

致尊敬的顾客

关于产品目录等资料中的旧公司名称

NEC电子公司与株式会社瑞萨科技于2010年4月1日进行业务整合（合并），整合后的新公司暨“瑞萨电子公司”继承两家公司的所有业务。因此，本资料中虽还保留有旧公司名称等标识，但是并不妨碍本资料的有效性，敬请谅解。

瑞萨电子公司网址：<http://www.renesas.com>

2010年4月1日
瑞萨电子公司

【发行】瑞萨电子公司（<http://www.renesas.com>）

【业务咨询】<http://www.renesas.com/inquiry>

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.

H8S/2282 群、H8S/2280 群

瑞萨 16 位单片机

H8S 族 / H8S/2200 系列

H8S/2282F	HD64F2282
H8S/2282	HD6432282
H8S/2281	HD6432281
H8S/2280F	HD64F2280B
	HD64F2280RB

Notes regarding these materials

1. This document is provided for reference purposes only so that Renesas customers may select the appropriate Renesas products for their use. Renesas neither makes warranties or representations with respect to the accuracy or completeness of the information contained in this document nor grants any license to any intellectual property rights or any other rights of Renesas or any third party with respect to the information in this document.
2. Renesas shall have no liability for damages or infringement of any intellectual property or other rights arising out of the use of any information in this document, including, but not limited to, product data, diagrams, charts, programs, algorithms, and application circuit examples.
3. You should not use the products or the technology described in this document for the purpose of military applications such as the development of weapons of mass destruction or for the purpose of any other military use. When exporting the products or technology described herein, you should follow the applicable export control laws and regulations, and procedures required by such laws and regulations.
4. All information included in this document such as product data, diagrams, charts, programs, algorithms, and application circuit examples, 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 products listed in this document, please confirm the latest product information with a Renesas sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas such as that disclosed through our website. (<http://www.renesas.com>)
5. Renesas has used reasonable care in compiling the information included in this document, but Renesas assumes no liability whatsoever for any damages incurred as a result of errors or omissions in the information included in this document.
6. When using or otherwise relying on the information in this document, you should evaluate the information in light of the total system before deciding about the applicability of such information to the intended application. Renesas makes no representations, warranties or guaranties regarding the suitability of its products for any particular application and specifically disclaims any liability arising out of the application and use of the information in this document or Renesas products.
7. With the exception of products specified by Renesas as suitable for automobile applications, Renesas products are not designed, manufactured or tested for applications or otherwise in systems the failure or malfunction of which may cause a direct threat to human life or create a risk of human injury or which require especially high quality and reliability such as safety systems, or equipment or systems for transportation and traffic, healthcare, combustion control, aerospace and aeronautics, nuclear power, or undersea communication transmission. If you are considering the use of our products for such purposes, please contact a Renesas sales office beforehand. Renesas shall have no liability for damages arising out of the uses set forth above.
8. Notwithstanding the preceding paragraph, you should not use Renesas products for the purposes listed below:
 - (1) artificial life support devices or systems
 - (2) surgical implantations
 - (3) healthcare intervention (e.g., excision, administration of medication, etc.)
 - (4) any other purposes that pose a direct threat to human lifeRenesas shall have no liability for damages arising out of the uses set forth in the above and purchasers who elect to use Renesas products in any of the foregoing applications shall indemnify and hold harmless Renesas Technology Corp., its affiliated companies and their officers, directors, and employees against any and all damages arising out of such applications.
9. You should use the products described herein within the range specified by Renesas, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas shall have no liability for malfunctions or damages arising out of the use of Renesas products beyond such specified ranges.
10. Although Renesas endeavors to improve the quality and reliability of its products, IC products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Please be sure to implement safety measures to guard against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas 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 applicable measures. Among others, since the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
11. In case Renesas products listed in this document are detached from the products to which the Renesas products are attached or affixed, the risk of accident such as swallowing by infants and small children is very high. You should implement safety measures so that Renesas products may not be easily detached from your products. Renesas shall have no liability for damages arising out of such detachment.
12. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written approval from Renesas.
13. Please contact a Renesas sales office if you have any questions regarding the information contained in this document, Renesas semiconductor products, or if you have any other inquiries.

注意

本文只是参考译文，前页所载英文版“Cautions”具有正式效力。

关于利用本资料时的注意事项

1. 本资料是为了让用户根据用途选择合适的本公司产品的参考资料，对于本资料中所记载的技术信息，并非意味着对本公司或者第三者的知识产权及其他权利做出保证或对实施权力进行的承诺。
2. 对于因使用本资料所记载的产品数据、图、表、程序、算法及其他应用电路例而引起的损害或者对第三者的知识产权及其他权利造成侵犯，本公司不承担任何责任。
3. 不能将本资料所记载的产品和技术用于大规模破坏性武器的开发等目的、军事目的或其他的军需用途方面。另外，在出口时必须遵守日本的《外汇及外国贸易法》及其他出口的相关法令并履行这些法令中规定的必要手续。
4. 本资料所记载的产品数据、图、表、程序、算法以及其他应用电路例等所有信息均为本资料发行时的内容，本公司有可能在未做事先通知的情况下，对本资料所记载的产品或者产品规格进行更改。所以在购买和使用本公司的半导体产品之前，请事先向本公司的营业窗口确认最新的信息并经常留意本公司通过公司主页（<http://www.renesas.com>）等公开的最新信息。
5. 对于本资料中所记载的信息，制作时我们尽力保证出版时的精确性，但不承担因本资料的叙述不当而使顾客遭受损失等的任何相关责任。
6. 在使用本资料所记载的产品数据、图、表等所示的技术内容、程序、算法及其他应用电路例时，不仅要对所使用的技术信息进行单独评价，还要对整个系统进行充分的评价。请顾客自行负责，进行是否适用的判断。本公司对于是否适用不负任何责任。
7. 本资料中所记载的产品并非针对万一出现故障或是错误运行就会威胁到人的生命或给人体带来危害的机器、系统（如各种安全装置或者运输交通用的、医疗、燃烧控制、航天器械、核能、海底中继用的机器和系统等）而设计和制造的，特别是对于品质和可靠性要求极高的机器和系统等（将本公司指定用于汽车方面的产品用于汽车时除外）。如果要用于上述的目的，请务必事先向本公司的营业窗口咨询。另外，对于用于上述目的而造成的损失等，本公司概不负责。
8. 除上述第7项内容外，不能将本资料中记载的产品用于以下用途。如果用于以下用途而造成的损失，本公司概不负责。
 - 1) 生命维持装置。
 - 2) 植埋于人体使用的装置。
 - 3) 用于治疗（切除患部、给药等）的装置。
 - 4) 其他直接影响到人的生命的装置。
9. 在使用本资料所记载的产品时，对于最大额定值、工作电源电压的范围、放热特性、安装条件及其他条件请在本公司规定的保证范围内使用。如果超出了本公司规定的保证范围使用时，对于由此而造成的故障和出现的事故，本公司将不承担任何责任。
10. 本公司一直致力于提高产品的质量和可靠性，但一般来说，半导体产品总会以一定的概率发生故障、或者由于使用条件不同而出现错误运行等。为了避免因本公司的产品发生故障或者错误运行而导致人身事故和火灾或造成社会性的损失，希望客户能自行负责进行冗余设计、采取延烧对策及进行防止错误运行等的安全设计（包括硬件和软件两方面的设计）以及老化处理等，这是作为机器和系统的出厂保证。特别是单片机的软件，由于单独进行验证很困难，所以要求在顾客制造的最终的机器及系统上进行安全检验工作。
11. 如果把本资料所记载的产品从其载体设备上卸下，有可能造成婴儿误吞的危险。顾客在将本公司产品安装到顾客的设备上时，请顾客自行负责将本公司产品设置为不容易剥落的安全设计。如果从顾客的设备上剥落而造成事故时，本公司将不承担任何责任。
12. 在未得到本公司的事先书面认可时，不可将本资料的一部分或者全部转载或者复制。
13. 如果需要了解关于本资料的详细内容，或者有其他关心的问题，请向本公司的营业窗口咨询。

产品使用时的注意事项

本文对适用于所有单片机产品的“使用注意事项”进行了说明。个别注意事项请参照手册正文。当本文与手册正文中的记载不同时，请优先参照正文中的内容。

1. 未使用管脚的处理方法

【注意】对于未使用的管脚，请按照手册正文中所记载的“未使用管脚的处理”进行处理。

CMOS 产品的输入管脚的阻抗一般为高阻抗。若在开路状态运行未使用管脚时，由于电感现象，会产生 LSI 外围噪音，而在 LSI 内部产生的直通电流等会被认为是输入信号，从而引起误动作。对于未使用的管脚，请按照手册正文中“未使用管脚的处理”中的说明进行处理。

2. 接通电源时的处理方法

【注意】接通电源时，产品状态不稳定。

接通电源时，LSI 的内部电路为不确定状态，寄存器的设定和各管脚为不稳定状态。

通过外部复位管脚进行复位的产品，从接通电源到复位有效为止，不能保证管脚的状态。

同样，通过内部加电复位功能进行复位的产品，从接通电源到达到复位所需电压为止，不能保证管脚的状态。

3. 禁止对保留地址进行存取

【注意】禁止对保留地址进行存取。

在地址区中存在用于将来扩展功能而分配的保留地址。对这些地址进行存取时，不能保证运行的正常进行，因此不能存取保留地址。

4. 关于时钟的注意事项

【注意】复位时，必须在时钟稳定后再解除复位。

在程序执行过程中切换时钟时，必须等需要切换的时钟稳定后再对其进行切换。

复位时，在通过带有外部振荡器（或者外部振荡电路）的时钟开始运行的系统中，必须等时钟完全稳定后，再解除复位。另外，在程序执行过程中切换为带有外部振荡器（或者外部振荡电路）的时钟时，必须等需要切换的时钟完全稳定后再对其进行切换。

5. 关于产品间差异的注意事项

【注意】当产品型号发生变化时，请事先确认是否存在问题。

即使是同一群内的单片机，产品型号如果不同，也有可能由于内部存储器和布线图的不同产生特性的差异。当产品型号发生变化时，必须对每个型号的产品进行系统评价测试。

本手册的篇章构成

本手册包含以下条目：

1. 产品操作一般注意事项
2. 本手册的篇章结构
3. 前言
4. 目录
5. 概要
6. 各功能模块的说明
 - CPU 及系统控制模块
 - 片上外围模块

各模块的功能结构描述针对不同的模块而会略有不同，但一般包含以下条目：

①特点、②输入 / 输出管脚、③寄存器描述、④操作说明、⑤使用时的注意事项。

在设计采用本 LSI 的应用系统时，请事先充分确认注意事项。

请务必阅读个章节中有关说明中的注意事项及个章节最后提示的注意事项。

（使用注意事项一般根据需要记载于本手册中。）

7. 寄存器一览表
8. 电特性
9. 附录
10. 此版本修订的主要内容（只适用于修订版）

修订目录是总结前版修订的主要内容，并没有记载所有的修订内容，
有关详细内容请在本手册的正文中查看。

11. 索引

前言

本 LSI 是以高速 H8S/2000 CPU 为核心，集成了系统构成所必需的外围功能的单片机。

它包括系统构成必须的外围功能，如 ROM、RAM、16 位定时器、看门狗定时器、串行通信接口、CAN、A/D 转换器、PWM 控制器、LCD 控制器 / 驱动器、时钟振荡电路、I/O 端口等，可作为高等级控制系统的嵌入式单片机使用。片上 ROM 带有单一电源闪存 (F-ZTATTM)*，可随时编程，迅速且灵活地适应规格流动性大的产品在量产初期到大规模量产过程中所发生的各种变化。

【注】 * F-ZTAT 是瑞萨科技 (有限公司) 的商标。

对象 本手册以 “在应用系统设计中采用 H8S/2282 群及 H8S/2280 群” 的用户为对象。

使用本手册的读者必须具备有关电路，逻辑电路及单片机的基础知识。

目的 本手册是以 “能让用户理解 H8S/2282 群、H8S/2280 群的硬件功能和电特性” 为目的。关于指令执

行的详细内容请参考 《H8S/2600 系列、H8S/2000 系列软件手册》。

阅读方法

- 希望了解全部芯片的功能时
→ 请按照目录的顺序阅读
本手册按照 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 《H8S/2600 系列、H8S/2000 系列软件手册》。
- 若知道寄存器名称，而希望了解详细功能时
→ 在本书后面附有 “索引”。请从索引搜索页号。
《第 21 章 寄存器一览表》有关地址、位内容及初始化值，汇总在 《第 21 章 寄存器一览表》。

范例 寄存器名称 : 多个通道带有串行通信接口等相同或类似的功能，
将采用以下的标记方法。

XXX_N (XXX 是基本寄存器的名称、N 是通道编号)

位标记顺序 : 左侧为高位、右侧为低位。

数字的标记 : 2 进制为 B'xxxx、16 进制为 H'xxxx、10 进制为 xxxx

信号的标记 : 低电平有效的信号带有上划线 xxx。

目 录

第 1 章 概要	1
1.1 特点	1
1.2 内部框图	2
1.3 管脚排列	5
1.4 管脚功能	8
1.4.1 H8S/2282 群、H8S/2280 群 (HD64F2280B) 管脚功能	8
1.4.2 H8S/2280 群 (HD64F2280RB) 的管脚功能	14
第 2 章 CPU	20
2.1 特点	20
2.1.1 H8S/2000 CPU 与 H8S/2600 CPU 的区别	21
2.1.2 H8S/2000 CPU 与 H8/300 CPU 的区别	21
2.1.3 H8S/2000 CPU 与 H8/300H CPU 的区别	22
2.2 CPU 运行模式	22
2.2.1 普通模式	22
2.2.2 高级模式	24
2.3 地址空间	25
2.4 寄存器构成	26
2.4.1 通用寄存器	27
2.4.2 程序计数器 (PC)	28
2.4.3 扩展寄存器 (EXR)	28
2.4.4 条件码寄存器 (CCR)	29
2.4.5 CPU 内部寄存器初始值	29
2.5 数据格式	30
2.5.1 通用寄存器的数据格式	30
2.5.2 存储器上的数据格式	32
2.6 指令系统	33
2.6.1 指令功能区别一览	34
2.6.2 基本指令格式	42
2.7 寻址模式和有效地址的计算方法	43
2.7.1 寄存器直接寻址 Rn	43
2.7.2 寄存器间接寻址 @ERn	43
2.7.3 带偏移量的寄存器间接寻址 @ (d:16,ERn) / @ (d:32,ERn)	43
2.7.4 后增寄存器间接寻址 @ERn+ / 先减寄存器间接寻址 @-ERn	44
2.7.5 绝对地址 @aa:8 / @aa:16 / @aa:24 / @aa:32	44
2.7.6 立即数寻址 #xx:8 / #xx:16 / #xx:32	45
2.7.7 程序计数器相对寻址 @ (d:8, PC) / @ (d:16, PC)	45
2.7.8 存储器间接寻址 @@aa:8	45
2.7.9 有效地址的计算方法	46
2.8 处理状态	48
2.9 使用注意事项	49
2.9.1 位操作指令使用注意事项	49
第 3 章 MCU 工作模式	50
3.1 工作模式的选择	50
3.2 寄存器的说明	50
3.2.1 模式控制寄存器 (MDCR)	50
3.2.2 系统控制寄存器 (SYSCR)	51
3.3 各工作模式的说明	52

3.4	地址映射	52
第 4 章	异常处理	54
4.1	异常处理的种类和优先级	54
4.2	异常源 和向量表	55
4.3	复位	56
4.3.1	复位异常处理	56
4.3.2	复位后的中断	57
4.3.3	复位解除后的内部外围功能	57
4.4	跟踪异常处理	58
4.5	中断异常处理	58
4.6	陷阱指令异常处理	59
4.7	异常处理后的堆栈状态	60
4.8	使用中的注意事项	61
第 5 章	中断控制器	62
5.1	特点	62
5.2	输入 / 输出管脚	63
5.3	寄存器的说明	63
5.3.1	中断优先级寄存器 A ~ G、J、K、M (IPRA ~ IPRG、IPRJ、IPRK、IPRM)	64
5.3.2	IRQ 允许寄存器 (IER)	65
5.3.3	IRQ 检测控制寄存器 H、L (ISCRH、ISCRL)	65
5.3.4	IRQ 状态寄存器 (ISR)	67
5.4	中断源	67
5.4.1	外部中断源	67
5.4.2	内部中断	68
5.5	中断异常处理向量表	68
5.6	中断控制模式和中断运行	70
5.6.1	中断控制模式 0	70
5.6.2	中断控制模式 2	72
5.6.3	中断异常处理顺序	74
5.6.4	中断响应时间	75
5.7	使用时的注意事项	76
5.7.1	中断发生与中断屏蔽的竞争	76
5.7.2	禁止中断指令	77
5.7.3	中断禁止期间	77
5.7.4	EEPMOV 指令执行中的中断。	77
5.7.5	有关 IRQ 中断	77
第 6 章	总线控制器	78
6.1	基本运行时序	78
6.1.1	内部存储器存取时序 (ROM, RAM)	78
6.1.2	内部外围模块存取时序	79
6.1.3	内部 HCAN 模块存取时序	79
6.1.4	内部 PWM、LCD、端口 H、J 模块存取时序	80
第 7 章	I/O 端口	81
7.1	端口 1	87
7.1.1	数据方向寄存器 (P1DDR)	87
7.1.2	数据寄存器 (P1DR)	87
7.1.3	端口 1 寄存器 (PORT1)	88
7.1.4	管脚功能	88

7.2	端口 3	96
7.2.1	数据方向寄存器 (P3DDR)	96
7.2.2	数据寄存器 (P3DR)	96
7.2.3	端口 3 寄存器 (PORT3)	97
7.2.4	漏极开路控制寄存器 (P3ODR)	97
7.2.5	管脚功能	98
7.3	端口 4	100
7.3.1	端口 4 寄存器 (PORT4)	100
7.3.2	端口功能	100
7.4	端口 A	101
7.4.1	数据方向寄存器 (PADDDR)	101
7.4.2	端口 A 数据寄存器 (PADR)	101
7.4.3	端口 A 寄存器 (PORTA)	102
7.4.4	漏极开路控制寄存器 (PAODR)	102
7.4.5	H8S/2282 群、H8S/2280 群 (HD64F2280B) 的管脚功能	103
7.4.6	H8S/2280 群 (HD64F2280RB) 的管脚功能	103
7.5	端口 B	104
7.5.1	端口 B 数据方向寄存器 (PBDDR)	104
7.5.2	端口 B 数据寄存器 (PBDR)	104
7.5.3	端口 B 寄存器 (PORTB)	105
7.5.4	端口 B 漏极开路控制寄存器 (PBODR)	105
7.5.5	H8S/2282 群、H8S/2280 群 (HD64F2280B) 的管脚功能	106
7.5.6	H8S/2280 群 (HD64F2280RB) 的管脚功能	106
7.6	端口 C	107
7.6.1	端口 C 数据方向寄存器 (PCDDR)	107
7.6.2	端口 C 数据寄存器 (PCDR)	107
7.6.3	端口 C 寄存器 (PORTC)	108
7.6.4	端口 C 漏极开路控制寄存器 (PCODR)	108
7.6.5	H8S/2282 群、H8S/2280 群 (HD64F2280B) 的管脚功能	109
7.6.6	H8S/2280 群 (HD64F2280RB) 的管脚功能	110
7.7	端口 D	110
7.7.1	端口 D 数据方向寄存器 (PDDDR)	110
7.7.2	端口 D 数据寄存器 (PDDR)	111
7.7.3	端口 D 寄存器 (PORTD)	111
7.7.4	H8S/2282 群、H8S/2280 群 (HD64F2280B) 的管脚功能	111
7.7.5	H8S/2280 群 (HD64F2280RB) 的管脚功能	112
7.8	端口 F	112
7.8.1	端口 F 数据方向寄存器 (PFDDR)	112
7.8.2	端口 F 数据寄存器 (PFDR)	113
7.8.3	端口 F 寄存器 (PORTF)	113
7.8.4	H8S/2282 群、H8S/2280 群 (HD64F2280B) 的管脚功能	114
7.8.5	H8S/2280 群 (HD64F2280RB) 的管脚功能	116
7.9	端口 H	118
7.9.1	端口 H 数据方向寄存器 (PHDDR)	118
7.9.2	端口 H 数据寄存器 (PHDR)	118
7.9.3	端口 H 寄存器 (PORTH)	119
7.9.4	管脚功能	119
7.10	端口 J	121
7.10.1	端口 J 数据方向寄存器 (PJDDR)	121
7.10.2	端口 J 数据寄存器 (PJDR)	121
7.10.3	端口 J 寄存器 (PORTJ)	122
7.10.4	管脚功能	122

7.11 管脚切换功能	124
7.11.1 传送寄存器 (TRPRT)	124
7.11.2 通过管脚的切换, 读端口寄存器。	125
第 8 章 16 位定时器脉冲单元 (TPU)	126
8.1 特点	126
8.2 输入 / 输出管脚	129
8.3 寄存器说明	130
8.3.1 定时器控制寄存器 (TCR)	131
8.3.2 计时器模式寄存器 (TMDR)	134
8.3.3 定时器 I/O 控制寄存器 (TIO)	135
8.3.4 定时器中断允许寄存器 (TIER)	144
8.3.5 定时器状态寄存器 (TSR)	145
8.3.6 定时器计数器 (TCNT)	147
8.3.7 定时器通用寄存器 (TGR)	147
8.3.8 定时器起始寄存器 (TSTR)	147
8.3.9 定时器同步寄存器 (TSYR)	147
8.4 工作说明	148
8.4.1 基本工作	148
8.4.2 同步工作	153
8.4.3 缓冲工作	155
8.4.4 PWM 模式	158
8.4.5 相位计数模式	162
8.5 中断条件	168
8.6 A/D 转换器的启动	169
8.7 工作时序	169
8.7.1 输入 / 输出时序	169
8.7.2 中断信号的时序	173
8.8 用时的注意事项	175
8.8.1 模块待机模式的设定	175
8.8.2 输入时钟的限制事项	175
8.8.3 周期设定时的注意事项	176
8.8.4 TCNT 的写和清除的竞争	176
8.8.5 TCNT 的写和累加计数的竞争	176
8.8.6 TGR 的写和比较匹配的竞争	177
8.8.7 缓冲寄存器的写和比较匹配的竞争	177
8.8.8 TGR 的读和输入捕捉的竞争	178
8.8.9 TGR 的写和输入捕捉的竞争	178
8.8.10 缓冲寄存器的写和输入捕捉的竞争	179
8.8.11 溢出 / 下溢和计数器清除的竞争	179
8.8.12 TCNT 的写和溢出 / 下溢的竞争	180
8.8.13 输入 / 输出管脚的兼用	180
8.8.14 模块停止时的中断	180
8.8.15 子激活模式 / 监视模式的转换	180
第 9 章 看门狗定时器 (WDT)	181
9.1 特点	181
9.2 寄存器的说明	182
9.2.1 定时器计数器 0、1 (TCNT_0、TCNT_1)	182
9.2.2 定时器控制 / 状态寄存器 0、1 (TCSR_0、TCSR_1)	183
9.2.3 复位控制 / 状态寄存器 (RSTCSR)	186
9.3 工作说明	186

9.3.1	看门狗定时器模式	186
9.3.2	间隔定时器模式	188
9.4	中断源	188
9.5	使用时的注意事项	189
9.5.1	寄存器存取时的注意事项	189
9.5.2	定时计数器 (TCNT) 的写和累加计数的竞争	190
9.5.3	CKS2 ~ CKS0 位的改写	190
9.5.4	看门狗定时器模式和间隔定时器模式的切换	190
9.5.5	看门狗定时器模式下的内部复位	190
9.5.6	间隔定时器模式下 OVF 标志的清除	190
第 10 章	串行通信接口 (SCI)	191
10.1	特点	191
10.2	输入 / 输出管脚	193
10.3	寄存器说明	193
10.3.1	接收移位寄存器 (RSR)	193
10.3.2	接收数据寄存器 (RDR)	193
10.3.3	发送数据寄存器 (TDR)	193
10.3.4	发送移位寄存器 (TSR)	194
10.3.5	串行模式寄存器 (SMR)	194
10.3.6	串行控制寄存器 (SCR)	196
10.3.7	串行通信状态寄存器 (SSR)	198
10.3.8	智能卡接口模式寄存器 (SCMR)	200
10.3.9	位速率寄存器 (BRR)	201
10.4	异步模式的运行	206
10.4.1	发送 / 接收格式	207
10.4.2	异步模式下接收数据的采样时序和接收容限	208
10.4.3	时钟	209
10.4.4	SCI 的初始化 (异步模式)	209
10.4.5	发送数据 (异步模式)	210
10.4.6	接收串行数据 (异步模式)	212
10.5	多处理器通信功能	215
10.5.1	多处理器模式串行数据的发送	216
10.5.2	多处理器模式串行数据的接收	217
10.6	时钟同步模式	219
10.6.1	时钟	220
10.6.2	SCI 的初始化	220
10.6.3	串行数据的发送 (时钟同步模式)	221
10.6.4	接收串行数据 (时钟同步模式)	223
10.6.5	发送 / 接收串行数据的同步运行 (时钟同步式)	225
10.7	智能卡接口操作说明	226
10.7.1	管脚连接图例	226
10.7.2	数据格式 (除数据块传送模式)	226
10.7.3	数据块传送模式	227
10.7.4	采样接收数据时序和智能卡接口模式的接收容限	228
10.7.5	初始设定	229
10.7.6	发送数据 (除数据块传送模式之外)	230
10.7.7	串行数据的接收 (除过数据块传送模式)	232
10.7.8	时钟输出控制	233
10.8	中断源	234
10.8.1	普通串行通信接口模式的中断	234
10.8.2	智能卡接口模式下的中断	235

10.9	使用时的注意事项	235
10.9.1	模块待机模式的设定	235
10.9.2	关于断点的检测和处理	235
10.9.3	标志状态和中断的发送	235
10.9.4	接收错误标志和发送运行（仅对时钟同步模式有效）	235
10.9.5	模式迁移操作	236
10.9.6	从 SCK 管脚向端口管脚切换时的注意事项	238
第 11 章	CAN 总线模块（HCAN）【只对应 H8S/2282 群】	240
11.1	特点	240
11.2	输入 / 输出管脚	242
11.3	寄存器的说明	242
11.3.1	主控制寄存器（MCR）	243
11.3.2	通用状态寄存器（GSR）	244
11.3.3	位配置寄存器（BCR）	245
11.3.4	信箱配置寄存器（MBCR）	247
11.3.5	发送等待寄存器（TXPR）	247
11.3.6	取消发送等待寄存器（TXCR）	248
11.3.7	发送应答寄存器（TXACK）	248
11.3.8	取消应答寄存器（ABACK）	249
11.3.9	接收完成寄存器（RXPR）	249
11.3.10	远程请求寄存器（RFPR）	250
11.3.11	中断寄存器（IRR）	251
11.3.12	信箱中断屏蔽寄存器（MBIMR）	253
11.3.13	中断屏蔽寄存器（IMR）	254
11.3.14	接收错误计数器（REC）	255
11.3.15	发送错误计数器（TEC）	255
11.3.16	未读信息状态寄存器（UMSR）	255
11.3.17	局部验收滤波器掩码（LAFML、LAFMH）	256
11.3.18	信息控制（MC0 ~ MC15）	257
11.3.19	信息数据（MD0 ~ MD15）	259
11.4	运行说明	259
11.4.1	硬件复位和软件复位	259
11.4.2	硬件复位后的初始设定	259
11.4.3	发送信息	265
11.4.4	接收信息	268
11.4.5	HCAN 休眠模式	271
11.4.6	HCAN HALT 模式	272
11.5	中断源	273
11.6	CAN 总线接口	273
11.7	使用时的注意事项	274
11.7.1	模块待机模式的设定	274
11.7.2	复位	274
11.7.3	HCAN 休眠模式	274
11.7.4	中断	274
11.7.5	错误计数器	274
11.7.6	存取寄存器	274
11.7.7	HCAN 的中速模式	274
11.7.8	待机时的寄存器保持	274
11.7.9	位操作命令的使用	274
11.7.10	关于 HCAN 的 TXCR 运行	275
11.7.11	关于 HCAN 发送过程	276

11.7.12	HCAN 软件复位及 HCAN 休眠解除	276
11.7.13	HCAN 休眠中的信箱存取	276
第 12 章	A/D 转换器	277
12.1	特点	277
12.2	输入 / 输出管脚	279
12.3	寄存器说明	280
12.3.1	A/D 数据寄存器 A ~ D (ADDRA ~ ADDR D)	280
12.3.2	A/D 控制 / 状态寄存器 (ADCSR)	281
12.3.3	A/D 控制寄存器 (ADCR)	282
12.4	运行说明	282
12.4.1	单通道模式	282
12.4.2	扫描模式	283
12.4.3	输入采样和 A/D 转换时间	283
12.4.4	外部触发器输入时序	284
12.5	中断源	285
12.6	A/D 转换精度的定义	285
12.7	使用时的注意事项	286
12.7.1	模块待机模式的设定	286
12.7.2	关于容许信号源阻抗	286
12.7.3	对绝对精度的影响	286
12.7.4	模拟电源管脚其他的设定范围	287
12.7.5	电路板设计时的注意事项	287
12.7.6	噪声对策上的注意事项	287
第 13 章	电机控制 PWM 定时器 (PWM)	289
13.1	特点	289
13.2	输入 / 输出管脚	291
13.3	寄存器说明	291
13.3.1	PWM 控制寄存器 _1、2 (PWCR_1、PWCR_2)	292
13.3.2	PWM 输出控制寄存器 _1、2 (PWOCR_1、PWOCR_2)	293
13.3.3	PWM 极性寄存器 _1、2 (PWPR_1、PWPR_2)	294
13.3.4	PWM 计数器 _1、2 (PWCNT_1、PWCNT_2)	294
13.3.5	PWM 周期寄存器 _1、2 (PWCYR_1、PWCYR_2)	295
13.3.6	PWM 占空比寄存器 _1A、1C、1E、1G (PWDTR_1A、PWDTR_1C、PWDTR_1E、PWDTR_1G)	296
13.3.7	PWM 缓冲寄存器 _1A、1C、1E、1G (PWBFR_1A、PWBFR_1C、PWBFR_1E、PWBFR_1G)	298
13.3.8	PWM 占空比寄存器 _2A ~ 2H (PWDTR_2A ~ PWDTR_2H)	299
13.3.9	PWM 缓冲寄存器 _2A ~ 2D (PWBFR_2A ~ PWBFR_2D)	301
13.4	与总线主控器的接口	302
13.4.1	16 位数据寄存器	302
13.4.2	8 位数据寄存器	302
13.5	运行说明	303
13.5.1	PWM 通道 1 的运行	303
13.5.2	PWM 通道 2 的运行	304
13.6	中断源	305
13.7	使用时的注意事项	305
第 14 章	LCD 控制器 / 驱动器 (LCD)	306
14.1	特点	306
14.2	输入 / 输出管脚	307

14.3 各寄存器的说明	308
14.3.1 LCD 端口控制寄存器 (LPCR)	308
14.3.2 LCD 控制寄存器 (LCR)	310
14.3.3 LCD 控制寄存器 2 (LCR2)	311
14.4 运行说明	312
14.4.1 LCD 显示的设置	312
14.4.2 LCD RAM 和显示的关系	313
14.4.3 低功耗模式的运行	318
14.4.4 增强 LCD 驱动电源	318
14.5 使用时的注意事项	319
14.5.1 禁止 LCD 显示	319
第 15 章 RAM	320
第 16 章 快速擦写存储器 (F-ZTAT 版)【H8S/2282 群】	321
16.1 特点	321
16.2 模式转换	322
16.3 块结构	325
16.4 输入 / 输出管脚	325
16.5 寄存器的说明	326
16.5.1 快速擦写存储器控制寄存器 1 (FLMCR1)	326
16.5.2 快速擦写存储器控制寄存器 2 (FLMCR2)	326
16.5.3 擦除块指定寄存器 1、2 (EBR1、EBR2)	327
16.5.4 RAM 仿真寄存器 (RAMER)	327
16.5.5 快速擦写存储器功率控制寄存器 (FLPWCR)	328
16.6 板载编程	328
16.6.1 引导模式	329
16.6.2 用户编程模式的编程 / 擦除	331
16.7 快速擦写存储器的 RAM 仿真	332
16.8 快速擦写存储器的编程 / 擦除	334
16.8.1 程序 / 程序验证	334
16.8.2 擦除 / 擦除验证	336
16.8.3 快速擦写存储器编程 / 擦除时的中断	336
16.9 编程 / 擦除保护	338
16.9.1 硬件保护	338
16.9.2 软件保护	338
16.9.3 错误保护	338
16.10 PROGRAMMER MODE	338
16.11 快速擦写存储器的低功耗状态	339
16.12 快速擦写存储器和低功耗状态	339
第 17 章 快速擦写存储器 (F-ZTAT 版)【H8S/2280 群】	340
17.1 特点	340
17.2 模式转换	341
17.3 块结构	344
17.4 输入 / 输出管脚	344
17.5 寄存器的说明	345
17.5.1 快速擦写存储器控制寄存器 1 (FLMCR1)	345
17.5.2 快速擦写存储器控制寄存器 2 (FLMCR2)	346
17.5.3 擦除块指定寄存器 1 (EBR1)	346
17.5.4 快速擦写存储器功率控制寄存器 (FLPWCR)	346
17.6 单板编程模式	347

17.6.1	引导模式	347
17.6.2	用户模式的编程 / 擦除	349
17.7	编程 / 擦除程序	350
17.7.1	编程 / 编程验证	350
17.7.2	擦除 / 擦除验证	352
17.7.3	快速擦写存储器的编程 / 擦除时的中断处理	352
17.8	编程 / 擦除保护	354
17.8.1	硬件保护	354
17.8.2	软件保护	354
17.8.3	错误保护	354
17.9	编程器模式	354
17.10	快速擦写存储器的低功耗运行	355
第 18 章	掩模型 ROM	356
18.1	使用时的注意事项	357
18.1.1	F-ZTAT 单片机的掩模型 ROM 化时的注意事项	357
第 19 章	时钟振荡器	358
19.1	寄存器的说明	359
19.1.1	系统时钟控制寄存器 (SCKCR)	359
19.1.2	低功耗控制寄存器 (LPWRCR)	360
19.2	振荡器	361
19.2.1	连接晶体谐振器的方法	361
19.2.2	输入外部时钟的方法	362
19.3	PLL 电路	363
19.4	子时钟分频器	363
19.5	中速时钟分频器	363
19.6	总线主控器时钟选择电路	363
19.7	使用时的注意事项	364
19.7.1	谐振器的注意事项	364
19.7.2	电路板设计上的注意事项	364
第 20 章	低功耗状态	365
20.1	寄存器说明	369
20.1.1	待机控制寄存器 (SBYCR)	369
20.1.2	低功耗控制寄存器 (LPWRCR)	370
20.1.3	模块待机控制寄存器 A ~ D (MSTPCRA ~ MSTPCRD)	371
20.2	中速模式	372
20.3	睡眠模式	373
20.3.1	向睡眠模式转移	373
20.3.2	睡眠模式的解除	373
20.4	软件待机模式	373
20.4.1	向软件待机模式转移	373
20.4.2	软件待机模式的解除	373
20.4.3	软件待机模式解除后振荡器稳定时间的设定	374
20.4.4	软件待机模式应用举例	375
20.5	硬件待机模式	375
20.5.1	向硬件待机模式转移	375
20.5.2	硬件待机模式的解除	375
20.5.3	硬件待机模式的时序	376
20.6	模块待机模式	376
20.7	监视模式	377

20.7.1	向监视模式转移	377
20.7.2	监视模式解除	377
20.8	子睡眠模式	377
20.8.1	向子睡眠模式转移	377
20.8.2	子睡眠模式的解除	377
20.9	子激活模式	378
20.9.1	向子激活模式转移	378
20.9.2	子激活模式的解除	378
20.10	直接转移	378
20.10.1	从高速模式向子激活模式转移	378
20.10.2	从 subactive 模式向高速模式直接转移	378
20.11	ϕ 时钟输出控制	379
20.12	使用注意事项	379
20.12.1	I/O 端口状态	379
20.12.2	振荡稳定待机中的电流消耗	379
20.12.3	内部外围模块中断	379
20.12.4	MSTPCR 的写入	379
第 21 章	寄存器一览表	380
21.1	寄存器地址一览表（按地址顺序）	381
21.2	寄存器位一览表	394
21.3	各工作模式下的寄存器状态	407
第 22 章	电特性	419
22.1	绝对最大额定值	419
22.2	DC 特性	420
22.3	AC 特性	423
22.3.1	时钟时序	423
22.3.2	控制信号时序	424
22.3.3	内部外围模块时序	426
22.4	A/D 转换特性	429
22.5	快速擦写存储器特性	430
22.6	LCD 特性	431
附录		432
A.	在各管脚状态的 I/O 端口状态	432
B.	产品型号名称一览	433
C.	外形尺寸图以《瑞萨半导体器件封装数据手册》中记载的尺寸图为准。	434
索引		435

第 1 章 概要

1.1 特点

- 16 位高速 H8S/2000CPU
在目标代码级，与 H8/300CPU、H8/300HCPU 向上兼容
通用寄存器：16 位 ×16 个
基本指令：65 种
- 丰富的外围功能
16 位定时器脉冲单元 (TPU)
看门狗定时器 (WDT)
异步或时钟同步串行通信接口 (SCI)
控制器局域网 (Controller Area Network) 总线模块 (HCAN) (仅 H8S/2282 群具备)
10 位 A/D 转换器
电机控制脉宽调制定时器 (PWM)
LCD 控制器 / 驱动器 (LCD)
时钟发生器
- 内部存储器

ROM	型号	ROM	RAM
闪存 (F-ZTAT™ 版)	HD64F2282	128K 字节	4k 字节
	HD64F2280B	64K 字节	2K 字节
	HD64F2280RB	64K 字节	2K 字节
掩膜 ROM 版	HD6432282	128K 字节	4K 字节
	HD6432281	64K 字节	4K 字节

- 通用 I/O 端口
- I/O 管脚：64 个
- 输入管脚：8 个
- 支持各种低功耗模式
- 小型封装

封装	代码	尺寸	管脚间距
QFP-100	FP-100A	14.0×20.0mm	0.65mm

H8S/2280 群的内部框图如图 1.2、图 1.3 所示。

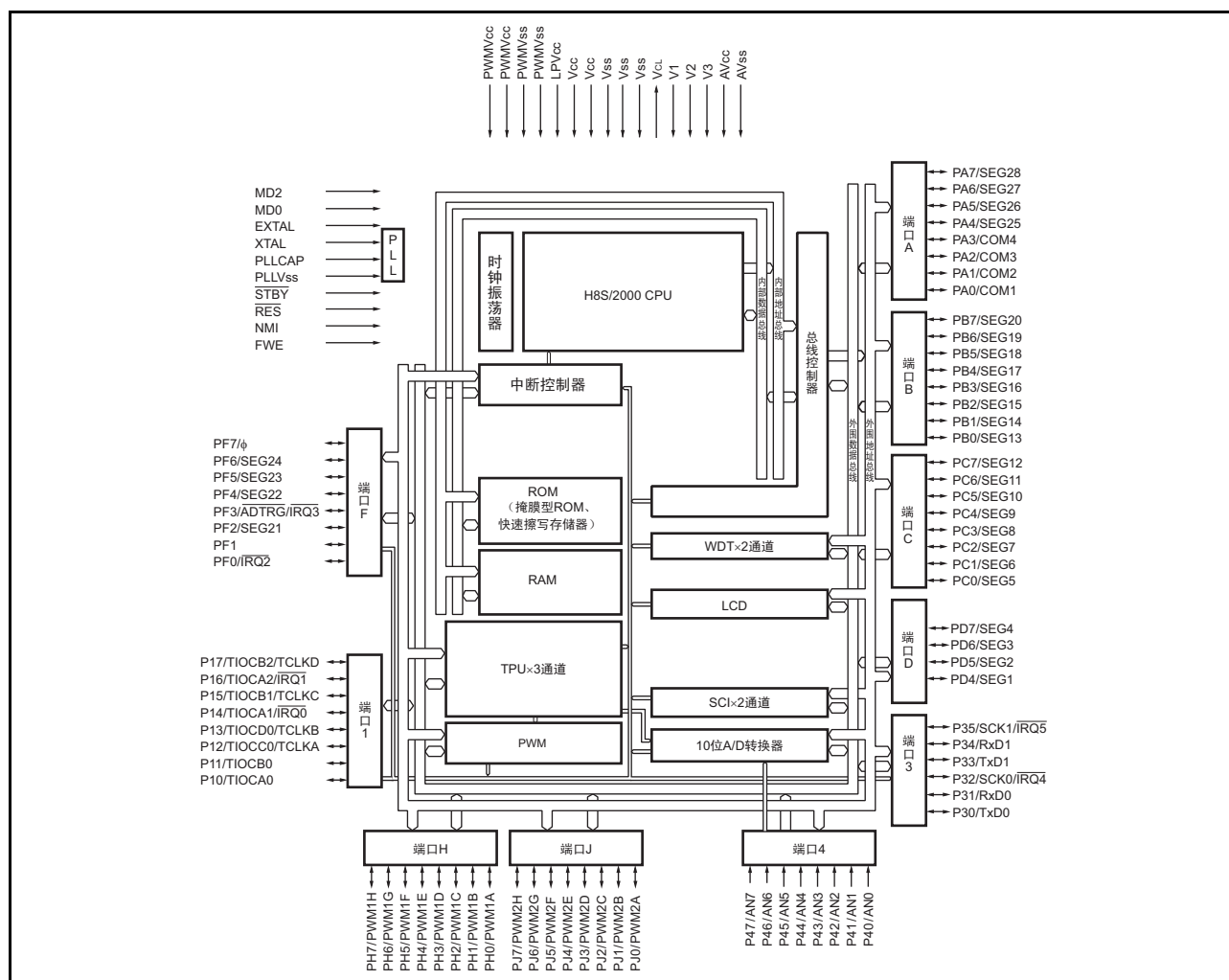


图 1.2 H8S/2280 群 (HD64F2280B) 内部框图

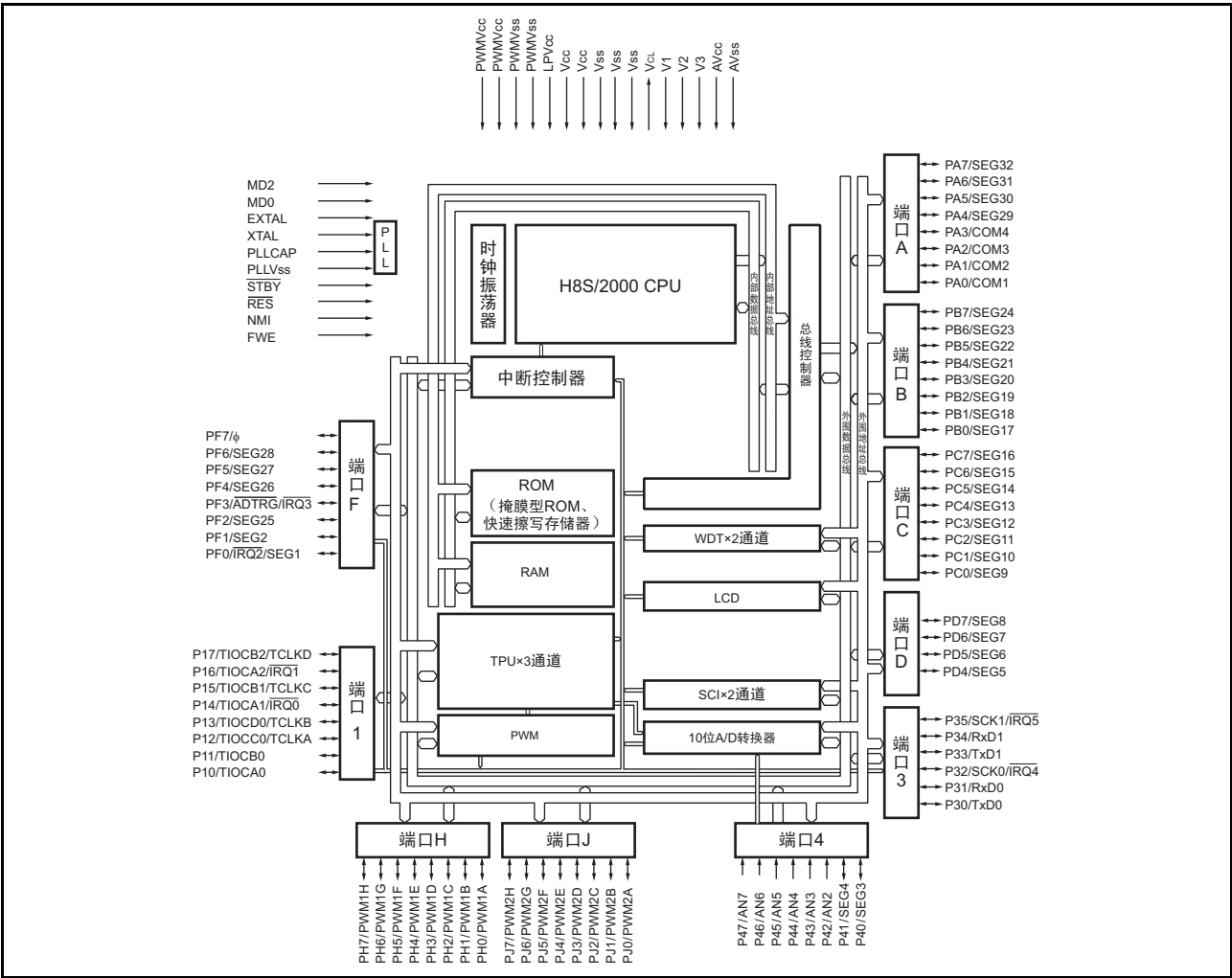


图 1.3 H8S/2280 群 (HD64F2280RB) 内部框图

1.3 管脚排列

H8S/2282 群的管脚排列如图 1.4 所示。

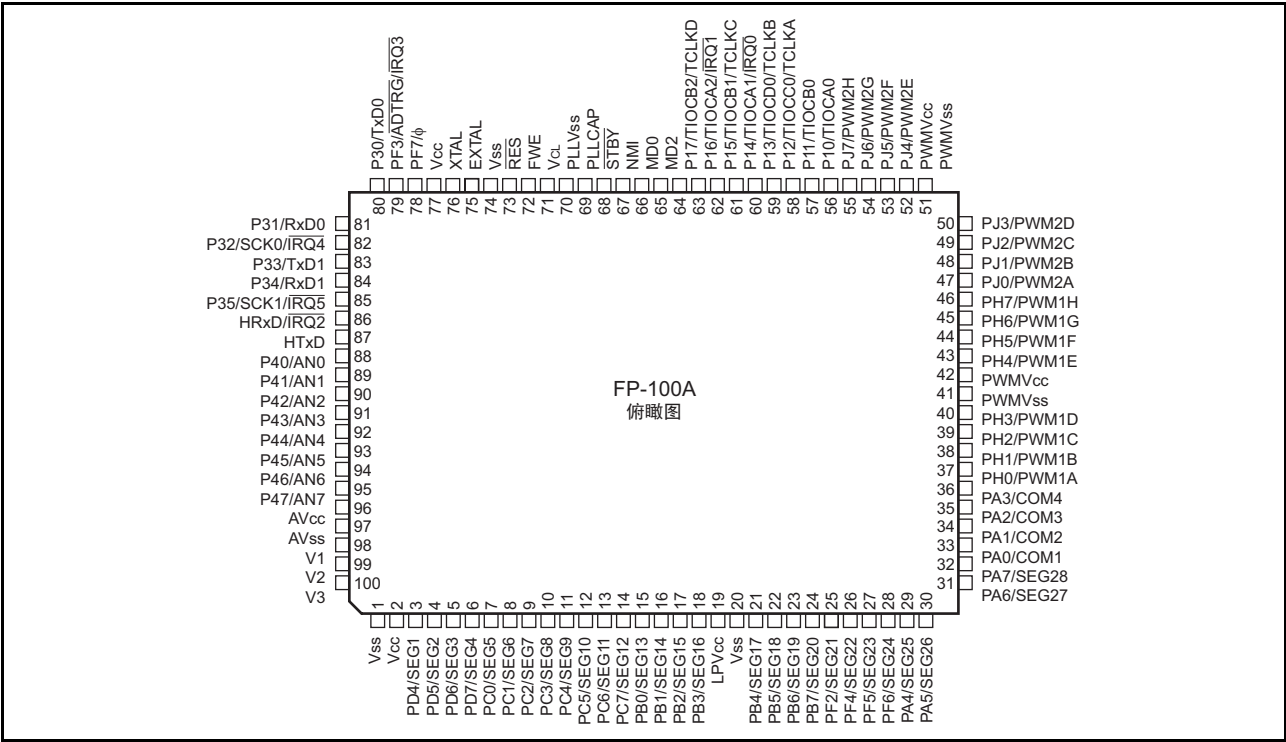


图 1.4 H8S/2282 群管脚排列图

H8S/2280 群管脚的排列如图 1.5、图 1.6 所示。

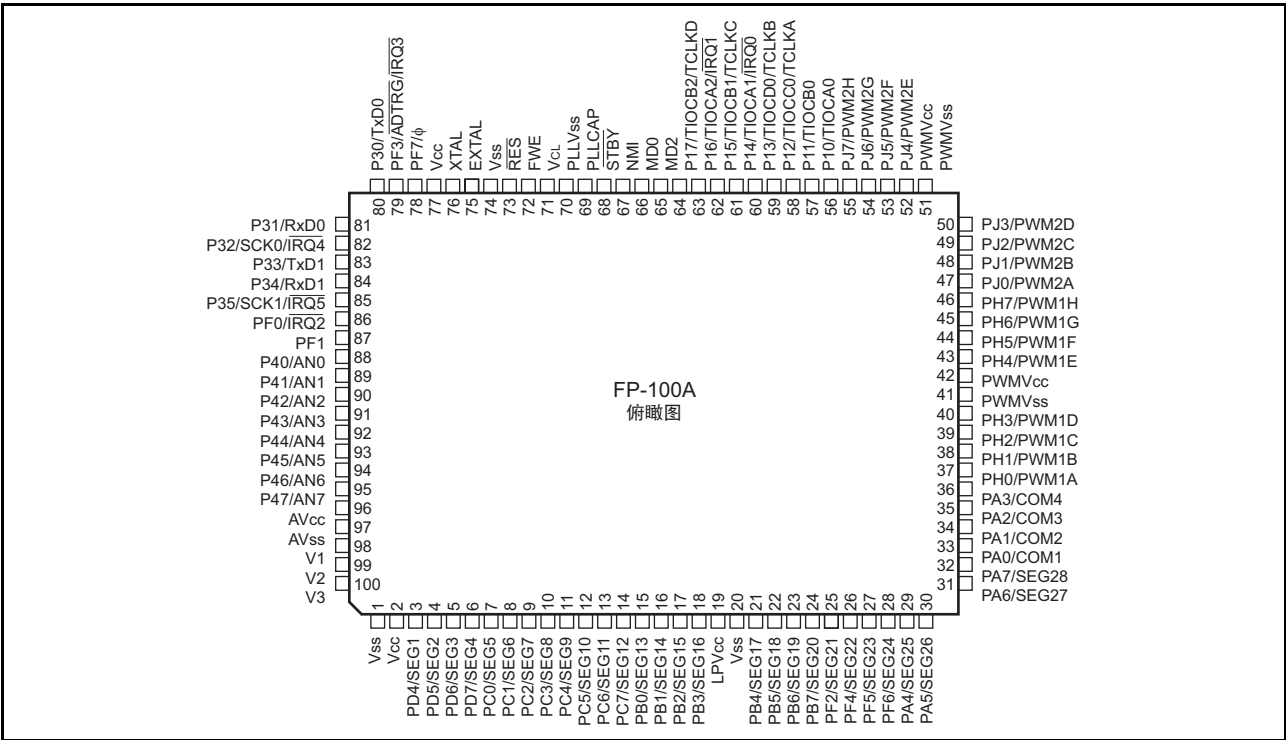


图 1.5 H8S/2280 群（HD64F2280B）管脚排列图

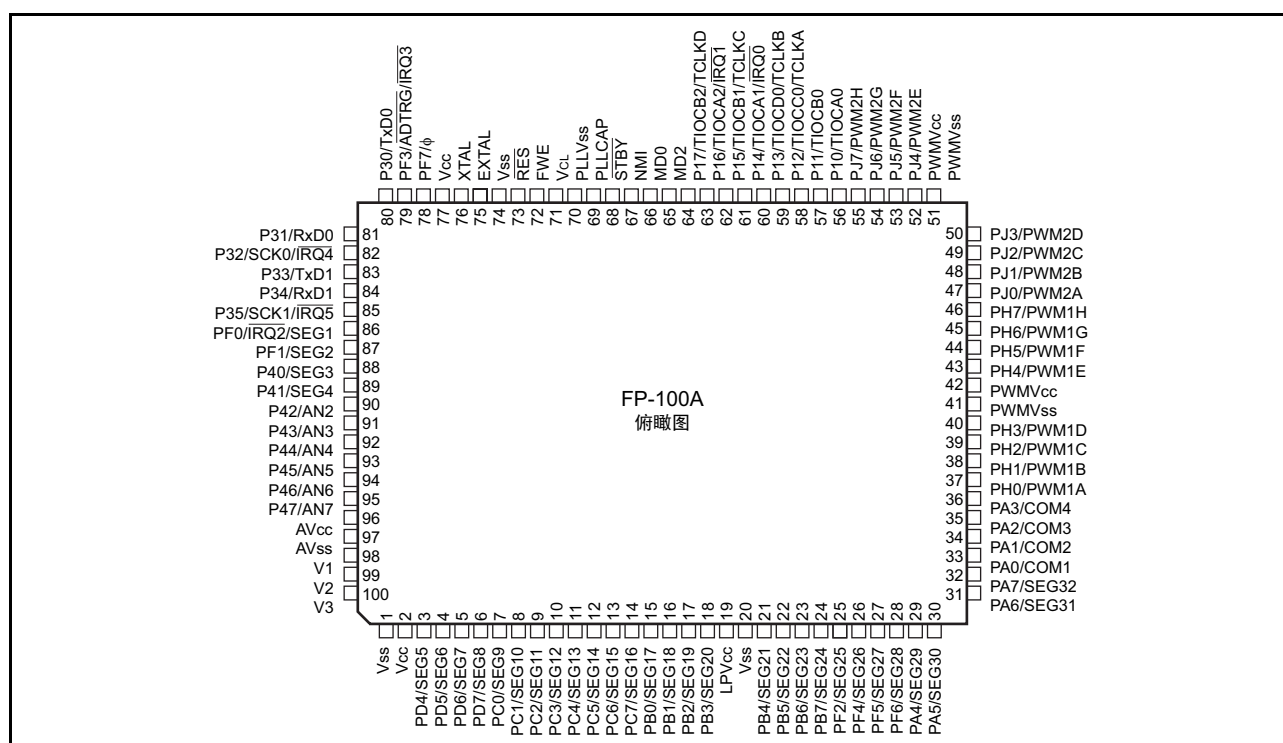


图 1.6 H8S/2280 群 (HD64F2280RB) 管脚排列图

1.4 管脚功能

1.4.1 H8S/2282 群、H8S/2280 群（HD64F2280B）管脚功能

类型	符号	管脚编号	输入 / 输出	功能
电源	Vcc	2 77	输入	电源管脚。请与系统电源连接。
	PWMVcc	42 52	输入	为端口 H、J 及电机控制的 PWM 定时器提供电源。
	LPVcc	19	输入	为端口 A ~ D 及 F(PF2、PF4 ~ PF6) 提供电源。
	V1 V2 V3	98 99 100	输入	为 LCD 控制器 / 驱动器提供电源。内部与分压电阻连接，通常处于开路状态。电源条件为 $LPVcc \geq V1 \geq V2 \geq V3 \geq Vss$ 。
	Vss	1 20 74	输入	接地。与系统电源地线连接 (0V)。
	PWMVss	41 51	输入	为端口 H、J 及电机控制的 PWM 定时器提供电源。连接至系统电源地线 (0V)。
	Vcl	71	输出	内部降压供电电路的外置电容管脚。该管脚和 VSS 管脚之间接入 0.1 μ F 的电容（位于管脚旁）。
时钟	PLLvss	70	输入	内部 PLL 振荡器的接地管脚。
	PLLCAP	69	输出	内置 PLL 振荡器的外接电容管脚。
	XTAL	76	输入	连接晶体谐振器。XTAL 管脚也可输入外部时钟。连接范例，请参考「第 19 章 时钟振荡器」。
	EXTAL	75	输出	连接晶体谐振器。EXTAL 管脚也可输入外部时钟。连接范例，请参考「第 19 章 时钟振荡器」。
	ϕ	78	输出	向外部设备提供系统时钟。
工作模式控制	MD2 MD0	65 66	输入	设定工作模式。运行中不可改变这些管脚的输入。
系统控制	RES	73	输入	复位管脚。此管脚为低电平时，芯片处于复位状态。
	STBY	68	输入	此管脚为低电平时，转换为硬件待机模式。
	FWE	72	输入	闪存使用管脚。仅闪存版本使用。

类型	符号	管脚编号	输入 / 输出	功能
中断	NMI	67	输入	非屏蔽中断管脚。不用时，应为高电平。
	$\overline{\text{IRQ5}}$	85	输入	可屏蔽中断请求输入管脚。
	$\overline{\text{IRQ4}}$	82		
	$\overline{\text{IRQ3}}$	79		
	$\overline{\text{IRQ2}}$	86		
	$\overline{\text{IRQ1}}$	63		
	$\overline{\text{IRQ0}}$	61		
16 位定时器 脉冲单元	TCLKA	59	输入	外部时钟输入管脚。
	TCLKB	60		
	TCLKC	62		
	TCLKD	64		
	TIOCA0	57	输入 / 输出	TGRA_0 ~ TGRD_0 的输入捕捉输入 / 输出比较输出 / PWM 输出的管脚。
	TIOCB0	58		
	TIOCC0	59		
	TIOCD0	60		
串行通信接 口 (SCI)/ 智 能卡接口	TIOCA1	61	输入 / 输出	TGRA_1 到 TGRB_1 的输入捕捉输入 / 输出与输出 / PWM 的输出管脚相比较。
	TIOCB1	62		
	TIOCA2	63	输入 / 输出	TGRA_2 到 TGRB_2 的输入捕捉输入 / 输出比较输出 / PWM 输出的管脚。
	TIOCB2	64		
	TxD1	83	输出	数据输出管脚。
	TxD0	80		
	RxD1	84	输出	数据输入管脚。
	RxD0	81		
	SCK1	85	输入 / 输出	时钟输入 / 输出管脚。
	SCK0	82		
HCAN	HTxD*1	87	输出	CAN 总线发送管脚。
	HRxD*1	86	输入	CAN 总线接收管脚。
A/D 转换器	AN7	95	输入	模拟信号输入管脚
	AN6	94		
	AN5	93		
	AN4	92		
	AN3	91		
	AN2	90		
	AN1	89		
	AN0	88		

类型	符号	管脚编号	输入 / 输出	功能
A/D 转换器	ADTRG	79	输入	外部的输入触发 A/D 转换器。
	AVcc	96	输入	A/D 转换器电源管脚。不用 A/D 转换器时，请连接系统电源 (+5V)。
	AVss	97	输入	A/D 转换器接地管脚。连接系统电源 (0V)。
电机控制 PWM 定时器	PWM1H	46	输出	PWM_1 脉冲输出管脚。
	PWM1G	45		
	PWM1F	44		
	PWM1E	43		
	PWM1D	40		
	PWM1C	39		
	PWM1B	38		
	PWM1A	37		
	PWM2H	56	输出	PWM_2 脉冲输出管脚。
	PWM2G	55		
	PWM2F	54		
	PWM2E	53		
	PWM2D	50		
	PWM2C	49		
	PWM2B	48		
	PWM2A	47		
LCD 控制器 / 驱动器	SEG28	32	输出	LCD 段驱动信号的输出管脚。
	SEG27	31		
	SEG26	30		
	SEG25	29		
	SEG24	28		
	SEG23	27		
	SEG22	26		
	SEG21	25		
	SEG20	24		
	SEG19	23		
	SEG18	22		
	SEG17	21		
	SEG16	18		
	SEG15	17		
	SEG14	16		
	SEG13	15		
	SEG12	14		

类型	符号	管脚编号	输入 / 输出	功能
LCD 控制器 / 驱动器	SEG11	13	输出	LCD 段驱动信号输出管脚。
	SEG10	12		
	SEG9	11		
	SEG8	10		
	SEG7	9		
	SEG6	8		
	SEG5	7		
	SEG4	6		
	SEG3	5		
	SEG2	4		
	SEG1	3		
	COM4	36	输出	LCD 公用驱动信号输出管脚。
	COM3	35		
	COM2	34		
	COM1	33		
I/O 端口	P17	64	输入 / 输出	8 个输入 / 输出管脚。
	P16	63		
	P15	62		
	P14	61		
	P13	60		
	P12	59		
	P11	58		
	P10	57		
	P35	85	输入 / 输出	6 个输入 / 输出管脚。
	P34	84		
	P33	83		
	P32	82		
	P31	81		
	P30	80		
	P47	95	输入	8 个输入管脚。
	P46	94		
	P45	93		
	P44	92		
	P43	91		
	P42	90		
	P41	89		
	P40	88		

类型	符号	管脚编号	输入 / 输出	功能
I/O 端口	PA7	32	输入 / 输出	8 个输入 / 输出管脚。
	PA6	31		
	PA5	30		
	PA4	29		
	PA3	36		
	PA2	35		
	PA1	34		
	PA0	33		
	PB7	24	输入 / 输出	8 个输入 / 输出管脚。
	PB6	23		
	PB5	22		
	PB4	21		
	PB3	18		
	PB2	17		
	PB1	16		
	PB0	15		
	PC7	14	输入 / 输出	8 个输入 / 输出管脚。
	PC6	13		
	PC5	12		
	PC4	11		
	PC3	10		
	PC2	9		
	PC1	8		
	PC0	7		
	PD7	6	输入 / 输出	4 个输入 / 输出管脚
	PD6	5		
	PD5	4		
	PD4	3		
	PF7	78	输入 / 输出	8 个输入 / 输出管脚。
	PF6	28		
	PF5	27		
	PF4	26		
	PF3	79		
	PF2	25		
	PF1*2	87		
	PF0*2	86		

类型	符号	管脚编号	输入 / 输出	功能
I/O 端口	PH7	46	输入 / 输出	8 个输入 / 输出管脚。
	PH6	45		
	PH5	44		
	PH4	43		
	PH3	40		
	PH2	39		
	PH1	38		
	PH0	37		
	PJ7	56	输入 / 输出	8 个输入 / 输出管脚。
	PJ6	55		
	PJ5	54		
	PJ4	53		
	PJ3	50		
	PJ2	49		
	PJ1	48		
	PJ0	47		

【注】 *1H8S/2280 系列没有 HCAN 管脚。

*2H8S/2282 系列没有 PF1、PF0 管脚。

1.4.2 H8S/2280 群（HD64F2280RB）的管脚功能

类型	符号	管脚编号	输入 / 输出	功能
电源	Vcc	2 77	输入	电源管脚。与系统电源连接。
	PWMVcc	42 52	输入	端口 H、J 及电机控制的 PWM 定时器的电源管脚。
	LPVcc	19	输入	端口 A ~ D 及 F(PF2、PF4 ~ PF6) 的电源管脚。
	V1 V2 V3	98 99 100	输入	LCD 控制器 / 驱动器电源管脚。内部与分压电阻连接，通常处于开路状态。电源条件为 $LPVcc \geq V1 \geq V2 \geq V3 \geq Vss$ 。
	Vss	1 20 74	输入	接地管脚。与系统电源地线连接 (0V)。
	PWMVss	41 51	输入	端口 H、J 及电机控制的 PWM 定时器电源管脚。与系统电源地线 (0V) 连接。
	Vcl	71	输出	内部降压供电电路的外置电容管脚。请在该管脚和 VSS 管脚之间接入 0.1μF 的电容（位于管脚旁）。
时钟	PLLvss	70	输入	内部 PLL 振荡器的接地管脚。
	PLLCAP	69	输出	内部 PLL 振荡器的外接电容管脚。
	XTAL	76	输入	连接晶体谐振器。XTAL 管脚可提供外部时钟。外部时钟输入和晶体谐振器的连接范例，请参考「第 19 章 时钟振荡器」。
	EXTAL	75	输出	连接晶体谐振器。EXTAL 管脚可提供外部时钟。外部时钟输入和晶体谐振器的连接范例，请参考「第 19 章 时钟振荡器」。
	φ	78	输出	向外部设备提供系统时钟。
工作模式控制	MD2 MD0	65 66	输入	设定工作模式。运行期间，不可改变这些管脚的输入。
系统控制	RES	73	输入	复位管脚。若此管脚为低电平，则处于复位状态。
	STBY	68	输入	若此管脚为低电平，则转为硬件待机模式。
	FWE	72	输入	闪存专用闪存管脚。仅闪存版本使用。

类型	符号	管脚编号	输入 / 输出	功能
中断	NMI	67	输入	非屏蔽中断请求输入管脚。不用时，应为高电平。
	$\overline{\text{IRQ5}}$	85	输入	可屏蔽中断请求输入管脚。
	$\overline{\text{IRQ4}}$	82		
	$\overline{\text{IRQ3}}$	79		
	$\overline{\text{IRQ2}}$	86		
	$\overline{\text{IRQ1}}$	63		
	$\overline{\text{IRQ0}}$	61		
16 位定时器 脉冲单元	TCLKA	59	输入	外部时钟输入管脚。
	TCLKB	60		
	TCLKC	62		
	TCLKD	64		
	TIOCA0	57	输入 / 输出	TGRA_0 ~ TGRD_0 的输入捕捉输入 / 输出比较输出 / PWM 输出的管脚。
	TIOCB0	58		
	TIOCC0	59		
	TIOCD0	60		
串行通信接 口 (SCI)/ 智 能卡接口	TIOCA1	61	输入 / 输出	TGRA_1、TGRB_1 的输入捕捉输入 / 输出比较输出 / PWM 输出的管脚。
	TIOCB1	62		
	TIOCA2	63	输入 / 输出	TGRA_2、TGRB_2 输入捕捉输入 / 输出比较输出 / PWM 输出的管脚。
	TIOCB2	64		
	TxD1	83	输出	数据输出管脚。
	TxD0	80		
A/D 转换器	RxD1	84	输入	数据输入管脚。
	RxD0	81		
	SCK1	85	输入 / 输出	时钟输入 / 输出管脚。
	SCK0	82		
	AN7	95	输入	模拟信号输入管脚
	AN6	94		
	AN5	93		
	AN4	92		
	AN3	91		
	AN2	90		
	$\overline{\text{ADTRG}}$	79	输入	外部触发器输入管脚。可触发 A/D 转换器。
	AVcc	96	输入	A/D 转换器电源管脚。不用 A/D 转换器时，该管脚与系统电源 (+5V) 连接。
	AVss	97	输入	A/D 转换器接地管脚。与系统电源地线连接 (0V)。

类型	符号	管脚编号	输入 / 输出	功能
电机控制 PWM 定时器	PWM1H	46	输出	PWM_1 脉冲输出管脚。
	PWM1G	45		
	PWM1F	44		
	PWM1E	43		
	PWM1D	40		
	PWM1C	39		
	PWM1B	38		
	PWM1A	37		
	PWM2H	56	输出	PWM_2 脉冲输出管脚。
	PWM2G	55		
	PWM2F	54		
	PWM2E	53		
	PWM2D	50		
	PWM2C	49		
	PWM2B	48		
	PWM2A	47		
LCD 控制器 / 驱动器	SEG32	32	输出	LCD 段驱动信号的输出管脚。
	SEG31	31		
	SEG30	30		
	SEG29	29		
	SEG28	28		
	SEG27	27		
	SEG26	26		
	SEG25	25		
	SEG24	24		
	SEG23	23		
	SEG22	22		
	SEG21	21		
	SEG20	18		
	SEG19	17		
	SEG18	16		
	SEG17	15		
	SEG16	14		
	SEG15	13		
	SEG14	12		
	SEG13	11		
	SEG12	10		

类型	符号	管脚编号	输入 / 输出	功能
LCD 控制器 / 驱动器	SEG11	9	输出	LCD 段驱动信号输出管脚。
	SEG10	8		
	SEG9	7		
	SEG8	6		
	SEG7	5		
	SEG6	4		
	SEG5	3		
	SEG4	89		
	SEG3	88		
	SEG2	87		
	SEG1	86		
	COM4	36	输出	LCD 公共信号输出管脚。
	COM3	35		
	COM2	34		
	COM1	33		
I/O 端口	P17	64	输入 / 输出	8 位输入 / 输出管脚。
	P16	63		
	P15	62		
	P14	61		
	P13	60		
	P12	59		
	P11	58		
	P10	57		
	P35	85	输入 / 输出	6 位输入 / 输出管脚。
	P34	84		
	P33	83		
	P32	82		
	P31	81		
	P30	80		
	P47	95	输入	8 位输入管脚。
	P46	94		
	P45	93		
	P44	92		
	P43	91		
	P42	90		
	P41	89		
	P40	88		

类型	符号	管脚编号	输入 / 输出	功能
I/O 端口	PA7	32	输入 / 输出	8 位输入 / 输出管脚。
	PA6	31		
	PA5	30		
	PA4	29		
	PA3	36		
	PA2	35		
	PA1	34		
	PA0	33		
	PB7	24	输入 / 输出	8 位输入 / 输出管脚。
	PB6	23		
	PB5	22		
	PB4	21		
	PB3	18		
	PB2	17		
	PB1	16		
	PB0	15		
	PC7	14	输入 / 输出	8 位输入 / 输出管脚。
	PC6	13		
	PC5	12		
	PC4	11		
	PC3	10		
	PC2	9		
	PC1	8		
	PC0	7		
	PD7	6	输入 / 输出	4 位输入 / 输出管脚。
	PD6	5		
	PD5	4		
	PD4	3		
	PF7	78	输入 / 输出	8 位输入 / 输出管脚。
	PF6	28		
	PF5	27		
	PF4	26		
	PF3	79		
	PF2	25		
	PF1	87		
	PF0	86		

类型	符号	管脚编号	输入 / 输出	功能
I/O 端口	PH7	46	输入 / 输出	8 位输入 / 输出管脚。
	PH6	45		
	PH5	44		
	PH4	43		
	PH3	40		
	PH2	39		
	PH1	38		
	PH0	37		
	PJ7	56	输入 / 输出	8 位输入 / 输出管脚。
	PJ6	55		
	PJ5	54		
	PJ4	53		
	PJ3	50		
	PJ2	49		
	PJ1	48		
	PJ0	47		

第 2 章 CPU

H8S/2000 CPU 是与 H8/300 CPU 及 H8/300H CPU 向上兼容，并具有 32 位体系结构的高速 CPU。H8S/2000CPU 有 16 个 16 位通用寄存器，可寻址 16M 字节的线性地址空间，十分适合于实时控制。本章描述 H8S/2000CPU。由于产品的型号不同其使用模式及地址空间也有所不同，详情请参考「第 3 章 MCU 工作模式」。

2.1 特点

- 与 H8/300 CPU 及 H8/300H CPU 向上兼容
可执行 H8/300 CPU 和 H8/300H CPU 的目标程序
- 通用寄存器：16 位 × 16 个
也可作为 8 位 × 16 个、32 位 × 8 个通用寄存器使用
- 基本命令：65 种
8 / 16 / 32 位运算指令
乘除运算指令
强大的位操作指令
- 寻址方式：8 种
寄存器直接寻址 (Rn)
寄存器间接寻址 (@ERn)
带移位量寄存器间接寻址 (@(d:16,ERn) / @(d:32,ERn))
后增 / 先减寄存器间接寻址 (@ERn+ / @-ERn)
绝对地址寻址 (@aa:8 / @aa:16 / @aa:24 / @aa:32)
立即数寻址 (#xx:8 / #xx:16 / #xx:32)
程序计数器相对寻址 (@(d:8,PC) / @(d:16,PC))
存储器间接寻址 (@@aa:8)
- 地址空间：16M 字节
程序：16M 字节
数据：16M 字节
- 高速运算
所有常用指令均以 1 ~ 2 个机器周期执行
8 / 16 / 32 位寄存器间加减运算：1 个态
8×8 位寄存器间乘法运算：12 个态
16÷8 位寄存器间除法运算：12 个态
16×16 位寄存器间乘法运算：20 个态
32÷16 位寄存器间除法运算：20 个态
- CPU 运行模式：2 种
普通模式 / 高级模式

【注】 在本 LSI 中不可使用普通模式。

- 低功耗状态
通过 SLEEP 指令可转换到低功耗状态
可选择 CPU 时钟速率

2.1.1 H8S/2000 CPU 与 H8S/2600 CPU 的区别

H8S/2000 CPU 与 H8S/2600 CPU 的区别如下。

- 寄存器结构
仅 H8S/2600 CPU 支持 MAC 寄存器。
- 基本命令
仅 H8S/2600 支持 MAC、CLRMAC、LDMAC、STM4C 项指令。
- MULXU、MULXS 指令执行的状态数不同。

指令	助记符	执行状态	
		H8S/2600	H8S/2000
MULXU	MULXU.B Rs, Rd	3	12
	MULXU.W Rs, ERd	4	20
MULXS	MULXS.B Rs, Rd	4	13
	MULXS.W Rs, ERd	5	21

此外，由于产品型号的不同，地址空间、CCR、EXR 寄存器的功能、低功耗状态也有所不同。

2.1.2 H8S/2000 CPU 与 H8/300 CPU 的区别

H8S/2000 CPU 相对于 H8/300 CPU 增加了以下功能。

- 扩充通用寄存器、控制寄存器
增加了 16 位 ×8 个扩展寄存器、8 位 ×1 个及 32 位 ×2 个控制寄存器。
- 扩充地址空间
普通模式可支持与 H8/300CPU 相同的 64K 字节的地址空间。
高级模式可支持最大 16M 字节地址空间。
- 增强寻址能力
能够有效使用 16M 字节的地址空间。
- 指令强化
增强了位操作指令的寻址模式
增加了带符号乘除法运算指令
增加 2 位移位指令
增加多个保存 / 恢复指令的寄存器
增加 test and set 指令
- 高速化
基本指令的执行速度提高 2 倍

2.1.3 H8S/2000 CPU 与 H8/300H CPU 的区别

H8S/2000 CPU 相对于 H8/300H CPU 增加了以下功能。

- 扩充控制寄存器
增加了 8 位 ×1 个及 32 位 ×2 个控制寄存器
- 指令强化
强化位操作指令的寻址模式
增加 2 位移位指令
增加多个保存 / 恢复指令的寄存器
增加 test and set 指令
- 高速化
基本指令的执行速度提高 2 倍

2.2 CPU 运行模式

H8S/2000 CPU 有普通和高级两种运行模式。普通模式支持最大 64K 字节的地址空间，高级模式支持 16M 字节的地址空间。由模式管脚选择 CPU 运行模式。

2.2.1 普通模式

普通模式下异常处理向量和堆栈构造与 H8/300CPU 相同。

- 地址空间
可存取最大 64K 字节的线性地址空间。
- 扩展寄存器 (En)
扩展寄存器 (E0 ~ E7) 可当作 16 位寄存器或 32 位寄存器中的高 16 位寄存器使用。
当扩展寄存器 En 作为 16 位寄存器使用时，可以存放任意值，即使在相关的通用寄存器 (Rn) 作为地址寄存器使用。(请注意，如果通用寄存器在先减寄存器间接寻址 (@-ERn)、后增寄存器间接寻址 (@ERn+) 中使用，并且在发生进位 / 借位，就会对相应的当扩展寄存器产生影响。)
- 指令系统
可使用所有指令及寻址模式。仅有效地址 (EA) 低 16 位有效。
- 异常处理向量表及内存间接转移地址 (Memory Indirect Branch Addresses)
普通模式下，从 H'0000 开始在起始区域分配异常处理向量表，可保存 16 位的转移地址。普通模式的异常处理向量表结构如图 2.1 所示。异常处理向量表请参照「第 4 章 异常处理」。
内存间接寻址 (@@aa:8) 可根据 JMP 及 JSR 指令调用。可通过指令码包含的 8 位绝对地址来指定存储器的操作数，其将成为转移地址。
普通模式的操作数为 16 位 (字段)，则此 16 位为转移地址。
此外，能够存放转移地址的区域为 H'0000 ~ H'00FF，此起始区域也存放异常处理向量表，敬请注意。
- 堆栈构造
普通模式的子程序调用 PC 堆栈结构和异常处理时的 PC 及 CCR、EXR 堆栈结构，如图 2.2 所示。
在中断控制模式 0 中，EXR 不被保存到堆栈。中断控制模式的详细内容请参考「第 4 章 异常处理」。

【注】 本 LSI 中不可使用普通模式。

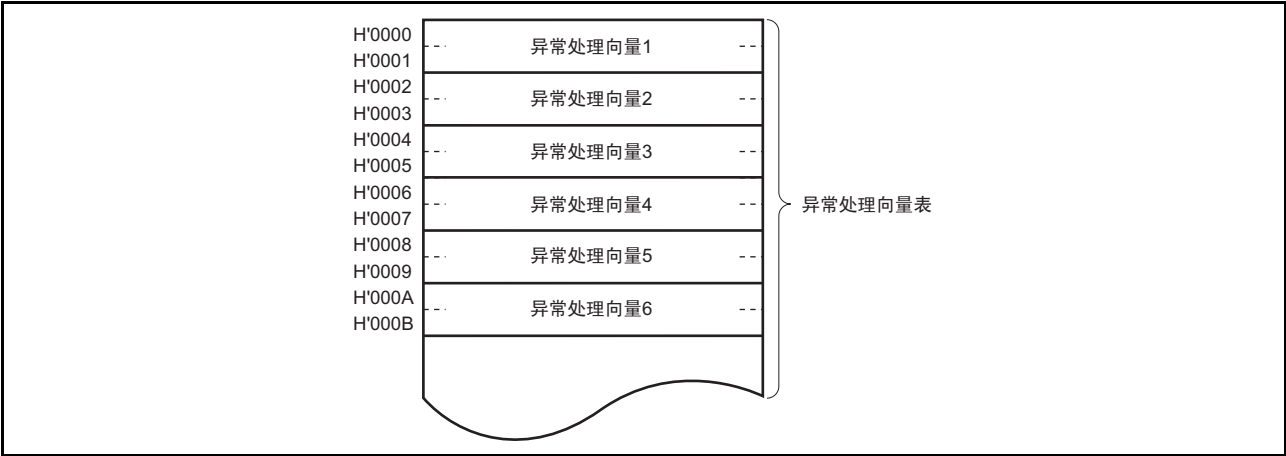


图 2.1 异常处理向量表（普通模式）

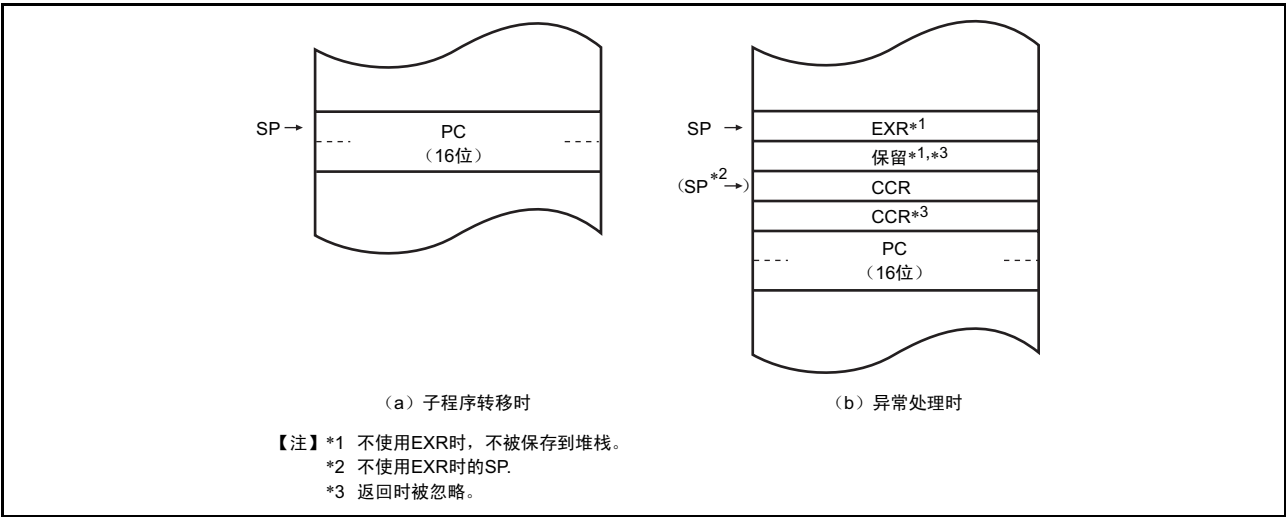


图 2.2 正常模式下的堆栈结构

2.2.2 高级模式

- 地址空间
可存取最大 16M 字节的线性地址空间。
- 扩展寄存器（En）
扩展寄存器（E0～E7）可作为 16 位寄存器、32 位寄存器或地址寄存器的高位 16 位使用。
- 指令系统
可用所有指令及寻址模式。
- 异常处理向量表、内存间接偏移地址
高级模式的环境下，从 H'00000000 开始的起始区域以 32 位为单位分配了异常向量表域，在此忽略高 8 位，保留低 24 位的偏移地址（参考图 2.3）。有关异常处理向量表请参照「第 4 章 异常处理」。

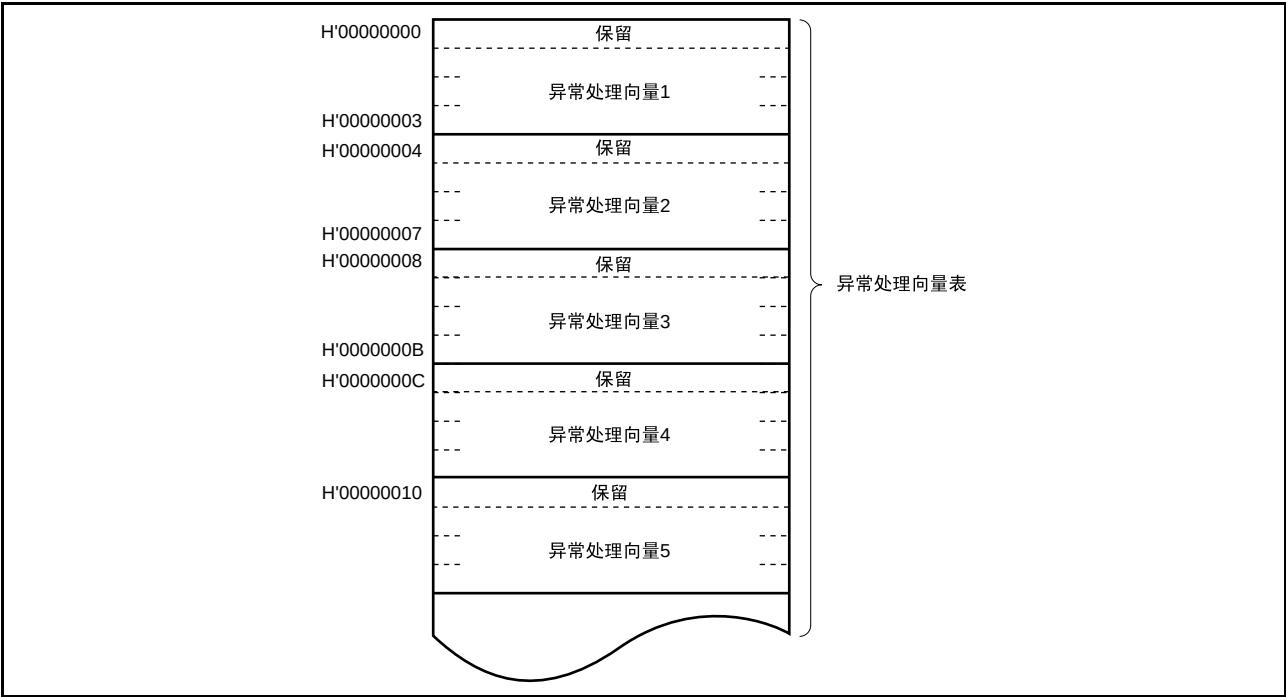


图 2.3 异常处理向量表（高级模式）

内存间接寻址（@@aa:8）可根据 JMP 及 JSR 指令调用。通过指令码包含的 8 位绝对地址指定内存操作数，此即为偏移地址。

高级模式的操作数为 32 位（长字段），则此 32 位为偏移地址，其中高 8 位作为预留区域 H'00。

此外，可存放偏移地址的范围为 H'00000000～H'000000FF，此起始区域也存放异常处理向量表。敬请注意。

• 堆栈构造

在高级模式的环境下，子程序偏移调用时的 PC 堆栈结构及异常处理时的 PC 和 CCR、EXR 堆栈结构如图 2.4 所示。EXR 在中断控制模式 0 不会保存到堆栈。有关中断控制模式请参考「第 4 章 异常处理」。

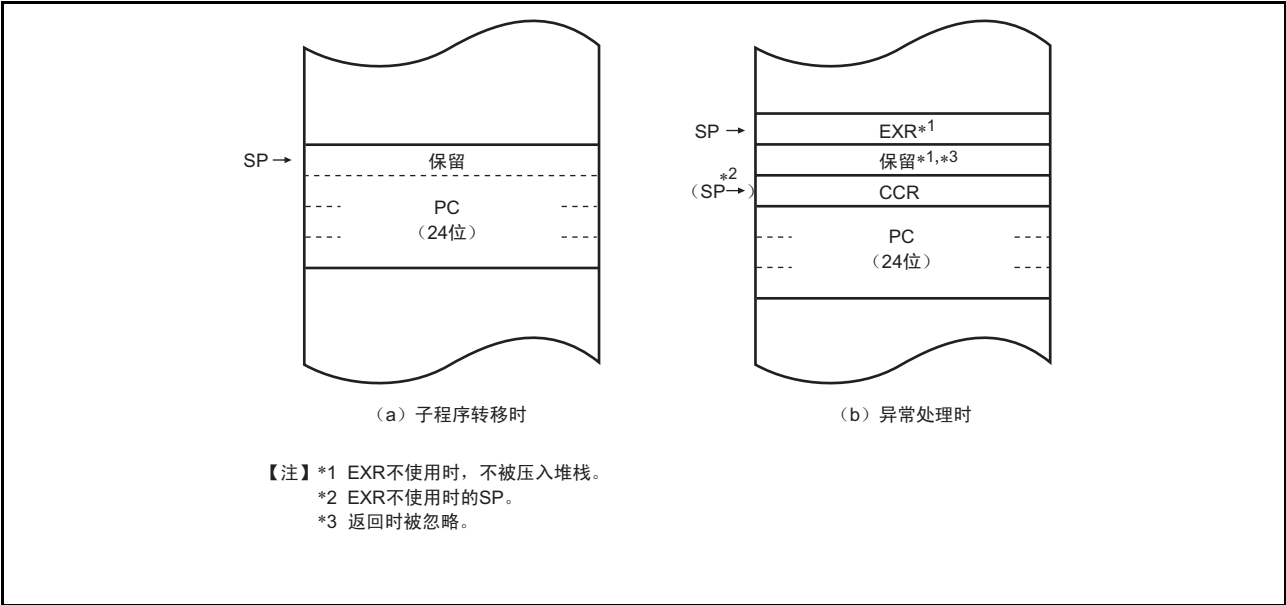


图 2.4 高级模式下的堆栈结构

2.3 地址空间

H8S/2000 CPU 的存储器映像图 2.5 所示。H8S/2000 CPU 在普通模式下能够提供最大 64K 字节的线性地址空间，高级模式下最大为 16M 字节（具有 4G 字节结构）。由于产品的不同，实际能够使用的模式和地址空间也不相同。详情请参考「第 3 章 MCU 工作模式」。

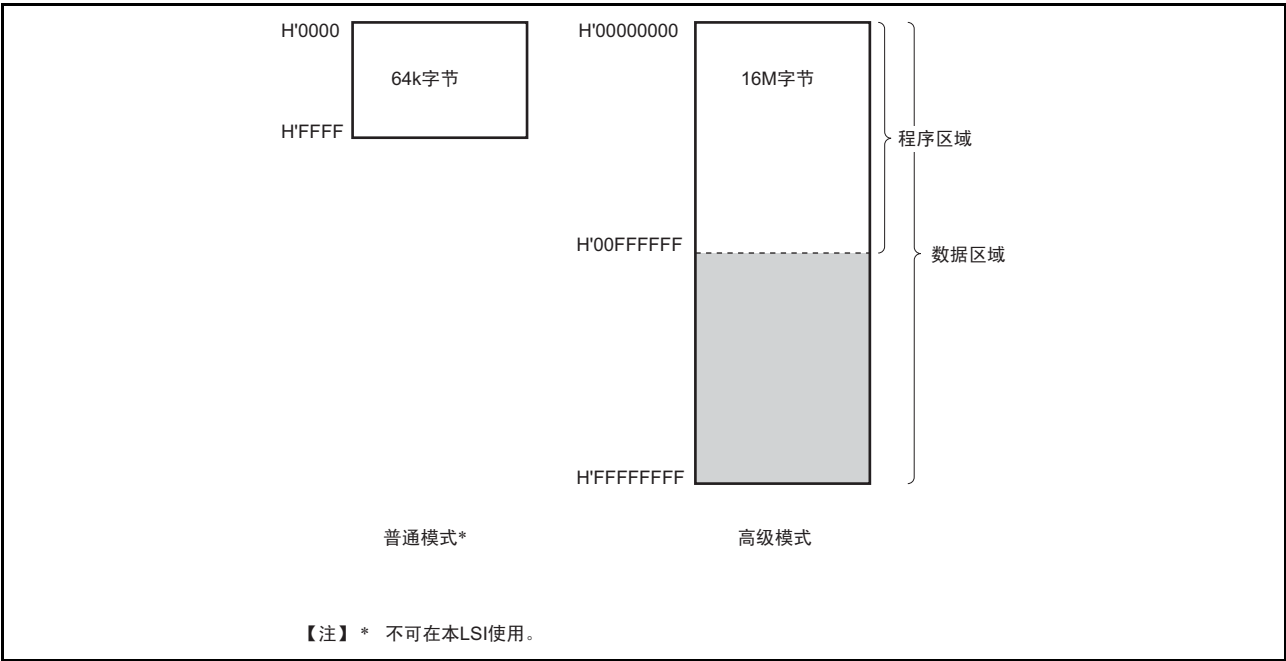


图 2.5 地址空间

2.4 寄存器构成

H8S/2000 CPU 的内部寄存器结构图如图 2.6 所示。这些寄存器分为通用寄存器和控制寄存器。控制寄存器包含了 24 位程序计数器（PC）、8 位扩展控制寄存器（EXR）、8 位条件码寄存器（CCR）。

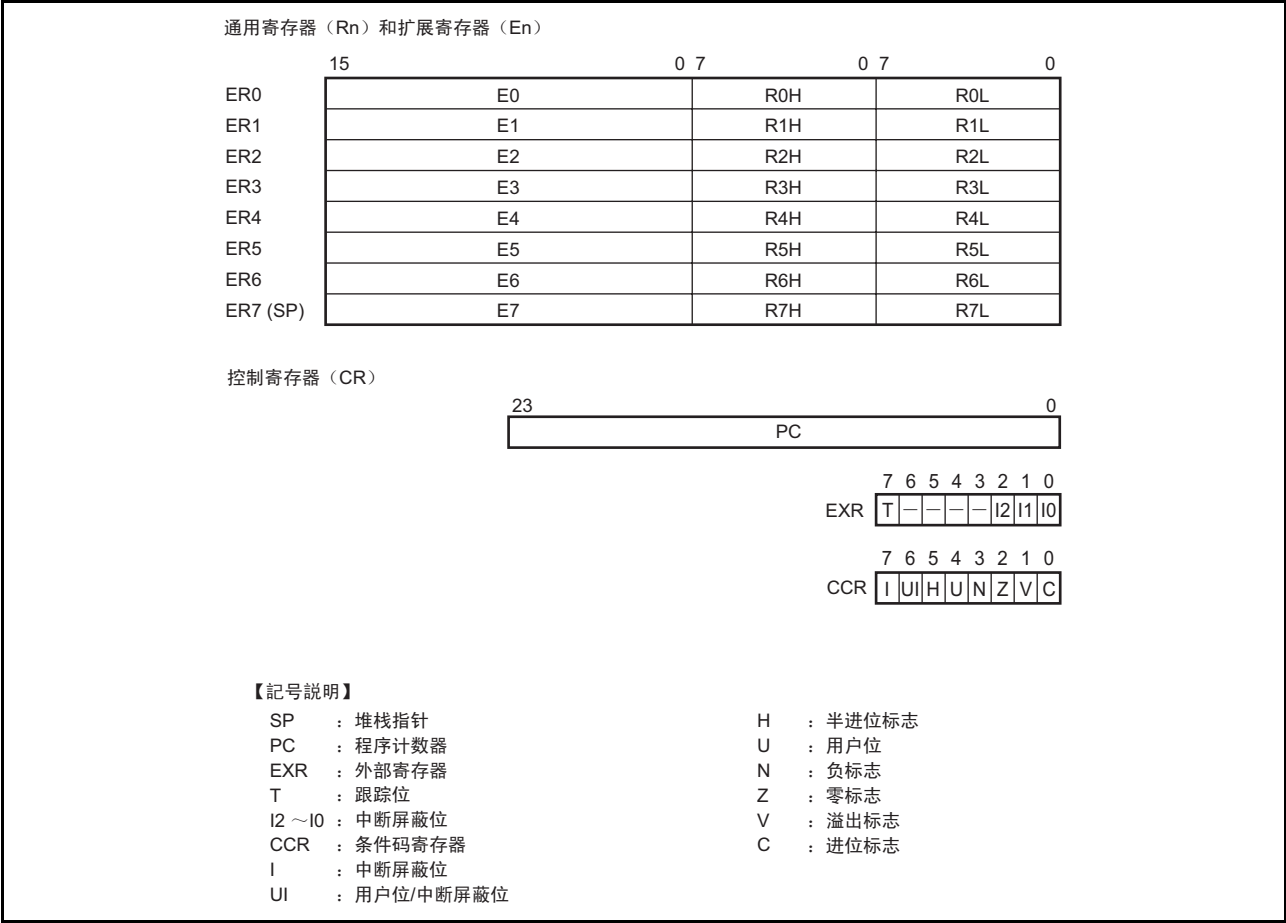


图 2.6 CPU 内部寄存器结构

2.4.1 通用寄存器

H8S/2000CPU 有 8 个 32 位通用寄存器。它们具有相同的功能，可作为地址寄存器或数据寄存器。其数据寄存器可作为 32 位、16 位或 8 位寄存器。通用寄存器的使用方法如图 2.7 所示。

作为地址寄存器或 32 位寄存器使用时，均指定为通用寄存器 ER（ER0～ER7）。

作为 16 位寄存器使用时，通用寄存器 ER 分为通用寄存器 E（E0～E7）、通用寄存器 R（R0～R7）。它们具有相同功能，并可使用最多 16 个 16 位寄存器。有时通用寄存器 E（E0～E7）被归类为扩展寄存器。

作为 8 位寄存器使用时，通用寄存器 R 分为通用寄存器 RH（R0H～R7H）和通用寄存器 RL（R0L～R7L）两组，它们具有相同的功能，并可使用最多 16 个 8 位寄存器。可独立制定各个寄存器的使用方法。

通用寄存器 ER7 除了通用寄存器的功能外，还有堆栈指针（SP）的功能，被隐含地用在异常处理及子程序调用等处。堆栈如图 2.8 所示。

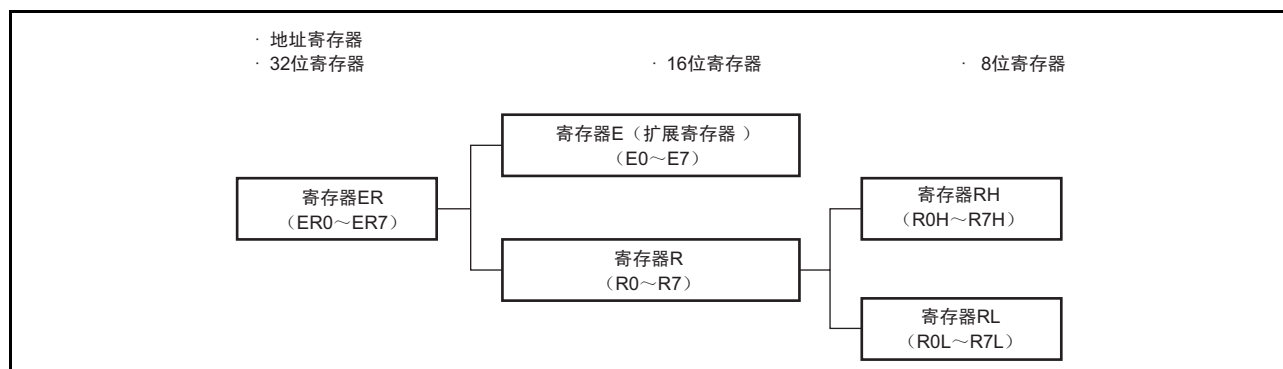


图 2.7 通用寄存器的使用方法

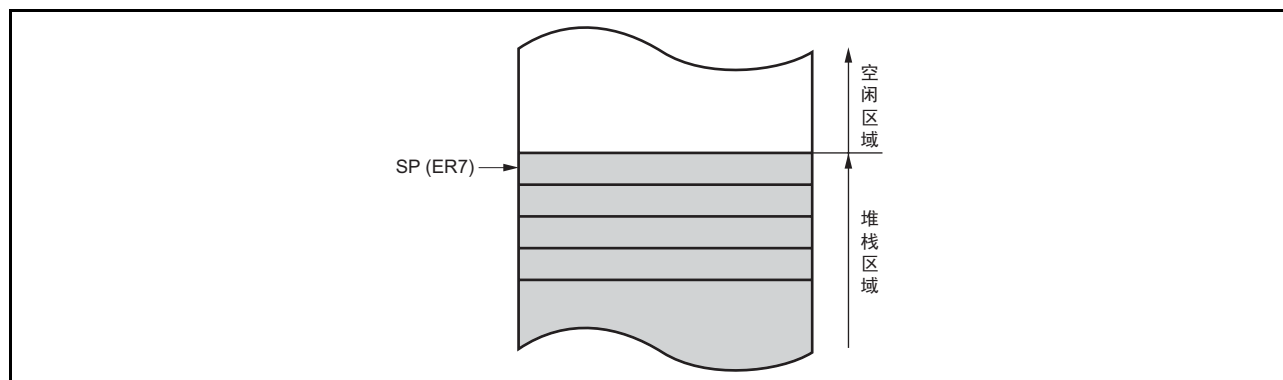


图 2.8 堆栈构造

2.4.2 程序计数器（PC）

PC 是一个 24 位计数器，指定 CPU 将要执行的下一条指令地址。因为 CPU 的指令都是从偶数地址开始，并以 2 字节（字）为单位，所以忽略 PC 的最低位。（在读指令码时，PC 的最低位当作 0）。

2.4.3 扩展寄存器（EXR）

EXR 是一个执行 LDC，STC，ANDC，ORC，XORC 指令的 8 位寄存器。除 STC 之外的指令执行时，指令完成后的三个状态时间里，包括 NMI 在内的所有中断将被屏蔽。

位	位名	初始值	R/W	说明
7	T	0	R/W	跟踪位 将此位置 1，每执行一条指令，就会产生跟踪异常。该位清零时，按顺序执行指令。
6~3	—	1	—	保留位 读取时通常为 1。
2~0	I2 I1 I0	1 1 1	R/W R/W R/W	指定中断请求屏蔽级别（0 ~ 7）。详细信息请参照「第 5 章 中断控制器」。

2.4.4 条件码寄存器（CCR）

CCR 是一个 8 位寄存器，表示 CPU 的内部状态。包含中断屏蔽位（I）、半进位（H）、符号位（N）、零位（Z）、溢出位（V）、进位（C）等 8 位结构的标志位。通过 LDC、STC、ANDC、ORC、XORC 等指令可操作 CCR 位。而 N、Z、V、C 等标志位可根据条件偏移指令调用。

位	符号	初始值	R/W	说明
7	I	1	R/W	中断屏蔽位 若本位置 1，则屏蔽中断请求。但是，NMI 不受 I 位设置影响，总被接受。执行异常处理时，I 位置 1。详情请参考「第 5 章 中断控制器」。
6	UI	不定	R/W	用户位 / 中断屏蔽位 可通过软件（使用 LDC、STC、ANDC、ORC、XORC 指令）进行读写。本 LSI 中，不可作为中断屏蔽位使用。
5	H	不定	R/W	半进位标志 执行 ADD.B、ADDX.B、SUB.B、SUBX.B、CMP.B、NEG.B 指令时，第 3 位有进位或借位时置 1，否则清零。执行 ADD.W、SUB.W、CMP.W、NEG.W 指令时，若第 11 位有进位或借位，或当执行 ADD.L、SUB.L、CMP.L、NEG.L 指令时，若第 27 位有进位或借位，设定为 1，否则清零。
4	U	不定	R/W	用户位 能够通过软件（使用 LDC、STC、ANDC、ORC、XORC 指令）进行读写。
3	N	不定	R/W	符号位标志 存放数据最高位的值作为符号位。
2	Z	不定	R/W	零标志 数据为零时置 1，否则为 0。
1	V	不定	R/W	溢出标志 执行算术运算指令，发生溢出时置 1，否则为 0。
0	C	不定	R/W	进位标志 执行运算有进位发生时置 1，否则为 0。进位有以下种类： • 加法运算的进位 • 减法运算的借位 • 移位 / 循环的进位 另外，进位标志也用作累加器，可以通过位操作指令使用。

2.4.5 CPU 内部寄存器初始值

在 CPU 内部寄存器中，PC 可通过复位进行异常处理，从向量地址读取起始地址来执行初始化。同时 EXR 的 T 位清零、EXR、CCR 的 I 位置为 1，但通用寄存器和 CCR 的其它位不初始化。SP（ER7）的初始值也不定。因此在复位后，请调用 MOV.L 指令执行 SP 的初始化。

2.5 数据格式

H8S/2000 CPU 可处理 1 位、4 位 BCD、8 位（字节）、16 位（字）、及 32 位（长字）的数据。通过存取字节操作数的第 n 位，（格式 n=0、1、2、…7），位操作指令操作 1 位数。此外，DAA 及 DAS 的 10 进制校正指令的字节数据为 2 个 4 位 BCD 数据。

2.5.1 通用寄存器的数据格式

通用寄存器的数据格式如图 2.9 所示。

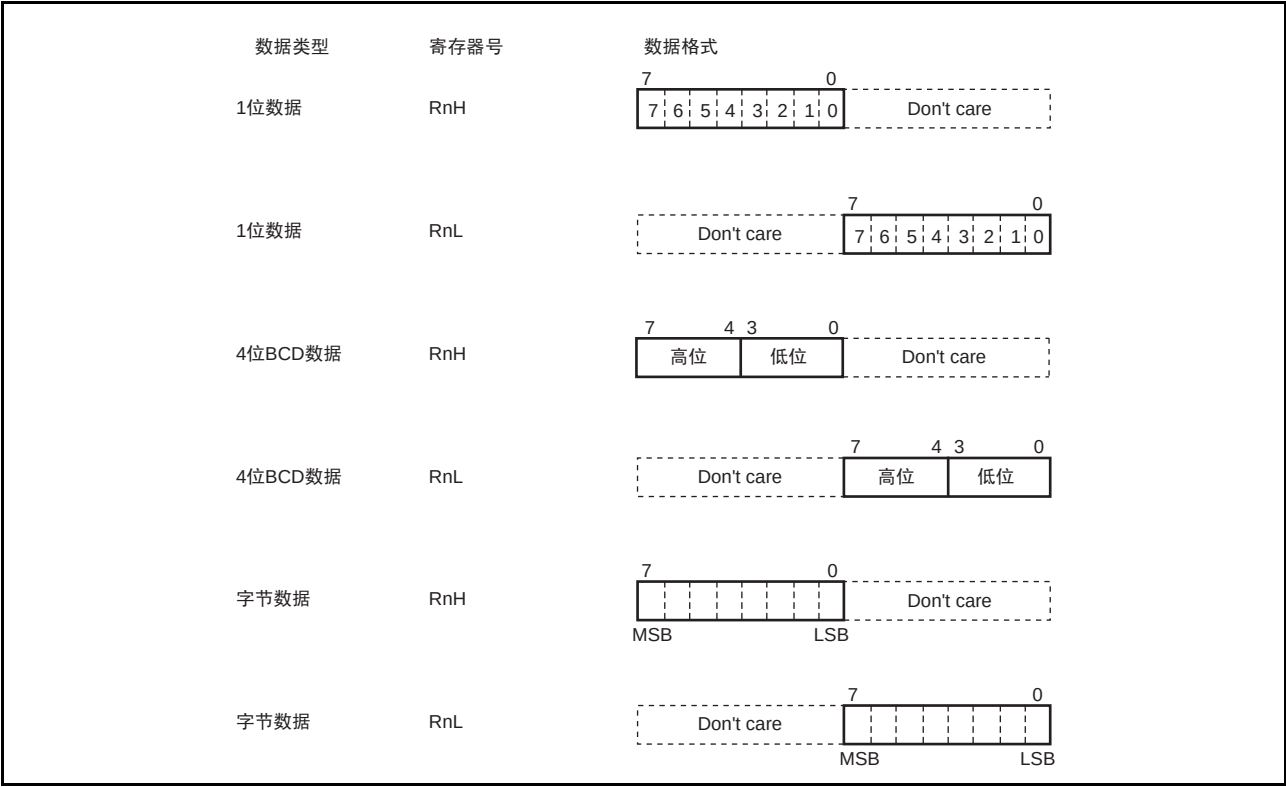


图 2.9 通用寄存器数据格式（1）

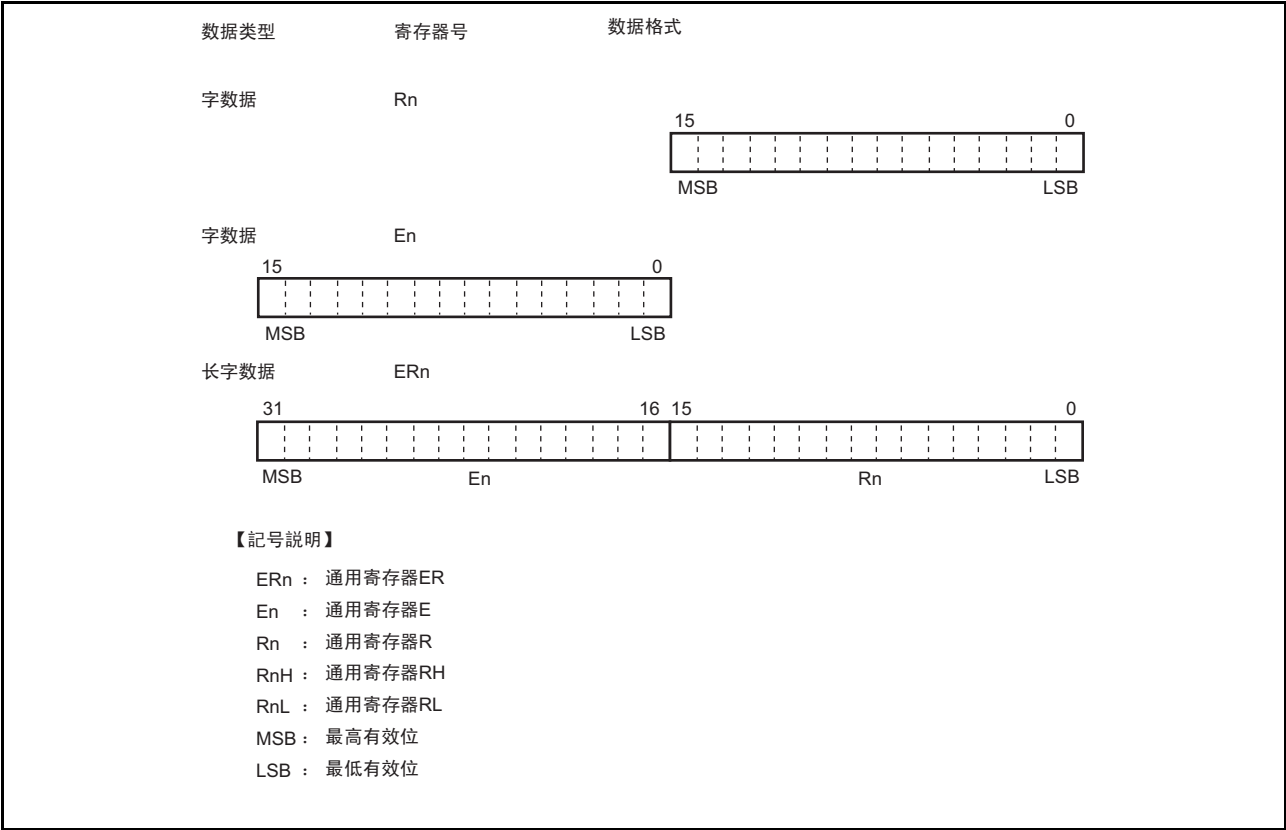


图 2.9 通用寄存器的数据格式（2）

2.5.2 存储器上的数据格式

内存的数据格式如图 2.10 所示。

H8S/2000CPU 可对内存的字节数据和长字节数据进行存取操作。这些数据必须从偶数地址开始。如果存取从奇数地址开始的字数据和长字段数据时，地址的最低位被当作 0，将存取从前一个地址开始的数据，在此不会发生地址操作错误，指令代码的获取也相同。

把 SP（ER7）作为地址寄存器进行堆栈区的存取操作时，必须以字或长字段进行存取操作。

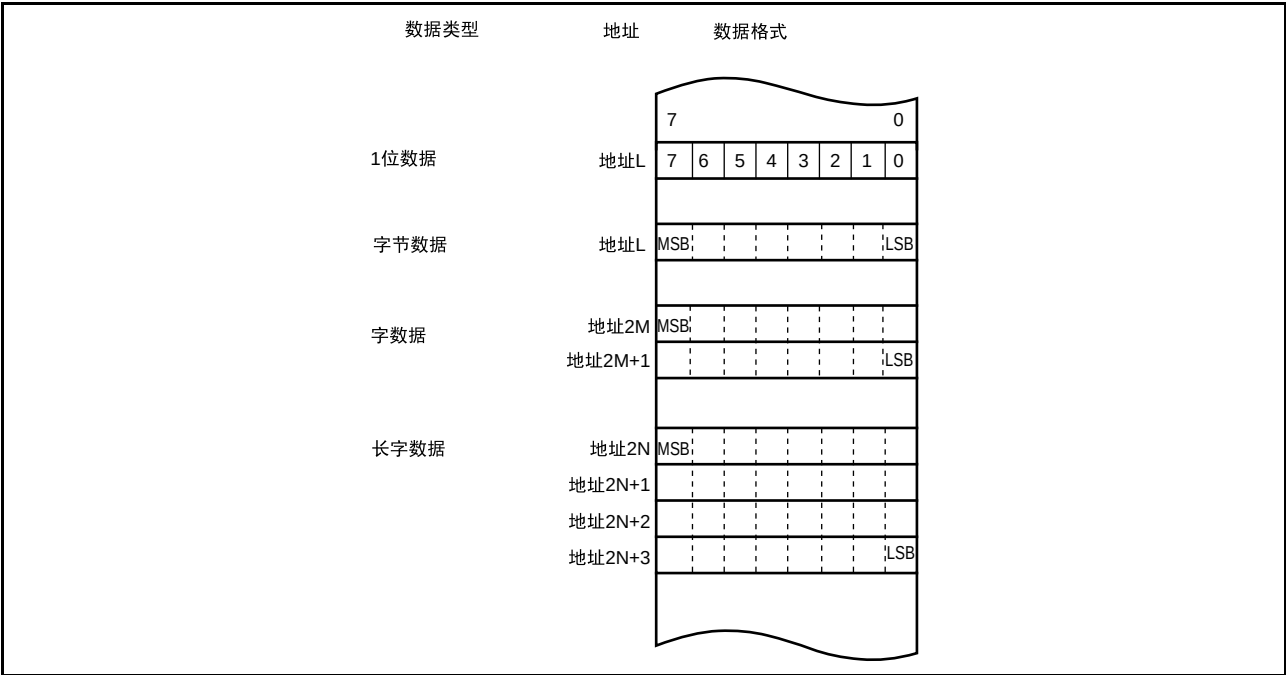


图 2.10 存储器的数据格式

2.6 指令系统

H8S/2000CPU 共有 65 种指令，各指令所含功能如表 2.1 所示。

表 2.1 指令分类

分类	指令	长度	种类
数据传输指令	MOV	B / W / L	5
	POP*1,PUSH*1	W / L	
	LDM,STM	L	
	MOVFP*3,MOVTP*3	B	
算术运算指令	ADD,SUB,CMP,NEG	B / W / L	19
	ADDX, SUBX,DAA,DAS	B	
	INC,DEC	B / W / L	
	ADDS,SUBS	L	
	MULXU,DIVXU, MULXS,DIVXS	B/W	
	EXTU, EXTS	W/L	
	TAS*4	B	
逻辑运算指令	AND, OR,XOR,NOT	B / W / L	4
移位指令	SHAL,SHAR,SHLL,SHLR,ROTL,ROTR,ROTXL,ROTXR	B / W / L	8
位操作指令	BSET,BCLR,BNOT,BTST,BLD,BILD,BST, BIST,BAND,BIAND,BOR,BIOR,BXOR,BIXOR	B	14
转移指令	Bcc*2,JMP,BSR,JSR,RTS	—	5
系统控制指令	TRAPA, RTE, SLEEP, LDC, STC, ANDC, ORC, XORC, NOP	—	9
块传输指令	EEPMOV	—	1

【注】 B: 字节 W: 字 L: 长字

*1 POP.W Rn、PUSH.W Rn 分别和 MOV.W @SP+,Rn、MOV.W Rn,@-SP 相同。

或、POP.L ERn、PUSH.L ERn 分别和 MOV.L @SP+,ERn、MOV.L ERn,@-SP 相同。

*2 Bcc 为条件转移指令的总称。

*3 不可用于本 LSI。

*4 要使用 TAS 指令时，请使用寄存器 ER0,ER1,ER4,ER5。

2.6.1 指令功能区别一览

各类指令功能如表 2.3 ～表 2.10 所示。使用各表时操作符号的意义如下图所示。

表 2.2 操作符号

符号	说明
Rd	目标通用寄存器 *
Rs	源通用寄存器 *
Rn	通用寄存器
ERn	通用寄存器（32 位寄存器）
(EAd)	目标操作数
(EAs)	源操作数
EXR	扩展寄存器
CCR	条件码寄存器
N	CCR 的 N（负）标志
Z	CCR 的 Z（零）标志
V	CCR 的 V（溢出）标志
C	CCR 的 C（进位）标志
PC	程序计数器
SP	堆栈指针
#IMM	立即数
disp	偏移量
+	加法
—	减法
×	乘法
÷	除法
∧	逻辑或
∨	逻辑与
⊕	逻辑异或
→	传送
~	非（逻辑补码）
:8 / :16 / :24 / :32	8 / 16 / 24 / 32 位长度

【注】* 通用寄存器有 8 位（R0H ～ R7H、R0L ～ R7L）、16 位（R0 ～ R7、E0 ～ E7）、或者 32 位（ER0 ～ ER7）寄存器。

表 2.3 数据传输指令

指令	长度 *	功能
MOV	B / W / L	(EAs)→Rd、Rs→(EAd) 在两个通用寄存器之间、寄存器和存储器间传输数据，或将立即数传送至通用寄存器。
MOVFP	B	不可用于本 LSI。
MOVTPE	B	不可用于本 LSI。
POP	W/L	@SP+→Rn 从堆栈弹出数据送往通用寄存器。 POP.W Rn 与 MOV.W @SP+, Rn 的功能一致，POP.L ERn 与 MOV.L @SP+, ERn 的功能一致。
PUSH	W/L	Rn→@-SP 将通用寄存器的内容保存到堆栈。 PUSH.W Rn 与 MOV.W Rn, @-SP 的功能相同。 PUSH.L ERn 与 MOV.L ERn, @-SP 的功能相同。
LDM	L	@SP+→Rn（寄存器系列） 从堆栈返回数据到多个通用寄存器。
STM	L	Rn（寄存器系列）→@-SP 将多个通用寄存器的数据保存到堆栈。

【注】：* 类型指操作类型

B: 字节

W: 字

L: 长字

表 2.4 算数运算

指令	长度 *	功能
ADD SUB	B / W / L	$Rd \pm Rs \rightarrow Rd$ 、 $Rd \pm \#IMM \rightarrow Rd$ 在两个通用寄存器之间或通用寄存器和立即数之间执行加减运算。（字节立即数不可同寄存器中的字节数据进行减法运算，请使用 SUBX 指令或 ADD 指令）。
ADDX SUBX	B	$Rd \pm Rs \pm C \rightarrow Rd$ 、 $Rd \pm \#IMM \pm C \rightarrow Rd$ 两个寄存器间或寄存器与立即数间执行带有进位的加减法。
INC DEC	B / W / L	$Rd \pm 1 \rightarrow Rd$ 、 $Rd \pm 2 \rightarrow Rd$ 通用寄存器执行加减 1 或 2。（字节操作数只可加减 1）。
ADDS SUBS	L	$Rd \pm 1 \rightarrow Rd$ 、 $Rd \pm 2 \rightarrow Rd$ 、 $Rd \pm 4 \rightarrow Rd$ 将 32 位寄存器中的数据加减 1、2 或 4。
DAA DAS	B	$Rd(10 \text{ 进制调整}) \rightarrow Rd$ 调整通用寄存器中的加减运算结果，通过存取 CCR 产生 4 位 BCD 数据。
MULXU	B / W	$Rd \times Rs \rightarrow Rd$ 两个寄存器间执行无符号的乘法运算。 8 位 $\times$ 8 位 $\rightarrow$ 16 位或 16 位 $\times$ 16 位 $\rightarrow$ 32 位。
MULXS	B / W	$Rd \times Rs \rightarrow Rd$ 两个通用寄存器之间执行带符号的乘法运算。 8 位 $\times$ 8 位 $\rightarrow$ 16 位、16 位 $\times$ 16 位 $\rightarrow$ 32 位。
DIVXU	B / W	$Rd \div Rs \rightarrow Rd$ 两个通用寄存器之间执行无符号的除法运算。 16 位 $\div$ 8 位 $\rightarrow$ 8 位商及 8 位余数。 32 位 $\div$ 16 位 $\rightarrow$ 16 位商及 16 位余数。
DIVXS	B / W	$Rd \div Rs \rightarrow Rd$ 两个通用寄存器之间执行带符号的除法运算。 16 位 $\div$ 8 位 $\rightarrow$ 8 位商及 8 位余数 32 位 $\div$ 16 位 $\rightarrow$ 16 位商及 16 位余数。
CMP	B / W / L	$Rd - Rs$ 、 $Rd - \#IMM$ 两个通用寄存器之间或通用寄存器和立即数据之间执行比较，且根据所得结果设置 CCR。
NEG	B / W / L	$0 - Rd \rightarrow Rd$ 取通用寄存器中数据的补码（算术取补）。
EXTU	W / L	$Rd(\text{零扩展}) \rightarrow Rd$ 通过左边加零将 16 位寄存器的低 8 位字节数据扩展为字数据，或将 32 位寄存器中的低 16 位字扩展为长字数据。
EXTS	W / L	$Rd(\text{符号扩展}) \rightarrow Rd$ 用扩展符号位将 16 位寄存器的低位 8 位扩展为字数据，或将 32 位寄存器的低位 16 位扩展为长字数据。
TAS*2	B	$@ERd - 0、1 \rightarrow (< \text{bit}7 > \text{ of } @ERd)$ 测试存储器数据之后，最高位的位（位 7）设置为 1。

【注】：*1. 类型指操作类型

B: 字节

W: 字

L: 长字

*2. 使用 TAS 指令时，请使用寄存器 ER0,ER1,ER4,ER5。

表 2.5 逻辑运算指令

指令	长度 *	功能
AND	B / W / L	$Rd \wedge Rs \rightarrow Rd$ 、 $Rd \wedge \#IMM \rightarrow Rd$ 通用寄存器和通用寄存器之间或通用寄存器与立即数之间执行逻辑与操作。
OR	B / W / L	$Rd \vee Rs \rightarrow Rd$ 、 $Rd \vee \#IMM \rightarrow Rd$ 通用寄存器和通用寄存器之间或通用寄存器与立即数之间执行逻辑或操作。
XOR	B / W / L	$Rd \oplus Rs \rightarrow Rd$ 、 $Rd \oplus \#IMM \rightarrow Rd$ 通用寄存器和通用寄存器之间或通用寄存器与立即数之间执行逻辑异或操作。
NOT	B / W / L	$\sim Rd \rightarrow Rd$ 对通用寄存器中的数据执行逻辑取反操作。

【注】：* 大小指操作大小。

B: 字节

W: 字

L: 长字

表 2.6 移位指令

指令	长度 *	功能
SHAL SHAR	B / W / L	$Rd(\text{移位处理}) \rightarrow Rd$ 对通用寄存器的内容执行算术移位。 可移动 1 位或 2 位。
SHLL SHLR	B / W / L	$Rd(\text{移位处理}) \rightarrow Rd$ 对通用寄存器的内容执行逻辑移位。 可移动 1 位或 2 位。
ROTL ROTR	B / W / L	$Rd(\text{循环处理}) \rightarrow Rd$ 对通用寄存器的内容执行循环移位。 可循环移动 1 位或 2 位。
ROTXL ROTXR	B / W / L	$Rd(\text{循环处理}) \rightarrow Rd$ 对通用寄存器的内容执行包含进位标志的循环移位。 可循环移动 1 位或 2 位。

【注】：* 大小指操作大小。

B: 字节

W: 字

L: 长字

表 2.7 位操作指令

指令	长度 *	功能
BSET	B	1→(< 位编号 >of<EAd>) 将通用寄存器或存储器操作数中指定的某位置 1。 位编号由一个 3 位立即数或一个通用寄存器的低 3 位来确定。
BCLR	B	0→(< 位编号 >of<EAd>) 将通用寄存器或存储器操作数中指定的某一位清零。 位编号由一个 3 位立即数或一个通用寄存器的低 3 位来确定。
BNOT	B	~ (< 位编号 >of<EAd>)→(< 位编号 >of<EAd>) 将通用寄存器或存储器操作数中指定的某一位取反。 位编号由一个 3 位立即数或一个通用寄存器的低 3 位来确定。
BTST	B	~ (< 位编号 >of<EAd>)→Z 测试通用寄存器或存储器操作数中指定的某一位并将结果反映到零标志位。 位编号由一个 3 位立即数或一个通用寄存器的低 3 位来确定。
BAND	B	$C \wedge (< \text{位编号} > \text{of} < \text{EAd} >) \rightarrow C$ 将通用寄存器或存储器操作数中指定的某一位与进位标志位进行逻辑与运算，结果保留在进位标志位中。
BIAND	B	$C \wedge (\sim (< \text{位编号} > \text{of} < \text{EAd} >)) \rightarrow C$ 将通用寄存器或存储器操作数中指定的某一位取反，并和进位标志位进行逻辑与运算，结果保留在进位标志位中。 位编号由一个 3 位立即数确定。
BOR	B	$C \vee (< \text{位编号} > \text{of} < \text{EAd} >) \rightarrow C$ 将通用寄存器或存储器操作数中指定的某一位与进位标志位进行逻辑或运算，结果保留在进位标志位中。
BIOR	B	$C \vee (\sim (< \text{位编号} > \text{of} < \text{EAd} >)) \rightarrow C$ 将通用寄存器或存储器操作数中指定的某一位取反，并和进位标志位进行逻辑或运算，结果保留在进位标志位中。 位编号由一个 3 位立即数确定。
BXOR	B	$C \oplus (\text{EAd 的位编号}) \rightarrow C$ 将通用寄存器或存储器操作数中指定的某一位与进位标志位进行逻辑异或运算，结果保留在进位标志位中。
BIXOR	B	$C \oplus (\sim \text{EAd 的位编号}) \rightarrow C$ 将通用寄存器或存储器操作数中指定的某一位取反，并与进位标志位进行逻辑异或运算，结果保留在进位标志位中。 位编号由一个 3 位立即数确定。
BLD	B	$(< \text{位编码} > \text{of} < \text{EAd} >) \rightarrow C$ 将通用寄存器或存储器操作数中被指定的某一位传送到进位标志位中。
BILD	B	$\sim (< \text{位编号} > \text{of} < \text{EAd} >) \rightarrow C$ 将通用寄存器或存储器操作数中被指定的某一位取反，并传送到进位标志位中。 位编号由一个 3 位立即数确定。
BST	B	$C \rightarrow (< \text{位编号} > \text{of} < \text{EAd} >)$ 将进位标志位的内容传送到通用寄存器或存储器操作数中被指定的某一位中。
BIST	B	$\sim C \rightarrow (< \text{位编号} > \text{of} < \text{EAd} >)$ 将进位标志位的内容取反后传送到通用寄存器或存储器操作数中被指定的某一个位中。 位编号由一个 3 位立即数确定。

【注】* 大小指操作数大小。

B: 字节

表 2.8 转移指令

指令	类型	功能																																																			
Bcc	—	当指定条件成立时，向已指定的地址转移，转移条件如下表所示。 <table> <tr> <th>助记符</th><th>说 明</th><th>转移条件</th></tr> <tr> <td>BRA(BT)</td><td>Always(true)</td><td>Always</td></tr> <tr> <td>BRN(BF)</td><td>Never(false)</td><td>Never</td></tr> <tr> <td>BHI</td><td>High</td><td>$C \vee Z = 0$</td></tr> <tr> <td>BLS</td><td>Low or Same</td><td>$C \vee Z = 1$</td></tr> <tr> <td>BCC(BHS)</td><td>Carry Clear(High or Same)</td><td>$C = 0$</td></tr> <tr> <td>BCS(BLO)</td><td>Carry Set(low)</td><td>$C = 1$</td></tr> <tr> <td>BNE</td><td>Not Equal</td><td>$Z = 0$</td></tr> <tr> <td>BEQ</td><td>Equal</td><td>$Z = 1$</td></tr> <tr> <td>BVC</td><td>Overflow Clear</td><td>$V = 0$</td></tr> <tr> <td>BVS</td><td>Overflow Set</td><td>$V = 1$</td></tr> <tr> <td>BPL</td><td>PLus</td><td>$N = 0$</td></tr> <tr> <td>BMI</td><td>Mluns</td><td>$N = 1$</td></tr> <tr> <td>BGE</td><td>Greater or Equal</td><td>$N \oplus V = 0$</td></tr> <tr> <td>BLT</td><td>Less Than</td><td>$N \oplus V = 1$</td></tr> <tr> <td>BGT</td><td>Greater Than</td><td>$Z \vee (N \oplus V) = 0$</td></tr> <tr> <td>BLE</td><td>Less or Equal</td><td>$Z \vee (N \oplus V) = 1$</td></tr> </table>	助记符	说 明	转移条件	BRA(BT)	Always(true)	Always	BRN(BF)	Never(false)	Never	BHI	High	$C \vee Z = 0$	BLS	Low or Same	$C \vee Z = 1$	BCC(BHS)	Carry Clear(High or Same)	$C = 0$	BCS(BLO)	Carry Set(low)	$C = 1$	BNE	Not Equal	$Z = 0$	BEQ	Equal	$Z = 1$	BVC	Overflow Clear	$V = 0$	BVS	Overflow Set	$V = 1$	BPL	PLus	$N = 0$	BMI	Mluns	$N = 1$	BGE	Greater or Equal	$N \oplus V = 0$	BLT	Less Than	$N \oplus V = 1$	BGT	Greater Than	$Z \vee (N \oplus V) = 0$	BLE	Less or Equal	$Z \vee (N \oplus V) = 1$
助记符	说 明	转移条件																																																			
BRA(BT)	Always(true)	Always																																																			
BRN(BF)	Never(false)	Never																																																			
BHI	High	$C \vee Z = 0$																																																			
BLS	Low or Same	$C \vee Z = 1$																																																			
BCC(BHS)	Carry Clear(High or Same)	$C = 0$																																																			
BCS(BLO)	Carry Set(low)	$C = 1$																																																			
BNE	Not Equal	$Z = 0$																																																			
BEQ	Equal	$Z = 1$																																																			
BVC	Overflow Clear	$V = 0$																																																			
BVS	Overflow Set	$V = 1$																																																			
BPL	PLus	$N = 0$																																																			
BMI	Mluns	$N = 1$																																																			
BGE	Greater or Equal	$N \oplus V = 0$																																																			
BLT	Less Than	$N \oplus V = 1$																																																			
BGT	Greater Than	$Z \vee (N \oplus V) = 0$																																																			
BLE	Less or Equal	$Z \vee (N \oplus V) = 1$																																																			
JMP	—	无条件转移到指定地址。																																																			
BSR	—	转移到指定地址的子程序。																																																			
JSR	—	转移到指定地址的子程序。																																																			
RTS	—	从子程序返回。																																																			

表 2.9 系统控制指令

指令	长度 *	功能
TRAPA	—	进行陷阱指令的异常处理。
RTE	—	从异常处理程序返回。
SLEEP	—	转移到低功耗模式。
LDC	B/W	(EAs)→CCR、(EAs)→EXR 将源操作数或立即数传送到 CCR、EXR 寄存器中。虽然 CCR、EXR 寄存器长度为 8 位，但是从存储器传送数据到 CCR、EXR 寄存器时，以字节长度读数据。高 8 位为有效位。
STC	B/W	CCR→(EAd)、EXR→(EAd) 将 CCR、EXR 寄存器的内容传送到通用寄存器或存储器中。虽然 CCR、EXR 寄存器长度为 8 位，CCR、EXR 寄存器和存储器及通用寄存器传送数据时，可实现字节转换。高 8 位为有效位。
ANDC	B	$CCR \wedge \#IMM \rightarrow CCR$ 、 $EXR \wedge \#IMM \rightarrow EXR$ 取 CCR、EXR 中的数据 and 立即数逻辑与。
ORC	B	$CCR \vee \#IMM \rightarrow CCR$ 、 $EXR \vee \#IMM \rightarrow EXR$ 取 CCR、EXR 中的数据 or 立即数逻辑或。
XORC	B	$CCR \oplus \#IMM \rightarrow CCR$ 、 $EXR \oplus \#IMM \rightarrow EXR$ 取 CCR、EXR 中的数据 xor 立即数逻辑异或。
NOP	—	$PC+2 \rightarrow PC$ 使 PC 增加 2。

【注】* 大小指操作数大小。

B: 字节

W: 字

表 2.10 块传输指令

指令	类型	功能
EEPMOV.B	—	if R4L $\neq$ 0 then Repeat @ER5+ $\rightarrow$ @ER6+ R4L-1 $\rightarrow$ R4L Until R4L = 0 else next;
EEPMOV.W	—	if R4 $\neq$ 0 then Repeat @ER5+ $\rightarrow$ @ER6+ R4-1 $\rightarrow$ R4 Until R4 = 0 else next; 数据块传输指令。从 ER5 所示的地址开始，将 R4L 或 R4 中所指定的字节数长度的数据传送到 ER6 所示的地址处。传送结束后，执行下一条指令。

2.6.2 基本指令格式

H8S/2000CPU 的指令以 2 字节（字）为单位。各指令由操作字段（op），寄存器字段（r），有效扩展字段（EA）及条件字段（cc）构成。指令格式的例子如图 2.11 所示。

- 操作字段
表示指令功能，寻址方式和执行操作数的操作。操作字段包含指令的前 4 位，一些指令也有两个操作字段的情况。
- 寄存器字段
指定通用寄存器。地址寄存器为 3 位，数据寄存器为 3 位或 4 位。一些有两个寄存器字段，也有没有寄存器字段的情况。
- EA 扩展字段
8 位、16 位、或 32 位，指定立即数据、绝对地址或偏移量。
- 条件字段
指定（Bcc）指令的转移条件。

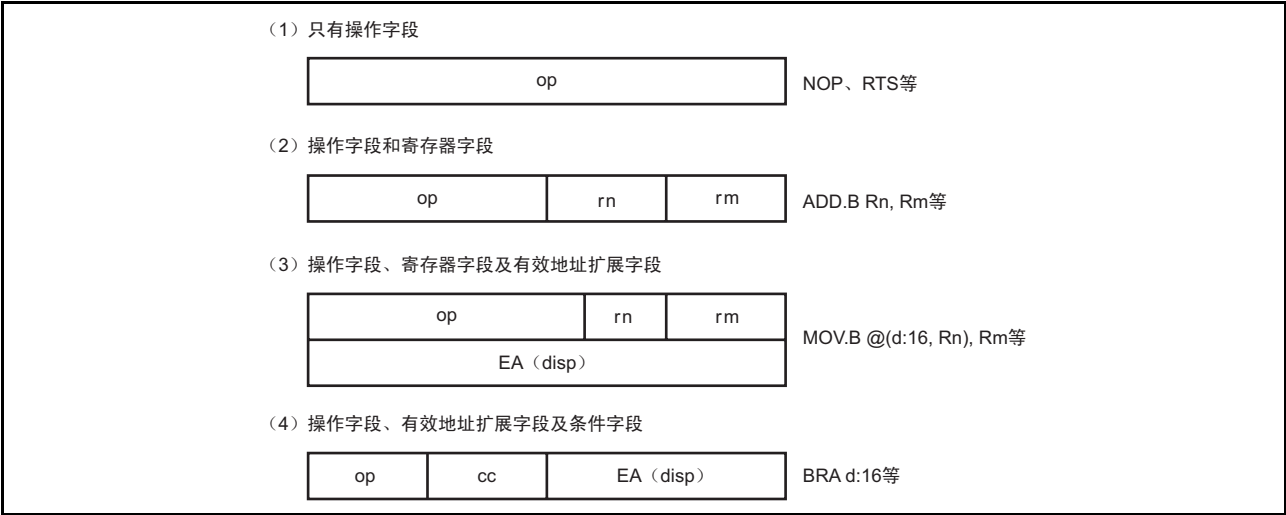


图 2.11 指令格式举例

2.7 寻址模式和有效地址的计算方法

H8S/2000 CPU 支持表 2.11 所示的 8 种寻址方式。各指令使用这些寻址方式中的一种。

运算指令和逻辑指令可使用寄存器直接和立即数寻址方式。传送指令除了程序计数器相对寻址和存储器间接寻址以外，所有的寻址方式都可使用。位操作指令可用寄存器直接、寄存器间接以及绝对地址指定操作数，并且可使用寄存器直接（BSET、BCLR、BNOT、BTST 的各指令）和立即数（3 位）寻址方式来指定操作数中的位编号。

表 2.11 寻址模式一览表

No.	寻址模式	记号
1	寄存器直接寻址	Rn
2	寄存器间接寻址	@ERn
3	带偏移量的寄存器间接寻址	@(d:16,ERn) / @(d:32,ERn)
4	后增寄存器间接寻址 先减寄存器间接寻址	@ERn+ @-ERn
5	绝对地址寻址	@aa:8 / @aa:16 / @aa:24 / @aa:32
6	立即寻址	#xx:8 / #xx:16 / #xx:32
7	程序计数器相对寻址	@(d:8,PC) / @(d:16,PC)
8	存储器间接寻址	@@aa:8

2.7.1 寄存器直接寻址 Rn

以指令码的寄存器字段指定的含有操作数的寄存器（8 位、16 位或 32 位）。

作为 8 位寄存器，可指定 R0H ~ R7H、R0L ~ R7L。

作为 16 位寄存器，可指定 R0 ~ R7、E0 ~ E7。

作为 32 位寄存器，可指定 ER0 ~ ER7。

2.7.2 寄存器间接寻址 @ERn

将指令码的寄存器字段指定为地址寄存器（ERn），其内容为操作数在内存中的地址。若地址是程序指令地址，低 24 位为有效位，高 8 位全部为 0。

2.7.3 带偏移量的寄存器间接寻址 @(d:16,ERn) / @(d:32,ERn)

将指令中包含的 16 位或 32 位偏移量与寄存器字段指定的地址寄存器的内容相加，其结果作为操作数在内存中的地址。相加时，16 位偏移量被符号扩展。

2.7.4 后增寄存器间接寻址 @ERn+ / 先减寄存器间接寻址 @-ERn

(1) 后增寄存器间接寻址 @ERn+

将指令码的寄存器字段指定的地址寄存器（ERn）内容作为存储器上的操作数的地址，。然后，地址寄存器的内容加 1、2 或 4，并将结果作为地址保存在地址寄存器中。字节存取加 1、字传送指令加 2、长字传送指令加 4。当转移指令是字或长字时，地址寄存器的内容应为偶数。

(2) 先减寄存器间接寻址 @-ERn

将指令码的寄存器字段指定的地址寄存器（ERn）内容减 1、减 2 或减 4，其结果作为地址指定存储器上的操作数。然后，结果也存入地址寄存器。字节存取减 1、字传送指令减 2、长传送指令字减 4。当转移指令是字或长字时，地址寄存器的内容应为偶数。

2.7.5 绝对地址 @aa:8 / @aa:16 / @aa:24 / @aa:32

用指令码中含有存储器的操作数的绝对地址。绝对地址可能为 8 位（@aa:8）、16 位（@aa:16）、24 位（@aa:24）或 32 位（@aa:32）。能够存取的绝对地址的范围如表 2.12 所示。

数据区域的操作中，使用 8 位（@aa:8）、16 位（@aa:16）、或 32 位（@aa:32）。8 位绝对地址时，高 24 位全部为 1（H'FFFF）。16 位绝对地址时，高 16 位为符号扩展。32 位绝对地址时，能够存取整个地址空间。

对于程序区域的操作，使用 24 位绝对地址（@aa:24）指示程序指令的地址，高 8 位全部为 0（H'00）。

表 2.12 绝对地址存取范围

绝对地址		普通模式	高级模式
数据区域	8 位（@aa:8）	H'FF00 ~ H'FFFF	H'FFFF00 ~ H'FFFFFF
	16 位（@aa:16）	H'0000 ~ H'FFFF	H'000000 ~ H'007FFF、 H'FF8000 ~ H'FFFFFF
	32 位（@aa:32）		H'000000 ~ H'FFFFFF
程序区域	24 位（@aa:24）		

【注】 * 不可用于本 LSI。

2.7.6 立即数寻址 #xx:8 / #xx:16 / #xx:32

将指令码中含有 8 位、(#xx:8)、16 位 (#xx:16) 或 32 位 (#xx:32) 数直接作为操作数使用。

ADDS、SUBS、INC、DEC 指令在指令码中隐含地包含立即数。位操作指令的指令码中包含一个指定位编号的 3 位立即数，TRAPA 指令的指令码中包含一个指示向量地址的 2 位立即数。

2.7.7 程序计数器相对寻址 @ (d:8, PC) / @ (d:16, PC)

在 Bcc、BSR 指令中使用。指令码中 8 位或 16 位偏移量加上 24 位的 PC 内容生成一个 24 位转移地址。加法运算执行时，偏移量被符号扩展为 24 位，加法运算结果的低 24 位有效，高 8 位全部为 0 (H'00)。与偏移量相加后的 PC 内容是下一条指令的起始地址，所以能够转移的范围为 -126 ~ +128 字节 (-63 ~ +64 字) 或者是 -32766 ~ +32768 字节 (-16383 ~ +16384 字)。此时，加法运算结果为偶数。

2.7.8 存储器间接寻址 @@aa:8

用于 JMP、JSR 指令。以指令码中包含的 8 位绝对地址来指定存储器上的操作数。该操作数作为一个偏移地址。由于 8 位绝对地址的高位全部为 0，因此转移地址的范围是 0 ~ 255 (普通模式时为 H'0000 ~ H'00FF、高级模式时为 H'000000 ~ H'0000FF)。

普通模式时，存储器的操作数指定为字操作数，跳转地址是 16 位。高级模式时，存储器操作数指定为长字操作数。操作数的首字节全部为 0 (H'00)。请注意，跳转地址的起始区域为异常向量区域。详细内容请参考「第 4 章 异常处理」。

对于用字或长字作为存储器的操作数或跳转地址时，如果所指向的存储器地址为奇数，那么最低位为 0，数据或指令码从前一个地址开始存取。(请参考「2.5.2 存储器上的数据格式」)

【注】 普通模式不可用于本 LSI。

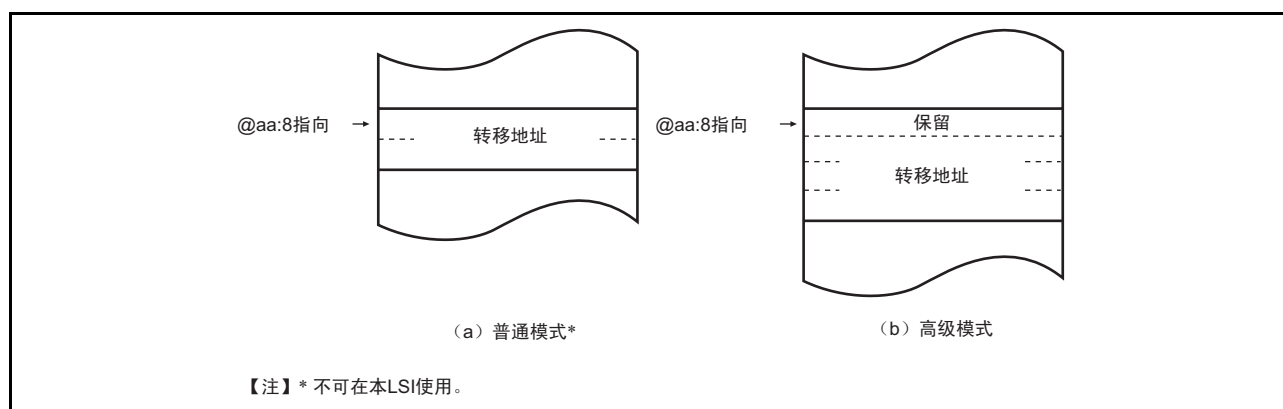


图 2.12 间接寻址模式下转移地址的说明

2.7.9 有效地址的计算方法

各种寻址方式的有效地址（EA：Effective Address）计算方法如表 2.13 所示。

普通模式时，有效地址的高 8 位被忽略，生成一个 16 位地址。

【注】 普通模式不可用于本 LSI。

表 2.13 有效地址的计算方法

No	寻址模式和指令格式	有效地址计算方法	有效地址（EA）
1	寄存器直接（Rn） op rm rn		操作数为通用寄存器的内容。
2	寄存器间接（@ERn） op r	31 通用寄存器的内容 0	31 Don't care 24 23 0
3	带位移的寄存器间接 @(d:16,ERn) / @(d:32,ERn) op r disp	31 通用寄存器的内容 0 31 符号扩展 0 disp	31 Don't care 24 23 0
4	后增寄存器间接 / 先减寄存器间接 · 后增寄存器间接 @ERn+ op r · 先减寄存器间接 @-ERn op r	31 通用寄存器的内容 0 1、2、或4 31 通用寄存器的内容 0 1、2、或4	31 Don't care 24 23 0 31 Don't care 24 23 0

操作数大小	加减值
字节	1
字	2
长字	4

No	寻址模式和指令格式	有效地址的计算方法	有效地址 (EA)
5	绝对地址 @aa:8 		
	@aa:16 		
	@aa:24 		
	@aa:32 		
6	立即方式 #xx:8 / #xx:16 / #xx:32 		操作数为立即数。
7	程序计数器相对 @(d:8,PC) / @(d:16,PC) 		
8	存储器间接 @@aa:8 · 普通模式* 		
	· 高级模式 		

【注】* 不可在本LSI中使用。

2.8 处理状态

H8S/2000 CPU 的处理状态有：复位状态、异常处理状态、程序执行状态、总线释放状态及低功耗状态 5 种。处理状态间的状态转移图如图 2.13 所示。

- 复位状态

CPU 以及内部外围模块全部为初始化、停止状态。复位管脚为低电平时，终止所有执行中的处理，CPU 处于复位状态。复位状态屏蔽所有的中断操作。复位管脚从低电平到高电平时，开始执行复位异常处理。复位的详细信息请参考「第 4 章 异常处理」。含有看门狗定时器的产品，由于看门狗定时器的溢出，它也能进行复位操作。

- 异常处理状态

异常处理状态是当 CPU 因为异常源，比如复位、跟踪、中断或陷阱指令而改变正常程序执行流程，从异常处理向量表中获得起始地址（向量），并向其起始地址转移而产生的一个过度状态。

- 程序执行状态

CPU 按顺序执行程序指令。

- 总线释放状态

CPU 响应从 CPU 以外的总线控制器发出的总线请求，进入总线释放状态。当处于总线释放状态时，CPU 操作停止。

- 程序停止状态

停止 CPU 操作处于掉电状态。执行 SLEEP 指令或向硬件待机模式转换时，为程序停止状态。详细信息请参考「第 20 章 低功耗状态」。

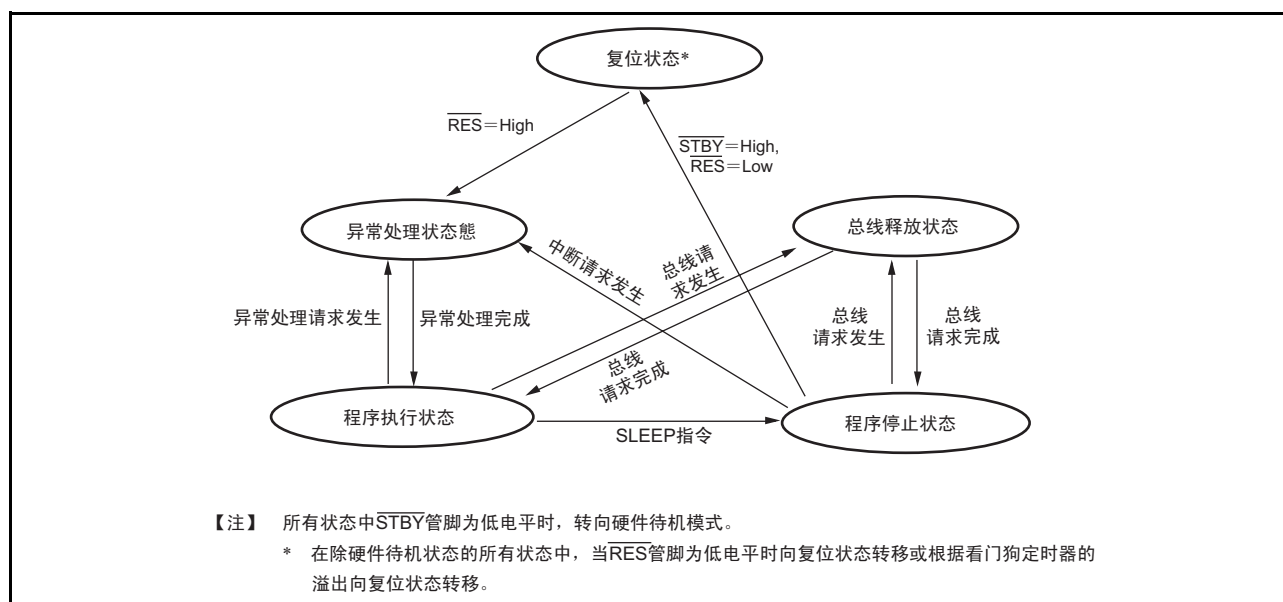


图 2.13 状态迁移图

2.9 使用注意事项

2.9.1 位操作指令使用注意事项

BSET、BCLR、BNOT、BST、BIST 位操作指令以字节为单位读取数据，位操作后再以字节为单位写入数据。但是在寄存器中或端口包含只写专用数据，在使用这些指令时，需要注意。另外，对于内部 I/O 寄存器标志清零，能够使用 BCLR 指令。此时，在中断处理程序中如果该标志明显被置为 1，那么就不必事先读取该标志。

