 时钟周期的时间；当 NANDWF 位为“1”时， t_{wfcyc} 为 2 个 FLCTL 时钟周期的时间。

t_{p0cyc} 表示 1 个外围时钟（P0 ϕ ）周期的时间。

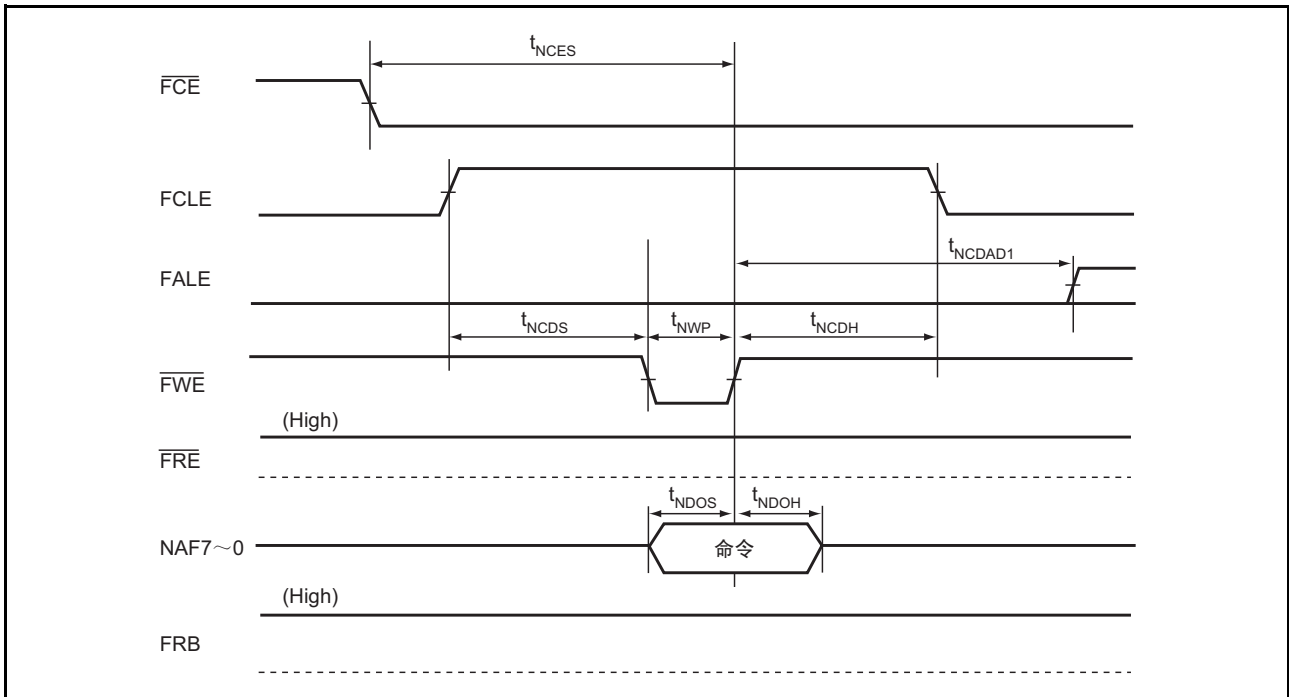


图 52.66 NAND 闪存 的命令发行时序

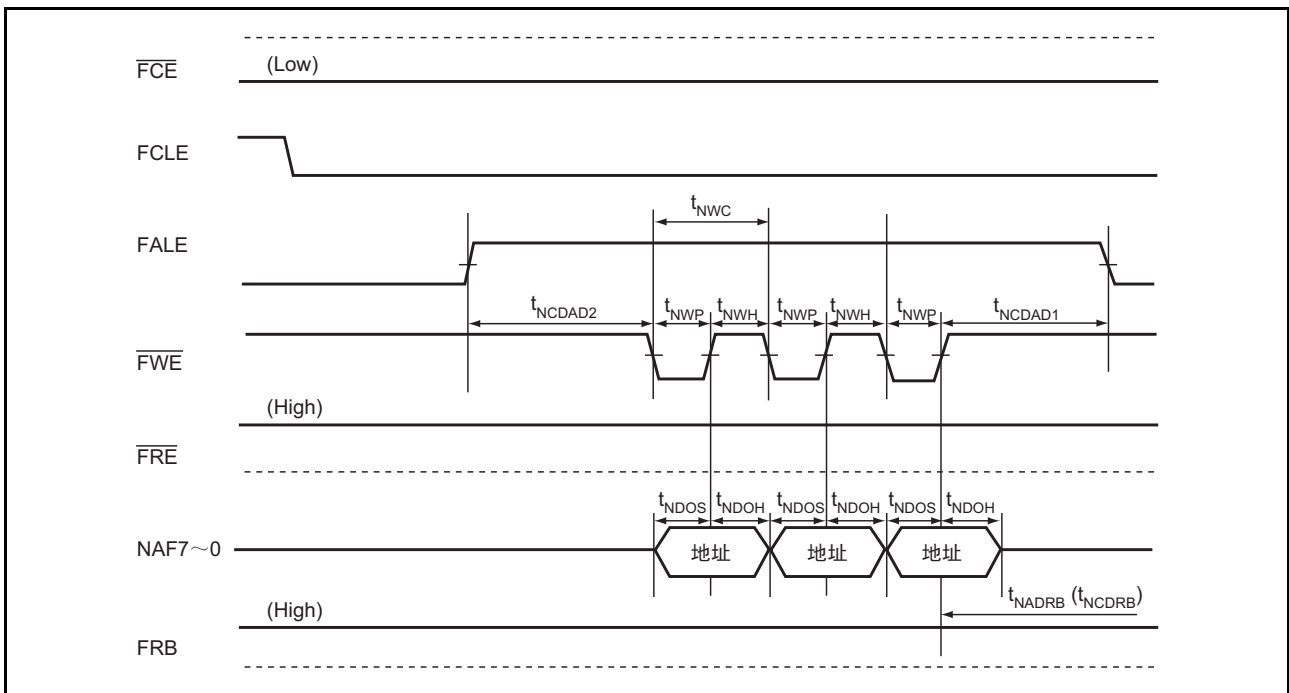


图 52.67 NAND 闪存的地址发行时序

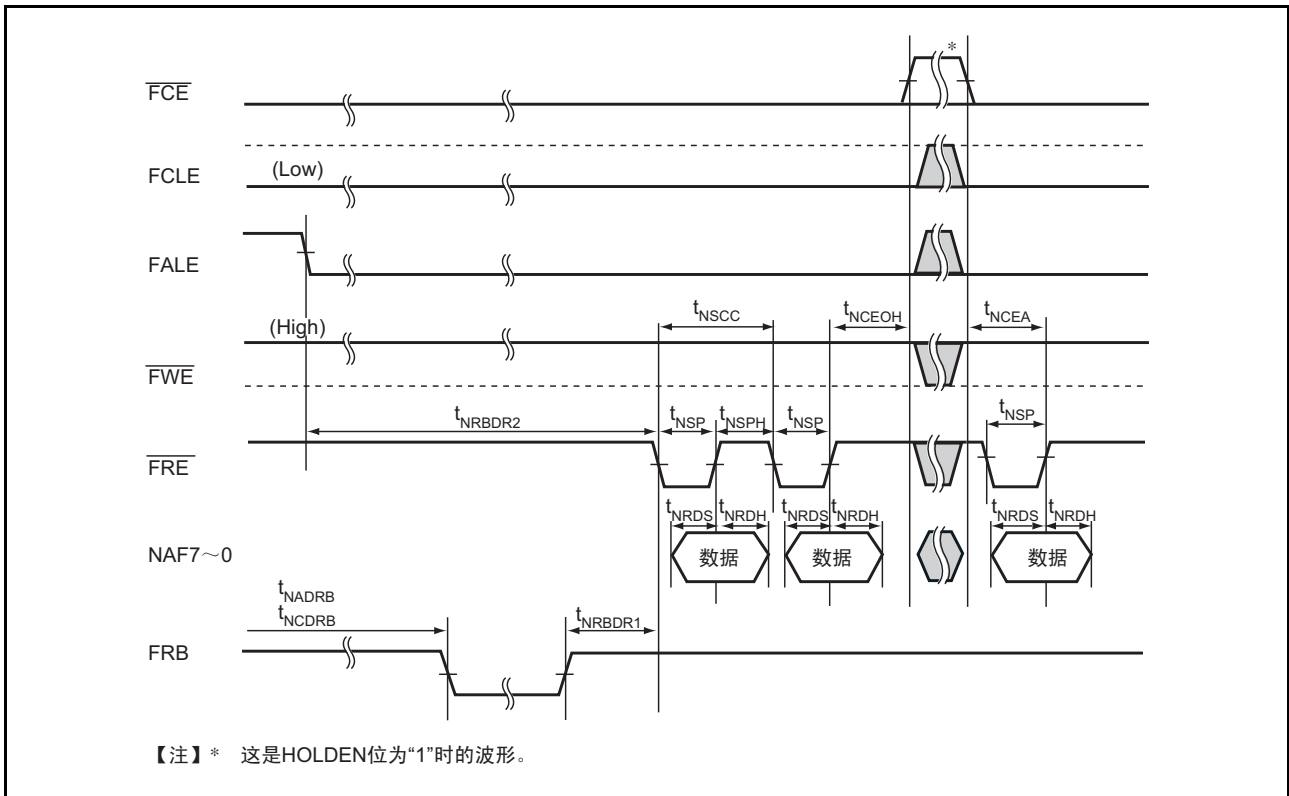


图 52.68 NAND 闪存的读数据时序

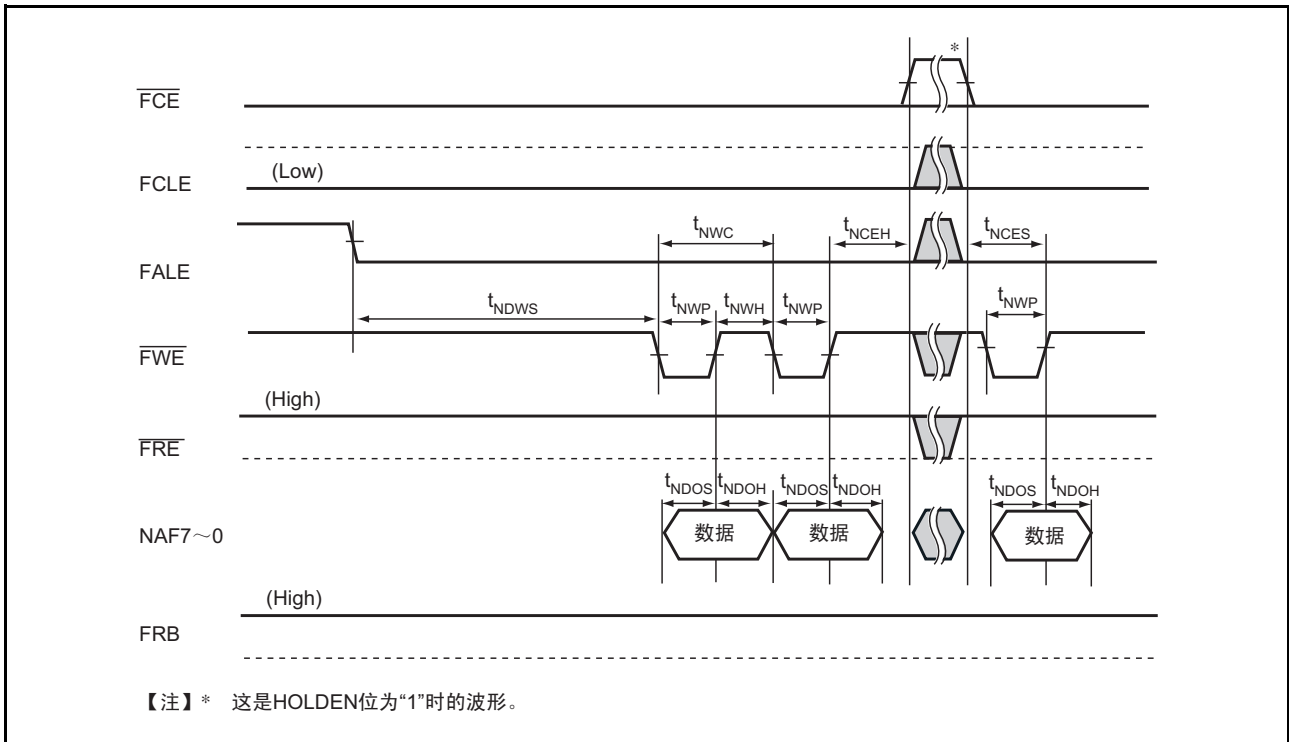


图 52.69 NAND 闪存的编程时序

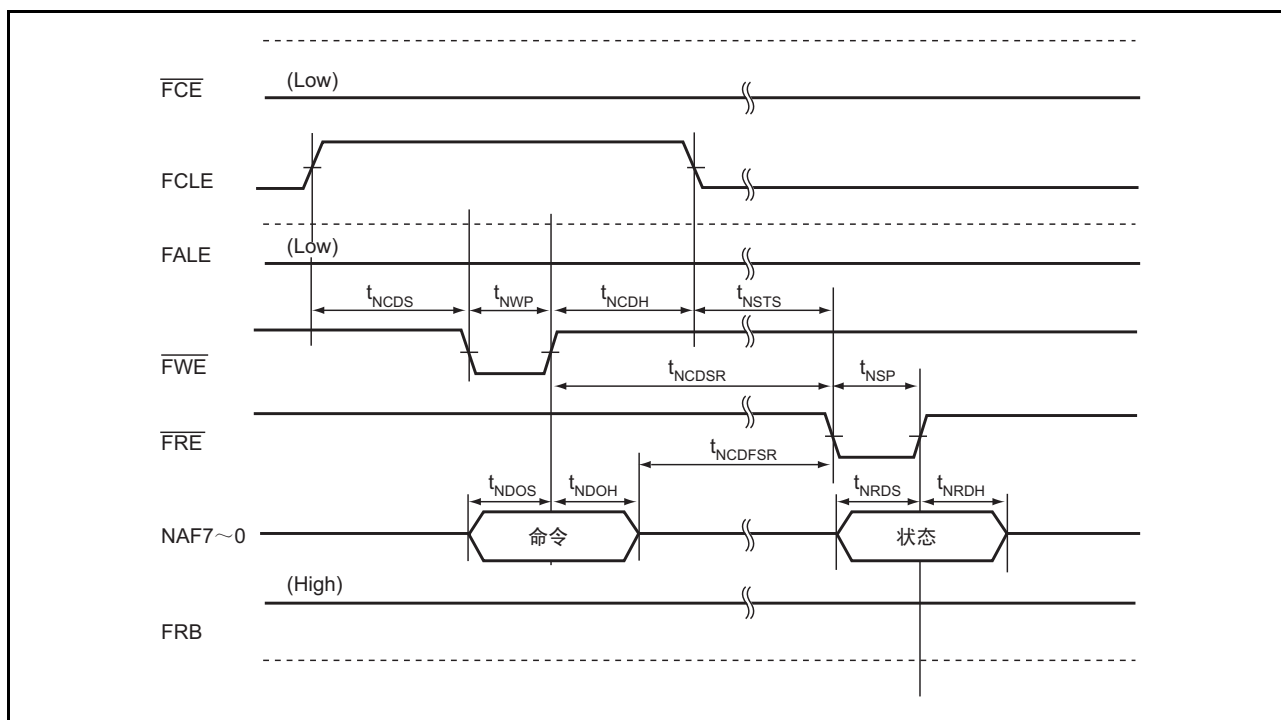


图 52.70 NAND 闪存的读状态时序

52.4.17 USB2.0 主机 / 功能模块的时序

表 52.21 USB 收发器的时序 (低速)

项目	符号	Min.	Typ.	Max.	单位	参照图
上升时间	t_{LR}	75	—	300	ns	52.71
下降时间	t_{LF}	75	—	300	ns	
上升 / 下降的时间比	t_{LR}/t_{LF}	80	—	125	%	

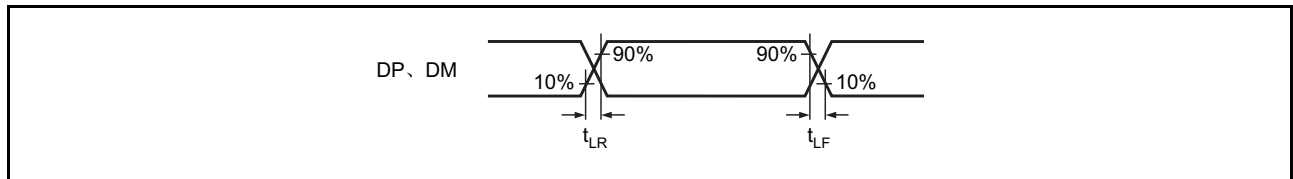


图 52.71 DP、DM 输出时序 (低速)

