

To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: <http://www.renesas.com>

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (<http://www.renesas.com>)

Send any inquiries to <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.



用户手册初稿

V850ES/JG3-H, V850ES/JH3-H

32 位单片微控制器

硬件

V850ES/JG3-H

μPD70F3760

μPD70F3761

μPD70F3762

μPD70F3770

V850ES/JH3-H

μPD70F3765

μPD70F3766

μPD70F3767

μPD70F3771

[备忘录]

CMOS 设备注意事项

① 输入引脚处的电压波形

输入噪音或一个反射波引起的波形失真可能导致错误发生。如果由于噪音等的影响使CMOS设备的输入电压范围保持在VIL(MAX)和VIH(MIN)之间,设备可能发生错误。在输入电平固定时以及输入电平从VIL(MAX)过渡到VIH(MIN)时的传输期间,要防止散射噪声影响设备。

② 未使用的输入引脚的处理

CMOS设备的输入端保持开路可能导致误操作。如果一个输入引脚未被连接,则由于噪音等原因可能会产生内部输入电平,从而导致误操作。CMOS设备的操作特性与Bipolar或NMOS设备不同。CMOS设备的输入电平必须借助上拉或下拉电路固定在高电平或低电平。每一个未使用引脚都应该通过附加电阻连接到VDD或GND。如果有可能尽量定义为输出引脚。对未使用引脚的处理因设备而异,必须遵循与设备相关的规定和说明。

③ ESD防护措施

如果MOS设备周围有强电场,将会击穿氧化栅极,从而影响设备的运行。因此必须采取措施,尽可能防止静电产生。一旦有静电,必须立即释放。对于环境必须有适当的控制。如果空气干燥,应当使用增湿器。建议避免使用容易产生静电的绝缘体。半导体设备的存放和运输必须使用抗静电容器、抗静电屏蔽袋或导电材料容器。所有的测试和测量工具包括工作台和工作面必须良好接地。操作员应当佩戴静电消除手带以保证良好接地。不能用手直接接触半导体设备。对于装配有半导体设备的PW板也应采取类似的静电防范措施。

④ 初始化之前的状态

在上电时MOS设备的初始状态是不确定的。在刚刚上电之后,具有复位功能的MOS设备并没有被初始化。因此上电不能保证输出引脚的电平,I/O设置和寄存器的内容。设备在收到复位信号后才进行初始化。具有复位功能的设备在上电后必须立即进行复位操作。

⑤ 电源开关顺序

在一个设备的内部操作和外部接口使用不同的电源的情况下,按照规定,应先在接通内部电源之后再接通外部电源。当关闭电源时,按照规定,先关闭外部电源再关闭内部电源。如果电源开关顺序颠倒,可能会导致设备的内部组件过电压,产生异常电流,从而引起内部组件的误操作和性能的退化。对于每个设备电源的正确开关顺序必须依据设备的规范说明分别进行判断。

⑥ 电源关闭状态下的输入信号

不要向没有加电的设备输入信号或提供I/O上拉电源。因为输入信号或提供I/O上拉电源将引起电流注入,从而引起设备的误操作,并产生异常电流,从而使内部组件退化。每个设备电源关闭时的信号输入必须依据设备的规范说明分别进行判断。

注意事项：该产品使用的 SuperFlash®技术获得了 **Silicon Storage Technology, Inc.**公司的授权。

EEPROM 是 **NEC Electronics Corporation** 的商标。

IECUBE 是一个 **NEC Electronics Corporation** 在日本和德国的注册商标

MINICUBE 是一个 **NEC Electronics Corporation** 在日本,德国及美国的注册商标。

Applilet 是一个 **NEC Electronics Corporation** 在日本,德国, 香港, 中国, 韩国, 英国及美国的注册商标。

Windows和**WindowsNT**是**Microsoft Corporation**在美国及其他国家的注册商标和商标。

SuperFlash®是**Silicon Storage Technology, Inc.**的一个注册商标, 已经在美国和日本等几个国家使用。

PC/AT 是国际商业机器公司的一个商标。

PC/AT是**International Business Machines Corporation**的商标。

SPARCstation是**SPARC International, Inc.**的商标。

Solaris和**SunOS**是**Sun Microsystems, Inc.**的商标。

TRON 是 **The Real-Time Operating system Nucleus** 的缩写。

ITRON 是 **Industrial TRON** 的缩写。

- 本文档所登载的内容有效期截止至 2009 年 2 月，信息先于产品的生产周期发布。将来可能未经预先通知而更改。在实际进行生产设计时，请参阅各产品最新的数据表或数据手册等相关资料以获取本公司产品的最新规格。
- 并非所有的产品和/或型号都向每个国家供应。请向本公司销售代表查询产品供应及其他信息。
- 未经本公司事先书面许可，禁止复制或转载本文件中的内容。否则因本文档所登载内容引发的错误，本公司概不负责。
- 本公司对于因使用本文件中列明的本公司产品而引起的，对第三者的专利、版权以及其它知识产权的侵权行为概不负责。本文件登载的内容不应视为本公司对本公司或其他人所有的专利、版权以及其它知识产权作出任何明示或默示的许可及授权。
- 本文件中的电路、软件以及相关信息仅用以说明半导体产品的运作和应用实例。用户如在设备设计中应用本文件中的电路、软件以及相关信息，应自行负责。对于用户或其他人因使用了上述电路、软件以及相关信息而引起的任何损失，本公司概不负责。
- 虽然本公司致力于提高半导体产品的质量及可靠性，但用户应同意并知晓，我们仍然无法完全消除出现产品缺陷的可能。为了最大限度地减少因本公司半导体产品故障而引起的对人身、财产造成损害（包括死亡）的危险，用户务必在其设计中采用必要的安全措施，如冗余度、防火和防故障等安全设计。
- 本公司产品质量分为：

“标准等级”、“专业等级”以及“特殊等级”三种质量等级。

“特殊等级”仅适用于为特定用途而根据用户指定的质量保证程序所开发的日电电子产品。另外，各种日电电子产品的推荐用途取决于其质量等级，详见如下。用户在选用本公司的产品时，请事先确认产品的质量等级。

“标准等级”：计算机，办公自动化设备，通信设备，测试和测量设备，音频·视频设备，家电，加工机械以及产业用机器人。

“专业等级”：运输设备（汽车、火车、船舶等），交通用信号控制设备，防灾装置，防止犯罪装置，各种安全装置以及医疗设备（不包括专门为维持生命而设计的设备）。

“特殊等级”：航空器械，宇航设备，海底中继设备，原子能控制系统，为了维持生命的医疗设备、用于维持生命的装置或系统等。

除在本公司半导体产品的数据表或数据手册等资料中另有特别规定以外，本公司半导体产品的质量等级均为“标准等级”。如果用户希望在本公司设计意图以外使用本公司半导体产品，务必事先与本公司销售代表联系以确认本公司是否同意为该项应用提供支持。

（注）

- （1）本声明中的“本公司”是指日本电气电子株式会社（NEC Electronics Corporation）及其控股公司。
- （2）本声明中的“本公司产品”是指所有由日本电气电子株式会社开发或制造的产品或为日本电气电子株式会社（定义如上）开发或制造的产品。

M5 02.11-1

引言

读者	本手册适用于想了解 V850ES/JG3-H 和 V850ES/JH3-H 功能和设计软件及硬件应用系统的用户。
目的	本手册旨在提供给用户 V850ES/JG3-H 和 V850ES/JH3-H 功能描述的了解，如下面组织中所示
组织	本手册分成两个部分: 硬件 (本手册)和结构化(V850ES 结构化用户手册).

硬件	结构化
• 引脚功能 • CPU 功能 • 片上外围功能 • Flash 存储器编程 • 电气特性(目标值)	• 数据类型 • 寄存器设置 • 指令格式及设置 • 中断和 Interrupts and exceptions • 流水线操作

如何阅读本手册	在阅读本手册前，读者应掌握电子工程、逻辑电路和微控制器等电子工程方面的基础知识。

要了解 V850ES/JG3-H 和 V850ES/JH3-H 功能的概要

→ 请按**目录**顺序阅读本手册

要了解寄存器的名称

→ 请参考 **附录 C 寄存器索引**.

如何阅读寄存器格式

→ 尖括号(<>)中的二进制位名称在 RA78K0S 中被定义为保留字，并且在 CC78K0S 中用 #pragma sfr 指令定义为一个 sfr 变量。

有关指令的详细内容

→ 请参考 **V850ES 结构化用户手册**.

要了解 V850ES/JG3-H 和 V850ES/JH3-H 电气特性

→ 请参考 **第 33 章 电气特性 (目标值)**.

本手册中将“xxx 寄存器的 yyy 位”描述为“xxx.yyy 位”. 注意如果在程序中描述为“xxx.yyy”, 然而, 编译器/汇编器无法正确识别它。

“<R>”表示主要修点。在 PDF 文档中通过复制“<R>”，并指定到查找区能方便的查询到修改点。

规则	数据规则:	数据的高位部分在左边，低位部分在右边
	有效低电平表示法:	$\overline{\text{xxx}}$ (在引脚和信号名称上加划一条线)
	存储器地址分布:	高位地址在上面，低位地址在下面
	注:	文中用 注 标注的相关术语的脚注
	注意事项:	需要特别关注的信息
	备注:	补充信息
	数值的表示:	二进制 ... xxxx 或 xxxxB 十进制... xxxx 十六进制... xxxxH
	前缀表示 2 的乘幂	
	(地址空间, 存储器容量):	K (k): $2^{10} = 1,024$ M (兆): $2^{20} = 1,024^2$ G (千兆): $2^{30} = 1,024^3$

相关文档

请阅读下列与本手册有关的文献。本手册中提到的相关文档可能包括有初稿版本。但是，初稿版本没有特别注明。

V850ES/JG3-H 和 V850ES/JH3-H 相关文档

文档名称	文档编号
V850ES 结构化用户手册	U15943E
V850ES/JG3-H, V850ES/JH3-H 硬件用户手册	本手册

开发工具相关文档

文档名称		文档编号
QB-V850ESJX3H [※] 片上仿真		准备中
QB-V850MINI 片上调试仿真		U17638E
QB-MINI2 具有编程功能的片上调试仿真器		U18371E
CA850 Ver. 3.20 C 编译包	操作篇	U18512E
	C 语言篇	U18513E
	汇编语言篇	U18514E
	链接指令	U18515E
PM+ Ver. 6.30 项目管理器		U18416E
ID850QB Ver. 3.40 综合调试器	操作篇	U18604E
SM850 Ver. 2.50 系统仿真	操作篇	U16218E
SM850 Ver. 2.00 or 更高版本系统仿真	外部器件用户开放接口规范	U14873E
SM+ 系统仿真	操作篇	U18601E
	用户开放接口	U18212E
RX850 Ver. 3.20 c	基础	U13430E
	安装	U17419E
	技术	U13431E
	任务调试	U17420E
RX850 Pro Ver. 3.21 操作篇	基础	U18165E
	In-Structure	U18164E
	任务调试	U17422E
AZ850 Ver. 3.30 系统执行分析		U17423E
PG-FP4 Flash 存储器编程器		U15260E
PG-FP5 存储器编程器		U18865E

注 开发中

目录

第一章 导言	22
1.1 概要	22
1.2 特性	25
1.3 应用领域	27
1.4 订货信息	27
1.5 引脚配置（顶视图）	28
1.6 功能块结构	31
1.6.1 内部框图	31
1.6.2 内部组件	33
第二章 引脚功能	36
2.1 引脚功能列表	36
2.2 引脚状态	50
2.3 引脚 I/O 电路类型、I/O 缓冲器电源及未使用引脚的连接	51
2.4 注意事项	56
第三章 CPU 功能	57
3.1 特性	57
3.2 CPU 寄存器组	58
3.2.1 程序寄存器组	59
3.2.2 系统寄存器组	60
3.3 操作模式	66
3.3.1 指定操作模式	66
3.4 地址空间	67
3.4.1 CPU 地址空间	67
3.4.2 CPU 地址空间的绕回	68
3.4.3 内存映射	69
3.4.4 存储区域	71
3.4.5 地址空间使用建议	77
3.4.6 外设 I/O 寄存器	80
3.4.7 可编程的外设 I/O 寄存器	94
3.4.8 特殊寄存器	95
3.4.9 注意事项	99
第四章 端口功能	103
4.1 特性	103
4.2 端口基本配置	103
4.3 端口配置	105
4.3.1 端口 0	111
4.3.2 端口 1	116
4.3.3 端口 2（仅限于 V850ES/JH3-H）	117
4.3.4 端口 3	121
4.3.5 端口 4	126
4.3.6 端口 5	129

4.3.7	端口 6	136
4.3.8	端口 7	140
4.3.9	端口 9	142
4.3.10	端口 CM	152
4.3.11	端口 CS (仅限于 V850ES/JH3-H)	155
4.3.12	端口 CT	157
4.3.13	端口 DH (仅限于 V850ES/JH3-H)	160
4.3.14	端口 DL	162
4.4	使用复用功能时的端口寄存器设置	164
4.5	注意事项	175
4.5.1	设置端口引脚的注意事项	175
4.5.2	端口 n 寄存器 (Pn) 位操作指令注意事项	178
4.5.3	片上调试引脚的注意事项 (仅限于 V850ES/JG3-H)	179
4.5.4	P56/INTP05/DRST 引脚的注意事项	179
4.5.5	电源开启时 P10, P11 和 P53 引脚的注意事项	179
4.5.6	迟滞特性	179
第五章	总线控制功能	180
5.1	特性	180
5.2	总线控制引脚	181
5.2.1	访问片内 ROM 存储器、片内 RAM 存储器或片上外围输入/输出端口时的引脚状态	182
5.2.2	各种工作模式下的引脚状态	182
5.3	存储块功能	183
5.4	总线访问	184
5.4.1	总线访问时钟数目	184
5.4.2	总线宽度设置功能	184
5.4.3	根据不同总线宽度进行访问	185
5.5	等待功能	192
5.5.1	可编程等待功能	192
5.5.2	外部等待功能	193
5.5.3	可编程等待和外部等待间的关系	194
5.5.4	可编程地址等待功能	195
5.6	空闲状态插入功能	196
5.7	总线保持功能 (仅限于 V850ES/JH3-H)	197
5.7.1	功能概述	197
5.7.2	总线保持步骤	198
5.7.3	节电模式时的操作	198
5.8	总线优先权	199
5.9	总线时序	200
第六章	时钟发生功能	203
6.1	概述	203
6.2	配置	204
6.3	寄存器	206
6.4	操作	211
6.4.1	各时钟的操作	211
6.4.2	时钟输出功能	211
6.5	PLL 功能	212

6.5.1	概述	212
6.5.2	寄存器	212
6.5.3	用途	215
第七章	16 位定时器/事件计数器 AA (TAA)	216
7.1	概述	216
7.2	功能	216
7.3	配置	217
7.3.1	引脚配置	219
7.4	寄存器	220
7.5	操作	237
7.5.1	间隔定时器模式 (TAAmMD2 至 TAAmMD0 位 = 000)	243
7.5.2	外部事件计数器模式 (TAAAnMD2 至 TAAAnMD0 位 = 001)	253
7.5.3	外部触发脉冲输出模式 (TAAAnMD2 至 TAAAnMD0 位 = 010)	261
7.5.4	单次脉冲输出模式 (TAAAnMD2 至 TAAAnMD0 位 = 011)	273
7.5.5	PWM (脉宽调制信号) 输出模式 (TAAAnMD2 至 TAAAnMD0 位 = 100)	280
7.5.6	自由运行定时器模式 (TAAAnMD2 至 TAAAnMD0 位 = 101)	289
7.5.7	脉冲宽度测量模式 (TAAAnMD2 至 TAAAnMD0 位 = 110)	306
7.5.8	定时器输出操作	311
7.6	定时器调谐操作功能	312
7.6.1	自由运行定时器模式 (在定时器调谐操作期间)	314
7.6.2	PWM 输出模式 (定时器调谐操作期间)	321
7.7	同时启动功能	323
7.7.1	PWM 输出模式 (同时启动操作)	324
7.8	级联连接	326
7.9	选择器功能	331
7.10	注意事项	332
第八章	16 位定时器/事件计数器 AB (TAB)	333
8.1	概述	333
8.2	功能	333
8.3	配置	334
8.4	寄存器	337
8.5	操作	354
8.5.1	间隔定时器模式 (TABnMD2 至 TABnMD0 位 = 000)	355
8.5.2	外部事件计数器模式 (TABnMD2 至 TABnMD0 位 = 001)	364
8.5.3	外部触发脉冲输出模式 (TABnMD2 至 TABnMD0 位 = 010)	373
8.5.4	单次脉冲输出模式 (TABnMD2 至 TABnMD0 位 = 011)	386
8.5.5	PWM 输出模式 (TABnMD2 至 TABnMD0 位 = 100)	395
8.5.6	自由运行定时器模式 (TABnMD2 至 TABnMD0 位 = 101)	406
8.5.7	脉冲宽度测量模式 (TABnMD2 至 TABnMD0 位 = 110)	426
8.5.8	三角波 PWM 模式 (TABnMD2 至 TABnMD0 位 = 111)	432
8.5.9	定时器输出操作	434
8.6	定时器调谐操作功能/同时启动功能	435
8.7	注意事项	436
第九章	16 位定时器/事件计数器 T (TMT)	437
9.1	概述	437

9.2	功能	437
9.3	配置	438
9.3.1	引脚配置	441
9.4	寄存器	442
9.5	定时器输出操作	463
9.6	操作	464
9.6.1	间隔定时器模式 (TT0MD3 至 TT0MD0 位=0000)	472
9.6.2	外部事件计数器模式 (TT0MD3 至 TT0MD0 位=0001)	482
9.6.3	外部触发脉冲输出模式 (TT0MD3 至 TT0MD0 位=0010)	492
9.6.4	单次脉冲输出模式 (TT0MD3 至 TT0MD0 位=0011)	505
9.6.5	PWM 输出模式 (TT0MD3 至 TT0MD0 位=0100)	512
9.6.6	自由运行定时器模式 (TT0MD3 至 TT0MD0 位=0101)	521
9.6.7	脉冲宽度测量模式 (TT0MD3 至 TT0MD0 位=0110)	537
9.6.8	三角波 PWM 输出模式 (TT0MD3 至 TT0MD0 位=0111)	543
9.6.9	编码器计数功能	545
9.6.10	编码器比较模式 (TT0MD3 至 TT0MD0 位=1000)	561
第十章	16 位间隔定时器 M (TMM)	569
10.1	概述	569
10.2	配置	570
10.3	寄存器	572
10.4	操作	574
10.4.1	间隔定时器模式	574
10.4.2	注意事项	578
第十一章	马达控制功能	579
11.1	功能概述	579
11.2	配置	580
11.3	控制寄存器	584
11.4	操作	594
11.4.1	系统概述	594
11.4.2	死区时间控制 (负相位波形信号的产生)	599
11.4.3	中断选择功能	606
11.4.4	重写带有传送功能的寄存器的操作	613
11.4.5	A/D 转换开始触发信号输出的 TAA4 联合操作	631
11.4.6	A/D 转换开始触发输出功能	634
第十二章	实时计数器	639
12.1	功能	639
12.2	配置	640
12.2.1	引脚配置	642
12.2.2	中断功能	642
12.3	寄存器	643
12.4	操作	658
12.4.1	初始设置	658
12.4.2	时钟操作期间重写各计数器	659
12.4.3	时钟操作期间读取各计数器	660
12.4.4	时钟操作期间改变 INTRTC0 中断设置	661

12.4.5	时钟操作期间改变 INTRTC1 中断设置	662
12.4.6	初始化 INTRTC2 中断设置	663
12.4.7	时钟操作期间改变 INTRTC2 中断设置	664
12.4.8	初始化实时计数器	665
12.4.9	实时计数器的钟表错误校正举例	666
第十三章	看门狗定时器 2 的功能	670
13.1	功能	670
13.2	配置	671
13.3	寄存器	672
13.4	操作	674
第十四章	实时输出功能 (RTO)	675
14.1	功能	675
14.2	配置	676
14.3	寄存器	678
14.4	操作	680
14.5	用途	681
14.6	注意事项	681
第十五章	A/D 转换器	682
15.1	概述	682
15.2	功能	682
15.3	配置	683
15.4	寄存器	686
15.5	操作	697
15.5.1	基本操作	697
15.5.2	转换操作时序	698
15.5.3	触发模式	699
15.5.4	操作模式	701
15.5.5	掉电比较模式	705
15.6	注意事项	710
15.7	A/D 转换器特征表的阅读方法	714
第十六章	D/A 转换器	718
16.1	功能	718
16.2	配置	718
16.3	寄存器	719
16.4	操作	721
16.4.1	正常模式下的操作	721
16.4.2	实时输出模式下的操作	721
16.4.3	注意事项	722
第十七章	异步串行接口 C (UARTC)	723
17.1	特性	723
17.2	配置	724
17.3	UARTC 和其它串行接口模式转换	726

17.3.1	UARTC0 和 CSIF4 间的模式转换	726
17.3.2	UARTC1 和 I ² C02 间的模式转换	727
17.3.3	UARTC2 和 CSIF3 间的模式转换	728
17.3.4	UARTC3, I ² C00 和 CAN0 间的模式转换	729
17.3.5	UARTC4, CSIF0 和 I ² C01 间模式的转换	730
17.4	寄存器.....	731
17.5	中断请求信号.....	741
17.6	操作	742
17.6.1	数据格式	742
17.6.2	SBF 发送/接收格式	744
17.6.3	SBF 发送	746
17.6.4	SBF 接收	747
17.6.5	UART 发送	748
17.6.6	连续发送过程	749
17.6.7	UART 接收	751
17.6.8	接收错误	751
17.6.9	奇偶类型和操作	754
17.6.10	接收数据噪声滤波器	755
17.7	专用波特率发生器	756
17.8	注意事项.....	764
第十八章	3-线可变长度串行 I/O (CSIF)	765
18.1	CSIF 和其它串行接口模式转换	765
18.1.1	CSIF4 和 UARTC0 模式转换	765
18.1.2	CSIF0、UARTC4 和 I ² C01 模式转换	766
18.1.3	CSIF3 和 UARTC2 模式转换.....	767
18.2	特性	768
18.3	配置	769
18.4	寄存器.....	772
18.5	中断请求信号.....	779
18.6	操作	780
18.6.1	单次传输模式 (主机模式, 发送模式)	780
18.6.2	单次传输模式 (主机模式, 接收模式)	782
18.6.3	单次传输模式 (主机模式, 发送/接收模式)	784
18.6.4	单次传输模式 (从机模式, 发送模式)	786
18.6.5	单次传输模式 (从机模式, 接收模式)	788
18.6.6	单次传输模式 (从机模式, 发送/接收模式)	790
18.6.7	连续传输模式 (主机模式, 发送模式)	792
18.6.8	连续传输模式 (主机模式, 接收模式)	794
18.6.9	连续传输模式 (主机模式, 发送/接收模式)	797
18.6.10	连续传输模式 (从机模式, 发送模式)	801
18.6.11	连续传输模式 (从机模式, 接收模式)	803
18.6.12	连续传输模式 (从机模式, 发送/接收模式)	806
18.6.13	接收错误	810
18.6.14	时钟时序.....	811
18.7	输出引脚.....	813
18.8	波特率发生器.....	814
18.8.1	波特率的产生.....	815

18.9	注意事项.....	816
第十九章	I ² C 总线.....	817
19.1	I ² C 总线和其它串行接口模式转换.....	817
19.1.1	UARTC3 和 I ² C00 模式转换.....	817
19.1.2	UARTC4, CSIF0 和 I ² C01 模式转换.....	818
19.1.3	UARTC1 和 I ² C02 模式转换.....	819
19.2	特性.....	820
19.3	配置.....	821
19.4	寄存器.....	825
19.5	I ² C 总线模式功能.....	840
19.5.1	引脚配置.....	840
19.6	I ² C 总线的定义和控制方法.....	841
19.6.1	开始条件.....	841
19.6.2	地址.....	842
19.6.3	传输方向指示.....	843
19.6.4	ACK.....	844
19.6.5	停止条件.....	845
19.6.6	等待状态.....	846
19.6.7	等待状态取消方法.....	848
19.7	I ² C 中断请求信号 (INTIICn).....	849
19.7.1	主机设备操作.....	849
19.7.2	从设备操作 (当接收从机地址数据 (地址匹配) 时).....	852
19.7.3	从机设备操作 (当接收扩展码时).....	856
19.7.4	不通信时的操作.....	860
19.7.5	仲裁失败操作 (从设备在仲裁失败后的操作).....	860
19.7.6	当仲裁失败发生时的操作 (在仲裁失败后不进行通信).....	862
19.8	中断请求信号 (INTIICn) 的产生时序和等待控制.....	869
19.9	地址匹配检测方法.....	871
19.10	错误检测.....	871
19.11	扩展代码.....	871
19.12	仲裁.....	872
19.13	唤醒功能.....	873
19.14	通信保留.....	874
19.14.1	当通信保留功能使能时 (IICFn.IICRSVn 位 = 0).....	874
19.14.2	当通信保留功能被禁止时 (IICFn.IICRSVn 位 = 1) 时.....	878
19.15	注意事项.....	879
19.16	通信操作.....	880
19.16.1	主机工作于单主机系统.....	881
19.16.2	主机工作于多主机系统.....	882
19.16.3	从机操作.....	885
19.17	数据通信时序.....	888
第二十章	CAN 控制器.....	895
20.1	概述.....	895
20.1.1	特性.....	895
20.1.2	功能概述.....	896
20.1.3	配置.....	897

20.2	CAN 协议	898
20.2.1	帧格式.....	898
20.2.2	帧类型.....	899
20.2.3	数据帧和远程帧	899
20.2.4	错误帧.....	907
20.2.5	过载帧.....	908
20.3	功能	909
20.3.1	决定总线优先级	909
20.3.2	位填充.....	909
20.3.3	多主机方式	909
20.3.4	多播方式	909
20.3.5	CAN 睡眠模式/CAN 停止模式功能.....	910
20.3.6	错误控制功能.....	910
20.3.7	波特率控制功能	914
20.4	与目标系统的连接	919
20.5	CAN 控制器的内部寄存器	920
20.5.1	CAN 控制器配置	920
20.5.2	寄存器访问类型	921
20.5.3	寄存器位配置	938
20.6	寄存器.....	942
20.7	位设置/清除功能.....	975
20.8	CAN 控制器初始化.....	978
20.8.1	CAN 模块的初始化.....	978
20.8.2	报文缓存器的初始化	978
20.8.3	报文缓存器的重新定义.....	978
20.8.4	从初始化模式到操作模式的转换	979
20.8.5	复位 CAN 模块的错误计数器 C0ERC	980
20.9	报文接收.....	981
20.9.1	报文接收	981
20.9.2	接收记录列表功能.....	982
20.9.3	屏蔽功能	984
20.9.4	多缓存接收块功能.....	986
20.9.5	远程帧接收	987
20.10	报文发送.....	988
20.10.1	报文发送	988
20.10.2	发送记录列表功能.....	990
20.10.3	自动块传输 (ABT)	992
20.10.4	发送中止处理.....	993
20.10.5	远程帧发送	994
20.11	节电模式.....	995
20.11.1	CAN 睡眠模式.....	995
20.11.2	CAN 停止模式.....	996
20.11.3	使用节电模式的举例	998
20.12	中断功能.....	999
20.13	诊断功能和特殊操作模式	1000
20.13.1	只收模式	1000
20.13.2	单次模式	1001
20.13.3	自检模式	1002

20.14	时间标记功能	1003
20.14.1	时间标记功能	1003
20.15	波特率设置	1005
20.15.1	波特率设置条件	1005
20.15.2	波特率设置的典型示例	1009
20.16	CAN 控制器的操作	1013
第二十一章 USB 功能控制器 (USBF)		1038
21.1	概述	1038
21.2	配置	1039
21.2.1	框图	1039
21.2.2	USB 存储器映射	1040
21.3	外部电路结构	1041
21.3.1	概要	1041
21.3.2	连接配置	1042
21.4	注意事项	1044
21.5	请求	1045
21.5.1	自动请求	1045
21.5.2	其它请求	1052
21.6	寄存器配置	1053
21.6.1	USB 控制寄存器	1053
21.6.2	USB 功能控制器寄存器列表	1055
21.6.3	EPC 控制寄存器	1071
21.6.4	数据保持寄存器	1123
21.6.5	EPC 请求数据寄存器	1146
21.6.6	桥寄存器	1161
21.6.7	DMA 寄存器	1165
21.6.8	Bulk-in 寄存器	1169
21.6.9	Bulk-out 寄存器	1170
21.6.10	外设控制寄存器	1171
21.7	STALL 握手信号或不握手信号	1175
21.8	特殊状态下的寄存器值	1176
21.9	FW 处理	1178
21.9.1	初始化处理	1180
21.9.2	中断服务	1183
21.9.3	USB 主处理	1184
21.9.4	中止/重传处理	1210
21.9.5	加电之后的处理	1213
21.9.6	在 DMA 模式下接收 bulk 传输 (OUT) 数据	1216
21.9.7	在 DMA 模式下发送 bulk 传输 (IN) 数据	1220
第二十二章 DMA 功能 (DMA 控制器)		1225
22.1	特性	1225
22.2	配置	1226
22.3	寄存器	1227
22.4	传送目标	1236
22.5	传送模式	1236
22.6	传送类型	1237

22.7	DMA 通道优先级	1238
22.8	与 DMA 传送有关的时间	1238
22.9	DMA 传送开始条件	1239
22.10	DMA 中止条件.....	1240
22.11	DMA 传送结束.....	1240
22.12	操作时序.....	1240
22.13	注意事项.....	1245
第二十三章 中断/异常处理功能		1250
23.1	特性	1250
23.2	不可屏蔽中断.....	1261
23.2.1	操作	1263
23.2.2	恢复	1264
23.2.3	NP 标志	1265
23.3	可屏蔽中断	1266
23.3.1	操作	1266
23.3.2	恢复	1268
23.3.3	可屏蔽中断的优先级	1269
23.3.4	中断控制寄存器 (xxICn)	1273
23.3.5	中断屏蔽寄存器 0 至 5 (IMR0 至 IMR5)	1277
23.3.6	服务中优先级寄存器 (ISPR)	1279
23.3.7	ID 标志.....	1280
23.3.8	看门狗定时器模式寄存器 2 (WDTM2)	1280
23.4	软件异常	1281
23.4.1	操作	1281
23.4.2	恢复	1282
23.4.3	EP 标志.....	1283
23.5	异常陷阱	1284
23.5.1	非法操作码	1284
23.5.2	调试陷阱	1286
23.6	外部中断请求输入引脚 (NMI 和 INTP00 至 INTP18)	1288
23.6.1	噪声抑制	1288
23.6.2	边沿检测	1288
23.7	CPU 的中断响应时间	1296
23.8	中断不被 CPU 响应期间	1297
23.9	注意事项.....	1297
第二十四章 按键中断功能.....		1298
24.1	功能	1298
24.2	寄存器.....	1299
24.3	注意事项.....	1299
第二十五章 待机功能		1300
25.1	概述	1300
25.2	寄存器.....	1302
25.3	HALT 模式.....	1305
25.3.1	设置和操作状态	1305
25.3.2	解除 HALT 模式	1305

25.4	IDLE1 模式	1307
25.4.1	设置和操作状态	1307
25.4.2	解除 IDLE1 模式	1308
25.5	IDLE2 模式	1310
25.5.1	设置和操作状态	1310
25.5.2	解除 IDLE2 模式	1311
25.5.3	解除 IDLE2 模式时确保建立时间	1313
25.6	STOP 模式	1314
25.6.1	设置和操作状态	1314
25.6.2	解除 STOP 模式	1314
25.6.3	在解除 STOP 模式时确保振荡稳定时间	1317
25.7	副时钟工作模式	1318
25.7.1	设置和操作状态	1318
25.7.2	解除副时钟工作模式	1318
25.8	Sub-IDLE 模式	1320
25.8.1	设置和操作状态	1320
25.8.2	解除 sub-IDLE 模式	1320
第二十六章	复位功能	1322
26.1	概述	1322
26.2	检测复位源的寄存器	1323
26.3	操作	1324
26.3.1	通过 RESET 引脚的复位操作	1324
26.3.2	通过看门狗定时器 2 的复位操作	1326
26.3.3	通过低电压检测器进行复位操作	1328
26.3.4	复位解除后的操作	1329
26.3.5	复位功能操作流程	1330
第二十七章	时钟监控器	1331
27.1	功能	1331
27.2	配置	1331
27.3	寄存器	1332
27.4	操作	1333
第二十八章	低电压检测器 (LVI)	1336
28.1	功能	1336
28.2	配置	1336
28.3	寄存器	1337
28.4	操作	1339
28.4.1	用作内部复位信号	1339
28.4.2	用作中断	1340
28.5	RAM 保持电压检测操作	1341
第二十九章	CRC 功能	1342
29.1	功能	1342
29.2	配置	1342
29.3	寄存器	1343

29.4	操作	1344
29.5	使用方法	1345
第三十章	稳压器	1347
30.1	概述	1347
30.2	操作	1348
第三十一章	FLASH 存储器	1349
31.1	特性	1349
31.2	存储器配置	1350
31.3	功能概述	1351
31.4	通过专用 Flash 编程器重写	1354
31.4.1	编程环境	1354
31.4.2	通信模式	1355
31.4.3	Flash 存储器控制	1366
31.4.4	通信模式的选择	1367
31.4.5	通信命令	1368
31.4.6	引脚连接	1369
31.5	通过自编程重写	1373
31.5.1	概述	1373
31.5.2	特性	1374
31.5.3	标准自编程流程	1375
31.5.4	Flash 功能	1376
31.5.5	引脚处理	1376
31.5.6	使用的内部资源	1377
第三十二章	片上调试功能	1378
32.1	用 DCU 调试	1379
32.1.1	连接电路示例	1379
32.1.2	接口信号	1379
32.1.3	可屏蔽功能	1381
32.1.4	寄存器	1381
32.1.5	操作	1383
32.1.6	注意事项	1383
32.2	不使用 DCU 调试	1384
32.2.1	电路连接举例	1384
32.2.2	可屏蔽功能	1387
32.2.3	用户资源安全性	1388
32.2.4	注意事项	1395
32.3	ROM 安全功能	1396
32.3.1	安全 ID	1396
32.3.2	设置	1397
第三十三章	电气规范（目标）	1399
33.1	最大绝对额定值	1399
33.2	电容	1401
33.3	工作条件	1401

33.4	振荡器特性	1402
33.4.1	主时钟振荡器特性.....	1402
33.4.2	副时钟振荡器特性.....	1403
33.4.3	PLL 特性.....	1404
33.4.4	内部振荡器特性	1404
33.5	DC 特性.....	1405
33.5.1	I/O 电平	1405
33.5.2	供应电流.....	1407
33.6	数据保持特性	1408
33.7	AC 特性.....	1409
33.7.1	CLKOUT 输出时序.....	1410
33.7.2	总线时序.....	1411
33.8	基本操作.....	1418
33.9	Flash 存储器编程特性	1431
第三十四章 封装图		1433
附录 A 开发工具		1435
A.1	软件包	1437
A.2	语言处理软件	1437
A.3	控制软件.....	1437
A.4	调试工具（硬件）	1438
A.4.1	当使用 IECUBE QB-V850ESJX3H ^注 时	1438
A.4.2	使用 MINICUBE QB-V850MINI 时	1441
A.4.3	使用 MINICUBE2 QB-MINI2 时	1442
A.5	调试工具（软件）	1443
A.6	嵌入式软件	1444
A.7	Flash 存储器写入工具	1444
附录 B V850ES/Jx3-H 和 V850ES/Jx3 之间的主要区别		1445
附录 C 寄存器索引		1446
附录 D 指令集列表		1460
D.1	习惯用法.....	1460
D.2	指令集 (按字母先后次序)	1463
附录 E 改版历史		1470
E.1	本版中主要改版内容	1470

第一章 引言

V850ES/JG3-H 和 V850ES/JH3-H 是 NEC 电子的 V850 单片机低功耗系列的产品，用于实时控制系统。

1.1 概要

V850ES/JG3-H 和 V850ES/JH3-H 都是 32 位单片机，采用 V850ES CPU 内核并整合了外设功能，如 ROM/RAM、定时器/计数器、串行接口、A/D 转换器、D/A 转换器、DMA 控制器、CAN 以及 USB 功能控制器。

除了高实时响应特性和单时钟基本指令外，V850ES/JG3-H 和 V850ES/JH3-H 单片机还具有通过硬件乘法器来实现的乘法指令、溢出操作指令和位处理指令等特性。

表 1-1 列出了 V850ES/JG3-H 的产品，表 1-2 列出了 V850ES/JH3-H 的产品。

表 1-1. V850ES/JG3-H 产品列表

通称		V850ES/JG3-H			
型号		μ PD70F3760	μ PD70F3761	μ PD70F3762	μ PD70F3770
内部存储器	Flash 存储器	256 KB	384 KB	512 KB	256 KB
	RAM ^{注 1}	40 KB	48 KB	56 KB	40 KB
存储空间	逻辑空间	64 MB			
	外部存储区	64 KB			
外部总线接口		地址数据总线: 16 复用总线			
通用寄存器		32 位 $\times$ 32 寄存器			
时钟	主时钟	PLL 模式: $f_x = 3 \sim 6$ MHz, $f_{xx} = 24 \sim 48$ MHz (乘以 8) 时钟直通模式: $f_x = 3 \sim 6$ MHz (内部: $f_{xx} = 3 \sim 6$ MHz)			
	副时钟	$f_{XT} = 32.768$ kHz			
	内部振荡器	$f_R = 220$ kHz (TYP.)			
	最短指令执行时间	20.8 ns (主时钟 (f_{xx}) = 48 MHz)			
I/O 端口 (耐压 5 V)		I/O: 77 (24)			
定时器	16-位 TAA	6 通道 (包括只用于间隔定时器功能的 1 通道)			
	16-位 TAB	2 通道			
	16-位 TMM	4 通道			
	16-位 TMT	1 通道			
	马达控制	1 通道 (结合 TAA 和 TAB 的功能, 包括 Hi-Z 输出控制功能)			
	钟表定时器	1 通道 (RTC)			
	WDT	1 通道			
实时输出功能		6 位 $\times$ 1 通道			
10-位 A/D 转换器		12 通道			
8-位 D/A 转换器		2 通道			
串行接口	CSIF/UARTC	2 通道	2 通道	2 通道	2 通道
	CSIF/UARTC/I ² C	1 通道	1 通道	1 通道	1 通道
	CSIF	2 通道	2 通道	2 通道	2 通道
	UARTC/I ² C	2 通道	2 通道	2 通道	1 通道
	UARTC/I ² C/CAN	—	—	—	1 通道
	USB 功能	1 通道	1 通道	1 通道	1 通道
DMA 控制器		4 通道 (传送目标: 片上外围 I/O、内部 RAM、外部存储器)			
中断源	外部 ^{注 2,3}	17 (17)	17 (17)	17 (17)	17 (17)
	内部	69	69	69	73
节电功能		HALT/IDLE1/IDLE2/STOP/副时钟/sub-IDLE 模式			
复位源		RESET 引脚输入、看门狗定时器 2 (WDT2)、时钟监视器(CLM)、低电压检测器(LVI)			
片上调试		支持 MINICUBE [®] 、MINICUBE2			
工作电源供电电压		2.85~3.6 V			
工作环境温度		-40~+85°C			
封装		100-引脚 LQFP 塑料封装 (密间距) (14 $\times$ 14 mm)			

- 注
1. 包括 8KB 的数据(仅)RAM 区域。
 2. 括号中的数字表示能够解除 STOP 模式的外部中断数目。
 3. 包括 NMI。

表 1-2. V850ES/JH3-H 产品列表

通称		V850ES/JH3-H			
型号		μ PD70F3765	μ PD70F3766	μ PD70F3767	μ PD70F3771
内部存储器	Flash 存储器	256 KB	384 KB	512 KB	256 KB
	RAM ^{注1}	40 KB	48 KB	56 KB	40 KB
存储空间	逻辑空间	64 MB			
	外部存储区	13 MB			
外部总线接口		地址总线: 24 地址数据总线: 16 独立总线/复用总线			
通用寄存器		32 位 $\times$ 32 检测器			
时钟	主时钟	PLL 模式: $f_x = 3 \sim 6$ MHz, $f_{xx} = 24 \sim 48$ MHz (乘以 8) 时钟直通模式: $f_x = 3 \sim 6$ MHz (内部: $f_{xx} = 3 \sim 6$ MHz)			
	副时钟	$f_{XT} = 32.768$ kHz			
	内部振荡器	$f_R = 220$ kHz (TYP.)			
	最短指令执行时间	20.8 ns (主时钟(f_{xx}) = 48 MHz)			
I/O 端口 (耐压 5 V)		I/O: 96 (27)			
定时器	16-位 TAA	6 通道 (包括只用于间隔定时器功能的 1 通道)			
	16-位 TAB	2 通道			
	16-位 TMM	4 通道			
	16-位 TMT	1 通道			
	马达控制	1 通道 (结合 TAA 和 TAB 的功能, 包括 Hi-Z 输出控制功能)			
	钟表定时器	1 通道 (RTC)			
	WDT	1 通道			
实时输出功能		6 位 $\times$ 1 通道			
10-位 A/D 转换器		12 通道			
8-位 D/A 转换器		2 通道			
串行接口	CSIF/UARTC	2 通道	2 通道	2 通道	2 通道
	CSIF/UARTC/I ² C	1 通道	1 通道	1 通道	1 通道
	CSIF	2 通道	2 通道	2 通道	2 通道
	UARTC/I ² C	2 通道	2 通道	2 通道	1 通道
	UARTC/I ² C/CAN	—	—	—	1 通道
	USB 功能	1 通道	1 通道	1 通道	1 通道
DMA 控制器		4 通道 (传送目标: 片上外围 I/O、内部 RAM、外部存储器)			
中断源	外部 ^{注2,3}	20 (20)			
	内部	69	69	69	73
节电功能		HALT/IDLE1/IDLE2/STOP/副时钟/sub-IDLE 模式			
复位源		RESET 引脚输入、看门狗定时器 2 (WDT2)、时钟监视器(CLM)、低电压检测器(LVI)			
片上调试		支持 MINICUBE、MINICUBE2			
工作电源供电电压		2.85~3.6 V			
工作环境温度		-40~+85°C			
封装		128-引脚 LQFP 塑料封装 (密间距) (14 $\times$ 20 mm)			

- 注
1. 包括 8KB 的数据(仅)RAM 区域。
 2. 括号中的数字表示能够解除 STOP 模式的外部中断数目。
 3. 包括 NMI。

1.2 特性

- 最短指令执行时间： 20.8 ns（主时钟(f_{xx}) = 48 MHz: $V_{\text{DD}} = 2.85 \sim 3.6 \text{ V}$)
30.5 μS （副时钟(f_{XT}) = 32.768 kHz）
- 通用寄存器： 32 位×32 寄存器
- CPU 特性： 带符号乘法（16X16→32）： 1 到 2 个时钟周期。
带符号乘法（32X32→64）： 1 到 5 个时钟周期。
溢出处理（包括上、下溢出检测功能）。
32 位移位指令： 1 个时钟
位操作指令
长/短两种格式的数据装载/存储指令。
- 存储空间： 64 MB 线性地址空间（存储程序和数据）
外部扩展： 最多 16 MB（包括用作内部 ROM/RAM 空间的 1 MB 区域）
 - 内部存储器 RAM： 40/48/56 KB（参见表 1-1 和表 1-2）
Flash 存储器： 256/384/512 KB（参见表 1-1 和表 1-2）
 - 外部总线接口： 可选择的独立总线/复用总线输出
（在 V850ES/JG3-H 中只存在一个复用总线）
8 位/16 位数据总线宽度
等待功能
 - 可编程等待功能
 - 外部等待功能
 空闲状态功能
总线保持功能
- 中断和异常处理：

		内部			外部		
		不可屏蔽中断	可屏蔽中断	总数	不可屏蔽中断	可屏蔽中断	总数
V850ES/JG3-H	$\mu\text{PD70F3760}$	1	68	69	1	16	17
	$\mu\text{PD70F3761}$	1	68	69	1	16	17
	$\mu\text{PD70F3762}$	1	68	69	1	16	17
	$\mu\text{PD70F3770}$	1	72	73	1	16	17
V850ES/JH3-H	$\mu\text{PD70F3765}$	1	68	69	1	19	20
	$\mu\text{PD70F3766}$	1	68	69	1	19	20
	$\mu\text{PD70F3767}$	1	68	69	1	19	20
	$\mu\text{PD70F3771}$	1	72	73	1	19	20

软件异常： 32 个产生源
异常陷阱： 2 个产生源

- I/O 线： I/O 端口： 77（V850ES/JG3-H）
96（V850ES/JH3-H）
- 定时器功能： 16-位间隔定时器 M（TMM）： 4 个通道。
16-位定时器/事件计数器 AA（TAA）： 6 个通道。
16-位定时器/事件计数器 AB（TAB）： 2 个通道。
16-位定时器/事件计数器 T（TMT）： 1 个通道。
马达控制功能（使用定时器：TAB1、TAA4）
带有死区时间的 16 位精度 6 相位 PWM 功能：
高阻抗输出控制功能。
定时器谐波生成 A/D 触发的功能

	任意周期设置功能。
	任意死区时间设置功能。
	实时控制器 (RTC) : 1 个通道。
	看门狗定时器: 1 个通道。
○ 实时输出端:	6 位×1 通道
○ 串行接口:	异步串行接口 C (UARTC) 3-线长度可变串行接口 F (CSIF) I ² C 总线接口 (I ² C) CAN 接口 USB 功能接口
	UARTC/CSIF: 2 个通道
	UARTC/CSIF/I ² C: 1 个通道
	UARTC/I ² C [※] : 2 个通道
	CSIF: 2 个通道
	USB 功能: 1 个通道
	 注 在μPD70F3770 和 70F3771 中, 与 CAN 共享一个通道。
○ A/D 转换器:	10-位分辨率: 12 个通道
○ D/A 转换器:	8-位分辨率: 2 个通道
○ DMA 控制器:	4 个通道
○ DCU (调试控制单元):	JTAG 接口
○ 时钟发生器:	主时钟或副时钟: 7 个等级 CPU 时钟 (f _{xx} , f _{xx} /2, f _{xx} /4, f _{xx} /8, f _{xx} /16, f _{xx} /32, f _{XT}) 可选择时钟直通模式/PLL 模式
○ 内部振荡器时钟:	220 kHz (TYP.)
○ 节电功能:	HALT/IDLE1/IDLE2/STOP/副时钟/sub-IDLE 模式
○ 封装:	100-引脚 LQFP 塑料封装 (密间距) (14× 14) (V850ES/JG3-H) 128-引脚 LQFP 塑料封装 (密间距) (14× 20) (V850ES/JH3-H)

1.3 应用领域

需要 USB 接口的设备，如家庭音响系统、打印机和扫描仪。

1.4 订货信息

• V850ES/JG3-H

型号	封装	内部 Flash 存储器
μPD70F3760GC-UEU-AX	100-引脚 LQFP 塑料封装（密间距）（14 × 14）	256 KB
μPD70F3761GC-UEU-AX	100-引脚 LQFP 塑料封装（密间距）（14 × 14）	384 KB
μPD70F3762GC-UEU-AX	100-引脚 LQFP 塑料封装（密间距）（14 × 14）	512 KB
μPD70F3770GC-UEU-AX	100-引脚 LQFP 塑料封装（密间距）（14 × 14）	256 KB

• V850ES/JH3-H

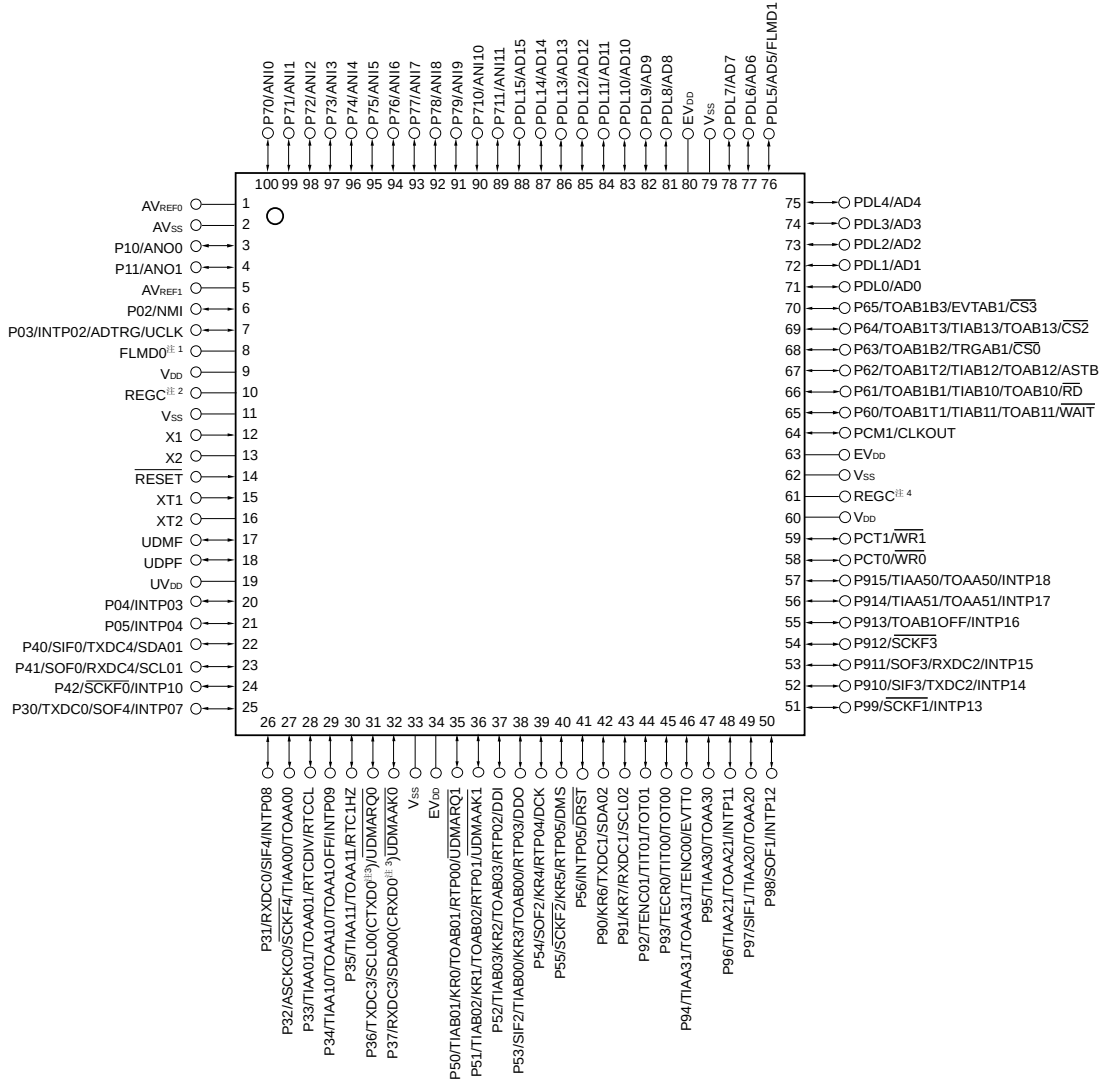
型号	封装	内部 Flash 存储器
μPD70F3765GF-GAT-AX	128-引脚 LQFP 塑料封装（密间距）（14 × 20）	256 KB
μPD70F3766GF-GAT-AX	128-引脚 LQFP 塑料封装（密间距）（14 × 20）	384 KB
μPD70F3767GF-GAT-AX	128-引脚 LQFP 塑料封装（密间距）（14 × 20）	512 KB
μPD70F3771GC-GAT-AX	100-引脚 LQFP 塑料封装（密间距）（14 × 20）	256 KB

备注 V850ES/JG3-H 和 V850ES/JH3-H 都是无铅产品。

1.5 引脚配置 (顶视图)

• V850ES/JH3-H

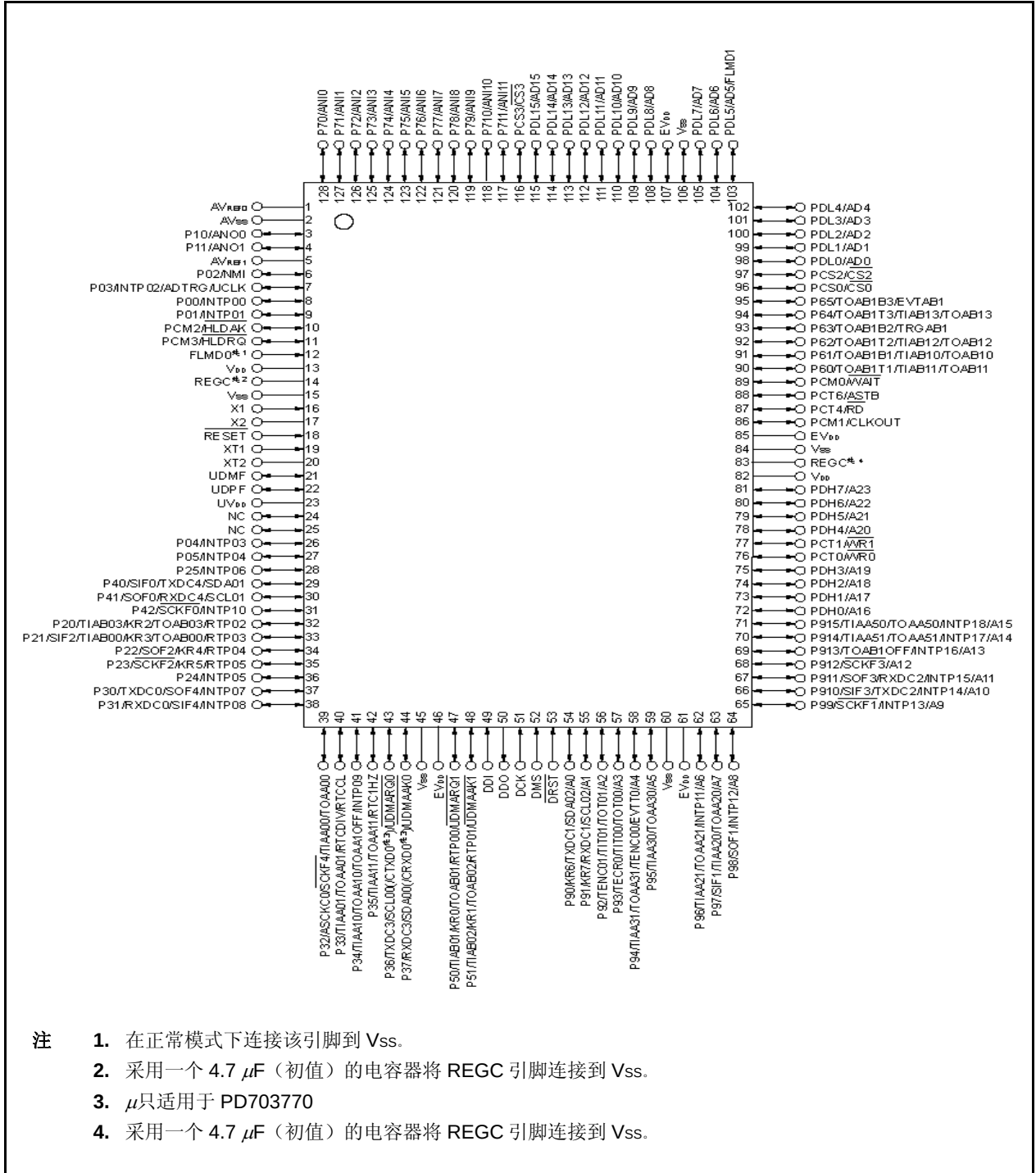
100-引脚 LQFP 塑料封装 (密间距) (14 × 14)

 μ PD70F3760GC-UEU-AX μ PD70F3761GC-UEU-AX μ PD70F3762GC-UEU-AX μ PD70F3770GC-UEU-AX

- 注
1. 在正常模式下连接该引脚到 Vss。
 2. 采用一个 4.7 μ F (初值) 的电容器将 REGC 引脚连接到 Vss。
 3. 仅 μ PD703770
 4. 采用一个 4.7 μ F (初值) 的电容器将 REGC 引脚连接到 Vss。

• V850ES/JH3-H

128-引脚 LQFP 塑料封装（密间距）（14 × 20）

 μ PD70F3765GF-GAT-AX μ PD70F3766GF-GAT-AX μ PD70F3767GF-GAT-AX μ PD70F3771GF-GAT-AX

- 注
1. 在正常模式下连接该引脚到 V_{SS}。
 2. 采用一个 4.7 μ F（初值）的电容器将 REGC 引脚连接到 V_{SS}。
 3. μ 只适用于 PD703770
 4. 采用一个 4.7 μ F（初值）的电容器将 REGC 引脚连接到 V_{SS}。

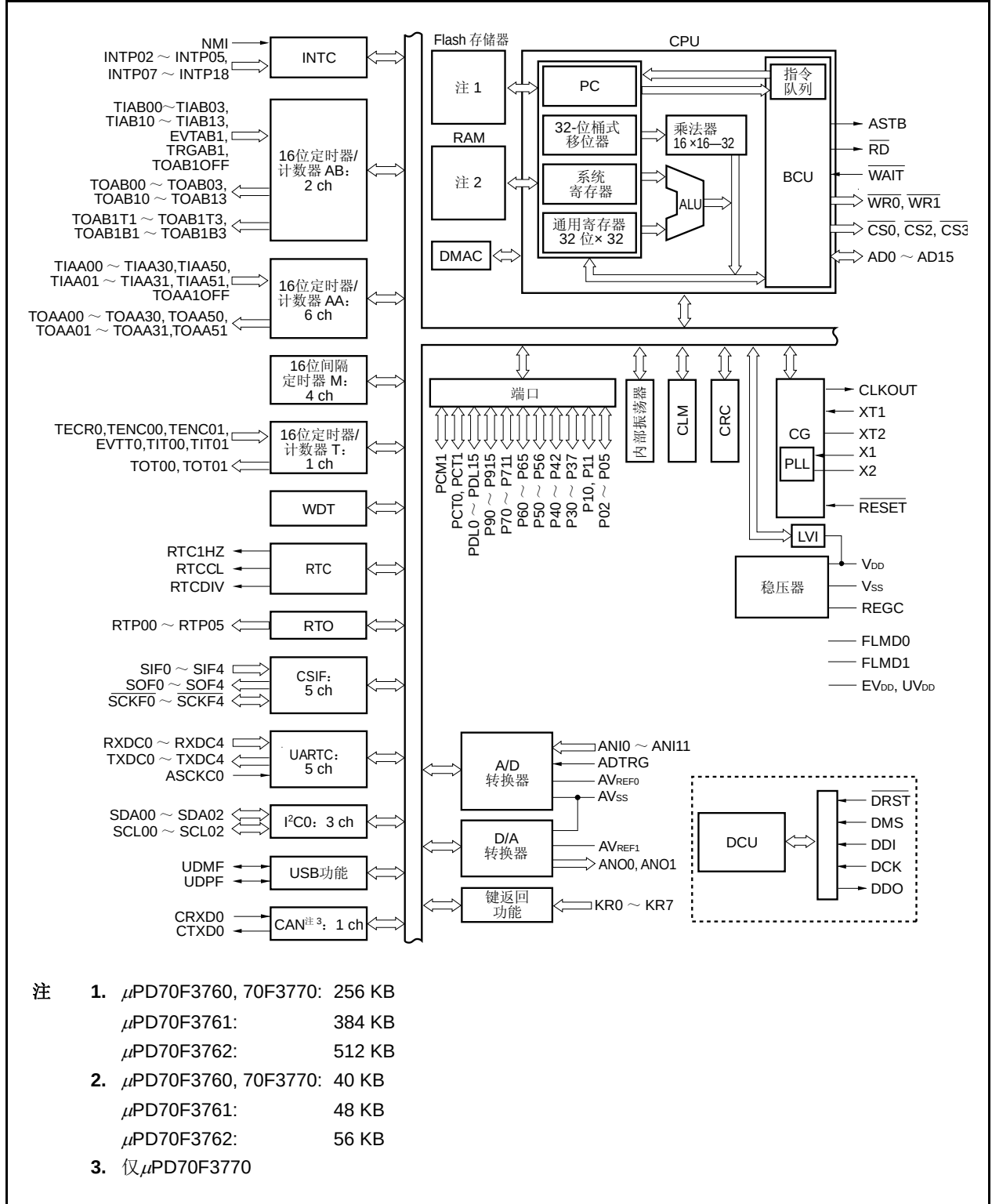
引脚名称

A0~A23:	地址总线	RTP00~RTP05:	实时输出端口
AD0~AD15:	地址/数据总线	RXDC0~RXDC4:	接收数据
ADTRG:	A/D 触发输入	$\overline{\text{SCKF0}}\sim\overline{\text{SCKF4}}:$	串行时钟
ANI0~ANI11:	模拟输入	SCL00~SCL02:	串行时钟
ANO0、ANO1:	模拟输出	SDA00~SDA02:	串行数据
ASCKC0:	异步串行时钟	SIF0~SIB4:	串行输入
ASTB:	地址选通	SOF0~SOF4:	串行输出
AVREF0、AVREF1:	模拟参考电压	TECR0:	定时编码器清零输入
AVss:	模拟引脚接地端	TENC00、TENC01:	定时编码器输入
CLKOUT:	时钟输出	TIAA00、TIAA01,	
CRXD0:	CAN 接收数据	TIAA10、TIAA11,	
$\overline{\text{CS0}}、\overline{\text{CS2}}、\overline{\text{CS3}}:$	片选	TIAA20、TIAA21、	
CTXD0:	CAN 发送数据	TIAA30、TIAA31、	
DCK:	调试时钟	TIAA50、TIAA51、	
DDI:	调试数据输入	TIAB00~TIAB03,	
DDO:	调试数据输出	TIAB10、TIAB13、	
DMS:	调试模式选择	TIT00、TIT01:	定时器输入
$\overline{\text{DRST}}:$	调试复位	TOAA00、TOAA01、	
EVDD:	外部引脚的供电电源	TOAA10、TOAA11、	
EVT0、EVTAB1:	定时器事件计数输入	TOAA20、TOAA21、	
FLMD0、FLMD1:	Flash 编程模式	TOAA30、TOAA31,	
HLDK:	保持应答	TOAA50~TOAA51,	
HLDKQ:	保持请求	TOAB00~TOAB03,	
INTP00~INTP18:	外部中断输入	TOAB10~TOAB13,	
KR0~KR7:	按键返回	TOAB1B1~TOAB1B3,	
NMI:	不可屏蔽中断请求	TOAB1T1~TOAB1T3,	
NC:	不连接	TOT00、TOT01:	定时器输出
P00~P05:	端口 0	TOAA1OFF,	
P10、P11:	端口 1	TOAB1OFF:	定时器输出关闭
P20~P25:	端口 2	TRGAB1:	定时器触发输入
P30~P37:	端口 3	TXDC0~TXDC4:	发送数据
P40~P42:	端口 4	UCLK:	USB 时钟
P50~P56:	端口 5	$\overline{\text{UDMAAK0}},$	
P60~P65:	端口 6	$\overline{\text{UDMAAK1}}:$	外部 USB 的 DMA 响应
P70~P711:	端口 7	$\overline{\text{UDMARQ0}},$	
P90~P915:	端口 9	$\overline{\text{UDMARQ1}}:$	外部 USB 的 DMA 请求
PCM0~PCM3:	端口 CM	UDMF:	USB 数据 I/O (-) 功能
PCS0,PCS2,PCS3:	端口 CS	UDPF:	USB 数据 I/O (+) 功能
PCT0、PCT1,	端口 CT	UVDD:	外部 USB 的供电电源
PCT4、PCT6:		VDD:	供电电源
PDH0~PDH7:	端口 DH	Vss:	接地
PDL0~PDL15:	端口 DL	$\overline{\text{WAIT}}:$	外部等待输入
$\overline{\text{RD}}:$	读选通	$\overline{\text{WR0}}:$	低字节写选通
REGC:	稳压器控制	$\overline{\text{WR1}}:$	高字节写选通
$\overline{\text{RESET}}:$	复位	X1、X2:	主时钟晶振
RTC1HZ、RTCCL,		XT1、XT2:	副时钟晶振
RTCDIV:	实时计数器时钟输出		

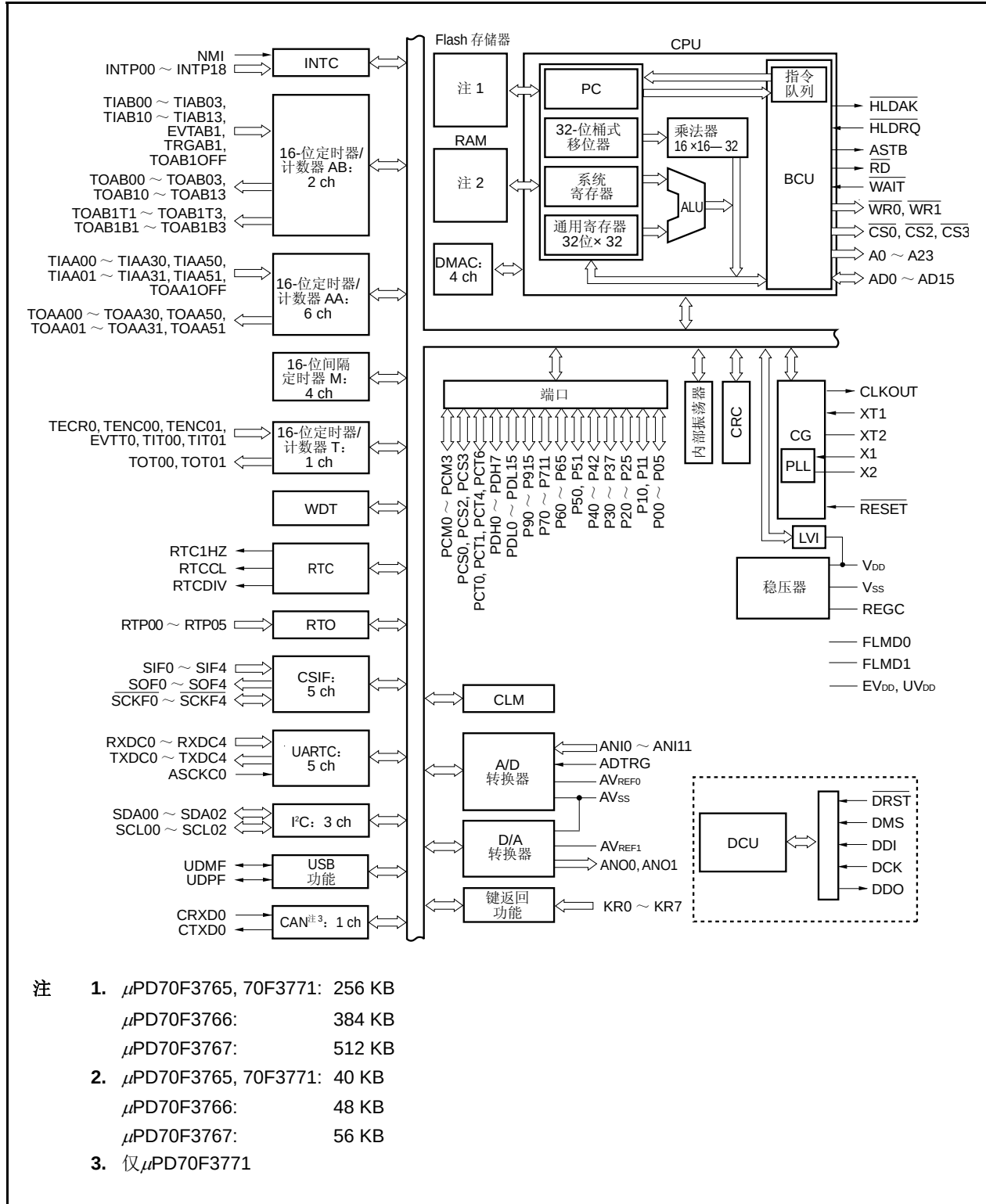
1.6 功能块结构

1.6.1 内部框图

• V850ES/JG3-H



• V850ES/JH3-H



1.6.2 内部组件

(1) CPU

CPU 使用五级流水线控制，从而实现单时钟周期内的地址运算、算术逻辑运算、数据传输以及几乎所有其它指令处理。

其它片内专用硬件，诸如乘法器（16 位×16 位→32 位）和桶式移位器（32 位），协助加速处理复杂操作。

(2) 总线控制单元（BCU）

总线控制单元根据从 CPU 获取的物理地址，启动所需的外总线处理周期。当从外部存储空间取指且 CPU 不发送总线周期开始请求时，BCU 将产生预取地址并预取指令代码。预取指令代码存储在指令队列中。

(3) Flash 存储器（ROM）

这是一个 512/384/256 KB 的 flash 存储器，其映射地址为 0000000H 至 007FFFFH/0000000H 至 005FFFFH/0000000H 至 003FFFFH。取指令期间，CPU 可在一个时钟周期内访问 Flash 存储器（ROM）。

(4) RAM

这是一个 48/40/32 KB 的 RAM，其映射地址为 3FF3000H 至 3FF5000H/3FF5000H 至 3FF7000H/3FF7000H 至 3FF7000H。数据访问期间，CPU 可在一个时钟周期内访问 RAM。将 8 KB 的数据 RAM 整合在地址 00280000H 至 002FFFFFH。

(5) 中断控制器（INTC）

该控制器用来处理自片上外围硬件和外部硬件的硬件中断请求（NMI、INTP0 至 INTP18）。可以对这些中断请求指定 8 个级别的中断优先级，同时也能执行多重服务控制。

(6) 时钟发生器（CG）

提供主时钟振荡器和副时钟振荡器，并且产生相应的主时钟振荡频率（fx）和副时钟振荡频率（fxT）。有两种模式：在时钟直通模式中，fx 用作主时钟频率（fxx）。在 PLL 模式下，使用 fx 的 8 倍频。

CPU 时钟频率（fCPU）可以有以下选择：fxx，fxx/2，fxx/4，fxx/8，fxx/16，fxx/32，和 fxT。

(7) 内部振荡器

芯片提供内部振荡器。振荡频率为 220 kHz（TYP）。内部振荡器为看门狗定时器 2 和定时器 M 提供时钟。

(8) 定时器/计数器

芯片提供了 6 个通道的 16 位定时器/事件计数器 AA（TAA），2 个通道的 16 位定时器/事件计数器 AB（TAB）和 1 个通道的 16 位定时器/事件计数器 T（TMT）和 4 个通道的 16 位间隔定时器 M（TMM）。结合使用 TAB1 和 TAA4 可以实现马达控制功能。

(9) 实时控制器（有关钟表定时器）：

实时计数器为基于副时钟（32.768 kHz）或主时钟的钟表计数器计算参考时间（1 秒）。同时也能将其用于基于主时钟的间隔定时器。提供了专用于年、月、星期、日、时、分和秒的硬件计数器，最多能计算 99 年。

(10) 看门狗定时器 2

片内集成的看门狗定时器用于检测无意中形成的程序死循环、系统异常等。

内部振荡时钟、主时钟、或副时钟能够选择为源时钟。

溢出发生后，看门狗定时器 2 产生一个不可屏蔽中断请求信号（INTWDT2）或系统复位信号（WDT2RES）。

(11) 串行接口

V850ES/JG3-H 和 V850ES/JH3-H 包括三种串行接口（异步串行接口 C（UARTC）、3 线长度可变串行接口 F（CSIF）和 I²C 总线接口（I²C）），一个 CAN 控制器（CAN）^注和一个 USB 功能控制器（USBF）。

UARTC 通过 TXDC0 至 TXDC2 引脚和 RXDC0 至 RXDC2 引脚传输数据。

CSIF 通过 SOF0 至 SOF4 引脚，SIF0 至 SIF4 引脚， $\overline{\text{SCKF0}}$ 至 $\overline{\text{SCKF4}}$ 引脚传输数据。

在 I²C 情况下，数据通过 SDA00 至 SDA02 引脚和 SCL00 至 SCL02 引脚传输数据。

CAN^注通过 CRXD0^注和 CTXD0^注引脚传输数据。

USBF 通过 UDMF 和 UDPF 引脚传输数据。

注 只适用于 μ PD70F3770, 70F3771

(12) A/D 转换器

10 位 A/D 转换器包含 12 个模拟输入引脚。使用逐次逼近的方式执行转换。

(13) D/A 转换器

片内集成了一个两通道的 8 位分辨率 D/A 转换器，采用 R-2R 阶梯变换模式。

(14) DMA 控制器

芯片提供了一个 4 通道的 DMA 转换器。为响应由片上外围 I/O 设备发出的中断请求，该控制器在内部 RAM、片上外围 I/O 设备和外部存储器之间进行数据传输。

(15) 按键中断功能

在键输入引脚（8 个通道）输入下降沿，可以产生按键中断请求信号（INTKR）。

(16) 实时输出功能

当定时器比较寄存器与信号匹配时，实时输出功能将预置的 6 位数据传送至输出锁存器。

(17) CRC 功能

芯片提供了 CRC 操作电路，该电路根据 8 位数据的设置产生 16 位 CRC（循环冗余检测）代码。

(18) DCU（调试控制单元）

提供使用 JTAG（联合测试行动小组）通讯规范的片上调试功能。通过控制引脚输入电平和 OCDM 寄存器来进行正常端口功能和片内调试功能之间的切换。

(19) 端口

有以下通用寄存器端口功能和控制引脚功能可用。

• V850ES/JG3-H

端口	I/O	复用功能
P0	4-位 I/O	NMI、外部中断、A/D 转换触发器、串行接口
P1	2-位 I/O	D/A 转换器模拟输出
P3	10-位 I/O	外部中断、实时计数器、串行接口、定时器 I/O
P4	3-位 I/O	串行接口、外部中断
P5	7-位 I/O	定时器 I/O、串行接口、实时输出、按键中断输入、调试 I/O
P6	6-位 I/O	外部控制信号、定时器 I/O、外部控制信号
P7	12-位 I/O	A/D 转换器模拟输入
P9	16-位 I/O	串行接口、按键中断输入、定时器 I/O、外部中断
PCM	1-位 I/O	外部控制信号
PCT	2-位 I/O	外部控制信号
PDL	16-位 I/O	外部地址/数据总线

• V850ES/JH3-H

端口	I/O	复用功能
P0	6-位 I/O	NMI、外部中断、A/D 转换触发器、串行接口
P1	2-位 I/O	D/A 转换器模拟输出
P2	6-位 I/O	定时器 I/O、实时输出、按键中断输入、串行接口
P3	10-位 I/O	外部中断、实时计数器、串行接口、定时器 I/O
P4	3-位 I/O	串行接口、外部中断
P5	2-位 I/O	定时器 I/O、实时输出、按键中断输入
P6	6-位 I/O	外部控制信号、定时器 I/O
P7	12-位 I/O	A/D 转换器模拟输入
P9	16-位 I/O	外部地址总线、串行接口、按键中断输入、定时器 I/O、外部中断
PCM	4-位 I/O	外部控制信号
PCS	3-位 I/O	外部控制信号
PCT	4-位 I/O	外部控制信号
PDH	8-位 I/O	外部地址总线
PDL	16-位 I/O	外部地址/数据总线

第二章 引脚功能

2.1 引脚功能列表

下表描述了 V850ES/JG3-H 和 V850ES/JH3-H 单片机中引脚的名称和功能。
有四种引脚 I/O 缓冲器供电电源：AV_{REF0}、AV_{REF1}、EV_{DD} 和 UV_{DD}。供电电源和引脚关系如下：

表 2-1. 引脚 I/O 缓冲器供电电源

供电电源	对应引脚	
	V850ES/JG3-H	V850ES/JH3-H
AV _{REF0}	端口 7	端口 7
AV _{REF1}	端口 1	端口 1
EV _{DD}	$\overline{\text{RESET}}$ 、端口 0、3 至 6、9、CM、CT、DL	$\overline{\text{RESET}}$ 、端口 0、2 至 6、9、CM、CS、CT、DH、DL
UV _{DD}	UDPF、UDMF	UDPF、UDMF

(1) 端口引脚

(1/4)

引脚名称	I/O	功能	复用功能	引脚编号	
				JG3-H	JH3-H
P00	I/O	端口 0 6-位 I/O 端口 (V850ES/JH3-H) 4-位 I/O 端口 (V850ES/JG3-H) 按位单元指定输入/输出。	INTP00	—	8
P01			INTP01	—	9
P02			NMI	6	6
P03			INTP02/ADTRG/UCLK	7	7
P04			INTP03	20	26
P05			INTP04	21	27
P10	I/O	端口 1 2-位 I/O 端口 按位单元指定输入/输出。	ANO0	3	3
P11			ANO1	4	4
P20	I/O	端口 2 6-位 I/O 端口 按位单元指定输入/输出。	TIAB03/KR2/TOAB03/RTP02	—	32
P21			SIF2/TIAB00/KR3/TOAB00/RTP03	—	33
P22			SOF2/KR4/RTP04	—	34
P23			SCKF2/KR5/RTP05	—	35
P24			INTP05	—	36
P25			INTP06	—	28
P30	I/O	端口 3 8-位 I/O 端口 按位单元指定输入/输出。	TXDC0/SOF4/INTP07	25	37
P31			RXDC0/SIF4/INTP08	26	38
P32			ASCKC0/SCKF4/TIAA00/TOAA00	27	39
P33			TIAA01/TOAA01/RTCDIV/RTCCL	28	40
P34			TIAA10/TOAA10/TOAA1OFF/INTP09	29	41
P35			TIAA11/TOAA11/RTC1HZ	30	42
P36			TXDC3/SCL00/CTXD0 ^{*/} /UDMARQ0	31	43
P37			RXDC3/SDA00/CRXD0 ^{*/} /UDMAAK0	32	44
P40	I/O	端口 4 3-位 I/O 端口 按位单元指定输入/输出。	SIF0/TXDC4/SDA01	22	29
P41			SOF0/RXDC4/SCL01	23	30
P42			SCKF0/INTP10	24	31
P50	I/O	端口 5 2-位 I/O 端口 (V850ES/JH3-H) 7-位 I/O 端口 (V850ES/JG3-H) 按位单元指定输入/输出。	TIAB01/KR0/TOAB01/RTP00 /UDMARQ1	35	47
P51			TIAB02/KR1/TOAB02/RTP01 /UDMAAK1	36	48
P52			TIAB03/KR2/TOAB03/RTP02/DDI	37	—
P53			SIF2/TIAB00/KR3/TOAB00 /RTP03/DDO	38	—
P54			SOF2/KR4/RTP04/DCK	39	—
P55			SCKF2/KR5/RTP05/DMS	40	—
P56			INTP05/DRST	41	—

注 仅适用于 μ PD70F3770、70F3771

备注 JG3-H: V850ES/JG3-H, JH3-H: V850ES/JH3-H

(2/4)

引脚名称	I/O	功能	复用功能	引脚编号	
				JG3-H	JH3-H
P60	I/O	端口 6 6-位 I/O 端口 按位单元指定输入/输出。	TOAB1T1/TOAB11/TIAB11/ $\overline{\text{WAIT}}$	65	—
			TOAB1T1/TOAB11/TIAB11	—	90
P61			TOAB1B1/TIAB10/TOAB10/ $\overline{\text{RD}}$	66	—
			TOAB1B1/TIAB10/TOAB10	—	91
P62			TOAB1T2/TOAB12/TIAB12/ASTB	67	—
			TOAB1T2/TOAB12/TIAB12	—	92
P63			TOAB1B2/TRGAB1/ $\overline{\text{CS0}}$	68	—
			TOAB1B2/TRGAB1	—	93
P64	I/O	端口 7 12-位 I/O 端口 按位单元指定输入/输出。	TOAB1T3/TOAB13/TIAB13/ $\overline{\text{CS2}}$	69	—
			TOAB1T3/TOAB13/TIAB13	—	94
P65			TOAB1B3/EVTAB1/ $\overline{\text{CS3}}$	70	—
			TOAB1B3/EVTAB1	—	95
P70			ANI0	100	128
P71			ANI1	99	127
P72			ANI2	98	126
P73			ANI3	97	125
P74	I/O	端口 9 16-位 I/O 端口 按位单元指定输入/输出。	ANI4	96	124
P75			ANI5	95	123
P76			ANI6	94	122
P77			ANI7	93	121
P78			ANI8	92	120
P79			ANI9	91	119
P710			ANI10	90	118
P711			ANI11	89	117
P90	I/O	端口 9 16-位 I/O 端口 按位单元指定输入/输出。	KR6/TXDC1/SDA02	42	—
			KR6/TXDC1/SDA02/A0	—	54
P91			KR7/RXDC1/SCL02	43	—
			KR7/RXDC1/SCL02/A1	—	55
P92			TENC01/TIT01/TOT01	44	—
			TENC01/TIT01/TOT01/A2	—	56
P93			TECR0/TIT00/TOT00	45	—
			TECR0/TIT00/TOT00/A3	—	57
P94			TIAA31/TOAA31/TENC00/EVTT0	46	—
			TIAA31/TOAA31/TENC00/EVTT0/A4	—	58
P95			TIAA30/TOAA30	47	—
			TIAA30/TOAA30/A5	—	59
P96			TIAA21/TOAA21/INTP11	48	—
			TIAA21/TOAA21/INTP11/A6	—	62
P97			SIF1/TIAA20/TOAA20	49	—
			SIF1/TIAA20/TOAA20/A7	—	63

备注 JG3-H: V850ES/JG3-H, JH3-H: V850ES/JH3-H

(3/4)

引脚名称	I/O	功能	复用功能	引脚编号	
				JG3-H	JH3-H
P98	I/O	端口 9 16-位 I/O 端口 按位单元指定输入/输出。	SOF1/INTP12	50	—
			SOF1/INTP12/A8	—	64
P99			SCKF1/INTP13	51	—
			SCKF1/INTP13/A9	—	65
P910			SIF3/TXDC2/INTP14	52	—
			SIF3/TXDC2/INTP14/A10	—	66
P911			SOF3/RXDC2/INTP15	53	—
			SOF3/RXDC2/INTP15/A11	—	67
P912			SCKF3	54	—
			SCKF3/A12	—	68
P913			TOAB1OFF/INTP16	55	—
			TOAB1OFF/INTP16/A13	—	69
P914			TIAA51/TOAA51/INTP17	56	—
			TIAA51/TOAA51/INTP17/A14	—	70
P915			TIAA50/TOAA50/INTP18	57	—
			TIAA50/TOAA50/INTP18/A15	—	71
PCM0	I/O	端口 CM 4-位 I/O 端口 (V850ES/JH3-H) 1-位 I/O 端口 (V850ES/JG3-H) 按位单元指定输入/输出。	WAIT	—	89
PCM1			CLKOUT	64	86
PCM2			HLD $\overline{\text{AK}}$	—	10
PCM3			HLD $\overline{\text{RQ}}$	—	11
PCS0	I/O	端口 CS 3-位 I/O 端口 按位单元指定输入/输出。	CS0	—	96
PCS2			CS2	—	97
PCS3			CS3	—	116
PCT0	I/O	端口 CT 4-位 I/O 端口 (V850ES/JH3-H) 2-位 I/O 端口 (V850ES/JG3-H) 按位单元指定输入/输出。	WR0	58	76
PCT1			WR1	59	77
PCT4			RD	—	87
PCT6			ASTB	—	88
PDH0	I/O	端口 DH 8-位 I/O 端口 按位单元指定输入/输出。	A16	—	72
PDH1			A17	—	73
PDH2			A18	—	74
PDH3			A19	—	75
PDH4			A20	—	78
PDH5			A21	—	79
PDH6			A22	—	80
PDH7			A23	—	81

备注 JG3-H: V850ES/JG3-H, JH3-H: V850ES/JH3-H

(4/4)

引脚名称	I/O	功能	复用功能	引脚编号	
				JG3-H	JH3-H
PDL0	I/O	端口 DL 16-位 I/O 端口 按位单元指定输入/输出。	AD0	71	98
PDL1			AD1	72	99
PDL2			AD2	73	100
PDL3			AD3	74	101
PDL4			AD4	75	102
PDL5			AD5/FLMD1	76	103
PDL6			AD6	77	104
PDL7			AD7	78	105
PDL8			AD8	81	108
PDL9			AD9	82	109
PDL10			AD10	83	110
PDL11			AD11	84	111
PDL12			AD12	85	112
PDL13			AD13	86	113
PDL14			AD14	87	114
PDL15			AD15	88	115

备注 JG3-H: V850ES/JG3-H, JH3-H: V850ES/JH3-H

(2) 非端口引脚

(1/9)

引脚名称	I/O	功能	复用功能	引脚编号	
				JG3-H	JH3-H
A0	输出	用于外部存储器的地址总线 (使用独立总线时)	P90/KR6/TXDC1/SDA02	—	54
A1			P91/KR7/RXDC1/SCL02	—	55
A2			P92/TENC01/TIT01/TOT01	—	56
A3			P93/TECR00/TIT00/TOT00	—	57
A4			P94/TIAA31/TOAA31/TENC0/EVTT0	—	58
A5			P95/TIAA30/TOAA30	—	59
A6			P96/TIAA21/TOAA21/INTP11	—	62
A7			P97/SIF1/TIAA20/TOAA20	—	63
A8			P98/SOF1/INTP12	—	64
A9			P99/SCKF1/INTP13	—	65
A10			P910/SIF3/TXDC2/INTP14	—	66
A11			P911/SOF3/RXDC2/INTP15	—	67
A12			P912/ $\overline{\text{SCKF3}}$	—	68
A13			P913/TOAB1OFF/INTP16	—	69
A14			P914/TIAA51/TOAA51/INTP17	—	70
A15			P915/TIAA50/TOAA50/INTP18	—	71
A16			PDH0	—	72
A17			PDH1	—	73
A18			PDH2	—	74
A19			PDH3	—	75
A20			PDH4	—	78
A21			PDH5	—	79
A22			PDH6	—	80
A23			PDH7	—	81
AD0	I/O	外部存储器地址/数据总线	PDL0	71	98
AD1			PDL1	72	99
AD2			PDL2	73	100
AD3			PDL3	74	101
AD4			PDL4	75	102
AD5			PDL5/FLMD1	76	103
AD6			PDL6	77	104
AD7			PDL7	78	105
AD8			PDL8	81	108
AD9			PDL9	82	109
AD10			PDL10	83	110
AD11			PDL11	84	111
AD12			PDL12	85	112
AD13			PDL13	86	113
AD14			PDL14	87	114
AD15			PDL15	88	115
ADTRG	输入	A/D 转换器的外部触发器输入	P03/INTP02/UCLK	7	7

备注 JG3-H: V850ES/JG3-H, JH3-H: V850ES/JH3-H

(2/9)

引脚名称	I/O	功能	复用功能	引脚编号	
				JG3-H	JH3-H
ANI0	输入	用于 A/D 转换器的模拟电压输入	P70	100	128
ANI1			P71	99	127
ANI2			P72	98	126
ANI3			P73	97	125
ANI4			P74	96	124
ANI5			P75	95	123
ANI6			P76	94	122
ANI7			P77	93	121
ANI8			P78	92	120
ANI9			P79	91	119
ANI10			P710	90	118
ANI11			P711	89	117
ANO0	输出	用于 D/A 转换器的模拟电压输出	P10	3	3
ANO1			P11	4	4
ASCKC0	输入	UARTC0 波特率时钟输入	P32/SCKF4/TIAA00/TOAA00	27	39
ASTB	输出	用于外部存储器的地址选通信号	P62/TOAB1T2/TIAB12/TOAB12	67	—
			PCT6	—	88
AV _{REF0}	—	用于 A/D 转换器的参考电压输入/用于端口 7 的正极电源供应	—	1	1
AV _{REF1}	—	用于 D/A 转换器的参考电压输入/用于端口 1 的正极电源供应	—	5	5
AV _{SS}	—	用于 A/D 和 D/A 转换器的接地电平	—	2	2
CLKOUT	输出	内部系统时钟输出	PCM1	64	86
CRXD0 [※]	输入	CAN 接收数据输入	P37/RXDC3/SDA00/UDMAAK0	32	44
CS0	输出	片选输出	P63/TOAB1B2/TRGAB1	68	—
			PCS0	—	96
CS2	输出	片选输出	P64/TOAB1T3/TIAB13/TOAB13	69	—
			PCS2	—	97
CS3	输出	片选输出	P65/TOAB1B3/EVTAB1	70	—
			PCS3	—	116
CTXD0 [※]	输出	CAN 发送数据输出	P36/TXDC3/SCL00/UDMARQ0	31	43
DCK	输入	用于片上调试的时钟输入	P54/SOF2/KR4/RTP04	39	—
			—	—	51
DDI	输入	用于片上调试的数据输入	P52/TIAB03/KR2/TOAB03/RTP02	37	—
			—	—	49
DDO	输出	用于片上调试的数据输出 在片上调试模式中，强制设置成高电平输出。	P53/SIF2/TIAB00/KR3/TOAB00/RTP03	38	—
			—	—	50
DMS	输入	用于片上调试的模式选择信号输入	P55/SCKF2/KR5/RTP05	40	—
			—	—	52
DRST	输入	用于片上调试的复位信号输入	P56/INTP05	41	—
			—	—	53

注 仅适用于μPD70F3770、70F3771

备注 JG3-H: V850ES/JG3-H, JH3-H: V850ES/JH3-H

(3/9)

引脚名称	I/O	功能	复用功能	引脚编号	
				JG3-H	JH3-H
EV _{DD}	—	用于外部设备的正极供电电源（与 V _{DD} 电位相同）	—	34、63、80	46、61、85、107
EVT _{T0}	输入	TMT0 的外部事件计数输入	P94/TIAA31/TOAA31/TENC00 P94/TIAA31/TOAA31/TENC00/A4	46 —	— 58
EVTAB1	输入	TAB1 的外部事件计数输入	P65/TOAB1B3/ $\overline{\text{CS3}}$ P65/TOAB1B3	70 —	— 95
FLMD0	输入	Flash 存储器编程模式设置引脚	—	8	12
FLMD1	输入		PDL5/AD5	76	103
H $\overline{\text{LDAK}}$	输出	总线保持响应输出	PCM2	—	10
H $\overline{\text{LDRQ}}$	输入	总线保持请求输入	PCM3	—	11
INTP00	输入	外部中断请求输入 （可屏蔽的、模拟噪声消除） 为 INTP02 引脚选择模拟噪声消除或数字噪声消除。	P00	—	8
INTP01			P01	—	9
INTP02			P03/ADTRG/UCLK	7	7
INTP03			P04	20	26
INTP04			P05	21	27
INTP05			P56/ $\overline{\text{DRST}}$ P24	41 —	— 36
INTP06			P25	—	28
INTP07			P30/TXDC0/SOF4	25	37
INTP08			P31/RXDC0/SIF4	26	38
INTP09			P34/TIAA10/TOAA10/TOAA1OFF	29	41
INTP10			P42/ $\overline{\text{SCKF0}}$	24	31
INTP11			P96/TIAA21/TOAA21/A6 P96/TIAA21/TOAA21	— 48	62 —
INTP12			P98/SOF1 P98/SOF1/A8	50 —	— 64
INTP13			P99/ $\overline{\text{SCKF1}}$ P99/SCKF1/A9	51 —	— 65
INTP14			P910/SIF3/TXDC2 P910/SIF3/TXDC2/A10	52 —	— 66
INTP15			P911/SOF3/RXDC2 P911/SOF3/RXDC2/A11	53 —	— 67
INTP16			P913/TOAB1OFF P913/TOAB1OFF/A13	55 —	— 69
INTP17			P914/TIAA51/TOAA51 P914/TIAA51/TOAA51/A14	56 —	— 70
INTP18			P915/TIAA50/TOAA50 P915/TIAA50/TOAA50/A15	57 —	— 71

备注 JG3-H: V850ES/JG3-H, JH3-H: V850ES/JH3-H

(4/9)

引脚名称	I/O	功能	复用功能	引脚编号			
				JG3-H	JH3-H		
KR0	输入	按键中断输入（片上模拟噪声消除器）。	P50/TIAB01/TOAB01/RTP00 /UDMARQ1	35	47		
KR1			P51/TIAB02/TOAB02/RTP01 /UDMARQ1	36	48		
KR2			P52/TIAB03/TOAB03/RTP02/DDI P20/TIAB03/TOAB03/RTP02	37 —	— 32		
KR3			P53/SIF2/TIAB00/TOAB00/RTP03/DDO P21/SIF2/TIAB00/TOAB00/RTP03	38 —	— 33		
KR4			P54/SOF2/RTP04/DCK P22/SOF2/RTP04	39 —	— 34		
KR5			P55/SCKF2/RTP05/DMS P23/SCKF2/RTP05	40 —	— 35		
KR6			P90/TXDC1/SDA02 P90/TXDC1/SDA02/A0	42 —	— 54		
KR7			P91/RXDC1/SCL02 P91/RXDC1/SCL02/A1	43 —	— 55		
NMI			输入	外部中断输入 （不可屏蔽的、模拟噪声消除）	P02	6	6
NC			—	非连接（保持开路）	—	—	24、25
RD			输出	用于外部存储器的读取选通信号输出	P61/TOAB1B1/TIAB10/TOAB10 PCT4	66 —	— 87
REGC					—	稳压器输出稳定电容的连接（4.7 μF：初始值）	—
RESET	输入	系统复位输入	—	14	18		
RTC1HZ	输出	实时计数器校正时钟（1 Hz）输出	P35/TIAA11/TOAA11	30	42		
RTCCL	输出	实时计数器时钟（32 kHz 原始振荡）输出	P33/TIAA01/TOAA01/RTCDIV	28	40		
RTCDIV	输出	实时计数器时钟（32 kHz 分频）输出	P33/TIAA01/TOAA01/RTCCL	28	40		
RTP00	输出	实时输出端口。 N 沟道漏极开路输出可选	P50/TIAB01/KR0/TOAB01/UDMARQ1	35	47		
RTP01			P51/TIAB02/KR1/TOAB02/UDMAAK1	36	48		
RTP02			P52/TIAB03/KR2/TOAB03/DDI P20/TIAB03/KR2/TOAB03	37 —	— 32		
			P53/SIF2/TIAB00/KR3/TOAB00/DDO P21/SIF2/TIAB00/KR3/TOAB00	38 —	— 33		
RTP03			P54/SOF2/KR4/DCK P22/SOF2/KR4	39 —	— 34		
			P55/SCKF2/KR5/DMS P23/SCKF2/KR5	40 —	— 35		
RTP04							
RTP05							

备注 JG3-H: V850ES/JG3-H, JH3-H: V850ES/JH3-H

(5/9)

引脚名称	I/O	功能	复用功能	引脚编号	
				JG3-H	JH3-H
RXDC0	输入	串行接收数据输入 (UARTC0 至 UARTC4)	P31/SIF4/INTP08	26	38
RXDC1			P91/KR7/SCL02	43	—
			P91/KR7/SCL02/A1	—	55
RXDC2			P911/SOF3/INTP15	53	—
			P911/SOF3/INTP15/A11	—	67
RXDC3			P37/SDA00/CRXD0 [※] /UDMAK0	32	44
RXDC4			P41/SOF0/SCL01	23	30
SCKF0	I/O	串行时钟 I/O (CSIF0 至 CSIF4) N 沟道漏极开路输出可选	P42/INTP10	24	31
SCKF1			P99/INTP13	51	—
			P99/INTP13/A9	—	65
SCKF2			P55/KR5/RTP05/DMS	40	—
			P23/KR5/RTP05	—	35
SCKF3			P912	54	—
			P912/A12	—	68
SCKF4			P32/ASCKC0/TIAA00/TOAA00	27	39
SCL00	I/O	串行时钟 I/O (I ² C00 至 I ² C02) N 沟道漏极开路输出可选	P36/TXDC3/CTXD0 [※] /UDMARQ0	31	43
SCL01			P41/SOF0/RXDC4	23	30
			P91/KR7/RXDC1	43	—
SCL02			P91/KR7/RXDC1/A1	—	55
SDA00	I/O	串行发送/接收数据时钟 I/O (I ² C00 至 I ² C02) N 沟道漏极开路输出可选	P37/RXDC3/CRXD0 [※] /UDMAK0	32	44
SDA01			P40/SIF0/TXDC4	22	29
			P90/KR6/TXDC1	42	—
SDA02			P90/KR6/TXDC1/A0	—	54
SIF0	输入	串行接收数据输入 (CSIF0 至 CSIF4)	P40/TXDC4/SDA01	22	29
SIF1			P97/TIAA20/TOAA20	49	—
			P97/TIAA20/TOAA20/A7	—	63
SIF2			P53/TIAB00/KR3/TOAB00/RTP03/DDO	38	—
			P21/TIAB00/KR3/TOAB00/RTP03	—	33
SIF3			P910/TXDC2/INTP14	52	—
			P910/TXDC2/INTP14/A10	—	66
SIF4			P31/RXDC0/INTP08	26	38
SOF0	输出	串行发送数据输出 (CSIF0 至 CSIF4) N 沟道漏极开路输出可选	P41/RXDC4/SCL01	23	30
SOF1			P98/INTP12	50	—
			P98/INTP12/A8	—	64
SOF2			P54/KR4/RTP04/DCK	39	—
			P22/KR4/RTP04	—	34
SOF3			P911/RXDC2/INTP15	53	—
			P911/RXDC2/INTP15/A11	—	67
SOF4			P30/TXDC0/INTP07	25	37

注 仅适用于μPD70F3770、70F3771

备注 JG3-H: V850ES/JG3-H, JH3-H: V850ES/JH3-H

(6/9)

引脚名称	I/O	功能	复用功能	引脚编号	
				JG3-H	JH3-H
TECR0	输入	TMT0 编码器清零输入	P93/TIT00/TOT00	45	—
			P93/TIT00/TOT00/A3	—	57
TENC00		TMT0 编码器输入	P94/TIAA31/TOAA31/EVTT0	46	—
			P94/TIAA31/TOAA31/EVTT0/A4	—	58
TENC01			P92/TIT01/TOT01	44	—
			P92/TIT01/TOT01/A2	—	56
TIAA00	输入	外部事件计数输入/捕获触发器输入/外部触发器输入 (TAA0)	P32/ASCKC0/SCKF4/TOAA00	27	39
TIAA01		捕获触发器输入 (TAA0)	P33/TOAA01/RTCDIV/RTCCL	28	40
TIAA10		外部事件计数输入/捕获触发器输入/外部触发器输入 (TAA1)	P34/TOAA10/TOAA1OFF/INTP09	29	41
TIAA11		捕获触发器输入 (TAA1)	P35/TOAA11/RTC1HZ	30	42
TIAA20		外部事件计数输入/捕获触发器输入/外部触发器输入 (TAA2)	P97/SIF1/TOAA20	49	—
			P97/SIF1/TOAA20/A7	—	63
TIAA21		捕获触发器输入 (TAA2)	P96/TOAA21/INTP11	48	—
			P96/TOAA21/INTP11/A6	—	62
TIAA30		外部事件计数输入/捕获触发器输入/外部触发器输入 (TAA3)	P95/TOAA30	47	—
			P95/TOAA30/A5	—	59
TIAA31		捕获触发器输入 (TAA3)。	P94/TOAA31/TENC00/EVTT0	46	—
			P94/TOAA31/TENC00/EVTT0/A4	—	58
TIAA50		外部事件计数输入/捕获触发器输入/外部触发器输入 (TAA5)	P915/TOAA50/INTP18	57	—
			P915/TOAA50/INTP18/A15	—	71
TIAA51		捕获触发器输入 (TAA5)	P914/TOAA51/INTP17	56	—
			P914/TOAA51/INTP17/A14	—	70
TIAB00	输入	外部事件计数输入/捕获触发器输入/外部触发器输入 (TAB0)	P53/SIF2/KR3/TOAB00/RTP03/DDO	38	—
			P21/SIF2/KR3/TOAB00/RTP03	—	33
TIAB01		捕获触发器输入 (TAB0)	P50/KR0/TOAB01/RTP00/UDMARQ1	35	47
TIAB02			P51/KR1/TOAB02/RTP01/UDMAAK1	36	48
TIAB03			P52/KR2/TOAB03/RTP02/DDI	37	—
			P20/KR2/TOAB03/RTP02	—	32
TIAB10	输入	捕获触发器输入 (TAB1)	P61/TOAB1B1/TOAB10/RD	66	—
			P61/TOAB1B1/TOAB10	—	91
TIAB11			P60/TOAB1T1/TOAB11/WAIT	65	—
			P60/TOAB1T1/TOAB11	—	90
TIAB12			P62/TOAB1T2/TOAB12/ASTB	67	—
			P62/TOAB1T2/TOAB12	—	92
TIAB13			P64/TOAB1T3/TOAB13/CS2	69	—
			P64/TOAB1T3/TOAB13	—	94

备注 JG3-H: V850ES/JG3-H, JH3-H: V850ES/JH3-H

(7/9)

引脚名称	I/O	功能	复用功能	引脚编号	
				JG3-H	JH3-H
TIT00	输入	TMT0 外部触发器输入/捕获触发器输入	P93/TECR0/TOT00	45	—
			P93/TECR0/TOT00/A3	—	57
TIT01	输入	TMT0 捕获触发器输入	P92/TENC01/TOT01	44	—
			P92/TENC01/TOT01/A2	—	56
TOAA00	输出	定时器输出（TAA0） N 沟道漏极开路输出可选	P32/ASCKC0/SCKF4/TIAA00	27	39
TOAA01			P33/TIAA01/RTCDIV/RTCCL	28	40
TOAA10		定时器输出（TAA1） N 沟道漏极开路输出可选	P34/TIAA10/TOAA1OFF/INTP09	29	41
TOAA11			P35/TIAA11/RTC1HZ	30	42
TOAA1OFF	输入	TAA1 高阻抗输出控制信号输入	P34/TIAA10/TOAA10/INTP09	29	41
TOAA20	输出	定时器输出（TAA2） N 沟道漏极开路输出可选	P97/SIF1/TIAA20	49	—
TOAA21			P97/SIF1/TIAA20/A7	—	63
			P96/TIAA21/INTP11	48	—
			P96/TIAA21/INTP11/A6	—	62
TOAA30	定时器输出（TAA3） N 沟道漏极开路输出可选	P95/TIAA30	47	—	
TOAA31			P95/TIAA30/A5	—	59
			P94/TIAA31/TENC00/EVTT0	46	—
			P94/TIAA31/TENC00/EVTT0/A4	—	58
TOAA50	定时器输出（TAA5） N 沟道漏极开路输出可选	P915/TIAA50/INTP18	57	—	
TOAA51			P915/TIAA50/INTP18/A15	—	71
			P914/TIAA51/INTP17	56	—
			P914/TIAA51/INTP17/A14	—	70
TOAB00	输出	定时器输出（TAB0） N 沟道漏极开路输出可选	P53/SIF2/TIAB00/KR3/RTP03/DDO	38	—
TOAB01			P21/SIF2/TIAB00/KR3/RTP03	—	33
			P50/TIAB01/KR0/RTP00/UDMARQ1	35	47
			P51/TIAB02/KR1/RTP01/UDMAAK1	36	48
			P52/TIAB03/KR2/RTP02/DDI	37	—
			P20/TIAB03/KR2/RTP02	—	32
TOAB1OFF	输入	TAB1 高阻抗输出控制信号输入	P913/INTP16	55	—
			P913/INTP16/A13	—	69
TOAB10	输出	定时器输出（TAB1）	P61/TOAB1B1/TIAB10/RD	66	—
TOAB11			P61/TOAB1B1/TIAB10	—	91
			P60/TOAB1T1/TIAB11/WAIT	65	—
			P60/TOAB1T1/TIAB11	—	90
TOAB12			P62/TOAB1T2/TIAB12/ASTB	67	—
			P62/TOAB1T2/TIAB12	—	92
TOAB13			P64/TOAB1T3/TIAB13/CS2	69	—
			P64/TOAB1T3/TIAB13	—	94

备注 JG3-H: V850ES/JG3-H, JH3-H: V850ES/JH3-H

(8/9)

引脚名称	I/O	功能	复用功能	引脚编号	
				JG3-H	JH3-H
TOAB1B1	输出	用于 6 相位 PWM 下桥臂的 TAB1 的脉冲信号输出	P61/TIAB10/TOAB10/RD	66	—
			P61/TIAB10/TOAB10	—	91
TOAB1B2			P63/TRGAB1/CS0	68	—
			P63/TRGAB1	—	93
TOAB1B3			P65/EVTAB1/CS3	70	—
			P65/EVTAB1	—	95
TOAB1T1	输出	用于 6 相位 PWM 上桥臂的 TAB1 的脉冲信号输出	P60/TOAB11/TIAB11/WAIT	65	—
			P60/TIAB11/TOAB11	—	90
TOAB1T2			P62/TIAB12/TOAB12/ASTB	67	—
			P62/TIAB12/TOAB12	—	92
TOAB1T3			P64/TOAB13/TIAB13/CS2	69	—
			P64/TIAB13/TOAB13	—	94
TOT00	输出	TMT0 定时器输出	P93/TECR0/TIT00	45	—
			P93/TECR0/TIT00/A3	—	57
TOT01			P92/TENC01/TIT01	44	—
			P92/TENC01/TIT01/A2	—	56
TRGAB1	输入	TAB1 的外部触发器输入	P63/TOAB1B2/CS0	68	—
			P63/TOAB1B2	—	93
TXDC0	输出	串行发送数据输出 (UARTC0 至 UARTC4) N 沟道漏极开路输出可选	P30/SOF4/INTP07	25	37
TXDC1			P90/KR6/SDA02	42	—
			P90/KR6/SDA02/A0	—	54
TXDC2			P910/SIF3/INTP14	52	—
			P910/SIF3/INTP14/A10	—	66
TXDC3			P36/SCL00/CTXD0 [‡] /UDMARQ0	31	43
TXDC4			P40/SIF0/SDA01	22	29
UCLK	输入	USB 时钟信号输入	P03/INTP02/ADTRG	7	7
UDMAAK0	输出	用于 USB 的 DMA 响应	P37/RXDC3/SDA00/CRXD0 [‡]	32	44
UDMAAK1		用于 USB 的 DMA 响应	P51/TIAB02/KR1/TOAB02/RTP01	36	48
UDMARQ0	输入	用于 USB 的 DMA 请求	P36/TXDC3/SCL00/CTXD0 [‡]	31	43
UDMARQ1		用于 USB 的 DMA 请求	P50/TIAB01/KR0/TOAB01/RTP00	35	47
UDMF	I/O	USB 数据 I/O (−) 功能	—	17	21
UDPF		USB 数据 I/O (+) 功能	—	18	22
UV _{DD}	—	用于 USB 的 3.3 V 正极电源	—	19	23
V _{DD}	—	内部器件的正极电源引脚	—	9、60	13、82
V _{SS}	—	内部器件的地电位	—	11、33、62、79	15、45、60、84、106

注 仅适用于 μ PD70F3770、70F3771

备注 JG3-H: V850ES/JG3-H, JH3-H: V850ES/JH3-H

(9/9)

引脚名称	I/O	功能	复用功能	引脚编号	
				JG3-H	JH3-H
$\overline{\text{WAIT}}$	输入	外部等待输入	P60/TOAB1T1/TIAB11/TOAB11 PCM0	65 —	— 89
$\overline{\text{WR0}}$	输出	用于外部存储器的写选通（低 8 位）	PCT0	58	76
$\overline{\text{WR1}}$		用于外部存储器的写选通（高 8 位）	PCT1	59	77
X1	输入	用于主时钟的振荡器连接	—	12	16
X2	—		—	13	17
XT1	输入	用于副时钟的振荡器连接	—	15	19
XT2	—		—	16	20

备注 JG3-H: V850ES/JG3-H, JH3-H: V850ES/JH3-H

2.2 引脚状态

各模式时的引脚工作状态如下所示：

表 2-2. 各模式下的引脚工作状态

引脚名称	启动电源时 ^{注1}	复位期间（除了启动电源时）	HALT 模式 ^{注2}	IDLE1、IDLE2、Sub-IDLE 模式 ^{注2}	STOP 模式 ^{注2}	Idle 状态 ^{注3}	总线保持
$\overline{\text{DRST}}$	下拉	下拉 ^{注4}	保持	保持	保持	保持	保持
P10/ANO0、P11/ANO1	不确定	Hi-Z	保持	保持	Hi-Z	保持	保持
AD0~AD15	Hi-Z ^{注5}	Hi-Z ^{注5}	注 6、7	Hi-Z	Hi-Z	Held	Hi-Z
A0~A15			不确定 ^{注6、8}				
A16~A21			不确定 ^{注6}				
$\overline{\text{WAIT}}$			—	—	—	—	—
CLKOUT			运行	L	L	运行	运行
$\overline{\text{WR0}}$ 、 $\overline{\text{WR1}}$			H ^{注6}	H	H	H	Hi-Z
$\overline{\text{RD}}$							
$\overline{\text{ASTB}}$			运行 ^{注6}	—	—	—	L
$\overline{\text{HLDAK}}$							运行
$\overline{\text{HLDRQ}}$							
其它端口引脚	Hi-Z	Hi-Z	保持	保持	保持	保持	保持

- 注
1. 接通电源时，供电电压达到工作电压范围后（下限），持续等待 1 ms。
 2. 使用复用功能时有效。
 3. 在 T3 状态后，插入该引脚在空闲状态时的状态。
 4. 外部复位期间下拉。通过看门狗定时器或时钟监视器等进行的内部复位期间，该引脚的状态根据 OCDM.OCDM0 位设置的不同而异。
 5. 总线控制引脚复用为端口引脚，因此将它们初始化为输入模式（端口模式）。
 6. DMA 操作期间，即使在 HALT 模式下仍然工作。
 7. 在独立总线模式下：： Hi-Z
在复用总线模式下：： 不确定
 8. 独立总线模式

备注

Hi-Z: 高阻抗
保持: 外部总线周期之前的瞬间的状态被保持下来。
L: 低电平输出
H: 高电平输出
—: 无采样输入（不响应）

2.3 引脚I/O电路类型、I/O缓冲器电源及未使用引脚的连接

表 2-3. 引脚 I/O 电路类型、I/O 缓冲器电源及未使用引脚的连接 (1/4)

引脚名称	复用功能	I/O 电路类型	推荐的连接方式	JG3-H	JH3-H
P00	INTP00	10-D	输入：通过电阻单独连接至 EV_{DD} 或 V_{SS} 。 输出：保持开路。	—	√
P01	INTP01			—	√
P02	NMI			√	√
P03	INTP02/ADTRG/UCLK			√	√
P04	INTP03			√	√
P05	INTP04			√	√
P10	ANO0	12-D	输入：通过电阻单独连接至 AV_{REF1} 或 AV_{SS} 。 输出：保持开路。	√	√
P11	ANO1			√	√
P20	TIAB03/KR2/TOAB03/RTP02	10-D	输入：通过电阻单独连接至 EV_{DD} 或 V_{SS} 。 输出：保持开路。	—	√
P21	SIF2/TIAB00/KR3/TOAB00/RTP03			—	√
P22	SOF2/KR4/RTP04			—	√
P23	SCKF2/KR5/RTP05			—	√
P24	INTP05			—	√
P25	INTP06			—	√
P30	TXDC0/SOF4/INTP07	10-D	输入：通过电阻单独连接至 EV_{DD} 或 V_{SS} 。 输出：保持开路。	√	√
P31	RXDC0/SIF4/INTP08			√	√
P32	ASCKC0/SCKF4/TIAA00/TOAA00			√	√
P33	TIAA01/TOAA01/RTCDIV/RTCCL			√	√
P34	TIAA10/TOAA10/TOAA1OFF/INTP09			√	√
P35	TIAA11/TOAA11/RTC1HZ			√	√
P36	TXDC3/SCL00/CTXD0 [※] /UDMARQ0			√	√
P37	RXDC3/SDA00/CRXD0 [※] /UDMAAK0			√	√
P40	SIF0/TXDC4/SDA01	10-D	输入：通过电阻单独连接至 EV_{DD} 或 V_{SS} 。 输出：保持开路。	√	√
P41	SOF0/RXDC4/SCL01			√	√
P42	SCKF0/INTP10			√	√
P50	TIAB01/KR0/TOAB01/RTP00/UDMARQ1	10-D	输入：通过电阻单独连接至 EV_{DD} 或 V_{SS} 。 输出：保持开路。	√	—
P51	TIAB02/KR1/TOAB02/RTP01/UDMAAK1			√	—
P52	TIAB03/KR2/TOAB03/RTP02/DDI			√	—
P53	SIF2/TIAB00/KR3/TOAB00/RTP03/DDO			√	—
P54	SOF2/KR4/RTP04/DCK			√	—
P55	SCKF2/KR5/RTP05/DMS			√	—
P56	INTP05/DRST	10-N	输入：通过电阻单独连接至 V_{SS} 。禁止固定连接至电平 V_{DD} 。 输出：保持开路。 通过 RESET 引脚复位后内部下拉。	√	—

注 仅适用于 μ PD70F3770、70F3771

备注 JG3-H: V850ES/JG3-H, JH3-H: V850ES/JH3-H

表 2-3. 引脚 I/O 电路类型、I/O 缓冲器电源及未使用引脚的连接 (2/4)

引脚名称	复用功能	I/O 电路类型	推荐的连接方式	JG3-H	JH3-H
P60	TOAB1T1/TIAB11/TOAB11/ $\overline{\text{WAIT}}$ TOAB1T1/TIAB11/TOAB11	10-D	输入：通过电阻单独连接至 EV_{DD} 或 V_{SS} 。 输出：保持开路。	√	—
P61	TOAB1B1/TIAB10/TOAB10/ $\overline{\text{RD}}$ TOAB1B1/TIAB10/TOAB10			√	—
P62	TOAB1T2/TIAB12/TOAB12/ $\overline{\text{ASTB}}$ TOAB1T2/TIAB12/TOAB12			√	—
P63	TOAB1B2/TRGAB1/ $\overline{\text{CS0}}$ TOAB1B2/TRGAB1			√	—
P64	TOAB1T3/TIAB13/TOAB13/ $\overline{\text{CS2}}$ TOAB1T3/TIAB13/TOAB13			√	—
P65	TOAB1B3/EVTAB1/ $\overline{\text{CS3}}$ TOAB1B3/EVTAB1			√	—
P70~ P711	ANI0~ANI11	11-G	输入：通过电阻单独连接至 AV_{REF0} 或 AV_{SS} 。 输出：保持开路。	√	√
P90	KR6/TXDC1/SDA02 KR6/TXDC1/SDA02/A0	10-D	输入：通过电阻单独连接至 EV_{DD} 或 V_{SS} 。 输出：保持开路。	√	—
P91	KR7/RXDC1/SCL02 KR7/RXDC1/SCL02/A1			√	—
P92	TENC01/TIT01/TOT01 TENC01/TIT01/TOT01/A2			√	—
P93	TECR0/TIT00/TOT00 TECR0/TIT00/TOT00/A3			√	—
P94	TIAA31/TOAA31/TENC00/EVTT0 TIAA31/TOAA31/TENC00/EVTT0/A4			√	—
P95	TIAA30/TOAA30 TIAA30/TOAA30/A5			√	—
P96	TIAA21/TOAA21/INTP11 TIAA21/TOAA21/INTP11/A6			√	—
P97	SIF1/TIAA20/TOAA20 SIF1/TIAA20/TOAA20/A7			√	—
P98	SOF1/INTP12 SOF1/INTP12/A8			√	—
P99	SCKF1/INTP13 SCKF1/INTP13/A9			√	—
P910	SIF3/TXDC2/INTP14 SIF3/TXDC2/INTP14/A10			√	—
P911	SOF3/RXDC2/INTP15 SOF3/RXDC2/INTP15/A11			√	—
P912	SCKF3 SCKF3/A12			√	—

备注 JG3-H: V850ES/JG3-H, JH3-H: V850ES/JH3-H

表 2-3. 引脚 I/O 电路类型、I/O 缓冲器电源及未使用引脚的连接 (3/4)

引脚名称	复用功能	I/O 电路类型	推荐的连接方式	JG3-H	JH3-H
P913	TOAB1OFF/INTP16	10-D	输入： 通过电阻单独连接至 EV_{DD} 或 V_{SS} 。 输出： 保持开路。	√	—
	TOAB1OFF/INTP16/A13			—	√
P914	TIAA51/TOAA51/INTP17			√	—
	TIAA51/TOAA51/INTP17/A14			—	√
P915	TIAA50/TOAA50/INTP18			√	—
	TIAA50/TOAA50/INTP18/A15			—	√
PCM0	WAIT	5	输入： 通过电阻单独连接至 EV_{DD} 或 V_{SS} 。 输出： 保持开路。	—	√
PCM1	CLKOUT			√	√
PCM2	HLDK			—	√
PCM3	HLDRQ			—	√
PCS0	CS0	5	输入： 通过电阻单独连接至 EV_{DD} 或 V_{SS} 。 输出： 保持开路。	—	√
PCS2	CS2			—	√
PCS3	CS3			—	√
PCT0	WR0	5	输入： 通过电阻单独连接至 EV_{DD} 或 V_{SS} 。 输出： 保持开路。	√	√
PCT1	WR1			√	√
PCT4	RD			—	√
PCT6	ASTB			—	√
PDH0~ PDH7	A16~A23	5	输入： 通过电阻单独连接至 EV_{DD} 或 V_{SS} 。 输出： 保持开路。	—	√
PDL0~ PDL4	AD0~AD4	5	输入： 通过电阻单独连接至 EV_{DD} 或 V_{SS} 。 输出： 保持开路。	√	√
PDL5	AD5/FLMD1			√	√
PDL6~ PDL15	AD6~AD15			√	√
AV _{REF0}	—	—	直接连接至 V_{DD} 且始终供电。	√	√
AV _{REF1}	—	—		√	√
AV _{SS}	—	—	直接连接至 V_{SS} 且始终供电。	√	√
DCK	—	—	连接至 V_{SS} 。	—	√
DDI	—	—	连接至 V_{SS} 。	—	√
DDO	—	—	保持开路。	—	√
DMS	—	—	连接至 V_{SS} 。	—	√
DRST	—	—	连接至 V_{SS} 。	—	√
EV _{DD}	—	—	直接连接至 V_{DD} 且始终供电。	√	√
FLMD0	—	—	在不处于 flash 模式时，直接连接至 V_{SS} 。	√	√
REGC	—	—	连接至稳压器输出稳定电容。	√	√
RESET	—	2	—	√	√
UDMF	—	—	保持开路。	√	√
UDPF	—	—	保持开路。	√	√

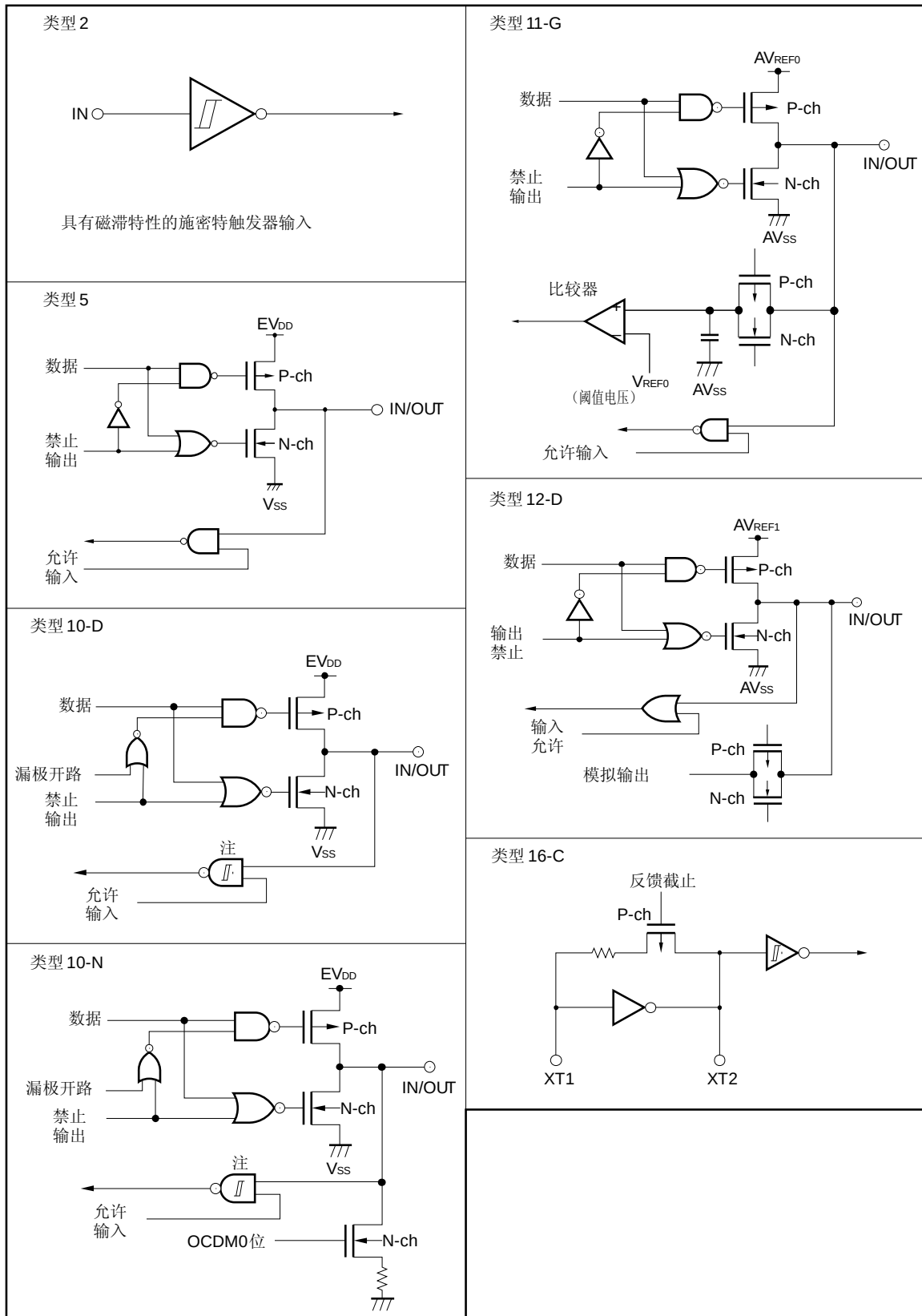
备注 JG3-H: V850ES/JG3-H, JH3-H: V850ES/JH3-H

表 2-3. 引脚 I/O 电路类型、I/O 缓冲器电源及未使用引脚的连接 (4/4)

引脚名称	复用功能	I/O 电路类型	推荐的连接方式	JG3-H	JH3-H
UVDD	—	—	直接连接至 V _{DD} 且始终供电。	√	√
VDD	—	—	—	√	√
VSS	—	—	—	√	√
X1	—	—	—	√	√
X2	—	—	—	√	√
XT1	—	16-C	通过电阻连接至 V _{SS} 。	√	√
XT2	—	16-C	保持开路。	√	√
NC	—	—	保持开路。	—	√

备注 JG3-H: V850ES/JG3-H, JH3-H: V850ES/JH3-H

图 2-1. 引脚 I/O 电路



注 端口模式下没有磁滞特性。

2.4 注意事项

电源开启时，在复位期间以下引脚可能暂时输出不确定电平。

- P10/ANO0 引脚
- P11/ANO1 引脚
- DDO 引脚（仅 V850ES/JH3-H）
- P53/SIF2/TIAB00/KR3/TOAB00/RTP03/DDO 引脚（仅 V850ES/JG3-H）

第三章 CPU功能

V850ES/JG3-H 和 V850ES/JH3-H 单片机的 CPU 基于 RISC 体系结构，通过使用 5 级流水线可在单时钟周期内执行几乎所有的指令。

3.1 特性

- 最短指令执行时间：20.8 ns（在 48 MHz 的主时钟（ f_{xx} ）： $V_{DD} = 2.85 \sim 3.6$ V 下工作）
30.5 μ s（在 32.768 kHz 的副时钟（ f_{xt} ）下工作）
- 存储空间：程序（物理地址）空间：64MB 线性空间。
数据（逻辑地址）空间：4GB 线性空间
- 通用寄存器：32 位×32 寄存器
- 内部 32 位体系结构
- 5 级流水线控制。
- 乘法/除法指令
- 溢出操作指令。
- 32 位移位指令：1 个时钟
- 长短两种格式的数据加载/存储指令
- 4 种类型位操作指令
 - SET1
 - CLR1
 - NOT1
 - TST1

3.2 CPU寄存器组

V850ES/JG3-H 和 V850ES/JH3-H 的寄存器可分为两种类型：通用程序寄存器和专用系统寄存器。所有寄存器都为 32 位寄存器。

详情参见 **V850ES 体系结构用户手册**。

(1) 程序寄存器组		(2) 系统寄存器组	
31	0	31	0
r0	(零寄存器)	EIPC	(中断状态保存寄存器)
r1	(汇编程序保留寄存器)	EIPSW	(中断状态保存寄存器)
r2			
r3	(堆栈指针 (SP))	FEPC	(NMI状态保存寄存器)
r4	(全局指针 (GP))	FEPSW	(NMI状态保存寄存器)
r5	(文本指针 (TP))		
r6		ECR	(中断源寄存器)
r7			
r8		PSW	(程序状态字)
r9			
r10		CTPC	(CALLT指令执行状态保存寄存器)
r11		CTPSW	(CALLT指令执行状态保存寄存器)
r12			
r13			
r14		DBPC	(异常/调试陷阱状态保存寄存器)
r15		DBPSW	(异常/调试陷阱状态保存寄存器)
r16			
r17			
r18			
r19			
r20			
r21			
r22			
r23			
r24			
r25			
r26			
r27			
r28			
r29			
r30	(元指针 (EP))		
r31	(链接指针 (LP))	CTBP	(CALLT指令基址指针)
31	0		
PC	(程序计数器)		

3.2.1 程序寄存器组

程序寄存器组包括若干通用寄存器和一个程序计数器。

(1) 通用寄存器（r0 至 r31）

共有 32 个通用寄存器（r0 至 r31）可供选择。这些寄存器中的任何一个都可用于存储数据变量或地址变量。然而，执行某些指令会隐性使用 r0 和 r30 寄存器，使用这些寄存器时一定要加以注意。寄存器 r0 总是保持为 0，用于使用数据 0 的操作或 0 偏移量寻址。当 SLD 和 SST 指令访问存储器时，r30 寄存器用作基址指针。汇编器和 C 语言编译器隐含使用 r1、r3 至 r5 和 r31。使用这些寄存器之前，必须保存寄存器中的内容以防其丢失，并在使用完寄存器后对其内容进行恢复。寄存器 r2 有时会用于实时操作系统。如果实时操作系统没有使用 r2，那么它将用作变量寄存器。

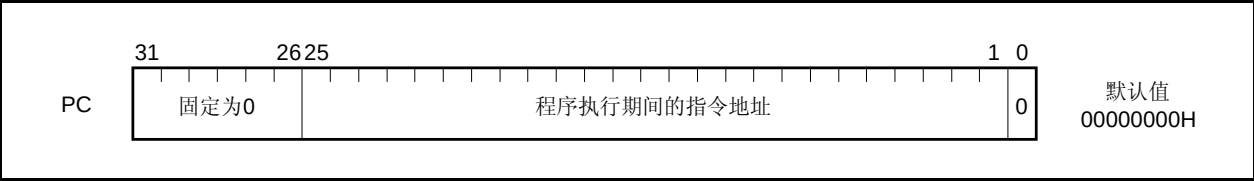
表 3-1. 程序寄存器

名称	用途	操作
r0	零寄存器	始终保持为零。
r1	汇编程序保留寄存器	用作生成 32 位瞬时数据的工作寄存器
r2	地址/数据变量寄存器（如果实时操作系统没有使用寄存器 r2）	
r3	堆栈指针	调用函数时用来产生堆栈框
r4	全局指针	用于访问数据区中的全局变量
r5	文本指针	用作指示文本区域起始地址的寄存器（该区域为程序代码存放区域）
r6~r29	地址/数据变量寄存器	
r30	元指针	用作访问内存的基址指针
r31	链接指针	编译器调用函数时使用
PC	程序计数器	在程序执行期间，保存指令地址

备注 有关用在汇编器和 C 编译器中的 r1、r3 至 r5 和 r31 的更多细节，请参见 CA850（C 编译程序软件包）汇编语言用户手册。

(2) 程序计数器（PC）

程序计数器用来保存程序执行期间的指令地址。该寄存器的低 32 位有效。其中位 31 至位 26 固定为 0。即使位 25 向位 26 有进位也将忽略不计。位 0 固定为 0。这意味着程序执行不会跳转到奇地址。



3.2.2 系统寄存器组

系统寄存器控制 CPU 的状态并保存中断信息。

可以使用下表所列的系统寄存器编号，利用系统寄存器的载入/存储指令（LDSR 和 STSR 指令），对这些寄存器进行读或写操作。

表 3-2. 系统寄存器编号

系统寄存器编号	系统寄存器名称	操作数说明	
		LDSR 指令	STSR 指令
0	中断状态保存寄存器（EIPC） ^{注 1}	√	√
1	中断状态保存寄存器（EIPSW） ^{注 1}	√	√
2	NMI 状态保存寄存器（FEPC） ^{注 1}	√	√
3	NMI 状态保存寄存器（FEPSW） ^{注 1}	√	√
4	中断源寄存器（ECR）	×	√
5	程序状态字（PSW）	√	√
6~15	未来功能扩展预留寄存器（如果对这些寄存器进行访问，不能保证操作的有效性）	×	×
16	CALLT 指令执行状态保存寄存器（CTPC）	√	√
17	CALLT 指令执行状态保存寄存器（CTPSW）	√	√
18	异常/调试陷阱状态保存寄存器（DBPC）	√ ^{注 2}	√ ^{注 2}
19	异常/调试陷阱状态保存寄存器（DBPSW）	√ ^{注 2}	√ ^{注 2}
20	CALLT 指令基址指针（CTBP）	√	√
21~31	未来功能扩展预留寄存器（如果对这些寄存器进行访问，不能保证操作的有效性）	×	×

注 1. 由于只有一组这样的中断状态保存寄存器可用，所以，如果允许多重中断，那么这些寄存器的内容必须由程序存储。

2. 对这些寄存器的访问，只有在执行 DBTRAP 指令或非法操作码与 DBRET 指令的间隔期间才能进行。

注意事项 即使 EIPC 或 FEPC、或 CTPC 的位 0 由 LDSR 指令设置为 1，在中断服务之后，通过 RETI 指令将程序返回到主例程时，仍然会忽略位 0（因为 PC 的位 0 固定为 0）。设置偶数值至 EIPC、FEPC 和 CTPC（位 0 = 0）。

备注 √: 能被访问
×: 禁止访问

(1) 中断状态保存寄存器（EIPC 和 EIPSW）

EIPC 和 EIPSW 用来保存中断发生时的状态。

如果产生软件异常或者可屏蔽中断，那么就将程序计数器（PC）中的内容保存到 EIPC 寄存器中，将程序状态字（PSW）中的内容保存到 EIPSW 寄存器中（如果产生的是非屏蔽中断，那么就将这些相应寄存器内容保存到 NMI 状态保存寄存器（FEPC 和 FEPSW）中）。

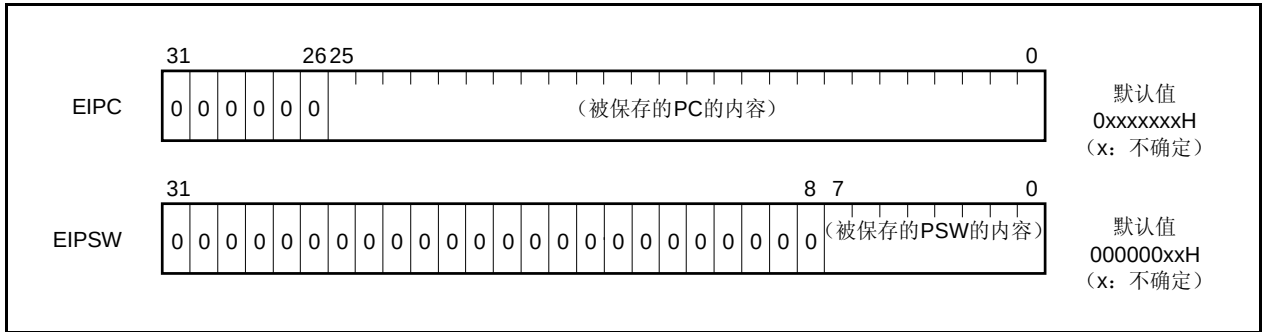
当有软件异常或者是可屏蔽中断产生时，就将正在执行指令的下一条指令地址保存到寄存器 EIPC 中，某些指令除外（可参见 **23.8 CPU 无法响应中断的周期**）。

将 PSW 的当前内容保存到 EIPSW 寄存器中。

由于只有一组中断状态保存寄存器可用，所以，当允许多重中断时，这些寄存器的内容必须在程序中通过编程加以保存。

EIPC 的位 31 至位 26 以及 EIPSW 的位 31 至位 8 作为保留位用于未来功能扩展（这些位始终固定为 0）。

通过执行 RETI 指令，将寄存器 EIPC 的值恢复到 PC 中、寄存器 EIPSW 的值恢复到 PSW 中。



(2) NMI 状态保存寄存器（FEPC 和 FEPSW）

FEPC 和 FEPSW 寄存器用来保存不可屏蔽中断（NMI）发生时的状态。

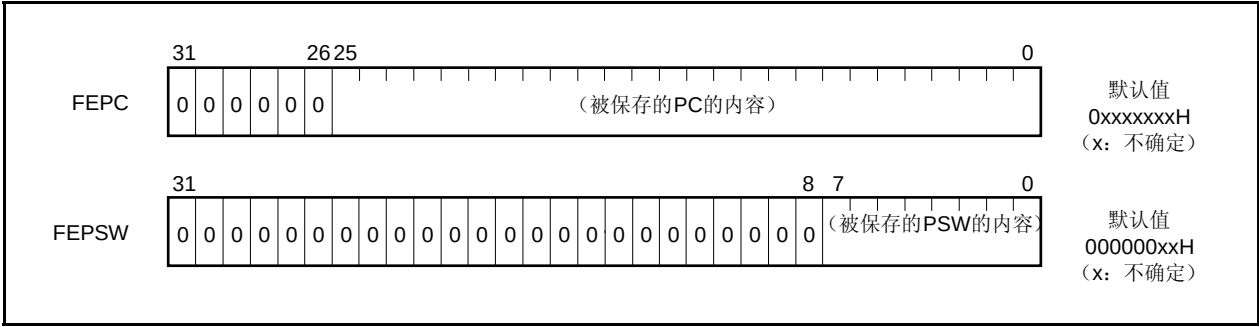
如果发生 NMI，那么就将程序计数器（PC）中的内容保存到 FEPC 寄存器中，将程序状态字（PSW）中的内容保存到 FEPSW 寄存器中。

当有 NMI 产生时，就将当前执行指令的下一条指令地址保存到寄存器 FEPC 中，某些指令除外。

将 PSW 的当前内容保存到 FEPSW 寄存器中。

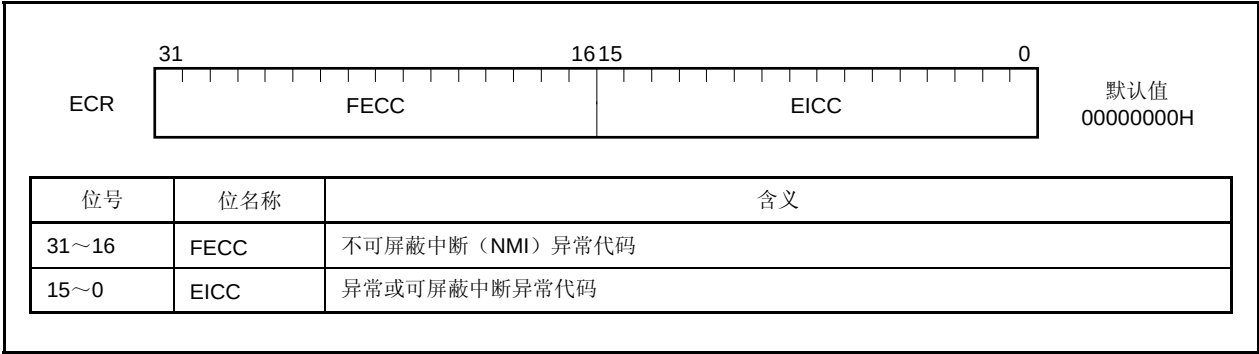
由于只有一组 NMI 状态保存寄存器可用，所以，当允许多重中断时，这些寄存器的内容必须在程序中通过编程加以保存。

FEPC 的位 31 至位 26 以及 FEPSW 的位 31 至位 8 作为保留位用于未来功能扩展（这些位始终固定为 0）。通过执行 RETI 指令，将寄存器 FEPC 的值恢复到 PC 中、寄存器 FEPSW 的值恢复到 PSW 中。



(3) 中断源寄存器（ECR）

如果有中断或异常产生，那么，异常或中断源信息就保存在该中断源寄存器（ECR）中。该寄存器保存各中断源的异常代码。由于该寄存器是只读寄存器，所以，不能通过 LDSR 指令将数据写入该寄存器。



(4) 程序状态字 (PSW)

程序状态字 (PSW) 是一个指示程序状态 (指令执行结果) 和 CPU 状态的标志位的集合。

如果使用 LDSR 指令更改了该寄存器的内容, 那么, 在 LDSR 指令完成后, 新内容就会立即生效。然而, LDSR 指令执行期间, 如果将 ID 标志设为 1, 则将不会响应中断请求。

该寄存器的位 31 至位 8 作为保留位用于未来功能扩展 (这些位固定为 0)。

(1/2)

PSW		31 <															

注 已经执行溢出处理操作的结果由 OV 标志和 S 标志的值来确定。当执行溢出操作时，只有在 OV 标志位置为 1 时，SAT 标志位才置为 1。

运算结果的状态	标志位状态			溢出处理的操作结果
	SAT	OV	S	
超过正值上界	1	1	0	7FFFFFFFH
越过负值下界	1	1	1	80000000H
正值（未超上界）	运算前保存该值	0	0	操作结果本身
负值（未越下界）			1	

(5) CALLT 指令执行状态保存寄存器（CTPC 和 CTPSW）

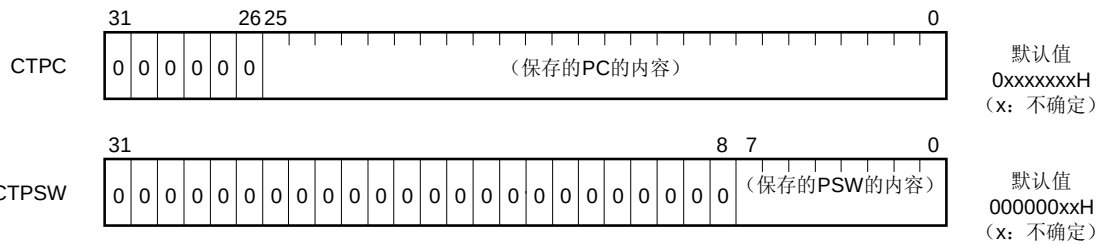
CTPC 和 CTPSW 是 CALLT 指令的执行状态保存寄存器。

当执行 CALLT 指令时，程序计数器（PC）中的内容就保存到 CTPC 寄存器中，同时程序状态字（PSW）中的内容保存到 CTPSW 寄存器中。

寄存器 CTPC 中保存的内容是 CALLT 指令下一条指令的地址。

将 PSW 的当前内容保存到 CTPSW 寄存器中。

CTPSW 的位 31 至位 8 和 CTPSW 的位 31 至位 26 作为保留位用于未来功能扩展（这些位固定为 0）。



(6) 异常/调试陷阱状态保存寄存器 (DBPC 和 DBPSW)

DBPC 和 DBPSW 为异常/调试陷阱状态寄存器。

如果产生异常陷阱或调试陷阱，那么就将程序计数器（PC）中的内容保存到 DBPC 寄存器中，同时将程序状态字（PSW）中的内容保存到 DBPSW 寄存器中。

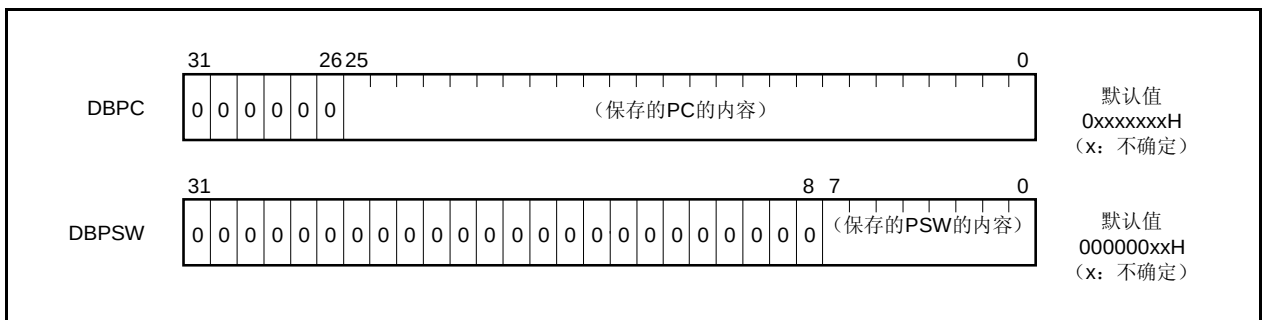
在异常陷阱或者调试陷阱产生时，保存到 DBPC 寄存器中的内容是正在执行指令的下一条指令的地址。

将 PSW 的当前内容保存到 DBPSW 寄存器中。

只有在 DBTRAP 指令或非法操作码与 DBRET 指令执行的间隔期间，才能对该寄存器进行读写操作。

DBPC 的位 31 至位 26 以及 DBPSW 的位 31 至位 8 作为保留位用于未来功能扩展（这些位始终固定为 0）。

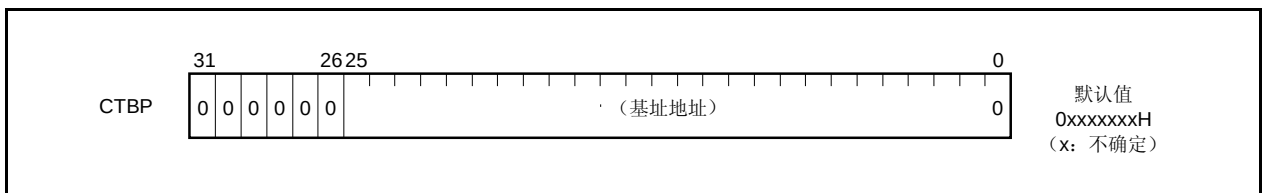
通过执行 DBRET 指令，将寄存器 DBPC 的值恢复到 PC 中，并将寄存器 DBPSW 的值恢复到 PSW 中。



(7) CALLT 指令基址指针 (CTBP)

CALLT 基址指针 (CTBP) 用于指定表地址或生成一个目标地址 (位 0 固定为 0)。

该寄存器的位 31 至位 26 作为保留位用于未来功能扩展（这些位固定为 0）。



3.3 操作模式

V850ES/JG3-H 和 V850ES/JH3-H 有以下操作模式。

(1) 正常操作模式

该模式下，在系统复位状态解除后，将各个与总线接口关联的引脚均设置为端口模式。程序执行转移至内部 ROM 中的复位入口地址处，然后开始指令处理。

(2) Flash 存储器编程模式

该模式下，使用 flash 编程器可以对内部 flash 存储器进行编程。

(3) 片上调试模式

V850ES/JG3-H 和 V850ES/JH3-H 提供使用 JTAG（联合测试行动小组）通讯规范的片上调试功能。
详情参见第 32 章 片上调试功能。

3.3.1 指定操作模式

通过 FLMD0 引脚和 FLMD1 引脚来指定操作模式。

在正常操作模式下，复位解除时，确保在 FLMD0 引脚输入低电平。

在 flash 存储器编程模式下，如果连通 flash 编程器后，可以从 flash 编程器向 FLMD0 引脚输入高电平，然而，在自编程模式下，必须从外部电路输入高电平。

复位解除时的操作		复位后的操作模式
FLMD0	FLMD1	
L	×	正常操作模式
H	L	Flash 存储器编程模式
H	H	禁止设置

备注

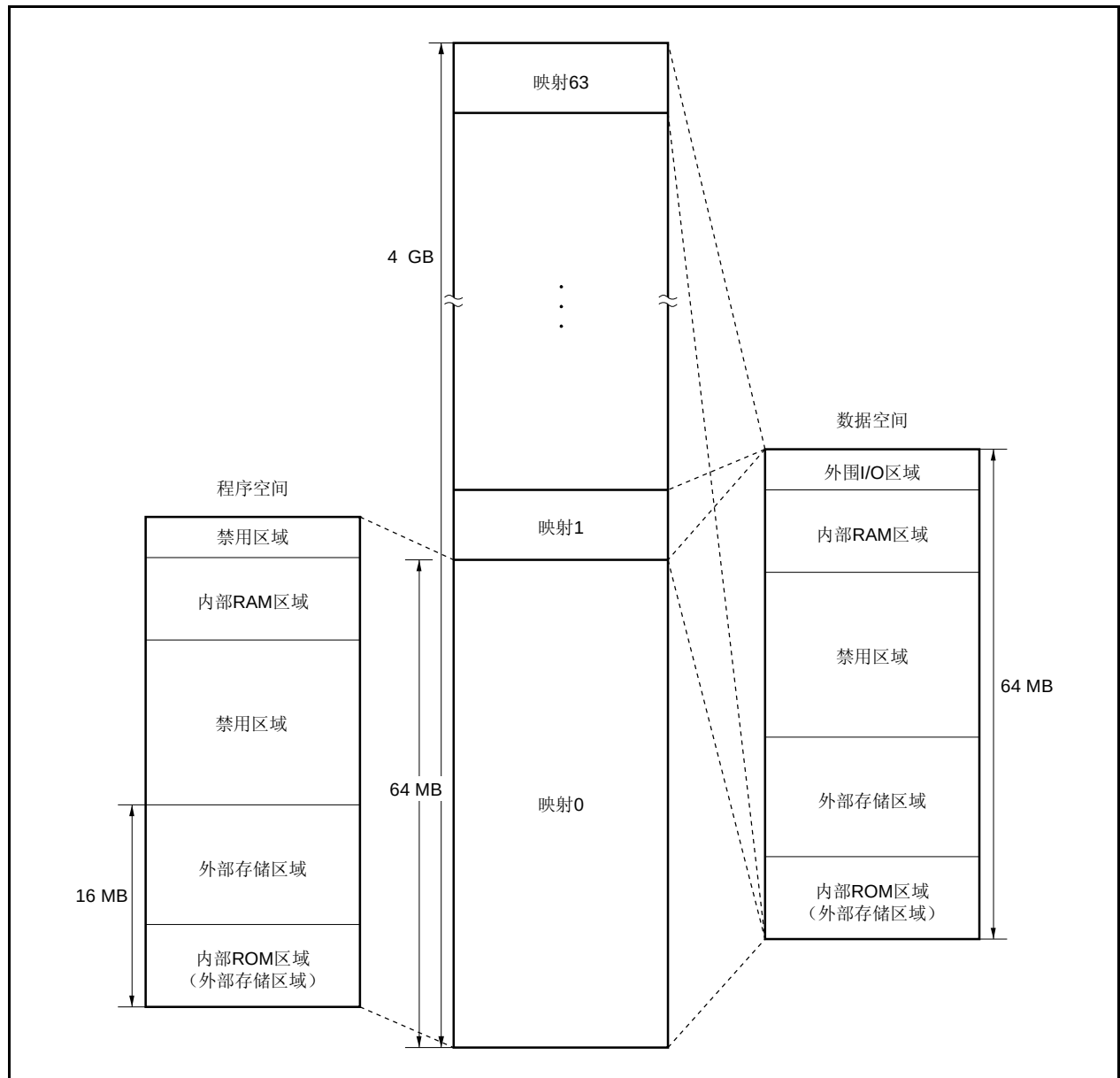
L: 低电平输入
H: 高电平输入
×: 不需理会

3.4 地址空间

3.4.1 CPU地址空间

对于指令地址而言，外部存储区域和内部 ROM 区域合起来可达 16 MB，再计入内部 RAM 区域，可以支持多达 64 MB 的线性地址空间。对于操作数寻址（数据访问）来说，则支持一个最大为 4GB 的线性地址空间（数据空间）。然而，4 GB 的地址空间可看作是 64 个 64 MB 的物理地址空间映射。这意味着无论位 31 至位 26 的取值如何，系统总是对同样 64MB 物理地址空间进行访问。

图 3-1. 地址空间映射

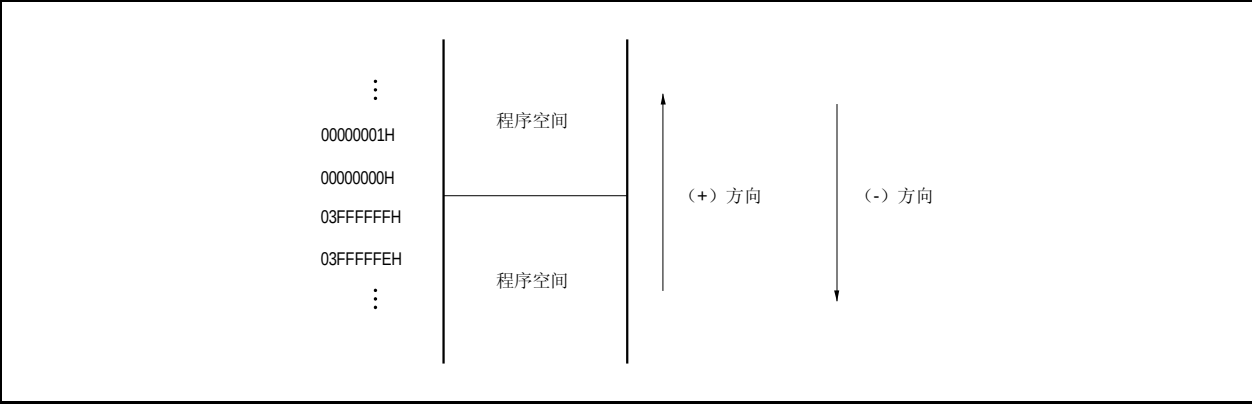


3.4.2 CPU 地址空间的绕回

(1) 程序空间

对于 32 位的 PC（程序计数器），其高 6 位固定为 0，且只有低 26 位是有效的。跳转地址计算期间，高 6 位会忽略位 25 至位 26 所发生的进位或借位。
因此，程序空间的上限地址 03FFFFFFH 和下限地址 00000000H 是连续相邻的地址。像这样，程序空间的最高地址和最低地址为连续相邻的地址，称为地址的绕回。

注意事项 由于地址 03FFF000H 至 03FFFFFFH 的 4 KB 地址区域为片上外围 I/O 区域，所以不能从该地址区域取指令。因此，不要在跳转地址计算影响的区域执行操作。



(2) 数据空间

操作数地址计算的结果超出 32 位时，该结果地址将被忽略。
因此，数据空间的上限地址 FFFFFFFFH 和下限地址 00000000H 是连续相邻地址，并且绕回发生于这两个边界地址处。

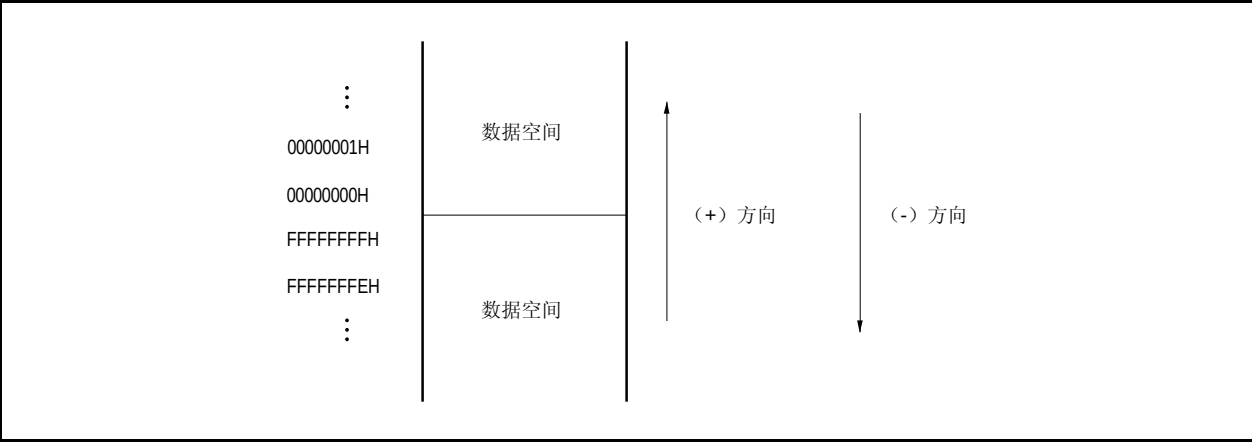
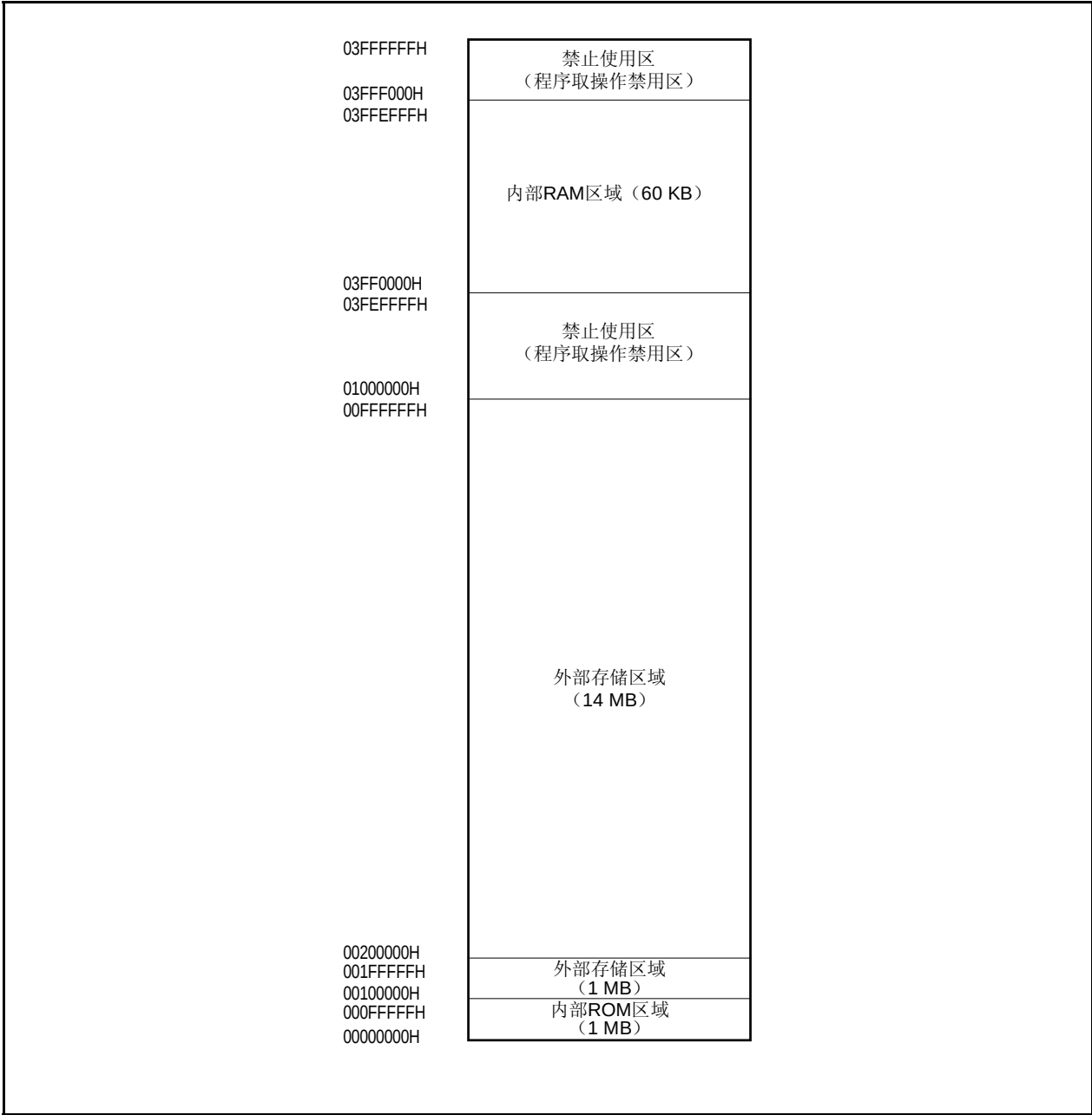


图 3-3. 程序内存映射



3.4.4 存储区域

(1) 内部 ROM 区域

多达 1 MB 的区域保留作为内部 ROM 区域。

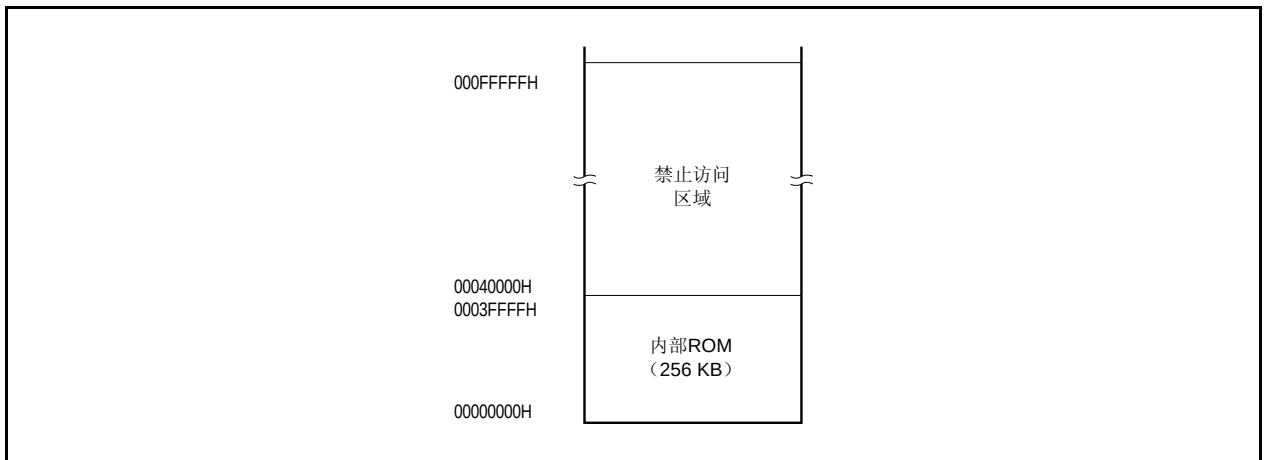
(a) 内部 ROM (256 KB)

在下列产品中，256 KB 的 ROM 分配给地址 00000000H 至 0003FFFFH。

禁止访问地址 00040000H 至 000FFFFFFH。

- μ PD70F3760、70F3770、70F3765、70F3771

图 3-4. 内部 ROM 区域 (256 KB)



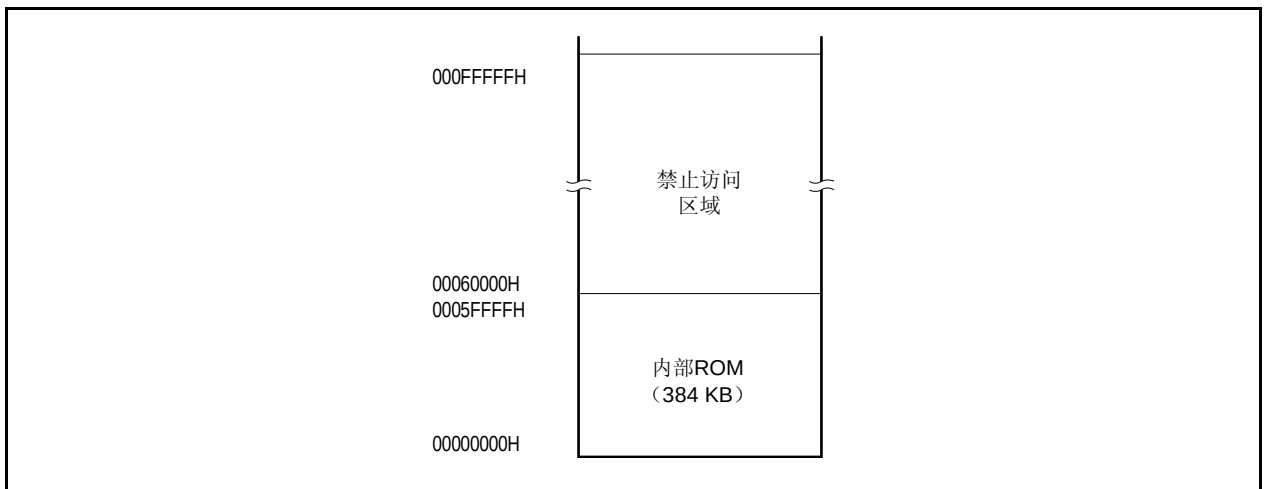
(b) 内部 ROM (384 KB)

在下列产品中，384 KB 的 ROM 分配给地址 00000000H 至 0005FFFFH。

禁止访问地址 00060000H 至 000FFFFFFH。

- μ PD70F3761、70F3766

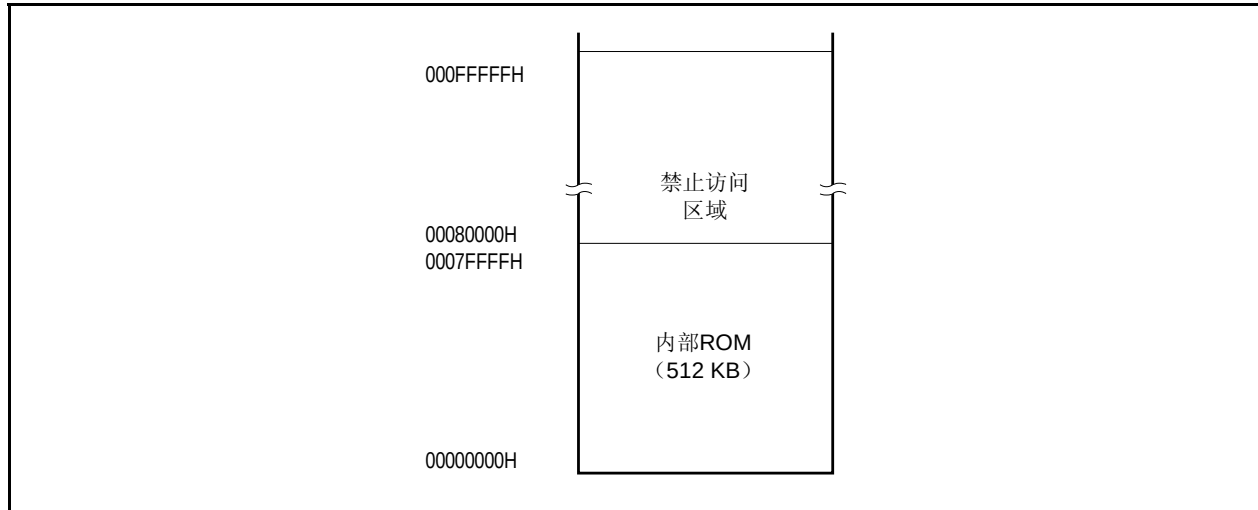
图 3-5. 内部 ROM 区域 (384 KB)



(c) 内部 ROM (512 KB)

在下列产品中，512 KB 的 ROM 分配给了地址 00000000H 至 0007FFFFH。
禁止访问地址 00080000H 至 000FFFFFFH。

- μ PD70F3762、70F3767

图 3-6. 内部 ROM 区域 (512 KB)

(2) 内部 RAM 区域

多达 60 KB 的区域保留作为内部 RAM 区域。
除了内部 RAM 外，V850ES/JG3-H 和 V850ES/JH3-H 单片机还包括 8 KB 仅数据的 RAM 区。
V850ES/JG3-H 和 V850ES/JH3-H 的 RAM 容量如下所示。

表 3-3 RAM 区域

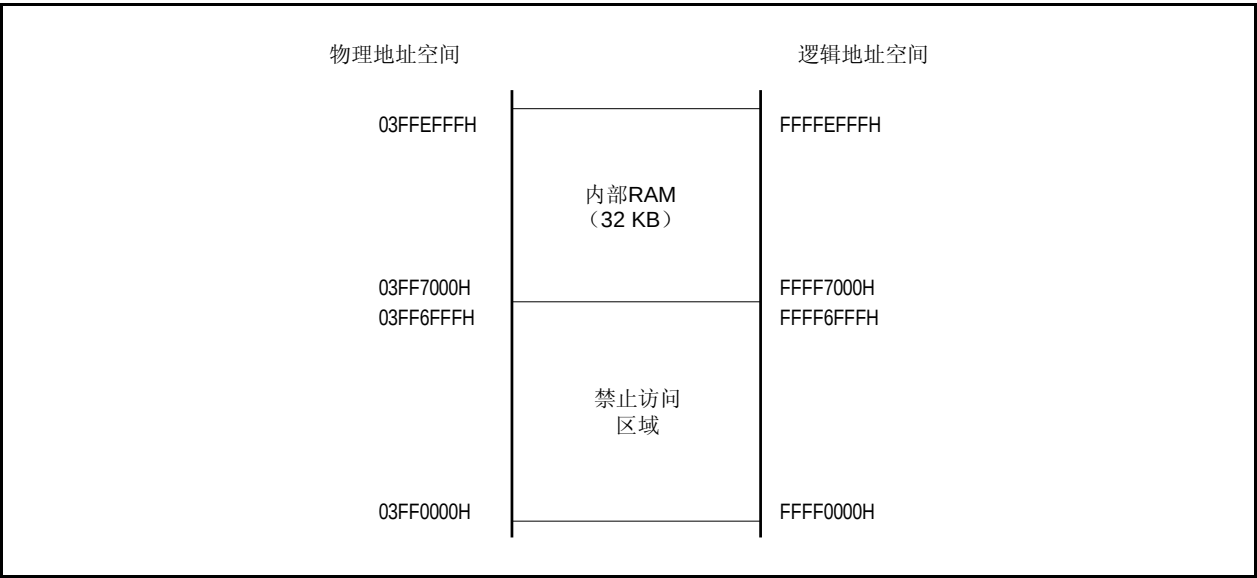
通称	产品名称	内部RAM	仅数据RAM	总RAM
V850ES/JG3-H	μPD70F3760、70F3770	32 KB	8 KB	40 KB
	μPD70F3761	40 KB		48 KB
	μPD70F3762	48 KB		56 KB
V850ES/JH3-H	μPD70F3765、70F3771	32 KB		40 KB
	μPD70F3766	40 KB		48 KB
	μPD70F3767	48 KB		56KB

(a) 内部 RAM（32 KB）

在下列产品中，32 KB 的 RAM 分配给了地址 03FF7000H 至 03FF6FFFH。
禁止访问地址 03FF0000H 至 03FF6FFFH。

- μPD70F3760、70F3770、70F3765、70F3771

图 3-7. 内部 RAM 区域（32 KB）

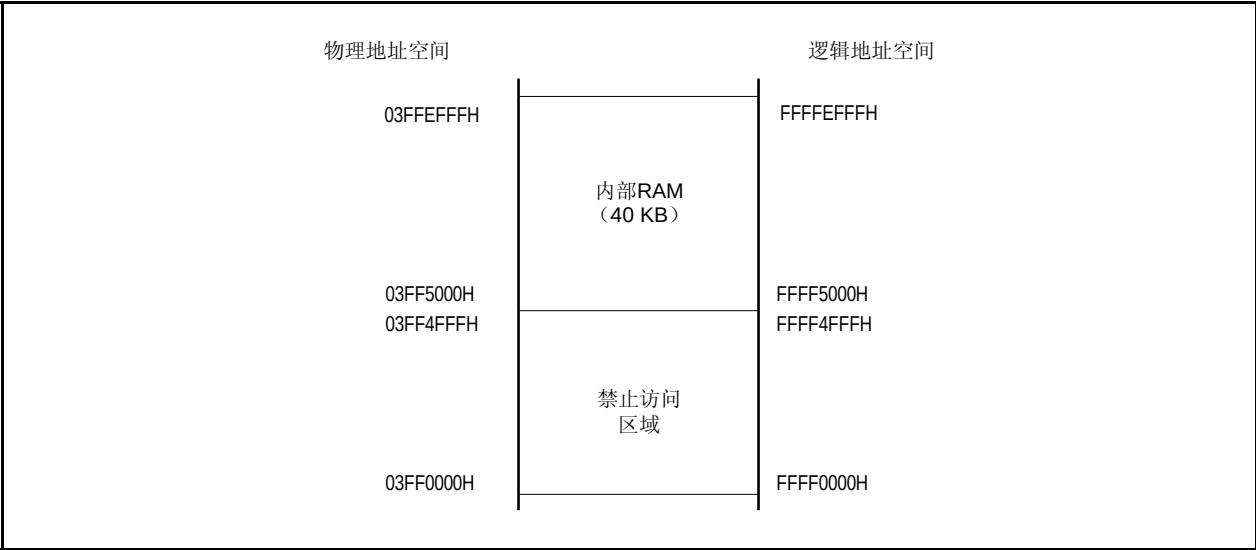


(b) 内部 RAM（40 KB）

在下列产品中，40 KB 的 RAM 分配给了地址 03FF5000H 至 03FFFFFFFFH。
禁止访问地址 03FF0000H 至 03FF4FFFFH。

- μ PD70F3761、70F3766

图 3-8. 内部 RAM 区域（40 KB）

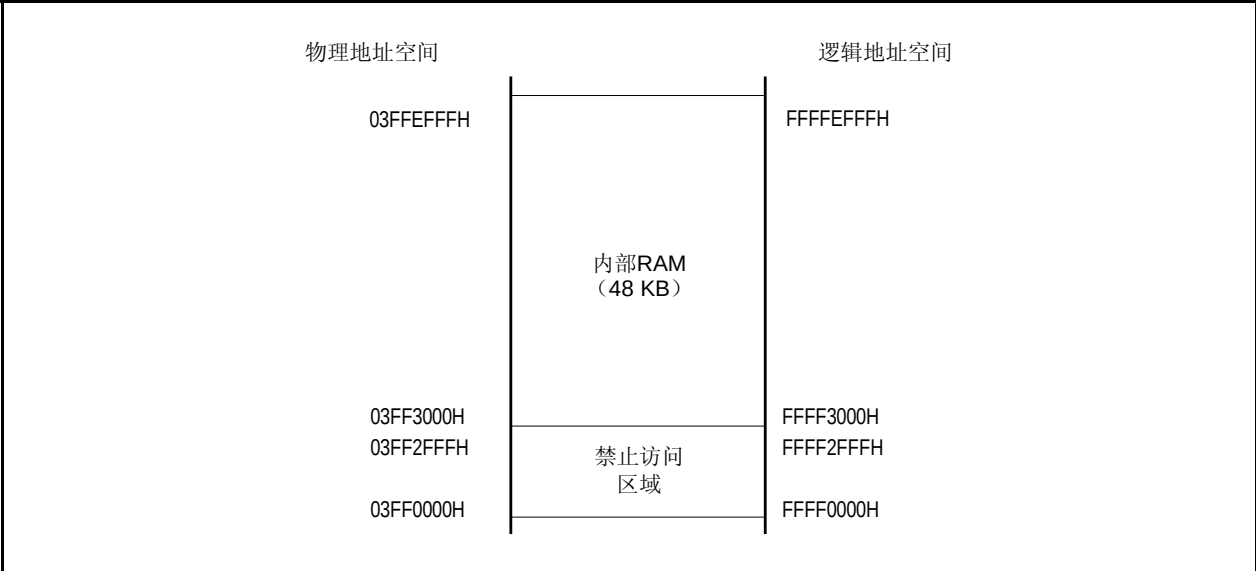


(c) 内部 RAM（48 KB）

在下列产品中，48 KB 的 RAM 分配给了地址 03FF3000H 至 03FFFFFFFFH。
禁止访问地址 03FF0000H 至 03FF2FFFFH。

- μ PD70F3762、70F3767

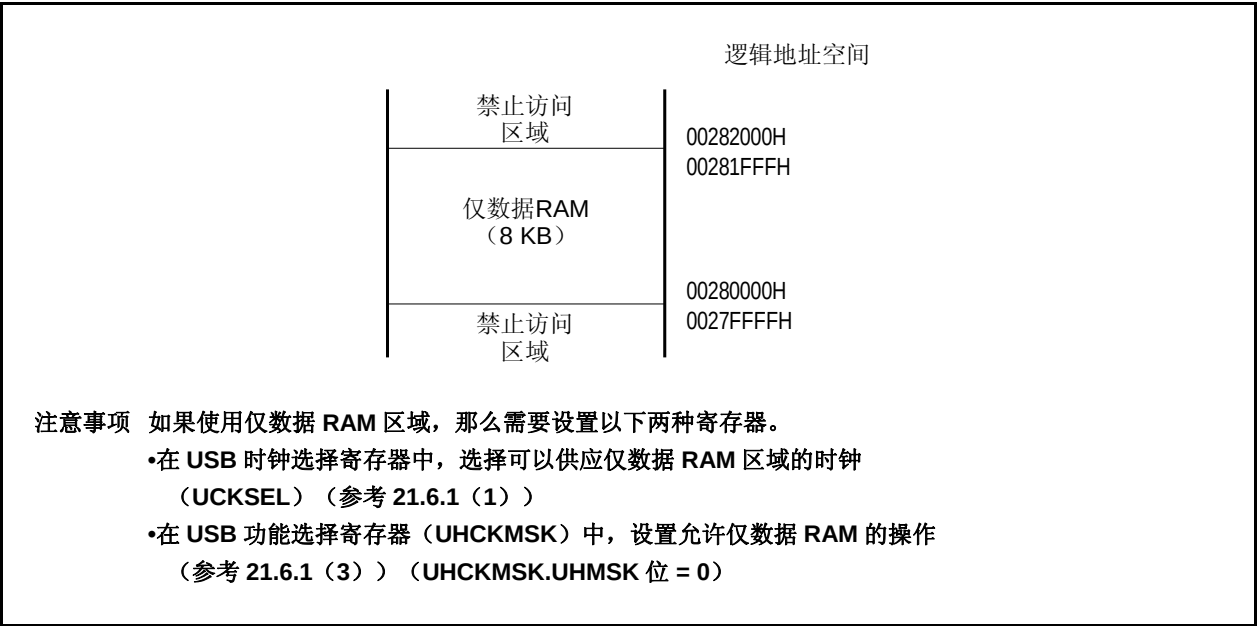
图 3-9. 内部 RAM 区域（48 KB）



- <R>
- (d) 仅数据 RAM (8 KB)

在 V850ES/JG3-H 和 V850ES/JH3-H 单片机中，8 KB 仅数据 RAM 分配给了地址 00280000H 至 00281FFFH。

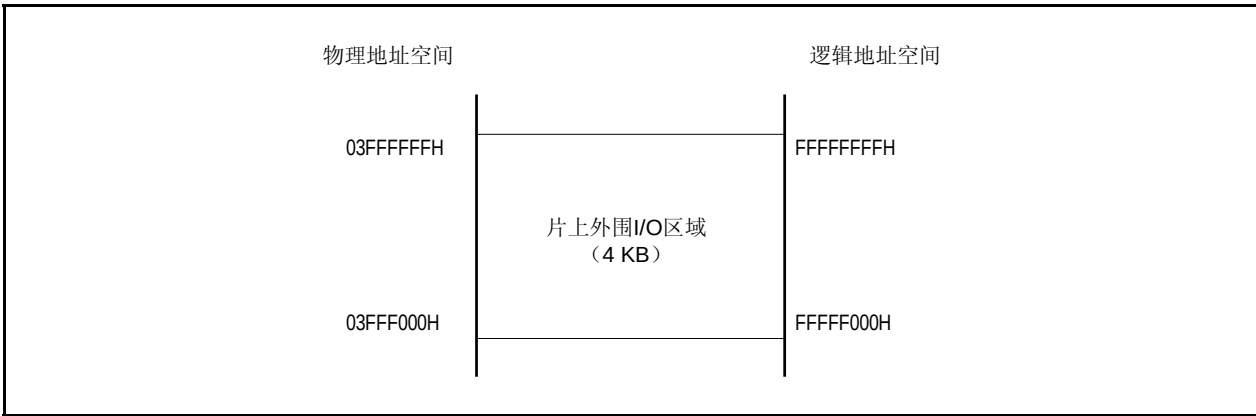
图 3-10. 仅数据 RAM 区域 (8 KB)



(3) 片上外围 I/O 区域

地址 03FFF000H 至 03FFFFFFH 的 4 KB 存储区域保留作为片上外围 I/O 区域。

图 3-11. 片上外围 I/O 区域



具有设定操作模式并监控其片上外设 I/O 状态的外设 I/O 寄存器映射到片内 I/O 区域。不能从该区域中取程序。

- 注意事项
1. 当寄存器以字为单元进行访问时，则会按照区域先低后高的顺序，以半字为单元对每个字区域访问两次，其中忽略低 2 位地址。
 2. 能够以字节为单位进行访问的寄存器，如果以半字为单位进行访问时，当访问操作是读取操作，那么该寄存器的高 8 位将不确定，写数据时，只写入低 8 位。
 3. 没有定义为寄存器的地址区域保留为将来扩展之用。如果对这些地址进行访问，其操作不确定而且也不保证操作的有效性。
 4. 内部 ROM/RAM 区域和片上外围 I/O 区域分配给连续地址。
使用指针或类似的方式增加或减小地址来访问内部 ROM/RAM 区域时，一定要注意不要错误地越过内部 ROM/RAM 区域边界去访问片内外围 I/O 区域。

<R> (4) 外部存储区

13 MB 区域（00100000H 至 001FFFFFFH，00400000H 至 00FFFFFFH）分配为外部存储区。详情参见第五章总线控制功能。

3.4.5 地址空间使用建议

V850ES/JG3-H 和 V850ES/JH3-H 的体系结构要求：当访问数据空间的操作数时，必须确保有一个用作指针的寄存器以生成地址。存储于该±32 KB 指针中的地址可以直接由操作数指令存取。然而，用作指针的通用寄存器数量受到限制，可以保证尽可能多的通用寄存器用于变量，所以在地址运算期间指针的数值改变时，可免于性能下降，并能减少程序的长度。

(1) 程序空间

对于 32 位的 PC（程序计数器），其高 6 位固定为 0，且只有低 26 位有效。因此，对于程序空间，从地址 00000000H 开始的 64MB 连续地址空间与存储器映射是无条件对应的。

要将内部 RAM 用作程序空间，请访问下列地址空间。

注意事项 如果跳转指令发生在内部 RAM 区间的上限地址处，那么将不会产生跨越片上外围 I/O 区域中的预取址操作（无效取址）。

RAM 容量	访问地址
48 KB	03FF3000H 至 03FFEFFFFH
40 KB	03FF5000H 至 03FFEFFFFH
32 KB	03FF7000H 至 03FFEFFFFH

(2) 数据空间

对于 V850ES/JG3-H 和 V850ES/JH3-H 来说，可看作是在 4GB 的 CPU 地址空间内有 64 个 64MB 的地址空间。因此，26 位地址的最高有效位（第 25 位）是带符号扩展成 32 位并用作地址值的。

(a) 地址环绕应用举例

如果将 R=r0（零寄存器）指定给 LD/ST disp16[R]指令，那么就可以通过带符号扩展的 disp16 对从 00000000H 开始的±32KB 地址范围空间进行寻址。通过一个指针就可以对包括内部硬件在内的所有资源进行寻址。

零寄存器（r0）是由硬件将其值固定为 0 的寄存器，而且不需要专门用作指针的寄存器。

例如：μPD70F3767

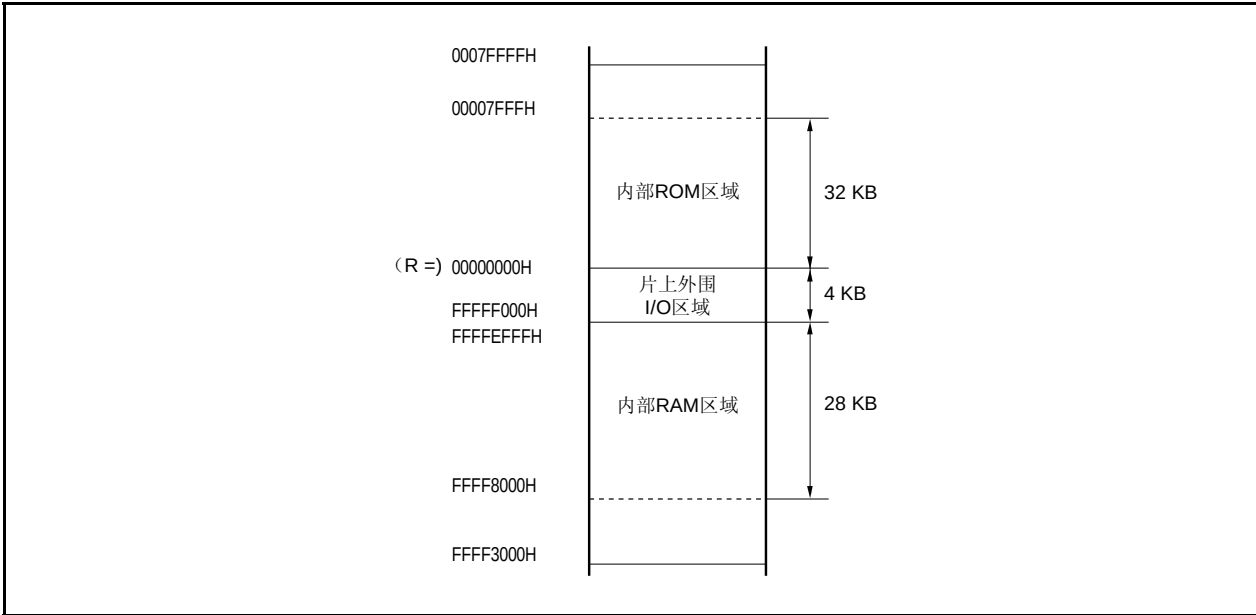
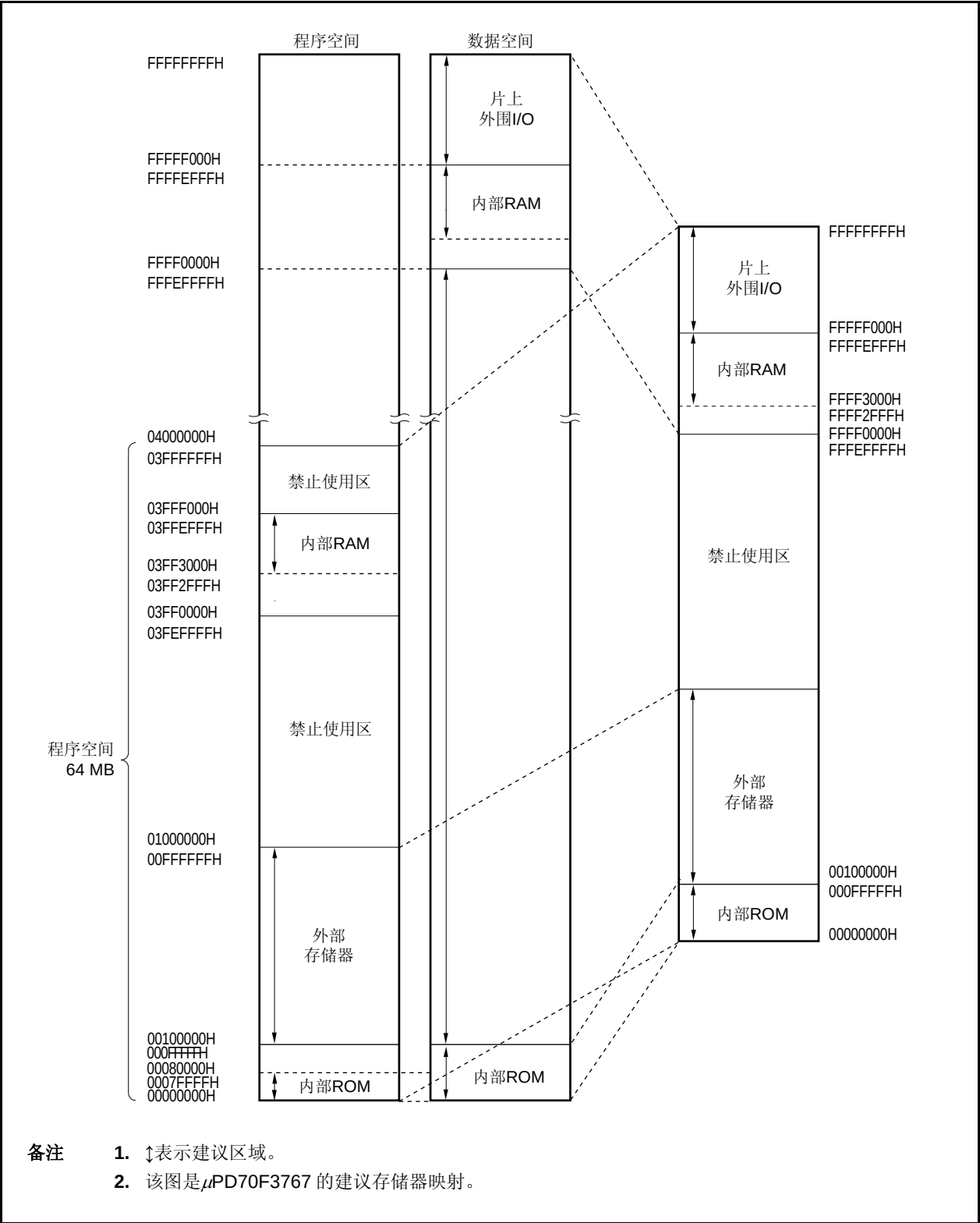


图 3-12. 建议存储器映射



3.4.6 外设 I/O 寄存器

(1/14)

	地址	功能寄存器名称	符号	R/W	可操作位			默认值
					1	8	16	
<R>	FFFFF004H	DL 端口寄存器	PDL	R/W			√	0000H ^{注1}
	FFFFF004H	DL 端口寄存器 L	PDLL		√	√		00H ^{注1}
	FFFFF005H	DL 端口寄存器 H	PDLH		√	√		00H ^{注1}
	FFFFF006H	DH 端口寄存器 ^{注2}	PDH ^{注2}		√	√		00H ^{注1}
	FFFFF008H	CS 端口寄存器 ^{注2}	PCS ^{注2}		√	√		00H ^{注1}
	FFFFF00AH	CT 端口寄存器	PCT		√	√		00H ^{注1}
	FFFFF00CH	CM 端口寄存器	PCM		√	√		00H ^{注1}
	FFFFF024H	DL 端口模式寄存器	PMDL				√	FFFFH
<R>	FFFFF024H	DL 端口模式寄存器 L	PMDLL		√	√		FFH
	FFFFF025H	DL 端口模式寄存器 H	PMDLH		√	√		FFH
	FFFFF026H	DH 端口模式寄存器 ^{注2}	PMDH ^{注2}		√	√		FFH
	FFFFF028H	CS 端口模式寄存器 ^{注2}	PMCS ^{注2}		√	√		FFH
	FFFFF02AH	CT 端口模式寄存器	PMCT		√	√		FFH
	FFFFF02CH	CM 端口模式寄存器	PMCM		√	√		FFH
	FFFFF044H	DL 端口模式控制寄存器	PMCDL				√	0000H
	FFFFF044H	DL 端口模式控制寄存器 L	PMCDLL		√	√		00H
<R>	FFFFF045H	DL 端口模式控制寄存器 H	PMCDLH		√	√		00H
	FFFFF046H	DH 端口模式控制寄存器 ^{注2}	PMCDH ^{注2}		√	√		00H
	FFFFF048H	CS 端口模式控制寄存器 ^{注2}	PMCCS ^{注2}		√	√		00H
	FFFFF04AH	CT 端口模式控制寄存器	PM CCT		√	√		00H
	FFFFF04CH	CM 端口模式控制寄存器	PMCCM		√	√		00H
	FFFFF064H	外围 I/O 区域选择控制寄存器 ^{注3}	BPC ^{注3}				√	0000H
	FFFFF066H	总线宽度设置寄存器	BSC				√	5555H
	FFFFF06EH	系统等待控制寄存器	VSWC			√		77H
	FFFFF080H	DMA 源地址寄存器 0L	DSA0L				√	不确定
	FFFFF082H	DMA 源地址寄存器 0H	DSA0H				√	不确定
	FFFFF084H	DMA 目的地址寄存器 0L	DDA0L				√	不确定
	FFFFF086H	DMA 目的地址寄存器 0H	DDA0H				√	不确定
	FFFFF088H	DMA 源地址寄存器 1L	DSA1L				√	不确定
	FFFFF08AH	DMA 源地址寄存器 1H	DSA1H				√	不确定
	FFFFF08CH	DMA 目的地址寄存器 1L	DDA1L				√	不确定
	FFFFF08EH	DMA 目的地址寄存器 1H	DDA1H				√	不确定
	FFFFF090H	DMA 源地址寄存器 2L	DSA2L				√	不确定
	FFFFF092H	DMA 源地址寄存器 2H	DSA2H				√	不确定
	FFFFF094H	DMA 目的地址寄存器 2L	DDA2L				√	不确定
	FFFFF096H	DMA 目的地址寄存器 2H	DDA2H				√	不确定
	FFFFF098H	DMA 源地址寄存器 3L	DSA3L				√	不确定
	FFFFF09AH	DMA 源地址寄存器 3H	DSA3H				√	不确定
	FFFFF09CH	DMA 目的地址寄存器 3L	DDA3L				√	不确定
	FFFFF09EH	DMA 目的地址寄存器 3H	DDA3H				√	不确定

注 1 输出锁存为 00H 或 0000H。当这些寄存器在输入模式下时，引脚状态会被读取。

2. 仅适用于 V850ES/JH3-H

3. 仅适用于 μ PD70F3770、70F3771

(2/14)

地址	功能寄存器名称	符号	R/W	可操作位			默认值
				1	8	16	
FFFFF0C0H	DMA 转移计数寄存器 0	DBC0	R/W			√	不确定
FFFFF0C2H	DMA 转移计数寄存器 1	DBC1				√	不确定
FFFFF0C4H	DMA 转移计数寄存器 2	DBC2				√	不确定
FFFFF0C6H	DMA 转移计数寄存器 3	DBC3				√	不确定
FFFFF0D0H	DMA 寻址控制寄存器 0	DADC0				√	0000H
FFFFF0D2H	DMA 寻址控制寄存器 1	DADC1				√	0000H
FFFFF0D4H	DMA 寻址控制寄存器 2	DADC2				√	0000H
FFFFF0D6H	DMA 寻址控制寄存器 3	DADC3				√	0000H
FFFFF0E0H	DMA 通道控制寄存器 0	DCHC0		√	√		00H
FFFFF0E2H	DMA 通道控制寄存器 1	DCHC1		√	√		00H
FFFFF0E4H	DMA 通道控制寄存器 2	DCHC2		√	√		00H
FFFFF0E6H	DMA 通道控制寄存器 3	DCHC3		√	√		00H
FFFFF100H	中断屏蔽寄存器 0	IMR0				√	FFFFH
FFFFF100H	中断屏蔽寄存器 0L	IMR0L		√	√		FFH
FFFFF101H	中断屏蔽寄存器 0H	IMR0H		√	√		FFH
FFFFF102H	中断屏蔽寄存器 1	IMR1				√	FFFFH
FFFFF102H	中断屏蔽寄存器 1L	IMR1L		√	√		FFH
FFFFF103H	中断屏蔽寄存器 1H	IMR1H		√	√		FFH
FFFFF104H	中断屏蔽寄存器 2	IMR2				√	FFFFH
FFFFF104H	中断屏蔽寄存器 2L	IMR2L		√	√		FFH
FFFFF105H	中断屏蔽寄存器 2H	IMR2H		√	√		FFH
FFFFF106H	中断屏蔽寄存器 3	IMR3				√	FFFFH
FFFFF106H	中断屏蔽寄存器 3L	IMR3L		√	√		FFH
FFFFF107H	中断屏蔽寄存器 3H	IMR3H		√	√		FFH
FFFFF108H	中断屏蔽寄存器 4	IMR4				√	FFFFH
FFFFF108H	中断屏蔽寄存器 4L	IMR4L		√	√		FFH
FFFFF109H	中断屏蔽寄存器 4H	IMR4H		√	√		FFH
FFFFF10AH	中断屏蔽寄存器 5	IMR5				√	FFFFH
FFFFF10AH	中断屏蔽寄存器 5L	IMR5L		√	√		FFH
FFFFF10BH	中断屏蔽寄存器 5H	IMR5H		√	√		FFH
FFFFF110H	中断控制寄存器	LVIIC		√	√		47H
FFFFF112H	中断控制寄存器	PIC00		√	√		47H
FFFFF114H	中断控制寄存器	PIC01		√	√		47H
FFFFF116H	中断控制寄存器	PIC02		√	√		47H
FFFFF118H	中断控制寄存器	PIC03		√	√		47H
FFFFF11AH	中断控制寄存器	PIC04		√	√		47H
FFFFF11CH	中断控制寄存器	PIC05		√	√		47H
FFFFF11EH	中断控制寄存器	PIC06		√	√		47H
FFFFF120H	中断控制寄存器	PIC07		√	√		47H
FFFFF122H	中断控制寄存器	PIC08		√	√		47H
FFFFF124H	中断控制寄存器	PIC09		√	√		47H

(3/14)

地址	功能寄存器名称	符号	R/W	可操作位			默认值
				1	8	16	
FFFFF126H	中断控制寄存器	PIC10	R/W	√	√		47H
FFFFF128H	中断控制寄存器	PIC11		√	√		47H
FFFFF12AH	中断控制寄存器	PIC12		√	√		47H
FFFFF12CH	中断控制寄存器	PIC13		√	√		47H
FFFFF12EH	中断控制寄存器	PIC14		√	√		47H
FFFFF130H	中断控制寄存器	PIC15		√	√		47H
FFFFF132H	中断控制寄存器	PIC16		√	√		47H
FFFFF134H	中断控制寄存器	PIC17		√	√		47H
FFFFF136H	中断控制寄存器	PIC18		√	√		47H
FFFFF138H	中断控制寄存器	TAB0OVIC		√	√		47H
FFFFF13AH	中断控制寄存器	TAB0CCIC0		√	√		47H
FFFFF13CH	中断控制寄存器	TAB0CCIC1		√	√		47H
FFFFF13EH	中断控制寄存器	TAB0CCIC2		√	√		47H
FFFFF140H	中断控制寄存器	TAB0CCIC3		√	√		47H
FFFFF142H	中断控制寄存器	TAB1OVIC		√	√		47H
FFFFF144H	中断控制寄存器	TAB1CCIC0		√	√		47H
FFFFF146H	中断控制寄存器	TAB1CCIC1		√	√		47H
FFFFF148H	中断控制寄存器	TAB1CCIC2		√	√		47H
FFFFF14AH	中断控制寄存器	TAB1CCIC3		√	√		47H
FFFFF14CH	中断控制寄存器	TT0OVIC		√	√		47H
FFFFF14EH	中断控制寄存器	TT0CCIC0		√	√		47H
FFFFF150H	中断控制寄存器	TT0CCIC1		√	√		47H
FFFFF152H	中断控制寄存器	TT0IECIC		√	√		47H
FFFFF154H	中断控制寄存器	TAA0OVIC		√	√		47H
FFFFF156H	中断控制寄存器	TAA0CCIC0		√	√		47H
FFFFF158H	中断控制寄存器	TAA0CCIC1		√	√		47H
FFFFF15AH	中断控制寄存器	TAA1OVIC		√	√		47H
FFFFF15CH	中断控制寄存器	TAA1CCIC0		√	√		47H
FFFFF15EH	中断控制寄存器	TAA1CCIC1		√	√		47H
FFFFF160H	中断控制寄存器	TAA2OVIC		√	√		47H
FFFFF162H	中断控制寄存器	TAA2CCIC0		√	√		47H
FFFFF164H	中断控制寄存器	TAA2CCIC1		√	√		47H
FFFFF166H	中断控制寄存器	TAA3OVIC		√	√		47H
FFFFF168H	中断控制寄存器	TAA3CCIC0		√	√		47H
FFFFF16AH	中断控制寄存器	TAA3CCIC1		√	√		47H
FFFFF16CH	中断控制寄存器	TAA4OVIC		√	√		47H
FFFFF16EH	中断控制寄存器	TAA4CCIC0		√	√		47H
FFFFF170H	中断控制寄存器	TAA4CCIC1		√	√		47H
FFFFF172H	中断控制寄存器	TAA5OVIC		√	√		47H
FFFFF174H	中断控制寄存器	TAA5CCIC0		√	√		47H
FFFFF176H	中断控制寄存器	TAA5CCIC1		√	√		47H
FFFFF178H	中断控制寄存器	TM0EQIC0		√	√		47H
FFFFF17AH	中断控制寄存器	TM1EQIC0		√	√		47H

(4/14)

地址	功能寄存器名称	符号	R/W	可操作位			默认值
				1	8	16	
FFFFF17CH	中断控制寄存器	TM2EQIC0	R/W	√	√		47H
FFFFF17EH	中断控制寄存器	TM3EQIC0		√	√		47H
FFFFF180H	中断控制寄存器	CF0RIC/IICIC1		√	√		47H
FFFFF182H	中断控制寄存器	CF0TIC		√	√		47H
FFFFF184H	中断控制寄存器	CF1RIC		√	√		47H
FFFFF186H	中断控制寄存器	CF1TIC		√	√		47H
FFFFF188H	中断控制寄存器	CF2RIC		√	√		47H
FFFFF18AH	中断控制寄存器	CF2TIC		√	√		47H
FFFFF18CH	中断控制寄存器	CF3RIC		√	√		47H
FFFFF18EH	中断控制寄存器	CF3TIC		√	√		47H
FFFFF190H	中断控制寄存器	CF4RIC		√	√		47H
FFFFF192H	中断控制寄存器	CF4TIC		√	√		47H
FFFFF194H	中断控制寄存器	UC0RIC		√	√		47H
FFFFF196H	中断控制寄存器	UC0TIC		√	√		47H
FFFFF198H	中断控制寄存器	UC1RIC/IICIC2		√	√		47H
FFFFF19AH	中断控制寄存器	UC1TIC		√	√		47H
FFFFF19CH	中断控制寄存器	UC2RIC		√	√		47H
FFFFF19EH	中断控制寄存器	UC2TIC		√	√		47H
FFFFF1A0H	中断控制寄存器	UC3RIC/IICIC0		√	√		47H
FFFFF1A2H	中断控制寄存器	UC3TIC		√	√		47H
FFFFF1A4H	中断控制寄存器	UC4RIC		√	√		47H
FFFFF1A6H	中断控制寄存器	UC4TIC		√	√		47H
FFFFF1A8H	中断控制寄存器	ADIC		√	√		47H
FFFFF1AAH	中断控制寄存器	DMAIC0		√	√		47H
FFFFF1ACH	中断控制寄存器	DMAIC1		√	√		47H
FFFFF1AEH	中断控制寄存器	DMAIC2		√	√		47H
FFFFF1B0H	中断控制寄存器	DMAIC3		√	√		47H
FFFFF1B2H	中断控制寄存器	KRIC		√	√		47H
FFFFF1B4H	中断控制寄存器	RTC0IC		√	√		47H
FFFFF1B6H	中断控制寄存器	RTC1IC		√	√		47H
FFFFF1B8H	中断控制寄存器	RTC2IC		√	√		47H
FFFFF1BAH	中断控制寄存器	ERRIC0n [※]		√	√		47H
FFFFF1BCH	中断控制寄存器	WUPIC0 [※]		√	√		47H
FFFFF1BEH	中断控制寄存器	RECIC0 [※]		√	√		47H
FFFFF1C0H	中断控制寄存器	TRXIC0 [※]		√	√		47H
FFFFF1C8H	中断控制寄存器	UFIC0		√	√		47H
FFFFF1CAH	中断控制寄存器	UFIC1		√	√		47H
FFFFF1FAH	中断服务优先级寄存器	ISPR	R	√	√		00H
FFFFF1FCH	命令寄存器	PRCMD	W		√		不确定
FFFFF1FEH	节能控制寄存器	PSC	R/W	√	√		00H
FFFFF200H	A/D 转换器模式寄存器 0	ADA0M0		√	√		00H
FFFFF201H	A/D 转换器模式寄存器 1	ADA0M1		√	√		00H

注 仅适用于μPD70F3770、70F3771

(5/14)

地址	功能寄存器名称	符号	R/W	可操作位			默认值
				1	8	16	
FFFFF202H	A/D 转换器通道指定寄存器	ADA0S	R/W	√	√		00H
FFFFF203H	A/D 转换模式寄存器 2	ADA0M2		√	√		00H
FFFFF204H	电源掉电比较模式寄存器	ADA0PFM		√	√		00H
FFFFF205H	电源掉电比较阈值寄存器	ADA0PFT		√	√		00H
FFFFF210H	A/D 转换结果寄存器 0	ADA0CR0	R			√	不确定
FFFFF211H	A/D 转换结果寄存器 0H	ADA0CR0H			√		不确定
FFFFF212H	A/D 转换结果寄存器 1	ADA0CR1				√	不确定
FFFFF213H	A/D 转换结果寄存器 1H	ADA0CR1H			√		不确定
FFFFF214H	A/D 转换结果寄存器 2	ADA0CR2				√	不确定
FFFFF215H	A/D 转换结果寄存器 2H	ADA0CR2H			√		不确定
FFFFF216H	A/D 转换结果寄存器 3	ADA0CR3				√	不确定
FFFFF217H	A/D 转换结果寄存器 3H	ADA0CR3H			√		不确定
FFFFF218H	A/D 转换结果寄存器 4	ADA0CR4				√	不确定
FFFFF219H	A/D 转换结果寄存器 4H	ADA0CR4H			√		不确定
FFFFF21AH	A/D 转换结果寄存器 5	ADA0CR5				√	不确定
FFFFF21BH	A/D 转换结果寄存器 5H	ADA0CR5H			√		不确定
FFFFF21CH	A/D 转换结果寄存器 6	ADA0CR6				√	不确定
FFFFF21DH	A/D 转换结果寄存器 6H	ADA0CR6H			√		不确定
FFFFF21EH	A/D 转换结果寄存器 7	ADA0CR7				√	不确定
FFFFF21FH	A/D 转换结果寄存器 7H	ADA0CR7H			√		不确定
FFFFF220H	A/D 转换结果寄存器 8	ADA0CR8				√	不确定
FFFFF221H	A/D 转换结果寄存器 8H	ADA0CR8H			√		不确定
FFFFF222H	A/D 转换结果寄存器 9	ADA0CR9				√	不确定
FFFFF223H	A/D 转换结果寄存器 9H	ADA0CR9H			√		不确定
FFFFF224H	A/D 转换结果寄存器 10	ADA0CR10				√	不确定
FFFFF225H	A/D 转换结果寄存器 10H	ADA0CR10H			√		不确定
FFFFF226H	A/D 转换结果寄存器 11	ADA0CR11				√	不确定
FFFFF227H	A/D 转换结果寄存器 11H	ADA0CR11H			√		不确定
FFFFF280H	D/A 转换值设置寄存器 0	DA0CS0	R/W		√		00H
FFFFF281H	D/A 转换值设置寄存器 1	DA0CS1			√		00H
FFFFF282H	D/A 转换模式寄存器	DA0M		√	√		00H
FFFFF300H	按键返回模式寄存器	KRM		√	√		00H
FFFFF308H	选择器操作控制寄存器 0	SELCNT0		√	√		00H
FFFFF310H	CRC 输入寄存器	CRCIN			√		00H
FFFFF312H	CRC 数据寄存器	CRCD				√	0000H
FFFFF320H	预分频器模式寄存器 1	PRSM1		√	√		00H
FFFFF321H	预分频器比较寄存器 1	PRSCM1			√		00H
FFFFF324H	预分频器模式寄存器 2	PRSM2		√	√		00H
FFFFF325H	预分频器比较寄存器 2	PRSCM2			√		00H
FFFFF328H	预分频器模式寄存器 3	PRSM3		√	√		00H
FFFFF329H	预分频器比较寄存器 3	PRSCM3			√		00H
FFFFF340H	IIC 分频时钟选择寄存器 0	OCKS0			√		00H
FFFFF344H	IIC 分频时钟选择寄存器 1	OCKS1			√		00H

(6/14)

地址	功能寄存器名称	符号	R/W	可操作位			默认值
				1	8	16	
FFFFF400H	0 端口寄存器	P0	R/W	√	√		00H ^{注1}
FFFFF402H	1 端口寄存器	P1		√	√		00H ^{注1}
FFFFF404H	2 端口寄存器 ^{注2}	P2 ^{注2}		√	√		00H ^{注1}
FFFFF406H	3 端口寄存器	P3		√	√		00H ^{注1}
FFFFF408H	4 端口寄存器	P4		√	√		00H ^{注1}
FFFFF40AH	5 端口寄存器	P5		√	√		00H ^{注1}
FFFFF40CH	6 端口寄存器	P6		√	√		00H ^{注1}
FFFFF40EH	7 端口寄存器 L	P7L		√	√		00H ^{注1}
FFFFF40FH	7 端口寄存器 H	P7H		√	√		00H ^{注1}
FFFFF412H	9 端口寄存器	P9				√	0000H ^{注1}
FFFFF412H	9 端口寄存器 L	P9L		√	√		00H ^{注1}
FFFFF413H	9 端口寄存器 H	P9H		√	√		00H ^{注1}
FFFFF420H	0 端口模式寄存器	PM0		√	√		FFH
FFFFF422H	1 端口模式寄存器	PM1		√	√		FFH
FFFFF424H	2 端口模式寄存器 ^{注2}	PM2 ^{注2}		√	√		FFH
FFFFF426H	3 端口模式寄存器	PM3		√	√		FFH
FFFFF428H	4 端口模式寄存器	PM4		√	√		FFH
FFFFF42AH	5 端口模式寄存器	PM5		√	√		FFH
FFFFF42CH	6 端口模式寄存器	PM6		√	√		FFH
FFFFF42EH	7 端口模式寄存器 L	PM7L		√	√		FFH
FFFFF42FH	7 端口模式寄存器 H	PM7H		√	√		FFH
FFFFF432H	9 端口模式寄存器	PM9				√	FFFFH
FFFFF432H	9 端口模式寄存器 L	PM9L		√	√		FFH
FFFFF433H	9 端口模式寄存器 H	PM9H		√	√		FFH
FFFFF440H	0 端口模式控制寄存器	PMC0		√	√		00H
FFFFF444H	2 端口模式控制寄存器 ^{注2}	PMC2 ^{注2}		√	√		00H
FFFFF446H	3 端口模式控制寄存器	PMC3		√	√		00H
FFFFF448H	4 端口模式控制寄存器	PMC4		√	√		00H
FFFFF44AH	5 端口模式控制寄存器	PMC5		√	√		00H
FFFFF44CH	6 端口模式控制寄存器	PMC6		√	√		00H
FFFFF452H	9 端口模式控制寄存器	PMC9				√	0000H
FFFFF452H	9 端口模式控制寄存器 L	PMC9L		√	√		00H
FFFFF453H	9 端口模式控制寄存器 H	PMC9H		√	√		00H
FFFFF460H	0 端口功能控制寄存器	PFC0		√	√		00H
FFFFF464H	2 端口功能控制寄存器 ^{注2}	PFC2 ^{注2}		√	√		00H
FFFFF466H	3 端口功能控制寄存器	PFC3		√	√		00H
FFFFF468H	4 端口功能控制寄存器	PFC4		√	√		00H
FFFFF46AH	5 端口功能控制寄存器	PFC5		√	√		00H
FFFFF46CH	6 端口功能控制寄存器	PFC6		√	√		00H

注 1 输出锁存为 00H 或 0000H。当输入这些寄存器时，该引脚状态会被读取。

2. 只适用于 V850ES/JH3-H

(7/14)

地址	功能寄存器名称	符号	R/W	可操作位			默认值
				1	8	16	
FFFFF472H	9 端口功能控制寄存器	PFC9	R/W			√	0000H
FFFFF472H	9 端口功能控制寄存器 L	PFC9L		√	√		00H
FFFFF473H	9 端口功能控制寄存器 H	PFC9H		√	√		00H
FFFFF484H	数据等待控制寄存器 0	DWC0				√	7777H
FFFFF488H	地址等待控制寄存器	AWC				√	FFFFH
FFFFF48AH	总线周期控制寄存器	BCC				√	AAAAH
FFFFF540H	TAB0 控制寄存器 0	TAB0CTL0		√	√		00H
FFFFF541H	TAB0 控制寄存器 1	TAB0CTL1		√	√		00H
FFFFF542H	TAB0 I/O 控制寄存器 0	TAB0IOC0		√	√		00H
FFFFF543H	TAB0 I/O 控制寄存器 1	TAB0IOC1		√	√		00H
FFFFF544H	TAB0 I/O 控制寄存器 2	TAB0IOC2		√	√		00H
FFFFF545H	TAB0 选项寄存器 0	TAB0OPT0		√	√		00H
FFFFF546H	TAB0 捕获/比较寄存器 0	TAB0CCR0				√	0000H
FFFFF548H	TAB0 捕获/比较寄存器 1	TAB0CCR1				√	0000H
FFFFF54AH	TAB0 捕获/比较寄存器 2	TAB0CCR2				√	0000H
FFFFF54CH	TAB0 捕获/比较寄存器 3	TAB0CCR3				√	0000H
FFFFF54EH	TAB0 计数器读缓冲寄存器	TAB0CNT	R			√	0000H
FFFFF550H	TAB0 I/O 控制寄存器 4	TAB0IOC4	R/W	√	√		00H
FFFFF560H	TAB1 控制寄存器 0	TAB1CTL0		√	√		00H
FFFFF561H	TAB1 控制寄存器 1	TAB1CTL1		√	√		00H
FFFFF562H	TAB1 I/O 控制寄存器 0	TAB1IOC0		√	√		00H
FFFFF563H	TAB1 I/O 控制寄存器 1	TAB1IOC1		√	√		00H
FFFFF564H	TAB1 I/O 控制寄存器 2	TAB1IOC2		√	√		00H
FFFFF565H	TAB1 选项寄存器 0	TAB1OPT0		√	√		00H
FFFFF566H	TAB1 捕获/比较寄存器 0	TAB1CCR0				√	0000H
FFFFF568H	TAB1 捕获/比较寄存器 1	TAB1CCR1				√	0000H
FFFFF56AH	TAB1 捕获/比较寄存器 2	TAB1CCR2				√	0000H
FFFFF56CH	TAB1 捕获/比较寄存器 3	TAB1CCR3				√	0000H
FFFFF56EH	TAB1 计数器读缓冲寄存器	TAB1CNT	R			√	0000H
FFFFF570H	TAB1 I/O 控制寄存器 4	TAB1IOC4	R/W	√	√		00H
FFFFF580H	TAB1 选项寄存器 1	TAB1OPT1		√	√		00H
FFFFF581H	TAB1 选项寄存器 2	TAB1OPT2		√	√		00H
FFFFF582H	TAB1 I/O 控制寄存器 3	TAB1IOC3		√	√		A8H
FFFFF584H	TAB1 死区时间比较寄存器 1	TAB1DTC				√	0000H
FFFFF590H	高阻抗输出控制寄存器 0	HZACTL0		√	√		00H
FFFFF591H	高阻抗输出控制寄存器 1	HZACTL1		√	√		00H
FFFFF600H	TMT0 控制寄存器 0	TT0CTL0		√	√		00H
FFFFF601H	TMT0 控制寄存器 1	TT0CTL1		√	√		00H
FFFFF602H	TMT0 控制寄存器 2	TT0CTL2		√	√		00H
FFFFF603H	TMT0 I/O 控制寄存器 0	TT0IOC0		√	√		00H
FFFFF604H	TMT0 I/O 控制寄存器 1	TT0IOC1		√	√		00H
FFFFF605H	TMT0 I/O 控制寄存器 2	TT0IOC2		√	√		00H
FFFFF606H	TMT0 I/O 控制寄存器 3	TT0IOC3		√	√		00H

(8/14)

地址	功能寄存器名称	符号	R/W	可操作位			默认值
				1	8	16	
FFFFF607H	TMT0 选项寄存器 0	TT0OPT0		√	√		00H
FFFFF608H	TMT0 选项寄存器 1	TT0OPT1		√	√		00H
FFFFF609H	TMT0 选项寄存器 2	TT0OPT2		√	√		00H
FFFFF60AH	TMT0 捕获/比较寄存器 0	TT0CCR0				√	0000H
FFFFF60CH	TMT0 捕获/比较寄存器 1	TT0CCR1	R/W			√	0000H
FFFFF60EH	TMT0 计数器读缓冲寄存器	TT0CNT	R			√	0000H
FFFFF610H	TMT0 计数器写寄存器	TT0TCW	R/W			√	0000H
FFFFF630H	TAA0 控制寄存器 0	TAA0CTL0		√	√		00H
FFFFF631H	TAA0 控制寄存器 1	TAA0CTL1		√	√		00H
FFFFF632H	TAA0 I/O 控制寄存器 0	TAA0IOC0		√	√		00H
FFFFF633H	TAA0 I/O 控制寄存器 1	TAA0IOC1		√	√		00H
FFFFF634H	TAA0 I/O 控制寄存器 2	TAA0IOC2		√	√		00H
FFFFF635H	TAA0 选项寄存器 0	TAA0OPT0		√	√		00H
FFFFF636H	TAA0 捕获/比较寄存器 0	TAA0CCR0				√	0000H
FFFFF638H	TAA0 捕获/比较寄存器 1	TAA0CCR1				√	0000H
FFFFF63AH	TAA0 计数器读缓冲寄存器	TAA0CNT	R			√	0000H
FFFFF63CH	TAA0 I/O 控制寄存器 4	TAA0IOC4	R/W	√	√		00H
FFFFF63DH	TAA0 选项寄存器 1	TAA0OPT1		√	√		00H
FFFFF640H	TAA1 控制寄存器 0	TAA1CTL0		√	√		00H
FFFFF641H	TAA1 控制寄存器 1	TAA1CTL1		√	√		00H
FFFFF642H	TAA1 I/O 控制寄存器 0	TAA1IOC0		√	√		00H
FFFFF643H	TAA1 I/O 控制寄存器 1	TAA1IOC1		√	√		00H
FFFFF644H	TAA1 I/O 控制寄存器 2	TAA1IOC2		√	√		00H
FFFFF645H	TAA1 选项寄存器 0	TAA1OPT0		√	√		00H
FFFFF646H	TAA1 捕获/比较寄存器 0	TAA1CCR0				√	0000H
FFFFF648H	TAA1 捕获/比较寄存器 1	TAA1CCR1				√	0000H
FFFFF64AH	TAA1 计数器读缓冲寄存器	TAA1CNT	R			√	0000H
FFFFF64CH	TAA1 I/O 控制寄存器 4	TAA1IOC4	R/W	√	√		00H
FFFFF650H	TAA2 控制寄存器 0	TAA2CTL0		√	√		00H
FFFFF651H	TAA2 控制寄存器 1	TAA2CTL1		√	√		00H
FFFFF652H	TAA2 I/O 控制寄存器 0	TAA2IOC0		√	√		00H
FFFFF653H	TAA2 I/O 控制寄存器 1	TAA2IOC1		√	√		00H
FFFFF654H	TAA2 I/O 控制寄存器 2	TAA2IOC2		√	√		00H
FFFFF655H	TAA2 选项寄存器 0	TAA2OPT0		√	√		00H
FFFFF656H	TAA2 捕获/比较寄存器 0	TAA2CCR0				√	0000H
FFFFF658H	TAA2 捕获/比较寄存器 1	TAA2CCR1				√	0000H
FFFFF65AH	TAA2 计数器读缓冲寄存器	TAA2CNT	R			√	0000H
FFFFF65CH	TAA2 I/O 控制寄存器 4	TAA2IOC4	R/W	√	√		00H
FFFFF65DH	TAA2 选项寄存器 1	TAA2OPT1		√	√		00H
FFFFF660H	TAA3 控制寄存器 0	TAA3CTL0		√	√		00H
FFFFF661H	TAA3 控制寄存器 1	TAA3CTL1		√	√		00H
FFFFF662H	TAA3 I/O 控制寄存器 0	TAA3IOC0		√	√		00H
FFFFF663H	TAA3 I/O 控制寄存器 1	TAA3IOC1		√	√		00H

(9/14)

地址	功能寄存器名称	符号	R/W	可操作位			默认值
				1	8	16	
FFFFF664H	TAA3 I/O 控制寄存器 2	TAA3IOC2	R/W	√	√		00H
FFFFF665H	TAA3 选项寄存器 0	TAA3OPT0		√	√		00H
FFFFF666H	TAA3 捕获/比较寄存器 0	TAA3CCR0				√	0000H
FFFFF668H	TAA3 捕获/比较寄存器 1	TAA3CCR1				√	0000H
FFFFF66AH	TAA3 计数器读缓冲寄存器	TAA3CNT	R			√	0000H
FFFFF66CH	TAA3 I/O 控制寄存器 4	TAA3IOC4	R/W	√	√		00H
FFFFF670H	TAA4 控制寄存器 0	TAA4CTL0		√	√		00H
FFFFF671H	TAA4 控制寄存器 1	TAA4CTL1		√	√		00H
FFFFF676H	TAA4 捕获/比较寄存器 0	TAA4CCR0				√	0000H
FFFFF678H	TAA4 捕获/比较寄存器 1	TAA4CCR1				√	0000H
FFFFF67AH	TAA4 计数器读缓冲寄存器	TAA4CNT	R			√	0000H
FFFFF680H	TAA5 控制寄存器 0	TAA5CTL0	R/W	√	√		00H
FFFFF681H	TAA5 控制寄存器 1	TAA5CTL1		√	√		00H
FFFFF682H	TAA5 I/O 控制寄存器 0	TAA5IOC0		√	√		00H
FFFFF683H	TAA5 I/O 控制寄存器 1	TAA5IOC1		√	√		00H
FFFFF684H	TAA5 I/O 控制寄存器 2	TAA5IOC2		√	√		00H
FFFFF685H	TAA5 选项寄存器 0	TAA5OPT0		√	√		00H
FFFFF686H	TAA5 捕获/比较寄存器 0	TAA5CCR0				√	0000H
FFFFF688H	TAA5 捕获/比较寄存器 1	TAA5CCR1				√	0000H
FFFFF68AH	TAA5 计数器读缓冲寄存器	TAA5CNT	R			√	0000H
FFFFF68CH	TAA5 I/O 控制寄存器 4	TAA5IOC4	R/W	√	√		00H
FFFFF6C0H	振荡稳定时间选择寄存器	OSTS			√		06H
FFFFF6C1H	PLL 锁定时间指定寄存器	PLLS			√		03H
FFFFF6D0H	看门狗定时器模式寄存器 2	WDTM2			√		67H
FFFFF6D1H	看门狗定时器使能寄存器	WDTE			√		9AH
FFFFF6E0H	实时输出缓冲寄存器 0L	RTBL0		√	√		00H
FFFFF6E2H	实时输出缓冲寄存器 0H	RTBH0		√	√		00H
FFFFF6E4H	实时输出端口模式寄存器 0	RTPM0		√	√		00H
FFFFF6E5H	实时输出端口控制寄存器 0	RTPC0		√	√		00H
FFFFF700H	0 端口功能控制扩展寄存器	PFCE0		√	√		00H
FFFFF704H	2 端口功能控制扩展寄存器 ^注	PFCE2 ^注		√	√		00H
FFFFF706H	3 端口功能控制扩展寄存器	PFCE3		√	√		00H
FFFFF708H	4 端口功能控制扩展寄存器	PFCE4		√	√		00H
FFFFF70AH	5 端口功能控制扩展寄存器	PFCE5		√	√		00H
FFFFF70CH	6 端口功能控制扩展寄存器	PFCE6		√	√		00H
FFFFF712H	9 端口功能控制扩展寄存器	PFCE9				√	00H
FFFFF712H	9 端口功能控制扩展寄存器 L	PFCE9L		√	√		0000H
FFFFF713H	2 端口功能控制扩展寄存器 H	PFCE9H		√	√		00H
FFFFF724H	TAA 噪声抑制控制寄存器	TANFC		√	√		00H
FFFFF726H	TMT 噪声抑制控制寄存器	TTNFC		√	√		00H
FFFFF728H	噪声抑制控制寄存器	INTNFC		√	√		00H
FFFFF802H	系统状态寄存器	SYS		√	√		00H

<R>

注 只适用于 V850ES/JH3-H

(10/14)

	地址	功能寄存器名称	符号	R/W	可操作位			默认值
					1	8	16	
	FFFFF80CH	内部振荡模式寄存器	RCM	R/W	√	√		00H
	FFFFF810H	DMA 触发因子寄存器 0	DTFR0		√	√		00H
	FFFFF812H	DMA 触发因子寄存器 1	DTFR1		√	√		00H
	FFFFF814H	DMA 触发因子寄存器 2	DTFR2		√	√		00H
	FFFFF820H	节电模式寄存器	PSMR		√	√		00H
	FFFFF822H	时钟控制寄存器	CKC		√	√		0AH
	FFFFF824H	加锁寄存器	LOCKR	R	√	√		00H
	FFFFF828H	处理器时钟控制寄存器	PCC	R/W	√	√		03H
	FFFFF82CH	PLL 控制寄存器	PLLCTL		√	√		01H
	FFFFF82EH	CPU 工作时钟状态寄存器	CCLS	R	√	√		00H
	FFFFF870H	时钟监视器模式寄存器	CLM	R/W	√	√		00H
	FFFFF888H	复位源标志寄存器	RESF		√	√		00H
	FFFFF890H	低电压检测寄存器	LVIM		√	√		00H
	FFFFF892H	内部 RAM 数据状态寄存器	RAMS		√	√		01H
<R>	FFFFF8B0H	预分频器模式寄存器 0	PRSM0		√	√		00H
<R>	FFFFF8B1H	预分频器比较寄存器 0	PRSCM0			√		00H
	FFFFF9FCH	片上调试模式寄存器 ^注	OCDM ^注		√	√		01H
	FFFFFA00H	UARTC0 控制寄存器 0	UC0CTL0		√	√		10H
	FFFFFA01H	UARTC0 控制寄存器 1	UC0CTL1			√		00H
	FFFFFA02H	UARTC0 控制寄存器 2	UC0CTL2			√		FFH
	FFFFFA03H	UARTC0 选项控制寄存器 0	UC0OPT0		√	√		14H
	FFFFFA04H	UARTC0 状态寄存器	UC0STR		√	√		00H
	FFFFFA06H	UARTC0 接收数据寄存器	UC0RX	R		√	√	01FFH
<R>	FFFFFA06H	UARTC0 接收数据寄存器 L	UC0RXL			√		FFH
	FFFFFA08H	UARTC0 发送数据寄存器	UC0TX	R/W			√	01FFH
<R>	FFFFFA08H	UARTC0 发送数据寄存器 L	UC0TXL			√		FFH
	FFFFFA0AH	UARTC0 选项控制寄存器 1	UC0OPT1		√	√		00H
	FFFFFA10H	UARTC1 控制寄存器 0	UC1CTL0		√	√		10H
	FFFFFA11H	UARTC1 控制寄存器 1	UC1CTL1			√		00H
	FFFFFA12H	UARTC1 控制寄存器 2	UC1CTL2			√		FFH
	FFFFFA13H	UARTC1 选项控制寄存器 0	UC1OPT0		√	√		14H
	FFFFFA14H	UARTC1 状态寄存器	UC1STR		√	√		00H
	FFFFFA16H	UARTC1 接收数据寄存器	UC1RX	R			√	01FFH
<R>	FFFFFA16H	UARTC1 接收数据寄存器 L	UC1RXL			√		FFH
	FFFFFA18H	UARTC1 发送数据寄存器	UC1TX	R/W			√	01FFH
<R>	FFFFFA18H	UARTC1 发送数据寄存器 L	UC1TXL			√		FFH
	FFFFFA1AH	UARTC1 选项控制寄存器 1	UC1OPT1		√	√		00H
	FFFFFA20H	UARTC2 控制寄存器 0	UC2CTL0		√	√		10H
	FFFFFA21H	UARTC2 控制寄存器 1	UC2CTL1			√		00H
	FFFFFA22H	UARTC2 控制寄存器 2	UC2CTL2			√		FFH
	FFFFFA23H	UARTC2 选项控制寄存器 0	UC2OPT0		√	√		14H

注 只适用于 V850ES/JG3-H

(11/14)

	地址	功能寄存器名称	符号	R/W	可操作位			默认值
					1	8	16	
<R>	FFFFFA24H	UARTC2 状态寄存器	UC2STR	R/W	√	√		00H
	FFFFFA26H	UARTC2 接收数据寄存器	UC2RX	R			√	01FFH
	FFFFFA26H	UARTC2 接收数据寄存器 L	UC2RXL			√		FFH
<R>	FFFFFA28H	UARTC2 发送数据寄存器	UC2TX	R/W			√	01FFH
	FFFFFA28H	UARTC2 发送数据寄存器 L	UC2TXL			√		FFH
	FFFFFA2AH	UARTC2 选项控制寄存器 1	UC2OPT1		√	√		00H
<R>	FFFFFA30H	UARTC3 控制寄存器 0	UC3CTL0		√	√		10H
	FFFFFA31H	UARTC3 控制寄存器 1	UC3CTL1			√		00H
	FFFFFA32H	UARTC3 控制寄存器 2	UC3CTL2			√		FFH
<R>	FFFFFA33H	UARTC3 选项控制寄存器 0	UC3OPT0		√	√		14H
	FFFFFA34H	UARTC3 状态寄存器	UC3STR		√	√		00H
	FFFFFA36H	UARTC3 接收数据寄存器	UC3RX	R			√	01FFH
<R>	FFFFFA36H	UARTC3 接收数据寄存器 L	UC3RXL			√		FFH
	FFFFFA38H	UARTC3 发送数据寄存器	UC3TX	R/W			√	01FFH
	FFFFFA38H	UARTC3 发送数据寄存器 L	UC3TXL			√		FFH
	FFFFFA3AH	UARTC3 选项控制寄存器 1	UC3OPT1		√	√		00H
<R>	FFFFFA40H	UARTC4 控制寄存器 0	UC4CTL0		√	√		10H
	FFFFFA41H	UARTC4 控制寄存器 1	UC4CTL1			√		00H
	FFFFFA42H	UARTC4 控制寄存器 2	UC4CTL2			√		FFH
<R>	FFFFFA43H	UARTC4 选项控制寄存器 0	UC4OPT0		√	√		14H
	FFFFFA44H	UARTC4 状态寄存器	UC4STR		√	√		00H
	FFFFFA46H	UARTC4 接收数据寄存器	UC4RX	R			√	01FFH
<R>	FFFFFA46H	UARTC4 接收数据寄存器 L	UC4RXL			√		FFH
	FFFFFA48H	UARTC4 发送数据寄存器	UC4TX	R/W			√	01FFH
	FFFFFA48H	UARTC4 发送数据寄存器 L	UC4TXL			√		FFH
	FFFFFA4AH	UARTC4 选项控制寄存器 1	UC4OPT1		√	√		00H
<R>	FFFFFA80H	TMM0 控制寄存器 0	TM0CTL0		√	√		00H
	FFFFFA84H	TMM0 比较寄存器 0	TM0CMP0				√	0000H
	FFFFFA90H	TMM1 控制寄存器 0	TM1CTL0		√	√		00H
<R>	FFFFFA94H	TMM1 比较寄存器 0	TM1CMP0				√	0000H
	FFFFFAA0H	TMM2 控制寄存器 0	TM2CTL0		√	√		00H
	FFFFFAA4H	TMM2 比较寄存器 0	TM2CMP0				√	0000H
<R>	FFFFFAB0H	TMM3 控制寄存器 0	TM3CTL0		√	√		00H
	FFFFFAB4H	TMM3 比较寄存器 0	TM3CMP0				√	0000H
	FFFFFAD0H	副-计数寄存器	RC1SUBC	R			√	0000H
<R>	FFFFFAD2H	秒计数寄存器	RC1SEC	R/W		√		00H
	FFFFFAD3H	分计数寄存器	RC1MIN			√		00H
	FFFFFAD4H	时计数寄存器	RC1HOUR			√		12H
<R>	FFFFFAD5H	星期计数寄存器	RC1WEEK			√		00H
	FFFFFAD6H	日计数寄存器	RC1DAY			√		01H
	FFFFFAD7H	月计数寄存器	RC1MONTH			√		01H
<R>	FFFFFAD8H	年计数寄存器	RC1YEAR			√		00H
	FFFFFAD9H	时间错误校正寄存器	RC1SUBU		√	√		00H

(12/14)

	地址	功能寄存器名称	符号	R/W	可操作位			默认值
					1	8	16	
<R>	FFFFFADAH	报警分钟设置寄存器	RC1ALM	R/W		√		00H
	FFFFFADBH	报警时间设置寄存器	RC1ALH			√		12H
	FFFFFADCH	报警星期设置寄存器	RC1ALW		√	√		00H
	FFFFFADDH	RTC 控制寄存器 0	RC1CC0		√	√		00H
	FFFFFADEH	RTC 控制寄存器 1	RC1CC1		√	√		00H
	FFFFFADFH	RTC 控制寄存器 2	RC1CC2		√	√		00H
	FFFFFAE0H	RTC 控制寄存器 3	RC1CC3		√	√		00H
	FFFFFC00H	外部中断下降沿指定寄存器 0	INTF0		√	√		00H
	FFFFFC04H	外部中断下降沿指定寄存器 2 ^{注1}	INTF2 ^{注1}		√	√		00H
	FFFFFC06H	外部中断下降沿指定寄存器 3	INTF3		√	√		00H
	FFFFFC08H	外部中断下降沿指定寄存器 4	INTF4		√	√		00H
	FFFFFC0AH	外部中断下降沿指定寄存器 5 ^{注2}	INTF5 ^{注2}		√	√		00H
	FFFFFC12H	外部中断下降沿指定寄存器 9	INTF9				√	0000H
	FFFFFC12H	外部中断下降沿指定寄存器 9H	INTF9H		√	√		00H
	FFFFFC13H	外部中断下降沿指定寄存器 9L	INTF9L		√	√		00H
<R>	FFFFFC20H	外部中断上升沿指定寄存器 0	INTR0	R/W	√	√		00H
	FFFFFC24H	外部中断上升沿指定寄存器 2 ^{注1}	INTR2 ^{注1}		√	√		00H
	FFFFFC26H	外部中断上升沿指定寄存器 3	INTR3		√	√		00H
	FFFFFC28H	外部中断上升沿指定寄存器 4	INTR4		√	√		00H
	FFFFFC2AH	外部中断上升沿指定寄存器 5 ^{注2}	INTR5 ^{注2}		√	√		00H
	FFFFFC32H	外部中断上升沿指定寄存器 9	INTR9				√	0000H
	FFFFFC32H	外部中断上升沿指定寄存器 9H	INTR9H		√	√		00H
	FFFFFC33H	外部中断上升沿指定寄存器 9L	INTR9L		√	√		00H
	FFFFFC60H	0 端口功能寄存器	PF0		√	√		00H
	FFFFFC64H	2 端口功能寄存器 ^{注1}	PF2 ^{注1}		√	√		00H
	FFFFFC66H	3 端口功能寄存器	PF3		√	√		00H
	FFFFFC68H	4 端口功能寄存器	PF4		√	√		00H
	FFFFFC6AH	5 端口功能寄存器	PF5		√	√		00H
	FFFFFC72H	9 端口功能寄存器 L	PF9L		√	√		00H
	FFFFFD00H	CSIF0 控制寄存器 0	CF0CTL0		√	√		01H
	FFFFFD01H	CSIF0 控制寄存器 1	CF0CTL1		√	√		00H
	FFFFFD02H	CSIF0 控制寄存器 2	CF0CTL2			√		00H
	FFFFFD03H	CSIF0 状态寄存器	CF0STR		√	√		00H
	FFFFFD04H	CSIF0 接收数据寄存器	CF0RX	R			√	0000H
	FFFFFD04H	CSIF0 接收数据寄存器 L	CF0RXL			√		00H
	FFFFFD06H	CSIF0 发送数据寄存器	CF0TX	R/W			√	0000H
	FFFFFD06H	CSIF0 发送数据寄存器 L	CF0TXL			√		00H
	FFFFFD10H	CSIF1 控制寄存器 0	CF1CTL0		√	√		01H
	FFFFFD11H	CSIF1 控制寄存器 1	CF1CTL1		√	√		00H
	FFFFFD12H	CSIF1 控制寄存器 2	CF1CTL2			√		00H
	FFFFFD13H	CSIF1 状态寄存器	CF1STR		√	√		00H

注 1. 只适用于 V850ES/JH3-H

2. 只适用于 V850ES/JG3-H

(13/14)

地址	功能寄存器名称	符号	R/W	可操作位			默认值
				1	8	16	
FFFFFD14H	CSIF1 接收数据寄存器	CF1RX	R			√	0000H
FFFFFD14H	CSIF1 接收数据寄存器 L	CF1RXL			√		00H
FFFFFD16H	CSIF1 发送数据寄存器	CF1TX	R/W			√	0000H
FFFFFD16H	CSIF1 发送数据寄存器 L	CF1TXL			√		00H
FFFFFD20H	CSIF2 控制寄存器 0	CF2CTL0		√	√		01H
FFFFFD21H	CSIF2 控制寄存器 1	CF2CTL1		√	√		00H
FFFFFD22H	CSIF2 控制寄存器 2	CF2CTL2	R		√		00H
FFFFFD23H	CSIF2 状态寄存器	CF2STR		√	√		00H
FFFFFD24H	CSIF2 接收数据寄存器	CF2RX				√	0000H
FFFFFD24H	CSIF2 接收数据寄存器 L	CF2RXL			√		00H
FFFFFD26H	CSIF2 发送数据寄存器	CF2TX	R/W			√	0000H
FFFFFD26H	CSIF2 发送数据寄存器 L	CF2TXL			√		00H
FFFFFD30H	CSIF3 控制寄存器 0	CF3CTL0		√	√		01H
FFFFFD31H	CSIF3 控制寄存器 1	CF3CTL1		√	√		00H
FFFFFD32H	CSIF3 控制寄存器 2	CF3CTL2	R		√		00H
FFFFFD33H	CSIF3 状态寄存器	CF3STR		√	√		00H
FFFFFD34H	CSIF3 接收数据寄存器	CF3RX				√	0000H
FFFFFD34H	CSIF3 接收数据寄存器 L	CF3RXL			√		00H
FFFFFD36H	CSIF3 发送数据寄存器	CF3TX	R/W			√	0000H
FFFFFD36H	CSIF3 发送数据寄存器 L	CF3TXL			√		00H
FFFFFD40H	CSIF4 控制寄存器 0	CF4CTL0		√	√		01H
FFFFFD41H	CSIF4 控制寄存器 1	CF4CTL1		√	√		00H
FFFFFD42H	CSIF4 控制寄存器 2	CF4CTL2	R		√		00H
FFFFFD43H	CSIF4 状态寄存器	CF4STR		√	√		00H
FFFFFD44H	CSIF4 接收数据寄存器	CF4RX				√	0000H
FFFFFD44H	CSIF4 接收数据寄存器 L	CF4RXL			√		00H
FFFFFD46H	CSIF4 发送数据寄存器	CF4TX	R/W			√	0000H
FFFFFD46H	CSIF4 发送数据寄存器 L	CF4TXL			√		00H
FFFFFD80H	IIC 移位寄存器 0	IIC0			√		00H
FFFFFD82H	IIC 控制寄存器 0	IICC0		√	√		00H
FFFFFD83H	从地址寄存器 0	SVA0	R		√		00H
FFFFFD84H	IIC 时钟选择寄存器 0	IICCL0		√	√		00H
FFFFFD85H	IIC 功能扩展寄存器 0	IICX0		√	√		00H
FFFFFD86H	IIC 状态寄存器 0	IICS0		√	√		00H
FFFFFD8AH	IIC 标志寄存器 0	IICF0	R/W	√	√		00H
FFFFFD90H	IIC 移位寄存器 1	IIC1			√		00H
FFFFFD92H	IIC 控制寄存器 1	IICC1		√	√		00H
FFFFFD93H	从地址寄存器 1	SVA1			√		00H
FFFFFD94H	IIC 时钟选择寄存器 1	IICCL1	R	√	√		00H
FFFFFD95H	IIC 功能扩展寄存器 1	IICX1		√	√		00H
FFFFFD96H	IIC 状态寄存器 1	IICS1		√	√		00H
FFFFFD9AH	IIC 标志寄存器 1	IICF1		√	√		00H
FFFFFDA0H	IIC 移位寄存器 2	IIC2	R/W		√		00H

(14/14)

地址	功能寄存器名称	符号	R/W	可操作位			默认值
				1	8	16	
FFFFFDA2H	IIC 控制寄存器 2	IICC2	R/W	√	√		00H
FFFFFDA3H	从地址寄存器 2	SVA2			√		00H
FFFFFDA4H	IIC 时钟选择寄存器 2	IICCL2		√	√		00H
FFFFFDA5H	IIC 功能扩展寄存器 2	IICX2		√	√		00H
FFFFFDA6H	IIC 状态寄存器 2	IICS2	R	√	√		00H
FFFFFDAAH	IIC 标志寄存器 2	IICF2	R/W	√	√		00H
FFFFFF40H	USB 时钟选择寄存器	UCKSEL		√	√		00H
FFFFFF41H	USB 功能控制寄存器	UFCKMSK		√	√		03H
FFFFFF42H	USB 功能选择寄存器	UHCKMSK		√	√		03H
FFFFFF60H	外部 DMA 请求允许寄存器	EXDRQEN			√		00H

<R>

3.4.7 可编程的外设I/O 寄存器

BPC 寄存器用于选择可编程的外设 I/O 寄存器区域。
BPC 寄存器只在μPD70F3770、70F3771 中有效。

(1) 外设 I/O 区域选择控制寄存器（BPC）

该寄存器可按 16 位为单位进行读写。
复位后本寄存器设置为 0000H。

复位后： 0000H R/W 地址： FFFFF064H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BPC	PA15	0	PA13	PA12	PA11	PA10	PA09	PA08	PA07	PA06	PA05	PA04	PA03	PA02	PA01	PA00
	PA15	允许/不允许使用可编程的外设I/O区域。														
	0	不允许使用可编程的外设I/O区域。														
	1	允许使用可编程的外设I/O区域。														
	PA13~PA00	设置可编程的外设I/O区域的地址。（对应A27~A14）														

注意事项 如果将 PA15 位设置为 1，则务必将 BPC 寄存器设置为 8FFBH。
如果将 PA15 位设置为 0，则务必将 BPC 寄存器设置为 0000H。

有关可编程外设 I/O 寄存器的列表，参见表 20-16 寄存器访问类型。

3.4.8 特殊寄存器

特殊寄存器是免受因程序环而引起的非法数据写入的寄存器。V850ES/JG3-H 和 V850ES/JH3-H 有以下八种特殊寄存器。

- 节电控制寄存器 (PSC)
- 时钟控制寄存器 (CKC)
- 处理器时钟控制寄存器 (PCC)
- 时钟监视器模式寄存器 (CLM)
- 复位源标志寄存器 (RESF)
- 低电压检测寄存器 (LVIM)
- 内部 RAM 数据状态寄存器 (RAMS)
- 片上调试模式寄存器 (OCDM) (只适用于 V850ES/JG3-H)

此外，器件还提供了 PRCDM 寄存器来防止对特殊寄存器的非法写操作，这样应用系统就不会因为程序环而无益停止工作。对特殊寄存器的写操作按特定次序进行，并把非法保存操作报告给 SYS 寄存器。

(1) 设置特殊寄存器数据

按照以下流程设置特殊寄存器数据。

- <1> 禁止 DMA 操作
- <2> 在通用寄存器中，准备将设置给特殊寄存器的数据。
- <3> 将步骤<2>准备好的数据写入 PRCMD 寄存器中。
- <4> 将设置的数据写入特殊寄存器（使用如下指令）。
 - 存储指令（ST/SST 指令）
 - 位操作指令（SET1/CLR1/NOT1 指令）。
- （<5>至<9>插入 NOP 指令（5 条指令）。）^注
- <10> 如果需要，恢复 DMA 操作。

[实例] 使用 PSC 寄存器（设置待机模式）

```

ST.B r11, PSMDR[r0]      ; 设置 PSMDR 寄存器（设置 IDLE1、IDLE2 和 STOP 模式）。
<1>CLR1 0, DCHCn[r0]      ; 禁止 DMA 操作，n = 0 至 3

<2>MOV0x02, r10
<3>ST.B r10, PRCMD[r0]    ; 写 PRCMD 寄存器。
<4>ST.B r10, PSC[r0]      ; 设置 PSC 寄存器。
<5>NOP注                  ; 空指令
<6>NOP注                  ; 空指令
<7>NOP注                  ; 空指令
<8>NOP注                  ; 空指令
<9>NOP注                  ; 空指令
<10>SET1 0, DCHCn[r0]    ; 恢复 DMA 操作，n = 0 至 3
（下一条指令）

```

不需要按特定次序读取特殊寄存器。

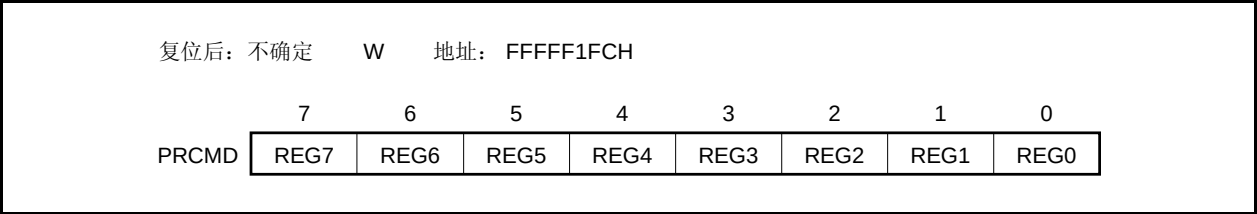
注 设置 IDLE1 模式、IDLE2 模式或 STOP 模式（通过将 PSC.STP 位置为 1）后，必须立即插入 5 条以上的 NOP 指令。

- 注意事项**
1. 在命令寄存器中执行存储指令存储数据时，不响应中断。这是因为我们认为上述的步骤<3>和<4>是由相继的存储指令执行的。若在步骤<3>和<4>之间安置了其他指令，且此指令响应中断，则上面的次序不成立，从而引发故障。
 2. 尽管已有空数据写入 PRCMD 寄存器，仍能利用与设置特殊寄存器（例中的步骤<4>）时所用的一样的通用寄存器将数据写入 PRCMD 寄存器（例中的步骤<3>）。当通用寄存器用于寻址时同样适用。

(2) 命令寄存器（PRCMD）

PRCMD 寄存器是一款 8 位寄存器，用于保护那些会严重影响应用系统的寄存器不被轻易写入，从而使应用系统不会因为程序挂起而无意停止工作。数据预先写入 PRCMD 寄存器后，对特殊寄存器的第一次写访问有效。这样，特殊寄存器的值仅能以特定的流程进行写入，这样就能使特殊寄存器免受非法写访问。

PRCMD 是以 8 位为单元的只写寄存器（当对其进行读取时，所读数据不确定）。



(3) 系统状态寄存器 (SYS)

该寄存器存有状态标志，这些状态标志指示整体系统的工作状态。

该寄存器可按 8 位或 1 位为单元进行读写。

复位后，该寄存器置为 00H。

复位后: 00H R/W 地址: FFFFF802H

	7	6	5	4	3	2	1	<0>
SYS	0	0	0	0	0	0	0	PRERR

PRERR	检测保护错误
0	未发生保护错误
1	发生保护错误

PRERR 标志按下列条件处理。

(a) 置位条件 (PRERR 标志=1)

- (i) 当没有写数据至 PRCMD 寄存器就在特殊寄存器写入数据时（当没有执行 **3.4.8 (1) 设置特殊寄存器数据** 的步骤 <3> 就执行步骤 <4> 时）
- (ii) 在数据写入 PRCMD 寄存器后，数据写入片内外设 I/O 寄存器而不是特殊寄存器（包括位处理指令的运行）时（如果 **3.4.8 (1) 设置特殊寄存器数据** 中的步骤<4>不是对特殊寄存器的设置时）

备注 即使在写入 PRCMD 寄存器与写入特殊寄存器的操作（位操作指令除外）之间读取片上外围 I/O 寄存器，也不会设置 PRERR 标志，且设置数据可能会写入特殊寄存器。

(b) 清零条件 (PRERR 标志=0)

- (i) 当 0 写入 PRERR 标志位时
- (ii) 当系统复位时

- 注意事项**
- 1. 如果在对 PRCMD 寄存器进行写访问之后，立即将 0 写入 SYS 寄存器（并非特殊寄存器）的 PRERR 位，那么 PRERR 位将清为零（写访问优先）。
 - 2. 如果在对 PRCMD 寄存器进行写访问之后立即将数据写入 PRCMD 寄存器（该寄存器并不是特殊寄存器），那么，PRERR 位被设为 1。

3.4.9 注意事项

(1) 需要先行设置的寄存器

当使用 V850ES/JG3-H 和 V850ES/JH3-H 单片机时，务必首先对下列寄存器进行设置。

- 系统等待控制寄存器（VSWC）
- 片上调试模式寄存器（OCDM）（只适用于 V850ES/JG3-H）
- 看门狗定时器模式寄存器 2（WDTM2）

设置 VSWC、OCDM 和 WDTM2 寄存器后，如果需要的话，再对其它寄存器进行设置。

当使用外部总线时，在上述寄存器设置完成之后，通过使用端口相关寄存器，将各个引脚设置为复用功能总线控制引脚模式。

(a) 系统等待控制寄存器（VSWC）

VSWC 寄存器控制总线访问片上外围 I/O 寄存器的等待时间。

访问片上外围 I/O 寄存器（无等待周期）需要三个时钟。工作频率变化，V850ES/JG3-H 和 V850ES/JH3-H 单片机的等待周期也随之变化。按照所用的工作频率，按以下各值设置 VSWC 寄存器。

BSC 寄存器可以按 8 位为单元进行读写。

复位后，将该寄存器置为 77H。

复位后：77H R/W 地址：FFFFF06EH

	7	6	5	4	3	2	1	0
VSWC								

工作频率 (f_{CPU})	VSWC 的设置值	等待数
$f_{CPU} < 16.6 \text{ MHz}$	00H	0（无等待周期）
$16.6 \text{ MHz} \leq f_{CPU} < 25 \text{ MHz}$	01H	1
$25 \text{ MHz} \leq f_{CPU} < 33.3 \text{ MHz}$	11H	2
$33.3 \text{ MHz} \leq f_{CPU} < 48 \text{ MHz}$	12H	3

(b) 片上调试模式寄存器（OCDM）（只适用于 V850ES/JG3-H）

详情参见第 32 章 片上调试功能。

(c) 看门狗定时器模式寄存器 2（WDTM2）

WDTM2 寄存器用以设置看门狗定时器 2 的溢出时间和工作时钟。

在复位模式下，当复位解除后看门狗定时器 2 会自动启动。通过写入 WDTM2 寄存器来激活该操作。

详情参见第 13 章 看门狗定时器 2 的功能。

(2) 访问特殊片上外设 I/O 寄存器

本产品有两种类型的内部系统总线。

一种是 CPU 总线，另外一种是用作为低速外设硬件接口的外设总线，。

CPU 总线时钟与外设总线时钟是异步的。因此，当访问 CPU 与访问外设硬件发生冲突的时候，就有可能传输无法预料的非法数据。若可能发生冲突，当访问外设硬件时，需改变访问 CPU 的时钟周期数目，从而传输正确的数据。这样，CPU 就不会启动下一条指令的处理而是进入等待状态。当这种等待状态发生，执行一条指令所需的时钟数就会根据下面显示的等待时钟的数目有所增加。

当执行实时处理任务时需要特别注意这种情况。

当访问指定的片上外设 I/O 寄存器时，除了由 VSWC 寄存器设置的等待状态之外，可能还需要更多的等待状态。

访问条件以及如何计算所要插入的等待状态数目（CPU 时钟数）如下表所示：

(1/2)

外设功能	寄存器名称	访问类型	k
16-位定时器/事件计数器 AA (TAA) (n = 0 至 5, m = 0 至 3、5)	TAA nCNT	读	1 或 2
	TAA nCCR0、TAA nCCR1	写	- 第一次访问：无等待 - 连续写入：0 至 3
		读	1 或 2
	TAA mIOC4	写	- 第一次访问：无等待 - 连续写入：0 至 3
		读	1 或 2
16-位定时器/事件计数器 AB (TAB) (n = 0、1)	TAB nCNT	读	1 或 2
	TAB nCCR0 至 TAB nCCR3	写	- 第一次访问：无等待 - 连续写入：0 至 3
		读	1 或 2
	TAB nIOC4	写	- 第一次访问：无等待 - 连续写入：0 至 3
		读	1 或 2
马达控制	TAB0OPT1	写	- 第一次访问：无等待 - 连续写入：0 至 3
	TAB0DTC	写	- 第一次访问：无等待 - 连续写入：0 至 3
TMT	TT0CNT	读	1 或 2
	TT0TCR0、TT0TCR1	写	- 第一次访问：无等待 - 连续写入：0 至 3
		读	1 或 2
看门狗定时器 2 (WDT2)	WDTM2	写 (当 WDT2 工作时)	3
实时输出功能 (RTO)	RTBL0、RTBH0	写 (RTPC0.RTPOE0 位=0)	1
A/D 转换器	ADA0M0	读	1 或 2
	ADA0CR0 至 ADA0CR11	读	1 或 2
	ADA0CR0H 至 ADA0CR11H	读	1 或 2

(2/2)

外设功能	寄存器名称	访问类型	k
I ² C00 至 I ² C02	IICS0 至 IICS2	读	1
CRC	CRCD	写	1
CAN 控制器 (m = 0 至 31, a = 1 至 4)	C0GMABT, C0GMABTD, C0MASKaL、C0MASKaH, C0LEC, C0INFO, C0ERC, C0IE, C0INTS, C0BRP, C0BTR, C0TS	读/写	$f_{xx}/f_{CANMOD} + 1 / (2 + j)$ (MIN.) [‡] $(2 \times f_{xx}/f_{CANMOD} + 1) / (2 + j)$ (MAX.) [‡]
	C0GMCTRL, C0GMCS, C0CTRL	读/写	$(f_{xx}/f_{CAN} + 1) / (2 + j)$ (MIN.) [‡] $(2 \times f_{xx}/f_{CAN} + 1) / (2 + j)$ (MAX.) [‡]
	C0RGPT, C0TGPT	写	$(f_{xx}/f_{CANMOD} + 1) / (2 + j)$ (MIN.) [‡] $(2 \times f_{xx}/f_{CANMOD} + 1) / (2 + j)$ (MAX.) [‡]
		读	$(3 \times f_{xx}/f_{CANMOD} + 1) / (2 + j)$ (MIN.) [‡] $(4 \times f_{xx}/f_{CANMOD} + 1) / (2 + j)$ (MAX.) [‡]
	C0LIPT, C0LOPT	读	$(3 \times f_{xx}/f_{CANMOD} + 1) / (2 + j)$ (MIN.) [‡] $(4 \times f_{xx}/f_{CANMOD} + 1) / (2 + j)$ (MAX.) [‡]
	C0MCTRLm	写	$(4 \times f_{xx}/f_{CAN} + 1) / (2 + j)$ (MIN.) [‡] $(5 \times f_{xx}/f_{CAN} + 1) / (2 + j)$ (MAX.) [‡]
		读	$(3 \times f_{xx}/f_{CAN} + 1) / (2 + j)$ (MIN.) [‡] $(4 \times f_{xx}/f_{CAN} + 1) / (2 + j)$ (MAX.) [‡]
	C0MDATA01m, C0MDATA0m, C0MDATA1m, C0MDATA23m, C0MDATA2m, C0MDATA3m, C0MDATA45m, C0MDATA4m, C0MDATA5m, C0MDATA67m, C0MDATA6m, C0MDATA7m, C0MDLCm, C0MCONFm, C0MIDLm, C0MIDHm	写 (8 位)	$(4 \times f_{xx}/f_{CANMOD} + 1) / (2 + j)$ (MIN.) [‡] $(5 \times f_{xx}/f_{CANMOD} + 1) / (2 + j)$ (MAX.) [‡]
		写 (16 位)	$(2 \times f_{xx}/f_{CANMOD} + 1) / (2 + j)$ (MIN.) [‡] $(3 \times f_{xx}/f_{CANMOD} + 1) / (2 + j)$ (MAX.) [‡]
		读 (8/16 位)	$(3 \times f_{xx}/f_{CANMOD} + 1) / (2 + j)$ (MIN.) [‡] $(4 \times f_{xx}/f_{CANMOD} + 1) / (2 + j)$ (MAX.) [‡]

访问时必需的时钟数目 $= 3 + i + j + (2 + j) \times k$

注 小数点后的数字只入不舍。

注意事项 下列状态下禁止访问上述寄存器。如果产生等待周期，只能通过复位清除。

- ~ 当 CPU 工作于副时钟，而且主时钟振荡停止时
- ~ 当 CPU 工作于内部振荡时钟时

备注 i: VSWC 寄存器高 4 位的值 (0)
j: VSWC 寄存器低 4 位的值 (0 或 1)

(3) sld 指令和中断请求之间冲突的限制

(a) 描述说明

如果在 sld 指令（该指令紧接在<1>中所列指令之后）之前就立即对<2>中所列指令进行译码，和在<1>中指令完成执行之前的中断请求之间产生冲突，那么在<1>中的指令执行结果就有可能无法储存到寄存器中。

指令 <1>

- ld 指令: ld.b、ld.h、ld.w、ld.bu、ld.hu
- Sld 指令: sld.b、sld.h、sld.w、sld.bu、sld.hu
- 乘法指令: Mul、mulh、mulhi、mulu

指令 <2>

mov reg1、reg2	not reg1、reg2	satsubr reg1、reg2	satsub reg1、reg2
satadd reg1、reg2	satadd imm5、reg2	or reg1、reg2	xor reg1、reg2
and reg1、reg2	tst reg1、reg2	subr reg1、reg2	sub reg1、reg2
add reg1、reg2	add imm5、reg2	cmp reg1、reg2	cmp imm5、reg2
mulh reg1、reg2	shr imm5、reg2	sar imm5、reg2	shl imm5、reg2

<举例说明>

- <i> ld.w [r11]、r10 如果在<iii>中 sld 指令之前立即对<ii>中 mov 指令进行译码，与在<i>中 ld 指令执行完毕之前出现中断请求产生冲突，那么，<i>中指令的执行结果可能无法储存到寄存器中。
 -
 -
 -
- <ii> mov r10、r28
- <iii> sld.w 0x28、r10

(b) 对策

<1> 当使用编译器（CA850）时

请使用 CA850 Ver.2.61 或更新版本，因为它将自动禁止相应指令序列的产生。

<2> 当使用汇编器时

在指令<ii>之后就立即执行 sld 指令时，可以采用以下任一种方法来避免上述操作。

- 在 sld 指令之前立即插入一个 nop 指令。
- 不要使用与在 sld 指令之前立即执行的上述<ii>指令中所使用的相同的寄存器来作为 sld 指令的目的寄存器。

第四章 端口功能

4.1 特性

- 输入/输出端口
 - V850ES/JG3-H: 77
 - 5 V 耐压/N 沟道漏极开路输出可选: 24
 - V850ES/JH3-H: 96
 - 5 V 耐压/N 沟道漏极开路输出可选: 27
- 用 1 位指定输入/输出

4.2 端口基本配置

V850ES/JG3-H 总共包括 77 个由端口 0, 1, 3 至 7, 9, CM, CT 以及 DL 组成的输入/输出端口。
V850ES/JH3-H 总共包括 96 个由端口 0 至 7, 9, CM, CS, CT, DH 以及 DL 组成的输入/输出端口。
端口配置如下所示:

表 4-1. 引脚的输入/输出缓冲器电源供应 (V850ES/JG3-H)

供电	对应引脚
AVREF0	端口 7
AVREF1	端口 1
EVDD	RESET、端口 0、3 至 6、9、CM、CT、DL

表 4-1. 引脚的输入/输出缓冲器电源供应 (V850ES/JH3-H)

供电	对应引脚
AVREF0	端口 7
AVREF1	端口 1
EVDD	RESET、端口 0、2 至 6、9、CM、CS、CT、DH、DL

图 4-1. 端口配置图（V850ES/JG3-H）

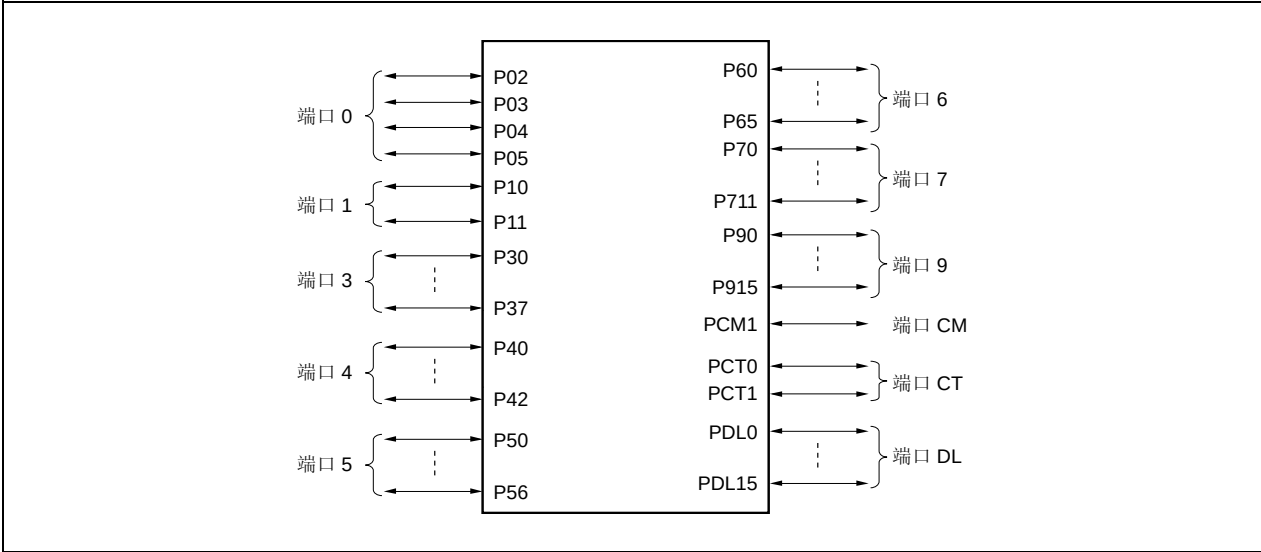
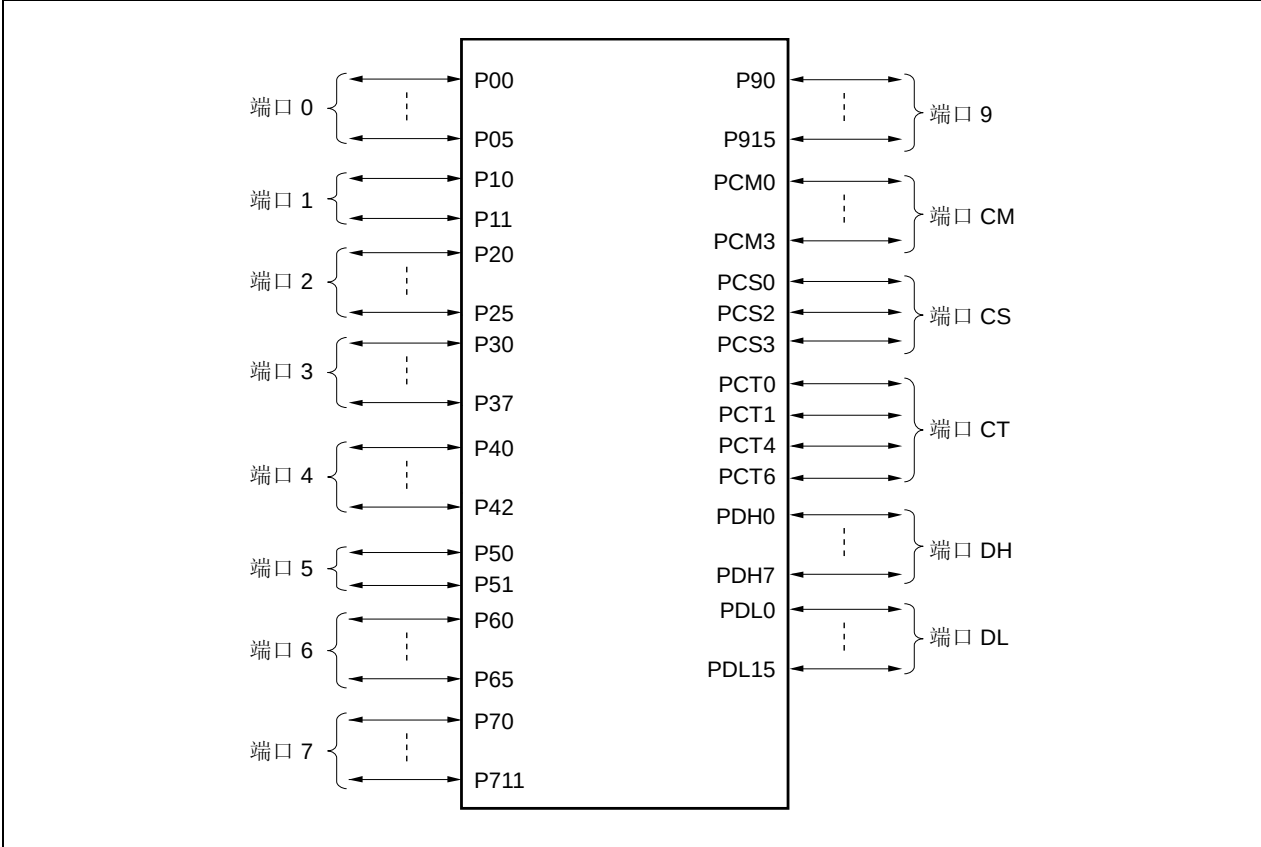


图 4-2. 端口配置图（V850ES/JH3-H）



4.3 端口配置

图 4-3. 端口配置 (V850ES/JG3-H)

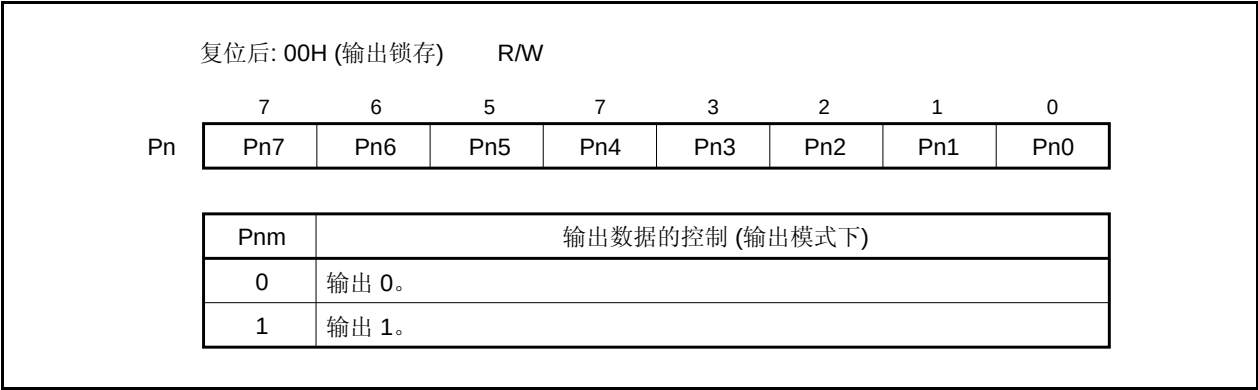
项目	配置
控制寄存器	端口 n 模式寄存器 (PMn: n = 0, 1, 3 至 7, 9, CM, CT, DL) 端口 n 模式控制寄存器 (PMCn: n = 0, 3 至 5, 9, CM, CT, DL) 端口 n 功能控制寄存器 (PFCn: n = 0, 3 至 6, 9) 端口 n 功能控制扩展寄存器 (PFCEn: n = 4 至 6, 9) 端口 n 功能寄存器 (PFn: n = 0, 3 至 5, 9)
端口	I/O: 77

图 4-4. 端口配置 (V850ES/JH3-H)

项目	配置
控制寄存器	端口 n 模式寄存器 (PMn: n = 0 至 7, 9, CM, CS, CT, DH, DL) 端口 n 模式控制寄存器 (PMCn: n = 0, 2 至 6, 9, CM, CS, CT, DH, DL) 端口 n 功能控制寄存器 (PFCn: n = 0, 2 至 6, 9) 端口 n 功能控制扩展寄存器 (PFCEn: n = 4 至 6, 9) 端口 n 功能寄存器 (PFn: n = 0, 2 至 5, 9)
端口	I/O: 96

(1) 端口 n 寄存器 (Pn)

通过写入或读取 Pn 寄存器，将数据从外部器件输入或输出到外部器件。
Pn 寄存器由一个保存输出数据的端口锁存和一个读取引脚状态的电路组成。
Pn 寄存器的每一位对应端口的一个引脚，而且可以按位读出或写入。



无论 PMCn 寄存器的如何设置，Pn 寄存器的数据写入或读取如下所示：

表 4-5. Pn 寄存器的写入和读出

寄存器 PMn 的设置	写入 Pn 寄存器	读取 Pn 寄存器
输出模式 (PMnm 位 = 0)	数据写入输出锁存器 ^注 。 端口模式下 (PMCn = 0)，输出锁存器的内容从相应引脚输出。	读取输出锁存器的值。
输入模式 (PMnm = 1)	数据写入输出锁存器。 不影响引脚状态 ^注 。	读取引脚状态。