第 3 章 MCU 工作模式

3.1 工作模式的选择

本 LSI 只支持工作模式 7 的高级单片模式。通过设置模式管脚（MD2、MD0）来决定工作模式。除模式 7 以外，其他所有模式均不可在本 LSI 环境下使用。因此，如表 3.1 所示必须将全部模式管脚设置为高电平。另外，在工作过程中不能改变模式管脚的状态。

表 3.1 MCU 工作模式的选择

MCU 工作模式	MD2	MD0	CPU 工作模式	内容	内置 ROM	外部数据总线	
						初始值	最大值
7	1	1	高级模式	单片模式	有效	—	—

3.2 寄存器的说明

与工作模式有关的寄存器有以下几种。

- 模式控制寄存器（MDCR）
- 系统控制寄存器（SYSCR）

3.2.1 模式控制寄存器（MDCR）

位	位名	初始值	R/W	说明
7	—	1	R/W	保留位 写入时，必须设置为 1。
6~3	—	全为 0	—	保留位 读取时，值固定为 0。写入定义为无效。
2	MDS2	—	R	表示模式管脚（MD2）的输入电平值（现在的工作模式）。此位与 MD2 管脚对应。此位专用于读取，写入定义为无效。读取 MDCR 时，这些位将 MD2 的输入电平锁存。通过复位，解除此锁存。
1	—	1	R	保留位 读取时，值固定为 1。写入定义为无效。
0	MDS0	—	R	表示模式管脚（MD0）的输入电平值（现在的工作模式）。此位与 MD0 管脚对应。此位专用于读取，写入定义为无效。读取 MDCR 时，这些位将 MD0 的输入电平锁存。通过复位，解除此锁存。

3.2.2 系统控制寄存器（SYSCR）

SYSCR 执行中断控制模式的选择，NMI 检测边沿的选择，内部 RAM 有效 / 无效的选择。

位	位名	初始值	R/W	说明
7	—	0	R/W	保留位 写入时，必须设置为 0。
6	—	0	—	保留位 读取时，值固定为 0。写入定义为无效。
5 4	INTM1 INTM0	0 0	R/W R/W	选择中断控制器的中断控制模式。关于中断控制模式请参照「5.6 中断控制模式和中断运行」。 00: 中断控制模式 0 01: 禁止设定 10: 中断控制模式 2 11: 禁止设定
3	NMIEG	0	R/W	NMI 边沿选择 选择 NMI 管脚的输入边沿。 0: 在 NMI 输入的下降沿，产生中断请求。 1: 在 NMI 输入的上升沿，产生中断请求。
2、1	—	全为 0	—	保留位 读取时，值固定为 0。写入定义为无效。
0	RAME	1	R/W	RAM 允许 选择内置 RAM 有效或无效。在解除复位时初始化此位。 0: 内置 RAM 无效 1: 内置 RAM 有效

3.3 各工作模式的说明

CPU 在高级模式下，地址空间为 16M 字节。内部 ROM 为有效，但是外部地址不能存取。所有的 I/O 端口都可作为输入 / 输出端口使用。

3.4 地址映射

各工作模式的地址映射如图 3.1，图 3.2 所示。

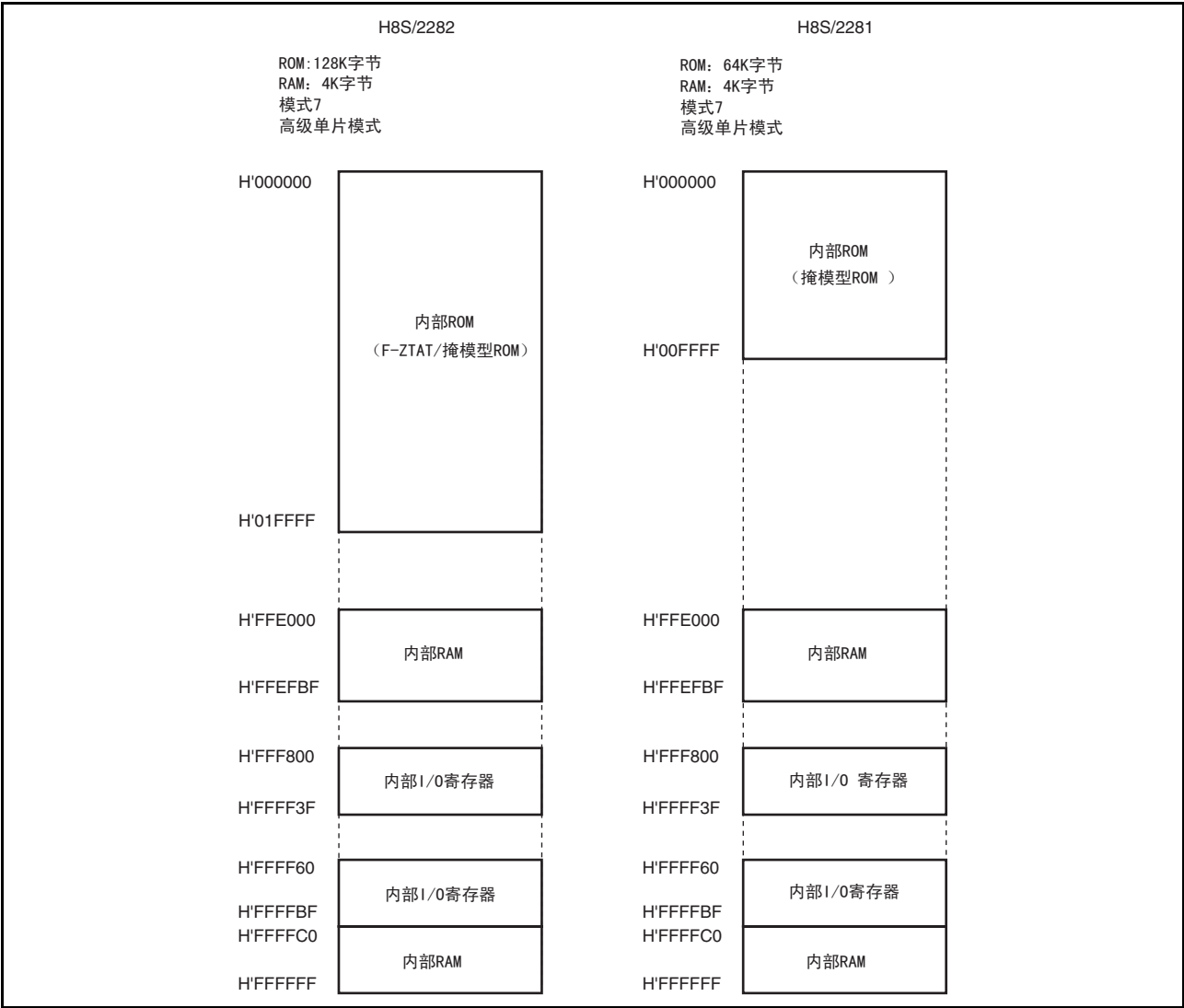


图 3.1 地址映射 (1)

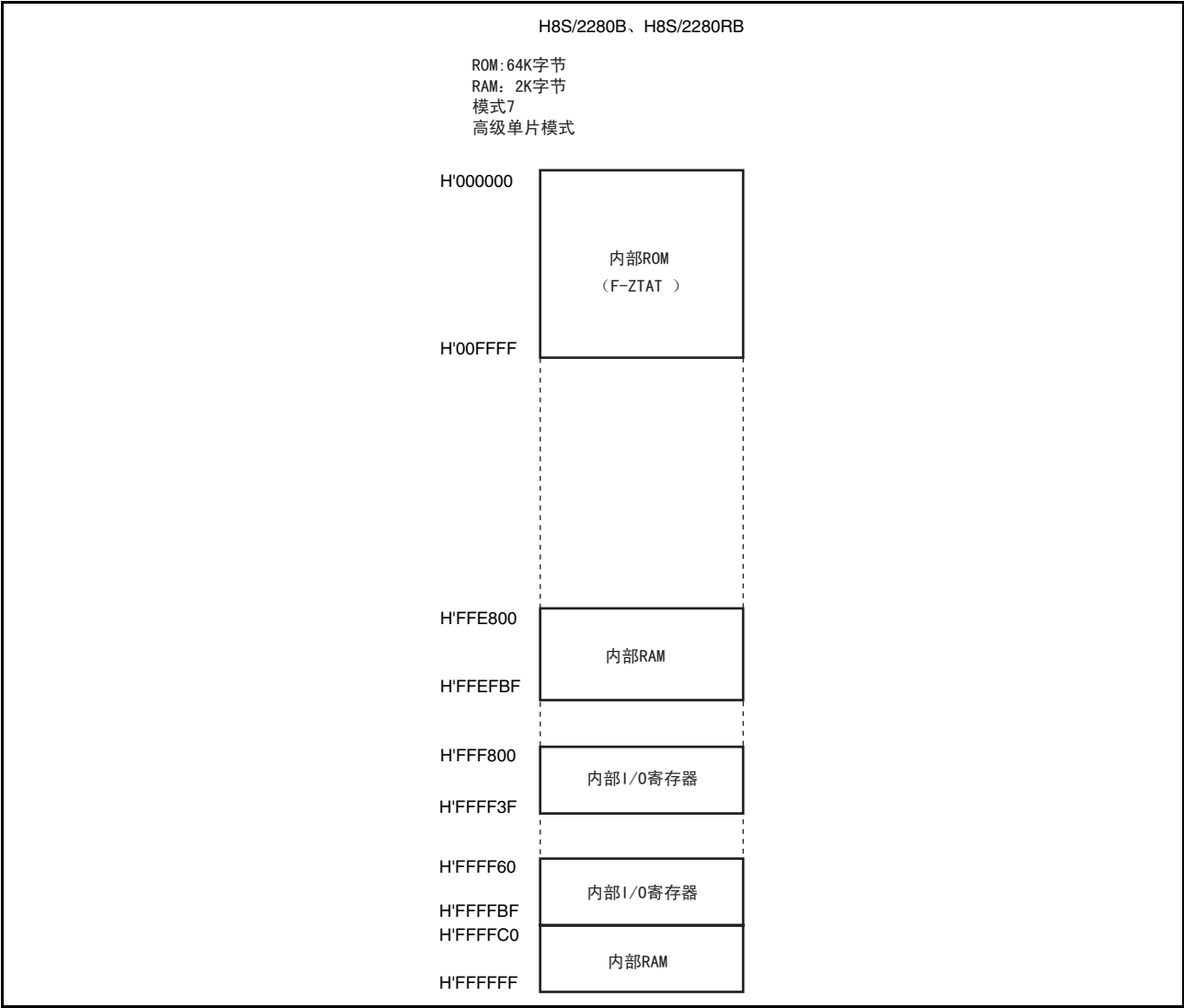


图 3.2 地址映射 (2)

第 4 章 异常处理

4.1 异常处理的种类和优先级

如表 4.1 所示，异常源由复位，跟踪，中断，及陷阱指令产生。这些异常源，设有优先级，当两个或多个异常源同时发生时，按照优先级顺序从高到低进行处理。异常处理根据中断控制模式不同，异常源，堆栈结构，CPU 运行亦有所不同。有关中断控制模式的详细情况，请参照「第 5 章 中断控制器」。

表 4.1 异常处理的种类和优先级

优先级	异常处理的种类	异常处理开始时序
高 ↑ 低	复位	当 $\overline{\text{RES}}$ 管脚从低电平转移到高电平时，或者看门狗定时器溢出，则开始异常处理。当 RES 管脚为低电平时，进入复位状态。
	跟踪 *1	当 EXR 的跟踪 (T) 位 =1 的状态时，在结束执行指令或者在结束异常处理时，开始异常处理。
	直接转移	当发生直接转移，则通过执行 SLEEP 指令，开始异常处理。
	中断	当发生中断请求，则在结束执行指令或者在结束异常处理时，开始异常处理。 *2
	陷阱指令 *3	通过执行陷阱 (TRAPA) 指令，开始异常处理。

【注】 *1 跟踪只在中断控制模式 2 中有效。结束执行 RTE 指令后，不执行跟踪异常处理。

*2 结束执行 ANDC、ORC、XORC、LDC 指令，或者结束复位异常处理时，不执行中断检测。

*3 在执行程序状态下，随时接受陷阱指令异常处理。

4.2 异常源 和 向量表

异常源有各种各样不同的向量地址。异常源和向量地址的对应关系如表 4.2 所示。可使用的模式则由产品而不同，详情参照「第 3 章 MCU 工作模式」。

表 4.2 异常处理向量表

异常源	向量号	向量地址 *1	
		普通模式 *2	高级模式
加电复位	0	H'0000 ~ H'0001	H'0000 ~ H'0003
手工复位 *2	1	H'0002 ~ H'0003	H'0004 ~ H'0007
系统保留用	2	H'0004 ~ H'0005	H'0008 ~ H'000B
	3	H'0006 ~ H'0007	H'000C ~ H'000F
	4	H'0008 ~ H'0009	H'0010 ~ H'0013
跟踪	5	H'000A ~ H'000B	H'0014 ~ H'0017
中断（直接转移）*3	6	H'000C ~ H'000D	H'0018 ~ H'001B
中断（NMI）	7	H'000E ~ H'000F	H'001C ~ H'001F
陷阱指令（#0）	8	H'0010 ~ H'0011	H'0020 ~ H'0023
陷阱指令（#1）	9	H'0012 ~ H'0013	H'0024 ~ H'0027
陷阱指令（#2）	10	H'0014 ~ H'0015	H'0028 ~ H'002B
陷阱指令（#3）	11	H'0016 ~ H'0017	H'002C ~ H'002F
系统保留用	12	H'0018 ~ H'0019	H'0030 ~ H'0033
	13	H'001A ~ H'001B	H'0034 ~ H'0037
	14	H'001C ~ H'001D	H'0038 ~ H'003B
	15	H'001E ~ H'001F	H'003C ~ H'003F
外部中断 IRQ0	16	H'0020 ~ H'0021	H'0040 ~ H'0043
外部中断 IRQ1	17	H'0022 ~ H'0023	H'0044 ~ H'0047
外部中断 IRQ2	18	H'0024 ~ H'0025	H'0048 ~ H'004B
外部中断 IRQ3	19	H'0026 ~ H'0027	H'004C ~ H'004F
外部中断 IRQ4	20	H'0028 ~ H'0029	H'0050 ~ H'0053
外部中断 IRQ5	21	H'002A ~ H'002B	H'0054 ~ H'0057
系统保留用	22	H'002C ~ H'002D	H'0058 ~ H'005B
	23	H'002E ~ H'002F	H'005C ~ H'005F
内部中断 *4	24	H'0030 ~ H'0031	H'0060 ~ H'0063
	127	H'00FE ~ H'00FF	H'01FC ~ H'01FF

【注】 *1 表示起始地址的低 16 位。

*2 在本 LSI 环境下不能使用。

*3 关于直接转移，请参照「20.10 直接转移」。

*4 中断的向量表，请参照「5.5 中断异常处理向量表」。

4.3 复位

复位是最高优先级的异常处理。当 $\overline{\text{RES}}$ 管脚变为低电平时，停止所有执行中的处理，本 LSI 进入复位状态。为了确保本 LSI 复位，在加电时，必须使 $\overline{\text{RES}}$ 管脚至少保持 20ms 为低电平。另外，在运行过程中， $\overline{\text{RES}}$ 管脚必须至少保持 20 个状态为低电平。通过复位，CPU 的内部状态和内部外围模块的各寄存器被初始化。另外，通过看门狗定时器的溢出，也可以变为复位状态。详细情况请参照「第 9 章 看门狗定时器 (WDT)」。

复位后，中断控制模式变为 0。

4.3.1 复位异常处理

$\overline{\text{RES}}$ 管脚在保持一定时间的低电平变为高电平时，开始复位异常处理，本 LSI 的运行如下。

- 1. 初始化 CPU 的内部状态和内部外围模块的各个寄存器，清除 EXR 的 T 位，并将 EXR 和 CCR 的 I 位置为 1。
- 2. 读取复位异常处理的向量地址，并将其传送到 PC 后，从 PC 显示的地址开始执行程序。

复位顺序的例子如图 4.1 和图 4.2 所示。

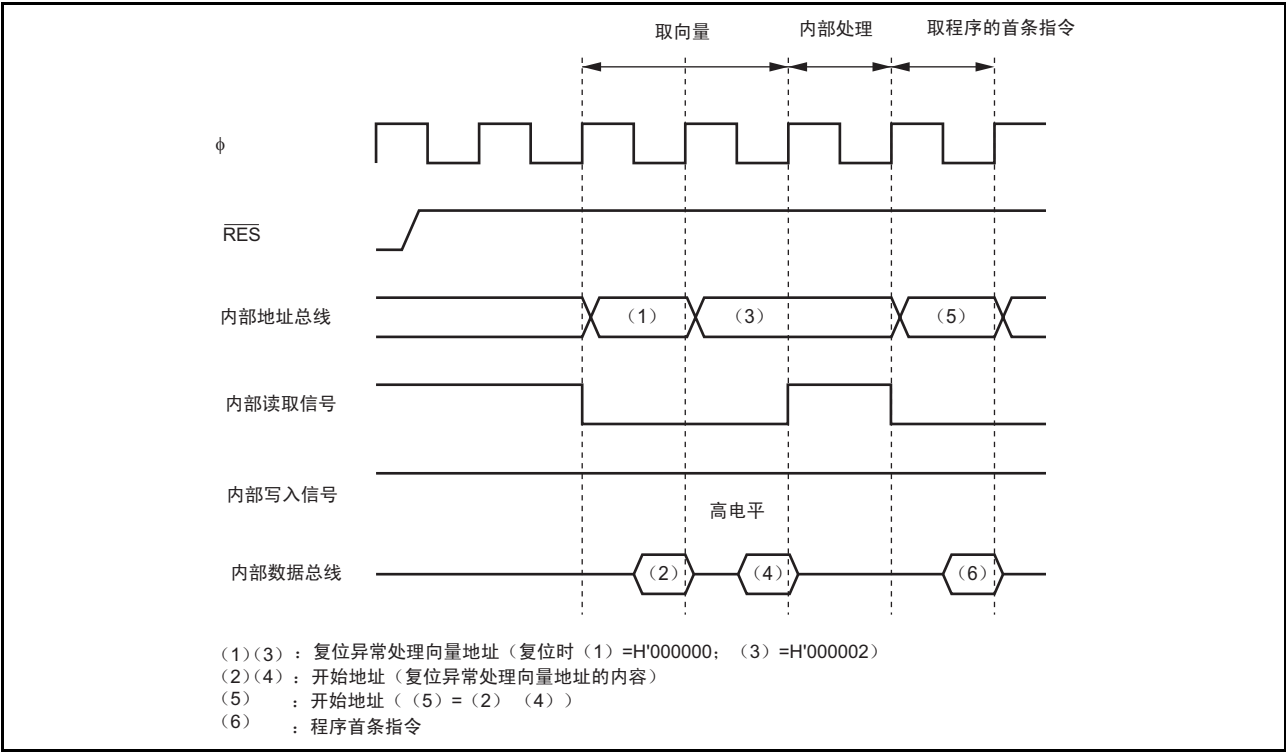


图 4.1 复位顺序 (高级模式 / 内部 ROM 有效)

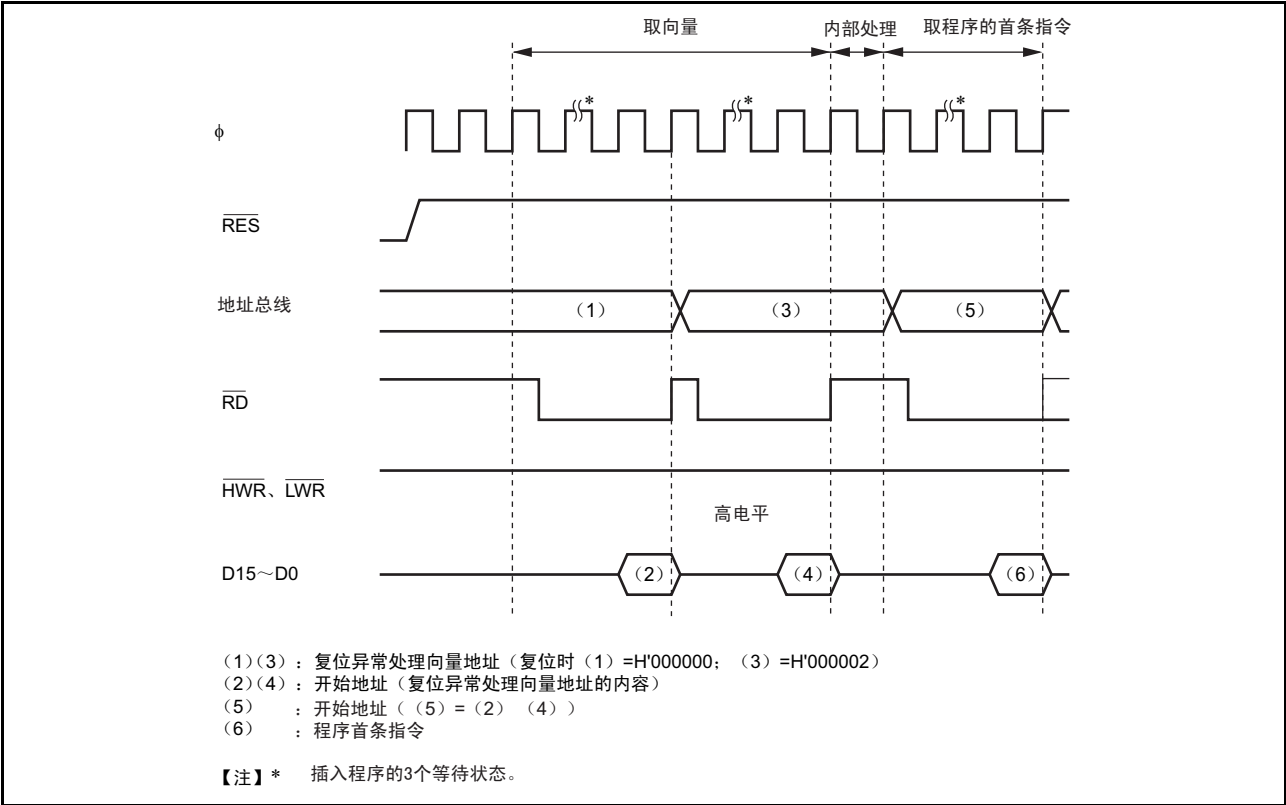


图 4.2 复位顺序 (高级模式 / 内部 ROM 无效: 本 LSI 环境下不能使用)

4.3.2 复位后的中断

复位后，如果在堆栈指针 (SP) 初始化前接受中断，PC 和 CCR 就无法正常保存，这样将导致程序失控。为了避免上述情况的发生，在执行完复位异常处理后，立即屏蔽包括 NMI 在内的所有中断请求。也就是说，复位后必须执行程序的第 1 条指令，因此，程序的首条指令必须为初始化 SP 的指令。(例: MOV.L # xx: 32, SP)。

4.3.3 复位解除后的内部外围功能

复位解除后，M STPCRA ~ MSTPCRD*1 初始化为 H'3F、H'FF、H'FF、B'11*****，所有模块均变为模块停止模式。

因此，各内部外围模块的寄存器不能读取 / 写入。通过解除模块停止模式，寄存器的读取 / 写入被使能。

【注】*1 MSTPCRD 的位 5 ~ 0 的初始值不定。

4.4 跟踪异常处理

跟踪在中断控制模式 2 下有效。在中断控制模式 0 下，不管 T 位的状态如何，都不变为跟踪模式。关于中断控制模式，请参照「第 5 章 中断控制器」。

EXR 的 T 位置为 1 时，变为跟踪模式。在跟踪模式下，每次 CPU 执行指令，都会开始跟踪异常处理。跟踪异常处理与 CCR 的中断屏蔽位无关。跟踪异常处理后 CCR，EXR 的状态如表 4.3 所示。通过跟踪异常处理，即使清除 EXR 的 T 位，并解除跟踪模式，保存在堆栈中的 T 位仍为 1；通过 RTE 指令，从跟踪异常处理程序恢复后，再次变为跟踪模式。在执行 RTE 指令时，不执行跟踪异常处理。

即使在跟踪异常处理程序中，也接受中断。

表 4.3 跟踪异常处理后 CCR，EXR 的状态

中断控制模式	CCR		EXR	
	I	UI	I2~I0	T
0	不能使用跟踪异常处理			
2	1	—	—	0

【符号说明】

1：置 1。

0：清 0。

—：保持执行前的值。

4.5 中断异常处理

中断由中断控制器控制。中断控制有两种模式，可以给除 NMI 以外的中断设置 8 个级别的优先级 / 屏蔽级，并可以执行多重中断的控制。根据产品不同，开始中断异常处理的因素和向量地址也均有所不同。详细情况请参照「第 5 章 中断控制器」。

中断异常处理的运行如下。

1. 将程序计数器（PC）和条件码寄存器（CCR），扩展寄存器（EXR）的内容保存在堆栈中。
2. 更新中断屏蔽位，并将 T 位清零。
3. 生成中断源对应的向量地址，从向量表中将起始地址装入 PC，从它的地址开始执行程序。

4.6 陷阱指令异常处理

如果执行 TRAPA 指令，就开始异常处理。在程序执行状态下，可随时执行陷阱指令异常处理。陷阱指令异常处理的运行如下。

- 1. 将程序计数器（PC）和条件码寄存器 (CCR), 扩展寄存器（EXR）的内容保存在堆栈中。
 - 2. 更新中断屏蔽位，并将 T 位清零。
 - 3. 向量地址是伴随着中断源产生而生成的，从向量表中将起始地址装入 PC，从它的地址开始执行程序。
- TRAPA 指令是从向量表中取出起始地址，此向量表与操作码中指定的 0 ~ 3 的向量号相对应。
- 结束执行陷阱指令异常处理后 CCR，EXR 的状态如表 4.4 所示。

表 4.4 陷阱命令异常处理后的 CCR，EXR 状态

中断控制模式	CCR		EXR	
	I	UI	I2~I0	T
0	1	—	—	—
2	1	—	—	0

【符号说明】

- 1：置 1。
- 0：清 0。
- ：保持执行前的值。

4.7 异常处理后的堆栈状态

陷阱指令异常处理及中断异常处理执行后的堆栈状态如图 4.3 所示。

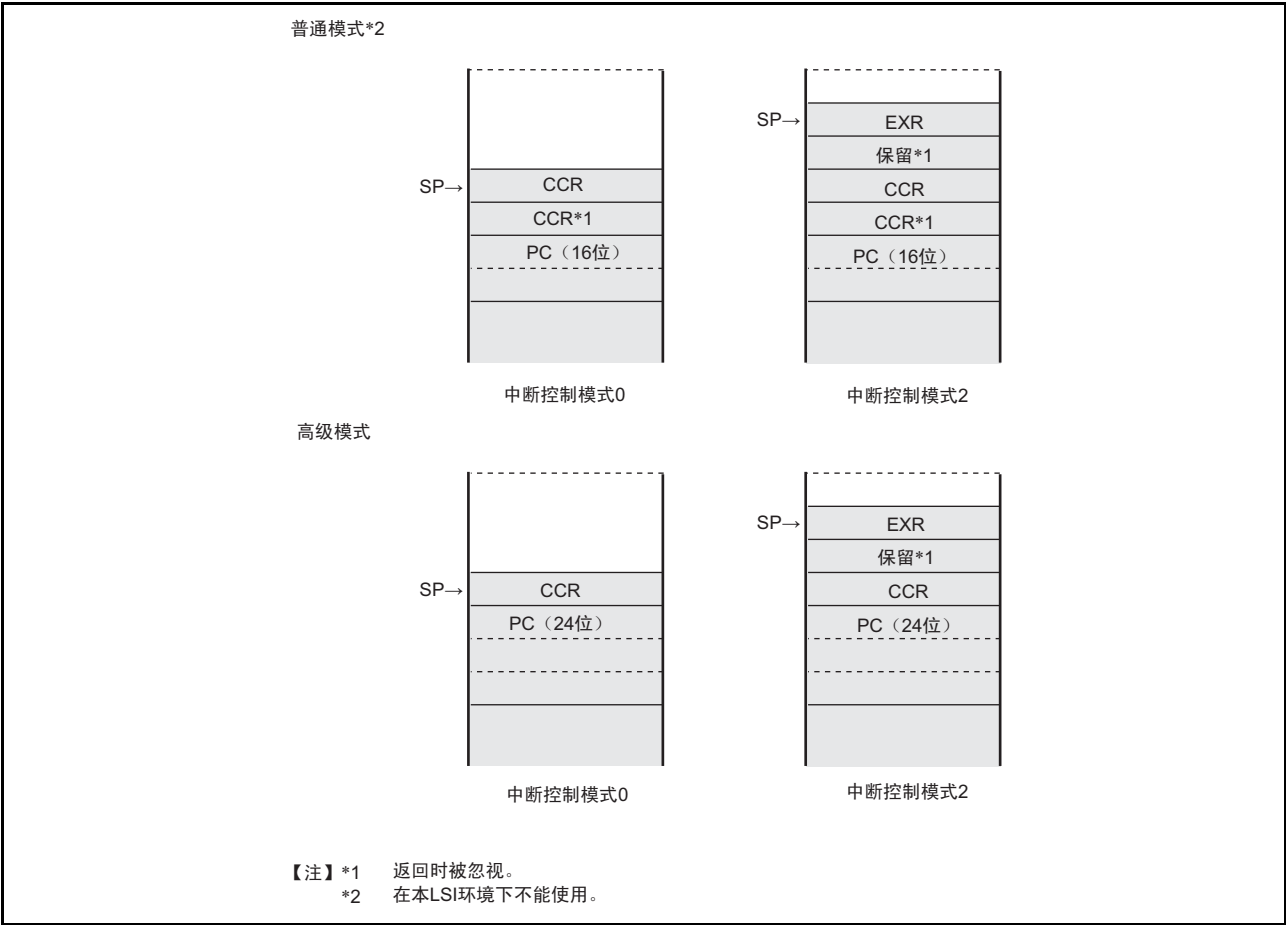


图 4.3 结束异常处理后的堆栈状态

4.8 使用中的注意事项

存取字数据或者长字数据时，地址的最低位视为 0。
堆栈区域的存取，总是按字符转移指令或长字符转移指令进行，堆栈指针（SP: ER7）的内容不能为奇数。

也就是说，保存寄存器使用的指令如下

```
PUSH.W Rn  (MOV.W Rn, @-SP)
PUSH.L ERn (MOV.L ERn, @-SP)
```

另外，寄存器的恢复必须使用

```
POP.W Rn  (MOV.W @SP+, Rn)
POP.L ERn (MOV.L @SP+, ERn)
```

如果将 SP 设置为奇数，则会成为误动作的原因。将 SP 设置为奇数时的可能运行情况如图 4.4 所示。

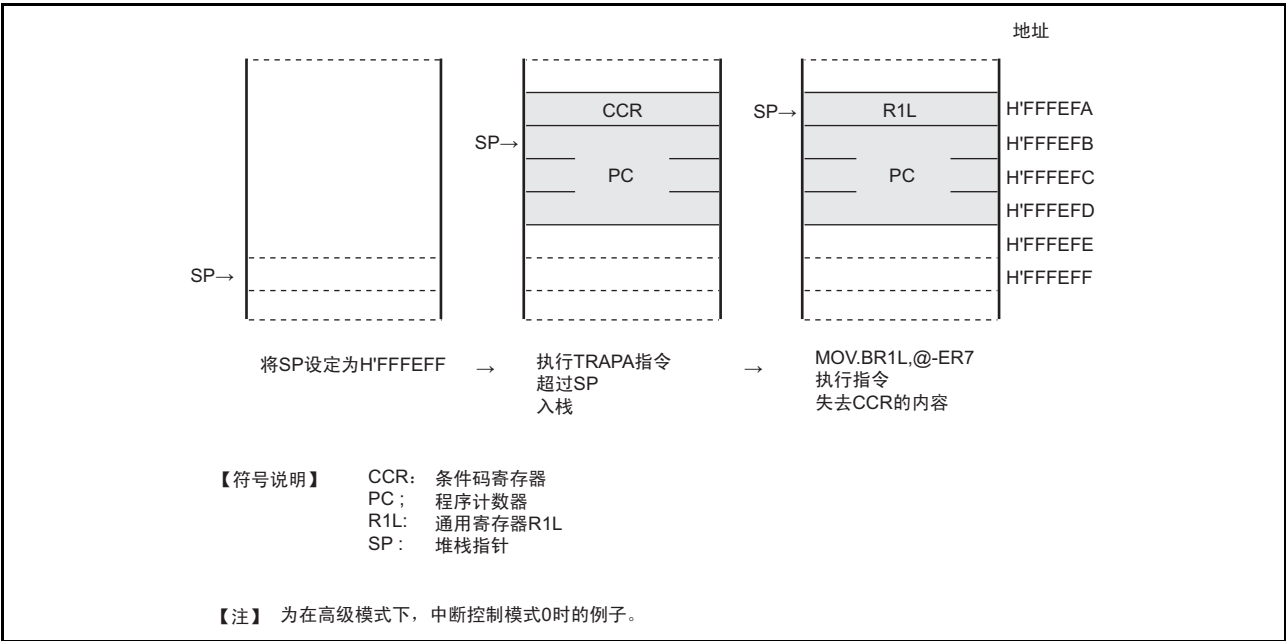


图 4.4 将 SP 设置为奇数时的运行

第 5 章 中断控制器

5.1 特点

- 2 种中断控制模式
通过设定系统控制寄存器（SYSCR）的 INTM1，INTM0 位，来设置 2 种中断控制模式。
- 可以通过 IPR，设定优先级。
通过中断优先级寄存器（IPR），可以给每个模块设置除 NMI 以外 8 个级别的优先级中断请求。NMI 作为 8 个级别中最优先的中断请求，随时都能被接受。
- 独立的向量地址
所有中断源都被赋予了独立的向量地址，不需要通过中断处理程序判别中断源。
- 7 个外部中断管脚
NMI 是优先级最高的中断，它随时都能被接受。NMI 可以选择上升沿或者下降沿。IRQ5 ~ IRQ0 可以独立并任意选择下降沿，上升沿，双边沿，或感知级别。

中断控制器的框图如图 5.1 所示。

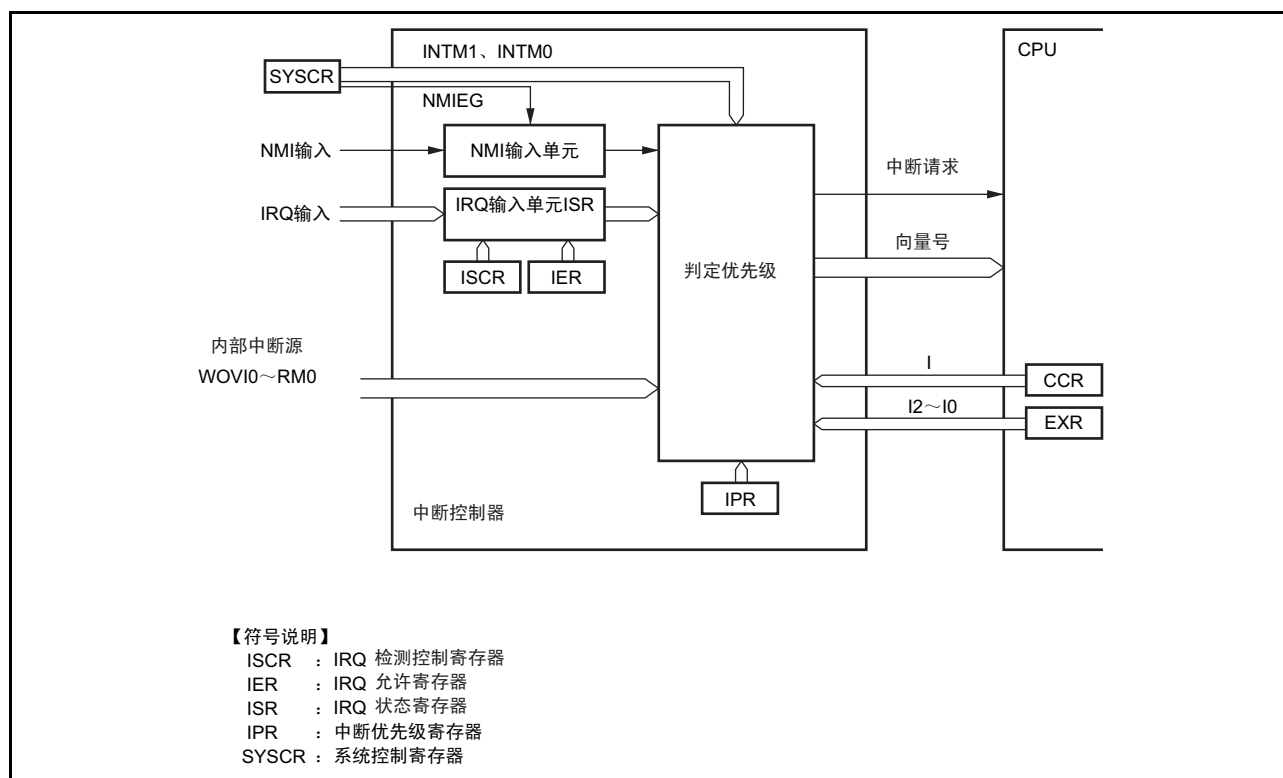


图 5.1 中断控制器的框图

5.2 输入 / 输出管脚

中断控制器的管脚结构如表 5.1 所示。

表 5.1 管脚结构

名称	输入 / 输出	功能
NMI	输入	非屏蔽外部中断管脚 可选择上升沿或下降沿
$\overline{\text{IRQ5}}$ $\overline{\text{IRQ4}}$ $\overline{\text{IRQ3}}$ $\overline{\text{IRQ2}}$ $\overline{\text{IRQ1}}$ $\overline{\text{IRQ0}}$	输入 输入 输入 输入 输入 输入	可屏蔽的外部中断管脚 可以独立并任意选择下降沿，上升沿，双边沿，或检测级别。

5.3 寄存器的说明

中断控制器有以下几种寄存器。关于系统控制寄存器（SYSCR），请参照「3.2.2 系统控制寄存器（SYSCR）」。

- 系统控制寄存器（SYSCR）
- IRQ 检测控制寄存器 H（ISCRH）
- IRQ 检测控制寄存器 L（ISCRL）
- IRQ 允许寄存器（IER）
- IRQ 状态寄存器（ISR）
- 中断优先级寄存器 A（IPRA）
- 中断优先级寄存器 B（IPRB）
- 中断优先级寄存器 C（IPRC）
- 中断优先级寄存器 D（IPRD）
- 中断优先级寄存器 E（IPRE）
- 中断优先级寄存器 F（IPRF）
- 中断优先级寄存器 G（IPRG）
- 中断优先级寄存器 J（IPRJ）
- 中断优先级寄存器 K（IPRK）
- 中断优先级寄存器 M（IPRM）

5.3.1 中断优先级寄存器 A ~ G、J、K、M (IPRA ~ IPRG、IPRJ、IPRK、IPRM)

IPR 设置除 NMI 以外的中断源优先级（级别 7 ~ 0）。IPR 有 10 个寄存器。各中断源和 IPR 的对应关系请参照表 5.2。通过给位 6 ~ 4，位 2 ~ 0 的每 3 位设置从 H'0 到 H'7 范围的值，来决定对应的中断请求优先级。

位	位名	初始值	R/W	说明
7	—	0	—	保留位 读取时，值固定为 0。
6 5 4	IPR6 IPR5 IPR4	1 1 1	R/W R/W R/W	设置对应的中断源优先级 000: 优先级 0（最低） 001: 优先级 1 010: 优先级 2 011: 优先级 3 100: 优先级 4 101: 优先级 5 110: 优先级 6 111: 优先级 7（最高）
3	—	0	—	保留位 读取时，值固定为 0。
2 1 0	IPR2 IPR1 IPR0	1 1 1	R/W R/W R/W	设置对应的中断源优先级。 000: 优先级 0（最低） 001: 优先级 1 010: 优先级 2 011: 优先级 3 100: 优先级 4 101: 优先级 5 110: 优先级 6 111: 优先级 7（最高）

5.3.2 IRQ 允许寄存器 (IER)

IER 允许 IRQ5 ~ IRQ0 中断请求。

位	位名	初始值	R/W	说明
7、6	—	全为 0	R/W	保留位 写入时，必须设置为 0。
5	IRQ5E	0	R/W	IRQ5 使能 此位为 1 时，IRQ5 中断请求使能。
4	IRQ4E	0	R/W	IRQ4 使能 此位为 1 时，IRQ4 中断请求使能。
3	IRQ3E	0	R/W	IRQ3 使能 此位为 1 时，IRQ3 中断请求使能。
2	IRQ2E	0	R/W	IRQ2 使能 此位为 1 时，IRQ2 中断请求使能。
1	IRQ1E	0	R/W	IRQ1 使能 此位为 1 时，IRQ1 中断请求使能。
0	IRQ0E	0	R/W	IRQ0 使能 此位为 1 时，IRQ0 中断请求使能。

5.3.3 IRQ 检测控制寄存器 H、L (ISCRH、ISCRL)

ISCR 从 $\overline{\text{IRQ5}}$ ~ $\overline{\text{IRQ0}}$ 管脚中选择发生中断请求的中断源。

位	位名	初始值	R/W	说明
15~12	—	全为 0	R/W	保留位 写入时，必须设置为 0。
11 10	IRQ5SCB IRQ5SCA	0 0	R/W R/W	IRQ5 检测控制 B IRQ5 检测控制 A 00: 在 $\overline{\text{IRQ5}}$ 输入的低电平，发生中断请求。 01: 在 $\overline{\text{IRQ5}}$ 输入的下降沿，发生中断请求。 10: 在 $\overline{\text{IRQ5}}$ 输入的上升沿，发生中断请求。 11: 在 $\overline{\text{IRQ5}}$ 输入的下降沿，上升沿双边沿，发生中断请求。

位	位名	初始值	R/W	说明
9 8	IRQ4SCB IRQ4SCA	0 0	R/W R/W	IRQ4 检测控制 B IRQ4 检测控制 A 00: 在 $\overline{\text{IRQ4}}$ 输入的低电平, 发生中断请求 01: 在 $\overline{\text{IRQ4}}$ 输入的下降沿, 发生中断请求 10: 在 $\overline{\text{IRQ4}}$ 输入的上升沿, 发生中断请求 11: 在 $\overline{\text{IRQ4}}$ 输入的下降沿, 上升沿双边沿, 发生中断请求。
7 6	IRQ3SCB IRQ3SCA	0 0	R/W R/W	IRQ3 检测控制 B IRQ3 检测控制 A 00: 在 $\overline{\text{IRQ3}}$ 输入的低电平, 发生中断请求 01: 在 $\overline{\text{IRQ3}}$ 输入的下降沿, 发生中断请求 10: 在 $\overline{\text{IRQ3}}$ 输入的上升沿, 发生中断请求 11: 在 $\overline{\text{IRQ3}}$ 输入的下降沿, 上升沿双边沿, 发生中断请求。
5 4	IRQ2SCB IRQ2SCA	0 0	R/W R/W	IRQ2 检测控制 B IRQ2 检测控制 A 00: 在 $\overline{\text{IRQ2}}$ 输入的低电平, 发生中断请求 01: 在 $\overline{\text{IRQ2}}$ 输入的下降沿, 发生中断请求 10: 在 $\overline{\text{IRQ2}}$ 输入的上升沿, 发生中断请求 11: 在 $\overline{\text{IRQ2}}$ 输入的下降沿, 上升沿双边沿, 发生中断请求。
3 2	IRQ1SCB IRQ1SCA	0 0	R/W R/W	IRQ1 检测控制 B IRQ1 检测控制 A 00: 在 $\overline{\text{IRQ1}}$ 输入的低电平, 发生中断请求 01: 在 $\overline{\text{IRQ1}}$ 输入的下降沿, 发生中断请求 10: 在 $\overline{\text{IRQ1}}$ 输入的上升沿, 发生中断请求 11: 在 $\overline{\text{IRQ1}}$ 输入的下降沿, 上升沿双边沿, 发生中断请求。
1 0	IRQ0SCB IRQ0SCA	0 0	R/W R/W	IRQ0 检测控制 B IRQ0 检测控制 A 00: 在 $\overline{\text{IRQ0}}$ 输入的低电平, 发生中断请求 01: 在 $\overline{\text{IRQ0}}$ 输入的下降沿, 发生中断请求 10: 在 $\overline{\text{IRQ0}}$ 输入的上升沿, 发生中断请求 11: 在 $\overline{\text{IRQ0}}$ 输入的下降沿, 上升沿双边沿, 发生中断请求。

5.3.4 IRQ 状态寄存器 (ISR)

ISR 是 IRQ5 ~ IRQ0 中断请求标志寄存器。

位	位名	初始值	R/W	说明
7、6	—	全为 0	R/W	保留位 写入时，必须设置为 0。
5	IRQ5F	0	R/W	[置位条件] • 在发生 ISCR 选择的中断源时 [清除条件] (初期值) • 在读取 1 的状态后，写入 0 时 • 在低电平检测设定状态并且 $\overline{\text{IRQn}}$ 输入为高电平状态下，执行中断异常处理 • 在检测出设定下降沿，上升沿，双边沿时的状态下，执行 IRQn 中断异常处理。
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	

5.4 中断源

5.4.1 外部中断源

外部中断有 NMI、IRQ5 ~ IRQ0，7 个中断源。其中，外部中断通常可用于从软件待机模式的唤醒。

(1) NMI 中断

非屏蔽中断请求 NMI 是优先级最高的外部中断请求，无论中断控制模式和 CPU 的中断屏蔽位的状态如何，它总是被接受。可以通过 SYSCR 的 NMIEG 位设置，选择是否在 NMI 管脚的上升沿或下降沿发生中断请求。

(2) IRQ5 ~ IRQ0 中断

IRQ5 ~ IRQ0 中断，通过 $\overline{\text{IRQ5}}$ ~ $\overline{\text{IRQ0}}$ 管脚的输入信号发生中断请求。IRQ5 ~ IRQ0 中断有以下特点。

- 可以通过 ISCR 选择中断请求是否在 $\overline{\text{IRQ5}}$ ~ $\overline{\text{IRQ0}}$ 管脚设置的低电平，下降沿，上升沿，或双边沿状态下发生。
- 可以通过 IER 使能 IRQ5 ~ IRQ0 中断请求。
- 可以通过 IPR 设置中断优先级。
- ISR 表示 IRQ5 ~ IRQ0 中断请求的状态。可以用软件清除 ISR 标志位。

IRQ5 ~ IRQ0 中断的检测，与将该管脚设定为输入还是输出无关。相反，作为外部中断输入管脚使用时，相应的 DDR 不能被清零，也不能作为其他功能的输入 / 输出管脚使用。

IRQ5 ~ IRQ0 中断的框图如图 5.2 所示。

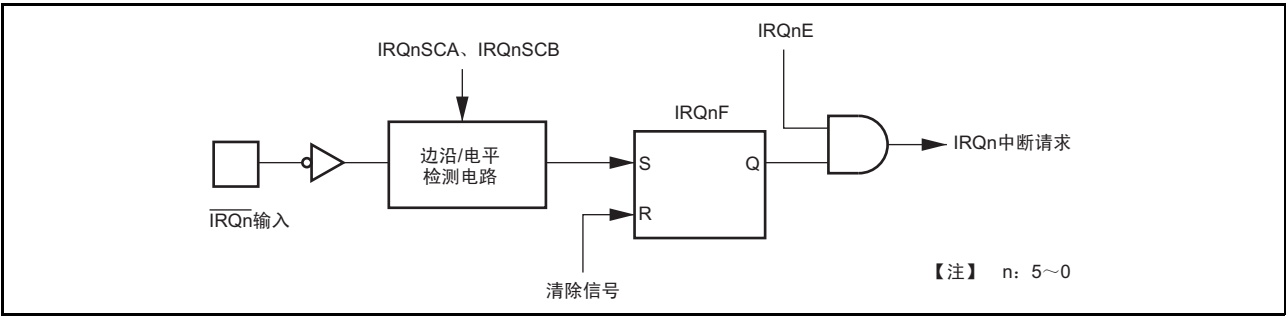


图 5.2 IRQ5 ~ IRQ0 中断的框图

5.4.2 内部中断

内部外围模块的内部中断源有以下特点。

- 各内部外围模块，有表示中断请求状态的标志位和允许位，并能独立屏蔽。如果这两个位对于某个特殊中断源都被置 1，中断请求被发送到中断控制器。
- 可以通过 IPR 设置中断优先级。

5.5 中断异常处理向量表

表 5.2 为中断异常源，向量地址及中断优先级一览表。向量号越小，默认值的优先级越高。模块间的优先级可以通过 IPR 来设置。优先级相同的模块，按默认的优先级选取。模块内的优先级是固定的。

表 5.2 中断源、向量地址及中断优先级一览表

中断源	名称	向量号	向量地址 *1	IPR	优先级
			高级模式		
外部管脚	NMI	7	H'001C		高 ↑

中断源	名称	向量号	向量地址 *1	IPR	优先级
			高级模式		
PWM	CMI1	104	H'01A0	IPRM6 ~ IPRM4	高 ↑
	CMI2	105	H'01A4		
系统保留用	—	106	H'01A8		
	—	107	H'01AC		
HCAN*2	ERS0/OVR0、RM1、SLE0、RM0 (mailbox0 收信)	108	H'01B0	IPRM2 ~ IPRM0	低 ↓
		109	H'01B4		
系统保留用	—	111	H'01BC		

【注】*1 表示起始地址的低 16 位。

*2 在 H8S/2280 群中，HCAN 的中断源被保留。

5.6 中断控制模式和中断运行

中断控制器有中断控制模式 0 和中断控制模式 2 两种模式，根据中断控制模式不同，运行也有所不同。通过 SYSCR 选择中断控制模式。中断控制模式 0 和中断控制模式 2 的不同点如表 5.3 所示。

表 5.3 中断控制模式

中断控制模式	中断优先级	中断屏蔽位	说明
0	默认值	I	各中断源的优先级在默认值中已固定。 通过 I 位，屏蔽除 NMI 以外的中断源。
2	IPR	I2~I0	可以通过 IPR，给除 NMI 以外的各中断源设定 8 个优先级。 通过 I2 ~ I0 位，执行 8 个级别的中断屏蔽控制。

5.6.1 中断控制模式 0

在中断控制模式 0 下，通过 CPU 的 CCR 的 I 位，屏蔽除 NMI 以外中断请求。接受中断的流程图如图 5.3 所示。

1. 如果发生将中断使能位置 1 的中断源时，发送中断请求到中断控制器。
2. 将 CPU 的 I 位置为 1 时，仅仅 NMI 中断请求被接受，其它中断请求保持。清除 I 位时，接受中断请求。
3. 有多个中断请求时，中断控制器按照优先级，选择优先级最高的中断请求，并向 CPU 请求中断处理，其他中断请求保持。如果 CPU 接受中断请求，则在处理完当前指令后，开始中断异常处理。
4. 如果 CPU 接受中断请求，则在处理完当前指令后，开始中断异常处理。
5. 通过中断异常处理，将 PC 和 CCR 保存在堆栈区中。PC 中保存有返回后要执行的第一条指令地址。
6. 将 CCR 的 I 位置为 1。以此来屏蔽除 NMI 以外的中断。
7. CPU 生成已接受的中断请求对应的向量地址，并从向量表中读取中断程序开始地址，开始中断处理。

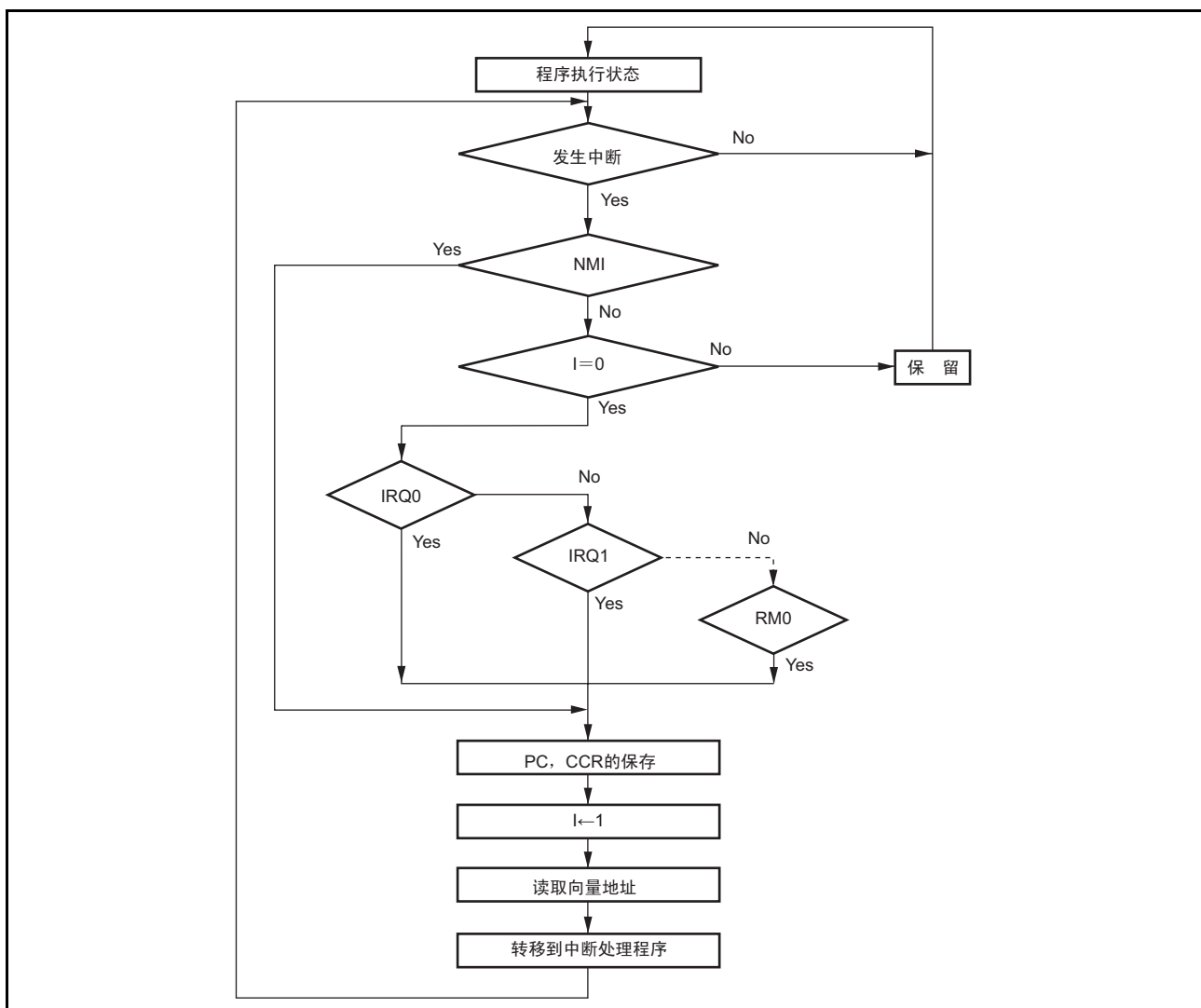


图 5.3 到中断控制模式 0 的中断接受为止的流程图

5.6.2 中断控制模式 2

在中断控制模式 2 中，除 NMI 以外的所有中断请求都是通过比较 IPR 与 CPU 的 EXR 中断屏蔽级（I2~I0 位），来执行 8 级屏蔽控制的。接受中断的流程如图 5.4 所示。

1. 当中断使能位置 1 时，如果相应的中断源发生中断，发送中断请求至中断控制器。
2. 有多个中断请求时，中断控制器按照 IPR 设置的中断优先级，选择优先级最高的中断，保持级别比较低的中断请求。优先级相同时，按照表 5.3 所示的默认优先级，选择中断请求。
3. 下一步，比较已选择的中断请求优先级与 EXR 的中断屏蔽级。如果优先级低于设置的屏蔽级，则保持，如果优先级比中断屏蔽级高，则向 CPU 请求中断处理。
4. 如果 CPU 接受中断请求，则在处理完当前指令后，开始中断异常处理。
5. 通过中断异常处理，将 PC，CCR，及 EXR 保存在堆栈区中。在 PC 中保存有返回后要执行的第一条指令的地址。
6. 清除 EXR 的 T 位。将中断屏蔽级改写为已接受的中断优先级。已接受的中断为 NMI 时，将中断屏蔽级设置为 H'7。
7. CPU 生成已接受中断请求对应的向量地址，并从向量表中读取中断程序开始地址，开始中断处理。

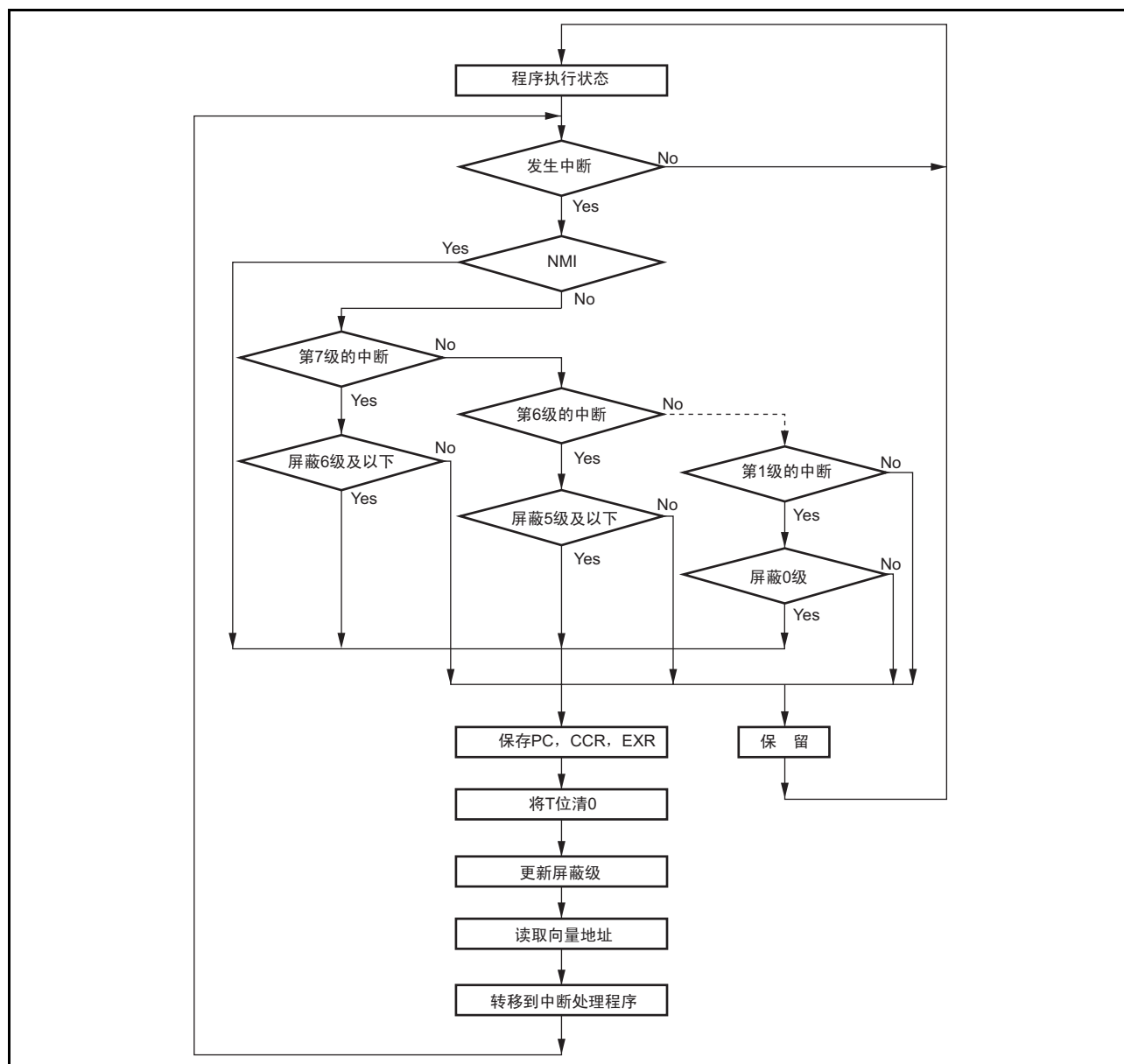


图 5.4 到中断控制模式 2 的中断接受为止的流程图

5.6.3 中断异常处理顺序

中断异常处理的顺序如图 5.5 所示。中断控制模式 0 被设置为该级模式，程序区和堆栈区为内部存储器时的例子。

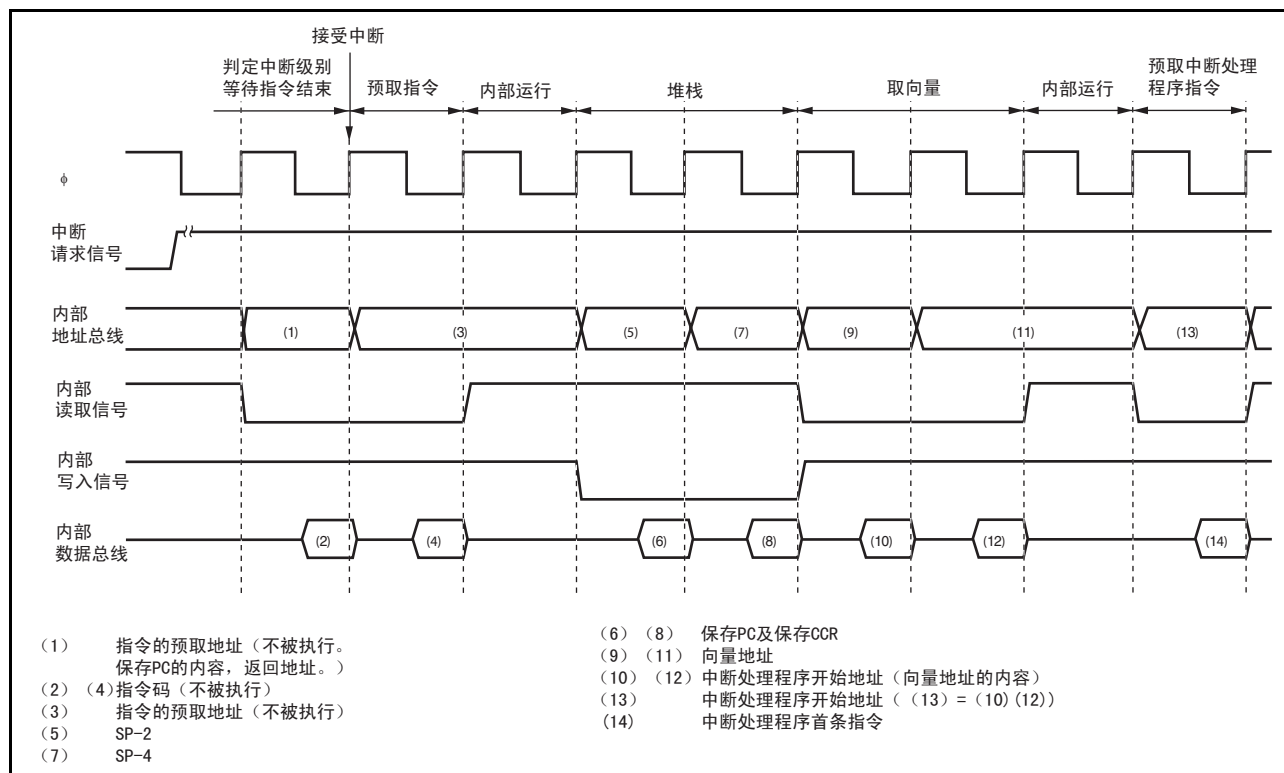


图 5.5 中断异常处理

5.6.4 中断响应时间

从发生中断请求，到执行中断异常处理程序的第 1 条指令为止的中断响应时间如表 5.4 所示。表 5.4 的执行状态符号，请参照表 5.5。因为本 LSI 可对内部存储器进行高速字存取，因此将程序区设为内部 ROM，将堆栈区设为内部 RAM，这样可以提高处理速度。

表 5.4 中断响应时间

No.	执行状态	普通模式 *5		高级模式	
		中断控制模式 0	中断控制模式 2	中断控制模式 0	中断控制模式 2
1	中断优先级判定 *1	3			
2	到结束当前指令为止的等待状态数 *2	1 ~ 19 + 2 · S _I			
3	PC、CCR、和 EXR 的堆栈	2 · S _K	3 · S _K	2 · S _K	3 · S _K
4	取向量	S _I		2 · S _I	
5	取指令 *3	2 · S _I			
6	内部处理 *4	2			
总计（使用内置存储器时）		11~31	12~32	12~32	13~33

【注】 *1 内部中断为 2 个状态周期。

*2 表示 MULXS、DIVXS 指令。

*3 接受中断后的预取和中断处理程序的预取。

*4 接受中断后的内部处理和取向量后的内部处理。

*5 在本 LSI 环境下不能使用。

表 5.5 中断异常处理的执行状态的状态数

符号	存取对象				
	内置存储器	外部器件 *			
		8 位总线		16 位总线	
		2 个状态存取	3 个状态存取	2 个状态存取	3 个状态存取
取指令 S_I	1	4	$6+2m$	2	$3+m$
转移地址列表 S_J					
栈操作 S_K					

【符号说明】

m: 外部器件存取时的等待状态数

【注】 * 在本 LSI 环境下不能使用。

5.7 使用时的注意事项

5.7.1 中断发生与中断屏蔽的竞争

清除中断使能位，并屏蔽中断请求时，中断屏蔽在结束执行指令后变得有效。如果通过 BCL，MOV 等指令清除中断使能位，在指令执行过程中发生中断时，由于在结束执行指令时，该中断为使能状态，因此，结束执行指令后开始中断异常处理。但是，如果一个优先级比较高的中断也提出请求时，那么执行优先级比较高的这个中断，忽视其他中断。这也适用于清除中断源标志。图 5.6 为清除 TPU 的 TIER_0 的 TCIEV 的例子。在屏蔽中断的状态下，如果清除使能位或中断源标志，就不会发生上述的竞争。

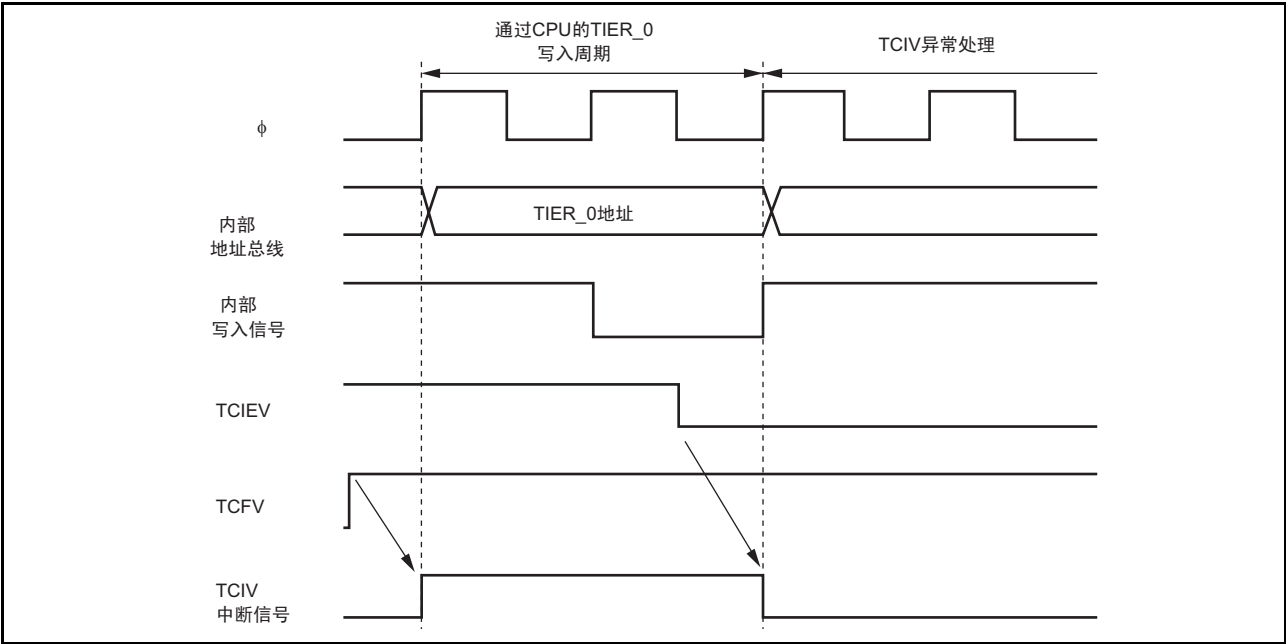


图 5.6 中断发生和中断屏蔽的竞争

5.7.2 禁止中断指令

LDC, ANDC, ORC, XORC 指令执行后, 不立即接受中断请求。结束执行这些指令后, 必须屏蔽包括 NMI 在内的所有中断, 而执行下一条指令。通过这些指令系统 I 位时, 在指令结束执行的 2 个状态以后, 新值变得有效。

5.7.3 中断禁止期间

中断控制器有屏蔽接受中断请求的期间。CPU 通过 LDC, ANDC, ORC, XORC 指令更新屏蔽级后的 3 个状态周期, 中断控制器不接受中断请求。

5.7.4 EEPMOV 指令执行中的中断。

EEPMOV.B 和 EEPMOV.W 指令产生的中断反应不同。

当 EEPMOV.B 指令正在执行一个传送时, 即使有中断请求, 直到传送结束中断才被接受, NMI 也不例外。

当 EEPMOV.W 指令正在执行一个传送时, 如果有中断请求, 则在传送周期的边界开始中断异常处理。这时保存在堆栈中的 PC 值是下一条指令的地址。因此, 在 EEPMOV.W 指令执行过程中发生中断时的程序如下。

```
L1: EEPMOV.W
    MOV.WR4, R4
    BNE    L    1
```

5.7.5 有关 IRQ 中断

$\overline{\text{IRQ}}$ 在时钟运行时, 通过时钟同步, 接受输入; 在软件待机时, 通过异步, 接受输入。输入条件请参照「22.3.2 控制信号时序」。

第 6 章 总线控制器

CPU 以系统时钟 (ϕ) 为标准运行。总线控制器控制存储器周期或总线周期, 根据内部存储器, 内部外围模块不同, 进行的存取也有所不同。另外, 总线控制器还拥有总线仲裁功能, 控制内部总线屏蔽的运行。

6.1 基本运行时序

从 ϕ 的一个上升到下一个上升的 1 个单位称为 1 个周期。存储器周期或总线周期由 1, 2 个或 3, 4 个状态构成, 根据内部存储器和内部外围模块不同, 进行的存取也有所不同。

6.1.1 内部存储器存取时序 (ROM, RAM)

内部存储器的存取执行 1 个状态存取。这时, 数据总线宽度为 16 位, 可以存取字节及字大小。内部存储器存取周期如图 6.1 所示。

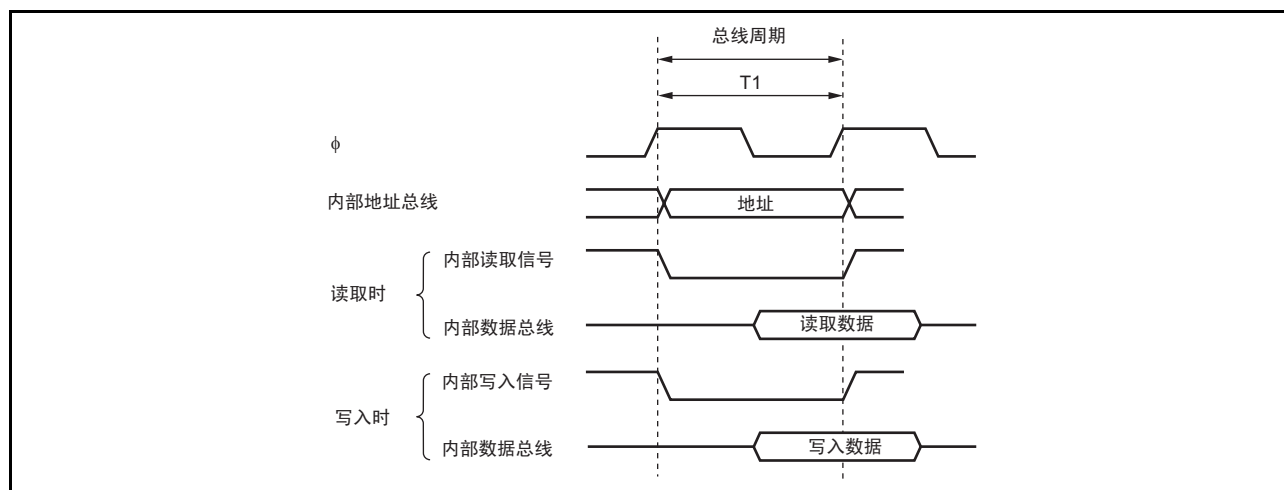


图 6.1 内部存储器存取周期

6.1.2 内部外围模块存取时序

除 HCAN、PWM、LCD、端口 H、J 外，内部外围模块的存取以 2 个状态进行。这时，根据内部 I/O 寄存器，决定数据总线宽度是 8 位还是 16 位。详细情况请参照「第 21 章 寄存器一览表」。内部外围寄存器存取时序如图 6.2 所示。

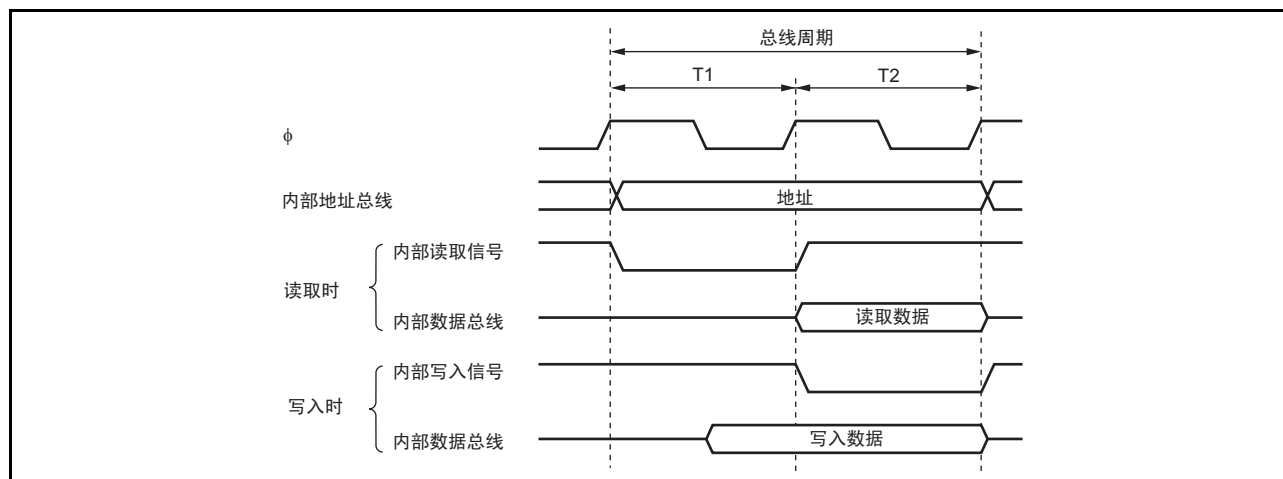


图 6.2 内部外围模块存取周期

6.1.3 内部 HCAN 模块存取时序

内部 HCAN 模块的存取以 4 个状态进行。这时，数据总线宽度为 16 位。另外，通过 HCAN 发出的等待请求，插入等待状态。内部 HCAN 模块存取时序如图 6.3 所示。

【注】 H8S/2280 群没有 HCAN。

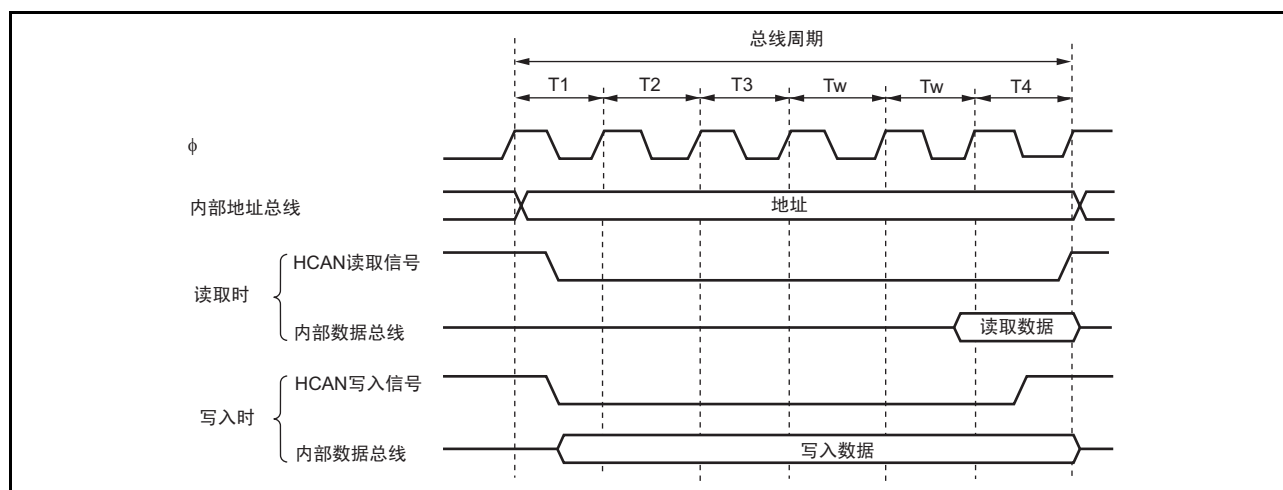


图 6.3 内部 HCAN 模块存取周期（有等待状态）

6.1.4 内部 PWM、LCD、端口 H、J 模块存取时序

内部 PWM、LCD、端口 H、J 模块的存取以 4 个状态进行。这时，数据总线宽为 16 位。PWM、LCD、端口 H、J 模块存取时序如图 6.4 所示。

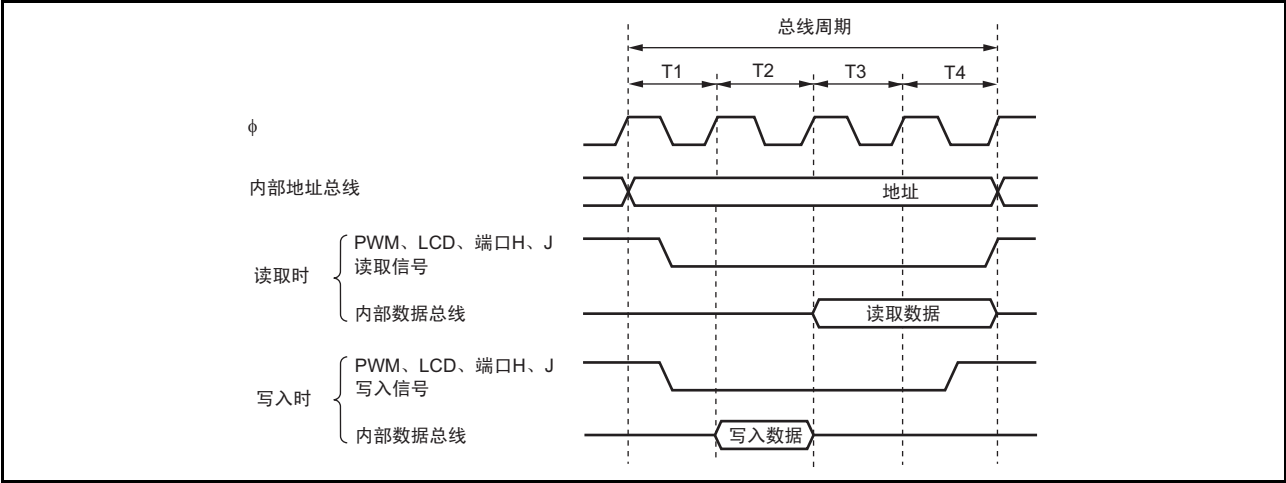


图 6.4 内部 PWM、LCD、端口 H、J 模块存取周期

第 7 章 I/O 端口

H8S/2282 群, H8S/2280 群 (HD64F2280B) 的端口功能如表 7.1 所示。H8S/2280 群 (HD64F2280RB) 的端口功能如表 7.2 所示。每个端口管脚都与连接外围模块的 I/O 管脚或中断输入管脚复用。I/O 端口由控制输入 / 输出数据的方向寄存器 (DDR)、保存输出数据的数据寄存器 (DR) 及读取管脚状态的端口寄存器 (PORT) 组成。输入专用端口没有 DR,DDR。

端口 3、端口 A ~ C 均内置漏极开路输出控制寄存器 (ODR)，能够选择输出缓冲器 PMOS 的 ON/OFF 状态。

所有的端口都能够驱动 1 个 TTL 及 30pF 电容负载。

表 7.1 H8S/2282 群、H8S/2280 群 (HD64F2280B) 的端口功能

端口名	概 要	端口及功能	输入 / 输出及输出类型
端口 1	通用 I/O 端口。也可作为与 TPU_0、TPU_1、TPU_2 输入 / 输出管脚同样功能用，同时可作为中断输入管脚。	P17/TIOCB2/TCLKD	
		P16 /TIOCA2/IRQ1	
		P15 /TIOCB1/TCLKC	
		P14 /TIOCA1/IRQ0	
		P13 /TIOCD0/TCLKB	
		P12 /TIOCC0/TCLKA	
		P11/TIOCB0	
		P10/TIOCA0	
端口 3	通用 I/O 端口。也可作为与 SCI_0、SCI_1 输入输出管脚同样功能用、同时可作为中断输入管脚。	P35/SCK1/IRQ5	推挽 / 漏极开路输出可选。
		P34/RxD1	
		P33/TxD1	
		P32/SCK0/IRQ4	
		P31/RxD0	
		P30/TxD0	
端口 4	通用 I/O 端口。也可作为 A/D 变换的模拟输入管脚。	P47/AN7	
		P46 /AN6	
		P45 /AN5	
		P44 /AN4	
		P43 /AN3	
		P42 /AN2	
		P41 /AN1	
		P40/AN0	

端口名	概 要	端口及功能	输入 / 输出及输出类型
端口 A	通用 I/O 端口。也可作为 LCD 的段和公共输出管脚。	PA7/SEG28	推挽 / 漏极开路输出可选。
		PA6/SEG27	
		PA5/SEG26	
		PA4/SEG25	
		PA3/COM4	
		PA2/COM3	
		PA1/COM2	
		PA0/COM1	
端口 B	通用 I/O 端口。也可作为 LCD 的段和公共输出管脚。	PB7/SEG20	推挽 / 漏极开路输出可选。
		PB6/SEG19	
		PB5/SEG18	
		PB4/SEG17	
		PB3/SEG16	
		PB2/SEG15	
		PB1/SEG14	
		PB0/SEG13	
端口 C	通用 I/O 端口。也可作为 LCD 的段和公共输出管脚。	PC7/SEG12	推挽 / 漏极开路输出可选。
		PC6/SEG11	
		PC5/SEG10	
		PC4/SEG9	
		PC3/SEG8	
		PC2/SEG7	
		PC1/SEG6	
		PC0/SEG5	
端口 D	通用 I/O 端口。也可作为 LCD 的段和公共输出管脚。	PD7/SEG4	
		PD6/SEG3	
		PD5/SEG2	
		PD4/SEG1	

端口名	概 要	端口及功能	输入 / 输出及输出类型
端口 F	通用 I/O 端口。也可作为中断输入管脚，A/D 转换器起始触发输入管脚，LCD 的段输出管脚，以及系统时钟输出管脚。	PF7/ ϕ	
		PF6/SEG24	
		PF5/SEG23	
		PF4/SEG22	
		PF3/ $\overline{\text{ADTRG}}/\overline{\text{IRQ3}}$	
		PF2/SEG21	
		PF1*	
		PF0/ $\overline{\text{IRQ2}}$ *	
端口 H	通用输入 / 输出端口也可作为 PWM_1 输出管脚。	PH7/PWM1H	
		PH6/PWM1G	
		PH5/PWM1F	
		PH4/PWM1E	
		PH3/PWM1D	
		PH2/PWM1C	
		PH1/PWM1B	
		PH0/PWM1A	
端口 J	通用输入 / 输出端口也可作为 PWM_2 输出管脚。	PJ7/PWM2H	
		PJ6/PWM2G	
		PJ5/PWM2F	
		PJ4/PWM2E	
		PJ3/PWM2D	
		PJ2/PWM2C	
		PJ1/PWM2B	
		PJ0/PWM2A	

【注】* H8S/2282 群内不存在此管脚。

表 7.2 H8S/2280 群（HD64F2280RB）的端口功能一览

端口名	概要	端口及功能	输入 / 输出及输出类型
端口 1	通用 I/O 端口。也可作为 TPU_0、TPU_1、TPU_2 输入 / 输出管脚同样功能用同时可作为中断输入管脚。	P17/TIOCB2/TCLKD	
		P16 /TIOCA2/ $\overline{\text{IRQ1}}$	
		P15 /TIOCB1/TCLKC	
		P14 /TIOCA1/ $\overline{\text{IRQ0}}$	
		P13 /TIOCD0/TCLKB	
		P12 /TIOCC0/TCLKA	
		P11/TIOCB0	
		P10/TIOCA0	
端口 3	通用 I/O 端口。也可作为与 SCI_0、SCI_1 输入输出管脚同样功能用、同时可作为中断输入管脚。	P35/SCK1/ $\overline{\text{IRQ5}}$	推挽 / 漏极开路输出可选。
		P34/RxD1	
		P33/TxD1	
		P32/SCK0/ $\overline{\text{IRQ4}}$	
		P31/RxD0	
		P30/TxD0	
端口 4	通用 I/O 端口。也可作为 A/D 变换的模拟输入管脚。	P47/AN7	
		P46 /AN6	
		P45 /AN5	
		P44 /AN4	
		P43 /AN3	
		P42 /AN2	
		P41 /SEG4	
		P40/SEG3	
端口 A	通用 I/O 端口。也可作为 LCD 的段和公共输出管脚。	PA7/SEG32	推挽 / 漏极开路输出可选。
		PA6/SEG31	
		PA5/SEG30	
		PA4/SEG29	
		PA3/COM4	
		PA2/COM3	
		PA1/COM2	
		PA0/COM1	

端口名	概要	端口及功能	输入 / 输出及输出类型
端口 B	通用 I/O 端口。也可作为 LCD 的段和公共输出管脚。	PB7/SEG24	推挽 / 漏极开路输出可选。
		PB6/SEG23	
		PB5/SEG22	
		PB4/SEG21	
		PB3/SEG20	
		PB2/SEG19	
		PB1/SEG18	
		PB0/SEG17	
端口 C	通用 I/O 端口。也可作为 LCD 的段和公共输出管脚。	PC7/SEG16	推挽 / 漏极开路输出可选。
		PC6/SEG15	
		PC5/SEG14	
		PC4/SEG13	
		PC3/SEG12	
		PC2/SEG11	
		PC1/SEG10	
		PC0/SEG9	
端口 D	通用 I/O 端口。也可作为 LCD 的段和公共输出管脚。	PD7/SEG8	
		PD6/SEG7	
		PD5/SEG6	
		PD4/SEG5	
端口 F	通用 I/O 端口。也可作为中断输入管脚，A/D 转换器起始触发输入管脚，LCD 的段输出管脚，以及系统时钟输出管脚。	PF7/ ϕ	
		PF6/SEG28	
		PF5/SEG27	
		PF4/SEG26	
		PF3/ADTRG/ $\overline{\text{IRQ3}}$	
		PF2/SEG25	
		PF1/SEG2	
		PF0/SEG1/ $\overline{\text{IRQ2}}$	

端口名	概要	端口及功能	输入 / 输出及输出类型
端口 H	通用 I/O 端口，可作为 PWM_1 输出管脚。	PH7/PWM1H	
		PH6/PWM1G	
		PH5/PWM1F	
		PH4/PWM1E	
		PH3/PWM1D	
		PH2/PWM1C	
		PH1/PWM1B	
		PH0/PWM1A	
端口 J	通用 I/O 端口，可作为 PWM_2 输出管脚。	PJ7/PWM2H	
		PJ6/PWM2G	
		PJ5/PWM2F	
		PJ4/PWM2E	
		PJ3/PWM2D	
		PJ2/PWM2C	
		PJ1/PWM2B	
		PJ0/PWM2A	

7.1 端口 1

端口 1 是 8 位输入 / 输出端口。端口 1 有以下寄存器：

- 数据方向寄存器（P1DDR）
- 数据寄存器（P1DR）
- 端口寄存器（PORT1）

7.1.1 数据方向寄存器（P1DDR）

P1DDR 的以每一位分别设定端口 1 上各脚的输入 / 输出状态。

位	位名	初始值	R/W	说 明
7	P17DDR	0	W	在选择通用 I/O 端口功能时，将该位置 1，对应的管脚为输出端口，如果清零，则为输入端口。
6	P16DDR	0	W	
5	P15DDR	0	W	
4	P14DDR	0	W	
3	P13DDR	0	W	
2	P12DDR	0	W	
1	P11DDR	0	W	
0	P10DDR	0	W	

7.1.2 数据寄存器（P1DR）

P1DR 用于保存端口 1 的输出数据。

位	位名	初始值	R/W	说 明
7	P17DR	0	R/W	保存被设置为通用 I/O 端口管脚所输出的数据。
6	P16DR	0	R/W	
5	P15DR	0	R/W	
4	P14DR	0	R/W	
3	P13DR	0	R/W	
2	P12DR	0	R/W	
1	P11DR	0	R/W	
0	P10DR	0	R/W	

7.1.3 端口 1 寄存器 (PORT1)

PORT1 表示端口 1 的管脚状态。不可修改 PORT1。

位	位名	初始值	R/W	说 明
7	P17	不定 *	R	当端口 1 读操作时, 如果 P1DDR 位置 1, 则读出 P1DR 的值。如果 P1DDR 位清零, 则读出管脚的状态。
6	P16	不定 *	R	
5	P15	不定 *	R	
4	P14	不定 *	R	
3	P13	不定 *	R	
2	P12	不定 *	R	
1	P11	不定 *	R	
0	P10	不定 *	R	

【注】 * 决定于 P17~P10 管脚的状态。

7.1.4 管脚功能

端口 1 可用作 TPU-0、TPU-1、TPU-2 输入 / 输出管脚、中断输入管脚。寄存器的设定值与管脚功能的关系如下所示。

- P17/TIOCB2/TCLKD

通过 TMDR-2 的 MD3~MD0 位、TIOB-2 的 JOB3~JOB0 位、TCR_2 的 CCLR1、CCLR0 位的搭配来决定 TPU 通道 2 的设定。同时, 通过 TCR_0 的 TPSC2 ~ TPSC0 位、及 P17DDR 位的组合产生以下切换结果。

TPU 通道 2 的设定	下表（1）	下表（2）	
P17DDR	—	0	1
管脚功能	TIOCB2 输出	P17 输入	P17 输出
		TIOCB2 输入 *1	
	TCLKD 输入 *2		

TPU 通道 2 的设定	(2)	(1)	(2)	(2)	(1)	(2)
MD3 ~ MD0	B'0000、B'01xx		B'0010	B'0011		
IOB3 ~ IOB0	B'0000 B'0100 B'1xxx	B'0001 ~ B'0011 B'0101 ~ B'0111	—	B'xx00	B'xx00 以外	
CCLR1、CCLR0	—	—	—	—	B'10 以外	B'10
输出功能	—	比较输出	—	—	PWM 模式 2 输出	—

【符号说明】 x: Don't Care

【注】 *1 MD3 ~ MD0=B'0000、B'01xx 且 IOB3=1 时, 成为 TIOCB2 输入状态。

*2 TCR-0 的 TPSC2~TPSC0=B'111 时, 成为 TCLKD 输入状态。并且, 通道 2 设定为相位计数模式时, 成为 TCLKD 输入状态。

• P16/TIOCA2/ $\overline{\text{IRQ1}}$

通过 TMDR_2 的 MD3 ~ MD0 位、TIOA2 的 IOA3 ~ IOA0 位、TCR_2 的 CCLR1、CCLR0 位的搭配来决定 TPU 通道的设定，以及通过 P16DDR 位的组合产生以下切换结果。

TPU 通道 2 的设定	下表（1）	下表（2）	
P16DDR	—	0	1
管脚功能	TIOCA2 输出	P16 输入	P16 输出
		TIOCA2 输入 *1	
	$\overline{\text{IRQ1}}$ 输入		

TPU 通道 2 的设定	(2)	(1)	(2)	(1)	(1)	(2)
MD3 ~ MD0	B'0000、B'01xx		B'001x	B'0011		
IOA3 ~ IOA0	B'0000 B'0100 B'1xxx	B'0001 ~ B'0011 B'0101 ~ B'0111	B'xx00	B'xx00 以外	B'xx00 以外	
CCLR1、CCLR0	—	—	—	—	B'01 以外	B'01
输出功能	—	比较输出	—	PWM 模式 1 输出 *2	PWM 模式 2 输出	—

【符号说明】x : Don't care

【注】 *1 MD3 ~ MD0 = B'0000、B'01xx 且 IOA3 = 1 时，成为 TIOCA2 输入状态。

*2 TIOCB2 成为禁止输出状态。

- P15/TIOCB1/TCLKC

通过 TMDR_1 的 MD3 ~ MD0 位、TIOB_1 的 IOB3 ~ IOB0 位、TCR_1 的 CCLR1、CCLR0 位的搭配决定 TPU 通道 1 的设定。通过 TCR_0、TCR_2 的 TPSC2 ~ TPSC0 位及 P15DDR 位的组合产生以下切换结果。

TPU 通道 1 的设置	下表 (1)	下表 (2)	
P15DDR	—	0	1
管脚功能	TIOCB1 输出	P15 输入	P15 输出
		TIOCB1 输入 *1	
	TCLKC 输入 *2		

TPU 通道 1 的设定	(2)	(1)	(2)	(2)	(1)	(2)
MD3 ~ MD0	B'0000、B'01xx		B'0010	B'0011		
IOB3 ~ IOB0	B'0000 B'0100 B'1xxx	B'0001 ~ B'0011 B'0101 ~ B'0111	—	B'xx00	B'xx00 以外	
CCLR1、CCLR0	—	—	—	—	B'10 以外	B'10
输出功能	—	比较输出	—	—	PWM 模式 2 输出	—

【符号说明】 x: Don't Care

【注】 *1 MD3 ~ MD0 = B'0000、B'01xx 且 IOB3 ~ IOB0 = B'10 时，成为 TIOCB1 输入状态。

*2 TCR_0、TCR_2 的任一个设定在 TPSC2 ~ TPSC0 = B'110 时，成为 TCLKC 输入。并且、通道 2 在设定相位计数模式后，成为 TCLKC 输入。

- P14/TIOCA1/ $\overline{\text{IRQ0}}$

通过 TMDR_1 的 MD3 ~ MD0 位、TIOA1 的 IOA3 ~ IOA0 位、TCR_1 的 CCLR1、CCLR0 位的搭配决定 TPU 通道 1 的设定，以及通过 P14DDR 位的组合产生以下切换结果。

TPU 通道 1 的设定	下表 (1)	下表 (2)	
P14DDR	—	0	1
管脚功能	TIOCA1 输出	P14 输入	P14 输出
		TIOCA1 输出 *1	
	$\overline{\text{IRQ0}}$ 输入		

TPU 通道 1 的设定	(2)	(1)	(2)	(1)	(1)	(2)
MD3 ~ MD0	B'0000、B'01xx		B'001x	B'0010	B'0011	
IOA3 ~ IOA0	B'0000 B'0100 B'1xxx	B'0001 ~ B'0011 B'0101 ~ B'0111	B'xx00	B'xx00 以外	B'xx00 以外	
CCLR1、CCLR0	—	—	—	—	B'01 以外	B'01
输出功能	—	比较输出	—	PWM 模式 1 输出 *2	PWM 模式 2 输出	—

【符号说明】 x: Don't Care

【注】 *1 MD3 ~ MD0=B'0000、B'01xx 且 IOA3 ~ IOA0=B'10xx 时，成为 TIOCA1 输入状态。

*2 TIOCB1 成为禁止输出状态。

• P13/ TIOCD0/TCLKB

通过 TMDR_0 的 MD3 ~ MD0 位、TIO RL_0 的 IOD3 ~ IOD0 位、TCR_0 的 CCLR2 ~ CCLR0 位的搭配决定 TPU 通道 0 的设定，通过 TCR_0 ~ TCR_2 的 TPSC2 ~ TPSC0 位及 P13DDR 位的组合产生以下切换结果。

TPU 通道 0 的设定	下表（1）	下表（2）	
P13DDR	—	0	1
管脚功能	TIOCD0 输出	P13 输入	P13 输出
		TIOCD0 输入 *1	
	TCLKB 输入 *2		

TPU 通道 0 的设定	(2)	(1)	(2)	(2)	(1)	(2)
MD3 ~ MD0	B'0000		B'0010	B'0011		
IOD3 ~ IOD0	B'0000 B'0100 B'1xxx	B'0001 ~ B'0011 B'0101 ~ B'0111	—	B'xx00	B'xx00 以外	
CCLR1 ~ CCLR0	—	—	—	—	B'110 以外	B'110
输出功能	—	比较输出	—	—	PWM 模式 2 输出	—

【符号说明】 x: Don't Care

【注】 *1 MD3 ~ MD0=B'0000、且 IOD3 ~ IOD0=B'10xx 时，成为 TIOCD0 输入状态。

*2 TCR_0 ~ TCR_2 的任一个设定在 TPSC2 ~ TPSC0=B'101 时，成为 TCLKB 输入状态。并且，通道 1 设定为相位计数模式时，成为 TCLKB 输入状态。

• P12/TIOCC0/TCLKA

通过 TMDR_0 的 MD3 ~ MD0 位、TIORL_0 的 IOC3 ~ IOC0 位、TCR_0 的 CCLR2 ~ CCLR0 位的搭配决定 TPU 通道 0 设定，通过 TCR_0 ~ TCR_2 的 TPSC2 ~ TPSC0 位及 P12DDR 位的组合产生以下切换结果。

TPU 通道 0 的设定	下表（1）	下表（2）	
P12DDR	—	0	1
管脚功能	TIOCC0 输出	P12 输入	P12 输出
		TIOCC0 输入 *1	
	TCLKA 输入 *2		

TPU 通道 0 的设定	(2)	(1)	(2)	(1)	(1)	(2)
MD3 ~ MD0	B'0000		B'001x	B'0010	B'0011	
IOC3 ~ IOC0	B'0000 B'0100 B'1xxx	B'0001 ~ B'0011 B'0101 ~ B'0111	B'xx00	B'xx00 以外	B'xx00 以外	
CCLR2 ~ CCLR0	—	—	—	—	B'101 以外	B'101
输出功能	—	比较输出	—	PWM 模式 1 输出 *3	PWM 模式 2 输出	—

【符号说明】 x: Don't Care

- 【注】 *1 MD3 ~ MD0=B'0000、且 IOC3 ~ IOC0=B'10xx 时，成为 TIOCC0 输入状态。
- *2 TCR_0 ~ TCR_2 的任一个设定在 TPSC2 ~ TPSC0=B'100 时，成为 TCLKA 输入状态。并且，通道 1 设定为相位计数模式时，成为 TCLKA 输入状态。
- *3 TIOCD0 成为禁止输出状态。TMDR_0 的 BFA=1 且 BFB=1 时，禁止输出，成为 (2) 的设定。

- P11/TIOCB0

通过 TMDR_0 的 MD3 ~ MD0 位、TIO RH_0 的 IOB3 ~ IOB0 位、TCR_0 的 CCLR2 ~ CCLR0 位的搭配决定 TPU 通道 0 的设定，以及通过 P11DDR 位的组合产生以下切换结果。

TPU 通道 0 的设定	下表 (1)	下表 (2)	
P11DDR	—	0	1
管脚功能	TIOCB0 输出	P11 输入	P11 输出
		TIOCB0 输入 *	

TPU 通道 0 的设定	(2)	(1)	(2)	(2)	(1)	(2)
MD3 ~ MD0	B'0000		B'0010	B'0011		
IOB3 ~ IOB0	B'0000 B'0100 B'1xxx	B'0001 ~ B'0011 B'0101 ~ B'0111	—	B'xx00	B'xx00 以外	
CCLR2 ~ CCLR0	—	—	—	—	B'010 以外	B'010
输出功能	—	比较输出	—	—	PWM 模式 2 输出	—

【符号说明】 x: Don't Care

【注】 * MD3 ~ MD0 = B'0000、且 IOB3 ~ IOB0 = B'10xx 时，成为 TIOCB0 输入状态。

- P10/TIOCA0

通过 TMDR_0 的 MD3 ~ MD0 位、TIO RH_0 的 IOA3 ~ IOA0 位、TCR_0 的 CCLR2 ~ CCLR0 位的搭配决定 TPU 通道 0 的设定，以及通过 P10DDR 位的组合产生以下的切换结果。

TPU 通道 0 的设定	下表 (1)	下表 (2)	
P10DDR	—	0	1
管脚功能	TIOCA0 输出	P10 输入	P10 输出
		TIOCA0 输入 *1	

TPU 通道 0 的设定	(2)	(1)	(2)	(1)	(1)	(2)
MD3 ~ MD0	B'0000		B'001x	B'0010	B'0011	
IOA3 ~ IOA0	B'0000 B'0100 B'1xxx	B'0001 ~ B'0011 B'0101 ~ B'0111	B'xx00	B'xx00 以外	B'xx00 以外	
CCLR2 ~ CCLR0	—	—	—	—	B'001 以外	B'001
输出功能	—	比较输出	—	PWM 模式 1 输出 *2	PWM 模式 2 输出	—

【符号说明】 x: Don't Care

【注】 *1 MD3 ~ MD0 = B'0000、且 IOA3 ~ IOA0 = B'10xx 时，成为 TIOCA0 输入状态。

*2 TIOCB0 成为禁止输出状态。

7.2 端口 3

端口 3 是 6 位输入 / 输出通用 I/O 端口，也可作其它功能使用。端口 3 有以下寄存器。

- 数据方向寄存器（P3DDR）
- 数据寄存器（P3DR）
- 端口寄存器（PORT3）
- 漏极开路控制寄存器（P3ODR）

7.2.1 数据方向寄存器（P3DDR）

P3DDR 的每一位分别设定端口 3 上各脚的输入 / 输出状态。

位	位名	初始值	R/W	说 明
7、6	—	不定	—	保留位 不定值。
5	P35DDR	0	W	在选择通用输入 / 输出端口功能时，将该位置 1，对应的管脚为输出端口；如果清零，则为输入端口。
4	P34DDR	0	W	
3	P33DDR	0	W	
2	P32DDR	0	W	
1	P31DDR	0	W	
0	P30DDR	0	W	

7.2.2 数据寄存器（P3DR）

P3DR 保存端口 3 的输出数据。

位	位名	初始值	R/W	说 明
7、6	—	不定	—	保留位 不定值。
5	P35DDR	0	R/W	保存被设置为通用输入 / 输出端口管脚后输出的数据。
4	P34DDR	0	R/W	
3	P33DDR	0	R/W	
2	P32DDR	0	R/W	
1	P31DDR	0	R/W	
0	P30DDR	0	R/W	

7.2.3 端口 3 寄存器 (PORT3)

PORT3 表示端口 3 的管脚状态。不可修改 PORT3。

位	位名	初始值	R/W	说 明
7、6	—	不定	—	保留位 不定值。
5	P35	不定	R	当端口 3 读操作时，如果 P3DDR 位置 1，则读出 P3DR 的值。如果 P3DDR 位清零，则读出管脚的状态。
4	P34	不定	R	
3	P33	不定	R	
2	P32	不定	R	
1	P31	不定	R	
0	P30	不定	R	

【注】 * 决定于 P35 ~ P30 管脚的状态。

7.2.4 漏极开路控制寄存器 (P3ODR)

P3ODR 选择端口 3 的输出类型。

位	位名	初始值	R/W	说 明
7、6	—	不定	—	保留位 不定值。
5	P35ODR	0	R/W	如果置 1 后，对应管脚 PMOS 通常是 off 状态，如果该脚设置成输出，则该管脚为漏极开路输出。如果清零，则该管脚为推挽输出。
4	P34ODR	0	R/W	
3	P33ODR	0	R/W	
2	P32ODR	0	R/W	
1	P31ODR	0	R/W	
0	P30ODR	0	R/W	

7.2.5 管脚功能

端口 3 也用作 SCI_0、SCI_1 输入 / 输出管脚和中断输入管脚。寄存器的设定值与管脚的关系如下所示。

- P35/SCK1/ $\overline{\text{IRQ5}}$

通过 SCI_1 的 SMR 的 $\overline{\text{C/A}}$ 位、SCR 的 CKE0、CKE1 位、及 P35DDR 位的组合产生以下切换结果。

CKE1	0				1
$\overline{\text{C/A}}$	0			1	—
CKE0	0		1	—	—
P35DDR	0	1	—	—	—
管脚功能	P35 输入	P35 输出	SCK1 输出	SCK1 输出	SCK1 输入
	$\overline{\text{IRQ5}}$ 输入				

- P34/RxD1

通过 SCI_1 的 SCR 的 RE 位和 P34DDR 位的组合产生以下切换结果。

RE	0		1
P34DDR	0	1	—
管脚功能	P34 输入	P34 输出	TxD1 输入

- P33/TxD1

通过 SCI_1 的 SCR 的 TE 位和 P33DDR 位的组合产生以下切换结果。

TE	0		1
P33DDR	0	1	—
管脚功能	P33 输入	P33 输出	RxD1 输入

- P32/SCK0/ $\overline{\text{IRQ4}}$

通过 SCI_0 的 SMR 的 $\overline{\text{C/A}}$ 位、SCR 的 CKE0、CKE1 位、及 P32DDR 位的组合产生以下切换结果。

CKE1	0				1
$\overline{\text{C/A}}$	0			1	—
CKE0	0		1	—	—
P32DDR	0	1	—	—	—
管脚功能	P32 输入	P32 输出	SCK0 输出	SCK0 输出	SCK0 输入
	$\overline{\text{IRQ4}}$ 输入				

- P31/RxD0

通过 SCI_0 的 SCR 的 RE 位和 P31DDR 位的组合产生以下切换结果。

RE	0		1
P31DDR	0	1	—
管脚功能	P31 输入	P31 输出	RxD0 输入

- P30/TxD0

通过 SCI_0 的 SCR 的 TE 位和 P30DDR 位的组合产生以下切换结果。

TE	0		1
P30DDR	0	1	—
管脚功能	P30 输入	P30 输出	TxD0 输入

7.3 端口 4

端口 4 是 8 位的模拟输入和 LCD 的段输出管脚*。端口 4 有以下寄存器。

- 端口寄存器 (PORT4)

【注】 * 仅 H8S/2280 群 (HD64F2280RB)。

7.3.1 端口 4 寄存器 (PORT4)

PORT4 表示端口 4 的管脚状态，不可修改 PORT4。

位	位名	初始值	R/W	说 明
7	P47	不定 *	R	读此寄存器后，通常读出管脚的状态。
6	P46	不定 *	R	
5	P45	不定 *	R	
4	P44	不定 *	R	
3	P43	不定 *	R	
2	P42	不定 *	R	
1	P41	不定 *	R	
0	P40	不定 *	R	

【注】 * 决定于 P47 ~ P40 管脚的状态。

7.3.2 端口功能

(1) H8S/2282 群、H8S/2280 群 (HD64F2280B)

端口 4 是作为模拟输入管脚的输入端口。

(2) H8S/2280 群 (HD64F2280RB)

端口 4 既可作为模拟输入管脚也可用作 LCD 段输出管脚。寄存器的设定值与管脚功能的关系如下所示。

- P47/AN7、P46/AN6、P45/AN5、P44/AN4、P43/AN3、P42/AN2
用作模拟输入管脚的输入端口
- P41/SEG4、P40/SEG3

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位产生以下的切换结果。

SGS3 ~ SGS0	0111 以外	0111
管脚功能	P41、P40 输入	SEG4、SEG3 输出

7.4 端口 A

端口 A 是 8 位的 I/O 端口，也可作其它功能用。端口 A 有以下寄存器。

- 数据方向寄存器（PADDDR）
- 数据寄存器（PADR）
- 端口寄存器（PORTA）
- 漏极开路控制寄存器（PAODR）

7.4.1 数据方向寄存器（PADDDR）

PADDDR 的每一位分别设定端口 A 上各脚的输入 / 输出状态。

位	位名	初始值	R/W	说 明
7	PA7DDR	0	W	在选择通用输入 / 输出端口的功能后，将此位置 1，对应管脚成为输出端口，如果清零，则为输入端口。
6	PA6DDR	0	W	
5	PA5DDR	0	W	
4	PA4DDR	0	W	
3	PA3DDR	0	W	
2	PA2DDR	0	W	
1	PA1DDR	0	W	
0	PA0DDR	0	W	

7.4.2 端口 A 数据寄存器（PADR）

PADR 保存端口 A 的输出数据。

位	位名	初始值	R/W	说 明
7	PA7DR	0	R/W	保存被设置为通用输出端口的管脚后输出的数据。
6	PA6DR	0	R/W	
5	PA5DR	0	R/W	
4	PA4DR	0	R/W	
3	PA3DR	0	R/W	
2	PA2DR	0	R/W	
1	PA1DR	0	R/W	
0	PA0DR	0	R/W	

7.4.3 端口 A 寄存器（PORTA）

PORTA 表示端口 A 的管脚状态，不可修改 PORTA。

位	位名	初始值	R/W	说 明
7	PA7	不定 *	R	此寄存器在读操作时，如果 PADDR 位置 1，则读出 PADDR 的值。如果 PADDR 位清零，则读出管脚的状态。
6	PA6	不定 *	R	
5	PA5	不定 *	R	
4	PA4	不定 *	R	
3	PA3	不定 *	R	
2	PA2	不定 *	R	
1	PA1	不定 *	R	
0	PA0	不定 *	R	

【注】 * 决定于 PA7 ~ PA0 管脚的状态。

7.4.4 漏极开路控制寄存器（PAODR）

PAODR 选择端口 A 的输出类型。

位	位名	初始值	R/W	说 明
7	PA7ODR	0	R/W	设置为 1 后，对应管脚的 PMOS 通常是 off 状态，如果相应管脚设定为输出，则为漏极开路输出。如果清零，则相应管脚为推挽输出。
6	PA6ODR	0	R/W	
5	PA5ODR	0	R/W	
4	PA4ODR	0	R/W	
3	PA3ODR	0	R/W	
2	PA2ODR	0	R/W	
1	PA1ODR	0	R/W	
0	PA0ODR	0	R/W	

7.4.5 H8S/2282 群、H8S/2280 群（HD64F2280B）的管脚功能

端口 A 可用作 LCD 的段输出管脚和 LCD 的公共输出管脚。寄存器的设定值与管脚功能的关系如下所示。

- PA7/SEG28、PA6/SEG27、PA5/SEG26、PA4/SEG25

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PAnDDR 位的组合产生以下切换结果。

SGS3 ~ SGS0	0000		0000 以外
PAnDDR	0	1	—
管脚功能	PA7 ~ PA4 输入	PA7 ~ PA4 输出	SEG28 ~ SEG25 输出

(n = 7 ~ 4)

- PA3/COM4、PA2/COM3、PA1/COM2、PA0/COM1

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PAnDDR 位的组合产生以下切换结果。

SGS3 ~ SGS0	0000		0000 以外
PAnDDR	0	1	—
管脚功能	PA3 ~ PA0 输入	PA3 ~ PA0 输出	COM4 ~ COM1 输出

(n = 3 ~ 0)

7.4.6 H8S/2280 群（HD64F2280RB）的管脚功能

管脚 A 兼用 LCD 的段输出管脚、公共输出管脚。寄存器的设定值与管脚功能的关系如下所示。

- PA7/SEG32、PA6/SEG31、PA5/SEG30、PA4/SEG29

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PAnDDR 位的组合产生以下切换结果。

SGS3 ~ SGS0	0000		0000 以外
PAnDDR	0	1	—
管脚功能	PA7 ~ PA4 输入	PA7 ~ PA4 输出	SEG32 ~ SEG29 输出

(n = 7 ~ 4)

- PA3/COM4、PA2/COM3、PA1/COM2、PA0/COM1

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PAnDDR 位的组合产生以下切换结果。

SGS3 ~ SGS0	0000		0000 以外
PAnDDR	0	1	—
管脚功能	PA3 ~ PA0 输入	PA3 ~ PA0 输出	COM4 ~ COM1 输出

(n = 3 ~ 0)

7.5 端口 B

端口 B 是 8 位的 I/O 端口，也可作其它功能用。端口 B 有以下寄存器。

- 数据方向寄存器（PBDDR）
- 数据寄存器（PBDR）
- 端口寄存器（PORTB）
- 漏极开路控制寄存器。（PBODR）

7.5.1 端口 B 数据方向寄存器（PBDDR）

PBDDR 的每一位分别设定端口 B 上各脚的输入 / 输出状态。

位	位名	初始值	R/W	说 明
7	PB7DDR	0	W	在选择通用输入 / 输出端口的功能后，将此位置 1，对应管脚成为输出端口，如果清零，则为输入端口。
6	PB6DDR	0	W	
5	PB5DDR	0	W	
4	PB4DDR	0	W	
3	PB3DDR	0	W	
2	PB2DDR	0	W	
1	PB1DDR	0	W	
0	PB0DDR	0	W	

7.5.2 端口 B 数据寄存器（PBDR）

PBDR 保存端口 B 的输出数据。

位	位名	初始值	R/W	说 明
7	PB7DR	0	R/W	保存被设置为通用输出端口的管脚后输出的数据。
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	PB0DR	0	R/W	

7.5.3 端口 B 寄存器（PORTB）

PORTB 表示端口 B 的管脚状态，不可修改 PORTB。

位	位名	初始值	R/W	说 明
7	PB7	不定 *	R	此寄存器在读操作时，如果 PBDDR 位置 1，则读出 PBDR 的值。如果 PBDDR 位清零，则读出管脚的状态。
6	PB6	不定 *	R	
5	PB5	不定 *	R	
4	PB4	不定 *	R	
3	PB3	不定 *	R	
2	PB2	不定 *	R	
1	PB1	不定 *	R	
0	PB0	不定 *	R	

【注】 * 决定于 PB7 ~ PB0 管脚的状态。

7.5.4 端口 B 漏极开路控制寄存器（PBODR）

PBODR 选择端口 B 的输出类型。

位	位名	初始值	R/W	说 明
7	PB7ODR	0	R/W	如果置 1，对应管脚的 PMOS 通常是 off 状态，如果相应管脚设置成输出后，则该管脚为漏极开路输出。如果清零，则该管脚为推挽输出。
6	PB6ODR	0	R/W	
5	PB5ODR	0	R/W	
4	PB4ODR	0	R/W	
3	PB3ODR	0	R/W	
2	PB2ODR	0	R/W	
1	PB1ODR	0	R/W	
0	PB0ODR	0	R/W	

7.5.5 H8S/2282 群、H8S/2280 群（HD64F2280B）的管脚功能

端口 B 可用作 LCD 的段输出管脚。寄存器的设定值与管脚功能的关系如下所示。

- PB7/SEG20、PB6/SEG19、PB5/SEG18、PB4/SEG17

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PBnDDR 位的组合产生以下切换结果。

SGS3 ~ SGS0	0000、0001		0000、0001 以外
PBnDDR	0	1	—
管脚功能	PB7 ~ PB4 输入	PB7 ~ PB4 输出	SEG20 ~ SEG17 输出

(n = 7 ~ 4)

- PB3/SEG16、PB2/SEG15、PB1/SEG14、PB0/SEG13

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PBnDDR 位的组合产生以下切换结果。

SGS3 ~ SGS0	0000 ~ 0010		0000 ~ 0010 以外
PBnDDR	0	1	—
管脚功能	PB3 ~ PB0 输入	PB3 ~ PB0 输出	SEG16 ~ SEG13 输出

(n = 3 ~ 0)

7.5.6 H8S/2280 群（HD64F2280RB）的管脚功能

端口 B 可用作 LCD 的段输出管脚。寄存器的设定值与管脚功能的关系如下所示。

- PB7/SEG24、PB6/SEG23、PB5/SEG22、PB4/SEG21

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PBnDDR 位的组合产生以下切换结果。

SGS3 ~ SGS0	0000、0001		0000、0001 以外
PBnDDR	0	1	—
管脚功能	PB7 ~ PB4 输入	PB7 ~ PB4 输出	SEG24 ~ SEG21 输出

(n = 7 ~ 4)

- PB3/SEG20、PB2/SEG19、PB1/SEG18、PB0/SEG17

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PBnDDR 位的组合产生以下切换结果。

SGS3 ~ SGS0	0000 ~ 0010		0000 ~ 0010 以外
PBnDDR	0	1	—
管脚功能	PB3 ~ PB0 输入	PB3 ~ PB0 输出	SEG20 ~ SEG17 输出

(n = 3 ~ 0)

7.6 端口 C

端口 C 是 8 位的 I/O 端口，端口 C 有以下寄存器。

- 数据方向寄存器（PCDDR）
- 数据寄存器（PCDR）
- 端口寄存器（PORTC）
- 漏极开路控制寄存器。（PCODR）

7.6.1 端口 C 数据方向寄存器（PCDDR）

PCDDR 的每一位分别设定端口 C 上各脚的输入 / 输出状态。

位	位名	初始值	R/W	说 明
7	PC7DDR	0	W	在选择通用输入 / 输出端口的功能后，将该位置 1，对应管脚成为输出端口，如果清零后，成为输入端口。
6	PC6DDR	0	W	
5	PC5DDR	0	W	
4	PC4DDR	0	W	
3	PC3DDR	0	W	
2	PC2DDR	0	W	
1	PC1DDR	0	W	
0	PC0DDR	0	W	

7.6.2 端口 C 数据寄存器（PCDR）

PCDR 保存端口 C 的输出数据。

位	位名	初始值	R/W	说 明
7	PC7DR	0	R/W	保存被设置为通用输出端口使用的管脚后输出的数据。
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	

7.6.3 端口 C 寄存器 (PORTC)

PORTC 表示端口 C 的管脚状态。不可修改 PORTC。

位	位名	初始值	R/W	说 明
7	PC7	不定 *	R	此寄存器在读操作时，如果 PCDDR 位置 1，则读出 PCDR 的值。如果 PCDDR 位清零，则读出管脚的状态。
6	PC6	不定 *	R	
5	PC5	不定 *	R	
4	PC4	不定 *	R	
3	PC3	不定 *	R	
2	PC2	不定 *	R	
1	PC1	不定 *	R	
0	PC0	不定 *	R	

【注】 * 决定于 PC7 ~ PC0 管脚的状态。

7.6.4 端口 C 漏极开路控制寄存器。(PCODR)

PCODR 选择端口 C 的输出类型。

位	位名	初始值	R/W	说 明
7	PC7ODR	0	R/W	如果置 1，对应管脚的 PMOS 通常是 off 状态，如果相应管脚设定为输出，则为漏极开路输出。如果清零，则该管脚为推挽输出。
6	PC6ODR	0	R/W	
5	PC5ODR	0	R/W	
4	PC4ODR	0	R/W	
3	PC3ODR	0	R/W	
2	PC2ODR	0	R/W	
1	PC1ODR	0	R/W	
0	PC0ODR	0	R/W	

7.6.5 H8S/2282 群、H8S/2280 群（HD64F2280B）的管脚功能

端口 C 可用作 LCD 的段输出管脚。寄存器的设定值与管脚功能的关系如下所示。

- PC7/SEG12、PC6/SEG11、PC5/SEG10、PC4/SEG9

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PCnDDR 位的组合产生以下切换。

SGS3 ~ SGS0	0000 ~ 0011		0000 ~ 0011 以外
PCnDDR	0	1	—
管脚功能	PC7 ~ PC4 输入	PC7 ~ PC4 输出	SEG12 ~ SEG9 输出

(n = 7 ~ 4)

- PC3/SEG8、PC2/SEG7、PC1/SEG6、PC0/SEG5

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PCnDDR 位的组合产生以下切换。

SGS3 ~ SGS0	0000 ~ 0100		0000 ~ 0100 以外
PCnDDR	0	1	—
管脚功能	PC3 ~ PC0 输入	PC3 ~ PC0 输出	SEG8 ~ SEG5 输出

(n = 3 ~ 0)

7.6.6 H8S/2280 群（HD64F2280RB）的管脚功能

端口 C 兼用 LCD 的段输出管脚。寄存器的设定值与管脚功能的关系如下所示。

- PC7/SEG16、PC6/SEG15、PC5/SEG14、PC4/SEG13

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PCnDDR 位的组合产生以下切换。

SGS3 ~ SGS0	0000 ~ 0011		0000 ~ 0011 以外
PCnDDR	0	1	—
管脚功能	PC7 ~ PC4 输入	PC7 ~ PC4 输出	SEG16 ~ SEG13 输出

(n = 7 ~ 4)

- PC3/SEG12、PC2/SEG11、PC1/SEG10、PC0/SEG9

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PCnDDR 位的组合产生以下切换。

SGS3 ~ SGS0	0000 ~ 0100		0000 ~ 0100 以外
PCnDDR	0	1	—
管脚功能	PC3 ~ PC0 输入	PC3 ~ PC0 输出	SEG12 ~ SEG9 输出

(n = 3 ~ 0)

7.7 端口 D

端口 D 是 4 位的 I/O 端口，端口 D 有以下寄存器。

- 数据方向寄存器（PDDDR）
- 数据寄存器（PDDR）
- 端口寄存器（PORTD）

7.7.1 端口 D 数据方向寄存器（PDDDR）

PDDDR 的每一位分别设定端口 D 上各管脚的输入 / 输出状态。

位	位名	初始值	R/W	说 明
7	PD7DDR	0	W	在选择通用输入 / 输出端口的功能后，将该位置 1，对应管脚成为输出端口，如果清零后，成为输入端口。
6	PD6DDR	0	W	
5	PD5DDR	0	W	
4	PD4DDR	0	W	
3 ~ 0	—	不定	—	保留位 读时，将返回不定值。

7.7.2 端口 D 数据寄存器 (PDDR)

PDDR 保存端口 D 的输出数据。

位	位名	初始值	R/W	说 明
7	PD7DR	0	R/W	保存被设置为通用输出端口使用的管脚后输出的数据。
6	PD6DR	0	R/W	
5	PD5DR	0	R/W	
4	PD4DR	0	R/W	
3 ~ 0	—	不定	—	保留位 读时, 将返回不定值。

7.7.3 端口 D 寄存器 (PORTD)

PORTD 表示端口 D 的管脚的状态。不可修改 PORTD。

位	位名	初始值	R/W	说 明
7	PD7	不定 *	R	此寄存器在读操作时, 如果 PDDDR 位置 1, 则读出 PDDR 的值。如果 PDDDR 位清零, 则读出管脚的状态。
6	PD6	不定 *	R	
5	PD5	不定 *	R	
4	PD4	不定 *	R	
3 ~ 0	—	不定	—	不定值。

【注】* 决定于 PD7 ~ PD4 管脚的状态。

7.7.4 H8S/2282 群、H8S/2280 群 (HD64F2280B) 的管脚功能

端口 D 可用作 LCD 的段的输出管脚。寄存器的设定值与管脚功能的关系如下所示。

- PD7/SEG4、PD6/SEG3、PD5/SEG2、PD4/SEG1

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PCnDDR 位的组合产生以下切换。

SGS3 ~ SGS0	0110 以外		0110
PDnDDR	0	1	—
管脚功能	PD7 ~ PD4 输入	PD7 ~ PD4 输出	SEG4 ~ SEG1 输出

(n=7 ~ 4)

7.7.5 H8S/2280 群（HD64F2280RB）的管脚功能

端口 D 可用作 LCD 的段的输出管脚。寄存器的设定值与管脚功能的关系如下所示。

- PD7/SEG8、PD6/SEG7、PD5/SEG6、PD4/SEG5

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PCnDDR 位的组合产生以下切换。

SGS3 ~ SGS0	0110、0111 以外		0110、0111
PDnDDR	0	1	—
管脚功能	PD7 ~ PD4 输入	PD7 ~ PD4 输出	SEG8 ~ SEG5 输出

(n=7 ~ 4)

7.8 端口 F

端口 F 是 8 位的 I/O 端口，也可作其它功能用。端口 F 有以下寄存器。

- 数据方向寄存器（PFDDR）
- 数据寄存器（PFDR）
- 端口寄存器（PORTF）

7.8.1 端口 F 数据方向寄存器（PFDDR）

PFDDR 的每一位分别设定端口 F 上各管脚的输入 / 输出状态。

位	位名	初始值	R/W	说 明
7	PF7DDR	0	W	在选择通用输入 / 输出端口的功能后，将该位置 1，PF7 管脚成为 ϕ 输出端口，如果清零后，则为输入端口。
6	PF6DDR	0	W	在选择通用输入 / 输出端口的功能后，将该位置 1，对应管脚成为输出端口，如果清零后，成为输入端口。
5	PF5DDR	0	W	
4	PF4DDR	0	W	
3	PF3DDR	0	W	
2	PF2DDR	0	W	
1	PF1DDR*	0	W	
0	PF0DDR*	0	W	

【注】 *H8S/2282 群为保留位。读后读出不定值。

7.8.2 端口 F 数据寄存器（PFDR）

PFDR 保存端口 F 的输出数据。

位	位名	初始值	R/W	说 明
7	—	0	R/W	保留位 总写为 0。
6	PF6DR	0	R/W	保存被设定为通用输出端口使用的管脚后输出的数据。
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	

【注】 *H8S/2282 群为保留位。读后读出不定值。

7.8.3 端口 F 寄存器（PORTF）

PORTF 表示端口 F 管脚的状态。

位	位名	初始值	R/W	说 明
7	PF7	不定 *1	R	此寄存器在读操作时，如果 PFDDR 位置 1，则读出 PFDR 的值。如果 PFDDR 位清零，则读出管脚的状态。
6	PF6	不定 *1	R	
5	PF5	不定 *1	R	
4	PF4	不定 *1	R	
3	PF3	不定 *1	R	
2	PF2	不定 *1	R	
1	PF1*2	不定 *1	R	
0	PF0*2	不定 *1	R	

【注】 *1 决定于 PF7 ~ PF0 管脚的状态。

*2 H8S/2282 群为保留位。读后读出不定值。

7.8.4 H8S/2282 群、H8S/2280 群（HD64F2280B）的管脚功能

端口 F 可用作中断输入管脚、A/D 转换器的起始触发输入管脚、LCD 的段输出管脚、及系统时钟输出管脚。寄存器的设定值与管脚功能的关系如下所示。

- PF7/ ϕ

通过 PF7DDR 产生以下切换结果。

PF7DDR	0	1
管脚功能	PF7 输入	ϕ 输出

- PF6/SEG24

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PF6DDR 位的组合产生以下切换。

SGS3 ~ SGS0	0000		0000 以外
PF6DDR	0	1	—
管脚功能	PF6 输入	PF6 输出	SEG24 输出

- PF5/SEG23

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PF5DDR 位的组合产生以下切换。

SGS3 ~ SGS0	0000		0000 以外
PF5DDR	0	1	—
管脚功能	PF5 输入	PF5 输出	SEG23 输出

- PF4/SEG22

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PF4DDR 位的组合产生以下切换。

SGS3 ~ SGS0	0000		0000 以外
PF4DDR	0	1	—
管脚功能	PF4 输入	PF4 输出	SEG22 输出

- PF3/ADTRG/ $\overline{\text{IRQ3}}$

通过 A/D 转换器的 ADCR 的 TRGS1、TRGS0 位、及 PF3DDR 位的组合产生以下切换。

PF3DDR	0	1
管脚功能	PF3 输入	PF3 输出
	$\overline{\text{ADTRG}}$ 输入 *	
	$\overline{\text{IRQ3}}$ 输入	

【注】 *TRGS1=1、TRGS0=1 时，成为 ADTRG 输入状态。

- PF2/SEG21

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PF2DDR 位的组合产生以下切换。

SGS3 ~ SGS0	0000		0000 以外
PF2DDR	0	1	—
管脚功能	PF2 输入	PF2 输出	SEG21 输出

- PF1[仅 H8S/2280 群 (HD64F2280B)]

通过 PF1DDR 产生以下切换。

PF1DDR	0	1
管脚功能	PF1 输入	PF1 输出

- PF0/ $\overline{\text{IRQ2}}$ [仅 H8S/2280 群 (HD64F2280B)]

通过 PF0DDR 产生以下切换。

PF0DDR	0	1
管脚功能	PF0 输入	PF0 输出
	$\overline{\text{IRQ2}}$ 输入	

7.8.5 H8S/2280 群（HD64F2280RB）的管脚功能

端口 F 可用作中断输入管脚、A/D 转换器的起始触发输入管脚、LCD 的段输出管脚，及系统时钟输出管脚。寄存器的设定值与管脚功能的关系如下所示。

- PF7/ ϕ

通过 PF7DDR 产生以下切换。

PF7DDR	0	1
管脚功能	PF7 输入	ϕ 输出

- PF6/SEG28

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PF6DDR 位的组合产生以下切换。

SGS3 ~ SGS0	0000		0000 以外
PF6DDR	0	1	—
管脚功能	PF6 输入	PF6 输出	SEG28 输出

- PF5/SEG27

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PF5DDR 位的组合产生以下切换。

SGS3 ~ SGS0	0000		0000 以外
PF5DDR	0	1	—
管脚功能	PF5 输入	PF5 输出	SEG27 输出

- PF4/SEG26

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PF4DDR 位的组合产生以下切换。

SGS3 ~ SGS0	0000		0000 以外
PF4DDR	0	1	—
管脚功能	PF4 输入	PF4 输出	SEG26 输出

- PF3/ADTRG/ $\overline{\text{IRQ3}}$

通过 A/D 转换器的 ADCR 的 TRGS1、TRGS0 位及 PF3DDR 位的组合产生以下切换。

PF3DDR	0	1
管脚功能	PF3 输入	PF3 输出
	$\overline{\text{ADTRG}}$ 输入 *	
	$\overline{\text{IRQ3}}$ 输入	

【注】 *TRGS1=1、TRGS0=1 时，成为 ADTRG 输入状态。

- PF2/SEG25

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PF2DDR 位的组合产生以下切换。

SGS3 ~ SGS0	0000		0000 以外
PF2DDR	0	1	—
管脚功能	PF2 输入	PF2 输出	SEG25 输出

- PF1/SEG2

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PF1DDR 位的组合产生以下切换。

SGS3 ~ SGS0	0111 以外		0111
PF1DDR	0	1	—
管脚功能	PF1 输入	PF1 输出	SEG2 输出

- PF0/ $\overline{\text{IRQ2}}$ /SEG1

通过 LCD 的 LPCR 的 SGS3 ~ SGS0 位、及 PF0DDR 位的组合产生以下切换。

SGS3 ~ SGS0	0111 以外		0111
PF0DDR	0	1	—
管脚功能	PF0 输入	PF0 输出	SEG1 输出
	$\overline{\text{IRQ2}}$ 输入		

7.9 端口 H

端口 H 是 8 位的 I/O 端口，也可作其它功能用。端口 H 有以下寄存器。

- 数据寄存器寄存器（PHDDR）
- 数据寄存器（PHDR）
- 端口寄存器（PORTH）

7.9.1 端口 H 数据方向寄存器（PHDDR）

PHDDR 的每一位分别设定端口 C 上各脚的输入 / 输出状态。

位	位名	初始值	R/W	说 明
7	PH7DDR	0	W	在选择通用输入 / 输出端口的功能后，将该位设置为 1，对应管脚成为输出端口，如果清零，则为输入端口。
6	PH6DDR	0	W	
5	PH5DDR	0	W	
4	PH4DDR	0	W	
3	PH3DDR	0	W	
2	PH2DDR	0	W	
1	PH1DDR	0	W	
0	PH0DDR	0	W	

7.9.2 端口 H 数据寄存器（PHDR）

PHDR 保存端口 H 的输出数据。

位	位名	初始值	R/W	说 明
7	PH7DR	0	R/W	保存被设置为通用输出端口使用的管脚后输出的数据。
6	PH6DR	0	R/W	
5	PH5DR	0	R/W	
4	PH4DR	0	R/W	
3	PH3DR	0	R/W	
2	PH2DR	0	R/W	
1	PH1DR	0	R/W	
0	PH0DR	0	R/W	

7.9.3 端口 H 寄存器（PORTH）

PORTH 表示端口 H 的管脚的状态。不可修改 PORT。

位	位名	初始值	R/W	说 明
7	PH7	不定 *	R	此寄存器在读操作时，如果 PHDDR 位置 1，则读出 PHDR 的值。如果 PHDDR 位清零，则读出管脚的状态。
6	PH6	不定 *	R	
5	PH5	不定 *	R	
4	PH4	不定 *	R	
3	PH3	不定 *	R	
2	PH2	不定 *	R	
1	PH1	不定 *	R	
0	PH0	不定 *	R	

【注】 * 决定于 PH7 ~ PH0 管脚的状态。

7.9.4 管脚功能

端口 H 可用作 PWM_1 输出管脚。寄存器的设定值与管脚的关系如下所示。

- PH7/PWM1H

通过 PWM_1 的 PWOCR_1 的 OE1H 位、及 PH7DDR 位的组合产生以下切换。

OE1H	0		1
PH7DDR	0	1	—
管脚功能	PH7 输入	PH7 输出	PWM1H 输出

- PH6/PWM1G

通过 PWM_1 的 PWOCR_1 的 OE1G 位、及 PH6DDR 位的组合产生以下切换。

OE1G	0		1
PH6DDR	0	1	—
管脚功能	PH6 输入	PH6 输出	PWM1G 输出

- PH5/PWM1F

通过 PWM_1 的 PWOCR_1 的 OE1F 位、及 PH5DDR 位的组合产生以下切换。

OE1F	0		1
PH5DDR	0	1	—
管脚功能	PH5 输入	PH5 输出	PWM1F 输出

- PH4/PWM1E

通过 PWM_1 的 PWOCR_1 的 OE1E 位、及 PH4DDR 位的组合产生以下切换。

OE1E	0		1
PH4DDR	0	1	—
管脚功能	PH4 输入	PH4 输出	PWM1E 输出

- PH3/PWM1D

通过 PWM_1 的 PWOCR_1 的 OE1D 位、及 PH3DDR 位的组合产生以下切换。

OE1D	0		1
PH3DDR	0	1	—
管脚功能	PH3 输入	PH3 输出	PWM1D 输出

- PH2/PWM1C

通过 PWM_1 的 PWOCR_1 的 OE1C 位、及 PH2DDR 位的组合产生以下切换。

OE1C	0		1
PH2DDR	0	1	—
管脚功能	PH2 输入	PH2 输出	PWM1C 输出

- PH1/PWM1B

通过 PWM_1 的 PWOCR_1 的 OE1B 位、及 PH1DDR 位的组合产生以下切换。

OE1B	0		1
PH1DDR	0	1	—
管脚功能	PH1 输入	PH1 输出	PWM1B 输出

- PH0/PWM1A

通过 PWM_1 的 PWOCR_1 的 OE1A 位、及 PH0DDR 位的组合产生以下切换。

OE1A	0		1
PH0DDR	0	1	—
管脚功能	PH0 输入	PH0 输出	PWM1A 输出

7.10 端口 J

端口 J 是 8 位的 I/O 端口，也可作其它功能用端口。端口 J 有以下寄存器。

- 数据方向寄存器（PJDDR）
- 数据寄存器（PJDR）
- 端口寄存器（PORTJ）

7.10.1 端口 J 数据方向寄存器（PJDDR）

PJDDR 的每一位分别设定端口 C 上各脚的输入 / 输出状态。

位	位名	初始值	R/W	说 明
7	PJ7DDR	0	W	在选择通用输入 / 输出端口的功能后，将该位置 1，对应管脚成为输出端口，如果清零，则为输入端口。
6	PJ6DDR	0	W	
5	PJ5DDR	0	W	
4	PJ4DDR	0	W	
3	PJ3DDR	0	W	
2	PJ2DDR	0	W	
1	PJ1DDR	0	W	
0	PJ0DDR	0	W	

7.10.2 端口 J 数据寄存器（PJDR）

PJDR 保存端口 J 的输出数据。

位	位名	初始值	R/W	说 明
7	PJ7DR	0	R/W	保存被设置为通用输入 / 输出端口使用的管脚后输出的数据。
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	

7.10.3 端口 J 寄存器 (PORTJ)

PORTJ 表示分端口 J 的管脚的状态。不可修改 PORTJ。

位	位名	初始值	R/W	说 明
7	PJ7	不定 *	R	此寄存器在读操作时，如果 PJDDR 位置 1，则读出 PJDR 的值。如果 PJDDR 位清零，则读出管脚的状态。
6	PJ6	不定 *	R	
5	PJ5	不定 *	R	
4	PJ4	不定 *	R	
3	PJ3	不定 *	R	
2	PJ2	不定 *	R	
1	PJ1	不定 *	R	
0	PJ0	不定 *	R	

【注】 * 决定于 PJ7 ~ PJ0 管脚的状态。

7.10.4 管脚功能

端口 J 可用作 PWM_2 输出管脚。寄存器的设定值与管脚的关系如下所示。

- PJ7/PWM2H

通过 PWM_2 的 PWOCR_2 的 OE1A 位、及 PJ7DDR 位的组合产生以下切换。

OE2H	0		1
PJ7DDR	0	1	—
管脚功能	PJ7 输入	PJ7 输出	PWM2H 输出

- PJ6/PWM2G

通过 PWM_2 的 PWOCR_2 的 OE2G 位、及 PJ6DDR 位的组合产生以下切换如。

OE2G	0		1
PJ6DDR	0	1	—
管脚功能	PJ6 输入	PJ6 输出	PWM2G 输出

- PJ5/PWM2F

通过 PWM_2 的 PWOCR_2 的 OE2F 位、及 PJ5DDR 位的组合产生以下切换。

OE2F	0		1
PJ5DDR	0	1	—
管脚功能	PJ5 输入	PJ5 输出	PWM2F 输出

- PJ4/PWM2E

通过 PWM_2 的 PWOCR_2 的 OE2E 位、及 PJ4DDR 位的组合产生以下切换。

OE2E	0		1
PJ4DDR	0	1	—
管脚功能	PJ4 输入	PJ4 输出	PWM2E 输出

- PJ3/PWM2D

通过 PWM_2 的 PWOCR_2 的 OE2D 位、及 PJ3DDR 位的组合产生以下切换。

OE2D	0		1
PJ3DDR	0	1	—
管脚功能	PJ3 输入	PJ3 输出	PWM2D 输出

- PJ2/PWM2C

通过 PWM_2 的 PWOCR_2 的 OE2C 位、及 PJ2DDR 位的组合产生以下切换。

OE2C	0		1
PJ2DDR	0	1	—
管脚功能	PJ2 输入	PJ2 输出	PWM2C 输出

- PJ1/PWM2B

通过 PWM_2 的 PWOCR_2 的 OE2B 位、及 PJ1DDR 位的组合产生以下切换。

OE2B	0		1
PJ1DDR	0	1	—
管脚功能	PJ1 输入	PJ1 输出	PWM2B 输出

- PJ0/PWM2A

通过 PWM_2 的 PWOCR_2 的 OE2A 位、及 PJ0DDR 位的组合产生以下切换。

OE2A	0		1
PJ0DDR	0	1	—
管脚功能	PJ0 输入	PJ0 输出	PWM2A 输出

7.11 管脚切换功能

端口 H、端口 J 通过 TRPRT 的 TRPB、TRPA 位，各自的高位和低位以每 4 位进行切换。

7.11.1 传送寄存器（TRPRT）

TRPRT 是通过各位的组合，来设定端口 H 与端口 J 的管脚功能的切换。

位	位名	初始值	R/W	说 明
7 6 5 4 3 2	— — — — — —	不定 不定 不定 不定 不定 不定	— — — — — —	保留位 不定值。
1 0	TRPB TRPA	0 0	R/W R/W	
				通过 TRPB 与 TRPA 的组合，来切换端口 H 与端口 J 的管脚功能。 00: 初始值 01: 切换 PH3 ~ PH0 与 PJ3 ~ PJ0 的管脚功能。 10: 切换 PH7 ~ PH4 与 PJ7 ~ PJ4 的管脚功能。 11: 切换 PH7 ~ PH4 与 PJ7 ~ PJ4 的管脚功能， 切换 PH3 ~ PH0 与 PJ3 ~ PJ0 的管脚功能。

7.11.2 通过管脚的切换，读端口寄存器。

在读 PORTH 和 PORTJ 的时候，是利用 TRPRT 的 TRPB 和 TRPA 位来区分所要读取的管脚。表 7.3 示出通过管脚切换，寄存器所读取的管脚。

PORTH、PORTJ 读取的管脚状态，请参考「7.9.3 端口 H 寄存器（PORTH）」及「7.10.3 端口 J 寄存器（PORTJ）」。TRPRT 的设定在写入数据方向寄存器（PHDDR、PJDDR）与数据寄存器（PHDR、PJDR）之前进行。