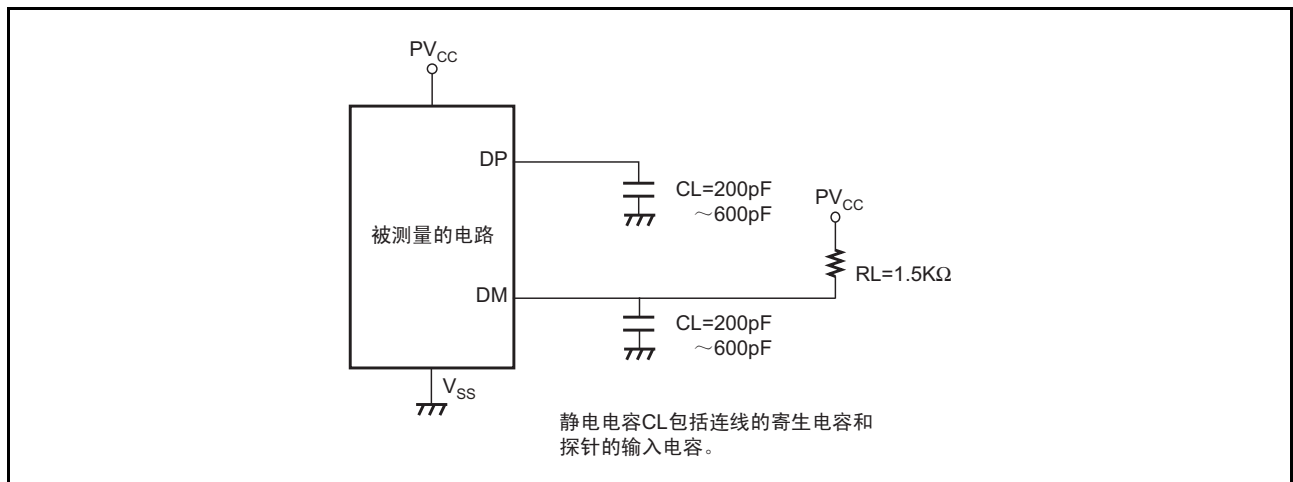


图 52.72 测量电路 (低速)

表 52.22 USB 收发器时序 (全速)

项目	符号	Min.	Typ.	Max.	单位	参照图
上升时间	t_{FR}	4	—	20	ns	52.73
下降时间	t_{FF}	4	—	20	ns	
上升 / 下降的时间比	t_{FR}/t_{FF}	90	—	111.11	%	

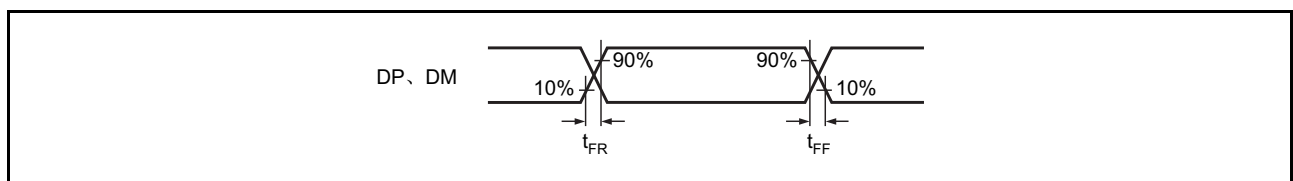


图 52.73 DP、DM 输出时序 (全速)

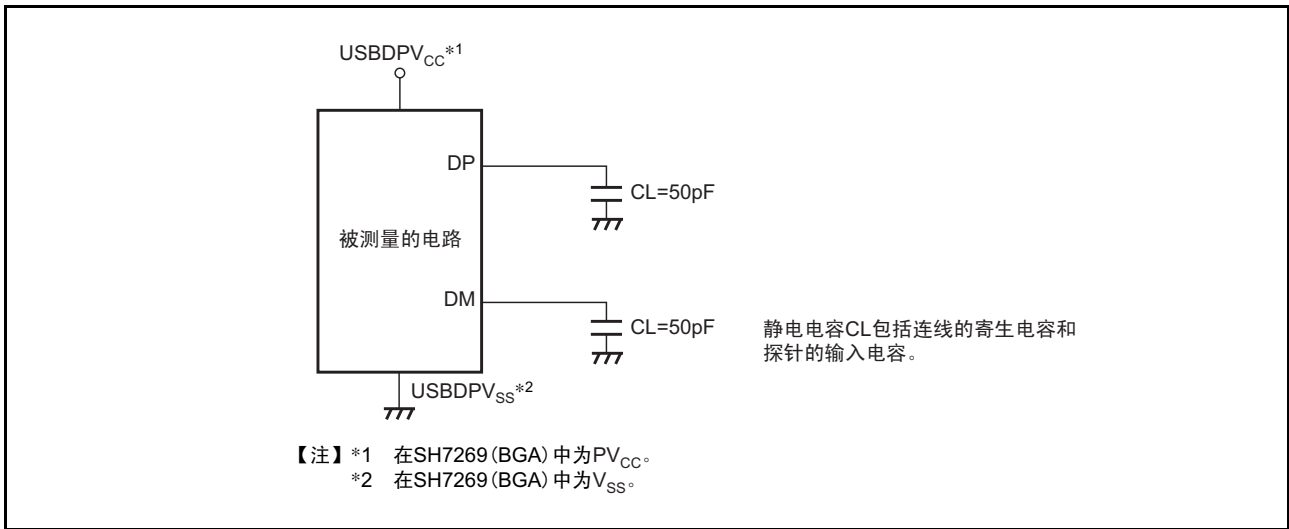


图 52.74 测量电路（全速时）

表 52.23 USB 收发器时序（高速）

项目	符号	Min.	Typ.	Max.	单位	参照图
上升时间	t _{HSR}	500	—	—	ps	52.75
下降时间	t _{HSF}	500	—	—	ps	
输出驱动电阻	Z _{HSDRV}	40.5	—	49.5	Ω	

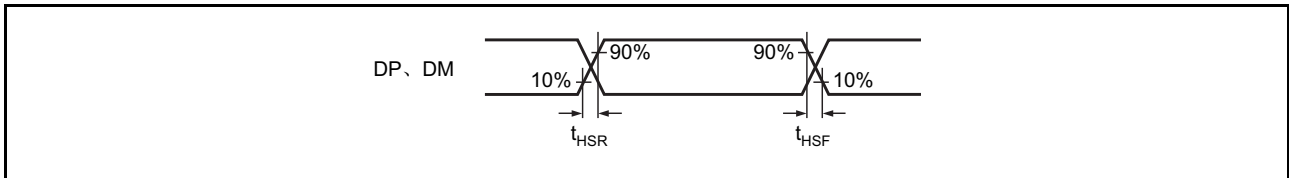


图 52.75 DP、DM 输出时序（高速）

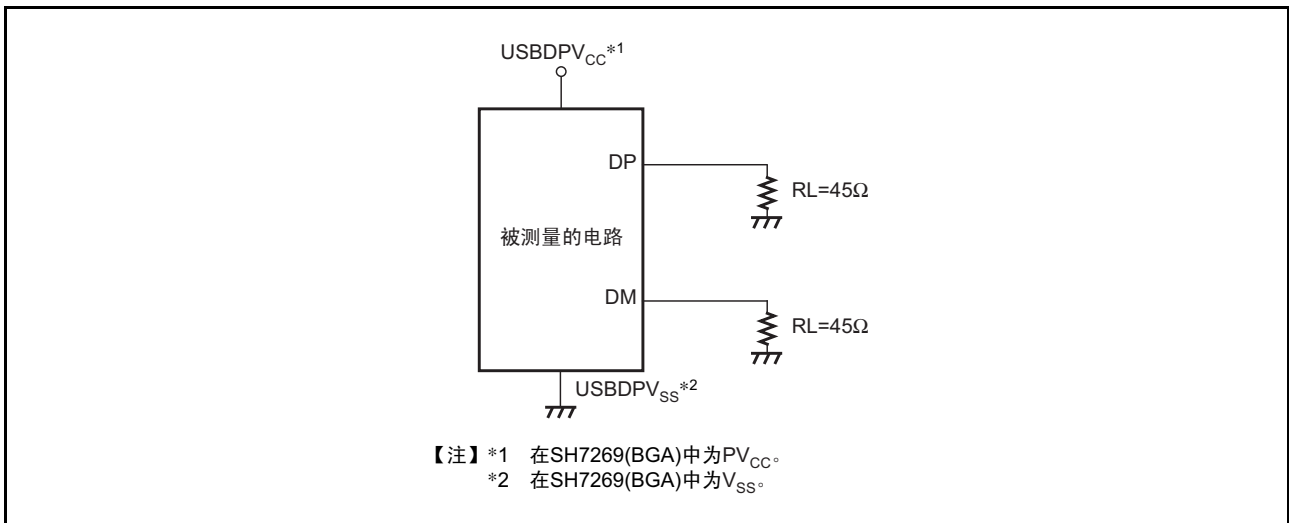


图 52.76 测量电路（高速）

52.4.18 视频显示控制器 4 的时序

表 52.24 视频显示控制器 4 的时序

项目	符号	Min.	Typ.	Max.	单位	参照图
DV_CLK 时钟输入频率	t_{Dcyc}	—	—	66.67	MHz	52.77
DV_CLK 时钟输入的低电平脉宽	t_{WIL}	0.4	—	—	t_{Dcyc}	
DV_CLK 时钟输入的高电平脉宽	t_{WH}	0.4	—	—	t_{Dcyc}	
LCD_EXTCLK 时钟输入频率	t_{Ecyc}	—	—	66.67	MHz	
LCD_EXTCLK 时钟输入的低电平脉宽	t_{WIL}	0.4	—	—	t_{Ecyc}	
LCD_EXTCLK 时钟输入的高电平脉宽	t_{WH}	0.4	—	—	t_{Ecyc}	
LCD_CLK 时钟输出频率	t_{Lcyc}	—	—	66.67	MHz	52.78
输入数据的准备时间	t_{VS}	4	—	—	ns	52.79
输入数据的保持时间	t_{VH}	4	—	—	ns	
输出数据的延迟时间	t_{DD}	-5	—	3	ns	52.80

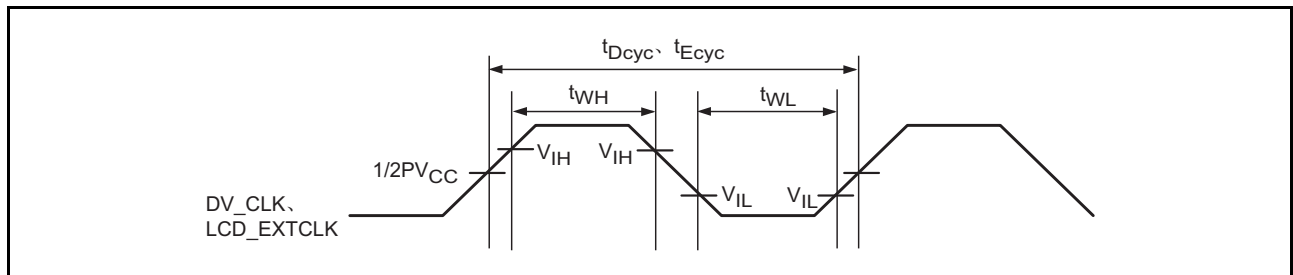


图 52.77 DV_CLK、LCD_EXTCLK 时钟输入时序

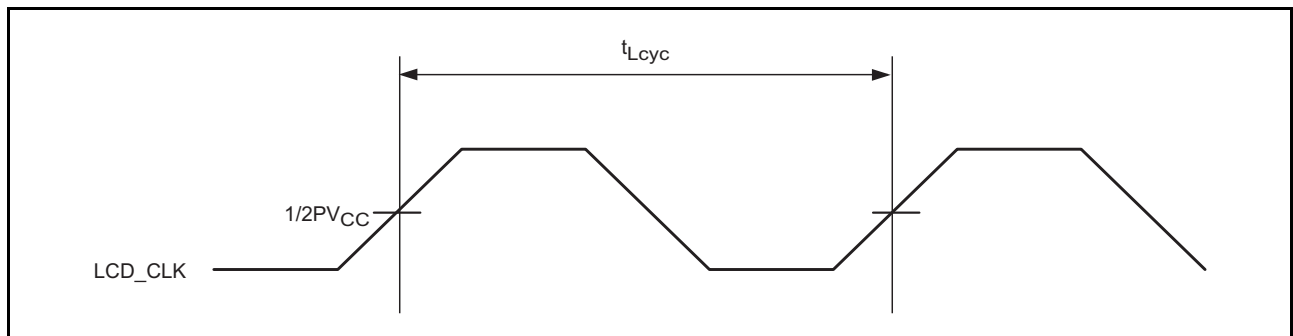


图 52.78 LCD_CLK 时钟输出时序

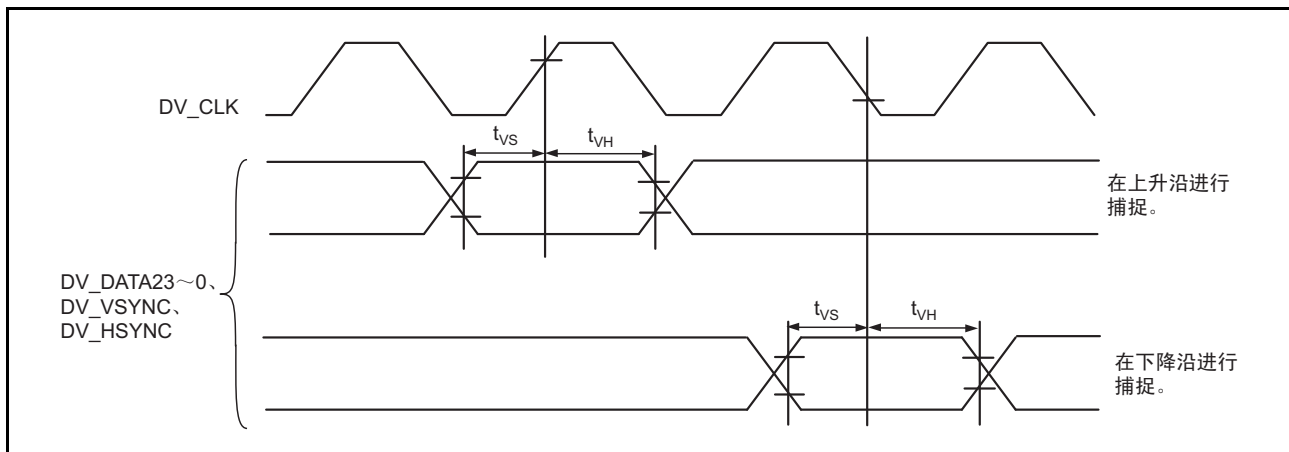


图 52.79 图像输入时序

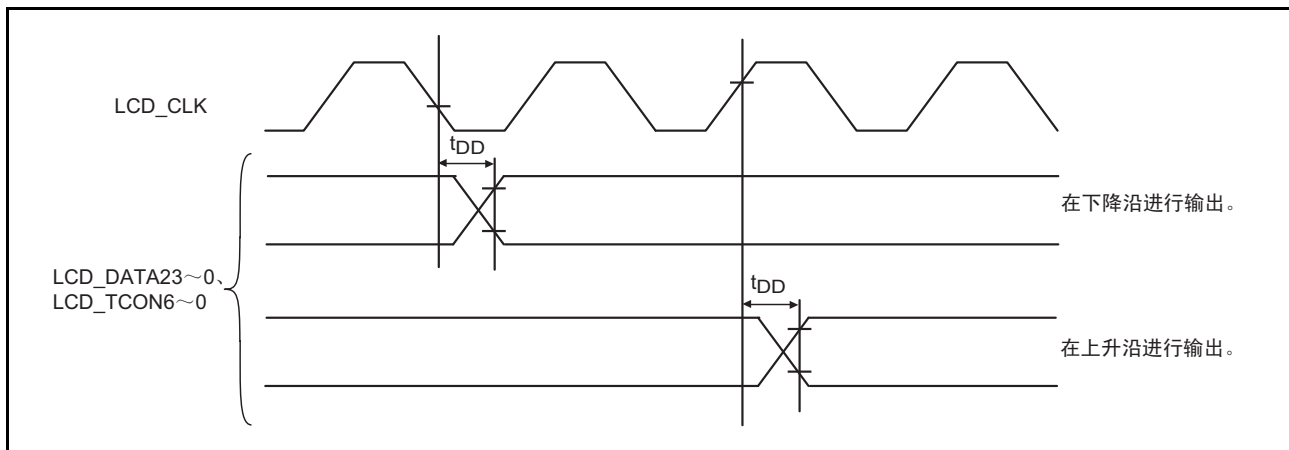


图 52.80 显示器输出时序

52.4.19 SD 主机接口的时序

表 52.25 SD 主机接口的时序

项目	符号	Min.	Max.	单位	参照图
SD_CLK 时钟周期	t_{SDPP}	$2 \times t_{p1cyc}$	—	ns	52.81
SD_CLK 时钟的高电平宽度	t_{SDWH}	$0.4 \times t_{SDPP}$	—	ns	
SD_CLK 时钟的低电平宽度	t_{SDWL}	$0.4 \times t_{SDPP}$	—	ns	
SD_CLK 时钟的上升时间	t_{SDLH}	—	3	ns	
SD_CLK 时钟的下降时间	t_{SDHL}	—	3	ns	
SD_CMD、SD_D3 ~ SD_D0 输出的数据延迟 (数据传送模式)	t_{SDODLY}	—	4	ns	
SD_CMD、SD_D3 ~ SD_D0 输入的数据准备	t_{SDISU}	5	—	ns	
SD_CMD、SD_D3 ~ SD_D0 输入的数据保持	t_{SDIH}	2	—	ns	