注 写入输出锁存器的值一直保留，直到新值写入输出锁存器为止。

(2) 端口 n 模式寄存器 (PMn)

PMn 寄存器用于指定对应端口引脚的输入或输出模式。
该寄存器的每一位对应端口 n 的一个引脚，而且可以按位设定输入或输出模式。

复位后: FFH R/W								
	7	6	5	4	3	2	1	0
PMn	PMn7	PMn6	PMn5	PMn4	PMn3	PMn2	PMn1	PMn0

PMnm	输入/输出模式的控制
0	输出模式
1	输入模式

(3) 端口 n 模式控制寄存器 (PMCn)

PMCn 寄存器指定端口模式或复用功能模式。
该寄存器的每一位对应端口 n 的一个引脚，而且可以按位设定端口的模式。

复位后: 00H								R/W
	7	6	5	4	3	2	1	0
PMCn	PMcN7	PMcN6	PMcN5	PMcN4	PMcN3	PMcN2	PMcN1	PMcN0
PMcNm	工作模式的说明							
0	端口模式							
1	复用功能模式							

(4) 端口 n 功能控制寄存器 (PFCn)

如果引脚有两个复用功能，PFCn 寄存器指定要使用的端口引脚的复用功能。
该寄存器的每一位对应端口 n 的一个引脚，而且可以按位设定端口的复用功能。

复位后: 00H

R/W

76543210

PFCn

PFCn7

PFCn6

PFCn5

PFCn4

PFCn3

PFCn2

PFCn1

PFCn0

PFCnm	复用功能说明
0	复用功能 1
1	复用功能 2

(5) 端口 n 功能控制扩展寄存器 (PFCEn)

如果引脚有三个或更多的复用功能，PFCEn 寄存器指定要使用的引脚的复用功能。
该寄存器的每一位对应端口 n 的一个引脚，而且可以按位设定端口的复用功能。

复位后: 00H R/W

	7	6	5	4	3	2	1	0
PFCEn	PFCEn7	PFCEn6	PFCEn5	PFCEn4	PFCEn3	PFCEn2	PFCEn1	PFCEn0

	7	6	5	4	3	2	1	0
PFCn	PFCn7	PFCn6	PFCn5	PFCn4	PFCn3	PFCn2	PFCn1	PFCn0

PFCEnm	PFCnm	复用功能说明
0	0	复用功能 1
0	1	复用功能 2
1	0	复用功能 3
1	1	复用功能 4

(6) 端口 n 功能寄存器 (PFn)

PFn 寄存器指定正常输出还是 N 沟道漏极开路输出。
该寄存器的每一位对应端口 n 的一个引脚，而且可以按位设定端口引脚的输出模式。

复位后: 00H R/W

	7	6	5	4	3	2	1	0
PFn	PFn7	PFn6	PFn5	PFn4	PFn3	PFn2	PFn1	PFn0

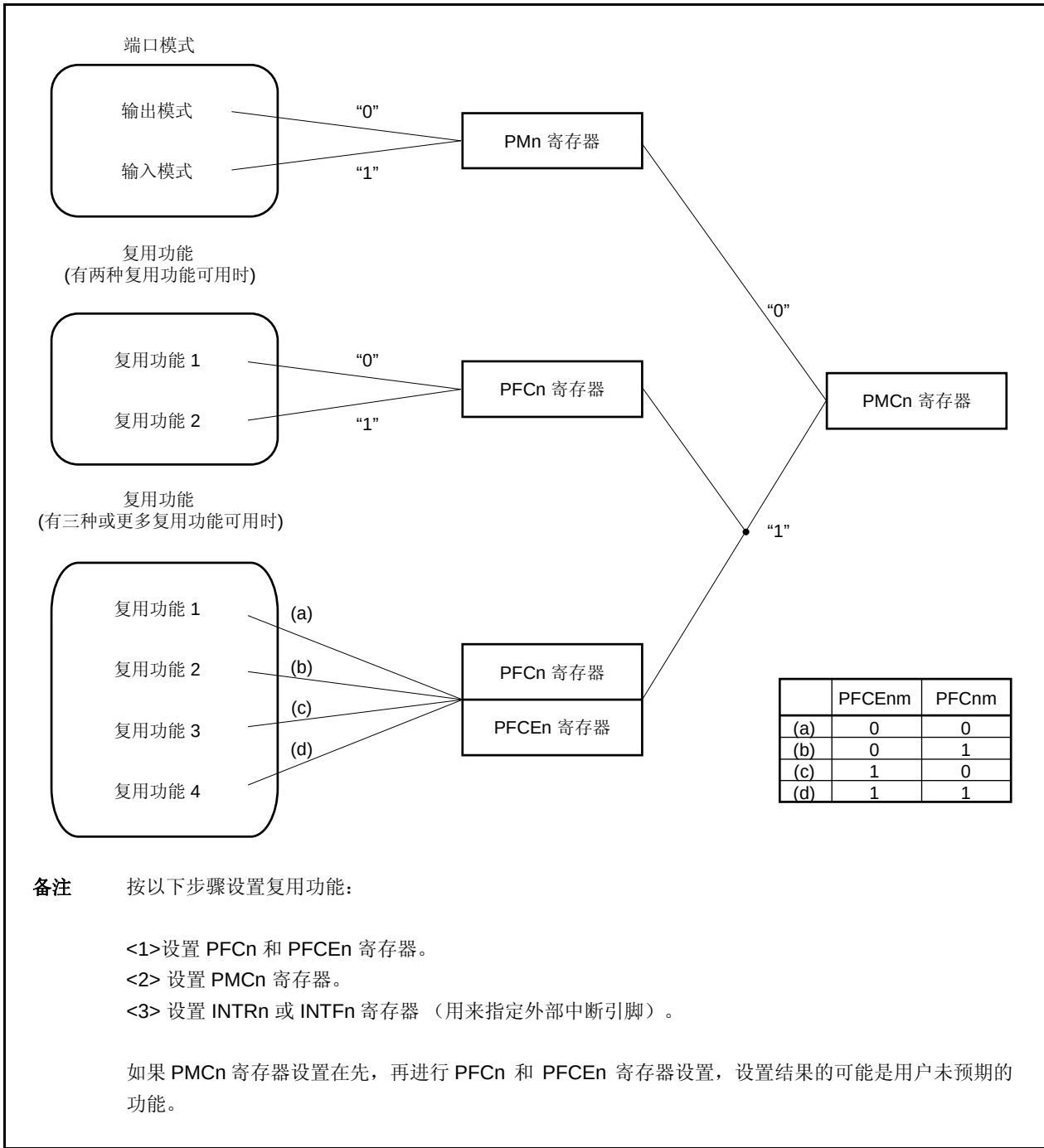
PFnm ^注	正常输出/N沟道漏极开路输出的控制
0	正常输出 (CMOS 输出)
1	N沟道漏极开路输出

注 端口模式下 (PMCnm 位=0)，只有 PMn 寄存器的 PMnm 位为 0 时（指定为输出模式时），PFn 寄存器的 PFnm 位才有效。当 PMnm 位为 1（指定为输入模式时），PFn 寄存器设置值无效。

(7) 端口设置

端口设置如下所示：

图 4-3.各寄存器的设置和引脚功能



4.3.1 端口 0

端口 0 是 2/4 位 (V850ES/JG3-H) /6 位 (V850ES/JH3-H)，其输入/输出设置可以按位控制。

端口 0 包括以下复用功能引脚：

表 4-6.端口 0 复用功能引脚

引脚名称	引脚编号		复用功能引脚名称	I/O	备注
	V850ES/ JG3-H	V850ES/ JH3-H			
P00	—	8	INTP00	输入	可选作 N 沟道漏极开路输出
P01	—	9	INTP01	输入	
P02	6	6	NMI	输入	
P03	7	7	INTP02/ADTRG/UCLK	输入	
P04	20	26	INTP03	输入	
P05	21	27	INTP04	输入	

注意事项 在复用功能的输入模式下，P00 至 P05 具有迟滞特性，但在端口模式下不具有迟滞特性。

(1) 端口 0 寄存器 (P0)

(a) V850ES/JG3-H

复位后: 00H (输出锁存) R/W 地址: FFFFF400H

	7	6	5	4	3	2	1	0
P0	0	0	P05	P04	P03	P02	0	0

P0n	输出数据控制 (输出模式下) (n = 2 至 5)
0	输出 0。
1	输出 1。

(b) V850ES/JH3-H

复位后: 00H (输出锁存) R/W 地址: FFFFF400H

	7	6	5	4	3	2	1	0
P0	0	0	P05	P04	P03	P02	P01	P00

P0n	输出数据控制 (输出模式下) (n = 2 至 5)
0	输出 0。
1	输出 1。

(2) 端口 0 模式寄存器 (PM0)

(a) V850ES/JG3-H

复位后: FFH R/W 地址: FFFFFFF420H

	7	6	5	4	3	2	1	0
PM0	1	1	PM05	PM04	PM03	PM02	1	1

PM0n	I/O 模式控制 (n = 2 至 5)
0	输出模式
1	输入模式

(b) V850ES/JH3-H

复位后: FFH R/W 地址: FFFFFFF420H

	7	6	5	4	3	2	1	0
PM0	1	1	PM05	PM04	PM03	PM02	PM01	PM00

PM0n	I/O 模式控制 (n = 0 至 5)
0	输出模式
1	输入模式

(3) 端口 0 模式控制寄存器 (PMC0)

(1/2)

(a) V850ES/JG3-H

复位后: 00H R/W 地址: FFFFF440H

	7	6	5	4	3	2	1	0
PMC0	0	0	PMC05	PMC04	PMC03	PMC02	0	0

PMC05	P05引脚工作模式说明
0	I/O 端口
1	INTP04 输入

PMC04	P04 引脚工作模式说明
0	I/O 端口
1	INTP03 输入

PMC03	P03 引脚工作模式说明
0	I/O 端口
1	INTP02 输入/ADTRG 输入/UCLK 输入

PMC02	P02 引脚工作模式说明
0	I/O 端口
1	NMI 输入

(b) V850ES/JH3-H

复位后: 00H R/W 地址: FFFFF440H

	7	6	5	4	3	2	1	0
PMC0	0	0	PMC05	PMC04	PMC03	PMC02	PMC01	PMC00
PMC05	P05 引脚工作模式说明							
0	I/O 端口							
1	INTP04 输入							
PMC04	P04 引脚工作模式说明							
0	I/O 端口							
1	INTP03 输入							
PMC03	P03 引脚工作模式说明							
0	I/O 端口							
1	INTP02 输入/ADTRG 输入/UCLK 输入							
PMC02	P02 引脚工作模式说明							
0	I/O 端口							
1	NMI 输入							
PMC01	P01 引脚工作模式说明							
0	I/O 端口							
1	INTP01 输入							
PMC00	P00 引脚工作模式说明							
0	I/O 端口							
1	INTP00 输入							

(4) 端口 0 功能控制寄存器 (PFC0)

复位后: 00H R/W 地址: FFFFF460H

	7	6	5	4	3	2	1	0
PFC0	0	0	0	0	PFC03	0	0	0

备注 关于复用功能规范的详情，参见 4.3.1 (6) 端口 0 复用功能规范。

(5) 端口 0 功能控制扩展寄存器 (PFCE0)

复位后: 00H R/W 地址: FFFFFFF700H

	7	6	5	4	3	2	1	0
PFCE0	0	0	0	0	PFCE03	0	0	0

备注 关于复用功能规范的详情, 参见 4.3.1 (6) 端口 0 复用功能规范。

(6) 端口 0 复用功能规范

PFCE03	PFC03	P03 引脚复用功能规范
0	0	INTP02 输入
0	1	ADTRG 输入
1	0	UCLK 输入
1	1	禁止设置

(7) 端口 0 功能寄存器 (PF0)

(a) V850ES/JG3-H

复位后: 00H R/W 地址: FFFFFC60H

	7	6	5	4	3	2	1	0
PF0	0	0	PF05	PF04	PF03	PF02	0	0

PF0n	正常输出或 N沟道漏极开路输出控制 (n = 2-5)
0	正常输出
1	N沟道漏极开路输出

(b) V850ES/JH3-H

复位后: 00H R/W 地址: FFFFFC60H

	7	6	5	4	3	2	1	0
PF0	0	0	PF05	PF04	PF03	PF02	PF01	PF00

PF0n	正常输出或 N沟道漏极开路输出控制 (n = 0 至 5)
0	正常输出
1	N沟道漏极开路输出

4.3.2 端口 1

端口 1 为 2 位宽度输入输出端口，其输入输出设置可以按位控制。

端口 1 包括以下复用功能引脚：

表 4-7.端口 1 复用功能引脚

引脚名称	引脚编号		复用功能引脚名称	I/O	备注
	V850ES/ JG3-H	V850ES/ JH3-H			
P10	3	3	AN00	输出	—
P11	4	4	AN01	输出	

注意事项 电源开启时，即使在复位期间 P10 和 P11 引脚也可能暂时输出不确定电平。

(1) 端口 1 寄存器 (P1)

复位后: 00H (输出锁存) R/W 地址: FFFFF402H

	7	6	5	4	3	2	1	0
P1	0	0	0	0	0	0	P11	P10

P1n	输出数据控制 (输出模式下) (n = 0, 1)
0	输出 0。
1	输出 1。

注意事项 D/A 转换期间不要读取或写入 P1 寄存器（参见 16.4.3 注意事项）。

(2) 端口 1 模式寄存器 (PM1)

复位后: 00H R/W 地址: FFFFF422H

	7	6	5	4	3	2	1	0
PM1	0	0	0	0	0	0	PM11	PM10

PM1n	I/O 模式控制 (n = 0, 1)
0	输出模式
1	输入模式

- 注意事项
1. P1n 用作复用功能 (ANOn 引脚输出) 时，将 PM1n 位设置为 1。
 2. 当 PM10 和 PM11 其中之一用作输入/输出端口而另一个用作 D/A 输出引脚时，确保 D/A 输出期间输入/输出端口电平不改变。

4.3.3 端口 2（仅限于V850ES/JH3-H）

端口 2 为 6 位宽度输入输出端口，其输入输出设置可以按位控制。

端口 2 包括以下复用功能引脚：

表 4-8.端口 2 复用功能引脚

引脚名称	引脚编号		复用功能引脚名称	I/O	备注
	V850ES/ JG3-H	V850ES/ JH3-H			
P20	—	32	TIAB03/KR2/TOAB03/RTP02	I/O	可选作 N 沟道漏极开路输出
P21	—	33	SIF2/KR3/TIAB00/TOAB00 /RTP03	I/O	
P22	—	34	SOF2/KR4/RTP04	I/O	
P23	—	35	SCKF2/KR5/RTP05	I/O	
P24	—	36	INTP05	输入	
P25	—	28	INTP06	输入	

注意事项 在复用功能引脚的输入模式下，P20 至 P25 具有迟滞特性，但在端口模式下不具有迟滞特性。

（1）端口 2 寄存器（P2）

复位后：00H (输出锁存) R/W 地址：FFFF404H

	7	6	5	4	3	2	1	0
P2	0	0	P25	P24	P23	P22	P21	P20

P2n	输出数据控制 (输出模式下) (n = 0 至 5)
0	输出 0。
1	输出 1。

（2）端口 2 模式寄存器（PM2）

复位后：FFH R/W 地址：FFFF424H

	7	6	5	4	3	2	1	0
PM2	1	1	PM25	PM24	PM23	PM22	PM21	PM20

PM2n	I/O 模式控制 (n = 0 至 5)
0	输出模式
1	输入模式

(3) 端口 2 模式控制寄存器 (PMC2)

复位后: 00H R/W 地址: FFFFF444H

	7	6	5	4	3	2	1	0
PMC2	0	0	PMC25	PMC24	PMC23	PMC22	PMC21	PMC20
PMC25	P25引脚工作模式说明							
0	I/O 端口							
1	INTP06 输入							
PMC24	P24 引脚工作模式说明							
0	I/O 端口							
1	INTP05 输入							
PMC23	P23 引脚工作模式说明							
0	I/O 端口							
1	SCKF2 I/O/KR5 输入/RTP05 输出							
PMC22	P22 引脚工作模式说明							
0	I/O 端口							
1	SOF2 输出/KR4 输入/RTP04 输出							
PMC21	P21 引脚工作模式说明							
0	I/O 端口							
1	SIF2 输出/KR3 输入/TIAB00 输入/TOAB00 输出/RTP03 输出							
PMC20	P20 引脚工作模式说明							
0	I/O 端口							
1	TIAB03 输入/KR2 输入/TOAB03 输出/RTP02 输出							

(4) 端口 2 功能控制寄存器 (PFC2)

复位后: 00H R/W 地址: FFFFF464H

	7	6	5	4	3	2	1	0
PFC2	0	0	0	0	PFC23	PFC22	PFC21	PFC20

备注 关于复用功能规范的详情，参见 4.3.3 (6) 端口 2 复用功能规范。

(5) 端口 2 功能控制扩展寄存器 (PFCE2)

复位后: 00H R/W 地址: FFFFF704H

	7	6	5	4	3	2	1	0
PFCE2	0	0	0	0	PFCE23	PFCE22	PFCE21	PFCE20

备注 关于复用功能规范的详情, 参见 4.3.3 (6) 端口 2 复用功能规范。

(6) 端口 2 复用功能规范

PFCE23	PFC23	P23 引脚复用功能规范
0	0	SCKF2 I/O
0	1	KR5 输入
1	0	RTP05 输出
1	1	禁止设置

PFCE22	PFC22	P22 引脚复用功能规范
0	0	SOF2 输出
0	1	KR4 输入
1	0	RTP04 输出
1	1	禁止设置

PFCE21	PFC21	P21 引脚复用功能规范
0	0	SIF2 输入
0	1	KR3 输入/TIAB00 输入 [※]
1	0	TOAB00 输出
1	1	RTP03 输出

注 KR3 和 TIAB00 是复用功能。当该引脚用作 TIAB00 引脚时, 禁止 KR3 引脚按键返回检测复用功能 (将 KRM.KRM3 位清除为 0)。同样, 当该引脚用作 KRn 引脚时, 禁止 TIAB00 引脚边沿检测复用功能 (TAB0IOC1.TAB0TIG0, TAB0TIG1 位 = 00 B, TAB0IOC2 寄存器 = 00H)。

PFCE20	PFC20	P20 引脚复用功能规范
0	0	TIAB03 输入
0	1	KR2 输入
1	0	TOAB03 输出
1	1	RTP02 输出

(7) 端口 2 功能寄存器 (PF2)

复位后: 00H R/W 地址: FFFFFFFC64H

	7	6	5	4	3	2	1	0
PF2	0	PF26	PF25	PF24	PF23	PF22	PF21	PF20

PF2n	正常输出或 N沟道漏极开路输出的控制 (n = 0 至 6)
0	正常输出
1	N沟道漏极开路输出

4.3.4 端口 3

端口 3 是 10 位宽度端口，可以按位控制 I/O。
端口 3 包括以下复用功能引脚：

表 4-9.端口 3 复用功能引脚

引脚名称	引脚编号		复用功能引脚名称	I/O	备注
	V850ES/ JG3-H	V850ES/ JH3-H			
P30	25	37	TXDC0/SOF4/INTP07	I/O	可选作 N 沟道漏极开路输出
P31	26	38	RXDC0/SIF4/INTP08	输入	
P32	27	39	ASCKC0/SCKF4/TIAA00/TOAA00	I/O	
P33	28	40	TIAA01/TOAA01/RTCDIV/RTCCL	I/O	
P34	29	41	TIAA10/TOAA10/TOAA1OFF/INTP09	I/O	
P35	30	42	TIAA11/TOAA11/RTC1HZ	I/O	
P36	31	43	TXDC3/SCL00/CTXD0 [※] /UDMARQ0	I/O	
P37	32	44	RXDC3/SDA00/CRXD0 [※] /UDMAAK0	I/O	

注 仅限于μPD70F3770，70F3771

注意事项 在复用功能引脚的输入模式下，P30 至 P37 具有迟滞特性，但在端口模式下不具有迟滞特性。

(1) 端口 3 寄存器（P3）

复位后: 00H (输出锁存) R/W 地址: FFFFF406H

	7	6	5	4	3	2	1	0
P3	P37	P36	P35	P34	P33	P32	P31	P30

P3n	输出数据控制 (输出模式下) (n = 0至 7)
0	输出 0。
1	输出 1。

(2) 端口 3 模式寄存器（PM3）

复位后: FFH R/W 地址: FFFFF426H								
PM3	7	6	5	4	3	2	1	0
	PM37	PM36	PM35	PM34	PM33	PM32	PM31	PM30
PM3n	I/O 模式控制 (n = 0 至 7)							
0	输出模式							
1	输入模式							

(3) 端口 3 模式控制寄存器 (PMC3)

复位后: 00H R/W 地址: FFFFF446H

76543210

PMC3

PMC37	PMC36	PMC35	PMC34	PMC33	PMC32	PMC31	PMC30

PMC37	P37 引脚工作模式说明
0	I/O 端口
1	RXDC3 输入/SDA00 I/O/CRXD 输入 ^注 /UDMAAK0 输出

PMC36	P36 引脚工作模式说明
0	I/O 端口
1	TXDC3 输出/SCL00 I/O/CTXD0 输出 ^注 /UDMARQ0 输入

PMC35	P35 引脚工作模式说明
0	I/O 端口
1	TIAA11 输入/TOAA11 输出/RTC1HZ 输出

PMC34	P34 引脚工作模式说明
0	I/O 端口
1	TIAA10 输入/TOAA10 输出/TOAA1OFF 输入/INTP09 输入

PMC33	P33 引脚工作模式说明
0	I/O 端口
1	TIAA01 输入/TOAA01 输出/RTCDIV 输出/RTCCL 输出

PMC32	P32 引脚工作模式说明
0	I/O 端口
1	ASCKA0 输入/SCKF4 I/O/TIAA00 输入/TOAA00 输出

PMC31	P31 引脚工作模式说明
0	I/O 端口
1	RXDC0 输入/SIF4 输入/INTP08 输入

PMC30	P30 引脚工作模式说明
0	I/O 端口
1	TXDC0 输出/SOF4 输出/INTP07 输入

注 仅限于 μ PD70F3770, 70F3771

(4) 端口 3 功能控制寄存器 (PFC3)

复位后: 00H R/W 地址: FFFFF466H

	7	6	5	4	3	2	1	0
PFC3	PFC37	PFC36	PFC35	PFC34	PFC33	PFC32	PFC31	PFC30

备注 关于复用功能规范的详情, 参见 4.3.4 (6) 端口 3 复用功能规范。

(5) 端口 3 功能控制扩展寄存器 (PFCE3)

复位后: 00H R/W 地址: FFFFF706H

	7	6	5	4	3	2	1	0
PFCE3	PFCE37	PFCE36	PFCE35	PFCE34	PFCE33	PFCE32	PFCE31	PFCE30

备注 关于复用功能规范的详情, 参见 4.3.4 (6) 端口 3 复用功能规范。

(6) 端口 3 复用功能规范

PFCE37	PFC37	P37 引脚复用功能规范
0	0	RXDC3 输入
0	1	SDA00 I/O
1	0	CRXD0 输入 ^注
1	1	UDMAAK0 输出

注 仅限于 μ PD70F3770, 70F3771

PFCE36	PFC36	P36 引脚复用功能说明
0	0	TXDC3 输出
0	1	SCL00 I/O
1	0	CTXD0 输出 ^注
1	1	UDMARQ0 输入

注 仅限于 μ PD70F3770, 70F3771

PFCE35	PFC35	P35 引脚复用功能说明
0	0	TIAA11 输入
0	1	TOAA11 输出
1	0	RTC1HZ 输出
1	1	禁止设置

PFCE34	PFC34	P34 引脚复用功能说明
0	0	TIAA10 输入
0	1	TOAA10 输出
1	0	TOAA1OFF 输入/INTP09 输入 ^注
1	1	禁止设置

注 TOAA1OFF 和 INTP09 是复用功能。当该引脚用作 TOAA1OFF 引脚时，禁止使用 INTP09 引脚边沿检测的复用功能。同样，当用作 INTP09 引脚时，停止高阻输出控制器。

PFCE33	PFC33	P33 引脚复用功能说明
0	0	TIAA01 输入
0	1	TOAA01 输出
1	0	RTCDIV 输出
1	1	RTCCL 输出

PFCE32	PFC32	P32 引脚复用功能说明
0	0	ASCKC0 输入
0	1	SCKF4 I/O
1	0	TIAA00 输入
1	1	TOAA00 输出

PFCE31	PFC31	P31 引脚复用功能说明
0	0	RXDC0 输入
0	1	SIF4 输入
1	0	INTP08 输入
1	1	禁止设置

PFCE30	PFC30	P30 引脚复用功能说明
0	0	TXDC0 输出
0	1	SOF4 输出
1	0	INTP07 输入
1	1	禁止设置

(7) 端口 3 功能寄存器 (PF3)

复位后: 00H		R/W	地址:		FFFFFFC66H			
	7	6	5	4	3	2	1	0
PF3	PF37	PF36	PF35	PF34	PF33	PF32	PF31	PF30

PF3n	正常输出或 N沟道漏极开路输出控制 (n = 0 至 7)
0	正常输出 (CMOS 输出)
1	N沟道漏极开路输出

4.3.5 端口 4

端口 4 是 3 位宽度端口，可以按位控制 I/O。

端口 4 包括以下复用功能引脚：

表 4-10.端口 4 复用功能引脚

引脚名称	引脚编号		复用功能引脚名称	I/O	备注
	V850ES/ JG3-H	V850ES/ JH3-H			
P40	22	29	SIF0/TXDC4/SDA01	I/O	可选作 N 沟道漏极开路输出
P41	23	30	SOF0/RXDC4/SCL01	I/O	
P42	24	31	SCKF0/INTP10	I/O	

注意事项 在复用功能引脚的输入模式下，P40 至 P42 具有迟滞特性，但在端口模式下不具有迟滞特性。

(1) 端口 4 寄存器 (P4)

复位后: 00H (输出锁存) R/W 地址: FFFFF408H

	7	6	5	4	3	2	1	0
P4	0	0	0	0	0	P42	P41	P40

P4n	输出数据控制 (输出模式下) (n = 0 至 2)
0	输出 0。
1	输出 1。

(2) 端口 4 模式寄存器 (PM4)

复位后: FFH R/W 地址: FFFFF428H

	7	6	5	4	3	2	1	0
PM4	1	1	1	1	1	PM42	PM41	PM40

PM4n	I/O 模式控制 (n = 0 至 2)
0	输出模式
1	输入模式

(3) 端口 4 模式控制寄存器 (PMC4)

复位后: 00H R/W 地址: FFFFF448H								
PMC4	7	6	5	4	3	2	1	0
	0	0	0	0	0	PMC42	PMC41	PMC40
PMC42		P42 引脚工作模式说明						
0		I/O 端口						
1		SCKF0 I/O/INTP10 输入						
PMC41		P41 引脚工作模式说明						
0		I/O 端口						
1		SOF0 输出/RXDC4 输入/SCL01 I/O						
PMC40		P40 引脚工作模式说明						
0		I/O 端口						
1		SIF0 输入/TXDC4 输出/SDA01 I/O						

(4) 端口 4 功能控制寄存器 (PFC4)

复位后: 00H R/W 地址: FFFFF468H								
PFC4	7	6	5	4	3	2	1	0
	0	0	0	0	0	PFC42	PFC41	PFC40
备注		关于复用功能规范的详情, 参见 4.3.5 (6) 端口 4 复用功能规范。						

(5) 端口 4 功能控制扩展寄存器 (PFCE4)

复位后: 00H R/W 地址: FFFFF708H								
PFCE4	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	PFCE41	PFCE40
备注		关于复用功能规范的详情, 参见 4.3.5 (6) 端口 4 复用功能规范。						

(6) 端口 4 复用功能规范

PFC42	P42引脚复用功能规范
0	SCKF0 I/O
1	INTP10 输入

PFCE41	PFC41	P41 引脚复用功能规范
0	0	SOF0 输出
0	1	RXDC4 输入
1	0	SCL01 I/O
1	1	禁止设置

PFCE40	PFC40	P40 引脚复用功能规范
0	0	SIF0 输入
0	1	TXDC4 输出
1	0	SDA01 I/O
1	1	禁止设置

(7) 端口 4 功能寄存器 (PF4)

复位后: 00H R/W 地址: FFFFFC68H

	7	6	5	4	3	2	1	0
PF4	0	0	0	0	0	PF42	PF41	PF40

PF4n	正常输出或 N沟道漏极开路输出控制 (n = 0 至 2)
0	正常输出(CMOS 输出)
1	N沟道漏极开路输出

4.3.6 端口 5

端口 5 是 6 位 (V850ES/JG3-H) /2 位 (V850ES/JH3-H) 端口，可以按位控制输入/输出。

端口 5 包括以下复用功能引脚：

表 4-11. 端口 5 复用功能引脚

引脚名称	引脚编号		复用功能引脚名称	I/O	备注
	V850ES/ JG3-H	V850ES/ JH3-H			
P50	35	47	TIAB01/KR0/TOAB01/RTP00 /UDMARQ1	I/O	可选作 N 沟道漏极开路输出
P51	36	48	TIAB02/KR1/TOAB02/RTP01 /UDMAAK1	I/O	
P52	37	—	TIAB03/KR2/TOAB13/RTP02 /DDI [‡]	I/O	
P53	38	—	SIF2/TIAB00/KR3/TOAB10 /RTP03/DDO [‡]	I/O	
P54	39	—	SOF2/KR4/RTP04/DCK [‡]	I/O	
P55	40	—	SCKF2/KR5/RTP05/DMS [‡]	I/O	
P56	41	—	INTP05/DRST [‡]	输入	

注 DDI, DDO, DCK, DMS 和 $\overline{\text{DRST}}$ 引脚用于片上调试。

如果不使用片上调试功能，当通过 $\overline{\text{RESET}}$ 引脚进行复位释放时和 OCDM.OCDM0 位清 0 时，将 P05/INTP02/ $\overline{\text{DRST}}$ 引脚固定为低电平。

细节参见 4.5.3 片上调试引脚的注意事项。

- 注意事项 1. 电源开启时，即使在复位期间 P53 引脚也可能输出一个不确定电平。
2. 在复用功能引脚的输入模式下，P50 至 P56 具有滞后特性，但在端口模式下不具有滞后特性。

(1) 端口 5 寄存器 (P5)

(a) V850ES/JG3-H

复位后: 00H (输出锁存) R/W 地址: FFFFF40AH

	7	6	5	4	3	2	1	0
P5	0	P56	P55	P54	P53	P52	P51	P50

P5n	输出数据控制 (输出模式下) (n = 0 至 6)
0	输出 0。
1	输出 1。

(b) V850ES/JH3-H

复位后: 00H (输出锁存) R/W 地址: FFFFF40AH

	7	6	5	4	3	2	1	0
P5	0	0	0	0	0	0	P51	P50

P5n	输出数据控制 (输出模式下) (n = 0, 1)
0	输出 0。
1	输出 1。

(2) 端口 5 模式寄存器 (PM5)

(a) V850ES/JG3-H

复位后: FFH R/W 地址: FFFFF42AH

	7	6	5	4	3	2	1	0
PM5	1	PM56	PM55	PM54	PM53	PM52	PM51	PM50

PM5n	I/O 模式控制 (n = 0 至 6)
0	输出模式
1	输入模式

(b) V850ES/JH3-H

复位后: FFH R/W 地址: FFFFF42AH

	7	6	5	4	3	2	1	0
PM5	1	1	1	1	1	1	PM51	PM50

PM5n	I/O 模式控制 (n = 0, 1)
0	输出模式
1	输入模式

(3) 端口 5 模式控制寄存器 (PMC5)

(a) V850ES/JG3-H

复位后: 00H R/W 地址: FFFFF44AH

	7	6	5	4	3	2	1	0
PMC5	0	PMC56	PMC55	PMC54	PMC53	PMC52	PMC51	PMC50
PMC65	P56 引脚工作模式说明							
0	I/O 端口							
1	INTP05 输入							
PMC55	P55 引脚工作模式说明							
0	I/O 端口							
1	SCKF2 I/O/KR5 输入/RTP05 输出							
PMC54	P54 引脚工作模式说明							
0	I/O 端口							
1	SOF2 输出/KR4 输入/RTP04 输出							
PMC53	P53 引脚工作模式说明							
0	I/O 端口							
1	SIF2 输入/KR3 输入/TIAB00 输入/TOAB00 输出/RTP03 输出							
PMC52	P52 引脚工作模式说明							
0	I/O 端口							
1	TIAB03 输入/KR2 输入/TOAB03 输出/RTP02 输出							
PMC51	P51 引脚工作模式说明							
0	I/O 端口							
1	TIAB02 输入/KR1 输入/TOAB02 输出/RTP01 输出/UDMAAK1 输出							
PMC50	P50 引脚工作模式说明							
0	I/O 端口							
1	TIAB01 输入/KR0 输入/TOAB01 输出/RTP00 输出/UDMARQ1 输入							

(b) V850ES/JH3-H

复位后: 00H R/W 地址: FFFFF44AH

	7	6	5	4	3	2	1	0
PMC5	0	0	0	0	0	0	PMC51	PMC50
PMC51	P51 引脚工作模式说明							
0	I/O 端口							
1	TIAB02 输入/KR1 输入/TOAB02 输出/RTP01 输出/UDMAAK1 输出							
PMC50	P50 引脚工作模式说明							
0	I/O 端口							
1	TIAB01 输入/KR0 输入/TOAB01 输出/RTP00 输出/UDMARQ1 输入							

(4) 端口 5 功能控制寄存器 (PFC5)

(a) V850ES/JG3-H

复位后: 00H R/W 地址: FFFFF46AH

	7	6	5	4	3	2	1	0
PFC5	0	0	PFC55	PFC54	PFC53	PFC52	PFC51	PFC50

(b) V850ES/JH3-H

复位后: 00H R/W 地址: FFFFF46AH

	7	6	5	4	3	2	1	0
PFC5	0	0	0	0	0	0	PFC51	PFC50

备注 关于复用功能规范的详情, 参见 4.3.6 (6) 端口 5 复用功能规范。

(5) 端口 5 功能控制扩展寄存器 (PFCE5)

(a) V850ES/JG3-H

复位后: 00H R/W 地址: FFFFF70AH

	7	6	5	4	3	2	1	0
PFCE5	0	0	PFCE55	PFCE54	PFCE53	PFCE52	PFCE51	PFCE50

(b) V850ES/JH3-H

复位后: 00H R/W 地址: FFFFF70AH

	7	6	5	4	3	2	1	0
PFCE5	0	0	0	0	0	0	PFCE51	PFCE50

备注 关于复用功能规范的详情, 参见 4.3.6 (6) 端口 5 复用功能规范。

(6) 端口 5 复用功能规范

PFCE55 ^{註1}	PFCE50 ^{註1}	P55 引脚复用功能说明 ^{註1}
0	0	SCKF2 I/O
0	1	KR5 输入
1	0	RTP05 输出
1	1	禁止设置

PFCE54 ^{注1}	PFC54 ^{注1}	P54 引脚复用功能说明 ^{注1}
0	0	SOF2 输出
0	1	KR4 输入
1	0	RTP04 输出
1	1	禁止设置

PFCE53 ^{注1}	PFC53 ^{注1}	P53引脚复用功能说明 ^{注1}
0	0	SIF2 输入
0	1	TIAB00 输入/KR3 ^{注2} 输入
1	0	TOAB00 输出
1	1	RTP03 输出

PFCE52 ^{注1}	PFC52 ^{注1}	P52 引脚复用功能说明 ^{注1}
0	0	TIAB03 输入/KR2 ^{注2} 输入
0	1	TOAB03 输出
1	0	RTP02 输出
1	1	禁止设置

PFCE51	PFC51	P51 引脚复用功能说明
0	0	TIAB02 输入/KR1 ^{注2} 输入
0	1	TOAB02 输出
1	0	RTP01 输出
1	1	UDMAK1 输出

PFCE50	PFC50	P50 引脚复用功能说明
0	0	TIAB01 输入/KR0 ^{注2} 输入
0	1	TOAB01 输出
1	0	RTP00 输出
1	1	UDMARQ1 输入

注 1. 仅限于 V850ES/JG3-H

2. KRn 和 TIAB0 是复用功能。当该引脚用作 TIAB0 引脚时，禁止 KRn 引脚按键返回检测复用功能（将 KRM.KRMn 位清除为 0）。同样，当该引脚用作 KRn 引脚时，禁止 TIAB0 引脚边沿检测复用功能（n = 0 至 3，m = 0 至 3）。

引脚名称	用作 TIAB0m 引脚	用作 KRn 引脚
KR0/TIAB01	KRM.KRM0 位 = 0	TAB0IOC1.TAB0TIG2, TAB0TIG3 位 = 0
KR1/TIAB02	KRM.KRM1 位 = 0	TAB0IOC1.TAB0TIG4, TAB0TIG5 位 = 0
KR2/TIAB03	KRM.KRM2 位 = 0	TAB0IOC1.TAB0TIG6, TAB0TIG7 位 = 0
KR3/TIAB00	KRM.KRM3 位 = 0	TAB0IOC1.TAB0TIG0, TAB0TIG1 位 = 0 TAB0IOC2.TAB0EES0, TAB0EES1 位 = 0 TAB0IOC2.TAB0ETS0, TAB0ETS1 位 = 0

(7) 端口 5 功能寄存器 (PF5)

(a) V850ES/JG3-H

复位后: 00H R/W 地址: FFFFFFFC6AH

	7	6	5	4	3	2	1	0
PF5	0	PF56	PF55	PF54	PF53	PF52	PF51	PF50

PF5n	正常输出或 N 沟道漏极开路输出控制 (n = 0 至 6)
0	正常输出 (CMOS 输出)
1	N 沟道漏极开路输出

(b) V850ES/JH3-H

复位后: 00H R/W 地址: FFFFFFFC6AH

	7	6	5	4	3	2	1	0
PF5	0	0	0	0	0	0	PF51	PF50

PF5n	正常输出或 N 沟道漏极开路输出控制 (n = 0, 1)
0	正常输出 (CMOS 输出)
1	N 沟道漏极开路输出

4.3.7 端口 6

端口 6 为 6 位宽度输入输出端口，其输入输出设置可以按位控制。
端口 6 包括以下复用功能引脚：

表 4-12.端口 6 复用功能引脚

引脚名称	引脚编号		复用功能引脚名称	I/O	备注
	V850ES/ JG3-H	V850ES/ JH3-H			
P60	65	90	TOAB1T1/TOAB11/TIAB11 /WAIT [‡]	I/O	—
P61	66	91	TOAB1B1/TOAB10/TIAB10 /RD [‡]	I/O	
P62	67	92	TOAB1T2/TOAB12/TIAB12 /ASTB [‡]	I/O	
P63	68	93	TOAB1B2/TRGAB1/CS0 [‡]	I/O	
P64	69	94	TOAB1T3/TOAB13/TIAB13 /CS2 [‡]	I/O	
P65	70	95	TOAB1B3/EVTAB1/CS3 [‡]	I/O	

注 仅限于 V850ES/JG3-H

注意事项 在复用功能引脚的输入模式下，P60 至 P65 具有迟滞特性，但在端口模式下不具有迟滞特性。

(1) 端口 6 寄存器（P6）

复位后: 00H (输出锁存) R/W 地址: FFFFF40CH								
P6	7	6	5	4	3	2	1	0
	0	0	P65	P64	P63	P62	P61	P60

P6n	输出数据控制(输出模式下) (n = 0 至 5)
0	输出 0。
1	输出 1。

(2) 端口 6 模式寄存器 (PM6)

复位后: FFH R/W 地址: FFFFF42CH

	7	6	5	4	3	2	1	0
PM6	1	1	PM65	PM64	PM63	PM62	PM61	PM60

PM6n	I/O模式控制 (n = 0 至 5)
0	输出模式
1	输入模式

(3) 端口 6 模式控制 (PMC6)

复位后: 00H R/W 地址: FFFFF44CH

	7	6	5	4	3	2	1	0
PMC6	0	0	PMC65	PMC64	PMC63	PMC62	PMC61	PMC60

PMC65	P65 引脚工作模式说明
0	I/O 端口
1	TOAB1B3 输出/EVTAB1 输入/ $\overline{\text{CS3}}$ 输出 ^注

PMC64	P64 引脚工作模式说明
0	I/O 端口
1	TOAB1T3 输出/TOAB13 输出/TIAB13 输入/ $\overline{\text{CS2}}$ 输出 ^注

PMC63	P63 引脚工作模式说明
0	I/O 端口
1	TOAB1B2 输出/TRGAB1 输入/ $\overline{\text{CS0}}$ 输出 ^注

PMC62	P62 引脚工作模式说明
0	I/O 端口
1	TOAB1T2 输出/TOAB12 输出/TIAB12 输出/ASTB输出 ^注

PMC61	P61 引脚工作模式说明
0	I/O 端口
1	TOAB1B1 输出/TIAB10 输入/TOAB10 输出/ $\overline{\text{RD}}$ 输出 ^注

PMC60	P60 引脚工作模式说明
0	I/O 端口
1	TOAB1T1 输出/TOAB11 输出/TIAB11 输入/ $\overline{\text{WAIT}}$ 输出 ^注

注 仅限于 V850ES/JG3-H

(4) 端口 6 功能控制寄存器 (PFC6)

复位后: 00H R/W 地址: FFFFF46CH

	7	6	5	4	3	2	1	0
PFC6	0	0	PFC65	PFC64	PFC63	PFC62	PFC61	PFC60

备注 关于复用功能规范的详情, 参见 4.3.7 (6) 端口 6 复用功能规范。

(5) 端口 6 功能控制扩展寄存器 (PFCE6)

(a) V850ES/JG3-H

复位后: 00H R/W 地址: FFFFF70CH

	7	6	5	4	3	2	1	0
PFCE6	0	0	PFCE65	PFCE64	PFCE63	PFCE62	PFCE61	PFCE60

(b) V850ES/JH3-H

复位后: 00H R/W 地址: FFFFF70CH

	7	6	5	4	3	2	1	0
PFCE6	0	0	0	0	0	0	PFCE61	0

备注 关于复用功能规范的详情, 参见 4.3.7 (6) 端口 6 复用功能规范。

(6) 端口 6 复用功能规范

PFCE65 [※]	PFC65	P65 引脚复用功能规范
0	0	TOAB1B3 输出
0	1	EVTAB1 输入
1	0	$\overline{\text{CS}}3$ 输出 [※]
1	1	禁止设置 [※]

PFCE64 [※]	PFC64	P64 引脚复用功能规范
0	0	TOAB1T3 输出/TOAB13 输出
0	1	TIAB13 输入
1	0	$\overline{\text{CS}}2$ 输出 [※]
1	1	禁止设置 [※]

PFCE63 ^注	PFC63	P63 引脚复用功能说明
0	0	TOAB1B2 输出
0	1	TRGAB1 输入
1	0	$\overline{CS0}$ 输出 ^注
1	1	禁止设置 ^注

PFCE62 ^注	PFC62	P62 引脚复用功能规范
0	0	TOAB1T2 输出/TOAB12 输出
0	1	TIAB12 输入
1	0	ASTB 输出 ^注
1	1	禁止设置 ^注

PFCE61	PFC61	P61 引脚复用功能规范
0	0	TOAB1B1 输出
0	1	TIAB10 输入
1	0	TOAB10 输出
1	1	$\overline{RD}$ 输出 (V850ES/JG3-H) 禁止设置 (V850ES/JH3-H)

PFCE60 ^注	PFC60	P60 引脚复用功能规范
0	0	TOAB1T1 输出/TOAB11 输出
0	1	TIAB11 输入
1	0	WAIT 输出 ^注
1	1	禁止设置 ^注

注 仅限于 V850ES/JG3-H

4.3.8 端口 7

端口 7 为 12 位宽度输入输出端口，其输入输出设置可以按位控制。

端口 7 包括以下复用功能引脚：

表 4-13.端口 7 复用功能引脚

引脚名称	引脚编号		复用功能引脚名称	I/O	备注
	V850ES/ JG3-H	V850ES/ JH3-H			
P70	100	128	ANI0	输入	—
P71	99	127	ANI1	输入	
P72	98	126	ANI2	输入	
P73	97	125	ANI3	输入	
P74	96	124	ANI4	输入	
P77	95	123	ANI5	输入	
P76	94	122	ANI6	输入	
P77	93	121	ANI7	输入	
P78	92	120	ANI8	输入	
P79	91	119	ANI9	输入	
P710	90	118	ANI10	输入	
P711	89	117	ANI11	输入	

(1) 端口 7 寄存器 H，端口 7 寄存器 L（P7H，P7L）

复位后: 00H (输出锁存) R/W 地址: P7L FFFFF40EH, P7H FFFFF40FH

	7	6	5	4	3	2	1	0
P7H	0	0	0	0	P711	P710	P79	P78

	7	6	5	4	3	2	1	0
P7L	P77	P76	P75	P74	P73	P72	P71	P70

P7n	输出数据控制 (输出模式下) (n = 0 至 11)
0	输出0。
1	输出1。

注意事项 A/D 转换期间不要读取或写入 P7H 和 P7L 寄存器（参见 15.6（4） 复用 I/O）。

备注 这些寄存器作为 P7 寄存器不能进行 16 位访问。它们能作为 P7H 和 P7L 寄存器以字节或位进行读写。

(2) 端口 7 模式寄存器 H，端口 7 模式寄存器 L (PM7H, PM7L)

复位后: FFH R/W 地址: PM7L FFFFF42EH, PM7H FFFFF42FH

	7	6	5	4	3	2	1	0
PM7H	1	1	1	1	PM711	PM710	PM79	PM78
	7	6	5	4	3	2	1	0
PM7L	PM77	PM76	PM75	PM74	PM73	PM72	PM71	PM70
	PM7n	I/O 模式控制(n = 0 至 11)						
	0	输出模式						
	1	输入模式						

注意事项 P7n 引脚用作其复用功能 (ANIn 引脚) 时，将 PM7n 位设置为 1。

备注 这些寄存器作为 PM7 寄存器不能进行 16 位访问。它们能作为 PM7H 和 PM7L 寄存器以字节或位读写。

4.3.9 端口 9

端口 9 为 16 位宽度输入输出端口，其输入输出设置可以按位控制。

端口 9 包括以下复用功能引脚：

表 4-14.端口 9 复用功能引脚

引脚名称	引脚编号		复用功能引脚名称	I/O	备注
	V850ES/ JG3-H	V850ES/ JH3-H			
P90	42	54	KR6/TXDC1/SDA02/A0 ^注	I/O	可选作 N 沟道漏极开路输出
P91	43	55	KR7/RXDC1/SCL02/A1 ^注	I/O	
P92	44	56	TENC01/TIT01/TOT01/A2 ^注	I/O	—
P93	45	57	TECR0/TIT00/TOT00/A3 ^注	I/O	
P94	46	58	TIAA31/TOAA31/TENC00 /EVTT0/A4 ^注	I/O	
P95	47	59	TIAA30/TOAA30/A5 ^注	I/O	
P96	48	62	TIAA21/TOAA21/INTP11/A6 ^注	I/O	
P97	49	63	SIF1/TIAA20/TOAA20/A7 ^注	I/O	
P98	50	64	SOF1/INTP12/A8 ^注	I/O	
P99	51	65	SCKF1/INTP13/A9 ^注	I/O	
P910	52	66	SIF3/TXDC2/INTP14/A10 ^注	I/O	
P911	53	67	SOF3/RXDC2/INTP15/A11 ^注	I/O	
P912	54	68	SCKF3/A12 ^注	I/O	
P913	55	69	TOAB1OFF/INTP16/A13 ^注	I/O	
P914	56	70	TIAA51/TOAA51/INTP17/A14 ^注	I/O	
P915	57	71	TIAA50/TOAA50/INTP18/A15 ^注	I/O	

注 仅限于 V850ES/JH3-H

注意事项 在复用功能引脚的输入模式下，P90 至 P915 具有滞后特性，但在端口模式下不具有滞后特性。

(1) 端口 9 寄存器 (P9)

复位后: 0000H (输出锁存) R/W 地址: P9 FFFFF412H,
P9L FFFFF412H, P9H FFFFF413H

	15	14	13	12	11	10	9	8
P9 (P9H)	P915	P914	P913	P912	P911	P910	P99	P98

	7	6	5	4	3	2	1	0
(P9L)	P97	P96	P95	P94	P93	P92	P91	P90

P9n	输出数据控制(输出模式下) (n = 0 至 15)
0	输出 0。
1	输出 1。

备注

1. P9 寄存器可以按 16 位宽度进行读写。
然而，寄存器 P9 的高 8 位和低 8 位分别用作 P9H 寄存器和 P9L 寄存器时，它们可以按字节或按位读出或写入。
2. 若要按字节或按位读取或写入 P9 的位 8 至 15，需指定这些位为 P9H 寄存器的位 0 至位 7。

(2) 端口 9 模式寄存器 (PM9)

复位后: FFFFH R/W 地址: PM9 FFFFF432H,
PM9L FFFFF432H, PM9H FFFFF433H

	15	14	13	12	11	10	9	8
PM9 (PM9H)	PM915	PM914	PM913	PM912	PM911	PM910	PM99	PM98

	7	6	5	4	3	2	1	0
(PM9L)	PM97	PM96	PM95	PM94	PM93	PM92	PM91	PM90

PM9n	I/O 模式控制 (输出模式下) (n = 0 至 15)
0	输出模式
1	输入模式

备注

1. PM9 寄存器可以按 16 位宽度进行读写。
然而，寄存器 PM9 的高 8 位和低 8 位分别用作 PM9H 寄存器和 PM9L 寄存器时，它们可以按字节和按位读出或写入。
2. 若要按字节或按位读取或写入 PM9 的位 8 至 15，需指定这些位为 PM9H 寄存器的位 0 至位 7。

复位后: 0000H R/W 地址: PMC9 FFFFF452H,
PMC9L FFFFF452H, PMC9H FFFFF453H

	15	14	13	12	11	10	9	8
PMC9 (PMC9H)	PMC915	PMC914	PMC913	PMC912	PMC911	PMC910	PMC99	PMC98
	7	6	5	4	3	2	1	0
(PMC9L)	PMC97	PMC96	PMC95	PMC94	PMC93	PMC92	PMC91	PMC90

PMC915	P915 引脚工作模式说明
0	I/O 端口
1	TIAA50 输入/TOAA50 输出/INTP18 输入/A15 输出 ^注
PMC914	P914 引脚工作模式说明
0	I/O 端口
1	TIAA51 输入/TOAA51 输出/INTP17 输入/A14 输出 ^注
PMC913	P913 引脚工作模式说明
0	I/O 端口
1	TOAB1OFF 输入/INTP16 输入/A13 输出 ^注
PMC912	P912 引脚工作模式说明
0	I/O 端口
1	SCKF3 I/O/A12 输出 ^注
PMC911	P911 引脚工作模式说明
0	I/O 端口
1	SOF3 输出/RXDC2 输入/INTP15 输入/A11 输出 ^注
PMC910	P910 引脚工作模式说明
0	I/O 端口
1	SIF3 输入/TXDC2 输出/INTP14 输入/A10 输出 ^注
PMC99	P99 引脚工作模式说明
0	I/O 端口
1	SCKF1 I/O/INTP13 输入/A9 输出 ^注
PMC98	P98 引脚工作模式说明
0	I/O 端口
1	SOF1 输出/INTP12 输入/A8 输出 ^注

注 仅限于 V850ES/JH3-H

备注 1. PMC9 寄存器可以按 16 位宽度进行读写。

然而，寄存器 PMC9 的高 8 位和低 8 位分别用作 PMC9H 寄存器和 PMC9L 寄存器时，它们可以按字节或按位读出或写入。

2. 若要按字节或按位读取或写入 PMC9 的位 8 至 15，需指定这些位为 PMC9H 寄存器的位 0 至位 7。

PMC97	P97 引脚工作模式说明
0	I/O 端口
1	SIF1 输入/TIAA20 输入/TOAA20 输出/A7 输出 ^注
PMC96	P96 引脚工作模式说明
0	I/O 端口
1	TIAA21 输入/TOAA21 输出/INTP11 输入/A6 输出 ^注
PMC95	P95 引脚工作模式说明
0	I/O 端口
1	TIAA30 输入/TOAA30 输出/A5 输出 ^注
PMC94	P94 引脚工作模式说明
0	I/O 端口
1	TIAA31 输入/TOAA31 输出/TENC00 输入/EVTT0 输入/A4 输出 ^注
PMC93	P93 引脚工作模式说明
0	I/O 端口
1	TECR0 输入/TIT00 输入/TOT00 输出/A3 输出 ^注
PMC92	P92 引脚工作模式说明
0	I/O 端口
1	TENC01 输入/TIT01 输入/TOT01 输出/A2 输出 ^注
PMC91	P91 引脚工作模式说明
0	I/O 端口
1	KR7 输入/RXDC1 输入/SCL02 I/O/A1 输出 ^注
PMC90	P90 引脚工作模式说明
0	I/O 端口
1	KR6 输入/TXDC1 输出/SDA02 I/O/A0 输出 ^注

注 仅限于 V850ES/JH3-H

注意事项 当 A0 至 A15 引脚用作 P90 至 P915 引脚的复用功能时，确保立即将 PMC9 寄存器的所有 16 位设置为 FFFFH（仅限于 V850ES/JH3-H）。

(4) 端口 9 功能控制寄存器 (PFC9)

注意事项 进行单独地址总线输出 (A0 至 A15) 时, 清除 PFC9 和 PFCE9 寄存器为 0000H 后立即将 PMC9 寄存器的所有 16 位设置为 FFFFH (仅限于 V850ES/JH3-H)。

(a) V850ES/JG3-H

复位后: 0000H R/W 地址: PFC9 FFFFF472H,
 PFC9L FFFFF472H, PFC9H FFFFF473H

	15	14	13	12	11	10	9	8
PFC9 (PFC9H)	PFC915	PFC914	0	0	PFC911	PFC910	PFC99	PFC98
	7	6	5	4	3	2	1	0
(PFC9L)	PFC97	PFC96	PFC95	PFC94	PFC93	PFC92	PFC91	PFC90

(b) V850ES/JH3-H

复位后: 0000H R/W 地址: PFC9 FFFFF472H,
 PFC9L FFFFF472H, PFC9H FFFFF473H

	15	14	13	12	11	10	9	8
PFC9 (PFC9H)	PFC915	PFC914	PFC913	PFC912	PFC911	PFC910	PFC99	PFC98
	7	6	5	4	3	2	1	0
(PFC9L)	PFC97	PFC96	PFC95	PFC94	PFC93	PFC92	PFC91	PFC90

- 备注**
1. 关于复用功能规范的详情, 参见 4.3.9 (6) 端口 9 复用功能规范。
 2. PFC9 寄存器可以按 16 位宽度进行读写。
然而, 寄存器 PFC9 的高 8 位和低 8 位分别用作 PFC9H 寄存器和 PFC9L 寄存器时, 它们可以按字节和按位读出或写入。
 3. 若要按字节或按位读取或写入 PFC9 的位 8 至 15, 需指定这些位为 PFC9H 寄存器的位 0 至位 7。

(5) 端口 9 功能控制扩展寄存器 (PFCE9)

注意事项 进行单独地址总线输出 (A0 至 A15) 时, 清除 PFC9 和 PFCE9 寄存器为 0000H 后立即将 PMC9 寄存器的所有 16 位设置为 FFFFH (仅限于 V850ES/JH3-H)。

(a) V850ES/JG3-H

复位后: 0000H R/W 地址: PFCE9 FFFFF712H,
PFCE9L FFFFF712H, PFCE9H FFFFF713H

	15	14	13	12	11	10	9	8
PFCE9 (PFCE9H)	PFCE915	PFCE914	0	0	PFCE911	PFCE910	0	0
	7	6	5	4	3	2	1	0
(PFCE9L)	PFCE97	PFCE96	0	PFCE94	PFCE93	PFCE92	PFCE91	PFCE90

(b) V850ES/JH3-H

复位后: 0000H R/W 地址: PFCE9 FFFFF712H,
PFCE9L FFFFF712H, PFCE9H FFFFF713H

	15	14	13	12	11	10	9	8
PFCE9 (PFCE9H)	PFCE915	PFCE914	0	0	PFCE911	PFCE910	PFCE99	PFCE98
	7	6	5	4	3	2	1	0
(PFCE9L)	PFCE97	PFCE96	PFCE95	PFCE94	PFCE93	PFCE92	PFCE91	PFCE90

- 备注**
1. 关于复用功能规范的详情, 参见 4.3.9 (6) 端口 9 复用功能规范。
 2. PFCE9 寄存器可以按 16 位宽度进行读写。
然而, 寄存器 PFCE9 的高 8 位和低 8 位分别用作 PFCE9H 寄存器和 PFCE9L 寄存器时, 它们可以按字节和按位读出或写入。
 3. 若要按字节或按位读取或写入 PFCE9 的位 8 至 15, 需指定这些位为 PFCE9H 寄存器的位 0 至位 7。

(6) 端口 9 复用功能规范

PFCE915	PFC915	P915 引脚复用功能说明
0	0	TIAA50 输入
0	1	TOAA50 输出
1	0	INTP18 输入
1	1	禁止设置 (V850ES/JG3-H) A15 输出 (V850ES/JH3-H)

PFCE914	PFC914	P914 引脚复用功能说明
0	0	TIAA51 输入
0	1	TOAA51 输出
1	0	INTP17 输入
1	1	禁止设置 (V850ES/JG3-H) A14 输出 (V850ES/JH3-H)

PFC913 [※]	P913 引脚复用功能说明
0	TOAB1OFF 输入/INTP16 输入
1	A13 输出 [※]

PFC912 [※]	P912 引脚复用功能说明
0	$\overline{\text{SCKF3}}$ I/O
1	A12 输出 [※]

PFCE911	PFC911	P911 引脚复用功能说明
0	0	SOF3 输出
0	1	RXDC2 输入
1	0	INTP15 输入
1	1	禁止设置 (V850ES/JG3-H) A11 输出 (V850ES/JH3-H)

PFCE910	PFC910	P910 引脚复用功能说明
0	0	SIF3 输入
0	1	TXDC2 输出
1	0	INTP14 输入
1	1	禁止设置 (V850ES/JG3-H) A10 输出 (V850ES/JH3-H)

PFCE99 [※]	PFC99	P99 引脚复用功能说明
0	0	$\overline{\text{SCKF1}}$ I/O
0	1	INTP14 输入
1	0	A9 输出 [※]
1	1	禁止设置 [※]

PFCE98 ^註	PFC98	P98 引脚复用功能说明
0	0	SOF1 输出
0	1	INTP12 输入
1	0	A8 输出 ^註
1	1	禁止设置 ^註

PFCE97	PFC97	P97 引脚复用功能说明
0	0	SIF1 输入
0	1	TIAA20 输入
1	0	TOAA20 输出
1	1	禁止设置 (V850ES/JG3-H) A7 输出 (V850ES/JH3-H)

PFCE96	PFC96	P96 引脚复用功能说明
0	0	TIAA21 输入
0	1	TOAA21 输出
1	0	INTP11 输入
1	1	禁止设置 (V850ES/JG3-H) A6 输出 (V850ES/JH3-H)

PFCE95 ^註	PFC95	P95 引脚复用功能说明
0	0	TIAA30 输入
0	1	TOAA30 输出
1	0	A5 输出 ^註
1	1	禁止设置 ^註

PFCE94	PFC94	P94 引脚复用功能说明
0	0	TIAA31 输入
0	1	TOAA31 输出
1	0	TENC00 输入/EVTT0 输入
1	1	禁止设置 (V850ES/JG3-H) A4 输出 (V850ES/JH3-H)

PFCE93	PFC93	P93 引脚复用功能说明
0	0	TECR0 输入
0	1	TIT00 输入
1	0	TOT00 输出
1	1	禁止设置 (V850ES/JG3-H) A3 输出 (V850ES/JH3-H)

PFCE92	PFC92	P92 引脚复用功能说明
0	0	TENC01 输入
0	1	TIT01 输入
1	0	TOT01 输出
1	1	禁止设置 (V850ES/JG3-H) A2 输出 (V850ES/JH3-H)

PFCE91	PFC91	P91 引脚复用功能说明
0	0	KR7 输入
0	1	RXDC1 输入
1	0	SCL02 I/O
1	1	禁止设置 (V850ES/JG3-H) A1 输出 (V850ES/JH3-H)

PFCE90	PFC90	P90 引脚复用功能说明
0	0	KR6 输入
0	1	TXDC1 输出
1	0	SDA02 I/O
1	1	禁止设置 (V850ES/JG3-H) A0 输出 (V850ES/JH3-H)

注 仅限于 V850ES/JH3-H

(7) 端口 9 功能寄存器 (PF9)

复位后: 0000H R/W 地址: PF9 FFFFC72H,
PF9L FFFFC72H

	15	14	13	12	11	10	9	8
PF9	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
(PF9L)	0	0	0	0	0	0	PF91	PF90

PF9n	正常输出或N沟道漏极开路输出控制 (n = 0, 1)
0	正常输出 (CMOS 输出)
1	N沟道漏极开路输出

注意事项 若输出引脚 P90，P91 上拉至 EVDD 或更高，确保将 PF9n 位设置为 1。

备注 PF9 寄存器可以按 16 位宽度进行读写。
然而，寄存器 PF9 的高 8 位和低 8 位分别用作 PF9H 寄存器和 PF9L 寄存器时，它们可以按字节和按位读出或写入。

4.3.10 端口 CM

端口 0 是 1 位（V850ES/JG3-H）/4 位（V850ES/JH3-H）端口，其输入/输出设置可以按位控制。
端口 CM 包括以下复用功能引脚：

表 4-15.端口 CM 复用功能引脚

引脚名称	引脚编号		复用功能引脚名称	I/O	备注
	V850ES/ JG3-H	V850ES/ JH3-H			
PCM0	—	89	WAIT	输入	—
PCM1	64	86	CLKOUT	输出	
PCM2	—	10	HLD $\overline{\text{AK}}$	输出	
PCM3	—	11	HLD $\overline{\text{RQ}}$	输入	

(1) 端口 CM 寄存器（PCM）

(a) V850ES/JG3-H

复位后: 00H (输出锁存) R/W 地址: FFFFF00CH

	7	6	5	4	3	2	1	0
PCM	0	0	0	0	0	0	PCM1	0

PCM1	输出数据控制 (输出模式下)
0	输出 0。
1	输出 1。

(b) V850ES/JH3-H

复位后: 00H (输出锁存) R/W 地址: FFFFF00CH

	7	6	5	4	3	2	1	0
PCM	0	0	0	0	PCM3	PCM2	PCM1	PCM0

PCMn	输出数据控制 (输出模式下) (n = 0 至 3)
0	输出 0。
1	输出 1。

(2) 端口 CM 模式寄存器 (PMCM)

(a) V850ES/JG3-H

复位后: FFH R/W 地址: FFFFF02CH

	7	6	5	4	3	2	1	0
PMCM	1	1	1	1	1	1	PMCM1	1

PMCM1	I/O 模式控制
0	输出模式
1	输入模式

(b) V850ES/JH3-H

复位后: FFH R/W 地址: FFFFF02CH

	7	6	5	4	3	2	1	0
PMCM	1	1	1	1	PMCM3	PMCM2	PMCM1	PMCM0

PMCMn	I/O 模式控制 (n = 0 至 3)
0	输出模式
1	输入模式

(3) 端口 CM 模式控制寄存器 (PMCCM)

(a) V850ES/JG3-H

复位后: 00H R/W 地址: FFFFF04CH

	7	6	5	4	3	2	1	0
PMCCM	0	0	0	0	0	0	PMCCM1	0

PMCCM1	PCM1 引脚工作模式说明
0	I/O 端口
1	CLKOUT 输出

(b) V850ES/JH3-H

复位后: 00H R/W 地址: FFFFF04CH

	7	6	5	4	3	2	1	0
PMCCM	0	0	0	0	PMCCM3	PMCCM2	PMCCM1	PMCCM0

PMCCM3	PCM3 引脚工作模式说明
0	I/O 端口
1	$\overline{\text{HLDRQ}}$ 输入

PMCCM2	PCM2 引脚工作模式说明
0	I/O 端口
1	$\overline{\text{HLDAK}}$ 输出

PMCCM1	PCM1 引脚工作模式说明
0	I/O 端口
1	CLKOUT 输出

PMCCM0	PCM0 引脚工作模式说明
0	I/O 端口
1	$\overline{\text{WAIT}}$ 输入

4.3.11 端口CS（仅限于V850ES/JH3-H）

端口 CS 为 3 位宽度输入输出端口，其输入输出设置可以按位控制。

端口 CS 包括以下复用功能引脚：

表 4-16.端口 CM 复用功能引脚

引脚名称	引脚编号		复用功能引脚名称	I/O	备注
	V850ES/ JG3-H	V850ES/ JH3-H			
PCS0	—	96	$\overline{\text{CS0}}$	输出	—
PCS2	—	97	$\overline{\text{CS2}}$	输出	
PCS3	—	116	$\overline{\text{CS3}}$	输出	

(1) 端口 CS 寄存器（PCS）

复位后: 00H (输出锁存) R/W 地址: FFFFF008H

	7	6	5	4	3	2	1	0
PCS	0	0	0	0	PCS3	PCS2	0	PCS0

PCS _n	输出数据控制 (输出模式) (n = 0, 2, 3)
0	输出 0。
1	输出 1。

(2) 端口 CS 模式寄存器（PMCS）

复位后: FFH R/W 地址: FFFFF028H

	7	6	5	4	3	2	1	0
PMCS	1	1	1	1	PMCS3	PMCS2	1	PMCS0

PMCS _n	I/O 模式控制 (n = 0, 2, 3)
0	输出模式
1	输入模式

(3) 端口 CS 模式控制寄存器 (PMCCS)

复位后: 00H R/W 地址: FFFFF048H

	7	6	5	4	3	2	1	0
PMCCS	0	0	0	0	PMCCS3	PMCCS2	0	PMCCS0

PMCCS3	PCS3 引脚工作模式说明
0	I/O 端口
1	$\overline{\text{CS3}}$ 输出

PMCCS2	PCS2 引脚工作模式说明
0	I/O 端口
1	$\overline{\text{CS2}}$ 输出

PMCCS0	PCS0 引脚工作模式说明
0	I/O 端口
1	$\overline{\text{CS0}}$ 输出

4.3.12 端口CT

端口 0 是 2 位 (V850ES/JG3-H) /4 位 (V850ES/JH3-H) 端口, 其输入/输出设置可以按位控制。

端口 CT 包括以下复用功能引脚:

表 4-17. 端口 CT 复用功能引脚

引脚名称	引脚编号		复用功能引脚名称	I/O	备注
	V850ES/ JG3-H	V850ES/ JH3-H			
PCT0	58	76	$\overline{\text{WR0}}$	输出	—
PCT1	59	77	$\overline{\text{WR1}}$	输出	
PCT4	—	87	$\overline{\text{RD}}$	输出	
PCT6	—	88	ASTB	输出	

(1) 端口 CT 寄存器 (PCT)

(a) V850ES/JG3-H

复位后: 00H (输出锁存) R/W 地址: FFFFF00AH

	7	6	5	4	3	2	1	0
PCT	0	0	0	0	0	0	PCT1	PCT0

PCTn	输出数据控制 (输出模式下) (n = 0, 1)
0	输出 0。
1	输出 1。

(b) V850ES/JH3-H

复位后: 00H (输出锁存) R/W 地址: FFFFF00AH

	7	6	5	4	3	2	1	0
PCT	0	PCT6	0	PCT4	0	0	PCT1	PCT0

PCMn	输出数据控制 (输出模式) (n = 0, 1, 4, 6)
0	输出 0。
1	输出 1。

(2) 端口 CT 模式寄存器 (PMCT)

(a) V850ES/JG3-H

复位后: FFH R/W 地址: FFFFF02AH

	7	6	5	4	3	2	1	0
PMCT	1	1	1	1	1	1	PMCT1	PMCT0

PMCTn	I/O 模式控制 (n = 0, 1)
0	输出模式
1	输入模式

(b) V850ES/JH3-H

复位后: FFH R/W 地址: FFFFF02AH

	7	6	5	4	3	2	1	0
PMCT	1	PMCT6	1	PMCT4	1	1	PMCT1	PMCT0

PMCTn	I/O 模式控制 (n = 0, 1, 4, 6)
0	输出模式
1	输入模式

(3) 端口 CT 模式控制寄存器 (PMCCT)

(a) V850ES/JG3-H

复位后: 00H R/W 地址: FFFFF04AH

	7	6	5	4	3	2	1	0
PMCCT	0	0	0	0	0	0	PMCCT1	PMCCT0

PMCCT1	PCT1 引脚工作模式说明	
0	I/O 端口	
1	$\overline{\text{WR1}}$ 输出	
PMCCT0	PCT0 引脚工作模式说明	
0	I/O 端口	
1	$\overline{\text{WR0}}$ 输出	

(b) V850ES/JH3-H

复位后: 00H R/W 地址: FFFFF04AH

	7	6	5	4	3	2	1	0
PMCCT	0	PMCCT6	0	PMCCT4	0	0	PMCCT1	PMCCT0

PMCCT6	PCT6 引脚工作模式说明	
0	I/O 端口	
1	ASTB 输出	
PMCCT4	PCT4 引脚工作模式说明	
0	I/O 端口	
1	$\overline{\text{RD}}$ 输出	
PMCCT1	PCT1 引脚工作模式说明	
0	I/O 端口	
1	$\overline{\text{WR1}}$ 输出	
PMCCT0	PCT0 引脚工作模式说明	
0	I/O 端口	
1	$\overline{\text{WR0}}$ 输出	

4.3.13 端口DH（仅限于V850ES/JH3-H）

端口 DH 为 8 位宽度输入输出端口，其输入输出设置可以按位控制。
端口 DH 包括以下复用功能引脚：

表 4-18.端口 DH 复用功能引脚

引脚名称	引脚编号		复用功能引脚名称	I/O	备注
	V850ES/ JG3-H	V850ES/ JH3-H			
PDH0	—	72	A16	输出	—
PDH1	—	73	A17	输出	
PDH2	—	74	A18	输出	
PDH3	—	75	A19	输出	
PDH4	—	78	A20	输出	
PDH5	—	79	A21	输出	
PDH6	—	80	A22	输出	
PDH7	—	81	A23	输出	

(1) 端口 DH 寄存器 (PDH)

复位后: 00H (输出锁存) R/W 地址: FFFFF006H

	7	6	5	4	3	2	1	0
PDH	PDH7	PDH6	PDH5	PDH4	PDH3	PDH2	PDH1	PDH0

PDHn	输出数据控制 (输出模式下) (n = 0 至7)
0	输出 0。
1	输出 1。

(2) 端口 DH 模式寄存器 (PMDH)

复位后: FFH R/W 地址: FFFFF026H

	7	6	5	4	3	2	1	0
PMDH	PMDH7	PMDH6	PMDH5	PMDH4	PMDH3	PMDH2	PMDH1	PMDH0

PMDHn	I/O 模式控制 (n = 0, 7)
0	输出模式
1	输入模式

(3) 端口 DH 模式控制寄存器 (PMCDH)

复位后: 00H R/W 地址: FFFFF046H

	7	6	5	4	3	2	1	0
PMCDH	PMCDH7	PMCDH6	PMCDH5	PMCDH4	PMCDH3	PMCDH2	PMCDH1	PMCDH0

PMCDHn	PDHn 引脚工作模式说明 (n = 0 至 7)
0	I/O 端口
1	Am 输出 (地址总线) (m = 16 至 23)

4.3.14 端口 DL

端口 DL 为 16 位宽度输入输出端口，其输入输出设置可以按位控制。
端口 DL 包括以下复用功能引脚：

表 4-19.端口 DL 复用功能引脚

引脚名称	引脚编号		复用功能引脚名称	I/O	备注
	V850ES/ JG3-H	V850ES/ JH3-H			
PDL0	71	98	AD0	I/O	—
PDL1	72	99	AD1	I/O	
PDL2	73	100	AD2	I/O	
PDL3	74	101	AD3	I/O	
PDL4	75	102	AD4	I/O	
PDL5	78	103	AD5/FLMD1 ^注	I/O	
PDL6	79	104	AD6	I/O	
PDL7	80	105	AD7	I/O	
PDL8	81	108	AD8	I/O	
PDL9	82	109	AD9	I/O	
PDL10	83	110	AD10	I/O	
PDL11	84	111	AD11	I/O	
PDL12	85	112	AD12	I/O	
PDL13	86	113	AD13	I/O	
PDL14	87	114	AD14	I/O	
PDL15	88	115	AD15	I/O	

注 该引脚设置为 flash 存储器编程模式后，其不需要通过端口控制寄存器进行操作。细节参见第三十一章 flash 存储器。

(1) 端口 DL 寄存器 (PDL)

复位后: 0000H (输出锁存) R/W 地址: PDL FFFF004H,
PDL FFFF004H, PDLH FFFF005H

	15	14	13	12	11	10	9	8
PDL (PDLH)	PDL15	PDL14	PDL13	PDL12	PDL11	PDL10	PDL9	PDL8

	7	6	5	4	3	2	1	0
(PDLL)	PDL7	PDL6	PDL5	PDL4	PDL3	PDL2	PDL1	PDL0

PDLn	输出数据控制 (输出模式下) (n = 0 至 15)
0	输出 0。
1	输出 1。

- 备注**
1. PDL 寄存器可以按 16 位宽度进行读写。
然而，寄存器 PDL 的高 8 位和低 8 位分别用作 PDLH 寄存器和 PDLL 寄存器时，它们可以按字节或按位读出或写入。
 2. 若要按字节或按位读取或写入 PDL 的位 8 至 15，需指定这些位为 PDLH 寄存器的位 0 至位 7。

(2) 端口 DL 模式寄存器 (PMDL)

复位后: FFFFH R/W 地址: PMDL FFFF024H,
PMDLL FFFF024H, PMDLH FFFF025H

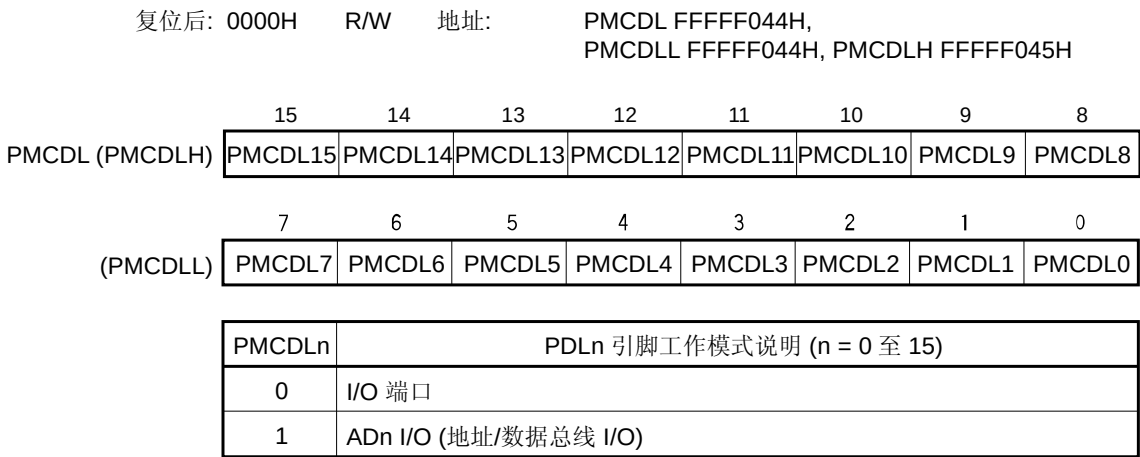
	15	14	13	12	11	10	9	8
PMDL (PMDLH)	PMDL15	PMDL14	PMDL13	PMDL12	PMDL11	PMDL10	PMDL9	PMDL8

	7	6	5	4	3	2	1	0
(PMDLL)	PMDL7	PMDL6	PMDL5	PMDL4	PMDL3	PMDL2	PMDL1	PMDL0

PMDLn	I/O 模式控制 (n = 0 至 15)
0	输出模式
1	输入模式

- 备注**
1. PMDL 寄存器可以按 16 位宽度进行读写。
然而，寄存器 PMDL 的高 8 位和低 8 位分别用作 PMDLH 寄存器和 PMDLL 寄存器时，它们可以按字节或按位读出或写入。
 2. 若要按字节或按位读取或写入 PMDL 的位 8 至 15，需指定这些位为 PMDLH 寄存器的位 0 至位 7。

(3) 端口 DL 模式控制寄存器 (PMCDL)



备注

- 1. PMCDL 寄存器可以按 16 位宽度进行读写。
然而，寄存器 PMCDL 的高 8 位和低 8 位分别用作 PMCDLH 寄存器和 PMCDLL 寄存器时，它们可以按字节或按位读出或写入。
- 2. 若要按字节或按位读取或写入 PMCDL 的位 8 至 15，需指定这些位为 PMCDLH 寄存器的位 0 至位 7。

4.4 使用复用功能时的端口寄存器设置

表 4-20 显示各个端口用作每一种复用功能时的端口寄存器设置。当某一端口引脚用作复用功能引脚时，请参考各引脚的描述。

表 4-20. 端口引脚用作复用功能引脚（1/10）

引脚名称	复用功能		Pn 寄存器的 Pnx 位	PMn 寄存器的 PMnx 位	PMCn 寄存器的 PMCnx 位	PFCEn 寄存器的 PFCEnx 位	PFCn 寄存器的 PFCnx 位	其它位 (寄存器)
	名称	I/O						
P00 ^{註 1}	INTP00	输入	P00 = 不需要设置	PM00 = 不需要设置	PMC00 = 1	–	–	
P01 ^{註 1}	INTP01	输入	P01 = 不需要设置	PM01 = 不需要设置	PMC01 = 1	–	–	
P02	NMI	输入	P02 = 不需要设置	PM02 = 不需要设置	PMC02 = 1	–	–	
P03	INTP02	输入	P03 = 不需要设置	PM03 = 不需要设置	PMC03 = 1	PFCE03 = 0	PFC03 = 0	
	ADTRG	输入	P03 = 不需要设置	PM03 = 不需要设置	PMC03 = 1	PFCE03 = 0	PFC03 = 1	
	UCLK	输入	P03 = 不需要设置	PM03 = 不需要设置	PMC03 = 1	PFCE03 = 1	PFC03 = 0	
P04	INTP03	输入	P04 = 不需要设置	PM04 = 不需要设置	PMC04 = 1	–	–	
P05	INTP04	输入	P05 = 不需要设置	PM05 = 不需要设置	PMC05 = 1	–	–	
P10	ANO0	输出	P10 = 不需要设置	PM10 = 1	–	–	–	
P11	ANO1	输出	P11 = 不需要设置	PM11 = 1	–	–	–	
P20 ^{註 1}	TIAB03	输入	P20 = 不需要设置	PM20 = 不需要设置	PMC20 = 1	PFCE20 = 0	PFC20 = 0	
	KR2	输入	P20 = 不需要设置	PM20 = 不需要设置	PMC20 = 1	PFCE20 = 0	PFC20 = 1	
	TOAB03	输出	P20 = 不需要设置	PM20 = 不需要设置	PMC20 = 1	PFCE20 = 1	PFC20 = 0	
	RTP02	输出	P20 = 不需要设置	PM20 = 不需要设置	PMC20 = 1	PFCE20 = 1	PFC20 = 1	
P21 ^{註 1}	SIF2	输入	P21 = 不需要设置	PM21 = 不需要设置	PMC21 = 1	PFCE21 = 0	PFC21 = 0	
	KR3 ^{註 2}	输入	P21 = 不需要设置	PM21 = 不需要设置	PMC21 = 1	PFCE21 = 0	PFC21 = 1	
	TIAB00 ^{註 2}	输入	P21 = 不需要设置	PM21 = 不需要设置	PMC21 = 1	PFCE21 = 0	PFC21 = 1	
	TOAB00	输出	P21 = 不需要设置	PM21 = 不需要设置	PMC21 = 1	PFCE21 = 1	PFC21 = 0	
	RTP03	输出	P21 = 不需要设置	PM21 = 不需要设置	PMC21 = 1	PFCE21 = 1	PFC21 = 1	

注 1. 仅限于 V850ES/JH3-H

2. KR3 引脚和 TIAB00 引脚为复用功能引脚。将该引脚用作 TIAB00 引脚时，禁止使用复用功能 KR3 引脚。同样，将该引脚用作 KR3 引脚时，禁止使用 TIAB00 引脚。

注意事项 电源开启时，即使在复位期间 P10 和 P11 引脚也可能暂时输出不确定电平。

表 4-20. 端口引脚用作复用功能引脚（2/10）

引脚名称	复用功能		Pn 寄存器的 Pnx 位	PMn 寄存器的 PMnx 位	PMCn 寄存器的 PMCnx 位	PFCEn 寄存器的 PFCEnx 位	PFCn 寄存器的 PFCnx 位	其它位 (寄存器)
	名称	I/O						
P22 ^注	SOF2	输出	P22 = 不需要设置	PM22 = 不需要设置	PMC22= 1	PFCE22 = 0	PFC22 = 0	
	KR4	输入	P22 = 不需要设置	PM22 = 不需要设置	PMC22= 1	PFCE22 = 0	PFC22 = 1	
	RTP04	输出	P22 = 不需要设置	PM22 = 不需要设置	PMC22= 1	PFCE22 = 1	PFC22 = 0	
P23 ^注	$\overline{\text{SCKF2}}$	I/O	P23 = 不需要设置	PM23 = 不需要设置	PMC23= 1	PFCE23 = 0	PFC23 = 0	
	KR5	输入	P23 = 不需要设置	PM23 = 不需要设置	PMC23= 1	PFCE23 = 0	PFC23 = 1	
	RTP05	输出	P23 = 不需要设置	PM23 = 不需要设置	PMC23= 1	PFCE23 = 1	PFC23 = 0	
P24 ^注	INTP05	输入	P24 = 不需要设置	PM24 = 不需要设置	PMC24= 1	–	–	
P25 ^注	INTP06	输入	P25 = 不需要设置	PM25 = 不需要设置	PMC25= 1	–	–	
P30	TXDC0	输出	P30 = 不需要设置	PM30 = 不需要设置	PMC30 = 1	PFCE30 = 0	PFC30 = 0	
	SOF4	输出	P30 = 不需要设置	PM30 = 不需要设置	PMC30 = 1	PFCE30 = 0	PFC30 = 1	
	INTP07	输入	P30 = 不需要设置	PM30 = 不需要设置	PMC30 = 1	PFCE30 = 1	PFC30 = 0	
P31	RXDC0	输入	P31 = 不需要设置	PM31 = 不需要设置	PMC31 = 1	PFCE31 = 0	PFC31 = 0	
	SIF4	输入	P31 = 不需要设置	PM31 = 不需要设置	PMC31 = 1	PFCE31 = 0	PFC31 = 1	
	INTP08	输入	P31 = 不需要设置	PM31 = 不需要设置	PMC31 = 1	PFCE31 = 1	PFC31 = 0	
P32	ASCKC0	输入	P32 = 不需要设置	PM32 = 不需要设置	PMC32 = 1	PFCE32 = 0	PFC32 = 0	
	$\overline{\text{SCKF4}}$	I/O	P32 = 不需要设置	PM32 = 不需要设置	PMC32 = 1	PFCE32 = 0	PFC32 = 1	
	TIAA00	输入	P32 = 不需要设置	PM32 = 不需要设置	PMC32 = 1	PFCE32 = 1	PFC32 = 0	
	TOAA00	输出	P32 = 不需要设置	PM32 = 不需要设置	PMC32 = 1	PFCE32 = 1	PFC32 = 1	
P33	TIAA01	输入	P33 = 不需要设置	PM33 = 不需要设置	PMC33 = 1	PFCE33 = 0	PFC33 = 0	
	TOAA01	输出	P33 = 不需要设置	PM33 = 不需要设置	PMC33 = 1	PFCE33 = 0	PFC33 = 1	
	RTCDIV	输出	P33 = 不需要设置	PM33 = 不需要设置	PMC33 = 1	PFCE33 = 1	PFC33 = 0	
	RTCCL	输出	P33 = 不需要设置	PM33 = 不需要设置	PMC33 = 1	PFCE33 = 1	PFC33 = 1	

注 仅限于 V850ES/JH3-H

表 4-20. 端口引脚用作复用功能引脚（3/10）

引脚名称	复用功能		Pn 寄存器的 Pnx 位	PMn 寄存器的 PMnx 位	PMCn 寄存器的 PMCnx 位	PFCEn 寄存器的 PFCEnx 位	PFCn 寄存器的 PFCnx 位	其它位 (寄存器)
	名称	I/O						
P34	TIAA10	输入	P34 = 不需要设置	PM34 = 不需要设置	PMC34 = 1	PFCE34 = 0	PFC34 = 0	
	TOAA10	输出	P34 = 不需要设置	PM34 = 不需要设置	PMC34 = 1	PFCE34 = 0	PFC34 = 1	
	TOAA10FF ^{#1}	输入	P34 = 不需要设置	PM34 = 不需要设置	PMC34 = 1	PFCE34 = 1	PFC34 = 0	
	INTP09 ^{#1}	输入	P34 = 不需要设置	PM34 = 不需要设置	PMC34 = 1	PFCE34 = 1	PFC34 = 0	
P35	TIAA11	输入	P35 = 不需要设置	PM35 = 不需要设置	PMC35 = 1	PFCE35 = 0	PFC35 = 0	
	TOAA11	输出	P35 = 不需要设置	PM35 = 不需要设置	PMC35 = 1	PFCE35 = 0	PFC35 = 1	
	RTC1HZ	输出	P35 = 不需要设置	PM35 = 不需要设置	PMC35 = 1	PFCE35 = 1	PFC35 = 0	
P36	TXDC3	输出	P36 = 不需要设置	PM36 = 不需要设置	PMC36 = 1	PFCE36 = 0	PFC36 = 0	
	SCL00	I/O	P36 = 不需要设置	PM36 = 不需要设置	PMC36 = 1	PFCE36 = 0	PFC36 = 1	PF36 (PF3) = 1
	CTXD0 ^{#2}	输出	P36 = 不需要设置	PM36 = 不需要设置	PMC36 = 1	PFCE36 = 1	PFC36 = 0	
	UDMARQ0	输入	P36 = 不需要设置	PM36 = 不需要设置	PMC36 = 1	PFCE36 = 1	PFC36 = 1	
P37	RXDC3	输入	P37 = 不需要设置	PM37 = 不需要设置	PMC37 = 1	PFCE37 = 0	PFC37 = 0	
	SDA00	I/O	P37 = 不需要设置	PM37 = 不需要设置	PMC37 = 1	PFCE37 = 0	PFC37 = 1	PF37 (PF3) = 1
	CRXD0 ^{#2}	输入	P37 = 不需要设置	PM37 = 不需要设置	PMC37 = 1	PFCE37 = 1	PFC37 = 0	
	UDMAAK0	输出	P37 = 不需要设置	PM37 = 不需要设置	PMC37 = 1	PFCE37 = 1	PFC37 = 1	
P40	SIF0	输入	P40 = 不需要设置	PM40 = 不需要设置	PMC40 = 1	PFCE40 = 0	PFC40 = 0	
	TXDC4	输出	P40 = 不需要设置	PM40 = 不需要设置	PMC40 = 1	PFCE40 = 0	PFC40 = 1	
	SDA01	I/O	P40 = 不需要设置	PM40 = 不需要设置	PMC40 = 1	PFCE40 = 1	PFC40 = 0	PF40 (PF4) = 0
P41	SOF0	输出	P41 = 不需要设置	PM41 = 不需要设置	PMC41 = 1	PFCE41 = 0	PFC41 = 0	
	RXDC4	输入	P41 = 不需要设置	PM41 = 不需要设置	PMC41 = 1	PFCE41 = 0	PFC41 = 1	
	SCL01	I/O	P41 = 不需要设置	PM41 = 不需要设置	PMC41 = 1	PFCE41 = 1	PFC41 = 0	PF41 (PF4) = 0
P42	SCKF0	I/O	P42 = 不需要设置	PM42 = 不需要设置	PMC42 = 1	–	PFC42 = 0	
	INTP10	输入	P42 = 不需要设置	PM42 = 不需要设置	PMC42 = 1	–	PFC42 = 1	

- 注
1. TOAA10FF 引脚和 INTP09 引脚为复用功能引脚。将该引脚用作 TOAA10FF 引脚时，禁止 INTP09 引脚的边沿检测复用功能（设置为 INTF3.INTF34 = 0，INTR3.INTR34 = 0）。同样，当用作 INTP09 引脚时，确保停用高阻输出电路。
 2. 仅限于 μ PD70F3770, 70F3771

表 4-20. 端口引脚用作复用功能引脚（4/10）

引脚名称	复用功能		Pn 寄存器的 Pnx 位	PMn 寄存器的 PMnx 位	PMCn 寄存器的 PMCnx 位	PFCEn 寄存器的 PFCEnx 位	PFCn 寄存器的 PFCnx 位	其它位 (寄存器)
	名称	I/O						
P50	TIAB01	输入	P50 = 不需要设置	PM50 = 不需要设置	PMC50 = 1	PFCE50 = 0	PFC50 = 0	KRM0 (KRM)= 0
	KR0	输入	P50 = 不需要设置	PM50 = 不需要设置	PMC50 = 1	PFCE50 = 0	PFC50 = 0	TAB0TIG2, TAB0TIG3 (TAB0IOC1) = 0
	TOAB01	输出	P50 = 不需要设置	PM50 = 不需要设置	PMC50 = 1	PFCE50 = 0	PFC50 = 1	
	RTP00	输出	P50 = 不需要设置	PM50 = 不需要设置	PMC50 = 1	PFCE50 = 1	PFC50 = 0	
	UDMARQ1	输入	P50 = 不需要设置	PM50 = 不需要设置	PMC50 = 1	PFCE50 = 1	PFC50 = 1	
P51	TIAB02	输入	P51 = 不需要设置	PM51 = 不需要设置	PMC51 = 1	PFCE51 = 0	PFC51 = 0	KRM1 (KRM) = 0
	KR1	输入	P51 = 不需要设置	PM51 = 不需要设置	PMC51 = 1	PFCE51 = 0	PFC51 = 0	TAB0TIG4, TAB0TIG5 (TAB0IOC1) = 0
	TOAB02	输出	P51 = 不需要设置	PM51 = 不需要设置	PMC51 = 1	PFCE51 = 0	PFC51 = 1	
	RTP01	输出	P51 = 不需要设置	PM51 = 不需要设置	PMC51 = 1	PFCE51 = 1	PFC51 = 0	
	UDMAAK1	输出	P51 = 不需要设置	PM51 = 不需要设置	PMC51 = 1	PFCE51 = 1	PFC51 = 1	
P52 ^注	TIAB03	输入	P52 = 不需要设置	PM52 = 不需要设置	PMC52 = 1	PFCE52 = 0	PFC52 = 0	KRM2 (KRM) = 0
	KR2	输入	P52 = 不需要设置	PM52 = 不需要设置	PMC52 = 1	PFCE52 = 0	PFC52 = 0	TAB0TIG6, TAB0TIG7 (TAB0IOC1) = 0
	TOAB03	输出	P52 = 不需要设置	PM52 = 不需要设置	PMC52 = 1	PFCE52 = 0	PFC52 = 1	
	RTP02	输出	P52 = 不需要设置	PM52 = 不需要设置	PMC52 = 1	PFCE52 = 1	PFC52 = 0	
P53 ^注	SIF2	输入	P53 = 不需要设置	PM53 = 不需要设置	PMC53 = 1	PFCE53 = 0	PFC53 = 0	
	TIAB00	输入	P53 = 不需要设置	PM53 = 不需要设置	PMC53 = 1	PFCE53 = 0	PFC53 = 1	KRM3 (KRM) = 0
	KR3	输入	P53 = 不需要设置	PM53 = 不需要设置	PMC53 = 1	PFCE53 = 0	PFC53 = 1	TAB0TIG0, TAB0TIG1 (TAB0IOC1) = 0, TAB0EES0, TAB0EES1 (TAB0IOC2) = 0, TAB0ETS0, TAB0ETS1 (TAB0IOC2) = 0
	TOAB00	输出	P53 = 不需要设置	PM53 = 不需要设置	PMC53 = 1	PFCE53 = 1	PFC53 = 0	
	RTP03	输出	P53 = 不需要设置	PM53 = 不需要设置	PMC53 = 1	PFCE53 = 1	PFC53 = 1	
P54 ^注	SOF2	输出	P54 = 不需要设置	PM54 = 不需要设置	PMC54 = 1	PFCE54 = 0	PFC54 = 0	
	KR4	输入	P54 = 不需要设置	PM54 = 不需要设置	PMC54 = 1	PFCE54 = 0	PFC54 = 1	
	RTP04	输出	P54 = 不需要设置	PM54 = 不需要设置	PMC54 = 1	PFCE54 = 1	PFC54 = 0	

注 仅限于 V850ES/JG3-H

表 4-20. 端口引脚用作复用功能引脚（5/10）

引脚名称	复用功能		Pn 寄存器的 Pnx 位	PMn 寄存器的 PMnx 位	PMCn 寄存器的 PMCnx 位	PFCEn 寄存器的 PFCEnx 位	PFCn 寄存器的 PFCnx 位	其它位 (寄存器)
	名称	I/O						
P55 [※]	SCKF2	I/O	P55 = 不需要设置	PM55 = 不需要设置	PMC55 = 1	PFCE55 = 0	PFC55 = 0	
	KR5	输入	P55 = 不需要设置	PM55 = 不需要设置	PMC55 = 1	PFCE55 = 0	PFC55 = 1	
	RTP05	输出	P55 = 不需要设置	PM55 = 不需要设置	PMC55 = 1	PFCE55 = 1	PFC55 = 0	
P60	TOAB1T1	输出	P60 = 不需要设置	PM60 = 不需要设置	PMC60 = 1	PFCE60 = 0 [※]	PFC60 = 0	
	TOAB11	输出	P60 = 不需要设置	PM60 = 不需要设置	PMC60 = 1	PFCE60 = 0 [※]	PFC60 = 0	
	TIAB11	输入	P60 = 不需要设置	PM60 = 不需要设置	PMC60 = 1	PFCE60 = 0 [※]	PFC60 = 1	
	WAIT [※]	输出	P60 = 不需要设置	PM60 = 不需要设置	PMC60 = 1	PFCE60 = 1 [※]	PFC60 = 0	
P61	TOAB1B1	输出	P61 = 不需要设置	PM61 = 不需要设置	PMC61 = 1	PFCE61 = 0	PFC61 = 0	
	TIAB10	输入	P61 = 不需要设置	PM61 = 不需要设置	PMC61 = 1	PFCE61 = 0	PFC61 = 1	
	TOAB10	输出	P61 = 不需要设置	PM61 = 不需要设置	PMC61 = 1	PFCE61 = 1	PFC61 = 0	
	RD [※]	输出	P61 = 不需要设置	PM61 = 不需要设置	PMC61 = 1	PFCE61 = 1	PFC61 = 1	
P62	TOAB1T2	输出	P62 = 不需要设置	PM62 = 不需要设置	PMC62 = 1	PFCE62 = 0 [※]	PFC62 = 0	
	TOAB12	输出	P62 = 不需要设置	PM62 = 不需要设置	PMC62 = 1	PFCE62 = 0 [※]	PFC62 = 0	
	TIAB12	输入	P62 = 不需要设置	PM62 = 不需要设置	PMC62 = 1	PFCE62 = 0 [※]	PFC62 = 1	
	ASTB [※]	输出	P62 = 不需要设置	PM62 = 不需要设置	PMC62 = 1	PFCE62 = 1 [※]	PFC62 = 0	
P63	TOAB1B2	输出	P63 = 不需要设置	PM63 = 不需要设置	PMC63 = 1	PFCE63 = 0 [※]	PFC63 = 0	
	TRGAB1	输入	P63 = 不需要设置	PM63 = 不需要设置	PMC63 = 1	PFCE63 = 0 [※]	PFC63 = 1	
	CS0 [※]	输出	P63 = 不需要设置	PM63 = 不需要设置	PMC63 = 1	PFCE63 = 1 [※]	PFC63 = 0	
P64	TOAB1T3	输出	P64 = 不需要设置	PM64 = 不需要设置	PMC64 = 1	PFCE64 = 0 [※]	PFC64 = 0	
	TOAB13	输出	P64 = 不需要设置	PM64 = 不需要设置	PMC64 = 1	PFCE64 = 0 [※]	PFC64 = 0	
	TIAB13	输入	P64 = 不需要设置	PM64 = 不需要设置	PMC64 = 1	PFCE64 = 0 [※]	PFC64 = 1	
	CS2 [※]	输出	P64 = 不需要设置	PM64 = 不需要设置	PMC64 = 1	PFCE64 = 1 [※]	PFC64 = 0	
P65	TOAB1B3	输出	P65 = 不需要设置	PM65 = 不需要设置	PMC65 = 1	PFCE65 = 0 [※]	PFC63 = 0	
	EVTAB1	输入	P65 = 不需要设置	PM65 = 不需要设置	PMC65 = 1	PFCE65 = 0 [※]	PFC65 = 1	
	CS3 [※]	输出	P65 = 不需要设置	PM65 = 不需要设置	PMC65 = 1	PFCE65 = 1 [※]	PFC65 = 0	

注 仅限于 V850ES/JG3-H

表 4-20. 端口引脚用作复用功能引脚（6/10）

引脚名称	复用功能		Pn 寄存器的 Pnx 位	PMn 寄存器的 PMnx 位	PMCn 寄存器的 PMCnx 位	PFCEn 寄存器的 PFCEnx 位	PFCn 寄存器的 PFCnx 位	其它位 (寄存器)
	名称	I/O						
P70	ANI0	输入	P70 = 不需要设置	PM70 = 1	–	–	–	
P71	ANI1	输入	P71 = 不需要设置	PM71 = 1	–	–	–	
P72	ANI2	输入	P72 = 不需要设置	PM72 = 1	–	–	–	
P73	ANI3	输入	P73 = 不需要设置	PM73 = 1	–	–	–	
P74	ANI4	输入	P74 = 不需要设置	PM74 = 1	–	–	–	
P75	ANI5	输入	P75 = 不需要设置	PM75 = 1	–	–	–	
P76	ANI6	输入	P76 = 不需要设置	PM76 = 1	–	–	–	
P77	ANI7	输入	P77 = 不需要设置	PM77 = 1	–	–	–	
P78	ANI8	输入	P78 = 不需要设置	PM78 = 1	–	–	–	
P79	ANI9	输入	P79 = 不需要设置	PM79 = 1	–	–	–	
P710	ANI10	输入	P710 = 不需要设置	PM710 = 1	–	–	–	
P711	ANI11	输入	P711 = 不需要设置	PM711 = 1	–	–	–	
P90	KR6	输入	P90 = 不需要设置	PM90 = 不需要设置	PMC90 = 1	PFCE90 = 0	PFC90 = 0	
	TXDC1	输出	P90 = 不需要设置	PM90 = 不需要设置	PMC90 = 1	PFCE90 = 0	PFC90 = 1	
	SDA02	I/O	P90 = 不需要设置	PM90 = 不需要设置	PMC90 = 1	PFCE90 = 1	PFC90 = 0	PF90 (PF9) = 1
	A0 ^{注 1}	输出	P90 = 不需要设置	PM90 = 不需要设置	PMC90 = 1	PFCE90 = 1	PFC90 = 1	注 2
P91	KR7	输入	P91 = 不需要设置	PM91 = 不需要设置	PMC91 = 1	PFCE91 = 0	PFC91 = 0	
	RXDC1	输入	P91 = 不需要设置	PM91 = 不需要设置	PMC91 = 1	PFCE91 = 0	PFC91 = 1	
	SCL02	I/O	P91 = 不需要设置	PM91 = 不需要设置	PMC91 = 1	PFCE91 = 1	PFC91 = 0	PF91 (PF9) = 1
	A1 ^{注 1}	输出	P91 = 不需要设置	PM91 = 不需要设置	PMC91 = 1	PFCE91 = 1	PFC91 = 1	注 2
P92	TENC01	输入	P92 = 不需要设置	PM92 = 不需要设置	PMC92 = 1	PFCE92 = 0	PFC92 = 0	
	TIT01	输入	P92 = 不需要设置	PM92 = 不需要设置	PMC92 = 1	PFCE92 = 0	PFC92 = 1	
	TOT01	输出	P92 = 不需要设置	PM92 = 不需要设置	PMC92 = 1	PFCE92 = 1	PFC92 = 0	
	A2 ^{注 1}	输出	P92 = 不需要设置	PM92 = 不需要设置	PMC92 = 1	PFCE92 = 1	PFC92 = 1	注 2

注 1. 仅限于 V850ES/JH3-H

2. 将 A0 至 A15 引脚用作复用功能时，立即将 PMC9 寄存器的所有 16 位设置为 FFFFH。

表 4-20. 端口引脚用作复用功能引脚（7/10）

引脚名称	复用功能		Pn 寄存器的 Pnx 位	PMn 寄存器的 PMnx 位	PMCn 寄存器的 PMCnx 位	PFCEn 寄存器的 PFCEnx 位	PFCn 寄存器的 PFCnx 位	其它位 (寄存器)
	名称	I/O						
P93	TECR0	输入	P93 = 不需要设置	PM93 = 不需要设置	PMC93 = 1	PFCE93 = 0	PFC93 = 0	
	TIT00	输入	P93 = 不需要设置	PM93 = 不需要设置	PMC93 = 1	PFCE93 = 0	PFC93 = 1	
	TOT00	输出	P93 = 不需要设置	PM93 = 不需要设置	PMC93 = 1	PFCE93 = 1	PFC93 = 0	
	A3 ^{注 1}	输出	P93 = 不需要设置	PM93 = 不需要设置	PMC93 = 1	PFCE93 = 1	PFC93 = 0	注 2
P94	TIAA31	输入	P94 = 不需要设置	PM94 = 不需要设置	PMC94 = 1	PFCE94 = 0	PFC94 = 0	
	TOAA31	输出	P94 = 不需要设置	PM94 = 不需要设置	PMC94 = 1	PFCE94 = 0	PFC94 = 1	
	TENC00	输入	P94 = 不需要设置	PM94 = 不需要设置	PMC94 = 1	PFCE94 = 1	PFC94 = 0	
	EVTT0	输入	P94 = 不需要设置	PM94 = 不需要设置	PMC94 = 1	PFCE94 = 1	PFC94 = 0	
	A4 ^{注 1}	输出	P94 = 不需要设置	PM94 = 不需要设置	PMC94 = 1	PFCE94 = 1	PFC94 = 1	注 2
P95	TIAA30	输入	P95 = 不需要设置	PM95 = 不需要设置	PMC95 = 1	PFCE95 = 0	PFC95 = 0	
	TOAA30	输出	P95 = 不需要设置	PM95 = 不需要设置	PMC95 = 1	PFCE95 = 0	PFC95 = 1	
	A5 ^{注 1}	输出	P95 = 不需要设置	PM95 = 不需要设置	PMC95 = 1	PFCE95 = 1	PFC95 = 0	注 2
P96	TIAA21	输入	P96 = 不需要设置	PM96 = 不需要设置	PMC96 = 1	PFCE96 = 0	PFC96 = 0	
	TOAA21	输出	P96 = 不需要设置	PM96 = 不需要设置	PMC96 = 1	PFCE96 = 1	PFC96 = 0	
	INTP11	输入	P96 = 不需要设置	PM96 = 不需要设置	PMC96 = 1	PFCE96 = 1	PFC96 = 0	
	A6 ^{注 1}	输出	P96 = 不需要设置	PM96 = 不需要设置	PMC96 = 1	PFCE96 = 1	PFC96 = 0	注 2
P97	SIF1	输入	P97 = 不需要设置	PM97 = 不需要设置	PMC97 = 1	PFCE97 = 0	PFC97 = 0	
	TIAA20	输入	P97 = 不需要设置	PM97 = 不需要设置	PMC97 = 1	PFCE97 = 0	PFC97 = 1	
	TOAA20	输出	P97 = 不需要设置	PM97 = 不需要设置	PMC97 = 1	PFCE97 = 1	PFC97 = 0	
	A7 ^{注 1}	输出	P97 = 不需要设置	PM97 = 不需要设置	PMC97 = 1	PFCE97 = 1	PFC97 = 1	注 2
P98	SOF1	输出	P98 = 不需要设置	PM98 = 不需要设置	PMC98 = 1	PFCE98 = 0	PFC98 = 0	
	INTP12	输入	P98 = 不需要设置	PM98 = 不需要设置	PMC98 = 1	PFCE98 = 0	PFC98 = 1	
	A8 ^{注 1}	输出	P98 = 不需要设置	PM98 = 不需要设置	PMC98 = 1	PFCE98 = 1	PFC98 = 0	注 2

注 1. 仅限于 V850ES/JH3-H
2. 将 A0 至 A15 引脚用作复用功能时，立即将 PMC9 寄存器的所有 16 位设置为 FFFFH。

表 4-20. 端口引脚用作复用功能引脚（8/10）

引脚名称	复用功能		Pn 寄存器的 Pnx 位	PMn 寄存器的 PMnx 位	PMCn 寄存器的 PMCnx 位	PFCEn 寄存器的 PFCEnx 位	PFCn 寄存器的 PFCnx 位	其它位 (寄存器)
	名称	I/O						
P99	SCKF1	I/O	P99 = 不需要设置	PM99 = 不需要设置	PMC99 = 1	PFCE99 = 0	PFC99 = 0	
	INTP14	输入	P99 = 不需要设置	PM99 = 不需要设置	PMC99 = 1	PFCE99 = 0	PFC99 = 1	
	A9 ^{注 1}	输出	P99 = 不需要设置	PM99 = 不需要设置	PMC99 = 1	PFCE99 = 1	PFC99 = 0	注 2
P910	SIF3	输入	P910 = 不需要设置	PM910 = 不需要设置	PMC910 = 1	PFCE910 = 0	PFC910 = 0	
	TXDC2	输出	P910 = 不需要设置	PM910 = 不需要设置	PMC910 = 1	PFCE910 = 0	PFC910 = 1	
	INTP14	输入	P910 = 不需要设置	PM910 = 不需要设置	PMC910 = 1	PFCE910 = 1	PFC910 = 0	
	A10 ^{注 1}	输出	P910 = 不需要设置	PM910 = 不需要设置	PMC910 = 1	PFCE910 = 1	PFC910 = 1	注 2
P911	SOF3	输出	P911 = 不需要设置	PM911 = 不需要设置	PMC911 = 1	PFCE911 = 0	PFC911 = 0	
	RXDC2	输入	P911 = 不需要设置	PM911 = 不需要设置	PMC911 = 1	PFCE911 = 0	PFC911 = 1	
	INTP15	输入	P911 = 不需要设置	PM911 = 不需要设置	PMC911 = 1	PFCE911 = 1	PFC911 = 0	
	A11 ^{注 1}	输出	P911 = 不需要设置	PM911 = 不需要设置	PMC911 = 1	PFCE911 = 1	PFC911 = 1	注 2
P912	SCKF3	I/O	P912 = 不需要设置	PM912 = 不需要设置	PMC912 = 1	–	PFC912 = 0	
	A12 ^{注 1}	输出	P912 = 不需要设置	PM912 = 不需要设置	PMC912 = 1	–	PFC912 = 1	注 2
P913	TOAB1OFF	输入	P913 = 不需要设置	PM913 = 不需要设置	PMC913 = 1	–	PFC913 = 0	
	INTP16	输入	P913 = 不需要设置	PM913 = 不需要设置	PMC913 = 1	–	PFC913 = 0	
	A13 ^{注 1}	输出	P913 = 不需要设置	PM913 = 不需要设置	PMC913 = 1	–	PFC913 = 1	注 2
P914	TIAA51	输入	P914 = 不需要设置	PM914 = 不需要设置	PMC914 = 1	PFCE914 = 0	PFC914 = 0	
	TOAA51	输出	P914 = 不需要设置	PM914 = 不需要设置	PMC914 = 1	PFCE914 = 0	PFC914 = 1	
	INTP17	输入	P914 = 不需要设置	PM914 = 不需要设置	PMC914 = 1	PFCE914 = 1	PFC914 = 0	
	A14 ^{注 1}	输出	P914 = 不需要设置	PM914 = 不需要设置	PMC914 = 1	PFCE914 = 1	PFC914 = 1	注 2
P915	TIAA50	输入	P915 = 不需要设置	PM915 = 不需要设置	PMC915 = 1	PFCE915 = 0	PFC915 = 0	
	TOP50	输出	P915 = 不需要设置	PM915 = 不需要设置	PMC915 = 1	PFCE915 = 0	PFC915 = 1	
	INTP18	输入	P915 = 不需要设置	PM915 = 不需要设置	PMC915 = 1	PFCE915 = 1	PFC915 = 0	
	A15 ^{注 1}	输出	P915 = 不需要设置	PM915 = 不需要设置	PMC915 = 1	PFCE915 = 1	PFC915 = 1	注 2

注 1. 仅限于 V850ES/JH3-H

2. 将 A0 至 A15 引脚用作复用功能时，立即将 PMC9 寄存器的所有 16 位设置为 FFFFH。

表 4-20. 端口引脚用作复用功能引脚（9/10）

引脚名称	复用功能		Pn 寄存器的 Pnx 位	PMn 寄存器的 PMnx 位	PMCn 寄存器的 PMCnx 位	PFCEn 寄存器的 PFCEnx 位	PFCn 寄存器的 PFCnx 位	其它位 (寄存器)
	名称	I/O						
PCM0	WAIT [※]	输入	PCM0 = 不需要设置	PMCM0 = 不需要设置	PMCCM0 = 1	—	—	
PCM1	CLKOUT	输出	PCM1 = 不需要设置	PMCM1 = 不需要设置	PMCCM1 = 1	—	—	
PCM2	HLD $\overline{\text{AK}}$ [※]	输出	PCM2 = 不需要设置	PMCM2 = 不需要设置	PMCCM2 = 1	—	—	
PCM3	H $\overline{\text{LDRQ}}$ [※]	输入	PCM3 = 不需要设置	PMCM3 = 不需要设置	PMCCM3 = 1	—	—	
PCS0	$\overline{\text{CS0}}$ [※]	输出	PCS0 = 不需要设置	PMCS0 = 不需要设置	PMCCS0 = 1	—	—	
PCS2	$\overline{\text{CS2}}$ [※]	输出	PCS2 = 不需要设置	PMCS2 = 不需要设置	PMCCS2 = 1	—	—	
PCS3	$\overline{\text{CS3}}$ [※]	输出	PCS3 = 不需要设置	PMCS3 = 不需要设置	PMCCS3 = 1	—	—	
PCT0	$\overline{\text{WR0}}$	输出	PCT0 = 不需要设置	PMCT0 = 不需要设置	PMCC $\overline{\text{T0}}$ = 1	—	—	
PCT1	$\overline{\text{WR1}}$	输出	PCT1 = 不需要设置	PMCT1 = 不需要设置	PMCC $\overline{\text{T1}}$ = 1	—	—	
PCT4	$\overline{\text{RD}}$ [※]	输出	PCT4 = 不需要设置	PMCT4 = 不需要设置	PMCC $\overline{\text{T4}}$ = 1	—	—	
PCT6	ASTB [※]	输出	PCT6 = 不需要设置	PMCT6 = 不需要设置	PMCC $\overline{\text{T6}}$ = 1	—	—	
PDH0	A16	输出	PDH0 = 不需要设置	PMDH0 = 不需要设置	PMCDH0 = 1	—	—	
PDH1	A17	输出	PDH1 = 不需要设置	PMDH1 = 不需要设置	PMCDH1 = 1	—	—	
PDH2	A18	输出	PDH2 = 不需要设置	PMDH2 = 不需要设置	PMCDH2 = 1	—	—	
PDH3	A19	输出	PDH3 = 不需要设置	PMDH3 = 不需要设置	PMCDH3 = 1	—	—	
PDH4	A20	输出	PDH4 = 不需要设置	PMDH4 = 不需要设置	PMCDH4 = 1	—	—	
PDH5	A21	输出	PDH5 = 不需要设置	PMDH5 = 不需要设置	PMCDH5 = 1	—	—	
PDL0	AD0	I/O	PDL0 = 不需要设置	PMDL0 = 不需要设置	PMCDL0 = 1	—	—	
PDL1	AD1	I/O	PDL1 = 不需要设置	PMDL1 = 不需要设置	PMCDL1 = 1	—	—	
PDL2	AD2	I/O	PDL2 = 不需要设置	PMDL2 = 不需要设置	PMCDL2 = 1	—	—	
PDL3	AD3	I/O	PDL3 = 不需要设置	PMDL3 = 不需要设置	PMCDL3 = 1	—	—	
PDL4	AD4	I/O	PDL4 = 不需要设置	PMDL4 = 不需要设置	PMCDL4 = 1	—	—	

注 仅限于 V850ES/JH3-H

表 4-20. 端口引脚用作复用功能引脚（10/10）

引脚名称	复用功能		Pn 寄存器的 Pnx 位	PMn 寄存器的 PMnx 位	PMCn 寄存器的 PMCnx 位	PFCEn 寄存器的 PFCEnx 位	PFCn 寄存器的 PFCnx 位	其它位 (寄存器)
	名称	I/O						
PDL5	AD5	I/O	PDL5 = 不需要设置	PMDL5 = 不需要设置	PMCDL5 = 1	–	–	
	FLMD1 ^注	输入	PDL5 = 不需要设置	PMDL5 = 不需要设置	PMCDL5 = 不需要设置	–	–	
PDL6	AD6	I/O	PDL6 = 不需要设置	PMDL6 = 不需要设置	PMCDL6 = 1	–	–	
PDL7	AD7	I/O	PDL7 = 不需要设置	PMDL7 = 不需要设置	PMCDL7 = 1	–	–	
PDL8	AD8	I/O	PDL8 = 不需要设置	PMDL8 = 不需要设置	PMCDL8 = 1	–	–	
PDL9	AD9	I/O	PDL9 = 不需要设置	PMDL9 = 不需要设置	PMCDL9 = 1	–	–	
PDL10	AD10	I/O	PDL10 = 不需要设置	PMDL10 = 不需要设置	PMCDL10 = 1	–	–	
PDL11	AD11	I/O	PDL11 = 不需要设置	PMDL11 = 不需要设置	PMCDL11 = 1	–	–	
PDL12	AD12	I/O	PDL12 = 不需要设置	PMDL12 = 不需要设置	PMCDL12 = 1	–	–	
PDL13	AD13	I/O	PDL13 = 不需要设置	PMDL13 = 不需要设置	PMCDL13 = 1	–	–	
PDL14	AD14	I/O	PDL14 = 不需要设置	PMDL14 = 不需要设置	PMCDL14 = 1	–	–	
PDL15	AD15	I/O	PDL15 = 不需要设置	PMDL15 = 不需要设置	PMCDL15 = 1	–	–	

注 该引脚设置为 flash 存储器编程模式后，其不需要通过端口控制寄存器进行操作。细节参见第三十一章 flash 存储器。

4.5 注意事项

4.5.1 设置端口引脚的注意事项

(1) 在 V850ES/JG3-H 和 V850ES/JH3-H 中，通用端口功能与一些外设功能 I/O 引脚共享引脚。通过 PMCN 寄存器设置使得通用端口（端口模式）与外围功能 I/O 引脚（复用功能模式）之间相互转换。关于该寄存器的设置顺序注意以下事项：

(a) 从端口模式至复用功能模式转换的注意事项

从端口模式至复用功能模式的转换按以下顺序进行：

<1> 设置 PFn 寄存器 ^{#1} ：	N 沟道漏极开路设置
<2>设置 PFCn 和 PFCEn 寄存器：	复用功能选择
<3> 设置 PMCn 寄存器的相应的位为 1：	转换至复用功能模式
<4>设置 INTRn 和 INTFn 寄存器 ^{#2}	外部中断设置

如果先设置 PMCN 寄存器，则有可能发生不可预料的操作，或者依照 PFn，PFCn 和 PFCEn 寄存器的设置而引脚状态发生变化时，也可能发生不可预料的操作。

正确举例如下：

- 注
1. 只有 N 沟道漏极开路输出
 2. 只有选择外部中断功能时

注意事项 不考虑端口模式还是复用功能模式，Pn 寄存器按以下进行读取和写入。

- Pn 寄存器读取： 读取端口输出锁存值（PMn.PMnm 位 = 0 时），或者读取引脚状态（PMn.PMnm 位 = 1）。
- Pn 寄存器写入： 写入端口输出锁存。

[实例] SCL01 引脚设置举例

SCL01 引脚用于复用 P41/SOF0 引脚。用 PMC4，PFC4 和 PF4 寄存器选择有效引脚功能。

PMC41 位	PFC41 位	PF41 位	有效引脚功能
0	无需理会	1	P41（输出端口模式下，N 沟道漏极开路输出）
1	0	1	SOF0 输出（N 沟道漏极开路输出）
	1	1	SCL01 I/O（N 沟道漏极开路输出）

从 P41 引脚转换至 SCL01 引脚时可能引起故障的设置步骤如下所示：

设置步骤：	设置内容	引脚状态	引脚电平
<1>	初始值 (PMC41 位 = 0, PFC41 位 = 0, PF41 位 = 0)	端口描述 (输入)	高阻
<2>	PMC41 位 ← 1	SOF0 输出	低电平 (根据 CSIF0 设置为高电平)
<3>	PFC41 位 ← 1	SCL01 I/O	高电平 (CMOS 输出)
<4>	PF41 位 ← 1	SCL01 I/O	高阻 (N 沟道漏极开路输出)

步骤<2>中，因为复用功能 SOF0 输出是输出至该引脚，所以 I²C 通信可能会受影响。步骤<2>或 <3>的 CMOS 输出期间，有可能产生不必要的电流。

(b) 复用功能模式（输入）的注意事项

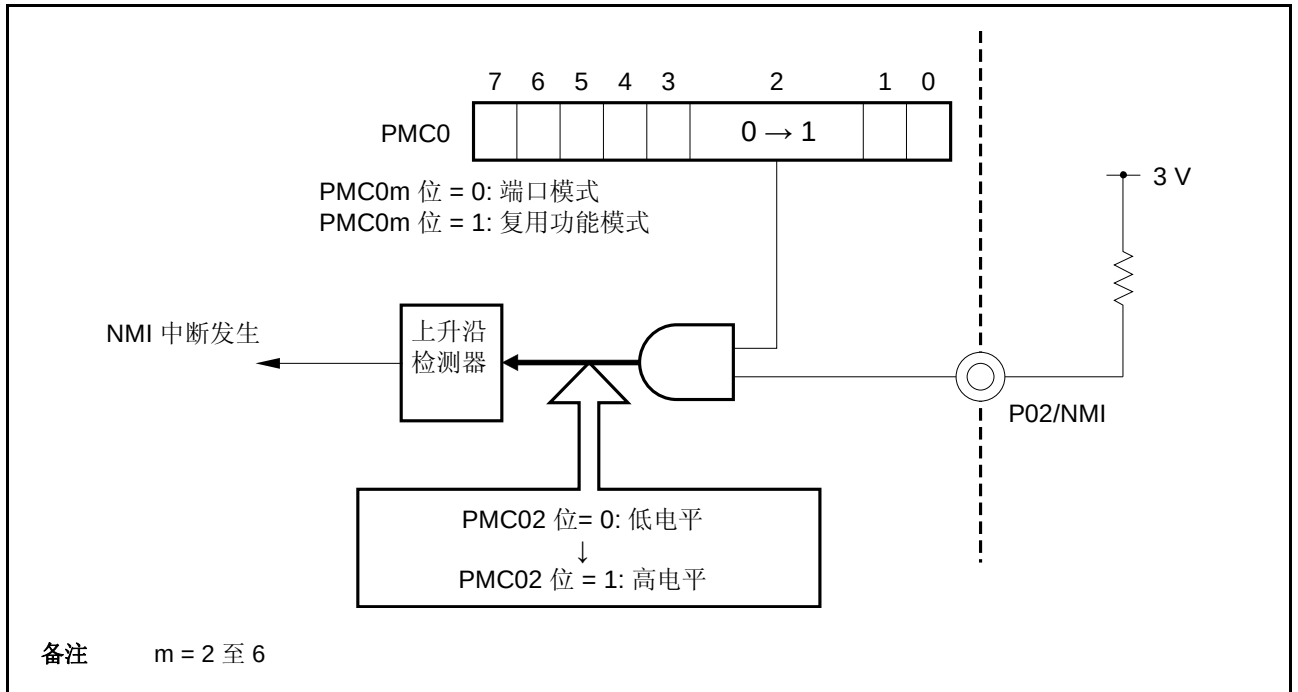
由于 PMCn 寄存器设定值和该引脚电平相与输出，所以，当 PMCn.PMCnm 位为 0 时，输入至复用功能模块的信号为低电平。因此，依据端口设置和复用功能操作使能时序，可能会发生不可预料操作。所以从端口模式至复用功能模式的转换按以下顺序进行：

- 从端口模式至复用功能模式转换（输入）
用 PMCn 寄存器将引脚设置为复用功能模式，然后允许复用功能操作。
- 从复用功能模式至端口模式转换（输入）
停止复用功能操作，然后将引脚转换至端口模式。

具体实例如[例 1]和[例 2]。

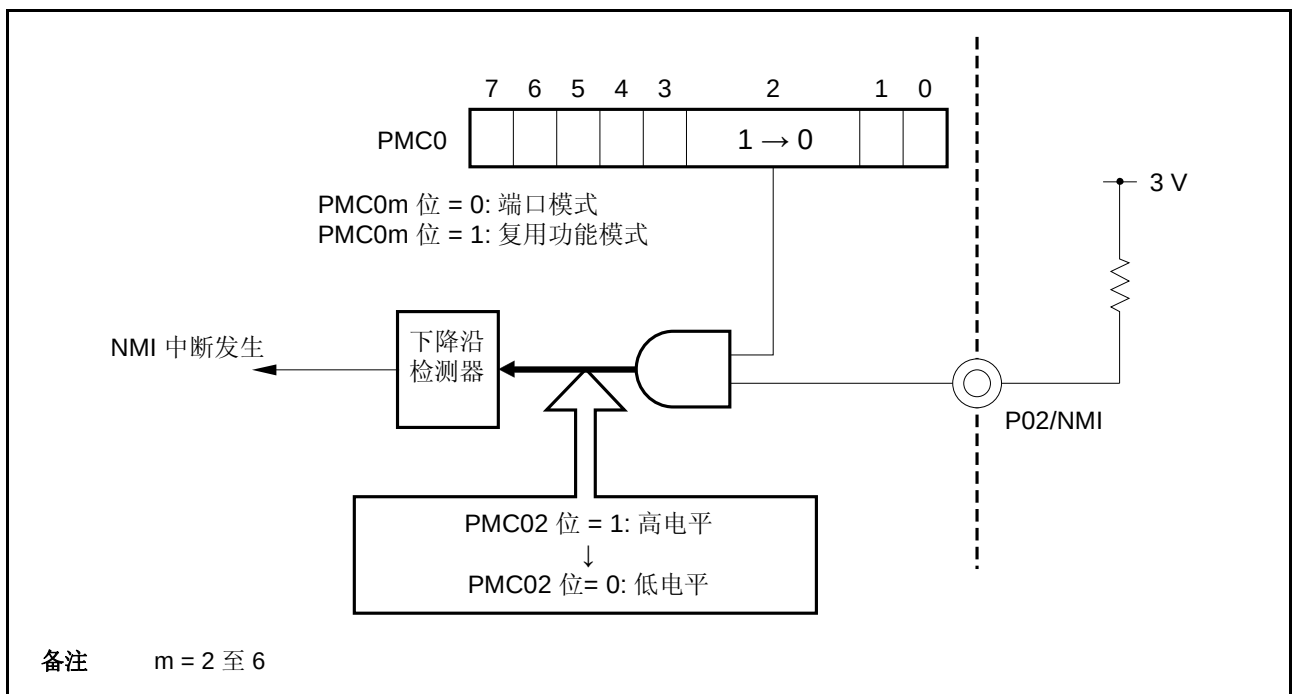
[例 1] 从通用端口（P02）转换至外部中断引脚（NMI）
P02/NMI 引脚上拉如图 4-4 且上升沿被 NMI 引脚边沿检测设置指定时，在 P02 引脚转换至 NMI 引脚（PMC02 位 = 0 → 1）期间，即使高电平连续输入至 NMI 引脚，也会检测为低电平变成高电平的上升沿，产生 NMI 中断。
为避免该情况，从 P02 引脚转换至 NMI 引脚完成后，再设置 NMI 引脚的有效沿。

图 4-4.从 P02 至 NMI 转换举例（不正确）



- [例 2] 从外部引脚（NMI）转换至通用端口（P02）
P02/NMI 引脚上拉如图 4-5 且下降沿被 NMI 引脚边沿检测设置指定时，在 NMI 引脚转换至 P02 引脚（PMC02 位 = 1 → 0）期间，即使高电平连续输入至 NMI 引脚，也会检测为高电平变成低电平的下降沿，产生 NMI 中断。
为避免此情况，在转换至 P02 引脚之前设置 NMI 引脚边沿检测为无边沿检测。

图 4-5.从 NMI 至 P02 转换举例（不正确）



(2) 端口模式下，PFn.PFnm 位仅在输出模式（PMn.PMnm 位 = 0）有效。输入模式（PMnm 位 = 1）下，PFnm 位的值不在缓冲器中反映。

4.5.2 端口n寄存器（Pn）位操作指令注意事项

在同时具有输入和输出功能的端口执行位操作指令时，除了正常写入目标位外，还可能对操作对象之外的输入端口的输出锁存器写入不需要的值。

所以，当一个端口由输入模式转变为输出模式时，建议重写输出锁存器。

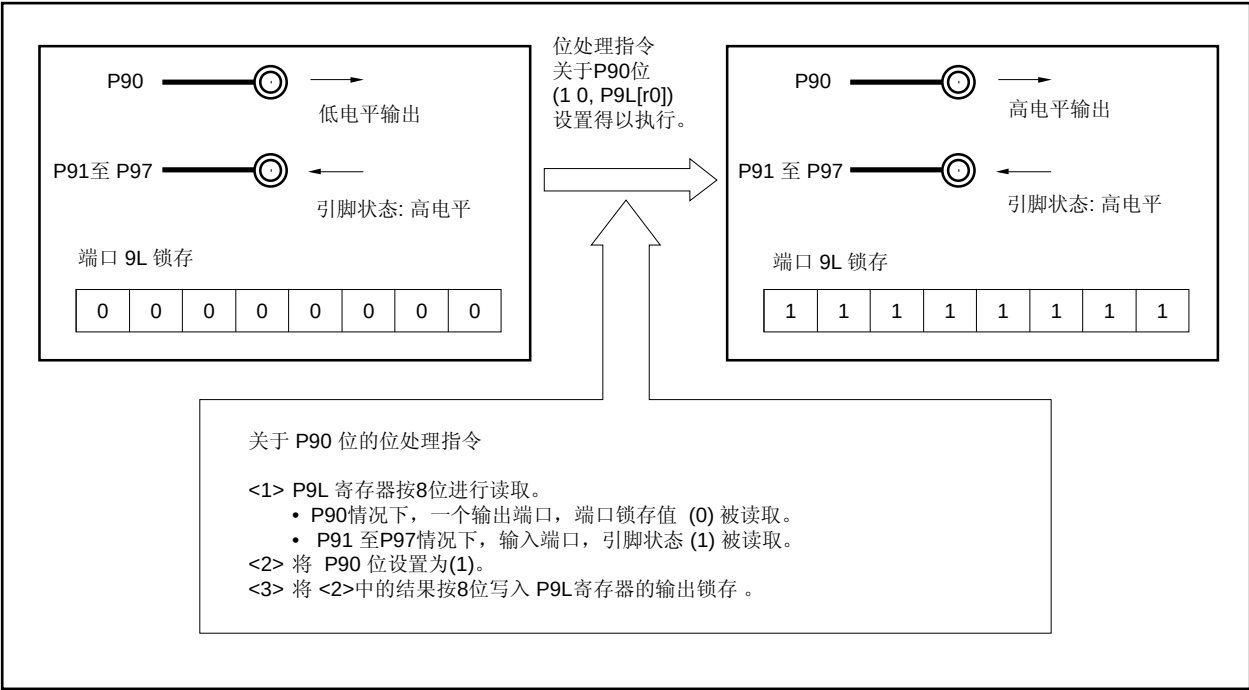
<举例说明> 当 P90 为一个输出端口，同时，P91 到 P97 作为输入端口（所有引脚状态为高电平），而端口锁存器的值为 00H。如果 P90 引脚的输出由于位操作指令的执行由低电平变为高电平，则端口锁存的值为 FFH。

解释：写入或读取 PMnm 位为 1 的端口的 Pn 寄存器的对象是各自的输出锁存器和引脚的状态。
V850ES/JG3-H 和 V850ES/JH3-H 按以下次序执行位处理指令：

- <1> 按 8 位读取 Pn 寄存器。
- <2> 处理目标位。
- <3> 按 8 位写入 Pn 寄存器。

在步骤 <1>中，当作为输入端口的 P91 至 P97 引脚状态被读取时，作为输出端口的 P90 引脚的输出锁存（0）的值被读取。如果此时 P91 至 P97 引脚的状态为高电平，则读取的值为 FEH。
通过在<2>中的操作该值变成 FFH。
通过在<3>中的操作 FFH 被写入输出锁存。

图 4-6. 位处理指令（P90 引脚）



4.5.3 片上调试引脚的注意事项（仅限于V850ES/JG3-H）

$\overline{\text{DRST}}$, DCK , DMS , DDI 和 DDO 引脚是片上调试引脚。

由 $\overline{\text{RESET}}$ 引脚复位以后, $\text{P56/INTP05/}\overline{\text{DRST}}$ 引脚被初始化为片上调试引脚 ($\overline{\text{DRST}}$)。如果此时 $\overline{\text{DRST}}$ 输入高电平, 则设置为片上调试功能, 就可以使用 DCK , DMS , DDI 和 DDO 引脚。

如果不使用片上调试功能, 必须进行以下操作。

- 将 OCDM 寄存器（特殊寄存器）的 OCDM0 位清 0。

此时, 从 $\overline{\text{REST}}$ 引脚的复位解除后直到上述操作完成, 应使 $\text{P56/INTP05/}\overline{\text{DRST}}$ 引脚固定为低电平。

如果上述动作之前 $\overline{\text{DRST}}$ 的输入为高电平, 这将引起一个故障（CPU 死锁）。操作 P56 引脚时必须格外小心。

注意事项 由 WDT2RES 信号复位以后, 时钟监视器（ CLM ）, 或低压检测器（ LVI ）, $\text{P56/INTP05/}\overline{\text{DRST}}$ 引脚并不会初始化为片上调试功能引脚 ($\overline{\text{DRST}}$)。 OCDM 寄存器保持当前值。

4.5.4 $\text{P56/INTP05/}\overline{\text{DRST}}$ 引脚的注意事项

$\text{P56/INTP05/}\overline{\text{DRST}}$ 引脚有一个内部下拉电阻（30 k Ω 典型值）。通过 $\overline{\text{RESET}}$ 引脚复位后, 一个下拉电阻会被连接。当 OCDM0 位被清除为 0 时, 该下拉电阻会被断开。

4.5.5 电源开启时 P10 , P11 和 P53 引脚的注意事项

电源开启时在复位期间以下引脚可能暂时输出不确定电平。

- P10/ANO0 引脚
- P11/ANO1 引脚
- $\text{P53/SIF2/TIAB00/KR3/TOAB00/RTP03/DDO}$ 引脚（仅限于 V850ES/JG3-H）

4.5.6 迟滞特性

端口模式下, 以下引脚不具备迟滞特性:

P00 至 P05
 P20 至 P25
 P30 至 P37
 P40 至 P42
 P50 至 P56
 P60 至 P65
 P90 至 P915

第五章 总线控制功能

V850ES/JG3-H 和 V850ES/JH3-H 单片机提供外部总线接口，通过该接口可以连接外部存储器（诸如 ROM、RAM 等）和 I/O 设备。

5.1 特性

- 输出可以选择为复用总线输出模式，最小存取周期为 3 个总线周期，或选择为独立总线输出模式。
（仅限于 V850ES/JH3-H；V850ES/JG3-H 只支持复用总线模式。）
- 8 位/16 位数据总线可选
- 等待功能
 - 最高可达 7 种状态的可编程等待功能
 - 使用 **WAIT** 引脚的外部等待功能
- 空闲状态功能
- 总线保持功能

5.2 总线控制引脚

用于连接外部设备的引脚如下表：

表 5-1. V850ES/JH3-总线控制引脚（复用总线）

总线控制引脚	复用功能引脚	I/O	功能
AD0 至 AD15	PDL0 至 PDL15	I/O	地址/数据总线
A16 至 A23	PDH0 至 PDH5	输出	地址总线
$\overline{\text{WAIT}}$	PCM0	输入	外部等待控制
CLKOUT	PCM1	输出	内部系统时钟
$\overline{\text{WR0}}, \overline{\text{WR1}}$	PCT0, PCT1	输出	写选通信号
$\overline{\text{RD}}$	PCT4	输出	读选通信号
ASTB	PCT6	输出	地址选通信号
$\overline{\text{HLDRQ}}$	PCM3	输入	总线保持控制
$\overline{\text{HLDK}}$	PCM2	输出	
$\overline{\text{CS0}}, \overline{\text{CS2}}, \overline{\text{CS3}}$	PCS0, PCS2, PCS3	输出	片选

表 5-2. V850ES/JH3-总线控制引脚（独立总线）

总线控制引脚	复用功能引脚	I/O	功能
AD0 至 AD15	PDL0 至 PDL15	I/O	数据总线
A0 至 A15	P90 至 P915	输出	地址总线
A16 至 A21	PDH0 至 PDH5	输出	地址总线
$\overline{\text{WAIT}}$	PCM0	输入	外部等待控制
CLKOUT	PCM1	输出	内部系统时钟
$\overline{\text{WR0}}, \overline{\text{WR1}}$	PCT0, PCT1	输出	写选通信号
$\overline{\text{RD}}$	PCT4	输出	读选通信号
$\overline{\text{HLDRQ}}$	PCM3	输入	总线保持控制
$\overline{\text{HLDK}}$	PCM2	输出	
$\overline{\text{CS0}}, \overline{\text{CS2}}, \overline{\text{CS3}}$	PCS0, PCS2, PCS3	输出	片选

表 5-3. V850ES/JG3-H 总线控制引脚（复用总线）

总线控制引脚	复用功能引脚	I/O	功能
AD0 至 AD15	PDL0 至 PDL15	I/O	地址/数据总线
$\overline{\text{WAIT}}$	PCM0	输入	外部等待控制
CLKOUT	PCM1	输出	内部系统时钟
$\overline{\text{WR0}}, \overline{\text{WR1}}$	PCT0, PCT1	输出	写选通信号
$\overline{\text{RD}}$	P61	输出	读选通信号
ASTB	P62	输出	地址选通信号
$\overline{\text{CS0}}, \overline{\text{CS2}}, \overline{\text{CS3}}$	P63, P64, P65	输出	片选

5.2.1 访问片内ROM存储器、片内RAM存储器或片上外围输入/输出端口时的引脚状态

当访问片内 ROM 存储器、片内 RAM 存储器或片上外围输入/输出端口时，引脚状态如下：

表 5-4. 当访问片内 ROM 存储器、片内 RAM 存储器或片上外围输入/输出端口时的引脚状态

总线控制引脚	独立总线模式		复用总线模式	
	内部 ROM/RAM	外围输入/输出	内部 ROM/RAM	外围输入/输出
地址/数据总线 (AD15 至 AD0)	未定义	未定义	未定义	未定义
地址总线 (A23 至 A16)	未定义	未定义 (访问期间地址输出)	未定义	未定义 (访问期间地址输出)
地址总线 (A15 至 A0)	未定义	未定义 (访问期间地址输出)	未定义	未定义 (访问期间地址输出)
控制信号	无效	无效	无效	无效

注意事项 当对内部 ROM 区域进行写入时，地址总线、数据总线和控制总线的信号均有效，访问外部存储器时亦然。

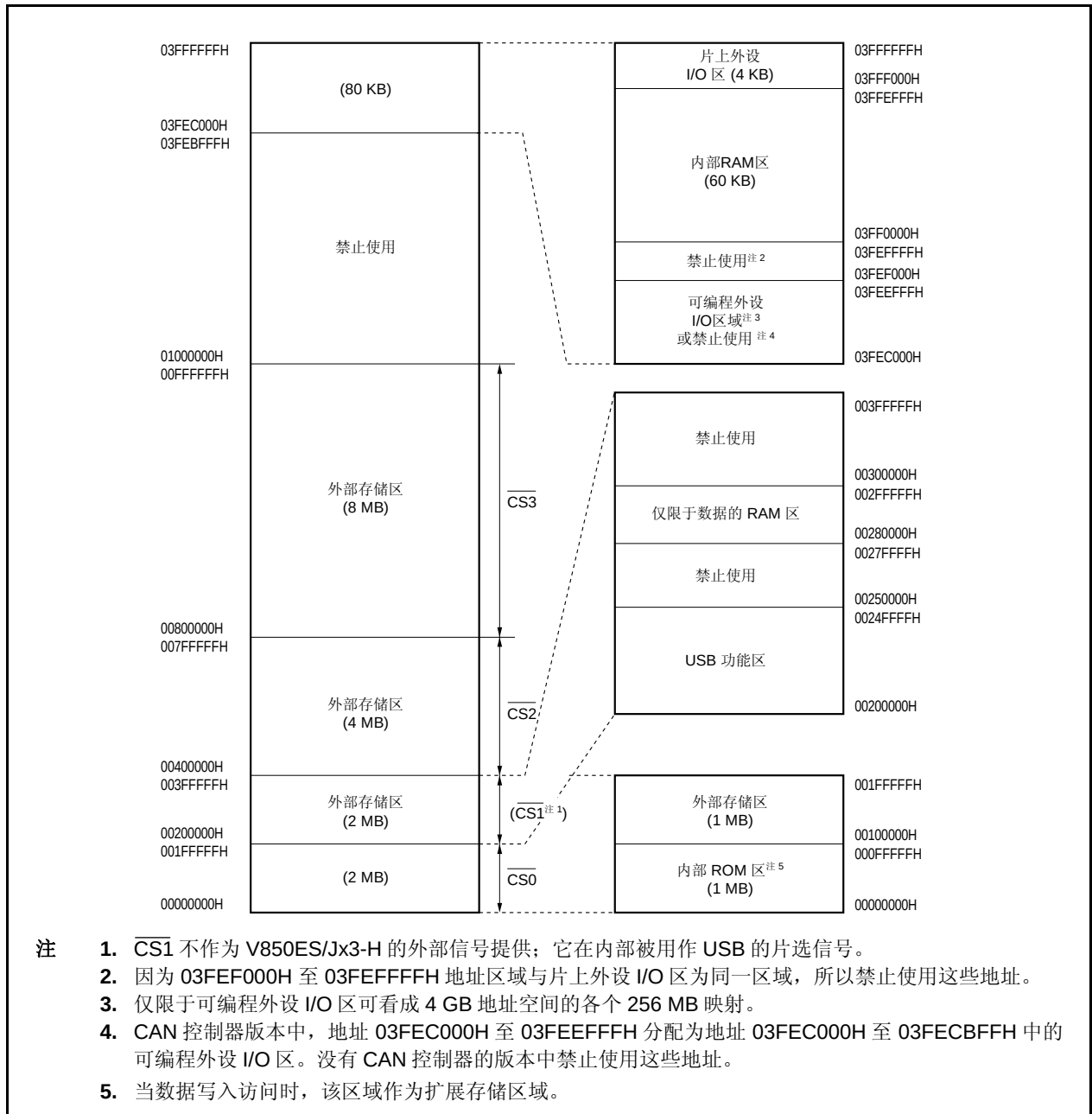
5.2.2 各种工作模式下的引脚状态

关于 V850ES/JG3-H 和 V850ES/JH3-H 在各种工作模式下的引脚状态，参见 2.2 引脚状态。

5.3 存储块功能

16 MB 外部存储空间从低被依次划分成 2 MB、4 MB 和 8 MB 的存储块。在可编程等待功能模式和总线周期工作模式下，可分别独立控制各片选区。

图 5-1. 数据存储映射：物理地址



5.4 总线访问

5.4.1 总线访问时钟数目

访问各资源所需的基本时钟数目如下表：

总线周期类型 \ 区域（总线宽度）	内部 ROM（32 位）	内部 RAM（32 位）	外部存储器（16 位）	
			复用	独立 ^{注 1}
取指周期（常规访问）	1	1 ^{注 2}	3 + n	
取指周期（分支跳转）	2	2 ^{注 1}	3 + n	
操作数访问	5	1	3 + n	

- 注 1. 仅限于 V850ES/JH3-H
 2. 如果与数据访问发生冲突则加 1。

备注 单元：时钟/访问

5.4.2 总线宽度设置功能

由存储模块 $\overline{CSn}$ 选择的每一个外存储区都可以通过 BSC 寄存器设置。然而，总线宽度只可设置为 8 位和 16 位。
V850ES/JG3-H 和 V850ES/JH3-H 的外部存储区通过存储块 $\overline{CS0}$ ， $\overline{CS2}$ 和 $\overline{CS3}$ 进行选择。

（1）总线宽度设置寄存器（BSC）

BSC 寄存器可以按 16 位宽度进行读写。
系统复位后该寄存器的值为 5555H。

注意事项 复位完成后对 BSC 寄存器执行写操作，之后不要改变该设定值。并且，在 BSC 寄存器初始设置完成之前，不要访问外部存储器。

复位后: 5555H R/W 地址: FFFFF066H

BSC

15	14	13	12	11	10	9	8
0	1	0	1	0	1	0	1
7	6	5	4	3	2	1	0
0	BS30	0	BS20	0	1	0	BS00
	CS3		CS2				CS0

BSn0	存储块 CSn 空间的数据总线宽度 (n = 0, 2, 3)
0	8 位
1	16 位

注意事项 务必将位 14, 12, 10, 8 和 2 设置为“1”，并且将位 15, 13, 11, 9, 7, 5, 3 和 1 清除为“0”。

5.4.3 根据不同总线宽度进行访问

V850ES/JG3-H 和 V850ES/JH3-H 按照 8 位、16 位或 32 位总线宽度访问片上外围输入/输出端口和外部存储器。总线宽度说明如下：

- 片上外围输入/输出端口的总线宽度固定为 16 位。
- 外部存储器的总线宽度可选择 8 位或 16 位（利用 BSC 寄存器进行选择）。

以上各情况访问时的操作描述如下：所有数据访问都从低位开始。

V850ES/JG3-H 和 V850ES/JH3-H 仅支持“小端序”数据格式。

图 5-2. 32 位小端序地址

31	24 23	16 15	8 7	0
000BH	000AH	0009H	0008H	
0007H	0006H	0005H	0004H	
0003H	0002H	0001H	0000H	

(1) 数据空间

V850ES/JG3-H 和 V850ES/JH3-H 具有地址不对齐功能。

基于该功能，各地址都可以存储数据，无论该数据是何种格式（字或半字）。然而，如果字数据或半字数据没有边界对齐，那么，至少产生两次总线访问周期，导致总线效率降低。

(a) 访问半字长度数据

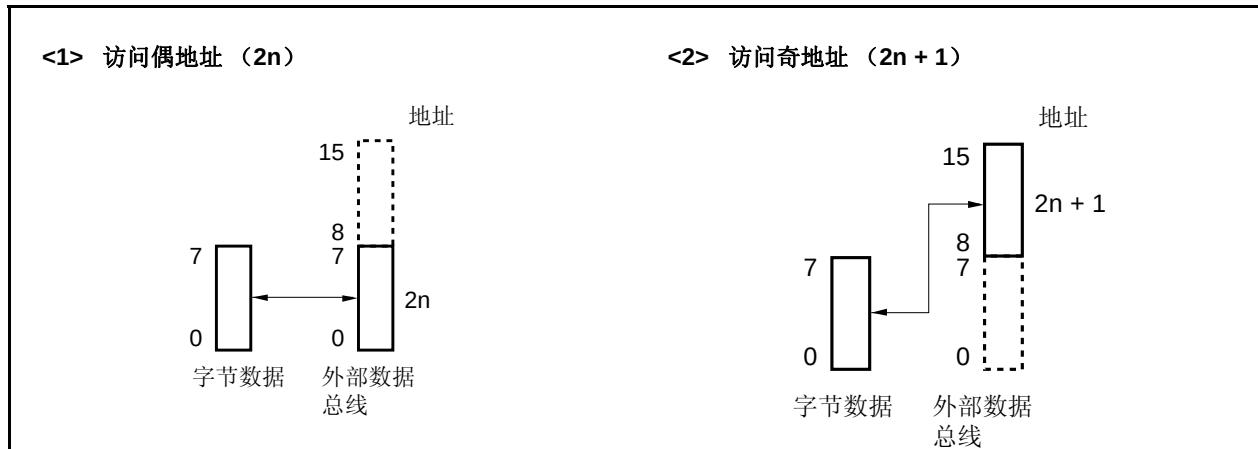
如果地址的最低有效位为 1，一个字节长度的总线访问周期需要产生两次。

(b) 访问字长度数据

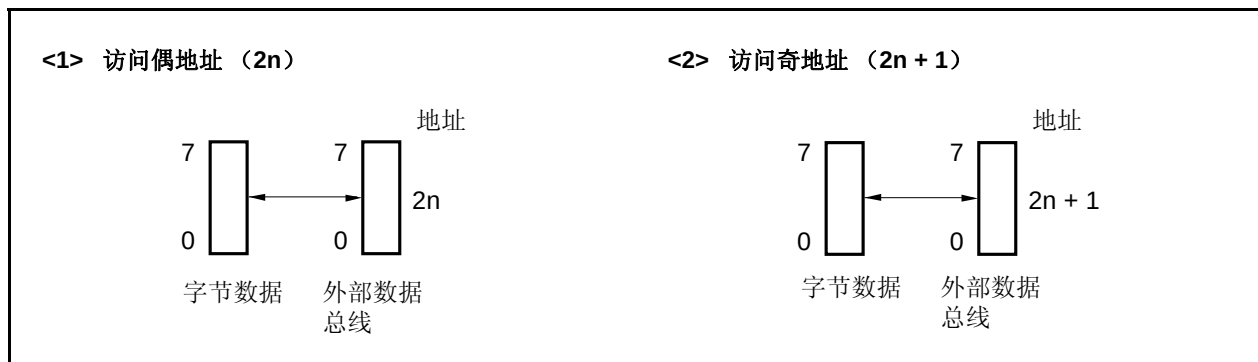
- (i) 如果地址的最低有效位为 1，所需总线访问周期按这样的次序：一个字节访问周期、一个半字访问周期、一个字节访问周期。
- (ii) 如果地址的最低两个有效位为 10，需要发生两次半字长度的总线访问周期。

(2) 字节访问 (8 位)

(a) 16 位数据总线宽度

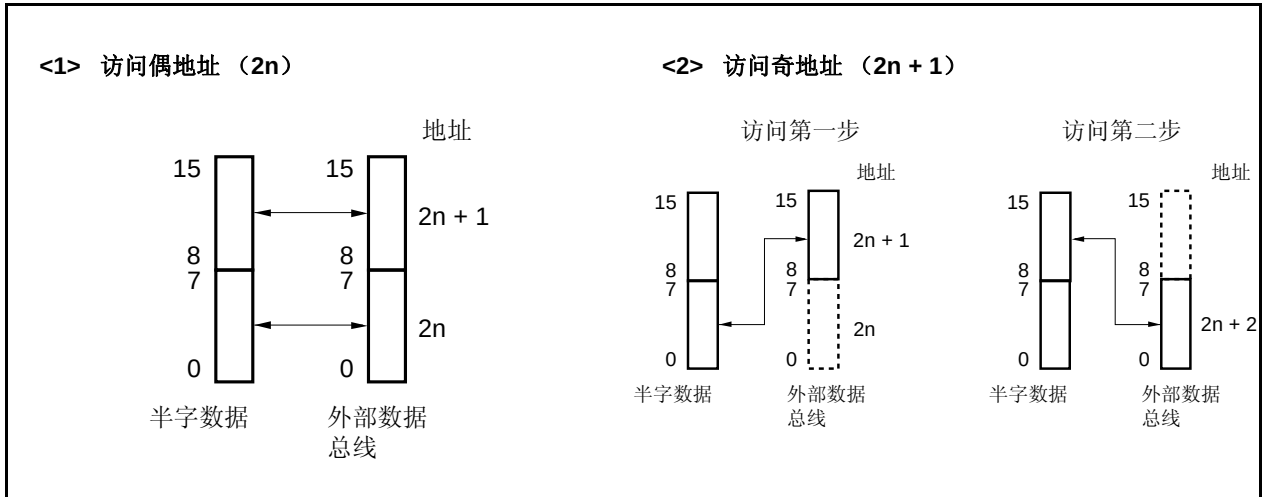


(b) 8 位数据总线宽度



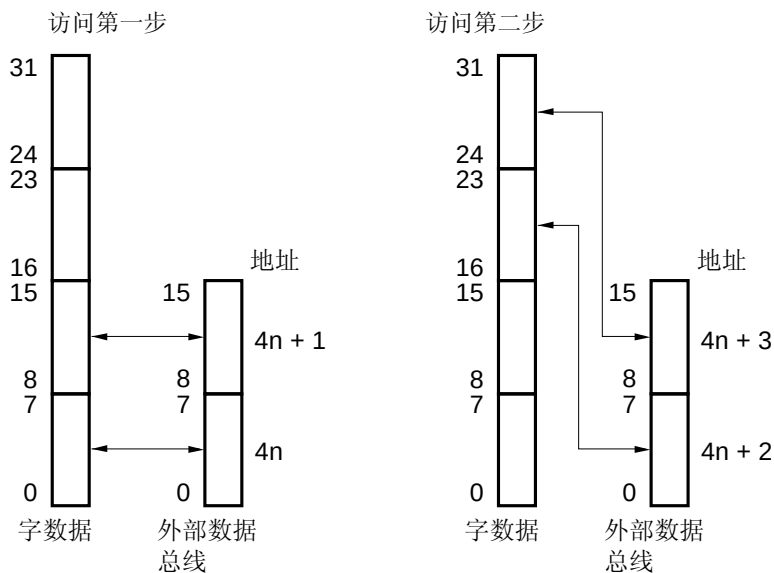
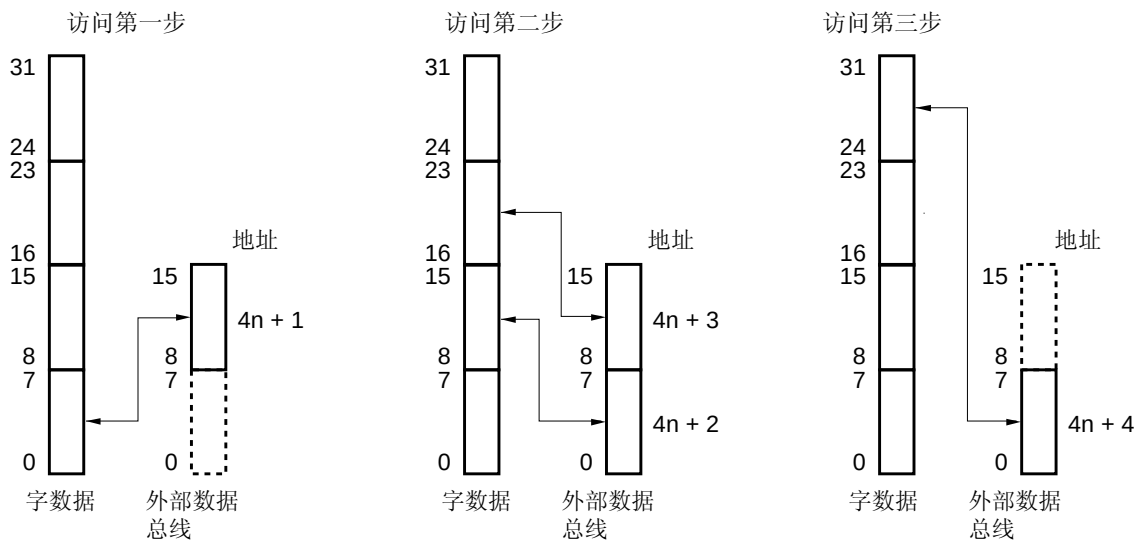
(3) 半字访问 (16 位)

(a) 16 位数据总线宽度

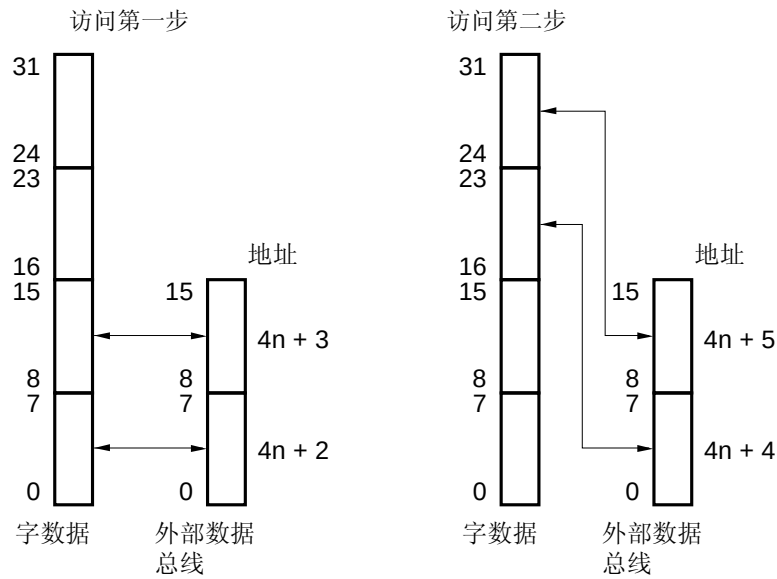
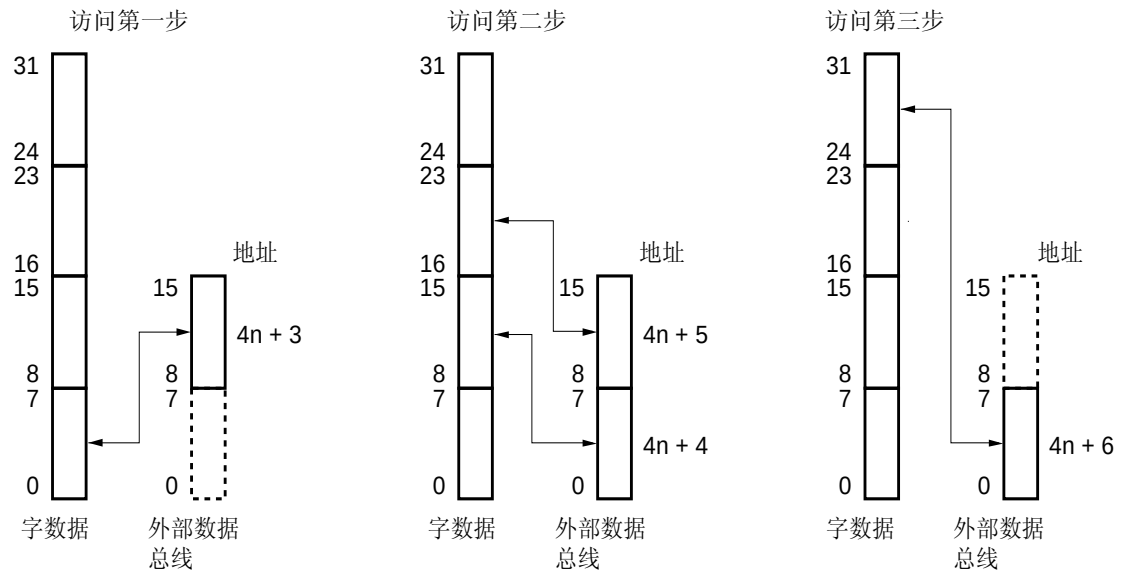


(4) 字访问 (32 位)

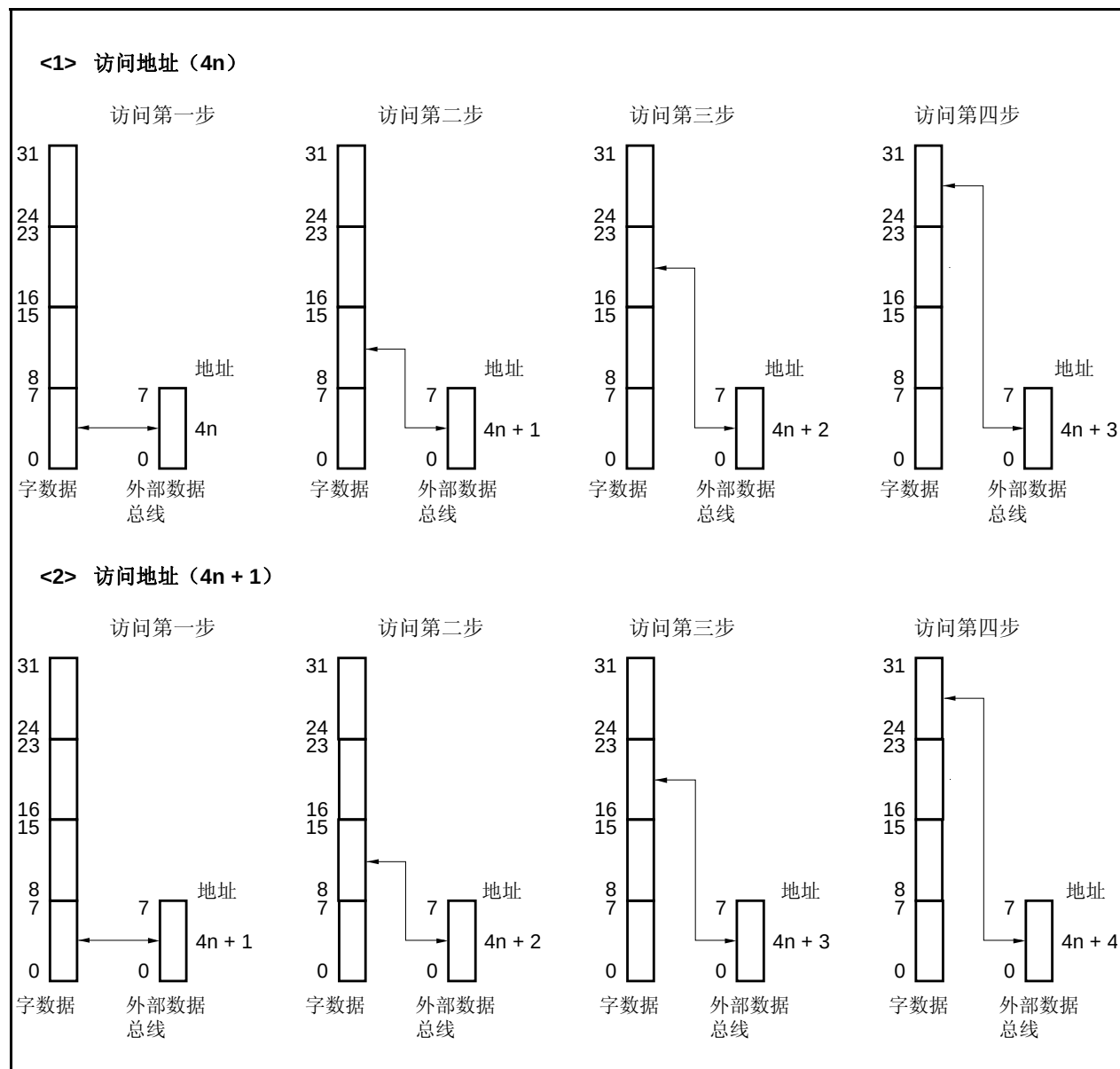
(a) 16 位数据总线宽度 (1/2)

<1> 访问地址 ($4n$)<2> 访问地址 ($4n + 1$)

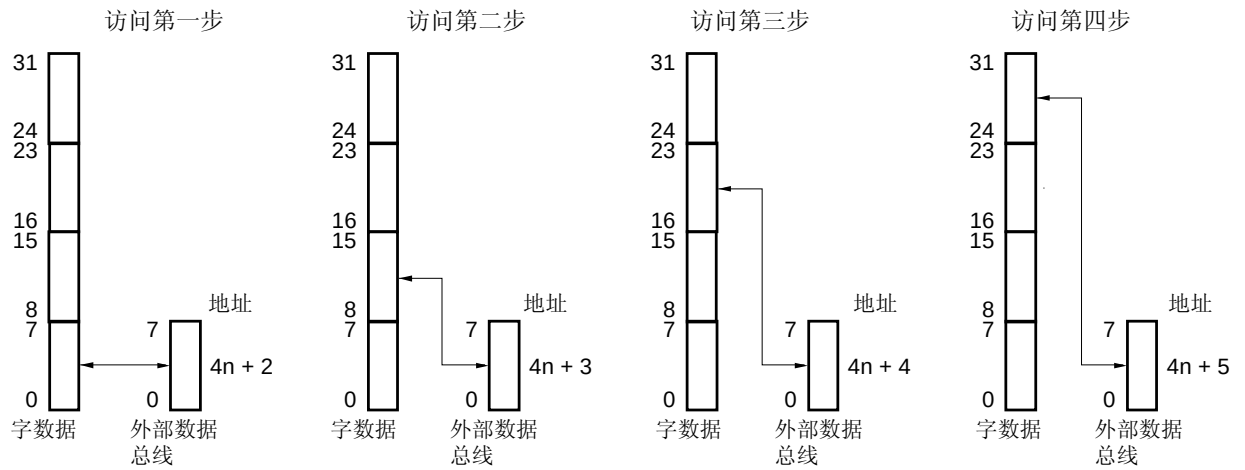
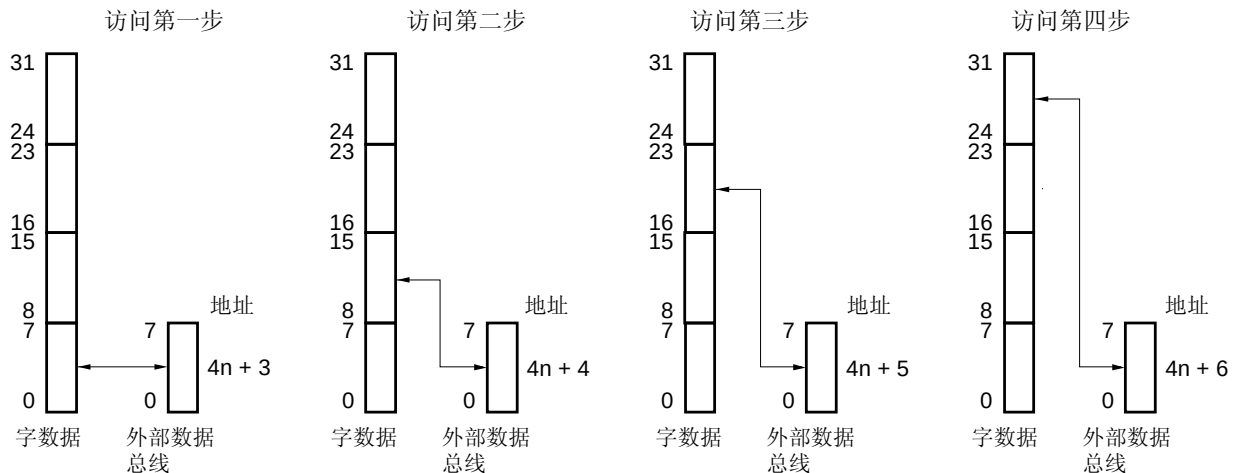
(a) 16 位数据总线宽度 (2/2)

<3> 访问地址 ($4n + 2$)<4> 访问地址 ($4n + 3$)

(b) 8 位数据总线宽度 (1/2)



(b) 8 位数据总线宽度 (2/2)

<3> 访问地址 ($4n + 2$)<4> 访问地址 ($4n + 3$)

5.5 等待功能

5.5.1 可编程等待功能

(1) 数据等待控制寄存器 0 (DWC0)

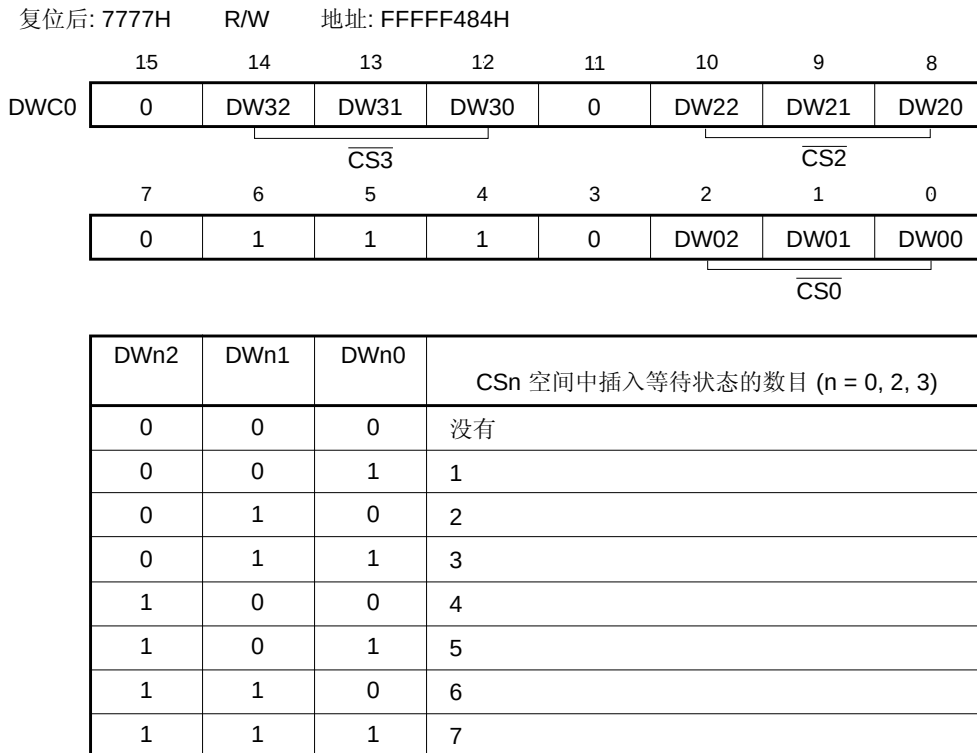
为了实现和低速存储器或者和 I/O 端口连接，执行各片选空间时，都可以在其总线周期内插入多达 7 个数据等待状态，以保证速率匹配。

等待状态的数目可使用 DWC0 寄存器编程设置。系统复位后，所有模块的总线周期都立即插入 7 个数据等待状态。

DWC0 寄存器可以按 16 位进行读写。

系统复位后，DWC0 寄存器被置为 7777H。

- 注意事项**
1. 内部 ROM 和内部 RAM 的访问不支持可编程等待，总是可以无等待访问。片上外设输入/输出端口亦不支持可编程等待，而是根据各自外设输入/输出端口的等待控制执行。
 2. 系统复位完成后写入 DWC0 寄存器，此后不要改变 DWC0 寄存器的值。而且，DWC0 寄存器初始设置完成以前，不要访问外部存储空间。



注意事项 确保将位 6 至位 4 设置为“1”并清除位 15，11 7 和位 3 为“0”。

5.5.2 外部等待功能

为了和外部低速存储器、输入/输出端口或异步设备保持同步，任意数目的等待状态都可以利用外部等待引脚（ $\overline{\text{WAIT}}$ ）插入到总线周期中。

当 $\text{P60}^{\#1}$ 或 $\text{PCM0}^{\#2}$ 引脚被设置为它的复用功能时，外部等待功能就会使能。

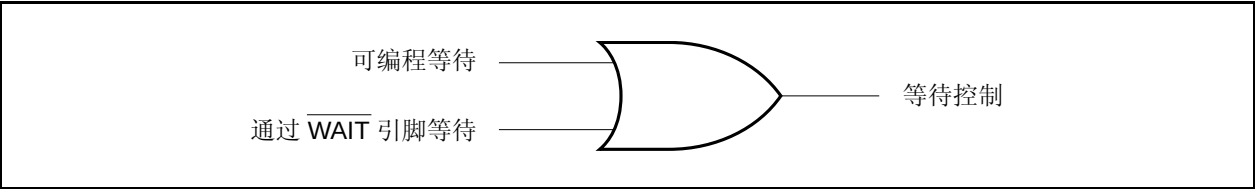
对内部 ROM、内部 RAM 的任何区域以及片上外设输入/输出端口的访问，均不受外部等待功能的控制，这一点和可编程等待功能一样。

$\overline{\text{WAIT}}$ 信号可异步输入至 CLKOUT ，复用总线模式下总线周期的 T2 和 TW 状态中，该信号在时钟的下降沿被采样。独立总线模式下，其在总线周期的 T1 和 TW 状态后立即在时钟的上升沿被采样。如果采样时序的建立/保持时间得不到满足，一个等待态被插入到下一状态，或者根本不插入。

- 注
1. V850ES/JG3-H
 2. V850ES/JH3-H

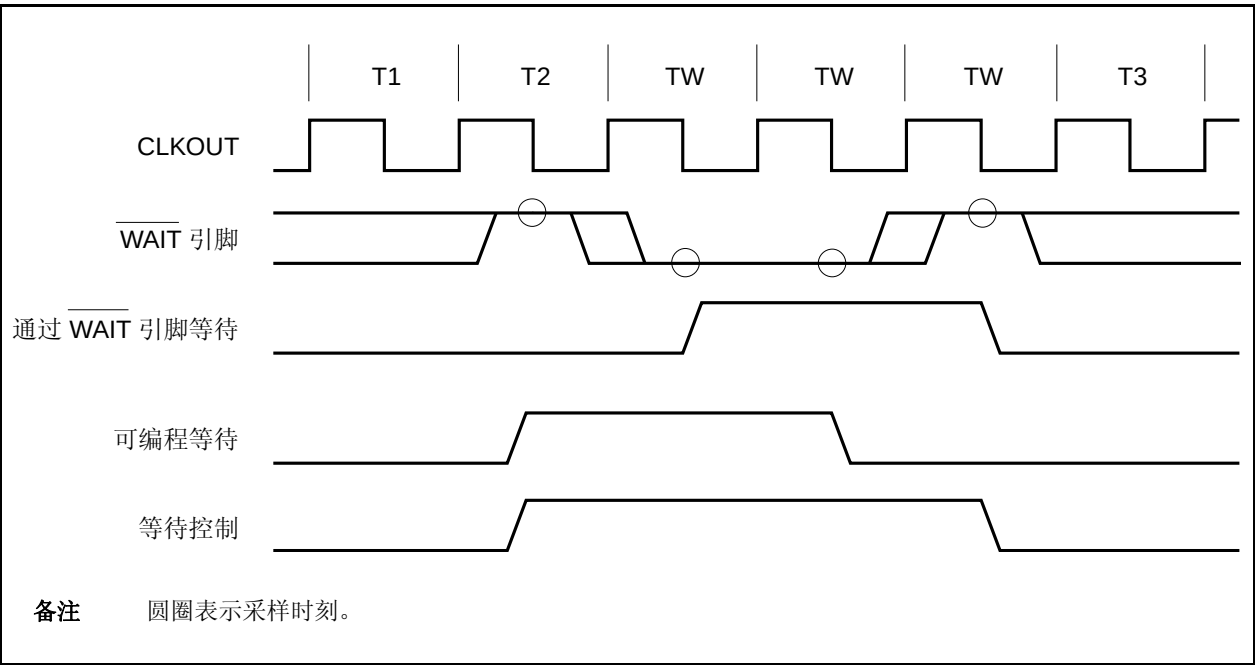
5.5.3 可编程等待和外部等待间的关系

在可编程等待的设定值确定的等待周期与由 $\overline{\text{WAIT}}$ 引脚控制的等待周期执行或操作时，其结果作为插入的等待周期。



例如，假设可编程等待时序和 $\overline{\text{WAIT}}$ 引脚信号的等待时序如下所示，则将有 3 个等待状态插入到总线访问周期中。

图 5-3.插入等待举例



5.5.4 可编程地址等待功能

利用 AWC 寄存器，可以设置插入到各个总线周期中的地址建立等待或地址保持等待。地址等待插入功能可以用于所有片选区域（CS0，CS2，CS3）。

当在总线周期中插入一个地址建立等待时，表现为总线时钟周期中 T1 状态的时钟高电平被延长了一个时钟周期。当插入一个地址保持等待时，表现为总线时钟周期中 T1 状态的时钟低电平被延长了一个时钟周期。

(1) 地址等待控制寄存器（AWC）

AWC 寄存器可以按 16 位宽度进行读写。
系统复位后 AWC 寄存器被设置为 FFFFH。

- 注意事项
- 1. 访问内部 ROM 区、内部 RAM 区和片上外围输入/输出区时，地址建立等待周期和地址保持等待周期不能插入。
 - 2. 系统复位完成后写入 AWC 寄存器，此后不要改变该设定值。另外，直到 AWC 寄存器的初始值设置完成后，才能访问外部存储器。

复位后: FFFFH R/W 地址: FFFFF488H

AWC

15	14	13	12	11	10	9	8
1	1	1	1	1	1	1	1
7	6	5	4	3	2	1	0
AHW3	ASW3	AHW2	ASW2	1	1	AHW0	ASW0
CS3		CS2		CS0			

AHWn	地址保持等待的插入说明 (n = 0, 2, 3)
0	没有插入
1	插入

ASWn	地址建立等待的插入说明 (n = 0, 2, 3)
0	没有插入
1	插入

注意事项 一定要将位 15 至 8，3 和 2 设置为“1”。

5.6 空闲状态插入功能

为便于和低速存储器连接，由片选选择的各片执行的总线周期中，T3 状态后插入一个空闲状态（T1）。通过插入空闲状态，在读取访问时，可以保证存储器数据输出的浮动延迟时间（写访问期间，不能插入空闲状态）。

利用 BCC 寄存器的编程，决定空闲状态是否被插入。

系统复位之后，一个空闲态立刻插入到所有区域的访问时序之中。

(1) 总线周期控制寄存器（BCC）

BCC 寄存器可以按 16 位宽度进行读写。

系统复位后该寄存器 被置为 AAAAH。

- 注意事项
- 1. 内部 ROM、内部 RAM、片上外围 I/O 端口 等区域的访问不支持空闲插入规则。
 - 2. 系统复位完成后写入 BCC 寄存器，此后不要改变该设定值。另外，直到 BCC 寄存器的初始值设置完成后，才能访问外部存储器。

复位后: AAAAH R/W 地址: FFFFF48AH

	15	14	13	12	11	10	9	8
BCC	1	0	1	0	1	0	1	0
	7	6	5	4	3	2	1	0
	BC31	0	BC21	0	1	0	BC01	0
	CS3		CS2				CS0	
BCCn1	空闲状态的插入说明 (n = 0, 2, 3)							
0	没有插入							
1	插入							

注意事项 务必将位 15, 13, 11, 9 和 3 设置为“1”，并且将位 14, 12, 10, 8, 6, 4, 2 和 0 清除为“0”。

5.7 总线保持功能（仅限于V850ES/JH3-H）

5.7.1 功能概述

如果 PCM2 和 PCM3 引脚被设为复用功能，则 $\overline{\text{HLDRQ}}$ 和 $\overline{\text{HLD\!AK}}$ 引脚功能有效。

当 $\overline{\text{HLDRQ}}$ 引脚有效时（低电平），表示其它总线主控方已经申请了总线控制权，这时，外部地址/数据总线进入高阻状态并被释放（总线保持状态）。如果总线控制权的请求被清除，则 $\overline{\text{HLDRQ}}$ 引脚无效（为高低电平时），引脚驱动工作又一次开始。

在总线保持期间，执行内部 ROM 和内部 RAM 中的程序，直到访问片上外围输入/输出寄存器或访问外部存储器时为止。

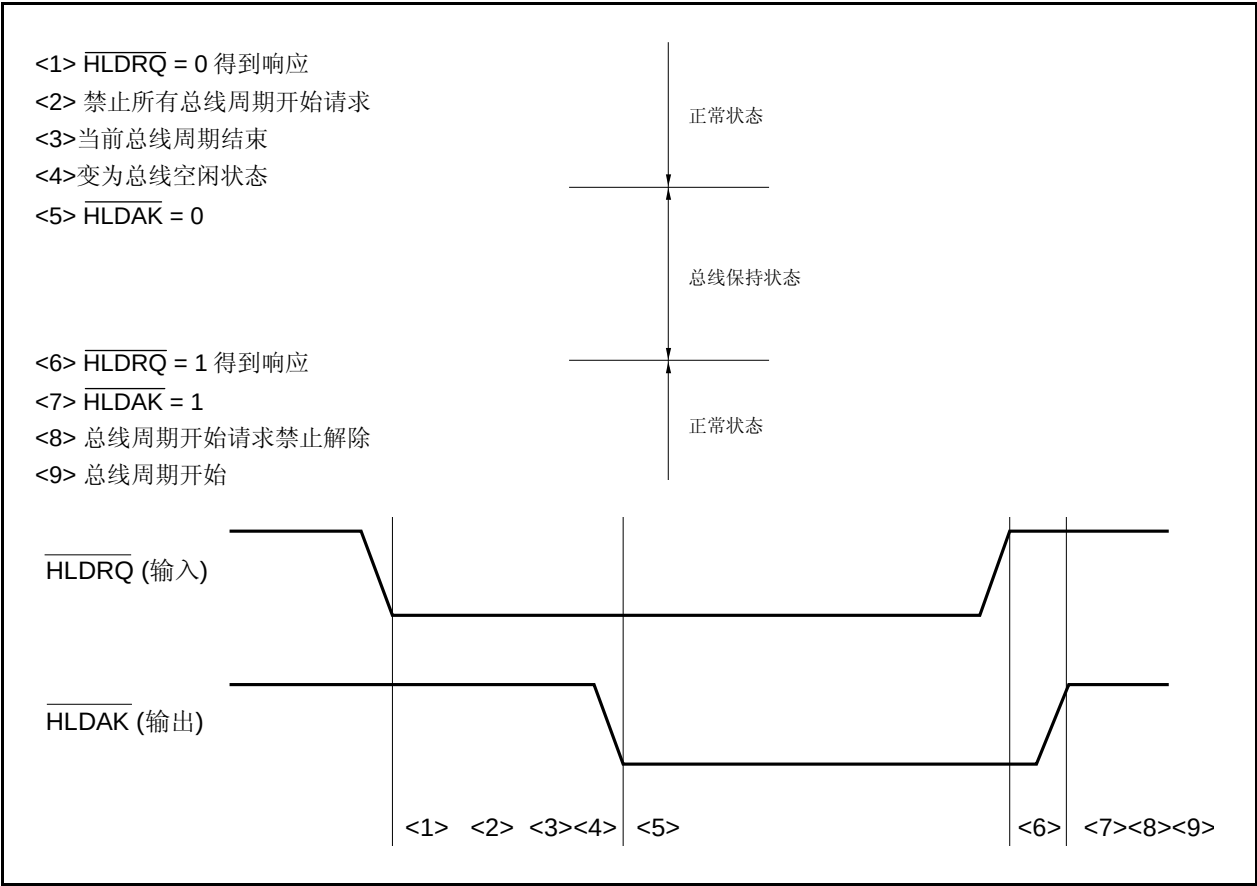
总线保持状态通过 $\overline{\text{HLD\!AK}}$ 引脚的有效（低电平）来表示。总线控制功能允许存在两个或更多个总线主控的多处理器类型系统的配置。

请注意，在由总线宽度功能或位操作指令发起的多重访问循环中，总线保持请求将得不到响应。

状态	数据总线宽度	访问类型	总线保持请求无响应的时间
CPU 总线锁定	16 位	偶地址的字访问	第一和第二次访问之间
		奇地址的字访问	第一和第二次访问之间
			第二和第三次访问之间
		奇地址的半字访问	第一和第二次访问之间
	8 位	字访问	第一和第二次访问之间
			第二和第三次访问之间
			第三和第四次访问之间
		半字访问	第一和第二次访问之间
位操作指令的读-修改-写访问	—	—	读访问和写访问之间

5.7.2 总线保持步骤

总线保持状态转变步骤如下所示：



5.7.3 节电模式时的操作

由于在 STOP、IDLE1 和 IDLE2 模式下内部系统时钟停止工作，所以即使 $\overline{\text{HLDRQ}}$ 引脚有效（为低电平）时也不会进入总线保持状态。

HALT 模式下， $\overline{\text{HLD\textbf{A}K}}$ 引脚和 $\overline{\text{HLDRQ}}$ 引脚同时变为低电平（即有效状态），并且进入总线保持状态。当 $\overline{\text{HLDRQ}}$ 引脚变为无效状态后， $\overline{\text{HLD\textbf{A}K}}$ 引脚也变为高电平（即无效状态），并退出总线保持状态。

5.8 总线优先权

总线保持、DMA 传输、运算对象数据访问、取指令（分支）、取指令（连续）等操作在外部总线周期中执行。总线保持具有最高的优先级，接着是 DMA 传输、运算对象数据访问、取指令（分支跳转）、取指令（连续取值）。在读-修改-写访问中的读访问和写访问之间可插入一个取指令操作。

如果执行一条指令需要进行两次或两次以上的数据访问时，由于总线长度的限制，则在每两次数据访问期间不插入取指令操作和总线保持状态。

表 5-5. 总线优先级

优先级	外部总线周期	总线主控器
高 ↑ ↓ 低	总线保持	外部设备
	DMA 传输	DMAC
	操作数访问	CPU
	取指周期（分支跳转）	CPU
	取指周期（顺序取指）	CPU

5.9 总线时序

图 5-4. 复用/独立总线读时序（总线宽度：16 位，16 位访问）

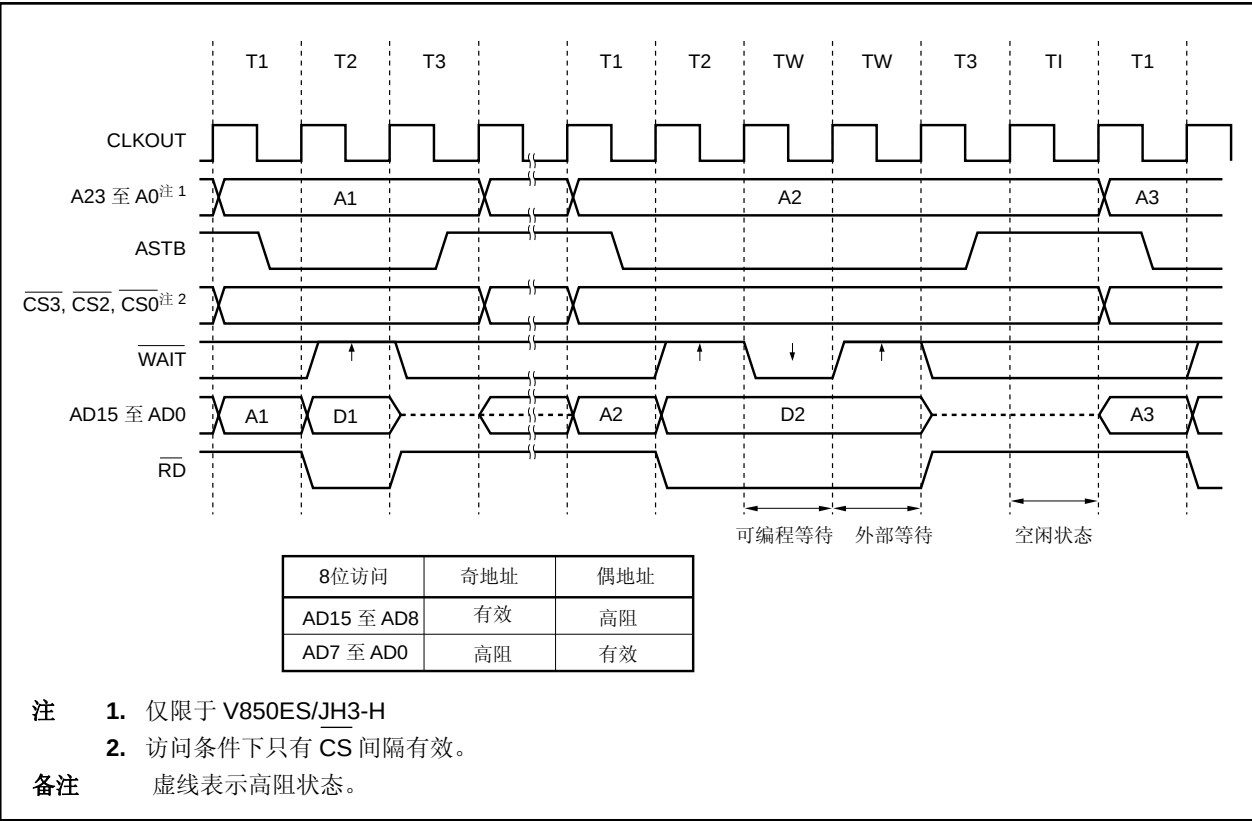


图 5-5. 复用/独立总线读时序（总线宽度：8 位）

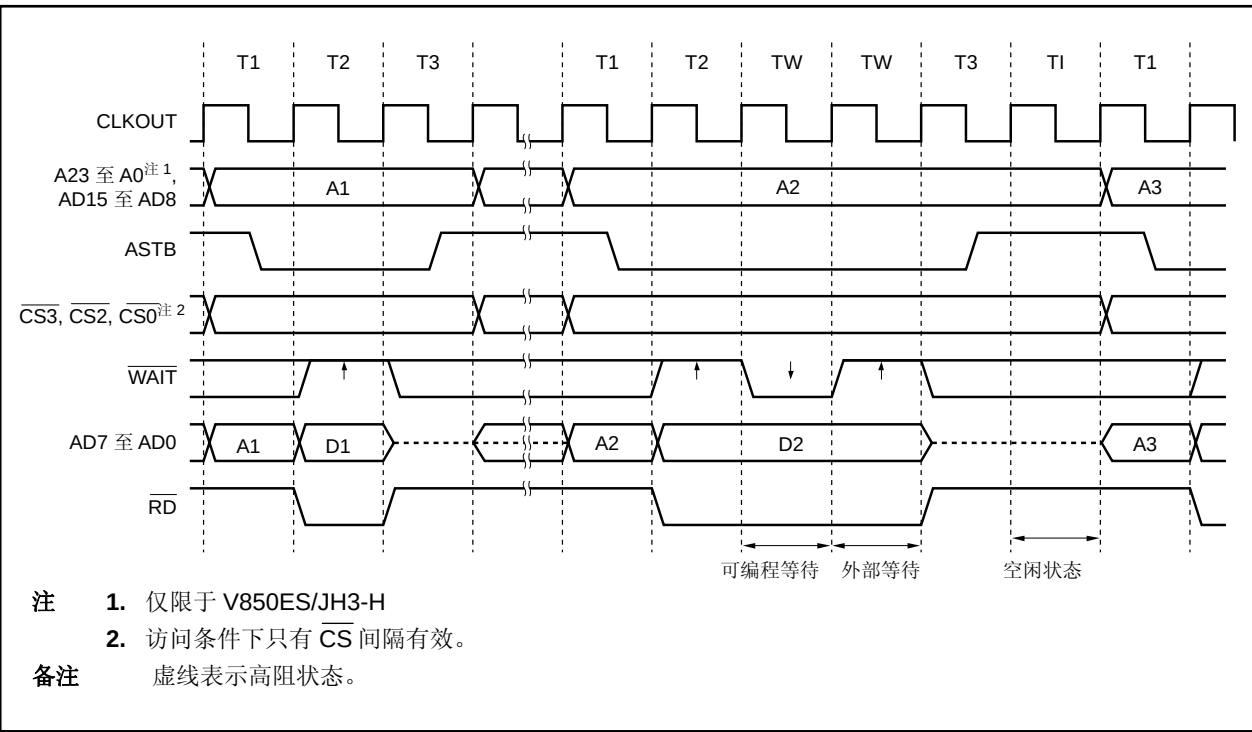


图 5-6. 复用/独立总线写时序（总线宽度：16 位，16 位访问）

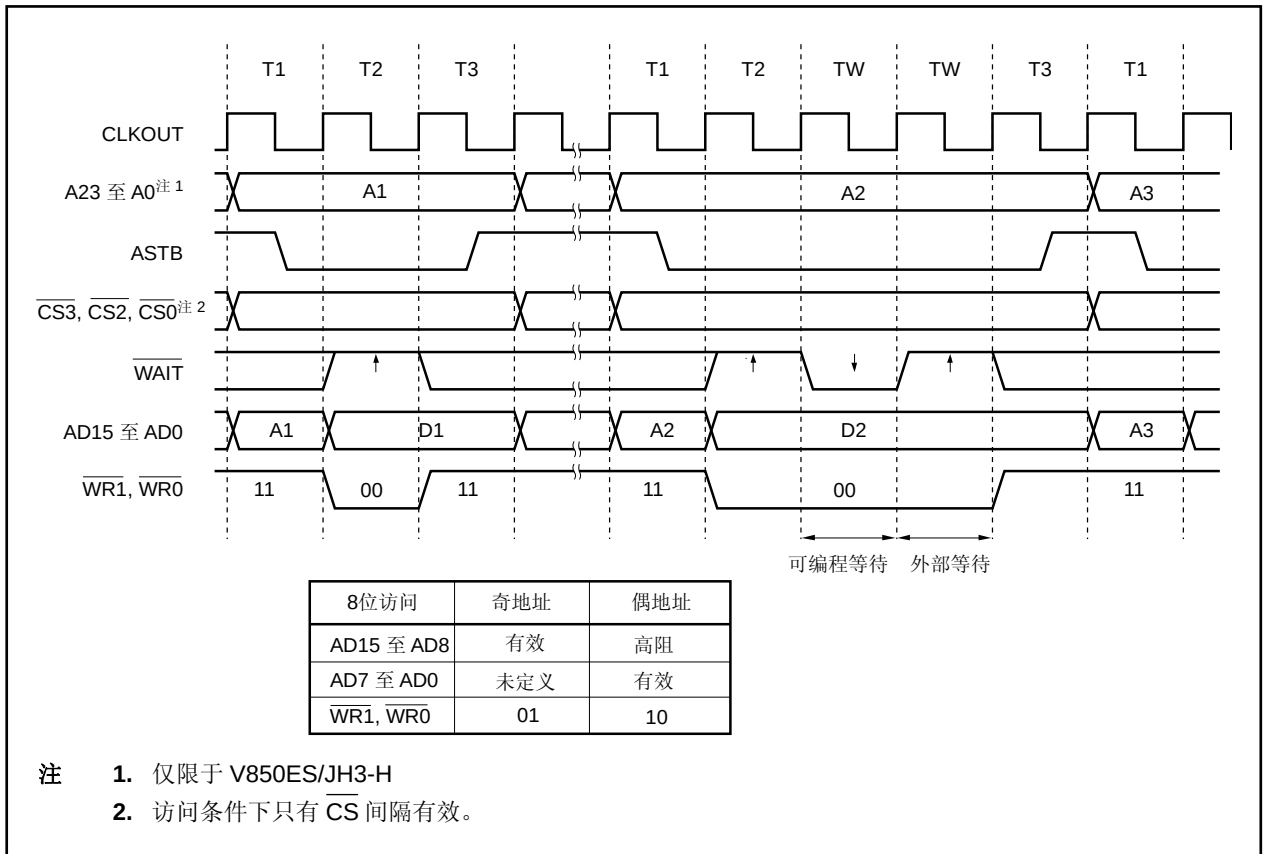


图 5-7. 复用/独立总线写时序（总线宽度：8 位）

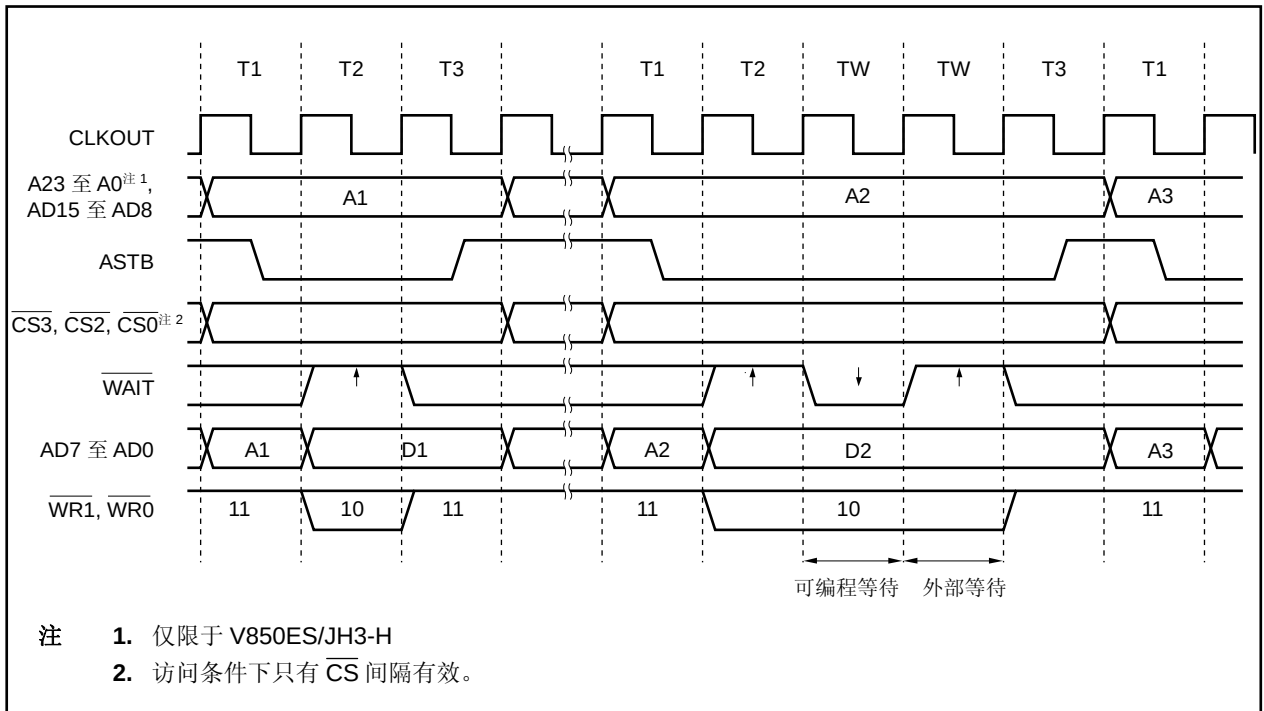
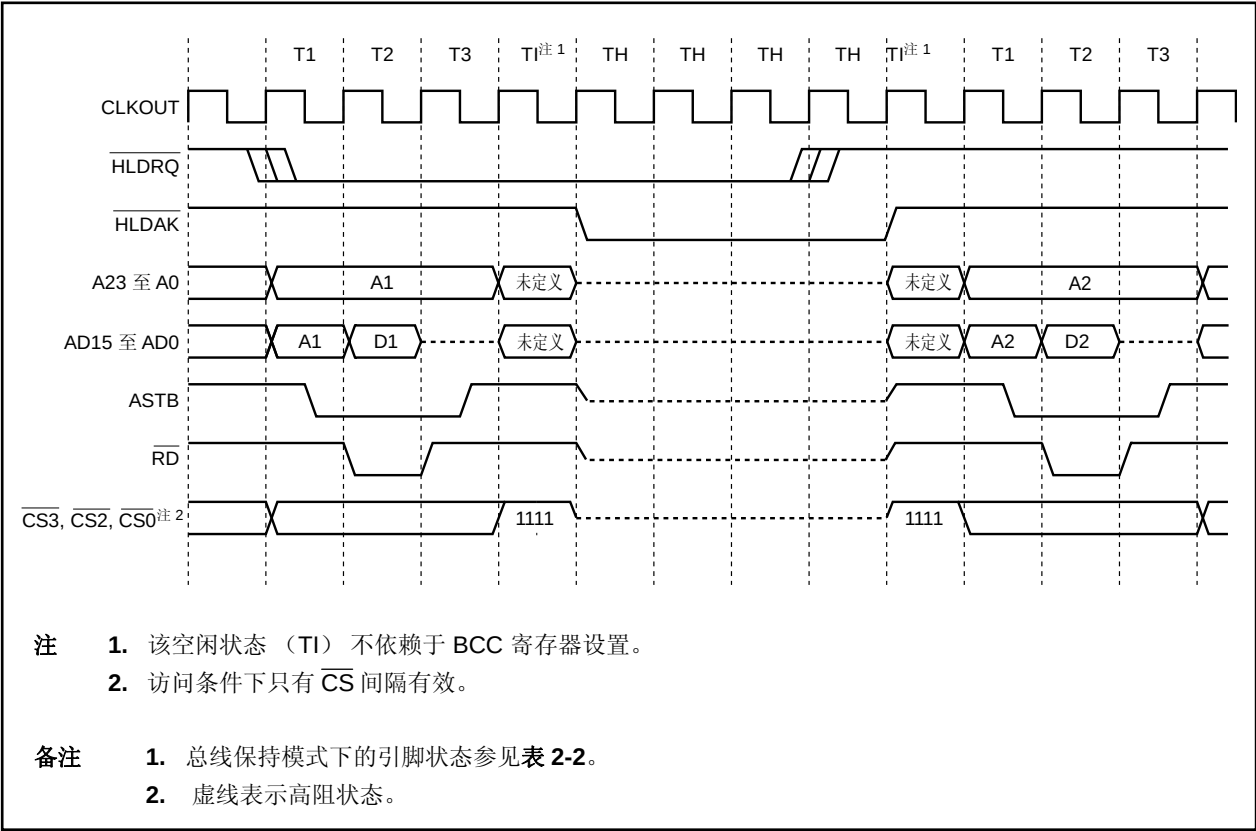


图 5-8. 复用/独立总线保持时序（总线宽度：16 位，16 位访问）（仅限于 V850ES/JH3-H）



6.1 概述

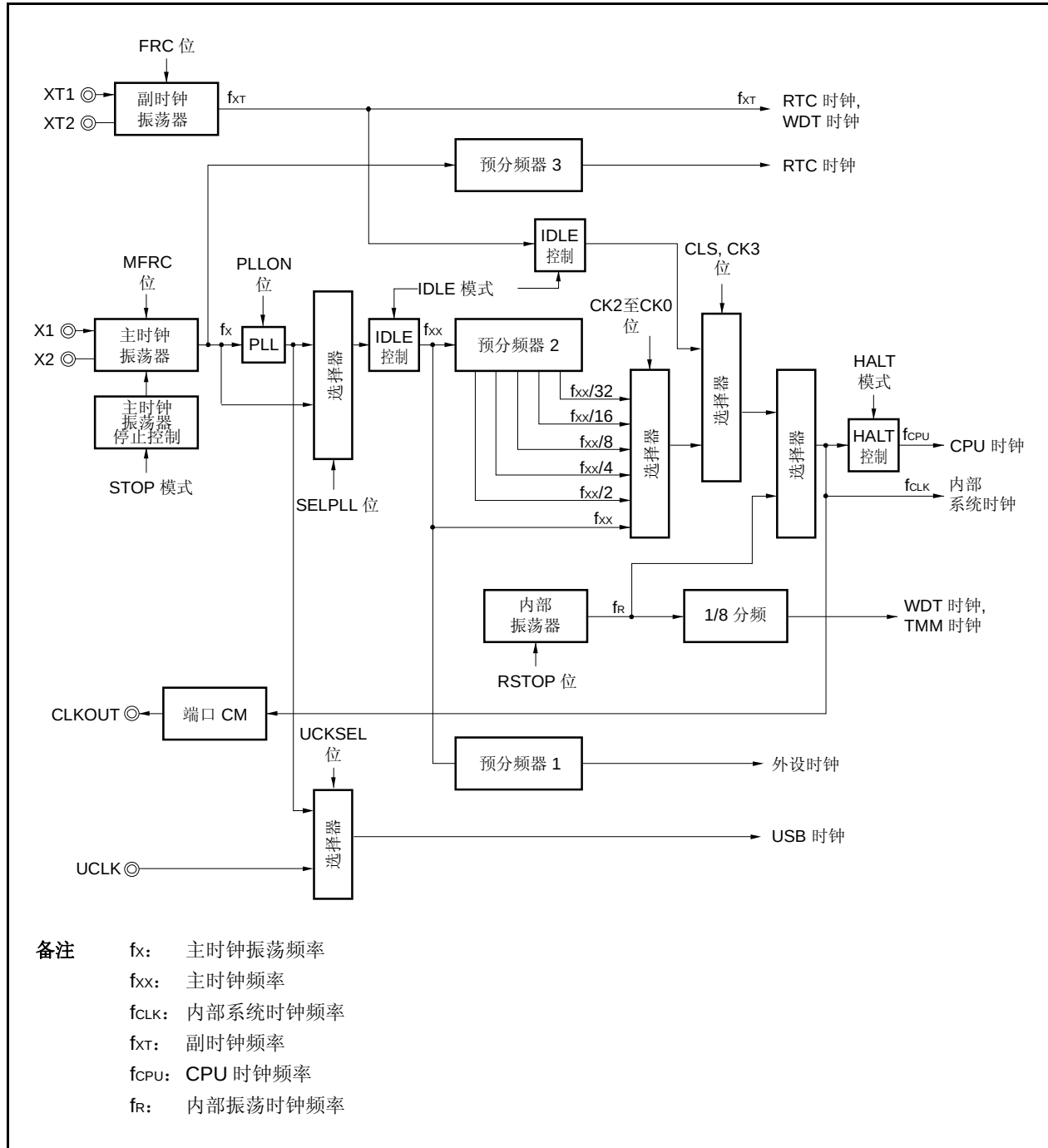
具备以下时钟产生功能：

- 主时钟振荡器
 - 时钟直通模式
 $f_x = 3.0$ 至 6.0 MHz ($f_{xx} = 3.0$ 至 6.0 MHz)
 - PLL 模式
 $f_x = 3.0$ 至 6.0 MHz ($\times 8$: $f_{xx} = 24$ 至 48 MHz)
- 副时钟振荡器
 - $f_{xt} = 32.768$ kHz
- PLL（锁相环）倍频（ $\times 8$ ）功能
 - 时钟直通模式/PLL 模式选择。
- 内部振荡器
 - $f_R = 220$ kHz（TYP.）
- 内部系统时钟产生
 - 7 等级（ f_{xx} , $f_{xx}/2$, $f_{xx}/4$, $f_{xx}/8$, $f_{xx}/16$, $f_{xx}/32$, f_{xt} ）
- 外围时钟产生功能。
- 时钟输出功能

备注	f_x :	主时钟振荡频率
	f_{xx} :	主时钟频率
	f_{xt} :	副时钟频率
	f_R :	内部振荡时钟频率

6.2 配置

图 6-1. 时钟发生器



(1) 主时钟振荡器

主时钟振荡器依照以下频率振荡 (f_x) :

- 时钟直通模式
 $f_x = 3.0$ 至 6.0 MHz
- PLL 模式
 $f_x = 3.0$ 至 6.0 MHz ($\times 8$)

(2) 副时钟振荡器

副时钟振荡器振荡于 32.768 kHz (f_{XT}) 频率。

(3) 主时钟振荡器停止控制

该电路产生一个控制信号，可以使主时钟振荡器停止工作。

系统处于停止模式时，主时钟振荡器停止工作，或者当 `PCC.MCK` 位 = 1 时（只有在 `PCC.CLS` 位 = 1 时有效），主时钟振荡器停止工作。

(4) 内部振荡器

以 220 kHz（典型值）频率振荡。

(5) 预分频器 1

该预分频器产生时钟， f_{xx} 至 $f_{xx} / 1, 024$ ），用来供给以下片上外设工作：TAA，TAB，TMM，TMT，CSIF，UARTC，I²C，CAN，ADC，DAC，WDT2

(6) 预分频器 2

本预分频器对主时钟 (f_{xx}) 进行分频。

预分频器 2 产生的时钟 (f_{xx} 至 $f_{xx} / 32$) 提供给一个选择器，产生 CPU 时钟 (f_{CPU}) 和内部系统时钟 (f_{CLK})。

f_{CLK} 时钟供给 INTC，ROM 以及 RAM 模块，还可以从 CLKOUT 引脚输出。

(7) 预分频器 3

该电路通过对主时钟振荡器产生的时钟 (f_x) 进行分频，得到特定频率 (32.768 kHz) 的时钟，并将该时钟供给实时计数器 (RTC) 模块。

(8) PLL

本电路对主时钟振荡器产生的时钟 (f_x) 进行 8 倍频。

有两种操作模式：时钟直通模式， f_x 直接输出；锁相环模式，倍频时钟输出。可以使用 `PLLCTL.SELPLL` 位设置，选择不同的工作模式。

6.3 寄存器

(1) 处理器时钟控制寄存器 (PCC)

处理器时钟控制寄存器 PCC 是一个特殊寄存器，数据只能以特定的序列写入。只有通过特定的流程才能将数据写入该寄存器（见 **3.4.8 特殊寄存器**）。

本寄存器可按 8 位或 1 位读写。

系统复位后该寄存器被置为 03H。

复位后: 03H R/W 地址: FFFFF828H

7

<6>

5

<4>

<3>

2

1

0

PCC

FRC

MCK

MFRC

CLS^注

CK3

CK2

CK1

CK0

FRC

副时钟片上反馈电阻的使用

0

使用

1

未使用

MCK

主时钟振荡器控制

0

允许振荡

1

停止振荡

• 系统工作在主时钟为CPU时钟时，即使 MCK 位设置为 (1)，主时钟的操作也不会停止。 CPU 时钟变成副时钟后，主时钟停止。

• 将 MCK 位进行从0到1的设置之前，停止使用主时钟进行操作的 片上外围功能。

• 主时钟停止且器件工作于副时钟时，清除 MCK位为(0)，并且在转换CPU时钟为主时钟或操作片上外围功能之前，通过软件确保振荡稳定时间。

MFRC

主时钟片上反馈电阻的使用

0

使用

1

不使用

CLS^注

CPU 时钟的状态 (f_{cpu})

0

主时钟操作

1

副时钟操作

CK3

CK2

CK1

CK0

时钟选择 (f_{CLK}/f_{cpu})

0

0

0

0

f_{xx}

0

0

0

1

f_{xx}/2

0

0

1

0

f_{xx}/4

0

0

1

1

f_{xx}/8

0

1

0

0

f_{xx}/16

0

1

0

1

f_{xx}/32

0

1

1

x

禁止设置

1

x

x

x

f_{XT}

注 CLS 为只读位。

- 注意事项
1. 当 CLKOUT 正在输出时，不要改变 CPU 时钟（使用 CK3 至 CK0 位）。
 2. 使用位处理指令来操作 CK3 位。当使用一个 8 位处理指令时，不要改变 CK2 至 CK0 位的设定值。

备注 x: 不需理会

(a) 设置主时钟运行→副时钟运行的举例：

- <1> CK3 位 ← 1: 建议使用位处理指令。不要改变 CK2 至 CK0 位。
- <2> 副时钟操作: 读取 CLS 检查副时钟是否开始工作。从 CK3 位设置后直到副时钟工作需要以下时间：
最大值：1/f_{XT}（1/副时钟频率）。
- <3> MCK 位 ← 1: 只有在主时钟停止时，才可设定 MCK 位为 1。

注意事项 1. 当停止主时钟时，停止锁相环工作。同样停止使用主时钟的片上外围功能的工作。
2. 如果不能满足以下条件时，调整 CK2 至 CK0 位的值，以满足条件，然后进入副时钟工作模式。

$$\text{内部系统时钟 (fCLK)} > \text{副时钟 (f}_{\text{XT}}: 32.768 \text{ kHz}) \times 4$$

备注 内部系统时钟 (fCLK): 通过设置 CK2 至 CK0 位，由主时钟 (f_{XX}) 产生。

[说明示例]

```

_DMA_DISABLE:
    clrl          0, DCHCn[r0]          -- DMA 运行禁止。n = 0 至 3
<1> _SET_SUB_RUN :
    st.b          r0, PRCMD[r0]
    set1          3, PCC[r0]            -- CK3 位 ← 1
<2> _CHECK_CLS :
    tst1          4, PCC[r0]            -- 等到副时钟开始工作为止。
    bz            _CHECK_CLS
<3> _STOP_MAIN_CLOCK :
    st.b          r0, PRCMD[r0]
    set1          6, PCC[r0]            -- MCK 位 ← 1，主时钟停止。
_DMA_ENABLE:
    setl          0, DCHCn[r0]          -- DMA 运行使能。n = 0 至 3

```

备注 以上描述是一个简单举例。注意<2>部分，在一个闭环内读取 CLS 位。

(b) 设置副时钟工作→主时钟工作的举例：

- <1> MCK 位 $\leftarrow$ 0: 主时钟振荡器开始工作。
- <2> 在程序中插入等待指令，一直等到主时钟振荡器稳定为止。
- <3> CK3 位 $\leftarrow$ 0: 建议使用位处理指令。不要改变 CK2 至 CK0 位。
- <4> 主时钟工作: 从 CK3 位置位到主时钟开始工作需要以下时间：
 最大值： $1/f_{xt}$ （1/副时钟频率）。
 所以，CK3 置位为 0 后，应紧接着插入一条 NOP 指令，或通过读取 CLS 位
 确定主时钟是否开始工作。

注意事项 只有主时钟振荡稳定后，才允许使用主时钟的片上外围功能运行。如果这些操作在振荡稳定时间完成前使能，则可能发生误操作。

[说明示例]

```

_DMA_DISABLE:
    clrl          0, DCHCn[r0]                -- DMA 运行禁止。n = 0 至 3
<1> _START_MAIN_OSC :
    st.b          r0, PRCMD[r0]                -- 解除特殊寄存器的保护。
    clr1          6, PCC[r0]                  -- 主时钟开始振荡。
<2> movea        0x55, r0, r11                -- 等待振荡器稳定时间。
    _WAIT_OST :
    nop
    nop
    nop
    addi          -1, r11, r11
    cmp           r0, r11
    bne           _WAIT_OST
<3> st.b          r0, PRCMD[r0]
    clr1          3, PCC[r0]                  -- CK3  $\leftarrow$  0
<4> _CHECK_CLS :
    tstl          4, PCC[r0]                  --等待直到主时钟开始工作为止。
    bnz           _CHECK_CLS
_DMA_ENABLE:
    setl          0, DCHCn[r0]                -- DMA 运行允许。n = 0 至 3

```

备注 以上描述是一个简单举例。注意<4>部分，在一个闭环内读取 CLS 位。

(2) 内部振荡模式寄存器 (RCM)

RCM 寄存器是 8 位寄存器，用于设置内部振荡器的工作模式。
本寄存器可按 8 位或 1 位读写。
复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: FFFFF80CH

	7	6	5	4	3	2	1	<0>
RCM	0	0	0	0	0	0	0	RSTOP

RSTOP	内部振荡器振荡/停止
0	内部振荡器振荡
1	内部振荡器停止

- 注意事项
1. CPU 工作于内部振荡时钟时 (CCLS.CCLSF 位 = 1)，不能停止内部振荡器。不要将 RSTOP 位设置为 1。
 2. 如果 CCLS.CCLSF 位设置为 1 (振荡稳定期间发生看门狗定时器溢出时)，即使 RSTOP 位设置为 1 时，内部振荡器会继续振荡。同时，RSTOP 位保持设定值 1。

(3) CPU 工作时钟状态寄存器 (CCLS)

CCLS 寄存器标明 CPU 工作时钟的状态。
本寄存器为只读寄存器，可以按 8 位或按位读出。
复位后本寄存器被设置为 00H。

复位后: 00H^注 R 地址: FFFFF82EH

	7	6	5	4	3	2	1	0
CCLS	0	0	0	0	0	0	0	CCLSF

CCLSF	CPU 工作时钟状态
0	工作于主时钟 (f _x) 或副时钟 (f _{xT})。
1	工作于内部振荡时钟 (f _R)。

注 复位释放后，振荡稳定期间如果发生看门狗定时器溢出，CCLSF 设置为 1 且复位值为 01H。

6.4 操作

6.4.1 各时钟的操作

下表显示各个时钟的操作状态：

表 6-1. 各时钟的操作状态

寄存器设置和 操作状态 目标时钟	PCC 寄存器								
	CLK 位 = 0, MCK 位 = 0					CLS 位 = 1, MCK 位 = 0		CLS 位 = 1, MCK 位 = 1	
	复位期间	振荡稳定定 时器计数期 间	HALT 模 式	IDLE1, IDLE2 模 式	STOP 模 式	副时钟 模 式	次级空闲 模式	副时钟 模 式	次级空闲 模式
主时钟振荡器 (fx)	×	○	○	○	×	○	○	×	×
副时钟振荡器 (fxT)	○	○	○	○	○	○	○	○	○
CPU 时钟 (fCPU)	×	×	×	×	×	○	×	○	×
系统内部时钟 (fCLK)	×	×	○	×	×	○	×	○	×
主时钟 (PLL 模式, fxx)	×	○ ^注	○	×	×	○	○	×	×
外设时钟 (fxx 至 fxx/1, 024)	×	×	○	×	×	○	×	×	×
实时时间时钟 (主时钟源)	×	○	○	○	×	○	○	×	×
实时时间时钟 (副时钟源)	○	○	○	○	○	○	○	○	○
WDT2 时钟 (内部振荡)	×	○	○	○	○	○	○	○	○
看门狗 2 时钟 (主时钟源)	×	×	○	×	×	○	×	×	×
看门狗 2 时钟 (副时钟源)	○	○	○	○	○	○	○	○	○

注 锁定时间

备注 ○: 可运行
×: 停止

6.4.2 时钟输出功能

时钟输出功能从 CLKOUT 引脚输出内部系统时钟 (fCLK)。

使用 PCC.CK3 至 PCC.CK0 位的设置选择系统内部时钟 (fCLK)。

CLKOUT 引脚功能可由端口 CM 控制寄存器选择，选择复用为 PCM1 引脚和时钟输出引脚。

CLKOUT 引脚的状态与表 6-1 中的内部系统时钟相同，此引脚在可工作状态下输出时钟。在停止状态下，它输出低电平。然而，复位后 CLKOUT 引脚处于端口模式 (PCM1 引脚：输入模式) 一直到该引脚被设置为输出模式。因此，该引脚为高阻状态。

6.5 PLL功能

6.5.1 概述

V850ES/JG3-H 和 V850ES/JH3-H 单片机中，可以为 CPU 和片上外设选择由 PLL 功能或时钟直通模式输出的 8 倍于振荡频率的工作时钟。

当使用锁相环功能时 (×8)： 输入时钟 = 3.0 至 6.0 MHz (输出： 24 至 48 MHz)
当选择时钟直通时： 输入时钟 = 3.0 至 6.0 MHz (输出： 3.0 至 6.0 MHz)

6.5.2 寄存器

(1) 锁相环控制寄存器 (PLLCTL)

PLLCTL 寄存器是控制 PLL 功能的 8 位寄存器。
本寄存器可按 8 位或 1 位读写。
复位后本寄存器被设置为 01H。

复位后: 01H R/W 地址: FFFFF82CH							
PLLCTL	7	6	5	4	3	2	<1> <0>
	0	0	0	0	0	0	SELPLL PLLON
PLLON		PLL 工作停止寄存器					
0		PLL 停止					
1		PLL 工作 (PLL 开始工作后，需要一段锁存时间达到频率稳定。)					
SELPLL		CPU 工作时钟选择寄存器					
0		时钟直通模式					
1		PLL 模式					

- 注意事项 1. PLLON 位清除为 0 时，SELPLL 位自动清除为 0（时钟直通模式）。
2. 只有在 PLL 时钟频率稳定后，SELPLL 位才能够被设置为 1。如果不是这样（未锁存），若向其写入数据，则写入 SELPLL 位将会是"0"。

(2) 时钟控制寄存器 (CKC)

CKC 寄存器是一个特殊寄存器。只有通过特定的流程才能将数据写入该寄存器（见 3.4.8 特殊寄存器）。

CKC 寄存器控制 PLL 模式下的内部系统时钟。

本寄存器可按 8 位或 1 位读写。

系统复位后该寄存器被设为 0AH。

复位后: 0AH R/W 地址: FFFFF822H

	7	6	5	4	3	2	1	0
CKC	0	0	0	0	1	0	1	CKDIV0

CKDIV0	PLL 模式下内部系统时钟(f_{xx})
0	禁止设置
1	$f_{xx} = 8 \times f_x$ ($f_x = 3.0$ 至 6.0 MHz)

- 注意事项
- 1. 确保将 CKC 寄存器设置为 0BH。当将该寄存器设置为一个非 0BH 的值或者保留其初始值而不将其设置为 0BH 时，禁止使能 PLL 运行（PLLCTL.SELPLL = 1）。
 - 2. 确保将位 3 和位 1 设置为“1”并清除位 7 至 4 和位 2 为“0”。

备注 CPU 时钟和外围时钟都被 CKC 寄存器分频，但只有 CPU 时钟被 PCC 寄存器分频。

(3) 锁存寄存器 (LOCKR)

相位锁存以给定频率发生在上电后或 STOP 模式刚被释放后，需要稳定的时间是锁存时间（频率稳定时间）。一直到稳定之前的状态称为锁定中状态，稳定后状态称为已锁定状态。

LOCKR 寄存器包括反映 PLL 频率稳定状态的 LOCK 位。

本寄存器为只读寄存器，可以按 8 位或按位读出。

复位后本寄存器被设置为 00H。

复位后: 00H R 地址: FFFFF824H

	7	6	5	4	3	2	1	<0>
LOCKR	0	0	0	0	0	0	0	LOCK

LOCK	PLL 锁定状态检查
0	锁定状态
1	未锁定状态

(4) PLL 锁定时间规范寄存器 (PLLS)

PLLS 寄存器是 8 位寄存器，用于选择 PLLCTL.PLLON 位由 0 变到 1 时 PLL 的锁定时间。

该寄存器可按 8 位为单位进行读写。

系统复位后该寄存器被置为 03H。

复位后: 03H R/W 地址: FFFFF6C1H

	7	6	5	4	3	2	1	0
PLLS	0	0	0	0	0	0	PLLS1	PLLS0

PLLS1	PLLS0	PLL 锁定时间的选择
0	0	$2^{10}/f_x$
0	1	$2^{11}/f_x$
1	0	$2^{12}/f_x$
1	1	$2^{13}/f_x$ (默认值)

- 注意事项**
1. 如此设置锁定时间为 800 μ s 或更长。
 2. 锁定期间不要改变 PLLS 寄存器的设置。

6.5.3 用途**(1) 当使用锁相环时**

- 复位信号释放后，锁相环开始工作（PLLCTL.PLLON 位= 1），但是，由于缺省模式为时钟直通模式（PLLCTL.SELPLL 位= 0），应选择锁相环模式（SELPLL 位= 1）。
- 若要允许锁相环工作，首先将 PLLON 位设为 1，然后在 LOCKR.LOCK 位= 0 之后设置 SELPLL 位为 1。若要停止锁相环工作，首先设置为时钟直通模式（SELPLL 位= 0），等待 8 个时钟周期或更长后，再停止锁相环工作（PLLON 位= 0）。
- 无论设置如何，在转变为 IDLE2 或 STOP 模式和从 IDLE2 或 STOP 模式恢复至转变前的状态期间，PLL 工作停止。恢复需要的时间如下：

(a) 从时钟直通模式转变至 IDLE2 或 STOP 模式

- STOP 模式: 设置 OSTS 寄存器使得振荡稳定时间为 1 ms（最少）或更长。
- IDLE2 模式: 设置 OSTS 寄存器使得建立时间为 350 μ s（最少）或更长。

(b) 保持在 PLL 模式下转变至 IDLE 2 或 STOP 模式

- STOP 模式: 设置 OSTS 寄存器使得振荡稳定时间为 1 ms（最少）或更长。
- IDLE2 模式: 设置 OSTS 寄存器使得建立时间为 800 μ s（最少）或更长。

转变至 IDLE1 模式时，PLL 不能停止。根据需要停止 PLL 工作。

(2) 当锁相环未使用时

- 复位信号释放后，选择时钟直通模式（SELPLL 位= 0），但是，锁相环处于工作状态（PLLON 位= 1），因此必须使其停止（PLLON 位= 0）。

第七章 16 位定时器/事件计数器 AA （TAA）

定时器 AA （TAA）是 16 位定时器/事件计数器。
V850ES/JG3-H 和 V850ES/JH3-H 有 TAA0 至 TAA5 个定时器。

7.1 概述

定时器 TAA_n 的概况如下所示：

- 时钟选择：8 种
- 捕获/触发输入引脚：2
- 外部事件计数输入引脚^注：1
- 外部触发输入引脚^注：1
- 定时器/计数器：1
- 捕获/比较寄存器：2
（通过使用 TAA0 和 TAA1，TAA2 和 TAA3 的串联，32 位捕获定时器功能可用。）
- 捕获/比较匹配中断请求信号：2
- 定时器输出引脚：2

注 外部事件计数输入引脚和外部触发输入引脚可复用为捕获/触发输入引脚（TIAAm0）。

备注 n = 0 至 5， m = 0 至 3， 5

7.2 功能

定时器 TAA_n 具有下列功能：

- 间隔定时器
- 外部事件计数器
- 外部触发脉冲输出
- 单次脉冲输出
- PWM（脉宽调制信号）输出
- 自由运行定时器
- 脉冲宽度测量
- 调谐定时器功能
- 同时启动功能

7.3 配置

TAA_n 包括以下硬件：

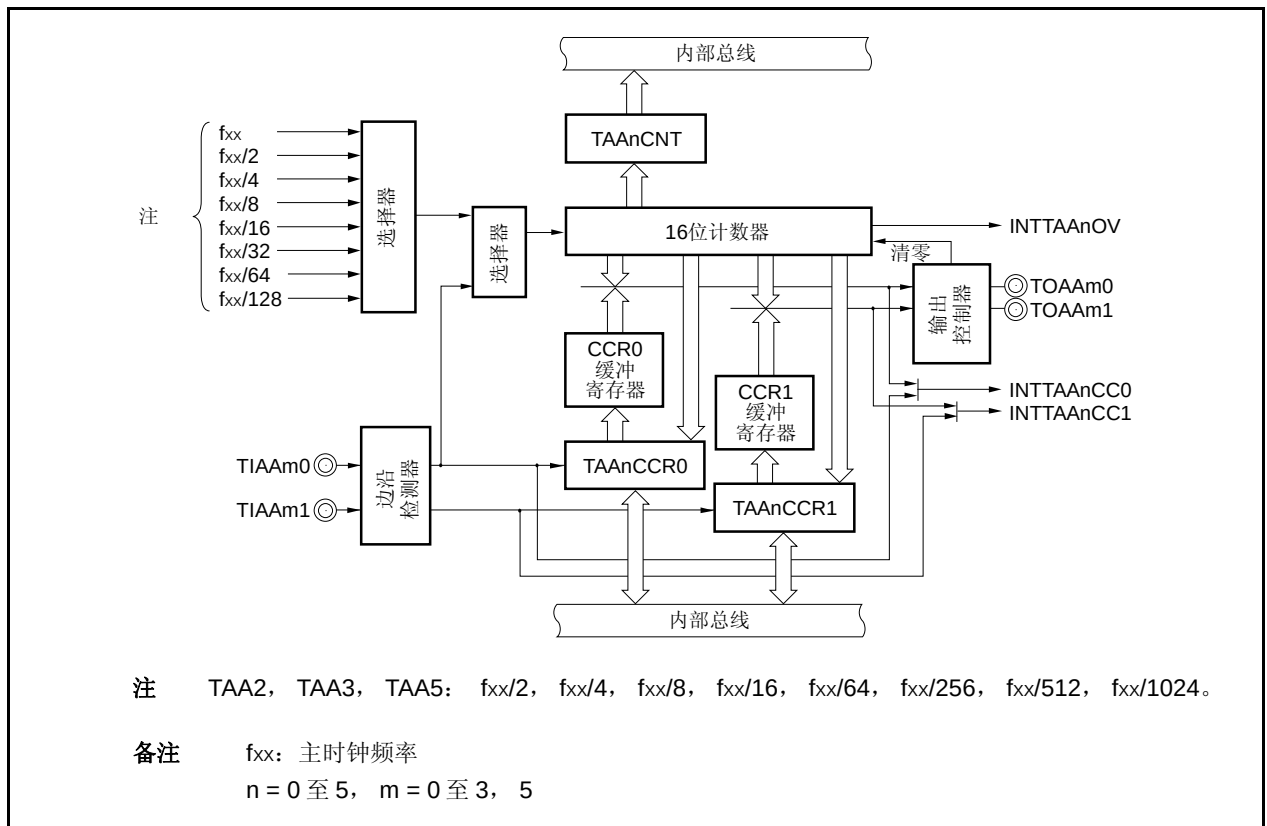
表 7-1. TAA_n 的配置

项目	配置
寄存器	16 位计数器 TAA _n 捕获/比较寄存器 0, 1 (TAA _n CCR0, TAA _n CCR1) TAA _n 计数器读取缓冲寄存器 (TAA _n CNT) CCR0, CCR1 缓冲寄存器 TAA _n 控制寄存器 0, 1 (TAA _n CTL0, TAA _n CTL1) TAA _m I/O 控制寄存器 0 至 2, 4 (TAA _m IOC0 至 TAA _m IOC2, TAA _m IOC4) TAA _m 选项寄存器 0, 1 (TAA _m OPT0, TAA _m OPT1) TAA 噪声抑制控制寄存器 (TANFC)
定时器输入 ^{注1}	2 (TIAA _m 0 ^{注2} , TIAA _m 1 引脚)
定时器输出 ^{注1}	2 (TOAA _m 0, TOAA _m 1 引脚)

- 注 1. 当使用 TIAA_m0, TIAA_m1, TOAA_m0 和 TOAA_m1 引脚的功能时, 参见表 4-20 端口引脚用作复用功能引脚。
2. TIAA_m0 引脚和捕获触发输入引脚、外部事件计数输入引脚以及外部触发输入引脚共用。

备注 n = 0 至 5, m = 0 至 3, 5

图 7-1. TAA_n 的功能框图



(1) 16 位计数器

此 16 位计数器可以给内部时钟或外部事件计数。

该计数器的计数值可以通过读取 TAAncNT 寄存器得到。

当 TAAncTL0.TAAncCE 位= 0 时, 16 位计数器的值为 FFFFH。如果此时读取寄存器 TAAncNT, 则读回的值为 0000H。

系统复位将 TAAncCE 位清 0, 因此, 16 位计数器被设置为 FFFFH。

(2) CCR0 缓冲寄存器

CCR1 缓冲寄存器为 16 位比较寄存器, 用于比较 16 位计数器的计数值。

当 TAAncCCR0 寄存器用作比较寄存器时, 写入 TAAncCCR0 寄存器的值会传送到 CCR0 缓冲寄存器。当 16 位计数器的计数值与 CCR0 缓冲寄存器的值相符时, 就会生成一个比较匹配中断请求信号 (INTTAAncCC0)。

CCR0 缓冲寄存器不能直接读写。

复位将 TAAncCCR0 寄存器清除为 0000H。因此 CCR0 缓冲寄存器被清除为 0000H。

(3) CCR1 缓冲寄存器

CCR1 缓冲寄存器为 16 位比较寄存器, 用于比较 16 位计数器的计数值。

当 TAAncCCR1 寄存器用作比较寄存器时, 写入 TAAncCCR1 寄存器的值会传送到 CCR1 缓冲寄存器。当 16 位计数器的计数值和 CCR1 缓冲寄存器的值匹配时, 就会产生一个比较匹配中断请求信号 (INTTAAncCC1)。

CCR1 缓冲寄存器不能直接读写。

复位将 TAAncCCR1 寄存器清除为 0000H。因此 CCR1 缓冲寄存器被清除为 0000H。

(4) 边沿检测器

该电路检测输入到 TIAAm0 和 TIAAm1 引脚信号的有效边沿。使用 TAAmIOC1 和 TAAmIOC2 寄存器设置, 可以选择各种有效沿: 无边沿、上升沿、下降沿、上升/下降双边沿。

(5) 输出控制器

该电路控制 TOAAm0 和 TOAAm1 引脚的输出。TOAAm0 和 TOAAm1 引脚的输出由 TAAmIOC0 寄存器控制。

(6) 选择器

此选择器选择 16 位计数器的计数时钟。可选择 8 种内部时钟或一个外部事件为计数时钟。

7.3.1 引脚配置

配置 TAA_n 的定时器的输入和输出与以下端口共享：使用各引脚时必须设置端口功能（参见表 4-20 端口引脚用作复用功能引脚）。

表 7-2. 引脚配置

通道	端口	定时器 AA 输入	定时器 AA 输出	其它复用功能
TAA0	P32	TIAA00 ^{註 1}	TOAA00	ASCK0/SCKF4
	P33	TIAA01	TOAA01	RTCDIV/RTCCL
TAA1	P34	TIAA10 ^{註 1}	TOAA10	TOAA1OFF/INTP09
	P35	TIAA11	TOAA11	RTC1HZ
TAA2	P97	TIAA20 ^{註 1}	TOAA20	SIF1/A7 ^{註 2}
	P96	TIAA21	TOAA21	INTP11/A6 ^{註 2}
TAA3	P95	TIAA30 ^{註 1}	TOAA30	A5 ^{註 2}
	P94	TIAA31	TOAA31	TENC00/EVTT0/A4 ^{註 2}
TAA4	—	—	—	—
	—	—	—	—
TAA5	P915	TIAA50 ^{註 1}	TOAA50	INTP18/A15 ^{註 2}
	P914	TIAA51	TOAA51	INTP17/A14 ^{註 2}

注 1. TAA_m0 引脚复用为捕获触发输入功能、外部事件计数输入功能以及外部触发输入功能。

2. 仅限于 V850ES/JH3-H

备注 TAA4 没有定时器输入和输出。因此，TAA4 本身只有间隔定时器功能。但是，通过结合使用 TAA4 和 TAB1，可以得到 6 相脉宽调制信号输出功能。

7.4 寄存器

以下寄存器控制 TAA_n:

- TAA_n 控制寄存器 0 (TAA_nCTL0)
- TAA_n 控制寄存器 1 (TAA_nCTL1)
- TAA_n I/O 控制寄存器 0 (TAA_mIOC0)
- TAA_n I/O 控制寄存器 1 (TAA_mIOC1)
- TAA_n I/O 控制寄存器 2 (TAA_mIOC2)
- TAA_n I/O 控制寄存器 4 (TAA_mIOC4)
- TAA_n 选项寄存器 0 (TAA_mOPT0)
- TAA_n 选项寄存器 1 (TAA_mOPT1)
- TAA_n 捕获/比较寄存器 0 (TAA_nCCR0)
- TAA_n 捕获/比较寄存器 1 (TAA_nCCR1)
- TAA_n 计数器读取缓冲寄存器 (TAA_nCNT)
- TAA 噪声抑制控制寄存器 (TANFC)

- 备注**
1. 当使用 TIAA_m0, TIAA_m1, TOAA_m0 和 TOAA_m1 引脚的功能时, 参见表 4-20 端口引脚用作复用功能引脚。
 2. n = 0 至 5, m = 0 至 3, 5

(1) TAA_n 控制寄存器 0 (TAA_nCTL0)

TAA_nCTL0 寄存器为 8 位寄存器，用于控制定时器 TAA_n 的操作。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

总是可以使用软件将相同的值写入 TAA_nCTL0 寄存器。

复位后: 00H R/W 地址: TAA0CTL0 FFFFF630H, TAA1CTL0 FFFFF640H,
TAA2CTL0 FFFFF650H, TAA3CTL0 FFFFF660H,
TAA4CTL0 FFFFF670H, TAA5CTL0 FFFFF680H

	7	6	5	4	3	2	1	0
TAA _n CTL0	TAA _n CE	0	0	0	0	TAA _n CKS2	TAA _n CKS1	TAA _n CKS0

(n = 0 至 5)

TAA _n CE	TAA _n 操作控制
0	TAA _n 操作禁止 (TAA _n 异步复位 ^注)。
1	TAA _n 操作使能。 TAA _n 操作启动。

TAA _n CKS2	TAA _n CKS1	TAA _n CKS0	内部计数时钟选择	
			n = 0, 1, 4	n = 2, 3, 5
0	0	0	f _{xx} (20.8 ns)	f _{xx} /2 (41.7 ns)
0	0	1	f _{xx} /2 (41.7 ns)	f _{xx} /4 (83.3 ns)
0	1	0	f _{xx} /4 (83.3 ns)	f _{xx} /8 (166.7 ns)
0	1	1	f _{xx} /8 (166.7 ns)	f _{xx} /16 (333.3 ns)
1	0	0	f _{xx} /16 (333.3 ns)	f _{xx} /64 (1.3333μs)
1	0	1	f _{xx} /32 (666.7 ns)	f _{xx} /256 (5.3333μs)
1	1	0	f _{xx} /64 (1.3333μs)	f _{xx} /512 (10.6667μs)
1	1	1	f _{xx} /128 (2.6667μs)	f _{xx} /1024 (21.3333μs)

注 TAA_nOPT0.TAA_nOVF 位，16 位计数器，定时器输出 (TOAAn0, TOAAn1 引脚)

- 注意事项
1. 当 TAA_nCE 位 = 0 时设置 TAA_nCKS2 至 TAA_nCKS0 位。
当 TAA_nCE 位的值从 0 改变到 1 时，可同时设置 TAA_nCKS2 至 TAA_nCKS0 位。
 2. 务必将 3 至 6 位设置为“0”。

备注 f_{xx}: 主时钟频率
圆括号内的值表示当 f_{xx} = 48 MHz 时的周期。

(2) TAA_n 控制寄存器 1 (TAA_nCTL1)

TAA_nCTL1 寄存器为 8 位寄存器，用于控制定时器 TAA_n 的操作。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

(1/2)

复位后: 00H R/W 地址: TAA0CTL1 FFFFFFF631H, TAA1CTL1 FFFFFFF641H,
TAA2CTL1 FFFFFFF651H, TAA3CTL1 FFFFFFF661H,
TAA4CTL1 FFFFFFF671H, TAA5CTL1 FFFFFFF681H

	7	6	5	4	3	2	1	0
TAA0CTL1	TAA0SYE	TAA0EST	TAA0EEE	TAA0SYM	0	TAA0MD2	TAA0MD1	TAA0MD0
TAA1CTL1	0	TAA1EST	TAA1EEE	0	0	TAA1MD2	TAA1MD1	TAA1MD0
TAA2CTL1	TAA2SYE	TAA2EST	TAA2EEE	TAA2SYM	0	TAA2MD2	TAA2MD1	TAA2MD0
TAA3CTL1	0	TAA3EST	TAA3EEE	0	0	TAA3MD2	TAA3MD1	TAA3MD0
TAA4CTL1	TAA4SYE	0	0	TAA4SYM	0	TAA4MD2	TAA4MD1	TAA4MD0
TAA5CTL1	TAA5SYE	TAA5EST	TAA5EEE	TAA5SYM	0	TAA5MD2	TAA5MD1	TAA5MD0

TAA _m SYE	TAA _m SYM	调谐操作模式使能控制 (m = 0, 2, 4, 5)
0	0	独立操作模式 (异步操作模式)
0	1	禁止设置
1	0	调谐操作功能 (从机操作的指定)
1	1	同时启动功能 (从定时器的指定)

这些位只能为从定时器设置 (禁止为主定时器设置它们)。
主定时器和从定时器的关系如下:

主定时器	从定时器
TAA1	TAA0
TAA3	TAA2
TAA5	TAA4

关于调谐操作功能, 参见 7.6 调谐定时器操作功能。

关于同时启动功能, 参见 7.7 同时启动功能。

TAA _n EST	软件触发控制 (n = 0 至 3, 5)
0	-
1	为外部触发输入产生一个有效的信号。 · 在单次脉冲输出模式: 将 1 写入 TAA _n EST 位作为触发, 输出一个单次脉冲。 · 在外部触发脉冲输出模式: 将 1 写入 TAA _n EST 位作为触发, 输出一个 PWM 波形。

注意事项 1. TAA_nEST 位只有在外触发脉冲输出模式或单次脉冲输出模式时有效。在其它模式下, 忽略向该位写入 1 的操作。

2. 确保将指定为 0 的各通道的 TAA_nCTL1 寄存器的部分清除为 0。

TAAmEEE	计数时钟的选择
0	禁止用外部事件计数输入操作。 (用TAAmCTL0.TAAmCK0 至TAAmCK2 位选择的计数时钟执行计数。)
1	允许用外部事件计数输入操作。 (在每一个外部事件计数输入信号的有效沿执行计数。)
TAAmEEE 位选择用内部计数时钟还是外部事件计数输入执行计数。	

TAAmMD2	TAAmMD1	TAAmMD0	定时器模式选择
0	0	0	间隔定时器模式
0	0	1	外部事件计数模式
0	1	0	外部触发脉冲输出模式
0	1	1	单次脉冲输出模式
1	0	0	PWM 输出模式
1	0	1	自由运行定时器模式
1	1	0	脉冲宽度测量模式
1	1	1	禁止设置

- 注意事项**
1. 不论 TAAmEEE 位的取值如何，在外部事件计数模式下，都选择外部事件输入端。
 2. 当 TAAmCTL0.TAAmCE 位 = 0 时设置 TAAmEEE 以及 TAAmMD2 至 TAAmMD0 位。(当 TAAmCE 位 = 1 时，可设置相同的值。)在 TAAmCE 位 = 1 时重写操作是无法保证的。若重写发生错误，应把 TAAmCE 位清为 0，再重新进行位设定 (m = 0 至 3, 5)。

(3) TAA_n I/O 控制寄存器 0 (TAA_nIOC0)

TAA_nIOC0 寄存器是 8 位寄存器，用于控制定时器输出 (TOAA_n0, TOAA_n1 引脚)。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: TAA0IOC0 FFFFF632H, TAA1IOC0 FFFFF642H,
TAA2IOC0 FFFFF652H, TAA3IOC0 FFFFF662H,
TAA5IOC0 FFFFF682H

	7	6	5	4	3	2	1	0
TAA _n IOC0	0	0	0	0	TAA _n OL1	TAA _n OE1	TAA _n OL0	TAA _n OE0
(n = 0 至 3, 5)								
	TAA _n OL1		TOAA _n 1 引脚输出电平设置 ^注					
	0		TOAA _n 1 引脚从高电平开始输出					
	1		TOAA _n 1 引脚从低电平开始输出					
	TAA _n OE1		TOAA _n 1 引脚输出电平设置					
	0		禁止定时器输出 · 当 TAA _n OL1 位 = 0: 从 TOAA _n 1 引脚输出低电平 · 当 TAA _n OL1 位 = 1: 从 TOAA _n 1 引脚输出高电平					
	1		允许定时器输出 (从 TOAA _n 1 引脚输出方波)。					
	TAA _n OL0		TOAA _n 0 引脚输出电平设置 ^注					
	0		TOAA _n 0 引脚从高电平开始输出					
	1		TOAA _n 0 引脚从低电平开始输出					
	TAA _n OE0		TOAA _n 0 引脚输出电平设置					
	0		禁止定时器输出 · 当 TAA _n OL0 位 = 0: 从 TOAA _n 0 引脚输出低电平 · 当 TAA _n OL0 位 = 1: 从 TOAA _n 0 引脚输出高电平					
	1		允许定时器输出 (从 TOAA _n 0 引脚输出方波)。					

注 由 TAA_nOL 位指定的定时器输出引脚 (TOAA_nm) 的输出电平如下所示:

· 当 TAA_nOL_m 位 = 0

· 当 TAA_nOL_m 位 = 1



- 注意事项**
1. 当 TAA_nCTL0.TAA_nCE 位 = 0 时，设置 TAA_nOL1, TAA_nOE1, TAA_nOL0, 和 TAA_nOE0 位。(当 TAA_nCE 位 = 1 时，可设置相同的值。)如果重新写入不能正确执行，将 TAA_nCE 位清为 0，然后重新进行位设定。
 2. 当 TAA_nCE 和 TAA_nOE_m 位为 0 时，即使 TAA_nOL_m 位被操作，TOAA_nm 引脚的输出电平也会发生变化。

备注 m = 0, 1

(4) TAA_n I/O 控制寄存器 1 (TAA_nIOC1)

TAA_nIOC1 寄存器为 8 位寄存器，用于控制捕获触发输入信号 (TIAAn0, TIAAn1 引脚) 的有效边沿。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: TAA0IOC1 FFFFF633H, TAA1IOC1 FFFFF643H,
TAA2IOC1 FFFFF653H, TAA3IOC1 FFFFF663H,
TAA5IOC1 FFFFF683H

	7	6	5	4	3	2	1	0
TAA _n IOC1 (n = 0至3, 5)	0	0	0	0	TAA _n IS3	TAA _n IS2	TAA _n IS1	TAA _n IS0

TAA _n IS3	TAA _n IS2	捕获触发输入信号 (TIAAn1 引脚)有效沿设置
0	0	无边沿检测 (捕获操作无效)
0	1	上升沿检测
1	0	下降沿检测
1	1	双边沿检测

TAA _n IS1	TAA _n IS0	捕获触发输入信号 (TIAAn0 引脚)有效沿设置
0	0	无边沿检测 (捕获操作无效)
0	1	上升沿检测
1	0	下降沿检测
1	1	双边沿检测

- 注意事项**
1. 当 TAA_nCTL0.TAA_nCE 位 = 0 时，设置 TAA_nIS3 至 TAA_nIS0 位。（当 TAA_nCE 位 = 1 时，可设置相同的值。）如果重新写入不能正确执行，将 TAA_nCE 位清为 0，然后重新进行位设定。
 2. 只有在自由运行定时器模式和脉冲宽度测量模式下，TAA_nIS3 至 TAA_nIS0 位才有效。在所有其它模式下，无法进行捕获操作。

(5) TAA_n I/O 控制寄存器 2 (TAA_nIOC2)

TAA_nIOC2 寄存器为 8 位寄存器，用于控制外部事件计数输入信号 (TIAAn0 引脚) 和外部触发输入信号 (TIAAn0 引脚) 的有效边沿。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: TAA0IOC2 FFFFF634H, TAA1IOC2 FFFFF644H,
TAA2IOC2 FFFFF654H, TAA3IOC2 FFFFF664H.
TAA5IOC2 FFFFF684H

	7	6	5	4	3	2	1	0
TAA _n IOC2 (n = 0 至 3, 5)	0	0	0	0	TAA _n EES1	TAA _n EES0	TAA _n ETS1	TAA _n ETS0

TAA _n EES1	TAA _n EES0	外部事件计数输入信号(TIAAn0 引脚)有效沿设置
0	0	无边沿检测 (外部事件计数无效)
0	1	上升沿检测
1	0	下降沿检测
1	1	双边沿检测

TAA _n ETS1	TAA _n ETS0	外部触发输入信号 (TIAAn0 引脚) 有效沿设置
0	0	无边沿检测 (外部触发无效)
0	1	上升沿检测
1	0	下降沿检测
1	1	双边沿检测

- 注意事项**
1. 当 TAA_nCTL0.TAA_nCE 位 = 0 时，设置 TAA_nEES1，TAA_nEES0，TAA_nETS1 和 TAA_nETS0 位。(当 TAA_nCE 位 = 1 时，可设置相同的值。)如果重新写入不能正确执行，将 TAA_nCE 位清为 0，然后重新进行位设定。
 2. 只有在 TAA_nCTL1.TAA_nEEE 位= 1 或设置了外部事件计数模式 (TAA_nCTL1.TAA_nMD2 至 TAA_nCTL1.TAA_nMD0 位 = 001) 时，TAA_nEES1 和 TAA_nEES0 位才有效。
 3. 只有在外部触发脉冲输出模式 (TAA_nCTL1.TAA_nMD2 至 TAA_nCTL1.TAA_nMD0 位 = 010) 或处于单次脉冲输出模式 (TAA_nCTL1.TAA_nMD2 至 TAA_nCTL1.TAA_nMD0 = 011) 时，TAA_nETS1 和 TAA_nETS0 位才有效。

(6) TAA_n I/O 控制寄存器 4 (TAA_nIOC4)

TAA_nIOC4 寄存器为 8 位寄存器，用来控制定时器输出。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。该寄存器不通过停止定时器操作进行复位 (TAA_nCTL0.TAA_nCE = 0)。

注意事项 1. 在以下状态禁止访问 TAA_nIOC4 寄存器。有关详细说明，参见 3.4.9 (2) 访问片上专用外围 I/O 寄存器。

- 当 CPU 工作于副时钟，而且主时钟振荡停止时
- 当 CPU 工作于内部振荡时钟时

2. TAA_nIOC4 寄存器只能在间隔定时器模式和自由运行定时器模式下进行设置。在其它模式下确保将 TAA_nIOC4 寄存器设置为 00H (有关模式设置的详情，参见 7.4 (2) TAA_n 控制寄存器 1 (TAA_nCTL1))。如果 TAA_nCCR0 和 TAA_nCCR1 寄存器设置为捕获功能，即使设置为自由运行定时器模式，TAA_nIOC4 寄存器的设置也无效。

复位后: 00H R/W 地址: TAA0IOC4 FFFFF63CH, TAA1IOC4 FFFFF64CH,
TAA2IOC4 FFFFF65CH, TAA3IOC4 FFFFF66CH,
TAA5IOC4 FFFFF68CH

	7	6	5	4	3	2	1	0
TAA _n IOC4 (n = 0至 3, 5)	0	0	0	0	TAA _n OS1	TAA _n OR1	TAA _n OS0	TAA _n OR0

TAA _n OS1	TAA _n OR1	TIAAn1 引脚的切换控制
0	0	无请求。正常切换操作。
0	1	复位请求 在下一个16位计数器的值和TAA _n CCR1寄存器匹配时 固定为无效电平
1	0	设置请求 在下一个16位计数器的值和TAA _n CCR1寄存器匹配时 固定为有效电平
1	1	保持请求 保持当前输出电平

TAA _n OS0	TAA _n OR0	TIAAn0 引脚的切换控制
0	0	无请求。正常切换操作。
0	1	复位请求 在下一个16位计数器的值和TAA _n CCR0寄存器匹配时 固定为无效电平
1	0	设置请求 在下一个16位计数器的值和TAA _n CCR0寄存器匹配时 固定为有效电平
1	1	保持请求 保持当前输出电平

(7) TAA_n 选项寄存器 0 (TAA_nOPT0)

TAA_nOPT0 寄存器为 8 位寄存器，用来设置捕获/比较操作和检测计数器溢出。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: TAA0OPT0 FFFFF635H, TAA1OPT0 FFFFF645H,
TAA2OPT0 FFFFF655H, TAA3OPT0 FFFFF665H,
TAA5OPT0 FFFFF685H

	7	6	5	4	3	2	1	0
TAA _n OPT0	0	0	TAA _n CCS1	TAA _n CCS0	0	0	0	TAA _n OVF

(n = 0 至 3, 5)

TAA _n CCS1	TAA _n CCR1 寄存器捕获/比较的选择
0	比较寄存器选择
1	捕获寄存器选择
TAA _n CCS1 位的设置仅在自由运行定时器模式下有效。	

TAA _n CCS0	TAA _n CCR0 寄存器捕获/比较的选择
0	比较寄存器选择
1	捕获寄存器选择
TAA _n CCS0 位的设置仅在自由运行定时器模式下有效。	

TAA _n OVF	TAA _n 溢出检测标志
设置 (1)	发生溢出
复位 (0)	向 TAA _n OVF 位写入0或 TAA _n CTL0.TAA _n CE 位 = 0
- 自由运行定时器模式下或脉冲宽度测量模式下，当 16 位计数器的计数值由 FFFFH 变为 0000H 发生溢出时，TAA_nOVF 位设置为 1。 - TAA_nOVF 设置为 1 的同时产生中断请求信号 (INTTAA_nOV)。 - 除自由运行定时器模式和脉冲宽度测量模式外不产生 INTTAA_nOV 信号。 - 当 TAA_nOVF 位 = 1 时即使读取 TAA_nOVF 位或 TAA_nOPT0 寄存器 TAA_nOVF 位也不清零。 - TAA_nOVF 位可读可写，但是 TAA_nOVF 不能由软件设置为 1。写入 1 对 TAA_n 操作没有影响。	

- 注意事项**
1. 当 TAA_nCE 位 = 0 时重写 TAA_nCCS1 和 TAA_nCCS0 位。
(当 TAA_nCE 位 = 1 时可写入相同值)。如果重新写入不能正确执行，将 TAA_nCE 位清为 0，然后重新进行位设定。
 2. 一定要将位 1 至 3，6 和 7 设置为“0”。

(8) TAA_n 选项寄存器 1（TAA_nOPT1）

TAA_nOPT1 寄存器是 8 位寄存器，控制通过级联实现的 32 位捕获功能。

在定时器操作时禁止重写该寄存器（TAA_nCTL0.TAA_nCE = 1）。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: TAA0OPT1 FFFFF63DH, TAA2OPT1 FFFFF65DH

TAA_nOPT1
(n = 0, 2)

7	6	5	4	3	2	1	0
TAA _n CSE	0	0	0	0	0	0	0

TAA _n CSE	级联控制
0	独立操作或作为级联功能的低位端操作
1	作为级联功能的高位端操作

注意事项

1. 级联和定时器调谐操作不能一起使用。为使用级联确保将 TAA_nCTL1.TAA_nSYE 设置为 0。

2. 为使用级联，需设置为自由运行定时器模式并且将 TAA_nCCR0 和 TAA_nCCR1 寄存器用作捕获寄存器。

有关级联的详情，参见 7.8 级联

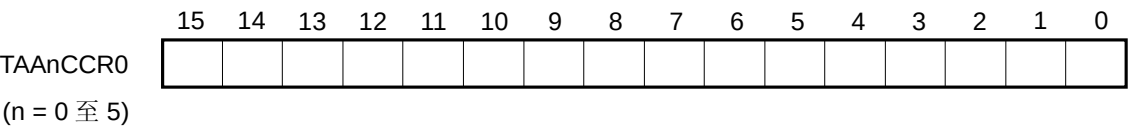
(9) TAA_n 捕获/比较寄存器 0 (TAA_nCCR0)

根据不同工作模式，TAA_nCCR0 寄存器可以用作捕获寄存器或比较寄存器。
只有在自由计时器模式下，依据 TAA_nOPT0.TAA_nCCS0 位的设定，该寄存器可用作捕获寄存器或比较寄存器。
在脉宽测量模式中，TAA_nCCR0 寄存器只可以用作捕获寄存器。而在其它模式时，该寄存器只可以用作比较寄存器。
TAA_nCCR0 寄存器可以在操作中进行读取和写入。
该寄存器可按 16 位为单位进行读写。
复位后本寄存器设置为 0000H。

注意事项 在以下状态中，禁止访问 TAA_nCCR0 寄存器。有关详细说明，参见 3.4.9 (2) 访问片上专用外围 I/O 寄存器。

- 当 CPU 工作于副时钟，而且主时钟振荡停止时
- 当 CPU 工作于内部振荡时钟时

复位后: 0000H R/W 地址: TAA0CCR0 FFFFF636H, TAA1CCR0 FFFFF646H,
TAA2CCR0 FFFFF656H, TAA3CCR0 FFFFF666H,
TAA4CCR0 FFFFF676H, TAA5CCR0 FFFFF686H



(a) 作为比较寄存器时的功能

即便在 TAAAnCTL0.TAAAnCE 位=1 时，TAAAnCCR0 寄存器仍可以重新写入。

TAAAnCCR0 寄存器的设定值传送给 CCR0 缓冲寄存器。当 16 位计数器的值与 CCR0 缓冲寄存器的值相匹配时，就会生成一个比较匹配中断请求信号（INTTAAAnCC0）。如果此时 TOAAAn0 引脚输出允许，则 TOAAAn0 引脚的输出会反相。

在间隔定时器模式、外部事件计数器模式、外部触发脉冲输出模式、单次脉冲输出模式或脉宽调制信号输出模式时，当 TAAAnCCR0 寄存器作为循环寄存器时，如果 16 位计数器的计数值与 CCR0 缓冲寄存器数值匹配，则计数器清零为（0000H）。

(b) 用作捕获寄存器

在自由运行定时器模式下，当 TAAAnCCR0 寄存器用作捕获寄存器时，若检测到捕捉触发输入引脚（TIAAn0 引脚）的有效边沿，则 16 位计数器的当前计数值存储于 TAAAnCCR0 寄存器中。在脉宽测量模式下，若检测到捕获触发输入引脚（TIAAn0 引脚）的有效边沿，16 位计数器的当前计数值就储存在 TAAAnCCR0 寄存器中，且 16 位计数器清零（0000H）。

即使捕获操作与对 TAAAnCCR0 寄存器的读取发生冲突，还是可以正确读取 TAAAnCCR0 寄存器的值。

下表显示了捕获/比较寄存器在各工作模式下的功能，以及怎样向比较寄存器写入数据。

表 7-3. 捕获/比较寄存器在各模式下的功能和如何写入比较寄存器

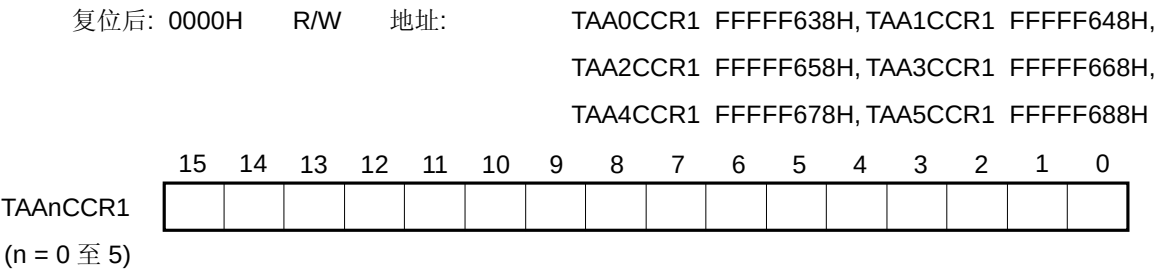
操作模式	捕获/比较寄存器	如何写入比较寄存器
间隔定时器	比较寄存器	可随时写入
外部事件计数器	比较寄存器	可随时写入
外部触发脉冲输出	比较寄存器	批写入
单次脉冲输出	比较寄存器	可随时写入
PWM（脉宽调制信号）输出	比较寄存器	批写入
自由运行定时器	捕获/比较寄存器	可随时写入
脉冲宽度测量	捕获寄存器	—

(10) TAA_n 捕获/比较寄存器 1 (TAA_nCCR1)

根据不同工作模式，TAA_nCCR1 寄存器可以用作捕获寄存器或比较寄存器。
只有在自由计数器模式下，依据 TAA_nOPT0.TAA_nCCS1 位的设定，该寄存器可用作捕获寄存器或比较寄存器。在脉宽测量模式中，TAA_nCCR1 寄存器只可以用作捕获寄存器。而在其它模式时，该寄存器只可以用作比较寄存器。
TAA_nCCR1 寄存器可以在操作中进行读取和写入。
该寄存器可按 16 位为单位进行读写。
复位后本寄存器设置为 0000H。

注意事项 在以下状态中，禁止访问 TAA_nCCR1 寄存器。有关详细说明，参见 3.4.9 (2) 访问片上专用外围 I/O 寄存器。

- 当 CPU 工作于副时钟，而且主时钟振荡停止时
- 当 CPU 工作于内部振荡时钟时



(a) 作为比较寄存器时的功能

即便在 TAAAnCTL0.TAAAnCE 位=1 时，TAAAnCCR1 寄存器仍可以重新写入。

TAAAnCCR1 寄存器的设定值传送给 CCR1 缓冲寄存器。当 16 位计数器的值与 CCR1 缓冲寄存器的值相匹配时，就会生成一个比较匹配中断请求信号（INTTAAAnCC1）。如果此时 TOAAAn1 引脚输出允许，则 TOAAAn1 引脚的输出会反相。

(b) 用作捕获寄存器

在自由运行定时器模式下，当 TAAAnCCR1 寄存器用作捕获寄存器时，若检测到捕捉触发输入引脚（TIAAn1 引脚）的有效边沿，则 16 位计数器的当前计数值存储于 TAAAnCCR1 寄存器中。在脉宽测量模式下，若检测到捕获触发输入引脚（TIAAn1 引脚）的有效边沿，16 位计数器的当前计数值就储存在 TAAAnCCR1 寄存器中，且 16 位计数器清零（0000H）。

即使捕获操作与对 TAAAnCCR1 寄存器的读取发生冲突，还是可以正确读取 TAAAnCCR1 寄存器的值。

下表显示了捕获/比较寄存器在各工作模式下的功能，以及怎样向比较寄存器写入数据。

表 7-4. 捕获/比较寄存器在各模式下的功能和如何写入比较寄存器

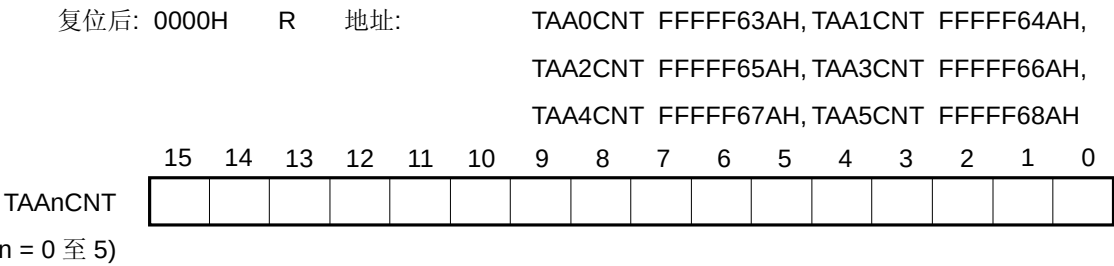
操作模式	捕获/比较寄存器	如何写入比较寄存器
间隔定时器	比较寄存器	可随时写入
外部事件计数器	比较寄存器	可随时写入
外部触发脉冲输出	比较寄存器	批写入
单次脉冲输出	比较寄存器	可随时写入
PWM（脉宽调制信号）输出	比较寄存器	批写入
自由运行定时器	捕获/比较寄存器	可随时写入
脉冲宽度测量	捕获寄存器	—

(11) TAA_n 计数器读取缓冲寄存器 (TAA_nCNT)

TAA_nCNT 寄存器为读出缓冲寄存器，可以读取 16 位计数器的计数值。
如果在 TAA_nCTL0.TAA_nCE 位=1 时读取该寄存器，可以读出 16 位定时器的计数值。
本寄存器为只读寄存器，可以按 16 位读取。
当 TAA_nCE 位= 0 时，TAA_nCNT 寄存器的值被清为 0000H。如果此时读取 TAA_nCNT 寄存器，则不会读出 16 位计数器的值 (FFFFH)，读出的值为 0000H。
复位将 TAA_nCE 清 0.因此 TAA_nCNT 寄存器的值被清除为 0000H。

注意事项 在以下状态中，禁止访问 TAA_nCNT 寄存器。有关详细说明，参见 3.4.9 (2) 访问片上专用外围 I/O 寄存器。

- 当 CPU 工作于副时钟，而且主时钟振荡停止时
- 当 CPU 工作于内部振荡时钟时



(12) 噪声抑制控制寄存器（TANFC）

TIAAn0 和 TIAAn1 引脚具有可选的数字噪声抑制器。使用 TANFC 寄存器来进行噪声抑制选择。

当选择数字噪声抑制时，用于数字采样的采样时钟可从 f_{xx} 和 f_{xx}/4 中选择。采样执行 3 次。

该寄存器可按 8 位为单位进行读写。

复位后本寄存器被设置为 00H。

注意事项 采样时钟改变后直到数字噪声抑制器被初始化所需要的时间等于采样时钟× 3。因此，如果在采样时钟改变后而未经过采样时钟× 3 个时钟前，输入 TIAAn0 和 TIAAn1 引脚的有效沿，可能产生中断请求信号。因此，使用 TAA 的外部触发功能、外部事件功能和捕获触发功能时，在经过采样时钟× 3 个时钟后，使能 TAA 操作。

备注 n = 0 至 3, 5

复位后: 00H R/W 地址: FFFFF724H

	7	6	5	4	3	2	1	0
TANFC	TANFEN	0	0	0	0	0	0	TANFC0

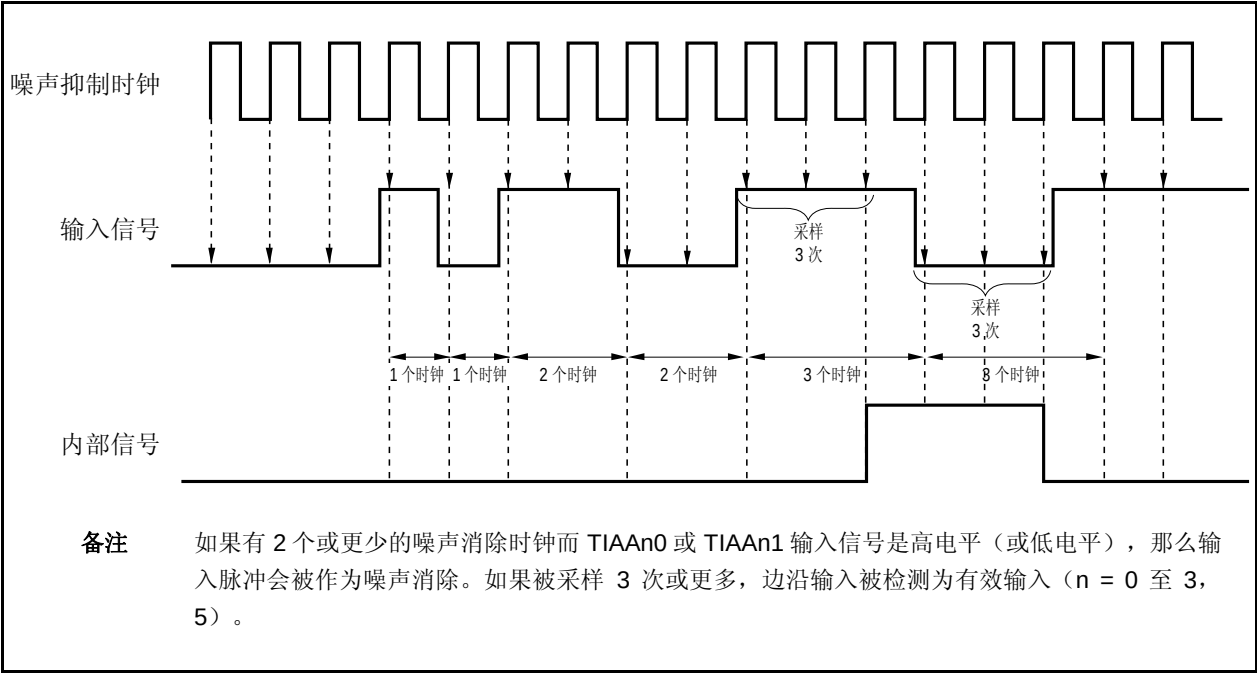
TANFEN	数字噪声抑制的设置
0	不执行数字噪声抑制
1	执行数字噪声抑制

TANFC0	数字采样时钟
0	f _{xx}
1	f _{xx} /4

- 备注**
1. 因为采样执行了 3 次，可靠的消除噪声宽度是 2 个采样时钟。
 2. 在噪声宽度小于 2 个采样时钟的情况下，如果采样时钟输入与噪声同步就会产生一个中断请求信号。

由定时器 AA 输入引脚数字滤波器执行的噪声抑制的时序举例如图 7-2:

图 7-2 数字噪声消除时序举例



7.5 操作

定时器 TAA_n 可以执行如下操作。

操作	TAA _n CTL1.TAA _n EST 位 (软件触发位)	TIAA _n 0 引脚 (外部触发输入)	捕获/比较寄存器设置	比较寄存器写入
间隔定时器模式	无效	无效	仅比较寄存器有效	可随时写入
外部事件计数器模式 ^{注 1}	无效	无效	仅比较寄存器有效	可随时写入
外部触发脉冲输出模式 ^{注 2}	有效	有效	仅比较寄存器有效	批写入
单次脉冲输出模式 ^{注 2}	有效	有效	仅比较寄存器有效	可随时写入
脉宽调制信号输出模式	无效	无效	仅比较寄存器有效	批写入
自由定时器模式	无效	无效	可以切换	可随时写入
脉冲宽度测量模式 ^{注 2}	无效	无效	仅捕获寄存器有效	无效

- 注**
1. 在用作外部事件计数模式时，应指定不检测的 TIAA_n0 引脚捕获触发输入的有效边沿（通过将 TAA_nIOC1.TAA_nIS1 和 TAA_nIOC1.TAA_nISO 位清为“00”）。
 2. 当用作外部触发脉冲输出模式、单次脉冲输出模式和脉冲宽度测量模式时，选择内部时钟作为计数时钟。（通过清除 TAA_nCTL1.TAA_nEEE 位为 0）。

备注 n = 0 至 3, 5

(1) 随时写入和批写入

在定时器操作 (TAACTL0.TAA nCE 位=1) 期间, TAA n, TAA nCCR0 和 TAA nCCR1 寄存器可以重写, 但 CCR0 和 CCR1 缓冲寄存器的写入方法 (随时写入, 批写入) 会根据模式的不同而变化。

(a) 可随时写入

在该模式中, 在定时器操作期间数据可以随时从 TAA nCCR0 和 TAA nCCR1 寄存器传送到 CCR0 和 CCR1 缓冲寄存器中。

图 7-3.基本的随时写入操作流程举例 (TAA0 的间隔定时器模式)