表 7.3 通过管脚切换，输出寄存器的读取管脚及 PWM

TRPB	TRPA	端口H								端口J																																											
0	0	<table><tr><td>管脚号</td><td>46</td><td>45</td><td>44</td><td>43</td><td>40</td><td>39</td><td>38</td><td>37</td></tr><tr><td>管脚状态</td><td>PH7输入/ PWM1H/ PHDR7</td><td>PH6输入/ PWM1G/ PHDR6</td><td>PH5输入/ PWM1F/ PHDR5</td><td>PH4输入/ PWM1E/ PHDR4</td><td>PH3输入/ PWM1D/ PHDR3</td><td>PH2输入/ PWM1C/ PHDR2</td><td>PH1输入/ PWM1B/ PHDR1</td><td>PH0输入/ PWM1A/ PHDR0</td></tr></table>								管脚号	46	45	44	43	40	39	38	37	管脚状态	PH7输入/ PWM1H/ PHDR7	PH6输入/ PWM1G/ PHDR6	PH5输入/ PWM1F/ PHDR5	PH4输入/ PWM1E/ PHDR4	PH3输入/ PWM1D/ PHDR3	PH2输入/ PWM1C/ PHDR2	PH1输入/ PWM1B/ PHDR1	PH0输入/ PWM1A/ PHDR0	<table><tr><td>管脚号</td><td>56</td><td>55</td><td>54</td><td>53</td><td>50</td><td>49</td><td>48</td><td>47</td></tr><tr><td>管脚状态</td><td>PJ7输入/ PWM2H/ PJDR7</td><td>PJ6输入/ PWM2G/ PJDR6</td><td>PJ5输入/ PWM2F/ PJDR5</td><td>PJ4输入/ PWM2E/ PJDR4</td><td>PJ3输入/ PWM2D/ PJDR3</td><td>PJ2输入/ PWM2C/ PJDR2</td><td>PJ1输入/ PWM2B/ PJDR1</td><td>PJ0输入/ PWM2A/ PJDR0</td></tr></table>								管脚号	56	55	54	53	50	49	48	47	管脚状态	PJ7输入/ PWM2H/ PJDR7	PJ6输入/ PWM2G/ PJDR6	PJ5输入/ PWM2F/ PJDR5	PJ4输入/ PWM2E/ PJDR4	PJ3输入/ PWM2D/ PJDR3	PJ2输入/ PWM2C/ PJDR2	PJ1输入/ PWM2B/ PJDR1	PJ0输入/ PWM2A/ PJDR0
		管脚号	46	45	44	43	40	39	38	37																																											
		管脚状态	PH7输入/ PWM1H/ PHDR7	PH6输入/ PWM1G/ PHDR6	PH5输入/ PWM1F/ PHDR5	PH4输入/ PWM1E/ PHDR4	PH3输入/ PWM1D/ PHDR3	PH2输入/ PWM1C/ PHDR2	PH1输入/ PWM1B/ PHDR1	PH0输入/ PWM1A/ PHDR0																																											
		管脚号	56	55	54	53	50	49	48	47																																											
管脚状态	PJ7输入/ PWM2H/ PJDR7	PJ6输入/ PWM2G/ PJDR6	PJ5输入/ PWM2F/ PJDR5	PJ4输入/ PWM2E/ PJDR4	PJ3输入/ PWM2D/ PJDR3	PJ2输入/ PWM2C/ PJDR2	PJ1输入/ PWM2B/ PJDR1	PJ0输入/ PWM2A/ PJDR0																																													
<table><tr><td>位</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr><tr><td>读取数据</td><td>PH7输入/ PWM1H/ PHDR7</td><td>PH6输入/ PWM1G/ PHDR6</td><td>PH5输入/ PWM1F/ PHDR5</td><td>PH4输入/ PWM1E/ PHDR4</td><td>PH3输入/ PWM1D/ PHDR3</td><td>PH2输入/ PWM1C/ PHDR2</td><td>PH1输入/ PWM1B/ PHDR1</td><td>PH0输入/ PWM1A/ PHDR0</td></tr></table>								位	7	6	5	4	3	2	1	0	读取数据	PH7输入/ PWM1H/ PHDR7	PH6输入/ PWM1G/ PHDR6	PH5输入/ PWM1F/ PHDR5	PH4输入/ PWM1E/ PHDR4	PH3输入/ PWM1D/ PHDR3	PH2输入/ PWM1C/ PHDR2	PH1输入/ PWM1B/ PHDR1	PH0输入/ PWM1A/ PHDR0	<table><tr><td>位</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr><tr><td>读取数据</td><td>PJ7输入/ PWM2H/ PJDR7</td><td>PJ6输入/ PWM2G/ PJDR6</td><td>PJ5输入/ PWM2F/ PJDR5</td><td>PJ4输入/ PWM2E/ PJDR4</td><td>PJ3输入/ PWM2D/ PJDR3</td><td>PJ2输入/ PWM2C/ PJDR2</td><td>PJ1输入/ PWM2B/ PJDR1</td><td>PJ0输入/ PWM2A/ PJDR0</td></tr></table>								位	7	6	5	4	3	2	1	0	读取数据	PJ7输入/ PWM2H/ PJDR7	PJ6输入/ PWM2G/ PJDR6	PJ5输入/ PWM2F/ PJDR5	PJ4输入/ PWM2E/ PJDR4	PJ3输入/ PWM2D/ PJDR3	PJ2输入/ PWM2C/ PJDR2	PJ1输入/ PWM2B/ PJDR1	PJ0输入/ PWM2A/ PJDR0		
位	7	6	5	4	3	2	1	0																																													
读取数据	PH7输入/ PWM1H/ PHDR7	PH6输入/ PWM1G/ PHDR6	PH5输入/ PWM1F/ PHDR5	PH4输入/ PWM1E/ PHDR4	PH3输入/ PWM1D/ PHDR3	PH2输入/ PWM1C/ PHDR2	PH1输入/ PWM1B/ PHDR1	PH0输入/ PWM1A/ PHDR0																																													
位	7	6	5	4	3	2	1	0																																													
读取数据	PJ7输入/ PWM2H/ PJDR7	PJ6输入/ PWM2G/ PJDR6	PJ5输入/ PWM2F/ PJDR5	PJ4输入/ PWM2E/ PJDR4	PJ3输入/ PWM2D/ PJDR3	PJ2输入/ PWM2C/ PJDR2	PJ1输入/ PWM2B/ PJDR1	PJ0输入/ PWM2A/ PJDR0																																													
0	1	<table><tr><td>管脚号</td><td>46</td><td>45</td><td>44</td><td>43</td><td>40</td><td>39</td><td>38</td><td>37</td></tr><tr><td>管脚状态</td><td>PH7输入/ PWM1H/ PHDR7</td><td>PH6输入/ PWM1G/ PHDR6</td><td>PH5输入/ PWM1F/ PHDR5</td><td>PH4输入/ PWM1E/ PHDR4</td><td>PH3输入/ PWM1D/ PJDR3</td><td>PH2输入/ PWM1C/ PJDR2</td><td>PH1输入/ PWM1B/ PJDR1</td><td>PH0输入/ PWM1A/ PJDR0</td></tr></table>								管脚号	46	45	44	43	40	39	38	37	管脚状态	PH7输入/ PWM1H/ PHDR7	PH6输入/ PWM1G/ PHDR6	PH5输入/ PWM1F/ PHDR5	PH4输入/ PWM1E/ PHDR4	PH3输入/ PWM1D/ PJDR3	PH2输入/ PWM1C/ PJDR2	PH1输入/ PWM1B/ PJDR1	PH0输入/ PWM1A/ PJDR0	<table><tr><td>管脚号</td><td>56</td><td>55</td><td>54</td><td>53</td><td>50</td><td>49</td><td>48</td><td>47</td></tr><tr><td>管脚状态</td><td>PJ7输入/ PWM2H/ PJDR7</td><td>PJ6输入/ PWM2G/ PJDR6</td><td>PJ5输入/ PWM2F/ PJDR5</td><td>PJ4输入/ PWM2E/ PJDR4</td><td>PJ3输入/ PWM1D/ PHDR3</td><td>PJ2输入/ PWM1C/ PHDR2</td><td>PJ1输入/ PWM1B/ PHDR1</td><td>PJ0输入/ PWM1A/ PHDR0</td></tr></table>								管脚号	56	55	54	53	50	49	48	47	管脚状态	PJ7输入/ PWM2H/ PJDR7	PJ6输入/ PWM2G/ PJDR6	PJ5输入/ PWM2F/ PJDR5	PJ4输入/ PWM2E/ PJDR4	PJ3输入/ PWM1D/ PHDR3	PJ2输入/ PWM1C/ PHDR2	PJ1输入/ PWM1B/ PHDR1	PJ0输入/ PWM1A/ PHDR0
		管脚号	46	45	44	43	40	39	38	37																																											
		管脚状态	PH7输入/ PWM1H/ PHDR7	PH6输入/ PWM1G/ PHDR6	PH5输入/ PWM1F/ PHDR5	PH4输入/ PWM1E/ PHDR4	PH3输入/ PWM1D/ PJDR3	PH2输入/ PWM1C/ PJDR2	PH1输入/ PWM1B/ PJDR1	PH0输入/ PWM1A/ PJDR0																																											
		管脚号	56	55	54	53	50	49	48	47																																											
管脚状态	PJ7输入/ PWM2H/ PJDR7	PJ6输入/ PWM2G/ PJDR6	PJ5输入/ PWM2F/ PJDR5	PJ4输入/ PWM2E/ PJDR4	PJ3输入/ PWM1D/ PHDR3	PJ2输入/ PWM1C/ PHDR2	PJ1输入/ PWM1B/ PHDR1	PJ0输入/ PWM1A/ PHDR0																																													
<table><tr><td>位</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr><tr><td>读取数据</td><td>PH7输入/ PWM1H/ PHDR7</td><td>PH6输入/ PWM1G/ PHDR6</td><td>PH5输入/ PWM1F/ PHDR5</td><td>PH4输入/ PWM1E/ PHDR4</td><td>PJ3输入/ PWM1D/ PHDR3</td><td>PJ2输入/ PWM1C/ PHDR2</td><td>PJ1输入/ PWM1B/ PHDR1</td><td>PJ0输入/ PWM1A/ PHDR0</td></tr></table>								位	7	6	5	4	3	2	1	0	读取数据	PH7输入/ PWM1H/ PHDR7	PH6输入/ PWM1G/ PHDR6	PH5输入/ PWM1F/ PHDR5	PH4输入/ PWM1E/ PHDR4	PJ3输入/ PWM1D/ PHDR3	PJ2输入/ PWM1C/ PHDR2	PJ1输入/ PWM1B/ PHDR1	PJ0输入/ PWM1A/ PHDR0	<table><tr><td>位</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr><tr><td>读取数据</td><td>PJ7输入/ PWM2H/ PJDR7</td><td>PJ6输入/ PWM2G/ PJDR6</td><td>PJ5输入/ PWM2F/ PJDR5</td><td>PJ4输入/ PWM2E/ PJDR4</td><td>PH3输入/ PWM2D/ PJDR3</td><td>PH2输入/ PWM2C/ PJDR2</td><td>PH1输入/ PWM2B/ PJDR1</td><td>PH0输入/ PWM2A/ PJDR0</td></tr></table>								位	7	6	5	4	3	2	1	0	读取数据	PJ7输入/ PWM2H/ PJDR7	PJ6输入/ PWM2G/ PJDR6	PJ5输入/ PWM2F/ PJDR5	PJ4输入/ PWM2E/ PJDR4	PH3输入/ PWM2D/ PJDR3	PH2输入/ PWM2C/ PJDR2	PH1输入/ PWM2B/ PJDR1	PH0输入/ PWM2A/ PJDR0		
位	7	6	5	4	3	2	1	0																																													
读取数据	PH7输入/ PWM1H/ PHDR7	PH6输入/ PWM1G/ PHDR6	PH5输入/ PWM1F/ PHDR5	PH4输入/ PWM1E/ PHDR4	PJ3输入/ PWM1D/ PHDR3	PJ2输入/ PWM1C/ PHDR2	PJ1输入/ PWM1B/ PHDR1	PJ0输入/ PWM1A/ PHDR0																																													
位	7	6	5	4	3	2	1	0																																													
读取数据	PJ7输入/ PWM2H/ PJDR7	PJ6输入/ PWM2G/ PJDR6	PJ5输入/ PWM2F/ PJDR5	PJ4输入/ PWM2E/ PJDR4	PH3输入/ PWM2D/ PJDR3	PH2输入/ PWM2C/ PJDR2	PH1输入/ PWM2B/ PJDR1	PH0输入/ PWM2A/ PJDR0																																													
1	0	<table><tr><td>管脚号</td><td>46</td><td>45</td><td>44</td><td>43</td><td>40</td><td>39</td><td>38</td><td>37</td></tr><tr><td>管脚状态</td><td>PH7输入/ PWM2H/ PJDR7</td><td>PH6输入/ PWM2G/ PJDR6</td><td>PH5输入/ PWM2F/ PJDR5</td><td>PH4输入/ PWM2E/ PJDR4</td><td>PH3输入/ PWM2D/ PHDR3</td><td>PH2输入/ PWM1C/ PHDR2</td><td>PH1输入/ PWM1B/ PHDR1</td><td>PH0输入/ PWM1A/ PHDR0</td></tr></table>								管脚号	46	45	44	43	40	39	38	37	管脚状态	PH7输入/ PWM2H/ PJDR7	PH6输入/ PWM2G/ PJDR6	PH5输入/ PWM2F/ PJDR5	PH4输入/ PWM2E/ PJDR4	PH3输入/ PWM2D/ PHDR3	PH2输入/ PWM1C/ PHDR2	PH1输入/ PWM1B/ PHDR1	PH0输入/ PWM1A/ PHDR0	<table><tr><td>管脚号</td><td>56</td><td>55</td><td>54</td><td>53</td><td>50</td><td>49</td><td>48</td><td>47</td></tr><tr><td>管脚状态</td><td>PJ7输入/ PWM1H/ PHDR7</td><td>PJ6输入/ PWM1G/ PHDR6</td><td>PJ5输入/ PWM1F/ PHDR5</td><td>PJ4输入/ PWM1E/ PHDR4</td><td>PJ3输入/ PWM2D/ PJDR3</td><td>PJ2输入/ PWM2C/ PJDR2</td><td>PJ1输入/ PWM2B/ PJDR1</td><td>PJ0输入/ PWM2A/ PJDR0</td></tr></table>								管脚号	56	55	54	53	50	49	48	47	管脚状态	PJ7输入/ PWM1H/ PHDR7	PJ6输入/ PWM1G/ PHDR6	PJ5输入/ PWM1F/ PHDR5	PJ4输入/ PWM1E/ PHDR4	PJ3输入/ PWM2D/ PJDR3	PJ2输入/ PWM2C/ PJDR2	PJ1输入/ PWM2B/ PJDR1	PJ0输入/ PWM2A/ PJDR0
		管脚号	46	45	44	43	40	39	38	37																																											
		管脚状态	PH7输入/ PWM2H/ PJDR7	PH6输入/ PWM2G/ PJDR6	PH5输入/ PWM2F/ PJDR5	PH4输入/ PWM2E/ PJDR4	PH3输入/ PWM2D/ PHDR3	PH2输入/ PWM1C/ PHDR2	PH1输入/ PWM1B/ PHDR1	PH0输入/ PWM1A/ PHDR0																																											
		管脚号	56	55	54	53	50	49	48	47																																											
管脚状态	PJ7输入/ PWM1H/ PHDR7	PJ6输入/ PWM1G/ PHDR6	PJ5输入/ PWM1F/ PHDR5	PJ4输入/ PWM1E/ PHDR4	PJ3输入/ PWM2D/ PJDR3	PJ2输入/ PWM2C/ PJDR2	PJ1输入/ PWM2B/ PJDR1	PJ0输入/ PWM2A/ PJDR0																																													
<table><tr><td>位</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr><tr><td>读取数据</td><td>PJ7输入/ PWM1H/ PHDR7</td><td>PJ6输入/ PWM1G/ PHDR6</td><td>PJ5输入/ PWM1F/ PHDR5</td><td>PJ4输入/ PWM1E/ PHDR4</td><td>PH3输入/ PWM1D/ PHDR3</td><td>PH2输入/ PWM1C/ PHDR2</td><td>PH1输入/ PWM1B/ PHDR1</td><td>PH0输入/ PWM1A/ PHDR0</td></tr></table>								位	7	6	5	4	3	2	1	0	读取数据	PJ7输入/ PWM1H/ PHDR7	PJ6输入/ PWM1G/ PHDR6	PJ5输入/ PWM1F/ PHDR5	PJ4输入/ PWM1E/ PHDR4	PH3输入/ PWM1D/ PHDR3	PH2输入/ PWM1C/ PHDR2	PH1输入/ PWM1B/ PHDR1	PH0输入/ PWM1A/ PHDR0	<table><tr><td>位</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr><tr><td>读取数据</td><td>PH7输入/ PWM2H/ PJDR7</td><td>PH6输入/ PWM2G/ PJDR6</td><td>PH5输入/ PWM2F/ PJDR5</td><td>PH4输入/ PWM2E/ PJDR4</td><td>PJ3输入/ PWM2D/ PJDR3</td><td>PJ2输入/ PWM2C/ PJDR2</td><td>PJ1输入/ PWM2B/ PJDR1</td><td>PJ0输入/ PWM2A/ PJDR0</td></tr></table>								位	7	6	5	4	3	2	1	0	读取数据	PH7输入/ PWM2H/ PJDR7	PH6输入/ PWM2G/ PJDR6	PH5输入/ PWM2F/ PJDR5	PH4输入/ PWM2E/ PJDR4	PJ3输入/ PWM2D/ PJDR3	PJ2输入/ PWM2C/ PJDR2	PJ1输入/ PWM2B/ PJDR1	PJ0输入/ PWM2A/ PJDR0		
位	7	6	5	4	3	2	1	0																																													
读取数据	PJ7输入/ PWM1H/ PHDR7	PJ6输入/ PWM1G/ PHDR6	PJ5输入/ PWM1F/ PHDR5	PJ4输入/ PWM1E/ PHDR4	PH3输入/ PWM1D/ PHDR3	PH2输入/ PWM1C/ PHDR2	PH1输入/ PWM1B/ PHDR1	PH0输入/ PWM1A/ PHDR0																																													
位	7	6	5	4	3	2	1	0																																													
读取数据	PH7输入/ PWM2H/ PJDR7	PH6输入/ PWM2G/ PJDR6	PH5输入/ PWM2F/ PJDR5	PH4输入/ PWM2E/ PJDR4	PJ3输入/ PWM2D/ PJDR3	PJ2输入/ PWM2C/ PJDR2	PJ1输入/ PWM2B/ PJDR1	PJ0输入/ PWM2A/ PJDR0																																													
1	1	<table><tr><td>管脚号</td><td>46</td><td>45</td><td>44</td><td>43</td><td>40</td><td>39</td><td>38</td><td>37</td></tr><tr><td>管脚状态</td><td>PH7输入/ PWM2H/ PJDR7</td><td>PH6输入/ PWM2G/ PJDR6</td><td>PH5输入/ PWM2F/ PJDR5</td><td>PH4输入/ PWM2E/ PJDR4</td><td>PH3输入/ PWM2D/ PJDR3</td><td>PH2输入/ PWM2C/ PJDR2</td><td>PH1输入/ PWM2B/ PJDR1</td><td>PH0输入/ PWM2A/ PJDR0</td></tr></table>								管脚号	46	45	44	43	40	39	38	37	管脚状态	PH7输入/ PWM2H/ PJDR7	PH6输入/ PWM2G/ PJDR6	PH5输入/ PWM2F/ PJDR5	PH4输入/ PWM2E/ PJDR4	PH3输入/ PWM2D/ PJDR3	PH2输入/ PWM2C/ PJDR2	PH1输入/ PWM2B/ PJDR1	PH0输入/ PWM2A/ PJDR0	<table><tr><td>管脚号</td><td>56</td><td>55</td><td>54</td><td>53</td><td>50</td><td>49</td><td>48</td><td>47</td></tr><tr><td>管脚状态</td><td>PJ7输入/ PWM1H/ PHDR7</td><td>PJ6输入/ PWM1G/ PHDR6</td><td>PJ5输入/ PWM1F/ PHDR5</td><td>PJ4输入/ PWM1E/ PHDR4</td><td>PJ3输入/ PWM1D/ PHDR3</td><td>PJ2输入/ PWM1C/ PHDR2</td><td>PJ1输入/ PWM1B/ PHDR1</td><td>PJ0输入/ PWM1A/ PHDR0</td></tr></table>								管脚号	56	55	54	53	50	49	48	47	管脚状态	PJ7输入/ PWM1H/ PHDR7	PJ6输入/ PWM1G/ PHDR6	PJ5输入/ PWM1F/ PHDR5	PJ4输入/ PWM1E/ PHDR4	PJ3输入/ PWM1D/ PHDR3	PJ2输入/ PWM1C/ PHDR2	PJ1输入/ PWM1B/ PHDR1	PJ0输入/ PWM1A/ PHDR0
		管脚号	46	45	44	43	40	39	38	37																																											
		管脚状态	PH7输入/ PWM2H/ PJDR7	PH6输入/ PWM2G/ PJDR6	PH5输入/ PWM2F/ PJDR5	PH4输入/ PWM2E/ PJDR4	PH3输入/ PWM2D/ PJDR3	PH2输入/ PWM2C/ PJDR2	PH1输入/ PWM2B/ PJDR1	PH0输入/ PWM2A/ PJDR0																																											
		管脚号	56	55	54	53	50	49	48	47																																											
管脚状态	PJ7输入/ PWM1H/ PHDR7	PJ6输入/ PWM1G/ PHDR6	PJ5输入/ PWM1F/ PHDR5	PJ4输入/ PWM1E/ PHDR4	PJ3输入/ PWM1D/ PHDR3	PJ2输入/ PWM1C/ PHDR2	PJ1输入/ PWM1B/ PHDR1	PJ0输入/ PWM1A/ PHDR0																																													
<table><tr><td>位</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr><tr><td>读取状态</td><td>PJ7输入/ PWM1H/ PHDR7</td><td>PJ6输入/ PWM1G/ PHDR6</td><td>PJ5输入/ PWM1F/ PHDR5</td><td>PJ4输入/ PWM1E/ PHDR4</td><td>PJ3输入/ PWM1D/ PHDR3</td><td>PJ2输入/ PWM1C/ PHDR2</td><td>PJ1输入/ PWM1B/ PHDR1</td><td>PJ0输入/ PWM1A/ PHDR0</td></tr></table>								位	7	6	5	4	3	2	1	0	读取状态	PJ7输入/ PWM1H/ PHDR7	PJ6输入/ PWM1G/ PHDR6	PJ5输入/ PWM1F/ PHDR5	PJ4输入/ PWM1E/ PHDR4	PJ3输入/ PWM1D/ PHDR3	PJ2输入/ PWM1C/ PHDR2	PJ1输入/ PWM1B/ PHDR1	PJ0输入/ PWM1A/ PHDR0	<table><tr><td>位</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr><tr><td>读取数据</td><td>PH7输入/ PWM2H/ PJDR7</td><td>PH6输入/ PWM2G/ PJDR6</td><td>PH5输入/ PWM2F/ PJDR5</td><td>PH4输入/ PWM2E/ PJDR4</td><td>PH3输入/ PWM2D/ PJDR3</td><td>PH2输入/ PWM2C/ PJDR2</td><td>PH1输入/ PWM2B/ PJDR1</td><td>PH0输入/ PWM2A/ PJDR0</td></tr></table>								位	7	6	5	4	3	2	1	0	读取数据	PH7输入/ PWM2H/ PJDR7	PH6输入/ PWM2G/ PJDR6	PH5输入/ PWM2F/ PJDR5	PH4输入/ PWM2E/ PJDR4	PH3输入/ PWM2D/ PJDR3	PH2输入/ PWM2C/ PJDR2	PH1输入/ PWM2B/ PJDR1	PH0输入/ PWM2A/ PJDR0		
位	7	6	5	4	3	2	1	0																																													
读取状态	PJ7输入/ PWM1H/ PHDR7	PJ6输入/ PWM1G/ PHDR6	PJ5输入/ PWM1F/ PHDR5	PJ4输入/ PWM1E/ PHDR4	PJ3输入/ PWM1D/ PHDR3	PJ2输入/ PWM1C/ PHDR2	PJ1输入/ PWM1B/ PHDR1	PJ0输入/ PWM1A/ PHDR0																																													
位	7	6	5	4	3	2	1	0																																													
读取数据	PH7输入/ PWM2H/ PJDR7	PH6输入/ PWM2G/ PJDR6	PH5输入/ PWM2F/ PJDR5	PH4输入/ PWM2E/ PJDR4	PH3输入/ PWM2D/ PJDR3	PH2输入/ PWM2C/ PJDR2	PH1输入/ PWM2B/ PJDR1	PH0输入/ PWM2A/ PJDR0																																													

第 8 章 16 位定时器脉冲单元 (TPU)

在本 LSI 内设置由 3 通道的 16 位定时器构成的 16 位定时器脉冲单元 (TPU)。16 位定时器脉冲单元的功能如表 8.1 所示, 框图如图 8.1 所示。

8.1 特点

- 最大能进行 8 个脉冲的输入 / 输出。
- 每个通道能选择 8 种计数器输入时钟。
- 各通道都能同时设定下面的工作 : 通过比较匹配产生波形输出工作、输入捕捉的输入管脚功能工作、计数器清除工作、同时写入复数的定时计数器 (TCNT) 工作、通过比较匹配 / 输入捕捉的输入管脚产生同时清除工作、通过计数器的同步工作产生各寄存器同步输入 / 输出工作、与同步运作组合输出最大 7 相的 PWM 工作。
- 通道 0 能设定缓冲工作。
- 通道 1、2 能设定各个独立的相位计数模式。
- 通过内部 16 位总线实现高速存取。
- 13 种中断源。
- 能产生 A/D 转换器的转换起始触发。
- 能设定模块待机模式。

表 8.1 TPU 的功能一览表

项 目	通道 0	通道 1	通道 2
计数时钟	$\phi / 1$ $\phi / 4$ $\phi / 16$ $\phi / 64$ TCLKA TCLKB TCLKC TCLKD	$\phi / 1$ $\phi / 4$ $\phi / 16$ $\phi / 64$ $\phi / 256$ TCLKA TCLKB	$\phi / 1$ $\phi / 4$ $\phi / 16$ $\phi / 64$ $\phi / 1024$ TCLKA TCLKB TCLKC
通用寄存器 (TGR)	TGRA_0 TGRB_0	TGRA_1 TGRB_1	TGRA_2 TGRB_2
通用寄存器 / 缓冲寄存器	TGRC_0 TGRD_0	—	—
输入 / 输出管脚	TIOCA0 TIOCB0 TIOCC0 TIOCD0	TIOCA1 TIOCB1	TIOCA2 TIOCB2
计数器清除功能	TGR 的比较匹配或输入捕捉的输入管脚	TGR 的比较匹配或输入捕捉的输入管脚	TGR 的比较匹配或输入捕捉的输入管脚

项 目		通道 0	通道 1	通道 2
比较匹配 输出	0 输出	○	○	○
	1 输出	○	○	○
	交替输出	○	○	○
输入捕捉的输入管脚功能		○	○	○
同步工作		○	○	○
PWM 模式		○	○	○
相位计数模式		—	○	○
缓冲工作		○	—	—
A/D 转换开始触发		TGRA_0 的比较匹配或输入捕捉的输入管脚	TGRA_1 的比较匹配或输入捕捉的输入管脚	TGRA_2 的比较匹配或输入捕捉的输入管脚
中断源		5 源 - 比较匹配 / 输入捕捉的输入管脚 OA - 比较匹配 / 输入捕捉的输入管脚 OB - 比较匹配 / 输入捕捉的输入管脚 OC - 比较匹配 / 输入捕捉的输入管脚 OD - 溢出	4 源 - 比较匹配 / 输入捕捉的输入管脚 1A - 比较匹配 / 输入捕捉的输入管脚 1B - 溢出 - 下溢	4 源 - 比较匹配 / 输入捕捉的输入管脚 2A - 比较匹配 / 输入捕捉的输入管脚 2B - 溢出 - 下溢

【符号说明】

○：可能

—：不可

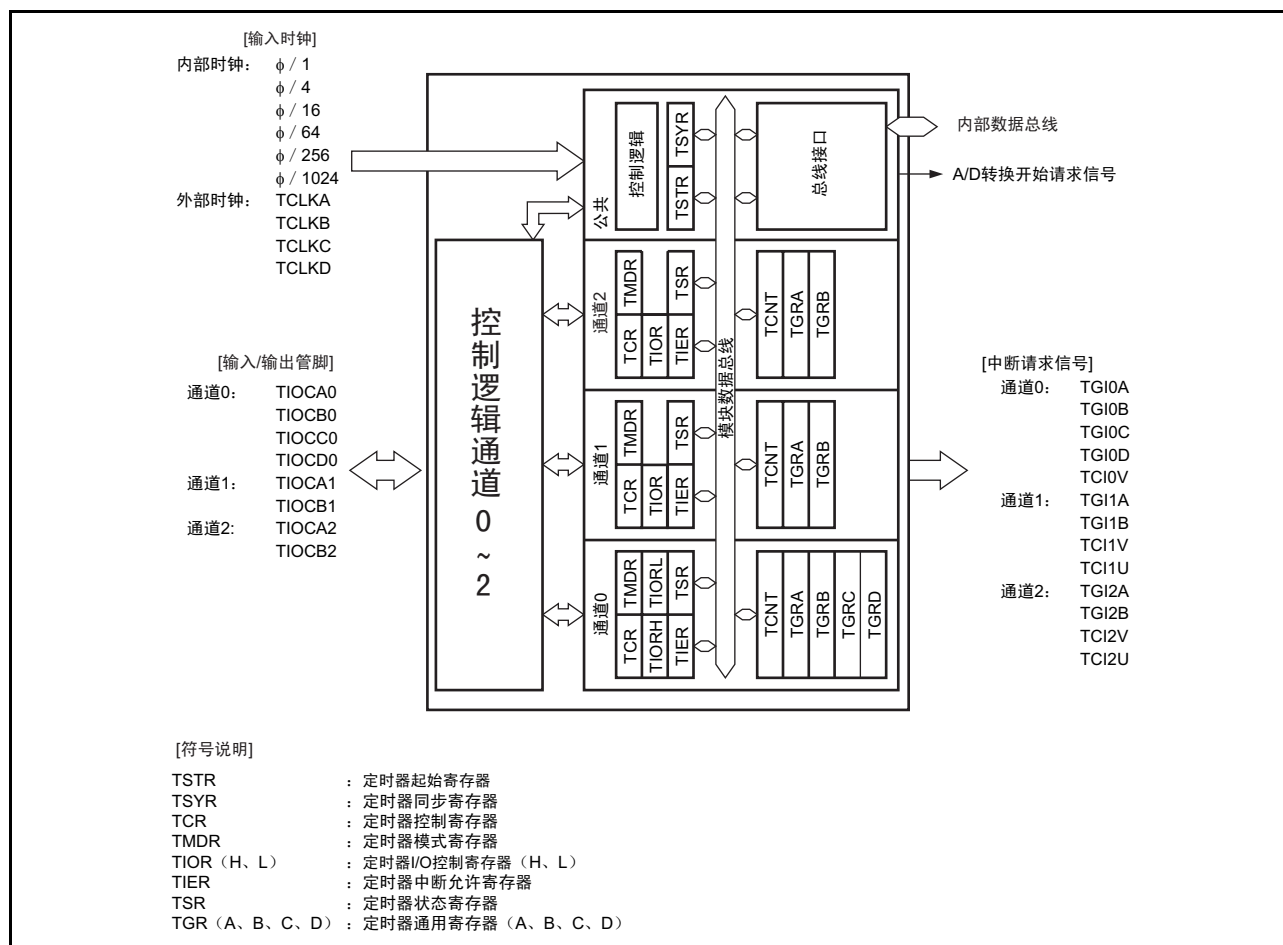


图 8.1 TPU 的时钟框图

8.2 输入 / 输出管脚

表 8.2 管脚构成表

通道	名称	输入 / 输出	功 能
公共	TCLKA	输入	外部时钟 A 输入管脚 (通道 1 的相位计数模式下 A 相输入)
	TCLKB	输入	外部时钟 B 输入管脚 (通道 1 的相位计数模式下 B 相输入)
	TCLKC	输入	外部时钟 C 输入管脚 (通道 2 的相位计数模式下 A 相输入)
	TCLKD	输入	外部时钟 D 输入管脚 (通道 2 的相位计数模式下 B 相输入)
0	TIOCA0	输入 / 输出	TGRA_0 输入捕捉的输入管脚 / TGRA_0 输出比较的输出管脚 / PWM 输出管脚
	TIOCB0	输入 / 输出	TGRB_0 输入捕捉的输入管脚 / TGRB_0 输出比较的输出管脚 / PWM 输出管脚
	TIOCC0	输入 / 输出	TGRC_0 输入捕捉的输入管脚 / TGRC_0 输出比较的输出管脚 / PWM 输出管脚
	TIOCD0	输入 / 输出	TGRD_0 输入捕捉的输入管脚 / TGRD_0 输出比较的输出管脚 / PWM 输出管脚
1	TIOCA1	输入 / 输出	TGRA_1 输入捕捉的输入管脚 / TGRA_1 输出比较的输出管脚 / PWM 输出管脚
	TIOCB1	输入 / 输出	TGRB_1 输入捕捉的输入管脚 / TGRB_1 输出比较的输出管脚 / PWM 输出管脚
2	TIOCA2	输入 / 输出	TGRA_2 输入捕捉的输入管脚 / TGRA_2 输出比较的输出管脚 / PWM 输出管脚
	TIOCB2	输入 / 输出	TGRB_2 输入捕捉的输入管脚 / TGRB_2 输出比较的输出管脚 / PWM 输出管脚

8.3 寄存器说明

TPU 各通道有以下寄存器。各通道的寄存器名、通道 0 的 TCR 记为 TCR_0。

- 定时器控制寄存器 _0(TCR_0)
- 定时器模式寄存器 _0(TMDR_0)
- 定时器 I/O 控制寄存器 H_0 (TIORH_0)
- 定时器 I/O 控制寄存器 L_0 (TIORL_0)
- 定时器中断允许寄存器 _0(TIER_0)
- 定时器状态寄存器 _0(TSR_0)
- 定时器计数器 _0(TCNT_0)
- 定时器通用寄存器 A_0(TGRA_0)
- 定时器通用寄存器 B_0(TGRB_0)
- 定时器通用寄存器 C_0(TGRC_0)
- 定时器通用寄存器 D_0(TGRD_0)
- 定时器控制寄存器 _1(TCR_1)
- 定时器模式寄存器 _1(TMDR_1)
- 定时器 I/O 控制寄存器 _1(TIOR_1)
- 定时器中断允许寄存器 _1(TIER_1)
- 定时器状态寄存器 _1(TSR_1)
- 定时器计数器 _1(TCNT_1)
- 定时器通用寄存器 A_1(TGRA_1)
- 定时器通用寄存器 B_1(TGRB_1)
- 定时器控制寄存器 _2(TCR_2)
- 定时器模式寄存器 _2(TMDR_2)
- 定时器 I/O 控制寄存器 _2(TIOR_2)
- 定时器中断允许寄存器 _2(TIER_2)
- 定时器状态寄存器 _2(TSR_2)
- 定时器计数器 _2(TCNT_2)
- 定时器通用寄存器 A_2(TGRA_2)
- 定时器通用寄存器 B_2(TGRB_2)

通用寄存器

- 定时器启动寄存器 (TSTR)
- 定时器同步寄存器 (TSYR)

8.3.1 定时器控制寄存器 (TCR)

TCR 是控制各通道 TCNT 的寄存器。TPU 各通道各有 1 个，共有 3 个 TCR。TCR 的设定是在 TCNT 工作停止状态下进行。

位	位名	初始值	R/W	说 明
7	CCLR2	0	R/W	计数器清除 2 ~ 0 选择 TCNT 的计数器清除源。详细请参照表 8.3、表 8.4。
6	CCLR1	0	R/W	
5	CCLR0	0	R/W	
4	CKEG1	0	R/W	时钟边沿 1、0 选择输入时钟的边沿。内部时钟用双边沿计数后，输入时钟的周期成为 1/2（例： $\phi / 4$ 的双边沿 = $\phi / 2$ 的上升沿）。通道 1、2 使用相位计数模式时，这个设定被忽视，优先相位计数模式的设定。选择内部时钟的边沿时，输入时钟在 $\phi / 4$ 或更迟时有效。在输入时钟选择 $\phi / 1$ 或其它的通道溢出 / 下溢时，该设定被忽视。 00: 在上升沿计数 01: 在下降沿计数 1x: 在双边沿计数
3	CKEG0	0	R/W	
2	TPSC2	0	R/W	定时预定标器 2 ~ 0 选择 TCNT 的计数器时钟，能选择各通道独立的时钟源。详细请参照表 8.5 ~ 表 8.7。
1	TPSC1	0	R/W	
0	TPSC0	0	R/W	

【符号说明】 x: Don't care

表 8.3 CCLR2 ~ CCLR0 (通道 0)

通道	位 7	位 6	位 5	说 明
	CCLR2	CCLR1	CCLR0	
0	0	0	0	禁止清除 TCNT
	0	0	1	通过 TGRA 的比较匹配 / 输出捕捉清除 TCNT
	0	1	0	通过 TGRB 的比较匹配 / 输出捕捉清除 TCNT
	0	1	1	同步清除。与正在同步工作的其它通道的计数器清除同步清除 TCNT*1
	1	0	0	禁止清除 TCNT
	1	0	1	通过 TGRC 的比较匹配 / 输出捕捉清除 TCNT*2
	1	1	0	通过 TGRD 的比较匹配 / 输出捕捉清除 TCNT*2
	1	1	1	同步清除。与正在同步工作的其它的通道的计数器清除同步清除 TCNT *1

【注】 *1 通过 TSYR 的 SYNC 位置 1，决定同步工作的设定。

*2 TGRC 及 TGRD 作为缓冲寄存器使用时，优先缓冲寄存器的设定，因不产生比较匹配 / 输入捕捉，所以不清除 TCNT。

表 8.4 CCLR2 ~ CCLR0 (通道 1、2)

通道	位 7	位 6	位 5	说 明
	保留 *2	CCLR1	CCLR0	
1、2	0	0	0	禁止清除 TCNT
	0	0	1	通过 TGRA 的比较匹配 / 输出捕捉清除 TCNT
	0	1	0	通过 TGRB 的比较匹配 / 输出捕捉清除 TCNT
	0	1	1	同步清除。与正在同步工作的其它通道的计数器清除同步清除 TCNT*1

【注】 *1 通过 TSYR 的 SYNC 位置 1，决定同步工作的设定。

*2 在通道 1，2 中，位 7 为保留位。总是读出 0。写无效。

表 8.5 TPSC2 ~ TPSC0 (通道 0)

通道	位 2	位 1	位 0	说 明
	TPSC2	TPSC1	TPSC0	
0	0	0	0	内部时钟: 用 ϕ / 1 分频
	0	0	1	内部时钟: 用 ϕ / 4 分频
	0	1	0	内部时钟: 用 ϕ / 16 分频
	0	1	1	内部时钟: 用 ϕ / 64 分频
	1	0	0	外部时钟: 用 TCLKA 管脚输入
	1	0	1	外部时钟: 用 TCLKB 管脚输入
	1	1	0	外部时钟: 用 TCLKC 管脚输入
	1	1	1	外部时钟: 用 TCLKD 管脚输入

表 8.6 TPSC2 ~ TPSC0 (通道 1)

通道	位 2	位 1	位 0	说 明
	TPSC2	TPSC1	TPSC0	
1	0	0	0	内部时钟: 用 ϕ / 1 分频
	0	0	1	内部时钟: 用 ϕ / 4 分频
	0	1	0	内部时钟: 用 ϕ / 16 分频
	0	1	1	内部时钟: 用 ϕ / 64 分频
	1	0	0	外部时钟: 用 TCLKA 管脚输入
	1	0	1	外部时钟: 用 TCLKB 管脚输入
	1	1	0	内部时钟: 用 ϕ / 256 分频
	1	1	1	禁止设定

【注】: 通道 1 在相位计数模式时, 此设定无效。

表 8.7 TPSC2 ~ TPSC0 (通道 2)

通道	位 2	位 1	位 0	说 明
	TPSC2	TPSC1	TPSC0	
2	0	0	0	内部时钟: 用 ϕ / 1 分频
	0	0	1	内部时钟: 用 ϕ / 4 分频
	0	1	0	内部时钟: 用 ϕ / 16 分频
	0	1	1	内部时钟: 用 ϕ / 64 分频
	1	0	0	外部时钟: 用 TCLKA 管脚输入
	1	0	1	外部时钟: 用 TCLKB 管脚输入
	1	1	0	外部时钟: 用 TCLKC 管脚输入
	1	1	1	内部时钟: 用 ϕ / 1024 分频

【注】: 通道 2 在相位计数模式时, 此设定无效。

8.3.2 计时器模式寄存器 (TMDR)

TMDR 是设定各通道的工作模式。TPU 下各通道一个, 共计 3 个 TMDR, TMDR 的设定在 TCNT 的工作停止状态时进行。

位	位名	初始值	R/W	说 明
7、6	—	全为 1	—	保留 总是读出 1, 写无效。
5	BFB	0	R/W	缓冲工作 B 设定 TGRB 正常工作或 TGRB 和 TGRD 组合缓冲工作。 TGRD 作为缓冲寄存器使用时, 不产生 TGRD 的输入捕捉 / 输出比较。 不具有 TGRD 的通道 1、2, 此位成为保留位。总是读出 0。写无效。 0: TGRB 正常工作 1: TGRB 与 TGRD 是缓冲工作
4	BFA	0	R/W	缓冲工作 A 设定 TGRA 正常工作或 TGRA 与 TGRC 组合缓冲器工作。 TGRC 作为缓冲器寄存器使用时, 不产生 TGRC 的输入捕捉 / 输出比较。 不具有 TGRC 的通道 1、2, 此位成为保留位。总是读出 0。写无效。 0: TGRA 正常工作 1: TGRA 与 TGRC 是缓冲器工作
3	MD3	0	R/W	模式 3 ~ 0 MD3 ~ MD0 设定为计时器的工作模式。 MD3 是保留位。总是写入 0。 详细请参照表 8.8。
2	MD2	0	R/W	
1	MD1	0	R/W	
0	MD0	0	R/W	

表 8.8 MD3 ~ MD0

位 3	位 2	位 1	位 0	说 明
MD3*1	MD2*2	MD1	MD0	
0	0	0	0	正常工作
0	0	0	1	保留
0	0	1	0	PWM 模式 1
0	0	1	1	PWM 模式 2
0	1	0	0	相位计数模式 1
0	1	0	1	相位计数模式 2
0	1	1	0	相位计数模式 3
0	1	1	0	相位计数模式 4
1	×	×	×	—

【符号说明】×: Don't care

【注】 *1 MD3 是保留位。总是写入 0。

*2 在通道 0，不能设定相位计数模式。向 MD2 总是写入 0。

8.3.3 定时器 I/O 控制寄存器 (TIOR)

TIOR 是控制 TGR 的寄存器。TPU 的通道 0 有 2 个，通道 1、2 各有 1 个，共有 4 个 TIOR。

请注意，TIOR 受 TMDR 设置影响。

在计数器停止状态时，TIOR 指定的初始输出（TSTR 的 CST 位清除为 0）有效。且在 PWM 模式 2，指定计数器在清 0 时输出。

设定 TGRC 或 TGRD 缓冲工作时，此设定无效，作为缓冲寄存器工作。

• TIORH_0、TIOR_1、TIOR_2

位	位名	初始值	R/W	说 明
7	IOB3	0	R/W	I/O 控制 B3 ~ B0 设定 TGRB 的功能。
6	IOB2	0	R/W	
5	IOB1	0	R/W	
4	IOB0	0	R/W	
3	IOA3	0	R/W	I/O 控制 A3 ~ A0 设定 TGRA 的功能。
2	IOA2	0	R/W	
1	IOA1	0	R/W	
0	IOA0	0	R/W	

• TIORL_0

位	位名	初始值	R/W	说 明
7	IOD3	0	R/W	I/O 控制 D3 ~ D0 设定 TGRD 的功能。
6	IOD2	0	R/W	
5	IOD1	0	R/W	
4	IOD0	0	R/W	
3	IOC3	0	R/W	I/O 控制 C3 ~ C0 设定 TGRC 的功能。
2	IOC2	0	R/W	
1	IOC1	0	R/W	
0	IOC0	0	R/W	

表 8.9 TIORH_0

位 7	位 6	位 5	位 4	说 明	
IOB3	IOB2	IOB1	IOB0	TGRB_0 的功能	TIOCB0 管脚的功能
0	0	0	0	输出比较寄存器	禁止输出
0	0	0	1		初始输出为 0 通过比较匹配输出 0
0	0	1	0		初始输出为 0 通过比较匹配输出 1
0	0	1	1		初始输出为 0 通过比较匹配交替输出
0	1	0	0		禁止输出
0	1	0	1		初始输出为 1 通过比较匹配输出 0
0	1	1	0		初始输出为 1 通过比较匹配输出 1
0	1	1	1		初始输出为 1 通过比较匹配交替输出
1	0	0	0	输入捕捉寄存器	捕捉输入源是 TIOCB0 管脚 通过上升沿输入捕捉
1	0	0	1		捕捉输入源是 TIOCB0 管脚 通过下降沿输入捕捉
1	0	1	×		捕捉输入源是 TIOCB0 管脚 通过双边沿输入捕捉
1	1	×	×		捕捉输入源是通道 1 / 计数器时钟 通过 TCNT_1 的累加计数 / 递减计数 输入捕捉 *

【符号说明】×: Don't care

【注】 *TCR_1 的 TPSC2 ~ TPSC0 位作为 B'000、TCNT_1 的计数时钟使用 $\phi / 1$ 时, 本设定无效, 不产生输入捕捉。

表 8.10 TIORL_0

位 7	位 6	位 5	位 4	说 明	
IOD3	IOD2	IOD1	IOD0	TGRD_0 的功能	TIOCD0 管脚的功能
0	0	0	0	输出比较 寄存器 *2	禁止输出
0	0	0	1		初始输出为 0 通过比较匹配输出 0
0	0	1	0		初始输出为 0 通过比较匹配输出 1
0	0	1	1		初始输出为 0 通过比较匹配交替输出
0	1	0	0		禁止输出
0	1	0	1		初始输出为 1 通过比较匹配输出 0
0	1	1	0		初始输出为 1 通过比较匹配输出 1
0	1	1	1		初始输出为 1 通过比较匹配交替输出
1	0	0	0	输入捕捉 寄存器 *2	捕捉输入源是 TIOCD0 管脚 通过上升沿输入捕捉
1	0	0	1		捕捉输入源是 TIOCD0 管脚 通过下降沿输入捕捉
1	0	1	×		捕捉输入源是 TIOCD0 管脚 通过双边沿输入捕捉
1	1	×	×		捕捉输入源是通道 1 / 计数器时钟 通过 TCNT_1 的累加计数 / 递减计数 输入捕捉 *1

【符号说明】×：Don't care

【注】 *1 TCR_1 的 TPSC2 ~ TPSC0 位作为 B'000，TCNT_1 的计数时钟使用 $\phi / 1$ 时，本设定无效，不产生输入捕捉。

*2 TMDR_0 的 BFB 位置 1，TGRD_0 作为缓冲寄存器使用时，本设定无效，不产生输入捕捉 / 输出比较。

表 8.11 TIOR_1

位 7	位 6	位 5	位 4	说 明	
IOB3	IOB2	IOB1	IOB0	TGRB_1 的功能	TGRB1 管脚的功能
0	0	0	0	输出比较寄存器	禁止输出
0	0	0	1		初始输出为 0 通过比较匹配输出 0
0	0	1	0		初始输出为 0 通过比较匹配输出 1
0	0	1	1		初始输出为 0 通过比较匹配交替输出
0	1	0	0		禁止输出
0	1	0	1		初始输出为 1 通过比较匹配输出 0
0	1	1	0		初始输出为 1 通过比较匹配输出 1
0	1	1	1		初始输出为 1 通过比较匹配交替输出
1	0	0	0	输入捕捉寄存器	捕捉输入源是 TIOCB1 管脚 通过上升沿输入捕捉
1	0	0	1		捕捉输入源是 TIOCB1 管脚 通过下降沿输入捕捉
1	0	1	×		捕捉输入源是 TIOCB1 管脚 通过双边沿输入捕捉
1	1	×	×		TGRC_0 比较匹配 / 输入捕捉 通过 TGRC_0 比较匹配 / 输入捕捉的 产生来输入捕捉

【符号说明】×: Don't care

表 8.12 TIOR_2

位 7	位 6	位 5	位 4	说 明	
IOB3	IOB2	IOB1	IOB0	TGRB_2 的功能	TGRB2 管脚的功能
0	0	0	0	输出比较寄存器	禁止输出
0	0	0	1		初始输出为 0 通过比较匹配输出 0
0	0	1	0		初始输出为 0 通过比较匹配输出 1
0	0	1	1		初始输出为 0 通过比较匹配交替输出
0	1	0	0		禁止输出
0	1	0	1		初始输出为 1 通过比较匹配输出 0
0	1	1	0		初始输出为 1 通过比较匹配输出 1
0	1	1	1		初始输出为 1 通过比较匹配交替输出
1	×	0	0	输入捕捉寄存器	捕捉输入源是 TIOCB2 管脚 通过上升沿输入捕捉
1	×	0	1		捕捉输入源是 TIOCB2 管脚 通过下降沿输入捕捉
1	×	1	×		捕捉输入源是 TIOCB2 管脚 通过双边沿输入捕捉

【符号说明】×: Don't care

表 8.13 TIORH_0

位 3	位 2	位 1	位 0	说 明	
IOA3	IOA2	IOA1	IOA0	TGRA_0 的功能	TIOCA0 管脚的功能
0	0	0	0	输出比较寄存器	禁止输出
0	0	0	1		初始输出为 0 通过比较匹配输出 0
0	0	1	0		初始输出为 0 通过比较匹配输出 1
0	0	1	1		初始输出为 0 通过比较匹配交替输出
0	1	0	0		禁止输出
0	1	0	1		初始输出为 1 通过比较匹配输出 0
0	1	1	0		初始输出为 1 通过比较匹配输出 1
0	1	1	1		初始输出为 1 通过比较匹配交替输出
1	0	0	1	输入捕捉寄存器	捕捉输入源是 TIOCA0 管脚 通过下降沿输入捕捉
1	0	0	0		捕捉输入源是 TIOCA0 管脚 通过上升沿输入捕捉
1	0	1	×		捕捉输入源是 TIOCA0 管脚 通过双边沿输入捕捉
1	1	×	×		捕捉输入源是通道 1 / 计数时钟 通过 TCNT_1 的累加计数 / 追减计数 输入捕捉

【符号说明】×: Don't care

表 8.14 TIORL_0

位 3	位 2	位 1	位 0	说 明	
IOC3	IOC2	IOC1	IOC0	TGRC_0 的功能	TIOCC0 管脚的功能
0	0	0	0	输出比较 寄存器 *	禁止输出
0	0	0	1		初始输出为 0 通过比较匹配输出 0
0	0	1	0		初始输出为 0 通过比较匹配输出 1
0	0	1	1		初始输出为 0 通过比较匹配交替输出
0	1	0	0		禁止输出
0	1	0	1		初始输出为 1 通过比较匹配输出 0
0	1	1	0		初始输出为 1 通过比较匹配输出 1
0	1	1	1		初始输出为 1 通过比较匹配交替输出
1	0	0	0	输入捕捉 寄存器 *	捕捉输入源是 TIOCC0 管脚 通过上升沿输入捕捉
1	0	0	1		捕捉输入源是 TIOCC0 管脚 通过下降沿输入捕捉
1	0	1	×		捕捉输入源是 TIOCC0 管脚 通过双边沿输入捕捉
1	1	×	×		捕捉输入源是通道 1 / 计数时钟 通过 TCNT_1 的累加计数 / 追减计数 输入捕捉

【符号说明】×: Don't care

【注】 *TMDR_0 的 BFA 位置 1, TGRC_0 作为缓冲寄存器使用时, 本设定无效, 不产生输入捕捉 / 输出比较。

表 8.15 TIOR_1

位 3	位 2	位 1	位 0	说 明	
IOA3	IOA2	IOA1	IOA0	TGRA_1 的功能	TIOCA1 管脚的功能
0	0	0	0	输出比较寄存器	禁止输出
0	0	0	1		初始输出为 0 通过比较匹配输出 0
0	0	1	0		初始输出为 0 通过比较匹配输出 1
0	0	1	1		初始输出为 0 通过比较匹配交替输出
0	1	0	0		禁止输出
0	1	0	1		初始输出为 1 通过比较匹配输出 0
0	1	1	0		初始输出为 1 通过比较匹配输出 1
0	1	1	1		初始输出为 1 通过比较匹配交替输出
1	0	0	0	输入捕捉寄存器	捕捉输入源是 TIOCA1 管脚 通过上升沿输入捕捉
1	0	0	1		捕捉输入源是 TIOCA1 管脚 通过下降沿输入捕捉
1	0	1	×		捕捉输入源是 TIOCA1 管脚 通过双边沿输入捕捉
1	1	×	×		捕捉输入源是 TGRA_0 比较匹配 / 输入捕捉 通过通道 0 / TGRA_0 的比较匹配 / 输入捕捉的产生来输入捕捉

【符号说明】×: Don't care

表 8.16 TIOR_2

位 7	位 6	位 5	位 4	说 明	
IOA3	IOA2	IOA1	IOA0	TGRA_2 的功能	TGRA2 管脚的功能
0	0	0	0	输出比较寄存器	禁止输出
0	0	0	1		初始输出为 0 通过比较匹配输出 0
0	0	1	0		初始输出为 0 通过比较匹配输出 1
0	0	1	1		初始输出为 0 通过比较匹配交替输出
0	1	0	0		禁止输出
0	1	0	1		初始输出为 1 通过比较匹配输出 0
0	1	1	0		初始输出为 1 通过比较匹配输出 1
0	1	1	1		初始输出为 1 通过比较匹配交替输出
1	×	0	0	输入捕捉寄存器	捕捉输入源是 TGRA2 管脚 通过上升沿输入捕捉
1	×	0	1		捕捉输入源是 TGRA2 管脚 通过下降沿输入捕捉
1	×	1	×		捕捉输入源是 TGRA2 管脚 通过双边沿输入捕捉

【符号说明】×: Don't care

8.3.4 定时器中断允许寄存器 (TIER)

TIER 控制各通道的中断请求、禁止。TPU 各通道有 1 个，共有 3 个 TIER。

位	位名	初始值	R/W	说 明
7	TTGE	0	R/W	允许 A/D 转换开始请求 通过 TGRA 的输入捕捉 / 比较匹配，禁止或允许产生 A/D 转换开始请求。 0: 禁止产生 A/D 转换开始请求 1: 允许产生 A/D 转换开始要求
6	—	1	—	保留位 总是读出 1。不可修改
5	TCIEU	0	R/W	允许下溢中断 在通道 1 和通道 2，TSR 的 TCFU 标志置 1 时，TCFU 标志可禁止或允许中断请求 (TCIU)。 通道 0 是保留位。 总是读出 0，不可修改 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	允许溢出中断 TSR 的 TCFV 标志置 1 时，TCFV 标志可禁止或允许中断请求 (TCIV)。 0: 禁止 TCFV 的中断请求 1: 允许 TCFV 的中断请求
3	TGIED	0	R/W	TGR 允许中断 D 在通道 0，TSR 的 TGFD 位置 1 时，TGFD 位可禁止或允许中断要求 (TGID)。通道 1 和通道 2 是保留位。总是读出 0。不可修改。 0: 禁止 TGFD 的中断请求 1: 允许 TGFD 的中断请求
2	TGIEC	0	R/W	TGR 允许中断 C 通道 0 在 TSR 的 TGFC 位置 1 时，TGFC 位可禁止或允许中断要求 (TGIC)。 通道 1 和通道 2 是保留位。总是读出 0。不可修改。 0: 禁止 TGFC 的中断请求 1: 允许 TGFC 的中断请求
1	TGIEB	0	R/W	TGR 允许中断 B TSR 的 TGFB 位置 1 时，TGFB 位可禁止或允许中断要求 (TGIB)。 0: 禁止 TGFB 的中断请求 1: 允许 TGFB 的中断请求
0	TGIEA	0	R/W	TGR 允许中断 A TSR 的 TGFA 位置 1 时，TGFA 位可禁止或允许中断要求 (TGIA)。 0: 禁止 TGFA 的中断请求 1: 允许 TGFA 的中断请求

8.3.5 定时器状态寄存器 (TSR)

TSR 表示各通道的状态，TPU 各通道有 1 个，共有 3 个 TSR。

位	位名	初始值	R/W	说 明
7	TCFD	1	R	计数方向标志 表示通道 1 和通道 2 的 TCNT 计数方向的状态标志。 通道 0 是保留位。总是读出 1。不可修改。 0: TCNT 是递减计数 1: TCNT 是累加计数
6	—	1	—	保留位 总是读出 1。不可修改。
5	TCFU	0	R/(W)*	下溢标志 通道 1 和通道 2 在相位计数模式时，表示 TCNT 的下溢产生的状态标志。 通道 0 是保留位。总是读出 0。不可修改。 [置位条件] TCNT 值下溢时 (H'0000→H'FFFF)。 [清除条件] 在 TCFU=1 的状态，读 TCFU 后，向 TCFU 写 0 时。
4	TCFV	0	R/(W)*	溢出标志 (TCFV) 表示 TCNT 的溢出产生的状态标志。 [置位条件] TCNT 值溢出时 (H'FFFF→H'0000)。 [清除条件] 在 TCFV=1 的状态，读 TCFV 后，向 TCFV 写 0 时。
3	TGFD	0	R/(W)*	输入捕捉 / 输出比较标志 D 表示通道 0 的 TGRD 输入捕捉或比较匹配产生的状态标志。 通道 1 和通道 2 是保留位。总是读出 0。不可修改。 [置位条件] 当 TGRD 作为输出比较寄存器功能使用，并且 TCNT=TGRD 时。 当 TGRD 作为输入捕捉功能使用，通过输入捕捉信号，将 TCNT 的值传送到 TGRD 时。 [清除条件] 在 TGFD=1 的状态，读 TGFD 后，向 TGFD 写 0 时。
2	TGFC	0	R/(W)*	输入捕捉 / 输出比较标志 C 表示通道 0 的 TGRC 的输入捕捉或比较匹配产生的状态标志。 通道 1 和通道 2 是保留位。总是读出 0。不可修改。 [置位条件] 在 TGRC 作为输出比较寄存器功能使用，TCNT=TGRC 时 在 TGRC 作为输入捕捉功能，通过输入捕捉信号，将 TCNT 的值向 TGRC 传送时。 [清除条件] 在 TGFC=1 状态，读 TGFC 后，向 TGFC 写 0 时。

位	位名	初始值	R/W	说 明
1	TGFB	0	R/(W)*	输入捕捉 / 输出比较标志 B 表示 TGRB 的输入捕捉的或比较匹配产生的状态标志。 [置位条件] 在 TGRB 作为输出比较寄存器功能, TCNT=TGRB 时 在 TGRB 作为输入捕捉的输入管脚功能, 通过输入捕捉信号, 将 TCNT 的值转送到 TGRB 时。 [清除条件] 在 TGFB=1 的状态, 读 TGFB 后, 向 TGFB 写 0 时。
0	TGFA	0	R/(W)*	输入捕捉 / 输出比较标志 A 表示 TGRA 的输入捕捉或比较匹配产生的状态标志。 [置位条件] 在 TGRA 作为输出比较寄存器功能, TCNT=TGRA 时 在 TGRA 作为输入捕捉的输入管脚功能, 通过输入捕捉信号, 将 TCNT 的值传送到 TGRA 时。 [清除条件] 在 TGFA=1 状态, 读 TGFA 后, 向 TGFA 写 0 时。

【注】 * 为清除标志, 仅能写 0。

8.3.6 定时器计数器 (TCNT)

TCNT 是 16 位的可读 / 写计数器。各通道有 1 个, 共有 3 个 TCNT。

TCNT 在复位或硬件待机模式时, 初始化为 H'0000。

禁止用 TCNT 的 8 位单位存取。一般用 16 位单位存取。

8.3.7 定时器通用寄存器 (TGR)

TGR 是 16 位可读 / 写, 兼用输出比较 / 输入捕捉的寄存器。通道 0 有 4 个, 通道 1、通道 2 各有 2 个, 共有 8 个通用寄存器。通道 0 的 TGRC 和 TGRD 能作为缓冲寄存器进行工作设定。禁止用 TGR 的 8 位存取。一般用 16 位存取。TGR 与缓冲寄存器的组合为 TGRA-TGRC、TGRB-TGRD。

8.3.8 定时器起始寄存器 (TSTR)

TSTR 选择通道 0 ~ 2 的 TCNT 工作 / 停止。

在 TMDR 设定工作模式或在 TCR 设定 TCNT 的计数时钟时, 请在停止 TCNT 的计数器工作后进行。

位	位名	初始值	R/W	说 明
7 ~ 3	—	全部是 0	—	保留位 总是写入 0。
2 1 0	CST2 CST1 CST0	0	R/W	计数器起始 2 ~ 0 选择 TCNT 的工作或停止。 在 TIOC 管脚输出状态工作中, 向 CST 位写 0 后计数器停止, 但保持 TIOC 管脚的输出比较的输出电平。CST 位在 0 状态, 写入 TIOR 后, 设定的初始输出值更新为管脚的输出电平。 0: 停止 TCNT_2 ~ TCNT_0 计数工作 1: TCNT_2 ~ TCNT_0 计数工作

8.3.9 定时器同步寄存器 (TSYR)

TSYR 选择通道 0 ~ 2 的 TCNT 单独工作或同步工作。对应位置 1 的通道同时工作。

位	位名	初始值	R/W	说 明
7 ~ 3	—	全部是 0	R/W	保留位 总是写入 0。
2 1 0	SYNC2 SYNC1 SYNC0	0	R/W	定时器同步 2 ~ 0 选择与其它通道单独工作或同步工作。 在选择同步工作后, 通过复数 TCNT 同步预置、其它通道的计数器清除, 能同步清除。 设定为同步工作, 需要最低 2 通道的 SYNC 位置 1。设定为同步清除, 需要通过 SYNC 位以外 TCR 的 CCLR2 ~ CCLR0 位设定 TCNT 清除原因。 0: TCNT_2 ~ TCNT_0 独立工作 (TCNT 的预置 / 清除与其它通道无关) 1: TCNT_2 ~ TCNT_0 同步工作 TCNT 能同步预置 / 同步清除

8.4 工作说明

8.4.1 基本工作

各个通道都有 TCNT 和 TGR。TCNT 能进行累加计数工作、自由工作、周期计数工作、及事件计数工作。TGR 能分别作为输入捕捉寄存器或输出比较寄存器使用。

(1) 计数器的工作

在 TSTR 的 CST0 ~ CST2 位置 1 后，对应通道的 TCNT 开始计数工作。能实现自由工作计数器工作、周期计数器工作等。

(a) 计数工作的设定步骤的例子

计数工作的设定步骤如图 8.2 所示。

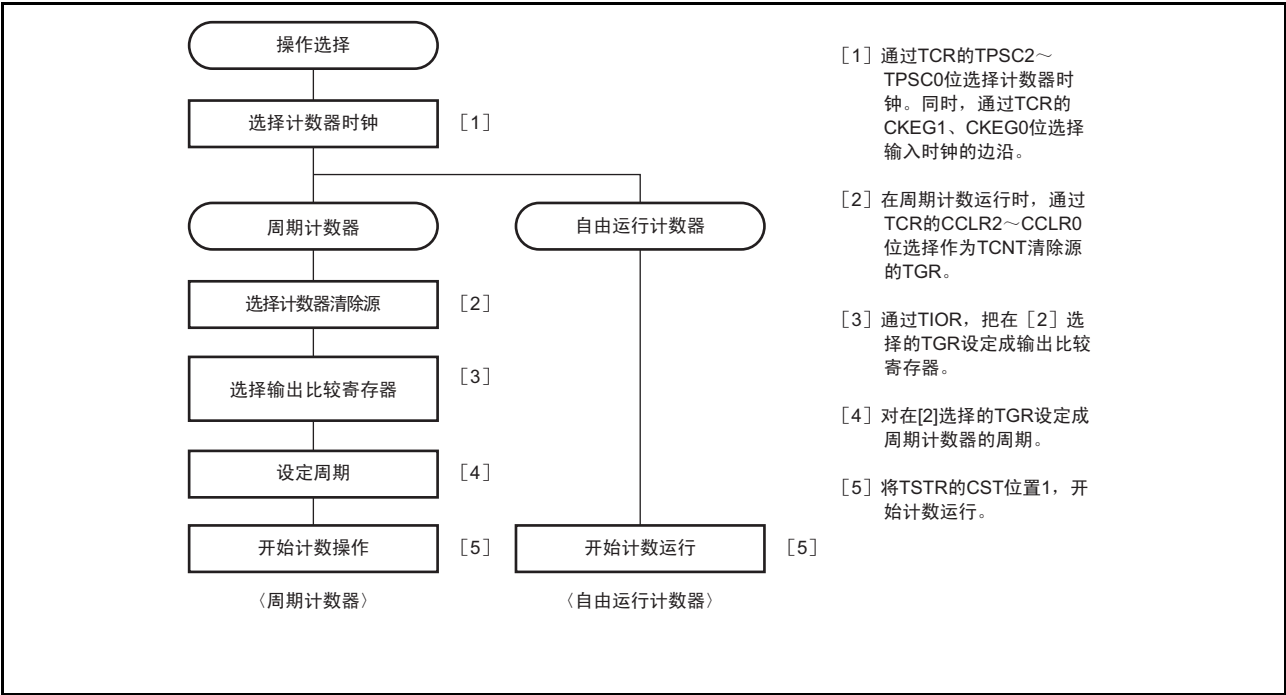


图 8.2 计数器工作设定步骤的例子

(b) 自由工作计数和周期计数

TPU 的 TCNT 复位后, 全部成为自由工作计数器的设定, TSTR 的对应位置 1 后, 作为自由工作计数器开始累加计数工作。TCNT 溢出 (H'FFFF→H'0000) 后, TSR 的 TCFV 位置 1。此时, 对应的 TIER 的 TCIEV 位如果为 1, TPU 请求中断。TCNT 溢出后, 从 H'0000 继续累加计数工作。

自由工作计数器的工作如图 8.3 所示。

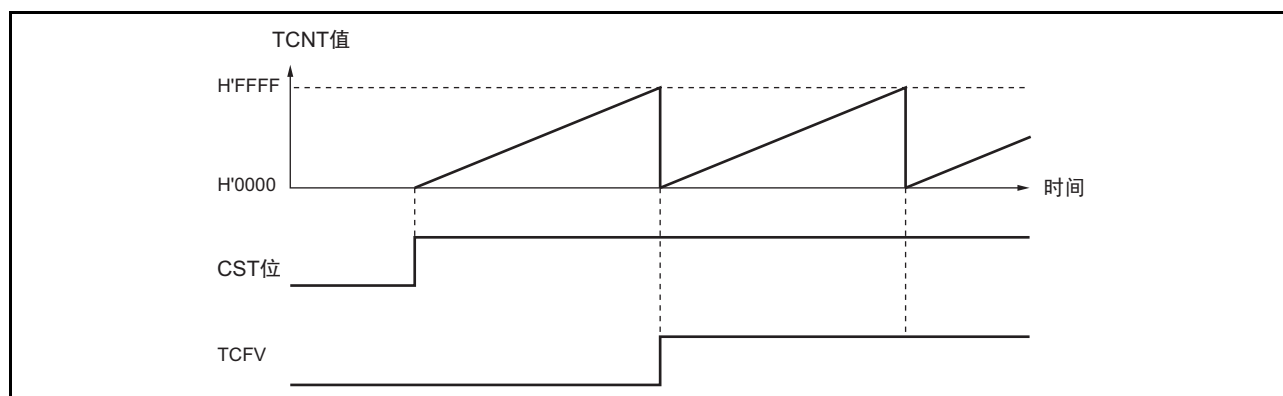


图 8.3 自由工作计数器的工作

TCNT 的清除原因在选择比较匹配时, 对应通道的 TCNT 进行周期计数工作。用于设定周期的 TGR 为输出比较寄存器时, 通过 TCR 的 CCLR2 ~ CCLR0 位、比较匹配来选择计数器清除。设定后, TSTR 对应的位置 1 后, 作为周期计数器开始累加计数工作。计数值与 TGR 值一致后, TSR 的 TGF 位置 1, TCNT 被清为 H'0000。

此时, 对应 TIER 的 TGIE 位如果为 1, TPU 请求中断。TCNT 在比较匹配后, 继续从 H'0000 开始累加计数工作。

周期计数器的工作如图 8.4 所示。

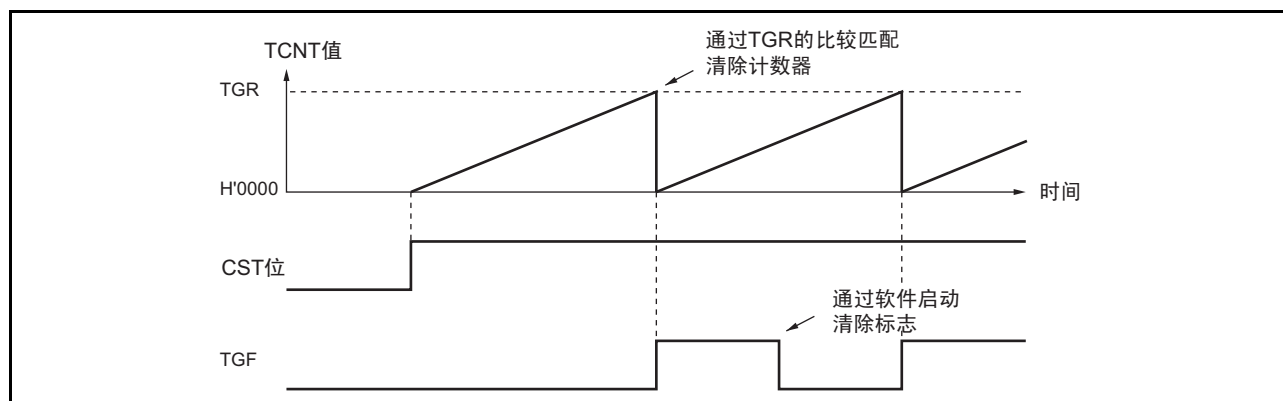


图 8.4 周期寄存器的工作

(2) 通过比较匹配输出波形的功能

TPU 通过比较匹配能从对应的输出管脚进行 0 输出 / 1 输出 / 交替输出。

(a) 通过比较匹配输出波形工作的设定步骤的例子

通过比较匹配输出波形工作的设定步骤的例子如图 8.5 所示。

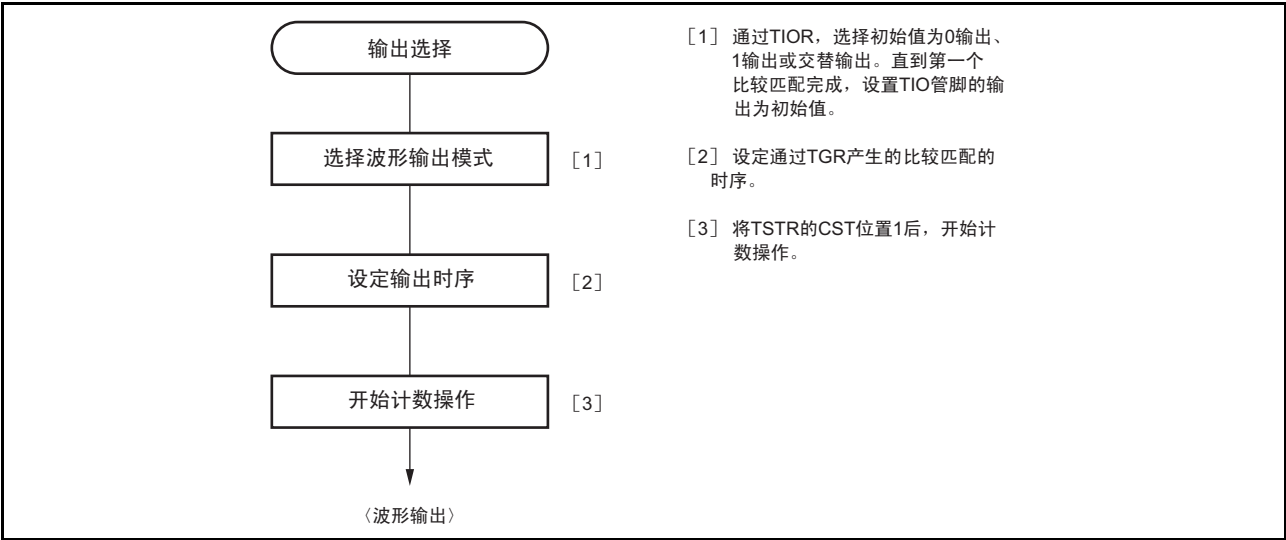


图 8.5 通过比较匹配输出波形工作例子

(b) 输出波形工作例子

0 输出 / 1 输出的例子如图 8.6 所示。

TCNT 进行自由工作计数，设定通过比较匹配 A 输出 1，通过比较匹配 B 输出 0 时的运行时序如图 8.6 所示。另外，设定的电平与管脚电平一致时，管脚电平不发生变化。

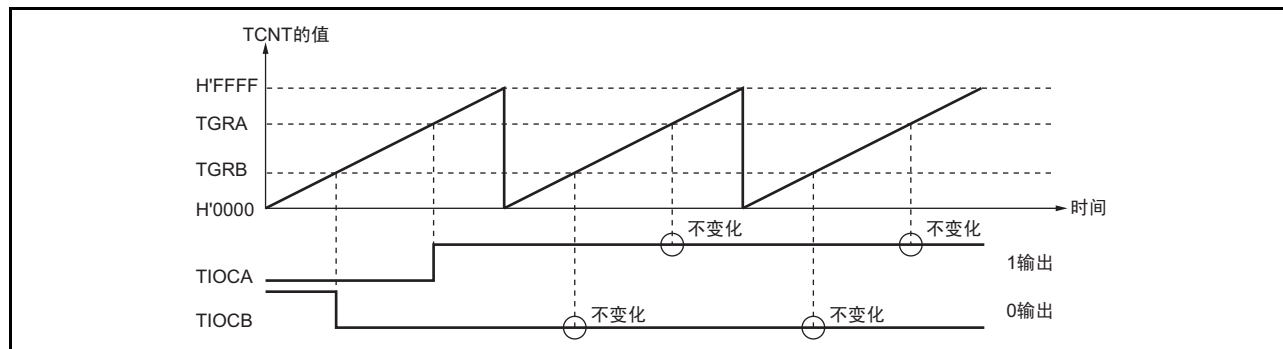


图 8.6 0 输出 / 1 输出的工作例子

交替输出的例子如图 8.7 所示。

TCNT 进行周期计数工作（通过比较匹配 B 清除计数器），比较匹配 A，比较匹配 B 同时都进行交替输出时的工作例子如图 8.14 所示。

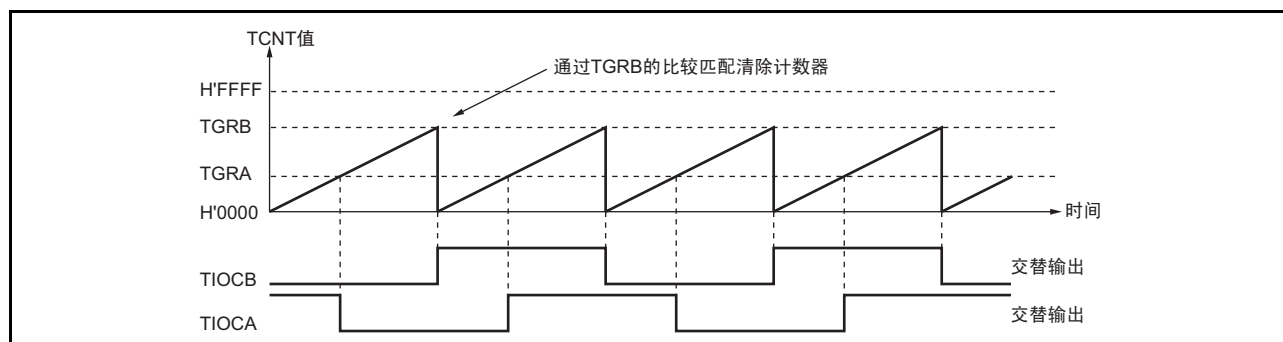


图 8.7 交替输出的工作例子

(3) 输入捕捉功能

能在检测到 TIOC 管脚的输入沿后，把 TCNT 的值传送到 TGR。

检测沿能选择上升沿 / 下降沿 / 两沿。另外，通道 0、通道 1 也是其它通道计数器输入时钟或比较匹配信号的输入捕捉原因。

【注】 通道 0 在其它通道计数器输入时钟成为输入捕捉时，请不要在输入捕捉的计数器输入时钟下选择 $\phi / 1$ ，选择 $\phi / 1$ 时，不产生输入捕捉。

(a) 输入捕捉工作的设定步骤的例子

输入捕捉工作的设定步骤的例子如图 8.8 所示。

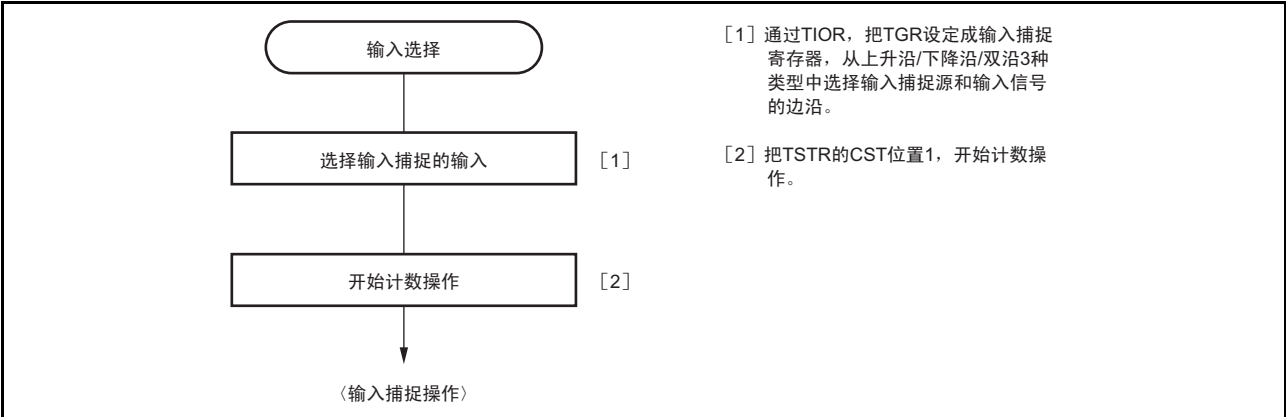


图 8.8 输入捕捉工作的设定例子

(b) 输入捕捉的工作例子

输入捕捉的工作例子如图 8.9 所示。

TIOCA 管脚的输入捕捉的输入边沿选择上升 / 下降两个边沿，TIOCB 管脚的输入捕捉的输入边沿选择下降沿，TCNT 通过 TGRB 的输入捕捉设定清除计数器时的运行时序如图 8.9 所示。

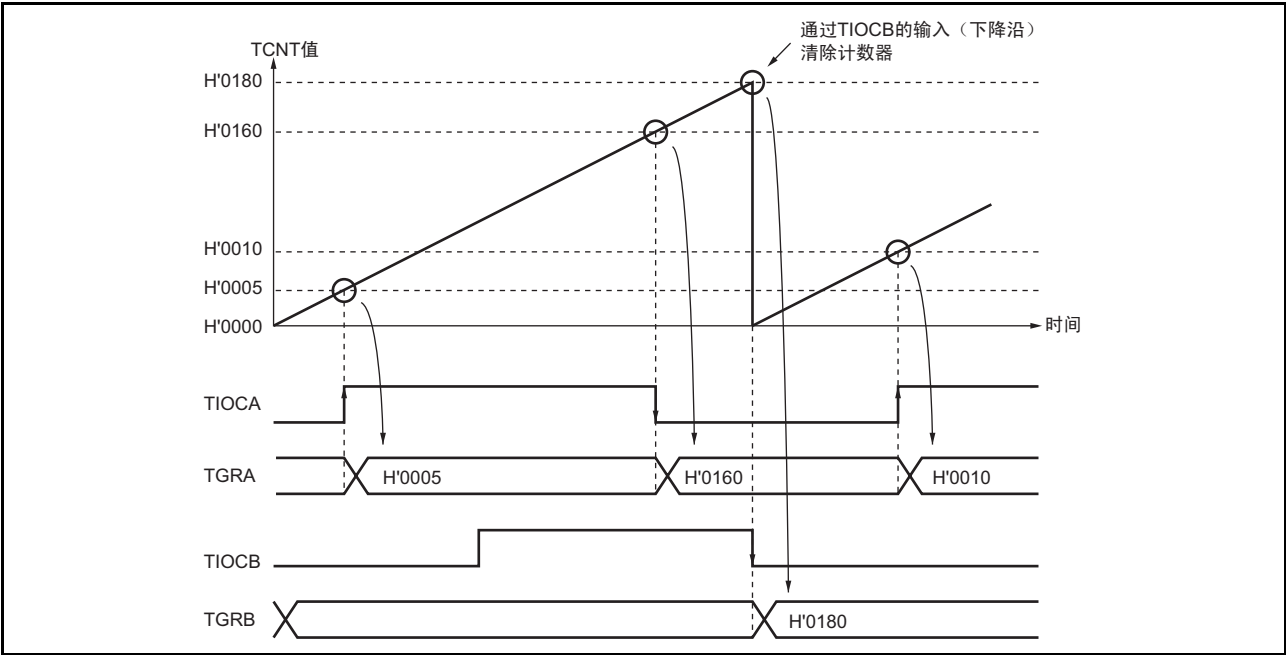


图 8.9 输入捕捉的工作例子

8.4.2 同步工作

同步工作能同时改写多个 TCNT 值 (同步预置)。并且通过设定 TCR, 能同时清除多个 TCNT (同步清除)。

通过同步工作, 能对一个时基增加 TGR。

能实现通道 0 ~ 2 所有同步工作的设定。

(1) 同步工作的设定步骤的例子

同步工作的设定步骤的例子如图 8.10 所示。

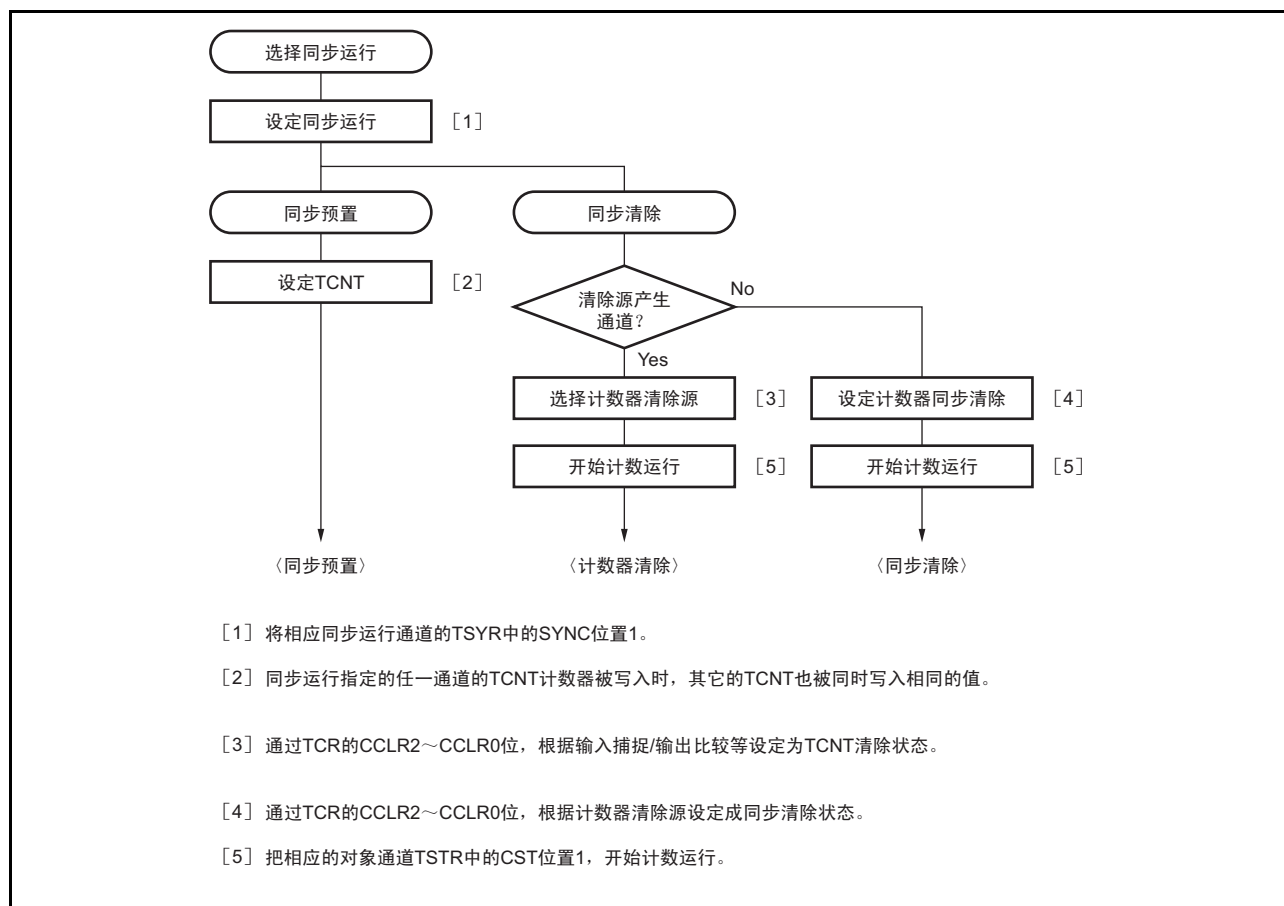


图 8.10 同步工作的设定步骤的例子

(2) 同步工作例子

同步工作例子如图 8.11 所示。

把通道 0 ~ 2 设定成同步工作且 PWM 模式 1，把通道 0 的计数器清除条件设定为 TGRB_0 的比较匹配。并且把通道 1 和通道 2 的计数器清除条件设定为同步清除时的例子如图 8.11 所示。

从 TIOC0A 管脚、TIOC1A 管脚、TIOC2A 管脚输出 3 相的 PWM 波形。此时，通道 0 ~ 2 的 TCNT 进行同步预置和通过 TGRB_0 的比较匹配进行同步清除，TGRB_0 设定的数据成为 PWM 周期。

关于 PWM 模式，请参照「8.4.4 PWM 模式」。

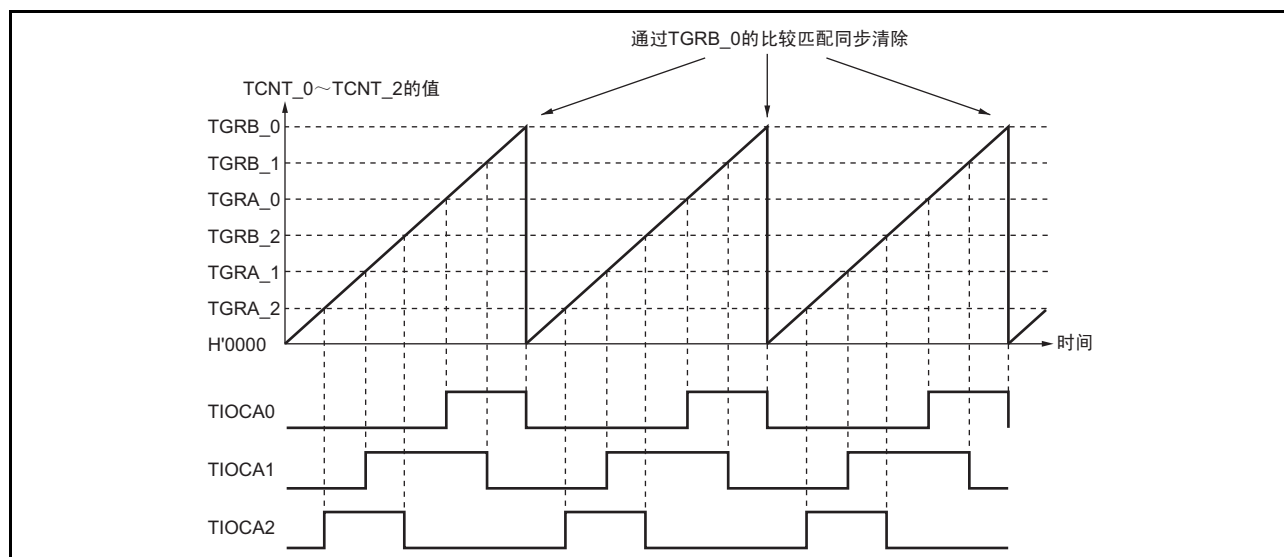


图 8.11 同步工作的工作例

8.4.3 缓冲工作

缓冲工作具有通道 0 功能。TGRC 和 TGRD 能作为缓冲寄存器使用。缓冲工作将 TGR 设定为输入捕捉寄存器和比较匹配寄存器时，各工作内容不同。缓冲工作时的寄存器组合如表 8.17 所示。

表 8.17 寄存器的组合

通道	定时器通用寄存器	缓冲寄存器
0	TGRA_0	TGRC_0
	TGRB_0	TGRD_0

- TGR 为输出比较寄存器时
如果发生比较匹配，就把对应通道的缓冲寄存器的值传送到定时器通用寄存器。此工作如图 8.12 所示。

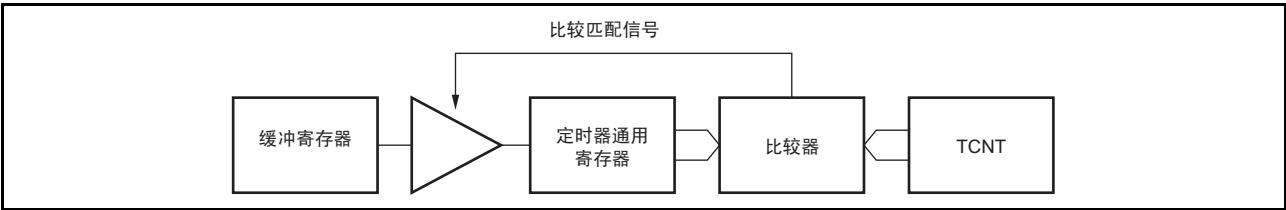


图 8.12 比较匹配缓冲器工作

- TGR 为输入捕捉寄存器时
如果发生输入捕捉，就在 TCNT 的值传送到 TGR 的同时，也将以前保存到 TGR 的值传送到缓冲寄存器。此工作如图 8.13 所示。

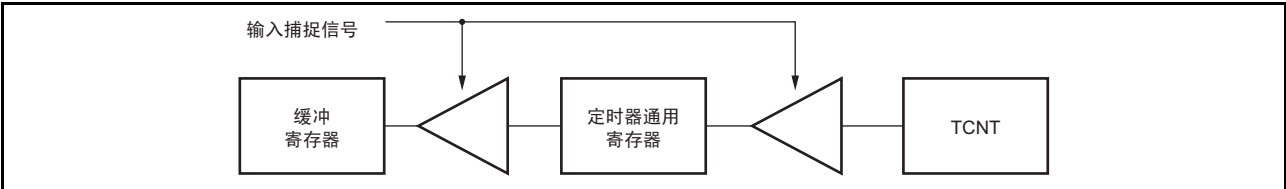


图 8.13 输入捕捉缓冲工作

(1) 缓冲工作设定步骤的例子

缓冲工作的设定步骤如图 8.14 所示。

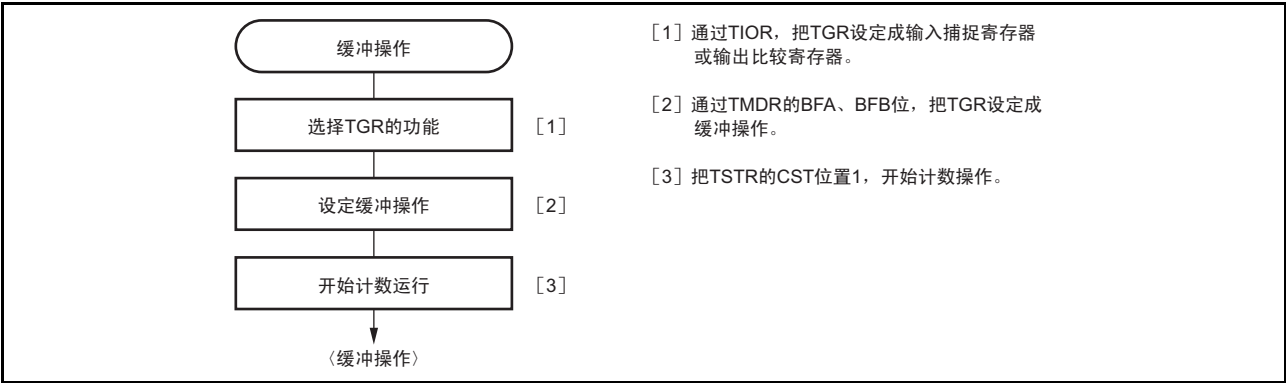


图 8.14 缓冲工作的设定步骤的例子

(2) 缓冲工作的例子

(a) TGR 为输出比较寄存器时

把通道 0 设定为 PWM 模式 1, 把 TGRA 和 TGRC 设定为缓冲工作时的运行时序如图 8.15 所示。这是设定通过比较匹配 B 清除 TCNT, 输出通过比较匹配 A 输出 1, 通过比较匹配 B 输出 0 时的例子。

由于设定为缓冲工作, 因此, 发生比较匹配 A 的同时输出发生变化, 缓冲器寄存器 TGRC 的值被传送到定时器通用寄存器 TGRA。每当发生比较匹配 A 时, 反复此工作。

关于 PWM 模式, 请参照「8.4.4 PWM 模式」。

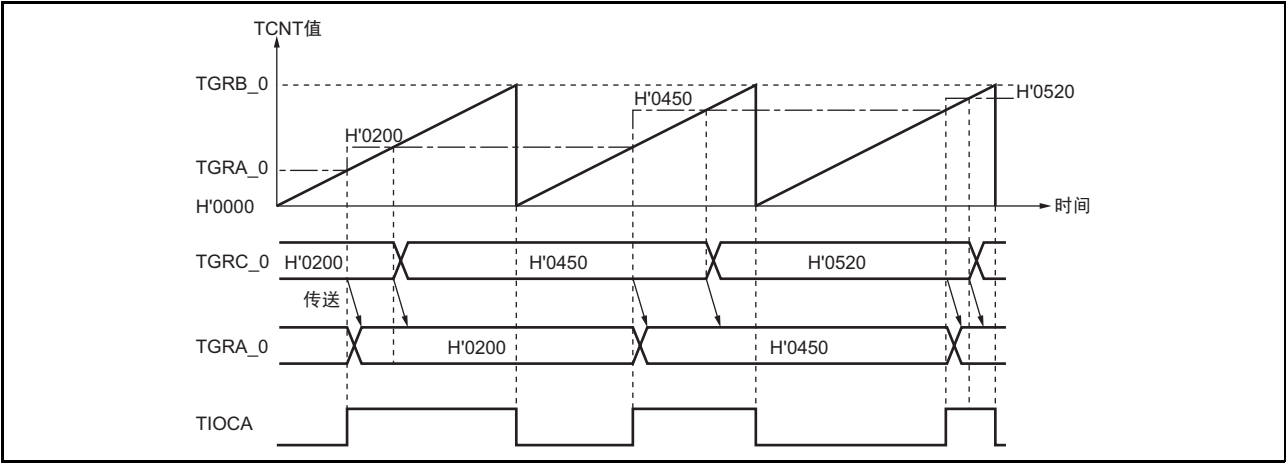


图 8.15 缓冲工作例 (1)

(b) TGR 为输入捕捉寄存器时

TGRA 设定为输入捕捉寄存器，TGRA 与 TGRC 设定为缓冲工作时的运行时序如图 8.16 所示。

TCNT 通过 TGRA 的输入捕捉清除计数器，TIOCA 管脚输入捕捉的输入沿能选择上升沿 / 下降沿的两边沿。

由于设定为缓冲工作，因此，通过输入捕捉 A，将 TCNT 的值保存到 TGRA 的同时，也将以前保存到 TGRA 的值转送到 TGRC。

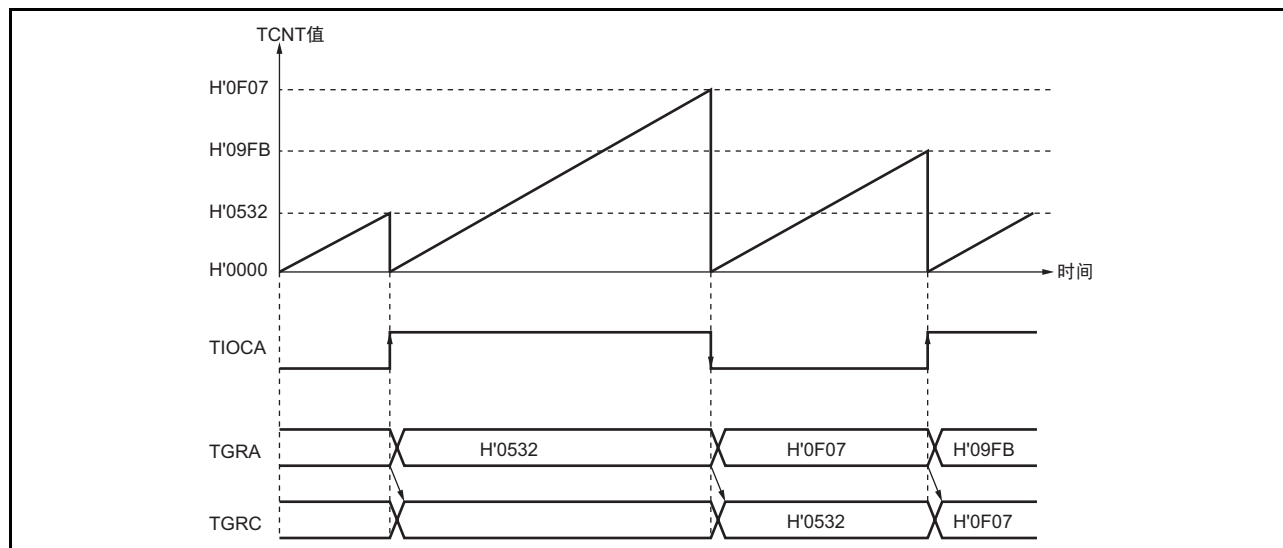


图 8.16 缓冲工作例子 (2)

8.4.4 PWM 模式

PWM 模式通过输出管脚输出各自的 PWM 波形。通过各 TGR 的比较匹配，输出电平能从 0 输出 /1 输出 / 交替输出中选择。

通过各 TGR 的设定，能输出占空比 0 ~ 100% 的 PWM 波形。

通过把 TGR 的比较匹配作为计数器清除源，能设定其寄存器的周期。全部通道能独立的设定 PWM 模式。也能同步工作。

PWM 模式有如下所示 2 种。

(a) PWM 模式 1

TGRA 和 TGRB 作为一对使用、TGRC 和 TGRD 作为一对使用，从 TIOCA、TIOCC 管脚产生 PWM 输出。从 TIOCA、TIOCC 管脚开始，通过比较匹配 A、C 进行指定 TIOR 的 IOA3 ~ IOA0、IOC3 ~ IOC0 位的输出，而且，通过比较匹配 B、D 进行指定 TIOR 的 IOB3 ~ IOB0、IOD3 ~ IOD0 位的输出。初始输出值依据 TGRA、TGRC 设定的值。成双使用的 TGR 设定值在一致时，即使发生比较匹配，输出值也不变化。

PWM 模式 1 能实现最多 4 相的 PWM 输出。

(b) PWMM 模式 2

1 个 TGR 作为周期寄存器，其它的 TGR 作为占空比寄存器生成 PWM 输出。通过比较匹配进行由 TIOR 指定的输出。而且，通过同步寄存器的比较匹配对寄存器清零，各管脚的输出值是 TIOR 中设定的初始值。如果周期寄存器和占空比寄存器的设定值一致，即使产生比较匹配，输出值也不改变。

在 PWM 模式 2，通过与同步工作并用能实现最大 7 相 PWM 输出。

PWM 输出管脚和寄存器的对应如表 8.18 所示。

表 8.18 各 PWM 输出寄存器和输出管脚

通道	寄存器	输出管脚	
		PWM 模式 1	PWM 模式 2
0	TGRA_0	TIOCA0	TIOCA0
	TGRB_0		TIOCB0
	TGRC_0	TIOCC0	TIOCC0
	TGRD_0		TIOCD0
1	TGRA_1	TIOCA1	TIOCA1
	TGRB_1		TIOCB1
2	TGRA_2	TIOCA2	TIOCA2
	TGRB_2		TIOCB2

【注】 在 PWM 模式 2，不能进行设定周期 TGR 的 PWM 输出。

(1) PWM 模式的设定步骤的例子

PWM 模式的设定步骤的例子如图 8.17 所示。



图 8.17 PWM 模式的设定步骤的例子

(2) PWM 模式的工作例子

PWM 模式 1 的工作例子如图 8.18 所示。

本图是 TGRA 的比较匹配作为 TCNT 的清除源, TGRA 的初始输出值和输出值设定为 0, TGRB 的输出值设定为 1 时的例子。

这时, TGRA 设定的值成为周期, TGRB 设定的值成为占空比。

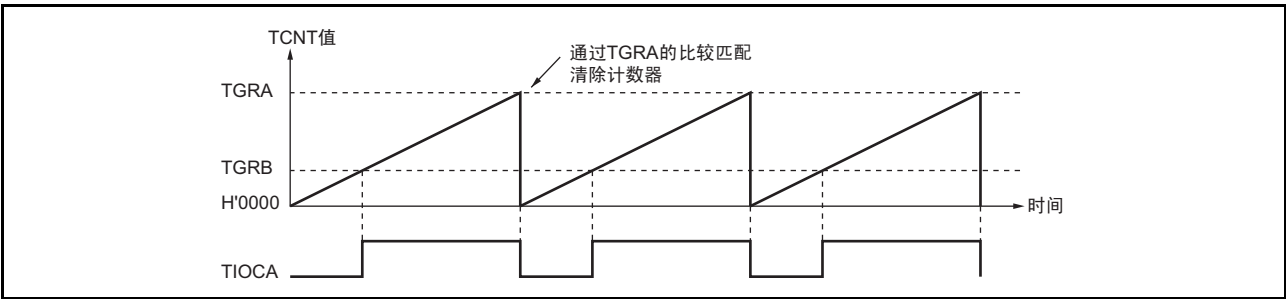


图 8.18 PWM 模式的工作例子 (1)

PWM 模式 2 的运行时序如图 8.19 所示。

本图是通道 0 和通道 1 作为同步工作，TCNT 的清除源作为 TGRB_1 的比较匹配，其他的 TGR (TGRA_0 ~ TGRD_0、TGRA_1) 的初始输出值设定为 0，输出值设定为 1，输出 5 相的 PWM 波形时的例子。

此时，TGRB_1 设定的值成为周期，其它的 TGR 设定的值成为占空比。

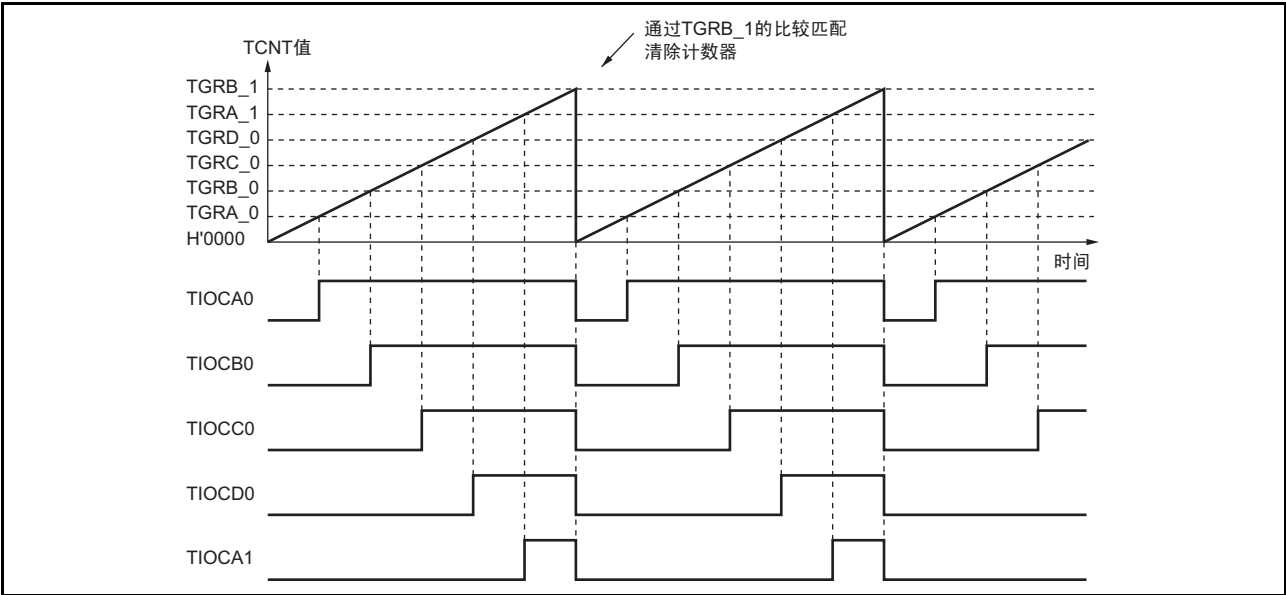


图 8.19 PWM 模式的工作例子 (2)

PWM 模式下，输出占空比 0%、占空比 100% 的 PWM 波形的运行时序如图 8.20 所示。

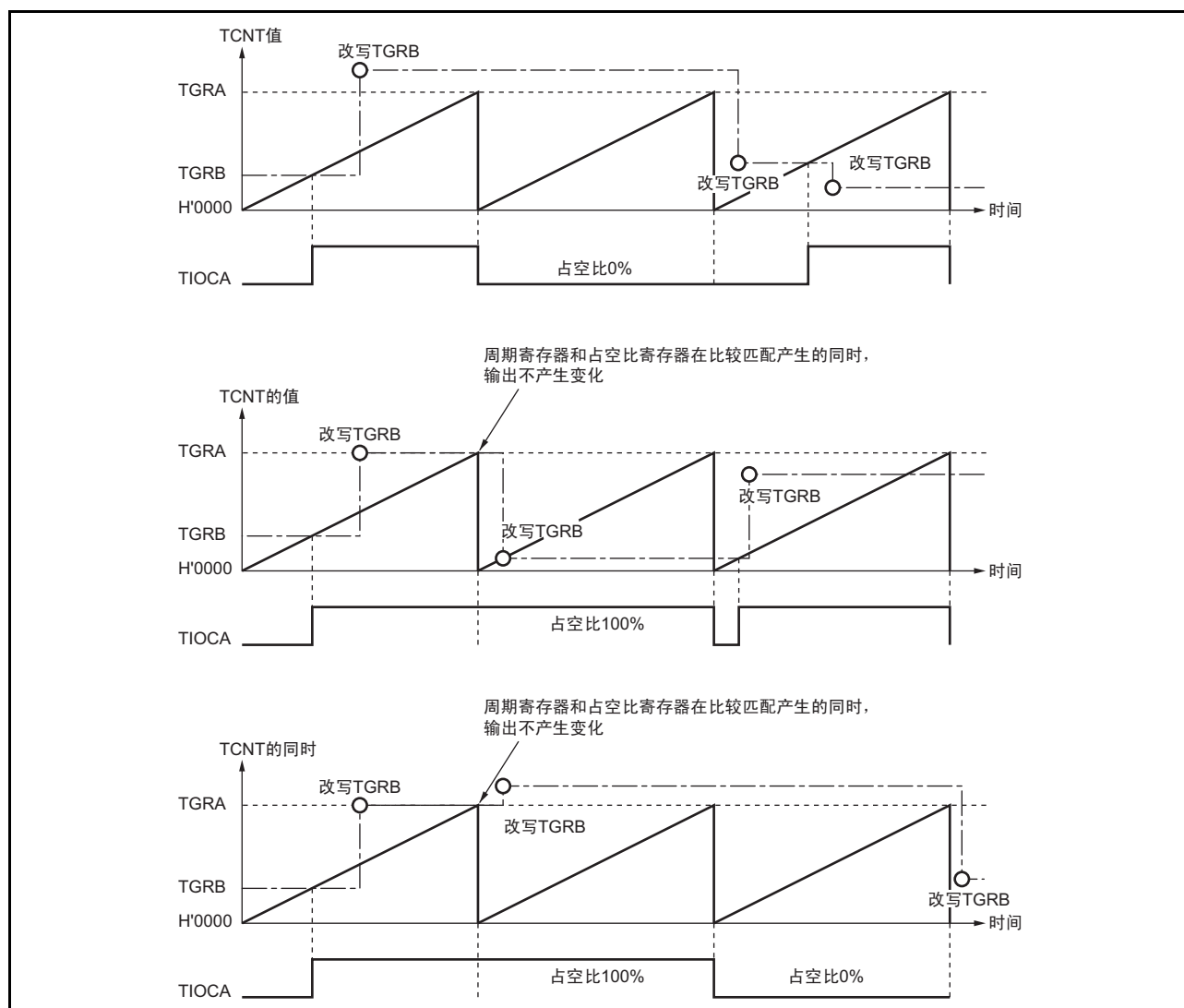


图 8.20 PWM 模式的工作例子 (3)

8.4.5 相位计数模式

在相位计数模式，通过通道 1 和通道 2 的设定，检测 2 个外部时钟输入的相位差，TCNT 进行累加 / 递减计数。

设定为相位计数模式，与 TCR 的 TPSC2 ~ TPSC0 位、CKEG1、CKEG0 位的设定无关，计数器输入时钟选择外部时钟，TCNT 作为累加 / 递减寄存器工作。但是，由于 TCR 的 CCLR1、CCLR0 位、TIOR、TIER、TGR 的功能有效，输入捕捉 / 比较匹配功能和中断功能能使用。

作为 2 相编码器脉冲的输入使用。

TCNT 在累加计数时，发生溢出后 TSR 的 TCFV 标志被置位。而且，递减计数时发生下溢后，TCFU 标志被置位。

TSR 的 TCFD 位是计数方向标志。通过读 TCFD 标志，能确认 TCNT 的累加计数或递减计数。

外部时钟管脚和通道的对应如表 8.19 所示。

表 8.19 相位计数模式时钟输入管脚

通 道	外部时钟管脚	
	A 相	B 相
通道 1 在相位计数模式时	TCLKA	TCLKB
通道 2 在相位计数模式时	TCLKC	TCLKD

(1) 相位计数模式的设定步骤的例子

相位计数模式的设定步骤的运行时序如图 8.21 所示。

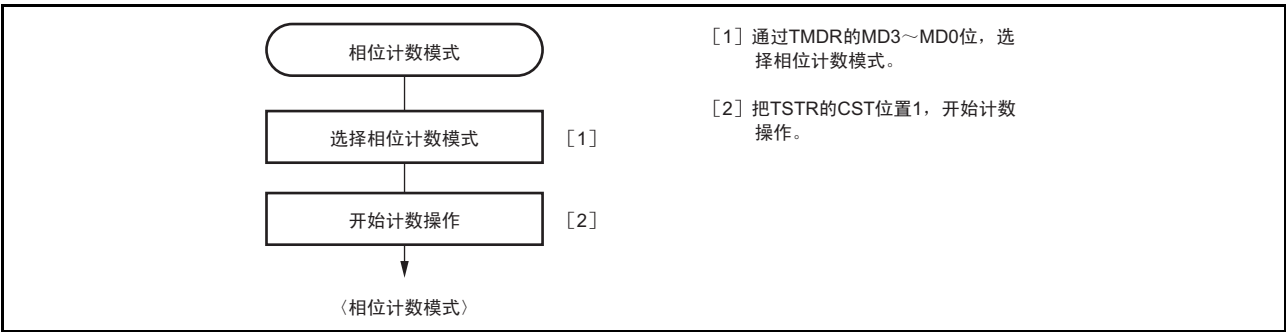


图 8.21 相位计数模式的设定步骤的例子

(2) 相位计数模式的工作例子

在相位计数模式，2 个外部时钟输入的相位差进行 TCNT 累加 / 递减计数。并且，根据计数条件的不同，有 4 种模式。

(a) 相位计数模式 1

相位计数模式 1 的运行时序如图 8.22 所示，TCNT 的累加 / 递减计数条件如表 8.20 所示。

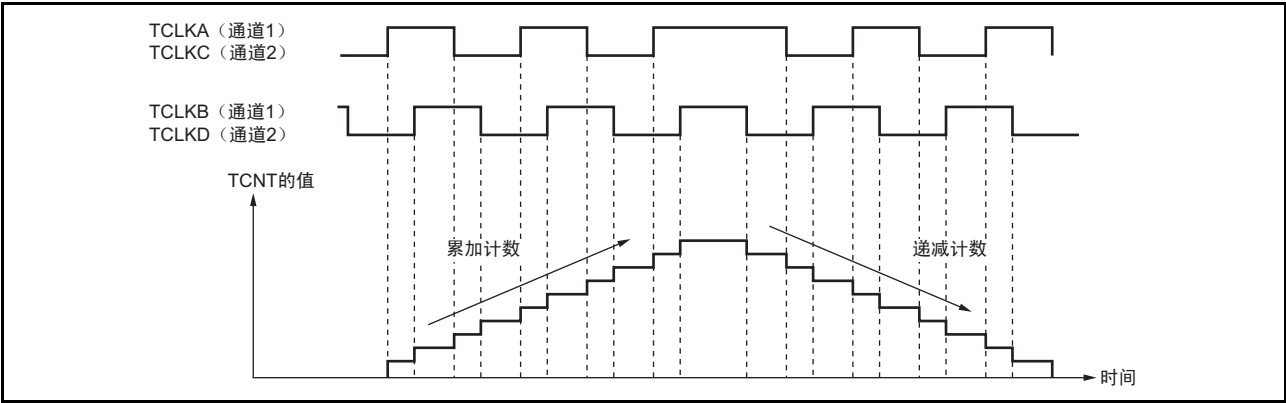


图 8.22 相位计数模式 1 的工作例子

表 8.20 相位计数模式 1 的累加 / 递减计数条件

TCLKA（通道 1） TCLKC（通道 2）	TCLKB（通道 1） TCLKD（通道 2）	工作内容
High 电平	↑	累加计数
Low 电平	↓	
↑	Low 电平	
↓	High 电平	
High 电平	↓	递减计数
Low 电平	↑	
↑	High 电平	
↓	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 的工作例子如图 8.23 所示，TCNT 的累加 / 递减计数条件如表 8.21 所示。

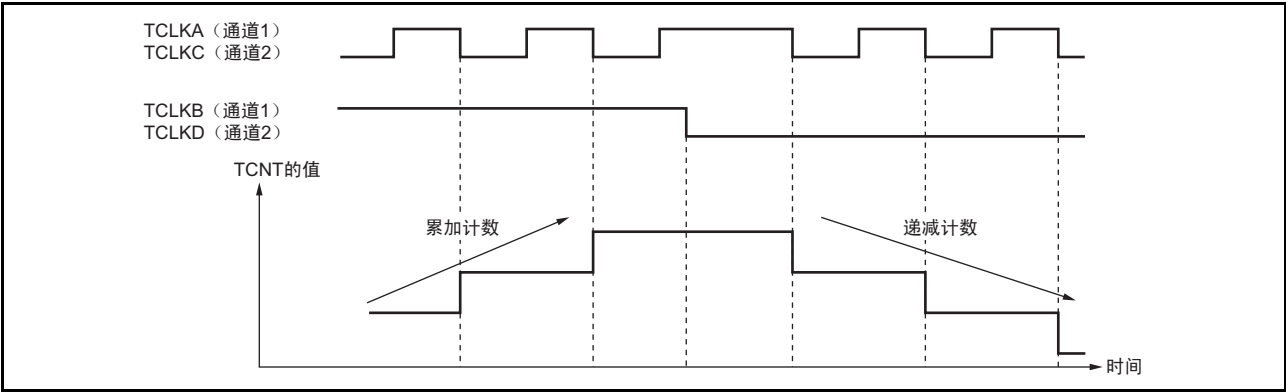


图 8.23 相位计数模式 2 的工作例子

表 8.21 相位计数模式 2 的累加 / 递减计数条件

TCLKA（通道 1） TCLKC（通道 2）	TCLKB（通道 1） TCLKD（通道 2）	工作内容
High 电平	↑	Don't care
Low 电平	↓	
↑	Low 电平	
↓	High 电平	累加计数
High 电平	↓	Don't care
Low 电平	↑	
↑	High 电平	
↓	Low 电平	递减计数

【符号说明】

- ↑ ： 上升沿
- ↓ ： 下降沿

(c) 相位计数模式 3

相位计数模式 3 的工作例子如图 8.24 所示，TCNT 的累加 / 递减计数条件如表 8.22 所示。

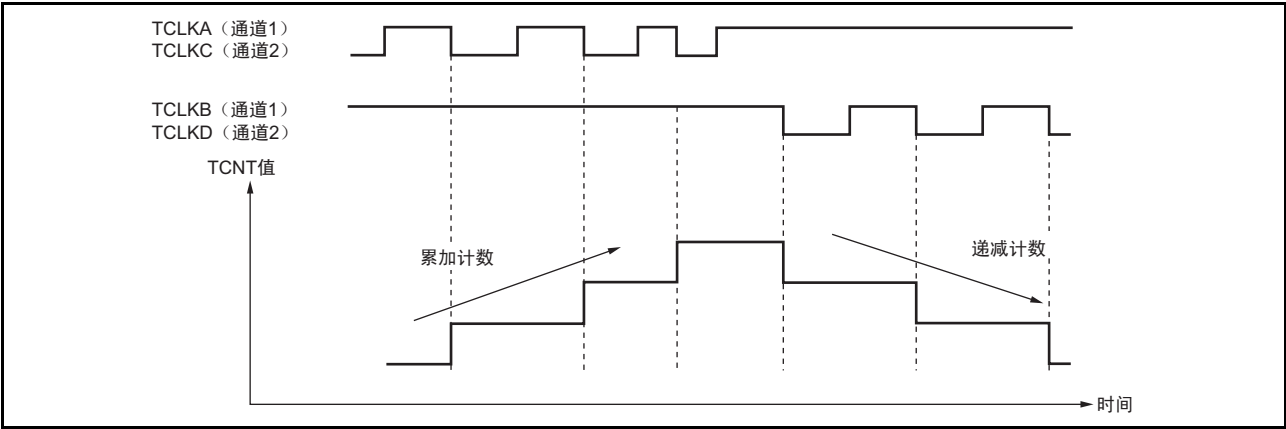


图 8.24 相位计数模式 3 的工作例子

表 8.22 相位计数模式 3 的累加 / 递减计数条件

TCLKA（通道 1） TCLKC（通道 2）	TCLKB（通道 1） TCLKD（通道 2）	工作内容
High 电平	↑	Don't care
Low 电平	↓	
↑	Low 电平	
↓	High 电平	累加计数
High 电平	↓	递减计数
Low 电平	↑	Don't care
↑	High 电平	
↓	Low 电平	

【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 的工作例子如图 8.25 所示，TCNT 的累加 / 递减计数条件如表 8.23 所示。

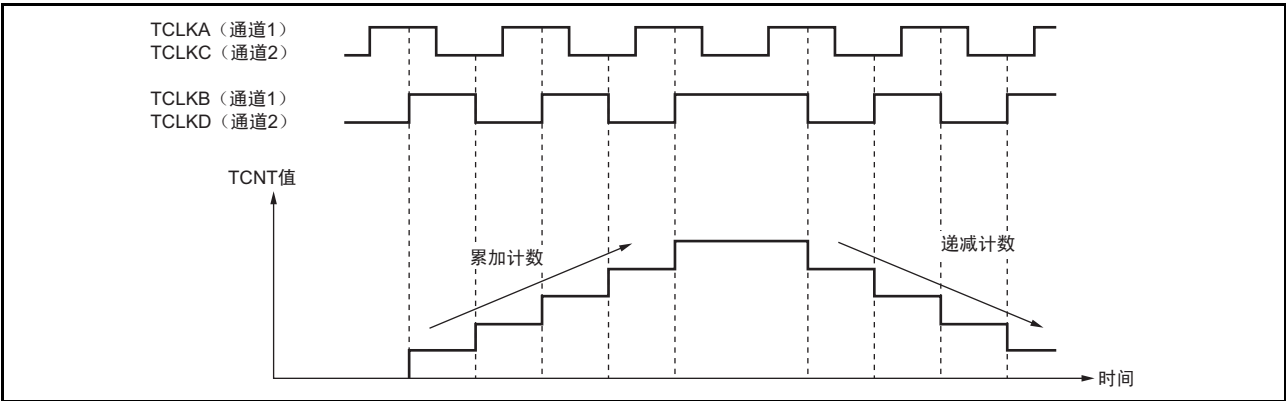


图 8.25 相位计数模式 4 的工作例子

表 8.23 相位计数模式 4 的累加 / 递减计数条件

TCLKA (通道 1) TCLKC (通道 2)	TCLKB (通道 1) TCLKD (通道 2)	工作内容
High 电平	↑	累加计数
Low 电平	↓	
↑	Low 电平	Don't care
↓	High 电平	
High 电平	↓	递减计数
Low 电平	↑	
↑	High 电平	Don't care
↓	Low 电平	

【符号说明】

- ↑ : 上升沿
- ↓ : 下降沿

(3) 相位计数模式应用例子

如图 8.26 所示，通道 1 设定为相位计数模式，通道 1 与通道 0 配对输入伺服马达的两相编码器脉冲，依此检测位置或速度。

通道 1 设定为相位计数模式，向 TCLKA 和 TCLKB 输入编码器脉冲的 A 相和 B 相。

在通道 0，通过 TGRC_0 的比较匹配，TCNT 作为计数器清除工作，TGRA_0 和 TGRC_0 使用比较匹配功能来设定速度控制周期和位置控制周期。TGRB_0 使用输入捕捉功能来缓冲工作 TGRB_0 和 TGRD_0。通道 1 的计数器输入时钟作为 TGRB_0 的输入捕捉源，检测出 2 相编码器的 4 倍频脉冲的脉冲宽度。

在通道 1 的 TGRA_1 和 TGRB_1，设定输入捕捉功能，通道 0 的 TGRA_0 和 TGRC_0 比较匹配的选择作为输入捕捉源，保存各个控制周期时间的累加 / 递减计数器的值。

通过上述进行正确的位置 / 速度检测。

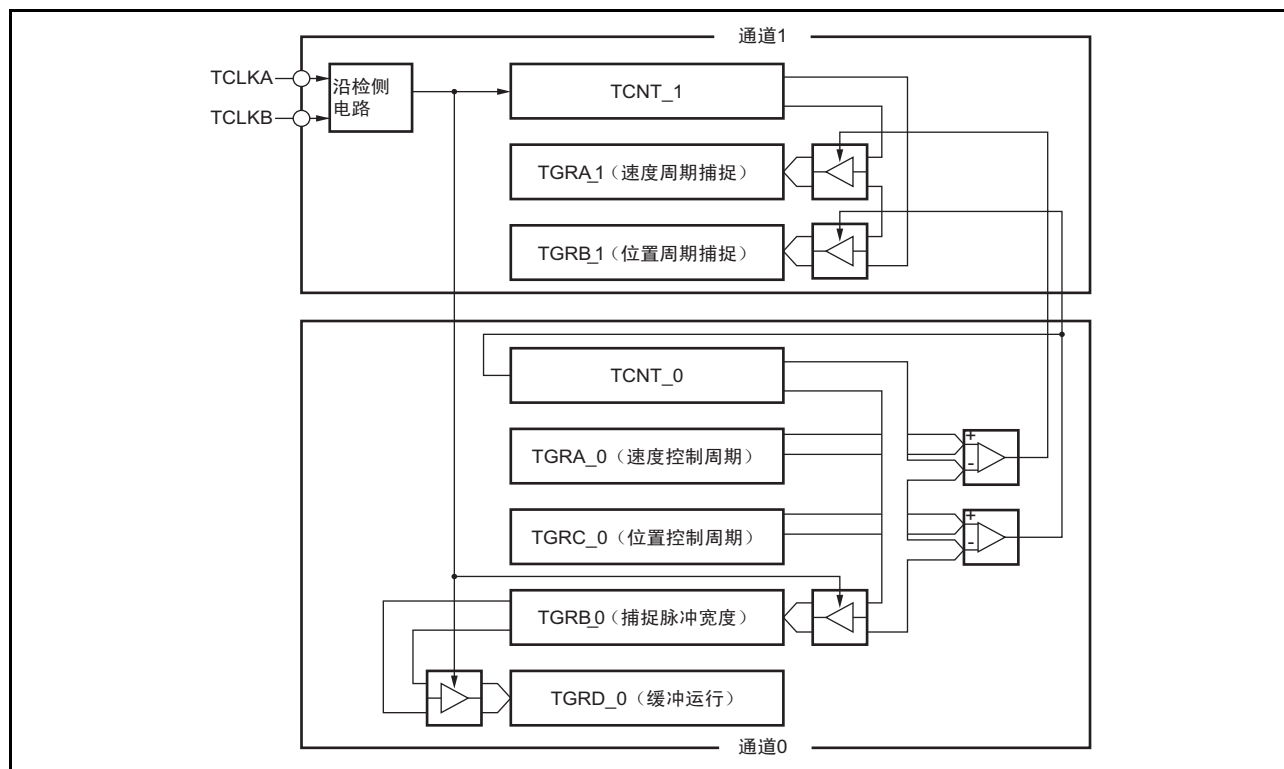


图 8.26 相位计数模式的应用例子

8.5 中断条件

TPU 的 3 种中断源有 TGR 的输入捕捉 / 比较匹配中断、TCNT 溢出中断、TCNT 下溢中断。各中断源由于具有各专用的状态标志和允许 / 禁止位，不能独立的允许或禁止产生中断请求信号。

如果满足中断条件，将 TSR 对应的状态标志置 1。此时如果 TIER 对应的允许 / 禁止位置 1，请求相应中断。由于状态标志清零，解除中断请求。

通道间的优先顺序，通过中断控制器能改变。通道内的优先顺序固定。详细请参照「第 5 章 中断控制器」。

TPU 的中断源如表 8.24 所示。

表 8.24 TPU 中断一览

通道	名	中断源	中断标志
0	TGI0A	TGRA_0 输入捕捉的输入管脚 / 比较匹配	TGFA0
	TGI0B	TGRB_0 输入捕捉的输入管脚 / 比较匹配	TGFB0
	TGI0C	TGRC_0 输入捕捉的输入管脚 / 比较匹配	TGFC0
	TGI0D	TGRD_0 输入捕捉的输入管脚 / 比较匹配	TGFD0
	TCI0V	TCNT_0 的溢出	TCFV0
1	TGI1A	TGRA_1 的输入捕捉的输入管脚 / 比较匹配	TGFA1
	TGI1B	TGRB_1 的输入捕捉的输入管脚 / 比较匹配	TGFB1
	TCI1V	TCNT_1 的溢出	TCFV1
	TCI1U	TCNT_1 的下溢	TCFU1
2	TGI2A	TGRA_2 的输入捕捉的输入管脚 / 比较匹配	TGFA2
	TGI2B	TGRB_2 的输入捕捉的输入管脚 / 比较匹配	TGFB2
	TCI2V	TCNT_2 的溢出	TCFV2
	TCI2U	TCNT_2 的下溢	TCFU2

【注】 表示复位后的初始状态。通道间的优先顺序通过中断控制器可以变更。

(1) 输入捕捉 / 比较匹配中断

根据各通道 TGR 的输入捕捉 / 比较匹配信号的产生，将 TSR 的 TGF 标志置 1，同时如果 TIER 的 TGIE 位置 1，请求相应中断。由于 TGF 标志清零，解除中断请求。在 TPU，通道 0 有 4 个，通道 1 和通道 2 各有 2 个，共有 8 个输入捕捉 / 比较匹配中断。

(2) 溢出中断

根据各通道 TCNT 的溢出信号的产生，将 TSR 的 TCFV 标志置 1，同时如果 TIER 的 TCIEV 位置 1，请求相应中断。由于 TCFV 标志清零，解除中断请求。在 TPU，各通道有 1 个，共有 3 个溢出中断。

(3) 下溢中断

根据各通道 TCNT 下溢信号的产生，将 TSR 的 TCFU 标志置 1，同时如果 TIER 的 TCIEU 置 1，请求相应中断。由于 TCFU 标志清零，解除中断请求。在 TPU，通道 1 和通道 2 各有 1 个，共有 2 个下溢中断。

8.6 A/D 转换器的启动

通过各通道 TGRA 的输入捕捉 / 比较匹配，能启动 A/D 转换器。

通过各通道 TGRA 的输入捕捉 / 比较匹配信号的产生，将 TSR 的 TGFA 标志置 1，同时如果 TIER 的 TTGE 位置 1，请求 A/D 转换器开始 A/D 转换。此时在 A/D 转换器一方，如果选择 TPU 的转换开始触发，就开始 A/D 转换。

在 TPU，各通道有 1 个，共有 3 个 TGRA 的输入捕捉 / 比较匹配中断成为 A/D 转换器的转换开始条件。

8.7 工作时序

8.7.1 输入 / 输出时序

(1) TCNT 的计数时序

当内部时钟工作时，内部 TCNT 的计数时序如图 8.27 所示。当外部时钟工作时，TCNT 的计数时序如图 8.28 所示。

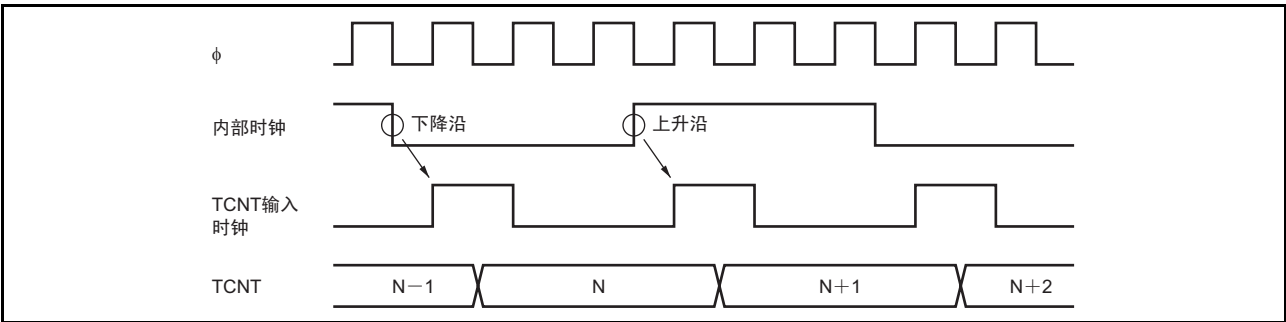


图 8.27 内部时钟工作时的计数时序

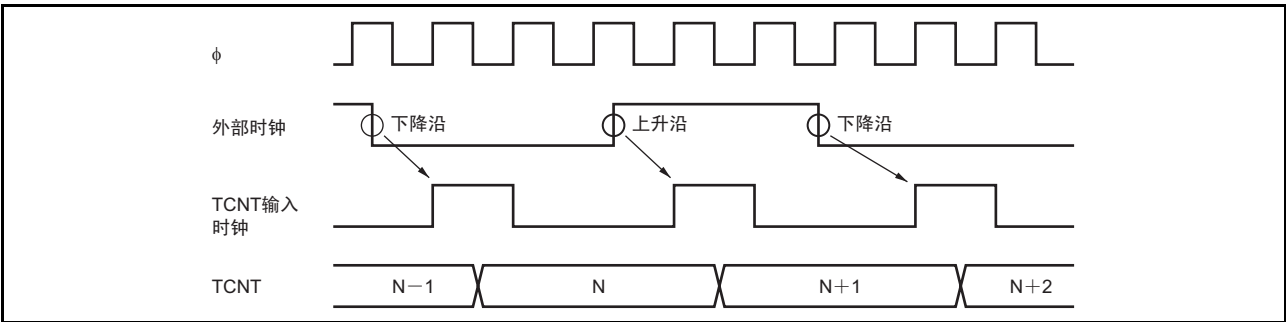


图 8.28 外部时钟工作时的计数时序

(2) 输出比较的时序

根据 TCNT 和 TGR 一致后的最后状态 (TCNT 更新一致的计数值的时序) 产生比较匹配信号。比较匹配信号产生时, TIOR 设定的输出值由输出比较的输出管脚 (TIOC 管脚) 输出。从 TCNT 和 TGR 一致后, 到产生 TCNT 输入时钟为止, 不产生比较匹配信号。

输出比较的输出管脚时序如图 8.29 所示。

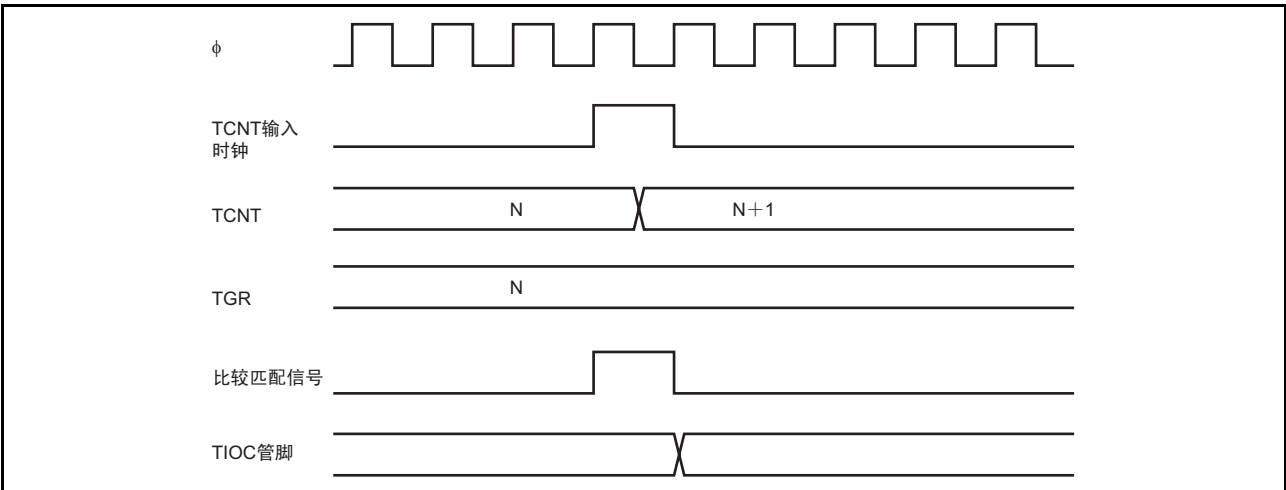


图 8.29 输出比较的输出管脚时序

(3) 输入捕捉信号的时序

输入捕捉的时序如图 8.30 所示。

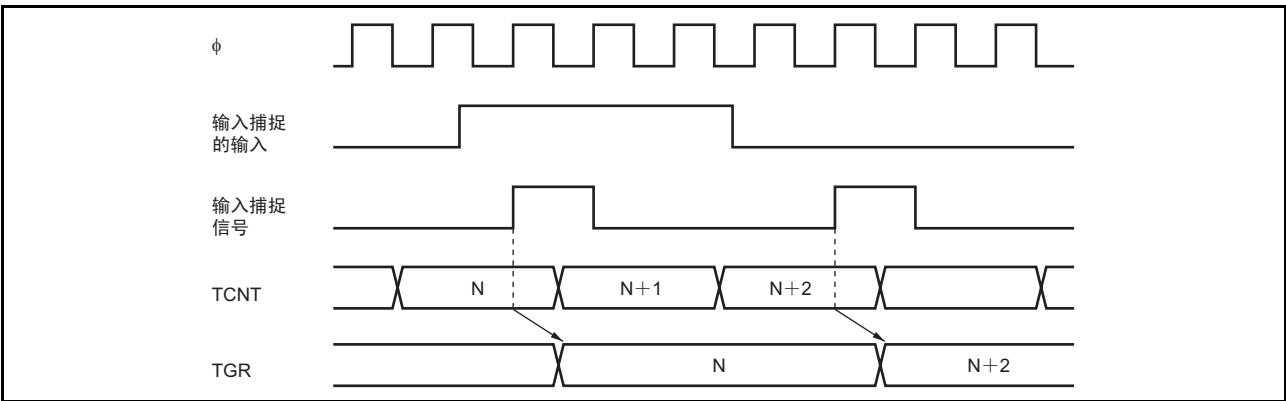


图 8.30 输入捕捉信号时序

(4) 通过比较匹配 / 输入捕捉的计数器清除的时序

根据比较匹配信号的产生，在指定计数器清除时的时序如图 8.31 所示。
根据输入捕捉信号的产生，在指定计数器清除时的时序如图 8.32 所示。

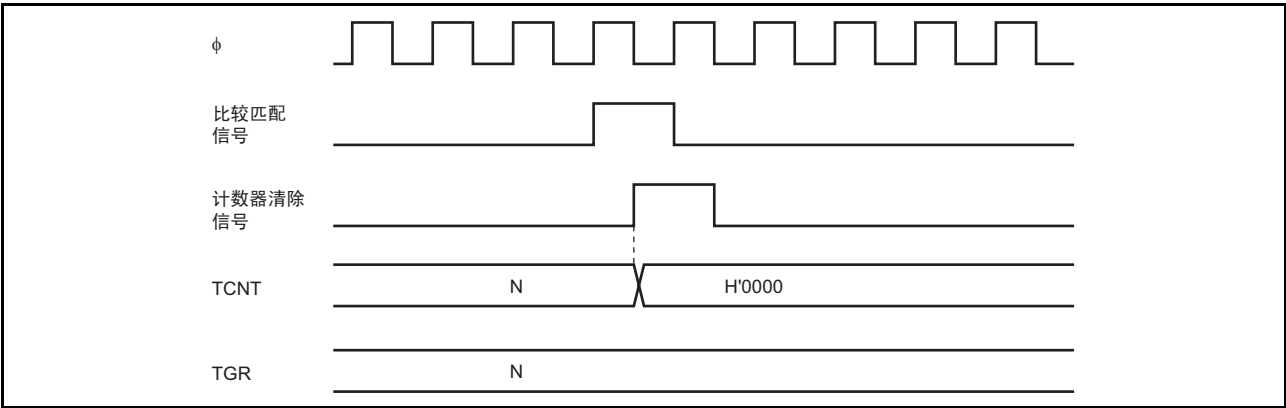


图 8.31 计数器清除的时序（比较匹配）

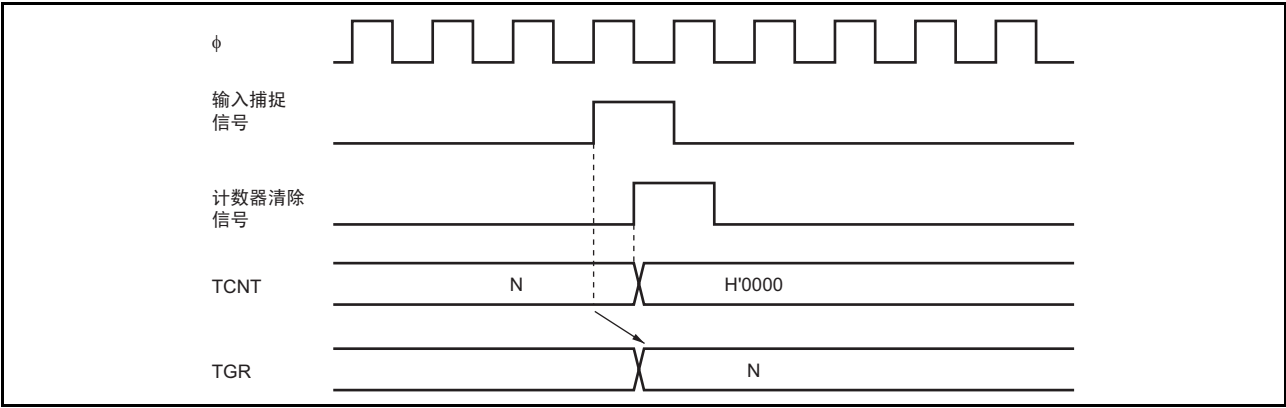


图 8.32 计数器清除的时序（输入捕捉）

(5) 缓冲工作的时序

缓冲工作时的时序如图 8.33、图 8.34 所示。

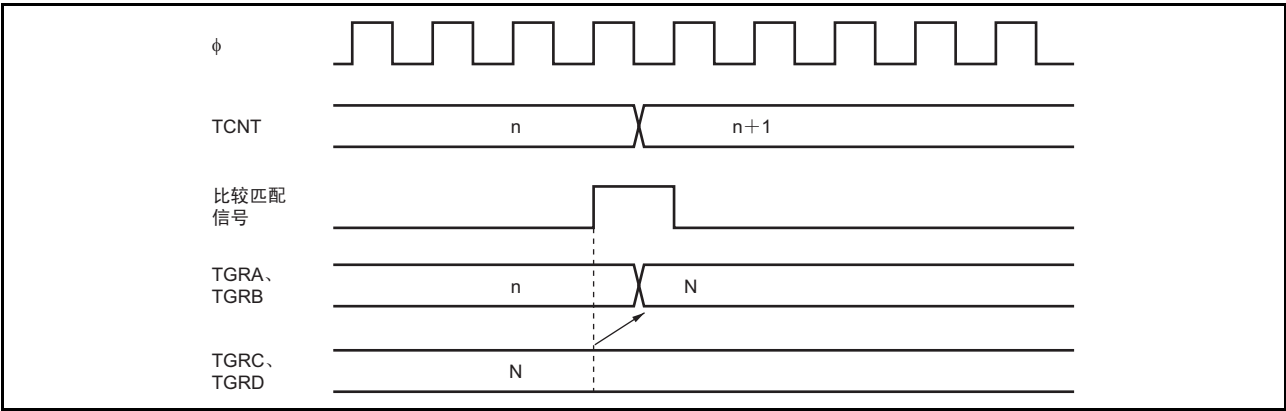


图 8.33 缓冲工作的时序（比较匹配）

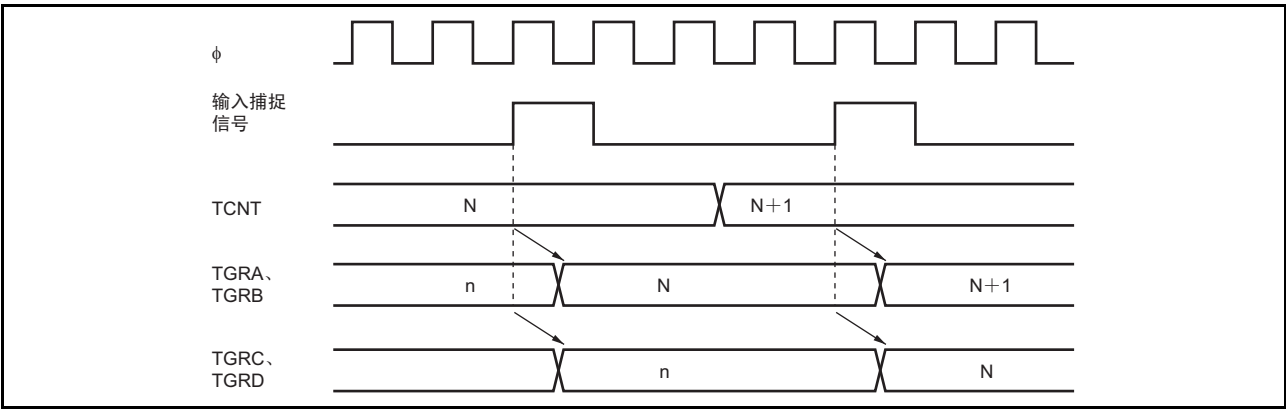


图 8.34 缓冲工作的时序（输入捕捉）

8.7.2 中断信号的时序

(1) 比较匹配时 TGF 标志的置位时序

根据比较匹配信号的产生，TSR 的 TGF 标志置位的时序和 TGI 中断请求信号的时序如图 8.35 所示。

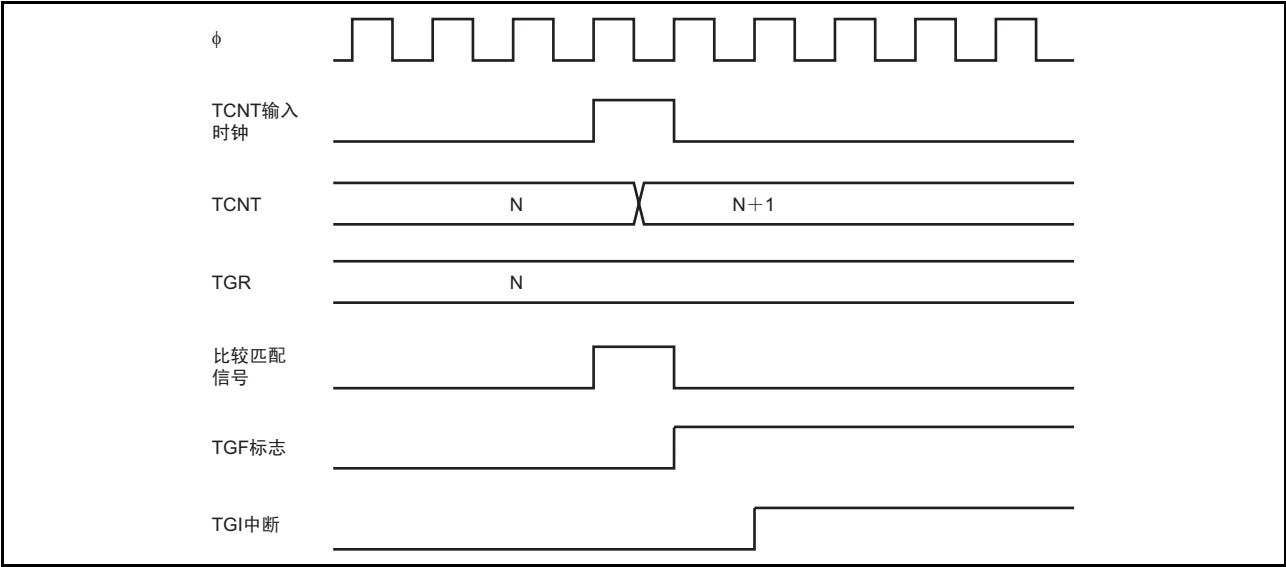


图 8.35 TGI 中断时序 (比较匹配)

(2) 输入捕捉时 TGR 标志置位的时序

根据输入捕捉信号的产生，TSR 的 TGF 标志置位的时序和 TGI 中断请求信号的时序如图 8.36 所示。

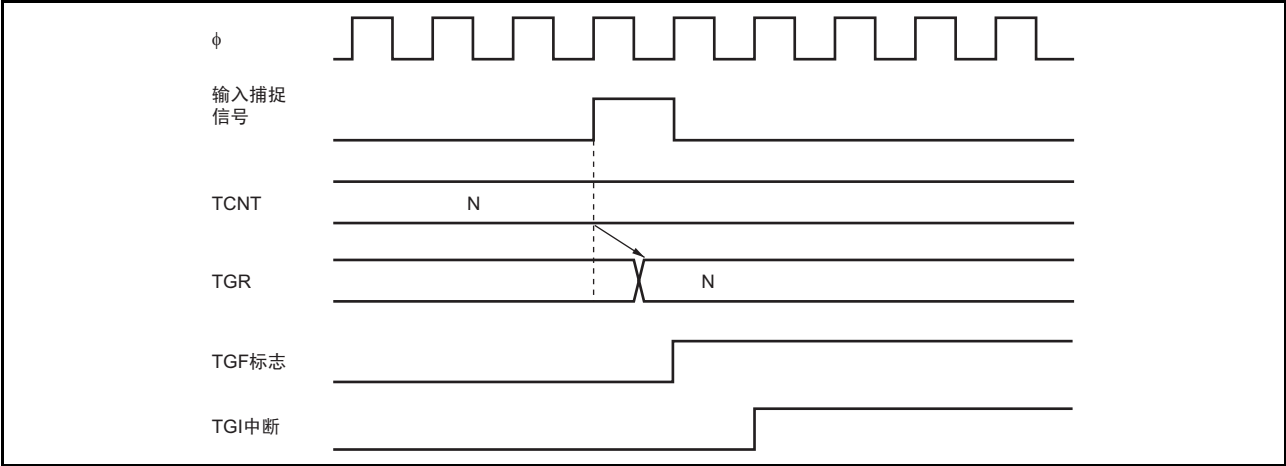


图 8.36 TGI 中断时序 (输入捕捉)

(3) TCFV 标志 / TCFU 标志置位的时序

根据溢出信号的产生，TSR 的 TCFV 标志置位的时序和 TCIV 中断请求信号的时序如图 8.37 所示。
根据下溢信号的产生，TSR 的 TCFU 标志置位的时序和 TCIU 中断请求信号的时序如图 8.38 所示。

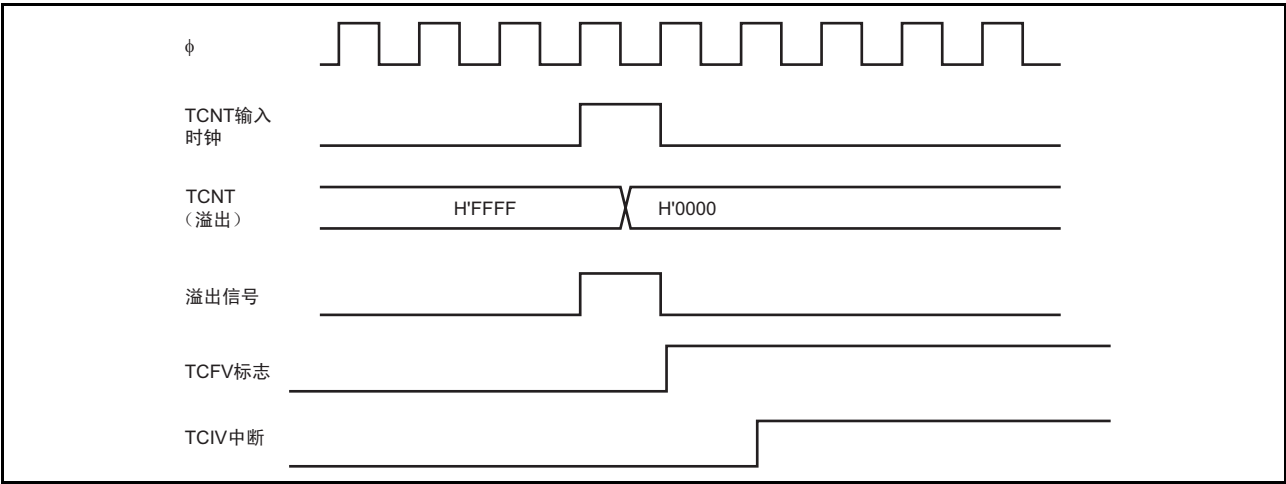


图 8.37 TCIV 中断置位的时序

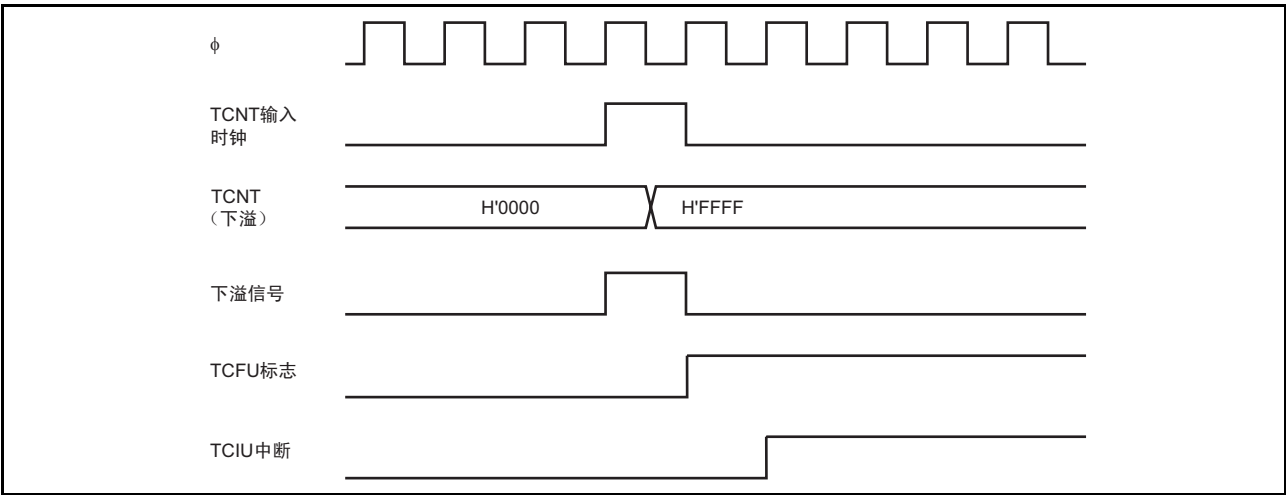


图 8.38 TCIU 中断置位的时序

(4) 状态标志的清除时序

状态标志在 CPU 读 1 的状态，写 0 后清除。根据 CPU 状态标志，清除时序如图 8.39 所示。

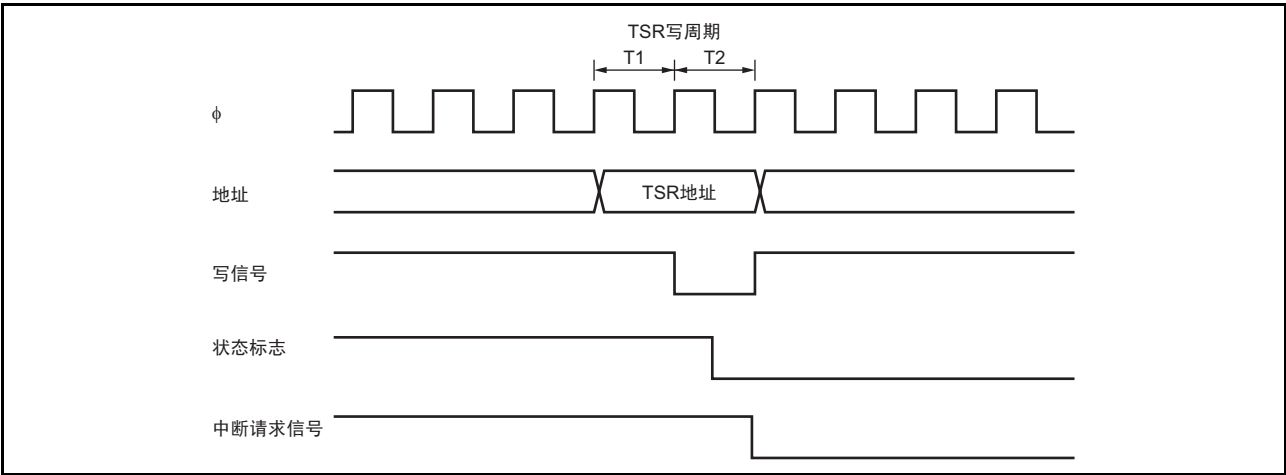


图 8.39 根据 CPU 状态标志的清除时序

8.8 用时的注意事项

8.8.1 模块待机模式的设定

根据模块待机控制寄存器，能设定禁止还是许可 TPU 工作。开始设置 TPU 工作处于暂停状态。当解除模块待机模式后，能实现寄存器的存取。详细请参考「第 20 章 低功耗状态」。