【注】 t_{p1cyc} 表示 1 个外围时钟 (P1 ϕ) 周期的时间。

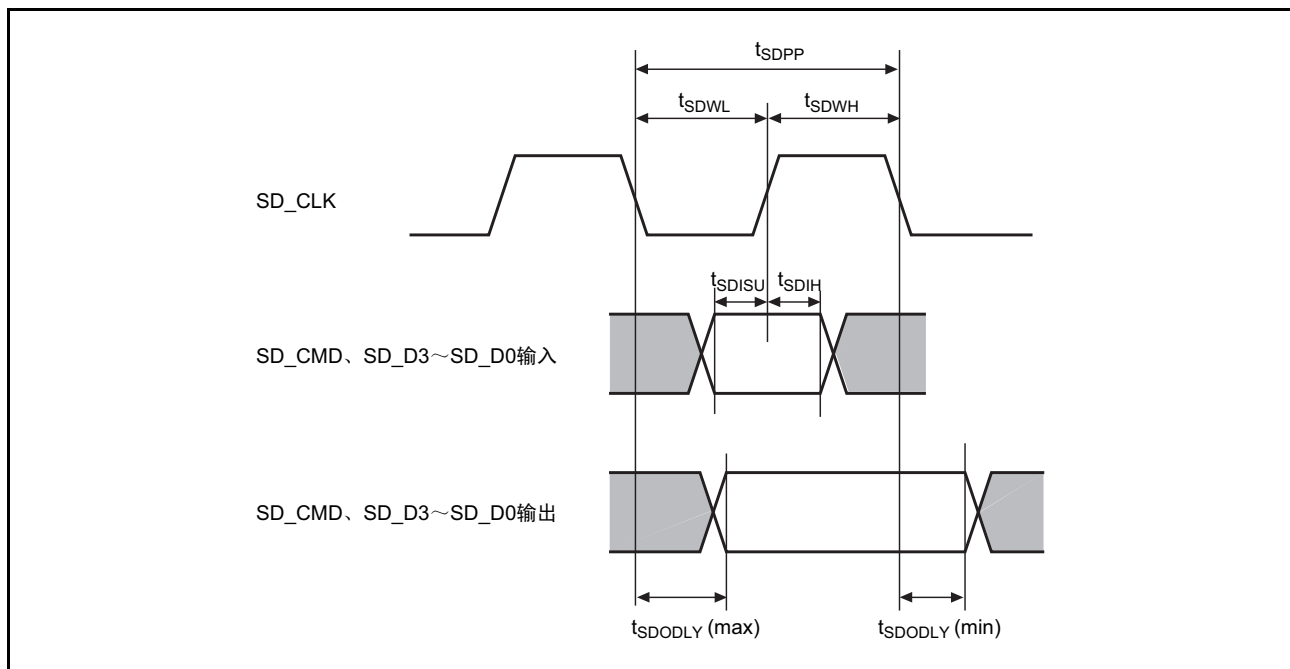


图 52.81 SD 卡接口

52.4.20 MMC 主机接口的时序

表 52.26 MMC 主机接口的时序

项目	符号	Min.	Max.	单位	参照图
MMC_CLK 时钟周期	$t_{\text{MMC}}^{\text{PP}}$	$2 \times t_{\text{p1cyc}}$	—	ns	52.82
MMC_CLK 时钟的高电平宽度	$t_{\text{MMC}}^{\text{WH}}$	6.5	—	ns	
MMC_CLK 时钟的低电平宽度	$t_{\text{MMC}}^{\text{WL}}$	6.5	—	ns	
MMC_CLK 时钟的上升时间	$t_{\text{MMC}}^{\text{LH}}$	—	3	ns	
MMC_CLK 时钟的下降时间	$t_{\text{MMC}}^{\text{HL}}$	—	3	ns	
MMC_CMD、MMC_D7 ~ MMC_D0 输出的数据延迟 (数据传送模式)	$t_{\text{MMC}}^{\text{ODLY}}$	-6.5	6.5	ns	
MMC_CMD、MMC_D7 ~ MMC_D0 输入的数据准备	$t_{\text{MMC}}^{\text{ISU}}$	4.5	—	ns	
MMC_CMD、MMC_D7 ~ MMC_D0 输入的数据保持	$t_{\text{MMC}}^{\text{IH}}$	2	—	ns	

【注】 t_{p1cyc} 表示 1 个外围时钟 (P1 ϕ) 周期的时间。

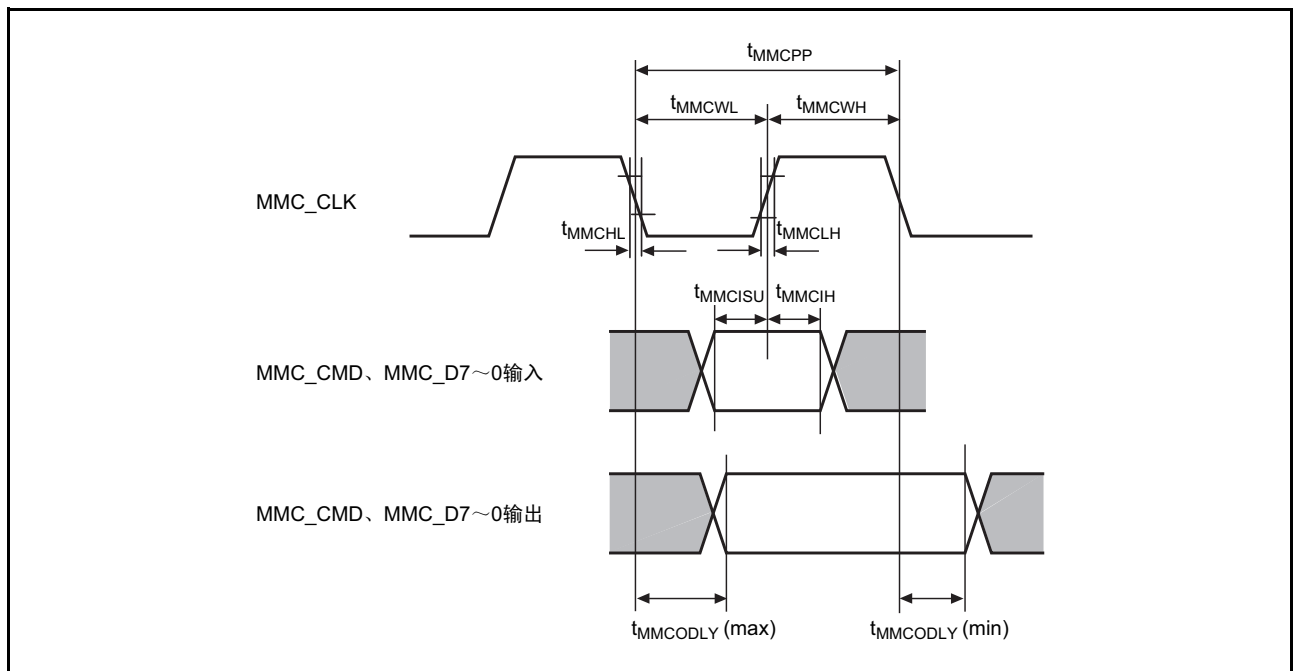


图 52.82 MMC 接口

52.4.21 通用输入 / 输出端口的时序

表 52.27 通用输入 / 输出端口的时序

项目	符号	Min.	Max.	单位	参照图
输出数据的延迟时间	t_{PORTD}	—	100	ns	52.83
输入数据的准备时间	t_{PORTS}	100	—		
输入数据的保持时间	t_{PORTH}	100	—		

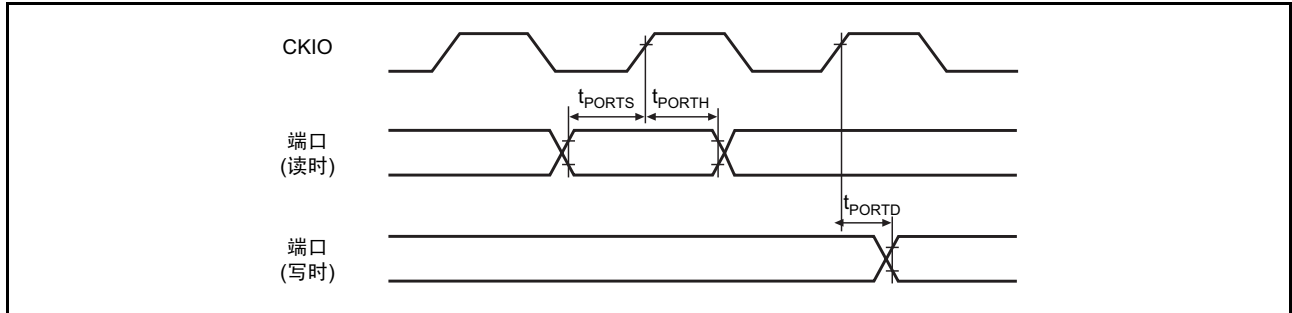


图 52.83 通用输入 / 输出端口的时序

52.4.22 用户调试接口的时序

表 52.28 用户调试接口的时序

项目	符号	Min.	Max.	单位	参照图
TCK 周期时间	t_{TCKcyc}	50*	—	ns	52.84
TCK 高电平脉宽	t_{TCKH}	0.4	0.6	t_{TCKcyc}	
TCK 低电平脉宽	t_{TCKL}	0.4	0.6	t_{TCKcyc}	
TDI 准备时间	t_{TDIS}	10	—	ns	52.85
TDI 保持时间	t_{TDIH}	10	—	ns	
TMS 准备时间	t_{TMSS}	10	—	ns	
TMS 保持时间	t_{TMSH}	10	—	ns	
TDO 延迟时间	t_{TDOD}	—	16	ns	52.86
捕捉寄存器的准备时间	t_{CAPTS}	10	—	ns	
捕捉寄存器的保持时间	t_{CAPTH}	10	—	ns	
更新寄存器的延迟时间	t_{UPDATED}	—	20	ns	

【注】 * 必须设定为大于外围时钟（ $P0\phi$ ）的周期时间。

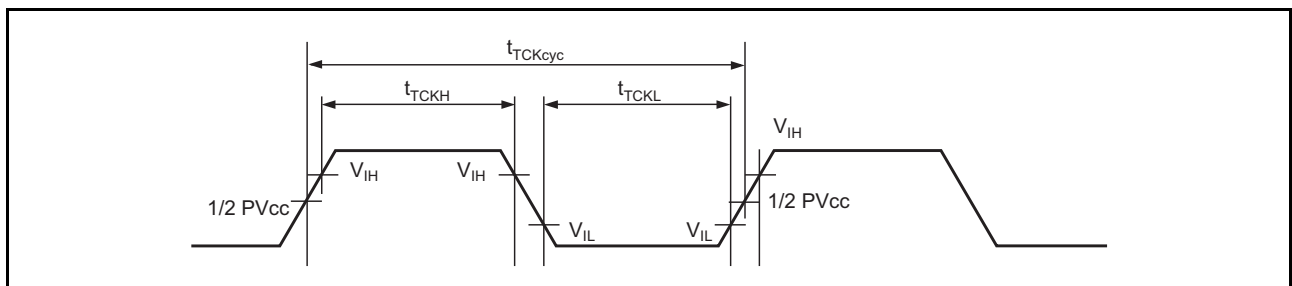


图 52.84 TCK 输入时序

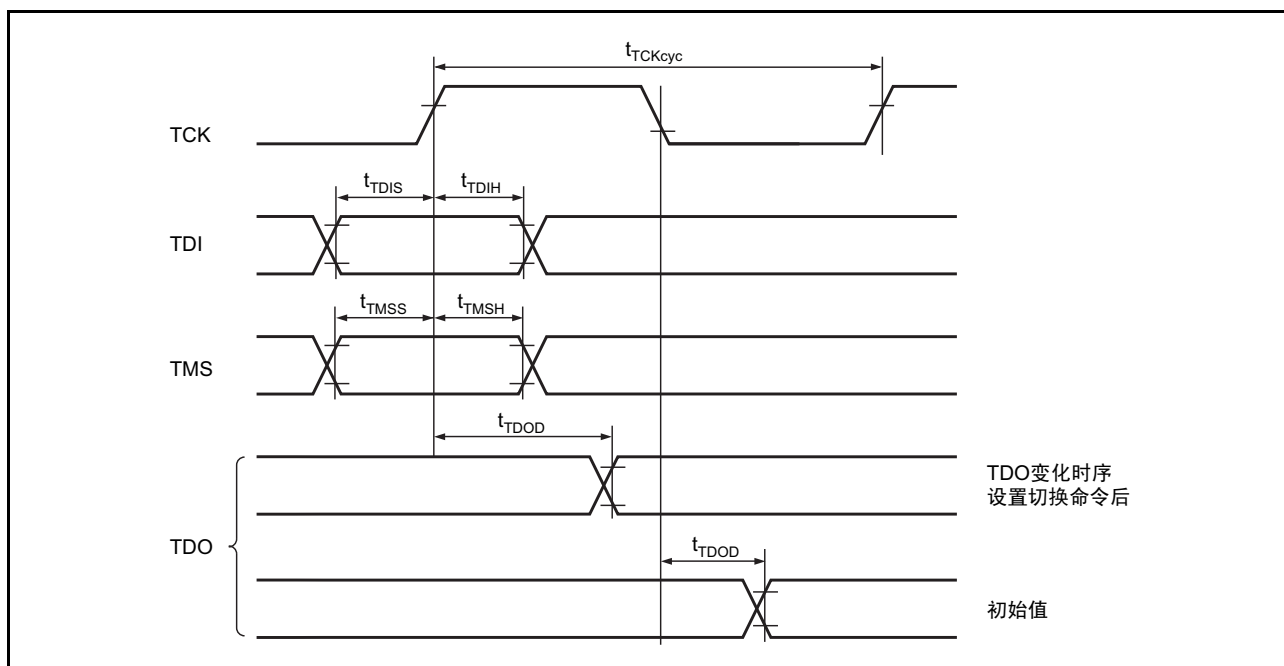


图 52.85 数据传送时序

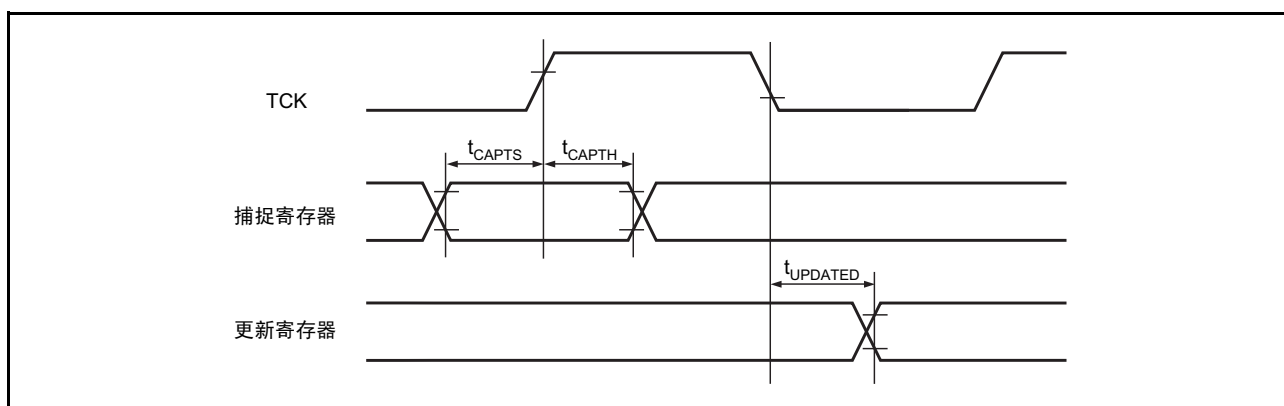


图 52.86 边界扫描输入 / 输出时序

52.4.23 AC 特性的测量条件

- 输入/输出信号的基准电平： $PV_{CC}/2$ ($PV_{CC}=3.0\sim 3.6V$ 、 $V_{CC}=1.15\sim 1.35V$)
- 输入脉冲电平： PV_{CC}
- 输入上升时间和下降时间：1ns