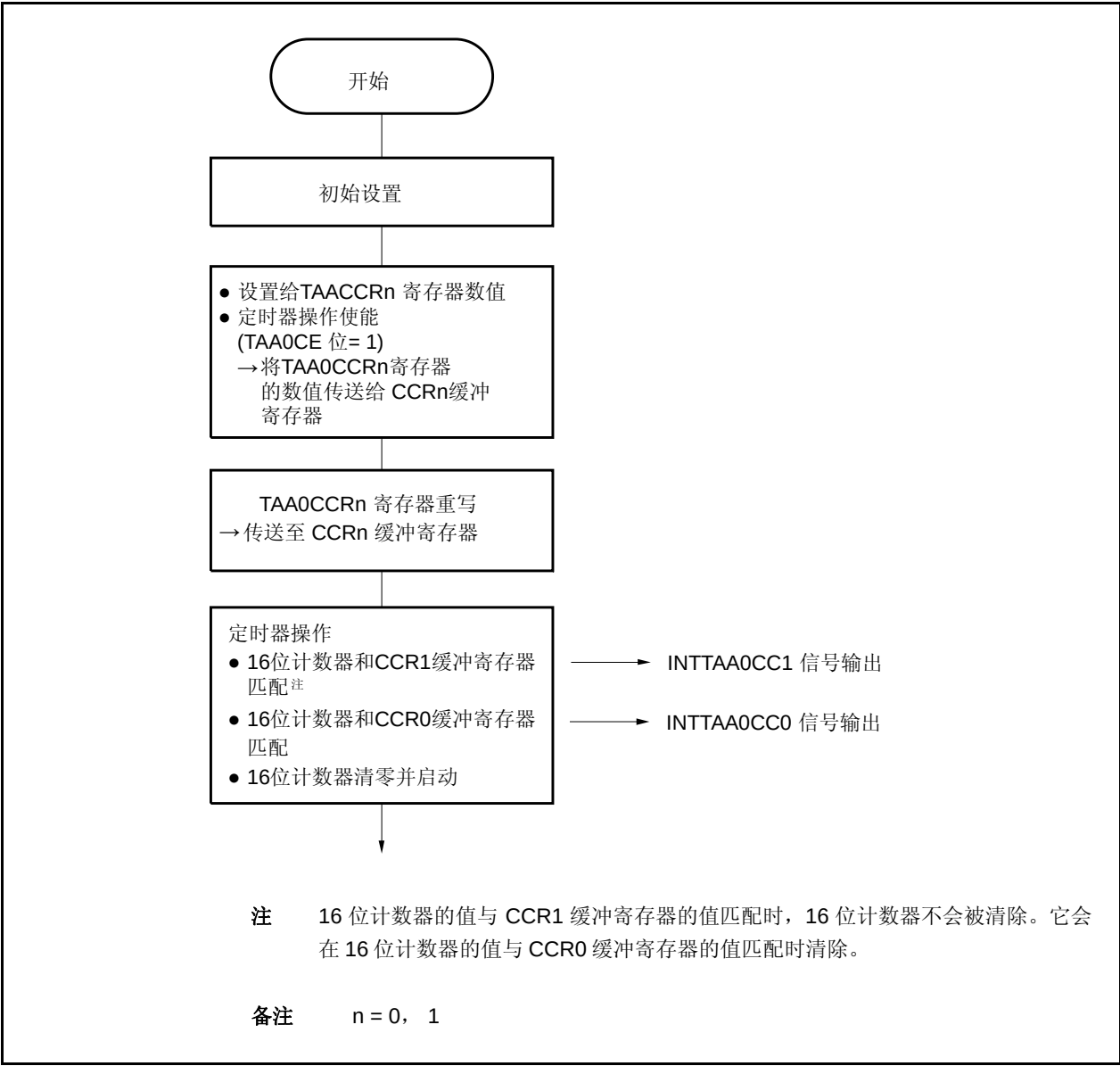
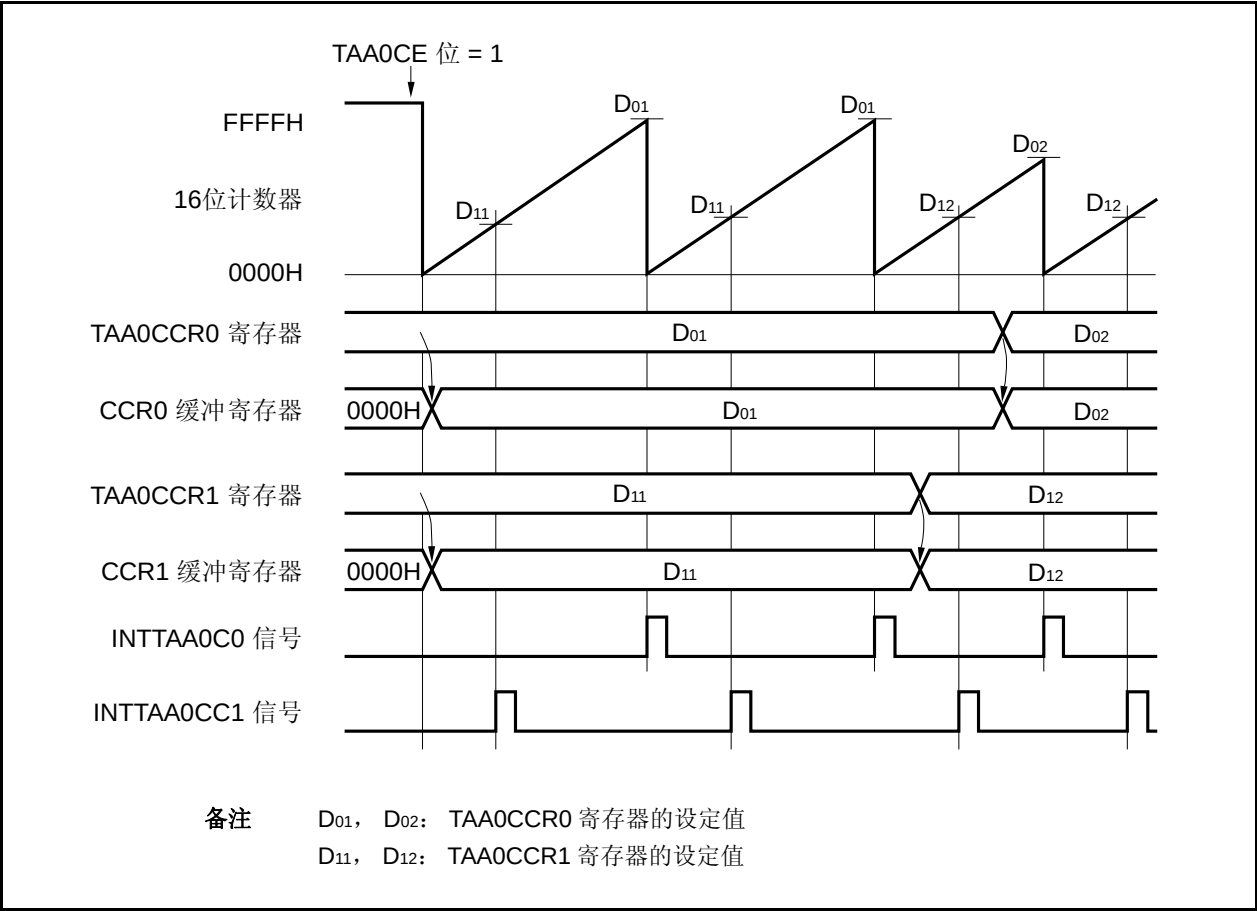


图 7-4.随时写入时序举例 (TAA0 的间隔定时器模式)



(b) 批写入

在该模式中，在定时器操作期间所有数据会一次都从 TAA_nCCR0 和 TAA_nCCR1 寄存器传送到 CCR0 和 CCR1 缓冲寄存器中。此数据会在 CCR0 缓冲寄存器的值与 16 位计数器的值匹配时转移。通过写入 TAA_nCCR1 寄存器允许触发。允许还是禁止下个转移时序是通过对 TAA_nCCR1 寄存器执行或不执行写入操作来控制的。

为了在改写 TAA_nCCR0 和 TAA_nCCR1 寄存器时将设定值转变为 16 位计数器的比较值（换句话说，为了让该值转移到 CCR0 和 CCR1 缓冲寄存器中），在 16 位计数器的值与 CCR0 缓冲寄存器的值匹配前，需要先对 TAA_nCCR0 寄存器进行改写，然后再对 TAA_nCCR1 寄存器进行写入操作。因此，一旦 16 位计数器的计数值与 CCR0 缓冲寄存器的值匹配，TAA_nCCR0 和 TAA_nCCR1 寄存器的值将会被转移到 CCR0 和 CCR1 缓冲寄存器中。因此即使只想改写 TAA_nCCR0 寄存器的值，相同的值（与 TAA_nCCR1 寄存器的预置值相同）也会被写入到 TAA_nCCR1 寄存器中。

图 7-5.基本的批写入操作流程举例（TAA0 的脉宽调制信号输出模式）

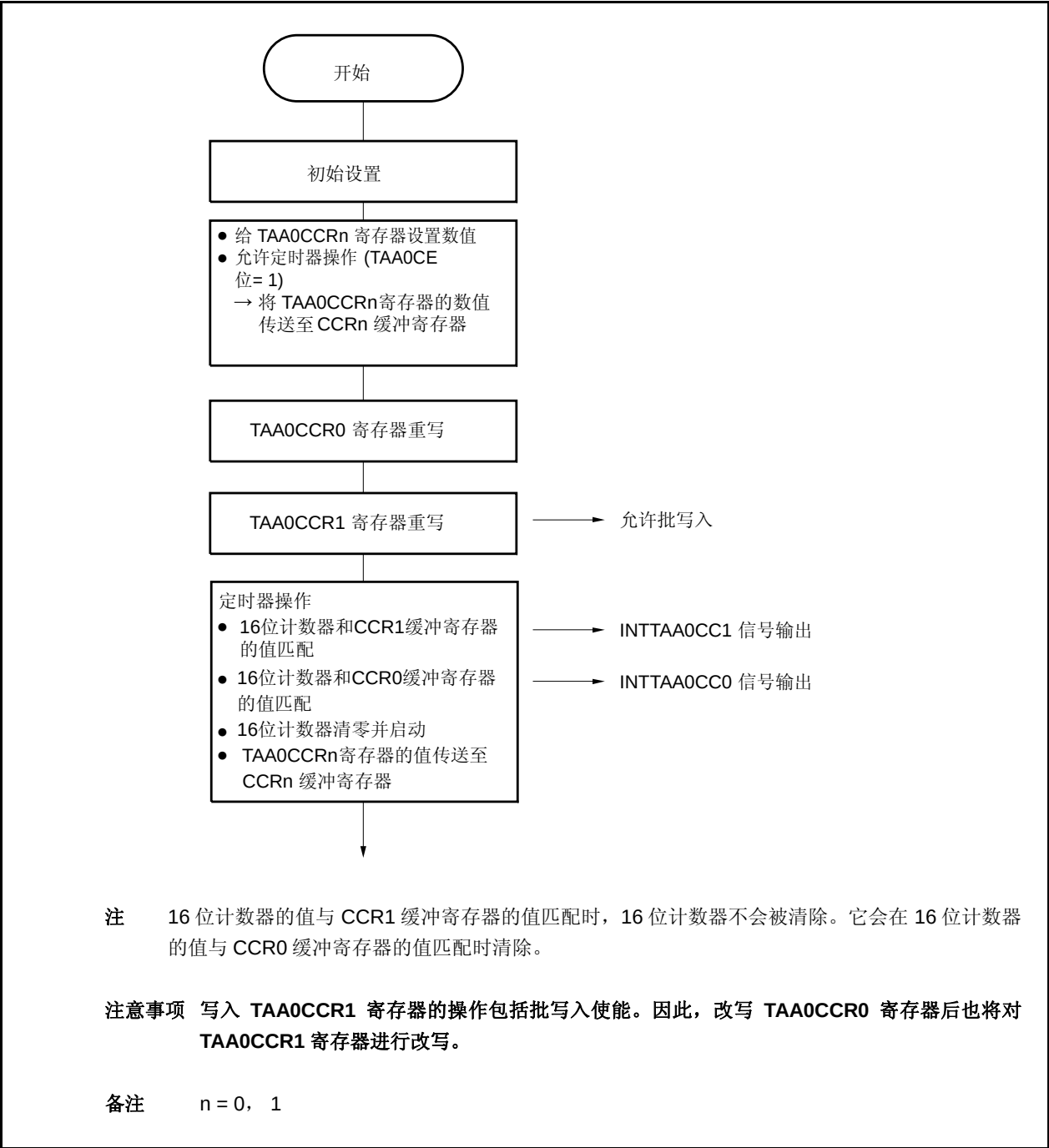
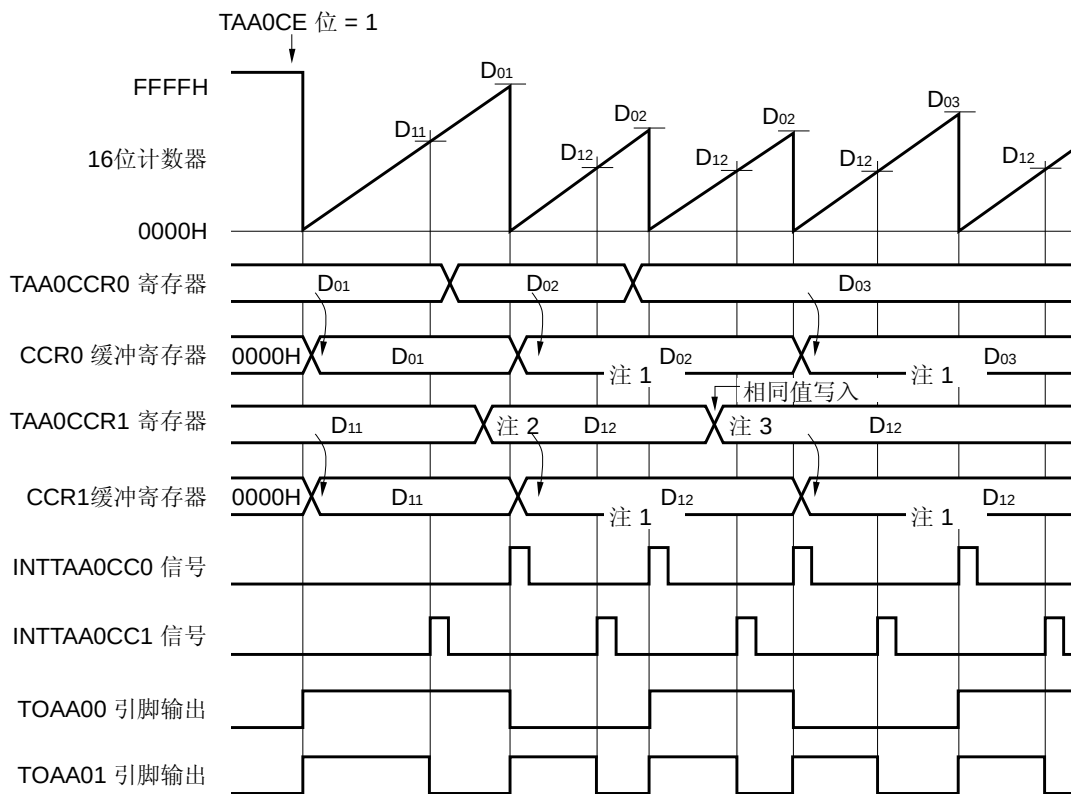


图 7-6.批写入时序举例 (TAA0 的间隔定时器模式)



- 注 1. 由于 TAA0CCR1 寄存器没有被重写，因此 D03 不会被转移。
- 注 2. 由于 TAA0CCR1 寄存器已经被写入 (D12)，因此，一旦 16 位计数器的值与 TAA0CCR0 寄存器 (D01) 的值匹配，数据将会被转移到 CCR1 缓冲寄存器中。
- 注 3. 由于 TAA0CCR1 寄存器已经被写入 (D12)，因此，一旦 16 位计数器的值与 TAA0CCR0 寄存器 (D02) 的值匹配，数据将会被转移到 CCR1 缓冲寄存器中。

备注 D01, D02, D03: TAA0CCR0 寄存器的设定值
D11, D12: TAA0CCR1 寄存器的设定值

7.5.1 间隔定时器模式 (TAAmMD2 至TAAmMD0 位 = 000)

在间隔定时器模式下，若 TAAAnCTL0.TAAAnCE 位被置为 1，则会以指定时间间隔生成中断请求信号 (INTTAAAnCC0)，从 TOAAAn0 引脚输出半周期等于该时间间隔的方波。
通常，在间隔定时器模式下不使用 TAAAnCCR1 寄存器。

图 7-7. 间隔定时器的配置

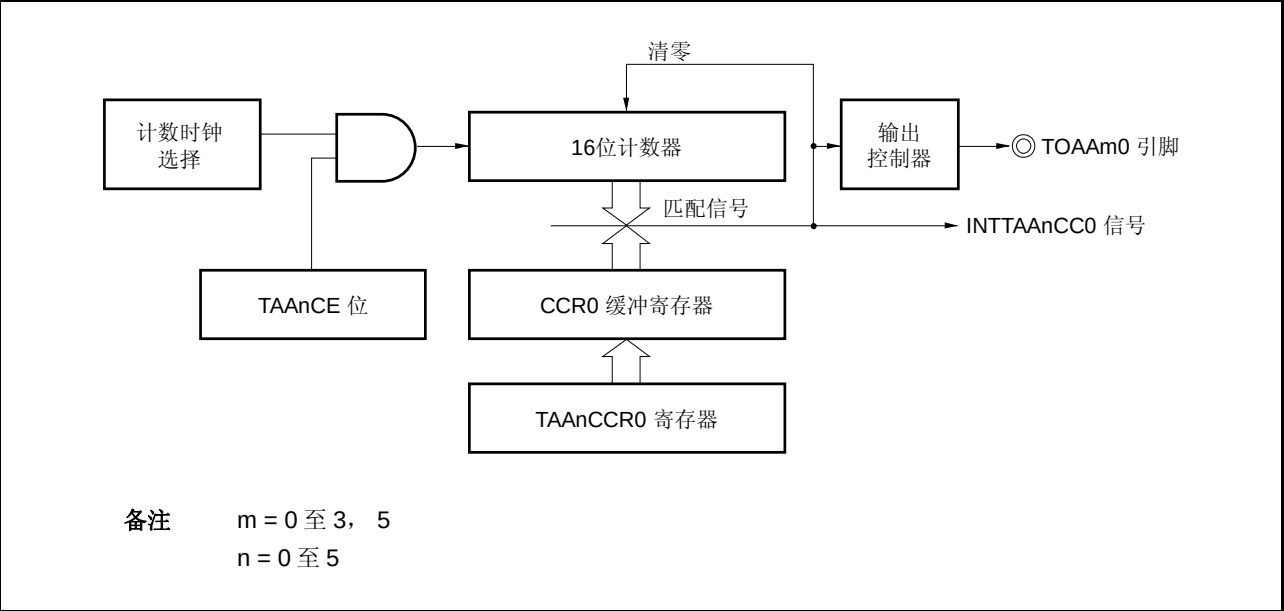
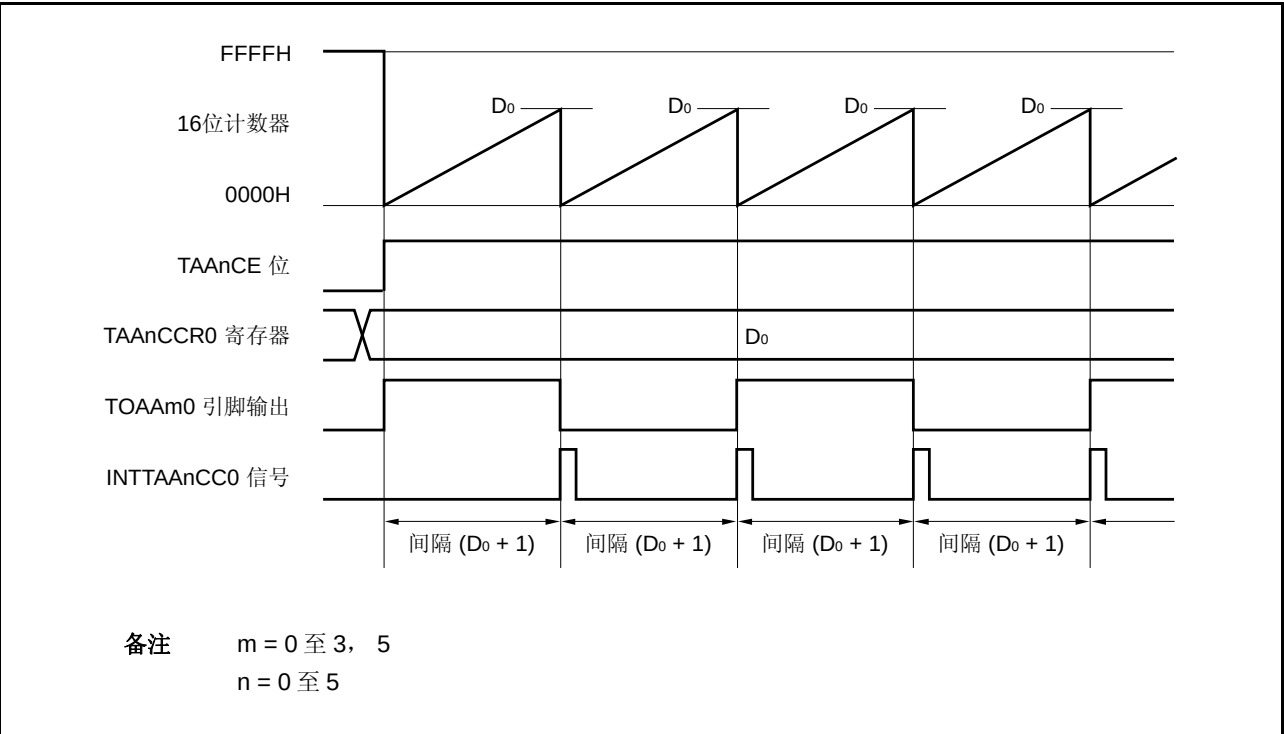


图 7-8. 间隔定时器工作的基本时序



当 TAA_nCE 位被置为 1 时，16 位计数器的值由计数时钟同步从 FFFFH 清零为 0000H，而且计数器开始计数。此时，TOAA_n0 引脚的输出反相。此外，TAA_nCCR0 寄存器的设定值传送给 CCR0 缓冲寄存器。

当 16 位计数器的计数值和 CCR0 缓冲寄存器的值相匹配时，则 16 位计数器清为 0000H，TOAA_n0 引脚输出反相，并生成一个比较匹配中断请求信号（INTTAA_nCC0）。

时间间隔可用下式计算。

$$\text{时间间隔} = (\text{TAA}_n\text{CCR0 寄存器的设定值} + 1) \times \text{计数时钟周期}$$

备注 m = 0 至 3, 5
 n = 0 至 5

图 7-9. 间隔定时器模式操作时的寄存器设置 (1/2)

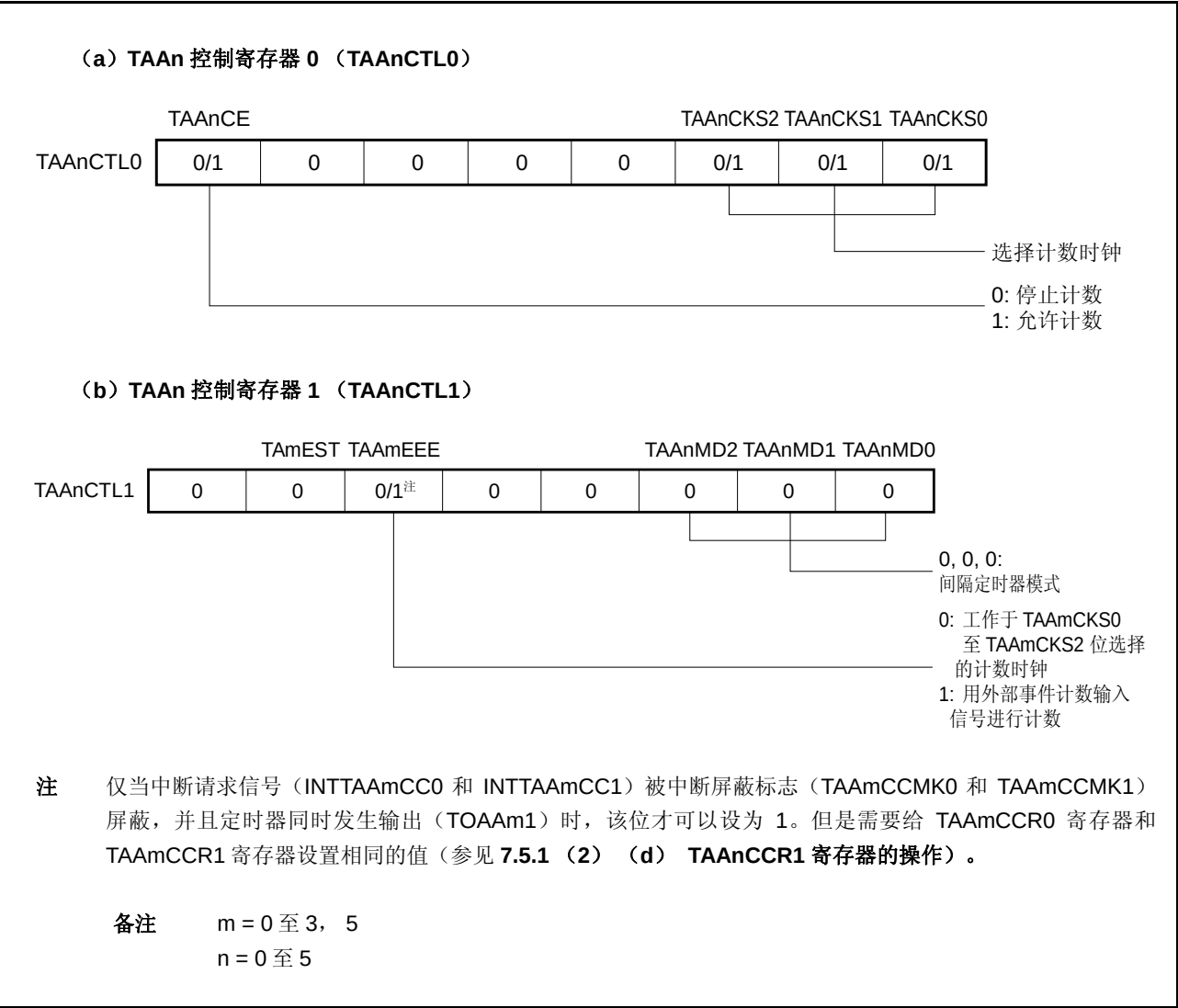
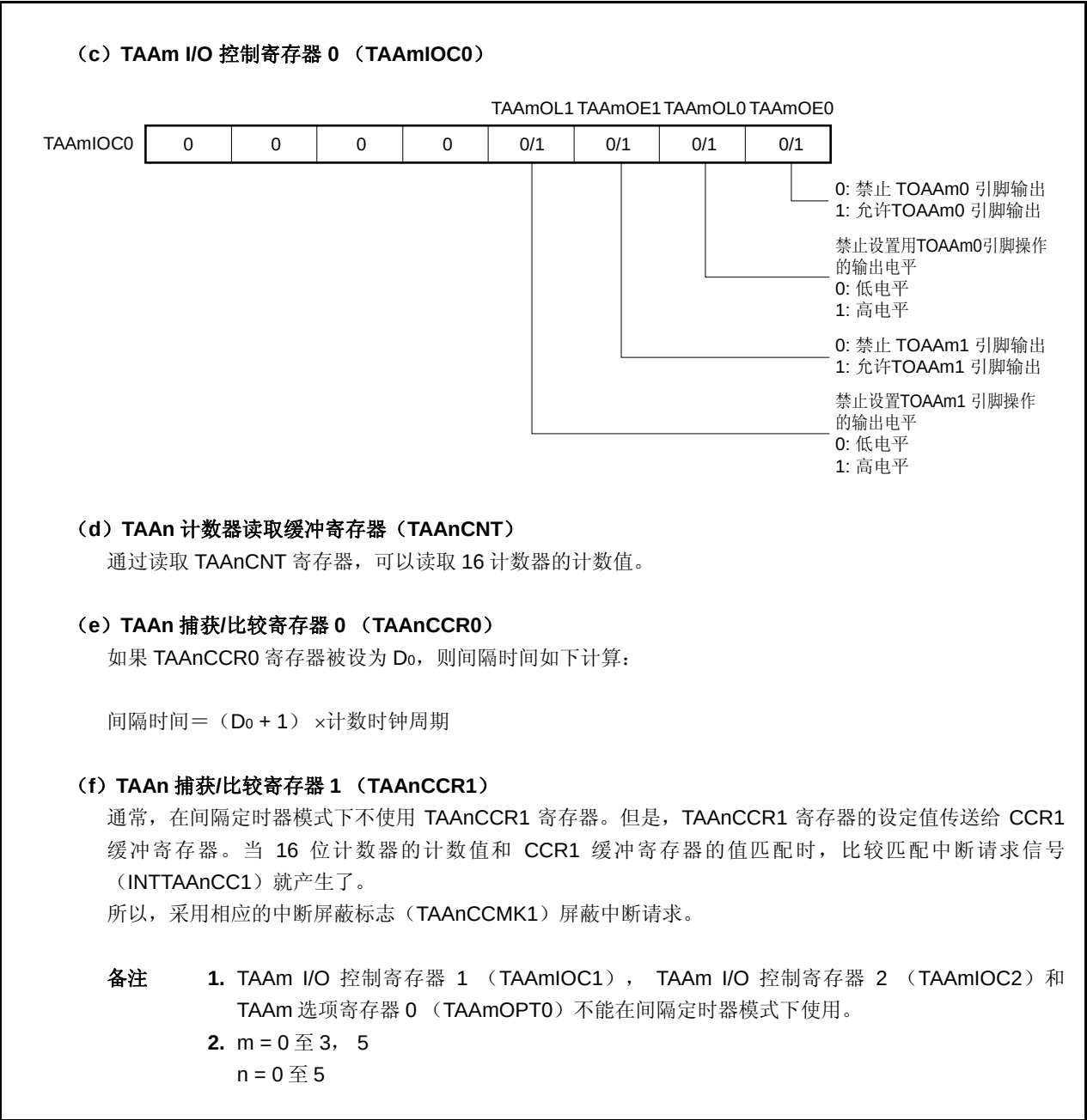
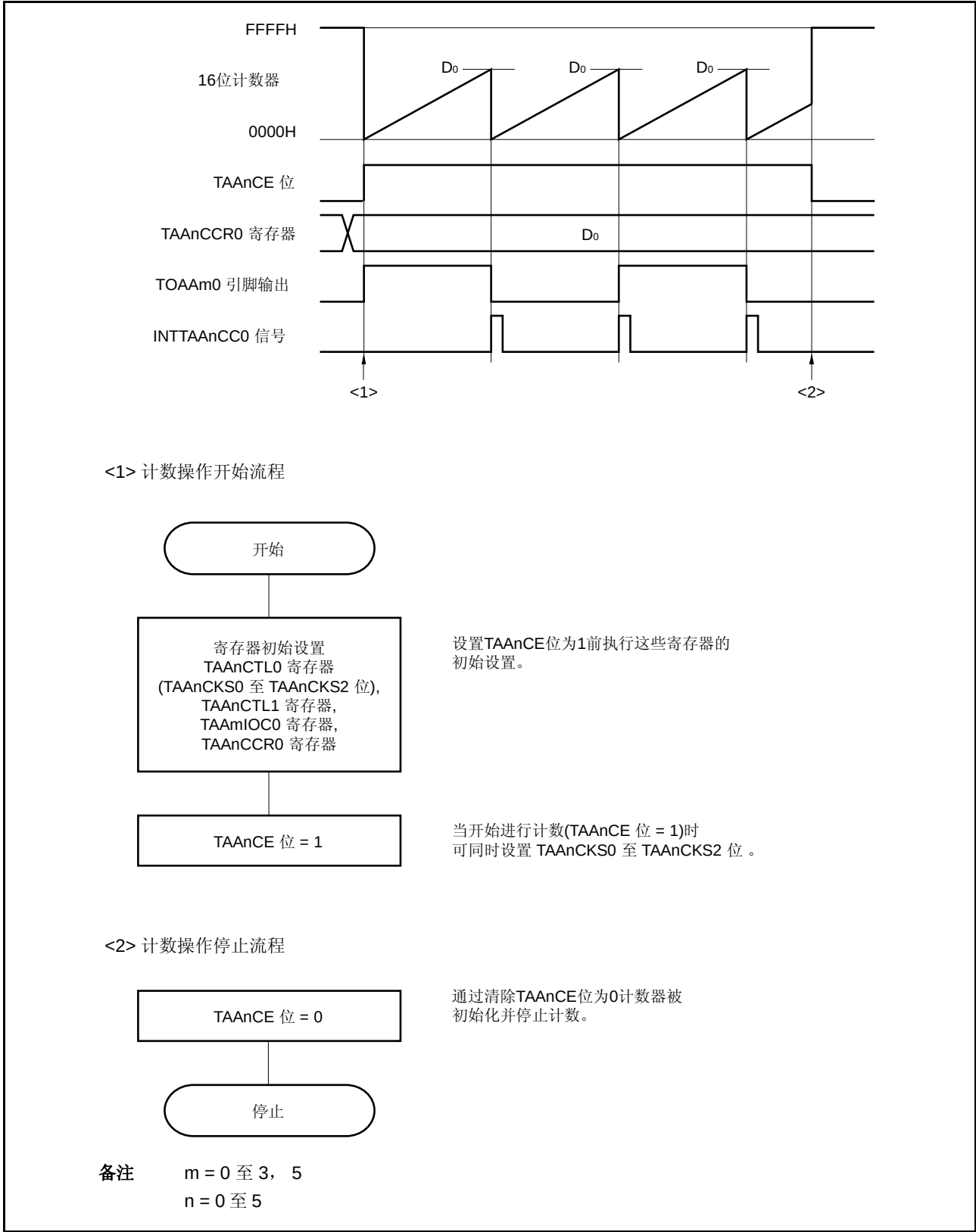


图 7-9. 间隔定时器模式操作时的寄存器设置 (2/2)



(1) 间隔定时器模式工作流程

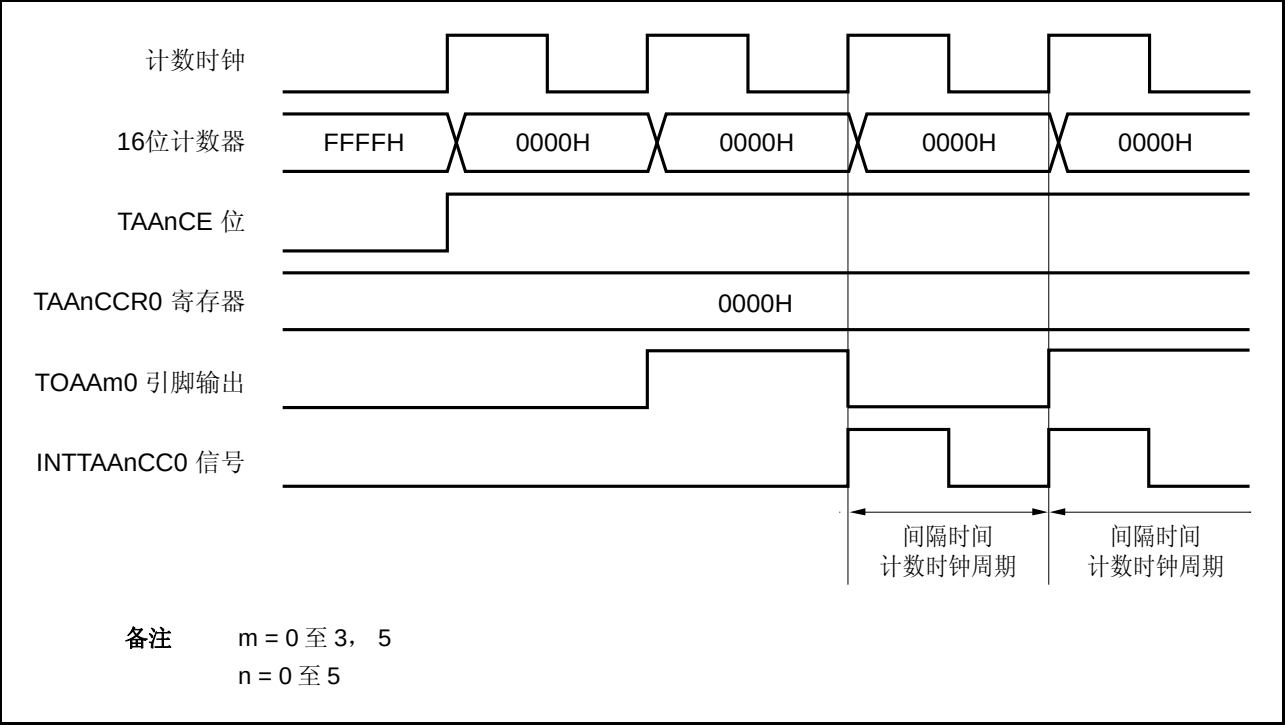
图 7-10. 间隔定时器模式下的软件处理流程



(2) 间隔定时器模式工作时序

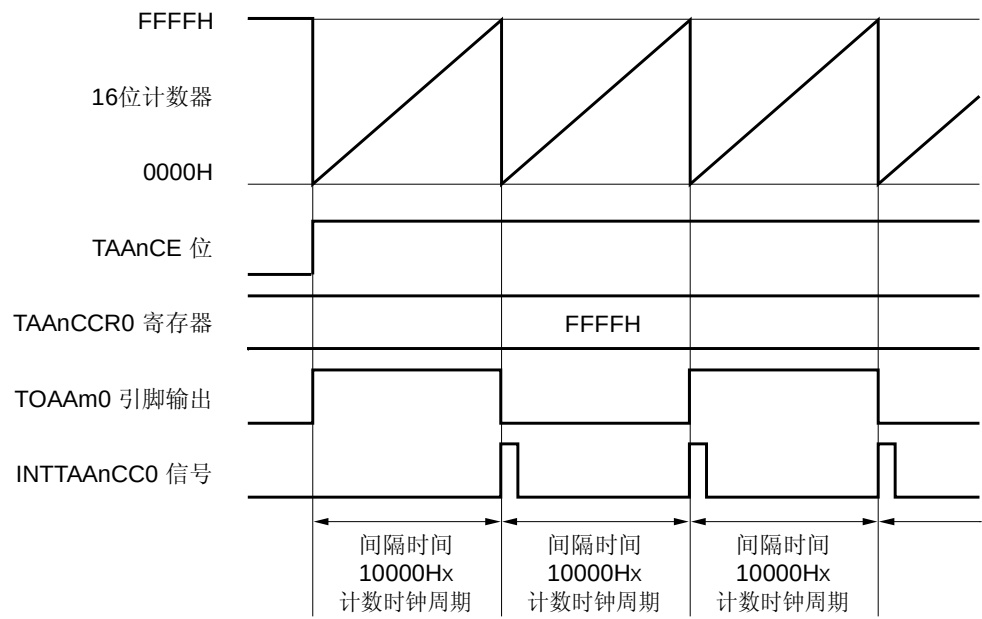
(a) TAAAnCCR0 寄存器清为 0000H 时的工作时序

如果 TAAAnCCR0 寄存器清为 0000H，第一个计数时钟后的每个计数时钟都会产生 INTTAAAnCC0 信号，而且，TOAAAn0 引脚的输出反相。
16 位计数器的值总是 0000H。



(b) TAAAnCCR0 寄存器被设置为 FFFFH 时的操作

如果 TAAAnCCR0 寄存器被设置为 FFFFH，那么 16 位计数器可以计数直至 FFFFH。当下一个计数到来时，计数器同步清为 0000H。产生 INTTAAAnCC0 信号，并且 TOAAAn0 引脚电平输出反转。此时，并不产生溢出中断请求信号（INTTAAAnOV），而且，溢出标志（TAAmOPT0.TAAmOVF 位）也不会置 1。

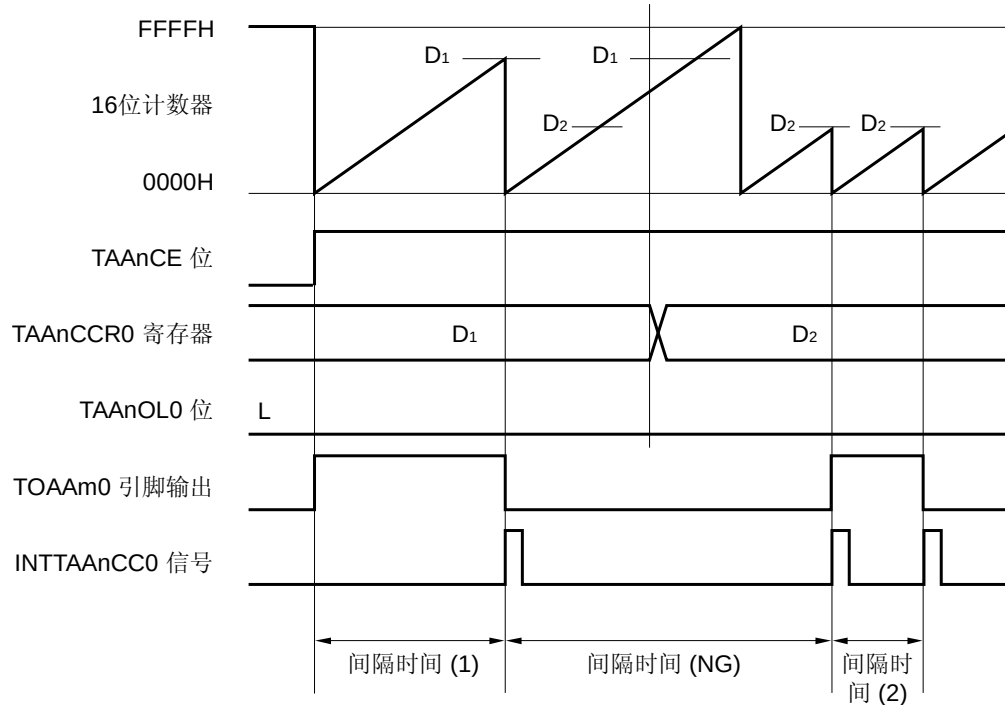


备注 m = 0 至 3, 5
n = 0 至 5

(c) 有关重写 TAAAnCCR0 寄存器的注意事项

要将 TAAAnCCR0 寄存器的值变为比当前值较小的值，应停止计数，然后改变其设定值。

如果在计数器进行计数期间，对寄存器 TAAAnCCR0 的值重写入一个较小的数值，则 16 位计数器可能发生溢出。



- 备注**
1. 间隔时间 (1): $(D_1 + 1) \times \text{计数时钟周期}$
 间隔时间 (NG): $(10000H + D_2 + 1) \times \text{计数时钟周期}$
 间隔时间 (2): $(D_2 + 1) \times \text{计数时钟周期}$
 2. $m = 0$ 至 3, 5
 $n = 0$ 至 5

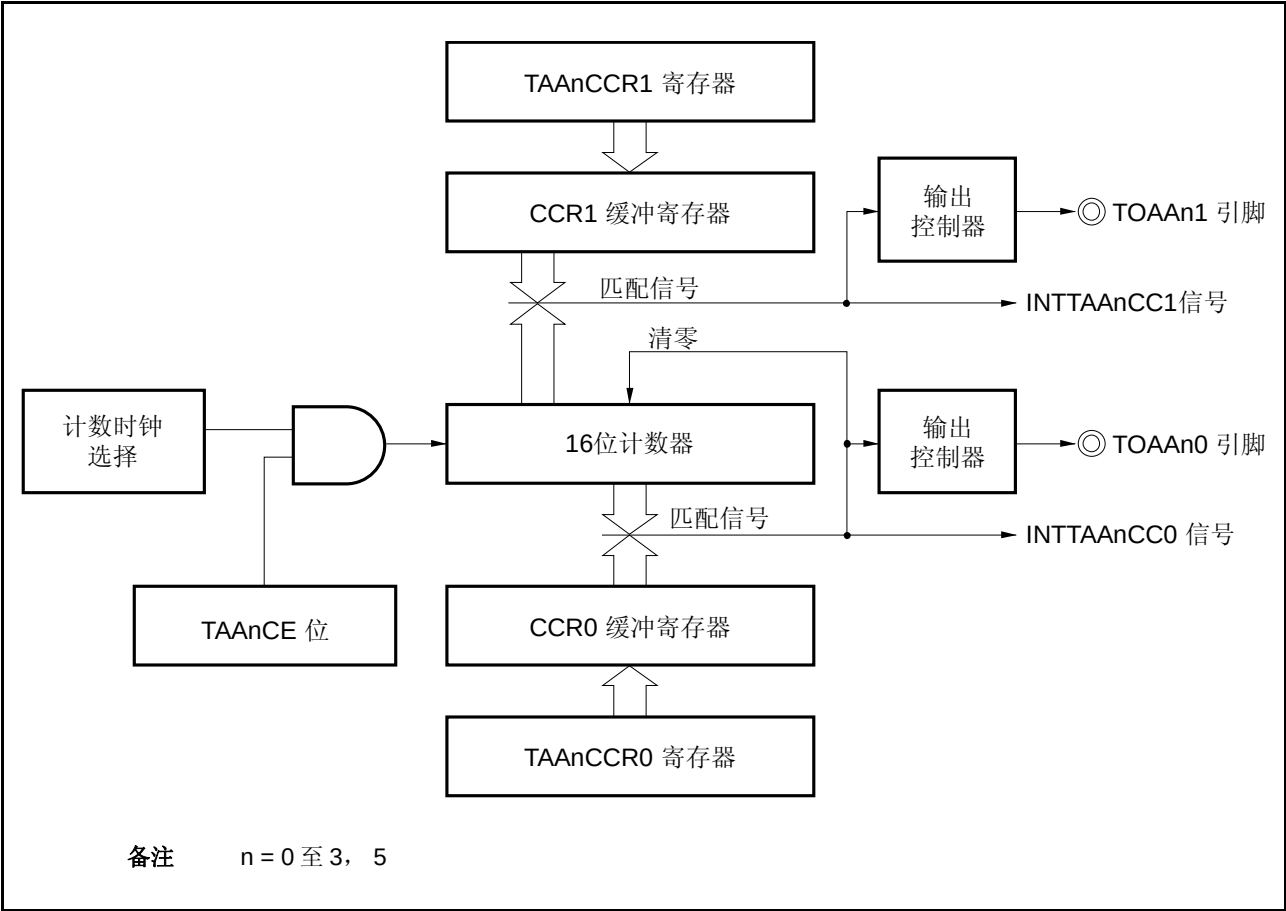
当 16 位计数器计数值大于 D_2 而小于 D_1 时，如果将 TAAAnCCR0 寄存器的值从 D_1 变为 D_2 ，则在重写 TAAAnCCR0 寄存器后，计数值会立即传送给 CCR0 缓冲寄存器。结果，用来比较的 16 位计数器的值是 D_2 。

但是，由于计数值已经超过了 D_2 ，所以 16 位计数器就会一直向上累积计数到 FFFFH 直到溢出，然后从 0000H 开始重新计数。当计数值和 D_2 相符时，会生成 INTTAAAnCC0 信号，且 TOAAm0 引脚输出信号反相。

因此，INTTAAAnCC0 信号可能不会在原先预计的间隔时间“ $(D_1 + 1) \times \text{计数时钟周期}$ ”或“ $(D_2 + 1) \times \text{计数时钟周期}$ ”下产生，而会以“ $(10000H + D_2 + 1) \times \text{计数时钟周期}$ ”的间隔产生”。

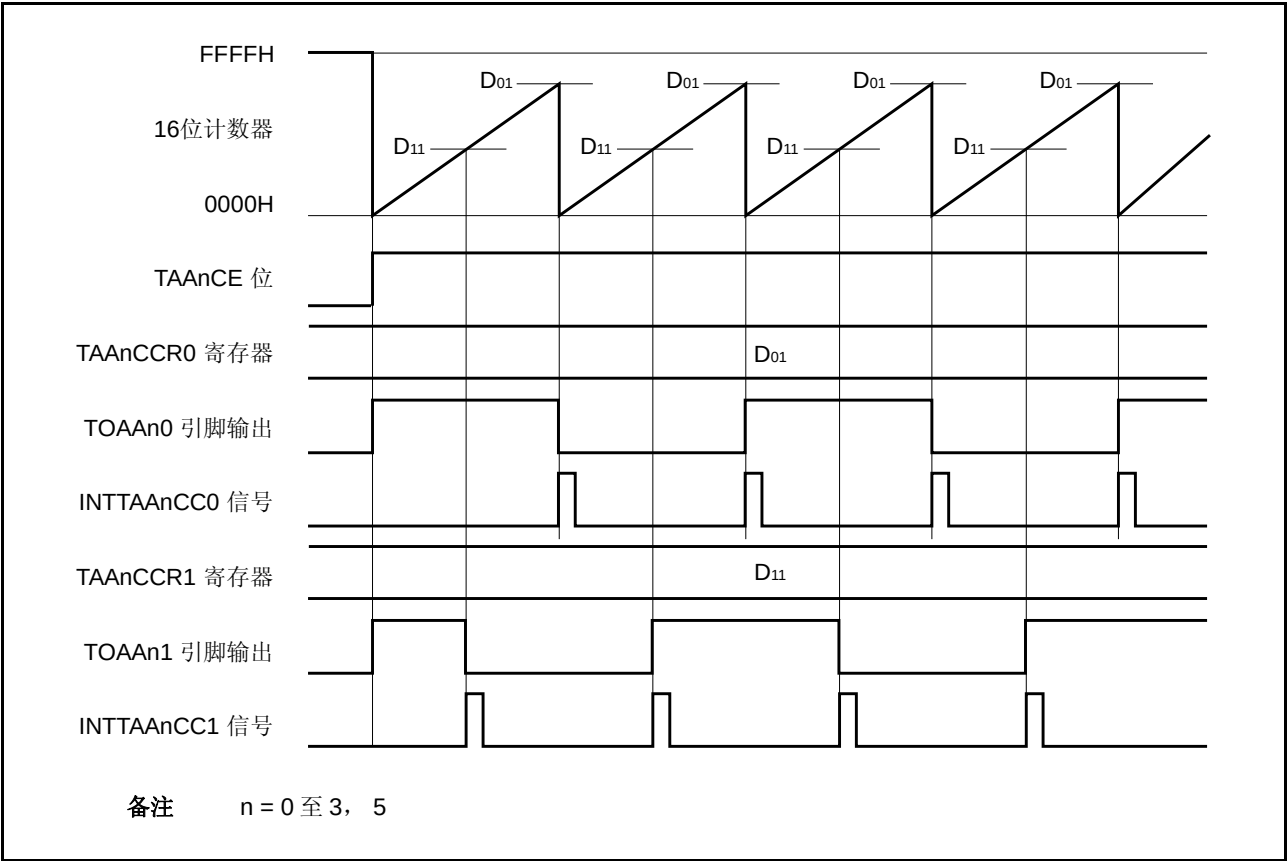
(d) TAAAnCCR1 寄存器的操作

图 7-11. TAAAnCCR1 寄存器的配置



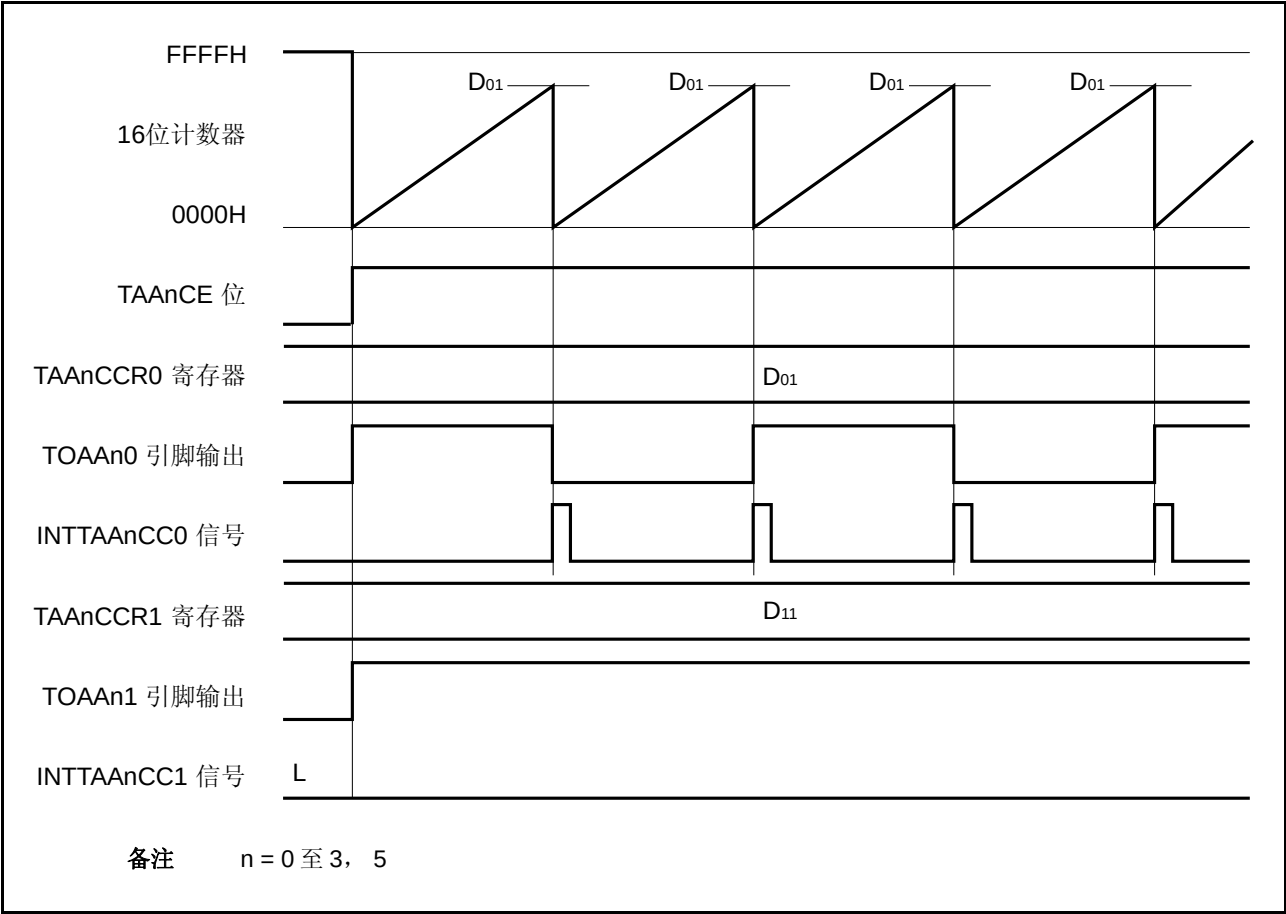
如果 TAAAnCCR1 寄存器的设定值小于 TAAAnCCR0 寄存器的设定值，则 INTTAAAnCC1 中断请求信号每个周期产生一次。此时，TOAAn1 引脚的输出反相。
TOAAn1 引脚输出一个与 TOAAn0 引脚输出周期相同的方波。

图 7-12. 当 $D_{01} \geq D_{11}$ 时的时序图



如果 TAAAnCCR1 寄存器的设定值大于 TAAAnCCR0 寄存器的设定值，16 位计数器的计数值和 TAAAnCCR1 寄存器的值不匹配。因而，不会产生 INTTAAAnCC1 信号，TOAAAn1 引脚的输出也不会改变。

图 7-13. 当 $D_{01} < D_{11}$ 时的时序图



7.5.2 外部事件计数器模式 (TAA_nMD2 至 TAA_nMD0 位=001)

在外部事件计数器模式下，当 TAA_nCTL0.TAA_nCE 位被设置为 1 时，外部事件计数输入信号的有效边沿被计数，每次指定数目的外部事件计数输入信号的有效边沿计数完成时，产生一个中断请求信号 (INTTAA_nCC0)。不使用 TAA_n0 引脚。

通常情况下，在外部事件计数器模式中不使用 TAA_nCCR1 寄存器。

图 7-14. 外部事件计数器模式的配置

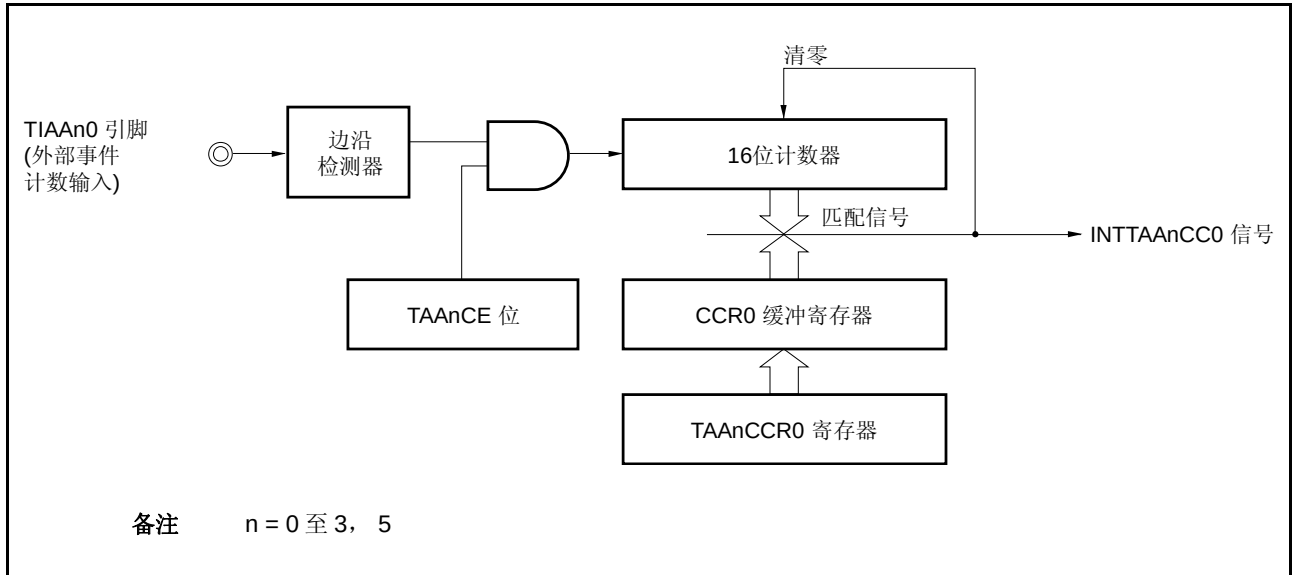
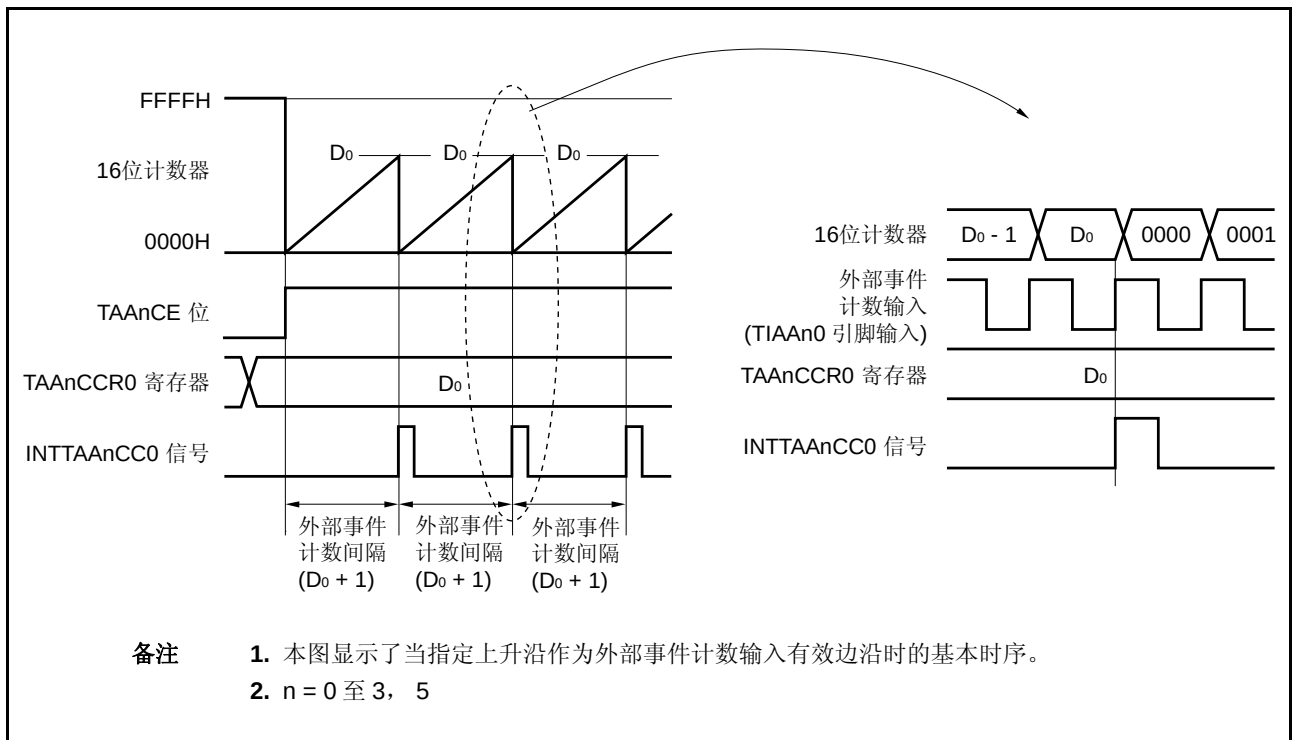


图 7-15. 外部事件计数器模式的基本时序



当 TAA_nCE 位设置为 1 时，16 位计数器的值由 FFFFH 被清为 0000H。计数器计数检测到的每一次外部事件计数输入的有效边沿。此外，TAA_nCCR0 寄存器的设定值传送给 CCR0 缓冲寄存器。

当 16 位计数器的计数值和 CCR0 缓冲寄存器的数值相匹配时，16 位计数器清零为 0000H，同时生成一个比较匹配中断请求信号（INTTAA_nCC0）。

每当外部事件计数输入的有效边沿检测计数达到（TAA_nCCR0 寄存器设定值+1）时，就会生成 INTTAA_nCC0（比较匹配中断请求）信号。

图 7-16. 外部事件计数模式下的寄存器操作设置 (1/2)

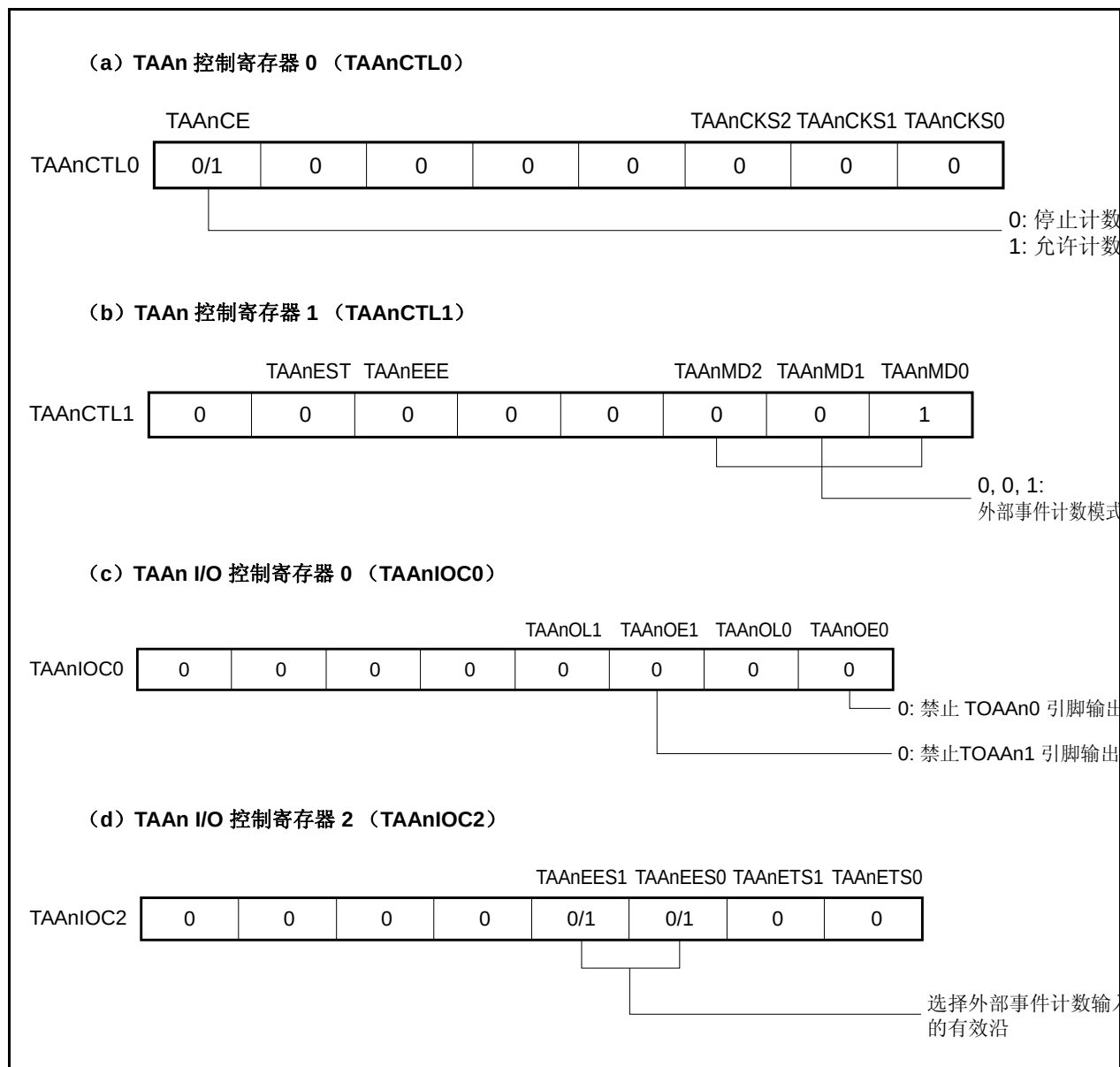


图 7-16. 外部事件计数模式下的寄存器操作设置 (2/2)

(e) TAA_n 计数器读取缓冲寄存器 (TAA_nCNT)

可以通过读取 TAA_nCNT 寄存器，读取 16 位计数器的计数值。

(f) TAA_n 捕获/比较寄存器 0 (TAA_nCCR0)

如果 TAA_nCCR0 寄存器的值被设定为 D₀，当外部事件计数值达到 (D₀ + 1) 时，计数器被清 0，同时产生一个比较匹配中断请求信号 (INTTAA_nCC0)。

(g) TAA_n 捕获/比较寄存器 1 (TAA_nCCR1)

通常情况下，在外部事件计数器模式中不使用 TAA_nCCR1 寄存器。但是，TAA_nCCR1 寄存器的设定值传送给 CCR1 缓冲寄存器。当 16 位计数器的计数值和 CCR1 缓冲寄存器的值匹配时，也会产生一个比较匹配中断请求信号 (INTTAA_nCC1)。

因此，可以使用中断屏蔽标志 (TAA_nCCMK1) 来屏蔽中断信号。

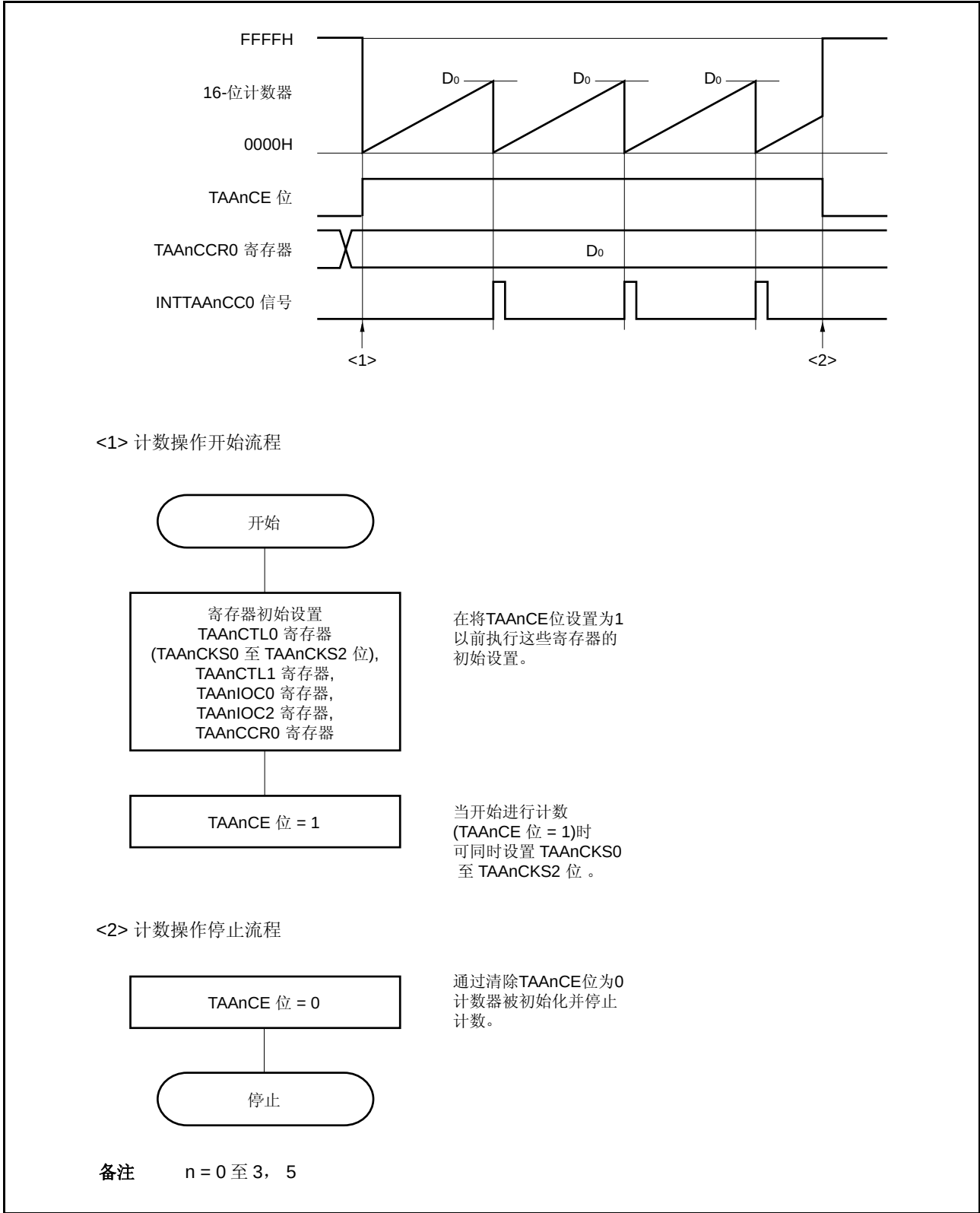
注意事项 当外部时钟用作计数时钟时，外部时钟只能从 TIAA_n0 引脚输入。此时，把 TAA_nIOC1.TAA_nIS1 和 TAA_nIOC1.TAA_nIS0 位清零为 00 (捕获触发输入 (TIAA_n0 引脚)：无边沿检测)。

备注

1. 在外部事件计数模式下，不使用 TAA_n 输入/输出控制寄存器 1 (TAA_nIOC1) 和 TAA_n 选项寄存器 0 (TAA_nOPT0)。
2. n = 0 至 3, 5

(1) 外部事件计数模式的操作系统

图 7-17. 外部事件计数模式下的软件处理流程

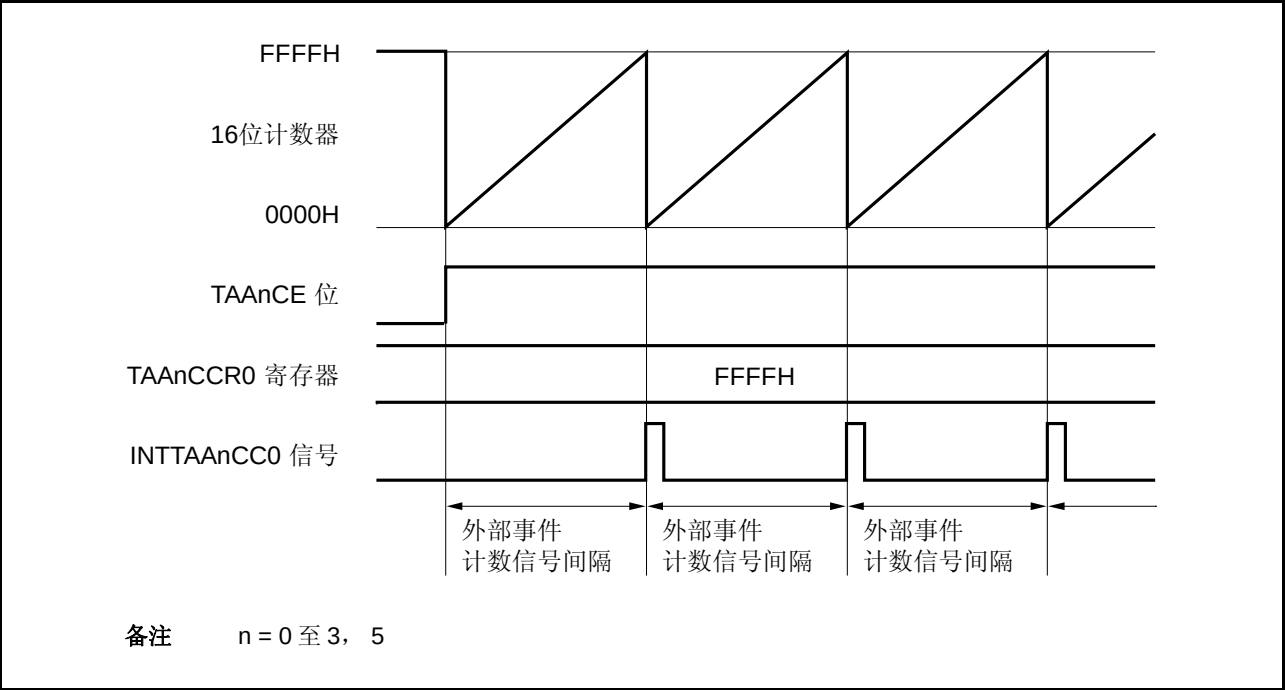


(2) 外部事件计数模式的工作时序

- 注意事项
- 1. 外部事件计数模式下，不要将 TAA_nCCR0 寄存器设置为 0000H。
 - 2. 外部事件计数模式下，禁止使用定时器输出。如果在外部事件计数模式下使用定时器输出，应设置为间隔定时器模式，并且通过外部事件计数输入选择计数时钟的操作（TAA_nCTL1.TAA_nMD2 至 TAA_nCTL1.TAA_nMD0 位=000，TAA_nCTL1.TAA_nEEE 位= 1）。

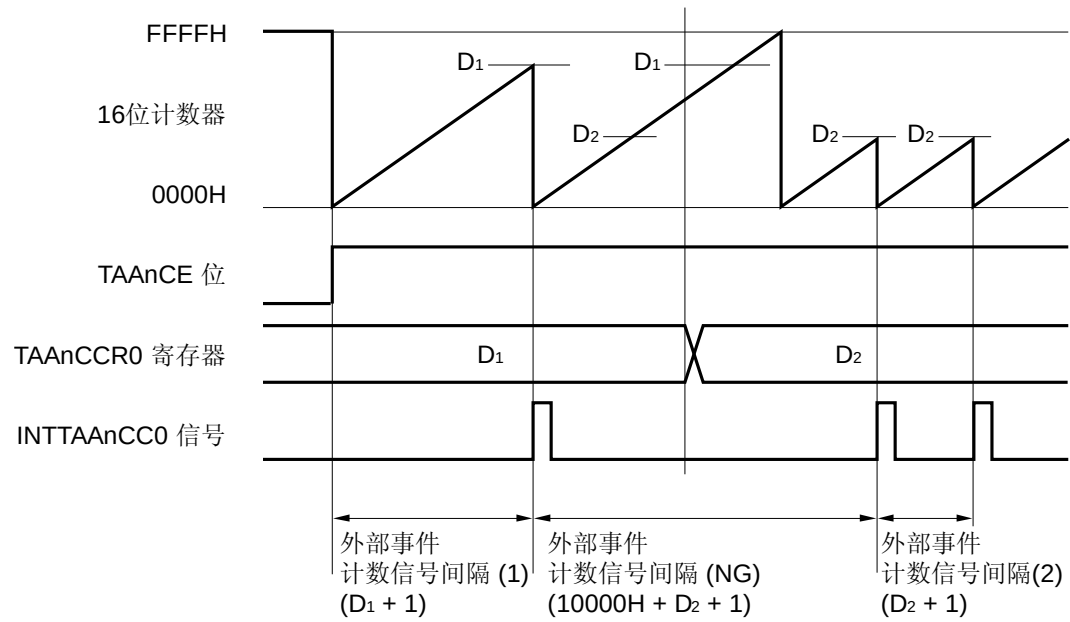
(a) TAA_nCCR0 寄存器被设置为 FFFFH 时的操作

如果 TAA_nCCR0 寄存器被设置为 FFFFH，16 位计数器在每次检测外部事件计数信号的有效边沿时，都计数到 FFFFH。当下一个计数到来时 16 位计数器同步清为 0000H，并且产生 INTTAA_nCC0 信号。此时，不设置 TAA_nOPT0.TAA_nOVF 位。



(b) 有关重写 TAAAnCCR0 寄存器的注意事项

要将 TAAAnCCR0 寄存器的值变为比当前值较小的值，应停止计数，然后改变其设定值。
如果在计数器进行计数期间，对寄存器 TAAAnCCR0 的值进行重写为较小的数值，则 16 位计数器可能发生溢出。



备注 n = 0 至 3, 5

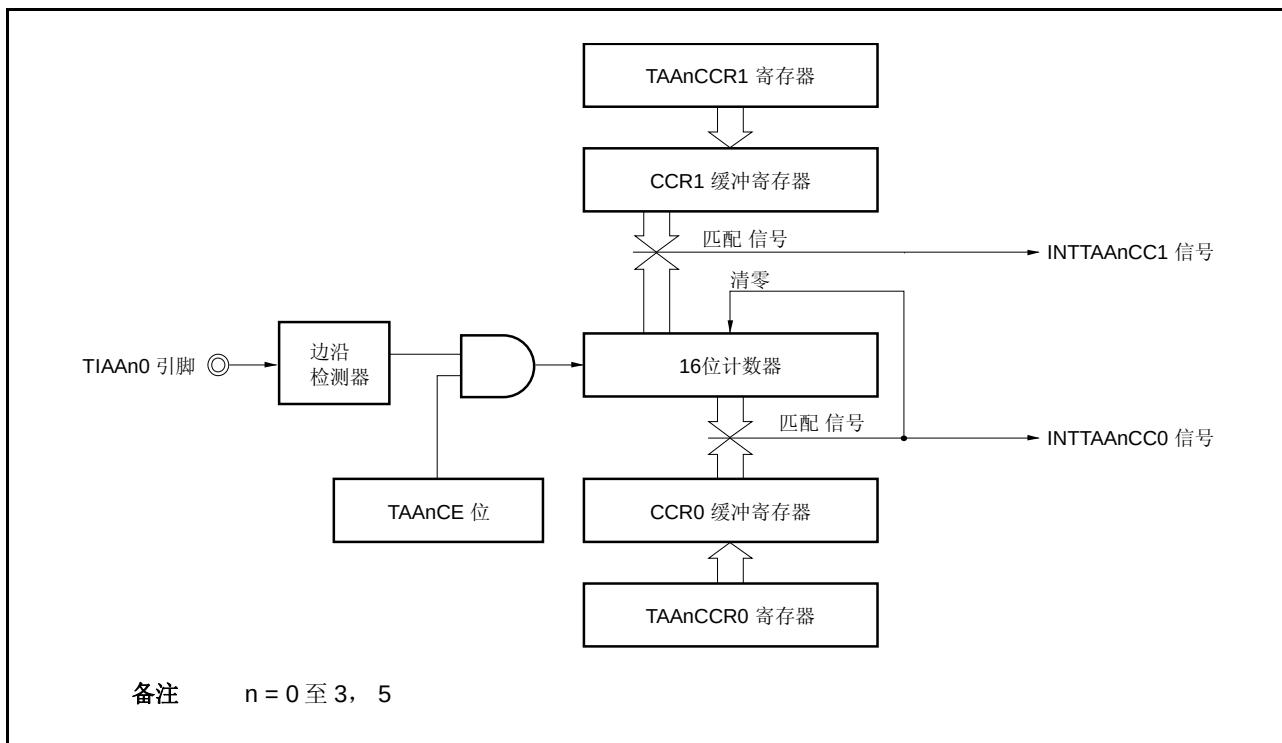
当 16 位计数器计数值大于 D_2 而小于 D_1 时，如果将 TAAAnCCR0 寄存器的值从 D_1 变为 D_2 ，则在重写 TAAAnCCR0 寄存器后，计数值会立即传送给 CCR0 缓冲寄存器。因而，与 16 位计数器相比较的值会是 D_2 。

但是，由于计数值已经超过了 D_2 ，所以 16 位计数器就会一直向上累积计数到 FFFFH 直到溢出，然后从 0000H 开始重新计数。当计数值与 D_2 匹配时，会生成 INTTAAAnCC0 信号。

因此，INTTAAAnCC0 中断请求信号可能不会在原先预计的“ $(D_1 + 1)$ ”次或“ $(D_2 + 1)$ ”次有效边沿上生成，而可能在“ $(10000H + D_2 + 1)$ ”次有效边沿上生成。

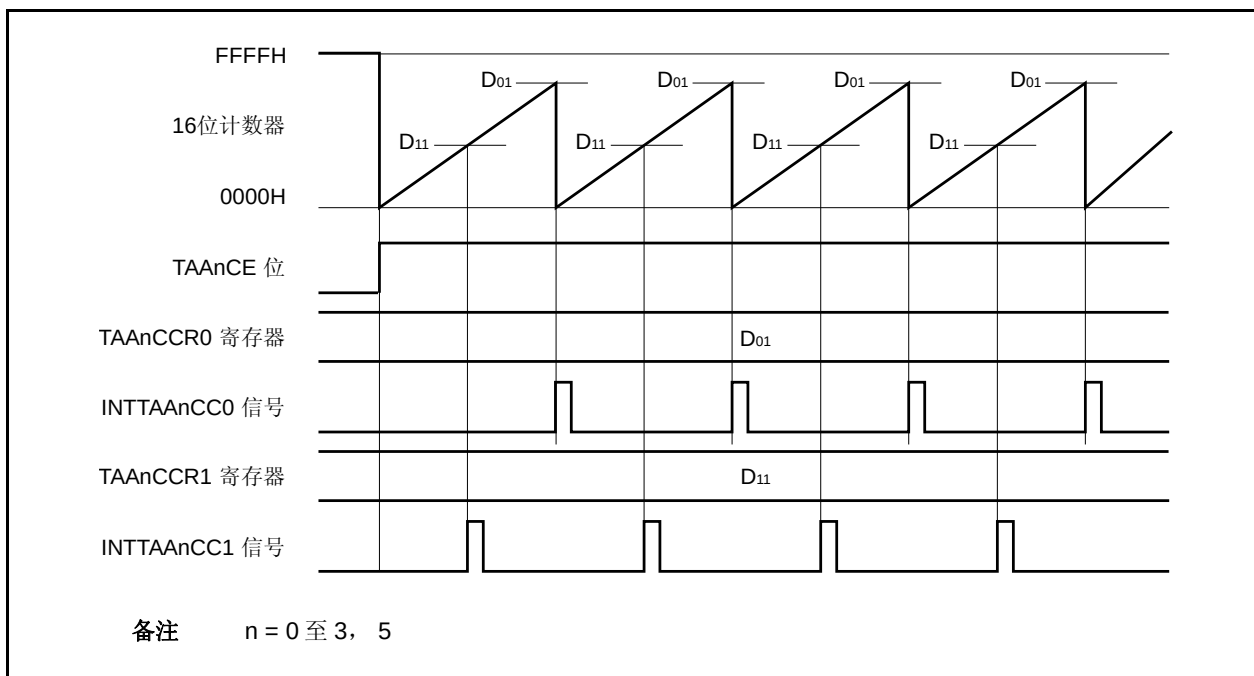
(c) TAAAnCCR1 寄存器的操作

图 7-18. TAAAnCCR1 寄存器的配置



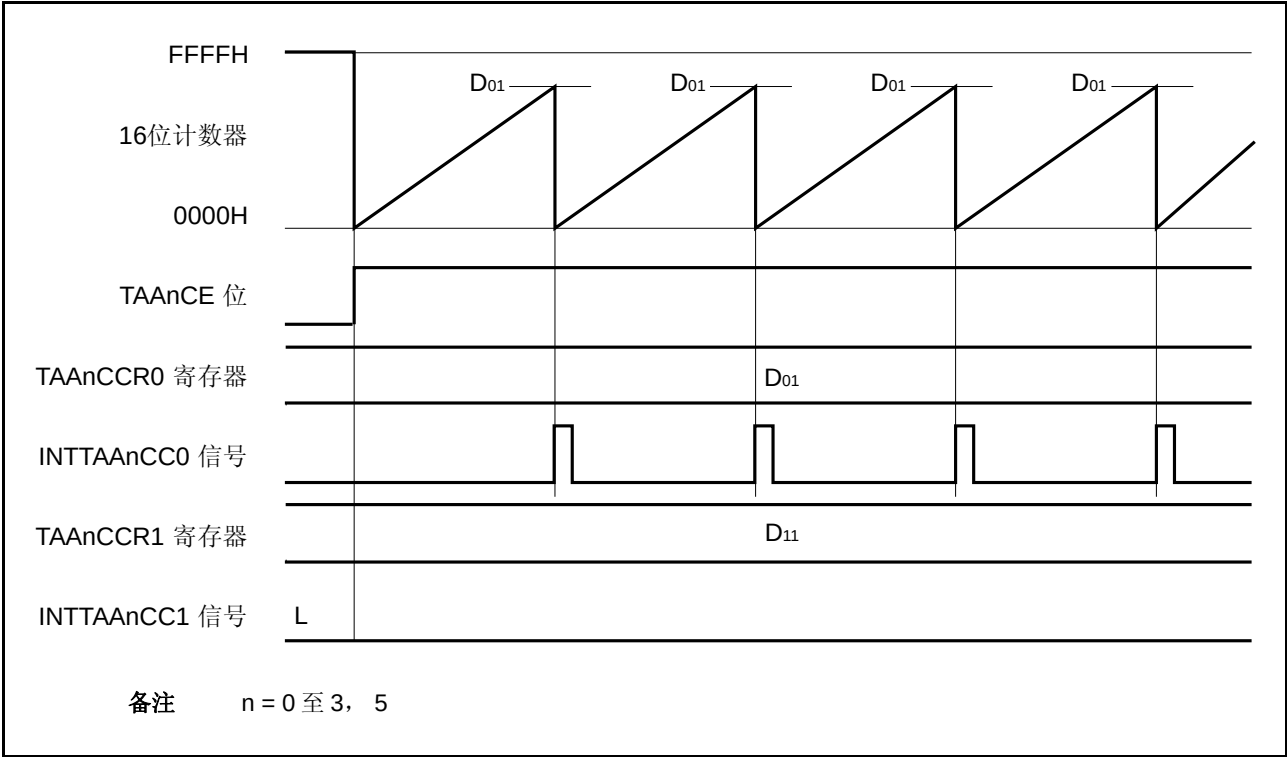
如果 TAAAnCCR1 寄存器的设定值小于 TAAAnCCR0 寄存器的设定值，则 INTTAAAnCC1 中断请求信号每个周期都产生一次。

图 7-19. 当 $D_{01} \geq D_{11}$ 时的时序图



如果 TAAAnCCR1 寄存器的设定值大于 TAAAnCCR0 寄存器的设定值，则不会产生 INTTAAAnCC1 中断请求信号，因为 16 位计数器的计数值和 TAAAnCCR1 寄存器的值不匹配。

图 7-20. 当 $D_{01} < D_{11}$ 时的时序图



7.5.3 外部触发脉冲输出模式 (TAA_nMD2 至 TAA_nMD0 位=010)

在外部触发脉冲输出模式下，当 TAA_nCTL0.TAA_nCE 位被设置为 1 时，16 位定时器/事件计数器 AA 等待触发，当检测到外部触发输入信号的有效边沿后，16 位定时器/事件计数器 AA 开始计数，同时从 TOA_n1 引脚输出一个脉宽调制信号波形。

不使用外部触发，而是使用软件触发，同样可以得到输出脉冲。当使用软件触发时，TOA_n0 引脚输出一个方波，其半周期为输出脉宽调制信号的一个周期。

图 7-21. 外部触发脉冲输出模式的配置

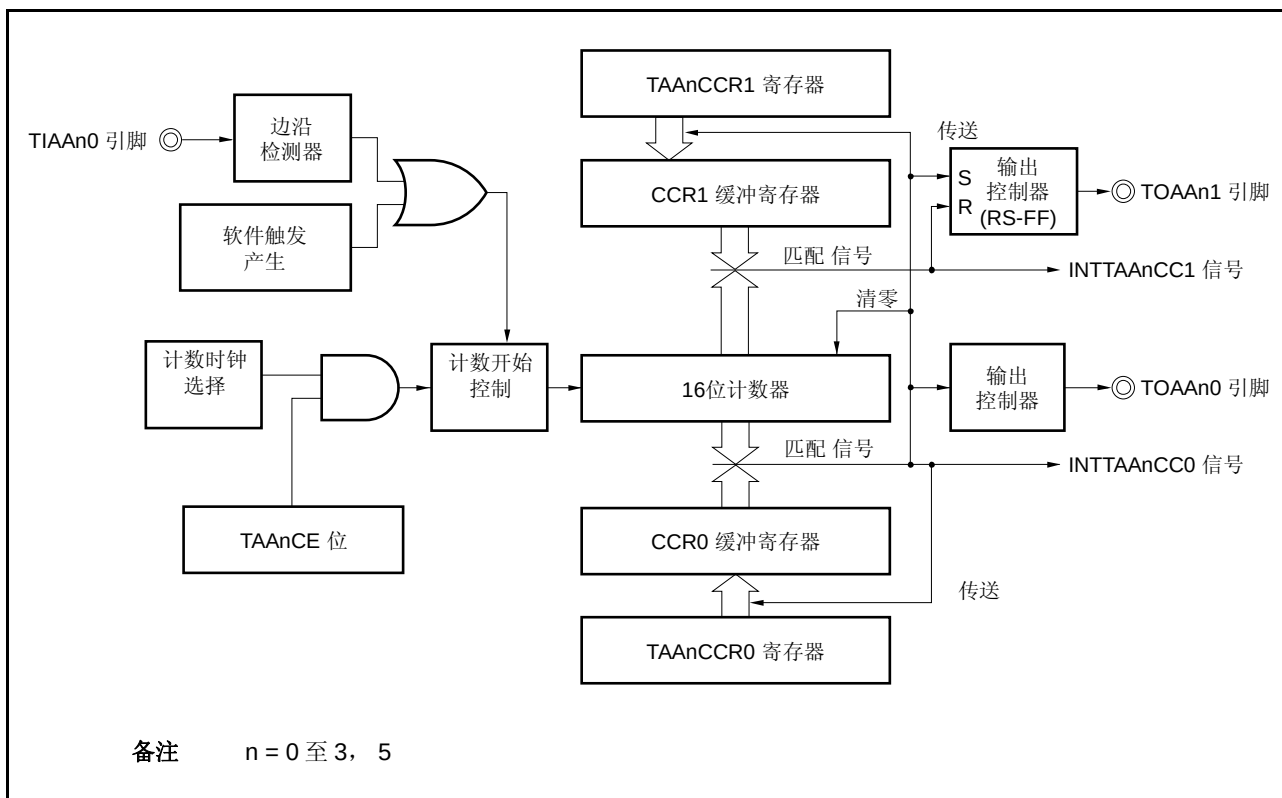
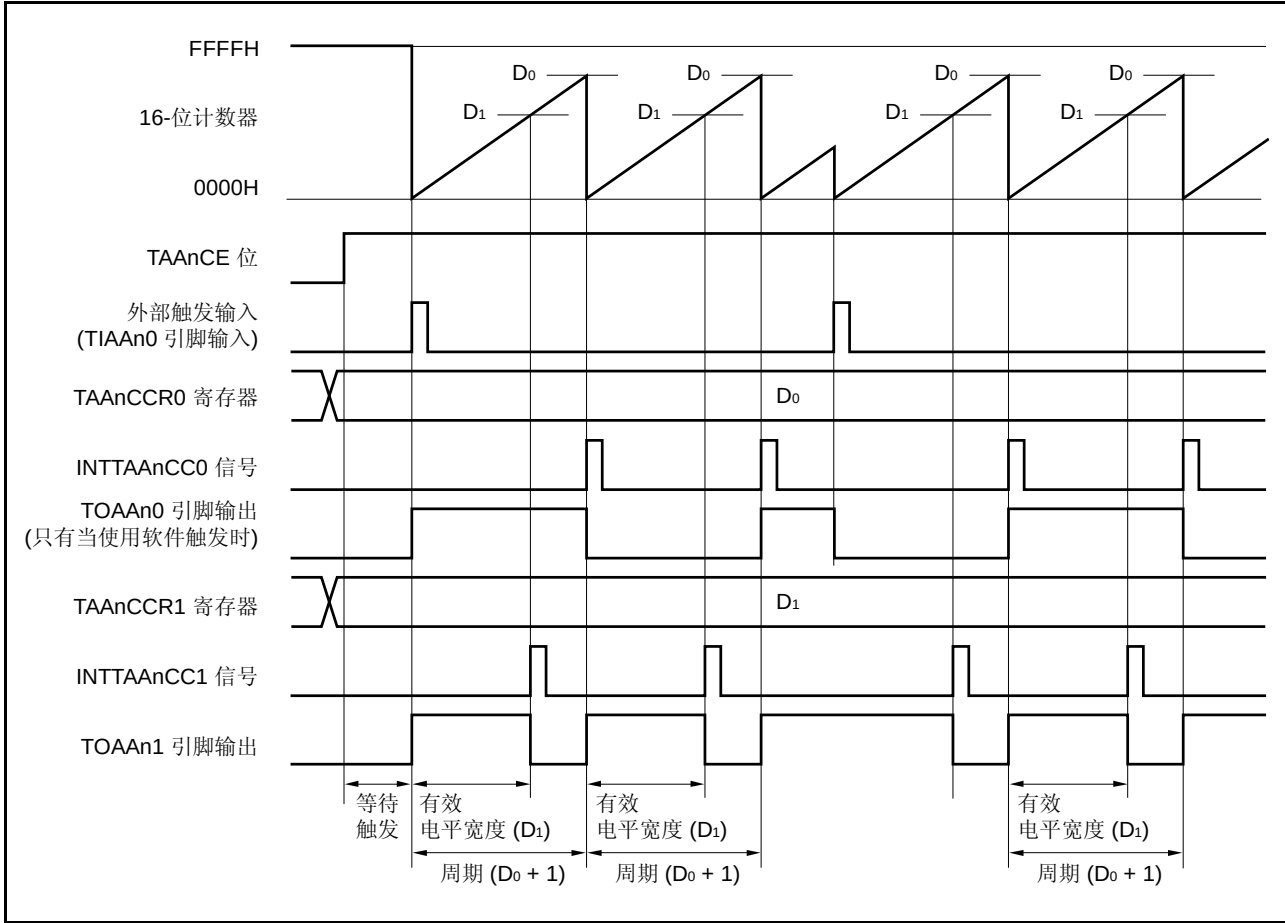


图 7-22. 外部触发脉冲输出模式的基本时序



当 TAAAnCE 位设置为 1 时，16 位定时器/事件计数器 AA 等待触发。当触发产生时，16 位计数器从 FFFFH 清零为 0000H，同时开始计数，并从 TOAAAn1 引脚输出一个脉宽调制信号波形。若在计数运行中再次产生触发，则计数器清零为 0000H 并重新开始计数。（TOAAAn0 引脚输出反转。在出现触发时，不管当前状态如何（高/低），TOAAAn1 引脚都会输出一个高电平。）

脉冲宽度调制信号波形的有效电平宽度、周期、占空比计算如下。

$$\begin{aligned} \text{有效电平宽度} &= (\text{TAAAnCCR1 寄存器的设定值}) \times \text{计数时钟周期} \\ \text{周期} &= (\text{TAAAnCCR0 寄存器的设定值} + 1) \times \text{计数时钟周期} \\ \text{占空比} &= (\text{TAAAnCCR1 寄存器的设定值}) / (\text{TAAAnCCR0 寄存器的设定值} + 1) \end{aligned}$$

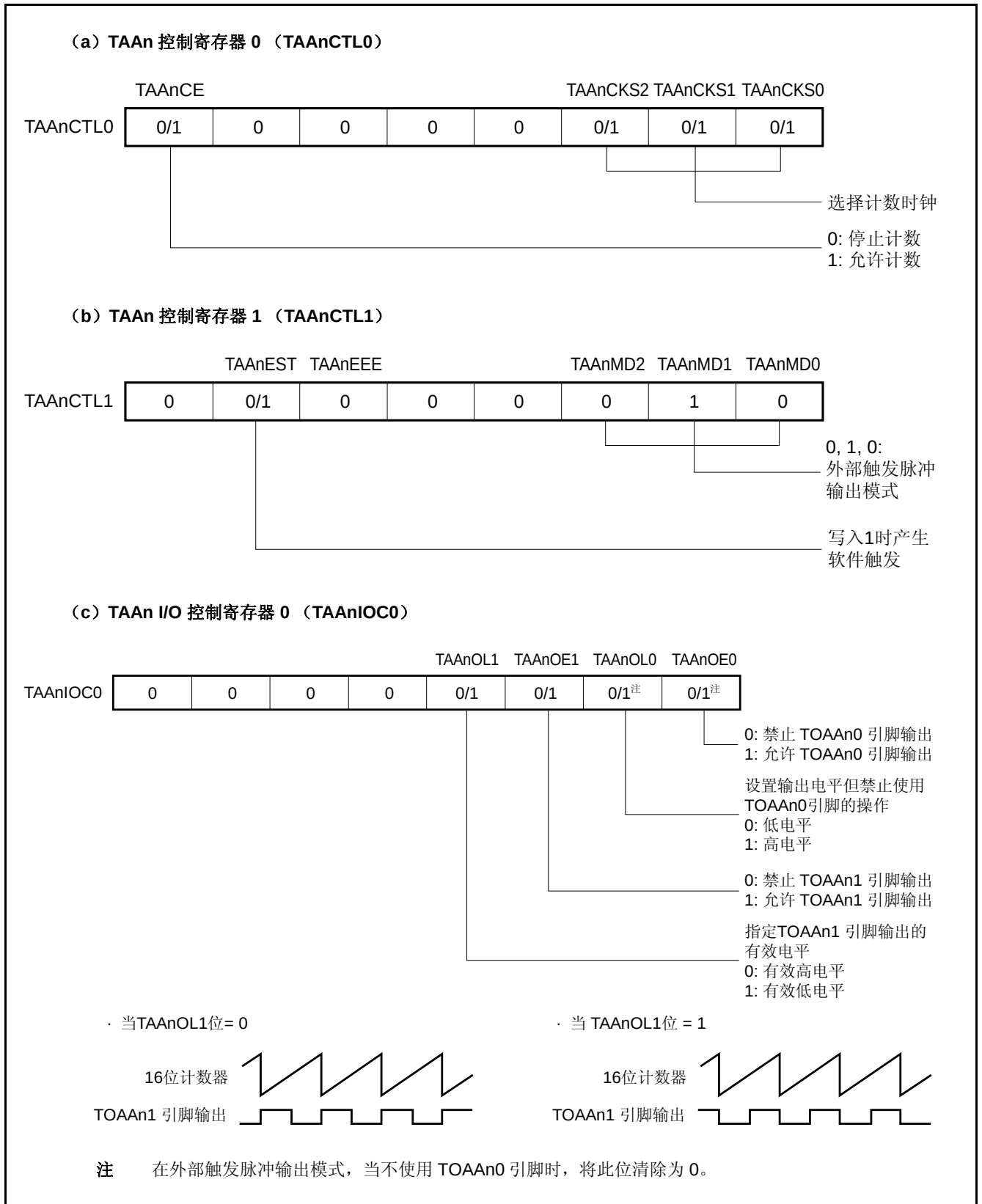
当 16 位计数器在其计数值与 CCR0 缓冲寄存器的值匹配之后进行下一次向上计数时，会同步产生比较匹配中断请求信号 INTTAAAnCC0，同时 16 位计数器清零为 0000H。当 16 位计数器的计数值与 CCR1 缓冲寄存器的值匹配时，会生成比较匹配中断请求信号 INTTAAAnCC1。

当 16 位计数器的计数值与 CCRm 缓冲寄存器的值匹配，而且 16 位计数器清零为 0000H 时，TAAAnCCRm 寄存器设置的值就传送到 CCRm 缓冲寄存器。

外部触发输入信号的有效边沿，或者将软件触发（TAAAnCTL1.TAAAnEST 位）设为 1，都可当作触发信号。

备注 n = 0 至 3, 5
 m = 0, 1

图 7-23. 外部触发脉冲输出模式的寄存器设置 (1/2)



(1) 外部触发脉冲输出模式的工作流程

图 7-24. 外部触发脉冲输出模式的软件处理流程 (1/2)

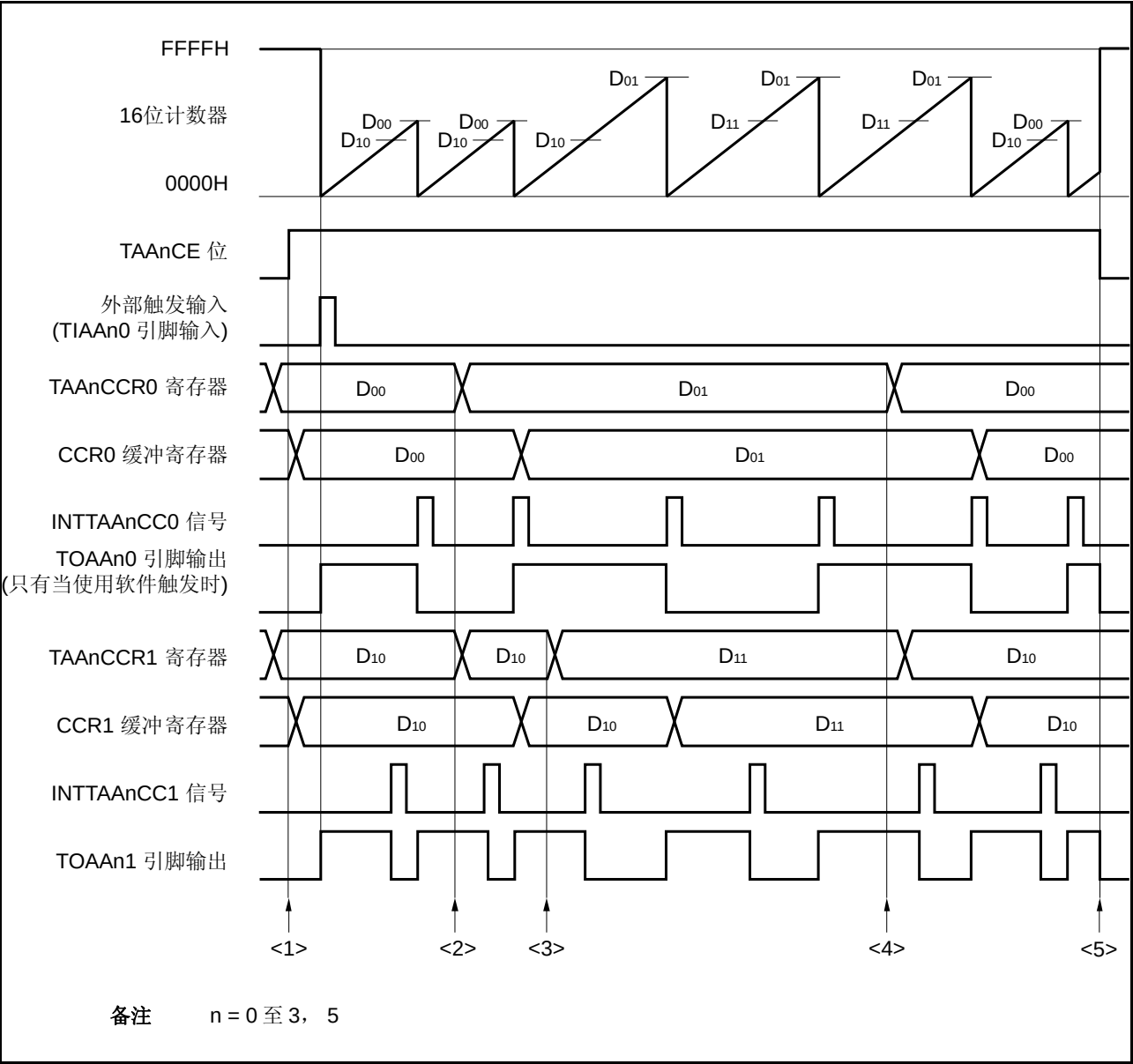
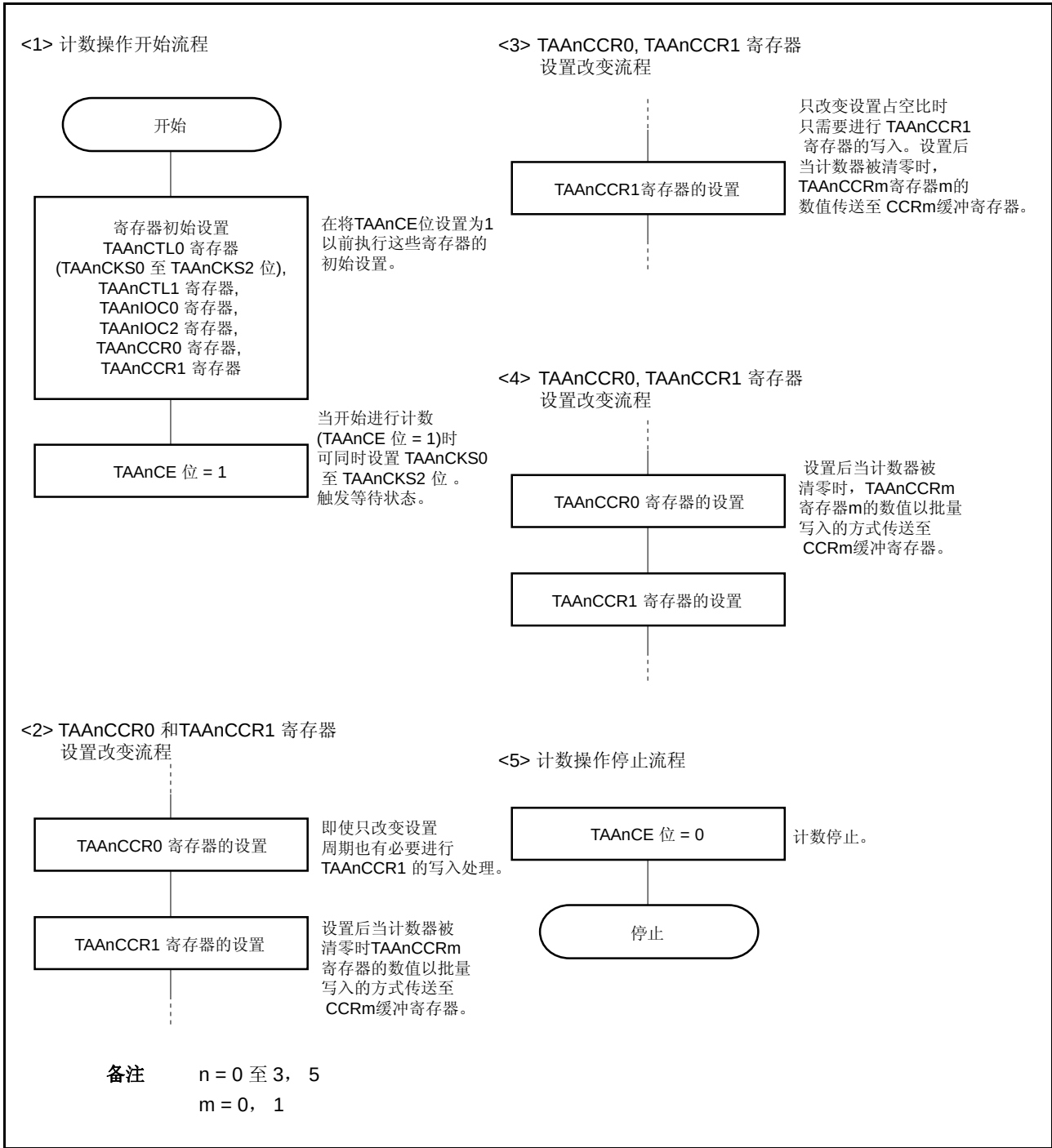


图 7-24. 外部触发脉冲输出模式的软件处理流程 (2/2)

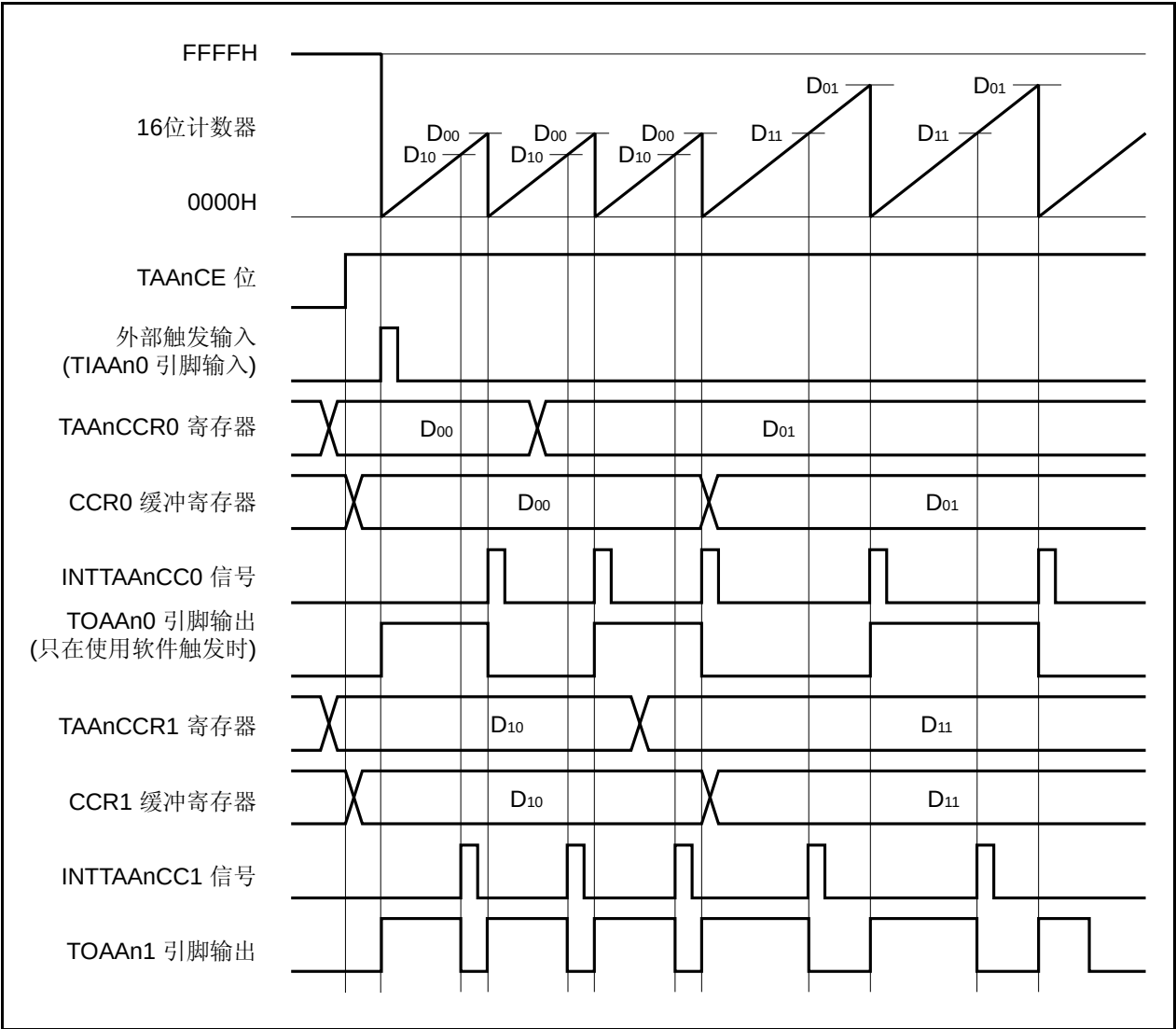


(2) 外部触发脉冲输出模式的操作时序

(a) 有关操作中改变脉冲宽度的注意事项

计数器计数过程中，若要改变脉宽调制信号波形，应最后写入 TAAAnCCR1 寄存器。

检测到 INTTAAAnCC0 信号以后，在写入 TAAAnCCR1 寄存器之后重写 TAAAnCCRm 寄存器。



若要将数据从 TAAAnCCRm 寄存器传送至 CCRm 缓冲寄存器，则必须写入 TAAAnCCR1 寄存器。

此时要改变脉宽调制信号波形的周期和有效电平宽度，应首先将周期设定给 TAAAnCCR0 寄存器，然后将有效电平宽度设定给 TAAAnCCR1 寄存器。

如果只改变脉宽调制信号波形的周期，应首先把周期设置给 TAAAnCCR0 寄存器，然后将相同值（原值）写入 TAAAnCCR1 寄存器。

只改变脉宽调制信号波形的有效电平宽度（占空比）时，那就只需要设置 TAAAnCCR1 寄存器。

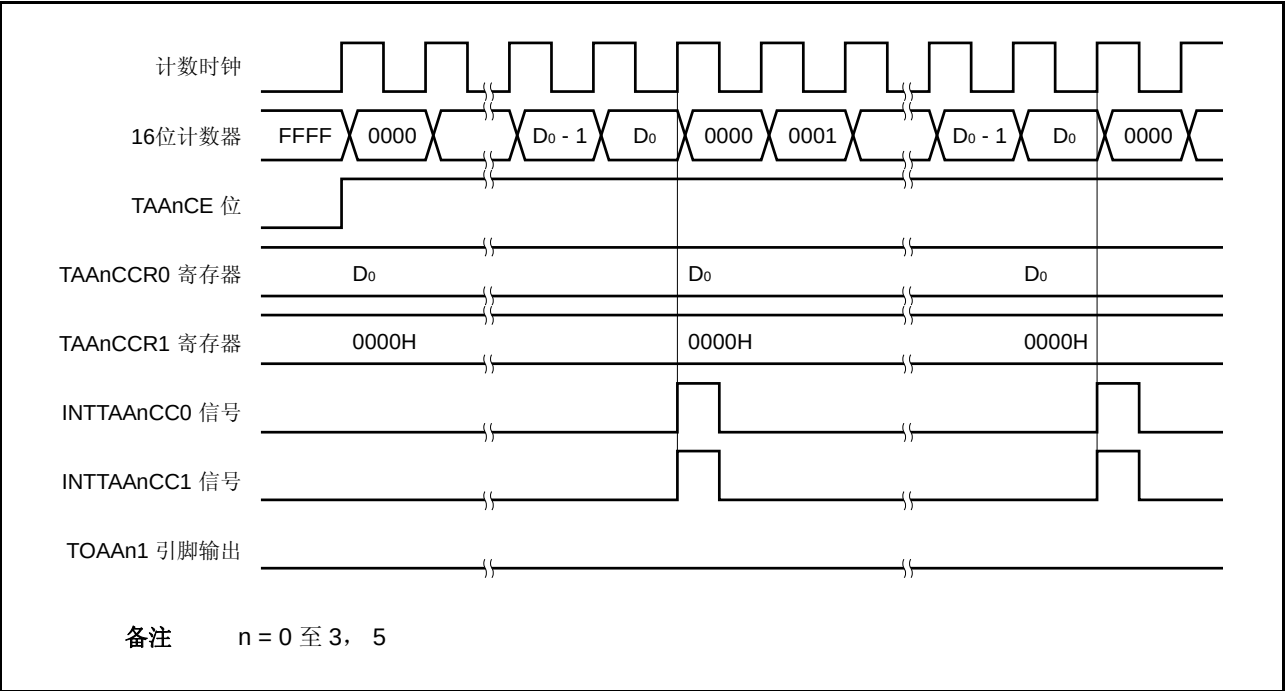
在数据写入 TAAAnCCR1 寄存器后，写入 TAAAnCCRm 寄存器的值会传送到 CCRm 缓冲寄存器，与 16 位计数器清零同步，并作为与 16 位计数器进行比较的值。

要在写入一次 TAAAnCCR1 寄存器之后，再次写 TAAAnCCR0 或 TAAAnCCR1 寄存器，应在 INTTAAAnCC0 信号产生之后进行。否则，CCRm 缓冲寄存器的值可能不确定，因为把数据从 TAAAnCCRm 寄存器传送到 CCRm 缓冲寄存器的时序与写 TAAAnCCRm 寄存器时序发生冲突。

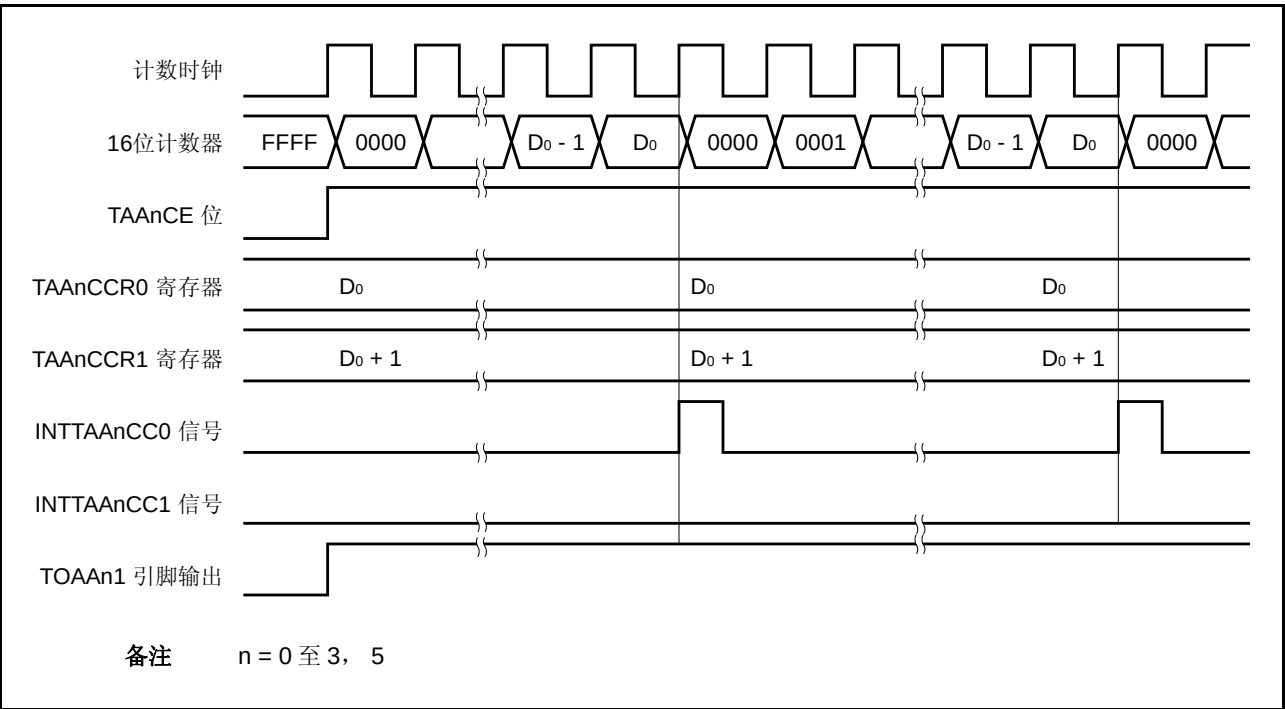
备注 n = 0 至 3, 5
 m = 0, 1

(b) 脉宽调制信号波形的 0%和 100% 输出

为了输出 0% 的波形，需要将 TAAAnCCR1 寄存器设置为 0000H。如果 TAAAnCCR0 寄存器的设定值为 FFFFH，则会周期性产生 INTTAAAnCC1 中断请求信号。

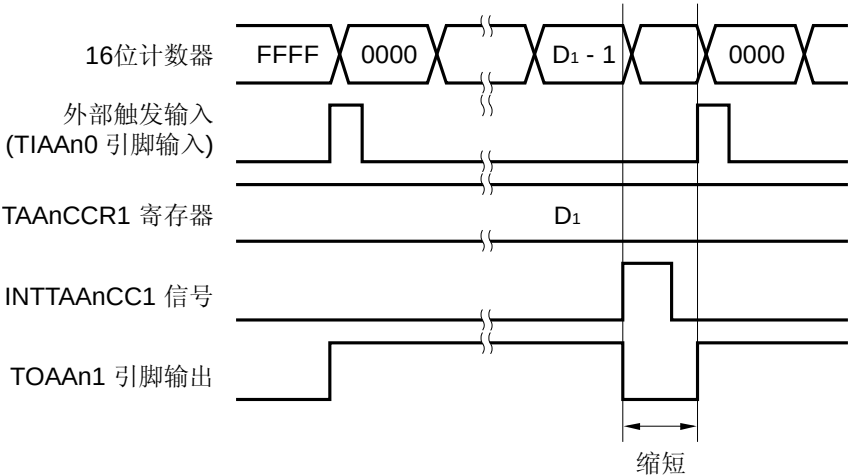


要输出 100%的波形，需给 TAAAnCCR1 寄存器设置一个（TAAAnCCR0 寄存器设定值 + 1）的数值。如果 TAAAnCCR0 寄存器的设定值为 FFFFH，则不会产生 100%波形输出。



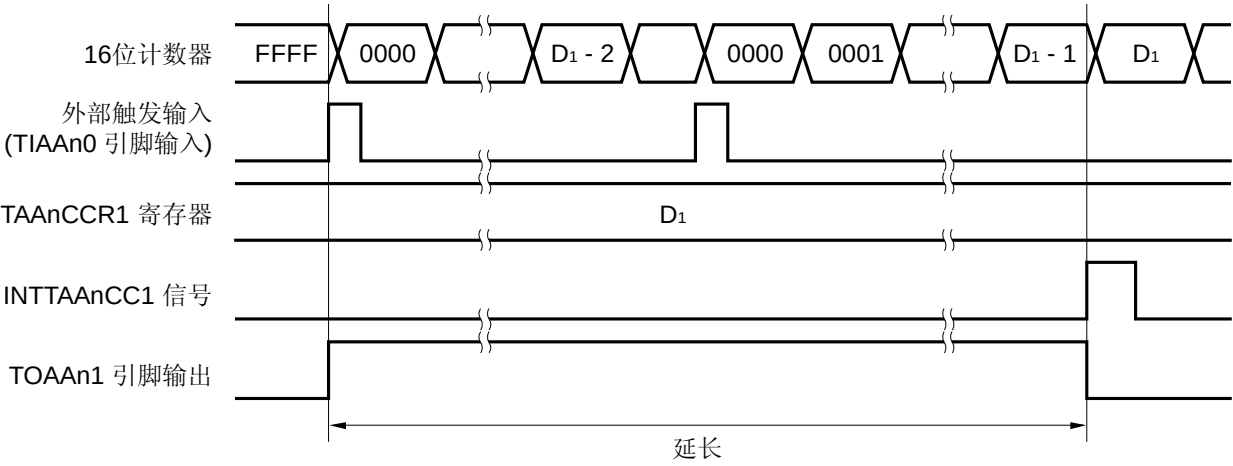
(c) 触发检测和与 TAAAnCCR1 寄存器匹配之间的冲突

如果在 INTTAAAnCC1 信号产生后，立刻检测到触发信号，则 16 位计数器立刻清零为 0000H，TOAAn1 引脚的输出信号被置位生效，计数器继续计数。结果，缩短了脉宽调制信号波形的无效时段。



备注 n = 0 至 3, 5

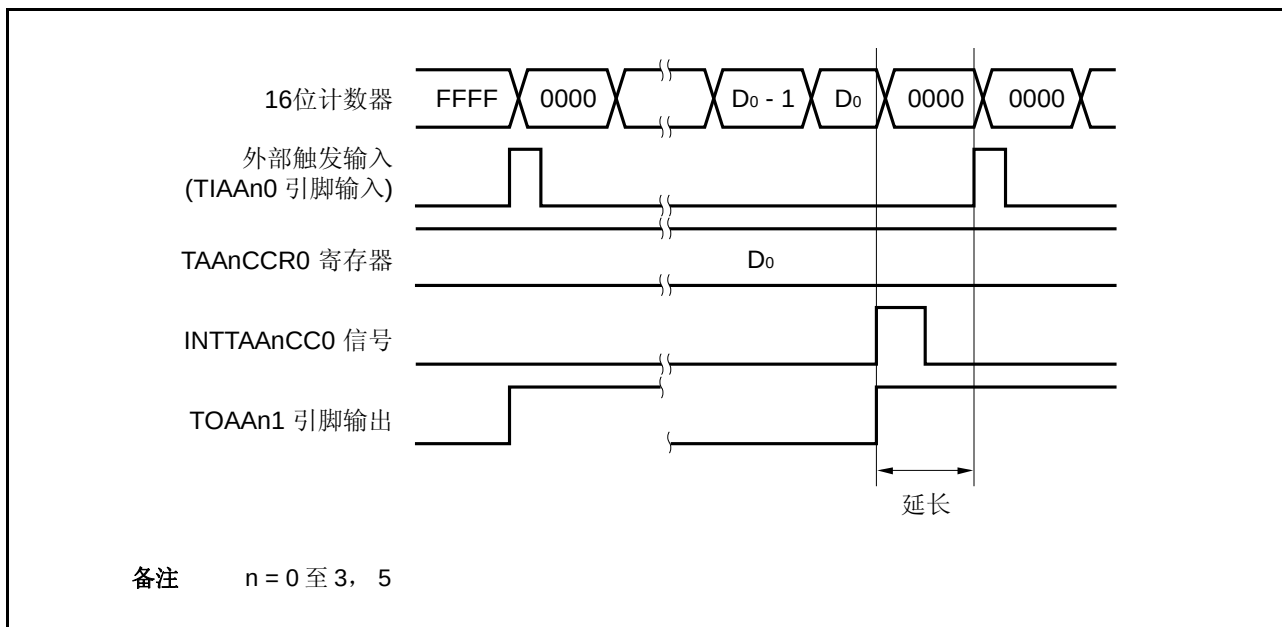
如果刚好在 INTTAAAnCC1 信号产生前检测到触发信号，则不会产生 INTTAAAnCC1 信号，而 16 位计数器清零为 0000H，计数继续进行。TOAAn1 引脚的输出信号仍然保持有效。结果，延长了脉宽调制信号波形的有效时域。



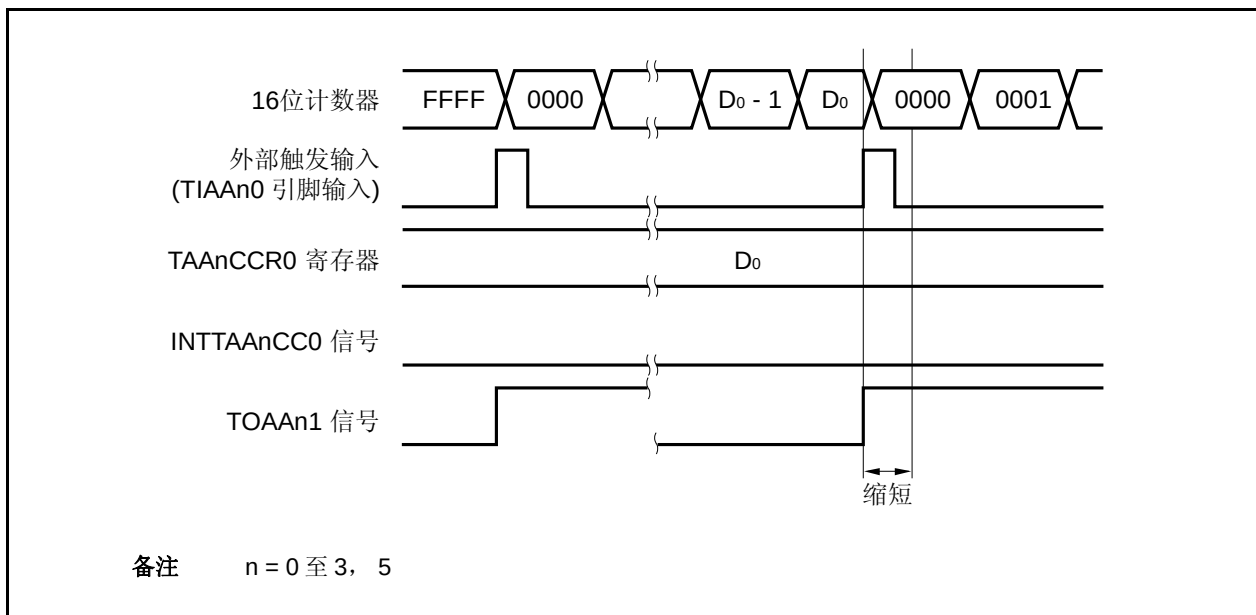
备注 n = 0 至 3, 5

(d) 触发检测与与 TAAAnCCR0 寄存器匹配之间的冲突

如果刚好在 INTTAAAnCC0 信号产生后检测到触发，则 16 位计数器清零为 0000H，并继续计数。因此，TOAAAn1 引脚的有效时段被延长，延长时间为自 INTTAAAnCC0 信号产生时刻到触发被检测到时刻。

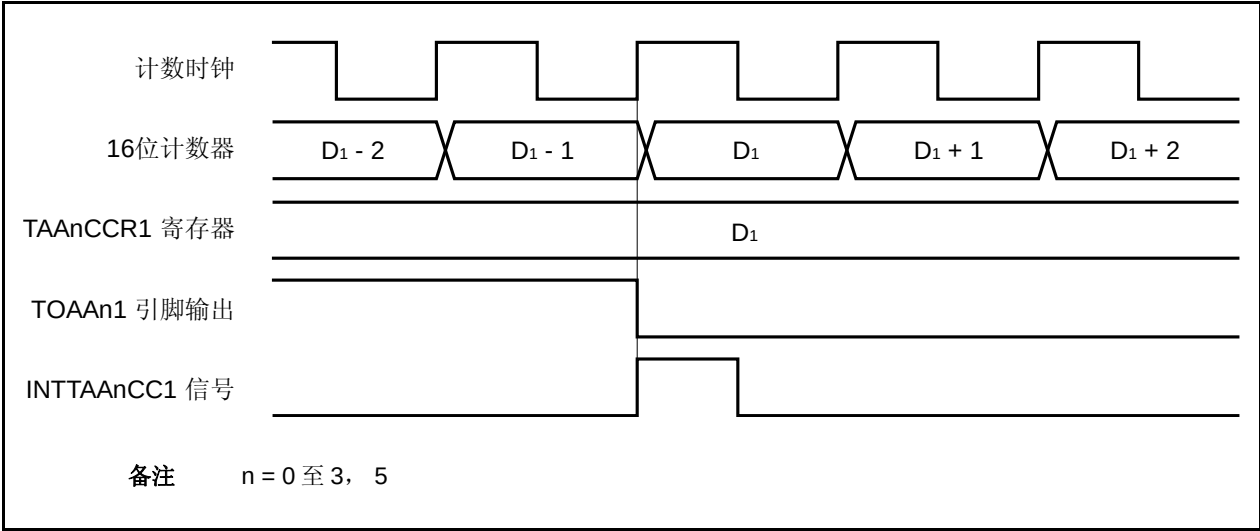


如果刚好在产生 INTTAAAnCC0 信号之前检测到触发信号，则不会产生 INTTAAAnCC0 信号。16 位计数器清零为 0000H，TOAAAn1 引脚输出被置位，计数器继续计数。结果，缩短了脉宽调制信号波形的无效时段。



(e) 比较匹配中断请求信号 (INTTAAAnCC1) 的发生时序

INTTAAAnCC1 信号在外部触发脉冲输出模式下的发生时序与其它 INTTAAAnCC1 信号的时序不同；当 16 位计数器的计数值与 TAAAnCCR1 寄存器的值匹配时，在外部触发脉冲输出模式下产生 INTTAAAnCC1 中断请求信号。



通常情况下，在 16 位计数器的当前计数值与 TAAAnCCR1 寄存器的值匹配之后，INTTAAAnCC1 信号与下一次计数同步发生。

但是，在外部触发脉冲输出模式下，其产生会提早一个计数时钟周期。这是由于为了匹配 TOAAAn1 引脚输出信号的变化，时序发生了变化。

7.5.4 单次脉冲输出模式 (TAA_nMD2 至 TAA_nMD0 位 = 011)

在单次脉冲输出模式下，当 TAA_nCTL0.TAA_nCE 位被置为 1 时，16 位定时器/事件计数器 AA 等待触发。在检测到外部触发输入的有效边沿时，16 位定时器/事件计数器 AA 开始计数，同时从 TOAAn1 引脚输出一个单次脉冲。

除外部触发外，也可以生成软件触发来输出该脉冲。在使用软件触发时，TOAAn0 引脚在 16 位计数器计数时输出有效电平，在计数器停止（等待触发）时输出无效电平。

图 7-25. 单次脉冲输出模式的配置

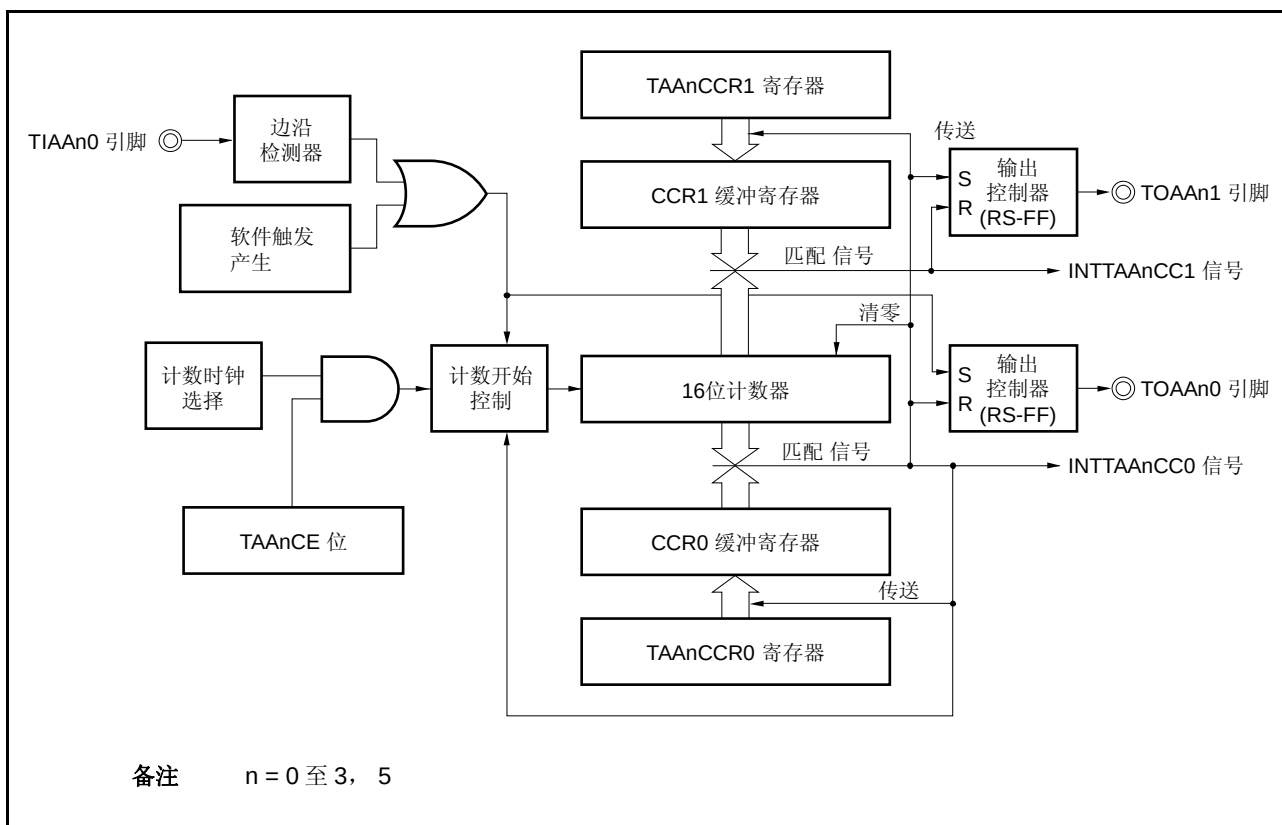
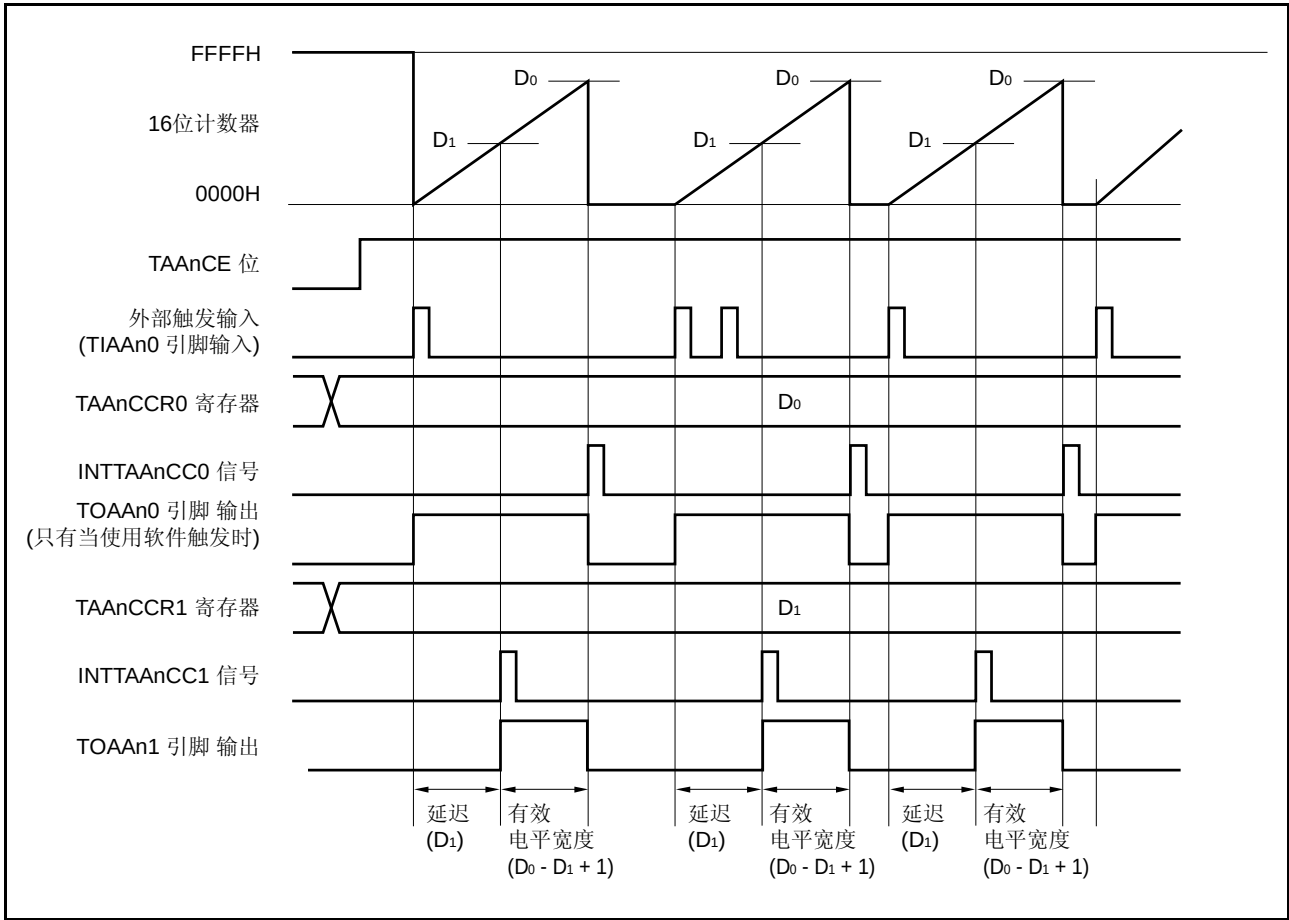


图 7-26. 单次脉冲输出模式的基本时序



当 TAAAnCE 位设置为 1 时，16 位定时器/事件计数器 AA 等待触发。在触发发生时，16 位计数器从 FFFFH 清零为 0000H，开始进行计数，并从 TOAAAn1 引脚输出一个单次脉冲。在输出单次脉冲后，16 位计数器被设置为 FFFFH，计数停止，并重新等待触发。当 TOP01 引脚正在输出单次脉冲期间再次发生触发，则该触发被忽略。

单次脉冲的输出延迟时间和有效电平宽度可计算如下：

输出延迟时间 = (TAAAnCCR1 寄存器的设定值) × 计数时钟周期
有效电平宽度 = (TAAAnCCR0 寄存器的设定值 - TAAAnCCR1 寄存器的设定值 + 1) × 计数时钟周期

当 16 位计数器在其计数值与 CCR0 缓冲寄存器的值匹配之后进行下一个计数时，产生 INTTAAAnCC0 比较匹配中断请求信号。当 16 位计数器的计数值与 CCR1 缓冲寄存器的值匹配时，会生成比较匹配中断请求信号 INTTAAAnCC1。外部触发输入信号的有效边缘或将软件触发 (TAAAnCTL1.TAAAnEST 位) 设置为 1，都可用作触发信号。

备注 n = 0 至 3, 5

图 7-27. 单次脉冲输出模式时的寄存器设置 (1/2)

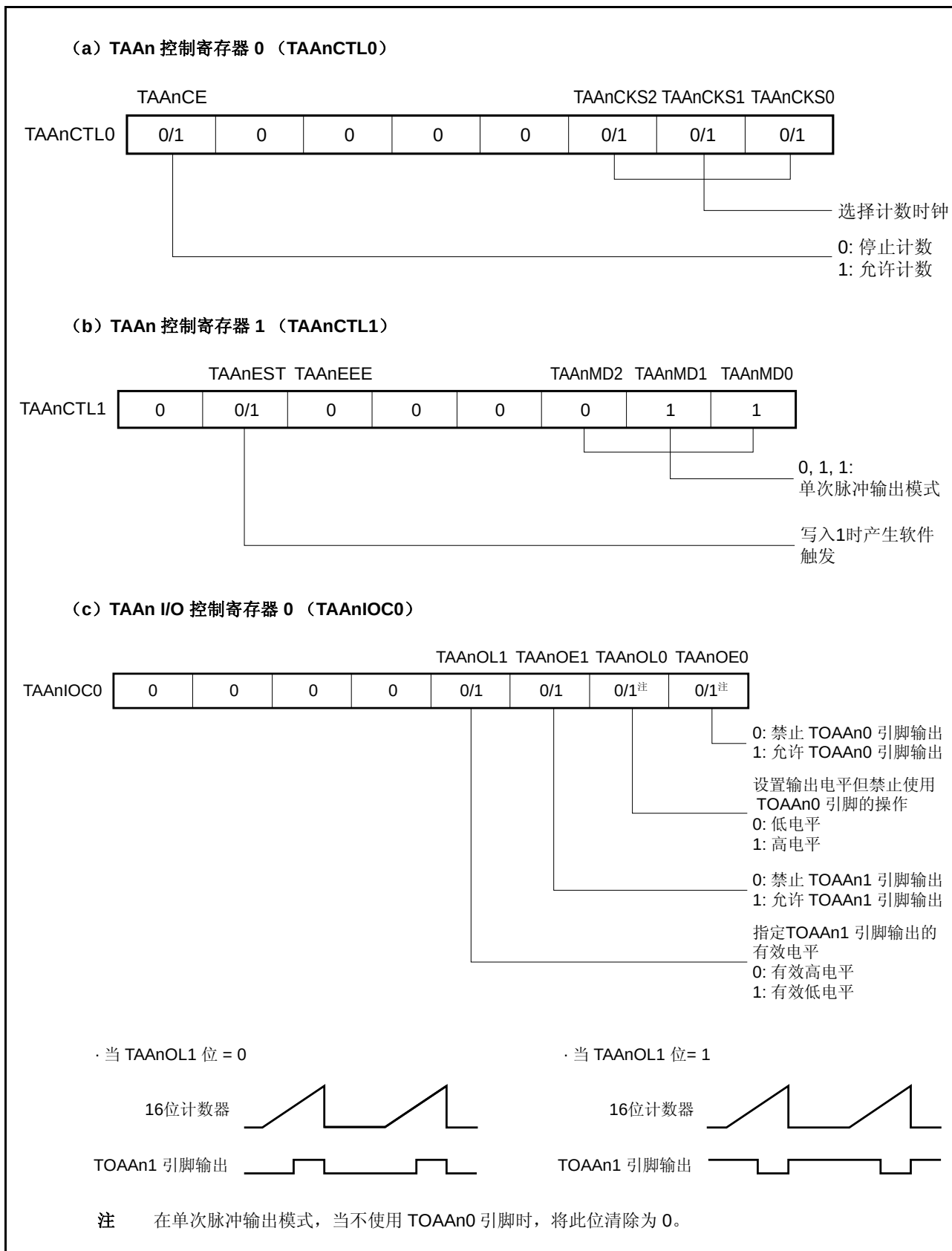


图 7-27. 单次脉冲输出模式时的寄存器设置 (2/2)

(d) TAA_n I/O 控制寄存器 2 (TAA_nIOC2)

(d) TAA_n I/O 控制寄存器 2 (TAA_nIOC2)

	TAAnees1	TAAnees0	TAAets1	TAAets0
TAAioc2	0	0	0	0
	0	0	0/1	0/1

	选择外部触发输入的有效沿

(e) TAA_n 计数器读取缓冲寄存器 (TAA_nCNT)

通过读取 TAAncCNT 寄存器，可以读出 16 位计数器的值。

(f) TAA_n 捕获/比较寄存器 0 和 1 (TAA_nCCR0 和 TAA_nCCR1)

若把 TAAAnCCR0 寄存器的值设置为 D₀, TAAAnCCR1 寄存器的值设置为 D₁, 则单次脉冲的有效电平宽度和出延迟期计算如下。

$$\text{有效电平宽度} = (D_0 - D_1 + 1) \times \text{计数时钟周期}$$

输出延迟时间 = (D₁) × 计数时钟周期

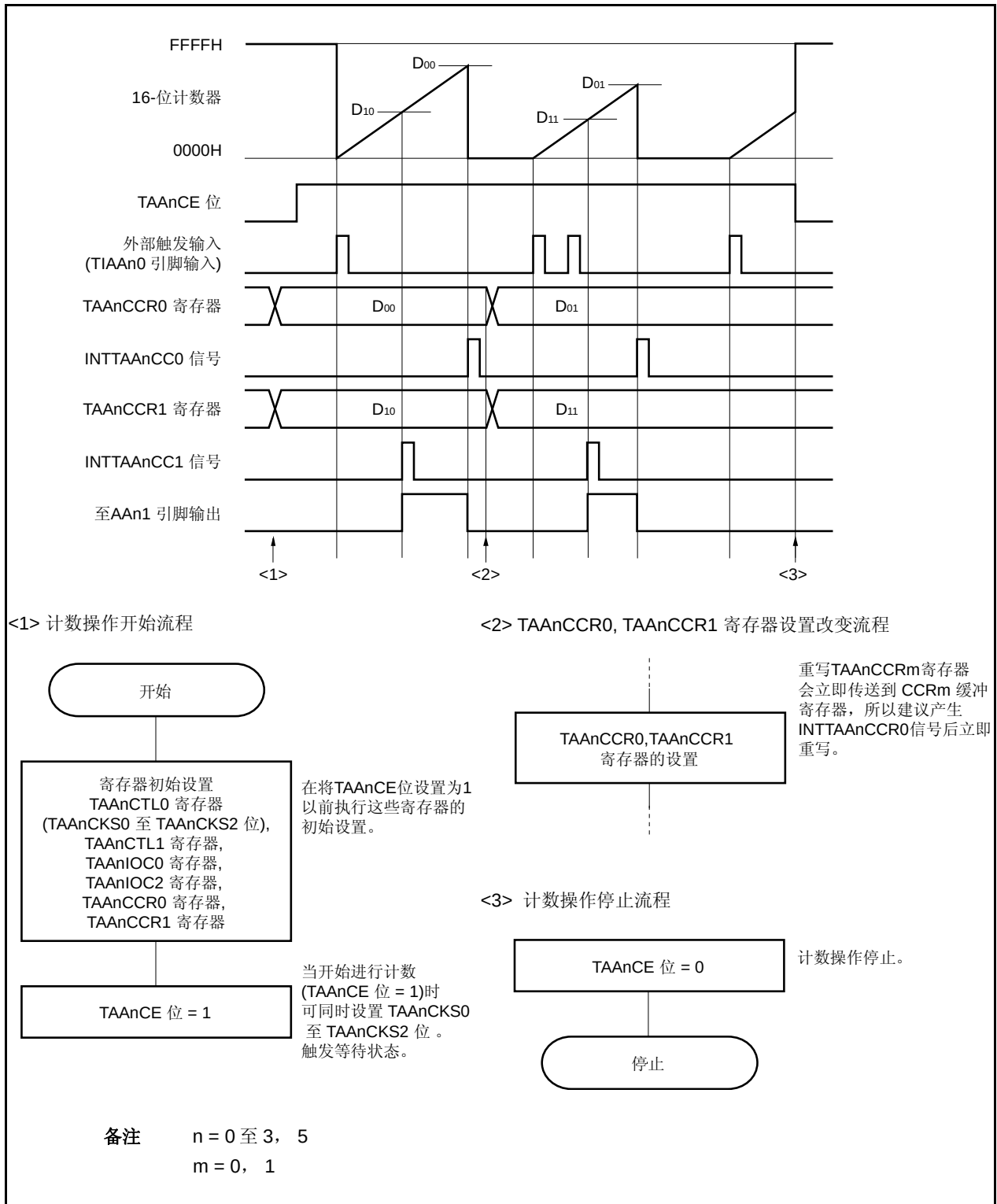
注意事项 如果 TAAAnCCR1 寄存器中设置的值大于 TAAAnCCR0 寄存器中设置的值, 那么, 即使在单次脉冲输出模式下也不会输出单次脉冲。

备注	1. 在单次脉冲输出模式下，不使用 TAA _n 输入/输出控制寄存器 1（TAA _n IOC1）和 TAA _n 选项寄存器 0（TAA _n OPT0）。

2. $n=0$ 至 3, 5

(1) 单次脉冲输出模式下的操作流程

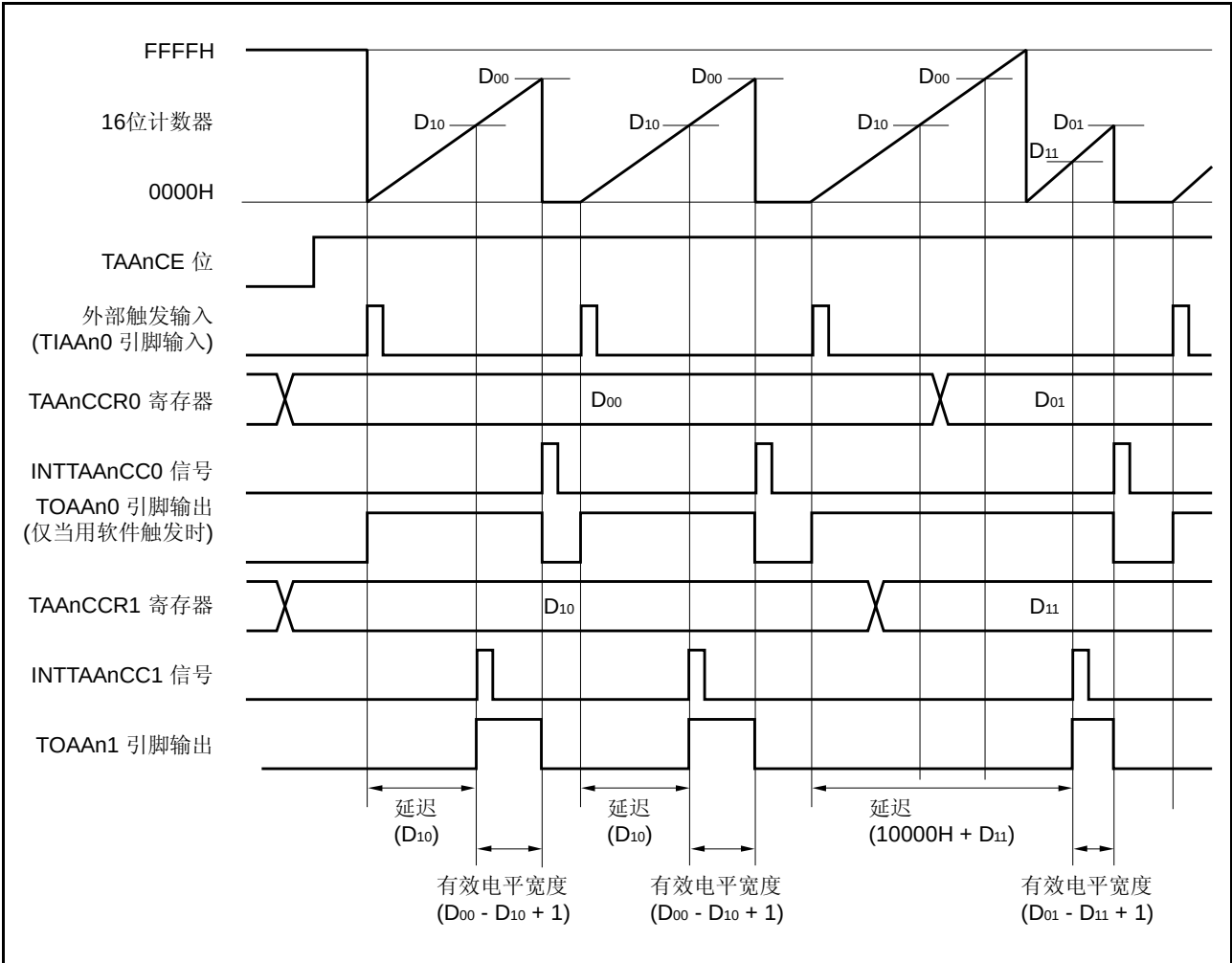
图 7-28. 单次脉冲输出模式下的软件处理流程



(2) 单次脉冲输出模式时的工作时序

(a) 关于重写 TAAAnCCRm 寄存器的注意事项

要将 TAAAnCCRm 寄存器的值改为较小的值，需要停止计数，而后改变设定值。
如果在计数器进行计数期间，对寄存器 TAAAnCCRm 写入一个较小的值，则 16 位计数器可能发生溢出。

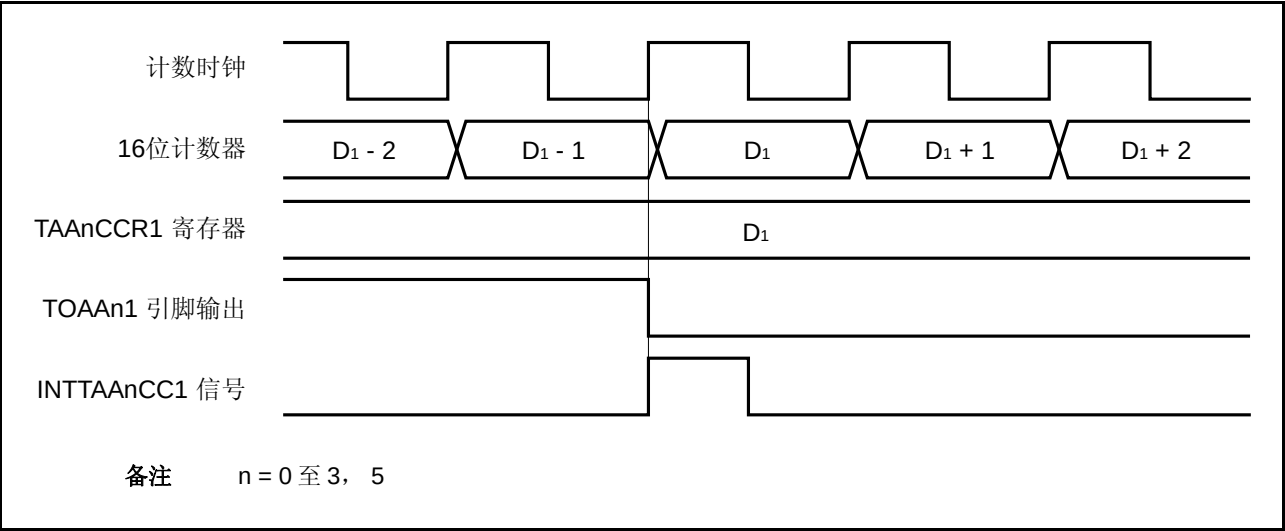


如果 $D_{00} > D_{01}$ 而且 $D_{10} > D_{11}$ ，当 TAAAnCCR0 寄存器从 D_{00} 重写为 D_{01} ，TAAAnCCR1 寄存器从 D_{10} 重写为 D_{11} 时，若 TAAAnCCR1 寄存器在 16 位计数器的计数值大于 D_{11} 且小于 D_{10} 时重写，以及 TAAAnCCR0 寄存器在计数值大于 D_{01} 且小于 D_{00} 时重写，则一旦重写寄存器，就会在各设置值上反映出来并立即与计数值进行比较。计数器向上计数到 FFFFH 后，从 0000H 开始重新向上计数。当计数值与 D_{11} 匹配时，计数器会产生 INTTAAAnCC1 信号并使 TOAAn1 引脚置位有效。当计数值与 D_{01} 匹配时，计数器产生 INTTAAAnCC0 信号，使 TOAAn1 引脚清除置位，并停止计数。
因此，计数器输出脉冲的延迟期或有效期可能会与原先预计的单次脉冲的有所不同。

备注 $n = 0$ 至 3, 5
 $m = 0, 1$

(b) 比较匹配中断请求信号 (INTTAAAnCC1) 的发生时序

单次脉冲输出模式下，INTTAAAnCC1 信号的发生时序和其它模式下的 INTTAAAnCC1 信号的发生时序不同，当 16 位计数器的计数值与 TAAAnCCR1 寄存器的值匹配时，在单次脉冲输出模式下就会产生 INTTAAAnCC1 中断请求信号。



通常情况下，在 16 位计数器的当前计数值与 TAAAnCCR1 寄存器的值匹配之后，INTTAAAnCC1 信号与下一次计数同步发生。

但是，在单次脉冲输出模式下，INTTP0CC1 信号会提早一个计数时钟周期产生。这是由于为了匹配 TOAAAn1 引脚输出信号的变化，时序发生了变化。

7.5.5 PWM（脉宽调制信号）输出模式（TAA_nMD2 至TAA_nMD0 位= 100）

在脉宽调制信号输出模式下，当 TAA_nCTL0.TAA_nCE 位被设定为 1 时，脉宽调制信号波形从 TOAAn 引脚输出。而且，由 TOAAn0 引脚输出的脉冲，其半周期等于脉宽调制信号的周期。

图 7-29. 脉宽调制信号输出模式的配置

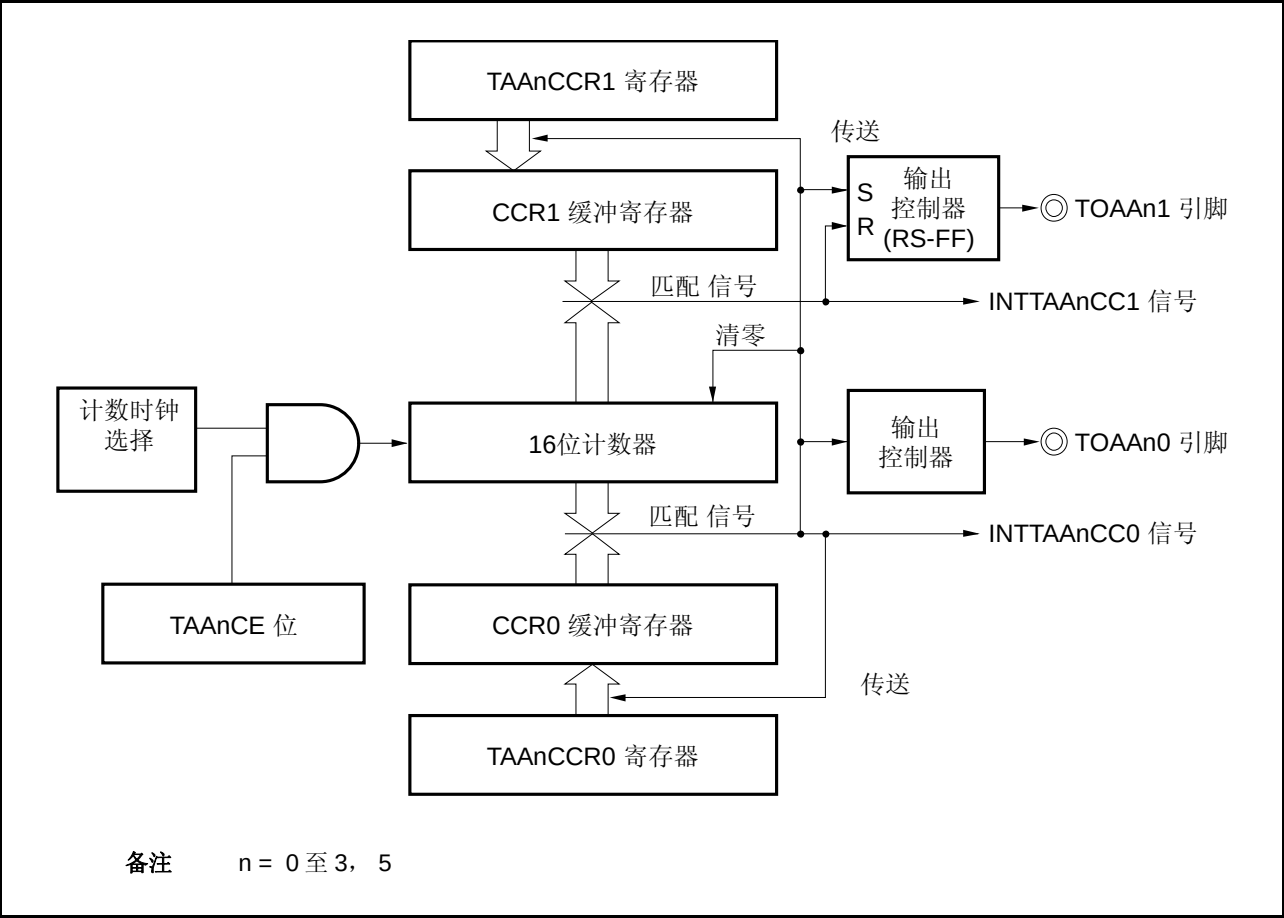
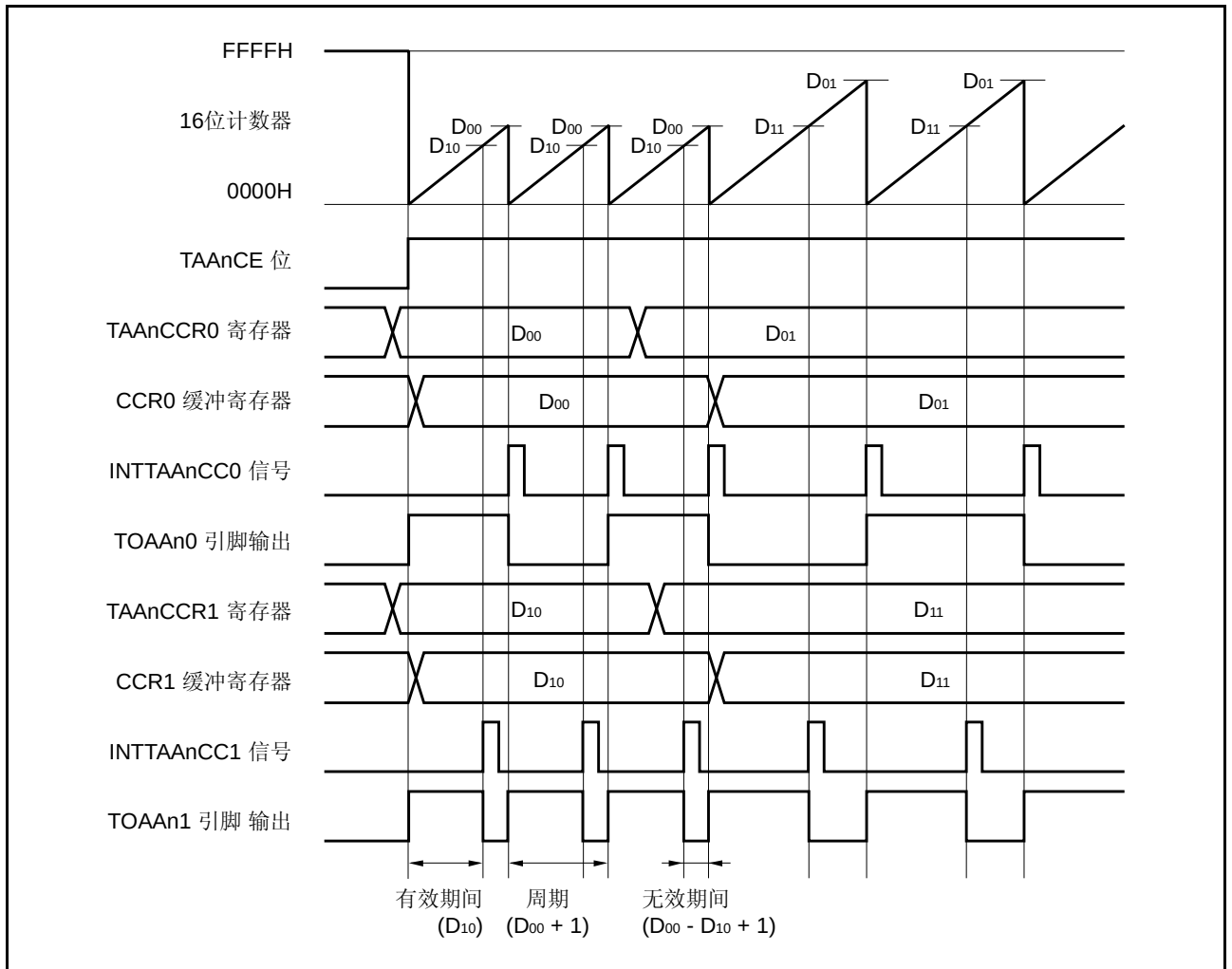


图 7-30. 脉宽调制信号输出模式的基本时序



当 TAAAnCE 位设置为 1 时，16 位计数器从 FFFFH 清零为 0000H，开始进行计数，同时从 TOAAAn1 引脚输出脉冲宽度调制信号波形。

脉冲宽度调制信号波形的有效电平宽度、周期、占空比计算如下。

有效电平宽度 = (TAAAnCCR1 寄存器的设定值) × 计数时钟周期

周期 = (TAAAnCCR0 寄存器的设定值 + 1) × 计数时钟周期

占空比 = (TAAAnCCR1 寄存器的设定值) / (TAAAnCCR0 寄存器的设定值 + 1)

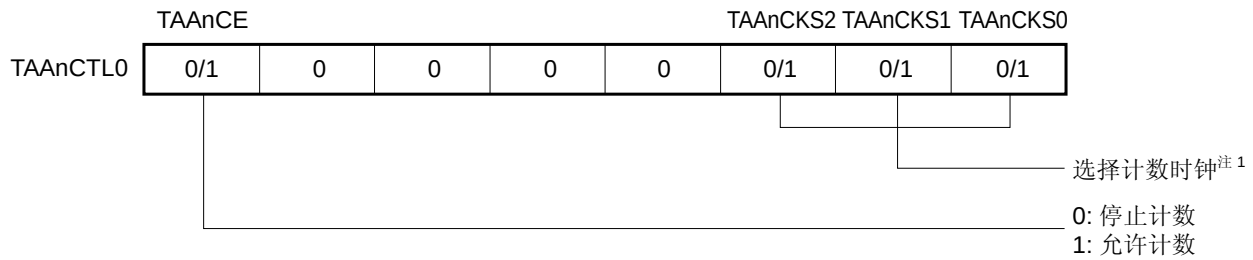
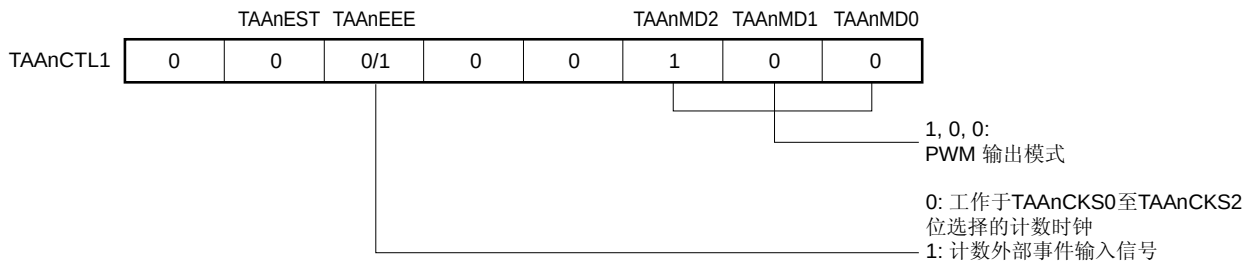
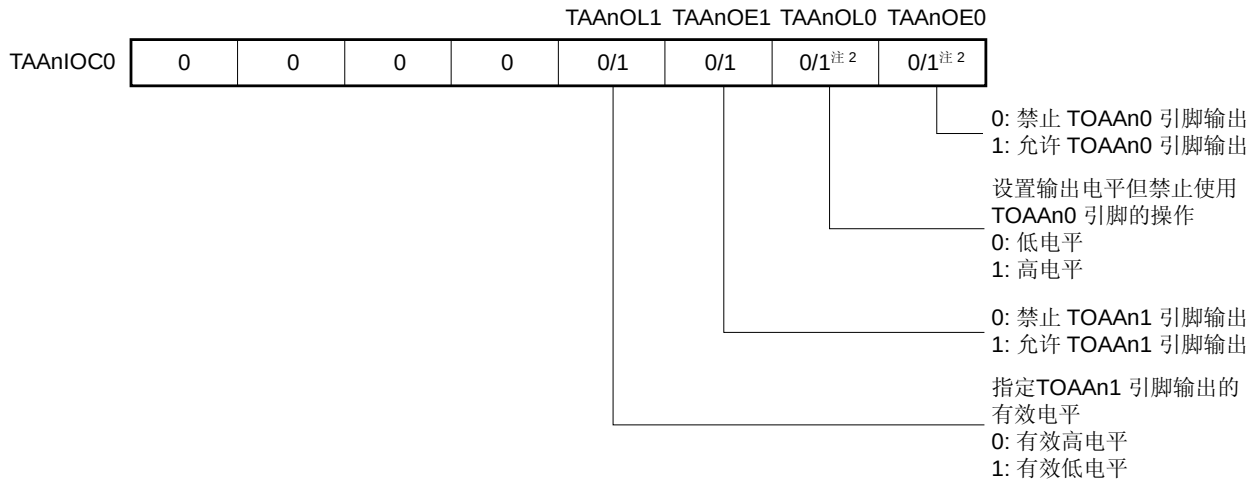
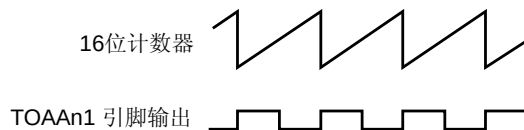
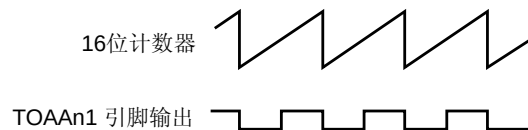
在进行计数时，可通过重写 TAAAnCCRm 寄存器来改变脉冲宽度调制信号波形。当 16 位计数器的计数值与 CCR0 缓冲寄存器的值匹配时，16 位计数器清零为 0000H 时，新写入的值会得到反映。

当 16 位计数器在其计数值与 CCR0 缓冲寄存器的值匹配之后进行下一次向上计数时，会同步产生比较匹配中断请求信号 INTTAAAnCC0，且 16 位计数器清零为 0000H。当 16 位计数器的计数值与 CCR1 缓冲寄存器的值匹配时，会生成比较匹配中断请求信号 INTTAAAnCC1。

当 16 位计数器的计数值与 CCRm 缓冲寄存器的值匹配，而且 16 位计数器清零为 0000H 时，TAAAnCCRm 寄存器设置的值就传送到 CCRm 缓冲寄存器。

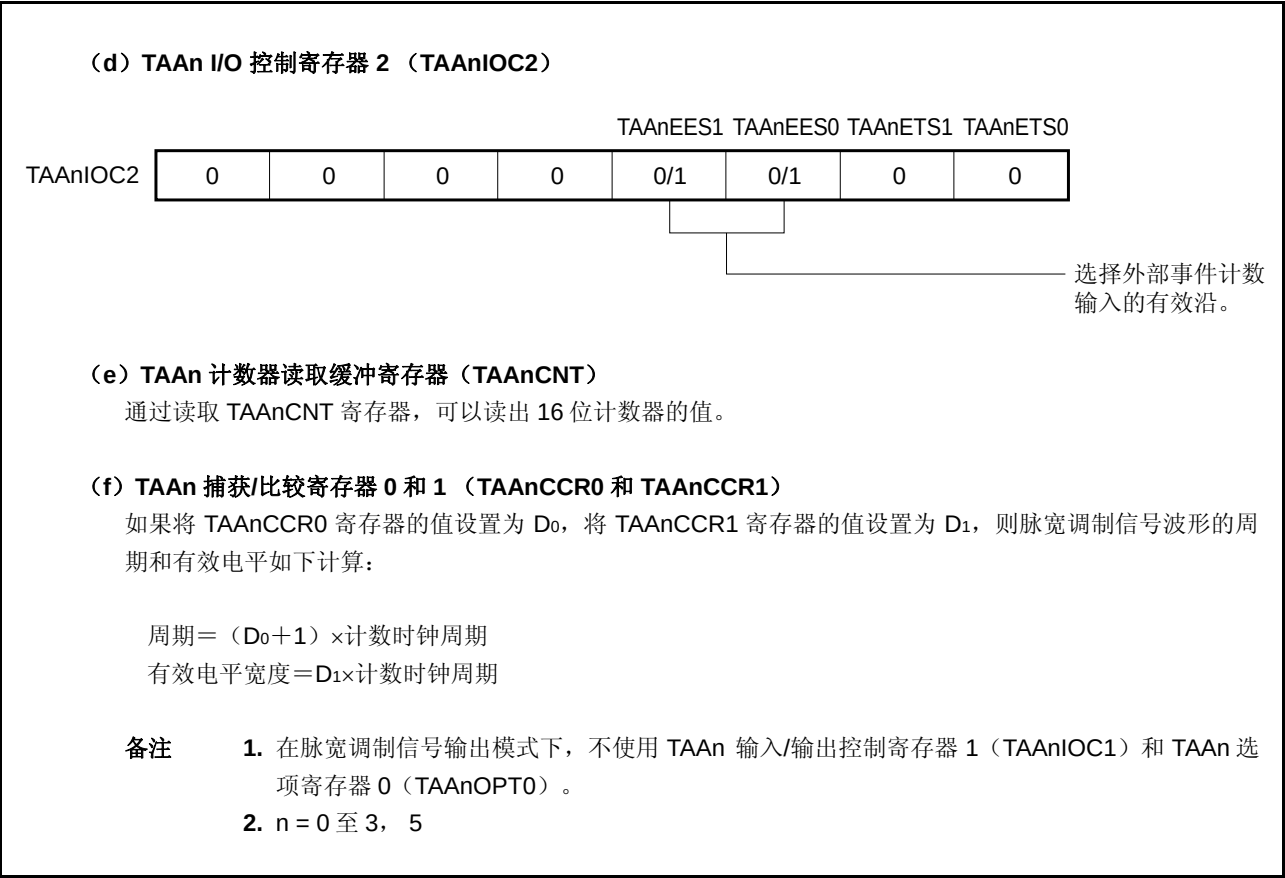
备注 n = 0 至 3, 5
 m = 0, 1

图 7-31. 脉宽调制信号输出模式时的寄存器设置 (1/2)

 (a) TAA_n 控制寄存器 0 (TAA_nCTL0)

 (b) TAA_n 控制寄存器 1 (TAA_nCTL1)

 (c) TAA_n I/O 控制寄存器 0 (TAA_nIOC0)

 · 当 TAA_nOL1 位 = 0

 · 当 TAA_nOL1 位 = 1


- 注
1. 当 TAA_nCTL1.TAA_nEEE 位 = 1 时, 设置无效。
 2. 在脉宽调制信号输出模式, 当不使用 TOAAn0 引脚时, 将此位清除为 0。

图 7-31. 脉宽调制信号输出模式时的寄存器设置 (2/2)



(1) 脉宽调制信号输出模式下的工作流程

图 7-32. 脉宽调制信号输出模式时的软件处理流程 (1/2)

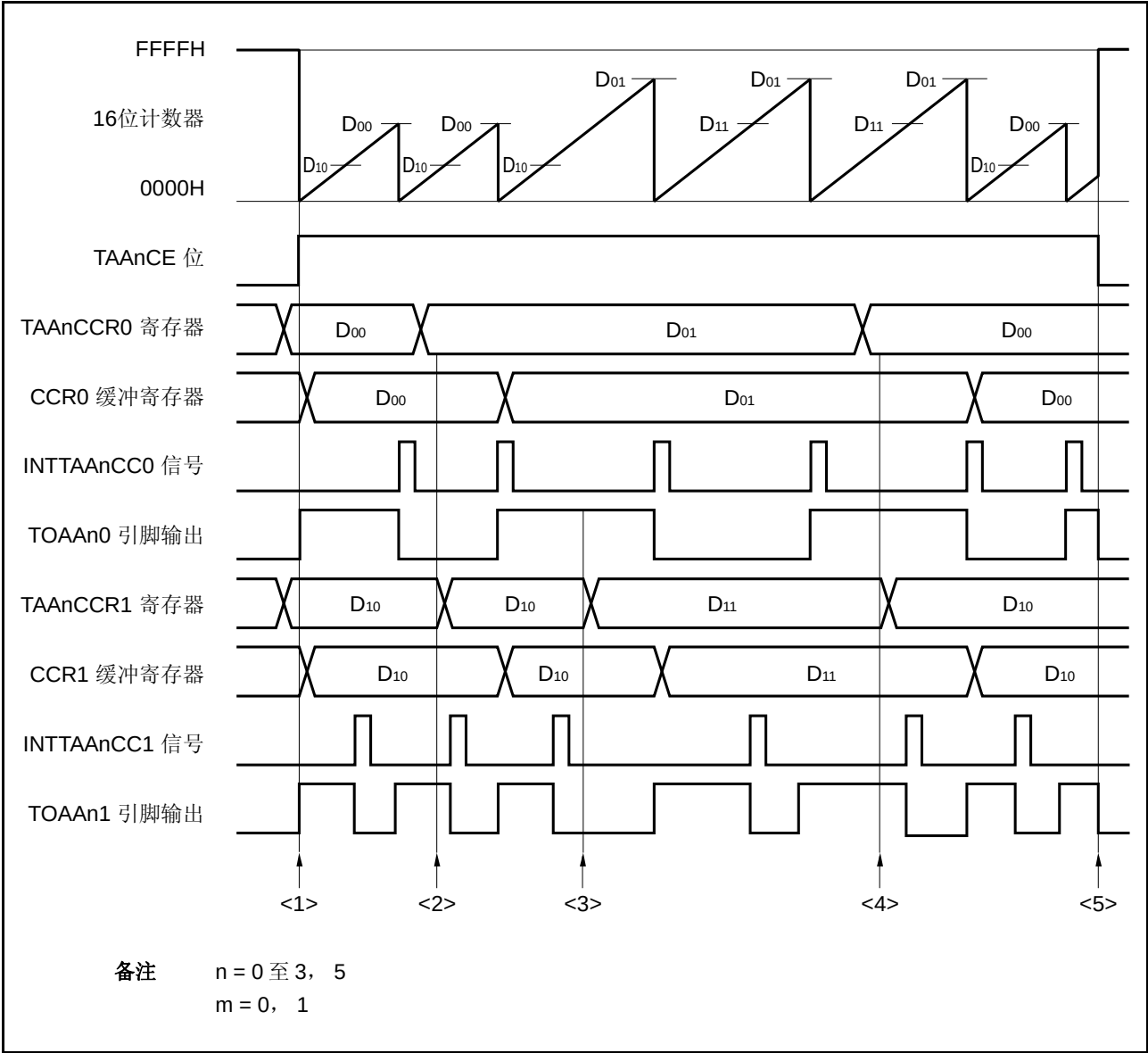
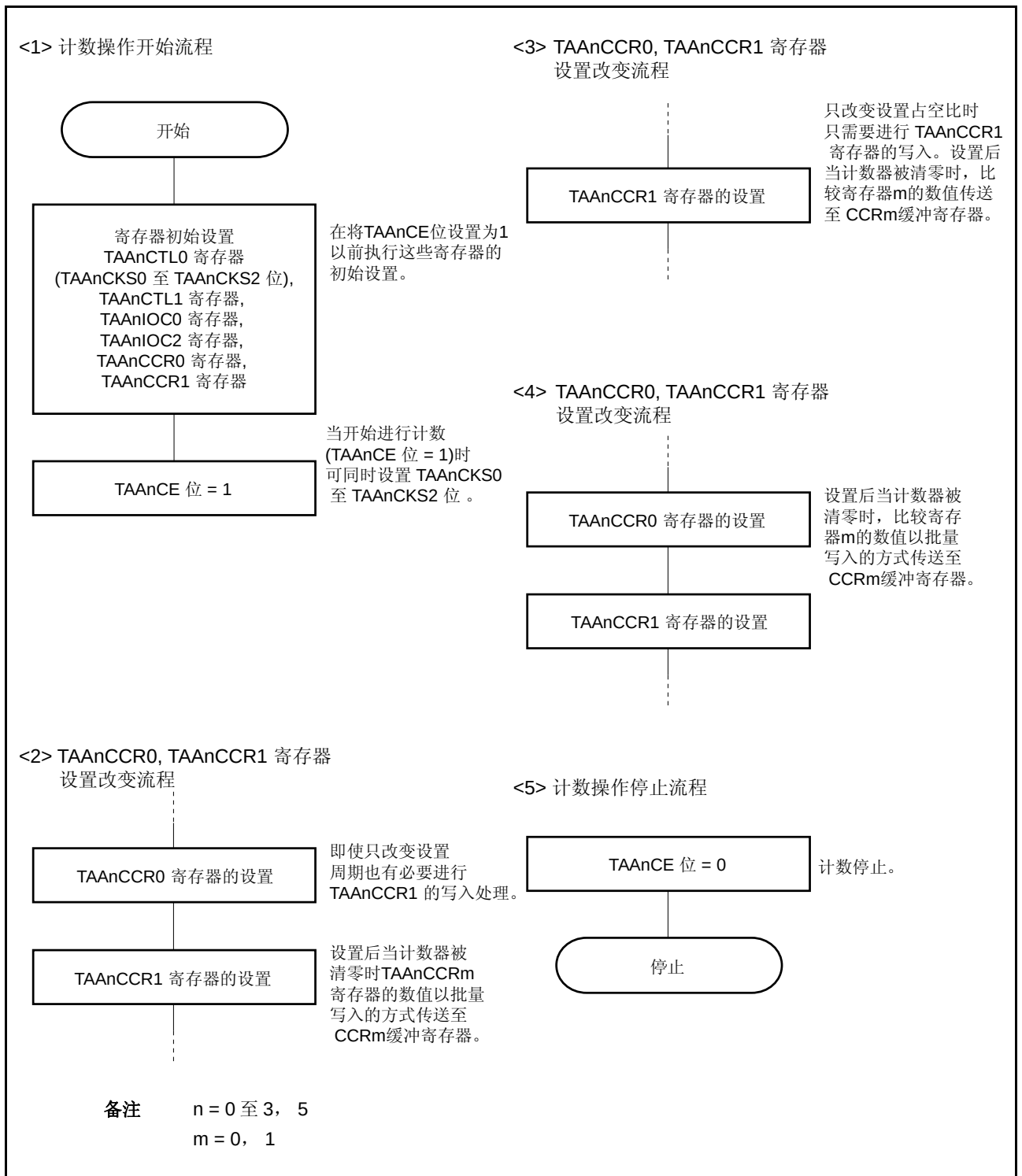


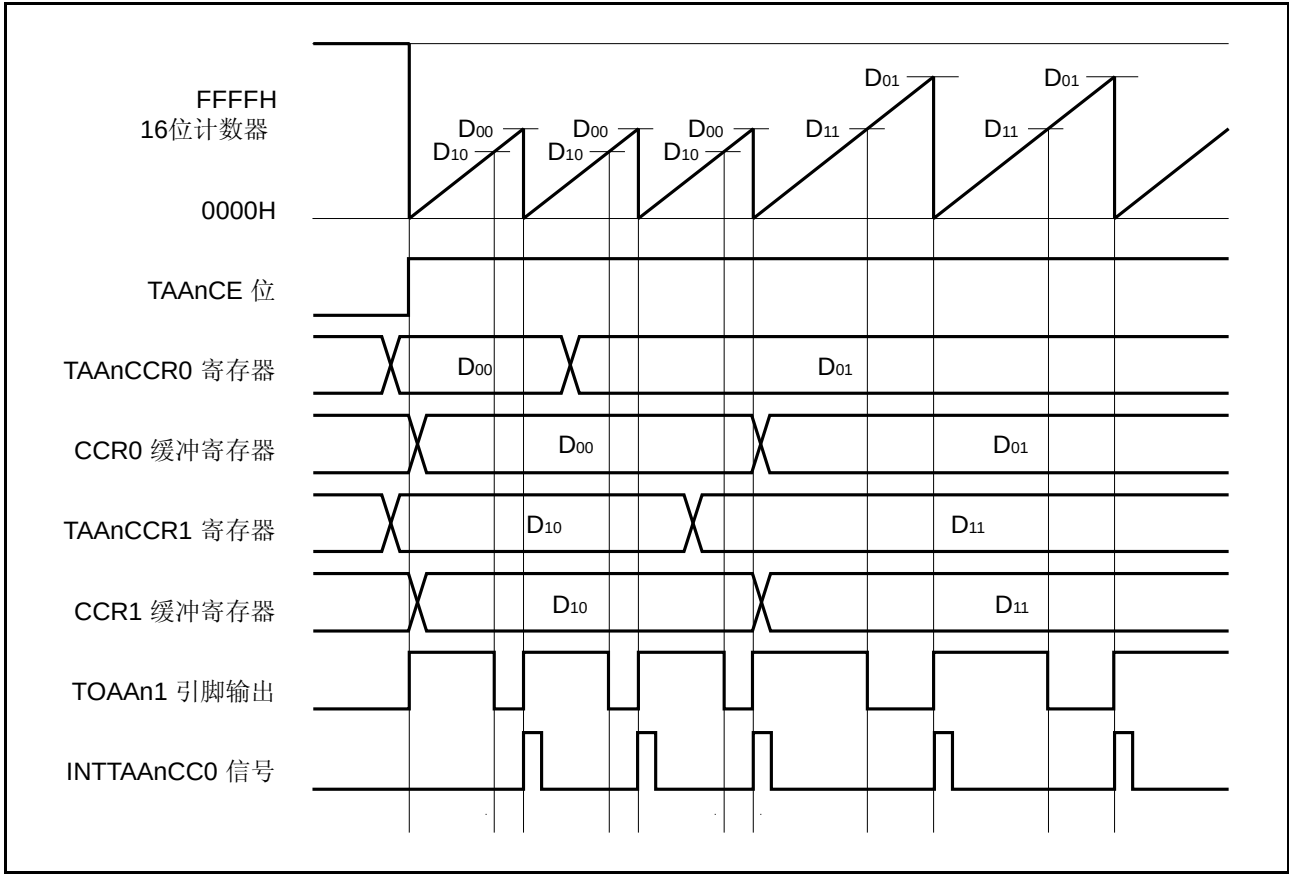
图 7-32. 脉宽调制信号输出模式时的软件处理流程 (2/2)



(2) 脉宽调制信号输出模式的操作时序

(a) 操作过程中改变脉冲宽度

计数器计数过程中，若要改变脉宽调制信号波形，应最后写入 TAAAnCCR1 寄存器。
检测到 INTTAAAnCC1 信号以后，在写入 TAAAnCCR1 寄存器之后重写 TAAAnCCRm 寄存器。

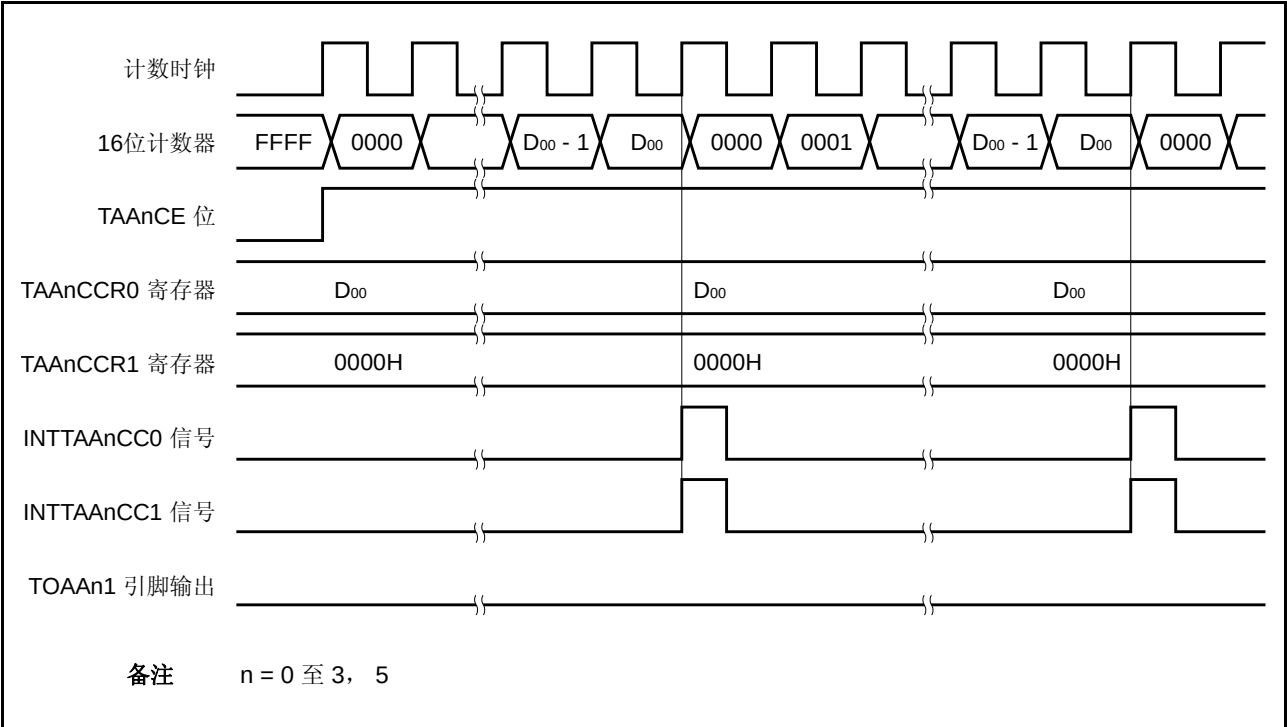


若要将数据从 TAAAnCCRm 寄存器传送到 CCRm 缓冲寄存器，则必须写入 TAAAnCCR1 寄存器。
此时要同时改变脉宽调制信号波形的周期和有效电平，应首先把周期设置给 TAAAnCCR0 寄存器，然后把有效电平设置给 TAAAnCCR1 寄存器。
如果只改变脉宽调制信号波形的周期，应首先把周期设置给 TAAAnCCR0 寄存器，然后将相同值写入 TAAAnCCR1 寄存器。
只改变脉宽调制信号波形的有效电平宽度（占空比）时，那就只需要设置 TAAAnCCR1 寄存器。
在数据写入 TAAAnCCR1 寄存器后，写入 TAAAnCCRm 寄存器的值会传送到 CCRm 缓冲寄存器，与 16 位计数器清零同步，并作为与 16 位计数器进行比较的值。
要在写入 TAAAnCCR1 寄存器一次之后，再次写 TAAAnCCR0 或 TAAAnCCR1 寄存器，应在 INTTAAAnCC0 信号产生之后进行。否则，CCRm 缓冲寄存器的值可能不确定，因为把数据从 TAAAnCCRm 寄存器传送到 CCRm 缓冲寄存器的时序与写 TAAAnCCRm 寄存器时序发生冲突。

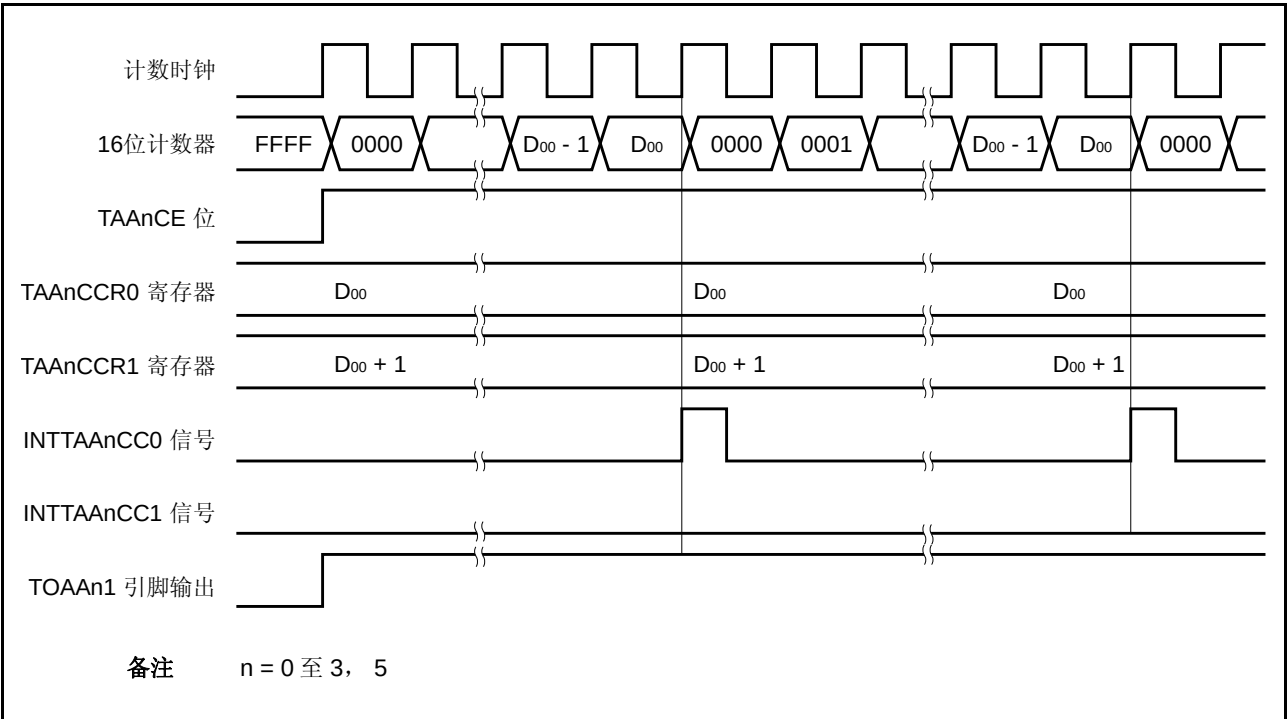
备注 n = 0 至 3, 5
 m = 0, 1

(b) 脉宽调制信号波形的 0%和 100% 输出

为了输出 0% 的波形，需要将 TAAAnCCR1 寄存器设置为 0000H。如果 TAAAnCCR0 寄存器的设定值为 FFFFH，则会周期性产生 INTTAAAnCC1 中断请求信号。

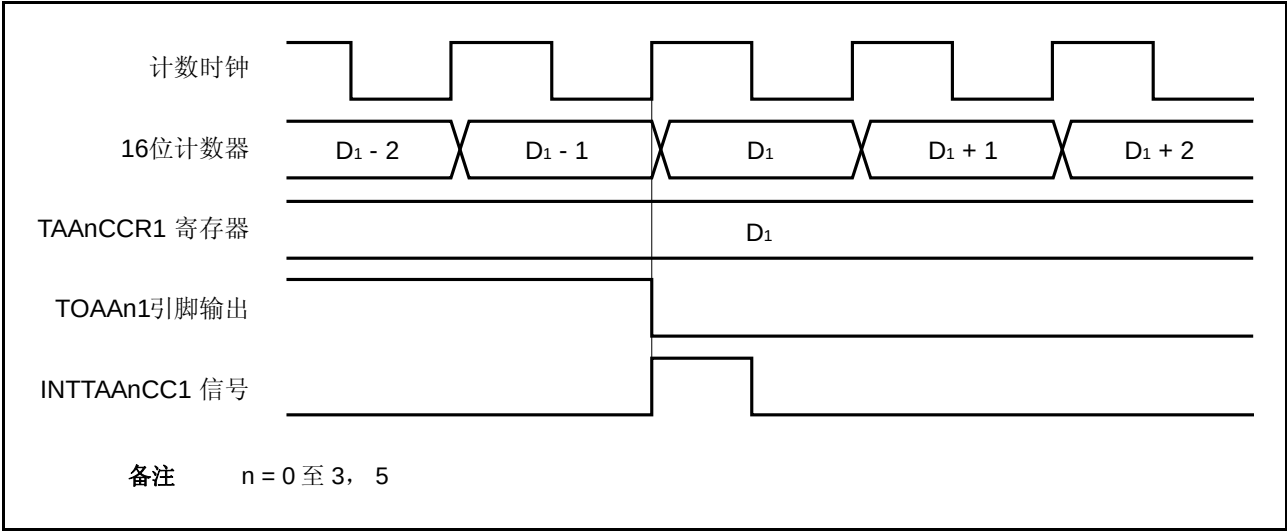


要输出 100%的波形，需给 TAAAnCCR1 寄存器设置一个（TAAAnCCR0 寄存器设定值 + 1）的数值。如果 TAAAnCCR0 寄存器的设定值为 FFFFH，则不会产生 100%波形输出。



(c) 比较匹配中断请求信号 (INTTAAAnCC1) 的发生时序

INTTAAAnCC1 信号在脉宽调制信号输出模式下的发生时序与其它 INTTAAAnCC1 信号的时序不同；当 16 位计数器的计数值与 TAAAnCCR1 寄存器的值匹配时，在脉宽调制信号输出模式下产生 INTTAAAnCC1 中断请求信号。



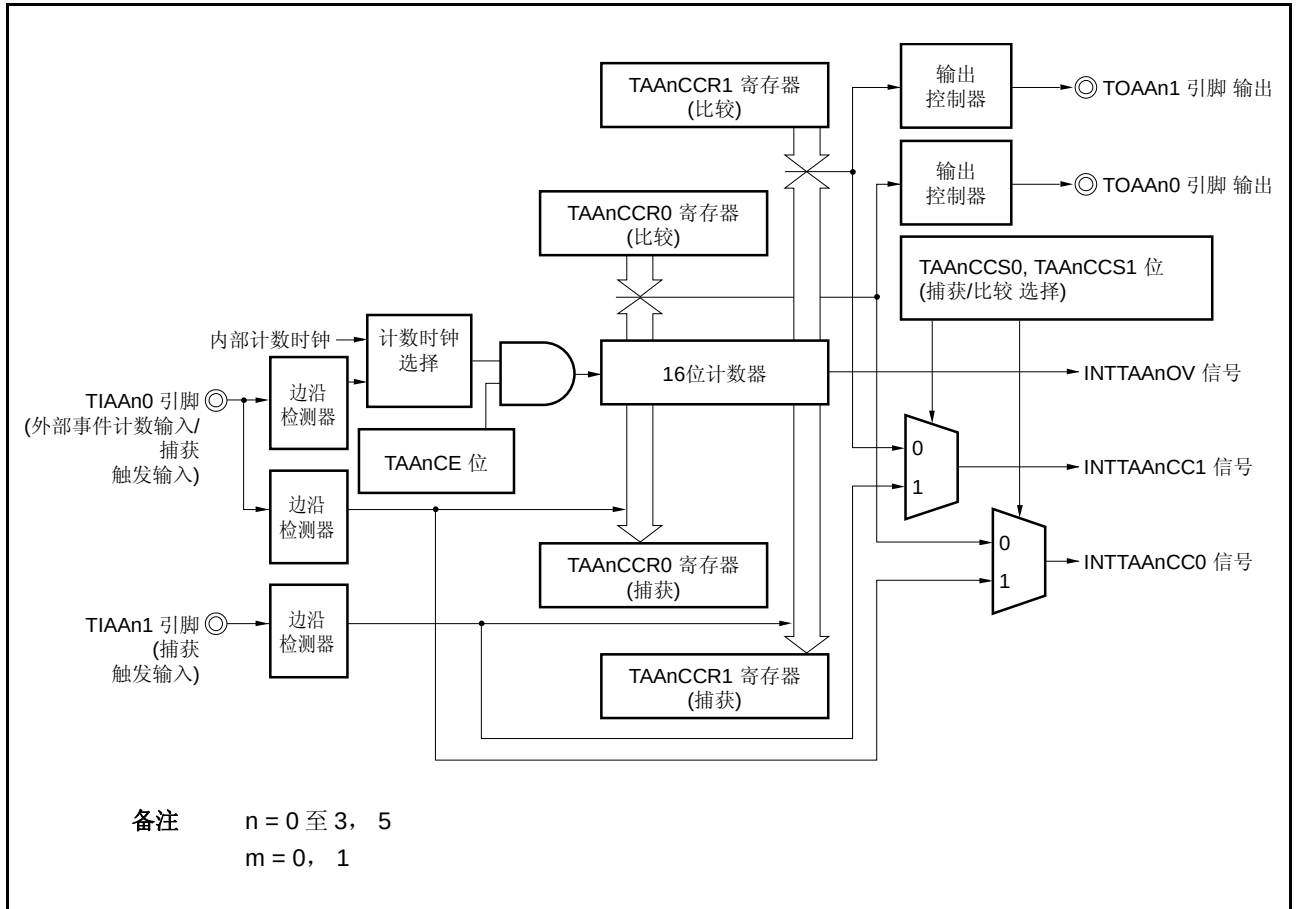
通常情况下，在 16 位计数器的当前计数值与 TAAAnCCR1 寄存器的值匹配之后，INTTAAAnCC1 信号与下一次计数同步发生。

但是，在脉宽调制信号输出模式下，INTTP0CC1 信号会提早一个计数时钟周期产生。这是由于为了匹配 TOAAAn1 引脚输出信号的变化，时序发生了变化。

7.5.6 自由运行定时器模式 (TAA_nMD2 至 TAA_nMD0 位= 101)

在自由运行定时器模式下，当 TAA_nCTL0.TAA_nCE 位被置为 1 时，16 位定时器/事件计数器 AA 开始计数。此时，TAA_nCCR_m 寄存器根据 TAA_nOPT0.TAA_nCCS0 和 TAA_nOPT0.TAA_nCCS1 位的设置，可以用作比较寄存器或捕获寄存器。

图 7-33. 自由运行定时器模式的配置

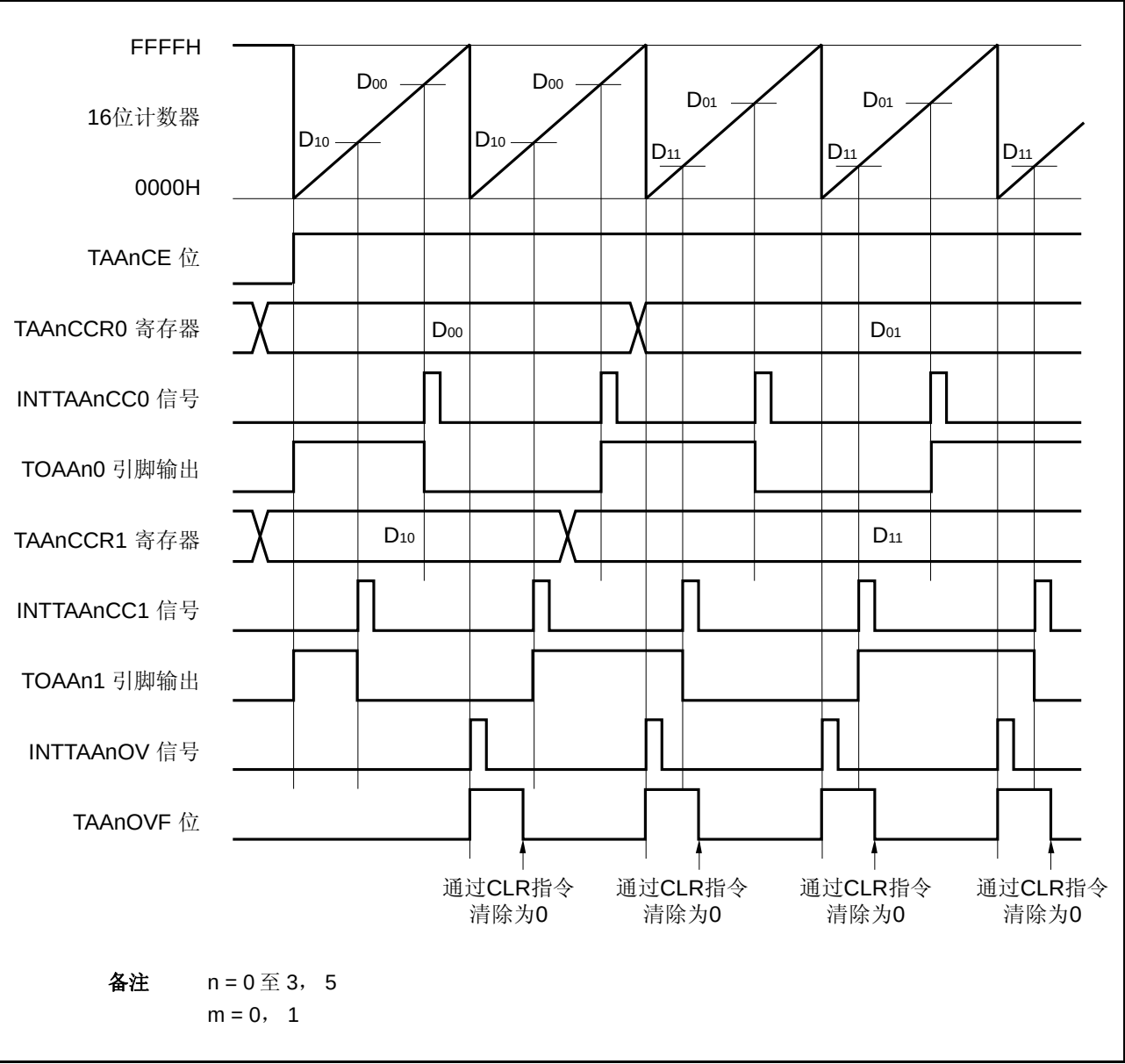


当 TAA_nCE 位被置为 1 时，16 位定时器/事件计数器 AA 开始计数，TOAA_n0 和 TOAA_n1 引脚的输出信号反相。此后当 16 位计数器的计数值与 TAA_nCCR_m 寄存器的设置值匹配时，产生一个比较匹配中断请求信号（INTTAA_nCC_m），而且 TOAA_nm 引脚的输出信号反相。

16 位计数器继续与计数时钟同步进行计数。当它计数到 FFFFH 时，会在下一个计数时钟时产生一个溢出中断请求信号（INTTAA_nOV），同时计数器清零为 0000H，继续进行计数。此时，溢出标志（TAA_nOPT0.TAA_nOVF 位）被置为 1，通过软件运行 CLR 指令可把溢出标志位清 0。

当计数器工作时，TAA_nCCR_m 寄存器可以重写。如果进行重写，则新写入的值会在重写时立刻反映出来，并与计数值进行比较。

图 7-34. 自由运行定时器模式下的基本时序（比较功能）



当 TAA_nCE 位被置为 1 时，16 位计数器开始计数。当检测到 TIAAn_m 引脚输入信号的有效沿时，16 位计数器的计数值存入 TAA_nCCR_m 寄存器中，并产生一个捕获中断请求信号（INTTAA_nCC_m）。

16 位计数器继续与计数时钟同步进行计数。当它计数到 FFFFH 时，会在下一个计数时钟时产生一个溢出中断请求信号（INTTAA_nOV），同时计数器清零为 0000H，继续进行计数。此时，溢出标志（TAA_nOPT0.TAA_nOVF 位）被置为 1，通过软件运行 CLR 指令可把溢出标志位清 0。

图 7-35. 自由运行定时器模式下的基本时序（捕获功能）

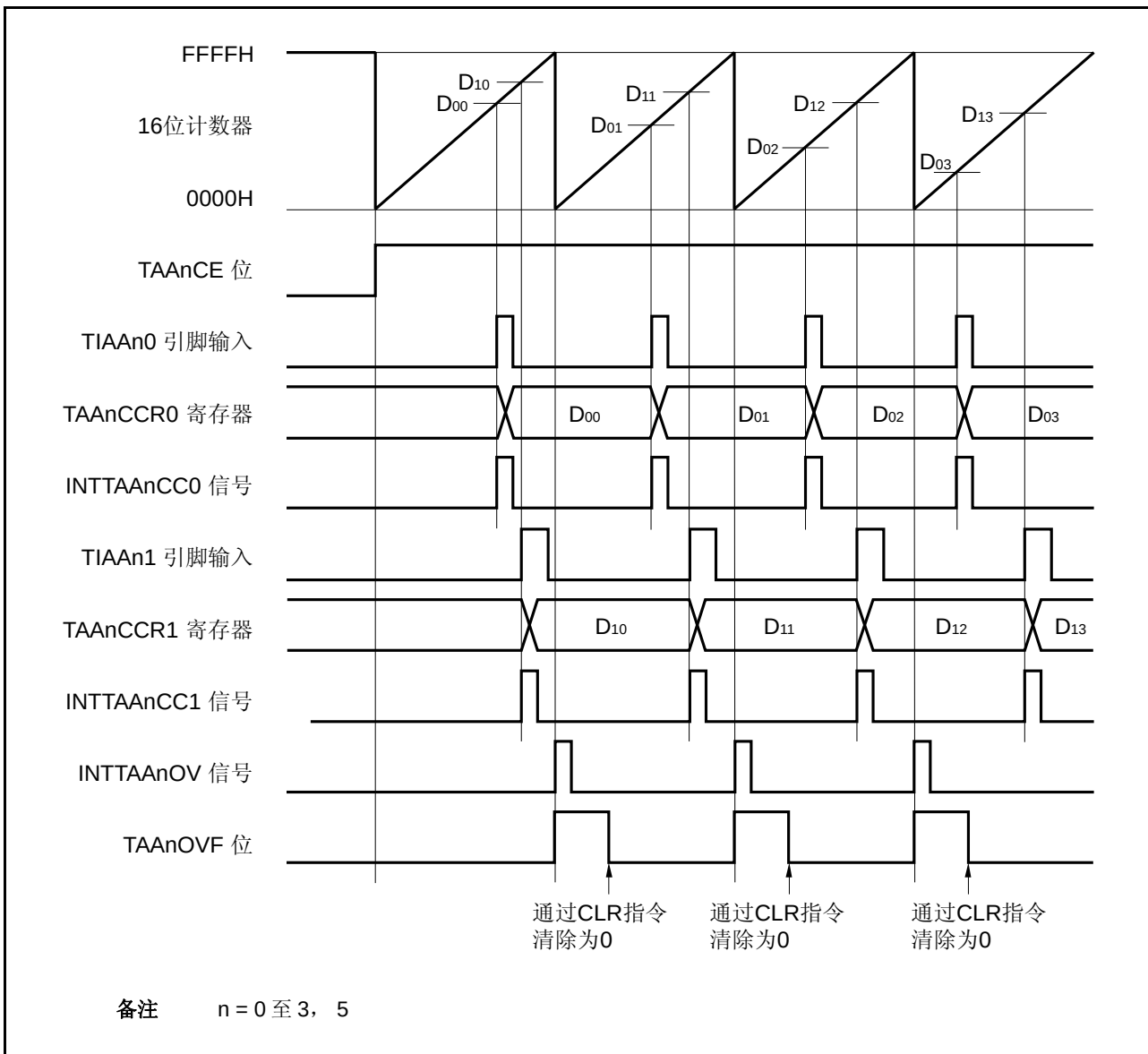


图 7-36. 自由运行定时器模式下的寄存器设置 (1/2)

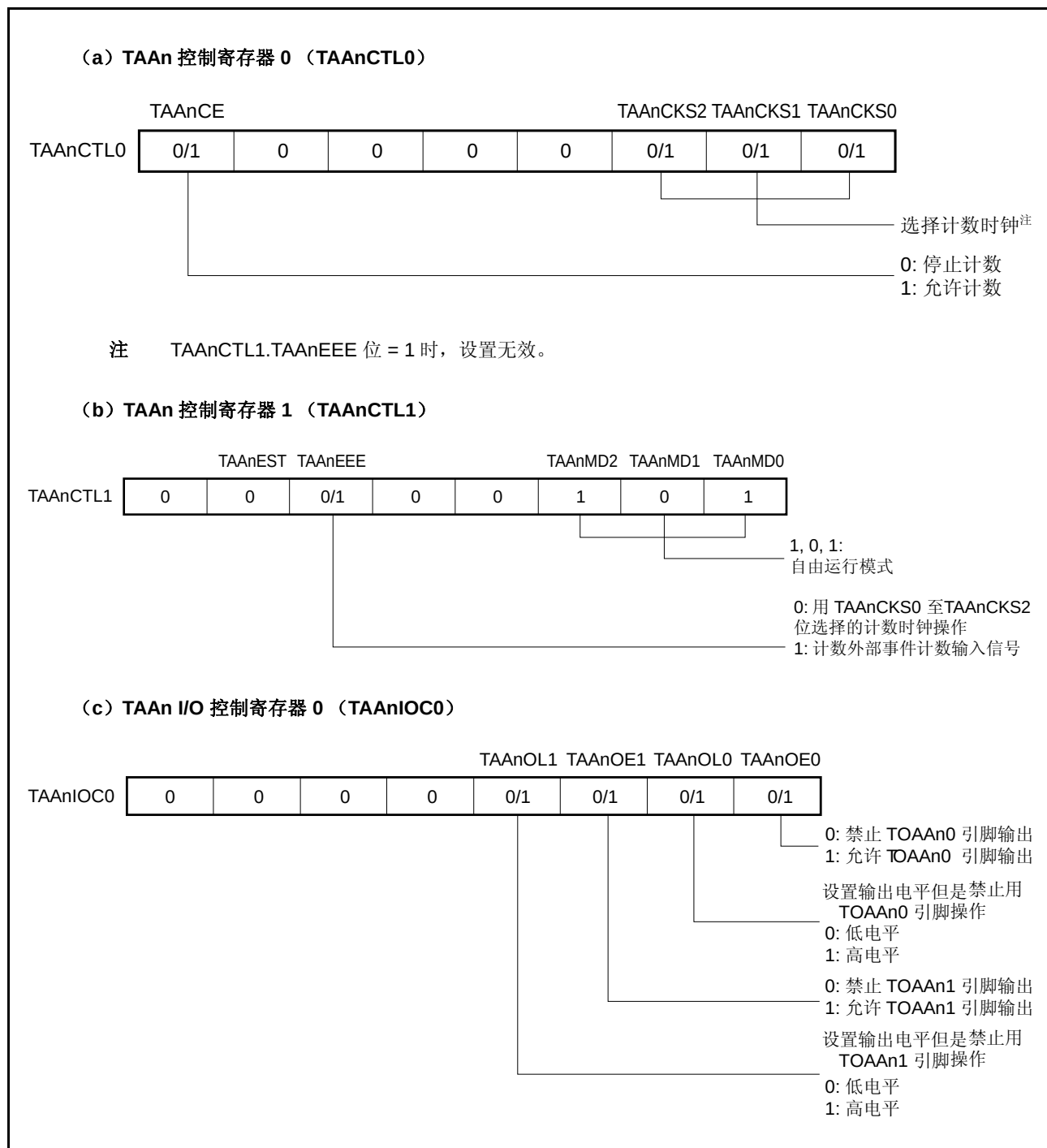
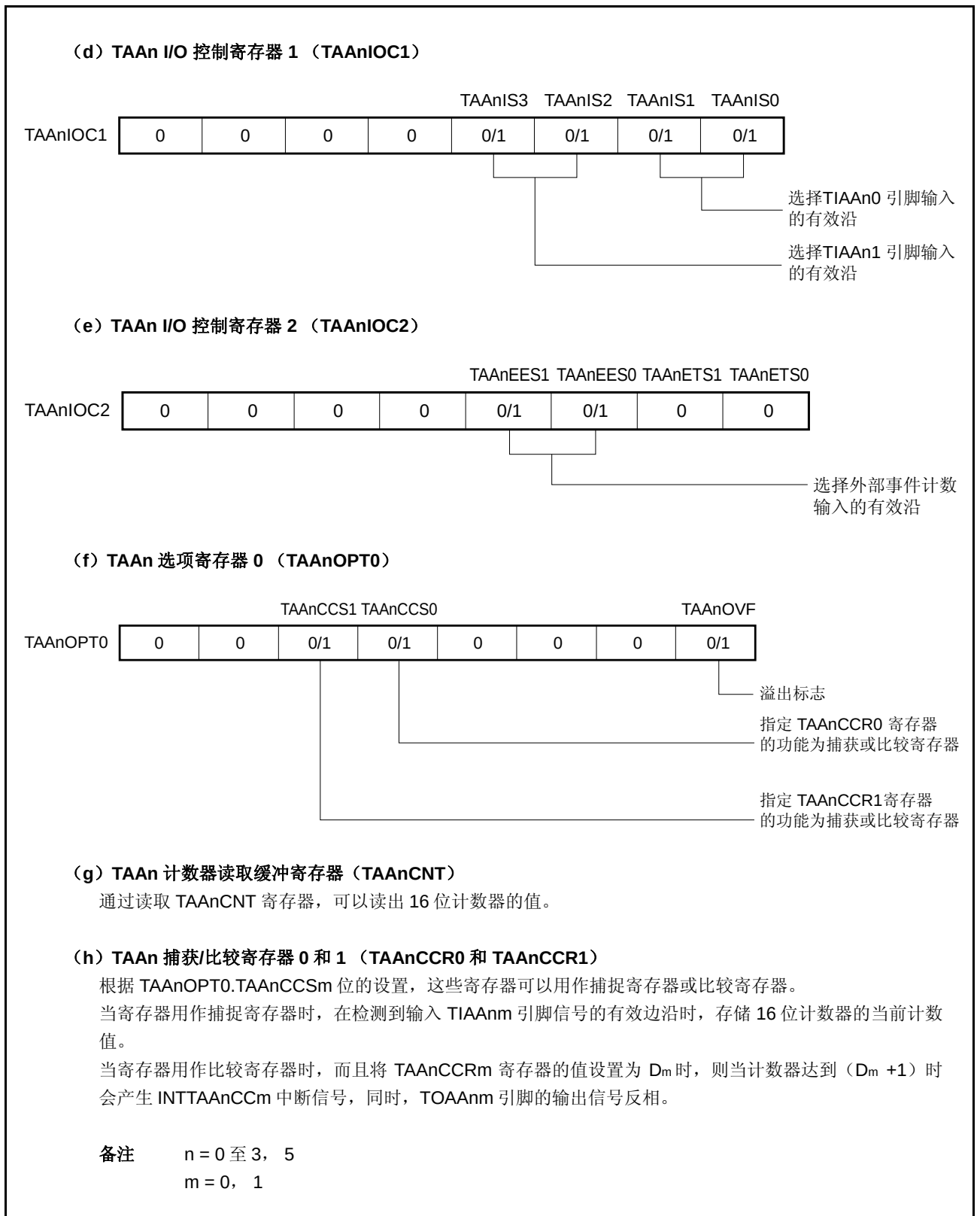


图 7-36. 自由运行定时器模式下的寄存器设置 (2/2)



(1) 自由运行定时器模式时的操作流程

(a) 当捕获/比较寄存器用作比较寄存器时

图 7-37. 自由运行定时器模式时的软件处理流程（比较功能）（1/2）

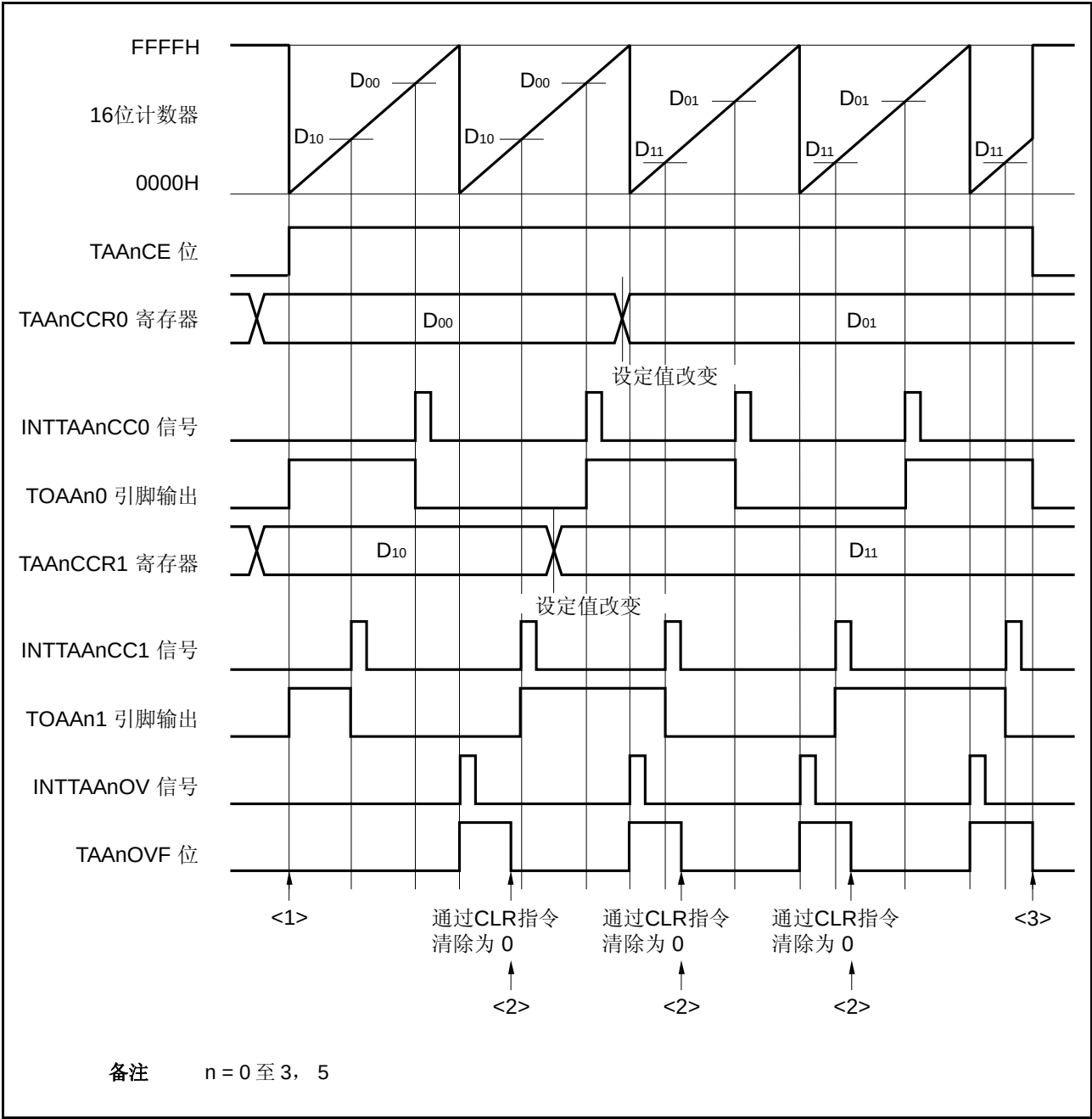
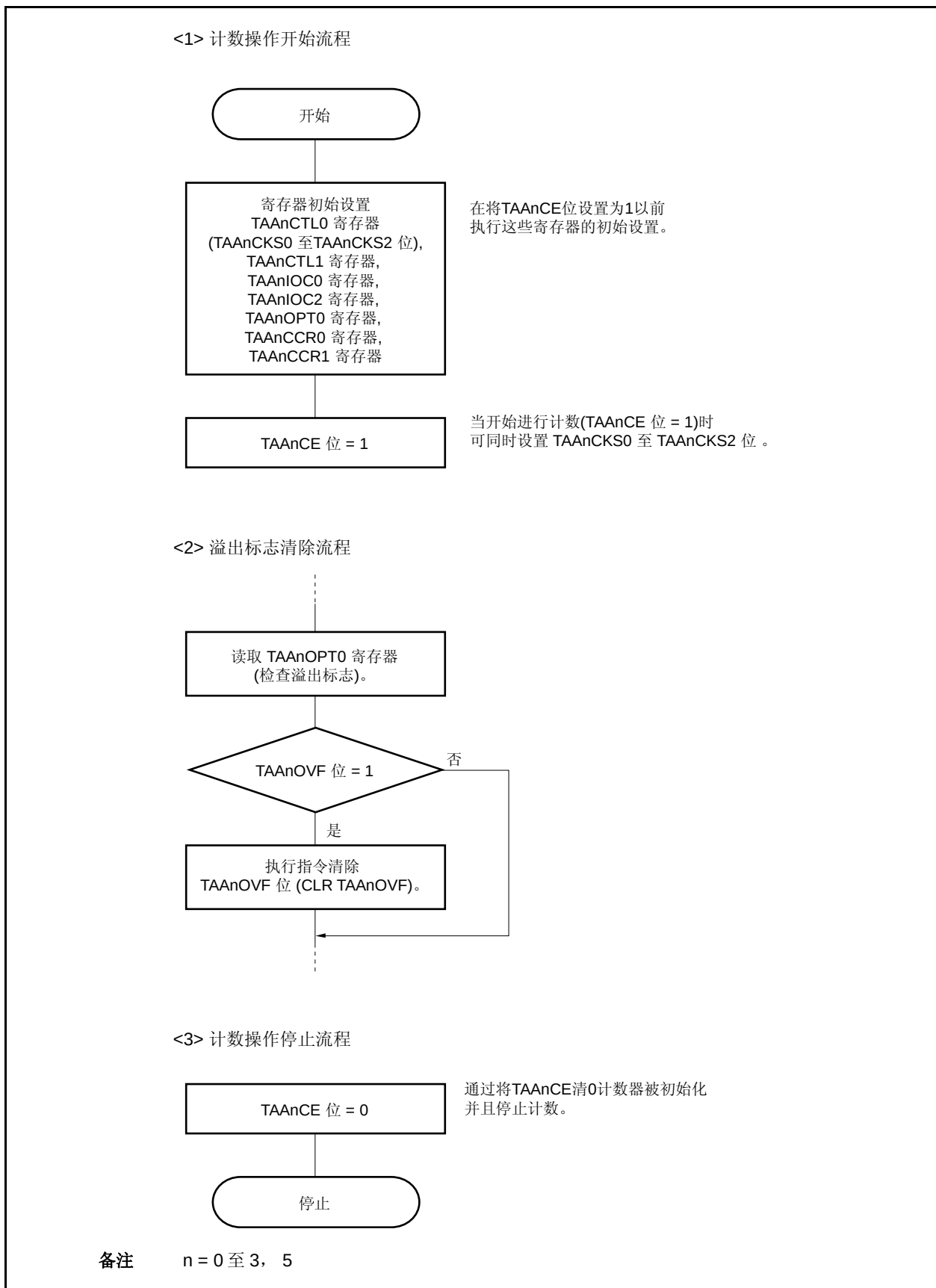


图 7-37. 自由运行定时器模式时的软件处理流程（比较功能）（2/2）



(b) 当捕获/比较寄存器用作捕获寄存器时

图 7-38. 自由运行定时器模式时的软件处理流程（捕获功能）（1/2）

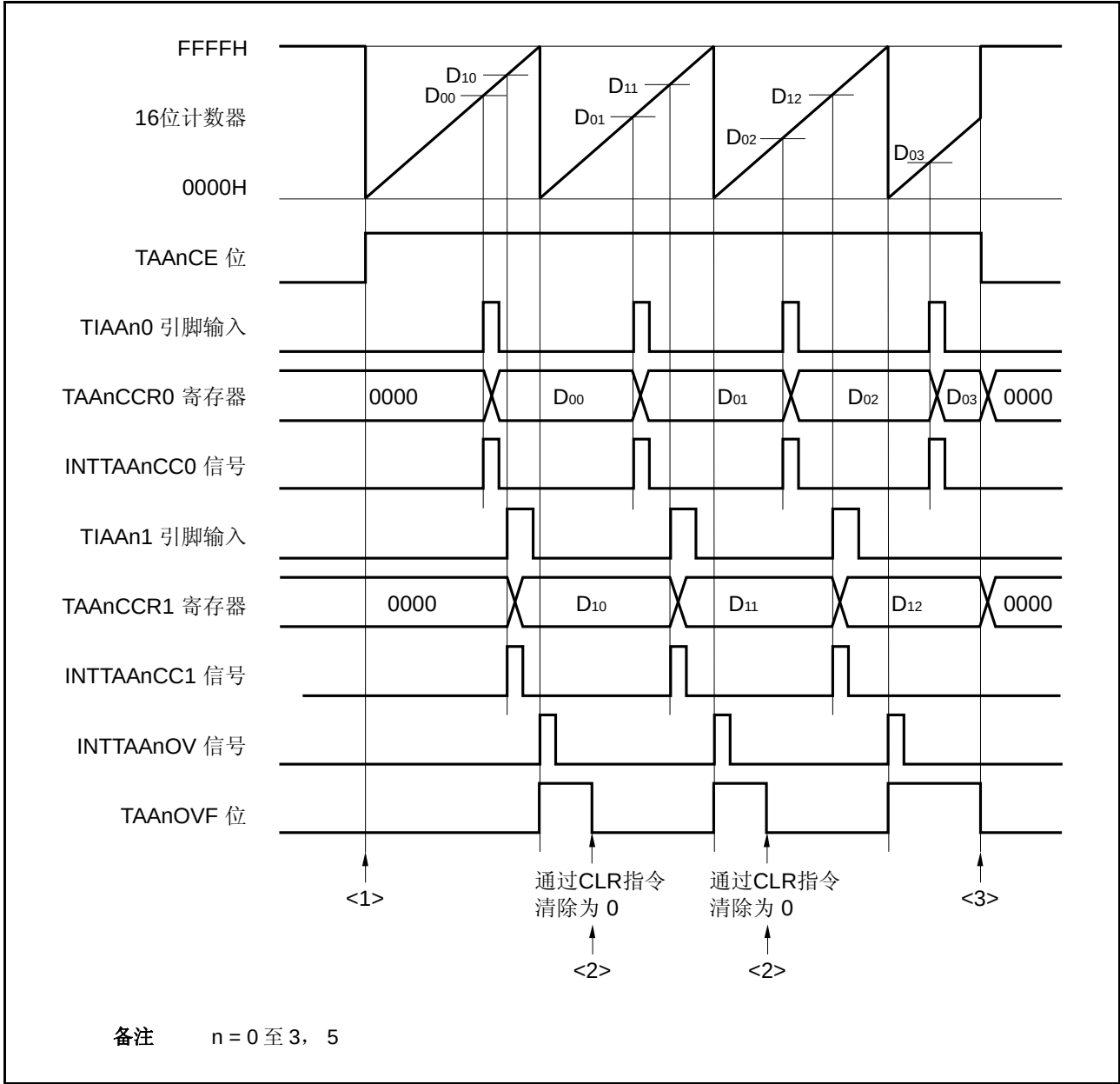
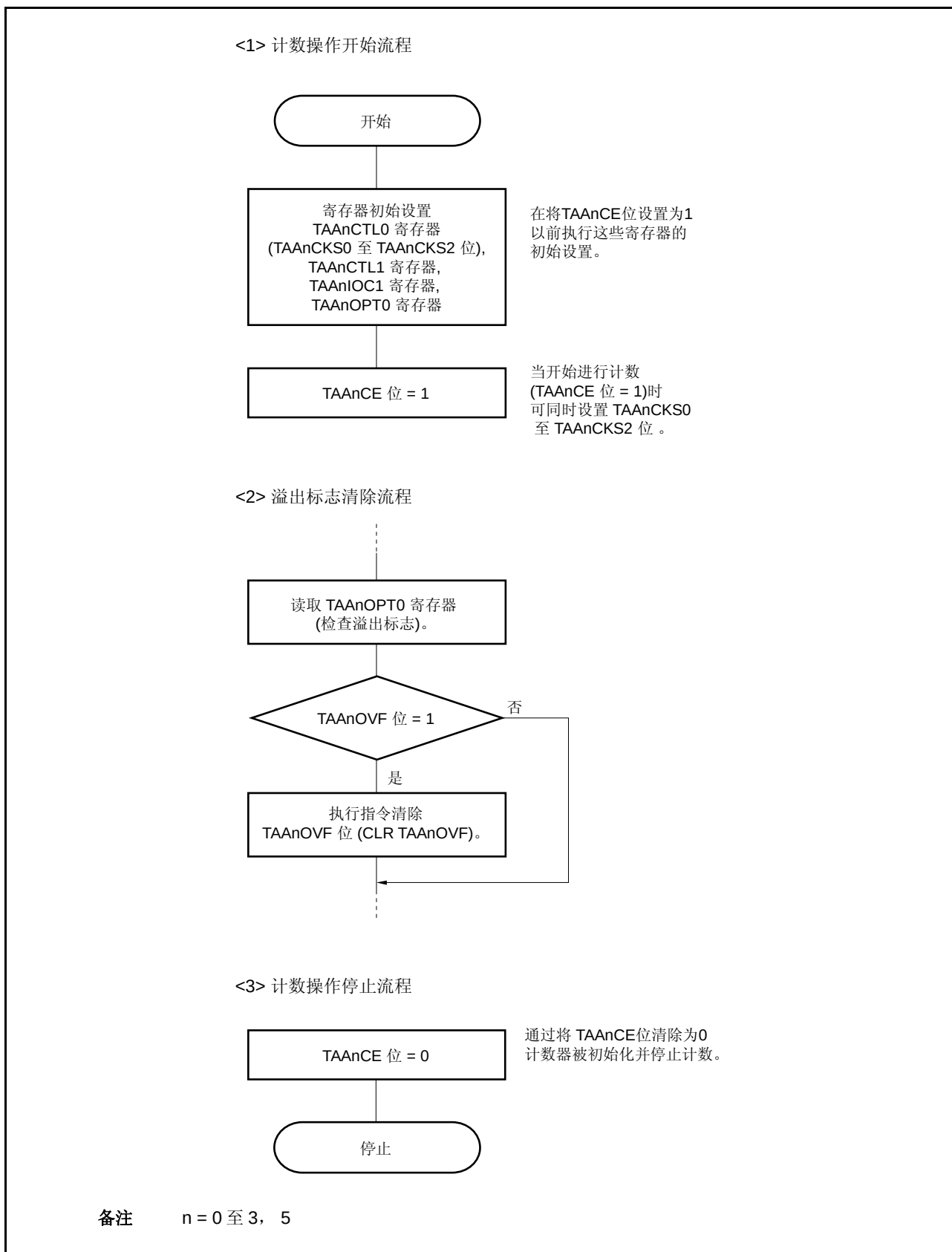


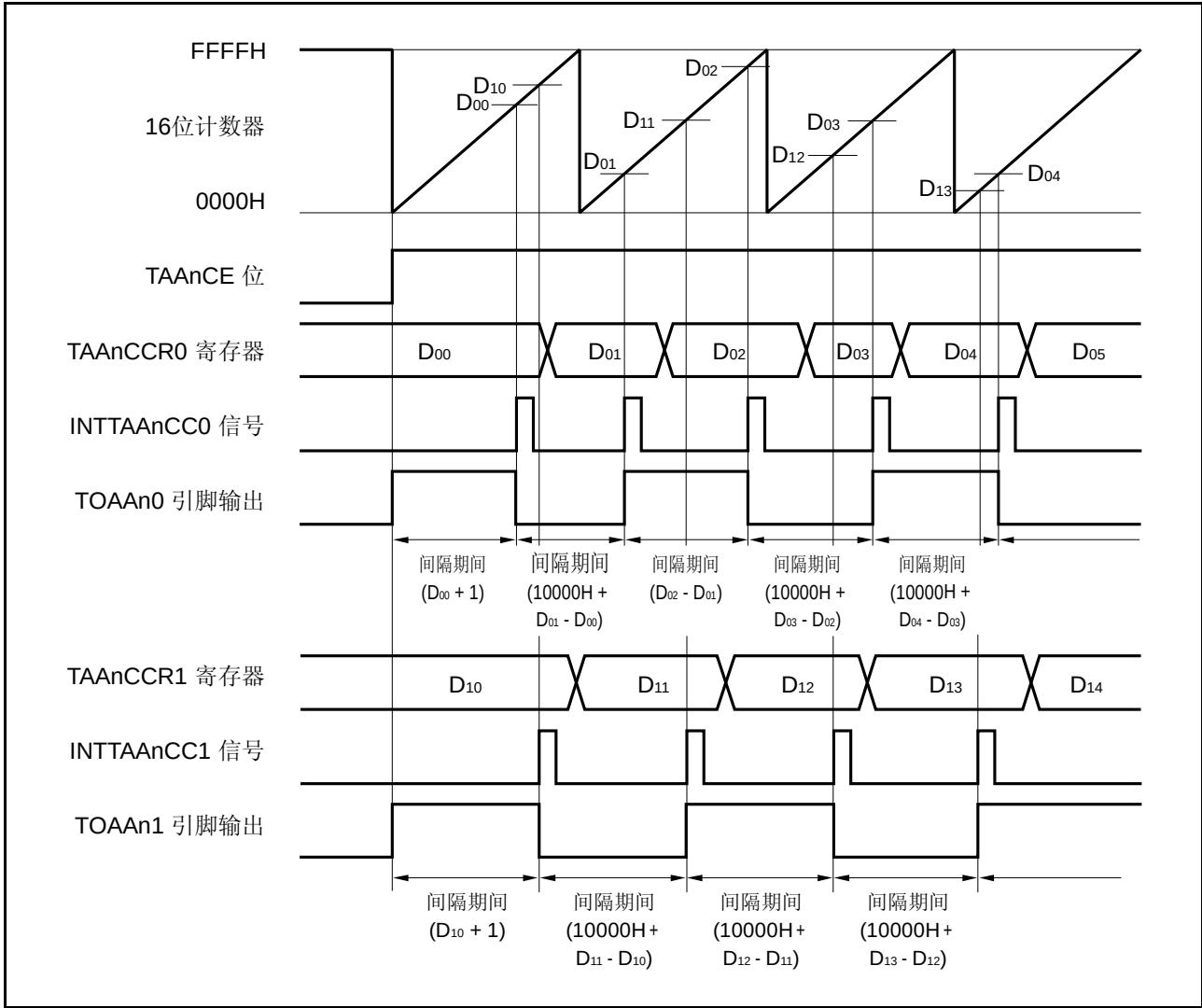
图 7-38. 自由运行定时器模式时的软件处理流程（捕获功能）（2/2）



(2) 自由运行定时器模式时的工作时序

(a) 将 TAAAnCCRm 寄存器用作比较寄存器的间隔操作

当 16 位定时器/事件计数器 AA 用作间隔定时器，而 TAAAnCCRm 寄存器用作比较寄存器时，在每次检测到 INTTAAAnCCm 信号后，都需要利用软件操作设置一个比较值以产生下一个中断请求信号。



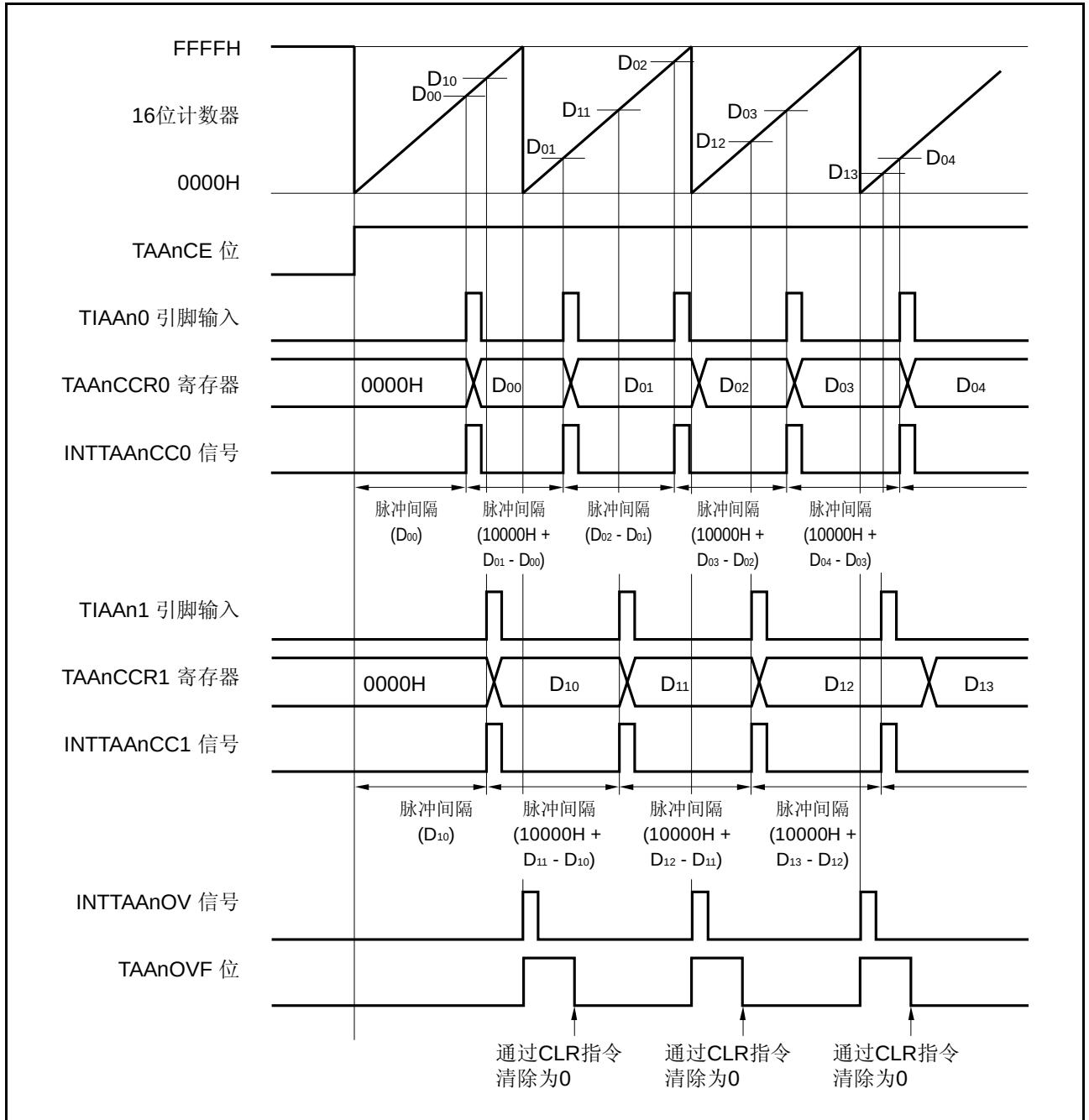
在自由运行定时器模式下进行时间间隔操作时，一个通道可以设置两个时间间隔。
要进行时间间隔操作，当检测到 INTTAAAnCCm 信号后，必须在中断服务程序中重置相应的 TAAAnCCRm 寄存器。
重置 TAAAnCCRm 寄存器的设置值可由下式计算，其中“Dm”是间隔时段：

比较寄存器默认值： $D_m - 1$
比较寄存器的第二次及其以后的设置值：上次设置值 + D_m
(若计算结果大于 FFFFH，则从结果中减去 10000H 并将此值设置给寄存器。)

备注 $m = 0, 1$
 $n = 0$ 至 $3, 5$

(b) 将 TAAAnCCRm 用作捕获寄存器的脉宽测量

在用 TAAAnCCRm 寄存器作为捕获寄存器进行脉宽测量时，每次检测到 INTTAAAnCCm 信号时读取捕获寄存器及计算时间间隔，都需要进行软件处理。



在自由运行定时器模式下执行脉宽测量时，可以用一个通道测量两个脉冲宽度。

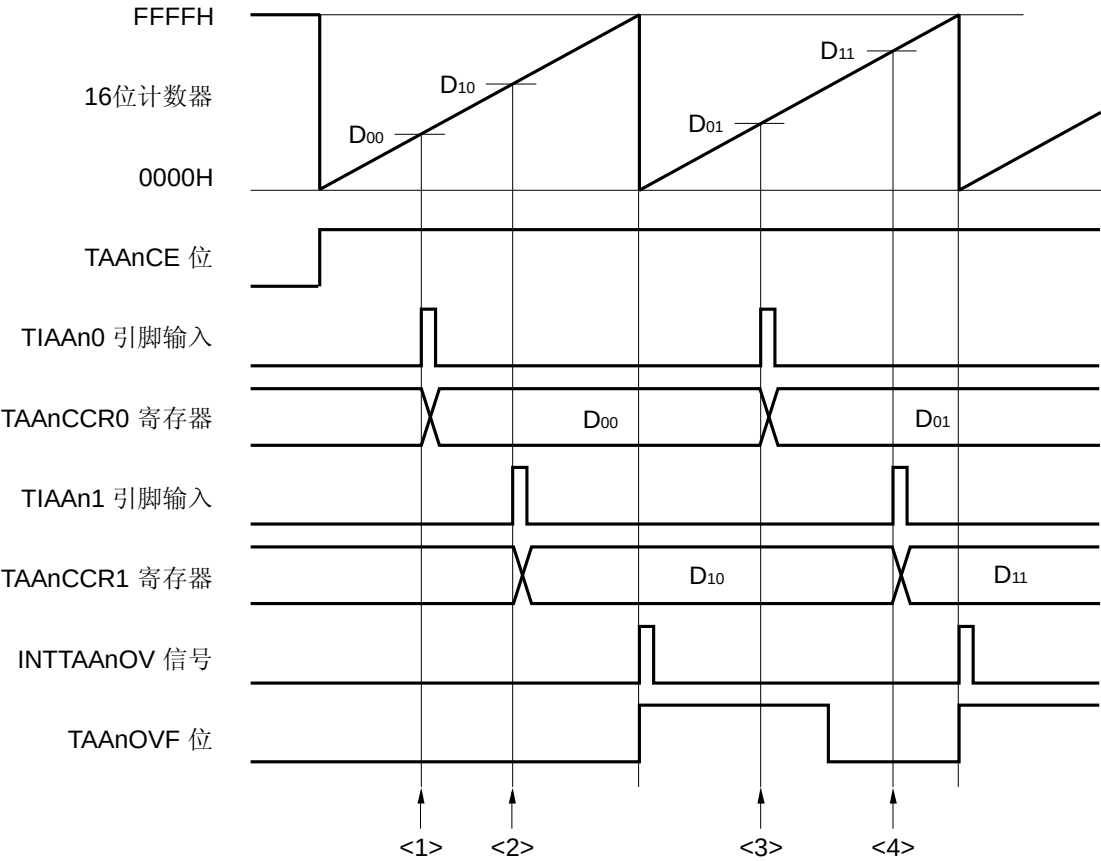
要测量脉冲宽度的话，可以通过与 INTTAAAnCCm 信号同步读取 TAAAnCCRm 寄存器的值，并计算本次读取值与先前读取值之间的差，计算得到脉冲宽度。

备注 $m = 0, 1$
 $n = 0$ 至 $3, 5$

(c) 当使用两个捕获寄存器时溢出的处理

当使用两个捕获寄存器时必须格外小心。首先，如下所示，是一个不正确处理的例子。

使用两个捕获寄存器时处理错误的例子



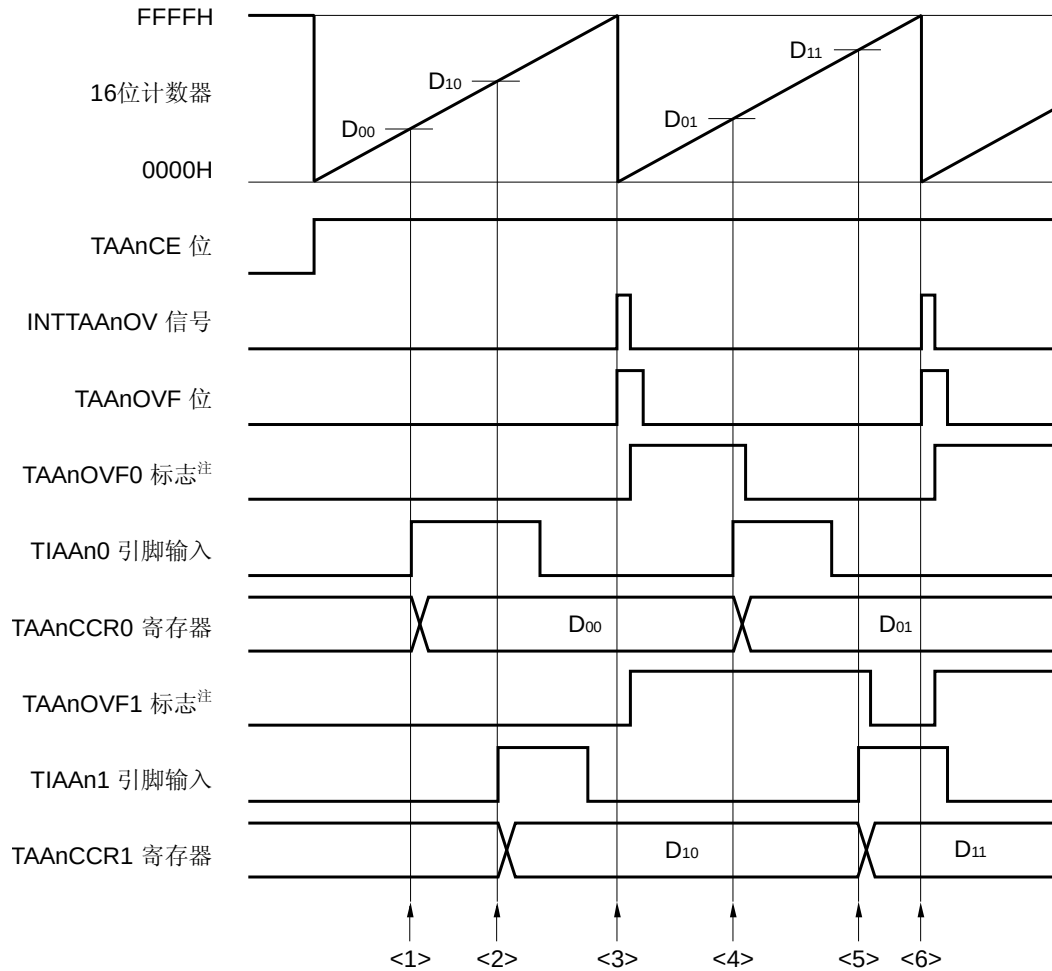
在自由运行定时器模式下测量两个脉冲宽度时，可能出现下列问题。

- <1> 读取 TAAAnCCR0 寄存器（设置 TIAAn0 引脚输入的默认值）。
- <2> 读取 TAAAnCCR1 寄存器（设置 TIAAn1 引脚输入的默认值）。
- <3> 读取 TAAAnCCR0 寄存器。
读溢出标志。若溢出标志为 1，将其清为 0。
因为溢出标志为 1，所以脉冲宽度可以用 $(10000H + D_{01} - D_{00})$ 计算。
- <4> 读取 TAAAnCCR1 寄存器。
读溢出标志。因为此标志在<3>中已清零，所以，读取到 0。
因为溢出标志为 0，所以脉冲宽度可以用 $(D_{11} - D_{10})$ 计算。（不正确）

在使用两个捕获寄存器时，如果溢出标志由一个捕获寄存器清为 0，则另一个捕捉寄存器可能不会得到正确的脉冲宽度。

在使用两个捕获寄存器时，使用软件解决。关于如何使用软件，示例如下。

使用两个捕获寄存器时的例子（使用溢出中断）



注 由软件对内部 RAM 中的 TAAAnOVF0 和 TAAAnOVF1 标志进行设置。

<1> 读取 TAAAnCCR0 寄存器（设置 TIAAn0 引脚输入的默认值）。

<2> 读取 TAAAnCCR1 寄存器（设置 TIAAn1 引脚输入的默认值）。

<3> 发生溢出。在溢出中断服务程序中将 TAAAnOVF0 和 TAAAnOVF1 标志设置为 1，并将溢出标志清为 0。

<4> 读取 TAAAnCCR0 寄存器。

读取 TAAAnOVF0 标志。若 TAAAnOVF0 标志为 1，将其清为 0。

因为 TAAAnOVF0 标志为 1，所以脉冲宽度可以用 $(10000H + D_{01} - D_{00})$ 计算。

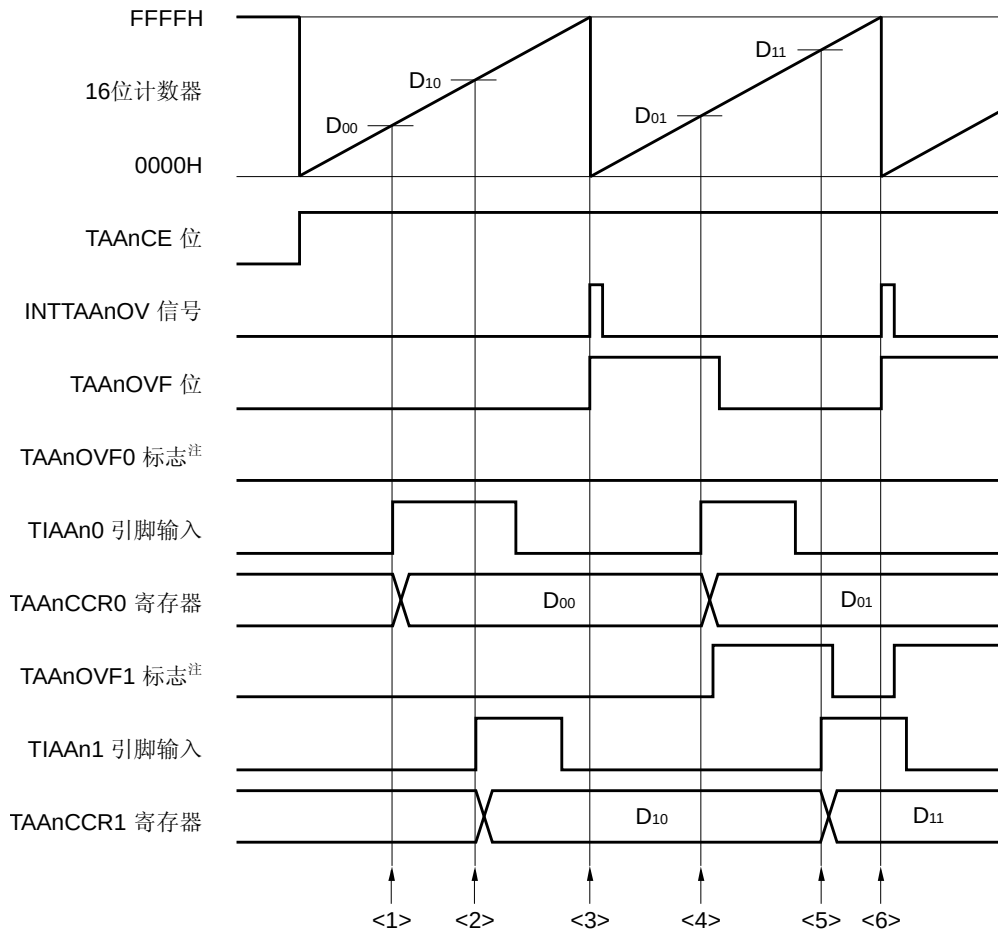
<5> 读取 TAAAnCCR1 寄存器。

读取 TAAAnOVF1 标志。如果 TAAAnOVF1 标志为 1，则将其清为 0（TAAAnOVF0 标志在<4>中已清零，而 TAAAnOVF1 标志仍然保持为 1）。

因为 TAAAnOVF1 标志为 1，所以脉冲宽度可以用 $(10000H + D_{11} - D_{10})$ 计算。（正确）

<6> 重复<3>。

使用两个捕获寄存器时的例子（不使用溢出中断）



注 由软件对内部 RAM 中的 TAAAnOVF0 和 TAAAnOVF1 标志进行设置。

<1> 读取 TAAAnCCR0 寄存器（设置 TIAAn0 引脚输入的默认值）。

<2> 读取 TAAAnCCR1 寄存器（设置 TIAAn1 引脚输入的默认值）。

<3> 发生溢出。软件不进行处理。

<4> 读取 TAAAnCCR0 寄存器。

读溢出标志。如果溢出标志为 1，则只需设置 TAAAnOVF1 标志为 1，并清除溢出标志为 0。

因为溢出标志为 1，所以脉冲宽度可以用 $(10000H + D_{01} - D_{00})$ 计算。

<5> 读取 TAAAnCCR1 寄存器。

读溢出标志。因为溢出标志在<4>中清零了，所以读取 0。

读取 TAAAnOVF1 标志。若 TAAAnOVF1 标志为 1，将其清为 0。

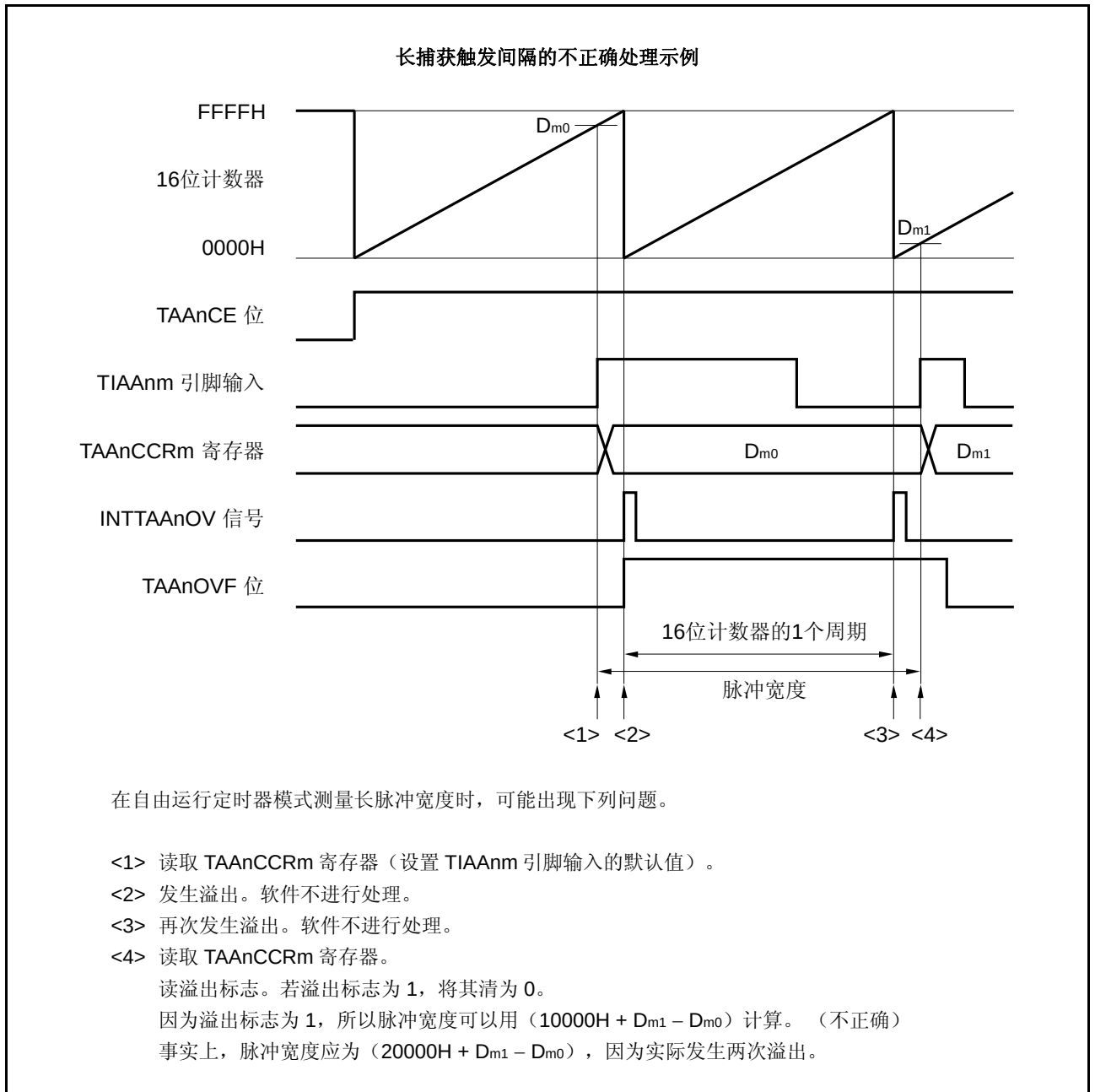
因为 TAAAnOVF1 标志为 1，所以脉冲宽度可以用 $(10000H + D_{11} - D_{10})$ 计算。（正确）

<6> 重复<3>

备注 $n = 0$ 至 3, 5

(d) 长捕获触发间隔时溢出的处理

如果脉冲宽度大于 16 位计数器的一个计数循环，则必须加以注意，因为从第一个捕获触发到下一个之间可能发生不止一次溢出。首先，如下所示，是一个不正确处理的例子。

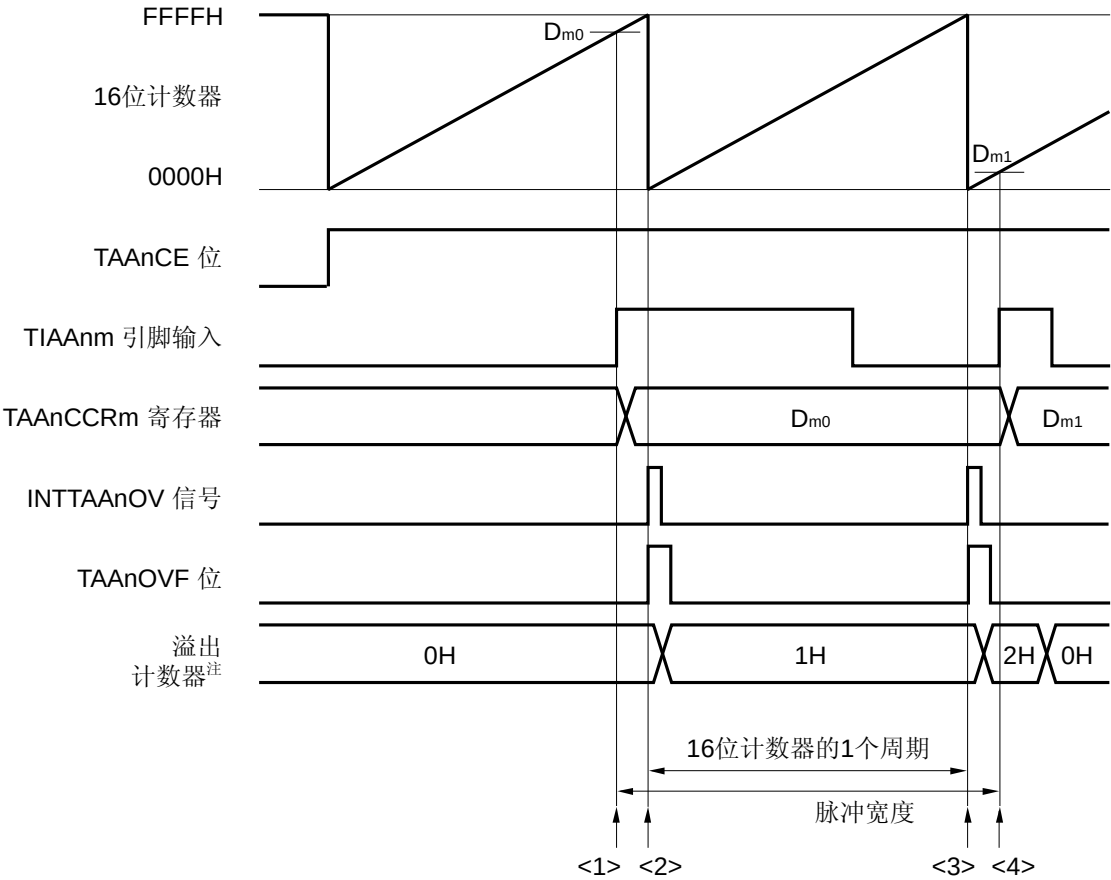


如果长捕获触发间隔期间，发生了两次或更多次的溢出，则无法获得正确的脉冲宽度。

如果捕获触发间隔为长间隔（相对于计数循环），应减慢计数时钟以延长 16 位计数器的循环周期，或者使用软件加以解决。关于如何使用软件，示例如下。

备注 m = 0, 1
 n = 0 至 3, 5

长捕获触发间隔时的举例



注 内部 RAM 中的溢出计数器由软件任意设置。

<1> 读取 TAAAnCCRm 寄存器（设置 TIAAnm 引脚输入的默认值）。

<2> 发生溢出。增加溢出计数器的值，并在溢出中断服务程序中把溢出标志清为 0。

<3> 再次发生溢出。增加（+1）溢出计数器的值，并在溢出中断服务程序中把溢出标志清为 0。

<4> 读取 TAAAnCCRm 寄存器。

读取溢出计数器。

→ 当溢出计数器为“N”时，脉冲宽度可用 $(N \times 10000H + D_{m1} - D_{m0})$ 计算。

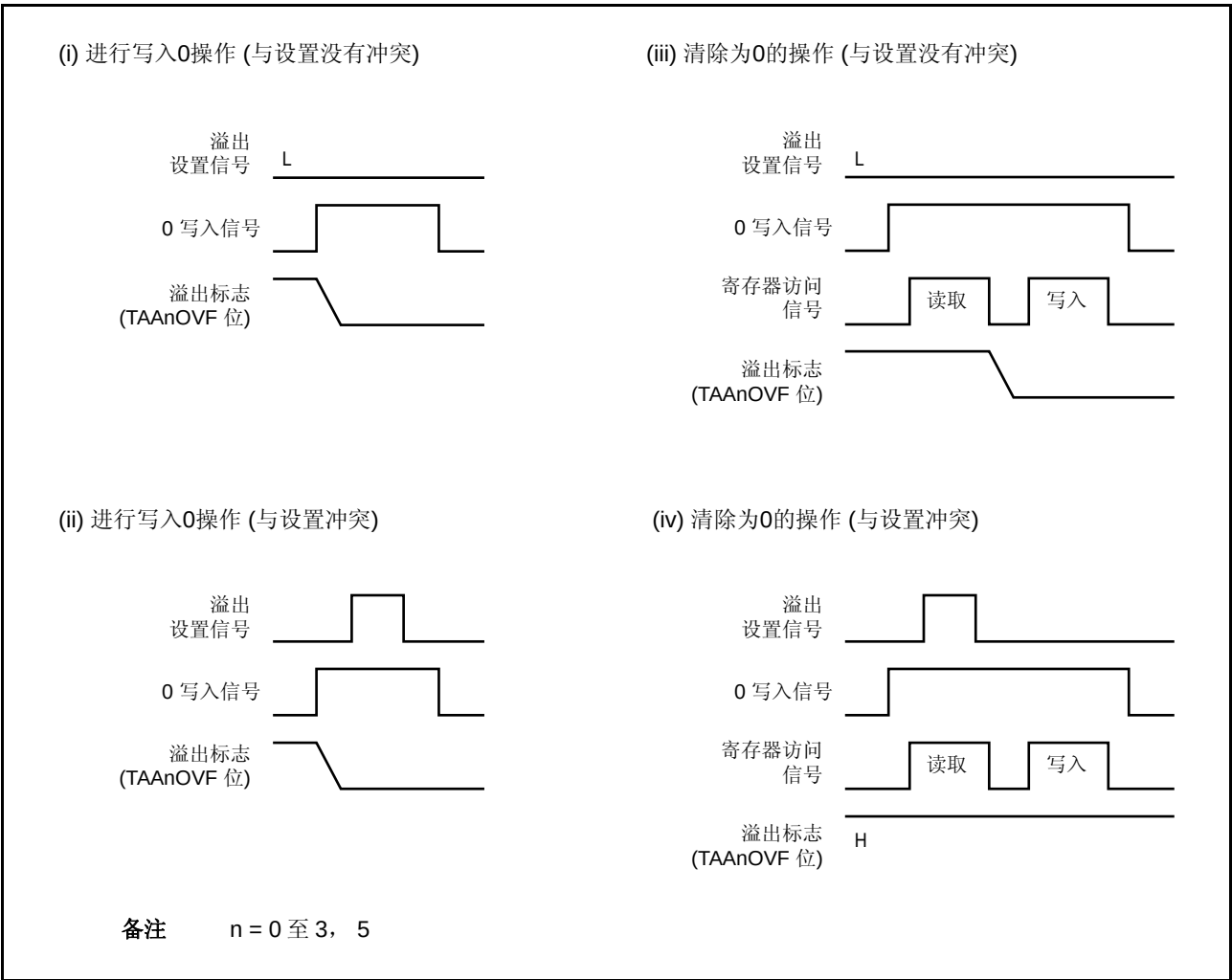
在本例中，脉冲宽度为 $(20000H + D_{m1} - D_{m0})$ ，因为溢出发生了两次。