8.8.2 输入时钟的限制事项

输入时钟的脉冲宽度在单沿时需要 1.5 个时钟周期以上，在双边沿时需要 2.5 个周期以上。低于这个数字的脉冲宽度不能正常工作。

在相位计数模式时，2 个输入时钟相位差及覆盖需要在各自 1.5 个周期以上，脉冲宽度需要在 2.5 个周期以上。相位计数模式的输入时钟条件如图 8.40 所示。

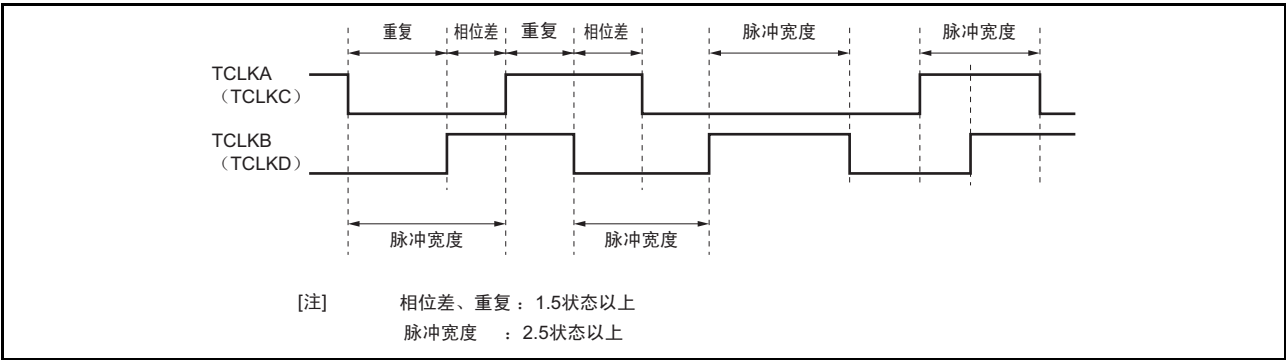


图 8.40 相位计数模式时的相位差、覆盖、及脉冲幅

8.8.3 周期设定时的注意事项

根据比较匹配设定计数器被清零时，TCNT 和 TGR 的值一致时的最后状态被清零（TCNT 更新一致的计数值的时序）。因此，实际的计数器频率数如下方程式所示。

$$f = \frac{\phi}{(N+1)}$$

f : 计数器频率

ϕ : 工作频率

N : TGR 的设定值

8.8.4 TCNT 的写和清除的竞争

在 TCNT 写周期中的 T2 状态，如果产生计数器清除信号，不进行向 TCNT 的写入，优先 TCNT 的清除。这个时序如图 8.41 所示。

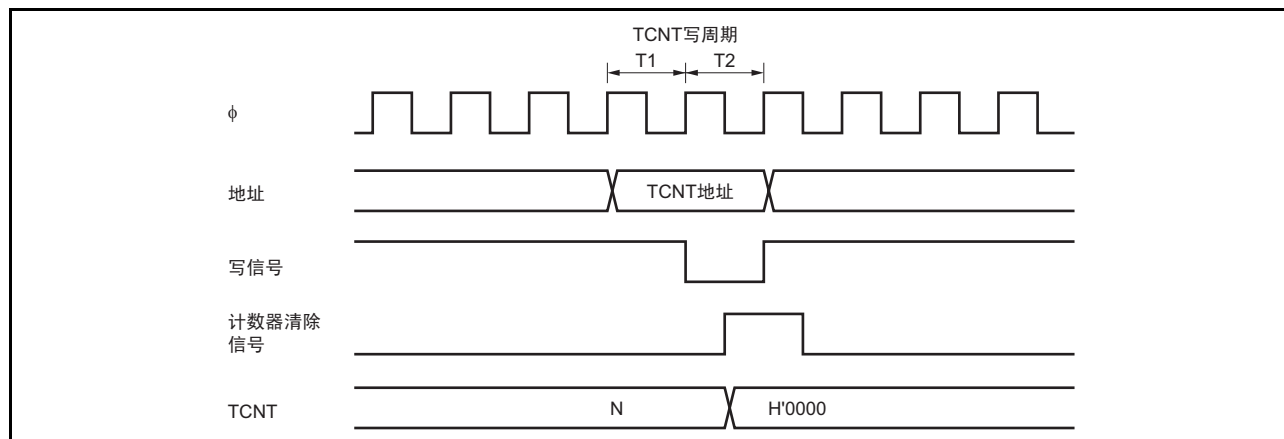


图 8.41 TCNT 的写和清除的竞争

8.8.5 TCNT 的写和累加计数的竞争

在 TCNT 写周期中的 T2 状态，即使产生累加计数，也不累加计数，优先向 TCNT 写入。这个时序如图 8.42 所示。

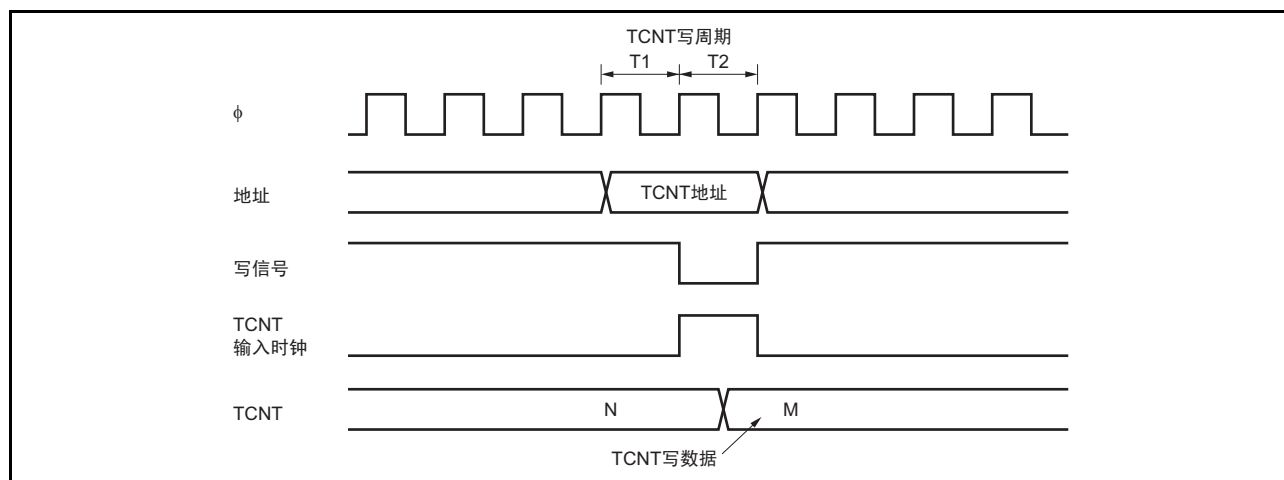


图 8.42 TCNT 的写和累加计数的竞争

8.8.6 TGR 的写和比较匹配的竞争

在 TGR 的写周期中 T2 状态，即使产生比较匹配，优先 TGR 的写，比较匹配信号被禁止。即使在写与前一次相同的值时，也不产生比较匹配。

这个时序如图 8.43 所示。

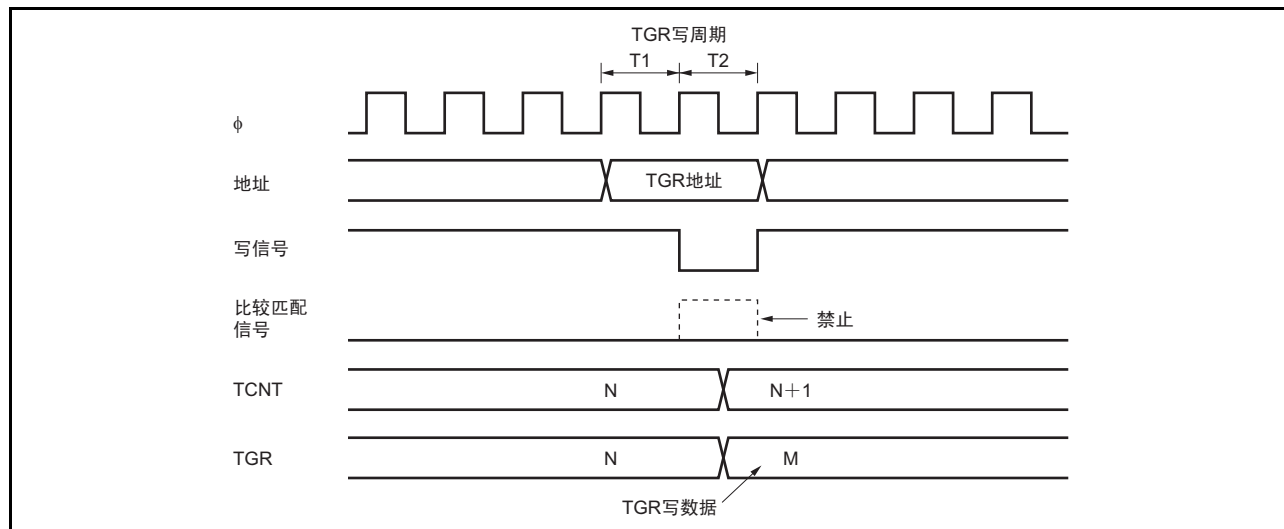


图 8.43 TGR 的写和比较匹配的竞争

8.8.7 缓冲寄存器的写和比较匹配的竞争

在 TGR 的写入周期中 T2 状态，如果产生比较匹配信号，根据缓冲器工作传送到 TGR 的数据成为写入数据。

这个时序如图 8.44 所示。

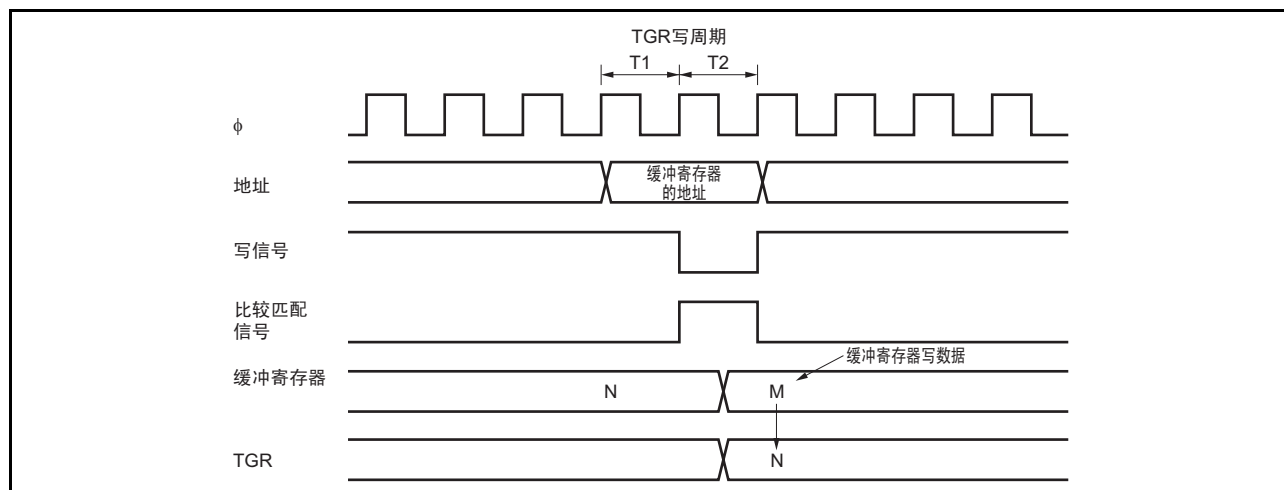


图 8.44 缓冲寄存器的写和比较匹配的竞争

8.8.8 TGR 的读和输入捕捉的竞争

在 TGR 的读周期中 T1 状态，如果产生输入捕捉信号，读的数据成为输入捕捉传送后的数据。
这个时序如图 8.45 所示。

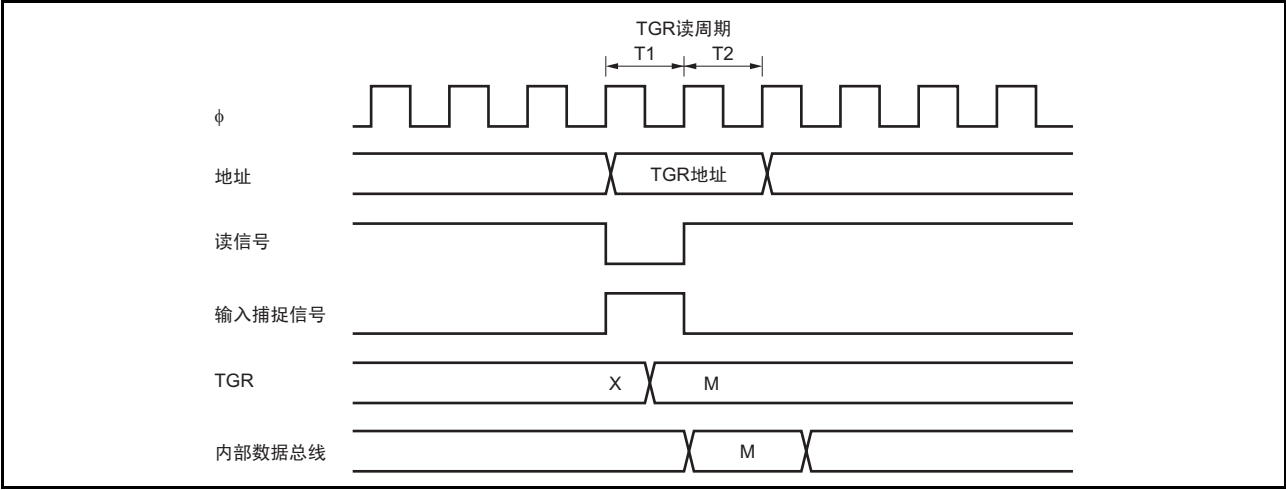


图 8.45 TGR 的读和输入捕捉的竞争

8.8.9 TGR 的写和输入捕捉的竞争

在 TGR 的写入周期中 T2 状态，如果产生输入捕捉，不进行向 TGR 的写入，优先输入捕捉。
此时序如图 8.46 所示。

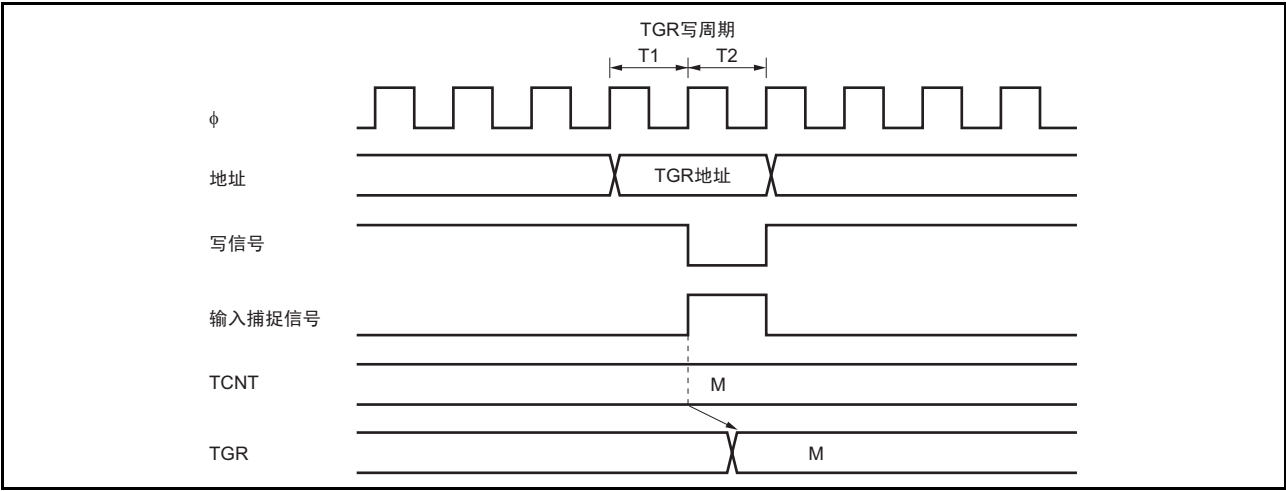


图 8.46 TGR 的写和输入捕捉的竞争

8.8.10 缓冲寄存器的写和输入捕捉的竞争

在缓冲寄存器写周期中的 T2 状态，如果产生输入捕捉信号，不进行向缓冲寄存器写，优先缓冲工作。此时序如图 8.47 所示。

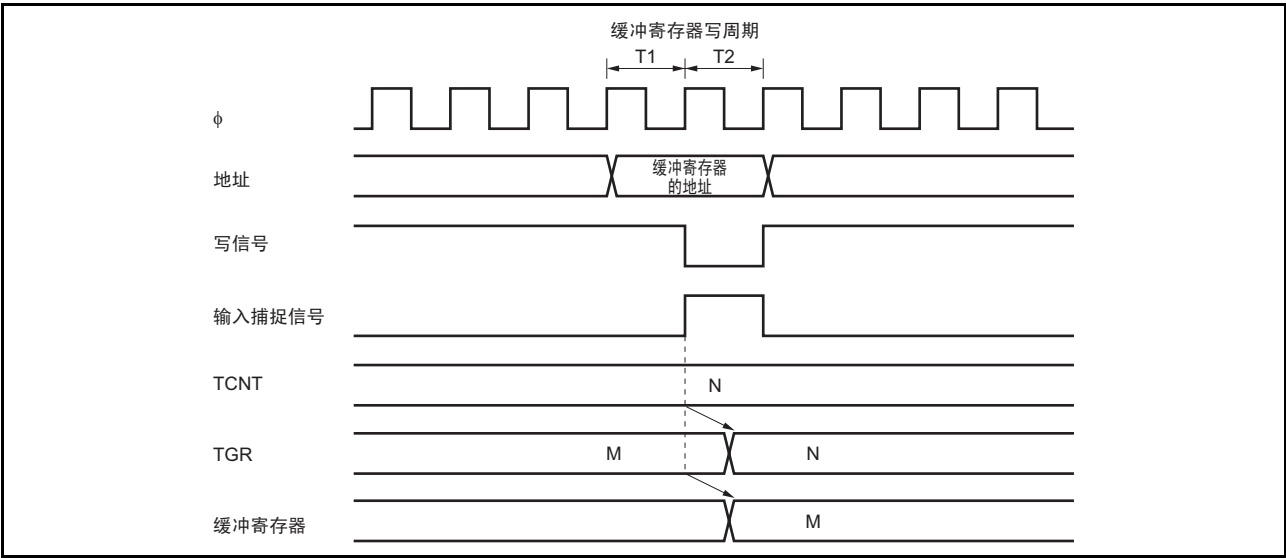


图 8.47 缓冲寄存器的写和输入捕捉的竞争

8.8.11 溢出 / 下溢和计数器清除的竞争

溢出 / 下溢和计数器清除同时产生后，TSR 的 TCFV/TCFU 标志不置位，优先 TCNT 的清除。作为清除 TGR 的比较匹配条件，在 TGR 设定 H'FFFF 时的运行时序如图 8.48 所示。

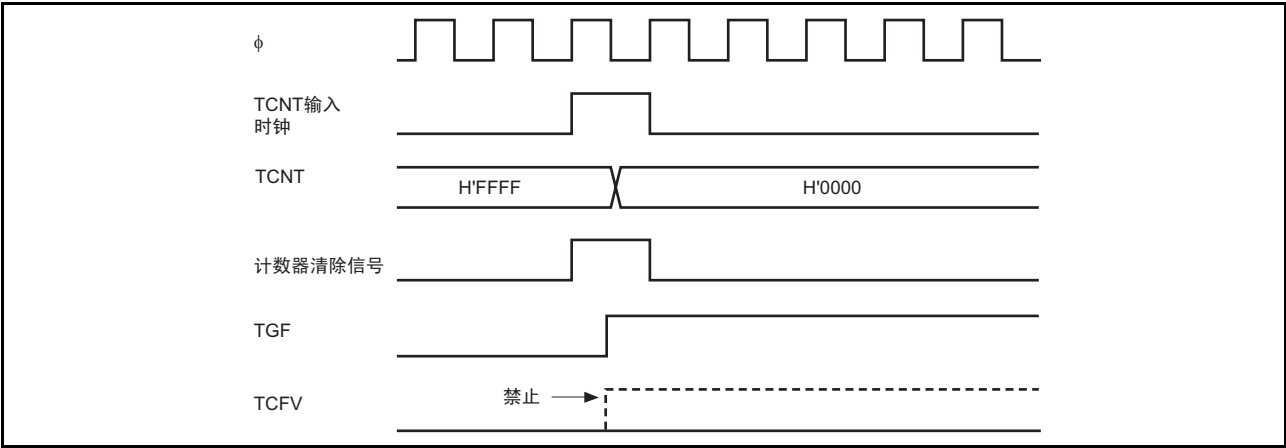


图 8.48 溢出和计数器清除的竞争

8.8.12 TCNT 的写和溢出 / 下溢的竞争

在 TCNT 的写周期中 T2 状态, 产生累加计数 / 递减计数, 即使产生溢出 / 下溢, 优先向 TCNT 的写, TSR 的 TCFV / TCFU 标志不置位。

TCNT 的写和溢出竞争时的运行时序如图 8.49 所示。

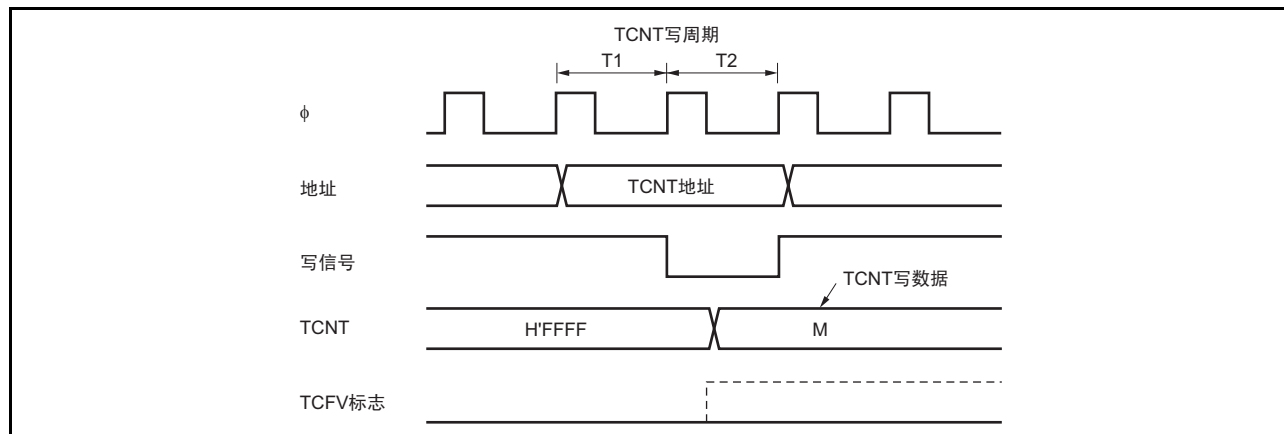


图 8.49 TCNT 的写和溢出的竞争

8.8.13 输入 / 输出管脚的兼用

在本 LSI, TCLKA 输入和 TIOCC0 输入 / 输出、TCLKB 输入和 TIOCD0 输入 / 输出、TCLKC 输入和 TIOCB1 输入 / 输出、TCLKD 输入和 TIOCB2 输入 / 输出的各管脚都兼用。输入外部时钟时, 请不要从兼用的管脚进行比较匹配输出。

8.8.14 模块停止时的中断

在请求中断状态, 如果是模块待机模式, 不能清除 CPU 的中断条件。禁止事前中断后作为模块待机模式。

8.8.15 子激活模式 / 监视模式的转换

在请求中断状态, 转换到子激活模式后, 不能清除 CPU 的中断源。禁止事前中断后向子激活模式 / 监视模式转换。

第 9 章 看门狗定时器 (WDT)

看门狗定时器是 (WDT_0、WDT_1) 8 位定时器, 当由于系统失控等原因不能改写计数器的值而产生溢出时, 对 LSI 内部进行复位。

不作为看门狗定时器使用时, 可当作间隔定时器使用。作为间隔定时器使用时, 计数器每次溢出时产生间隔定时器中断。WDT_0 的框图如图 9.1 所示, WDT_1 的框图如图 9.2 所示。

9.1 特点

- 可选择 8 种计数器输入时钟
- 看门狗定时器模式和间隔定时器模式可进行切换

看门狗定时器模式

- 计数器溢出时, 可选择是否产生 LSI 内部复位或内部 NMI 中断

间隔定时器模式

- 在计数器溢出时, 产生间隔计时器中断 (WOVI)

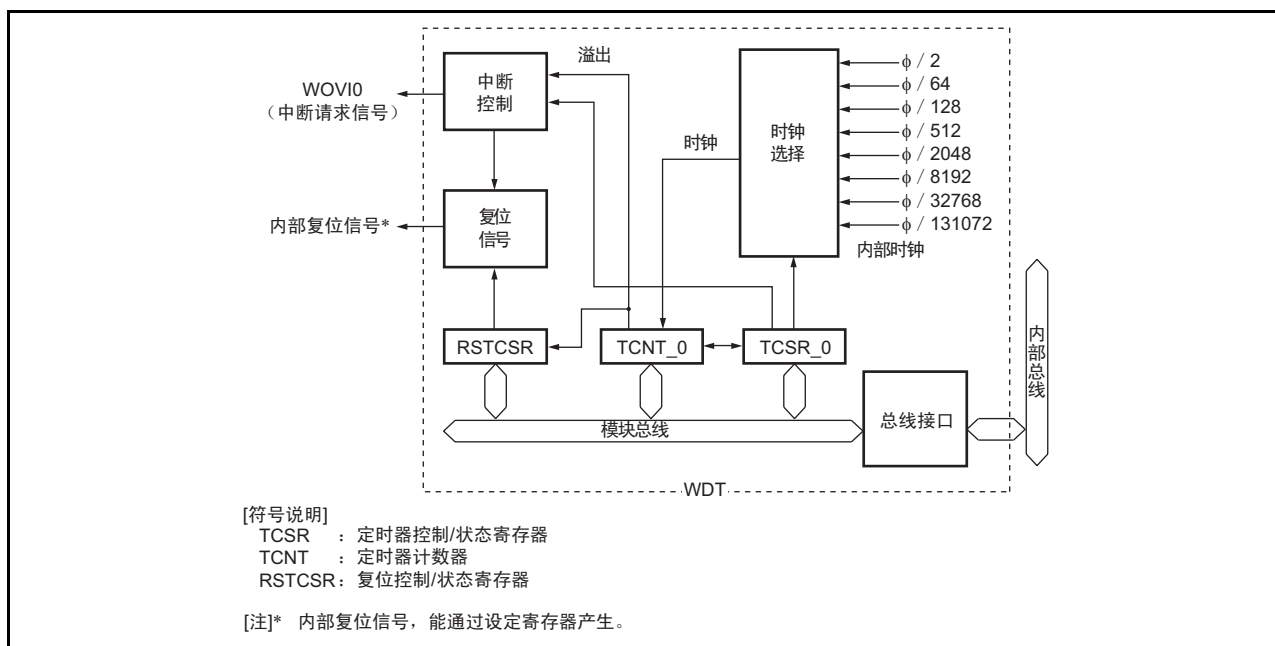


图 9.1 WDT_0 的框图

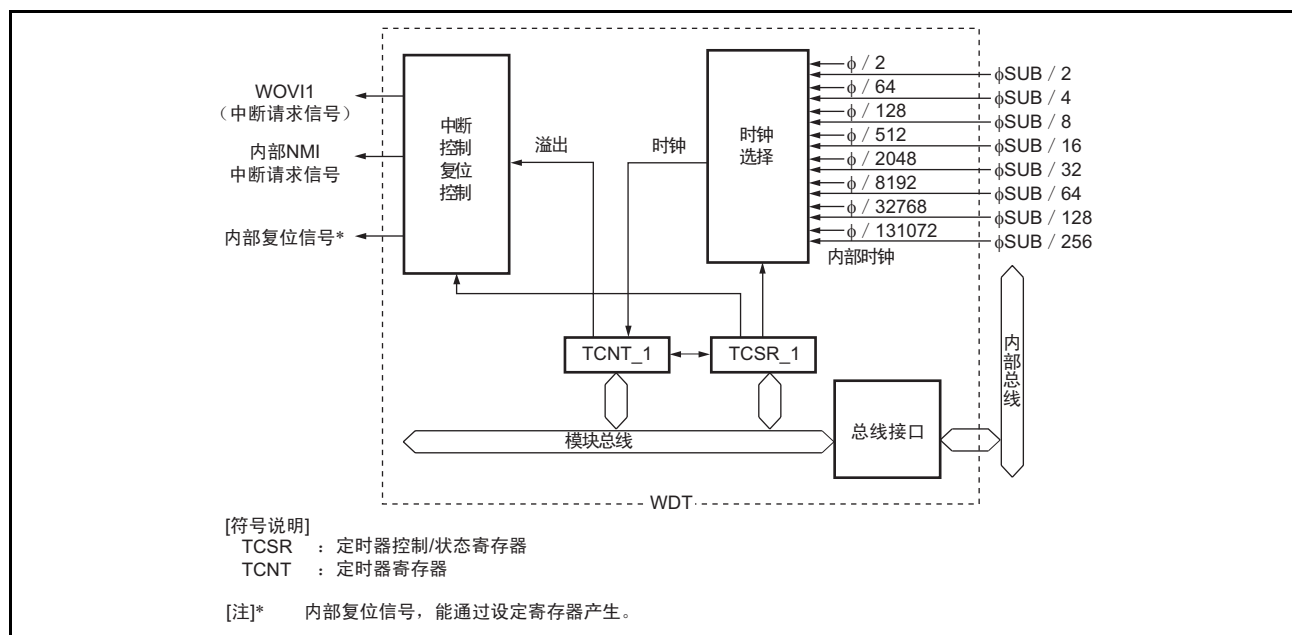


图 9.2 WDT_1 的框图

9.2 寄存器的说明

WDT 有以下寄存器。为了 TCSR、TCNT、RSTCSR 不能简单的被改写, 写入方法与一般的寄存器不同。详细请参照「9.5.1 寄存器存取时的注意事项」。

- 定时器控制 / 状态寄存器 0、1 (TCSR_0、TCSR_1)
- 定时计数器 0、1 (TCNT_0、TCNT_1)
- 置位控制 / 状态寄存器 (RSTCSR)

9.2.1 定时器计数器 0、1 (TCNT_0、TCNT_1)

TCNT 是能读 / 写 8 位的累加计数器。TCNT 在计数器控制 / 状态寄存器 (TCSR) 的 TME 位是 0 时, 初始化为 H'00。

9.2.2 定时器控制 / 状态寄存器 0、1 (TCSR_0、TCSR_1)

TCSR 进行向 TCNT 输入时钟、模式的选择。

• TCSR_0

位	位名	初始值	R/W	说明
7	OVF	0	R/(W)*	溢出标志 TCNT 表示溢出。仅在清除标志的 0 清除时。 [置位条件] 在 TCNT 溢出时 (H'FF→H'00)。 但是, 在看门狗定时器模式下, 选择内部置位请求时, 置位后用内部置位自动清除。 [清除条件] 在 OVF=1 的状态下, 读 TCSR 后, 给 OVF 写入 0 时。
6	WT/IT	0	R/W	定时器模式选择 选择作为看门狗定时器使用或间隔定时器使用。 0: 间隔定时器模式 1: 看门狗定时器模式
5	TME	0	R/W	定时器选择 此位置 1 后 TCNT 开始计数。清除后 TCNT 停止计数, 初始化是 H'00。
4、3	—	全为 1	—	保留位 总是读出 1, 不可修改。
2 1 0	CKS2 CKS1 CKS0	0 0 0	R/W R/W R/W	时钟选择 2 ~ 0 选择输入 TCNT 的时钟。() 内表示在 $\phi=20\text{MHz}$ 时的溢出周期。 000: 时钟 $\phi/2$ (周期 25.6s) 001: 时钟 $\phi/64$ (周期 819.2s) 010: 时钟 $\phi/128$ (周期 1.6ms) 011: 时钟 $\phi/512$ (周期 6.6ms) 100: 时钟 $\phi/2048$ (周期 26.2ms) 101: 时钟 $\phi/8192$ (周期 104.9ms) 110: 时钟 $\phi/32768$ (周期 419.4ms) 111: 时钟 $\phi/131072$ (周期 1.68s)

【注】 * 为清除标志只能写 0。

• TCSR_1

位	位名	初始值	R/W	说明
7	OVF	0	R/(W)*	溢出标志 表示 TCNT 溢出。仅在清除标志的 0 清除时。 [置位条件] 在 TCNT 溢出时 (H'FF→H'00)。 但是, 看门狗定时器模式下, 选择内部置位请求时, 置位后用内部置位自动清除。 [清除条件] 在 OVF=1 的状态下, 读 TCSR 后, 给 OVF 写入 0 时。
6	WT/IT	0	R/W	定时器模式选择 选择作为看门狗定时器使用或间隔定时器使用。 0: 间隔定时器模式 1: 看门狗定时器模式
5	TME	0	R/W	定时器选择 此位置 1 后 TCNT 开始计数。清除后 TCNT 停止计数, 初始化是 H'00。
4	PSS	0	R/W	预定标器选择 选择 TCNT 的输入时钟源。 0: TCNT 将 ϕ 基极的预定标器 (PSM) 的分频时钟计数 1: TCNT 将 ϕ SUB 基极的预定标器 (PSS) 的分频时钟计数
3	RST/NMI	0	R/W	置位或 NMI 在看门狗定时器模式下, TCNT 溢出时, 选择请求内部置位和 NMI 中断请求的任一个。 0: 要求 NMI 终止 1: 要求内部置位

位	位名	初始值	R/W	说明
2	CKS2	0	R/W	时钟选择 2 ~ 0 选择输入到 TCNT 的时钟。() 内表示 $\phi = 20\text{MHz}$ (给 LSI 输入 5 MHz 的 4 倍频, $\phi_{\text{SUB}} = 39.1\text{kHz}$) 时的溢出周期, 溢出周期是指 TCNT 从 H'00 开始累加计数到溢出的时间。 在 PSS=0 时 000: 时钟 $\phi/2$ (周期 25.6s) 001: 时钟 $\phi/64$ (周期 819.2s) 010: 时钟 $\phi/128$ (周期 1.6ms) 011: 时钟 $\phi/512$ (周期 6.6ms) 100: 时钟 $\phi/2048$ (周期 26.2ms) 101: 时钟 $\phi/8192$ (周期 104.9ms) 110: 时钟 $\phi/32768$ (周期 419.4ms) 111: 时钟 $\phi/131072$ (周期 1.68s) 在 PSS=1 时 000: 时钟 $\phi_{\text{SUB}}/2$ (周期 13.1ms) 001: 时钟 $\phi_{\text{SUB}}/4$ (周期 26.2ms) 010: 时钟 $\phi_{\text{SUB}}/8$ (周期 52.4ms) 011: 时钟 $\phi_{\text{SUB}}/16$ (周期 104.9ms) 100: 时钟 $\phi_{\text{SUB}}/32$ (周期 209.7ms) 101: 时钟 $\phi_{\text{SUB}}/64$ (周期 419.4ms) 110: 时钟 $\phi_{\text{SUB}}/128$ (周期 838.9ms) 111: 时钟 $\phi_{\text{SUB}}/256$ (周期 1.6777s)
1	CKS1	0	R/W	
0	CKS0	0	R/W	

【注】 * 为清除标志只能写入 0。

9.2.3 复位控制 / 状态寄存器 (RSTCSR)

RSTCSR 是通过 TCNT 的溢出控制内部复位信号的产生，选择内部复位信号的种类。RSTCSR 是用 $\overline{\text{RES}}$ 管脚的复位信号初始化为 H'1F。根据 WDT 的溢出不能初始化内部复位信号。

位	位名	初始值	R/W	说明
7	WOVF	0	R/(W)*	看门狗定时器溢出标志 在看门狗定时器模式下，TCNT 产生溢出后置位。在间接定时器模式下，不能置位。只能写 0。 [置位条件] 在看门狗定时器模式下，TCNT 溢出时 (H'FF→H'00)。 [清除条件] 在读 1 的状态下，写 0 时。
6	RSTE	0	R/W	复位允许 在看门狗定时器模式下，根据 TCNT 的溢出，选择在 LSI 内部是否复位。 0: 即使 TCNT 溢出，内部也不复位。 (本 LSI 内部不复位，但 WDT 内的 TCNT、TCSR 复位。) 1: TCNT 溢出后内部复位。
5	RSTS	0	R/W	复位选择 在看门狗定时器模式下，TCNT 溢出后，选择产生内部复位的种类。 0: 加电复位 1: 设定禁止
4 ~ 0	—	全为 1	—	保留位 总是读出 1。不可修改。

【注】 * 为清除标志只能写入 0。

9.3 工作说明

9.3.1 看门狗定时器模式

作为看门狗定时器模式使用时，设定 TCSR 的 $\overline{\text{WT/IT}}$ 位 = 1，TME 位 = 1。

作为看门狗定时器工作，当由于系统失控等不能改写 TCNT 的值而产生溢出时，输出 WDTOVF 信号。

在系统正常工作时，不产生 TCNT 的溢出。TCNT 溢出前必须改写 TCNT 的值（一般写入 H'00），为了不产生溢出，请编程。

而且，在看门狗定时器模式下，根据 WDTOVF 信号在 LSI 内部能复位。

在 RSTCSR 的 RSTE 位置 1 后，当 TCNT 溢出时，产生 WDTOVF 信号的同时，在 LSI 的内部产生复位信号。复位通过设定 RSTCSR 的 RSTS 位 = 0，选择加电复位。

在根据 $\overline{\text{RES}}$ 管脚的输入信号复位和根据 WDT 的溢出复位同时产生时， $\overline{\text{RES}}$ 管脚优先复位，RSTCSR 的 WOVF 位清 0。

WDTOVF 信号在 RSTCSR 的 RSTE 位 = 1 时的 132 状态和 RSTE 位 = 0 时 130 状态之间输出。

内部复位信号在 518 状态间输出。

在看门狗定时器模式下，TCNT 溢出后，RSTCSR 的 WOVF 位置 1。而且，RSTCSR 的 RSTE 位置 1 后，TCNT 溢出时，对于该 LSI 全部不产生内部复位信号。

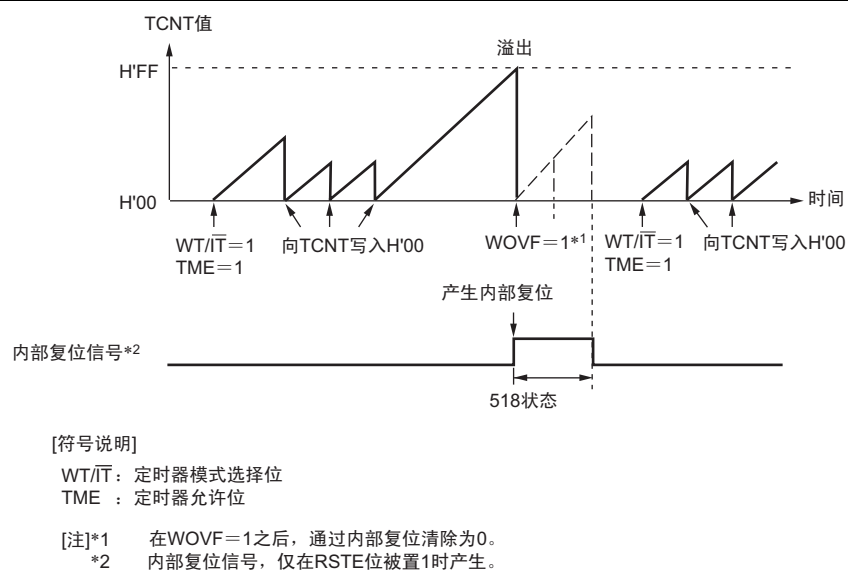


图 9.3(a) WDT_0 看门狗定时器模式时的工作

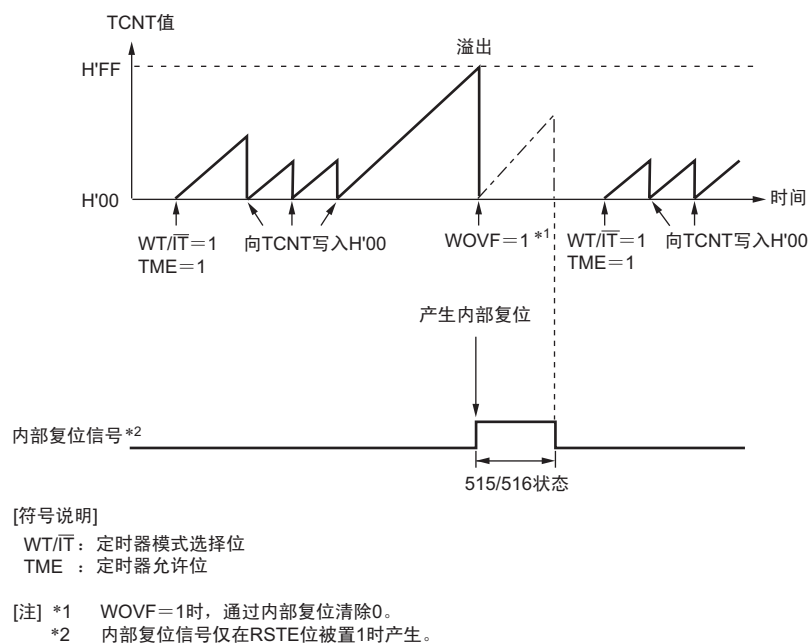


图 9.3 (b) WDT_1 看门狗定时器模式时的工作

9.3.2 间隔定时器模式

作为间隔定时器工作，在 TCNT 在每次溢出时产生间隔定时中断（WOVI）。随之，每隔一定时间能够产生中断。

在间隔定时器模式下，TCNT 溢出后，TCSR 的 OVF 位置 1 的同时要求间隔定时中断（WOVI）。

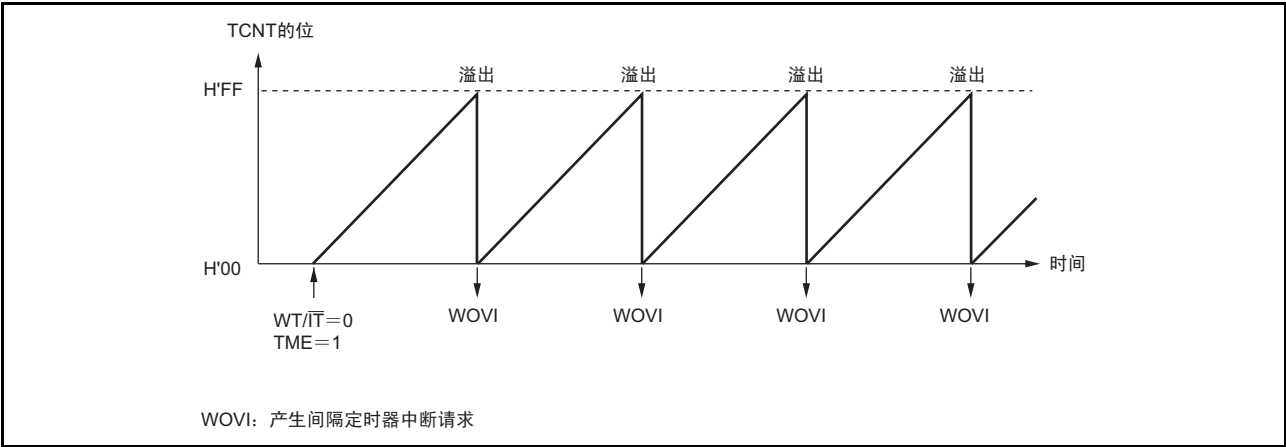


图 9.4 间隔定时器模式时的工作

9.4 中断源

在间隔定时器模式下，根据溢出产生间隔定时中断（WOVI）。在 TCSR 的 OVF 标志置 1 后通常请求间隔定时器中断。在中断处理程序时务必将 OVF 清零。

在看门狗定时器模式下，选择 NMI 中断请求时，根据溢出产生 NMI 中断请求。

表 9.1 WDT 的中断源

名称	中断原因	中断标志
WOVI	TCNT 的溢出（间隔定时器模式）	OVF
NMI	TCNT 的溢出（看门狗定时器模式）	OVF

9.5 使用时的注意事项

9.5.1 寄存器存取时的注意事项

为了 TCNT、TCSR、RSTCSR 不能简单的改写，写方法与一般的寄存器不同。请用以下的方法进行读 / 写。

(1) 向 TCNT、TCSR、RSTCSR 写入

向 TCNT、TCSR 写入时，务必使用字传送指令。字节传送指令下不能写。

写入时，TCNT 和 TCSR 赋值到同一个地址。因此，如图 9.5 所示请转送数据。转送后，低位字节的数据写入 TCNT 或 TCSR。

给 RSTCSR 写时，向地址 H'FF76 进行字转送。在字节转送指令下，不能写入。

给 WOVF 位写 0 和给 RSTE 位和 RSTS 位写时，写的方法不同。因此，如图 9.5 所示请转送数据。

转送后，WOVF 位清零。此时，RSTE、RSTS 位不受影响。给 RSTE、RSTS 位写时，如图 9.5 所示，请转送数据。转送后，低位字节的位 6 和位 5 的值分别写入 RSTE 位和 RSTS 位。此时，WOVF 位不受影响。

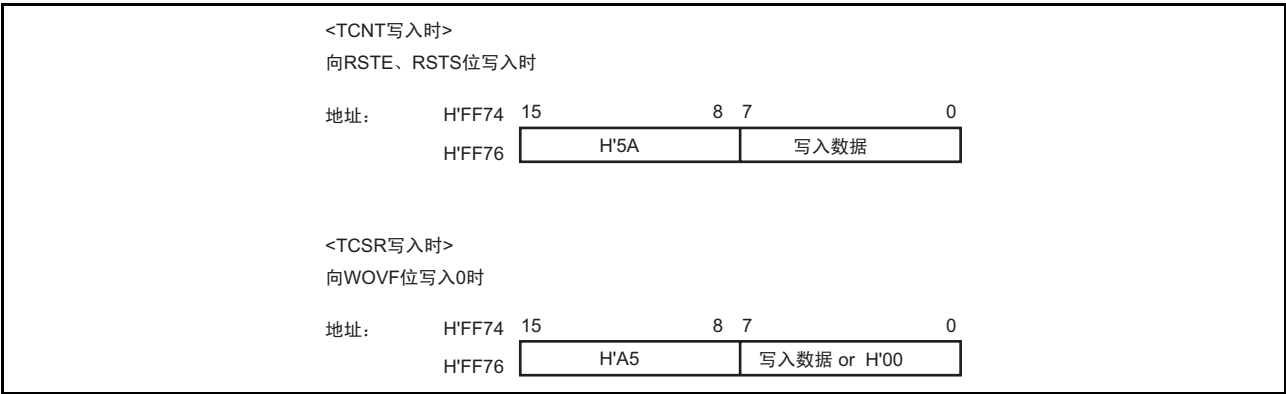


图 9.5 给 TCNT、TCSR、RSTCSR 写 (WDT0 例)

(2) 从 TCNT、TCSR、RSTCSR 读 (WDT0 例)

读后，能用与通常的寄存器同样的方法进行。TCSR 是向地址 H'FF74，TCNT 是向地址 H'FF75，RSTCSR 是向地址 H'FF77 分别赋值。

9.5.2 定时计数器 (TCNT) 的写和累加计数的竞争

即使在 TCNT 写周期中的 T2 状态下产生累加计数, 也不进行累加计数, 向 TCNT 的计数器写入被优先。如图 9.6 所示。

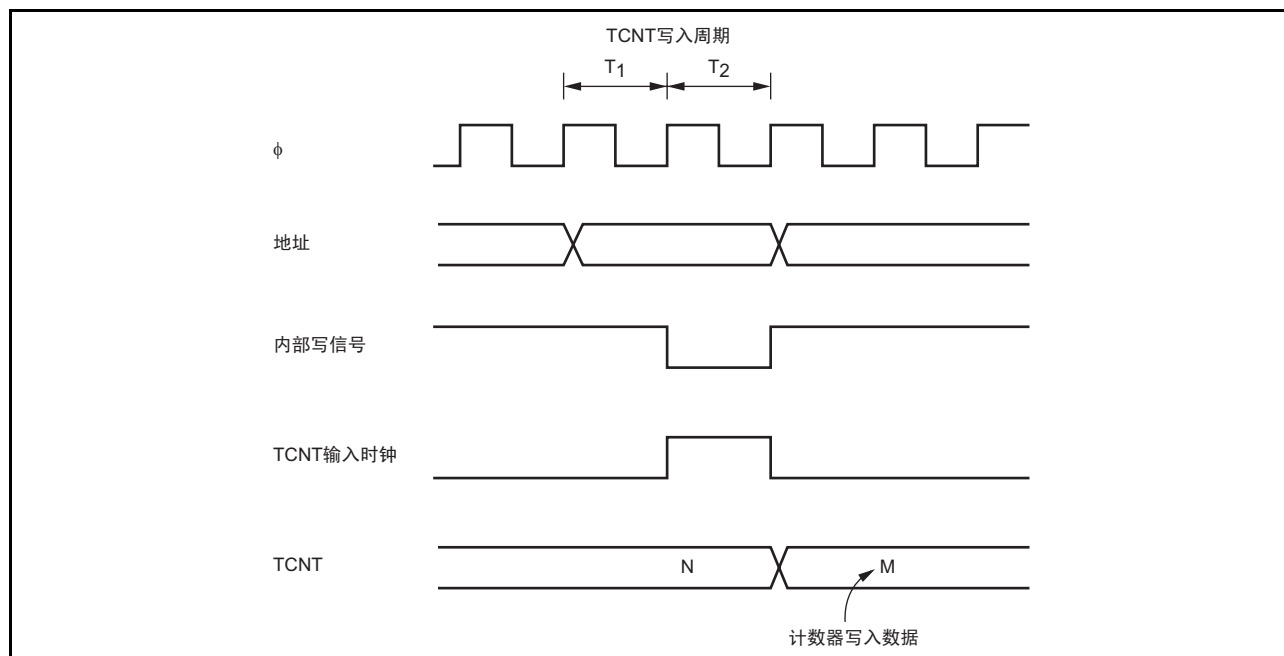


图 9.6 TCNT 的写和累加计数的竞争

9.5.3 CKS2 ~ CKS0 位的改写

WDT 在工作时, 改写 TCSR 的 CKS2 ~ CKS0 位后, 会产生累加计数不能正常进行。改写 CKS2 ~ CKS0 位时, 务必在停止 WDT 后 (TME 位清零后) 进行。

9.5.4 看门狗定时器模式和间隔定时器模式的切换

WDT 在工作时, 看门狗定时器模式和间隔定时器模式切换后, 会产生累加计数不能正常进行。定时器模式的切换, 务必在停止 WDT 后 (TME 位清除为零后) 进行。

9.5.5 看门狗定时器模式下的内部复位

在看门狗定时器模式时, RSTE 位为 0 后, 即使 TCNT 溢出该 LSI 内部也不进行复位。但是 WDT 的 TCNT、TCSR 被复位。

从产生溢出的 132 状态期间, 不能给 TCNT、TCSR、RSTCR 写入。而且, 此期间不认可 WO VF 标志的读取。因此, 清除 WO VF 标志, 从产生溢出开始到等待 132 状态后, 请向 WO VF 标志写 0。

9.5.6 间隔定时器模式下 OVF 标志的清除

间隔定时器模式, 在 OVF 标志的置位和 OVF 标志的读取竞争时, 无论读取 $OVF = 1$ 的状态还是给 OVF 写 0, 会产生不清除标志。在禁止间隔定时器中断、查询 OVF 标志等时, 有可能产生 OVF 标志的置位和读取的竞争, 在清除标志时, 最少 $OVF = 1$ 的状态读 2 次以上后, 给 OVF 写 0。

第 10 章 串行通信接口（SCI）

本 LSI 配备了独立的双通道串行通信接口（SCI: Serial Communication Interface）。SCI 可进行异步和同步两种方式的串行通信。异步方式下，可与通用异步收发设备（UART）和异步通讯接口适配器（ACIA）等标准异步式通信芯片进行串行通信。而且在异步模式还配备有多个处理器间的串行通信功能（多处理器通信功能）。除此之外，SCI 作为异步模式的扩展功能，支持 ISO/IEC 7816-3 标准的智能卡（IC 卡 - Identification Card）接口。SCI 的框图如图 10.1 所示。

10.1 特点

- 串行通信方式可设定为异步或时钟同步模式
- 可执行全双工通信。发送和接收功能相互独立，并具有双缓冲结构。因此，可实现同时连续的发送和接收。
- 内部波特率发生器允许选择任意位速率。
- 作为发送或接收源也可选择外部时钟（除智能卡接口之外）。
- 可选择 LSB 优先 / MSB 优先（除异步模式下的 7 位数据之外。）
- 中断源：4 种。
- 发送结束、发送缓冲空、接收缓冲满、接收错误 4 种中断源。
- 可设定模块待机模式。

异步模式

- 数据长度：可选择 7 位 / 8 位。
- 停止位长度：可选择 1 位 / 2 位。
- 奇偶校验性：可选择偶校验 / 奇校验 / 无奇偶校验。
- 接收错误检测：奇偶校验错误、溢出错误、帧错误。
- 检测中止：发生帧错误时，直接读取 RxD 管脚的电平，就可检测出中止。

同步时钟模式

- 数据长度：8 位
- 检测接收错误：溢出错误

智能卡接口

- 接收时，如检测出奇偶校验错误，会自动发送错误信号
- 发送时，如接收到错误信号，会再次自动发送信息
- 支持直接转换 / 反码转换

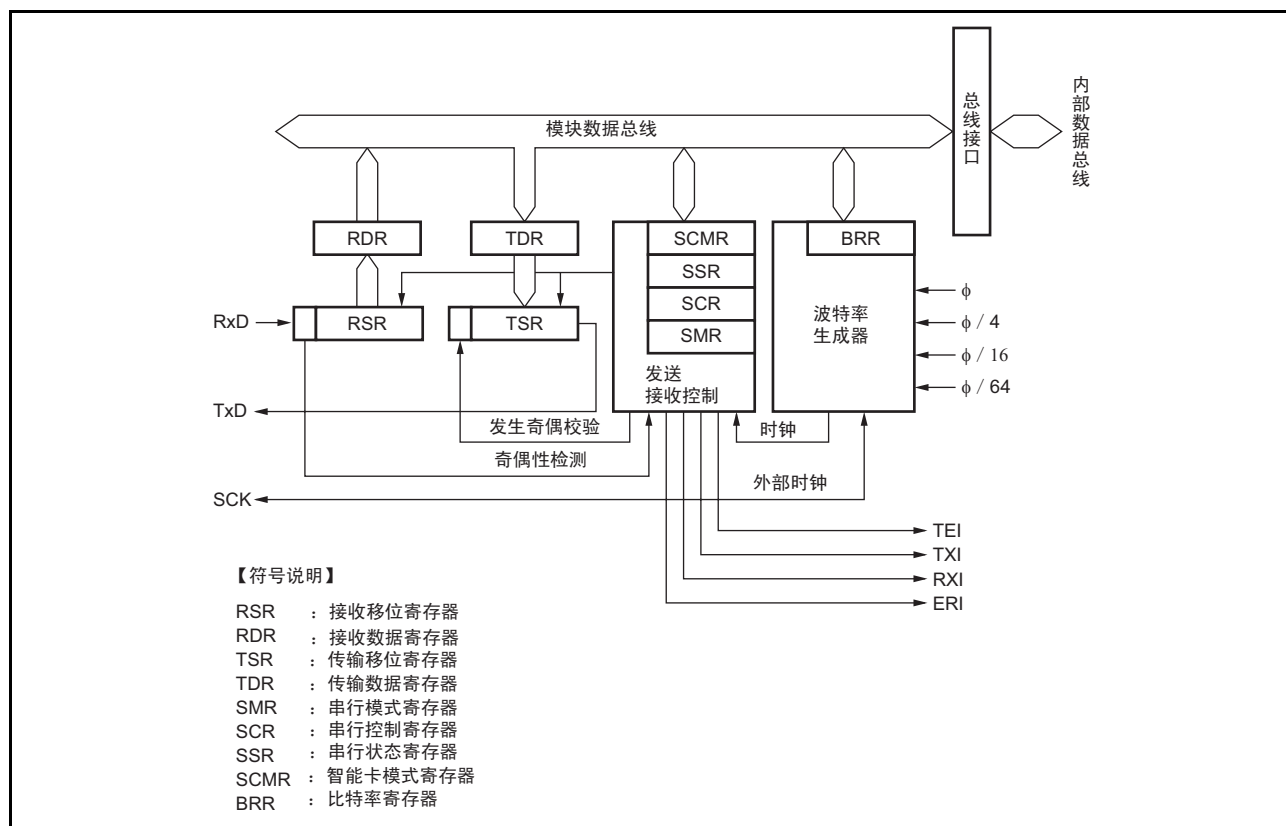


图 10.1 SCI 的框图

10.2 输入 / 输出管脚

SCI 中有如表 10.1 所示的输入 / 输出管脚。

表 10.1 管脚结构

通道	管脚名称 *	输入 / 输出	说明
0	SCK0	输入 / 输出	时钟的输入 / 输出
	RxD0	输入	接收数据的输入
	TxD0	输出	发送数据的输出
1	SCK1	输入 / 输出	时钟的输入 / 输出
	RxD1	输入	接收数据的输入
	TxD1	输出	发送数据的输出

【注】 * 本文中省略了通道，分别以 SCK、RxD、TxD 简称。

10.3 寄存器说明

SCI 的每个通道都有以下寄存器。并且串行模式寄存器（SMR）、串行状态寄存器（SSR）、串行控制寄存器（SCR），在通常的串行通信接口模式和智能卡接口模式中有部分位的功能不同，详情另有记载。

- 接收移位寄存器（RSR）
- 接收数据寄存器（RDR）
- 发送数据寄存器（TDR）
- 发送移位寄存器（TSR）
- 串行模式寄存器（SMR）
- 串行控制寄存器（SCR）
- 串行状态寄存器（SSR）
- 智能卡模式寄存器（SCMR）
- 位速率寄存器（BRR）

10.3.1 接收移位寄存器（RSR）

RSR 是用来接收输入 RxD 管脚的串行数据并对其进行并行转换的移位寄存器。当接收 1 帧的数据后，数据就自动传送到 RDR。此寄存器不能通过 CPU 直接存取。

10.3.2 接收数据寄存器（RDR）

RDR 是保存接收数据的 8 位寄存器。当接收到 1 帧的数据，接收的数据从 RSR 传送到 RDR 寄存器，RSR 就可接收下一个数据。RSR 和 RDR 是双缓冲结构，因此可连续接收信息。确认 SSR 的 RDRF 被置位为 1 后，只需读取一次 RDR。不能通过 CPU 对 RDR 进行写入操作。

10.3.3 发送数据寄存器（TDR）

TDR 是保存发送数据的 8 位寄存器。如果检测到 TSR 为空，写入 TDR 的发送数据会被传送到 TSR，并开始发送数据。TDR 和 TSR 是双缓冲结构，可连续发送数据。发送 1 帧数据时，如果将下一次要发送的数据写入 TDR，则会把它传送到 TSR 继续发送。TDR 通常可通过 CPU 进行读 / 写操作，但是为了正确进行串行发送，必须在确认 SSR 的 TDRE 被置位为 1 后，仅对 TDR 写入 1 次发送数据。

10.3.4 发送移位寄存器 (TSR)

TSR 是发送串行数据的移位寄存器。为了进行串行数据的发送，SCI 首先把发送数据从 TDR 传到 TSR，再发送到 TxD 管脚。不能直接通过 CPU 存取 TSR。

10.3.5 串行模式寄存器 (SMR)

SMR 是用于选择通信形式和内部波特率生成器时钟源的寄存器。SMR 在通常模式和智能卡接口模式，有部分位的功能不同。

- 通常串行通信接口模式 (SCMR 的 SMIF=0 时)

位	位名	初始值	R/W	说明
7	C/A	0	R/W	通信模式 0: 异步模式。 1: 同步时钟模式。
6	CHR	0	R/W	字符长度 (只对异步模式有效) 0: 以 8 位数据长度发送接收。 1: 以 7 位数据长度发送接收。 发送数据时不发送 TDR 的 MSB (7 位)。 同步时钟模式数据长度固定为 8 位。
5	PE	0	R/W	奇偶校验允许 (只针对异步模式有效) 当此位为 1 时, 发送时附加奇偶校验位, 接收时要检查奇偶校验。 在多处理器模式中与此位的设置无关, 既不添加奇偶校验位也不检查。
4	O/E	0	R/W	奇偶模式 (只在异步模式中 PE = 1 时有效) 0: 对发送和接收数据进行偶校验。 1: 对发送和接收数据进行奇校验。
3	STOP	0	R/W	停止位的长度 (只在异步模式下有效) 选择发送时停止位的长度。 0: 1 个停止位 1: 2 个停止位 接收时与此位的设定无关, 只检查停止位的第 1 位, 第 2 位为 0 时, 看作是下一帧的起始位。
2	MP	0	R/W	多处理器模式 (只针对异步模式有效) 此位为 1 时, 可以实现多处理器通信功能。 在多处理器模式下, PE、O/E 位的设定无效。
1 0	CKS1 CKS0	0 0	R/W R/W	选择时钟 1 ~ 0 选择内部波特率发生器的时钟源。 00: ϕ 时钟 (n=0) 01: $\phi/4$ 时钟 (n=1) 10: $\phi/16$ 时钟 (n=2) 11: $\phi/64$ 时钟 (n=3) 此位的设定值和波特率的关系, 请参考「10.3.9 位速率寄存器 (BRR)」。n 是用 10 进制表示的设定值, 表示「10.3.9 位速率寄存器 (BRR)」中 n 的值。

• 智能卡接口 (SCMR 的 SMIF=1 时)

位	位名	初始值	R/W	说明
7	GM	0	R/W	GSM 模式 当此位置为 1, SCI 在 GSM 模式下工作。GSM 模式下 TEND 的设定时间从开始前推到 11.0 μ s, 并追加时钟输出控制功能。详细请参考「10.7.8 时钟输出控制」。
6	BLK	0	R/W	将此位置为 1, SCI 在数据块传送模式下工作。数据块传送模式的详细请参考「10.7.3 数据块传送模式」。
5	PE	0	R/W	奇偶校验允许 (只在异步模式有效) 当此位置为 1 时, 发送要附加发送数据的奇偶位, 接收要检查接收数据的奇偶位。在智能卡接口模式中请把此位置 1 后再使用。
4	O/ $\overline{E}$	0	R/W	奇偶校验模式 (在异步模式下只有 PE = 1 时有效) 0: 对发送和接收数据进行偶校验。 1: 对发送和接收数据进行奇校验。 关于智能卡接口模式中此位的设置方法请参考「10.7.2 数据格式 (除数据块传送模式)」。
3 2	BCP1 BCP0	0	R/W	基本时钟脉冲 1 ~ 0 智能卡接口模式下, 选择传送 1 位数据期间的基本时钟数。 00: 32 时钟 (S=32) 01: 64 时钟 (S=64) 10: 372 时钟 (S=372) 11: 256 时钟 (S=256) 详细内容请参考「10.7.4 采样接收数据时序和智能卡接口模式的接收容限」。S 是「10.3.9 位速率寄存器 (BRR)」中所示的 S 的值。
1 0	CKS1 CKS0	0 0	R/W R/W	时钟选择 1 ~ 0 选择内部波特率发生器的时钟源。 00: ϕ 时钟 (n=0) 01: $\phi/4$ 时钟 (n=1) 10: $\phi/16$ 时钟 (n=2) 11: $\phi/64$ 时钟 (n=3) 关于此位设定值和波特率的关系, 请参考「10.3.9 位速率寄存器 (BRR)」。 n 是用 10 进制表示的设定值, 表示「10.3.9 位速率寄存器 (BRR)」中 n 的值。

10.3.6 串行控制寄存器 (SCR)

SCR 是进行以下发送 / 接收控制、中断控制及发送 / 接收时钟源选择的寄存器。关于各中断请求细节请参考「10.8 中断源」。SCR 在串行接口模式和智能卡模式下有部分位的功能不同。

- 一般串行通信接口模式 (SCMR 的 SMIF=0 时)

位	位名	初始值	R/W	说明
7	TIE	0	R/W	发送中断请求 当该位置 1, 可以接收 TXI 中断请求。
6	RIE	0	R/W	接收中断请求 当该位置 1 时, 可以接收 RXI 及 ERI 中断请求。
5	TE	0	R/W	发送响应 当该位置 1 时, 就可发送信息。
4	RE	0	R/W	接收响应 当该位置 1 时, 就可接收信息。
3	MPIE	0	R/W	多处理器中断响应 (异步模式下 SMR 的 MP=1 时有效) 此位置为 1 时, 跳过多处理器位为 0 时的接收数据, 禁止设定 SSR 的 RDRF、FER、ORER 各状态标记。如果接收到多处理器位为 1 时的数据, 该位会被自动清除, 而返回正常接收状态。详细内容请参考「10.5 多处理器通信功能」。
2	TEIE	0	R/W	发送结束中断请求 将此位置为 1 后, TEI 中断请求被响应。
1 0	CKE1 CKE0	0 0	R/W R/W	时钟允许 1 ~ 0 选择时钟源及 SCK 管脚功能。 异步方式 00: 内部时钟 (SCK 管脚可当作 I/O 端口使用) 01: 内部时钟 (从 SCK 管脚输出与位速率相同频率的时钟) 1X: 外部时钟 (从 SCK 管脚输入频率为位速率 16 倍的时钟。) 时钟同步方式 0X: 内部时钟 (SCK 管脚做为时钟输出管脚。) 1X: 外部时钟 (SCK 管脚做为时钟输入管脚。)

【注】 X: Don't care

• 智能卡接口模式 (SCMR 的 SMIF=1 时)

位	位名	初始值	R/W	说明
7	TIE	0	R/W	发送中断请求 当该位置 1, 可以接收 TXI 中断请求。
6	RIE	0	R/W	接收中断请求 当该位置 1 时, 可以接收 RXI 及 ERI 中断请求。
5	TE	0	R/W	发送响应 如果该位置 1, 就可发送信息。
4	RE	0	R/W	接收响应 如果该位置 1, 就可接收信息。
3	MPIE	0	R/W	多处理器中断响应 (在异步模式下 SMR 的 MP=1 时有效) 智能卡接口模式下, 此位写入 0 后才可使用。
2	TEIE	0	R/W	发送结束中断响应 智能卡接口模式下, 此位写入 0 后才可使用。
1 0	CKE1 CKE0	0 0	R/W R/W	时钟允许 1 ~ 0 该位控制 SCK 管脚的时钟输出。在 GSM 模式中, 可以把时钟输出进行动态地切换。详细请参考 "10.7.8 时钟输出控制"。 SMR 的 GM=0 时 00: 禁止输出 (SCK 管脚可当作 I/O 端口使用) 01: 时钟输出 1X: 保留 SMR 的 GM=1 时 00: 固定为低电平输出 01: 时钟输出 10: 固定为高电平输出 11: 时钟输出

10.3.7 串行通信状态寄存器 (SSR)

SSR 由 SCI 的状态标志和发送接收多处理器位构成。TDRE、RDRF、ORER、PER、FER 只能被清除。SSR 在通常模式和智能卡接口模式有部分位的功能不同。

- 通常串行通信接口模式 (SCMR 的 SMIF=0 时)

位	位名	初始值	R/W	说明
7	TDRE	1	R/(W)*	发送数据寄存器为空 表示 TDR 内是否有发送数据。 [置位条件] • SCR 的 TE 位为 0 时 • 从 TDR 向 TSR 传送数据时 [清除条件] • 当读取 TDRE=1 的状态后, 将置 TDRE=0。
6	RDRF	0	R/(W)*	接收数据寄存器饱和 表明接收数据保存在 RDR 内。 [置位条件] • 接收正常结束, 将接收数据从 RSR 传送到 RDR [清除条件] • 当读取 RDRF=1 后, 将 RDRF 写入 0 即使 SCR 中的 RE 位被清零, RDRF 的状态也不会受到影响。仍保持它的原值。
5	ORER	0	R/(W)*	溢出错误 [置位条件] • 当 RDRF=1 时, 接收到下一个信息 [清除条件] • 当读取 ORER=1 的状态后, 将 ORER 置为 0。
4	FER	0	R/(W)*	帧错误 [置位条件] • 停止位为 0 时 [清除条件] • 读取 FER=1 的状态后, 将写入 FER=0 时 在 2 个停止位的模式下, 也只检查第 1 个停止位。
3	PER	0	R/(W)*	奇偶校验错误 [置位条件] • 接收过程中, 检测出奇偶校验错误 [清除条件] • 读取 PER=1 的状态后, 将写入 PER=0 时
2	TEND	1	R	发送结束 [置位条件] • SCR 的 TE 位为 0 时 • 当 TDRE 为 1 时, 发送字符的最后一位, [清除条件] • 读取 TDRE=1 的状态后, 将 TDRE 写为 0。
1	MPB	0	R	多处理器位 在接收数据时, MPB 保存多处理器位的值。当 SCR 中的 RE 清 0 时, 它的状态不改变。
0	MPBT	0	R/W	多处理器位传送 保存附加在发送数据中多处理器位的值。

【注】 * 只能在要清除标志中写入 0。

• 智能卡接口 (SCMR 的 SMIF=1 时)

位	位名	初始值	R/W	说明
7	TDRE	1	R/(W)*	传输数据寄存器为空 表示 TDR 内是否有发送数据。 [置位条件] • 当 SCR 中的 TE 位为 0 时 • 将数据从 TDR 传送到 TSR 时, 可在 TDR 中写入数据时 [清除条件] • 读取 TDRE=1 的状态后, 将 TDRE 写入 0
6	RDRF	0	R/(W)*	接收数据寄存器满 表明接收数据被存放在 RDR 中。 [置位条件] • 接收正常结束时, 接收数据从 RSR 传送到 RDR [清除条件] • 读取 RDRF=1 的状态后, 将 RDRF 写入 0 即使 SCR 中的 RE 位被清零, RDRF 的状态也不会受到影响。仍保持它的原值。
5	ORER	0	R/(W)*	溢出错误 [置位条件] • 当 RDRF=1 时, 接收到下一个数据 [清除条件] • 读取 ORER=1 的状态后, 将 ORER 写入 0
4	ERS	0	R/(W)*	R/(W)* 错误信号状态 [置位条件] • 采样到错误信号低电平时 [清除条件] • 读取 ERS=1 的状态后, 将 ERS 写入 0
3	PER	0	R/(W)*	奇偶校验错误 [置位条件] • 接收过程中, 检测到奇偶校验错误时 [清除条件] • 读取到 PER=1 的状态后, 将 PER 写入 0
2	TEND	1	R	发送结束 当没有收到来自接收端的错误信号应答时, 到可以向 TDR 发送下一个发送数据时, 才可以将该位设定为 1。 [置位条件] • 当 SCR 的 TE=0 且 ESR=0 时 • 发送 1 字节的数据, 一定时间后 ESR=0 且 TDRE=1 时。 置位的时序根据寄存器的设定有如下的不同。 GM=0、BLK=0 时, 发送开始 2.5etu 后 GM=0、BLK=1 时, 发送开始 1.5etu 后 GM=1、BLK=0 时, 发送开始 1.0etu 后 GM=1、BLK=1 时, 发送开始 1.0etu 后 [清除条件] 读取 TDRE=1 的状态后, 将 TDRE 写入 0
1	MPB	0	R	多处理器位 该位在智能卡接口模式中不能被使用。
0	MPBT	0	R/W	多处理器位传送 智能卡接口模式中, 在此位写入 0 后才可使用。

【注】 * 要清除标志只能写入 0。

10.3.8 智能卡接口模式寄存器（SCMR）

SCMR 是选择智能卡接口及其通信数据格式的寄存器。

位	位名	初始值	R/W	说明
7 ~ 4	—	全为 1	—	保留位 该位读取时为 1。
3	SDIR	0	R/W	智能卡数据传输方向 选择串行 / 并行转换格式。 0: 根据 LSB 优先发送 / 接收 1: 根据 MSB 优先发送 / 接收 仅在发送数据格式为 8 位时有效。如果是 7 位数据时, 固定为 LSB 优先。
2	SINV	0	R/W	智能卡数据转换 数据逻辑层的特定转换。SINV 位不影响奇偶位数据的逻辑层。 为了转换奇偶位数据, 就对 SMR 中的 $O/\bar{E}$ 位进行转换。 0: 直接发送 TDR 的内容, 把接收数据直接保存到 RDR 中 1: TDR 的内容在发送前已经被转换, 接收数据以转换后的格式保存到 RDR 中
1	—	1	—	保留位 该位读取时通常读出 1。
0	SMIF	0	R/W	选择智能卡接口模式 SCI 的操作在智能卡接口模式下工作时该位要置为 1。 0: 通常的异步模式或者同步时钟模式 1: 智能卡接口模式

10.3.9 位速率寄存器（BRR）

BRR 是调整位速率的 8 位寄存器。由于 SCI 每个通道的波特率发生器是相互独立的，因此每个通道可设定不同的位速率。在通常异步模式、时钟同步模式、智能卡接口模式下 BRR 的设定值 N 和位速率的关系如表 10.2 所示。BRR 的初始值是 H'FF，可以随时通过 CPU 进行读 / 写。

表 10.2 BRR 的设定值 N 和位速率 B 的关系

模式	比特率	误差
异步模式	$B = \frac{\phi \times 10^6}{64 \times 2^{2n-1} \times (N+1)}$	$\text{误差}(\%) = \left\{ \frac{\phi \times 10^6}{B \times 64 \times 2^{2n-1} \times (N+1)} - 1 \right\} \times 100$
时钟同步模式	$B = \frac{\phi \times 10^6}{8 \times 2^{2n-1} \times (N+1)}$	——
智能卡接口	$B = \frac{\phi \times 10^6}{S \times 2^{2n+1} \times (N+1)}$	$\text{误差}(\%) = \left\{ \frac{\phi \times 10^6}{B \times S \times 2^{2n+1} \times (N+1)} - 1 \right\} \times 100$

【注】 B：比特率（bit/s）
N：波特率生成器的 BRR 的设定值（ $0 \leq N \leq 255$ ）
 ϕ ：运行频率（MHz）
N 和 S：如下表，由 SMR 的设定值决定。

SMR 的设定值			SMR 的设定值		
CKS1	CKS0	n	BCP1	BCP0	n
0	0	0	0	0	32
0	1	1	0	1	64
1	0	2	1	0	372
1	1	3	1	1	256

设定通常异步模式的 BRR 值 N 的例子如表 10.3，各运行频率中可设定的最大位速率如表 10.4 所示。时钟同步模式的 BRR 值 N 的设定例子如表 10.6，智能卡接口的 BRR 值 N 的设定例子如表 10.8 所示。在智能卡接口可选择传送 1 位期间的基本时钟数 S。详细请参考「10.7.4 采样接收数据时序和智能卡接口模式的接收容限」。表 10.5、表 10.7 所示的是外部时钟输入时的最大位速率。

表 10.3 位速率对应的 BRR 的设定样例（异步模式）

位速率 (bit/s)	运行频率 ϕ (MHz)								
	4			4.9152			5		
	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)
110	2	70	0.03	2	86	0.31	2	88	-0.25
150	1	107	0.16	1	255	0.00	2	64	0.16
300	1	103	0.16	1	127	0.00	1	129	0.16
600	0	207	0.16	0	255	0.00	1	64	0.16
1200	0	103	0.16	0	127	0.00	0	129	0.16
2400	0	51	0.16	0	63	0.00	0	64	0.16
4800	0	25	0.16	0	31	0.00	0	32	-1.36
9600	0	12	0.16	0	15	0.00	0	15	1.73
19200	—	—	—	0	7	0.00	0	7	1.73
31250	0	3	0.00	0	2	-1.70	0	4	0.00
38400	—	—	—	0	3	0.00	0	3	1.73

位速率 (bit/s)	运行频率 ϕ (MHz)											
	6			6.144			7.3728			8		
	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)
110	2	106	-0.44	2	108	0.08	2	130	-0.07	2	141	0.03
150	2	77	0.16	2	79	0.00	2	95	0.00	2	103	0.16
300	1	155	0.16	1	159	0.00	1	191	0.00	1	207	0.16
600	1	77	0.16	1	79	0.00	1	95	0.00	1	103	0.16
1200	0	155	0.16	0	159	0.00	0	191	0.00	0	207	0.16
2400	0	77	0.16	0	79	0.00	0	95	0.00	0	103	0.16
4800	0	38	0.16	0	39	0.00	0	47	0.00	0	51	0.16
9600	0	19	-2.34	0	19	0.00	0	23	0.00	0	25	0.16
19200	0	9	-2.34	0	9	0.00	0	11	0.00	0	12	0.16
31250	0	5	0.00	0	5	2.40	—	—	—	0	7	0.00
38400	0	4	-2.34	0	4	0.00	0	5	0.00	—	—	—

位速率 (bit/s)	运行频率 ϕ (MHz)											
	9.8304			10			12			12.288		
	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)
110	2	174	-0.26	2	177	-0.25	2	212	0.03	2	217	0.08
150	2	127	0.00	2	129	0.16	2	155	0.16	2	159	0.00
300	1	255	0.00	2	64	0.16	2	77	0.16	2	79	0.00
600	1	127	0.00	1	129	0.16	1	155	0.16	1	159	0.00
1200	0	255	0.00	1	64	0.16	1	77	0.16	1	79	0.00
2400	0	127	0.00	0	129	0.16	0	155	0.16	0	159	0.00
4800	0	63	0.00	0	64	0.16	0	77	0.16	0	79	0.00
9600	0	31	0.00	0	32	-1.36	0	38	0.16	0	39	0.00
19200	0	15	0.00	0	15	1.73	0	19	-2.34	0	19	0.00
31250	0	9	-1.70	0	9	0.00	0	11	0.00	0	11	2.40
38400	0	7	0.00	0	7	1.73	0	9	-2.34	0	9	0.00

位速率 (bit/s)	运行频率 ϕ (MHz)											
	14			14.7456			16			17.2032		
	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)
110	2	248	-0.17	3	64	0.70	3	70	0.03	3	75	0.48
150	2	181	0.13	2	191	0.00	2	207	0.13	2	223	0.00
300	2	90	0.13	2	95	0.00	2	103	0.13	2	111	0.00
600	1	181	0.13	1	191	0.00	1	207	0.13	1	223	0.00
1200	1	90	0.13	1	95	0.00	1	103	0.13	1	111	0.00
2400	0	181	0.13	0	191	0.00	0	207	0.13	0	223	0.00
4800	0	90	0.13	0	95	0.00	0	103	0.13	0	111	0.00
9600	0	45	-0.93	0	47	0.00	0	51	0.13	0	55	0.00
19200	0	22	-0.93	0	23	0.00	0	25	0.13	0	27	0.00
31250	0	13	0.00	0	14	-0.17	0	15	0.00	0	13	1.20
38400	—	—	—	0	11	0.00	0	12	0.13	0	13	0.00

位速率 (bit/s)	运行频率 ϕ (MHz)								
	18			19.6608			20		
	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)
110	3	79	-0.12	3	86	0.31	3	88	-0.25
150	2	233	0.16	2	255	0.00	3	64	0.16
300	2	116	0.16	2	127	0.00	2	129	0.16
600	1	233	0.16	1	255	0.00	2	64	0.16
1200	1	116	0.16	1	127	0.00	1	129	0.16
2400	0	233	0.16	0	255	0.00	1	64	0.16
4800	0	116	0.16	0	127	0.00	0	129	0.16
9600	0	58	-0.69	0	63	0.00	0	64	0.16
19200	0	28	1.02	0	31	0.00	0	32	-1.36
31250	0	17	0.00	0	19	-1.70	0	19	0.00
38400	0	14	-2.34	0	15	0.00	0	15	1.73

表 10.4 各运行频率中最大位速率 (异步模式)

ϕ (MHz)	最大位速率 (bit/s)	n	N	ϕ (MHz)	最大位速率 (bit/s)	n	N
4	125000	0	0	12	375000	0	0
4.9152	153600	0	0	12.288	384000	0	0
5	156250	0	0	14	437500	0	0
6	187500	0	0	14.7456	460800	0	0
6.144	192000	0	0	16	500000	0	0
7.3728	230400	0	0	17.2032	537600	0	0
8	250000	0	0	18	562500	0	0
9.8304	307200	0	0	19.6608	614400	0	0
10	312500	0	0	20	625000	0	0

表 10.5 外部时钟输入时的最大位速率 (异步模式)

ϕ (MHz)	外部输入时钟 (MHz)	最大位速率 (bit/s)	ϕ (MHz)	外部输入时钟 (MHz)	最大位速率 (bit/s)
4	1.0000	62500	12	3.0000	187500
4.9152	1.2288	76800	12.288	3.0720	192000
5	1.2500	78125	14	3.5000	218750
6	1.5000	93750	14.7456	3.6864	230400
6.144	1.5360	96000	16	4.000	250000
7.3728	1.8432	115200	17.2032	4.3008	268800
8	2.0000	125000	18	4.5000	281250
9.8304	2.4576	153600	19.6608	4.9152	307200
10	2.5000	156250	20	5.0000	312500

表 10.6 位速率对应的 BRR 的设定例子 (时钟同步模式)

位速率 (bit/s)	运行频率 ϕ (MHz)									
	4		8		10		16		20	
	n	N	n	N	n	N	n	N	n	N
110	—	—								
250	2	249	3	124	—	—	3	249		
500	2	124	2	249	—	—	3	124	—	—
1k	1	249	2	124	—	—	2	249	—	—
2.5k	1	99	1	199	1	249	2	99	2	124
5k	0	199	1	99	1	124	1	199	1	249
10k	0	99	0	199	0	249	1	99	1	124
25k	0	39	0	79	0	99	0	159	0	199
50k	0	19	0	39	0	49	0	79	0	99
100k	0	9	0	19	0	24	0	39	0	49
250k	0	3	0	7	0	9	0	15	0	19
500k	0	1	0	3	0	4	0	7	0	9
1M	0	0*	0	1	0		0	3	0	4
2.5M					0	0*			0	1
5M									0	0*

【符号说明】

空栏 : 不能设定。

— : 可设定, 但有误差。

* : 不能连续发送 / 接收信息。

【注】请尽量把误差保持在 1% 以内进行设定。

表 10.7 外部时钟输入时的最大位速率 (时钟同步模式)

ϕ (MHz)	外部输入时钟 (MHz)	最大位速率 (bit/s)	ϕ (MHz)	外部输入时钟 (MHz)	最大位速率 (bit/s)
4	0.6667	666666.7	14	2.3333	2333333.3
6	1.0000	1000000.0	16	2.6667	2666666.7
8	1.3333	1333333.3	18	3.0000	3000000.0
10	1.6667	1666666.7	20	3.3333	3333333.3
12	2.0000	2000000.0			

表 10.8 位速率对应的 BRR 的设定例子 (在智能卡接口模式下 n=0、S=372 时)

位速率 (bit/s)	运行频率 ϕ (MHz)											
	7.1424			10.00			10.7136			13.00		
	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)
9600	0	0	0.00	0	1	30	0	1	25	0	1	8.99

位速率 (bit/s)	运行频率 ϕ (MHz)											
	14.2848			16.00			18.00			20.00		
	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)
9600	0	1	0.00	0	1	12.01	0	2	15.99	0	2	6.60

表 10.9 各运行频率中的最大位速率 (在智能卡接口模式下 S=372 时)

ϕ (MHz)	最大位速率 (bit/s)	n	N	ϕ (MHz)	最大位速率 (bit/s)	n	N
7.1424	9600	0	0	14.2848	19200	0	0
10.00	13441	0	0	16.00	21505	0	0
10.7136	14400	0	0	18.00	24194	0	0
13.00	17473	0	0	20.00	26882	0	0

10.4 异步模式的运行

异步模式串行通信的一般格式如图 10.2 所示。1 帧是从开始位 (低电平状态) 开始, 按照接收发送数据、奇偶校验位、停止位 (高电平状态) 的顺序构成。异步模式串行通信中, 通信线路通常保持屏蔽状态 (高电平状态)。SCI 监视通信线路, 一旦检测出空白 (低电平状态), 则视为开始位开始串行通信。在 SCI 内部, 发送和接收部是独立的, 可进行全双工通信。且发送和接收部都采用了双缓冲结构, 可以读 / 写发送及接收中的数据, 也可以连续发送 / 接收数据。

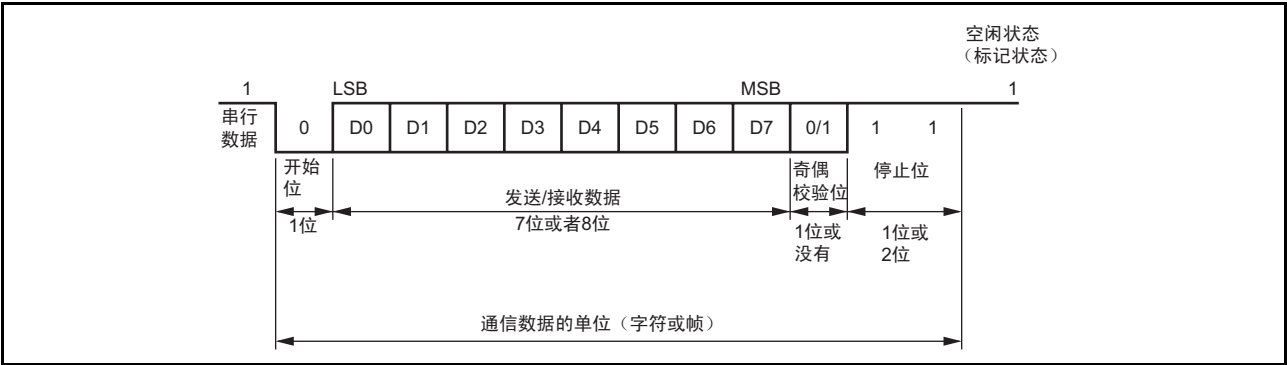


图 10.2 异步模式通信的数据格式 (8 位数据 / 有奇偶校验 / 2 停止位的例子)

10.4.1 发送 / 接收格式

在异步模式下，可设定的发送 / 接收信息格式如表 10.10 所示。格式有 12 种，可根据 SMR 的选定来选择。关于多处理器位，请参考「10.5 多处理器通信功能」。

表 10.10 串行发送 / 接收格式（异步模式）

SMR 的设定				串行发送/接收格式和帧长											
CHR	PE	MP	STOP	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	0	S	8 位数据								STOP		
0	0	0	1	S	8 位数据								STOP	STOP	
0	1	0	0	S	8 位数据								P	STOP	
0	1	0	1	S	8 位数据								P	STOP	STOP
1	0	0	0	S	7 位数据							STOP			
1	0	0	1	S	7 位数据							STOP	STOP		
1	1	0	0	S	7 位数据							P	STOP		
1	1	0	1	S	7 位数据							P	STOP	STOP	
0	—	1	0	S	8 位数据								MPB	STOP	
0	—	1	1	S	8 位数据								MPB	STOP	STOP
1	—	1	0	S	7 位数据							MPB	STOP		
1	—	1	1	S	7 位数据							MPB	STOP	STOP	

【符号说明】

S : 开始位

STOP : 停止位

P : 奇偶位

MPB : 多处理器位

10.4.2 异步模式下接收数据的采样时序和接收容限

异步模式下，SCI 运行的基本时钟的频率是 16 倍的位速率。接收信息时用基本时钟采样开始位的下降沿使内部同步。如图 10.3 所示在基本时钟的第 8 上升沿采样接收数据，在各位中央调出数据。因此在异步模式的接收容限如公式（1）所示。

$$M = \left| \left(0.5 - \frac{1}{2N} \right) - \frac{D-0.5}{N} - (L-0.5) F \right| \times 100 \quad (\%) \quad \cdots \text{公式(1)}$$

- M: 接收容限
- N: 对于时钟的位速率比（N=16）
- D: 时钟的占空比（D=0.5 ~ 1.0）
- L: 帧长度（L=9 ~ 12）
- F: 时钟频率偏差的绝对值

在公式（1）中，F（时钟频率偏差的绝对值）= 0、D（时钟的占空比）= 0.5、

$$M = \{ 0.5 - 1 / (2 \times 16) \} \times 100 \quad (\%) = 46.875 \%$$

但是，此值完全是理论上的值，在系统设计时请保持 20 ~ 30% 的余地。

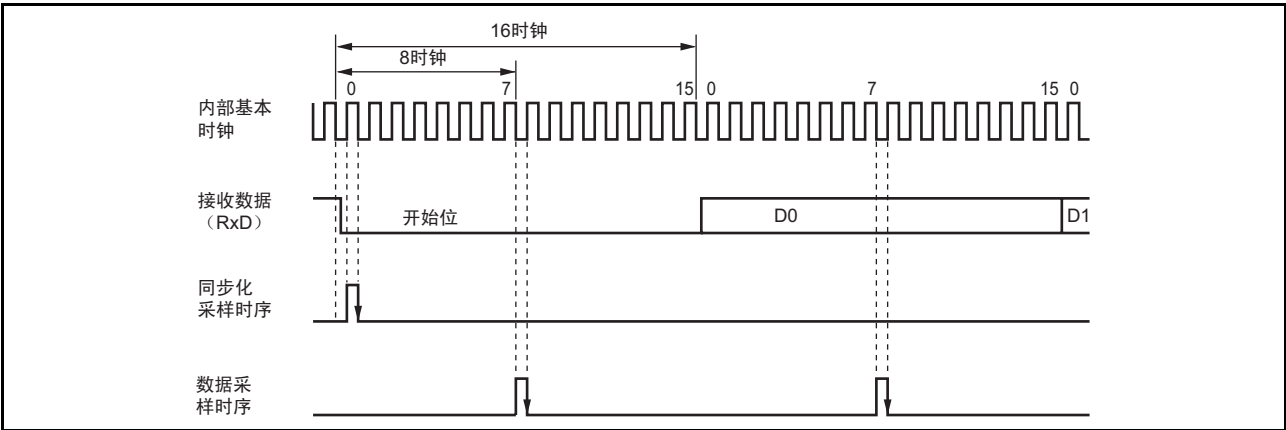


图 10.3 异步模式的接收采样时序

10.4.3 时钟

通过设定 SMR 的 C/A 位和 SCR 的 CKE1、CKE0 位，可任意选择内部波特率生成器生成的内部时钟或从 SCK 管脚输入的外部时钟，作为 SCI 的发送 / 接收时钟。使用外部时钟时，请在 SCK 管脚输入频率为位速率的 16 倍的时钟。

在内部时钟运行时，就可从 SCK 管脚输出时钟。此时输出的时钟频率与位速率相等，发送信息时的相位如图 10.4 所示在发送信息的中央，时钟上升。

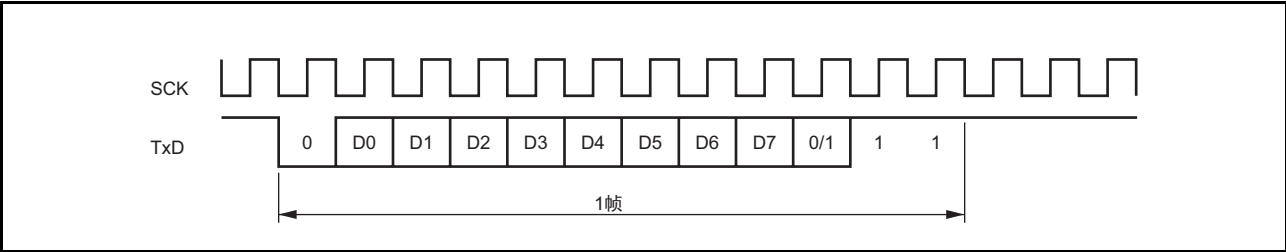


图 10.4 输出时钟和发送数据的相位关系 (异步模式)

10.4.4 SCI 的初始化 (异步模式)

发送 / 接收数据前，首先把 SCR 的 TE、RE 位清零，然后按照图 10.5 的流程进行初始化。必须将 TE 位及 RE 位清零后，再按照下面的顺序变更运行模式、通信格式等。应特别注意的是，如果 TE 清零，TDRE 就被置为 1，即使将 RE 也清零，RDRF、PER、FER、ORER 的各标志及 RDR 的内容也不会被初始化。在异步模式下使用外部时钟时，必须从初始化开始持续保持时钟。

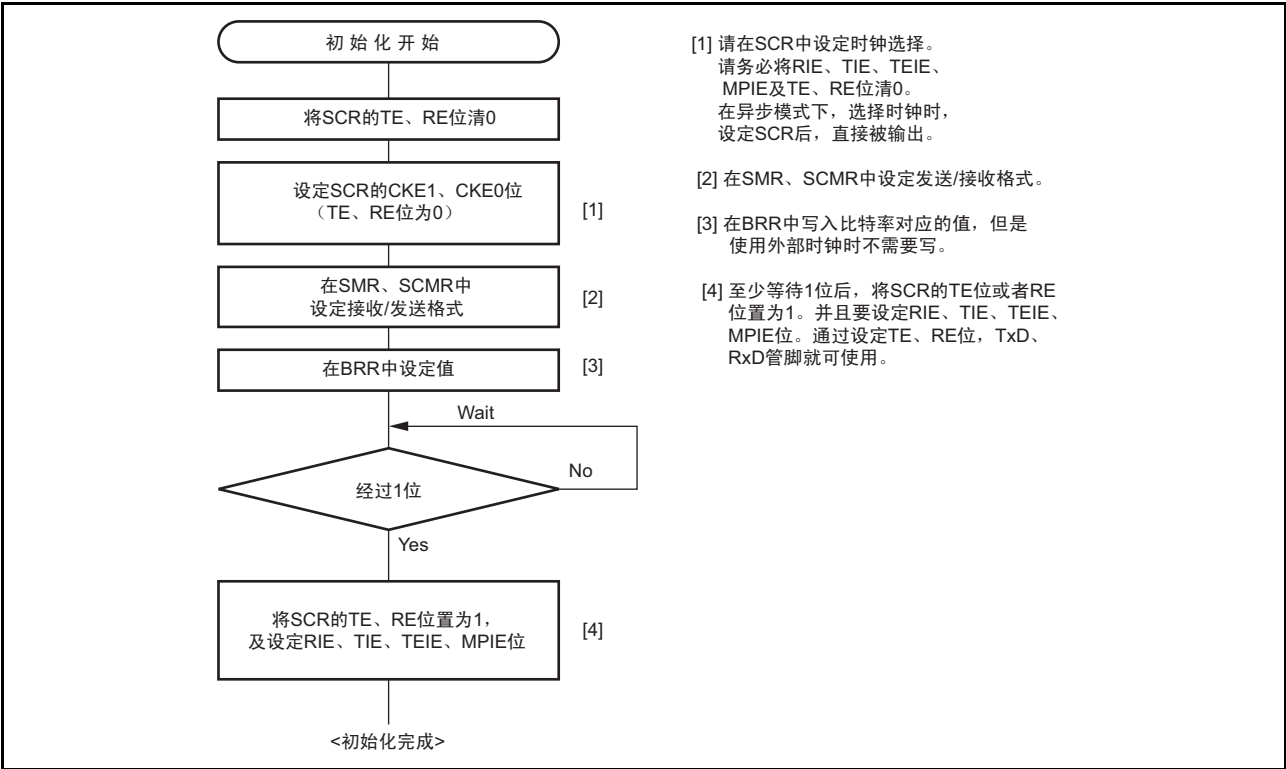


图 10.5 SCI 的初始化流程图

10.4.5 发送数据 (异步模式)

图 10.6 所示的是异步模式发送的运行例子。发送数据时，SCI 运行步骤如下。

1. SCI 监视 SSR 的 TDRE，如果被清零，则认为数据已被写入 TDR，将数据从 TDR 传送到 TSR 中。
2. 将数据从 TDR 传送到 TSR 后，SCI 把 TDRE 置为 1，开始发送数据。这时，如果 SCR 的 TIE 被置为 1，TXI 就发出一个发送数据为空的中断请求。在 TXI 中断程序中，前一个传送的数据发送完成后，在 TDR 中写入下一个发送数据，就可以连续发送信息。
3. 从 TxD 管脚按照起始位、发送数据、奇偶位或者多处理器位（根据格式决定该位有无）、停止位的顺序发送。
4. 在发送出停止位的同时 SCI 检查 TDRE。
5. 如果 TDRE 为 0，将发送数据从 TDR 传送到 TSR，直到停止位发送完成后，开始下一帧的发送。
6. 如果 TDRE 为 1，把 SSR 的 TEND 置为 1，停止位送出后，输出 1，变为标记状态。此时如果 SCR 的 TEIE 被置为 1，就发生 TEI 中断请求。

图 10.7 所示的是发送数据的流程图。

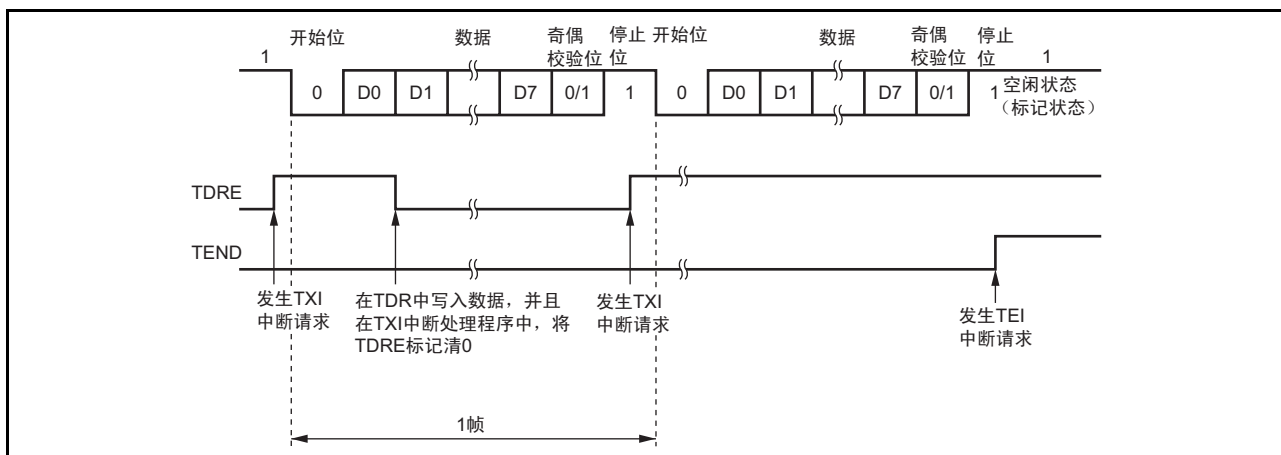


图 10.6 异步模式下发送时的运行例子 (8 位数据 / 有奇偶性 / 1 停止位)

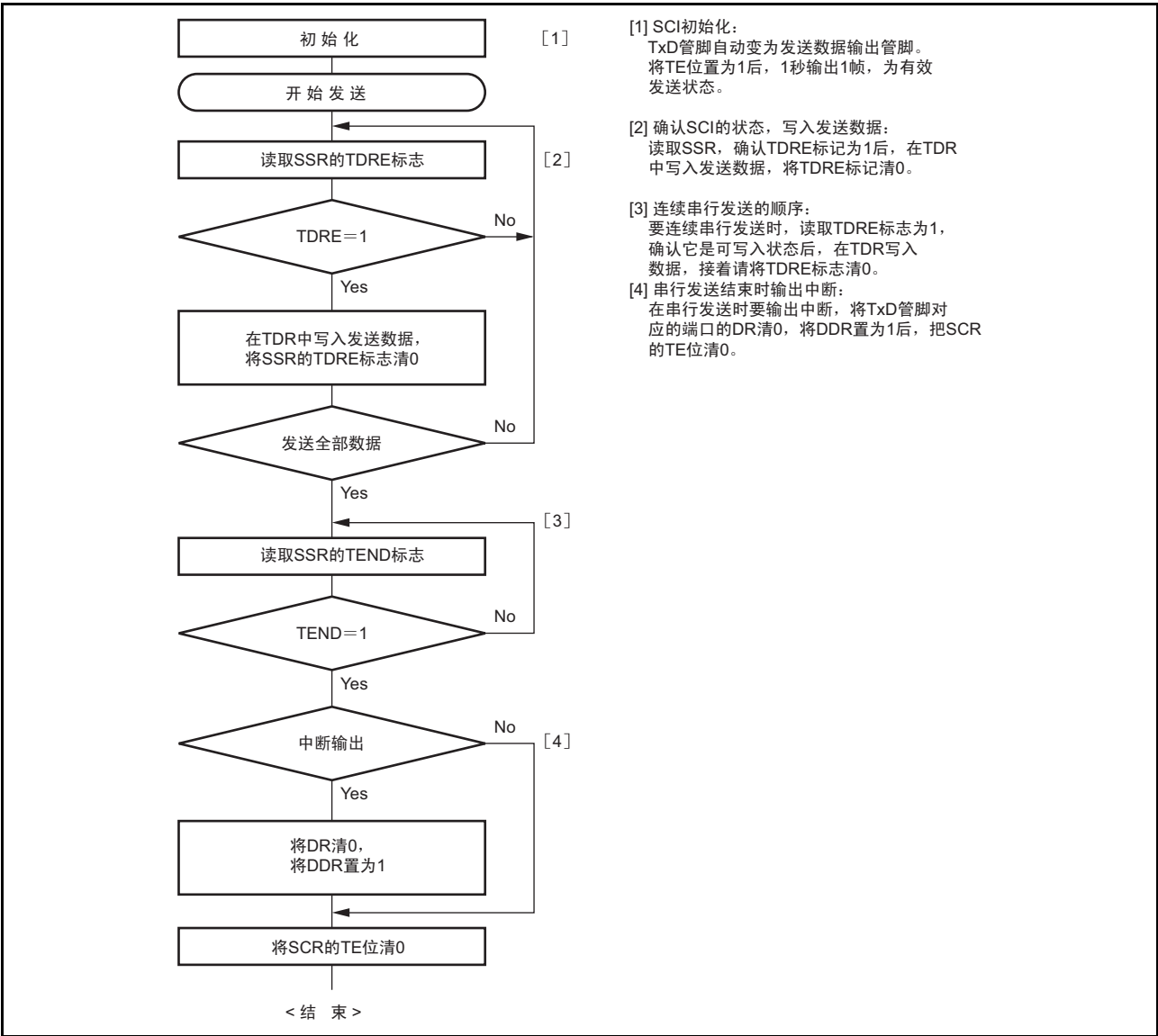


图 10.7 串行发送信息的流程图

10.4.6 接收串行数据 (异步模式)

图 10.8 所示的是异步模式下发送的运行例子。接收时 SCI 运行步骤如下。

1. 监视通信线路，如果检测出起始位，SCI 使内部同步，将接收数据传送到 RSR，并检查奇偶校验位和停止位。
2. 发生溢出错误时（在 SSR 的 RDRF 被置为 1 的状态下，完成下一个数据的接收），要设置 SSR 的 ORER 为 1。此时如果 SCR 的 RIE 被置为 1，就会发生 ERI 中断请求。接收的数据不会被传送到 RDR，且 RDRF 保持置 1 的状态。
3. 检测出奇偶校验错误时，要将 SSR 的 PER 置 1，将接收数据传送到 RDR。此时，如果此时 SCR 的 RIE 被置为 1，就发生 ERI 中断请求。
4. 检测出帧错误时（停止位为 0），SSR 的 FER 位被置 1，将接收数据传送到 RDR。此时，如果 SCR 的 RIE 被置为 1，就发生 ERI 中断请求。
5. 正常接收时，SSR 的 RDRF 被置位，并将接收数据传送到 RDR。此时，如果 SCR 的 RIE 被置为 1，就发生 RXI 中断请求。可以通过该 RXI 中断处理程序，在下一个数据接收完成前，读取传送到 RDR 中的接收数据，就可连续接收信息。

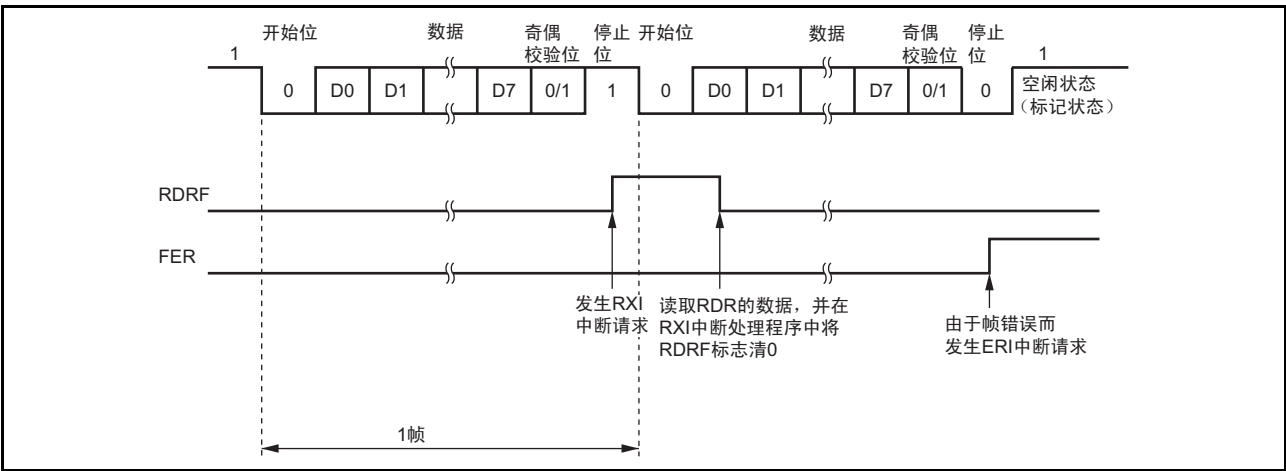


图 10.8 SCI 接收信息时的运行例子 (8 位数据 / 有奇偶性 / 1 停止位)

在检测到接收错误时，SSR 各状态标志的状态和接收数据的处理如表 10.11 所示。一旦检测出接收错误，RDRF 就保持接收数据前的状态。在接收错误标志被置位的状态，不能进行以后的接收操作。因此，在继续接收数据前，必须将 ORER、FER、PER 及 RDRF 清零。图 10.9 所示的是接收数据的流程图。

表 10.11 SSR 的状态标记状态和接收数据的处理

SSR 的状态标志				接收数据	接收错误的状态
RDRF*	ORER	FER	PER		
1	1	0	0	丢失	溢出错误
0	0	1	0	传送到 RDR	帧错误
0	0	0	1	传送到 RDR	奇偶校验错误
1	1	1	0	丢失	溢出错误 + 帧错误
1	1	0	1	丢失	溢出错误 + 奇偶校验错误
0	0	1	1	传送到 RDR	帧错误 + 奇偶校验错误
1	1	1	1	丢失	溢出错误 + 帧错误 + 奇偶校验错误

【注】 *RDRF 保持接收数据前的状态。

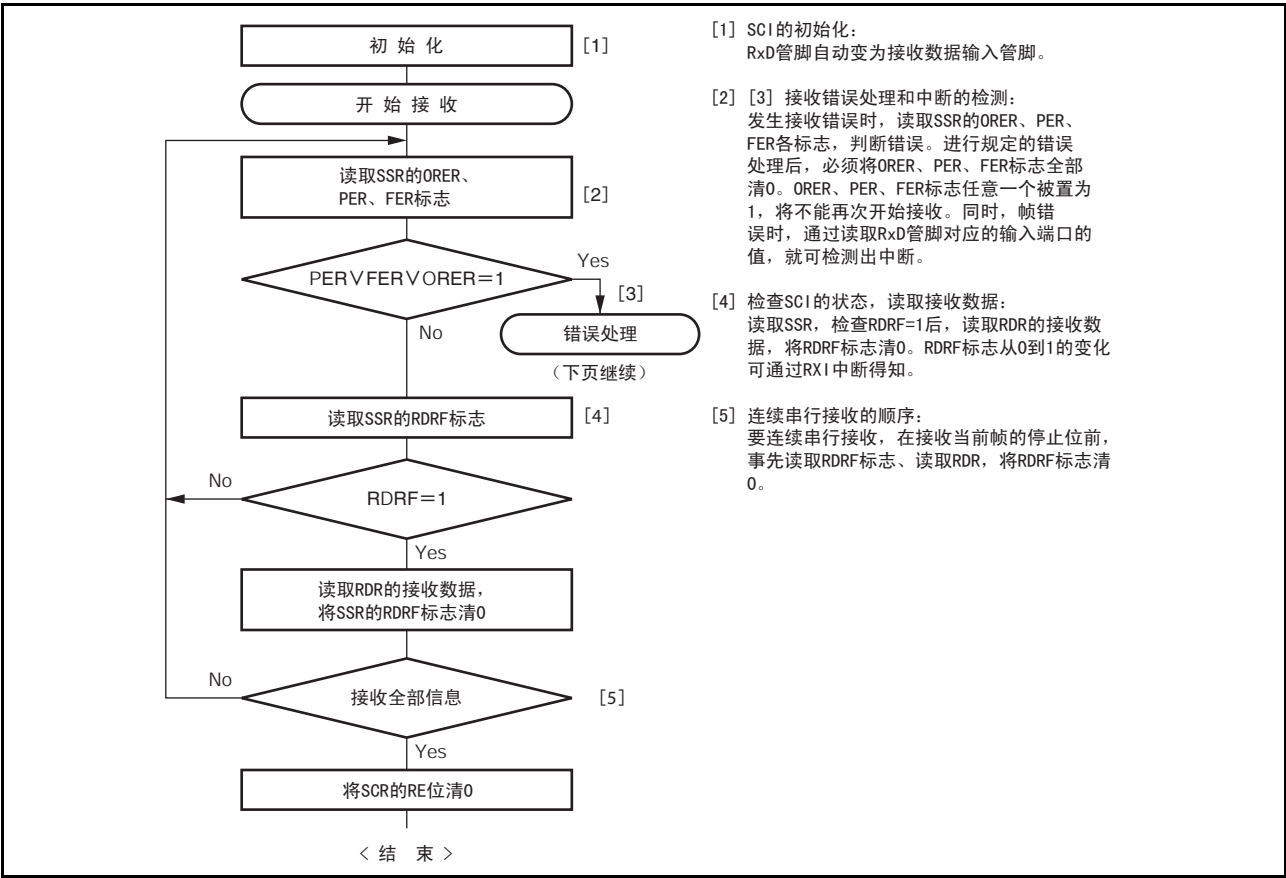


图 10.9 串行接收数据流程图（1）

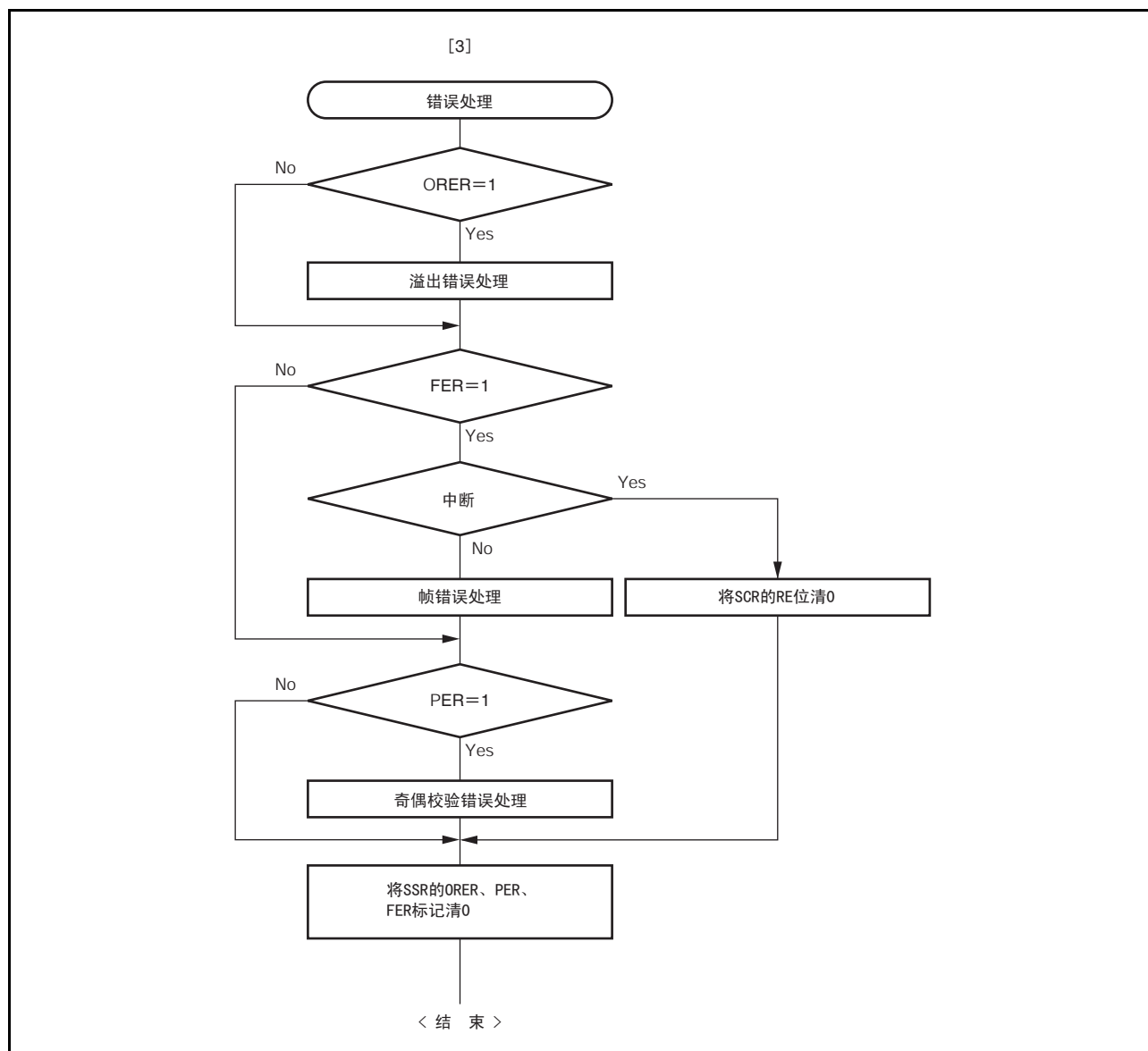


图 10.9 串行接收数据流程图 (2)

10.5 多处理器通信功能

如果使用多处理器通信功能，就能通过附加了多处理器位的异步串行通信，在多个处理器之间共用通信线路，进行数据的发送和接收。多处理器通信时，给各接收站分别分配了其固有的 ID 代码。串行通信周期由指定接收站的 ID 发送周期和对应的数据发送周期构成。ID 发送周期和数据发送周期的区别在于是否以多处理器位运行。多处理器位为 1 时，是 ID 发送周期，多处理器位为 0 时，是数据发送周期。图 10.10 所示的是使用了多处理器格式的处理器之间通信的样例。发送站首先发送将多处理器位为 1 的数据附加于接收站 ID 代码的通信数据，接着发送将多处理器位为 0 的数据附加于发送数据的通信数据。接收站接收到多处理器位为 1 的通信数据后，接收站与本站 ID 进行比较，一致时继续接收被发送来的通信数据。不一致时，就在再次接收到多处理器位为 1 的通信数据前，跳过通信数据。

SCI 为支持此功能，在 SCR 中配置了 MPIE 位。如果将 MPIE 置为 1，在接收到多处理器位为 1 的数据前，禁止从 RSR 向 RDR 传送接收数据、禁止检测接收错误和设定 SSR 的 RDRF、FER、ORER 各状态标志。接收到多处理器位为 1 的接收字符后，SSR 的 MPBR 被置为 1，同时 MPIE 被自动清零，并返回通常的接收状态。此时如果 SCR 的 RIE 被置 1，就发生 RXI 中断。

指定了多处理器格式时，再指定奇偶校验位是无效的。其它位的设置都和通常异步模式相同。进行多处理器通信时的时钟设置也和通常异步模式的时钟相同。

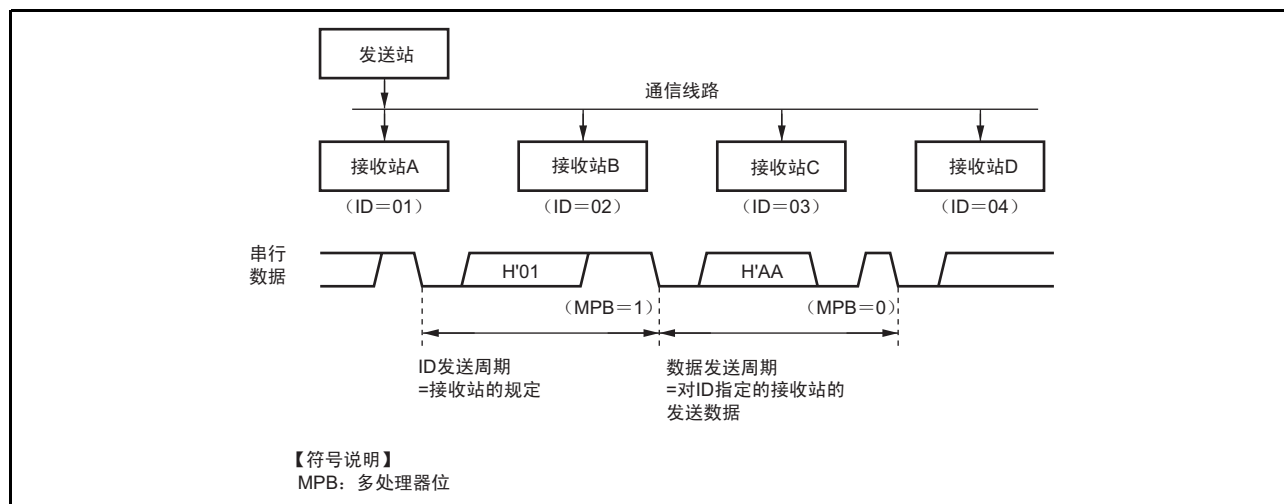


图 10.10 使用了多处理器格式的通信例子（向接收站 A 发送数据 H'AA）

10.5.1 多处理器模式串行数据的发送

图 10.11 中所示的是多处理器模式发送数据的流程图。在 ID 发送周期将 SSR 的 MPBT 置为 1，再发送数据。在数据发送周期将 SSR 的 MPBT 清零，再发送。其它的 SCI 操作均与异步模式的操作相同。

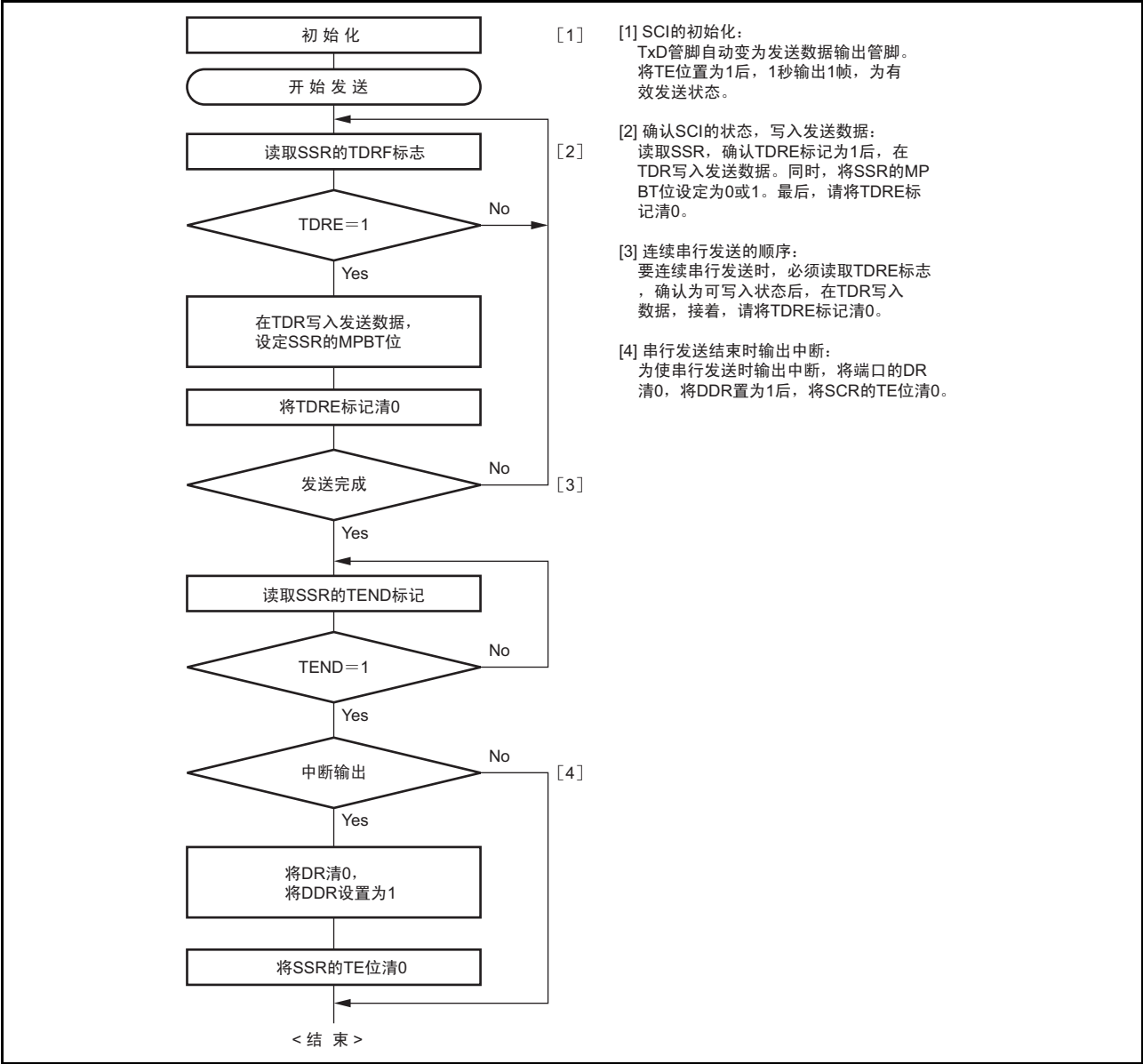


图 10.11 多处理器串行发送的流程图

10.5.2 多处理器模式串行数据的接收

图 10.13 所示的是多处理器模式串行数据接收流程图。如果将 SCR 的 MPIE 置 1，在接收到多处理器位为 1 的通信数据前，跳过通信数据。接收到多处理器位为 1 的通信数据后，将接收数据传送到 RDR。这时会发生 RXI 中断请求。其它的 SCI 操作均与异步模式的操作相同。图 10.12 所示的 SCI 多处理器接收数据时的操作样例。

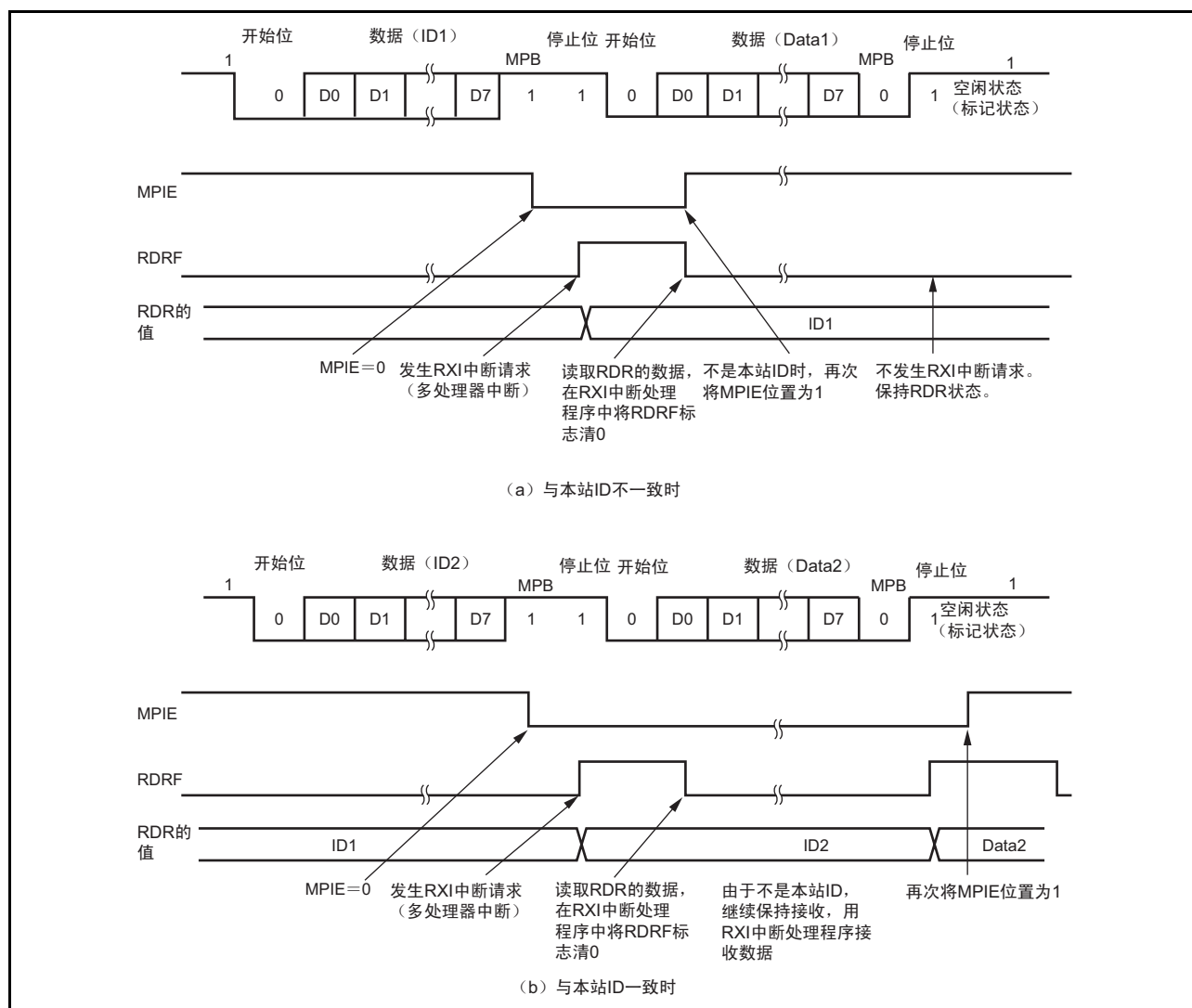


图 10.12 SCI 接收的运行 (8 位数据 / 有多处理器位 / 1 停止位)

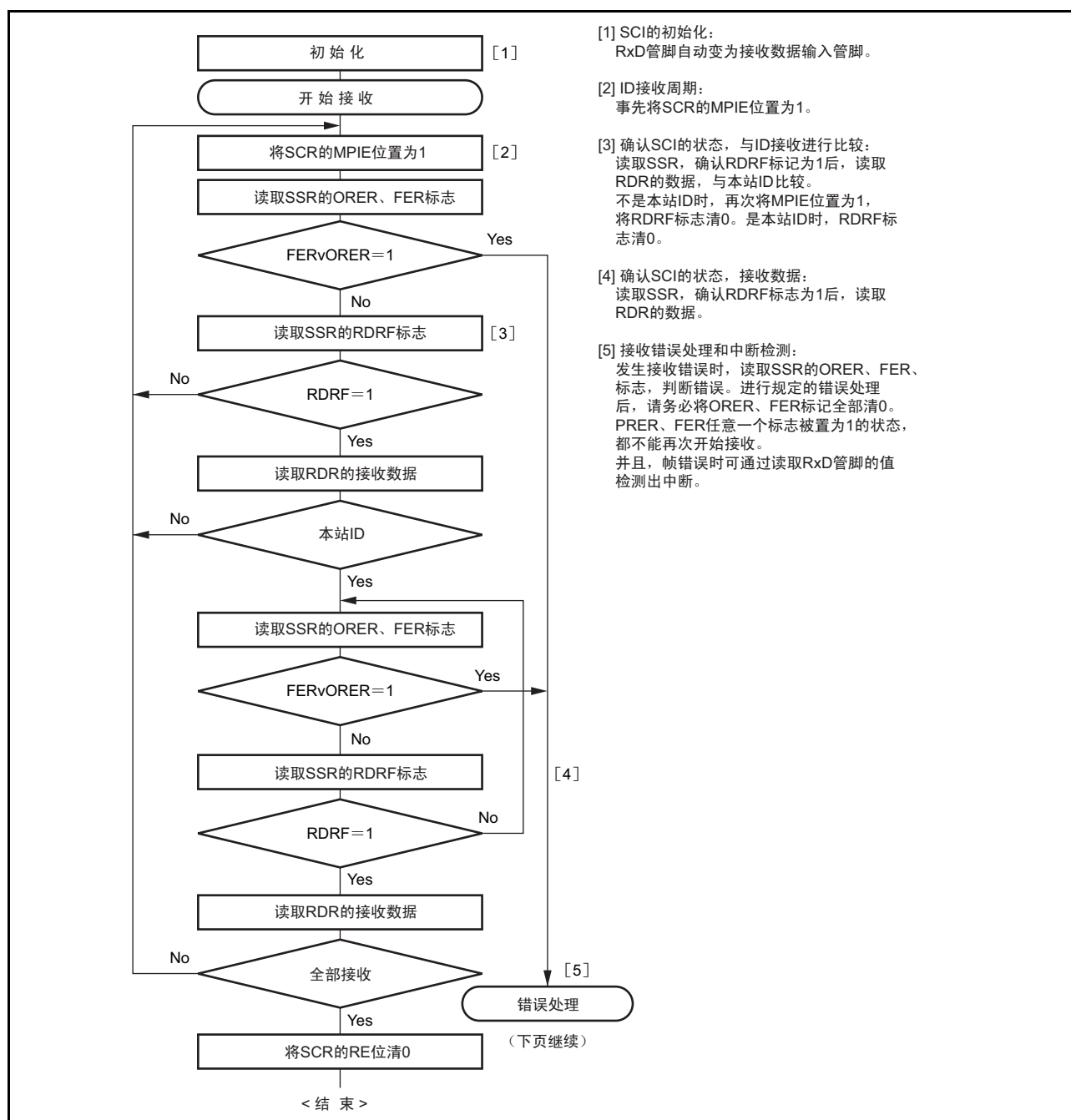


图 10.13 多处理器串行接收的流程图 (1)

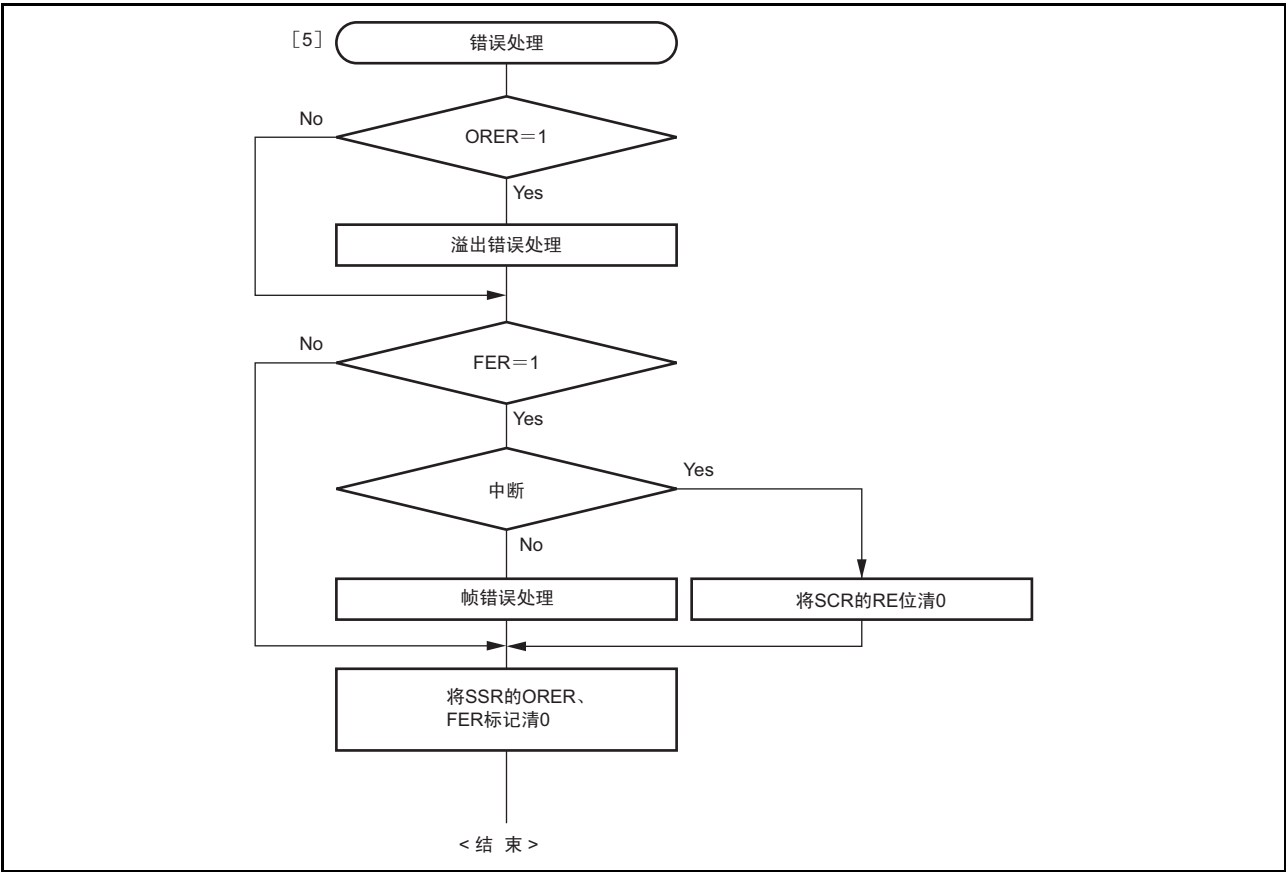


图 10.13 多处理器串行接收的流程图（2）

10.6 时钟同步模式

时钟同步模式的通信数据格式如图 10.14 所示。在时钟同步模式下，发送和接收数据与时钟脉冲同步。在同步时钟串行通信时，从同步时钟的一个下降沿开始到下一个下降沿为止发送数据。接收数据时，与同步时钟的上升沿同步接收数据。通信数据的 1 个字符由 8 位数据构成。传输 8 位数据后通信线路保持 MSB 传输状态。时钟同步模式下，不能附加奇偶位和多处理器位。在 SCI 内部发送和接收单元是相互独立的，可共用时钟进行全双工通信。由于发送 / 接收单元采用的是双缓冲结构，发送中写入下一个发送数据，接收中读取前一个接收数据，来实现连续接收和发送。

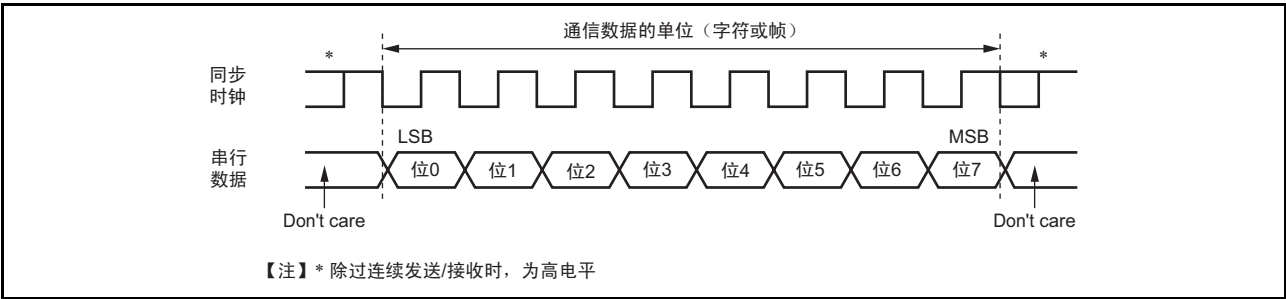


图 10.14 时钟同步模式通信的数据格式（LSB 优先）

10.6.1 时钟

通过设定 SCR 的 CKE1、CKE0，可选择内部波特率发生器产生的内部时钟或从 SCK 管脚输入的外部同步时钟。当内部时钟运行时，可从 SCK 管脚输出持续的时钟。发送和接收 1 个字符，将要输出 8 个持续脉冲，不进行发送 / 接收时，时钟被固定为高电平。

10.6.2 SCI 的初始化

发送 / 接收数据前，将 SCR 的 TE、RE 位清 0 后，按照图 10.15 的流程图的样例对 SCI 进行初始化。务必将 TE 位及 RE 位清零后，再变更运行模式、通信格式等。将 TE 清零后，TDRE 被置为 1，注意，即使 RE 也被清零，RDRF、PER、FER、ORER 的各标志及 RDR 也不会被初始化。

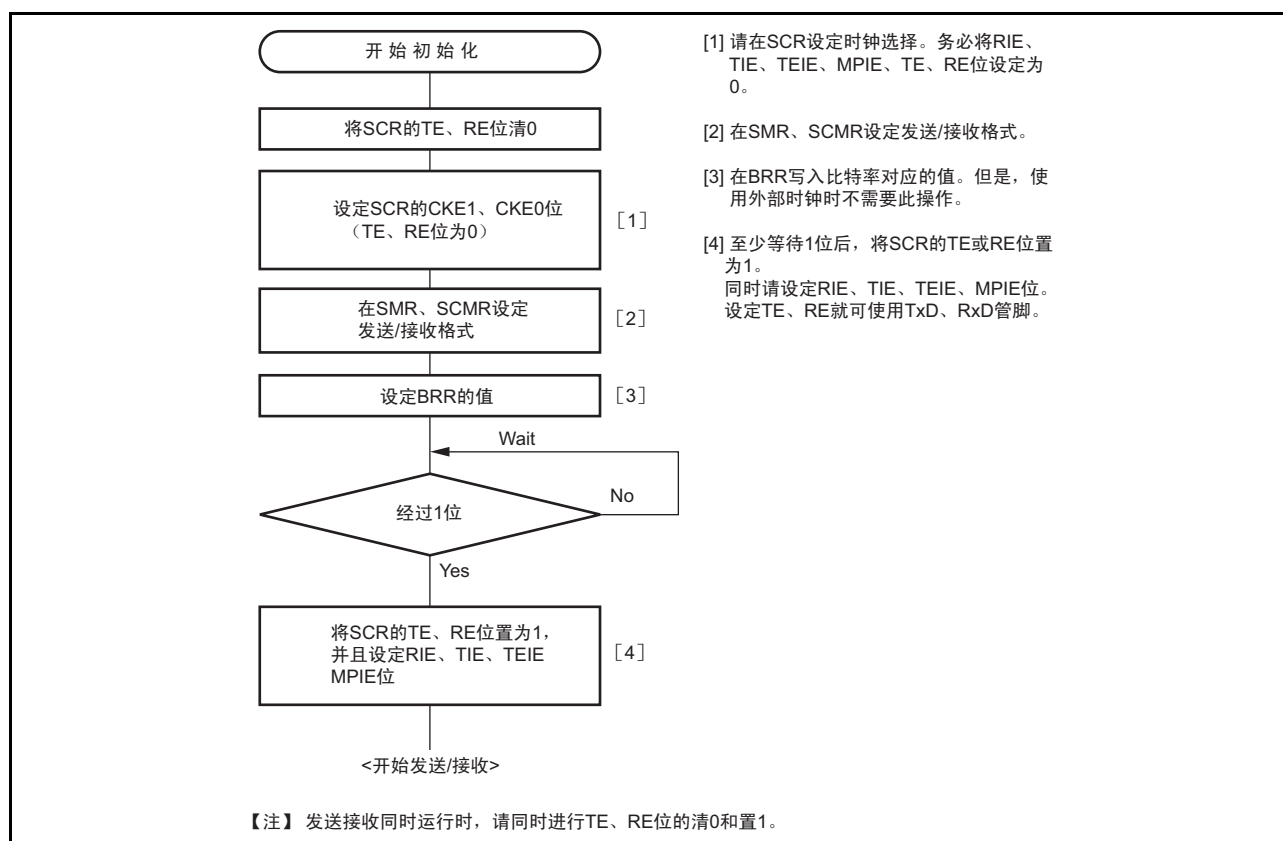


图 10.15 SCI 的初始化流程图

10.6.3 串行数据的发送 (时钟同步模式)

图 10.16 所示的是时钟同步模式下发送的操作。发送时 SCI 运行步骤如下。

1. SCI 监视 SSR 的 TDRE，如果清零，则识别为已经在 TDR 写入数据，将数据从 TDR 传送到 TSR。
2. 数据从 TDR 传送到 TSR 后，将 TDRE 置为 1，开始发送。这时如果 SCR 的 TIE 置为 1，会发生 TXI 中断请求。在 TXI 中断程序中，发送完上一个传送的数据前，给 TDR 写入下一个数据，就可连续发送。
3. 设定为时钟输出模式时，与输出时钟同步，设定为外部时钟时，与输入时钟同步，从 TxD 管脚输出 8 位数据。
4. 发送出最终位的同时检查 TDRE。
5. 如果 TDRE 为 0，将下一个发送数据从 TDR 传送到 TSR，开始发送下一帧。
6. 如果 TDRE 为 1，将 SSR 的 TEND 置 1，保持最终位输出状态。这时 SCR 的 TEIE 置为 1 后，会发生 TEI。SCK 管脚固定为高电平。

图 10.17 所示的是串行数据发送的流程图。在接收错误标记 (ORER、FER、PER) 置为 1 的状态，即使将 TDRE 清零，也不开始发送数据。发送开始前，必须事先将接收错误标记清零。注意：仅清除 RE 位，接收错误标记是会被清除的。

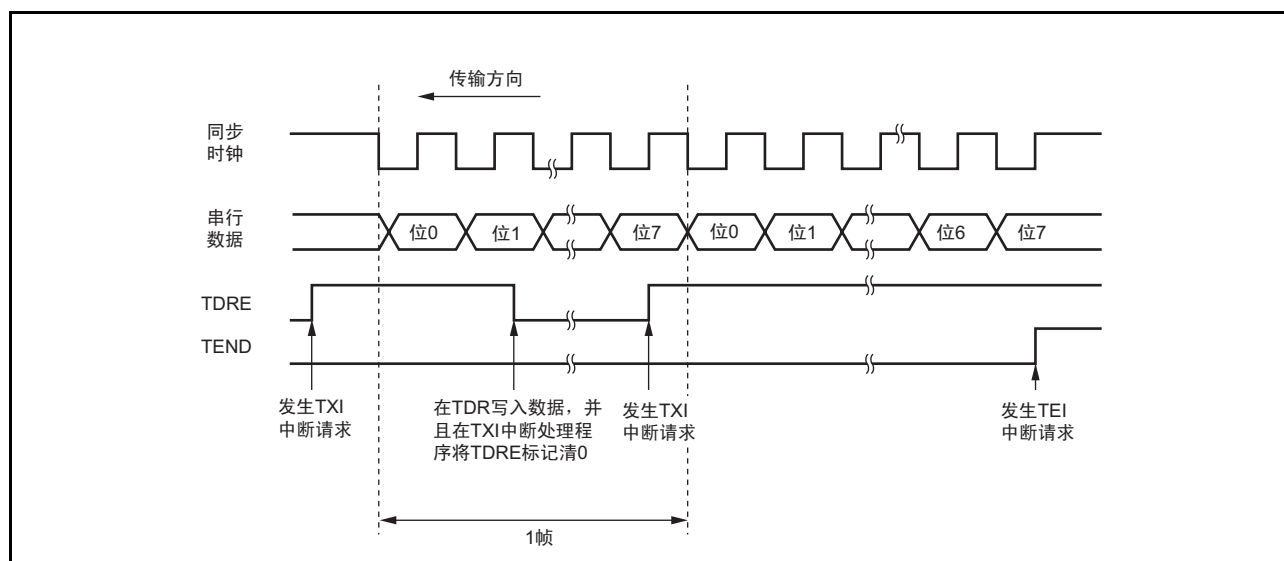


图 10.16 时钟同步模式发送时的运行

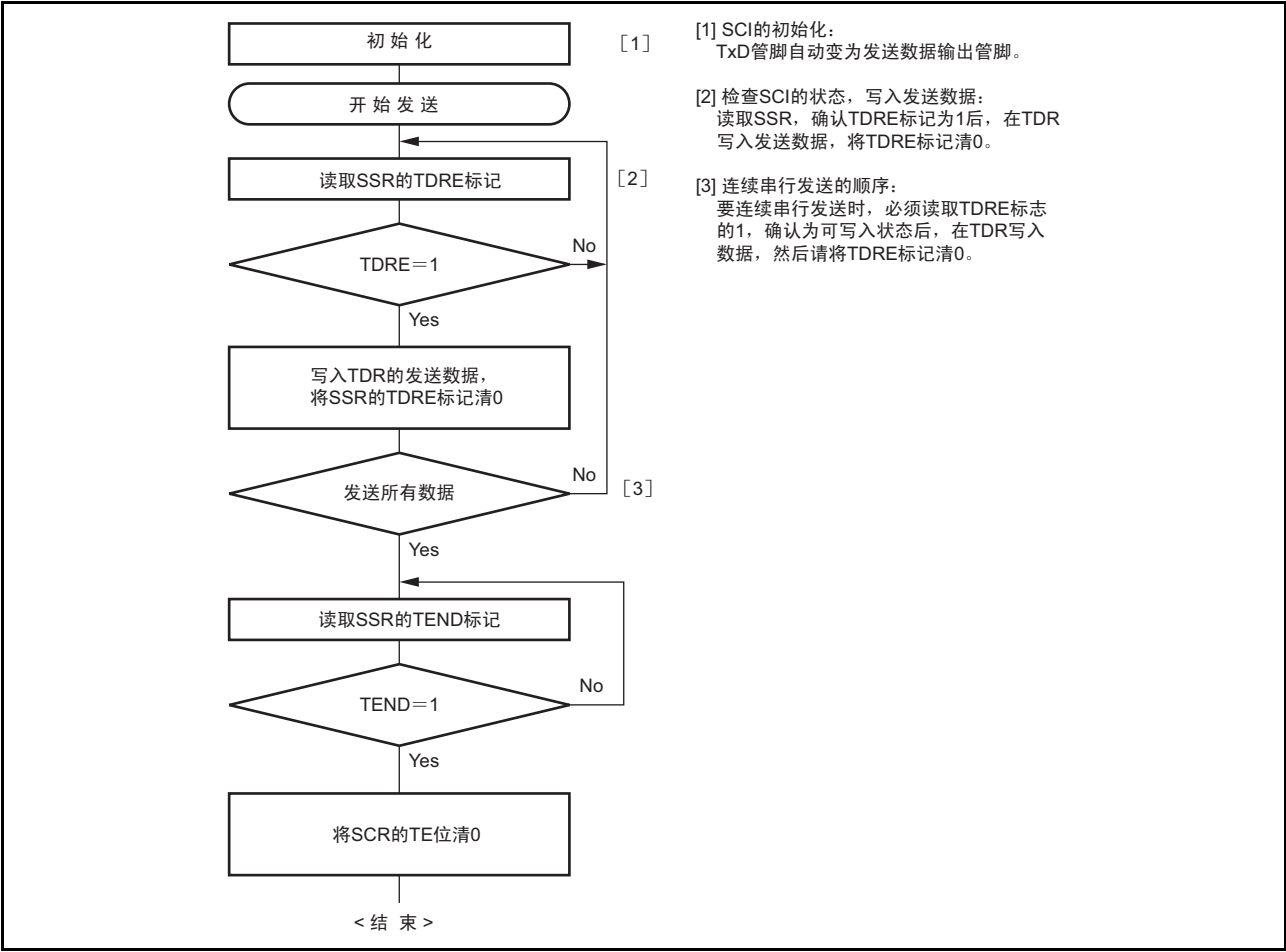


图 10.17 串行发送的流程图

10.6.4 接收串行数据 (时钟同步模式)