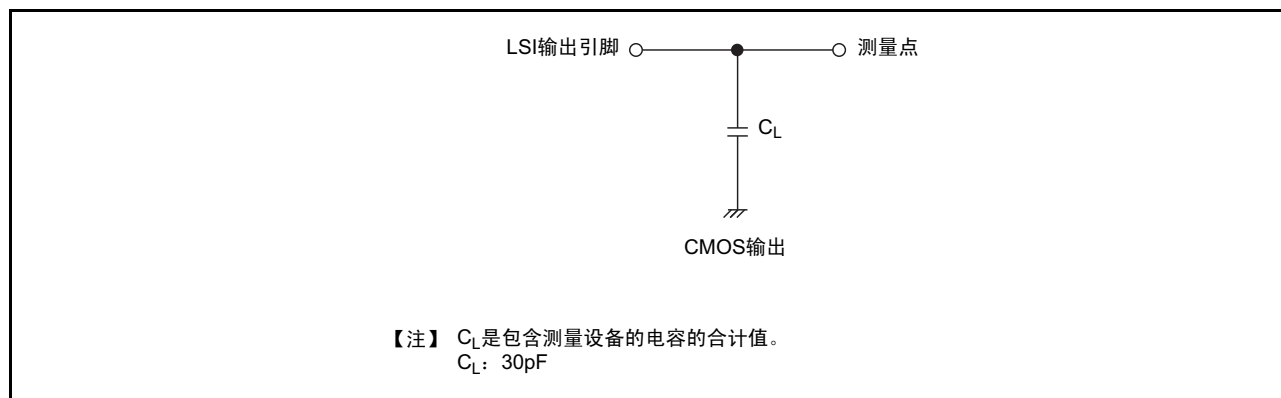


图 52.87 输出负荷电路

52.5 A/D 转换器特性

条件： $V_{CC}=PLL_{V_{CC}}=USB_{DV_{CC}}=USB_{UV_{CC}}=1.15\sim 1.35V$ ， $PV_{CC}=USB_{DPV_{CC}}=3.0\sim 3.6V$ ，
 $AV_{CC}=3.0\sim 3.6V$ ， $USB_{AV_{CC}}=1.15\sim 1.35V$ ， $USB_{APV_{CC}}=3.0\sim 3.6V$ ， $V_{DAV_{CC}}=3.0\sim 3.6V$ ，
 $V_{SS}=PLL_{V_{SS}}=AV_{SS}=USB_{DV_{SS}}=USB_{AV_{SS}}=USB_{DPV_{SS}}=USB_{APV_{SS}}=USB_{UV_{SS}}=V_{DAV_{SS}}=0V$ ，
 $T_a=-20\sim 85^\circ C$ （标准温度范围）、 $-40\sim 85^\circ C$ （大温度范围）

【注】 SH7269（BGA）没有 $USB_{DV_{CC}}$ 、 $USB_{UV_{CC}}$ 、 $USB_{DPV_{CC}}$ 、 $PLL_{V_{SS}}$ 、 $USB_{DV_{SS}}$ 、 $USB_{AV_{SS}}$ 、 $USB_{UV_{SS}}$ 、 $USB_{DPV_{SS}}$ 、 $USB_{APV_{SS}}$ 引脚。

表 52.29 A/D 转换器特性

项目	Min.	Typ.	Max.	单位
分辨率	10	10	10	位
转换时间	6	—	—	μs
模拟输入电容	—	—	20	pF
容许信号源阻抗	—	—	5	k Ω
非线性误差	—	—	$\pm 3.0^*$	LSB
偏移误差	—	—	$\pm 2.0^*$	LSB
满刻度误差	—	—	$\pm 2.0^*$	LSB
量化误差	—	—	$\pm 0.5^*$	LSB
绝对精度	—	—	± 5.0	LSB

【注】 * 参考值

52.6 用于输入视频信号的 A/D 转换器特性

条件: $V_{CC}=PLL_{VCC}=USBDV_{CC}=USBV_{CC}=1.15 \sim 1.35V$, $PV_{CC}=USBDPV_{CC}=3.0 \sim 3.6V$,
 $AV_{CC}=3.0 \sim 3.6V$, $USBAV_{CC}=1.15 \sim 1.35V$, $USBAPV_{CC}=3.0 \sim 3.6V$, $VDAV_{CC}=3.0 \sim 3.6V$,
 $V_{SS}=PLL_{VSS}=AV_{SS}=USBDV_{SS}=USBAV_{SS}=USBDPV_{SS}=USBAPV_{SS}=USBV_{SS}=VDAV_{SS}=0V$,
 $T_a=-20 \sim 85^\circ C$ (标准温度范围)、 $-40 \sim 85^\circ C$ (大温度范围)

【注】 SH7269 (BGA) 没有 $USBDV_{CC}$ 、 $USBV_{CC}$ 、 $USBDPV_{CC}$ 、 PLL_{VSS} 、 $USBDV_{SS}$ 、 $USBAV_{SS}$ 、 $USBV_{SS}$ 、 $USBDPV_{SS}$ 、 $USBAPV_{SS}$ 引脚。

表 52.30 用于输入视频信号的 A/D 转换器特性 (基准电压)

项目	Min.	Typ.	Max.	单位	测量条件
基准电压 (Top)	—	2.0	—	V	
基准电压 (Bottom)	—	1.0	—	V	

表 52.31 用于输入视频信号的 A/D 转换器特性 (钳位)

项目	Min.	Typ.	Max.	单位	测量条件
钳位电压电平	—	1.0	—	V	VRB 电压钳位
吸收电流	—	10	—	μA	
源电流	—	1.0	—	mA	

表 52.32 用于输入视频信号的 A/D 转换器特性 (PGA)

项目	Min.	Typ.	Max.	单位	测量条件
增益阶梯数	—	32	—	step	
增益阶梯宽度	—	0.2	—	dB	
最小增益	—	1.835	—	dB	
最大增益	—	8.023	—	dB	

表 52.33 用于输入视频信号的 A/D 转换器特性 (ADC)

项目	Min.	Typ.	Max.	单位	测量条件
分辨率	—	10	—	位	
A/D 范围	—	2.0	—	V _{pp}	(VRT-VRB)×2
积分线性误差	—	—	±5.0	LSB	ADC+PGA fs=27MHz
微分线性误差	—	—	±2.0	LSB	ADC+PGA fs=27MHz
S/N	—	54*	—	dB	fin=1MHz、fs=27MHz PGA_GAIN=01000
S/(N+D)	—	51*	—	dB	fin=1MHz、fs=27MHz PGA_GAIN=01000

【注】 * 参考值

53. 引脚状态和处理方法

本章节说明各运行模式中的引脚状态和处理方法。

53.1 引脚状态

各运行模式中的引脚状态如表 53.1 所示。

有关输入 / 输出功能，上段记载的是输入缓冲器状态，下段记载的是输出缓冲器状态。

在总线权释放的状态下正规状态和引脚状态不同的引脚如表 53.2 所示。

表 53.1 引脚状态

引脚功能			引脚状态						
分类	引脚名		正规状态 (右记以外)	上电复位 *1	引脚状态的保持 *2		低功耗状态		
					EBUSKEEPE*3 (右记以外)		上电复位 *4	深度待机 模式	软件待机 模式
					0	1			
时钟	EXTAL *6		I	I	I		I/Z*5	I	
	XTAL *6		O	O	O		O/L*5	O/L*5	
	CKIO		O/Z*7	O	O	O/Z*7	O/Z*7	O/Z*7	
	AUDIO_CLK		I	—	—		Z	Z	
	AUDIO_X1*6		I/Z*8	I	I		Z	Z	
	AUDIO_X2*6		O/L*8	O	O		L	L	
	AUDIO_XOUT		O/L*8	—	O/Z*9*16		O/Z*9*16	L/Z*9	
系统控制	RES		I	I	I		I	I	
	WDTOVF		O	—	H		H	H	
	BREQ		I	—	—		Z	Z	
	BACK		O	—	Z		Z	Z	
运行模式控制	MD_BOOT2 ~ MD_BOOT0		—	I	—		—	—	
	MD_CLK0		—	I	—		—	—	
	ASEMD		I	I	I		I	I	
中断	NMI		I	I	I		I	I	
	IRQ7 (PF19)、IRQ6 (PF18)、 IRQ5 (PF17)、IRQ4 (PF16)、 IRQ3、IRQ2、IRQ1 (PC7、 PJ21)、IRQ0 (PC5、PJ20)		I	—	—		I/Z*12	I	
	IRQ7 (PG7)、IRQ6 (PG6)、 IRQ5 (PG5)、IRQ4 (PG4)、 IRQ1 (PG1)、IRQ0 (PG0)		I	—	—		Z	I	
	PINT7 ~ PINT0		I	—	—		Z	Z	
用户断点 控制器	UBCTRG		O	—	O/Z*9		O/Z*9	O/Z*9	
地址总线	A25 ~ A21、A0		O	—	O/Z*10		O/Z*10	O/Z*10	
	A20 ~ A2	引导模式	0、1	O	Z	O	O/Z*10	O/Z*10	O/Z*10
			2 ~ 5	O	—	O/Z*10		O/Z*10	O/Z*10
	A1	引导模式	0	O	Z	O	O/Z*10	O/Z*10	O/Z*10
			1 ~ 5	O	—	O/Z*10		O/Z*10	O/Z*10

引脚功能				引脚状态						
分类	引脚名			正规状态 (右记以外)	上电复位 *1	引脚状态的保持 *2		低功耗状态		
						EBUSKEEPE*3 (右记以外)		上电复位 *4	深度待机 模式	软件待机 模式
						0	1			
数据总线	D15 ~ D0	引导模式	0、1	I/Z	Z	I/Z		Z	Z	
				O/Z	Z	O/Z	Z	Z	Z	
		2 ~ 5	I/Z	—	—		Z	Z		
			O/Z	—	Z		Z	Z		
	D31 ~ D16	引导模式	1	I/Z	Z	I/Z		Z	Z	
				O/Z	Z	O/Z	Z	Z	Z	
		0、2 ~ 5	I/Z	—	—		Z	Z		
			O/Z	—	Z		Z	Z		
总线控制	CS0	引导模式	0、1	O	Z	O	H/Z*10	H/Z*10	H/Z*10	
			2 ~ 5	O	—	H/Z*10		H/Z*10	H/Z*10	
	CS5 ~ CS1、CE1A、CE2A			O	—	H/Z*10		H/Z*10	H/Z*10	
	RD	引导模式	0、1	O	Z	O	H/Z*10	H/Z*10	H/Z*10	
			2 ~ 5	O	—	H/Z*10		H/Z*10	H/Z*10	
	RD/WR			O	—	H/Z*10		H/Z*10	H/Z*10	
	BS			O	—	H/Z*10		H/Z*10	H/Z*10	
	WAIT			I	—	—		Z	Z	
	WE3/ICIORW/AH/DQMUU、 WE2/ICIORW/DQMUL、 WE1/DQMLU/WE、WE0/DQMLL			O	—	H/Z*10		H/Z*10	H/Z*10	
	RAS、CAS			O	—	O/Z*11		O/Z*11	O/Z*11	
	CKE			O	—	O/Z*11		O/Z*11	O/Z*11	
	IOIS16			I	—	—		Z	Z	
直接存储器存取控制器	DREQ0			I	—	—		Z	Z	
	DACK0			O	—	O/Z*9		O/Z*9	O/Z*9	
	TEND0			O	—	O/Z*9		O/Z*9	O/Z*9	
多功能定时器脉冲单元 2	TCLKA ~ TCLKD			I	—	—		Z	Z	
	TIOC0A、TIOC0B、TIOC0C (PB3、PJ18)、TIOC0D (PB4、P19)、 TIOC1A、TIOC1B、TIOC2A、 TIOC2B、TIOC3A ~ TIOC3D、 TIOC4A ~ TIOC4D			I	—	—		Z	Z	
				O/Z	—	O/Z*9		O/Z*9	O/Z*9	
	TIOC0C (PG3)、TIOC0D (PG4)			I	—	—		I/Z*12	I	
			O/Z	—	O/Z*9		O/Z*9	O/Z*9		
实时时钟	RTC_X1*6			I/Z*13	I	I/Z*13		I/Z*13	I/Z*13	
	RTC_X2*6			O/H*13	O	O/H*13		O/H*13	O/H*13	
FIFO 内置型 串行通信接口	TxD7 ~ TxD0			O/Z	—	O/Z*9		O/Z*9	O/Z*9	
	RxD7 (PE7、PJ25)、RxD6 ~ RxD1			I	—	—		Z	Z	
	RxD7 (PC7)			I	—	—		I/Z*12	I	
	SCK7 ~ SCK0			I	—	—		Z	Z	
				O/Z	—	O/Z*9		O/Z*9	O/Z*9	

引脚功能			引脚状态						
分类	引脚名		正规状态 (右记以外)	上电复位 *1	引脚状态的保持 *2		低功耗状态		
					EBUSKEEPE*3 (右记以外)		上电复位 *4	深度待机 模式	软件待机 模式
					0	1			
FIFO 内置型 串行通信接口	RTS7、RTS5、RTS1		I	—	—		Z	Z	
			O/Z	—	O/Z*9		O/Z*9	O/Z*9	
	CTS7、CTS5、CTS1		I	—	—		Z	Z	
			O/Z	—	O/Z*9		O/Z*9	O/Z*9	
瑞萨串行外围 接口	MISO1、MISO0 (PJ19)		I	—	—		Z	Z	
	O/Z		—	O/Z*9		O/Z*9	O/Z*9		
	MISO0 (PB20)		I	—	—		Z	Z	
	引导模式	2、4、5	O/Z	—	O/Z*9		O/Z*9	O/Z*9	
		3	O/Z	—	—	O/Z*9	O/Z*9	O/Z*9	
	MOSI1、MOSI0 (PJ18)		I	—	—		Z	Z	
	O/Z		—	O/Z*9		O/Z*9	O/Z*9		
	MOSI0 (PB19)		I	—	—		Z	Z	
	引导模式	2、4、5	O/Z	—	O/Z*9		O/Z*9	O/Z*9	
		3	O/Z	—	—	O/Z*9	O/Z*9	O/Z*9	
	RSPCK1、RSPCK0 (PJ16)		I	—	—		Z	Z	
	O/Z		—	O/Z*9		O/Z*9	O/Z*9		
	RSPCK0 (PB17)		I	—	—		Z	Z	
	引导模式	2、4、5	O/Z	—	O/Z*9		O/Z*9	O/Z*9	
		3	O/Z	—	—	O/Z*9	O/Z*9	O/Z*9	
	SSL10、SSL00 (PJ17)		I	—	—		Z	Z	
O/Z		—	O/Z*9		O/Z*9	O/Z*9			
SSL00 (PB18)		I	—	—		Z	Z		
引导模式	2、4、5	O/Z	—	O/Z*9		O/Z*9	O/Z*9		
	3	O/Z	—	—	O/Z*9	O/Z*9	O/Z*9		
瑞萨四路串行 外围接口	QIO3_1、QIO3_0、 QIO2_1、QIO2_0、 QMI_1/QIO1_1、QMI_1/QIO0_0、 QMO_1/QIO0_1、QMO_0/QIO0_0		I	—	—		Z	Z	
	O/Z		—	O/Z*9		O/Z*9	O/Z*9		
	QSPCLK_1、QSPCLK_0		O/Z	—	O/Z*9		O/Z*9	O/Z*9	
	QSSL_1、QSSL_0		O/Z	—	O/Z*9		O/Z*9	O/Z*9	
SPI 多 I/O 总 线控制器	SPBIO3_1、SPBIO3_0、 SPBIO2_1、SPBIO2_0、 SPBMI_1/SPBIO1_1、SPBMI_0/ SPBIO1_0、 SPBMO_1/SPBIO0_1、 SPBMO_0/SPBIO0_0		I	—	—		Z	Z	
	O/Z		—	O/Z*9		O/Z*9	O/Z*9		
	SPBCLK、SPBSSL		O/Z	—	O/Z*9		O/Z*9	O/Z*9	
I ² C 总线接口 3	SCL3 ~ SCL0		I	—	—		Z	Z	
			O/Z	—	Z		Z	Z	
	SDA3 ~ SDA0		I	—	—		Z	Z	
			O/Z	—	Z		Z	Z	