最后，溢出计数器清零（0H）。

备注 m = 0, 1
n = 0 至 3, 5

(e) 清除溢出标志

通过用 CLR 指令将 TAA_nOVF 位清为 0 以及将 8 位数据（位 0 是 0）写入 TAA_nOPT0 寄存器，可以将溢出标志清 0。要准确检测溢出，当 TAA_nOVF 位为 1 时读取 TP0OVF 位，然后，用位操作指令将溢出标志清零。



要将溢出标志清为 0，应读取溢出标志，检查是否被设置为 1，并用 CLR 指令把它清零。若在未检查溢出标志是否为 1 的情况下把 0 写入溢出标志中，则溢出的设置信息可能因为写入 0 而被清除（上图中的（ii））。因此，即使实际发生了溢出，软件也可能会判断为无溢出。

如果用 CLR 指令将溢出标志清为 0 时，CLR 指令的执行与当前发生的溢出相冲突，则溢出标志即使在清零指令执行后也会保持置位（1）。

7.5.7 脉冲宽度测量模式 (TAA_nMD2 至TAA_nMD0 位= 110)

在脉冲宽度测量模式下，当 TAA_nCTL0.TAA_nCE 位被置为 1 时，16 位计时器/事件计数器 AA 开始计数。每次检测到 TIAAn_m 引脚输入信号的有效边沿时，16 位计数器的计数值就存入 TIAAn_m 寄存器中，且 16 位计数器清零为 0000H。

在出现捕获中断请求信号（INTTAAAnCCm）之后，可以通过读取 TAAAnCCRm 寄存器，测量有效边缘的间隔。

选择 TIAAn0 或 TIAAn1 引脚作为捕获触发信号输入引脚。使用 TAAAnIOC1 寄存器将未使用的引脚指定为“无边沿检测”引脚。

在用外部时钟作为计数时钟时，测量 TIAAn1 引脚输入的脉冲宽度，因为外部时钟固定在 TIAAn0 引脚上。此时，把 TAAAnOC1.TAAAnIS1 和 TAAAnOC1.TAAAnIS0 位清零为 00（捕获触发输入（TIAAn0 引脚）：无边沿检测）。

图 7-39. 脉冲宽度测量模式的配置

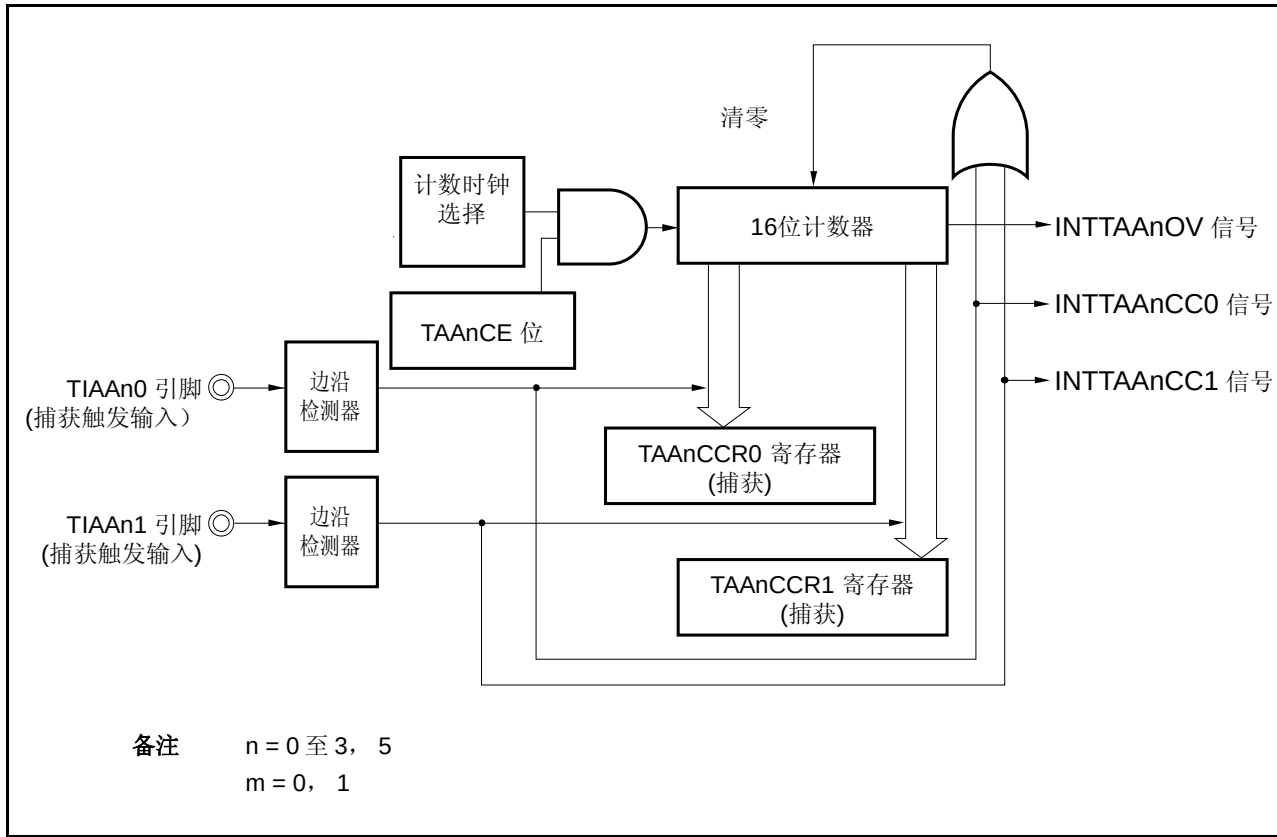
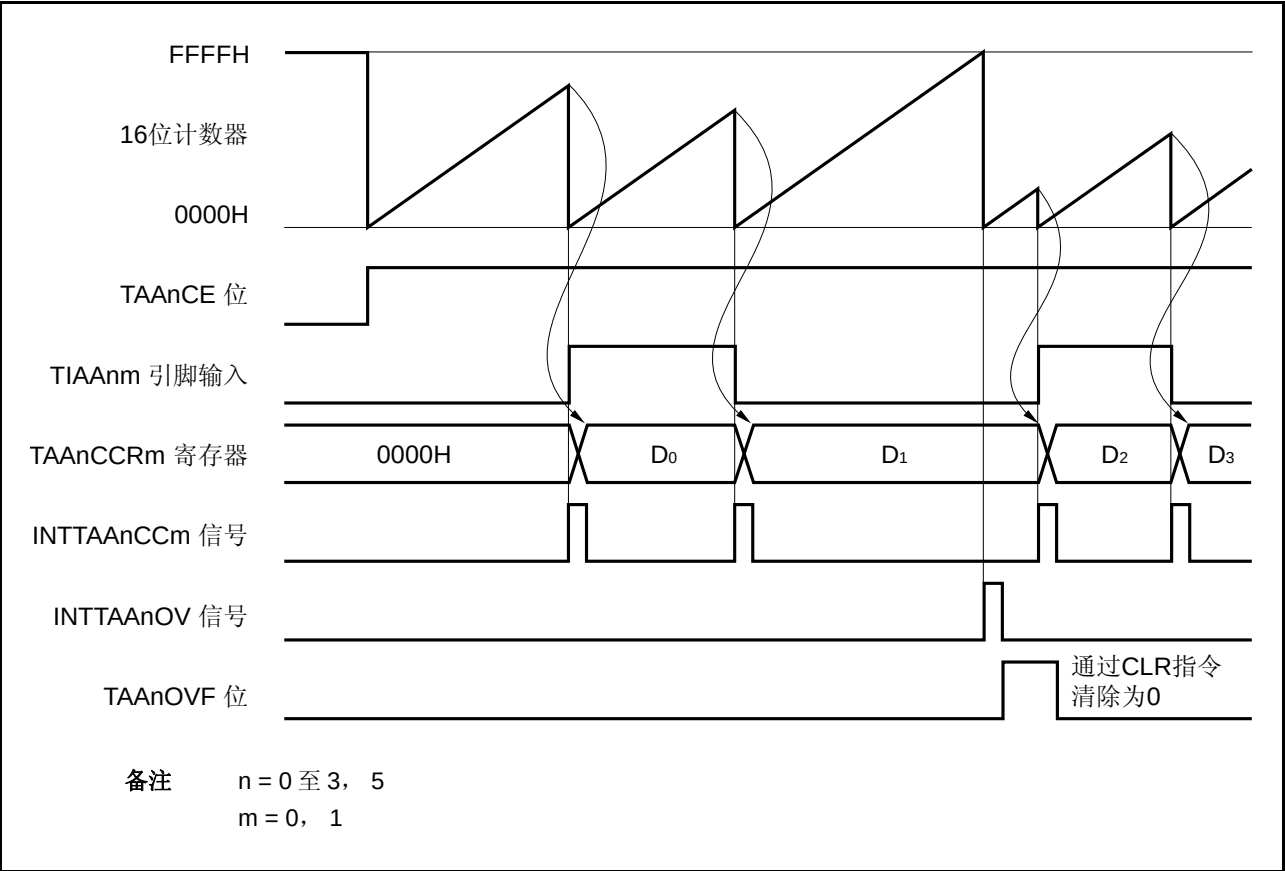


图 7-40. 脉冲宽度测量模式的基本时序



当 TAAAnCE 位被置为 1 时，16 位计数器开始计数。此后检测到 TIAAnm 引脚输入信号的有效边沿时，16 位计数器的计数值存入 TAAAnCCRm 寄存器中，16 位计数器清零为 0000H，并生成一个捕获中断请求信号（INTTAAAnCCm）。脉冲宽度计算如下。

脉冲宽度= 捕获值×计数时钟周期

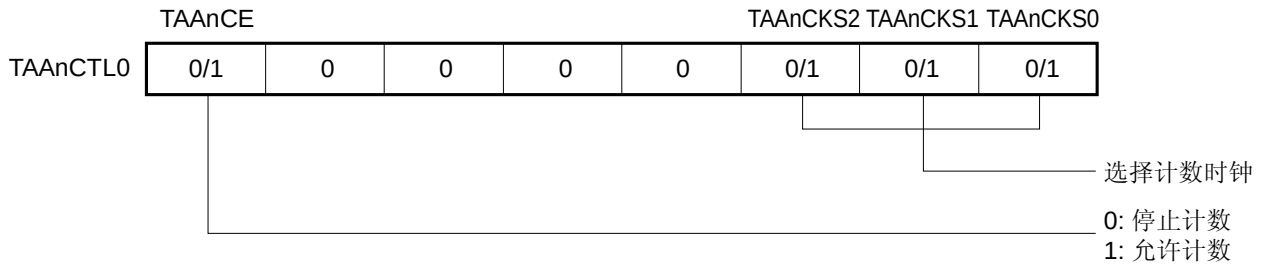
即使 16 位计数器一直计数到 FFFFH，TIAAnm 引脚输入信号一直没有有效边沿，仍然会在下一个计数时钟时产生溢出中断请求信号（INTTAAAnOV），且计数器清零为 0000H，计数器继续计数。此时，溢出标志（TAAAnOPT0.TAAAnOVF 位）也被置为 1。使用软件执行 CLR 指令将溢出标志清为 0。若溢出标志设置为 1，则脉冲宽度可计算如下。

脉冲宽度=（10000H×TAAAnOVF 位置（1）的次数+捕获值）×计数时钟周期

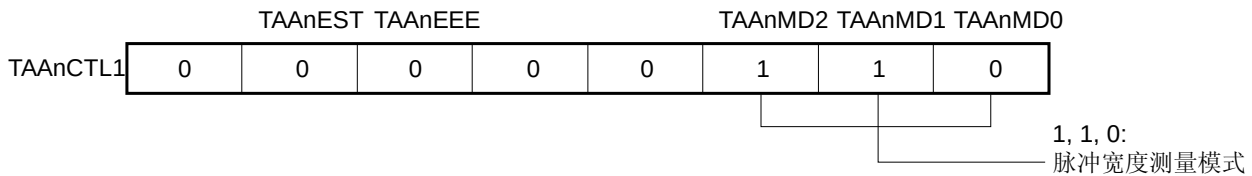
备注 n = 0 至 3, 5
m = 0, 1

图 7-41. 脉冲宽度测量模式下寄存器的设置

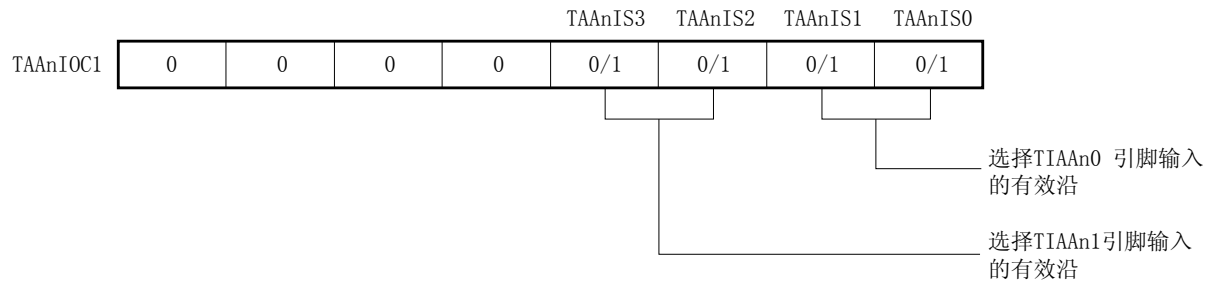
(a) TAA_n 控制寄存器 0 (TAA_nCTL0)



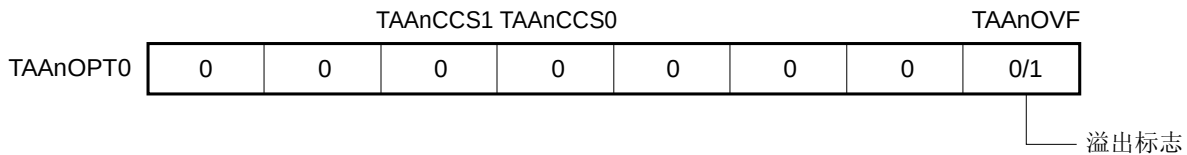
(b) TAA_n 控制寄存器 1 (TAA_nCTL1)



(c) TAA_n I/O 控制寄存器 1 (TAA_nIOC1)



(d) TAA_n 选项寄存器 0 (TAA_nOPT0)



(e) TAA_n 计数器读取缓冲寄存器 (TAA_nCNT)

通过读取 TAA_nCNT 寄存器，可以读出 16 位计数器的值。

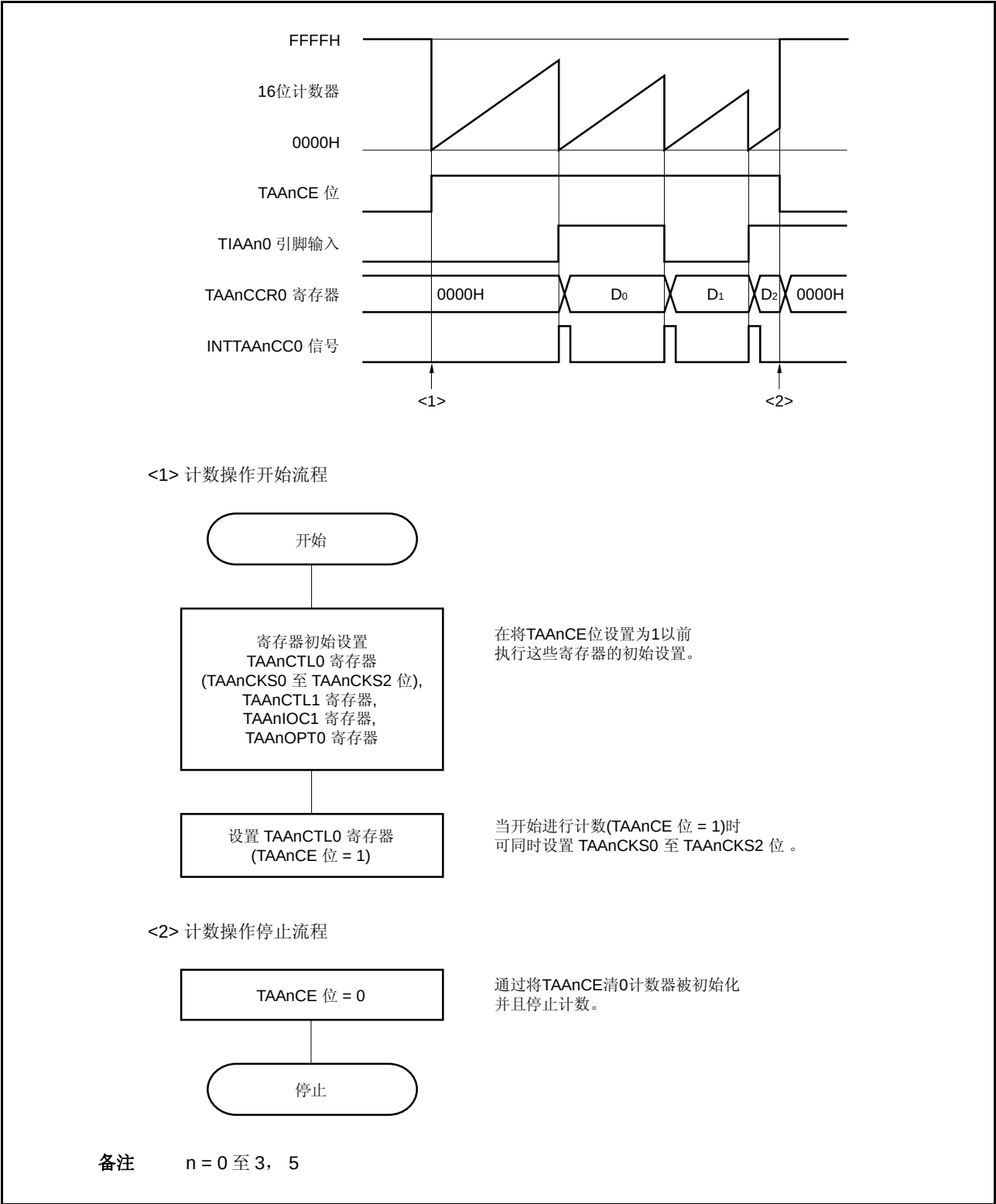
(f) TAA_n 捕获/比较寄存器 0 和 1 (TAA_nCCR0 和 TAA_nCCR1)

当检测到 TIAAn_m 引脚输入信号的有效边沿时，这些寄存器保存计数器的当前值。

- 备注**
1. TAA_n 输入/输出控制寄存器 0 (TAA_nIOC0) 和 TAA_n 输入/输出控制寄存器 2 (TAA_nIOC2) 不会在脉冲宽度测量模式下使用。
 2. m = 0, 1
n = 0 至 3, 5

(1) 脉冲宽度测量模式时的工作流程

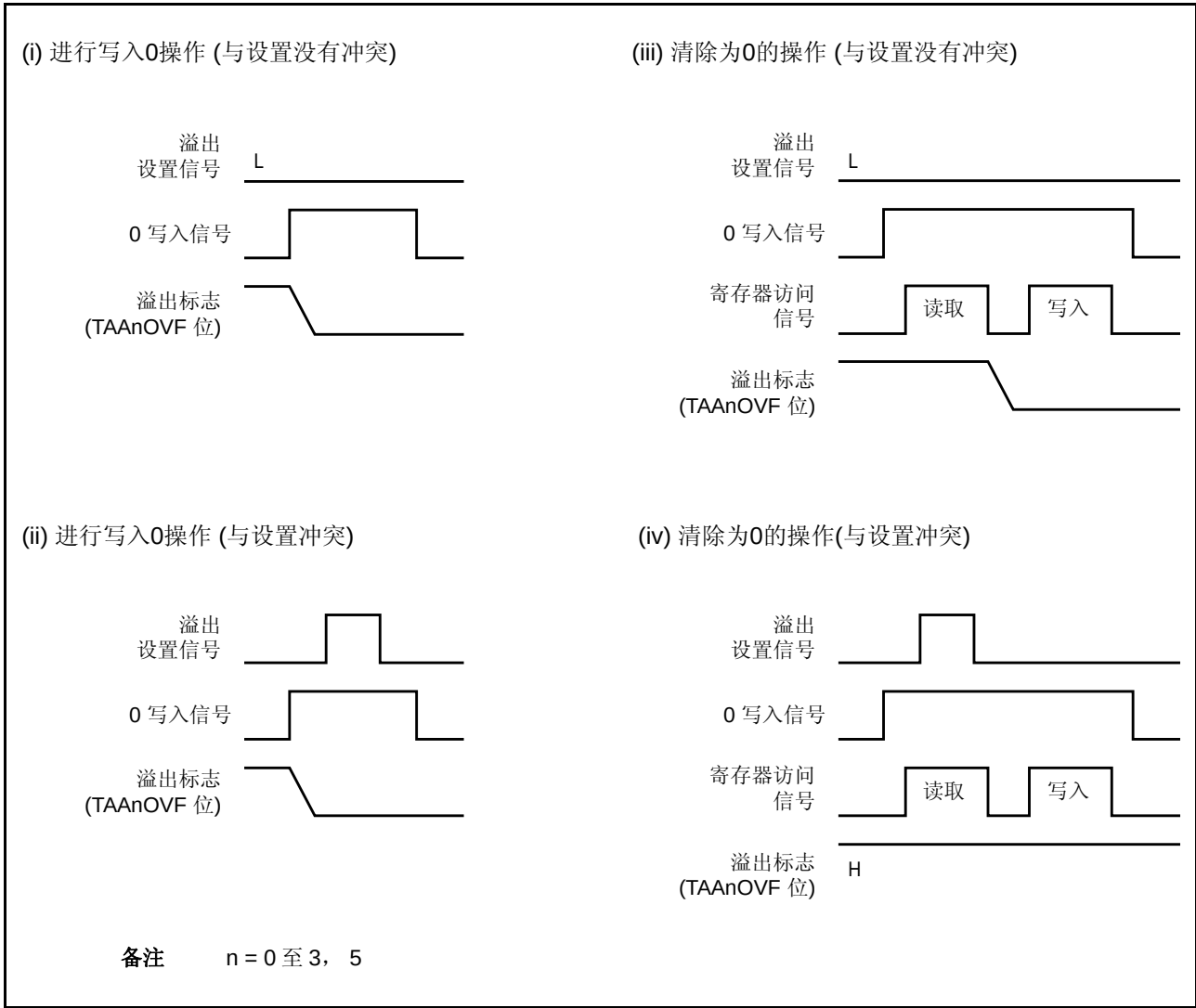
图 7-42. 脉冲宽度测量模式时的软件处理流程



(2) 脉冲宽度测量模式时的工作时序

(a) 清除溢出标志

通过用 CLR 指令将 TAA_nOVF 位清为 0 以及将 8 位数据 (位 0 是 0) 写入 TAA_nOPT0 寄存器, 可以将溢出标志清 0。要准确检测溢出, 当 TAA_nOVF 位为 1 时读取 TP0OVF 位, 然后, 用位操作指令将溢出标志清零。



要将溢出标志清为 0, 应读取溢出标志, 检查是否被设置为 1, 并用 CLR 指令把它清零。若在未检查溢出标志是否为 1 的情况下把 0 写入溢出标志中, 则溢出的设置信息可能因为写入 0 而被清除 (上图中的 (ii))。因此, 即使实际发生了溢出, 软件也可能会判断为无溢出。

如果用 CLR 指令将溢出标志清为 0 时, CLR 指令的执行与当前发生的溢出相冲突, 则溢出标志即使在清零指令执行后也会保持置位 (1)。

7.5.8 定时器输出操作

下表显示了 TOAAn0 和 TOAAn1 引脚的操作和输出电平。

表 7-5. 各种模式下的定时器输出控制

操作模式	TOAAn1 引脚	TOAAn0 引脚
间隔定时器模式	方波输出	
外部事件计数模式	方波输出	—
外部触发脉冲输出模式	外部触发脉冲输出	方波输出
单次脉冲输出模式	单次脉冲输出	
脉宽调制信号输出模式	PWM（脉宽调制信号）输出	
自由运行定时器模式	方波输出（仅当使用比较功能时）	
脉冲宽度测量模式	—	

备注 n = 0 至 3, 5

表 7-6. TOAAn0 至 TOAAn1 引脚受控于定时器输出控制位的真值表

TAAAnIOC0.TAAAnOLm 位	TAAAnIOC0.TAAAnOEm 位	TAAAnCTL0.TAAAnCE 位	TOAAnm 引脚的电平
0	0	×	低电平输出
	1	0	低电平输出
		1	计数前为低电平，开始计数后为高电平
1	0	×	高电平输出
	1	0	高电平输出
		1	计数前为高电平，开始计数后为低电平

备注 n = 0 至 3, 5
m = 0, 1

7.6 定时器调谐操作功能

定时器 AA 和定时器 AB 有定时器调谐操作功能。

定时器调谐操作功能用于调谐 V850ES/JG3-H 和 V850ES/JH3-H 的内部定时器，因此，从定时器的捕获或比较寄存器的数目（即从定时器的定时器输出数目和比较匹配中断的数目）可以添加到主定时器的输出上。可调谐的定时器列表于表 7-7。

表 7-7. 定时器的调谐操作模式

主定时器	从定时器
TAA1	TAA0
TAA3	TAA2
TAB0	TAA5

调谐操作功能有以下模式：

- 脉宽调制信号输出模式
- 自由定时器模式

图 7-43 是在脉宽调制信号输出模式下，TAA0（作为主定时器）和 TAA1（作为从定时器）的独立操作和调谐操作的举例。

图 7-43. 使用 TAA0 和 TAA1 进行独立操作与调谐操作的区别

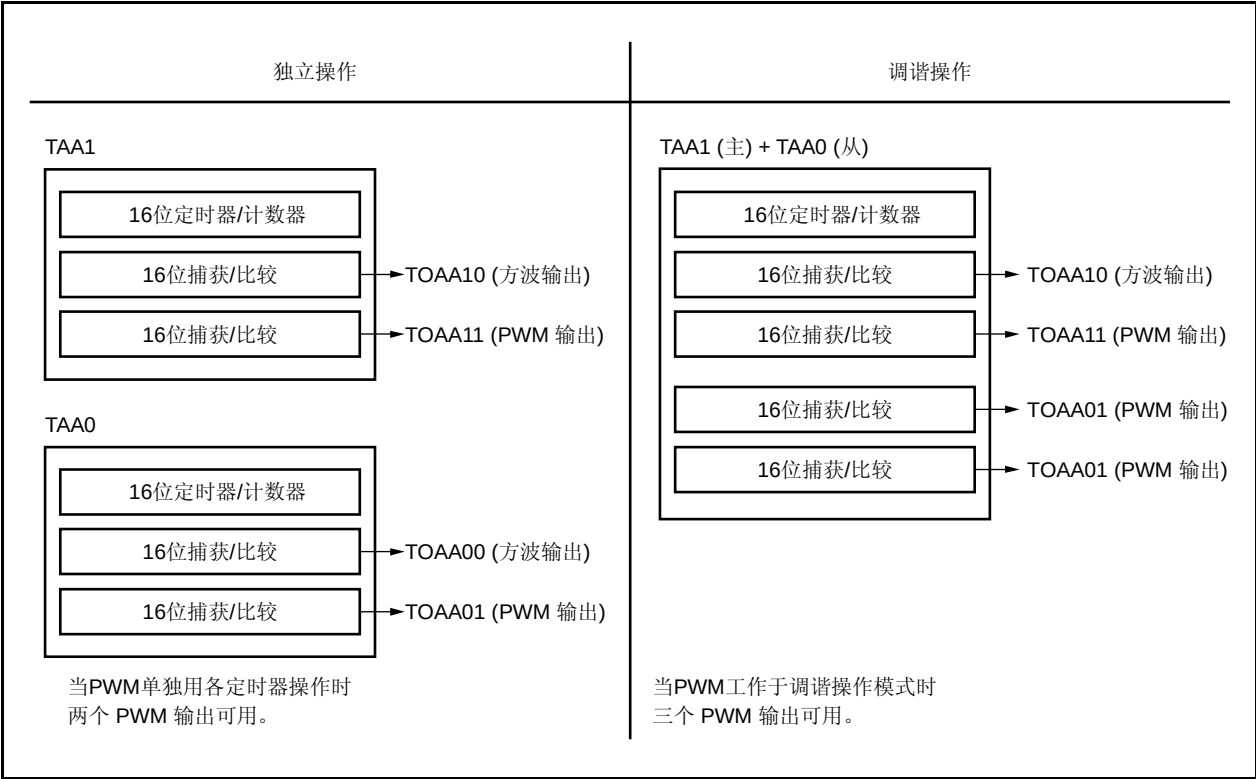


表 7-8 显示了可用于调谐操作模式的定时器模式，而表 7-9 显示了独立操作与调谐操作中定时器输出功能的区别（√：可设置，×：不可设置）。

表 7-8. 调谐操作模式下可用的定时器模式

主定时器	从定时器	自由运行定时器模式	PWM 模式
TAA1	TAA0	√	√
TAA3	TAA2	√	√
TAB0	TAA5	√	√

表 7-9. 定时器输出功能

调谐通道	定时器	引脚	自由定时器模式		PWM 模式	
			独立操作	调谐操作	独立操作	调谐操作
Ch0	TAA1 (主)	TOAA10	PPG	←	切换	←
		TOAA11	PPG	←	PWM	←
	TAA0 (从)	TOAA00	PGP	←	切换	PWM
		TOAA01	PPG	←	PWM	←
Ch1	TAA3 (主)	TOAA30	PPG	←	切换	←
		TOAA31	PPG	←	PWM	←
	TAA2 (从)	TOAA20	PPG	←	切换	PWM
		TOAA21	PPG	←	PWM	←
Ch2	TAB0 (主)	TOAB00	PPG	←	切换	←
		TOAB01 至 TOAB03	PPG	←	PWM	←
	TAA5 (从)	TOAA50	PPG	←	切换	PWM
		TOAA51	PPG	←	PWM	←

备注 从缓冲寄存器的比较寄存器传送数据的时序如下：

- PPG：CPU 写时序
- 切换，PWM，三角波 PWM：定时器计数器和比较寄存器与 TOAAn0 和 TOABm0 匹配时的时序。

7.6.1 自由运行定时器模式（在定时器调谐操作期间）

本节阐述调谐定时器操作的自由运行定时器模式。关于调谐定时器操作的组成，参见表 7-7。本节中用 TAA1 和 TAA0 的定时器调谐操作举例，显示如下。

(i) 选择捕获/比较寄存器

当定时器调谐操作的自由运行定时器模式使用 TAA1 和 TAA0 相互连接时，TAA1 的两个捕获/比较寄存器与 TAA0 的两个捕获/比较寄存器可以相结合使用。

捕获和比较寄存器怎样结合不受限制，可通过主定时器或从定时器的 TAA_nCCSn 位进行选择。当选择比较寄存器时，操作期间可以重写该比较寄存器的设定值而重写方式为随时写入（n = 0, 1）。

(ii) 溢出

如果计数器溢出，则产生主定时器的溢出中断（INTTAA1OV）且溢出标志（TAA1OVF）设置为“1”。

从定时器的溢出中断（INTTAA0OV）和溢出标志（TAA0OVF）不进行操作，并且总是处于低电平。

(1) 在自由运行定时器模式下的设置（比较功能）

[初始设置]

主定时器：TAA1CTL0.TAA1CE = 0（禁止操作）

从定时器：TAA0CTL0.TAA0CE = 0（禁止操作）

[主定时器的初始设置（TAA1）]

- TAA1CTL1.TAA1MD2 至 TAA1CTL1.TAA1MD0 = 101（自由运行定时器模式的设置）
- TAA1OPT0.TAA1CCS1 和 TAA1OPT0.TAA1CCS0 = 00（捕获/比较选择位的设置为“比较”。）
- TAA1CTL1.TAA1CKS2 和 TAA1CTL1.TAA1CKS0（计数时钟的设置（任意））
- TAA1CCR1 和 TAA1CCR0 寄存器被设置。

[从定时器的初始设置（TAA0）]

- TAA0CTL1.TAA0SYE = 1（定时器调谐操作的设置）
- TAA0CTL1.TAA0MD2 至 TAA0CTL1.TAA0MD0 = 101（自由运行定时器模式的设置）
- TAA0OPT0.TAA0CCS1 和 TAA0OPT0.TAA0CCS0 = 00（捕获/比较选择位的设置为“比较”。）
- TAA0CCR0 和 TAA0CCR1 寄存器被设置。

备注 主定时器和从定时器的初始设置可以按任何顺序执行。

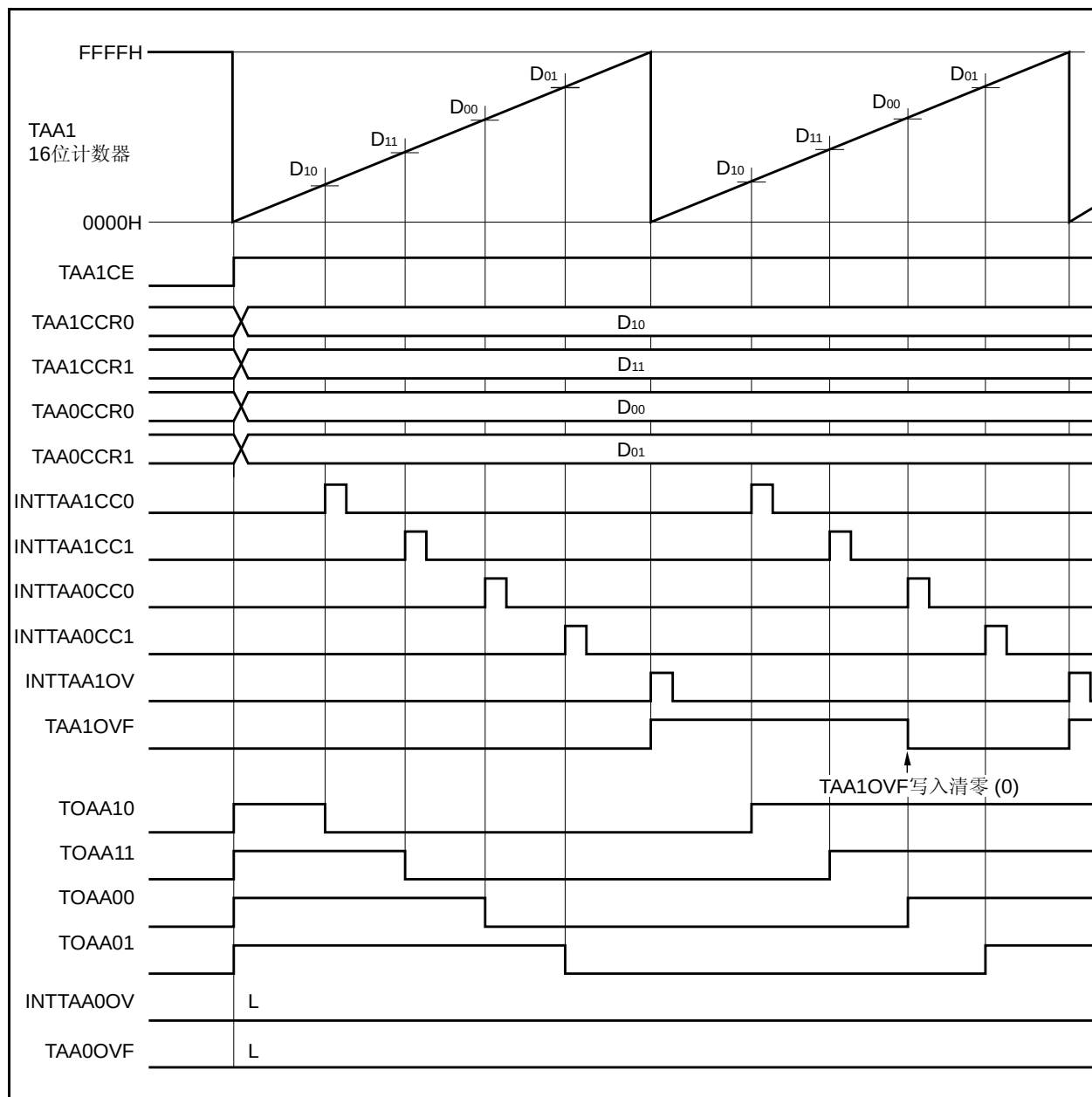
[开始计数]

- <1> 设置主定时器的 TAA1CTL0.TAA1CE 位为 1。
- <2> 开始计数。
- <3> 操作期间改变寄存器的设置。
 - 比较寄存器可被重写（随时写入）。

[结束条件]

- 设置主定时器的 TAA1CTL0.TAA1CE 位为 0。

图 7-44. 自由运行模式下的时序举例（比较功能）



(2) 在自由运行定时器模式下的设置（捕获功能）

[初始设置]

主定时器：TAA1CTL0.TAA1CE = 0（禁止操作）

从定时器：TAA0CTL0.TAA0CE = 0（禁止操作）

[主定时器的初始设置（TAA1）]

- TAA1CTL1.TAA1MD2 至 TAA1CTL1.TAA1MD0 = 101（自由运行定时器模式的设置）
- TAA1OPT0.TAA1CCS1 和 TAA1OPT0.TAA1CCS0 = 11（捕获/比较选择位的设置为“捕获”。）
- TAA1CTL1.TAA1CKS2 至 TAA1CTL1.TAA1CKS0（计数时钟的设置（任意））
- TAA1IOC1.TAA1IS3 至 TAA1IOC1.TAA1IS0（捕获触发有效沿的指定）

[从定时器的初始设置（TAA0）]

- TAA0CTL1.TAA0SYE = 1（调谐定时器操作的设置）
- TAA0CTL1.TAA0MD2 至 TAA0CTL1.TAA0MD0 = 101（自由运行定时器模式的设置）
- TAA0OPT0.TAA0CCS1 and TAA0OPT0.TAA0CCS0 = 11（捕获/比较选择位的设置为“捕获”。）
- TAA0IOC1.TAA0IS3 至 TAA0IOC1.TAA0IS0（捕获触发有效沿的指定）

备注 主定时器和从定时器的初始设置可以按任何顺序执行。

[开始计数]

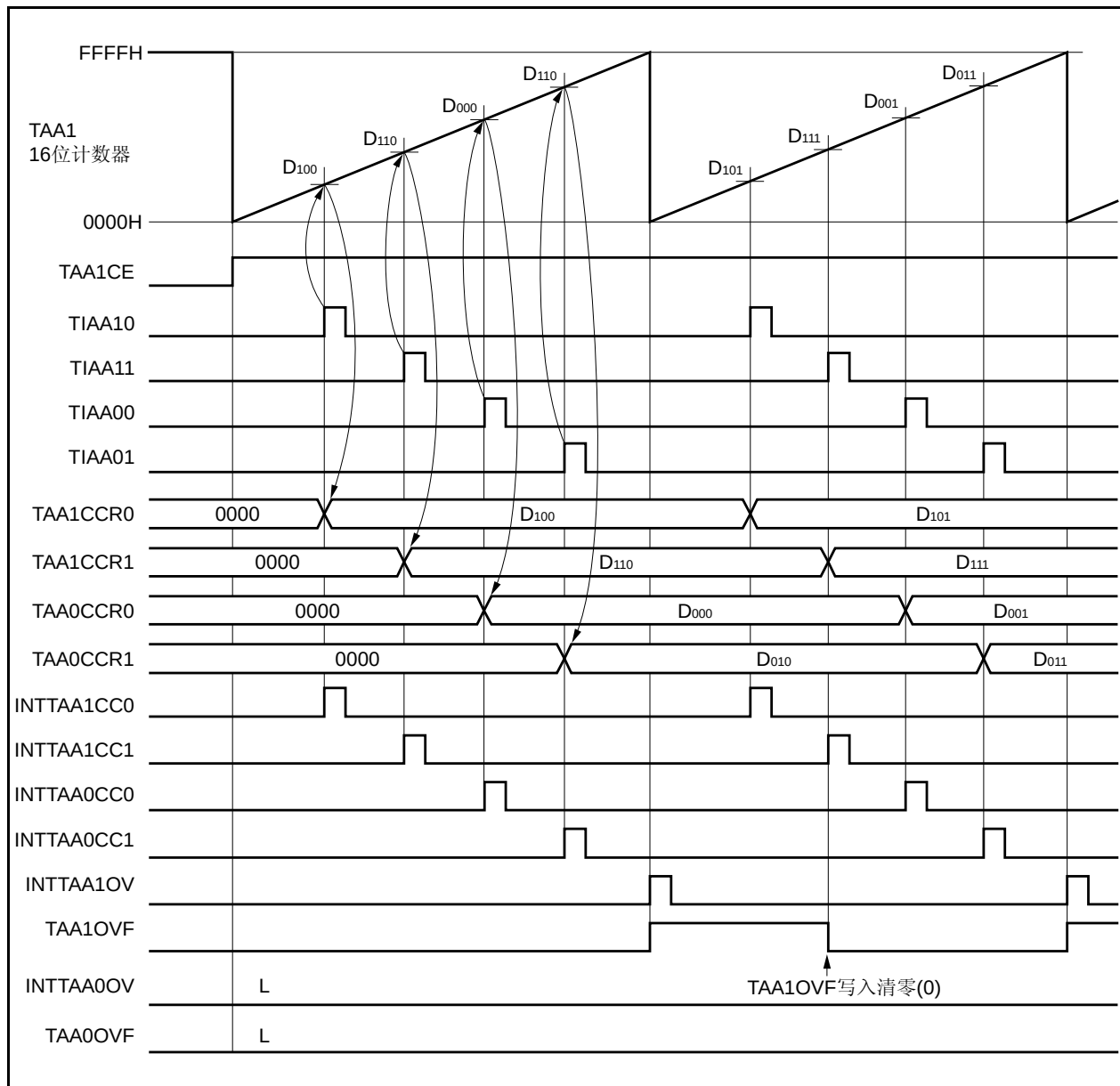
<1> 设置主定时器的 TAA1CTL0.TAA1CE 位为 1。

<2> 开始计数。

[结束条件]

- 设置主定时器的 TAA1CTL0.TAA1CE 位为 0。

图 7-45. 自由运行模式下的时序举例（捕获功能）



(3) 在自由运行定时器模式下的设置（一起使用捕获/比较功能）

以下是 TAA0 用作捕获寄存器而 TAA1 用作比较寄存器的举例说明：

[初始设置]

主定时器：TAA1CTL0.TAA1CE = 0（禁止操作）

从定时器：TAA0CTL0.TAA0CE = 0（禁止操作）

[主定时器的初始设置（TAA1）]

- TAA1CTL1.TAA1MD2 至 TAA1CTL1.TAA1MD0 = 101（自由运行定时器模式的设置）
- TAA1OPT0.TAA1CCS1 和 TAA1OPT0.TAA1CCS0 = 11（捕获/比较选择位的设置为“捕获”。）
- TAA1CTL1.TAA1CKS2 至 TAA1CTL1.TAA1CKS0（计数时钟的设置（任意））
- TAA1.TAA0IS3 至 TAA1.TAA1IS0（捕获触发有效沿的指定）

[从定时器的初始设置（TAA0）]

- TAA0CTL1.TAA0SYE = 1（定时器调谐操作的设置）
- TAA0CTL1.TAA0MD2 至 TAA0CTL1.TAA0MD0 = 101（自由运行定时器模式的设置）
- TAA0OPT0.TAA0CCS1 和 TAA0OPT0.TAA0CCS0 = 00（捕获/比较选择位的设置为“比较”。）
- TAA0CCR0 和 TAA0CCR1 寄存器被设置。

备注 主定时器和从定时器的初始设置可以按任何顺序执行。

[开始计数]

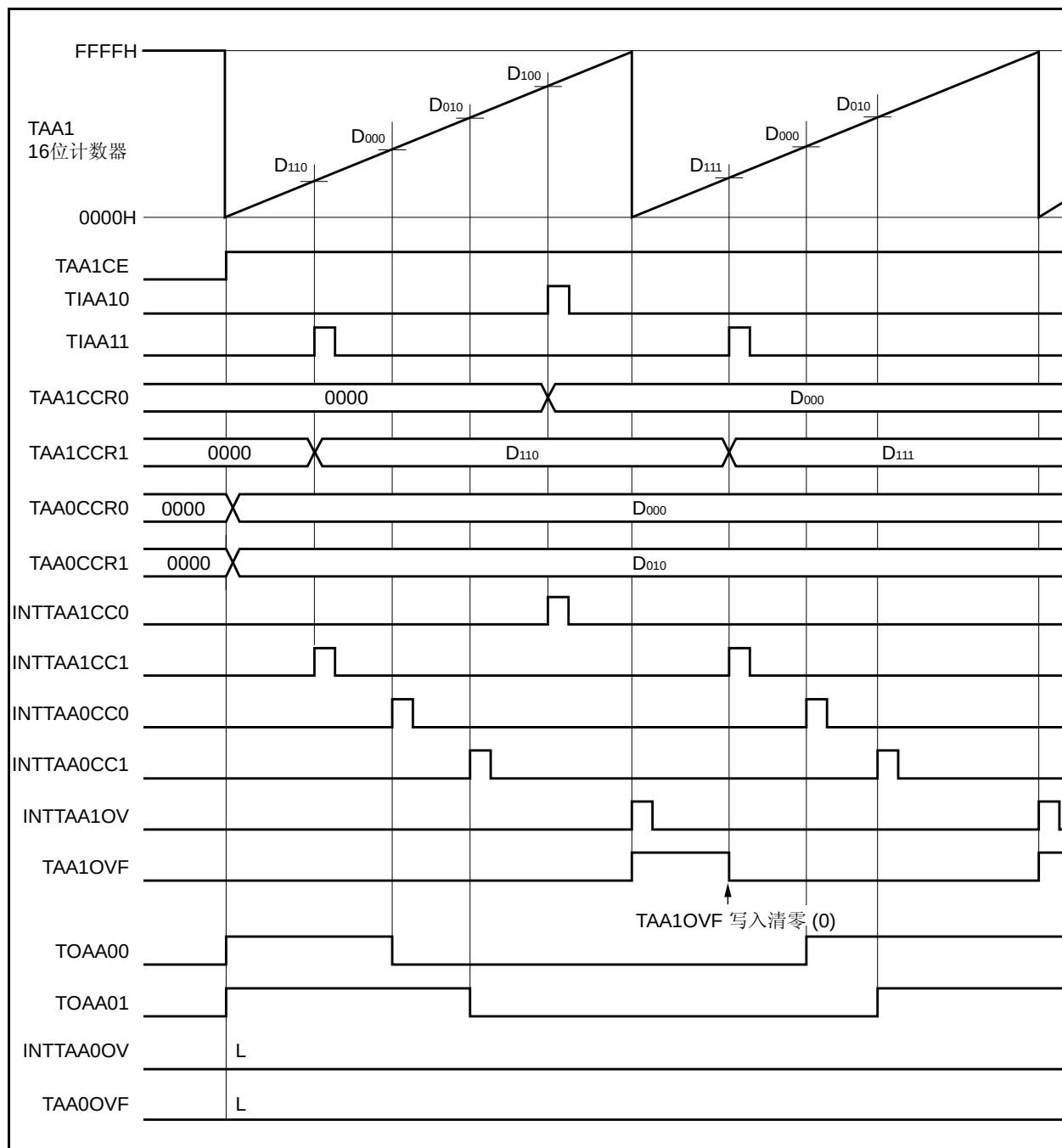
<1> 设置主定时器的 TAA1CTL0.TAA1CE 位为 1。

<2> 开始计数。

[结束条件]

- 设置主定时器的 TAA1CTL0.TAA1CE 位为 0。

图 7-46. 自由运行模式下的时序举例（一起使用捕获/比较）



7.6.2 PWM输出模式（定时器调谐操作期间）

本节阐述定时器调谐操作的 PWM 输出模式。关于定时器调谐操作的组合，参见表 7-7。本节以 TAB0 和 TAA5 的定时器调谐操作举例。

主定时器（TAB0）的 TAB0CCR0 寄存器用作比较寄存器以确定周期，主定时器（TAB0）的 TAB0CCR1，TAB0CCR2，和 TAB0CCR3 寄存器以及从定时器（TAA5）的 TAA5CCR0 和 TAA5CCR1 寄存器用作比较寄存器以确定占空比。

操作期间可以重写这些比较寄存器而重写方式为批写入。

写入主定时器（TAB0）的 TAB0CCR1 寄存器，并且主定时器和从定时器的所有比较寄存器都被重写或者，当中断产生时向它们写入相同值时，允许进行批写入，其中的中断是基于主定时器（TAB0）的 TAB0CCR0 寄存器的值与定时器计数器的值相匹配而产生。

（1）PWM 输出模式中的设置

[初始设置]

主定时器：TAB0CTL0.TAB0CE = 0（禁止操作）

从定时器：TAA5CTL0.TAA5CE = 0（禁止操作）

[主定时器的初始设置（TAB0）]

- TAB0CTL1.TAB0MD2 至 TAB0CTL1.TAB0MD0 = 100（PWM 输出模式的设置）
- TAB0OPT0.TAB0CCS3 至 TAB0OPT0.TAB0CCS0 = 0000（捕获/比较选择位的设置为“比较”。）
- TAB0CCR0，TAB0CCR1，TAB0CCR2 和 TAB0CCR3 寄存器被设置。

[从定时器的初始设置（TAA5）]

- TAA5CTL1.TAA5SYE = 1（定时器调谐操作的设置）
- TAA5CTL1.TAA5MD2 至 TAA5CTL1.TAA5MD0 = 101（自由运行定时器模式的设置）
- TAA5OPT0.TAA5CCS1 和 TAA5OPT0.TAA5CCS0 = 00（捕获/比较选择位的设置为“比较”。）
- TAA5CCR0 和 TAA5CCR1 寄存器被设置。

备注 主定时器和从定时器的初始设置可以按任何顺序执行。

[开始计数]

- <1> 设置主定时器的 TAB0CTL0.TAB0CE 位为 1。
- <2> 开始计数。
- <3> 操作期间改变寄存器的设置。
 - 比较寄存器可被重写（批写入）。

[结束条件]

- 设置主定时器的 TAB0CTL0.TAB0CE 位为 0。

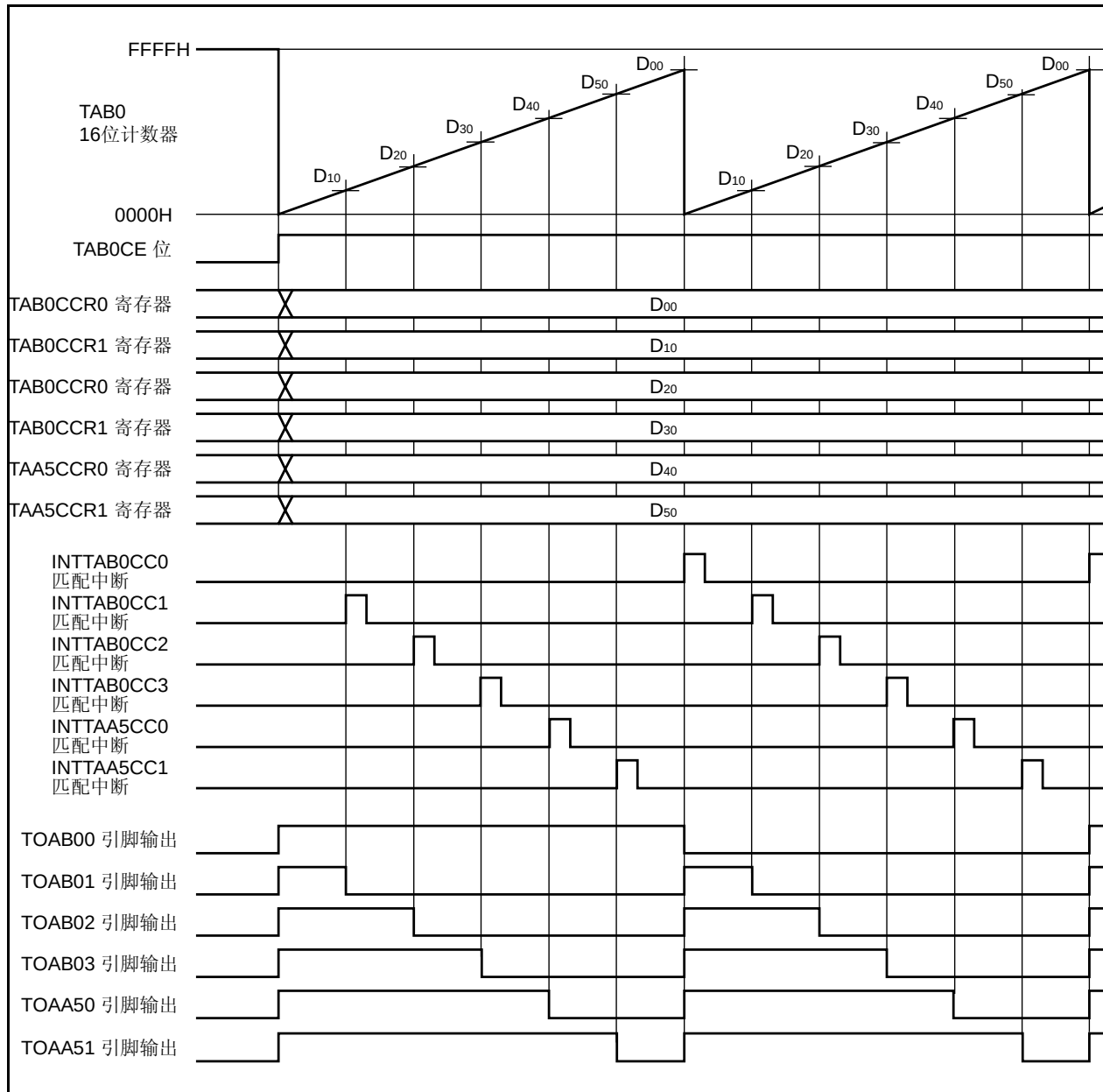
[批写入]

在 PWM 输出模式下，通过写入主定时器（TAB0）的 TAB0CCR1 寄存器使能下一个批写入。因此在所有必须重写的比较寄存器被重写后，必须重写主定时器（TAB0）的 TAB0CCR1 寄存器。

当定时器计数器的值和比较寄存器（TAB0CCR0）的周期值匹配时执行批写入。

如果不写入主定时器（TAB0）的 TAB0CCR1 寄存器，即使重写其它任何寄存器也不能进行批写入。从而，即使当定时器计数器的值与比较寄存器（TAB0CCR0）的周期值匹配时，比较寄存器的值也不能被重写。

图 7-47. 调谐 PWM 功能的时序举例（TAB0，TAA5）



7.7 同时启动功能

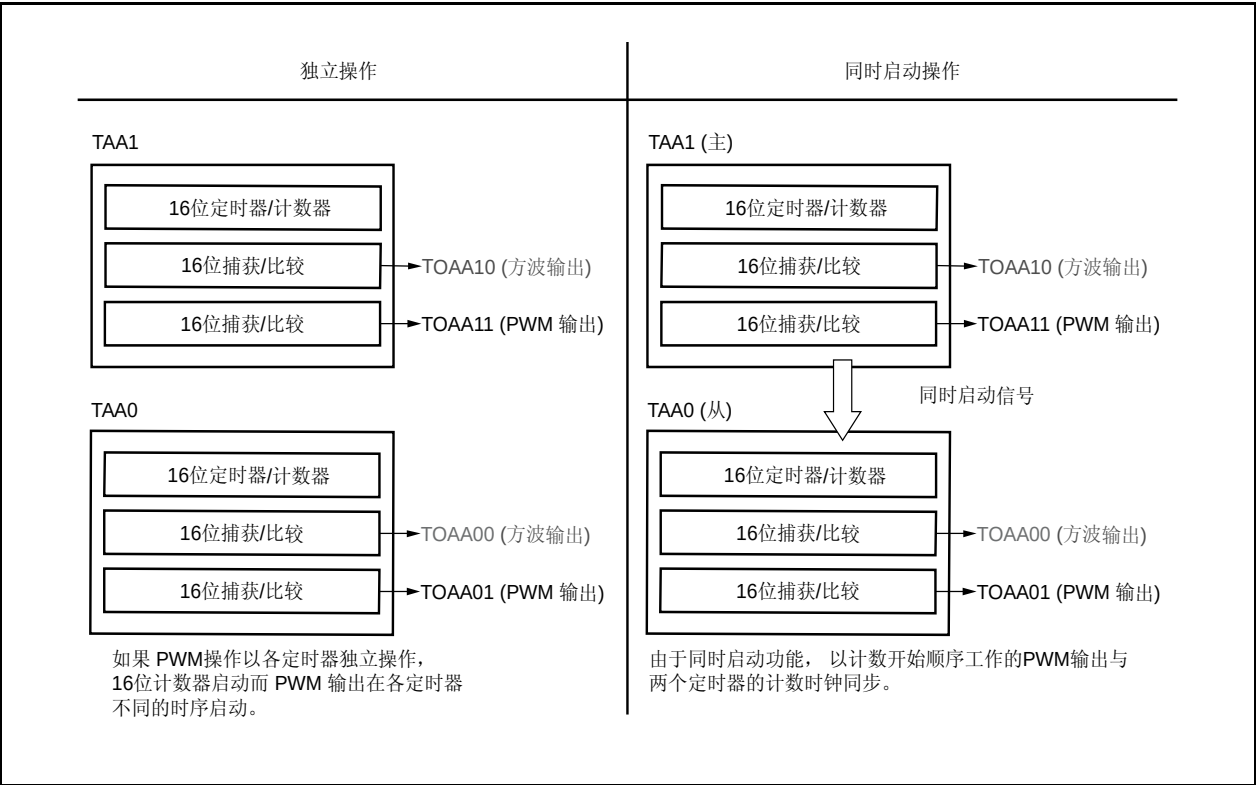
定时器 AA 和定时器 AB 有定时器调谐操作功能。
通过使用同时启动功能，可在主定时器和从定时器的操作启动时序和向上计数时序同步的基础上执行定时器操作。
只有 PWM 输出模式可在同时启动功能中使用。
可使用同时启动功能的定时器的组合列于表 7-10。

表 7-10. 定时器同时启动功能

主定时器	从定时器
TAA1	TAA0
TAA3	TAA2
TAB0	TAA5

图 7-48 是在脉宽调制信号输出模式下，TAA0（作为主定时器）和 TAA1（作为从定时器）的独立操作和同时启动操作的举例。

图 7-48. 使用 TAA1 和 TAA0 进行独立操作与同时启动操作的区别



7.7.1 PWM输出模式（同时启动操作）

本节中，展示了同时启动功能的操作，其中 TAA1 用作主定时器而 TAA0 用作从定时器操作。

当主定时器的 TAA1CTL0.TAA0CE 位设置为 1 时，主定时器（TAA1）和从定时器（TAA0）同时开始操作。通过主定时器（TAA1）提供计数时钟从定时器进行操作。但是，从定时器开始操作后，因为各定时器单独操作，所以即使基于主定时器（TAA1）的 16 位计数器的值与 TAA1CCR0 寄存器的值匹配而使得主定时器（TAA1）的 16 位计数器清除为 0000H，从定时器（TAA0）的 16 位计数器也不清零。

同样，如果通过批写入对主定时器（TAA1）的比较寄存器的值进行重写，则从定时器的比较寄存器的值不受影响。

[初始设置]

主定时器：TAA1CTL0.TAA1CE = 0（禁止操作）

从定时器：TAA0CTL0.TAA0CE = 0（禁止操作）

[主定时器的初始设置（TAA1）]

- TAA1CTL1.TAA1MD2 至 TAA1CTL1.TAA1MD0 = 100（PWM 输出模式的设置）
- TAA1CTL1.TAA1CKS2 至 TAA1CTL1.TAA1CKS0（计数时钟的设置（任意））
- TAA1CCR1, TAA1CCR0（捕获触发有效沿的指定）
- TAA1IOC0（捕获触发有效沿的指定）

[从定时器的初始设置（TAA0）]

- TAA0CTL1.TAA0SYE = 1, TAA0SYM = 1（同时启动操作）
- TAA0CTL1.TAA0MD2 至 TAA0CTL1.TAA0MD0 = 100（PWM 输出模式的设置）
- TAA0CCR0, TAA1CCR1（捕获触发有效沿的指定）
- TAA0IOC0（捕获触发有效沿的指定）

备注 主定时器和从定时器的初始设置可以按任何顺序执行。

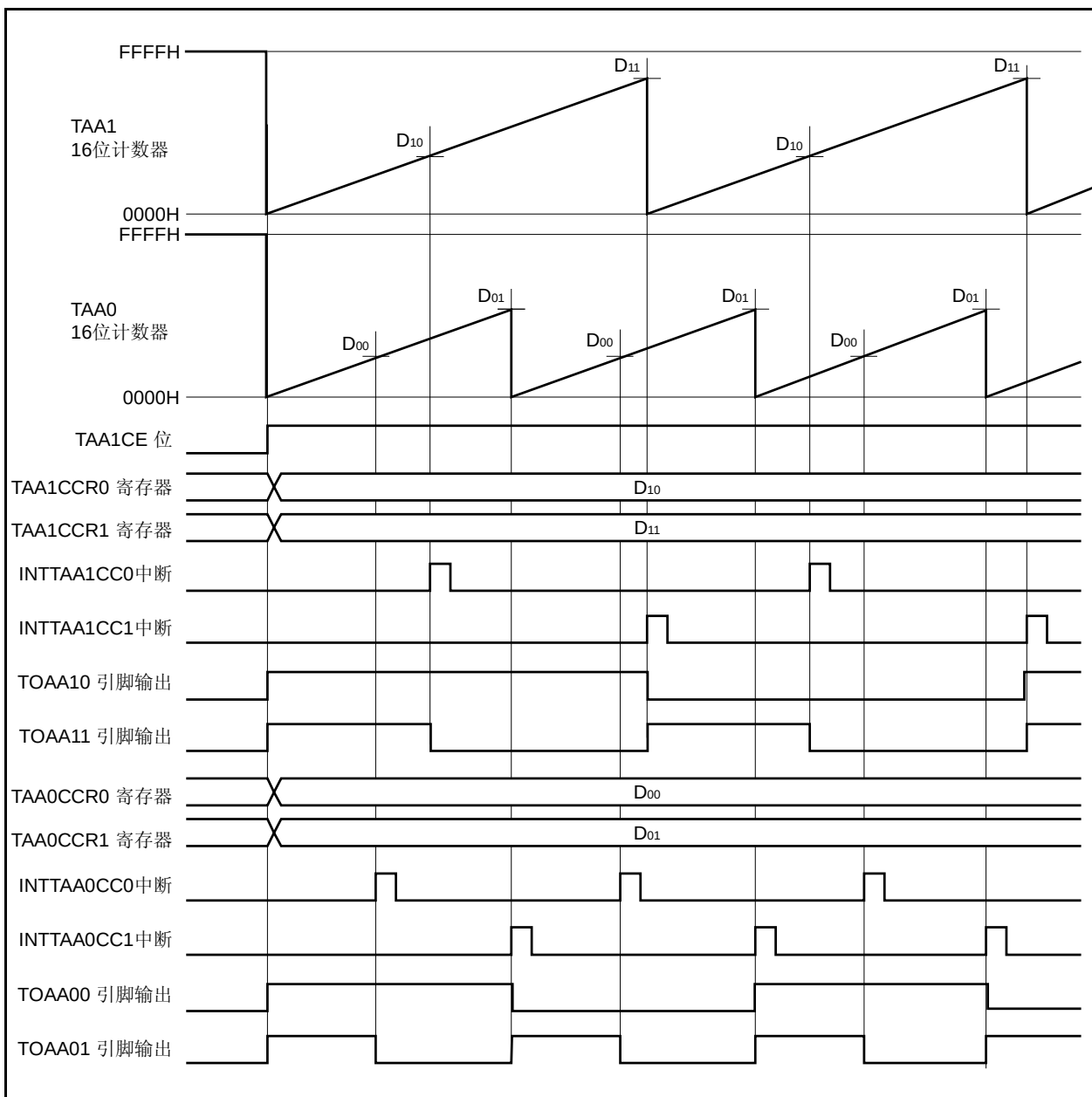
[开始计数]

- <1> 设置主定时器的 TAA1CTL0.TAA1CE 位为 1。
- <2> 开始计数。
- <3> 操作期间改变寄存器的设置。
 - 比较寄存器可被重写（随时写入）。

[结束条件]

- 设置主定时器的 TAA1CTL0.TAA0CE 位为 0。

图 7-49. 同时启动功能的时序举例 (TAA1: 主定时器, TAA0: 从定时器)



<R> 7.8 级联连接

本节阐述以级联的方式连接 TAA 的两个通道以形成一个 32 位捕获定时器的操作。

对于级联连接，必须设置自由运行定时器模式并且所有的捕获/比较寄存器都必须设置为捕获寄存器（TAA0CCSn = 1）。

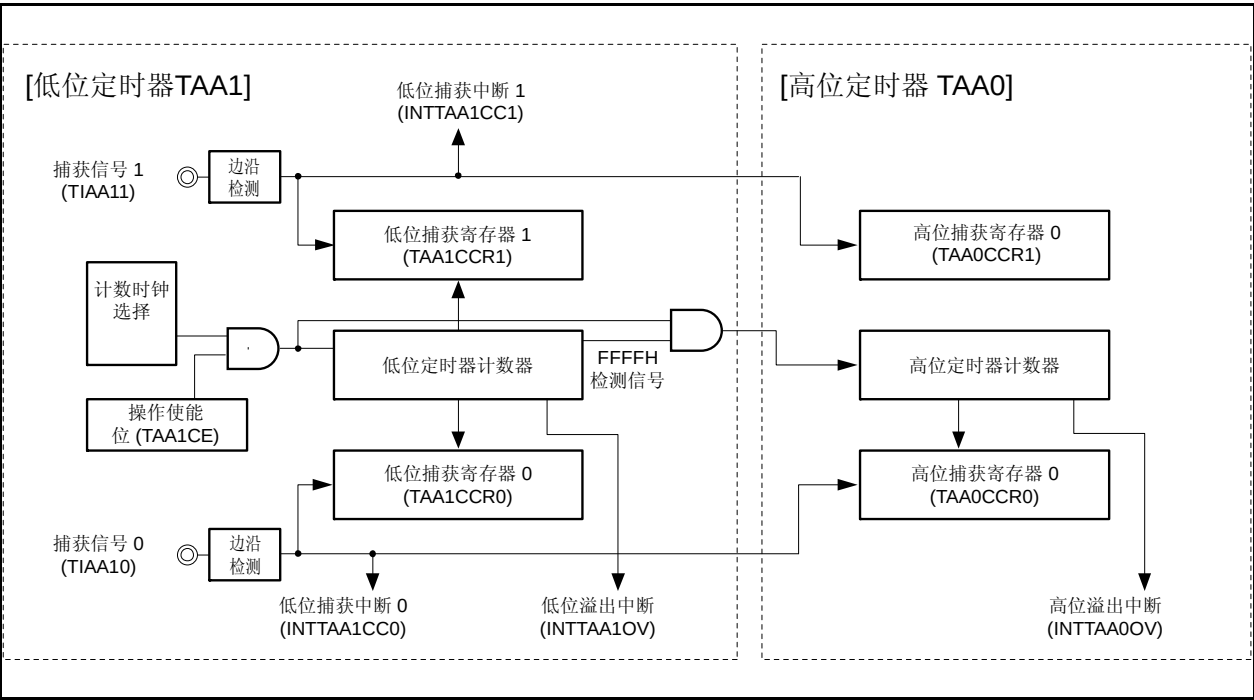
TAA 通道的结合可以按级联的方式连接如下表所示。

表 7-11. TAA 的级联连接

低位定时器（主定时器）	高位定时器（从定时器）
TAA1	TAA0
TAA3	TAA2

以下举例中，TAA1 用作低位定时器（主定时器）而 TAA0 用作高位定时器（从定时器）并且通过级联连接将它们用作 32 位捕获定时器。

图 7-50. 级联连接举例



当 TAA1 和 TAA0 级联时，各引脚和信号的操作如下所示：

表 7-12. 级联连接中的状态

名称	高位/低位	功能	操作
TIAA10 引脚输入	低位	捕获输入 0	当检测到输入的有效沿时，低位定时器计数器的值被存入 TAA1CCR0 寄存器，而高位定时器计数器的值存入 TAA0CCR0。
TIAA11 引脚输入	低位	捕获输入 1	当检测到输入的有效沿时，低位定时器计数器的值被存入 TAA1CCR1 寄存器，而高位定时器计数器的值存入 TAA0CCR1。
INTTAA1CCR0 中断信号	低位	捕获中断 0	当检测到 TIAA10 引脚的有效边沿时，产生中断信号。
INTTAA1CCR1 中断信号	低位	捕获中断 1	当检测到 TIAA11 引脚的有效边沿时，产生中断信号。
INTTAA1OV 中断信号	低位	溢出中断	当检测到低位定时器计数器的溢出时产生中断。
TIAA00 引脚输入	高位	捕获输入 0	没有进行操作。
TIAA01 引脚输入	高位	捕获输入 1	没有进行操作。
INTTAA0CCR0 中断信号	高位	捕获中断 0	没有进行操作。
INTTAA0CCR1 中断信号	高位	捕获中断 1	没有进行操作。
INTTAA0OV 中断信号	高位	溢出中断	当检测到高位定时器计数器的溢出时产生中断。

图 7-51. TAA1 和 TAA0 的级联连接的操作流程 (1/2)

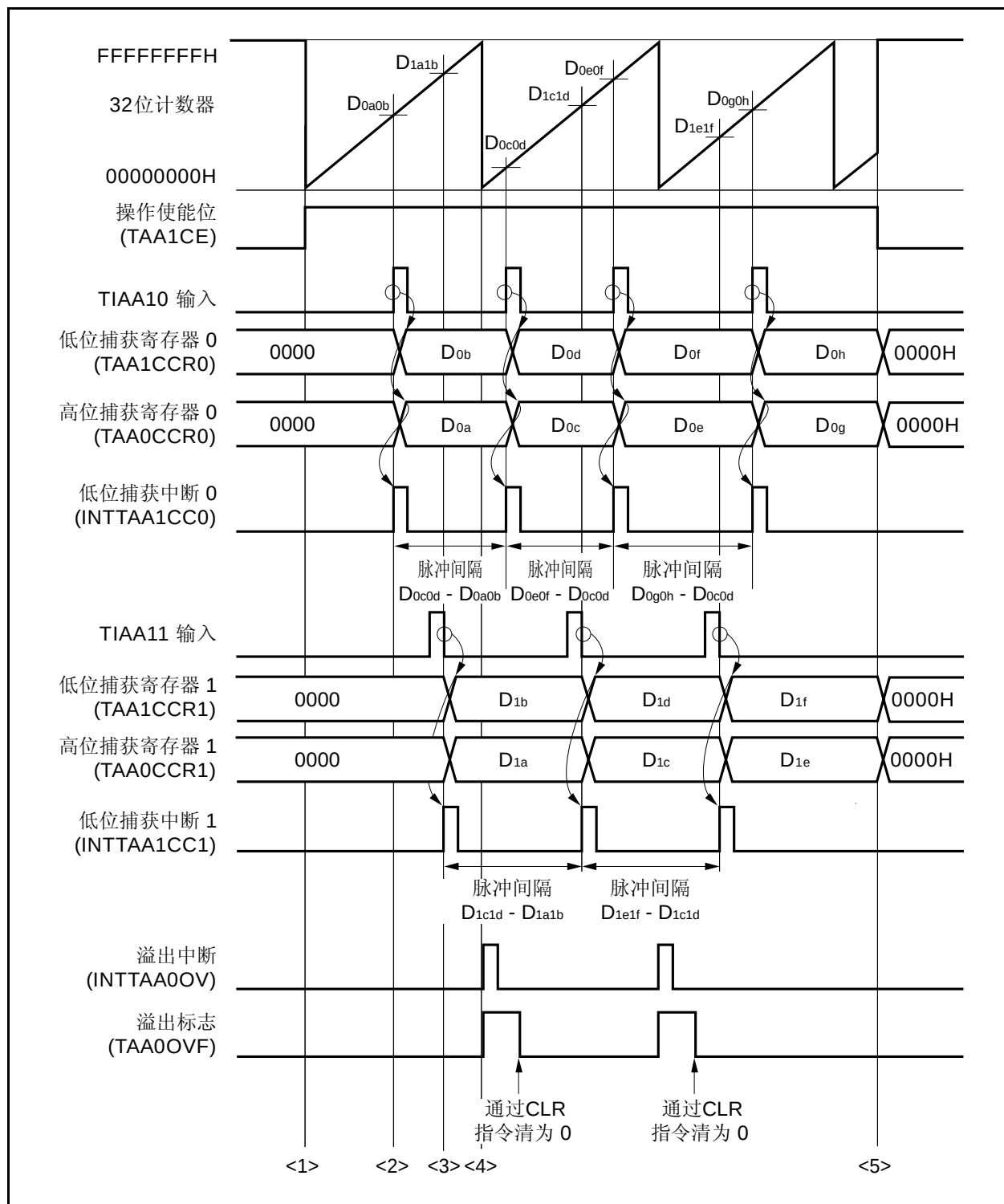


图 7-51. TAA1 和 TAA0 的级联连接的操作流程 (2/2)

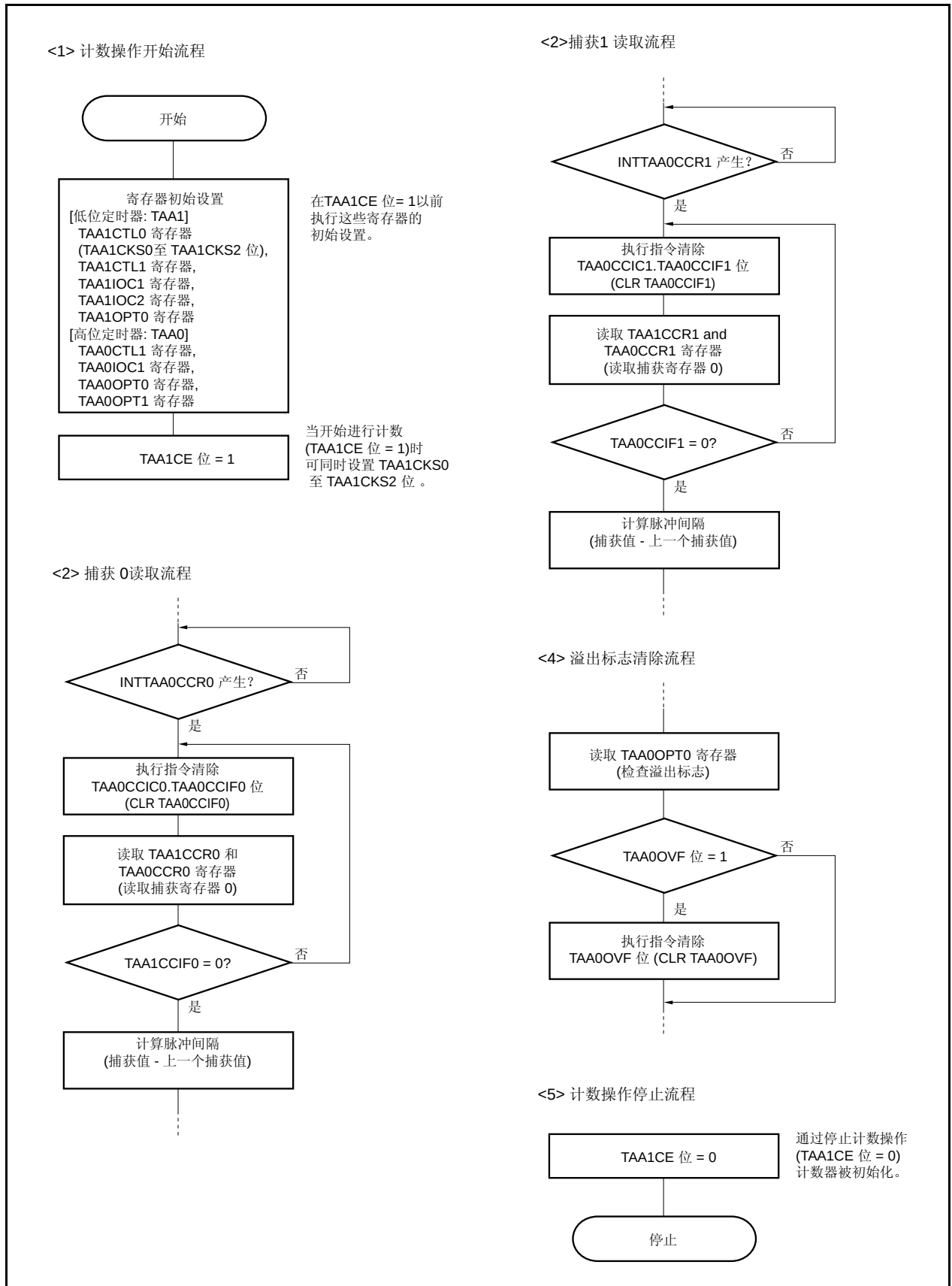
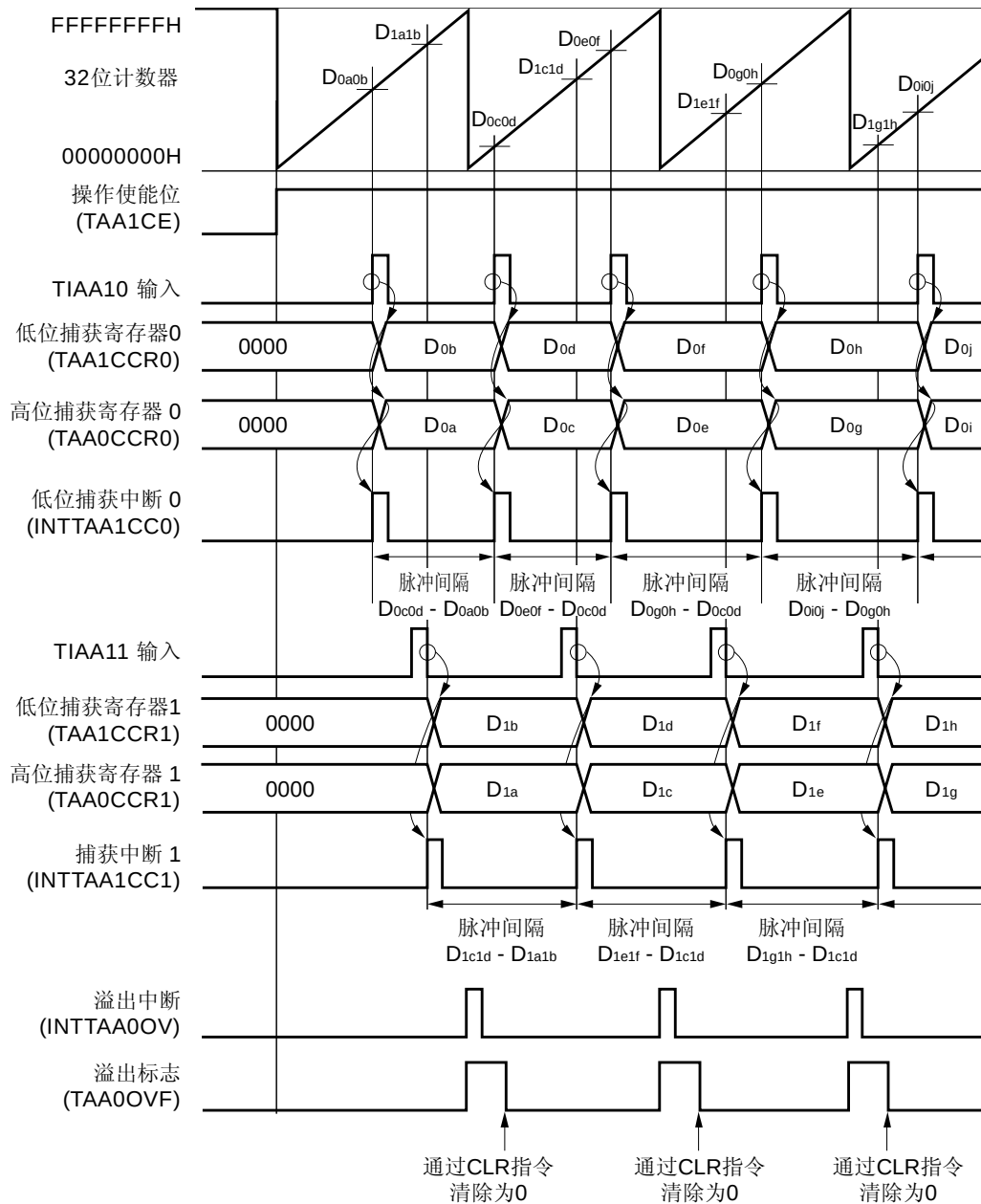


图 7-52. 当 TAA1 和 TAA0 以级联方式连接时基本时序举例



当 TAA1CTL.TAA1CE 位设置为 1 且提供计数时钟后就开始计数操作。当检测到输入 TIAA10 引脚的有效沿时，计数值存入捕获寄存器 0 (TAA1CCR0 和 TAA0CCR0)，并发出捕获中断信号 (INTTAA1CC0)。

定时器计数器继续与计数时钟同步进行计数。当向上计数至 FFFFFFFFH 时，在下一个时钟产生溢出中断 (INTTAA0OV) 并将溢出标志 (TAA0OVF) 设置为 1。定时器计数器被清除为 00000000H 并继续向上计数。

通过 CPU 发出向其写入“0”的指令溢出标志 (TAA0OVF) 被清零。

因为设置为自由运行定时器模式，所以定时器计数器不被检测到输入 TIAA10 引脚的有效沿清零。

因为 TOAA10 的复用功能为 TIAA10 输入，所以禁止使用 TOAA10 输出。

捕获寄存器 1 (TAA1CCR1 和 TAA0CCR1) 可在相同方式下操作。

如果低位定时器计数器 (TAA1) 溢出，则产生溢出中断 (TAA1OVF)。但是，因为其不能用作 32 位计数器的溢出中断，所以建议将其屏蔽。

7.9 选择器功能

V850ES/JG3-H 和 V850ES/JH3-H 中，端口和外设 I/O (TAA1, TAB0, UARTC0 或 UARTC1) 信号的复用功能引脚可选作 TAA1 和 TAB0 的捕获触发输入。

当使用 RXDCn 时，如果通过选择器功能选择从 UARTCn 引脚输入的信号，则 UARTCn 的 LIN 接收传输率的波特率误差可以计算 ($n = 0, 1$)。

(1) 选择器工作控制寄存器 0 (SELCNT0)

SELCNT0 寄存器为 8 位寄存器，用于选择定时器 CAN0, TAA1 和 TAB0 的捕获触发。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: FFFFFFF308H

	7	6	5	4	3	2	1	0
SELCNT0	0	0	0	ISEL4	ISEL3	0	0	ISEL0 ^注

ISEL4	TIAA11 捕获触发输入信号的选择
0	TIAA11 (复用功能为 P35) 引脚
1	RXDC1 (复用功能为 P91) 引脚

ISEL3	TIAA10 捕获触发输入信号的选择
0	TIAA10 (复用功能为 P34) 引脚
1	RXDC0 (复用功能为 P31) 引脚

ISEL0 ^注	TIAB02 捕获触发输入信号的选择
0	TIAB02 (复用功能为 P51) 引脚
1	CAN0 TSOUT 信号

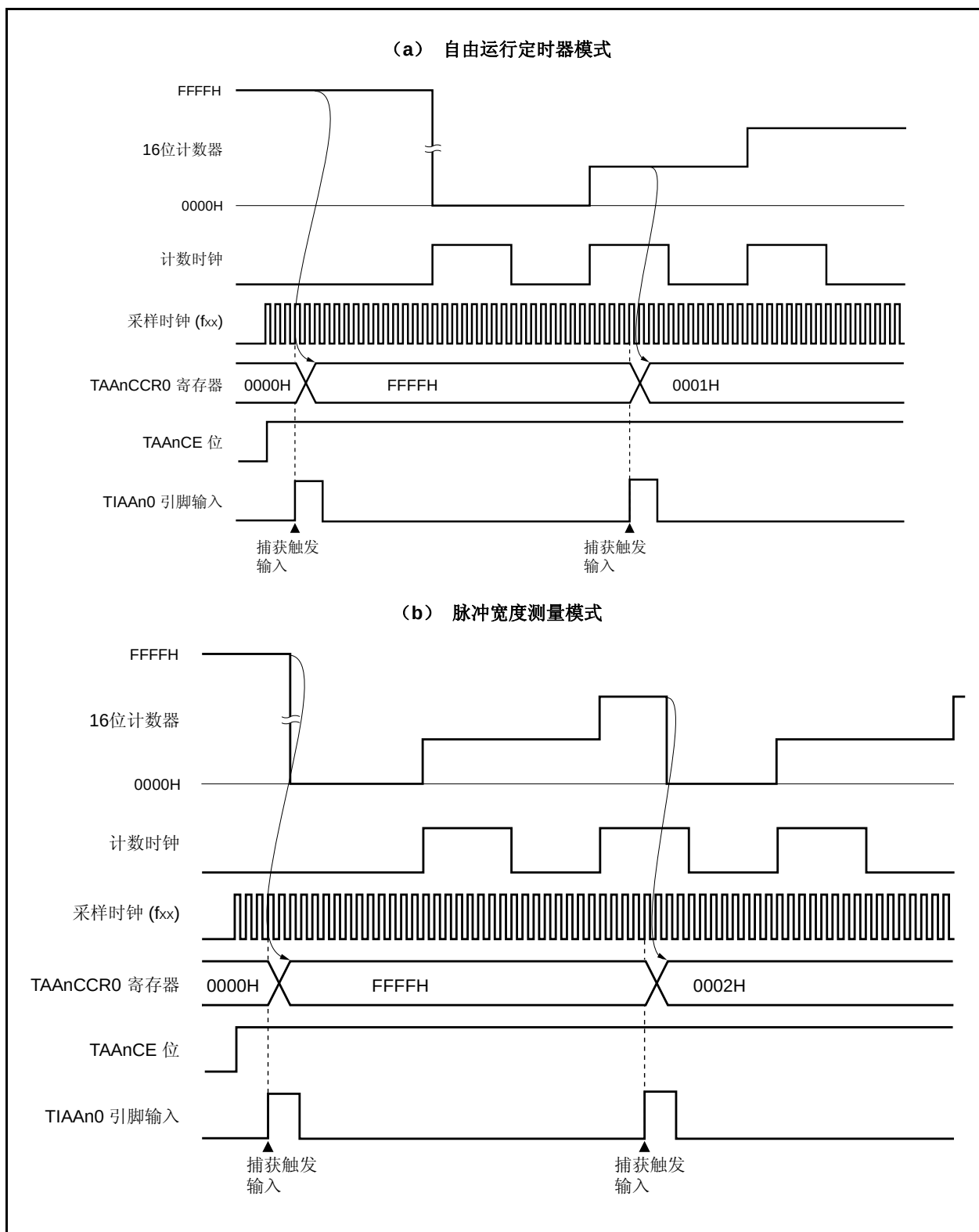
注 仅限于 μ PD70F3770 和 70F3771

- 注意事项
1. 要设置 ISEL4, ISEL3 和 ISEL0 位为 1，将相应引脚设置为捕获输入模式。
 2. 当 TAA1, TAB0 的操作以及 UARTC0, UARTC1 和 CAN0 停止时，设置 ISEL4, ISEL3 和 ISEL0。
 3. 一定要将位 7 至 5, 2 和 1 设置为“0”。

7.10 注意事项

(1) 捕获操作

当使用捕获操作并且将低速时钟选择为计数时钟时，如果在 TAAAnCE 位置 1 之后立即输入捕获触发，则被捕获到 TAAAnCCR0 和 TAAAnCCR1 寄存器中的是 FFFFH，而不是 0000H。



第八章 16 位定时器/事件计数器 AB (TAB)

定时器 AB (TAB) 是一个 16 位定时器/事件计数器。

V850ES/JG3-H 和 V850ES/JH3-H 有 TAB0 和 TAB1 定时器。

8.1 概述

定时器 TABn 概况如下所示：

- 时钟选择：8 种
- 捕获/触发输入引脚：4
- 外部事件计数输入引脚：1
- 外部触发输入引脚：1
- 定时器计数器：1
- 捕获/比较寄存器：4
- 捕获/比较匹配中断请求信号：4
- 定时器输出引脚：4

备注 n = 0, 1

8.2 功能

定时器 TABn 具有下列功能：

- 间隔定时器
- 外部事件计数器
- 外部触发脉冲输出
- 单次脉冲输出
- PWM (脉宽调制信号) 输出
- 自由运行定时器
- 脉冲宽度测量
- 三角波 PWM 输出
- 定时器调谐操作功能
- 同时启动功能

备注 n = 0, 1

8.3 配置

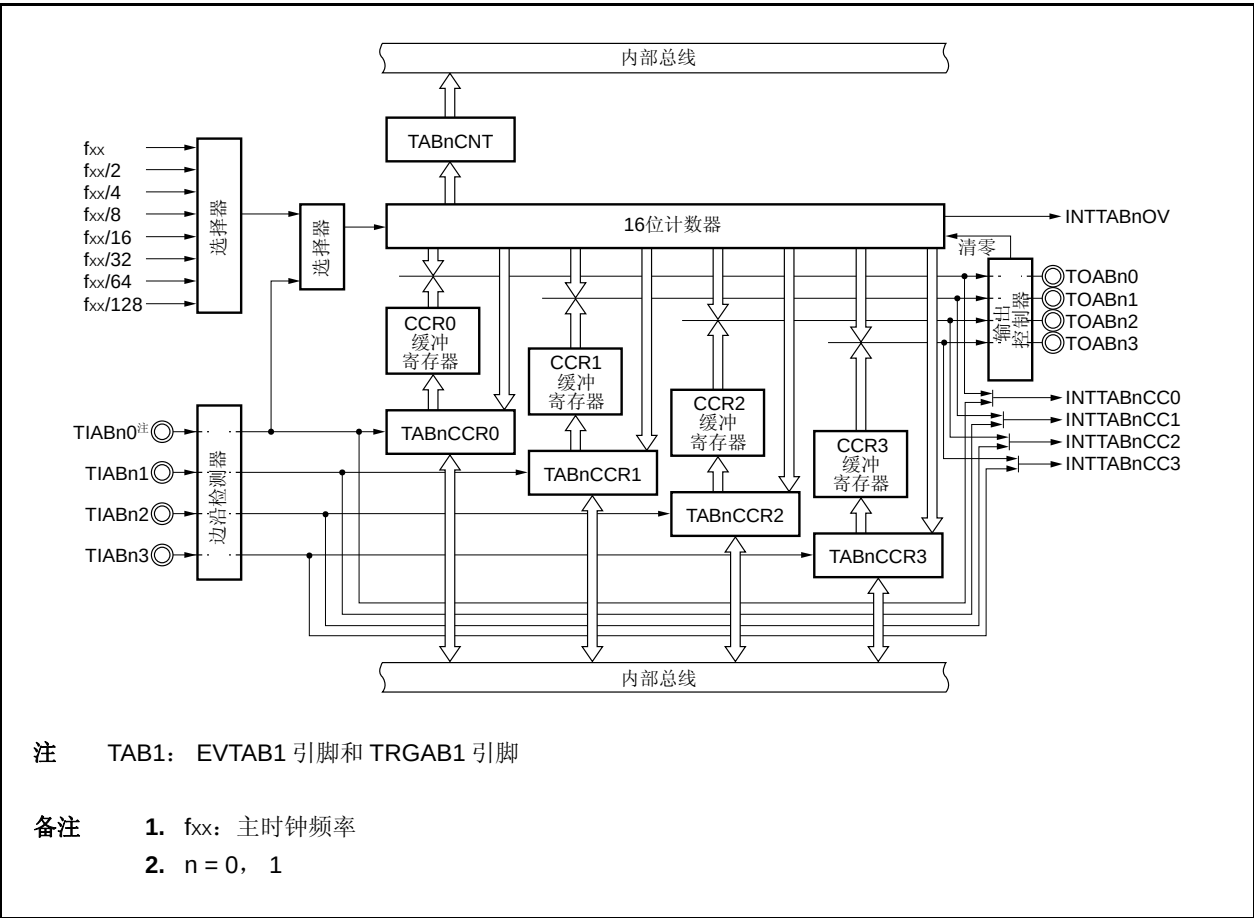
TABn 包括以下硬件：

表 8-1. TABn 的配置

项目	配置
寄存器	16 位计数器 TABn 捕获/比较寄存器 0 至 3 (TABnCCR0 至 TABnCCR3) TABn 计数器读取缓冲寄存器 (TABnCNT) CCR0 至 CCR3 缓冲寄存器 TABn 控制寄存器 0, 1 (TABnCTL0, TABnCTL1) TABn I/O 控制寄存器 0 至 2 (TABnIOC0 至 TABnIOC2, TABnIOC4) TABn 选项寄存器 0 (TABnOPT0)
定时器输入 ^{※2}	4 (TIABn0 ^{※1} 至 TIABn3 引脚)
定时器输出 ^{※2}	4 (TOABn0 至 TOABn3 引脚)

- 注
1. 当使用 TIABn0 至 TIABn3 和 TOABn0 至 TOABn3 引脚的功能时，参见表 4-20 端口引脚用作复用功能引脚。
 2. TIAB00 引脚和捕获触发输入引脚、外部事件计数输入引脚以及外部触发输入引脚共用。

图 8-1. TABn 的功能框图



(1) 16 位计数器

此 16 位计数器可以给内部时钟或外部事件计数。

该计数器的计数值可以通过读取 TABnCNT 寄存器得到。

当 TABnCTL0.TABnCE 位= 0 时, 16 位计数器的值为 FFFFH。如果此时读取寄存器 TABnCNT, 则读出值为 0000H。

系统复位将 TABnCE 位清 0, 因此, 16 位计数器被设置为 FFFFH。

(2) CCR0 缓冲寄存器

CCR1 缓冲寄存器为 16 位比较寄存器, 用于比较 16 位计数器的计数值。

当 TABnCCR0 寄存器用作比较寄存器时, 写入 TABnCCR0 寄存器的值会传送到 CCR0 缓冲寄存器。当 16 位计数器的计数值与 CCR0 缓冲寄存器的值相符时, 就会生成一个比较匹配中断请求信号 (INTTABnCC0)。

CCR0 缓冲寄存器不能直接读写。

CCR0 缓冲寄存器在复位后清零为 0000H, TABnCCR0 寄存器清零为 0000H。

(3) CCR1 缓冲寄存器

该缓冲寄存器为 16 位比较寄存器, 用于比较 16 位计数器的计数值。

当 TABnCCR1 寄存器用作比较寄存器时, 写入 TABnCCR1 寄存器的值会传送到 CCR1 缓冲寄存器。当 16 位计数器的计数值和 CCR1 缓冲寄存器的值匹配时, 也会产生一个比较匹配中断请求信号 (INTTABnCC1)。

CCR1 缓冲寄存器不能直接读写。

CCR1 缓冲寄存器在复位后清零为 0000H, TABnCCR1 寄存器清零为 0000H。

(4) CCR2 缓冲寄存器

该缓冲寄存器为 16 位比较寄存器, 用于比较 16 位计数器的计数值。

当 TABnCCR2 寄存器用作比较寄存器时, 写入 TABnCCR2 寄存器的值会传送到 CCR2 缓冲寄存器。当 16 位计数器的计数值和 CCR2 缓冲寄存器的值匹配时, 也会产生一个比较匹配中断请求信号 (INTTABnCC2)。

CCR2 缓冲寄存器不能直接读写。

CCR2 缓冲寄存器在复位后清零为 0000H, TABnCCR2 寄存器清零为 0000H。

(5) CCR3 缓冲寄存器

该缓冲寄存器为 16 位比较寄存器, 用于比较 16 位计数器的计数值。

当 TABnCCR3 寄存器用作比较寄存器时, 写入 TABnCCR3 寄存器的值会传送到 CCR3 缓冲寄存器。当 16 位计数器的计数值与 CCR3 缓冲寄存器的值相符时, 就会生成一个比较匹配中断请求信号 (INTTABnCC3)。

CCR3 缓冲寄存器不能直接读写。

CCR3 缓冲寄存器在复位后清零为 0000H, TABnCCR3 寄存器清零为 0000H。

(6) 边沿检测器

该电路检测输入到 TIABn0 至 TIABn3 引脚信号的有效边沿。使用 TABnIOC1 和 TABnIOC2 寄存器设置, 可以选择各种有效沿: 无边沿、上升沿、下降沿、上升/下降双边沿。

(7) 输出控制器

该电路控制 TOABn0 至 TOABn3 引脚的输出。输出控制器由 TABnIOC0 寄存器控制。

(8) 选择器

此选择器选择 16 位计数器的计数时钟。可选择 8 种内部时钟或一个外部事件为计数时钟。

8.4 寄存器

以下寄存器控制 TABn:

- TABn 控制寄存器 0 (TABnCTL0)
- TABn 控制寄存器 1 (TABnCTL1)
- TABn I/O 控制寄存器 0 (TABnIOC0)
- TABn I/O 控制寄存器 1 (TABnIOC1)
- TABn I/O 控制寄存器 2 (TABnIOC2)
- TABn I/O 控制寄存器 4 (TABnIOC4)
- TABn 选项寄存器 0 (TABnOPT0)
- TABn 捕获/比较寄存器 0 (TABnCCR0)
- TABn 捕获/比较寄存器 1 (TABnCCR1)
- TABn 捕获/比较寄存器 2 (TABnCCR2)
- TABn 捕获/比较寄存器 3 (TABnCCR3)
- TABn 计数器读取缓冲寄存器 (TABnCNT)

- 备注**
1. 当使用 TIABn0 至 TIABn3 以及 TOABn0 至 TOABn3 引脚的功能时, 参见表 4-20 端口引脚用作复用功能引脚。
 2. n = 0, 1

(1) TABn 控制寄存器 0 (TABnCTL0)

TABnCTL0 寄存器为 8 位寄存器，用于控制定时器 TABn 的操作。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器设置为 00H。

可使用软件总向 TABnCTL0 寄存器写入相同值。

复位后: 00H R/W 地址: TAB0CTL0 FFFFF540H, TAB1CTL0 FFFFF560H

	7	6	5	4	3	2	1	0
TABnCTL0 (n = 0, 1)	TABnCE	0	0	0	0	TABnCKS2	TABnCKS1	TABnCKS0

TABnCE	TABn 操作控制
0	禁止TABn 操作 (TABn 异步复位 ^注)。
1	允许TABn 操作。启动TABn 操作。

TABnCKS2	TABnCKS1	TABnCKS0	内部计数时钟设置
0	0	0	f _{xx}
0	0	1	f _{xx} /2
0	1	0	f _{xx} /4
0	1	1	f _{xx} /8
1	0	0	f _{xx} /16
1	0	1	f _{xx} /32
1	1	0	f _{xx} /64
1	1	1	f _{xx} /128

注 TABnOPT0.TABnOVF 位，16 位计数器，定时器输出 (TOABn0 至 TOABn3 引脚)

- 注意事项
1. 当 TABnCE 位 = 0 时设置 TABnCKS2 至 TABnCKS0 位。当 TABnCE 位的值从 0 改变到 1 时，可同时设置 TABnCKS2 至 TABnCKS0 位。
 2. 务必将 3 至 6 位设置为“0”。

备注 f_{xx}: 主时钟频率

TAPnCTL1 客存器为 8 位客存器 用于控制字时器 TAPn 的操作

本案左翼可按 8 位或 1 位读写

复位后本寄存器的被设置为 00H

TARGET	物件歸半控制

进行计数。

1	1	1	三角波PWM 模式

2. 一定要将位 3、4 以及 7 设置为“0”。

3. 不论 TABnEEE 位的取值如何, 在外部事件计数模式下, 都选择外部事件计数输

4. 当 TABnCTL0.TABnCE 位 = 0 时, 设置 TABnEEE 以及 TABnMD2 至

(3) TABn I/O 控制寄存器 0 (TABnIOC0)

TABnIOC0 寄存器是 8 位寄存器，用于控制定时器输出 (TOABn0 至 TOABn3 引脚)。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器设置为 00H。

复位后: 00H R/W 地址: TAB0IOC0 FFFFF542H, TAB1IOC0 FFFFF562H

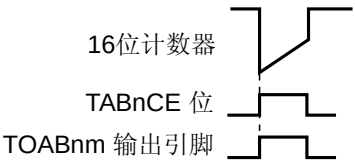
	7	6	5	4	3	2	1	0
TABnIOC0 (n = 0, 1)	TABnOL3	TABnOE3	TABnOL2	TABnOE2	TABnOL1	TABnOE1	TABnOL0	TABnOE0

TABnOLm	TOABnm 引脚输出电平设置 (m = 0 至 3) ^注
0	TOABnm 引脚从高电平启动
1	TOABnm 引脚从低电平启动

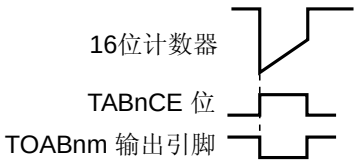
TABnOEm	TOABnm 引脚输出电平设置 (m = 0 至 3)
0	禁止定时器输出 · 当 TABnOLm 位 = 0: 从 TOABnm 引脚输出低电平 · 当 TABnOLm 位 = 1: 从 TOABnm 引脚输出高电平
1	允许定时器输出 (从 TOABnm 引脚输出方波)。

注 由 TABnOLm 位指定的定时器输出引脚 (TOABnm) 的输出电平如下所示:

· 当 TABnOLm 位 = 0



· 当 TABnOLm 位 = 1



- 注意事项**
1. 当 TABnCTL0.TABnCE 位 = 0 时，重置 TABnOLm 和 TABnOEm 位。(当 TABnCE 位 = 1 时，可设置相同的值。)如果重新写入不能正确执行，将 TABnCE 位清为 0，然后重新进行位设定。
 2. 当 TABnCE 和 TABnOEm 位为 0 时，即使 TABnOLm 位被操作，TOABnm 引脚的输出电平也会发生变化。

备注 m = 0 至 3

(4) TABn I/O 控制寄存器 1 (TABnIOC1)

TABnIOC1 寄存器为 8 位寄存器，用于控制捕获触发输入信号 (TIABn0 至 TIABn3 引脚) 的有效沿沿。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: TAB0IOC1 FFFFF543H, TAB1IOC1 FFFFF563H

	7	6	5	4	3	2	1	0
TABnIOC1 (n = 0, 1)	TABnIS7	TABnIS6	TABnIS5	TABnIS4	TABnIS3	TABnIS2	TABnIS1	TABnIS0

TABnIS7	TABnIS6	捕获触发输入信号 (TIABn3 引脚) 有效沿设置
0	0	无边沿检测 (捕获操作无效)
0	1	上升沿检测
1	0	下降沿检测
1	1	双边沿检测

TABnIS5	TABnIS4	捕获触发输入信号 (TIABn2 引脚) 有效沿设置
0	0	无边沿检测 (捕获操作无效)
0	1	上升沿检测
1	0	下降沿检测
1	1	双边沿检测

TABnIS3	TABnIS2	捕获触发输入信号 (TIABn1 引脚) 有效沿设置
0	0	无边沿检测 (捕获操作无效)
0	1	上升沿检测
1	0	下降沿检测
1	1	双边沿检测

TABnIS1	TABnIS0	捕获触发输入信号 (TIABn0 引脚) 有效沿设置
0	0	无边沿检测 (捕获操作无效)
0	1	上升沿检测
1	0	下降沿检测
1	1	双边沿检测

- 注意事项**
1. 当 TABnCTL0.TABnCE 位 = 0 时，重写 TABnIS7 至 TABnIS0 位。（当 TABnCE 位 = 1 时，可写入相同的值。）如果重新写入不能正确执行，将 TABnCE 位清为 0，然后重新进行位设定。
 2. 只有在自由运行定时器模式和脉冲宽度测量模式下，TABnIS7 至 TABnIS0 位才有效。在所有其它模式下，无法进行捕获操作。

(5) TABn I/O 控制寄存器 2 (TABnIOC2)

TABnIOC2 寄存器为 8 位寄存器，用于控制外部事件计数输入信号 (TIAB00/EVTAB1 引脚) 和外部触发输入信号 (TIAB00/TRGAB1 引脚) 的有效边沿。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: TAB0IOC2 FFFFF544H, TAB1IOC2 FFFFF564H

	7	6	5	4	3	2	1	0
TABnIOC2 (n = 0, 1)	0	0	0	0	TABnEES1	TABnEES0	TABnETS1	TABnETS0

TABnEES1	TABnEES0	外部事件计数输入信号 (TIAB00/EVTAB1 引脚) 有效沿设置
0	0	无边沿检测 (外部事件计数无效)
0	1	上升沿检测
1	0	下降沿检测
1	1	双边沿检测

TABnETS1	TABnETS0	外部触发输入信号 (TIAB00/TRGAB1 引脚) 有效沿设置
0	0	无边沿检测 (外部触发无效)
0	1	上升沿检测
1	0	下降沿检测
1	1	双边沿检测

- 注意事项**
1. 当 TABnCTL0.TABnCE 位 = 0 时，重写 TABnEES1，TABnEES0，TABnETS1 和 TABnETS0 位。（当 TABnCE 位 = 1 时，可写入相同的值。）如果重新写入不能正确执行，将 TABnCE 位清为 0，然后重新进行位设定。
 2. 只有在 TABnCTL1.TABnEEE 位 = 1 或设置了外部事件计数模式 (TABnCTL1.TABnMD2 至 TABnCTL1.TABnMD0 位 = 001) 时，TABnEES1 和 TABnEES0 位才有效。
 3. 只有在外部触发脉冲输出模式 (TABnCTL1.TABnMD2 至 TABnCTL1.TABnMD0 位 = 010) 或处于单次脉冲输出模式 (TABnCTL1.TABnMD2 至 TABnCTL1.TABnMD0 = 011) 时，TABnETS1 和 TABnETS0 位才有效。

(6) TABn I/O 控制寄存器 4 (TABnIOC4)

TABnIOC4 寄存器为 8 位寄存器，用来控制定时器输出。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。该寄存器不通过停止定时器操作进行复位 (TABnCTL0.TABnCE = 0)。

注意事项 1. 在以下状态中，禁止访问 TABnIOC4 寄存器。有关详细说明，参见 3.4.9 (2) 访问片上专用外围 I/O 寄存器。

- 当 CPU 工作于副时钟，而且主时钟振荡停止时
- 当 CPU 工作于内部振荡时钟时

2. TABnIOC4 寄存器只能在间隔定时器模式和自由运行定时器模式下进行设置。在其它模式下确保将 TABnIOC4 寄存器设置为 00H (有关模式设置的详情，参见 8.4 (2) TABn 控制寄存器 1 (TABnCTL1))。即使在自由运行模式下，如果 TABnCCR0 至 TABnCCR3 寄存器设置为捕获功能，TABnIOC4 寄存器的设置也会变得无效。

复位后: 00H R/W 地址: TAB0IOC4 FFFFF550H, TAB1IOC4 FFFFF570H

	7	6	5	4	3	2	1	0
TABnIOC4 (n = 0, 1)	TABnOS3	TABnOR3	TABnOS2	TABnOR2	TABnOS1	TABnOR1	TABnOS0	TABnOR0

TABnOSm	TABnORm	TIABnm 引脚的切换控制 (m = 0 至 3)
0	0	无请求。正常切换操作。
0	1	复位请求 当16位计数器与TAA nCCRm寄存器的值下一次匹配时 固定为无效电平。
1	0	设置请求 当16位计数器与TAA nCCRm寄存器的值下一次匹配时 固定为有效电平。
1	1	保持请求 保持当前输出电平。

(7) TABn 选项寄存器 0 (TABnOPT0)

TABnOPT0 寄存器为 8 位寄存器，用来设置捕获/比较操作和检测计数器溢出。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器设置为 00H。

复位后: 00H R/W 地址: TAB0OPT0 FFFFF545H, TAB1OPT0 FFFFF565H

	7	6	5	4	3	2	1	0
TABnOPT0 (n = 0, 1)	TABnCCS3	TABnCCS2	TABnCCS1	TABnCCS0	0	TAB1CMS ^注	TABnCUF	TABnOVF

TABnCCSm	TABnCCRm寄存器捕获/比较的选择
0	选择比较寄存器
1	选择捕获寄存器
TABnCCSm位的设置仅在自由运行定时器模式下有效。	

TABnOVF	TABn 溢出检测
设置 (1)	发生溢出
复位 (0)	TABnOVF 位写入 0或 TABnCTL0.TABnCE 位 = 0
- 在自由运行定时器模式或脉冲宽度测量模式下，当16位计数器从FFFFH至0000H计数值溢出时，TABnOVF 位设置为1。 - 指定请求信号 (INTTABnOV) 在TABnOVF 位设置为1的同时产生。 - 除自由运行定时器模式和脉冲测量模式外其它模式下不产生INTTABnOV 信号。 - 当TABnOVF位 = 1时即使读取TABnOVF位或TABnOPT0 寄存器，TABnOVF 位也不会清零。 - TABnOVF 位可读可写，但 TABnOVF位不能由软件设置为1。写1不影响TABn的操作。	

注 TAB1CMS 位用于马达控制功能。详情参见第十一章 马达控制功能。

- 注意事项
1. 当 TABnCTL0.TABnCE 位 = 0 时，重写 TABnCCS3 至 TABnCCS0 位。（当 TABnCE 位 = 1 时，可写入相同的值。）如果重新写入不能正确执行，将 TABnCE 位清为 0，然后重新进行位设定。
 2. 务必将位 3 设置为“0”。不使用马达控制功能时，确保也将位 2 设置为“0”。

备注 m = 0 至 3

(8) TABn 捕获/比较寄存器 0 (TABnCCR0)

根据不同工作模式，TABnCCR0 寄存器可以用作捕获寄存器或比较寄存器。

仅在自由计数器模式下，依据 TABnOPT0.TABnCCS0 位的设定，该寄存器可用作捕获寄存器或比较寄存器。在脉宽测量模式中，TABnCCR0 寄存器只可以用作捕获寄存器。而在其它模式时，该寄存器只可以用作比较寄存器。

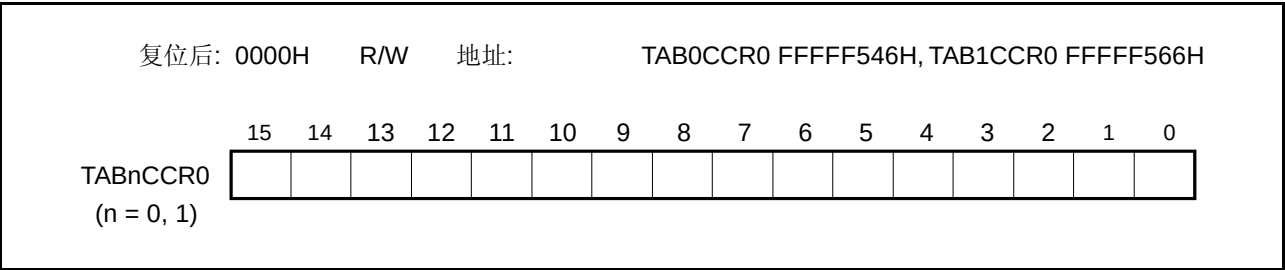
TABnCCR0 寄存器可以在操作中进行读取和写入。

该寄存器可按 16 位为单位进行读写。

复位后本寄存器设置为 0000H。

注意事项 在以下状态中，禁止访问 TABnCCR0 寄存器。有关详细说明，参见 3.4.9 (2) 访问片上专用外围 I/O 寄存器。

- 当 CPU 工作于副时钟，而且主时钟振荡停止时
- 当 CPU 工作于内部振荡时钟时



(a) 作为比较寄存器时的功能

即便在 TABnCTL0.TABnCE 位=1 时，TABnCCR0 寄存器仍可以重新写入。

TABnCCR0 寄存器的设定值传送给 CCR0 缓冲寄存器。当 16 位计数器的值与 CCR0 缓冲寄存器的值相匹配时，就会生成一个比较匹配中断请求信号（INTTABnCC0）。如果此时 TOABn0 引脚输出允许，则 TOABn0 引脚的输出会反相。

在间隔定时器模式、外部事件计数模式、外部触发脉冲输出模式、单次脉冲输出模式、脉宽调制信号输出模式或三角波 PWM 模式时，当 TABnCCR0 寄存器作为循环寄存器时，如果 16 位计数器的计数值与 CCR0 缓冲寄存器数值匹配，则计数器清零为（0000H）。

(b) 用作捕获寄存器

在自由运行定时器模式下，当 TABnCCR0 寄存器用作捕获寄存器时，若检测到捕获触发输入引脚（TIABn0 引脚）的有效边沿，则 16 位计数器的当前计数值存储于 TABnCCR0 寄存器中。在脉宽测量模式下，若检测到捕获触发输入引脚（TIABn0）的有效边沿，16 位计数器的当前计数值就储存在 TABnCCR0 寄存器中，且 16 位计数器清零（0000H）。

即使捕获操作与对 TABnCCR0 寄存器的读取发生冲突，还是可以正确读取 TABnCCR0 寄存器的值。

备注 n = 0, 1

下表显示了捕获/比较寄存器在各工作模式下的功能，以及怎样向比较寄存器写入数据。

表 8-2. 捕获/比较寄存器在各模式下的功能和如何写入比较寄存器

操作模式	捕获/比较寄存器	如何写入比较寄存器
间隔定时器	比较寄存器	可随时写入
外部事件计数器	比较寄存器	可随时写入
外部触发脉冲输出	比较寄存器	批写入
单次脉冲输出	比较寄存器	可随时写入
PWM（脉宽调制信号）输出	比较寄存器	批写入
自由运行定时器	捕获/比较寄存器	可随时写入
脉冲宽度测量	捕获寄存器	—
三角波 PWM 模式	比较寄存器	批写入

(9) TABn 捕获/比较寄存器 1 (TABnCCR1)

根据不同工作模式，TABnCCR1 寄存器可以用作捕获寄存器或比较寄存器。

仅在自由计时器模式下，依据 TABnOPT0.TABnCCS1 位的设定，该寄存器可用作捕获寄存器或比较寄存器。在脉宽测量模式中，TABnCCR1 寄存器只可以用作捕获寄存器。而在其它模式时，该寄存器只可以用作比较寄存器。

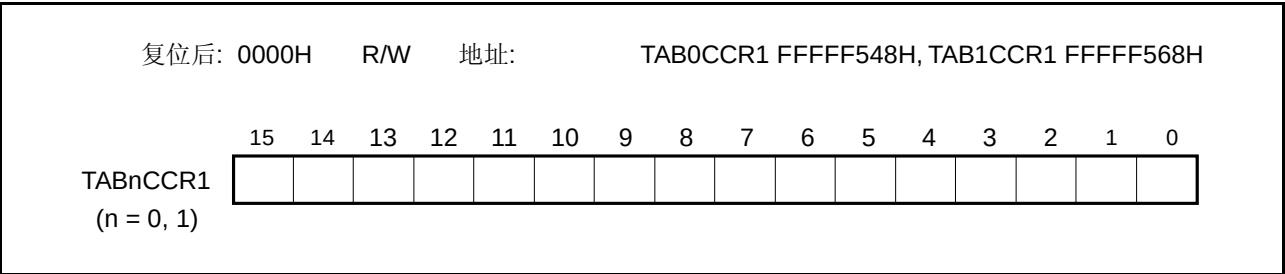
TABnCCR1 寄存器可以在操作中进行读取和写入。

该寄存器可按 16 位为单位进行读写。

复位后本寄存器设置为 0000H。

注意事项 在以下状态中，禁止访问 TABnCCR1 寄存器。有关详细说明，参见 3.4.9 (2) 访问片上专用外围 I/O 寄存器。

- 当 CPU 工作于副时钟，而且主时钟振荡停止时
- 当 CPU 工作于内部振荡时钟时



(a) 作为比较寄存器时的功能

即便在 TABnCTL0.TABnCE 位=1 时，TABnCCR1 寄存器仍可以重新写入。
TABnCCR1 寄存器的设定值传送给 CCR1 缓冲寄存器。当 16 位计数器的值与 CCR1 缓冲寄存器的值相匹配时，就会生成一个比较匹配中断请求信号（INTTABnCC1）。如果此时 TOABn1 引脚输出允许，则 TOABn1 引脚的输出会反相。

(b) 用作捕获寄存器

在自由运行定时器模式下，当 TABnCCR1 寄存器用作捕获寄存器时，若检测到捕获触发输入引脚（TIABn1 引脚）的有效边沿，则 16 位计数器的当前计数值存储于 TABnCCR1 寄存器中。在脉宽测量模式下，若检测到捕获触发输入引脚（TIABn1）的有效边沿，16 位计数器的当前计数值就储存在 TABnCCR1 寄存器中，且 16 位计数器清零（0000H）。
即使捕获操作与对 TABnCCR1 寄存器的读取发生冲突，还是可以正确读取 TABnCCR1 寄存器的值。

备注 n = 0, 1

下表显示了捕获/比较寄存器在各工作模式下的功能，以及怎样向比较寄存器写入数据。

表 8-3. 捕获/比较寄存器在各模式下的功能和如何写入比较寄存器

操作模式	捕获/比较寄存器	如何写入比较寄存器
间隔定时器	比较寄存器	可随时写入
外部事件计数器	比较寄存器	可随时写入
外部触发脉冲输出	比较寄存器	批写入
单次脉冲输出	比较寄存器	可随时写入
PWM（脉宽调制信号）输出	比较寄存器	批写入
自由运行定时器	捕获/比较寄存器	可随时写入
脉冲宽度测量	捕获寄存器	—
三角波 PWM 模式	比较寄存器	批写入

(10) TABn 捕获/比较寄存器 2 (TABnCCR2)

根据不同工作模式，TABnCCR2 寄存器可以用作捕获寄存器或比较寄存器。

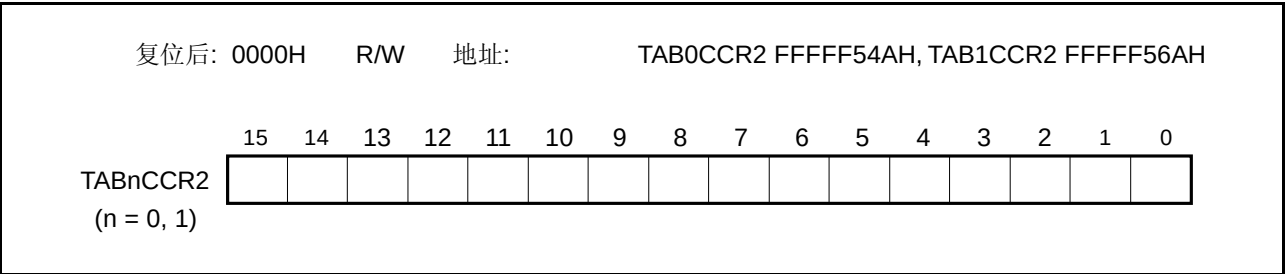
仅在自由计时器模式下，依据 TABnOPT0.TABnCCS2 位的设定，该寄存器可用作捕获寄存器或比较寄存器。在脉宽测量模式中，TABnCCR2 寄存器只可以用作捕获寄存器。而在其它模式时，该寄存器只可以用作比较寄存器。

TABnCCR2 寄存器可以在运行中进行读取和写入。

该寄存器可按 16 位为单位进行读写。

复位后本寄存器设置为 0000H。

- 注意事项** 在以下状态中，禁止访问 TABnCCR2 寄存器。有关详细说明，参见 3.4.9 (2) 访问片上专用外围 I/O 寄存器。
- 当 CPU 工作于副时钟，而且主时钟振荡停止时
 - 当 CPU 工作于内部振荡时钟时



(a) 作为比较寄存器时的功能

即便在 TABnCTL0.TABnCE 位=1 时，TABnCCR2 寄存器仍可以重新写入。
TABnCCR2 寄存器的设定值传送给 CCR2 缓冲寄存器。当 16 位计数器的值与 CCR2 缓冲寄存器的值相匹配时，就会生成一个比较匹配中断请求信号（INTTABnCC2）。如果此时 TOABn2 引脚输出允许，则 TOABn2 引脚的输出会反相。

(b) 用作捕获寄存器

在自由运行定时器模式下，当 TABnCCR2 寄存器用作捕获寄存器时，若检测到捕获触发输入引脚（TIABn2 引脚）的有效边沿，则 16 位计数器的当前计数值存储于 TABnCCR2 寄存器中。在脉宽测量模式下，若检测到捕获触发输入引脚（TIABn2）的有效边沿，16 位计数器的当前计数值就储存在 TABnCCR2 寄存器中，且 16 位计数器清零（0000H）。
即使捕获操作与对 TABnCCR2 寄存器的读取发生冲突，还是可以正确读取 TABnCCR2 寄存器的值。

备注 n = 0, 1

下表显示了捕获/比较寄存器在各工作模式下的功能，以及怎样向比较寄存器写入数据。

表 8-4. 捕获/比较寄存器在各模式下的功能和如何写入比较寄存器

操作模式	捕获/比较寄存器	如何写入比较寄存器
间隔定时器	比较寄存器	可随时写入
外部事件计数器	比较寄存器	可随时写入
外部触发脉冲输出	比较寄存器	批写入
单次脉冲输出	比较寄存器	可随时写入
PWM（脉宽调制信号）输出	比较寄存器	批写入
自由运行定时器	捕获/比较寄存器	可随时写入
脉冲宽度测量	捕获寄存器	—
三角波 PWM 模式	比较寄存器	批写入

(11) TABn 捕获/比较寄存器 3 (TABnCCR3)

根据不同工作模式，TABnCCR3 寄存器可以用作捕获寄存器或比较寄存器。
仅

在自由计时器模式下，依据 TABnOPT0.TABnCCS3 位的设定，该寄存器可用作捕获寄存器或比较寄存器。在脉宽测量模式中，TABnCCR3 寄存器只可以用作捕获寄存器。而在其它模式时，该寄存器只可以用作比较寄存器。

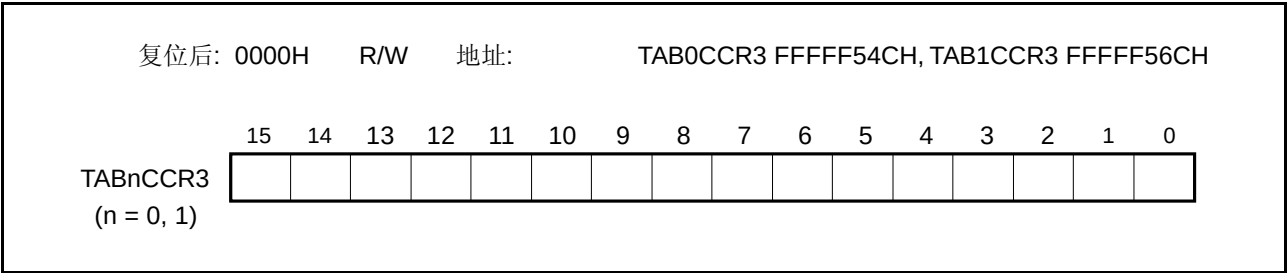
TABnCCR3 寄存器可以在运行中进行读取和写入。

该寄存器可按 16 位为单位进行读写。

复位后本寄存器设置为 0000H。

注意事项 在以下状态中，禁止访问 TABnCCR3 寄存器。有关详细说明，参见 3.4.9 (2) 访问片上专用外围 I/O 寄存器。

- 当 CPU 工作于副时钟，而且主时钟振荡停止时
- 当 CPU 工作于内部振荡时钟时



(a) 作为比较寄存器时的功能

即便在 TABnCTL0.TABnCE 位=1 时，TABnCCR3 寄存器仍可以重新写入。

TABnCCR3 寄存器的设定值传送给 CCR3 缓冲寄存器。当 16 位计数器的值与 CCR3 缓冲寄存器的值相匹配时，就会生成一个比较匹配中断请求信号（INTTABnCC3）。如果此时 TOABn3 引脚输出允许，则 TOABn3 引脚的输出会反相。

(b) 用作捕获寄存器

在自由运行定时器模式下，当 TABnCCR3 寄存器用作捕获寄存器时，若检测到捕获触发输入引脚（TIABn3 引脚）的有效边沿，则 16 位计数器的当前计数值存储于 TABnCCR3 寄存器中。在脉宽测量模式下，若检测到捕获触发输入引脚（TIABn3）的有效边沿，16 位计数器的当前计数值就储存在 TABnCCR3 寄存器中，且 16 位计数器清零（0000H）。

即使捕获操作与对 TABnCCR3 寄存器的读取发生冲突，还是可以正确读取 TABnCCR3 寄存器的值。

备注 n = 0, 1

下表显示了捕获/比较寄存器在各工作模式下的功能，以及怎样向比较寄存器写入数据。

表 8-5. 捕获/比较寄存器在各模式下的功能和如何写入比较寄存器

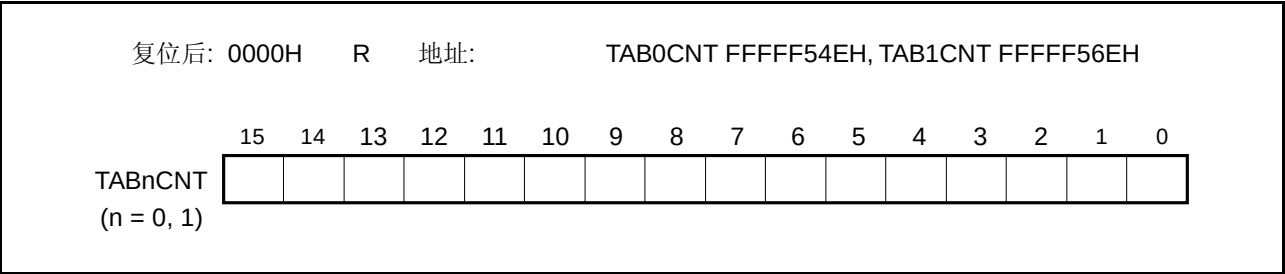
操作模式	捕获/比较寄存器	如何写入比较寄存器
间隔定时器	比较寄存器	可随时写入
外部事件计数器	比较寄存器	可随时写入
外部触发脉冲输出	比较寄存器	批写入
单次脉冲输出	比较寄存器	可随时写入
PWM（脉宽调制信号）输出	比较寄存器	批写入
自由运行定时器	捕获/比较寄存器	可随时写入
脉冲宽度测量	捕获寄存器	—
三角波 PWM 模式	比较寄存器	批写入

(12) TABn 计数器读取缓冲寄存器 (TABnCNT)

TABnCNT 寄存器为读取缓冲寄存器，可以读取 16 位计数器的计数值。
当 TABnCTL0.TABnCE 位 = 1 时，如果读取该寄存器，则会读到 16 位定时器的计数值。
本寄存器为只读寄存器，只能按 16 位读取。
当 TABnCE 位 = 0 时，TABnCNT 寄存器的值被清为 0000H。如果此时读取 TABnCNT 寄存器，则不会读出 16 位计数器的值 (FFFFH)，读出的值为 0000H。
复位后，TABnCE 位被清零的同时 TABnCNT 寄存器的值被清除为 0000H。

注意事项 在以下状态中，禁止访问 TABnCNT 寄存器。有关详细说明，参见 3.4.9 (2) 访问片上专用外围 I/O 寄存器。

- 当 CPU 工作于副时钟，而且主时钟振荡停止时
- 当 CPU 工作于内部振荡时钟时



8.5 操作

定时器 TABn 可以执行如下操作。

操作	TABnCTL1.TABnEST 位 (软件触发位)	TIABn0 引脚 (外部触发输入)	捕获/比较寄存器设置	比较寄存器写入
间隔定时器模式	无效	无效	仅比较寄存器有效	可随时写入
外部事件计数模式 ^{注 1}	无效	无效	仅比较寄存器有效	可随时写入
外部触发脉冲输出模式 ^{注 2}	有效	有效	仅比较寄存器有效	批写入
单次脉冲输出模式 ^{注 2}	有效	有效	仅比较寄存器有效	可随时写入
脉宽调制信号输出模式	无效	无效	仅比较寄存器有效	批写入
自由运行定时器模式	无效	无效	可以切换	可随时写入
脉冲宽度测量模式 ^{注 2}	无效	无效	仅捕获寄存器有效	无效
三角波 PWM 模式	无效	无效	仅比较寄存器有效	批写入

- 注**
1. 在用作外部事件计数模式时，指定不检测 TIABn0 引脚捕获触发输入的有效边沿（通过将 TABnIOC1.TABnIS1 和 TABnIOC1.TABnIS0 位清为“00”）。
 2. 当用作外部触发脉冲输出模式、单次脉冲输出模式和脉冲宽度测量模式时，选择内部时钟作为计数时钟（通过清除 TABnCTL1.TABnEEE 位为 0）。

备注 n = 0, 1

8.5.1 间隔定时器模式 (TABnMD2 至TABnMD0 位 = 000)

在间隔定时器模式下，若 TABnCTL0.TABnCE 位被置为 1，则会以指定时间间隔生成中断请求信号 (INTTABnCC0)，从 TOABn0 引脚输出半周期等于该时间间隔的方波。

通常情况下，在间隔定时器模式中不使用 TABnCCR1 至 TABnCCR33 寄存器。

图 8-2. 间隔定时器的配置

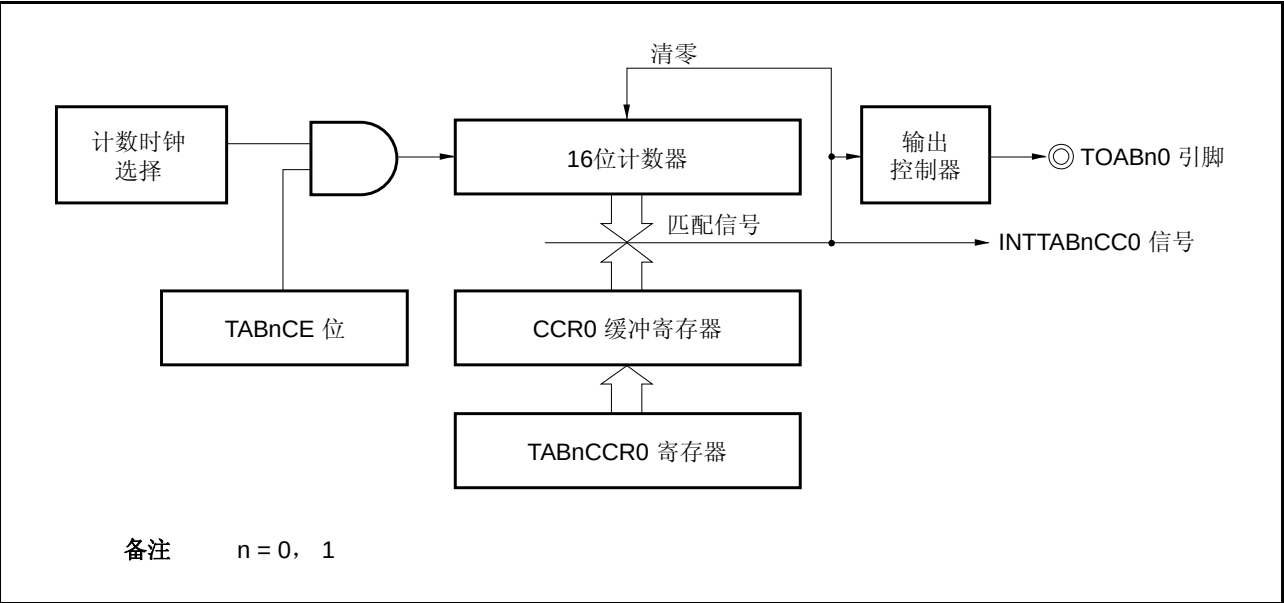
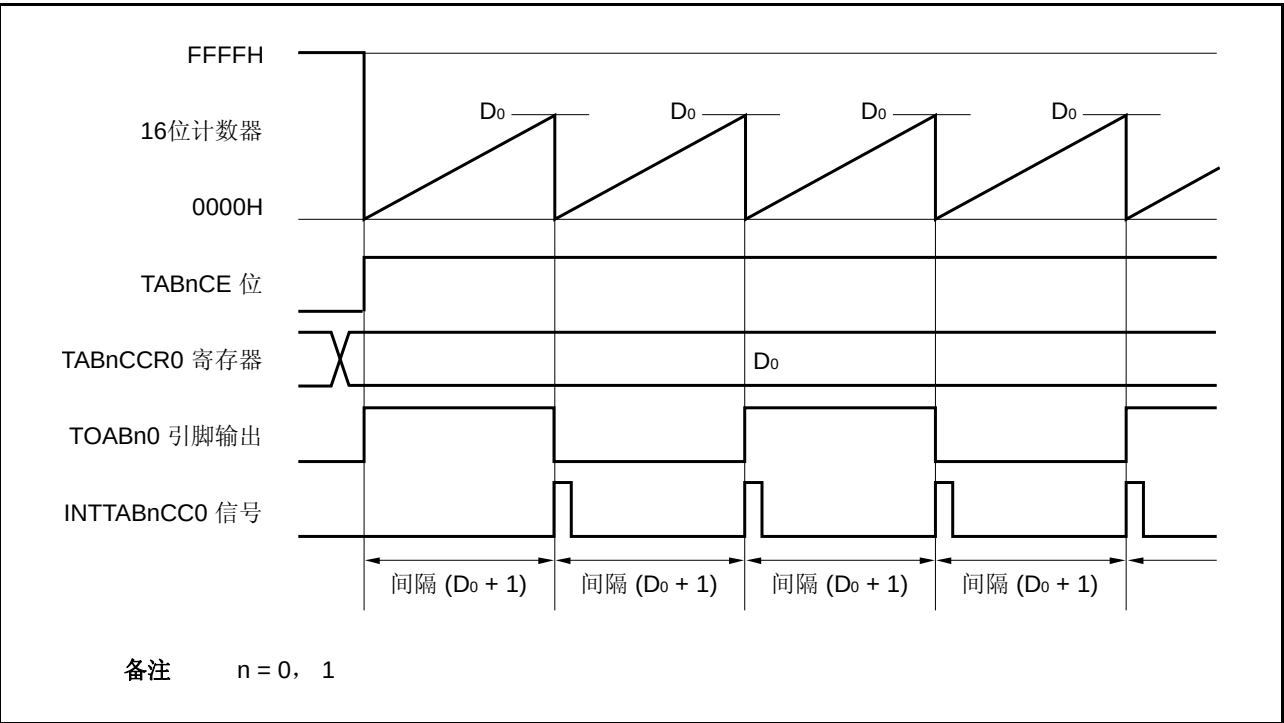


图 8-3. 间隔定时器工作的基本时序



当 TABnCE 位被置为 1 时，16 位计数器的值由计数时钟同步从 FFFFH 清零为 0000H，而且计数器开始计数。此时，TOABn0 引脚的输出反相。此外，TABnCCR0 寄存器的设定值传送给 CCR0 缓冲寄存器。

当 16 位计数器的计数值和 CCR0 缓冲寄存器的值相匹配时，则 16 位计数器清为 0000H，TOABn0 引脚输出反相，并生成一个比较匹配中断请求信号（INTTABnCC0）。

时间间隔可用下式计算。

时间间隔 = (TABnCCR0 寄存器的设定值+ 1) ×计数时钟周期

图 8-4. 间隔定时器模式操作时的寄存器设置 (1/2)

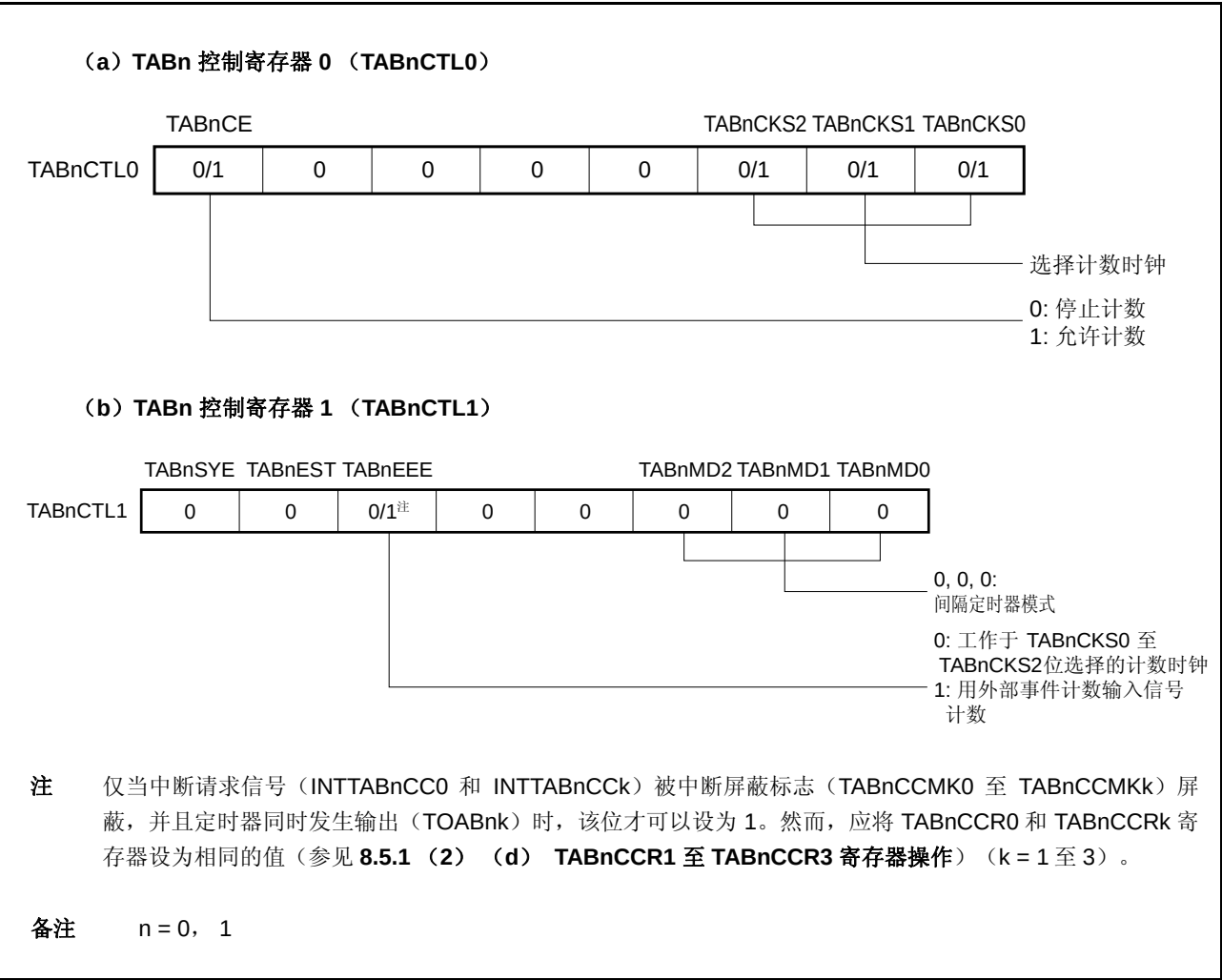
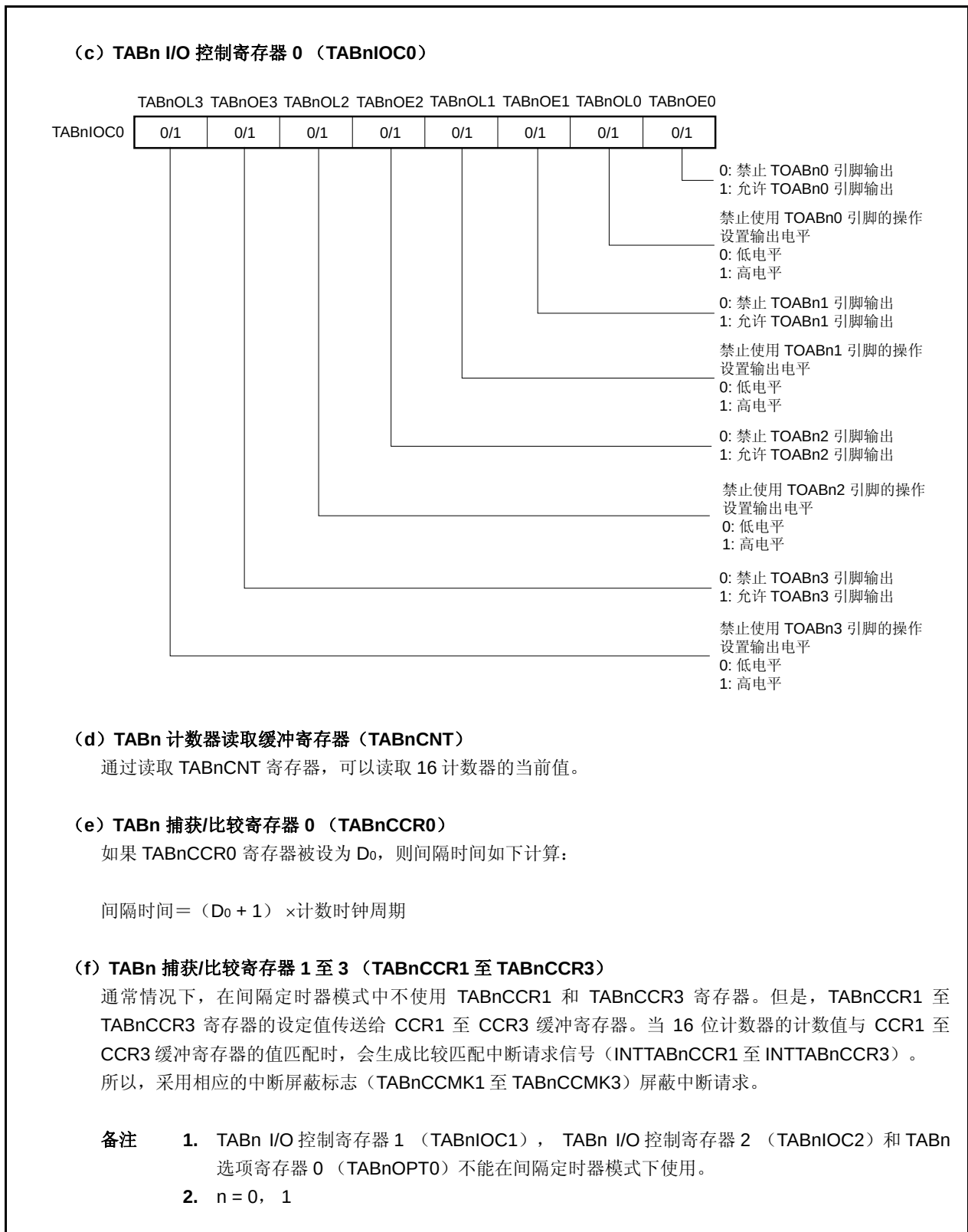
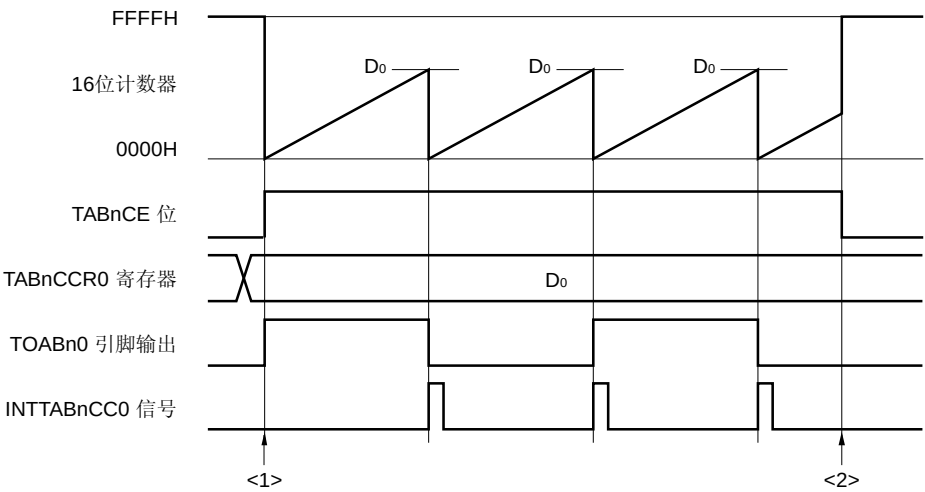


图 8-4. 间隔定时器模式操作时的寄存器设置 (2/2)

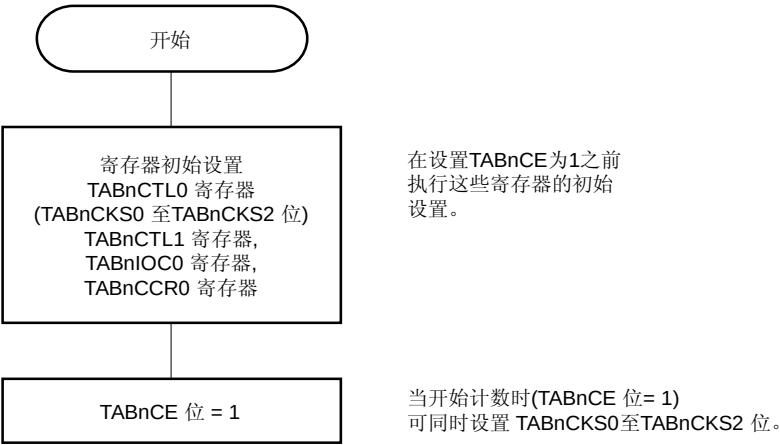


(1) 间隔定时器模式工作流程

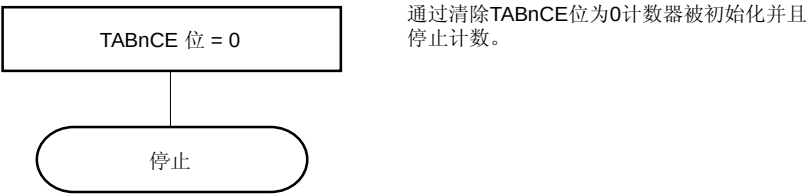
图 8-5. 间隔定时器模式下的软件处理流程



<1> 计数操作开始流程



<2> 计数操作停止流程

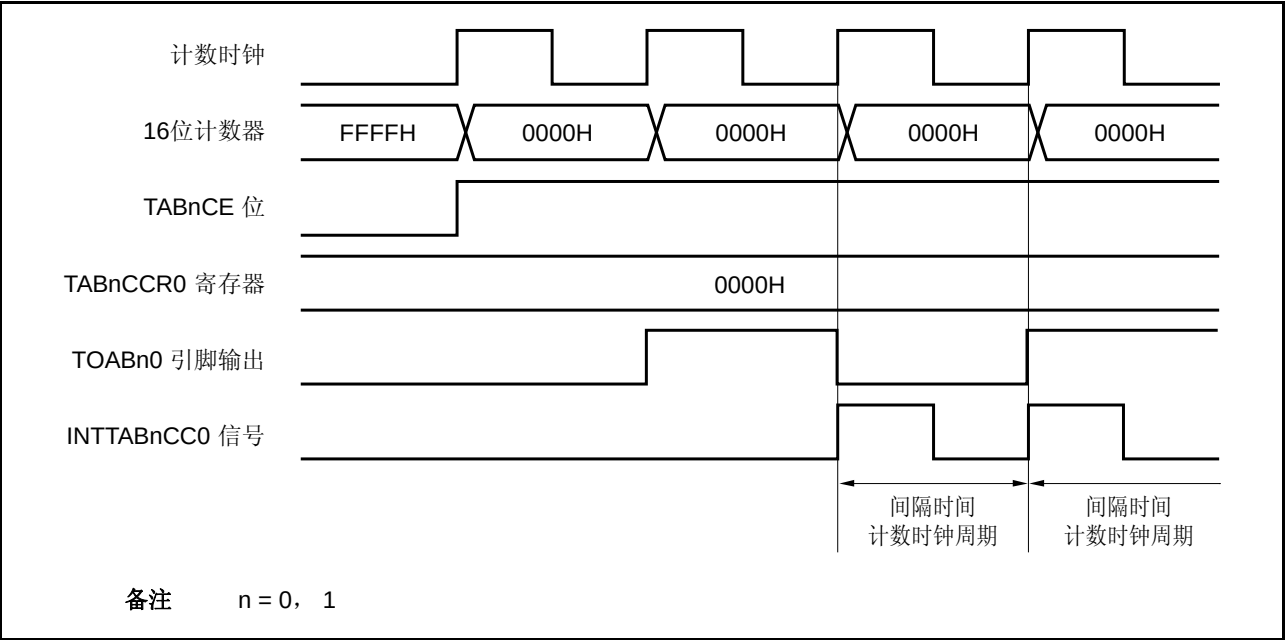


备注 n = 0, 1

(2) 间隔定时器模式工作时序

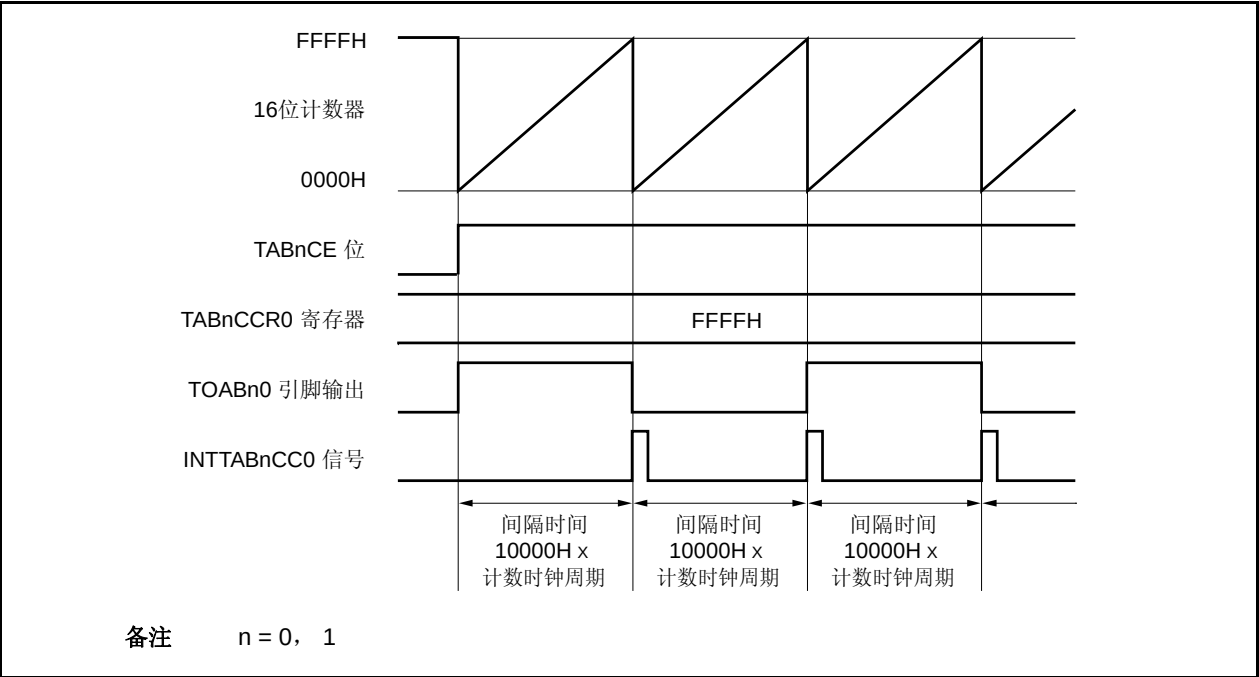
(a) TABnCCR0 寄存器设置为 0000H 时的操作

如果 TABnCCR0 寄存器清为 0000H，第一个计数时钟后的每个计数时钟都会产生 INTTABnCC0 信号，而且，TOABn0 引脚的输出反相。
16 位计数器的值总是 0000H。



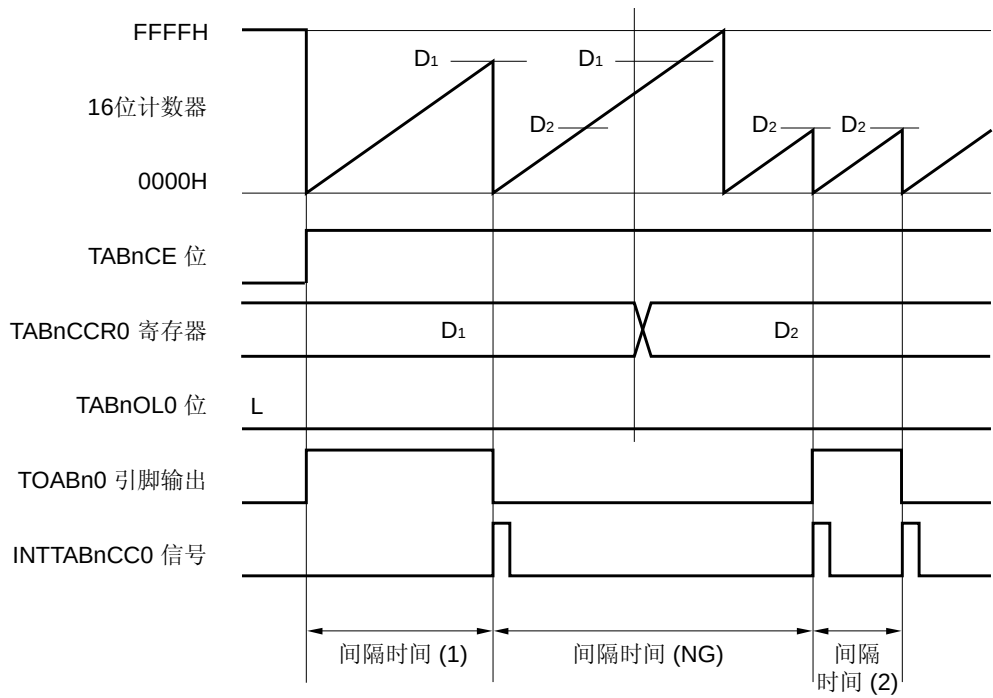
(b) TABnCCR0 寄存器被设置为 FFFFH 时的操作

如果 TABnCCR0 寄存器被设置为 FFFFH，那么 16 位计数器可以计数直至 FFFFH。当下一个计数到来时，计数器同步清为 0000H。产生 INTTABnCC0 信号，并且 TOABn0 引脚电平输出反转。此时，并不产生溢出中断请求信号（INTTABnOV），而且，溢出标志（TABnOPT0.TABnOVF 位）也不会置 1。



(c) 有关重写 TABnCCR0 寄存器的注意事项

要将 TABnCCR0 寄存器的值变为比当前值较小的值，应停止计数，然后改变其设定值。
如果在计数器进行计数期间，对寄存器 TABnCCR0 写入一个较小的数值，则 16 位计数器可能发生溢出。



- 备注
- 1. 间隔时间 (1) : $(D_1 + 1) \times \text{计数时钟周期}$
间隔时间 (NG) : $(10000H + D_2 + 1) \times \text{计数时钟周期}$
间隔时间 (2) : $(D_2 + 1) \times \text{计数时钟周期}$
 - 2. $n = 0, 1$

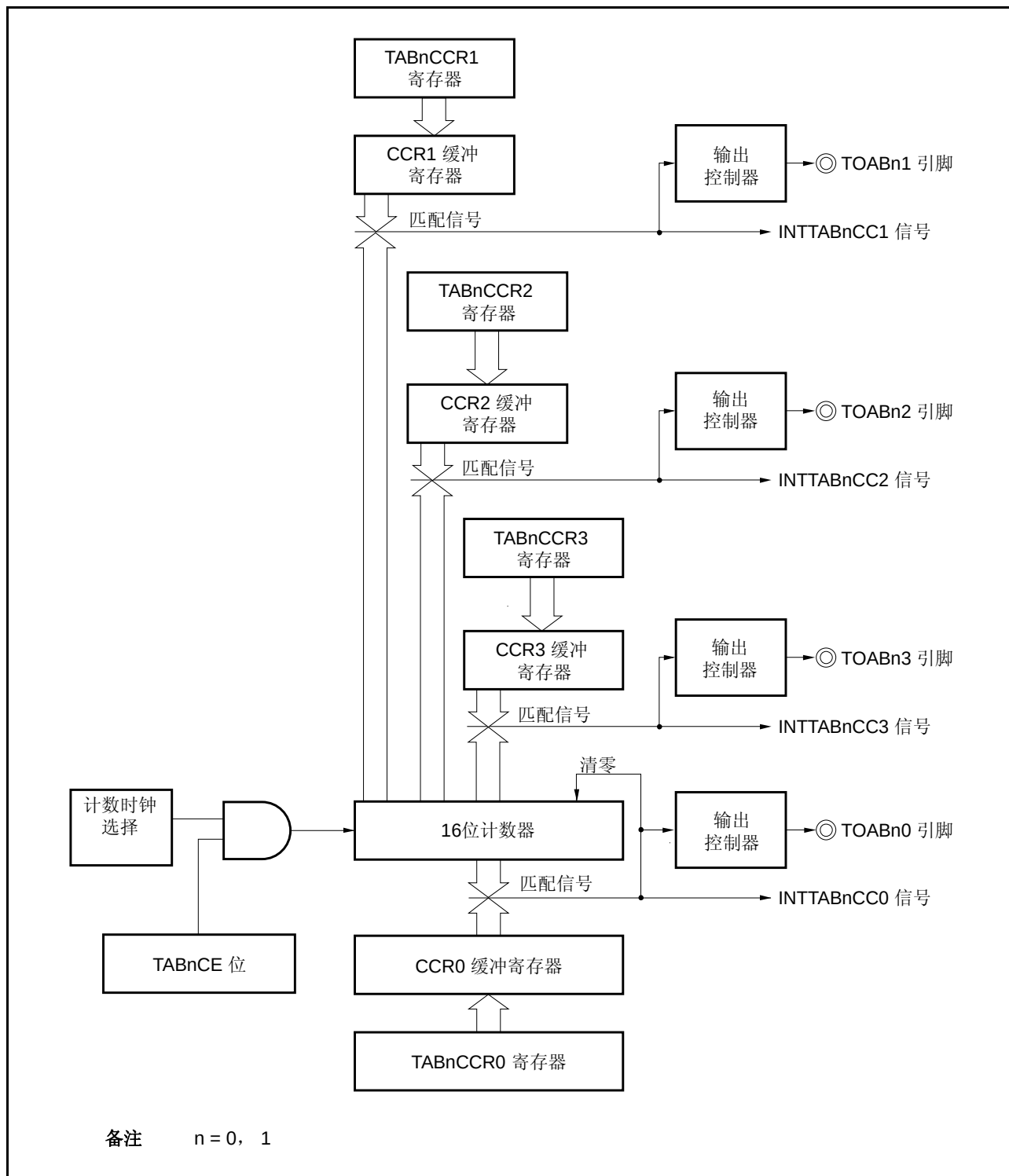
当 16 位计数器计数值大于 D_2 而小于 D_1 时，如果将 TABnCCR0 寄存器的值从 D_1 变为 D_2 ，则在重写 TABnCCR0 寄存器后，计数值会立即传送给 CCR0 缓冲寄存器。结果，用来比较的 16 位计数器的值是 D_2 。

但是，由于计数值已经超过了 D_2 ，所以 16 位计数器就会一直向上累积计数到 FFFFH 直到溢出，然后从 0000H 开始重新计数。当计数值和 D_2 相符时，会生成 INTTABnCC0 信号，且 TOABn0 引脚输出信号反相。

因此，INTTABnCC0 信号可能不会在原先预计的间隔时间“ $(D_1 + 1) \times \text{计数时钟周期}$ ”或“ $(D_2 + 1) \times \text{计数时钟周期}$ ”下产生，而会以 “ $(10000H + D_2 + 1) \times \text{计数时钟周期}$ ” 的间隔产生。

(d) TABnCCR1 至 TABnCCR3 寄存器的操作

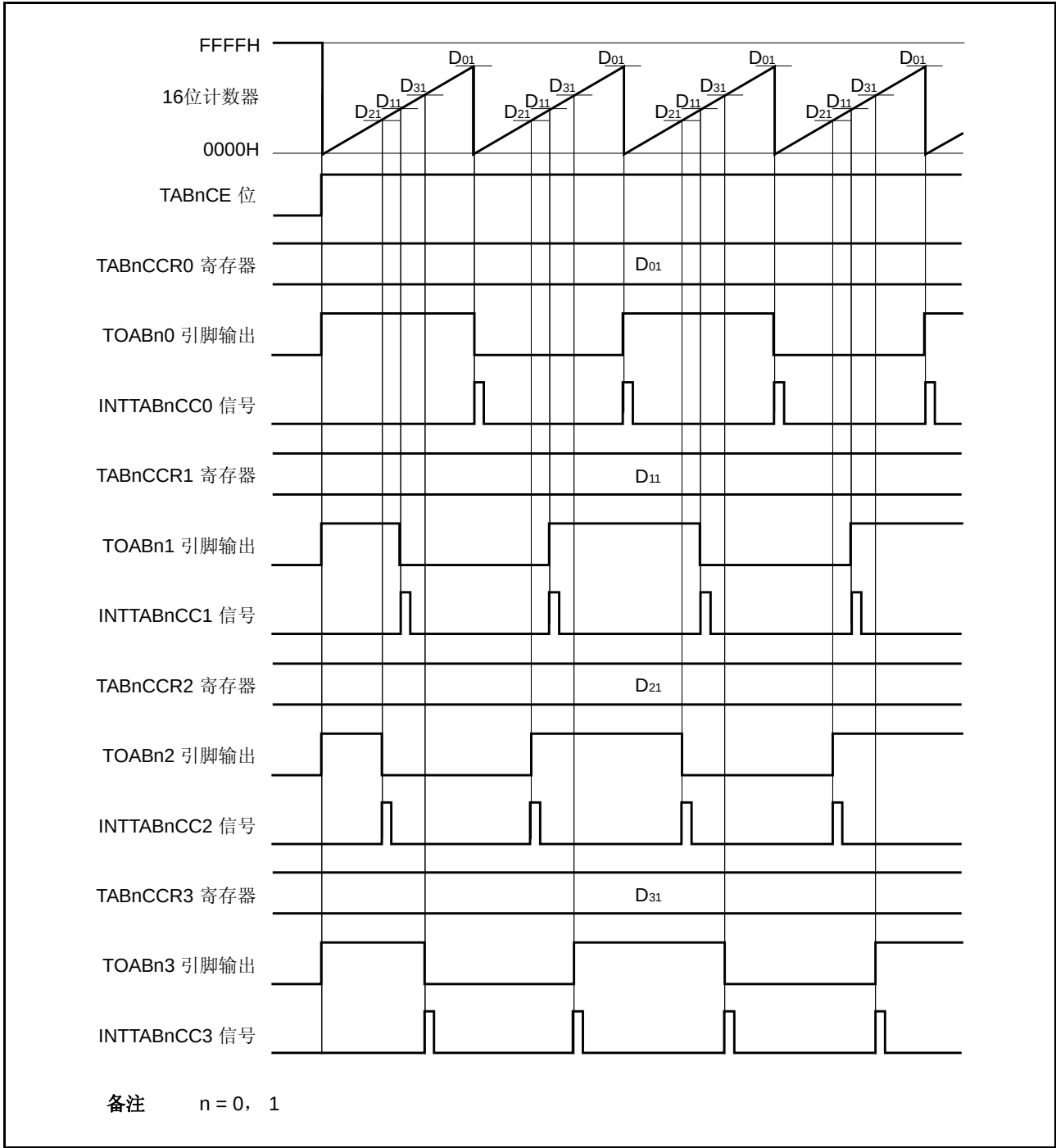
图 8-6. TABnCCR1 至 TABnCCR3 寄存器的配置



如果 TABnCCRk 寄存器的设定值小于 TABnCCR0 寄存器的设定值，则每一个周期产生一次 INTTABnCCk 信号。同时，TOABnk 引脚的输出反相。
TOABnk 引脚输出一个与 TOABn0 引脚输出周期相同的方波。

备注 k = 1 至 3,
 n = 0, 1

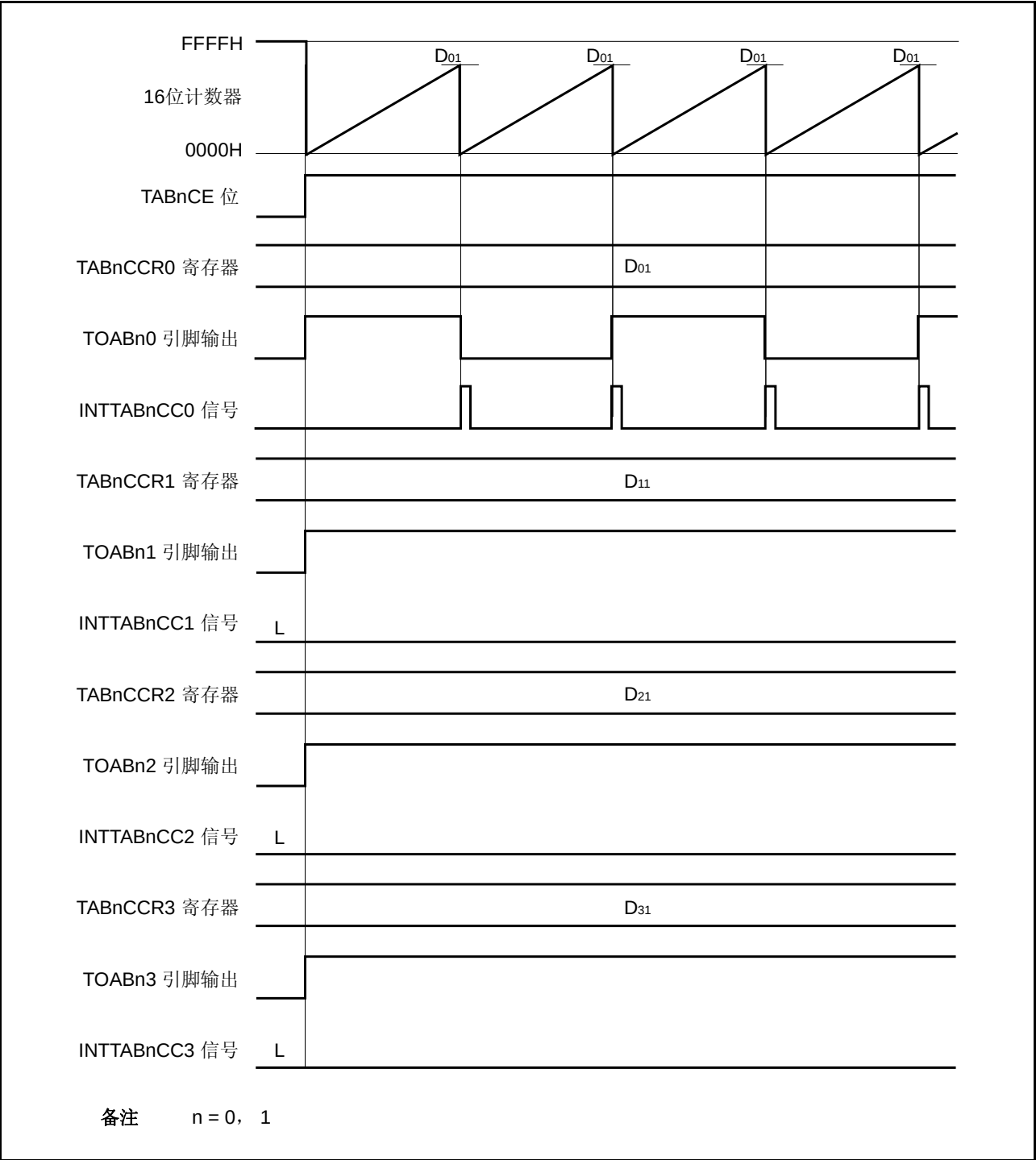
图 8-7. 当 D01 ≥ Dk1 时的时序图



如果 TABnCCRk 寄存器的设定值大于 TABnCCR0 寄存器的设定值，16 位计数器的计数值和 TABnCCRk 寄存器的值不匹配。因而，不会产生 INTTABnCCk 信号，TOABnk 引脚的输出也不会改变。

备注 k = 1 至 3，
 n = 0, 1

图 8-8. 当 D01 < Dk1 时的时序图



8.5.2 外部事件计数器模式 (TABnMD2 至 TABnMD0 位=001)

在外部事件计数器模式下，当 TABnCTL0.TABnCE 位被设置为 1 时，外部事件计数输入信号的有效边沿被计数，每次指定数目的外部事件计数输入信号的有效边沿计数完成时，产生一个中断请求信号 (INTTABnCC0)。不使用 TOABn0 引脚。

通常情况下，在外部事件计数模式中不使用 TABnCCR1 至 TABnCCR3 寄存器。

图 8-9. 外部事件计数器模式的配置

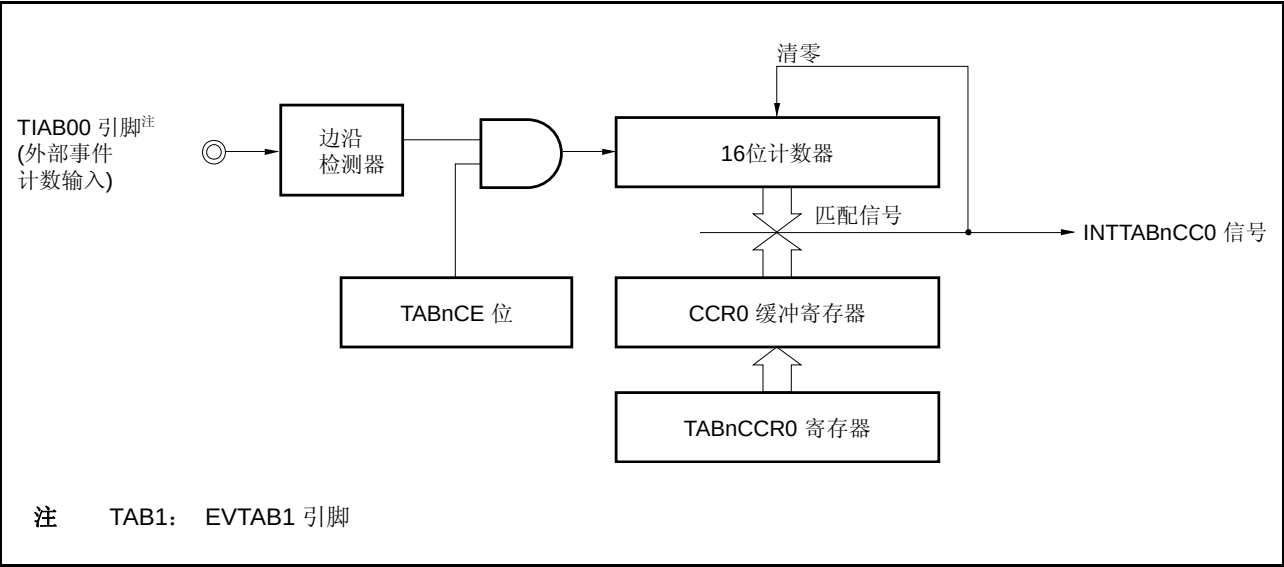
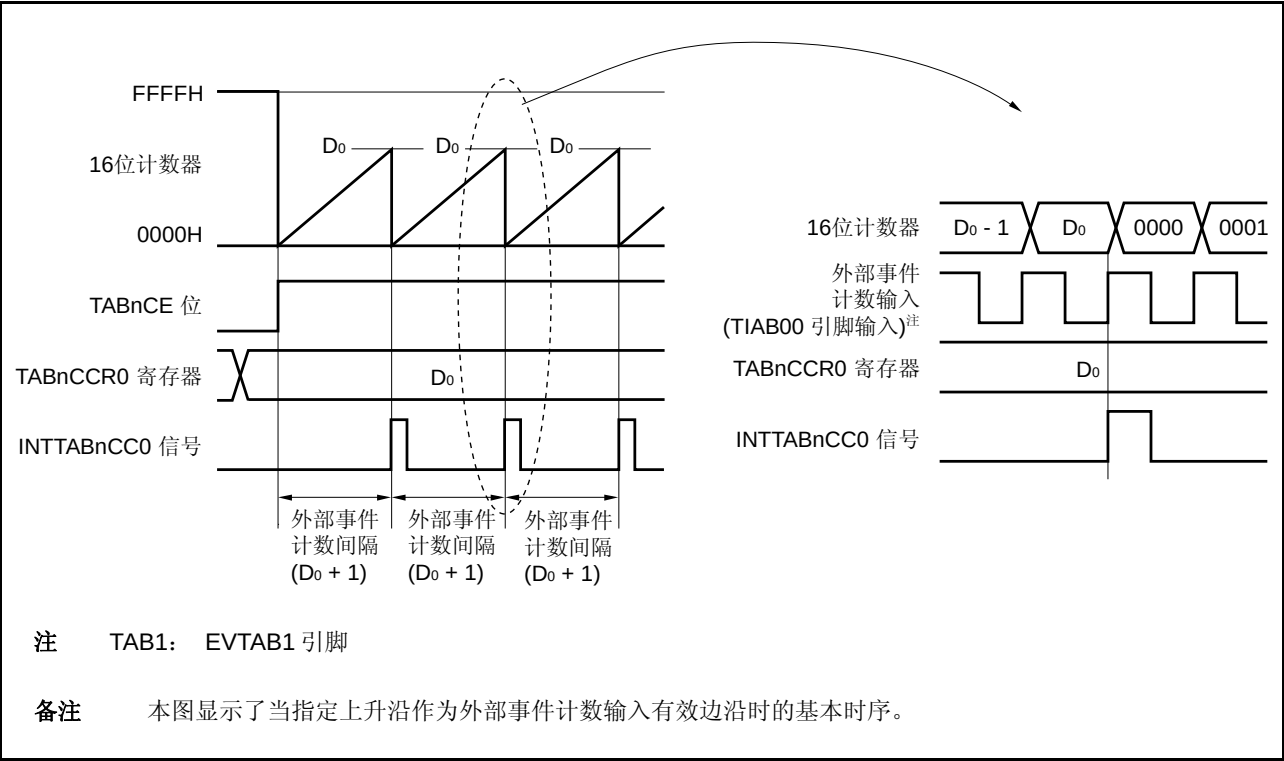


图 8-10. 外部事件计数器模式的基本时序



当 TABnCE 位设置为 1 时，16 位计数器的值由 FFFFH 被清为 0000H。计数器计数检测到的每一次外部事件计数输入的有效边沿。此外，TABnCCR0 寄存器的设定值传送给 CCR0 缓冲寄存器。

当 16 位计数器的计数值和 CCR0 缓冲寄存器的数值相匹配时，16 位计数器清零为 0000H，同时生成一个比较匹配中断请求信号 (INTTABnCC0)。

每当外部事件计数输入的有效边沿检测计数达到 (TABnCCR0 寄存器设定值+1) 时，就会生成 INTTABnCC0 (比较匹配中断请求) 信号。

图 8-11. 外部事件计数模式下的寄存器操作设置 (1/2)

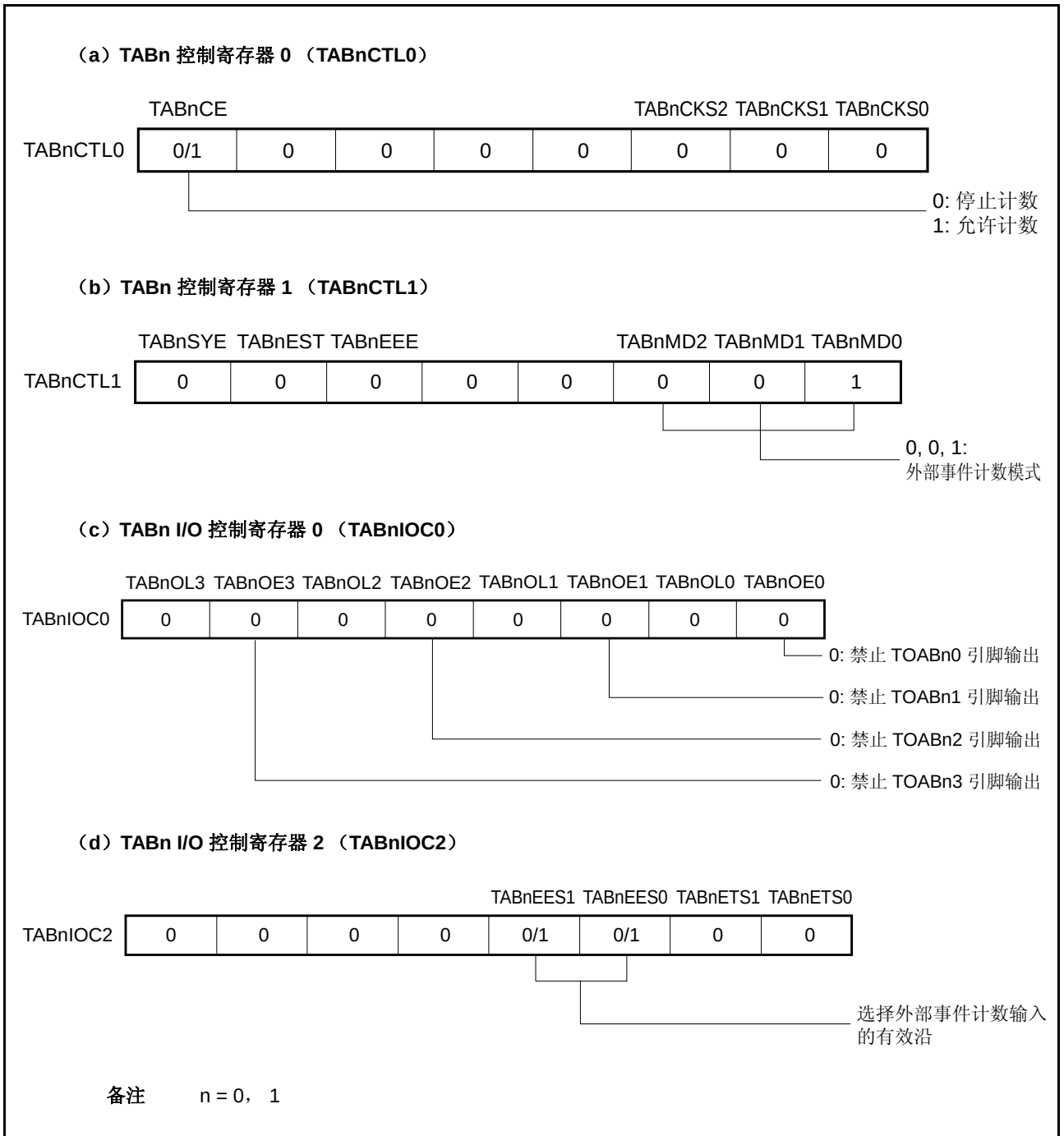


图 8-11. 外部事件计数模式下的寄存器操作设置 (2/2)

(e) TABn 计数器读取缓冲寄存器 (TABnCNT)

可以通过读取 TABnCNT 寄存器，读取 16 位计数器的计数值。

(f) TABn 捕获/比较寄存器 0 (TABnCCR0)

如果 TABnCCR0 寄存器的值被设定为 D_0 ，当外部事件计数值达到 $(D_0 + 1)$ 时，计数器被清 0，同时产生一个比较匹配中断请求信号 (INTTABnCC0)。

(g) TABn 捕获/比较寄存器 1 至 3 (TABnCCR1 至 TABnCCR3)

通常情况下，在外部事件计数模式中不使用 TABnCCR1 至 TABnCCR3 寄存器。但是，TABnCCR1 至 TABnCCR3 寄存器的设定值传送给 CCR1 至 CCR3 缓冲寄存器。当 16 位计数器的计数值和 CCR1 至 CCR3 缓冲寄存器的值匹配时，会产生比较匹配中断请求信号 (INTTABnCC1 至 INTTABnCC3)。因此，可以使用中断屏蔽标志 (TABnCCMK1 至 TABnCCMK3) 来屏蔽中断信号。

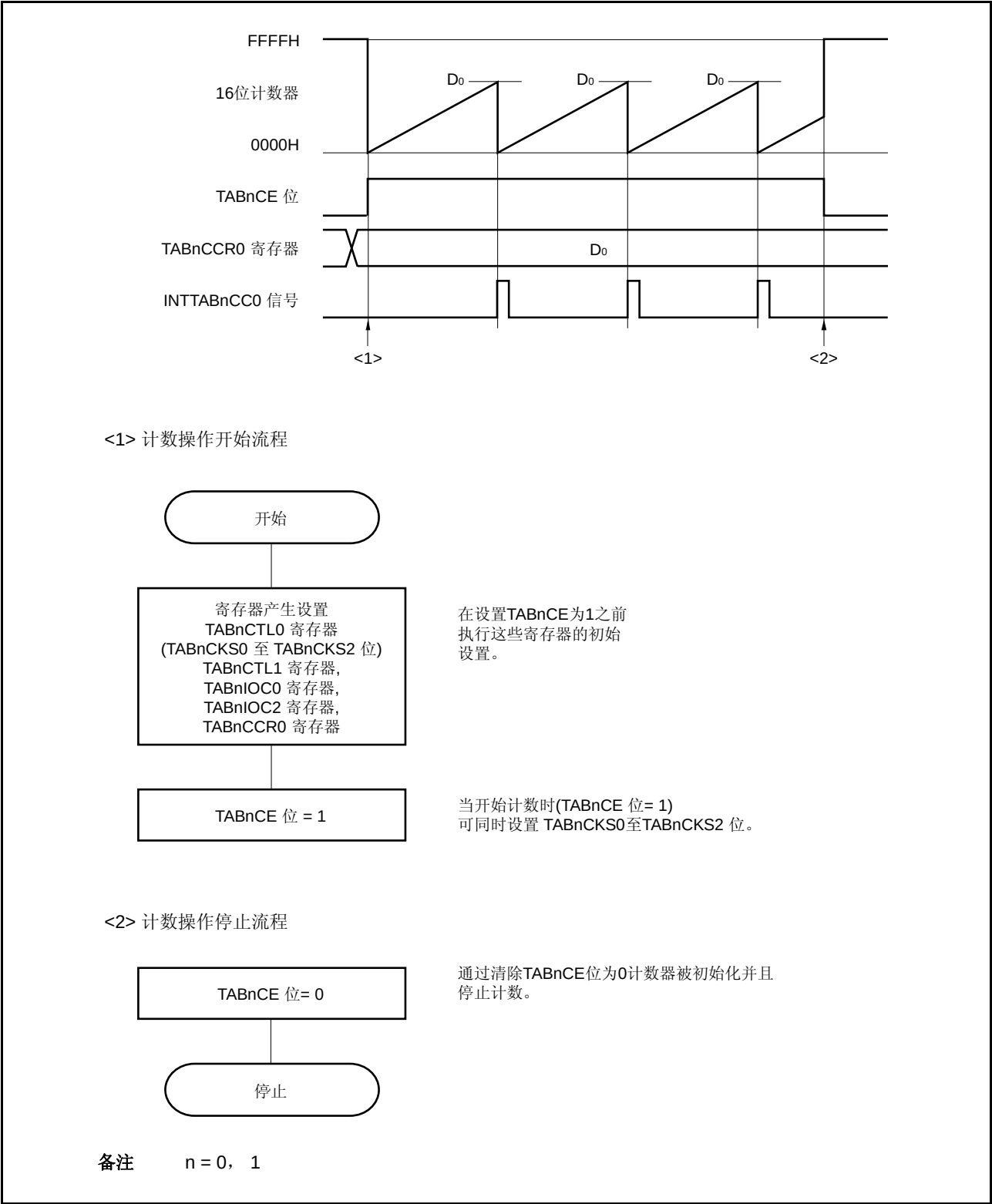
注意事项 对于 TAB0 来说，当外部时钟用作计数时钟时，外部时钟只能从 TIAB00 引脚输入。此时，把 TAB0IOC1.TAB0IS1 和 TAB0IOC1.TAB0IS0 位清零为 00 (捕获触发输入 (TIAB00 引脚)：无边沿检测)。

备注

1. 在外部事件计数模式下，不使用 TABn 输入/输出控制寄存器 1 (TABnIOC1) 和 TABn 选项寄存器 0 (TABnOPT0)。
2. $n = 0, 1$

(1) 外部事件计数模式的操作流程

图 8-12. 外部事件计数模式下的软件处理流程

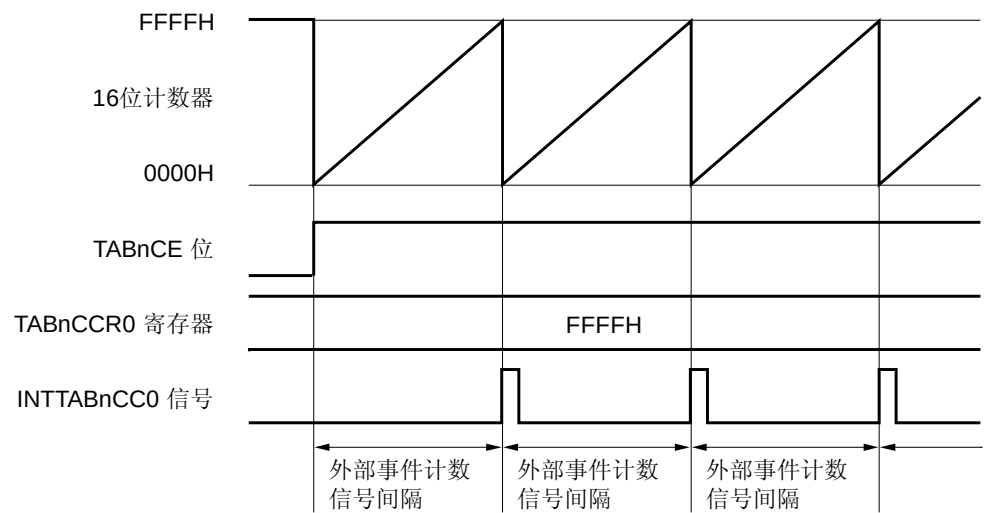


(2) 外部事件计数模式的工作时序

- 注意事项
- 1. 外部事件计数模式下，不要将 TABnCCR0 寄存器设置为 0000H。
 - 2. 外部事件计数模式下，禁止使用定时器输出。如果在外部事件计数模式下使用定时器输出，应设置为间隔定时器模式，并且选择外部事件计数输入作为计数时钟（TABnCTL1.TABnMD2 至 TABnCTL1.TABnMD0 位=000，TABnCTL1.TABnEEE 位= 1）。

(a) TABnCCR0 寄存器被设置为 FFFFH 时的操作

如果 TABnCCR0 寄存器被设置为 FFFFH，16 位计数器在每次检测到外部事件输入信号的有效边沿时计数到 FFFFH。当下一个计数到来时 16 位计数器同步清为 0000H，并且产生 INTTABnCC0 信号。此时，不设置 TABnOPT0.TABnOVF 位。

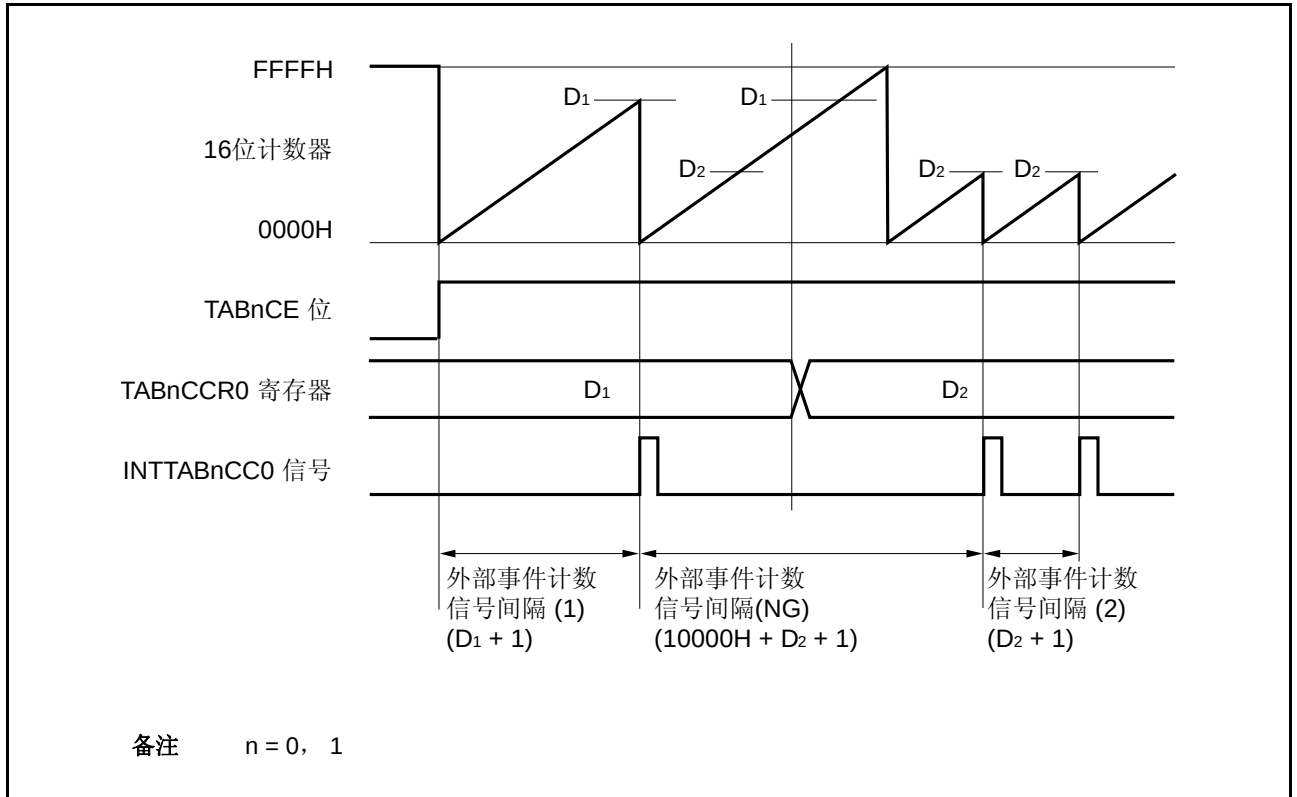


备注 n = 0, 1

(b) 重写 TABnCCR0 寄存器的注意事项

要将 TABnCCR0 寄存器的值变为比当前值较小的值，应停止计数，然后改变其设定值。

如果在计数器进行计数期间，对寄存器 TABnCCR0 写入一个较小的数，则 16 位计数器可能发生溢出。



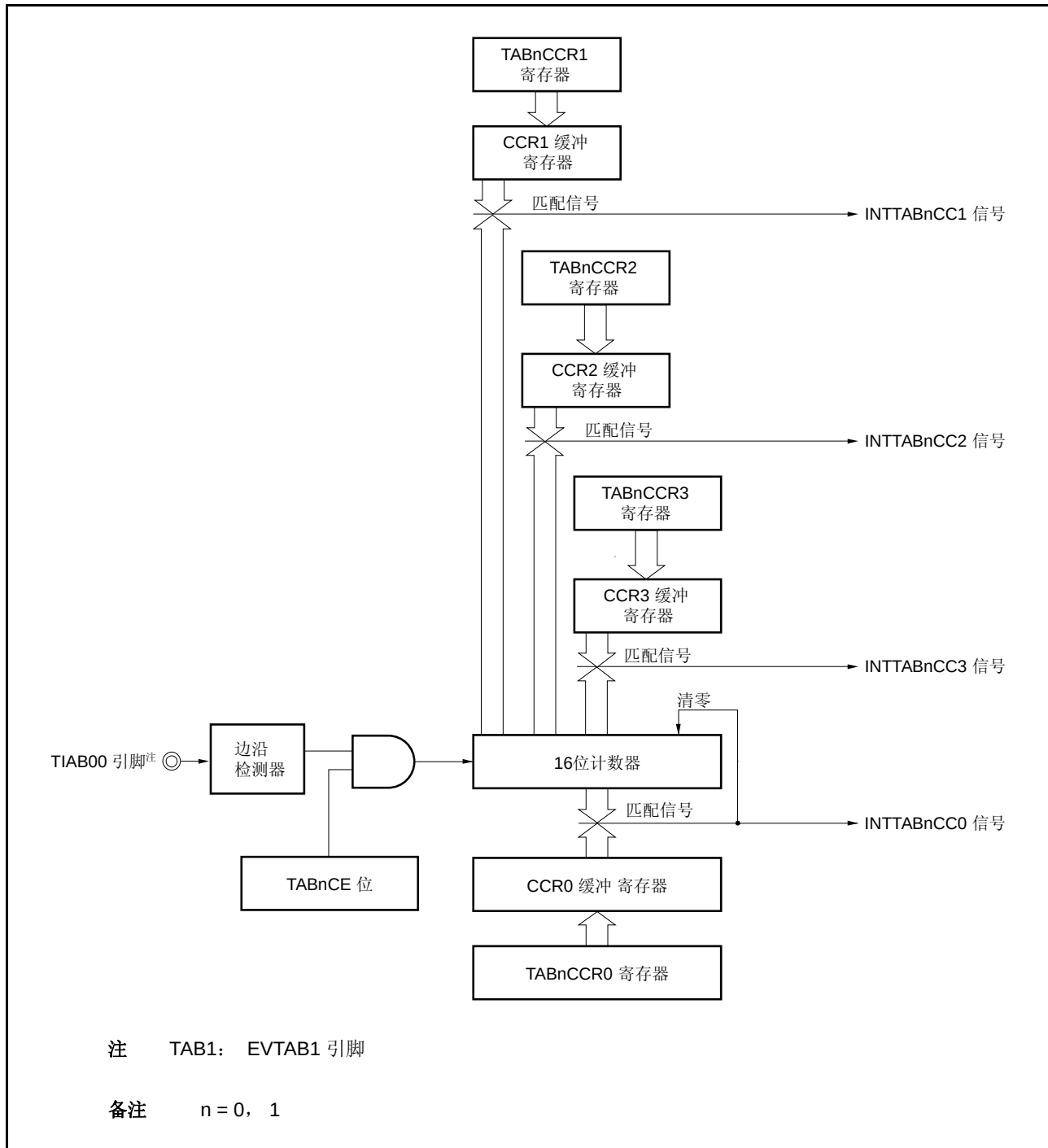
当 16 位计数器计数值大于 D₂ 而小于 D₁ 时，如果将 TABnCCR0 寄存器的值从 D₁ 变为 D₂，则在重写 TABnCCR0 寄存器后，计数值会立即传送给 CCR0 缓冲寄存器。因而，与 16 位计数器相比较的值会是 D₂。

但是，由于计数值已经超过了 D₂，所以 16 位计数器就会一直向上累积计数到 FFFFH 直到溢出，然后从 0000H 开始重新计数。当计数值与 D₂ 匹配时，会生成 INTTABnCC0 信号。

因此，INTTABnCC0 中断请求信号可能不会在原先预计的“(D₁ + 1)”次或“(D₂ + 1)”次有效边沿上生成，而可能在“(10000H + D₂ + 1)”次有效边沿上生成。

(c) TABnCCR1 至 TABnCCR3 寄存器的操作

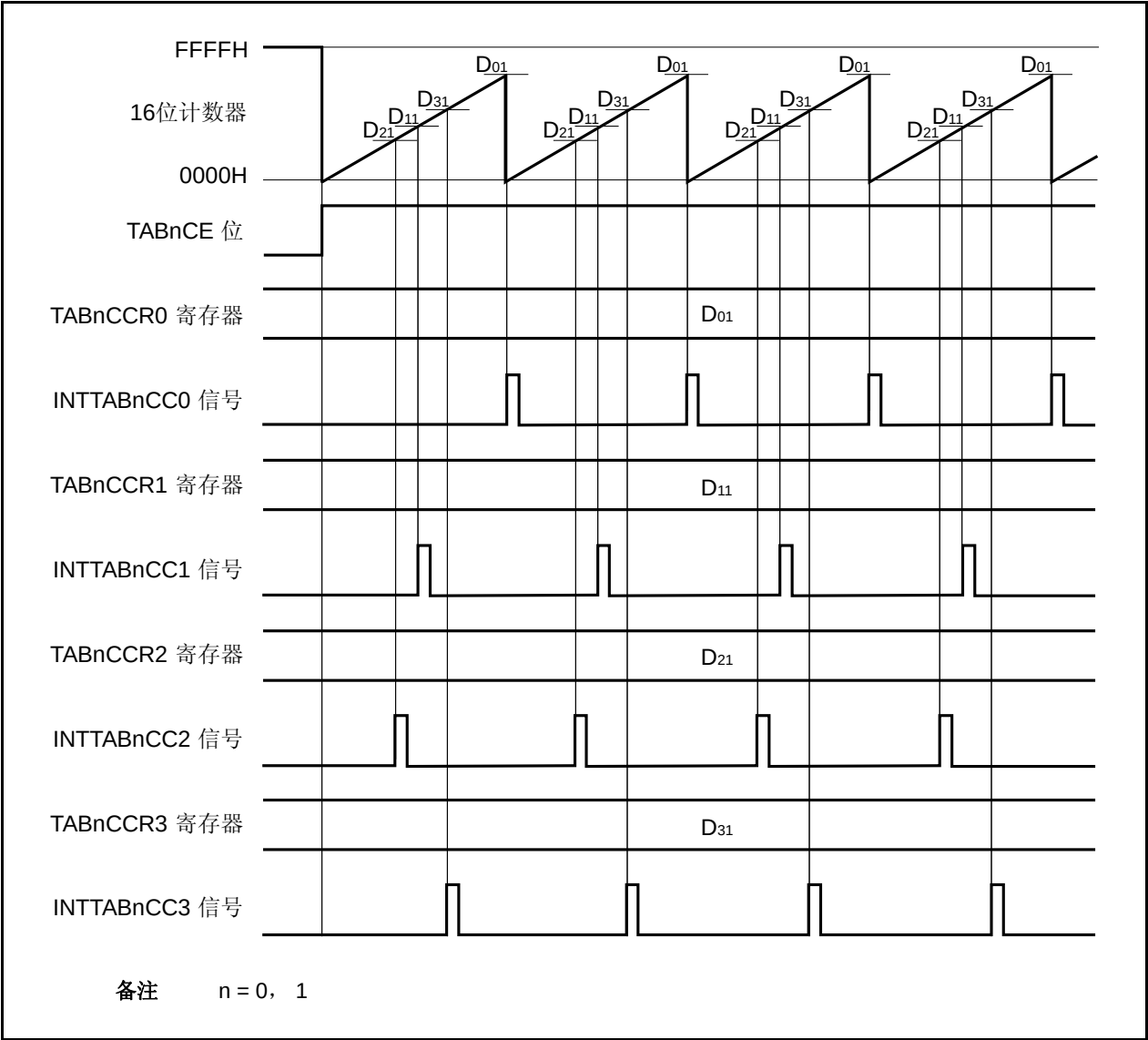
图 8-13. TABnCCR1 至 TABnCCR3 寄存器的配置



如果 TABnCCRk 寄存器的设定值小于 TABnCCR0 寄存器的设定值，则每一个周期产生一次 INTTABnCCk 信号。

备注 k = 1 至 3
 n = 0, 1

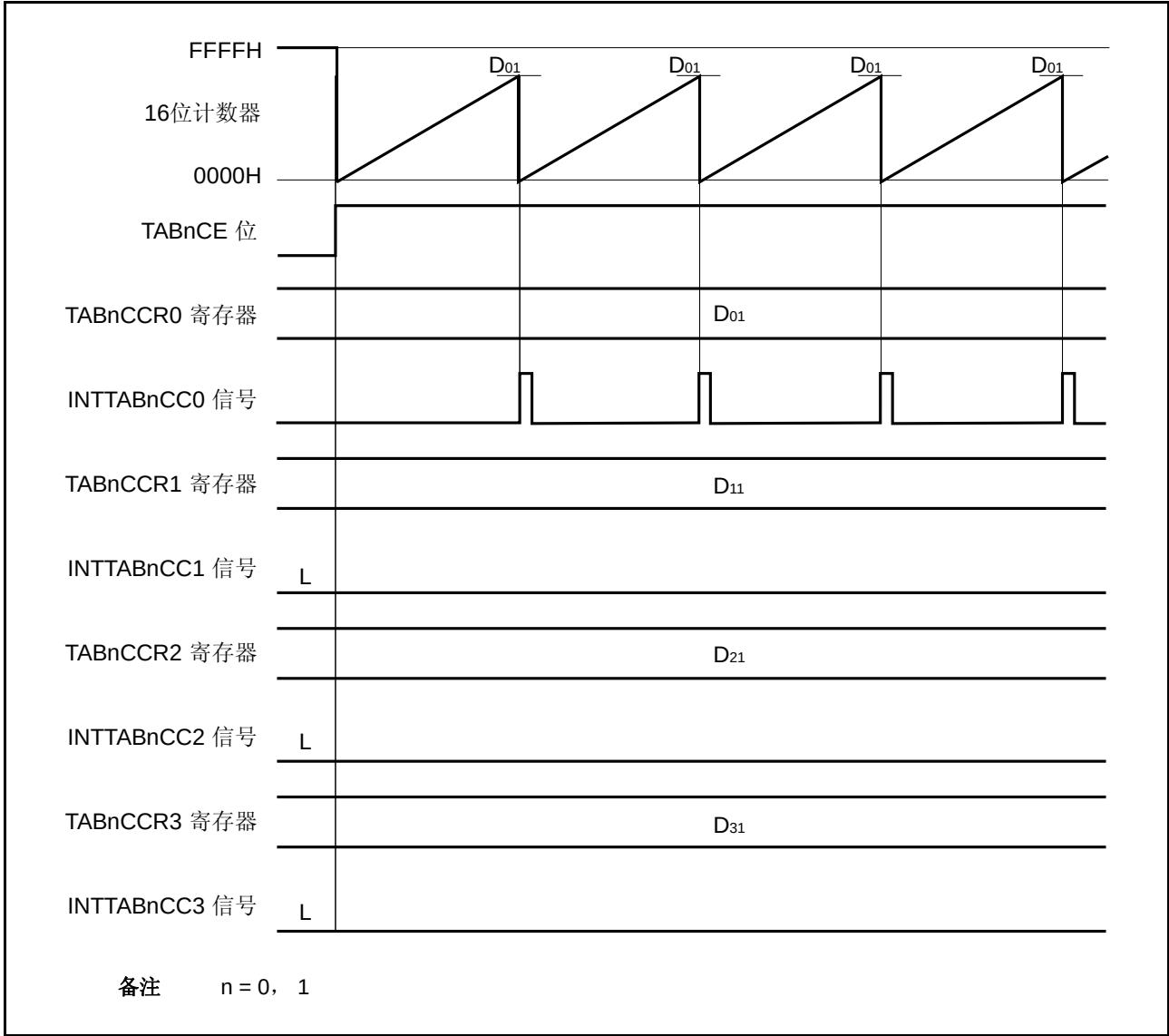
图 8-14. 当 $D_{01} \geq D_{k1}$ 时的时序图



如果 TABnCCRk 寄存器的设定值大于 TABnCCR0 寄存器的设定值，则不会产生 INTTABnCCk 中断请求信号，因为 16 位计数器的计数值和 TABnCCRk 寄存器的值不匹配。

备注 k = 1 至 3
 n = 0, 1

图 8-15. 当 D01 < Dk1 时的时序图



8.5.3 外部触发脉冲输出模式 (TABnMD2 至 TABnMD0 位=010)

在外部触发脉冲输出模式下, 当 TABnCTL0.TABnCE 位设置为 1 时, TABn 等待触发。当检测到外部触发输入信号的有效沿时, TABn 开始计数, 并从 TOABn1 至 TOABn3 引脚输出 PWM 波形。

不使用外部触发, 而是使用软件触发, 同样可以得到输出脉冲。当使用软件触发时, TOABn0 引脚输出一个方波, 其半周期为输出脉宽调制信号的一个周期。

图 8-16. 外部触发脉冲输出模式的配置

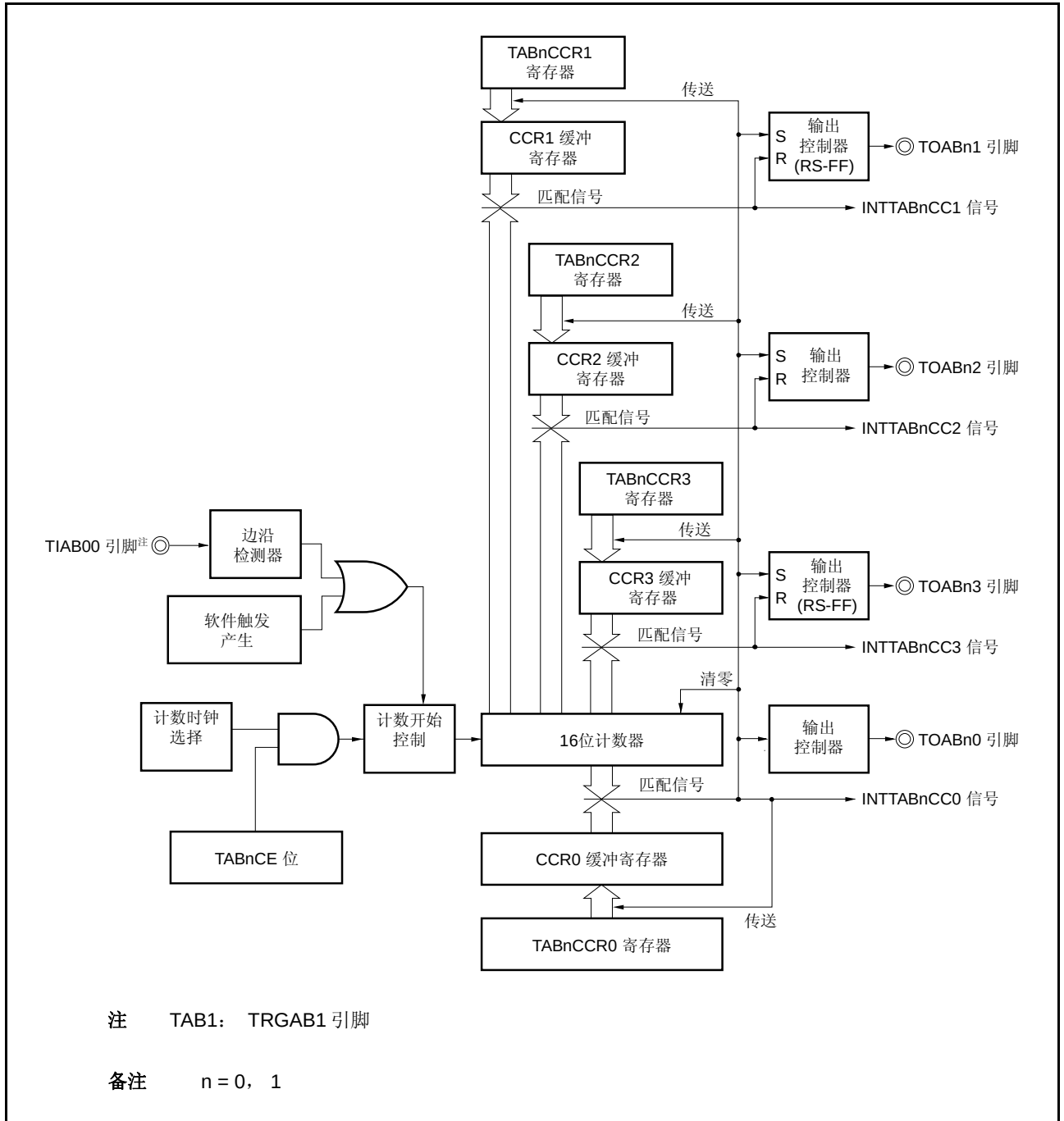
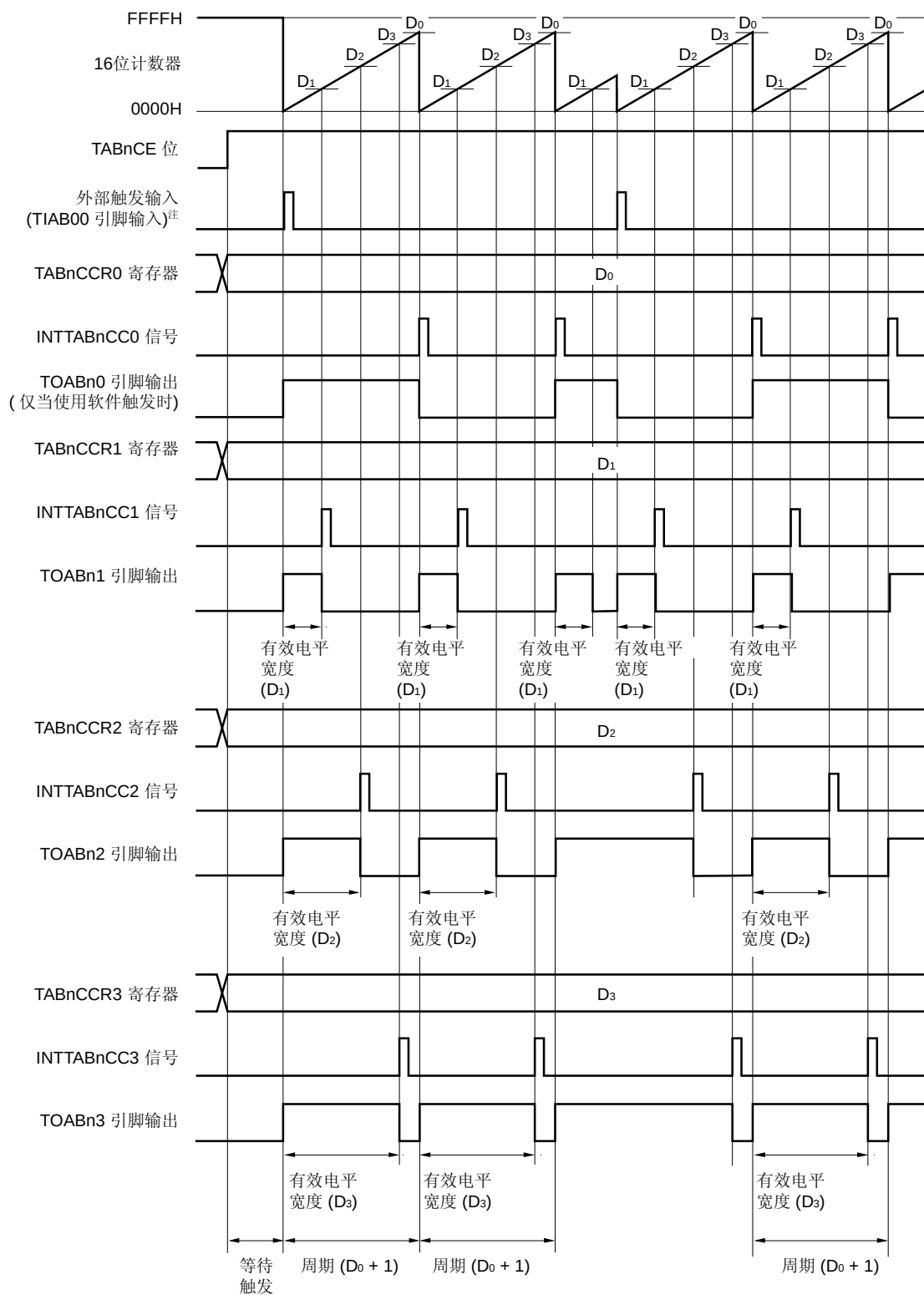


图 8-17. 外部触发脉冲输出模式的基本时序



注 TAB1: TRGAB1 引脚

备注 $n = 0, 1$

当 TABnCE 位设置为 1 时，TABn 等待触发。当触发产生时，16 位计数器从 FFFFH 清零为 0000H，同时开始计数，并从 TOABnk 引脚输出一个脉宽调制信号波形。若在计数运行中再次产生触发，则计数器清为 0000H 并重新开始计数。（TOABn0 引脚输出反转。在出现触发时，不管当前状态如何（高/低），TOABnk 引脚都会输出一个高电平。）

脉冲宽度调制信号波形的有效电平宽度、周期、占空比计算如下。

有效电平宽度 = (TABnCCRk 寄存器的设定值) × 计数时钟周期

周期 = (TABnCCR0 寄存器的设定值 + 1) × 计数时钟周期

占空比 = (TABnCCRk 寄存器的设定值) / (TABnCCR0 寄存器的设定值 + 1)

当 16 位计数器在其计数值与 CCR0 缓冲寄存器的值匹配之后进行下一次向上计数时，会产生比较匹配请求信号 INTTABnCC0，且 16 位计数器清零为 0000H。当 16 位计数器的计数值和 CCRk 缓冲寄存器的值匹配时，比较匹配中断请求信号 (INTTABnCCk) 就产生了。

当 16 位计数器的计数值与 CCR0 缓冲寄存器的值匹配，而且 16 位计数器清零为 0000H 时，TABnCCRm 寄存器设置的值就传送到 CCRm 缓冲寄存器。

外部触发输入信号的有效边沿，或将软件触发 (TABnCTL1.TABnEST 位) 设为 1，都可当作触发信号。

备注 k = 1 至 3
m = 0 至 3
n = 0, 1

图 8-18. 外部触发脉冲输出模式的寄存器设置 (1/3)

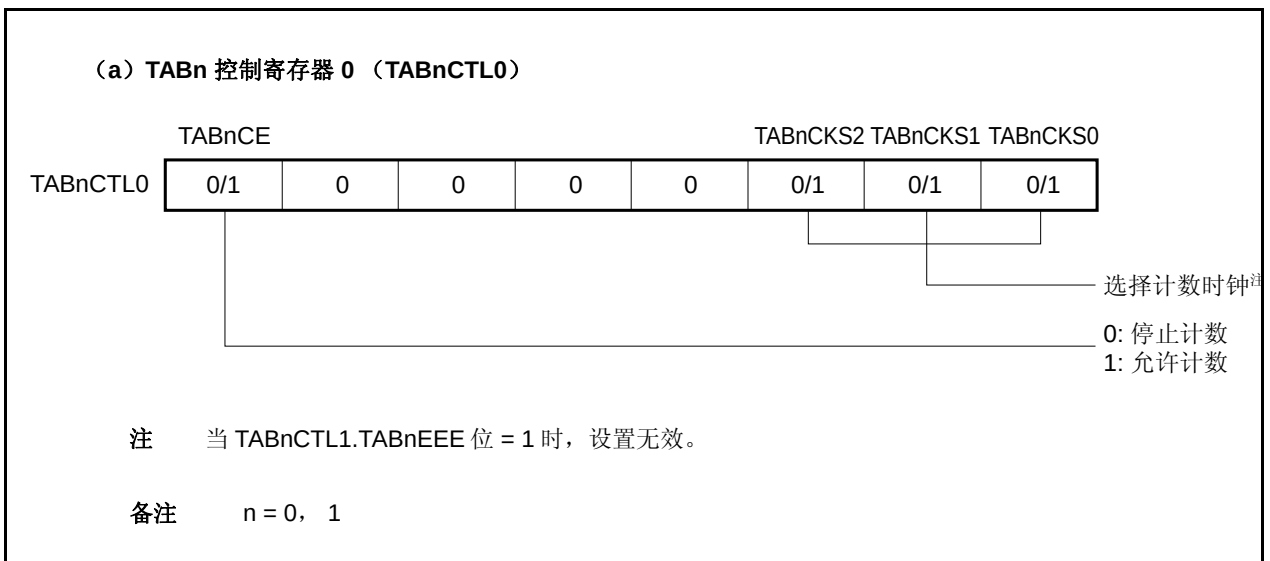
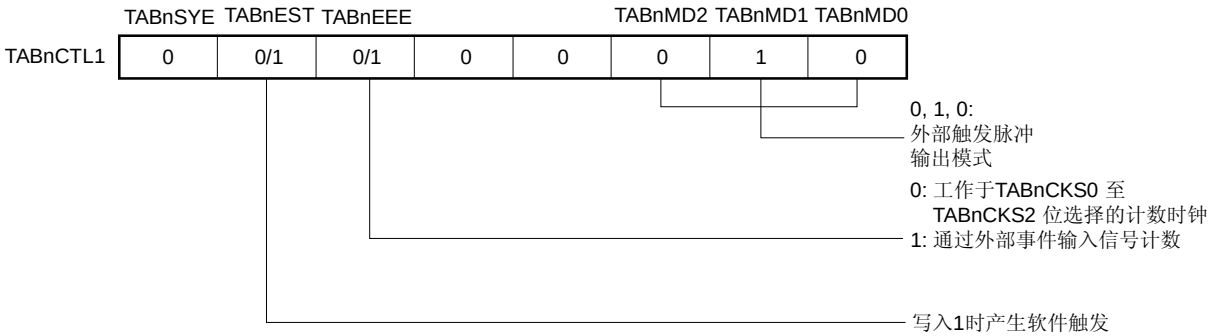
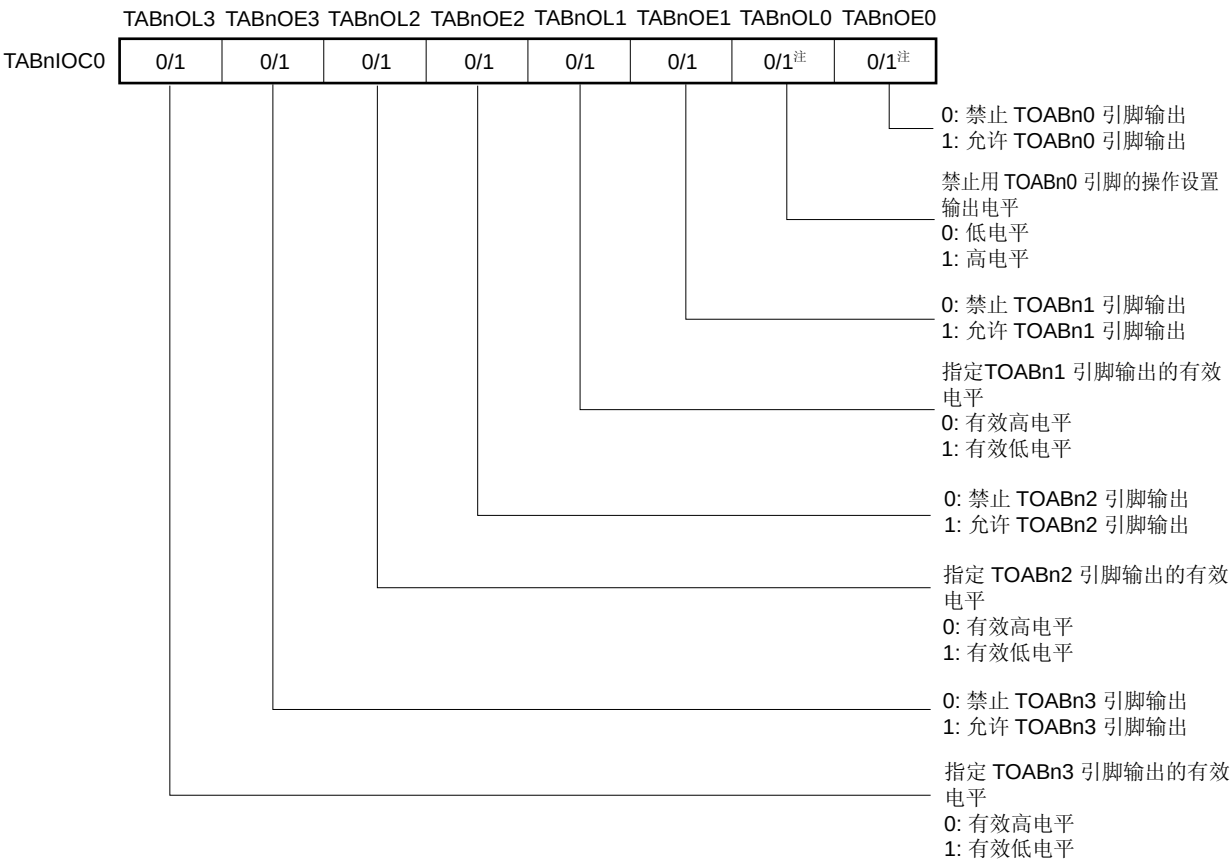


图 8-18. 外部触发脉冲输出模式的寄存器设置 (2/3)

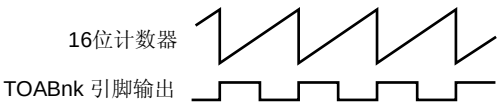
(b) TABn 控制寄存器 1 (TABnCTL1)



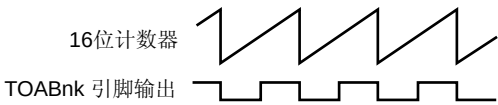
(c) TABn I/O 控制寄存器 0 (TABnIOC0)



· 当 TABnOLk 位 = 0



· 当 TABnOLk 位 = 1

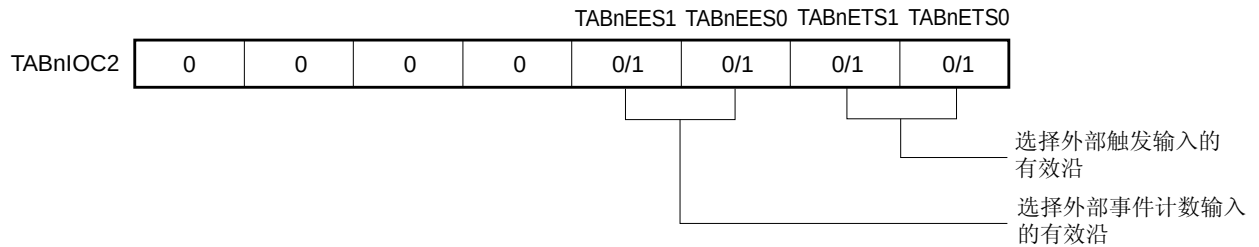


注 在外部触发脉冲输出模式，当不使用 TOABn0 引脚时，将此位清除为 0。

备注 n = 0, 1

图 8-18. 外部触发脉冲输出模式的寄存器设置 (3/3)

(d) TABn I/O 控制寄存器 2 (TABnIOC2)



(e) TABn 计数器读取缓冲寄存器 (TABnCNT)

通过读取 TABnCNT 寄存器，可以读出 16 位计数器的值。

(f) TABn 捕获/比较寄存器 0 至 3 (TABnCCR0 至 TABnCCR3)

如果将 TABnCCR0 寄存器的值设置为 D_0 ，将 TABnCCR1 寄存器的值设置为 D_1 ，将 TABnCCR2 寄存器的值设置为 D_2 ，将 TABnCCR3 寄存器的值设置为 D_3 ，那么 PWM 波形的周期和有效电平如下所示。

周期 = $(D_0 + 1) \times$ 计数时钟周期

TOABn1 引脚脉宽调制波形有效电平宽度 = $D_1 \times$ 计数时钟周期

TOABn2 引脚脉宽调制波形有效电平宽度 = $D_2 \times$ 计数时钟周期

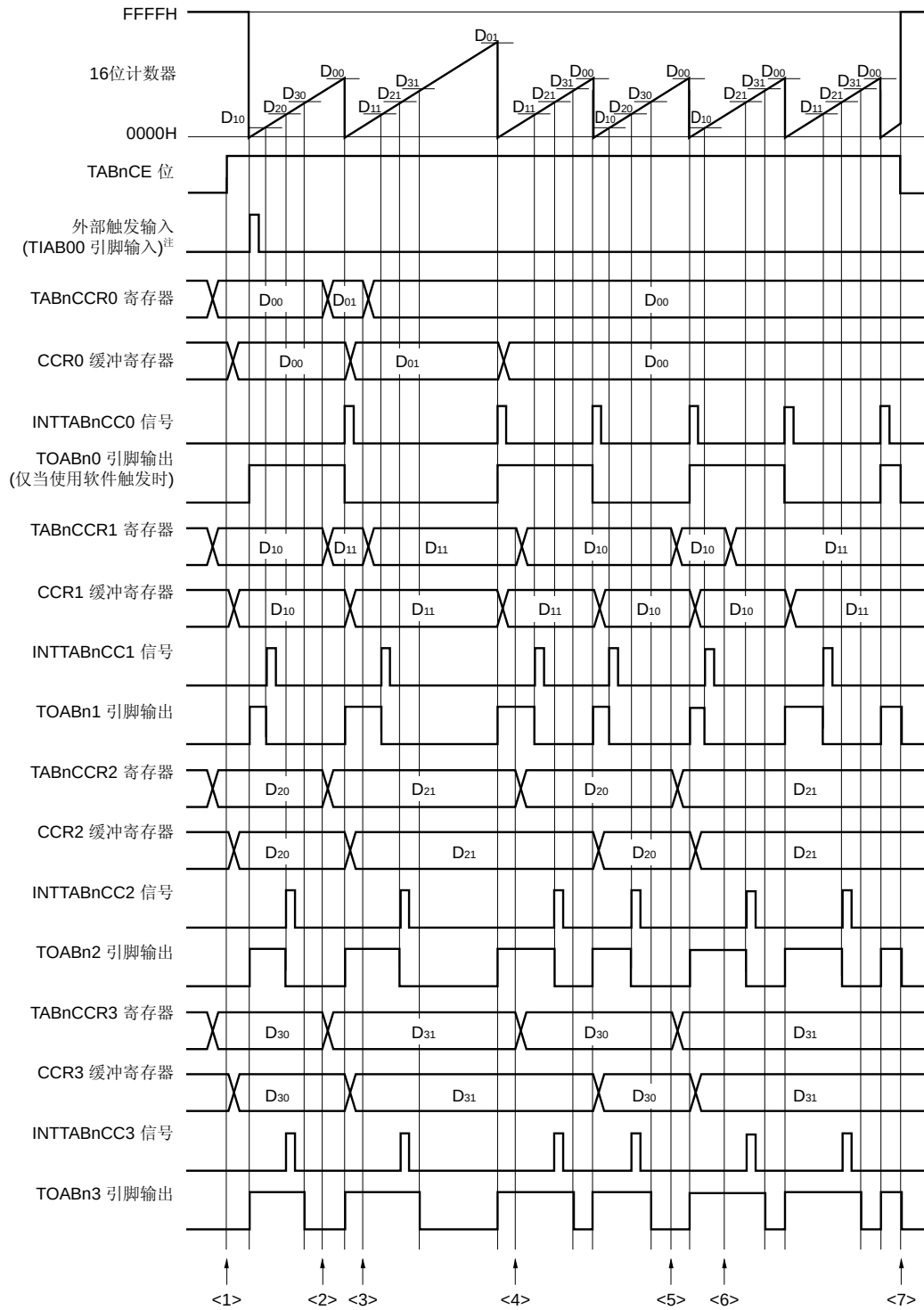
TOABn3 引脚脉宽调制波形有效电平宽度 = $D_3 \times$ 计数时钟周期

备注

1. 在外部触发脉冲输出模式下，不使用 TABn 输入/输出控制寄存器 1 (TABnIOC1) 和 TABn 选项寄存器 0 (TABnOPT0)。
2. 通过写入 TABn 捕获/比较寄存器 1 (TABnCCR1) 允许更新 TABn 捕获/比较寄存器 2 (TABnCCR2) 和 TABn 捕获/比较寄存器 3 (TABnCCR3)。
3. $n = 0, 1$

(1) 外部触发脉冲输出模式的工作流程

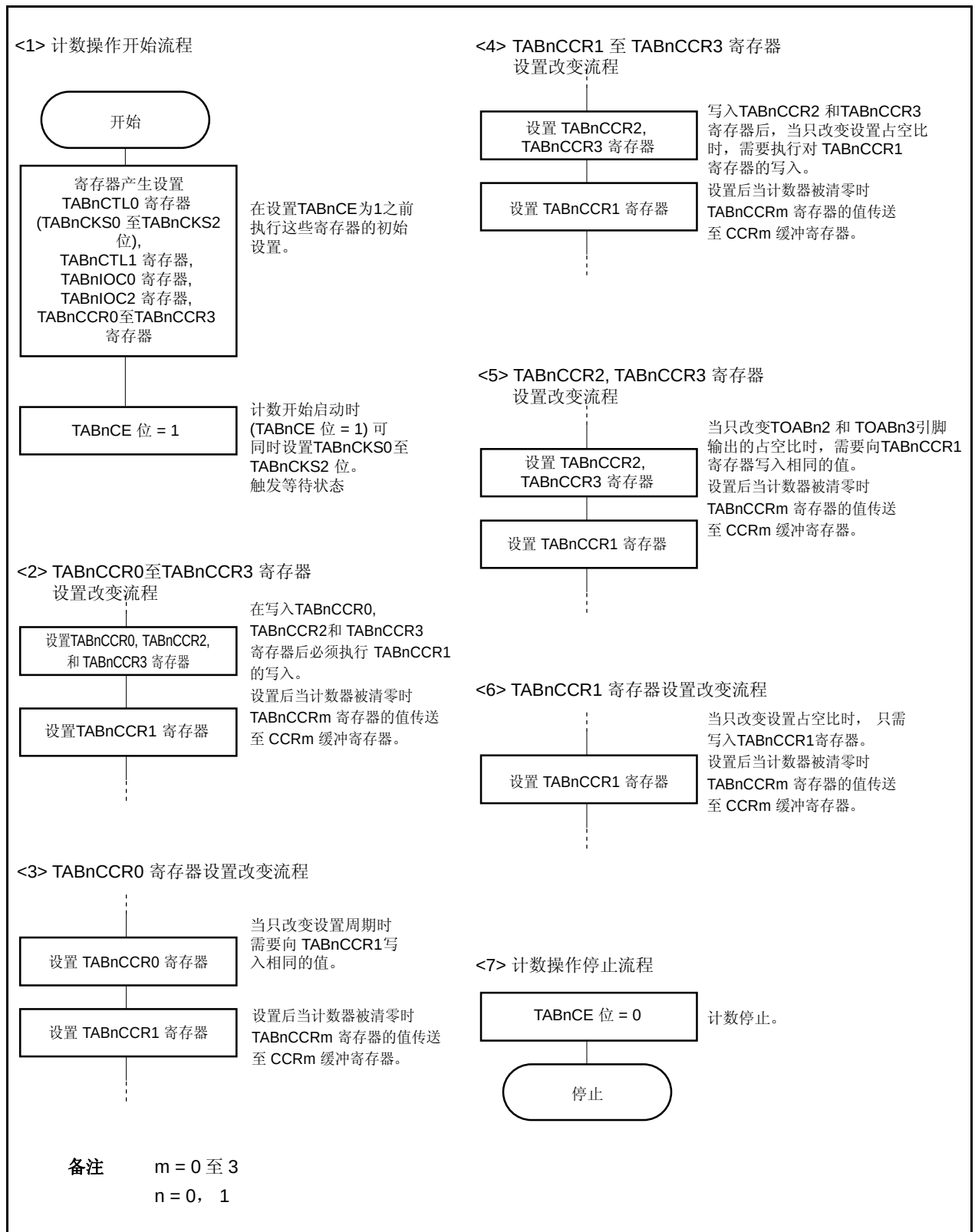
图 8-19. 外部触发脉冲输出模式的软件处理流程 (1/2)



注 TAB1: TRGAB1 引脚

备注 $n = 0, 1$

图 8-19. 外部触发脉冲输出模式的软件处理流程 (2/2)

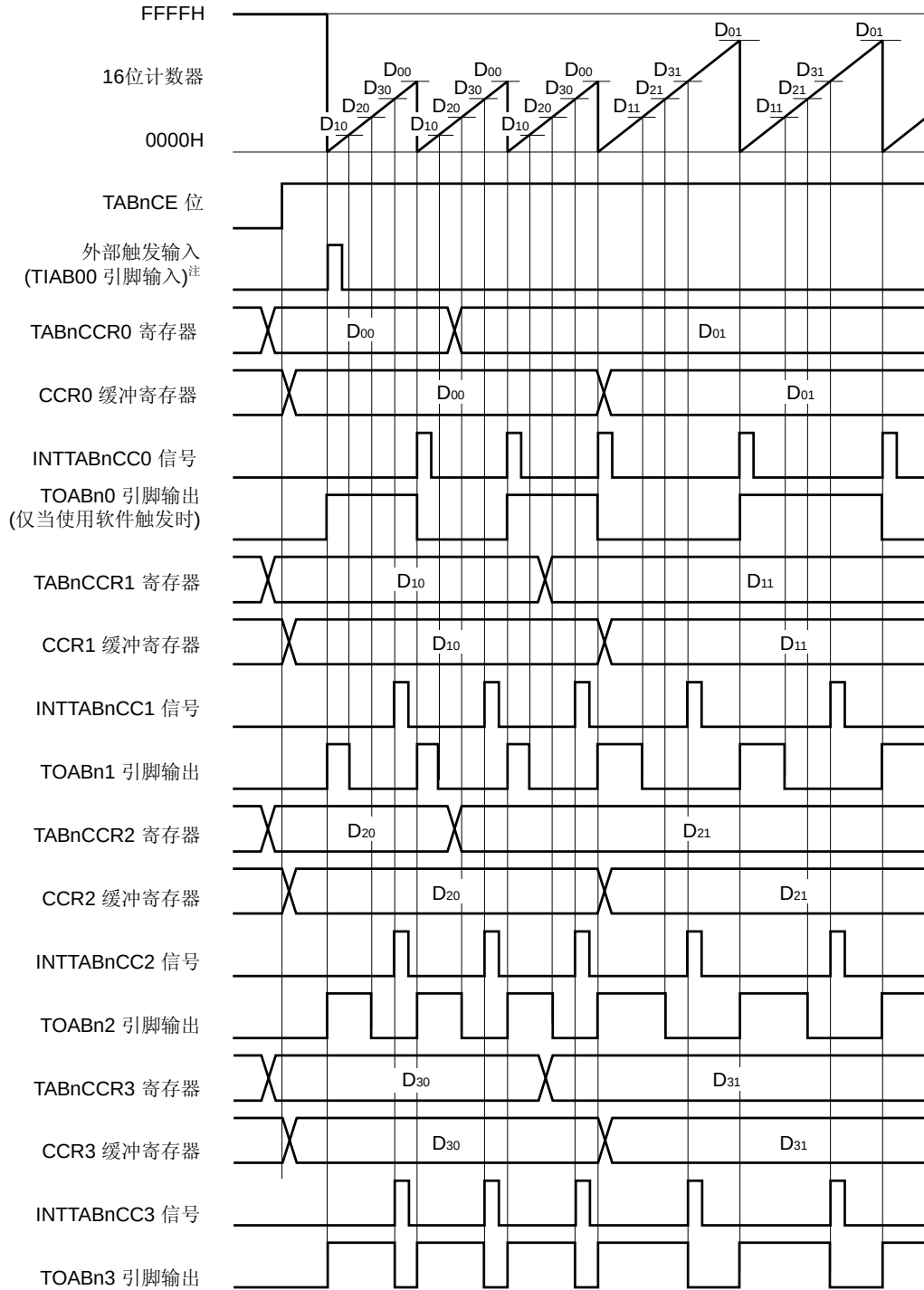


(2) 外部触发脉冲输出模式的操作时序

(a) 有关操作中改变脉冲宽度的注意事项

计数器计数过程中，若要改变脉宽调制信号波形，应最后写入 TABnCCR1 寄存器。

检测到 INTTABnCC0 信号以后，在写入 TABnCCR1 寄存器之后重写 TABnCCRk 寄存器。



注 TAB1: TRGAB1 引脚

备注 n = 0, 1

若要将数据从 TABnCCRM 寄存器传送到 CCRm 缓冲寄存器，则必须写入 TABnCCR1 寄存器。

要同时改变脉宽调制信号波形的周期和有效电平宽度，应首先将周期设定给 TABnCCR0 寄存器，将有效电平宽度设定给 TABnCCR2 和 TABnCCR3 寄存器，然后将有效电平设定给 TABnCCR1 寄存器。

如果只改变脉宽调制信号波形的周期，应首先把周期设置给 TABnCCR0 寄存器，然后将相同值（原值）写入 TABnCCR1 寄存器。

只改变脉宽调制信号波形的有效电平宽度（占空比系数）时，首先需要向 TABnCCR2 和 TABnCCR3 寄存器设置有效电平值，然后再向 TABnCCR1 寄存器设置有效电平。

若只改变通过 TOABn1 引脚输出的脉宽调制信号波形的有效电平宽度（占空比系数）时，只需要设置 TABnCCR1 寄存器。

若只改变通过 TOABn2 和 TOABn3 引脚输出的脉宽调制信号波形的有效电平宽度（占空比系数）时，首先要向 TABnCCR2 和 TABnCCR3 寄存器设置有效电平宽度，然后向 TABnCCR1 寄存器写入相同的值。

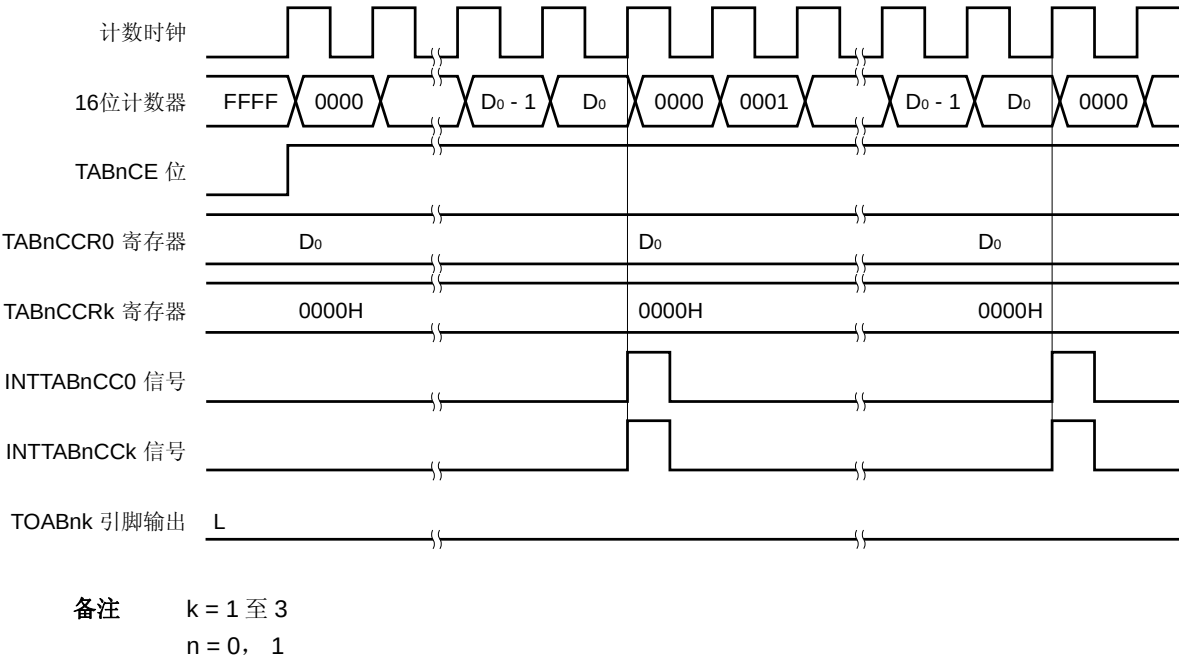
在数据写入 TABnCCR1 寄存器后，写入 TABnCCRM 寄存器的值会传送到 CCRm 缓冲寄存器，与 16 位计数器清零同步，并作为与 16 位计数器进行比较的值。

要在写入 TABnCCR1 寄存器一次之后，再次写 TABnCCR0 至 TABnCCR3 寄存器，应在 INTTABnCC0 信号产生之后进行。否则，CCRM 缓冲寄存器的值可能不确定，因为把数据从 TABnCCRM 寄存器传送到 CCRM 缓冲寄存器的时序与写 TABnCCRM 寄存器时序发生冲突。

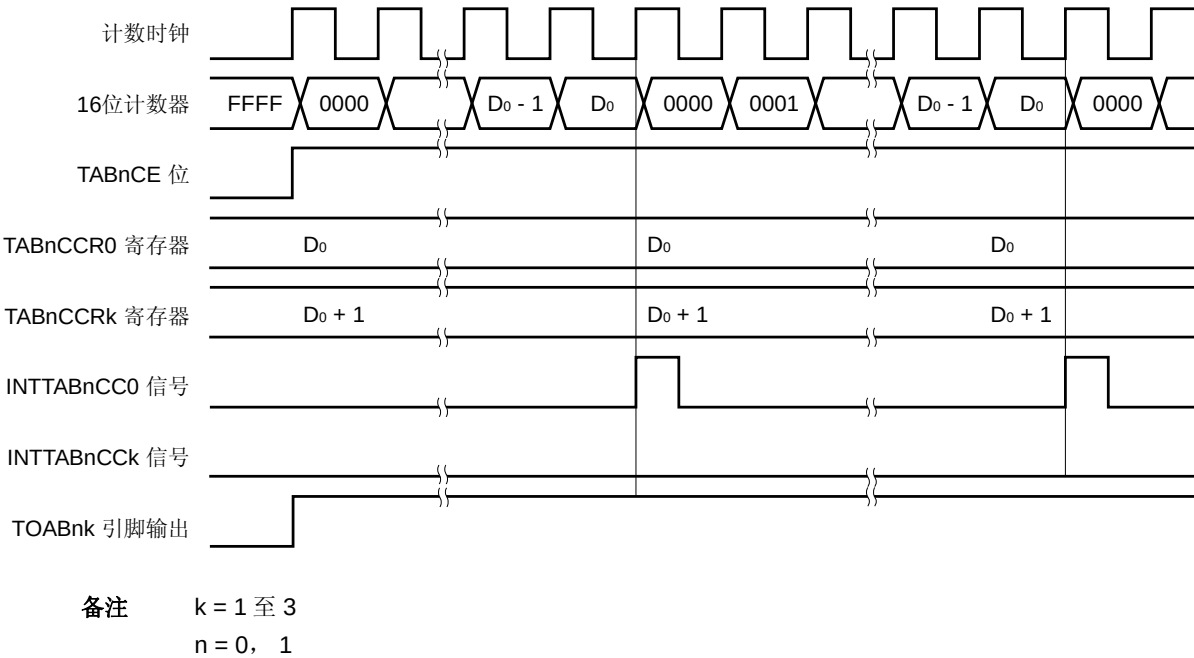
备注 m = 0 至 3
 n = 0, 1

(b) 脉宽调制信号波形的 0%和 100% 输出

为了输出 0% 的波形，需要将 TABnCCRk 寄存器设置为 0000H。如果 TABnCCR0 寄存器的设定值为 FFFFH，则会周期性产生 INTTABnCCk 中断请求信号。

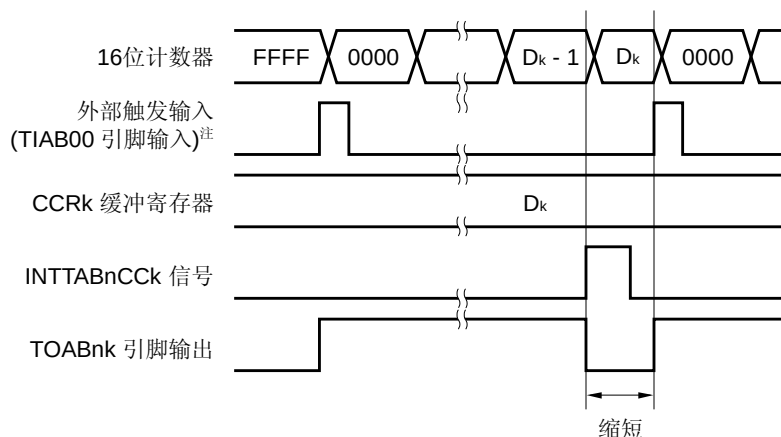


要输出 100%的波形，需给 TABnCCRk 寄存器设置一个“TABnCCR0 寄存器设定值 + 1”的数值。如果 TABnCCR0 寄存器的设定值为 FFFFH，则不会产生 100%波形输出。



(c) 触发检测与 CCRk 缓冲寄存器匹配之间的冲突

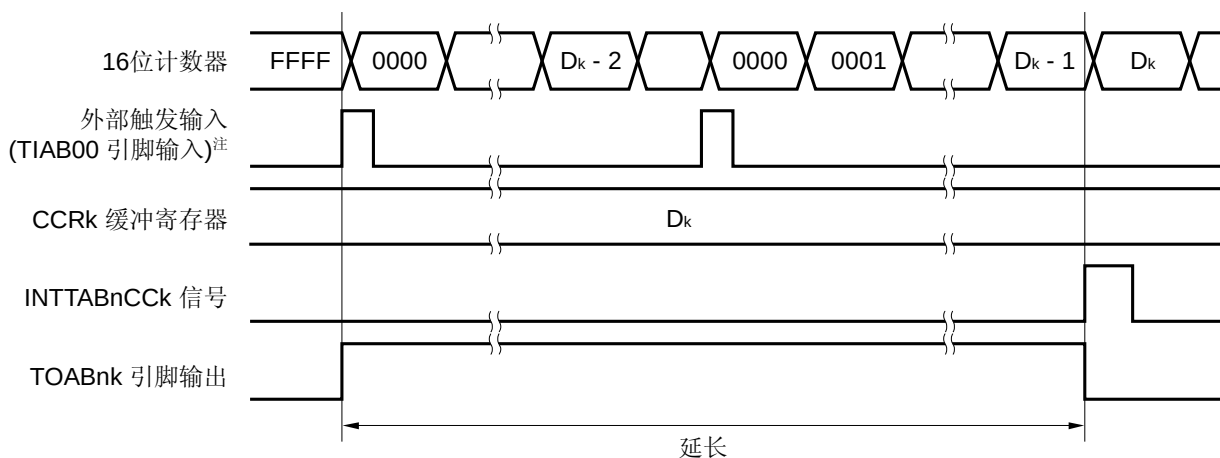
如果在紧接 INTTABnCCk 信号产生后，立刻检测到触发信号，则 16 位计数器立刻清零为 0000H，TOABnk 引脚的输出信号被置位生效，计数器继续计数。结果，缩短了脉宽调制信号波形的无效时段。



注 TAB1: TRGAB1 引脚

备注 $k = 1$ 至 3
 $n = 0, 1$

如果刚好在 INTTABnCCk 信号产生前检测到触发信号，则不会产生 INTTABnCCk 信号，而 16 位计数器清零为 0000H，计数继续进行。TOABnk 引脚的输出信号仍然保持有效。结果，延长了脉宽调制信号波形的有效时域。

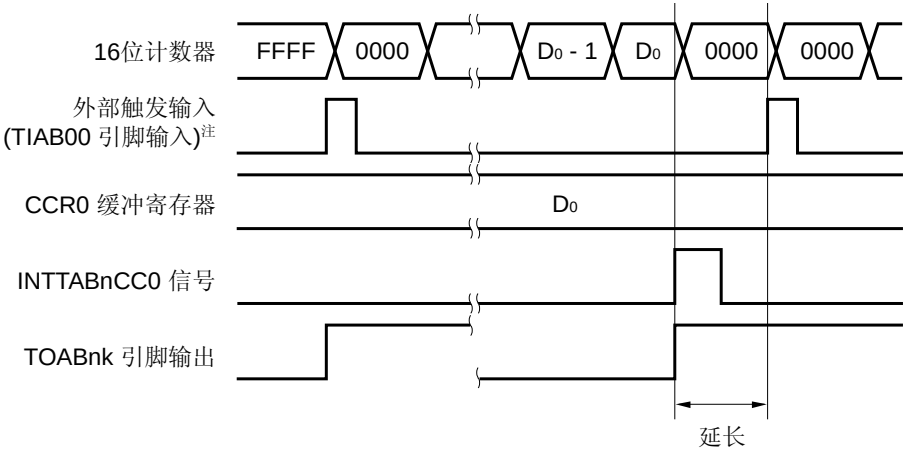


注 TAB1: TRGAB1 引脚

备注 $k = 1$ 至 3
 $n = 0, 1$

(d) 触发检测和与 CCR0 缓冲寄存器匹配之间的冲突

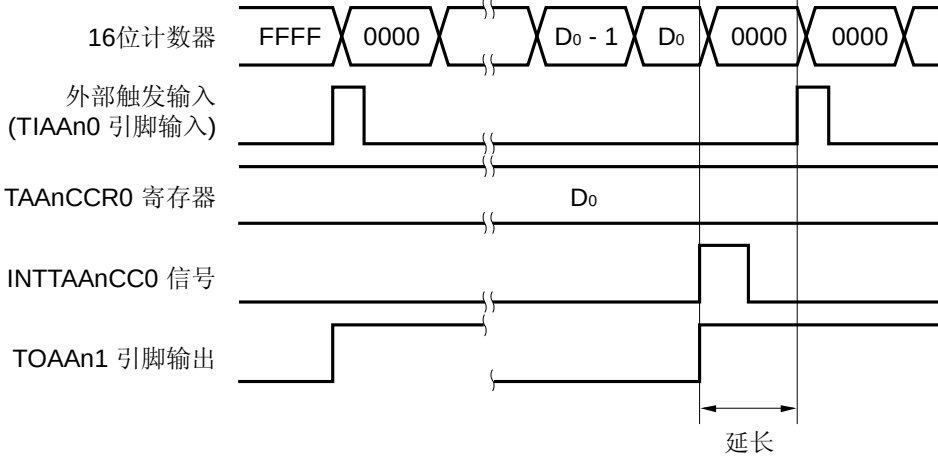
如果刚好在 INTTABnCC0 信号产生后检测到触发，则 16 位计数器清零为 0000H，并继续计数。因此，TOABnk 引脚的有效时段被延长，延长时间为自 INTTABnCC0 信号产生时刻到触发被检测到时刻。



注 TAB1: TRGAB1 引脚

备注 k = 1 至 3
n = 0, 1

如果刚好在产生 INTTABnCC0 信号之前检测到触发信号，则不会产生 INTTABnCC0 信号。16 位计数器清零为 0000H，TOABnk 引脚输出被置位生效，计数器继续计数。结果，缩短了脉宽调制信号波形的无效时段。

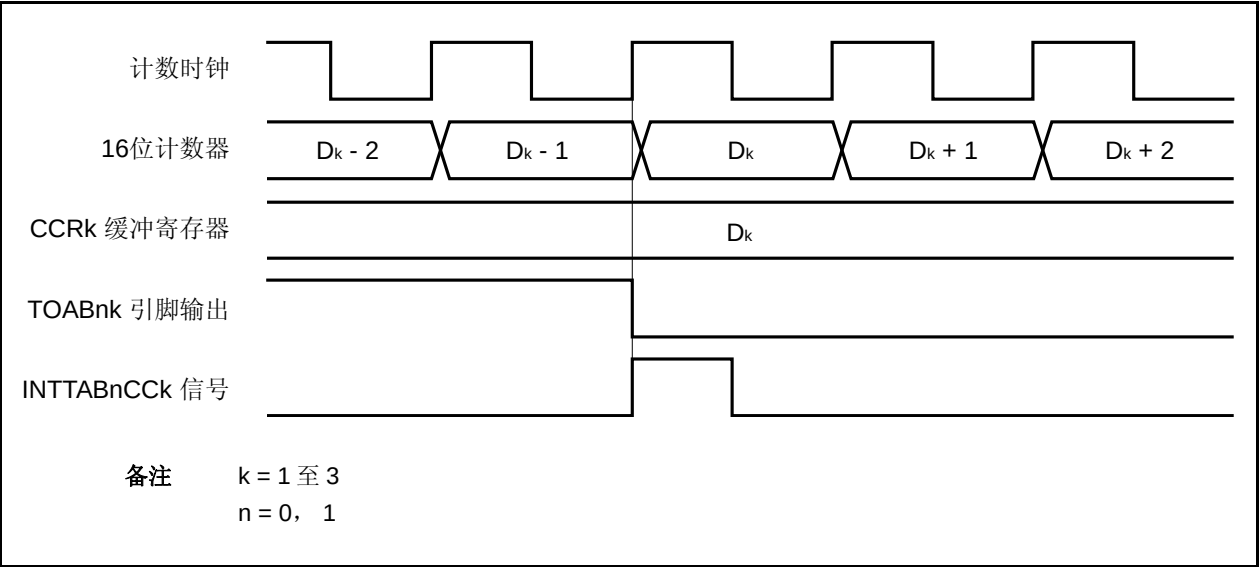


注 TAB1: TRGAB1 引脚

备注 k = 1 至 3
n = 0, 1

(e) 比较匹配中断请求信号 (INTTABnCCK) 的发生时序

INTTABnCCK 信号在外部触发脉冲输出模式下的发生时序与其它 INTTABnCCK 信号的时序不同；当 16 位计数器的计数值与 CCRk 缓冲寄存器的值匹配时，产生 INTTABnCCK 信号。



通常情况下，在 16 位计数器的当前计数值与 CCRk 缓冲寄存器的值匹配之后，INTTABnCCK 信号与下一次计数同步发生。

但是，在外部触发脉冲输出模式下，其产生会提早一个计数时钟周期。这是由于为了匹配 TOABnk 引脚输出信号的变化，时序发生了变化。

8.5.4 单次脉冲输出模式 (TABnMD2 至 TABnMD0 位 = 011)

在单次脉冲输出模式下, 当 TABnCTL0.TABnCE 位设置为 1 时 TABn 等待触发。当检测到外部触发输入信号的有效沿时, TABn 开始计数, 并从 TOABn1 至 TOABn3 引脚输出单次脉冲。

除外部触发外, 也可以生成软件触发来输出该脉冲。在使用软件触发时, TOABn0 引脚在 16 位计数器计数时输出有效电平, 在计数器停止 (等待触发) 时输出无效电平。

图 8-20. 单次脉冲输出模式的配置

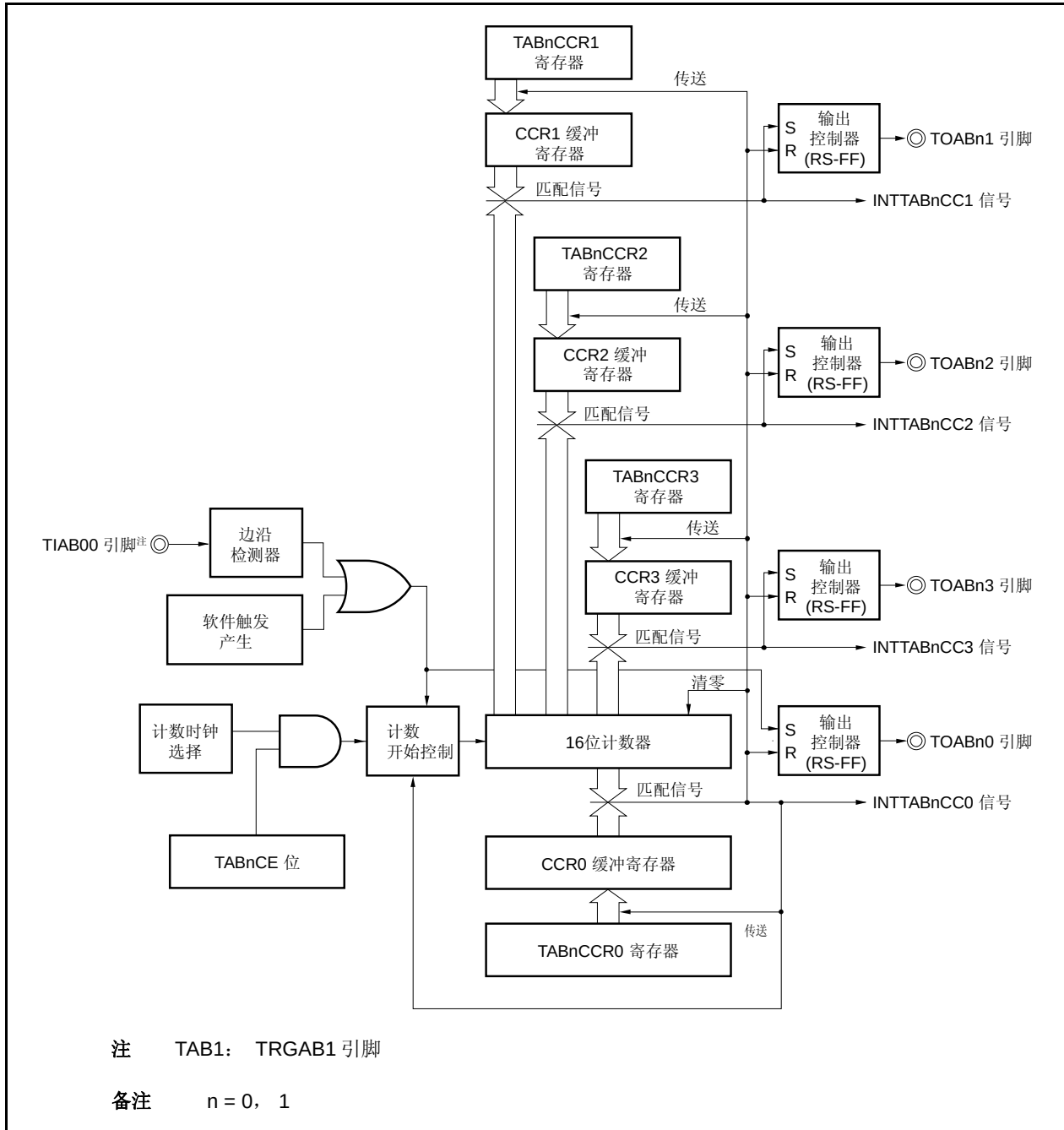
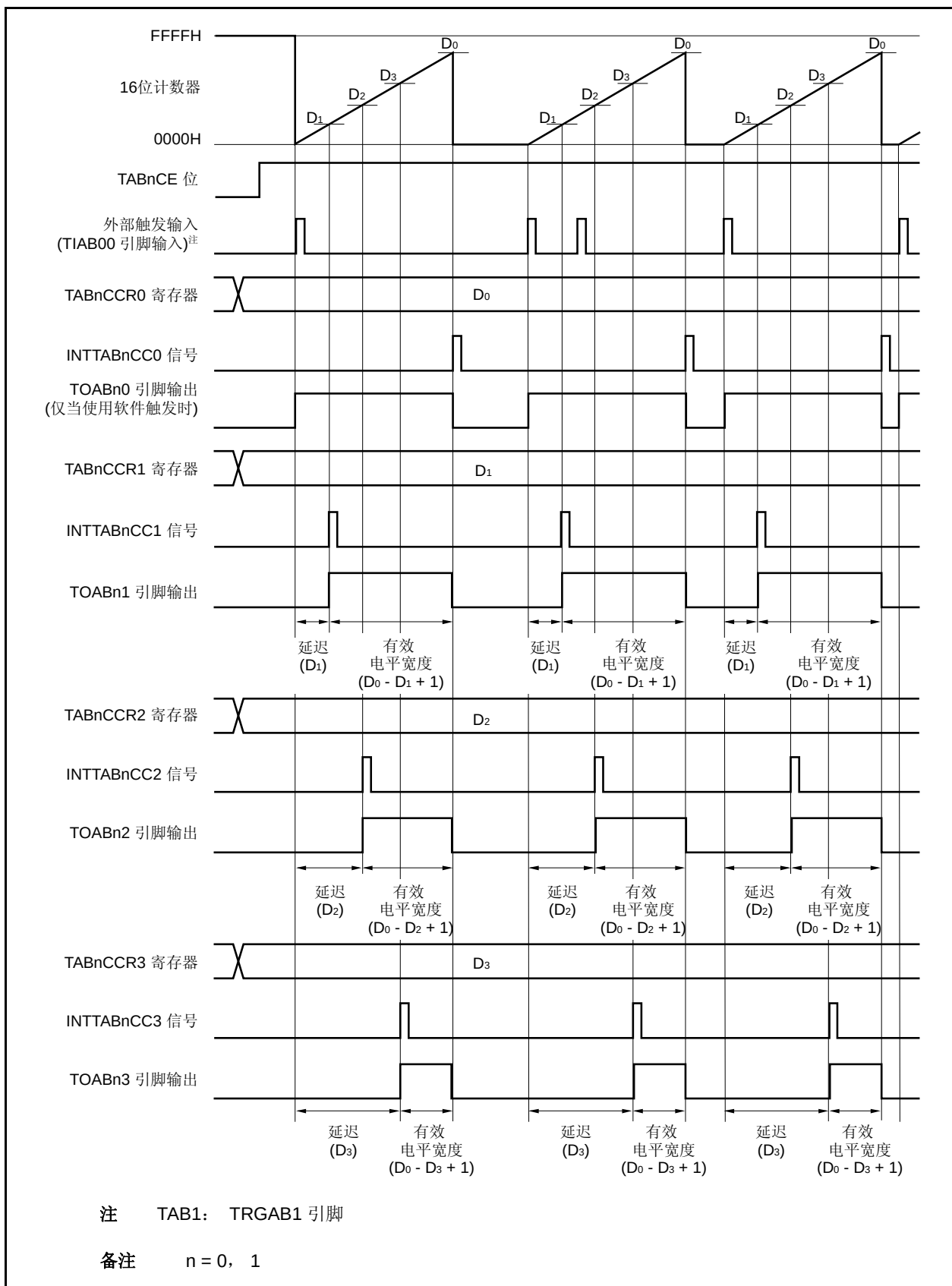


图 8-21. 单次脉冲输出模式的基本时序



当 TABnCE 位设置为 1 时，TABn 等待触发。在触发发生时，16 位计数器从 FFFFH 清零为 0000H，开始进行计数，TOABnk 引脚输出一个单次脉冲。在输出单次脉冲后，16 位计数器被设置为 FFFFH，计数停止，并重新等待触发。当 TOP01 引脚正在输出单次脉冲期间再次发生触发，则该触发被忽略。

单次脉冲的输出延迟时间和有效电平宽度可计算如下：

输出延迟时间 = (TABnCCRk 寄存器的设定值) × 计数时钟周期
有效电平宽度 = (TABnCCR0 寄存器的设定值 - TABnCCRk 寄存器的设定值 + 1) × 计数时钟周期

当 16 位计数器在其计数值与 CCR0 缓冲寄存器的值匹配之后进行下一个计数时，产生 INTTABnCC0 比较匹配中断请求信号。当 16 位计数器的计数值和 CCRk 缓冲寄存器的值匹配时，比较匹配中断请求信号 (INTTABnCk) 就产生了。

外部触发输入信号的有效沿和将软件触发 (TABnCTL1.TABnEST 位) 设置为 1，都可用作触发信号。

备注 k = 1 至 3
 n = 0, 1

图 8-22. 单次脉冲输出模式时的寄存器设置 (1/3)