图 10.18 所示的是时钟同步模式下接收数据的样例。接收数据时 SCI 运行步骤如下。

1. SCI 与同步时钟的输入或输出同步，将内部初始化后，开始接收，并将接收数据放入 RSR。
2. 发生溢出错误时（在 SSR 的 RDRF 置为 1 的状态，接收到下一个数据时），要设置 SSR 的 ORER。这时如果 SCR 的 RIE 置为 1，会发生 ERI 中断请求。接收数据不传送到 RDR。保持 RDRF 置为 1 的状态。
3. 正常接收时，SSR 的 RDRF 置位，将接收数据传送到 RDR。这时如果 SCR 的 RIE 置为 1，会发生 RXI 中断请求。在 RXI 中断处理程序中，接收下一个数据完成前，读取传送到 RDR 的接收数据，这样就可以连续接收。

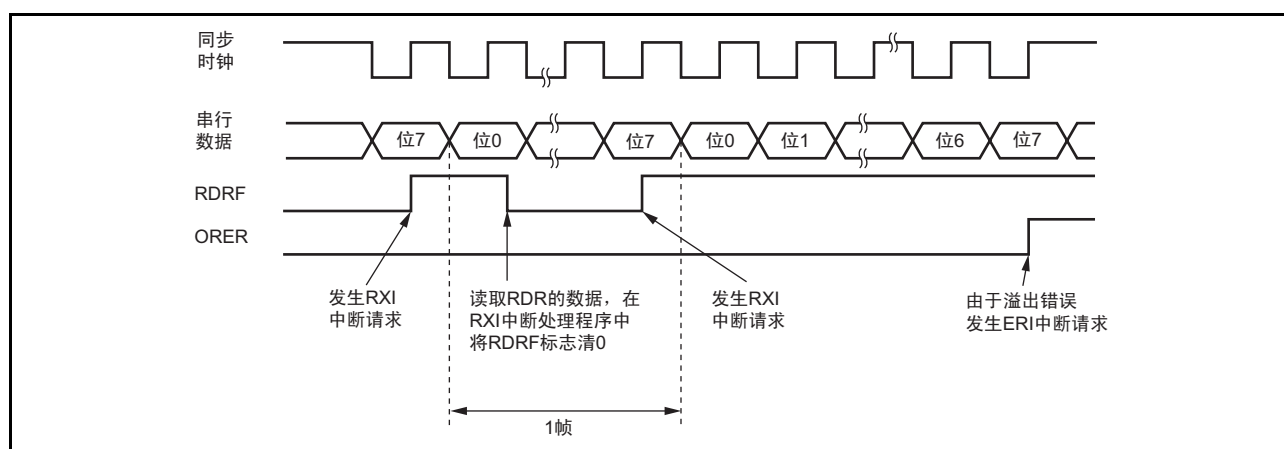


图 10.18 SCI 接收时的运行例子

在接收错误标志被置 1 的状态，不能进行之后的接收动作。因此，在继续接收前请务必将 ORER、FER、PER 及 RDRF 清零。图 10.19 所示的是接收数据的流程图。

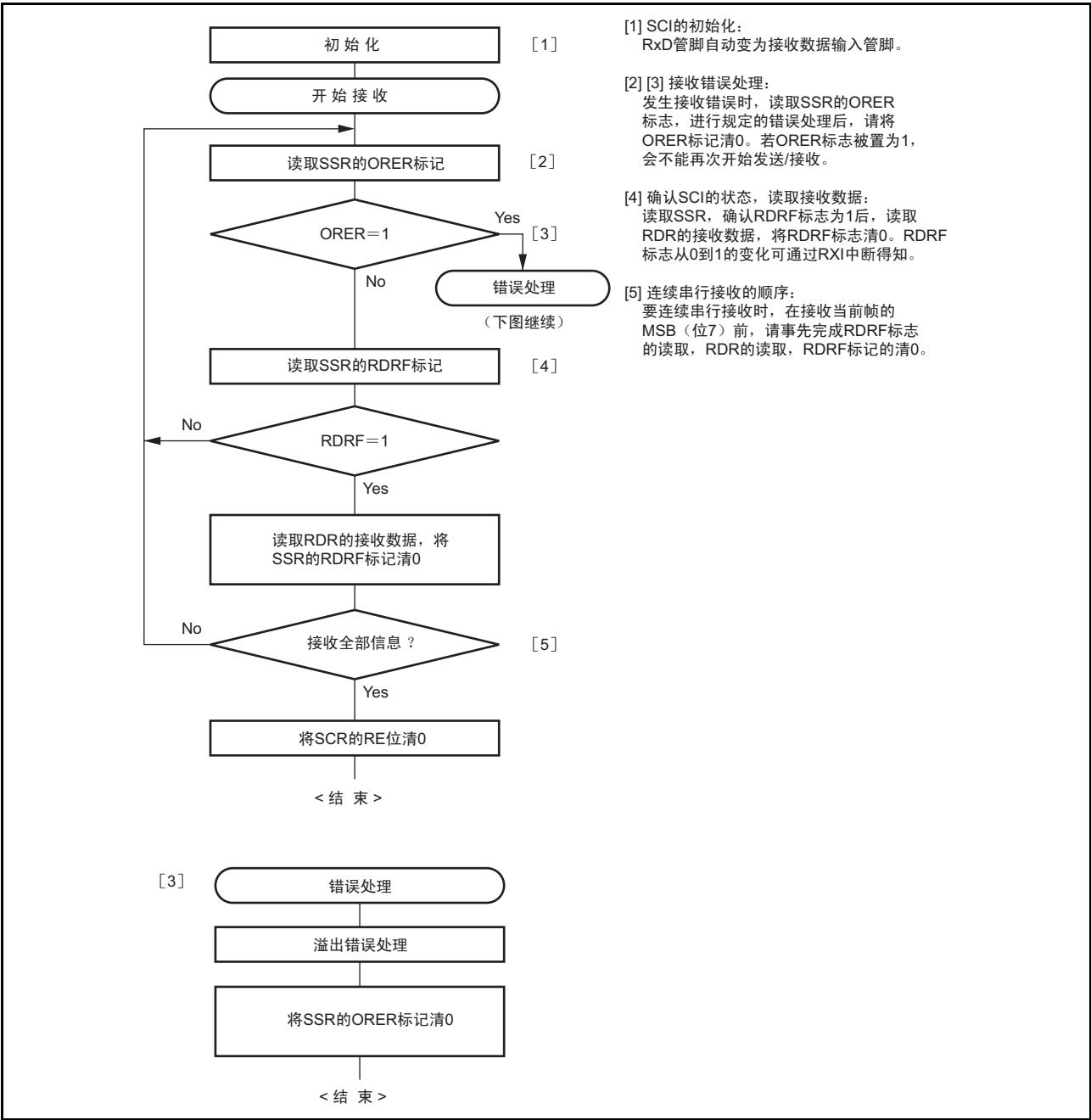


图 10.19 接收串行数据流程图

10.6.5 发送 / 接收串行数据的同步运行 (时钟同步式)

图 10.20 所示的是发送 / 接收同步运行的流程图。在 SCI 初始化后, 按照以下顺序, 同时发送 / 接收数据。从发送切换到同时发送 / 接收时, 确认 SCI 为发送完成状态, TDRE 及 TEND 被置为 1, TE 清零后, 将 TE 及 RE 根据指令同时置为 1。从接收模式切换到同时发送 / 接收模式时, 确认 SCI 为接收完成状态, 将 RE 清零后, 确认 RDRF 及接收错误标记 (ORER、FER、PER) 都被清零后, 根据指令将 TE 及 RE 同时置为 1。

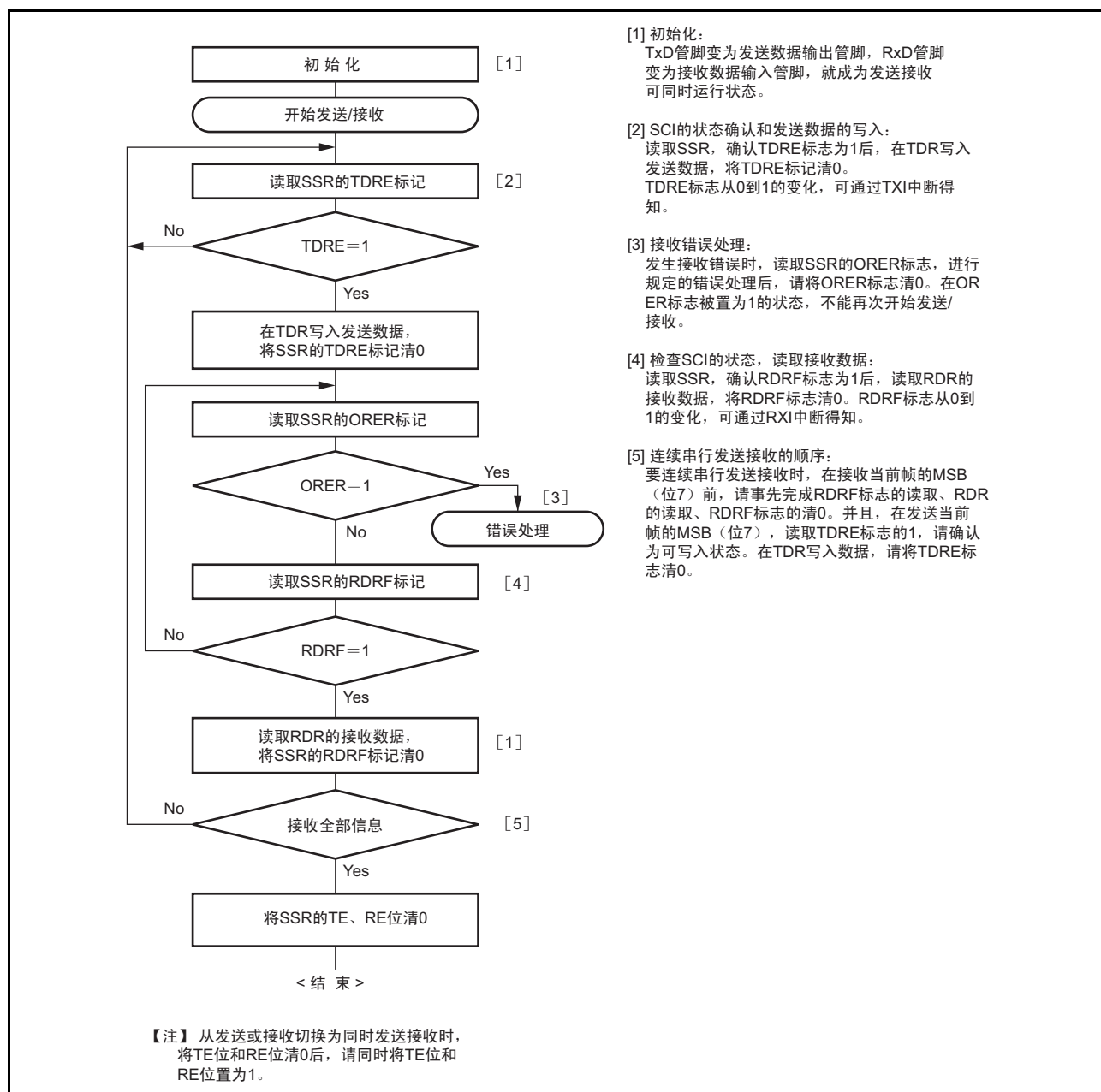


图 10.20 串行发送 / 接收同时运行的流程图

10.7 智能卡接口操作说明

SCI 支持 ISO/IEC 7816-3（Identification Card）标准的 IC 卡（智能卡）接口，并将其作为串行通信接口的扩展功能。可通过设置寄存器的方法进行常规的串行通信接口模式和智能卡接口模式的切换。

10.7.1 管脚连接图例

图 10.21 是与智能卡连接的图例。在 IC 卡通信过程中，用一根数据传输线接收 / 发送数据，连接时将 TxD 管脚和 Rx D 管脚短接，在数据传输线上用电阻向电源 VCC 方向上拉。如果在没有连接 IC 卡的状态，设定 RE=TE=1，就变为封闭的发送 / 接收，可进行自我检测。要将 SCI 产生的时钟提供给 IC 卡时，请把 SCK 管脚的输出连接到 IC 卡的 CLK 管脚。IC 卡的复位信号可与本 LSI 的输出端口连接。

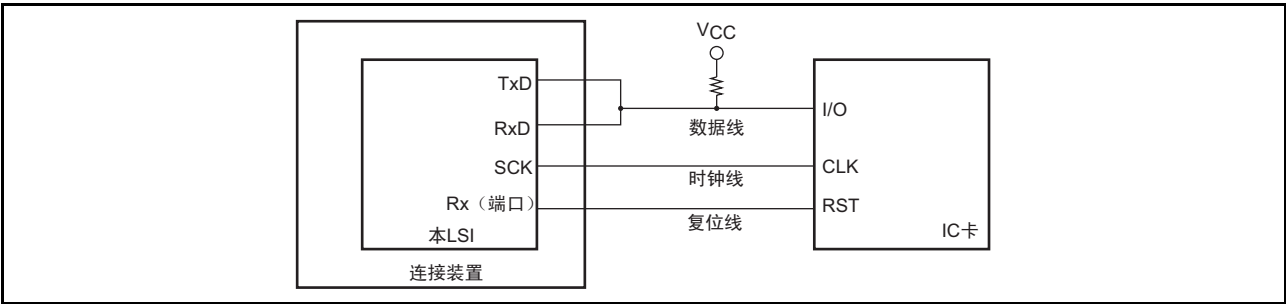


图 10.21 智能卡接口管脚连接概要

10.7.2 数据格式（除数据块传送模式）

图 10.22 是智能卡接口模式下的发送 / 接收格式。

- 异步模式下，1 帧数据由 8 位数据和奇偶校验位构成。
- 发送时，从奇偶校验位结束到下一帧开始，要留 2etu（Elementary Time Unit: 传送 1 位）以上的保护时间。
- 接收时，检测出奇偶校验错误，从起始位经过 10.5etu 后，错误信号的低电平将在 1etu 时间输出。
- 发送时，如果采样到错误信号，经过 2etu 以上后，会自动再次发送相同数据。

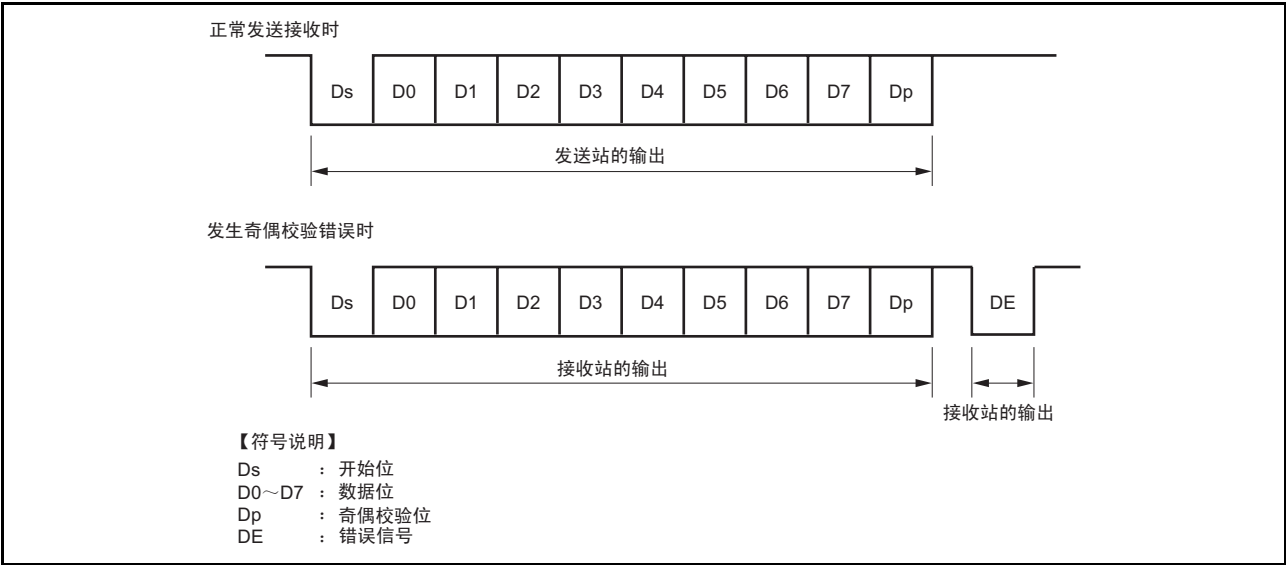


图 10.22 通常智能卡接口模式的数据格式

如下与直接转换及反码转换 2 种类型的 IC 卡进行发送 / 接收。

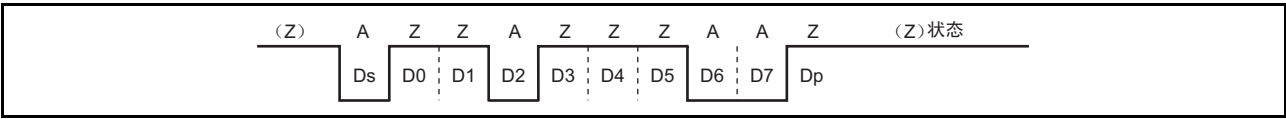


图 10.23 直接转换 (SDIR=SINV=O/E=0)

如下图进行字符直接转换的样例，直接转换类型逻辑 1 与状态 Z 对应，逻辑 0 与状态 A 对应，以 LSB 格式优先发送 / 接收。以上进行字符转换的过程中，数据为 H'3B。在直接转换类型中，请将 SCMR 的 SDIR 和 SINV 位置为 0。根据智能卡规范，要变为偶数校验请将 SMR 的 O/E 位置为 0。

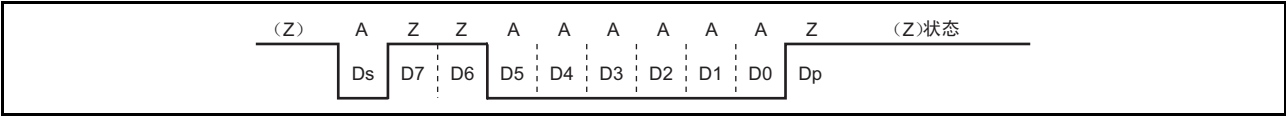


图 10.24 反码转换 (SDIR=SINV=O/E=1)

反码转换类型，逻辑 1 与状态 A 对应，逻辑 0 与状态 Z 对应，以 MSB 格式优先发送 / 接收。以上进行字符转换的过程中，数据为 H'3F。反码转换类型中，请将 SCMR 的 SDIR 位，SINV 位均置为 1。根据智能卡规范，奇偶校验位若是偶校验则为逻辑 0，与状态 Z 对应。在本 LSI 中，SINV 位只能反转数据位 D0 ~ D7，因此，发送 / 接收都要将 SMR 的 O/E 位置为 1，才可反转奇偶校验位。

10.7.3 数据块传送模式

数据块传送模式与通常的 SCI 异步模式相比较大致相同，但数据块传送模式有以下几点不同。

- 接收时，虽然进行奇偶校验，即使检查到错误也不会输出错误信号。由于 SSR 的 PER 被置 1，因此在接收下一帧的奇偶校验位之前请将其清除。
- 发送时，从奇偶校验位结束到下一帧开始的保护时间至少为 1etu。
- 发送时，不执行再次发送，所以从发送开始后 11.5etu 的时间，设置 TEND 标记为 1。
- 作为通常的智能卡接口，ERS 标记表示错误信号的状态，但由于不发送和接收错误信号，所以通常为 0。

10.7.4 采样接收数据时序和智能卡接口模式的接收容限

在智能卡接口模式，可使用的发送接收时钟只有内置波特率生成器生成的内部时钟。在智能卡接口模式下，通过设定 BCP1、BCP0，SCI 以频率为位速率的 32 倍、64 倍、372 倍、256 倍（通常异步模式中固定为 16 倍）的基本时钟运行。接收时，用基本时钟采样起始位的下降沿，并使内部同步。如图 10.25 所示，分别在基本时钟的第 16、32、186、128 个上升沿进行采样，在各位的中间放入数据。这时的接收容限可用下面的公式表示。

$$M = \left| \left(0.5 - \frac{1}{2N} \right) - (L - 0.5) F - \frac{|D - 0.5|}{N} (1 + F) \right| \times 100\%$$

M: 接收容限 (%)

N: 时钟对应的比特率比 (N=32、64、372、256)

D: 时钟占空比 (D=0~1.0)

L: 帧长度 (L=10)

F: 时钟频率偏差的绝对值

以上的式子中，如果 F=0、D=0.5、N=372，接收容限变为以下所示。

$$\begin{aligned} M &= (0.5 - 1/2 \times 372) \times 100\% \\ &= 49.866\% \end{aligned}$$

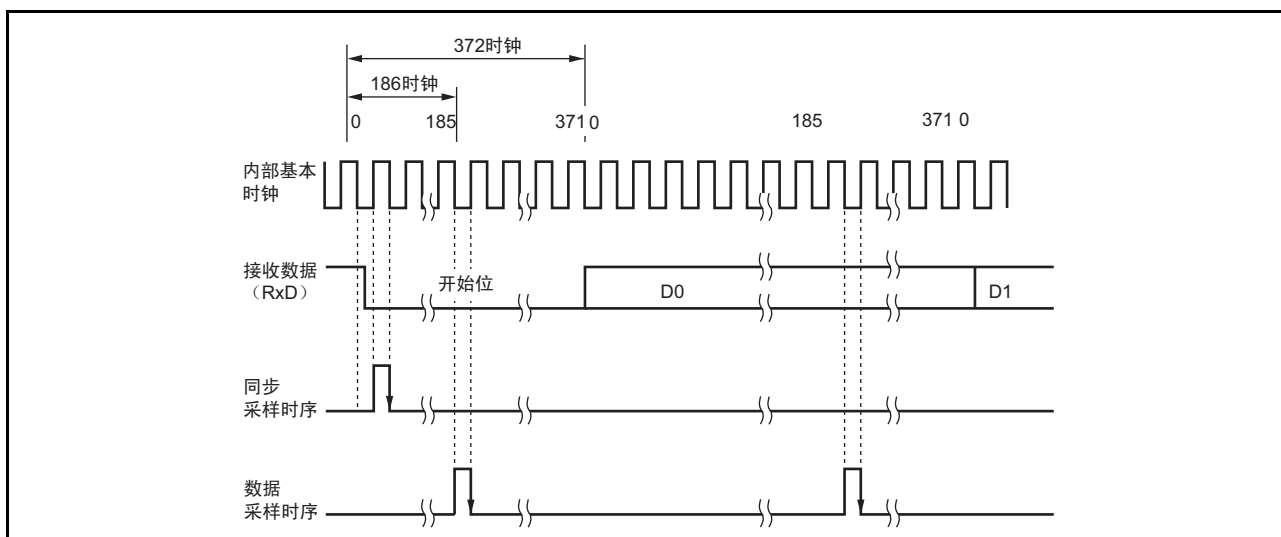


图 10.25 智能卡接口模式时的采样接收数据的时序（使用 372 倍的时钟时）

10.7.5 初始设定

发送 / 接收数据前，按以下顺序初始化 SCI。无论从发送模式转换为接收模式，还是从接收模式转换为发送模式，都需要初始化。

1. 将 SCR 的 TE、RE 位清零。
2. 将 SSR 的错误标记 ERS、PER、ORER 清零。
3. 设置 SMR 的 GM、BLK、BCP1、BCP0、CKS1、CKS0 位。此时，请将 PE 位置为 1。
4. 设置 SCMR 的 SMIF、SDIR、SINV 位。
5. 将 SMIF 位置为 1 后，TxD 管脚及 RxD 管脚同时从端口切换到 SCI 管脚，从而置为高阻抗状态。
6. 在 BRR 中设置与位速率相对应的值。
7. 设定 SCR 的 CKE1、CKE0 位。将 TIE、RIE、TE、RE、MPIE、TEIE 位均置为 0。
8. 如果将 CKE0 位置为 1，将从 SCK 管脚输出时钟。
9. 至少等候 1 位的间隔时间，然后设定 SCR 的 TIE、RIE、TE、RE 位。除过自我检查外，请不要同时设置 TE 位和 RE 位。

从接收模式向发送模式切换时，确认 SCI 接收已经完成，从初始化开始，设定 RE=0、TE=1。可根据 RDRF 标记、PER 或 ORER 标记确认 SCI 接收是否完成。从发送模式向接收模式切换时，确认发送完成，从初始化开始，设定 TE=0、RE=1。可根据 TEND 标记确认 SCI 发送是否完成。

10.7.6 发送数据（除数据块传送模式之外）

智能卡模式下的数据发送具有错误信号的采样和再发送处理，因此与通常串行通信接口的运行不同（除数据块传送模式之外）。发送数据的再传送机制如图 10.26 所示。

1. 发送 1 帧数据完成后，如果采样到接收方的错误信号，SSR 的 ERS 位会被置为 1。这时如果 SCR 的 RIE 位被置为 1，会产生 ERI 中断请求。在采样下一个奇偶校验位前，请将 ERS 清零。
2. 在接收到错误信号的帧中，SSR 的 TEND 不被置位。会再次将数据从 TDR 传送到 TSR，执行自动再次发送。
3. 如果没有从接收方返回错误信号时，表明 SSR 的 ERS 位未被置位，判断一帧数据发送完成（包括再次传送数据），则 SSR 的 TEND 会被置 1，这时如果 SCR 的 TIE 被置位，就产生 TXI 中断请求，此时将发送数据写到 TDR，下一个数据就会被发送。

发送处理流程的样例如图 10.28 所示。在发送过程中，SSR 的 TEND 标志被置为 1，同时 TDRE 标记也被置位，如果事先设定 SCR 的 TIE = 1，就会发生 TXI 中断请求。当发生错误时，SCI 会自动再次发送相同数据，在此期间 TEND 保持为 0。因此，对于发生错误时的再次发送，SCI 和 DTC 会自动发送指定的字节数。但是发生错误时，ERS 标记不会被自动清除，所以要事先将 RIE 位置为 1，发生错误时，ERI 发生中断请求，同时 ERS 标记被清除。

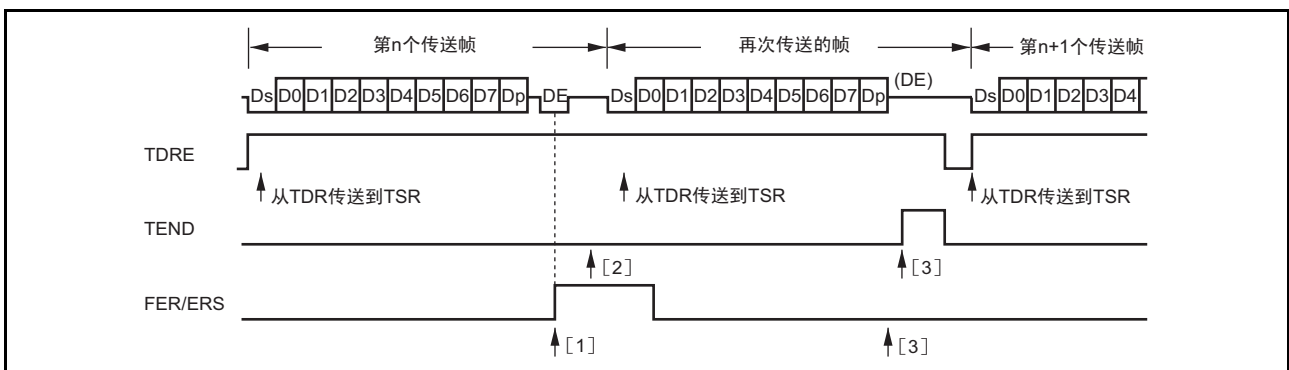


图 10.26 SCI 发送模式时的再传送动作

SMR 的 GM 位设定不同，TEND 标记的设定时序就会有所不同。图 10.27 所示的是 TEND 标记发生时序。

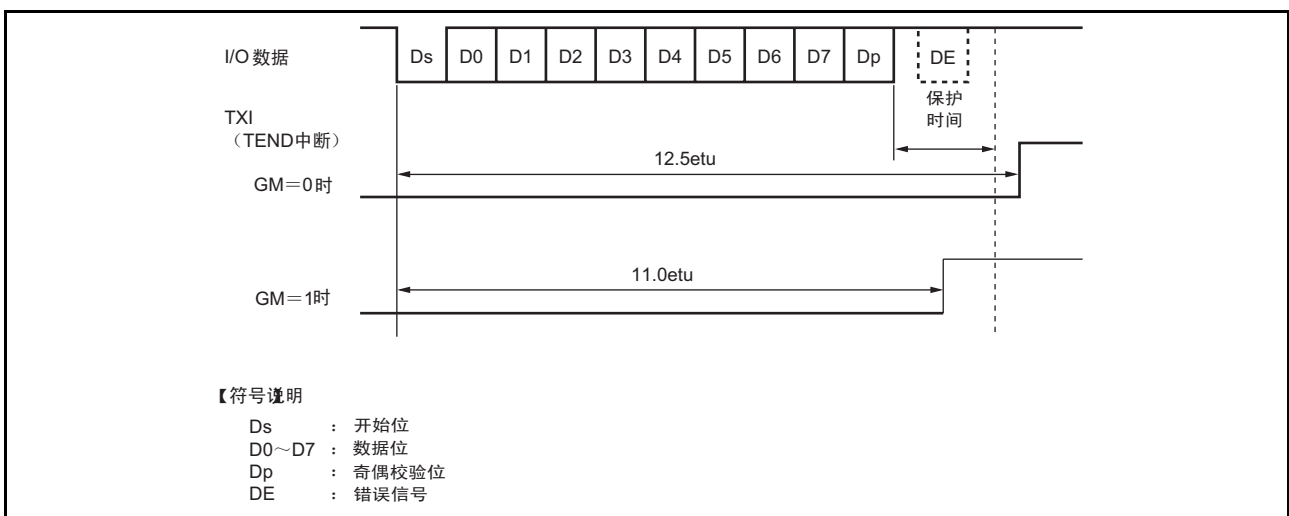


图 10.27 发送运行时的 TEND 标记发生时序

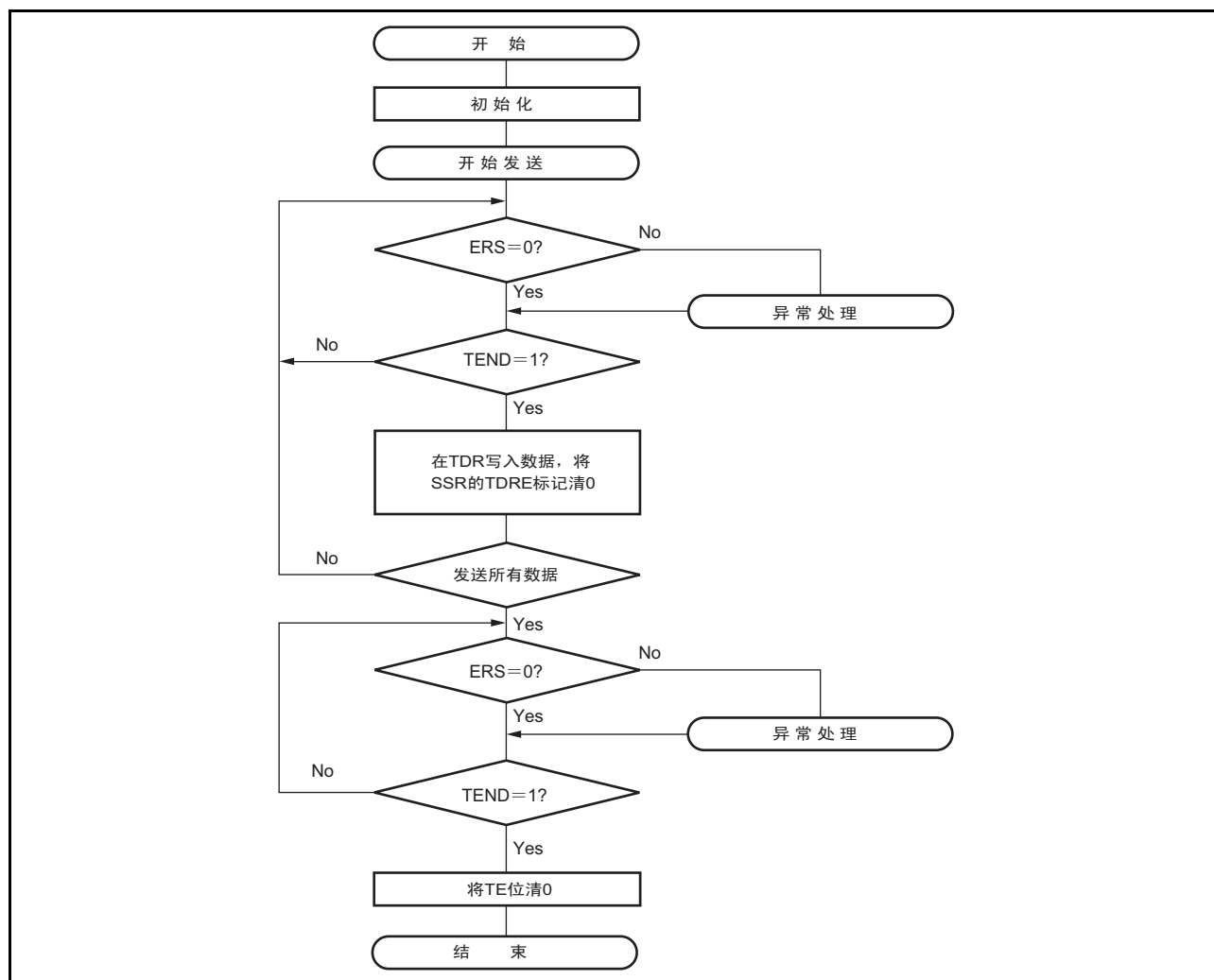


图 10.28 发送处理流程

10.7.7 串行数据的接收（除过数据块传送模式）

智能卡接口模式下的数据接收处理顺序与通常串行通信接口模式的相同。SCI 的接收模式的再次传送如图 10.29 所示。

- 1. 接收数据时，如果检测出奇偶校验错误，SSR 的 PER 位自动置为 1。这时，如果 SCR 的 RIE 被置位，就会发生 ERI 中断请求。在采样到下一个奇偶校验位前，请保持 PER 位清零状态。
- 2. 在检测出奇偶校验错误的帧中，SSR 的 RDRF 位不被置位。
- 3. 没有检测出奇偶校验错误时，SSR 的 PER 位未被置为 1。判断其正常完成接收后，SSR 的 RDRF 自动置位为 1。这时如果 SCR 的 RIE 位被置位，则会发生 RXI 中断请求。

接收数据流程的例子如图 10.30 所示。在接收过程中，将 RDRF 标志置为 1 时会发生 RXI 中断请求。而在接收时发生错误，ORER 或 PER 标记被置为 1，就会发生接收 / 发送错误中断（ERI）请求，所以请及时将错误标记清零。接收时发生奇偶校验错误，即使 PER 被置为 1，接收的数据也会传送到 RDR，也能够读取数据。

【注】 块传送模式的情况请参照「10.4 异步模式的运行」。

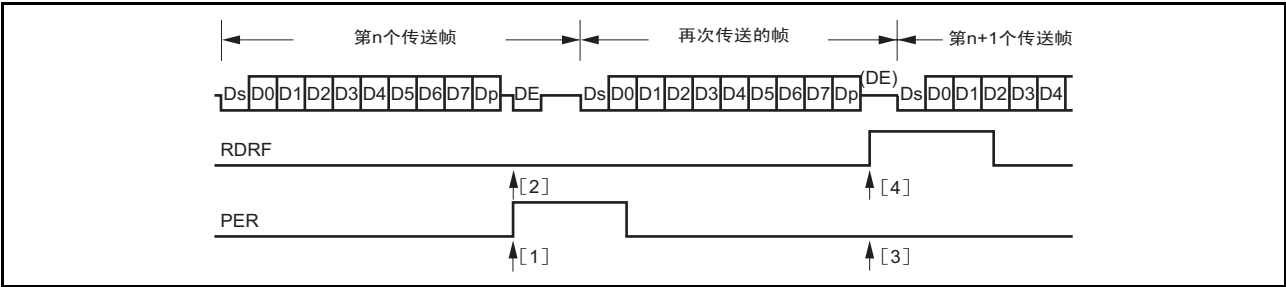


图 10.29 SCI 接收模式时的再次传送操作

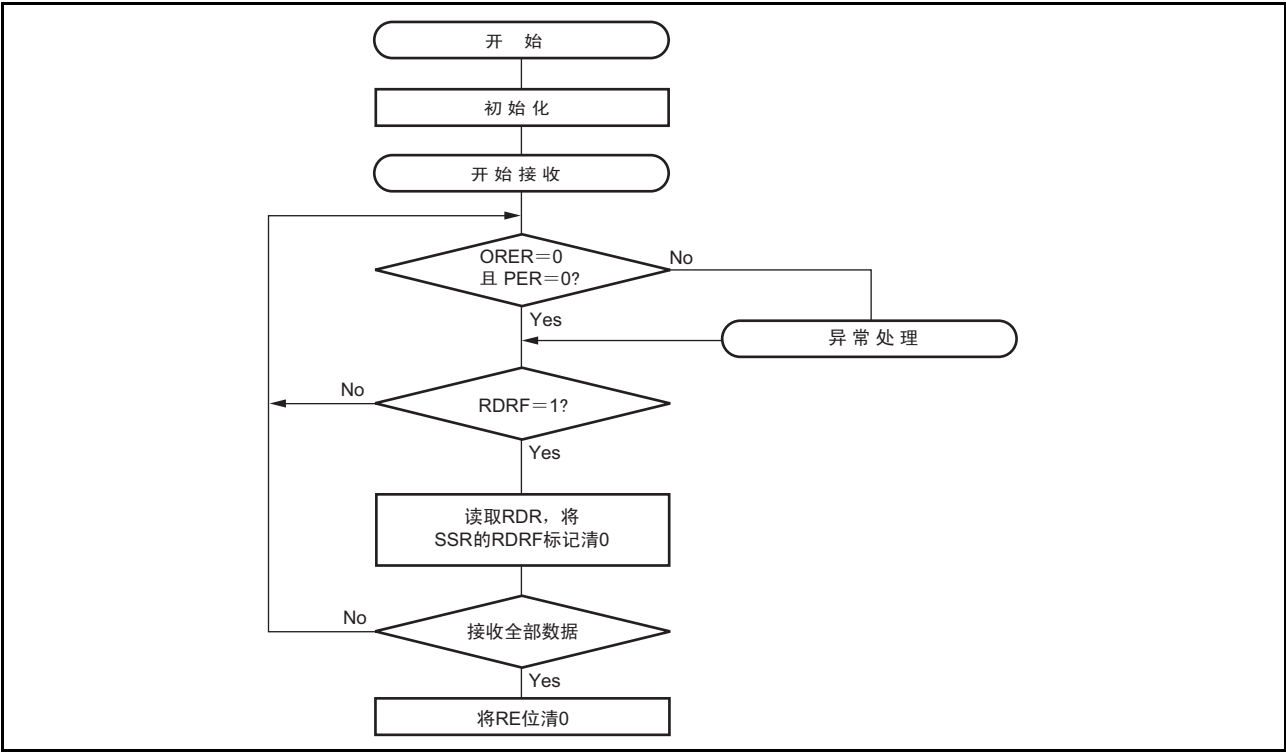


图 10.30 接收流程的例子

10.7.8 时钟输出控制

当 SMR 的 GM 位被置为 1 时, 根据 SCR 的 CKE1、CKE0 位可固定时钟输出。此时可将时钟脉冲的最小宽度定为指定宽度。

图 10.31 所示的是时钟输出的固定时序。在这个样例中 GM=1、CKE1=0 时, 控制 CKE0 位。

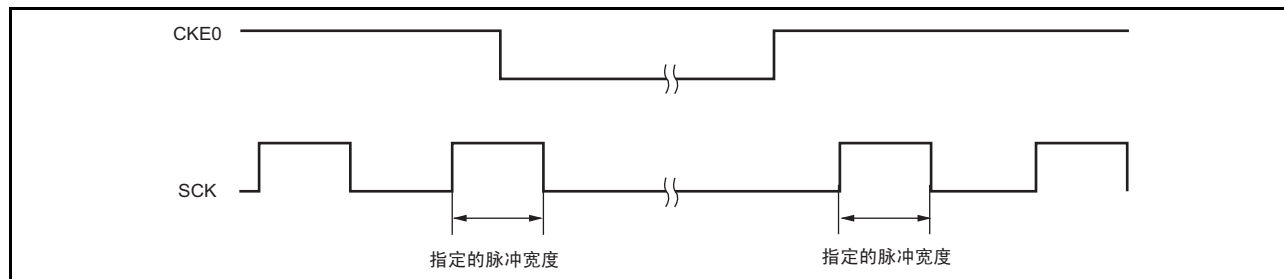


图 10.31 时钟输出固定时序

当加载电源时, 向软件待机模式迁移或者从软件待机模式恢复时, 为了确保时钟的占空比, 请按以下顺序处理。

- 加载电源时

从加载电源时就要确保时钟占空比, 按照以下的切换顺序进行处理。

 1. 初始状态时输入端口保持高阻抗状态。使用上拉电阻 / 下拉电阻来调节电位。
 2. 用 SCR 的 CKE1 位将 SCK 管脚固定为指定的输出。
 3. 设置 SMR 和 SCMR, 切换为智能卡模式。
 4. 将 SCR 的 CKE0 位设定为 1, 开始时钟输出。
- 从智能卡接口模式迁移到软件待机模式时
 1. 将 SCK 管脚对应的数据寄存器 (DR) 和数据方向寄存器 (DDR) 设定为软件待机模式的输出固定状态值。
 2. 在 SCR (串行控制寄存器) 的 TE 位和 RE 位写入 0, 使发送 / 接收停止。
 3. 同时也将 CKE1 位设定为软件待机时的输出状态值。
 4. 在 SCR 的 CKE0 位写入 0, 使时钟输出停止。
 5. 等待一个串行时钟周期。
 6. 在此期间, 保持占空比, 用指定电平固定时钟输出。
 7. 转换到软件待机状态。
- 从软件待机模式返回到智能卡接口模式时
 1. 退出软件待机状态。
 2. 在 SCR 的 CKE0 位写入 1, 输出时钟。根据正常的占空比开始发生信号。

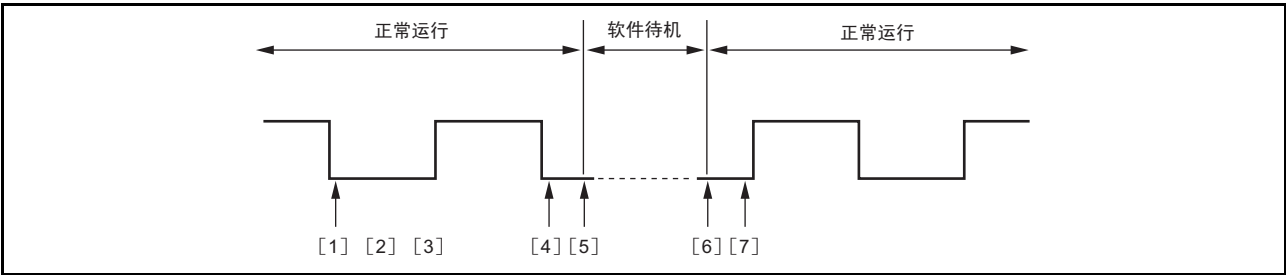


图 10.32 时钟停止 . 复位顺序

10.8 中断源

10.8.1 普通串行通信接口模式的中断

表 10.12 所示的是普通串行通信接口模式的中断源。给各中断源分配了不同的中断向量，可通过 SCR 的允许位进行独立允许操作。

当 SSR 的 TDRE 标记被置为 1，会发生 TXI 中断请求。当 SSR 的 TEND 标记置为 1，会发生 TEI 中断请求。

当 SSR 的 RDRF 标记置为 1，会发生 RXI 中断请求。当 SSR 的 ORER、PER、FER 标记任意一个被置为 1，均会发生 ERI 中断请求。

在 TEIE 位置为 1 的状态，TEND 标记被置为 1 时将发生 TEI 中断。同时发生 TEI 中断和 TXI 中断的状态，先接受 TXI 中断。在 TXI 中断程序中，如果要同时清除 TDRE 标记和 TEND 标记，请注意不能转移至 TEI 中断程序了。

表 10.12 SCI 中断源

通道	名称	中断源	中断标记
0	ERI0	接收错误	ORER、FER、PER
	RXI0	接收数据满	RDRF
	TXI0	发送数据空	TDRE
	TEI0	发送结束	TEND
1	ERI1	接收错误	ORER、FER、PER
	RXI1	接收数据满	RDRF
	TXI1	发送数据空	TDRE
	TEI1	发送结束	TEND

10.8.2 智能卡接口模式下的中断

在智能卡接口模式，有如表 10.13 中的中断源。在此模式下不能使用发送完成中断（TEI）请求。

表 10.13 SCI 中断源

通道	名称	中断源	中断标记
0	ERI0	检测出接收信息错误	ORER、PER、ERS
	RXI0	接收信息 full	RDRF
	TXI0	发送数据 empty	TEND
1	ERI1	检测出接收信息错误	ORER、PER、ERS
	RXI1	接收信息 full	RDRF
	TXI1	发送数据 empty	TEND

发送运行中，如果 SSR 的 TEND 标记被置为 1，同时 TDRE 标记也被置位，就会发生 TXI 中断。发生错误时，SCI 会自动再次发送相同数据。在此期间 TEND 保持为 0。发生错误时，ERS 标记不会被自动清除，因此要事先将 RIE 位置为 1，在发生错误时使其发生 ERI 请求，请清除 ERS 标记。

接收中，如果 SSR 的 RDRF 标记被置为 1，发生 RXI 中断请求。发生错误时，RDRF 标记不置位，而对错误标记置位，一个 ERI 中断请求将会发送给 CPU，因此，务必删除该错误标记。

10.9 使用时的注意事项

10.9.1 模块待机模式的设定

根据模块停止控制寄存器，可设定 SCI 运行的禁止 / 允许。在初始值状态下 SCI 的运行被禁止。解除模块停止模式，就可以存取寄存器。详细请参考「第 20 章 低功耗状态」。

10.9.2 关于断点的检测和处理

在检测帧错误时，通过直接读取 RxD 管脚值就可检测出中止状态。中止状态期间，RxD 管脚的输入全部为 0，因此 FER 和 PER 都被置位。SCI 接收到中止状态信号后，还可以继续接收。因此，请注意，即使将 FER 清零，也会被再次置为 1。

10.9.3 标志状态和中断的发送

TE 为 0 时，TxD 管脚作为通过 DR 和 DDR 决定输入输出方向和电平的 I/O 端口。使用它将 TxD 管脚置为标记状态或在发送数据时送出中止信号。为了在 TE 置为 1 前，使通信线路保持标记状态（1 的状态），设定 DDR=1、DR=1。这时，TE 被清零，TxD 管脚变为 I/O 端口，并输出 1。另一方面，发送数据时，如果希望发送中止信号时，设定 DDR=1、DR=0 后，将 TE 清零。一旦 TE 清零，发送功能被初始化与当前发送状态无关，TxD 管脚变为 I/O 端口，从 TxD 管脚输出 0。

10.9.4 接收错误标志和发送运行（仅对时钟同步模式有效）

在接收错误标志（ORER、PER、FER）被置为 1 的状态，即使将 TDRE 清零，也不能开始发送。必须在发送开始时事先将接收错误标志清零。另外，请注意，即使将 RE 清零，也不能将接收错误标志清零。

10.9.5 模式迁移操作

(1) 发送

停止运行后 (TE=TIE=TEIE=0) 才能向模块停止模式、软件待机模式、看门狗模式、子激活模式或子睡眠模式迁移。并且 TSR、TDR 及 SSR 被复位。依据端口的设定决定模块处于待机模式、软件待机模式、监视模式、子激活模式或子睡眠模式期间的输出管脚状态, 模式解除后再次设定 TE=1, 就变为高电平输出。如果正在发送数据时发生迁移, 正在发送的数据将会变的不确定。

模式解除后, 不改变发送模式发送时, 设定 TE=1, 按照 “SSR 读取 → TDR 写入 → TDRE 清 0” 的顺序, 开始发送。改变发送模式发送时, 首先要对 SCI 进行初始化。图 10.33 中所示的是发送模式迁移流程图。图 10.34、图 10.35 所示的是发送时的管脚状态。

通过 DTC 传送, 从发送模式向模块待机模式、软件待机模式、监视模式、子激活模式或子睡眠模式迁移时, 请在停止运行后 (TE=TIE=TEIE=0) 执行。模式解除后, 如果设定 TE=1、TIE=1, 会发生 TXI 中断请求, 并通过 DTC 开始发送。

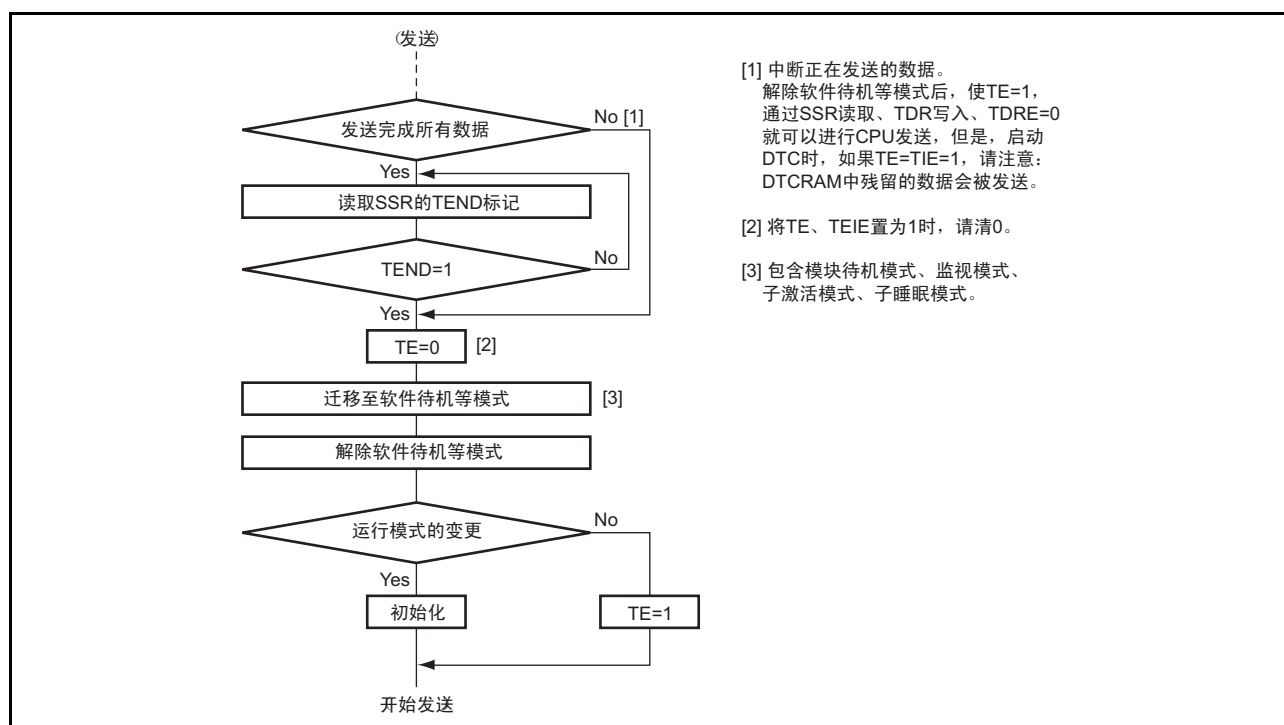


图 10.33 发送时的模式迁移的流程图

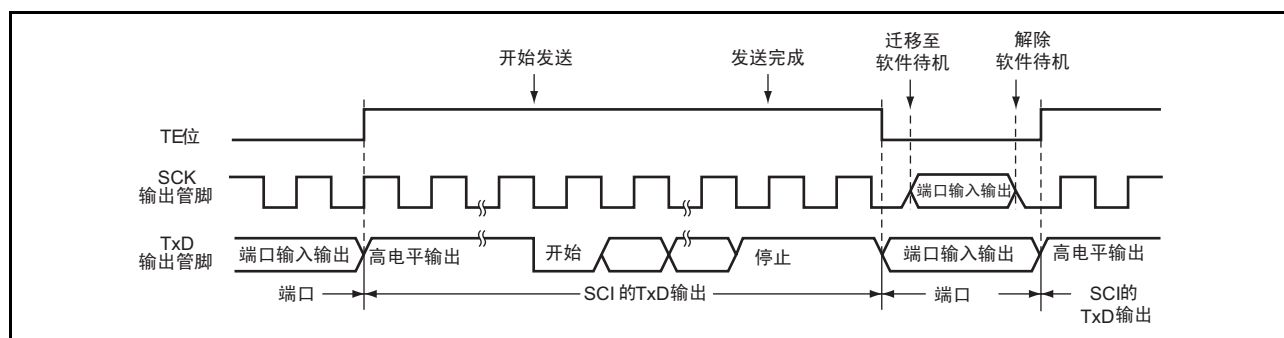


图 10.34 异步模式发送时 (内部时钟) 的管脚状态

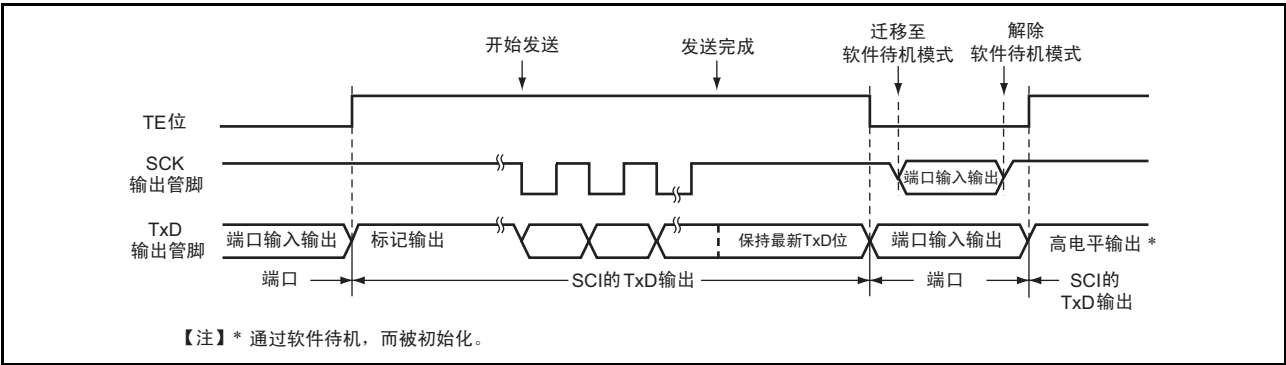


图 10.35 时钟同步模式发送时（内部时钟）的管脚状态

(2) 接收

在接收运行停止（RE=0）后才能向模块待机模式、软件待机模式、监视模式、子激活模式或子睡眠模式迁移，并且 RSR、RDR 及 SSR 被复位。如果正在接收数据时发生迁移，正在接收的数据将是无效的。

模式解除后，不改变接收模式接收数据时，设定 RE=1 后，开始接收。改变接收模式接收数据时，首先要对 SCI 进行初始化。

图 10.36 所示的是接收信息时模式迁移流程。

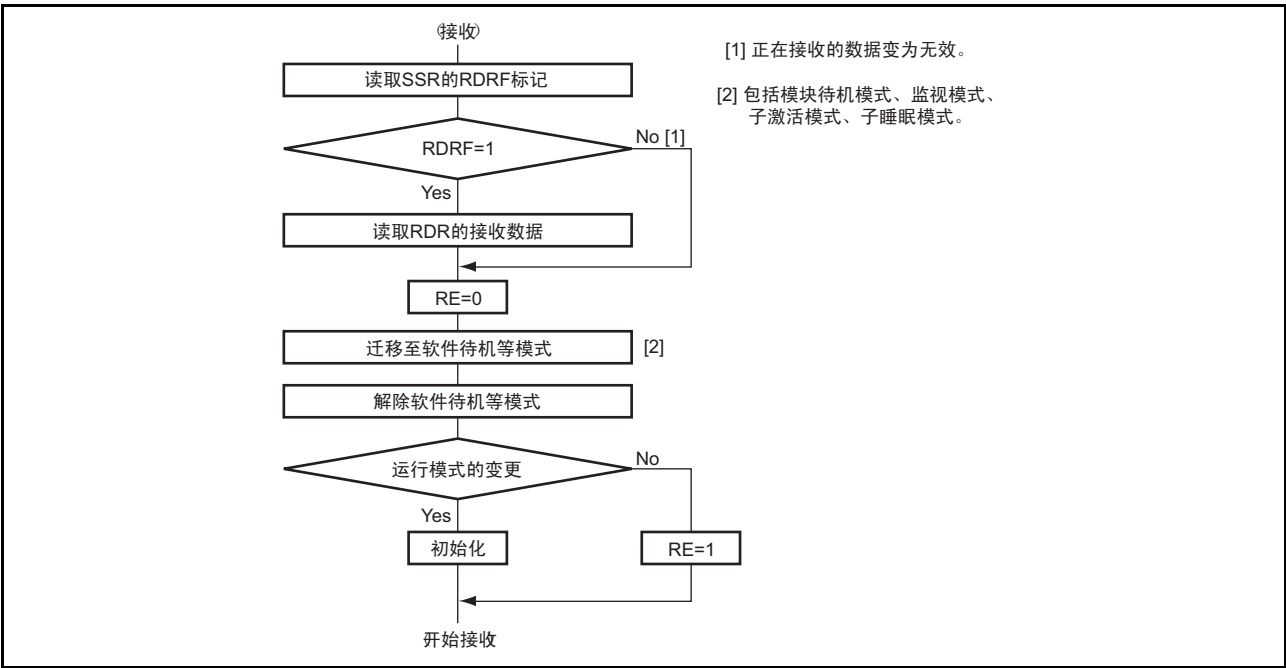


图 10.36 接收时的模式迁移流程图

10.9.6 从 SCK 管脚向端口管脚切换时的注意事项

(a) 运行现象

设定 $DDR=1$ 、 $DR=1$ ，在同步时钟模式下使用 SCI 时钟输出，在发送完成状态将 SCK 管脚切换到端口管脚时，经过半个周期的低电平输出后，端口输出被激活。

在 $DDR=1$ 、 $DR=1$ 、 $C/\bar{A}=1$ 、 $CKE1=0$ 、 $CKE0=0$ 、 $TE=1$ 的状态，根据以下设定切换到端口管脚时，会发生半个周期的低电平输出。

1. 发送串行数据完成
2. TE 位 = 0
3. $C/\bar{A}$ 位 = 0...切换到端口输出
4. 发生低电平输出 (参考图 10.37)

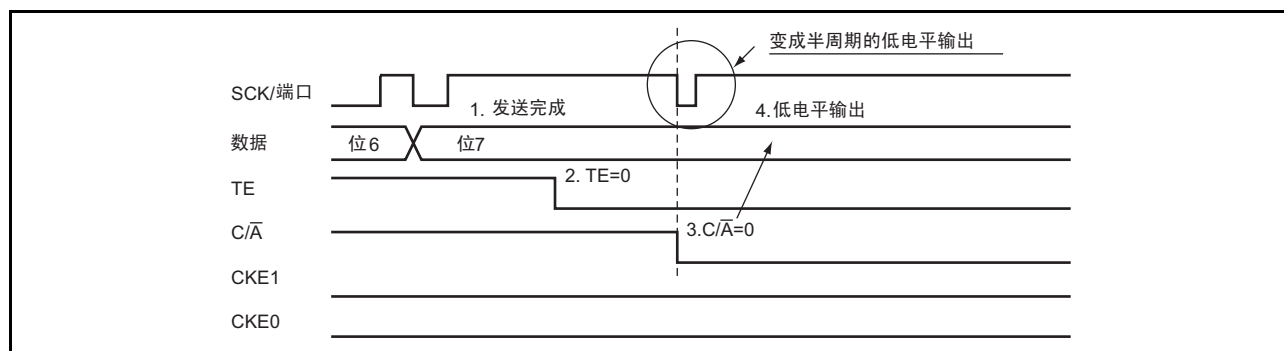


图 10.37 从 SCK 管脚向端口管脚切换时的操作

(b) 使用中的注意事项

为避免将 SCK 管脚切换为端口时，发生低电平输出，请按以下的顺序进行。

由于这个顺序是将 SCK 管脚作为一个输入管脚，所以请事先使用外部电路上拉 SCK / 端口管脚。

在 $\text{DDR}=1$ 、 $\text{DR}=1$ 、 =1 、 $\text{CKE1}=0$ 、 $\text{CKE0}=0$ 、 $\text{TE}=1$ 的状态开始按照 1 ~ 5 的顺序进行设定。

1. 串行数据发送完成。
2. TE 位 = 0。
3. CKE1 位 = 1。
4. $\text{C}/\overline{\text{A}}$ 位 = 0...切换到端口输出。
5. CKE1 位 = 0。

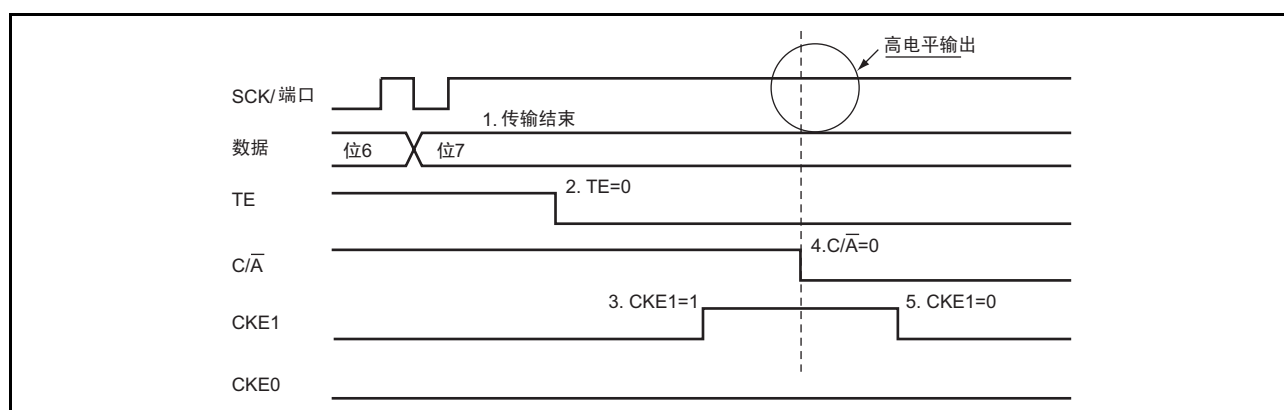


图 10.38 从 SCK 管脚向端口管脚切换时的运行 (避免低电平输出的样例)

第 11 章 CAN 总线模块（HCAN）【只对应 H8S/2282 群】

HCAN 是在汽车及产业机器等系统中进行实时通信的 CAN（Controller Area Network）总线模块。关于 CAN 的规格，请参考「BOSCH CAN Specification Version 2.0 1991, Robert Bosch GmbH」。HCAN 的框图如图 11.1 所示。

【注】 H8S/2280 群没有此功能。

11.1 特点

- CAN 版本：对应 Bosch 2.0B active
 - 通信方式：NRZ（Non-Return to Zero）方式（有 bit-staffing 功能）
 - 广播通信方式
 - 传送线路：双向两线式串行通信
 - 通信速度：最高 1Mbps
 - 数据长度：0 ～ 8 字节
- 通道数：1
- 数据缓冲器：16（接收专用缓冲器 1 个，可设定发送 / 接收缓冲器 15 个）
- 发送数据方式：2 种
 - 信箱号顺序（上升顺序）
 - 信息优先级（Identifier）高的顺序
- 接收数据方式：2 种
 - 信息标识符一致（发送 / 接收设定缓冲器）
 - 屏蔽信息标识符，接收信息（接收专用）
- 中断源：12
 - 各种错误中断
 - 复位处理中断
 - 接收信息中断
 - 发送信息中断
- HCAN 运行模式
- 支持各种模式
 - 硬件复位
 - 软件复位
 - 通常状态（error active, error passive）
 - 总线关闭状态
 - HCAN 配置模式
 - HCAN 休眠模式
 - HCAN HALT 模式
- 可设定模块待机模式

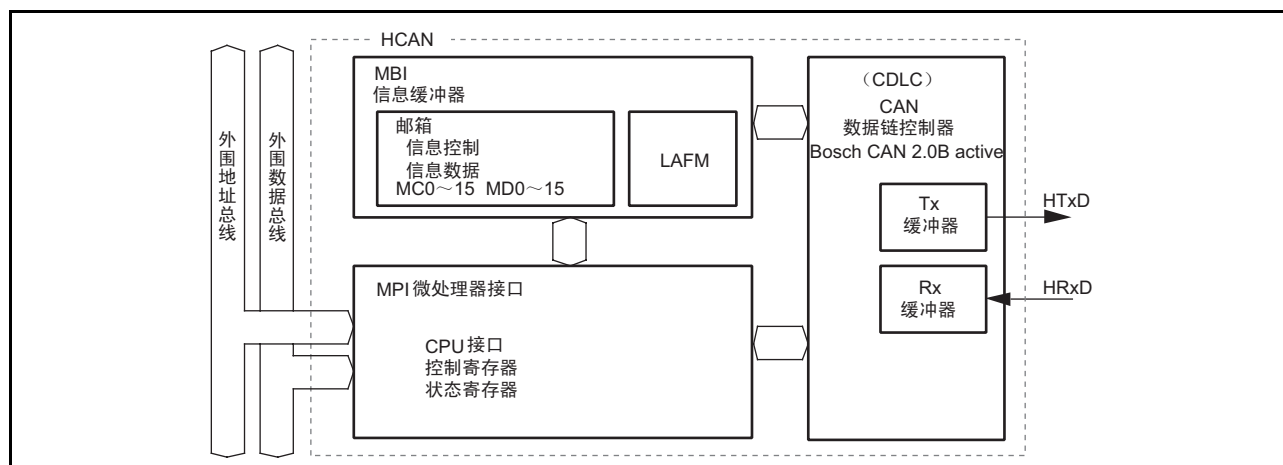


图 11.1 HCAN 的框图

信息缓冲器（Message Buffer Interface）

MBI 由信箱和局部验收滤波器掩码（LAFM）构成，用于保存 CAN 发送 / 接收信息（标识符及数据等）。从 CPU 写入发送信息。用 CDLC 自动保存接收到的数据。

微处理器接口（MicroProcessor Interface）

MPI 由 CPU 的总线接口、控制寄存器、状态寄存器等构成，用于控制 HCAN 内部数据及状态。

CAN 数据链接控制器（CAN Data Link Controller）

CDLC 依据 Bosch CAN ver.2.0B 标准，执行发送 / 接收信息（数据帧、远程帧、错误帧、过载帧、帧间隔）、CRC 检测、总线仲裁等。

11.2 输入 / 输出管脚

HCAN 的管脚结构如表 11.1 所示。

使用 HCAN 管脚时，必须要设定为 HCAN 配置模式（初始设定时间：MCR0=1 且 GSR3=1）。

表 11.1 管脚结构

名称	简称	输入 / 输出	功能
HCAN 传输数据管脚	HTxD	输出	CAN 总线发送管脚
HCAN 接收数据管脚	HRxD	输入	CAN 总线接收管脚

管脚和 CAN 总线之间必须要总线驱动器。推荐使用与 Philips PCA82C250 兼容的产品。

11.3 寄存器的说明

HCAN 中有以下寄存器。

- 主控制寄存器（MCR）
- 通用状态寄存器（GSR）
- 位配置寄存器（BCR）
- 信箱结构寄存器（MBCR）
- 发送等待寄存器（TXPR）
- 取消发送等待寄存器（TXCR）
- 发送应答寄存器（TXACK）
- 取消发送应答寄存器（ABACK）
- 接收完成寄存器（RXPR）
- 远程请求寄存器（RFPR）
- 中断寄存器（IRR）
- 信箱中断屏蔽寄存器（MBIMR）
- 中断屏蔽寄存器（IMR）
- 接收错误计数器（REC）
- 发送错误计数器（TEC）
- 未读信息状态寄存器（UMSR）
- 局部验收滤波器掩码 H（LAFMH）
- 局部验收滤波器掩码 L（LAFML）
- 信息控制（8 位 × 8 × 16）（MC0 ～ MC15）
- 信息数据（8 位 × 8 × 16）（MD0 ～ MD15）

11.3.1 主控制寄存器 (MCR)

MCR 控制 HCAN 的运行。

位	位名	初始值	R/W	说明
7	MCR7	0	R/W	解除 HCAN 休眠模式 如果将此位置为 1，检测出的 CAN 总线运行，就会自动解除 HCAN 休眠模式。
6	—	0	R	保留位 读取值常为 0。写入的值只为 0。
5	MCR5	0	R/W	HCAN 休眠模式 如果将此位置为 1，HCAN 会迁移到 HCAN 休眠模式。清除此位后，HCAN 休眠模式被解除。
4、3	—	全为 0	R	保留位 读取值常为 0。写入的值只为 0。
2	MCR2	0	R/W	信息发送方式 0: 根据信息校验的优先顺序决定发送顺序。 1: 根据信箱号码的优先顺序决定发送顺序。优先顺序是 TXPR1 > ... > TXPR15。
1	MCR1	0	R/W	HALT 请求 如果将此位置为 1，HCAN 会迁移到 HCAN HALT 模式。清除此位后，HCAN HALT 模式也被解除。
0	MCR0	1	R/W	复位请求 如果将此位置为 1，HCAN 迁移到复位模式。详细请参考「11.4.1 硬件复位和软件复位」。 [置位条件] • 加电复位 • 硬件待机 • 软件待机 • 写入 1 (软件复位) [清除条件] • GSR 的 GSR3 为 1 的状态，在此位写入 0 时

11.3.2 通用状态寄存器 (GSR)

GSR 表示 HCAN 的状态。

位	位名	初始值	R/W	说明
7 ~ 4	—	全部为 0	R	保留位 读取值常为 0。写入值只为 0。
3	GSR3	1	R	复位状态位 表示 HCAN 模块是通常运行状态还是复位状态。写入定义为无效。 [置位条件] - HCAN 内部的复位完成，进入配置模式时。 - 休眠模式 [清除条件] - 清除 MCR 的 MCR0，进入通常运行状态时（清除 MCR0 后，会发生延迟直至 GSR3 被清除）。
2	GSR2	1	R	发送信息状态标记 它是表示正在发送信息的标记。写入定义为无效。 [置位条件] - EOF (End of Frame) 后的 Intermission 第 3 位 [清除条件] - 开始发送信息 (SOF)
1	GSR1	0	R	发送 / 接收警告标记 写入定义为无效。 [清除条件] $TEC < 96$ 且 $REC < 96$ 时 $TEC \geq 256$ 时 [置位条件] $TEC \geq 96$ 或 $REC \geq 96$ 时
0	GSR0	0	R	总线关闭标记 写入定义为无效。 [置位条件] $TEC \geq 256$ 时（总线关闭状态） [清除条件] 从总线关闭恢复时

11.3.3 位配置寄存器 (BCR)

BCR 设定 HCAN 的位时序和波特率。各参数的详细请参考「11.4.2 硬件复位后的初始设定」。

位	位名	初始值	R/W	说明
15 14	BCR7 BCR6	0 0	R/W R/W	Re-Synchronization Jump Width (SJW) (再同步补偿宽度) 设定最大位同步宽度 00: 1 time quantum 01: 2 time quanta 10: 3 time quanta 11: 4 time quanta
13 12 11 10 9 8	BCR5 BCR4 BCR3 BCR2 BCR1 BCR0	0 0 0 0 0 0	R/W R/W R/W R/W R/W R/W	波特率预分频 (BRP) 设定 time quantum 的长度。 000000: 2 系统时钟 000001: 4 系统时钟 000010: 6 系统时钟 : 111111: 128 系统时钟
7	BCR15	0	R/W	位采样点 (BSP) 设定采样数据点。 0: 位采样 1 处 (TSEG1 的末尾) 1: 位采样 3 处 (TSEG1 的末尾和前后 1 time quantum)
6 5 4	BCR14 BCR13 BCR12	0 0 0	R/W R/W R/W	时间段 2 (TSEG2) 在 2 ~ 8 time quanta 的范围内设定 TSEG2 的宽度。 000: 禁止设定 001: 2 time quanta 010: 3 time quanta 011: 4 time quanta 100: 5 time quanta 101: 6 time quanta 110: 7 time quanta 111: 8 time quanta

位	位名	初始值	R/W	说明
3	BCR11	0	R/W	时间段 1 (TSEG1)
2	BCR10	0	R/W	在 4 ~ 16 time quanta 范围内设定 TSEG1 (PRSEG+PHSE
1	BCR9	0	R/W	G1) 的宽度。
0	BCR8	0	R/W	0000: 禁止设定
				0001: 禁止设定
				0010: 禁止设定
				0011: 4 time quanta
				0100: 5 time quanta
				0101: 6 time quanta
				0110: 7 time quanta
				0111: 8 time quanta
				1000: 9 time quanta
				1001: 10 time quanta
				1010: 11 time quanta
				1011: 12 time quanta
				1100: 13 time quanta
				1101: 14 time quanta
				1110: 15 time quanta
				1111: 16 time quanta

11.3.4 信箱配置寄存器 (MBCR)

MBCR 设定各信箱的发送 / 接收方向。

位	位名	初始值	R/W	说明
15	MBCR7	0	R/W	该位分别设置相应信箱 1 到 15 的发送 / 接收方向。MBCRn (n=1 ~ 15) 决定了信箱 n 的发送 / 接收方向。 0: 对应的信箱被设定为发送用。 1: 对应的信箱被设定为接收用。 位 8 是保留位, 读取值常为 1。写入值也常为 1。
14	MBCR6	0	R/W	
13	MBCR5	0	R/W	
12	MBCR4	0	R/W	
11	MBCR3	0	R/W	
10	MBCR2	0	R/W	
9	MBCR1	0	R/W	
8	—	1	R	
7	MBCR15	0	R/W	
6	MBCR14	0	R/W	
5	MBCR13	0	R/W	
4	MBCR12	0	R/W	
3	MBCR11	0	R/W	
2	MBCR10	0	R/W	
1	MBCR9	0	R/W	
0	MBCR8	0	R/W	

11.3.5 发送等待寄存器 (TXPR)

发送的信息被保存在信箱 (缓冲) 后, TXPR 置为发送等待状态 (等待 CAN 总线仲裁)。

位	位名	初始值	R/W	说明
15	TXPR7	0	R/W	该位分别设置相应信箱 1 至 15 置为发送等待状态 (等待 CAN 总线仲裁)。如果将 TXPRn (n=1 ~ 15) 置为 1, 信箱 n 的信息变为等待发送状态。 [清除条件] 信息发送完成时 发送取消完成时 位 8 是保留位, 读取值常为 0。写入值也常为 0。
14	TXPR6	0	R/W	
13	TXPR5	0	R/W	
12	TXPR4	0	R/W	
11	TXPR3	0	R/W	
10	TXPR2	0	R/W	
9	TXPR1	0	R/W	
8	—	0	R	
7	TXPR15	0	R/W	
6	TXPR14	0	R/W	
5	TXPR13	0	R/W	
4	TXPR12	0	R/W	
3	TXPR11	0	R/W	
2	TXPR10	0	R/W	
1	TXPR9	0	R/W	
0	TXPR8	0	R/W	

11.3.6 取消发送等待寄存器 (TXCR)

TXCR 取消信箱内 (缓冲) 发送等待信息。

位	位名	初始值	R/W	说明
15	TXCR7	0	R/W	该位分别取消相应信箱 1 到 15 的等待发送信息。如果将 TXCRn (n=1 ~ 15) 置为 1, 信箱 n 的等待发送信息就会被取消。 [清除条件] • 发送信息正常取消, TXPR 被清除时 位 8 是保留位, 读取值常为 0。写入值也常为 0。
14	TXCR6	0	R/W	
13	TXCR5	0	R/W	
12	TXCR4	0	R/W	
11	TXCR3	0	R/W	
10	TXCR2	0	R/W	
9	TXCR1	0	R/W	
8	—	0	R	
7	TXCR15	0	R/W	
6	TXCR14	0	R/W	
5	TXCR13	0	R/W	
4	TXCR12	0	R/W	
3	TXCR11	0	R/W	
2	TXCR10	0	R/W	
1	TXCR9	0	R/W	
0	TXCR8	0	R/W	

11.3.7 发送应答寄存器 (TXACK)

TXACK 是信箱的信息正常被发送的状态寄存器。

位	位名	初始值	R/W	说明
15	TXACK7	0	R/(W)*	该位是相应信箱 1 到 15 的信息被正常发送的状态标记。信箱 n (n=1 ~ 15) 的信息被正常发送时, TXACK n 被置为 1。 [设定条件] • 对应信箱的信息发送完成时 [清除条件] • 写入 1 位 8 为保留位, 读取值常为 0。写入值也常为 0。
14	TXACK6	0	R/(W)*	
13	TXACK5	0	R/(W)*	
12	TXACK4	0	R/(W)*	
11	TXACK3	0	R/(W)*	
10	TXACK2	0	R/(W)*	
9	TXACK1	0	R/(W)*	
8	—	0	R	
7	TXACK15	0	R/(W)*	
6	TXACK14	0	R/(W)*	
5	TXACK13	0	R/(W)*	
4	TXACK12	0	R/(W)*	
3	TXACK11	0	R/(W)*	
2	TXACK10	0	R/(W)*	
1	TXACK9	0	R/(W)*	
0	TXACK8	0	R/(W)*	

【注】 * 为了清除标记只能写入 1。

11.3.8 取消应答寄存器（ABACK）

ABACK 是表示信箱内的发送信息被取消的状态寄存器。

位	位名	初始值	R/W	说明
15	ABACK7	0	R/(W)*	该位是相应信箱 1 到 15 的发送信息正常被取消的状态标记。信箱 n（n=1 ~ 15）的信息被正常取消时，ABACK n 会被设置为 1。 [置位条件] • 完成取消对应信箱的信息时 [清除条件] • 写入 1 位 8 是保留位，读取值常为 0。写入值也常为 0。
14	ABACK6	0	R/(W)*	
13	ABACK5	0	R/(W)*	
12	ABACK4	0	R/(W)*	
11	ABACK3	0	R/(W)*	
10	ABACK2	0	R/(W)*	
9	ABACK1	0	R/(W)*	
8	—	0	R	
7	ABACK15	0	R/(W)*	
6	ABACK14	0	R/(W)*	
5	ABACK13	0	R/(W)*	
4	ABACK12	0	R/(W)*	
3	ABACK11	0	R/(W)*	
2	ABACK10	0	R/(W)*	
1	ABACK9	0	R/(W)*	
0	ABACK8	0	R/(W)*	

【注】 * 为了清除标记只能写入 1。

11.3.9 接收完成寄存器（RXPR）

RXPR 表示信箱正常接收信息（数据帧或远程帧）的状态寄存器。因此，接收远程帧时，该寄存器的位置 1，则对应的远程请求寄存器（RFPR）也要同时置位。

位	位名	初始值	R/W	说明
15	RXPR7	0	R/(W)*	信箱 n（n=0 ~ 15）正常接收信息时，RXPRn 被置为 1。 [置位条件] • 对应的信箱接收完数据帧或远程帧时 [清除条件] • 写入 1
14	RXPR6	0	R/(W)*	
13	RXPR5	0	R/(W)*	
12	RXPR4	0	R/(W)*	
11	RXPR3	0	R/(W)*	
10	RXPR2	0	R/(W)*	
9	RXPR1	0	R/(W)*	
8	RXPR0	0	R/(W)*	
7	RXPR15	0	R/(W)*	
6	RXPR14	0	R/(W)*	
5	RXPR13	0	R/(W)*	
4	RXPR12	0	R/(W)*	
3	RXPR11	0	R/(W)*	
2	RXPR10	0	R/(W)*	
1	RXPR9	0	R/(W)*	
0	RXPR8	0	R/(W)*	

【注】 * 为了清除标记只能写入 1。

11.3.10 远程请求寄存器（RFPR）

RFPR 是信箱接收到远程帧的状态寄存器。该寄存器被设置 1 后，则对应的接收完成寄存器（RXPR）也同时被设置 1。

位	位名	初始值	R/W	说明
15	RFPR7	0	R/(W)*	信箱 n（n=0～15）正常接收到远程帧时，RFPRn（n=0～15）被置为 1。 [置位条件] • 对应的信箱接收完远程帧时 [清除条件] • 写入 1
14	RFPR6	0	R/(W)*	
13	RFPR5	0	R/(W)*	
12	RFPR4	0	R/(W)*	
11	RFPR3	0	R/(W)*	
10	RFPR2	0	R/(W)*	
9	RFPR1	0	R/(W)*	
8	RFPR0	0	R/(W)*	
7	RFPR15	0	R/(W)*	
6	RFPR14	0	R/(W)*	
5	RFPR13	0	R/(W)*	
4	RFPR12	0	R/(W)*	
3	RFPR11	0	R/(W)*	
2	RFPR10	0	R/(W)*	
1	RFPR9	0	R/(W)*	
0	RFPR8	0	R/(W)*	

【注】 * 为了清除标记只能写入 1。

11.3.11 中断寄存器 (IRR)

IRR 是中断标记寄存器。

位	位名	初始值	R/W	说明
15	IRR7	0	R/(W)*	过载帧中断标记 [置位条件] • 在错误主动 / 被动状态发送过载帧时 [清除条件] • 写入 1
14	IRR6	0	R/(W)*	总线关闭中断标记 它是通过发送错误计数器表示总线关闭状态的状态标记。 [设定条件] • $TEC \geq 256$ 时 [清除条件] • 写入 1
13	IRR5	0	R/(W)*	错误无源中断标记 通过发送 / 接收错误计数器表示错误无源状态的状态标记。 [设定条件] $TEC \geq 128$ 或者 $REC \geq 128$ 时 [清除条件] • 写入 1
12	IRR4	0	R/(W)*	接收过载警告中断标记 它是根据接收错误计数器表示错误警告状态的标记。 [设定条件] $REC \geq 96$ 时 [清除条件] • 写入 1
11	IRR3	0	R/(W)*	发送超负荷警告中断标记 它是通过发送错误计数器表示错误警告状态的状态标记。 [设定条件] • $TEC \geq 96$ 时 [清除条件] • 写入 1

位	位名	初始值	R/W	说明
10	IRR2	0	R	远程帧请求中断标记 它是表示信箱中接收到远程帧的状态标记。 [置位条件] • 接收远程帧完成, 对应 MBIMR 中的位为 0 时 [清除条件] • 清除 RFPR (远程请求寄存器) 的全部位
9	IRR1	0	R	接收信息中断标记 它是表示信箱正常接收到接收信息的状态标记。 [设定条件] • 接收完数据帧及远程帧, 对应 MBIMR 中的位为 0 时 [清除条件] • 清除 RXPR (接收完成寄存器) 的全部位
8	IRR0	0	R/(W)*	复位中断标记 它是 HCAN 模块被复位的状态标记。规定该位在中断屏蔽寄存器 (IMR) 中不可屏蔽。加电复位后及软件待机状态恢复后, 没有清除该位时, 如果中断控制器允许中断, 直接就进行中断处理。 [设定条件] • 上电复位及软件待机后复位处理完成时 [清除条件] • 写入 1
7 ~ 5	—	全为 0	—	保留位 读取时通常读出 0。写入的值也通常设为 0。
4	IRR12	0	R/(W)*	总线运行中断标记 它是表示 HCAN 模块在 HCAN 休眠模式下检测出总线运行的支配位的状态标记。 [设定条件] • 检测出 HCAN 休眠模式下的总线运行 (支配位) 时 [清除条件] • 写入 1
3、2	—	全部为 0	—	保留位 读取时通常读出 0。写入的值通常也设为 0。
1	IRR9	0	R	未读中断标记 它是表示由于有新的接收信息, 未读信息被覆盖的状态标记。 [设定条件] • UMSR (未读信息状态寄存器) 被置位时 [清除条件] • 清除 UMSR (未读信息状态寄存器) 的全部位
0	IRR8	0	R/(W)*	信箱为空中断标记 表示可以把下一次要发送的信息保存到信箱的状态标记。 [设定条件] • 发送完成及取消发送完成, TXPR (等待发送寄存器) 被清除时 [清除条件] • 写入 1

【注】 * 为了清除标记只能写入 1。

11.3.12 信箱中断屏蔽寄存器（MBIMR）

MBIMR 用于控制各信箱的中断请求是否屏蔽。

位	位名	初始值	R/W	说明
15	MBIMR7	1	R/W	信箱中断屏蔽 MBIMR n（n=0 ~ 15）清零，则信箱 n 的中断请求变为允许，如果置为 1，中断请求就被屏蔽。 中断源是指在发送用信箱中，根据发送完成或取消来清除 TXPR，在接收信箱中，根据接收完成置位 RXPR。
14	MBIMR6	1	R/W	
13	MBIMR5	1	R/W	
12	MBIMR4	1	R/W	
11	MBIMR3	1	R/W	
10	MBIMR2	1	R/W	
9	MBIMR1	1	R/W	
8	MBIMR0	1	R/W	
7	MBIMR15	1	R/W	
6	MBIMR14	1	R/W	
5	MBIMR13	1	R/W	
4	MBIMR12	1	R/W	
3	MBIMR11	1	R/W	
2	MBIMR10	1	R/W	
1	MBIMR9	1	R/W	
0	MBIMR8	1	R/W	

11.3.13 中断屏蔽寄存器 (IMR)

IMR 是通过 IRR 的中断标记确定是否响应中断请求。复位中断标记不能屏蔽。

位	位名	初始值	R/W	说明
15	IMR7	1	R/W	过载帧中断屏蔽 如果将此位清零, 根据 IRR7 标记, 中断请求 (OVR0) 可响应, 置位后被屏蔽。
14	IMR6	1	R/W	总线关闭中断屏蔽 如果将此位清零, 根据 IRR6 标记, 中断请求 (ERS0) 可响应, 置位后被屏蔽。
13	IMR5	1	R/W	错误被动中断屏蔽 如果将此位清零, 根据 IRR5 标记, 中断请求 (ERS0) 可响应, 置位后被屏蔽。
12	IMR4	1	R/W	接收过载警告中断屏蔽 如果将此位清零, 根据 IRR4 标记, 中断请求 (OVR0) 可响应, 置位后会被屏蔽。
11	IMR3	1	R/W	发送过载警告屏蔽 如果将此位清零, 根据 IRR3 标记, 中断请求 (OVR0) 可响应, 置位后被屏蔽。
10	IMR2	1	R/W	远程帧请求中断屏蔽 如果将此位清零, 根据 IRR2 标记, 中断请求 (OVR0) 可响应, 置位后被屏蔽。
9	IMR1	1	R/W	接收信息中断屏蔽 如果将此位清零, 根据 IRR1 标记, 中断请求 (RM1) 可响应, 置位后被屏蔽。
8	—	1	R	读取值常为 0。写入值也常为 0。
7 ~ 5	—	全为 1	R	读取值常为 1。写入值也常为 1。
4	IMR12	1	R/W	总线运行中断屏蔽 如果将此位清零, 根据 IRR12 标记, 中断请求 (OVR0) 可响应, 置位后被屏蔽。
3、2	—	全为 1	R	读取值常为 1。写入值也常为 1。
1	IMR9	1	R/W	未读中断屏蔽 如果将此位清零, 根据 IRR9 标记, 中断请求 (OVR0) 可响应, 置位后被屏蔽。
0	IMR8	1	R/W	如果将此位清零, 根据 IRR8 标记, 中断请求 (SLE0) 可响应, 置位后被屏蔽。

11.3.14 接收错误计数器 (REC)

REC 是 8 位只读寄存器，计算 CAN 总线上的接收信息错误。数值在 CAN 协议中有规定。

11.3.15 发送错误计数器 (TEC)

TEC 是 8 位只读寄存器，计算 CAN 总线上的发送信息错误。数值在 CAN 协议中有规定。

11.3.16 未读信息状态寄存器 (UMSR)

UMSR 表示每个信箱（缓存）新接收信息覆盖未读信息的状态寄存器。因此，新接收信息写入未读信息时，旧的数据就会丢失。

位	位名	初始值	R/W	说明
15	UMSR7	0	R/(W)*	[清除条件] 写入 1 接收到信息后，在没有读之前，写入新信息 [置位条件] 清除 RXPR 前，接收到新信息时
14	UMSR6	0	R/(W)*	
13	UMSR5	0	R/(W)*	
12	UMSR4	0	R/(W)*	
11	UMSR3	0	R/(W)*	
10	UMSR2	0	R/(W)*	
9	UMSR1	0	R/(W)*	
8	UMSR0	0	R/(W)*	
7	UMSR15	0	R/(W)*	
6	UMSR14	0	R/(W)*	
5	UMSR13	0	R/(W)*	
4	UMSR12	0	R/(W)*	
3	UMSR11	0	R/(W)*	
2	UMSR10	0	R/(W)*	
1	UMSR9	0	R/(W)*	
0	UMSR8	0	R/(W)*	

【注】 * 为了清除标记只能写入 1。

11.3.17 局部验收滤波器掩码 (LAFML、LAFMH)

LAFML、LAFMH 将保存在信箱 0 中的标识符每位都设为无关位。详细请参考「11.4.4 接收信息」。标识符和屏蔽位的关系如下。

• LAFML

位	位名	初始值	R/W	说明
15	LAFML7	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-7。
14	LAFML6	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-6。
13	LAFML5	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-5。
12	LAFML4	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-4。
11	LAFML3	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-3。
10	LAFML2	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-2。
9	LAFML1	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-1。
8	LAFML0	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-0。
7	LAFML15	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-15。
6	LAFML14	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-14。
5	LAFML13	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-13。
4	LAFML12	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-12。
3	LAFML11	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-11。
2	LAFML10	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-10。
1	LAFML9	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-9。
0	LAFML8	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-8。

• LAFMH

位	位名	初始值	R/W	说明
15	LAFMH7	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-20。
14	LAFMH6	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-19。
13	LAFMH5	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-18。
12 ~ 10	—	全部为 0	R	保留位 读取时通常读出 0。写入的值也通常设为 0。
9	LAFMH1	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-17。
8	LAFMH0	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-16。
7	LAFMH15	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-28。
6	LAFMH14	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-27。
5	LAFMH13	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-26。
4	LAFMH12	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-25。
3	LAFMH11	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-24。
2	LAFMH10	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-23。
1	LAFMH9	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-22。
0	LAFMH8	0	R/W	为 1 时, 屏蔽接收 ID 的 ID-21。

11.3.18 信息控制（MC0 ～ MC15）

通常每个信箱的信息控制由 8 位 ×8 个寄存器构成的。HCAN 中有 16 个信息控制。信息控制寄存器内容由 RAM 中的数据构成，所以加载电源后的初始值是不确定的。必须写入 0 或 1 来初始化。寄存器的名称和信箱的关系如下。

信箱0	MC0[1]	MC0[2]	MC0[3]	MC0[4]	MC0[5]	MC0[6]	MC0[7]	MC0[8]
信箱1	MC1[1]	MC1[2]	MC1[3]	MC1[4]	MC1[5]	MC1[6]	MC1[7]	MC1[8]
信箱2	MC2[1]	MC2[2]	MC2[3]	MC2[4]	MC2[5]	MC2[6]	MC2[7]	MC2[8]
信箱3	MC3[1]	MC3[2]	MC3[3]	MC3[4]	MC3[5]	MC3[6]	MC3[7]	MC3[8]
信箱15	MC15[1]	MC15[2]	MC15[3]	MC15[4]	MC15[5]	MC15[6]	MC15[7]	MC15[8]

图 11.2 消息控制结构

信息控制的设定方法如下页所示。ID 和寄存器位名对应如下。

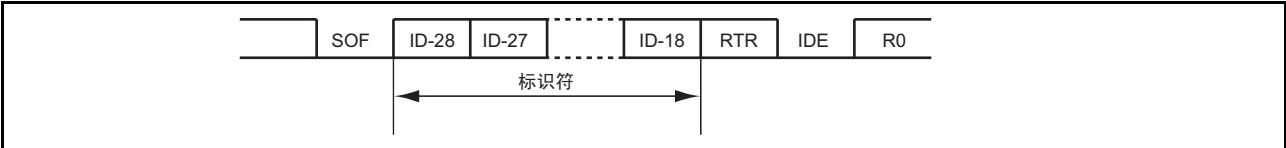


图 11.3 标准格式

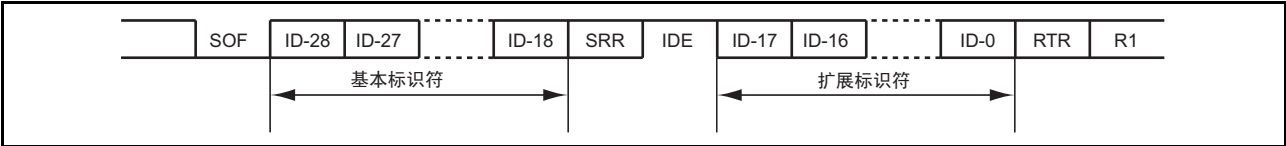


图 11.4 扩展格式

寄存器名称	位	位名	R/W	说明
MCx[1]	7 ~ 4	—	R/W	初始值不确定。必须写入 0 或 1 进行初始化。
	3 ~ 0	DLC3 ~ DLC0	R/W	数据长度码 在 0 ~ 8 字节范围内, 设定数据帧的数据长度或远程帧中要求的数据长度。 0000: 0 字节 0001: 1 字节 0010: 2 字节 0011: 3 字节 0100: 4 字节 0101: 5 字节 0110: 6 字节 0111: 7 字节 1000: 8 字节 : 1111: 8 字节
MCx[2] ~ MCx[4]	全为 7 ~ 0	—	R/W	初始值不确定。必须写入 0 或 1 进行初始化。
MCx[5]	7 ~ 5	ID-20 ~ ID-18	R/W	设定标识符中的 ID-20 ~ ID-18 位。
	4	RTR	R/W	远程传输要求 (RTR) 用于识别数据帧和远程帧的位。 0: 数据帧 1: 远程帧
	3	IDE	R/W	标识符 Identifier 扩展 (IDE) 用于识别标准格式和延长格式的位。 0: 标准格式 1: 延长格式
	2	—	R/W	初始值不确定。必须写入 0 或 1 进行初始化。
	1 ~ 0	ID-17 ~ ID-16	R/W	设定标识符 Identifier 的 ID-17 ~ ID-16 位。
MCx[6]	7 ~ 0	ID-28 ~ ID-21	R/W	设定标识符 Identifier 的 ID-28 ~ ID-21 位。
MCx[7]	7 ~ 0	ID-7 ~ ID-0	R/W	设定标识符 Identifier 的 ID-7 ~ ID-0 位。
MCx[8]	7 ~ 0	ID-15 ~ ID-8	R/W	设定标识符 Identifier 的 ID-15 ~ ID-8 位。

【注】×: 表示信箱号码。

11.3.19 信息数据（MD0 ～ MD15）

通常每个信箱的信息数据寄存器由 8 位 × 8 个寄存器构成。HCAN 中有 16 个信息数据字节。信息数据寄存器内容是由 RAM 中的数据构成的，因此加载电源后初始值不确定。必须写入 0 或 1 进行初始化。寄存器的名称和信箱的关系如下。

信箱0	MD0[1]	MD0[2]	MD0[3]	MD0[4]	MD0[5]	MD0[6]	MD0[7]	MD0[8]
信箱1	MD1[1]	MD1[2]	MD1[3]	MD1[4]	MD1[5]	MD1[6]	MD1[7]	MD1[8]
信箱2	MD2[1]	MD2[2]	MD2[3]	MD2[4]	MD2[5]	MD2[6]	MD2[7]	MD2[8]
信箱3	MD3[1]	MD3[2]	MD3[3]	MD3[4]	MD3[5]	MD3[6]	MD3[7]	MD3[8]
信箱15	MD15[1]	MD15[2]	MD15[3]	MD15[4]	MD15[5]	MD15[6]	MD15[7]	MD15[8]

图 11.5 信息数据结构

11.4 运行说明

11.4.1 硬件复位和软件复位

HCAN 的复位包含硬件复位和软件复位。

- 硬件复位

在加电复位、硬件待机、软件待机中 MCR 的复位请求位（MCR0）和 GSR 的复位状态位（GSR3）被自动置位，HCAN 就会被初始化。通过硬件复位，除信息控制域和信息数据域外，内部寄存器全部会被初始化。

- 软件复位

用软件将 MCR 的 MCR0 置位，就可复位 HCAN。软件复位中，虽然错误计数器（TEC、REC）被初始化，但是其它寄存器并不被初始化。CAN 控制器在发送或接收时，MCR0 置位，发送或接收该信息完成后，进入初始状态。初始化完成后，GSR 的 GSR3 会被置位。

11.4.2 硬件复位后的初始设定

硬件复位后请进行以下初始设定。

1. 清除中断寄存器（IRR）的 IRR0 位
2. 设定速率
3. 设定各信箱的发送接收方向
4. 信箱信息（RAM）的初始化
5. 设定信息发送方式

初始设定必须在 HCAN 为配置模式状态下进行。配置模式是通过复位将 GSR 的 GSR3 位置位的状态。初始设定后清除 MCR 的 MCR0，GSR 的 GSR3 就会被自动清除，配置模式也被解除（HCAN 内部复位需要时间，因此清除 MCR0 后到 GSR3 被清除前会发生延迟）。配置模式一旦被解除，HCAN 会进入上电序列，检测出连续 11 位隐性位的同时，就可以和 CAN 总线通信。

(1) IRR0 的清除

加电复位后及从软件模式恢复后，必须设置复位中断标记 (IRR0)。允许中断后，会马上发生 HCAN 中断请求，因此请清除 IRR0。

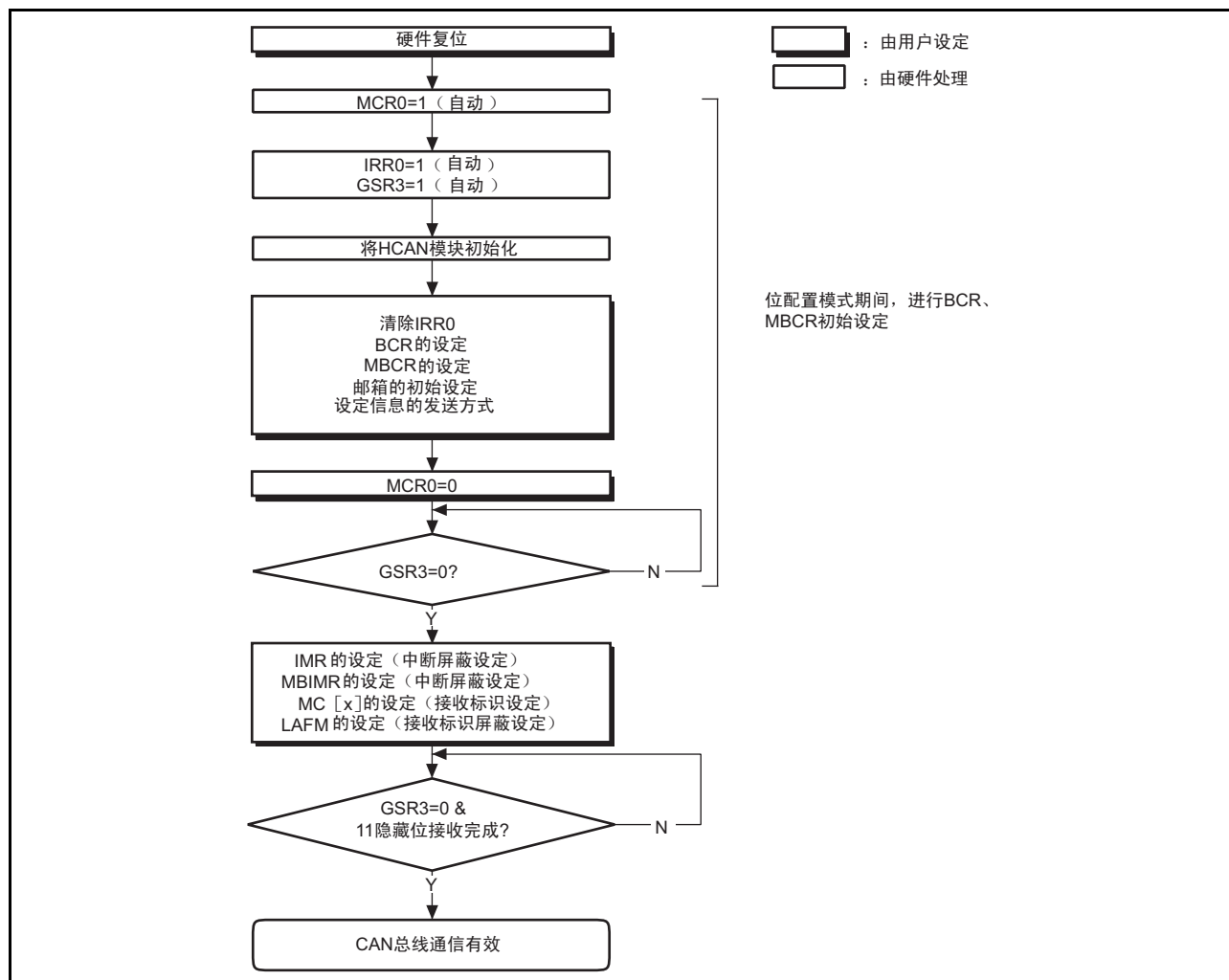


图 11.6 硬件复位时的流程图

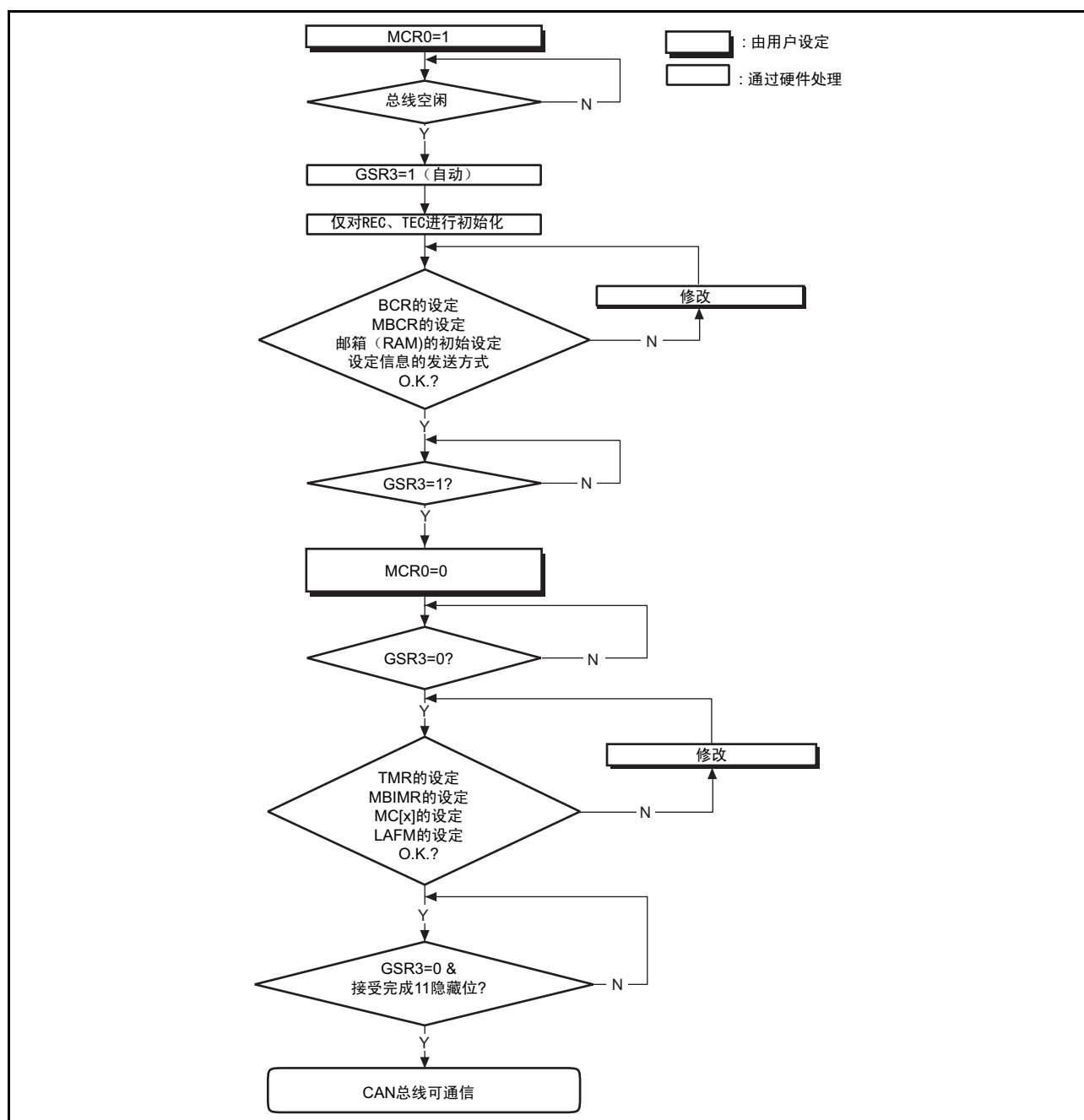


图 11.7 软件复位时的流程图

(2) 位速率及位时序的设定

通过位配置寄存器（BCR）设定位速率及位时序。将连接在 CAN 总线上的 CAN 控制器全部设定为相同波特率，相同位宽。每位时间由可以设定的 Time Quantaum（TQ）的总数构成。

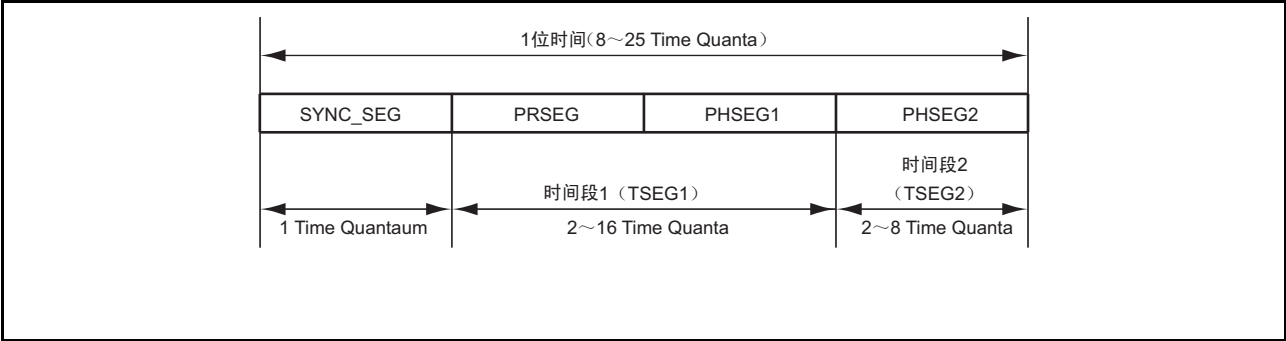


图 11.8 1 位时间的详细描述

SYNC_SEG 是为了取得 CAN 总线上节点同步的时间段。通常都是在这一部分发生位沿变化。PRSEG 是补偿网络间物理延迟的时间段。PHSEG1 是补偿相位偏移（plus/positive）的缓冲段。要重新取得同步（Re-Synchronization），需要延长 PHSEG1。PHSEG2 是补偿相位偏移（minus/negative）的缓冲段。要重新取得同步（Re-Synchronization），需要缩短 PHSEG2。BCR 中能够设定的值（TSEG1、TSEG2、BRP、BSP、SJW）的范围如表 11.2 所示。

表 11.2 BCR 中能够设定的寄存器值范围

名称	简称	最小值	最大值
时间段 1	TSEG1	3*2	15
时间段 2	TSEG2	1*3	7
波特率预分频	BRP	0	63
位采样点	BSP	0	1
Re-Synchronization Jump Width (再同步补偿宽度)	SJW*1	0	3

【注】 *1 SJW 根据 CAN 规格规定为 $3 \geq \text{SJW} \geq 0$ 。
*2 TSEG2 的最小值根据 CAN 规格规定为 $\text{TSEG2} \geq \text{SJW}$ 。
*3 TSEG1 的最小值根据 CAN 规格规定为 $\text{TSEG1} > \text{TSEG2}$ 。

Time Quanta (TQ) 是系统时钟数的整数倍, 根据波特率预分频 (BRP) 作出如下规定。f CLK 表示系统时钟频率。

$$TQ = 2 \times (\text{BRP 的设定值} + 1) / f_{CLK}$$

位时间及位速率用下式可计算得出。

$$\text{位时间} = TQ \times (3 + TSEG1 + TSEG2)$$

$$\text{位速率} = 1 / \text{位时间}$$

$$= f_{CLK} / \{2 \times (\text{BRP 的设定值} + 1) \times (3 + TSEG1 + TSEG2)\}$$

【注】f_{CLK}=φ (系统时钟)

BRP、TSEG1、TSEG2 使用 BCR 值

【例】系统时钟为 20MHz、BRP 的设定值=B'000000、TSEG1=B'0100、TSEG2=B'011 时

$$\text{位速率} = 20 / \{2 \times (0 + 1) \times (3 + 4 + 3)\} = 1 \text{ Mbps}$$

表 11.3 BCR 的 TSEG1、TSEG2 可设定的范围

		TSEG2 (BCR[14~12])						
		001	010	011	100	101	110	111
TSEG1 (BCR[11~8])	0011	×	○	×	×	×	×	×
	0100	×	○	○	×	×	×	×
	0101	×	○	○	○	×	×	×
	0110	×	○	○	○	○	×	×
	0111	×	○	○	○	○	○	×
	1000	×	○	○	○	○	○	○
	1001	×	○	○	○	○	○	○
	1010	×	○	○	○	○	○	○
	1011	×	○	○	○	○	○	○
	1100	×	○	○	○	○	○	○
	1101	×	○	○	○	○	○	○
	1110	×	○	○	○	○	○	○
	1111	×	○	○	○	○	○	○

【注】* 波特率预分频 (BRP) 在 B'000000 (2×系统时钟) 以外时可以设定。

(3) 设定信箱发送接收方向

HCAN 中有 16 个信箱。信箱 0 是接收专用。信箱 1 ~ 15 用于发送或可设定为用于接收, 在初始状态为发送用。软件复位时, 信箱发送 / 接收方向的设定不被初始化。

要将信箱设定为发送用, 就要把信箱配置寄存器 (MBCR) 对应的位清零。要将信箱设定为接收用, 就要将 MBCR 的对应的位置为 1。为了提高接收效率要把高优先级的信息分配给信箱号码小的信箱。

(4) 信箱（信息控制域、信息数据域）的初始化

由于信息控制域、信息数据域内容在 RAM 中，加载电源后其初始值不确定。因此要初始化信箱内所有的值（写入 0 或者 1）。

(5) 设定发送信息方式

发送信息有以下 2 种方式。

- 按照信息标识符的优先顺序决定发送顺序
- 按照信箱号码的优先顺序决定发送顺序

通过主控制寄存器 (MCR) 的信息发送方式位 (MCR2) 选择发送方式。根据信息 ID 的优先顺序发送时，如果多个信息同时变为等待发送状态 (TXPR=1)，设定信息标识符，优先级最高的消息被保存在发送缓冲器。保存在缓冲器的消息通过 CAN 总线的仲裁获得发送权后，就被发送。每当 TXPR 被置位时，HCAN 寻找优先级别最高的信息，保存在发送缓冲器中。

按照信箱号码的优先顺序发送时，如果多个信息同时变为等待发送状态 (TXPR=1)，信箱号码最小的信息被保存在发送缓冲器中。保存在缓冲器中的信息通过 CAN 总线的仲裁获得发送权后，就被发送。

11.4.3 发送信息

使用信箱 1 ~ 15 发送信息。在进行初始设定后按照以下的顺序发送。图 11.9 所示的是发送信息的流程图。

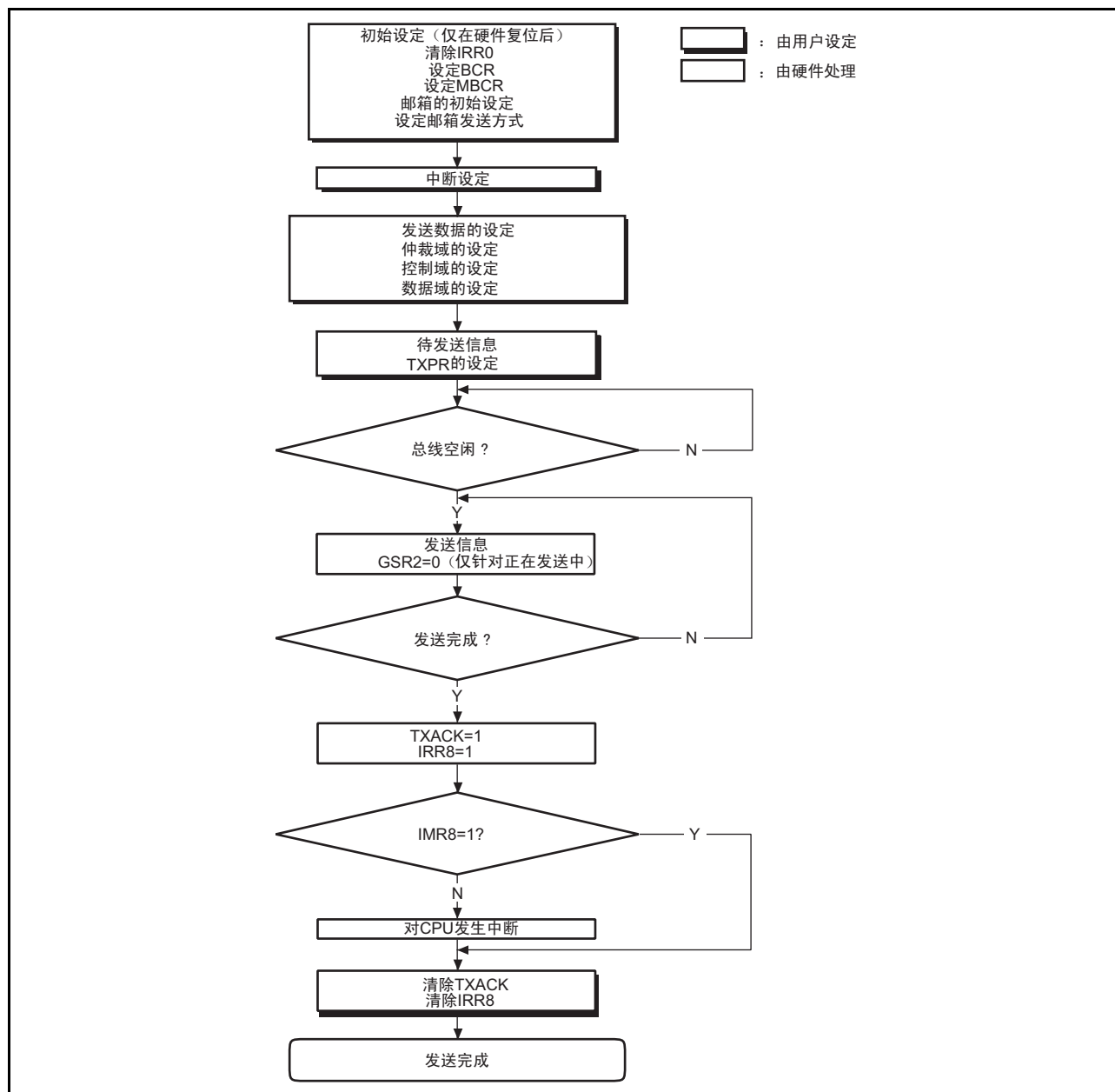


图 11.9 发送信息时的流程图

(1) 对 CPU 设定中断源

根据中断屏蔽寄存器 (IMR) 和信箱中断寄存器 (MBIMR) 的值, 设定 CPU 的中断源。通过 MBIMR, 可使每个信箱产生发送应答及取消发送应答的中断请求。

(2) 仲裁域的设置

通过发送信箱的信息控制域 MCx[5] ~ MC[8] 设定仲裁域。标准格式是设定 11 位的标识符 (ID 28 ~ ID-18) 和 RTR 位, 清零 IDE 位。在扩展位格式下, 设定 29 位的标识符 (ID-28 ~ ID-0) 和 RTR, 将 IDE 位置为 1。

(3) 控制域的设置

通过发送信箱的信息控制域 MCx[1] 设定, 在 0 ~ 8 字节范围内设定发送数据的字节数。

(4) 数据域的设置

将发送到 MDx[1] ~ [8] 中的信息数据设定为 0 ~ 8 字节范围。被发送的信息字节数根据控制域内的数据长度代码决定。即使设定了比在控制域设定的值更多的数据, 也只发送在控制域设定的字节数。

(5) 发送信息

设定信息控制域、信息数据域后, 如果将待发送寄存器 (TXPR) 对应信箱的等待发送位 (TXPR1 ~ 15) 置为 1, 则变为等待发送状态。如果为正常发送信息, 发送应答寄存器 (TXACK) 对应的应答位 (TXACK1 ~ 15) 被置位, 等待发送寄存器 (TXPR) 的等待发送位 (TXPR1 ~ 15) 会被自动清零。同时, 如果信箱中断屏蔽寄存器 (MBIMR) 对应的位 (MBIMR1 ~ 15) 和中断屏蔽寄存器 (IMR) 的信箱空中断 (IRR8) 被置为允许中断, 就可以发生对 CPU 的中断请求。

而在以下条件发送中断会自动再次发送信息。

- CAN 总线仲裁失败 (获得总线权失败)
- 发送中的错误 (位错误、stuff error、CRC 错误、帧错误、ACK 错误)

(6) 取消发送信息

可取消等待发送状态消息的发送。要取消等待发送的信息, 将取消待发送寄存器 (TXCR) 对应信箱的位 (TXCR1 ~ 15) 置为 1 (即使清除等待发送寄存器 (TXPR) 也不能取消发送)。一旦执行取消, 等待发送寄存器 (TXPR) 就会被自动清零, 而取消应答寄存器 (ABACK) 对应的位会被置为 1, 由此可以产生对 CPU 的中断请求。同时, 如果信箱中断屏蔽寄存器 (MBIMR) 对应的位 (MBIMR1 ~ 15) 和中断屏蔽寄存器 (IMR) 的信箱空中断 (IRR8) 被置为中断允许, 就可发生对 CPU 的中断请求。

但是，在以下条件的等待发送信息不能被取消。

- 内部仲裁及 CAN 总线仲裁中
- 发送数据帧、远程帧中

图 11.10 所示的是取消发送信息的流程图。

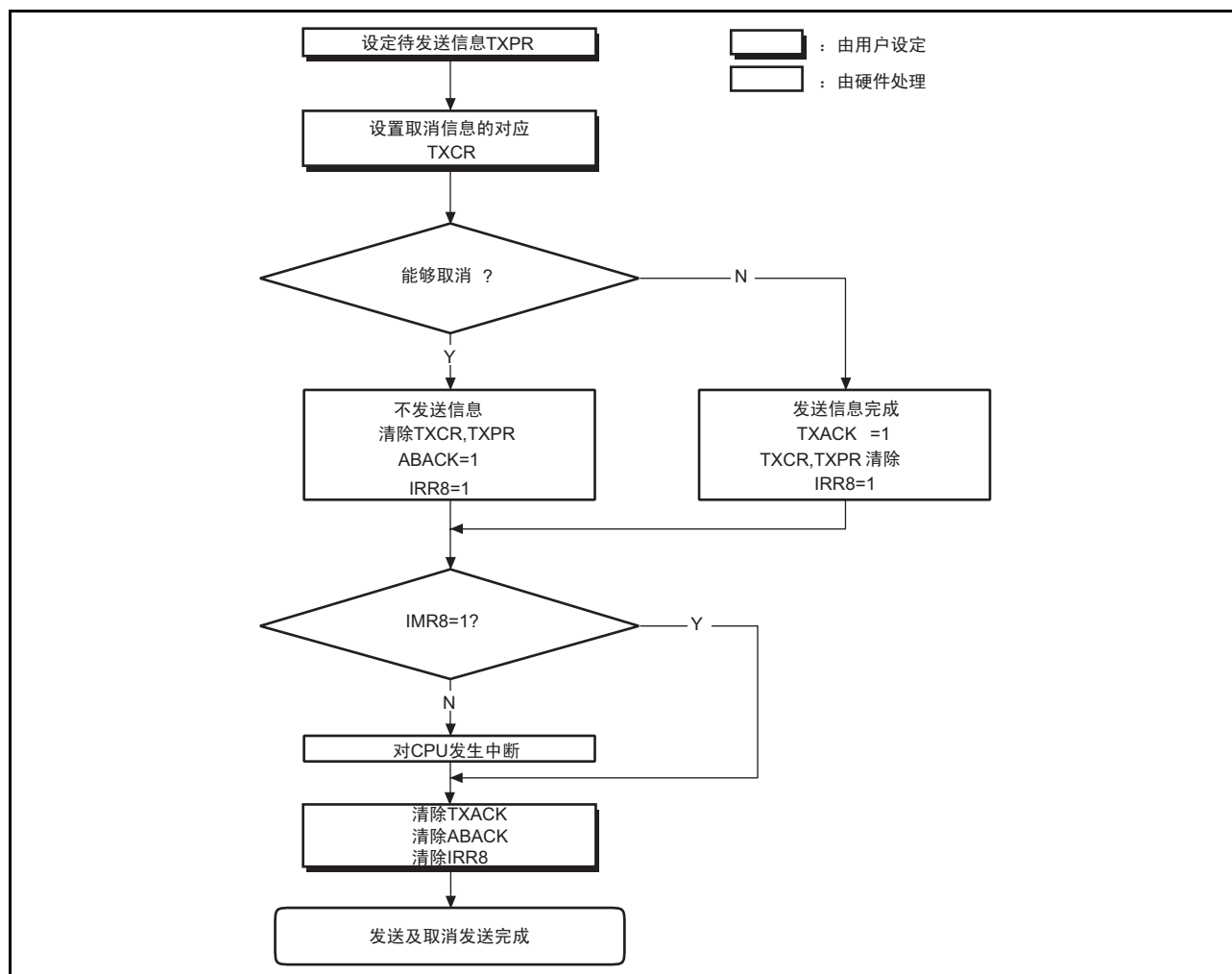


图 11.10 发送信息的取消的流程图

11.4.4 接收信息

在初始设定后按以下顺序进行接收。图 11.11 所示的是接收的流程图。

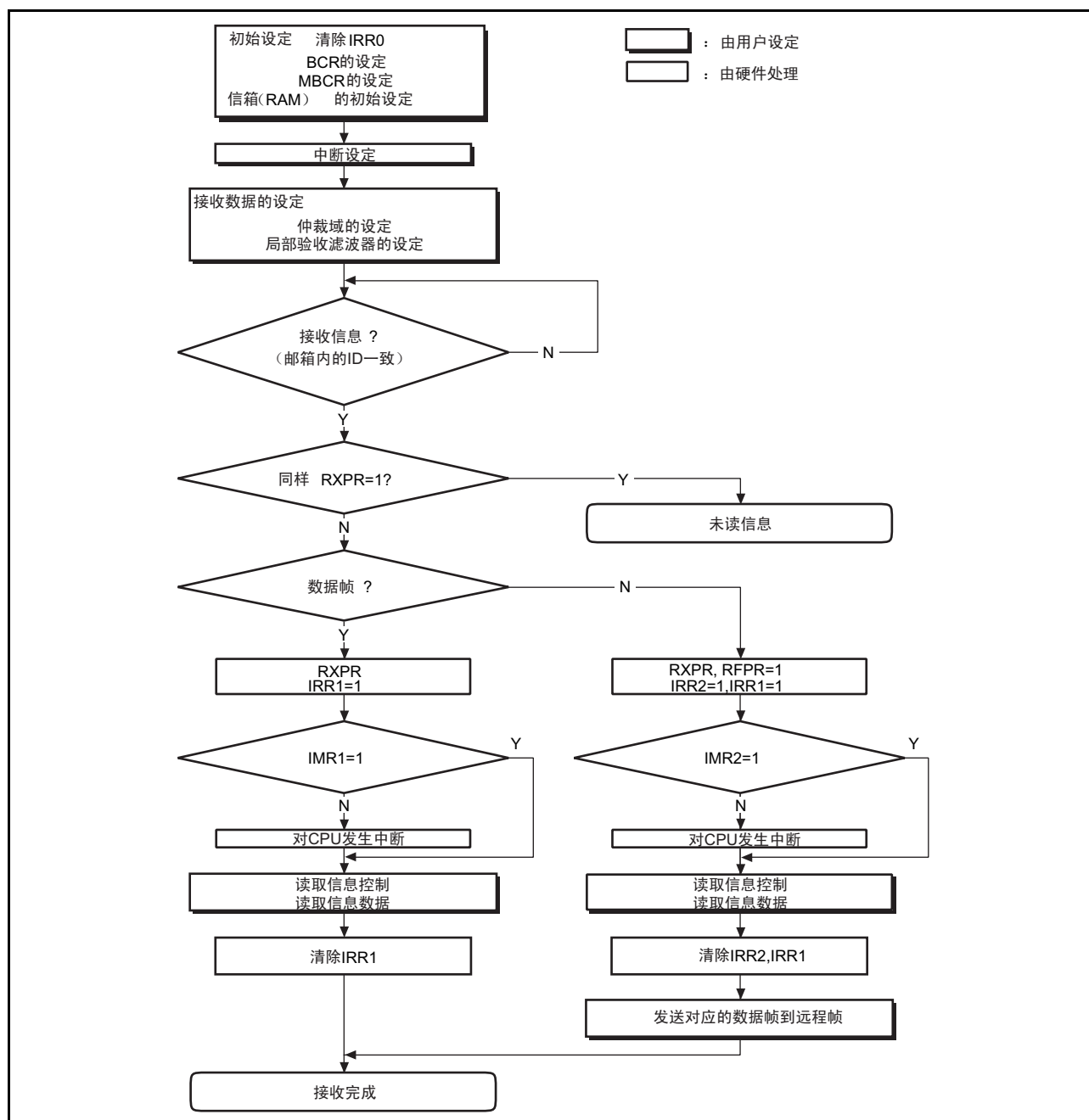


图 11.11 接收时的流程图

(1) 对 CPU 设定中断源

根据中断屏蔽寄存器 (IMR) 和信箱中断寄存器 (MBIMR) 设定中断源及接收信息的设定。通过 MBIMR, 可在每个信箱中产生等待接收数据帧、远程帧的中断请求。

(2) 仲裁域的设定

为接收信息, 必须提前在信箱的信息控制域 (MCx[1] ~ [8]) 域内设定信息标识符。接收到信息后, 将接收信息标识符和信息控制域内 I 标识符的所有位进行比较, 将信息保存在完全一致的信箱。但是, 信箱 0 配置有可设定无关位的局部验收滤波器掩码 local acceptance filter mask (LAFM)。只有在信箱 0 中设定 LAFM 才会有效, 对于接收标识符的所有位, 指定无关位就可接收多个标识符的信息。

[例]:

- 信箱 1 的标识符为 010_1010_1010 (标准格式) 时
信箱 1 可以接收的信息标识符只有以下 1 种。
标识符 Identifier 1: 010_1010_1010
- 信箱 0 的标识符为 010_1010_1010 (标准格式)
LAFM 的设定值为 000_0000_0011 (0: Care, 1: Don't Care) 时
信箱 0 可接收的信息标识符为以下 4 种
Identifier 1: 010_1010_1000
Identifier 2: 010_1010_1001
Identifier 3: 010_1010_1010
Identifier 4: 010_1010_1011

(3) 接收信息

接收信息后自动进行 CRC 检查。CRC 检查结果如果没有错误, 不管可不可以接收信息, 都在 ACK 域中发送 ACK。

- 接收数据帧
如果接收到的信息中没有检测到 CRC 等错误, 比较接收信息的标识符和接收用信箱内的标识符 (信箱 0 包含 LAFM), 如果完全一致, 则将信息保存在相应的信箱中。从信箱 0 开始比较标识符, 按照顺序依次一直到信箱 15。如果有完全一致的信箱, 在该时间结束比较, 并将信息保存在相应信箱中, 将接收完成寄存器 (RXPR) 对应的接收完成位 (RXPR0 ~ 15) 置位。但是, 与信箱 0 的 LAFM 相比较, 若标识符一致, 不结束标识符的比较, 继续从信箱 1 开始按顺序进行比较。这样就可以在别的信箱接收和信箱 0 相同的信息。信箱 1 ~ 15 中 2 个以上的信箱不能接收同一消息。如果接收信息, 根据信箱中断屏蔽寄存器 (MBIMR) 和中断屏蔽寄存器 (IMR) 的置位, 向 CPU 发出中断请求。
- 接收远程帧
信箱中可以保存数据帧和远程帧 2 种信息。远程帧与数据帧有 2 点不同, 分别为保存在信息控制域的远程传输请求位 (RTR) 的值不同, 数据域为 0 字节。在信息控制域的数据长度码 (DLC) 中, 保存数据帧应该返回的数据长度。
如果接收远程帧 (RTR=recessive), 要将等待远程请求寄存器 (RFPR) 对应的位置位。此时, 通过设定信箱中断屏蔽寄存器 (MBIMR) 对应的位 (MBIMR0 ~ 15) 和中断屏蔽寄存器 (IMR) 远程帧请求中断屏蔽 (IRR2), 向 CPU 发出中断请求。

(4) 未读信息的覆盖写入

如果接收的信息和信箱内的标识符一致，无论有无未读信息，都会被保存在信箱。如果发生对未读信息的覆盖写入，未读信息寄存器 (UMSR) 对应的位 (UMSR0 ~ 15) 将被置位。在接收完成寄存器 (RXPR) 的位没有被清零的状态下，接收新信息时，(UMSR) 被置位。这时，通过设定中断屏蔽寄存器 (IMR) 的未读中断标记 (IRR9)，向 CPU 发出中断请求。图 11.12 所示的是未读信息覆盖写入的流程图

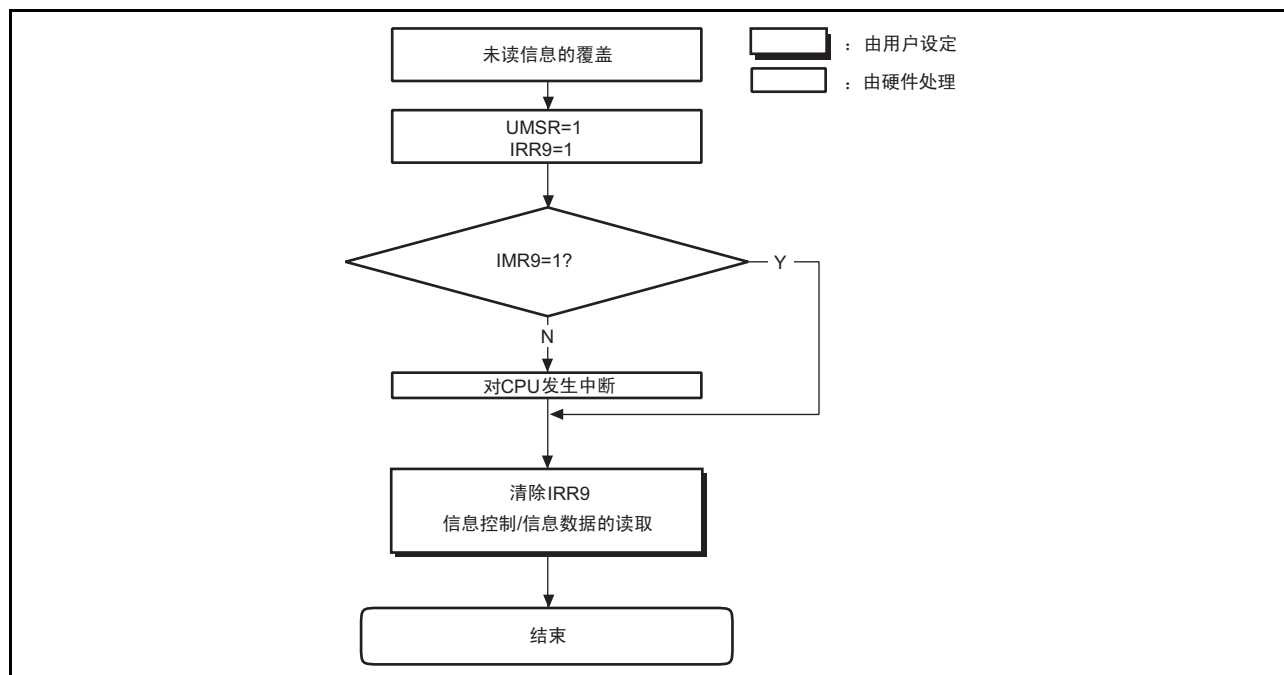


图 11.12 未读信息覆盖写入的流程图

11.4.5 HCAN 休眠模式

HCAN 模块中，处于休眠状态，具有降低功耗的功能。图 11.13 所示的是 HCAN 休眠模式的流程图。

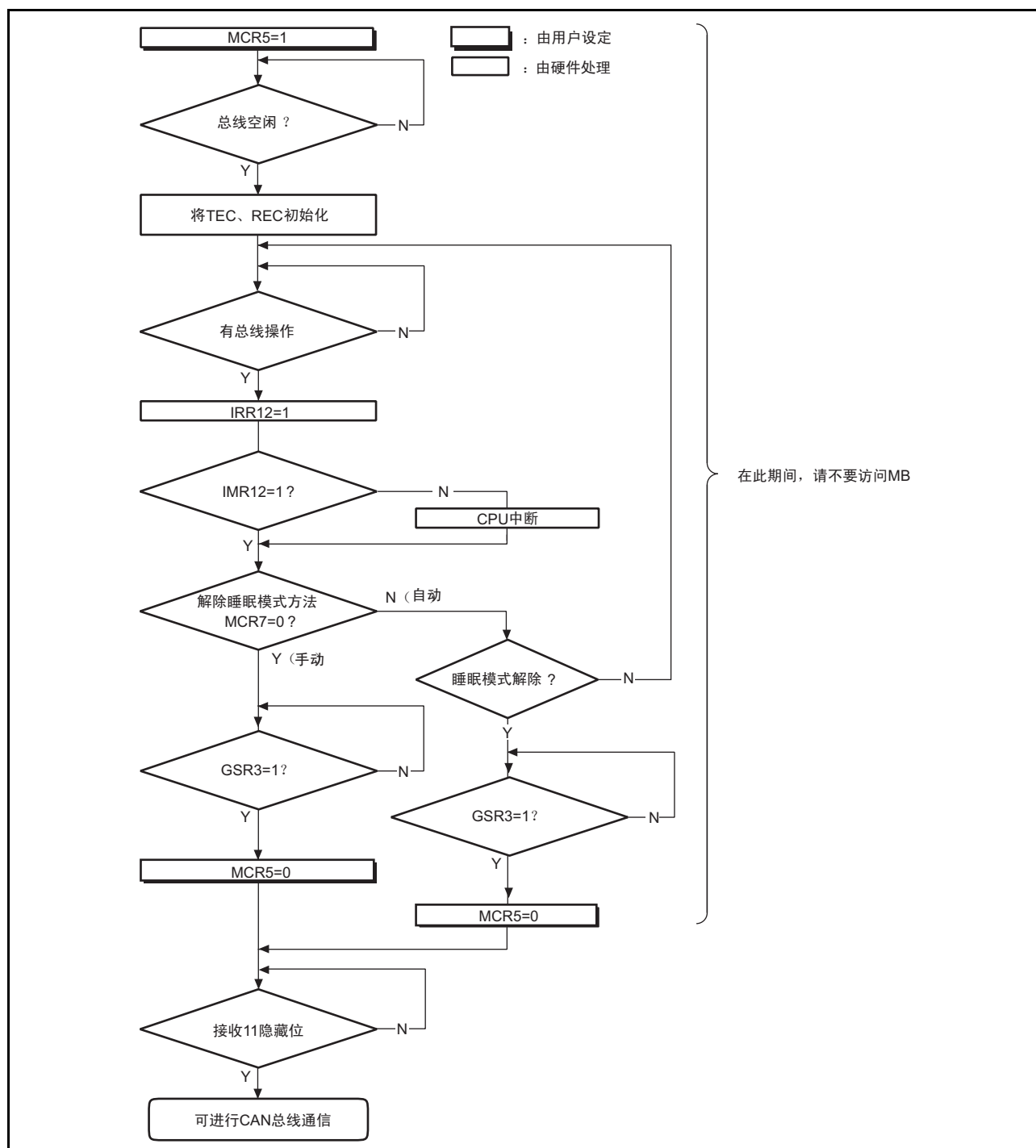


图 11.13 HCAN 休眠模式的流程图

要进入到 HCAN 休眠模式，将主控制寄存器 (MCR) 的 HCAN 休眠模式位 (MCR5) 置为 1。在 CAN 总线运行时，HCAN 等到总线变为空闲状态，将进入到 HCAN 休眠模式。

HCAN 休眠模式可用以下 2 种方法解除。

- 通过软件解除
- 通过 CAN 总线运行解除

从 HCAN 休眠模式再次变为可以进行 CAN 总线通信的状态，模式解除后必须要接收 11 隐性位。

(1) 通过软件解除

用软件解除是从 CPU 向 MCR5 写入 0。

(2) 通过 CAN 总线运行解除

可通过 MCR 的 MCR7 进行解除方法的选择。通过 CAN 总线进行解除时，使 CAN 总线运行，一旦检测出变化就自动解除。这时第 1 个信息不接收到信箱中，从下一个信息开始正常接收。检测到从 CAN 总线变为 HCAN 休眠模式时，中断寄存器 (IRR) 的总线运行中断标记 (IRR12) 就会被置位。同时，如果中断屏蔽寄存器 (IMR) 的总线运行中断屏蔽 (IMR12) 被置为中断允许，就可以向 CPU 发出中断请求。

11.4.6 HCAN HALT 模式

HCAN HALT 模式是不进行 HCAN 的硬件复位、软件复位，而改变信箱设定的模式。图 11.14 所示的是 HCAN HALT 模式的流程图。

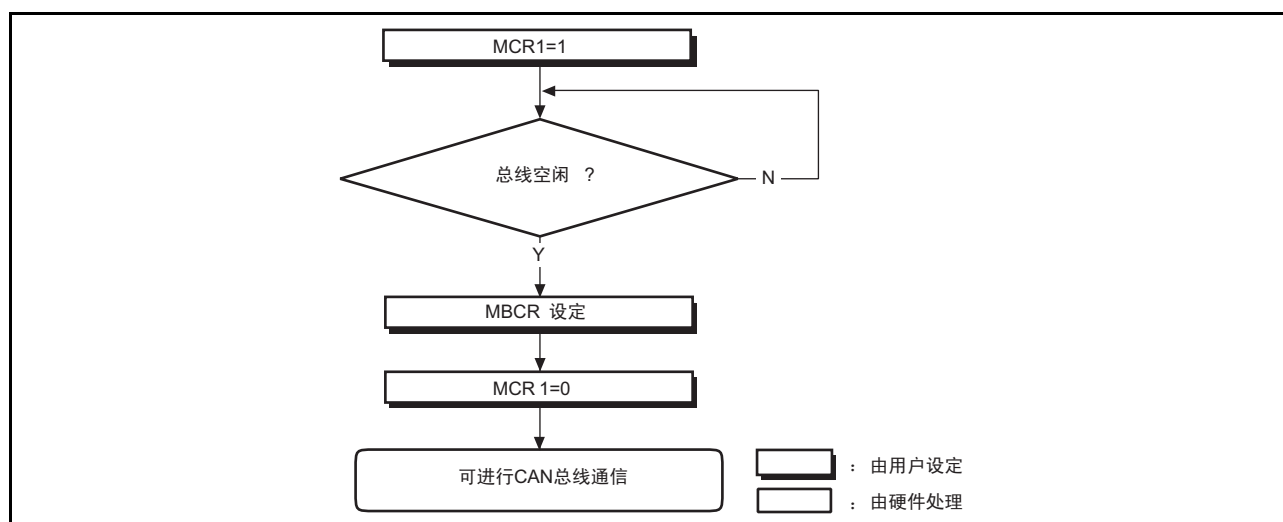


图 11.14 HCAN HALT 模式的流程图

将主控制寄存器 (MCR) 的 HALT 请求位 (MCR1) 置为 1，就会进入到 HCAN HALT 模式。但是，当 CAN 总线正在运行时，一直等到总线变为空闲状态后才会进入到 HCAN HALT 模式。

将 MCR1 清除，就可解除 HCAN HALT 模式。

11.5 中断源

HCAN 中有如表 11.4 所示的中断源。这些中断源除复位中断（IRR0）外，都可通过加电复位屏蔽。屏蔽要使用信箱中断屏蔽寄存器（MBIMR）及中断屏蔽寄存器（IMR）。关于各中断请求的中断向量请参考「第 5 章 中断控制器」。

表 11.4 HCAN 的中断源

名称	简称	最小值
ERS0/OVR0	错误无源中断（ $TEC \geq 128$ 或者 $REC \geq 128$ ）	IRR5
	总线关闭中断（ $TEC \geq 256$ ）	IRR6
	由于加电复位的复位处理中断	IRR0
	接收远程帧	IRR2
	错误警告中断（ $TEC \geq 96$ ）	IRR3
	错误警告中断（ $REC \geq 96$ ）	IRR4
	发送过载帧	IRR7
	未读信息的覆盖写入	IRR9
	HCAN 休眠中检测出 CAN 总线运行	IRR12
RM0	接收信箱 0 的信息	IRR1
RM1	接收信箱 1 ~ 15 的信息	IRR1
SLE0	发送信息或取消发送	IRR8

11.6 CAN 总线接口

为连接本 LSI 和 CAN 总线，需要总线收发器 IC。关于收发器 IC，推荐使用 Philips 公司的 PCA82C250 设备。使用 PCA82C250 以外的产品时，请使用和 PCA82C250 兼容的产品。图 11.15 所示的是连接例子。

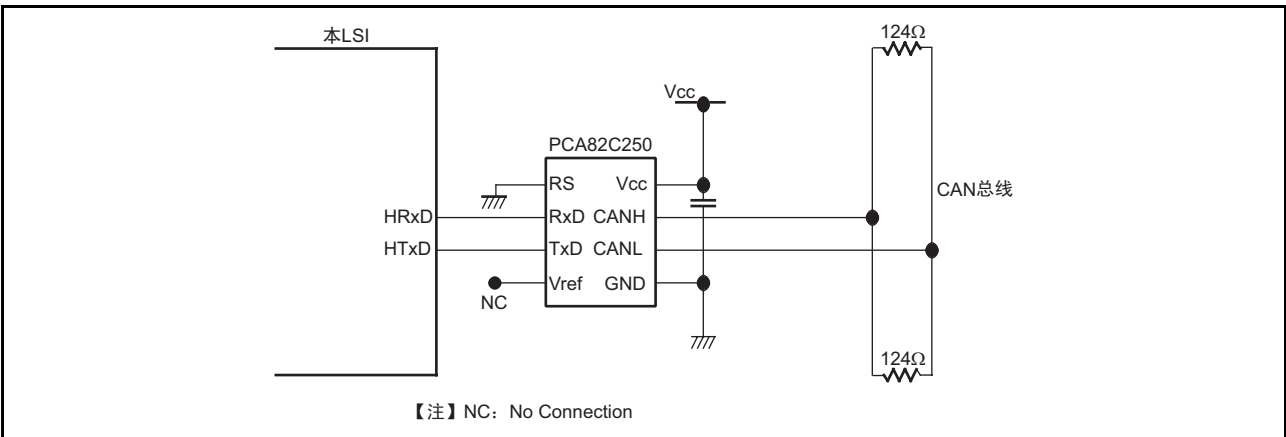


图 11.15 使用 PCA82C250 的高速接口

11.7 使用时的注意事项

11.7.1 模块待机模式的设定

通过模块待机控制寄存器，可设定 HCAN 的运行禁止 / 允许。如果是初始值，HCAN 将停止运行。解除模块待机模式可存取该寄存器。详细请参考 [第 20 章 低功耗状态]。

11.7.2 复位

通过加电复位、硬件待机、软件待机，可以使 HCAN 复位，这时寄存器虽然全部被初始化，但是信箱（信息控制域 (MCx[x]) / 信息数据域 (MDx[x])）并没有被初始化。加载电源后信箱（信息控制域 (MCx[x]) / 信息数据域 (MDx[x])）未初始化，其值不确定，因此，加电复位、硬件待机、软件待机后，必须要将信箱初始化。加电复位及软件待机恢复后，必须将复位中断标记 (IRR0) 置位，该位在中断屏蔽寄存器 (IMR) 中不能被屏蔽，如果不删除标记，在中断控制器中设定为允许 HCAN 中断，就会直接进入 HCAN 中断，因此在初始化时要清零 IRR0。