引脚功能				引脚状态						
分类	引脚名			正规状态 (右记以外)	上电复位 *1	引脚状态的保持 *2		低功耗状态		
						EBUSKEEPE*3 (右记以外)		上电复位 *4	深度待机 模式	软件待机 模式
						0	1			
串行音频接口	SSITxD0			O	—	O/Z*9		O/Z*9	O/Z*9	
	SSIRxD0			I	—	—		Z	Z	
	SSIDATA5 ~ SSIDATA1			I	—	—		Z	Z	
				O/Z	—	O/Z*9		O/Z*9	O/Z*9	
	SSISCK5、SSISCK4、SSISCK2 ~ SSISCK0			I	—	—		Z	Z	
				O/Z	—	O/Z*9		O/Z*9	O/Z*9	
	SSISCK3			I	—	—		I/Z*12	I	
				O/Z	—	O/Z*9		O/Z*9	O/Z*9	
SSIWS5、SSIWS4、SSIWS2 ~ SSIWS0			I	—	—		Z	Z		
			O/Z	—	O/Z*9		O/Z*9	O/Z*9		
SSIWS3			I	—	—		I/Z*12	I		
			O/Z	—	O/Z*9		O/Z*9	O/Z*9		
带FIFO的时钟 同步串行 I/O	SIOFCK			I	—	—		Z	Z	
				O/Z	—	O/Z*9		O/Z*9	O/Z*9	
	SIOFSYNC			I	—	—		Z	Z	
				O/Z	—	O/Z*9		O/Z*9	O/Z*9	
	SIOFTxD			O/Z	—	O/Z*9		O/Z*9	O/Z*9	
SIOFRxD			I	—	—		Z	Z		
控制器局域网	CTx2 ~ CTx0			O	—	O/Z*9		O/Z*9	O/Z*9	
	CRx2 (PJ20)、CRx1、CRx0			I	—	—		I/Z*12	I	
	CRx2 (PB21)			I	—	—		Z	Z	
IEBus™ 控制 器	IETxD			O	—	O/Z*9		O/Z*9	O/Z*9	
	IERxD			I	—	—		Z	Z	
瑞萨 SPDIF 接口	SPDIF_OUT			O	—	O/Z*9		O/Z*9	O/Z*9	
	SPDIF_IN			I	—	—		Z	Z	
A/D 转换器	AN7 ~ AN0			I	—	—		Z	Z	
	ADTRG			I	—	—		Z	Z	
NAND 闪存控 制器	FRB			I	—	—		I/Z*12	I	
	FCE			O	—	O/Z*9		O/Z*9	O/Z*9	
	FALE	引导模式	0、1、3~5	O	—	O/Z*9		O/Z*9	O/Z*9	
			2	O	—	—	O/Z*9	O/Z*9	O/Z*9	
	FRE	引导模式	0、1、3~5	O	—	O/Z*9		O/Z*9	O/Z*9	
			2	O	—	—	O/Z*9	O/Z*9	O/Z*9	
	FCLE	引导模式	0、1、3~5	O	—	O/Z*9		O/Z*9	O/Z*9	
			2	O	—	—	O/Z*9	O/Z*9	O/Z*9	
	FWE	引导模式	0、1、3~5	O	—	O/Z*9		O/Z*9	O/Z*9	
			2	O	—	—	O/Z*9	O/Z*9	O/Z*9	
	NAF7 ~ NAF0			I	—	—		Z	Z	
		引导模式	0、1、3~5	O/Z	—	O/Z*9		O/Z*9	O/Z*9	
2			O/Z	—	—	O/Z*9	O/Z*9	O/Z*9		

引脚功能				引脚状态						
分类	引脚名			正规状态 (右记以外)	上电复位 *1	引脚状态的保持 *2		低功耗状态		
						EBUSKEEPE*3 (右记以外)		上电复位 *4	深度待机 模式	软件待机 模式
						0	1			
USB2.0 主机 / 功能模块	DP、DM			I/Z	Z	I/Z		Z	I/Z	
				O/Z	Z	O/Z		Z	O/Z	
	VBUS			I	I	I		I	I	
	REFRIN			I	I	I		I	I	
	USB_X1*6			I	I	I		Z	Z	
	USB_X2*6			O	O	O		L	L	
视频显示控制 器 4	LCD_DATA23 ~ LCD_DATA0			O	—	O/Z*9		O/Z*9	O/Z*9	
	LCD_TCON6 ~ LCD_TCON0			O	—	O/Z*9		O/Z*9	O/Z*9	
	LCD_CLK			O	—	O/Z*9		O/Z*9	O/Z*9	
	LCD_EXTCLK			I	—	—		Z	Z	
	DV_CLK			I	—	—		Z	Z	
	DV_DATA23 ~ DV_DATA20			I	—	—		I/Z*12	I	
	DV_DATA19 ~ DV_DATA0			I	—	—		Z	Z	
	DV_VSYNC、DV_HSYNC			I	—	—		Z	Z	
SD 主机接口	SD_CLK_0	引导模式	0 ~ 3	O	—	O/Z*9		O/Z*9	O/Z*9	
			4	O	—	—	O/Z*9	O/Z*9	O/Z*9	
	SD_CLK_1			O	—	O/Z*9		O/Z*9	O/Z*9	
	SD_CMD_0			I	—	—		Z	Z	
		引导模式	0 ~ 3	O/Z	—	O/Z*9		O/Z*9	O/Z*9	
			4	O/Z	—	—	O/Z*9	O/Z*9	O/Z*9	
	SD_CMD_1			I	—	—		Z	Z	
				O/Z	—	O/Z*9		O/Z*9	O/Z*9	
	SD_D3_0、SD_D2_0			I	—	—		Z	Z	
		引导模式	0 ~ 3	O/Z	—	O/Z*9		O/Z*9	O/Z*9	
			4	O/Z	—	—	O/Z*9	O/Z*9	O/Z*9	
	SD_D1_0、SD_D0_0			I	—	—		I/Z*12	I	
		引导模式	0 ~ 3	O/Z	—	O/Z*9		O/Z*9	O/Z*9	
			4	O/Z	—	—	O/Z*9	O/Z*9	O/Z*9	
	SD_D3_1 ~ SD_D0_1			I	—	—		Z	Z	
				O/Z	—	O/Z*9		O/Z*9	O/Z*9	
SD_CD_0			I	—	—		I/Z*12	I		
SD_CD_1			I	—	—		Z	Z		
SD_WP_0			I	—	—		I/Z*12	I		
SD_WP_1			I	—	—		Z	Z		
MMC 主机接口	MMC_CLK	引导模式	0 ~ 3	O	—	O/Z*9		O/Z*9	O/Z*9	
			5	O	—	—	O/Z*9	O/Z*9	O/Z*9	
	MMC_CMD			I	—	—		Z	Z	
		引导模式	0 ~ 3	O/Z	—	O/Z*9		O/Z*9	O/Z*9	
			5	O/Z	—	—	O/Z*9	O/Z*9	O/Z*9	

引脚功能			引脚状态							
分类	引脚名		正规状态 (右记以外)	上电复位 *1	引脚状态的保持 *2		低功耗状态			
					EBUSKEEPE*3 (右记以外)		上电复位 *4	深度待机 模式	软件待机 模式	
					0	1				
MMC 主机接口	MMC_D3、MMC_D2		I	—	—		Z	Z		
	引导模式	0 ~ 3	O/Z	—	O/Z*9		O/Z*9	O/Z*9		
		5	O/Z	—	—	O/Z*9	O/Z*9	O/Z*9		
	MMC_D1、MMC_D0		I	—	—		I/Z*12	I		
	引导模式	0 ~ 3	O/Z	—	O/Z*9		O/Z*9	O/Z*9		
		5	O/Z	—	—	O/Z*9	O/Z*9	O/Z*9		
	MMC_D7 ~ MMC_D4		I	—	—		Z	Z		
		O/Z	—	O/Z*9		O/Z*9	O/Z*9			
	MMC_CD		I	—	—		I/Z*12	I		
音频发生器	SGOUT_3 ~ SGOUT_0		O	—	O/Z*9		O/Z*9	O/Z*9		
视频解码器	VIDEO_X1*6		I	I	I		Z	Z		
	VIDEO_X2*6		O	O	O		L	L		
	VIN1、VIN2		I	Z	I		Z	I		
通用输入 / 输出端口	PA1、PA0		I	—	I	Z	Z	I		
			O/Z	—	O/Z*9		O/Z*9	O/Z*9		
	PB22、PB21、PB16 ~ PB1		I	Z	I	Z	Z	Z		
			O/Z	Z	O/Z*9		O/Z*9	O/Z*9		
	PB20 ~ PB17		I	Z	I	Z	Z	Z		
			引导模式	3	O/Z	Z	O/Z	O/Z*9	O/Z*9	O/Z*9
	2、4、5	O/Z		Z	O/Z*9		O/Z*9	O/Z*9		
	PC8、PC6、PC4 ~ PC1		I	Z	I	Z	Z	Z		
			O/Z	Z	O/Z*9		O/Z*9	O/Z*9		
	PC7、PC5		I	Z	I	I/Z*12	I/Z*12	I		
			O/Z	Z	O/Z*9		O/Z*9	O/Z*9		
	PC0		I	—	I	Z	Z	I		
			O/Z	—	O/Z*9		O/Z*9	O/Z*9		
	PD15 ~ PD4		I	Z	I	Z	Z	Z		
			引导模式	2	O/Z	Z	O/Z	O/Z*9	O/Z*9	O/Z*9
				3 ~ 5	O/Z	Z	O/Z*9		O/Z*9	O/Z*9
	PD3 ~ PD0		I	Z	I	Z	Z	Z		
			O/Z	Z	O/Z*9		O/Z*9	O/Z*9		
	PE7 ~ PE0		I	Z	I	Z	Z	Z		
			O/Z	Z	O/Z	Z	Z	Z		
PF23 ~ PF20		I	Z	Z		Z	Z			
		引导模式	0 ~ 3	O/Z	Z	O/Z*9		O/Z*9	O/Z*9	
			4、5	O/Z	Z	O/Z	O/Z*9	O/Z*9	O/Z*9	
PF19、PF18		I	Z	I	I/Z*12	I/Z*12	I			
		引导模式	0 ~ 3	O/Z	Z	O/Z*9		O/Z*9	O/Z*9	
			4、5	O/Z	Z	O/Z	O/Z*9	O/Z*9	O/Z*9	

引脚功能			引脚状态						
分类	引脚名		正规状态 (右记以外)	上电复位 *1	引脚状态的保持 *2		低功耗状态		
					EBUSKEEPE*3 (右记以外)		上电复位 *4	深度待机 模式	软件待机 模式
					0	1			
通用输入 / 输出 端口	PF17、PF16		I	Z	I	I/Z*12	I/Z*12	I	
	引导模式	0 ~ 3、5	O/Z	Z	O/Z*9		O/Z*9	O/Z*9	
		4	O/Z	Z	O/Z	O/Z*9	O/Z*9	O/Z*9	
	PF15 ~ PF0		I	Z	I	Z	Z	Z	
			O/Z	Z	O/Z*9		O/Z*9	O/Z*9	
	PG27 ~ PG4、PG1、PG0		I	Z	I	Z	Z	Z	
			O/Z	Z	O/Z*9		O/Z*9	O/Z*9	
	PG3、PG2		I	Z	I	I/Z*12	I/Z*12	I	
			O/Z	Z	O/Z*9		O/Z*9	O/Z*9	
	PH7 ~ PH0		I	Z	I	Z	Z	Z	
	PJ31 ~ PJ24、PJ19 ~ PJ0		I	Z	I	Z	Z	Z	
			O/Z	Z	O/Z*9		O/Z*9	O/Z*9	
	PJ23 ~ PJ20		I	Z	I	I/Z*12	I/Z*12	I	
			O/Z	Z	O/Z*9		O/Z*9	O/Z*9	
马达控制 PWM 定时器	PWM1A、PWM1B、PWM1C、 PWM1D、PWM1E、PWM1F、 PWM1G、PWM1H、PWM2A、 PWM2B、PWM2C、PWM2D、 PWM2E、PWM2F、PWM2G、 PWM2H		O	—	O/Z*9		O/Z*9	O/Z*9	
用户调试 接口 *15	TRST		I	I	I		Z	I	
	TCK		I	I	I		Z	I	
	TDI		I	I	I		Z	I	
	TDO		O/Z*14	O/Z*14	O/Z*14		O/Z*14	O/Z*14	
	TMS		I	I	I		Z	I	
仿真器 *15	AUDSYNC		—	—	—		—	—	
	AUDCK		—	—	—		—	—	
	AUDATA3 ~ AUDATA0		—	—	—		—	—	
	ASEBRKAK/ASEBRK		Z	Z	Z		Z	Z	

表 53.2 总线权释放时的引脚状态

引脚功能		引脚状态
分类	引脚名	总线权释放状态
时钟	CKIO	O/Z*7
系统控制	$\overline{\text{BREQ}}$	I
	$\overline{\text{BACK}}$	L
地址总线	A25 ~ A0	Z
数据总线	D31 ~ D0	Z
总线控制	$\overline{\text{CS5}} \sim \overline{\text{CS0}}$ 、 $\overline{\text{CE1A}}$ 、 $\overline{\text{CE2A}}$	Z
	$\overline{\text{RD}}$	Z
	$\overline{\text{RD}}/\overline{\text{WR}}$	Z
	$\overline{\text{BS}}$	Z
	WAIT	Z
	$\overline{\text{WE3}}/\overline{\text{ICIOR}}/\overline{\text{AH}}/\overline{\text{DQMUU}}$ 、 $\overline{\text{WE2}}/\overline{\text{ICIOR}}/\overline{\text{DQMUL}}$ 、 $\overline{\text{WE1}}/\overline{\text{DQMLU}}/\overline{\text{WE}}$ 、 $\overline{\text{WE0}}/\overline{\text{DQMLL}}$	Z
	RAS、 $\overline{\text{CAS}}$	O/Z*11
	CKE	O/Z*11
NAND 闪存控制器	FALE	Z
	$\overline{\text{FRE}}$	Z
	FCLE	Z
	$\overline{\text{FWE}}$	Z
	NAF7 ~ NAF0	Z

【符号说明】

- I : 输入
- O : 输出
- H : 高电平输出
- L : 低电平输出
- Z : 高阻抗

- 【注】 *1 表示通过给 $\overline{\text{RES}}$ 引脚输入低电平进行上电复位。在通过用户调试接口复位有效命令和看门狗定时器上溢进行上电复位时，引脚状态和各引脚中初始功能的正常运行时的状态相同（请参照“48. 通用输入/输出端口”）。
- *2 表示在通过 NMI、PC8 ~ PC5、PJ3、PJ1 的各引脚输入从深度待机模式返回后到深度待机解除源标志寄存器（DSFR）的 IOKEEP 位被清除为止的状态（请参照“49. 低功耗模式”）。
- *3 这是深度待机控制寄存器（DSCTR）的 EBUSKEEPE 位（请参照“49. 低功耗模式”）。
- *4 在从深度待机模式返回后，此 LSI 处于一定期间的上电复位状态（请参照“49. 低功耗模式”）。
- *5 取决于实时时钟的控制寄存器 5（RCR5）的 RCKSEL 位的设定（请参照“15. 实时时钟”）。
- *6 如果不使用连接晶体谐振器的引脚，就必须将输入引脚（EXTAL、RTC_X1、AUDIO_X1、USB_X1、VIDEO_X1）置为固定电平（上拉、下拉、连接电源或者接地），并且将输出引脚（XTAL、RTC_X2、AUDIO_X2、USB_X2、VIDEO_X2）置为开路。
- *7 取决于时钟脉冲振荡器的频率控制寄存器（FRQCR）的 CKOEN 位的设定（请参照“5. 时钟脉冲振荡器”）。
- *8 取决于软件复位控制寄存器（SWRSTCR）的 AXTALE 位的设定（请参照“49. 低功耗模式”）。
- *9 取决于待机控制寄存器 3（STBCR3）的 HIZ 位的设定（请参照“49. 低功耗模式”）。
- *10 取决于总线状态控制器的公共控制寄存器（CMNCR）的 HIZMEM 位的设定（请参照“10. 总线状态控制器”）。
- *11 取决于总线状态控制器的公共控制寄存器（CMNCR）的 HIZCNT 位的设定（请参照“10. 总线状态控制器”）。
- *12 取决于深度待机解除源选择寄存器（DSSSR）的各位的设定（请参照“49. 低功耗模式”）。
- *13 取决于实时时钟的控制寄存器 2（RCR2）的 RTCEN 位的设定（请参照“15. 实时时钟”）。