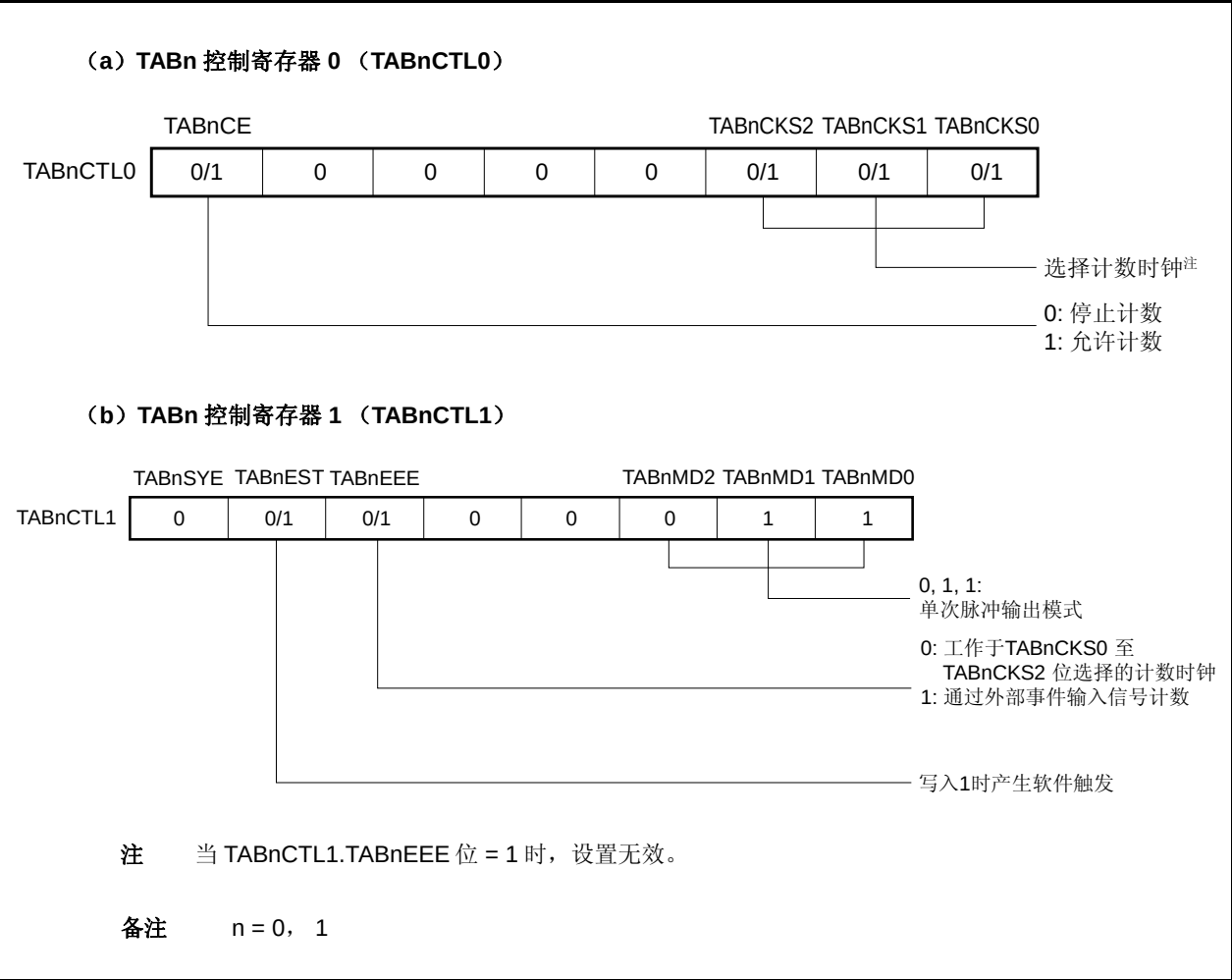


图 8-22. 单次脉冲输出模式时的寄存器设置 (2/3)

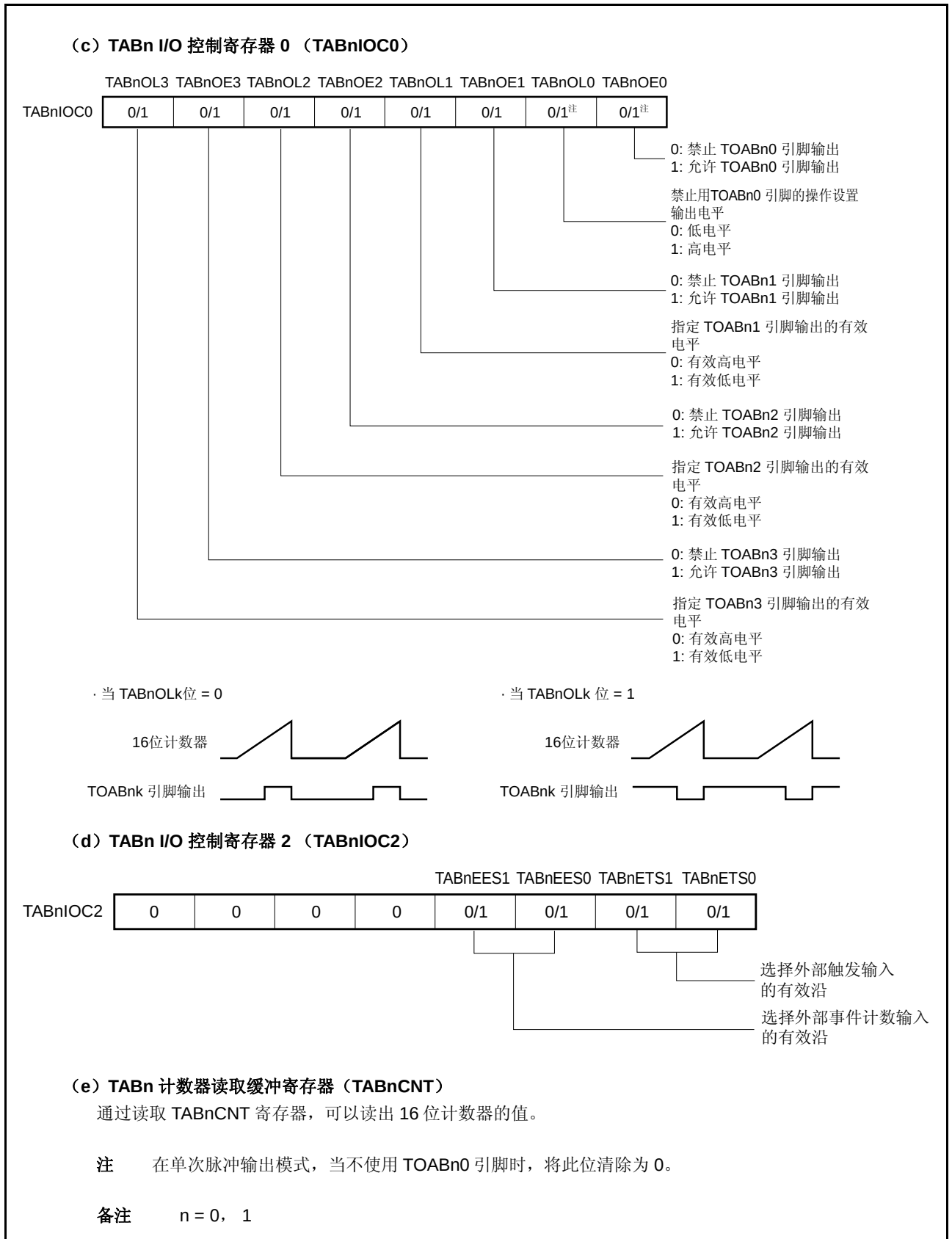


图 8-22. 单次脉冲输出模式时的寄存器设置 (3/3)

(f) TABn 捕获/比较寄存器 0 至 3 (TABnCCR0 至 TABnCCR3)

若把 TABnCCR0 寄存器的值设置为 D_0 , TABnCCRk 寄存器的值设置为 D_k , 则单次脉冲的有效电平宽度和输出延迟期计算如下。

有效电平宽度 = $(D_0 - D_k + 1) \times$ 计数时钟周期

输出延迟时间 = $(D_k) \times$ 计数时钟周期

注意事项 如果 TABnCCRk 寄存器中设置的值大于 TABnCCR0 寄存器中设置的值, 那么, 即使在单次脉冲输出模式下也不会输出单次脉冲。

备注

1. 在单次脉冲输出模式下, 不使用 TABn 输入/输出控制寄存器 1 (TABnIOC1) 和 TABn 选项寄存器 0 (TABnOPT0)。
2. $k = 1$ 至 3
 $n = 0, 1$

(1) 单次脉冲输出模式下的操作流程

图 8-23. 单次脉冲输出模式的软件处理流程 (1/2)

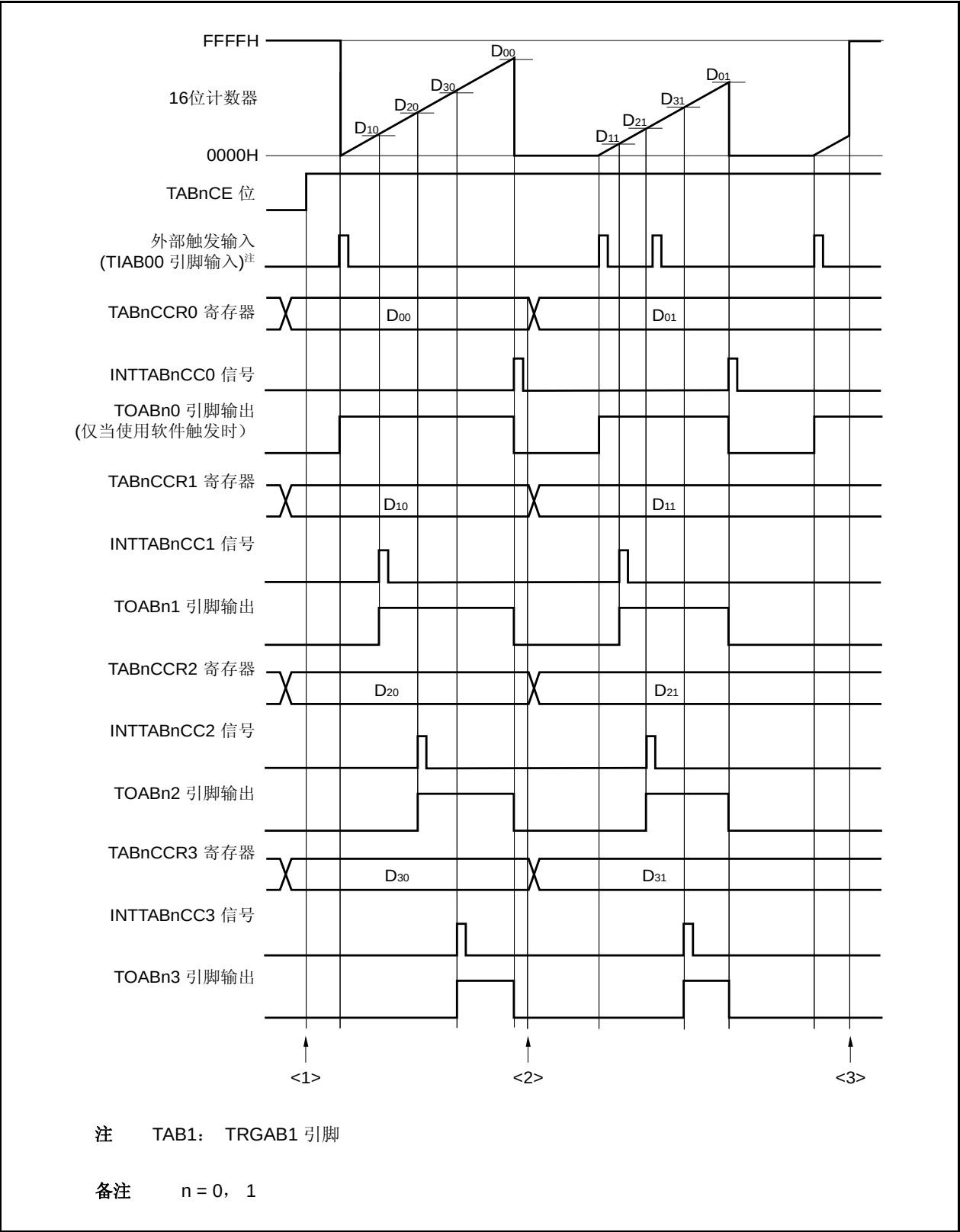
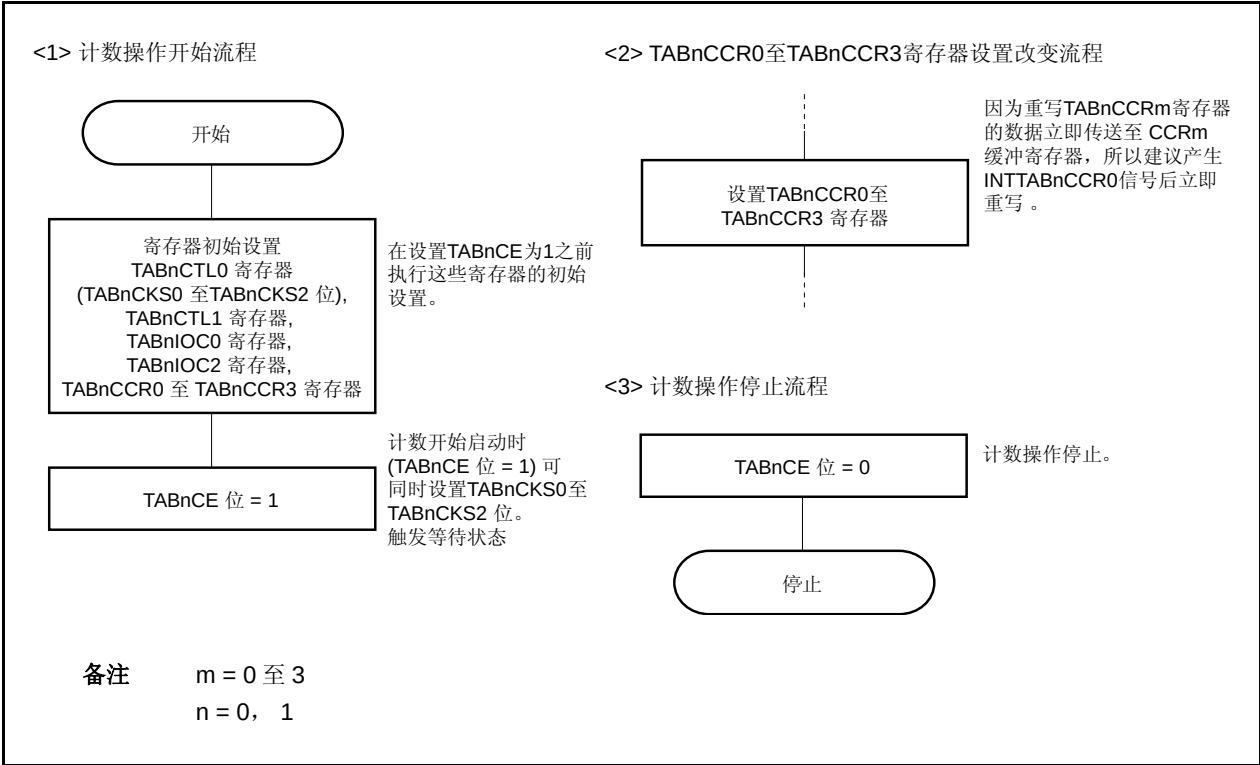


图 8-23. 单次脉冲输出模式的软件处理流程 (2/2)

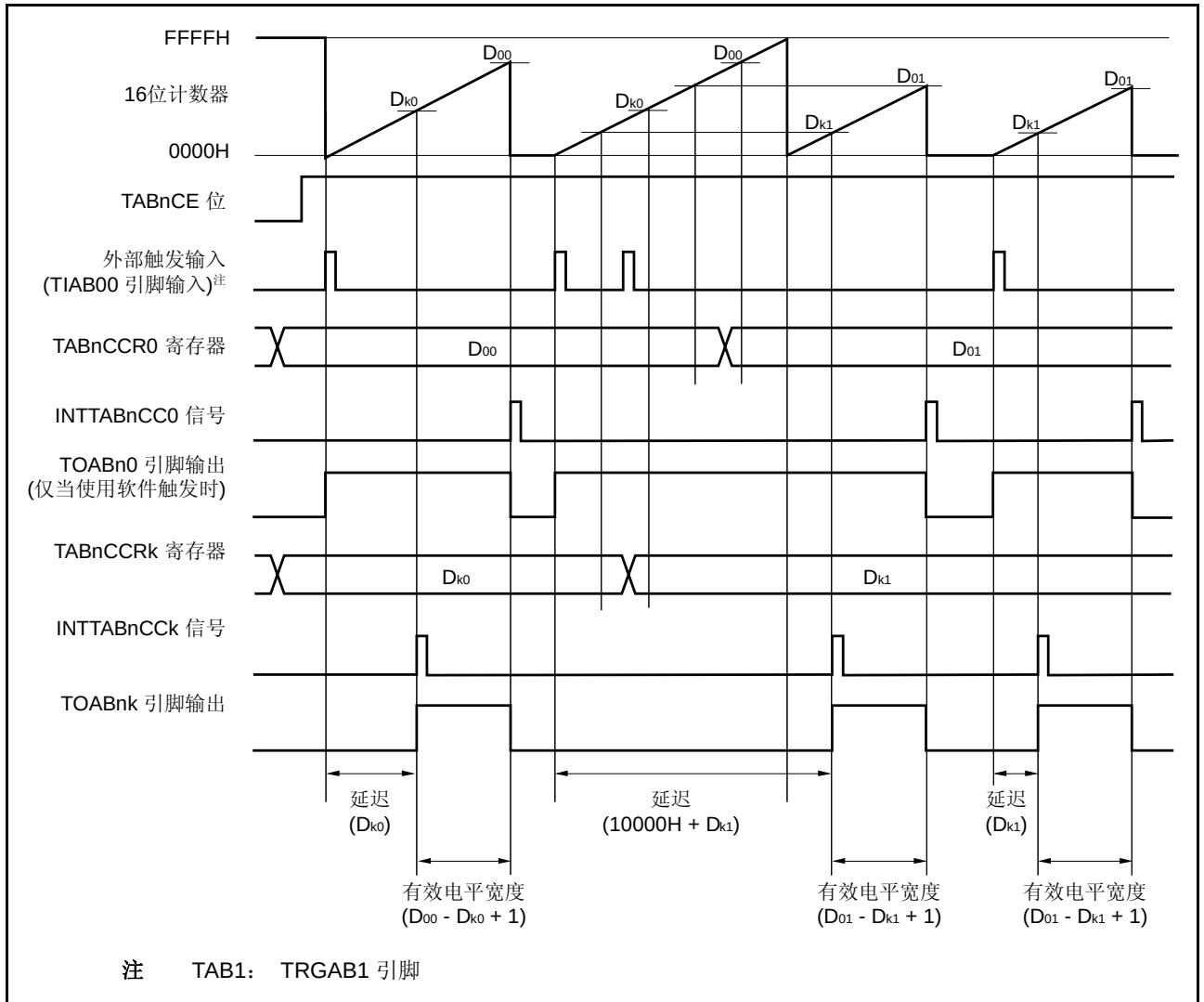


(2) 单次脉冲输出模式时的工作时序

(a) 关于重写 TABnCCRM 寄存器的注意事项

要将 TABnCCRM 寄存器的值改为较小的值, 需要停止计数, 而后改变设定值。

如果在计数器工作期间, 将 TABnCCR0 寄存器的值重写为比当前值较小的值, 16 位计数器会发生溢出。



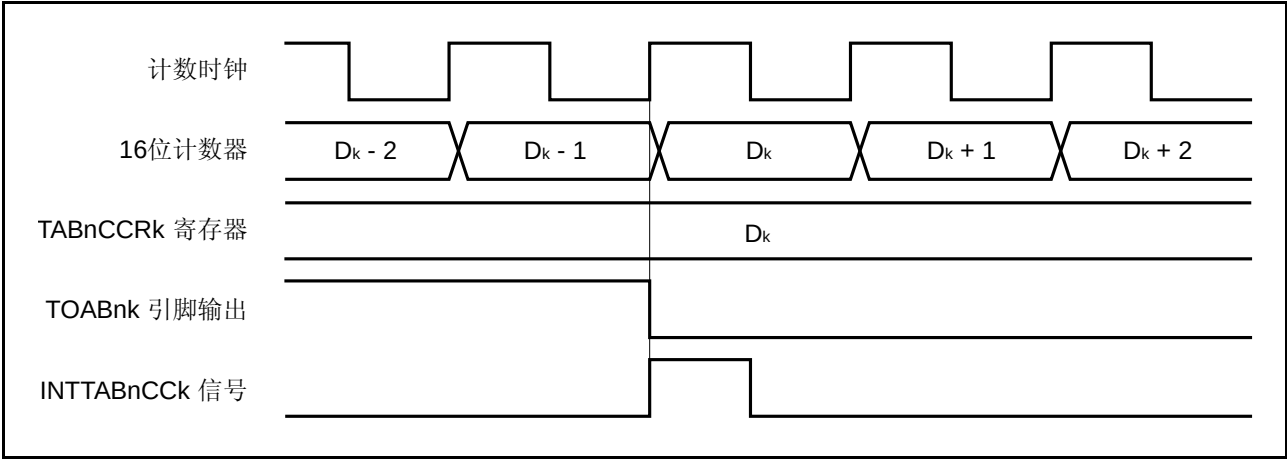
如果 $D_{00} > D_{01}$ 而且 $D_{k0} > D_k$, 当 TABnCCR0 寄存器从 D_{00} 重写为 D_{01} , TABnCCRk 寄存器从 D_{k0} 重写为 D_{k1} 时, 若 TABnCCRk 寄存器在 16 位计数器的计数值大于 D_{k1} 且小于 D_{k0} 时重写, 以及 TABnCCR0 寄存器在计数值大于 D_{01} 且小于 D_{00} 时重写, 则一旦重写寄存器, 就会在各设置值上反映出来并立即与计数值进行比较。计数器向上计数到 FFFFH 后, 从 0000H 开始重新向上计数。当计数值与 D_{k1} 匹配时, 计数器会产生 INTTABnCCk 信号并使 TOABnk 引脚置位有效。当计数值与 D_{01} 匹配时, 计数器产生 INTTABnCC0 信号, 使 TOABnk 引脚清除置位, 并停止计数。

因此, 计数器输出脉冲的延迟期或有效期可能会与原先预计的单次脉冲有所不同。

备注 k = 1 至 3
n = 0, 1

(b) 比较匹配中断请求信号 (INTTABnCCK) 的发生时序

单次脉冲输出模式下，比较匹配中断请求信号 (INTTABnCCK) 的发生时序和其它情况下的 INTTABnCCK 信号不同，当 16 位计数器的计数值与 TABnCCRk 寄存器的值匹配时，就会产生 TABnCCRk 中断请求信号。



通常情况下，在 16 位计数器的当前计数值与 TABnCCRk 寄存器的值匹配之后，INTTABnCCK 信号与下一次计数同步发生。

但是，在单次脉冲输出模式下，INTTP0CC1 信号会提早一个计数时钟周期产生。这是由于为了匹配 TOABnk 引脚输出信号的变化，时序发生了变化。

备注 $k = 1$ 至 3
 $n = 0, 1$

8.5.5 PWM输出模式 (TABnMD2 至TABnMD0 位= 100)

在 PWM 输出模式下, 当 TABnCTL0.TABnCE 位被设定为 1 时, PWM 波形从 TOABn1 至 TOABn3 引脚输出。
而且, 由 TOABn0 引脚输出的脉冲, 其半周期等于 PWM 的周期。

图 8-24. PWM 输出模式的配置

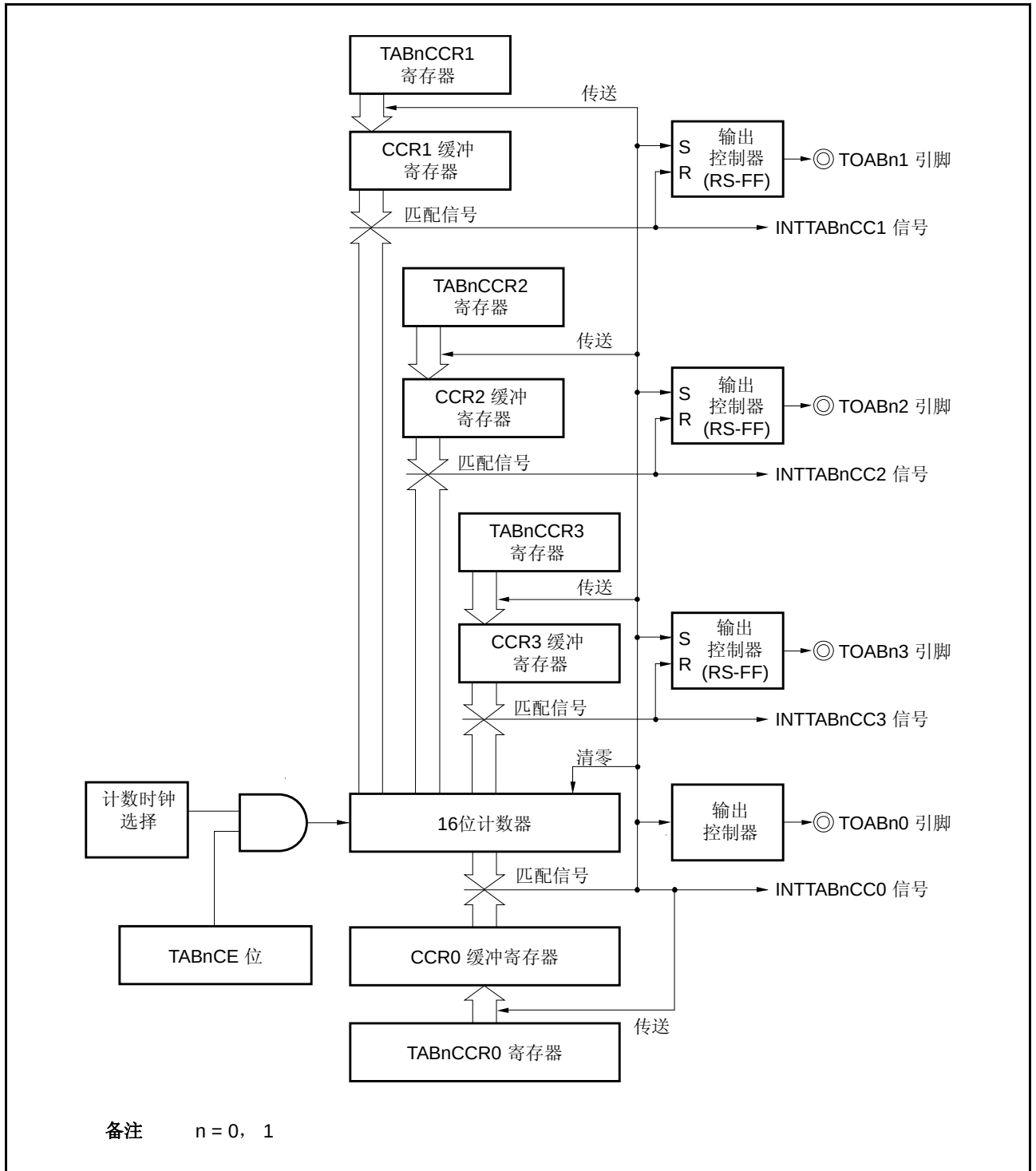
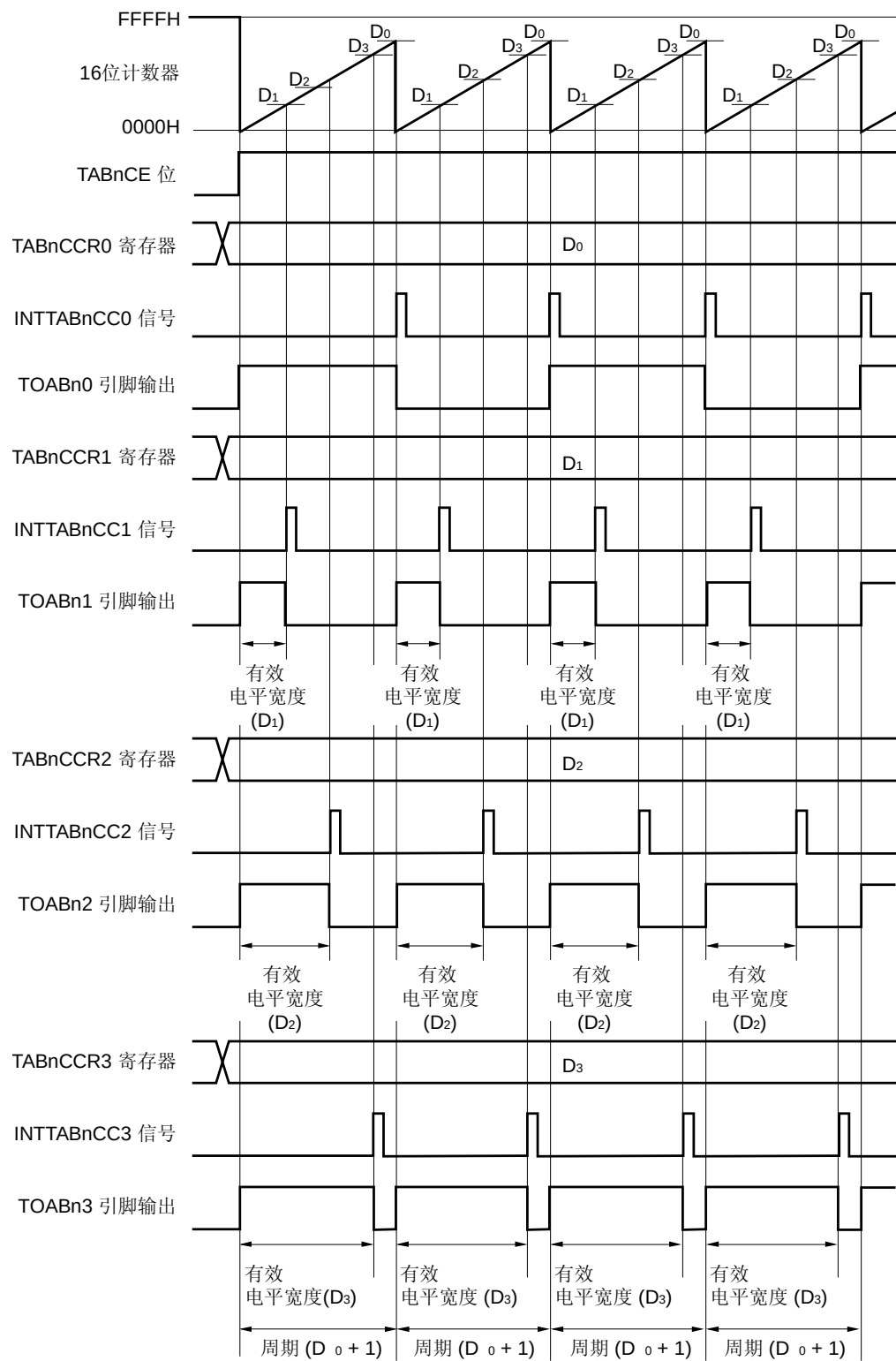


图 8-25. PWM 输出模式的基本时序



备注 n = 0, 1

当 TABnCE 位设置为 1 时，16 位计数器从 FFFFH 清零为 0000H，开始进行计数，同时从 TOABnk 引脚输出脉冲宽度调制信号波形。

脉冲宽度调制信号波形的有效电平宽度、周期、占空比计算如下。

有效电平宽度 = (TABnCCRk 寄存器的设定值) × 计数时钟周期

周期 = (TABnCCR0 寄存器的设定值 + 1) × 计数时钟周期

占空比 = (TABnCCRk 寄存器的设定值) / (TABnCCR0 寄存器的设定值 + 1)

在进行计数时，可通过重写 TABnCCRM 寄存器来改变脉冲宽度调制信号波形。当 16 位计数器的计数值与 CCR0 缓冲寄存器的值匹配时，16 位计数器清零为 0000H 时，新写入的值会得到反映。

当 16 位计数器在其计数值与 CCR0 缓冲寄存器的值匹配之后进行下一次向上计数时，会同步产生比较匹配中断请求信号 (INTTABnCC0)，且 16 位计数器清零为 0000H。当 16 位计数器的计数值和 CCRk 缓冲寄存器的值匹配时，比较匹配中断请求信号 (INTTABnCCK) 就产生了。

备注 k = 1 至 3
m = 0 至 3
n = 0, 1

图 8-26. PWM 输出模式时的寄存器设置 (1/3)

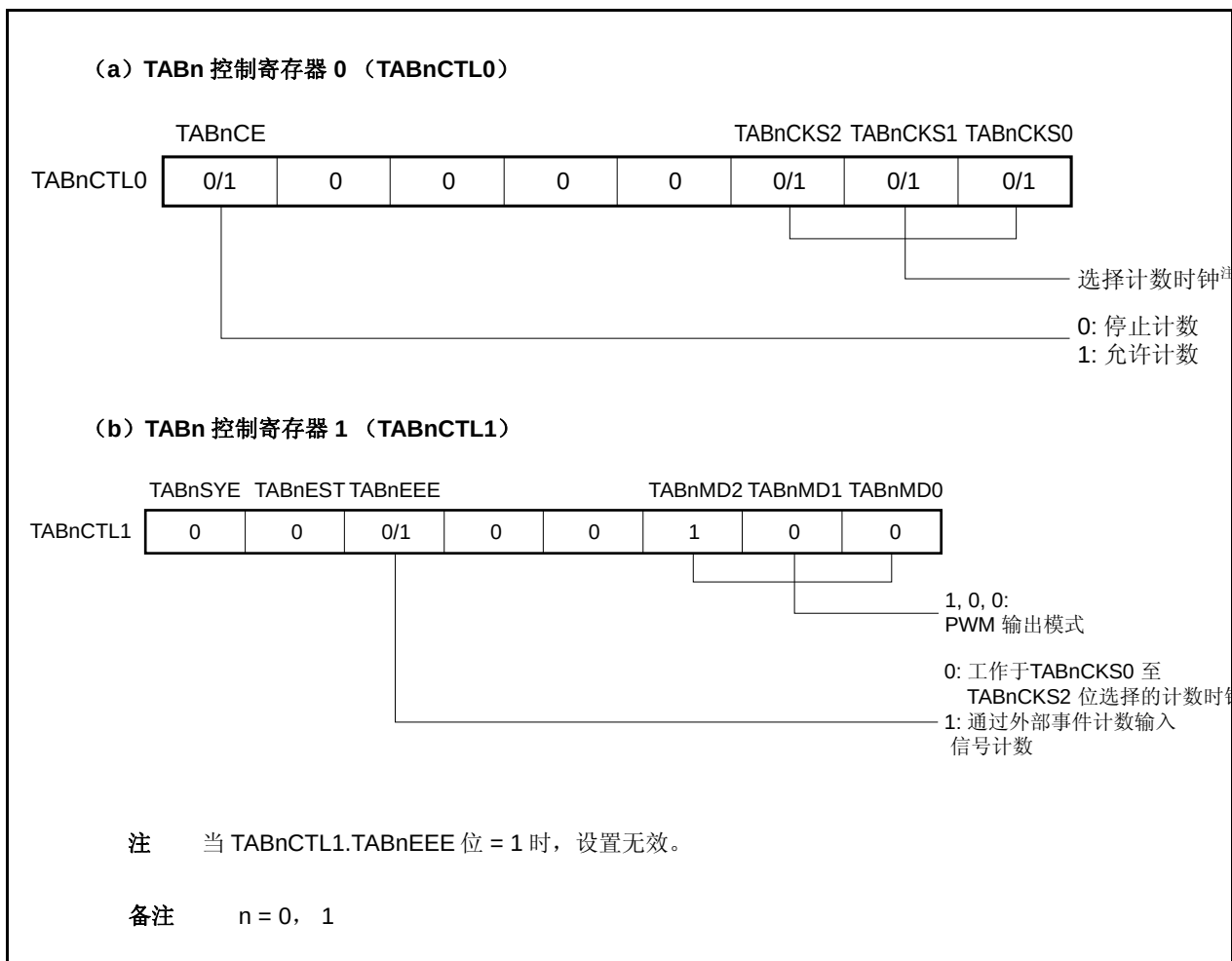
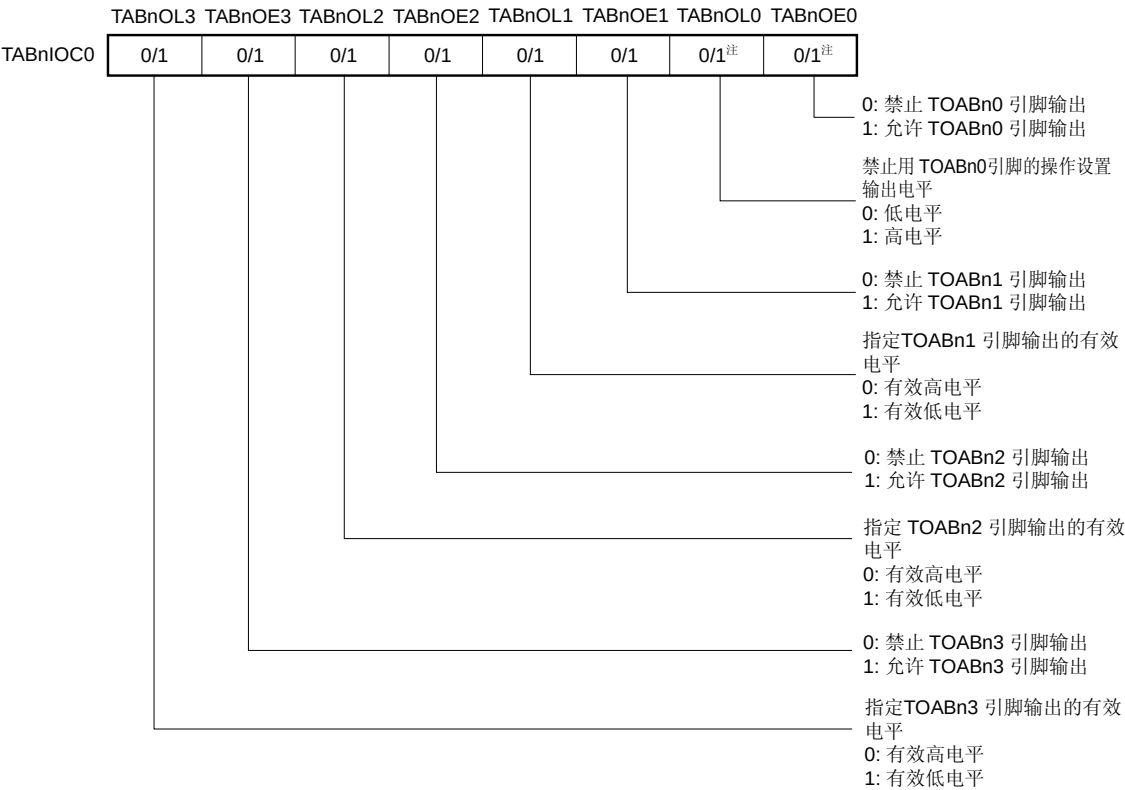
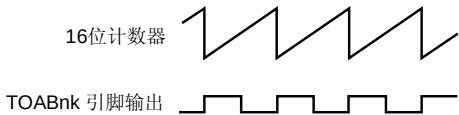


图 8-26. PWM 输出模式时的寄存器设置 (2/3)

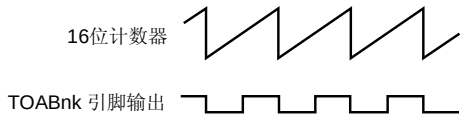
(c) TABn I/O 控制寄存器 0 (TABnIOC0)



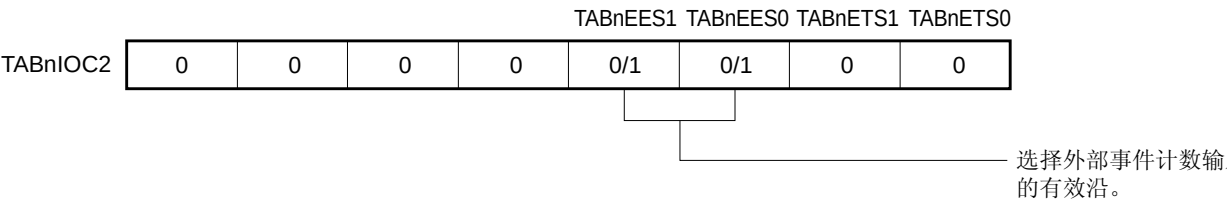
· 当 TABnOLk 位 = 0



· 当 TABnOLk 位 = 1



(d) TABn I/O 控制寄存器 2 (TABnIOC2)



(e) TABn 计数器读取缓冲寄存器 (TABnCNT)

通过读取 TABnCNT 寄存器，可以读出 16 位计数器的值。

注 在 PWM 输出模式，当不使用 TOABn0 引脚时，将此位清除为 0。

备注 n = 0, 1

图 8-26. PWM 输出模式时的寄存器设置 (3/3)

(f) TABn 捕获/比较寄存器 0 至 3 (TABnCCR0 至 TABnCCR3)

如果将 TABnCCR0 寄存器的值设置为 D_0 ，将 TABnCCRk 寄存器的值设置为 D_k ，则 PWM 波形的周期和有效电平如下计算：

周期 = $(D_0 + 1) \times$ 计数时钟周期

有效电平宽度 = $D_k \times$ 计数时钟周期

备注

1. 在 PWM 输出模式下，不使用 TABn 输入/输出控制寄存器 1 (TABnIOC1) 和 TABn 选项寄存器 0 (TABnOPT0)。
2. 通过写入 TABn 捕获/比较寄存器 1 (TABnCCR1) 允许更新 TABn 捕获/比较寄存器 2 (TABnCCR2) 和 TABn 捕获/比较寄存器 3 (TABnCCR3)。
3. $n = 0, 1$

(1) PWM 输出模式下的工作流程

图 8-27. PWM 输出模式时的软件处理流程 (1/2)

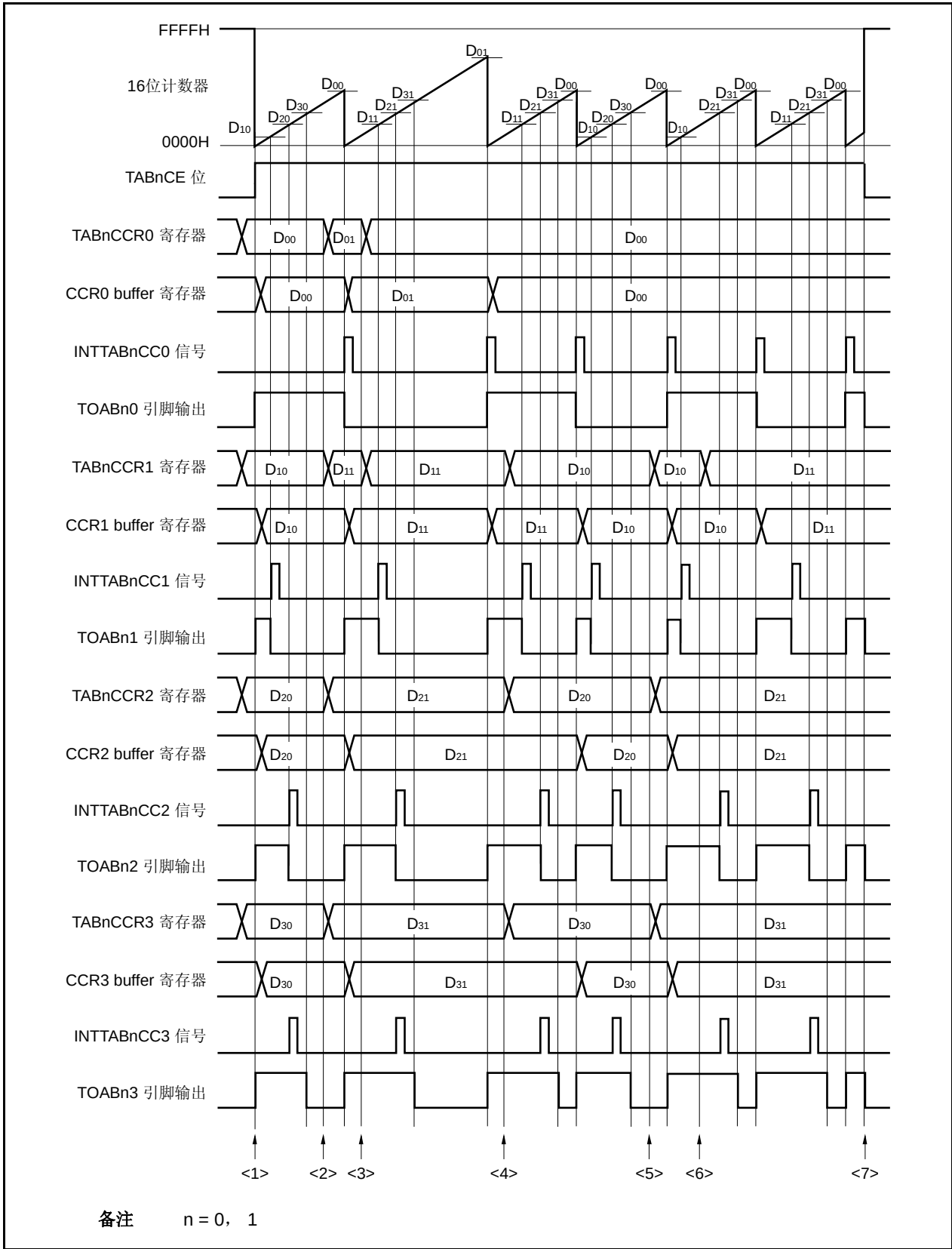
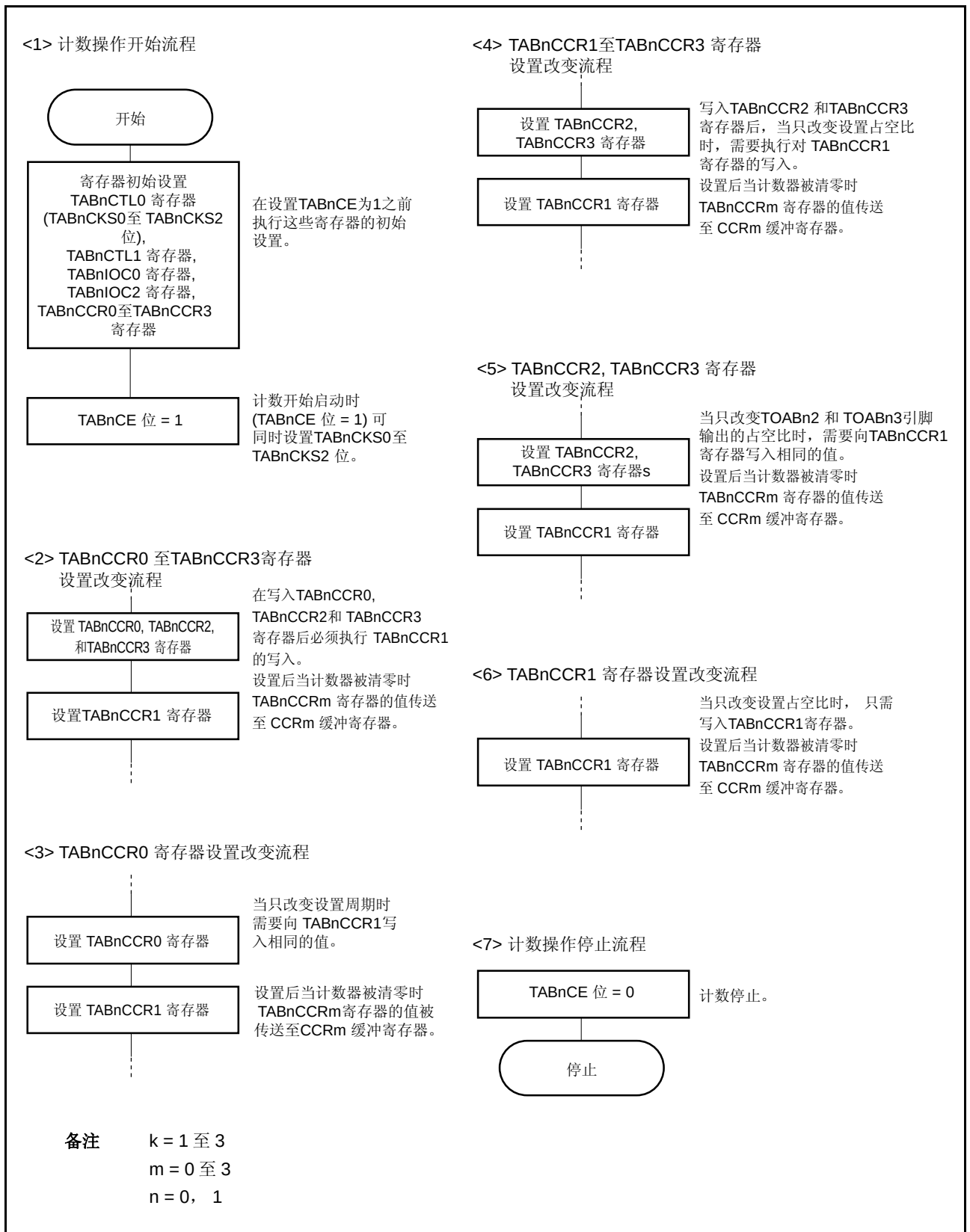


图 8-27. PWM 输出模式时的软件处理流程 (2/2)

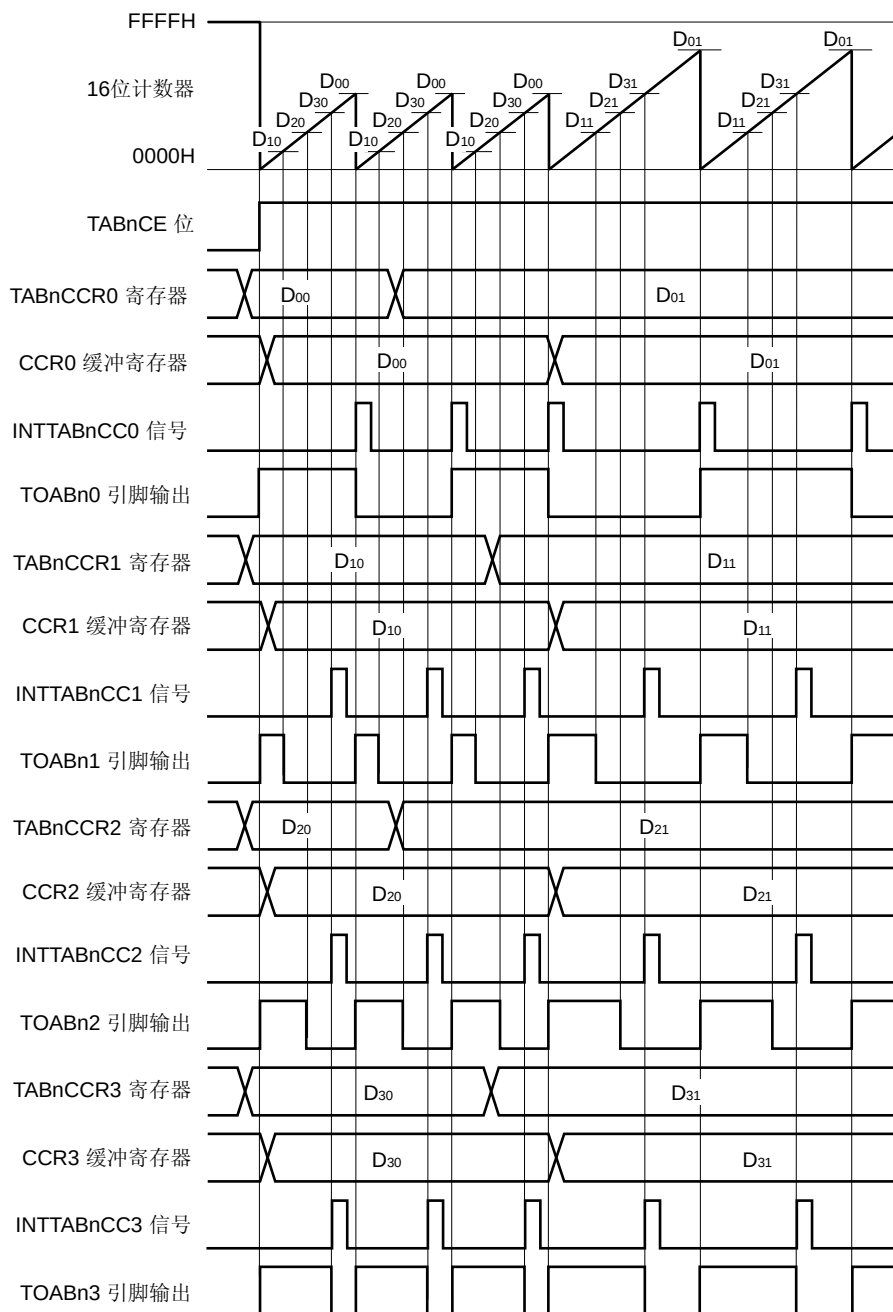


(2) PWM 输出模式的操作时序

(a) 操作过程中改变脉冲宽度

计数器计数过程中，若要改变 PWM 波形，应最后写入 TABnCCR1 寄存器。

检测到 INTTABnCC1 信号写入 TABnCCR1 寄存器后对 TABnCCRk 寄存器进行重写。



备注 n = 0, 1

若要将数据从 TABnCCRm 寄存器传送到 CCRm 缓冲寄存器，则必须写入 TABnCCR1 寄存器。

要同时改变 PWM 波形的周期和有效电平，应首先将周期设定给 TABnCCR0 寄存器，将有效电平宽度设定给 TABnCCR2 和 TABnCCR3 寄存器，然后将有效电平宽度设定给 TABnCCR1 寄存器。

如果只改变 PWM 波形的周期，应首先把周期设置给 TABnCCR0 寄存器，然后将相同值写入 TABnCCR1 寄存器。

只改变 PWM 波形的有效电平宽度（占空比系数）时，首先需要向 TABnCCR2 和 TABnCCR3 寄存器设置有效电平值，然后再向 TABnCCR1 寄存器设置有效电平。

若只改变通过 TOABn1 引脚输出的 PWM 波形的有效电平宽度（占空比系数）时，只需要设置 TABnCCR1 寄存器。

若只改变通过 TOABn2 和 TOABn3 引脚输出的 PWM 波形的有效电平宽度（占空比系数）时，首先要向 TABnCCR2 和 TABnCCR3 寄存器设置有效电平宽度，然后向 TABnCCR1 寄存器写入相同的值。

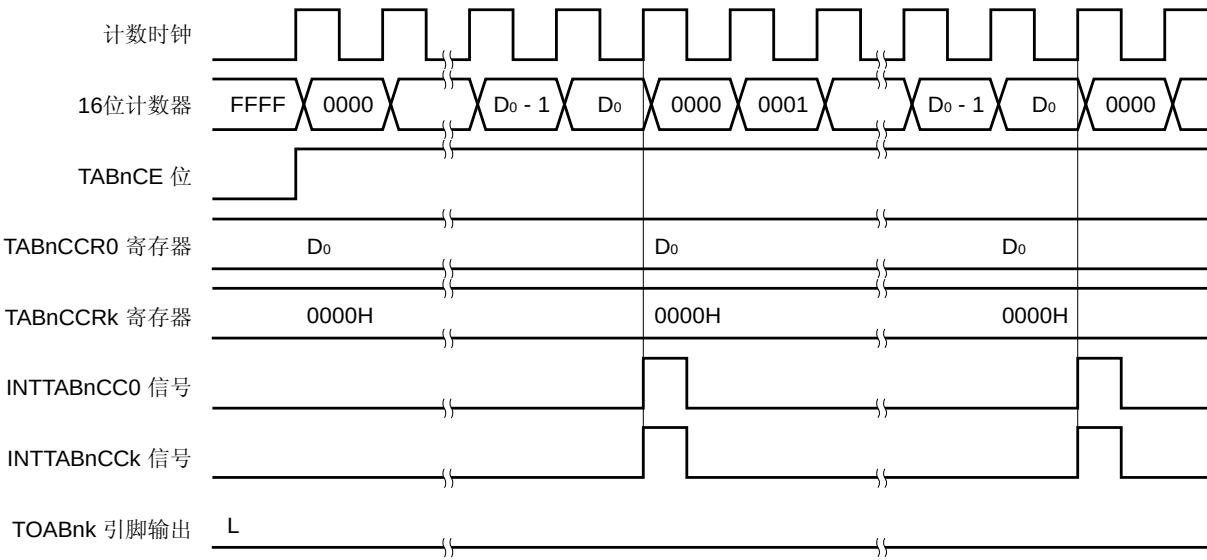
在数据写入 TABnCCR1 寄存器后，写入 TABnCCRm 寄存器的值会传送到 CCRm 缓冲寄存器中，与 16 位计数器清零同步，并作为与 16 位计数器进行比较的值。

要在写入 TABnCCR1 寄存器一次之后，再次写 TABnCCR0 至 TABnCCR3 寄存器，应在 INTTABnCC0 信号产生之后进行。否则，CCRM 缓冲寄存器的值可能不确定，因为把数据从 TABnCCRm 寄存器传送到 CCRm 缓冲寄存器的时序与写 TABnCCRm 寄存器时序发生冲突。

备注 m = 0 至 3
 n = 0, 1

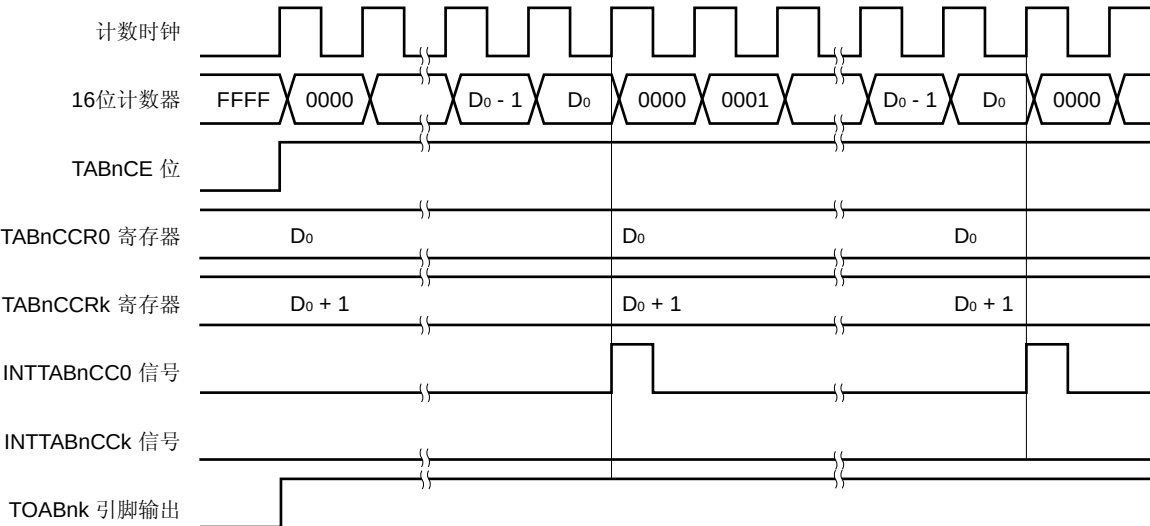
(b) PWM 波形的 0%和 100% 输出

为了输出 0% 的波形，需要将 TABnCCRk 寄存器设置为 0000H。如果 TABnCCR0 寄存器的设定值为 FFFFH，则会周期性产生 INTTABnCCk 中断请求信号。



备注 k = 1 至 3
n = 0, 1

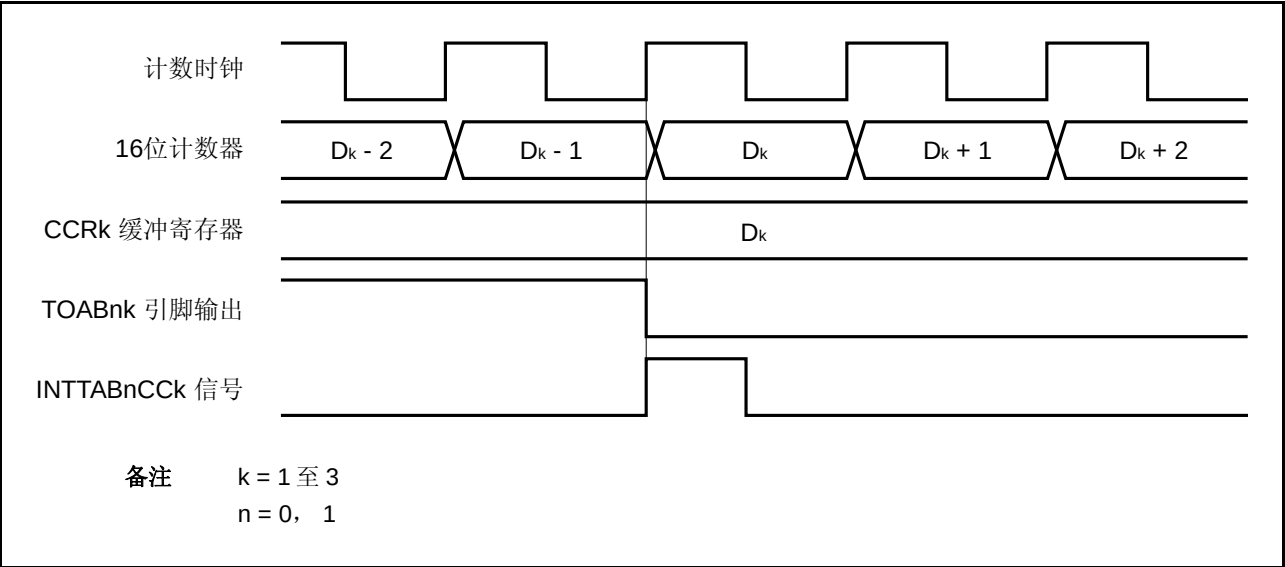
要输出 100%的波形，需给 TABnCCRk 寄存器设置一个“TABnCCR0 寄存器设定值 + 1”的数值。如果 TABnCCR0 寄存器的设定值为 FFFFH，则不会产生 100%波形输出。



备注 k = 1 至 3
n = 0, 1

(c) 比较匹配中断请求信号 (INTTABnCCK) 的发生时序

在 PWM 输出模式下，比较匹配中断请求信号 (INTTABnCCK) 的发生时序与其它情况下 INTTABnCCK 信号的时序有所不同；当 16 位计数器的计数值与 TABnCCRk 寄存器的值匹配时，产生 INTTABnCCK 信号。



通常情况下，在 16 位计数器的当前计数值与 TABnCCRk 寄存器的值匹配之后，INTTABnCCK 信号与和下一次计数同步发生。

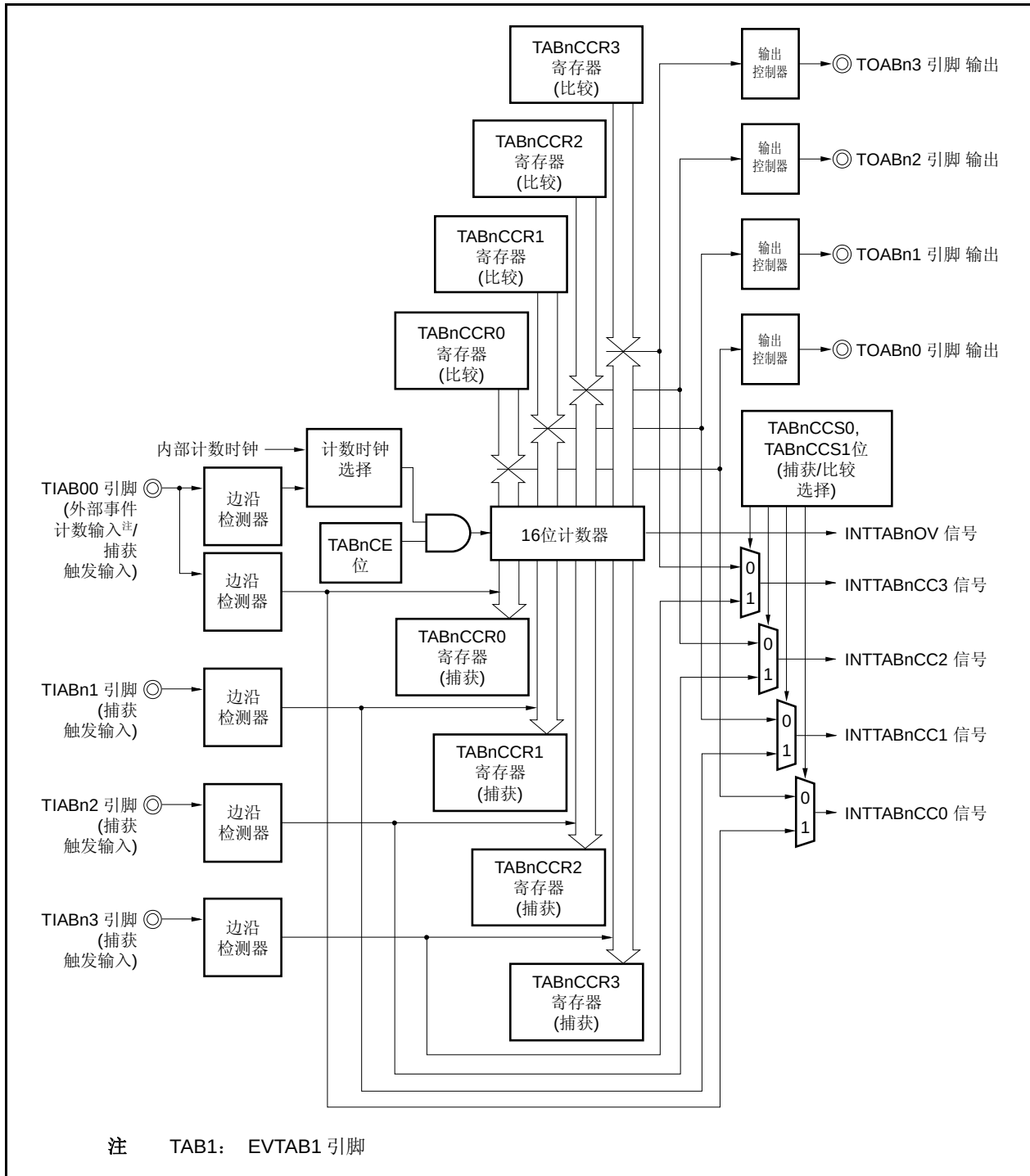
但是，在 PWM 输出模式下，INTTP0CC1 信号会提早一个计数时钟周期产生。这是由于为了匹配 TOABnk 引脚输出信号的变化，时序发生了变化。

8.5.6 自由运行定时器模式 (TABnMD2 至 TABnMD0 位= 101)

在自由运行定时器模式下, 当 TABnCTL0.TABnCE 位被置为 1 时, TABn 开始计数。此时, TABnCCRm 寄存器根据 TABnOPT0.TABnCCS0 和 TABnOPT0.TABnCCS1 位的设置, 可以用作比较寄存器或捕获寄存器。

备注 m = 0 至 3
 n = 0, 1

图 8-28. 自由运行定时器模式的配置



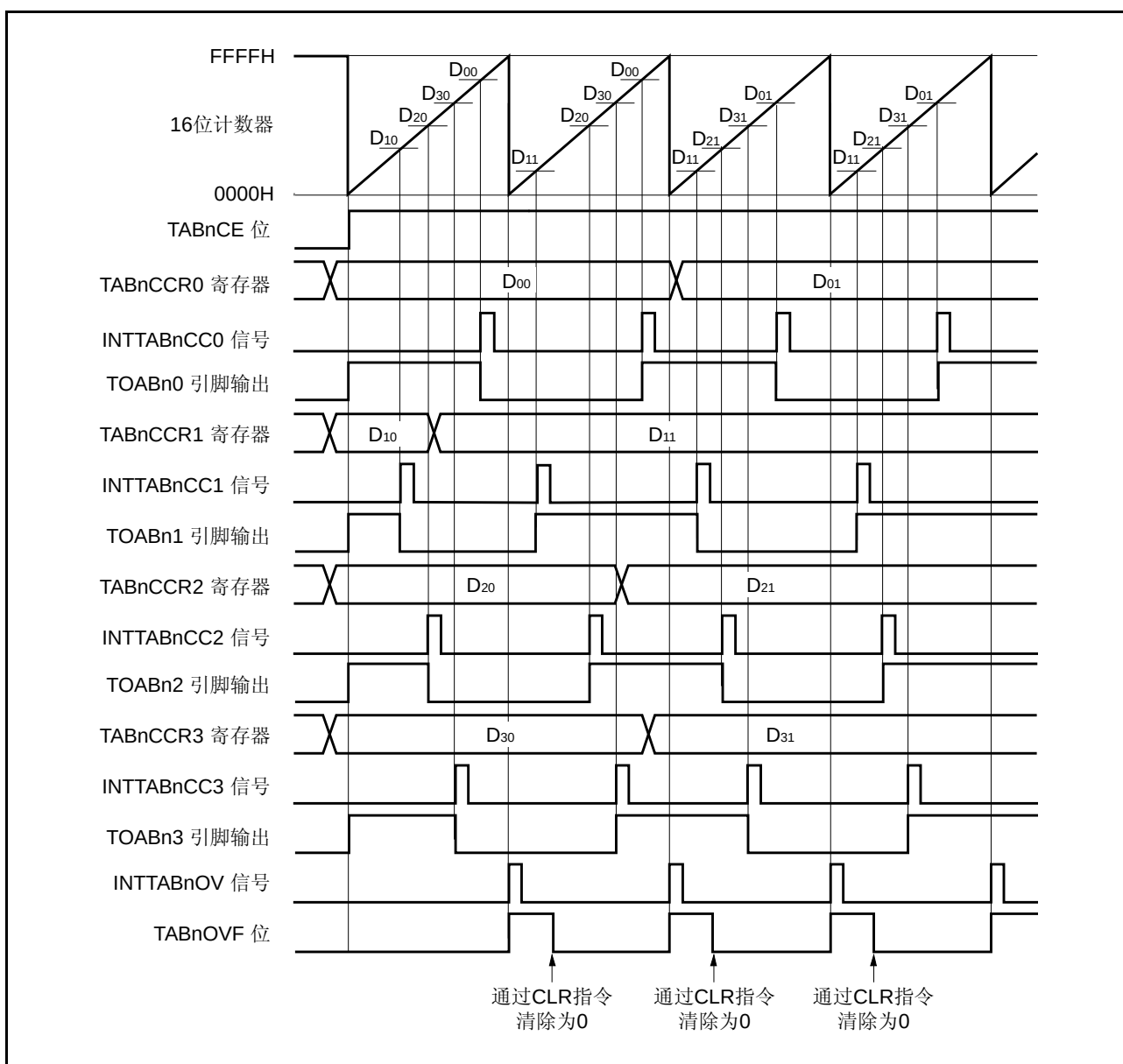
当 TABnCE 位设置为 1 时, TABn 开始计数并且 TOABn0 至 TOABn3 引脚的输出信号反相。此后当 16 位计数器的计数值与 TABnCCRm 寄存器的设置值匹配时, 产生一个比较匹配中断请求信号 (INTTABnCCm), 而且 TOABnm 引脚的输出信号反相。

16 位计数器继续与计数时钟同步进行计数。当它计数到 FFFFH 时, 会在下一个计数时钟时产生一个溢出中断请求信号 (INTTABnOV), 同时计数器清零为 0000H, 继续进行计数。此时, 溢出标志 (TABnOPT0.TABnOVF 位) 被置为 1, 通过软件运行 CLR 指令可把溢出标志位清 0。

当计数器工作时, TABnCCRm 寄存器可以重写。如果进行重写, 则新写入的值会在重写时立刻反映出来, 并与计数值进行比较。

备注 m = 0 至 3
 n = 0, 1

图 8-29. 自由运行定时器模式下的基本时序 (比较功能)



当 TABnCE 位被置为 1 时，16 位计数器开始计数。当检测到 TIABnm 引脚输入信号的有效沿时，16 位计数器的计数值存入 TABnCCRm 寄存器中，并产生一个捕获中断请求信号 (INTTABnCCm)。

16 位计数器继续与计数时钟同步进行计数。当它计数到 FFFFH 时，会在下一个计数时钟时产生一个溢出中断请求信号 (INTTABnOV)，同时计数器清零为 0000H，继续进行计数。此时，溢出标志 (TABnOVF 位) 被置为 1，通过软件运行 CLR 指令可把溢出标志位清 0。

备注 m = 0 至 3
 n = 0, 1

图 8-30. 自由运行定时器模式下的基本时序（捕获功能）

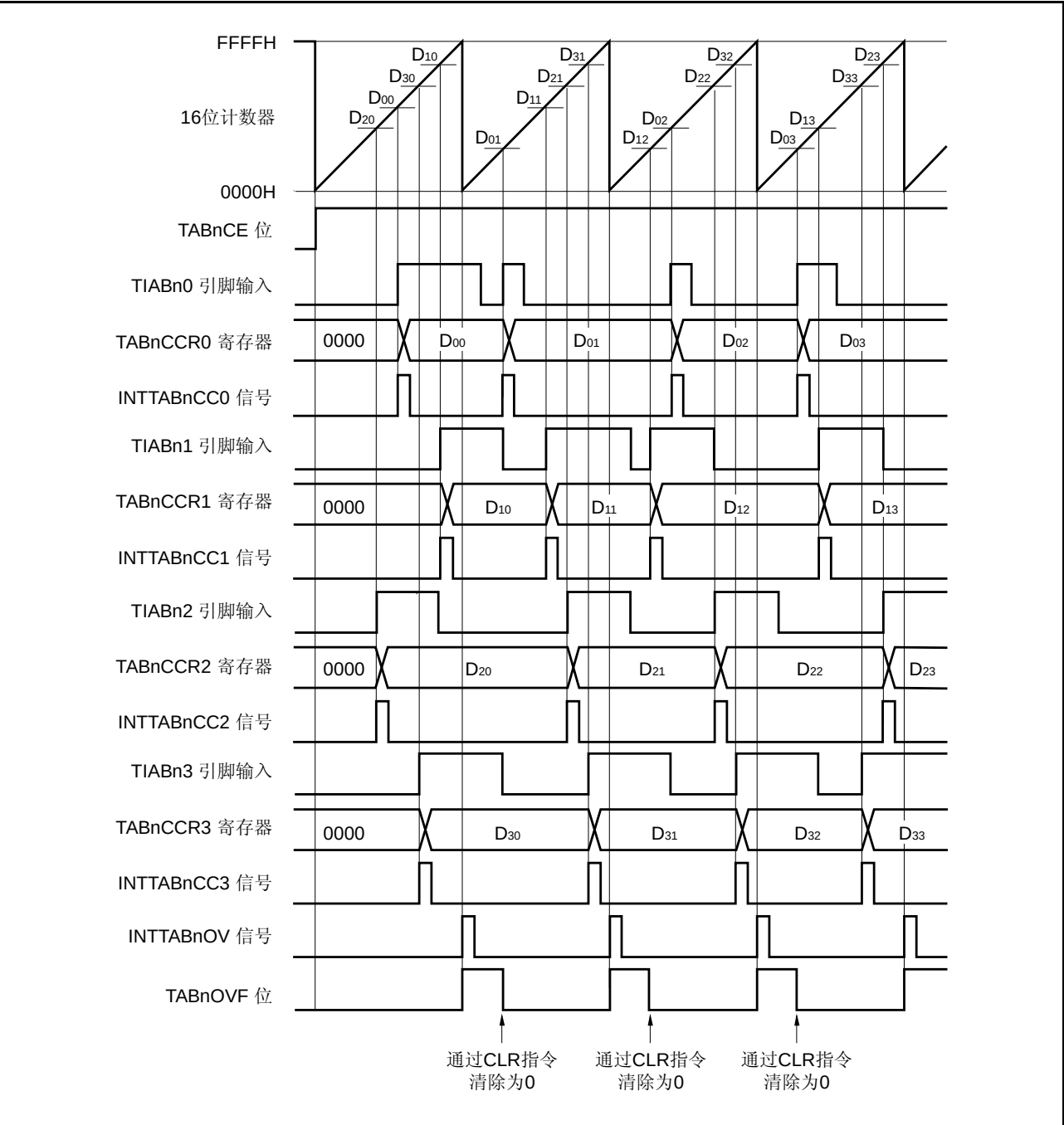


图 8-31. 自由运行定时器模式下的寄存器设置 (1/3)

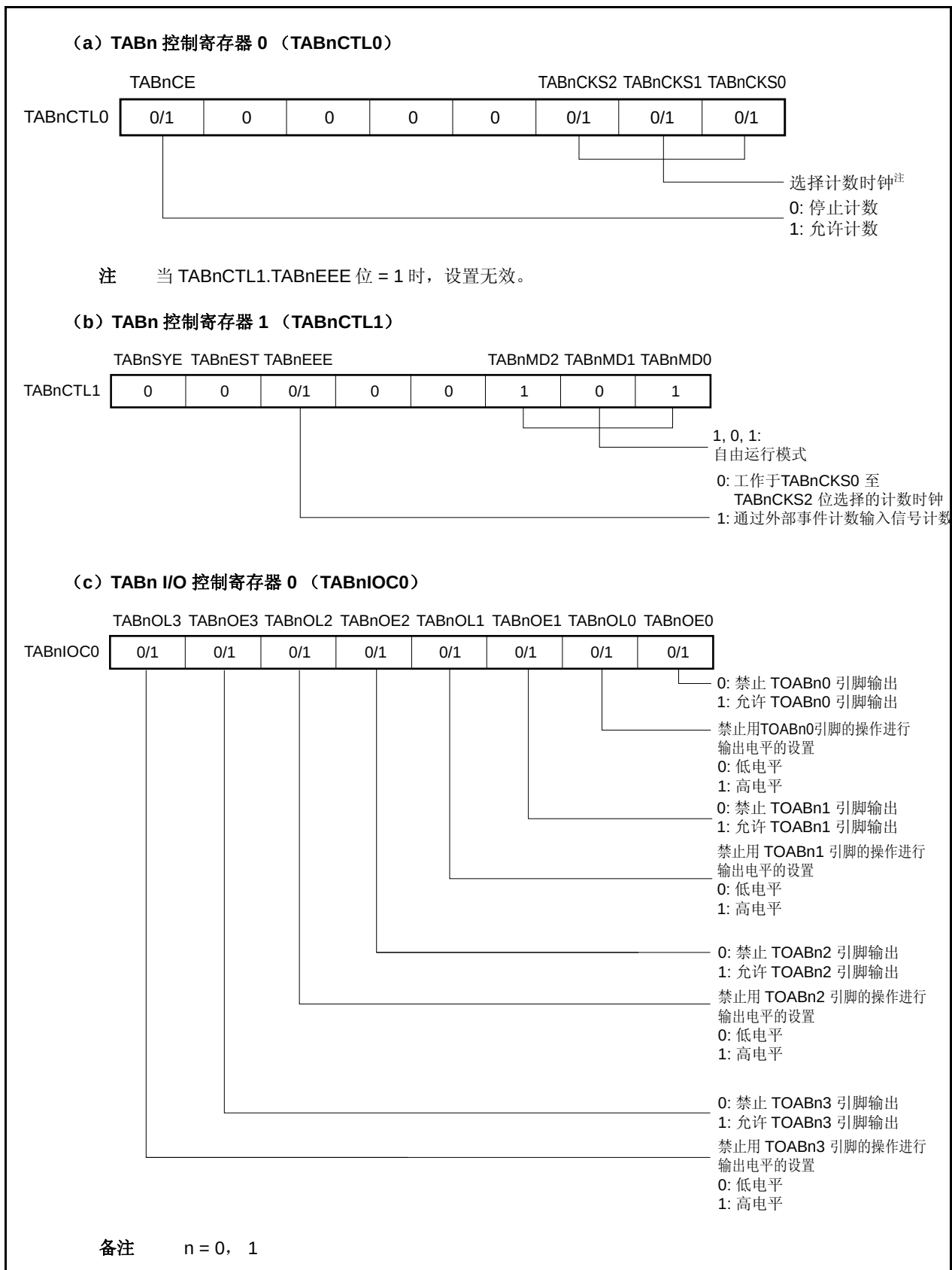
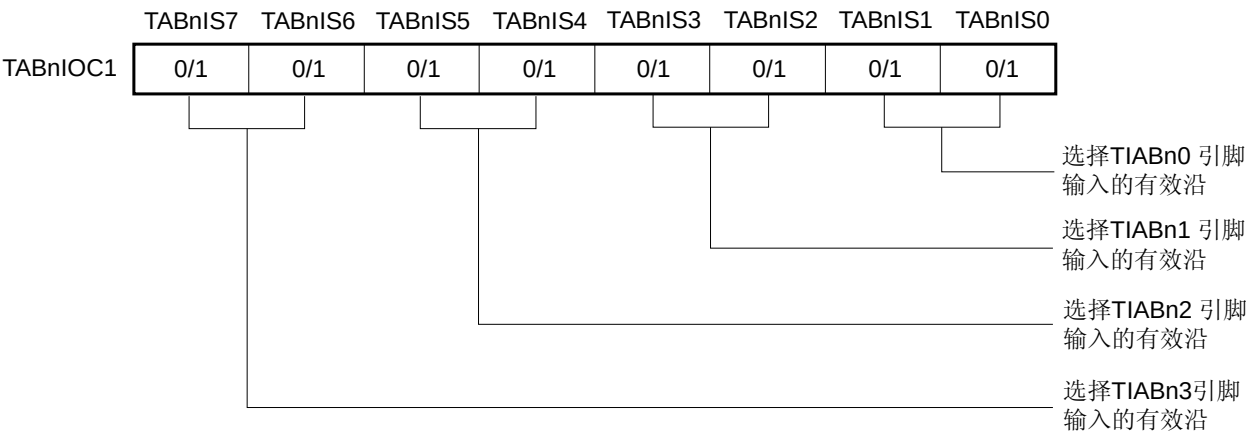
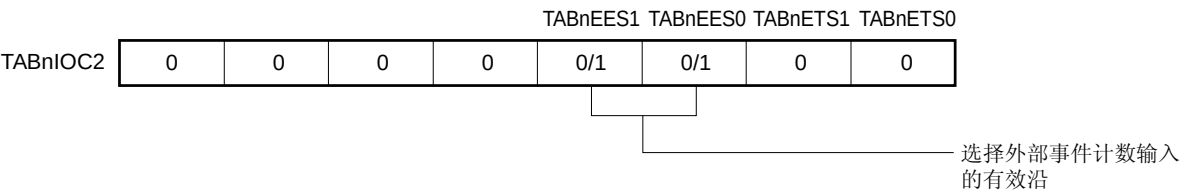


图 8-31. 自由运行定时器模式下的寄存器设置 (2/3)

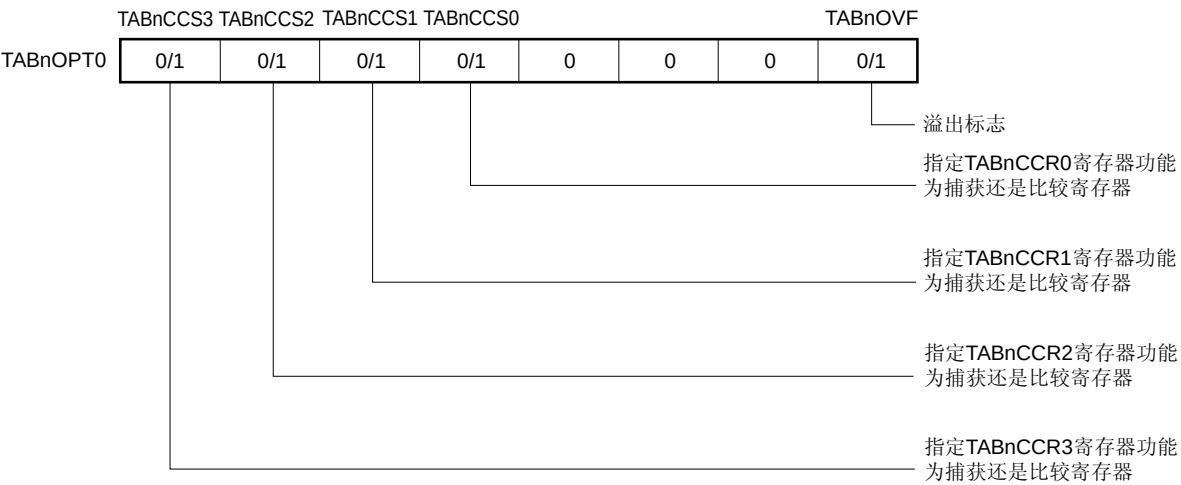
(d) TABn I/O 控制寄存器 1 (TABnIOC1)



(e) TABn I/O 控制寄存器 2 (TABnIOC2)



(f) TABn 选项寄存器 0 (TABnOPT0)



(g) TABn 计数器读取缓冲寄存器 (TABnCNT)

通过读取 TABnCNT 寄存器，可以读出 16 位计数器的值。

备注 n = 0, 1

图 8-31. 自由运行定时器模式下的寄存器设置 (3/3)

(h) TABn 捕获/比较寄存器 0 至 3 (TABnCCR0 至 TABnCCR3)

这些寄存器用作捕获寄存器还是比较寄存器需根据 TABnOPT0.TABnCCSm 位的设置。

当寄存器用作捕捉寄存器时，在检测到输入 TIABnm 引脚信号的有效边沿时，存储 16 位计数器的当前计数值。

当寄存器用作比较寄存器时，而且将 TABnCCRm 寄存器的值设置为 D_m 时，则当计数器达到 $(D_m + 1)$ 时会产生 INTTABnCCm 中断信号，同时，TOABnm 引脚的输出信号反相。

备注 $m = 0$ 至 3
 $n = 0, 1$

(1) 自由运行定时器模式时的操作流程

(a) 当捕获/比较寄存器用作比较寄存器时

图 8-32. 自由运行定时器模式时的软件处理流程（比较功能）（1/2）

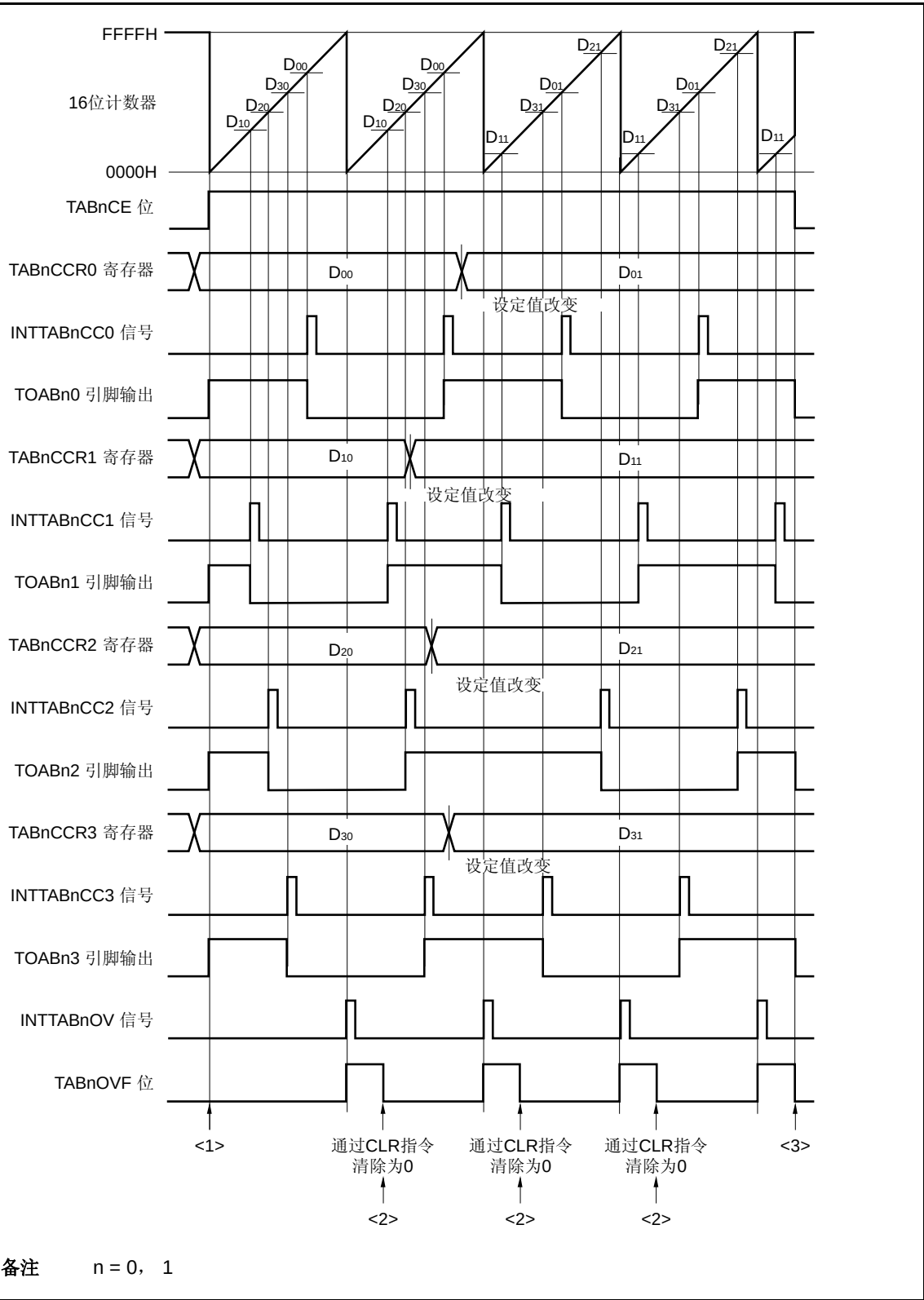
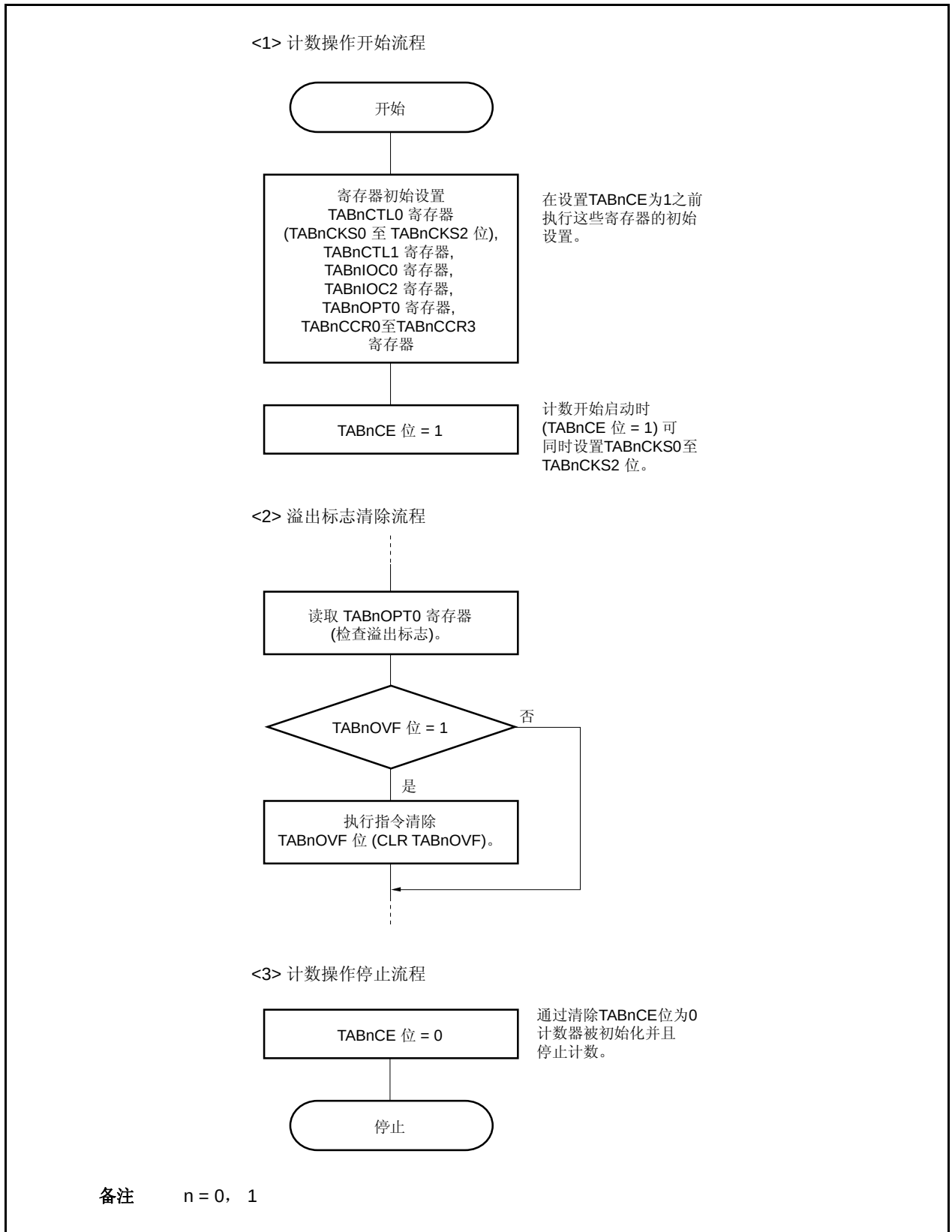
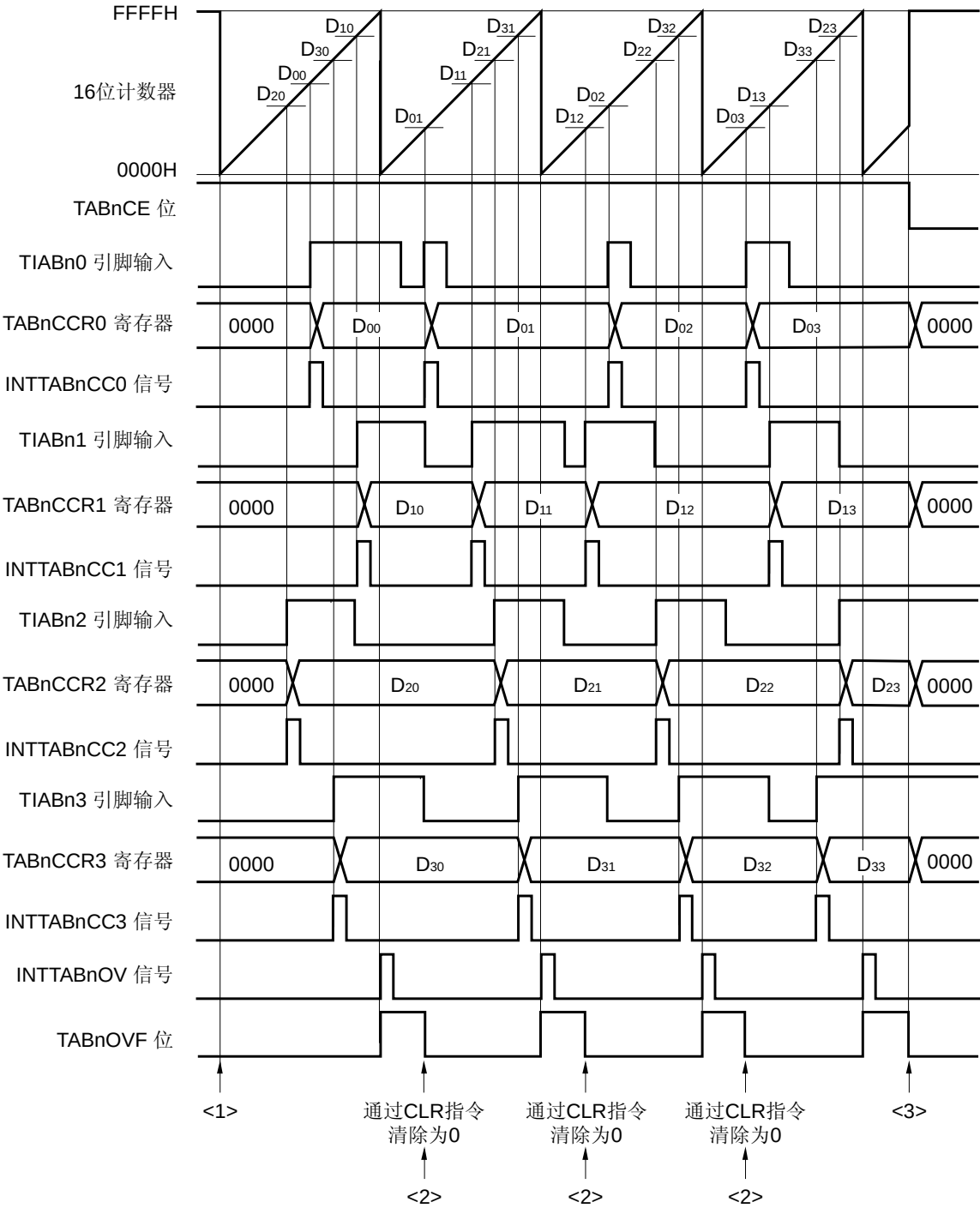


图 8-32. 自由运行定时器模式时的软件处理流程（比较功能）（2/2）



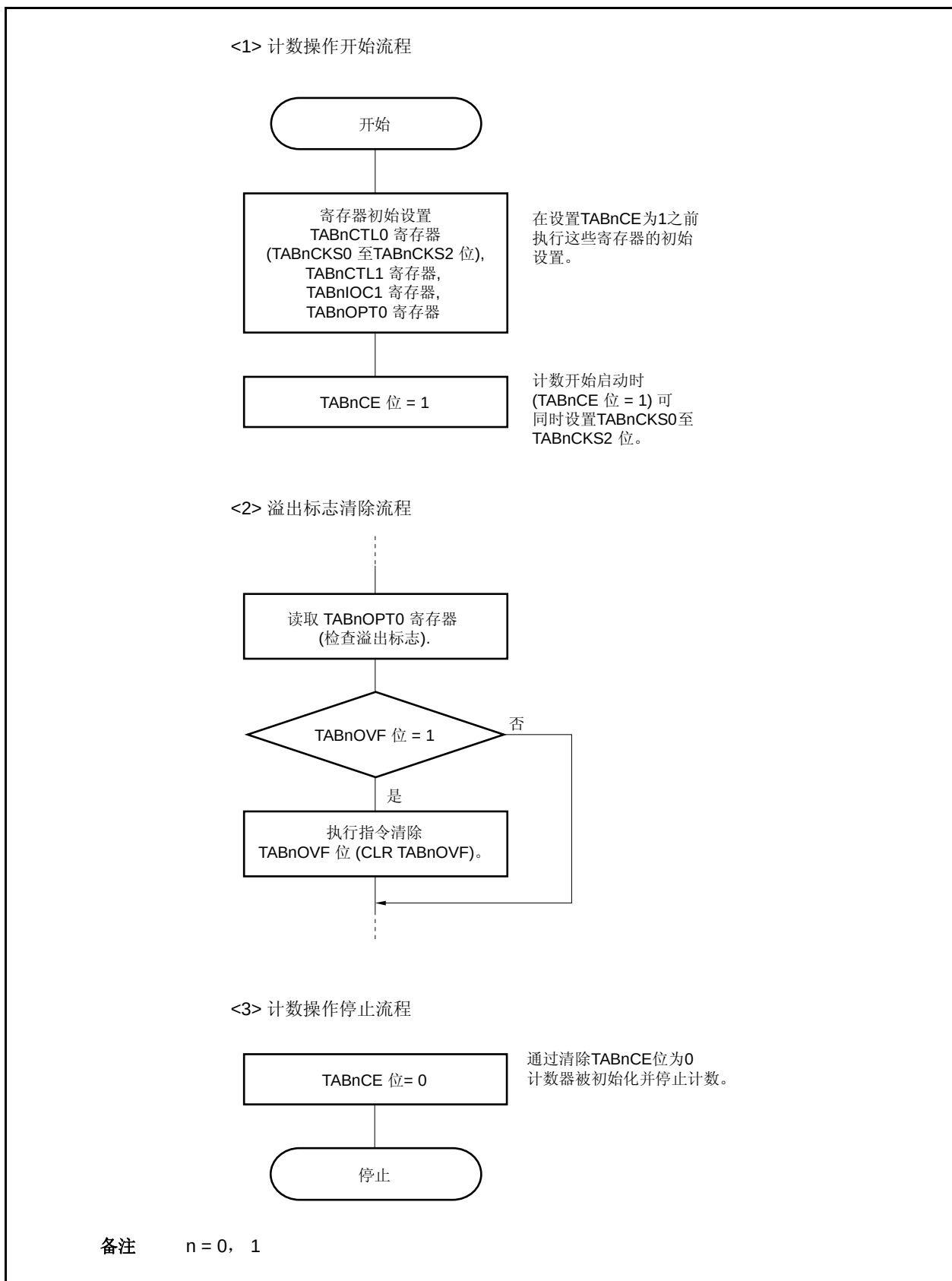
(b) 当捕获/比较寄存器用作捕获寄存器时

图 8-33. 自由运行定时器模式时的软件处理流程（捕获功能）（1/2）



备注 n = 0, 1

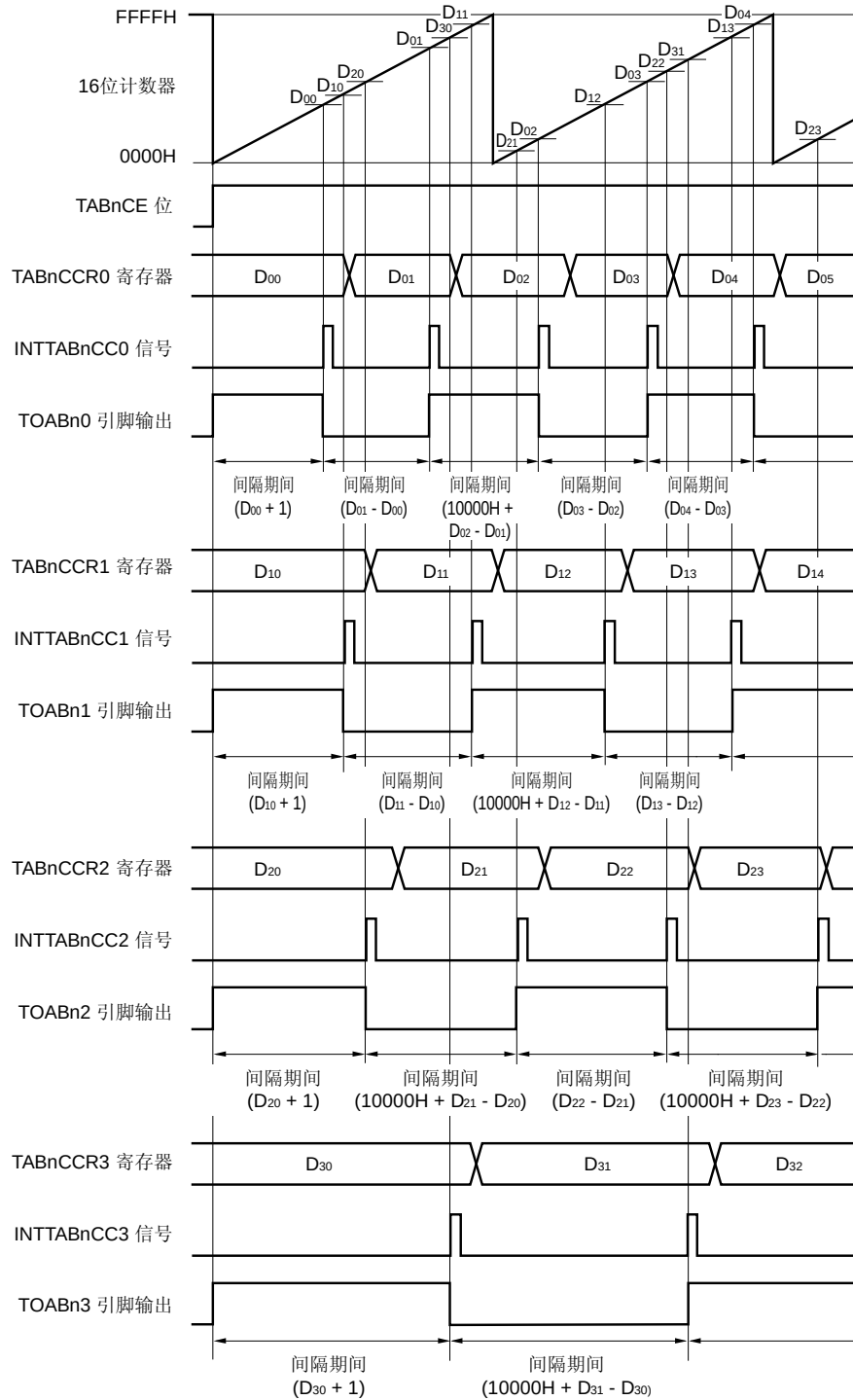
图 8-33. 自由运行定时器模式时的软件处理流程（捕获功能）（2/2）



(2) 自由运行定时器模式时的工作时序

(a) 使用比较寄存器的间隔定时操作

当 TABn 用作间隔定时器而 TABnCCRm 寄存器用作比较寄存器时，在每次检测到 INTTABnCCm 信号后，都需要利用软件操作设置一个比较值以产生下一个中断请求信号。



备注 $n = 0, 1$

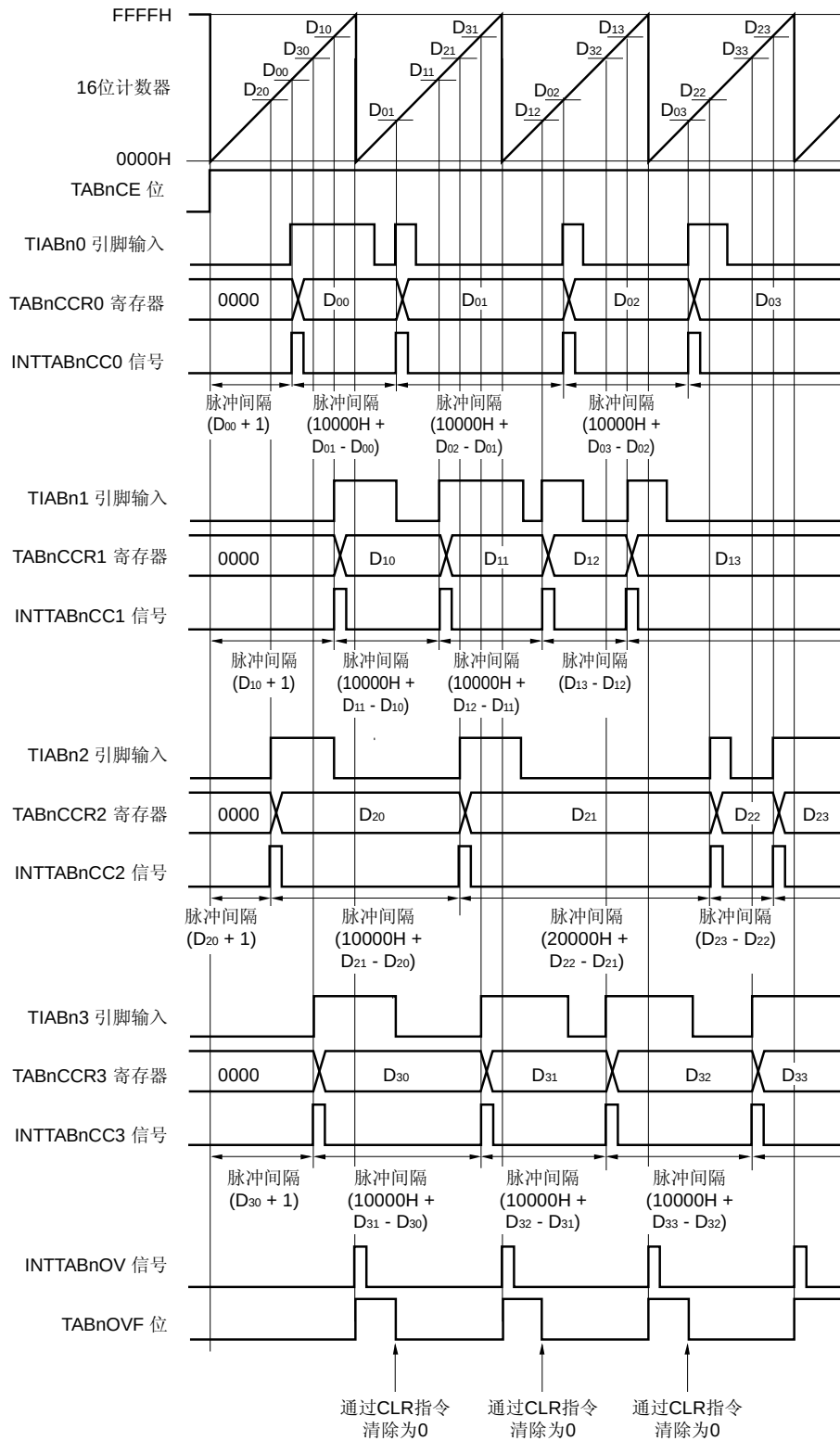
在自由运行定时器模式下进行时间间隔操作时，一个通道可以设置四个时间间隔。
要进行时间间隔操作，当检测到 INTTABnCCm 信号后，必须在中断服务程序中重置相应的 TABnCCRm 寄存器。
重置 TABnCCRm 寄存器的设置值可由下式计算，其中“Dm”是间隔时段：

比较寄存器默认值： $D_m - 1$
比较寄存器的第二次及其以后的设置值：上次设置值 + D_m
(若计算结果大于 FFFFH，则从结果中减去 10000H 并将此值设置给寄存器。)

备注 m = 0 至 3
 n = 0, 1

(b) 使用捕获寄存器进行脉冲宽度测量

在用 TABnCCRm 寄存器作为捕获寄存器进行脉宽测量时，每次检测到 INTTABnCCm 信号时读取捕获寄存器及计算时间间隔，都需要进行软件处理。



备注 n = 0, 1

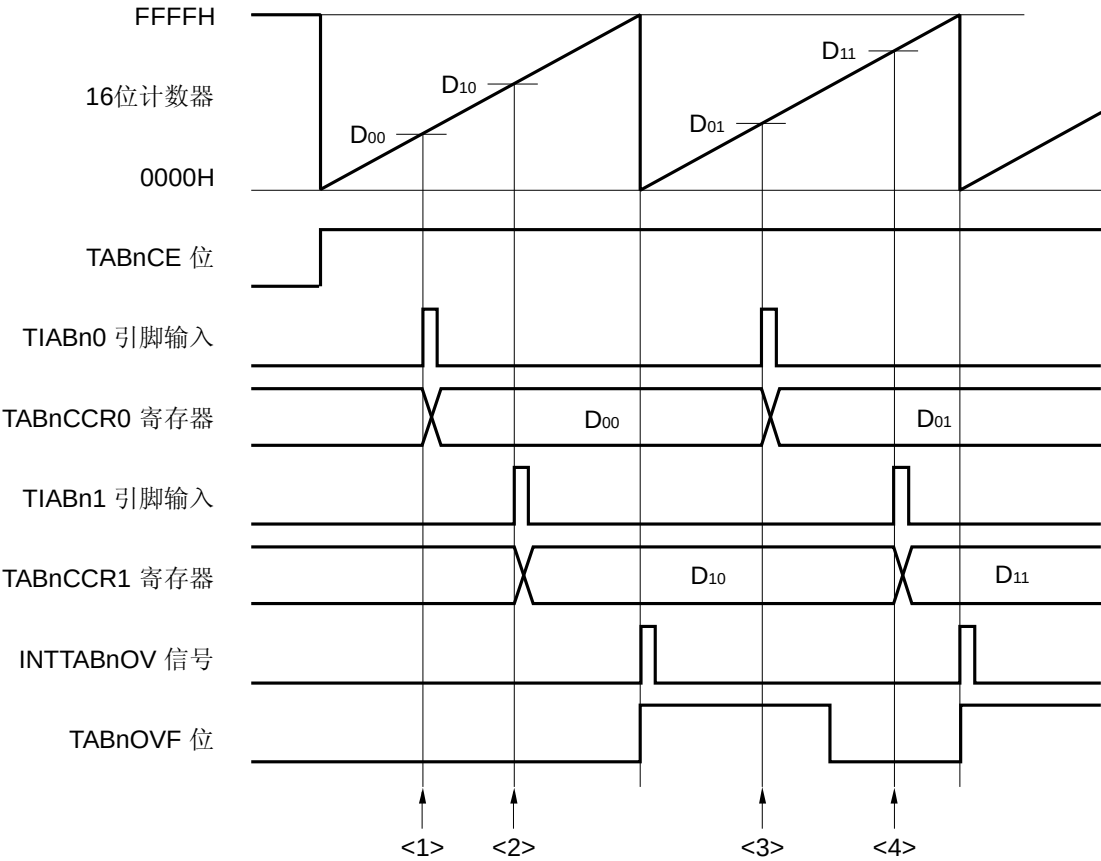
在自由运行定时器模式下执行脉宽测量时，可以用一个通道测量四个脉冲宽度。
要测量脉冲宽度的话，可以通过与 INTTABnCCm 信号同步读取 TABnCCRm 寄存器的值，并计算本次读取值与先前读取值之间的差，计算得到脉冲宽度。

备注 m = 0 至 3
 n = 0, 1

(c) 当使用两个或更多的捕获寄存器时溢出的处理

当使用两个或更多的捕获寄存器时必须格外小心。首先，如下所示，是一个不正确处理的例子。

使用两个或更多捕获寄存器时处理错误的例子



在自由运行定时器模式下测量两个脉冲宽度时，可能出现下列问题。

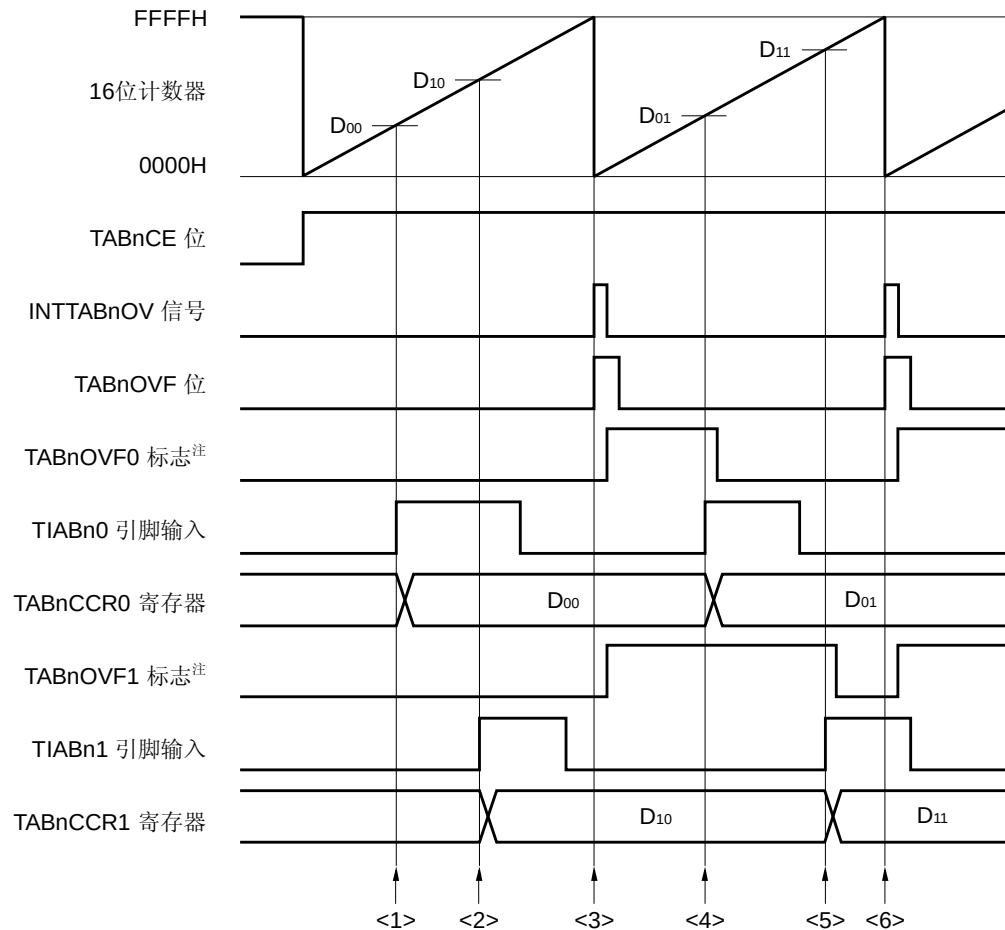
- <1> 读取 TABnCCR0 寄存器（设置 TIABn0 引脚输入的默认值）。
- <2> 读取 TABnCCR1 寄存器（设置 TIABn1 引脚输入的默认值）。
- <3> 读取 TABnCCR0 寄存器。
读溢出标志。若溢出标志为 1，将其清为 0。
因为溢出标志为 1，所以脉冲宽度可以用 $(10000H + D_{01} - D_{00})$ 计算。
- <4> 读取 TABnCCR1 寄存器。
读溢出标志。因为此标志在<3>中已清零，所以，读取到 0。
因为溢出标志为 0，所以脉冲宽度可以用 $(D_{11} - D_{10})$ 计算。（不正确）

备注 $n = 0, 1$

在使用两个或更多的捕获寄存器时，如果溢出标志由一个捕获寄存器清为 0，则另一个捕捉寄存器可能不会得到正确的脉冲宽度。

在使用两个或更多的捕获寄存器时，使用软件解决。关于如何使用软件，示例如下。

使用两个捕获寄存器时的例子 (使用溢出中断)



注 由软件对内部 RAM 中的 TABnOVF0 和 TABnOVF1 标志进行设置。

<1> 读取 TABnCCR0 寄存器 (设置 TIABn0 引脚输入的默认值)。

<2> 读取 TABnCCR1 寄存器 (设置 TIABn1 引脚输入的默认值)。

<3> 发生溢出。在溢出中断服务程序中将 TABnOVF0 和 TABnOVF1 标志设置为 1，并将溢出标志清为 0。

<4> 读取 TABnCCR0 寄存器。

读取 TABnOVF0 标志。如果 TABnOVF0 标志为 1，则将其清为 0。

因为 TABnOVF0 标志为 1，所以脉冲宽度可以用 $(10000H + D_{01} - D_{00})$ 计算。

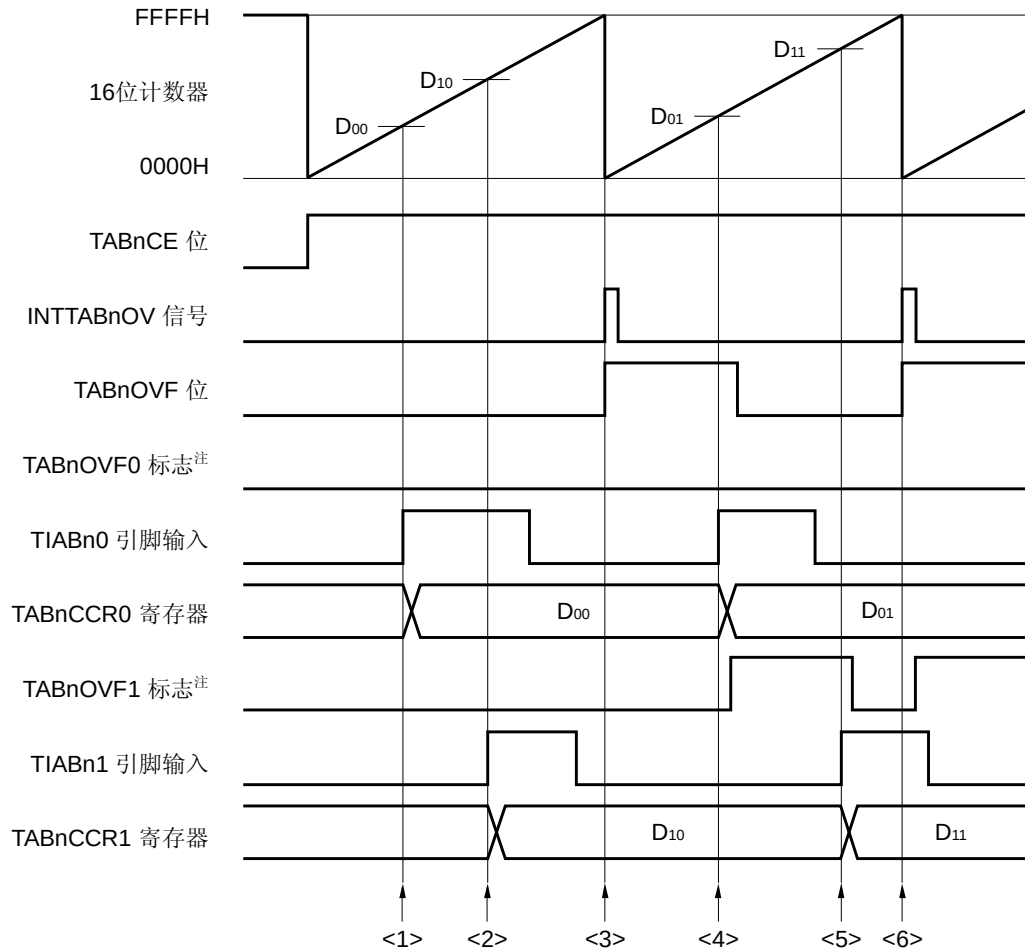
<5> 读取 TABnCCR1 寄存器。

读取 TABnOVF1 标志。如果 TABnOVF1 标志为 1，则将其清为 0 (TABnOVF0 标志在<4>中已清零，而 TABnOVF1 标志仍然保持为 1)。

因为 TABnOVF1 标志为 1，所以脉冲宽度可以用 $(10000H + D_{11} - D_{10})$ 计算。(正确)

<6> 重复<3>。

使用两个捕获寄存器时的例子 (不使用溢出中断)

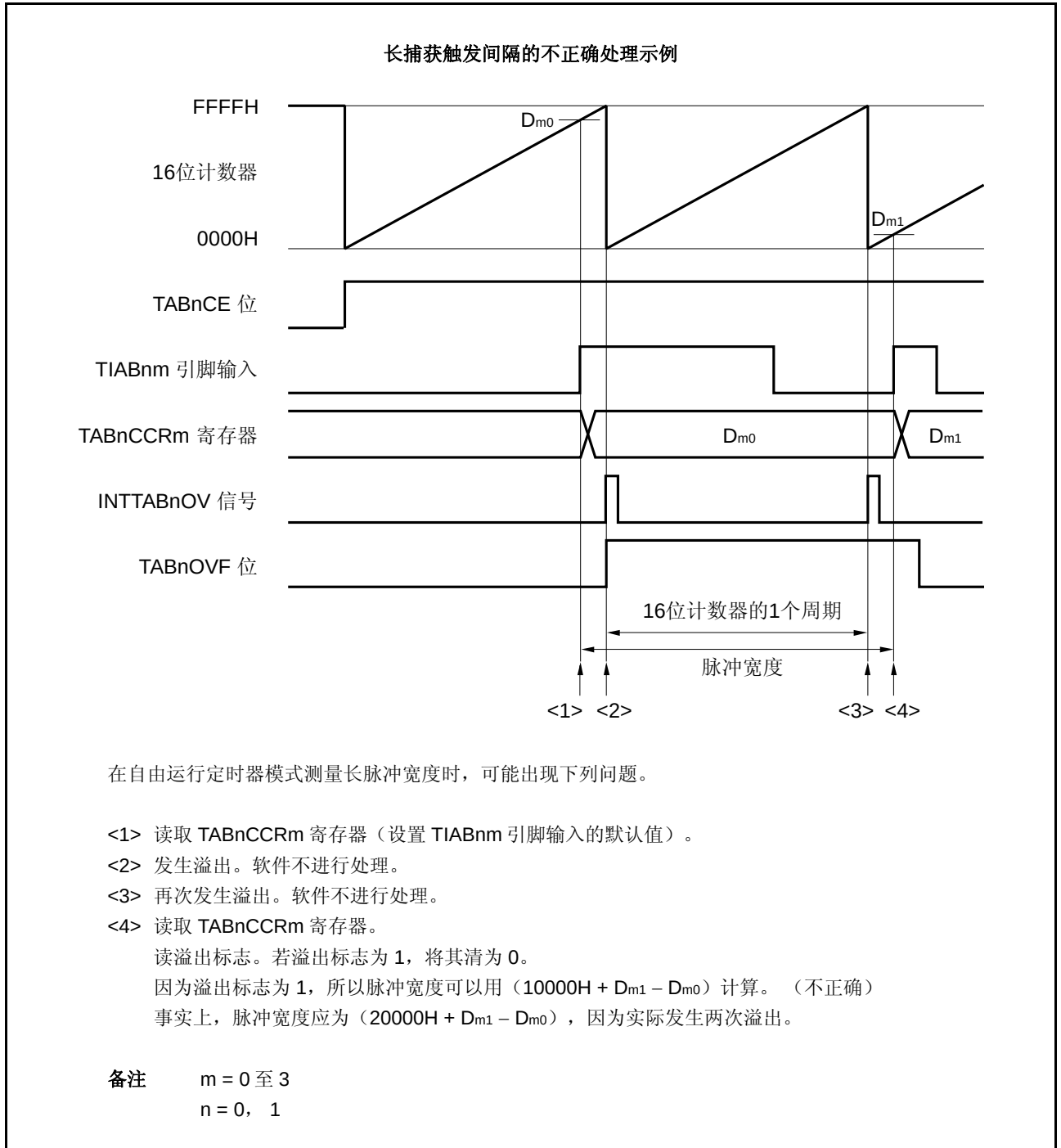


注 由软件对内部 RAM 中的 TABnOVF0 和 TABnOVF1 标志进行设置。

- <1> 读取 TABnCCR0 寄存器 (设置 TIABn0 引脚输入的默认值)。
- <2> 读取 TABnCCR1 寄存器 (设置 TIABn1 引脚输入的默认值)。
- <3> 发生溢出。软件不进行处理。
- <4> 读取 TABnCCR0 寄存器。
读溢出标志。如果溢出标志为 1, 则只需设置 TABnOVF1 标志为 1, 并清除溢出标志为 0。
因为溢出标志为 1, 所以脉冲宽度可以用 $(10000H + D_{01} - D_{00})$ 计算。
- <5> 读取 TABnCCR1 寄存器。
读溢出标志。因为溢出标志在 <4> 中清零了, 所以读取 0。
读取 TABnOVF1 标志。若 TABnOVF1 标志为 1, 将其清为 0。
因为 TABnOVF1 标志为 1, 所以脉冲宽度可以用 $(10000H + D_{11} - D_{10})$ 计算。 (正确)
- <6> 重复 <3>。

(d) 长捕获触发间隔时溢出的处理

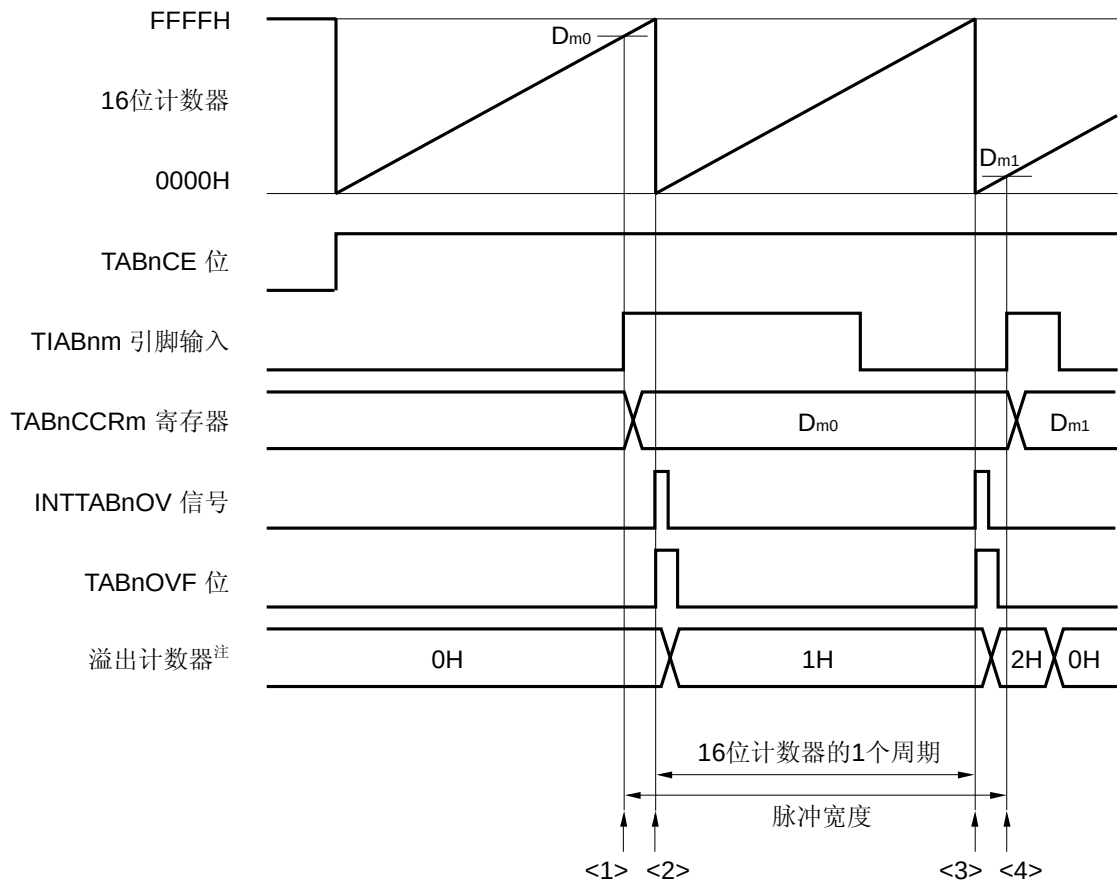
如果脉冲宽度大于 16 位计数器的一个计数循环，则必须加以注意，因为从第一个捕获触发到下一个之间可能发生不止一次溢出。首先，如下所示，是一个不正确处理的例子。



如果长捕获触发间隔期间，发生了两次或更多次的溢出，则无法获得正确的脉冲宽度。

如果捕获触发间隔为长间隔（相对于计数循环），应减慢计数时钟以延长 16 位计数器的循环周期，或者使用软件加以解决。关于如何使用软件，示例如下。

长捕获触发间隔时的举例



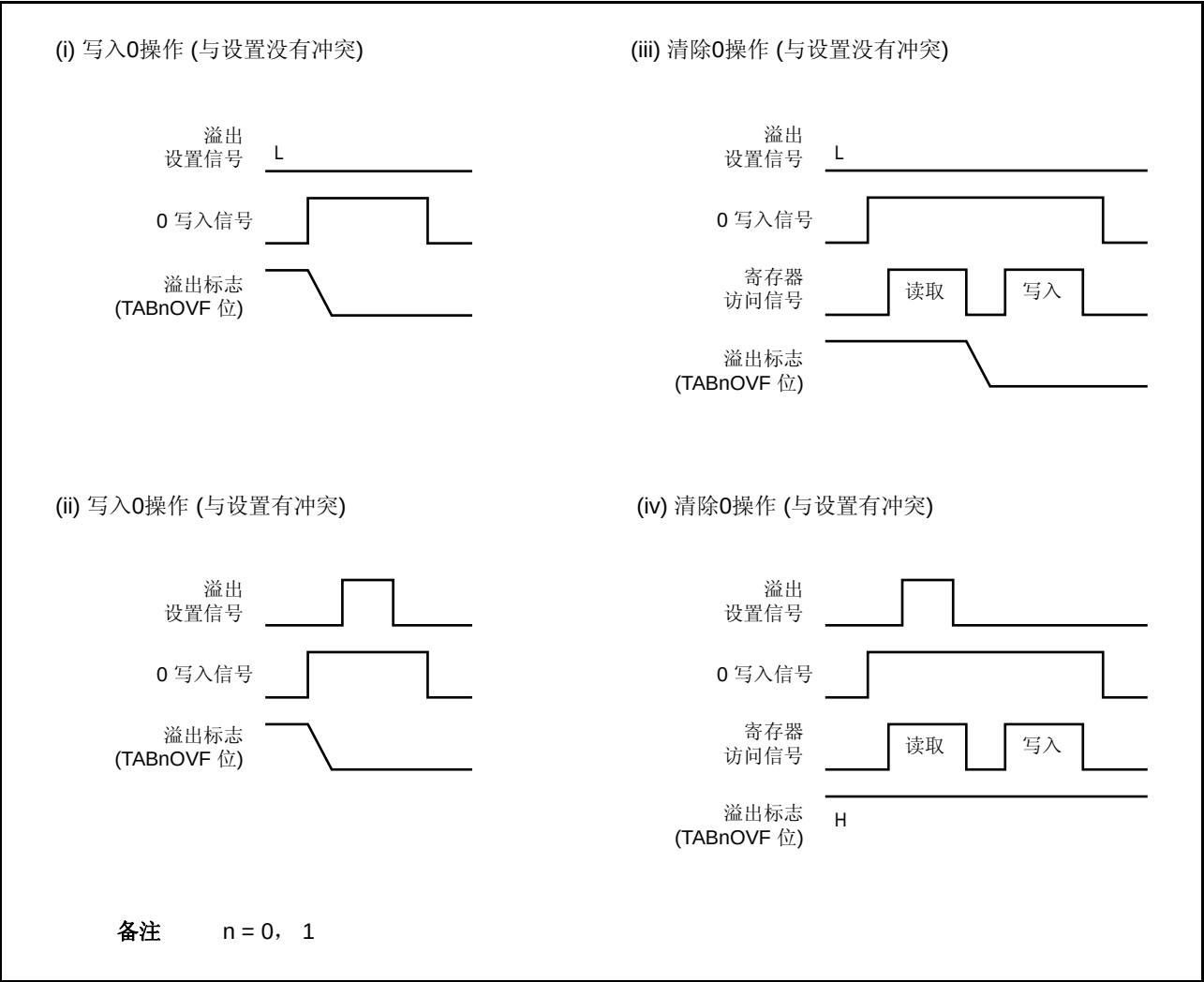
注 内部 RAM 中的溢出计数器由软件任意设置。

- <1> 读取 TABnCCRm 寄存器（设置 TIABnm 引脚输入的默认值）。
- <2> 发生溢出。增加溢出计数器的值，并在溢出中断服务程序中把溢出标志清为 0。
- <3> 再次发生溢出。增加 (+1) 溢出计数器的值，并在溢出中断服务程序中把溢出标志清为 0。
- <4> 读取 TABnCCRm 寄存器。
读取溢出计数器。
当溢出计数器为“N”时，脉冲宽度可用 $(N \times 10000H + D_{m1} - D_{m0})$ 计算。
在本例中，脉冲宽度为 $(20000H + D_{m1} - D_{m0})$ ，因为溢出发生了两次。
最后，溢出计数器清零 (0H)。

备注 m = 0 至 3
n = 0, 1

(e) 清除溢出标志

通过用 CLR 指令将 TABnOVF 位清为 0 以及将 8 位数据 (位 0 是 0) 写入 TABnOPT0 寄存器, 可以将溢出标志清 0。要准确检测溢出, 当 TABnOVF 位为 1 时读取 TP0OVF 位, 然后, 用位操作指令将溢出标志清零。



要将溢出标志清为 0, 应读取溢出标志, 检查是否被设置为 1, 并用 CLR 指令把它清零。若在未检查溢出标志是否为 1 的情况下把 0 写入溢出标志中, 则溢出的设置信息可能因为写入 0 而被清除 (上图中的 (ii))。因此, 即使实际发生了溢出, 软件也可能会判断为无溢出。

如果用 CLR 指令将溢出标志清为 0 时, CLR 指令的执行与当前发生的溢出相冲突, 则溢出标志即使在 CLR 指令执行后也会保持置位状态。

8.5.7 脉冲宽度测量模式 (TABnMD2 至TABnMD0 位= 110)

在脉冲宽度测量模式下，当 TABnCTL0.TABnCE 位被置为 1 时，TABn 开始计数。每次检测到 TIABnm 引脚输入信号的有效边沿时，16 位计数器的计数值就存储在 TABnCCRm 寄存器中，且 16 位计数器清为 0000H。

在出现捕获中断请求信号 (INTTABnCCm) 之后，可以通过读取 TABnCCRm 寄存器，测量有效边缘的间隔。

选择 TIABn0 至 TIABn3 引脚中的一个作为捕获触发输入引脚。使用 TABnIOC1 寄存器将未使用的引脚指定为“无边沿检测”引脚。

在用外部时钟作为计数时钟时，测量 TIAB0k 引脚输入的脉冲宽度，因为外部时钟固定在 TIAB00 引脚上。此时，把 TAB0IOC1.TAB0IS1 和 TAB0IOC1.TAB0IS0 位清零为 00（捕获触发输入 (TIAB00 引脚)：无边沿检测）。

对于 TAB1，从 EVTAB1 引脚输入外部时钟并用 TIAB10 至 TIAB13 引脚进行脉冲宽度测量。

备注 m = 0 至 3
 n = 0, 1
 k = 1 至 3

图 8-34. 脉冲宽度测量模式的配置

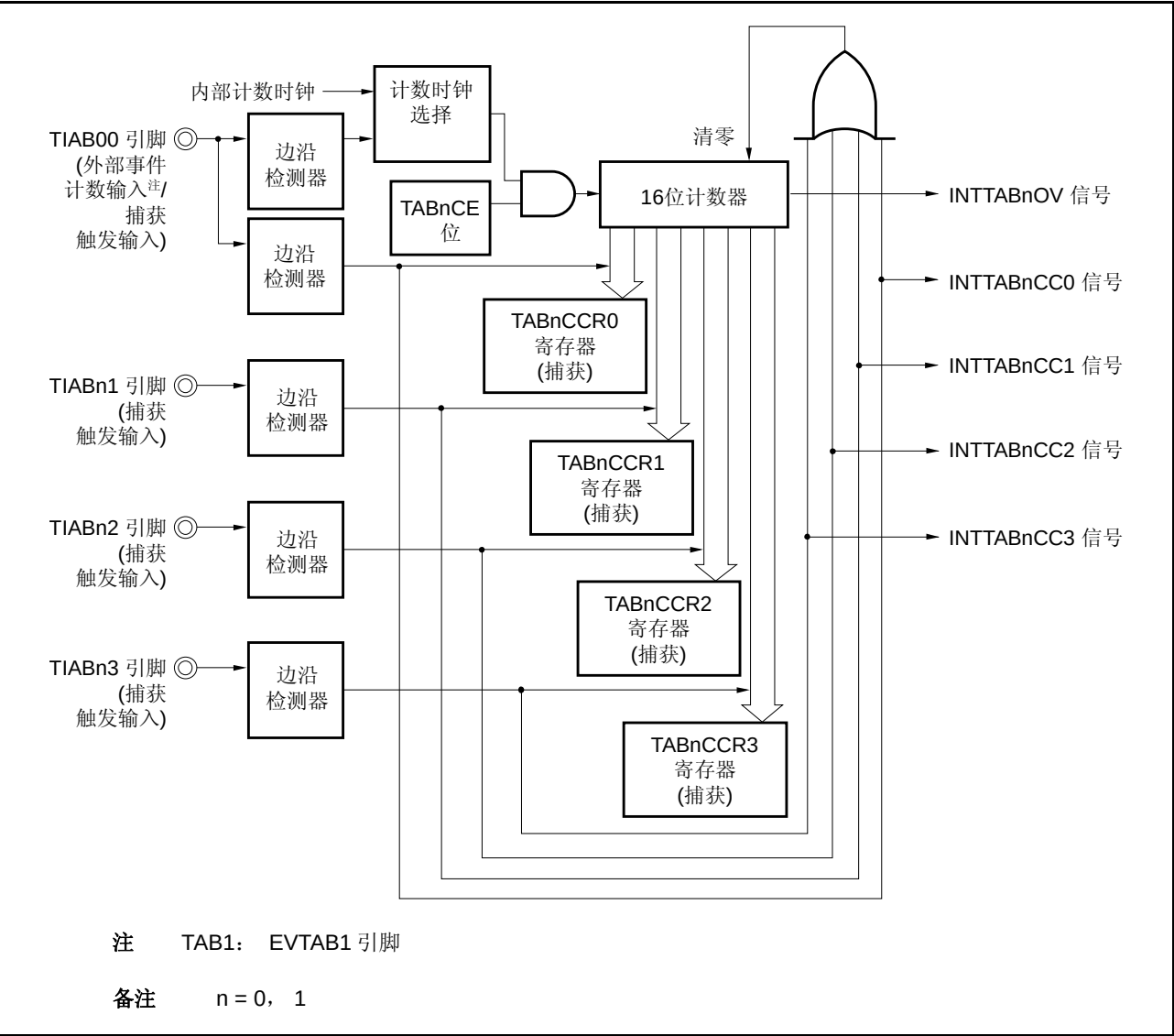
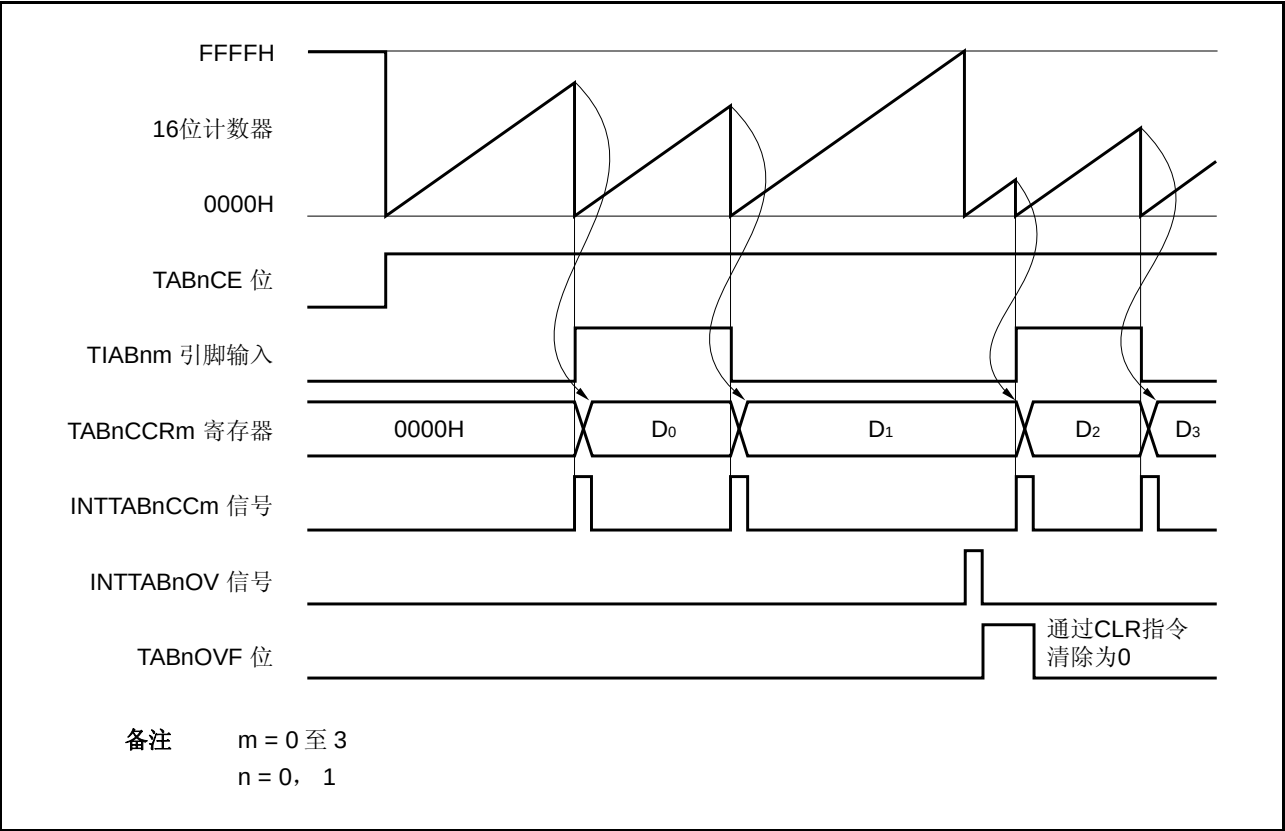


图 8-35. 脉冲宽度测量模式的基本时序



当 TABnCE 位被置为 1 时，16 位计数器开始计数。此后检测到 TIABnm 引脚输入信号的有效边沿时，16 位计数器的计数值存储在 TABnCCRm 寄存器中，16 位计数器清零为 0000H，并生成一个捕获中断请求信号 (INTTABnCCm)。

脉冲宽度计算如下。

脉冲宽度= 捕获值×计数时钟周期

即使 16 位计数器一直计数到 FFFFH，TIABnm 引脚输入信号一直没有有效边沿，仍然会在下一个计数时钟时产生溢出中断请求信号 (INTTABnOV)，且计数器清零为 0000H，计数器继续计数。此时，溢出标志 (TABnOPT0.TABnOVF 位) 也被置为 1。使用软件执行 CLR 指令将溢出标志清为 0。

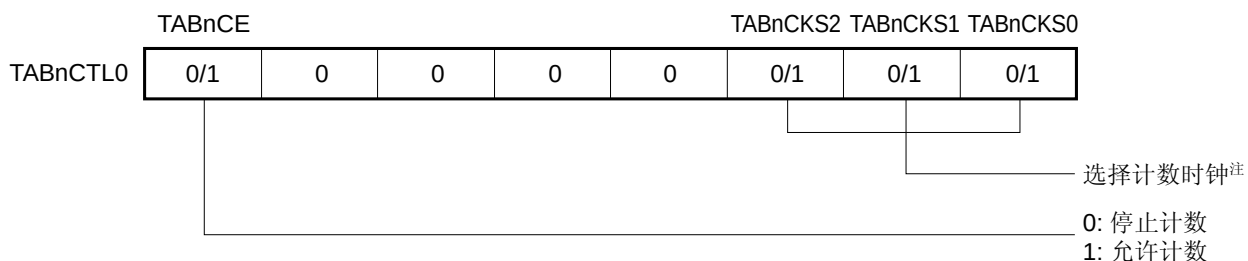
若溢出标志设置为 1，则脉冲宽度可计算如下。

脉冲宽度= (10000H×TABnOVF 位置 (1) 的次数+捕获值) ×计数时钟周期

备注 m = 0 至 3
n = 0, 1

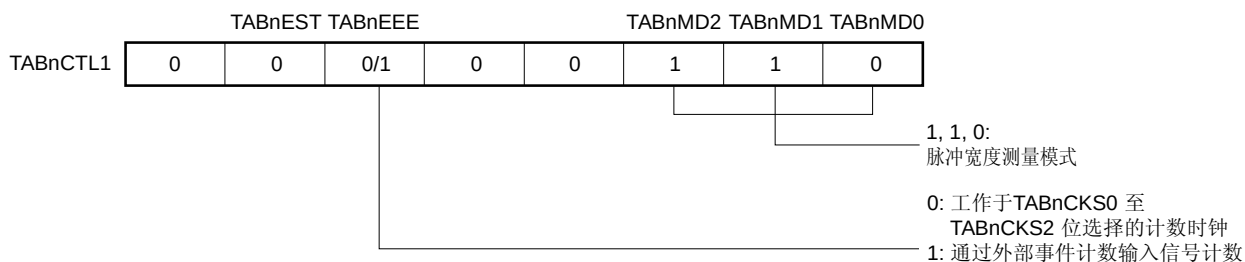
图 8-36. 脉冲宽度测量模式时的寄存器设置 (1/2)

(a) TABn 控制寄存器 0 (TABnCTL0)

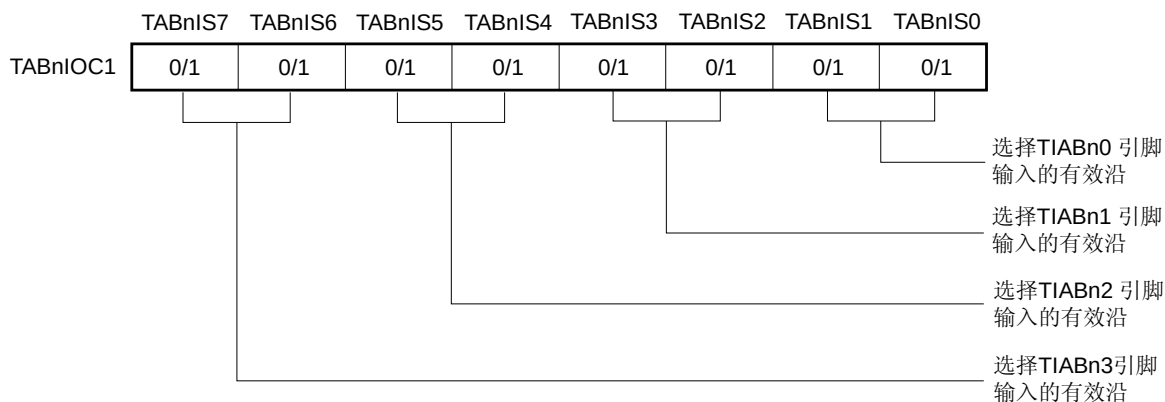


注 当 TABnEEE 位 = 1 时，设置无效。

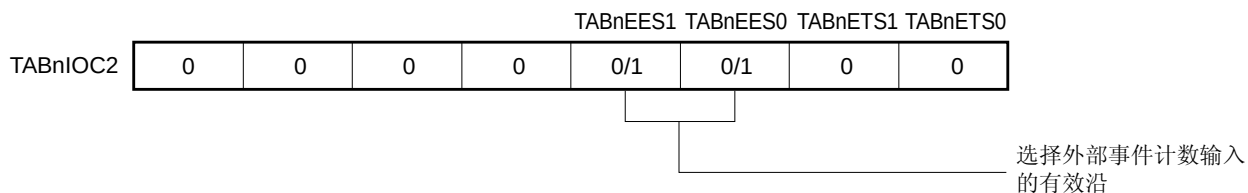
(b) TABn 控制寄存器 1 (TABnCTL1)



(c) TABn I/O 控制寄存器 1 (TABnIOC1)

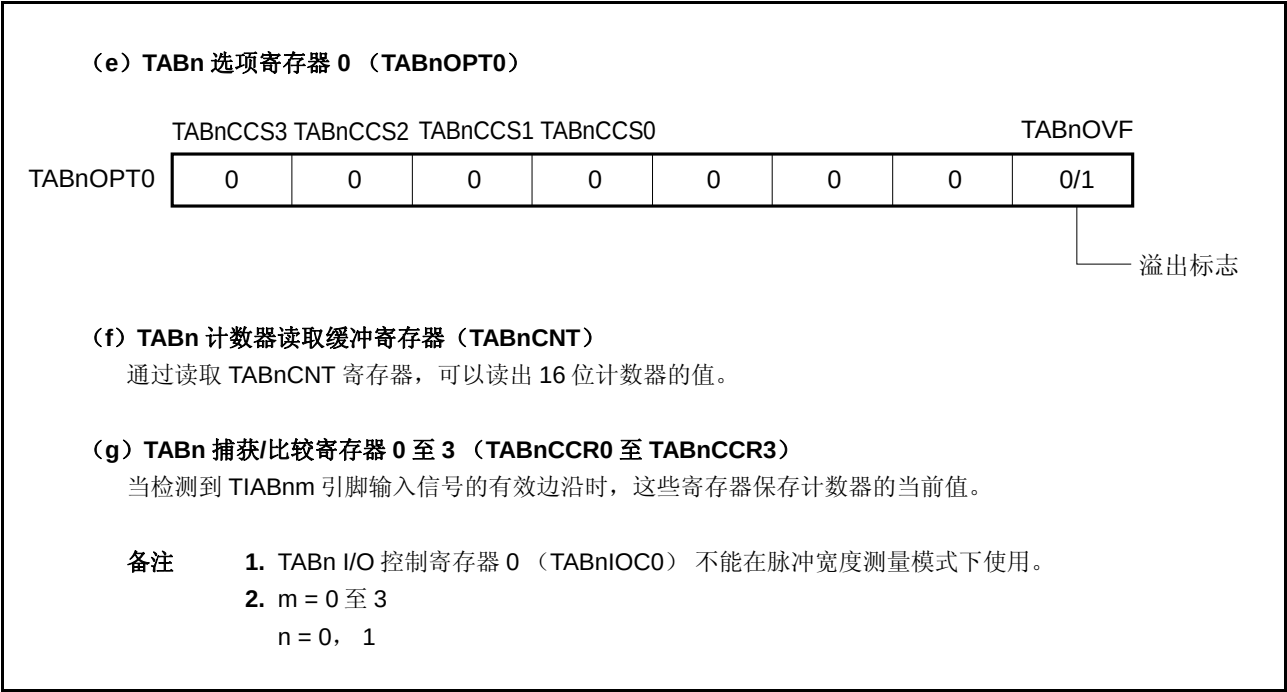


(d) TABn I/O 控制寄存器 2 (TABnIOC2)



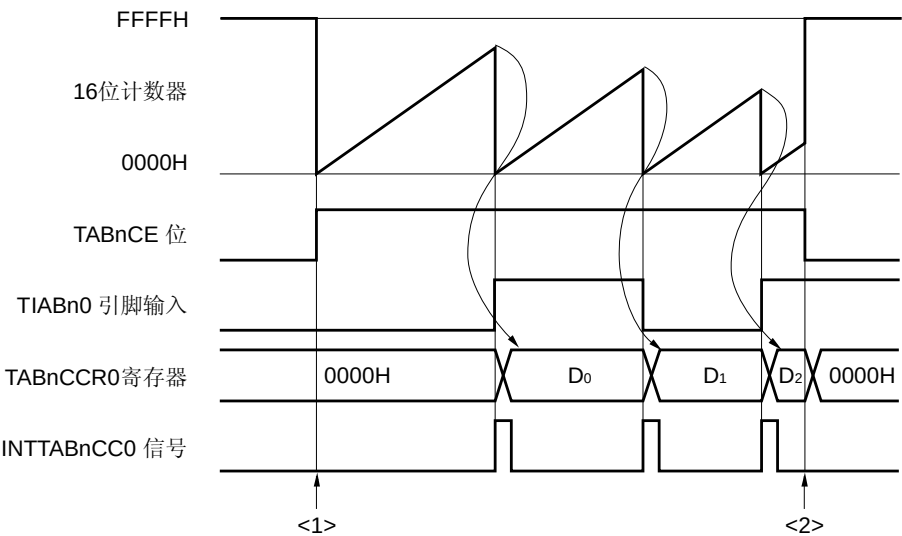
备注 n = 0, 1

图 8-36. 脉冲宽度测量模式时的寄存器设置 (2/2)

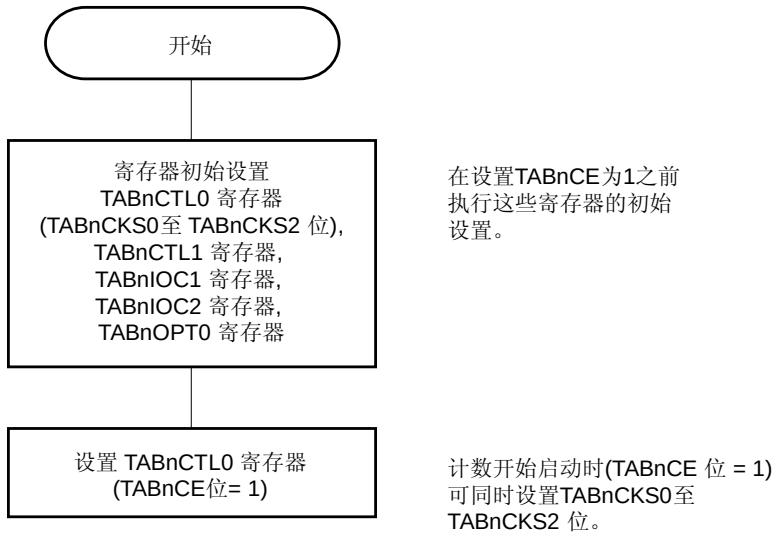


(1) 脉冲宽度测量模式时的工作流程

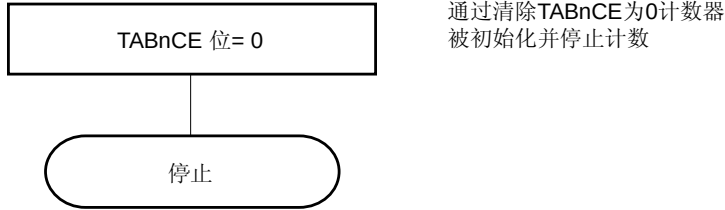
图 8-37. 脉冲宽度测量模式时的软件处理流程



<1> 计数操作开始流程



<2> 计数操作停止流程

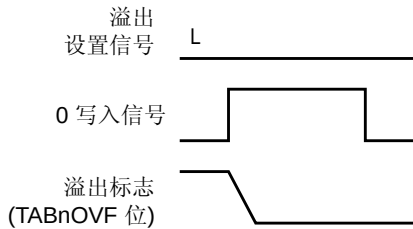


(2) 脉冲宽度测量模式时的工作时序

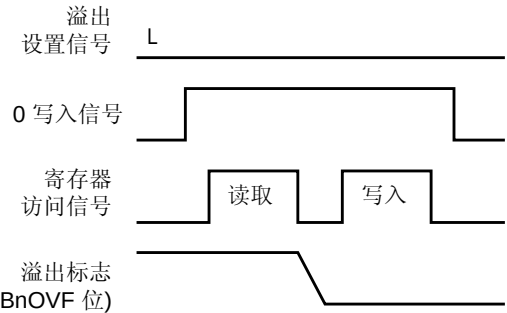
(a) 清除溢出标志

通过用 CLR 指令将 TABnOVF 位清为 0 以及将 8 位数据 (位 0 是 0) 写入 TABnOPT0 寄存器, 可以将溢出标志清 0。要准确检测溢出, 当 TP0OVF 位为 1 时读取 TABnOVF 位, 然后, 用位操作指令将溢出标志清零。

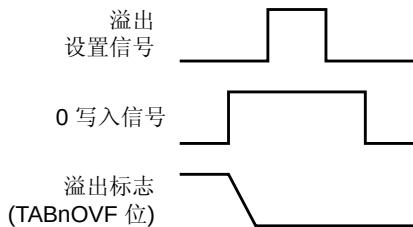
(i) 写入0操作 (与设置没有冲突)



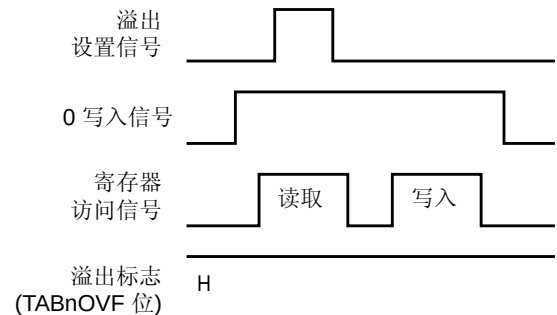
(iii) 清除0操作 (与设置没有冲突)



(ii) 写入0操作 (与设置有冲突)



(iv) 清除0操作 (与设置有冲突)



备注 $n = 0, 1$

要将溢出标志清为 0, 应读取溢出标志, 检查是否被设置为 1, 并用 CLR 指令把它清零。若在未检查溢出标志是否为 1 的情况下把 0 写入溢出标志中, 则溢出的设置信息可能因为写入 0 而被清除 (上图中的 (ii))。因此, 即使实际发生了溢出, 软件也可能会判断为无溢出。

如果用 CLR 指令将溢出标志清为 0 时, CLR 指令的执行与当前发生的溢出相冲突, 则溢出标志即使在 CLR 指令执行后也会保持置位状态。

8.5.8 三角波PWM模式 (TABnMD2 至TABnMD0 位= 111)

在三角波 PWM 模式中, TABn 捕获/比较寄存器 k (TABnCCRk) 用于设置占空比, 而 TABn 捕获/比较寄存器 0 (TABnCCR0) 用于设置周期。

通过使用这四个寄存器以及操作该定时器, 可输出具有可变周期的三角波 PWM。

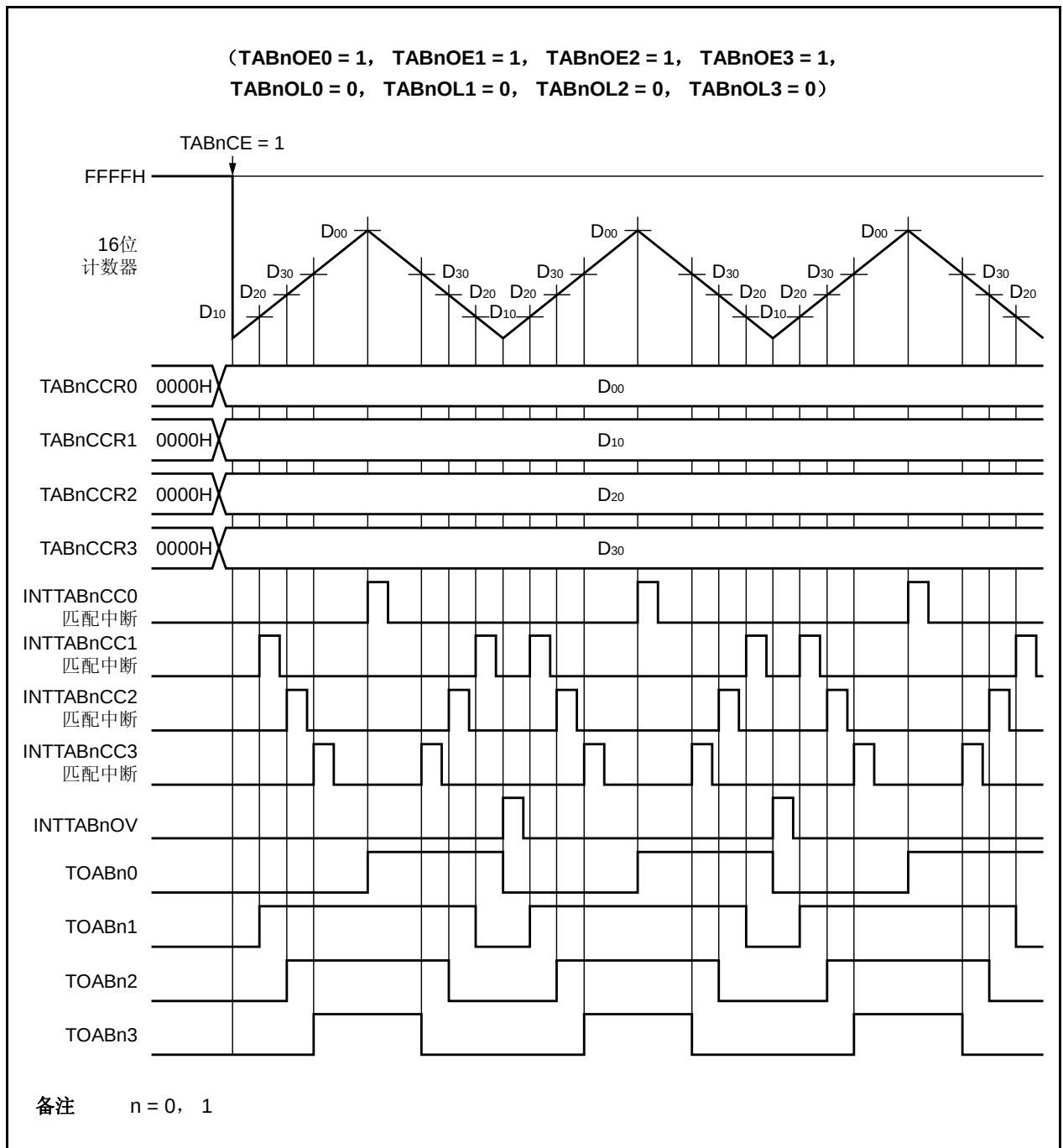
当 TABnCE = 1 时, 可重写 TABnCCRm 寄存器的值。

若要停止定时器 AB, 需将 TABnCE 清除为 0。PWM 波形从 TOABnk 引脚输出。当 16 位计数器的值与 TABnCCR0 寄存器匹配以及计数器向下溢出时, TOABn0 引脚产生一个切换输出。

注意事项 在 PWM 模式下, 因为 TABnCCRm 寄存器只能用作比较寄存器所以不可使用该寄存器的捕获功能。

备注 n = 0, 1; m = 0 至 3; k = 1 至 3

图 8-38. 三角波 PWM 模式下基本操作的时序



8.5.9 定时器输出操作

下表显示了 TOABn0 至 TOABn3 引脚的操作和输出电平。

表 8-6. 各种模式下的定时器输出控制

操作模式	TOABn0 引脚	TOABn1 引脚	TOABn2 引脚	TOABn3 引脚
间隔定时器模式	方波输出			
外部事件计数模式	方波输出	—		
外部触发脉冲输出模式	方波输出	外部触发脉冲输出	外部触发脉冲输出	外部触发脉冲输出
单次脉冲输出模式		单次脉冲输出	单次脉冲输出	单次脉冲输出
PWM 输出模式		PWM 输出	PWM 输出	PWM 输出
自由定时器模式	方波输出（仅当使用比较功能时）			
脉冲宽度测量模式	—			
三角波 PWM 输出模式	方波输出	三角波 PWM 输出	三角波 PWM 输出	三角波 PWM 输出

表 8-7. TOABn0 至 TOABn3 引脚受控于定时器输出控制位的真值表

TABnIOC0.TABnOLm 位	TABnIOC0.TABnOEm 位	TABnCTL0.TABnCE 位	TOABnm 引脚电平
0	0	×	低电平输出
	1	0	低电平输出
		1	计数前为低电平，开始计数后为高电平
1	0	×	高电平输出
	1	0	高电平输出
		1	计数前为高电平，开始计数后为低电平

备注 m = 0 至 3
 n = 0, 1

8.6 定时器调谐操作功能/同时启动功能

定时器 AA 和定时器 AB 有定时器调谐操作功能/同时启动功能。
可同步的定时器列表于表 8-8。

表 8-8. 定时器调谐操作模式

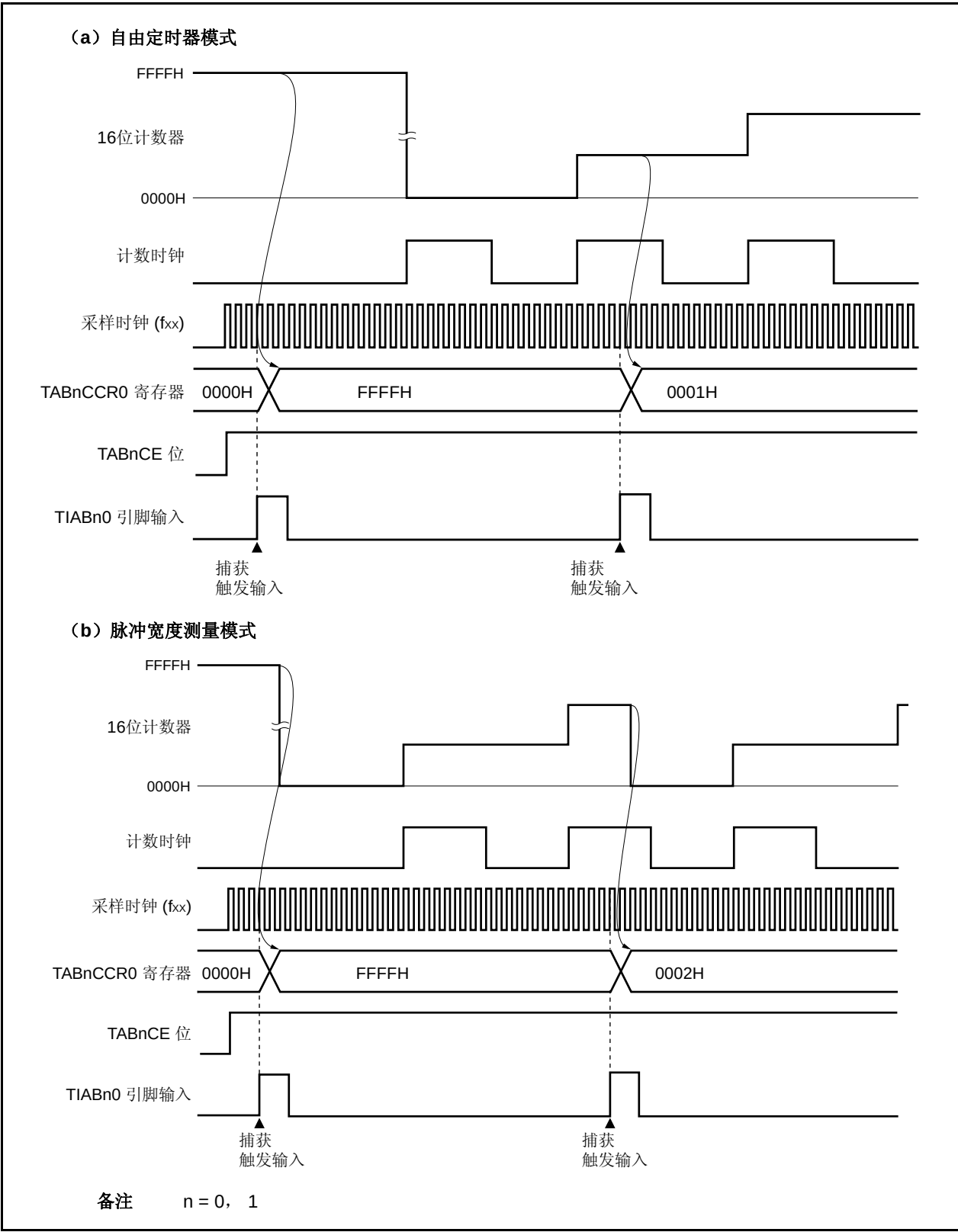
主定时器	从定时器
TAA1	TAA0
TAA3	TAA2
TAB0	TAA5

关于调谐定时器操作功能的详细情况，参见 7.6 定时器调谐操作功能，而关于同时启动功能参见 7.7 同时启动功能。

8.7 注意事项

(1) 捕获操作

当使用捕获操作并且将低速时钟选择为计数时钟时，如果在 TABnCE 位置 1 之后立即输入捕获触发，则被捕获到 TABnCCR0，TABnCCR1，TABnCCR2 和 TABnCCR3 寄存器中的是 FFFFH，而不是 0000H。



第九章 16 位定时器/事件计数器T（TMT）

定时器 T（TMT）是一款 16-位定时器/事件计数器。

在定时器 AA（TAA）上增加了编码器计数功能和其它功能。但是，当其工作在间隔定时器模式中时，TMT 不具有在外部事件计数输入下运行的功能。

V850ES/JG3-H 和 V850ES/JH3-H 有一个 TMT 通道。

9.1 概述

TMT0 的综述如下。

• 时钟选择:	8 种类型
• 捕获触发输入引脚（TIT00、TIT01）:	2
• 外部事件计数输入引脚（EVT00）:	1
• 编码器输入引脚（TENC00、TENC01）:	2
• 编码器清除输入引脚（TECR0）:	1
• 外部触发输入引脚 ^注 :	1
• 定时器计数器:	1
• 捕获/比较寄存器:	2
• 捕获/比较匹配中断请求信号:	2
• 定时器输出引脚:	2

注 外部触发输入引脚和外部事件计数输入引脚（EVT00）与编码器输入引脚（TENC00）共享一个引脚。

9.2 功能

TMT0 的功能如下所示。

- 间隔定时器
- 外部事件计数器
- 外部触发脉冲输出
- 单次脉冲输出
- PWM 输出
- 自由运行定时器
- 脉冲宽度测量
- 三角波 PWM 输出
- 编码器计数

9.3 配置

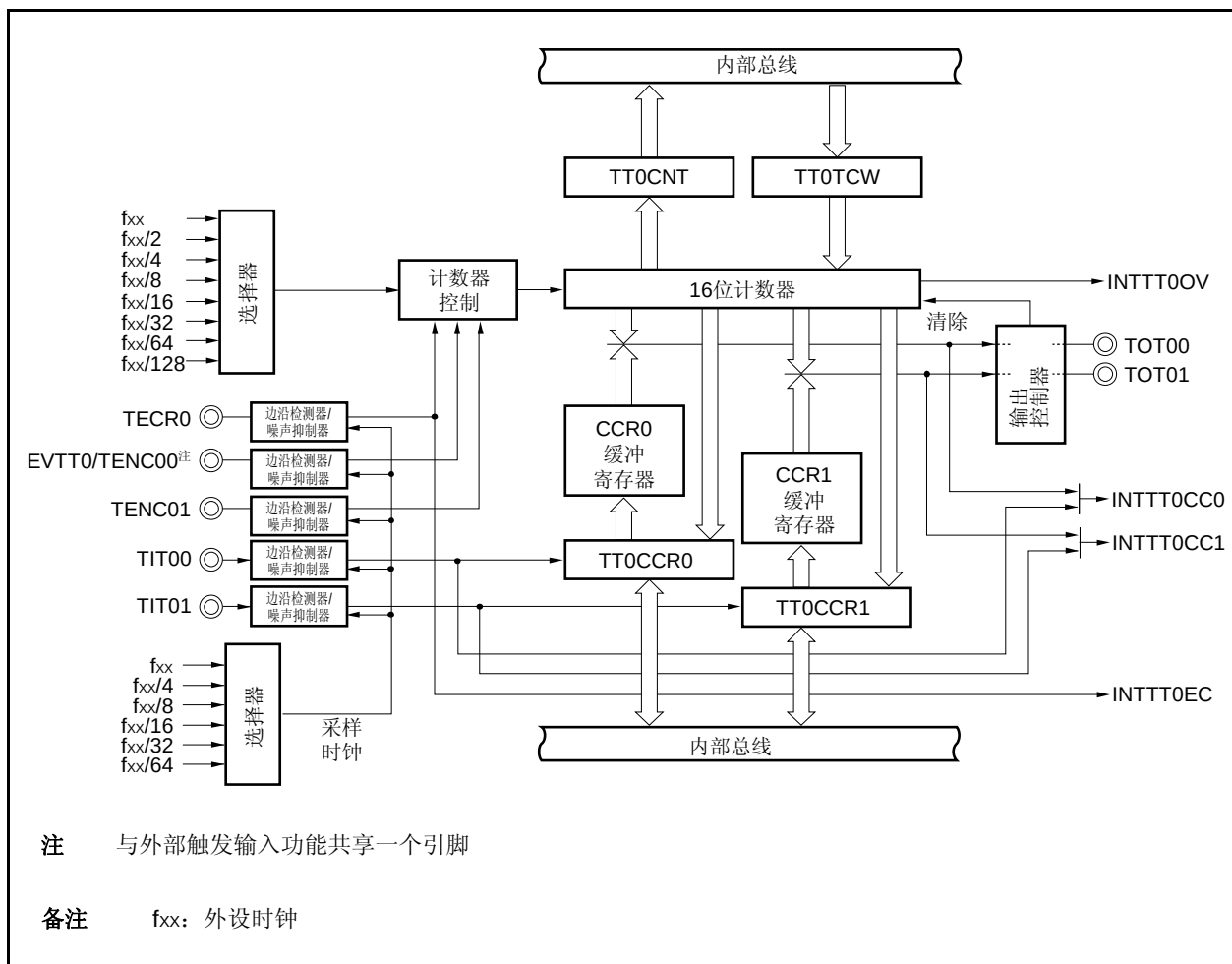
TMT0 包括以下硬件：

表 9-1. TMT0 的配置

项目	配置
寄存器	16 位计数器 × 1 TMT0 捕获/比较寄存器 0、1（TT0CCR0、TT0CCR1） TMT0 计数器读缓冲寄存器（TT0CNT） TMT0 计数器写寄存器（TT0TCW） CCR0、CCR1 缓冲寄存器 TMT0 控制寄存器 0、1（TT0CTL0、TT0CTL1） TMT0 控制寄存器 2（TT0CTL2） TMT0 I/O 控制寄存器 0~3（TT0IOC0~TT0IOC3） TMT0 选项寄存器 0（TT0OPT0） TMT0 选项寄存器 1（TT0OPT1） TMT 噪声抑制控制寄存器（TTNFC）
定时器输入	• TIT00、TIT01（捕获触发输入引脚） • EVTT0/TENC00（外部事件输入/编码器 0 输入引脚）^注 • TENC01（编码器 1 输入引脚） • TENC00（编码器清除输入引脚）
定时器输出	TOT00、TOT01

注 与外部触发输入功能共享一个引脚

图 9-1. TMT0 框图



(1) 16 位计数器

此 16 位计数器可以给内部时钟或外部事件计数。

该计数器的计数值可以通过读取 TT0CNT 寄存器得到。

当 TT0CTL0.TT0CE 位=0 时，16 位计数器的值为 FFFFH。如果此时读取寄存器 TT0CNT，则读到的值为 0000H。

复位后，TT0CE 位置为 0。

(2) CCR0 缓冲寄存器

这是一款 16 位比较寄存器，用于比较 16 位计数器的计数值。

当 TT0CCR0 寄存器用作比较寄存器时，写入 TT0CCR0 寄存器的值会传送到 CCR0 缓冲寄存器。当 16 位计数器的计数值与 CCR0 缓冲寄存器的值匹配时，就会生成一个比较匹配中断请求信号（INTTTCC00）。

不能直接读写 CCR0 缓冲寄存器。

复位后，CCR0 缓冲寄存器置为 0000H，TT0CCR0 寄存器置为 0000H。

(3) CCR1 缓冲寄存器

CCR1 缓冲寄存器为 16 位比较寄存器，用于比较 16 位计数器的计数值。

当 TT0CCR1 寄存器用作比较寄存器时，写入 TT0CCR1 寄存器的值会传送到 CCR1 缓冲寄存器。当 16 位计数器的计数值和 CCR1 缓冲寄存器的值匹配时，也会产生一个比较匹配中断请求信号（INTTTCC01）。

不能直接读写 CCR1 缓冲寄存器。

复位后，CCR1 缓冲寄存器置为 0000H，TT0CCR1 寄存器置为 0000H。

(4) 边沿检测器

该电路检测输入到 TIT00、TIT01、EVTT0/TENC00、TENC01 和 TECR0 引脚的信号的有效边沿。使用 TT0IOC1、TT0IOC2 和 TT0IOC3 寄存器，可以选择无边沿、上升沿、下降沿或上升/下降双边沿来作为有效沿。

(5) 输出控制器

该电路通过 TT0IOC0 寄存器控制 TOT00 和 TOT01 引脚的输出。

(6) 选择器

该选择器选择 16 位计数器的计数时钟。可选择 8 种内部时钟或一个外部事件作为计数时钟。

(7) 计数器控制

由 TT0CTL1 寄存器选择的定时器模式来控制计数操作。

9.3.1 引脚配置

TMT0 配置中定时器的输入和输出共享下列端口。当使用各个引脚时必须设置端口功能（参见表 4-20 端口引脚用作复用功能引脚）。

表 9-2. 引脚配置

端口	定时器输入引脚		定时器输出	其他复用功能
P92	TIT01（捕获触发输入 1）	TENC01（编码器输入）	TOT01	A2 ^{註 1}
P93	TIT00（捕获触发输入 0）	TECR0（编码器清除输入）	TOT00	A3 ^{註 1}
P94	EVT0/TENC00 ^{註 2}			

- 注
- 1. 仅适用于 V850ES/JH3-H
 - 2. 在不能使用端口功能进行控制的状态中，外部事件计数输入（EVT0）、编码器输入（TENC00）和外部触发输入共享一个端口。要使用各个功能，要在设置完相应的端口后，使用 TT0IOC2 和 TT0IOC3 寄存器设置它们。

(2) TMT0 控制寄存器 1 (TT0CTL1)

TT0CTL1 寄存器为 8 位寄存器，控制 TMT0 的操作。

该寄存器可按 8 位或 1 位为单元进行读写。

复位后，该寄存器置为 00H。

(1/2)

复位后：00H R/W 地址： FFFFF601H

	7	6	5	4	3	2	1	0
TT0CTL1	0	TT0EST	TT0EEE	0	TT0MD3	TT0MD2	TT0MD1	TT0MD0

TT0EST	软件触发控制
0	-
1	为外部触发输入生成有效信号。 · 单次脉冲输出模式： 将1写入TT0EST位作为触发输出一个单次脉冲。 · 外部触发脉冲输出模式： 将1写入TT0EST位作为触发输出一个PWM波形。
TT0EST位的读取值恒为0。	

TT0EEE	计数时钟选择
0	禁止使用外部事件计数输入（EVTT0引脚）的操作。 （对由TT0CTL0.TT0CKS0至TT0CTL0.TT0CKS2位选择的计数时钟计数。）
1	允许使用外部事件计数输入（EVTT0引脚）的操作。 （对外部事件计数输入信号（EVTT0引脚）的每个有效沿计数。）
TT0EEE位选择是对内部计数时钟还是对外部事件计数输入的有效沿执行计数。	

TT0MD3	TT0MD2	TT0MD1	TT0MD0	定时器模式选择
0	0	0	0	间隔定时器模式
0	0	0	1	外部事件计数模式
0	0	1	0	外部触发脉冲输出模式
0	0	1	1	单次脉冲输出模式
0	1	0	0	PWM输出模式
0	1	0	1	自由运行定时器模式
0	1	1	0	脉冲宽度测量模式
0	1	1	1	三角波PWM输出模式
1	0	0	0	编码器比较模式
其它				禁止设置

- 注意事项
1. **TT0EST** 位只有在外触发脉冲输出模式或单次脉冲输出模式时有效。在其它模式下，忽略向该位写入 1 的操作。
 2. **TT0EEE** 位只在间隔定时器模式、外部触发脉冲输出模式、单次脉冲输出模式、PWM 输出模式、自由运行定时器模式、脉冲宽度测量模式或三角波 PWM 输出模式时有效。在其它模式下，忽略向该位写入 1 的操作。
 3. 在外部事件计数模式或编码器比较模式中选择外部事件计数输入（**EVTT0**）或编码器输入（**TENC00**、**TENC01**），与 **TT0EEE** 位的值无关。
 4. 当 **TT0CTL0.TT0CE** 位=0 时，设置 **TT0EEE** 位和 **TT0MD3** 至 **TT0MD0** 位。（当 **TT0CE** 位=1 时可写入相同的值。）在 **TT0CE** 位=1 时重写操作是无法保证的。若重写发生错误，应把 **TT0CE** 位清为 0，再重新进行位设定。
 5. 务必将 4 和 7 位设置为“0”。

(3) TMT0 控制寄存器 2 (TT0CTL2)

TT0CTL2 寄存器是 8 位寄存器，用来控制编码器计数功能操作。

TT0CTL2 寄存器只在编码器比较模式中有效。

该寄存器可按 8 位或 1 位为单元进行读写。

复位后，该寄存器置为 00H。

注意事项 有关 TT0CTL2 寄存器各个位的细节，参见 9.6.9 (5) TT0CTL2 寄存器的控制位

(1/2)

复位后: 00H R/W 地址 FFFF602H							
	7	6	5	4	3	2	1 0
TT0CTL2	TT0ECC	0	0	TT0LDE	TT0ECM1	TT0ECM0	TT0UDS1 TT0UDS0
TT0ECC		编码器计数器控制					
0		正常操作					
1		当TT0CTL0.TT0CE位=0时，保持16位计数器的计数值。					
TT0LDE		16位计数器的传输设置					
0		下溢出时，禁止将TT0CCR0位的设置值传送至16位计数器。					
1		下溢出时，允许将TT0CCR0位的设置值传送至16位计数器。					
TT0ECM1		编码器清除操作1的控制					
0		当16位计数器的计数值与CCR1寄存器的值匹配时，16位计数器不清除为0000H。					
1		当计数器计数值和CCR1寄存器的值匹配后向下计数时，16位计数器清除为0000H。					
TT0ECM0		编码器清除操作0的控制					
0		当16位计数器的计数值与CCR0寄存器的值匹配时，16位计数器不清除为0000H。					
1		当计数器计数值和CCR0寄存器的值匹配后向上计数时，16位计数器清除为0000H。					

TT0UDS1	TT0UDS0	上/下计数选择
0	0	检测到 TENC00 输入的有效沿时： 当 TENC01 = 高电平，向下计数。 当 TENC01 = 低电平，向上计数。
0	1	检测到 TENC00 输入的有效沿时，向上计数。 检测到 TENC01 输入的有效沿时，向下计数。
1	0	检测到 TENC00 输入的上升沿时，向下计数。 检测到 TENC00 输入的下降沿时，向上计数。 但是，仅在 TENC01 = 低电平时才执行计数操作。
1	1	检测到 TENC00 和 TENC01 的上升和下降沿时，可通过 配合使用边沿检测和电平检测自动识别计数操作。

注意事项 1. TT0ECC 位只在编码器比较模式中有效。在其它模式下，忽略向该位写入 1 的操作。

当 TT0ECC 位=1 时，如果 TT0CTL0.TT0CE 位清除为 0，那么保留定时器/计数器和捕获寄存器（TT0CCR0 和 TT0CCR1）的值，以及 TT0OPT1、TT0EUF、TT0EOF 和 TT0ESF 标志。

当 TT0ECC=1 时，如果 TT0CE 位的值从 0 变为 1，那么不会将 TT0TCW 寄存器的值传送至 16 位计数器。

2. TT0LDE 位仅在 TT0ECM1 和 TT0ECM0 位=00、01 时有效。当 TT0ECM1 和 TT0ECM0 位=10、11 时，忽略向该位写入“1”的操作。

3. 由 TT0IOC3.TT0EIS1 和 TT0IOC3.TT0EIS0 位指定的 TENC00 和 TENC01 输入的边沿检测有效，并且当 TT0UDS1 和 TT0UDS0 位=10、11 时固定为上升和下降双边沿。

4. 当 TT0CTL0.TT0CE 位=0 时，设置 TT0LDE、TT0ECM1、TT0ECM0、TT0UDS1 和 TT0UDS0 位（当 TT0CE 位=1 时，可向这些位写入相同的值）。如果在 TT0CE 位=1 时更改这些位的值，那么操作将得不到保证。如果是因错误而更改，则清除 TT0CE 位，然后设置正确值。

5. 务必将 5 和 6 位设置为“0”。

(4) TMT0 I/O 控制寄存器 0 (TT0IOC0)

TT0IOC0 寄存器为 8 位寄存器，用于控制定时器的输出 (TOT00、TOT01 引脚)。

该寄存器可按 8 位或 1 位为单元进行读写。

复位后，该寄存器置为 00H。

复位后: 00H R/W 地址: FFFFF603H

	7	6	5	4	3	<2>	1	<0>
TT0IOC0	0	0	0	0	TT0OL1	TT0OE1	TT0OL0	TT0OE0

TT0OL1	TOT01引脚输出电平设置 ^注
0	高电平时, TOT01引脚开始输出。
1	低电平时, TOT01引脚开始输出。

TT0OE1	TOT01引脚输出设置
0	定时器输出禁止 · 当TT0OL1位= 0时, 从TOT01引脚输出低电平。 · 当TT0OL1位= 1时, 从TOT01引脚输出高电平。
1	定时器输出允许 (从TOT01引脚输出一个脉冲。)

TT0OL0	TOT00引脚输出电平设置 ^注
0	高电平时, TOT00引脚开始输出。
1	低电平时, TOT00引脚开始输出。

TT0OE0	TOT00引脚输出设置
0	定时器输出禁止 · 当TT0OL0位=0时, 从TOT00引脚输出低电平。 · 当TT0OL0位=1时, 从TOT00引脚输出高电平。
1	定时器输出允许 (从TOT00引脚输出一个脉冲。)

注 由 TT0OLn 位指定的定时器输出引脚 (TOT00 和 TOT01) 的输出电平如下所示 (n=0、1)。



- 注意事项**
1. 当 TOT00 和 TOT01 输出设置为端口模式时, 如果改变 TT0IOC0 寄存器的设置, 那么引脚的输出也会改变。设置输入模式下的端口, 使端口进入高阻抗状态, 则引脚状态不发生变化。
 2. 当 TT0CTL0.TT0CE 位= 0 时, 重新写入 TT0OL1、TT0OE1、TT0OL0 和 TT0OE0 位。(当 TT0CE 位=1, 可以写入相同的值。)如果重新写入不能正确执行, 将 TT0CE 位清为 0, 然后重新进行位设定。
 3. 当 TT0CE、TT0OE0 和 TT0OE1 位为 0 时, 即使对 TT0OL0 或 TT0OL1 位进行处理, TOT00 和 TOT01 引脚的输出电平也会改变。

(5) TMT0 I/O 控制寄存器 1 (TT0IOC1)

TT0IOC1 寄存器为 8 位寄存器，用于控制捕获触发输入信号 (TIT00、TIT01 引脚) 的有效边沿。
该寄存器可按 8 位或 1 位为单元进行读写。
复位后，该寄存器置为 00H。

复位后: 00H R/W 地址: FFFFF604H

	7	6	5	4	3	2	1	0
TT0IOC1	0	0	0	0	TT0IS3	TT0IS2	TT0IS1	TT0IS0

TT0IS3	TT0IS2	捕获触发输入信号（TIT01引脚）有效边沿设置
0	0	无边沿检测（捕获操作无效）
0	1	上升沿检测
1	0	下降沿检测
1	1	双边沿检测

TT0IS1	TT0IS0	捕获触发输入信号（TIT00引脚）有效边沿设置
0	0	无边沿检测（捕获操作无效）
0	1	上升沿检测
1	0	下降沿检测
1	1	双边沿检测

- 注意事项
1. 在 TT0CTL0.TT0CE 位=0 时才可对 TT0IS3 至 TT0IS0 位进行改写。
(当 TT0CE 位=1 时，可写入相同的值。)如果重新写入不能正确执行，将 TT0CE 位清为 0，然后重新进行位设定。
 2. 只有在自由运行定时器模式 (仅当 TT0OPT0.TT0CCS1 位=1 时) 和脉冲宽度测量模式下，TT0IS3 至 TT0IS2 位才有效。在所有其它模式下，无法执行捕获操作。
只有在自由运行定时器模式 (仅当 TT0OPT0. TT0CCS0 位=1 时) 和脉冲宽度测量模式下，TT0IS1 和 TT0IS0 位才有效。在所有其它模式下，无法执行捕获操作。

(6) TMT0 I/O 控制寄存器 2（TT0IOC2）

TT0IOC2 寄存器为 8 位寄存器，用于控制外部事件计数输入信号（EVTT0 引脚）和外部触发输入信号（EVTTO 引脚）的有效边沿。

该寄存器可按 8 位或 1 位为单元进行读写。

复位后，该寄存器置为 00H。

复位后：00H R/W 地址：FFFFF605H

	7	6	5	4	3	2	1	0
TT0IOC2	0	0	0	0	TT0EES1	TT0EES0	TT0ETS1	TT0ETS0

TT0EES1	TT0EES0	外部事件计数输入信号（EVTT0引脚）有效边沿设置
0	0	无边沿检测（外部事件计数无效）
0	1	上升沿检测
1	0	下降沿检测
1	1	双边沿检测

TT0ETS1	TT0ETS0	外部触发输入信号（EVTTO引脚）有效边沿设置
0	0	无边沿检测（外部触发无效）
0	1	上升沿检测
1	0	下降沿检测
1	1	双边沿检测

- 注意事项**
1. 当 TT0CTL0.TT0CE 位 = 0 时，重新写入 TT0EES1、TT0EES0、TT0ETS1 和 TT0ETS0 位。（当 TT0CE 位 = 1，可以写入相同的值。）如果重新写入不能正确执行，将 TT0CE 位清为 0，然后重新进行位设定。
 2. 只有在 TT0CTL1.TT0EEE 位 = 1 或设置了外部事件计数模式（TT0CTL1.TT0MD3 至 TT0CTL1.TT0MD0 位 = 0001）时，TT0EES1 和 TT0EES0 位才有效。
 3. TT0ETS1 和 TT0ETS0 位仅在外部触发脉冲模式或单次脉冲输出模式时有效。

(7) TMT0 I/O 控制寄存器 3 (TT0IOC3)

TT0IOC3 寄存器是 8 位寄存器，用来控制编码器清除功能操作。

TT0IOC3 寄存器仅在编码器比较模式中有效。

该寄存器可按 8 位或 1 位为单元进行读写。

复位后，该寄存器置为 00H。

(1/2)

复位后：00H R/W 地址：FFFFF606H

	7	6	5	4	3	2	1	0
TT0IOC3	TT0SCE	TT0ZCL	TT0BCL	TT0ACL	TT0ECS1	TT0ECS0	TT0EIS1	TT0EIS0

TT0SCE	编码器清除选择
0	检测到编码器清除信号 (TECR0 引脚) 边沿，清除 16 位计数器。
1	检测到 TENC00、TENC01 和 TECR0 引脚的清除电平状态，清除 16 位计数器。
- 当 TT0SCE 位 = 0，且检测到由 TT0ECS1 和 TT0ECS0 位指定的 TECR0 引脚的有效沿时，将 16 位计数器清除为 0000H。 - 当 TT0SCE 位 = 1，且 TT0ZCL、TT0BCL 和 TT0ACL 位的清除电平状态与 TECR0、TENC01 和 TENC00 引脚的输入电平匹配时，将 16 位计数器清除为 0000H。 - 当 TT0SCE 位 = 1 时，TT0ZCL、TT0BCL 和 TT0ACL 位的设置有效，TT0ECS1 和 TT0ECS0 位无效。 不生成编码器清除中断请求信号 (INTTTI0EC)。 - 当 TT0SCE 位 = 0 时，TT0ZCL、TT0BCL 和 TT0ACL 位的设置无效，TT0ECS1 和 TT0ECS0 位的设置有效。 - 当检测到由 TT0ECS1 和 TT0ECS0 位指定的有效沿时，生成 INTTTI0EC 信号。 - 当 TT0SCE 位 = 1 时，务必将 TT0CTL2.TT0UDS1 和 TT0CTL2.TT0UDS0 位设置为 10 或 11。 - 如果 TT0UDS1 和 TT0UDS0 位 = 00 或 01，且 TT0SCE 位 = 1 时，操作将得不到保证。	

TT0ZCL	编码器清除信号 (TECR0 引脚) 的清除电平选择
0	清除 TECR0 引脚的低电平。
1	清除 TECR0 引脚的高电平。
仅当 TT0SCE 位 = 1 时 TT0ZCL 的设置有效。	

TT0BCL	编码器输入信号 (TENC01 引脚) 的清除电平选择
0	清除 TENC01 引脚的低电平。
1	清除 TENC01 引脚的高电平。
仅当 TT0SCE 位 = 1 时 TT0BCL 位的设置有效。	

TT0ACL	编码器输入信号 (TENC00 引脚) 的清除电平选择
0	清除 TENC00 引脚的低电平。
1	清除 TENC00 引脚的高电平。
仅当 TT0SCE 位 = 1 时 TT0ACL 位的设置有效。	

TT0ECS1	TT0ECS0	编码器清除信号（TECR0引脚）的有效沿设置
0	0	无边沿检测（清除编码器无效）。
0	1	检测上升沿。
1	0	检测下降沿。
1	1	检测双边沿。

TT0EIS1	TT0EIS0	编码器输入信号（TENC00、TENC01引脚）的有效沿设置
0	0	无边沿检测（输入编码器无效）。
0	1	检测上升沿。
1	0	检测下降沿。
1	1	检测双边沿。

- 注意事项**
1. 当 TT0CTL0.TT0CE 位 = 0 时，重写 TT0SCE、TT0ZCL、TT0BCL、TT0ACL、TT0ECS1、TT0ECS0、TT0EIS1 和 TT0EIS0 位。（当 TT0CE 位 = 1 时，可向这些位写入相同的值）如果重新写入不能正确执行，将 TT0CE 位清为 0，然后重新进行位设定。
 2. 仅当 TT0SCE 位 = 0 且设置为编码器比较模式时 TT0ECS1 和 TT0ECS0 位有效。
 3. 仅当 TT0CTL2.TT0UDS1 和 TT0CTL2.TT0UDS0 位 = 00 或 01 时，TT0EIS1 和 TT0EIS0 位有效。

(8) TMT0 选项寄存器 0（TT0OPT0）

TT0OPT0 寄存器为 8 位寄存器，用来设置捕获/比较操作和检测溢出。

该寄存器可按 8 位或 1 位为单元进行读写。

复位后，该寄存器置为 00H。

复位后: 00H R/W 地址: FFFF607H

	7	6	5	4	3	2	1	<0>
TT0OPT0	0	0	TT0CCS1	TT0CCS0	0	0	0	TT0OVF

TT0CCS1	TT0CCR1 寄存器捕获/比较选择
0	选作为比较寄存器
1	选作为捕获寄存器（TT0CTL0.TT0CE位= 0时清除）
仅在自由运行定时器模式中TT0CCS1位的设置有效。	

TT0CCS0	TT0CCR0 寄存器捕获/比较选择
0	选作为比较寄存器
1	选作为捕获寄存器（TT0CTL0.TT0CE位= 0时清除）
仅在自由运行定时器模式中TT0CCS0位的设置有效。	

TT0OVF	TMT0 上溢出检测标志
设置 (1)	发生上溢出
复位 (0)	在TT0OVF位写入0或 TT0CTL0.TT0CE位 = 0
· 在自由运行定时器模式或脉冲宽度测量模式中，当16位计数器的计数值由FFFFH变为0000H发生溢出时，TT0OVF位置为1。 · 当TT0OVF位置为1时，生成溢出中断请求信号（INTTT0OV）。在除自由运行定时器模式和脉冲宽度测量模式之外的任何模式中，不生成INTTT0OV 信号。 · 即使当TT0OVF=1，且读取TT0OVF 位或TT0OPT0寄存器时，TT0OVF位不清除为0。 · 生成 INTTT0OV 信号后，清除TT0OVF位为0之前，务必确认（通过读取）TT0OVF位置为1。 · TT0OVF位可读取也可写入，但是TT0OVF位不能用软件设置为1。写入1对TMT0的操作没有影响。	

- 注意事项
1. 当 TT0CE 位=0 时，重新写入 TT0CCS1 和 TT0CCS0 位。（当 TT0CE 位=1 时，可写入相同值。）如果重新写入不能正确执行，将 TT0CE 位清为 0，然后重新进行位设定。
2. 务必将位 1 至 3、6 和 7 设置为“0”。

(9) TMT0 选项寄存器 1 (TT0OPT1)

TT0OPT1 寄存器是 8 位寄存器，用来检测上溢出、下溢出和编码器计数功能的向上/下计数操作。

TT0OPT1 寄存器仅在编码器比较模式中有效。

该寄存器可按 8 位或 1 位为单元进行读写。

复位后，该寄存器置为 00H。

即便在 TT0CTL0.TT0CE 位=1 时，该寄存器仍可以重新写入。

(1/2)

复位后: 00H R/W 地址: FFFFF608H

	7	6	5	4	3	<2>	<1>	<0>
TT0OPT1	0	0	0	0	0	TT0EUF	TT0EOF	TT0ESF

TT0EUF	TMT0下溢出检测标志
设置 (1)	发生下溢出。
复位 (0)	通过写入TT0EUF位或当TT0CTL0.TT0CE位= 0时清除。
· 当16位计数器在编码器比较模式中从0000H变为FFFFH发生下溢出时，TT0EUF位置为 1。 · 当TT0CTL2.TT0LDE位 = 1，且16位计数器的值从0000H变到TT0CCR0寄存器的设置值时，TT0EUF位置为1。 · 一将TT0EUF位置为1就会生成上溢出中断请求信号（INTTTOV0）。 · 即使当TT0EUF位 = 1时读取TT0EUF位或 TT0OPT1寄存器，TT0EUF位也不清除为0。 · 当TT0CTL2.TT0ECC位 = 1时，即使将TT0CTL0.TT0CE位清除为0，TT0EUF位的状态也会保留。 · 生成INTTTOV0信号后，在清除TT0EUF位为0之前，务必确认（通过读取）TT0EUF位设置为 1。 · TT0EUF位可读取也可写入，但是不能通过软件将其设置为1。 设置该位为 1对TMT0的操作没有影响。	

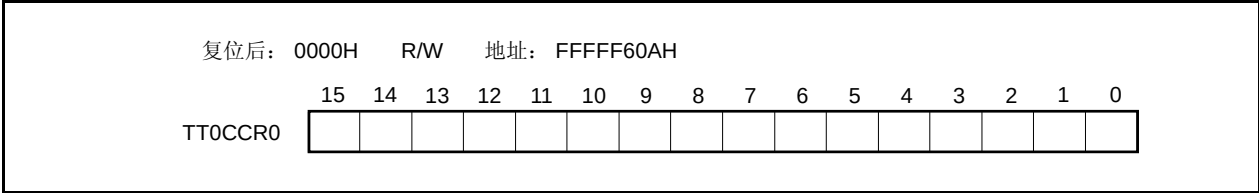
TT0EOF	TMT0编码器功能的 上溢出检测标志
设置(1)	发生上溢出。
复位(0)	通过对TT0EOF位写入0或当TT0CTL0.TT0CE位=0时清除。
- 当16位计数器在编码器比较模式中从FFFFH变为0000H发生上溢出时，TT0EOF位设置为1。 - 在TT0EOF位设置为1的同时会生成上溢出中断请求信号（INTTTOV0）。此时，TT0OPT0.TT0OVF位不置为1。 - 当TT0EOF位 = 1时，即使读取TT0EOF位或TT0OPT1寄存器，TT0EOF位也不清除为0。 - 当TT0CTL2.TT0ECC 位 = 1时，即使 TT0CTL0.TT0CE位清除为0，TT0EOF 位的状态也会保留。 - 生成INTTTOV0信号后，在清除TT0EOF位为0之前，务必确认（通过读取）TT0EOF位设置为 1。 - TT0EOF位可读取也可写入，但是不能用软件将其设置为1。在该位写入1对TMT0的操作没有影响。	

TT0ESF	TMT0 向上/下计数操作状态检测标志
0	TMT0正向上计数。
1	TMT0正向下计数。
- 当TT0CTL2.TT0ECC位 = 0时，如果TT0CTL0.TT0CE位 = 0，该位清除为0。 - 当TT0ECC位 = 1时，即使TT0CE位 = 0，也会保留TT0ESF位的状态。	

注意事项 务必将 3 至 7 位设置为“0”。

(10) TMT0 捕获/比较寄存器 0（TT0CCR0）

TT0CCR0 寄存器是一个 16 位寄存器，根据不同工作模式可以用作捕获寄存器或比较寄存器。
在自由运行定时器模式下，依据 TT0OPT0.TT0CCS0 位的设定，该寄存器可用作捕获寄存器或比较寄存器。
在脉宽测量模式中，TT0CCR0 寄存器只可以用作捕获寄存器。而在其它模式时，该寄存器只可以用作比较寄存器。
TT0CCR0 寄存器可以在操作期间进行读取和写入。
该寄存器可按 16 位为单位进行读写。
复位后本寄存器设置为 0000H。



(a) 作为比较寄存器时的功能

即便在 TT0CTL0.TT0CE 位=1 时，TT0CCR0 寄存器仍可以重新写入。

TT0CCR0 寄存器的设定值传送给 CCR0 缓冲寄存器。当 16 位计数器的值与 CCR0 缓冲寄存器的值相匹配时，就会生成一个比较匹配中断请求信号（INTTT0CC0）。如果此时允许 TOT00 引脚输出，则 TOT00 引脚的输出会反相。

当 TT0CCR0 寄存器在间隔定时器模式中作为循环寄存器时，或者在外部事件计数器模式、外部触发脉冲输出模式、单次脉冲输出模式、PWM 输出模式、三角波 PWM 输出模式或编码器比较模式中 TT0CCR0 寄存器作为循环寄存器时，如果 16 位计数器的计数值与 CCR0 缓冲寄存器的值匹配，则清除计数器（0000H）。

通过设置 TT0CTL0.TT0CE 位=0，不能清除比较寄存器。

(b) 作为捕获寄存器时的功能

在自由运行定时器模式（当 TT0CCR0 寄存器用作捕获寄存器时）下，若检测到捕获触发输入引脚（TIT00 引脚）的有效边沿，则 16 位计数器的计数值存储于 TT0CCR0 寄存器中。在脉宽测量模式下，若检测到捕获触发输入引脚（TIT00 引脚）的有效边缘，则 16 位计数器的计数值就储存在 TT0CCR0 寄存器中，且清除 16 位计数器（0000H）。

即使捕获操作与对 TT0CCR0 寄存器的读取发生冲突，还是可以正确读取 TT0CCR0 寄存器的值。

通过设置 TQ0CTL0.TQ0CE 位为 0，可以清除捕获寄存器。

下表显示了捕获/比较寄存器在各工作模式下的功能，以及怎样向比较寄存器写入数据。

表 9-3. 捕获/比较寄存器在各模式下的功能和如何写入比较寄存器

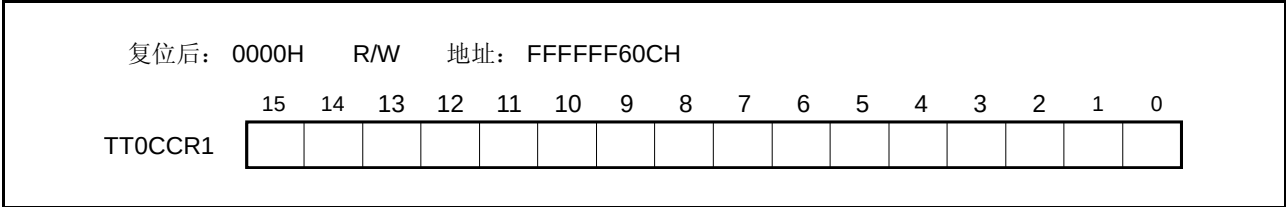
操作模式	TT0CCR0 寄存器	如何写入比较寄存器
间隔定时器	比较寄存器	可随时写入
外部事件计数器	比较寄存器	可随时写入
外部触发脉冲输出	比较寄存器	批写入 ^注
单次脉冲输出	比较寄存器	可随时写入
PWM 输出	比较寄存器	批写入 ^注
自由运行定时器	捕获/比较寄存器	可随时写入
脉冲宽度测量	捕获寄存器	无
三角波 PWM 输出	比较寄存器	批写入 ^注
编码器比较	比较寄存器	可随时写入

注 写入 TT0CCR1 寄存器的是触发脉冲。

备注 关于随时写入和批写入的详情，参见 9.6（2）随时写入和批写入。

(11) TMT0 捕获/比较寄存器 1（TT0CCR1）

TT0CCR1 寄存器是一个 16 位寄存器，根据不同工作模式可以用作捕获寄存器或比较寄存器。
依据 TT0OPT0.TT0CCS1 位的设定，仅在自由运行定时器模式下，该寄存器可用作捕获寄存器或比较寄存器。在脉宽测量模式中，TT0CCR1 寄存器仅能用作捕获寄存器。而在其它模式时，该寄存器只可以用作比较寄存器。
TT0CCR1 寄存器可以在操作期间进行读取和写入。
该寄存器可按 16 位为单位进行读写。
复位后，该寄存器置为 0000H。



(a) 作为比较寄存器时的功能

即便在 TT0CTL0.TT0CE 位=1 时，TT0CCR1 寄存器仍可以重新写入。

TT0CCR1 寄存器的设定值传送给 CCR1 缓冲寄存器。当 16 位计数器的值与 CCR1 缓冲寄存器的值相匹配时，就会生成一个比较匹配中断请求信号（INTTT0CC01）。如果此时允许 TOT01 引脚输出，则 TOT01 引脚的输出会反相。

通过设置 TQnCTL0.TQnCE 位为 0，不能清除比较寄存器。

(b) 作为捕获寄存器时的功能

在自由运行定时器模式（当 TT0CCR1 寄存器用作捕获寄存器时）下，若检测到捕获触发输入引脚（TIT01 引脚）的有效边缘，则 16 位计数器的计数值存储于 TT0CCR1 寄存器中。在脉宽测量模式下，若检测到捕捉触发输入引脚（TIT01 引脚）的有效边缘，16 位计数器的计数值就储存在 TT0CCR1 寄存器中，且清除 16 位计数器（0000H）。

即使捕获操作与对 TT0CCR1 寄存器的读取发生冲突，还是可以正确读取 TT0CCR1 寄存器的值。

通过设置 TQ0CTL0.TQ0CE 位为 0，清除捕获寄存器。

下表显示了捕获/比较寄存器在各工作模式下的功能，以及怎样向比较寄存器写入数据。

表 9-4. 捕获/比较寄存器在各模式下的功能和如何写入比较寄存器

操作模式	TT0CCR1 寄存器	如何写入比较寄存器
间隔定时器	比较寄存器	可随时写入
外部事件计数器	比较寄存器	可随时写入
外部触发脉冲输出	比较寄存器	批写入 ^注
单次脉冲输出	比较寄存器	可随时写入
PWM（脉宽调制信号）输出	比较寄存器	批写入 ^注
自由运行定时器	捕获/比较寄存器	可随时写入
脉冲宽度测量	捕获寄存器	无
三角波 PWM 输出	比较寄存器	批写入 ^注
编码器比较	比较寄存器	可随时写入

注 写入 TT0CCR1 寄存器的是触发脉冲。

备注 关于随时写入和批写入的详情，参见 9.6（2）随时写入和批写入。

(12) TMT0 计数器写寄存器 (TT0TCW)

TT0TCW 寄存器用来设置 16 位计数器的初始值。

TT0TCW 寄存器仅在编码器比较模式中有效。

该寄存器可按 16 位为单位进行读写。

当 TT0CTL0.TT0CE 位=0 时，重新写入 TT0TCW 寄存器。

当 TT0CE 位设置为 (1) 时，TT0TCW 寄存器的值传送至 16 位计数器。

复位后，该寄存器置为 0000H。

复位后: 0000H R/W 地址: FFFFF610H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TT0TCW																

(13) TMT0 计数器读缓冲寄存器 (TT0CNT)

TT0CNT 寄存器是读缓冲寄存器，可以读取 16 位计数器的计数值。

如果在 TT0CTL0.TT0CE 位=1 时读取该寄存器，则可以读出 16 位定时器的计数值。

本寄存器为只读寄存器，可以按 16 位为单位读取。

当 TT0CTL2.TT0ECC 和 TT0CE 位=0 时，TT0CNT 寄存器的值置为 0000H。如果此时读取 TT0CNT 寄存器，则不会读出 16 位计数器的值 (FFFFH)，而是读取值 0000H。当 TT0ECC 位=1 且 TT0CE 位=0 时，TT0CNT 寄存器不会置为 0000H，而是读取其先前值。

复位后，将 TT0ECC 和 TT0CE 位置为 0，并将 TT0CNT 寄存器的值置为 0000H。

复位后: 0000H R 地址: FFFFFFF60EH

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TT0CNT																

(14) 噪声抑制控制寄存器 (TTNFC)

可以为 TIT00、TIT01、TENC01、TECR0 和 EVTT00 引脚选择数字噪声抑制。使用 TTNFC 寄存器来执行噪声抑制设置。

当选择数字噪声抑制时，数字采样的采样时钟可从以下范围选择：f_{xx}、f_{xx}/4、f_{xx}/8、f_{xx}/16、f_{xx}/32 和 f_{xx}/64。执行 3 次采样。

该寄存器可按 8 位为单位进行读写。

复位后，该寄存器置为 00H。

注意事项 在采样时钟改变后，需要 3 个采样时钟来初始化数字噪声抑制。在采样时钟改变后，经过 3 个采样时钟前，如果输入 TIT00、TIT01、TENC01、TECR0 和 EVTT00 引脚的有效边沿，那么，就可能生成中断请求信号。因此，当使用 TMT 的外部触发功能、外部事件功能、捕获触发功能和编码器功能时，在经过 3 个采样时钟后允许 TMT 操作。

复位后： 00H R/W 地址： FFFFF726H

	7	6	5	4	3	2	1	0
TTNFC	TTNFEN	0	0	0	0	TTNFC2	TTNFC1	TTNFC0

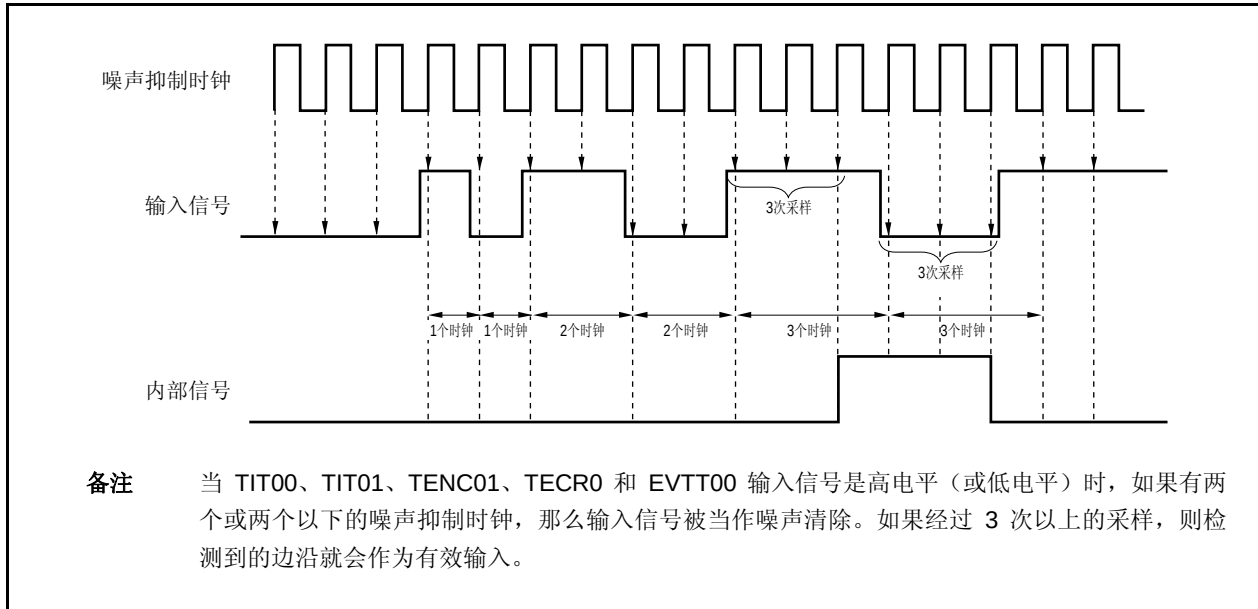
TTNFEN	数字噪声抑制的设置
0	不执行数字噪声抑制
1	执行数字噪声抑制

TTNFC2	TTNFC1	TTNFC0	数字采样时钟
0	0	0	f _{xx}
0	0	1	f _{xx} /4
0	1	0	f _{xx} /8
0	1	1	f _{xx} /16
1	0	0	f _{xx} /32
1	0	1	f _{xx} /64
其他			禁止设置

- 备注**
1. 因为执行了 3 次采样，消除噪声的可靠宽度是 2 个采样时钟。
 2. 在噪声宽度小于 2 个采样时钟的情况下，如果输入与采样时钟同步的噪声，就会生成一个中断请求信号。

图 9-2 是一个由定时器 T 输入引脚数字滤波器执行的噪声抑制的时序示例。

图 9-2 数字噪声抑制时序举例



9.5 定时器输出操作

下表显示了 TOT00 和 TOT01 引脚的操作和输出电平。

表 9-5. 各种模式下的定时器输出控制

操作模式	TOT01 引脚		TOT00 引脚
间隔定时器模式	输出方波		
外部事件计数模式	无		
外部触发脉冲输出模式	输出外部触发脉冲	输出方波	
单次脉冲输出模式	输出单次脉冲		
PWM 输出模式	输出 PWM		
自由运行定时器模式	输出方波（仅当用于比较功能时）		
脉冲宽度测量模式	无		
三角波 PWM 输出模式	输出三角波 PWM		
编码器比较模式	无		

表 9-6. TOT00 和 TOT01 引脚受控于定时器输出控制位的真值表

TT0IOC0.TT0OLn 位	TT0IOC0.TT0OEn 位	TT0CTL0.TT0CE 位	TOT0n 引脚的电平
0	0	×	低电平输出
	1	0	低电平输出
		1	计数前为低电平，开始计数后为高电平
1	0	×	高电平输出
	1	0	高电平输出
		1	计数前为高电平，开始计数后为低电平

备注 n=0、1

9.6 操作

TMT0 的功能在不同的通道实现方法不同。各通道的功能如下所示。

表 9-7. 各个模式中 TMT0 的规范

操作	TT0CTL1.TT0EST 位 (软件触发位)	EVTTO 引脚 (外部触发输入)	捕获/比较寄存器设置	比较寄存器写入方式
间隔定时器模式	无效	无效	仅比较寄存器有效	可随时写入
外部事件计数模式	无效	无效	仅比较寄存器有效	可随时写入
外部触发脉冲输出模式	有效	有效	仅比较寄存器有效	批写入
单次脉冲输出模式	有效	有效	仅比较寄存器有效	可随时写入
PWM 输出模式	无效	无效	仅比较寄存器有效	批写入
自由运行定时器模式	无效	无效	可变换的	可随时写入
脉冲宽度测量模式	无效	无效	仅捕获寄存器有效	无效
三角波 PWM 输出模式	无效	无效	仅比较寄存器有效	批写入
编码器比较模式	无效	无效	仅比较寄存器有效	可随时写入

(1) 基本的计数器操作

这部分介绍了 16 位计数器的基本操作。有关详细内容，参考各个模式中操作的描述。

(a) 计数开始操作

- **编码器比较模式**

由 TENC00 和 TENC01 的状态控制着计数操作。

通过将 TT0TCW 寄存器的设置值传送至 16 位计数器执行 16 位计数器的初始设置，并开始计数操作。

(当 TT0CTL2.TT0ECC 位=1 时，在 TT0CTL0.TT0CE 位从 0 变为 1 的同时 TT0TCW 寄存器的设置值传送至 16 位计数器。)

- **三角波 PWM 模式**

16 位计数器从初始值 FFFFH 开始计数。

它向上计数：FFFFH、0000H、0001H、0002H、0003H，等等。

向上计数操作之后，16 位计数器的计数值与 CCR0 缓冲寄存器匹配时计数器向下计数。

- **其他模式**

16 位计数器从初始值 FFFFH 开始计数。

它向上计数：FFFFH、0000H、0001H、0002H、0003H，等等。

(b) 清除模式

当 16 位计数器的值与比较寄存器的值匹配、当捕获到计数器的值、当检测到编码器清除信号的边沿、当检测到 TENC00、TENC01 和 TECR0 引脚的清除电平状态时，计数器清除为 0000H。在紧接计数器开始计数后或当计数器上溢出时，所发生的从 FFFFH 至 0000H 的计数操作不是清除操作。因此，不会生成 INTTT0CC0 和 INTTT0CC1 中断信号。

(c) 上溢出操作

在自由运行模式、脉宽测量模式和编码器比较模式中，计数器从 FFFFH 向上计数至 0000H 时，16 位计数器将会发生上溢出。如果在自由运行模式和脉冲宽度测量模式中计数器发生上溢出，那么 TT0OPT0.TT0OVF 位设置为 1，并生成一个中断请求信号 (INTTT0OV)。

如果在编码器比较模式中计数器发生上溢出，则 TT0OPT1.TT0EOF 位将会被置为 1，并生成一个中断请求信号 (INTTT0OV)。

需注意的是，INTTT0OV 信号不会在以下条件中生成。

- 计数操作开始后的瞬间
- 如果计数器的值与比较值 FFFFH 匹配并被清除
- 在脉冲宽度测量模式中捕获到 FFFFH 值并清除为 0000H 时

注意事项 生成上溢出中断请求信号 (INTTT0OV) 后，务必检查溢出标志 (TT0OVF、TT0EOF 位) 已置为 1。

(d) 计数值保持操作

在编码器比较模式中，通过 TT0CTL2.TT0ECC 位可以保持 16 位计数器的值。当 TT0ECC 位=0 且 TT0CTL0.TT0CE 位=0 时，16 位计数器的值复位为 FFFFH。当紧接着 TT0CE 位置为 1 时，TT0TCW 寄存器的设置值传送至 16 位计数器，并执行计数操作。

如果 TT0ECC 位=1 且 TT0CE 位=0，则保持 16 位计数器的值。当紧接着 TT0CE 位也置为 1 时，计数器从保持值重新开始计数操作。

(e) 计数操作期间计数器读取操作

计数操作期间，TMT0 的 16 位计数器的值可以通过使用 TT0CNT 寄存器来读取。

当 TT0CTL0.TT0CE 位=1 时，可以通过读取 TT0CNT 寄存器，读取 16 位计数器的计数值。但是，如果在 TT0CTL2.TT0ECC 位=0 且 TT0CE 位=0 时读取 TT0CNT 寄存器，就会读到 0000H 值。当 TT0ECC 位=1 且 TT0CE 位=0 时，如果读取寄存器，则会读取 TT0CNT 寄存器的保持值。

(f) 下溢出操作

在编码器比较模式中，16 位计数器的值从 0000H 变为 FFFFH 的同时，16 位计数器发生下溢出。当发生下溢出时，TT0OPT1.TT0EUF 位将会置为 1，并生成一个中断请求信号（INTTT0OV）。

(g) 中断操作

TMT0 生成以下四种类型的中断请求信号。

- INTTT0CC0 中断：该信号的功能相当于 CCR0 缓冲寄存器的匹配中断请求信号以及 TT0CCR0 寄存器的捕获中断请求信号。
- INTTT0CC1 中断：该信号的功能相当于 CCR1 缓冲寄存器的匹配中断请求信号以及 TT0CCR1 寄存器的捕获中断请求信号。
- INTTT0OV 中断：该信号的功能相当于一个上溢出中断请求信号。
- INTTT0EC 中断：该信号的功能相当于编码器清除输入的一个有效沿检测中断请求信号（TECR0 引脚）。

(2) 随时写入和批写入

在定时器操作 (TT0CTL0.TT0CE 位=1) 期间, TMT0 中的 TT0CCR0 和 TT0CCR1 寄存器可以被重写, 但 CCR0 和 CCR1 缓冲寄存器的写入方法 (随时写入, 批写入) 会根据模式的不同而变化。

(a) 随时写入

在该模式中, 在定时器操作期间 (n=0、1) 数据可以随时从 TT0CCR0 和 TT0CCR1 寄存器传送到 CCR0 和 CCR1 缓冲寄存器中。

图 9-3. 随时写入的基本操作流程

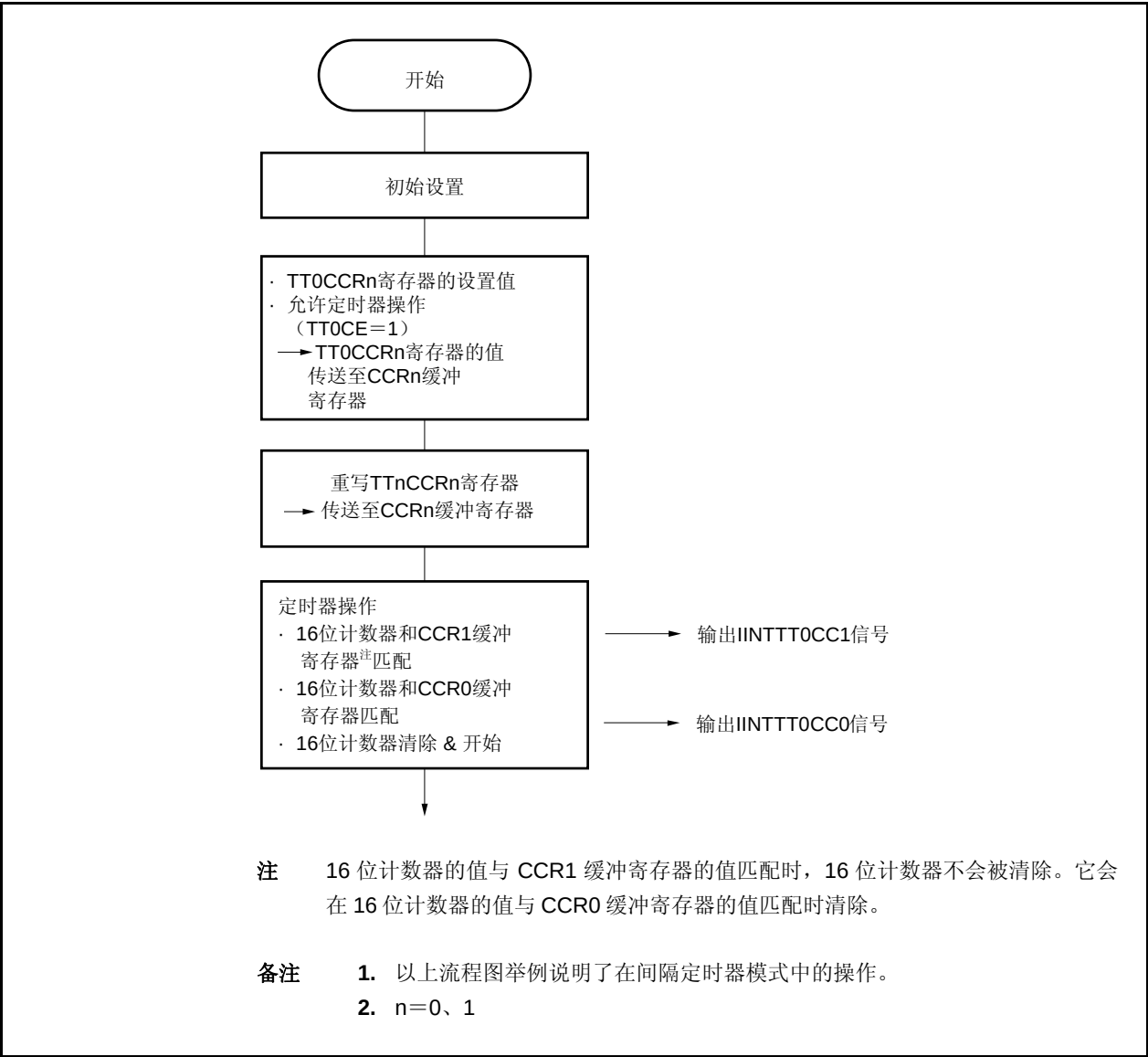
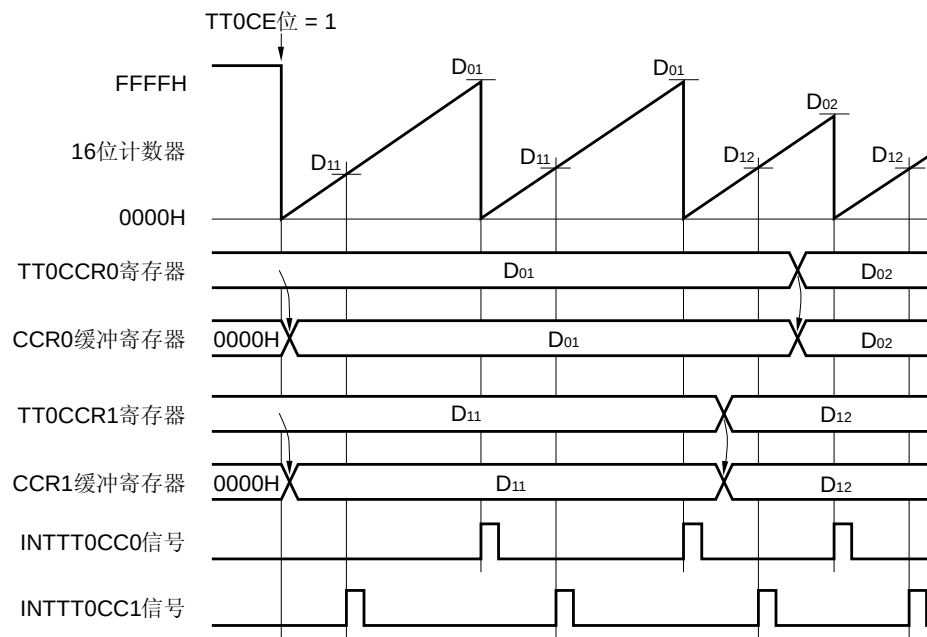


图 9-4. 随时写入的时序



- 备注**
1. D01、D02: TMTCCR0 寄存器的设置值
D11、D12: TMTCCR1 寄存器的设置值
 2. 上面的时序图举例说明了在间隔定时器模式中的操作。
 3. n=0、1

(b) 批写入

在该模式中，在定时器操作期间所有数据会立即从 TT0CCR0 和 TT0CCR1 寄存器传送到 CCR0 和 CCR1 缓冲寄存器中。此数据会在 CCR0 缓冲寄存器的值与 16 位计数器的值匹配时传输。通过写入 TT0CCR1 寄存器允许传输。允许还是禁止下个转移时序是通过对 TT0CCR1 寄存器执行或不执行写入操作来控制的。

为了在改写 TT0CCR0 和 TT0CCR1 寄存器时将设定值转变为 16 位计数器的比较值（换句话说，为了让该值传送至 CCR0 和 CCR1 缓冲寄存器中），在 16 位计数器的值与 CCR0 寄存器的值匹配前，需要先对 TT0CCR0 寄存器进行改写，然后再对 TT0CCR1 寄存器进行写入操作。因此，一旦 16 位计数器的计数值与 CCR0 缓冲寄存器的值匹配，TT0CCR0 和 TT0CCR1 寄存器的值将会传送至 CCR0 和 CCR1 缓冲寄存器中。因此即使只想改写 TT0CCR0 寄存器的值，相同的值（与 TT0CCR1 寄存器的预置值相同）也会被写入到 TT0CCR1 寄存器中。

图 9-5. 批写入的基本操作流程

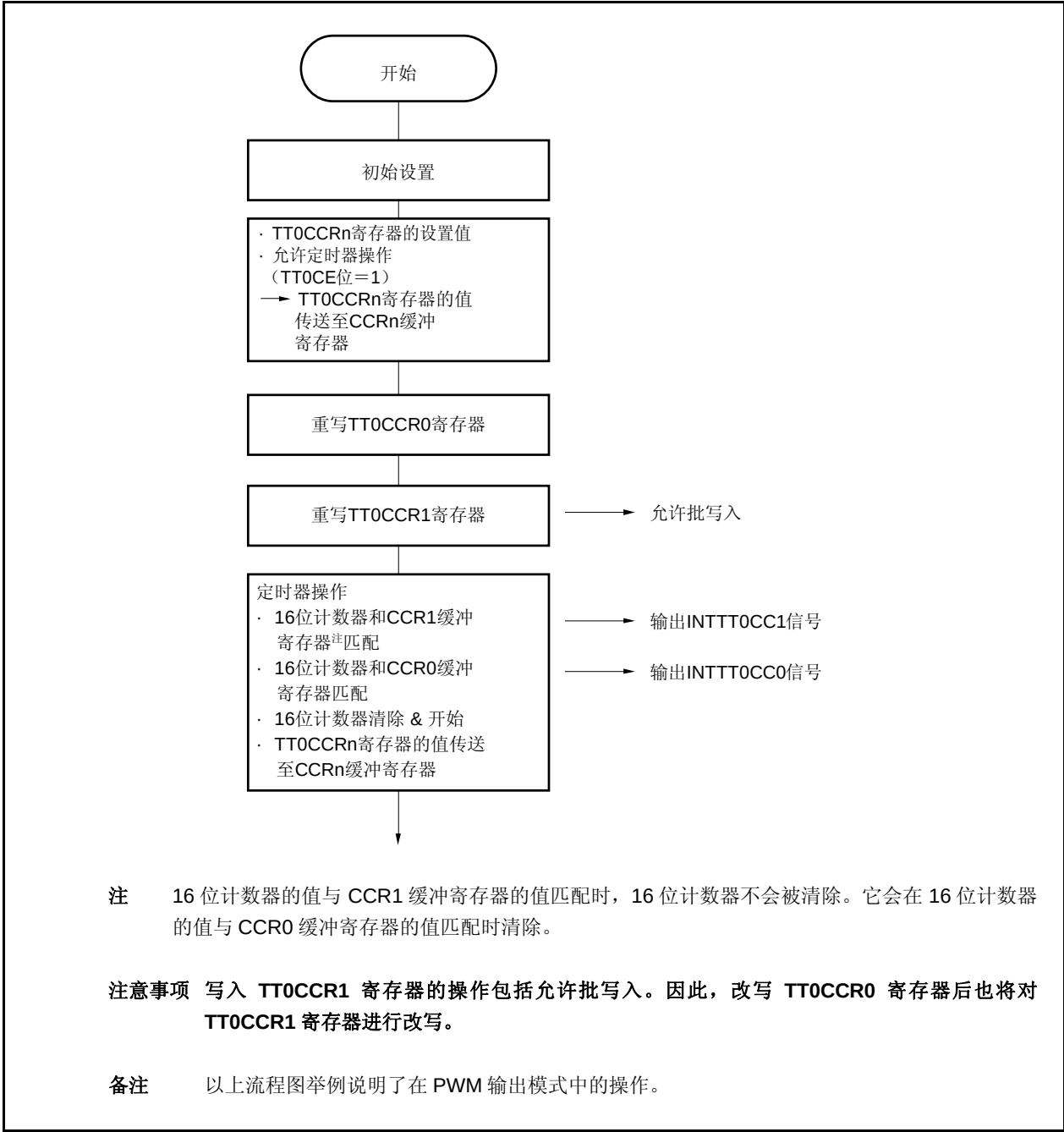
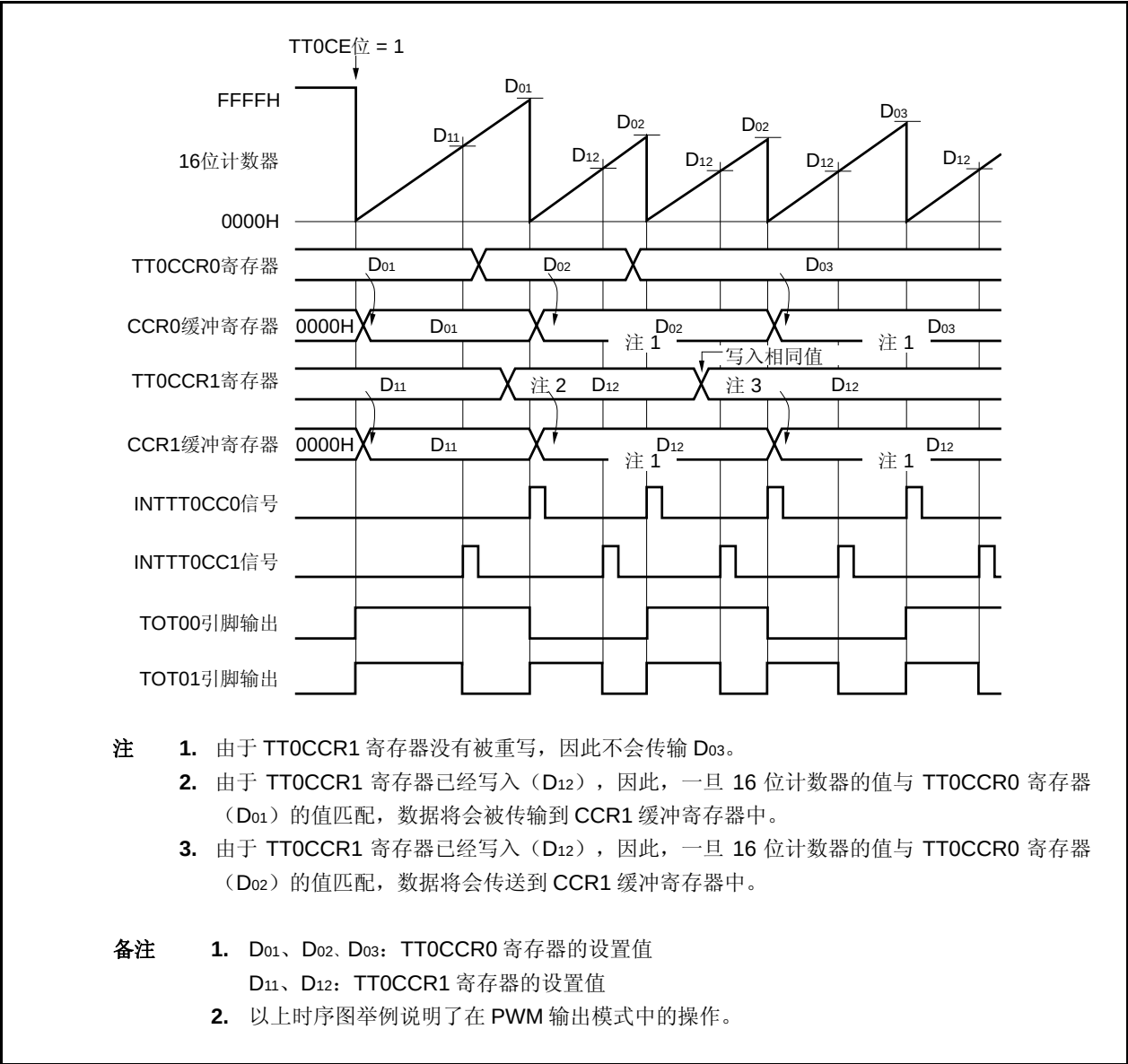


图 9-6. 批写入的时序



9.6.1 间隔定时器模式 (TT0MD3 至 TT0MD0 位=0000)

在间隔定时器模式中，如果 TT0CTL0.TT0CE 位被设为 1，那么中断请求信号 (INTTT0CC0) 就会在 TT0CCR0 寄存器所设定的间隔中生成。可以从 TOT00 引脚输出半周期等于间隔时间 50% 占空比的方波。

在间隔定时器模式下不使用 TT0CCR1 寄存器。然而，TT0CCR1 寄存器的设定值将会传送到 CCR1 缓冲寄存器中，而且当 16 位计数器的计数值和 CCR1 缓冲寄存器的值匹配时，会生成一个比较匹配中断请求信号 (INTTT0CC1)。此外，在 INTTT0CC1 信号生成时反转的 50% 占空比的方波可以从 TOT01 引脚中输出。

即使在定时器运行时，TT0CCR0 和 TT0CCR1 寄存器的值也可以进行重写。

图 9-7. 间隔定时器的配置

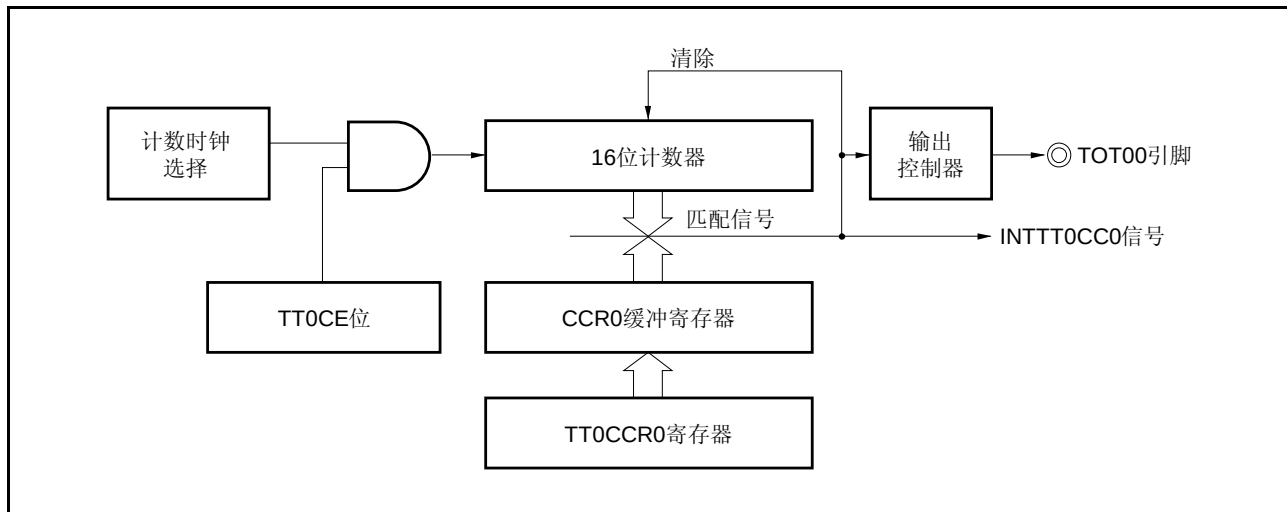
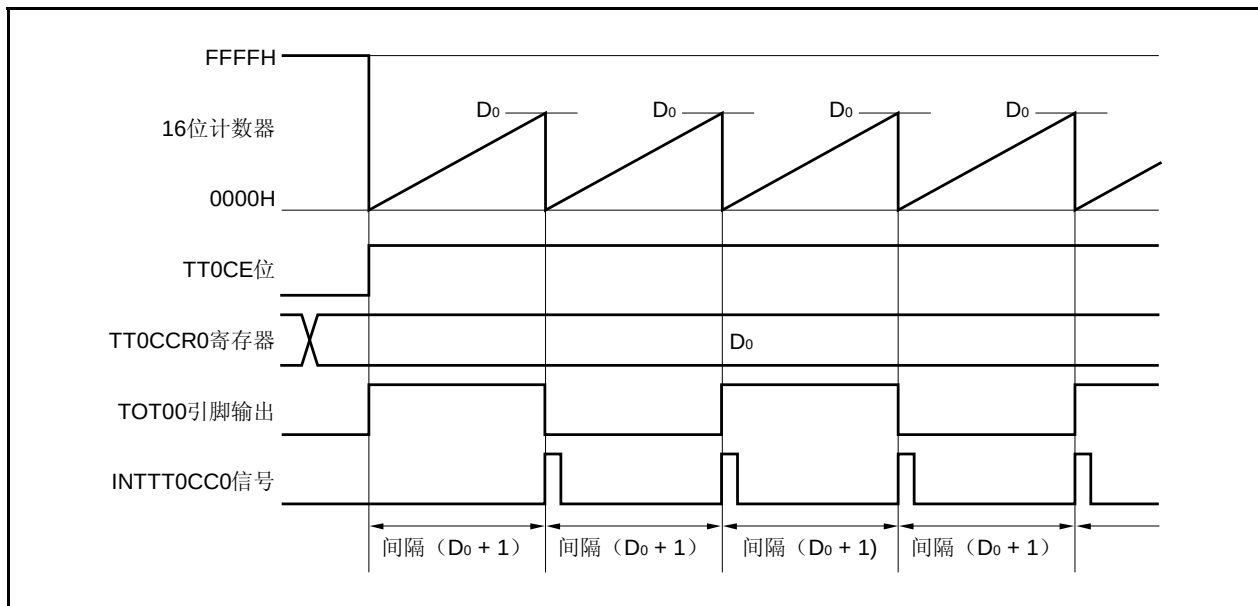


图 9-8. 间隔定时器模式下工作的基本时序



当 TT0CE 位被设置为 1，16 位计数器会与计数时钟同步将计数值从 FFFFH 清零为 0000H，并开始计数。此时，TOT00 引脚的输出反相。此外，TT0CCR0 寄存器的设定值传送给 CCR0 缓冲寄存器。

当 16 位计数器的计数值和 CCR0 缓冲寄存器的值相匹配时，16 位计数器清除为 0000H，TOT00 引脚输出反相，并生成一个比较匹配中断请求信号（INTTT0CC0）。

时间间隔可用下式计算。

$$\text{时间间隔} = (\text{TT0CCR0 寄存器的设定值} + 1) \times \text{计数时钟周期}$$

图 9-9. 间隔定时器模式操作时的寄存器设置 (1/2)

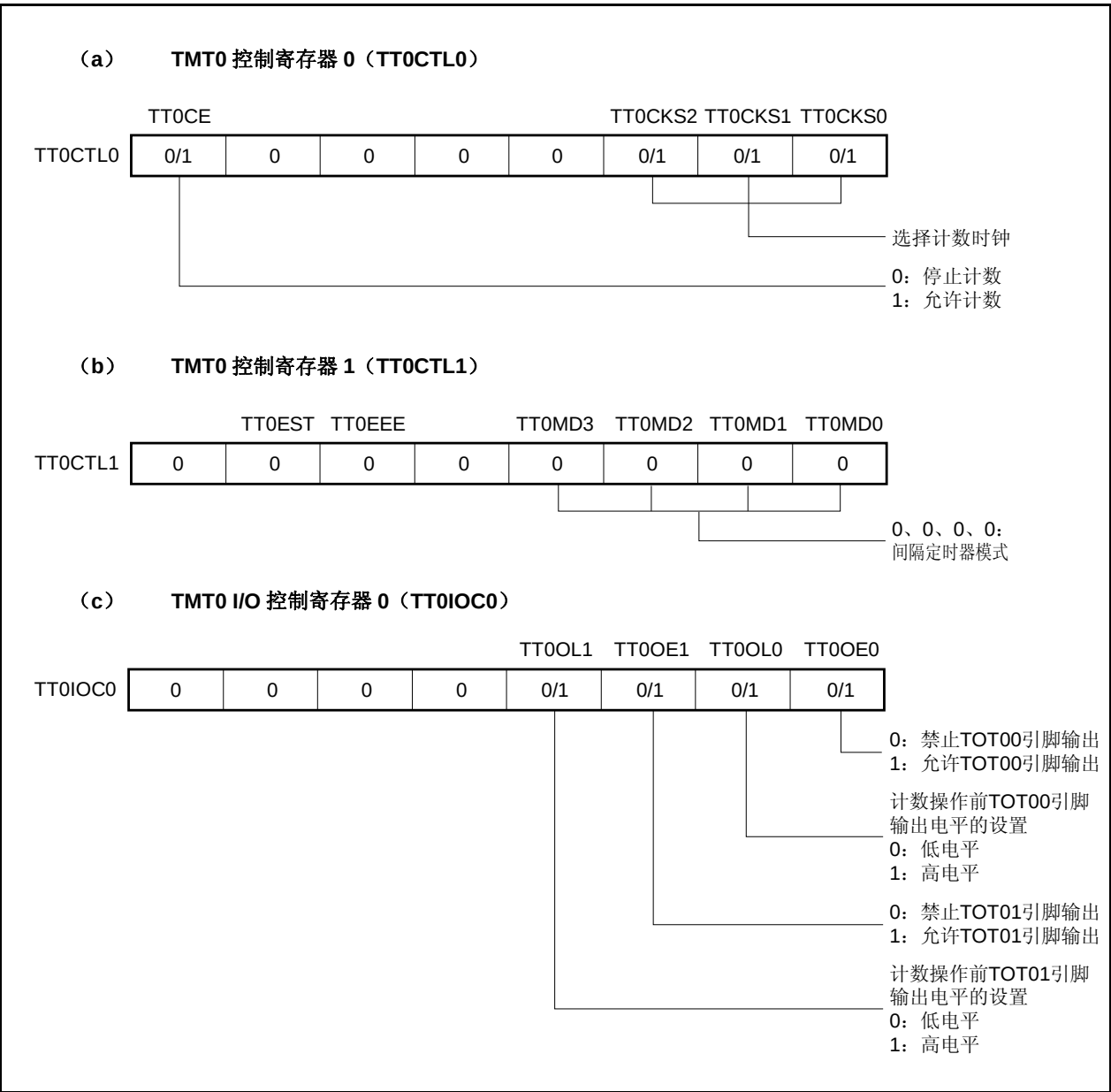


图 9-9. 间隔定时器模式操作时的寄存器设置 (2/2)

(d) TMT0 计数器读取缓冲寄存器 (TT0CNT)

通过读取 TT0CNT 寄存器，可以读取 16 计数器的计数值。

(e) TMT0 捕获/比较寄存器 0 (TT0CCR0)

如果 TT0CCR0 寄存器的值被设置为 D_0 ，则时间间隔计算如下：

间隔时间 = $(D_0 + 1) \times$ 计数时钟周期

(f) TMT0 捕获/比较寄存器 1 (TT0CCR1)

在间隔定时器模式下不使用 TT0CCR1 寄存器。但是，TT0CCR1 寄存器的设定值传送给 CCR1 缓冲寄存器。当 16 位计数器的计数值与 CCR1 缓冲寄存器的值匹配时，TOT01 引脚输出反相，且生成一个比较匹配中断请求信号 (INTTT0CC1)。

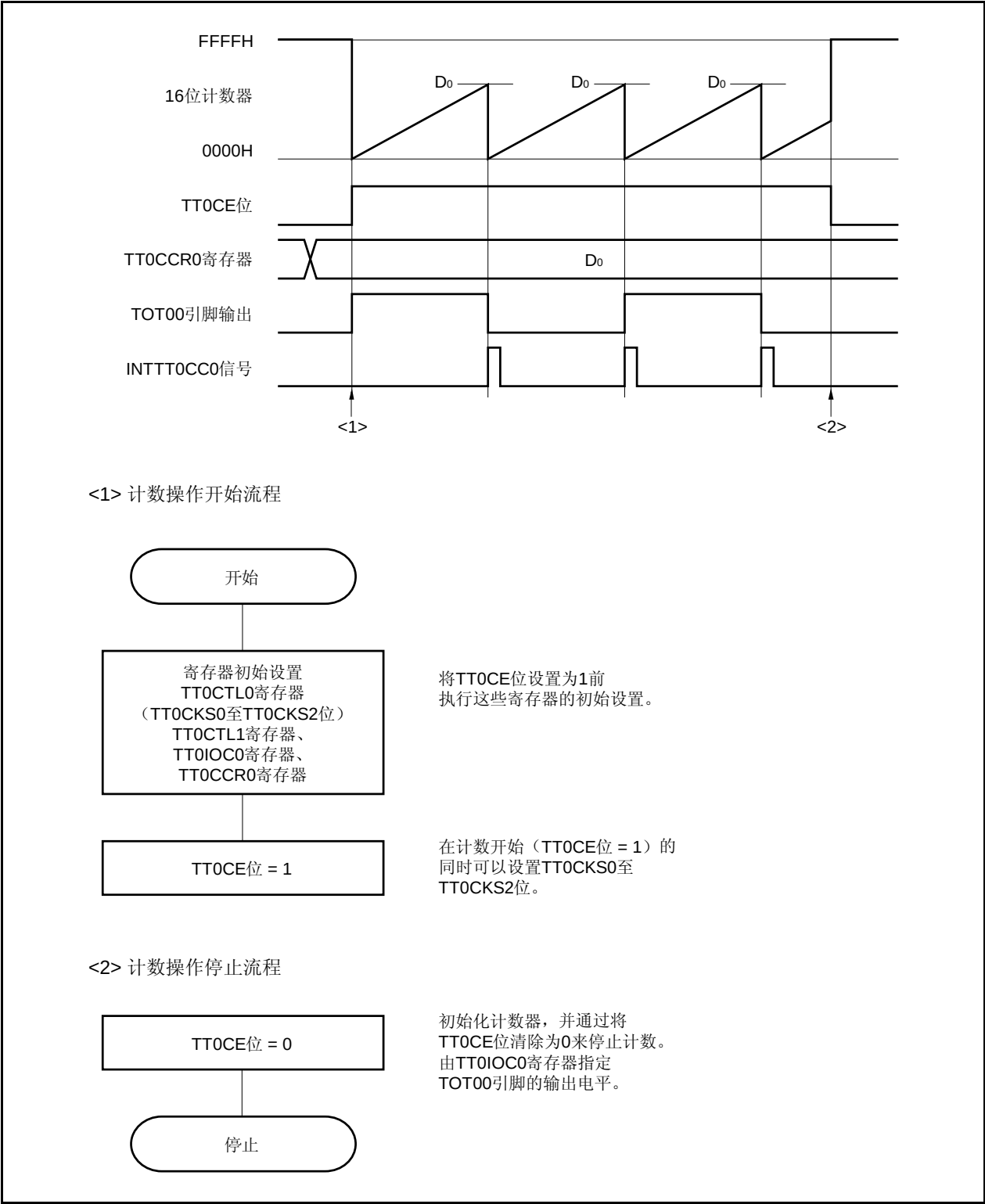
通过把与 TT0CCR0 寄存器中所设定值相同的值设置到该寄存器中，带有 50% 占空比的方波将从 TOT01 引脚输出。

当不使用 TT0CCR 寄存器时，建议将其值设为 FFFFH。此外，还会通过中断屏蔽标志 (TT0CCIC1.TT0CCMK1) 来屏蔽该寄存器。

备注 在间隔计数器模式下，不使用 TMT0 控制寄存器 2 (TT0CTL2)、TMT0 I/O 控制寄存器 1 (TT0IOC1)、TMT0 I/O 控制寄存器 2 (TT0IOC2)、TMT0 I/O 控制寄存器 3 (TT0IOC3)、TMT0 选项寄存器 0 (TT0OPT0)、TMT0 选项寄存器 1 (TT0OPT1) 和 TMT0 计数器写寄存器 (TT0TCW)。

(1) 间隔定时器模式工作流程

图 9-10. 间隔定时器模式下的软件处理流程

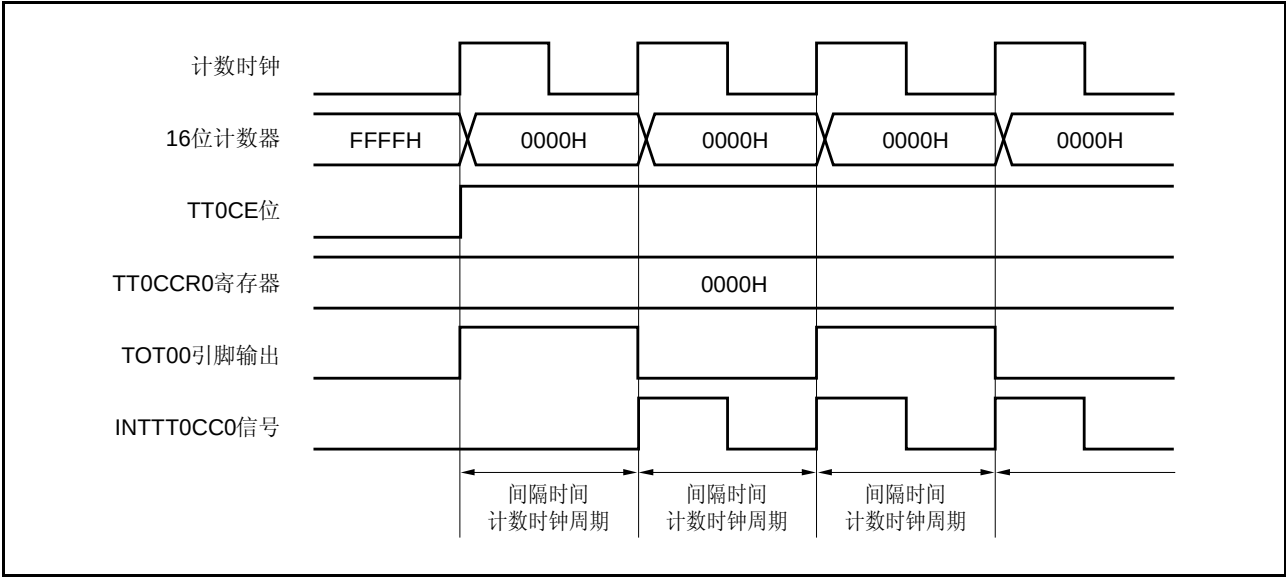


(2) 间隔定时器模式工作时序

(a) TT0CCR0 寄存器清除为 0000H 时的操作

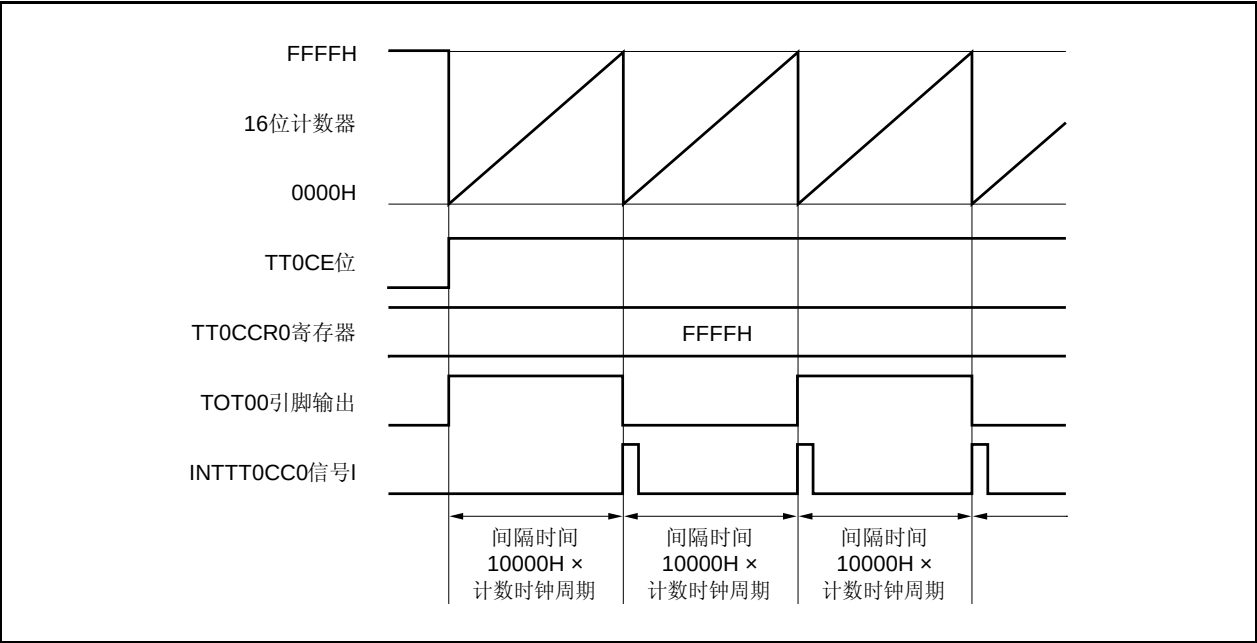
如果 TT0CCR0 寄存器设置为 0000H，每个计数时钟都会生成 INTTT0CC0 信号，并且 TOT00 引脚的输出反相。

16 位计数器的值恒为 0000H。



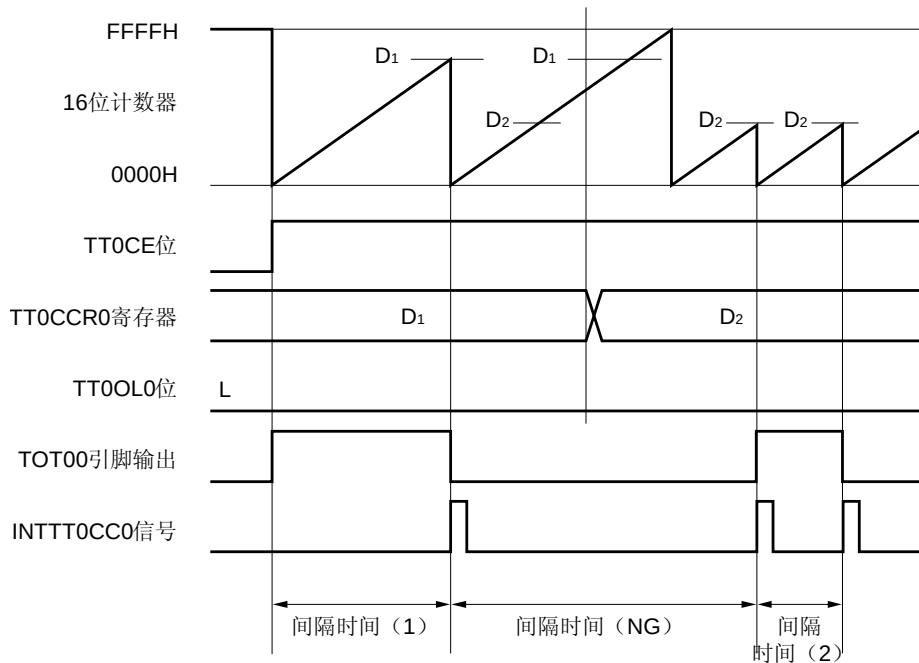
(b) TT0CCR0 寄存器置为 FFFFH 时的操作

如果 TT0CCR0 寄存器置为 FFFFH，那么 16 位计数器可以计数至 FFFFH。当下一个向上计数时序到来时，计数器同步清除为 0000H。生成 INTTT0CC0 信号，且 TOT00 引脚的输出反相。此时，并不生成上溢出中断请求信号（INTTT0OV），上溢出标志（TT0OPT0.TT0OVF 位）也不会置为 1。



(c) 重新写入 TT0CCR0 寄存器时的注意事项

如果在计数器计数期间，将 TT0CCR0 寄存器的值重写为较小点的值，则 16 位计数器可能发生上溢出。当可能发生上溢出时，停止计数，然后更改设定值。



备注

间隔时间 (1) : $(D_1 + 1) \times \text{计数时钟周期}$

间隔时间 (NG) : $(10000H + D_2 + 1) \times \text{计数时钟周期}$

间隔时间 (2) : $(D_2 + 1) \times \text{计数时钟周期}$

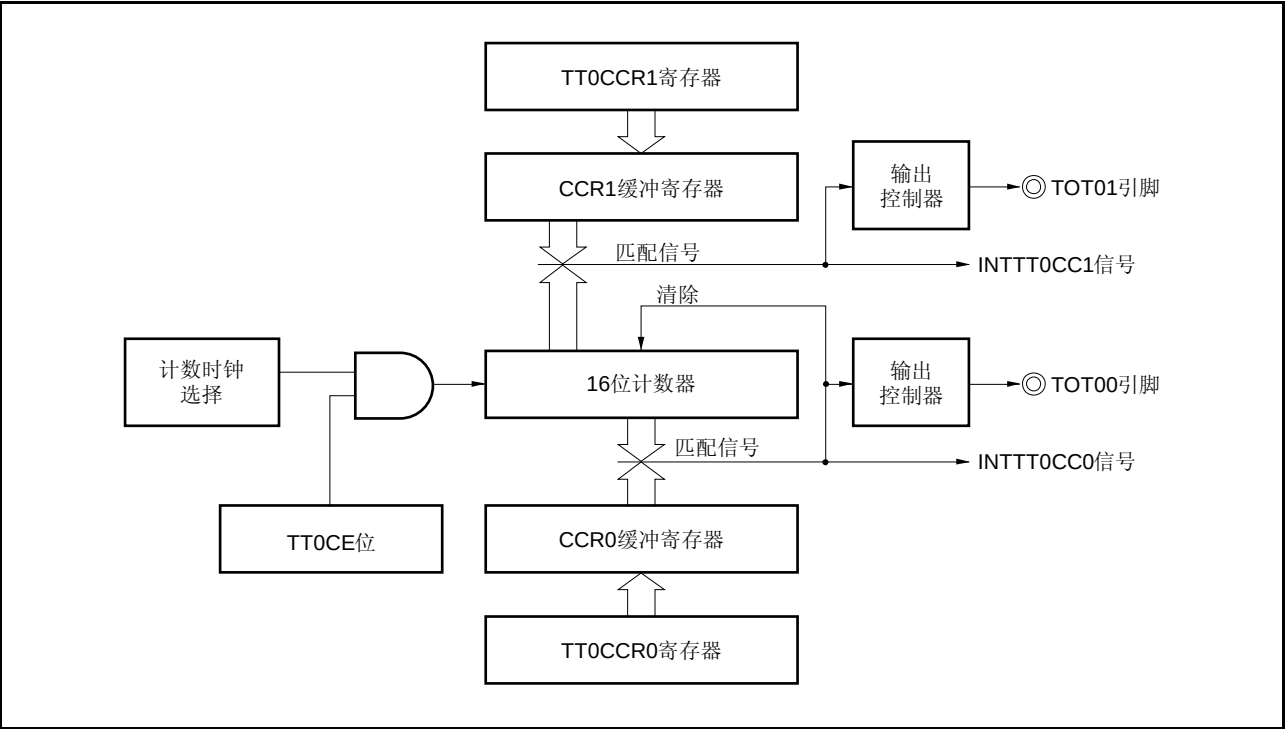
当计数值大于 D_2 而小于 D_1 时，如果将 TT0CCR0 寄存器的值从 D_1 变为 D_2 ，则随着重写 TT0CCR0 寄存器，计数值就会立即传送给 CCR0 缓冲寄存器。因此，用来比较的 16 位计数器的值是 D_2 。

但是，由于计数值已经超过了 D_2 ，所以 16 位计数器就会向上计数到 FFFFH，从而上溢出，然后从 0000H 开始重新向上计数。当计数值和 D_2 匹配时，会生成 INTTT0CC0 信号，且 TOT00 引脚的输出反相。

因此，INTTT0CC0 信号可能不会在原先预计的间隔时间“ $(D_1 + 1) \times \text{计数时钟周期}$ ”或“ $(D_2 + 1) \times \text{计数时钟周期}$ ”下生成，而会在“ $(10000H + D_2 + 1) \times \text{计数时钟周期}$ ”的间隔生成。

(d) TT0CCR1 寄存器的操作

图 9-11. TT0CCR1 寄存器的配置



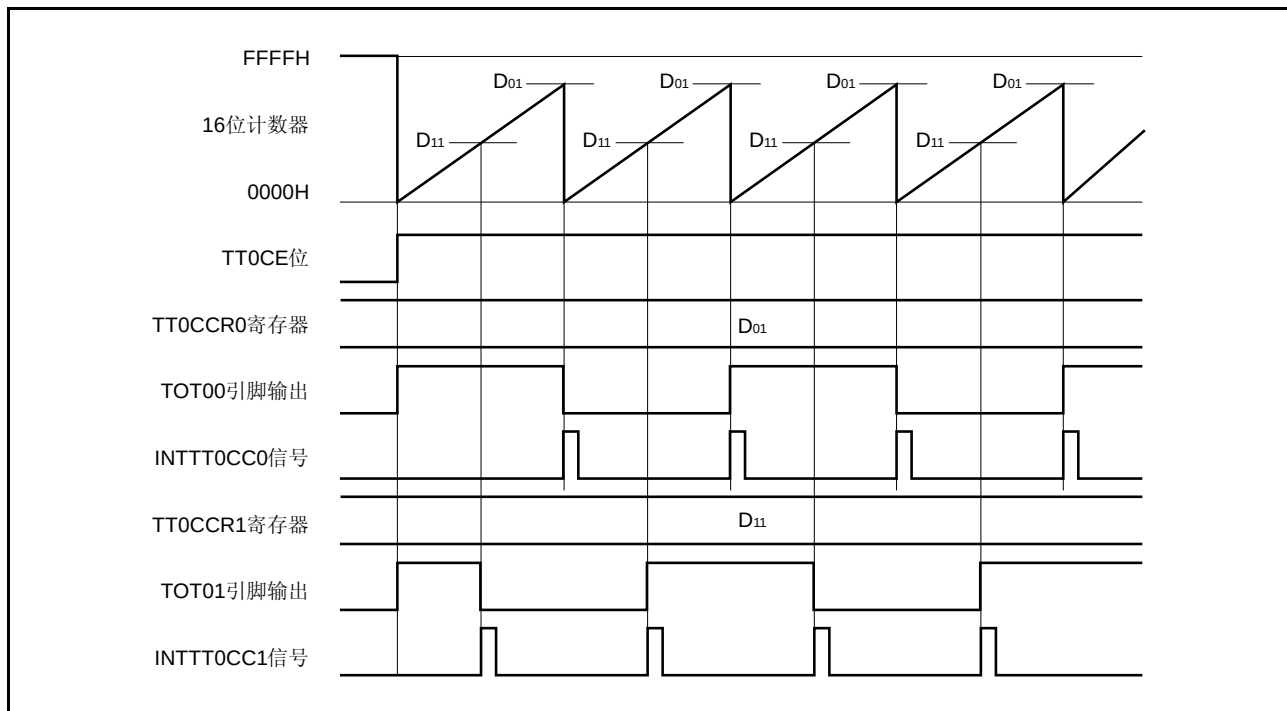
当 TT0CCR1 寄存器被设置为与 TT0CCR0 寄存器相同的值时，INTTT0CC0 信号与 INTTT0CC1 信号在相同的时序生成，且 TOT01 引脚输出反相。换句话说，可以从 TOPn1 引脚中输出带有 50%占空比的方波。