11.7.3 HCAN 休眠模式

根据 HCAN 休眠模式的 CAN 总线运行状态，可设定中断寄存器 (IRR) 的总线运行中断标记 (IRR12)。因此，HCAN 不解除 HCAN 休眠模式的标记。同时，通用状态寄存器 (GSR) 的复位状态位 (GSR3) 在 HCAN 休眠模式中也置位。

11.7.4 中断

设定信箱中断屏蔽寄存器 (MBIMR) 后，就已经设定了信箱的接收完成、发送完成、取消发送，所以中断寄存器 (IRR8、2、1) 不会被置位。

11.7.5 错误计数器

在 error active、error passive，REC、TEC 累加计数、递减计数、总线关闭中使用 REC 计算 11 隐性位 (REC+1)。如果 REC=96，IRR4 和 GSR1 就会被置位。

11.7.6 存取寄存器

HCAN 的所有寄存器只能以字节和字为单位进行存取，不能存取长字。

11.7.7 HCAN 的中速模式

在中速模式下不能读写 HCAN 的寄存器。

11.7.8 待机时的寄存器保持

HCAN 在硬件待机及软件待机时，所有的寄存器都被初始化。

11.7.9 位操作命令的使用

写入 1，HCAN 的状态标记就会被清零，所以不要清除使用位操作命令的标记。清零标记时使用 MOV 命令，只有清零的位可以写入 1。

11.7.10 关于 HCAN 的 TXCR 运行

1. 使用取消等待发送寄存器 (TXCR) 取消等待发送信箱的信息时, 不管发送是否被取消, 在以下条件下可能会出现 TXCR 及待发送寄存器 (TXPR) 对应的位不被清除的情况。
 - 由于 CAN 总线错误, HRxD 管脚 stack 1
 - 等待发送 (或发送中) 的信箱为一个以上
 - 通过 TXCR 取消正在发送的信箱的信息发生此现象时, 虽然发送被取消, 但是 TXPR 和 TXCR 的状态还是持续显示 " 信息取消中 " 的错误状态, 解除 HRxD 管脚的 stack 1, CAN 总线即使恢复到正常状态也不能再次开始发送。发送信息为 2 条以上时, 不正在发送的信息会被取消, 正在发送的信息还是保持原来状态。
为了避免该问题, 请实施以下对策。
 - 不要通过 TXCR 取消发送。CAN 总线恢复后, 正常发送完成, TXPR 被清除, HCAN 会恢复到正常运行。
 - 需要取消发送时, TXCR 对应的位变为 0 前继续在相应的位写入 1。TXPR 和 TXCR 被清除, HCAN 恢复到正常运行。
2. 迁移到总线关闭时, 需设定 TXPR, 变成等待发送状态, 总线关闭中即使设定 TXCR, 内部的状态机也不运行, 所以也不能取消, 总线关闭恢复后, 通过发送 1 条信息或发送错误, 就可以取消发送。总线关闭恢复后的信息清除请实施以下对策。
 - 总线关闭期间通过复位 HCAN 模块清除等待发送的信息。通过设定 / 解除模块停止位 (MSTPCRC 的 MSTPC3) 进行 HCAN 模块的复位。这时, HCAN 内部全部被复位, 所以要再次进行初始设定。

11.7.11 关于 HCAN 发送过程

从总线空闲到发送设定后，如果 50μs 以内通过以下条件进行下一个信息的发送设定或发送取消，有时会破坏之前设定的发送信息标识符。

- 比设定为第 1 次发送的信息优先级高的信息，设定为第 2 次发送时。
- 设定的第 1 次发送中，取消优先级最高的信息时。

为了不破坏信息标识符，请进行以下设定。

- 通过 1 次 TXPR 进行发送设定，发送完所有信息后，再次进行发送设定（全部一起设定发送），其间隔为 50μs 以上。
- 按照发送信息的优先顺序设定发送。
- TXPR 和 TXPR 的设定时间间隔，或 TXPR 和 TXCR 的设定时间间隔保持 50μs 以上。

表 11.5 TXPR 和 TXPR 设定时间或 TXPR 和 TXCR 设定时间的间隔限制

波特率 (bps)	设定间隔 (μs)
1M	50
500K	50
250K	50

11.7.12 HCAN 软件复位及 HCAN 休眠解除

要解除 HCAN 软件复位或 HCAN 休眠时（MCR0=0 或者 MCR5=0），确认 GSR3（复位状态位）为 1 后再解除。

11.7.13 HCAN 休眠中的信箱存取

HCAN 休眠中不要存取信箱。如果在 HCAN 休眠中存取信箱，有时会出现 CPU 停止。

在 HCAN 休眠中存取寄存器，CPU 不停止。而且，在 HCAN 休眠以外的模式，存取信箱，CPU 也不停止。

第 12 章 A/D 转换器

本 LSI 内置逐次逼近型 10 位 A/D 转换器，包括 8 个可选模拟输入通道。A/D 转换器的框图如图 12.1 所示。

12.1 特点

- 分辨率：10 位
- 输入通道：最大 8 通道（HD64F2280RB 是 6 通道）
- 转换时间：平均 1 通道 13.3 μ s（20MHz 运行时）
- 运行模式：2 种
单通道模式：1 个通道的 A/D 转换
扫描模式：1 ~ 4 通道的连续 A/D 转换
- 数据寄存器：4 个
每个通道的 A/D 转换结果保存于各通道对应的一个 16 位数据寄存器
- 采样保持功能
- 开始 A/D 转换的方法：3 种
软件
16 位定时脉冲单元（TPU）触发开始转换
外部触发器信号
- 中断源
能产生 A/D 转换完成中断请求（ADI）
- 可设定模块待机模式

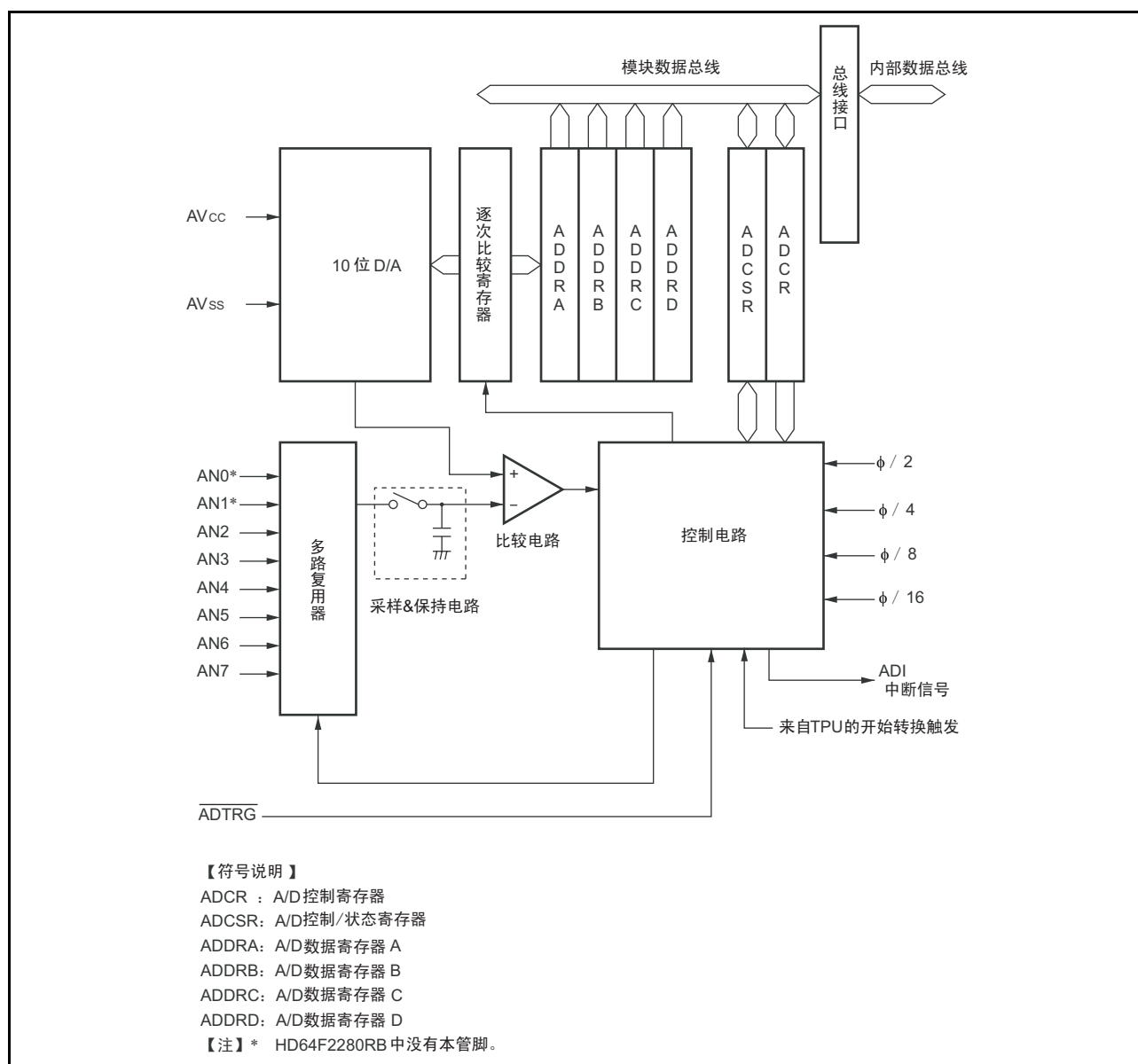


图 12.1 A/D 转换器的框图

12.2 输入 / 输出管脚

A/D 转换器中使用的管脚如表 12.1 所示。8 个模拟输入管脚被分为 4 通道 ×2 组。模拟输入管脚 0 ~ 3(AN0 ~ AN3) 是组 0，模拟输入管脚 4 ~ 7(AN4 ~ AN7) 是组 1。AVCC、AVSS 管脚是 A/D 转换器内模拟部分的电源

表 12.1 管脚构成

管脚名称	简称	输入 / 输出	功能
模拟电源管脚	AVcc	输入	模拟部分的电源管脚及基准电压
模拟接地管脚	AVss	输入	模拟部分的地及基准电压
模拟输入管脚 0*	AN0	输入	组 0 的模拟输入管脚
模拟输入管脚 1*	AN1	输入	
模拟输入管脚 2	AN2	输入	
模拟输入管脚 3	AN3	输入	
模拟输入管脚 4	AN4	输入	组 1 的模拟输入管脚
模拟输入管脚 5	AN5	输入	
模拟输入管脚 6	AN6	输入	
模拟输入管脚 7	AN7	输入	
A/D 外部触发器输入管脚	ADTRG	输入	用于开始 A/D 转换的外部触发器管脚

【注】 *HD64F2280RB 中没有。

12.3 寄存器说明

A/D 转换器中有以下寄存器。通过模块控制寄存器 A(MSTPCRA) 的 MSTPA1 位设定模块待机模式。关于 MSTPCRA, 请参考「20.1.3 模块待机控制寄存器 A ~ D (MSTPCRA ~ MSTPCRD)」。

- A/D 数据寄存器 A(ADDRA)
- A/D 数据寄存器 B(ADDRB)
- A/D 数据寄存器 C(ADDRC)
- A/D 数据寄存器 D(ADDRD)
- A/D 控制 / 状态寄存器 (ADCSR)
- A/D 控制寄存器 (ADCR)

12.3.1 A/D 数据寄存器 A ~ D (ADDRA ~ ADDRD)

ADDR 是保存 A/D 转换结果的 16 位只读专用寄存器, 有 ADDRA ~ ADDRD 4 个。保存各模拟输入通道转换结果的 ADDR 如表 12.2。

10 位转换数据被保存在 ADDR 的位 6 到 15。低 6 位的读取值总是为 0。

CPU 与 A/D 转换器之间由 8 位数据总线连接。CPU 可直接读取高位字节, 而低位字节通过临时寄存器读取。高位字节数据读取后, ADDR 传输数据给临时寄存器。读取 ADDR 时, 必须以字为单位读取, 或按照先高位字节, 后低位字节的顺序读取。只读取低位字节时不能保证读取内容。

表 12.2 模拟输入通道和 ADDR 的对应

模拟输入通道		保存转换结果的 A/D 数据寄存器
组 0 (CH2=0)	组 1 (CH2=1)	
AN0*	AN4	ADDRA
AN1*	AN5	ADDRB
AN2	AN6	ADDRC
AN3	AN7	ADDRD

【注】 *HD64F2280RB 中没有本管脚。

12.3.2 A/D 控制 / 状态寄存器 (ADCSR)

ADCSR 用于控制 A/D 转换的运行。

位	位名	初始值	R/W	说明
7	ADF	0	R/(W)*1	A/D 结束标志 表示 A/D 转换结束的状态标志。 [置位条件] 在单通道模式, A/D 转换结束时 • 扫描模式下, 可选择的所有通道的 A/D 转换结束时 [清除条件] 读取到 1 的状态后, 写入 0 时
6	ADIE	0	R/W	A/D 中断允许 如果该位置 1, 就允许 ADF 产生 ADI 中断请求。
5	ADST	0	R/W	A/D 开始 该位置 1, 开始 A/D 转换, 清零后, 中止 A/D 转换, 变为待机状态。在单通道模式下, 所选择通道的 A/D 转化结束后, 该位会自动清零。扫描模式下, 该位在被软件复位、软件待机、硬件待机或模块待机模式清除前, 依次将被选择的通道进行连续转换。
4	SCAN	0	R/W	扫描模式 选择 A/D 转换的运行模式 0: 单通道模式 1: 扫描模式
3	—	0	R/W	保留位 写入值常为 0。
2 1 0	CH2 CH1 CH0	0 0 0	R/W R/W R/W	通道选择 2 ~ 0 选择模拟输入通道。 SCAN=0 时 000: AN0*2 001: AN1*2 010: AN2 011: AN3 100: AN4 101: AN5 110: AN6 111: AN7 SCAN=1 时 000: AN0*2 001: AN0、AN1*2 010: AN0 ~ AN2*2 011: AN0 ~ AN3*2 100: AN4 101: AN4 ~ AN5 110: AN4 ~ AN6 111: AN4 ~ AN7

【注】 *1 为清除标志只能写入 0。

*2 HD64F2280RB 中没有 AN0、AN1。使用中需要注意。

SCAN=1 时, 进行了 010 或 011 设定, 被保存在 ADDRA、ADDRB 中的转换数据变为不定。

12.3.3 A/D 控制寄存器 (ADCR)

ADCR 允许通过外部触发器开始 A/D 转换。

位	位名	初始值	R/W	说明
7 6	TRGS1 TRGS0	0 0	R/W R/W	定时触发选择 1、0 通过外部触发信号，开始 A/D 转换。在 A/D 转换停止时 (ADST=0) 置位。 00: 根据软件，开始 A/D 转换 01: 根据 TPU 的输出，开始 A/D 转换 10: 禁止设定 11: 根据 ADTRG，开始 A/D 转换
5、4	—	全为 0	—	保留位 该位读取值常为 1。
3 2	CKS1 CKS0	0 0	R/W R/W	时钟选择 1、0 设定 A/D 转换时间。在 A/D 转换停止时 (ADST=0)，切换 A/D 转换时间。A/D 转换时间必须在「第 22 章 电特性」的表 22.7 所示的范围内设定。 00: 530 状态 (max) 01: 266 状态 (max) 10: 134 状态 (max) 11: 68 状态 (max)
1、0	—	全为 1	—	保留位 该位读取值常为 1。

12.4 运行说明

A/D 转换器采用逐次逼近方式，其分辨率是 10 位。其运行模式有单通道模式和扫描模式。切换运行模式或模拟输入通道时，为了避免发生误操作，先将 ADCSR 的 ADST 位清零；在变换运行模式或模拟输入通道的同时，可对 ADST 位进行置位。

12.4.1 单通道模式

单通道模式是将指定的 1 个通道的模拟输入进行 1 次如下的 A/D 转换。

1. 如果通过软件或外部触发器输入，将 ADCSR 的 ADST 位置 1，被选择的通道开始 A/D 转换。
2. A/D 转换结束后，A/D 转换结果被传送到该通道对应的 A/D 数据寄存器。
3. A/D 转换结束后，ADCSR 的 ADF 位被置为 1。此时，如果 ADIE 位被置为 1，则会产生 ADI 中断请求。
4. ADST 位在 A/D 转换过程中保持为 1，并且在转换结束后自动被清除，A/D 转换器变为待机状态。A/D 转换中如果将 ADST 位清零，就会中止转换，A/D 转换器变为待机状态。

12.4.2 扫描模式

扫描模式是将指定的最多 4 个通道的模拟输入按照如下操作连续进行 A/D 转换。

- 1. 如果通过软件、TPU 或外部触发器输入，将 ADCSR 的 ADST 位置 1，就从组里的第 1 个通道 (CH2=0 时为 AN0、CH2=1 时为 AN4) 开始 A/D 转换。
- 2. 各通道的 A/D 转换结束后，A/D 转换结果被依次传送到每个通道对应的 A/D 寄存器。
- 3. 如果选择的所有通道的 A/D 转换均结束，ADCSR 的 ADF 位就被置 1。这时如果 ADIE 位被置为 1，就会产生 ADI 中断请求。A/D 转换器会再次从组里的第一通道开始 A/D 转换。
- 4. ADST 位保持为 1 时 A/D 转换器重复 2. ~ 3. 的操作。如果将 ADST 位清零，将会中止 A/D 转换，A/D 转换器变为待机状态。

12.4.3 输入采样和 A/D 转换时间

A/D 转换器中内置采样保持电路。将 ADCSR 的 ADST 位置为 1 后，经过 A/D 转换开始延迟时间 (t_D)，A/D 转换器对输入进行采样，之后才开始转换。A/D 转换的时序图如图 12.2 所示，A/D 转换时间如表 12.3 所示。

如图 12.2 所示，A/D 转换时间 (t_{CONV}) 包含 t_D 和输入采样时间 (t_{SPL})。这里的 t_D 由 ADCSR 的写入时序决定，不是固定值。因此转换时间在如表 12.3 所示的范围内变化。

扫描模式的转换时间是以表 12.3 所示的值为第 1 次的转换时间，第 2 次以后的转换时间则为表 12.4 所示的值。任何情况下都要在第 22 章 电特性的表 22.7 所示的范围设定 ADCR 的 CKS1、CKS0 位。

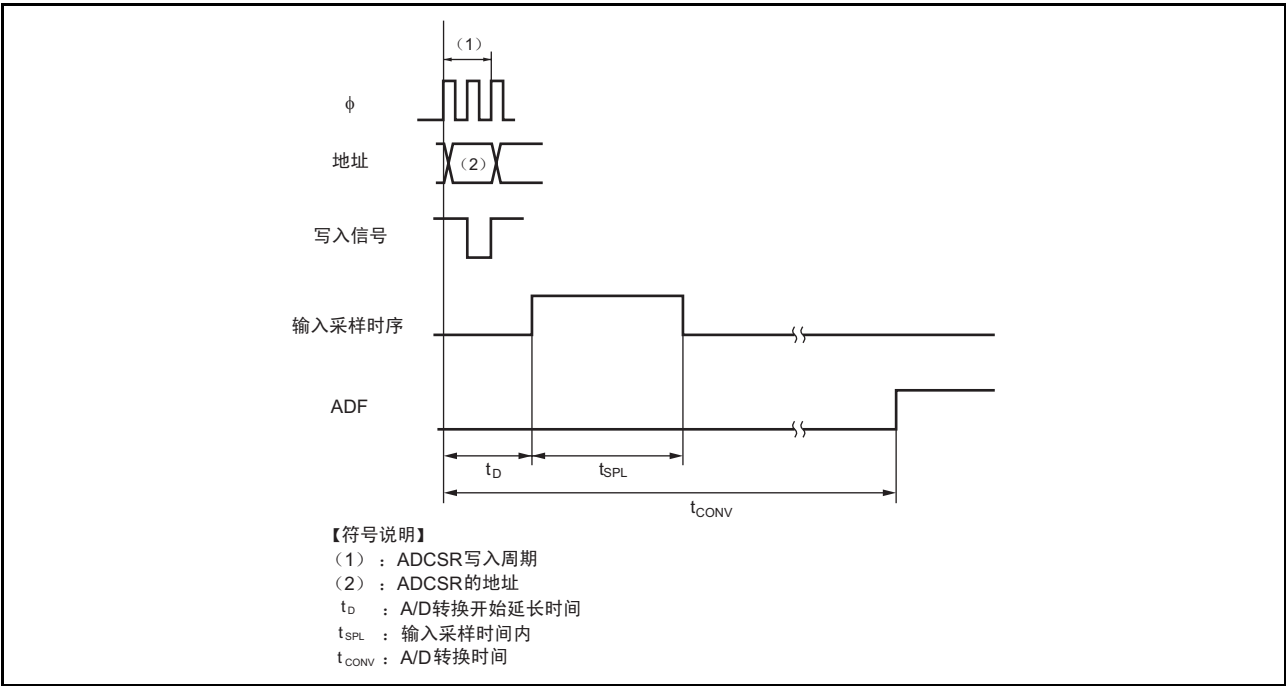


图 12.2 A/D 转换时序

表 12.3 A/D 转换时间（单模式）

项目	记号	CKS1=0						CKS1=1					
		CKS1=0			CKS1=1			CKS1=0			CKS1=1		
		min	typ	max	min	typ	max	min	typ	max	min	typ	max
A/D 转换开始延迟时间	tD	18	—	33	10	—	17	6	—	9	4	—	5
输入采样时间	tSPL	—	127	—	—	63	—	—	31	—	—	15	—
A/D 转换时间	tCONV	515	—	530	259	—	266	131	—	134	67	—	68

【注】表中数值单位是状态。

表 12.4 A/D 转换时间（扫描模式）

CKS1	CKS0	转换时间（state）
0	0	512（固定）
	1	256（固定）
1	0	128（固定）
	1	64（固定）

12.4.4 外部触发器输入时序

可以通过外部触发器的输入，开始 A/D 转换。ADCR 的 TRGS1、TRGS0 位被置为 11 时，会从 $\overline{\text{ADTRG}}$ 管脚引入外部触发信号。在 $\overline{\text{ADTRG}}$ 下降沿时，ADCSR 的 ADST 位置为 1，开始 A/D 转换。无论是在单通道模式还是扫描模式，其他的运行都与通过软件将 ADST 位置为 1 的运行情况相同。此时序如图 12.3 所示。

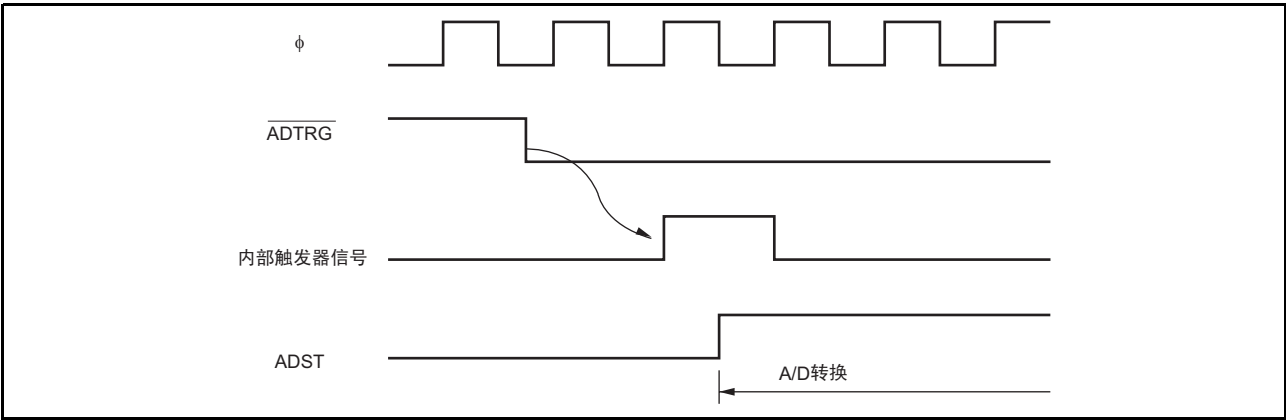


图 12.3 外部触发器输入时序

12.5 中断源

在 A/D 转换结束后，A/D 转换器会产生 A/D 转换结束中断（ADI）请求。在 A/D 转换结束后，ADCSR 的 ADF 会被置 1，这时如果 ADIE 置为 1，ADI 中断请求就会被允许。

表 12.5 A/D 转换器的中断源

名称	中断源	中断标记
ADI	A/D 转换结束	ADF

12.6 A/D 转换精度的定义

本 LSI 的 A/D 转换精度定义如下。

- 分辨率
A/D 转换器的数字输出码位数
- 量化误差
它是 A/D 转换器的固有偏差，为 1/2 LSB（图 12.4）
- 偏移误差
数字输出从最小电压值 B'0000000000（H'000）变为 B'0000000001（H'001）时，模拟输入电压值与理想 A/D 转换特性的偏差（图 12.5）
- 满刻度 Full-scale 误差
数字输出从 B'1111111110（H'3FE）变为 B'1111111111（H'3FF）时，模拟输入电压值与理想 A/D 转换特性的偏差（图 12.5）
- 非线性误差
从零电压到满刻度电压之间的理想 A/D 转换特性的误差。不包含偏移误差、满刻度误差、量化误差（图 12.5）。
- 绝对精度
数字值和模拟输入值之间的误差。包含偏移误差、满刻度误差、量化误差及非线性误差。

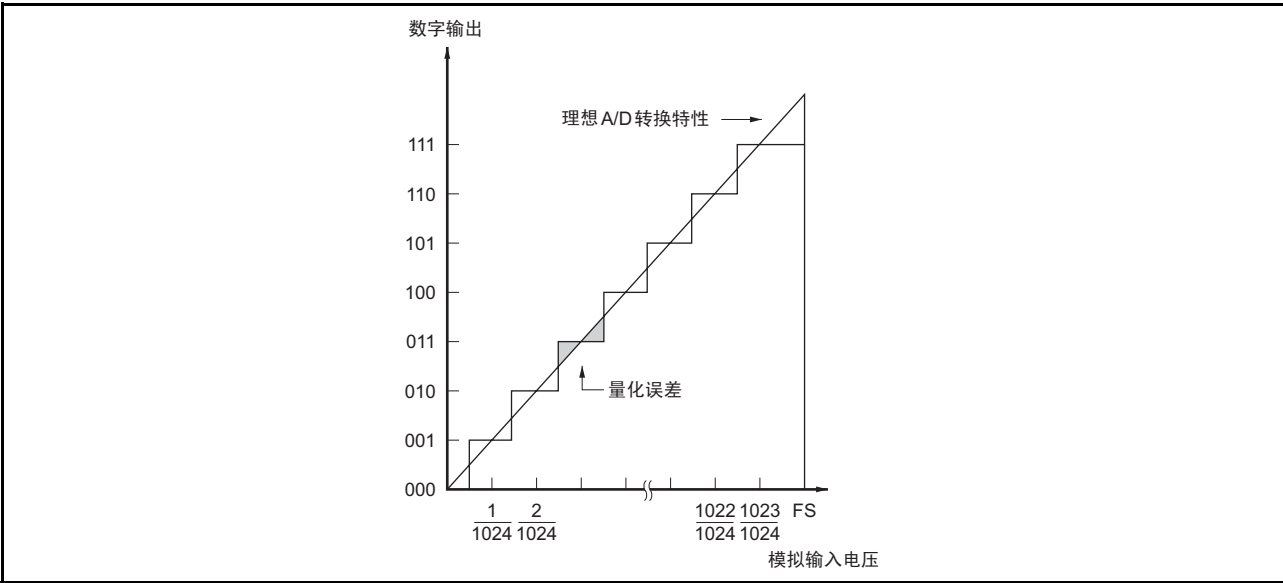


图 12.4 A/D 转换精度定义

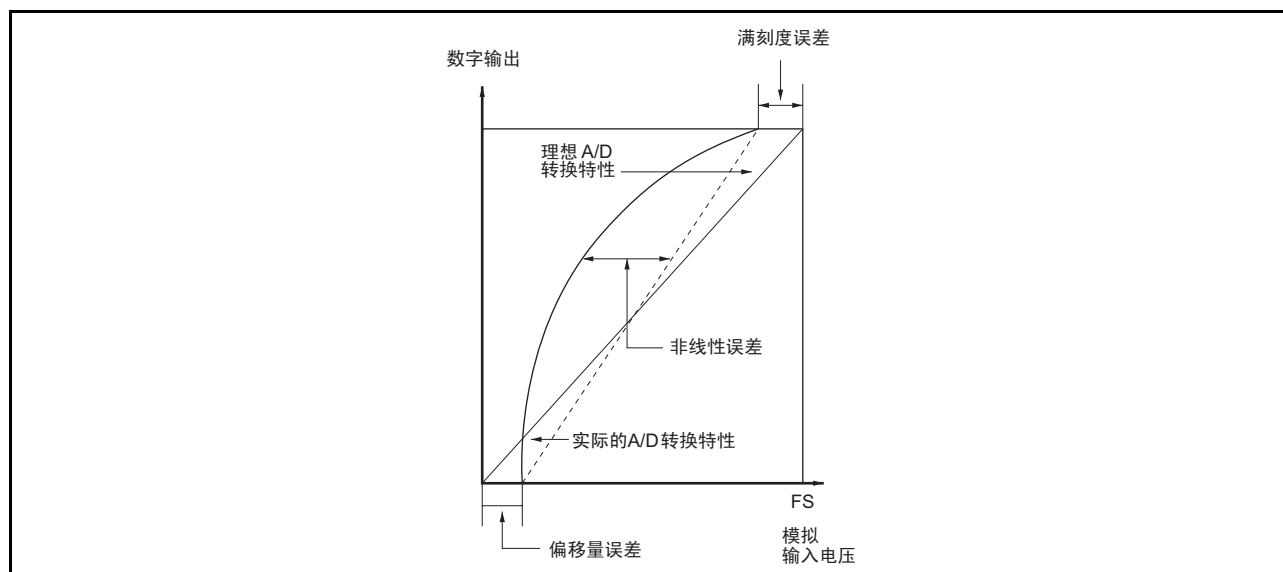


图 12.5 A/D 转换精度的定义

12.7 使用时的注意事项

12.7.1 模块待机模式的设定

可通过模块待机控制寄存器设定 A/D 转换器的禁止 / 允许。初始值状态，A/D 转换器为停止运行模式。解除模块停止模式，就可存取寄存器。详细请参考「第 20 章 低功耗状态」。

12.7.2 关于容许信号源阻抗

针对信号源阻抗为 $10\text{k}\Omega$ 以下的输入信号，本 LSI 的模拟输入端设计可保证其转换精度。这是在采样时间内，对 A/D 转换器的采样保持电路的输入电容进行充电所制定的规格，传感器的输出阻抗超过 $10\text{k}\Omega$ 时，会出现充电不足，而不能保证 A/D 转换精度的情况。在单通道模式进行转换时，如果外部配置一个大容量电容，输入负载实质上只是 $10\text{k}\Omega$ 的内部输入电阻，因此可忽略信号源的阻抗，但是，在此情况下，会形成低通滤波器，有时会无法跟踪微分系数大的模拟输入信号（例如 $5\text{mV}/\mu\text{s}$ 以上）（图 12.6）。在转换高速模拟信号或在扫描模式进行转换时，必须插入一个低阻抗的缓冲器。

12.7.3 对绝对精度的影响

由于附加电容，会与 GND 耦合，因此如果 GND 中有噪声，就可能会降低绝对精度，必须与 AVSS 等电位稳定的 GND 连接。

在实装电路板上滤波器电路不要干扰数字信号。（不要充当天线）

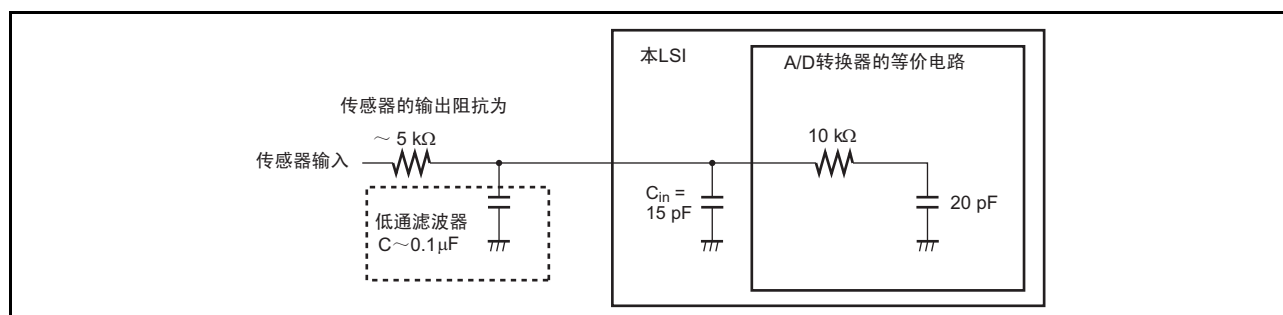


图 12.6 模拟输入电路

12.7.4 模拟电源管脚其他的设定范围

超过以下所示电压设定范围时，有时会给 LSI 的可靠性带来不良影响。

- 模拟输入电压的设定范围

A/D 转换中，向模拟输入管脚 ANn 输入的电压范围为 $AVSS \leq ANn \leq AVCC$ 。

- AVCC、AVSS 和 VCC、VSS 的关系

AVSS 和 VSS 之间的关系是 $AVSS=VSS$ ，不使用 A/D 转换器时，是 $AVCC=VCC$ ，不要将 AVCC、AVSS 管脚置为开放状态。

12.7.5 电路板设计时的注意事项

电路板设计时，要尽可能的分离数字电路和模拟电路。数字电路的信号线和模拟电路的信号线交叉或接近，应尽可能避免。由于耦合会使模拟电路发生误动作，而给 A/D 转换值带来不良影响。模拟输入管脚（AN0 ~ AN7）、模拟电源电压（AVCC）在模拟地（AVSS）上要 and 数字电路分离。将模拟地（AVSS）连接在板上较稳定的数字地（VSS）。

12.7.6 噪声对策上的注意事项

保护电路可以阻止异常电压造成的损坏，如为了防止由于过大的电涌引起的模拟输入管脚（AN0 ~ AN7）的破坏，连接如图 12.7 所示的 AVCC-AVSS 间的保护电路。务必将连接在 AVCC 上的旁路电容和连接在 AN0 ~ AN7 上的滤波器用的电容连接在 AVSS 上。

由于连接滤波用的电容，AN0 ~ AN7 的输入电流被平均化，有时会产生误差。在扫描模式下，频繁进行 A/D 转换时，向 A/D 转换器内部的采样 & 保持电路的电容充放电的电流会经过输入阻抗（ R_{in} ），如果超过输入电流范围，模拟输入管脚的电压会产生误差。因此请充分考虑后再决定电路系数。

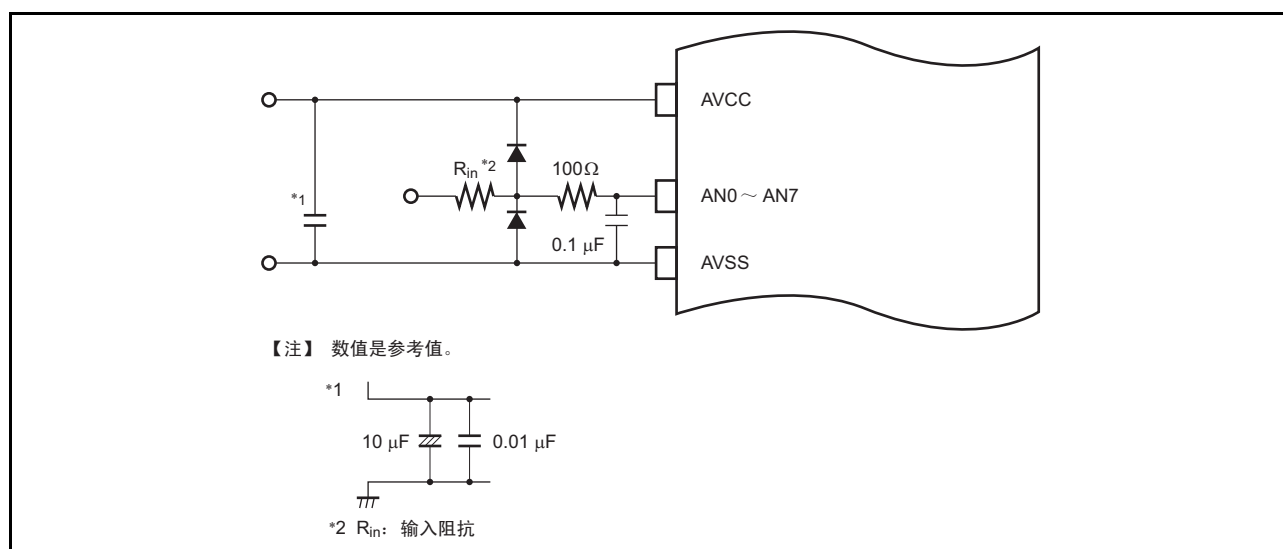


图 12.7 模拟输入保护电路

表 12.6 模拟管脚的规格

项目	min	max	单位
模拟输入电容	—	20	pF
容许信号源阻抗	—	5	k Ω

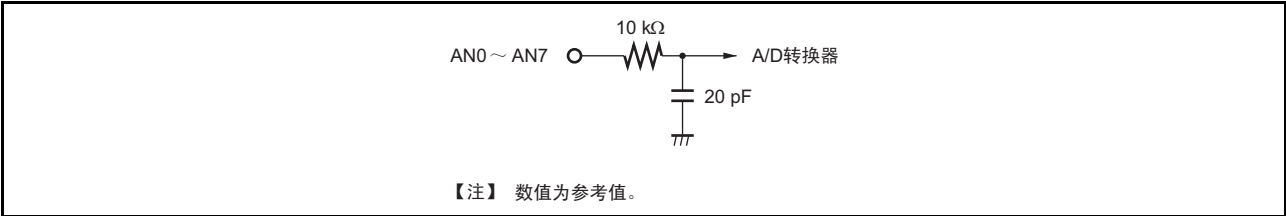


图 12.8 模拟输入管脚等价电路

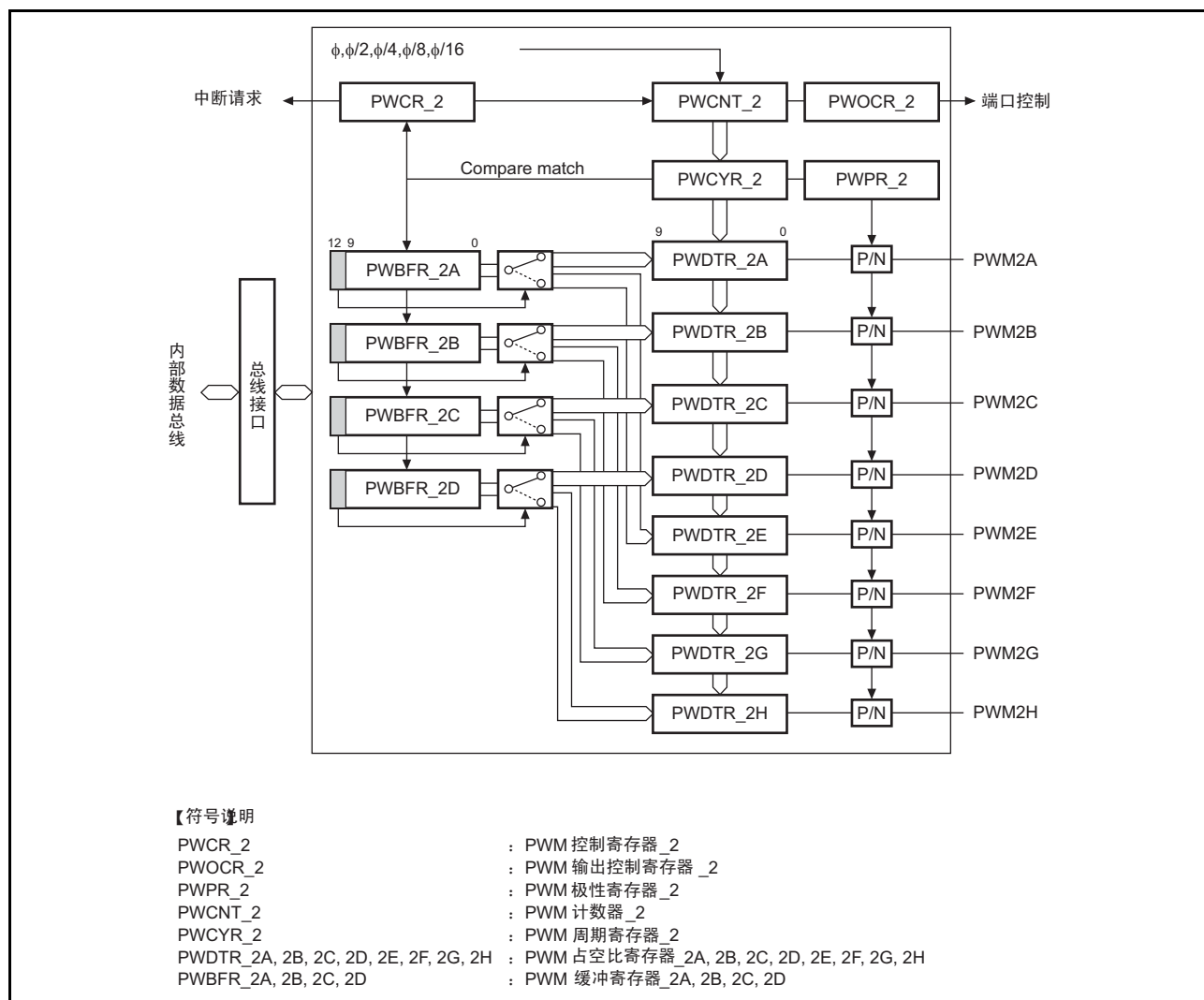


图 13.2 PWM 通道 2 的框图

13.2 输入 / 输出管脚

PWM 的管脚结构如表 13.1 所示。

表 13.1 管脚结构

名称	记号	输入 / 输出	功能
PWM 输出管脚 1A	PWM1A	输出	通道 1A 的 PWM 输出
PWM 输出管脚 1B	PWM1B	输出	通道 1B 的 PWM 输出
PWM 输出管脚 1C	PWM1C	输出	通道 1C 的 PWM 输出
PWM 输出管脚 1D	PWM1D	输出	通道 1D 的 PWM 输出
PWM 输出管脚 1E	PWM1E	输出	通道 1E 的 PWM 输出
PWM 输出管脚 1F	PWM1F	输出	通道 1F 的 PWM 输出
PWM 输出管脚 1G	PWM1G	输出	通道 1G 的 PWM 输出
PWM 输出管脚 1H	PWM1H	输出	通道 1H 的 PWM 输出
PWM 输出管脚 2A	PWM2A	输出	通道 2A 的 PWM 输出
PWM 输出管脚 2B	PWM2B	输出	通道 2B 的 PWM 输出
PWM 输出管脚 2C	PWM2C	输出	通道 2C 的 PWM 输出
PWM 输出管脚 2D	PWM2D	输出	通道 2D 的 PWM 输出
PWM 输出管脚 2E	PWM2E	输出	通道 2E 的 PWM 输出
PWM 输出管脚 2F	PWM2F	输出	通道 2F 的 PWM 输出
PWM 输出管脚 2G	PWM2G	输出	通道 2G 的 PWM 输出
PWM 输出管脚 2H	PWM2H	输出	通道 2H 的 PWM 输出

13.3 寄存器说明

PWM 中存在以下寄存器。关于模块待机控制请参考「20.1.3 模块待机控制寄存器 A ~ D (MSTPCRA ~ MSTPCRD)」。

- PWM 控制寄存器 _1、2(PWCR_1、PWCR_2)
- PWM 输出控制寄存器 _1、2(PWOOCR_1、PWOOCR_2)
- PWM 极性寄存器 _1、2(PWPR_1、PWPR_2)
- PWM 计数器 _1、2(PWCNT_1、PWCNT_2)
- PWM 周期寄存器 _1、2(PWCYR_1、PWCYR_2)
- PWM 占空比寄存器 _1A、1C、1E、1G(PWDTR_1A、PWDTR_1C、PWDTR_1E、PWDTR_1G)
- PWM 缓冲寄存器 _1A、1C、1E、1G(PWBFR_1A、PWBFR_1C、PWBFR_1E、PWBFR_1G)
- PWM 占空比寄存器 _2A ~ 2H(PWDTR_2A ~ PWDTR_2H)
- PWM 缓冲寄存器 _2A ~ 2D(PWBFR_2A ~ PWBFR_2D)

13.3.1 PWM 控制寄存器 _1、2 (PWCR_1、PWCR_2)

PWCR 进行中断控制、计数器启动 / 停止、计数器的时钟选择。并且具有表示 PWCYR 比较匹配的标记。

位	位名	初始值	R/W	说明
7、6	—	全为 1	—	保留位 读取时通常读出 1。写入定义为无效。
5	IE	0	R/W	中断许可使能位 允许或禁止 PWCYR 比较匹配时的中断请求。 0: 禁止中断 1: 允许中断
4	CMF	0	R/(W)*	比较匹配标志位 表示发生 PWCYR 比较匹配。 [置位条件] PWCNT = PWCYR 时 [清除条件] 在 CMF = 1 的状态读取 CMF 后, 在 CMF 中写入 0 时
3	CST	0	R/W	计数器开启位 启动或停止 PWCNT 0: 停止 PWCNT 1: 启动 PWCNT
2 1 0	CK2 CK1 CK0	0 0 0	R/W R/W R/W	时钟频率选择位 选择 PWCNT 的运行时钟。 000: 以 $\phi/1$ 计数 001: 以 $\phi/2$ 计数 010: 以 $\phi/4$ 计数 011: 以 $\phi/8$ 计数 1xx: 以 $\phi/16$ 计数

【注】 * 为清除标记只能写入 0。

【符号说明】x: Don't Care

13.3.2 PWM 输出控制寄存器 _1、2 (PWOCR_1、PWOCR_2)

PWOCR 禁止或允许 PWM 输出。PWOCR_1 各位对应 PWM1H ~ PWM1A 输出、PWOCR_2 对应 PWM2H ~ PWM2A 输出。

• PWOCR_1

位	位名	初始值	R/W	说明
7	OE1H	0	R/W	输出允许
6	OE1G	0	R/W	各位允许或禁止 PWM1H ~ PWM1A 输出。
5	OE1F	0	R/W	0: 禁止 PWM 输出
4	OE1E	0	R/W	1: 允许 PWM 输出
3	OE1D	0	R/W	
2	OE1C	0	R/W	
1	OE1B	0	R/W	
0	OE1A	0	R/W	

• PWOCR_2

位	位名	初始值	R/W	说明
7	OE2H	0	R/W	输出允许
6	OE2G	0	R/W	各位允许或禁止 PWM2H ~ PWM2A 输出。
5	OE2F	0	R/W	0: 禁止 PWM 输出
4	OE2E	0	R/W	1: 允许 PWM 输出
3	OE2D	0	R/W	
2	OE2C	0	R/W	
1	OE2B	0	R/W	
0	OE2A	0	R/W	

13.3.3 PWM 极性寄存器 _1、2 (PWPR_1、PWPR_2)

PWPR 用于选择 PWM 输出极性。PWPR_1 对应 PWM1H ~ PWM1A 输出、PWPR_2 对应 PWM2H ~ PWM2A 输出。

• PWPR_1

位	位名	初始值	R/W	说明
7	OPS1H	0	R/W	极性选择
6	OPS1G	0	R/W	各位用于选择 PWM1H ~ PWM1A 输出的极性。 0: PWM 正相输出 1: PWM 反相输出
5	OPS1F	0	R/W	
4	OPS1E	0	R/W	
3	OPS1D	0	R/W	
2	OPS1C	0	R/W	
1	OPS1B	0	R/W	
0	OPS1A	0	R/W	

• PWPR_2

位	位名	初始值	R/W	说明
7	OPS2H	0	R/W	极性选择
6	OPS2G	0	R/W	各位用于选择 PWM2H ~ PWM2A 输出的极性。 0: PWM 正相输出 1: PWM 反相输出
5	OPS2F	0	R/W	
4	OPS2E	0	R/W	
3	OPS2D	0	R/W	
2	OPS2C	0	R/W	
1	OPS2B	0	R/W	
0	OPS2A	0	R/W	

13.3.4 PWM 计数器 _1、2 (PWCNT_1、PWCNT_2)

PWCNT 是 10 位递增计数器，用输入的时钟递增计数。根据 PWCR 的 CKS2 ~ CKS0 位选择输入的时钟频率。

PWCNT_1 作为通道 1 的时基，PWCNT_2 作为通道 2 的时基使用。

PWCR 的 CST 位为 0 及复位、待机模式、监视模式、子激活模式、子睡眠模式、模块停止模式时，PWCNT 被初始化。PWCNT 的初始值是 H'FC00。

13.3.5 PWM 周期寄存器 _1、2 (PWCYR_1、PWCYR_2)

PWCYR 是可进行 16 位读取 / 写入及设定 PWM 转换周期的寄存器。如果发生 PWCYR 比较匹配，PWCNT 被清 0，从缓冲寄存器 (PWBFR) 向占空比寄存器 (PWDTR) 传送数据。PWCYR_1 用于设定通道 1 转换周期，PWCYR_2 用于设定通道 2 的转换周期。

在 PWCNT 停止中，进行对 PWCYR 的写入。同时请不要在 PWCYR 设定 H'FC00。PWCYR 的初始值是 H'FFFF。周期寄存器的比较匹配如图 13.3 所示。

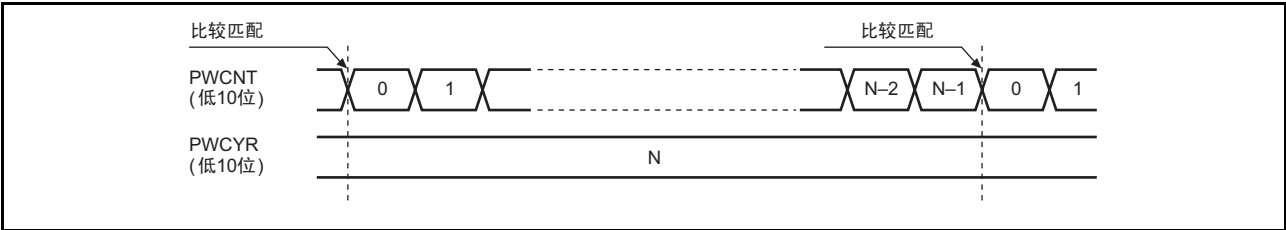


图 13.3 周期寄存器的比较匹配

13.3.6 PWM 占空比寄存器 _1A、1C、1E、1G (PWDTR_1A、PWDTR_1C、PWDTR_1E、PWDTR_1G)

PWDTR_1 中有 4 个寄存器。用 OTS 位选择 PWM 输出，PWDTR_1A 对应 PWM1A、PWM1B 输出，PWDTR_1C 对应 PWM1C、PWM1D 输出，PWDTR_1E 对应 PWM1E、PWM1F 输出，PWDTR_1G 对应 PWM1G、PWM1H 输出。PWDTR_1 不能直接读取 / 写入。如果发生 PWCYR_1 的比较匹配，数据会从缓冲寄存器 1 (PWBFR_1) 传送到 PWDTR_1。

PWDTR_1 在 PWCR_1 的 CST 位清零后被初始化。而且在复位、待机模式、模块待机模式也会被初始化。

位	位名	初始值	R/W	说明
15~13	—	全为 0	—	保留位 不能读取 / 写入。
12	OTS	0	—	输出终端选择 通过 PWCYR_1 的比较匹配，反映被传送的 PWBFR_1 的第 12 位值，选择进行 PWM 输出的管脚。没有选择的管脚输出低电平 (PWPR_1 的该位为 1 时，是高电平)。 PWDTR_1A 寄存器 0: 选择 PWM1A 输出 1: 选择 PWM1B 输出 PWDTR_1C 寄存器 0: 选择 PWM1C 输出 1: 选择 PWM1D 输出 PWDTR_1E 寄存器 0: 选择 PWM1E 输出 1: 选择 PWM1F 输出 PWDTR_1G 寄存器 0: 选择 PWM1G 输出 1: 选择 PWM1H 输出
11、10	—	全为 1	—	保留位 不能读取 / 写入。
9	DT9	0	—	占空比 通过 PWCYR_1 的比较匹配，反映被传送的 PWBFR_1 位 9 ~ 0 的数据，设定 PWM 输出的占空比。发生 PWCYR_1 比较匹配，PWCNT_1 被清 0 后，到发生 PWDTR_1 的比较匹配为止，在此期间，输出高电平 (PWPR_1 的该位为 1 时，输出低电平)。所有位为 0 时，不输出高电平 (PWPR_1 的该位为 1 时，输出低电平)。
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	—	

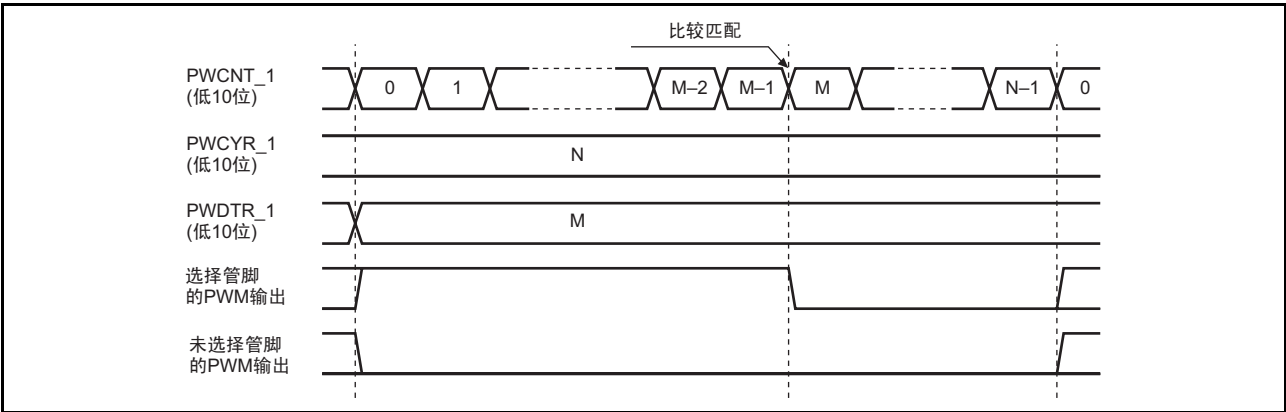


图 13.4 占空比寄存器的比较匹配 (PWPR_1 的 OPS=0)

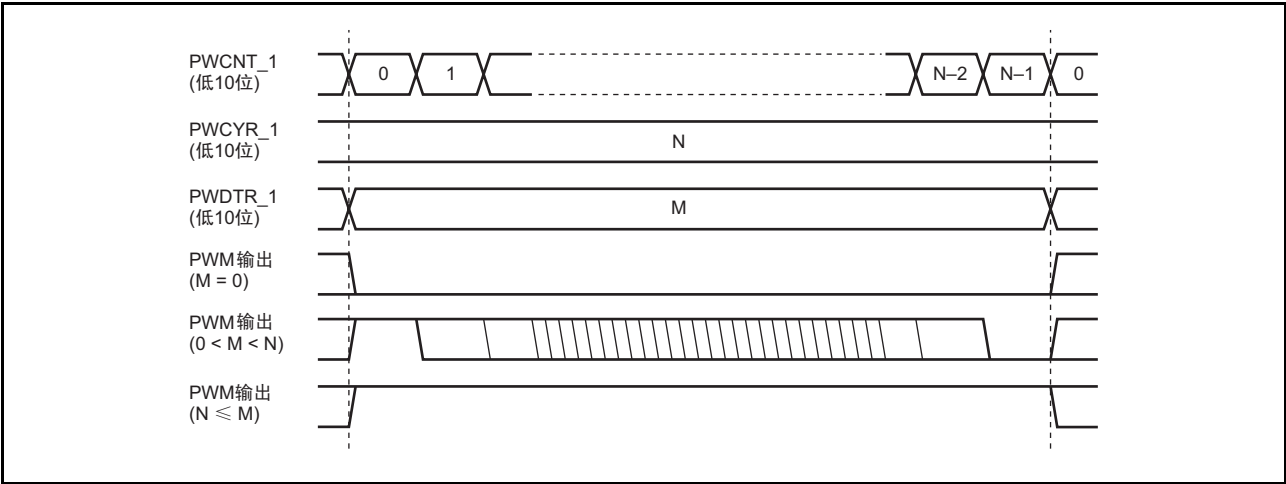


图 13.5 占空比寄存器设定值不同 PWM 输出也不同 (PWPR_1 的 OPS=0)

13.3.7 PWM 缓冲寄存器_1A、1C、1E、1G (PWBFR_1A、PWBFR_1C、PWBFR_1E、PWBFR_1G)

PWBFR_1 中有 4 个寄存器。如果发生 PWCYR_1 的比较匹配，数据从 PWBFR_1A 传送到 PWDTR_1A，从 PWBFR_1C 传送到 PWDTR_1C，从 PWBFR_1E 传送到 PWDTR_1E，从 PWBFR_1G 传送到 PWDTR_1G。

位	位名	初始值	R/W	说明
15~13	—	全为 1	—	保留位 读取时通常读出 1。写入定义为无效。
12	OTS	0	R/W	输出终端选择 传送到 PWDTR_1 的位 12 的数据。
11、10	—	全为 1	—	保留位 读取时通常读出 1。写入定义为无效。
9	DT9	0	R/W	占空比 传送到 PWDTR_1 的位 9 ~ 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	

13.3.8 PWM 占空比寄存器 _2A ~ 2H (PWDTR_2A ~ PWDTR_2H)

PWDTR_2 中有 8 个寄存器。分别是 PWDTR_2A 对应 PWM2A 输出，PWDTR_2B 对应 PWM2B 输出，PWDTR_2C 对应 PWM2C 输出，PWDTR_2D 对应 PWM2D 输出，PWDTR_2E 对应 PWM2E 输出，PWDTR_2F 对应 PWM2F 输出，PWDTR_2G 对应 PWM2G 输出，PWDTR_2H 对应 PWM2H 输出。PWDTR_2 不能直接读取 / 写入。如果发生 PWCYR_2 的比较匹配，数据会从缓冲寄存器 2 (PWBFR_2) 传送到 PWDTR_2。

如果 PWCR_2 的 CST 位被清零，PWDTR_2 被初始化。而且在复位、待机模式、模块待机模式时也会被初始化。

位	位名	初始值	R/W	说明
15~10	—	全为 1	—	保留位 不能读取 / 写入。
9	DT9	0	—	占空比 通过 PWCYR_2 的比较匹配，反映传送的 PWBFR_2 的位 9 ~ 0 的数据，设定 PWM 输出的占空比。发生 PWCYR_2 的比较匹配，PWCNT_2 被清 0 后，到发生 PWDTR_2 的比较匹配为止，在此期间，输出高电平 (PWPR_2 的该位为 1 时，输出低电平)。所有位为 0 时，不输出高电平 (PWPR_2 的该位为 1 时，输出低电平)。
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	—	

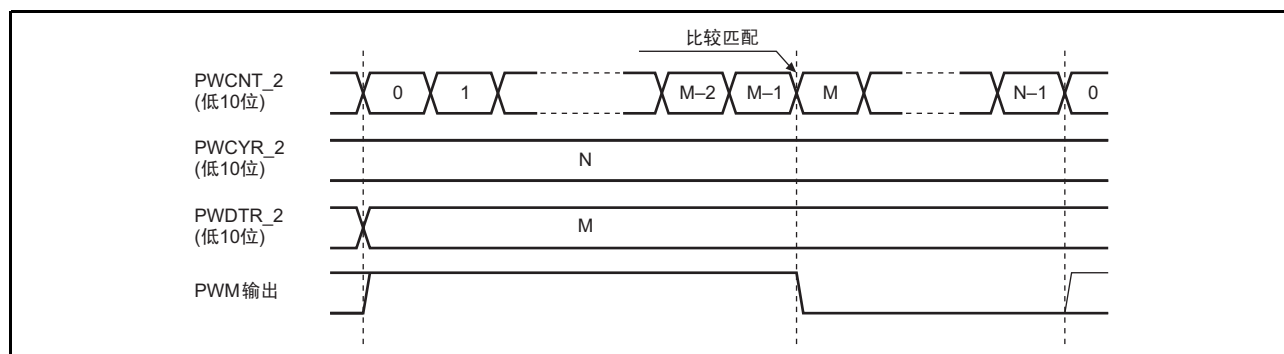


图 13.6 占空比寄存器的比较匹配 (PWPR_2 的 OPS=0)

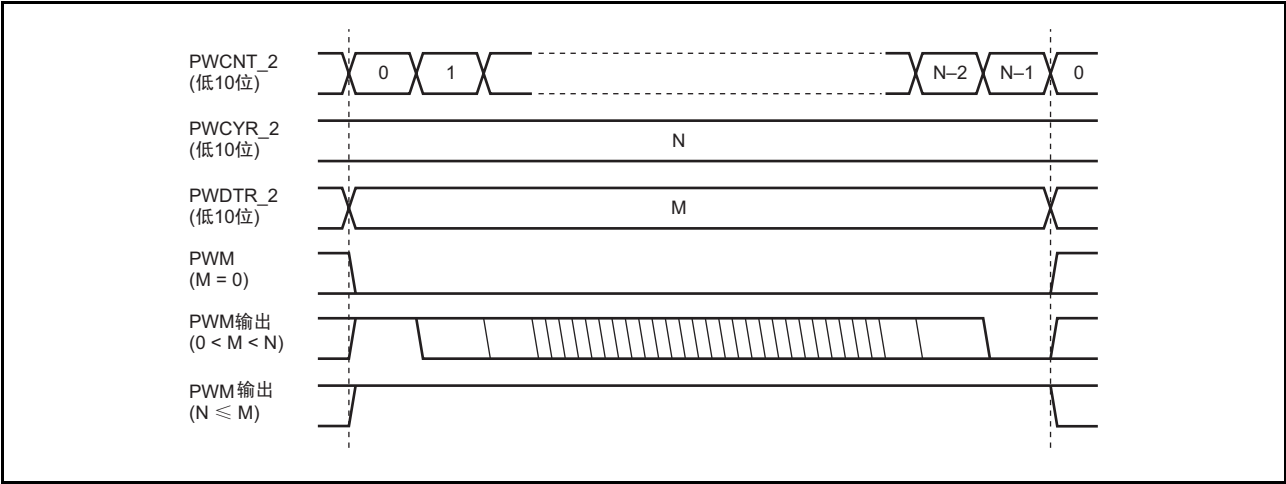


图 13.7 占空比寄存器的设定值不同 PWM 输出也不同 (PWPR_2 的 OPS=0)

13.3.9 PWM 缓冲寄存器 _2A ~ 2D (PWBFR_2A ~ PWBFR_2D)

PWBFR_2 中有 4 个寄存器。用 TDS 位选择传送地址，如果发生 PWCYR_2 的比较匹配，数据就会从 PWBFR_2A 传送到 PWDTR_2A 或 PWDTR_2E，从 PWBFR_2B 传送到 PWDTR_2B 或 PWDTR_2F，从 PWBFR_2C 传送到 PWDTR_2C 或 PWDTR_2G，从 PWBFR_2D 传送到 PWDTR_2D 或 PWDTR_2H。

位	位名	初始值	R/W	说明
15~13	—	全为 1	—	保留位 读取时通常读出 1。写入定义为无效。
12	TDS	0	R/W	选择转移目标 选择数据传送地址的 PWDTR_2。 PWBFR_2A 寄存器 0: 选择 PWDTR_2A 1: 选择 PWDTR_2E PWBFR_2B 寄存器 0: 选择 PWDTR_2B 1: 选择 PWDTR_2F PWBFR_2C 寄存器 0: 选择 PWDTR_2C 1: 选择 PWDTR_2G PWBFR_2D 寄存器 0: 选择 PWDTR_2D 1: 选择 PWDTR_2H
11、10	—	全为 1	—	保留位 读取时通常读出 1。写入定义为无效。
9	DT9	0	R/W	占空比 传送到 PWDTR_2 的位 9 ~ 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	

13.4 与总线主控器的接口

13.4.1 16 位数据寄存器

PWCYR_1、PWCYR_2、PWBFR_1A、PWBFR_1C、PWBFR_1E、PWBFR_1G、PWBFR_2A ~ PWBFR_2D 是 16 位寄存器。和总线主控器之间的数据总线是 16 位宽，可进行 16 位的读取 / 写入。由于不能以 8 位读取 / 写入，通常要以 16 位存取。

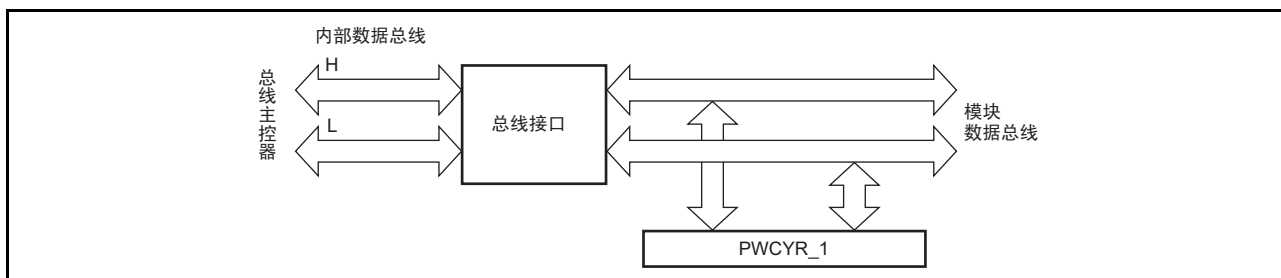


图 13.8 16 位寄存器的存取运行 [总线主控器 $\longleftrightarrow$ PWCYR_1 (16 位)]

13.4.2 8 位数据寄存器

PWCR_1、PWCR_2、PWOCR_1、PWOCR_2、PWPR_1、PWPR_2 是 8 位寄存器。可进行 8 位的读取 / 写入。因总线主控器之间的数据总线是 16 位宽，可进行 16 位的读取 / 写入。但即使读出低 8 位，也不能保证读出的值。

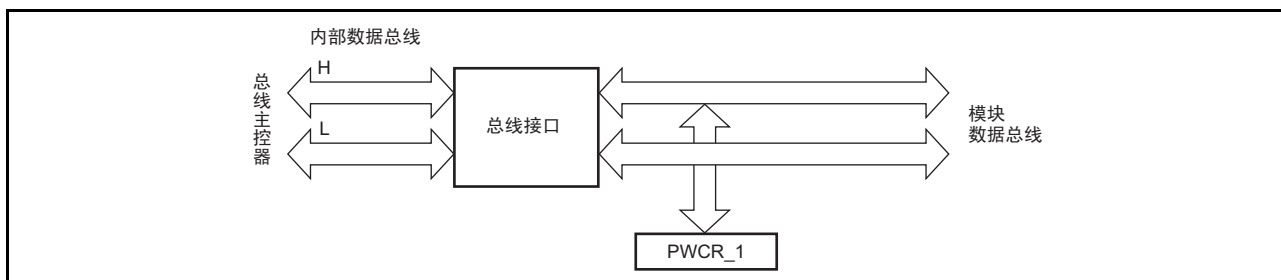


图 13.9 8 位寄存器的存取运行 [总线主控器 $\longleftrightarrow$ PWCR_1 (高 8 位)]

13.5 运行说明

13.5.1 PWM 通道 1 的运行

从 PWM1A ~ PWM1H 管脚输出如图 13.10 所示的 PWM 波形。

(1) 初始设定

用 PWPR_1 设定 PWM 输出极性。用 PWOCR_1 将 PWM1A ~ PWM1H 管脚设定为 PWM 输出。通过 PWCR_1 的 CKS2 ~ CKS0 位, 选择输入到 PWCNT_1 的时钟频率。通过 PWCYR_1 设定 PWM 的转换周期。在 PWBFR_1A、PWBFR_1C、PWBFR_1E、PWBFR_1G 设定第 1 帧数据。

(2) 启动

如果将 PWCR_1 的 CST 位置为 1, 会发生 PWCNT_1 和 PWCYR_1 的比较匹配。数据从 PWBFR_1A 传送到 PWDTR_1A, 从 PWBFR_1C 传送到 PWDTR_1C, 从 PWBFR_1E 传送到 PWDTR_1E, 从 PWBFR_1G 传送到 PWDTR_1G。PWCNT_1 开始递增计数。这时 PWCR_1 的 CMF 位被置位, 如果将 PWCR_1 的 IE 位置位, 就可发生中断请求。

(3) 波形输出

如果发生 PWCNT_1 和 PWCYR_1 的比较匹配, 通过 PWDTR_1A、PWDTR_1C、PWDTR_1E、PWDTR_1G 的 OTS 位选择的 PWM 输出, 就会输出高电平。没有被选择的 PWM, 则输出低电平。如果发生 PWCNT_1 和 PWDTR_1A、PWDTR_1C、PWDTR_1E、PWDTR_1G 的比较匹配, 就会从对应的 PWM 输出低电平。PWPR_1 对应的位为 1 时, 反相输出。

(4) 下一帧

如果发生 PWCNT_1 和 PWCYR_1 的比较匹配, 数据会从 PWBFR_1A 传送到 PWDTR_1A, 从 PWBFR_1C 传送到 PWDTR_1C, 从 PWBFR_1E 传送到 PWDTR_1E, 从 PWBFR_1G 传送到 PWDTR_1G。PWCNT_1 被复位, 从 H'000 开始递增计数。PWCR_1 的 CMF 位置位后, 如果 PWCR_1 的 IE 位被置位, 就可发生中断请求。

(5) 停止

如果将 PWCR_1 的 CST 位清 0, PWCNT_1 被复位并停止。从各 PWM 输出低电平 (PWPR_1 对应的位为 1 时, 输出高电平)。

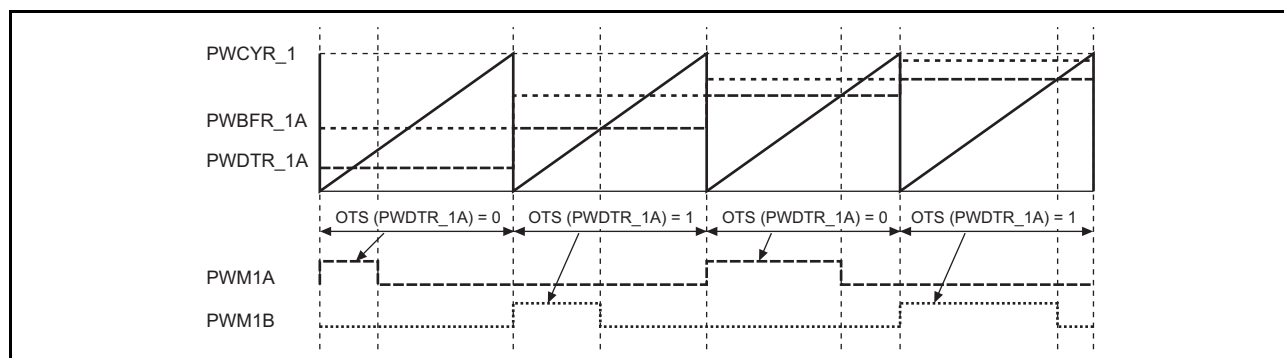


图 13.10 PWM 通道 1 的运行

13.5.2 PWM 通道 2 的运行

从 PWM2A ~ PWM2H 管脚输出如图 13.11 所示的 PWM 波形。

(1) 初始设定

用 PWPR_2 设定 PWM 输出极性。用 PWOCR_2 将 PWM2A ~ PWM2H 管脚设定为 PWM 输出。通过 PWCR_2 的 CKS2 ~ CKS0 位选择输入到 PWCNT_2 的时钟频率。通过 PWCYR_2 设定 PWM 的转换周期。在 PWBFR_2A、PWBFR_2B、PWBFR_2C、PWBFR_2D 设定第 1 帧数据。

(2) 启动

如果将 PWCR_2 的 CST 位置为 1，会发生 PWCNT_2 和 PWCYR_2 的比较匹配。通过 TDS 位，数据会从 PWBFR_2A 传送到 PWDTR_2A 或 PWDTR_2E，从 PWBFR_2B 传送到 PWDTR_2B 或 PWDTR_2F，从 PWBFR_2C 传送到 PWDTR_2C 或 PWDTR_2G，从 PWBFR_2D 传送到 PWDTR_2D 或 PWDTR_2H。PWCNT_2 开始递增计数。这时 PWCR_2 的 CMF 位被置位，如果 PWCR_2 的 IE 位被置位，就可发生中断请求。

(3) 波形输出

如果发生 PWCNT_2 和 PWCYR_2 的比较匹配，各 PWM 输出高电平。如果发生 PWCNT_2 和 PWDTR_2A ~ PWDTR_2H 的比较匹配，会从对应的 PWM 输出低电平。PWPR_2 对应的位为 1 时，反相输出。

(4) 下一帧

如果发生 PWCNT_2 和 PWCYR_2 的比较匹配，根据 TDS 位的值，数据从 PWBFR_2A 被传送到 PWDTR_2A 或 PWDTR_2E，从 PWBFR_2B 被传送到 PWDTR_2B 或 PWDTR_2F，从 PWBFR_2C 传送到 PWDTR_2C 或 PWDTR_2G，从 PWBFR_2D 传送到 PWDTR_2D 或 PWDTR_2H。PWCNT_2 复位后，从 H'000 开始递增计数。PWCR_2 的 CMF 位被置位，如果 PWCR_2 的 IE 位被置位，就可发生中断请求。

(5) 停止

将 PWCR_2 的 CST 位清 0，PWCNT_2 被复位并且停止。PWDTR_2A ~ PWDTR_2H 被复位。从各 PWM 输出低电平 (PWPR_2 对应的位为 1 时，输出高电平)。

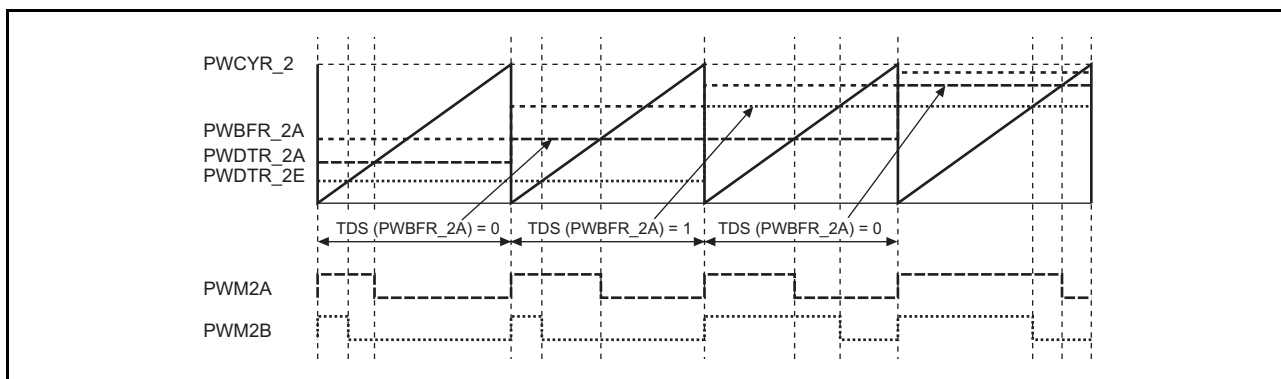


图 13.11 PWM 通道 2 的运行

13.6 中断源

由于发生 PWCNT 和 PWCYR 比较匹配，PWCR 的 CMF 标志位被置为 1 时，如果 PWCR 的 IE 位被置为 1，就会请求中断。表 13.2 所示的是 PWM 的中断源。

表 13.2 PWM 的中断源

名称	中断源	中断标记
CMI1	PWCYR_1 的比较匹配	CMF
CMI2	PWCYR_2 的比较匹配	CMF

13.7 使用时的注意事项

(1) 缓冲寄存器的写入和比较匹配的竞争

如果在周期寄存器比较匹配后的状态进行 PWBFR 的写入，占空比寄存器与缓冲寄存器会同时被改写。由于竞争而改写占空比寄存器时，因周期寄存器的比较匹配而改变的 PWM 输出不发生变化，因此时常会出现随机输出占空比的情况。同时，在通道 2，由于竞争而改写占空比寄存器时，也会因为选择了用缓冲寄存器的 TDS 位传送的占空比寄存器，而出现随机改写占空比寄存器的情况。由于比较匹配中断而引起的意外处理或检测到 PWCR 的 CMF 标志位为高后，要在发生周期寄存器的比较匹配前，结束缓冲寄存器的改写。

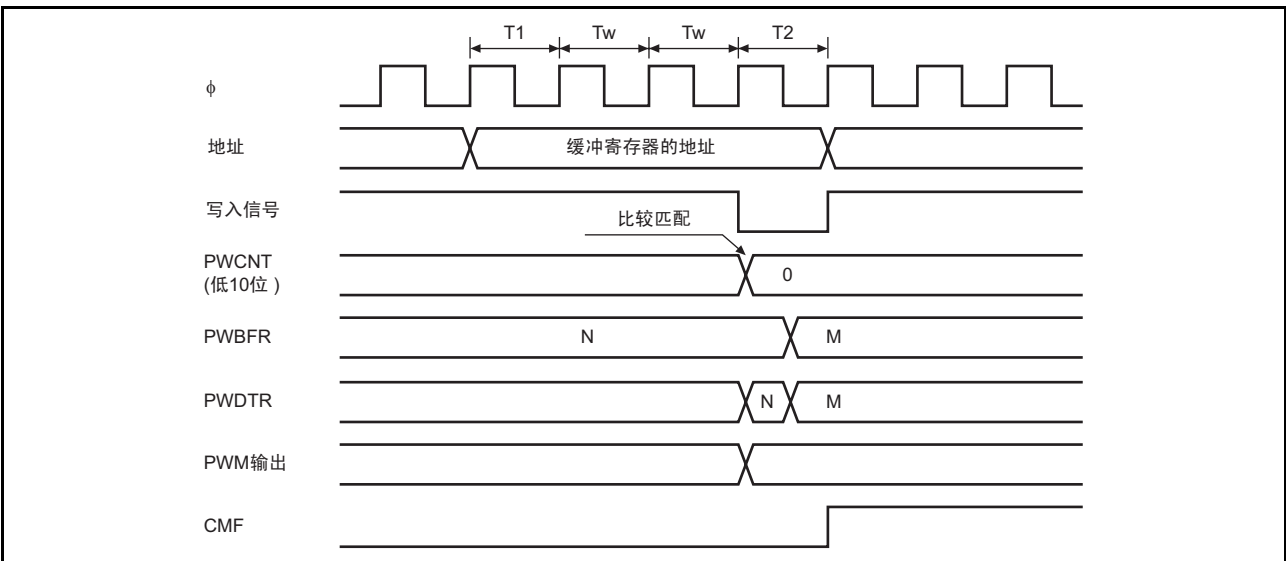


图 13.12 缓冲寄存器的写入和比较匹配的竞争

第 14 章 LCD 控制器 / 驱动器 (LCD)

LSI 内部段类型包括 LCD 控制器电路、LCD 驱动器、电源电路，可直接驱动的一个 LCD 面板。

14.1 特点

- 显示能力

占空比	内部驱动	
	H8S/2282 组、HD64F2280B	HD64F2280RB
静态	28SEG	32SEG
1/3	28SEG	32SEG
1/4	28SEG	32SEG

- LCD RAM 的显示能力
8 位 ×20 字节 (160 位)
LCD RAM 可存取字节或字符
- 段的输出管脚每四个一组都可以作为通用端口使用
- 由于占空比的原因可将不使用的公共输出管脚作为公共双缓冲使用 (并联线路)
在静态模式下，平行连接的 COM1 和 COM2、COM3 和 COM4 可作为并联线路使用
- 有 11 种帧频可供选择
- 可根据软件选择 A 波形、B 波形
- 内部电源分压电阻
- 可在待机模式、模块停止模式以外的运行模式显示

LCD 控制器 / 驱动器的框图如图 14.1 所示。

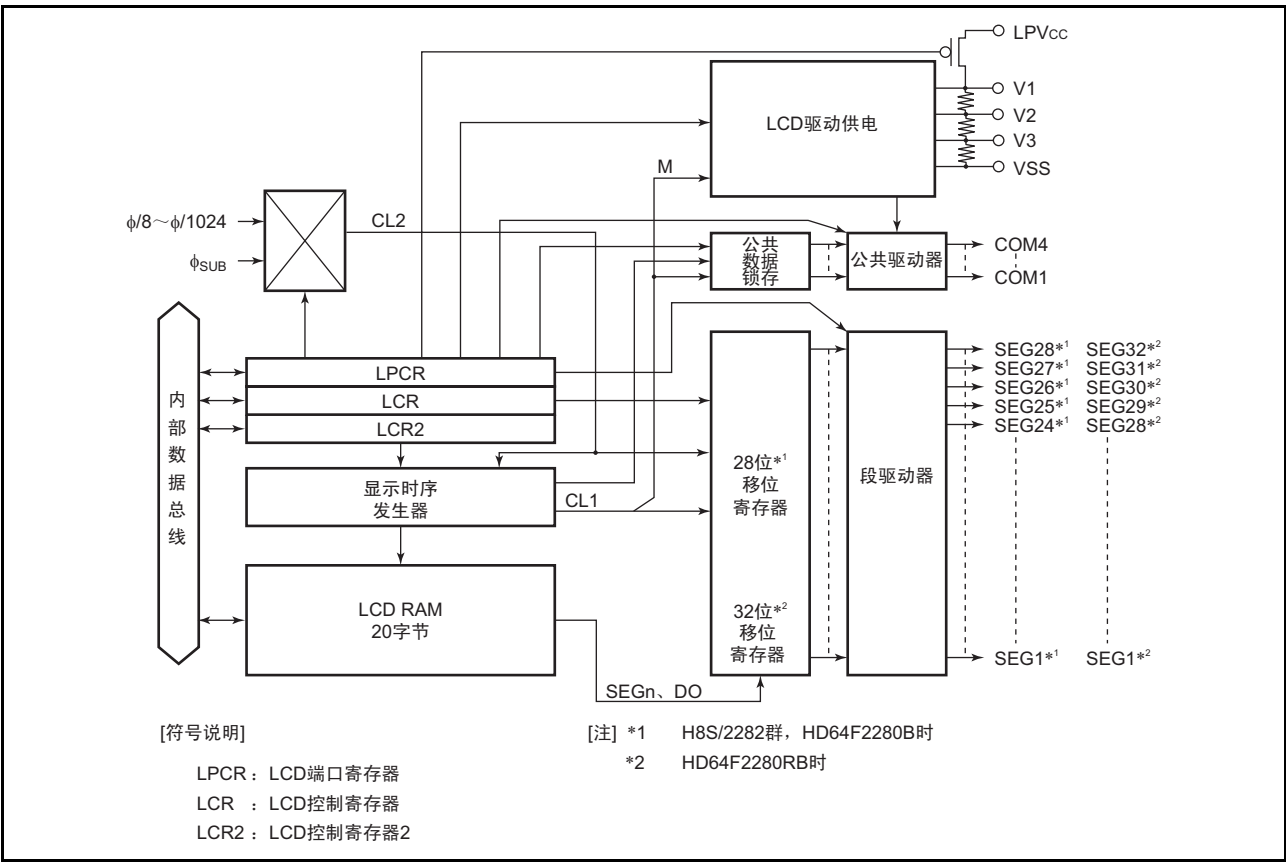


图 14.1 LCD 控制器 / 驱动器的框图

14.2 输入 / 输出管脚

LCD 控制器 / 驱动器的管脚结构如表 14.1 所示。

表 14.1 管脚结构

管脚名	符号	输入 / 输出	功能
段输出管脚	SEG32 ~ SEG1*	输出	液晶段驱动的管脚。 所有管脚可作为端口管脚多路复用。设定为可编程。
公共输出管脚	COM4 ~ COM1	输出	液晶的公共驱动管脚。 静态时管脚可以并联。
LCD 电源管脚	V1 ~ V3	—	连接外置旁路电容或外部电源电路时使用。

【注】 *H8S/2282 组、HD64F2280B 的情况下是 SEG28 ~ SEG1。

14.3 各寄存器的说明

LCD 控制器 / 驱动器有以下寄存器。关于模块待机控制请参照「20.1.3 模块待机控制寄存器 A ~ D (MSTPCRA ~ MSTPCRD)」。

- LCD 端口控制寄存器 (LPCR)
- LCD 控制寄存器 (LCR)
- LCD 控制寄存器 2 (LCR2)

14.3.1 LCD 端口控制寄存器 (LPCR)

LPCR 可进行占空比、LCD 驱动器和管脚功能的选择。

位	位名	初始值	R/W	说明
7	DTS1	0	R/W	占空比选择 1、0 任意选择静态、1/3、1/4 占空比。 详细内容请参考表 14.2。
6	DTS0	0	R/W	
5	CMX	0	R/W	公共功能选择 当所有的公共管脚因占空比设定而不被使用时，为增加公共驱动力，指定从多个管脚输出的波形是否相同。详细内容请参考表 14.2。
4	—	0	R/W	保留位 通常写入 0。
3	SGS3	0	R/W	段驱动器选择 3 ~ 0。 选择要使用的 3~0 段驱动器。详细内容请参考表 14.3。
2	SGS2	0	R/W	
1	SGS1	0	R/W	
0	SGS0	0	R/W	

表 14.2 占空比和公共功能的选择

位 7	位 6	位 5	占空比	公共驱动器	说明
DTS1	DTS0	CMX			
0	0	0	静态	COM1	COM4 ~ COM2 可作为端口使用
0	0	1	静态	COM4 ~ COM1	COM4 ~ COM2 输出与 COM1 相同的波形
0	1	X	—	—	禁止设定
1	0	0	1/3 占空比	COM3 ~ COM1	COM4 可作为端口使用
1	0	1	1/3 占空比	COM4 ~ COM1	禁止使用 COM4
1	1	X	1/4 占空比	COM4 ~ COM1	

【符号说明】X: 与设定无关

表 14.3 段驱动器的选择

• < H8S/2282 群、HD64F2280B >

位 3	位 2	位 1	位 0	SEG28 ~ SEG1 管脚的功能					
SGS3	SGS2	SGS1	SGS0	SEG28 ~ SEG21	SEG20 ~ SEG17	SEG16 ~ SEG13	SEG12 ~ SEG9	SEG8 ~ SEG5	SEG4 ~ SEG1
0	0	0	0	端口	端口	端口	端口	端口	端口
0	0	0	1	SEG	端口	端口	端口	端口	端口
0	0	1	0	SEG	SEG	端口	端口	端口	端口
0	0	1	1	SEG	SEG	SEG	端口	端口	端口
0	1	0	0	SEG	SEG	SEG	SEG	端口	端口
0	1	0	1	SEG	SEG	SEG	SEG	SEG	端口
0	1	1	0	SEG	SEG	SEG	SEG	SEG	SEG
0	1	1	1	禁止设定					
1	X	X	X	禁止设定					

【符号说明】X: 与设定无关

• < HD64F2280RB 的情况 >

位 3	位 2	位 1	位 0	SEG28 ~ SEG1 管脚的功能						
SGS3	SGS2	SGS1	SGS0	SEG32 ~ SEG25	SEG24 ~ SEG21	SEG20 ~ SEG17	SEG16 ~ SEG13	SEG12 ~ SEG9	SEG8 ~ SEG5	SEG4 ~ SEG1
0	0	0	0	端口	端口	端口	端口	端口	端口	端口
0	0	0	1	SEG	端口	端口	端口	端口	端口	端口
0	0	1	0	SEG	SEG	端口	端口	端口	端口	端口
0	0	1	1	SEG	SEG	SEG	端口	端口	端口	端口
0	1	0	0	SEG	SEG	SEG	SEG	端口	端口	端口
0	1	0	1	SEG	SEG	SEG	SEG	SEG	端口	端口
0	1	1	0	SEG	SEG	SEG	SEG	SEG	SEG	端口
0	1	1	1	SEG	SEG	SEG	SEG	SEG	SEG	SEG
1	X	X	X	禁止设定						

【符号说明】X: 与设定无关

14.3.2 LCD 控制寄存器 (LCR)

LCR 可执行 LCD 电源分压电阻连接控制、数据显示控制、帧频的选择。

位	位名	初始值	R/W	说明
7	—	1	—	保留位 读取时通常读入 1，不能更改。
6	PSW	0	R/W	LCD 电源分压电阻连接控制 低功耗模式下不需要 LCD 显示或使用外部电源时，位 6 可从 VCC 切断 LCD 电源分压电阻。 ACT=0 或待机模式时，与该位的设定无关，可从 VCC 切断 LCD 电源分压电阻。 0: 从 VCC 切断 LCD 电源分压电阻 1: 将 LCD 电源分压电阻连接到 VCC
5	ACT	0	R/W	激活显示功能 位 5 指出是否使用 LCD 控制器 / 驱动器。将此位清 0，LCD 控制器 / 驱动器停止运行。LCD 驱动电源的梯形电阻变成 OFF 状态，与 PSW 位的设置无关。但是仍保持寄存器的内容。 0: LCD 控制器 / 驱动器运行中断。 1: LCD 控制器 / 驱动器运行
4	DISP	0	R/W	数据显示控制 位 4 可以指出是显示 LCD RAM 的内容，还是显示空数据。 0: 显示空数据 1: 显示 LCD RAM 数据
3	CKS3	0	R/W	帧频选择 3 ~ 0。 位 3~0 是选择时钟和帧频。详细内容请参照表 14.4。
2	CKS2	0	R/W	
1	CKS1	0	R/W	
0	CKS0	0	R/W	

表 14.4 使用时钟和帧频的选择

位 3	位 2	位 1	位 0	时钟	帧频 *1
CKS3	CKS2	CKS1	CKS0		$\phi=20\text{MHz}$
0	X	0	0	ϕ_{SUB}	128Hz^{*2}
0	X	0	1	$\phi_{\text{SUB}}/2$	64Hz^{*2}
0	X	1	X	$\phi_{\text{SUB}}/4$	32Hz^{*2}
1	0	0	0	$\phi/8$	4880Hz
1	0	0	1	$\phi/16$	2440Hz
1	0	1	0	$\phi/32$	1220Hz
1	0	1	1	$\phi/64$	610Hz
1	1	0	0	$\phi/128$	305Hz
1	1	0	1	$\phi/256$	152.6Hz
1	1	1	0	$\phi/512$	76.3Hz
1	1	1	1	$\phi/1024$	38.1Hz

【注】 *1 选择 1/3 占空比时，帧频变为如表所示值的 4/3 倍。

*2 $\phi_{\text{SUB}}=32.768\text{kHz}$ 时的帧频。

【符号说明】X: 与设定无关

14.3.3 LCD 控制寄存器 2 (LCR2)

LCR2 控制 A 波形 /B 波形的切换。

位	位名	初始值	R/W	说明
7	LCDAB	0	R/W	A 波形 /B 波形切换控制 位 7 指出 LCD 的驱动波形是 A 波形还是 B 波形。 0: 用 A 波形驱动 1: 用 B 波形驱动
6、5	—	全部 1	—	保留位 读取时通常读出 1，不能更改。
4~0	—	全部 0	—	保留位 写入值通常为 0。

14.4 运行说明

14.4.1 LCD 显示的设置

为了 LCD 显示，必须根据硬件和软件的关联决定以下几点。

(1) 硬件设置

(a) 液晶板

内部电源分压电阻阻抗较大，时常会与液晶板的驱动不匹配。显示不清晰时，参考「14.4.4 增强 LCD 驱动电源」。选择静态模式，可增加公共输出的驱动能力。选择占空比时，应将 LPCR 的 CMX 位置为 1。

静态时，COM4 ~ COM1 管脚将输出相同波形。

(b) LCD 驱动电源的设定

对于 H8S/2282 有使用内部电源电路和外部电源电路两种方法作为 LCD 驱动电源。使用外部电源电路作为 LCD 驱动电源时，要在 V1 管脚连接外部电源。

(2) 软件设置

(a) 占空比的选择

通过设定 DTS1、DTS0 位来选择占空比。

(b) 段驱动器的选择

通过设定 SGS3 ~ SGS0 位来选择段驱动器。

(c) 帧频的选择

通过设定 CKS3 ~ CKS0 位来选择帧频。。帧频的选择应和 LCD 面板说明相一致，对于监视模式、子激活模式、睡眠模式、子睡眠模式时的时钟选择方法，参照「14.4.3 低功耗模式的运行」。

(d) A、B 波形的选择

通过 LCD A B 位选择使用 A 还是 B 波形。

(e) LCD 驱动电源的选择

使用外部电源电路时，通过将 PSW 位清零将 LCD 驱动电源转换为 OFF 状态。

14.4.2 LCD RAM 和显示的关系

(1) H8S/2282 群、HD64F2280B

LCD RAM 和显示段的关系因占空比的不同而不同。占空比对应的 LCD RAM 映射如图 14.2 ~ 图 14.4 所示。

设定了显示所必须的寄存器后，在与占空比对应的部分，用与通用 RAM 相同的指令写入数据，当将状态转换为 ON 时就可自动显示。RAM 可使用字 / 字节存取命令来设定。

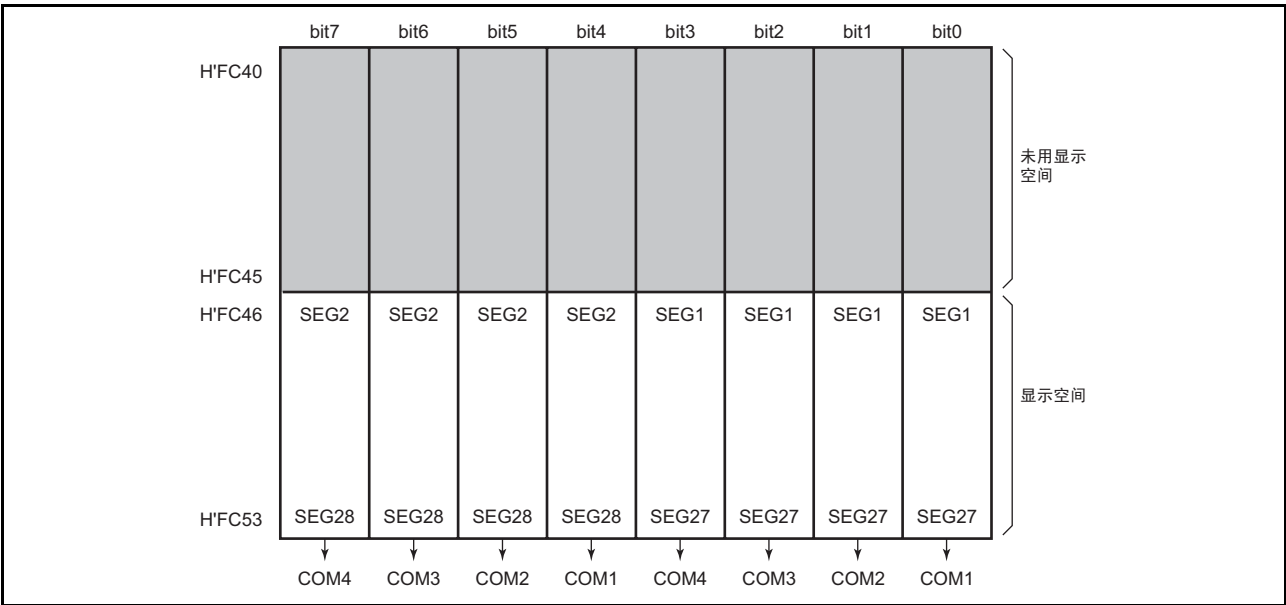


图 14.2 LCD RAM 图 (1/4 占空比)

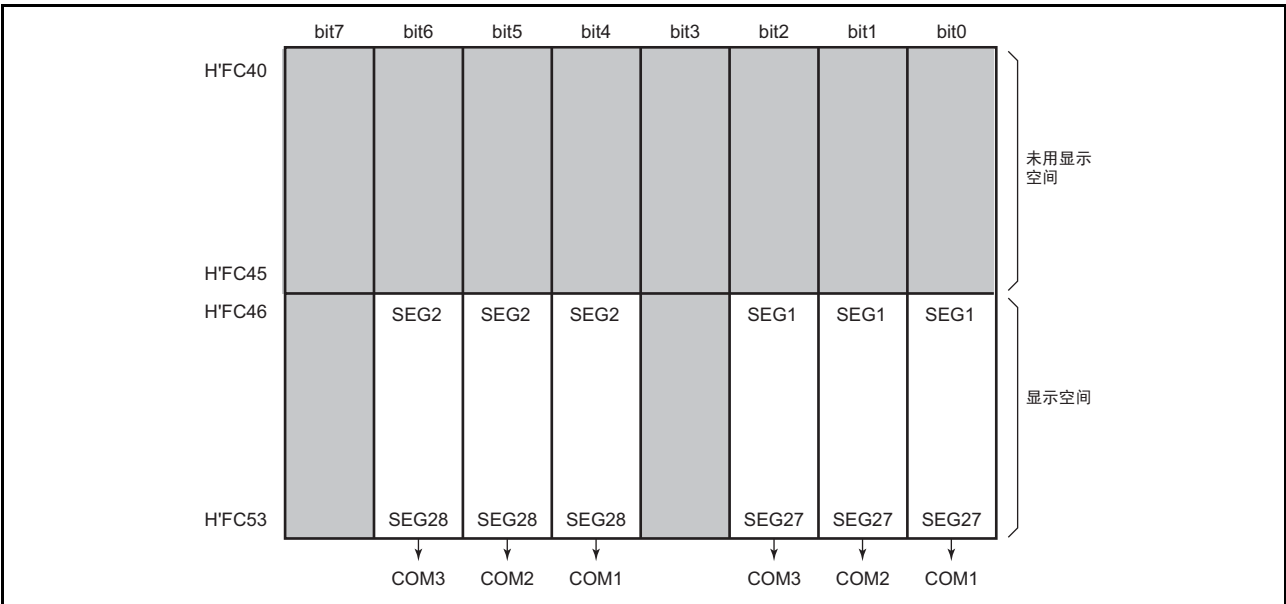


图 14.3 LCD RAM 图 (1/3 占空比)

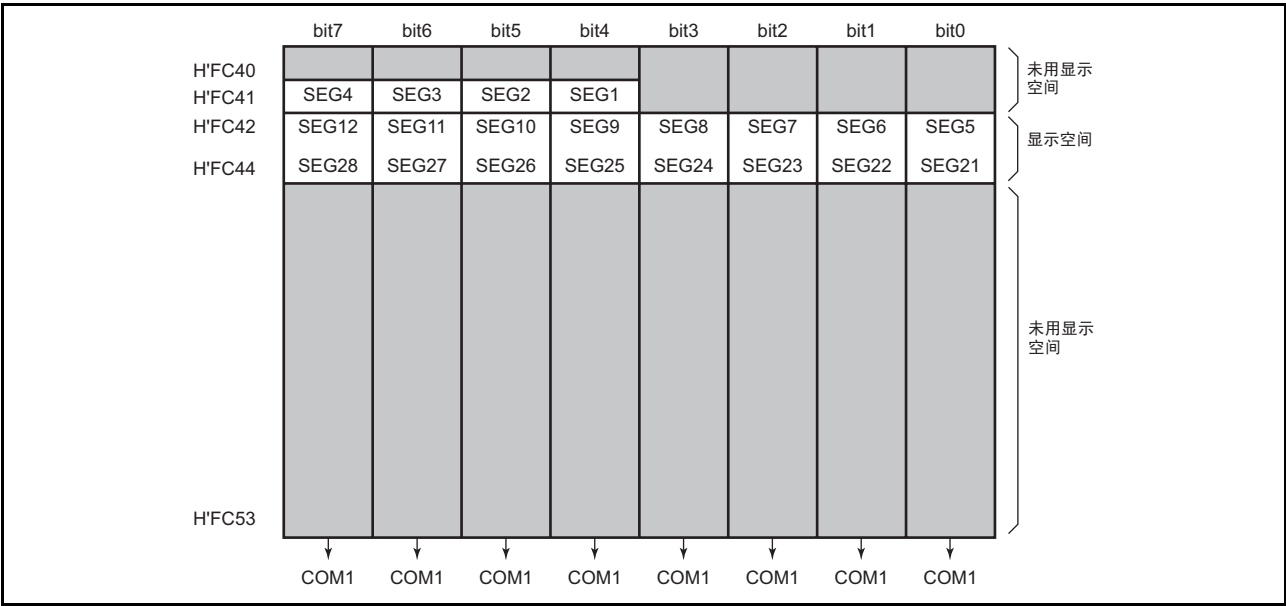


图 14.4 LCD RAM 图 (静态)

(2) HD64F2280RB

LCD RAM 和显示段的关系因占空比的不同而不同。占空比对应的 LCD RAM 映射如图 14.5 ~ 图 14.7 所示。

设定了显示所必须的寄存器后，在占空比对应的部分，用和通用 RAM 相同的指令写入数据，当将状态转换为 ON 时就可自动显示。RAM 可使用字 / 字节存取命令来设定。

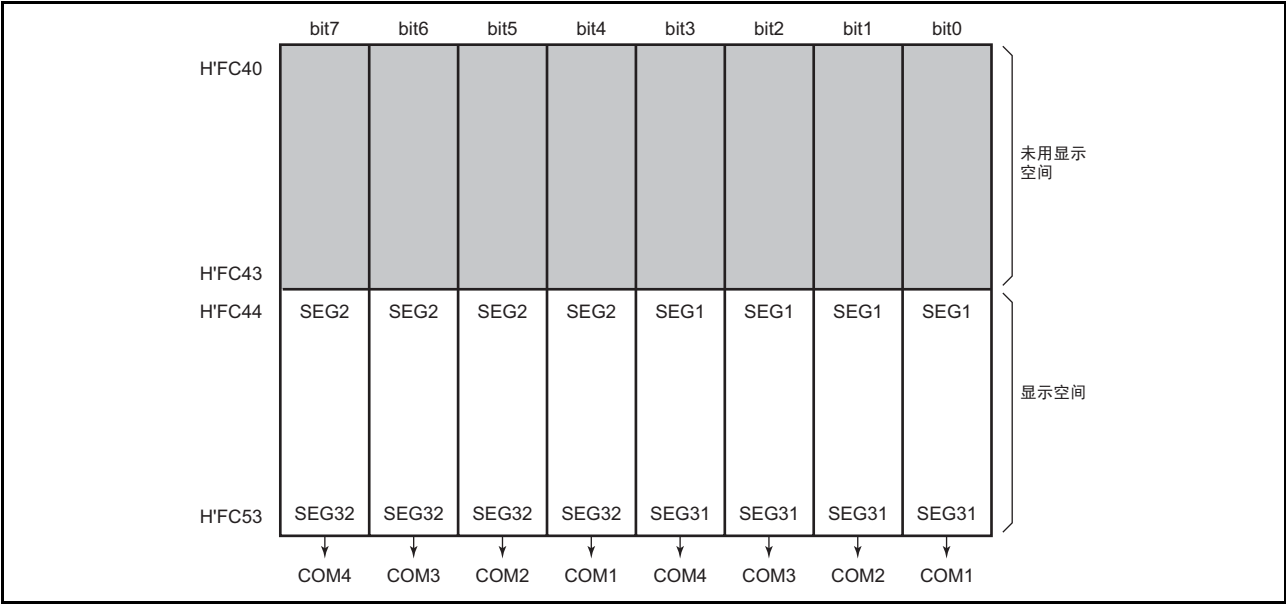


图 14.5 LCD RAM 图 (1/4 占空比)

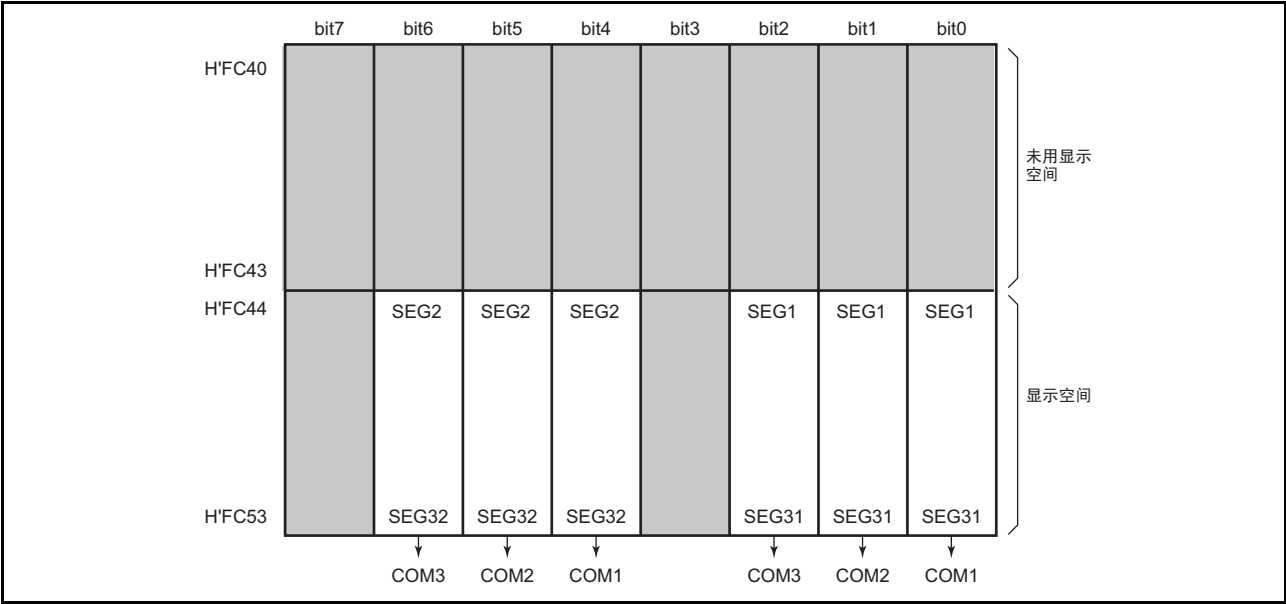


图 14.6 LCD RAM 图 (1/3 占空比)

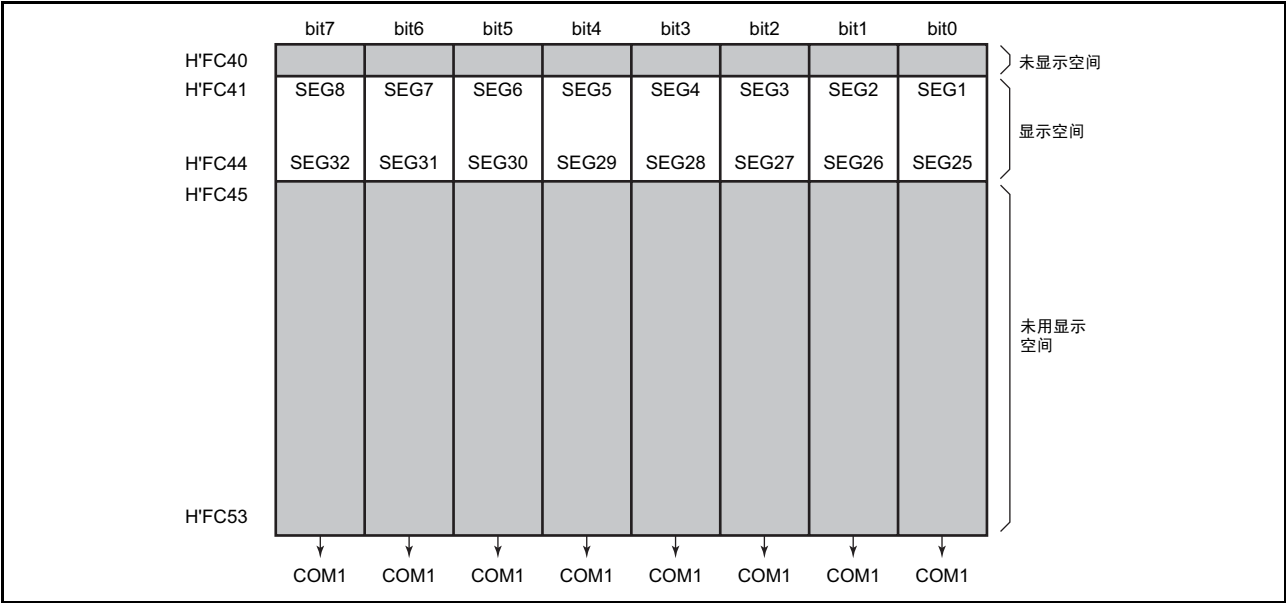


图 14.7 LCD RAM 图 (静态)

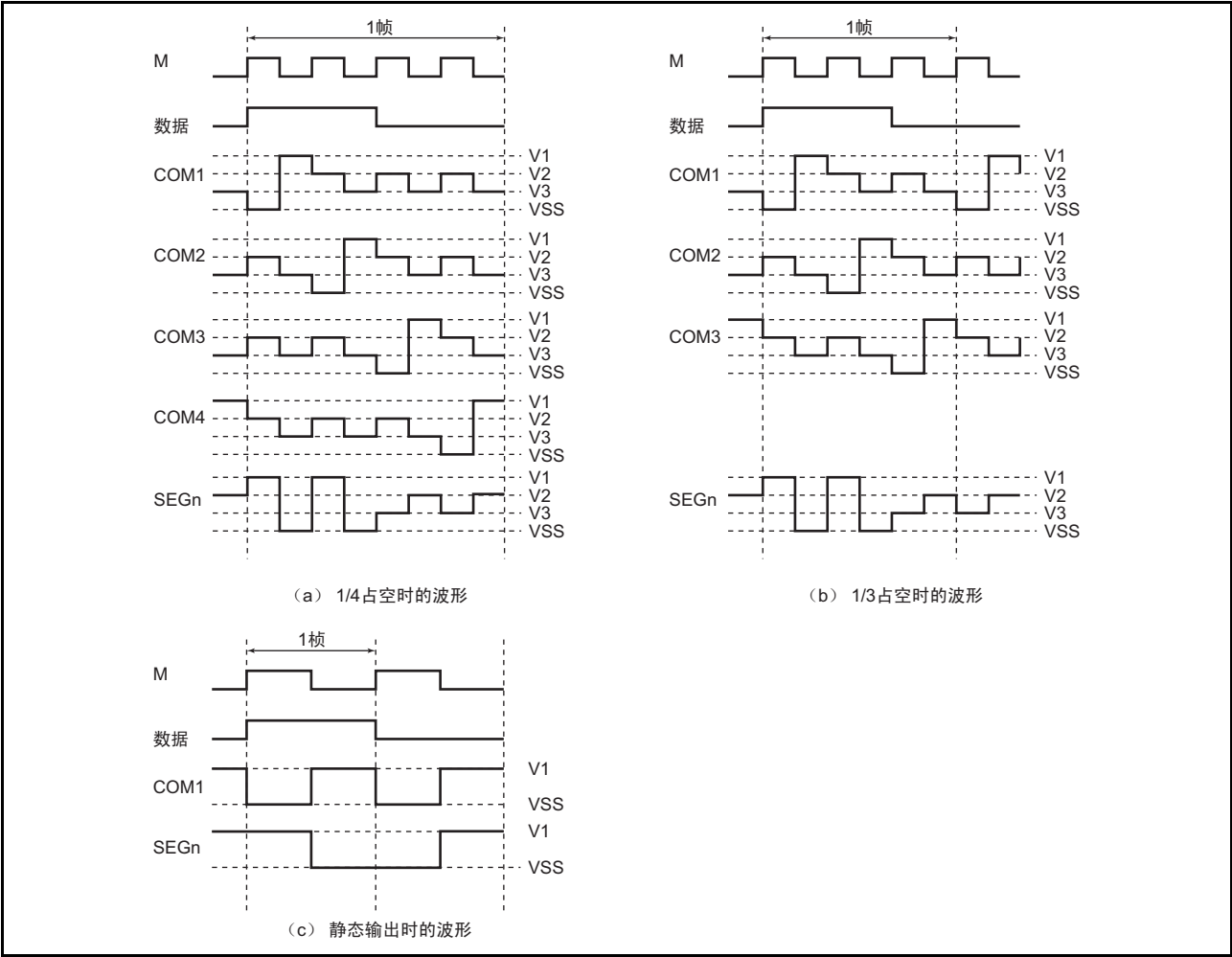


图 14.8 各占空比的输出波形 (A 波形)

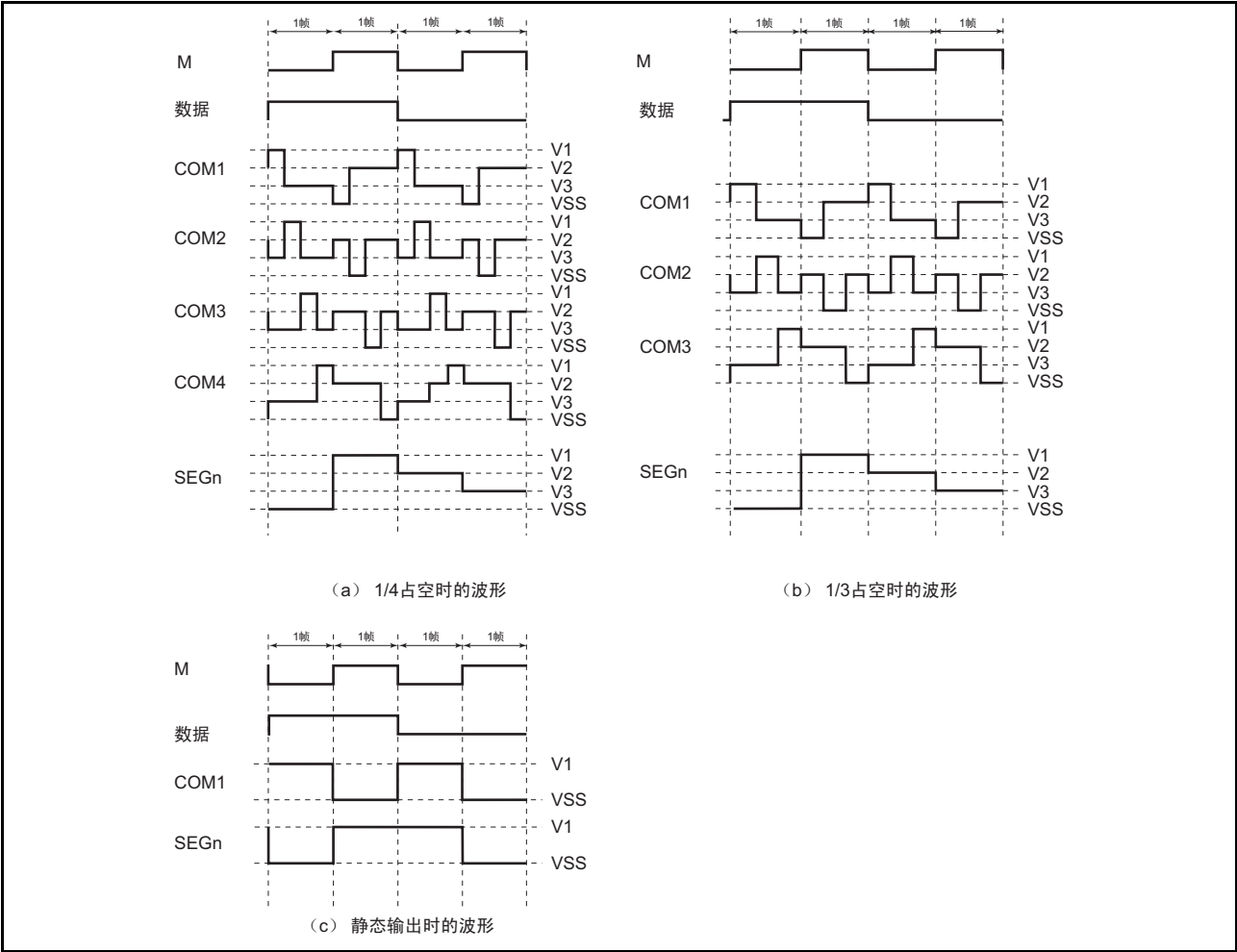


图 14.9 各种占空比的输出波形 (B 波形)

表 14.5 输出电平的关系 (A 波形)

数据		0	0	1	1
M		0	1	0	1
静态	公共输出	V1	VSS	V1	VSS
	段输出	V1	VSS	VSS	V1
1/3 占空比	公共输出	V3	V2	V1	VSS
	段输出	V2	V3	VSS	V1
1/4 占空比	公共输出	V3	V2	V1	VSS
	段输出	V2	V3	VSS	V1

14.4.3 低功耗模式的运行

在低功耗模式下 LCD 控制器 / 驱动器也可以运行。低功耗模式下，LCD 控制器 / 驱动器运行状态如表 14.6 所示。

虽然在中速模式下不能读 / 写 LCD 寄存器，但 LCD 显示的持续运行与高速模式是相同的。在子激活模式、子睡眠模式及监视模式下，系统时钟要切换为子时钟必须对 SUB、SUB/2 或 SUB/4 进行选择。特别是在监视模式下，没有选择 SUB、SUB/2 或 SUB/4，就不能提供时钟，将会导致显示停止。在这种情况下液晶板也可能需要消耗直流电压。因此必须选择 SUB、SUB/2 或 SUB/4。

当增强了 LCD 驱动电源，在软件待机模式下，段输出和公共输出管脚将会保留它们的值，同时 LCD 液晶板也可应用直流电压。因此，将用于段输出和公共输出端口的 DDR 和 SGS3 ~ SGS0 位全部置 0 之后，再进入软件待机模式。

表 14.6 低功耗模式和显示运行的关系

模式		复位	激活	睡眠	监视	子激活	子睡眠	待机	模块备用
时钟	ϕ	运行	运行	运行	停止	停止	停止	停止 *1	停止 *1
	ϕ_{SUB}	运行	运行	运行	运行	运行	运行	停止 *1	停止 *1
显示运行	ACT=0	停止	停止	停止	停止	停止	停止	停止 *2	停止
	ACT=1	停止	显示	显示	显示 *3	显示 *3	显示 *3	停止 *2	停止

【注】 *1 供给 LCD 的时钟停止。

*2 关闭与 PSW 位设置无关的 LCD 驱动电源。

*3 如果使用时钟没有选择 SUB、SUB/2 或 SUB/4，不运行显示。

14.4.4 增强 LCD 驱动电源

驱动液晶板时，内部电源容量可能会不足，在这种情况下必须降低电源电阻。有两种方法作可以处理此情况，在 V1 ~ V3 管脚连接 0.1 ~ 0.3 μ F 的旁路电容器，如图 14.10 所示，或外置新的分压电阻。

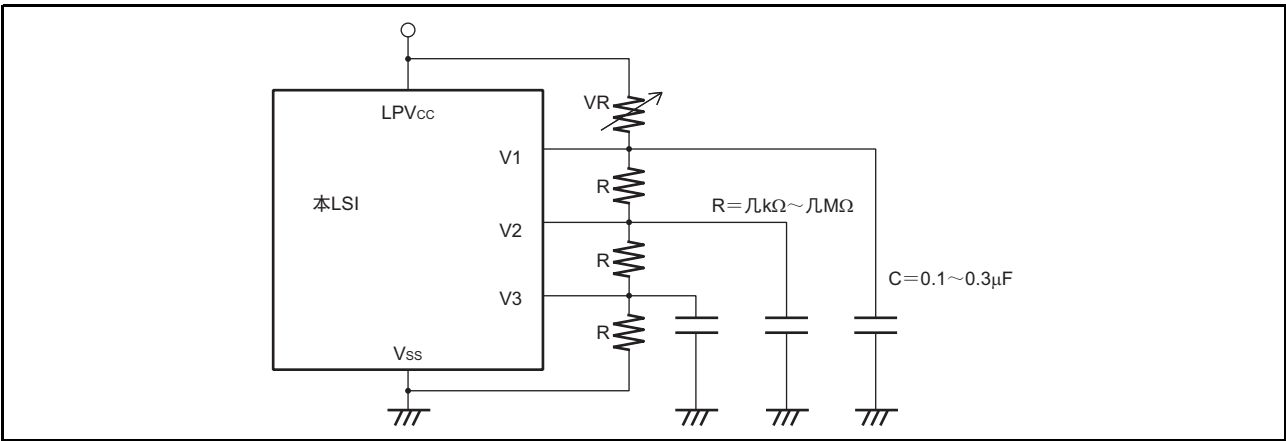


图 14.10 连接外部分压电阻

14.5 使用时的注意事项

14.5.1 禁止 LCD 显示

要关闭 LCD 输出，应先通过 SGS 位将管脚从 SEG 切换到运行端口，然后将 LCR 寄存器的 ACT 位清零。如果 ACT 位清零的同时管脚仍然设置为 SEG 输出，直流电压会直接作用到液晶板上。

第 15 章 RAM

H8S/2282 群有 4k 字节的内部高速静态 RAM，而 H8S/2280 群有 2k 字节。RAM 通过 16 位的数据总线与 CPU 连接，字节数据和字符数据都能通过 CPU 进行 1 个状态的存取。

可通过控制系统控制寄存器（SYSCR）的 RAME 位的方法对内部 RAM 进行激活或无效的转换。关于 SYSCR 的详细信息，请参照「3.2.2 系统控制寄存器（SYSCR）」。

产品类型		ROM 型	RAM 容量	RAM 地址
H8S/2282 群	HD64F2282	闪存型	4K 字节	H'FFE000 ~ H'FFEFBF、 H'FFFFC0 ~ H'FFFFFF
	HD6432282	掩模型 ROM 版	4K 字节	H'FFE000 ~ H'FFEFBF、 H'FFFFC0 ~ H'FFFFFF
	HD6432281		4K 字节	H'FFE000 ~ H'FFEFBF、 H'FFFFC0 ~ H'FFFFFF
H8S/2280 群	HD64F2280B	闪存型	2K 字节	H'FFE800 ~ H'FFEFBF、 H'FFFFC0 ~ H'FFFFFF
	HD64F2280RB		2K 字节	H'FFE800 ~ H'FFEFBF、 H'FFFFC0 ~ H'FFFFFF

第 16 章 快速擦写存储器（F-ZTAT 版）【H8S/2282 群】

快速擦写存储器的特点概述如下。

快速擦写存储器的框图如图 16.1 所示。

16.1 特点

- 容量：128k 字节
- 编程 / 擦除方式
编程方式为 128 字节同时写入。擦除是以块为单位进行擦除。快速擦写存储器由 32k 字节 ×2 块、28k 字节 ×1 块、16k 字节 ×1 块、8k 字节 ×2 块、1k 字节 × 4 块构成。全部擦除时也要按块逐个擦除。
- 重复编程能力
最多可重复编程 100 次。
- 2 种板载编程模式
引导模式
用户编程模式
引导模式下通过启动内置的引导程序，全部擦除或者写入快速擦写存储器，可以完成板载编程 / 擦除。正常用户模式下可以进行分块擦除和改写。
- 位速率自动匹配
引导模式下传送数据时，LSI 的位速率会自动调整使之与主机的传送位速率相匹配。
- 编程 / 擦除保护
可通过设定硬件保护、软件保护、错误保护对快速擦写存储器的编程 / 擦除进行保护。
- 程序器模式
除板载编程以外，还可使用 PROM 编程器进行编程 / 擦除。

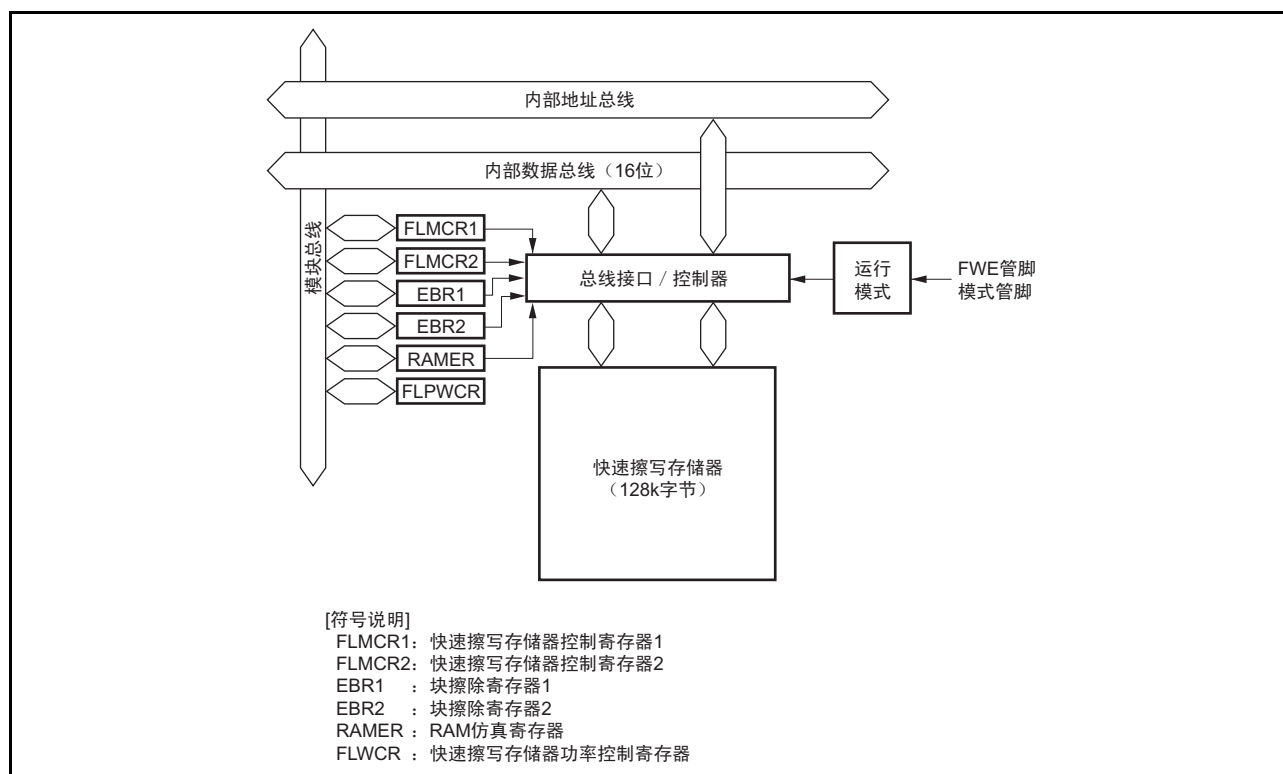


图 16.1 快速擦写存储器的框图

16.2 模式转换

在复位状态，设定模式管脚和 FWE 管脚开始复位，本 LSI 将进入一个运行模式如图 16.2 所示。用户模式下，虽然可从快速擦写存储器进行读出操作，但是不能编程 / 擦除。可以进行快速擦写存储器编程 / 擦除操作的有引导模式、用户编程模式、程序器模式。

引导模式和用户编程模式的不同点如表 16.1 所示。引导模式如图 16.3 所示，用户编程模式如图 16.4 所示。

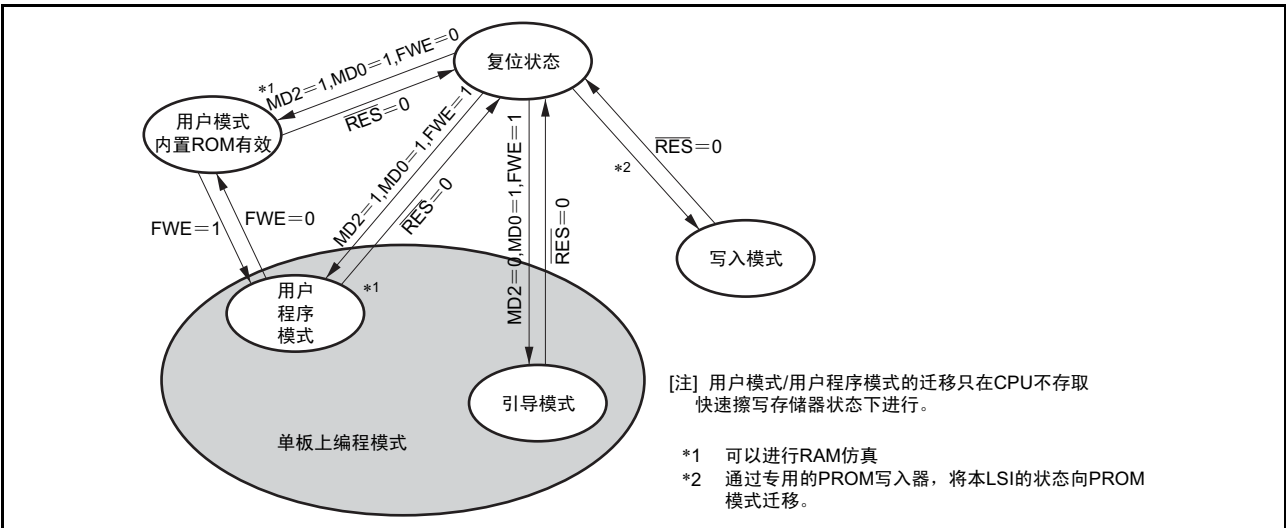


图 16.2 关于快速擦写存储器的状态转换

表 16.1 引导模式和用户编程模式的不同点

	引导模式	用户编程模式
全部擦除	○	○
块分割擦除	X	X
改写控制程序 *	程序 / 程序验证	程序 / 程序验证 擦除 / 擦除验证 仿真

【注】 * 用户需注意，应按照推荐的运算法则计算。

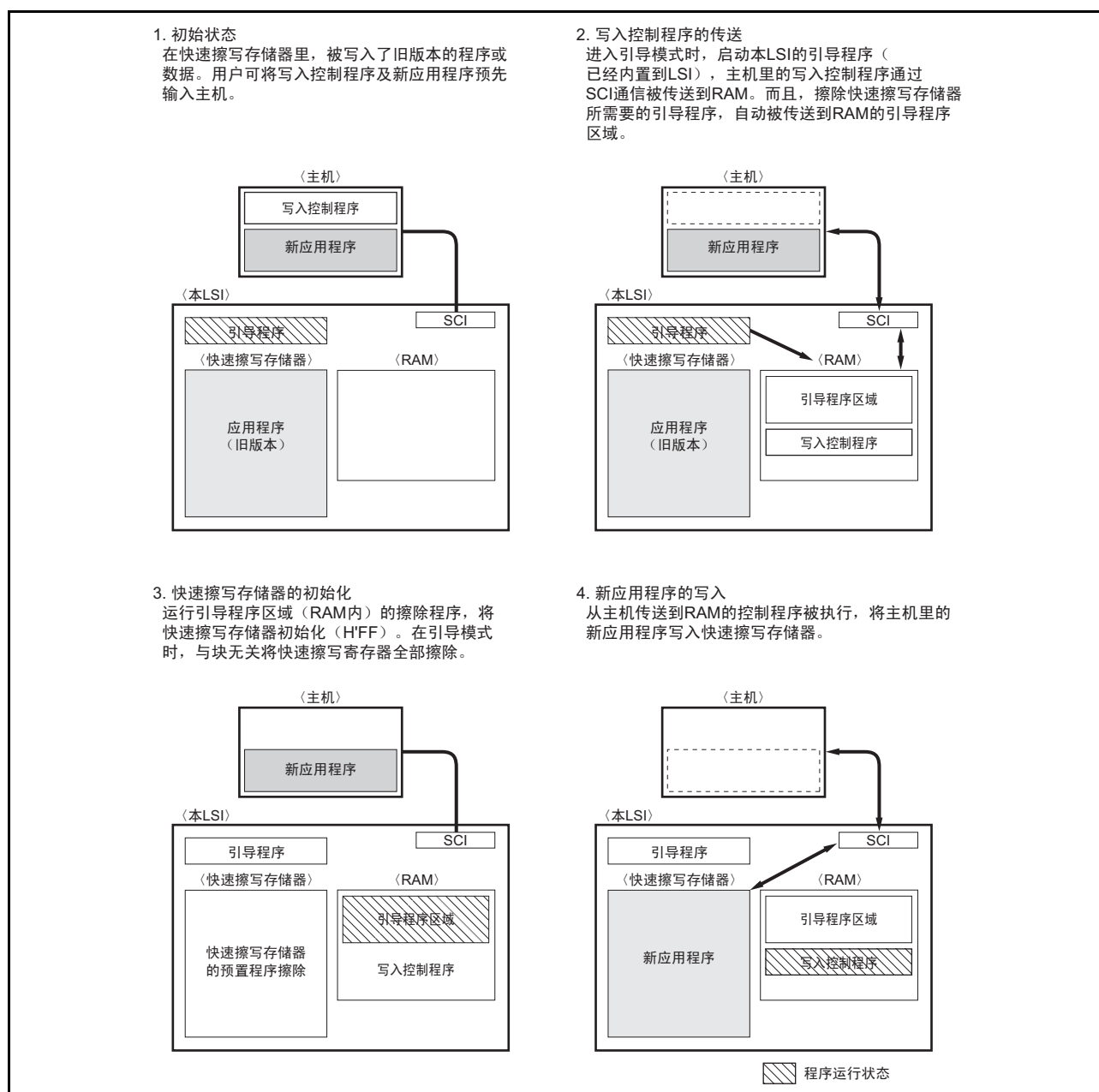
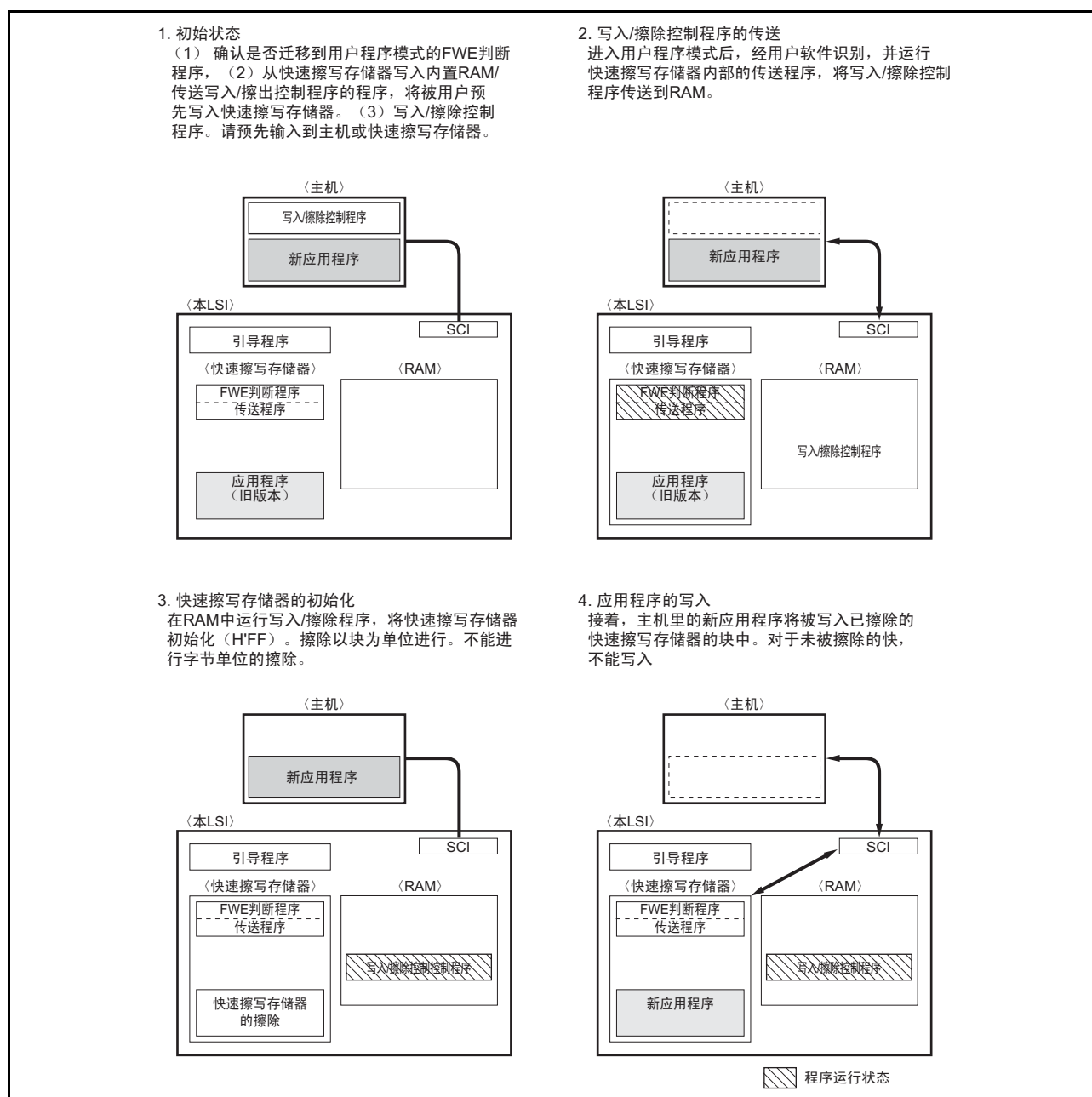


图 16.3 引导模式



16.3 块结构

128k 字节快速擦写存储器的块结构如图 16.5 所示。粗线边框表示擦除块。细线边框表示写入的单位，框内的数值表示地址。快速擦写存储器被分成 32k 字节（2 块）、28k 字节（1 块）、16k 字节（1 块）、8k 字节（2 块）、1k 字节（4 块），在这些单位中进行擦除操作。写入操作是在以低位地址 H'00 或 H'80 开始的 128 字节为单元进行的。

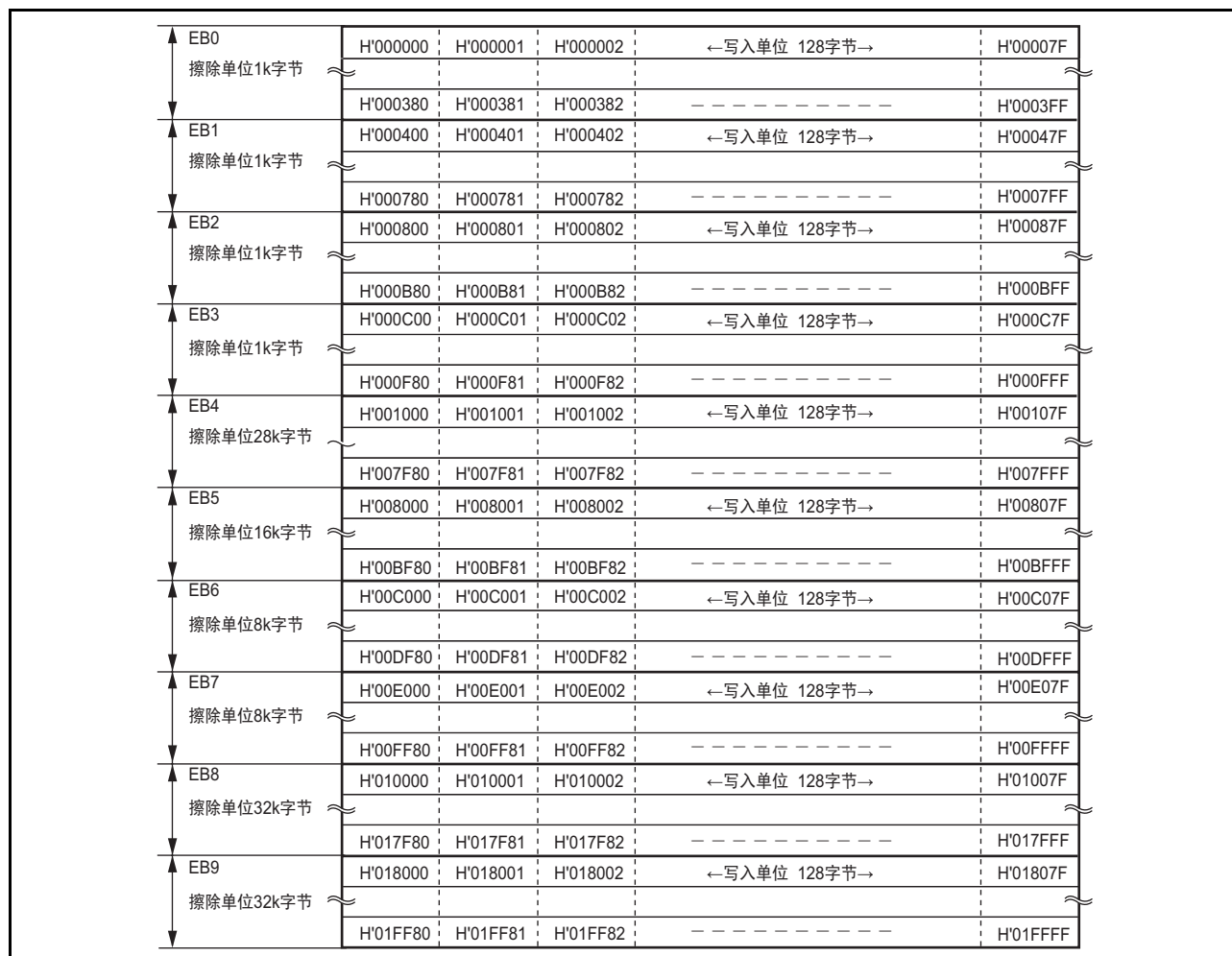


图 16.5 快速擦写存储器的块结构

16.4 输入 / 输出管脚

快速擦写存储器由表 16.2 所示的管脚控制。

表 16.2 管脚结构

管脚名	输入 / 输出	功能
RES	输入	复位
FWE	输入	硬件保护 FLASH 存储器的编程 / 擦除
MD2	输入	设定本 LSI 的运行模式
MD0	输入	设定本 LSI 的运行模式
TxD	输入	串行发送数据输入
RxD	输入	串行接受数据输入

16.5 寄存器的说明

快速擦写存储器有以下几种寄存器。

- 快速擦写存储器控制寄存器 1 (FLMCR1)
- 快速擦写存储器控制寄存器 2 (FLMCR2)
- 擦除块指定寄存器 1 (EBR1)
- 擦除块指定寄存器 2 (EBR2)
- RAM 仿真寄存器 (RAMER)
- 快速擦写存储器功率控制寄存器 (FLPWCR)

16.5.1 快速擦写存储器控制寄存器 1 (FLMCR1)

FLMCR1 可使快速擦写存储器在程序模式、程序验证模式、擦除模式、擦除验证模式这几个模式中转换。关于具体的设定方法，请参照 [16.8 快速擦写存储器的编程 / 擦除]。

位	位名称	初始值	R/W	说明
7	FWE	—	R	设置 FWE 管脚的输入电平。FWE 管脚低电平时为 0，高电平时为 1。
6	SWE	0	R/W	软件写入允许 此位置 1 时，能够编程 / 擦除 FLASH 存储器。为 0 时，此寄存器的其它位和 EBR1 的各位均不能置位。
5	ESU	0	R/W	擦除设置 如果置 1，变成擦除设置状态，清 0 则解除设置状态。
4	PSU	0	R/W	程序设置 如果置位 1，变成程序设置状态，清 0 则解除设置状态。请将 FLMCR1 的 P 位置为 1 之前，进行此位设置。
3	EV	0	R/W	擦除验证 如果置位 1，则转换到擦除验证模式，清 0，解除擦除验证模式。
2	PV	0	R/W	程序验证 如果置位 1，则转换到程序验证模式，清 0，解除程序验证模式。
1	E	0	R/W	擦除 在 SWE=1、ESU=1 的状态下，如果将此位置 1，则转换到擦除模式，清 0，解除擦除模式。
0	P	0	R/W	程序 在 SWE=1、PSU=1 的状态，如果将此位置 1，则转换到程序模式，清 0，解除程序模式。

16.5.2 快速擦写存储器控制寄存器 2 (FLMCR2)

FLMCR2 显示快速擦写存储器的编程 / 擦除状态。FLMCR2 是只读专用寄存器，不能写入。

位	位名称	初始值	R/W	说明
7	FLER	0	R	对 FLASH 存储器的一个操作进行编程 / 擦除的错误检测，变为错误保护状态时此位被置 1。 详细内容请参照「16.9.3 错误保护」
6~0	—	0	R	保留位 读取时通常读出 0。

16.5.3 擦除块指定寄存器 1、2（EBR1、EBR2）

EBR1,EBR2 指定为快速擦写存储器的擦除块的寄存器。FLMCR1 的 SWE 位置 0 时，EBR1 和 EBR2 被初始化为 H'00。对于 EBR1、EBR2 不要同时置 1。这样将会导致 EBR1 和 EBR2 自动清 0。

• EBR1

位	位名称	初始值	R/W	说明
7	EB7	0	R/W	此位为 1 时，EB7（H'00E000 ~ H'00FFFF）的 8k 字节变为擦除对象。
6	EB6	0	R/W	此位为 1 时，EB6（H'00C000 ~ H'00DFFF）的 8k 字节变为擦除对象。
5	EB5	0	R/W	此位为 1 时，EB5（H'008000 ~ H'00BFFF）的 16k 字节变为擦除对象。
4	EB4	0	R/W	此位为 1 时，EB4（H'001000 ~ H'007FFF）的 28k 字节变为擦除对象。
3	EB3	0	R/W	此位为 1 时，EB3（H'000C00 ~ H'000FFF）的 1k 字节变为擦除对象。
2	EB2	0	R/W	此位为 1 时，EB2（H'000800 ~ H'000BFF）的 1k 字节变为擦除对象。
1	EB1	0	R/W	此位为 1 时，EB1（H'000400 ~ H'0007FF）的 1k 字节变为擦除对象。
0	EB0	0	R/W	此位为 1 时，EB0（H'000000 ~ H'0003FF）的 1k 字节变为擦除对象。

• EBR2

位	位名称	初始值	R/W	说明
7~2	—	全为 0	R/W	保留位 读取时通常读出 0。
1	EB9	0	R/W	此位为 1 时，EB 9（H'018000 ~ H'01FFFF）的 32k 字节变为擦除对象。
0	EB8	0	R/W	此位为 1 时，EB 8（H'010000 ~ H'017FFF）的 32k 字节变为擦除对象。

16.5.4 RAM 仿真寄存器（RAMER）

在模拟快速擦写存储器的实时写入时，RAMER 指定快速擦写存储器区域与 RAM 部分交迭。应该在用户模式、用户编程模式进行 RAMER 的设定。为了能够确实运行仿真功能，改写本寄存器后请不要立即存取 RAM 仿真的对象 ROM。如果立即存取，会无法保证正常的存取。

位	位名称	初始值	R/W	说明
7、6	—	0	R	保留位 读取时通常读出 0。
5、4	—	0	R/W	保留位 写入时务必为 0。
3	RAMS	0	R/W	RAM 选择 是通过 RAM 选择 FLASH 存储器的仿真位。此位为 1 时，RAM 部分与 FLASH 存储器交迭，FLASH 存储器变为所有块的编程 / 擦除保护状态。
2	RAM2	0	R/W	FLASH 存储器区域选择 RAMS 为 1 时，选择与 RAM 的 H'FFE000 ~ H'FFE3FF 区域交迭的 FLASH 存储器区域。这部分区域对应 1 k 字节的擦除块。 00X: H'000000 ~ H'0003FF（EB0） 01X: H'000400 ~ H'0007FF（EB1） 10X: H'000800 ~ H'000BFF（EB2） 11X: H'000C00 ~ H'000FFF（EB3）
1	RAM1	0	R/W	
0	RAM0	0	R/W	

【注】X：与设定无关

16.5.5 快速擦写存储器功率控制寄存器（FLPWCR）

在 LSI 转换到子激活模式时，FLPWCR 可以激活 / 禁止快速擦写存储器低功耗模式。请参照「16.12 快速擦写存储器和低功耗状态」。

位	位名称	初始值	R/W	说明
7	PDWND	0	R/W	低功耗禁止 此位为 1 时，禁止向 FLASH 存储器的低功耗模式转换。
6~0	—	全为 0	R	保留位 读取时通常读出 0。