*14 在用户调试接口的 TAP 控制器不处于 Shift-DR 和 Shift-IR 的状态下，为“Z”。

*15 这是产品芯片模式中 ($\overline{\text{ASEMD}}=\text{H}$) 的引脚状态。有关 ASE 模式中 ($\overline{\text{ASEMD}}=\text{L}$) 的引脚状态，请参照《仿真手册》。

*16 在输出的情况下，固定输出高电平或者低电平。不发生振荡。

53.2 未使用引脚的处理

未使用引脚的处理方法如下所示。

表 53.3 未使用引脚的处理（用户调试接口 / 仿真器接口的引脚除外）

引脚	处理
NMI	固定为高电平（上拉或者连接电源）。
DP、DM、VBUS	连接 USBDPV _{SS} （SH7268/SH7269（QFP））。 连接 V _{SS} （SH7269（BGA））。
REFRIN	通过 5.6kΩ±20% 的电阻连接 USBAPV _{CC} 。
USB 专用 1.2V 电源（USB _{AV} CC、USB _{DV} CC、USB _{UV} CC）	提供 1.2V 电源。 【注】 SH7269（BGA）没有 USB _{DV} CC 引脚和 USB _{UV} CC 引脚。
USB 专用 3.3V 电源（USB _{AP} CC、USB _{DP} CC）	提供 3.3V 电源。 【注】 SH7269（BGA）没有 USB _{DP} CC 引脚。
USB 专用接地（USB _{AP} V _{SS} 、USB _{DP} V _{SS} 、USB _{AV} V _{SS} 、USB _{DV} V _{SS} 、USB _{UV} V _{SS} ）	接地 【注】 SH7269（BGA）没有左记的引脚。
AVref	连接 AV _{CC} 。
A/D 专用电源（AV _{CC} ）	提供 3.3V 电源。
A/D 专用接地（AV _{SS} ）	接地
视频信号输入的 A/D 专用电源（V _{DA} V _{CC} ）	提供 3.3V 电源。
视频信号输入的 A/D 专用接地（V _{DA} V _{SS} ）	接地
BIAS	通过 24kΩ±10% 的电阻连接 V _{DA} V _{SS} 。
VIN1、VIN2、VRT、VRB	开路
上述以外的输入专用引脚	固定电平（上拉、下拉、连接电源或者接地）
上述以外的输入 / 输出引脚	设定为输入引脚并且固定电平（上拉或者下拉）， 或者设定为输出引脚并且置为开路。
输出专用引脚	开路

【注】 建议上拉或者下拉的电阻为 4.7kΩ ~ 100kΩ。

表 53.4 在产品芯片模式中未使用用户调试接口时的引脚处理

引脚	处理
$\overline{\text{ASEMD}}$	固定为高电平（上拉或者连接电源）。
TRST	固定为低电平（下拉或者接地）。
TCK、TMS、TDI	固定电平（上拉、下拉、连接电源或者接地）
TDO、 $\overline{\text{ASEBRKAK}}/\overline{\text{ASEBRK}}$	开路

【注】 1. 使用用户调试接口时的引脚处理必须符合所使用的仿真器的规格。
2. 建议上拉或者下拉的电阻为 4.7kΩ ~ 100kΩ。

53.3 深度待机模式中的引脚处理

深度待机模式中的引脚处理方法如下所示。

有关深度待机模式中的引脚状态，请参照“53.1 引脚状态”的深度待机模式的项目。在未使用引脚时，即使在深度待机模式中，也必须遵循“53.2 未使用引脚的处理”。

表 53.5 深度待机模式中的引脚处理

引脚	处理
1.2V 电源 (V _{CC} 、USB _{DDV} _{CC} 、USB _{UV} _{CC} 、USB _{AV} _{CC})	提供 1.2V 电源。 【注】 SH7269 (BGA) 没有 USB _{DDV} _{CC} 引脚和 USB _{UV} _{CC} 引脚。
3.3V 电源 (P _V _{CC} 、A _V _{CC} 、USB _{DDP} _{CC} 、USB _{AP} _{CC} 、PLL _V _{CC} 、V _{DAV} _{CC})	提供 3.3V 电源。 【注】 SH7269 (BGA) 没有 USB _{DDP} _{CC} 引脚。
接地 (V _{SS} 、PLL _V _{SS} 、USB _{DDV} _{SS} 、USB _{UV} _{SS} 、USB _{AV} _{SS} 、A _V _{SS} 、USB _{DDP} _{SS} 、USB _{AP} _{SS} 、V _{DAV} _{SS})	接地 【注】 SH7269 (BGA) 没有 PLL _V _{SS} 、USB _{DDV} _{SS} 、USB _{AV} _{SS} 、USB _{UV} _{SS} 、USB _{DDP} _{SS} 、USB _{AP} _{SS} 引脚。
VBUS	固定电平 (上拉、下拉、连接电源或者接地) 或者开路 但是, 如果固定为高电平, 就会消耗“表 52.2 DC 特性 (2) 【消耗电流】 SH7268/SH7269 (QFP)”中记载的电流。
REFRIN	通过 5.6kΩ±1% 的电阻连接 USB _{AP} _{SS} 。
DP、DM	固定电平 (上拉、下拉、连接电源或者接地) 或者开路
AVref	固定电平 (3.0V ~ A _V _{CC})
BIAS	通过 24kΩ±1% 的电阻连接 V _{DAV} _{SS} 。
VRT、VRB	通过 0.1μF 的电容连接 V _{DAV} _{SS} 。
VIN1、VIN2	固定电平 (上拉、下拉、连接电源或者接地) 或者开路
EXTAL、RTC_X1、AUDIO_X1、USB_X1、VIDEO_X1	连接晶体谐振器或者输入时钟或者固定电平 (上拉、下拉、连接电源或者接地)
XTAL、RTC_X2、AUDIO_X2、USB_X2、VIDEO_X2	连接晶体谐振器或者开路
上述以外的输入专用引脚	固定电平 (上拉、下拉、连接电源或者接地)
上述以外的输入状态的输入 / 输出引脚	固定电平 (上拉或者下拉)
上述以外的输入 / 输出高阻抗状态的输入 / 输出引脚	固定电平 (上拉或者下拉) 或者开路
上述以外的输出状态的输入 / 输出引脚	开路
上述以外的输出专用引脚	开路

【注】 建议上拉或者下拉的电阻为 4.7kΩ ~ 100kΩ。

53.4 旁路电容的推荐组合

必须在各对电源引脚之间插入多层陶瓷电容作为旁路电容，并且尽量将旁路电容安装在 LSI 的电源引脚附近。必须使用 0.1 μ F ~ 0.33 μ F（推荐值）的电容。有关晶体振荡相关的电容，请参照“5. 时钟脉冲振荡器”。

SH7268 和 SH7269（QFP）的外接电容的配置例子分别如图 53.1 和图 53.2 所示，SH7269（BGA）的外接电容组合如表 53.6 所示。

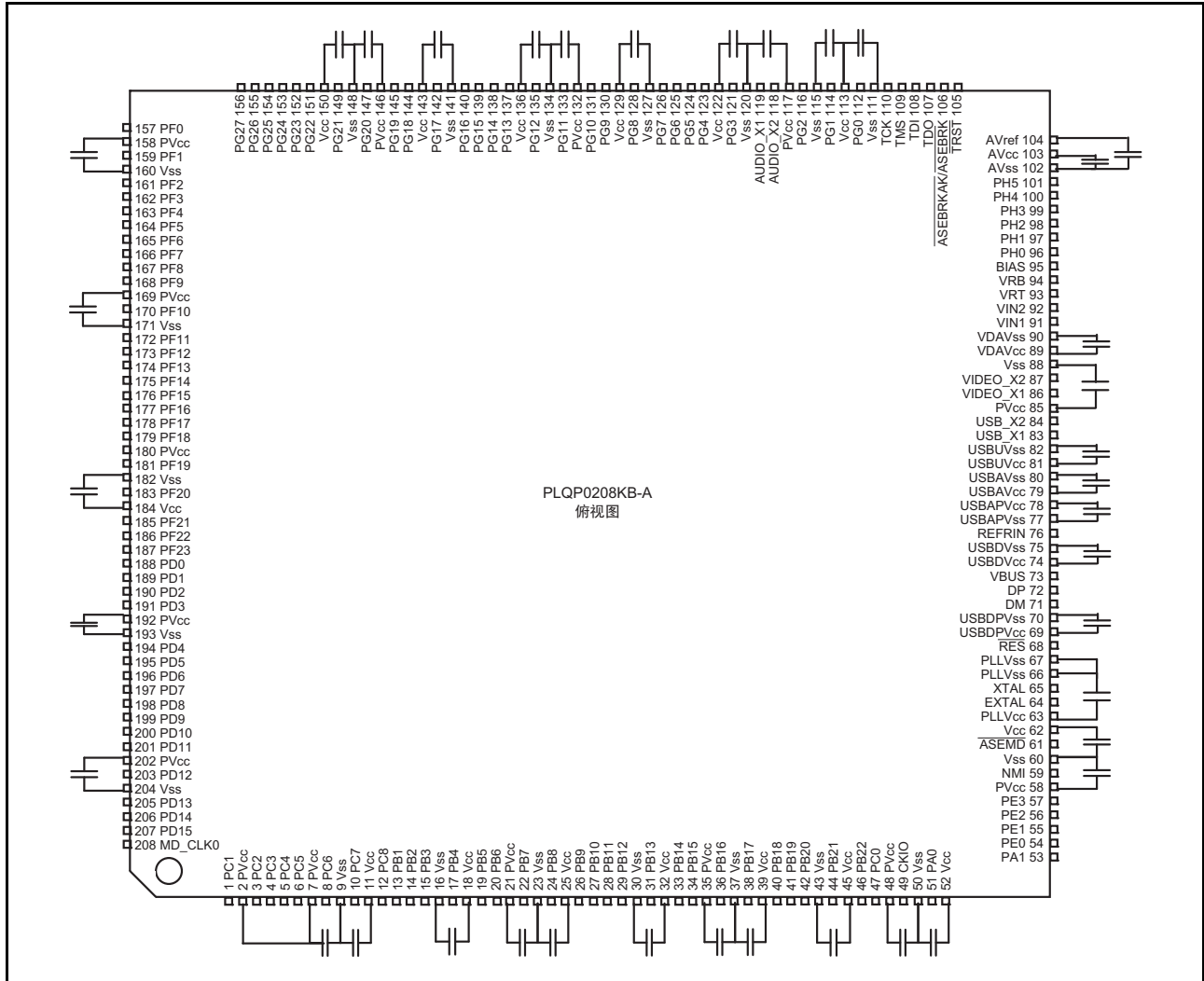


图 53.1 SH7268 的外接电容的配置例子

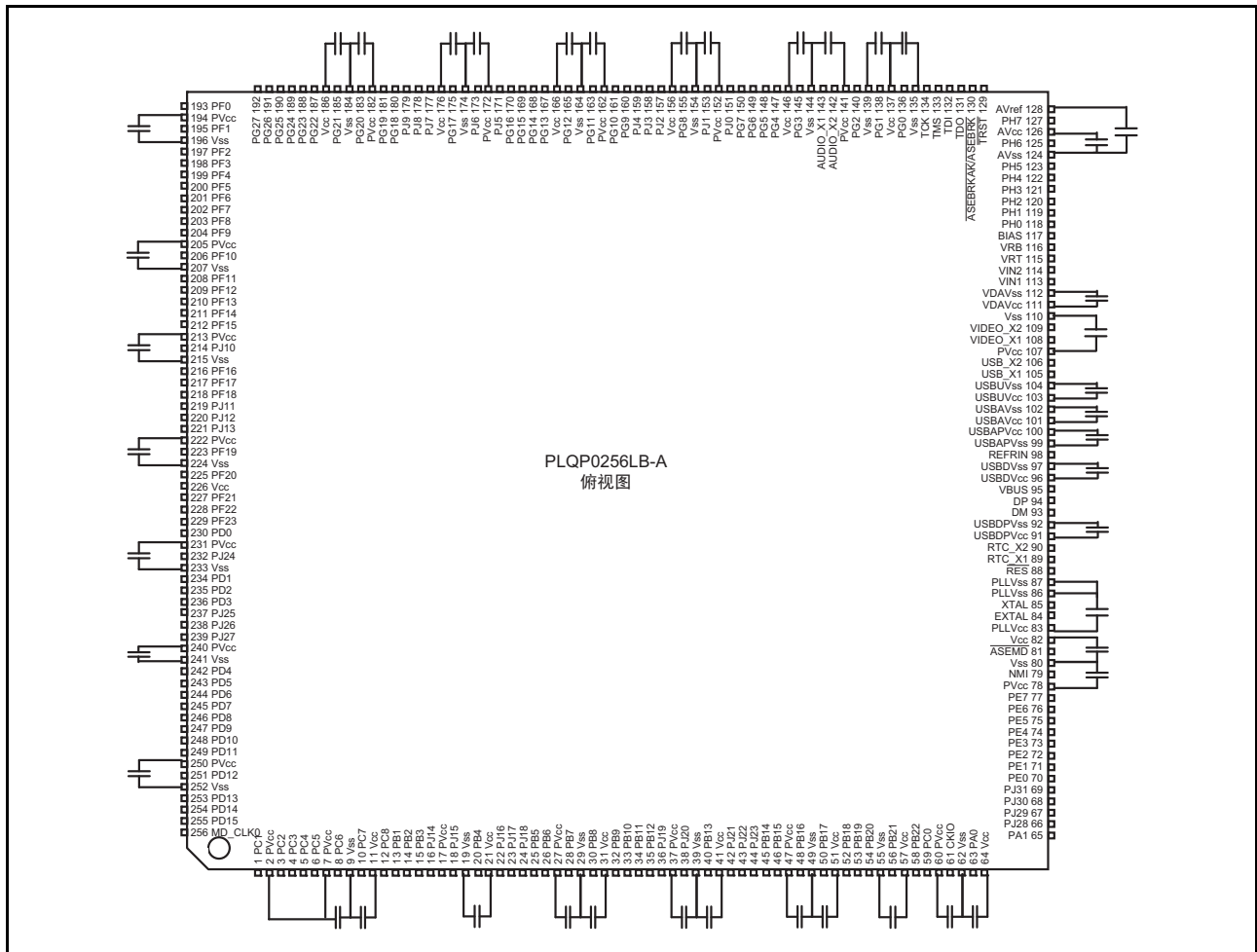
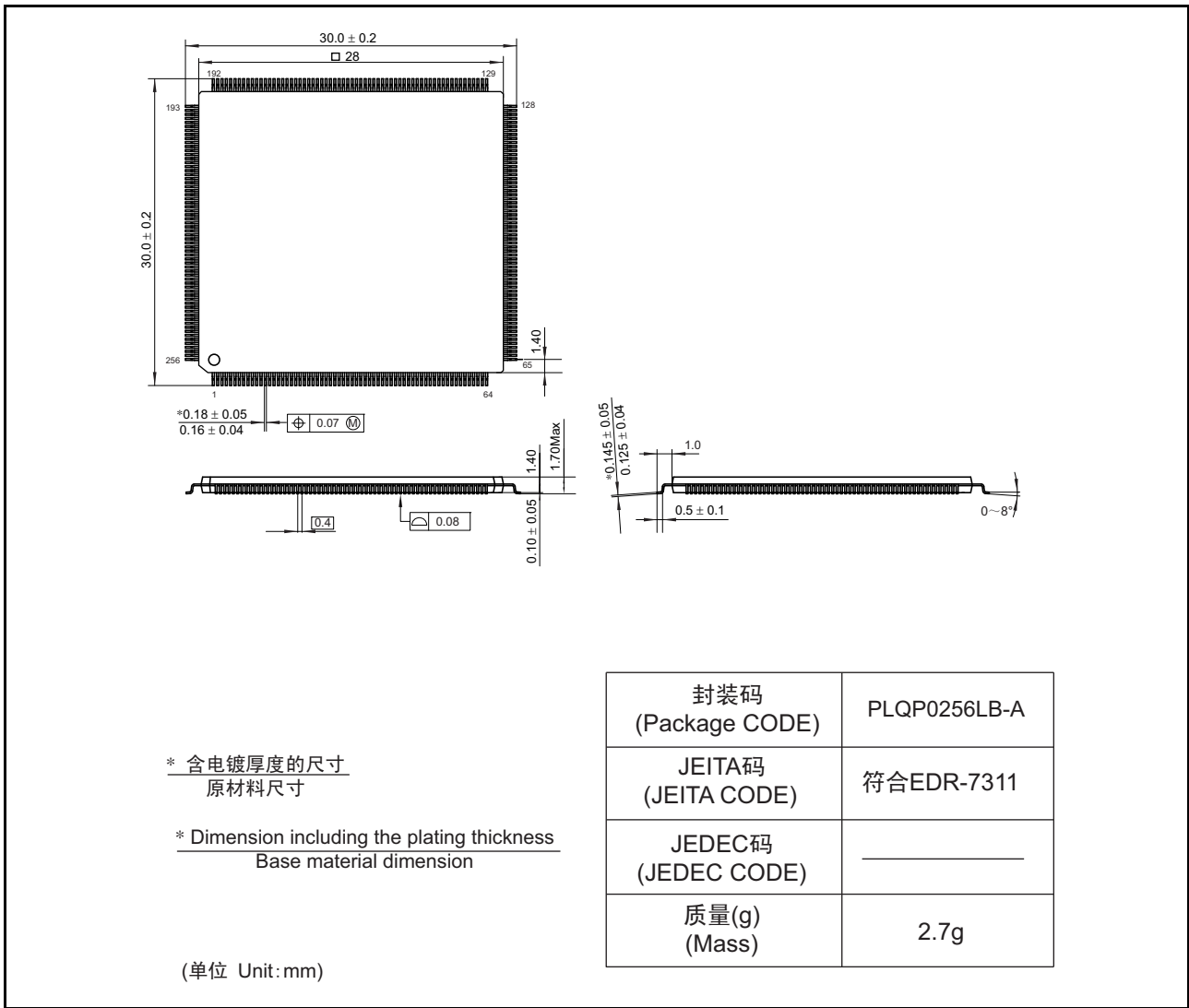