以下显示了当将不同于 TT0CCR0 寄存器中设定的值设置至 TT0CCR1 寄存器时的操作。

如果 TT0CCR1 寄存器的设定值小于 TT0CCR0 寄存器的设定值，则会在每个计数周期生成一次 INTTT0CC1 信号。同时，TOT01 引脚输出反相。

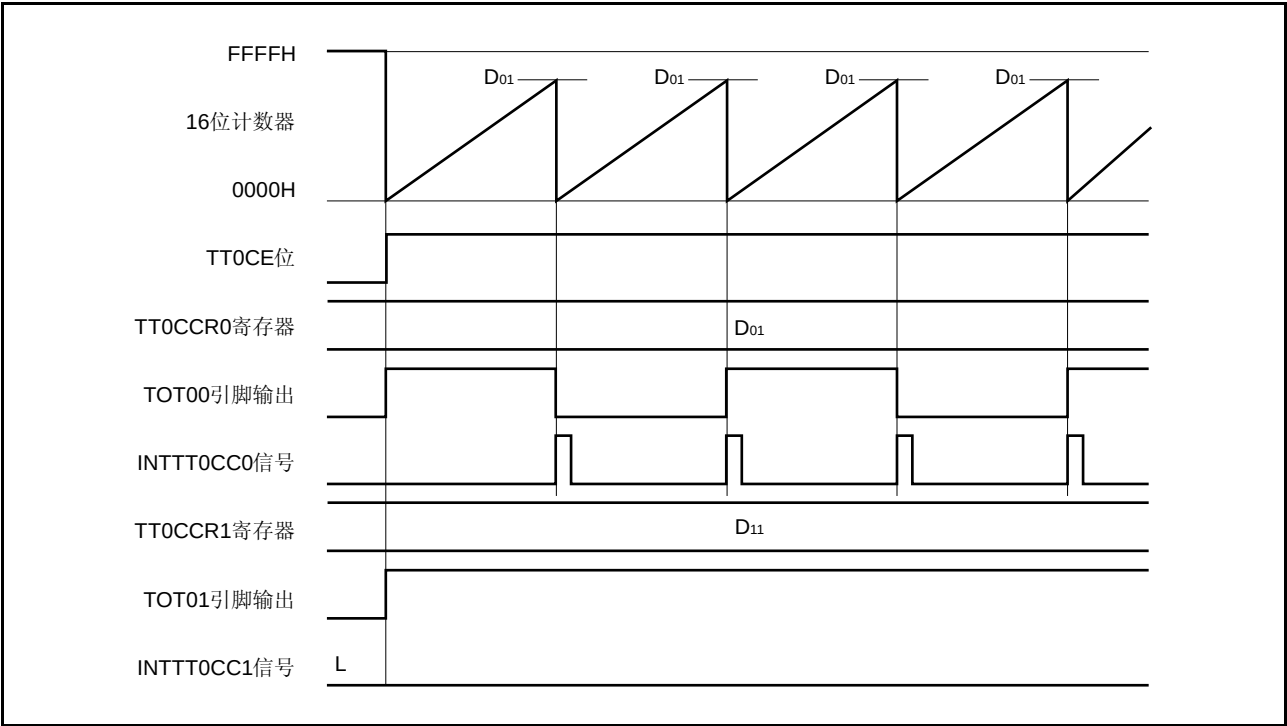
在输出一个短脉冲后，TOT01 引脚将会输出一个带有 50%占空比的方波。

图 9-12. $D_{01} \geq D_{11}$ 时的时序图



如果 TT0CCR1 寄存器的设定值大于 TT0CCR0 寄存器的设定值，则 16 位计数器的计数值和 TT0CCR1 寄存器的值不匹配。因而，不会生成 INTTT0CC1 信号，TOT01 引脚的输出也不会改变。
当不使用 TT0CCR1 寄存器时，建议将其值设为 FFFFH。

图 9-13. D₀₁ < D₁₁ 时的时序图



9.6.2 外部事件计数器模式 (TT0MD3 至 TT0MD0 位=0001)

在外部事件计数模式下，当 TT0CTL0.TT0CE 位置为 1 时，外部事件计数输入 (EVTT0) 的有效边沿被计数，而且每次由 TT0CCR0 寄存器设置的有效边沿数目计数完成时，就会生成一个中断请求信号 (INTTT0CC0)。不使用 TOT00 和 TOT01 引脚。

在外部事件计数模式中不使用 TT0CCR1 寄存器。

图 9-14. 外部事件计数器模式的配置

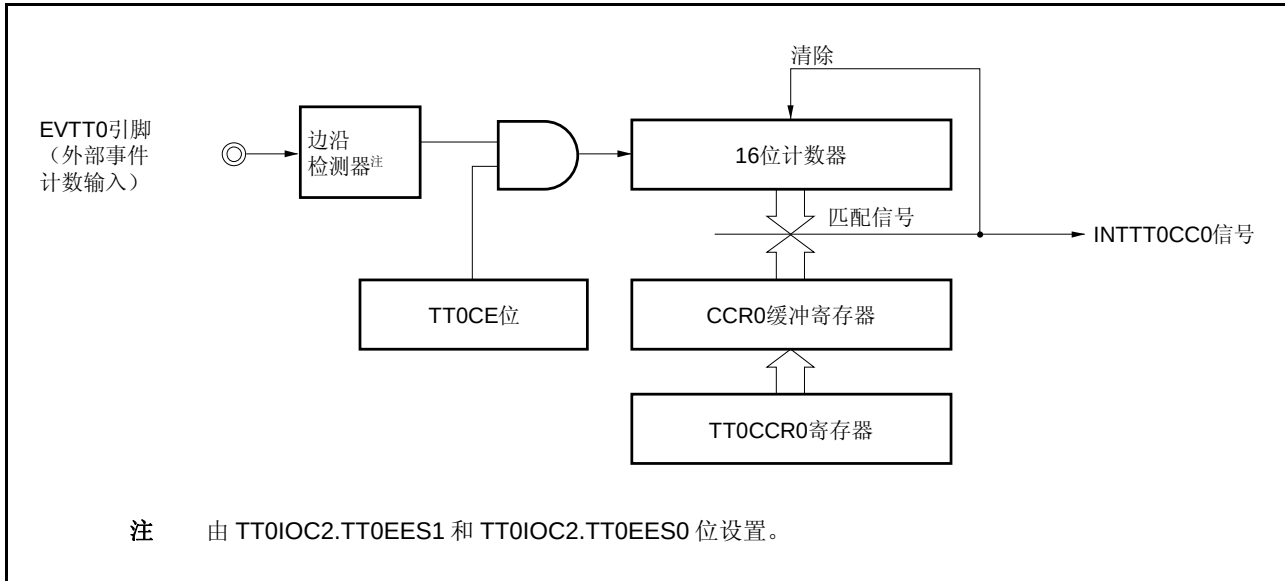
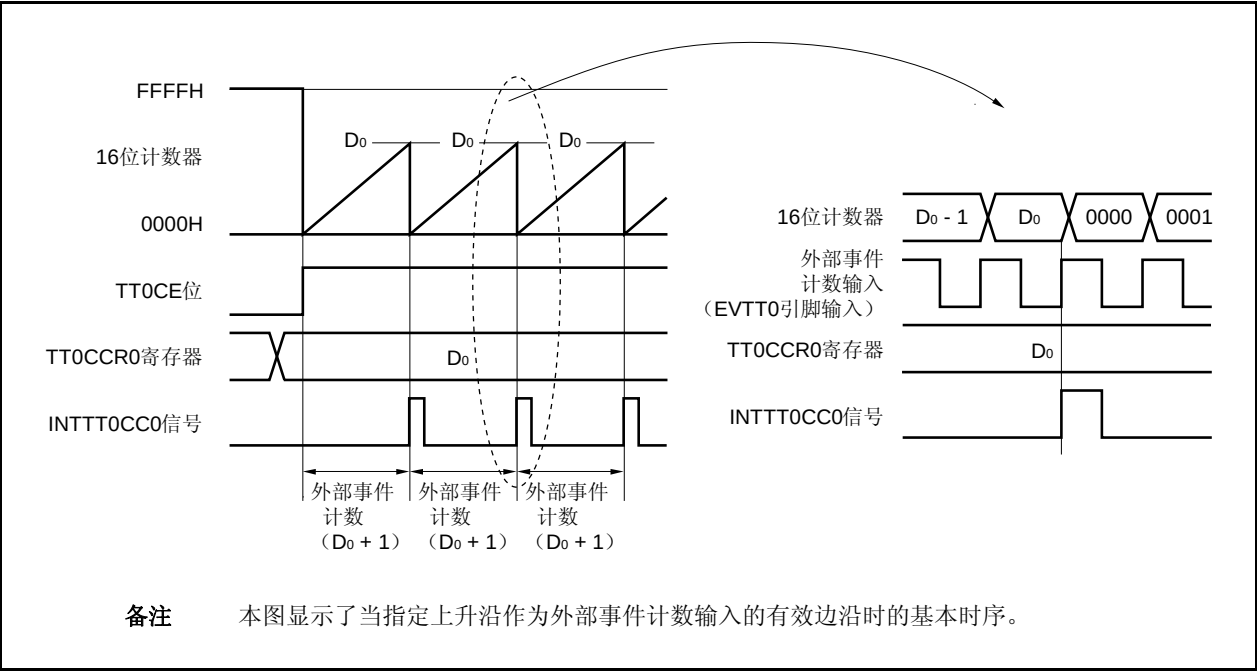


图 9-15. 外部事件计数模式的基本时序



当 TT0CE 位置为 1 时，16 位计数器的值由 FFFFH 清除为 0000H。每次检测到外部事件计数输入的有效沿，计数器就会计数。此外，TT0CCR0 寄存器的设定值传送给 CCR0 缓冲寄存器。

当 16 位计数器的计数值和 CCR0 缓冲寄存器的数值匹配时，16 位计数器清除为 0000H，同时生成一个比较匹配中断请求信号 (INTTT0CC0)。

每当外部事件计数输入的有效边沿检测 (计数) 达到“TT0CCR0 寄存器设定值+1”时，就会生成 INTTT0CC0 信号。

图 9-16. 外部事件计数模式下的寄存器操作设置 (1/2)

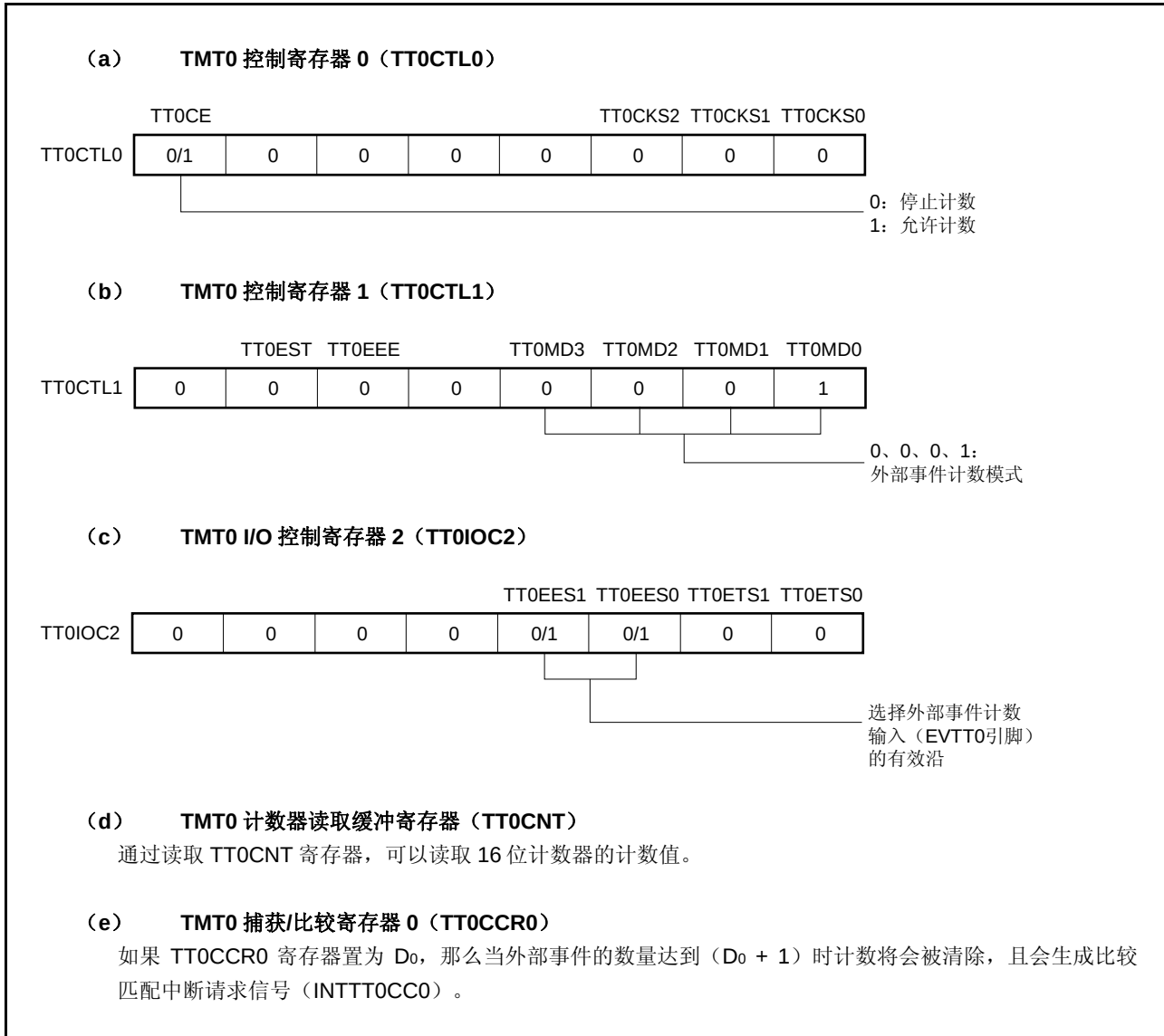


图 9-16. 外部事件计数模式下的寄存器操作设置 (2/2)

(f) TMT0 捕获/比较寄存器 1 (TT0CCR1)

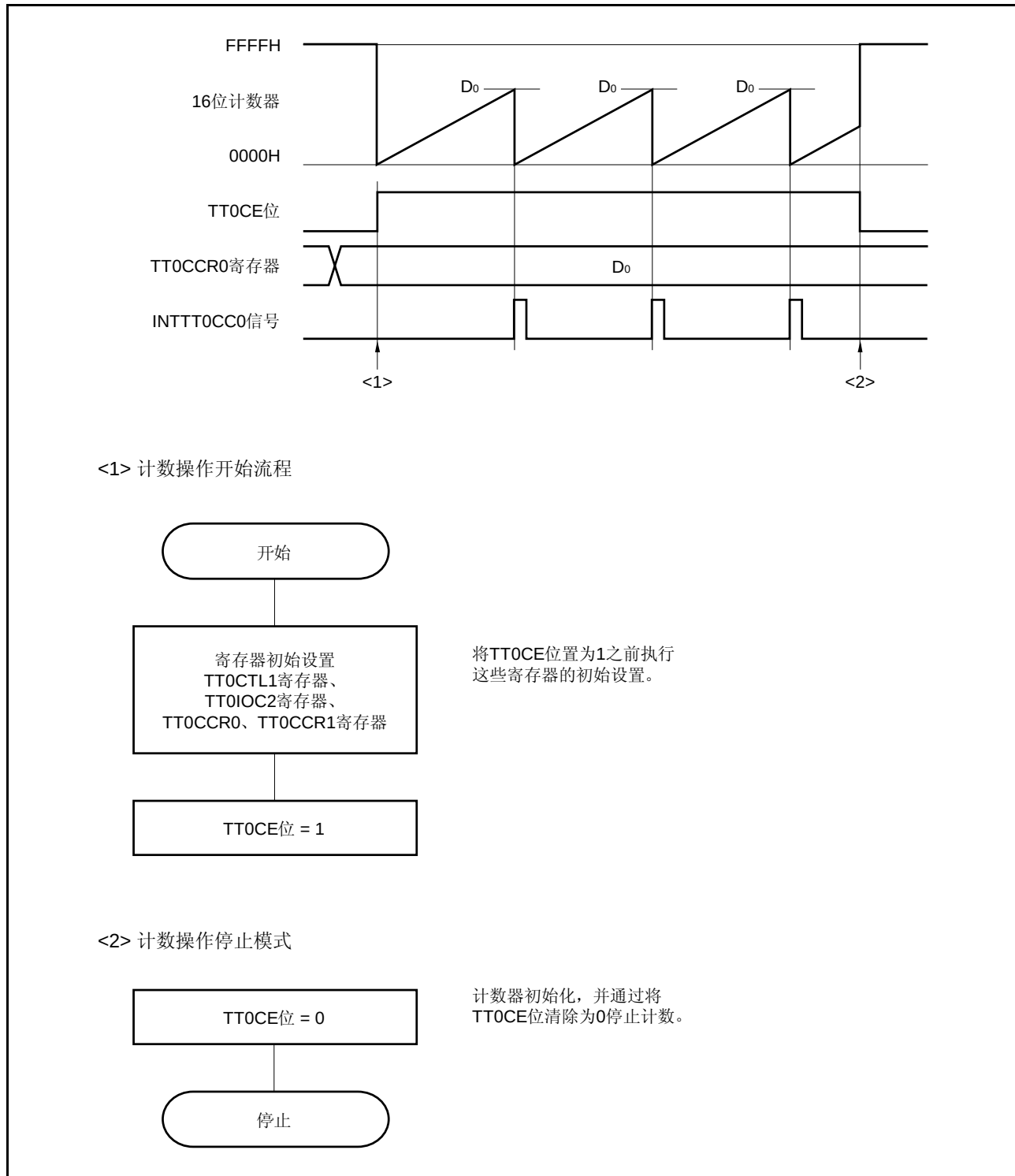
在外部事件计数模式中不使用 TT0CCR1 寄存器。但是，将 TT0CCR1 寄存器的设定值传送给 CCR1 缓冲寄存器。当 16 位计数器的计数值和 CCR1 缓冲寄存器的值匹配时，也会生成一个比较匹配中断请求信号 (INTTT0CC1)。

当不使用 TT0CCR1 寄存器时，建议将其值设为 FFFFH。此外，还会通过中断屏蔽标志 (TT0CCIC1.TT0CCMK1) 来屏蔽该寄存器。

备注 在外部事件计数模式下，不使用 TMT0 控制寄存器 2 (TT0CTL2)、TMT0 I/O 控制寄存器 0 (TT0IOC0)、TMT0 I/O 控制寄存器 1 (TT0IOC1)、TMT0 I/O 控制寄存器 3 (TT0IOC3)、TMT0 选项寄存器 0 (TT0OPT0)、TMT0 选项寄存器 1 (TT0OPT1) 和 TMT0 计数器写寄存器 (TT0TCW)。

(1) 外部事件计数模式的操作流程

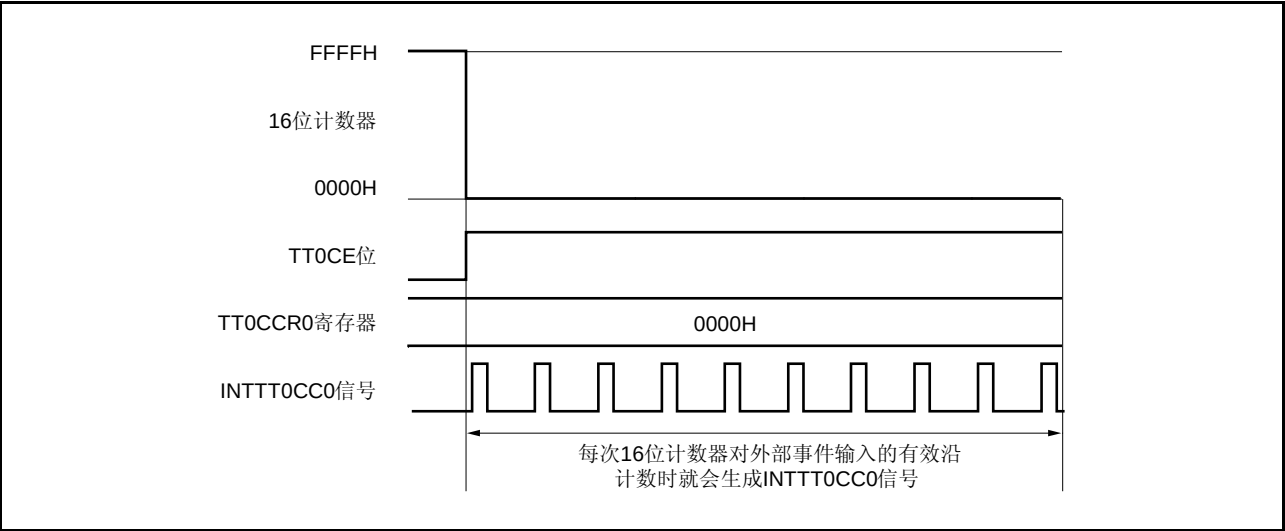
图 9-17. 外部事件计数模式下的软件处理流程



(2) 外部事件计数模式下工作时序

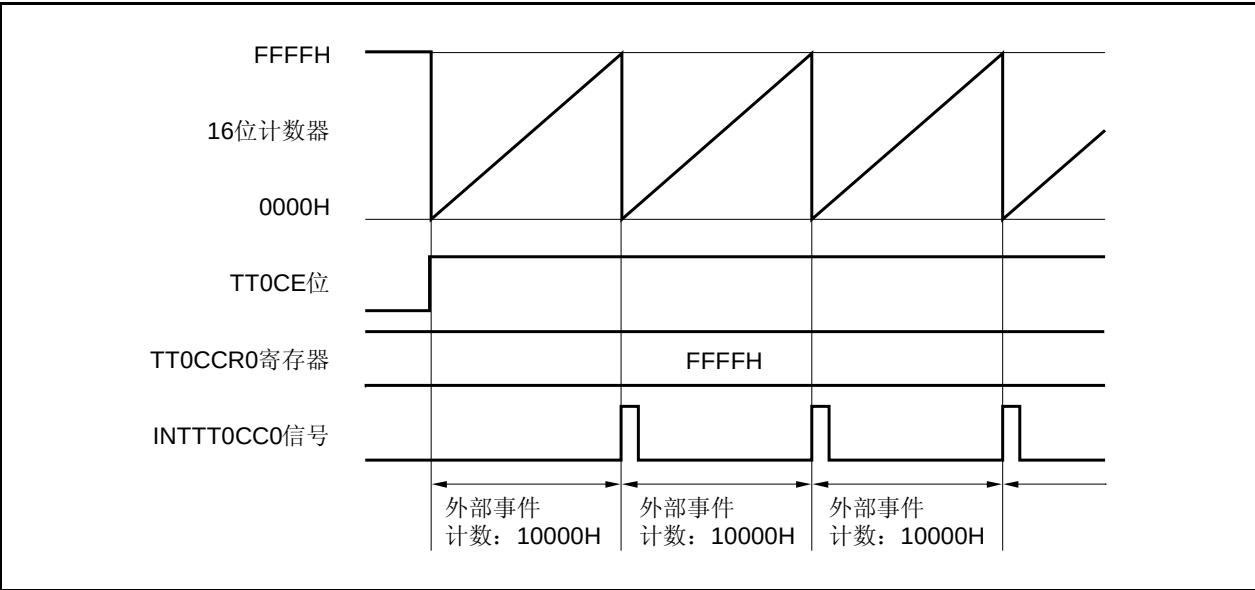
(a) TT0CCR0 寄存器置为 0000H 时的操作

，当 TT0CCR0 寄存器置为 0000H 时，每次计数器检测到外部事件计数信号的有效沿和计数器的值与 CCR0 缓冲寄存器匹配时 16 位计数器就会重复地清除为 0000H，并生成 INTTT0CC0 信号。
16 位计数器的值总是为 0000H。



(b) TT0CCR0 寄存器置为 FFFFH 时的操作

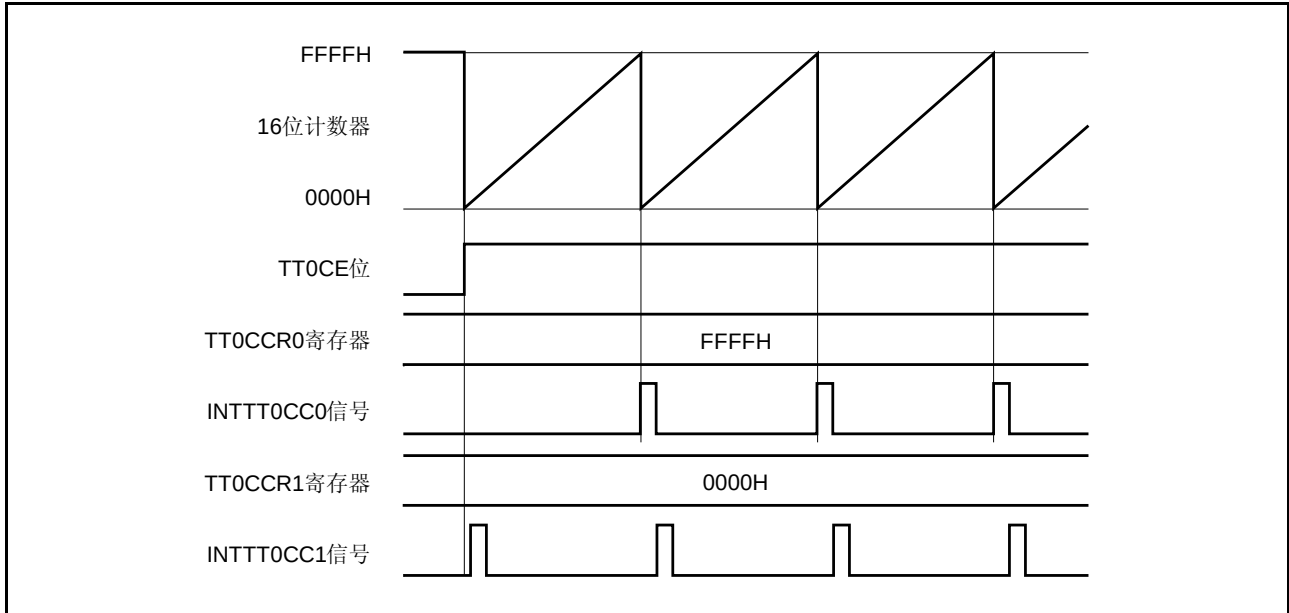
如果 TT0CCR0 寄存器置为 FFFFH，每次检测到外部事件计数信号的有效边沿时 16 位计数器就会计数。当下一个向上计数到来时 16 位计数器同步清为 0000H，并且生成 INTTT0CC0 信号。此时，TT0OPT0.TT0OVF 不置位。



(c) TT0CCR0 置为 FFFFH 和 TT0CCR1 置为 0000H 的操作

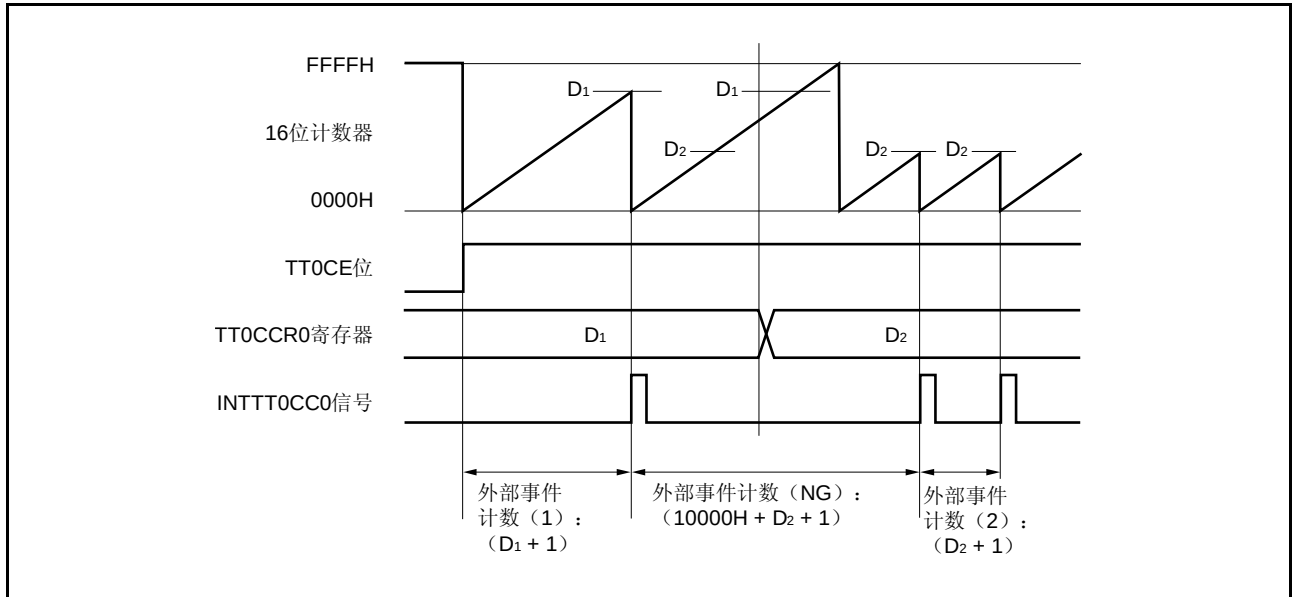
当 TT0CCR0 寄存器置为 FFFFH 时，每次检测到外部事件计数信号的有效边沿时 16 位计数器就会计数。然后，当下一个向上计数到来时 16 位计数器同步清为 0000H，并生成 INTTT0CC0 信号。此时，TT0OPT0.TT0OVF 不置位。

如果将 TT0CCR1 寄存器置为 0000H，则当 16 位计数器清除为 0000H 时就会生成 INTTT0CC1 信号。



(d) 重新写入 TT0CCR0 寄存器时的注意事项

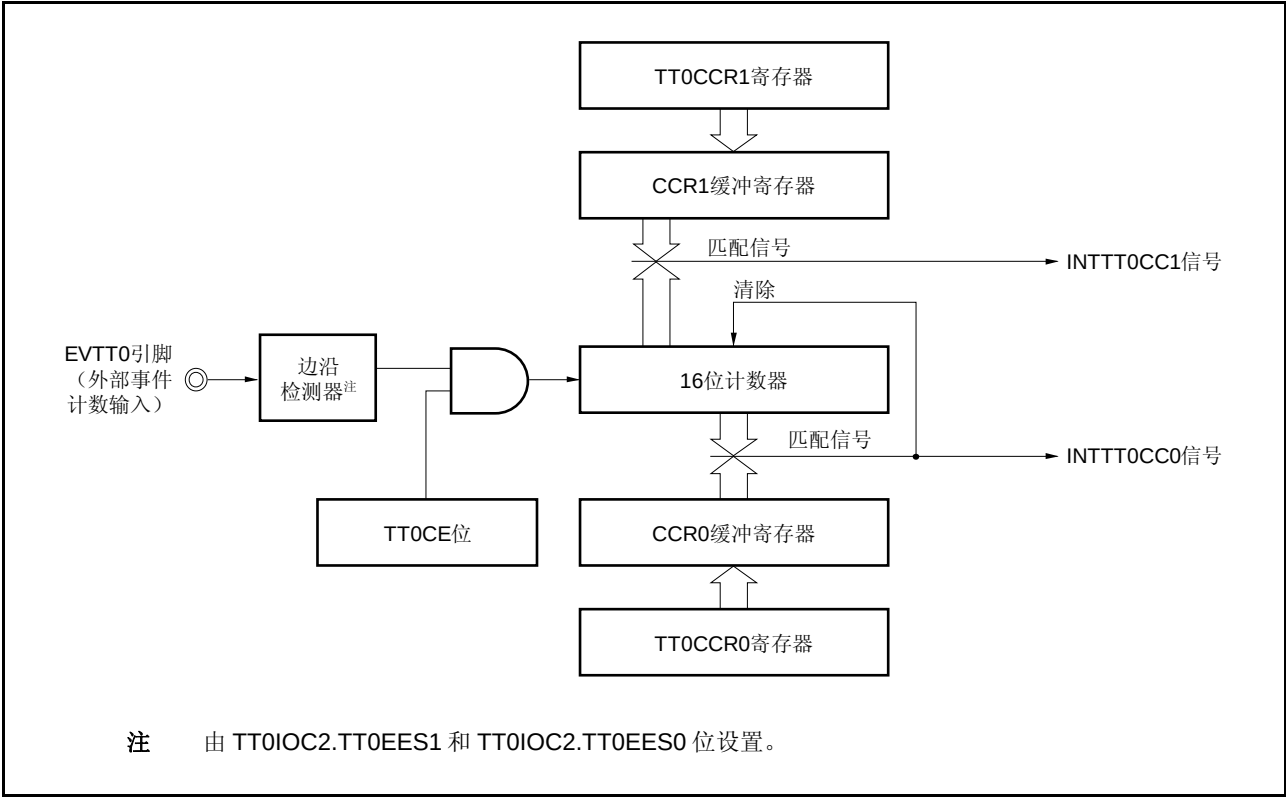
如果在计数器计数期间，将寄存器 TT0CCR0 的值重写为一个较小点的值，则 16 位计数器可能发生上溢出。当可能发生上溢出时，停止一次计数并更改设定值。



当 16 位计数器计数值大于 D_2 而小于 D_1 时，如果将 TT0CCR0 寄存器的值从 D_1 变为 D_2 ，则在重写 TT0CCR0 寄存器后，计数值会立即传送给 CCR0 缓冲寄存器。因而，与 16 位计数器相比较的值会是 D_2 。但是，由于计数值已经超过了 D ，所以 16 位计数器就会一直向上累积计数到 FFFFH 直到上溢出，然后从 0000H 开始重新计数。当计数值与 D_2 匹配时，会生成 INTTT0CC0 信号。因此，INTTT0CC0 中断请求信号可能不会在原先预计的“ $(D_1 + 1)$ 次”或“ $(D_2 + 1)$ 次”有效边缘上生成，而可能在“ $(10000H + D_2 + 1)$ 次”的有效边缘上生成。

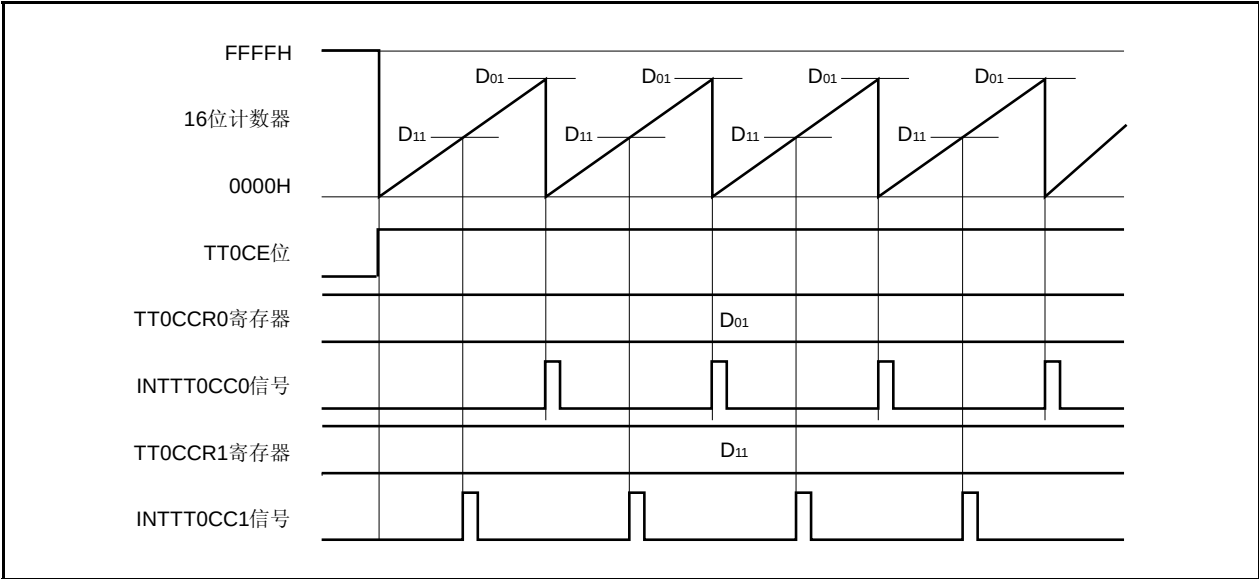
(e) TT0CCR1 寄存器的操作

图 9-18. TT0CCR1 寄存器的配置



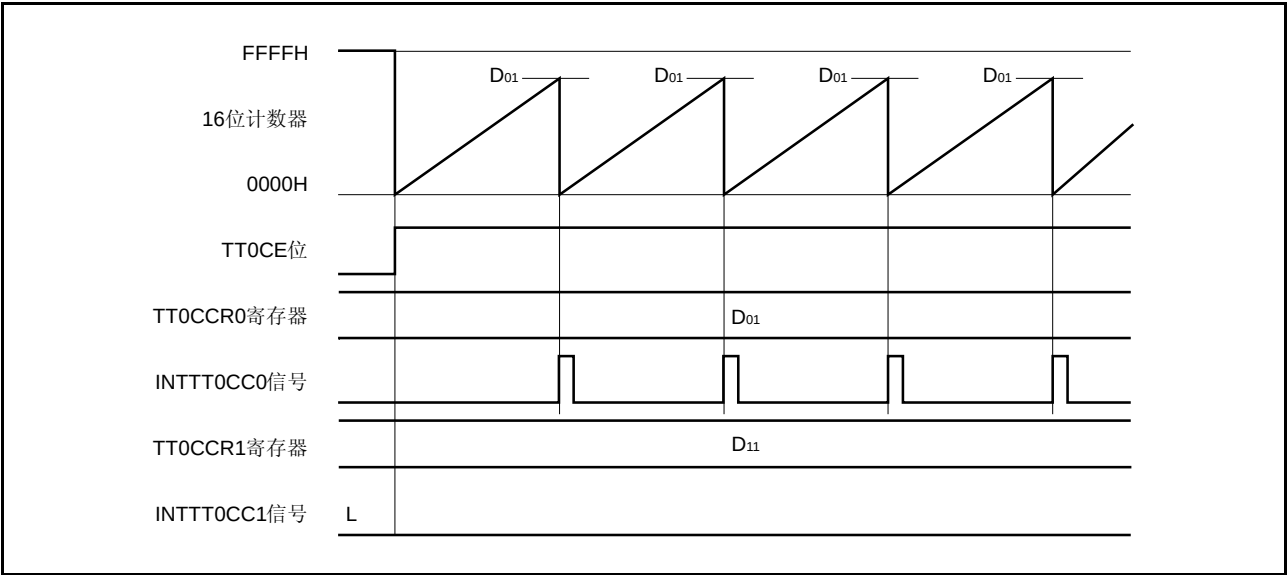
如果 TT0CCR1 寄存器的设定值小于 TT0CCR0 寄存器的设定值，则 INTTT0CC1 中断请求信号每个计数周期产生一次。

图 9-19. $D_{01} \geq D_{11}$ 时的时序图



如果 TT0CCR1 寄存器的设定值大于 TT0CCR0 寄存器的设定值，则不会生成 INTTT0CC1 信号，因为 16 位计数器的计数值和 TT0CCR1 寄存器的值不匹配。
当不使用 TT0CCR1 寄存器时，建议将其值设为 FFFFH。

图 9-20. D₀₁ < D₁₁ 时的时序图



9.6.3 外部触发脉冲输出模式 (TT0MD3 至 TT0MD0 位=0010)

在外部触发脉冲输出模式下，当 TT0CTL0.TT0CE 位置为 1 时，16 位定时器/事件计数器 T 等待触发。当检测到外部触发输入 (EVTTO) 的有效边沿后，16 位定时器/事件计数器 T 开始计数，同时从 TOT01 引脚输出一个 PWM 波形。

不使用外部触发，而是使用软件触发代替，同样可以输出脉冲。当使用软件触发时，也可以从 TOT00 引脚中输出带有 50% 占空比的方波，它的半周期等于 TT0CCR0 寄存器的设定值+1。

图 9-21. 外部触发脉冲输出模式的配置

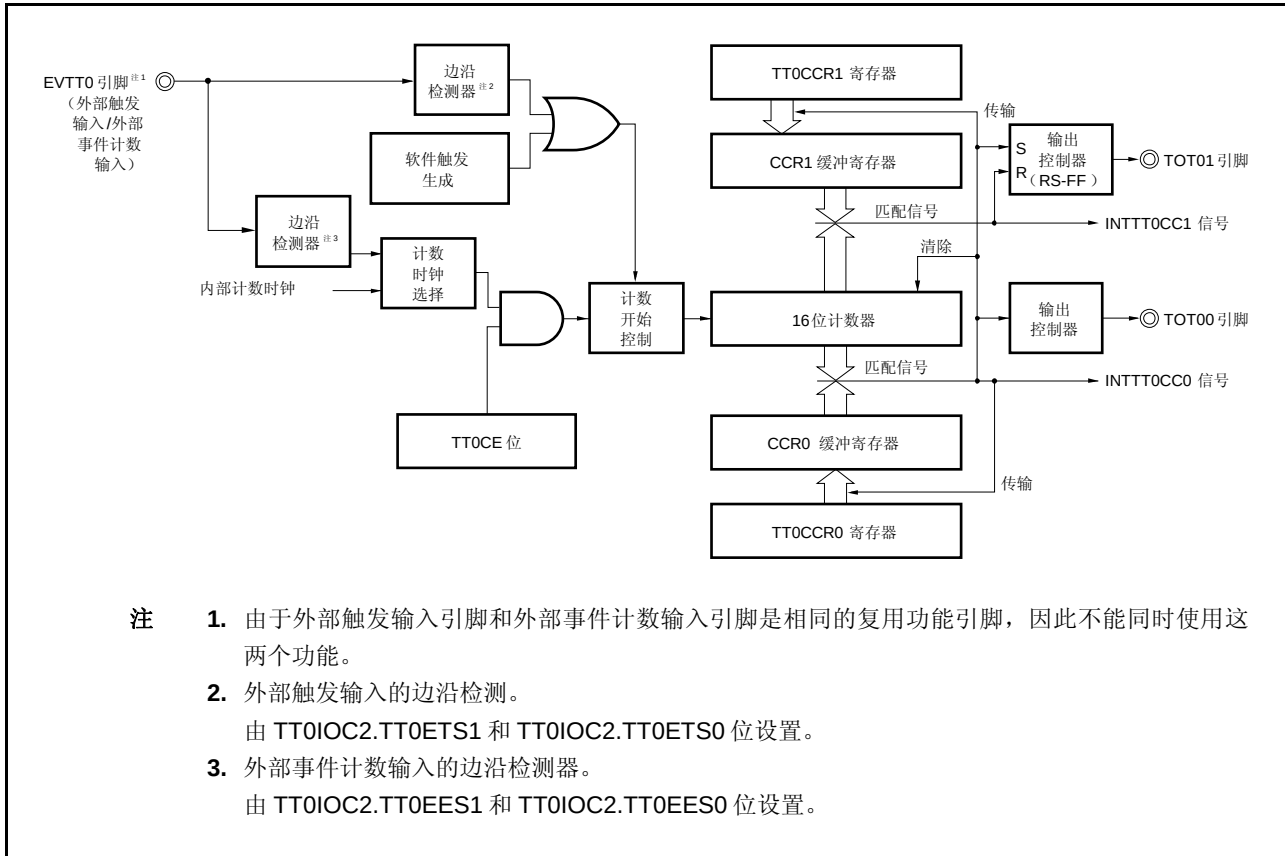
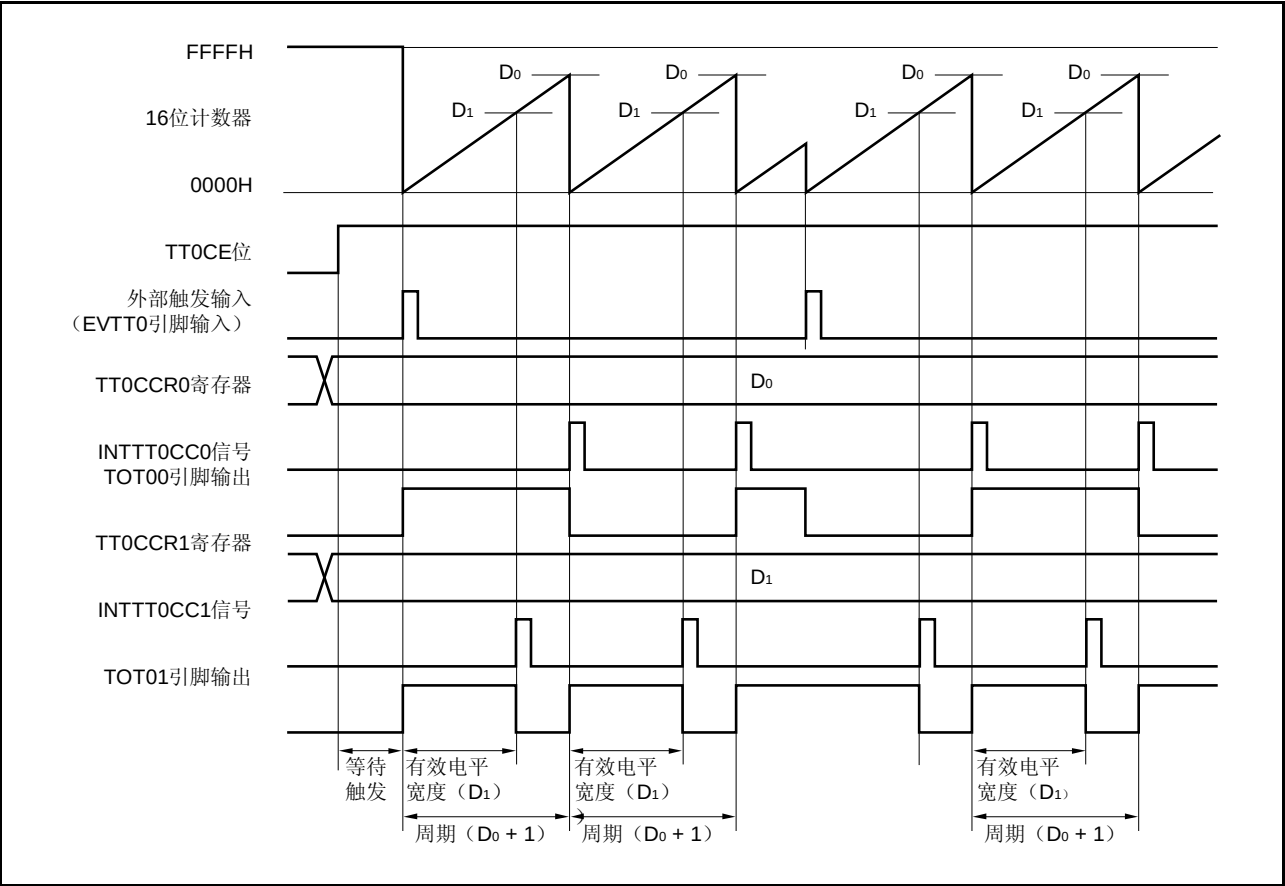


图 9-22. 外部触发脉冲输出模式的基本时序



当 TT0CE 位置为 1 时，16 位定时器/事件计数器 T 等待触发。当触发产生时，16 位计数器从 FFFFH 清除为 0000H，同时开始计数，并从 TOT01 引脚输出一个 PWM 波形。若在计数运行中再次产生触发，则计数器清除为 0000H 并重新开始计数。（TOT00 引脚的输出反相。当在出现触发时，不管当前状态如何（高/低），TOT01 引脚都会输出一个高电平。）

PWM 波形的有效电平宽度、周期和占空比计算如下。

有效电平宽度 = (TT0CCR1 寄存器的设定值) × 计数时钟周期
周期 = (TT0CCR0 寄存器的设定值 + 1) × 计数时钟周期
占空比系数 = (TT0CCR1 寄存器的设置值) / (TT0CCR0 寄存器的设置值 + 1)

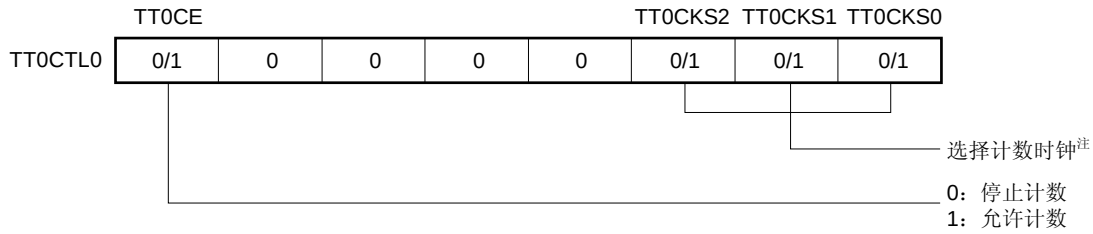
当 16 位计数器在其计数值与 CCR0 缓冲寄存器的值匹配之后进行下一次计数时，会生成比较匹配中断请求信号 (INTT0CC0)，且 16 位计数器清除为 0000H。当 16 位计数器的计数值与 CCR1 缓冲寄存器的值匹配时，会生成比较匹配中断请求信号 (INTT0CC1)。

当 16 位计数器的计数值与 CCRn 缓冲寄存器的值匹配，且 16 位计数器清零为 0000H 时，TT0CCRn 寄存器设置的值就传送到 CCRn 缓冲寄存器。

外部触发输入 (EVT0) 的有效边沿，或将软件触发 (TT0CTL1.TT0EST 位) 设为 1，都可当作触发信号 (n=0、1)。

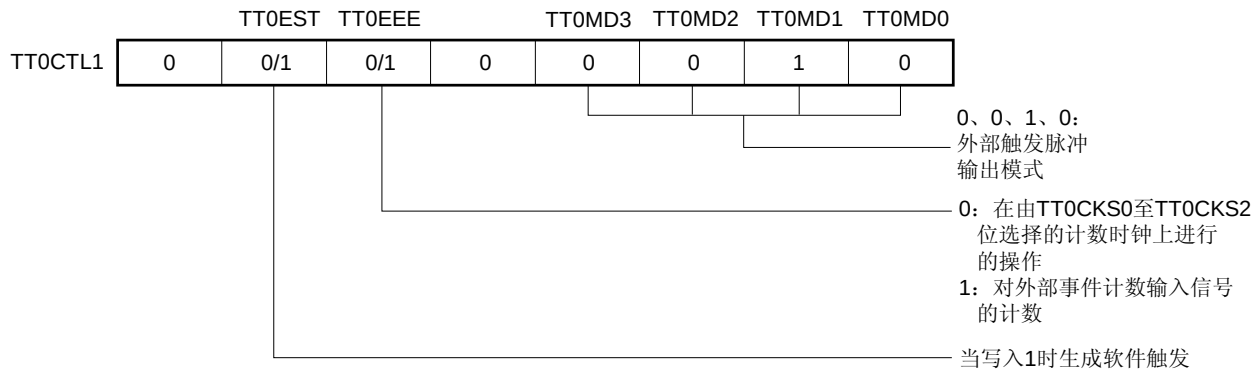
图 9-23. 外部触发脉冲输出模式的寄存器设置 (1/2)

(a) TMT0 控制寄存器 0 (TT0CTL0)

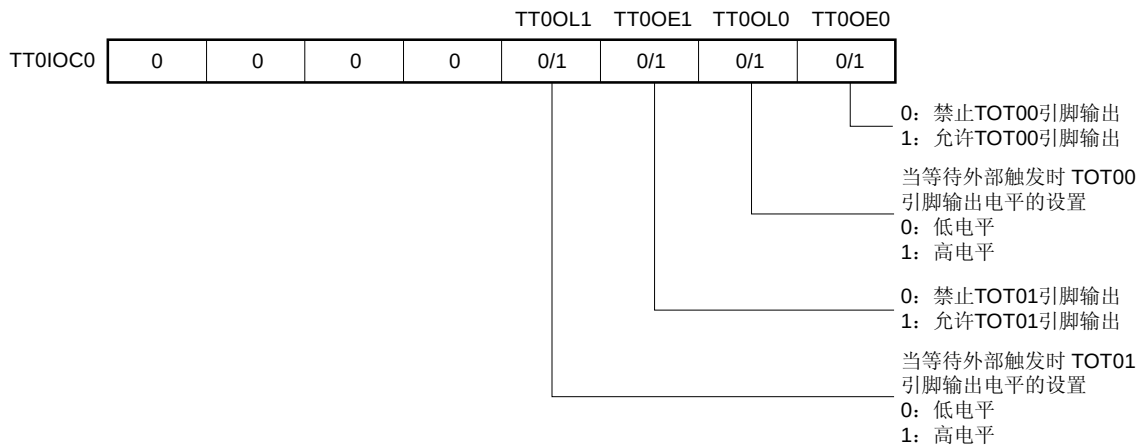


注 当 TT0CTL1.TT0EEE 位=1 时, 设置无效。

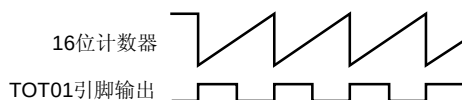
(b) TMT0 控制寄存器 1 (TT0CTL1)



(c) TMT0 I/O 控制寄存器 0 (TT0IOC0)



· 当 TT0OL1 位 = 0



· 当 TT0OL1 位 = 1

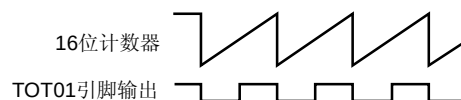
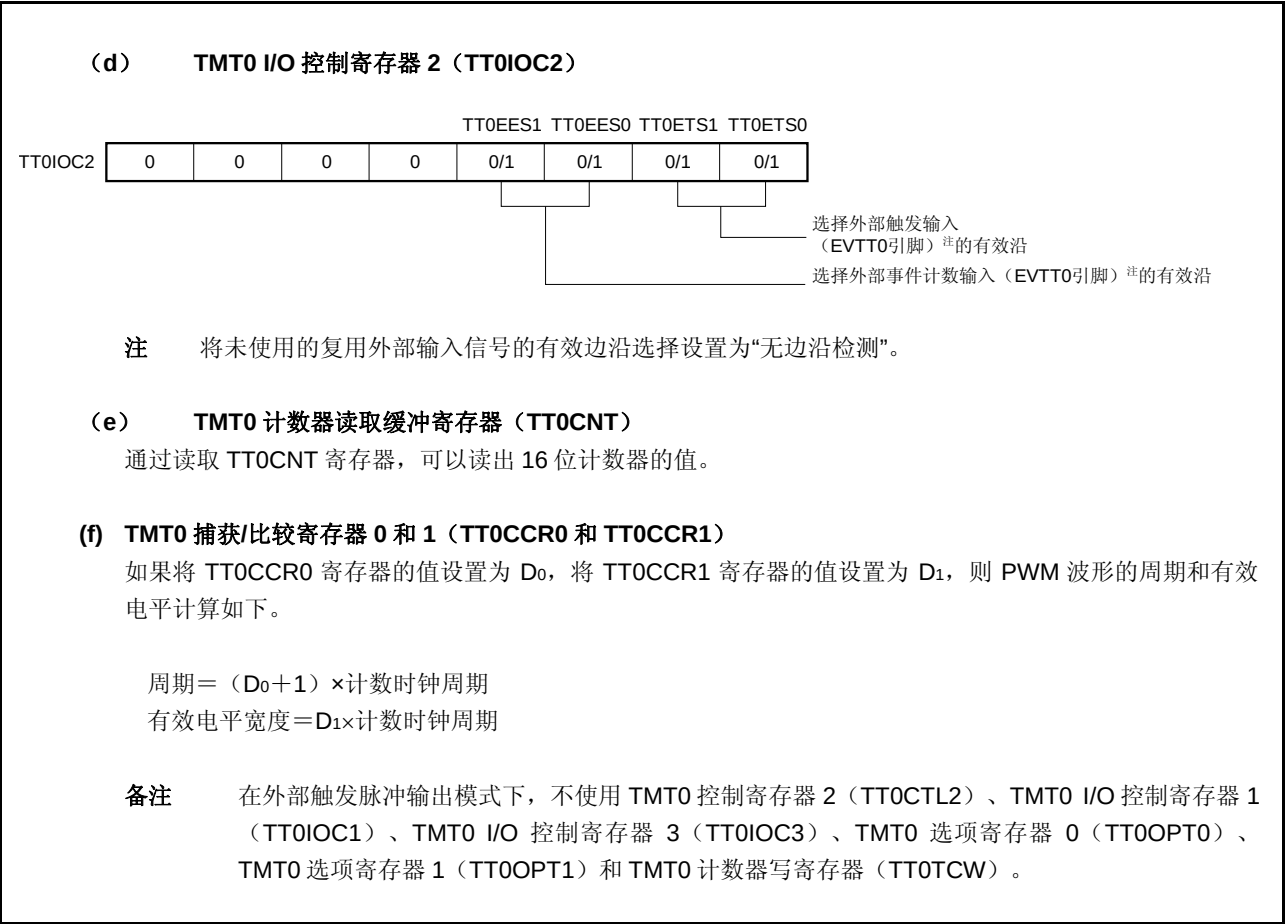


图 9-23. 外部触发脉冲输出模式的寄存器设置 (2/2)



(1) 外部触发脉冲输出模式的工作流程

图 9-24. 外部触发脉冲输出模式的软件处理流程 (1/2)

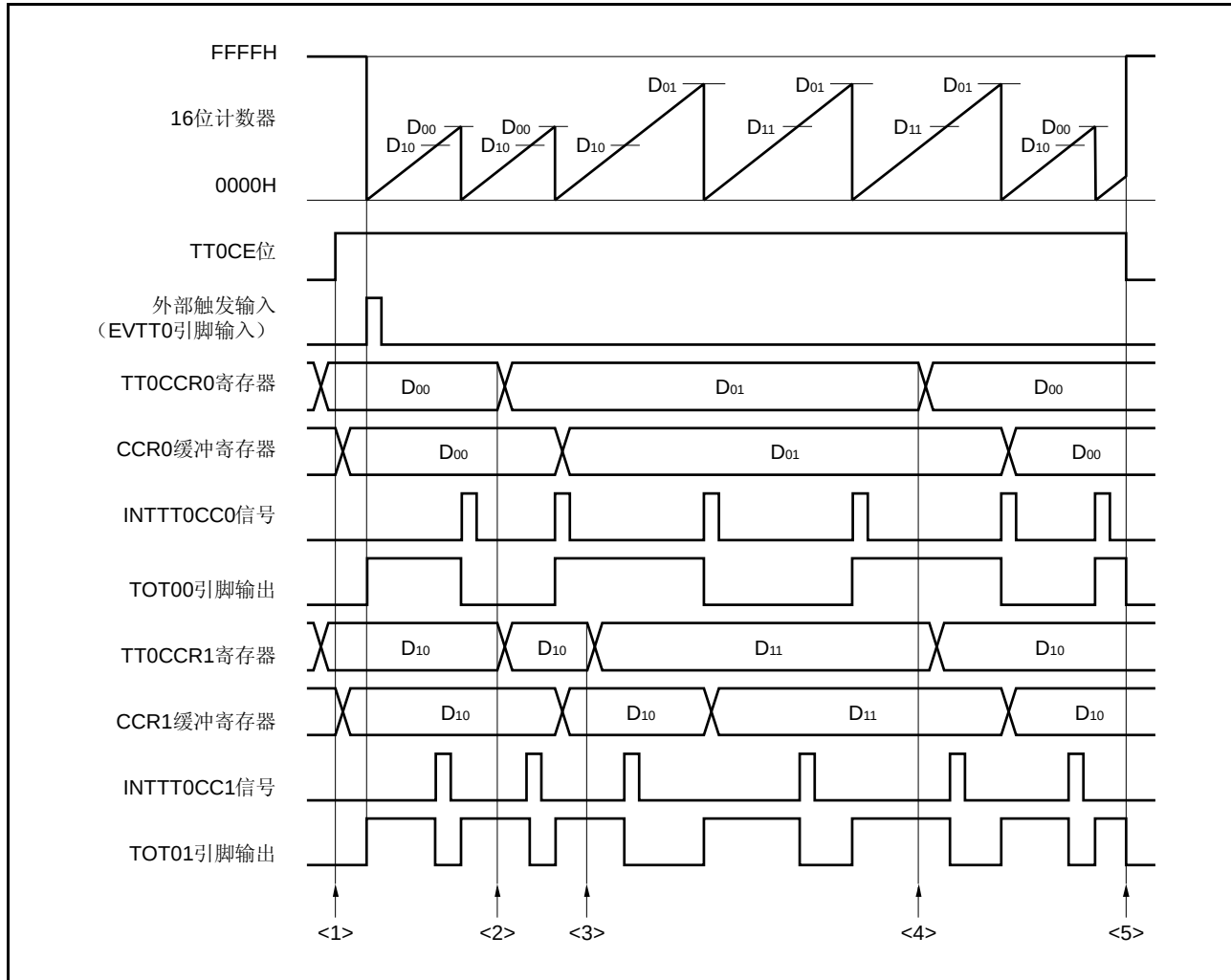
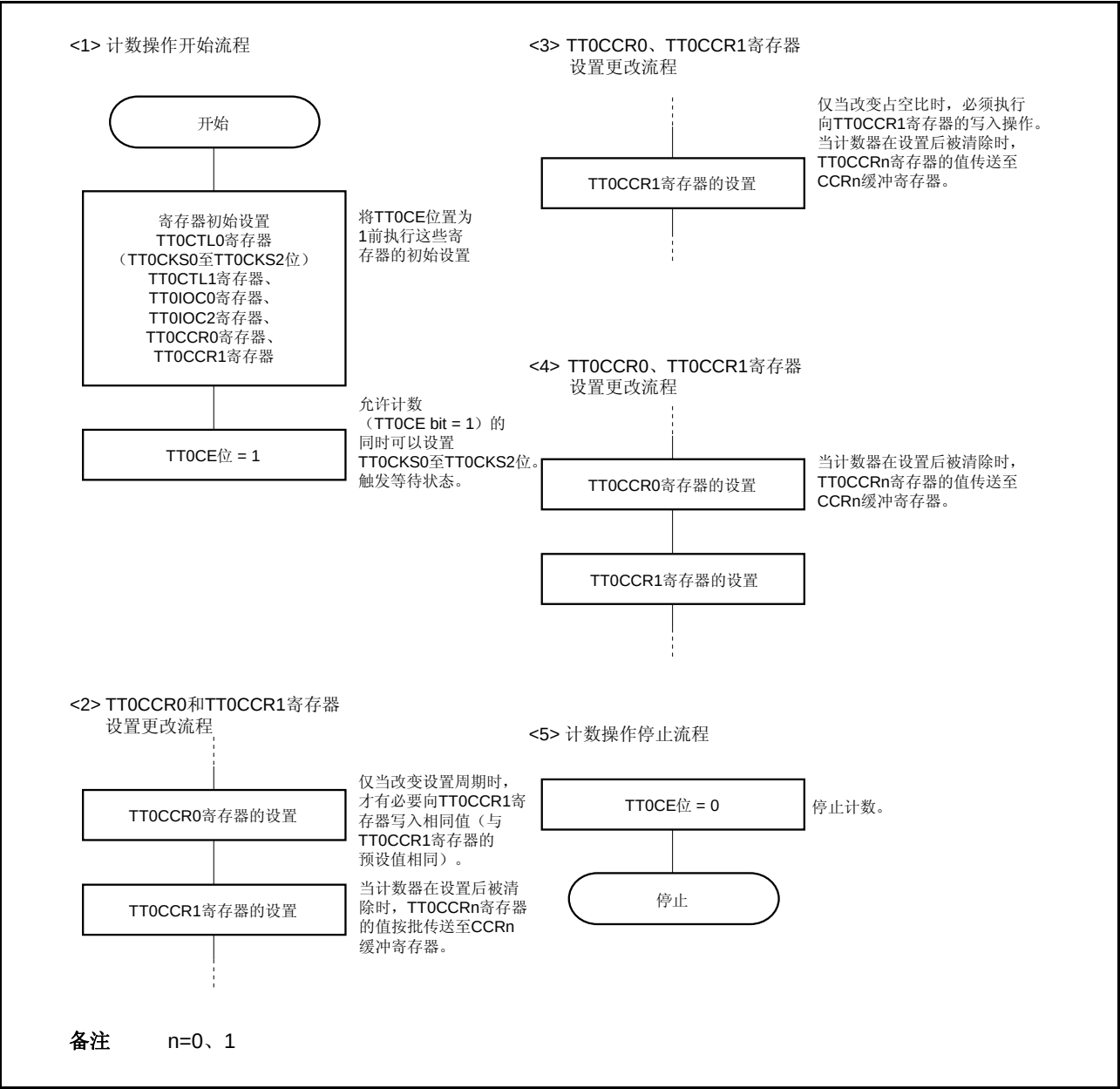


图 9-24. 外部触发脉冲输出模式的软件处理流程（2/2）

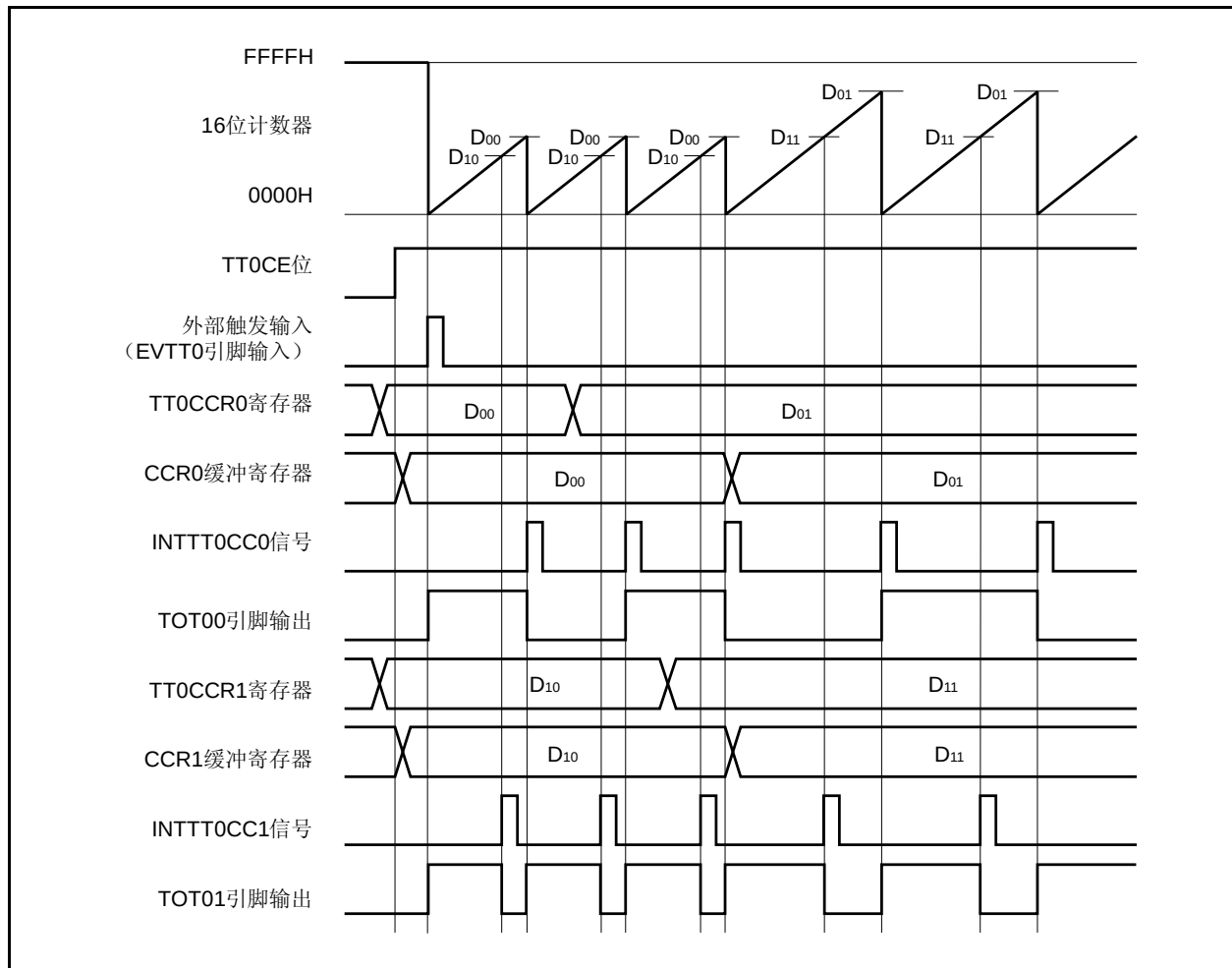


(2) 外部触发脉冲输出模式的操作时序

(a) 有关操作中改变脉冲宽度的注意事项

计数器计数过程中，若要改变 PWM 波形，应最后写入 TT0CCR1 寄存器。

检测到 INTT0CC0 信号以后，在写入 TT0CCR1 寄存器之后重写 TT0CCRn 寄存器。



为了将数据从 TT0CCRn 寄存器传送到 CCRn 缓冲寄存器，必须写入 TT0CCR1 寄存器。

此时，要改变 PWM 波形的周期和有效电平宽度，应首先将周期设定给 TT0CCR0 寄存器，然后将有效电平宽度设定给 TT0CCR1 寄存器。

如果只改变 PWM 波形的周期，应首先把周期设置给 TT0CCR0 寄存器，然后将相同值（与 TT0CCR1 寄存器预设值相同的值）写入 TT0CCR1 寄存器。

只改变 PWM 波形的有效电平宽度（占空比）时，那就只需要设置 TT0CCR1 寄存器。

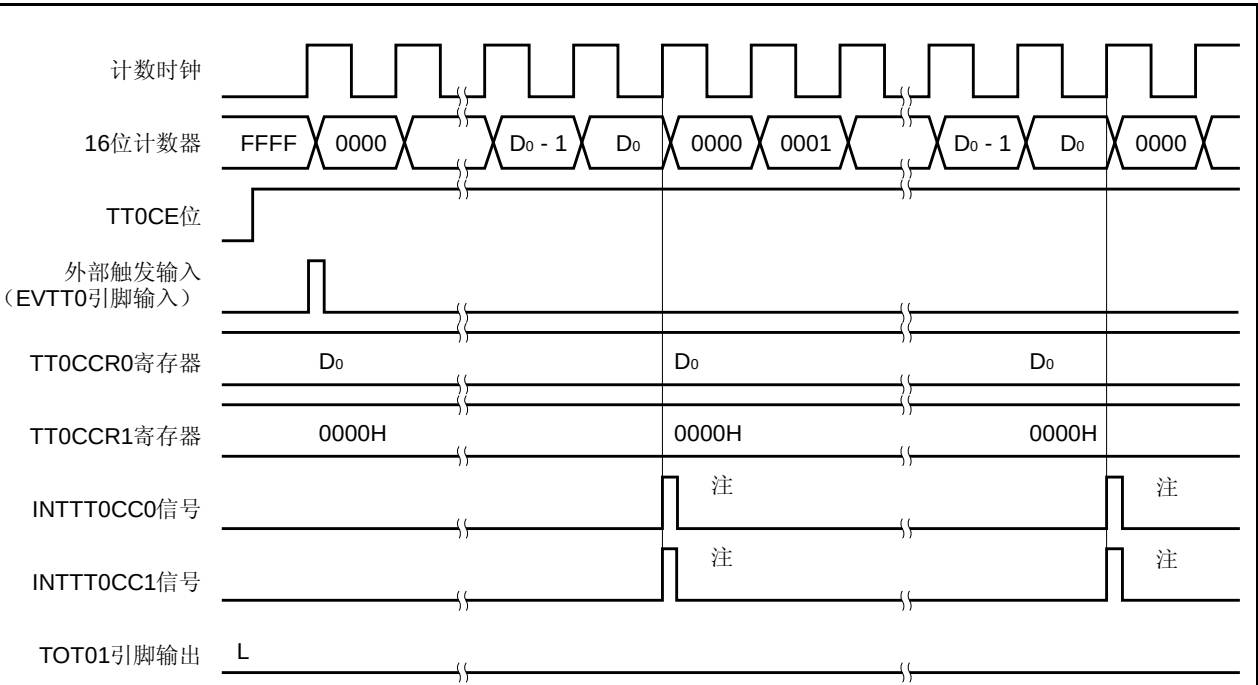
在数据写入 TT0CCR1 寄存器后，会与 16 位计数器清零同步，将写入 TT0CCRn 寄存器的值传送到 CCRn 缓冲寄存器，并作为与 16 位计数器进行比较的值。

要在写入 TT0CCR1 寄存器一次之后，再次写 TT0CCR0 或 TT0CCR1 寄存器，应在 INTTT0CC0 信号生成之后进行。否则，CCRn 缓冲寄存器的值可能不确定，因为把数据从 TT0CCRn 寄存器传送到 CCRn 缓冲寄存器的时序与写 TT0CCRn 寄存器的时序发生冲突。

备注 n=0、1

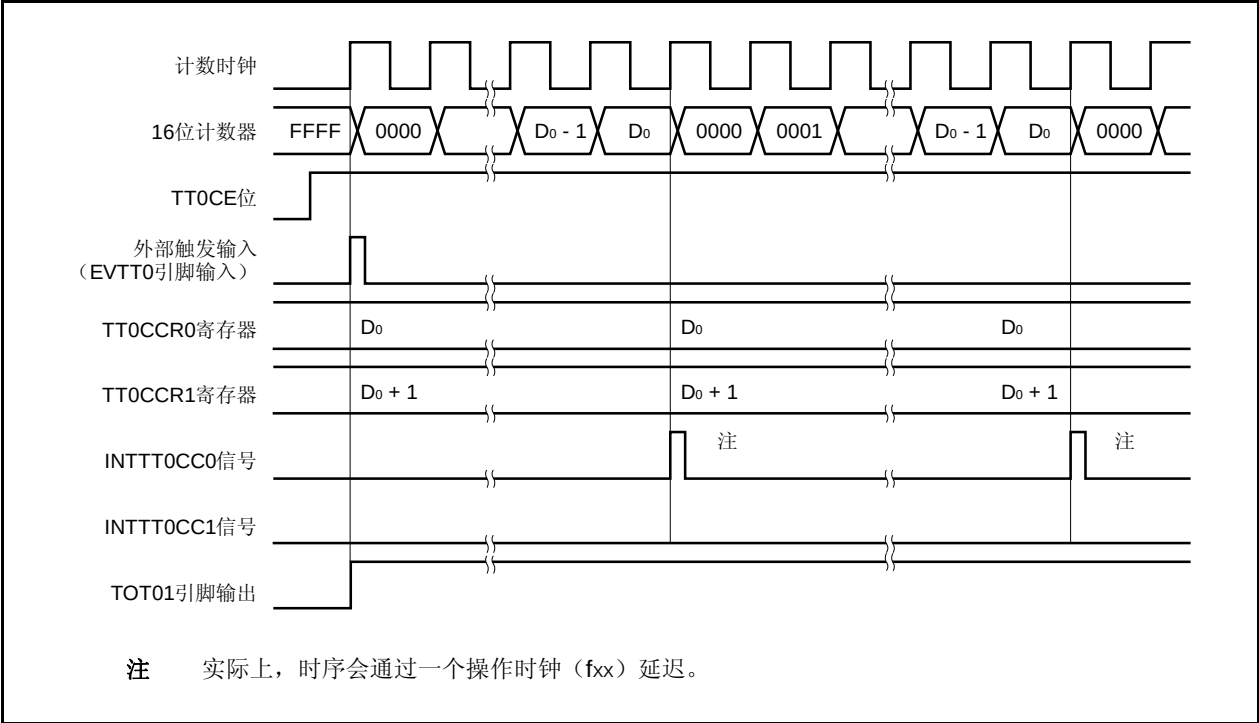
(b) PWM 波形的 0%和 100% 输出

为了输出 0% 的波形，需要将 TT0CCR1 寄存器设置为 0000H。在 16 位计数器的值与 CCR0 缓冲寄存器的值匹配后，16 位计数器将会被清除为 0000H 且会生成 INTTT0CC0 和 INTTT0CC1 信号。



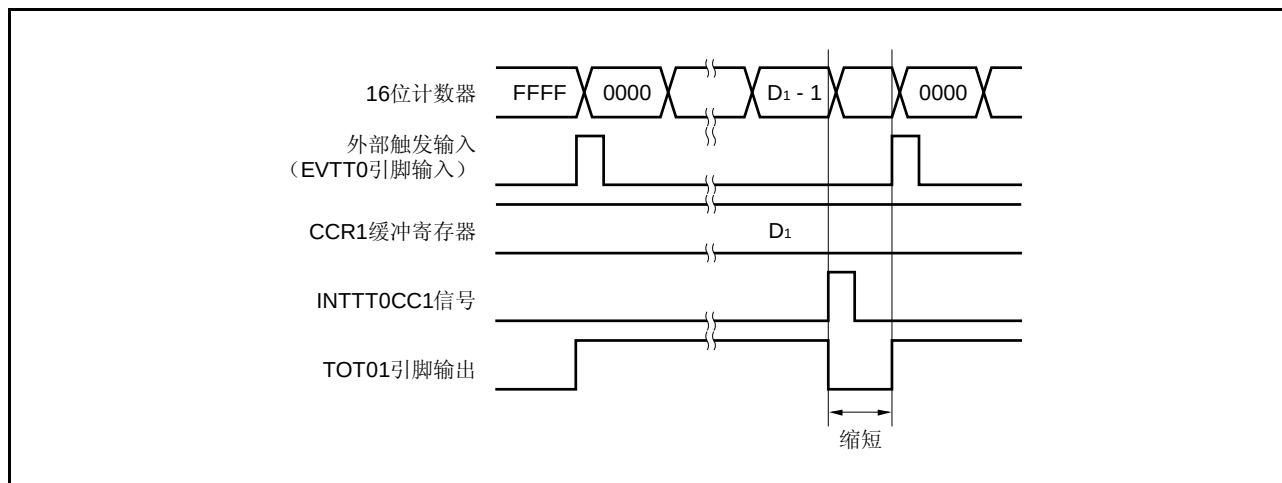
注 实际上，时序会通过一个操作时钟（f_{xx}）延迟。

为了输出 100%的波形，需要将 TT0CCR1 寄存器的值设为：（TT0CCR0 寄存器的设定值 + 1）。如果 TT0CCR0 寄存器的设定值为 FFFFH，则不会产生 100%波形输出。

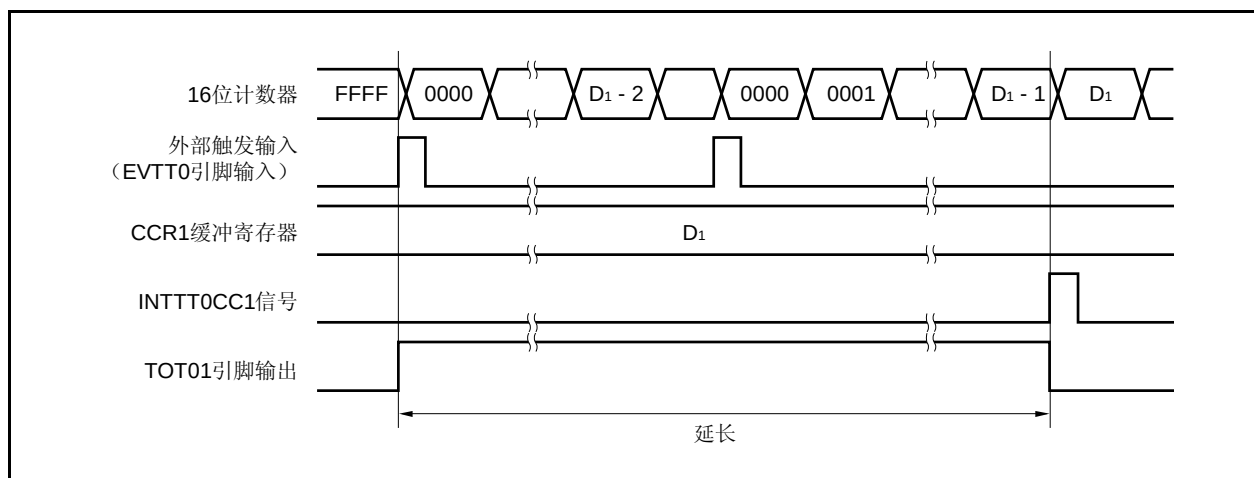


(c) 触发检测和 CCR1 缓冲寄存器匹配之间的冲突

如果在紧接 INTTT0CC1 信号产生后，立刻检测到触发信号，则 16 位计数器立刻清除为 0000H，TOT01 引脚被置位生效，计数器继续计数。结果，缩短了 PWM 波形的无效时段。

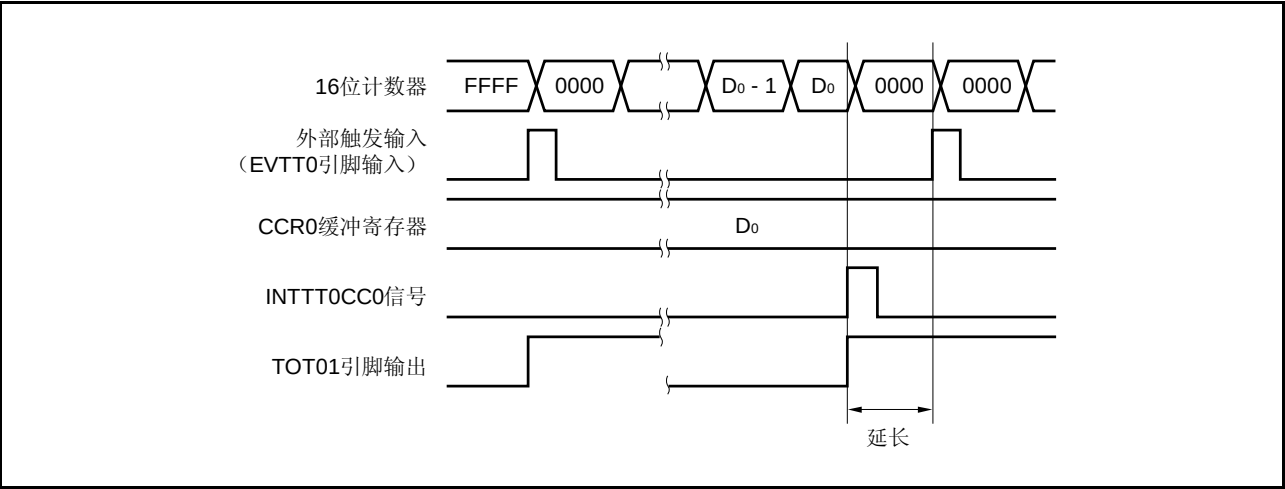


如果刚好在 INTTT0CC1 信号产生前检测到触发信号，则不会生成 INTTT0CC1 信号，而 16 位计数器清零为 0000H 并继续进行计数。TOT01 引脚的输出信号仍然有效。结果，延长了 PWM 波形的有效时域。

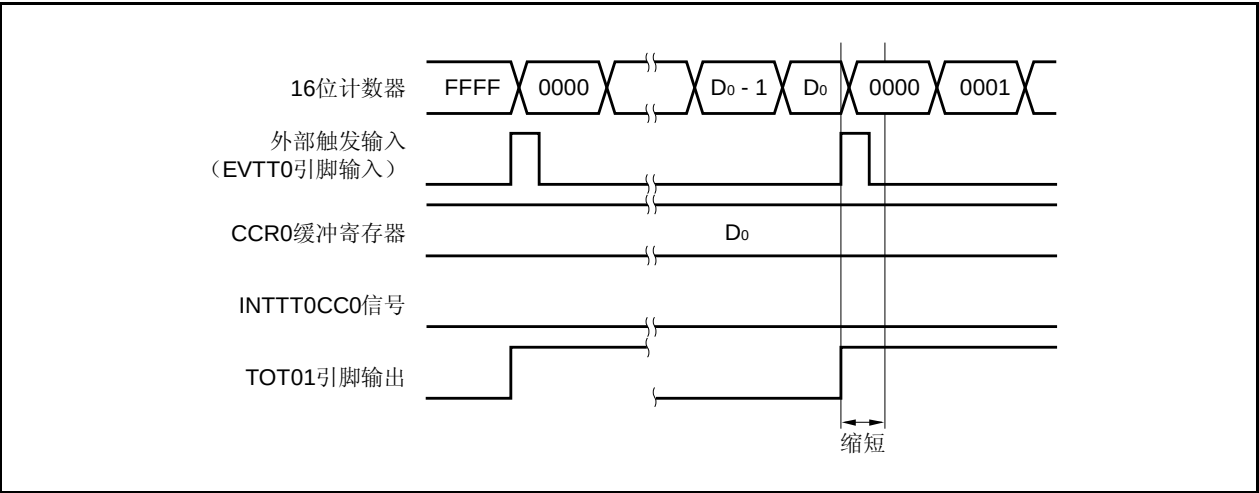


(d) 触发检测和 CCR0 缓冲寄存器匹配之间的冲突

如果刚好在 INTTT0CC0 信号产生后检测到触发，则 16 位计数器清零为 0000H，并继续计数。因此，TOT01 引脚的有效时段被延长，延长时间为自 INTTT0CC0 信号产生时刻到触发被检测到时刻。

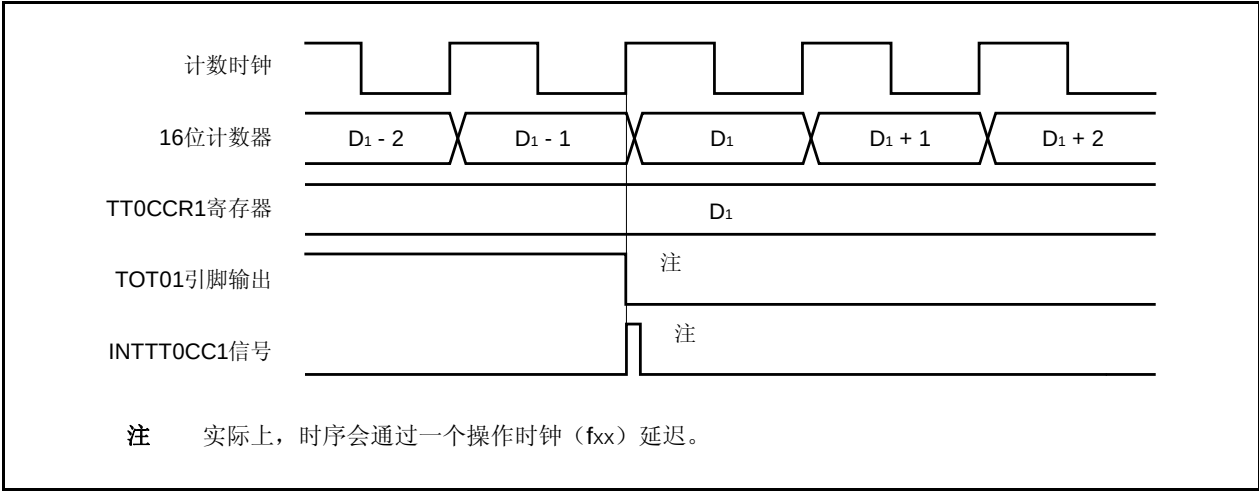


如果刚好在产生 INTTT0CC0 信号之前检测到触发信号，则不会生成 INTTT0CC0 信号。16 位计数器清除为 0000H，TOT01 引脚置位生效，计数器继续计数。结果，缩短了 PWM 波形的无效时段。



(e) 比较匹配中断请求信号（INTTT0CC1）的生成时序

在外部触发脉冲输出模式下生成 INTTT0CC1 信号的时序与在其它模式下生成 INTTT0CC1 信号的时序不同；当 16 位计数器的计数值与 TT0CCR1 寄存器的值匹配时，生成 INTTT0CC1 信号。



通常情况下，在 16 位计数器的计数值与 TT0CCR1 寄存器的值匹配之后，INTTT0CC1 信号的生成与下一次计数同步发生。

但是，在外部触发脉冲输出模式下，其生成会提早一个计数时钟周期。这是由于时序发生了变化，以便与 TOT01 引脚输出信号变化了的时序匹配。

9.6.4 单次脉冲输出模式 (TT0MD3 至 TT0MD0 位=0011)

在单次脉冲输出模式下，当 TT0CTL0.TT0CE 位置为 1 时，16 位定时器/事件计数器 T 等待触发。在检测到外部触发输入 (EVTTO) 的有效边沿时，16 位定时器/事件计数器 T 开始计数，同时从 TOT01 引脚输出一个单次脉冲。

除外部触发输入 (EVTTO) 外，也可生成软件触发来输出该脉冲。在使用软件触发时，TOT00 引脚在 16 位计数器计数时输出有效电平，在计数器停止（等待触发）时输出无效电平。

图 9-25. 单次脉冲输出模式的配置

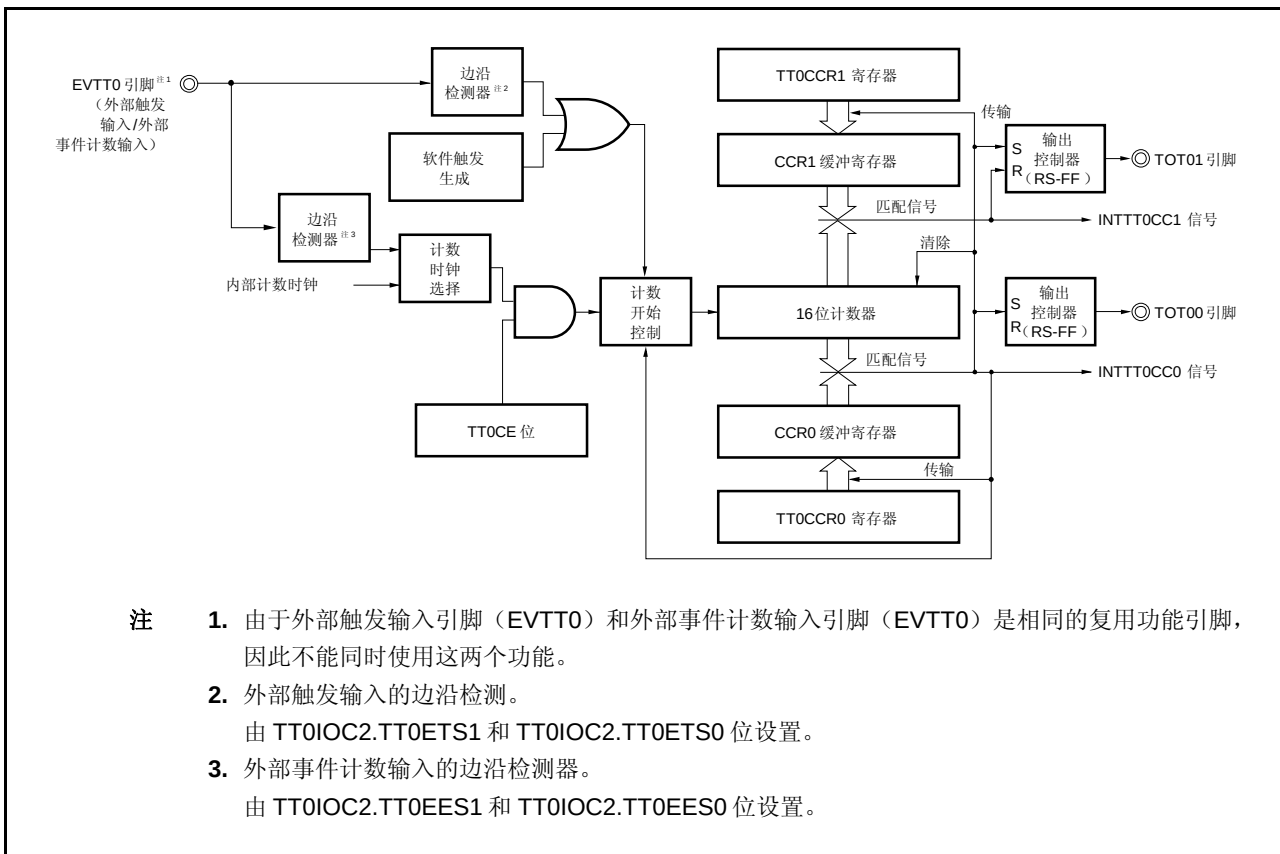
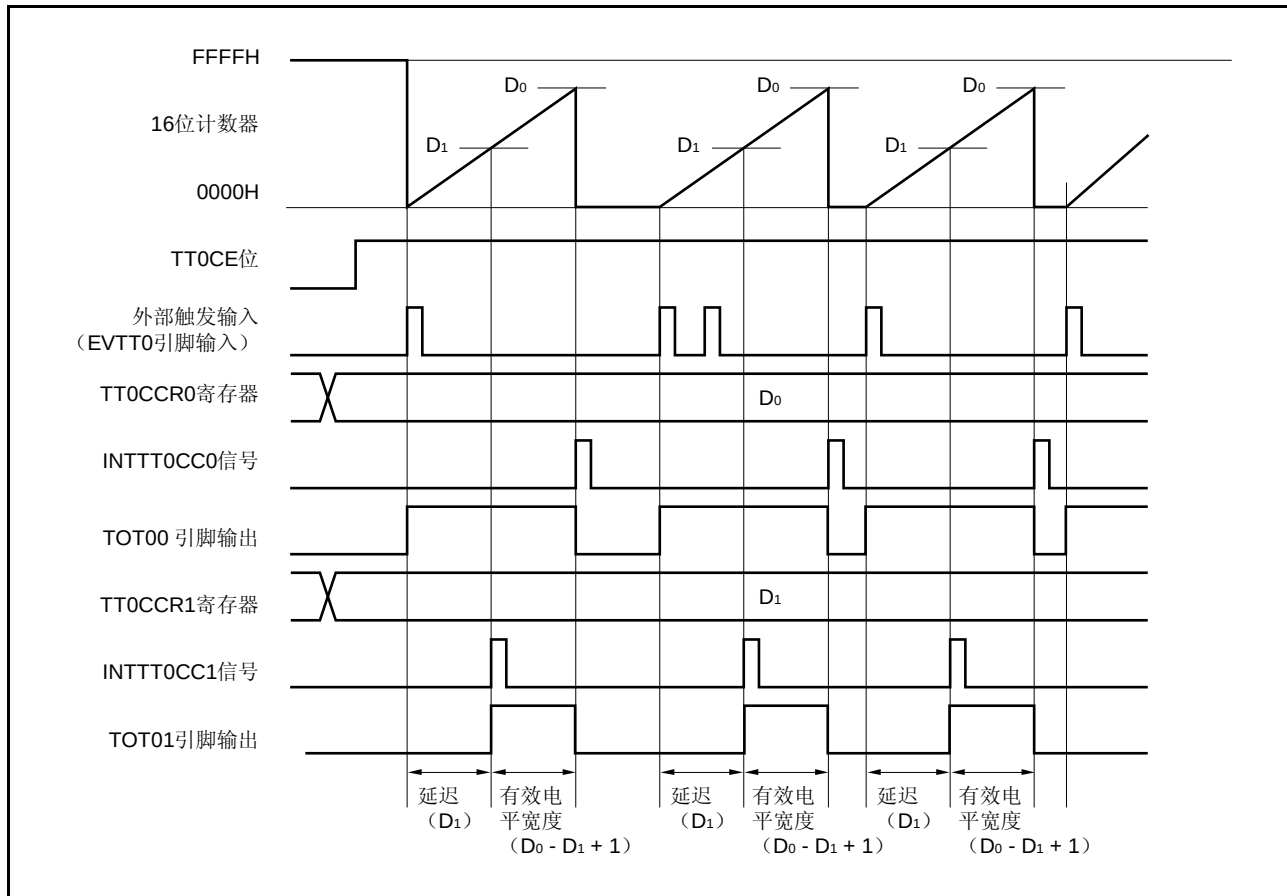


图 9-26. 单次脉冲输出模式的基本时序



当 TT0CE 位设置为 1 时，16 位定时器/事件计数器 T 等待触发。在触发发生时，16 位计数器从 FFFFH 清除为 0000H，开始进行计数，并从 TOT01 引脚输出一个单次脉冲。在输出单次脉冲后，16 位计数器被清除为 0000H 时，计数停止，并重新等待触发。当再次产生触发时，16 位计数器从 0000H 开始计数。当在输出单次脉冲期间再次发生触发，则忽略该触发。

单次脉冲的输出延迟时间和有效电平宽度可计算如下：

输出延迟时间 = (TT0CCR1 寄存器的设定值) × 计数时钟周期

有效电平宽度 = (TT0CCR0 寄存器的设定值 - TT0CCR1 寄存器的设定值 + 1) × 计数时钟周期

当 16 位计数器在其计数值与 CCR0 缓冲寄存器的值匹配之后进行下一个计数时，生成比较匹配中断请求信号 (INTTT0CC0)。当 16 位计数器的计数值与 CCR1 缓冲寄存器的值匹配时，会生成比较匹配中断请求信号 (INTTT0CC1)。

外部触发输入 (EVTTO 引脚) 的有效边沿，或将软件触发 (TT0CTL1.TT0EST 位) 设为 1，都可当作触发信号。

图 9-27. 单次脉冲输出模式时的寄存器设置 (1/2)

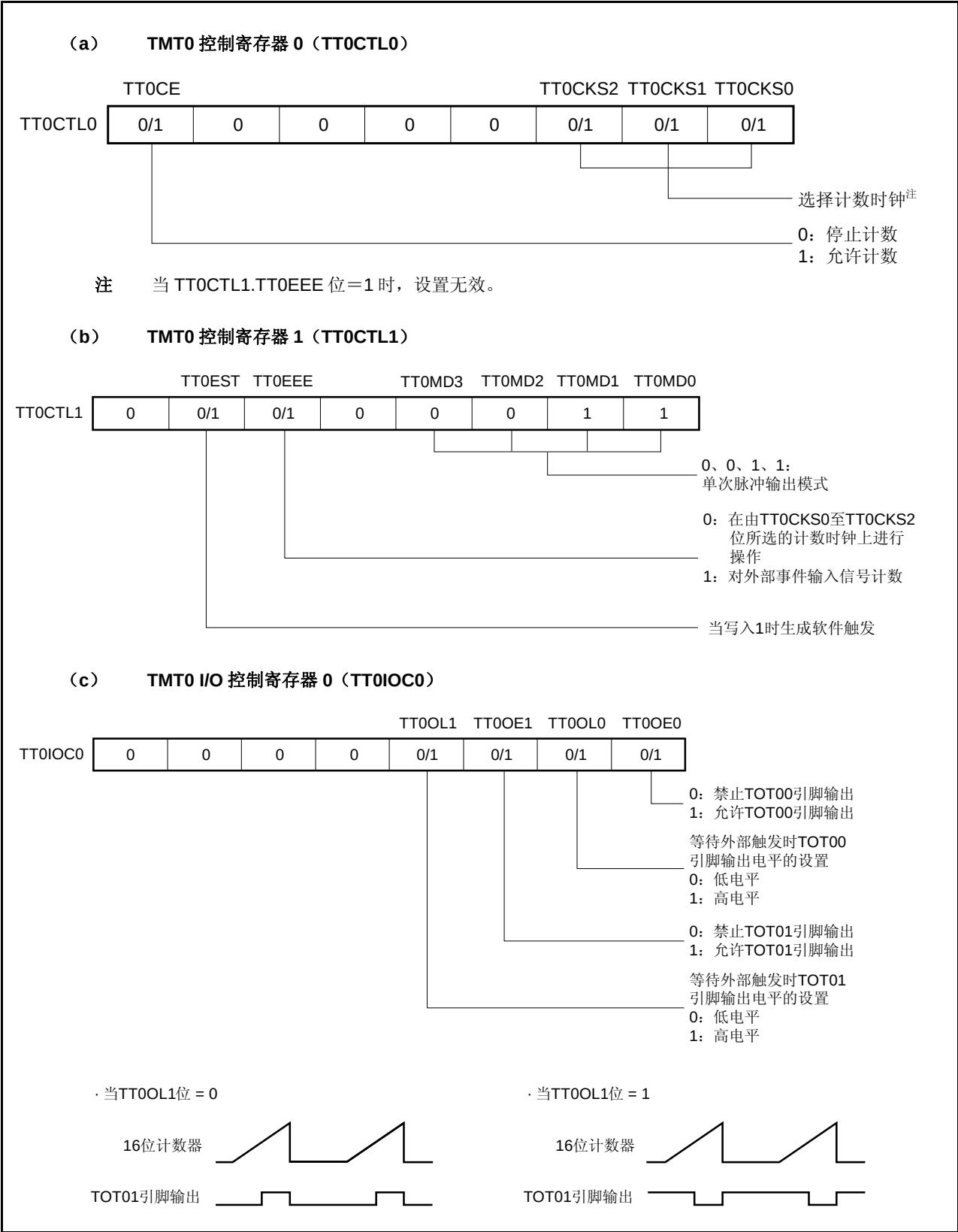
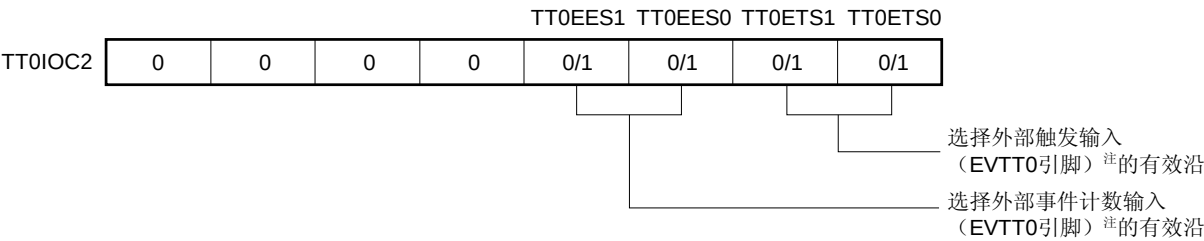


图 9-27. 单次脉冲输出模式时的寄存器设置 (2/2)

(d) TMT0 I/O 控制寄存器 2 (TT0IOC2)



注 将未使用的复用外部输入信号的有效边沿选择设置为“无边沿检测”。

(e) TMT0 计数器读取缓冲寄存器 (TT0CNT)

通过读取 TT0CNT 寄存器，可以读出 16 位计数器的值。

(f) TMT0 捕获/比较寄存器 0 和 1 (TT0CCR0 和 TT0CCR1)

若把 TT0CCR0 寄存器的值设置为 D₀，TT0CCR1 寄存器的值设置为 D₁，则单次脉冲的有效电平宽度和输出延迟期计算如下。

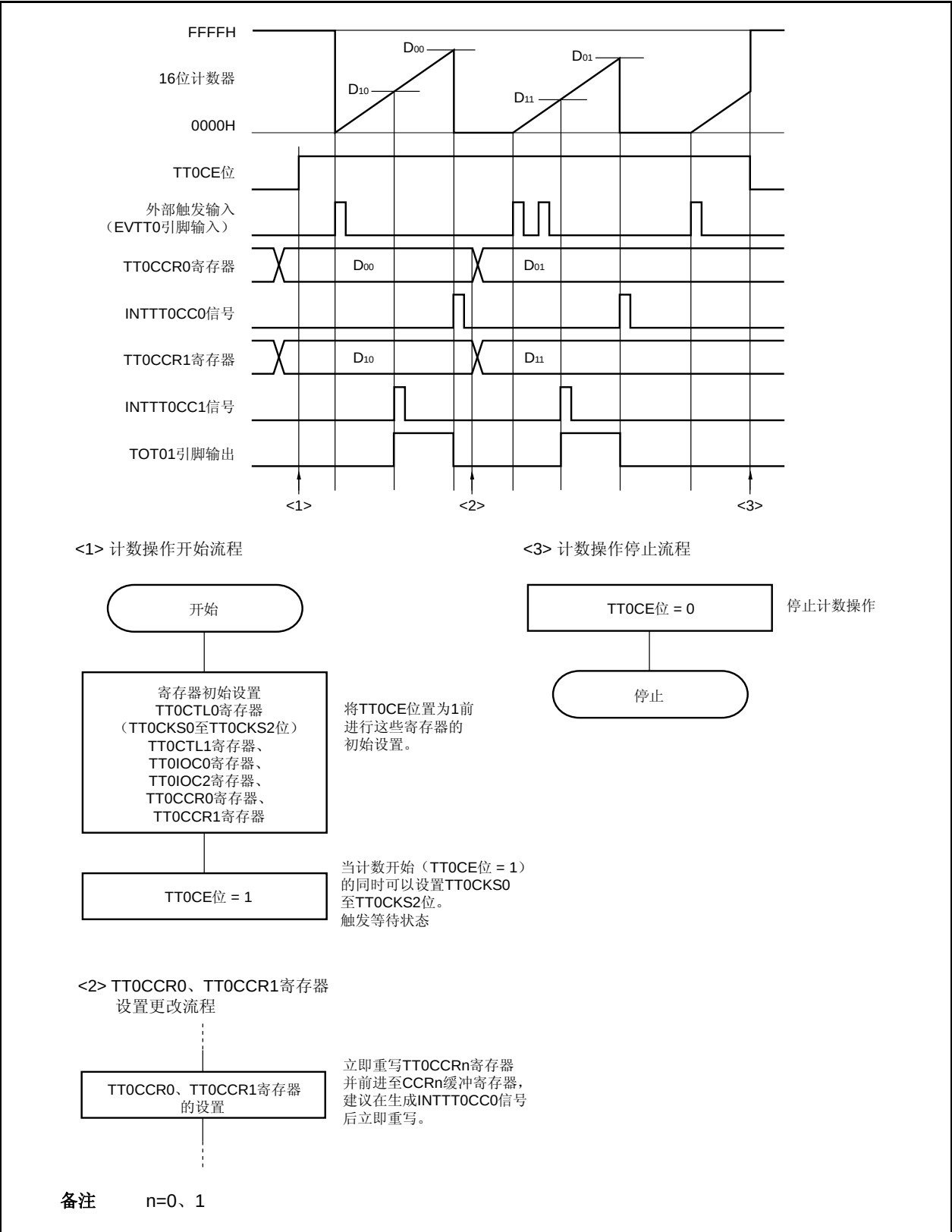
有效电平宽度 = (D₀ - D₁ + 1) × 计数时钟周期

输出延迟时间 = D₁ × 计数时钟周期

备注 在单次脉冲输出下，不使用 TMT0 控制寄存器 2 (TT0CTL2)、TMT0 I/O 控制寄存器 1 (TT0IOC1)、TMT0 I/O 控制寄存器 3 (TT0IOC3)、TMT0 选项寄存器 0 (TT0OPT0)、TMT0 选项寄存器 1 (TT0OPT1) 和 TMT0 计数器写寄存器 (TT0TCW)。

(1) 单次脉冲输出模式下的操作流程

图 9-28. 单次脉冲输出模式下的软件处理流程

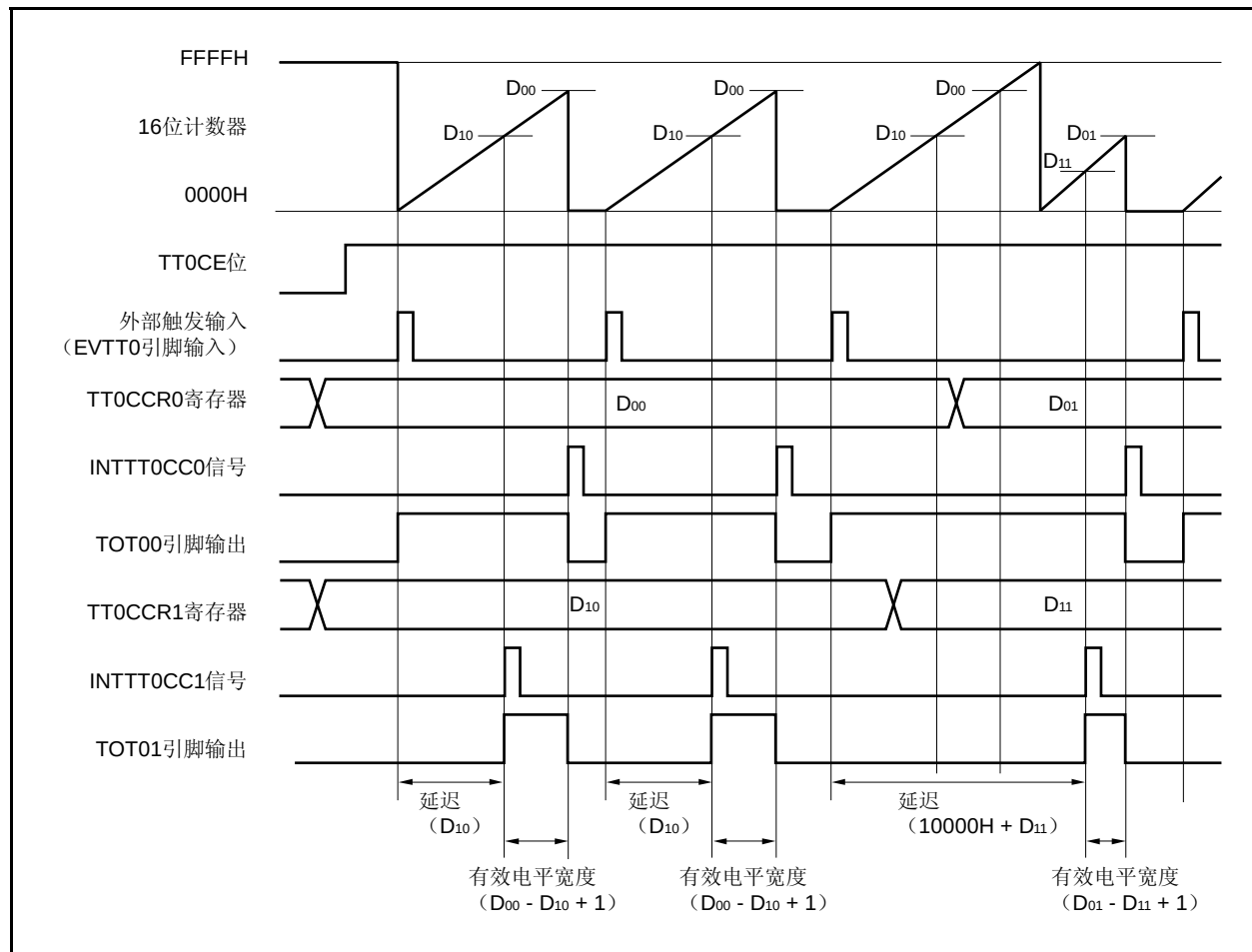


(2) 单次脉冲输出模式时的工作时序

(a) 关于重写 TT0CCRn 寄存器的注意事项

如果在计数器计数期间，将寄存器 TT0CCRn 的值重写为一个较小点的值，则 16 位计数器可能发生上溢出。当可能发生上溢出时，停止计数，然后更改设定值。

备注 $n = 0, 1$

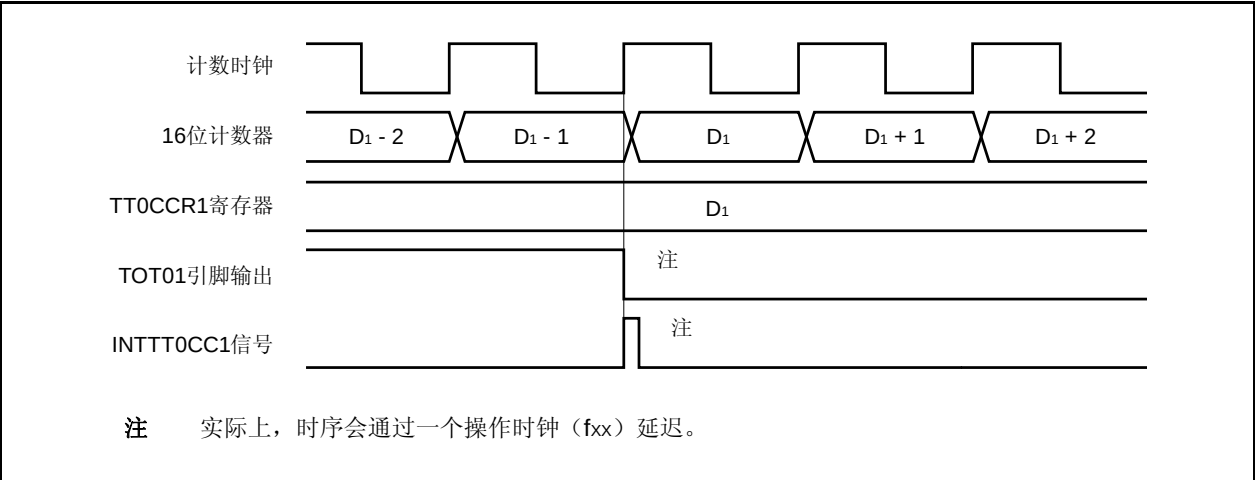


在 $D_{00} > D_{01}$ 且 $D_{10} > D_{11}$ 处，当 TT0CCR0 寄存器从 D_{00} 重写为 D_{01} ，TT0CCR1 寄存器从 D_{10} 重写为 D_{11} 时，若 TT0CCR1 寄存器在 16 位计数器的计数值大于 D_{11} 且小于 D_{10} 时重写，以及如果 TT0CCR0 寄存器在计数值大于 D_{01} 且小于 D_{00} 时重写，则一旦重写寄存器，就会在各设置值上反映出来并立即与计数值进行比较。计数器向上计数到 FFFFH 后，从 0000H 开始重新向上计数。当计数值与 D_{11} 匹配时，计数器会生成 INTTT0CC1 信号并使 TOT01 引脚置位有效。当计数值与 D_{01} 匹配时，计数器生成 INTTT0CC0 信号，使 TOT01 引脚清除置位，并停止计数。

因此，计数器输出的脉冲可能带有延迟期或其有效期与原先预计的单次脉冲有所不同。

(b) 比较匹配中断请求信号 (INTTT0CC1) 的生成时序

单次脉冲输出模式下，INTTT0CC1 信号的生成时序和其它模式下的 INTTT0CC1 信号不同，当 16 位计数器的计数值与 TT0CCR1 寄存器的值匹配时，就会生成 INTTT0CC1 信号。



通常情况下，在 16 位计数器的计数值与 TT0CCR1 寄存器的值匹配之后，INTTT0CC1 信号在计数器下一次计数时生成。

但是，在单次脉冲输出模式下，会提早一个计数时钟周期产生。这是由于时序发生了变化，以便与 TOT01 引脚输出信号变化了的时序匹配。

9.6.5 PWM输出模式 (TT0MD3 至 TT0MD0 位=0100)

在 PWM 输出模式下, 当 TT0CTL0.TT0CE 位设定为 1 时, 从 TOT01 引脚输出 PWM 波形。

此外, 带有 50%占空比的方波也可以从 TOT00 引脚中输出, 它的半周期等于 TT0CCR0 寄存器的设定值+1。

图 9-29. PWM 输出模式的配置

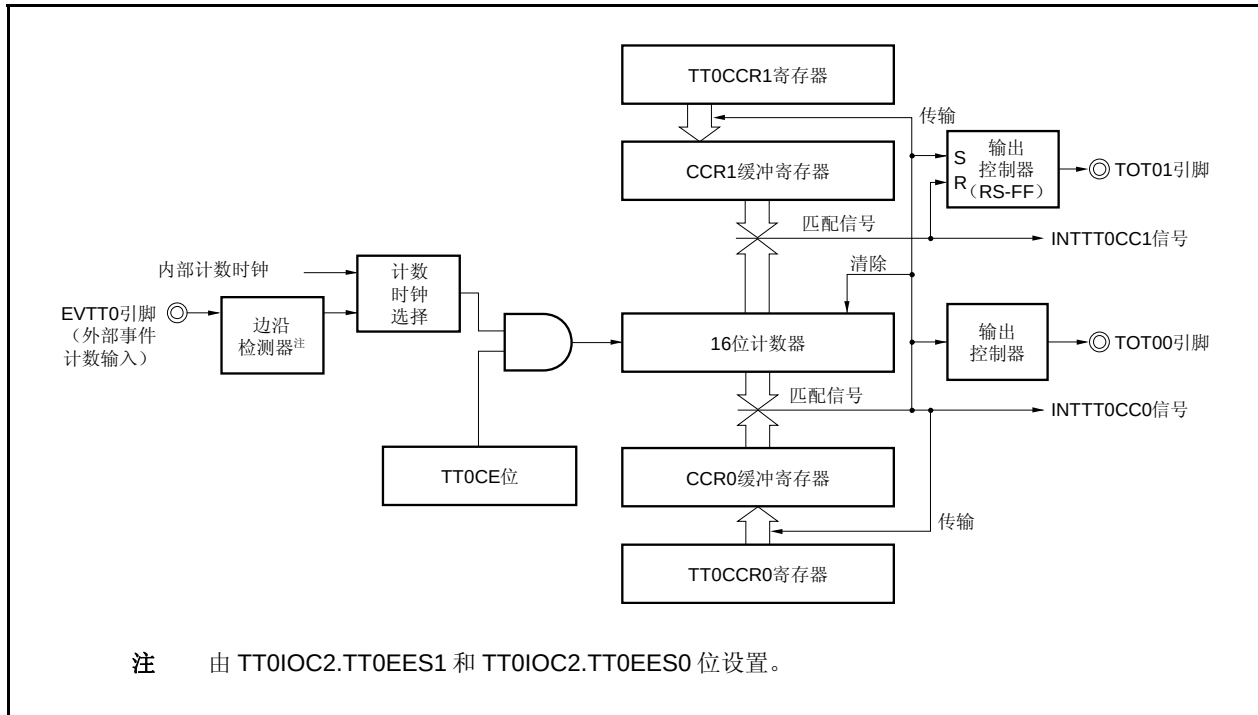
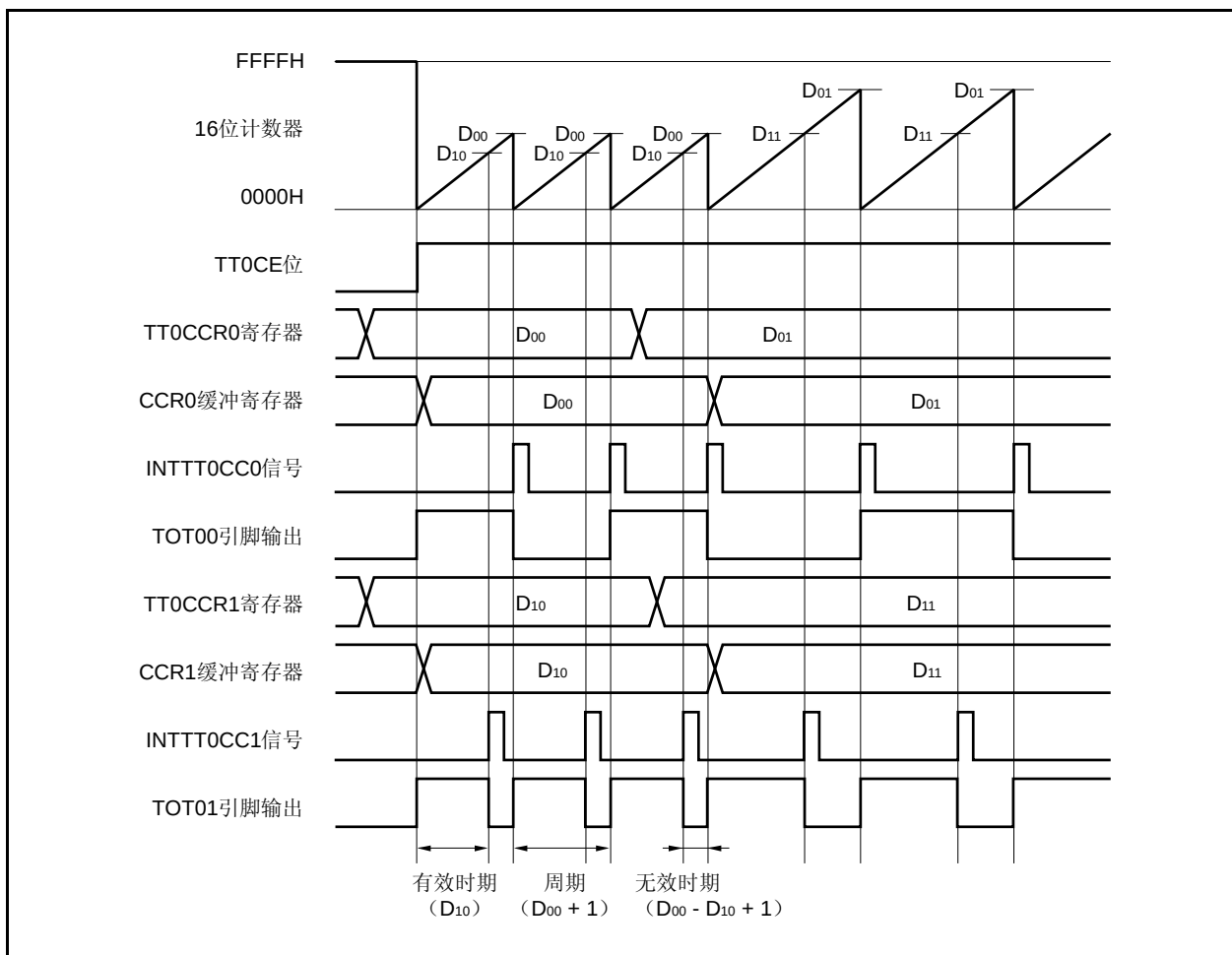


图 9-30. PWM 输出模式的基本时序



当 TT0CE 位设置为 1 时, 16 位计数器从 FFFFH 清零为 0000H, 开始进行计数, 同时从 TOT01 引脚输出 PWM 波形。

PWM 波形的有效电平宽度、周期和占空比计算如下。

$$\text{有效电平宽度} = (\text{TT0CCR1 寄存器的设定值}) \times \text{计数时钟周期}$$

$$\text{周期} = (\text{TT0CCR0 寄存器的设定值} + 1) \times \text{计数时钟周期}$$

$$\text{占空比系数} = (\text{TT0CCR1 寄存器的设置值}) / (\text{TT0CCR0 寄存器的设置值} + 1)$$

在进行计数时，可通过重写 TT0CCRn 寄存器来改变 PWM 波形。当 16 位计数器的计数值与 CCR0 缓冲寄存器的值匹配，且 16 位计数器清零为 0000H 时，新写入的值会得到反映。

当 16 位计数器在其计数值与 CCR0 缓冲寄存器的值匹配之后进行下一次计数时，会生成比较匹配中断请求信号（INTTT0CC0），且 16 位计数器清除为 0000H。当 16 位计数器的计数值与 CCR1 缓冲寄存器的值匹配时，会生成比较匹配中断请求信号（INTTT0CC1）。

当 16 位计数器的计数值与 CCRn 缓冲寄存器的值匹配，且 16 位计数器清零为 0000H 时，TT0CCRn 寄存器设置的值就传送到 CCRn 缓冲寄存器。

备注 $n=0, 1$

图 9-31. PWM 输出模式时的寄存器设置 (1/2)

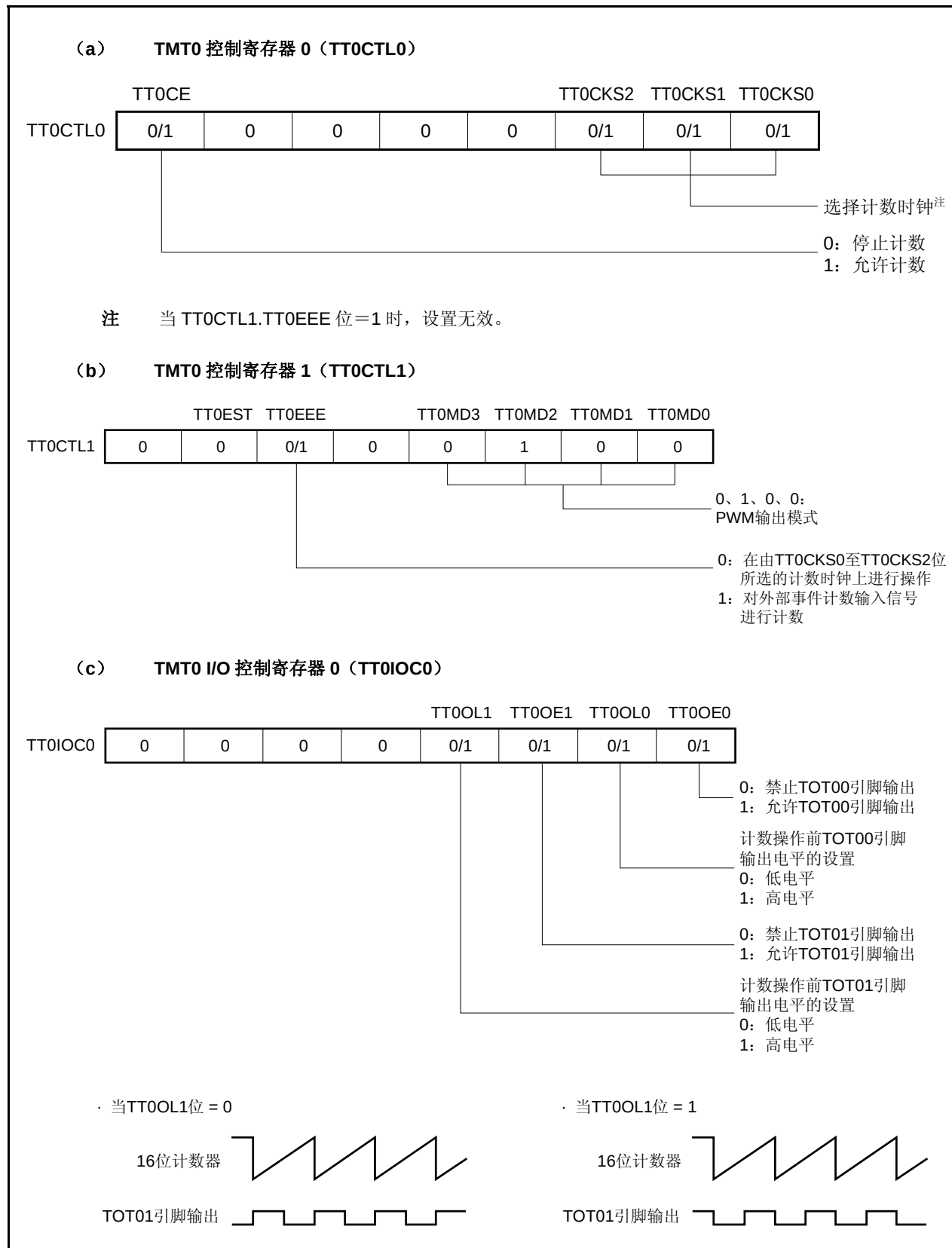
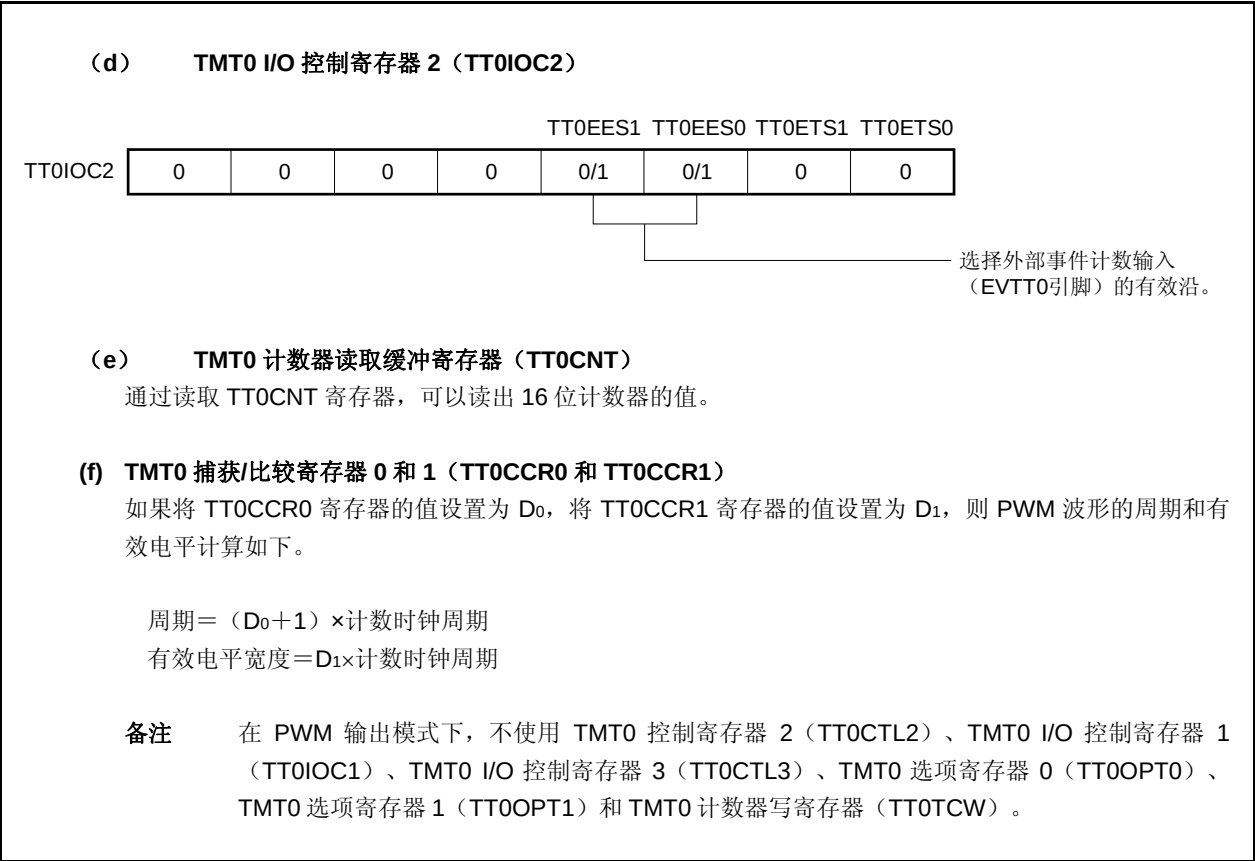


图 9-31. PWM 输出模式时的寄存器设置 (2/2)



(1) PWM 输出模式下的工作流程

图 9-32. PWM 输出模式时的软件处理流程 (1/2)

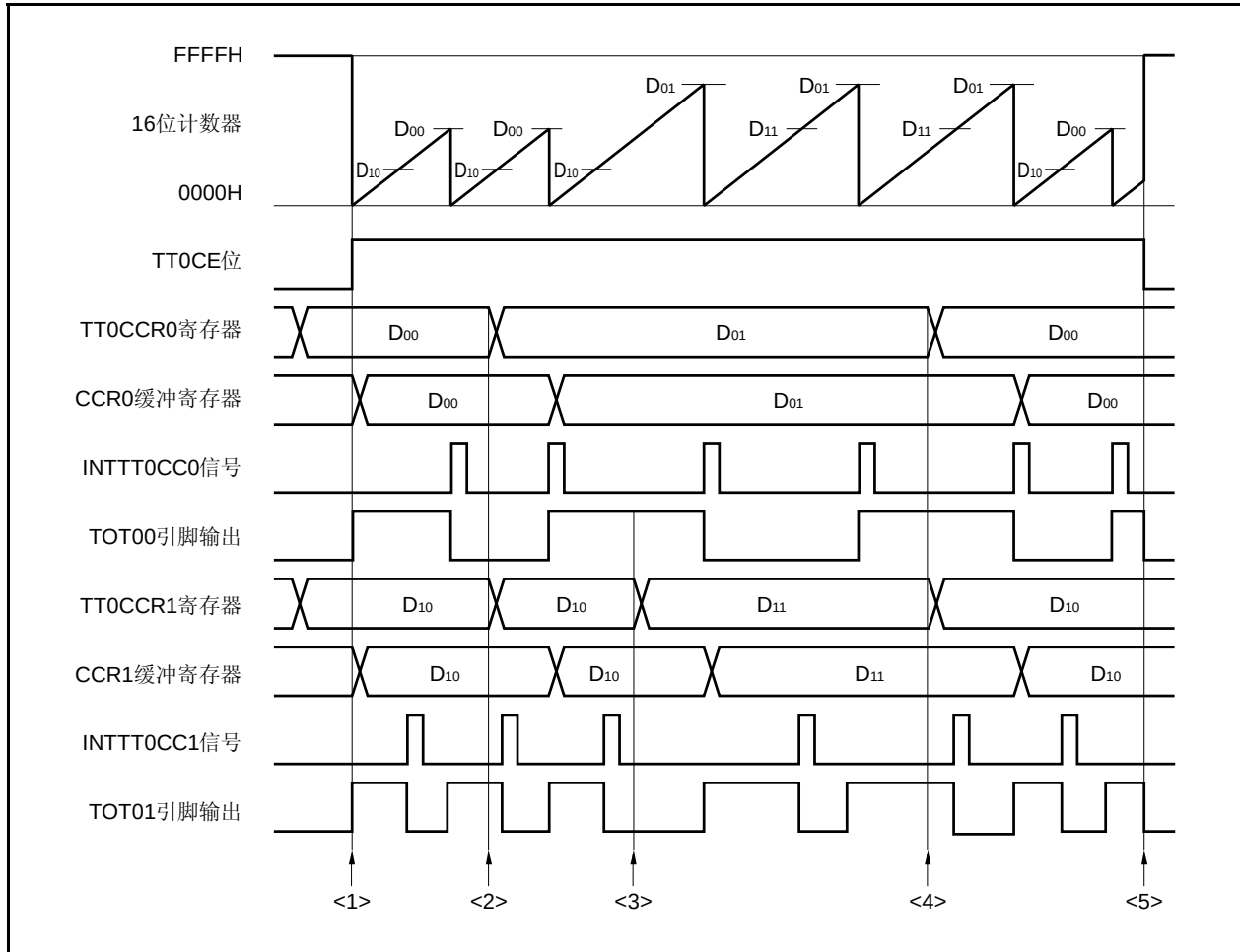
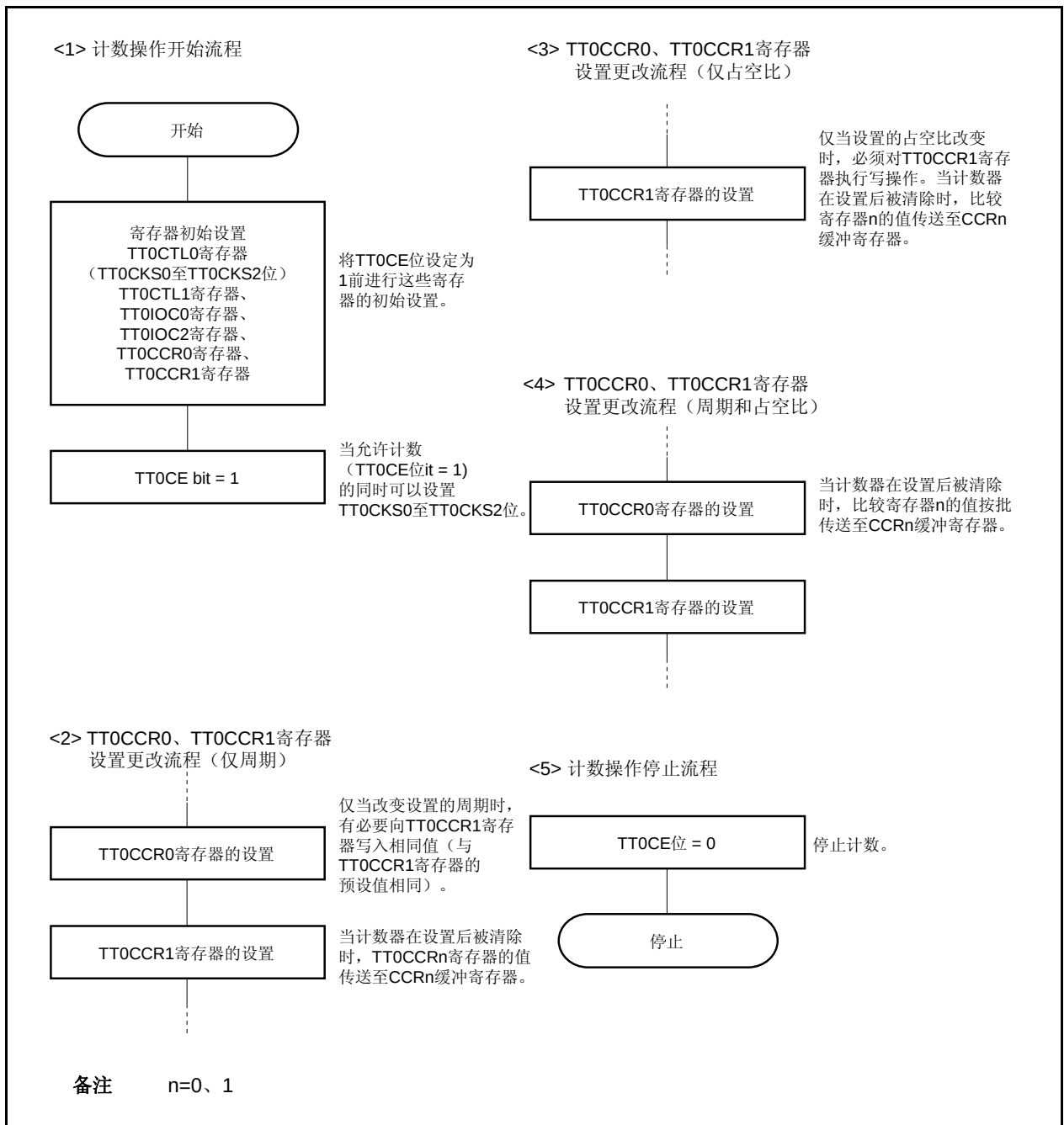


图 9-32. PWM 输出模式时的软件处理流程 (2/2)

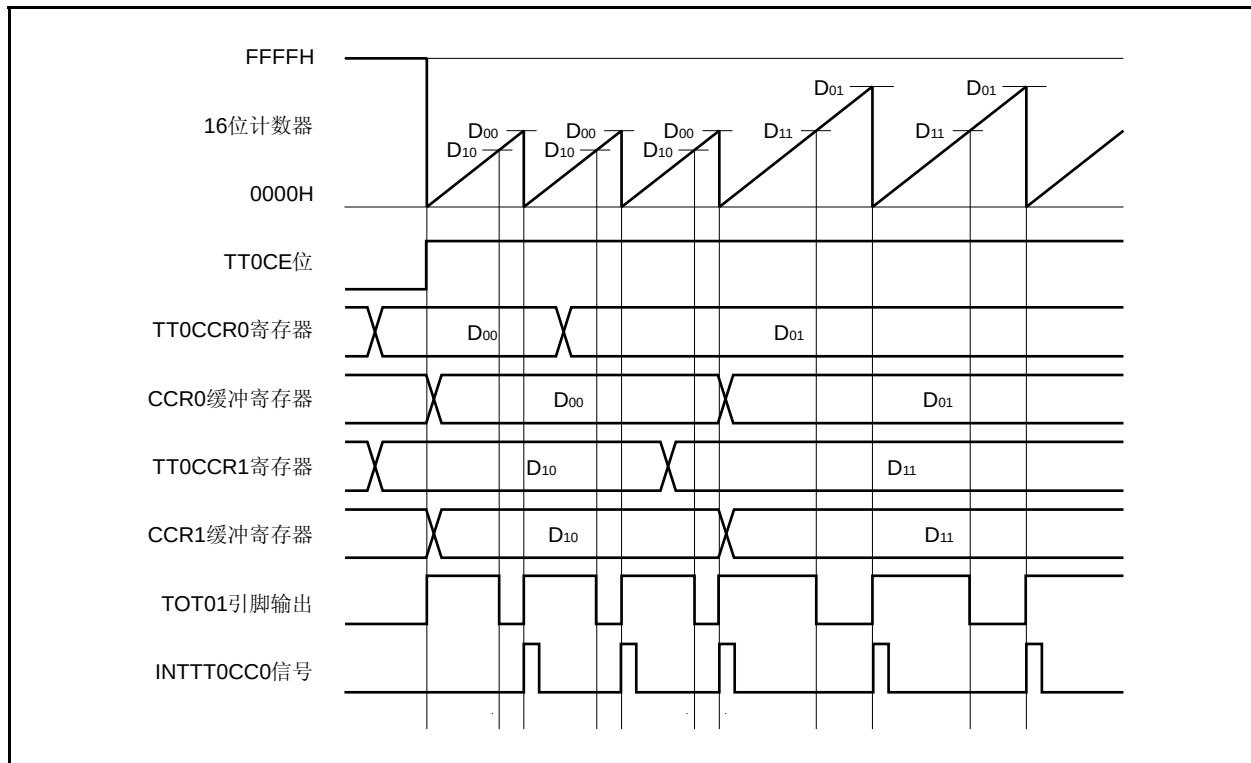


(2) PWM 输出模式的操作时序

(a) 操作过程中改变脉冲宽度

计数器计数过程中，若要改变 PWM 波形，应最后写 TT0CCR1 寄存器。

检测到 INTT0CC1 信号以后，在写入 TT0CCR1 寄存器之后重写 TT0CCRn 寄存器。



要把数据从 TT0CCRn 寄存器传送到 CCRn 缓冲寄存器，必须对 TT0CCR1 寄存器进行写入操作。

此时，要改变 PWM 波形的周期和有效电平，应首先将周期设定给 TT0CCR0 寄存器，然后将有效电平宽度设定给 TT0CCR1 寄存器。

如果只改变 PWM 波形的周期，应首先把周期设置给 TT0CCR0 寄存器，然后将相同值（与 TT0CCR1 寄存器预设值相同的值）写入 TT0CCR1 寄存器。

只改变 PWM 波形的有效电平宽度（占空比）时，那就只需要设置 TT0CCR1 寄存器。

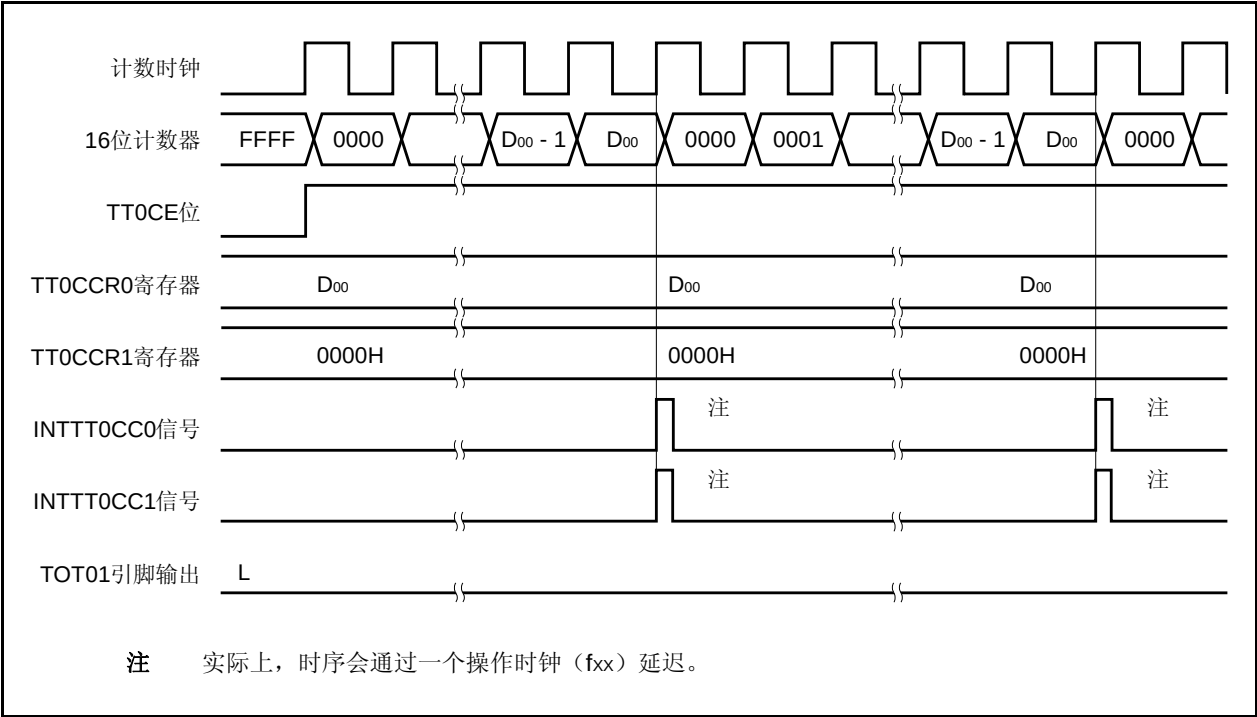
在数据写入 TT0CCR1 寄存器后，会与 16 位计数器清零同步，将写入 TT0CCRn 寄存器的值传送到 CCRn 缓冲寄存器，并作为与 16 位计数器进行比较的值。

要在写入 TT0CCR1 寄存器一次之后，再次写 TT0CCR0 或 TT0CCR1 寄存器，应在 INTT0CC0 信号生成之后进行。否则，CCRn 缓冲寄存器的值可能不确定，因为把数据从 TT0CCRn 寄存器传送到 CCRn 缓冲寄存器的时序与写 TT0CCRn 寄存器的时序发生冲突。

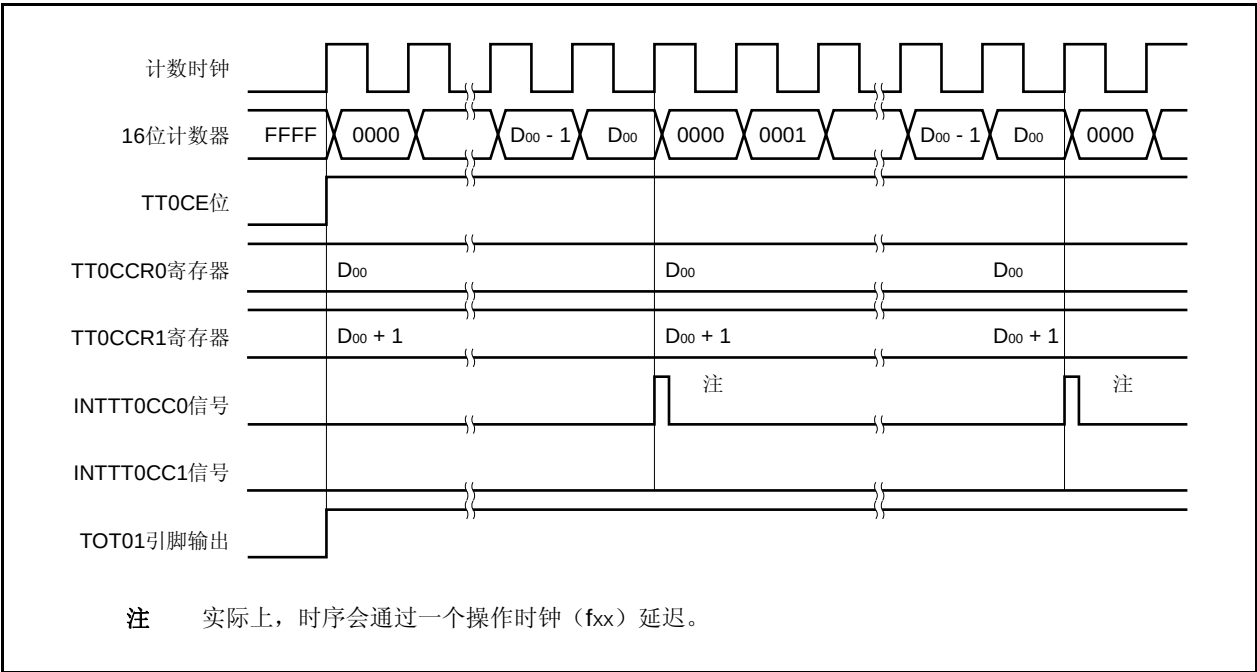
备注 n=0、1

(b) PWM 波形的 0%和 100% 输出

为了输出 0% 的波形，需要将 TT0CCR1 寄存器设置为 0000H。在 16 位计数器的值与 CCR0 缓冲寄存器的值匹配后，16 位计数器将会被清除为 0000H 且会生成 INTTT0CC0 和 INTTT0CC1 信号。

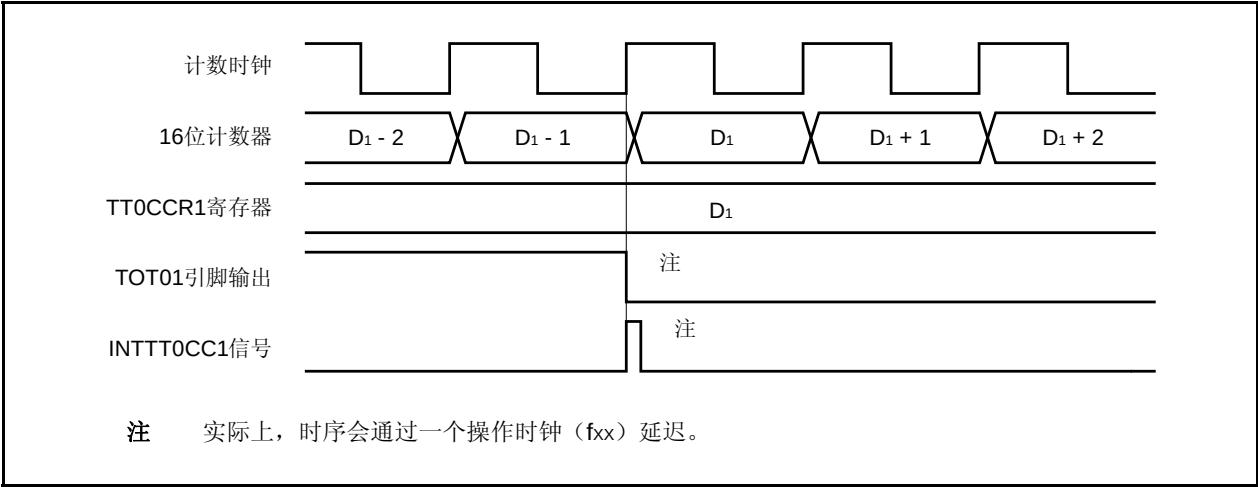


为了输出 100%的波形，需要将 TT0CCR1 寄存器的值设为：（TT0CCR0 寄存器的设定值 + 1）。如果 TT0CCR0 寄存器的设定值为 FFFFH，则不会产生 100%波形输出。



(c) 比较匹配中断请求信号 (INTTT0CC1) 的生成时序

在 PWM 输出模式下，INTTT0CC1 信号的生成时序与其它模式下 INTTT0CC1 信号的时序有所不同；当 16 位计数器的计数值与 TT0CCR1 寄存器的值匹配时，生成 INTTT0CC1 信号。



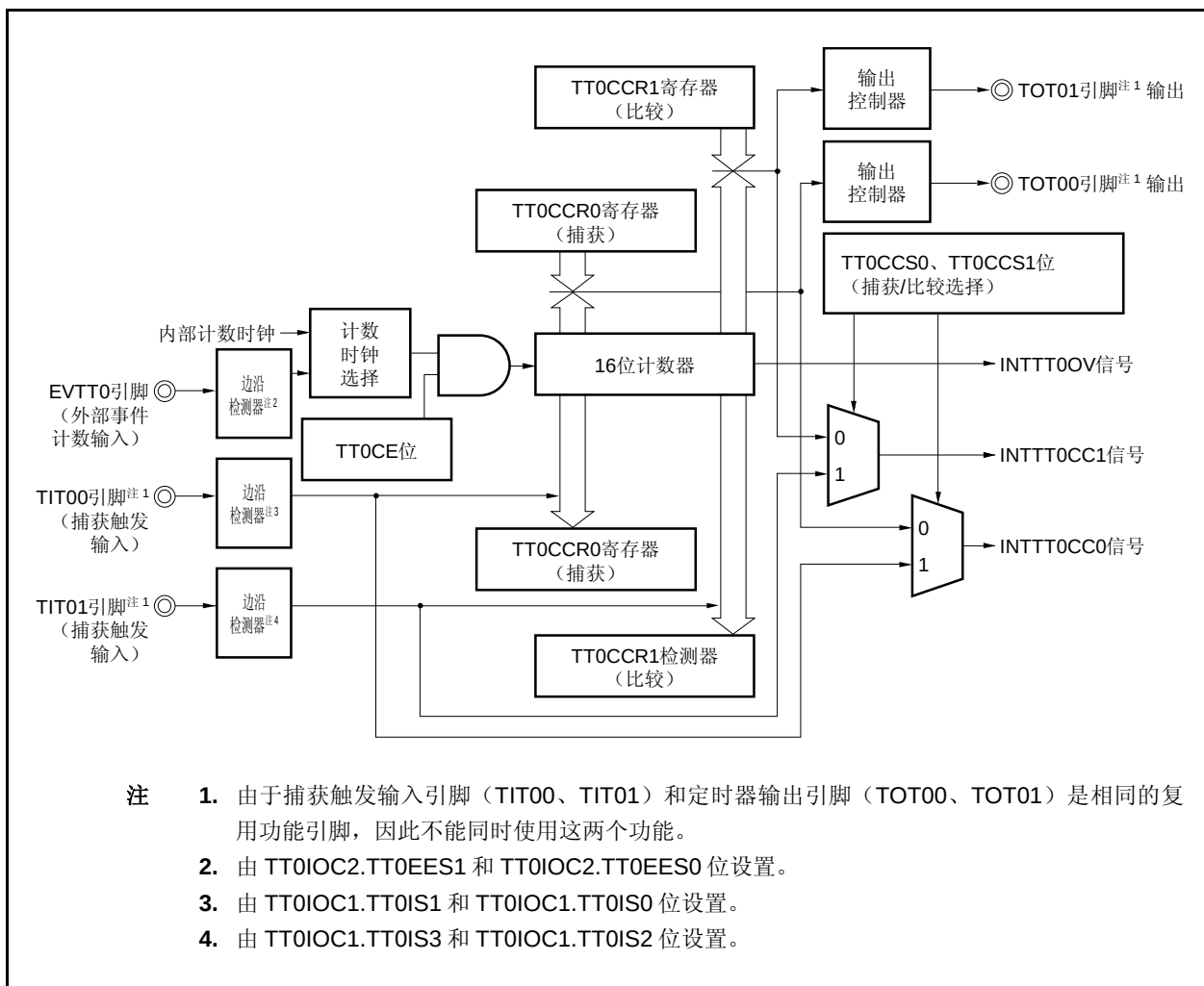
通常情况下，在 16 位计数器的计数值与 TT0CCR1 寄存器的值匹配之后，INTTT0CC1 信号的生成与下一次计数同步发生。

但是，在 PWM 输出模式下，会提早一个计数时钟周期生成。这是由于时序发生了变化，以便与 TOT01 引脚输出信号变化了的时序匹配。

9.6.6 自由运行定时器模式 (TT0MD3 至 TT0MD0 位=0101)

在自由运行定时器模式下，当 TT0CTL0.TT0CE 位置为 1 时，16 位定时器/事件计数器 T 开始计数。此时，根据 TT0OPT0.TT0CCS0 和 TT0OPT0.TT0CCS1 位的设置，TT0CCR0 和 TT0CCR1 寄存器可以用作比较寄存器或捕获寄存器。

图 9-33. 自由运行定时器模式的配置



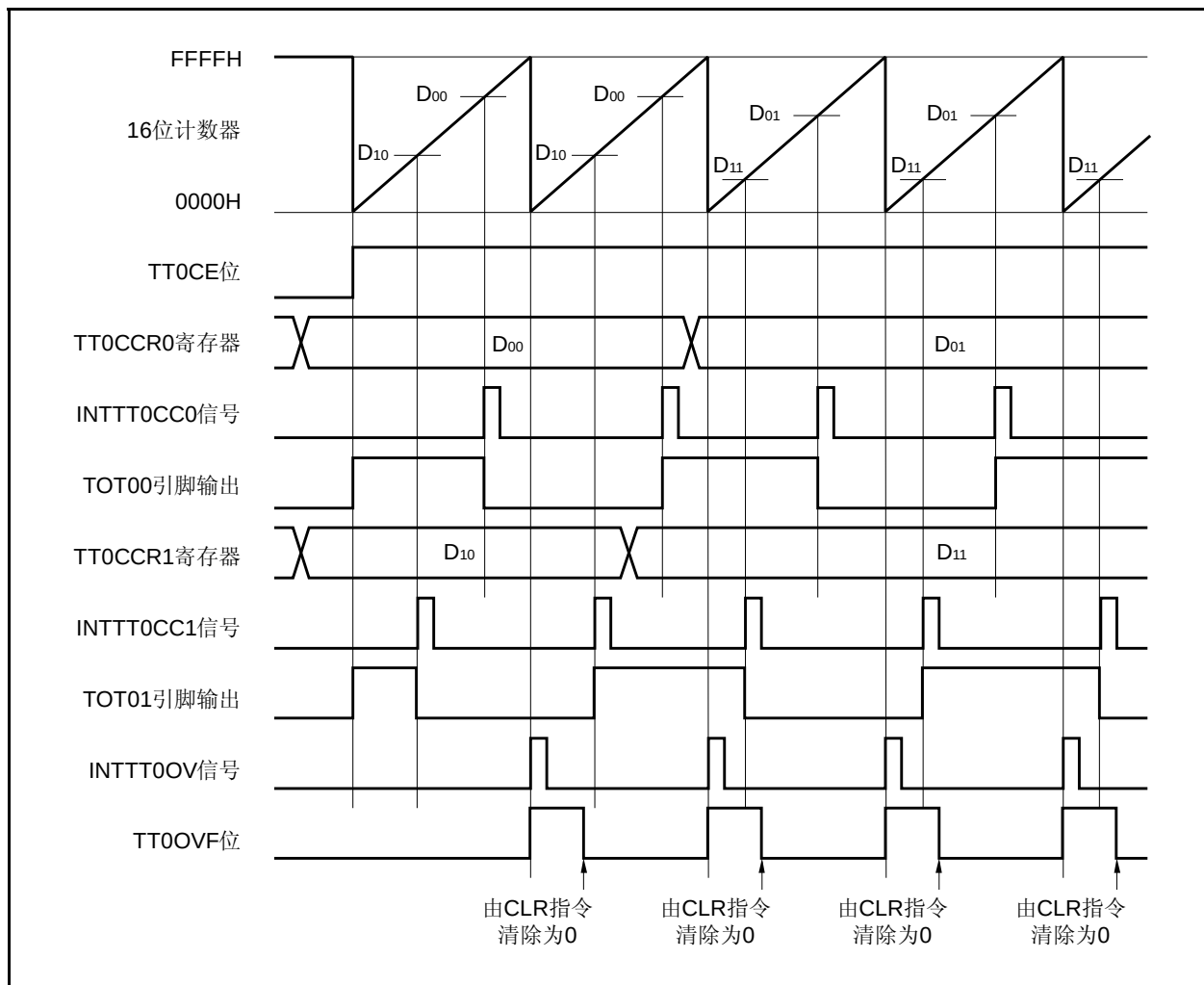
- 比较操作

当 **TT0CE** 位置为 1 时，16 位定时器/事件计数器 T 开始计数，**TOT0n** 引脚的输出信号反相。此后当 16 位计数器的计数值与 **TT0CCRn** 寄存器的设置值匹配时，会生成一个比较匹配中断请求信号 (**INTTT0CCn**)，且 **TOT0n** 引脚的输出信号反相。

16 位计数器继续与计数时钟同步进行计数。当它计数到 **FFFFH** 时，会在下一个计数时钟时产生一个溢出中断请求信号 (**INTTT0OV**)，同时计数器清零为 **0000H**，继续进行计数。此时，上溢出标志 (**TT0OPT0.TT0OVF** 位) 也被置为 1。通过使用软件执行 **CLR** 指令确认上溢出标志被置为 1，然后将其清为 0。

当计数器工作时，可以重写 **TT0CCRn** 寄存器。如果对寄存器进行了重写，则新写入的值将随时生效并与计数值进行比较。

图 9-34. 自由运行定时器模式下的基本时序（比较功能）



- 捕获操作
当 TT0CE 位置为 1 时，16 位计数器开始计数。当检测到输入 TIT0n 引脚的有效边沿时，16 位计数器的计数值储存在 TT0CCRn 寄存器中，并生成一个捕获中断请求信号 (INTTT0CCn)。
16 位计数器继续与计数时钟同步进行计数。当它计数到 FFFFH 时，会在下一个计数时钟时产生一个上溢出中断请求信号 (INTTT0OV)，同时计数器清零为 0000H，继续进行计数。此时，上溢出标志 (TT0OPT0.TT0OVF 位) 也被置为 1。通过使用软件执行 CLR 指令确认上溢出标志被设置为 1，然后将其清为 0。

图 9-35. 自由运行定时器模式下的基本时序（捕获功能）

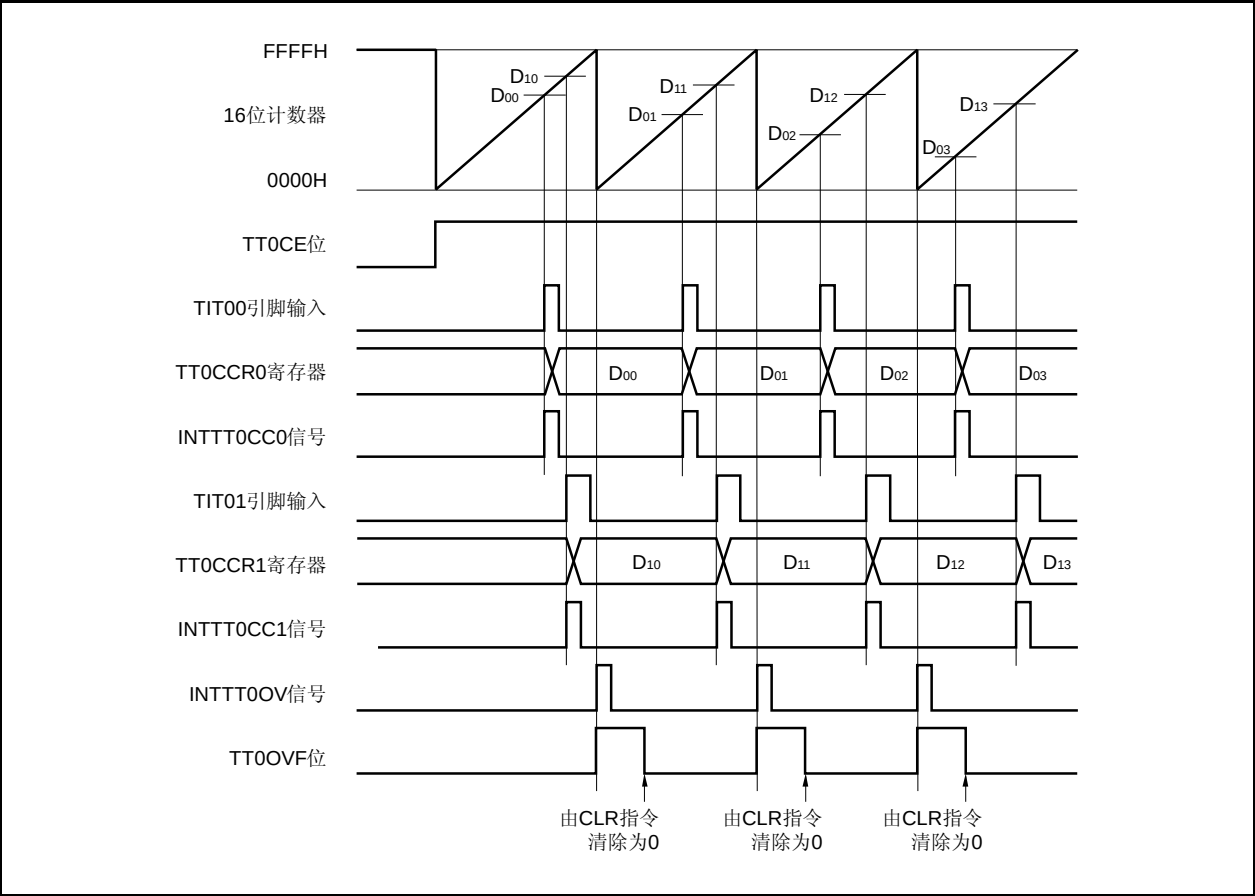


图 9-36. 自由运行定时器模式下的寄存器设置 (1/2)

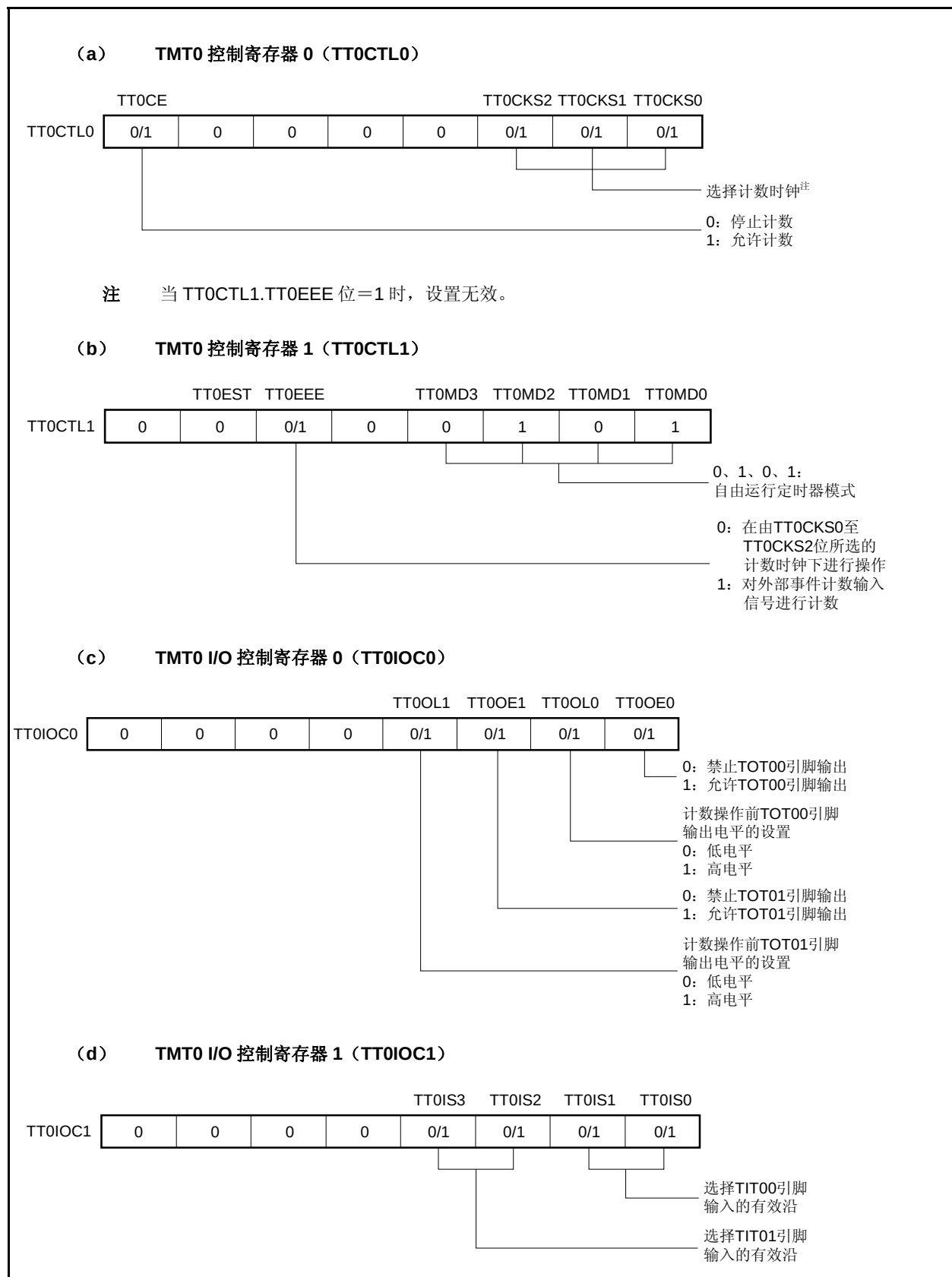
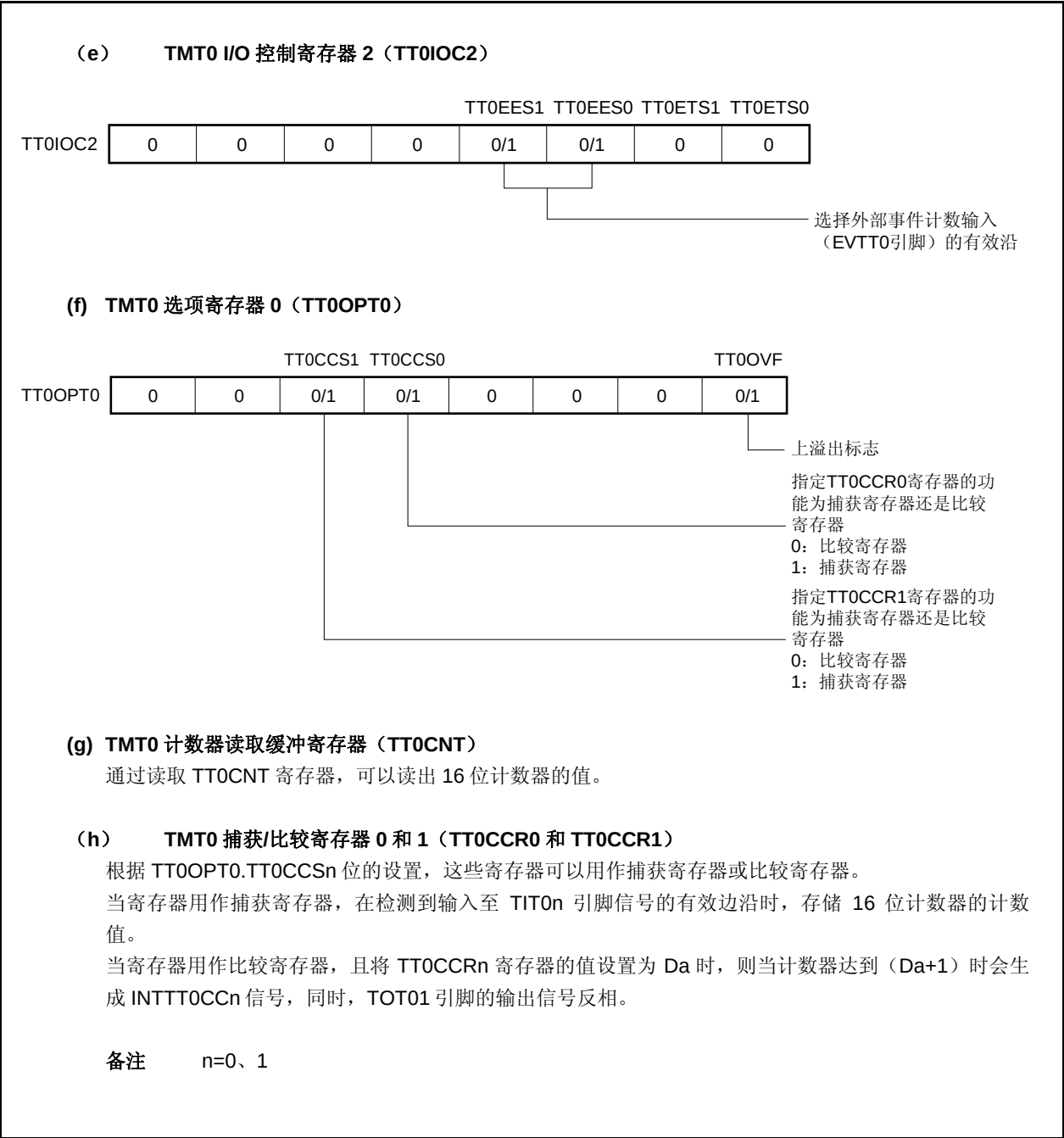


图 9-36. 自由运行定时器模式下的寄存器设置 (2/2)



(1) 自由运行定时器模式下的操作流程

(a) 捕获/比较寄存器用作比较寄存器时

图 9-37. 自由运行定时器模式时的软件处理流程（比较功能）（1/2）

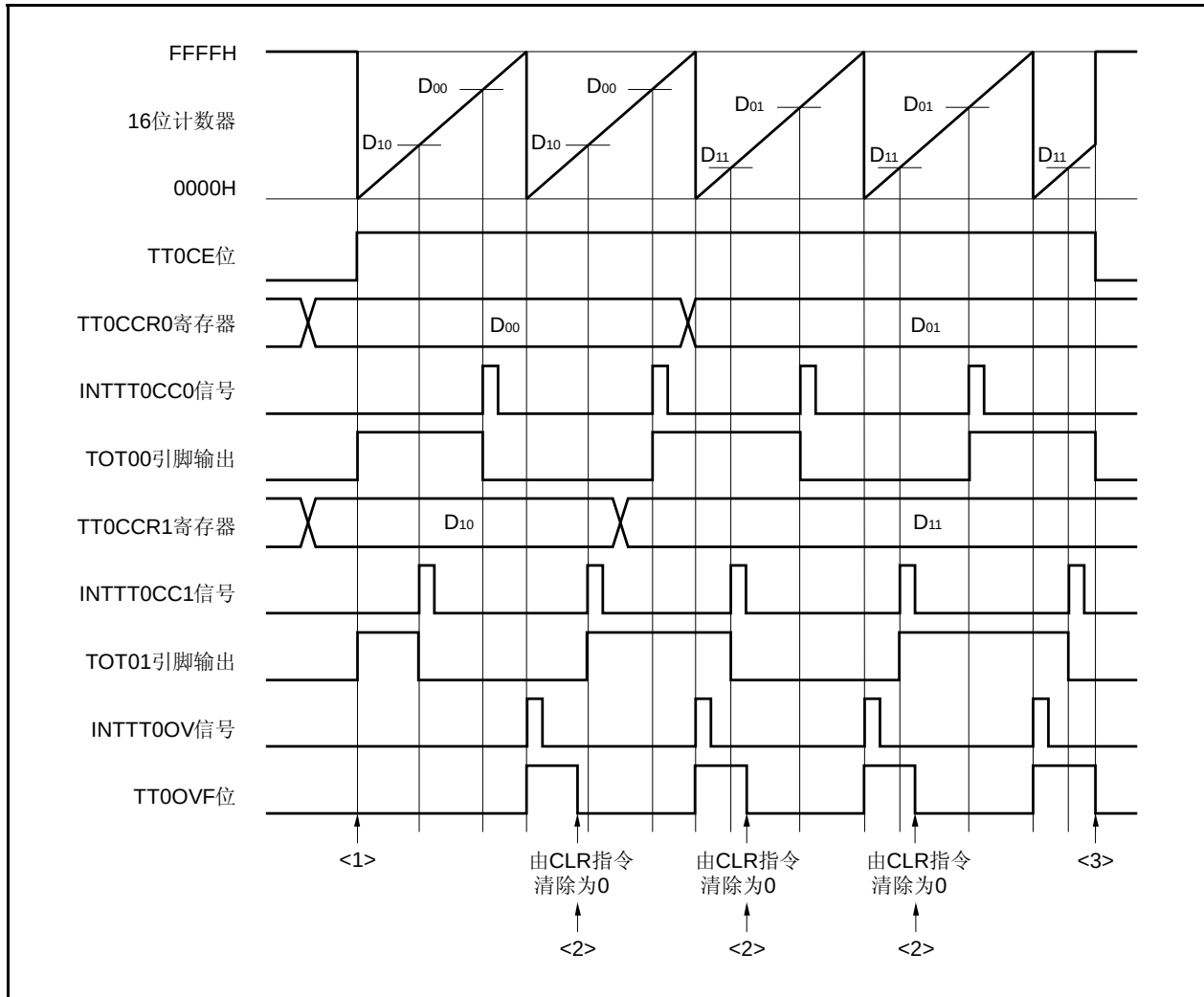
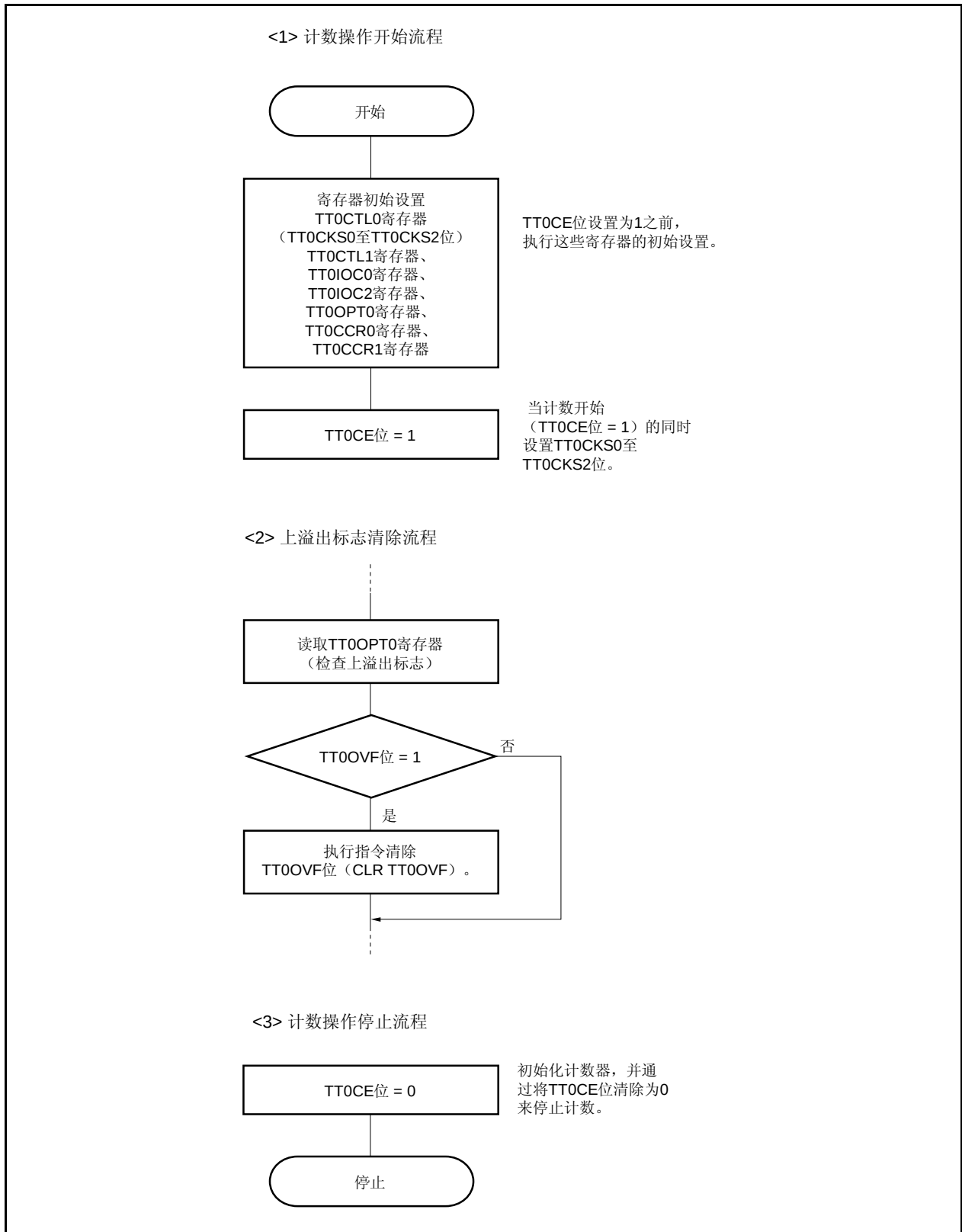


图 9-37. 自由运行定时器模式时的软件处理流程（比较功能）（2/2）



(b) 捕获/比较寄存器用作捕获寄存器时

图 9-38. 自由运行定时器模式时的软件处理流程（捕获功能）（1/2）

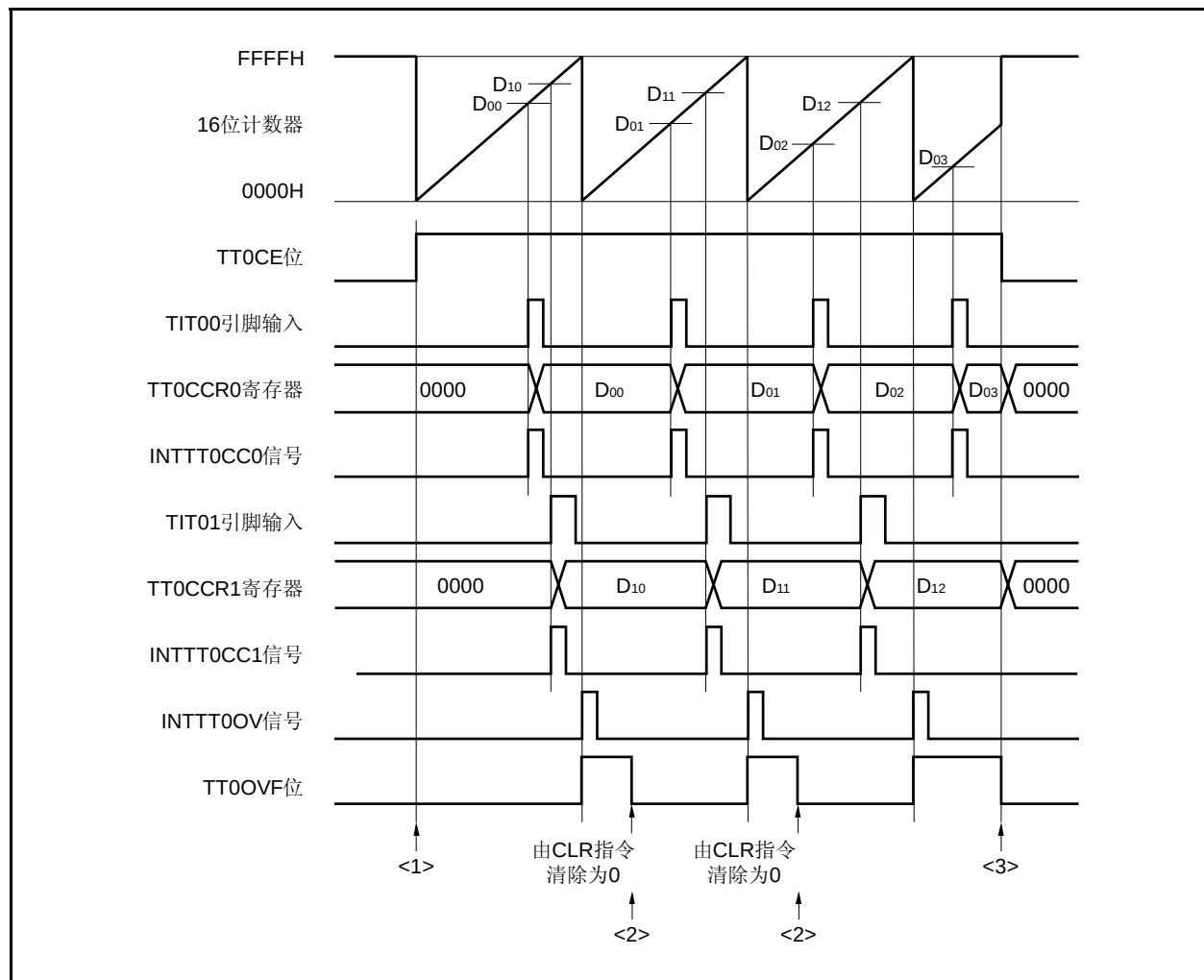
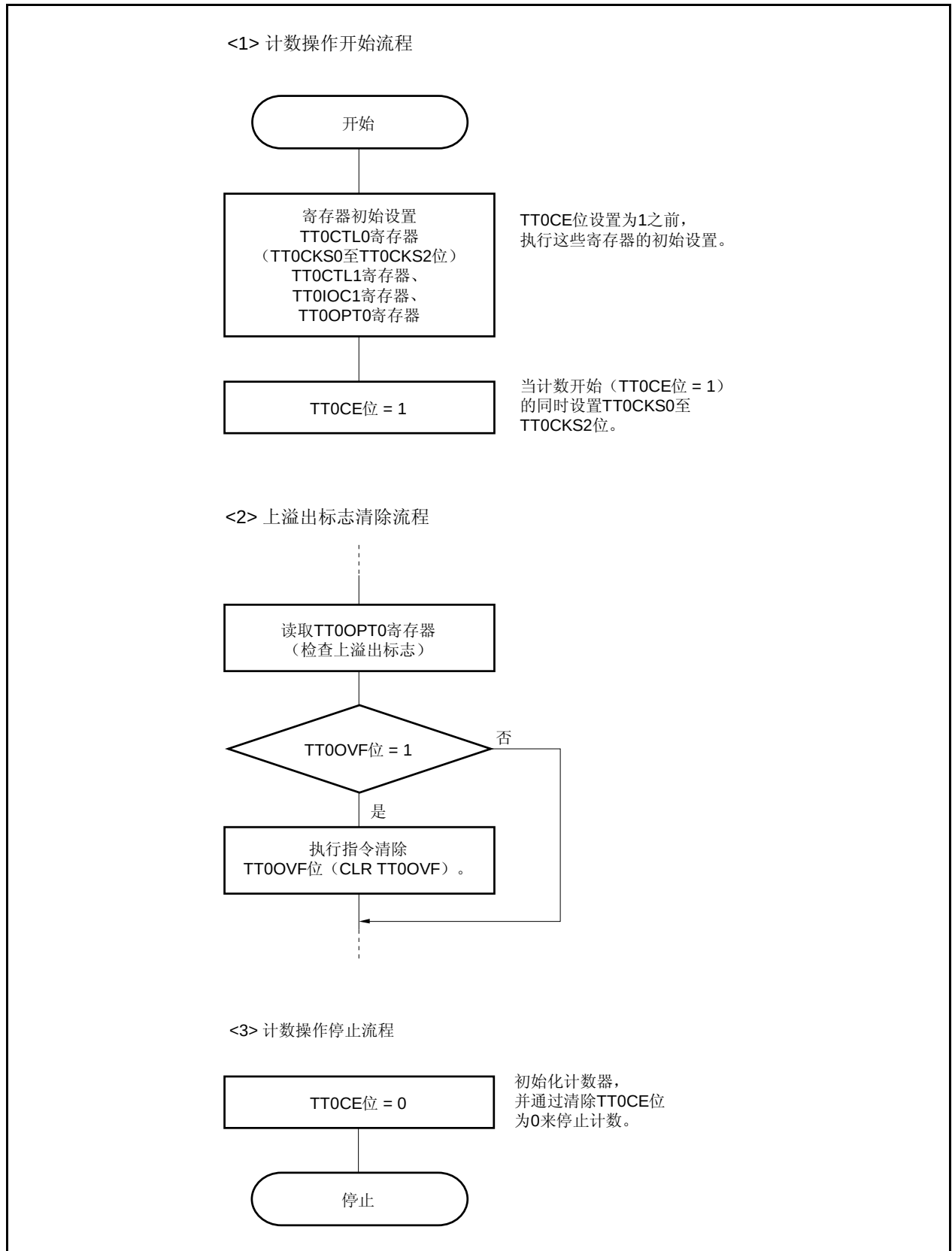


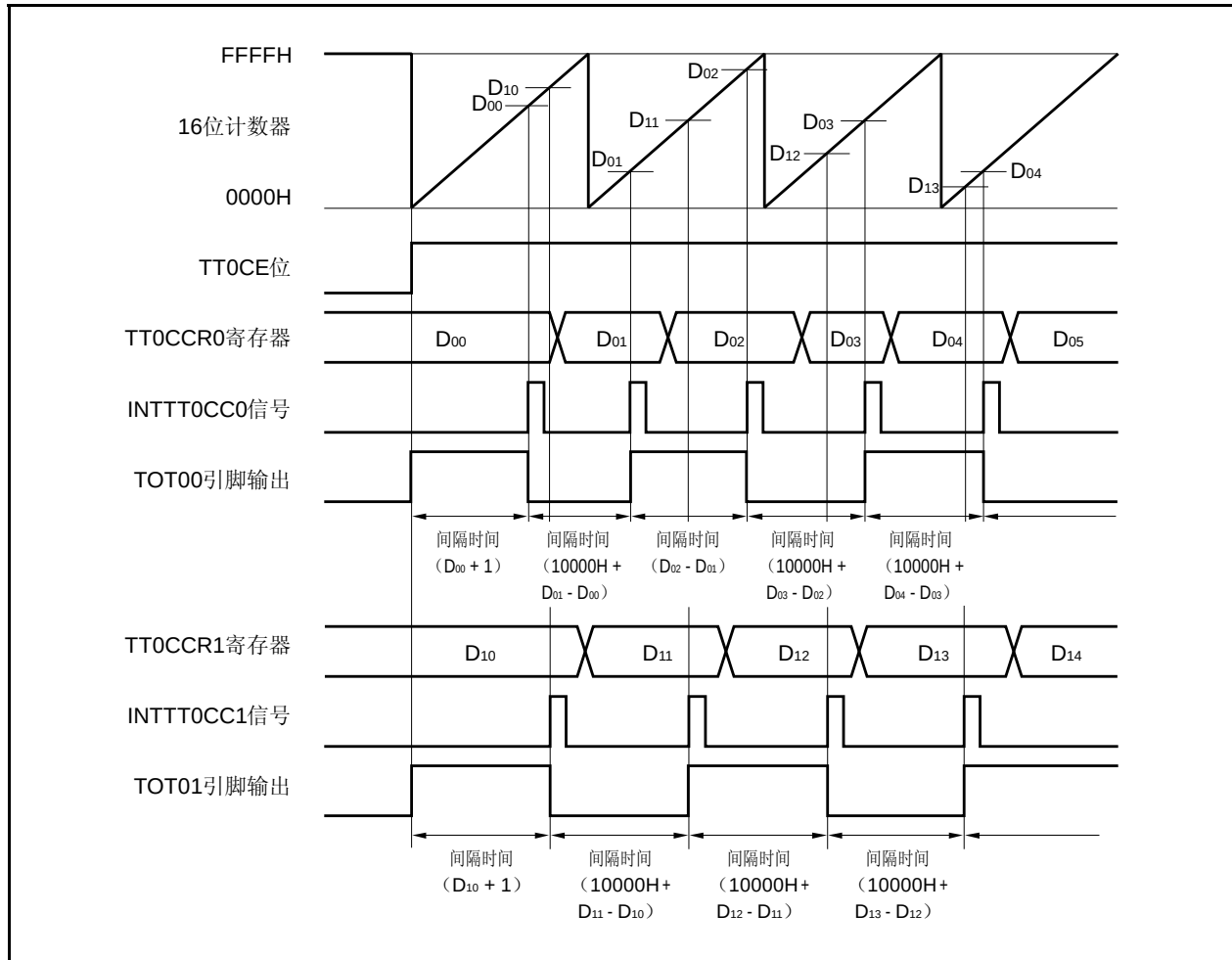
图 9-38. 自由运行定时器模式时的软件处理流程（捕获功能）（2/2）



(2) 自由运行定时器模式时的工作时序

(a) 比较寄存器的间隔定时操作

在 TT0CCRn 寄存器用作比较寄存器的情况下，当 16 位定时器/事件计数器 T 用作间隔定时器时，在每次检测到 INTTT0CCn 信号后，都需要利用软件处理来设置一个比较值以生成下一个中断请求信号。



在自由运行定时器模式下进行时间间隔操作时，一个通道可以设置两个时间间隔。

要进行时间间隔操作，当检测到 INTTT0CCn 信号后，必须在执行的中断服务程序中重置相应的 TT0CCRn 寄存器。

重置 TT0CCRn 寄存器的设置值可由下式计算，其中“Da”是间隔时间。

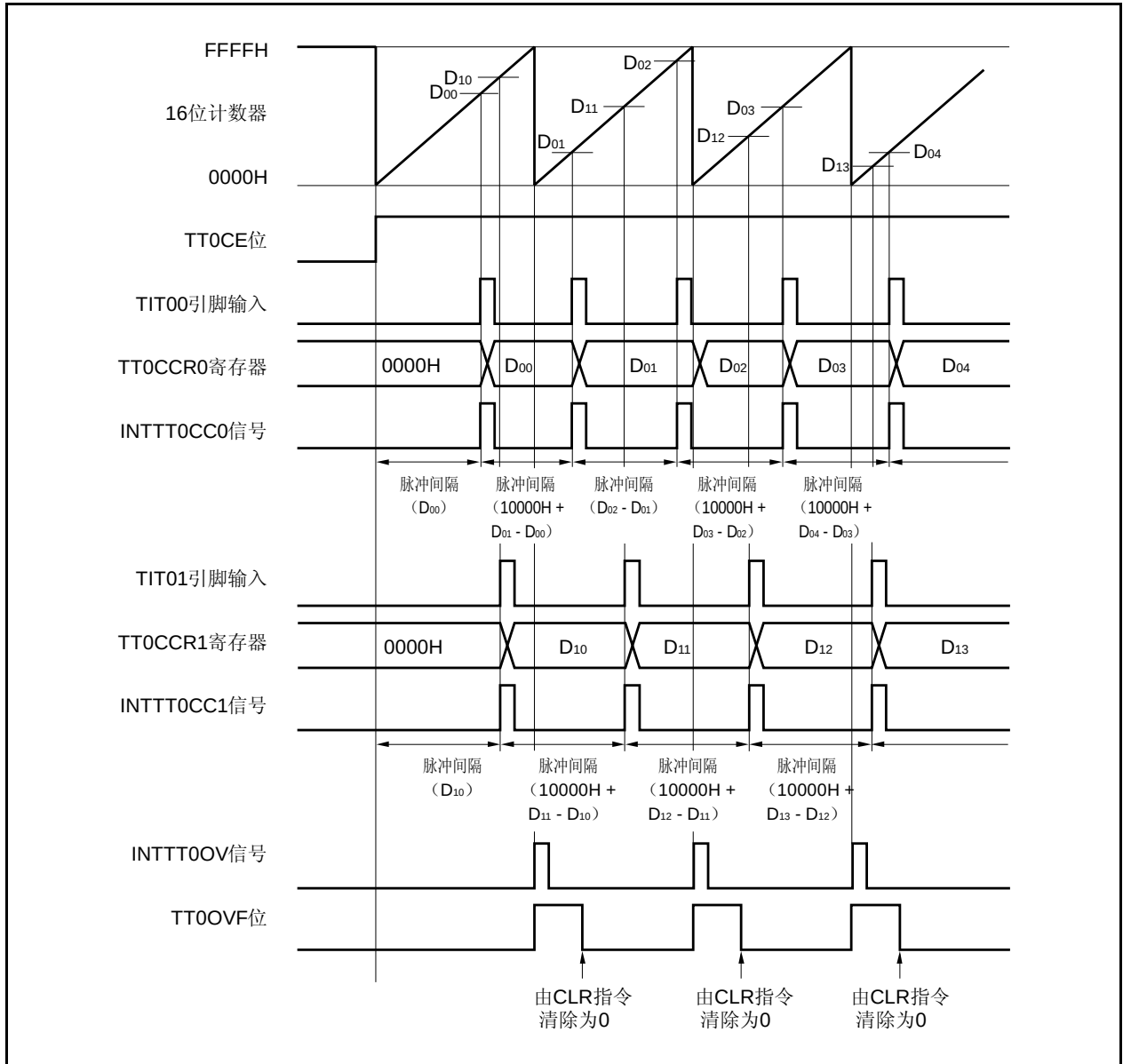
比较寄存器默认值: $D_a - 1$

比较寄存器的第二次及其以后的设置值: 上次设置值 + D_a

(若计算结果大于 FFFFH, 则从结果中减去 10000H 并将此值设置给寄存器。)

(b) 用捕获寄存器进行脉冲宽度测量

在用 TT0CCRn 寄存器作为捕获寄存器进行脉冲宽度测量时，每次检测到 INTTT0CCn 信号，都需要用软件处理来读取捕获寄存器和计算时间间隔。



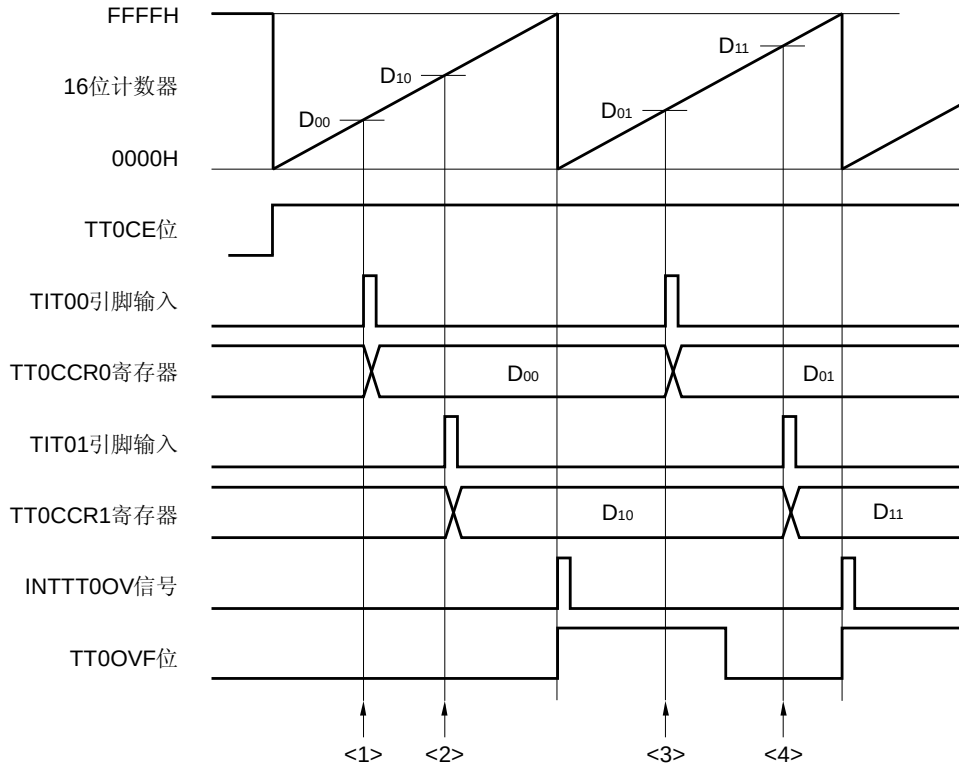
在自由运行定时器模式下执行脉宽测量时，可以用一个通道测量两个脉冲宽度。

要测量脉冲宽度的话，可以通过读取与 INTTT0CCn 信号同步的 TT0CCRn 寄存器的值，并计算本次读取值与先前读取值之间的差，来计算脉冲宽度。

(c) 当使用两个捕获寄存器时对上溢出的处理

当使用两个捕获寄存器时，在处理上溢出标志时必须格外小心。首先，一个错误的处理示例如下所示。

使用两个捕获寄存器时错误处理示例



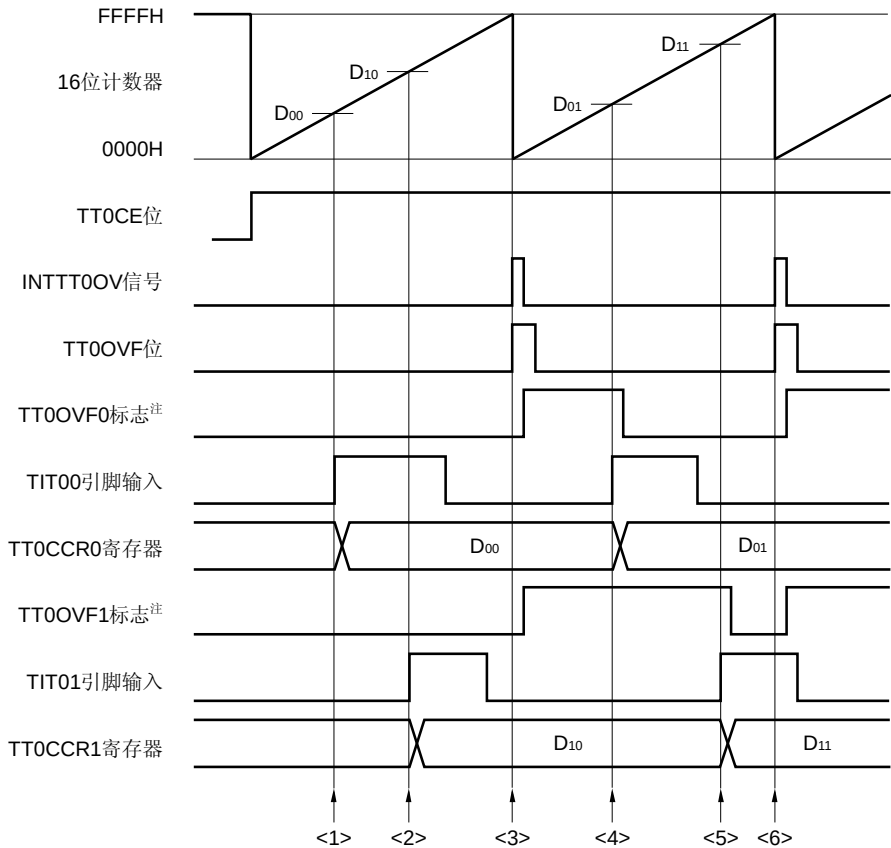
在自由运行定时器模式下测量两个脉冲宽度时，可能出现下列问题。

- <1> 读取 TT0CCR0 寄存器（设置 TIT00 引脚输入的默认值）。
- <2> 读取 TT0CCR1 寄存器（设置 TIT01 引脚输入的默认值）。
- <3> 读取 TT0CCR0 寄存器。
读上溢出标志。若上溢出标志为 1，则将其清为 0。
因为上溢出标志为 1，所以脉冲宽度可以用 $(10000H + D_{01} - D_{00})$ 计算。
- <4> 读取 TT0CCR1 寄存器。
读上溢出标志。因为此标志在<3>中清零了，所以，读取到 0。
因为上溢出标志为 0，所以脉冲宽度可以用 $(D_{11} - D_{10})$ 计算。（错误）

在使用两个捕获寄存器时，如果上溢出标志由一个捕获寄存器清为 0，则另一个捕获寄存器可能不会得到正确的脉冲宽度。

在使用两个捕获寄存器时，使用软件解决。关于如何使用软件，示例如下。

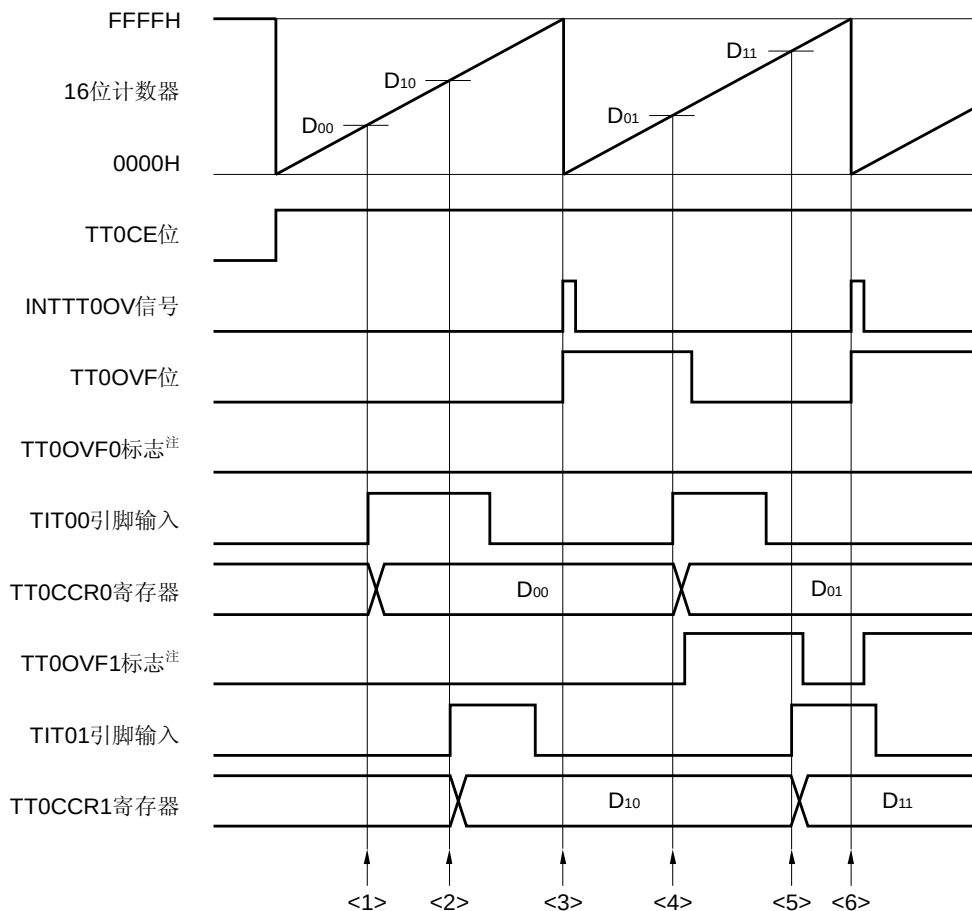
使用两个捕获寄存器时的示例（使用上溢出中断）



注 由软件对内部 RAM 中的 TT0OVF0 和 TT0OVF1 标志进行设置。

- <1> 读取 TT0CCR0 寄存器（设置 TIT00 引脚输入的默认值）。
- <2> 读取 TT0CCR1 寄存器（设置 TIT01 引脚输入的默认值）。
- <3> 发生上溢出。在上溢出中断服务程序中将 TT0OVF0 和 TT0OVF1 标志设置为 1，并将上溢出标志清除为 0。
- <4> 读取 TT0CCR0 寄存器。
读取 TT0OVF0 标志。如果 TT0OVF0 标志置为 1，则将其清除为 0。
因为 TT0OVF0 标志为 1，所以脉冲宽度可以用 $(10000H + D_{01} - D_{00})$ 计算。
- <5> 读取 TT0CCR1 寄存器。
读取 TT0OVF1 标志。如果 TT0OVF1 标志为 1，则将其清为 0（TT0OVF0 标志在<4>中已清零，而 TT0OVF1 标志仍然保持为 1）。
因为 TT0OVF1 标志为 1，所以脉冲宽度可以用 $(10000H + D_{11} - D_{10})$ 计算。（正确）
- <6> 重复<3>

使用两个捕获寄存器时的示例（不使用上溢出中断）



注 由软件对内部 RAM 中的 TT0OVF0 和 TT0OVF1 标志进行设置。

<1> 读取 TT0CCR0 寄存器（设置 TIT00 引脚输入的默认值）。

<2> 读取 TT0CCR1 寄存器（设置 TIT01 引脚输入的默认值）。

<3> 发生上溢出。软件不进行处理。

<4> 读取 TT0CCR0 寄存器。

读上溢出标志。若上溢出标志为 1，则仅将 TT0OVF1 标志置为 1，并把上溢出标志清为 0。

因为上溢出标志为 1，所以脉冲宽度可以用 $(10000H + D_{01} - D_{00})$ 计算。

<5> 读取 TT0CCR1 寄存器。

读上溢出标志。因为上溢出标志在<4>中清零了，所以读到 0。

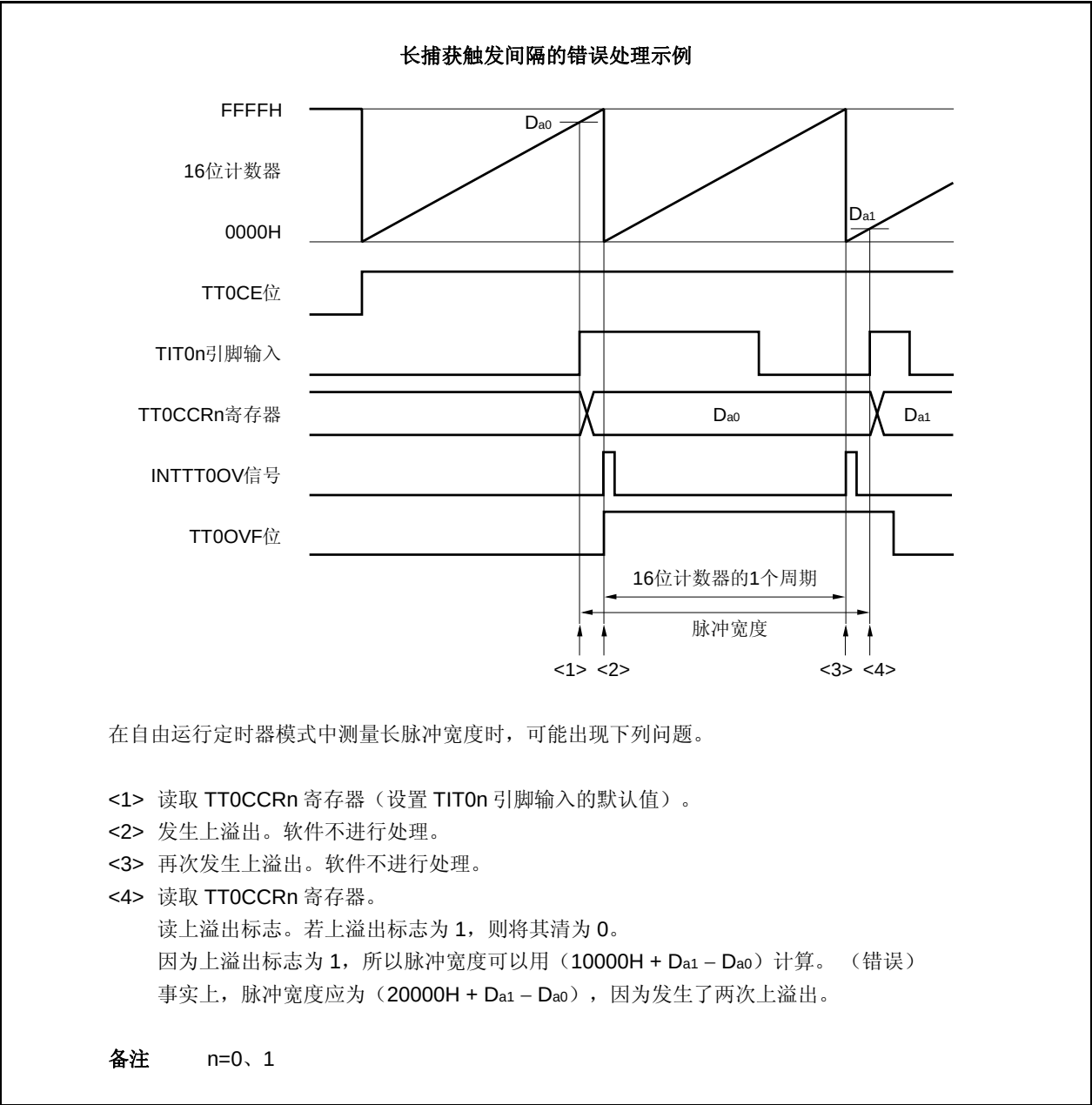
读取 TT0OVF1 标志。如果 TT0OVF1 标志为 1，则将其清为 0。

因为 TT0OVF1 标志为 1，所以脉冲宽度可以用 $(10000H + D_{11} - D_{10})$ 计算。（正确）

<6> 重复 <3>

(d) 长捕获触发间隔时对上溢出的处理

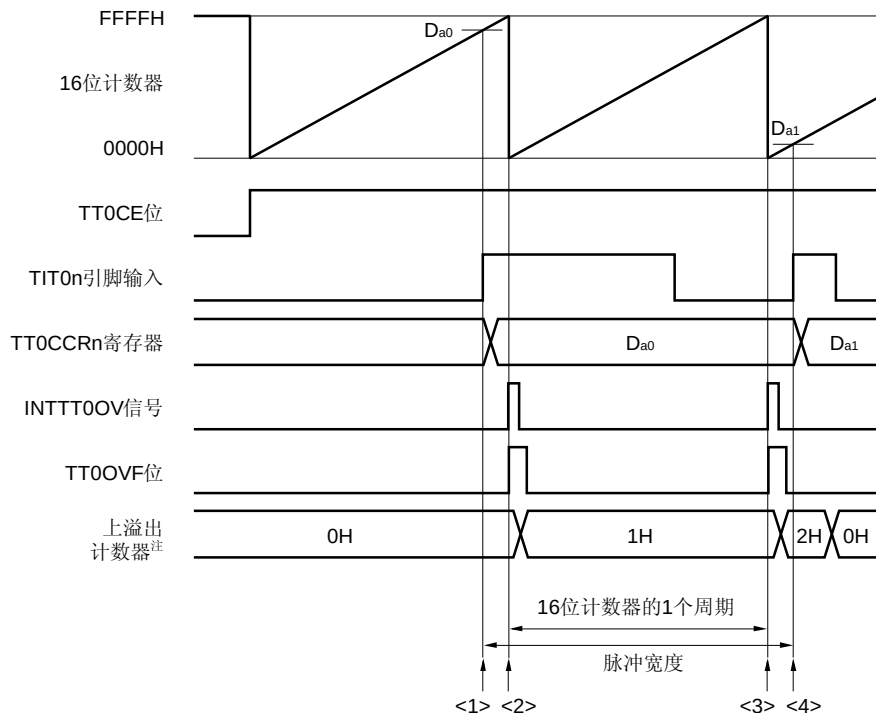
如果脉冲宽度大于 16 位计数器的一个周期，则必须加以注意，因为从第一个捕获触发到下一个之间可能发生不止一次上溢出。首先，如下所示，是一个错误处理的例子。



如果在长捕获触发间隔期间，发生了两次或更多次的上溢出，则无法获得正确的脉冲宽度。

如果捕获触发间隔为长间隔，应减慢计数时钟以延长 16 位计数器的周期，或者使用软件加以解决。关于如何使用软件，示例如下。

长捕获触发间隔时的示例



注 内部 RAM 中的上溢出计数器由软件任意设置。

<1> 读取 TT0CCRn 寄存器（设置 TIT0n 引脚输入的默认值）。

<2> 发生上溢出，增加上溢出计数器的值，并在溢出中断服务程序中把溢出标志清为 0。

<3> 再次发生上溢出。增加上溢出计数器的值，并在溢出中断服务程序中把溢出标志清为 0。

<4> 读取 TT0CCRn 寄存器。

读取上溢出计数器。

→ 当溢出计数器为捏时，脉冲宽度可用 $(N \times 10000H + Da1 - Da0)$ 计算。

在本例中，脉冲宽度为 $(20000H + Da1 - Da0)$ ，因为发生了两次上溢出。

最后，清零上溢出计数器 (0H)。

备注 n=0、1

(e) 清除上溢出标志

在 TT0OVF 位为 1 时读取该位后，使用 CLR 指令将 TT0OVF 位清零可以将上溢出标志清零，而在 TT0OVF 位为 1 时读取该位后，通过将 8 位数据（位 0 为 0）写入 TT0OPT0 寄存器也可以将溢出标志清零。

9.6.7 脉冲宽度测量模式 (TT0MD3 至 TT0MD0 位=0110)

在脉冲宽度测量模式下，当 TT0CTL0.TT0CE 位置为 1 时，16 位计时器/事件计数器 T 开始计数。每次检测到 TIT0n 引脚输入信号的有效边沿时，16 位计数器的计数值就存储在 TT0CCRn 寄存器中，且 16 位计数器清零为 0000H。

在出现捕获中断请求信号 (INTTT0CCn) 之后，可以通过读取 TT0CCRn 寄存器，测量有效边缘的间隔。

如图 9-39 所示，通过使用 TT0IOC1 寄存器，选择 TIT00 或 TIT01 引脚作为捕获触发输入引脚，并将未使用的引脚设置为“无边沿检测”。

图 9-39. 脉冲宽度测量模式的配置

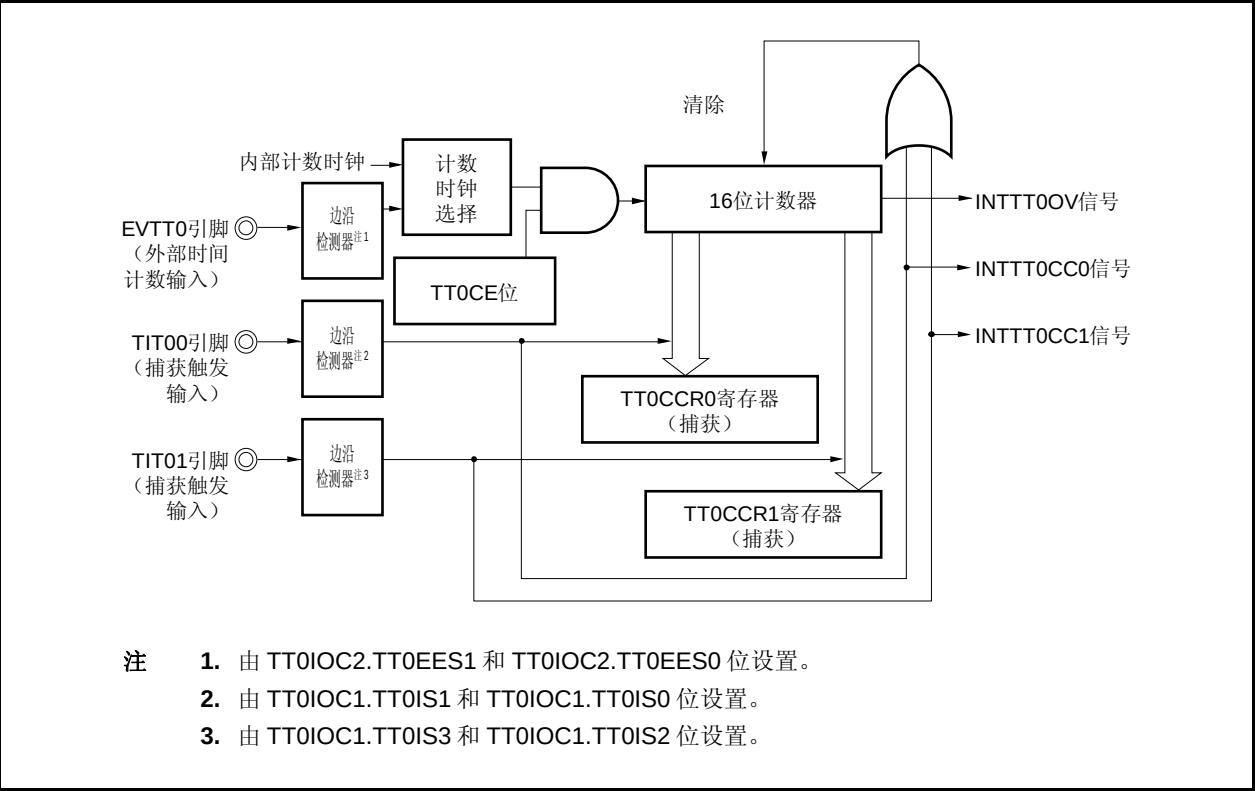
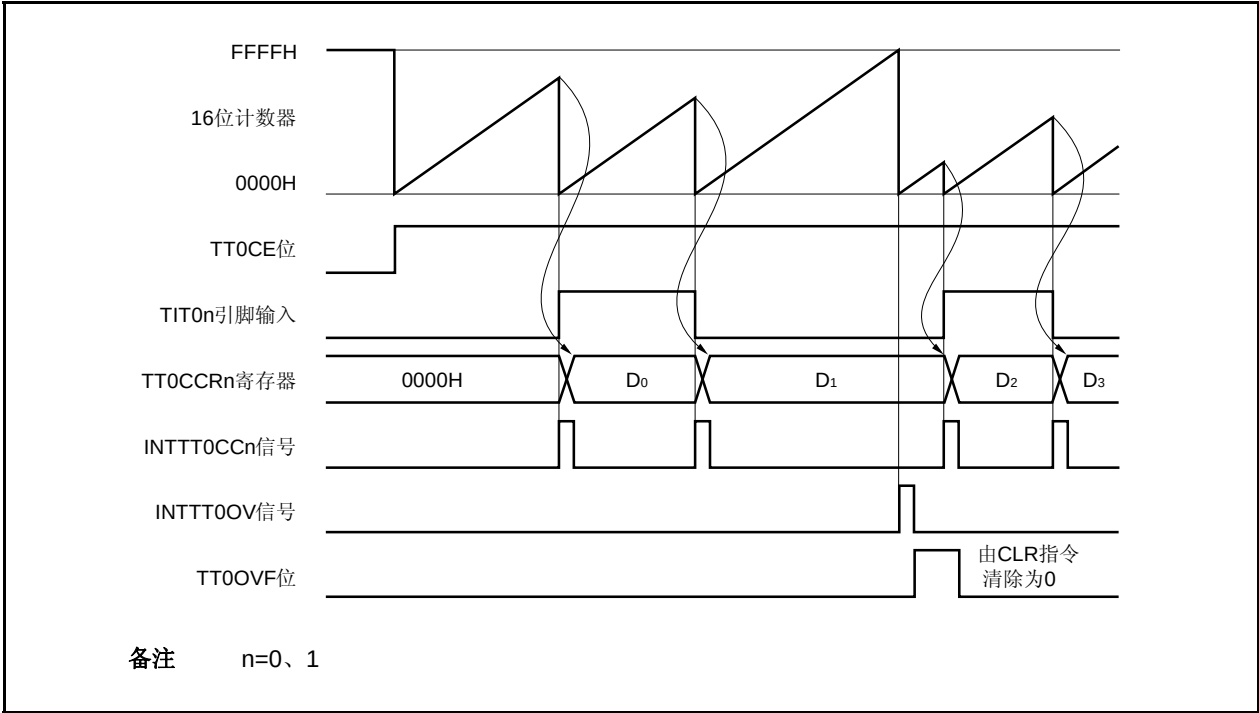


图 9-40. 脉冲宽度测量模式的基本时序



当 TT0CE 位置为 1 时，16 位计数器开始计数。此后检测到 TIT0n 引脚输入信号的有效边沿时，16 位计数器的计数值存储在 TT0CCRn 寄存器中，16 位计数器清零为 0000H，并生成一个捕获中断请求信号（INTTT0CCn）。

脉冲宽度计算如下。

$$\text{脉冲宽度} = \text{捕获值} \times \text{计数时钟周期}$$

即使 16 位计数器一直计数到 FFFFH，也没有有效边沿输入到 TIT0n 引脚，仍然会在下一个计数时钟生成上溢出中断请求信号（INTTT0OV），且计数器清零为 0000H，并继续计数。此时，上溢出标志（TT0OPT0.TT0OVF 位）也置为 1。使用软件执行 CLR 指令将上溢出标志清为 0。

若上溢出标志设置为 1，则脉冲宽度计算如下。

$$\text{脉冲宽度} = (10000H \times \text{TT0OVF 位置为 (1) 的次数} + \text{捕获值}) \times \text{计数时钟周期}$$

备注 n=0、1

图 9-41. 脉冲宽度测量模式时的寄存器设置 (1/2)

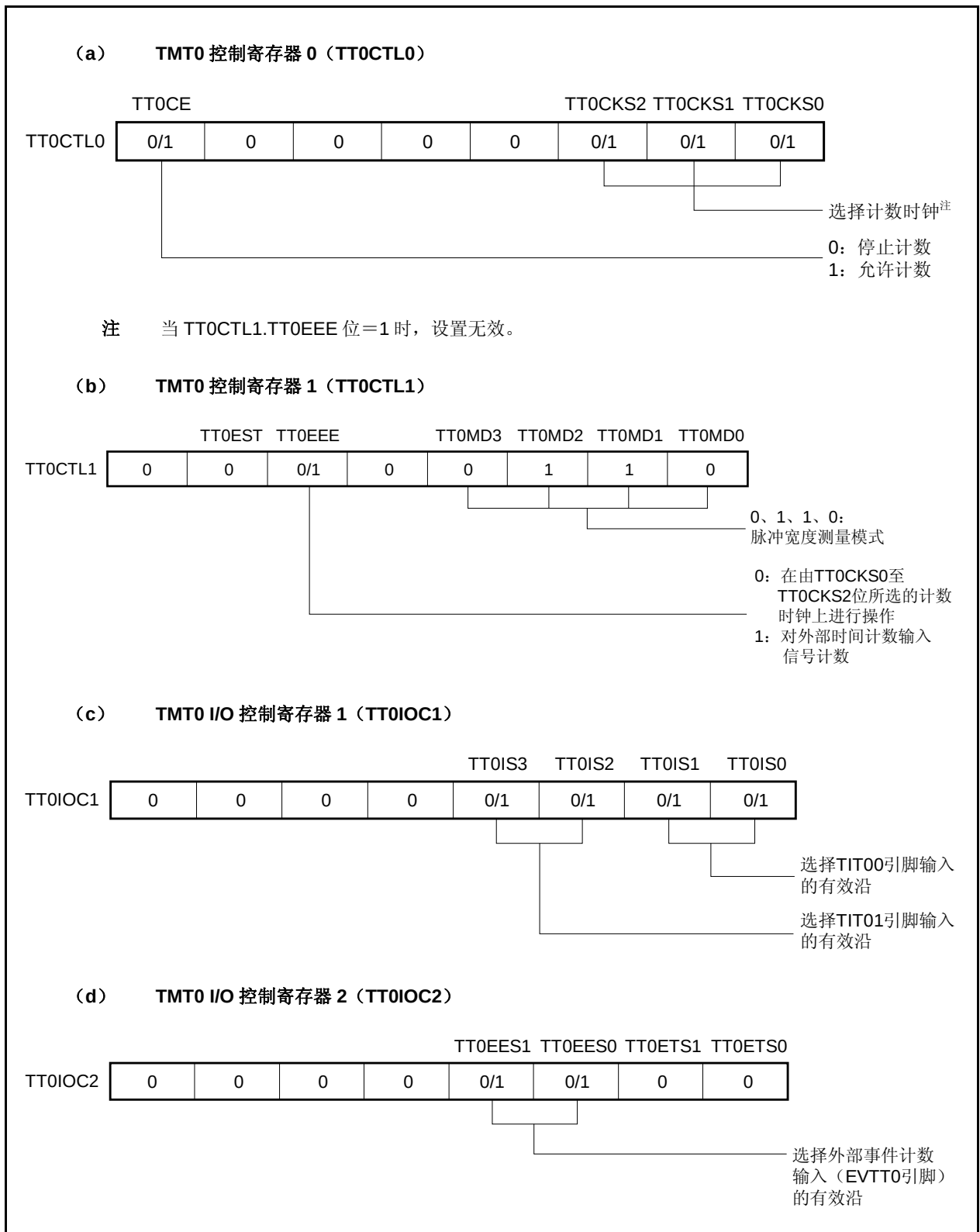
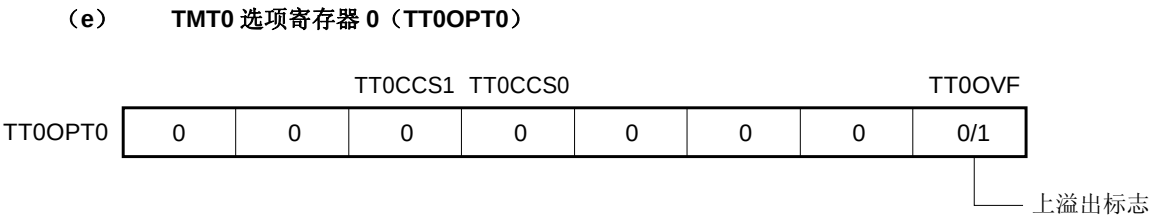


图 9-41. 脉冲宽度测量模式时的寄存器设置（2/2）



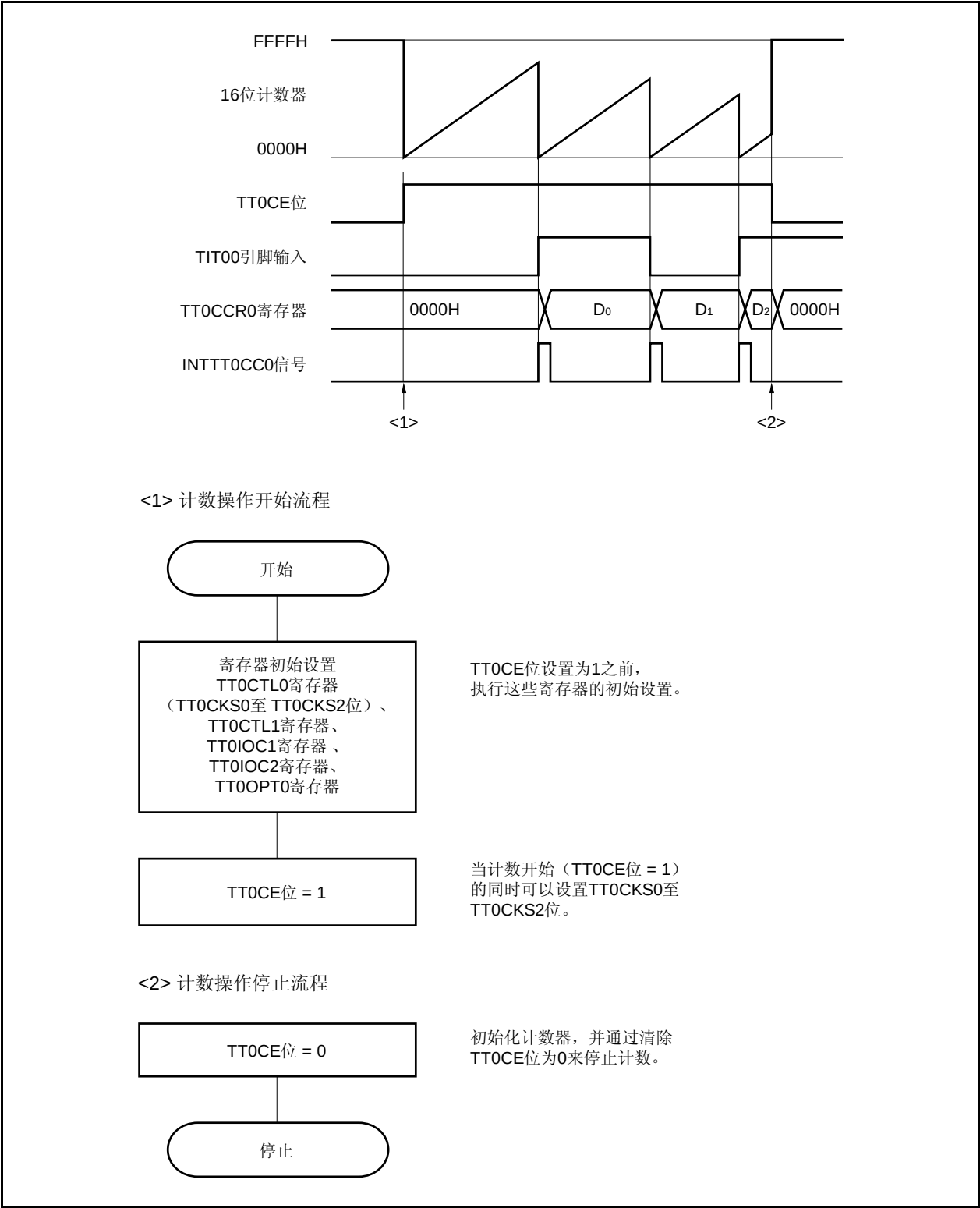
(f) TMT0 计数器读取缓冲寄存器 (TT0CNT)
通过读取 TT0CNT 寄存器，可以读出 16 位计数器的值。

(g) TMT0 捕获/比较寄存器 0 和 1 (TT0CCR0 和 TT0CCR1)
当检测到输入至 TIT00 和 TIT01 引脚的有效边沿时，这些寄存器保存 16 位计数器的计数值。

备注 在脉冲宽度测量模式下，不使用 TMT0 控制寄存器 2 (TT0CTL2)、TMT0 I/O 控制寄存器 0 (TT0IOC0)、TMT0 I/O 控制寄存器 3 (TT0IOC3)、TMT0 选项寄存器 1 (TT0OPT1) 和 TMT0 计数器写寄存器 (TT0TCW)。

(1) 脉冲宽度测量模式时的工作流程

图 9-42. 脉冲宽度测量模式时的软件处理流程



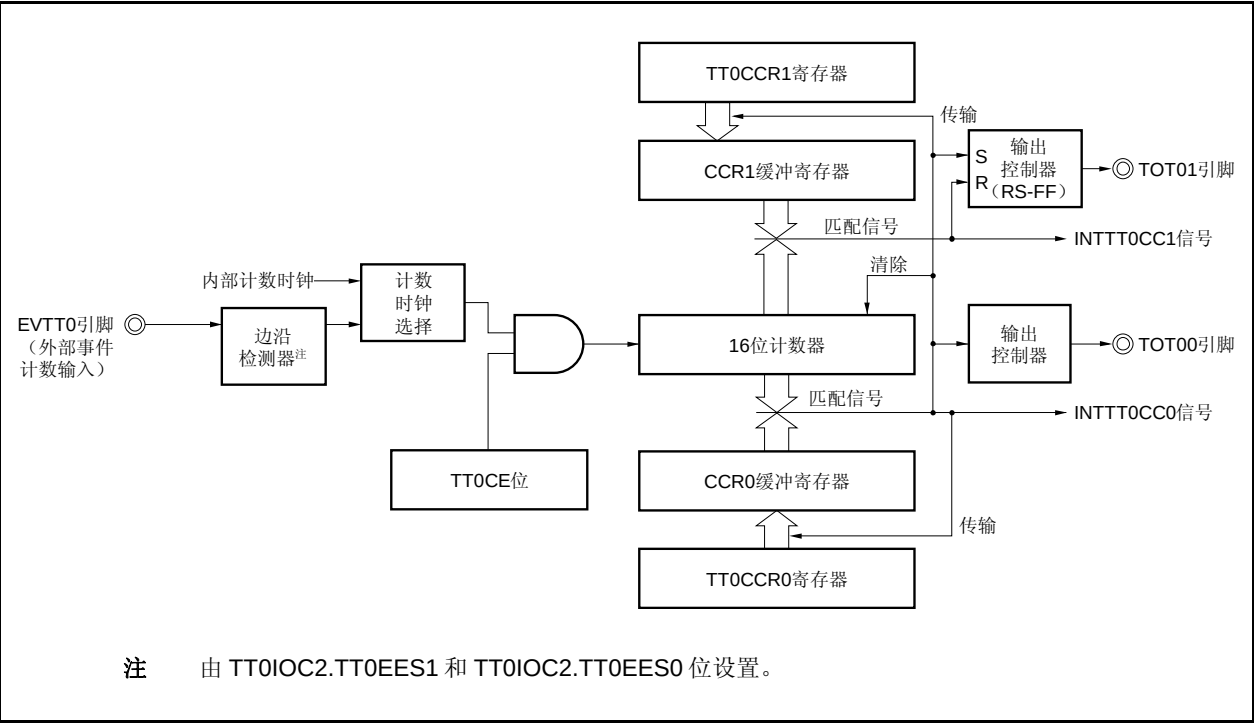
(2) 脉冲宽度测量模式时的工作时序**(a) 清除上溢出标志**

在 TT0OVF 位置为 1 时读取该位后，使用 CLR 指令将 TT0OVF 位清零可以将上溢出标志清零，而在 TT0OVF 位置为 1 时读取该位后，通过将 8 位数据（位 0 为 0）写入 TT0OPT0 寄存器也可以将上溢出标志清零。

9.6.8 三角波PWM输出模式 (TT0MD3 至TT0MD0 位=0111)

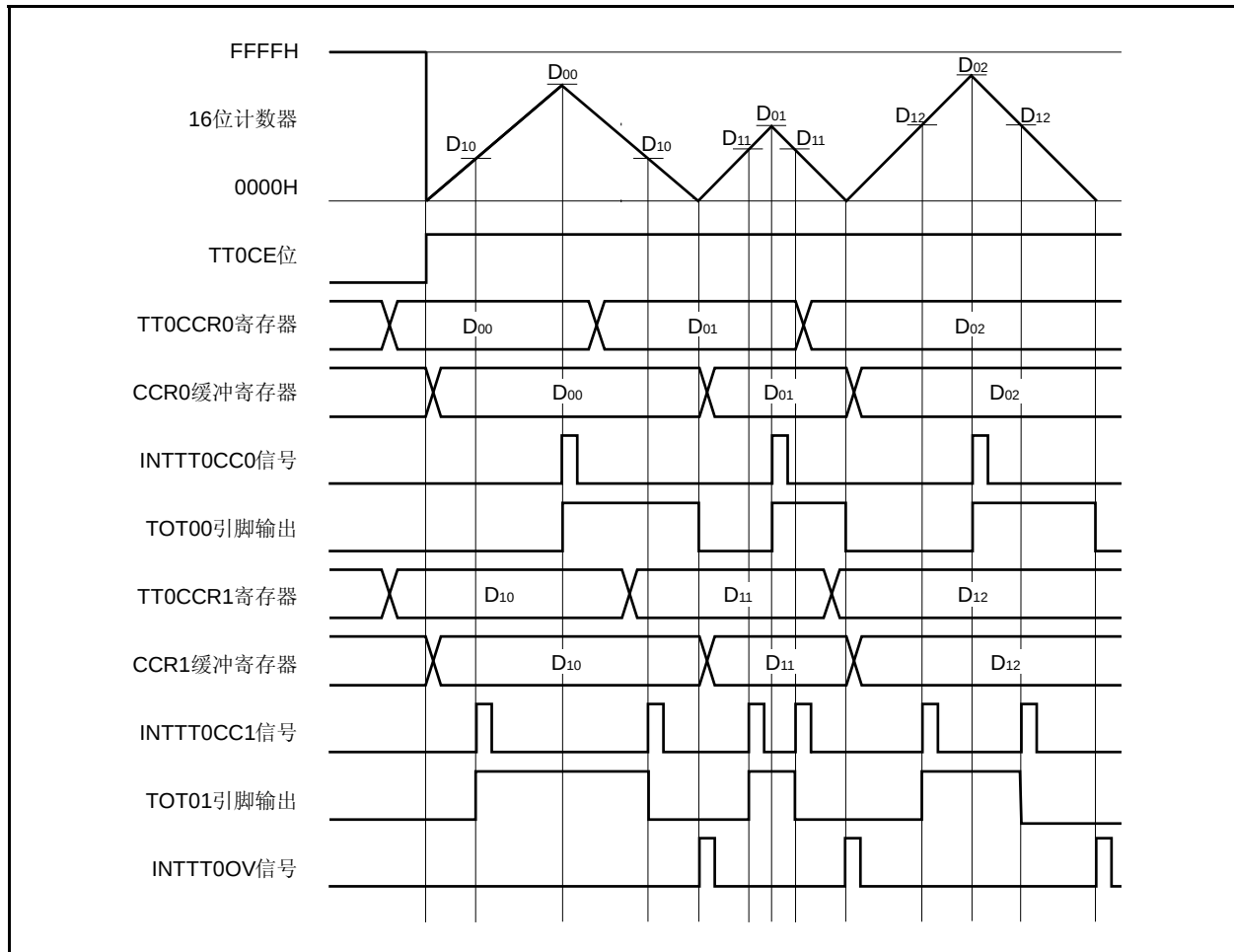
在三角波 PWM 输出模式下，当 TT0CTL0.TT0CE 位置为 1 时，从 TOT01 引脚输出一个三角波 PWM 波形。
当 16 位计数器的计数值与 CCR0 缓冲寄存器的值匹配且 16 位计数器置为 0000H 时，从 TOT00 引脚输出反相的 PWM 波形。

图 9-43. 三角波 PWM 输出模式的配置



注 由 TT0IOC2.TT0EES1 和 TT0IOC2.TT0EES0 位设置。

图 9-44. 三角波 PWM 输出模式的基本时序



当 TT0CE 位置为 1 时，16 位计数器从 FFFFH 清除为 0000H，并开始计数。从 TOT01 引脚输出三角形的 PWM 波形。

在三角波 PWM 输出模式中，计数器向上或向下计数。当 16 位计数器向下计数至 0000H 时，生成上溢出中断请求信号 (INTTT0OV)。此时，不将 TT0OPT0.TT0OVF 位置为 1。当 16 位计数器向上计数时，若其计数值与 CCR0 缓冲寄存器的值匹配，就会生成一个比较匹配中断请求信号 (INTTT0CC0)。

当 16 位计数器的值与 CCR0 缓冲寄存器的值匹配时，计数方向从向上变为向下，而当计数器清除为 0000H 时，计数方向从向下变为向上。

在操作期间，通过写 TT0CCRn 寄存器可以改变 PWM 波形。在操作期间要想改变 PWM 波形，就要最后写 TT0CCR1 寄存器。

通过 TT0CCR0 寄存器设置三角形的 PWM 波形的周期，由 TT0CCR1 寄存器设置它的占空比。设置 TT0CCR0 寄存器的值的范围为“ $0 \leq \text{TT0CCR0} \leq \text{FFFEH}$ ”。当 16 位计数器向下计数达到 0000H 时，会反映重写值。

即使仅当改变 PWM 波形的周期时，应首先把一个周期设置给 TT0CCR0 寄存器，然后将相同值（与 TT0CCR1 寄存器预设值相同的值）写入 TT0CCR1 寄存器。

要将数据从 TT0CCRn 寄存器传输至 CCRn 缓冲寄存器，就必须将数据写入 TT0CCR1 寄存器 (n=0、1)。

9.6.9 编码器计数功能

编码器计数功能包括一个编码器比较模式（参见 9.6.10 编码器比较模式（TT0MD3 至 TT0MD0 位=1000））。

模式	TT0CCR0 寄存器	TT0CCR1 寄存器
编码器比较模式	仅比较模式	仅比较模式

(1) 向上/下计数控制

16 位计数器向上或向下计数由输入编码器信号（TENC00 和 TENC01）的相位及 TT0CTL2.TT0UDS1 和 TT0CTL2.TT0UDS0 位的设置情况控制的。

当使用编码器计数功能时，不能使用内部计数时钟和外部事件计数输入（EVTT0）。将 TT0CTL0.TT0CKS2 至 TT0CTL0.TT0CKS0 位置为 000，并将 TT0CTL1.TT0EEE 位置为 0。

(2) 设置 16 位计数器的初始值

计时器开始操作（TT0CTL0.TT0CE 位=0 → 1）之后，立即将在 TT0CTL2.TT0ECC 位=0 时设置给 TT0TCW 寄存器的初始计数值传输给 16 位计数器，在检测到编码器输入信号（TENC00 或 TENC01）的有效沿后，计数器开始操作。

(3) 基本操作

当 16 位计数器的计数值与 CCRn 缓冲寄存器的值匹配时，TT0CCRn 寄存器会生成比较匹配中断请求信号（INTTT0CCn）。

(4) 清除操作

在编码器比较模式中，当满足下列情况时，清除 16 位计数器。

- 当 16 位计数器的值与比较寄存器（设置了 TT0CTL2.TT0ECM1 和 TT0CTL2.TT0ECM0 位）的值匹配时
- 当检测到编码器清除输入信号（TECR0）的边沿时（当 TT0IOC3.TT0SCE 位=0 时，设置 TT0ECS1 和 TT0ECS0 位）
- 当检测到 TENC00、TENC01 和 TECR0 引脚的清除电平状态时（当 TT0SCE 位=1 时，设置 TT0ZCL、TT0BCL 和 TT0ACL 位）

备注 n=0、1

(5) TT0CTL2 寄存器的控制位

在编码器比较模式中，TT0CTL2 寄存器的设置如下。

表 9-8. TT0CTL2 寄存器的设置

模式	TT0UDS1、 TT0UDS0 位 (<1>)	TT0ECM1 位 (<2>)	TT0ECM0 位 (<2>)	TT0LDE 位 (<3>)	计数器清零 (目 标比较寄存器)	传输至计数器
编码器比较模式	可以置为 00、 01、10 或 11。	0	0	0	—	—
				1		可能
		1	1	0	TT0CCR0	—
				1		可能 ^注
	1	1	0	无效	TT0CCR1	—
			1	无效	TT0CCR0, TT0CCR1	—

注 计数器可以在从 0000H 到 TT0CCR0 寄存器的设置值的范围中进行操作。

(a) 各个位的概述

- <1> 根据从编码器输入引脚 (TENC00 或 TENC01) 输入的相位，TT0UDS1 和 TT0UDS0 位识别 16 位计数器的计数方向 (上或下)。
- <2> 当 16 位计数器的计数值与 CCR0 或 CCR1 缓冲寄存器的值匹配时，TT0ECM1 和 TT0ECM0 位控制计数器的清除操作。
- <3> 当计数器下溢出时，TT0LDE 位控制着将 TT0CCR0 寄存器的设置值传输至 16 位计数器的功能。TT0LDE 位仅在 TT0ECM1 和 TT0ECM0 位分别为 00 和 01 时有效。这些位在被置成其他值时无效。

(b) 各个位的详细说明

<1> TT0UDS1 和 TT0UDS0 位：向上/下计数选择

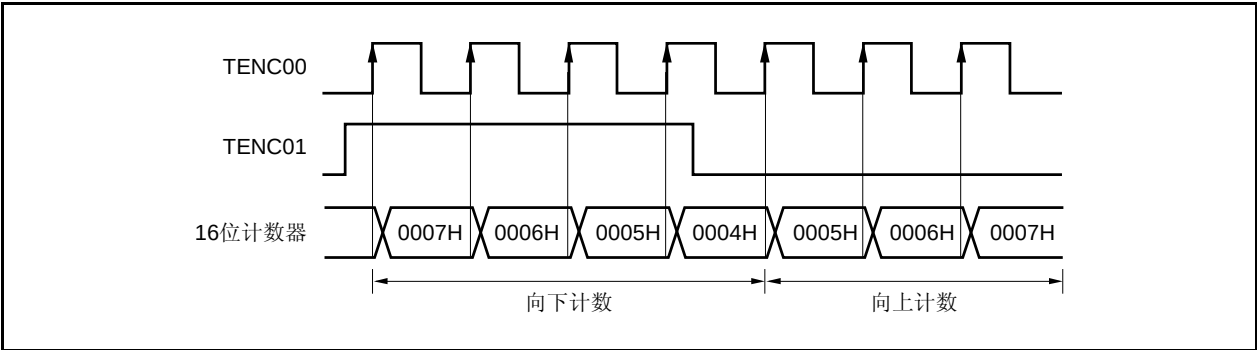
16 位计数器是向上还是向下计数取决于从 TENC00 或 TENC01 引脚输入的相位以及 TT0UDS1 和 TT0UDS0 位的设置。这些位仅在编码器比较模式中有效。

- 当 TT0UDS1 和 TT0UDS0 位=00 时

TENC00 引脚	TENC01 引脚	计数操作
上升沿	高电平	向下计数
下降沿		
双边沿		
上升沿	低电平	向上计数
下降沿		
双边沿		

备注 TENC00 引脚边缘的检测由 TT0IOC3.TT0EIS1 和 TT0EIS0 位指定。

图 9-45. 操作示例（当 TENC00 引脚有效沿指定为上升沿且没有为 TENC01 引脚指定有效边沿）

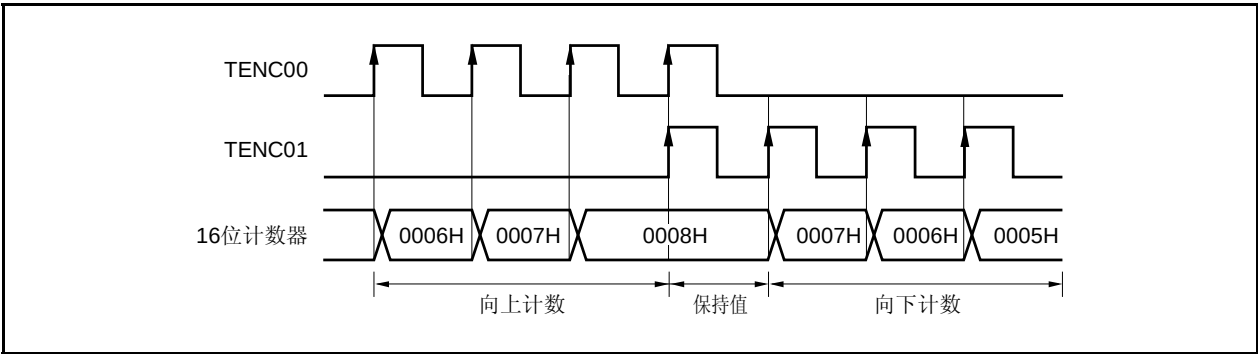


- 当 TT0UDS1 和 TT0UDS0 位=01 时

TENC00 引脚	TENC01 引脚	计数操作
低电平	上升沿	向下计数
	下降沿	
	双边沿	
高电平	上升沿	
	下降沿	
	双边沿	
上升沿	高电平	向上计数
下降沿		
双边沿		
上升沿	低电平	
下降沿		
双边沿		
同时输入至 TENC00 和 TENC01 引脚		计数器不执行计数操作，而是保持之前的瞬时值。

备注 TENC00 和 TENC01 引脚边缘的检测由 TT0IOC3.TT0EIS1 和 TT0IOC3.TT0EIS0 位指定。

图 9-46. 操作示例（当将上升沿指定为 TENC00 和 TENC01 引脚的有效沿时）



- 当 TT0UDS1 和 TT0UDS0 位=10 时

TENC00 引脚	TENC01 引脚	Count Operation
低电平	下降沿	计数器不执行计数操作，而是保持之前的瞬时值。
上升沿	低电平	向下计数
高电平	上升沿	计数器不执行计数操作，而是保持之前的瞬时值。
下降沿	高电平	
上升沿	高电平	
高电平	下降沿	向上计数
下降沿	低电平	
低电平	上升沿	
上升沿	下降沿	向下计数
下降沿		向上计数

注意事项 TENC00 和 TENC01 引脚的有效沿的指定无效。

图 9-47. 操作示例（当 TENC00 和 TENC01 引脚的有效沿不发生重叠时进行计数操作）

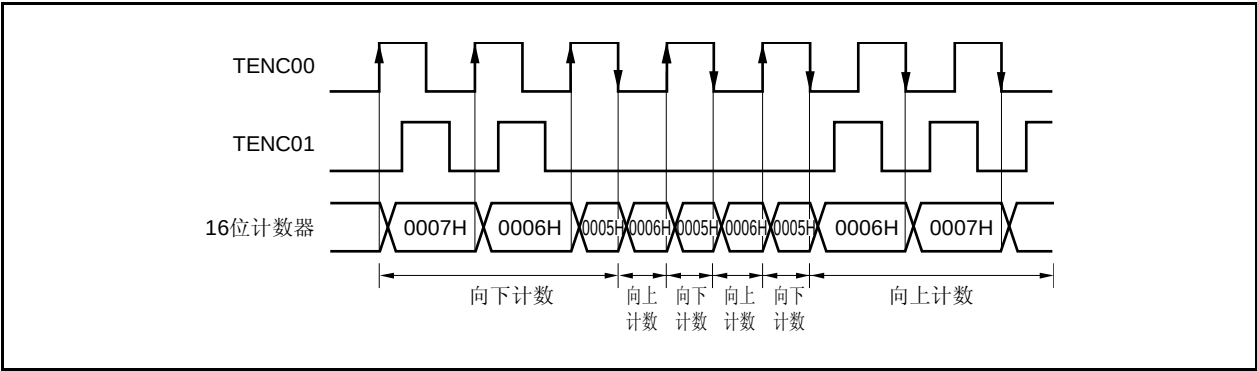
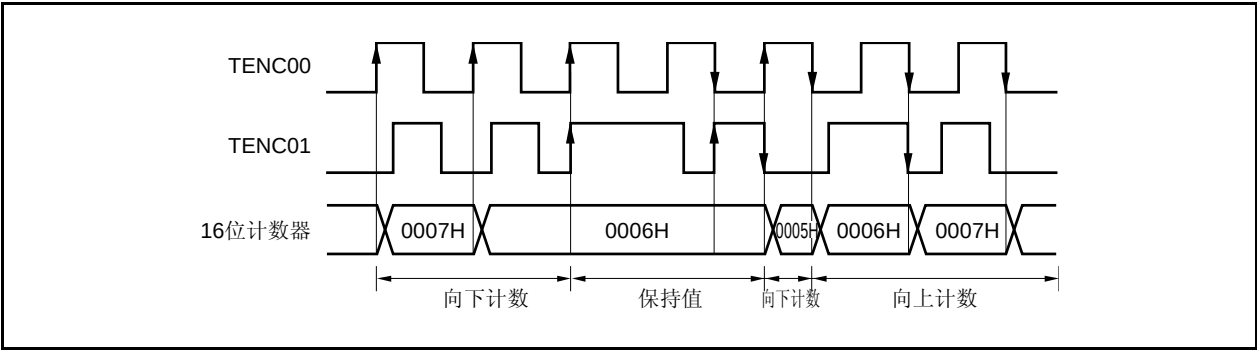


图 9-48. 操作示例（当 TENC00 和 TENC01 引脚的有效沿发生重叠时进行计数操作）



- 当 TT0UDS1 和 TT0UDS0 位=11 时

TENC00 Pin	TENC01 Pin	Count Operation
低电平	下降沿	向下计数
上升沿	低电平	
高电平	上升沿	
下降沿	高电平	
上升沿		
高电平	下降沿	向上计数
下降沿	低电平	
低电平	上升沿	
同时输入至 TENC00 和 TENC01 引脚		

注意事项 TENC00 和 TENC01 引脚的有效沿的指定无效。

图 9-49. 操作示例（当 TENC00 和 TENC01 引脚的有效沿不发生重叠时进行计数操作）

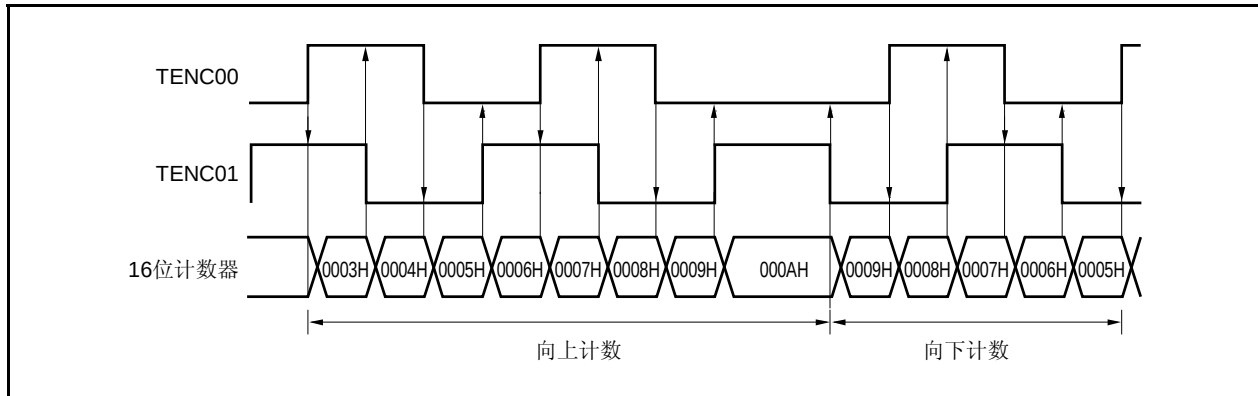
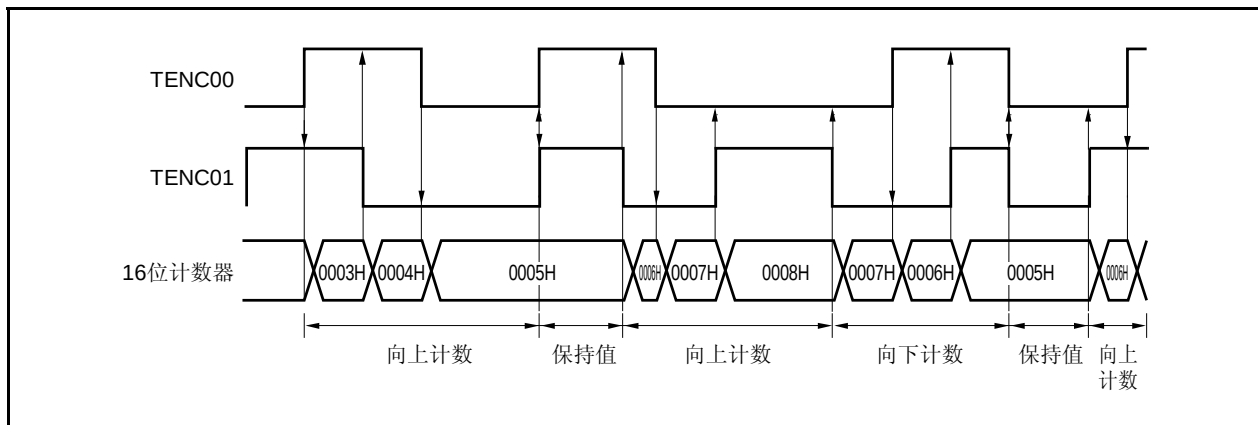


图 9-50. 操作示例（当 TENC00 和 TENC01 引脚的有效沿发生重叠时进行计数操作）



<2> TT0ECM1 和 TT0ECM0 位：与比较寄存器匹配之际定时器/计数器清除功能

当 16 位计数器的计数值与 CCRn 缓冲寄存器的值匹配时，计数器执行对应 TT0ECM1 和 TT0ECM0 位设置值的计数操作。

- 当 TT0ECM1 和 TT0ECM0 位=00 时
当 16 位计数器的值与 CCRn 缓冲寄存器的值匹配时，不将计数器清零。
- 当 TT0ECM1 和 TT0ECM0 位=01 时
当 16 位计数器的计数值与 CCR0 缓冲寄存器的值匹配时，计数器执行下列情况中的计数操作。

下一次计数操作	描述
向上计数	16 位计数器清除为 0000H。
向下计数	向下计数 16 位计数器的计数值。

- 当 TT0ECM1 和 TT0ECM0 位=10 时
当 16 位计数器的计数值与 CCR1 缓冲寄存器的值匹配时，计数器执行下列情况中的计数操作。

下一次计数操作	Description
向上计数	向上计数 16 位计数器的计数值。
向下计数	16 位计数器清除为 0000H。

- 当 TT0ECM1 和 TT0ECM0 位=11 时
当 16 位计数器的计数值与 CCR0 缓冲寄存器的值匹配时，计数器执行下列情况中的计数操作。

下一次计数操作	Description
向上计数	16 位计数器清除为 0000H。
向下计数	向下计数 16 位计数器的计数值。

当 16 位计数器的计数值与 CCR1 缓冲寄存器的值匹配时，计数器执行下列情况中的计数操作。

下一次计数操作	Description
向上计数	向上计数 16 位计数器的计数值。
向下计数	16 位计数器清除为 0000H。

<3> TT0LDE 位：当计数器下溢出时，将寄存器的设定值传输至 16 位计数器的功能
 当 TT0LDE 位=1 时，在计数器下溢出时 TT0CCR0 寄存器的设定值可以传输至 16 位计数器。
 TT0LDE 位仅在编码器比较模式中有效。

- 计数操作的范围是：0000H 到 TT0CCR0 寄存器的设定值。
 如果在 TT0LDE 位=1 且 TT0ECM1 和 TT0ECM0 位=01 时 16 位计数器执行一个计数操作，那么当计数器的计数值与 CCR0 缓冲寄存器的设定值匹配且当 TT0ECM0 位=1 时，若下次计数操作为向上计数，则 16 位计数器清除为 0000H。
 如果 16 位计数器在 TT0LDE 位=1 时下溢出，在则 TT0CCR0 寄存器的设定值传输至计数器。
 因此，计数器可以工作的范围是：0000H 到 TT0CCR0 寄存器的设定值，即上限计数值是 TT0CCR0 寄存器的设定值，下限值是 0000H。

图 9-51. 操作示例（从 0000H 到 TT0CCR0 寄存器设定值范围内的计数操作）

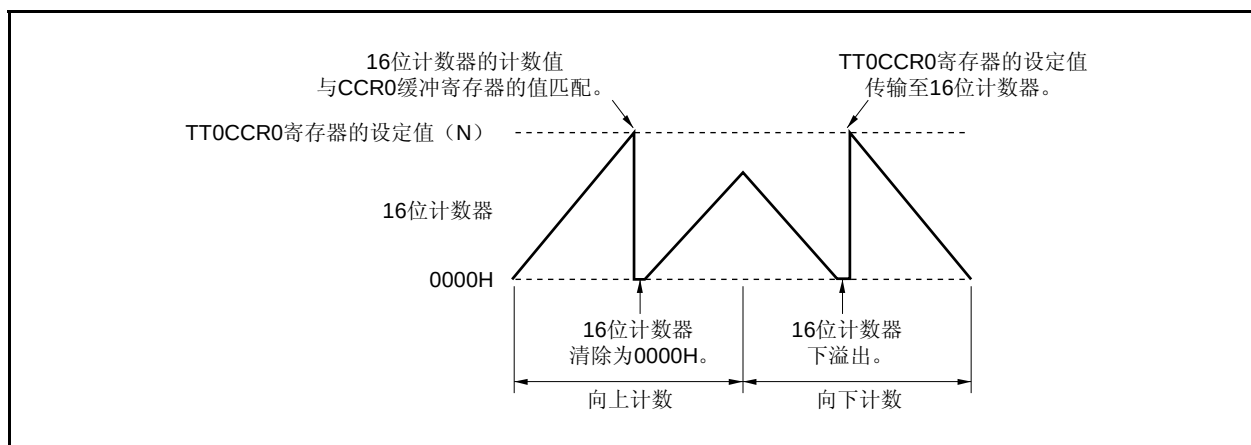
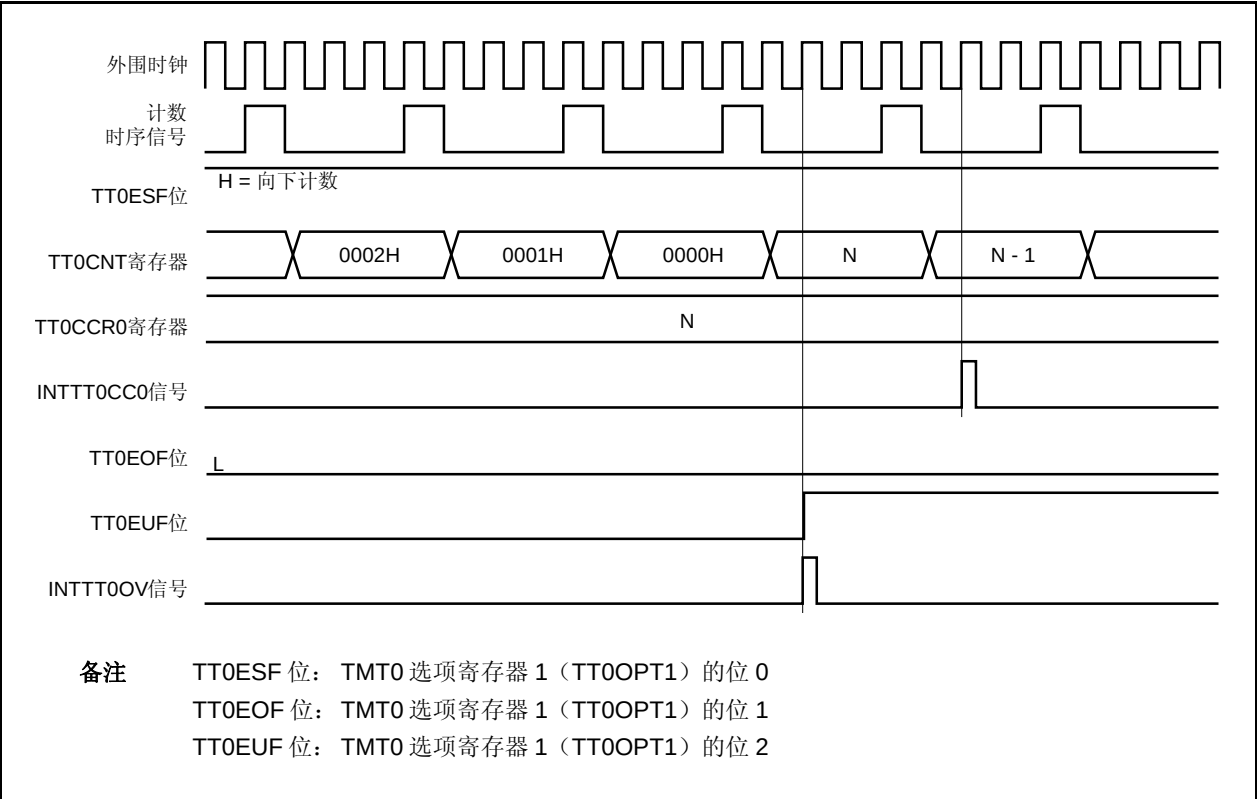


图 9-52. 操作时序（从 0000H 到 TT0CCR0 寄存器设定值范围内的计数操作）



(6) 通过编码器清除信号（TECR0 引脚）将计数器清除为 0000H 的功能

通过 TECR0 引脚的输入信号，16 位计数器可以由 TT0IOC3.TT0SCE 位选择的两种方法清除为 0000H。根据位的设置，TT0SCE 位还能控制 TT0IOC3.TT0ZCL、TT0IOC3.TT0BCL、TT0IOC3.TT0ACL、TT0IOC3.TT0ESC1 和 TT0IOC3.TT0ECS0 位。