16.6 板载编程

快速擦写存储器的编程 / 擦除模式有两种：可以进行板载编程 / 擦除的引导模式和用 PROM 编程器进行编程 / 擦除的程序器模式。除此之外，而用户模式也可以进行板载编程 / 擦除。在复位模式下开始复位时，LSI 通过对 MD 管脚和 FWE 管脚的设定来转换模式，如图 16.3 所示。在复位结束之前，各管脚的输入电平必须被定义为 4 个状态。

当转换到引导模式时，启动嵌入 LSI 内部的引导程序。引导程序通过 SCI_1，从连接在外部的将写入控制程序传送到内部 RAM，在全部擦除快速擦写存储器之后，执行写入控制程序。可用于板载状态下的初始写入或不能在用户模式下进行编程 / 擦除的强制恢复。用户编程模式下，通过用户准备好的编程 / 擦除的分支控制程序，就可以擦除 / 写入个别块。

表 16.3 编程模式选择方法

MD2	MD0	FWE	复位解除后的 LSI 状态
1	1	1	用户模式
0	1	1	引导模式

16.6.1 引导模式

从引导模式的复位解除转移至写入控制程序的过程如表 16.4 所示。

1. 引导模式下，快速擦写存储器的写入控制程序必须作为主机的预指令准备好。如何准备一个写入控制程序应与「16.8 快速擦写存储器的编程 / 擦除」内容一致。
2. SCI_1 应被设定为异步模式，发送接收格式是“8 位数据、1 停止位、无奇偶校验”。
3. 当引导程序启动时，芯片将会测定在低电平期间从主机连续发送的异步 SCI 通信数据 (H'00)。然后计算从主机传送的位速率，调整 SCI_1 的位速率与主机的位速率相匹配。在 RXD 管脚高电平状态下进行复位解除。根据需要在板上上拉 RXD 管脚及 TXD 管脚。在复位完成后，在芯片能够测定低电平之前需要约 100 个状态。
4. 当位速率的匹配结束，作为调整完成的信号发送 1 个字节的 H'00 给主机，主机如果正常接收到调整完成指令，会发送 1 个字节 H'55 给芯片。无法正常接收时，通过复位重新起动引导模式。这取决于主机的位速率和本 LSI 的系统时钟频率，在主机位速率和芯片位速率之间也存在一些差异。为了 SCI 的适当的运行将主机的传送位速率和本 LSI 的系统时钟频率控制在表 16.5 的范围内。
5. 引导模式下，引导程序占用内部 RAM 的一部分。H'FFE800 ~ H'FFEFBF 地址是可保存主机发送的写入控制程序区域。程序运行向写入控制程序转换前，引导程序区域不能使用。
6. 转移到写入控制程序之前，SCI_1 虽然完成发送接收 (SCR 的 RE = 0、TE = 0)，但是与 BRR 匹配的位速率将被保持，因此可继续用于通过写入控制程序与主机间写入数据或验证数据的发送 / 接收。TXD 管脚变成高电平输出状态。向写入控制程序转移后，CPU 的通用寄存器值不确定。特别是子程序调用会隐藏地使用堆栈指针，所以在写入控制程序的开头必须进行初始化。
7. 引导模式通过复位解除时。需把复位管脚调至低电平，经过至少 20 个状态后，设置 MD 管脚，解除复位。发生 WDT 的溢出时，引导模式被解除。
8. 引导模式中请不要改变 MD 管脚的输入电平。
9. 在向快速擦写存储器进行编程或擦除的过程中，不能使用中断。

表 16.4 引导模式的运行

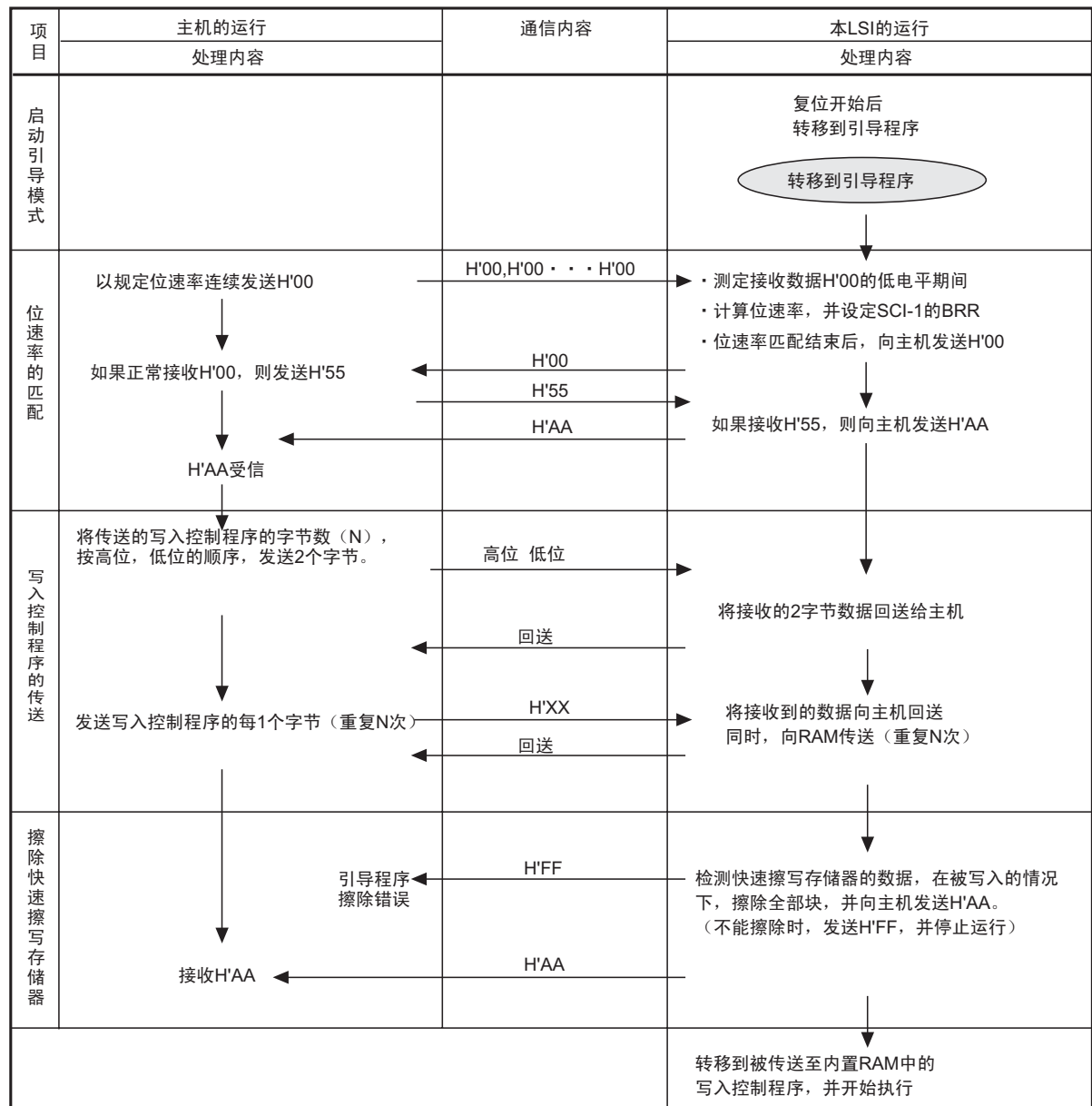


表 16.5 可自动匹配位速率的系统时钟频率

主机位速率	本 LSI 的系统时钟频率范围
19200bps	20MHz
9600bps	8 ~ 20MHz
4800bps	4 ~ 20MHz

16.6.2 用户编程模式的编程 / 擦除

用户模式具有通过转移到用户预备的编程 / 擦除程序，而实现在内部进行擦除或改写用户编程模式的功能。用户需要准备进行转移的条件设定和提供改写数据的方法。并且根据需要，必须事先在快速擦写存储器的一部分输入编程 / 擦除程序或事先写入外部提供的编程 / 擦除程序。由于编程 / 擦除过程中不能读取快速擦写存储器，所以同引导模式相同，将编程 / 擦除程序传送到内部 RAM，再运行。用户模式下编程 / 擦除顺序的范例如图 16.6 所示。编程 / 擦除程序请参考 [16.8 快速擦写存储器的编程 / 擦除] 进行准备。

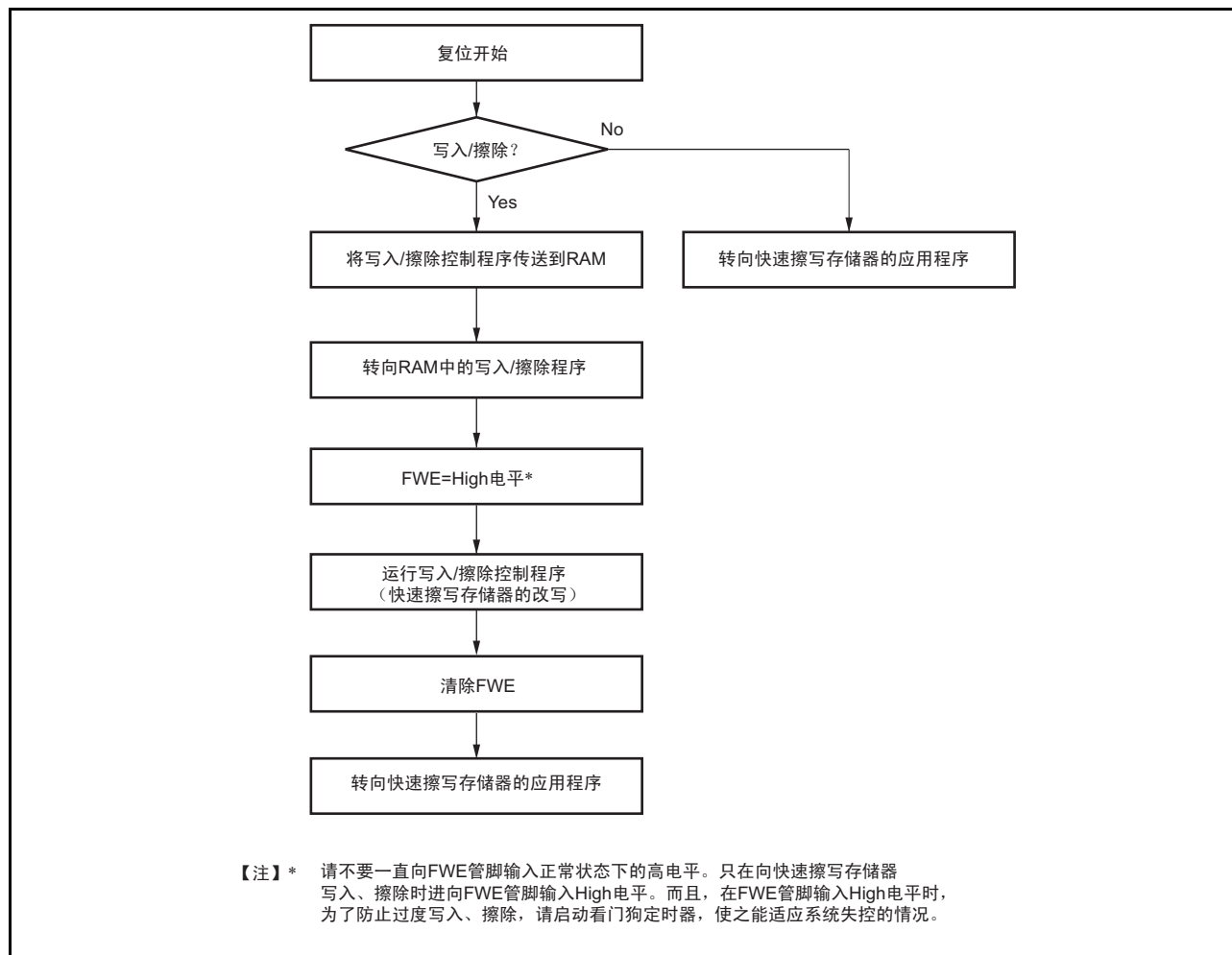


图 16.6 用户模式下编程 / 擦除范例

16.7 快速擦写存储器的 RAM 仿真

为了可以在内置 RAM 中实时仿真对快速擦写存储器的数据改写，可通过设置 RAM 仿真寄存器（RAMER），在快速擦写存储器的一部分交迭使用 RAM。能够仿真的模式是用户模式及用户编程模式。图 16.7 所示的是仿真快速擦写存储器实时改写的样例。

1. 设置 RAMER，在 RAM 上根据实时改写的需要设定交迭区域
2. 使用交迭的 RAM 进行仿真。
3. 改写数据确定后，清除 RAMS 位，解除 RAM 的交迭。
4. 将写入到交迭的 RAM 的数据写入快速擦写存储器空间（EB0）。

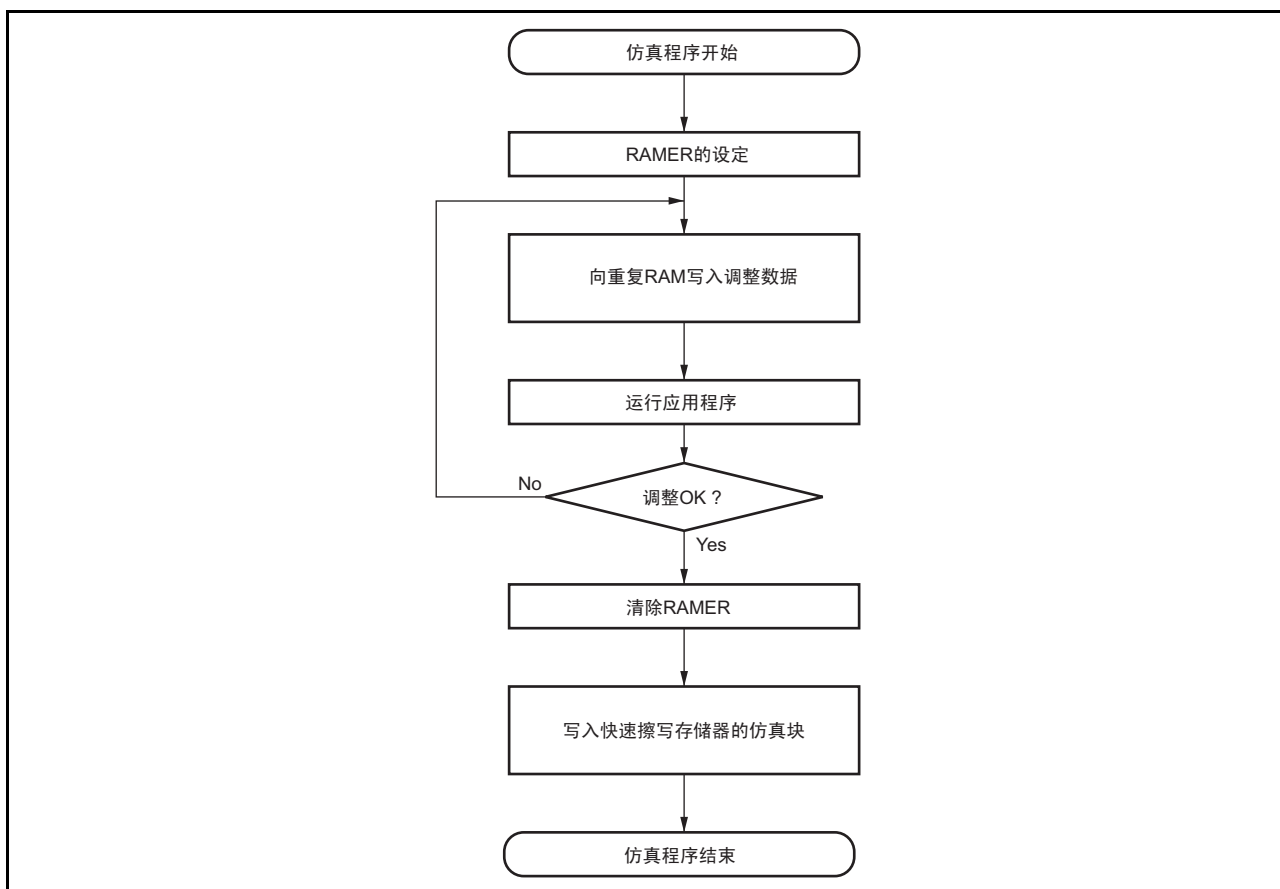


图 16.7 RAM 的仿真流程图

- 一个快速擦写存储器块的交迭 EB0 区域的例子，如图 16.8 所示。
- 1. 交迭的 RAM 区域被固定在 H'FFE000 ~ H'FFE3FF 的 1 k 字节。
 - 2. 可交迭的快速擦写存储器区域是 1 k 字节的 EB0 ~ EB3 内的 1 块，可根据 RAMER 进行选择。
 - 3. 交迭的 RAM 区域，可通过快速擦写存储器内的地址和原本的 RAM 地址进行存取。
 - 4. RAMER 的 RAMS 位置为 1 期间，快速擦写存储器所有块进入编程 / 擦除保护状态（仿真保护），即使设置 FLMCR1 的 P 位或 E 位，也不会转换至编程模式或擦写模式。
 - 5. 即使 RAM 区域按照擦除算法运行程序，也不能执行擦除。
 - 6. 块 EB0 中包含向量表。RAM 仿真时，交迭 RAM 中也需要向量表。

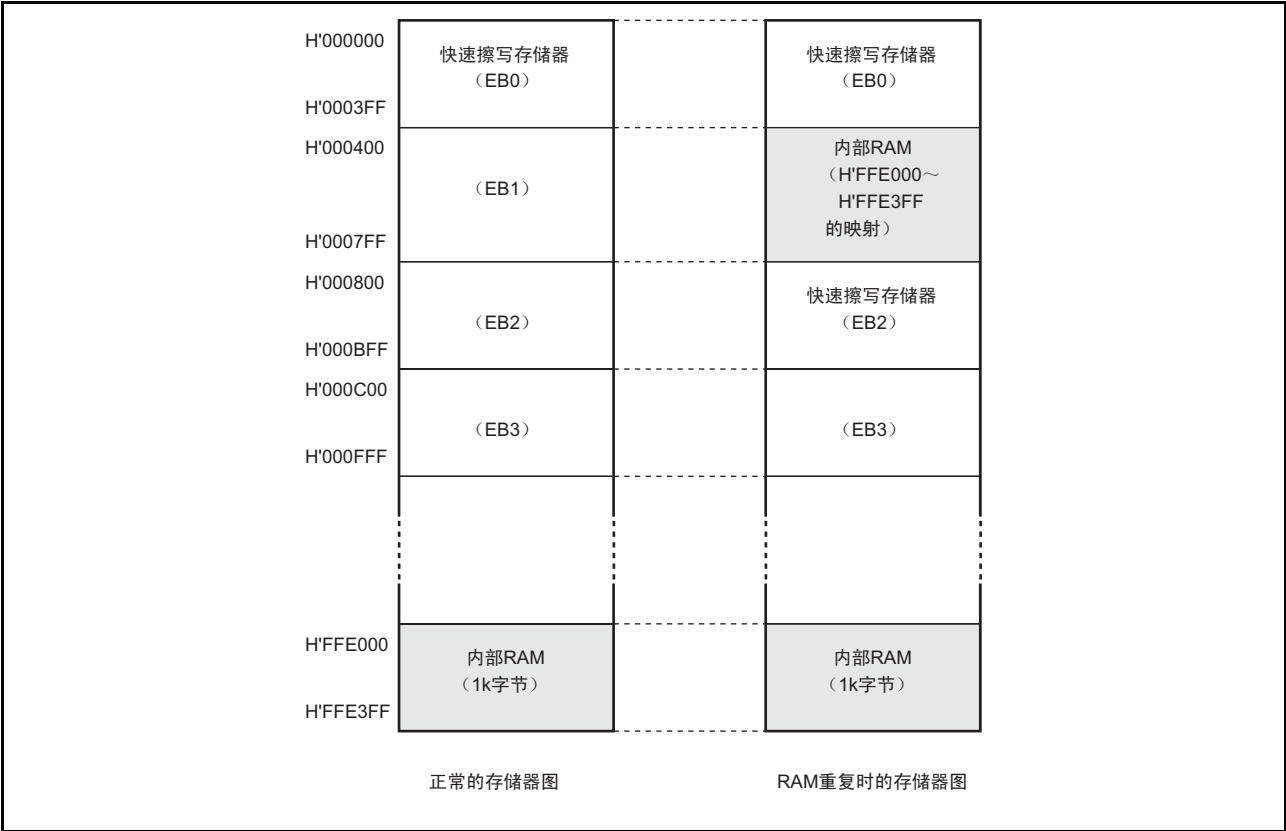


图 16.8 RAM 交迭使用范例

16.8 快速擦写存储器的编程 / 擦除

板载编程模式下快速擦写存储器的编程 / 擦除，采取了使用 CPU 运行软件的方法。快速擦写存储器根据 FLMCR1 的设定，有以下四个运行模式：编程模式、程序验证模式、擦除模式、擦除验证模式。引导模式的写入控制程序、用户编程模式下的编程 / 擦除程序通过这些运行模式可以对编程 / 擦除进行组合执行。快速擦写存储器的写入请按照「16.8.1 程序 / 程序验证」，快速擦写存储器的擦除请按照「16.8.2 擦除 / 擦除验证」进行。

16.8.1 程序 / 程序验证

快速擦写存储器按照如图 16.9 所示的程序 / 程序验证流程进行快速擦写存储器的数据和程序写入。按照此流程进行编程操作可以使数据或程序的写入不受芯片电压的影响和保证数据的可靠性。

1. 写入操作只能在一个空地址上进行，对已经写入的地址不能再次写入。
2. 一次必须写入 128 字节。写入不足 128 字节数据时，也需要向快速擦写存储器传送 128 字节的数据。在这种情况下 HFF 必须被写入到额外的地址中。
3. 在 RAM 中请确保写入数据区域、改写数据区域、追加写入数据区域均为 128 字节。请按照图 16.9 进行改写数据和追加数据的计算。
4. 在改写数据区域或追加写入数据区域，以字节为单位向快速擦写存储器连续传送 128 字节。程序地址和 128 字节的数据被锁存在快速擦写存储器内。传送给快速擦写存储器的起始地址低 8 位应置为 H'00 或 H'80。
5. P 位置 1 的时间为写入时间。写入时间请参考图 16.9。
6. 设定看门狗定时器是为了避免由过剩写入引起程序失控。溢出周期请按照 $(\gamma + z2 + \alpha + \beta) \mu s$ 扩大。
7. 对验证地址的虚写是在低 2 位为 b'00 的地址写入 1 字节 H'FF。可从进行虚写的地址以长字符读出验证数据。
8. 对于同一位地址进行程序 / 程序验证的重复次数请不要超过 (N) 次。

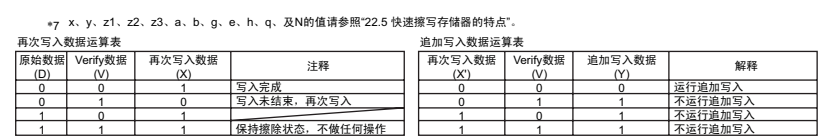


图 16.9 程序 / 程序验证流程图

16.8.2 擦除 / 擦除验证

擦除请参考图 16.10 的擦除 / 擦除验证流程图。

1. 擦除之前，不必进行预写入（将擦除的存储器所有数据清 0）
2. 必须以块为单位进行擦除。使用擦除块寄存器 1（EBR1）、擦除块寄存器 2（EBR2）选择 1 个块进行擦除。要擦除多块时，也是分块依次擦除。
3. E 位被置 1 的时间为擦除时间。
4. 设定看门狗定时器是为了避免由过剩擦除引起的程序失控。溢出周期请按照 $(y + z + \alpha + \beta)$ ms 扩大。
5. 对验证地址的虚写，是在低 2 位为 b'00 的地址写入 1 字节 H'FF。可从进行虚写的地址中以长字符读出验证数据。
6. 未擦除读出数据时，设定为再次擦除模式，同样地反复进行擦除 / 擦除验证的次数请不要超过 (N) 次。

16.8.3 快速擦写存储器编程 / 擦除时的中断

所有的中断，包括 NMI 中断在内，在快速擦写存储器开始编程 / 擦除时或者在引导程序执行时都是不可执行的。具体有以下 3 个原因：

1. 在编程 / 擦除中发生中断，将导致违背编程 / 擦除的法则，将不能确定是否可以正常运行。
2. 向量地址被写入之前或编程 / 擦除中开始进行中断异常处理，不能保证正常向量读取，CPU 会失去控制。
3. 如果在引导程序运行中发生中断，正常顺序引导模式会变得无法执行。

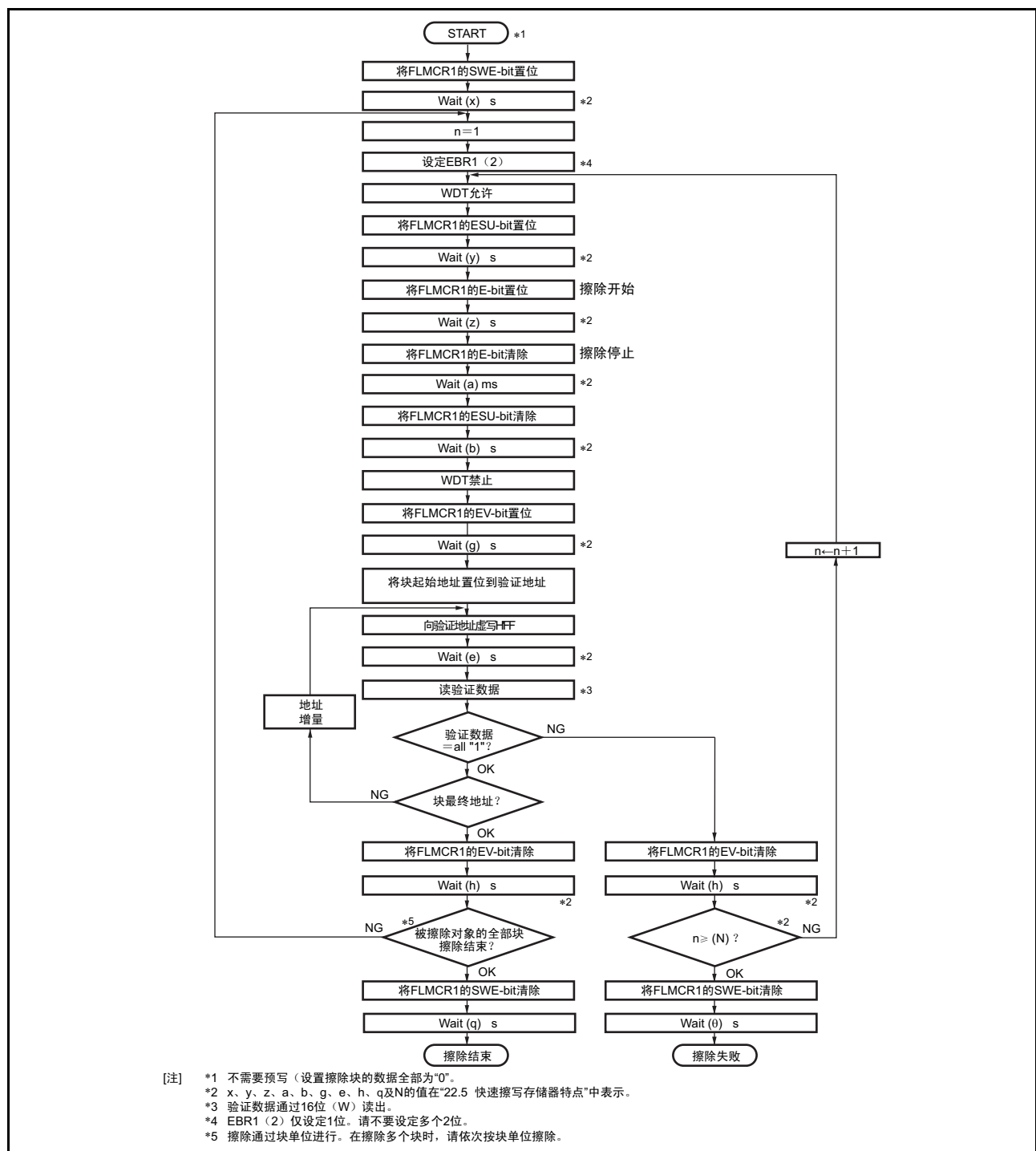


图 16.10 擦除 / 擦除验证流程图

16.9 编程 / 擦除保护

快速擦写存储器的编程 / 擦除保护具有硬件保护、软件保护、错误保护 3 种保护状态。

16.9.1 硬件保护

硬件保护是指由于转换至复位或者待机模式时，而强制禁止或中断对快速擦写存储器进行编程 / 擦除操作的一种状态。快速擦写存储器控制寄存器 1 (FLMCR1)、快速擦写存储器控制寄存器 2 (FLMCR2)、擦除块寄存器 1 (EBR1)、擦除块寄存器 2 (EBR2) 被初始化。加载电源后，通过复位 RES 管脚，直到电磁振荡稳定前，如果 RES 管脚不保持为低电平，就不能变为复位状态。同时，运行中的复位，应该用 AC 特性中规定的 RES 脉冲宽度在 RES 管脚上保持低电平。

16.9.2 软件保护

软件保护可通过清除 FLMCR1 的 SWE 位，对快速擦写存储器的所有块进行反向编程 / 擦除来实现。在软件保护下即使设置 FLMCR1 的 P 位或 E 位，也不能转换至编程模式或擦除模式。并且，根据擦除块指定寄存器 1 (EBR1)、擦除块指定寄存器 2 (EBR2) 的设定，可进行个别块的擦除保护。若将 EBR1、EBR2 设置为 H'00，所有块处在擦除保护状态。

如果设定 RAMER 的 RAMS，所有块处在编程 / 擦除保护状态。

16.9.3 错误保护

错误保护是在快速擦写存储器进行编程 / 擦除时对 CPU 发生失控时或操作没有遵照编程 / 擦除法则的错误侦查，而强制中断编程 / 擦除操作的状态。中断编程 / 擦除运行，可防止由于过剩写入或过剩擦除对快速擦写存储器的损坏。

在向快速擦写存储器编程 / 擦除时，检测出以下错误，FLMCR2 的 FLER 位会被置为 1，进入错误保护状态。

- 在编程 / 擦除过程中相关地址区域的快速擦写存储器被读取时（包含向量读及命令抽取）
- 开始除了编程 / 擦除中的复位以外的异常处理
- 编程 / 擦除过程中执行 SLEEP 指令

此时，FLMCR1、FLMCR2、EBR1、EBR2 的设置虽然被保留，在侦察出错误的时间点，写入模式或擦除模式被强制中断。即使设置 P 位、E 位，也不会转换至编程模式或擦除模式。但是，PV 位、EV 位的设置被激活，因此可以转换至验证模式。只能通过加电复位才能解除错误保护状态。

16.10 PROGRAMMER MODE

PROGRAMMER MODE 下，通过插槽适配器，快速擦写存储器用 PROM 编程器进行编程 / 擦除。使用支持瑞萨科技内置 128k 字节快速擦写存储器内部 MCU 驱动的 PROM 编程器进行写入，设备类型 (FZTAT128V5A)。

16.11 快速擦写存储器的低功耗状态

用户模式下，快速擦写存储器可以在以下任意一种状态运行。

- 正常运行状态
能够进行快速擦写存储器的读 / 写。
- 低功耗状态
中断部分电源电路，LSI 在子时钟状态运行，可读取快速擦写存储器。
- 待机状态
中断快速擦写存储器的所有电路。

LSI 的运行模式与快速擦写存储器的之间的通讯如表 16.6 所示。快速擦写存储器从待机状态恢复到正常运行状态时，电源电路需要一定的稳定周期。包括使用外部时钟时，需要设定 SBYCR 的 STS2 ~ STS0，使恢复到正常运行模式的待机时间为 20 s 以上。

表 16.6 快速擦写存储器的运行状态

本 LSI 的运行模式	FLASH 存储器的状态
高速模式 中速模式 睡眠模式	正常运行状态
子激活模式 子睡眠模式	PDWND = 0 时 低功耗状态（只读） PDWND = 1 时 低功耗状态（只读）
监视模式 软件待机模式 硬件待机模式	待机状态

16.12 快速擦写存储器和低功耗状态

低功耗模式时，不能对（FLMCR1、FLMCR2、EBR1、EBR2、RAMER、FLPWCR）快速擦写存储器寄存器进行读取 / 写入。

第 17 章 快速擦写存储器（F-ZTAT 版）【H8S/2280 群】

快速擦写存储器版中内置的快速擦写存储器有如下特点。

快速擦写存储器的框图如图 17.1 所示。

17.1 特点

- 容量：64K 字节
- 编程 / 擦除方式
编程是 128 字节单位的同时编程方式。擦除以块为单位进行。快速擦写存储器被如下配置：28K 字节 ×1 块，16K 字节 ×1 块，8K 字节 ×2 块，1K 字节 ×4 块。全部擦除时，也必须按块分别擦除。
- 可重复编程性能
可重复编程次数：可达 100 次。
- 编程模式：3 种
引导模式
用户模式
编程器模式
引导模式下通过启动内置的引导程序，全部擦除或者写入快速擦写存储器，可以完成板载写入 / 擦除。正常用户模式下可以进行分块擦除和改写。
- 编程器模式
除板载编程以外，还有使用 PROM 编程器进行编程 / 擦除。
- 位速率自动匹配
在引导模式传送数据时，本 LSI 的位速率能自动地匹配这个主机的传送位速率。
- 编程 / 擦除保护
通过软件可以设定对快速擦写存储器的编程 / 擦除保护。

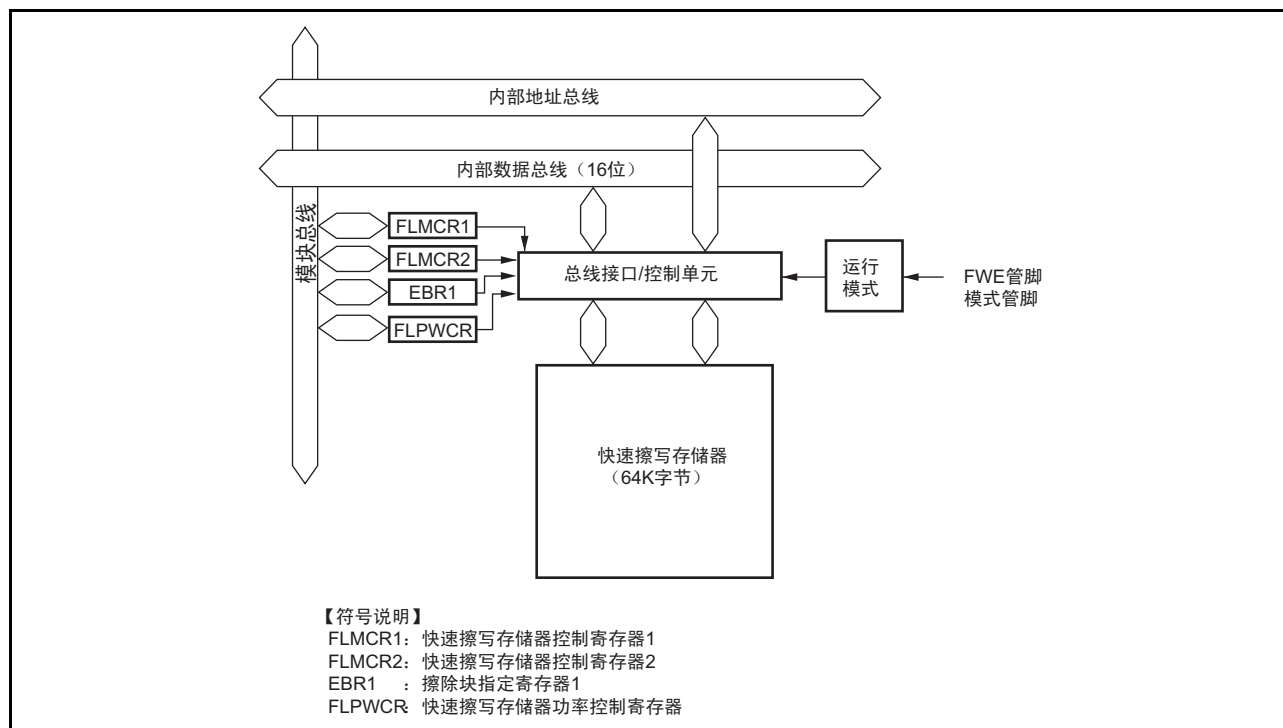


图 17.1 快速擦写存储器的框图

17.2 模式转换

在复位状态下设定模式管脚和 FWE 管脚，开始复位时，本 LSI 转换到如图 17.2 所示的工作模式。在用户模式下可以读取快速擦写存储器，但不可以写入 / 擦除快速擦写存储器。引导模式，用户程序模式，编程器模式都可以作为快速擦写存储器的写入 / 擦除模式。

引导模式和用户程序模式的不同点如表 17.1 所示。引导模式，用户程序模式分别如图 17.3，图 17.4 所示。

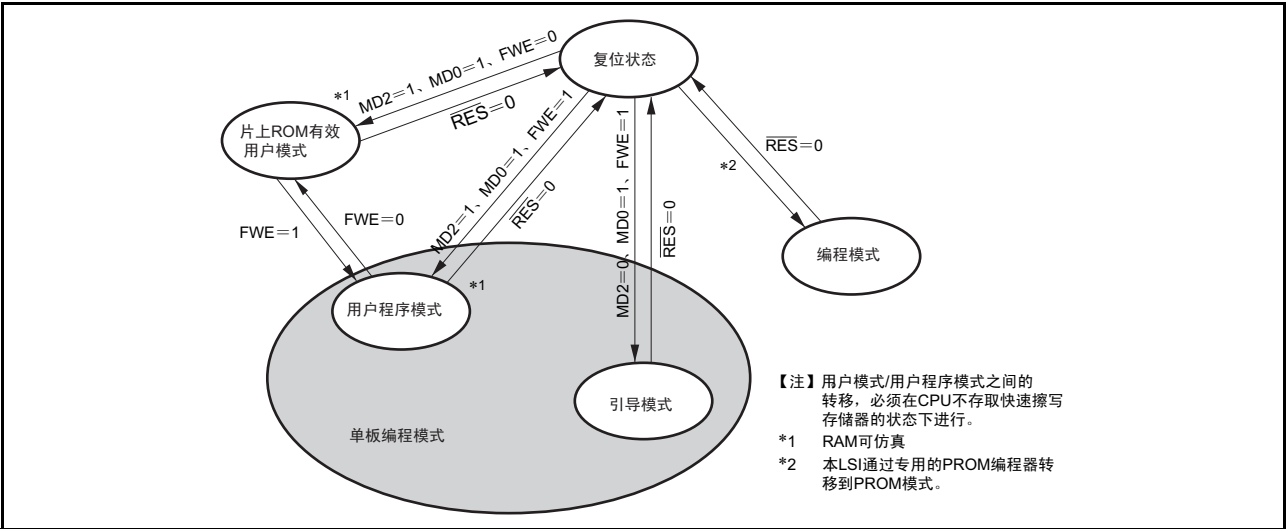
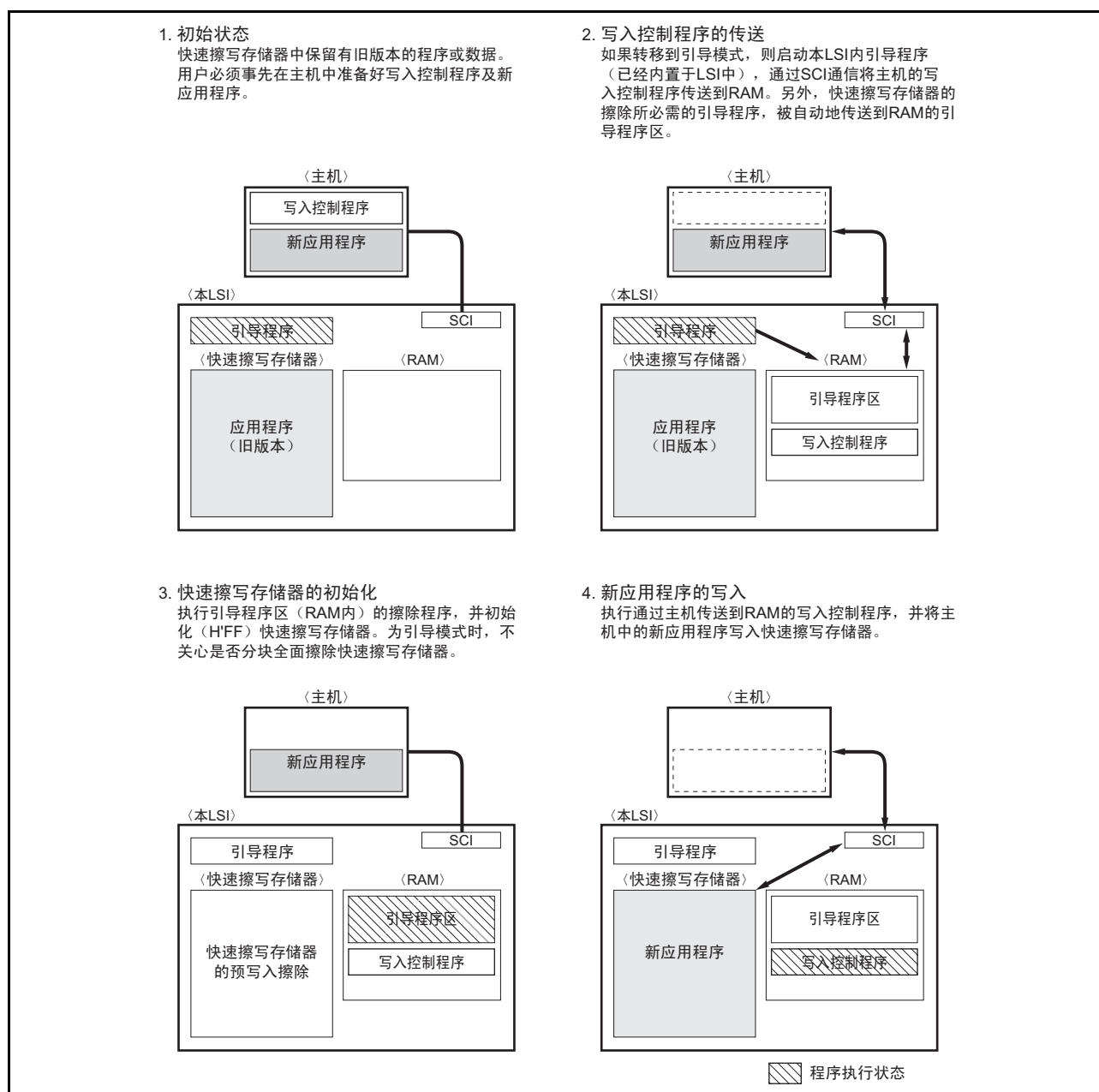


图 17.2 关于快速擦写存储器的状态转换

表 17.1 引导模式和用户程序模式的不同点

	引导模式	用户程序模式
全部擦除	○	○
块分别擦除	X	○
改写控制程序 *	程序 程序验证	擦除 / 擦除验证 程序 / 程序验证 仿真

【注】 * 用户必须按照推荐的算法，准备。



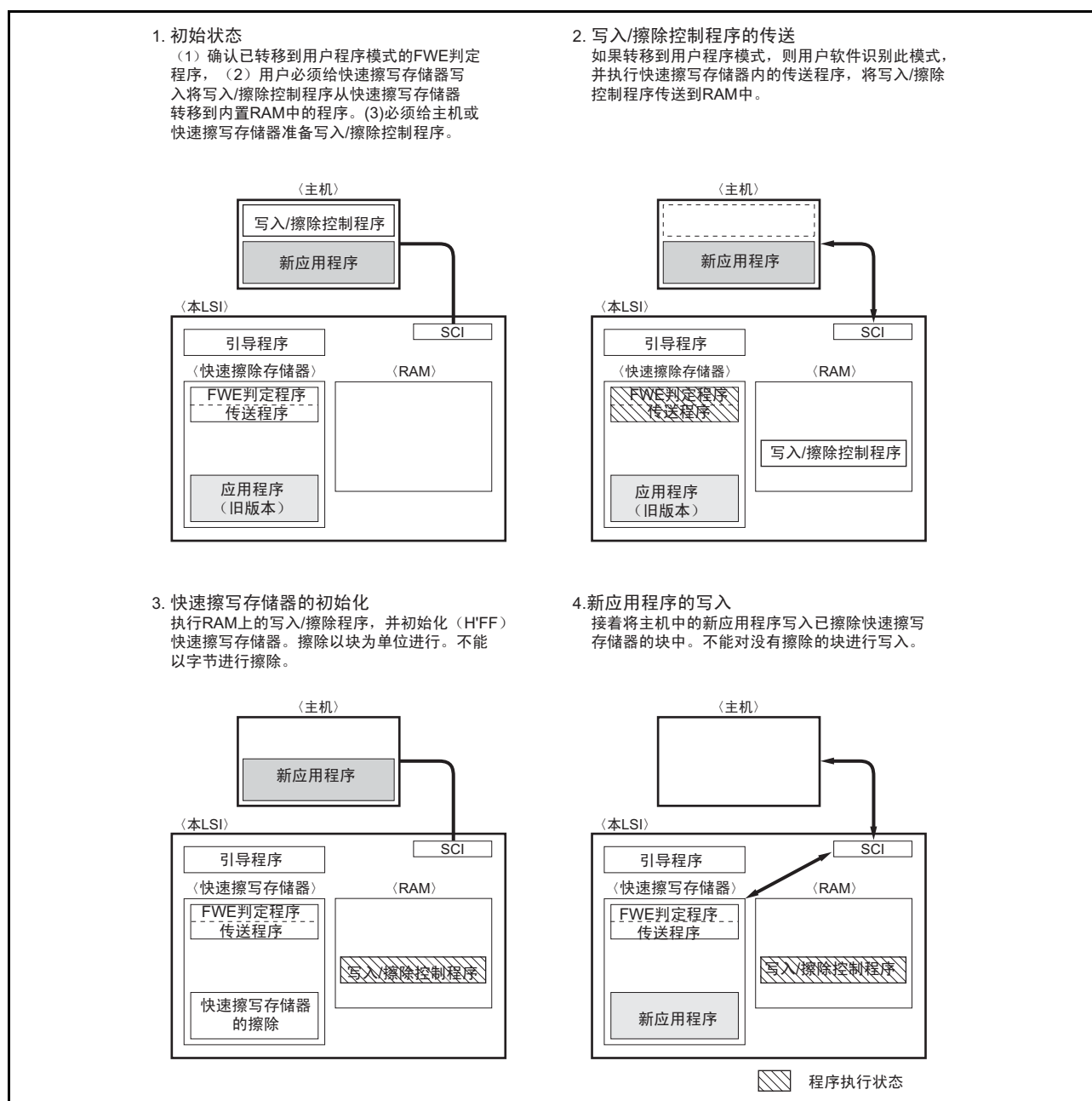


图 17.4 用户程序模式

17.3 块结构

64K 字节快速擦写存储器的块结构如图 17.5 所示。粗线框表示擦除块。细线框表示编程单位，框内的数值表示地址。快速擦写存储器被分成 28K 字节（1 块），16K 字节（1 块），8 K 字节（2 块），1K 字节（4 块），擦除以这些单位进行。编程在 128 字节单位里被执行，起始地址从 H'00 或者 H'80 开始。

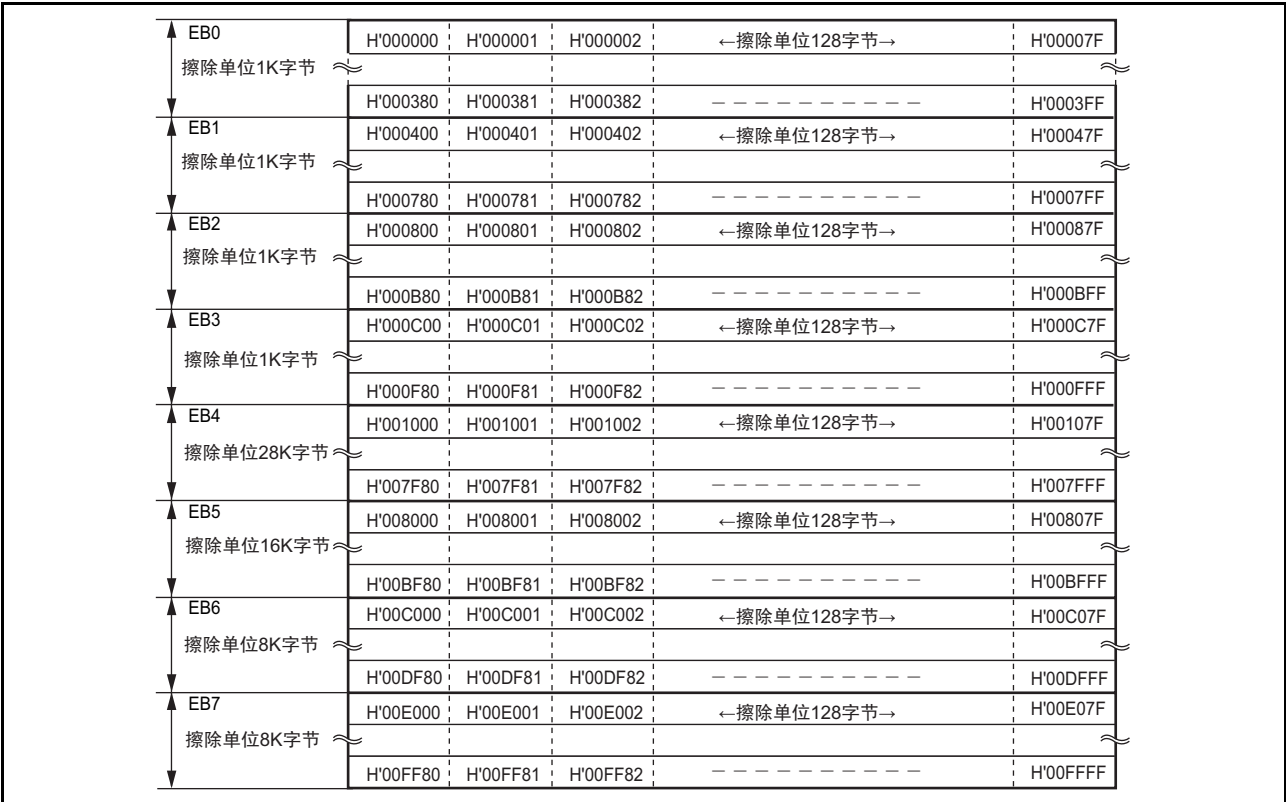


图 17.5 快速擦写存储器的块结构

17.4 输入 / 输出管脚

通过管脚控制快速擦写存储器如表 17.2 所示。

表 17.2 管脚结构

管脚名称	输入 / 输出	功能
RES	输入	复位
FWE	输入	对快速擦写的编程 / 擦除进行硬件保护
MD2	输入	设定本 LSI 的工作模式
MD0	输入	设定本 LSI 的工作模式
TxD1	输出	串行发送数据输出
RxD1	输入	串行接收数据输入

17.5 寄存器的说明

快速擦写存储器有以下的寄存器。

- 快速擦写存储器控制寄存器 1（FLMCR1）
- 快速擦写存储器控制寄存器 2（FLMCR2）
- 擦除块指定寄存器 1（EBR1）
- 快速擦写存储器功率控制寄存器（FLPWCR）

17.5.1 快速擦写存储器控制寄存器 1（FLMCR1）

FLMCR1 使快速擦写存储器转移到编程模式，编程验证模式，擦除模式或擦除验证模式。有关具体的设定方法，可参照「17.7 编程 / 擦除程序」。

位	位名	初始值	R/W	说明
7	FWE	—	R	反映 FWE 管脚的输入电平。FWE 管脚为低电平时，则为 0。为高电平时，则为 1。
6	SWE	0	R/W	软件写允许 位为 1 时，快速擦写存储器可写入 / 擦除。 为 0 时，该寄存器的其它位和 EBR1 的各位不能置位。
5	ESU	0	R/W	擦除准备 置 1 时，变为擦除准备状态，清除时，解除准备状态。
4	PSU	0	R/W	程序准备 置 1 时，变为程序准备状态，清除时，解除准备状态。必须在将 FLMCR1 的 P 位置 1 前置位。
3	EV	0	R/W	擦除验证 置 1 时，转移到擦除验证模式，清除时，解除擦除验证模式。
2	PV	0	R/W	程序验证 置 1 时转移到程序验证模式，清除时，解除程序验证模式。
1	E	0	R/W	擦除 如果在 SWE=1、ESU=1 的状态下将此位置 1，则转移到擦除模式，清除时，解除擦除模式。
0	P	0	R/W	编程 如果在 SWE=1、PSU=1 的状态下将此位置 1，则转移到程序模式，清除时，解除程序模式。

17.5.2 快速擦写存储器控制寄存器 2（FLMCR2）

FLMCR2 表示快速擦写存储器的编程 / 擦除状态。FLMCR2 是只读寄存器，不能写入。

位	位名	初始值	R/W	说明
7	FLER	0	R	在对快速擦写存储器写入 / 擦除过程中，如果检测出错误，则变为错误保护状态，将此位置位。 详细情况请参照「17.8.3 错误保护」。
6~0	—	全为 0	—	保留位 读取时，值固定为 0。

17.5.3 擦除块指定寄存器 1（EBR1）

EBR1 是指定快速擦写存储器擦除块的寄存器。当 FLMCR1 的 SWE 位是 0 时，将 EBR1 初始化为 H'00。
不可将该寄存器 2 个以上的位同时设定为 1。如果设定，则 EBR1 被自动清零。

位	位名	初始值	R/W	说明
7	EB7	0	R/W	此位为 1 时，EB7（H'00E000 ~ H'00FFFF）的 8K 字节变为擦除对象。
6	EB6	0	R/W	此位为 1 时，EB6（H'00C000 ~ H'00DFFF）的 8K 字节变为擦除对象。
5	EB5	0	R/W	此位为 1 时，EB5（H'008000 ~ H'00BFFF）的 16K 字节变为擦除对象。
4	EB4	0	R/W	此位为 1 时，EB4（H'001000 ~ H'007FFF）的 28K 字节变为擦除对象。
3	EB3	0	R/W	此位为 1 时，EB3（H'000C00 ~ H'000FFF）的 1K 字节变为擦除对象。
2	EB2	0	R/W	此位为 1 时，EB2（H'000800 ~ H'000BFF）的 1K 字节变为擦除对象。
1	EB1	0	R/W	此位为 1 时，EB1（H'000400 ~ H'0007FF）的 1K 字节变为擦除对象。
0	EB0	0	R/W	此位为 1 时，EB0（H'000000 ~ H'0003FF）的 1K 字节变为擦除对象。

17.5.4 快速擦写存储器功率控制寄存器（FLPWCR）

当 LSI 转换到子激活模式时，FLRWCR 选择是否将快速擦写存储器变为低功耗模式。

位	位名	初始值	R/W	说明
7	PDWND	0	R/W	此位为 1 时，禁止转移到快速擦写存储器的低功耗模式。
6	—	全为 0	R	保留位 读取值固定为 0。

17.6 单板编程模式

作为进行快速擦写存储器的编程 / 擦除的模式，有能在单板上编程 / 擦除的引导模式和用 PROM 编程器编程 / 擦除的编程器模式。此外，还可以在用户模式下进行单板上编程 / 擦除。如果从复位状态开始启动复位时，本 LSI 就通过 MD 管脚和 FWE 管脚转换到如表 17.3 所示的不同模式。在解除复位以前各管脚输入电平必须被定义为 4 个状态。

当转换到引导模式，就启动 LSI 内部引导程序。引导程序经 SCI_1，将编程控制程序从与外部连接的主机传送到内部 RAM 中，全部擦除快速擦写存储器后，编程控制程序被执行。这个能被用于在单板状态下的初次编程，或者用于用户模式无法进行编程 / 擦除时的强制返回等。在用户编程模式下，可以通过转换到用户准备的编程 / 擦除程序，擦除改写任意块。

表 17.3 编程模式选择方法

MD2	MD0	FWE	复位解除后的 LSI 的状态
1	1	1	用户模式
0	1	1	引导模式

17.6.1 引导模式

引导模式在复位解除转换到编程控制程序的运行如表 17.4 所示。

1. 在引导模式下，主机侧需要预先准备好快速擦写存储器的编程控制程序。编程控制程序必须按照「17.7 编程 / 擦除程序」准备。
2. SCI_1 设定为异步方式，收发格式格式为（8 位数据，1 停止位，无奇偶位）
3. 如果启动引导程序，则测定从主机连续发送的异步串行通信数据 H'00 的低电平宽度，计算位速率，并将 SCI_1 的位速率与主机的位速率匹配。解除复位要在 RxD 管脚为高电平的状态下进行。根据需要，在电路板上将 RxD 管脚及 TxD 管脚上拉。从复位解除到能测定低电平宽度为止，大约需要 100 个状态。
4. 由于在位速率的匹配结束后，作为调整结束信号，发送 1 个字节的 H'00，因此，如果主机正常接收到调整结束信号的话，则必须发送 1 个字节的 H'55。如果不能正常接收，则必须通过复位再次启动引导模式。根据主机的位速率和本 LSI 的系统时钟频率的组合，会产生容许范围内位速率不匹配的情况。因此，主机的传送位速率和本 LSI 的系统时钟频率要设定在表 17.5 的范围内。
5. 在引导模式下，内部 RAM 的一部分被引导程序使用。能存放主机发送的编程控制程序的区域是 H'FFE800 ~ H'FFEFBF。从程序的执行开始转换到编程控制程序为止，不能使用引导程序的区域。
6. 虽然在转换到编程控制程序时，SCI_1 结束接收发送操作，但是因为在 BRR 中匹配的位速率的值被保持，因此能继续使用编程控制模式发送接收与主机间的编程数据和验证数据。TxD 管脚变为高电平输出状态。转换到编程控制程序后的 CPU 的通用寄存器不定。特别是堆栈指针被隐含地使用在子程序调用等，因此，必须在编程控制程序的开头初始化。
7. 通过复位解除引导模式。使复位管脚为低电平，在经过最少 20 个状态后，设定 MD 管脚并解除复位。发生 WDT 的溢出复位时也能解除引导模式。
8. 在引导模式时，不能改变 MD 管脚的输入电平。
9. 对快速擦写存储器进行编程或擦除的过程中，所有的中断不能使用。

表 17.4 引导模式的操作

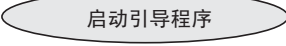
项目	主机的运行	通信内容	本LSI的运行
	处理内容		处理内容
启动引导模式			复位开始后 转移到引导程序 
位速率的匹配	以规定的位速率，连续发送H'00 ↓ 如果正常接收H'00，则发送H'55 ↓ 接收H'AA	H'00、H'00 · · · H'00 ↓ H'00 ↓ H'55 ↓ H'AA	· 测定接收数据H'00 的低电平宽度 · 计算位速率，设定SCI-1的BRR · 位速率匹配结束后，向主机发送H'00 ↓ 如果接收H'55，则向主机发送H'AA
写入控制程序的传送	将传送的写入控制程序的字节数 (N)， 按高位字节，低位字节的顺序发送2个字节 ↓ 发送写入控制程序的每1字节 (重复N次)	↓ 回送 ↓ H'XX ↓ 回送	将接收的2字节数据 回送给主机 ↓ 将接收到的数据回送给主机的 同时传送给RAM (重复N次)
擦除快速擦写存储器	↓ 接收H'AA	↓ H'FF ↓ H'AA	检测快速擦写存储器的数据，在被写入 的情况下，擦除全部块，并向主机发送 H'AA (不能擦除时，发送H'FF，停止运行)
			↓ 转移到被传送到内置RAM中的写入控制 程序，并开始执行

表 17.5 可以自动匹配位速率的系统时钟频率

主机的位速率	本 LSI 的系统时钟频率范围
19200bps	20MHz
9600bps	8 ~ 20MHz
4800bps	4 ~ 20MHz

17.6.2 用户模式的编程 / 擦除

在用户模式下，通过转换到用户准备的编程 / 擦除程序，可以在单板上擦除改写任意块。不仅需要用户设定转换的条件和提供在单板上改写数据的手段。另外，根据需要，有必要给快速擦写存储器的一部分写入编程 / 擦除程序，或者写入为了从外部提供的编程 / 擦除程序的程序。由于在编程 / 擦除过程中不能读出快速擦写存储器，必须和引导模式相同，把编程 / 擦除程序传送到内部 RAM 中，并执行。用户模式下的编程 / 擦除步骤的例子如图 17.6 所示。请根据「17.7 编程 / 擦除程序」准备编程 / 擦除程序。

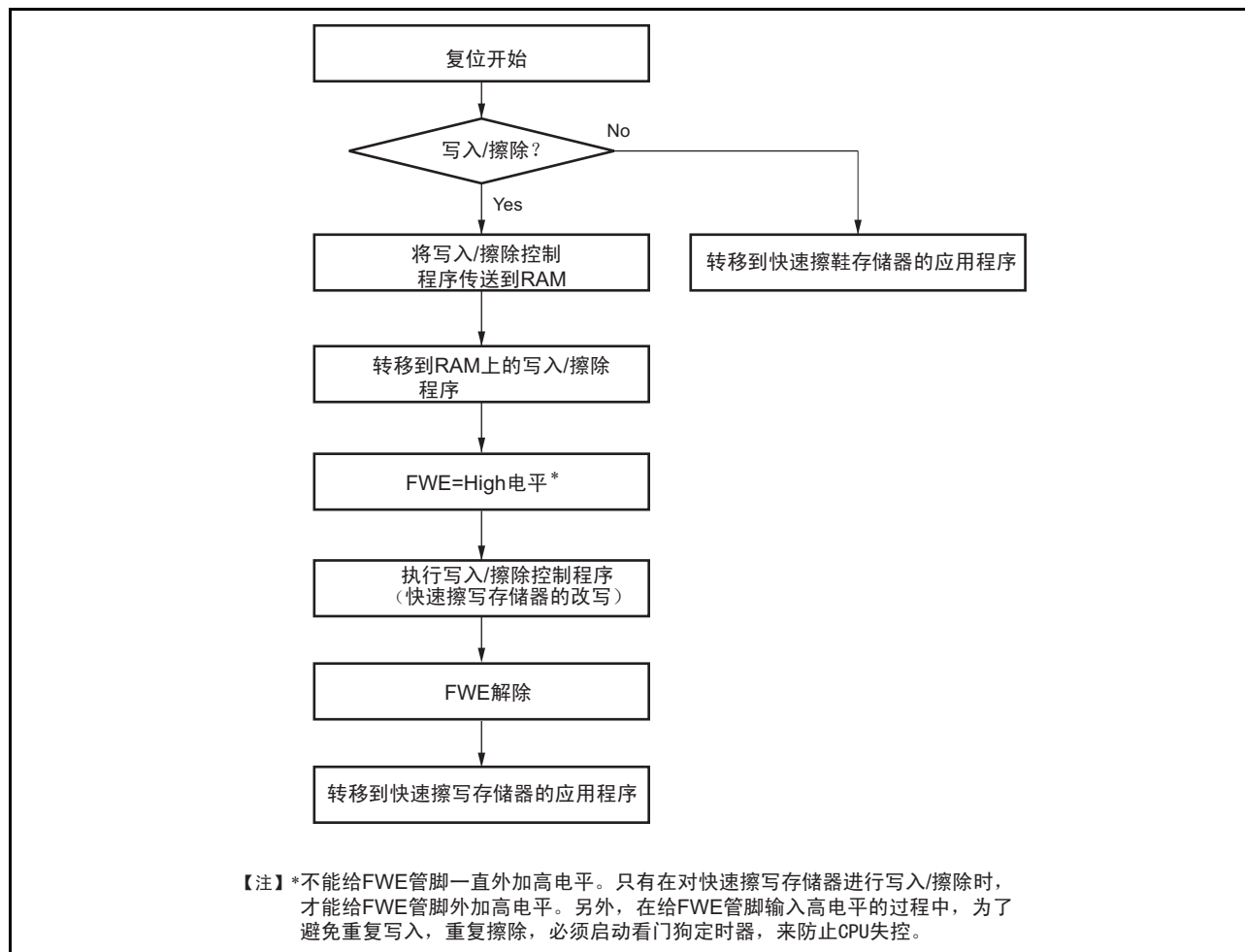


图 17.6 用户模式的编程 / 擦除例子

17.7 编程 / 擦除程序

采用软件方式，用 CPU 对单板上的快速擦写存储器编程 / 擦除。快速擦写存储器根据 FLMCR1 的设定转换到程序模式，程序验证模式，擦除模式和擦除验证模式。在引导模式下的编程控制程序，用户模式下的编程 / 擦除程序，组合这些模式进行编程 / 擦除。快速擦写存储器的编程请按照「17.7.1 编程 / 编程验证」进行，快速擦写存储器的擦除请按照「17.7.2 擦除 / 擦除验证」进行。

17.7.1 编程 / 编程验证

快速擦写存储器的编程请依照图 17.7 所示的编程 / 编程验证流程图。如果按照这个流程图进行编程操作的话，可以不损坏对器件的电压应力或数据的可靠性进行编程。

1. 在擦除状态下进行编程，对已经被编程的地址不可进行再次编程。
2. 一次编程的单位为 128 字节。即使编程不满 128 字节的数据时也必须向快速擦写存储器传送 128 字节的数据。额外地址的数据必须写入 H'FF。
3. 必须在 RAM 上保证 128 字节的写入数据区，128 字节的再次写入数据区和 128 字节的追加写入数据区。再次写入数据的运算，追加写入数据的运算如图 17.7 进行。
4. 必须以字节为单位，从再次写入数据区或者追加写入数据区连续传送 128 字节到快速擦写存储器。程序地址和 128 字节的数据被锁存在快速擦写存储器内。必须把传送地的快速擦写存储器的起始地址的低 8 位设定为 H'00 或 H'80。
5. P 位置 1 的时间为编程时间。请按照图 17.7 设定编程时间。
6. 为了避免因为程序失控等而造成的重复编程，设定看门狗定时器。溢出周期必须设定在 6.6ms 左右。
7. 对验证地址进行虚写，必须给低 2 位为 B'00 的地址写入 1 字节的 H'FF。验证数据以长字读取从验证的地址到虚写的地址进行。
8. 对同一位进行重复编程 / 编程验证，最多次数不要超过 1,000 次。

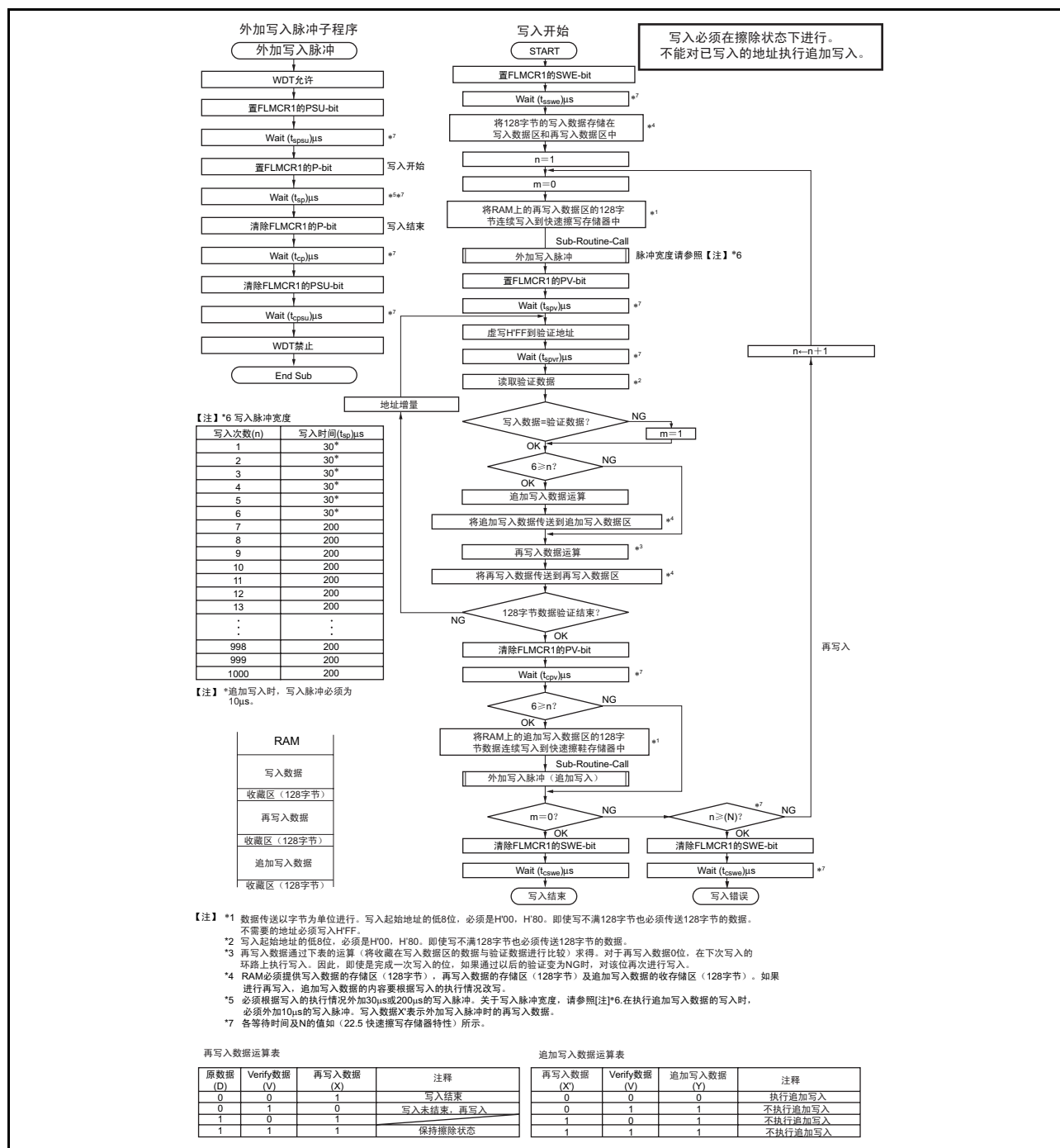


图 17.7 程序 / 程序验证流程图

17.7.2 擦除 / 擦除验证

按照图 17.8 的擦除 / 擦除验证流程图进行擦除。

1. 在擦除前不必预写入（将要擦除的存储器的所有数据全部置为 0）。
2. 擦除以块为单位进行。在擦除块寄存器（EBR1）仅仅按照单一一位的规范执行。擦除多块也必须按块分别依次擦除。
3. E 位置 1 的时间为擦除时间。
4. 为了避免由于程序失控而造成的重复擦除，设定看门狗定时器。溢出周期必须设定在 19.8ms 左右。
5. 为验证地址的虚拟写，必须给低 2 位为 B'00 的地址写入 1 个字节的数据 H'FF。验证数据以长字读取从验证的地址到虚写的地址进行。
6. 如果读出的数据在未成功擦除时，再次将其设定为擦除模式，并且按之前的顺序重复擦除 / 擦除验证。但是，重复次数最多不要超过 100 次。

17.7.3 快速擦写存储器的编程 / 擦除时的中断处理

在对快速擦写存储器编程 / 擦除的过程中或执行引导程序过程中，由于以下原因屏蔽包括 NMI 在内的全部中断请求。

1. 编程 / 擦除过程中，如果发生中断，不能保证按照编程 / 擦除算法正常运行。
2. 如果在写入向量地址前或者在编程 / 擦除过程中开始中断异常处理，一个正确的向量不能被读取，并且 CPU 失控。
3. 在执行引导程序过程中，如果发生中断，就不能按正常顺序执行引导模式。

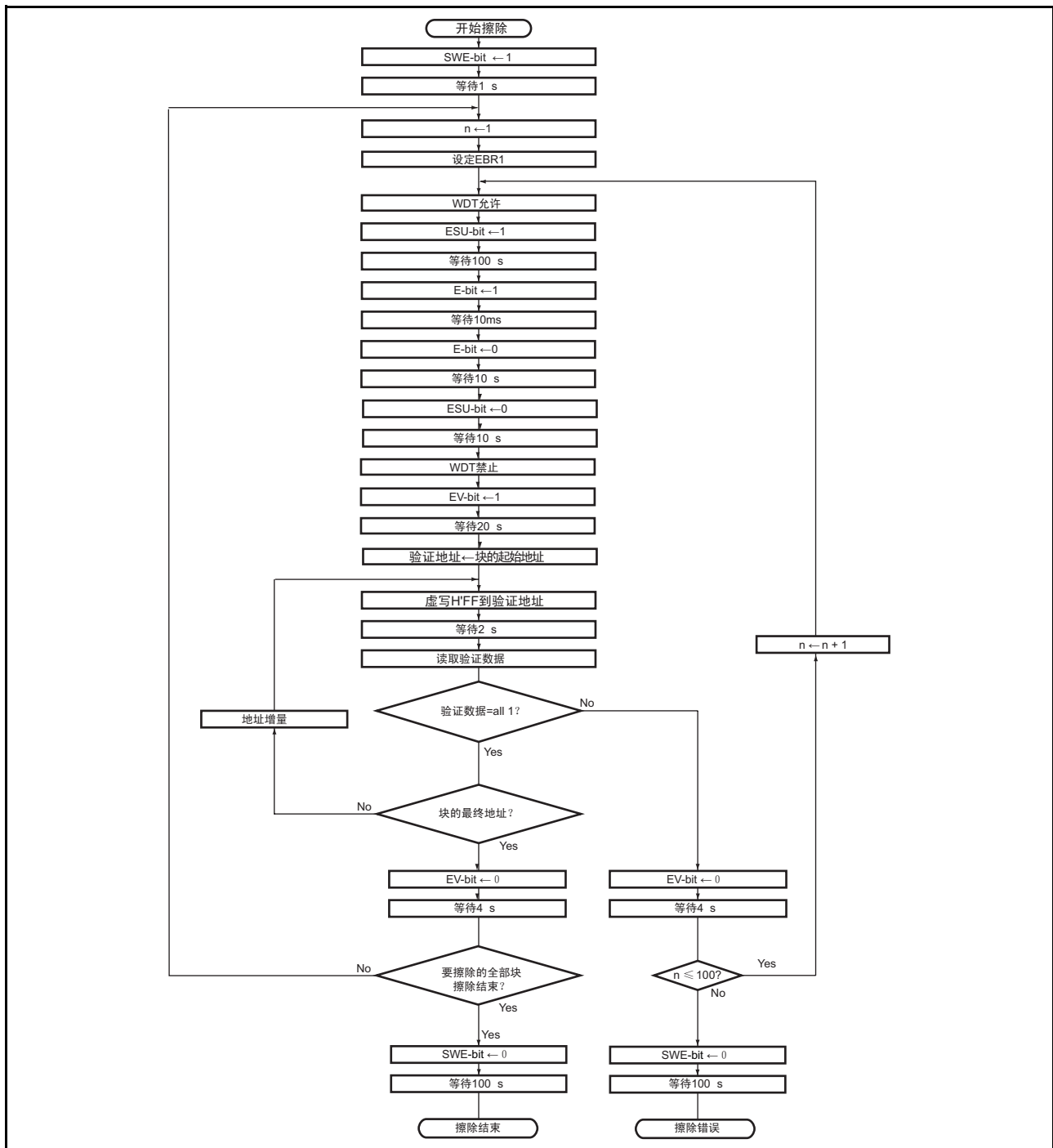


图 17.8 擦除 / 擦除验证流程图

17.8 编程 / 擦除保护

快速擦写存储器的编程 / 擦除保护状态有硬件保护，软件保护和错误保护 3 种。

17.8.1 硬件保护

硬件保护是通过向复位或者待机状态的状态转换，对快速擦写存储器的编程 / 擦除被强制禁止和中断的状态。快速擦写存储器控制寄存器 1（FLMCR1）快速擦写存储器控制寄存器（FLMCR2）和块指定寄存器 1（EBR1）被初始化。由 $\overline{\text{RES}}$ 管脚复位时，从加入电源到震荡稳定为止，如果不保持 $\overline{\text{RES}}$ 管脚为低电平，就不能变为复位状态。另外，运行中的复位，在 AC 特性规定的 $\overline{\text{RES}}$ 脉冲宽度之间，必须保持 $\overline{\text{RES}}$ 管脚为低电平。

17.8.2 软件保护

通过软件清除 FLMCR1 的 SWE 位，使全部块的编程 / 擦除变为保护状态。在此状态下，即使将 FLMCR1 的 P 位或 E 位置位，也不转移到程序模式或擦除模式。另外，通过设定块指定寄存器 1（EBR1）可以按块进行擦除保护。如果设定 EBR1 为 H'00 时，全部块变为擦除保护状态。

17.8.3 错误保护

错误保护是在对快速擦写存储器进行写入 / 擦除过程中，检测出 CPU 失控或不按照编程 / 擦除算法的运行，强制中断编程 / 擦除运行的状态。通过中断编程 / 擦除运行，防止因重复编程 / 重复擦除给快速擦写存储器造成的损坏。

如果在对快速擦写存储器编程 / 擦除过程中检测出以下错误，FLMCR2 的 FLER 位被置为 1，成为错误保护状态。

- 在编程 / 擦除过程中，读出相关地址区域的快速擦写存储器（包括读向量及取指令）
- 在编程 / 擦除过程中，开始除复位以外的异常处理
- 在编程 / 擦除过程中，执行 SLEEP 指令

此时，虽然保持 FLMCR1，FLMCR2 和 EBR1 的内容，但在检测出错误时，强制中断编程模式或擦除模式。即使将 P 位和 E 位重复置位，也不转换到编程模式或擦除模式。但是，PV 位和 EV 位被使能并且能转换到验证模式。只能通过加电复位解除错误保护状态。

17.9 编程器模式

在编程器模式，通过插座适配器，与单个快速擦写存储器相同，能使用 PROM 编程器编程 / 擦除。使用支持 64K 字节快速擦写存储器的内部微处理器类型（FZTAT64V5A）的 PROM 编程器。

17.10 快速擦写存储器的低功耗运行

在用户模式，快速擦写存储器将运行下边的任一种状态。

- 正常运行状态
快速擦写存储器可读 / 写。
- 低功耗状态
部分电源电路停止， LSI 为子时钟操作时快速擦写存储器可读。
- 待机状态
停止快速擦写存储器的所有电路。

本 LSI 的工作模式和快速擦写存储器状态的关系如表 17.6 所示。快速擦写存储器从待机状态恢复到正常工作状态时，需要从已停止的电源电路的稳定运行一段时间。包括使用外部时钟的情况，为了使恢复到正常工作模式时的等待时间保持在 20μs 以上，必须设置 SBYCR 的 STS2 ~ STS0。

表 17.6 快速擦写存储器的运行状态

本 LSI 的工作模式	快速擦写存储器的状态
高速模式 中速模式 睡眠模式	通常工作模式
子激活模式 子睡眠模式	PDWND=0 时 低功耗状态（只读） PDWND=1 时 通常运行状态（只读）
监视模式 软件待机模式 硬件待机模式	待机模式

第 18 章 掩模型 ROM

本群内置有 64k 及 128k 字节的掩模型 ROM。内部 ROM 通过 16 位宽度的数据总线与 CPU 相连接。内部 ROM 数据总是以 1 个状态存取。

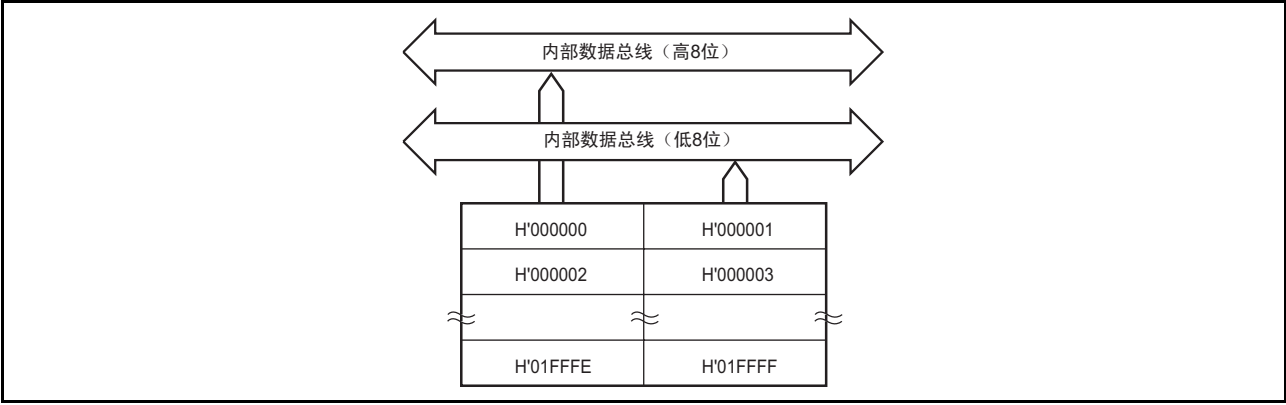


图 18.1 128k 字节掩模型 ROM 的框图 (HD6432282)

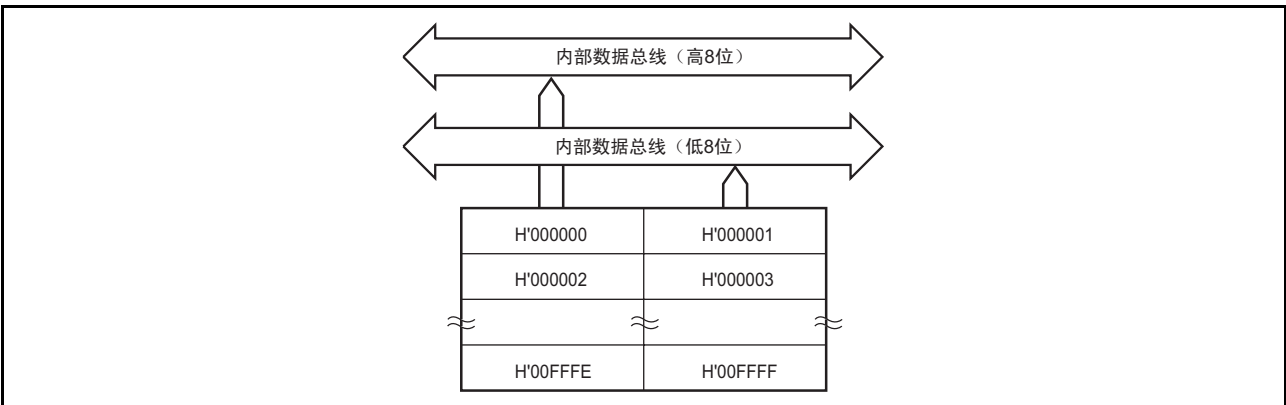


图 18.2 64k 字节掩模型 ROM 的框图 (HD6432281)

18.1 使用时的注意事项

18.1.1 F-ZTAT 单片机的掩模型 ROM 化时的注意事项

掩模版中有存在于 F-ZTAT 版中的用于控制快速擦写存储器的内部寄存器。如表 18.1 所示的寄存器存在于 F-ZTAT 版中，掩模版中没有。如果在掩模版中读取表 18.1 所示的寄存器时，就会读出不定值。因此，将在 F-ZTAT 版中开发的应用软件改变为掩模型 ROM 版时，必须修改应用软件使其不受表 18.1 所示寄存器的影响。

表 18.1 存在于 F-ZTAT 版中但不存在于掩模版中的寄存器

寄存器名称	简称	地址
Flash 存储器控制寄存器 1	FLMCR1	H'FFA8
Flash 存储器控制寄存器 2	FLMCR2	H'FFA9
块擦除指定寄存器 1	EBR1	H'FFAA
块擦除指定寄存器 2	EBR2	H'FFAB
RAM 仿真寄存器	RAMER	H'FFDB
Flash 存储器功率控制寄存器	FLPWCR	H'FFAC

第 19 章 时钟振荡器

本 LSI 有一个内部时钟振荡器，并生成系统时钟 (ϕ)，总线主控器时钟，内部时钟和子时钟。时钟振荡器由振荡器，PLL 电路，子时钟分频器，时钟选择电路，中速时钟分频器和总线主控器时钟选择电路构成。时钟振荡器的框图如图 19.1 所示。

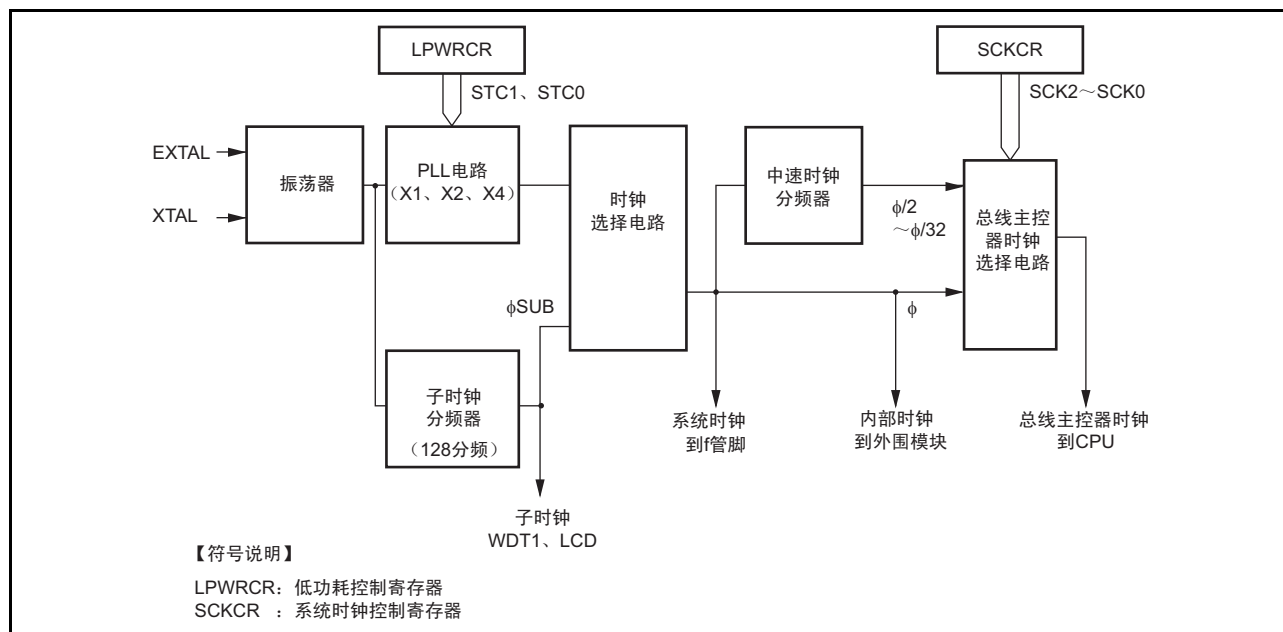


图 19.1 时钟振荡器的框图

通过 PLL 电路，可以改变振荡器发出的频率。频率通过软件设置低功耗控制寄存器 (LPWRCR) 和系统时钟控制寄存器 (SCKCR) 来改变。

19.1 寄存器的说明

时钟振荡器有以下寄存器。

- 系统时钟控制寄存器 (SCKCR)
- 低功耗控制寄存器 (LPWRCR)

19.1.1 系统时钟控制寄存器 (SCKCR)

SCKCR 执行 ϕ 输出，选择改变 PLL 电路倍频率时的运行，控制中速模式。

位	位名	初始值	R/W	说明
7	PSTOP	0	R/W	ϕ 禁止 输出。 ϕ 控制 输出。 高速模式、中速模式 0: ϕ 输出 1: 设置为高电平 睡眠模式 0: ϕ 输出 1: 设置为高电平 软件待机模式 0: 设置为高电平 1: 设置为高电平 硬件待机模式 0: 高阻抗 1: 高阻抗
6 ~ 4	—	全为 0	—	保留位 该位读取时值为 1。
3	STCS	0	R/W	选择倍频切换模式 选择改变 PLL 电路倍频时的运行。 0: 已改变的倍频，在转移到软件待机模式后有效 1: 已改变的倍频，在 STC1、STC0 位改写后有效
2 1 0	SCK2 SCK1 SCK0	0 0 0	R/W R/W R/W	系统时钟选择 2 ~ 0 选择总线主控制器时钟。 000: 高速模式 001: 中速时钟 $\phi/2$ 010: 中速时钟 $\phi/4$ 011: 中速时钟 $\phi/8$ 100: 中速时钟 $\phi/16$ 101: 中速时钟 $\phi/32$ 11x: 禁止设置

【符号说明】x: Don't care

19.1.2 低功耗控制寄存器 (LPWRCR)

LPWRCR 执行低功耗模式控制，子时钟生成控制，振荡电路反馈电阻控制，倍频率设定。

位	位名称	初始值	R/W	说明
7	DTON	0	R/W	请参照「20.1.2 低功耗控制寄存器 (LPWRCR)」。
6	LSON	0	R/W	
5	—	0	R/W	保留位 可读取 / 写入，只能写为 0。
4	SUBSTP	0	R/W	子时钟生成控制 0: 使能生成子时钟 1: 禁止生成子时钟
3	RFCUT	0	R/W	控制振荡电路反馈电阻 0: 主时钟振荡时，反馈电阻 ON，主时钟振荡停止时，反馈电阻 OFF 1: OFF 反馈电阻，改变在软件待机转移返回后有效 【注】使用晶体谐振器时，如果此位置为 1，则振荡器不运行。
2	—	0	R/W	保留位 可读取 / 写入，只能写为 0。
1 0	STC1 STC0	0 0	R/W R/W	设置倍频率 设置 PLL 电路的倍频。 00: 1 01: 2 10: 4 11: 禁止设置

19.2 振荡器

提供时钟的方法有，连接晶体谐振器的方法和输入外部时钟的方法。输入时钟必须设置为 4MHz ~ 20MHz。

19.2.1 连接晶体谐振器的方法

连接晶体谐振器的连接例如图 19.2 所示。请使用表 19.1 列举的阻尼电阻 R_d 。另外，晶体谐振器，必须使用 AT-cut 并联谐振形。

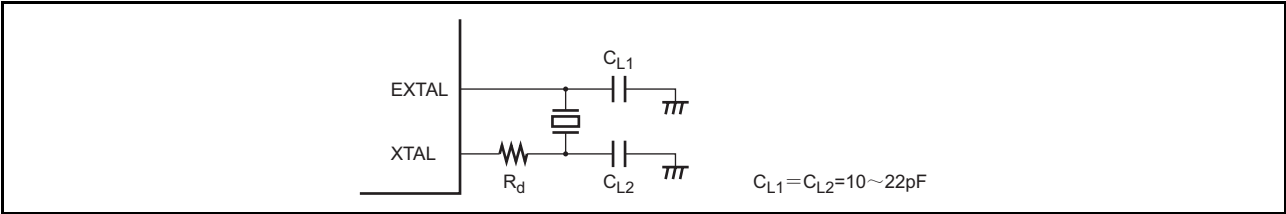


图 19.2 晶体谐振器的连接例

表 19.1 阻尼电阻值

频率 (MHz)	4	8	12	16	20
R_d (Ω)	500	200	0	0	0

晶体谐振器的等效电路如图 19.3 所示。必须使用如表 19.2 所示特性的谐振器。

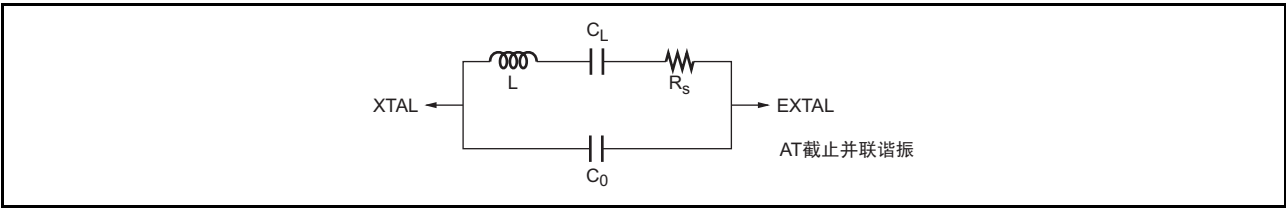


图 19.3 晶体谐振器的等效电路

表 19.2 晶体谐振器的特性

频率 (MHz)	4	8	12	16	20
R_s max (Ω)	120	80	60	50	40
C_0 max (pF)	7				

19.2.2 输入外部时钟的方法

外部时钟输入的连接例如图 19.4 所示。在使 XTAL 管脚为开路状态时，确保寄生电容要在 10pF 以下。在待机模式下，如果给 XTAL 管脚输入反相时钟，必须将外部时钟设置为高电平。

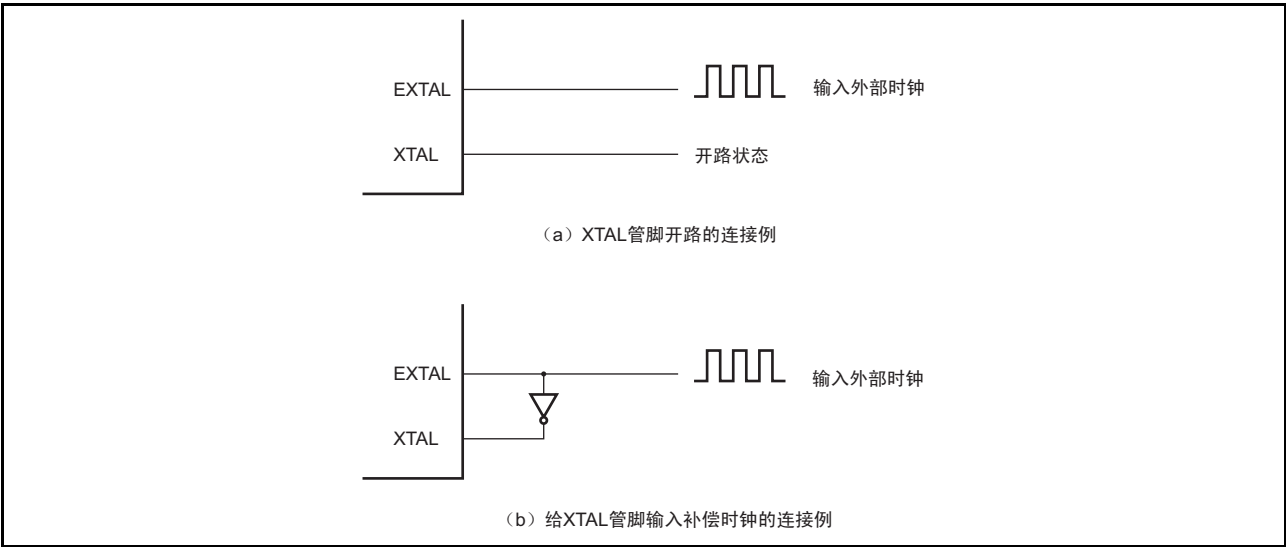


图 19.4 外部时钟的连接例

外部时钟的输入条件如表 19.3 所示。

表 19.3 外部时钟的输入条件

项目	符号	V _{CC} = 5.0V±10%		单位	测定条件	
		min	max			
外部时钟输入脉冲 低电平宽度	t _{EXL}	15	—	ns	图 19.5	
外部时钟输入脉冲 高电平宽度	t _{EXH}	15	—	ns		
外部时钟上升时间	t _{EXr}	—	5	ns		
外部时钟下降时间	t _{EXf}	—	5	ns		
时钟脉冲宽度 低电平	t _{CL}	0.4	0.6	t _{cyc}	φ ≥ 5MHz	图 22.2
		80	—	ns	φ < 5MHz	
时钟脉冲宽度 高电平	t _{CH}	0.4	0.6	t _{cyc}	φ ≥ 5MHz	
		80	—	ns	φ < 5MHz	

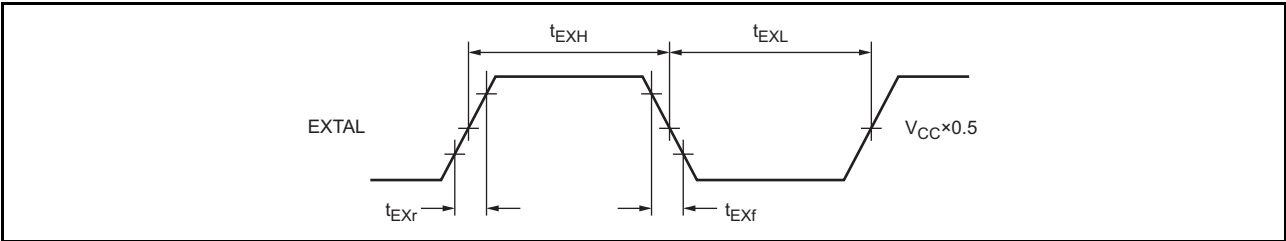


图 19.5 外部时钟输入时序

19.3 PLL 电路

PLL 电路拥有使振荡器的频率实现 1 倍，2 倍，4 倍频的功能。倍频由 LPWRCR 的 STC1 和 STC0 位设置。这时，内部时钟的上升沿的相位要与 EXTAL 管脚的上升沿的相位匹配。

当 PLL 电路改变时，相应 SCKCR 的 STCS 位的设置改变，运行也有所不同。

STCS 位为 0 时，已改变的倍频在软件待机模式转移后变得有效。转移时间由待机控制寄存器（SBYCR）的 STS2 ~ STS0 位来设置。SBYCR 请参照「20.1.1 待机控制寄存器（SBYCR）」。

1. 初始状态下，PLL 电路的倍频为 1 倍。
2. 通过 STS2 ~ STS0 位，设置转移时间。
3. 通过 STC1、STC0 位，设置倍频，并转移为软件待机模式。
4. 停止时钟振荡器，STC1、STC0 的设置变得有效。
5. 解除软件待机模式，随着 STS2 ~ STS0 位的设置，确保转移时间是可靠的。
6. 本 LSI，在经过设置的转移时间以后，以已改变的倍频，重新运作。

19.4 子时钟分频器

子时钟分频器将振荡器生成的时钟 128 分频，生成子时钟。将子时钟作为计时时钟使用时，需要通过软件进行校正。

19.5 中速时钟分频器

中速时钟分频器，将系统时钟分频，并生成 $\phi/2$ 、 $\phi/4$ 、 $\phi/8$ 、 $\phi/16$ 、和 $\phi/32$ 。

19.6 总线主控器时钟选择电路

总线主控器时钟选择电路，通过设置 SCKCR 的 SCK2 ~ SCK0 位，选择提供给总线主控器的时钟是高速模式还是中速时钟（ $\phi/2$ 、 $\phi/4$ 、 $\phi/8$ 、 $\phi/16$ 、和 $\phi/32$ ）。

19.7 使用时的注意事项

19.7.1 谐振器的注意事项

因为有关谐振器的诸多特性密切关系到用户的电路板设计，所以希望用户参照本章介绍的谐振器的连接例子，充分研究后再使用。谐振器的电路额定值取决于谐振器，安装电路的寄生电容等因素，因此必须和谐振器制造厂家磋商后再做决定。谐振器的外加电压不能超过最大额定值。

19.7.2 电路板设计上的注意事项

使用晶体谐振器时，必须尽量把谐振器及负载电容设置在 XTAL、EXTAL 管脚附近。如图 19.6 所示谐振电路附近不要让信号线通过。否则，可能发生因感应而不能正确振荡的情况。

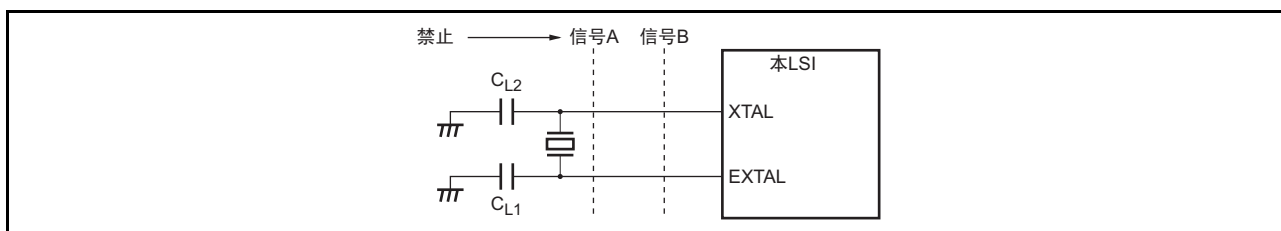


图 19.6 有关振荡电路的电路板设计的注意事项

PLL 电路的外置推荐电路如图 19.7 所示。为了稳定振荡，要把电容 C1 及电阻 R1 设置在 PLLCAP 管脚附近。另外，不要和其他信号交叉。必须分离 PLLVCL、PLLVSS 和 VCC、VSS 与电路板的电源提供源，在管脚附近插入旁路电容 CB。

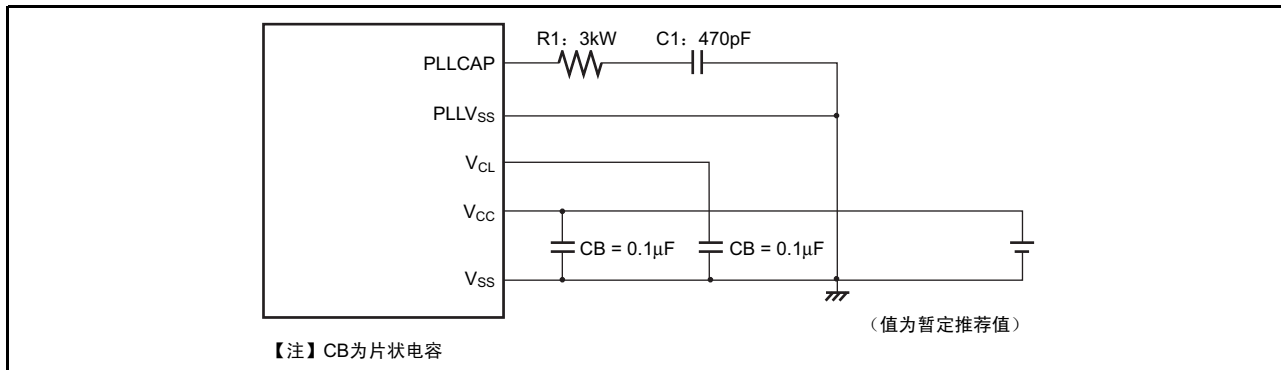


图 19.7 PLL 电路的外置推荐电路

第 20 章 低功耗状态

本 LSI、除具备通常的程序执行状态以外，还有 CPU 及振荡器的停止运行状态、显著降低功耗的低功耗状态。通过对 CPU、内部外围设备模块的分别控制，可以实现电能的低功耗化。

本 LSI 的运行模式，除高速模式外还有：

- 中速模式
- 子激活模式
- 睡眠模式
- 子睡眠模式
- 监视模式
- 模块停止模式
- 硬件待机模式
- 软件待机模式

等低功耗状态。睡眠模式、子睡眠模式为 CPU 的状态、中速模式为 CPU 及总线控制器的状态、子激活模式为 CPU 及总线控制器、内部外围设备功能的状态、模块停止状态为内部外围设备功能（包含 CPU 外的总线控制器）的状态。这些状态能够进行组合设定。

本 LSI 复位后的运行模式为高速模式 / 模块停止模式。

模式间状态转移图如图 20.1 所示。表 20.1 中描述执行 SLEEP 指令时各模式间的转移条件、各模式下的 LSI 内部状态如表 20.2 所示。

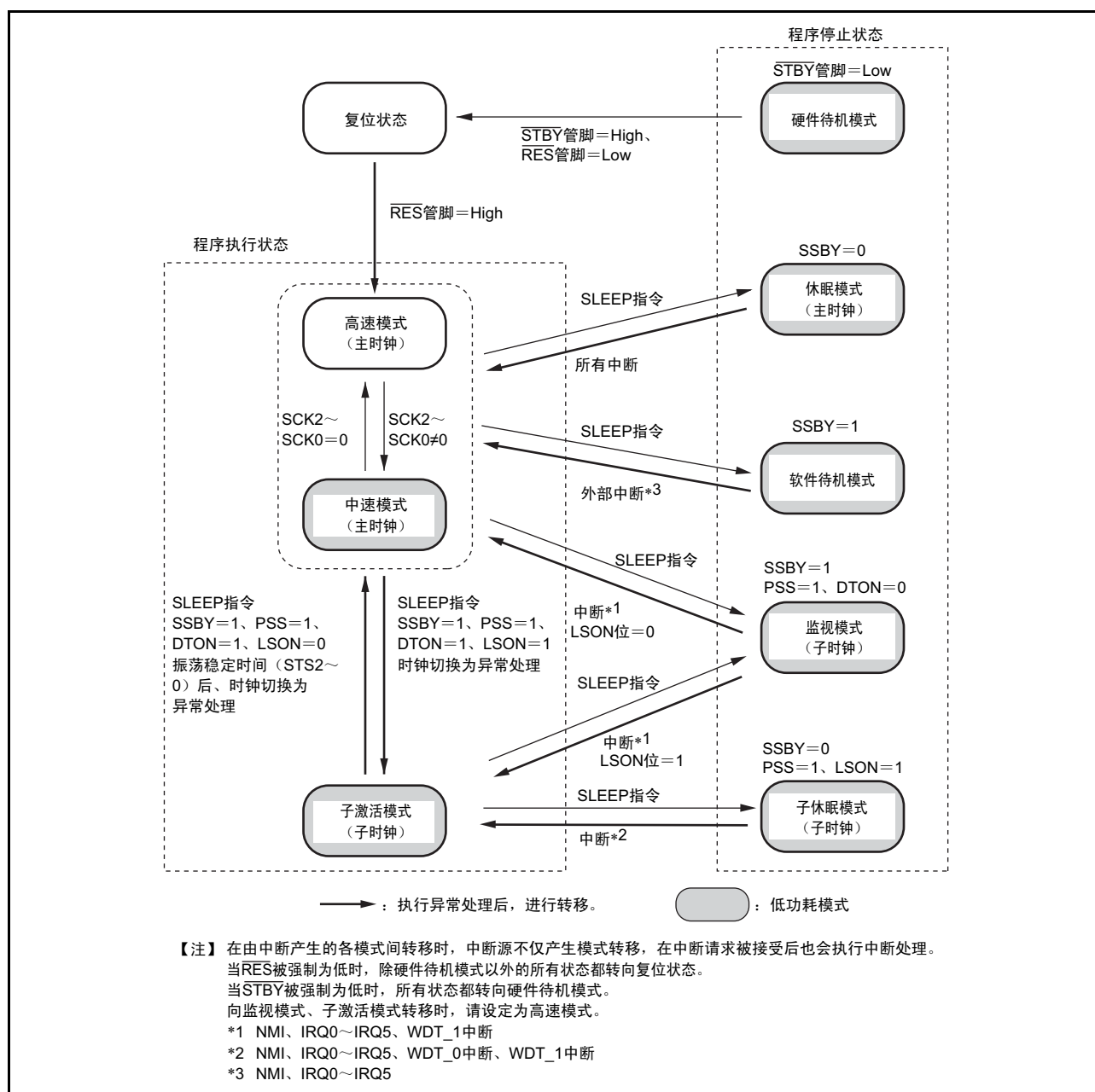


图 20.1 模式间转换图

表 20.1 低功耗模式转移条件

转移前的状态	转移时的控制位状态				调用 SLEEP 指令 转移后的状态	调用中断低功耗 返回转移后的状态
	SSBY	PSS	LS0N	DT0N		
高速 / 中速	0	*	0	*	睡眠	高速 / 中速
	0	*	1	*	—	—
	1	0	0	*	软件待机	高速 / 中速
	1	0	1	*	—	—
	1	1	0	0	监视	高速
	1	1	1	0	监视	子激活
	1	1	0	1	—	—
	1	1	1	1	子激活	—
子激活	0	0	*	*	—	—
	0	1	0	*	—	—
	0	1	1	*	子睡眠	子激活
	1	0	*	*	—	—
	1	1	0	0	监视	高速
	1	1	1	0	监视	子激活
	1	1	0	1	高速	—
	1	1	1	1	—	—

【符号说明】 *: Don't care

—: 禁止设定的状态

表 20.2 各运行模式下 LSI 的内部状态

功能		高速	中速	睡眠	模块停止	监视	子激活	子睡眠	软件待机	硬件待机
系统时钟发生器		运行	运行	运行	运行	运行	运行	运行	停止	停止
CPU	指令寄存器	运行	中速运行	停止 (保持)	高 / 中速 运行	停止 (保持)	子时钟 运行	停止 (保持)	停止 (保持)	停止 (不定)
外部中断	NMI	运行	运行	运行	运行	运行	运行	运行	运行	停止
	IRQ0 ~ IRQ5									
外围设备功能	WDT_1	运行	运行	运行	—	子时钟 运行	子时钟 运行	子时钟 运行	停止 (保持)	停止 (复位)
	WDT_0	运行	运行	运行	—	停止 (保持)	子时钟 运行	子时钟 运行	停止 (保持)	停止 (复位)
	TPU_0	运行	运行	运行	停止 (保持)	停止 (保持)	停止 (保持)	停止 (保持)	停止 (保持)	停止 (复位)
	TPU_1									
	TPU_2									
	SCI_0	运行	运行	运行	停止 (复位)	停止 (复位)	停止 (复位)	停止 (复位)	停止 (复位)	停止 (复位)
	SCI_1									
	PWM									
	HCAN*									
	A/D									
	LCD	运行	运行	运行	停止 (保持)	运行	运行	运行	停止 (保持)	停止 (复位)
	RAM	运行	运行	停止 (保持)	运行	保持	运行	保持	保持	保持
	I/O	运行	运行	运行	运行	保持	运行	保持	保持	高阻抗

- 【注】 1. 停止（保持）时，内部寄存器的值保持。内部状态运行中断。
 2. 停止（复位）时，初始化内部寄存器的值及内部状态。
 3. 在模块停止模式下，执行停止设定后仅模块停止（复位保持）。
 4. 在监视、子激活、子睡眠模式下运行 LCD 时，请在时钟模式中选择使用子时钟模式。
 * H8S/2280 群中不具备此项功能。

20.1 寄存器说明

和低功耗模式相关的寄存器如下所示。关于系统时钟控制寄存器请参考（SCKCR）「19.1.1 系统时钟控制寄存器（SCKCR）」。

- 系统时钟控制寄存器（SCKCR）
- 待机控制寄存器（SBYCR）
- 低功耗控制寄存器（LPWRCR）
- 模块待机控制寄存器 A（MSTPCRA）
- 模块待机控制寄存器 B（MSTPCRB）
- 模块待机控制寄存器 C（MSTPCRC）
- 模块待机控制寄存器 D（MSTPCRD）

20.1.1 待机控制寄存器（SBYCR）

SBYCR 实现软件待机模式控制。

位	位名	初始值	R/W	说 明
7	SSBY	0	R/W	软件待机 选择执行 SLEEP 指令后的转移模式。 0: 转移到睡眠模式。 1: 转移到软件待机模式。 使用外部中断并转换为普通模式来解除软件待机模式时，该位的值不会改变。清除时该位设定为 0。
6 5 4	STS2 STS1 STS0	0 0 0	R/W R/W R/W	待机定时器选择 2 ~ 0 由于外部中断，软件待机模式解除时，该位选择 MCU 在时钟稳定时的待机时间。晶体振荡情况请参考表 20.3，根据操作频率请选择待机时间为 8ms（振荡稳定时间）以上。外部时钟的情况，待机时间请设定为 2ms 以上。 000: 待机时间 = 8192 状态 001: 待机时间 = 16384 状态 010: 待机时间 = 32768 状态 011: 待机时间 = 65536 状态 100: 待机时间 = 131072 状态 101: 待机时间 = 262144 状态 110: 保留 111: 待机时间 = 16 状态
3	—	1	R/W	保留位 总是写为 1。
2 ~ 0	—	全为 0	—	保留位 总是读出 0。写定义为无效。

20.1.2 低功耗控制寄存器 (LPWRCR)

LPWRCR 是一个 8 位可读，可写寄存器。执行低功耗模式控制、子时钟生成控制、振荡电路反馈电阻控制、倍频率设定。

位	位名	初始值	R/W	说明
7	DTON	0	R/W	直接转移 ON 标志 0: 在高速模式或中速模式执行完 SLEEP 指令后，向睡眠模式、软件待机模式或监视模式转移。在子睡眠模式执行完 SLEEP 指令后，向子睡眠或监视状态转移。 在子激活模式下执行完 SLEEP 指令，向子睡眠或监视模式转移。 1: 在高速模式或中速模式执行完 SLEEP 指令后，向子激活模式直接转移或向睡眠模式、软件待机模式转移。在子激活模式中执行完 SLEEP 指令后，向高速模式直接转移或向子睡眠模式转移。
6	LSON	0	R/W	低速 ON 标记 0: 在高速模式或中速模式执行完 SLEEP 指令后，向睡眠模式、软件待机模式、或监视模式转移。在子激活模式执行完 SLEEP 指令后，向监视模式转移或向高速模式直接转移。 监视模式解除后向高速模式转移。 1: 在高速模式执行完 SLEEP 指令后，向监视模式或子激活模式 * 转移。在子激活模式执行完 SLEEP 指令后，向子激活模式或监视模式转移。监视模式解除后，向子激活模式转移。
5	—	0	R/W	保留位 可读取 / 写入，请勿写入 1。
4	SUBSTP	0	R/W	子时钟生成控制 0: 允许子时钟生成 1: 禁止子时钟生成
3	RFCUT	0	R/W	振荡电路反馈电阻控制 0: 主时钟振荡时，反馈电阻设置为 ON、主时钟振荡停止时，反馈电阻设置为 OFF 1: 设置反馈电阻 OFF 【注】使用晶体振荡器时，设置为 1，振荡器停止操作。
2	—	0	R/W	保留位 可读取 / 写入，请勿写入 1。
1 0	STC1 STC0	0 0	R/W R/W	倍频率设定 PLL 电路的倍频率设定。 00: $\times 1$ 01: $\times 2$ 10: $\times 4$ 11: 禁止设定

【注】* 向监视模式、子激活模式转移时，请设定为高速模式。

20.1.3 模块待机控制寄存器 A ~ D (MSTPCRA ~ MSTPCRD)

MSTPCR 执行模块待机模式的控制。对于设定为 1 的模块，转化为模块待机模式。设定为 0 的模块，解除模块待机模式。

• MSTPCRA

位	位名	初始值	R/W	对象模块
7	MSTPA7*1	0	R/W	
6	MSTPA6*1	0	R/W	
5	MSTPA5	1	R/W	16 位定时脉冲单元 (TPU)
4	MSTPA4*1	1	R/W	
3	MSTPA3*1	1	R/W	
2	MSTPA2*1	1	R/W	
1	MSTPA1	1	R/W	A/D 转换器
0	MSTPA0*1	1	R/W	

• MSTPCRB

位	位名	初始值	R/W	对象模块
7	MSTPB7	1	R/W	串行通信接口 _0 (SCI_0)
6	MSTPB6	1	R/W	串行通信接口 _1 (SCI_1)
5	MSTPB5*1	1	R/W	
4	MSTPB4*1	1	R/W	
3	MSTPB3*1	1	R/W	
2	MSTPB2*1	1	R/W	
1	MSTPB1*1	1	R/W	
0	MSTPB0*1	1	R/W	

• MSTPCRC

位	位名	初始值	R/W	模块
7	MSTPC7*1	1	R/W	
6	MSTPC6*1	1	R/W	
5	MSTPC5*1	1	R/W	
4	MSTPC4*1	1	R/W	
3	MSTPC3	1	R/W	CAN 控制器区域网 (HCAN) *2
2	MSTPC2*1	1	R/W	
1	MSTPC1*1	1	R/W	
0	MSTPC0*1	1	R/W	

• MSTPCRD

位	位名	初始值	R/W	模块
7	MSTPD7	1	R/W	LCD 控制器 / 驱动器 (LCD)
6	MSTPD6	1	R/W	电机控制的 PWM 定时器 (PWM)
5	—	不定	—	
4	—	不定	—	
3	—	不定	—	
2	—	不定	—	
1	—	不定	—	
0	—	不定	—	

【注】 *1 MSTPA7、MSTPA6 可以读出 / 写入、初始值为 0。总写为 0。
MSTPA4 ~ MSTPA2、MSTPA0、MSTPB5 ~ MSTPB0、MSTPC7 ~ MSTPC4、MSTPC2 ~ MSTPC0 可以
读取 / 写入，初始值为 1。总写为 1。
*2 H8S/2280 群不具备此功能。

20.2 中速模式

若 SCKCR 的 SCK2 ~ SCK0 位为 1，操作模式在其总线周期结束时转为中速模式。中速模式下，CPU 通过 SCK2 ~ SCK0 位中指定的运行时钟 ($\phi / 2$ 、 $\phi / 4$ 、 $\phi / 8$ 、 $\phi / 16$ 、 $\phi / 32$) 运行。

总线控制器以外的内部外围设备模块通常通过高速时钟 (ϕ) 运行。

中速模式针对数据总线的运行时钟以指定的数据状态进行总线存取。例如，作为运行时钟选择 $\phi/4$ 时，内部存储器为 4 状态时存取、内部 I/O 寄存器为 8 状态时存取。

中速模式通过 SCK2 ~ SCK0 位全部清零，总线周期完成时向高速模式转换，同时中速模式解除。

SBYCR 的 SSBY 位在清零后的状态中执行 SLEEP 指令，则转向睡眠模式。睡眠模式因中断而解除时，中速模式恢复。

在 SSBY 位设定为 1 的状态中执行 SLEEP 指令，则转向软件待机模式。当软件待机模式由于外部中断而解除时，中速模式恢复。

RES 管脚为低电平时，转向复位状态，中速模式被解除。同样由于看门狗定时器溢出，也会产生复位。

STBY 管脚为低电平时，转向硬件待机模式。

向中速模式转移、解除时序如图 20.2 所示。

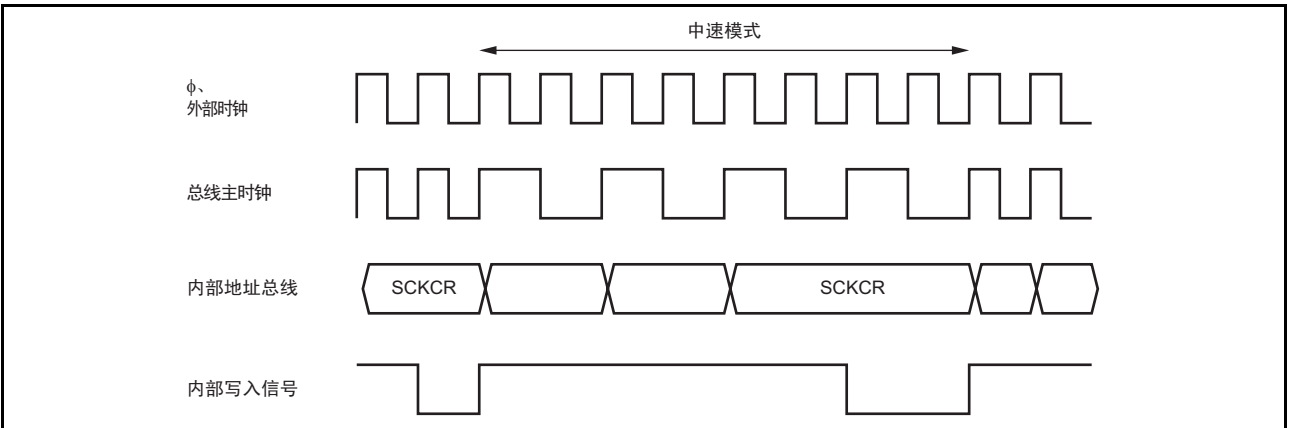


图 20.2 向中速模式转移・解除时序

20.3 睡眠模式

20.3.1 向睡眠模式转移

在 SBYCR 的 SSBY 位 =0 时，执行 SLEEP 指令，CPU 为睡眠模式。睡眠模式时，虽然 CPU 运行停止，但是内部寄存器内容保持不变。CPU 以外的周边外围设备不停止。

20.3.2 睡眠模式的解除

睡眠模式通过所有的中断、 $\overline{\text{RES}}$ 或 $\overline{\text{STBY}}$ 管脚解除。

- 通过中断解除
发生中断请求，睡眠模式被解除，并开始中断异常处理。中断被禁止或 NMI 之外的中断被 CPU 屏蔽时，不可解除睡眠模式。
- 通过 $\overline{\text{RES}}$ 管脚解除
若 $\overline{\text{RES}}$ 管脚为低电平，则转向复位状态。经过规定的复位输入时间后，若 $\overline{\text{RES}}$ 管脚为高电平，CPU 开始复位异常处理。
- 通过 $\overline{\text{STBY}}$ 管脚解除
若 $\overline{\text{STBY}}$ 管脚为低电平，则转向硬件待机模式。

20.4 软件待机模式

20.4.1 向软件待机模式转移

在 SBYCR 的 SSBY 位为 1 时，执行 SLEEP 指令，则转向软件待机模式。在此模式中，CPU、内部外围设备模块以及振荡器的所有功能都停止。CPU 内部寄存器内容和内部 RAM 数据以及除了 SCI、PWM、HCAN*、A/D 转换器、I/O 端口的状态以外的内部外围设备的状态均保持不变。由于振荡器停止，此模式耗电量明显降低。

【注】 * H8S/2280 群不具备此功能。

20.4.2 软件待机模式的解除

软件待机模式通过外部中断（NMI 管脚、 $\overline{\text{IRQ0}} \sim \overline{\text{IRQ5}}$ 管脚）、 $\overline{\text{RES}}$ 管脚或 $\overline{\text{STBY}}$ 管脚解除。

- 通过中断解除
NMI、 $\overline{\text{IRQ0}} \sim \overline{\text{IRQ5}}$ 发生中断请求，时钟振荡开始，经过由 SBYCR 的 STS2 ~ STS0 位设定的时间后，稳定后的时钟被供给整个 LSI、软件待机模式被解除并开始中断异常处理。
由 $\overline{\text{IRQ0}} \sim \overline{\text{IRQ5}}$ 中断解除软件待机模式时，对应的允许位设定为 1，且不可发生比 $\overline{\text{IRQ0}} \sim \overline{\text{IRQ5}}$ 中断优先级高的中断。CPU 屏蔽中断时，不可解除软件待机模式。
- 通过 $\overline{\text{RES}}$ 管脚解除
 $\overline{\text{RES}}$ 管脚为低电平，时钟开始振荡。时钟振荡开始的同时，时钟被供给整个 LSI。到时钟的振荡稳定为止， $\overline{\text{RES}}$ 管脚必须保持低电平。 $\overline{\text{RES}}$ 管脚为高电平时，CPU 就开始复位异常处理。
- 通过 $\overline{\text{STBY}}$ 管脚解除
 $\overline{\text{STBY}}$ 管脚为低电平，则转向硬件待机模式。

20.4.3 软件待机模式解除后振荡器稳定时间的设定

SBYCR 的 STS2 ~ STS0 位的设定，如下所示。

- 在晶体振荡器的情况下
待机时间至少为 8ms（振荡稳定时间）如上所示，请设置 STS2 ~ STS0 位。
对应不同的运行频率和 STS2 ~ STS0 位的设定的待机时间，如表 20.3 所示。
- 在外部时钟的情况下
PLL 电路需要时间稳定。请设定 2ms 以上的待机时间。

表 20.3 振荡稳定时间的设定

STS2	STS1	STS0	待机时间	20MHz	16MHz	12MHz	10MHz	8MHz	6MHz	4MHz	单位
0	0	0	8192 状态	0.41	0.51	0.68	0.8	1.0	1.3	2.0	ms
		1	16384 状态	0.82	1.0	1.3	1.6	2.0	2.7	4.1	
	1	0	32768 状态	1.6	2.0	2.7	3.3	4.1	5.5	8.2	
		1	65536 状态	3.3	4.1	5.5	6.6	8.2	0.9	16.4	
1	0	0	131072 状态	6.6	8.2	10.9	13.1	16.4	21.8	32.8	
		1	262144 状态	3.1	16.4	21.8	26.2	32.8	43.6	65.6	
	1	0	保留	—	—	—	—	—	—	—	
		1	16 状态 *	0.8	1.0	1.3	1.6	2.0	1.7	4.0	

:  : 推荐设定时间

【注】* 请勿设定

20.4.4 软件待机模式应用举例

通过 NMI 管脚的下降沿向软件待机模式转移、通过 NMI 管脚的上升沿解除软件待机模式的例子如图 20.3 所示。

例子中，SYSCR 的 NMIEG 位清 0（指定下降沿）状态下，接受 NMI 中断后、NMIEG 位置 1（指定上升沿）、SSBY 位置 1 后、执行 SLEEP 指令并向软件待机模式转移。

此后，通过 NMI 管脚的上升沿解除软件待机模式。

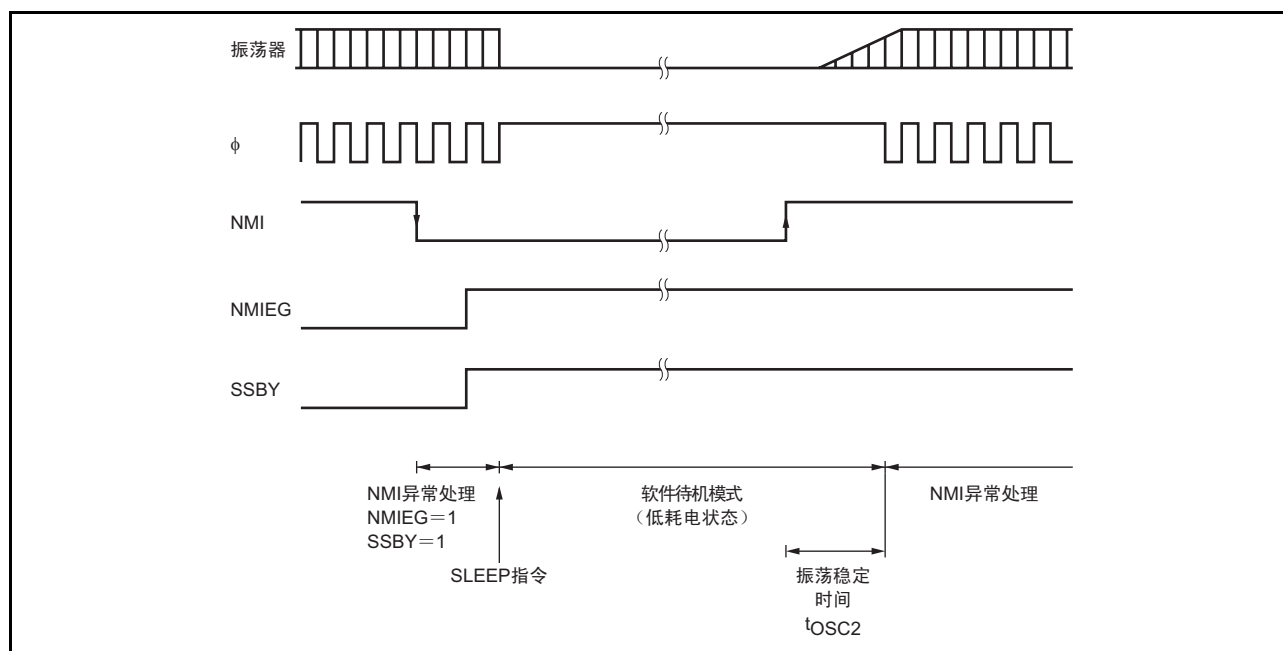


图 20.3 软件待机模式应用实例

20.5 硬件待机模式

20.5.1 向硬件待机模式转移

若 $\overline{\text{STBY}}$ 管脚为低电平，任何模式都会变为硬件待机模式。

在硬件待机模式下，由于停止运行所有功能，CPU 处于复位状态，耗电量明显降低。只要在规定电压的范围内，内部 RAM 的数据保持。I/O 端口为高阻抗状态。

为了保持内部 RAM 的数据，在 $\overline{\text{STBY}}$ 管脚降为低电平前，请将 SYSCR 的 RAME 位清零。在硬件待机模式中，模式管脚（MD2、MD0）的状态不要改变。

20.5.2 硬件待机模式的解除

通过 $\overline{\text{STBY}}$ 管脚和 $\overline{\text{RES}}$ 管脚解除硬件待机模式。 $\overline{\text{RES}}$ 管脚为低电平， $\overline{\text{STBY}}$ 管脚为高电平时，CPU 为复位状态，时钟振荡开始。此时， $\overline{\text{RES}}$ 管脚必须保持低电平直到时钟振荡稳定（晶体振荡时，稳定时间为 8ms 以上）。此后， $\overline{\text{RES}}$ 管脚为高电平，经过复位异常处理状态向程序执行状态转移。

20.5.3 硬件待机模式的时序

(1) 硬件待机模式的转移时序

- SYSCR 的 RAME 位设定为 1 后保持 RAM 内容

如图 20.4 所示，在 $\overline{\text{STBY}}$ 到达下降沿之前，至少 10 个状态使 RES 信号为低电平。在 $\overline{\text{STBY}}$ 到达下降沿之后，RES 至少等待 0ns 才变为高电平。

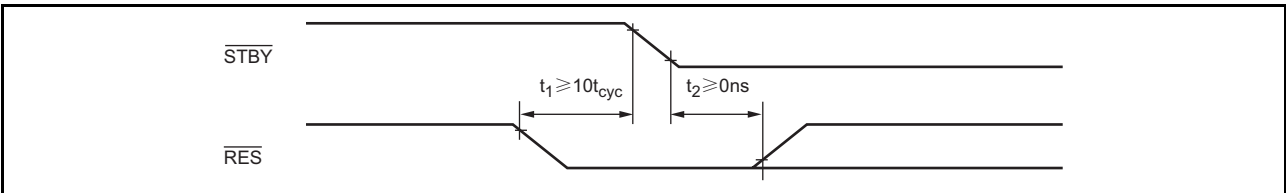


图 20.4 硬件待机模式的转移时序图

- SYSCR 的 RAME 位清零则不需要保持 RAM 的内容时。
上述情况下，RES 信号不必降低。

(2) 硬件待机模式的返回时序

在 $\overline{\text{STBY}}$ 成为高电平之前，大约需要 100ns 或更长时间使 RES 信号成为低电平，执行加电复位操作。

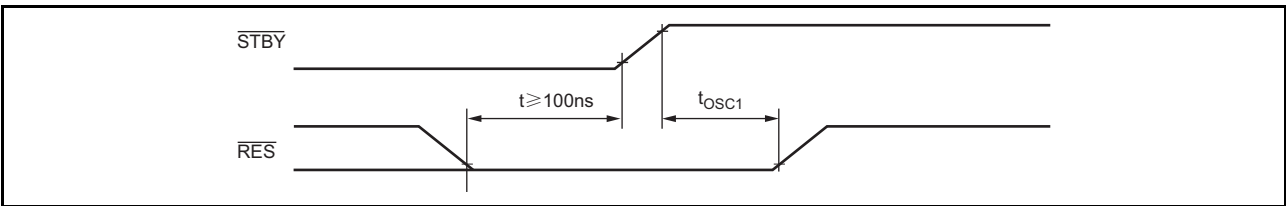


图 20.5 硬件待机模式的返回时序

20.6 模块待机模式

模块待机模式可通过内部外围设备模块单独进行设定。

若将对应 MSTPCR 的 MSTP 位置 1，在总线周期完成时模块停止运行，转向模块待机模式。此时，CPU 继续独立运行。

对应的 MSTP 位清零，模块待机模式解除，在总线周期完成时模块开始运行。在模块待机模式下，除 SCI，（部分 SCI 寄存器的内容是保持的）PWM，HCAN*，以及 A/D 转换器以外的内部模块状态均保持不变。

复位解除后，所有模块转为模块待机模式。

处于模块待机模式的内部外围设备模块，不可读取 / 写入。

【注】* H8S/2280 群不具备此功能。

20.7 监视模式

20.7.1 向监视模式转移

在高速模式或子激活模式，SBYCR 的 SSBY = 1、LPWRCR 的 DTON = 0、TCSR_1 (WDT_1) 的 PSS = 1 的状态下执行 SLEEP 指令，CPU 转向监视模式。

监视模式下停止运行 CPU，停止除了 WDT_1、LCD 以外的外围设备模块，CPU 内部寄存器的内容和内部 RAM 数据，除 SCI，PWM，HCAN*，以及 A/D 转换器和 I/O 端口以外的内部外围设备模块状态均保持不变。

【注】* H8S/2280 群中没有此功能。

20.7.2 监视模式解除

监视模式通过中断 (WOVI1 中断、NMI 管脚、IRQ0 ~ IRQ5 管脚)、 $\overline{\text{RES}}$ 或 $\overline{\text{STBY}}$ 管脚解除。

- 通过中断解除

发生中断请求，监视模式被解除，LPWRCR 的 LSON = 0 或 LSON = 1 时转向子激活模式。当转向高速模式时、经过 SBYCR 的 STS2 ~ STS0 设定的时间后、稳定的时钟被供给整个 LSI 电路，中断异常处理开始。此外， $\overline{\text{IRQ0}} \sim \overline{\text{IRQ5}}$ 与中断对应的许可位清 0 或从内置外围功能开始的中断许可寄存器禁止该中断的受理时或被 CPU 屏蔽时，监视模式不可被解除。

从监视模式向高速模式转移时时钟振荡稳定时间设定请参考「20.4.3 软件待机模式解除后振荡器稳定时间的设定」。

- 通过 $\overline{\text{RES}}$ 管脚解除

有关通过 $\overline{\text{RES}}$ 管脚解除的内容请参考「20.4.2 软件待机模式的解除」。

- 通过 $\overline{\text{STBY}}$ 管脚解除

当 $\overline{\text{STBY}}$ 管脚为低电平，转向硬件待机模式。

20.8 子睡眠模式

20.8.1 向子睡眠模式转移

在子激活模式，SBYCR 的 SSBY = 0、LPWRCR 的 LSON = 1、TCSR_1 (WDT_1) 的 PSS = 1 的状态中执行 SLEEP 指令时，CPU 转向子睡眠模式。

在子睡眠模式，CPU 停止运行。同时 WDT_0、WDT_1、LCD 以外的外围模块停止运行。CPU 的内部寄存器内容、内部 RAM 的数据、SCI、PWM、HCAN* 及除 A/D 转换器外的内置外围功能、I/O 端口的状态保持不变。

【注】* H8S/2280 系列不具备此功能。

20.8.2 子睡眠模式的解除

子睡眠模式通过中断 (WOVI0、WOVI1 中断、NMI 管脚、 $\overline{\text{IRQ0}} \sim \overline{\text{IRQ5}}$ 管脚)、 $\overline{\text{RES}}$ 或 $\overline{\text{STBY}}$ 管脚的信号解除。

- 通过中断解除

发生中断请求，subsleep 模式解除，中断异常处理开始。

假设 $\overline{\text{IRQ0}} \sim \overline{\text{IRQ5}}$ 与中断对应的许可位清 0 或从内置外围功能开始的中断许可寄存器禁止该中断的受理时或被 CPU 屏蔽时，子睡眠模式不被解除。

- 通过 $\overline{\text{RES}}$ 管脚解除

通过 $\overline{\text{RES}}$ 管脚解除，请参考「20.4.2 软件待机模式的解除」。

- 通过 $\overline{\text{STBY}}$ 管脚解除

管脚 $\overline{\text{STBY}}$ 为低电平时，转向硬件待机模式。

20.9 子激活模式

20.9.1 向子激活模式转移

高速模式中，在 SBYCR 的 SSBY = 1、LPWRCR 的 DTON = 1、LSON = 1、TCSR_1 (WDT_1) 的 PSS = 1 的状态执行 SLEEP 指令，则 CPU 转向子激活模式或在监视模式下发生中断后，如果 LPWRCR 的 LSON = 1，转向子激活模式。另外，在子睡眠模式，发生中断，则转向子激活模式。

在子激活模式，CPU 通过子时钟低速运行状态顺序执行程序。子激活状态 WDT_0、WDT_1、LCD 以外的外围模块停止运行。

当 CPU 在子激活模式运行时，SCKCR 的 SCK2 ~ SCK0 位必须设定为 0。

20.9.2 子激活模式的解除

子激活模式通过 SLEEP 指令、或 $\overline{\text{RES}}$ 或 $\overline{\text{STBY}}$ 管脚信号解除。

- 通过 SLEEP 指令解除

在 SBYCR 的 SSBY = 1、LPWRCR 的 DTON = 0、TCSR_1 (WDT_1) 的 PSS = 1 的状态执行 SLEEP 指令，子激活模式解除转向监视模式。在 SBYCR 的 SSBY = 0、LPWRCR 的 LSON = 1、TCSR_1 (WDT_1) 的 PSS = 1 的状态执行 SLEEP 指令，则转向子睡眠模式或在 SBYCR 的 SSBY = 1、LPWRCR 的 DTON = 1、LSON = 0、TCSR_1 (WDT_1) 的 PSS = 1 的状态执行 SLEEP 指令，则直接转向高速模式 (SCK0 ~ SCK2 全为 0)。

- 通过 $\overline{\text{RES}}$ 管脚解除

通过 $\overline{\text{RES}}$ 管脚解除，请参考「20.4.2 软件待机模式的解除」。

- 通过 $\overline{\text{STBY}}$ 管脚解除

如果 $\overline{\text{STBY}}$ 管脚为低电平，则转向硬件待机状态。

20.10 直接转移

CPU 执行程序的运行模式有高速模式、中速模式、子激活模式。当一个在直接转移被执行，即使在高速模式及子激活模式间进行不间断程序运行的转移，也不会产生中断请求。可以通过 LPWRCR 的 DTON 位为 1 并执行 SLEEP 指令进行直接转移。转移后，将开始直接转移中断异常处理。

20.10.1 从高速模式向子激活模式转移

在高速模式下，SBYCR 的 SSBY = 1、LPWRCR 的 LSON = 1、DTON = 1、TCSR_1 (WDT_1) 的 PSS = 1 的状态时，若执行 SLEEP 指令，则直接转向子激活模式。

20.10.2 从 subactive 模式向高速模式直接转移

在子激活模式下，SBYCR 的 SSBY = 1、LPWRCR 的 LSON = 0、DTON = 1、TCSR_1 (WDT_1) 的 PSS = 1 的状态时，若执行 SLEEP 指令，经过 SBYCR 的 STS2 ~ STS0 设定的时间后，直接转向高速模式。

20.11 ϕ 时钟输出控制

通过 SCKCR 的 PSTOP 位及对应端口的 DDR 可以控制 时钟的输出。当 PSTOP 位为 1 时、在总线周期的完成时刻 时钟停止、输出为高电平。当 PSTOP 清零时、时钟允许输出。当对应端口的 DDR 清零、时钟输出被禁止，输入端口被设置。各处理状态的 管脚状态如表 20.4 所示。

表 20.4 各处理状态的 ϕ 管脚状态

寄存器设定值		高速模式、 中速模式	子激活模式	睡眠模式 子睡眠模式	软件待机模式 监视模式 直接转移	硬件待机模式
DDRP	STOP					
0	X	高阻抗	高阻抗	高阻抗	高阻抗	高阻抗
1	0	ϕ 输出	ϕ_{SUB} 输出	ϕ 输出	High 固定	高阻抗
1	1	High 固定	High 固定	High 固定	High 固定	高阻抗

20.12 使用注意事项

20.12.1 I/O 端口状态

监视模式下，I/O 端口的状态保持不变。OPE 位设定为 1 时，地址总线、总线控制信号保持输出。因此，输出高电平时，输出电流耗电不会降低。

20.12.2 振荡稳定待机中的电流消耗

振荡稳定等待过程中电流消耗增加。

20.12.3 内部外围模块中断

在子激活模式停止运行内部外围模块（TPU）时，该中断不可在子激活模式清除。因此，在请求中断状态向子激活模式转移，不可清除 CPU 中断源。事先停止各模块的中断后，执行 SLEEP 指令，转向子激活模式或监视模式。

20.12.4 MSTPCR 的写入

MSTPCR 通过 CPU 写入。

第 21 章 寄存器一览表

在地址一览表中，汇总了有关内部寄存器的地址，寄存器位如何配置，及在各种工作模式下的寄存器状态信息。表的记述方法如下。

1. 寄存器地址一览表（按地址顺序）

- 从分配的低地址寄存器开始顺序记载。
- 根据模块名称分类。
- 表示存取大小。

2. 寄存器位一览表

- 按《寄存器地址一览表（按地址顺序）》的顺序，记载寄存器的位配置。
- 保持位，在位名称部分用「__」表示。
- 位名称部为空白时，表示它的所有寄存器被赋予计数器或保持数据。

3. 各工作模式下的寄存器状态

- 按《寄存器地址一览表（按地址顺序）》的顺序，记载寄存器状态。
- 表示在基本工作模式中的寄存器状态，在内部模块固有复位等的情况下，请参照内置模块这一章节。

21.1 寄存器地址一览表（按地址顺序）

数据总线宽度表示位数。

存取状态数表示指定标准时钟的状态数。

寄存器名称	简称	位数	地址 *1	模块	数据总线宽度	存取状态数
主控制寄存器	MCR	8	H'F800	HCAN*2	16	4
通用状态寄存器	GSR	8	H'F801	HCAN*2	16	4
位结构寄存器	BCR	16	H'F802	HCAN*2	16	4
Mailbox 结构寄存器	MBCR	16	H'F804	HCAN*2	16	4
发送等待寄存器	TXPR	16	H'F806	HCAN*2	16	4
发送等待取消寄存器	TXCR	16	H'F808	HCAN*2	16	4
发送应答寄存器	TXACK	16	H'F80A	HCAN*2	16	4
取消应答寄存器	ABACK	16	H'F80C	HCAN*2	16	4
接收完成寄存器	RXPR	16	H'F80E	HCAN*2	16	4
远程请求寄存器	RFPR	16	H'F810	HCAN*2	16	4
中断寄存器	IRR	16	H'F812	HCAN*2	16	4
mailbox 中断屏蔽寄存器	MBIMR	16	H'F814	HCAN*2	16	4
中断屏蔽寄存器	IMR	16	H'F816	HCAN*2	16	4
接收错误计数器	REC	8	H'F818	HCAN*2	16	4
发送错误计数器	TEC	8	H'F819	HCAN*2	16	4
未读信息状态寄存器	UMSR	16	H'F81A	HCAN*2	16	4
本地受主过滤屏蔽 L	LAFML	16	H'F81C	HCAN*2	16	4
本地受主过滤屏蔽 H	LAFMH	16	H'F81E	HCAN*2	16	4
信息控制 0[1]	MC0[1]	8	H'F820	HCAN*2	16	4
信息控制 0[2]	MC0[2]	8	H'F821	HCAN*2	16	4
信息控制 0[3]	MC0[3]	8	H'F822	HCAN*2	16	4
信息控制 0[4]	MC0[4]	8	H'F823	HCAN*2	16	4
信息控制 0[5]	MC0[5]	8	H'F824	HCAN*2	16	4
信息控制 0[6]	MC0[6]	8	H'F825	HCAN*2	16	4
信息控制 0[7]	MC0[7]	8	H'F826	HCAN*2	16	4
信息控制 0[8]	MC0[8]	8	H'F827	HCAN*2	16	4
信息控制 1[1]	MC1[1]	8	H'F828	HCAN*2	16	4
信息控制 1[2]	MC1[2]	8	H'F829	HCAN*2	16	4
信息控制 1[3]	MC1[3]	8	H'F82A	HCAN*2	16	4
信息控制 1[4]	MC1[4]	8	H'F82B	HCAN*2	16	4
信息控制 1[5]	MC1[5]	8	H'F82C	HCAN*2	16	4
信息控制 1[6]	MC1[6]	8	H'F82D	HCAN*2	16	4
信息控制 1[7]	MC1[7]	8	H'F82E	HCAN*2	16	4

寄存器名称	简称	位数	地址 *1	模块	数据总线宽度	存取状态数
信息控制 1[8]	MC1[8]	8	H'F82F	HCAN*2	16	4
信息控制 2[1]	MC2[1]	8	H'F830	HCAN*2	16	4
信息控制 2[2]	MC2[2]	8	H'F831	HCAN*2	16	4
信息控制 2[3]	MC2[3]	8	H'F832	HCAN*2	16	4
信息控制 2[4]	MC2[4]	8	H'F833	HCAN*2	16	4
信息控制 2[5]	MC2[5]	8	H'F834	HCAN*2	16	4
信息控制 2[6]	MC2[6]	8	H'F835	HCAN*2	16	4
信息控制 2[7]	MC2[7]	8	H'F836	HCAN*2	16	4
信息控制 2[8]	MC2[8]	8	H'F837	HCAN*2	16	4
信息控制 3[1]	MC3[1]	8	H'F838	HCAN*2	16	4
信息控制 3[2]	MC3[2]	8	H'F839	HCAN*2	16	4
信息控制 3[3]	MC3[3]	8	H'F83A	HCAN*2	16	4
信息控制 3[4]	MC3[4]	8	H'F83B	HCAN*2	16	4
信息控制 3[5]	MC3[5]	8	H'F83C	HCAN*2	16	4
信息控制 3[6]	MC3[6]	8	H'F83D	HCAN*2	16	4
信息控制 3[7]	MC3[7]	8	H'F83E	HCAN*2	16	4
信息控制 3[8]	MC3[8]	8	H'F83F	HCAN*2	16	4
信息控制 4[1]	MC4[1]	8	H'F840	HCAN*2	16	4
信息控制 4[2]	MC4[2]	8	H'F841	HCAN*2	16	4
信息控制 4[3]	MC4[3]	8	H'F842	HCAN*2	16	4
信息控制 4[4]	MC4[4]	8	H'F843	HCAN*2	16	4
信息控制 4[5]	MC4[5]	8	H'F844	HCAN*2	16	4
信息控制 4[6]	MC4[6]	8	H'F845	HCAN*2	16	4
信息控制 4[7]	MC4[7]	8	H'F846	HCAN*2	16	4
信息控制 4[8]	MC4[8]	8	H'F847	HCAN*2	16	4
信息控制 5[1]	MC5[1]	8	H'F848	HCAN*2	16	4
信息控制 5[2]	MC5[2]	8	H'F849	HCAN*2	16	4
信息控制 5[3]	MC5[3]	8	H'F84A	HCAN*2	16	4
信息控制 5[4]	MC5[4]	8	H'F84B	HCAN*2	16	4
信息控制 5[5]	MC5[5]	8	H'F84C	HCAN*2	16	4
信息控制 5[6]	MC5[6]	8	H'F84D	HCAN*2	16	4
信息控制 5[7]	MC5[7]	8	H'F84E	HCAN*2	16	4
信息控制 5[8]	MC5[8]	8	H'F84F	HCAN*2	16	4
信息控制 6[1]	MC6[1]	8	H'F850	HCAN*2	16	4
信息控制 6[2]	MC6[2]	8	H'F851	HCAN*2	16	4
信息控制 6[3]	MC6[3]	8	H'F852	HCAN*2	16	4