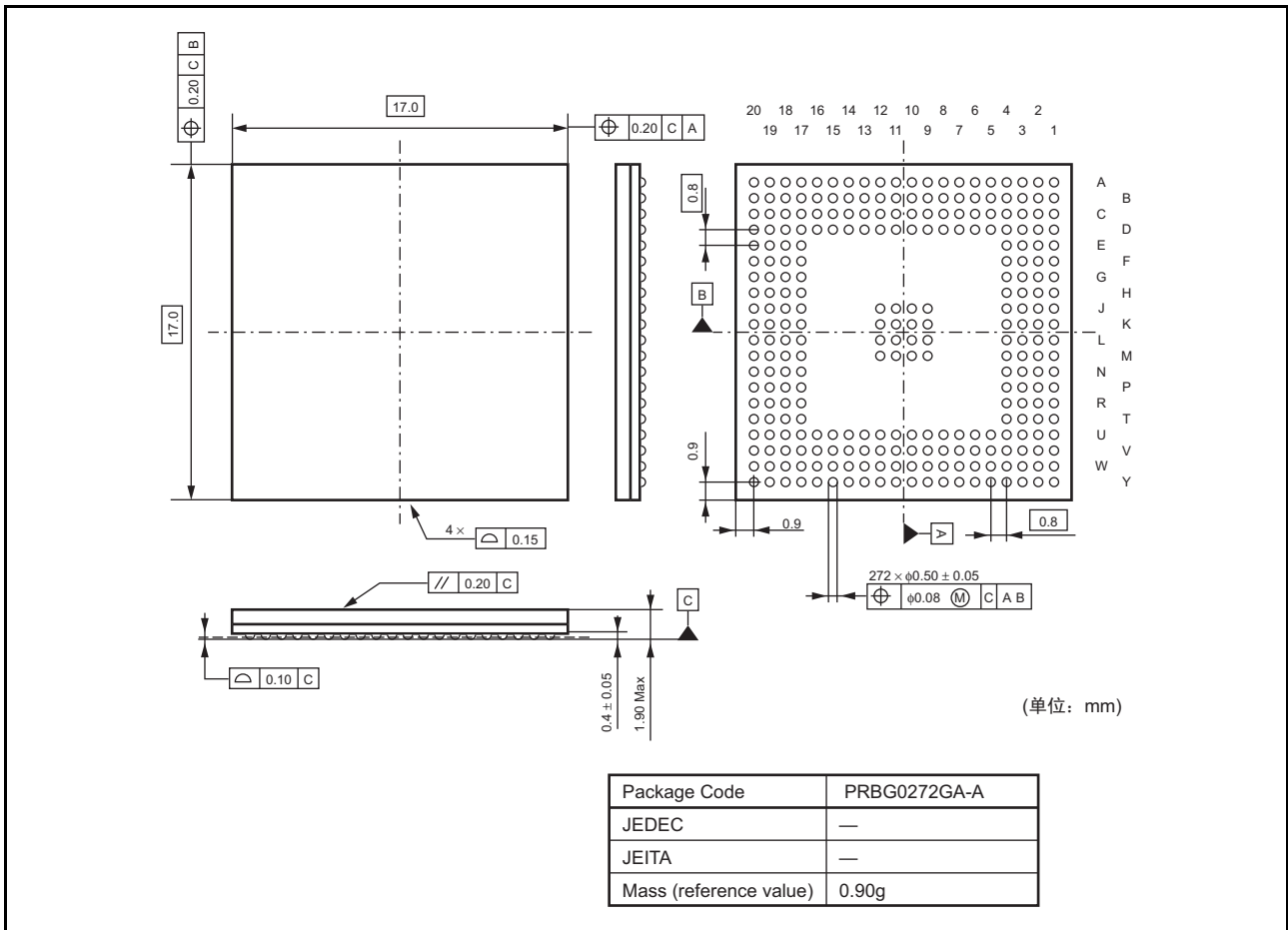
图 53.2 SH7269 (QFP) 的外接电容的配置例子

表 53.6 SH7269 (BGA) 的外接电容组合

电源		接地	
引脚号	引脚名	引脚号	引脚名
A2、B20、C19、D5、D6、D18、E17、H4、J4、M17、N17、T4、U3、U10、V2、V10、W1	V _{CC}	A1、A20、B2、B19、C3、C10、C18、D4、D10、D17、J9、J10、J11、J12、K4、K9、K10、K11、K12、K17、L4、L9、L10、L11、L12、L17、M4、M9、M10、M11、M12、U1、U4、U8、U9、U12、U17、V3、V9、V13、V18、W2、W11、W19、Y1、Y7、Y13、Y20	V _{SS}
A19、B1、B18、C2、D2、D3、D11、D12、D15、D16、E4、J17、J18、N3、N4、T17、U18、V19、W20、Y11	PV _{CC}		
U6	PLLV _{CC}		
V11	USBAPV _{CC}		
V12	USBV _{CC}		
U13	VDAV _{CC}	U15	VDAV _{SS}
Y18	AV _{CC}	W18	AV _{SS}
Y19	AV _{ref}		



附图 1.2 SH7269 (QFP 版) 的封装尺寸图



附图 1.3 SH7269 (BGA 版) 的封装尺寸图

索引

数字

16 位 /32 位的位移量 47

A

A/D 转换精度的定义 936
 A/D 转换开始请求的延迟功能 391
 A/D 转换器 922
 A/D 转换器的启动 397
 A/D 转换器的时序 1944
 A/D 转换器特性 1957
 A/D 转换时间（多通道模式 / 扫描模式） 934
 A/D 转换时序 933
 AC 特性 1895
 AC 特性的测量条件 1957
 ATTCH 中断 1051

B

BCHG 中断 1050
 BEMP 中断 1047
 BRDY 中断 1041
 本地接收滤波器屏蔽（LAFM） 743
 本 LSI 的产品一览表 9
 本 LSI 的封装尺寸图 1971, 1972, 1973
 本 LSI 的框图 10
 本 LSI 的引脚功能 14
 本 LSI 的引脚排列图 11
 本 LSI 的引脚一览表 21
 本 LSI 特点 1
 比较匹配定时器 441
 边界扫描 1745
 表的设定 1525
 波特率发生器 719

C

CAN 接口 734
 CAN 睡眠模式 786
 CAN 总线接口 803
 CD-ROM 解码器 878
 CMCNT 的计数时序 444
 CMCNT 的写操作和比较匹配的竞争 446
 CMCNT 的字节写和递增计数的竞争 447
 CMCNT 的字写和递增计数的竞争 446
 CPU 41
 CSn 有效期间的扩展 208
 采样率转换器 1540
 槽非法指令 99
 测试模式的设定 788
 乘法 / 乘加运算 46
 乘加低位寄存器（MACL） 43
 乘加高位寄存器（MACH） 43
 程序计数器（PC） 43

程序执行状态 69
 初始化 550
 传输协议 809
 传送格式 540, 541
 传送率 646
 串行音频接口 671
 串行音频接口的时序 1941
 从属发送 658
 从属接收 660
 从属模式的运行 557
 存储器的数据格式 44
 存储体的保存 133
 存储体的对象寄存器和输入 / 输出方式 132
 存储体的恢复 134
 存储体激活 224
 存取长度和数据调整 196
 存取长度和突发数的关系 219
 存取等待的控制 206
 存取周期之间的空闲 246
 错误标记 1531

D

DC 特性 1888
 DMA 传送流程图 273
 DREQ 引脚的采样时序 284
 DTCH 中断 1051
 大容量内部 RAM 的地址空间 1618
 带 FIFO 的时钟同步串行 I/O 704
 待机控制电路 85
 带字节选择的 SRAM 接口 237
 单次读 221
 单次写 223
 单地址模式 280
 单精度浮点格式 70
 单通道模式 927
 等时传送 1069
 低功耗模式 1700
 低功耗状态 69
 低功率 SDRAM 232
 地址错误 96
 地址的多路复用 212
 地址映像 164
 地址阵列 150, 158
 电特性 1887
 掉电模式 230
 定时器的寄存器 774
 读地址阵列 158
 读数据阵列 159
 对绝对精度的影响 939
 多功能定时器脉冲单元 2 288
 多功能定时器脉冲单元 2 的功能一览表 289
 多功能定时器脉冲单元 2 的输出引脚的初始化方法 418
 多功能定时器脉冲单元 2 的中断源 395

多通道模式	929	I ² C 总线接口 3 的时序	1940
E		IBUF 中断	919
ECC 校正功能	914	ID 排序	752
EDC 校验功能	914	IEBus 的位格式	816
EOFERR 中断	1051	IEBusTM 控制器	806
F		IEBus 通信协议	807
FIFO 缓冲存储器	1057	IERR 中断	919
FIFO 内置型串行通信接口	475	IREADY 中断	920
FPU 相关的 CPU 指令	68	IRQ 中断	117
FPU 异常源	76	ISEC 中断	919
发送触发控制字段	746	ISY 中断	919
发送触发时间 (TTT)	746	ITARG 中断	919
发送的内部仲裁	790	J	
非规格化数	72	JPEG 符号数据格式	1525
非数 (NaN)	72	JPEG 解压错误	1531
非线性误差	936	寄存器	
非压缩模式	688	ABACK0	769
分频比的更改	88	ABACK1	769
分频器 1	85	ADCSR	925
浮点单元 (FPU)	70	ADDRA ~ ADDRH	924
浮点的范围	71	BAMR	140
浮点寄存器	73	BAR	140
浮点运算指令	66, 100	BBR	142
复位顺序	785	BCR0	756
复位同步 PWM 模式	359	BCR1	755
复位状态	69	BDMR	141
G		BDR	141
高速缓存	150	BEMPENB	995
高速缓存的搜索	155	BEMPSTS	1005
高速缓存和外部存储器 / 大容量内部 RAM 的相关性	157	BRCCR	143
高速缓存运行的汇总	157	BRDYENB	993
高速内部 RAM 的地址空间	1618	BRDYSTS	1003
各运行模式中的寄存器状态一览表	1885	BSBPR	1736
管道的安排	1077	BSID	1741
管道控制	1052	BSIR	1737
过程寄存器 (PR)	43	BUSWAIT	976
H		CBUFCTL0	902
互补 PWM 模式	361	CBUFCTL1	903
缓冲格式	915	CBUFCTL2	903
环回模式	562	CBUFCTL3	903
恢复中断	1050	CBUFST0	894
回写缓冲器 (只限于操作数高速缓存)	157	CBUFST1	895
霍夫曼表的设定	1526	CBUFST2	895
霍夫曼符号化段错误	1532	CCR	780
I		CCR1	152
I ² C 总线格式	654	CCR2	153
I ² C 总线接口 3	641	CE_ARG	1570
		CE_ARG_CMD12	1570
		CE_BLOCK_SET	1571
		CE_BUF_ACC	1573
		CE_CLK_CTRL	1572
		CE_CMD_CTRL	1571
		CE_CMD_SET	1568
		CE_DATA	1575
		CE_HOST_STS1	1582
		CE_HOST_STS2	1583

CE_INT	1576	DVSTCTR.....	978
CE_INT_EN.....	1580	FLADR.....	947
CE_RESP.....	1574	FLADR2.....	948
CE_RESP_CMD12.....	1575	FLBSYCNT.....	954
CE_VERSION.....	1587	FLBSYTMR.....	953
CFIFO.....	983	FLCMCDR.....	947
CFIFOCTR.....	989	FLCMDCR.....	945
CFIFOSEL.....	985	FLCMNCR.....	944
CHCR.....	260	FLDATAR.....	950
CMAX_TEW.....	776	FLDTCNTR.....	949
CMCNT.....	443	FLDTFIFO.....	954
CMCOR.....	444	FLECFIFO.....	955
CMCSR.....	443	FLHOLDCR.....	956
CMNCR.....	167	FLINTDMACR.....	951
CMSTR.....	442	FLTRCR.....	955
CROMCTL0.....	886	FPSCR.....	74
CROMCTL1.....	887	FPUL.....	75
CROMCTL3.....	888	FRMNUM.....	1006
CROMCTL4.....	889	FRQCR.....	87
CROMCTL5.....	890	GSR.....	753
CROMEN.....	884	HEAD00.....	896
CROMST0.....	890	HEAD01.....	896
CROMST0M.....	904	HEAD02.....	896
CROMST1.....	891	HEAD03.....	896
CROMST3.....	891	HEAD20.....	899
CROMST4.....	892	HEAD21.....	899
CROMST5.....	893	HEAD22.....	899
CROMST6.....	894	HEAD23.....	899
CROMSY0.....	885	IBCR.....	114
CS0WCR.....	172, 181, 190	IBNR.....	115
CS1WCR.....	174	ICCR1.....	645
CS2WCR.....	176, 185	ICCR2.....	647
CS3WCR.....	176, 186	ICDRR.....	652
CS4WCR.....	177, 183	ICDRS.....	652
CS5WCR.....	179, 188	ICDRT.....	652
CSnBCR (n = 0 ~ 5).....	169	ICIER.....	649
CTRL.....	856	ICMR.....	648
CYCTR.....	781	ICR0.....	109
D0FBCFG.....	982	ICR1.....	110
D0FIFO.....	983	ICR2.....	111
D0FIFOCTR.....	989	ICSR.....	650
D0FIFOSEL.....	987	IEAR1.....	823
D1FBCFG.....	982	IEAR2.....	823
D1FIFO.....	983	IECKSR.....	836
D1FIFOCTR.....	989	IECMR.....	821
D1FIFOSEL.....	987	IECTR.....	820
DAR.....	259	IEFLG.....	827
DCPCFG.....	1010	IEIER.....	835
DCPCTR.....	1012	IEIET.....	831
DCPMAXP.....	1011	IELA1.....	826
DEVADDn (n=0 ~ A).....	1035	IELA2.....	827
DMAOR.....	266	IEMA1.....	825
DMARSO ~ DMARS7.....	268	IEMA2.....	825
DMATCR.....	259	IEMCR.....	822
DSCTR.....	1720	IERB.....	837
DSESR.....	1722	IERBFL.....	826
DSFR.....	1724	IERCTL.....	826
DSSSR.....	1721	IERSR.....	832