只有在编码器比较模式中，可以通过下面描述的方法清除计数器。

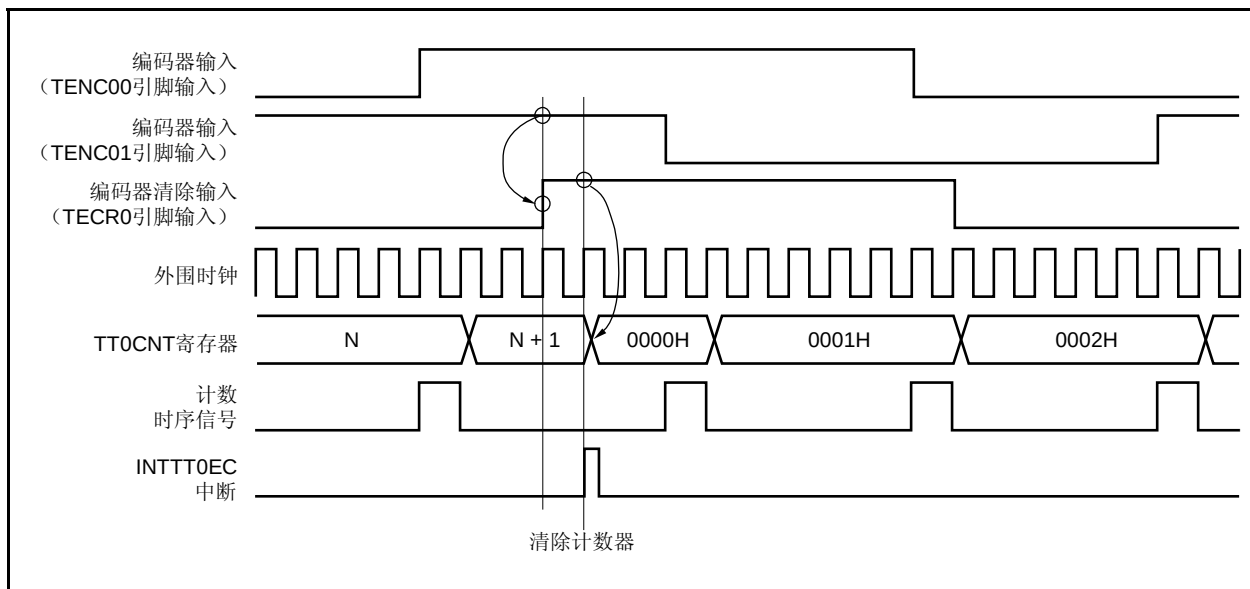
表 9-9. TT0SCE 位和 TT0ZCL、TT0BCL、TT0ACL、TT0ECS1 和 TT0ECS0 位之间的关系

清除方法	TT0SCE 位	TT0ZCL 位	TT0BCL 位	TT0ACL 位	TT0ECS1、TT0ECS0 位
<1>	0	无效	无效	无效	有效
<2>	1	有效	有效	有效	无效

(a) 清除方法 <1>：通过检测编码器清除信号（TECR0 引脚）的边沿（TT0SCE 位=0）

当 TT0SCE 位=0 时，如果检测到由 TT0ECS1 和 TT0ECS0 位指定的 TECR0 引脚的有效沿，则 16 位计数器与外围时钟同步清除为 0000H。此时，生成编码器清除中断请求信号（INTTT0EC）。当 TT0SCE 位=0 时，TT0ZCL、TT0BCL 和 TT0ACL 位的设置无效。

图 9-53. 操作示例（当 TT0SCE 位=0、TT0ECS1 和 TT0ECS0 位=01、TT0UDS1 和 TT0UDS0 位=11 时）



- (b) 清除方法 <2>: 通过检测 TENC00、TENC01 和 TECR0 引脚的清除电平状态 (TT0SCE 位=1)
- 当 TT0SCE 位=1 时, 如果检测到由 TT0ZCL、TT0BCL 和 TT0ACL 位指定的 TECR0、TENC00 或 TENC01 引脚的清除电平状态, 则 16 位计数器清除为 0000H。此时, 不生成编码器清除中断请求信号 (INTTT0EC)。当 TT0SCE 位=1 时, TT0ECS1 和 TT0ECS0 位的设置无效。

表 9-10. TT0SCE 位=1 时的 16 位计数器清除状态

清除电平状态设置			编码器引脚的输入电平		
TT0ZCL 位	TT0BCL 位	TT0ACL 位	TECR0 引脚	TENC01 引脚	TENC00 引脚
0	0	0	L	L	L
0	0	1	L	L	H
0	1	0	L	H	L
0	1	1	L	H	H
1	0	0	H	L	L
1	0	1	H	L	H
1	1	0	H	H	L
1	1	1	H	H	H

注意事项 当 TT0ZCL、TT0BCL 和 TT0ACL 位的清除电平状态与 TECR0、TENC01 或 TENC00 引脚的输入电平匹配时, 16 位计数器清除为 0000H。

图 9-54. 操作示例（当 TT0SCE 位=1、TT0ZCL 位=1、TT0BCL 位=0、TT0ACL 位=1、TT0UDS1 和 TT0UDS0 位=11、TECR0=高电平、TENC01=低电平、和 TENC00=高电平）（1/3）

(i) 当计数器向上计数时，如果将高电平输入到 **TECR0** 引脚滞后于将低电平输入到 **TENC01** 引脚，则计数器在向上计数之后被清除。

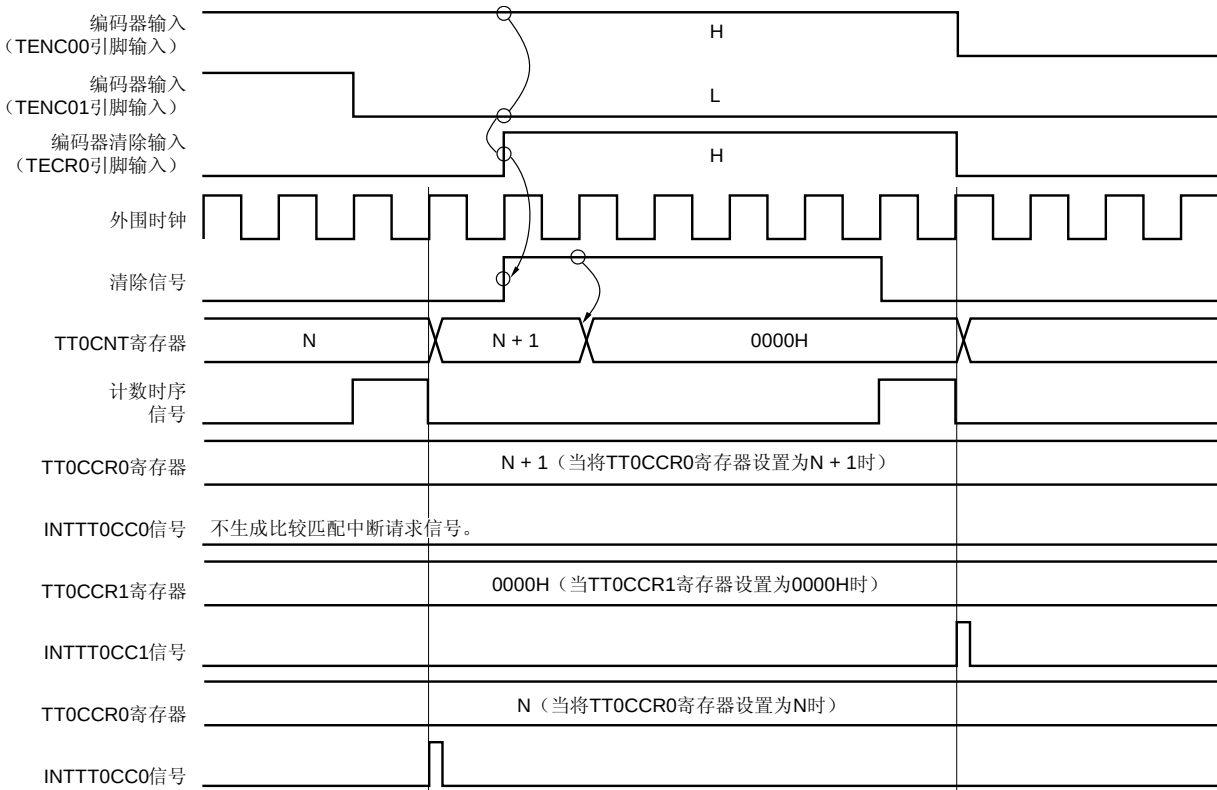


图 9-54. 操作示例（当 TT0SCE 位=1、TT0ZCL 位=1、TT0BCL 位=0、TT0ACL 位=1、TT0UDS1 和 TT0UDS0 位=11、TECR0=高电平、TENC01=低电平、和 TENC00=高电平）（2/3）

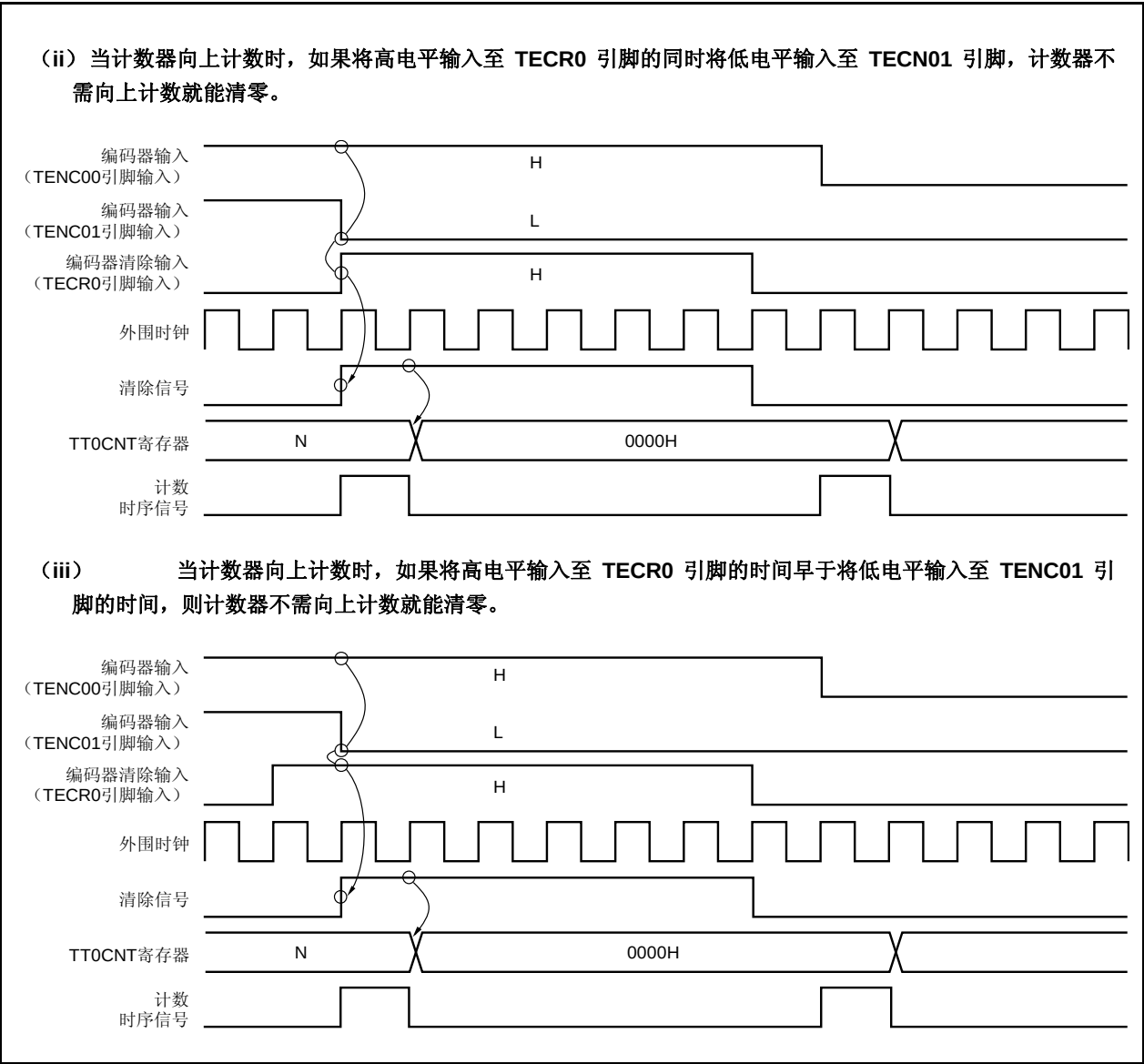
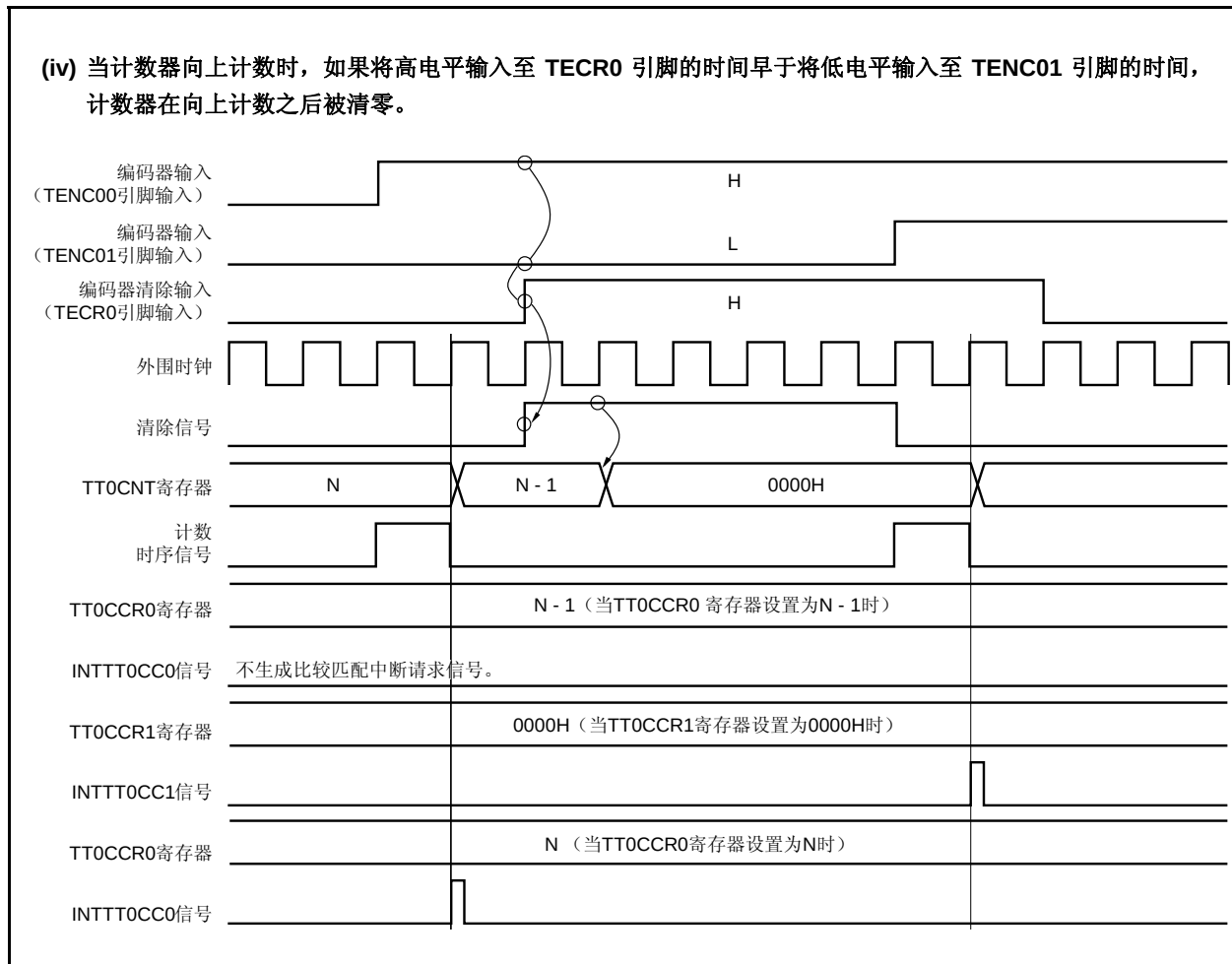


图 9-54. 操作示例（当 TT0SCE 位=1、TT0ZCL 位=1、TT0BCL 位=0、TT0ACL 位=1、TT0UDS1 和 TT0UDS0 位=11、TECR0=高电平、TENC01=低电平、和 TENC00=高电平）（3/3）

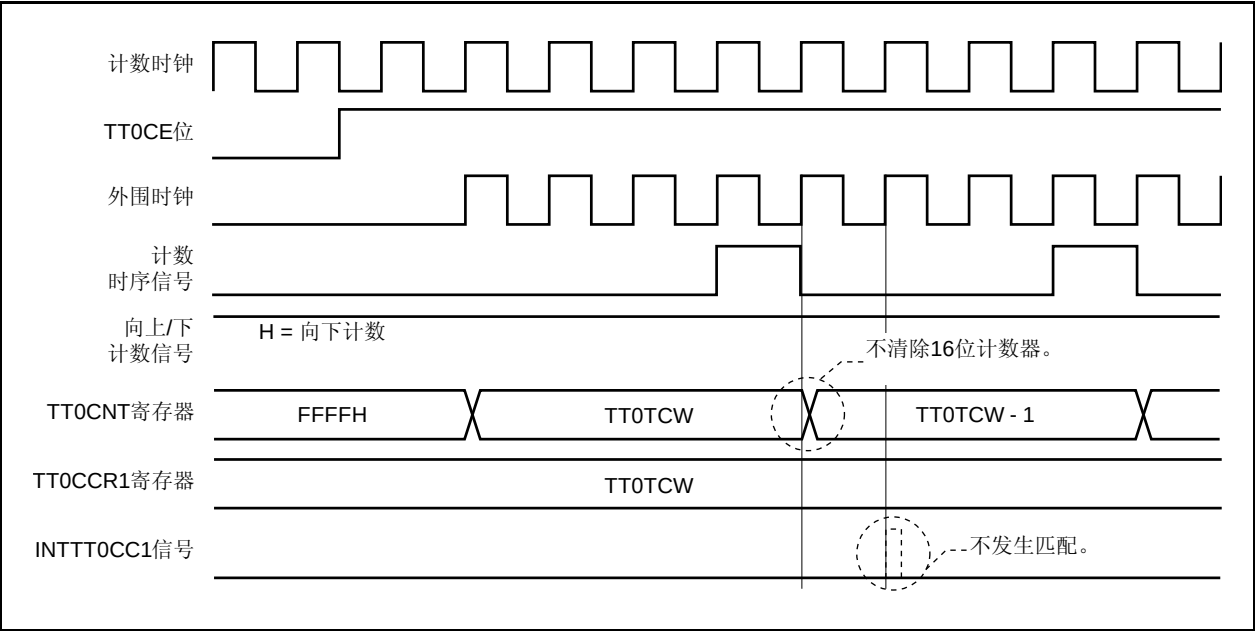


如果计数器按这种方式清零，即使最后向 TECR0 引脚输入信号也不会发生计数错误，这是因为设置了 TECR0、TENC01 和 TENC00 引脚的清除电平状态，且当检测到清除电平状态时 16 位计数器清除为 0000H。

(7) 有关编码器计数功能的注意事项

(a) 如果在操作开始后的瞬间不生成比较匹配中断

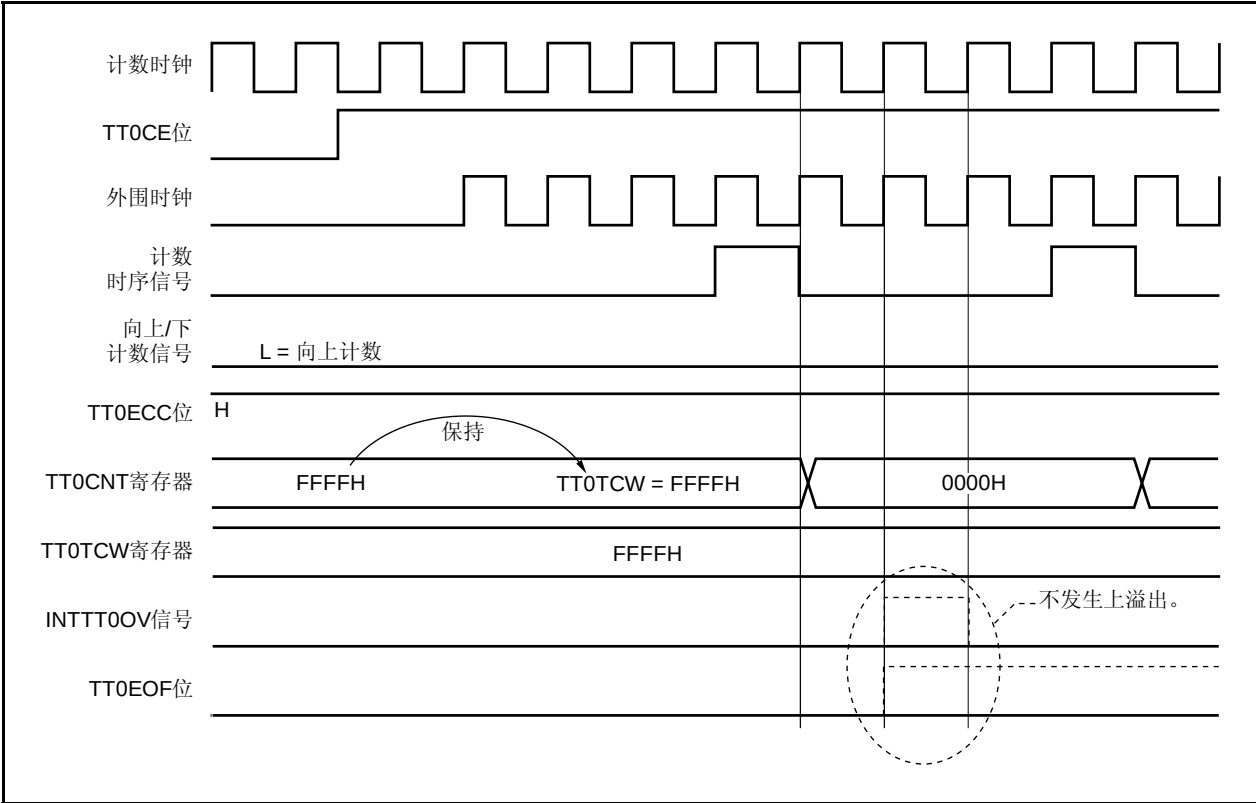
如果将与 TT0TCW 寄存器相同的值设置给 TT0CCR0 或 TT0CCR1 寄存器，并且当 TT0CTL2.TT0ECC 位 =0 时开始计数操作，如果在开始操作后的瞬间 16 位计数器的计数值 (TT0TCW) 与 CCRn 缓冲寄存器的值匹配，则屏蔽该匹配且不生成比较匹配中断请求信号 (INTTT0CCn) (n=0、1)。此外，通过设置 TT0CTL2.TT0ECM1 和 TT0CTL2.TT0ECM0 位，不能将 16 位计数器清除为 0000H。



(b) 如果在开始操作后的瞬间不发生上溢出

当 TT0CTL2.TT0ECC 位=1 时，如果重新开始计数操作，且 16 位计数器的计数值保持为 FFFFH 和下一次的计数操作为向上计数的话，计数器就不发生上溢出。

计数器开始操作且从一个计数值（TT0TCW 寄存器的值=FFFFH）向上计数之后，计数器从 FFFFH 到 0000H 发生上溢出。但是，上溢出的方向被屏蔽，不设置上溢出标志（TT0EOF），也不生成上溢出中断请求信号（INTTT0OV）。



9.6.10 编码器比较模式 (TT0MD3 至 TT0MD0 位=1000)

在编码器比较模式中，通过将 TT0CCR0 和 TT0CCR1 都用作比较寄存器和编码器计数功能的输入引脚，从而控制编码器。

在该模式中，16 位计数器可以通过三种方式清除为 0000H。当计数器的计数值与 CCRn 缓冲寄存器的值匹配时（生成比较匹配中断请求信号（INTTT0CCn）），当检测到编码器清除输入（TECR0 引脚）的边沿时，和当检测到 TENC00、TENC01 和 TECR0 引脚的清除电平状态时。

当 16 位计数器发生下溢出时，TT0CCR0 寄存器的设定值可以传输至计数器。

(1) 编码器比较模式操作流程

图 9-55. 编码器比较模式操作流程

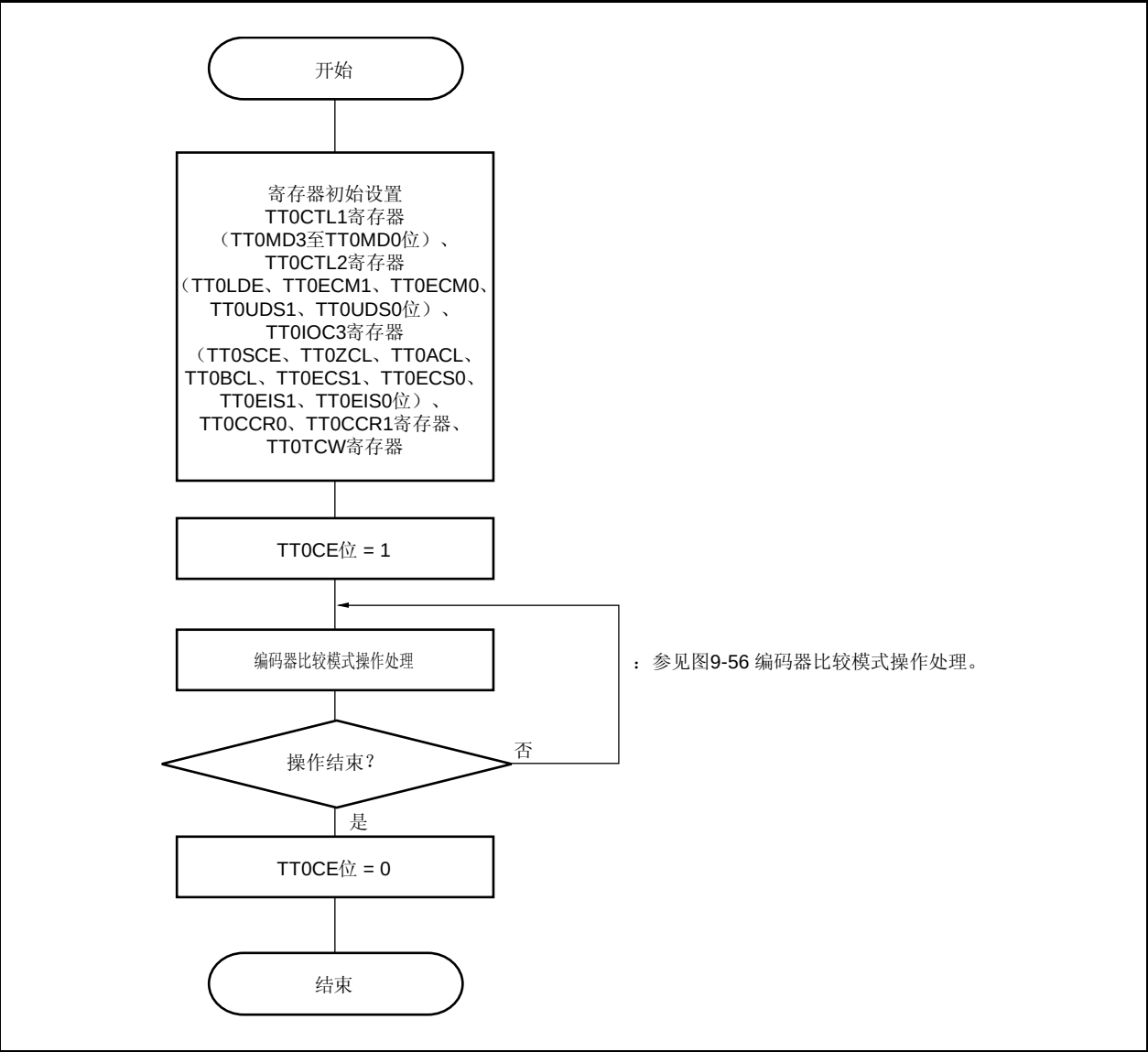
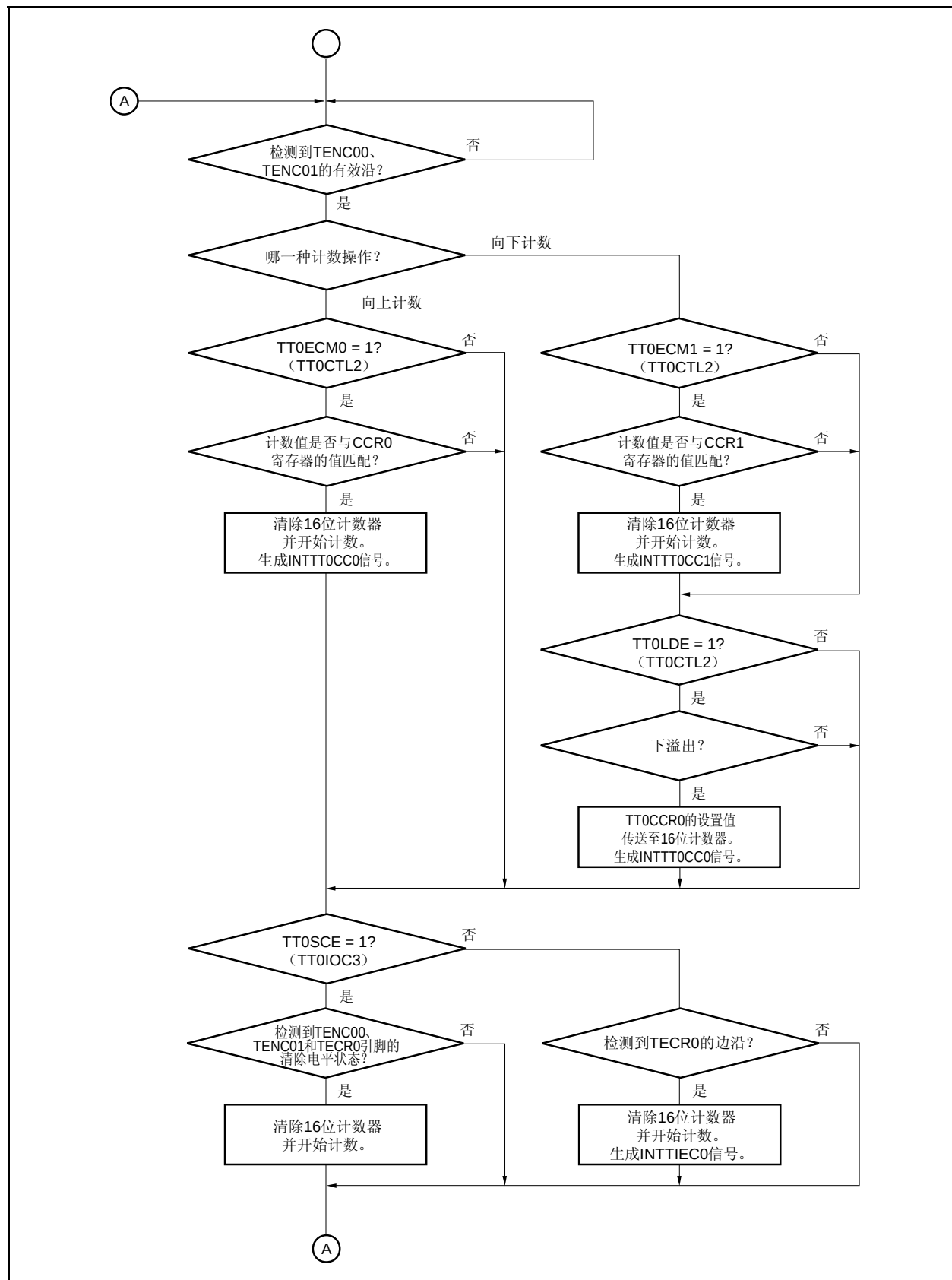


图 9-56. 编码器比较模式操作处理

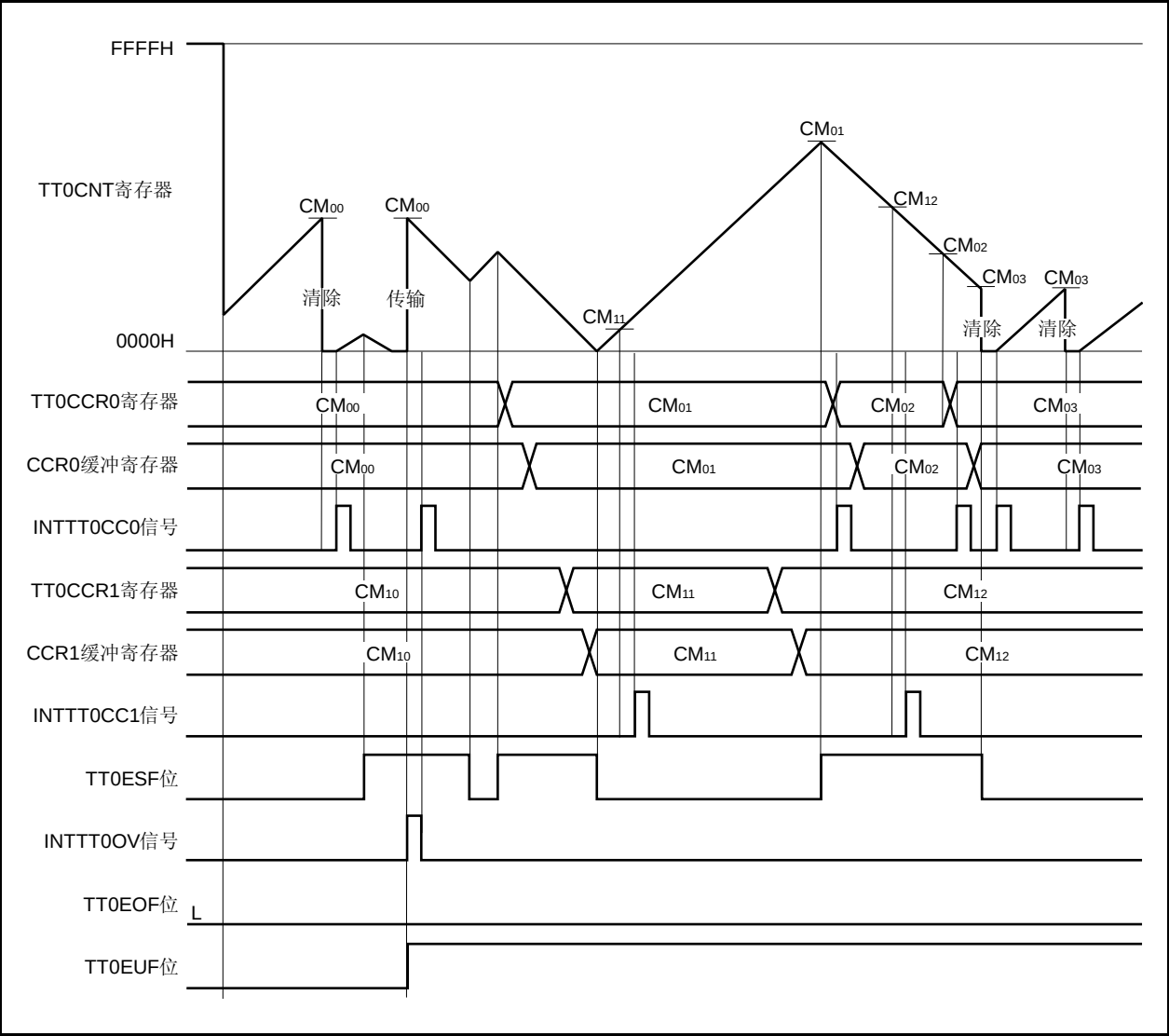


(2) 编码器比较模式操作时序

(a) 基本时序 1

[寄存器设置条件]

- TT0CTL2.TT0ECM1 和 TT0CTL2.TT0ECM0 位=01
当 16 位计数器的计数值与 CCR0 缓冲寄存器的值匹配时，计数器清除为 0000H。
- TT0CTL2.TT0LDE 位=1
当发生上溢出时，TT0CCR0 寄存器的设置值传送至 16 位计数器。
- TT0IOC3.TT0SCE 位=0、TT0IOC3.TT0ECS1 和 TT0IOC3.TT0ECS0 位=00
当检测到编码器输入信号（TECR0 引脚）的边沿时有关于清除 16 位计数器的说明（没有指定边沿）



当 16 位计数器开始操作 (TT0CE 位=0 → 1) 时, TT0TCW 寄存器的设置值传送至计数器, 并且 16 位计数器开始操作。

当计数器的计数值与 CCR0 缓冲寄存器的值匹配时, 就会生成比较匹配中断请求信号 (INTTT0CC0)。因为 TT0ECM0 位=1, 所以若下一次计数操作为向上计数则 16 位计数器清除为 0000H。

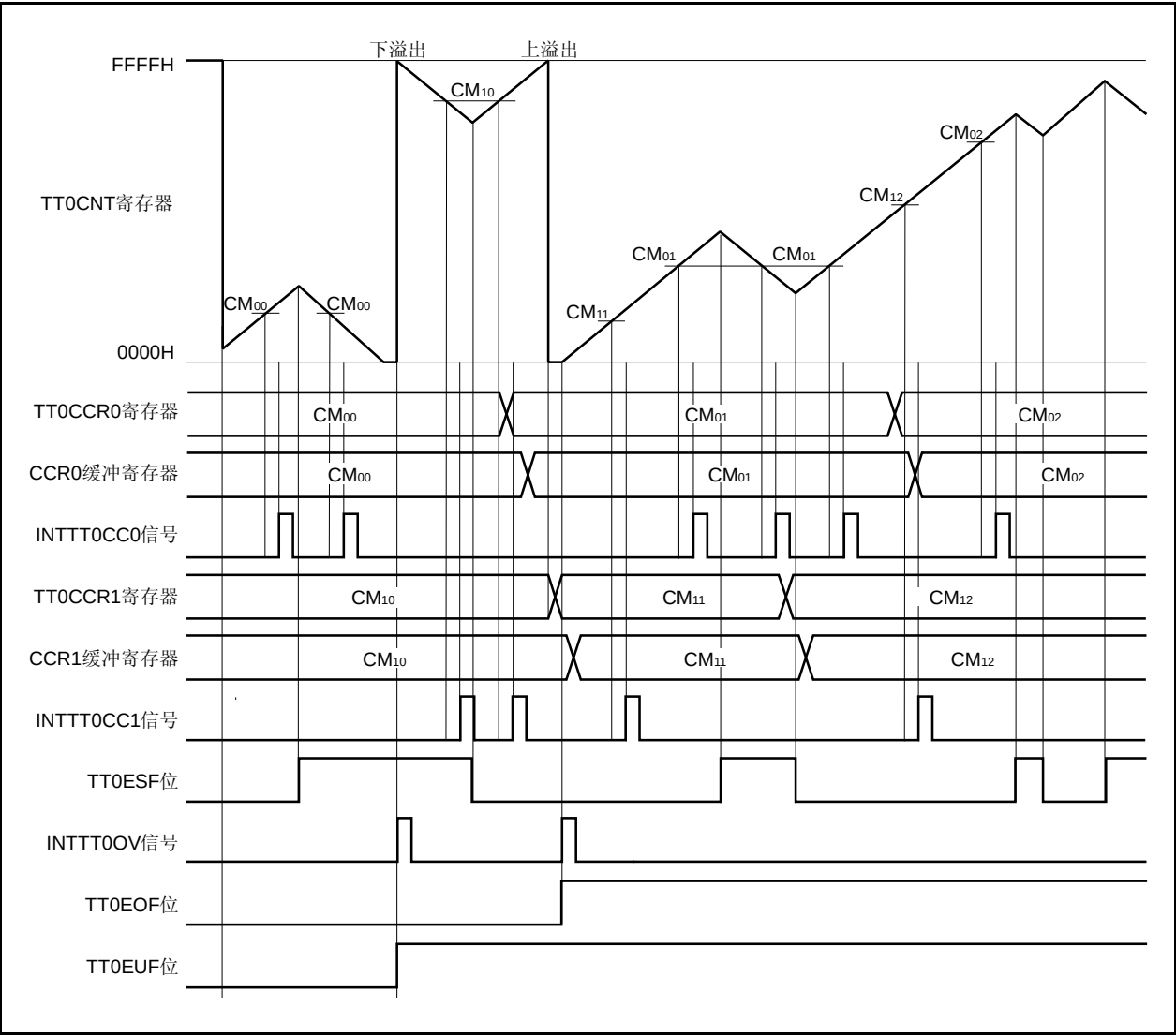
当 16 位计数器的计数值和 CCR1 缓冲寄存器的值匹配时, 会生成比较匹配中断请求信号 (INTTT0CC1)。因为 TT0ECM1 位=0, 所以当 16 位计数器的值与 CCR1 缓冲寄存器的值匹配时, 计数器不清除为 0000H。

当 TT0LDE 位=1 且 TT0ECM0 位=1 时, 计数器的操作范围是: 0000H 到 TT0CCR0 寄存器的设定值。

(b) 基本时序 2

[寄存器设置条件]

- TT0CTL2.TT0ECM1 和 TT0CTL2.TT0ECM0 位=00
即使当 16 位计数器的值与 CCRn 缓冲寄存器的值匹配（a=0、1）时，也不清除计数器。
- TT0CTL2.TT0LDE 位=0
计数器发生下溢出之后，不将 TT0CCR0 寄存器的设置值传送至 16 位计数器。
- TT0IOC3.TT0SCE 位=0、TT0IOC3.TT0ECS1 和 TT0IOC3.TT0ECS0 位=00
当检测到编码器输入信号（TECR0 引脚）的边沿时有关于清除 16 位计数器的说明（没有指定边沿）



当 16 位计数器开始操作 (TT0CE 位=0 → 1) 时, TT0TCW 寄存器的设置值传送至 16 位计数器, 并且计数器开始操作。

当 16 位计数器的计数值与 CCR0 缓冲寄存器的值相符时, 就会生成一个比较匹配中断请求信号 (INTTT0CC0)。

当 16 位计数器的计数值和 CCR1 缓冲寄存器的值匹配时, 也会生成一个比较匹配中断请求信号 (INTTT0CC1)。

即使当 16 位计数器的计数值与 CCRn 缓冲寄存器的值匹配时, 也不将计数器清除为 0000H, 这是因为 TT0ECM1 和 TT0ECM0 位=00 (n=0、1)。

(c) 基本时序 3

[寄存器设置条件]

- TT0CTL2.TT0ECM1 和 TT0CTL2.TT0ECM0 位=11

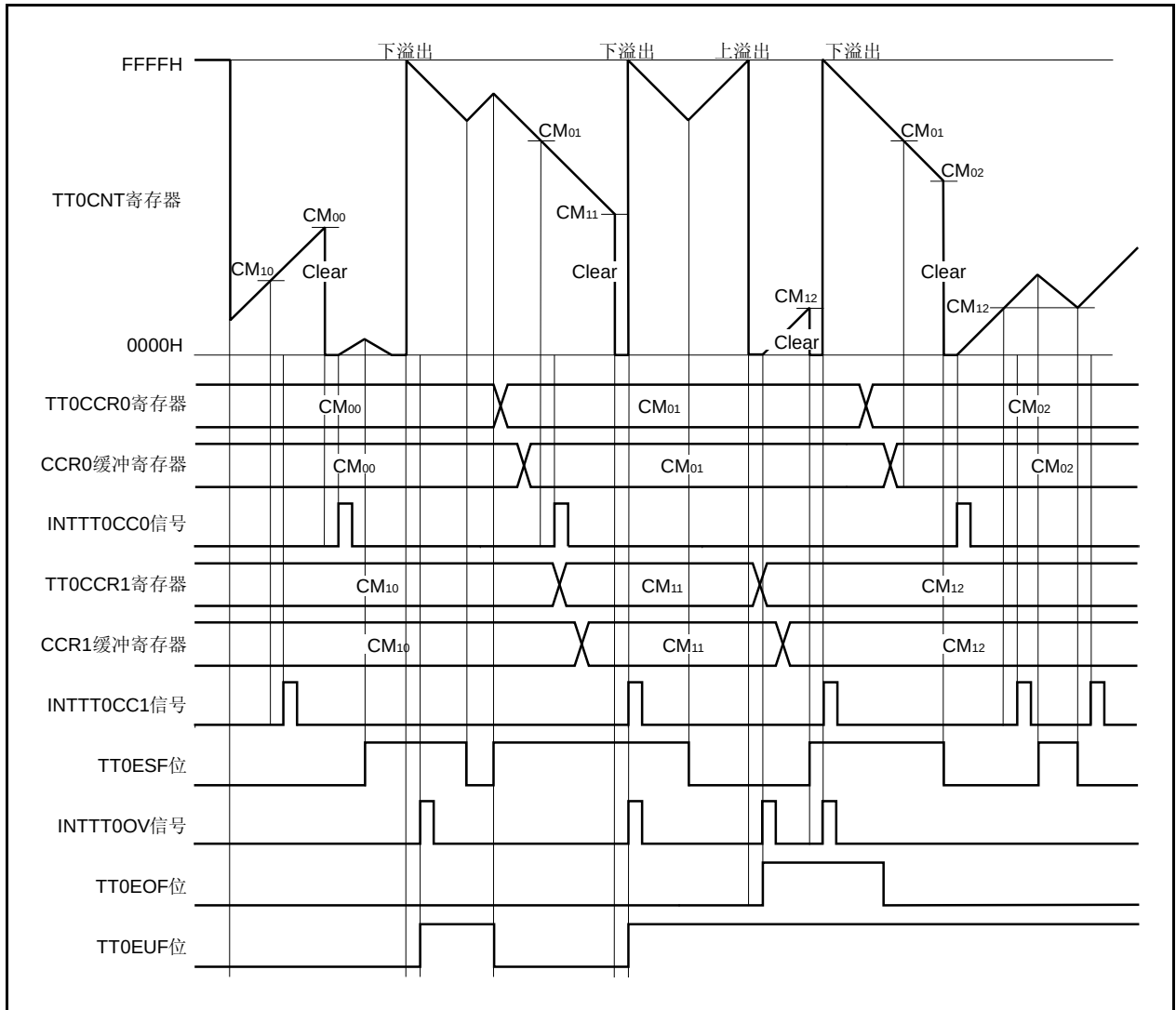
当 16 位计数器的值与 CCR0 缓冲寄存器的值匹配时，计数器的计数值清除为 0000H。

当 16 位计数器的值与 CCR1 缓冲寄存器的值匹配时，计数器的计数值清除为 0000H。

- TT0CTL2.TT0LDE 位的设置无效。

- TT0IOC3.TT0SCE 位=0、TT0IOC3.TT0ECS1 和 TT0IOC3.TT0ECS0 位=00

当检测到编码器输入信号（TECR0 引脚）的边沿时有关于清除 16 位计数器的说明（没有指定边沿）



当 16 位计数器开始操作 (TT0CE 位=0 → 1) 时, TT0TCW 寄存器的设置值传送至 16 位计数器, 并且计数器开始操作。

当 16 位计数器的计数值与 CCR0 缓冲寄存器的值相符时, 就会生成一个比较匹配中断请求信号 (INTTT0CC0)。此时, 如果下一次计数操作为向上计数, 则 16 位计数器清除为 0000H。

当 16 位计数器的计数值和 CCR1 缓冲寄存器的值匹配时, 也会生成一个比较匹配中断请求信号 (INTTT0CC1)。此时, 如果下一次计数操作为向下计数, 则 16 位计数器清除为 0000H。

第十章 16 位间隔定时器 M (TMM)

V850ES/JG3-H 和 V850ES/JH3-H 有四个 TMM 通道 (TMMn)。

10.1 概述

TMMn 具有以下特性:

- 间隔定时器功能
- 8 种时钟可选
- 16 位计数器 × 1
(定时器计数操作过程中不可对 16 位计数器进行读取。)
- 比较寄存器 × 1
(定时器计数操作过程中不可向比较寄存器写入数据。)
- 比较匹配中断 × 1

TMMn 只支持清零并启动模式。不支持自运行定时器模式。

备注 n = 0 至 3

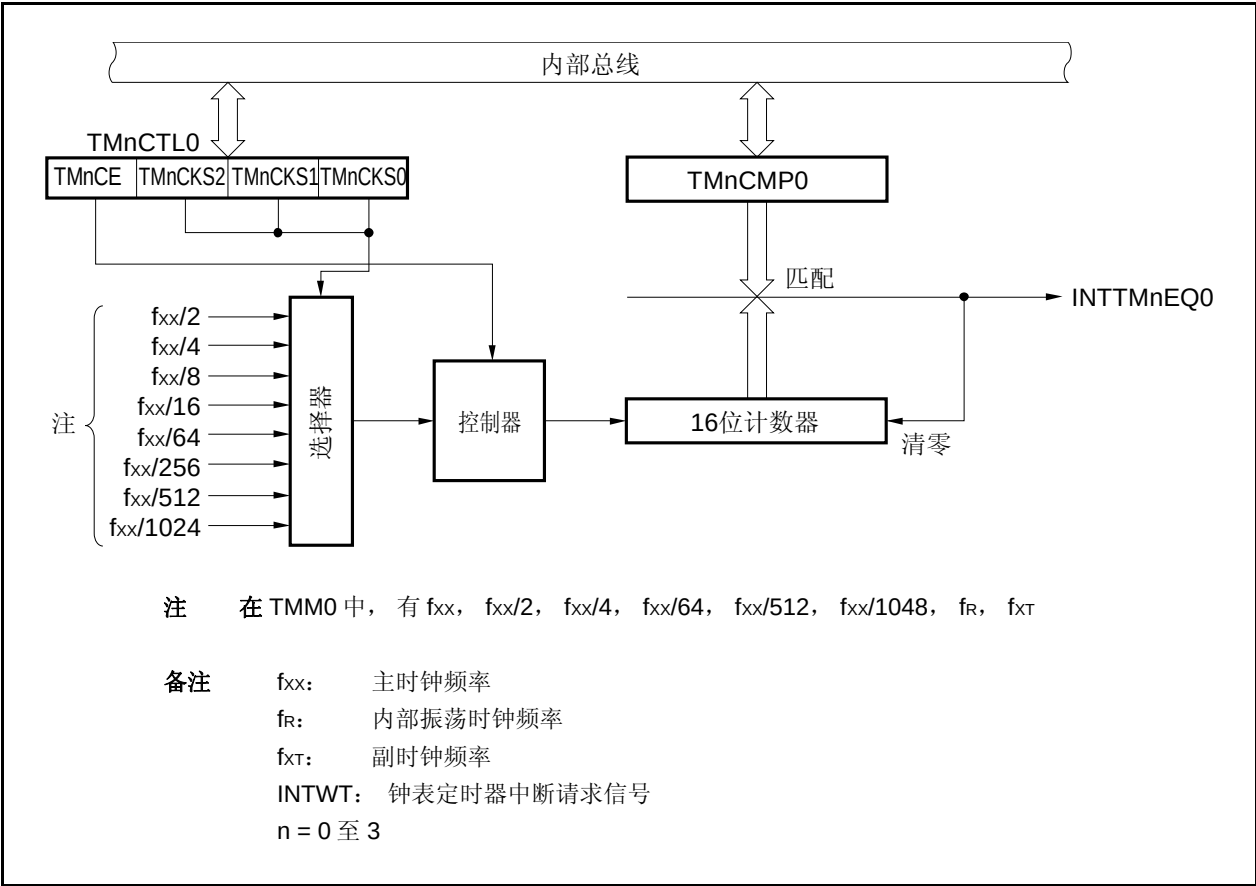
10.2 配置

TMMn 包括以下硬件：

表 10-1. TMMn 的配置

项目	配置
定时器寄存器	16 位计数器
寄存器	TMMn 比较寄存器 0（TMnCMP0）
控制寄存器	TMMn 控制寄存器 0（TMnCTL0）

图 10-1. TMMn 的功能框图

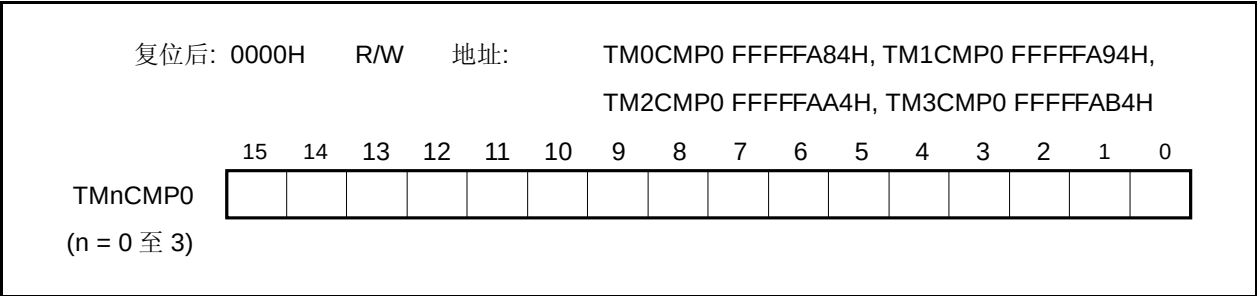


(1) 16 位计数器

该计数器是一个计数内部时钟的 16 位计数器。
该计数器不可以被读写。

(2) TMMn 比较寄存器 0（TMnCMP0）

TMnCMP0 寄存器是一个 16 位的比较寄存器。
该寄存器可按 16 位为单位进行读写。
复位后本寄存器设置为 0000H。
可使用软件总向 TMnCMP0 寄存器写入相同值。
当 TMnCTL0.TMnCE 位 = 1 时禁止重写 TMnCMP0 寄存器。



10.3 寄存器

(1) TMMn 控制寄存器 (TMnCTL0)

TMnCTL0 寄存器为 8 位寄存器，控制 TMMn 的操作。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

可使用软件一直 TMnCTL0 寄存器写入相同值。

备注 n = 0 至 3

复位后: 00H R/W 地址: TM0CTL0 FFFFFFFA80H, TM1CTL0 FFFFFFFA90H,
TM2CTL0 FFFFFFFAA0H, TM3CTL0 FFFFFFFAB0H

	<7>	6	5	4	3	2	1	0
TMnCTL0 (n = 0至 3)	TMnCE	0	0	0	0	TMnCKS2	TMnCKS1	TMnCKS0

TMnCE	内部时钟操作允许/禁止说明
0	禁止TMMn 操作 (16位计数器异步复位)。 操作时钟应用停止。
1	允许TMMn 操作。操作时钟应用开始。TMMn 操作开始。
TMMn的内部时钟控制和内部电路复位用 TMnCE 位异步执行。 当 TMnCE 位清除为0时，禁止使用 TMMnd 的内部时钟 (固定为低电平) 且16位计数器异步复位。	

(m = 0)

TMmCKS2	TMmCKS1	TMmCKS0		计数时钟选择		
				fxx = 48 MHz	fxx = 32 MHz	fxx = 24 MHz
0	0	0	fxx	20.8 ns	31.3 ns	41.7 ns
0	0	1	fxx/2	41.7 ns	62.5 ns	83.3 ns
0	1	0	fxx/4	83.3 ns	125 ns	167 ns
0	1	1	fxx/64	1.33 ms	2.00 ms	2.67 ms
1	0	0	fxx/512	10.7 ms	16.0 ms	21.3 ms
1	0	1	fxx/1024	21.3 ms	32.0 ms	42.7 ms
1	1	0	fR/8	36.4 ms	36.4 ms	36.4 ms
1	1	1	fXT	30.5 ms	30.5 ms	30.5 ms

(m = 1 至 3)

TMmCKS2	TMmCKS1	TMmCKS0		计数时钟选择		
				fxx = 48 MHz	fxx = 32 MHz	fxx = 24 MHz
0	0	0	fxx/2	41.7 ns	62.5 ns	83.3 ns
0	0	1	fxx/4	83.3 ns	125 ns	167 ns
0	1	0	fxx/8	167 ns	250 ns	333 ns
0	1	1	fxx/16	333 ns	500 ns	667 ns
1	0	0	fxx/64	1.33 ms	2.00 ms	2.67 ms
1	0	1	fxx/256	5.33 ms	8.00 ms	10.7 ms
1	1	0	fxx/512	10.7 ms	16.0 ms	21.3 ms
1	1	1	fxx/1024	21.3 ms	32.0 ms	42.7 ms

- 注意事项**
1. 当 TMnCE 位=0 时，设置 TMnCKS2 至 TMnCKS0 位。
当将 TMnCE 的值从 0 改变为 1 时，不可以同时设置 TMnCKS2 至 TMnCKS0 位的值。
 2. 务必将位 3 至位 6 清为“0”。

备注

fxx: 主时钟频率
fR: 内部振荡时钟频率
fXT: 副时钟频率

10.4 操作

注意事项 不要将 TMnCMP0 寄存器设置为 FFFFH。

10.4.1 间隔定时器模式

在间隔定时器模式中，如果 TMnCTL0.TMnCE 位被设为 1，那么中断请求信号（INTTMnEQ0）就会在指定间隔中生成。

图 10-2. 间隔定时器的配置

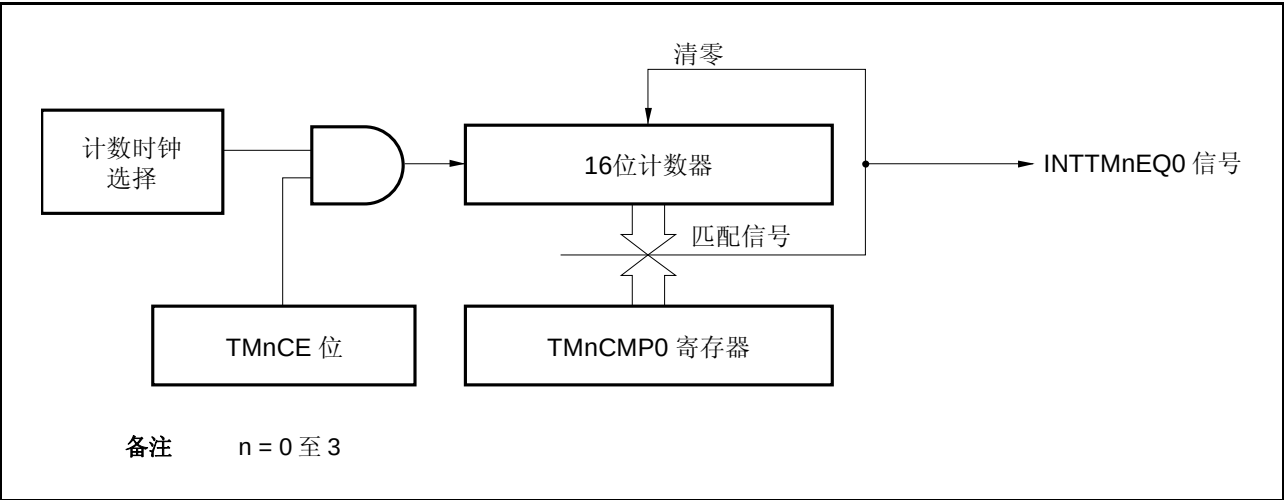
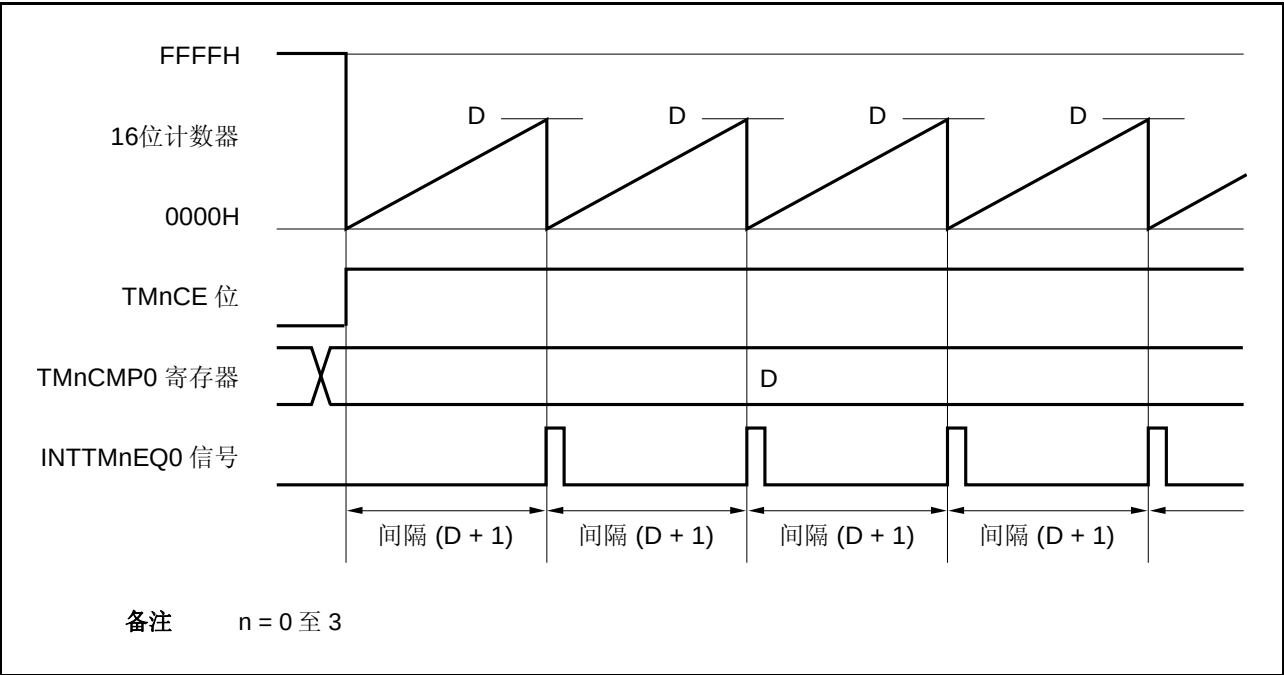


图 10-3. 间隔定时器工作的基本时序



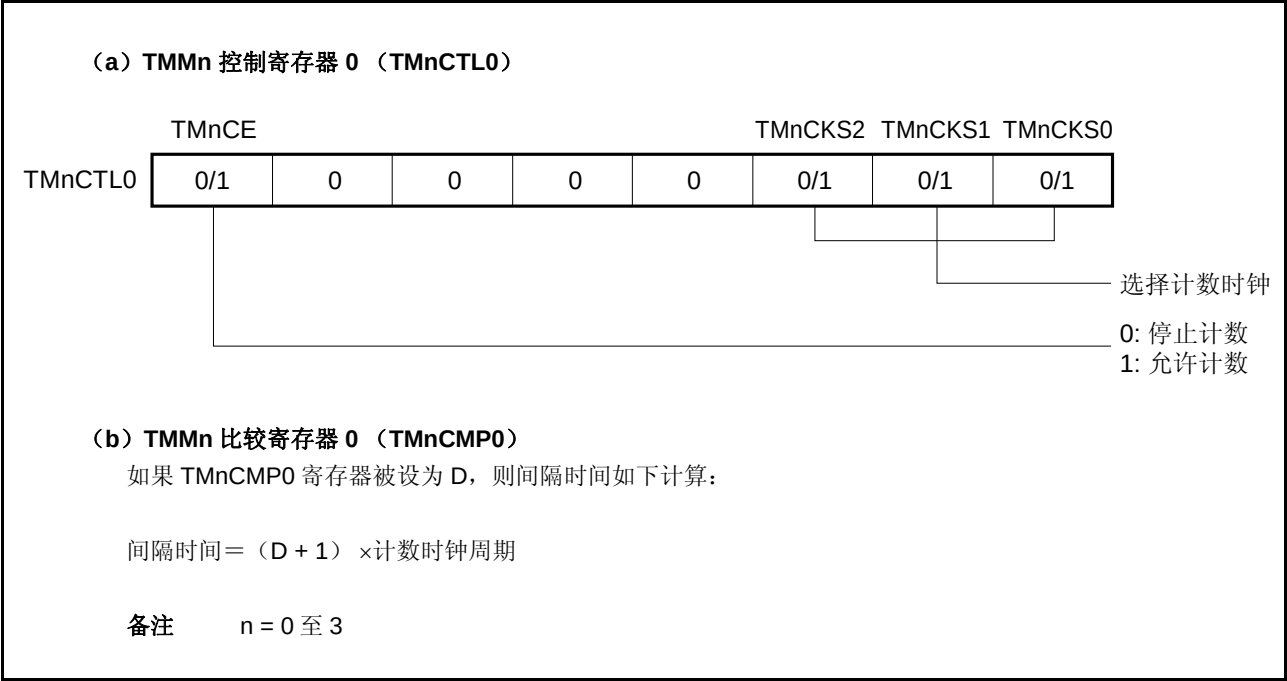
当 TMnCE 位被设置为 1，16 位计数器会与计数时钟同步将计数值从 FFFFH 清零为 0000H，并开始计数。

当 16 位计数器的计数值和 TMnCMP0 寄存器的数值相匹配时，16 位计数器清零为 0000H，同时生成一个比较匹配中断请求信号（INTTMnEQ0）。

时间间隔可用下式计算。

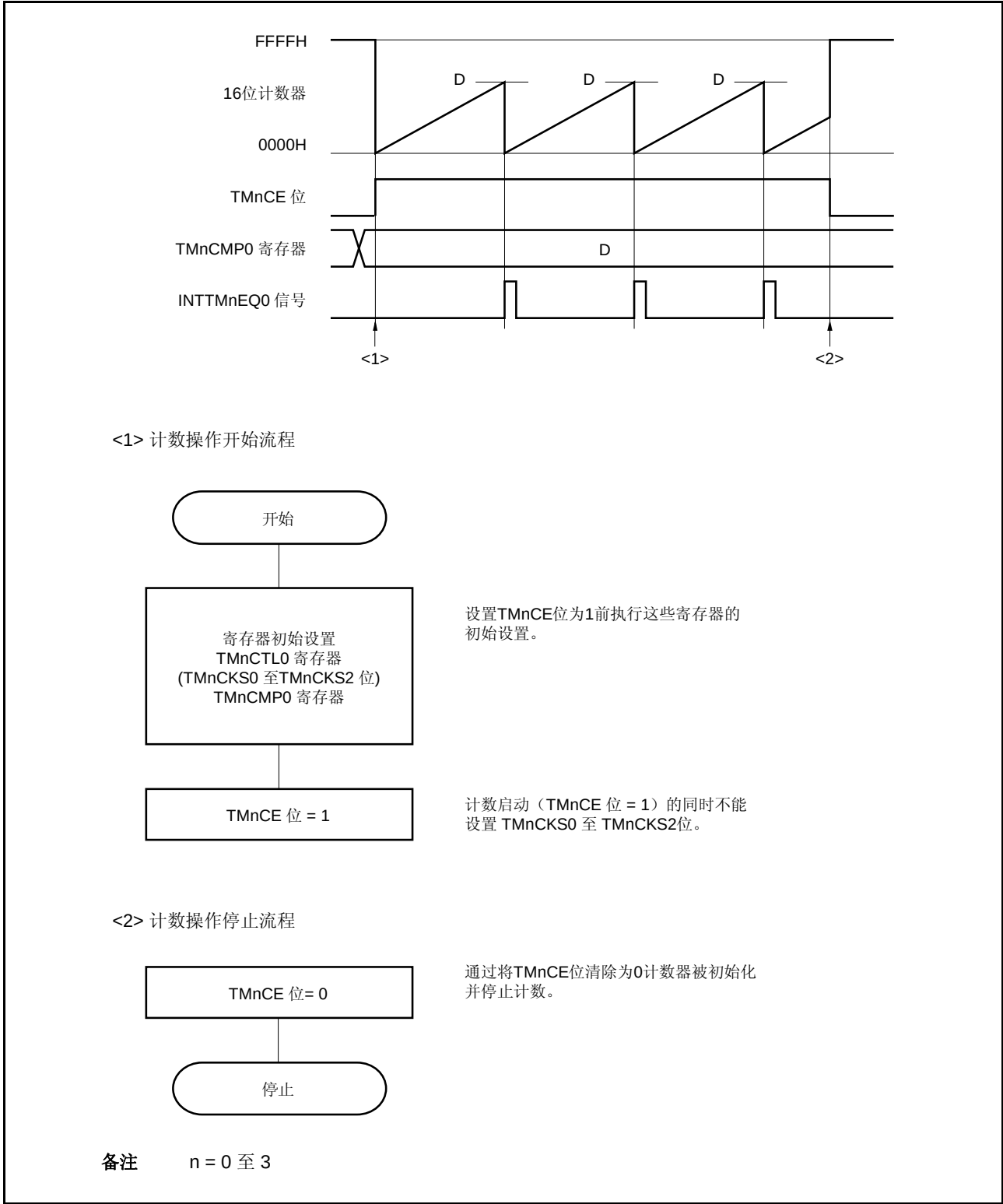
时间间隔 = (TMnCMP0 寄存器的设定值 + 1) × 计数时钟周期

图 10-4. 间隔定时器模式操作时的寄存器设置



(1) 间隔定时器模式工作流程

图 10-5. 间隔定时器模式下的软件处理流程

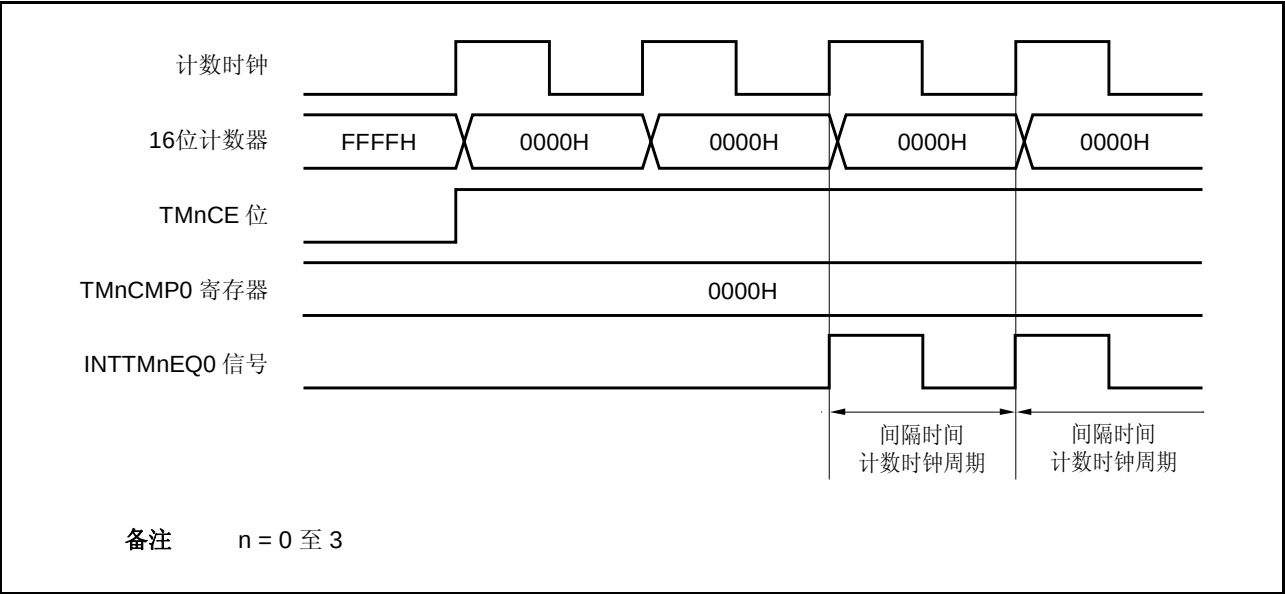


(2) 间隔定时器模式工作时序

注意事项 不要将 TMnCMP0 寄存器设置为 FFFFH。

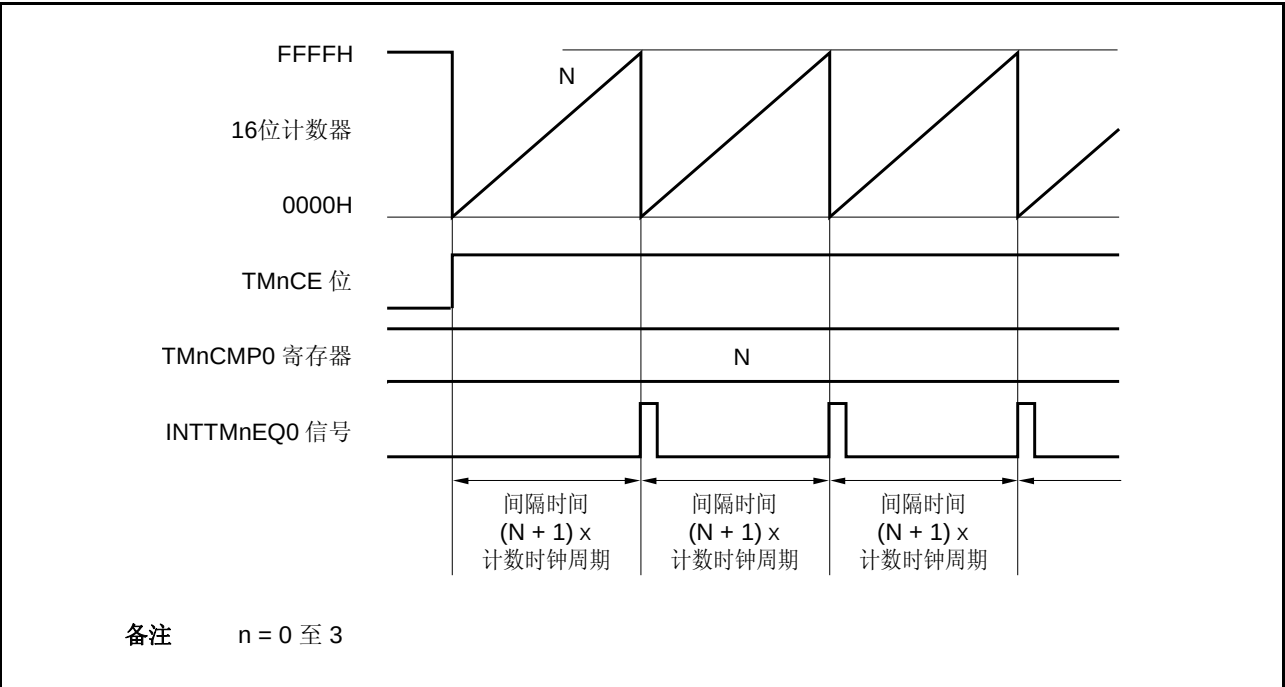
(a) TMnCMP0 寄存器设置为 0000H 时的操作

如果 TMnCMP0 寄存器被设置为 0000H，那么每个计数时钟都会产生 INTTMnEQ0 信号。
16 位计数器的值总是 0000H。



(b) TMnCMP0 寄存器被设置为 N 时的操作

如果 TMnCMP0 寄存器设置为 N，则 16 位计数器计数最大至 N。计数器和下一个计数时序同步清除为 0000H 并产生 INTTMnEQ0 信号。



10.4.2 注意事项

(1) 根据选择的计数时钟，TMnCTL0.TMnCE 位设置为 1 后，需要 16 位计数器计数至如下时间后启动计数：

(n = 0)

选择的计数时钟	计数启动前的最大时间
f _{xx}	2/f _{xx}
f _{xx} /2	3/f _{xx}
f _{xx} /4	6/f _{xx}
f _{xx} /64	128/f _{xx}
f _{xx} /512	1024/f _{xx}
f _{xx} /1024	2048/f _{xx}
f _R /8	16/f _R
f _{XT}	2/f _{XT}

(n = 1 至 3)

选择的计数时钟	计数启动前的最大时间
f _{xx} /2	4/f _{xx}
f _{xx} /4	6/f _{xx}
f _{xx} /8	12/f _{xx}
f _{xx} 16	32/f _{xx}
f _{xx} /64	128/f _{xx}
f _{xx} /256	512/f _{xx}
f _{xx} /512	1024/f _{xx}
f _{xx} /1024	2048/f _{xx}

(2) TMMn 工作时，禁止重写 TMnCMP0 和 TMnCTL0 寄存器的值。

如果在 TMnCE 位 = 1 时对这两个寄存器进行重写，那么操作结果将得不到保证。

如果这些寄存器被错误地改写，先将 TMnCTL0.TMnCE 位清零，然后重置这些寄存器

备注 n = 0 至 3

第十一章 马达控制功能

11.1 功能概述

定时器 AB1 (TAB1) 和 TMQ0 选项 (TMQOP0) 可用作控制马达的逆变器功能。它可以使用定时器 AA4 (TAA4) 来执行联合功能, 并且当 TAB1 的值与 TAA4 的值匹配时, 可以启动 A/D 转换器的 A/D 转换。以下操作可以用于实现马达控制功能。

- 带有 16 位精度的 6 相位 PWM 输出功能。
- 定时器联合操作功能 (使用 TAA4 时可调)。
- 周期设置功能 (在峰值或谷值中断操作期间可以改变周期)。
- 比较寄存器重写: 随时重写, 批量重写, 或间隙重写 (在 TAB1 操作过程中可选)。
- 中断及传送挑选功能。
- 死区时间设置功能。
- A/D 转换器的 A/D 触发时序功能。
- 0%输出和 100%输出可用。
- 通过峰值中断和谷值中断可以选择 0%输出和 100%输出。
- 强行输出停止功能。
 - 由外部引脚输入 (TOAB1OFF, TOAA1OFF) 检测到有效沿时。
 - 由 A/D 转换器模块的比较器功能检测到过电压时。
 - 由时钟监视器功能检测到主时钟振荡停止时。

11.2 配置

马达控制功能包含以下硬件：

项目	配置
定时器寄存器	死区时间计数器
比较寄存器	TAB1 死区时间比较寄存器（TAB1DTC 寄存器）
控制寄存器	TAB1 选项寄存器 1（TAB1OPT1） TAB1 选项寄存器 2（TAB1OPT2） TAB1 I/O 控制寄存器 3（TAB1IOC3） 高阻输出控制寄存器 0（HZA0CTL0） 高阻输出控制寄存器 1（HZA0CTL1）

- 通过使用 TAB1（TOAB11，TOAB12，TOAB13）的输出可以产生带有死区时间的 6 相位 PWM 输出。
- 6 相位 PWM 输出的输出电平可以进行单独设置。
- TAB1 的 16 位定时器 / 计数器递增/递减计数三角波。当定时器 / 计数器下溢且发生周期匹配时会生成一个中断。然而，中断生成被限制为 31 次。
- TAA4 可以与 TAB1（定时器联合操作功能）同时执行计数操作。TAA4 可以设置为三种方式，这是因为它可以产生一种 A/D 触发信号源（TABTADT0）和两种类型的中断：TAB1 向下溢出中断（INTTAB1OV）和周期匹配中断（INTTAB1CC0）。

图 11-1. 马达控制的框图

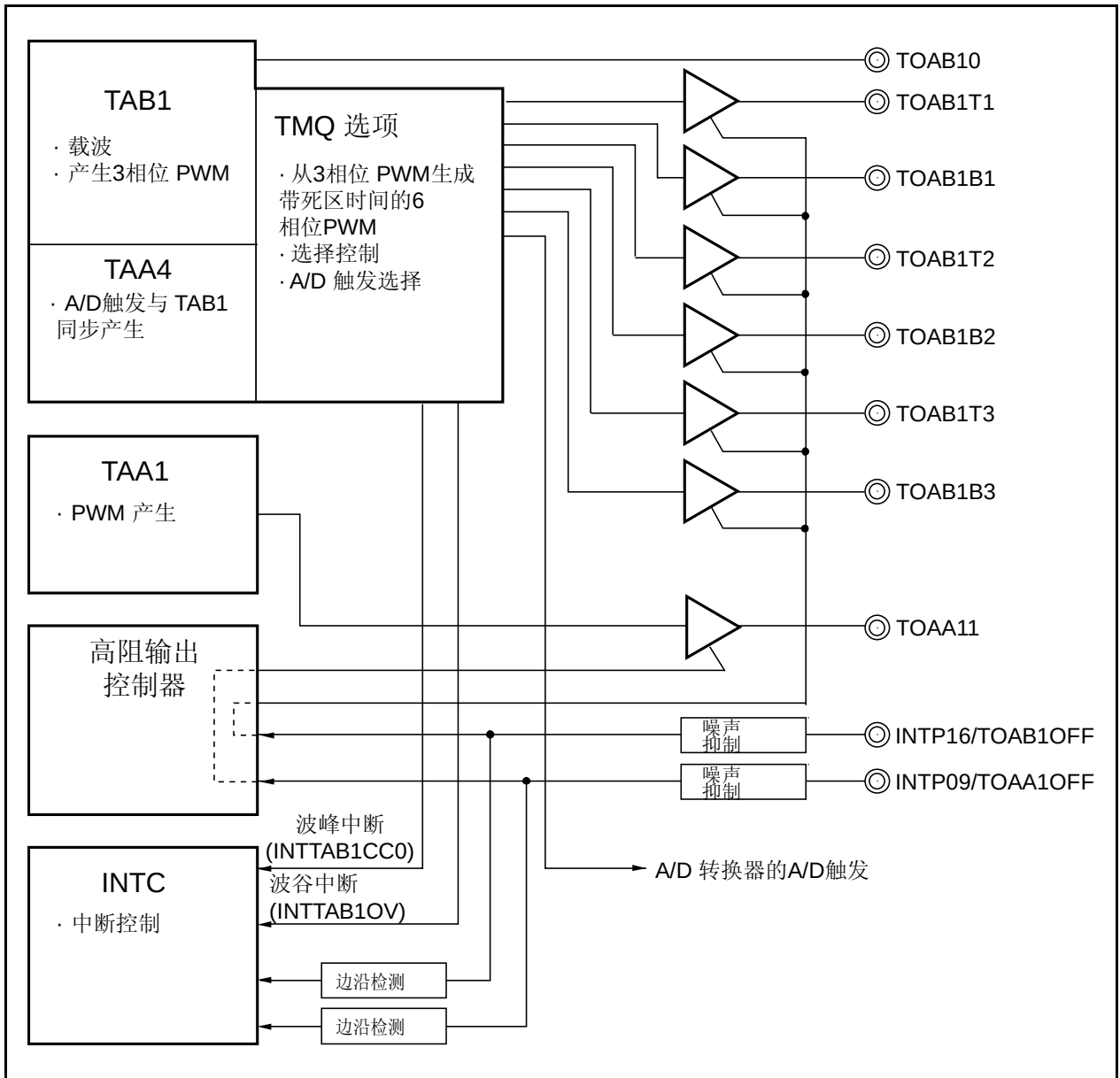
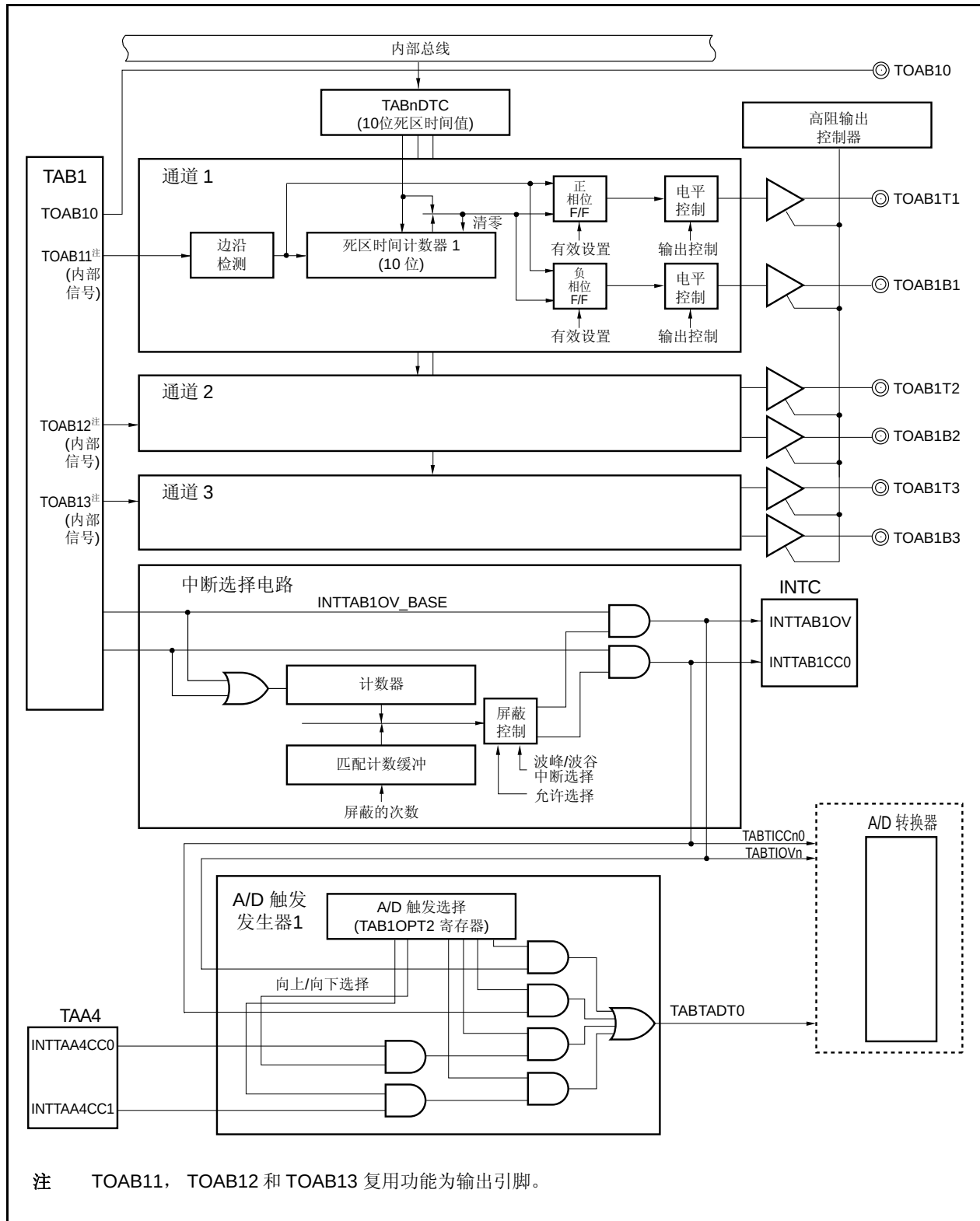


图 11-2. TMQ1 选项



(1) TAB1 死区时间比较寄存器 (TAB1DTC)

TAB1DTC 寄存器是用于指定死区时间的 10 位比较寄存器。

当 TAB1CTL0.TAB1CE 位 = 1 时禁止重写该寄存器。

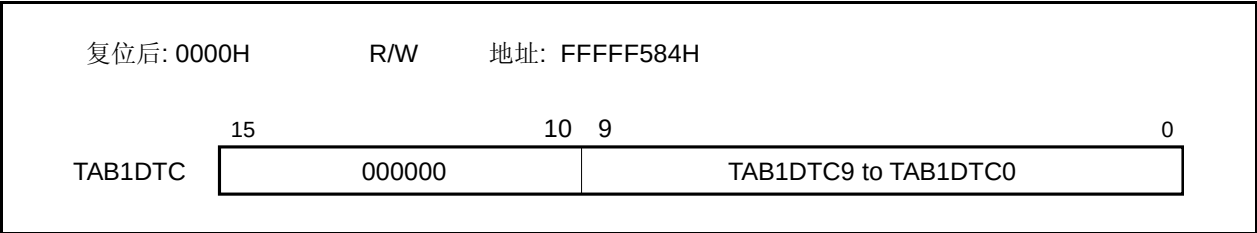
该寄存器可按 16 位为单位进行读写。

复位后本寄存器设置为 0000H。

注意事项 产生死区时间区间时，将 TAB1DTC 寄存器设置为 1 或更高的数值。

需注意，当操作停止 (TAB1CTL0.TAB1CE 位 = 0) 时，不产生死区时间区间，因此 TOAB1T1 至 TOAB1T3 以及 TOAB1B1 至 TOAB1B3 引脚的输出电平为它们的默认状态。所以，为保护系统，需采取措施例如：在停止操作前使 TOAB1T1 至 TOAB1T3 以及 TOAB1B1 至 TOAB1B3 引脚进入高阻状态，或者在转换端口模式前设置这些引脚的输出电平。

如果不需要死区时间区间，将 TAB1DTC 寄存器设置为 0。



(2) 死区时间计数器 1 至 3

死区时间计数器是用于计数死区时间的 10 位计数器。

这些寄存器会在 TOAB1m 输出信号的上升沿或下降沿处通过 TAB1 来清除或计数，并且它们会在其计数值与 TAB1DTC 寄存器的值匹配时被清除或停止。这些计数器的计数时钟同 TAB1 的 TAB1CTL0.TAB1CKS2 至 TAB1CTL0.TAB1CKS0 位所设置的计数时钟相同。

- 备注**
- 1. TAB1OPT2.TAB1DTM 位 = 1 时的操作是不同的。详细信息，请参见 11.4.2 (4) 自动死区时间宽度缩小功能 (TAB1OPT2.TAB1DTM 位 = 1)。
 - 2. m = 1 至 3

11.3 控制寄存器

(1) TAB1 选项寄存器 1 (TAB1OPT1)

TAB1OPT1 寄存器是用来控制由定时器 Q 选项功能所生成的中断请求信号的 8 位寄存器。
该寄存器可以在 TAB1CTL0.TAB1CE 位为 1 时进行重写。
根据 TAB1OPT0.TAB1CMS 位的设置有两种重写模式（批量写入模式和任意写入模式）可选择。
本寄存器可按 8 位或 1 位读写。
复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: FFFFF580H

TAB1OPT1

<7>	<6>	5	4	3	2	1	0
TAB1ICE	TAB1IOE	0	TAB1ID4	TAB1ID3	TAB1ID2	TAB1ID1	TAB1ID0

TAB1ICE	波峰中断 (INTTAB1CC0 信号) 使能
0	不使用 INTTAB1CC0 信号 (不用它作为中断选择的计数信号)。
1	使用 INTTAB1CC0 信号 (将它用作中断选择的计数信号)。

TAB1IOE	波谷中断 (INTTAB1OV 信号) 使能
0	不使用 INTTAB1OV 信号 (不用它作为中断选择的计数信号)。
1	使用 INTTAB1OV 信号 (将它用作中断选择的计数信号)。

TAB1ID4	TAB1ID3	TAB1ID2	TAB1ID1	TAB1ID0	中断屏蔽的数目
0	0	0	0	0	不选择(所有中断都输出)
0	0	0	0	1	1 个被屏蔽 (两个中断中一个被输出)
0	0	0	1	0	2 个被屏蔽 (三个中断中一个被输出)
0	0	0	1	1	3 个被屏蔽 (四个 中断中一个被输出)
:	:	:	:	:	:
1	1	1	0	0	28个被屏蔽 (29个 中断中一个被输出)
1	1	1	0	1	29 个被屏蔽 (30个 中断中一个被输出)
1	1	1	1	0	30 个被屏蔽 (31 个中断中一个被输出)
1	1	1	1	1	31 个被屏蔽 (32个 中断中一个被输出)

(2) TAB1 选项寄存器 2 (TAB1OPT2)

TAB1OPT2 寄存器是 8 位寄存器，用于控制定时器 Q 选项功能。

该寄存器可以在 TAB1CTL0.TAB1CE 位为 1 时进行重写。然而，在 TAB1CE 为 1 时禁止重写 TAB1DTM 位。相同的值可以进行重写。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

(1/2)

复位后: 00H R/W 地址: FFFFF581H

	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
TAB1OPT2	TAB1RDE	TAB1DTM	TAB1ATM3	TAB1ATM2	TAB1AT3	TAB1AT2	TAB1AT1	TAB1AT0

TAB1RDE	传送操作使能
0	不选择传送 (传送时序在每次的波峰和波谷处生成)。
1	在TAB1OPT1寄存器设置的中断选择的相同间隔选择传送。

TAB1DTM	死区时间计数器操作模式选择 (m = 1 至3)
0	死区时间计数器正常向上计数，如果TAB1的TOAB1m 输出为窄间隔(TOAB1m 输出宽度 < 死区时间宽度)，则死区时间计数器清零并重新向上计数。
1	死区时间计数器正常向上计数，如果TAB1的TOAB1m 输出为窄间隔(TOAB1m 输出宽度 < 死区时间宽度)，死区时间计数器向下计数而死区时间控制宽度自动变窄。
禁止在定时器操作期间重写 TAB1DTM 位。如果被错误的重写，通过清除TAB1CE位为0停止定时器操作并重置TAB1DTM 位。	

- 注意事项**
1. 使用中断选择时 (TAB1OPT1.TAB1ID4 至 TAB1OPT1.TAB1ID0 位设置为 00000 外的值) 时，确保将 TAB1RDE 位设置为 1。
这就意味着中断和传送可以同时生成。中断和传送不能分开设置。如果中断和传送分开设置 (TAB1RDE 位 = 0)，传送将不能正常进行。
 2. 产生死区时间区间时，将 TAB1DTC 寄存器设置为 1 或更高的数值。
需注意，当操作停止 (TAB1CTL0.TAB1CE 位 = 0) 时，不产生死区时间区间，因此 TOAB1T1 至 TOAB1T3 以及 TOAB1B1 至 TOAB1B3 引脚的输出电平为它们的默认状态。所以，为保护系统，需采取措施例如：在停止操作前使 TOAB1T1 至 TOAB1T3 以及 TOAB1B1 至 TOAB1B3 引脚进入高阻状态，或者在转换端口模式前设置这些引脚的输出电平。
如果不需要死区时间区间，将 TAB1DTC 寄存器设置为 0。

TAB1ATM3	TAB1ATM3 模式选择
0	为 INTTAA4CC1 中断输出A/D触发信号(TABTADT0) 而死区数据计数器正在向上计数。
1	为 INTTAA4CC1 中断输出A/D触发信号(TABTADT0) 而死区数据计数器正在向下计数。

TAB1ATM2	TAB1ATM2 模式选择
0	为INTTAA4CC0中断输出A/D触发信号(TABTADT0) 而死区数据计数器正在向上计数。
1	为 INTTAA4CC0 中断输出A/D触发信号(TABTADT0) 而死区数据计数器正在向下计数。

TAB1AT3 ^注	A/D 触发输出控制 3
0	禁止为 INTTAA4CC1 中断输出A/D触发信号(TABTADT0)
1	允许 为 INTTAA4CC1 中断输出A/D触发信号(TABTADT0)

TAB1AT2 ^注	A/D 触发输出控制 2
0	禁止为 INTTAA4CC0 中断输出A/D触发信号(TABTADT0)
1	允许 为 INTTAA4CC0 中断输出A/D触发信号(TABTADT0)

TAB1AT1 ^注	A/D 触发输出控制 1
0	禁止为INTTAB1CC0 (波峰中断)输出A/D触发信号(TABTADT0)
1	允许 为 INTTAB1CC0 (波峰中断)输出A/D触发信号(TABTADT0)

TAB1AT0 ^注	A/D 触发输出控制 0
0	禁止为 INTTAB1OV (波谷中断)输出A/D触发信号(TABTADT0)
1	允许为 INTTAB1OV (波谷中断)输出A/D触发信号(TABTADT0)

注 关于设置 TAB1AT3 至 TAB1AT0 位，参见**第十五章 A/D 转换器**。

(3) TAB1 I/O 控制寄存器 3 (TAB1IOC3)

TAB1IOC3 寄存器是 8 位寄存器，用于控制定时器 Q 选项功能的输出。

要从 TOAB1Tm 引脚输出，需将 TAB1IOC0.TAB1OEm 位设置为 1，然后设置 TAB1IOC3 寄存器。

只有在 TAB1CTL0.TAB1CE 位为 0 时 TAB1IOC3 寄存器才可重写。

当 TAB1CTL0.TAB1CE 位为 1 时，禁止重写 TAB1IOC3 寄存器的各位；但当 TAB1CTL0.TAB1CE 位为 1 时，可向 TAB1IOC3 寄存器的各位重写入相同的值。

本寄存器可按 8 位或 1 位读写。

复位设置将寄存器清为 A8H。

注意事项 在不同于 6 相位 PWM 输出模式下使用定时器时，应将 TAB1IOC3 寄存器设置为复位值（A8H）。

- 备注**
- 1. 使用 TAB1IOC0 寄存器设置 TOAB1Tm 引脚的输出电平。
 - 2. m = 1 至 3

复位后: A8H

R/W

地址: FFFFF582H

TAB1IOC3

<7>	<6>	<5>	<4>	<3>	<2>	1	0
TAB1OLB3	TAB1OEB3	TAB1OLB2	TAB1OEB2	TAB1OLB1	TAB1OEB1	0	0

TAB1OLBm	TOAB1Bm 引脚输出电平的设置(m = 1 至 3)
0	禁止TOAB1Bm 引脚输出反向
1	允许 TOAB1Bm 引脚输出反向

TAB1OEBm	TOAB1Bm 引脚输出 (m = 1 至 3)
0	禁止TOAB1Bm 引脚输出。 · 当 TAB1OLBm 位 = 0, 从 TOAB1Bm 引脚输出低电平。 · 当 TAB1OLBm 位 = 1, 从 TOAB1Bm 引脚输出高电平。
1	允许 TOAB1Bm 引脚输出。

(a) 从 TOAB1Tm 和 TOAB1Bm 引脚输出

TOAB1Tm 引脚输出可以通过 TAB1IOC0.TAB1OLm 和 TAB1IOC0.TAB1OEm 位来控制。TOAB1Bm 引脚输出可以通过 TAB1IOC3.TAB1OLBm 和 TAB1IOC3.TAB1OEBm 位来控制。

以下显示了 6 相位 PWM 输出模式下各种设置的定时器输出。

图 11-3. TOAB1Tm 和 TOAB1Bm 引脚的输出控制（不带死区时间）

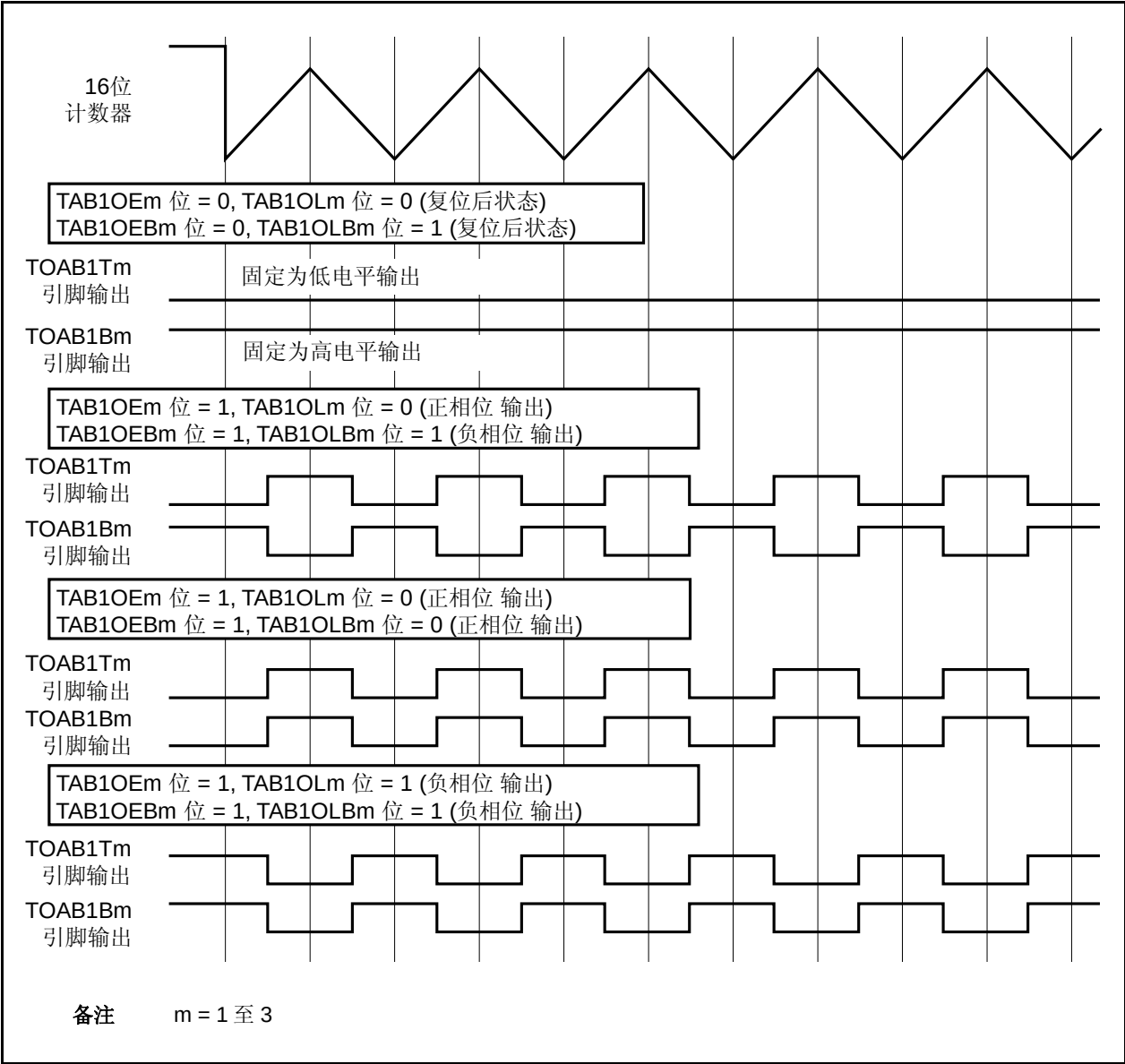


表 11-1. TOAB1Tm 引脚输出

TAB1OLm 位	TAB1OEm 位	TAB1CE 位	TOAB1Tm 引脚输出
0	0	x	低电平输出
	1	0	低电平输出
		1	TOAB1Tm 正相位输出
1	0	x	高电平输出
	1	0	高电平输出
		1	TOAB1Tm 负相位输出

备注 m = 1 至 3

表 11-2. TOAB1Bm 引脚输出

TAB1OLBm 位	TAB1OEBm 位	TAB1CE 位	TOAB1Bm 引脚输出
0	0	x	低电平输出
	1	0	低电平输出
		1	TOAB1Bm 正相位输出
1	0	x	高电平输出
	1	0	高电平输出
		1	TOAB1Bm 负相位输出

备注 m = 1 至 3

(6) 高阻输出控制寄存器 0, 1 (HZA0CTL0, HZA0CTL1)

HZA0CTL0 和 HZA0CTL1 寄存器是用于控制输出缓冲的高阻抗状态的 8 位寄存器。

这些寄存器可以按 8 位或按 1 位为单位进行读写。然而, HZA0DCF_n 位为只读位且不能被写入。

不能进行 16 位访问。

复位后这些寄存器设置为 00H。

可使用软件总向 HZA0CTL_n 寄存器写入相同值。

以下显示了检测因素和控制寄存器之间的关系。

用于高阻输出的引脚	高阻控制因素	控制寄存器
	外部引脚	
输出 TOAB1T1 至 TOAB1T3 时 输出 TOAB1B1 至 TOAB1B3 时	TOAB1OFF/INTP16	HZA0CTL0
输出 TOAA11 时	TOAA1OFF/INTP09	HZA0CTL1

注意事项 只有在目标端口指定为上表中所显示的目标引脚时才可以进行高阻控制。

备注 n = 0, 1

复位后: 00H R/W 地址: HZA0CTL0 FFFFF590H, HZA0CTL1 FFFFF591H

	<7>	<6>	5	4	<3>	<2>	1	<0>
HZA0CTLn	HZA0DCEn	HZA0DCMn	HZA0DCNn	HZA0DCPn	HZA0DCTn	HZA0DCCn	0	HZA0DCFn

(n = 0, 1)

HZA0DCEn	高阻输出控制
0	禁止高阻输出控制操作。引脚可用作输出引脚。
1	允许高阻输出控制操作。

HZA0DCMn	由HZA0DCCn位清除高阻状态的条件
0	HZA0DCCn 位的设置有效，不考虑外部引脚输入。
1	HZA0DCCn 位的设置无效而外部引脚输入保持检测为异常的电平(有效电平)。

当 HZA0DCEn 位 = 0时重写HZA0DCMn位。

HZA0DCNn	HZA0DCPn	外部引脚输入边沿说明
0	0	没有有效沿(禁止由外部引脚输入设置HZA0DCFn位)。
0	1	外部引脚的上升沿有效(由上升沿输入检测到异常)。
1	0	外部引脚的下降沿有效(由下降沿输入检测到异常)。
1	1	禁止设置

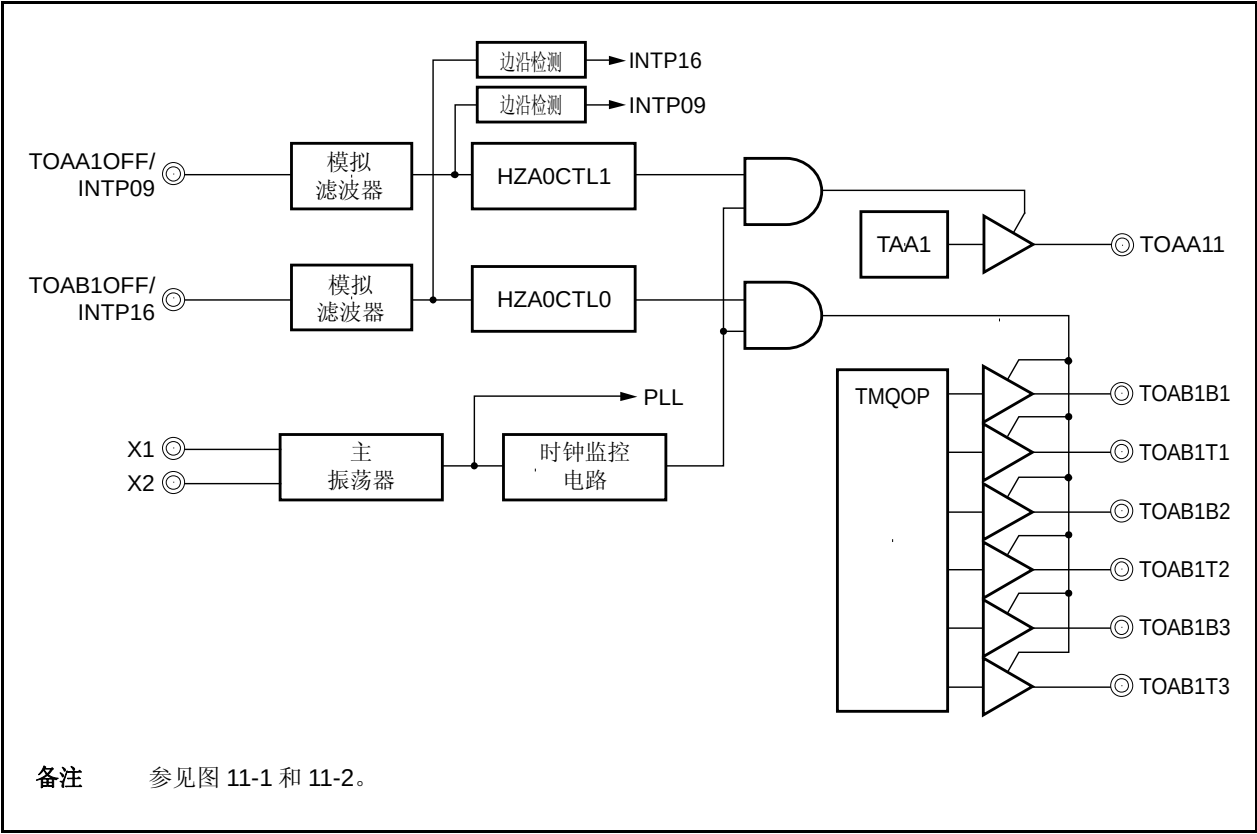
- 当 HZA0DCEn 位为 0时重写HZA0DCNn 和 HZA0DCPn位。
- 关于INTP09和 INTP16引脚中断的有效沿说明，参见23.6.2 (3)
外部中断下降、上升沿说明寄存器 3 (INTR3, INTF3) 和 (6)
外部中断下降、上升沿说明寄存器9H (INTR9, INTF9)。
- 对于外部引脚的边沿说明，从 TOAB1OFF 和TOAA1OFF 引脚开始。
然后，进行除 TOAB1OFF 和 TOAA1OFF引脚的外部引脚边沿说明。
否则，当说明TOAB1OFF和TOAA1OFFFO引脚的边沿时可能检测到一个不确定边沿。
- 操作使能后（通过将HZA0DCEn位设置为1），当输入有效沿时执行高阻输出控制。因而，当操作使能时外部引脚处于有效电平，不执行高阻输出控制。

HZA0DCTn	高阻输出触发位
0	未操作
1	由软件将引脚设置为高阻状态且 HZA0DCFn位设置为 1。
· 如果一个表示异常的边沿输入到外部引脚 (根据 HZA0DCNn和HZA0DCPn 位的设置进行检测), 即使HZA0DCTn位设置为1该位也无效。 · 因为HZA0DCTn 位是软件触发位, 所以读取时该位总为0。 · 当 HZA0DCEn 位 = 0时, 即使HZA0DCTn位设置为1此位也无效。 · 禁止同时设置 HZA0DCTn和 HZA0DCCn 位为 1 。	

HZA0DCCn	高阻输出控制清零位
0	未操作
1	进入高阻状态的引脚由软件允许输出且 HZA0DCFn位清除为 0。
· 当 HZA0DCM 位 = 0时, 这些引脚可作为输出引脚, 而不考虑外部引脚的状态。 · 当HZA0DCM 位 = 1时如果一个表示异常的边沿输入到外部引脚 (由HZA0DCNn 和 HZA0DCPn 位设置), 即使 HZA0DCCn设置为1该位也无效。 · 读取时 HZA0DCCn 位总为0。 · 当HZA0DCEn 位 = 0时, 即使 HZA0DCCn 位设置为1此位也无效。 · 禁止同时设置 HZA0DCTn和 HZA0DCCn 位为1。	

HZA0DCFn	高阻输出状态标志
清零 (0)	表示允许该引脚输出。 · 当HZA0DCEn 位 = 0时该位清除为0。 · 当HZA0DCEn 位 = 1时该位清除为0。
置位 (1)	表示引脚进入高阻状态。 · 当HZA0DCTn 位 = 1时该位设置为1。 · 当表示异常的边沿输入到外部引脚(根据HZA0DCNn 和 HZA0DCPn 位的设置进行检测)时该位设置为1。

图 11-4. 高阻抗输出控制器的配置



(a) 设置步骤:**(i) 高阻抗控制操作的设置**

<1> 设置 HZA0DCMn, HZA0DCNn 和 HZA0DCPn 位。

<2> 设置 HZA0DCEn 位为 1 (允许高阻抗控制)。

(ii) 允许高阻抗控制操作后更改设置

<1> 将 HZA0DCEn 位清零 (用来停止高阻抗控制操作)。

<2> 更改 HZA0DCMn, HZA0DCNn 和 HZA0DCPn 位的设置。

<3> 将 HZA0DCEn 位设为 1 (用来再次允许高阻抗控制操作)。

(iii) 引脚处于高阻抗状态时恢复输出

如果 HZA0DCMn 位为 1, 则在检测到外部引脚的有效沿后将 HZA0DCCn 位设为 1 以清除高阻抗状态。然而, 除非在外部引脚的输入电平为非激活状态时设置该位, 否则高阻抗状态将不能被清除。

<1> 将 HZA0DCCn 位设为 1 (命令信号清除高阻抗状态)。

<2> 读取 HZA0DCFn 位并检查标志状态。

<3> 如果 HZA0DCFn 位为 1 则返回<1>。且必须检查外部引脚的输入电平。

如果 HZA0DCFn 位为 0, 引脚可以用作输出引脚。

(iv) 用软件使引脚进入高阻状态

外部引脚的输入电平为非激活状态时必须通过软件将 HZA0DCTn 位设为 1 以使引脚进入高阻抗状态。

示例中步骤如下所示, 其中的设置独立于 HZA0DCMn 位的设置。

<1> 将 HZA0DCTn 位设为 1 (高阻抗输出命令)。

<2> 读取 HZA0DCFn 位并检查标志状态。

<3> 如果 HZA0DCFn 位为 0 则返回<1>。且必须检查外部引脚的输入电平。

如果 HZA0DCFn 位为 1, 那么引脚将处于高阻抗状态。

然而, 如果没有使用外部引脚且 HZA0DCPn 和 HZA0DCNn 位被清为 0, 那么在 HZA0DCTn 位被设为 1 时引脚将进入高阻抗状态。

备注 n = 0, 1

11.4 操作

11.4.1 系统概述

(1) 6 相位 PWM 输出概述

通过结合使用 AB1 (TAB1) 和 TMQ 选项 (TMQOPA)，6 相位 PWM 输出模式可以用于生成一个 6 相位 PWM 输出波形。

通过将 TAB1 的 TAB1CTL1.TAB1MD2 至 TAB1CTL1.TAB1MD0 位设为“111”可以使能 6 相位 PWM 输出模式。

TAB1 中的一个 16 位计数器和四个 16 位比较寄存器用于生成一个基本的 3 相位波形。

比较寄存器的功能如下：

TAA4 可与 TAB1 执行协调操作，为 A/D 转换器生成转换触发信号源。

比较寄存器	功能	设置范围
TAB1CCR0 寄存器	设置周期	$0002H \leq m \leq FFFE H$
TAB1CCR1 寄存器	指定相位 U 的输出宽度	$0000H \leq i \leq m + 1$
TAB1CCR2 寄存器	指定相位 V 的输出宽度	$0000H \leq j \leq m + 1$
TAB1CCR3 寄存器	指定相位 W 的输出宽度	$0000H \leq k \leq m + 1$

备注 m = AB1CCR0 寄存器的设定值
 i = TAB1CCR1 寄存器的设定值
 j = TAB1CCR2 寄存器的设定值
 k = TAB1CCR3 寄存器的设定值

死区时间间隔从基本 3 相位波形中生成，该波形通过使用三个 10 位死区时间计数器和一个用于创建与基本 3 相位波形相位相反的波形的比较寄存器来生成。接着会生成一个 6 相位 PWM 输出波形 (U, $\bar{U}$, V, $\bar{V}$, W 以及 $\bar{W}$)。

用于生成基本 3 相位波形的 16 位计数器向上计数或向下计数。操作开始后，该寄存器将开始向上计数。当它的计数值与 TAB1CCR0 寄存器设置的周期匹配时，计数器将开始向下计数。当计数值与 0001H 匹配时，计数器将再次开始向上计数。这意味着载波周期等于两倍于 TAB1CCR0 寄存器中设定值+1 的值。

生成死区时间间隔的 10 位死区时间计数器 1 至 3 向上计数。因此，TAB1 死区时间比较寄存器 (TAB1DTC) 中的设定值被用作死区时间值。由于使用了三种计数器，因此死区时间可以在相位 U, V, W 中独立地生成。然而，由于只有一个指定死区时间值 (TAB1DTC) 的寄存器，因此在三种相位中使用同一个死区时间值。

图 11-5. 6 相位 PWM 输出模式的概要

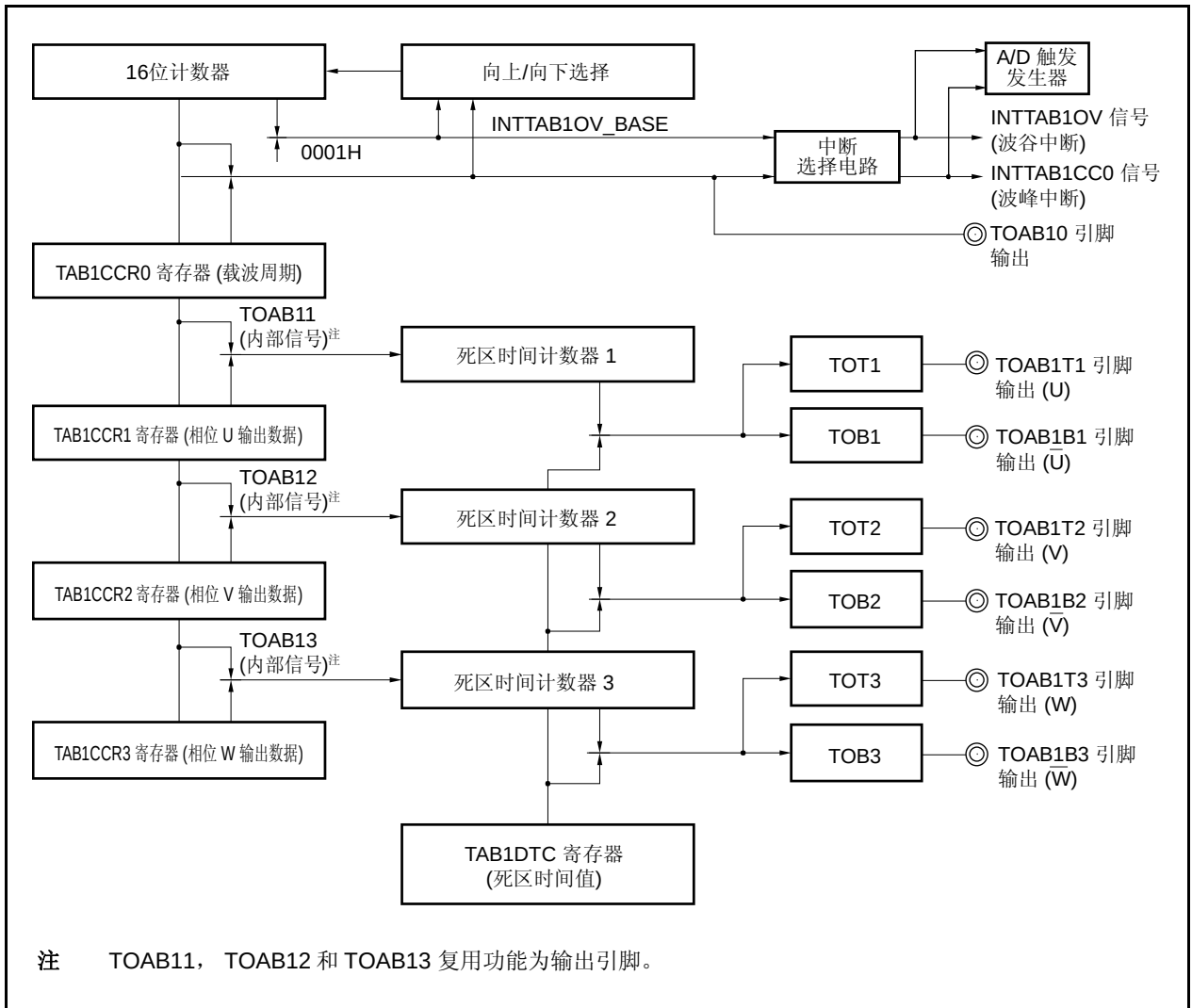
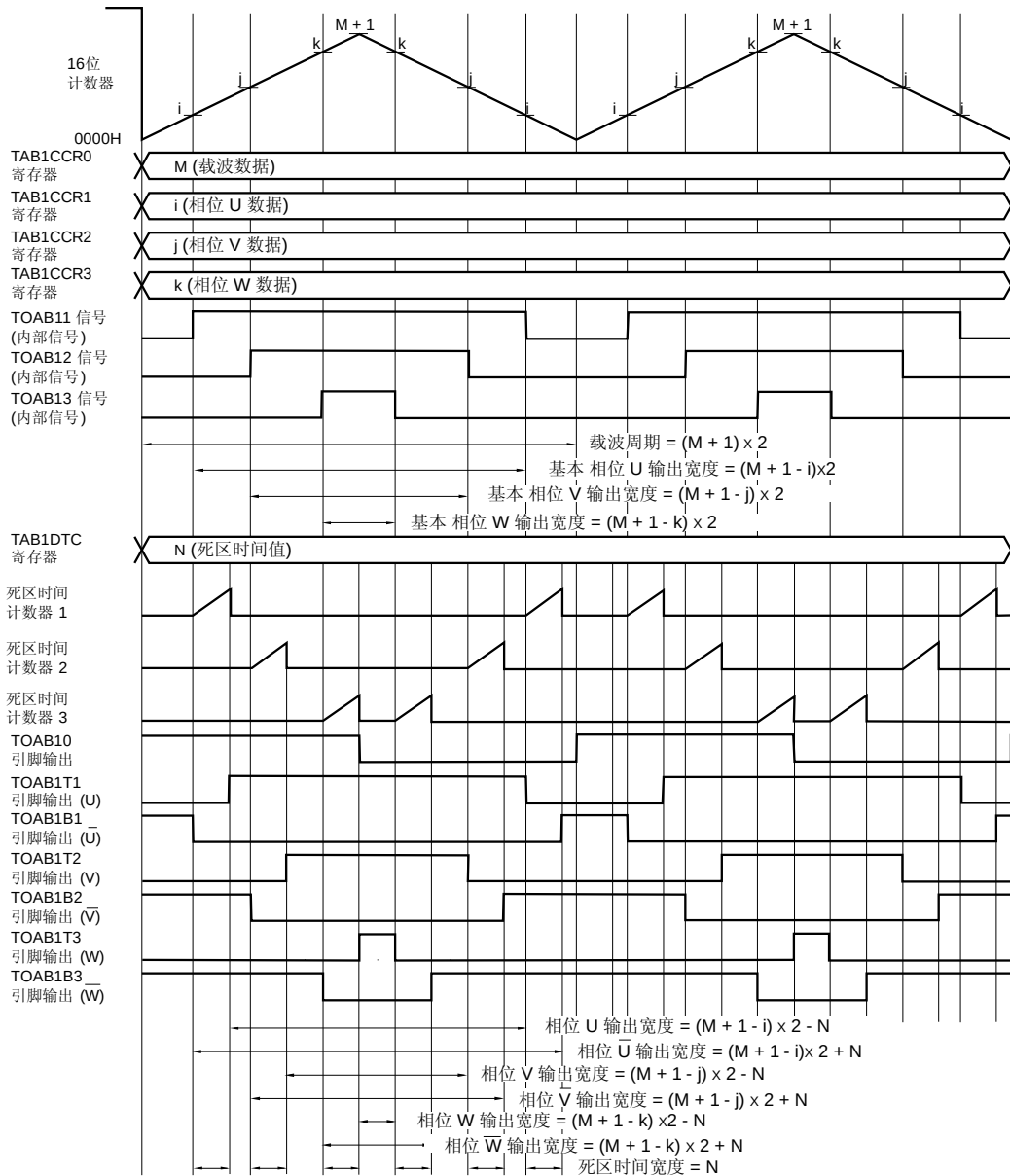


图 11-6. 6 相位 PWM 输出模式的时序图



- 注意事项
1. 在 6 相位 PWM 输出模式中，应将 TAB1CCR0 寄存器的值“M”设定在 $0002H \leq M \leq FFFE H$ 的范围内。
 2. 只有最大到“M + 1”值可以被设置到 TAB1CCR1，TAB1CCR2 以及 TAB1CCR3 寄存器中。
 3. 如果“0000H”被设置到 TAB1CCR1，TAB1CCR2 以及 TAB1CCR3 寄存器中，将会输出 100%。
如果“M + 1”被设置到 TAB1CCR1，TAB1CCR2 以及 TAB1CCR3 寄存器中，将会输出 0%。
如果“M + 2”或更高的值被设置到 TAB1CCR1，TAB1CCR2 以及 TAB1CCR3 寄存器中，输出（50%占空比）会在 16 位计数器的峰值（M + 1）处上升，在谷值（0000H）处下降。
 4. 如果相位 U，V，W 的输出宽度的等式（例如 $(M + 1 - i) \times 2 - N$ ）的操作值为 0 或低于 0，那么它将会收敛为 0（100% 输出）。如果操作值高于 $(M + 1) \times 2$ ，它将会收敛为 $(M + 1) \times 2$ （0% 输出）。

(2) 中断请求

两种类型的中断请求可用：INTTAB1CC0（波峰中断）信号和 INTTAB1OV（波谷中断）信号。

INTTAB1CC0 和 INTTAB1OV 信号可以使用 TAB1OPT1 寄存器来选择。

关于选择中断的详细信息，请参见 **11.4.3 中断选择功能**。

- INTTAB1CC0（波峰中断）信号：显示用于向上计数的 16 位计数器的值与 TAB1CCR0 寄存器的值间匹配的中断信号
- INTTAB1OV（波谷中断）信号：显示用于向下计数的 16 位计数器的值与 0001H 值间匹配的中断信号

(3) 定时器操作过程中重写寄存器

以下寄存器具有一个缓冲寄存器并可以在随时重写模式，批量重写模式或间隙批量重写模式中进行重写。

相关元件	寄存器
定时器 AA1	TAA1 捕获/比较寄存器 0（TAA1CCR0） TAA1 捕获/比较寄存器 1（TAA1CCR1）
定时器 AB1	TAB1 捕获/比较寄存器 0（TAB1CCR0） TAB1 捕获/比较寄存器 1（TAB1CCR1） TAB1 捕获/比较寄存器 2（TAB1CCR2） TAB1 捕获/比较寄存器 3（TAB1CCR3）
定时器 Q1 选项	TAB1 选项寄存器 1（TAB1OPT1）

关于比较寄存器传送功能的详细信息，请参见 **11.4.4 使用传送功能重写寄存器的操作**。

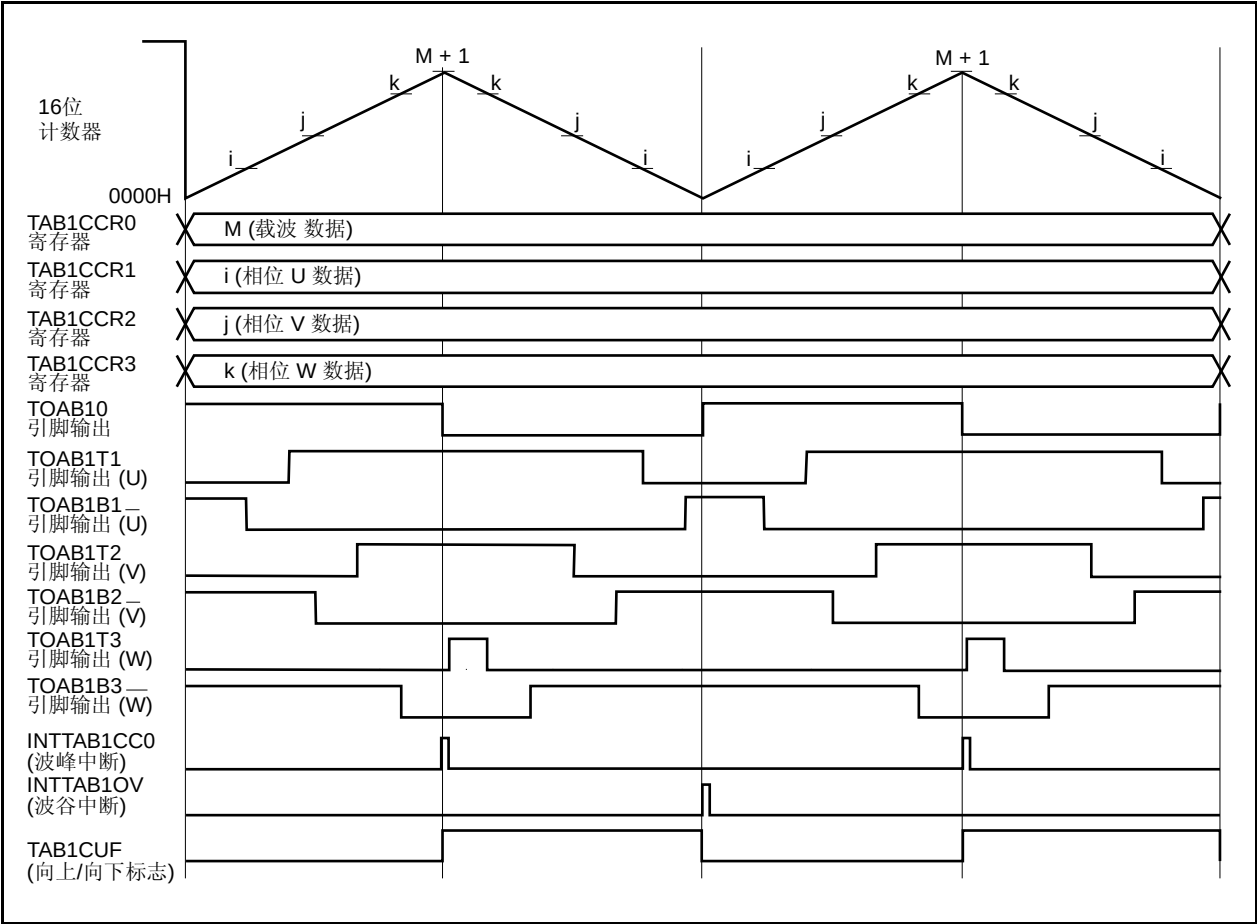
(4) 16 位计数器的向上 / 向下计数操作

通过使用 TAB1 选项寄存器 0（TAB1OPT0）的 TAB1CUF 位检查 16 位计数器的操作状态。

TAB1CUF 位的状态	16 位计数器的状态	16 位计数器值的范围
TAB1CUF 位 = 0	向上计数	0000H – m
TAB1CUF 位 = 1	向下计数	(m + 1) – 0001H

备注 m = TAB1CCR0 寄存器的设定值

图 11-7. 中断及向上 / 向下标志



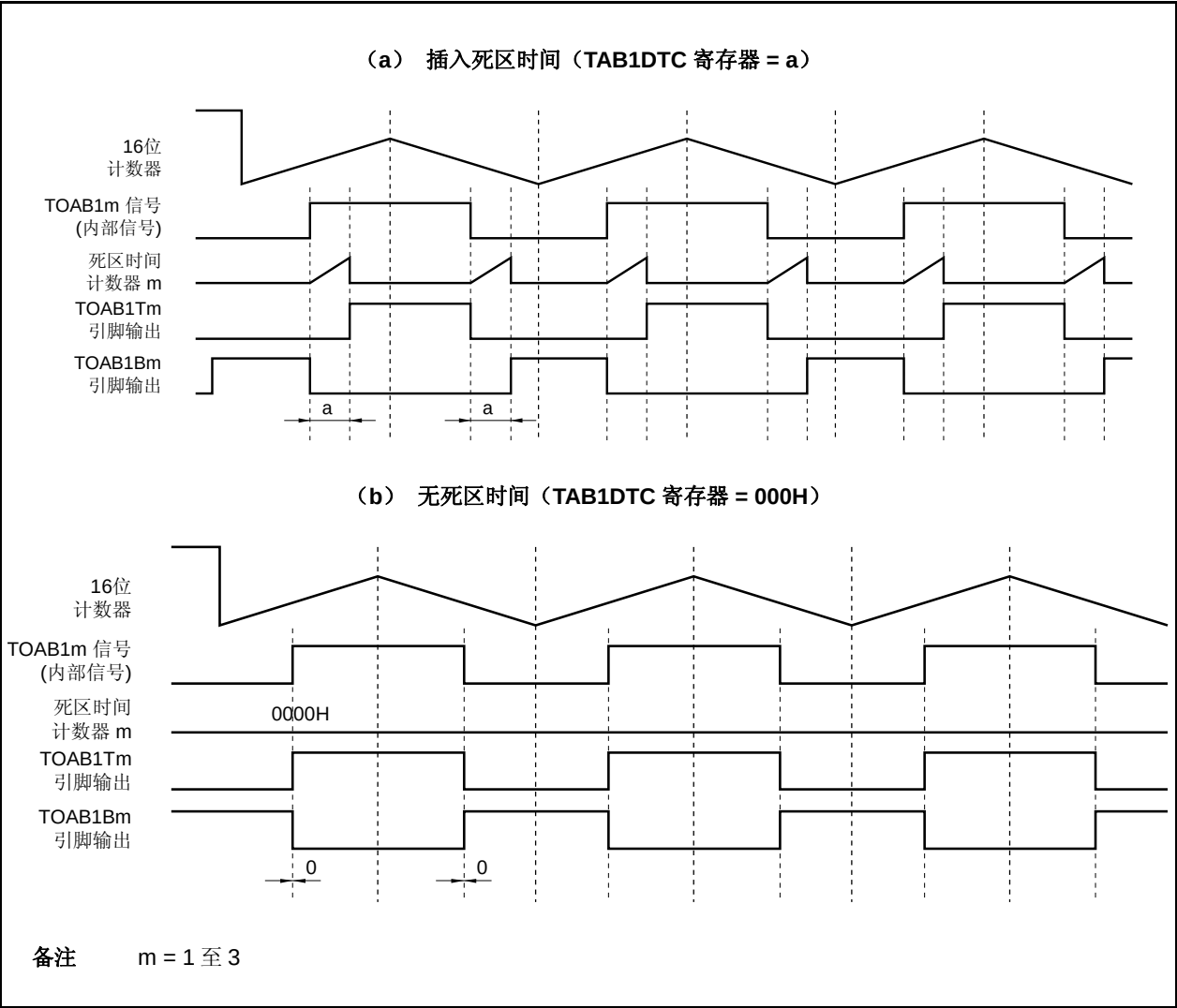
11.4.2 死区时间控制（负相位波形信号的产生）

(1) 死区时间控制原理

在 6 相位 PWM 输出模式中，比较寄存器 1 至 3（TAB1CCR1， TAB1CCR2 和 TAB1CCR3）用于设置占空比系数，而比较寄存器 0（TAB1CCR0）则用于设置周期。通过设置这四种寄存器并启动 TAB1 操作，可以生成三种带有可变占空系数的 PWM 输出波形（基本 3 相位波形）。这三种 PWM 输出波形被输入到定时器 Q 选项元件（TMQOP）中，并创建带有死区时间的反转信号以生成三组（六个）PWM 波形。

TMQOP 单元由三个 10 位计数器（死区时间计数器 1 至 3）组成，这三种计数器与 TAB1 的计数时钟以及指定死区时间的 TAB1 死区时间比较寄存器（TAB1DTC）同步操作。如果 TAB1DTC 寄存器的值被设置为“a”，那么死区时间的值为“a”且间隔“a”将会创建在正相位波形和负相位波形之间。

图 11-8. 带有死区时间的 PWM 输出波形（1）



(2) 0%/100%的 PWM 输出

V850ES/V850ES/JG3-H 和 V850ES/JH3-H 可以实现 PWM 输出的 0%波形输出和 100%波形输出。
 低电平会作为 0%波形输出从 TOAB1Tm 引脚中连续输出。高电平会作为 100%波形输出从 TOAB1Tm 引脚中连续输出。
 TAB1CCR0 寄存器 = M 时，通过将 TAB1CCRm 寄存器设为“M + 1”来输出 0%波形。
 通过将 TAB1CCRm 寄存器设为“0000H”来输出 100%波形。
 定时器进行操作时允许对 TAB1CCRm 寄存器进行重写，0%波形输出或 100%波形输出可以在发生波峰中断（INTTAB1CC0）和波谷中断（INTTAB1OV）时进行选择。

备注 m = 1 至 3

图 11-9. 0%PWM 输出波形（带有死区时间）

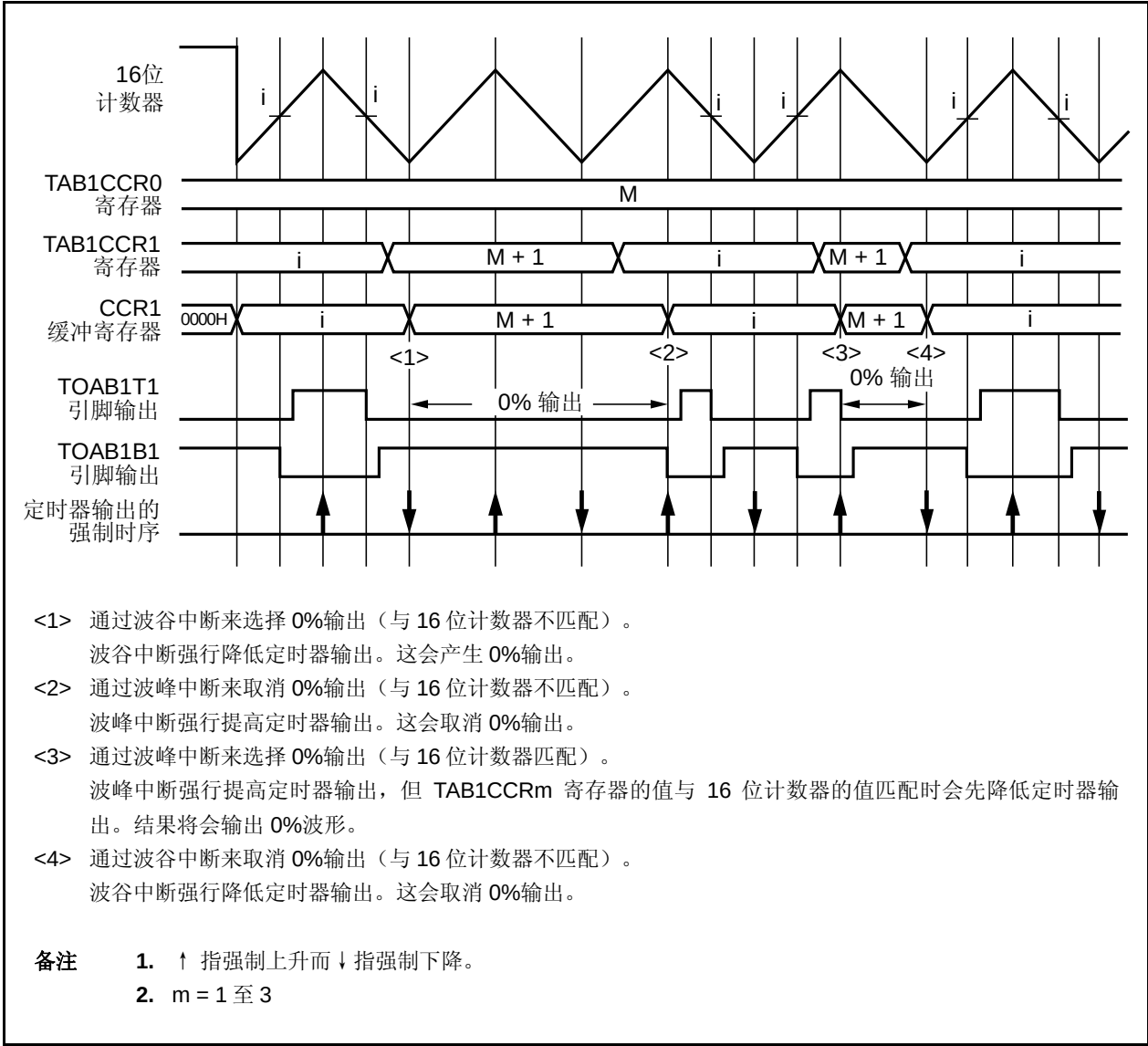


图 11-10. 100%PWM 输出波形（带有死区时间）

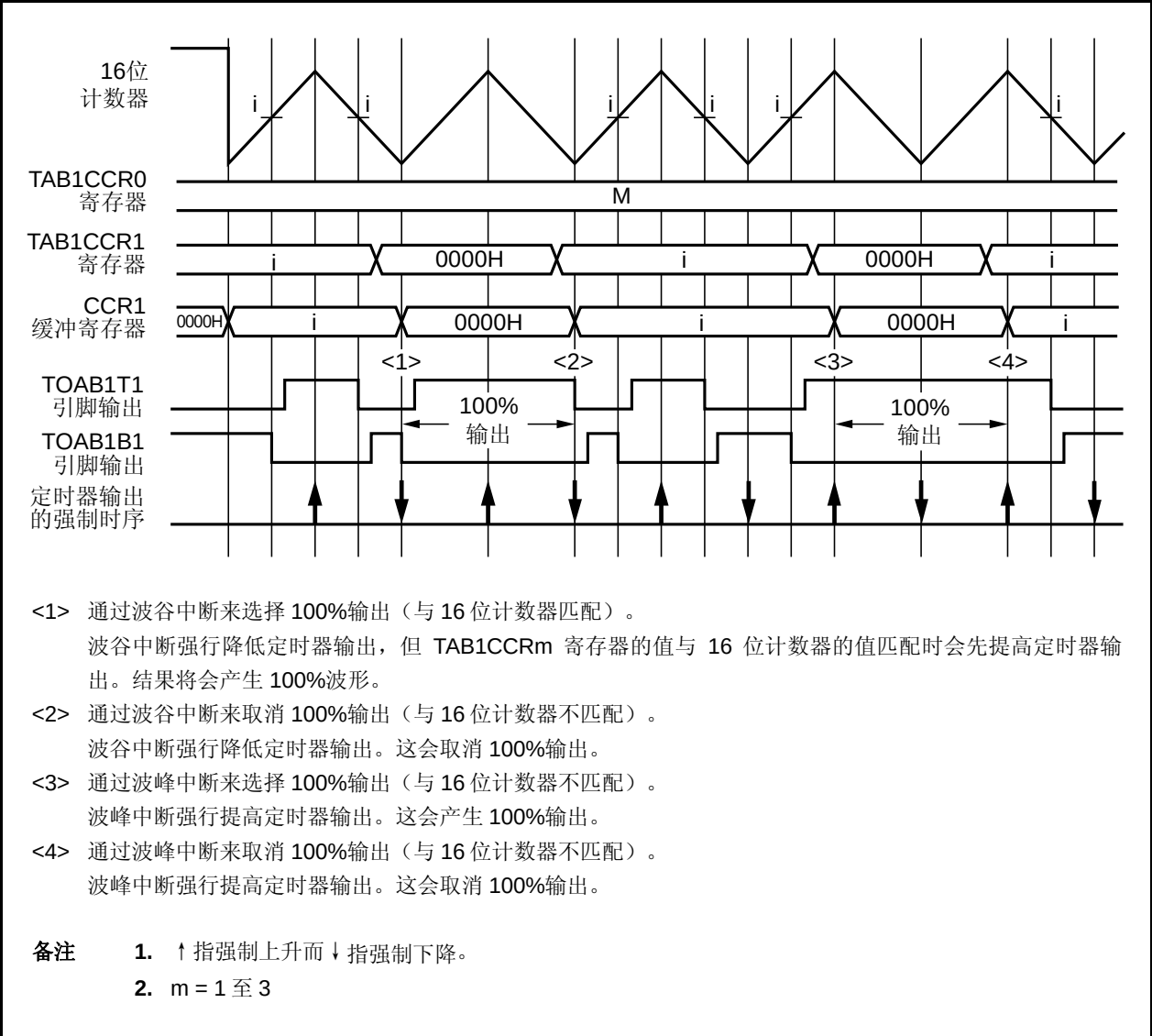
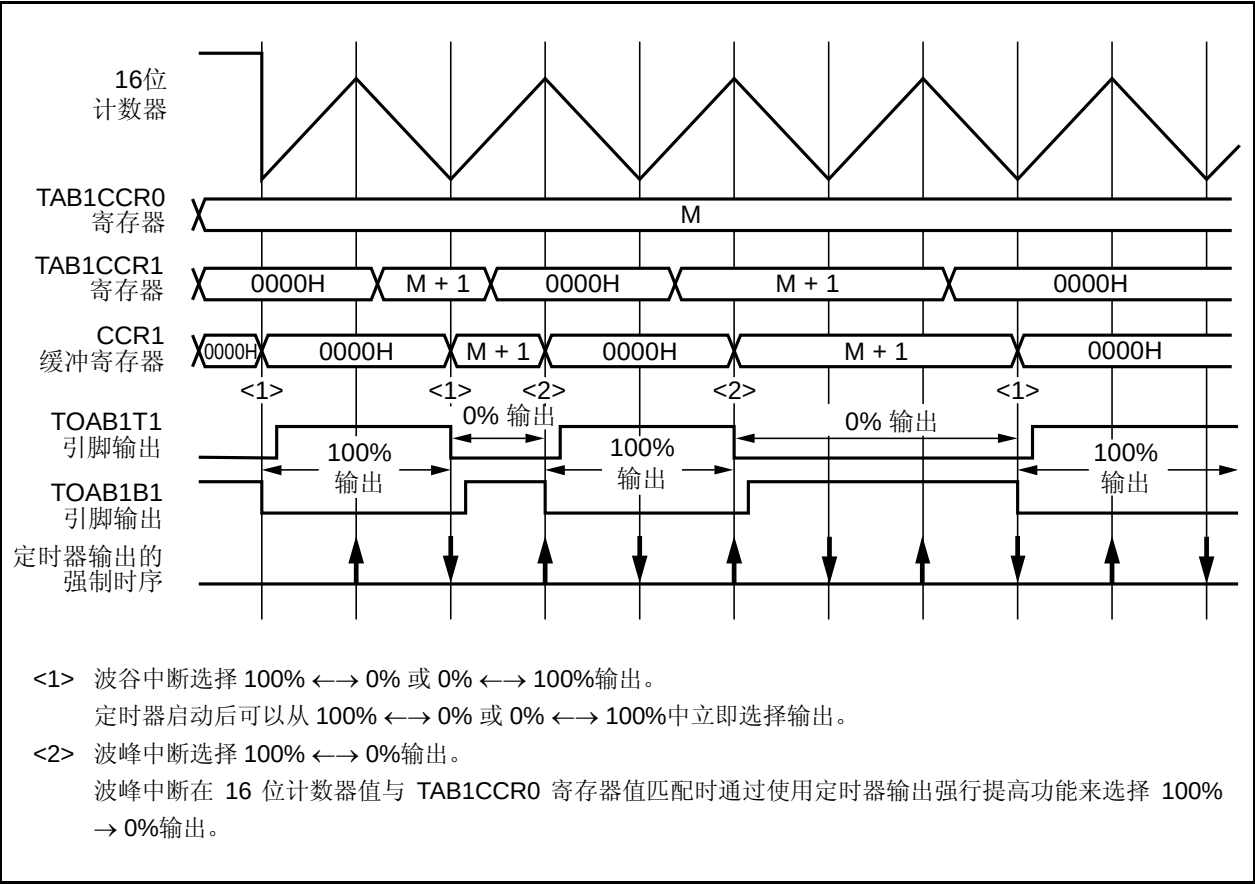


图 11-11. 从 0% 到 100%以及从 100% 到 0%的 PWM 输出波形（带有死区时间）

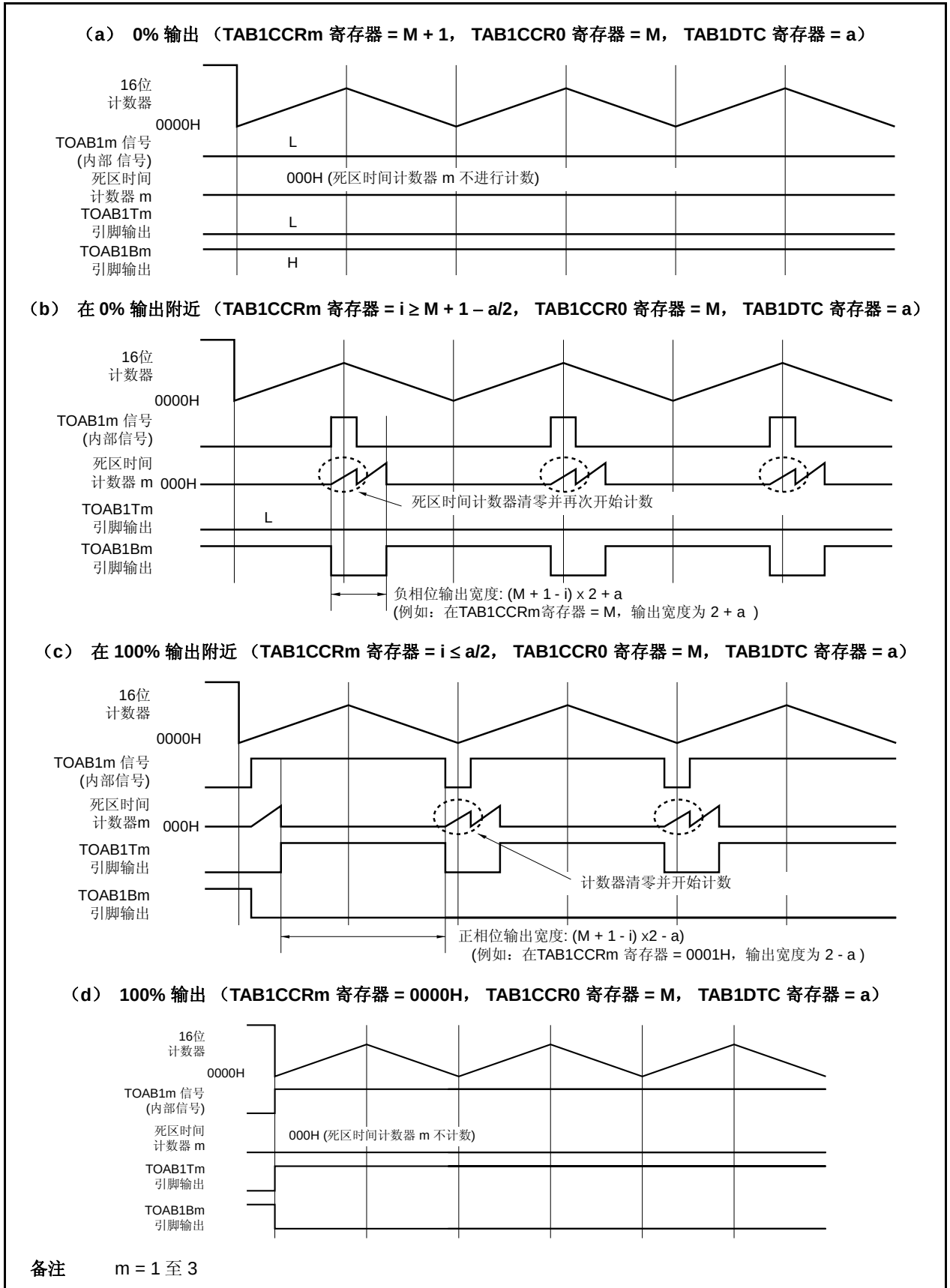


(3) 在 0% 和 100%输出临近的输出波形

如果由于 16 位计数器的值与比较寄存器的值在死区时间计数时相匹配而生成了中断，那么死区时间计数器将会被清除并再次开始计数操作。

以下显示了在 0% 和 100%输出临近的死区时间控制的输出波形。

图 11-12. 带有死区时间的 PWM 输出波形 (2)



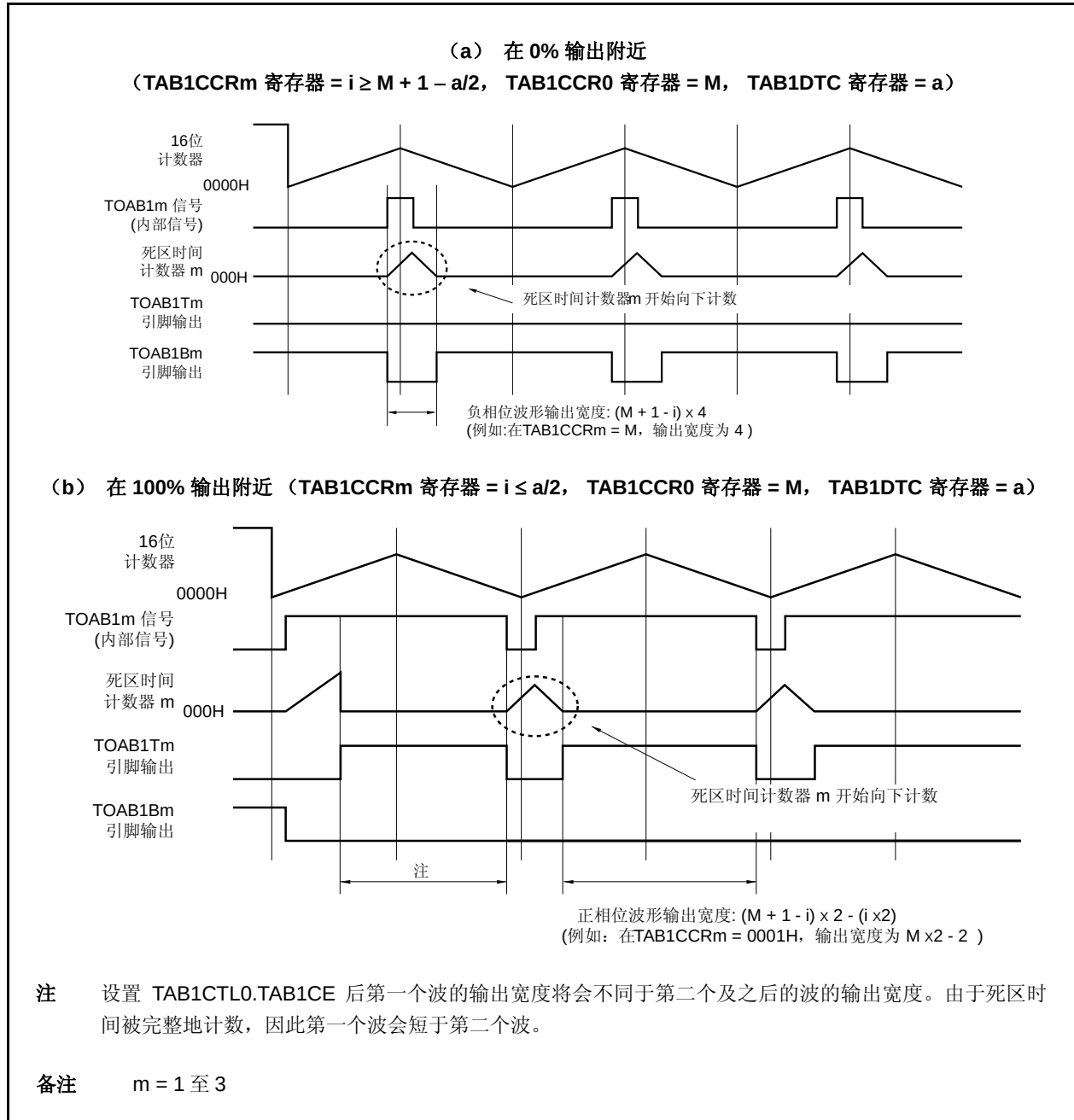
(4) 自动缩小死区时间宽度功能 (TAB1OPT2.TAB1DTM 位 = 1)

通过将 TAB1OPT2.TAB1DTM 位设为 1 可以在 0%输出或 100%输出的附近自动缩小死区时间宽度。

通过将 TAB1DTM 位设为 1 不能清除死区时间计数器，但如果在死区时间计数过程中改变定时器 AB 的 TOAB1 (内部信号) 输出，则将会开始向下计数。

以下时序图说明了 TAB1DTM 位设为 1 时死区计数器的操作。

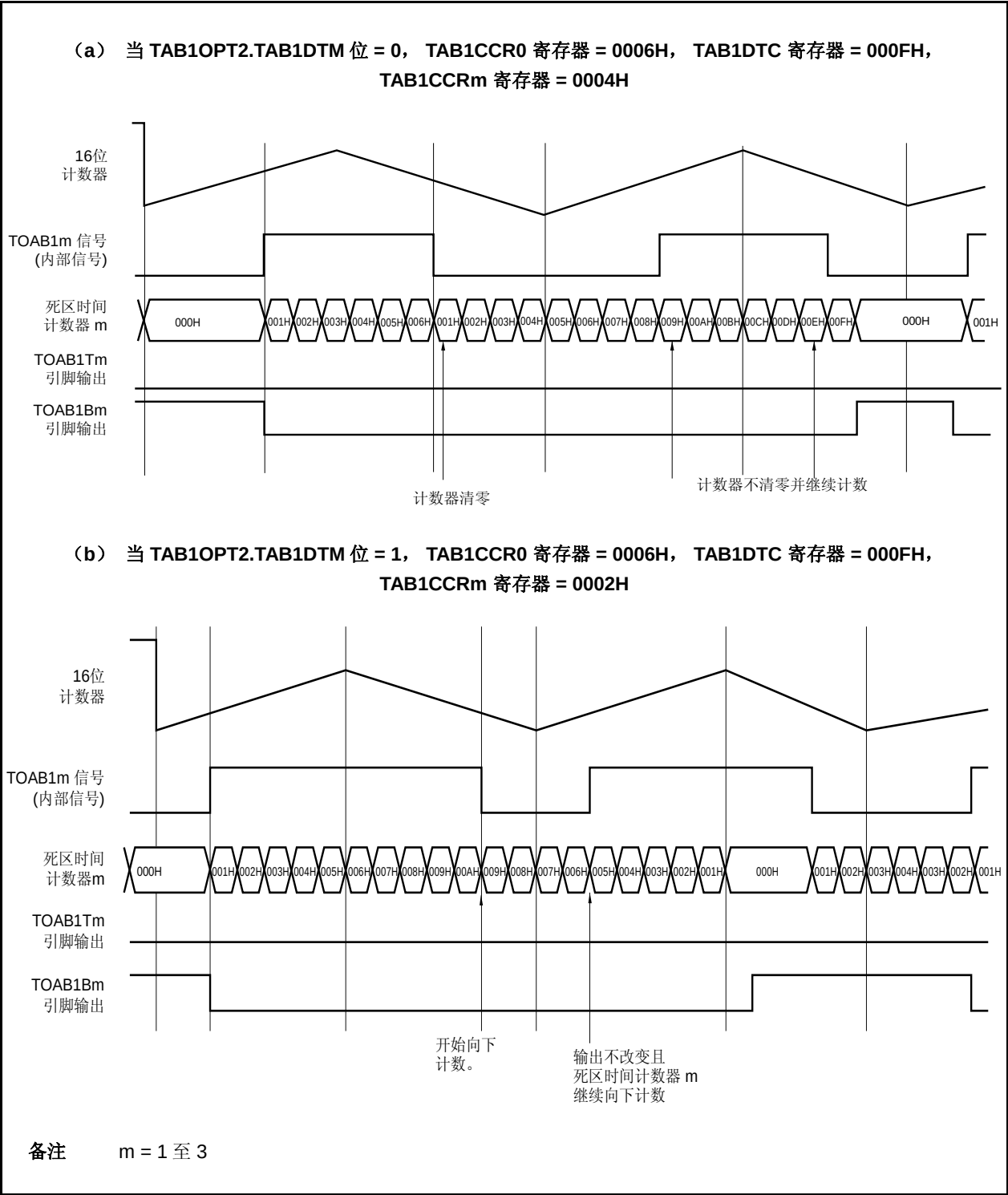
图 11-13. 死区时间计数器 m 的操作 (1)



(5) 设置不正确时死区时间的控制

通常情况下，在死区时间计数过程中，TAB1 的 TOAB1m（内部信号）输出只会在 0% 和 100%输出附近发生变化。本节介绍 TAB1CCR0 寄存器（载波周期）和 TAB1DTC 寄存器（死区时间值）没有被正确设置时的例子。如果这些寄存器没有被正确设置，那么在死区时间计数过程中，TAB1 的 TOAB1m（内部信号）输出将变化两或三次。以下的流程图显示了在这种情况下 6 相位 PWM 输出波形。

图 11-14. 死区时间计数器 m 的操作 (2)



11.4.3 中断选择功能

- 可以选择的中断为 INTTAB1CC0（波峰中断）以及 INTTAB1OV（波谷中断）。
- TAB1OPT1.TAB1ICE 位用于允许 INTTAB1CC0 中断的输出以及中断要进行选择的次数。
- TAB1OPT1.TAB1IOE 位用于允许 INTTAB1OV 中断的输出以及中断要进行选择的次数。
- TAB1OPT1.TAB1ID4 至 TAB1OPT1.TAB1ID0 位用于说明要进行选择的指定中断的计数次数。该中断在指定的计数次数期间被屏蔽而在下一个中断时序生成。
- TAB1OPT2.TAB1RDE 位用于指定否要对传送进行选择。
如果指定要对传送进行选择，那么选择后传送将会与中断输出同时进行。如果指定不对传送进行选择，那么在 TAB1CCR1 寄存器被写入后将会在传送时序进行传送。
- TAB1OPT0.TAB1CMS 位用于指定带有传送功能的寄存器是执行批量重写还是随时重写。
TAB1CMS 位为 0 时，寄存器的值会在传送的同时进行更新。当 TAB1CMS 位为 1 时，寄存器的值则会在寄存器写入一个新的值后立即更新。
从 TAB1CCRm 寄存器至 CCRm 缓冲寄存器的传送操作与中断选择时序同步进行。

注意事项 1. 在批量重写模式（传送模式）中使用中断选择功能时，将会按间隙批量重写模式（传送选择模式）来执行操作。

2. 完成选择后将会生成中断。

备注 m = 1 至 3

(1) 中断选择操作

图 11-15. TAB1OPT1.TAB1ICE 位 = 1, TAB1OPT1.TAB1IOE 位 = 1, TAB1OPT2.TAB1RDE 位 = 1 时的中断选择操作（波峰 / 波谷中断输出）

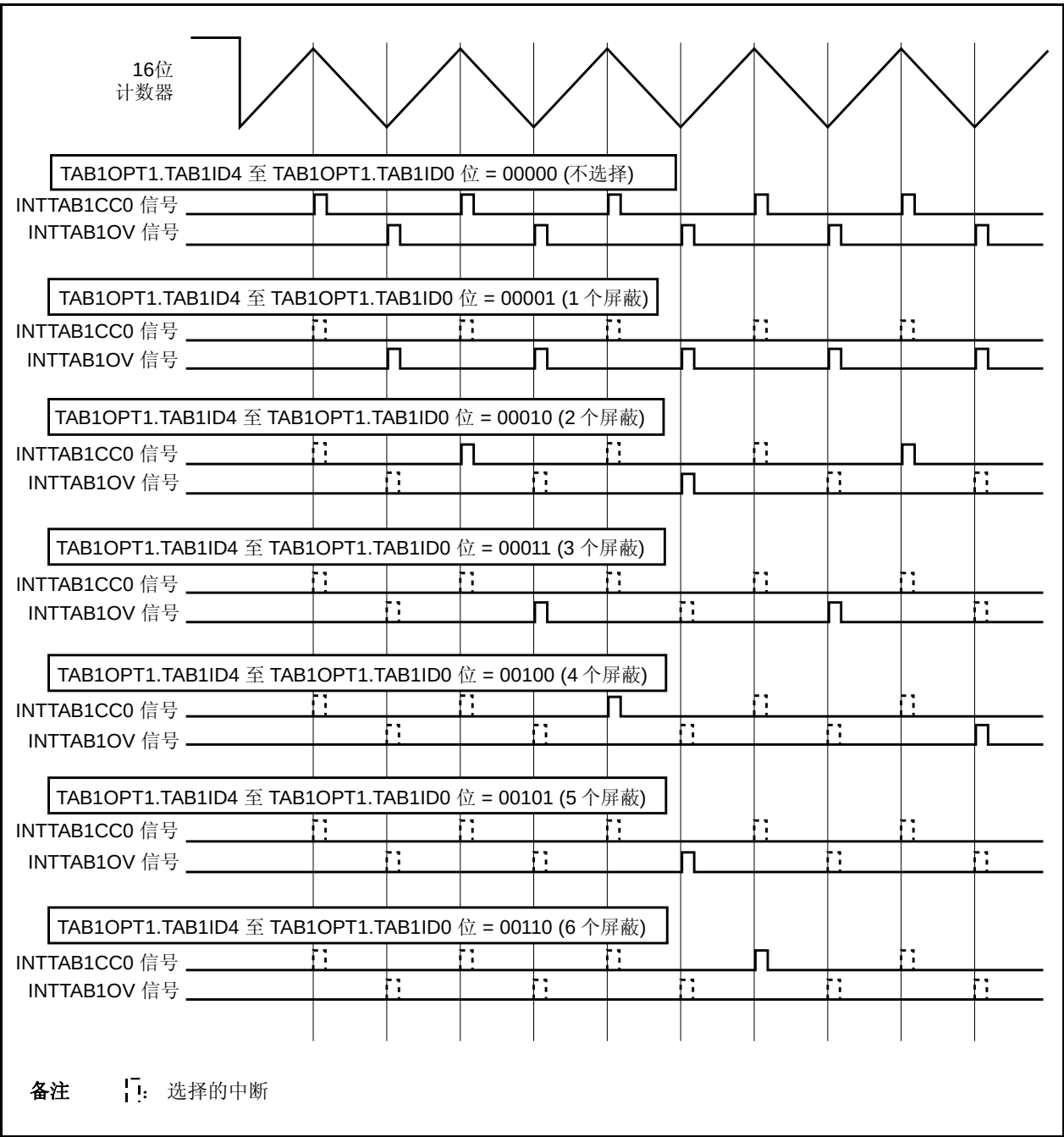


图 11-16. TAB1OPT1.TAB1ICE 位 = 1, TAB1OPT1.TAB1IOE 位 = 0, TAB1OPT2.TAB1RDE 位 = 1 时的中断选择操作（波峰中断输出）

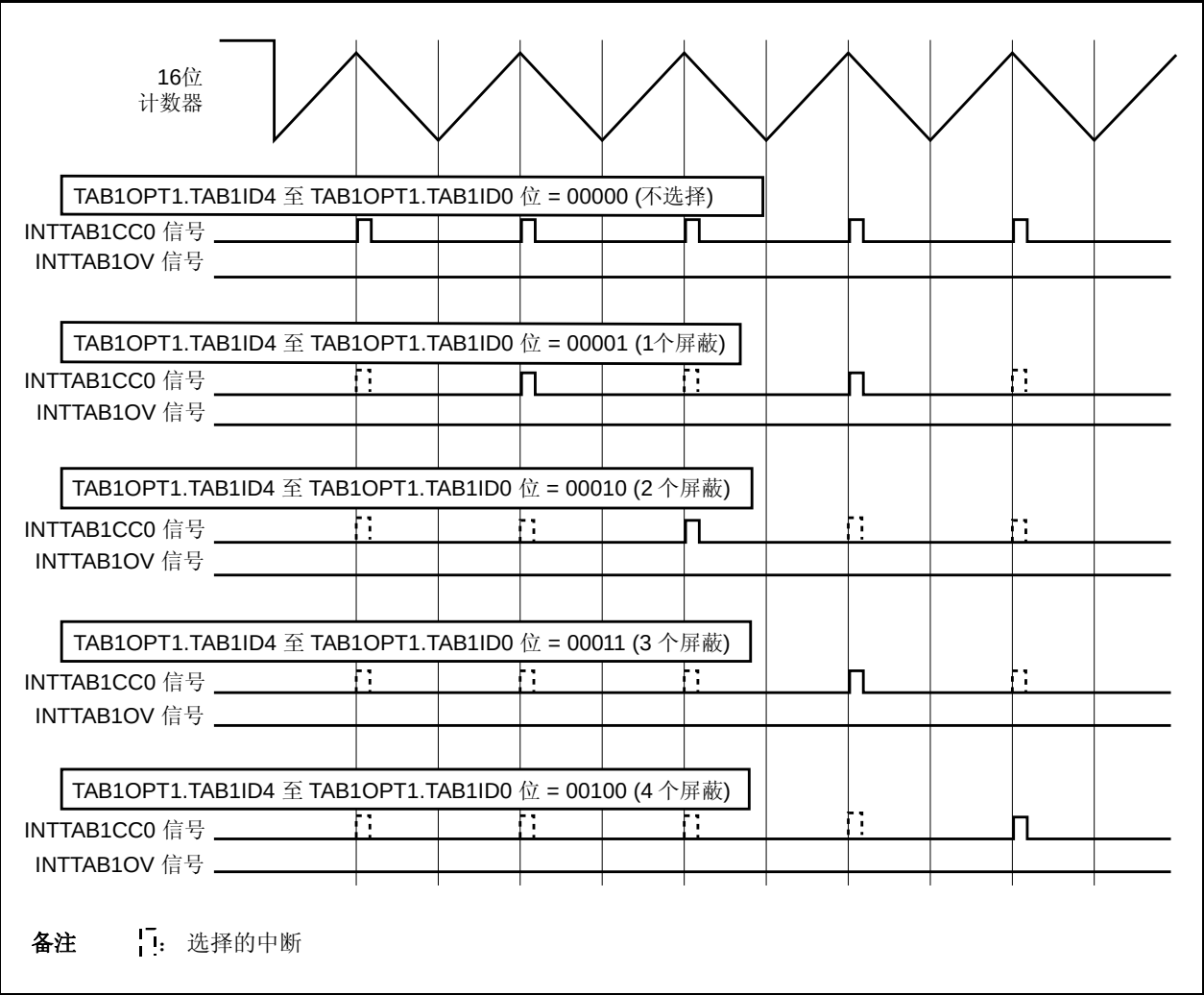
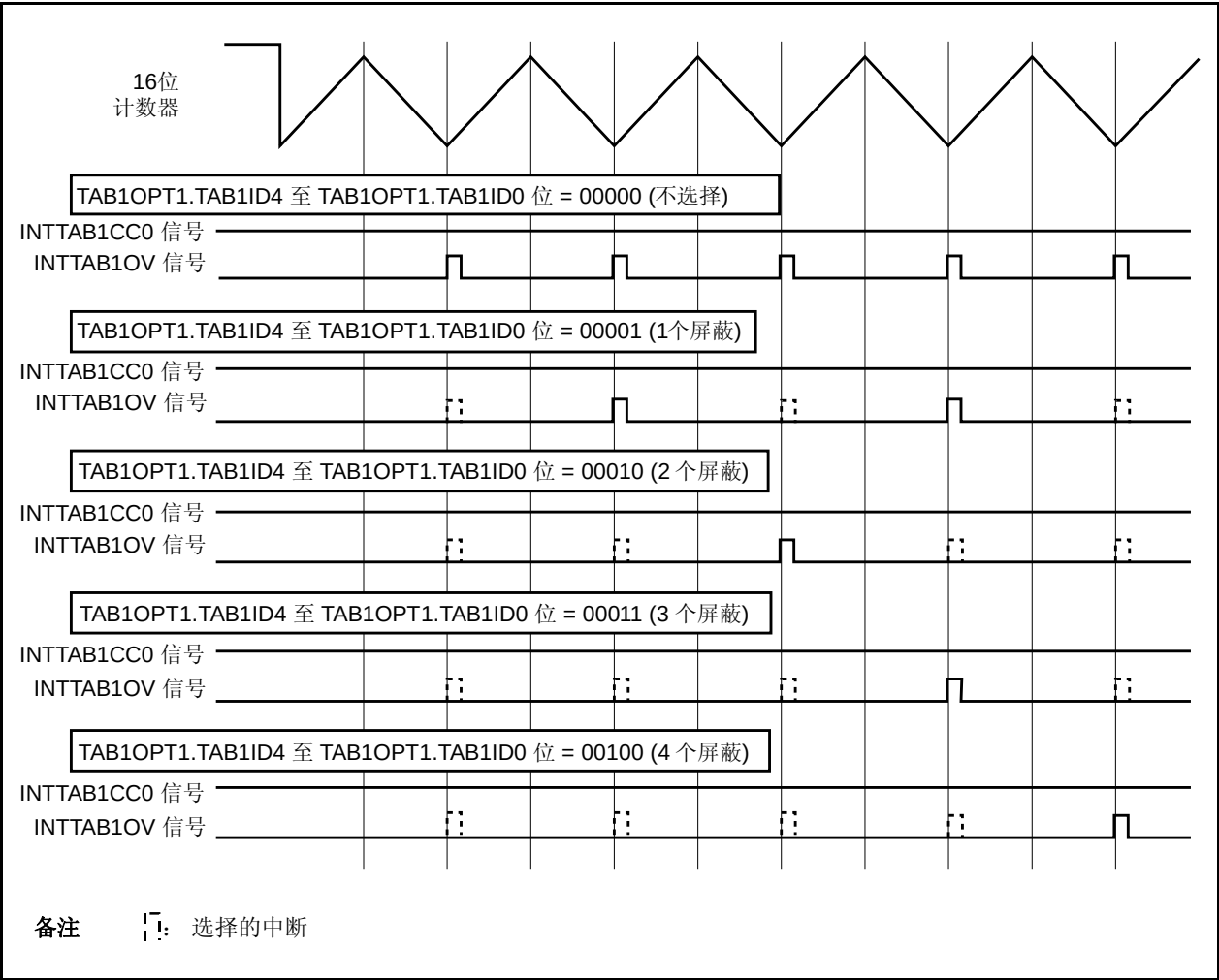


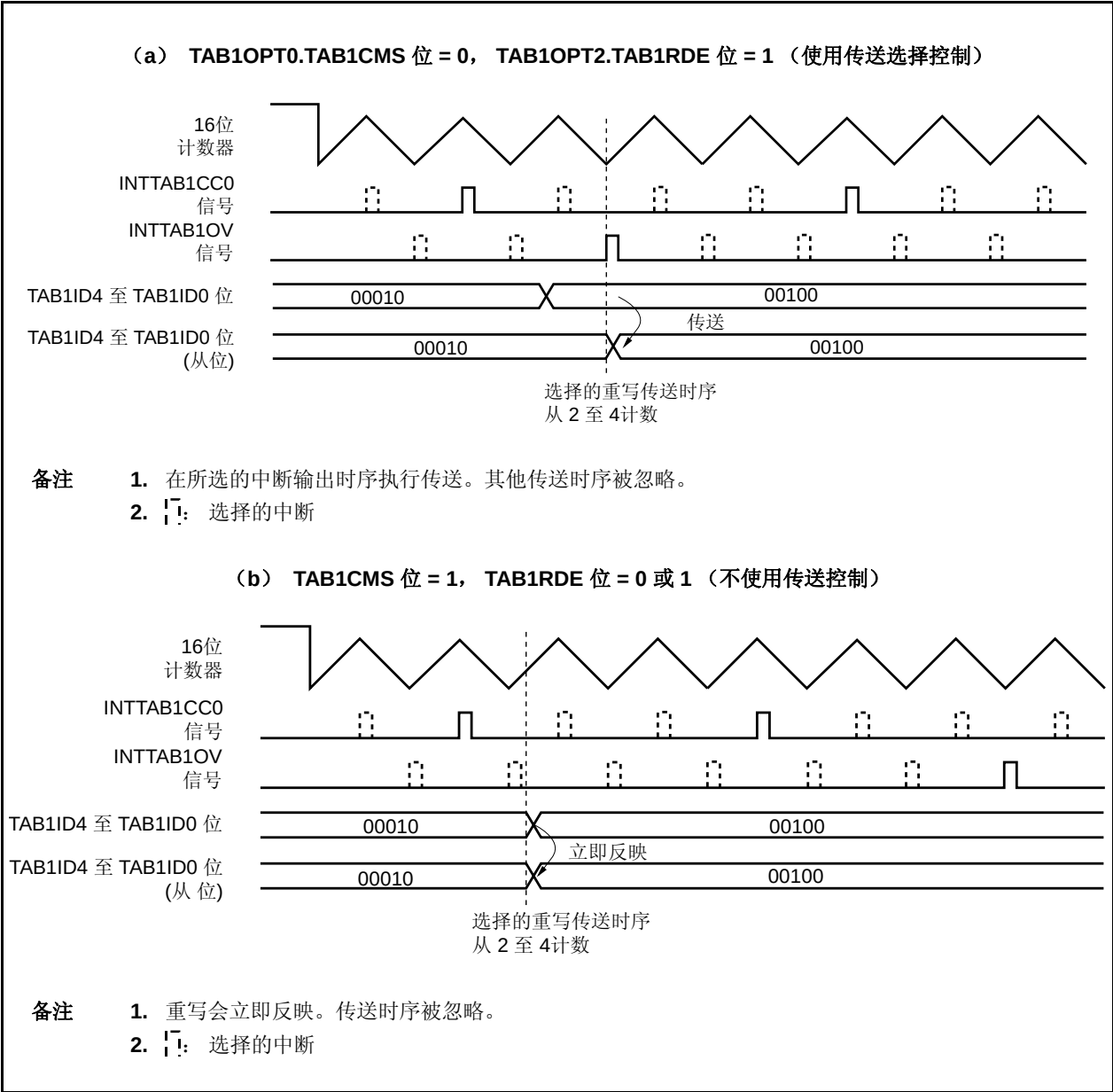
图 11-17. TAB1OPT1.TAB1ICE 位 = 0, TAB1OPT1.TAB1IOE 位 = 1, TAB1OPT2.TAB1RDE 位 = 1 时的中断选择操作（波谷中断输出）



(2) 交替输出波峰中断 (INTTAB1CC0) 和波谷中断 (INTTAB1OV)

要交替输出波峰及波谷中断，则应将 TAB1OPT1.TAB1ICE 和 TAB1OPT1.TAB1IOE 位都设为 1。

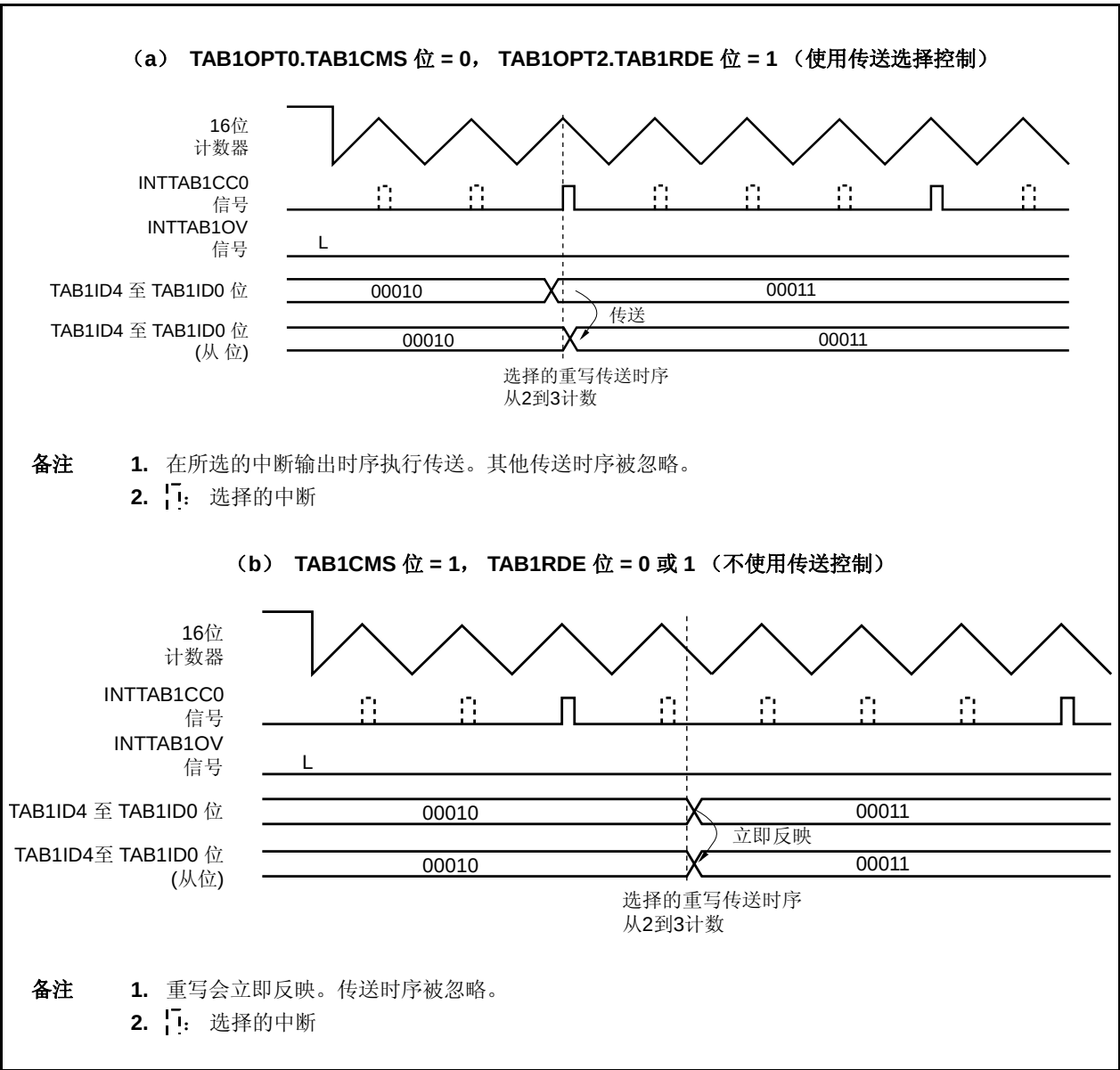
图 11-18. 波峰 / 波谷中断输出



(3) 只输出波峰中断 (INTTAB1CC0)

将 TAB1OPT1.TAB1ICE 位设为 1 并将 TAB1OPT1.TAB1IOE 位清零。

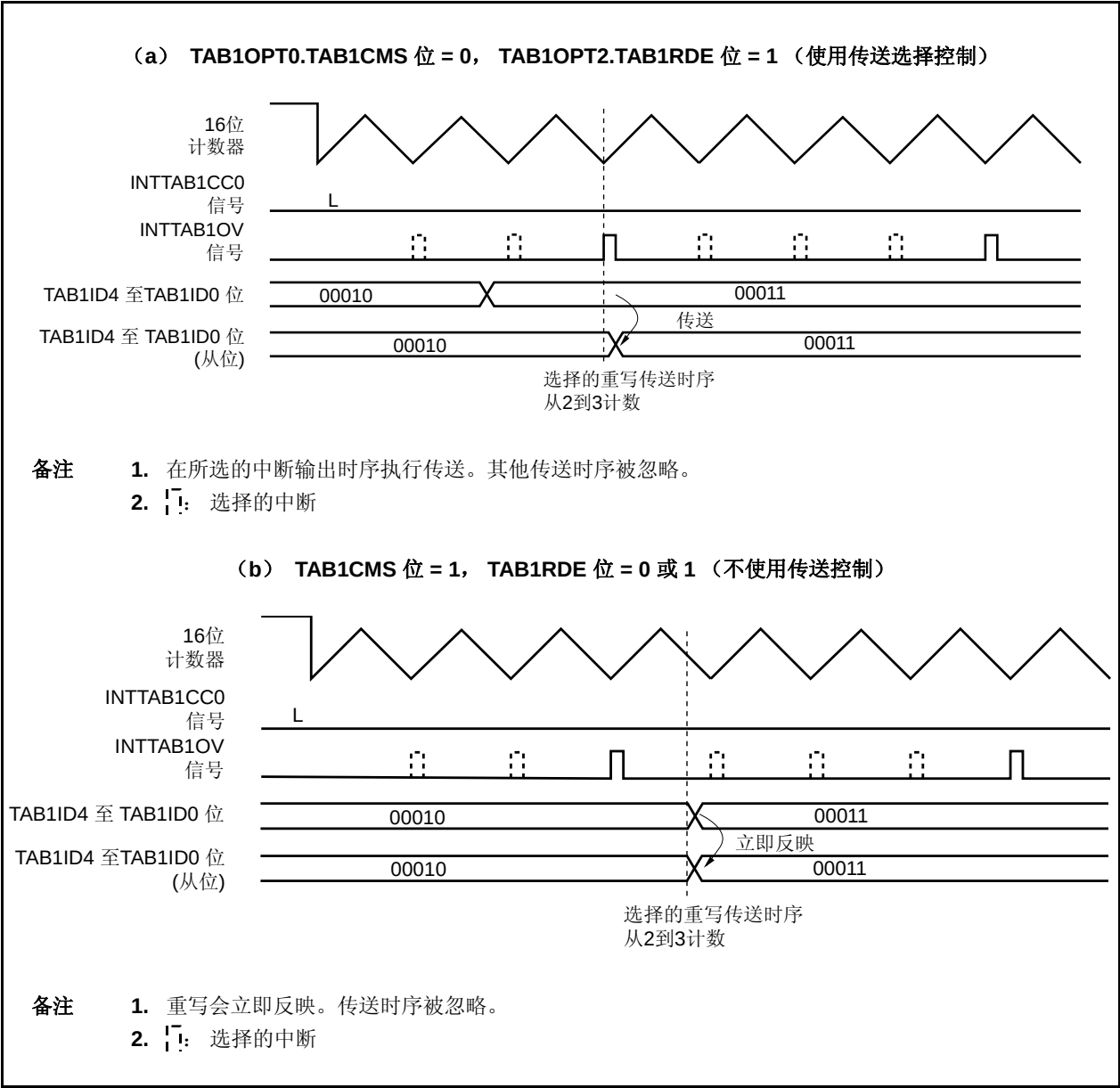
图 11-19. 波峰中断输出



(4) 只输出波谷中断 (INTTAB1OV)

将 TAB1OPT1.TAB1ICE 位清除为 0 并且设置 TAB1IOE 位为 1。

图 11-20. 波谷中断输出



11.4.4 重写带有传送功能的寄存器的操作

以下七种寄存器提供了传送功能并用于控制马达。每个寄存器都有一个缓冲寄存器。

- TAB1CCR0: 指定 16 位计数器 (TAB) 周期的寄存器。
- TAB1CCR1: 指定 TOAB1T1 (U) 及 TOAB1B1 ($\bar{U}$) 的占空系数的寄存器。
- TAB1CCR2: 指定 TOAB1T2 (V) 及 TOAB1B2 ($\bar{V}$) 的占空系数的寄存器。
- TAB1CCR3: 指定 TOAB1T3 (W) 及 TOAB1B3 ($\bar{W}$) 的占空系数的寄存器。
- TAB1OPT1: 指定中断选择的寄存器。
- TAA4CCR0: 指定 A/D 转换开始触发生成时序 (联合操作中的 TAA4) 的寄存器。
- TAA4CCR1: 指定 A/D 转换开始触发生成时序 (联合操作中的 TAA4) 的寄存器。

带有传送功能的寄存器中包含以下三种重写模式：

- 随时重写模式

通过将 TAB1OPT0.TAB1CMS 位设置为 1 来设置该模式。TAB1OPT2.TAB1RDE 位的说明被忽略。
在该模式中，每个比较寄存器都会独立地更新，而比较寄存器的值则会在写入新的值后立即更新。

- 批量重写模式 (传送模式)

通过将 TAB1OPT0.TAB1CMS 位清 0，TAB1OPT1.TAB1ID4 至 TAB1OPT1.TAB1ID0 位清除为 00000 以及 TAB1OPT2.TAB1RDE 位清 0 可以设置该模式。

当数据被写入 TAB1CCR1 寄存器时，这七种寄存器中的数据会在下一个传送时序时立即传送到缓冲寄存器中。除非 TAB1CCR1 寄存器被重写，否则即使其他六个寄存器被重写传送操作也不会被执行。

不考虑中断，传送时序是每个波峰 (16 位计数器值与 TAB1CCR0 寄存器值匹配) 与波谷 (16 位计数器值与 0001H 匹配) 的时序。

- 间隙批量重写模式 (传送选择模式)

该模式可以通过将 TAB1OPT0.TAB1CMS 位清零并将 TAB1OPT2.TAB1RDE 位设为 1 来进行设置。

当数据被写入 TAB1CCR1 寄存器时，这七种寄存器的数据会在下一个传送时序时立即传送到缓冲寄存器中。除非 TAB1CCR1 寄存器被重写，否则即使其他六个寄存器被重写传送操作也不会被执行。

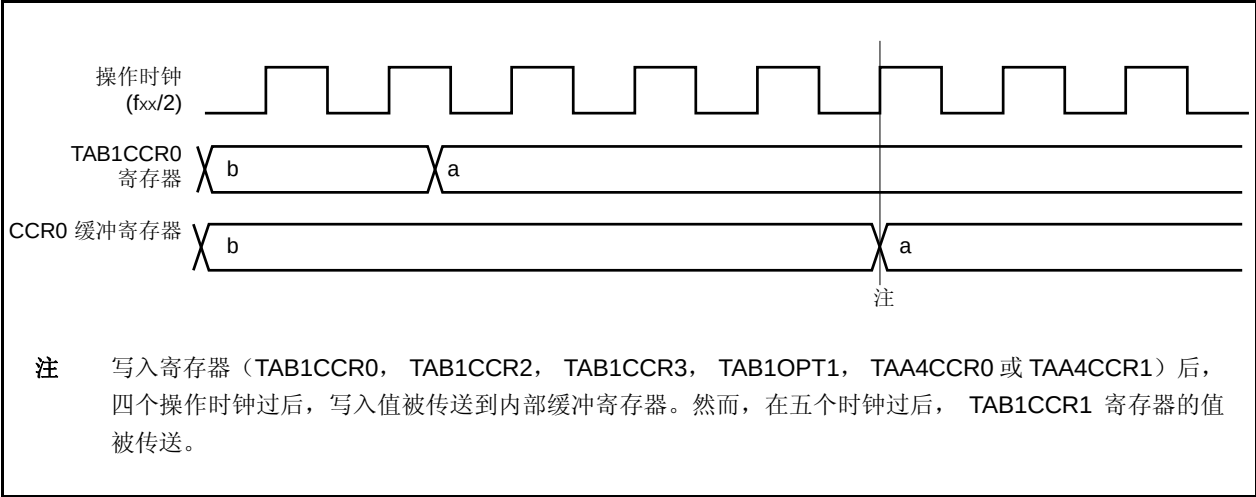
如果通过 TAB1OPT1 寄存器来指定中断选择，那么当中断被选择时传送时序也将会被选择，而这七种寄存器的数据也会在波峰中断 (16 位计数器值与 TAB1CCR0 寄存器值匹配) 或波谷中断 (16 位计数器值与 0001H 匹配) 选中时序处立即被传送。

关于中断选择功能的详细信息，请参见 11.4.3 中断选择功能。

(1) 随时重写模式

该模式通过将 TAB1OPT0.TAB1CMS 位置为 1 来设置。TAB1OPT2.TAB1RDE 位的设置被忽略。
在该模式中，写入带有传送功能的各个寄存器中的值被立即传送到内部缓冲寄存器中并会与计数器的值进行比较。在 16 位计数器的计数值与 TAB1CCRm 寄存器的值匹配后，若带有传送功能的寄存器在该模式中被重写，由于第一次匹配发生后下一次匹配会被忽略，因此重写值将不会被反映出来。如果寄存器在向上计数过程中被重写，那么在计数器开始向下计数后新的寄存器值将会变为有效的。

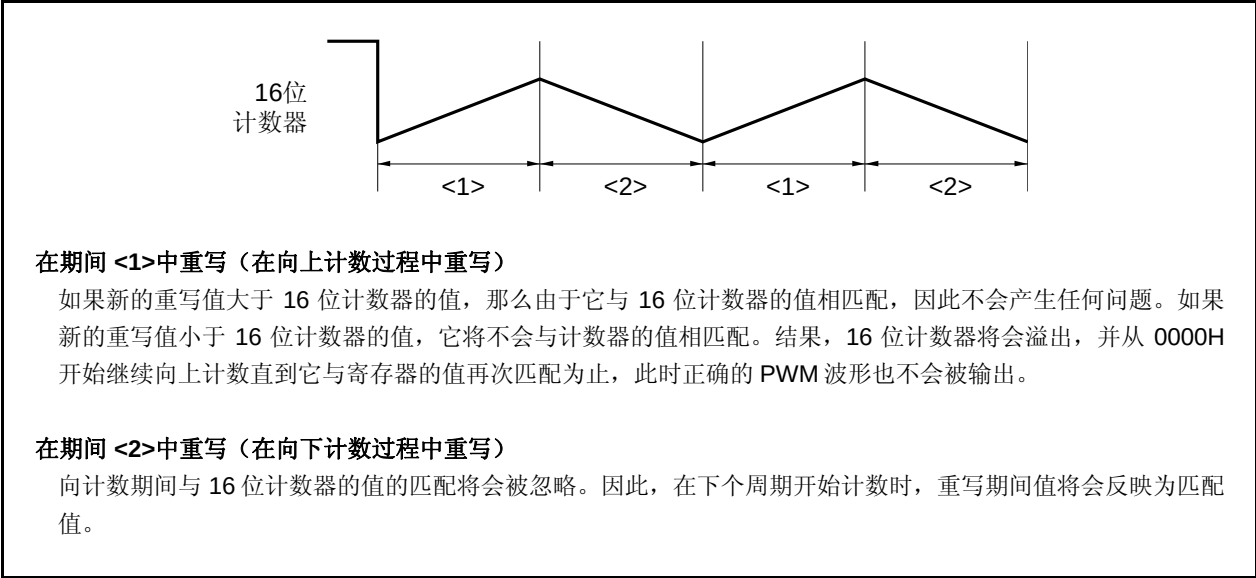
图 11-21. 反映重写值的时序



(a) 重写 TAB1CCR0 寄存器

即使在随时重写模式中对 TAB1CCR0 寄存器进行重写，在某些情况下，新的值也不能反映出来。

图 11-22. 重写 TAB1CCR0 寄存器举例



(b) 重写 TAB1CCRm 寄存器

图 11-24 显示了 16 位计数器的值与 TAB1CCRm 寄存器的值匹配前的重写时序（图 11-23 中的<1>），图 11-25 显示了 16 位计数器的值与 TAB1CCRm 寄存器的值匹配后的重写时序（图 11-23 中的<2>）。

图 11-23. 16 位计数器与 TAB1CCRm 寄存器的基本操作

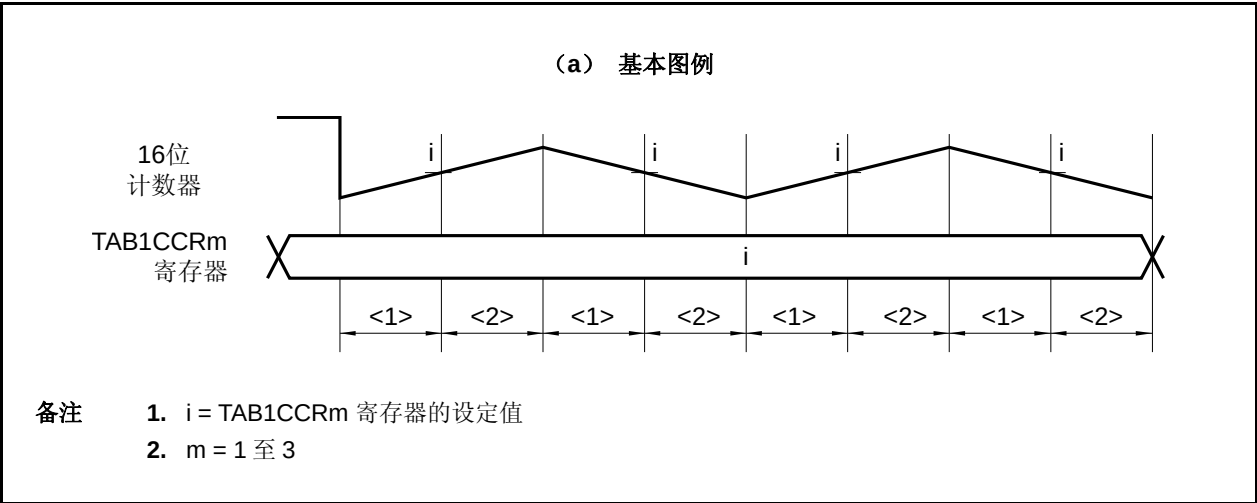
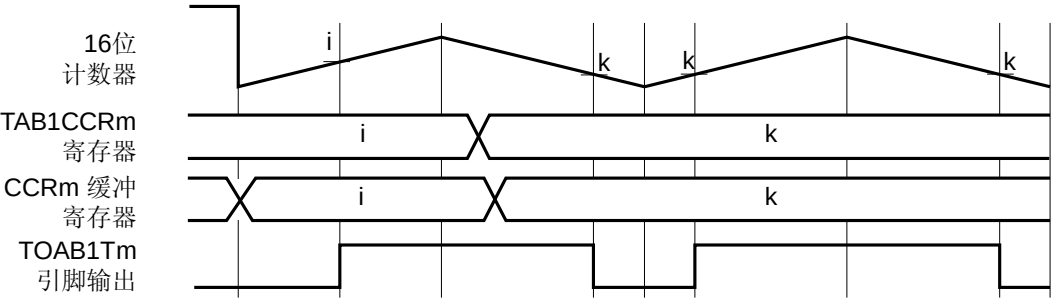


图 11-24. 重写 TAB1CCR1 至 TAB1CCR3 寄存器的示例（匹配发生前重写）

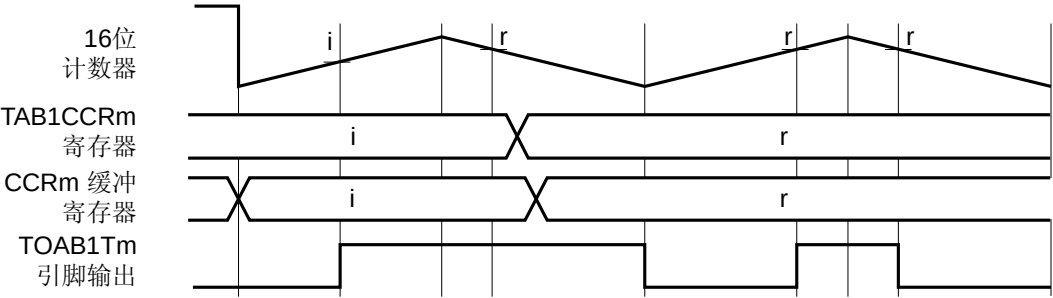
(a)

如果在 TAB1CCRm 寄存器的值与 16 位计数器的值匹配前重写该寄存器，那么寄存器的值将会在寄存器被重写后与 16 位计数器的值相匹配。结果，新的寄存器值将会立即反映。



(b)

如果一个小于 16 位计数器的值（如果计数器执行向下数则大于 16 位计数器值）被写入 TAB1CCRm 寄存器，那么由于寄存器的值与计数器的值不匹配，输出波形将如下所示：



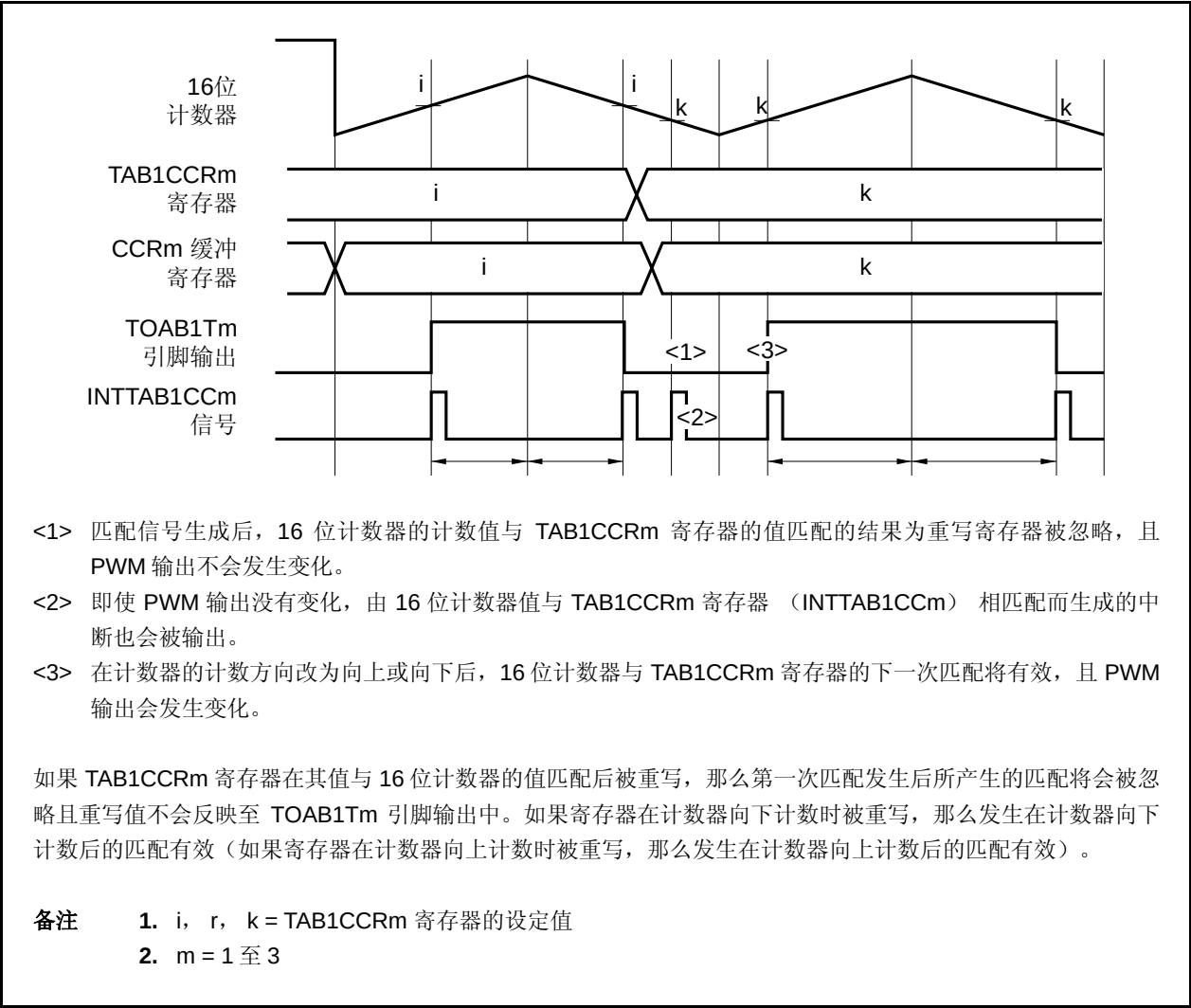
如果寄存器的值与计数器的值不匹配，那么 TOAB1Tm 引脚输出将不会发生变化。即使 16 位计数器的值与 TAB1CCRm 寄存器的值不匹配，TOAB1Tm 引脚输出也会在波峰中断发生时变为高电平，在波谷中断发生时变为低电平。

该功能提供 0% 输出以及 100% 输出。

详细信息，请参见 11.4.2 (2) 0%/100% 的 PWM 输出。

- 备注
- 1. i, r, k = TAB1CCRm 寄存器的设定值
 - 2. m = 1 至 3

图 11-25. 重写 TAB1CCR1 至 TAB1CCR3 寄存器的示例（匹配后重写）



(c) 重写 TAB1OPT1 寄存器

中断选择计数器在 TAB1OPT1 寄存器写入时被清除。中断选择计数器被清除后，所检测到的已发生的中断次数被丢弃。结果，中断产生间隔会暂时延长。

为了避免该操作，应在间隙批量重写模式（传送选择模式）中重写 TAB1OPT1 寄存器。

关于重写 TAB1OPT1 寄存器的详细信息，请参见 11.4.3 中断选择功能。

(2) 批量重写模式（传送模式）

通过将 TAB1OPT0.TAB1CMS 位清 0，TAB1OPT1.TAB1ID4 至 TAB1OPT1.TAB1ID0 位清除为 00000 以及 TAB1OPT2.TAB1RDE 位清 0 可以设置该模式。

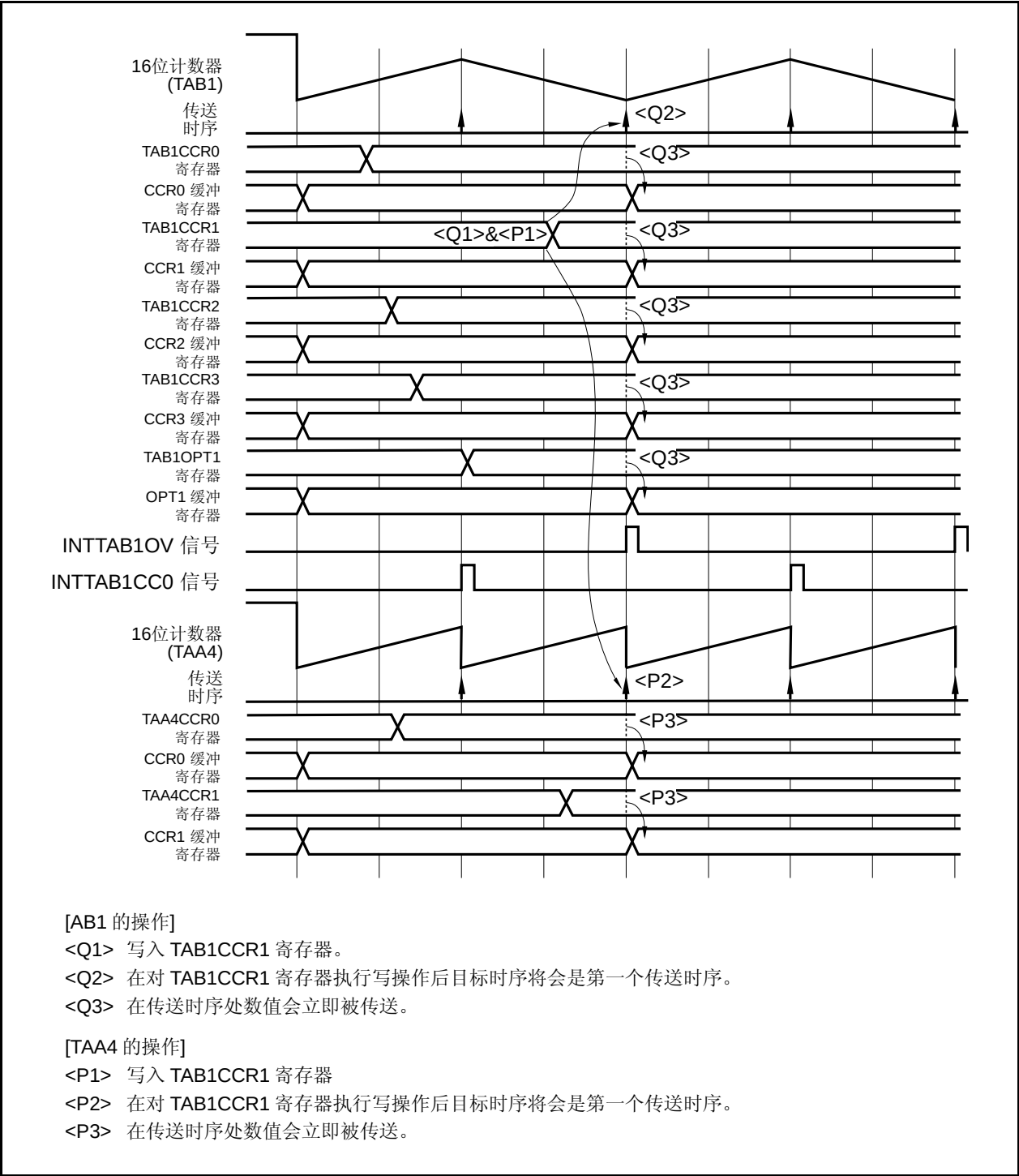
在该模式中，在传送时序处写入各个比较寄存器的值会被立即传送到内部缓冲寄存器中并会与计数器值进行比较。

(a) 重写步骤

如果数据被写入 TAB1CCR1 寄存器，那么在下个传送时序处，设置在 TAB1CCR0 至 TAB1CCR3，TAB1OPT1，TAA4CCR0 以及 TAA4CCR1 寄存器中的值会被立即传送到内部缓冲寄存器中。因此，应最后写入 TAB1CCR1 寄存器。传送时序生成前，禁止在写入 TAB1CCR1 寄存器后对寄存器进行写操作（直到波峰（16 位计数器值与 TAB1CCR0 寄存器值匹配）或波谷（16 位计数器值与 0001H 匹配））。操作步骤如下：

- <1> 重写 TAB1CCR0，TAB1CCR2，TAB1CCR3，TAB1OPT1，TAA4CCR0 以及 TAA4CCR1 寄存器
不要对没必要重写的寄存器进行重写。
- <2> 重写 TAB1CCR1 寄存器
即使没有必要重写 TAB1CCR1 寄存器，也应将相同的值重写到该寄存器中。
- <3> 保持下次重写为未决状态直至传送时序生成。
发生 INTTAB1OV 或 INTTAB1CC0 后对寄存器再次进行重写。
- <4> 返回步骤<1>。

图 11-26. 批量写入模式中的基本操作



(b) 重写 TAB1CCR0 寄存器

在批量重写模式中对 TAB1CCR0 寄存器进行重写时，根据传送是发生在波峰（16 位计数器值与 TAB1CCR0 寄存器值匹配）还是在波谷（16 位计数器值与 0001H 匹配），输出波形会有所不同。通常情况下，建议在 16 位计数器进行向下计数时重写 TAB1CCR0 寄存器，并在波峰的传送时序处传送寄存器的值。

图 11-28 显示了 16 位计数器向上计数时对 TAB1CCR0 寄存器进行重写的例子（图 11-27 的期间<1>中）。图 11-29 显示了 16 位计数器向下计数时对 TAB1CCR0 寄存器进行重写的例子（图 11-27 的期间<2>中）。

图 11-27. 16 位计数器的基本操作

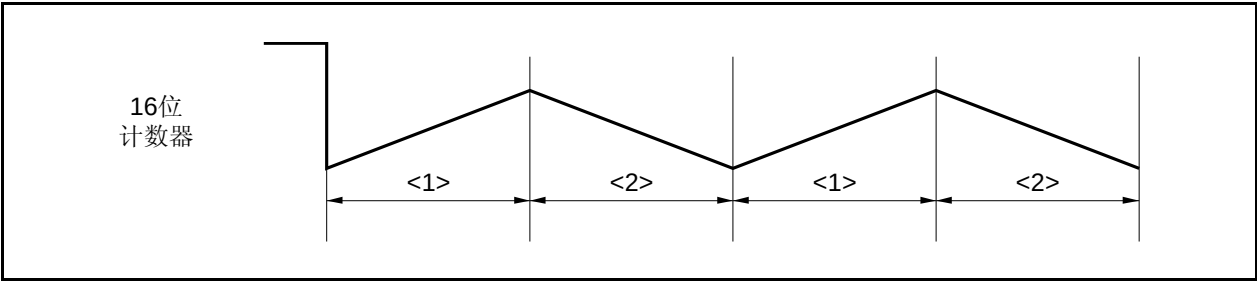


图 11-28 中的传送时序是发生波峰时序的位置。16 位计数器向下计数时，周期将会发生变化并会输出一个不对称的三角波由于周期发生变化，因此需要重写占空系数（电压数据值）。

图 11-28. 重写 TAB1CCR0 寄存器的示例（向上计数期间）

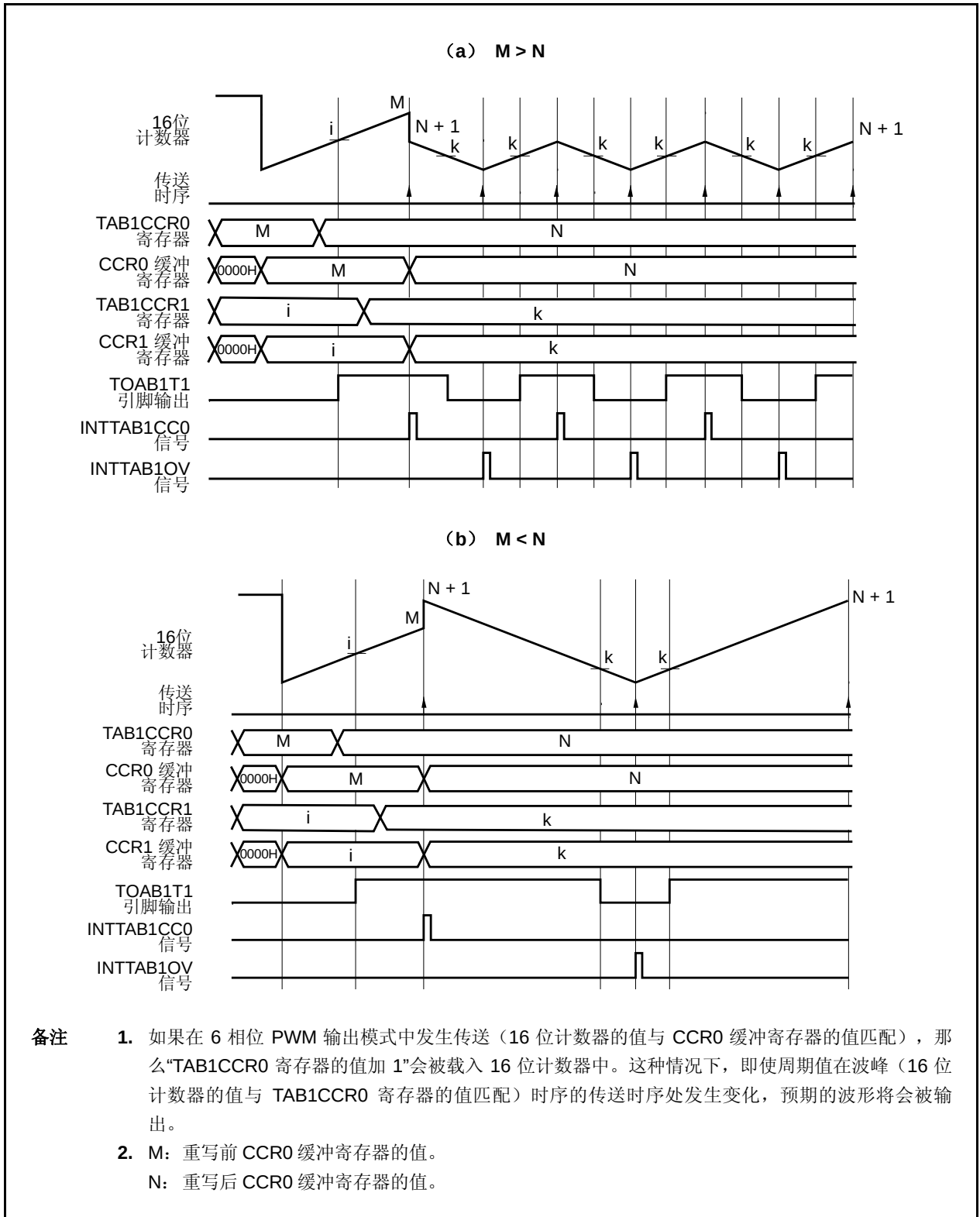
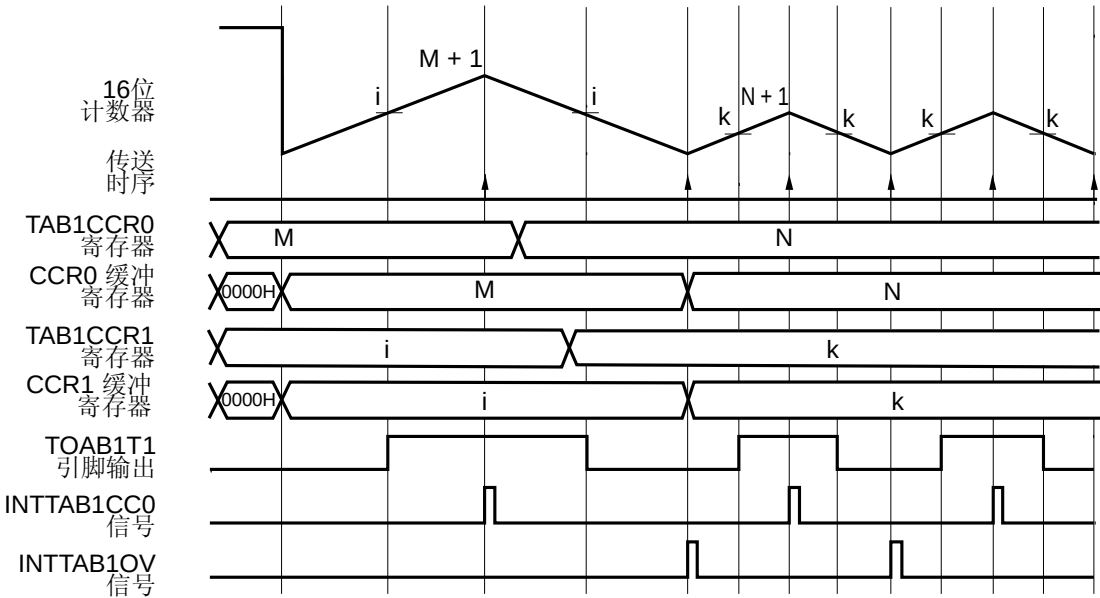


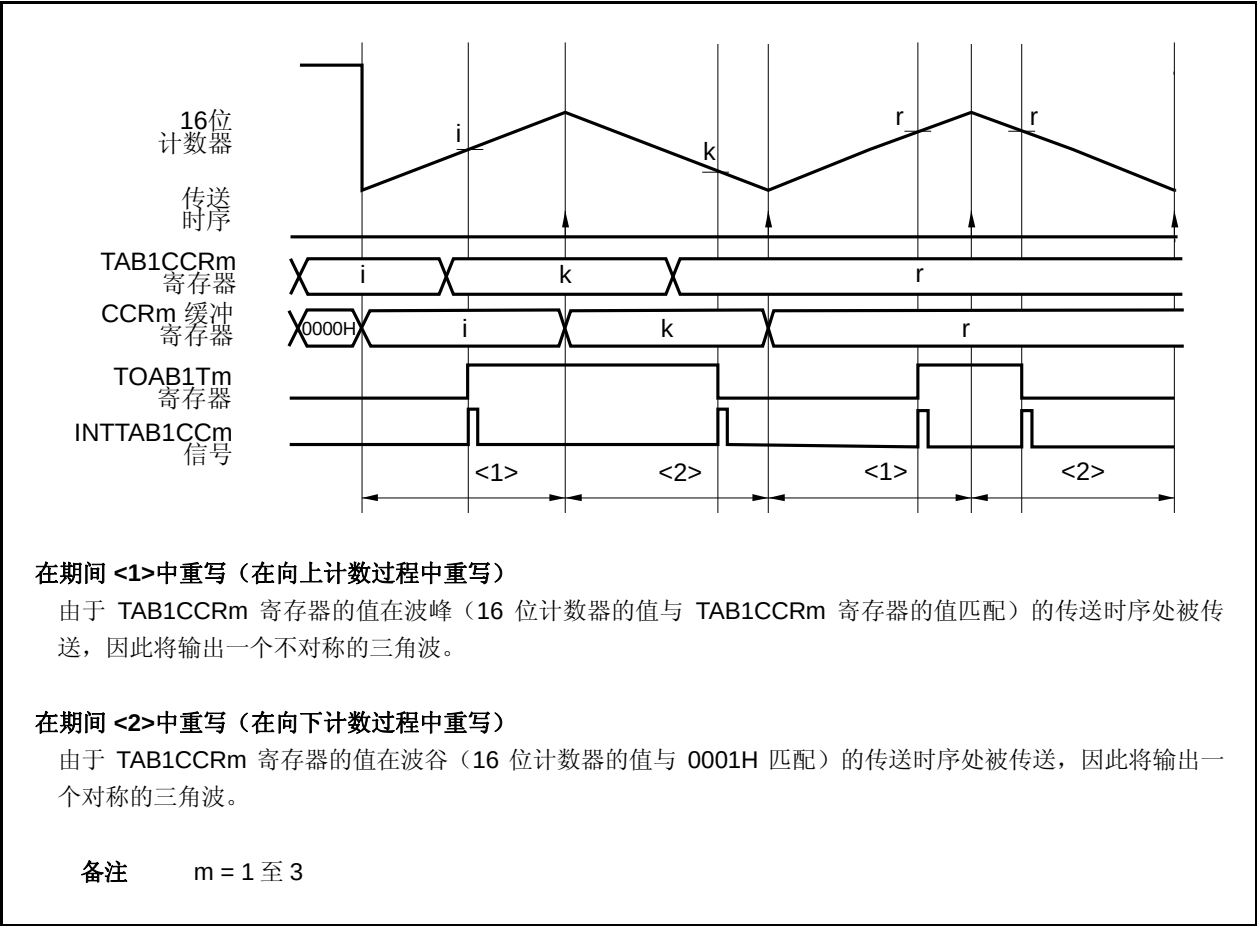
图 11-29. 重写 TAB1CCR0 寄存器的示例（向下计数期间）



由于下个传送时序处于波谷（16 位计数器的值与 0001H 匹配），因此下个周期中的周期值将会发生变化，且对称的三角波的输出也会被保持。由于周期发生变化，因此应按照需要来重写占空系数（电压数据值）。

(c) 重写 TAB1CCRm 寄存器

图 11-30. 重写 TAB1CCRm 寄存器举例



(d) 传送 TAB1OPT1 寄存器值

不要将 TAB1OPT1.TAB1ID4 至 TAB1OPT1.TAB1ID0 位设为不同于 00000 的值。使用中断选择功能时，应在间隙批量重写模式（传送选择模式）中重写 TAB1OPT1 寄存器。
关于重写 TAB1OPT1 寄存器的详细信息，请参见 11.4.3 中断选择功能。

(3) 间隙批量重写模式（传送选择模式）

该模式可以通过将 TAB1OPT0.TAB1CMS 位清零并将 TAB1OPT2.TAB1RDE 位设为 1 来进行设置。

在该模式中，在选择传送时序后写入各个比较寄存器的值会被立即传送到内部缓冲寄存器中并会与计数器值进行比较。传送时序是通过中断选择生成中断（INTTAB1CC0，INTTAB1OV）时的时序。

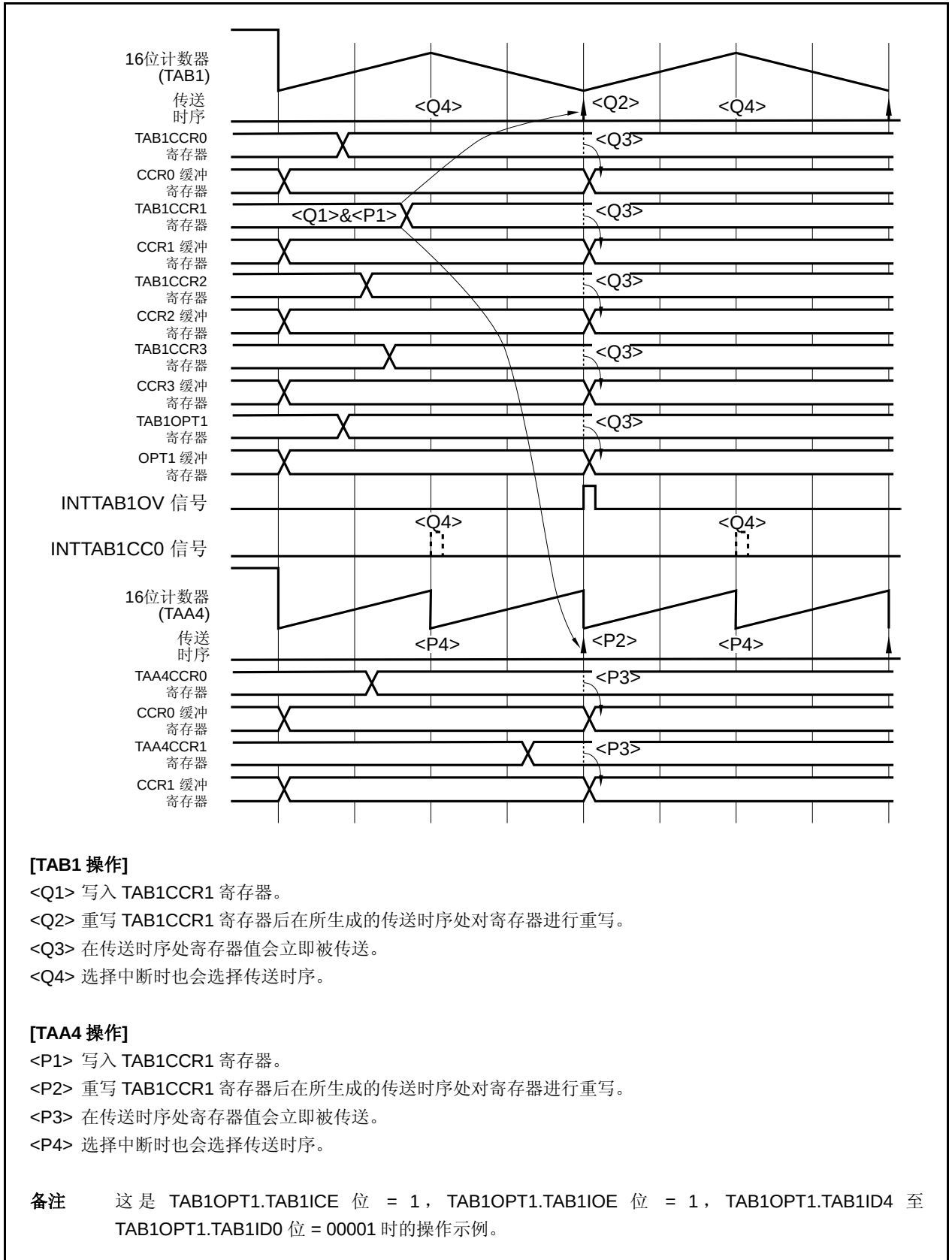
关于中断选择功能的详细信息，请参见 11.4.3 中断选择功能。

(a) 重写步骤

如果数据被写入 TAB1CCR1 寄存器，则在下一个传送时序 TAB1CCR0 至 TAB1CCR3，TAB1OPT1，TAA4CCR0 以及 TAA4CCR1 寄存器的数据被立即传送到内部缓冲寄存器。因此，应最后写入 TAB1CCR1 寄存器。写入 TAB1CCR1 寄存器后禁止写入寄存器直到生成传送时序为止（直到产生 INTTAB1OV 或 INTTAB1CC0 中断）。操作步骤如下：

- <1> 重写 TAB1CCR0，TAB1CCR2，TAB1CCR3，TAB1OPT1，TAA4CCR0 以及 TAA4CCR1 寄存器。
不要对没必要重写的寄存器进行重写。
- <2> 重写 TAB1CCR1 寄存器。
即使没有必要重写 TAB1CCR1 寄存器，也应将相同的值重写到该寄存器中。
- <3> 保持下次重写为未决状态直至传送时序生成。
发生 INTTAB1OV 或 INTTAB1CC0 中断后对寄存器再次进行重写。
- <4> 返回步骤<1>。

图 11-31. 间隙批量写入模式中的基本操作



(b) 重写 TAB1CCR0 寄存器

在间隙批量写入模式中重写 TAB1CCR0 寄存器时，输出波形会根据中断选择设置所指定的波峰或波谷中断发生的位置而变化。下图说明了中断被选择时输出波形的变化。

图 11-32. 重写 TAB1CCR0 寄存器（设为波峰中断时）

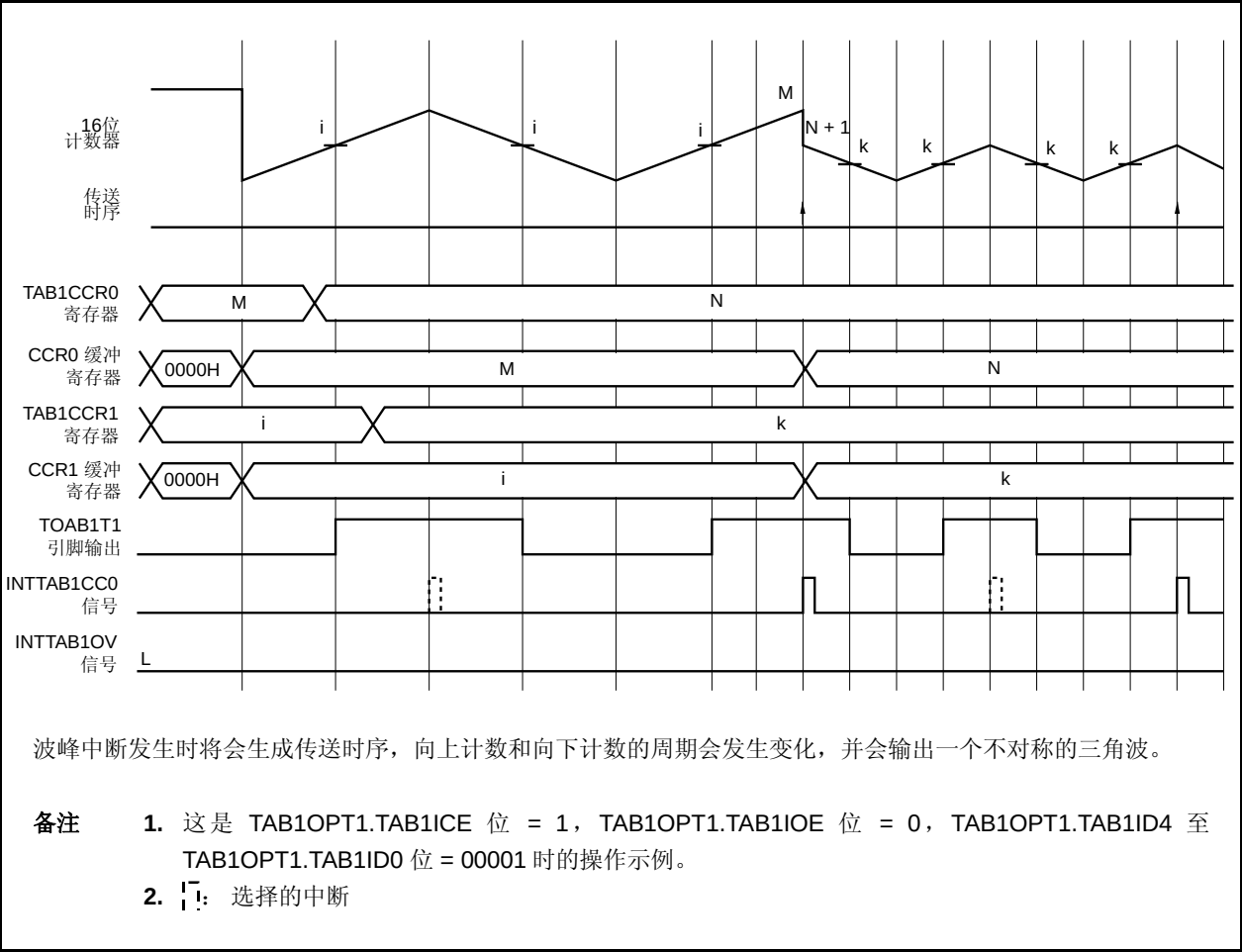
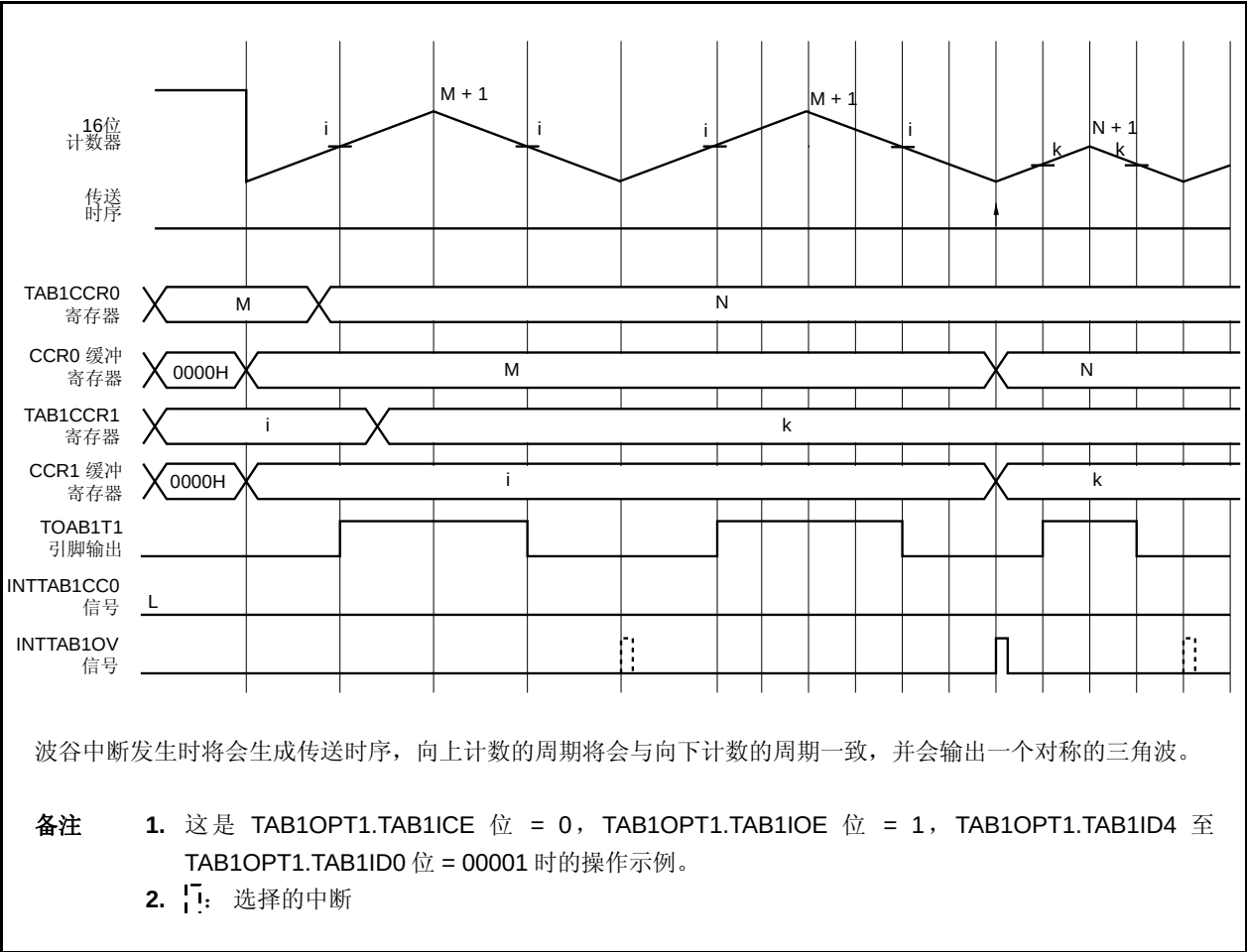


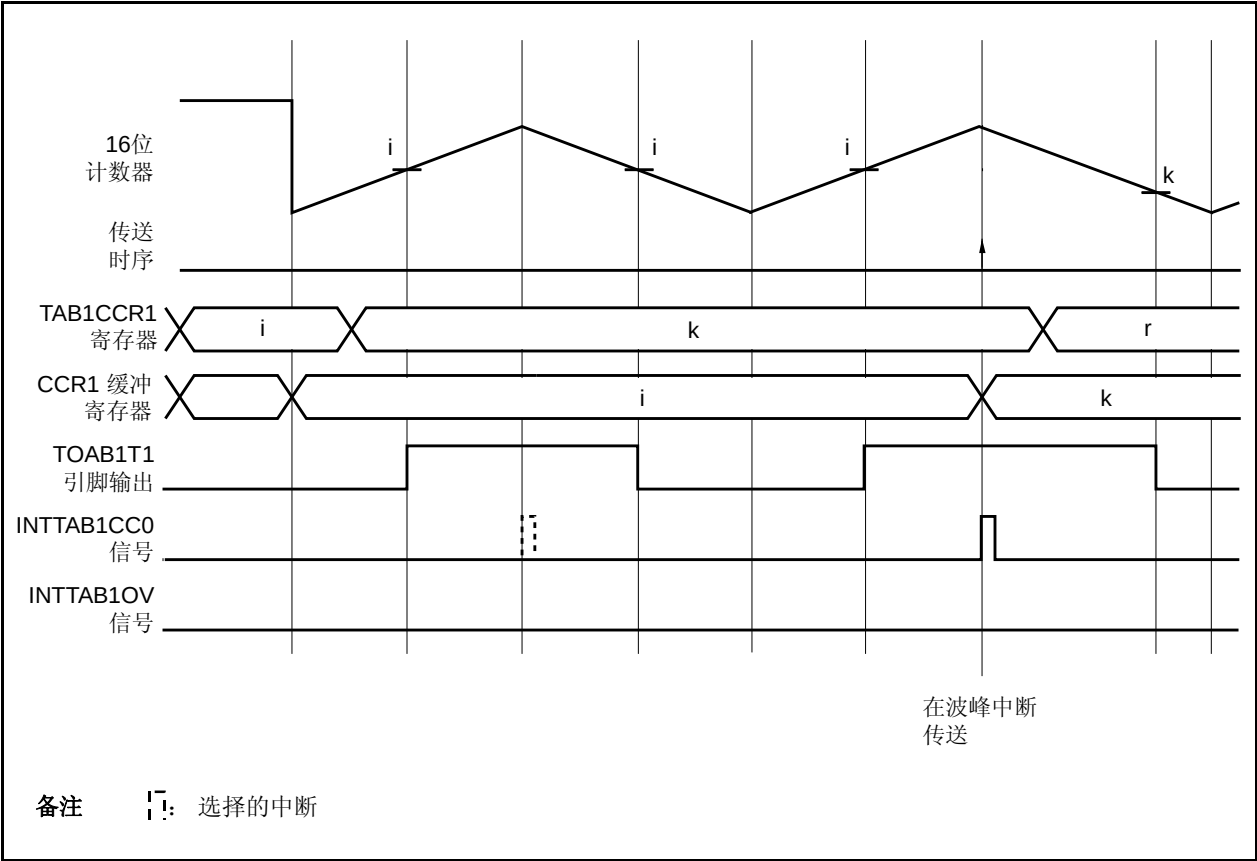
图 11-33. 重写 TAB1CCR0 寄存器（设为波谷中断时）



(c) 重写 TAB1CCR1 至 TAB1CCR3 寄存器

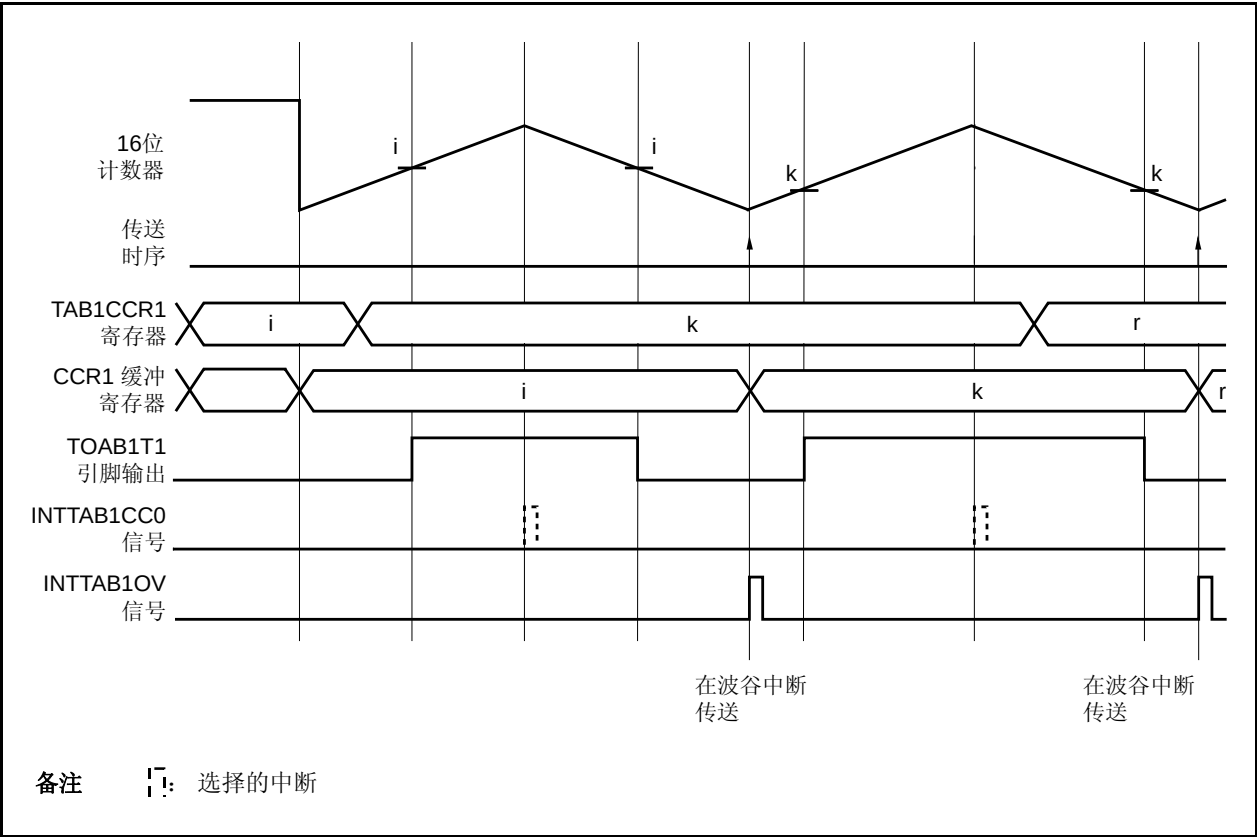
- 设为波峰中断时在波峰传送
由于寄存器在波峰中断的发送时序处被传送，因此会输出一个不对称的三角波。

图 11-34. 重写 TAB1CCR1 寄存器 (TAB1OPT1.TAB1ICE 位 = 1, TAB1OPT1.TAB1IOE 位 = 0, TAB1OPT1.TAB1ID4 至 TAB1OPT1.TAB1ID0 位 = 00001)



- 设为波谷中断时在波谷传送
由于寄存器在波谷中断的发送时序处被传送，因此会输出一个对称的三角波。

图 11-35. 重写 TAB1CCR1 寄存器（TAB1OPT1.TAB1ICE 位 = 1，TAB1OPT1.TAB1IOE 位 = 1，TAB1OPT1.TAB1ID4 至 TAB1OPT1.TAB1ID0 位 = 00001）



(d) 重写 TAB1OPT1 寄存器

由于新的中断选择值会在中断选择计数器的值与 16 位计数器的值匹配时被传送，因此下次中断以及之后的中断会在设定的间隔时间发生。
关于重写 TAB1OPT1 寄存器的详细信息，请参见 11.4.3 中断选择功能。

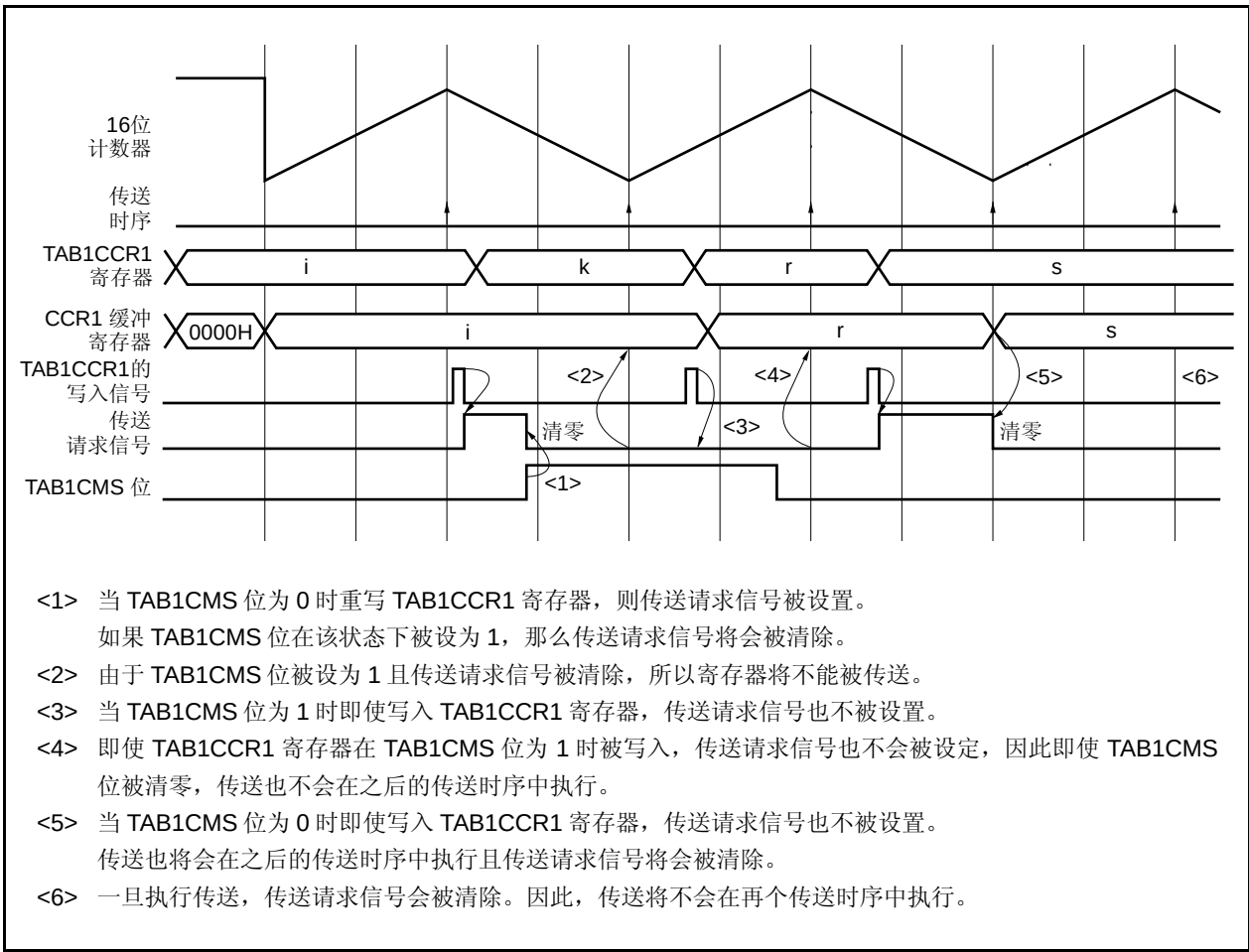
(4) 重写 TAB1OPT0.TAB1CMS 位

TAB1CMS 位可以选择随时重写模式和批量重写模式。该位可以在定时器操作过程中（TAB1CTL0.TAB1CE 位 = 1 时）进行重写。然而，图 11-36 中说明的操作和注意事项是必需的。

如果在 TAB1CMS 位被清零时写入 TAB1CCR1 寄存器，那么将会设定一个传送请求信号（内部信号）。

当传送请求信号被设定时，寄存器将会在下个传送时序处发送，且传送请求信号将会被清除。当 TAB1CMS 位被设为 1 时该传送请求信号也将会被清除。

图 11-36. 重写 TAB1CMS 位



11.4.5 A/D转换开始触发信号输出的TAA4 联合操作

本节介绍在 6 相位 PWM 输出模式中 TAA4 和 TAB1 的联合操作。

在 6 相位 PWM 输出模式中，TAB1 作为主定时器而 TAA4 作为从定时器来执行联合操作。A/D 转换器的转换开始触发信号可以通过 TAA4 的 INTTAA4CC0 和 INTTAA4CC1 信号以及 TAB1 的 INTTAB1OV 和 INTTAB1CC0 信号来设为 A/D 转换开始触发源。

(1) 联合操作开始步骤

应该使用以下步骤来设置 TAA4 和 TAB1 寄存器用来完成联合操作。

(a) TAA4 寄存器的设置（停止 TAB1 和 TAA4 的操作（通过将 TAB1CTL0.TAB1CE 位和 TAA4CTL0.TAA4CE 位清除为 0））

- 将 TAA4CTL1 寄存器设为 85H（设置联合操作从模式以及自运行定时器模式）。
- 将 TAA4OPT0 寄存器清除为 00H（选择比较寄存器）。
- 为 TAA4CCR0 和 TAA4CCR1 寄存器设置一个适当的值（为开始操作的比较设定默认值）。

(b) TAB1 寄存器的设置

- 将 TAB1CTL1 寄存器设为 07H（设置主模式以及 6 相位 PWM 输出模式）。
- 为 TAB1IOC0 寄存器设置合适数值（设置 TOAB1T1 至 TOAB1T3 的输出模式）
然而，应将 TAB1OL0 位清零并将 TAB1OE0 位设为 1（允许正相位输出）。除非完成该设置，否则波峰中断（INTTAB1CC0）和波谷中断（INTTAB1OV）都不能产生。因此，A/D 转换器的转换开始触发信号将不能正确生成。
- 将 TAB1IOC1 和 TAB1IOC2 寄存器设置为 00H（不使用 TAB1 的 TIAB10 至 TIAB13，EVTB1 以及 TRGB1 引脚）
- 将 TAB1OPT0 寄存器的值清除为 00H（选择比较寄存器）。
- 为 TAB1CCR0 至 TAB1CCR3 寄存器设置一个适当的值（为开始操作的比较设定默认值）。
- 将 TAB1CTL0 寄存器的值设置为 0xH（将 TAB1CE 位清除为 0 并设置 TAB1 的操作时钟。）
- 由 TAB1CTL0 寄存器设置的 TAB1 的操作时钟也会提供给 TAA4，并会在相同时序执行计数操作。由 TAA4CTL0 寄存器设置的 TAA4 的操作时钟则被忽略。

(c) TMQOP（TMQ 选项）寄存器的设置

- 为 TAB1OPT1 和 TAB1OPT2 寄存器设置一个适当的值。
- 为 TAB1IOC3 寄存器设置一个适当的值（在输出模式中设置 TOAB1B1 至 TOAB1B3）。
- 为 TAB1DTC 寄存器设置一个适当的值（为开始操作的比较设定默认值）。

(d) 复用功能的设置

- 将端口设置为使用端口控制模式设置的复用功能。

(e) 开始 6 相位 PWM 输出操作后立即将 TAA4CE 位和 TAB1CE 位设为 1

操作期间禁止重写 TAB1CTL0, TAB1CTL1, TAB1IOC1, TAB1IOC2, TAA4CTL0 以及 TAA4CTL1 寄存器。如果在操作期间对这些寄存器中的任一寄存器进行重写, 操作以及 PWM 输出波形将不能被保证。然而, 允许对 TAB1CTL0.TAB1CE 位进行清零的重写操作。禁止操作 (读取 / 写入) 其他 TAB1, TAA4 以及 TMQ 选项寄存器直到先将 TAA4CTL0.TAA4CE 位设为 1, 再将 TAB1CE 位设为 1 为止。

(2) 联合操作清除步骤

要清除联合操作并退出 6 相位 PWM 输出模式, 应使用以下步骤来设置 TAA4 和 TAB1 寄存器。

- <1> 将 TAB1CTL0.TAB1CE 位清除为 0 并停止定时器操作。
- <2> 将 TAA4CTL0.TAA4CE 位清零使得 TAA4 可以被分离。
- <3> 通过使用 TAB1IOC0 寄存器停止定时器输出。
- <4> 将 TAA4CTL1.TAA4SYE 位清零以清除联合操作。

注意事项 禁止操作 (读取 / 写入) 其他 TAB1, TAA4 以及 TMQ 选项寄存器直到先将 TAB1CE 位设为 1, 再将 TAA4CE 位设为 1 为止。

(3) 没有联合 TAA4 时

当 TAA4 的匹配中断信号不需要作为启动 A/D 转换器的转换触发源时, TAA4 可以用作一个不需要联合的独立的定时器。在这种情况下, TAA4 的匹配中断信号不能在 6 相位 PWM 输出模式中用作启动 A/D 转换的触发源。因此, 将 TAB1OPT2.TAB1AT0 至 TAB1OPT2.TAB1AT3 位固定为 0。

其他控制位可以按联合 TAA4 时的相同的方式来使用。

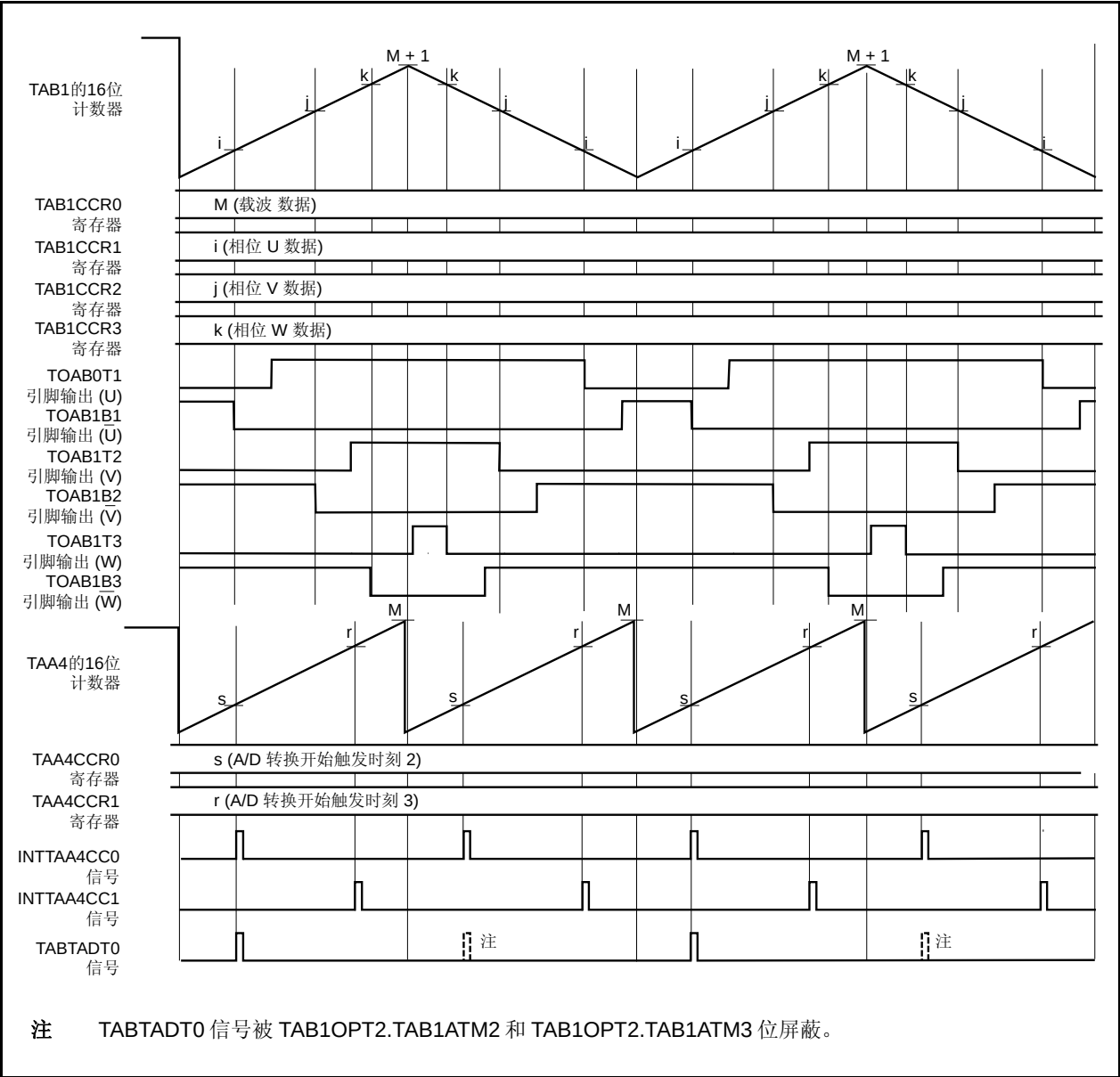
如果 TAA4 没有被联合, 那么 TAA4 的比较寄存器 (TAA4CCR0 和 TAA4CCR1) 将不会受到 TAB1OPT0.TAB1CMS 和 TAB1OPT2.TAB1RDE 位的设置的影响。关于 TAA4 没有被联合时的初始化步骤, 请参见 11.4.5 (1) 联合操作开始步骤中的 (b) 至 (e)。不需要步骤 (a), 因为它是用于为联合操作设置 TAA4 的步骤。

(4) 联合操作过程中 TAA4 的基本操作

TAA4 的 16 位计数器仅限于向上计数。16 位计数器通过 TAB1CCR0 寄存器的设定周期值来清除, 并从 0000H 开始再次计数。当该计数器开始计数时, 它的计数值与 TAB1 的 16 位计数器的值相同。然而, 它的计数值在 TAA4 的 16 位计数器向下计数时将不同于 16 位计数器的值。

- TAB1 向上计数时 (相同的值)
 - TAB1 的 16 位计数器: 0000H → M (向上计数)
 - TAA4 的 16 位计数器: 0000H → M (向上计数)
- TAB1 向下计数时 (不相同的值)
 - TAB1 的 16 位计数器: M + 1 → 0001H (向下计数)
 - TAA4 的 16 位计数器: 0000H → M (向上计数)

图 11-37. 联合操作期间的 TAA4



11.4.6 A/D转换开始触发输出功能

V850ES/JG3-H 和 V850ES/JH3-H 具有选择四种触发源（INTTAB1OV， INTTAB1CC0， INTTAA4CC0， INTTAA4CC1）以生成 A/D 转换开始触发信号（TABTADT0）的功能。

触发源由 TAB1OPT2.TAB1AT0 至 TAB1OPT2.TAB1AT3 位指定。

- TAB1AT0 位 = 1:
发生 INTTAB1OV（计数器下溢）时生成 A/D 转换开始触发信号。
- TAB1AT1 位 = 1:
发生 INTTAB1CC0（周期匹配）时生成 A/D 转换开始触发信号。
- TAB1AT2 位 = 1:
发生 INTTAA4CC0（联合操作期间 TAA4 的 TAA4CCR0 寄存器的匹配）时生成 A/D 转换开始触发信号。
- TAB1AT3 位 = 1:
发生 INTTAA4CC1（联合操作期间 TAA4 的 TAA4CCR1 寄存器的匹配）时生成 A/D 转换开始触发信号。

由 TAB1AT0 至 TAB1AT3 位选择的 A/D 转换开始触发信号是只读的并且会被输出。因此，可以同时指定两个或更多的触发源。

由 TAB1AT0 和 TAB1AT1 位选择的 INTTAB1OV 和 INTTAB1CC0 信号是选择的中断信号。

因此，这些信号会在中断被选择后输出，并且除非允许中断输出（TAB1OPT1.TAB1ICE 和 TAB1OPT1.TAB1IOE 位），否则将不会输出 A/D 转换开始触发信号。

如果 TAB1AT2 和 TAB1AT3 位进行如此设置，则 TAA4 的触发源（INTTAA4CC0 和 INTTAA4CC1）具有根据 16 位计数器向上计数 / 向下计数的状态来屏蔽 A/D 转换开始触发信号的功能。

- TAB1ATM2 位：与 TAB1AT2 位一致并控制 TAA4 的 INTTAA4CC0（匹配中断信号）。
 - TAB1ATM2 位 = 0: A/D 转换开始触发信号在 16 位计数器向上计数时（TAB1OPT0.TAB1CUF 位 = 0）输出，A/D 转换开始触发信号在 16 位计数器向下计数时（TAB1OPT0.TAB1CUF 位 = 1）不会输出。
 - TAB1ATM2 位 = 1: A/D 转换开始触发信号在 16 位计数器向上计数时（TAB1OPT0.TAB1CUF 位 = 1）输出，A/D 转换开始触发信号在 16 位计数器向下计数时（TAB1OPT0.TAB1CUF 位 = 0）不会输出。
- TAB1ATM3 位：与 TAB1AT3 位一致并控制 TAA4 的 INTTAA4CC1（匹配中断信号）。
 - TAB1ATM3 位 = 0: A/D 转换开始触发信号在 16 位计数器向上计数时（TAB1OPT0.TAB1CUF 位 = 0）输出，A/D 转换开始触发信号在 16 位计数器向下计数时（TAB1OPT0.TAB1CUF 位 = 1）不会输出。
 - TAB1ATM3 位 = 1: A/D 转换开始触发信号在 16 位计数器向上计数时（TAB1OPT0.TAB1CUF 位 = 1）输出，A/D 转换开始触发信号在 16 位计数器向下计数时（TAB1OPT0.TAB1CUF 位 = 0）不会输出。

定时器操作时可以重写 TAB1ATM3， TAB1ATM2 以及 TAB1AT3 至 TAB1AT0 位。如果设置 A/D 转换开始触发信号的位在定时器操作时被重写，那么新的设置会立即反映在 A/D 转换开始触发信号的输出状态上。这些控制位不具有传送功能，并且只能在随时重写模式中使用。

- 注意事项
1. 由 **TAB1AT2** 和 **TAB1AT3** 位设置的 **A/D** 转换开始触发信号输出只能在 **TAA4** 作为 **TAB1** 的从定时器来执行联合功能时使用。如果 **TAB1** 和 **TAA4** 都不能执行联合操作，或者不同于 **6** 相位 **PWM** 输出模式的模式被使用，那么输出将得不到保证。
 2. **TAB1** 信号输出用于在内部识别 **16** 位计数器是向上计数还是向下计数。因此，通过将 **TAB1IOC0.TAB1OL0** 位清零并将 **TAB1IOC0.TAB1OE0** 位设为 **1** 可以允许 **TOAB10** 引脚输出。

图 11-38. A/D 转换开始触发 (TABTADT0) 信号输出的示例 (TAB1OPT1.TAB1ICE 位 = 1, TAB1OPT1.TAB1IOE 位 = 1, TAB1OPT1.TAB1ID4 至 TAB1OPT1.TAB1ID0 位 = 00000: 没有中断选择)

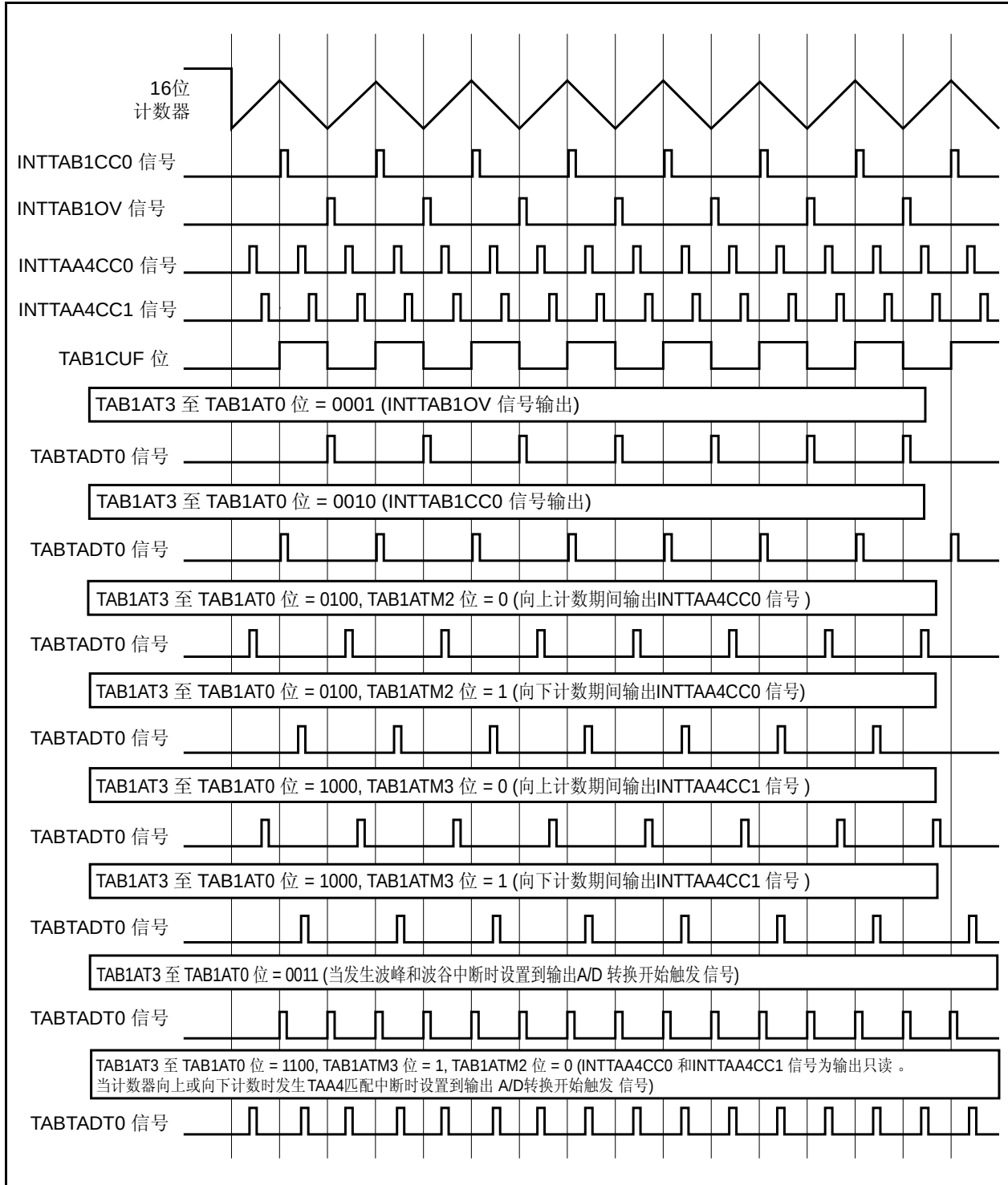


图 11-39. A/D 转换开始触发（TABTADT0）信号输出的示例（TAB1OPT1.TAB1ICE 位 = 0，TAB1OPT1.TAB1IOE 位 = 1，TAB1OPT1.TAB1ID4 至 TAB1OPT1.TAB1ID0 位 = 00010：带有中断选择）（1）

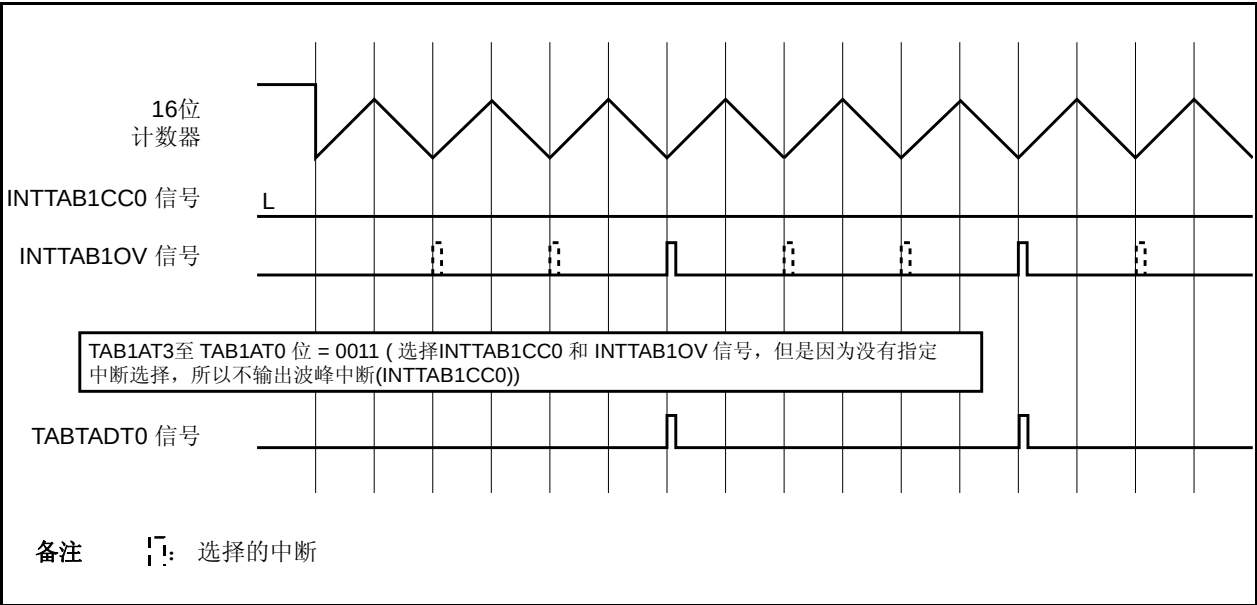
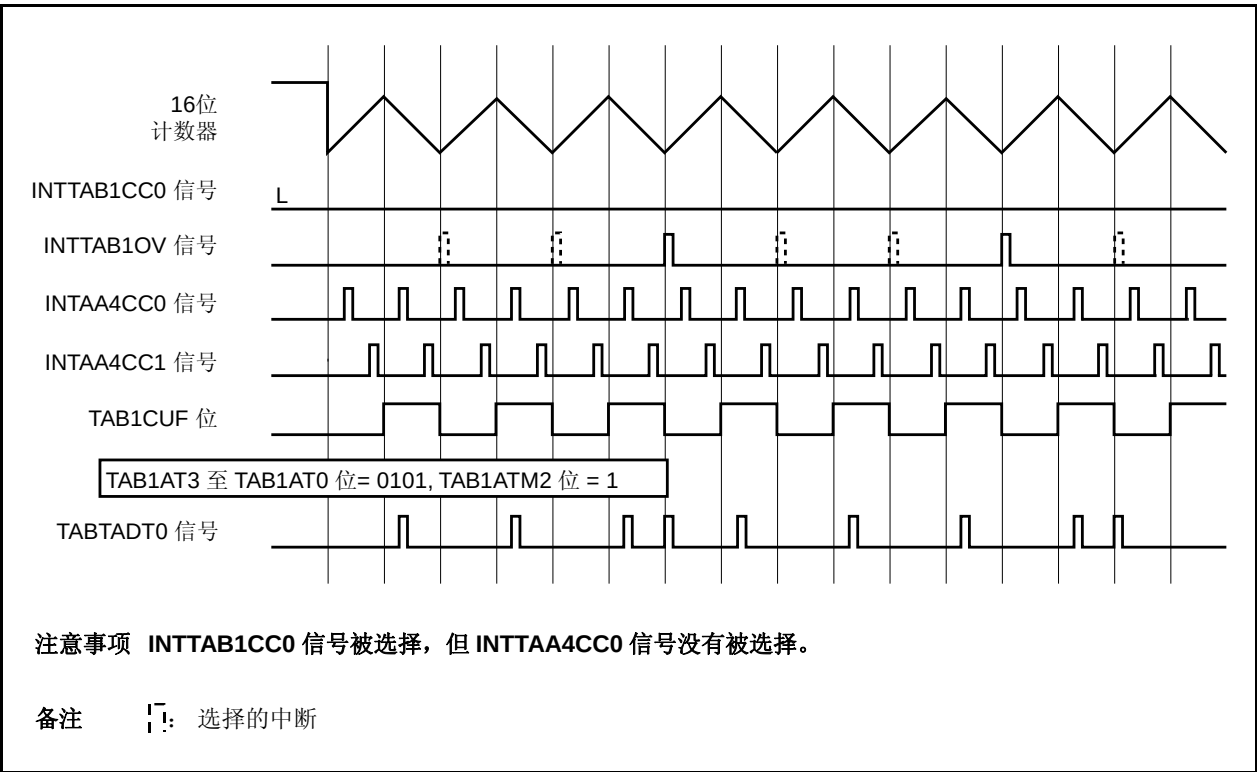


图 11-40. A/D 转换开始触发（TABTADT0）信号输出的示例（TAB1OPT1.TAB1ICE 位 = 0，TAB1OPT1.TAB1IOE 位 = 1，TAB1OPT1.TAB1ID4 至 TAB1OPT1.TAB1ID0 位 = 00010：带有中断选择）（2）



(1) 边界条件下的操作 (16 位计数器与 INTTAA4CC0 信号匹配时的操作)

表 11-3. TAB1CCR0 寄存器 = M, TAB1AT2 位 = 1, TAB1ATM2 位 = 0 时的操作
(选择向上计数期间)

TAA4CCR0 寄存器的值	TAB1 的 16 位计数器的值	TAA4 的 16 位计数器的值	TAB1 的 16 位计数器的状态	由 INTTAA4CC0 信号选择的 TABTADT0 信号输出
0000H	0000H	0000H	—	输出
0000H	M + 1	0000H	—	不输出
0001H	0001H	0001H	向上计数	输出
0001H	M	0001H	向下计数	不输出
M	M	M	向上计数	输出
M	0001H	M	向下计数	不输出

表 11-4. TAB1CCR0 寄存器 = M, TAB1AT2 位 = 1, TAB1ATM2 位 = 1 时的操作
(选择向下计数期间)

TAA4CCR0 寄存器的值	TAB1 的 16 位计数器的值	TAA4 的 16 位计数器的值	TAB1 的 16 位计数器的状态	INTTAA4CC0 信号选择的 TABTADT0 信号输出
0000H	0000H	0000H	—	不输出
0000H	M + 1	0000H	—	输出
0001H	0001H	0001H	向上计数	不输出
0001H	M	0001H	向下计数	输出
M	M	M	向上计数	不输出
M	0001H	M	向下计数	输出

注意事项 TAB1CCR0 寄存器 = M 时 TAA4CCRm 寄存器允许“0”至“M”的设置。但禁止“M + 1”及更高的值的设置。

如果设置了“M + 1”或更高的值，则 TAA4 的 16 位计数器将会通过“M”来清除。因此，不输出 TABTADT0 信号。

备注 m = 0, 1

第十二章 实时计数器

12.1 功能

实时计数器（RTC）具有以下特点：

- 提供用年、月、星期、日、小时、分和秒格式向上计数至 99 年。
- 年、月、星期、日、小时、分和秒用 BCD 码显示^{注1}。
- 具有报警中断功能。
- 稳定时期中断功能（时期：1 个月到 0.5 秒）。
- 间隔中断功能（时期：1.95 ms 至 125 ms）。
- 1 Hz 的引脚输出功能。
- 32.768 kHz 的引脚输出功能。
- 512 Hz 或 16.384 kHz 的引脚输出功能。
- 钟表错误校正功能。
- 副时钟操作或主时钟操作^{注2}可选。

- 注
1. BCD（二进制编码的十进制数）码以 4 位二进制格式表示十进制数的各位数字。
 2. 波特率发生器专用于实时计数器将主时钟频率分为 32.768 kHz 以供使用。

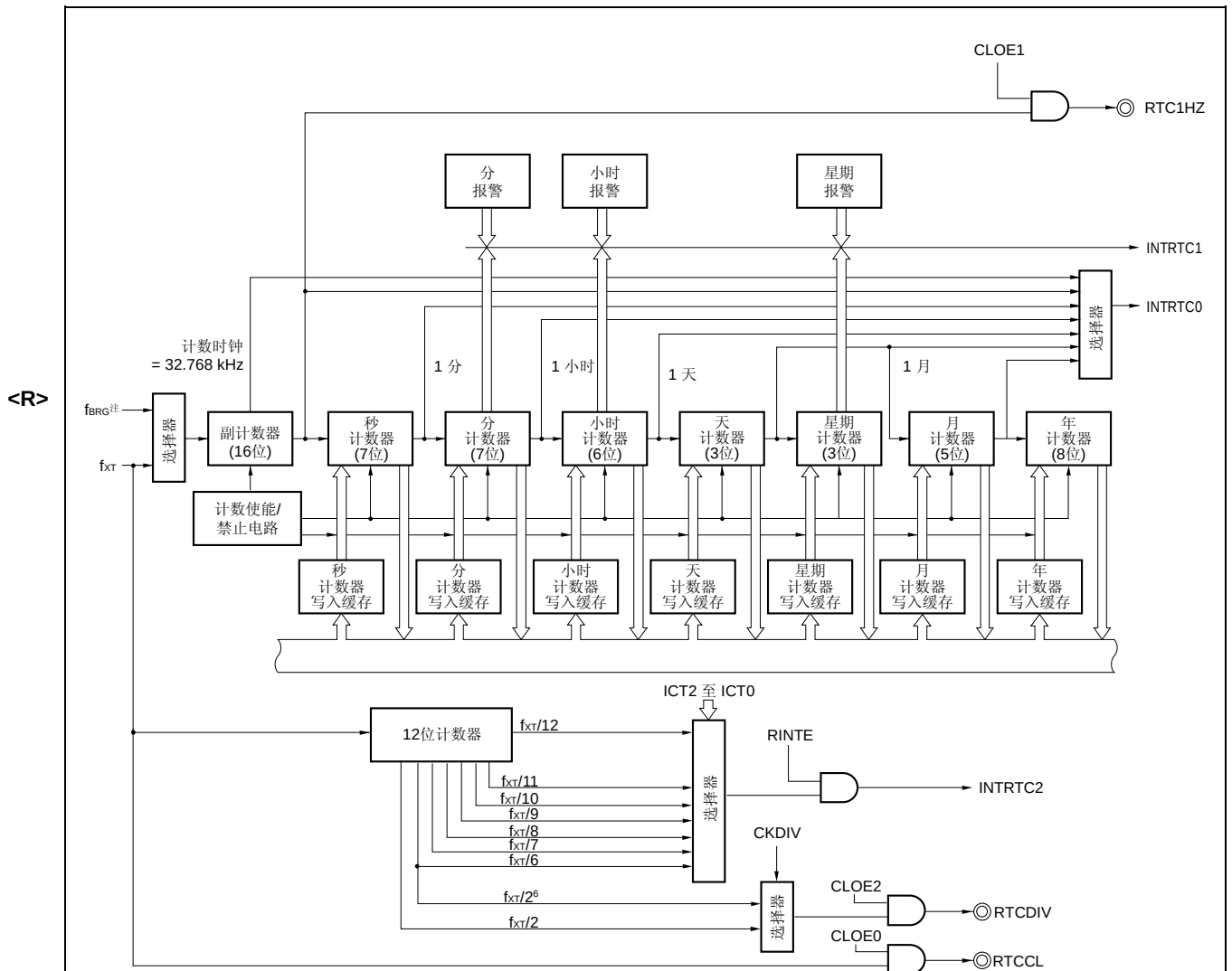
12.2 配置

实时计数器包含以下硬件：

表 12-1. 实时计数器的配置

项目	配置
<R> <R>	实时计数器控制寄存器 0（RC1CC0）
	实时计数器控制寄存器 1（RC1CC1）
	实时计数器控制寄存器 2（RC1CC2）
	实时计数器控制寄存器 3（RC1CC3）
	副计数寄存器（RC1SUBC）
	秒计数寄存器（RC1SEC）
	分计数寄存器（RC1MIN）
	小时计数寄存器（RC1HOUR）
	天计数寄存器（RC1DAY）
	星期计数寄存器（RC1WEEK）
	月计数寄存器（RC1MONTH）
	年计数寄存器（RC1YEAR）
	钟表错误校正寄存器（RC1SUBU）
	分报警寄存器（RC1ALM）
	小时报警寄存器（RC1ALH）
	星期报警寄存器（RC1ALW）
	预分频模式寄存器 0（PRSM0）
	预分频比较寄存器 0（PRSCM0）

图 12-1. 实时计数器的框图



<R> 注 关于 f_{BRG} 的详细情况，参见12.3 (17) 预分频模式寄存器0 (PRSM0) 和12.3 (18) 预分频比较寄存器0 (PRSCM0)。

备注

- f_{BRG} :** 实时计数器计数时钟频率
- f_{XT} :** 副时钟频率
- INTRTC0:** 实时计数器固定周期信号
- INTRTC1:** 实时计数器报警匹配信号
- INTRTC2:** 实时计数器间隔信号

12.2.1 引脚配置

包括在实时计数器中的 RTC 输出复用为表 12-2 中的功能。使用各引脚时必须设置端口功能（参见表 4-20 端口引脚用作复用功能引脚）。

表 12-2. 引脚配置

引脚编号		端口	RTC输出	其它复用功能
V850ES/JG3-H	V850ES/JH3-H			
30	42	P35	RTC1HZ	TIAA11/TOAA11
28	40	P33	RTCDIV	TIAA01/TOAA01/RTCCL
28	40	P33	RTCCL	TIAA01/TOAA01/RTCDIV

12.2.2 中断功能

RTC 包括以下三种类型的中断信号：

- (1) INTRTC0
按 0.5 秒、秒、分、小时、天或月产生固定周期中断信号。
- (2) INTRTC1
报警中断信号。
- (3) INTRTC2
以 $f_{XT}/6$, $f_{XT}/7$, $f_{XT}/8$, $f_{XT}/9$, $f_{XT}/10$, $f_{XT}/11$ 或 $f_{XT}/12$ 为周期产生的间隔中断信号。

12.3 寄存器

实时计数器由以下寄存器控制：

- (1) 实时计数器控制寄存器 0（RC1CC0）
- RC1CC0 寄存器选择实时计数器输入时钟。
- 本寄存器可按 8 位或 1 位读写。
- 复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: FFFFFADDH

RC1CC0

7	6	5	4	3	2	1	0
RC1PWR	RC1CKS	0	0	0	0	0	0

RC1PWR	实时计数器操作控制
0	停止实时计数器操作
1	允许实时计数器操作

RC1CKS	操作时钟选择
0	选择 f _{XT} 作为操作时钟
1	选择 f _{BRG} 作为操作时钟

注意事项

1.