寄存器名称	简称	位数	地址 *1	模块	数据总线宽度	存取状态数
信息控制 6[4]	MC6[4]	8	H'F853	HCAN*2	16	4
信息控制 6[5]	MC6[5]	8	H'F854	HCAN*2	16	4
信息控制 6[6]	MC6[6]	8	H'F855	HCAN*2	16	4
信息控制 6[7]	MC6[7]	8	H'F856	HCAN*2	16	4
信息控制 6[8]	MC6[8]	8	H'F857	HCAN*2	16	4
信息控制 7[1]	MC7[1]	8	H'F858	HCAN*2	16	4
信息控制 7[2]	MC7[2]	8	H'F859	HCAN*2	16	4
信息控制 7[3]	MC7[3]	8	H'F85A	HCAN*2	16	4
信息控制 7[4]	MC7[4]	8	H'F85B	HCAN*2	16	4
信息控制 7[5]	MC7[5]	8	H'F85C	HCAN*2	16	4
信息控制 7[6]	MC7[6]	8	H'F85D	HCAN*2	16	4
信息控制 7[7]	MC7[7]	8	H'F85E	HCAN*2	16	4
信息控制 7[8]	MC7[8]	8	H'F85F	HCAN*2	16	4
信息控制 8[1]	MC8[1]	8	H'F860	HCAN*2	16	4
信息控制 8[2]	MC8[2]	8	H'F861	HCAN*2	16	4
信息控制 8[3]	MC8[3]	8	H'F862	HCAN*2	16	4
信息控制 8[4]	MC8[4]	8	H'F863	HCAN*2	16	4
信息控制 8[5]	MC8[5]	8	H'F864	HCAN*2	16	4
信息控制 8[6]	MC8[6]	8	H'F865	HCAN*2	16	4
信息控制 8[7]	MC8[7]	8	H'F866	HCAN*2	16	4
信息控制 8[8]	MC8[8]	8	H'F867	HCAN*2	16	4
信息控制 9[1]	MC9[1]	8	H'F868	HCAN*2	16	4
信息控制 9[2]	MC9[2]	8	H'F869	HCAN*2	16	4
信息控制 9[3]	MC9[3]	8	H'F86A	HCAN*2	16	4
信息控制 9[4]	MC9[4]	8	H'F86B	HCAN*2	16	4
信息控制 9[5]	MC9[5]	8	H'F86C	HCAN*2	16	4
信息控制 9[6]	MC9[6]	8	H'F86D	HCAN*2	16	4
信息控制 9[7]	MC9[7]	8	H'F86E	HCAN*2	16	4
信息控制 9[8]	MC9[8]	8	H'F86F	HCAN*2	16	4
信息控制 10[1]	MC10[1]	8	H'F870	HCAN*2	16	4
信息控制 10[2]	MC10[2]	8	H'F871	HCAN*2	16	4
信息控制 10[3]	MC10[3]	8	H'F872	HCAN*2	16	4
信息控制 10[4]	MC10[4]	8	H'F873	HCAN*2	16	4
信息控制 10[5]	MC10[5]	8	H'F874	HCAN*2	16	4
信息控制 10[6]	MC10[6]	8	H'F875	HCAN*2	16	4

寄存器名称	简称	位数	地址 *1	模块	数据总线宽度	存取状态数
信息控制 10[7]	MC10[7]	8	H'F876	HCAN*2	16	4
信息控制 10[8]	MC10[8]	8	H'F877	HCAN*2	16	4
信息控制 11[1]	MC11[1]	8	H'F878	HCAN*2	16	4
信息控制 11[2]	MC11[2]	8	H'F879	HCAN*2	16	4
信息控制 11[3]	MC11[3]	8	H'F87A	HCAN*2	16	4
信息控制 11[4]	MC11[4]	8	H'F87B	HCAN*2	16	4
信息控制 11[5]	MC11[5]	8	H'F87C	HCAN*2	16	4
信息控制 11[6]	MC11[6]	8	H'F87D	HCAN*2	16	4
信息控制 11[7]	MC11[7]	8	H'F87E	HCAN*2	16	4
信息控制 11[8]	MC11[8]	8	H'F87F	HCAN*2	16	4
信息控制 12[1]	MC12[1]	8	H'F880	HCAN*2	16	4
信息控制 12[2]	MC12[2]	8	H'F881	HCAN*2	16	4
信息控制 12[3]	MC12[3]	8	H'F882	HCAN*2	16	4
信息控制 12[4]	MC12[4]	8	H'F883	HCAN*2	16	4
信息控制 12[5]	MC12[5]	8	H'F884	HCAN*2	16	4
信息控制 12[6]	MC12[6]	8	H'F885	HCAN*2	16	4
信息控制 12[7]	MC12[7]	8	H'F886	HCAN*2	16	4
信息控制 12[8]	MC12[8]	8	H'F887	HCAN*2	16	4
信息控制 13[1]	MC13[1]	8	H'F888	HCAN*2	16	4
信息控制 13[2]	MC13[2]	8	H'F889	HCAN*2	16	4
信息控制 13[3]	MC13[3]	8	H'F88A	HCAN*2	16	4
信息控制 13[4]	MC13[4]	8	H'F88B	HCAN*2	16	4
信息控制 13[5]	MC13[5]	8	H'F88C	HCAN*2	16	4
信息控制 13[6]	MC13[6]	8	H'F88D	HCAN*2	16	4
信息控制 13[7]	MC13[7]	8	H'F88E	HCAN*2	16	4
信息控制 13[8]	MC13[8]	8	H'F88F	HCAN*2	16	4
信息控制 14[1]	MC14[1]	8	H'F890	HCAN*2	16	4
信息控制 14[2]	MC14[2]	8	H'F891	HCAN*2	16	4
信息控制 14[3]	MC14[3]	8	H'F892	HCAN*2	16	4
信息控制 14[4]	MC14[4]	8	H'F893	HCAN*2	16	4
信息控制 14[5]	MC14[5]	8	H'F894	HCAN*2	16	4
信息控制 14[6]	MC14[6]	8	H'F895	HCAN*2	16	4
信息控制 14[7]	MC14[7]	8	H'F896	HCAN*2	16	4
信息控制 14[8]	MC14[8]	8	H'F897	HCAN*2	16	4
信息控制 15[1]	MC15[1]	8	H'F898	HCAN*2	16	4
信息控制 15[2]	MC15[2]	8	H'F899	HCAN*2	16	4

寄存器名称	简称	位数	地址 *1	模块	数据总线宽度	存取状态数
信息控制 15[3]	MC15[3]	8	H'F89A	HCAN*2	16	4
信息控制 15[4]	MC15[4]	8	H'F89B	HCAN*2	16	4
信息控制 15[5]	MC15[5]	8	H'F89C	HCAN*2	16	4
信息控制 15[6]	MC15[6]	8	H'F89D	HCAN*2	16	4
信息控制 15[7]	MC15[7]	8	H'F89E	HCAN*2	16	4
信息控制 15[8]	MC15[8]	8	H'F89F	HCAN*2	16	4
信息数据 0[1]	MD0[1]	8	H'F8B0	HCAN*2	16	4
信息数据 0[2]	MD0[2]	8	H'F8B1	HCAN*2	16	4
信息数据 0[3]	MD0[3]	8	H'F8B2	HCAN*2	16	4
信息数据 0[4]	MD0[4]	8	H'F8B3	HCAN*2	16	4
信息数据 0[5]	MD0[5]	8	H'F8B4	HCAN*2	16	4
信息数据 0[6]	MD0[6]	8	H'F8B5	HCAN*2	16	4
信息数据 0[7]	MD0[7]	8	H'F8B6	HCAN*2	16	4
信息数据 0[8]	MD0[8]	8	H'F8B7	HCAN*2	16	4
信息数据 1[1]	MD1[1]	8	H'F8B8	HCAN*2	16	4
信息数据 1[2]	MD1[2]	8	H'F8B9	HCAN*2	16	4
信息数据 1[3]	MD1[3]	8	H'F8BA	HCAN*2	16	4
信息数据 1[4]	MD1[4]	8	H'F8BB	HCAN*2	16	4
信息数据 1[5]	MD1[5]	8	H'F8BC	HCAN*2	16	4
信息数据 1[6]	MD1[6]	8	H'F8BD	HCAN*2	16	4
信息数据 1[7]	MD1[7]	8	H'F8BE	HCAN*2	16	4
信息数据 1[8]	MD1[8]	8	H'F8BF	HCAN*2	16	4
信息数据 2[1]	MD2[1]	8	H'F8C0	HCAN*2	16	4
信息数据 2[2]	MD2[2]	8	H'F8C1	HCAN*2	16	4
信息数据 2[3]	MD2[3]	8	H'F8C2	HCAN*2	16	4
信息数据 2[4]	MD2[4]	8	H'F8C3	HCAN*2	16	4
信息数据 2[5]	MD2[5]	8	H'F8C4	HCAN*2	16	4
信息数据 2[6]	MD2[6]	8	H'F8C5	HCAN*2	16	4
信息数据 2[7]	MD2[7]	8	H'F8C6	HCAN*2	16	4
信息数据 2[8]	MD2[8]	8	H'F8C7	HCAN*2	16	4
信息数据 3[1]	MD3[1]	8	H'F8C8	HCAN*2	16	4
信息数据 3[2]	MD3[2]	8	H'F8C9	HCAN*2	16	4
信息数据 3[3]	MD3[3]	8	H'F8CA	HCAN*2	16	4
信息数据 3[4]	MD3[4]	8	H'F8CB	HCAN*2	16	4
信息数据 3[5]	MD3[5]	8	H'F8CC	HCAN*2	16	4

寄存器名称	简称	位数	地址 *1	模块	数据总线宽度	存取状态数
信息数据 3[6]	MD3[6]	8	H'F8CD	HCAN*2	16	4
信息数据 3[7]	MD3[7]	8	H'F8CE	HCAN*2	16	4
信息数据 3[8]	MD3[8]	8	H'F8CF	HCAN*2	16	4
信息数据 4[1]	MD4[1]	8	H'F8D0	HCAN*2	16	4
信息数据 4[2]	MD4[2]	8	H'F8D1	HCAN*2	16	4
信息数据 4[3]	MD4[3]	8	H'F8D2	HCAN*2	16	4
信息数据 4[4]	MD4[4]	8	H'F8D3	HCAN*2	16	4
信息数据 4[5]	MD4[5]	8	H'F8D4	HCAN*2	16	4
信息数据 4[6]	MD4[6]	8	H'F8D5	HCAN*2	16	4
信息数据 4[7]	MD4[7]	8	H'F8D6	HCAN*2	16	4
信息数据 4[8]	MD4[8]	8	H'F8D7	HCAN*2	16	4
信息数据 5[1]	MD5[1]	8	H'F8D8	HCAN*2	16	4
信息数据 5[2]	MD5[2]	8	H'F8D9	HCAN*2	16	4
信息数据 5[3]	MD5[3]	8	H'F8DA	HCAN*2	16	4
信息数据 5[4]	MD5[4]	8	H'F8DB	HCAN*2	16	4
信息数据 5[5]	MD5[5]	8	H'F8DC	HCAN*2	16	4
信息数据 5[6]	MD5[6]	8	H'F8DD	HCAN*2	16	4
信息数据 5[7]	MD5[7]	8	H'F8DE	HCAN*2	16	4
信息数据 5[8]	MD5[8]	8	H'F8DF	HCAN*2	16	4
信息数据 6[1]	MD6[1]	8	H'F8E0	HCAN*2	16	4
信息数据 6[2]	MD6[2]	8	H'F8E1	HCAN*2	16	4
信息数据 6[3]	MD6[3]	8	H'F8E2	HCAN*2	16	4
信息数据 6[4]	MD6[4]	8	H'F8E3	HCAN*2	16	4
信息数据 6[5]	MD6[5]	8	H'F8E4	HCAN*2	16	4
信息数据 6[6]	MD6[6]	8	H'F8E5	HCAN*2	16	4
信息数据 6[7]	MD6[7]	8	H'F8E6	HCAN*2	16	4
信息数据 6[8]	MD6[8]	8	H'F8E7	HCAN*2	16	4
信息数据 7[1]	MD7[1]	8	H'F8E8	HCAN*2	16	4
信息数据 7[2]	MD7[2]	8	H'F8E9	HCAN*2	16	4
信息数据 7[3]	MD7[3]	8	H'F8EA	HCAN*2	16	4
信息数据 7[4]	MD7[4]	8	H'F8EB	HCAN*2	16	4
信息数据 7[5]	MD7[5]	8	H'F8EC	HCAN*2	16	4
信息数据 7[6]	MD7[6]	8	H'F8ED	HCAN*2	16	4
信息数据 7[7]	MD7[7]	8	H'F8EE	HCAN*2	16	4
信息数据 7[8]	MD7[8]	8	H'F8EF	HCAN*2	16	4
信息数据 8[1]	MD8[1]	8	H'F8F0	HCAN*2	16	4

寄存器名称	简称	位数	地址 *1	模块	数据总线宽度	存取状态数
信息数据 8[2]	MD8[2]	8	H'F8F1	HCAN*2	16	4
信息数据 8[3]	MD8[3]	8	H'F8F2	HCAN*2	16	4
信息数据 8[4]	MD8[4]	8	H'F8F3	HCAN*2	16	4
信息数据 8[5]	MD8[5]	8	H'F8F4	HCAN*2	16	4
信息数据 8[6]	MD8[6]	8	H'F8F5	HCAN*2	16	4
信息数据 8[7]	MD8[7]	8	H'F8F6	HCAN*2	16	4
信息数据 8[8]	MD8[8]	8	H'F8F7	HCAN*2	16	4
信息数据 9[1]	MD9[1]	8	H'F8F8	HCAN*2	16	4
信息数据 9[2]	MD9[2]	8	H'F8F9	HCAN*2	16	4
信息数据 9[3]	MD9[3]	8	H'F8FA	HCAN*2	16	4
信息数据 9[4]	MD9[4]	8	H'F8FB	HCAN*2	16	4
信息数据 9[5]	MD9[5]	8	H'F8FC	HCAN*2	16	4
信息数据 9[6]	MD9[6]	8	H'F8FD	HCAN*2	16	4
信息数据 9[7]	MD9[7]	8	H'F8FE	HCAN*2	16	4
信息数据 9[8]	MD9[8]	8	H'F8FF	HCAN*2	16	4
信息数据 10[1]	MD10[1]	8	H'F900	HCAN*2	16	4
信息数据 10[2]	MD10[2]	8	H'F901	HCAN*2	16	4
信息数据 10[3]	MD10[3]	8	H'F902	HCAN*2	16	4
信息数据 10[4]	MD10[4]	8	H'F903	HCAN*2	16	4
信息数据 10[5]	MD10[5]	8	H'F904	HCAN*2	16	4
信息数据 10[6]	MD10[6]	8	H'F905	HCAN*2	16	4
信息数据 10[7]	MD10[7]	8	H'F906	HCAN*2	16	4
信息数据 10[8]	MD10[8]	8	H'F907	HCAN*2	16	4
信息数据 11[1]	MD11[1]	8	H'F908	HCAN*2	16	4
信息数据 11[2]	MD11[2]	8	H'F909	HCAN*2	16	4
信息数据 11[3]	MD11[3]	8	H'F90A	HCAN*2	16	4
信息数据 11[4]	MD11[4]	8	H'F90B	HCAN*2	16	4
信息数据 11[5]	MD11[5]	8	H'F90C	HCAN*2	16	4
信息数据 11[6]	MD11[6]	8	H'F90D	HCAN*2	16	4
信息数据 11[7]	MD11[7]	8	H'F90E	HCAN*2	16	4
信息数据 11[8]	MD11[8]	8	H'F90F	HCAN*2	16	4
信息数据 12[1]	MD12[1]	8	H'F910	HCAN*2	16	4
信息数据 12[2]	MD12[2]	8	H'F911	HCAN*2	16	4
信息数据 12[3]	MD12[3]	8	H'F912	HCAN*2	16	4
信息数据 12[4]	MD12[4]	8	H'F913	HCAN*2	16	4

寄存器名称	简称	位数	地址 *1	模块	数据总线宽度	存取状态数
信息数据 12[5]	MD12[5]	8	H'F914	HCAN*2	16	4
信息数据 12[6]	MD12[6]	8	H'F915	HCAN*2	16	4
信息数据 12[7]	MD12[7]	8	H'F916	HCAN*2	16	4
信息数据 12[8]	MD12[8]	8	H'F917	HCAN*2	16	4
信息数据 13[1]	MD13[1]	8	H'F918	HCAN*2	16	4
信息数据 13[2]	MD13[2]	8	H'F919	HCAN*2	16	4
信息数据 13[3]	MD13[3]	8	H'F91A	HCAN*2	16	4
信息数据 13[4]	MD13[4]	8	H'F91B	HCAN*2	16	4
信息数据 13[5]	MD13[5]	8	H'F91C	HCAN*2	16	4
信息数据 13[6]	MD13[6]	8	H'F91D	HCAN*2	16	4
信息数据 13[7]	MD13[7]	8	H'F91E	HCAN*2	16	4
信息数据 13[8]	MD13[8]	8	H'F91F	HCAN*2	16	4
信息数据 14[1]	MD14[1]	8	H'F920	HCAN*2	16	4
信息数据 14[2]	MD14[2]	8	H'F921	HCAN*2	16	4
信息数据 14[3]	MD14[3]	8	H'F922	HCAN*2	16	4
信息数据 14[4]	MD14[4]	8	H'F923	HCAN*2	16	4
信息数据 14[5]	MD14[5]	8	H'F924	HCAN*2	16	4
信息数据 14[6]	MD14[6]	8	H'F925	HCAN*2	16	4
信息数据 14[7]	MD14[7]	8	H'F926	HCAN*2	16	4
信息数据 14[8]	MD14[8]	8	H'F927	HCAN*2	16	4
信息数据 15[1]	MD15[1]	8	H'F928	HCAN*2	16	4
信息数据 15[2]	MD15[2]	8	H'F929	HCAN*2	16	4
信息数据 15[3]	MD15[3]	8	H'F92A	HCAN*2	16	4
信息数据 15[4]	MD15[4]	8	H'F92B	HCAN*2	16	4
信息数据 15[5]	MD15[5]	8	H'F92C	HCAN*2	16	4
信息数据 15[6]	MD15[6]	8	H'F92D	HCAN*2	16	4
信息数据 15[7]	MD15[7]	8	H'F92E	HCAN*2	16	4
信息数据 15[8]	MD15[8]	8	H'F92F	HCAN*2	16	4
PWM 控制寄存器 _1	PWCR_1	8	H'FC00	PWM_1	16	4
PWM 输出控制寄存器 _1	PWOCR_1	8	H'FC02	PWM_1	16	4
PWM 优先级寄存器 _1	PWPR_1	8	H'FC04	PWM_1	16	4
PWM 周期寄存器 _1	PWCYR_1	16	H'FC06	PWM_1	16	4
PWM 缓冲寄存器 _1A	PWBFR_1A	16	H'FC08	PWM_1	16	4
PWM 缓冲寄存器 _1C	PWBFR_1C	16	H'FC0A	PWM_1	16	4
PWM 缓冲寄存器 _1E	PWBFR_1E	16	H'FC0C	PWM_1	16	4

寄存器名称	简称	位数	地址 *1	模块	数据总线宽度	存取状态数
PWM 缓冲寄存器 _1G	PWBFR_1G	16	H'FC0E	PWM_1	16	4
PWM 控制寄存器 _2	PWCR_2	8	H'FC10	PWM_2	16	4
PWM 输出控制寄存器 _2	PWOCR_2	8	H'FC12	PWM_2	16	4
PWM 优先级寄存器 _2	PWPR_2	8	H'FC14	PWM_2	16	4
PWM 周期寄存器 _2	PWCYR_2	16	H'FC16	PWM_2	16	4
PWM 缓冲寄存器 _2A	PWBFR_2A	16	H'FC18	PWM_2	16	4
PWM 缓冲寄存器 _2B	PWBFR_2B	16	H'FC1A	PWM_2	16	4
PWM 缓冲寄存器 _2C	PWBFR_2C	16	H'FC1C	PWM_2	16	4
PWM 缓冲寄存器 _2D	PWBFR_2D	16	H'FC1E	PWM_2	16	4
端口 H 数据方向寄存器	PHDDR	8	H'FC20	PORT	16	4
端口 J 数据方向寄存器	PJDDR	8	H'FC21	PORT	16	4
端口 H 数据寄存器	PHDR	8	H'FC24	PORT	16	4
端口 J 数据寄存器	PJDR	8	H'FC25	PORT	16	4
端口 H 寄存器	PORTH	8	H'FC28	PORT	16	4
端口 J 寄存器	PORTJ	8	H'FC29	PORT	16	4
传输端口寄存器	TRPRT	8	H'FC2E	PORT	8	4
LCD 端口控制寄存器	LPCR	8	H'FC30	LCD	16	4
LCD 控制寄存器	LCR	8	H'FC31	LCD	16	4
LCD 控制寄存器 2	LCR2	8	H'FC32	LCD	16	4
模块停止控制寄存器 D	MSTPCRD	8	H'FC60	SYSTEM	8	4
待机控制寄存器	SBYCR	8	H'FDE4	SYSTEM	8	2
系统控制寄存器	SYSCR	8	H'FDE5	SYSTEM	8	2
系统时钟控制寄存器	SCKCR	8	H'FDE6	SYSTEM	8	2
模式控制寄存器	MDCR	8	H'FDE7	SYSTEM	8	2
模块停止控制寄存器 A	MSTPCRA	8	H'FDE8	SYSTEM	8	2
模块停止控制寄存器 B	MSTPCRB	8	H'FDE9	SYSTEM	8	2
模块停止控制寄存器 C	MSTPCRC	8	H'FDEA	SYSTEM	8	2
低功耗控制寄存器	LPWRCR	8	H'FDEC	SYSTEM	8	2
IRQ 感知控制寄存器 H	ISCRH	8	H'FE12	INT	8	2
IRQ 感知控制寄存器 L	ISCRL	8	H'FE13	INT	8	2
IRQ 使能寄存器	IER	8	H'FE14	INT	8	2
IRQ 状态寄存器	ISR	8	H'FE15	INT	8	2
端口 1 数据方向寄存器	P1DDR	8	H'FE30	PORT	8	2
端口 3 数据方向寄存器	P3DDR	8	H'FE32	PORT	8	2
端口 A 数据方向寄存器	PADDR	8	H'FE39	PORT	8	2

寄存器名称	简称	位数	地址 *1	模块	数据总线宽度	存取状态数
端口 B 数据方向寄存器	PBDDR	8	H'FE3A	PORT	8	2
端口 C 数据方向寄存器	PCDDR	8	H'FE3B	PORT	8	2
端口 D 数据方向寄存器	PDDDR	8	H'FE3C	PORT	8	2
端口 F 数据方向寄存器	PFDDR	8	H'FE3E	PORT	8	2
端口 3 漏极开路控制寄存器	P3ODR	8	H'FE46	PORT	8	2
端口 A 漏极开路控制寄存器	PAODR	8	H'FE47	PORT	8	2
端口 B 漏极开路控制寄存器	PBODR	8	H'FE48	PORT	8	2
端口 C 漏极开路控制寄存器	PCODR	8	H'FE49	PORT	8	2
定时器启动寄存器	TSTR	8	H'FEB0	TPU	16	2
定时器同步寄存器	TSYR	8	H'FEB1	TPU	16	2
中断优先级寄存器 A	IPRA	8	H'FEC0	INT	8	2
中断优先级寄存器 B	IPRB	8	H'FEC1	INT	8	2
中断优先级寄存器 C	IPRC	8	H'FEC2	INT	8	2
中断优先级寄存器 D	IPRD	8	H'FEC3	INT	8	2
中断优先级寄存器 E	IPRE	8	H'FEC4	INT	8	2
中断优先级寄存器 F	IPRF	8	H'FEC5	INT	8	2
中断优先级寄存器 G	IPRG	8	H'FEC6	INT	8	2
中断优先级寄存器 J	IPRJ	8	H'FEC9	INT	8	2
中断优先级寄存器 K	IPRK	8	H'FECA	INT	8	2
中断优先级寄存器 M	IPRM	8	H'FECC	INT	8	2
RAM 仿真寄存器	RAMER*2	8	H'FEDB	FLASH (F-ZTAT 版)	8	2
端口 1 数据寄存器	P1DR	8	H'FF00	PORT	8	2
端口 3 数据寄存器	P3DR	8	H'FF02	PORT	8	2
端口 A 数据寄存器	PADR	8	H'FF09	PORT	8	2
端口 B 数据寄存器	PBDR	8	H'FF0A	PORT	8	2
端口 C 数据寄存器	PCDR	8	H'FF0B	PORT	8	2
端口 D 数据寄存器	PDDR	8	H'FF0C	PORT	8	2
端口 F 数据寄存器	PFDR	8	H'FF0E	PORT	8	2
定时器控制寄存器 _0	TCR_0	8	H'FF10	TPU_0	16	2
定时器模式寄存器 _0	TMDR_0	8	H'FF11	TPU_0	16	2
定时器 I/O 控制寄存器 H_0	TIORH_0	8	H'FF12	TPU_0	16	2
定时器 I/O 控制寄存器 L_0	TIORL_0	8	H'FF13	TPU_0	16	2
定时器中断使能寄存器 _0	TIER_0	8	H'FF14	TPU_0	16	2
定时器状态寄存器 _0	TSR_0	8	H'FF15	TPU_0	16	2
定时计数器 H_0	TCNTH_0	8	H'FF16	TPU_0	16	2

寄存器名称	简称	位数	地址 *1	模块	数据总线宽度	存取状态数
定时计数器 L_0	TCNTL_0	8	H'FF17	TPU_0	16	2
定时器通用寄存器 AH_0	TGRAH_0	8	H'FF18	TPU_0	16	2
定时器通用寄存器 AL_0	TGRAL_0	8	H'FF19	TPU_0	16	2
定时器通用寄存器 BH_0	TGRBH_0	8	H'FF1A	TPU_0	16	2
定时器通用寄存器 BL_0	TGRBL_0	8	H'FF1B	TPU_0	16	2
定时器通用寄存器 CH_0	TGRCH_0	8	H'FF1C	TPU_0	16	2
定时器通用寄存器 CL_0	TGRCL_0	8	H'FF1D	TPU_0	16	2
定时器通用寄存器 DH_0	TGRDH_0	8	H'FF1E	TPU_0	16	2
定时器通用寄存器 DL_0	TGRDL_0	8	H'FF1F	TPU_0	16	2
定时器控制寄存器 _1	TCR_1	8	H'FF20	TPU_1	16	2
定时器模式寄存器 _1	TMDR_1	8	H'FF21	TPU_1	16	2
定时器 I/O 控制寄存器 _1	TIOR_1	8	H'FF22	TPU_1	16	2
定时器中断使能寄存器 _1	TIER_1	8	H'FF24	TPU_1	16	2
定时器状态寄存器 _1	TSR_1	8	H'FF25	TPU_1	16	2
定时计数器 H_1	TCNTH_1	8	H'FF26	TPU_1	16	2
定时计数器 L_1	TCNTL_1	8	H'FF27	TPU_1	16	2
定时器通用寄存器 AH_1	TGRAH_1	8	H'FF28	TPU_1	16	2
定时器通用寄存器 AL_1	TGRAL_1	8	H'FF29	TPU_1	16	2
定时器通用寄存器 BH_1	TGRBH_1	8	H'FF2A	TPU_1	16	2
定时器通用寄存器 BL_1	TGRBL_1	8	H'FF2B	TPU_1	16	2
定时器控制寄存器 _2	TCR_2	8	H'FF30	TPU_2	16	2
定时器模式寄存器 _2	TMDR_2	8	H'FF31	TPU_2	16	2
定时器 I/O 控制寄存器 _2	TIOR_2	8	H'FF32	TPU_2	16	2
定时器中断使能寄存器 _2	TIER_2	8	H'FF34	TPU_2	16	2
定时器状态寄存器 _2	TSR_2	8	H'FF35	TPU_2	16	2
定时计数器 H_2	TCNTH_2	8	H'FF36	TPU_2	16	2
定时计数器 L_2	TCNTL_2	8	H'FF37	TPU_2	16	2
定时器通用寄存器 AH_2	TGRAH_2	8	H'FF38	TPU_2	16	2
定时器通用寄存器 AL_2	TGRAL_2	8	H'FF39	TPU_2	16	2
定时器通用寄存器 BH_2	TGRBH_2	8	H'FF3A	TPU_2	16	2
定时器通用寄存器 BL_2	TGRBL_2	8	H'FF3B	TPU_2	16	2
定时器控制 / 状态寄存器 _0	TCSR_0	8	H'FF74	WDT_0	16	2
定时计数器 _0	TCNT_0	8	H'FF75	WDT_0	16	2
复位控制 / 状态寄存器	RSTCSR	8	H'FF77	WDT_0	16	2
串行模式寄存器 _0	SMR_0	8	H'FF78	SCI_0	8	2

寄存器名称	简称	位数	地址 *1	模块	数据总线宽度	存取状态数
波特率寄存器 _0	BRR_0	8	H'FF79	SCI_0	8	2
串行控制寄存器 _0	SCR_0	8	H'FF7A	SCI_0	8	2
传输数据寄存器 _0	TDR_0	8	H'FF7B	SCI_0	8	2
串行状态寄存器 _0	SSR_0	8	H'FF7C	SCI_0	8	2
接收数据寄存器 _0	RDR_0	8	H'FF7D	SCI_0	8	2
智能卡模式寄存器 _0	SCMR_0	8	H'FF7E	SCI_0	8	2
串行模式寄存器 _1	SMR_1	8	H'FF80	SCI_1	8	2
波特率寄存器 _1	BRR_1	8	H'FF81	SCI_1	8	2
串行控制寄存器 _1	SCR_1	8	H'FF82	SCI_1	8	2
传输数据寄存器 _1	TDR_1	8	H'FF83	SCI_1	8	2
串行状态寄存器 _1	SSR_1	8	H'FF84	SCI_1	8	2
接收数据寄存器 _1	RDR_1	8	H'FF85	SCI_1	8	2
智能卡模式寄存器 _1	SCMR_1	8	H'FF86	SCI_1	8	2
A/D 数据寄存器 AH	ADDRAH	8	H'FF90	A/D	8	2
A/D 数据寄存器 AL	ADDRAL	8	H'FF91	A/D	8	2
A/D 数据寄存器 BH	ADDRBH	8	H'FF92	A/D	8	2
A/D 数据寄存器 BL	ADDRBL	8	H'FF93	A/D	8	2
A/D 数据寄存器 CH	ADDRCH	8	H'FF94	A/D	8	2
A/D 数据寄存器 CL	ADDRCL	8	H'FF95	A/D	8	2
A/D 数据寄存器 DH	ADDRDH	8	H'FF96	A/D	8	2
A/D 数据寄存器 DL	ADDRDL	8	H'FF97	A/D	8	2
A/D 控制 / 状态寄存器	ADCSR	8	H'FF98	A/D	8	2
A/D 控制寄存器	ADCR	8	H'FF99	A/D	8	2
定时器控制 / 状态寄存器 _1	TCSR_1	8	H'FFA2	WDT_1	16	2
定时计数器 _1	TCNT_1	8	H'FFA3	WDT_1	16	2
Flash 存储器控制寄存器 1	FLMCR1	8	H'FFA8	FLASH (F-ZTAT 版)	8	2
Flash 存储器控制寄存器 2	FLMCR2	8	H'FFA9	FLASH (F-ZTAT 版)	8	2
块擦除指定寄存器 1	EBR1	8	H'FFAA	FLASH (F-ZTAT 版)	8	2
块擦除指定寄存器 2	EBR2*2	8	H'FFAB	FLASH (F-ZTAT 版)	8	2
Flash 存储器功率控制寄存器	FLPWCR	8	H'FFAC	FLASH (F-ZTAT 版)	8	2
端口 1 寄存器	PORT1	8	H'FFB0	PORT	8	2
端口 3 寄存器	PORT3	8	H'FFB2	PORT	8	2

寄存器名称	简称	位数	地址 *1	模块	数据总线宽度	存取状态数
端口 4 寄存器	PORT4	8	H'FFB3	PORT	8	2
端口 A 寄存器	PORTA	8	H'FFB9	PORT	8	2
端口 B 寄存器	PORTB	8	H'FFBA	PORT	8	2
端口 C 寄存器	PORTC	8	H'FFBB	PORT	8	2
端口 D 寄存器	PORTD	8	H'FFBC	PORT	8	2
端口 F 寄存器	PORTF	8	H'FFBE	PORT	8	2

【注】 *1 表示地址的低 16 位。
 *2 在 H8S/2280 群中，本寄存器被保留。

21.2 寄存器位一览表

内部外围模块的寄存器各位如下。

16 位寄存器每 8 位分 2 段表示。

寄存器简称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	模块
MCR	MCR7	—	MCR5	—	—	MCR2	MCR1	MCR0	HCAN*2
GSR	—	—	—	—	GSR3	GSR2	GSR1	GSR0	
BCR	BCR7	BCR6	BCR5	BCR4	BCR3	BCR2	BCR1	BCR0	
	BCR15	BCR14	BCR13	BCR12	BCR11	BCR10	BCR9	BCR8	
MBCR	MBCR7	MBCR6	MBCR5	MBCR4	MBCR3	MBCR2	MBCR1	—	
	MBCR15	MBCR14	MBCR13	MBCR12	MBCR11	MBCR10	MBCR9	MBCR8	
TXPR	TXPR7	TXPR6	TXPR5	TXPR4	TXPR3	TXPR2	TXPR1	—	
	TXPR15	TXPR14	TXPR13	TXPR12	TXPR11	TXPR10	TXPR9	TXPR8	
TXCR	TXCR7	TXCR6	TXCR5	TXCR4	TXCR3	TXCR2	TXCR1	—	
	TXCR15	TXCR14	TXCR13	TXCR12	TXCR11	TXCR10	TXCR9	TXCR8	
TXACK	TXACK7	TXACK6	TXACK5	TXACK4	TXACK3	TXACK2	TXACK1	—	
	TXACK15	TXACK14	TXACK13	TXACK12	TXACK11	TXACK10	TXACK9	TXACK8	
ABACK	ABACK7	ABACK6	ABACK5	ABACK4	ABACK3	ABACK2	ABACK1	—	
	ABACK15	ABACK14	ABACK13	ABACK12	ABACK11	ABACK10	ABACK9	ABACK8	
RXPR	RXPR7	RXPR6	RXPR5	RXPR4	RXPR3	RXPR2	RXPR1	RXPR0	
	RXPR15	RXPR14	RXPR13	RXPR12	RXPR11	RXPR10	RXPR9	RXPR8	
RFPR	RFPR7	RFPR6	RFPR5	RFPR4	RFPR3	RFPR2	RFPR1	RFPR0	
	RFPR15	RFPR14	RFPR13	RFPR12	RFPR11	RFPR10	RFPR9	RFPR8	
IRR	IRR7	IRR6	IRR5	IRR4	IRR3	IRR2	IRR1	IRR0	
	—	—	—	IRR12	—	—	IRR9	IRR8	
MBIMR	MBIMR7	MBIMR6	MBIMR5	MBIMR4	MBIMR3	MBIMR2	MBIMR1	MBIMR0	
	MBIMR15	MBIMR14	MBIMR13	MBIMR12	MBIMR11	MBIMR10	MBIMR9	MBIMR8	
IMR	IMR7	IMR6	IMR5	IMR4	IMR3	IMR2	IMR1	—	
	—	—	—	IMR12	—	—	IMR9	IMR8	
REC	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
TEC	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
UMSR	UMSR7	UMSR6	UMSR5	UMSR4	UMSR3	UMSR2	UMSR1	UMSR0	
	UMSR15	UMSR14	UMSR13	UMSR12	UMSR11	UMSR10	UMSR9	UMSR8	
LAFML	LAFML7	LAFML6	LAFML5	LAFML4	LAFML3	LAFML2	LAFML1	LAFML0	
	LAFML15	LAFML14	LAFML13	LAFML12	LAFML11	LAFML10	LAFML9	LAFML8	
LAFMH	LAFMH7	LAFMH6	LAFMH5	—	—	—	LAFMH1	LAFMH0	
	LAFMH15	LAFMH14	LAFMH13	LAFMH12	LAFMH11	LAFMH10	LAFMH9	LAFMH8	

寄存器简称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	模块
MC0[1]	—	—	—	—	DLC3	DLC2	DLC1	DLC0	HCAN*2
MC0[2]	—	—	—	—	—	—	—	—	
MC0[3]	—	—	—	—	—	—	—	—	
MC0[4]	—	—	—	—	—	—	—	—	
MC0[5]	ID-20	ID-19	ID-18	RTR	IDE	—	ID-17	ID-16	
MC0[6]	ID-28	ID-27	ID-26	ID-25	ID-24	ID-23	ID-22	ID-21	
MC0[7]	ID-7	ID-6	ID-5	ID-4	ID-3	ID-2	ID-1	ID-0	
MC0[8]	ID-15	ID-14	ID-13	ID-12	ID-11	ID-10	ID-9	ID-8	
MC1[1]	—	—	—	—	DLC3	DLC2	DLC1	DLC0	
MC1[2]	—	—	—	—	—	—	—	—	
MC1[3]	—	—	—	—	—	—	—	—	
MC1[4]	—	—	—	—	—	—	—	—	
MC1[5]	ID-20	ID-19	ID-18	RTR	IDE	—	ID-17	ID-16	
MC1[6]	ID-28	ID-27	ID-26	ID-25	ID-24	ID-23	ID-22	ID-21	
MC1[7]	ID-7	ID-6	ID-5	ID-4	ID-3	ID-2	ID-1	ID-0	
MC1[8]	ID-15	ID-14	ID-13	ID-12	ID-11	ID-10	ID-9	ID-8	
MC2[1]	—	—	—	—	DLC3	DLC2	DLC1	DLC0	
MC2[2]	—	—	—	—	—	—	—	—	
MC2[3]	—	—	—	—	—	—	—	—	
MC2[4]	—	—	—	—	—	—	—	—	
MC2[5]	ID-20	ID-19	ID-18	RTR	IDE	—	ID-17	ID-16	
MC2[6]	ID-28	ID-27	ID-26	ID-25	ID-24	ID-23	ID-22	ID-21	
MC2[7]	ID-7	ID-6	ID-5	ID-4	ID-3	ID-2	ID-1	ID-0	
MC2[8]	ID-15	ID-14	ID-13	ID-12	ID-11	ID-10	ID-9	ID-8	
MC3[1]	—	—	—	—	DLC3	DLC2	DLC1	DLC0	
MC3[2]	—	—	—	—	—	—	—	—	
MC3[3]	—	—	—	—	—	—	—	—	
MC3[4]	—	—	—	—	—	—	—	—	
MC3[5]	ID-20	ID-19	ID-18	RTR	IDE	—	ID-17	ID-16	
MC3[6]	ID-28	ID-27	ID-26	ID-25	ID-24	ID-23	ID-22	ID-21	
MC3[7]	ID-7	ID-6	ID-5	ID-4	ID-3	ID-2	ID-1	ID-0	
MC3[8]	ID-15	ID-14	ID-13	ID-12	ID-11	ID-10	ID-9	ID-8	
MC4[1]	—	—	—	—	DLC3	DLC2	DLC1	DLC0	
MC4[2]	—	—	—	—	—	—	—	—	
MC4[3]	—	—	—	—	—	—	—	—	

寄存器简称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	模块
MC4[4]	—	—	—	—	—	—	—	—	HCAN*2
MC4[5]	ID-20	ID-19	ID-18	RTR	IDE	—	ID-17	ID-16	
MC4[6]	ID-28	ID-27	ID-26	ID-25	ID-24	ID-23	ID-22	ID-21	
MC4[7]	ID-7	ID-6	ID-5	ID-4	ID-3	ID-2	ID-1	ID-0	
MC4[8]	ID-15	ID-14	ID-13	ID-12	ID-11	ID-10	ID-9	ID-8	
MC5[1]	—	—	—	—	DLC3	DLC2	DLC1	DLC0	
MC5[2]	—	—	—	—	—	—	—	—	
MC5[3]	—	—	—	—	—	—	—	—	
MC5[4]	—	—	—	—	—	—	—	—	
MC5[5]	ID-20	ID-19	ID-18	RTR	IDE	—	ID-17	ID-16	
MC5[6]	ID-28	ID-27	ID-26	ID-25	ID-24	ID-23	ID-22	ID-21	
MC5[7]	ID-7	ID-6	ID-5	ID-4	ID-3	ID-2	ID-1	ID-0	
MC5[8]	ID-15	ID-14	ID-13	ID-12	ID-11	ID-10	ID-9	ID-8	
MC6[1]	—	—	—	—	DLC3	DLC2	DLC1	DLC0	
MC6[2]	—	—	—	—	—	—	—	—	
MC6[3]	—	—	—	—	—	—	—	—	
MC6[4]	—	—	—	—	—	—	—	—	
MC6[5]	ID-20	ID-19	ID-18	RTR	IDE	—	ID-17	ID-16	
MC6[6]	ID-28	ID-27	ID-26	ID-25	ID-24	ID-23	ID-22	ID-21	
MC6[7]	ID-7	ID-6	ID-5	ID-4	ID-3	ID-2	ID-1	ID-0	
MC6[8]	ID-15	ID-14	ID-13	ID-12	ID-11	ID-10	ID-9	ID-8	
MC7[1]	—	—	—	—	DLC3	DLC2	DLC1	DLC0	
MC7[2]	—	—	—	—	—	—	—	—	
MC7[3]	—	—	—	—	—	—	—	—	
MC7[4]	—	—	—	—	—	—	—	—	
MC7[5]	ID-20	ID-19	ID-18	RTR	IDE	—	ID-17	ID-16	
MC7[6]	ID-28	ID-27	ID-26	ID-25	ID-24	ID-23	ID-22	ID-21	
MC7[7]	ID-7	ID-6	ID-5	ID-4	ID-3	ID-2	ID-1	ID-0	
MC7[8]	ID-15	ID-14	ID-13	ID-12	ID-11	ID-10	ID-9	ID-8	
MC8[1]	—	—	—	—	DLC3	DLC2	DLC1	DLC0	
MC8[2]	—	—	—	—	—	—	—	—	
MC8[3]	—	—	—	—	—	—	—	—	
MC8[4]	—	—	—	—	—	—	—	—	
MC8[5]	ID-20	ID-19	ID-18	RTR	IDE	—	ID-17	ID-16	
MC8[6]	ID-28	ID-27	ID-26	ID-25	ID-24	ID-23	ID-22	ID-21	
MC8[7]	ID-7	ID-6	ID-5	ID-4	ID-3	ID-2	ID-1	ID-0	

寄存器简称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	模块
MC8[8]	ID-15	ID-14	ID-13	ID-12	ID-11	ID-10	ID-9	ID-8	HCAN*2
MC9[1]	—	—	—	—	DLC3	DLC2	DLC1	DLC0	
MC9[2]	—	—	—	—	—	—	—	—	
MC9[3]	—	—	—	—	—	—	—	—	
MC9[4]	—	—	—	—	—	—	—	—	
MC9[5]	ID-20	ID-19	ID-18	RTR	IDE	—	ID-17	ID-16	
MC9[6]	ID-28	ID-27	ID-26	ID-25	ID-24	ID-23	ID-22	ID-21	
MC9[7]	ID-7	ID-6	ID-5	ID-4	ID-3	ID-2	ID-1	ID-0	
MC9[8]	ID-15	ID-14	ID-13	ID-12	ID-11	ID-10	ID-9	ID-8	
MC10[1]	—	—	—	—	DLC3	DLC2	DLC1	DLC0	
MC10[2]	—	—	—	—	—	—	—	—	
MC10[3]	—	—	—	—	—	—	—	—	
MC10[4]	—	—	—	—	—	—	—	—	
MC10[5]	ID-20	ID-19	ID-18	RTR	IDE	—	ID-17	ID-16	
MC10[6]	ID-28	ID-27	ID-26	ID-25	ID-24	ID-23	ID-22	ID-21	
MC10[7]	ID-7	ID-6	ID-5	ID-4	ID-3	ID-2	ID-1	ID-0	
MC10[8]	ID-15	ID-14	ID-13	ID-12	ID-11	ID-10	ID-9	ID-8	
MC11[1]	—	—	—	—	DLC3	DLC2	DLC1	DLC0	
MC11[2]	—	—	—	—	—	—	—	—	
MC11[3]	—	—	—	—	—	—	—	—	
MC11[4]	—	—	—	—	—	—	—	—	
MC11[5]	ID-20	ID-19	ID-18	RTR	IDE	—	ID-17	ID-16	
MC11[6]	ID-28	ID-27	ID-26	ID-25	ID-24	ID-23	ID-22	ID-21	
MC11[7]	ID-7	ID-6	ID-5	ID-4	ID-3	ID-2	ID-1	ID-0	
MC11[8]	ID-15	ID-14	ID-13	ID-12	ID-11	ID-10	ID-9	ID-8	
MC12[1]	—	—	—	—	DLC3	DLC2	DLC1	DLC0	
MC12[2]	—	—	—	—	—	—	—	—	
MC12[3]	—	—	—	—	—	—	—	—	
MC12[4]	—	—	—	—	—	—	—	—	
MC12[5]	ID-20	ID-19	ID-18	RTR	IDE	—	ID-17	ID-16	
MC12[6]	ID-28	ID-27	ID-26	ID-25	ID-24	ID-23	ID-22	ID-21	
MC12[7]	ID-7	ID-6	ID-5	ID-4	ID-3	ID-2	ID-1	ID-0	
MC12[8]	ID-15	ID-14	ID-13	ID-12	ID-11	ID-10	ID-9	ID-8	
MC13[1]	—	—	—	—	DLC3	DLC2	DLC1	DLC0	
MC13[2]	—	—	—	—	—	—	—	—	

寄存器简称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	模块
MC13[3]	—	—	—	—	—	—	—	—	HCAN*2
MC13[4]	—	—	—	—	—	—	—	—	
MC13[5]	ID-20	ID-19	ID-18	RTR	IDE	—	ID-17	ID-16	
MC13[6]	ID-28	ID-27	ID-26	ID-25	ID-24	ID-23	ID-22	ID-21	
MC13[7]	ID-7	ID-6	ID-5	ID-4	ID-3	ID-2	ID-1	ID-0	
MC13[8]	ID-15	ID-14	ID-13	ID-12	ID-11	ID-10	ID-9	ID-8	
MC14[1]	—	—	—	—	DLC3	DLC2	DLC1	DLC0	
MC14[2]	—	—	—	—	—	—	—	—	
MC14[3]	—	—	—	—	—	—	—	—	
MC14[4]	—	—	—	—	—	—	—	—	
MC14[5]	ID-20	ID-19	ID-18	RTR	IDE	—	ID-17	ID-16	
MC14[6]	ID-28	ID-27	ID-26	ID-25	ID-24	ID-23	ID-22	ID-21	
MC14[7]	ID-7	ID-6	ID-5	ID-4	ID-3	ID-2	ID-1	ID-0	
MC14[8]	ID-15	ID-14	ID-13	ID-12	ID-11	ID-10	ID-9	ID-8	
MC15[1]	—	—	—	—	DLC3	DLC2	DLC1	DLC0	
MC15[2]	—	—	—	—	—	—	—	—	
MC15[3]	—	—	—	—	—	—	—	—	
MC15[4]	—	—	—	—	—	—	—	—	
MC15[5]	ID-20	ID-19	ID-18	RTR	IDE	—	ID-17	ID-16	
MC15[6]	ID-28	ID-27	ID-26	ID-25	ID-24	ID-23	ID-22	ID-21	
MC15[7]	ID-7	ID-6	ID-5	ID-4	ID-3	ID-2	ID-1	ID-0	
MC15[8]	ID-15	ID-14	ID-13	ID-12	ID-11	ID-10	ID-9	ID-8	
MD0[1]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD0[2]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD0[3]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD0[4]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD0[5]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD0[6]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD0[7]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD0[8]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD1[1]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD1[2]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD1[3]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD1[4]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD1[5]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

寄存器简称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	模块
MD1[6]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	HCAN*2
MD1[7]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD1[8]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD2[1]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD2[2]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD2[3]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD2[4]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD2[5]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD2[6]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD2[7]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD2[8]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD3[1]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD3[2]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD3[3]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD3[4]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD3[5]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD3[6]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD3[7]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD3[8]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD4[1]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD4[2]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD4[3]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD4[4]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD4[5]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD4[6]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD4[7]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD4[8]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD5[1]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD5[2]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD5[3]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD5[4]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD5[5]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD5[6]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD5[7]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD5[8]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

寄存器简称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	模块
MD6[1]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	HCAN*2
MD6[2]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD6[3]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD6[4]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD6[5]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD6[6]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD6[7]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD6[8]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD7[1]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD7[2]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD7[3]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD7[5]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD7[6]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD7[7]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD7[8]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD8[1]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD8[2]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD8[3]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD8[4]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD8[5]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD8[6]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD8[7]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD8[8]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD9[1]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD9[2]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD9[3]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD9[4]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD9[5]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD9[6]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD9[7]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD9[8]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD10[1]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD10[2]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD10[3]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD10[4]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD10[5]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

寄存器简称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	模块
MD10[6]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	HCAN*2
MD10[7]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD10[8]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD11[1]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD11[2]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD11[3]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD11[4]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD11[5]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD11[6]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD11[7]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD11[8]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD12[1]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD12[2]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD12[3]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD12[4]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD12[5]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD12[6]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD12[7]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD12[8]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD13[1]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD13[2]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD13[3]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD13[4]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD13[5]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD13[6]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD13[7]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD13[8]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD14[1]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD14[2]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD14[3]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD14[4]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD14[5]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD14[6]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD14[7]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD14[8]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

寄存器简称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	模块
MD15[1]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	HCAN*2
MD15[2]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD15[3]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD15[4]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD15[5]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD15[6]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD15[7]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
MD15[8]	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
PWCR_1	—	—	IE	CMF	CST	CKS2	CKS1	CKS0	PWM_1
PWOCR_1	OE1H	OE1G	OE1F	OE1E	OE1D	OE1C	OE1B	OE1A	
PWPR_1	OPS1H	OPS1G	OPS1F	OPS1E	OPS1D	OPS1C	OPS1B	OPS1A	
PWCYR_1	—	—	—	—	—	—	Bit9	Bit8	
	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
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	PWM_2
PWOCR_2	OE2H	OE2G	OE2F	OE2E	OE2D	OE2C	OE2B	OE2A	
PWPR_2	OPS2H	OPS2G	OPS2F	OPS2E	OPS2D	OPS2C	OPS2B	OPS2A	PWM_2
PWCYR_2	—	—	—	—	—	—	Bit9	Bit8	
	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
PWBFR_2A	—	—	—	TDS	—	—	DT9	DT8	
	DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0	
PWBFR_2B	—	—	—	TDS	—	—	DT9	DT8	
	DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0	
PWBFR_2C	—	—	—	TDS	—	—	DT9	DT8	
	DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0	
PWBFR_2D	—	—	—	TDS	—	—	DT9	DT8	
	DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0	
PHDDR	PH7DDR	PH6DDR	PH5DDR	PH4DDR	PH3DDR	PH2DDR	PH1DDR	PH0DDR	PORT
PJDDR	PJ7DDR	PJ6DDR	PJ5DDR	PJ4DDR	PJ3DDR	PJ2DDR	PJ1DDR	PJ0DDR	

寄存器简称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	模块
PHDR	PH7DR	PH6DR	PH5DR	PH4DR	PH3DR	PH2DR	PH1DR	PH0DR	PORT
PJDR	PJ7DR	PJ6DR	PJ5DR	PJ4DR	PJ3DR	PJ2DR	PJ1DR	PJ0DR	
PORTH	PH7	PH6	PH5	PH4	PH3	PH2	PH1	PH0	
PORTJ	PJ7	PJ6	PJ5	PJ4	PJ3	PJ2	PJ1	PJ0	
TRPRT	—	—	—	—	—	—	TRPB	TRPA	
LPCR	DTS1	DTS0	CMX	—	SGS3	SGS2	SGS1	SGS0	LCD
LCR	—	PSW	ACT	DISP	CKS3	CKS2	CKS1	CKS0	
LCR2	LCDAB	—	—	—	—	—	—	—	
MSTPCRD	MSTPD7	MSTPD6	—	—	—	—	—	—	SYSTEM
SBYCR	SSBY	STS2	STS1	STS0	—	—	—	—	
SYSCR	—	—	INTM1	INTM0	NMIEG	—	—	RAME	
SCKCR	PSTOP	—	—	—	STCS	SCK2	SCK1	SCK0	
MDCR	—	—	—	—	—	MDS2	—	MDS0	
MSTPCRA	MSTPA7	MSTPA6	MSTPA5	MSTPA4	MSTPA3	MSTPA2	MSTPA1	MSTPA0	
MSTPCRB	MSTPB7	MSTPB6	MSTPB5	MSTPB4	MSTPB3	MSTPB2	MSTPB1	MSTPB0	
MSTPCRC	MSTPC7	MSTPC6	MSTPC5	MSTPC4	MSTPC3	MSTPC2	MSTPC1	MSTPC0	
LPWRCR	DTON	LSON	—	SUBSTP	RFCUT	—	STC1	STC0	
ISCRH	—	—	—	—	IRQ5SCB	IRQ5SCA	IRQ4SCB	IRQ4SCA	INT
ISCR L	IRQ3SCB	IRQ3SCA	IRQ2SCB	IRQ2SCA	IRQ1SCB	IRQ1SCA	IRQ0SCB	IRQ0SCA	
IER	—	—	IRQ5E	IRQ4E	IRQ3E	IRQ2E	IRQ1E	IRQ0E	
ISR	—	—	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F	
P1DDR	P17DDR	P16DDR	P15DDR	P14DDR	P13DDR	P12DDR	P11DDR	P10DDR	PORT
P3DDR	—	—	P35DDR	P34DDR	P33DDR	P32DDR	P31DDR	P30DDR	
PADDR	PA7DDR	PA6DDR	PA5DDR	PA4DDR	PA3DDR	PA2DDR	PA1DDR	PA0DDR	
PBDDR	PB7DDR	PB6DDR	PB5DDR	PB4DDR	PB3DDR	PB2DDR	PB1DDR	PB0DDR	
PCDDR	PC7DDR	PC6DDR	PC5DDR	PC4DDR	PC3DDR	PC2DDR	PC1DDR	PC0DDR	
PDDDR	PD7DDR	PD6DDR	PD5DDR	PD4DDR	—	—	—	—	
PFDDR	PF7DDR	PF6DDR	PF5DDR	PF4DDR	PF3DDR	PF2DDR	PF1DDR*3	PF0DDR*3	
P3ODR	—	—	P35ODR	P34ODR	P33ODR	P32ODR	P31ODR	P30ODR	
PAODR	PA7ODR	PA6ODR	PA5ODR	PA4ODR	PA3ODR	PA2ODR	PA1ODR	PA0ODR	
PBODR	PB7ODR	PB6ODR	PB5ODR	PB4ODR	PB3ODR	PB2ODR	PB1ODR	PB0ODR	
PCODR	PC7ODR	PC6ODR	PC5ODR	PC4ODR	PC3ODR	PC2ODR	PC1ODR	PC0ODR	TPU 共通
TSTR	—	—	—	—	—	CST2	CST1	CST0	
TSYR	—	—	—	—	—	SYNC2	SYNC1	SYNC0	

寄存器简称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	模块
IPRA	—	IPR6	IPR5	IPR4	—	IPR2	IPR1	IPR0	INT
IPRB	—	IPR6	IPR5	IPR4	—	IPR2	IPR1	IPR0	
IPRC	—	IPR6	IPR5	IPR4	—	IPR2	IPR1	IPR0	
IPRD	—	IPR6	IPR5	IPR4	—	IPR2	IPR1	IPR0	
IPRE	—	IPR6	IPR5	IPR4	—	IPR2	IPR1	IPR0	
IPRF	—	IPR6	IPR5	IPR4	—	IPR2	IPR1	IPR0	
IPRG	—	IPR6	IPR5	IPR4	—	IPR2	IPR1	IPR0	
IPRJ	—	IPR6	IPR5	IPR4	—	IPR2	IPR1	IPR0	
IPRK	—	IPR6	IPR5	IPR4	—	IPR2	IPR1	IPR0	
IPRM	—	IPR6	IPR5	IPR4	—	IPR2	IPR1	IPR0	
RAMER*2	—	—	—	—	RAMS	RAM2	RAM1	RAM0	FLASH (F-ZTAT 版)
P1DR	P17DR	P16DR	P15DR	P14DR	P13DR	P12DR	P11DR	P10DR	PORT
P3DR	—	—	P35DR	P34DR	P33DR	P32DR	P31DR	P30DR	
PADR	PA7DR	PA6DR	PA5DR	PA4DR	PA3DR	PA2DR	PA1DR	PA0DR	
PBDR	PB7DR	PB6DR	PB5DR	PB4DR	PB3DR	PB2DR	PB1DR	PB0DR	
PCDR	PC7DR	PC6DR	PC5DR	PC4DR	PC3DR	PC2DR	PC1DR	PC0DR	
PDDR	PD7DR	PD6DR	PD5DR	PD4DR	—	—	—	—	
PFDR	—	PF6DR	PF5DR	PF4DR	PF3DR	PF2DR	PF1DR*3	PF0DR*3	
TCR_0	CCLR2	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	TPU_0
TMDR_0	—	—	BFB	BFA	MD3	MD2	MD1	MD0	
TIORH_0	IOB3	IOB2	IOB1	IOB0	IOA3	IOA2	IOA1	IOA0	TPU_0
TIORL_0	IOD3	IOD2	IOD1	IOD0	IOC3	IOC2	IOC1	IOC0	
TIER_0	TTGE	—	—	TCIEV	TGIED	TGIEC	TGIEB	TGIEA	
TSR_0	—	—	—	TCFV	TGFD	TGFC	TGFB	TGFA	
TCNTH_0	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	
TCNTL_0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
TGRAH_0	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	
TGRAL_0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
TGRBH_0	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	
TGRBL_0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
TGRCH_0	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	TPU_1
TGRCL_0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
TGRDH_0	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	
TGRDL_0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
TCR_1	—	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	

寄存器简称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	模块
TMDR_1	—	—	—	—	MD3	MD2	MD1	MD0	TPU_1
TIOR_1	IOB3	IOB2	IOB1	IOB0	IOA3	IOA2	IOA1	IOA0	
TIER_1	TTGE	—	TCIEU	TCIEV	—	—	TGIEB	TGIEA	
TSR_1	TCFD	—	TCFU	TCFV	—	—	TGFB	TGFA	
TCNTH_1	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	
TCNTL_1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
TGRAH_1	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	
TGRAL_1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
TGRBH_1	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	
TGRBL_1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
TCR_2	—	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	TPU_2
TMDR_2	—	—	—	—	MD3	MD2	MD1	MD0	
TIOR_2	IOB3	IOB2	IOB1	IOB0	IOA3	IOA2	IOA1	IOA0	
TIER_2	TTGE	—	TCIEU	TCIEV	—	—	TGIEB	TGIEA	
TSR_2	TCFD	—	TCFU	TCFV	—	—	TGFB	TGFA	
TCNTH_2	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	
TCNTL_2	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
TGRAH_2	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	
TGRAL_2	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
TGRBH_2	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8	
TGRBL_2	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	TPU_2
TCSR_0	OVF	WT/ $\overline{\text{IT}}$	TME	—	—	CKS2	CKS1	CKS0	
TCNT_0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	WDT_0
RSTCSR	WOVF	RSTE	RSTS	—	—	—	—	—	
SMR_0*1	C/ $\overline{\text{A}}$ (GM)	CHR (BLK)	PE (PE)	O/ $\overline{\text{E}}$ (O/ $\overline{\text{E}}$)	STOP (BCP1)	MP (BCP0)	CKS1 (CKS1)	CKS0 (CKS0)	SCI_0
BRR_0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
SCR_0	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	
TDR_0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
SSR_0*1	TDRE (TDRE)	RDRF (RDRF)	ORER (ORER)	FER (ERS)	PER (PER)	TEND (TEND)	MPB (MPB)	MPBT (MPBT)	
RDR_0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
SCMR_0	—	—	—	—	SDIR	SINV	—	SMIF	
SMR_1*1	C/ $\overline{\text{A}}$ (GM)	CHR (BLK)	PE (PE)	O/ $\overline{\text{E}}$ (O/ $\overline{\text{E}}$)	STOP (BCP1)	MP (BCP0)	CKS1 (CKS1)	CKS0 (CKS0)	
BRR_1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	

寄存器简称	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	模块
SCR_1	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	SCI_1
TDR_1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
SSR_1*1	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT	
	(TDRE)	(RDRF)	(ORER)	(ERS)	(PER)	(TEND)	(MPB)	(MPBT)	
RDR_1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
SCMR_1	—	—	—	—	SDIR	SINV	—	SMIF	
ADDRAH	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	A/D
ADDRAL	AD1	AD0	—	—	—	—	—	—	
ADDRBH	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
ADDRBL	AD1	AD0	—	—	—	—	—	—	
ADDRCH	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
ADDRCL	AD1	AD0	—	—	—	—	—	—	
ADDRDH	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
ADDRDL	AD1	AD0	—	—	—	—	—	—	
ADCSR	ADF	ADIE	ADST	SCAN	—	CH2	CH1	CH0	
ADCR	TRGS1	TRGS0	—	—	CKS1	CKS0	—	—	
TCSR_1	OVF	WT/IT	TME	PSS	RST/ NMI	CKS2	CKS1	CKS0	WDT_1
TCNT_1	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	
FLMCR1	FWE	SWE	ESU	PSU	EV	PV	E	P	FLASH (F-ZTAT 版)
FLMCR2	FLER	—	—	—	—	—	—	—	
EBR1	EB7	EB6	EB5	EB4	EB3	EB2	EB1	EB0	
EBR2*2	—	—	—	—	—	—	EB9	EB8	
FLPWCR	PDWND	—	—	—	—	—	—	—	
PORT1	P17	P16	P15	P14	P13	P12	P11	P10	PORT
PORT3	—	—	P35	P34	P33	P32	P31	P30	
PORT4	P47	P46	P45	P44	P43	P42	P41	P40	
PORTA	PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0	
PORTB	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0	
PORTC	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0	
PORTD	PD7	PD6	PD5	PD4	—	—	—	—	
PORTF	PF7	PF6	PF5	PF4	PF3	PF2	PF1*3	PF0*3	

【注】 *1 在通常模式和智能卡接口模式下，有一部分位不同。

() 中的内容表示在智能卡接口模式下的位功能。

*2 在 H8S/2280 群中，本寄存器被保留。

*3 在 H8S/2282 群中，为保留位。

21.3 各工作模式下的寄存器状态

寄存器简称	复位	高速	中速	睡眠	模块停止	监视	子激活	子睡眠	软件待机	硬件待机	模块
MCR	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	HCAN*
GSR	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
BCR	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
MBCR	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
TXPR	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
TXCR	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
TXACK	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
ABACK	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
RXPR	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
RFPR	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
IRR	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
MBIMR	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
IMR	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
REC	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
TEC	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
UMSR	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
LAFML	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
LAFMH	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
MC0[1]	—	—	—	—	—	—	—	—	—	—	
MC0[2]	—	—	—	—	—	—	—	—	—	—	
MC0[3]	—	—	—	—	—	—	—	—	—	—	
MC0[4]	—	—	—	—	—	—	—	—	—	—	
MC0[5]	—	—	—	—	—	—	—	—	—	—	
MC0[6]	—	—	—	—	—	—	—	—	—	—	
MC0[7]	—	—	—	—	—	—	—	—	—	—	
MC0[8]	—	—	—	—	—	—	—	—	—	—	
MC1[1]	—	—	—	—	—	—	—	—	—	—	
MC1[2]	—	—	—	—	—	—	—	—	—	—	
MC1[3]	—	—	—	—	—	—	—	—	—	—	
MC1[4]	—	—	—	—	—	—	—	—	—	—	
MC1[5]	—	—	—	—	—	—	—	—	—	—	
MC1[6]	—	—	—	—	—	—	—	—	—	—	
MC1[7]	—	—	—	—	—	—	—	—	—	—	
MC1[8]	—	—	—	—	—	—	—	—	—	—	

寄存器简称	复位	高速	中速	睡眠	模块停止	监视	子激活	子睡眠	软件待机	硬件待机	模块
MC2[1]	—	—	—	—	—	—	—	—	—	—	HCAN*
MC2[2]	—	—	—	—	—	—	—	—	—	—	
MC2[3]	—	—	—	—	—	—	—	—	—	—	
MC2[4]	—	—	—	—	—	—	—	—	—	—	
MC2[5]	—	—	—	—	—	—	—	—	—	—	
MC2[6]	—	—	—	—	—	—	—	—	—	—	
MC2[7]	—	—	—	—	—	—	—	—	—	—	
MC2[8]	—	—	—	—	—	—	—	—	—	—	
MC3[1]	—	—	—	—	—	—	—	—	—	—	
MC3[2]	—	—	—	—	—	—	—	—	—	—	
MC3[3]	—	—	—	—	—	—	—	—	—	—	
MC3[4]	—	—	—	—	—	—	—	—	—	—	
MC3[5]	—	—	—	—	—	—	—	—	—	—	
MC3[6]	—	—	—	—	—	—	—	—	—	—	
MC3[7]	—	—	—	—	—	—	—	—	—	—	
MC3[8]	—	—	—	—	—	—	—	—	—	—	
MC4[1]	—	—	—	—	—	—	—	—	—	—	
MC4[2]	—	—	—	—	—	—	—	—	—	—	
MC4[3]	—	—	—	—	—	—	—	—	—	—	
MC4[4]	—	—	—	—	—	—	—	—	—	—	
MC4[5]	—	—	—	—	—	—	—	—	—	—	
MC4[6]	—	—	—	—	—	—	—	—	—	—	
MC4[7]	—	—	—	—	—	—	—	—	—	—	
MC4[8]	—	—	—	—	—	—	—	—	—	—	
MC5[1]	—	—	—	—	—	—	—	—	—	—	
MC5[2]	—	—	—	—	—	—	—	—	—	—	
MC5[3]	—	—	—	—	—	—	—	—	—	—	
MC5[4]	—	—	—	—	—	—	—	—	—	—	
MC5[5]	—	—	—	—	—	—	—	—	—	—	
MC5[6]	—	—	—	—	—	—	—	—	—	—	
MC5[7]	—	—	—	—	—	—	—	—	—	—	
MC5[8]	—	—	—	—	—	—	—	—	—	—	
MC6[1]	—	—	—	—	—	—	—	—	—	—	
MC6[2]	—	—	—	—	—	—	—	—	—	—	
MC6[3]	—	—	—	—	—	—	—	—	—	—	
MC6[4]	—	—	—	—	—	—	—	—	—	—	
MC6[5]	—	—	—	—	—	—	—	—	—	—	
MC6[2]	—	—	—	—	—	—	—	—	—	—	

寄存器简称	复位	高速	中速	睡眠	模块停止	监视	子激活	子睡眠	软件待机	硬件待机	模块
MC6[6]	—	—	—	—	—	—	—	—	—	—	HCAN*
MC6[7]	—	—	—	—	—	—	—	—	—	—	
MC6[8]	—	—	—	—	—	—	—	—	—	—	
MC7[1]	—	—	—	—	—	—	—	—	—	—	
MC7[2]	—	—	—	—	—	—	—	—	—	—	
MC7[3]	—	—	—	—	—	—	—	—	—	—	
MC7[4]	—	—	—	—	—	—	—	—	—	—	
MC7[5]	—	—	—	—	—	—	—	—	—	—	
MC7[6]	—	—	—	—	—	—	—	—	—	—	
MC7[7]	—	—	—	—	—	—	—	—	—	—	
MC7[8]	—	—	—	—	—	—	—	—	—	—	
MC8[1]	—	—	—	—	—	—	—	—	—	—	
MC8[2]	—	—	—	—	—	—	—	—	—	—	
MC8[3]	—	—	—	—	—	—	—	—	—	—	
MC8[4]	—	—	—	—	—	—	—	—	—	—	
MC8[5]	—	—	—	—	—	—	—	—	—	—	
MC8[6]	—	—	—	—	—	—	—	—	—	—	
MC8[7]	—	—	—	—	—	—	—	—	—	—	
MC8[8]	—	—	—	—	—	—	—	—	—	—	
MC9[1]	—	—	—	—	—	—	—	—	—	—	
MC9[2]	—	—	—	—	—	—	—	—	—	—	
MC9[3]	—	—	—	—	—	—	—	—	—	—	
MC9[4]	—	—	—	—	—	—	—	—	—	—	
MC9[5]	—	—	—	—	—	—	—	—	—	—	
MC9[6]	—	—	—	—	—	—	—	—	—	—	
MC9[7]	—	—	—	—	—	—	—	—	—	—	
MC9[8]	—	—	—	—	—	—	—	—	—	—	
MC10[1]	—	—	—	—	—	—	—	—	—	—	
MC10[2]	—	—	—	—	—	—	—	—	—	—	
MC10[3]	—	—	—	—	—	—	—	—	—	—	
MC10[4]	—	—	—	—	—	—	—	—	—	—	
MC10[5]	—	—	—	—	—	—	—	—	—	—	
MC10[6]	—	—	—	—	—	—	—	—	—	—	
MC10[7]	—	—	—	—	—	—	—	—	—	—	
MC10[8]	—	—	—	—	—	—	—	—	—	—	

寄存器简称	复位	高速	中速	睡眠	模块停止	监视	子激活	子睡眠	软件待机	硬件待机	模块
MC11[1]	—	—	—	—	—	—	—	—	—	—	HCAN*
MC11[2]	—	—	—	—	—	—	—	—	—	—	
MC11[3]	—	—	—	—	—	—	—	—	—	—	
MC11[4]	—	—	—	—	—	—	—	—	—	—	
MC11[5]	—	—	—	—	—	—	—	—	—	—	
MC11[6]	—	—	—	—	—	—	—	—	—	—	
MC11[7]	—	—	—	—	—	—	—	—	—	—	
MC11[8]	—	—	—	—	—	—	—	—	—	—	
MC12[1]	—	—	—	—	—	—	—	—	—	—	
MC12[2]	—	—	—	—	—	—	—	—	—	—	
MC12[3]	—	—	—	—	—	—	—	—	—	—	
MC12[4]	—	—	—	—	—	—	—	—	—	—	
MC12[5]	—	—	—	—	—	—	—	—	—	—	
MC12[6]	—	—	—	—	—	—	—	—	—	—	
MC12[7]	—	—	—	—	—	—	—	—	—	—	
MC12[8]	—	—	—	—	—	—	—	—	—	—	
MC13[1]	—	—	—	—	—	—	—	—	—	—	
MC13[2]	—	—	—	—	—	—	—	—	—	—	
MC13[3]	—	—	—	—	—	—	—	—	—	—	
MC13[4]	—	—	—	—	—	—	—	—	—	—	
MC13[5]	—	—	—	—	—	—	—	—	—	—	
MC13[6]	—	—	—	—	—	—	—	—	—	—	
MC13[7]	—	—	—	—	—	—	—	—	—	—	
MC13[8]	—	—	—	—	—	—	—	—	—	—	
MC14[1]	—	—	—	—	—	—	—	—	—	—	
MC14[2]	—	—	—	—	—	—	—	—	—	—	
MC14[3]	—	—	—	—	—	—	—	—	—	—	
MC14[4]	—	—	—	—	—	—	—	—	—	—	
MC14[5]	—	—	—	—	—	—	—	—	—	—	
MC14[6]	—	—	—	—	—	—	—	—	—	—	
MC14[7]	—	—	—	—	—	—	—	—	—	—	
MC14[8]	—	—	—	—	—	—	—	—	—	—	
MC15[1]	—	—	—	—	—	—	—	—	—	—	
MC15[2]	—	—	—	—	—	—	—	—	—	—	
MC15[3]	—	—	—	—	—	—	—	—	—	—	
MC15[4]	—	—	—	—	—	—	—	—	—	—	

寄存器简称	复位	高速	中速	睡眠	模块停止	监视	子激活	子睡眠	软件待机	硬件待机	模块
MC15[5]	—	—	—	—	—	—	—	—	—	—	HCAN*
MC15[6]	—	—	—	—	—	—	—	—	—	—	
MC15[7]	—	—	—	—	—	—	—	—	—	—	
MC15[8]	—	—	—	—	—	—	—	—	—	—	
MD0[1]	—	—	—	—	—	—	—	—	—	—	
MD0[2]	—	—	—	—	—	—	—	—	—	—	
MD0[3]	—	—	—	—	—	—	—	—	—	—	
MD0[4]	—	—	—	—	—	—	—	—	—	—	
MD0[5]	—	—	—	—	—	—	—	—	—	—	
MD0[6]	—	—	—	—	—	—	—	—	—	—	
MD0[7]	—	—	—	—	—	—	—	—	—	—	
MD0[8]	—	—	—	—	—	—	—	—	—	—	
MD1[1]	—	—	—	—	—	—	—	—	—	—	
MD1[2]	—	—	—	—	—	—	—	—	—	—	
MD1[3]	—	—	—	—	—	—	—	—	—	—	
MD1[4]	—	—	—	—	—	—	—	—	—	—	
MD1[5]	—	—	—	—	—	—	—	—	—	—	
MD1[6]	—	—	—	—	—	—	—	—	—	—	
MD1[7]	—	—	—	—	—	—	—	—	—	—	
MD1[8]	—	—	—	—	—	—	—	—	—	—	
MD2[1]	—	—	—	—	—	—	—	—	—	—	
MD2[2]	—	—	—	—	—	—	—	—	—	—	
MD2[3]	—	—	—	—	—	—	—	—	—	—	
MD2[4]	—	—	—	—	—	—	—	—	—	—	
MD2[5]	—	—	—	—	—	—	—	—	—	—	
MD2[6]	—	—	—	—	—	—	—	—	—	—	
MD2[7]	—	—	—	—	—	—	—	—	—	—	
MD2[8]	—	—	—	—	—	—	—	—	—	—	
MD3[1]	—	—	—	—	—	—	—	—	—	—	
MD3[2]	—	—	—	—	—	—	—	—	—	—	
MD3[3]	—	—	—	—	—	—	—	—	—	—	
MD3[4]	—	—	—	—	—	—	—	—	—	—	
MD3[5]	—	—	—	—	—	—	—	—	—	—	
MD3[6]	—	—	—	—	—	—	—	—	—	—	
MD3[7]	—	—	—	—	—	—	—	—	—	—	
MD3[8]	—	—	—	—	—	—	—	—	—	—	
MD4[1]	—	—	—	—	—	—	—	—	—	—	

寄存器简称	复位	高速	中速	睡眠	模块停止	监视	子激活	子睡眠	软件待机	硬件待机	模块
MD4[2]	—	—	—	—	—	—	—	—	—	—	HCAN*
MD4[3]	—	—	—	—	—	—	—	—	—	—	
MD4[4]	—	—	—	—	—	—	—	—	—	—	
MD4[5]	—	—	—	—	—	—	—	—	—	—	
MD4[6]	—	—	—	—	—	—	—	—	—	—	
MD4[7]	—	—	—	—	—	—	—	—	—	—	
MD4[8]	—	—	—	—	—	—	—	—	—	—	
MD5[1]	—	—	—	—	—	—	—	—	—	—	
MD5[2]	—	—	—	—	—	—	—	—	—	—	
MD5[3]	—	—	—	—	—	—	—	—	—	—	
MD5[4]	—	—	—	—	—	—	—	—	—	—	
MD5[5]	—	—	—	—	—	—	—	—	—	—	
MD5[6]	—	—	—	—	—	—	—	—	—	—	
MD5[7]	—	—	—	—	—	—	—	—	—	—	
MD5[8]	—	—	—	—	—	—	—	—	—	—	
MD6[1]	—	—	—	—	—	—	—	—	—	—	
MD6[2]	—	—	—	—	—	—	—	—	—	—	
MD6[3]	—	—	—	—	—	—	—	—	—	—	
MD6[4]	—	—	—	—	—	—	—	—	—	—	
MD6[5]	—	—	—	—	—	—	—	—	—	—	
MD6[6]	—	—	—	—	—	—	—	—	—	—	
MD6[7]	—	—	—	—	—	—	—	—	—	—	
MD6[8]	—	—	—	—	—	—	—	—	—	—	
MD7[1]	—	—	—	—	—	—	—	—	—	—	
MD7[2]	—	—	—	—	—	—	—	—	—	—	
MD7[3]	—	—	—	—	—	—	—	—	—	—	
MD7[4]	—	—	—	—	—	—	—	—	—	—	
MD7[5]	—	—	—	—	—	—	—	—	—	—	
MD7[6]	—	—	—	—	—	—	—	—	—	—	
MD7[7]	—	—	—	—	—	—	—	—	—	—	
MD7[8]	—	—	—	—	—	—	—	—	—	—	
MD8[1]	—	—	—	—	—	—	—	—	—	—	
MD8[2]	—	—	—	—	—	—	—	—	—	—	
MD8[3]	—	—	—	—	—	—	—	—	—	—	
MD8[4]	—	—	—	—	—	—	—	—	—	—	
MD8[5]	—	—	—	—	—	—	—	—	—	—	

寄存器简称	复位	高速	中速	睡眠	模块停止	监视	子激活	子睡眠	软件待机	硬件待机	模块
MD8[6]	—	—	—	—	—	—	—	—	—	—	HCAN*
MD8[7]	—	—	—	—	—	—	—	—	—	—	
MD8[8]	—	—	—	—	—	—	—	—	—	—	
MD9[1]	—	—	—	—	—	—	—	—	—	—	
MD9[2]	—	—	—	—	—	—	—	—	—	—	
MD9[3]	—	—	—	—	—	—	—	—	—	—	
MD9[4]	—	—	—	—	—	—	—	—	—	—	
MD9[5]	—	—	—	—	—	—	—	—	—	—	
MD9[6]	—	—	—	—	—	—	—	—	—	—	
MD9[7]	—	—	—	—	—	—	—	—	—	—	
MD9[8]	—	—	—	—	—	—	—	—	—	—	
MD10[1]	—	—	—	—	—	—	—	—	—	—	
MD10[2]	—	—	—	—	—	—	—	—	—	—	
MD10[3]	—	—	—	—	—	—	—	—	—	—	
MD10[4]	—	—	—	—	—	—	—	—	—	—	
MD10[5]	—	—	—	—	—	—	—	—	—	—	
MD10[6]	—	—	—	—	—	—	—	—	—	—	
MD10[7]	—	—	—	—	—	—	—	—	—	—	
MD10[8]	—	—	—	—	—	—	—	—	—	—	
MD11[1]	—	—	—	—	—	—	—	—	—	—	
MD11[2]	—	—	—	—	—	—	—	—	—	—	
MD11[3]	—	—	—	—	—	—	—	—	—	—	
MD11[4]	—	—	—	—	—	—	—	—	—	—	
MD11[5]	—	—	—	—	—	—	—	—	—	—	
MD11[6]	—	—	—	—	—	—	—	—	—	—	
MD11[7]	—	—	—	—	—	—	—	—	—	—	
MD11[8]	—	—	—	—	—	—	—	—	—	—	
MD12[1]	—	—	—	—	—	—	—	—	—	—	
MD12[2]	—	—	—	—	—	—	—	—	—	—	
MD12[3]	—	—	—	—	—	—	—	—	—	—	
MD12[4]	—	—	—	—	—	—	—	—	—	—	
MD12[5]	—	—	—	—	—	—	—	—	—	—	
MD12[6]	—	—	—	—	—	—	—	—	—	—	
MD12[7]	—	—	—	—	—	—	—	—	—	—	
MD12[8]	—	—	—	—	—	—	—	—	—	—	
MD13[1]	—	—	—	—	—	—	—	—	—	—	
MD13[2]	—	—	—	—	—	—	—	—	—	—	

寄存器简称	复位	高速	中速	睡眠	模块停止	监视	子激活	子睡眠	软件待机	硬件待机	模块
MD13[3]	—	—	—	—	—	—	—	—	—	—	HCAN*
MD13[4]	—	—	—	—	—	—	—	—	—	—	
MD13[5]	—	—	—	—	—	—	—	—	—	—	
MD13[6]	—	—	—	—	—	—	—	—	—	—	
MD13[7]	—	—	—	—	—	—	—	—	—	—	
MD13[8]	—	—	—	—	—	—	—	—	—	—	
MD14[1]	—	—	—	—	—	—	—	—	—	—	
MD14[2]	—	—	—	—	—	—	—	—	—	—	
MD14[3]	—	—	—	—	—	—	—	—	—	—	
MD14[4]	—	—	—	—	—	—	—	—	—	—	
MD14[5]	—	—	—	—	—	—	—	—	—	—	
MD14[6]	—	—	—	—	—	—	—	—	—	—	
MD14[7]	—	—	—	—	—	—	—	—	—	—	
MD14[8]	—	—	—	—	—	—	—	—	—	—	
MD15[1]	—	—	—	—	—	—	—	—	—	—	
MD15[2]	—	—	—	—	—	—	—	—	—	—	
MD15[3]	—	—	—	—	—	—	—	—	—	—	
MD15[4]	—	—	—	—	—	—	—	—	—	—	
MD15[5]	—	—	—	—	—	—	—	—	—	—	
MD15[6]	—	—	—	—	—	—	—	—	—	—	
MD15[7]	—	—	—	—	—	—	—	—	—	—	
MD15[8]	—	—	—	—	—	—	—	—	—	—	
PWCR_1	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	PWM_1
PWOCR_1	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
PWPR_1	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
PWCYR_1	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
PWBFR_1A	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
PWBFR_1C	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
PWBFR_1E	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
PWBFR_1G	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
PWCR_2	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	PWM_2
PWOCR_2	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
PWPR_2	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
PWCYR_2	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
PWBFR_2A	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
PWBFR_2B	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
PWBFR_2C	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
PWBFR_2D	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	

寄存器简称	复位	高速	中速	睡眠	模块停止	监视	子激活	子睡眠	软件待机	硬件待机	模块
PHDDR	初始化	—	—	—	—	—	—	—	—	初始化	PORT
PJDDR	初始化	—	—	—	—	—	—	—	—	初始化	
PHDR	初始化	—	—	—	—	—	—	—	—	初始化	
PJDR	初始化	—	—	—	—	—	—	—	—	初始化	
PORTH	初始化	—	—	—	—	—	—	—	—	初始化	
PORTJ	初始化	—	—	—	—	—	—	—	—	初始化	
TRPRT	初始化	—	—	—	—	—	—	—	—	初始化	
LPCR	初始化	—	—	—	—	—	—	—	—	初始化	LCD
LCR	初始化	—	—	—	—	—	—	—	—	初始化	
LCR2	初始化	—	—	—	—	—	—	—	—	初始化	
MSTPCRD	初始化	—	—	—	—	—	—	—	—	初始化	SYSTEM
SBYCR	初始化	—	—	—	—	—	—	—	—	初始化	
SYSCR	初始化	—	—	—	—	—	—	—	—	初始化	
SCKCR	初始化	—	—	—	—	—	—	—	—	初始化	
MDCR	初始化	—	—	—	—	—	—	—	—	初始化	
MSTPCRA	初始化	—	—	—	—	—	—	—	—	初始化	
MSTPCRB	初始化	—	—	—	—	—	—	—	—	初始化	
MSTPCRC	初始化	—	—	—	—	—	—	—	—	初始化	
LPWRCR	初始化	—	—	—	—	—	—	—	—	初始化	INT
ISCRH	初始化	—	—	—	—	—	—	—	—	初始化	
ISCR L	初始化	—	—	—	—	—	—	—	—	初始化	
IER	初始化	—	—	—	—	—	—	—	—	初始化	
ISR	初始化	—	—	—	—	—	—	—	—	初始化	
P1DDR	初始化	—	—	—	—	—	—	—	—	初始化	PORT
P3DDR	初始化	—	—	—	—	—	—	—	—	初始化	
PADDR	初始化	—	—	—	—	—	—	—	—	初始化	
PBDDR	初始化	—	—	—	—	—	—	—	—	初始化	
PCDDR	初始化	—	—	—	—	—	—	—	—	初始化	
PDDDR	初始化	—	—	—	—	—	—	—	—	初始化	
PFDDR	初始化	—	—	—	—	—	—	—	—	初始化	
P3ODR	初始化	—	—	—	—	—	—	—	—	初始化	TPU
PAODR	初始化	—	—	—	—	—	—	—	—	初始化	
PBODR	初始化	—	—	—	—	—	—	—	—	初始化	
PCODR	初始化	—	—	—	—	—	—	—	—	初始化	
TSTR	初始化	—	—	—	—	—	—	—	—	初始化	
TSYR	初始化	—	—	—	—	—	—	—	—	初始化	
IPRA	初始化	—	—	—	—	—	—	—	—	初始化	INT

寄存器简称	复位	高速	中速	睡眠	模块停止	监视	子激活	子睡眠	软件待机	硬件待机	模块
IPRB	初始化	—	—	—	—	—	—	—	—	初始化	INT
IPRC	初始化	—	—	—	—	—	—	—	—	初始化	
IPRD	初始化	—	—	—	—	—	—	—	—	初始化	
IPRE	初始化	—	—	—	—	—	—	—	—	初始化	
IPRF	初始化	—	—	—	—	—	—	—	—	初始化	
IPRG	初始化	—	—	—	—	—	—	—	—	初始化	
IPRJ	初始化	—	—	—	—	—	—	—	—	初始化	
IPRK	初始化	—	—	—	—	—	—	—	—	初始化	
IPRM	初始化	—	—	—	—	—	—	—	—	初始化	
RAMER*	初始化	—	—	—	—	—	—	—	—	初始化	FLASH (F-ZTAT 版)
P1DR	初始化	—	—	—	—	—	—	—	—	初始化	PORT
P3DR	初始化	—	—	—	—	—	—	—	—	初始化	
PADR	初始化	—	—	—	—	—	—	—	—	初始化	
PBDR	初始化	—	—	—	—	—	—	—	—	初始化	
PCDR	初始化	—	—	—	—	—	—	—	—	初始化	
PDDR	初始化	—	—	—	—	—	—	—	—	初始化	
PFDR	初始化	—	—	—	—	—	—	—	—	初始化	
TCR_0	初始化	—	—	—	—	—	—	—	—	初始化	TPU_0
TMDR_0	初始化	—	—	—	—	—	—	—	—	初始化	
TIORH_0	初始化	—	—	—	—	—	—	—	—	初始化	
TIORL_0	初始化	—	—	—	—	—	—	—	—	初始化	
TIER_0	初始化	—	—	—	—	—	—	—	—	初始化	
TSR_0	初始化	—	—	—	—	—	—	—	—	初始化	
TCNTH_0	初始化	—	—	—	—	—	—	—	—	初始化	
TCNTL_0	初始化	—	—	—	—	—	—	—	—	初始化	
TGRAH_0	初始化	—	—	—	—	—	—	—	—	初始化	
TGRAL_0	初始化	—	—	—	—	—	—	—	—	初始化	
TGRBH_0	初始化	—	—	—	—	—	—	—	—	初始化	
TGRBL_0	初始化	—	—	—	—	—	—	—	—	初始化	
TGRCH_0	初始化	—	—	—	—	—	—	—	—	初始化	
TGRCL_0	初始化	—	—	—	—	—	—	—	—	初始化	
TGRDH_0	初始化	—	—	—	—	—	—	—	—	初始化	
TGRDL_0	初始化	—	—	—	—	—	—	—	—	初始化	
TCR_1	初始化	—	—	—	—	—	—	—	—	初始化	TPU_1
TMDR_1	初始化	—	—	—	—	—	—	—	—	初始化	

寄存器简称	复位	高速	中速	睡眠	模块停止	监视	子激活	子睡眠	软件待机	硬件待机	模块
TIOR_1	初始化	—	—	—	—	—	—	—	—	初始化	TPU_1
TIER_1	初始化	—	—	—	—	—	—	—	—	初始化	
TSR_1	初始化	—	—	—	—	—	—	—	—	初始化	
TCNTH_1	初始化	—	—	—	—	—	—	—	—	初始化	
TCNTL_1	初始化	—	—	—	—	—	—	—	—	初始化	
TGRAH_1	初始化	—	—	—	—	—	—	—	—	初始化	
TGRAL_1	初始化	—	—	—	—	—	—	—	—	初始化	
TGRBH_1	初始化	—	—	—	—	—	—	—	—	初始化	
TGRBL_1	初始化	—	—	—	—	—	—	—	—	初始化	
TCR_2	初始化	—	—	—	—	—	—	—	—	初始化	TPU_2
TMDR_2	初始化	—	—	—	—	—	—	—	—	初始化	
TIOR_2	初始化	—	—	—	—	—	—	—	—	初始化	
TIER_2	初始化	—	—	—	—	—	—	—	—	初始化	
TSR_2	初始化	—	—	—	—	—	—	—	—	初始化	
TCNTH_2	初始化	—	—	—	—	—	—	—	—	初始化	
TCNTL_2	初始化	—	—	—	—	—	—	—	—	初始化	
TGRAH_2	初始化	—	—	—	—	—	—	—	—	初始化	
TGRAL_2	初始化	—	—	—	—	—	—	—	—	初始化	
TGRBH_2	初始化	—	—	—	—	—	—	—	—	初始化	WDT_0
TGRBL_2	初始化	—	—	—	—	—	—	—	—	初始化	
TCSR_0	初始化	—	—	—	—	—	—	—	—	初始化	
TCNT_0	初始化	—	—	—	—	—	—	—	—	初始化	SCI_0
RSTCSR	初始化	—	—	—	—	—	—	—	—	初始化	
SMR_0	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
BRR_0	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
SCR_0	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
TDR_0	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
SSR_0	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
RDR_0	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
SCMR_0	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
SMR_1	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	SCI_1
BRR_1	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
SCR_1	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
TDR_1	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
SSR_1	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	

寄存器简称	复位	高速	中速	睡眠	模块停止	监视	子激活	子睡眠	软件待机	硬件待机	模块
RDR_1	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	SCI_1
SCMR_1	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
ADDRAH	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	A/D
ADDRAL	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
ADDRBH	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
ADDRBL	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
ADDRCH	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
ADDRCL	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
ADDRDH	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
ADDRDL	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
ADCSR	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
ADCR	初始化	—	—	—	初始化	初始化	初始化	初始化	初始化	初始化	
TCSR_1	初始化	—	—	—	—	—	—	—	—	初始化	WDT_1
TCNT_1	初始化	—	—	—	—	—	—	—	—	初始化	
FLMCR1	初始化	—	—	—	—	—	—	—	—	初始化	FLASH (F-ZTAT 版)
FLMCR2	初始化	—	—	—	—	—	—	—	—	初始化	
EBR1	初始化	—	—	—	—	—	—	—	—	初始化	
EBR2*	初始化	—	—	—	—	—	—	—	—	初始化	
FLPWCR	初始化	—	—	—	—	—	—	—	—	初始化	
PORT1	—	—	—	—	—	—	—	—	—	—	PORT
PORT3	—	—	—	—	—	—	—	—	—	—	
PORT4	—	—	—	—	—	—	—	—	—	—	
PORTA	—	—	—	—	—	—	—	—	—	—	
PORTB	—	—	—	—	—	—	—	—	—	—	
PORTC	—	—	—	—	—	—	—	—	—	—	
PORTD	—	—	—	—	—	—	—	—	—	—	
PORTF	—	—	—	—	—	—	—	—	—	—	

【注】 — 表示不被初始化。

* 在 H8S/2280 群中，本寄存器被保留。

第 22 章 电特性

22.1 绝对最大额定值

绝对最大额定值如表 22.1 所示。

表 22.1 绝对最大额定值

项目	符号	额定值	单位
电源电压	V _{CC} 、LPV _{CC}	-0.3 ~ +7.0	V
	PWMV _{CC}	-0.3 ~ +7.0	V
输入电压 (XTAL、EXTAL)	V _{in}	-0.3 ~ V _{CC} +0.3	V
输入电压 (端口 4)	V _{in}	-0.3 ~ AV _{CC} +0.3	V
输入电压 (XTAL、EXTAL、端口 4 以外)	V _{in}	-0.3 ~ V _{CC} +0.3	V
输入电压 (端口 H、J)	V _{in}	-0.3 ~ PWMV _{CC} +0.3	V
模拟电源电压	AV _{CC}	-0.3 ~ +7.0	V
模拟输入电压	V _{AN}	-0.3 ~ AV _{CC} +0.3	V
运行温度	T _{opr}	通常规格产品: -20 ~ +75	°C
		宽温度范围规格产品: -40 ~ +85	
保存温度	T _{stg}	-55 ~ +125	°C

【使用时的注意事项】

在超过绝对额定值使用 LSI 的情况下, 会对 LSI 造成永久性破坏。

22.2 DC 特性

DC 特性如表 22.2 所示。输出容许电流如表 22.3 所示。

表 22.2 DC 特性

条件: $V_{CC}=LPV_{CC}=4.5 \sim 5.5V$ 、 $AV_{CC}=4.5 \sim 5.5V$ 、 $PWMV_{CC}=4.5 \sim 5.5V$ 、 $V_{SS}=AV_{SS}=PWMV_{SS}=PLL_{VSS}=0V$ 、 $T_a = -20 \sim +75^{\circ}C$ (通常规格产品)、 $T_a = -40 \sim +85^{\circ}C$ (宽温度范围规格产品) *1

项目	符号	min.	typ.	max.	单位	测定条件
施密特 触发器输入 电压	$\overline{IRQ0} \sim \overline{IRQ5}$ 、 端口 1、3、A ~ D、F、H、 J	V_{T-}	$V_{CC} \times 0.2$	—	—	V
		V_{T+}		—	$V_{CC} \times 0.7$	V
		$V_{T+} - V_{T-}$	$V_{CC} \times 0.05$	—	—	V
输入高电平 电压	$\overline{RES}$ 、 $\overline{STBY}$ 、 $\overline{NMI}$ 、 MD2、MD0、FWE	V_{IH}	$V_{CC} \times 0.9$	—	$V_{CC} + 0.3$	V
	EXTAL		$V_{CC} \times 0.7$	—	$V_{CC} + 0.3$	V
	SCK0、SCK1、RxD0、 RxD1、HRxD*4		$V_{CC} \times 0.7$	—	$V_{CC} + 0.3$	V
	端口 4		$AV_{CC} \times 0.7$	—	$AV_{CC} + 0.3$	V
输入低电平 电压	$\overline{RES}$ 、 $\overline{STBY}$ 、 $\overline{NMI}$ 、 MD2、MD0、FWE	V_{IL}	-0.3	—	$V_{CC} \times 0.1$	V
	EXTAL		-0.3	—	$V_{CC} \times 0.2$	V
	SCK0、SCK1、RxD0、 RxD1、HRxD*4		-0.3	—	$V_{CC} \times 0.2$	V
	端口 4		-0.3	—	$AV_{CC} \times 0.2$	V
输出高 电平电压	全部输出管脚	V_{OH}	$V_{CC} - 0.5$	—	—	V $I_{OH} = -200\mu A$
			$V_{CC} - 1.0$	—	—	V $I_{OH} = -1mA$
输出低电平 电压	全部输出管脚	V_{OL}	—	—	0.4	V $I_{OL} = 1.6mA$
输入漏泄 电流	$\overline{RES}$	$ I_{in} $	—	—	1.0	μA
	$\overline{STBY}$ 、 $\overline{NMI}$ 、 MD2、MD0、FWE、 HRxD*4		—	—	1.0	μA
	端口 4		—	—	1.0	μA $V_{in}=0.5 \sim AV_{CC}-0.5V$
	上述以外的端口		—	—	1.0	μA $V_{in}=0.5 \sim V_{CC}-0.5V$
输入容量	$\overline{RES}$	C_{in}	—	—	30	pF $V_{in}=0$
	$\overline{NMI}$		—	—	30	pF $f=1MHz$
	$\overline{RES}$ 、 $\overline{NMI}$ 以外的全部输入 管脚		—	—	15	pF $T_a=25^{\circ}C$

项目		符号	min.	typ.	max.	单位	测定条件
消耗电流 *2	通常运行时	I _{CC} *3	—	45 (V _{CC} =5.0V)	55 (V _{CC} =5.5V)	mA	f=20MHz
	睡眠时		—	35 (V _{CC} =5.0V)	45 (V _{CC} =5.5V)	mA	f=20MHz
	所有模块停止时		—	30	—	mA	f=20MHz、 V _{CC} =5.0V (参考值)
	中速模式 (φ/32) 时		—	30	—	mA	f=20MHz、 V _{CC} =5.0V (参考值)
	子激活模式时		—	0.7	1.0	mA	子时钟时
	子睡眠模式时		—	0.7	1.0	mA	子时钟时
	监视模式时		—	0.6	1.0	mA	子时钟时
	待机时		—	2	5.0	μA	Ta ≤ 50°C
			—	—	20		50°C<Ta
LCD 电源部	运行中	LPI _{CC}	—	10	20	mA	
端口 电源电流	待机时		—	0.1	10	μA	Ta ≤ 50°C
			—	—	80		50°C<Ta
模拟 电源电流	A/D 转换中	AI _{CC}	—	2.5	4.0	mA	AV _{CC} =5.0V
	A/D 转换待机时		—	—	5.0	μA	
RAM 待机电压		V _{RAM}	2.0	—	—	V	

【注】 *1 即使不使用 A/D 转换器，也不要将 AV_{CC}、AV_{SS} 管脚置为开路状态。另外，通过将 AV_{CC} 管脚连接到 V_{CC} 等的方法，外加 4.5 ~ 5.5V 的电压。

*2 在 V_{IH}=V_{CC} (EXTAL)、AV_{CC} (端口 4)、PWMV_{CC}、LPV_{CC}、V_{CC} (其他)、V_{IL}=0V 的条件下，消耗电流值为所有输出管脚置为无负载状态时的值。

*3 I_{CC} 与 V_{CC} 和 f 的关系如下述式子所示。
 $I_{CC}(\text{max.}) = 22 + 0.3 \times V_{CC} \times f$ (通常运行时)
 $I_{CC}(\text{max.}) = 18 + 0.25 \times V_{CC} \times f$ (睡眠时)

*4 H8S/2280 群不具有本功能。

表 22.3 输出容许电流

条件: $V_{CC}=LPV_{CC}=4.5 \sim 5.5V$ 、 $AV_{CC}=4.5 \sim 5.5V$ 、 $PWMV_{CC}=4.5 \sim 5.5V$ 、 $V_{SS}=AV_{SS}=PWMV_{SS}=PLL_{VSS}=0V$ 、 $T_a = -20 \sim +75^{\circ}C$ (通常规格产品)、 $T_a = -40 \sim +85^{\circ}C$ (宽温度范围规格产品)

项目		符号	min.	typ	max	单位	测定条件
输出低电平容许电流 (每 1 个管脚)	除 PWM1A ~ 1H PWM2A ~ 2H 以外的全部输出管脚	I_{OL}	—	—	10	mA	
	PWM1A ~ 1H PWM2A ~ 2H				25	mA	$T_a=75 \sim 85^{\circ}C$
			—	—	30	mA	$T_a=25^{\circ}C$
			—	—	40	mA	$T_a=-40^{\circ}C$
输出低电平容许电流 (总和)	除 PWM1A ~ 1H PWM2A ~ 2H 以外的全部输出管脚的总和	I_{OL}	—	—	80	mA	
	PWM1A ~ 1H PWM2A ~ 2H 的总和		—	—	150	mA	$T_a=75 \sim 85^{\circ}C$
			—	—	180	mA	$T_a=25^{\circ}C$
			—	—	220	mA	$T_a=-40^{\circ}C$
输出高电平容许电流 (每 1 个端口)	除 PWM1A ~ 1H PWM2A ~ 2H 以外的全部输出管脚	$-I_{OH}$	—	—	2.0	mA	
	PWM1A ~ 1H PWM2A ~ 2H		—	—	25	mA	$T_a=75 \sim 85^{\circ}C$
			—	—	30	mA	$T_a=25^{\circ}C$
			—	—	40	mA	$T_a=-40^{\circ}C$
输出高电平容许电流 (总和)	除 PWM1A ~ 1H PWM2A ~ 2H 以外的全部输出管脚的总和	$-\Sigma I_{OH}$	—	—	40	mA	
	PWM1A ~ 1H PWM2A ~ 2H 的总和		—	—	150	mA	$T_a=75 \sim 85^{\circ}C$
			—	—	180	mA	$T_a=25^{\circ}C$
			—	—	220	mA	$T_a=-40^{\circ}C$

【注】 为了确保 LSI 的可靠性, 输出电流值不能超过表 22.3 所示的值。

22.3 AC 特性

AC 测定条件如图 22.1 所示。

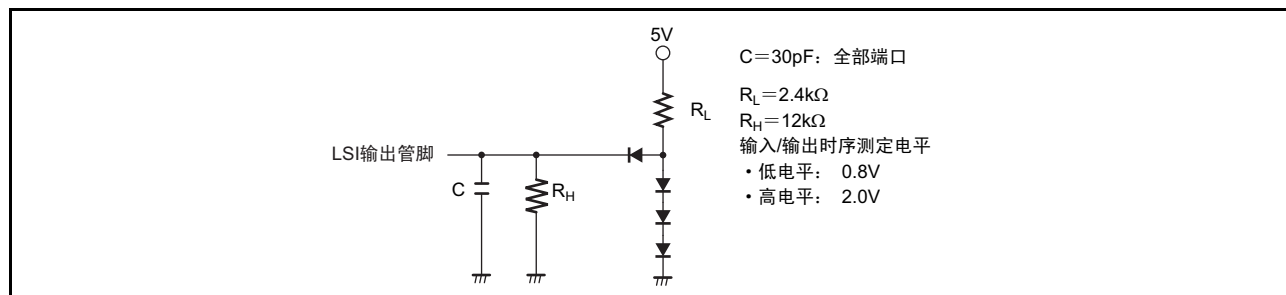


图 22.1 输出负载电路

22.3.1 时钟时序

时钟时序如表 22.4 所示。

表 22.4 时钟时序

条件: $V_{CC}=LPV_{CC}=4.5 \sim 5.5\text{V}$ 、 $AV_{CC}=4.5 \sim 5.5\text{V}$ 、 $PWMV_{CC}=4.5 \sim 5.5\text{V}$ 、 $V_{SS}=AV_{SS}=PWMV_{SS}=PLL_{VSS}=0\text{V}$ 、 $\phi=4 \sim 20\text{MHz}$ 、 $T_a=-20 \sim +75^\circ\text{C}$ (通常规格产品)、 $T_a=-40 \sim +85^\circ\text{C}$ (宽温度范围规格产品)

项目	符号	min	max.	单位	测定条件
时钟周期时间	t_{cyc}	50	250	ns	图 22.2
时钟高电平脉冲宽度	t_{CH}	15	—	ns	
时钟低电平脉冲宽度	t_{CL}	15	—	ns	
时钟上升时间	t_{Cr}	—	5	ns	
时钟下降时间	t_{Cf}	—	5	ns	
复位振荡稳定时间 (晶体)	t_{OSC1}	20	—	ms	图 22.3
软件待机振荡稳定时间 (晶体)	t_{OSC2}	8	—	ms	图 20.3
外部时钟输出稳定延迟时间	t_{DEXT}	2	—	ms	图 22.3

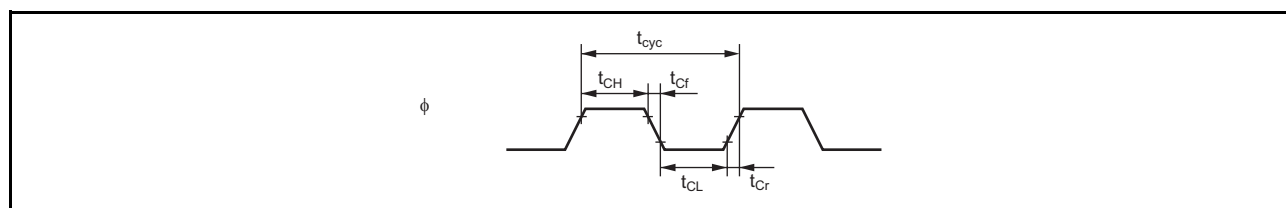


图 22.2 系统时钟时序

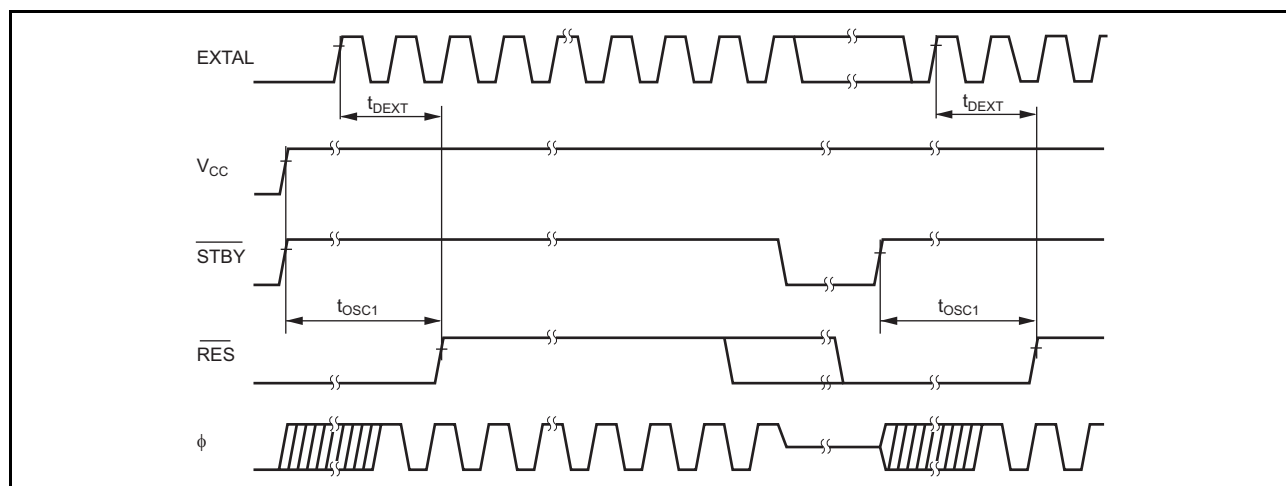


图 22.3 振荡稳定时间时序

22.3.2 控制信号时序

控制信号时序如表 22.5 所示。

表 22.5 控制信号时序

条件: $V_{cc}=LPV_{cc}=4.5 \sim 5.5V$ 、 $AV_{cc}=4.5 \sim 5.5V$ 、 $PWMV_{cc}=4.5 \sim 5.5V$ 、 $V_{ss}=AV_{ss}=PWMV_{ss}=PLL_{ss}=0V$ 、 $\phi=4 \sim 20MHz$ 、 $T_a=-20 \sim +75^{\circ}C$ (通常规格产品)、 $T_a=-40 \sim +85^{\circ}C$ (宽温度范围规格产品)

项目	符号	min.	max.	单位	测定条件
RES 设置时间	t_{RESS}	200	—	ns	图 22.4
RES 脉冲宽度	t_{RESW}	20	—	t_{cyc}	
NMI 设置时间	t_{NMIS}	150	—	ns	图 22.5
NMI 保持时间	t_{NMIH}	10	—	ns	
NMI 脉冲宽度 (从软件待机模式恢复时)	t_{NMIW}	200	—	ns	
IRQ 设置时间	t_{IRQS}	150	—	ns	
IRQ 保持时间	t_{IRQH}	10	—	ns	
IRQ 脉冲宽度 (从软件待机模式恢复时)	t_{IRQW}	200	—	ns	

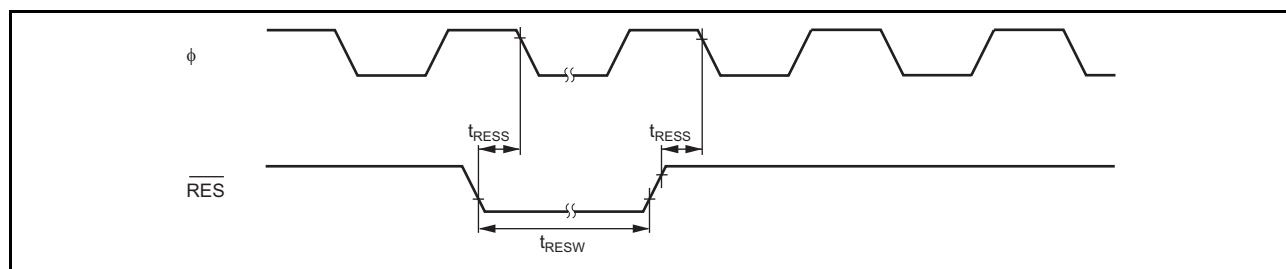


图 22.4 复位输入时序

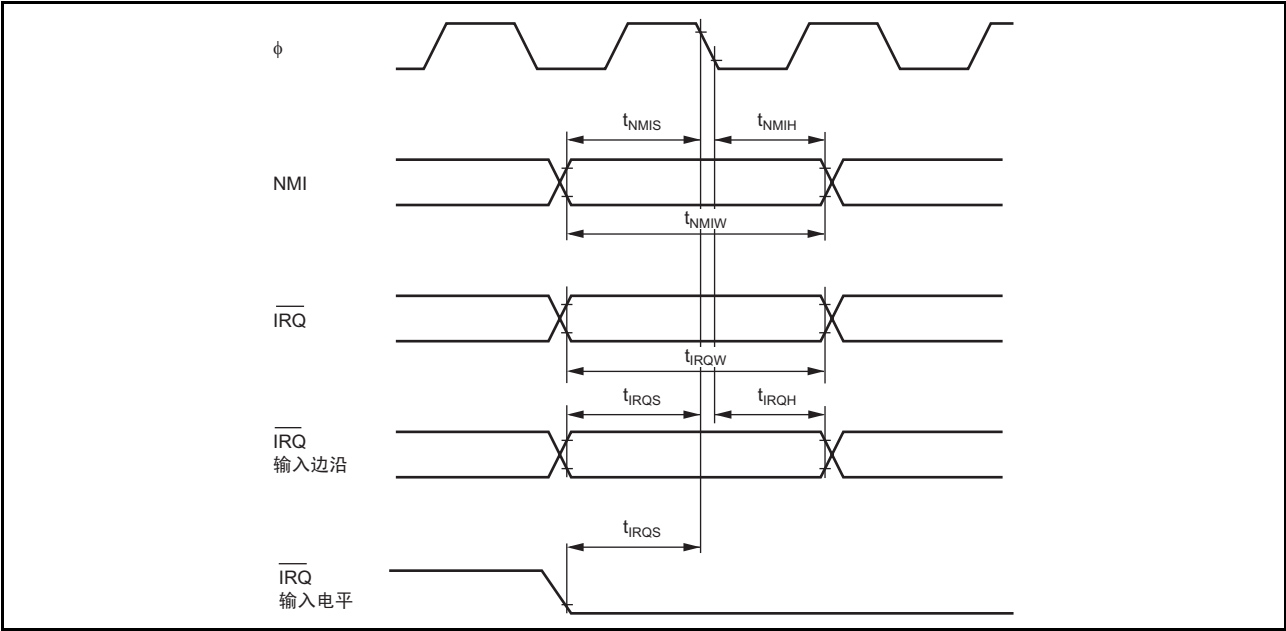


图 22.5 中断输入时序

22.3.3 内部外围模块时序

内部外围模块时序如表 22.6 所示。

表 22.6 内部外围模块时序

条件: $V_{CC}=LPV_{CC}=4.5 \sim 5.5V$ 、 $AV_{CC}=4.5 \sim 5.5V$ 、 $PWMV_{CC}=4.5 \sim 5.5V$ 、 $V_{SS}=AV_{SS}=PWMV_{SS}=PLL_{VSS}=0V$ 、 $\phi=4 \sim 20MHz$ 、 $T_a=-20 \sim +75^{\circ}C$ (通常规格产品)、 $T_a=-40 \sim +85^{\circ}C$ (宽温度范围规格产品)

项目		符号	min.	max.	单位	测定条件
I/O 端口	输出数据延迟时间	t_{PWD}	—	50	ns	图 22.6
	输入数据设置时间	t_{PRS}	30	—	ns	
	输入数据保持时间	t_{PRH}	30	—	ns	
TPU	定时器输出延迟时间	t_{OCD}	—	50	ns	图 22.7
	定时器输入设置时间	t_{TICS}	30	—	ns	图 22.8
	定时器时钟输入设置时间	t_{TCKS}	30	—	ns	
	定时器时钟 脉冲宽度	单沿指定 t_{TCKWH}	1.5	—	t_{cyc}	
		两边沿指定 t_{TCKWL}	2.5	—	t_{cyc}	
SCI	输入时钟 周期	异步	4	—	t_{cyc}	图 22.9
			6	—	t_{cyc}	
	输入时钟脉冲宽度	时钟同步	0.4	0.6	t_{Scyc}	
			—	1.5	t_{cyc}	图 22.10
	输入时钟上升时间	t_{SCKr}	—	1.5	t_{cyc}	
	输入时钟下降时间	t_{SCKf}	—	1.5	t_{cyc}	
	发送数据延迟时间	t_{TXD}	—	50	ns	
	接收数据设置时间 (时钟同步)	t_{RXS}	50	—	ns	
	接收数据保持时间 (时钟同步)	t_{RXH}	50	—	ns	
A/D 转换器	触发输入设置时间	t_{TRGS}	30	—	ns	图 22.11
HCAN*	发送数据延迟时间	t_{HTXD}	—	100	ns	图 22.12
	接收数据设置时间	t_{HRXS}	100	—	ns	
	接收数据保持时间	t_{HRXH}	100	—	ns	
PWM	脉冲输出延迟时间	t_{MPWMOD}	—	50	ns	图 22.13

【注】*HCAN 的输入信号为异步信号，它在图 22.12 所示的系统时钟 (ϕ) 的上升沿 (2 时钟周期间隔) 发生变化。
HCAN 输出信号是异步信号，它的变化以图 22.12 所示的系统时钟 (ϕ) 的上升沿 (2 时钟周期间隔) 为基准。
H8S/2280 群不具有本功能。

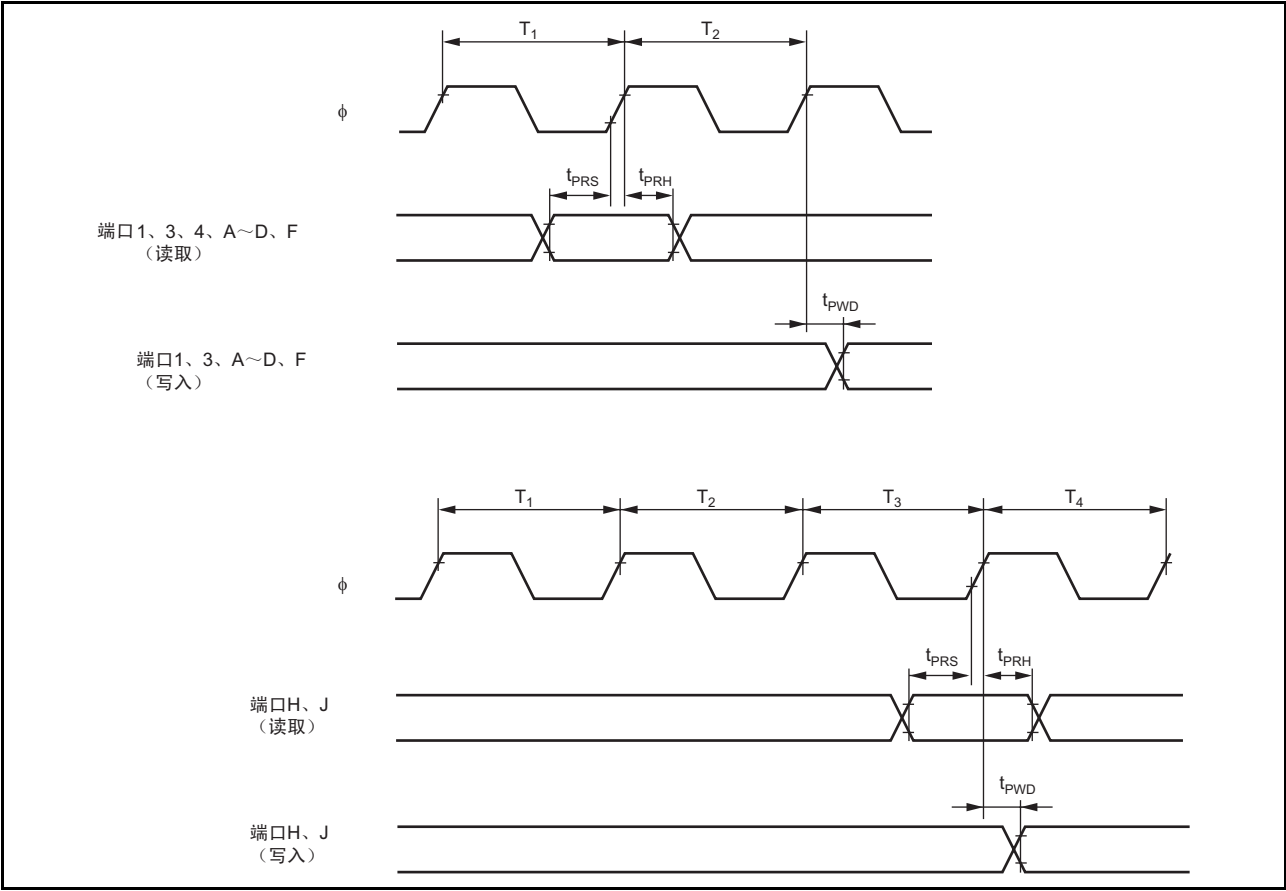


图 22.6 I/O 端口输入 / 输出时序

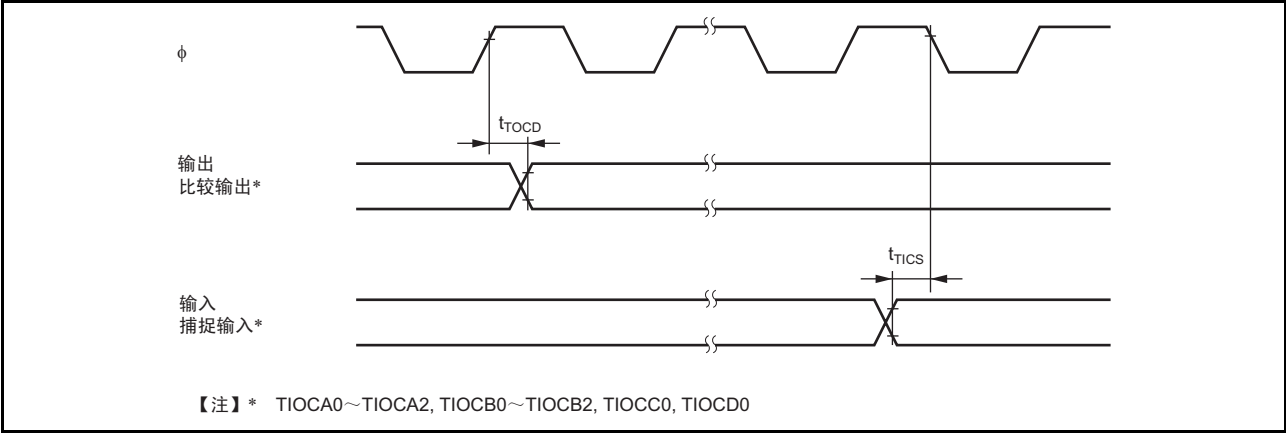


图 22.7 TPU 输入 / 输出时序

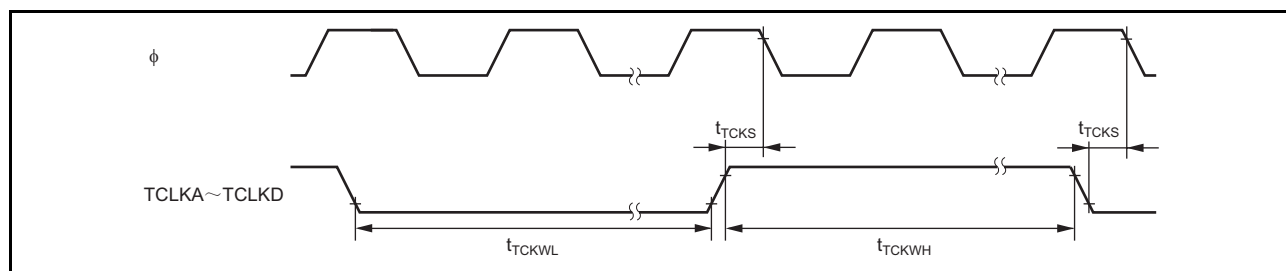


图 22.8 TPU 时钟输入时序

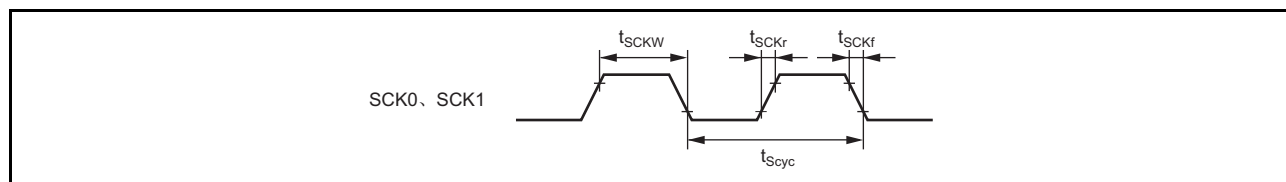


图 22.9 SCK 时钟输入时序

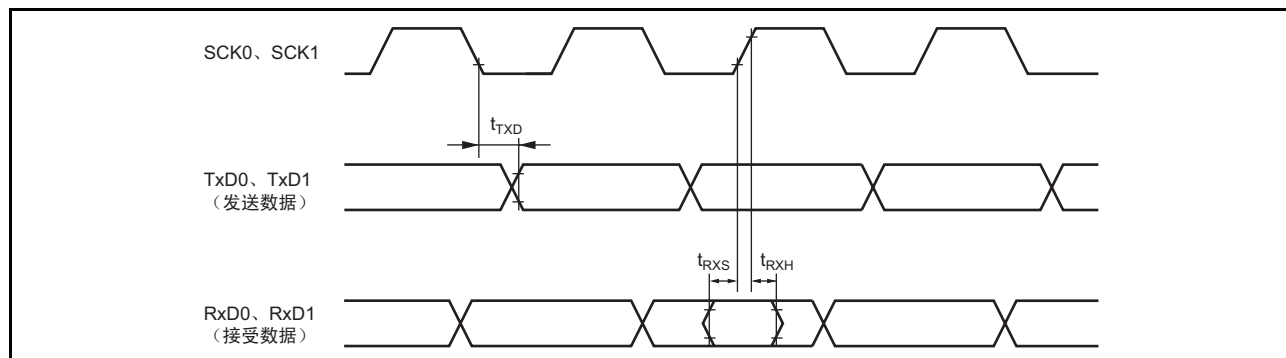


图 22.10 SCI 输入输出时序 / 时钟同步模式

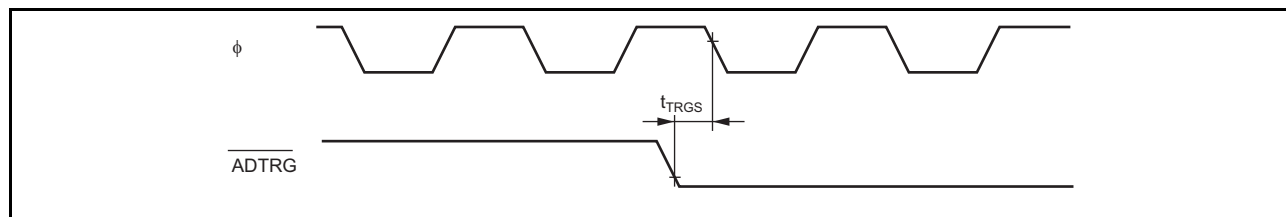


图 22.11 A/D 转换器外部触发器输入时序

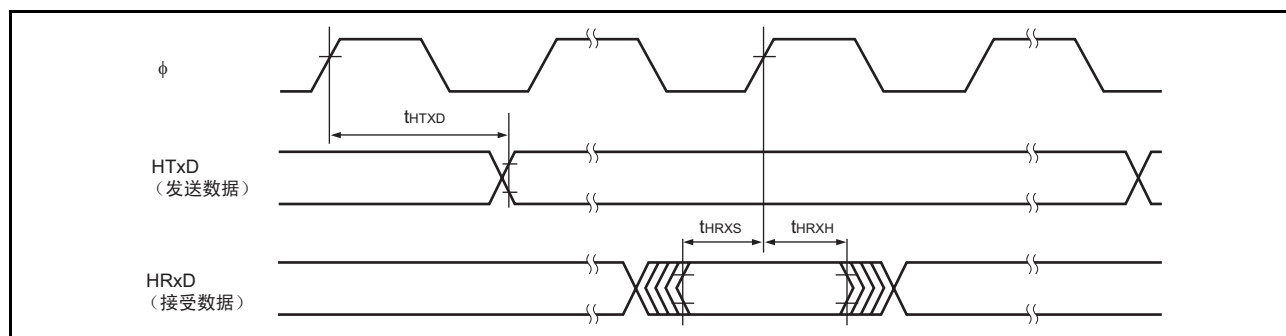


图 22.12 HCAN 输入 / 输出时序

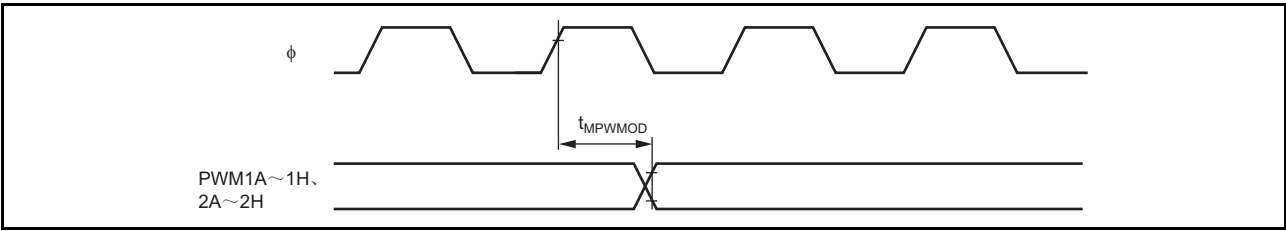


图 22.13 马达控制 PWM 输出时序

22.4 A/D 转换特性

A/D 转换特性如表 22.7 所示。

表 22.7 A/D 转换特性

条件: $V_{cc}=LPV_{cc}=4.5 \sim 5.5V$ 、 $AV_{cc}=4.5 \sim 5.5V$ 、 $PWMV_{cc}=4.5 \sim 5.5V$ 、 $V_{ss}=AV_{ss}=PWMV_{ss}=PLL_{Vss}=0V$ 、 $\phi=4 \sim 20MHz$ 、 $T_a=-20 \sim +75^{\circ}C$ (通常规格产品)、 $T_a=-40 \sim +85^{\circ}C$ (宽温度范围规格产品)

项目	min.	typ.	max.	单位
分辨率	10	10	10	位
转换时间	10	—	200	μs
模拟输入容量	—	—	20	pF
容许信号源阻抗	—	—	5	k Ω
非直线性误差	—	—	± 3.5	LSB
偏移误差	—	—	± 3.5	LSB
满刻度误差	—	—	± 3.5	LSB
量化误差	—	± 0.5	—	LSB
绝对精度	—	—	± 4.0	LSB

22.5 快速擦写存储器特性

快速擦写存储器特性如表 22.8 所示。

表 22.8 快速擦写存储器特性

条件: $V_{cc}=PVMV_{cc}=LPV_{cc}=4.5 \sim 5.5V$ 、 $AV_{cc}=4.5 \sim 5.5V$ 、 $V_{ss}=PVMV_{ss}=PLL_{Vss}=AV_{ss}=0V$ 、 $T_a=0 \sim +75^{\circ}C$ (写入 / 擦除时的运行温度范围)

项目		符号	min.	typ.	max.	单位	备注
写入时间 *1*2*4		t_P	—	10	200	ms/128 字节	
擦除时间 *1*3*5		t_E	—	100	1200	ms/ 块	
改写次数		N_{WEC}	—	—	100	次	
写入时	置 SWE 位后的等待时间 *1	t_{sswe}	1	1	—	μs	
	置 PSU 位后的等待时间 *1	t_{psu}	50	50	—	μs	
	置 P 位后的等待时间 *1*4	t_{sp30}	28	30	32	μs	写入时间等待
		t_{sp200}	198	200	202	μs	写入时间等待
		t_{sp10}	8	10	12	μs	追加写入时间等待
	清除 P 位后的等待时间 *1	t_{cp}	5	5	—	μs	
	清除 PSU 位后的等待时间 *1	t_{cpsu}	5	5	—	μs	
	置 PV 位后的等待时间 *1	t_{spv}	4	4	—	μs	
	虚写 H'FF 后的等待时间 *1	t_{spvr}	2	2	—	μs	
	清除 PV 位后的等待时间 *1	t_{cpv}	2	2	—	μs	
	清除 SWE 位后的等待时间 *1	t_{cswe}	100	100	—	μs	
	最大改写次数 *1*4	N	—	—	1000	次	
擦除时	置 SWE 位后的等待时间 *1	t_{sswe}	1	1	—	μs	
	置 ESU 位后的等待时间 *1	t_{sesu}	100	100	—	μs	擦除时间等待
	置 E 位后的等待时间 *1*5	t_{se}	10	10	100	ms	
	清除 E 位后的等待时间 *1	t_{ce}	10	10	—	μs	
	清除 ESU 位后的等待时间 *1	t_{cesu}	10	10	—	μs	
	置 EV 位后的等待时间 *1	t_{sev}	20	20	—	μs	
	虚写 H'FF 后的等待时间 *1	t_{sevr}	2	2	—	μs	
	清除 EV 位后的等待时间 *1	t_{cev}	4	4	—	μs	
	清除 SWE 位后的等待时间 *1	t_{cswe}	100	100	—	μs	
	最大擦除次数 *1*5	N	12	—	120	次	

【注】 *1 必须按照编程 / 擦除的算法，设置各时间。

*2 每 128 字节的写入时间（表示将快速擦写存储器控制寄存器 (FLMCR1) 的 P 位置位的总时间。
不含程序验证 时间。）

*3 擦除 1 块的时间（表示将 FLMCR1 的 E 位置位的总时间。不含擦除验证时间。）

*4 为了规定 128 字节编程算法的编程时间最大值 ($t_{p(max)}$)，必须设置最大编程次数 (N) 的值
为 max. 值 (1000) 另外，必须通过编程次数计数器 (n) 的次数，来切换 P 位置位后的等待时间，如下所示。

编程次数计数器 (n) 1 ~ 6 次时 $t_{sp30}=30\mu s$

编程次数计数器 (n) 7 ~ 1000 次时 $t_{sp200}=200\mu s$

（追加编程时间）

编程次数计数器 (n) 1 ~ 6 次时 $t_{sp10}=10\mu s$

*5 关于擦除时间的最大值 ($t_E(max)$)，与 E 位置位后的等待时间 (t_{se}) 和最大擦除次数 (N) 的关系如下。

$t_E(max)=E$ 位置位后的等待时间 (t_{se}) × 最大擦除次数 (N)

为了规定擦除时间的最大值，必须使 (t_{se}) 及 (N) 的值满足上述的计算式。

（例） $t_{se}=100ms$ 时，N=12 次

（例） $t_{se}=10ms$ 时，N=120 次

22.6 LCD 特性

LCD 特性如表 22.9 所示。

表 22.9 LCD 特性

条件：Vcc=LPVcc=4.5 ~ 5.5V、AVcc=4.5 ~ 5.5V、PWMVcc=4.5 ~ 5.5V、Vss=AVss=PWMVss=PLLvss=0V、
Ta=-20 ~ +75°C（通常规格产品）、Ta=-40 ~ +85°C（宽温度范围规格产品）

项目	符号	适用管脚	测定条件	min	typ	max	单位	参考
段驱动器下降电压	V _{DS}	SEG1 ~ SEG28 (H8S/2282 群、 HD64F2280B)	ID=2 μ A	—	—	0.6	V	*1
		SEG1 ~ SEG32 (HD64F2280RB)						
公共驱动器下降电压	V _{DC}	COM1 ~ COM4	ID=2 μ A	—	—	0.3	V	*1
LCD 电源分块电阻	R _{LCD}		V1 ~ Vss 间	40	300	1000	K Ω	
液晶显示电压	V _{LCD}	V1		4.5			V	*2

【注】 *1 从电源管脚、V1、V2、V3、Vss 到各段管脚或公共管脚的电压下降。

*2 由外部电源提供液晶显示电压时，必须保持 LPVcc $\geq$ V1 $\geq$ V2 $\geq$ V3 $\geq$ Vss 的关系。

附录

A. 在各管脚状态的 I/O 端口状态

端口名称	MCU 运行模式	复位	硬件待机模式	软件待机模式	子激活模式	程序执行状态 SLEEP
端口 1	7	T	T	Keep	输入 / 输出端口	输入 / 输出端口
端口 3	7	T	T	Keep	输入 / 输出端口	输入 / 输出端口
端口 4	7	T	T	T	输入端口	输入端口
端口 A	7	T	T	Keep	输入 / 输出端口	输入 / 输出端口
端口 B	7	T	T	Keep	输入 / 输出端口	输入 / 输出端口
端口 C	7	T	T	Keep	输入 / 输出端口	输入 / 输出端口
端口 D	7	T	T	Keep	输入 / 输出端口	输入 / 输出端口
PF7	7	T	T	[DDR=0] T [DDR=1] H	[DDR=0] T [DDR=1] H	[DDR=0] T [DDR=1] 时钟输出
PF6 PF5 PF4 PF3 PF2 PF1* ¹ PF0* ²	7	T	T	Keep	输入 / 输出端口	输入 / 输出端口
端口 H	7	T	T	Keep	输入 / 输出端口	输入 / 输出端口
端口 J	7	T	T	Keep	输入 / 输出端口	输入 / 输出端口
HTxD* ²	7	H	T	H	H	输出
HRxD* ²	7	输入	T	T	T	输入

【符号说明】

H: 高电平

T: 高阻抗

Keep: 输入端口为高阻抗
输出端口为保持

【注】 *1 H8S/2282 系列中不具备此项功能。

*2 H8S/2280 系列中不具备此项功能。

B. 产品型号名称一览

产品分类		产品型号	标记型号	封装（代码）
H8S/2282	F-ZTAT 版	HD64F2282	HD64F2282	100pinQFP（FP-100A）
	掩膜 ROM 版	HD6432282	HD6432282(***)	100pinQFP（FP-100A）
H8S/2281	掩膜 ROM 版	HD6432281	HD6432281(***)	100pinQFP（FP-100A）
H8S/2280B	F-ZTAT 版	HD64F2280B	HD64F2280	100pinQFP（FP-100A）
H8S/2280BR	F-ZTAT 版	HD64F2280RB	HD64F2280R	100pinQFP（FP-100A）

【符号说明】

(***) 为 ROM 代码。

【注】以上包含正在开发及策划中的产品。各产品的详细情况，请与我公司销售负责人进行确认。

C. 外形尺寸图以《瑞萨半导体器件封装数据手册》中记载的尺寸图为准。

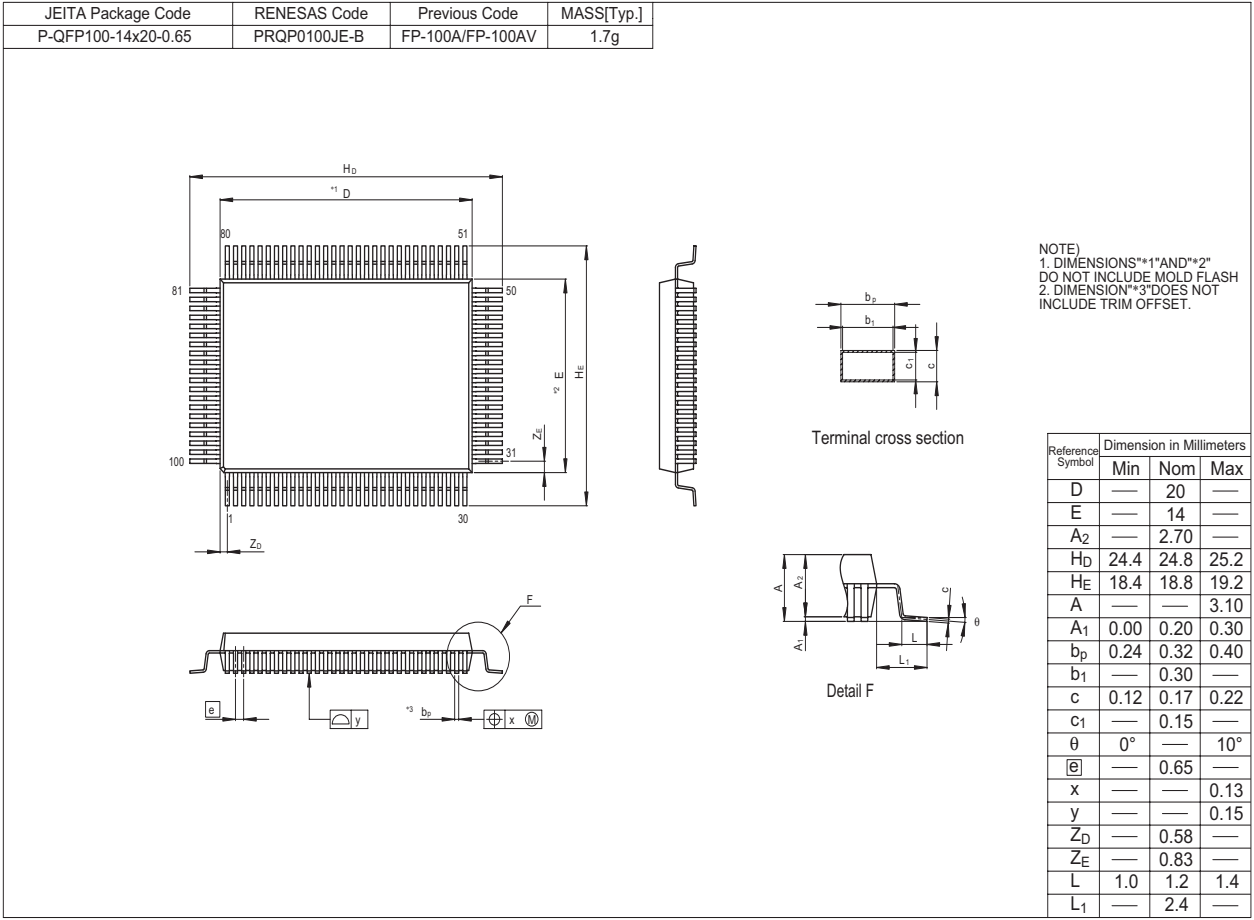


图 C.1 FP-100A 外形尺寸图

索引

数字

16 位定时器脉冲单元	126
交替输出	150
输入捕捉功能	152
通过比较匹配输出波形	150

A

A/D 转换器	277
单通道模式	282
模拟输入通道	280
扫描模式	283
外部触发器	284
转换时间	283

B

Bcc	33, 39
板载编程	328

C

CAN 总线模块	240
CAN 总线模块	
CAN 总线接口	273
发送信息	265
HCAN HALT 模式	272
HCAN 休眠模式	271
接收信息	268
软件复位	259
硬件复位	259
操作符号	34
操作字段	42
程序计数器 (PC)	28
串行通信接口	191
比特率	201
标志状态	235
断点	235
多处理器通信功能	215
奇偶校验错误	212
时钟同步模式	219
异步模式	206
溢出错误	212
帧错误	212
存储器周期	78

D

低功耗状态	
监视模式	377
模块待机模式	376
软件待机模式	373
睡眠模式	373

硬件待机模式	375
直接转移	378
中速模式	372
子激活模式	378
子睡眠模式	377
地址空间	25
地址映射	52
电机控制 PWM 定时器	289
PWM 通道 1	303
PWM 通道 2	304
堆栈指针 (SP)	27

E

EA 扩展字段	42
---------------	----

F

复位	56
----------	----

G

概要	277
工作模式的选择	50

J

寄存器	
ABACK	249, 381, 394, 407
ADCR	282, 392, 406, 418
ADCSR	281, 392, 406, 418
ADDR	280, 392, 406, 418
BCR	245, 381, 394, 407
BRR	201, 392, 405, 417
EBR1	327, 392, 406, 418
EBR2	327, 392, 406, 418
FLMCR1	326, 392, 406, 418
FLMCR2	326, 392, 406, 418
FLPWCR	328, 392, 406, 418
GSR	244, 381, 394, 407
IER	65, 389, 403, 415
IMR	254, 381, 394, 407
IPR	64, 390, 404, 415
IRR	251, 381, 394, 407
ISCR	65, 389, 403, 415
ISR	67, 389, 403, 415
LAFM	256, 381, 394, 407
LCR	310, 389, 403, 415
LCR2	311, 389, 403, 415
LPCR	308, 389, 403, 415
LPWRCR	370, 389, 403, 415
MBCR	247, 381, 394, 407
MBIMR	253, 381, 394, 407
MC	257, 381, 395, 407

MCR	243, 381, 394, 407
MD	259, 385, 398, 411
MDCR	50, 389, 403, 415
MSTPCR	371, 389, 403, 415
P1DDR	87, 389, 403, 415
P1DR	87, 390, 404, 416
P3DDR	96, 389, 403, 415
P3DR	96, 390, 404, 416
P3ODR	97, 390, 403, 415
PADDR	101, 389, 403, 415
PADR	101, 390, 404, 416
PAODR	102, 390, 403, 415
PBDDR	104, 390, 403, 415
PBDR	104, 390, 404, 416
PBODR	105, 390, 403, 415
PCDDR	107, 390, 403, 415
PCDR	107, 390, 404, 416
PCODR	108, 390, 403, 415
PDDDR	110, 390, 403, 415
PDDR	111, 390, 404, 416
PFDDR	112, 390, 403, 415
PFDR	113, 390, 404, 416
PHDDR	118, 389, 402, 415
PHDR	118, 389, 403, 415
PJDDR	121, 389, 402, 415
PJDR	121, 389, 403
PORT1	88, 392, 406, 418
PORT3	97, 392, 406, 418
PORT4	100, 393, 406, 418
PORTA	102, 393, 406, 418
PORTB	105, 393, 406, 418
PORTC	108, 393, 406, 418
PORTD	111, 393, 406, 418
PORTF	113, 393, 406, 418
PORTH	119, 389, 403, 415
PORTJ	122, 389, 403, 415
PWBFR	298, 388, 402, 414
PWCNT	294
PWCR	292, 388, 402, 414
PWCYR	295, 402, 414
PWDTR	296, 298
PWOCR	293, 388, 402, 414
PWPR	294, 388, 402, 414
RAMER	327, 390, 404, 416
RDR	193, 392, 405, 417
REC	255, 381, 394, 407
RFPR	250, 381, 394, 407
RSR	193
RSTCSR	186, 391, 405, 417
RXPR	249, 381, 394, 407
SBYCR	369, 389, 403, 415
SCKCR	359, 389, 403, 415
SCMR	200, 392, 406, 417
SCR	196, 392, 405, 417
SMR	194, 391, 405, 417
SSR	198, 392, 405, 417
SYSCR	51, 389, 403, 415

TCNT	147, 182, 391, 405, 406, 416, 418
TCR	131, 390, 404, 416
TCSR	183, 392, 405, 418
TDR	193, 392, 406, 417
TEC	255, 381, 394, 407
TGR	147, 391, 404, 416
TIER	391, 405, 416
TIOR	135, 391, 405, 416
TMDR	134, 391, 404, 416
TRPRT	124, 389, 403, 415
TSR	145, 391, 404, 416
TSTR	147, 390, 403, 415
TSYR	147, 390, 403, 415
TXACK	248, 381, 394, 407
TXCR	248, 381, 394, 407
TXPR	247, 381, 394, 407
UMSR	255, 381, 394, 407

寄存器一览表

各工作模式下的寄存器状态	407
寄存器地址一览表	381
寄存器位一览表	394
寄存器字段	42

K

开路输出控制寄存器	81
看门狗定时器	181
间隔定时器模式	188
看门狗定时器模式	186
溢出	188
快速擦写存储器	321
编程 / 擦除保护	338
擦除 / 擦除验证	336
擦除块	325
程序 / 程序验证	334
仿真	332
PROGRAMMER MODE	338
写入的单位	325
引导模式	329
用户编程模式	331
扩展寄存器 (EXR)	28

L

LCD 控制器 / 驱动器	
段驱动器	309
公共驱动器	308
LCD RAM	313
LCD 显示	312
占空比	306
LCD 控制器 / 驱动器	306

M

M16C/28 群	9, 191
MCU 工作模式	50

MD0	259	中断	
命令设置		ADI	285
操作符号	34	CMI	305
S		ERI	234
时钟振荡器	358	ERS0/OVR0	273
PLL 电路	363	NMI 中断	67, 188
数据方向寄存器	81	RM0	273
数据寄存器	81	RM1	273
T		RXI	234
TRAPA 指令	59	SLE0	273
条件码寄存器 (CCR)	29	TCI	168
条件字段	42	TEI	234
通用寄存器	27	TGR	168
X		TXI	234
寻址模式	43	WOVI	188
程序计数器相对寻址	45	中断控制模式	70
存储器间接寻址	45	中断控制器	62
带偏移量的寄存器间接寻址	43	中断屏蔽位	29
后增寄存器间接寻址	44	中断异常处理向量表	68
寄存器间接寻址	43	总线周期	78
寄存器直接寻址	43		
绝对地址	44		
立即数寻址	45		
先减寄存器间接寻址	44		
Y			
异常处理	54		
堆栈状态	60		
复位异常处理	56		
跟踪异常处理	58		
陷阱指令异常处理	59		
中断异常处理	58		
异常源和向量表	55		
有效地址	43, 46		
Z			
指令设置	33		
操作符号	34		
块传输指令	41		
逻辑运算指令	37		
数据传输指令	35		
位操作指令	38		
系统控制指令	40		
移位指令	37		
转移指令	39		
指令设置 * 操作符号	34		
智能卡接口	226		

**瑞萨 16 位单片机
硬件手册
H8S/2282 群、H8S/2280 群**

Publication Date: 1st Edition, Mar. 19, 2007
Published by: Sales Strategic Planning Div.
Renesas Technology Corp.
Edited by: Customer Support Department
Global Strategic Communication Div.
Renesas Solutions Corp.

Renesas Technology Corp. Sales Strategic Planning Div. Nippon Bldg., 2-6-2, Ohte-machi, Chiyoda-ku, Tokyo 100-0004, Japan



RENESAS SALES OFFICES

<http://www.renesas.com>

Refer to "<http://www.renesas.com/en/network>" for the latest and detailed information.

Renesas Technology America, Inc.

450 Holger Way, San Jose, CA 95134-1368, U.S.A
Tel: <1> (408) 382-7500, Fax: <1> (408) 382-7501

Renesas Technology Europe Limited

Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K.
Tel: <44> (1628) 585-100, Fax: <44> (1628) 585-900

Renesas Technology (Shanghai) Co., Ltd.

Unit 204, 205, AZIACenter, No.1233 Lujiazui Ring Rd, Pudong District, Shanghai, China 200120
Tel: <86> (21) 5877-1818, Fax: <86> (21) 6887-7898

Renesas Technology Hong Kong Ltd.

7th Floor, North Tower, World Finance Centre, Harbour City, 1 Canton Road, Tsimshatsui, Kowloon, Hong Kong
Tel: <852> 2265-6688, Fax: <852> 2730-6071

Renesas Technology Taiwan Co., Ltd.

10th Floor, No.99, Fushing North Road, Taipei, Taiwan
Tel: <886> (2) 2715-2888, Fax: <886> (2) 2713-2999

Renesas Technology Singapore Pte. Ltd.

1 Harbour Front Avenue, #06-10, Keppel Bay Tower, Singapore 098632
Tel: <65> 6213-0200, Fax: <65> 6278-8001

Renesas Technology Korea Co., Ltd.

Kukje Center Bldg. 18th Fl., 191, 2-ka, Hangang-ro, Yongsan-ku, Seoul 140-702, Korea
Tel: <82> (2) 796-3115, Fax: <82> (2) 796-2145

Renesas Technology Malaysia Sdn. Bhd

Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No.18, Jalan Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: <603> 7955-9390, Fax: <603> 7955-9510

H8S/2282 群、 H8S/2280 群



瑞萨电子株式会社

RCJ09B0051-0100