IESA1.....	824	PBPR0.....	1643
IESA2.....	824	PBPR1.....	1642
IETB.....	837	PCCR0.....	1645
IETBFL.....	824	PCCR1.....	1644
IETSR.....	829	PCCR2.....	1643
IMR.....	763	PCDR0.....	1646
INHINT.....	908	PCIOR0.....	1646
INTENB0.....	990	PCPR0.....	1647
INTENB1.....	991	PDCR0.....	1651
INTHOLD.....	907	PDCR1.....	1650
INTSTS0.....	997	PDCR2.....	1649
INTSTS1.....	1000	PDCR3.....	1648
IPR01、IPR02、IPR05 ~ IPR26.....	107	PDDR0.....	1652
IRQRR.....	112	PDIOR0.....	1652
IRR.....	758	PDPR0.....	1653
JCCMD.....	1503	PECR0.....	1655
JCDERR.....	1509	PECR1.....	1654
JCDRID.....	1505	PEDR0.....	1656
JCDRIU.....	1505	PEIOR0.....	1656
JCDTCD.....	1507	PEPR0.....	1657
JCDTCM.....	1507	PFCR0.....	1664
JCDTCU.....	1507	PFCR1.....	1663
JCHSZD.....	1506	PFCR2.....	1662
JCHSZU.....	1506	PFCR3.....	1661
JCHTN.....	1504	PFDR0.....	1666
JCMOD.....	1502	PFIOR0.....	1665
JCQTN.....	1504	PFPR0.....	1668
JCRST.....	1510	PGCR0.....	1677
JCVSZD.....	1506	PGCR1.....	1675
JCVSZU.....	1505	PGCR2.....	1674
JIFDCNT.....	1515	PGCR3.....	1673
JIFDDA.....	1518	PGCR4.....	1672
JIFDDOFST.....	1517	PGCR5.....	1671
JIFDSA.....	1517	PGCR6.....	1670
JIFECNT.....	1510	PGDR0.....	1680
JIFEDA.....	1513	PGDR1.....	1679
JIFESA.....	1512	PGIOR0.....	1678
JIFESOFST.....	1512	PGIOR1.....	1678
JINTE0.....	1508	PGPR0.....	1682
JINTS0.....	1509	PGPR1.....	1681
MBIMR0.....	772	PHCR0.....	1684
MBIMR1.....	772	PHCR1.....	1683
MCR.....	749	PHPR0.....	1685
NF2CYC.....	653	PINTER.....	113
NRDYENB.....	994	PIPEBUF.....	1020
NRDYSTS.....	1004	PIPECFG.....	1017
PAIOR0.....	1631	PIPEMAXP.....	1022
PAPR0.....	1632	PIPEnCTR (n=1 ~ 9).....	1024
PBCR0.....	1639	PIPEnTRE (n=1 ~ 5).....	1033
PBCR1.....	1638	PIPEnTRN (n=1 ~ 5).....	1034
PBCR2.....	1637	PIPEPERI.....	1023
PBCR3.....	1635	PIPESEL.....	1016
PBCR4.....	1634	PIRR.....	113
PBCR5.....	1633	PJCR0.....	1693
PBDR0.....	1641	PJCR7.....	1686
PBDR1.....	1640	PJDR0.....	1695
PBIOR0.....	1640	PJIOR0.....	1694
PBIOR1.....	1640	PJPR0.....	1697

PWBFR_n.....	1613	SCLSR.....	495
PWBTCR.....	1614	SCRSR.....	479
PWCNT_n.....	1610	SCSCR.....	482
PWCR_n.....	1609	SCSMR.....	480
PWCYR_n.....	1610	SCSPTR.....	493
PWDTR_n.....	1611	SCTSR.....	480
PWPR_n.....	1610	SDBPR.....	1741
R64CNT.....	459	SDBSR.....	1737
RCR1.....	467	SDCR.....	191
RCR2.....	468	SDIR.....	1742
RCR3.....	469	SGCR1.....	1556
RCR5.....	469	SGCSR.....	1557
RDAD.....	867	SGLR.....	1558
RDAR.....	265	SGSFR.....	1558
RDAYAR.....	465	SGTFR.....	1558
RDAYCNT.....	461	SHEAD00.....	897
RDMATCR.....	266	SHEAD01.....	897
REC.....	763	SHEAD02.....	897
RFMK.....	782	SHEAD03.....	897
RFPR0.....	771	SHEAD04.....	898
RFPR1.....	771	SHEAD05.....	898
RFRH/L.....	470	SHEAD06.....	898
RFTR OFF.....	777	SHEAD07.....	898
RHRAR.....	464	SHEAD20.....	900
RHRCNT.....	460	SHEAD21.....	900
RLCA.....	866	SHEAD22.....	900
RLCS.....	869	SHEAD23.....	900
RMINAR.....	463	SHEAD24.....	901
RMINCNT.....	460	SHEAD25.....	901
RMONAR.....	465	SHEAD26.....	901
RMONCNT.....	462	SHEAD27.....	901
ROMDECRST.....	904	SICTR.....	708
RRAMKP.....	1719	SIFCTR.....	715
RRCA.....	867	SIIER.....	713
RRCS.....	870	SIMDR.....	706
RSAR.....	265	SIRDAR.....	718
RSECAR.....	463	SIRDAR.....	710
RSECCNT.....	459	SISCR.....	716
RSTSTAT.....	905	SISTR.....	711
RTCNT.....	194	SITDAR.....	717
RTCOR.....	195	SITDR.....	710
RTCSR.....	193	SOFCFG.....	996
RUI.....	868	SPBFCR.....	534
RWKAR.....	464	SPBFDR.....	535
RWKCNT.....	461	SPBR.....	527
RXPR0.....	770	SPCKD.....	529
RXPR1.....	770	SPCMD.....	532
RYRAR.....	466	SPCR.....	520
RYRCNT.....	462	SPDCR.....	528
SAR.....	259	SPDR.....	525
SAR (I ² C 总线接口 3).....	651	SPND.....	531
SCBRR.....	487	SPPCR.....	522
SCEMR.....	496	SPSCR.....	526
SCFCR.....	491	SPSR.....	523
SCFDR.....	493	SPSSR.....	526
SCFRDR.....	479	SRCCTRL.....	1545
SCFSR.....	484	SRCID.....	1541
SCFTDR.....	480	SRCIDCTRL.....	1543

SRCOD.....	1542	TIOR.....	300
SRCODCTRL.....	1544	TITCNT.....	331
SRCSTAT.....	1548	TITCR.....	330
SSI.....	906	TLCA.....	861
SSICR.....	675	TLCS.....	864
SSIFCR.....	683	TMDR.....	298
SSIFRDR.....	687	TOCR1.....	323
SSIFSR.....	685	TOCR2.....	325
SSIFTDR.....	686	TOER.....	322
SSIRDR.....	682	TOLBR.....	327
SSISR.....	680	TRCA.....	862
SSITDR.....	682	TRCS.....	865
SSLND.....	530	TRWER.....	321
SSLP.....	521	TSR.....	311, 778
STAT.....	859	TSTR.....	319
STBCR1.....	1702	TSYR.....	320
STBCR2.....	1703	TTCR0.....	774
STBCR3.....	1704	TTTSEL.....	784
STBCR4.....	1705	TWCR.....	334
STBCR5.....	1706	TUI.....	863
STBCR6.....	1707	TXACK0.....	768
STBCR7.....	1708	TXACK1.....	768
STBCR8.....	1709	TXCR0.....	767
STRMDIN0.....	908	TXCR1.....	767
STRMDIN2.....	909	TXPR0.....	766
STRMDOUT0.....	909	TXPR1.....	766
SWRSTCR1.....	1712	UFRMNUM.....	1007
SWRSTCR2.....	1713	UMSR0.....	773
SYSCFG.....	973	UMSR1.....	773
SYSCR1.....	1714	WRCSR.....	451
SYSCR2.....	1715	USBADDR.....	1007
SYSCR3.....	1716	USBINDX.....	1009
SYSCR4.....	1717	USBLENG.....	1009
SYSCR5.....	1718	USBREQ.....	1008
SYSSTS.....	977	USBVAL.....	1008
TADCOBRA_4.....	318	WTCNT.....	449
TADCOBRB_4.....	318	WTCSR.....	450
TADCORA_4.....	318	XTALCTR.....	1725
TADCORB_4.....	318	寄存器的数据格式.....	44
TADCR.....	316	寄存器地址一览表	
TBTER.....	332	(按功能模块和手册章节号的顺序).....	1748
TBTM.....	314	寄存器位一览表.....	1786
TCBR.....	330	寄存器组.....	43, 132
TCDR.....	329	寄存器组错误.....	97
TCMR0 ~ 2.....	782	寄存器组的错误异常处理.....	97, 134
TCNT.....	318	寄存器组异常.....	134
TCNTR.....	781	级联运行.....	345
TCNTS.....	329	间隔定时器模式的用法.....	454
TCR.....	295	接通和切断电源的顺序.....	1887
TDAD.....	862	解压处理.....	1528
TDDR.....	329	晶体振荡器.....	85
TDER.....	333	决定空闲周期数的项目.....	246
TEC.....	763	绝对地址.....	47
TESTMODE.....	981	绝对地址的参照.....	47
TGCR.....	328	绝对最大额定值.....	1887
TGR.....	319		
TICCR.....	315		
TIER.....	309		

K	P
看门狗定时器 448	PCMCIA 接口 241
看门狗定时器的时序 1932	PINT 中断 117
看门狗定时器模式的用法 453	PLL 电路 85
控制寄存器的初始值 43	PWM 模式 348
控制器局域网 732	配置模式 698
控制器局域网的存储器映像 736	批量传送 1067
控制器局域网的控制寄存器 749	偏移误差 936
控制器局域网的设定 785	频率的更改方法 88
控制器局域网的信箱寄存器 764	
控制器局域网的引脚端口的设定 803	Q
控制器局域网的中断源 801	(潜在) 时间主控 792
控制传送阶段转移中断 1049	求 SCBRR 设定值的表达式 487
	取指令周期的断点 145
L	全局基址寄存器 (GBR) 42
LRU 151	
立即数 46	R
立即数的参照 46	RISC 方式 45
立即数的数据格式 44	容许信号源阻抗 939
量化表的设定 1525	软件待机模式 1726
量化误差 936	软件待机模式的解除步骤 (看门狗定时器) 453
逻辑运算指令 62	瑞萨串行外围接口 516
	瑞萨 SPDIF 接口 852
M	
MPX-I/O 接口 209	S
马达控制 PWM 定时器 1607	SACK 中断 1051
满刻度误差 936	SDRAM 接口 211
命令存取模式 959	SD 主机接口 1565
模块待机功能 1733	SD 主机接口的时序 1953
模块有效模式 698	SIGN 中断 1051
模拟电压的设定 937	SOF 内插功能 1076
模拟输入引脚的处理 937	SSL 无效期间的 MOSI 信号值的决定方法 537
模拟输入引脚的规格 938	扫描模式 931
模式和 SPCR 设定的关系 536	扇区存取模式 962
目标扇区的缓冲功能 917	上电复位 95
	上电顺序 231
N	设备状态转移中断 1048
NAND 闪存控制器 940	设定周期时的注意事项 407
NAND 闪存控制器的时序 1945	舍入 75
NAND 闪存控制器的中断请求 965	深度待机模式 1729
NRDY 中断 1044	深度掉电模式 234
NYET 握手信号应答表 1067	时戳 745
闹钟功能的使用方法 473	时间触发发送 790
内部 RAM 1618	时间触发控制 (TT 控制) 746
内部外围模块请求模式 275	时间触发系统的例子 796
内部外围模块中断 118	时间从属 793
内存映射高速缓存的结构 158	视频显示控制器 4 的时序 1951
能支持的 DMA 传送 278	实时时钟 456
	事务发行条件 1077
O	使用 PLL 振荡电路时的注意事项 89, 90
OpenVG 瑞萨图形处理器 1498	使用直接存储器存取控制器时的注意事项 514
	时钟脉冲振荡器 83
	时钟模式 86
	时钟频率的控制电路 85

时钟同步串行格式	661
时钟同步模式中的运行	506
手动复位	95
输出负荷电路	1957
数据存取周期的断点	146
数据格式	542
数据 PID 顺序位	1056
数据阵列	151, 158
数据传送指令	57
输入 JPEG 符号数据	1531
输入数据流字节序的转换功能	910
刷新请求和总线周期的关系	230
双地址模式	279
双精度浮点格式	70
睡眠模式	1726
算术运算指令	60

T

TAP 控制器	1743
TDO 输出时序	1744
TSG 和 TSEG 的设置	757
TTW[1:0] (时间触窗口)	746
T 位	46
停机模式	786
停止和重新开始 CD-DSP 时的注意点	921
同步码保护功能	911
通过中断请求信号进行的数据传送	135
通用寄存器	41
通用寄存器的初始值	43
通用输入 / 输出端口	1622
通用输入 / 输出端口的时序	1955
突发读	219
突发模式	283
突发 ROM (时钟同步) 接口	245
突发 ROM (异步) 接口	235
突发写	222
退栈	134

U

USB2.0 主机 / 功能模块	968
USB2.0 主机 / 功能模块的时序	1949
USB 数据总线的电阻控制	1036

V

VBUS 中断	1050
---------------	------

W

外部触发的输入时序	934
外部请求模式	274
位操作指令	68
微处理单元接口 (MPI)	734
位同步电路	668
位移量的参照	47

无延迟槽的无条件转移指令	46
--------------------	----

X

系统寄存器的初始值	43
系统结构例子	538
系统矩阵	748
系统控制指令	65
陷阱指令	99
向量基址寄存器 (VBR)	42
相位计数模式	353
校正子计算功能	914
写地址阵列 (无关联)	158
写地址阵列 (有关联)	158
写数据阵列	159
信息发送请求	789, 795
信息接收顺序	798
信息控制字段	740
信息数据字段	744
信箱	734, 737
信箱的重新设定	799
信箱的作用	738
信箱功能的设定	742
信箱控制	734
选择功能控制器功能时的控制传送	1066
选择主机控制器功能时的控制传送	1065
寻址方式	48

Y

压栈	134
延迟转移指令	45
页面竞争	1620
一般非法指令	100
异步模式的接收数据采样时序和接收容限	515
异步模式中的运行	498
异常处理	91
异常处理的种类和优先级	91
异常处理后的堆栈状态	101
异常处理向量表	93, 119
异常处理向量表地址的计算方法	94
异常处理状态	69
异常传送的发生条件和错误检测功能	548
移位指令	63
引导模式	77
引脚的控制	537
音频发生器	1554
应答 PID	1055
用户断点控制器	137
用户断点运行的流程	145
用户调试接口	1735
用户调试接口的时序	1955
用户调试接口复位	1745
用户调试接口中断	116, 1745
用于保持的内部 RAM 的地址空间	1618, 1619
有效地址的计算方法	48
由指令引起的异常	99

预取操作（只限于操作数高速缓存）	156
------------------------	-----

Z

在紧接着延迟转移指令之后发生的异常源	101
噪声消除电路	663
帧号更新中断	1050
帧缓冲器模式（压缩）	1523
正规空间接口	201
整数除法运算异常	100
直接存储器存取控制器的启动	397
直接存储器存取控制器	253
直接存储器存取控制器接口	802
指令的特点	45
指令格式	51
指令系统	53
中断控制器	104
中断响应时间	127
中断异常处理	98
中断异常处理结束后的堆栈状态	126
中断优先级	98
中断源	513
中断源的清除时序	136
中断传送	1068
中止的发送	514
中止的检测和处理	514
周期挪用模式	282
主控发送	655
主控接收	656
主控模式的运行	551
转移表基址寄存器（TBR）	42
转移指令	64
装入 - 存储结构	45
状态寄存器（SR）	42
自动解码的停止功能	915
自动请求模式	274
自动刷新	228
字节序	196
字数据的符号扩展	45
自刷新	229
总线权释放状态	69
总线仲裁	250
总线状态控制器	161

修订记录	SH7268 群、SH7269 群 用户手册 硬件篇
------	----------------------------

Rev.	发行日	修订内容	
		页	修订处
1.00	2011.09.26	—	初版发行

SH7268 群、SH7269 群
用户手册 硬件篇

Publication Date: Rev.1.00 Sep 26, 2011

Published by: Renesas Electronics Corporation

**SALES OFFICES****Renesas Electronics Corporation**<http://www.renesas.com>Refer to "<http://www.renesas.com/>" for the latest and detailed information.**Renesas Electronics America Inc.**2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130**Renesas Electronics Canada Limited**1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220**Renesas Electronics Europe Limited**Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900**Renesas Electronics Europe GmbH**Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327**Renesas Electronics (China) Co., Ltd.**7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679**Renesas Electronics (Shanghai) Co., Ltd.**Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898**Renesas Electronics Hong Kong Limited**Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044**Renesas Electronics Taiwan Co., Ltd.**13F, No. 363, Fu Shing North Road, Taipei, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670**Renesas Electronics Singapore Pte. Ltd.**1 harbourFront Avenue, #06-10, Keppel Bay Tower, Singapore 098632
Tel: +65-6213-0200, Fax: +65-6278-8001**Renesas Electronics Malaysia Sdn.Bhd.**Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510**Renesas Electronics Korea Co., Ltd.**11F., Samik Lavied' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

SH7268群、SH7269群



瑞萨电子株式会社

R01UH0048CJ0100