实时计数器工作中停止它时（RC1PWR = 1 → 0）按照12.4.8初始化实时计数器中的描述进行。

2.

仅当实时计数器停止（RC1PWR 位 = 0）时，才可重写RC1CKS位。此外，禁止在将RC1PWR 位从0设置为1的同时重写RC1CKS位。

- (2) 实时计数器控制寄存器 1（RC1CC1）
- RC1CC1 寄存器是 8 位寄存器，用于启动或停止实时计数器，控制 RTCCL 和 RTC1HZ 引脚，选择 12 小时或 24 小时系统，并具有设置固定周期中断功能。
- 本寄存器可按 8 位或 1 位读写。
- 复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: FFFFFADEH

	7	6	5	4	3	2	1	0
RC1CC1	RTCE	0	CLOE1	CLOE0	AMPM	CT2	CT1	CT0

RCTE	各计数器操作的控制
0	停止计数器操作
1	允许计数器操作

CLOE1	RTC1HZ 引脚输出控制
0	禁止 RTC1HZ 引脚输出 (1 Hz)
1	允许 RTC1HZ 引脚输出 (1 Hz)

CLOE0	RTCCL 引脚输出控制
0	禁止 RTCCL 引脚输出 (32.768 kHz)
1	允许 RTCCL 引脚输出 (32.768 kHz)

AMPM	12小时系统/24小时系统 选择
0	12小时系统 (显示a.m. 和 p.m.)
1	24小时系统

CT2	CT1	CT0	固定周期中断 (INTRTC0) 选择
0	0	0	不使用周期中断
0	0	1	0.5秒产生一次 (与秒同步向上计数)
0	1	0	1 秒产生一次 (与秒同步向上计数)
0	1	1	1 分产生一次 (在每分钟的00秒)
1	0	0	1 小时产生一次 (在每小时的00分00秒)
1	0	1	1 天产生一次 (在每天的00小时00分00秒)
1	1	x	1 个月产生一次 (在每月一天的00小时00分00秒 a.m.)

- 注意事项
1. 禁止在RTCE位为1时将0写入RTCE位。按照12.4.8初始化实时计数器中的描述通过清除RC1PWR位将RTCE清零。
 2. 当CLOE1位设置改变时RTC1HZ输出操作如下：
 - 从0改变为1时：两个或更短的时钟过后，RTC1HZ输出一个1 HZ脉冲。
 - 从1改变为0时：两个或更短的时钟过后，RTC1HZ输出停止（固定为低电平）。
 3. 关于设置或改变AMPM位参见12.4.1 初始设置和12.4.2 时钟操作期间重写各计数器。此外，当重写AMPM位时重置RC1HOUR寄存器。
 4. 当重写CT2至CT0位而实时计数器操作时（RC1PWR位= 1），参见12.4.4 时钟操作期间改变INTRTC0中断设置。

(3) 实时计数器控制寄存器 2 (RC1CC2)

RC1CC2 寄存器是 8 位寄存器，用于控制报警中断功能和计数器的等待。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: FFFFFADFH

	7	6	5	4	3	2	1	0
RC1CC2	WALE	0	0	0	0	0	RWST	RWAIT

WALE	报警中断 (INTRTC1) 操作控制
0	不基于报警匹配产生中断
1	基于报警匹配产生中断

RWST	实时计数器等待状态
0	计数器操作
1	停止从秒至年计数器的向上计数 (允许读取和写入计数器值)

这是一个状态标志以表明 RWAIT 位的设置是否有效。
确认RWST位为1后再读取或写入计数器值。

RWAIT	实时计数器等待控制
0	设置计数器操作。
1	停止秒到年计数器的计数操作。 (计数器值读取/写入模式)

该位控制计数器的操作。
当读取或写入计数器值时确保将1写入该位。
如果当RWAIT位为1时RC1SUBC寄存器溢出，则溢出信息在内部保存，将0写入 RWAIT位两个时钟或更短的时间过后， RC1SEC寄存器向上计数。
但是，当RWAIT位为1时如果秒计数器值被重写，则保存的溢出信息被丢弃。

- 注意事项
1. 当重写WALE位而实时计数器操作时（RC1PWR位= 1），参见12.4.5 时钟操作期间改变INTRTC1中断设置。
 2. 当读取或写入各计数器时确保RWST位设置为1。
 3. 即使RWAIT位设置为0，当正在写入各计数器时，RWST位不能变成0。当写入各计数器完成时，其变为0。

(4) 实时计数器控制寄存器 3 (RC1CC3)

RC1CC3 寄存器是 8 位寄存器，用于控制间隔中断功能和 RTCDIV 引脚。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: FFFFFAE0H

	7	6	5	4	3	2	1	0
RC1CC3	RINTE	CLOE2	CKDIV	0	0	ICT2	ICT1	ICT0

RINTE	间隔中断(INTRTC2)控制
0	不产生间隔中断。
1	产生间隔中断。

CLOE2	RTCDIV 引脚输出控制
0	禁止RTCDIV 引脚输出。
1	允许 RTCDIV 引脚输出。

CKDIV	RTCDIV 引脚输出频率选择
0	从 RTCDIV 引脚输出512 Hz (1.95 ms)
1	从 RTCDIV 引脚输出16.384 kHz (0.061 ms)

ICT2	ICT1	ICT0	间隔中断选择 (INTRTC2)
0	0	0	$2^6/f_{XT}$ (1.953125 ms)
0	0	1	$2^7/f_{XT}$ (3.90625 ms)
0	1	0	$2^8/f_{XT}$ (7.8125 ms)
0	1	1	$2^9/f_{XT}$ (15.625 ms)
1	0	0	$2^{10}/f_{XT}$ (31.25 ms)
1	0	1	$2^{11}/f_{XT}$ (62.5 ms)
1	1	X	$2^{12}/f_{XT}$ (125 ms)

- 注意事项
1. 当重写RINTE位而实时计数器操作时 (RC1PWR位= 1)，参见12.4.7 时钟操作期间改变 INTRTC2中断设置。
 2. 当CLOE2位设置改变时RTCDIV输出操作如下：
 - 从0改变为1时：两个或更短的时钟过后，CKDIV位设置的脉冲被输出。
 - 从1改变为0时：两个或更短的时钟过后，RTCDIV输出停止（固定为低电平）。
 3. 当重写ICT2至ICT0位而实时计数器操作时 (RC1PWR位= 1)，参见12.4.7 时钟操作期间改变 INTRTC2中断设置。

(5) 副计数寄存器 (RC1SUBC)

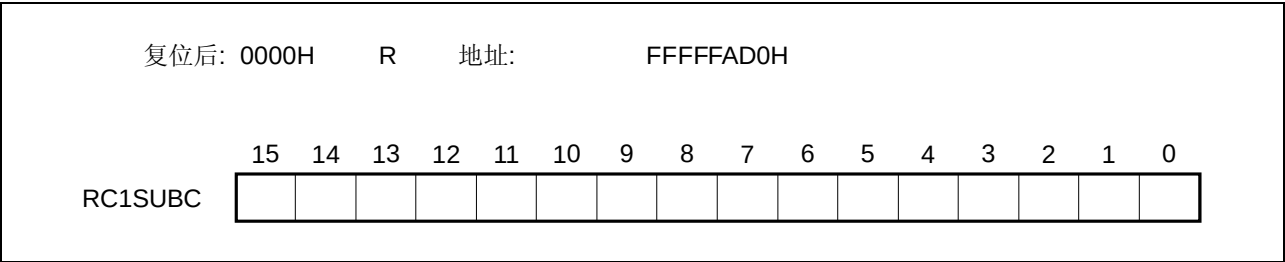
RC1SUBC 寄存器是 16 位寄存器，用于计数实时计数器的 1 秒的参考时间。

计数值的范围为 0000H 至 7FFFH，用 32.768 kHz 时钟计数一秒。

本寄存器为只读寄存器，可以按 16 位读取。

复位后本寄存器设置为 0000H。

- 注意事项
- 1 当用 RC1SUBU 寄存器进行校正时，该值可能变成 8000H 或更大的数值。
 - 2. 写入秒计数寄存器也可将该寄存器清零。
 - 3. 如果操作期间读取该寄存器，则不能保证读取值，因为读取到的是正在变化的数值。



(6) 秒计数寄存器 (RC1SEC)

RC1SEC 寄存器是 8 位寄存器，从 0 到 59（十进制）中取值，表示秒的计数值。

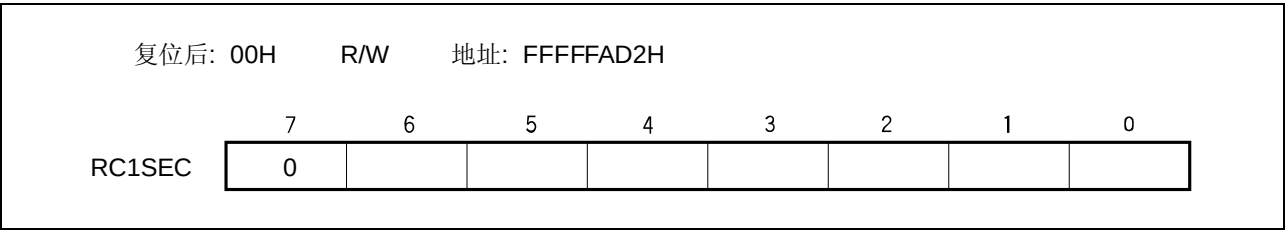
当副计数器溢出时其向上计数。

当数据写入该寄存器时，其先被写入缓存，最多两个时钟（32.768 kHz）过后被写入计数器。给该寄存器以 BCD 码的格式设置一个 00 到 59 的十进制数。如果设置的数值超出范围，则一个周期过后寄存器返回正常值。

该寄存器可按 8 位为单位进行读写。

复位后本寄存器被设置为 00H。

备注 当读取或写入 RC1SEC 寄存器时，参见 12.4.1 初始设置，12.4.2 时钟操作期间重写各计数器以及 12.4.3 设置操作期间读取各计数器。

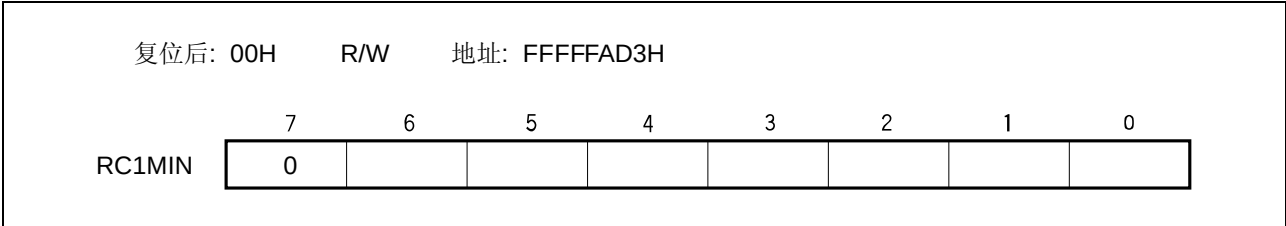


(7) 分计数寄存器 (RC1MIN)

RC1MIN 寄存器是 8 位寄存器，从 0 到 59（十进制）中取值，表示分的计数值。
当秒计数器溢出时其向上计数。
当数据写入该寄存器时，其先被写入缓存，最多两个时钟（32.768 kHz）过后被写入计数器。给该寄存器以 BCD 码的格式设置一个 00 到 59 的十进制数。
本寄存器可以按 8 位进行读写。
复位后本寄存器被设置为 00H。

注意事项 禁止给 RC1MIN 寄存器设置 00 至 59 外的值。

备注 当读取或写入 RC1MIN 寄存器时，参见 12.4.1 初始设置，12.4.2 时钟操作期间重写各计数器以及 12.4.3 设置操作期间读取各计数器。



(8) 小时计数寄存器 (RC1HOUR)

RC1HOUR 寄存器是 8 位寄存器，从 0 到 23 或 1 到 12（十进制）中取值，表示小时的计数值。
当分计数器溢出时其向上计数。
当数据写入该寄存器时，其先被写入缓存，最多两个时钟（32.768 kHz）过后被写入计数器。以 BCD 码的格式给该寄存器设置 00 至 23、01 至 12 或 21 至 32 中的十进制数值。如果设置的数值超出范围，则一个周期过后寄存器返回正常值。
本寄存器可以按 8 位进行读写。
复位后本寄存器被设置为 12H。
但是，复位后如果 AMPM 位设置为 1 则该寄存器的值为 00H。

注意事项 1. 如果 AMPM = 0（如果选择 12 小时的系统）RC1HOUR 寄存器的位 5 表示 a.m.（0）或 p.m.（1）。
2. 禁止给 RC1HOUR 寄存器设置除 01 至 12、21 至 32（AMPM 位= 0）或 00 至 23（AMPM 位= 1）外的值。

备注 当读取或写入 RC1HOUR 寄存器时，参见 12.4.1 初始设置，12.4.2 时钟操作期间重写各计数器以及 12.4.3 设置操作期间读取各计数器。

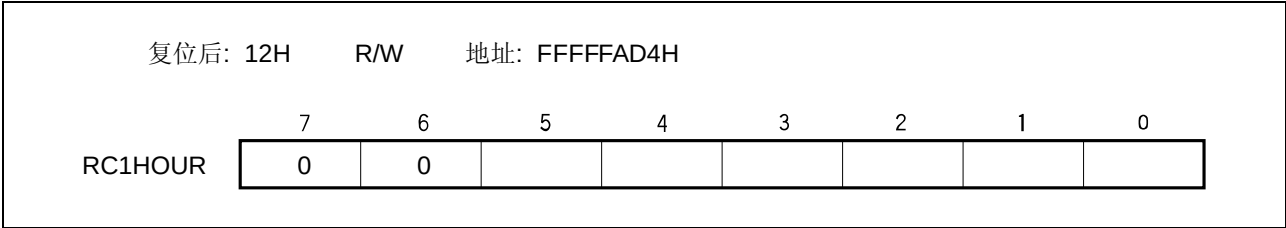


表 12-3 显示了 AMPM 位的设定值、RC1HOUR 寄存器的值以及时间之间的关系。

表 12-3. 时间数字显示

12小时显示 (AMPM 位 = 0)		24小时显示 (AMPM 位 = 1)	
时间	RC1HOUR 寄存器数值	时间	RC1HOUR 寄存器数值
0: 00 a.m.	12 H	0: 00	00H
1: 00 a.m.	01 H	1: 00	01 H
2: 00 a.m.	02 H	2: 00	02 H
3: 00 a.m.	03 H	3: 00	03 H
4: 00 a.m.	04 H	4: 00	04 H
5: 00 a.m.	05 H	5: 00	05 H
6: 00 a.m.	06 H	6: 00	06 H
7: 00 a.m.	07 H	7: 00	07 H
8: 00 a.m.	08 H	8: 00	08 H
9: 00 a.m.	09 H	9: 00	09 H
10: 00 a.m.	10 H	10: 00	10 H
11: 00 a.m.	11 H	11: 00	11 H
0: 00 p.m.	32 H	12: 00	12 H
1: 00 p.m.	21 H	13: 00	13 H
2: 00 p.m.	22 H	14: 00	14 H
3: 00 p.m.	23 H	15: 00	15 H
4: 00 p.m.	24 H	16: 00	16 H
5: 00 p.m.	25 H	17: 00	17 H
6: 00 p.m.	26 H	18: 00	18 H
7: 00 p.m.	27 H	19: 00	19 H
8: 00 p.m.	28 H	20: 00	20 H
9: 00 p.m.	29 H	21: 00	21 H
10: 00 p.m.	30 H	22: 00	22 H
11: 00 p.m.	31 H	23: 00	23 H

当 AMPM 位为 0 时 RC1HOUR 寄存器的值按 12 小时的格式显示，当 AMPM 位为 1 时，按 24 小时的格式显示。

以 12 小时显示时，RCHOUR 的第五位表示 a.m.或 p.m.: 0 表示中午 (a.m.) 之前，1 表示中午或下午 (p.m.)。

(9) 天计数寄存器 (RC1DAY)

RC1DAY 寄存器是 8 位寄存器，从 1 到 31（十进制）中取值，表示天的计数值。

当小时计数器溢出时其向上计数。

该计数器进行以下计数：

- 01 到 31（一月、三月、五月、七月、八月、十月、十二月）
- 01 到 30（四月、六月、九月、十一月）
- 01 到 29（闰年中的二月）
- 01 到 28（正常年中的二月）

当数据写入该寄存器时，其先被写入缓存，最多两个时钟（32.768 kHz）过后被写入计数器。给该寄存器以 BCD 码的格式设置一个 00 到 31 的十进制数。

该寄存器可按 8 位为单位进行读写。

复位后本寄存器被设置为 01H。

注意事项 禁止给 RC1DAY 寄存器设置 01 至 31 外的值。禁止设置以上提到的计数范围外的值，比如“二月 30”。

备注 当读取或写入 RC1DAY 寄存器时，参见 12.4.1 初始设置，12.4.2 时钟操作期间重写各计数器以及 12.4.3 设置操作期间读取各计数器。

复位后: 01H R/W 地址: FFFFFAD6H

	7	6	5	4	3	2	1	0
RC1DAY	0	0						

(10) 星期计数寄存器 (RC1WEEK)

RC1WEEK 寄存器是 8 位寄存器，从 0 到 6（十进制）中取值，表示星期的计数值。

它与天计数器同步向上计数。

当数据写入该寄存器时，其先被写入缓存，最多两个时钟（32.768 kHz）过后被写入计数器。给该寄存器以 BCD 码的格式设置一个 00 到 06 的十进制数。如果设置的数值超出范围，则一个周期过后寄存器返回正常值。

该寄存器可按 8 位为单位进行读写。

复位后本寄存器被设置为 00H。

复位后: 00H		R/W	地址: FFFFFAD5H					
	7	6	5	4	3	2	1	0
RC1WEEK	0	0	0	0	0			

- 注意事项
1. 禁止给 RC1WEEK 寄存器设置 00 至 06 外的值。
 2. 与月计数寄存器和天计数寄存器相一致的值不能自动存入星期寄存器。
复位解除后务必按照下面进行设置。

星期几	周
星期天	00H
星期一	01H
星期二	02H
星期三	03H
星期四	04H
星期五	05H
星期六	06H

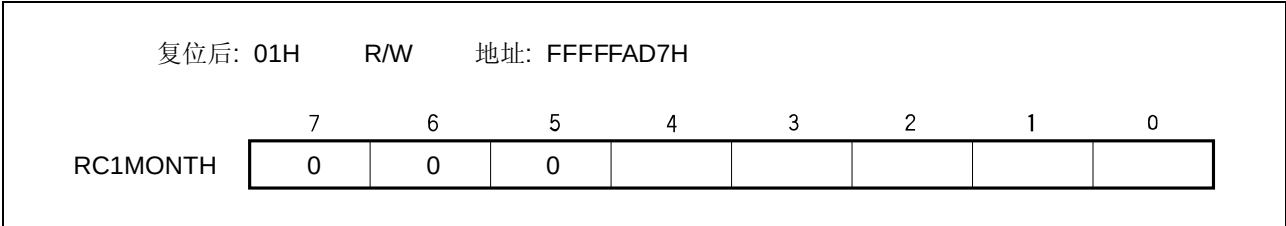
备注 当读取或写入 RC1WEEK 寄存器时，参见 12.4.1 初始设置，12.4.2 时钟操作期间重写各计数器以及 12.4.3 设置操作期间读取各计数器。

(11) 月计数寄存器 (RC1MONTH)

RC1MONTH 寄存器是 8 位寄存器，从 1 到 12（十进制）中取值，表示月的计数值。
当天计数器溢出时其向上计数。
当数据写入该寄存器时，其先被写入缓存，最多两个时钟（32.768 kHz）过后被写入计数器。给该寄存器以 BCD 码的格式设置一个 01 到 12 的十进制数。
该寄存器可按 8 位为单位进行读写。
复位后本寄存器被设置为 01H。

注意事项 禁止给 RC1MONTH 寄存器设置 01 至 12 外的值。

备注 当读取或写入 RC1MONTH 寄存器时，参见 12.4.1 初始设置，12.4.2 时钟操作期间重写各计数器以及 12.4.3 设置操作期间读取各计数器。

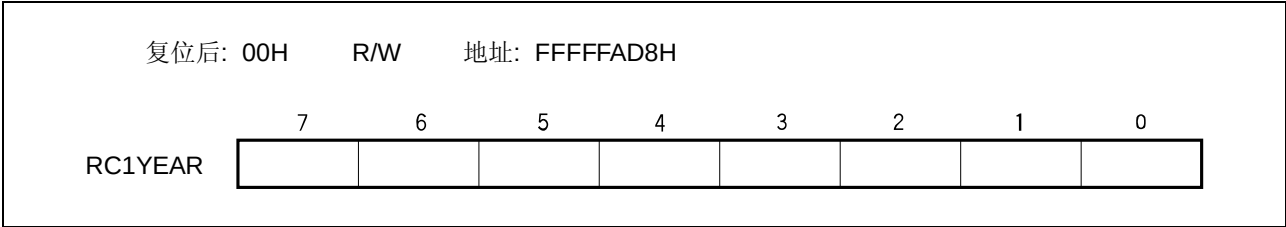


(12) 年计数寄存器 (RC1YEAR)

RC1YEAR 寄存器是 8 位寄存器，从 0 到 99（十进制）中取值，表示年的计数值。
当月计数器溢出时其向上计数。
00， 04， 08， ...， 92 和 96 值表示闰年。
当数据写入该寄存器时，其先被写入缓存，最多两个时钟（32.768 kHz）过后被写入计数器。给该寄存器以 BCD 码的格式设置一个 00 到 99 的十进制数。
该寄存器可按 8 位为单位进行读写。
复位后本寄存器被设置为 00H。

注意事项 禁止给 RC1YEAR 寄存器设置 00 至 99 外的值。

备注 当读取或写入 RC1YEAR 寄存器时，参见 12.4.1 初始设置，12.4.2 时钟操作期间重写各计数器以及 12.4.3 设置操作期间读取各计数器。



(13) 钟表错误校正寄存器 (RC1SUBU)

当钟表快或慢时，RC1SUBU 寄存器用于进行高精度的校正，其通过改变从副计数寄存器 (RSUBC) 溢出至秒计数器寄存器的值 (参考值: 7FFFH) 来实现。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

- 备注
- 1. 仅当实时计数器设置为初始值时 RC1SUBU 寄存器才可被重写。参见 12.4.1 初始设置。
 - 2. 关于钟表错误校正的详情，参见 12.4.9 实时计数器的钟表错误校正举例。

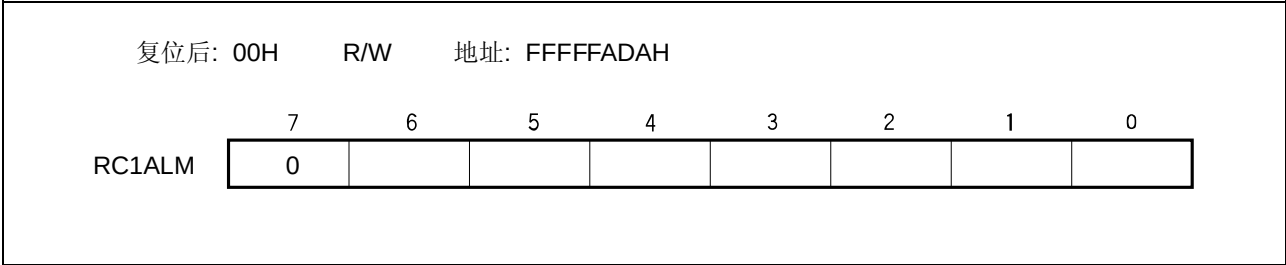
复位后: 00H R/W 地址: FFFFFAD9H

RC1SUBU	7	6	5	4	3	2	1	0
	DEV	F6	F5	F4	F3	F2	F1	F0
DEV		钟表校正时刻的设置						
0		当 RC1SEC (秒计数器) 为 00, 20或 40秒 (每 20 秒)时校正钟表错误。						
1		RC1SEC (秒计数器) 为 00 秒 (每 60秒)时校正钟表错误。						
F6		钟表错误校正值的设置						
0		RC1SUBC计数值增加用F5 至 F0 位设置的值 (正值校正)。增加值的计算表达式为: (F5至 F0 位的设置值 - 1) × 2						
1		RC1SUBC计数值减小用F5 至 F0 位设置的值 (负值校正)。减小值的计算表达式为: (F5至 F0 位的设置值的相反值 + 1) × 2						
		如果 F6 至 F0 位的值为 {1/0, 0, 0, 0, 0, 0, 1/0}, 则不执行钟表错误校正。						

(14) 分报警设置寄存器 (RC1ALM)

RC1ALM 寄存器用于设置分报警。
该寄存器可按 8 位为单位进行读写。
复位后本寄存器被设置为 00H。

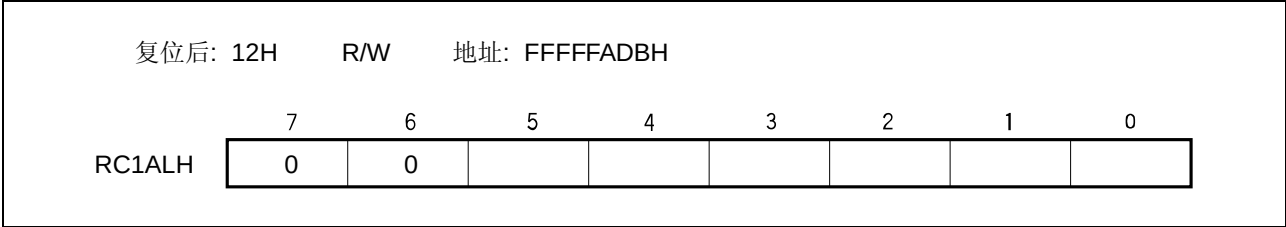
注意事项 给该寄存器以 BCD 码的格式设置一个 00 到 59 的十进制数。如果设置值超出范围，则检测不到报警。



(15) 小时报警设置寄存器 (RC1ALH)

RC1ALH 寄存器用于设置小时报警。
该寄存器可按 8 位为单位进行读写。
复位后本寄存器被设置为 12H。

- 注意事项
- 1. 以 BCD 码的格式给该寄存器设置 00 至 23、01 至 12 或 21 至 32 中的十进制数值。如果设置值超出范围，则检测不到报警。
 - 2. 如果选择 AMPM 位= 0 (12 小时的系统) RC1ALH 寄存器的位 5 表示 a.m. (0) 或 p.m. (1)。



(16) 星期报警设置寄存器 (RC1ALW)

RC1ALW 寄存器用于设置星期的报警。

该寄存器可按 8 位为单位进行读写。

复位后本寄存器被设置为 00H。

注意事项 当重写 RC1ALW 寄存器而实时计数器操作时 (RC1PWR 位= 1)，参见 12.4.5 时钟操作期间改变 INTRTC1 中断设置。

复位后: 00H		R/W		地址: FFFFFADCH				
RC1ALW	7	6	5	4	3	2	1	0
	0	RC1ALW6	RC1ALW5	RC1ALW4	RC1ALW3	RC1ALW2	RC1ALW1	RC1ALW0
RC1ALWn		报警中断星期位 (n = 0 至 6)						
0		如果 RC1WEEK = nH则不产生报警中断。						
1		如果RC1WEEK = nH则产生报警中断。 (由RC1ALM 和 RC1ALH寄存器设置时间)						

(a) 报警中断设置举例 (RC1ALM, RC1ALH 和 RC1ALW 设置举例)

表 12-4 和 12-5 显示了当星期日为 RC1WEEK = 00, 星期一为 RC1WEEK = 01, 星期二为 RC1WEEK = 02, ..., 星期六为 RC1WEEK = 06 的设置举例。

表 12-4. AMPM = 0 时报警设置举例 (RC1HOUR 寄存器 12 小时显示)

报警设置时间 \ 寄存器	RC1ALW	RC1ALH	RC1ALM
星期日, 7: 00 a.m.	01H	07H	00H
星期日/星期一, 00: 15 p.m.	03H	32H	15H
星期一/星期二/星期五, 5: 30 p.m.	26H	25H	30H
每天, 10: 45 p.m.	7FH	30H	45H

表 12-5. AMPM = 1 时报警设置举例 (RC1HOUR 寄存器 24 小时显示)

报警设置时间 \ 寄存器	RC1ALW	RC1ALH	RC1ALM
星期日, 7: 00	01H	07H	00H
星期日/星期一, 12: 15	03H	12H	15H
星期一/星期二/星期五, 17: 30	26H	17H	30H
每天, 22: 45	7FH	22H	45H

<R> (17) 预分频模式寄存器 0 (PRSM0)

PRSM0 寄存器控制实时计数器计数时钟 (f_{BRG}) 的发生。
本寄存器可按 8 位或 1 位读写。
复位后本寄存器被设置为 00H。

复位后：00H R/W 地址： FFFFF8B0H

	7	6	5	<4>	3	2	1	0
PRSM0	0	0	0	BGCE0	0	0	BGCS01	BGCS00

BGCE0	主时钟操作使能
0	禁止
1	允许

BGCS01	BGCS00	实时计数器源时钟的选择(f_{BGCS})		
			5 MHz	4 MHz
0	0	fx	200 ns	250 ns
0	1	fx/2	400 ns	500 ns
1	0	fx/4	800 ns	1 ms
1	1	fx/8	1.6 ms	2 ms

- 注意事项
1. 实时计数器工作期间禁止改变BGCS00 和 BGCS01位的值。
 2. 在设置BGCE0位为1之前设置PRSM0寄存器。
 3. 根据主时钟频率设置PRSM0 和 PRSCM0寄存器，以得到一个32.768 kHz的f_{BRG} 频率。

<R>

(18) 预分频比较寄存器 0 (PRSCM0)

PRSCM0 寄存器 为 8 位比较寄存器。
该寄存器可按 8 位为单位进行读写。
复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: FFFFF8B1H

PRSCM0

7	6	5	4	3	2	1	0
PRSCM07	PRSCM06	PRSCM05	PRSCM04	PRSCM03	PRSCM02	PRSCM01	PRSCM00

注意事项

1. 实时计数器工作期间禁止重写PRSCM0寄存器。

2. 在设置PRSM0.BGCE0位为1之前设置PRSCM0寄存器。

3. 根据主时钟频率设置PRSM0 和 PRSCM0寄存器，以得到一个32.768 kHz的fBRG 频率。

关于 fBRG 的计算如下：

$f_{BRG} = f_{BGCS}/2N$

备注

fBGCS:

N:

PRSM0 寄存器设置的钟表定时器源时钟

PRSCM0 寄存器的设定值= 1 至 256

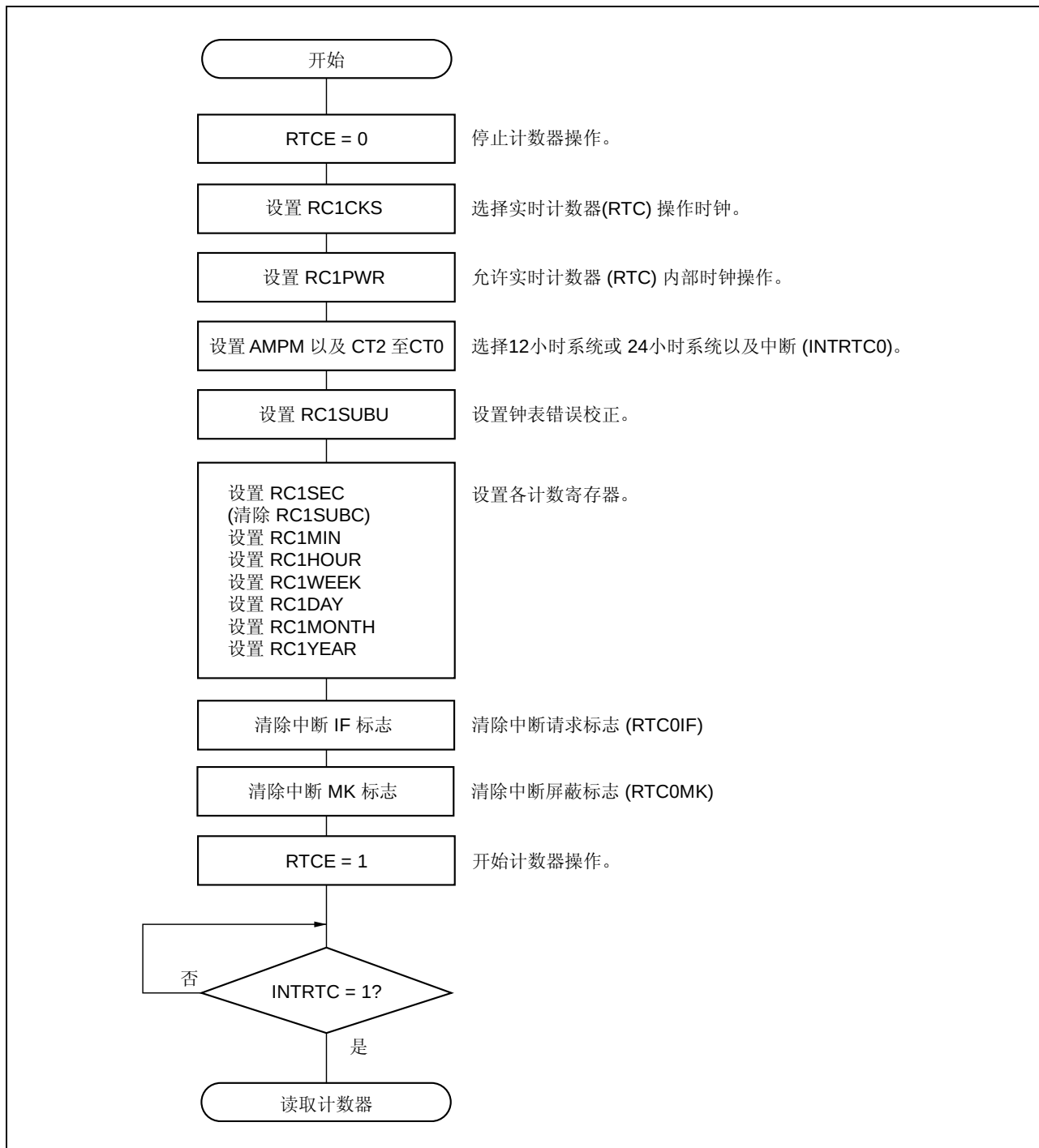
然而，PRSCM0 寄存器设置为 00H 时 N = 256。

12.4 操作

12.4.1 初始设置

操作钟表功能和执行固定周期中断操作时进行初始设置。

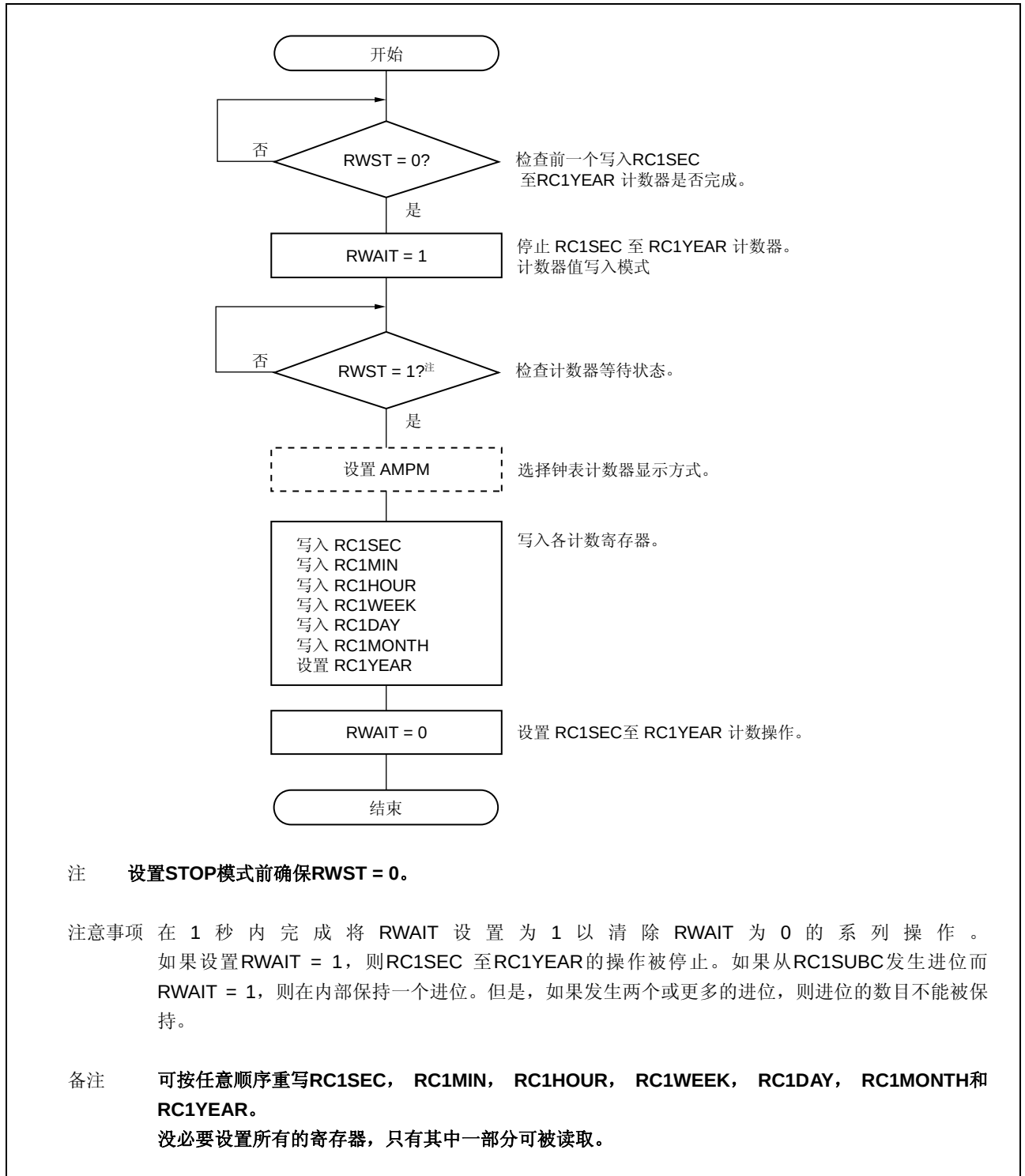
图 12-2.初始设置步骤



12.4.2 时钟操作期间重写各计数器

时钟操作期间（RC1PWR = 1）重写各计数器（RC1SEC， RC1MIN， RC1HOUR， RC1WEEK， RC1DAY， RC1MONTH， RC1YEAR）时按如下进行设置。

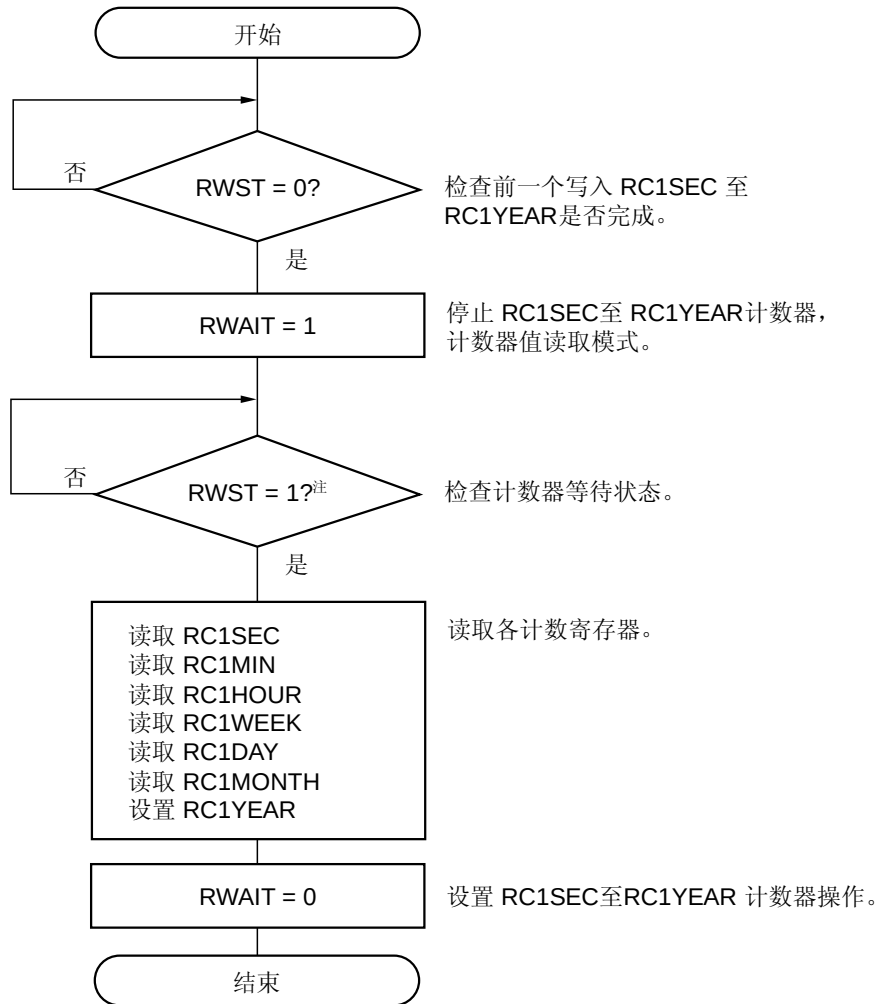
图 12-3. 时钟操作期间重写各计数器



12.4.3 时钟操作期间读取各计数器

时钟操作期间（RC1PWR = 1）读取各计数器（RC1SEC， RC1MIN， RC1HOUR， RC1WEEK， RC1DAY， RC1MONTH， RC1YEAR）时按如下进行设置。

图 12-4. 时钟操作期间读取各计数器



注 设置STOP模式前确保RWST = 0。

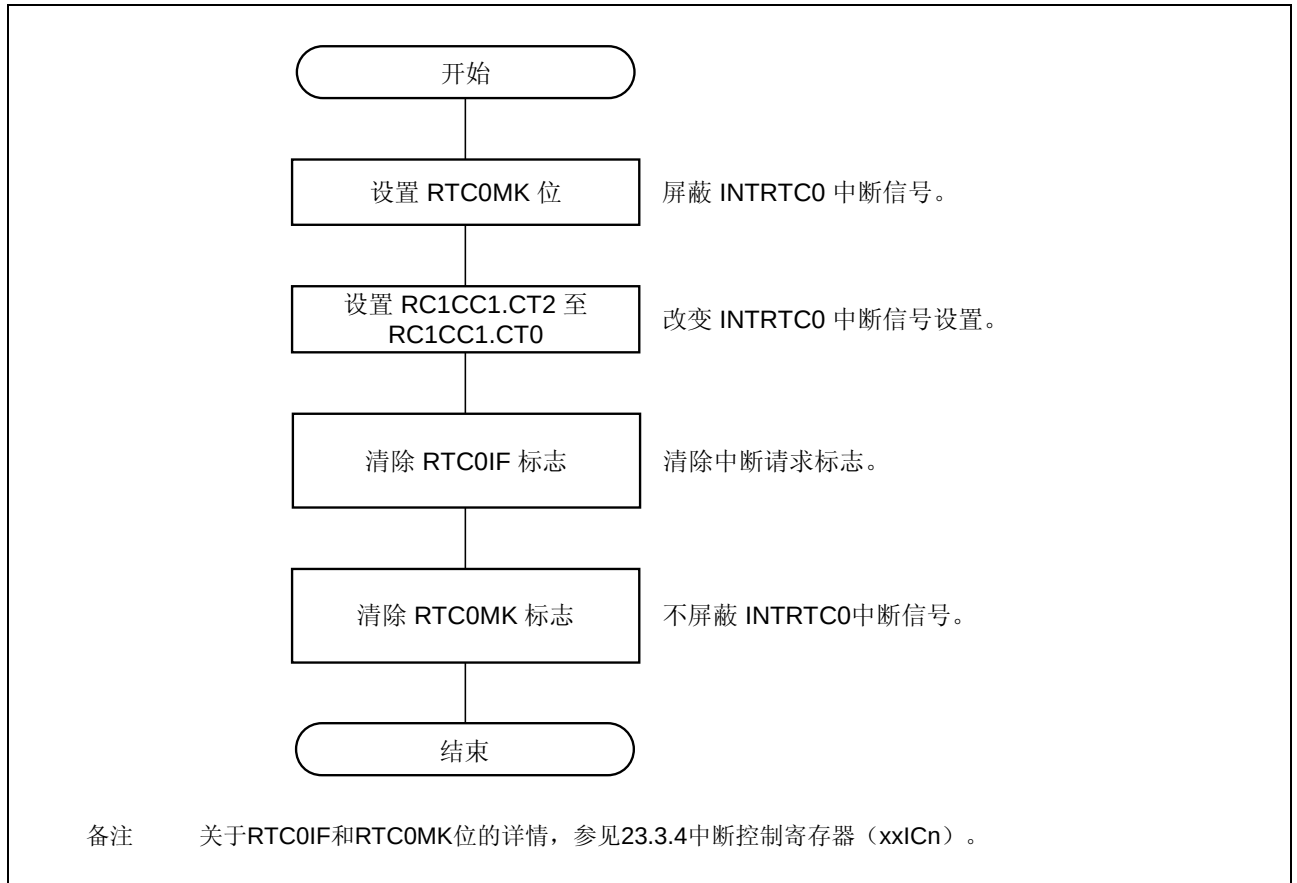
注意事项 在 1 秒内完成将 RWAIT 设置为 1 以清除 RWAIT 为 0 的系列操作。如果设置 RWAIT = 1，则 RC1SEC 至 RC1YEAR 的操作被停止。如果从 RC1SUBC 发生进位而 RWAIT = 1，则在内部保持一个进位。但是，如果发生两个或更多的进位，则进位的数目不能被保持。

备注 可按任意顺序读取 RC1SEC， RC1MIN， RC1HOUR， RC1WEEK， RC1DAY， RC1MONTH 和 RC1YEAR。
没必要设置所有的寄存器，只有其中一部分可被读取。

12.4.4 时钟操作期间改变INTRTC0中断设置

如果在实时计数器时钟操作（PC1PWR = 1）时改变 INTRTC0 中断（固定周期中断）信号的设置，则 INTRCT0 中断波形可能有毛刺并且有可能输出计划外的信号。为屏蔽这些毛刺，在时钟操作（RC1PWR = 1）期间改变 INTRTC0 中断信号的设置时需按以下进行。

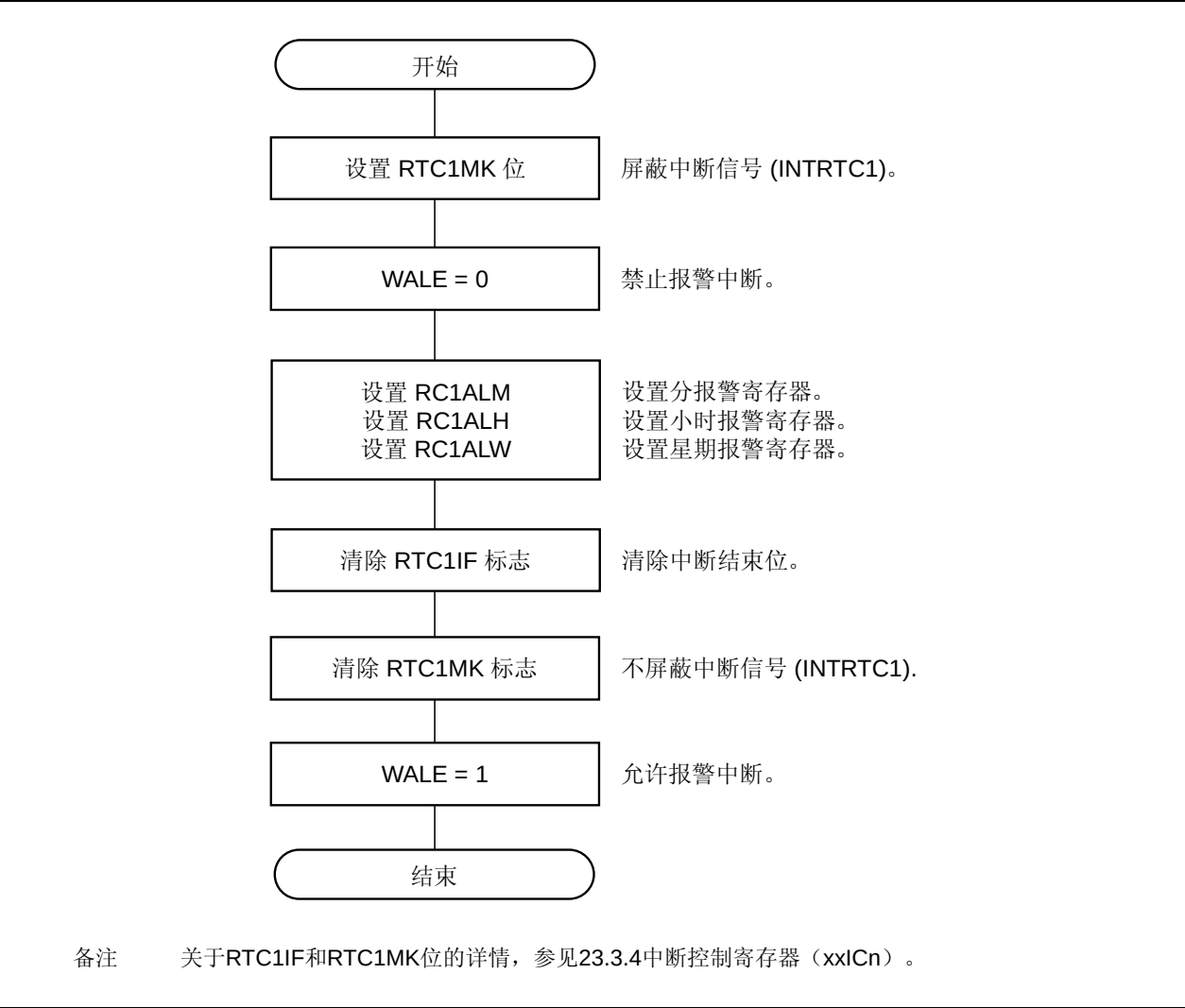
图 12-5. 时钟操作期间改变 INTRTC0 中断设置



12.4.5 时钟操作期间改变INTRTC1 中断设置

如果在实时计数器时钟操作（PC1PWR = 1）时改变 INTRTC1 中断（报警中断）信号的设置，则 INTRCT1 中断波形可能有毛刺并且有可能输出计划外的信号。为屏蔽这些毛刺，在时钟操作（PC1PWR = 1）期间改变 INTRTC1 中断信号的设置时需按以下进行。

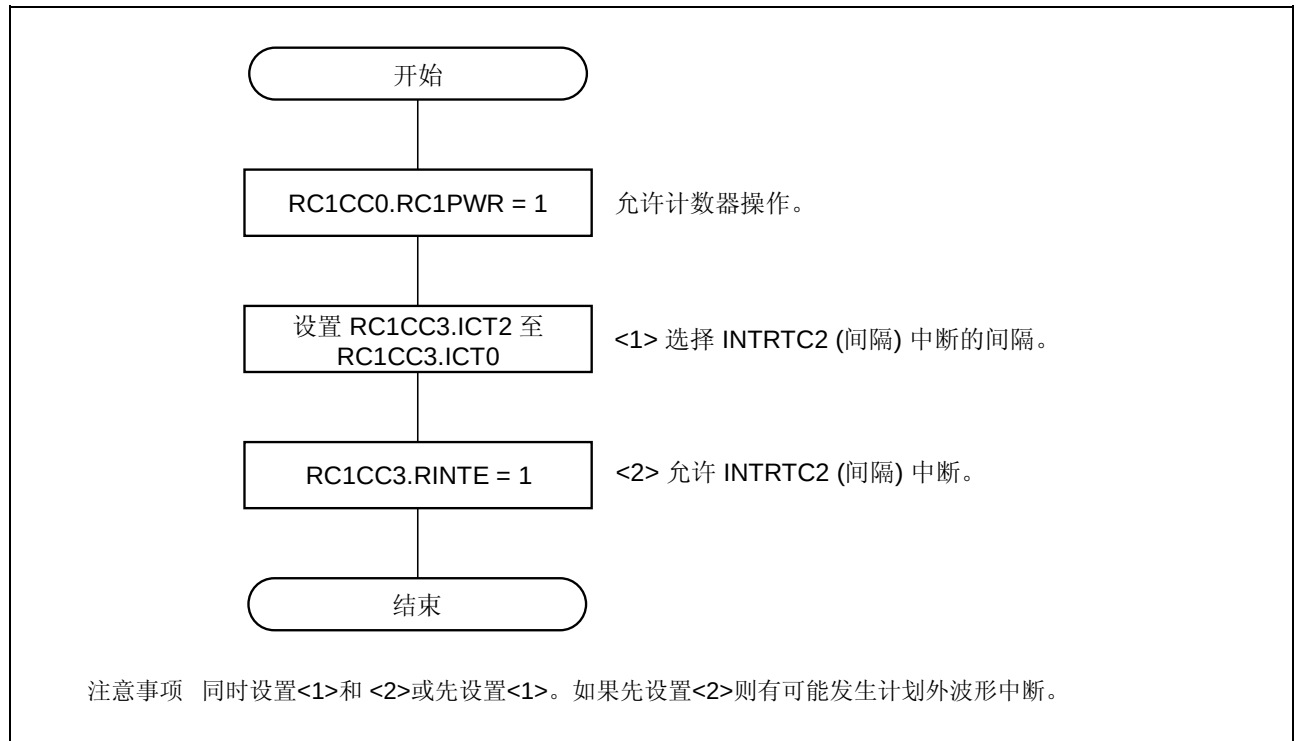
图 12-6. 时钟操作期间改变 INTRTC1 中断设置



12.4.6 初始化INTRTC2 中断设置

按如下设置 INTRTC1 中断（间隔中断）。

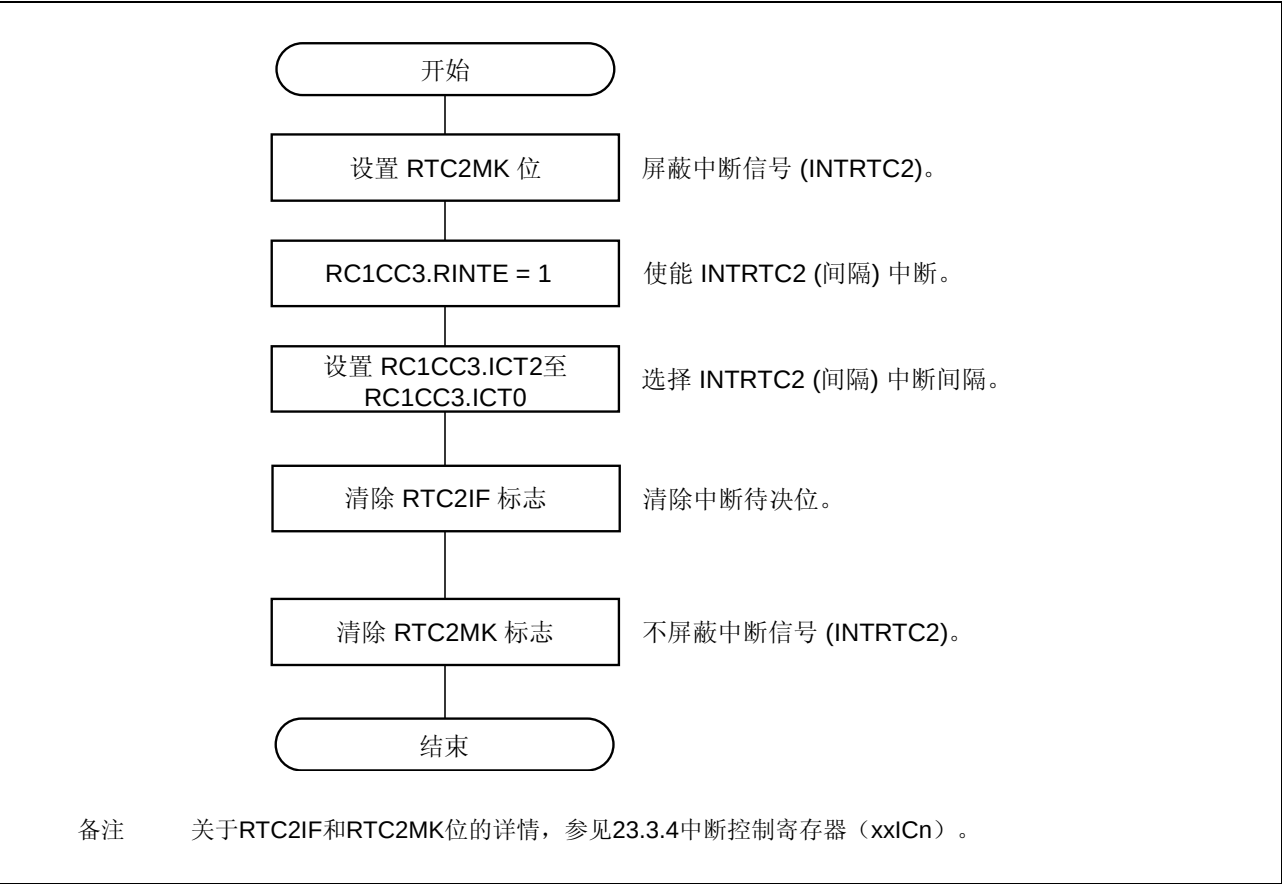
图 12-7. INTRTC2 中断设置



12.4.7 时钟操作期间改变INTRTC2 中断设置

如果在实时计数器时钟操作（PC1PWR = 1）时改变 INTRTC2 中断（间隔中断）信号的设置，则 INTRTC2 中断波形可能有毛刺并且有可能输出计划外的信号。为屏蔽这些毛刺，在时钟操作（PC1PWR = 1）期间改变 INTRTC2 中断信号的设置时需按以下进行。

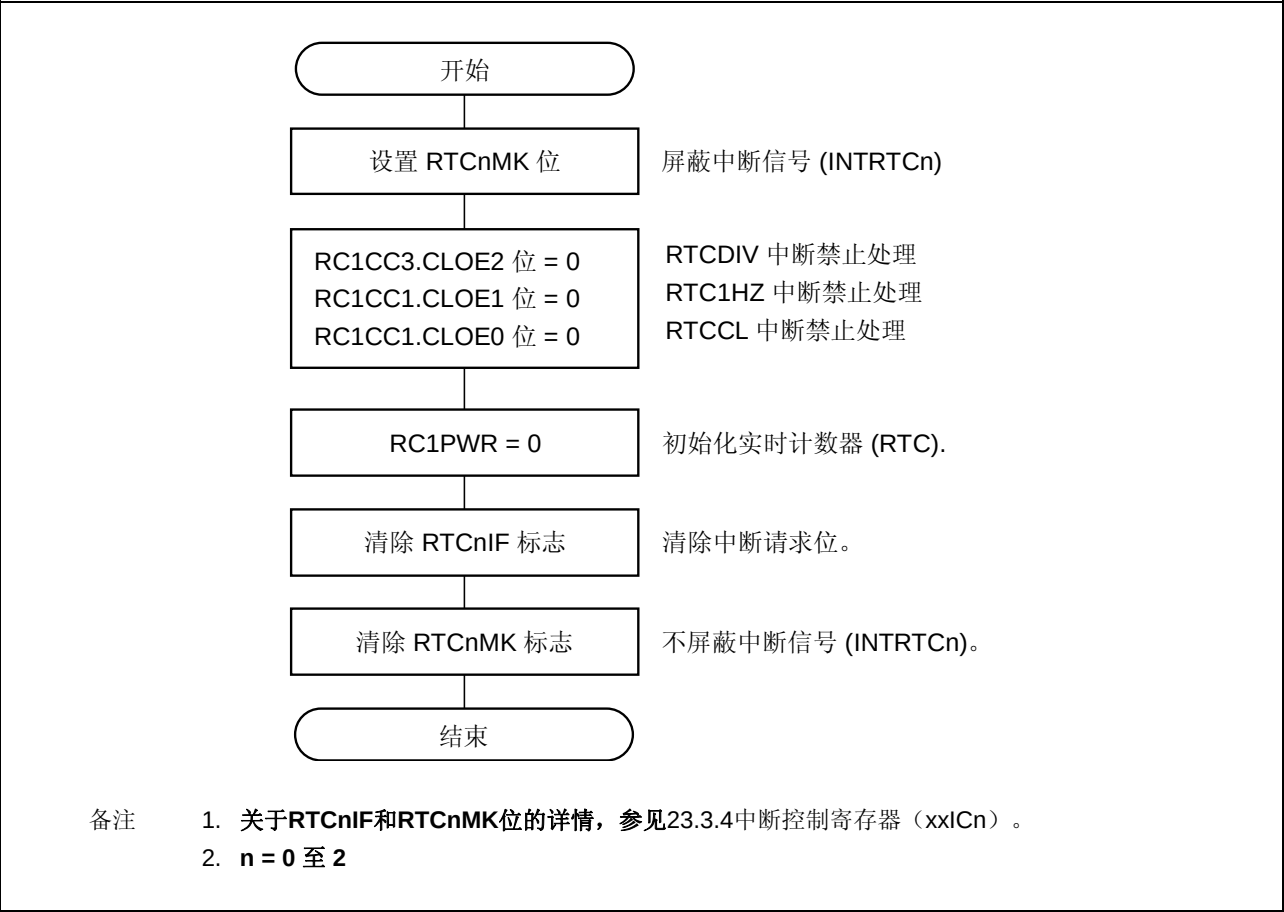
图 12-8. 时钟操作期间改变 INTRTC2 中断设置



12.4.8 初始化实时计数器

初始化实时计数器的步骤如下：

图 12-9. 初始化实时计数器



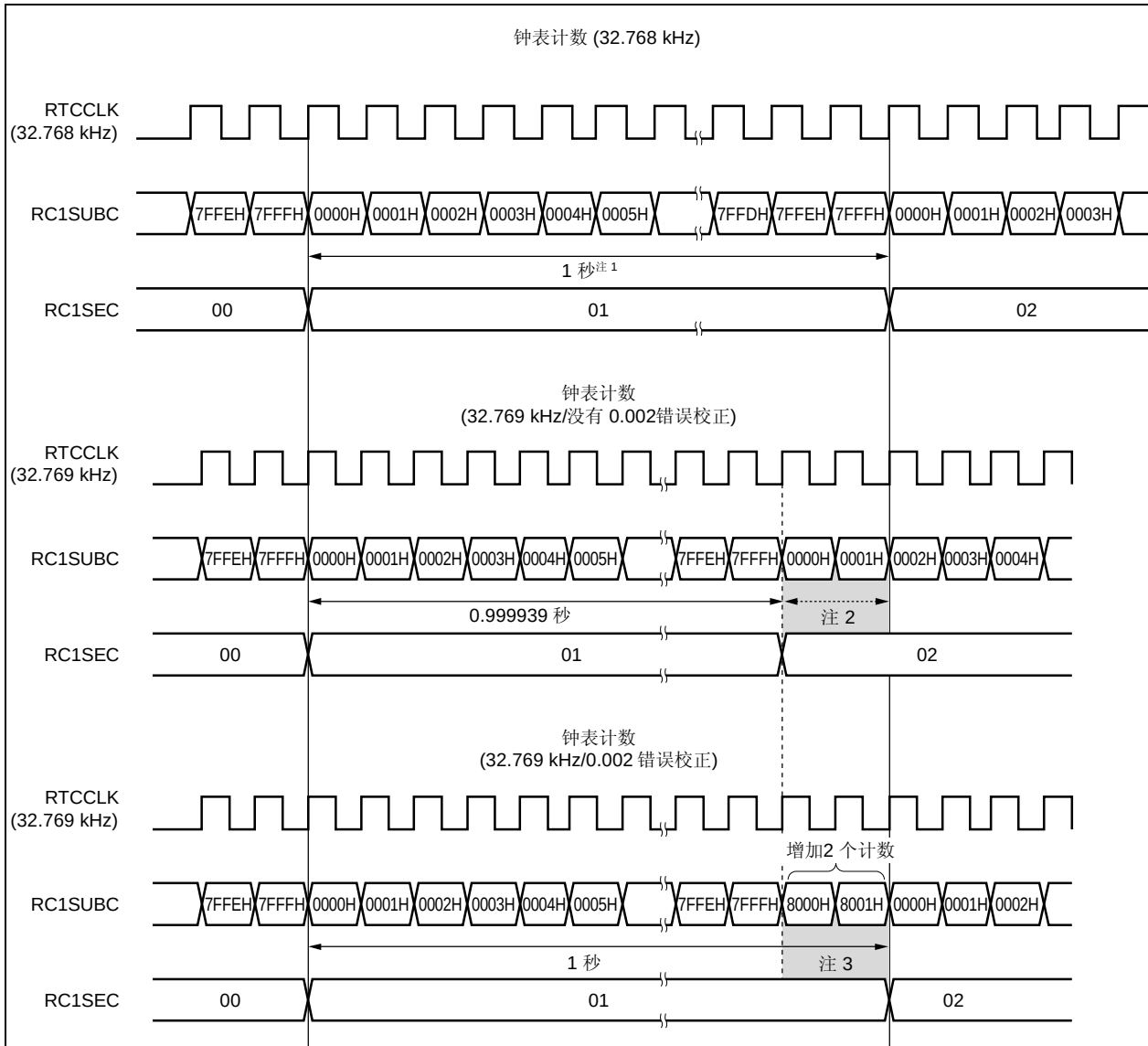
12.4.9 实时计数器的钟表错误校正举例

钟表错误校正功能校正连接 V850ES/Jx3H 的振荡器的振荡频率中的偏差。

这的偏差指稳定状态的偏差，即设计振荡器时频率中的偏差。

设计系统时当错误发生在应该为 32.768 kHz 的输入时钟但连接了一个 32.769 kHz 的振荡器时的时序流程和进行校正的 RC1SUBC 和 RC1SEC 的计数操作如下所示：

图 12-10.钟表错误校正举例



- 注
1. 通过用32.768 kHz时钟计数32,768 (0000H至7FFFH) 来计数1秒。
 2. 用32.770 kHz时钟计数32,768 (0000H至7FFFH) 导致 $32,768/32,770 = 0.999939$ 秒。
每个月产生 $(32,768/32,770 - 1) \times 60$ (秒) $\times 60$ (分) $\times 24$ (小时) $\times 30$ (天) = -158.2 秒的错误。
 3. 用32.770 kHz时钟计数32,770 (0000H至8001H)，则计数1秒不发生错误。

如图 12-10 所示，如果在振荡器处发生比 32.768 kHz 快的正值的错误，则通过增加 RC1SUBC 计数值，钟表可进行精确计数。同理，如果在在振荡器处发生比 32.768 kHz 慢的负值的错误，则通过减小 RC1SUBC 计数值，钟表可进行精确计数。

RC1SUBC 校正值由 RC1SUBU.F6 至 RC1SUBU.F0 位决定。

F6 位用于决定增加还是减小 RC1SUBC 而 F5 至 F0 位用于决定 RC1SUBC 值。

(1) 增加 RC1SUBC 寄存器的计数值

将 F6 位设置为 0，通过用 F5 至 F0 位设置的值增加 RC1SUBC 计数值。

增加值的计算表达式为： $(F5 \text{ 至 } F0 \text{ 位的值} - 1) \times 2$

[增加 RC1SUBC 计数值举例：F6 位 = 0]

当 15H (010101B) 设置给 F5 至 F0 位时

$(15H - 1) \times 2 = 40$ (RC1SUBC 计数值增加 40)

RC1SUBC 计数值 = 32, $768 + 40 = 808$

(2) 减小 RC1SUBC 寄存器的计数值

将 F6 位设置为 1，通过用 F5 至 F0 位设置值的相反值减小 RC1SUBC 计数值。

减小值的计算表达式为： $(F5 \text{ 至 } F0 \text{ 位值的相反值} + 1) \times 2$

[减小 RC1SUBC 计数值举例：F6 位 = 1]

当 15H (010101B) 设置给 F5 至 F0 位

15H (010101B) 的相反值 = 2AH (101010B)

$(2AH + 1) \times 2 = 86$ (RC1SUBC 计数值减小 86)

RC1SUBC 计数值 = 32, $768 - 86 = 682$

(3) DEV 位

DEV 位决定允许用 F6 至 F0 位进行设置的时间。

F6 至 F0 位设置的数值反映在下表所列的时刻上，而不是每次都反映到 RC1SUBC 计数值。

表 12-6. DVE 位设置

DEV 位的数值	将值反映到 RC1SUBC 的时刻
0	当 RC1SEC 是 00, 20 或 40 秒。
1	当 RC1SEC 是 00 秒。

当 0010101B 设置给 F6 至 F0 位时的举例]

- 如果 DEV 位为 0
RC1SUBC 计数值在 00, 20 或 40 秒为 32, 808。
否则, 其为 32, 768。
- 如果 DEV 位为 1
The RC1SUBC 计数值在 00 秒为 32, 808。
否则, 其为 32, 768。

按照以上的描述, 为了使 RC1SUBC 计数值与谐振器的偏差幅度相匹配, RC1SUBC 计数值每 20 秒或 60 秒被校正一次, 而不是每秒都校正,
谐振器频率可实际校正的范围如下所示:

- 如果 DEV 位为 0: 32.76180000 kHz 至 32.77420000 kHz
- 如果 DEV 位为 1: 32.76593333 kHz 至 32.77006667 kHz

当 DEV 位为 0 时可校正频率的范围是 DEV 位为 1 时范围的三倍。

但是, 当 DEV 位是 1 时设置频率的精确度是 DEV 位为 0 时精确度的三倍。

表 12-7 和 12-8 显示了 DEV 以及 F6 至 F0 位的设定值与相应的可校正频率。

表 12-7. DEV 位 = 0 时可校正的频率范围

F6	F5 至 F0	RC1SUBC校正值	连接时钟的频率（包括稳定状态偏差）
0	000000	不校正	—
0	000001	不校正	—
0	000010	每20秒RC1SUBC计数值增加2	32.76810000 kHz
0	000011	每20秒RC1SUBC计数值增加4	32.76820000 kHz
0	000100	每20秒RC1SUBC计数值增加6	32.76830000 kHz
⋮			
0	111011	每20秒RC1SUBC计数值增加120	32.77400000 kHz
0	111110	每20秒RC1SUBC计数值增加122	32.77410000 kHz
0	111111	每20秒RC1SUBC计数值增加124	32.77420000 kHz（上限）
1	000000	不校正	—
1	000001	不校正	—
1	000010	每20秒RC1SUBC计数值减小124	32.76180000 kHz（下限）
1	000011	每20秒RC1SUBC计数值减小122	32.76190000 kHz
1	000100	每20秒RC1SUBC计数值减小120	32.76200000 kHz
⋮			
1	11011	每20秒RC1SUBC计数值减小6	32.76770000 kHz
1	11110	每20秒RC1SUBC计数值减小4	32.76780000 kHz
1	11111	每20秒RC1SUBC计数值减小2	32.76790000 kHz

表 12-8. DEV 位 = 1 时可校正的频率范围

F6	F5至F0	RC1SUBC 校正值	连接时钟的频率（包括稳定状态偏差）
0	000000	不校正	—
0	000001	不校正	—
0	000010	每60秒RC1SUBC计数值增加2	32.76803333 kHz
0	000011	每60秒RC1SUBC计数值增加4	32.76806667 kHz
0	000100	每60秒RC1SUBC计数值增加6	32.76810000 kHz
⋮			
0	111011	每60秒RC1SUBC计数值增加120	32.77000000 kHz
0	111110	每60秒RC1SUBC计数值增加122	32.77003333 kHz
0	111111	每60秒RC1SUBC计数值增加124	32.77006667 kHz（上限）
1	000000	不校正	—
1	000001	不校正	—
1	000010	每60秒RC1SUBC计数值减小124	32.76593333 kHz（下限）
1	000011	每60秒RC1SUBC计数值减小122	32.76596667 kHz
1	000100	每60秒RC1SUBC计数值减小120	32.76600000 kHz
⋮			
1	11011	每60秒RC1SUBC计数值减小6	32.76790000 kHz
1	11110	每60秒RC1SUBC计数值减小4	32.76793333 kHz
1	11111	每60秒RC1SUBC计数值减小2	32.76796667 kHz

第十三章 看门狗定时器 2 的功能

13.1 功能

看门狗定时器 2 具有以下功能：

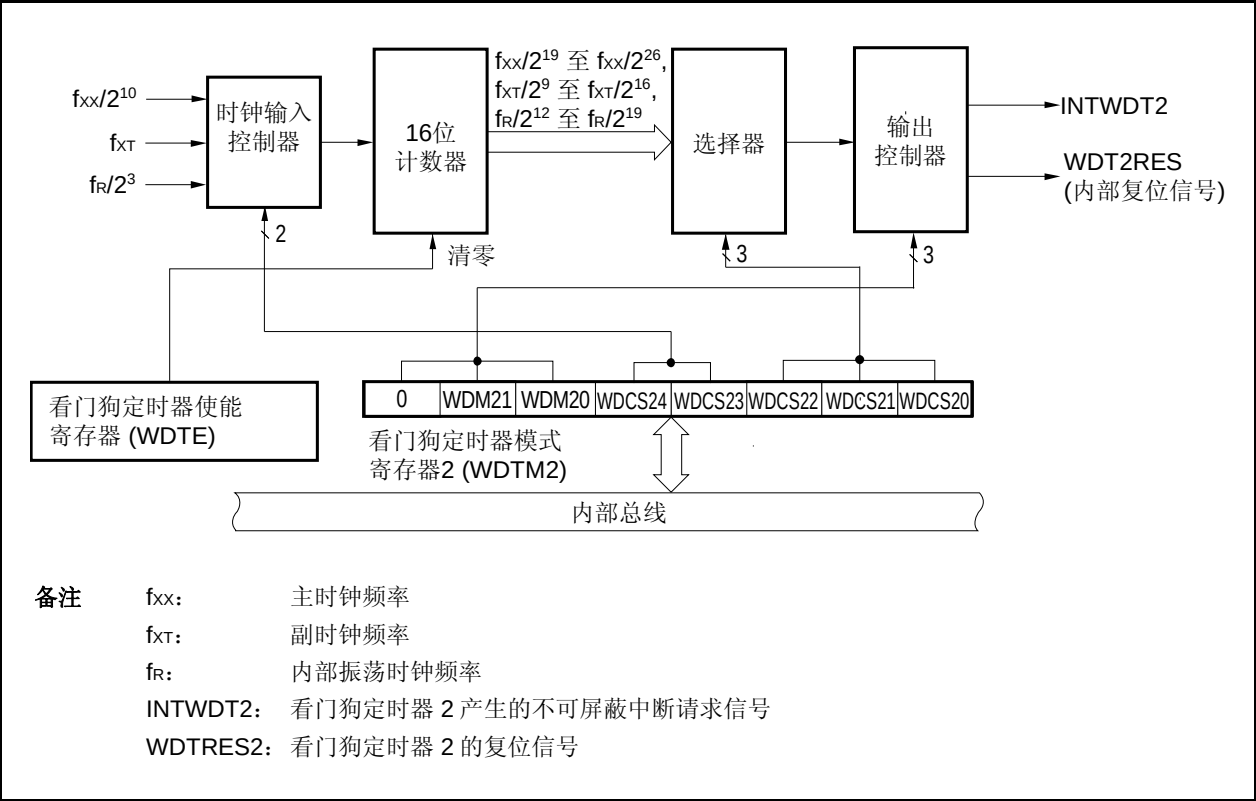
- 默认启动的看门狗定时器^{註1}
 - 复位模式：看门狗定时器 2 溢出后执行复位操作（产生 WDT2RES 信号）
 - 不可屏蔽中断请求模式：看门狗定时器 2 溢出后执行 NMI 操作（产生 INTWDT2 信号）^{註2}
- 可选择主时钟、内部振荡时钟及副时钟之一作为时钟源

- 注**
1. 在复位模式下，当复位解除后看门狗定时器 2 会自动启动。
当不使用看门狗定时器 2 时，或者在系统复位前由该功能将其停止，或者在下一个间隔时间内清除一次看门狗定时器 2 并停止工作。
为了确认操作的正确性，即使在不需要改变默认设置（复位模式，间隔时间： $f_R/2^{19}$ ）的情况下，也要对 WDTM2 寄存器执行一次写入操作。
 2. 关于由不可屏蔽中断请求信号（INTWDT2）进入不可屏蔽中断服务程序，参见 23.2.2（2）自 INTWDT2 的信号。

13.2 配置

看门狗定时器 2 的功能框图如下所示：

图 13-1. 看门狗定时器 2 的功能框图



看门狗定时器 2 包括以下硬件：

表 13-1. 看门狗定时器 2 的配置

项目	配置
控制寄存器	看门狗定时器模式寄存器 2（WDTM2） 看门狗定时器使能寄存器（WDTE）

13.3 寄存器

(1) 看门狗定时器模式寄存器 2 (WDTM2)

WDTM2 寄存器用以设置看门狗定时器 2 的溢出时间和工作时钟。

该寄存器可按 8 位为单位进行读写。该寄存器能够读取任意次，但是仅能在复位解除后写入一次。

复位将该寄存器设置为 67H。

注意事项 下列状态下禁止访问 WDTM2 寄存器。有关详细说明，参见 3.4.9 (2) 访问片上专用外围 I/O 寄存器。

- 当 CPU 工作于副时钟，而且主时钟振荡停止时
- 当 CPU 工作于内部振荡时钟时

复位后: 67H R/W 地址: FFFFF6D0H

	7	6	5	4	3	2	1	0
WDTM2	0	WDM21	WDM20	WDSCS24	WDSCS23	WDSCS22	WDSCS21	WDSCS20

WDM21	WDM20	看门狗定时器2操作模式的选择
0	0	停止操作
0	1	不可屏蔽中断请求模式 (INTWDT2信号产生)
1	—	复位模式 (WDT2RES 信号产生)

- 注意事项**
1. 关于 WDSCS20 至 WDSCS24 位的细节，参见表 13-2 看门狗定时器 2 的时钟选择。
 2. 虽然看门狗定时器 2 可通过终止内部振荡器的运行来停止，还是要将 WDTM2 寄存器清零以确保安全停止该定时器（为了避免由于误写入操作而选择主时钟或副时钟）。
 3. 复位后，如果对 WDTM2 寄存器执行了两次写入操作，那么将强制产生溢出信号且计数器复位。
 4. 为了有意产生溢出信号，可写数据至 WDTM2 寄存器两次，或者写入 WDTE 寄存器一次非“ACH”。
- 然而，当看门狗定时器 2 设置为停止运行时，无论写入数据至 WDTM2 寄存器两次，还是写入 WDTE 寄存器一次非“ACH”，都不会产生溢出信号。
5. 为了停止看门狗定时器 2，将 RCM.RSTOP 位设置为 1（为了停止内部振荡器），并且将 00H 写入 WDTM2 寄存器。如果 RCM.RSTOP 位不能设置为 1，那么就将 WDSCS23 位设为 1（ $2^{\text{f}_{\text{xx}}}$ 被选择且该时钟可以在 IDLE1、IDLW2、sub-IDLE 和副时钟操作模式下停止）。

表 13-2. 看门狗定时器 2 时钟的选择

WDCS24	WDCS23	WDCS22	WDCS21	WDCS20	选择时钟	100 kHz (MIN.)	220 kHz (TYP.)	400 kHz (MAX.)
0	0	0	0	0	$2^{12}/f_R$	41.0 ms	18.6 ms	10.2 ms
0	0	0	0	1	$2^{13}/f_R$	81.9 ms	37.2 ms	20.5 ms
0	0	0	1	0	$2^{14}/f_R$	163.8 ms	74.5 ms	41.0 ms
0	0	0	1	1	$2^{15}/f_R$	327.7 ms	148.9 ms	81.9 ms
0	0	1	0	0	$2^{16}/f_R$	655.4 ms	297.9 ms	163.8 ms
0	0	1	0	1	$2^{17}/f_R$	1, 310.7 ms	595.8 ms	327.7 ms
0	0	1	1	0	$2^{18}/f_R$	2, 621.4 ms	1191.6 ms	655.4 ms
0	0	1	1	1	$2^{19}/f_R$ (默认值)	5, 242.9 ms	2383.1 ms	1, 310.7 ms
						$f_{XX} = 24 \text{ MHz}$	$f_{XX} = 32 \text{ MHz}$	$f_{XX} = 48 \text{ MHz}$
0	1	0	0	0	$2^{19}/f_{XX}$	21.8 ms	16.4 ms	10.9 ms
0	1	0	0	1	$2^{20}/f_{XX}$	43.7 ms	32.8 ms	21.8 ms
0	1	0	1	0	$2^{21}/f_{XX}$	87.4 ms	65.5 ms	43.7 ms
0	1	0	1	1	$2^{22}/f_{XX}$	174.8 ms	131.1 ms	87.4 ms
0	1	1	0	0	$2^{23}/f_{XX}$	349.5 ms	262.1 ms	174.8 ms
0	1	1	0	1	$2^{24}/f_{XX}$	699.1 ms	524.3 ms	349.5 ms
0	1	1	1	0	$2^{25}/f_{XX}$	1398.1 ms	1048.6 ms	699.1 ms
0	1	1	1	1	$2^{26}/f_{XX}$	2796.2 ms	2097.2 ms	1398.1 ms
						$f_{XT} = 32.768 \text{ kHz}$		
1	×	0	0	0	$2^9/f_{XT}$	15.625 ms		
1	×	0	0	1	$2^{10}/f_{XT}$	31.25 ms		
1	×	0	1	0	$2^{11}/f_{XT}$	62.5 ms		
1	×	0	1	1	$2^{12}/f_{XT}$	125 ms		
1	×	1	0	0	$2^{13}/f_{XT}$	250 ms		
1	×	1	0	1	$2^{14}/f_{XT}$	500 ms		
1	×	1	1	0	$2^{15}/f_{XT}$	1, 000 ms		
1	×	1	1	1	$2^{16}/f_{XT}$	2, 000 ms		

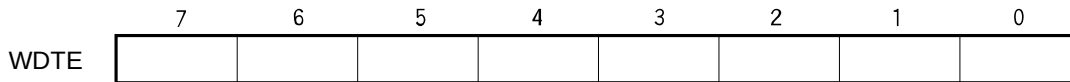
(2) 看门狗定时器使能寄存器 (WDTE)

通过将“ACH”写入 WDTE 寄存器来清除看门狗定时器 2 的计数值，并且重新开始计数。

WDTE 寄存器可以按 8 位宽度进行读写。

复位输入将该寄存器设为 9AH。

复位后: 9AH R/W 地址: FFFFF6D1H



- 注意事项**
1. 当非“ACH”值写入 WDTE 寄存器时，强制输出一个溢出信号。
 2. 当对 WDTE 寄存器执行 1 位存储器操作指令时，强制输出一个溢出信号。
 3. 为了有意产生溢出信号，至 WDTE 寄存器写入非“ACH”数据一次，或者至 WDTM2 寄存器写入数据两次。
然而，当看门狗定时器 2 被设置为停止运行，无论写数据至 WDTM2 寄存器两次，还是写入 WDTE 寄存器一次“ACH”的数据，都不产生溢出信号。
 4. 寄存器的读出值总是“9AH”（与写入值“ACH”不同）。

13.4 操作

在复位模式下，当复位解除后看门狗定时器 2 会自动启动。

使用字节访问复位后，仅能够对 WDTM2 寄存器写入一次。要使用看门狗定时器 2，应执行字节存储器处理指令将工作模式以及间隔时间写入 WDTM2 寄存器。之后，将无法停止看门狗定时器的操作。

WDTM2.WDCS24 至 WDTM2.WDCS20 位用于选择看门狗定时器 2 循环检测时间间隔。

将 ACH 写入 WDTE 寄存器来清除看门狗定时器 2 的计数值，并且重新开始计数操作。计数操作开始后，在循环检测时间间隔内将 ACH 写入 WDTE。

如果在 ACH 未写入 WDTE 寄存器时间间隔时间期满，那么根据 WDTM2.WDM21 和 WDTM2.WDM20 位的设定值，将产生一个复位信号 (WDT2RES) 或 不可屏蔽中断请求信号 (INTWDT2)。

当 WDTM2.WDM21 位设置位 1 (复位模式) 时，复位或待机状态解除后，如果在振荡稳定期间产生 WDT 溢出，那么将不发生内部重置，并且 CPU 时钟将切换到内部振荡时钟。

若不使用看门狗定时器 2，则将 00H 写入 WDTM2 寄存器。

关于设置不可屏蔽中断请求模式时的不可屏蔽中断服务，参见 23.2.2 (2) (2) 自 INTWDT2 的信号。

第十四章 实时输出功能（RTO）

14.1 功能

实时输出功能通过硬件将提前设置到 RTBL0 和 RTBH0 寄存器的数据传送至输出锁存，在发生定时器中断的同时将数据输出到外部设备中。数据传输到外部设备使用的引脚构成所谓实时输出（RTO）端口。

因为用 RTO 能输出无抖动信号，所以适合用于控制步进电机。

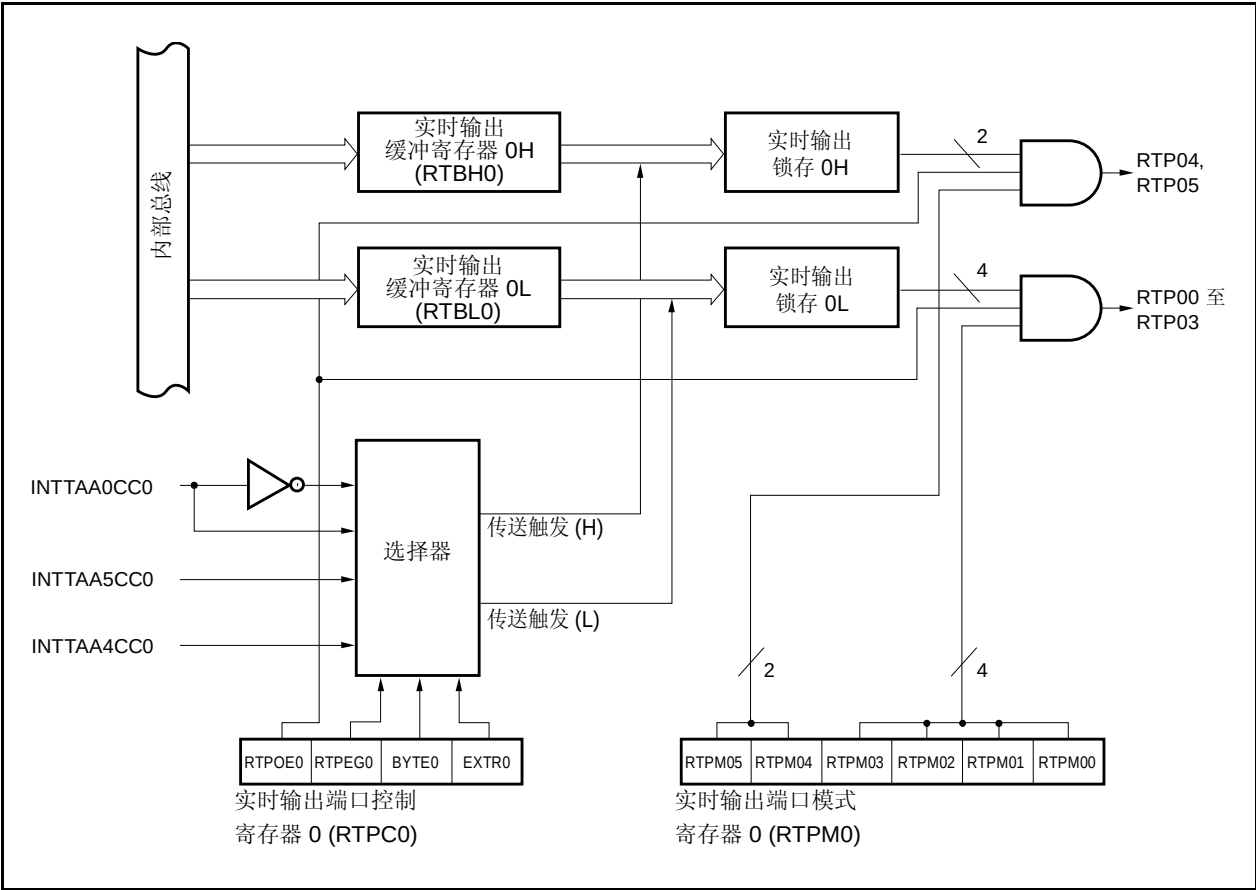
在 V850ES/JG3-H 和 V850ES/JH3-H 中，提供了一个 6 位的实时输出端口通道。

实时输出端口能以位为单位设置为端口模式或实时输出端口模式。

14.2 配置

RTO 的功能框图如下所示：

图 14-1. RTO 的功能框图



RTO 包括以下硬件：

表 14-1. RTO 的配置

项目	配置
寄存器	实时输出缓冲寄存器 0L, 0H (RTBL0, RTBH0)
控制寄存器	实时输出端口模式寄存器 0 (RTPM0) 实时输出端口控制寄存器 0 (RTPC0)

(1) 实时输出缓冲寄存器 0L, 0H (RTBL0, RTBH0)

RTBL0 和 RTBH0 寄存器为 4 位寄存器，用于保持预先设置的输出数据。
这些寄存器被分别映射到外围 I/O 寄存器区中单独的地址之上。
这些寄存器可以按 8 位或按 1 位为单位进行读写。
复位后这些寄存器设置为 00H。
如果指定 4 位×1 通道 或 2 位×1 通道的操作模式（RTPC0.BYTE0 位= 0），数据能被分别设置到 RTBL0 和 RTBH0 寄存器，通过指定寄存器的地址，可以读出这两个寄存器的数据。
如果指定 6 位×1 通道的操作模式（BYTE0 位= 1），向其中一个寄存器写入数据，都可以把 8 位数据设置到 RTBL0 和 RTBH0 寄存器。此外，通过指定其寄存器的地址，可以读这两个寄存器的数据。
RTBL0 和 RTBH0 寄存器的处理过程如 表 14-2 所示：

复位后: 00H R/W 地址: RTBL0 FFFFF6E0H, RTBH0 FFFFF6E2H

7 6 5 4 3 2 1 0

RTBL0

				RTBL03	RTBL02	RTBL01	RTBL00

RTBH0

0	0	RTBH05	RTBH04				

注意事项

1. 如果向 RTBH0 寄存器位 6、位 7 写入数据，写入值总是 0。

2. 在以下状态中，禁止访问 RTBL0 和 RTBH0 寄存器。有关详细说明，参见 3.4.9（2）访问片上专用外围 I/O 寄存器。

- 当 CPU 工作于副时钟，而且主时钟振荡停止时
- 当 CPU 工作于内部振荡时钟时

表 14-2. 使用 RTBL0 和 RTBH0 寄存器的处理操作

操作模式	处理的寄存器	读取		写 ^注	
		高 4 位	低 4 位	高 4 位	低 4 位
4 位×1 通道	RTBL0	RTBH0	RTBL0	无效	RTBL0
2 位×1 通道	RTBH0	RTBH0	RTBL0	RTBH0	Invalid
6 位×1 通道	RTBL0	RTBH0	RTBL0	RTBH0	RTBL0
	RTBH0	RTBH0	RTBL0	RTBH0	RTBL0

注 设置实时输出端口后，在实时输出触发信号产生时，把输出数据设置到 RTBL0 和 RTBH0 寄存器中。

14.3 寄存器

RTO 由以下两个寄存器控制：

- 实时输出端口模式寄存器 0（RTPM0）
- 实时输出端口控制寄存器 0（RTPC0）

（1）实时输出端口模式寄存器 0（RTPM0）

RTPM0 寄存器选择实时输出端口模式或以 1 位为单位的端口模式。
本寄存器可按 8 位或 1 位读写。
复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: RTPM0 FFFFF6E4H

	7	6	5	4	3	2	1	0
RTPM0	0	0	RTPM05	RTPM04	RTPM03	RTPM02	RTPM01	RTPM00

RTPM0m	实时输出端口控制 (m = 0 至 5)
0	禁止实时输出
1	允许实时输出

- 注意事项
1. 通过使能实时输出操作（RTPC0.RTPOE0 位 = 1），允许 RTP00 至 RTP05 信号使能实时输出的位进行实时输出，并且这些位设置为端口模式输出 0。
 2. 如果禁止实时输出操作（RTPOE0 位= 0），不管 RTPM0 寄存器的如何设置，实时输出引脚（RTP00 至 RTP05）均输出 0。
 3. 为将该寄存器用作实时输出引脚（RTP00 至 RTP05），需用 PMC 和 PFC 寄存器将这些引脚设置为实时输出端口引脚。

(2) 实时输出端口控制寄存器 0（RTPC0）

RTPC0 寄存器用于设置操作模式和实时输出端口的输出触发。

操作模式和实时输出端口输出触发的关系如表 14-3 所示。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: FFFFF6E5H

RTPC0

<7>	6	5	4	3	2	1	0
RTPOE0	RTPEG0	BYTE0	EXTR0	0	0	0	0

RTPOE0	实时输出操作控制
0	禁止操作 ^{注 1}
1	允许操作

RTPEG0	INTTP0CC0 信号的有效沿
0	下降沿 ^{注 2}
1	上升沿

BYTE0	为实时输出指定通道配置
0	4 位 x 2 通道, 2 位 x 2 通道
1	6 位 x 2 通道

注

1.

禁止实时输出操作（RTPOE0 位 = 0）时，所有实时输出引脚（RTP00 至 RTP05）都输出“0”。

2.

TAA0 选择 1 个计数时钟时输出 INTTAA0CC0 信号。

注意事项 只有 RTPOE0 位 = 0 时才能设置 RTPEG0、BYTE0 和 EXTR0 位。

表 14-3. 实时输出端口的操作模式和输出触发

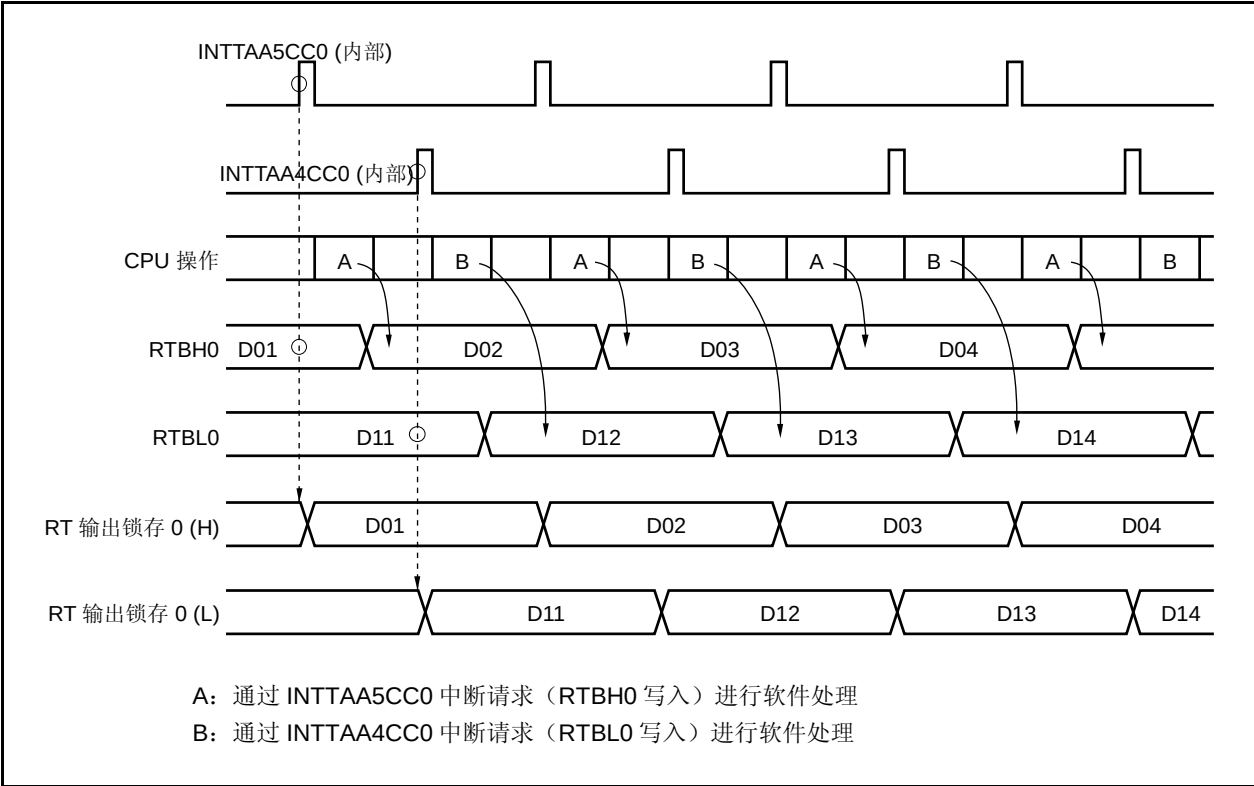
BYTE0	EXTR0	操作模式	RTBH0（RTP04， RTP05）	RTBL0（RTP00 至 RTP03）
0	0	4 位x1 通道	INTTAA5CC0	INTTAA4CC0
	1	2 位x1 通道	INTTAA4CC0	INTTAA0CC0
1	0	6 位x1 通道	INTTAA4CC0	
	1		INTTAA0CC0	

14.4 操作

如果通过设置 RTPC0.RTPOE0 位为 1 使能实时输出操作，在传输触发信号产生（由 RTPC0.EXTR0 和 RTPC0.BYTE0 位设置）的同时，RTBH0 和 RTBL0 寄存器中的数据被传输到实时输出锁存器中。在传送的数据中，只有通过 RTPM0 寄存器使能实时输出的位的数据会从 RTP00 至 RTP05 位输出。通过 RTPM0 寄存器禁止实时输出的位输出 0。

若通过把 RTPOE0 位清为 0 禁止实时输出操作，则不管 RTPM0 寄存器如何设置，RTP00 至 RTP05 信号均输出 0。

图 14-2. RTO0 的操作时序示例（当 EXTR0= 0，BYTE0 位= 0）



备注 关于待机期间的操作，参见第二十五章 待机功能。

14.5 用途

(1) 禁止实时输出

将 RTPC0.RTPOE0 位清除为 0。

(2) 进行下列初始化操作。

- 设置端口 2 或端口 5 的复用功能引脚
设置 PFC2.PFC2m 位和 PFCE2.PFCE2m 位为 RTO 引脚后，将 PMC2.PMC2m 位设置为 1 (m = 0 至 3)。
设置 PFC5.PFC5m 位和 PFCE5.PFCE5m 位为 RTO 引脚后，将 PMC5.PMC5m 位设置为 1 (m = 0 至 5)。
- 用 1 位单元指定实时输出端口模式或端口模式。
设置 RTPM0 寄存器。
- 通道配置：选择输出触发和有效沿
设置 RTPC0.EXTR0、RTPC0.BYTE0 和 RTPC0.RTPEG0 位。
- 将初始值设置到 RTBH0 和 RTBL0 寄存器^{注 1}。

(3) 实时输出使能

设置 RTPOE0 位 = 1。

(4) 在所选的传输触发信号产生时，把下一个输出值设置到 RTBH0 和 RTBL0 寄存器^{注 2}。

(5) 通过所选触发相应的中断服务程序，把下一个实时输出值送到 RTBH0 和 RTBL0 寄存器。

- 注
1. RTPOE0 位 = 0 时，如果写入 RTBH0 和 RTBL0 寄存器，则该值被传送到实时输出锁存 0H 和 0L。
 2. RTPOE0 位 = 1 时，即使写入 RTBH0 和 RTBL0 寄存器，数据不被传送到实时输出锁存 0H 和 0L。

14.6 注意事项

(1) 使用软件时避免以下的冲突：

- 实时输出禁止/使能转换 (RTPOE0 位) 和选择的实时输出输出触发之间的冲突。
- 在实时输出状态下对 RTBH0 和 RTBL0 寄存器的写入和选择的实时输出输出触发之间的冲突。

(2) 在进行初始化之前，禁止实时输出 (RTPOE0 位 = 0)。

(3) 一旦实时输出被禁止 (RTPOE0 位 = 0)，确保重新使能实时输出 (RTPOE0 位 = 0 → 1) 之前初始化 RTBH0 和 RTBL0 寄存器。

第十五章 A/D 转换器

15.1 概述

A/D 转换器可以将模拟输入信号转换为数字值，具有 10 位分辨率且能够处理 12 个模拟输入信号通道（AN10 至 AN11）。

A/D 转换器具有以下功能：

- 10 位分辨率。
- 12 个通道。
- 渐次逼近方式
- 工作电压： $AV_{REF0} = 3.0$ 至 3.6 V
- 模拟输入电压： 0 V 至 AV_{REF0}
- 提供以下操作模式：
 - 连续选择模式
 - 连续扫描模式
 - 单次选择模式
 - 单次扫描模式
- 提供以下触发模式：
 - 软件触发模式
 - 外部触发模式（外部，1）
 - 定时器触发模式
- 掉电监控功能（转换结果比较功能）

15.2 功能

（1）10 位分辨率的 A/D 转换

模拟通道从 AN10 至 AN11 通道中选择，A/D 转换操作以 10 位分辨率进行重复。每次 A/D 转换完成后，都会产生一个中断请求信号（INTAD）。

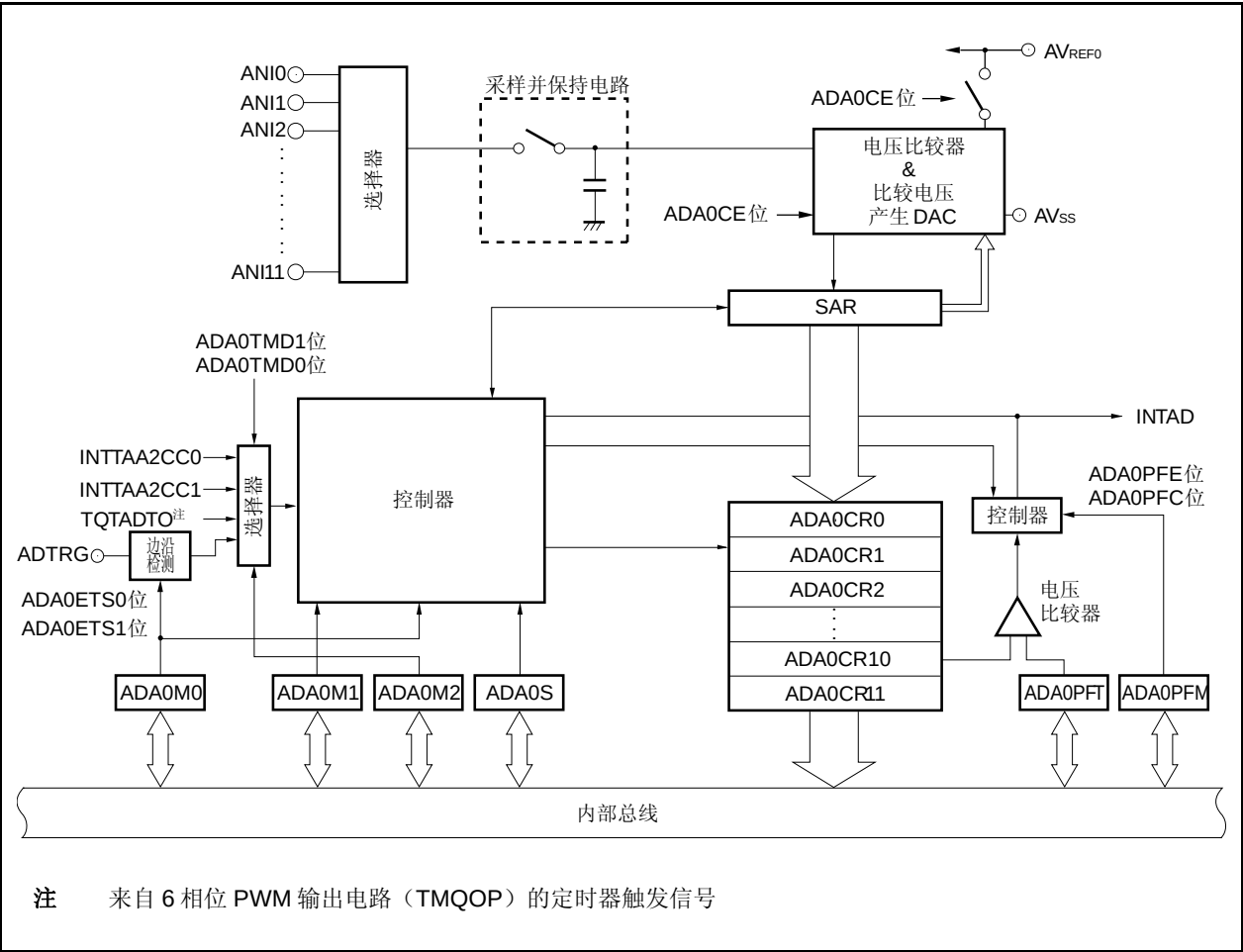
（2）掉电检测功能

使用此功能检测电池电压的下降。将 A/D 转换结果（ADA0CRnH 寄存器的值）和 ADA0PFT 寄存器值进行比较，仅当满足指定的比较条件（ $n = 0$ 至 11 ）时，才会产生 INTAD 信号。

15.3 配置

A/D 转换器的框图如下所示：

图 15-1. A/D 转换器框图



A/D 转换器包括以下硬件：

表 15-1. A/D 转换器的配置

项目	配置
模拟输入	12 个通道（ANI0 引脚至 ANI11 引脚）
寄存器	逐次逼近寄存器（SAR） A/D 转换结果寄存器 0 至 11（ADA0CR0 至 ADA0CR11） A/D 转换结果寄存器 0H 至 11H（ADCR0H 至 ADCR11H）：仅高 8 位可读
控制寄存器	A/D 转换器模式寄存器 0 至 2（ADA0M0 至 ADA0M2） A/D 转换器通道指定寄存器 0（ADA0S） 掉电比较模式寄存器（ADA0PFM） 掉电比较阈值寄存器（ADA0PFT）

(1) 逐次逼近寄存器 (SAR)

SAR 将模拟输入信号的电压值与比较电压产生 DAC (比较电压) 的输出值进行比较, 并且从最高有效位 (MSB) 开始保存比较结果。

当比较结果已经保存至最低有效位 (LSB) (也就是 A/D 转换完成时) 时, 将 SAR 的内容传输至 ADA0CRn 寄存器。

备注 n = 0 至 11

(2) A/D 转换结果寄存器 n (ADA0CRn), A/D 转换结果寄存器 nH (ADA0CRnH)

ADA0CRn 寄存器是一个存储 A/D 转换结果的 16 位寄存器。ADA0ARn 由 12 个寄存器组成, A/D 转换结果存储在对应于模拟输入的 AD0CRn 寄存器的高 10 位。(低 6 位固定为 0)

(3) A/D 转换器模式寄存器 0 (ADA0M0)

该寄存器指定操作模式并且控制 A/D 转换器的转换操作。

(4) A/D 转换器模式寄存器 1 (ADA0M1)

该寄存器设置待转换模拟输入信号的转换时间。

(5) A/D 转换器模式寄存器 2 (ADA0M2)

该寄存器设置硬件触发模式。

(6) A/D 转换器通道指定寄存器 (ADA0S)

该寄存器设置输入端口, 该端口用于输入待转换的模拟电压。

(7) 掉电比较模式寄存器 (ADA0PFM)

该寄存器设置掉电监控模式。

(8) 掉电比较阈值寄存器 (ADA0PFT)

ADA0PFT 寄存器设置与 A/D 转换结果寄存器 nH (ADA0CRnH) 的值进行比较的阈值。

设置至 ADA0PFT 寄存器的 8 位数据与 A/D 转换结果寄存器 (ADA0CRnH) 的高 8 位进行比较。

(9) 控制器

当 A/D 转换完成时或当使用掉电检测功能时, 控制器将 A/D 转换的结果 (ADA0CRnH 寄存器的值) 与 ADA0PFT 寄存器的值进行比较, 仅当满足指定的比较条件时产生 INTAD 信号。

(10) 采样&保持电路

采样保持电路采集由输入电路选择的各个模拟输入信号, 并将采样数据发送到电压比较器。A/D 转换期间, 该电路保持所采样的模拟输入信号的电压。

(11) 电压比较器

电压比较器将已经采样并保持的电压值与比较电压产生 DAC 的输出电压进行比较。

(12) 比较电压产生 DAC

比较电压产生 DAC 连接在 AV_{REF0} 和 AV_{SS} 之间，并产生用于同模拟输入信号作比较的电压。

(13) $ANI0$ 至 $ANI11$ 引脚

这些引脚是 12 个 A/D 转换器通道的模拟输入引脚，用于输入将要转换为数字信号的模拟信号。除了通过 ADA0S 寄存器选定的模拟输入引脚外，其他引脚都能被用作输入端口引脚。

注意事项 确保输入 $ANI0$ 引脚至 $ANI11$ 引脚的电压没有超过额定值。如果输入至某通道的电压为 AV_{REF0} 的电压或者更高，那么此通道的转换值不确定，并且其它通道的转换值也会受到影响。

(14) AV_{REF0} 引脚

该引脚用于输入 A/D 转换器的参考电压。即使不使用 A/D 转换器，也要保证该引脚的电压与 V_{DD} 引脚的电压相同。

根据应用在 AV_{REF0} 引脚和 AV_{SS} 引脚之间的电压，将输入到 $ANI0$ 至 $ANI11$ 引脚的信号转化为数字信号。

(15) AV_{SS} 引脚

该引脚为 A/D 转换器的接地引脚。即使不使用 A/D 转换器，也要保证该引脚的电压与 V_{SS} 引脚的电压相同。

15.4 寄存器

A/D 转换器是由以下寄存器控制：

- A/D 转换器模式寄存器 0、1、2（ADA0M0、ADA0M1、ADA0M2）
- A/D 转换器通道指定寄存器 0（ADA0S）
- 掉电比较模式寄存器（ADA0PFM）

以下寄存器也被使用：

- A/D 转换结果寄存器 n（ADA0CRn）
- A/D 转换结果寄存器 nH（ADA0CRnH）
- 掉电比较阈值寄存器（ADA0PFT）

（1）A/D 转换器模式寄存器 0（ADA0M0）

ADA0M0 寄存器是 8 位寄存器，用于指定操作模式并控制转换操作。

本寄存器可按 8 位或 1 位读写。但是，ADA0EF 0 是只读位。

复位后本寄存器被设置为 00H。

(1/2)

复位后: 00H R/W 地址: FFFFF200H

	<7>	6	5	4	3	2	1	<0>
ADA0M0	ADA0CE	0	ADA0MD1	ADA0MD0	ADA0ETS1	ADA0ETS0	ADA0TMD	ADA0EF

ADA0CE	A/D 转换控制
0	停止 A/D 转换
1	允许 A/D 转换

ADA0MD1	ADA0MD0	A/D 转换操作模式的说明
0	0	连续选择模式
0	1	连续扫描模式
1	0	单次选择模式
1	1	单次扫描模式

ADA0ETS1	ADA0ETS0	外部触发(ADTRG 引脚) 输入有效沿的说明
0	0	无边沿检测
0	1	下降沿检测
1	0	上升沿检测
1	1	双边沿检测

ADA0TMD	触发模式说明
0	软件触发模式
1	外部触发模式/定时器触发模式

ADA0EF	A/D 转换器状态显示
0	A/D 转换停止
1	A/D 转换正在进行

- 注意事项**
- 下列状态下禁止访问 **ADA0M0** 寄存器。有关详细说明，参见 3.4.9 (2) 访问片上专用外围 I/O 寄存器。
 - 当 CPU 工作于副时钟，而且主时钟振荡停止时
 - 当 CPU 工作于内部振荡时钟时
 - 忽略对位 0 的写入操作。
 - A/D 转换使能（**ADA0CE** 位 = 1）时，禁止改变 **ADA0M1.ADA0FR2** 位至 **ADA0M1.ADA0FR0** 位。
 - 下列模式中，当 A/D 转换停止（**ADA0CE** 位 = 0）时，写 **ADA0M0**、**ADA0M2**、**ADA0S**、**ADA0PFM** 或 **ADA0PFT** 寄存器，然后使能 A/D 转换器的操作（**ADA0CE** 位 = 1）
 - 正常转换模式下
 - 高速转换模式下的单次选择模式/单次扫描模式
 A/D 转换期间（**ADA0EF** 位 = 1），如果在其它模式下写入 **ADA0M0**、**ADA0M2**、**ADA0S**、**ADA0PFM** 和 **ADA0PFT** 寄存器，那么根据模式将会执行下列操作。
 - 软件触发模式中：

A/D 转换停止并再次从头开始执行。
 - 硬件触发模式中：

A/D 转换停止且设置为触发等待状态。
 - 若要选择外部触发模式/定时器触发模式（**ADA0TMD** 位 = 1），需设置为高速转换模式（**ADA0M1.ADA0HS1** 位 = 1）A/D 转换操作使能（**ADA0CE** 位 = 1）后，在随即插入的振荡稳定时间期间，不要输入触发信号。
 - 当不使用 A/D 转换器时，通过将 **ADA0CE** 位设置为 0 停止该操作来降低功耗。

(2) A/D 转换器模式寄存器 1 (ADA0M1)

ADA0M1 是用来指定转换时间的 8 位寄存器。

本寄存器可按 8 位或 1 位读写。

复位后该位清 00H。

复位后: 00H R/W 地址: FFFFF201H

	7	6	5	4	3	2	1	0
ADA0M1	ADA0HS1	0	0	0	0	ADA0FR2	ADA0FR1	ADA0FR0

ADA0HS1	正常转换模式/高速模式的说明(A/D 转换时间)
0	正常转换模式
1	高速转换模式

- 注意事项**
1. A/D 转换使能 (ADA0M0.ADA0CE 位= 1) 时, 禁止改变 ADA0M1 寄存器。
 2. 为了选择外部触发模式/定时器触发模式 (ADA0M0.ADA0TMD 位= 1), 设置为高速转换模式 (ADA0HS1 位 = 1) A/D 转换操作使能 (ADA0CE 位= 1) 后, 在随即插入的振荡稳定时间期间, 不要输入触发信号。
 3. 务必将位 6 至位 3 清为“0”。

备注 关于 A/D 转换时间设置的例子, 参见表 15-2 和 15-3。

表 15-2. 正常转换模式下转换时间的选择 (ADA0HS1 位= 0)

ADA0FR3 至 ADA0FR0 位	A/D 转换时间			
	稳定时间 + 转换时间 + 等待时间	48 MHz	32 MHz	24 MHz
0000	$26/f_{xx} + 52/f_{xx} + 54/f_{xx}$	$2.75 \mu s$	$4.13 \mu s$	$5.50 \mu s$
0001	$52/f_{xx} + 104/f_{xx} + 106/f_{xx}$	$5.46 \mu s$	$8.19 \mu s$	禁止设置
0010	$78/f_{xx} + 156/f_{xx} + 158/f_{xx}$	$8.17 \mu s$	禁止设置	禁止设置
0011	$100/f_{xx} + 208/f_{xx} + 210/f_{xx}$	禁止设置	禁止设置	禁止设置
0100	$100/f_{xx} + 260/f_{xx} + 262/f_{xx}$	禁止设置	禁止设置	禁止设置
0101	$100/f_{xx} + 312/f_{xx} + 314/f_{xx}$	禁止设置	禁止设置	禁止设置
0110	$100/f_{xx} + 364/f_{xx} + 366/f_{xx}$	禁止设置	禁止设置	禁止设置
0111	$100/f_{xx} + 416/f_{xx} + 418/f_{xx}$	禁止设置	禁止设置	禁止设置
1000	$100/f_{xx} + 468/f_{xx} + 470/f_{xx}$	禁止设置	禁止设置	禁止设置
1001	$100/f_{xx} + 520/f_{xx} + 522/f_{xx}$	禁止设置	禁止设置	禁止设置
1010	$100/f_{xx} + 572/f_{xx} + 574/f_{xx}$	禁止设置	禁止设置	禁止设置
1011	$100/f_{xx} + 624/f_{xx} + 626/f_{xx}$	禁止设置	禁止设置	禁止设置
1100	$100/f_{xx} + 676/f_{xx} + 678/f_{xx}$	禁止设置	禁止设置	禁止设置
1101	$100/f_{xx} + 728/f_{xx} + 730/f_{xx}$	禁止设置	禁止设置	禁止设置
1110	$100/f_{xx} + 780/f_{xx} + 782/f_{xx}$	禁止设置	禁止设置	禁止设置
1111	$100/f_{xx} + 832/f_{xx} + 834/f_{xx}$	禁止设置	禁止设置	禁止设置
其它	禁止设置			

备注 稳定时间: A/D 转换建立时间 ($1 \mu s$ 或更长)
 转换时间: 实际的 A/D 转换时间 (2.17 至 $9.75 \mu s$)
 等待时间: 下一次转换之前插入的等待时间
 f_{xx} : 主时钟频率

在正常转换模式下, 当 ADA0M0.ADA0CE 位设置为 1 后, 稳定时间过后开始转换, 并且 A/D 转换仅在转换时间 (2.17 至 $9.75 \mu s$) 期间执行。转换结束后操作停止, 在等待时间过后, 产生 A/D 转换结束中断请求信号 (INTAD)。

由于转换操作在等待时间期间停止, 所以工作电流可能减小。

- 注意事项 1. 将转换时间设置在 $2.17 \mu s \leq \text{转换时间} \leq 9.75 \mu s$ 。
2. A/D 转换期间, 如果写入 ADA0M0、ADA0M2、ADA0S、ADA0PFM 以及 ADA0PFT 寄存器或者输入触发信号, 则重新执行转换。但是, 如果稳定时间结束时序与写入这些寄存器发生冲突, 或者稳定时间结束时序与触发信号输入发生冲突, 那么将会插入 64 个时钟的稳定时间。如果与插入的稳定时间结束时序再次发生冲突, 则仍然插入稳定时间。因此, 不要将触发输入间隔和控制寄存器写入间隔设置为 64 个时钟或更少。

表 15-3. 高速转换模式下转换时间的选择 (ADA0HS1 位= 1)

ADA0FR3 至 ADA0FR0 位	A/D 转换时间			
	转换时间 (+ 稳定时间)	48 MHz	32 MHz	24 MHz
0000	$52/f_{xx} (+26/f_{xx})$	禁止设置	禁止设置	$2.17 \mu s$
0001	$104/f_{xx} (+52/f_{xx})$	$2.17 \mu s$	$3.25 \mu s$	$4.33 \mu s$
0010	$156/f_{xx} (+78/f_{xx})$	$3.25 \mu s$	$4.88 \mu s$	$6.50 \mu s$
0011	$208/f_{xx} (+100/f_{xx})$	$4.33 \mu s$	$6.50 \mu s$	$8.67 \mu s$
0100	$260/f_{xx} (+100/f_{xx})$	$5.42 \mu s$	$8.13 \mu s$	禁止设置
0101	$312/f_{xx} (+100/f_{xx})$	$6.50 \mu s$	$9.75 \mu s$	禁止设置
0110	$364/f_{xx} (+100/f_{xx})$	$7.58 \mu s$	禁止设置	禁止设置
0111	$416/f_{xx} (+100/f_{xx})$	$8.67 \mu s$	禁止设置	禁止设置
1000	$468/f_{xx} (+100/f_{xx})$	$9.75 \mu s$	禁止设置	禁止设置
1001	$520/f_{xx} (+100/f_{xx})$	禁止设置	禁止设置	禁止设置
1010	$572/f_{xx} (+100/f_{xx})$	禁止设置	禁止设置	禁止设置
1011	$624/f_{xx} (+100/f_{xx})$	禁止设置	禁止设置	禁止设置
1100	$676/f_{xx} (+100/f_{xx})$	禁止设置	禁止设置	禁止设置
1101	$728/f_{xx} (+100/f_{xx})$	禁止设置	禁止设置	禁止设置
1110	$780/f_{xx} (+100/f_{xx})$	禁止设置	禁止设置	禁止设置
1111	$832/f_{xx} (+100/f_{xx})$	禁止设置	禁止设置	禁止设置
其它	禁止设置			

备注 稳定时间: A/D 转换建立时间 ($1 \mu s$ 或更长)
 转换时间: 实际的 A/D 转换时间 (2.17 至 $9.75 \mu s$)
 f_{xx} : 主时钟频率

在高速转换模式下, 当 ADA0M0.ADA0CE 位设置为 1 后, 稳定时间过后开始转换, 并且 A/D 转换仅在转换时间 (2.17 至 $9.75 \mu s$) 期间执行。转换结束后将立即产生 A/D 转换结束中断请求信号 (INTAD)。

在连续转换模式下, 稳定时间仅在第一次转换之前插入稳定时间, 而不会在第二次转换后 (A/D 转换器仍然运行) 插入。

- 注意事项 1.** 将转换时间设置在 $2.17 \mu s \leq \text{转换时间} \leq 9.75 \mu s$ 。
- 2.** 在高速转换模式下, 振荡稳定期间, 禁止触发输入和对 ADA0M0、ADA0M2、ADA0S、ADA0PFM 和 ADA0PFT 寄存器的重写。

(3) A/D 转换器模式寄存器 2 (ADA0M2)

ADA0M2 寄存器指定硬件触发模式。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: FFFFF203H

	7	6	5	4	3	2	1	0
ADA0M2	0	0	0	0	0	0	ADA0TMD1	ADA0TMD0

ADA0TMD1	ADA0TMD0	硬件触发模式的说明
0	0	外部触发模式 (当选择 ADTRG 引脚有效沿时)
0	1	定时器触发模式 0 (当产生 INTTAA2CC0 中断请求时)
1	0	定时器触发模式 1 (当产生 INTTAA2CC1 中断请求时)
1	1	定时器触发模式 2 (TQTADT0 信号)

- 注意事项 1. 下列模式中，当 A/D 转换停止 (ADA0M0.ADA0CE 位 = 0) 时，写 ADA0M2 寄存器，然后使能 A/D 转换器的操作 (ADA0CE 位 = 1)。
- 正常转换模式下
 - 高速转换模式下的单次选择模式/单次扫描模式
2. 务必将位 7 至位 2 清为“0”。

(4) 模拟输入通道指定寄存器 0 (ADA0S)

ADA0S 寄存器用于指定输入模拟电压（将转换为数字信号）的引脚。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: FFFFF202H

	7	6	5	4	3	2	1	0
ADA0S	0	0	0	0	ADA0S3	ADA0S2	ADA0S1	ADA0S0

ADA0S3	ADA0S2	ADA0S1	ADA0S0	选择模式	扫描模式
0	0	0	0	ANI0	ANI0
0	0	0	1	ANI1	ANI0, ANI1
0	0	1	0	ANI2	ANI0 至 ANI2
0	0	1	1	ANI3	ANI0 至 ANI3
0	1	0	0	ANI4	ANI0 至 ANI4
0	1	0	1	ANI5	ANI0 至 ANI5
0	1	1	0	ANI6	ANI0 至 ANI6
0	1	1	1	ANI7	ANI0 至 ANI7
1	0	0	0	ANI8	ANI0 至 ANI8
1	0	0	1	ANI9	ANI0 至 ANI9
1	0	1	0	ANI10	ANI0 至 ANI10
1	0	1	1	ANI11	ANI0 至 ANI11

注意事项 1. 下列模式中，当 A/D 转换停止（ADA0M0.ADA0CE 位 = 0）时，将数据写入 ADA0S 寄存器，然后使能 A/D 转换器的操作（ADA0CE 位 = 1）。

- 正常转换模式下
- 高速转换模式下的单次选择模式/单次扫描模式

2. 务必将位 7 至位 4 清为“0”。

(5) A/D 转换结果寄存器 n、nH (ADA0CRn, ADA0CRnH)

ADA0CRn 寄存器和 ADA0CRnH 寄存器存储 A/D 转换结果。

这些寄存器是以字节或字为单位进行访问的只读寄存器。ADA0CRn 寄存器用于字访问，将 ADA0CRnH 寄存器用于字节访问。从 ADA0CRn 寄存器的高 10 位读取 10 位转换结果，同时低 6 位的读取值为 0。自 ADA0CRnH 寄存器读取高 8 位转换结果。

注意事项 下列状态下，禁止访问 ADA0CRn 寄存器和 ADA0CRnH 寄存器。

有关详细说明，参见 3.4.9 (2) 访问片上专用外围 I/O 寄存器。

- 当 CPU 工作于副时钟，而且主时钟振荡停止时
- 当 CPU 工作于内部振荡时钟时

复位后: 未定义 R 地址: ADA0CR0 FFFFF210H, ADA0CR1 FFFFF212H,
ADA0CR2 FFFFF214H, ADA0CR3 FFFFF216H,
ADA0CR4 FFFFF218H, ADA0CR5 FFFFF21AH,
ADA0CR6 FFFFF21CH, ADA0CR7 FFFFF21EH,
ADA0CR8 FFFFF220H, ADA0CR9 FFFFF222H,
ADA0CR10 FFFFF224H, ADA0CR11 FFFFF226H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADA0CRn (n = 0 至 11)	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0	0	0	0	0	0	0

复位后: 未定义 R 地址: ADA0CR0H FFFFF211H, ADA0CR1H FFFFF213H,
ADA0CR2H FFFFF215H, ADA0CR3H FFFFF217H,
ADA0CR4H FFFFF219H, ADA0CR5H FFFFF21BH,
ADA0CR6H FFFFF21DH, ADA0CR7H FFFFF21FH,
ADA0CR8H FFFFF221H, ADA0CR9H FFFFF223H,
ADA0CR10H FFFFF225H, ADA0CR11H FFFFF227H

	7	6	5	4	3	2	1	0
ADA0CRnH (n = 0 至 11)	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2

注意事项 对 ADA0M0 寄存器和 ADA0S 寄存器的写入操作可能会导致 ADA0CRn 寄存器的内容不确定。转换后，在写入 ADA0M0 寄存器和 ADA0S 寄存器之前读取转换结果。不同于上述时序时，可能读取不到正确的转换结果。

至模拟输入引脚（ANI0 至 ANI11）的模拟输入与转换结果（ADA0CRn 寄存器）之间的关系如下：

$$SAR = \text{INT} \left(\frac{V_{IN}}{AV_{REF0}} \times 1,024 + 0.5 \right)$$

$$ADA0CR^{\#} = SAR \times 64$$

或，

$$(SAR - 0.5) \times \frac{AV_{REF0}}{1,024} \leq V_{IN} < (SAR + 0.5) \times \frac{AV_{REF0}}{1,024}$$

INT () : 括号内数值取整的函数

V_{IN}: 模拟输入电压

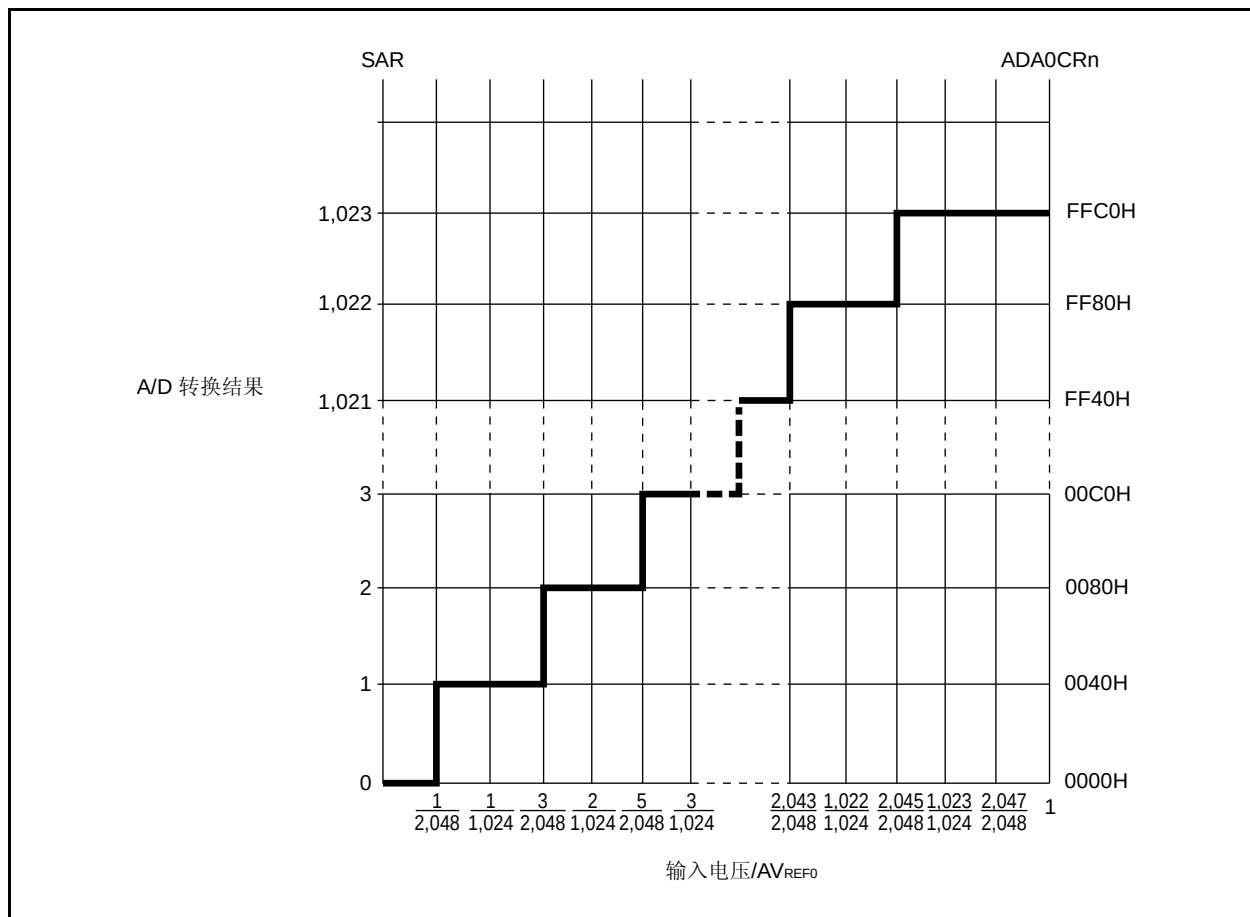
AV_{REF0}: AV_{REF0} 引脚电压

ADA0CR: ADA0CRn 寄存器的值

注 ADA0CRn 寄存器的低 6 位固定为 0。

以下显示了模拟输入电压和 A/D 转换结果之间关系。

图 15-2. 模拟输入电压和 A/D 转换结果的关系



(6) 掉电比较模式寄存器 (ADA0PFM)

ADA0PFM 寄存器是一个用于设置掉电比较模式的 8 位寄存器

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: FFFFF204H

	<7>	6	5	4	3	2	1	0
ADA0PFM	ADA0PFE	ADA0PFC	0	0	0	0	0	0

ADA0PFE	掉电比较使能/禁止的选择
0	禁止掉电比较
1	允许掉电比较

ADA0PFC	掉电比较模式的选择
0	当 $ADA0CRnH \geq ADA0PFT$ 时产生中断请求信号 (INTAD)
1	当 $ADA0CRnH < ADA0PFT$ 时产生中断请求信号 (INTAD)

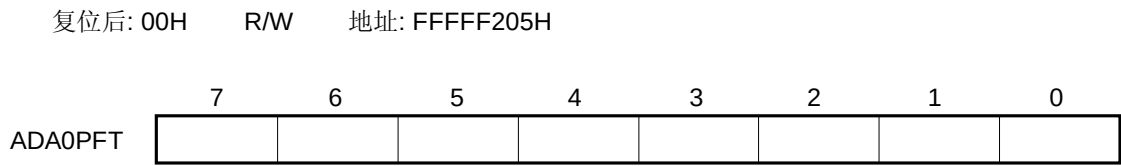
- 注意事项
1. 在选择模式下，将设置至 ADA0PFT 寄存器的 8 位数据与由 ADA0S 寄存器所指定的 ADA0CRnH 寄存器的值进行比较。如果该结果符合由 ADA0PFC 位指定的条件，那么将转换结果存储在 ADA0CRn 寄存器，并且产生 INTAD 信号。但是，如果不符合，将不产生中断信号。
 2. 在扫描模式下，将设置至 ADA0PFT 寄存器的 8 位数据与 ADA0CR0H 寄存器的内容进行比较。如果该结果符合由 ADA0PFC 位指定的条件，那么将转换结果存储在 ADA0CR0 寄存器中，并且产生 INTAD 信号。但是，如果不匹配，将不产生 INTAD 信号。无论比较结果如何，扫描操作完成之前，会继续扫描，并且将转换结果存储在 ADA0CRn 寄存器中。但是，扫描操作完成后不产生 INTAD 信号。
 3. 下列模式中，当 A/D 转换停止 (ADA0M0.ADA0CE 位 = 0) 时，将数据写入 ADA0PFM 寄存器，然后使能 A/D 转换器的操作 (ADA0CE 位 = 1)。
 - 正常转换模式下
 - 高速转换模式下的单次选择模式/单次扫描模式

(7) 掉电比较阈值寄存器 (ADA0PFT)

ADA0PFT 寄存器用于设置掉电比较模式下的比较值。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。



注意事项 下列模式中，当 A/D 转换停止（ADA0M0.ADA0CE 位 = 0）时，将数据写入 ADA0PFT 寄存器，然后使能 A/D 转换器的操作（ADA0CE 位 = 1）。

- 正常转换模式下
- 高速转换模式下的单次选择模式/单次扫描模式

15.5 操作

15.5.1 基本操作

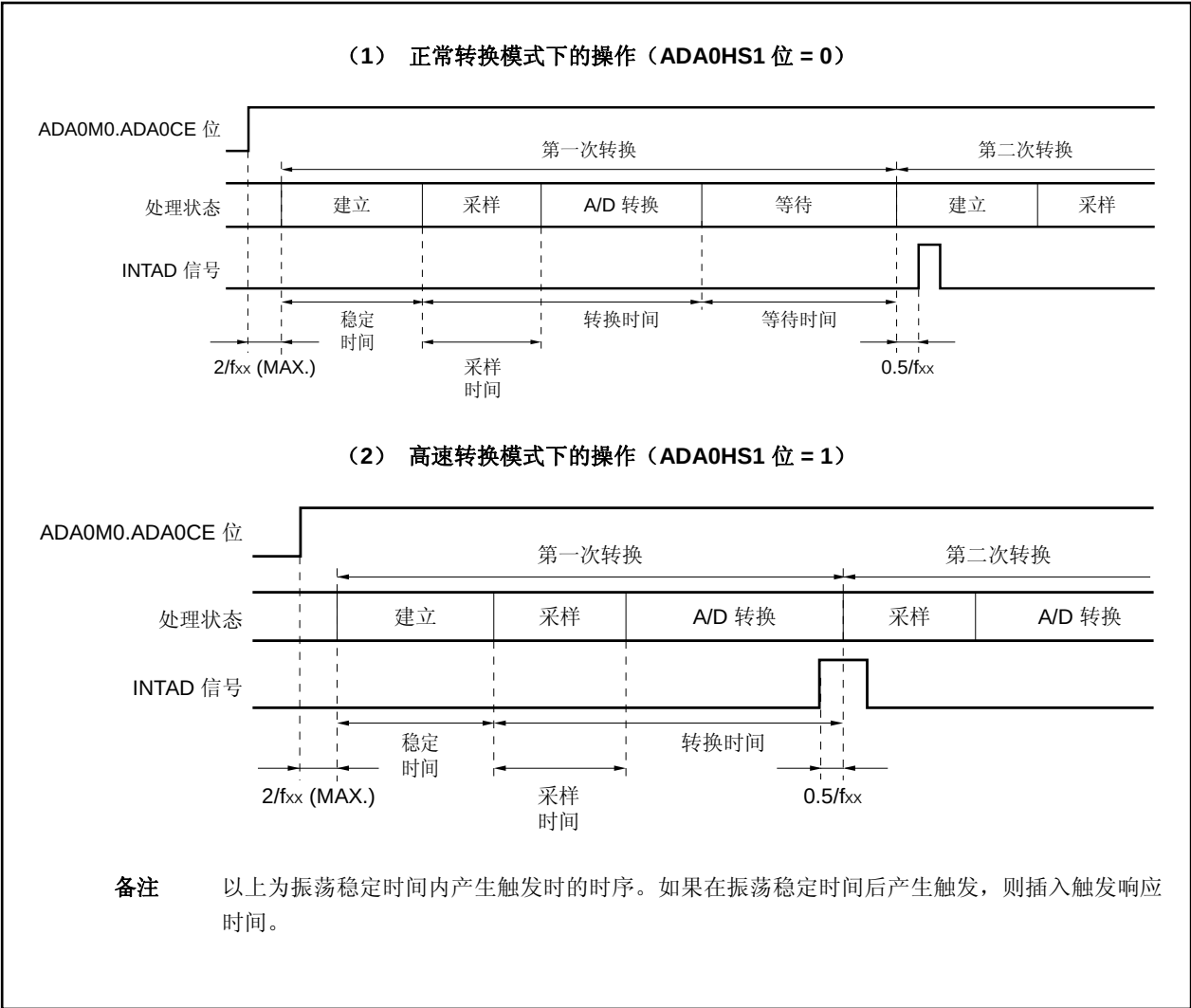
- <1> 使用 ADA0M0、ADA0M1、ADA0M2 和 ADA0S 寄存器设置执行 A/D 转换的操作模式、触发模式以及转换时间。当设置 ADA0M0 寄存器的 ADA0CE 位时，在软件触发模式下转换开始，在外部或定时器触发模式下 A/D 转换器等待触发。
- <2> A/D 转换开始后，采样&保持电路对所选模拟输入通道的输入电压进行采样。
- <3> 当采样&保持电路采样输入通道到指定时间时，采样&保持电路进入保持状态，并且保持输入模拟电压直到 A/D 转换完成。
- <4> 设置逐次逼近寄存器（SAR）的位 9，将比较电压发生 DAC 为设置 $(1/2) AV_{REF0}$ 。
- <5> 比较电压产生 DAC 的电压和模拟输入电压之间的电压差由电压比较器进行比较。如果模拟输入电压高于 $(1/2) AV_{REF0}$ ，SAR 的 MSB 保持设置。如果模拟输入电压低于 $(1/2) AV_{REF0}$ ，MSB 复位。
- <6> 接着，自动设置逐次逼近寄存器（SAR）的位 8 并启动下一次比较。根据先前所设定的位 9 的值，比较电压产生 DAC 的电压档位选择如下：
 - 位 9 = 1: $(3/4) AV_{REF0}$
 - 位 9 = 0: $(1/4) AV_{REF0}$
 该电压和模拟输入电压比较，并且根据比较结果，位 8 如下动作：
 模拟输入电压 $\geq$ 比较电压：位 8 = 1
 模拟输入电压 $\leq$ 比较电压：位 8 = 0
- <7> 此项比较一直持续到 SAR 寄存器的位 0。
- <8> 10 位比较完成时，先将有效数字结果存储在 SAR 寄存器，然后再传输并存储在 ADA0CRn 寄存器。之后，产生 A/D 转换结束中断请求信号（INTAD）。
- <9> 单次选择模式下，停止转换^注。单次扫描模式下，扫描之后，立刻停止转换^注。连续选择模式下，重复步骤<2>至步骤<8>直到 ADA0M0.ADA0CE 位清为 0。连续扫描模式下，每个通道都重复步骤<2>至步骤<8>。

注 在外部触发模式时、定时器触发模式 0 时或定时器触发模式 1 时，进入触发等待状态。

备注 触发等待状态指稳定时间过后的状态。

15.5.2 转换操作时序

图 15-3. 转换操作时序（连续转换）



15.5.3 触发模式

通过设置触发模式指定启动转换操作的时序。触发模式包括软件触发模式和硬件触发模式。硬件触发模式包含定时器触发模式 0 和 1，以及外部触发模式。ADA0M0.ADA0TMD 位用于设置触发模式。通过 ADA0M2.ADA0TMD1 位和 ADA0M2.ADA0TMD0 位设置硬件触发模式。

(1) 软件触发模式

当 ADA0M0.ADA0CE 位为 1 时，转换由 ADA0S 寄存器指定的模拟输入引脚（ANI0 引脚至 ANI11 引脚）的信号。当转换完成时，转换结果存储在 ADA0CRn 寄存器。同时，产生 A/D 转换结束中断信号（INTAD）。

如果通过 ADA0M0.ADA0MD1 位和 ADA0M0.ADA0MD0 位指定的操作模式为连续选择/扫描模式，那么转换完成后，将重复下一次转换，除非 ADA0CE 位清 0。如果操作模式是单脉冲选择/扫描模式，那么转换执行一次并结束。

转换开始时，ADA0M0.ADA0EF 位设为 1（表示转换正在进行）。

如果在转换期间写入 ADA0M0、ADA0M2、ADA0S、ADA0PFM 或 ADA0PFT 寄存器，那么此次转换被放弃并再次从头开始转换。但是，在正常转换模式和高速转换模式下的单次选择模式/单次扫描模式中，禁止写入这些寄存器。

(2) 外部触发模式

该模式下，输入外部触发（至 ADTRG 引脚）时，由 ADA0S 寄存器指定的模拟输入引脚（ANI0 至 ANI11）的信号开始转换。通过 ADA0M0.ADA0ETS1 位和 ADA0M0.ADA0ETS0 位来指定外部触发器的哪个边沿（也就是，上升沿、下降沿或双边沿）被检测。ADA0CE 位设为 1 时，A/D 转换器等待触发，且当输入外部触发后开始转换。

无论通过 ADA0MD1 位和 ADA0MD0 位设置的操作模式是连续选择、连续扫描、单次选择还是单次扫描模式，转换完成后，转换结果都存储在 ADA0CRn 寄存器中。同时，产生 INTAD 信号，并且 A/D 转换器等待下一次触发。

转换开始时，ADA0EF 位设为 1（表示转换正在进行）。然而，在 A/D 转换器等待触发期间，ADA0EF 位清为 0，则转换停止。如果在转换操作期间输入有效触发，转换操作停止并从头开始。

如果在转换操作期间写入 ADA0M0、ADA0M2、ADA0S、ADA0PFM 或 ADA0PFT 寄存器，那么此次转换会放弃，并且 A/D 转换器等待再次触发信号。然而，在单次选择模式/单次扫描模式下，禁止写入这些寄存器。

注意事项 若要选择外部触发模式，需设置为高速转换模式。A/D 转换操作使能（ADA0M0.ADA0CE 位= 1）后，在随即插入的振荡稳定时间期间，不要输入触发信号。

备注 触发等待状态指稳定时间过后的状态。

(3) 定时器触发模式

该模式下，通过捕获/比较寄存器（连接至定时器）的比较匹配中断请求信号（INTTAA2CC0 或 INTTAA2CC1），由 ADA0S 寄存器指定的模拟输入引脚（ANI0 至 ANI11）的信号开始转换。通过 ADA0TMD1 位和 ADA0TMD0 位选择 INTTAA2CC0 或 INTTAA2CC1 信号，并且在指定的比较匹配中断请求信号的上升沿处开始转换。当 ADA0CE 位设为 1 时，A/D 转换器等待触发，当输入定时器的比较匹配中断请求信号后，开始转换。

转换完成时，无论通过 ADA0MD1 位和 ADA0MD0 位设置的操作模式是连续选择、连续扫描、单脉冲选择还是单次扫描模式，转换结果都存储在 ADA0CRn 寄存器中。同时，产生 INTAD 信号，并且 A/D 转换器等待下一次触发。

转换开始时，ADA0EF 位设为 1（表示转换正在进行）。然而，在 A/D 转换器等待触发期间，ADA0EF 位清为 0，则转换停止。如果在转换操作期间输入有效触发，转换操作停止并从头开始。

如果在转换操作期间写入 ADA0M0、ADA0M2、ADA0S、ADA0PFM 或 ADA0PFT 寄存器，那么此次转换停止，并且 A/D 转换器等待再次触发信号。然而，在单次选择模式/单次扫描模式下，禁止写入这些寄存器。

注意事项 若要选择定时器触发模式，需设置为高速转换模式。A/D 转换操作使能（ADA0M0.ADA0CE 位= 1）后，在随即插入的振荡稳定时间期间，不要输入触发信号。

备注 触发等待状态指稳定时间过后的状态。

15.5.4 操作模式

有四个操作模式可供选择，在这四个操作模式中，需设置 ANI0 引脚至 ANI11 引脚。连续选择模式，连续扫描模式，单次选择模式以及单次扫描模式。

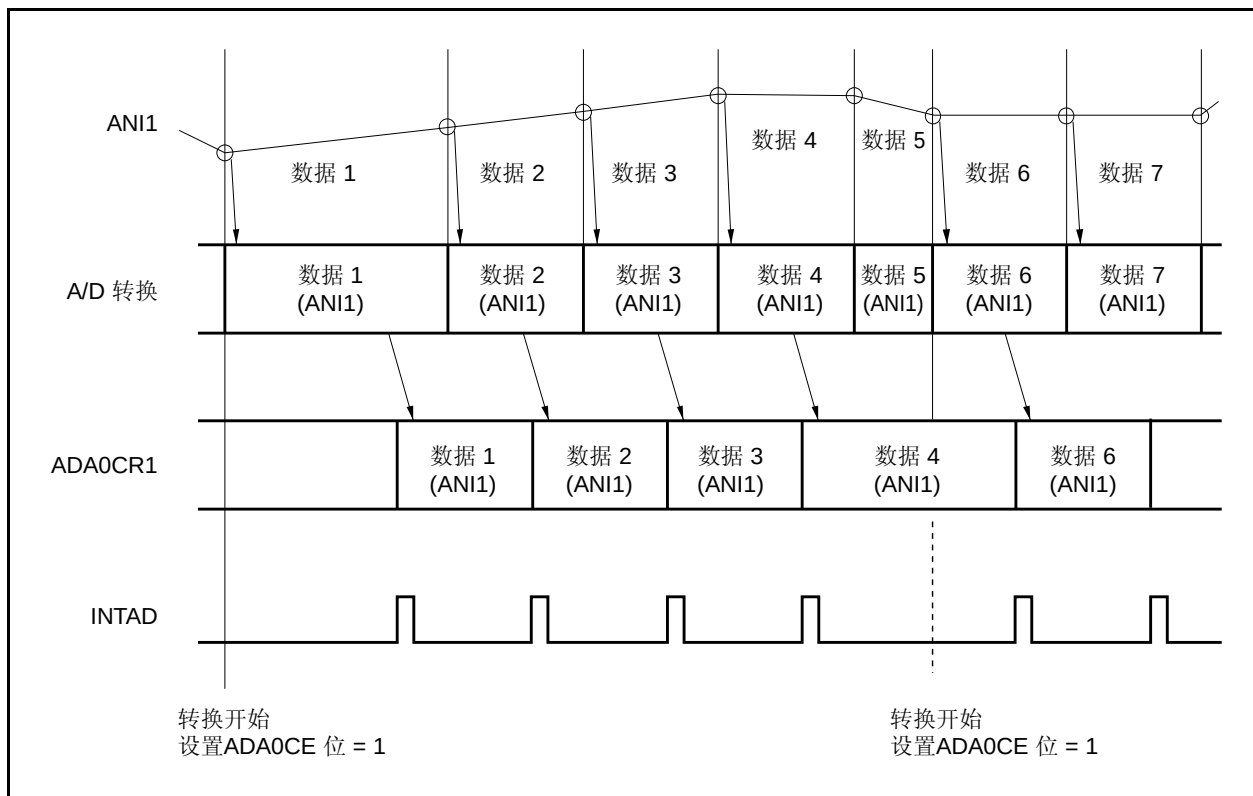
通过 ADA0M0.ADA0MD1 位和 ADA0M0.ADA0MD0 位来选择操作模式。

(1) 连续选择模式

该模式下，由 ADA0S 寄存器选择一个模拟输入引脚电压连续转换为数字值。

转换结果保存在与模拟输入引脚相对应的 ADA0CRn 寄存器中。该模式下，按照一一对应的方式，一个模拟输入引脚对应一个 ADA0CRn 寄存器。每次 A/D 转换完成后，都会产生 A/D 转换结束中断请求信号 (INTAD)。转换完成后，则开始下一次转换，除非 ADA0M0.ADA0CE 位清为 0 (n = 0 至 11)。

图 15-4. 连续选择模式操作的时序举例 (ADA0S 寄存器= 01H)



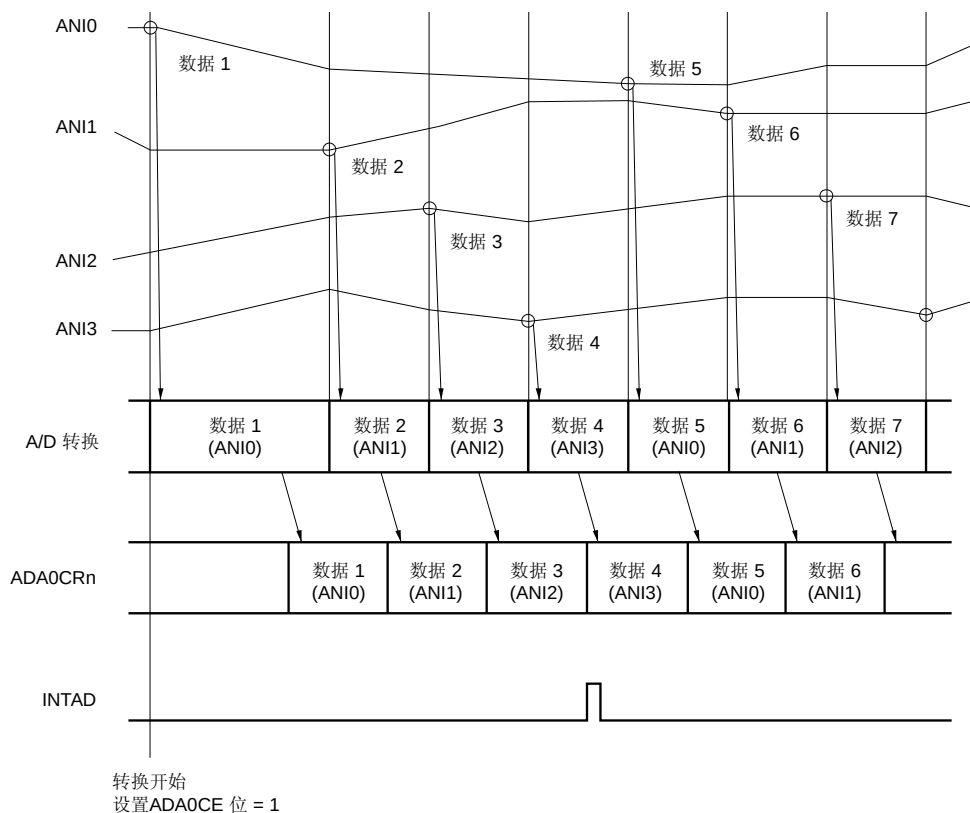
(2) 连续扫描模式

该模式下，模拟输入引脚从 ANI0 引脚至由 ADA0S 寄存器指定的引脚中连续选择，并且将它们的值连续转换为数字值。

各转换结果保存在与模拟输入引脚相对应的 ADA0CRn 寄存器中。当由 ADA0S 寄存器指定的模拟输入引脚的转换完成时，产生 INTAD 信号，并且 A/D 转换又会从 ANI0 引脚开始，除非 ADA0CE 位清为 0 (n = 0 至 11)。

图 15-5. 连续扫描模式操作的时序举例（ADA0S 寄存器= 03H）

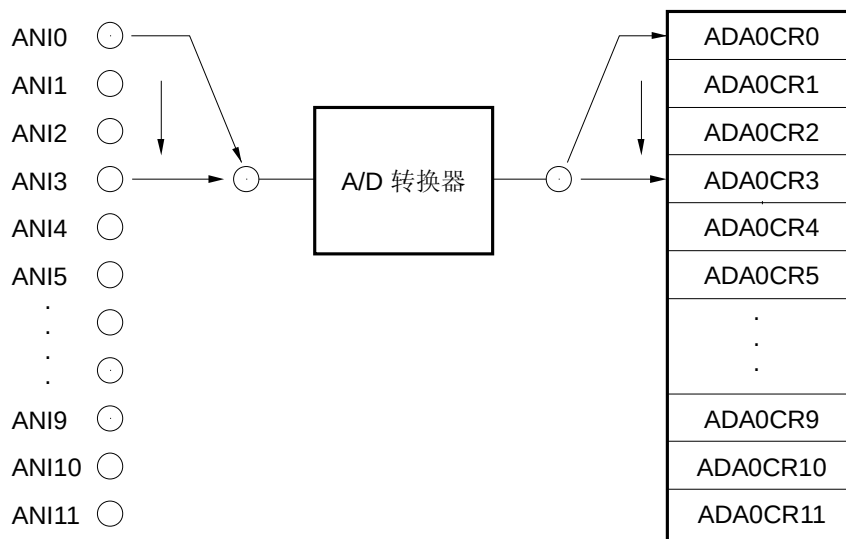
(a) 时序举例



(b) 框图

模拟输入引脚

ADA0CRn 寄存器

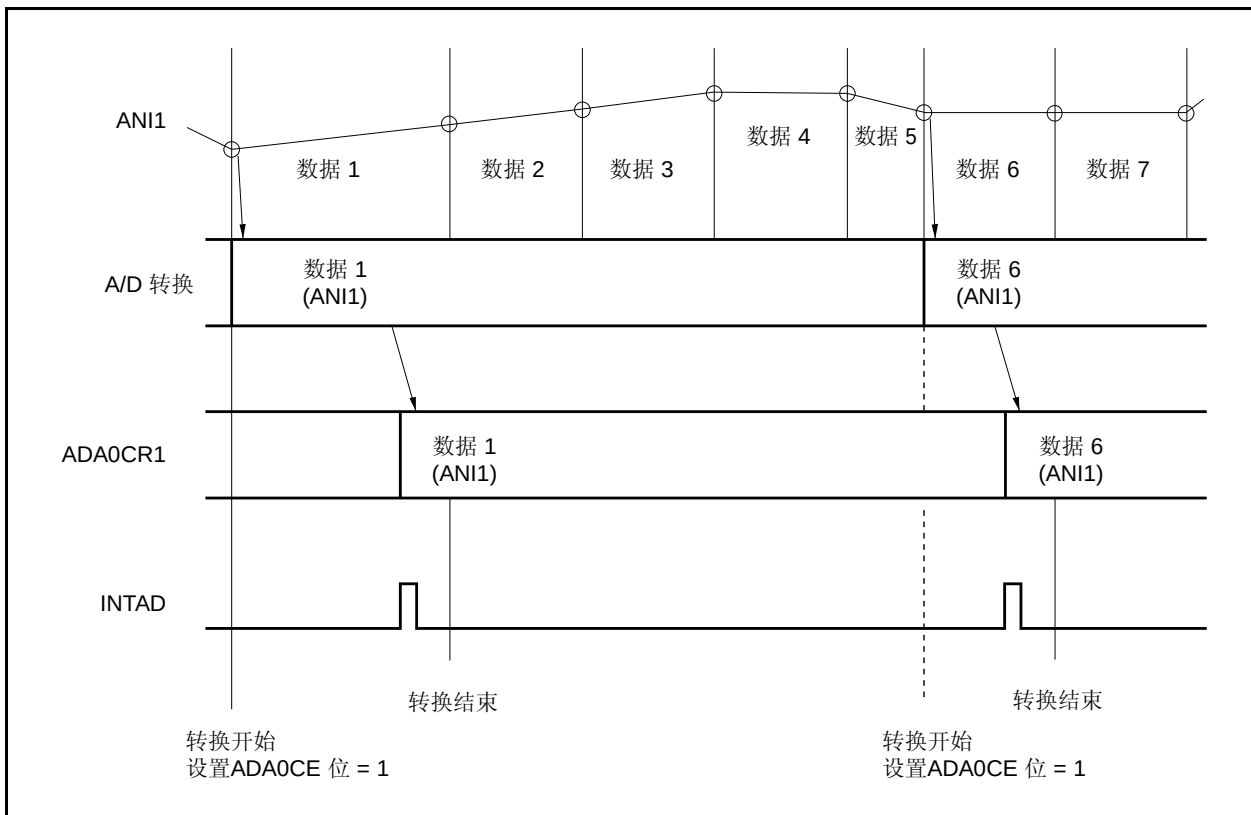


(3) 单次选择模式

该模式下，对通过 ADA0S 寄存器选择的模拟输入引脚电压仅进行一次转换为数字值的操作。

转换结果保存在与模拟输入引脚相对应的 ADA0CRn 寄存器中。该模式下，模拟输入引脚与 ADA0CRn 寄存器一一对应。A/D 转换完成一次后，产生 INTAD 信号。上述操作完成后，A/D 转换操作停止（n = 0 至 11）。

图 15-6. 单次选择模式操作的时序举例（ADA0S 寄存器= 01H）



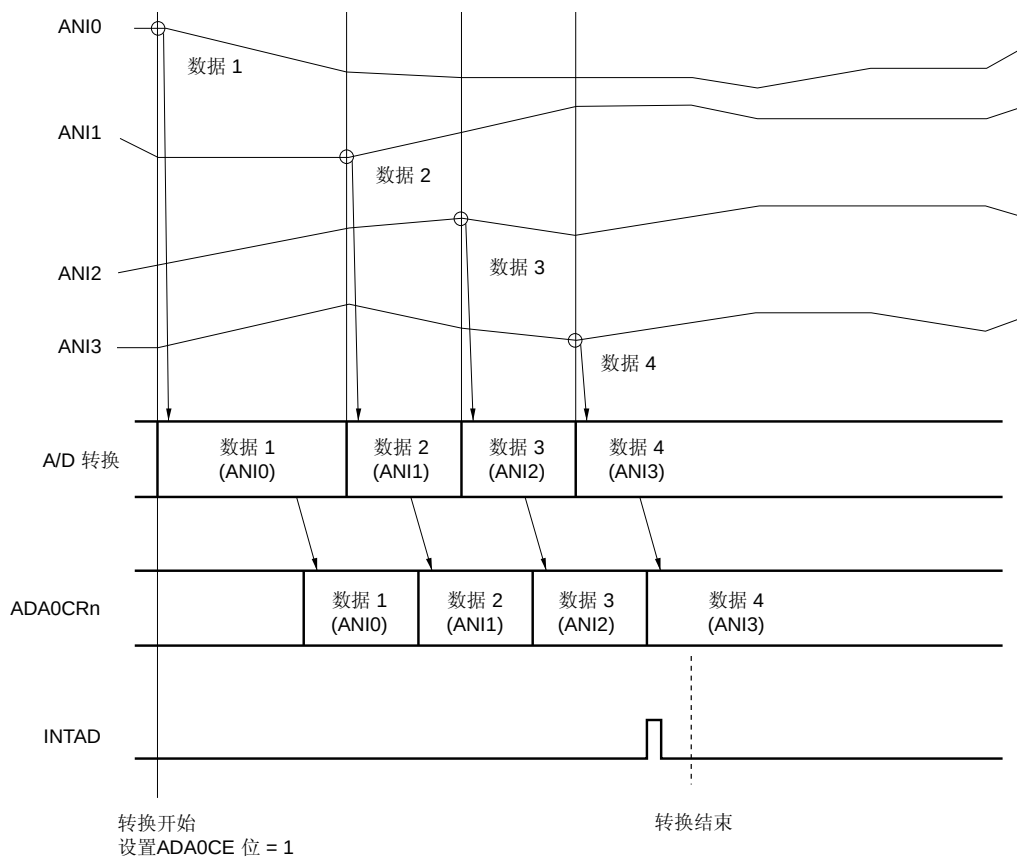
(4) 单次扫描模式

该模式下，模拟输入引脚从 ANI0 引脚至由 ADA0S 寄存器指定的引脚中连续选择，并且将它们值转换为数字值。

各个转换结果保存在与模拟输入引脚相对应的 ADA0CRn 寄存器中。当由 ADA0S 寄存器指定的模拟输入引脚的转换完成时，产生 INTAD 信号。上述操作完成后，A/D 转换停止（）。

图 15-7. 单次扫描模式操作的时序举例（ADA0S 寄存器= 03H）

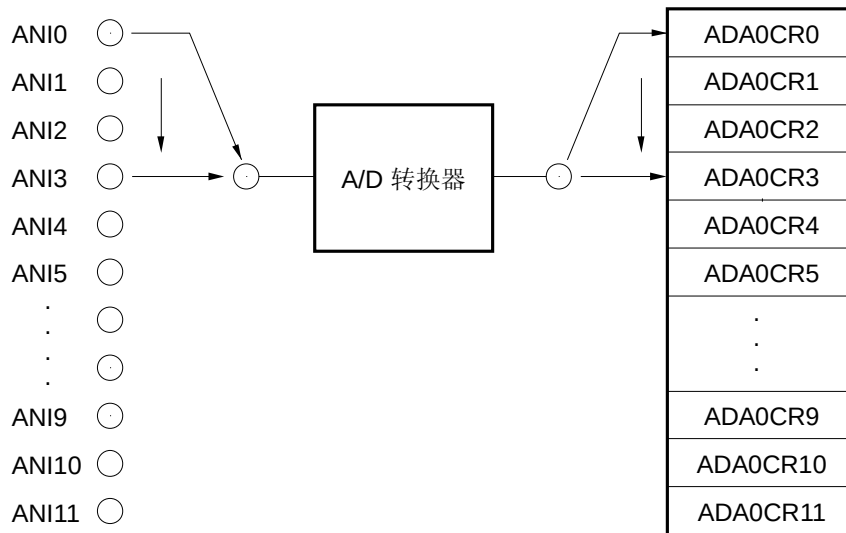
(a) 时序举例



(b) 框图

模拟输入引脚

ADA0CRn 寄存器



15.5.5 掉电比较模式

A/D 转换结束中断请求信号 (INTAD) 可以通过 ADA0PFM 寄存器和 ADA0PFT 寄存器控制。

- 当 ADA0PFM.ADA0PFE 位= 0 时，每次转换完成（正常使用 A/D 转换器）都产生 INTAD 信号。
- 当 ADA0PFE 位= 1 同时 ADA0PFM.ADA0PFC 位= 0 时，ADA0CRnH 寄存器的值与转换完成时 ADA0PFT 寄存器的值进行比较，并且仅当 $\text{ADA0CRnH} \geq \text{ADA0PFT}$ 时，产生 INTAD 信号。
- 当 ADA0PFE 位= 1，同时 ADA0PFC 位= 1 时，ADA0CRnH 寄存器的值与转换完成时 ADA0PFT 寄存器的值进行比较，并且仅当 $\text{ADA0CRnH} < \text{ADA0PFT}$ 时，产生 INTAD 信号。

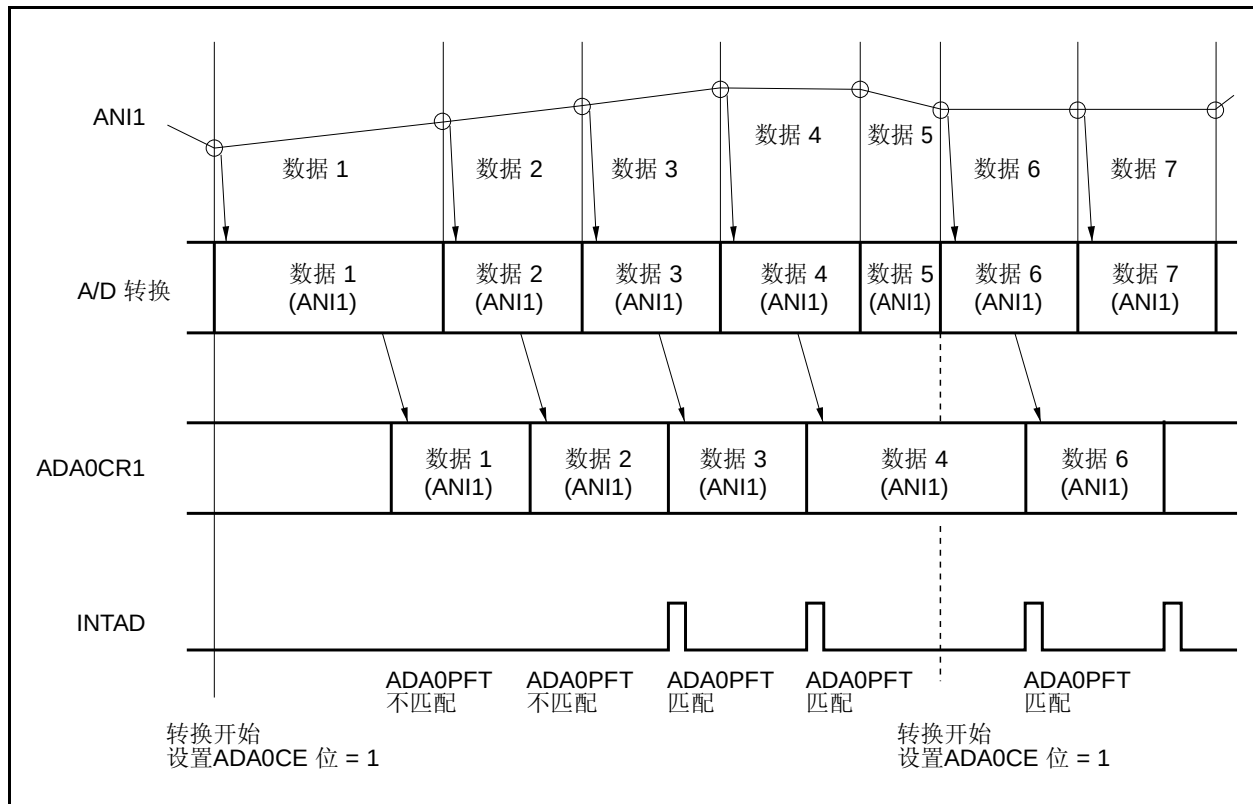
备注 n = 0 至 11

掉电比较模式中，有四个模式可供选择，在这四个操作模式中，需设置 ANI0 引脚至 ANI11 引脚。连续选择模式，连续扫描模式，单次选择模式以及单次扫描模式。

(1) 连续选择模式

该模式下，由 ADA0S 寄存器指定的模拟输入引脚的电压转换结果与 ADA0PFT 寄存器的设置值相比较。如果掉电比较的结果匹配由 ADA0PFC 位指定的状态，那么将转换结果存储在 ADA0CRn 寄存器，并产生 INTAD 信号。如果不匹配，转换结果仍存储在 ADA0CRn 寄存器中，但不产生 INTAD 信号。第一次转换完成后，则开始下一次转换，除非 ADA0M0.ADA0CE 位清为 0 (n = 0 至 11)。

图 15-8. 连续选择模式操作的时序举例
(掉电比较模式时：ADA0S 寄存器 = 01H)

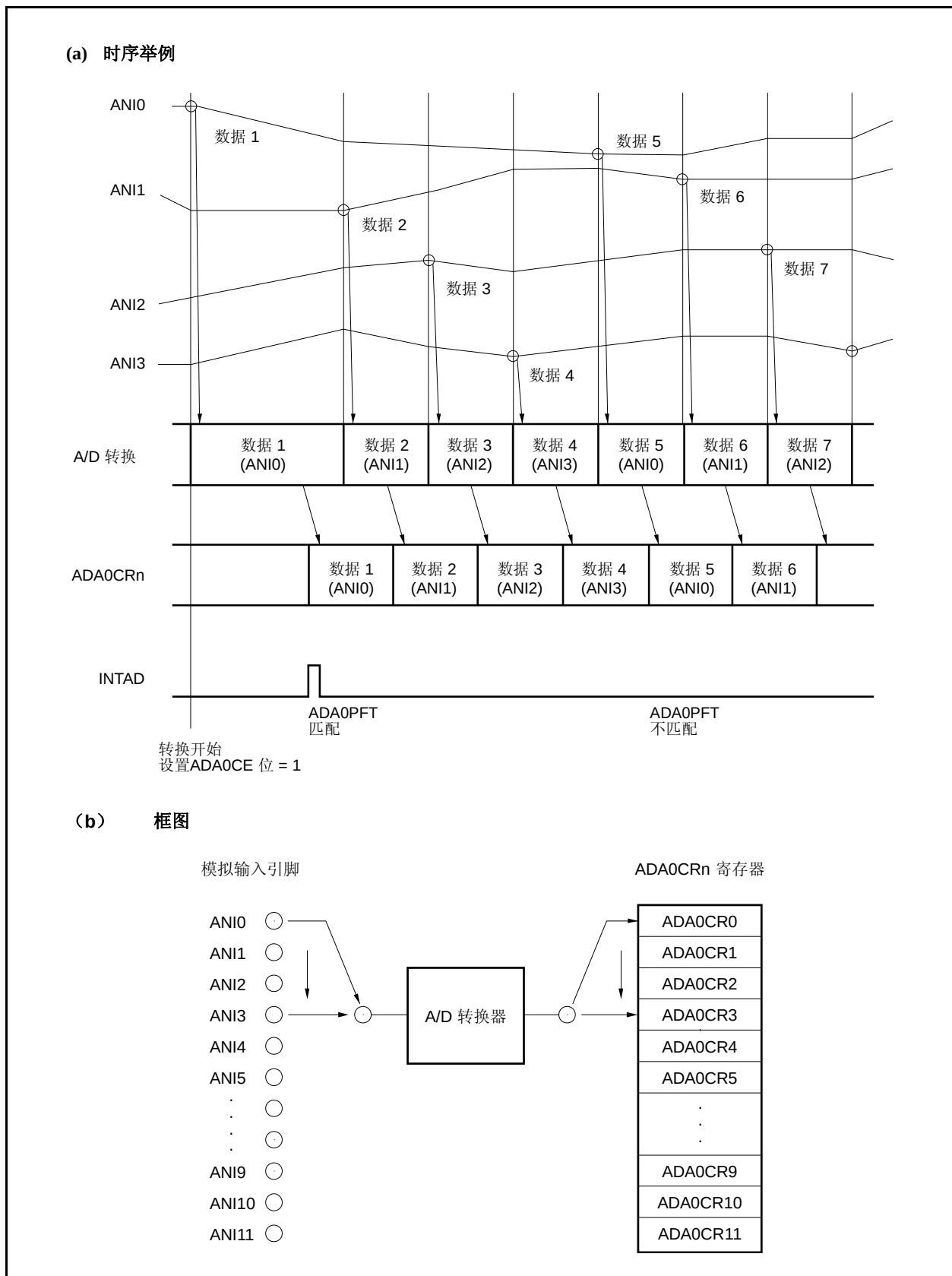


(2) 连续扫描模式

该模式下，将 ANI0 引脚中连续选择的模拟输入引脚的电压转换结果存储至由 ADA0S 寄存器指定的引脚，并且将通道 0 的 ADA0CR0H 寄存器的设置值与 ADA0PFT 寄存器的值作比较。如果掉电比较的结果匹配由 ADA0PFC 位指定的状态，那么将转换结果存储在 ADA0CR0 寄存器，并产生 INTAD 信号。如果不匹配，转换结果仍存储在 ADA0CR0 寄存器中，但不产生 INTAD 信号。

第一次转换结果存储在 ADA0CR0 寄存器后，连续存储关于模拟输入引脚至由 ADA0S 寄存器指定引脚的顺次转换电压结果。转换完成后，下一轮转换又从 ANI0 引脚开始，除非 ADA0CE 位清为 0。

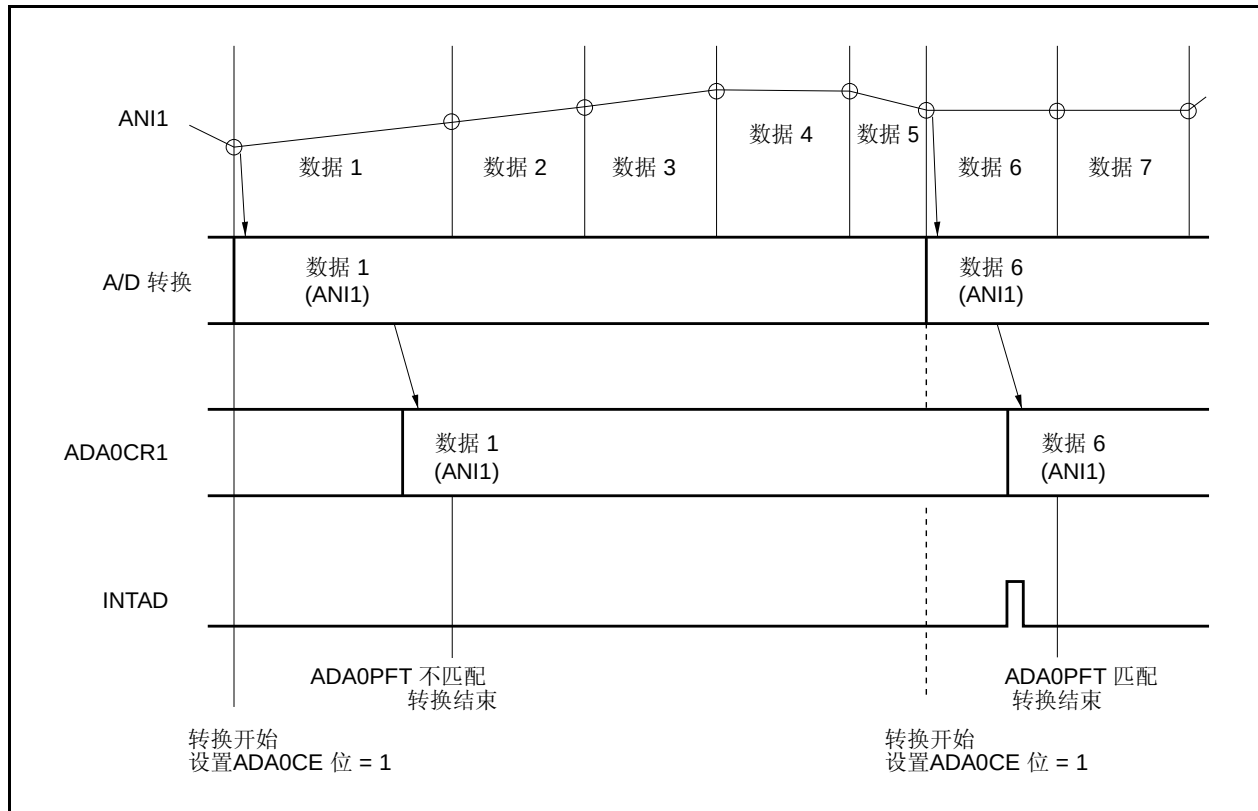
图 15-9. 连续扫描模式操作的时序举例
(掉电比较模式时: ADA0S 寄存器 = 03H)



(3) 单次选择模式

该模式下，由 ADA0S 寄存器指定的模拟输入引脚的电压转换结果与 ADA0PFT 寄存器的设置值相比较。如果掉电比较的结果匹配由 ADA0PFC 位指定的状态，那么将转换结果存储在 ADA0CRn 寄存器，并产生 INTAD 信号。如果不匹配，转换结果仍存储在 ADA0CRn 寄存器中，但不产生 INTAD 信号。上述操作完成后，停止转换。

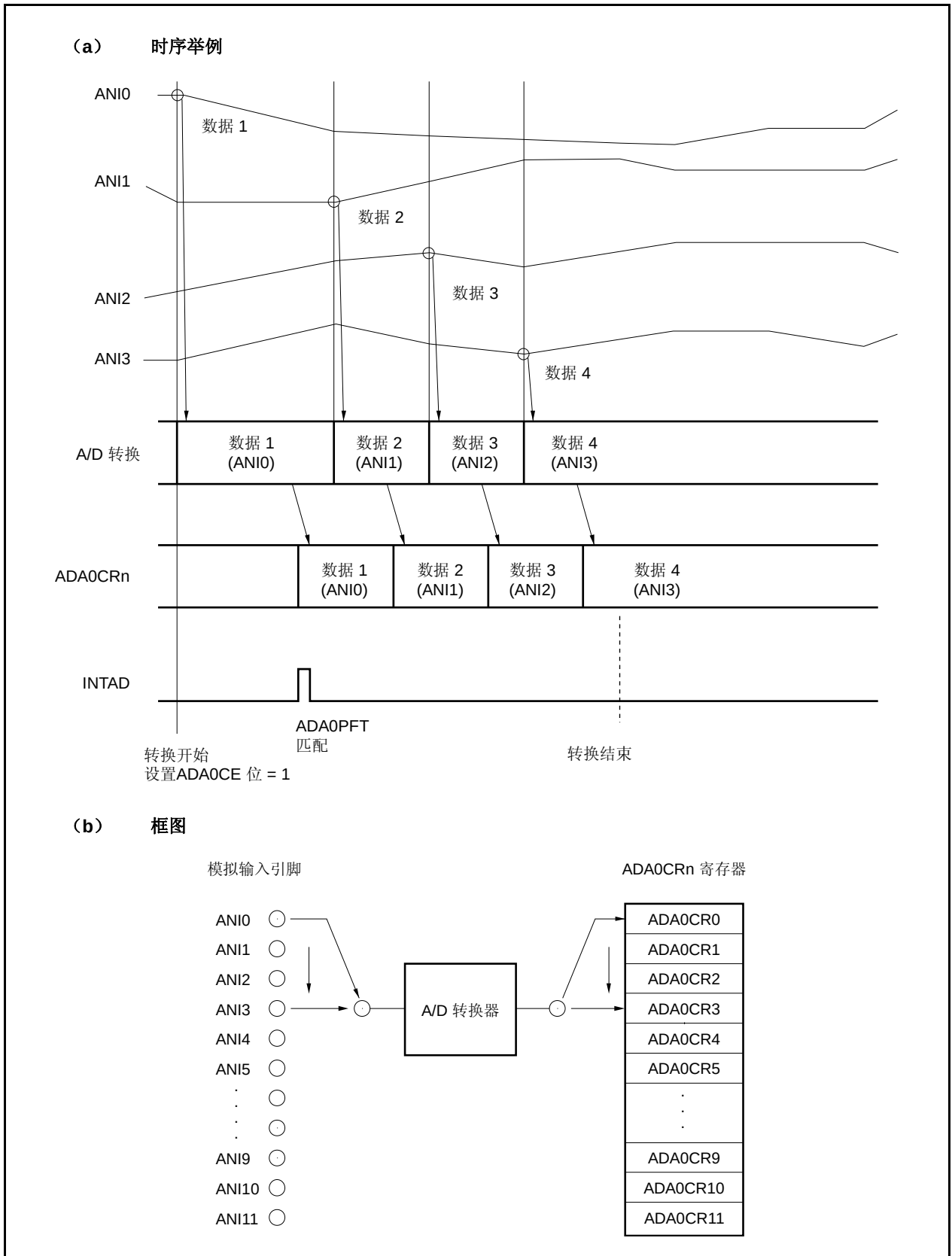
图 15-10. 单次选择模式操作的时序举例
(掉电比较模式时：ADA0S 寄存器 = 01H)



(4) 单次扫描模式

该模式下，将 ANI0 引脚中连续选择的模拟输入引脚的电压转换结果存储至由 ADA0S 寄存器指定的引脚，并且将通道 0 的 ADA0CR0H 寄存器的设置值与 ADA0PFT 寄存器的值作比较。如果掉电比较的结果匹配由 ADA0PFC 位指定的状态，那么将转换结果存储在 ADA0CR0 寄存器，并产生 INTAD 信号。如果不匹配，转换结果仍存储在 ADA0CR0 寄存器中，但不产生 INTAD0 信号。第一次转换结果存储在 ADA0CR0 寄存器后，依次存储由 ADA0S 寄存器指定的模拟输入引脚的转换信号结果。上述操作完成后，停止转换。

图 15-11. 单次扫描模式操作的时序举例
(掉电比较模式时: ADA0S 寄存器 = 03H)



15.6 注意事项

(1) 不使用 A/D 转换器时

不使用 A/D 转换器时，通过将 ADA0M0.ADA0CE 位清 0 来降低功耗。

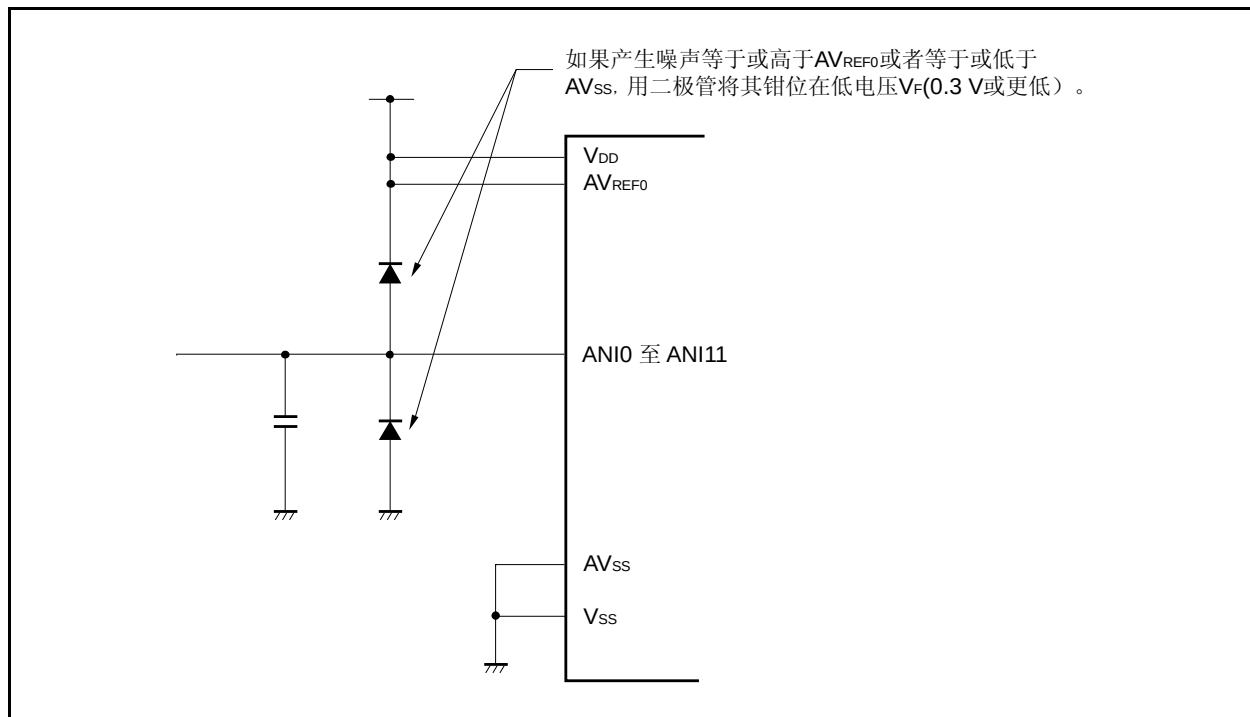
(2) ANI0 至 ANI11 引脚输入范围

向 ANI0 引脚至 ANI11 引脚输入指定范围内的电压。如果高于等于 AV_{REF0} 或低于等于 AV_{SS} （即使在最大额定值范围内）的电压输入到这些引脚中的任意引脚，则那个通道的转换值将不确定，而其它通道的转换值也会受到影响。

(3) 应对噪声的措施

为了维持 10 位分辨率，必须有效地防止噪声对 ANI0 引脚至 ANI11 引脚的干扰。噪声的影响随着模拟输入源输出阻抗的增高而加强。为了减小噪声，如图 15-12 中所建议的方式连接一个外部电容器。

图 15-12. 模拟输入引脚的处理



(4) 复用 I/O

模拟输入引脚（ANI0 至 ANI11）的复用功能作为端口引脚。当选择 ANI0 引脚至 ANI11 引脚之一来执行 A/D 转换时，因为转换的分辨率可能下降，所以转换期间不执行读取输入端口或写入输出端口的指令。

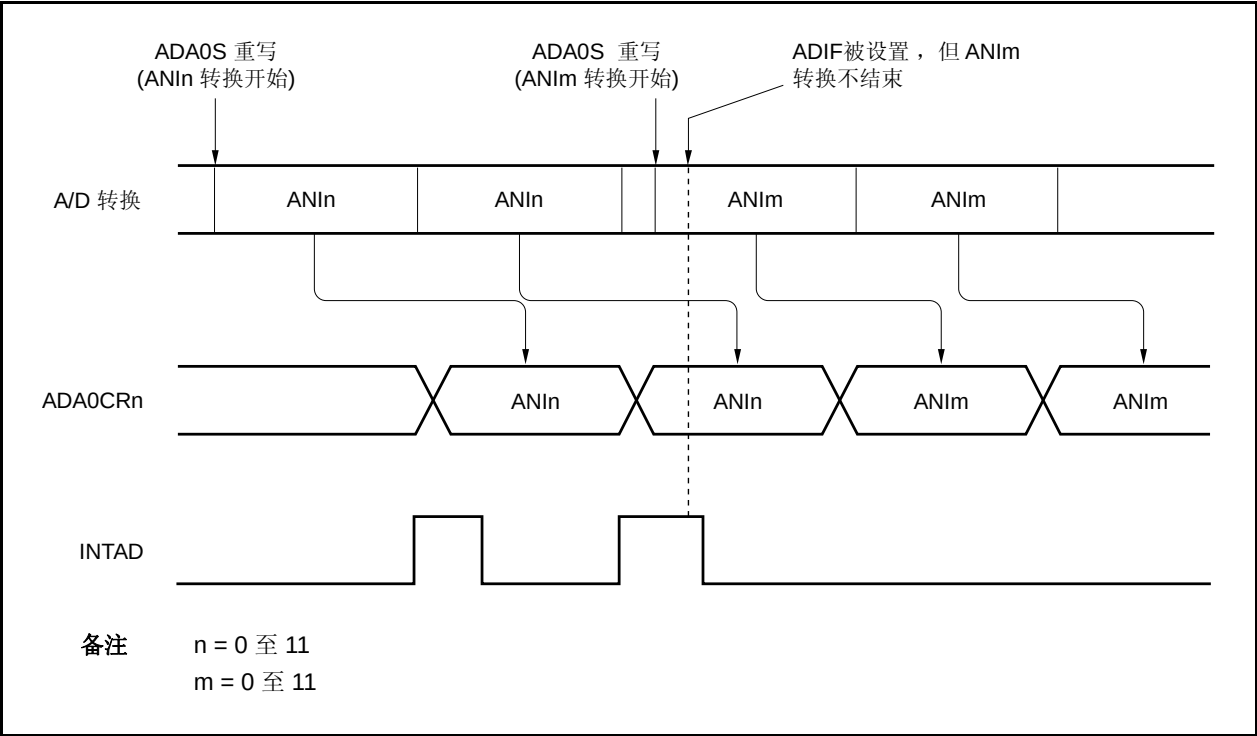
A/D 转换期间，如果当前输出由于连接至端口引脚的外部电路而改变，那么在设置为输出端口引脚处的转换分辨率也会下降。

如果将数字脉冲施加在与输入信号正被转换的引脚相邻的引脚上，由于耦合噪声的影响，A/D 转换值可能会无法预料。因此，不要把脉冲加到正在进行 A/D 转换引脚的邻近引脚上。

(5) 中断请求标志 (ADIF)

即使 ADA0S 寄存器的内容改变，也不会清除中断请求标志 (ADIF)。因此，A/D 转换期间如果改变模拟输入引脚，那么存储先前所选模拟输入引脚信号的转换结果，且 ADA0S 寄存器写入之前，立即设置转换结束中断请求标志。如果在 ADA0S 寄存器写入之后立即读取 ADIF 标志，那么，即使最新选择的模拟输入引脚的 A/D 转换还没有完成，也会设置 ADIF 标志。A/D 转换停止时，重新开始转换之前清除 ADIF 标志。

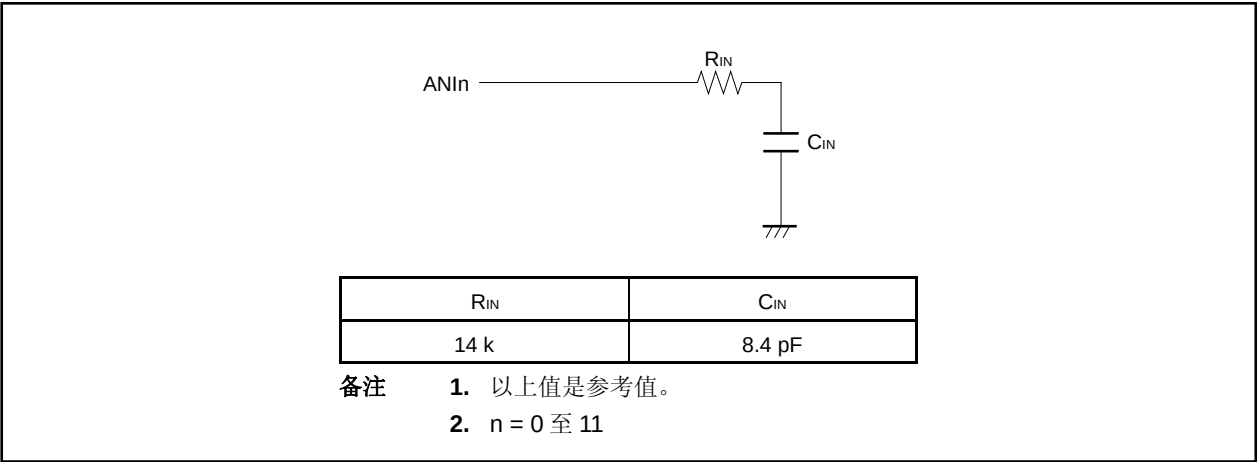
图 15-13. A/D 转换结束中断请求的发生时序



(6) 内部等效电路

以下显示了模拟输入模块的等效电路

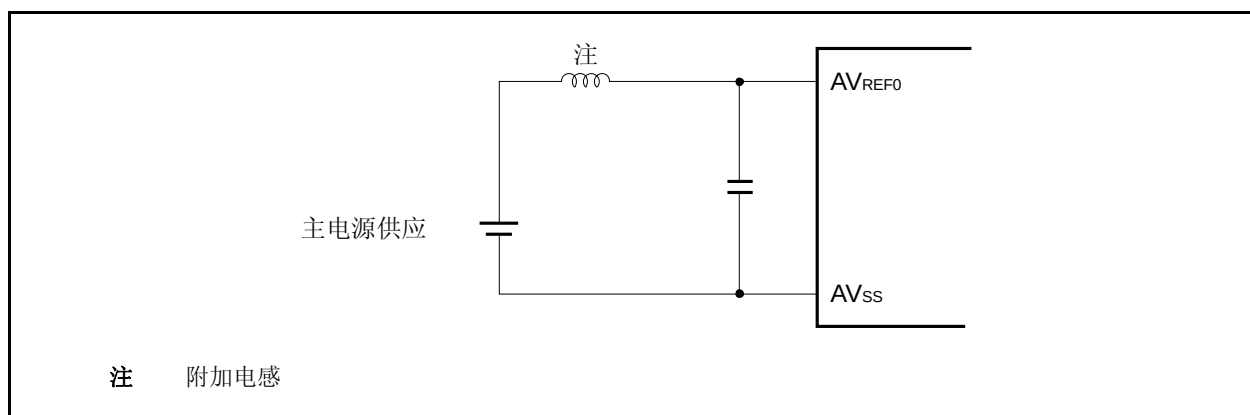
图 15-14. ANIn 引脚的内部等效电路



(7) AV_{REF0} 引脚

- (a) AV_{REF0} 引脚用作 A/D 转换器的电源，也给复用功能端口提供电源。使用备用电源时，确保提供至 AV_{REF0} 引脚的电压与 V_{DD} 相同，如图 15-15 所示。
- (b) AV_{REF0} 引脚用作 A/D 转换器的参考电压引脚。如果 AV_{REF0} 引脚的供应电源具有高阻抗或者低供电电流性能，那么转换期间，由于流过电流的影响，参数电压可能会波动（尤其在紧接转换操作使能位 $ADA0CE$ 已经设为 1 之后）。结果，转换精度可能下降。为了避免这种情况，建议在 AV_{REF0} 引脚和 AV_{SS} 引脚之间连接一个电容器来抑制参考电压的波动，如图 15-15 所示。
- (c) 如果 AV_{REF0} 引脚的供应电源具有一个高 DC 电阻（例如，由于二极管的插入），由于 A/D 转换电流引起电压下降，所以转换使能时的电压可能低于转换停止时的电压。

图 15-15. AV_{REF0} 引脚处理实例



(8) 读取 $ADA0CRn$ 寄存器

当写入 $ADA0M0$ 至 $ADA0M2$ ， $ADA0S$ ， $ADA0PFM$ 或 $ADA0PFT$ 寄存器时， $ADA0CRn$ 寄存器的内容可能不确定。在转换完成之后且写入 $ADA0M0$ 至 $ADA0M2$ ， $ADA0S$ ， $ADA0PFM$ 或 $ADA0PFT$ 寄存器之前，读取转换结果。当响应外部/定时器触发时， $ADA0CRn$ 寄存器的内容也可能不确定。在转换完成之后且下一次外部/定时器触发响应之前，读取转换结果。不同于上述时序时，可能读取不到正确的转换结果。

(9) 待机模式

在 STOP 模式下，由于 A/D 转换器停止工作，转换结果无效，因此能够降低功耗。STOP 模式解除后，操作重新开始，但是 STOP 模式解除后的 A/D 转换结果无效。STOP 模式解除后使用 A/D 转换器时，在设置 STOP 模式之前或解除 STOP 模式后，将 ADA0M0.ADA0CE 位清 0，然后在解除 STOP 模式后，将 ADA0CE 位设为 1。

在 IDLE1、IDLE2 或副时钟操作模式下，操作继续。因此，为了降低功耗，将 ADA0M0.ADA0CE 位清 0。在 IDLE1 和 IDLE2 模式下，由于不能够保留模拟输入引脚的电压值，所以 IDLE1 和 IDLE2 模式解除后，A/D 转换结果无效。设置 IDLE1 和 IDLE2 模式之前的转换结果是有效的。

(10) 高速转换模式下

在高速转换模式下，振荡稳定期间，禁止触发输入和对 ADA0M0、ADA0M2、ADA0S、ADA0PFM 和 ADA0PFT 寄存器的重写。

(11) A/D 转换时间

A/D 转换时间为稳定时间，转换时间，等待时间和触发应答时间的总和（关于这些时间的细节，参见表 15-2 正常转换模式（ADA0HS1 位= 0）下转换时间的选择和表 15-3 高速转换模式（ADA0HS1 位 = 1）下转换时间的选择。

在正常转换模式下，A/D 转换期间，如果写入 ADA0M0、ADA0M2、ADA0S、ADA0PFM 以及 ADA0PFT 寄存器或者输入触发信号，则重新执行转换。但是，如果稳定时间结束时序与写入这些寄存器发生冲突，或者稳定时间结束时序与触发信号输入发生冲突，那么将会插入 64 个时钟的稳定时间。

如果与插入的稳定时间结束时序再次发生冲突，则仍然插入稳定时间。因此，不要将触发输入间隔和控制寄存器写入间隔设置为 64 个时钟或更少。

(12) A/D 转换结果的变化

由于供应电压波动或噪声影响 A/D 转换结果可能不同。为了减小变化，在程序中应用抵消措施，例如平均 A/D 转换结果。

(13) A/D 转换结果滞后特性

逐次比较类型 A/D 转换器在内部采样&保持电容器上保持输入电压，然后进行 A/D 转换。完成后，模拟输入电压仍然保持在内部采样&保持电容器上。因此，可能发生以下现象：

- 当相同通道用于 A/D 转换时，如果电压比先前 A/D 转换的电压高或低，滞后特性可能出现在转换结果被先前值影响的地方。因此，即使转换是在相同的电压下进行，结果也可能不同。
- 当转换模拟输入通道时，滞后特征出现可能在转换结果受到先前通道值影响的地方。这是因为只有一个 A/D 转换器用于 A/D 转换。因此，即使转换是在相同的电压下进行，结果也可能不同。

15.7 A/D转换器特征表的阅读方法

本部分描述了 A/D 转换器的相关术语。

(1) 分辨率

此为能够识别的最低模拟输入电压，也就是说，每位数字输出所占模拟输入电压的百分比，叫作 **1 LSB**（最低有效位）。最大范围的 **1 LSB** 比率是由 %FSR（最大范围）表示的。%FSR 为由百分比表示的可转换的模拟输入电压比率，不管分辨率如何，总是由以下的公式表示：

$$\begin{aligned} 1\%FSR &= (\text{可转换模拟输入电压的最大值} - \text{可转换模拟输入电压的最小值}) / 100 \\ &= (AV_{REF0} - 0) / 100 \\ &= AV_{REF0} / 100 \end{aligned}$$

当分辨率为 10 位时，1 LSB 表示如下：

$$\begin{aligned} 1 \text{ LSB} &= 1/2^{10} = 1/1,024 \\ &= 0.098\%FSR \end{aligned}$$

精确度是由总体的误差决定的，而与分辨率无关。

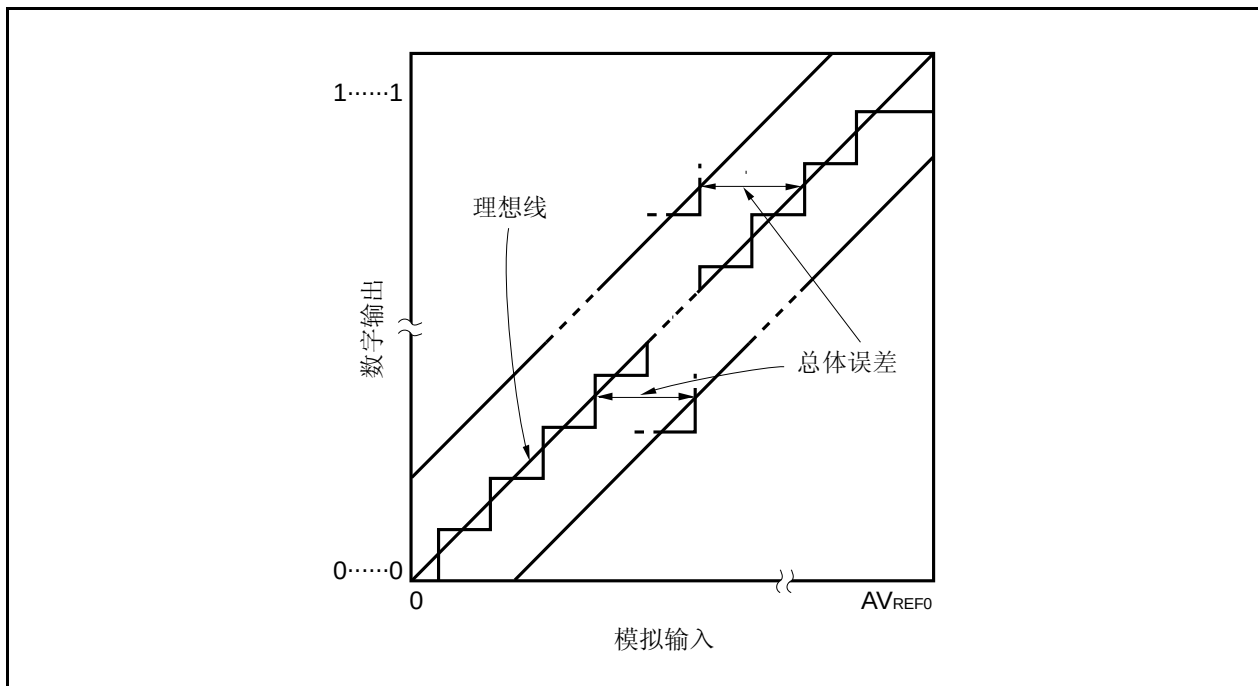
(2) 总误差

此项显示了实际测量值和理论值的最大误差值。

零标称误差、满量程误差、线性误差以及这些误差的综合构成总体误差。

在参数表中的总体误差里没有包含量化误差。

图 15-16. 总体误差

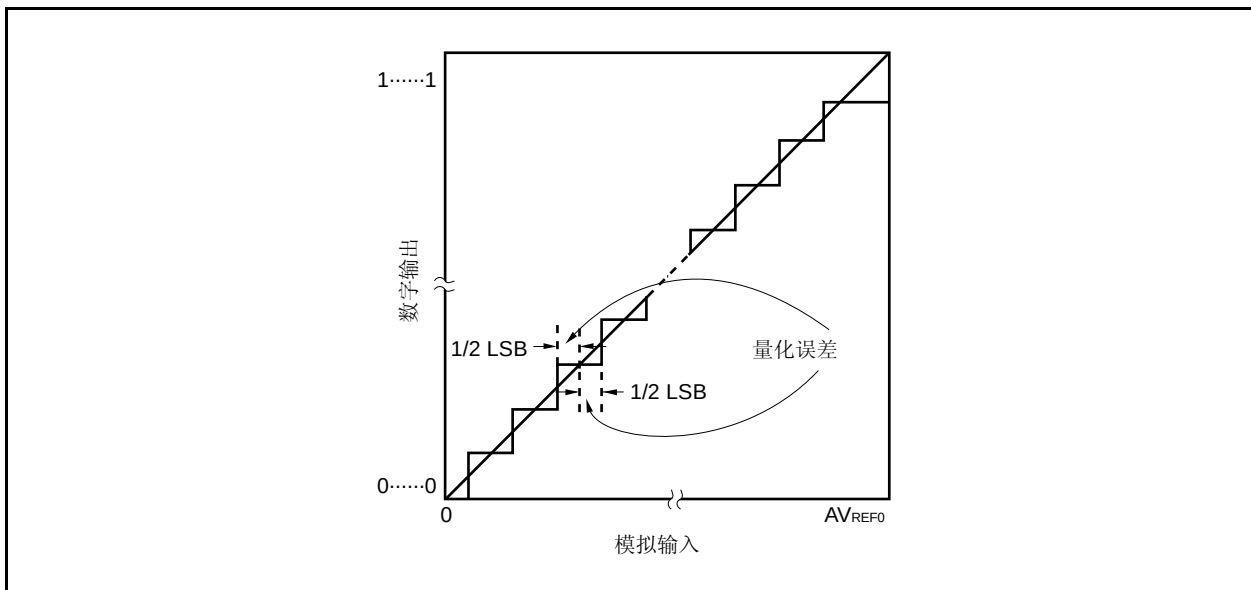


(3) 量化误差

当模拟值转换为数字值时，会产生 $\pm 1/2$ LSB 的误差。因为 A/D 转换器将在 $\pm 1/2$ LSB 范围内的模拟输入电压转换成相同的数字代码，所以量化误差不能避免。

量化误差不包括在参数表中的总体误差、零标称误差、满量程误差、积分线性误差或微分线性误差之中。

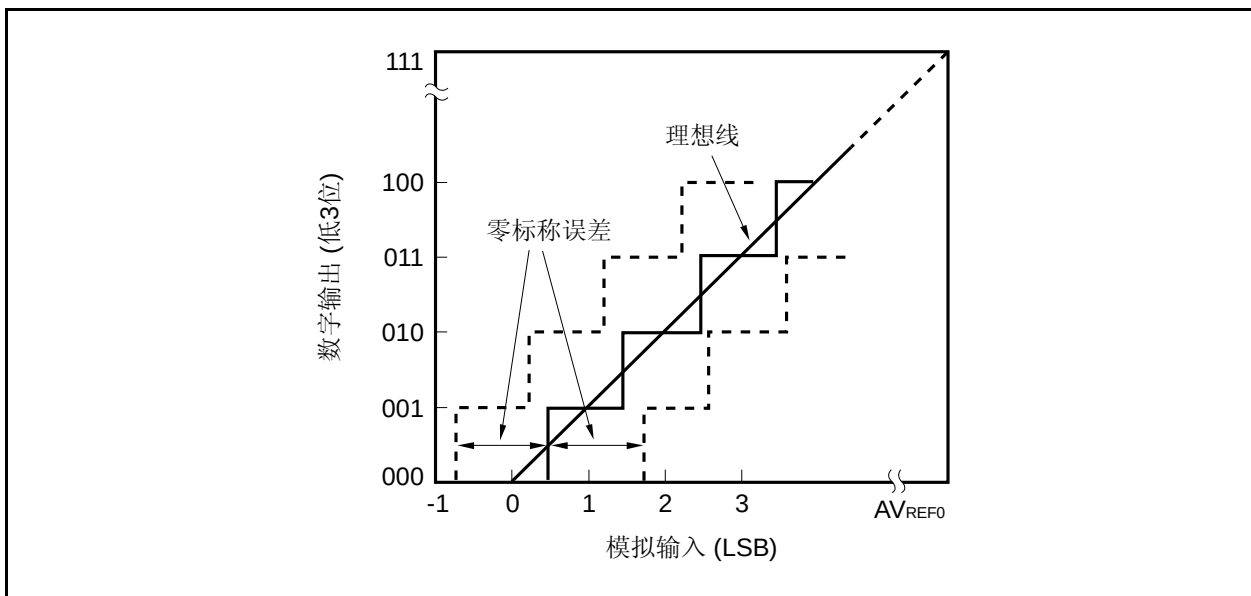
图 15-17. 量化误差



(4) 零标称误差

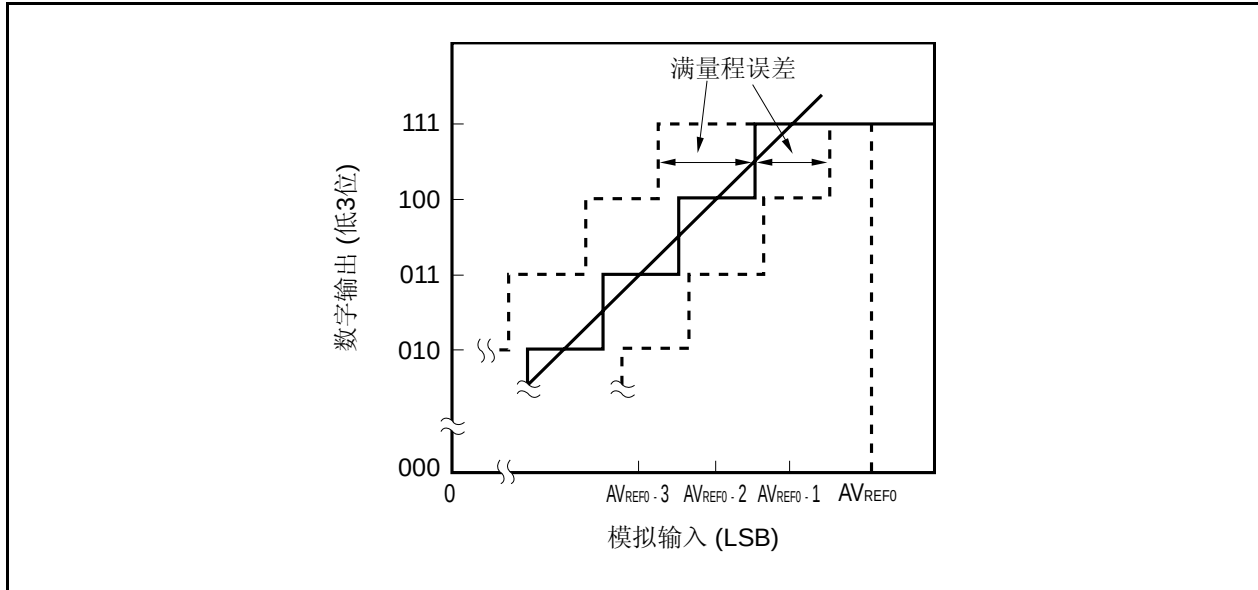
当数字输出从 0.....000 改变到 0.....001 ($1/2$ LSB) 时，这显示了实际测量的模拟输入电压和其理论值之间的差别。

图 15-18. 零标称误差

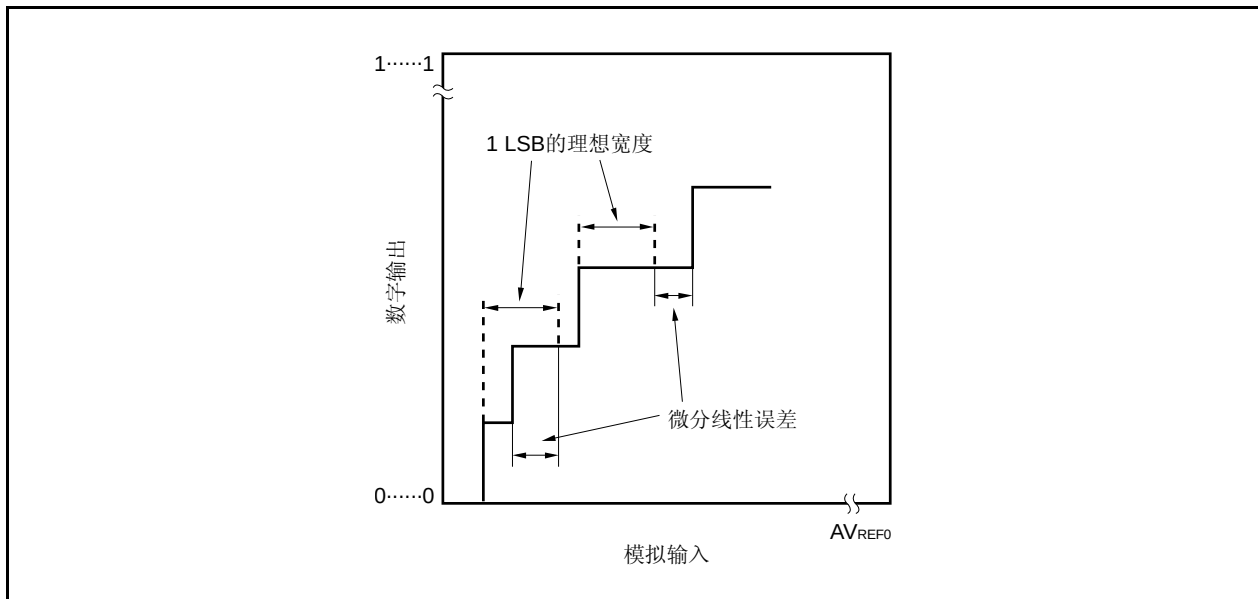


(5) 满量程误差

当数字输出从 1...110 改变到 1...111 (满量程- 3/2 LSB) 时, 这显示了实际测量的模拟输入电压和其理论值之间差别。

图 15-19. 满量程误差**(6) 微分线性误差**

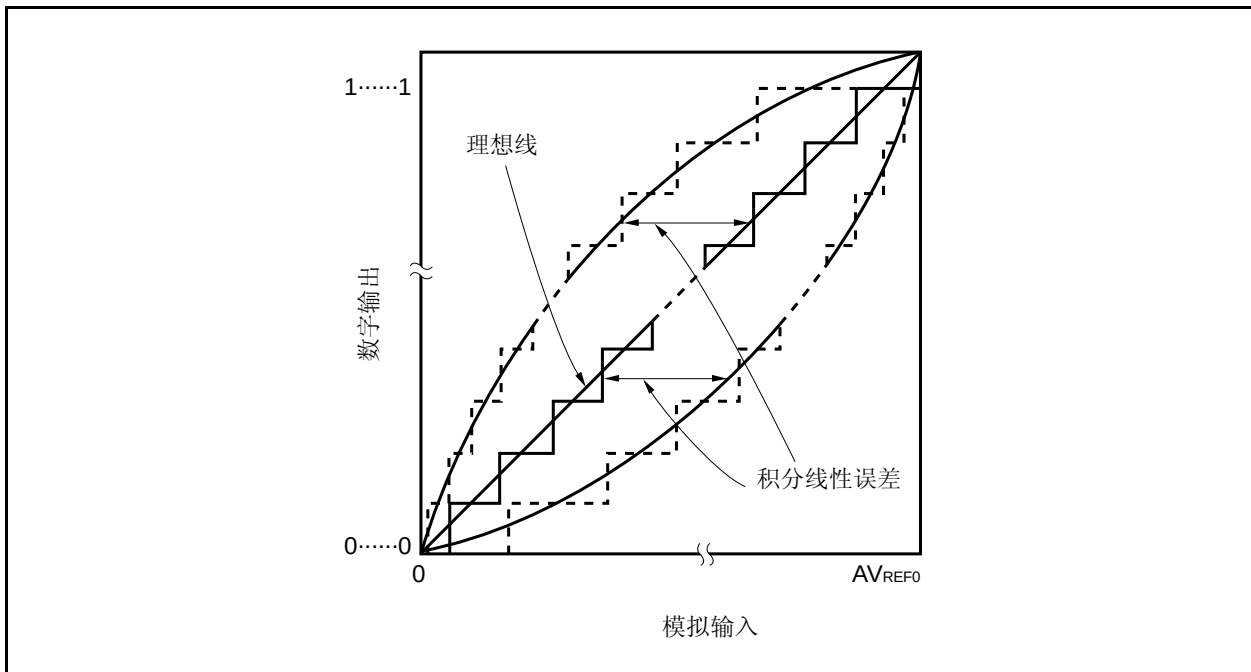
理想状态下, 输出指定编码的宽度是 1 LSB。输出指定编码时, 该误差显示了实际测量值和其理论值之间的差别。当输入到相同通道的模拟输入引脚的电压从 AV_{SS} 一位一位增加到 AV_{REF0} , 这代表了 A/D 转换的基本特性。当输入电压增加或减小时, 或当使用两个或更多的通道时, 参见 15.7 (2) 总体误差。

图 15-20. 微分线性误差

(7) 积分线性误差

此误差显示了转换参数偏离理想线性关系的程度。它表示当零标称误差和满量程误差均为 0 时，实际测量值和其理论值之间的最大误差。

图 15-21. 积分线性误差



(8) 转换时间

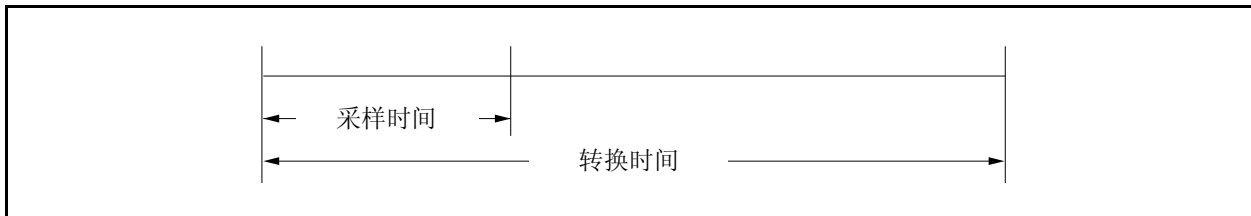
此为各触发信号产生后获得数字输出所需的时间。

参数表中，转换时间包含采样时间。

(9) 采样时间

采样时间，即模拟开关开启，将模拟电压载入采样&保持电路的时间。

图 15-22. 采样时间



16.1 功能

D/A 转换器具有以下功能：

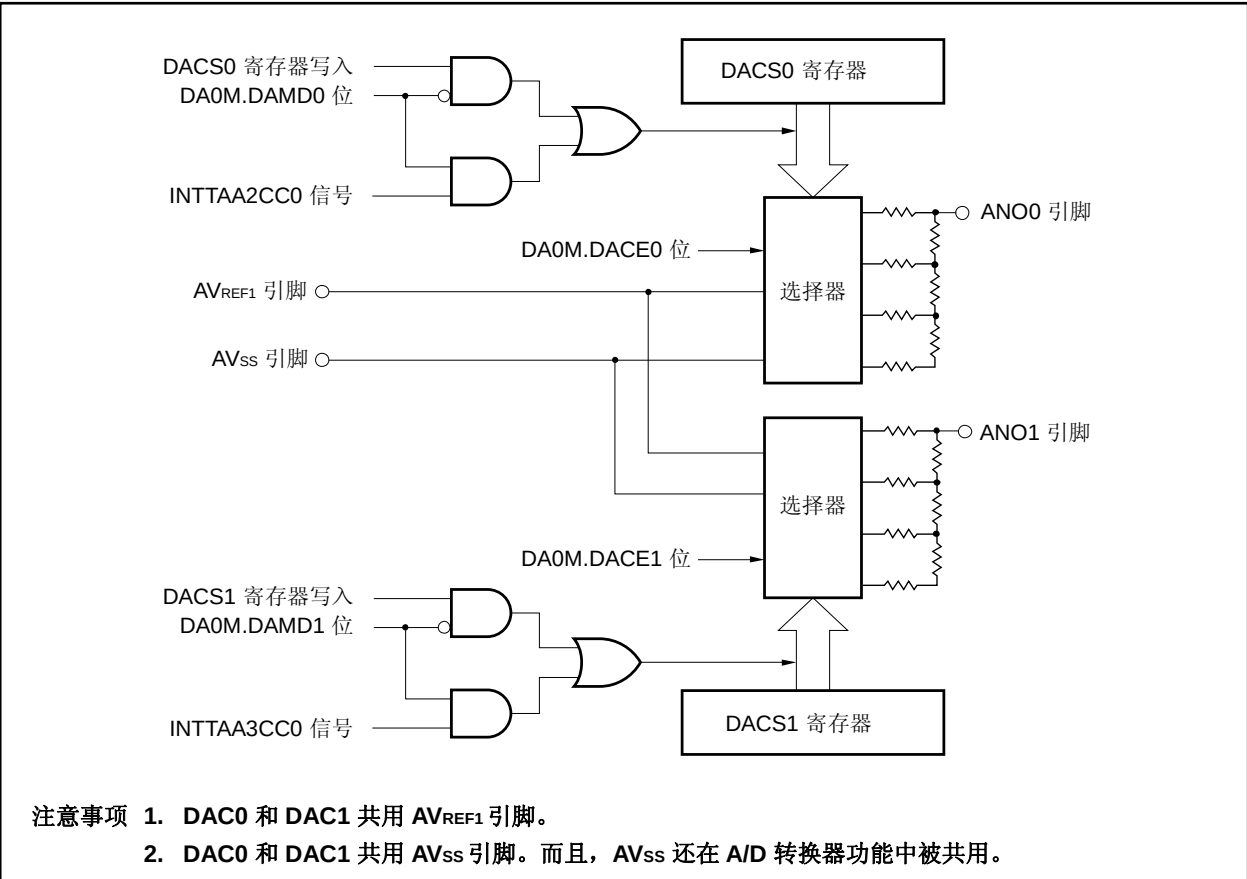
- 8 位精度×2 通道（DA0CS0， DA0CS1）
- R-2R 阶梯变换方法
- 设置时间：最大 3 μ S （当 AVREF1 为 3.0 至 3.6 V 外部负载为 20 pF）
- 模拟输出电压： $AV_{REF1} \times m/256$ （m = 0 至 255; 设置给 DA0CSn 寄存器的值）
- 操作模式： 正常模式， 实时输出模式

备注 n = 0, 1

16.2 配置

D/A 转换器配置如下所示：

图 16-1. D/A 转换器功能框图



D/A 转换器包括以下硬件：

表 16-1. D/A 转换器的配置

项目	配置
控制寄存器	D/A 转换器模式寄存器 （DA0M）
	D/A 转换值设置寄存器 0， 1 （DA0CS0， DA0CS1）

16.3 寄存器

控制 D/A 转换器的寄存器如下所示：

- D/A 转换器模式寄存器 （DA0M）
- D/A 转换值设置寄存器 0， 1 （DA0CS0， DA0CS1）

(1) D/A 转换器模式寄存器 （DA0M）

DA0M 寄存器控制 D/A 转换器的操作。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: FFFFF282H

DA0M

7	6	<5>	<4>	3	2	1	0
0	0	DA0CE1	DA0CE0	0	0	DA0MD1	DA0MD0

DA0CEn	D/A转换器操作使能/禁止控制 (n = 0, 1)
0	禁止操作
1	允许操作

DA0MDn	D/A 转换器操作模式的选择 (n = 0, 1)
0	正常模式
1	实时输出模式 ^注

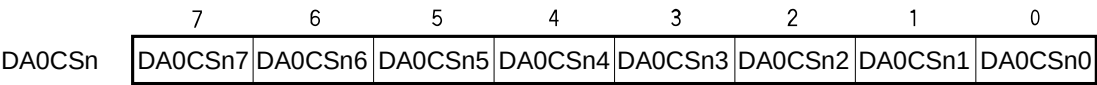
注 实时输出模式（DA0MDn 位 = 1）的输出触发信号如下：

- 当 n = 0 时：INTTAA2CC0 信号（参见第七章 16 位定时器/事件计数器 AA（TAA））。
- 当 n = 1 时：INTTAA3CC0 信号（参见第七章 16 位定时器/事件计数器 AA（TAA））。

(2) D/A 转换值设置寄存器 0，1 (DA0CS0， DA0CS1)

DA0CS0 和 DA0CS1 寄存器设定 ANO0 和 ANO1 引脚输出的模拟电压值。
这些寄存器可以按 8 位读取或写入。
复位后这些寄存器设置为 00H。

复位后: 00H R/W 地址: DA0CS0 FFFFF280H, DA0CS1 FFFFF281H



注意事项 实时输出模式 (DA0M.DA0MDn 位 = 1) 时，在 INTTAA2CC0/INTTAA3CC0 信号产生以前，设置 DA0CSn 寄存器。当 INTTAA2CC0/INTTAA3CC0 信号产生时，D/A 转换器开始工作。

备注 n = 0， 1

16.4 操作

16.4.1 正常模式下的操作

将 DA0CSn 寄存器的写入操作作为触发信号，执行 D/A 转换器操作。

设置方法如下：

- <1> 清除 DA0M.DA0MDn 位为 0（正常模式）。
- <2> 将 ANOn 引脚待输出的模拟电压值写入 DA0CSn 寄存器。
以上步骤<1>和步骤<2>为初始设置。
- <3> 设置 DA0M.DA0CEn 位为 1（D/A 转换使能）。
执行该设置时启动 D/A 转换。
- <4> 写入 DA0CSn 寄存器进行接下来的 D/A 转换。
前一个 D/A 转换结果一直保持到进行下一个 D/A 转换。

- 备注**
- 1. 关于复用功能引脚的设置，参见表 4-20 用作复用功能引脚的端口引脚。
 - 2. $n = 0, 1$

16.4.2 实时输出模式下的操作

使用 TAA2 和 TAA3 的中断请求信号（INTTAA2CC0 和 INTTAA3CC0）作为触发信号进行 D/A 转换。

设置方法如下：

- <1> 设置 DA0M.DA0MDn 位为 1（实时输出模式）。
- <2> 将 ANOn 引脚待输出的模拟电压值写入 DA0CSn 寄存器。
- <3> 设置 DA0M.DA0CEn 位为 1（D/A 转换使能）。
以上步骤<1>至步骤<3>为初始设置。
- <4> 操作 TAA2 和 TAA3。
- <5> 当 INTTAA2CC0 和 INTTAA3CC0 信号产生时，D/A 转换器开始工作。
- <6> 此后，设置给 DA0CSn 寄存器的值在每次产生 INTTAA2CC0 和 INTTAA3CC0 信号时输出。

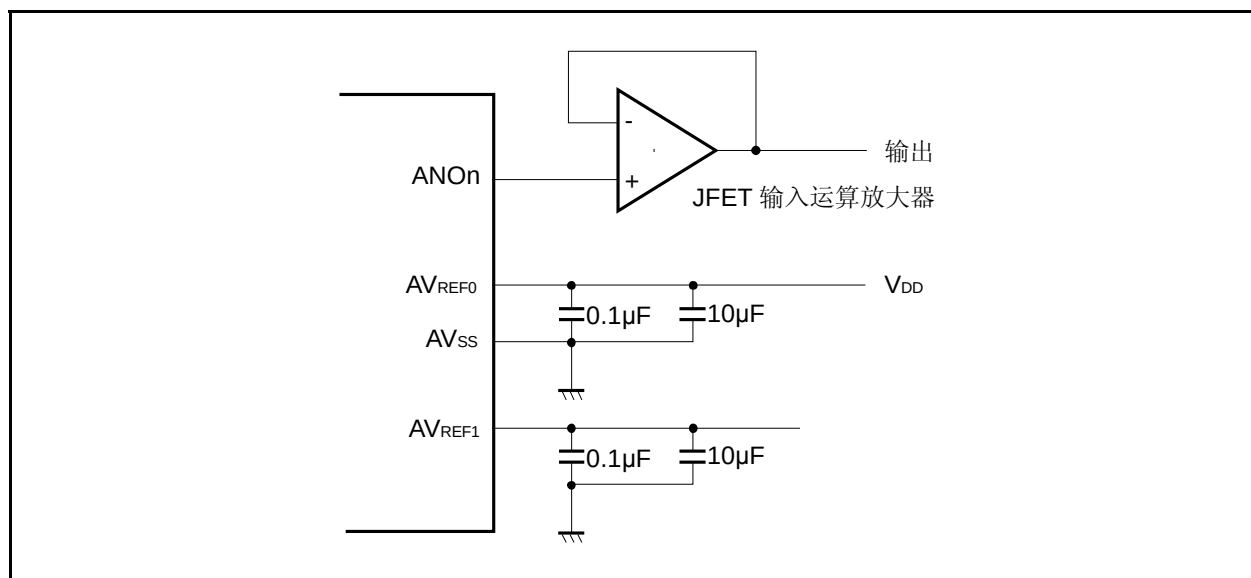
- 备注**
- 1. 步骤<5>之前的 ANO0 和 ANO1 引脚输出值不确定。
 - 2. 关于 ANO0 和 ANO1 引脚在 HALT，IDLE1，IDLE2 和 STOP 模式时的输出值，参见 第二十一章 待机功能。
 - 3. 关于复用功能引脚的设置，参见表 4-20 用作复用功能引脚的端口引脚。

16.4.3 注意事项

使用 V850ES/JG3-H 和 V850ES/JH3-H 的 D/A 转换器时，请遵循以下注意事项：

- (1) 实时输出模式下触发信号发出时，不要改变 DA0CSn 寄存器的设定值。
- (2) 改变操作模式前，确保将 DA0M.DA0CEn 位清除为 0。
- (3) 当 P10/AN00 和 P11/AN01 其中之一用作输入/输出端口而另一个用作 D/A 输出引脚时，确保 D/A 输出期间输入/输出端口电平不改变。
- (4) 确保 $AV_{REF0} = V_{DD} = AV_{REF1} = 3.0$ 至 3.6 V。如果超出该范围，则操作得不到保证。
- (5) 在给 AV_{REF0} 供电的同时给 AV_{REF1} 供电。
- (6) 因为 D/A 转换器的输出阻抗为高电平，所以 ANOn 引脚 ($n = 0, 1$) 没有电流输出。如果连接 $2\text{ M}\Omega$ 或更小的电阻，需在电阻和 ANOn 引脚之间插入一个 JFET 输入运算放大器。

图 16-2. 外部引脚连接举例



- (7) 因为 STOP 模式下 D/A 转换器停止操作，ANO0 和 ANO1 引脚进入高阻抗状态，因而功耗降低。在 IDLE1、IDLE2 或副时钟操作模式下，操作继续。所以为降低功耗，将 DA0M.DA0CEn 位清除为 0。

第十七章 异步串行接口 C（UARTC）

V850ES/JG3-H 和 V850ES/JH3-H 有一个 5 通道 UARTC。

17.1 特性

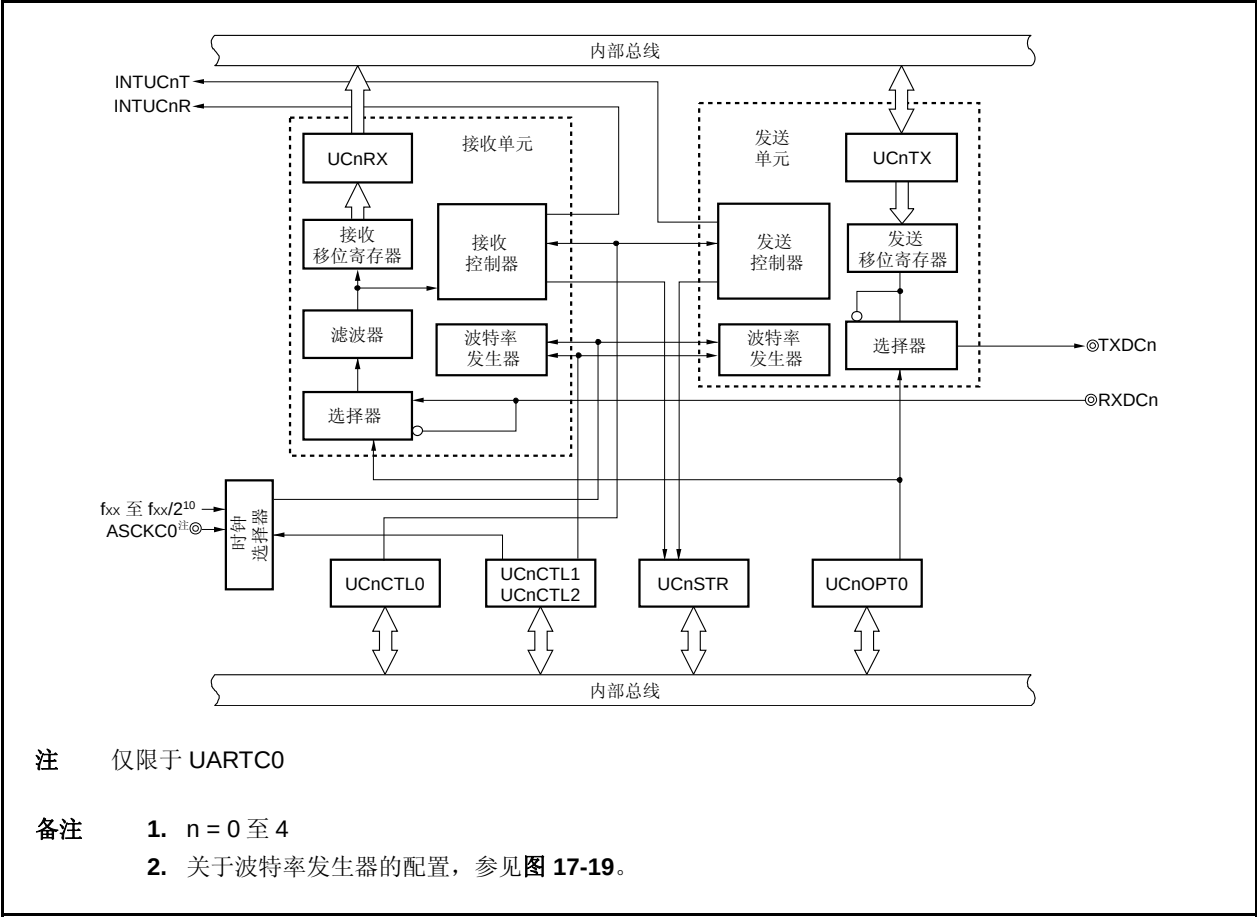
- 传输速率：300 bps 至 1.25 Mbps（使用 24 MHz 的内部系统时钟和专用波特率发生器）
- 全双工通信：
 - 内部 UARTCn 接收数据寄存器（UCnRX）
 - 内部 UARTCn 发送数据寄存器（UCnTX）
- 2 引脚配置：
 - TXDCn：发送数据输出引脚
 - RXDCn：接收数据输入引脚
- 接收错误校验功能
 - 奇偶错误
 - 帧错误
 - 超限错误
- 中断源：2 种类型
 - 接收完成中断（INTUCnR）：
 - 接收使能状态下，串行传输完成后当接收数据从接收移位寄存器转移至接收数据寄存器时中断产生。
 - 发送使能中断（INTUCnT）：
 - 发送使能状态下，当数据从发送数据寄存器转移至发送移位寄存器时中断产生。
- 字符长度：7 至 9 位
- 奇偶检验功能：奇检验、偶校验、0 校验、无校验。
- 发送停止位：1，2 位
- 片上专用波特率发生器。
- 可选择 MSB-/LSB-传输。
- 发送/接收数据可反转输入/输出。
- 在 LIN（本地互联网）通信格式下 SBF（同步中断域）发送。
 - SBF 发送时 13 至 20 位可选。
 - SBF 接收时 11 位或更高的分辨率。
 - 提供 SBF 接收标志。

备注 n = 0 至 4

17.2 配置

UARTCn 的功能框图如下所示：

图 17-1. 异步串行接口 Cn 功能框图



UARTCn 包括以下硬件：

表 17-1. UARTCn 的配置

项目	配置
寄存器	UARTCn 控制寄存器 0（UCnCTL0） UARTCn 控制寄存器 1（UCnCTL1） UARTCn 控制寄存器 2（UCnCTL2） UARTCn 选项控制寄存器 0（UCnOPT0） UARTCn 选项控制寄存器 1（UCnOPT1） UARTCn 状态寄存器（UCnSTR） UARTCn 接收移位寄存器 UARTCn 接收数据寄存器（UCnRX） UARTCn 发送移位寄存器 UARTCn 接收数据寄存器（UCnTX）

(1) UARTCn 控制寄存器 0 (UCnCTL0)

UCnCTL0 寄存器为 8 位寄存器，用来指定 UARTCn 的操作。

(2) UARTCn 控制寄存器 1 (UCnCTL1)

UCnCTL1 寄存器为 8 位寄存器，用于选择 UARTCn 的输入时钟。

(3) UARTCn 控制寄存器 2 (UCnCTL2)

UCnCTL2 寄存器为 8 位寄存器，用来控制 UARTCn 的波特率。

(4) UARTCn 选项控制寄存器 0 (UCnOPT0)

UCnOPT0 寄存器为 8 位寄存器，用来控制 UARTCn 的串行传输。

(5) UARTCn 选项控制寄存器 1 (UCnOPT1)

UCnOPT1 寄存器为 8 位寄存器，用来控制 UARTAn 的 9 位长度的串行传输。

(6) UARTCn 状态寄存器 (UCnSTR)

UCnSTRn 寄存器由发生接收错误时错误内容的标志组成。当发生一个接收错误时，接收错误标志之一被置 1。

(7) UARTCn 接收移位寄存器

本寄存器为移位寄存器，用来将输入至 RXDCn 引脚的串行数据转换为并行数据。单字节接收并检测到停止位时，则接收数据传输至 UCnRX 寄存器。

本寄存器不能直接操作。

(8) UARTCn 接收数据寄存器 (UCnRX)

UCnRX 为 8 位寄存器，用来保持接收数据。当接收 7 个字符时，最高有效位置 0（当数据以 LSB 接收）。

在接收使能状态下，与 1 帧移入操作完毕同步，接收数据从 UARTCn 接收移位寄存器传输至 UCnRX 寄存器。

发送至 UCnRX 寄存器同样会输出接收完成中断请求信号（INTUCnR）。

(9) UARTCn 发送移位寄存器

该移位寄存器用于将来自 UCnTX 寄存器的并行数据转换为串行数据。

当一个字节的数据从 UCnTX 寄存器传送过来时，移位寄存器数据从 TXDCn 引脚输出。

本寄存器不能直接操作。

(10) UARTCn 接收数据寄存器 (UCnTX)

UCnTX 寄存器为 8 位发送数据缓存器。当数据写入 UCnTX 寄存器后启动发送。当数据可以被写入 UCnTX 寄存器时（当一帧数据从 UCnTX 寄存器传输至 UARTCn 发送移位寄存器时），产生发送允许中断请求信号（INTUCnT）。

17.3 UARTC和其它串行接口模式转换

17.3.1 UARTC0 和CSIF4 间的模式转换

V850ES/JG3-H 和 V850ES/JH3-H 中, CSIF4 和 UARTC0 共享相同引脚, 因而不能同时使用。使用前, 用 PMC3 和 PFC3 寄存器预先设置 UARTC0。

注意事项 如果发送或接收期间转换这些功能, **CSIF4** 和 **UARTC0** 的发送/接收操作得不到保证。确保禁止未使用的模式。

图 17-2. CSIF4 和 UARTC0 模式转换设置

复位后: 00H R/W 地址: FFFFF446H

	7	6	5	4	3	2	1	0
PMC3	PMC37	PMC36	PMC35	PMC34	PMC33	PMC32	PMC31	PMC30

复位后: 00H R/W 地址: FFFFF466H

	7	6	5	4	3	2	1	0
PFC3	PFC37	PFC36	PFC35	PFC34	PFC33	PFC32	PFC31	PFC30

复位后: 00H R/W 地址: FFFFF706H

	7	6	5	4	3	2	1	0
PFCE3	PFCE37	PFCE36	PFCE35	PFCE34	PFCE33	PFCE32	PFCE31	PFCE30

PMC32	PFCE32	PFC32	操作模式
0	X	X	端口 I/O 模式
1	0	0	ASCKC0 (UARTC0)
1	0	1	SCKF4 (CSIF4)

PMC31	PFCE31	PFC31	操作模式
0	X	X	端口 I/O 模式
1	0	0	RXDC0 (UARTC)
1	0	1	SIF4 (CSIF4)

PMC30	PFCE30	PFC30	操作模式
0	X	X	端口 I/O 模式
1	0	0	TXDC0 (UARTC)
1	0	1	SOF4 (CSIF4)

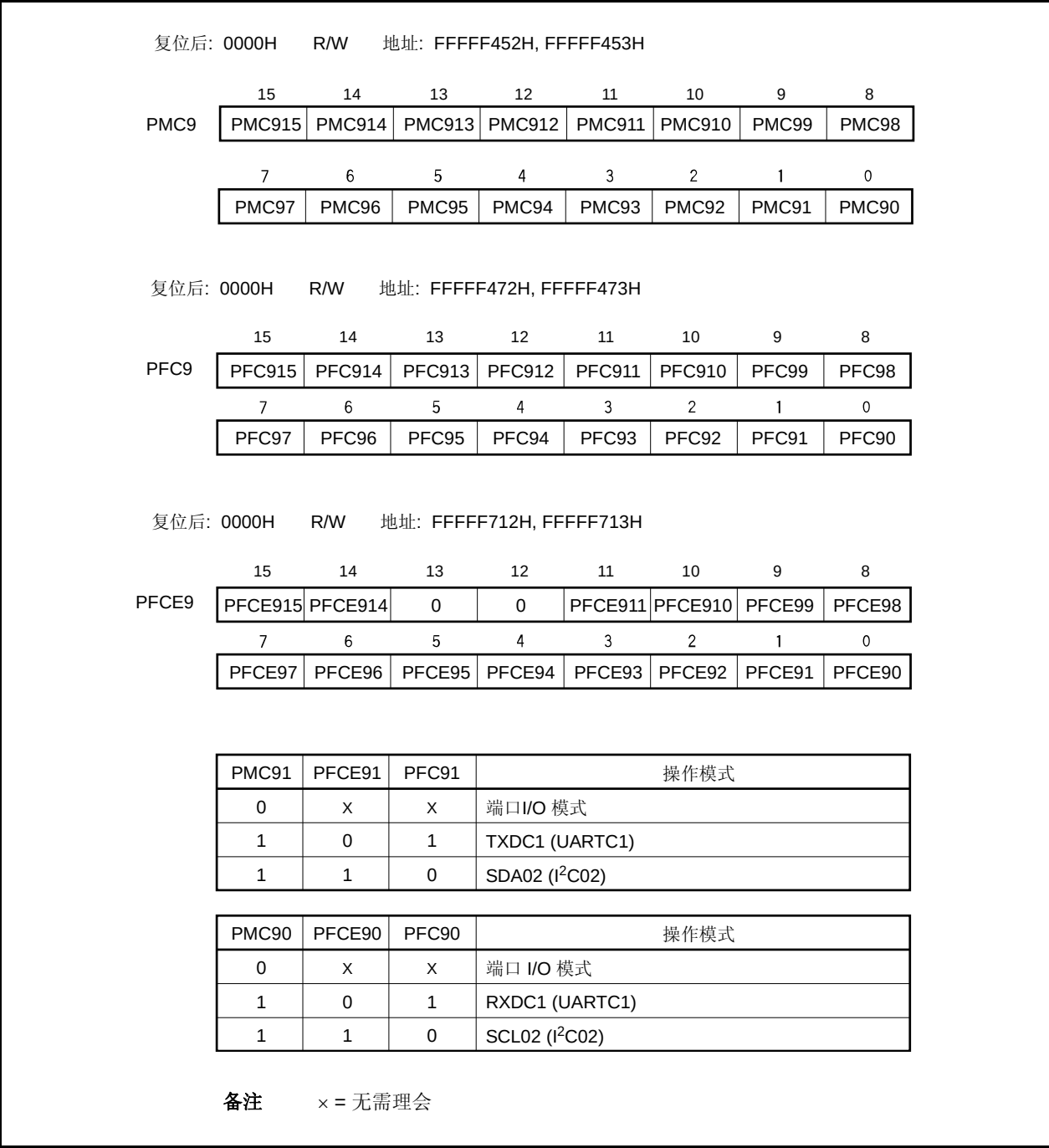
备注 x = 无需理会

17.3.2 UARTC1 和 I²C02 间的模式转换

V850ES/JG3-H 和 V850ES/JH3-H 中, UARTC1 和 I²C02 共享相同引脚, 因而不能同时使用。使用前, 用 PMC9, PFC9 和 PFCE9 寄存器预先设置 UARTC1。

注意事项 如果发送或接收期间转换这些功能, UARTC1 和 I²C02 的发送/接收操作得不到保证。确保禁止未使用的模式。

图 17-3. UARTC1 和 I²C02 模式转换设置



17.3.3 UARTC2 和 CSIF3 间的模式转换

V850ES/JG3-H 和 V850ES/JH3-H 中, UARTC2 和 CSIF3 共享相同引脚, 因而不能同时使用。使用前, 用 PMC9, PFC9 和 PFCE9 寄存器预先设置 UARTC2。

注意事项 如果发送或接收期间转换这些功能, UARTC2 和 CSIF3 的发送/接收操作得不到保证。确保禁止未使用的模式。

图 17-4. UARTC2 和 CSIF3 模式转换设置

复位后: 0000H R/W 地址: FFFFF452H, FFFFF453H

	15	14	13	12	11	10	9	8
PMC9	PMC915	PMC914	PMC913	PMC912	PMC911	PMC910	PMC99	PMC98
	7	6	5	4	3	2	1	0
	PMC97	PMC96	PMC95	PMC94	PMC93	PMC92	PMC91	PMC90

复位后: 0000H R/W 地址: FFFFF472H, FFFFF473H

	15	14	13	12	11	10	9	8
PFC9	PFC915	PFC914	PFC913	PFC912	PFC911	PFC910	PFC99	PFC98
	7	6	5	4	3	2	1	0
	PFC97	PFC96	PFC95	PFC94	PFC93	PFC92	PFC91	PFC90

复位后: 0000H R/W 地址: FFFFF712H, FFFFF713H

	15	14	13	12	11	10	9	8
PFCE9	PFCE915	PFCE914	0	0	PFCE911	PFCE910	PFCE99	PFCE98
	7	6	5	4	3	2	1	0
	PFCE97	PFCE96	PFCE95	PFCE94	PFCE93	PFCE92	PFCE91	PFCE90

PMC91	PFCE91	PFC91	操作模式
0	x	x	端口 I/O 模式
1	0	1	TXDC1 (UARTC1)
1	1	0	SDA02 (I ² C02)

PMC90	PFCE90	PFC90	操作模式
0	x	x	端口 I/O 模式
1	0	1	RXDC1 (UARTC1)
1	1	0	SCL02 (I ² C02)

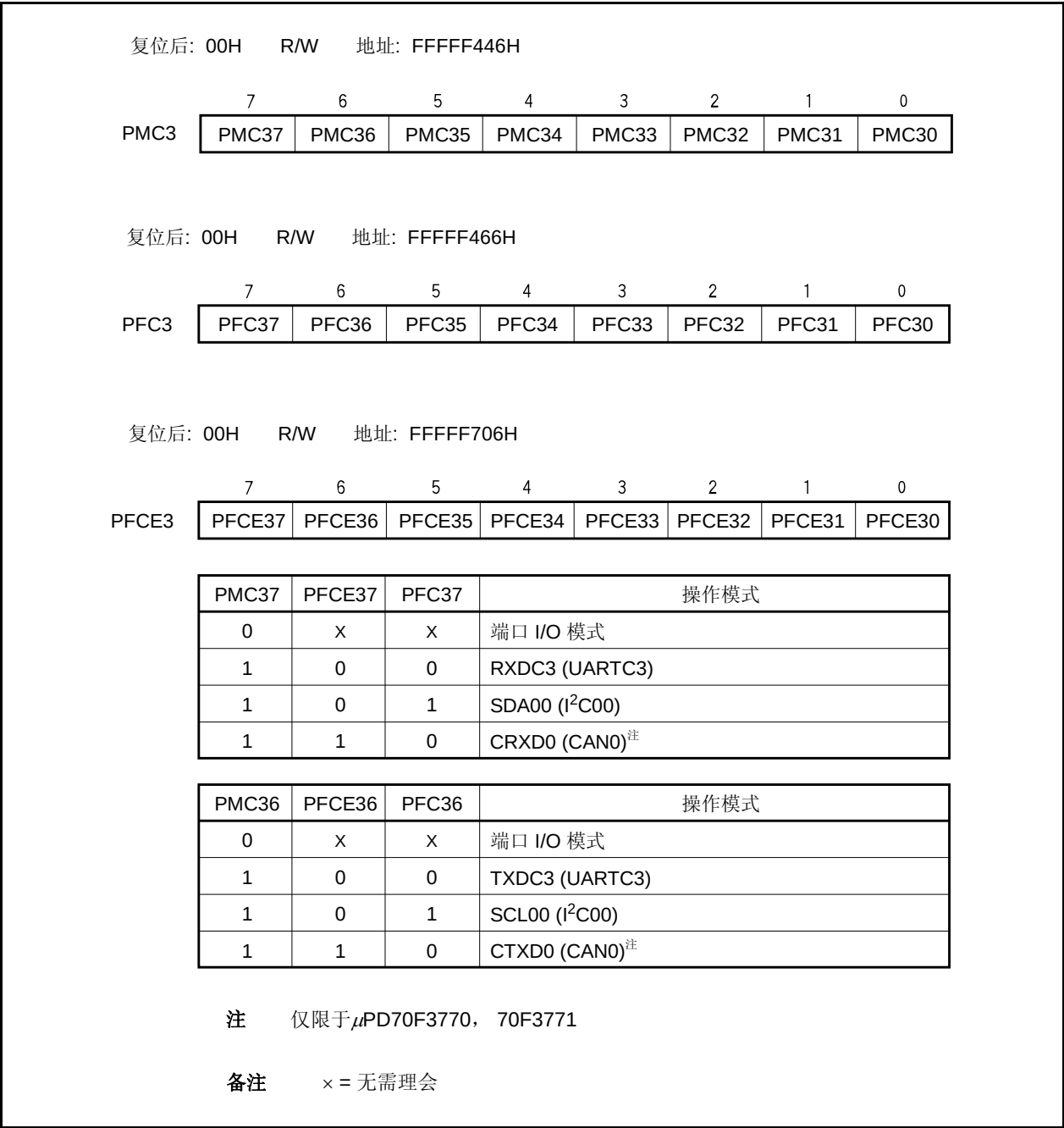
备注 x = 无需理会

17.3.4 UARTC3, I²C00 和 CAN0 间的模式转换

V850ES/JG3-H 和 V850ES/JH3-H 中, UARTC2, I²C00 和 CAN0 (仅限于μPD70F3770, 70F3771) 共享相同引脚, 因而不能同时使用。使用前, 用 PMC3, PFC3 和 PFCE3 寄存器预先设置 UARTC3。

注意事项 如果发送或接收期间转换这些功能, UARTC3, I²C00 和 CAN0 (仅限于μPD70F3770, 70F3771) 的发送/接收操作得不到保证。确保禁止未使用的模式。

图 17-5. UARTC3, I²C00 和 CAN0 模式转换设置



17.3.5 UARTC4, CSIF0 和 I²C01 间模式的转换

V850ES/JG3-H 和 V850ES/JH3-H 中, UARTC4, CSIF0 和 I²C01 共享相同引脚, 因而不能同时使用。使用前, 用 PMC4, PFC4 和 PMCE4 寄存器预先设置 UARTC4。

注意事项 如果发送或接收期间转换这些功能, UARTC4, CSIF0 和 I²C01 的发送/接收操作得不到保证。确保禁止未使用的模式。

图 17-6. UARTC4, CSIF0 和 I²C01 模式转换设置

复位后: 00H R/W 地址: FFFFF448H

	7	6	5	4	3	2	1	0
PMC4	0	0	0	0	0	PMC42	PMC41	PMC40

复位后: 00H R/W 地址: FFFFF468H

	7	6	5	4	3	2	1	0
PFC4	0	0	0	0	0	PFC42	PFC41	PFC40

复位后: 00H R/W 地址: FFFFF708H

	7	6	5	4	3	2	1	0
PFCE4	0	0	0	0	0	0	PFCE41	PFCE40

PMC41	PFCE41	PFC41	操作模式
0	x	x	端口 I/O 模式
1	0	0	SOF0 (CSIF0)
1	0	1	RXDC4 (UARTC4)
1	1	0	SCL01 (I ² C01)

PMC40	PFCE40	PFC40	操作模式
0	x	x	端口 I/O 模式
1	0	0	SIF0 (CSIF0)
1	0	1	TXDC4 (UARTC4)
1	1	0	SDA01 (I ² C01)

备注 x = 无需理会

17.4 寄存器

(1) UARTCn 控制寄存器 0 (UCnCTL0)

UCnCTL0 寄存器为 8 位寄存器，用来控制 UARTCn 的串行传输操作。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器设置为 10H。

(1/2)

复位后: 10H R/W 地址: UC0CTL0 FFFFFFFA00H, UC1CTL0 FFFFFFFA10H,
UC2CTL0 FFFFFFFA20H, UC3CTL0 FFFFFFFA30H,
UC4CTL0 FFFFFFFA40H

	<7>	<6>	<5>	<4>	3	2	1	0
UCnCTL0	UCnPWR	UCnTXE	UCnRXE	UCnDIR	UCnPS1	UCnPS0	UCnCL	UCnSL

(n = 0 至 4)

UCnPWR	UCRTCn 操作控制
0	禁止 UCRTCn 操作 (UCRTCn 异步复位)
1	允许 UCRTCn 操作
UARTCn 操作由 UCnPWR 位控制。通过将UCnPWR位清除为0，TXDCn 引脚输出固定为高电平 (如果UCnOPT0.UCnTDL 位 = 1则固定为低电平)。	

UCnTXE	发送操作使能
0	禁止发送操作
1	允许发送 操作
· 若要开始发送, 将 UCnPWR位设置为 1而后再设置 UCnTXE位为1。 · 若要停止发送, 将UCnTXE 位清除为 0而后再设置 UCnPWR 位为 0。 · 若要初始化 发送 单元, 将 UCnTXE 位清除为 0, 等待基础时钟的两个周期过后, 而后再次将 UCnTXE 位设置为 1。否则, 可能不执行初始化 (关于基础时钟, 参见 17.7 (1) (a) 基础时钟)。	

UCnRXE	接收操作使能
0	禁止接收 操作
1	允许接收 操作
· 若要开始接收, 将UCnPWR位设置为1 而后再设置 UCnRXE 位为 1。 · 若要停止接收, 将 UCnRXE 位清除为 0而后再设置 UCnPWR 位为 0。 · 若要初始化接收单元, 将 UCnRXE位清除为 0, 等待基础时钟的两个周期过后, 而后再次将 UCnRXE 位设置为1。否则, 可能不执行初始化 (关于基础时钟, 参见 17.7 (1) (a) 基础时钟)。	

UCnDIR	传输方向选择
0	MSB在先传输
1	LSB在先传输
- 仅当 UCnPWR 位 = 0 或 UCnTXE 位 = UCnRXE 位 = 0 时可重写该寄存器。 - 当发送和接收以 LIN 格式进行时，将 UCnDIR 位设置为 1。	

UCnPS1	UCnPS0	发送期间奇偶选择	接收期间奇偶选择
0	0	无奇偶输出	无奇偶接收
0	1	0 奇偶输出	0 奇偶接收
1	0	奇数奇偶输出	奇数奇偶校验
1	1	偶数奇偶输出	偶数奇偶校验
- 仅当 UCnPWR 位 = 0 或 UCnTXE 位 = UCnRXE 位 = 0 时可重写该寄存器。 - 如果接收期间选择“0 奇偶接收”，则不执行奇偶检查。因此，不需要设置 UCnSTR. UCnPE 位。 - 当发送和接收以 LIN 格式进行时，将 UCnPS1 和 UCnPS0 位清除为 00。			

UCnCL	发送/接收数据的 1 帧数据字符长度的说明
0	7 位
1	8 位
- 仅当 UCnPWR 位 = 0 或 UCnTXE 位 = UCnRXE 位 = 0 时可重写该寄存器。 - 当发送和接收以 LIN 格式进行时，将 UCnCL 位设置为 1。	

UCnSL	发送数据停止位长度的说明
0	1 位
1	2 位
仅当 UCnPWR 位 = 0 或 UCnTXE 位 = UCnRXE 位 = 0 时可重写该寄存器。	

备注 关于奇偶校验的详情，参见 17.6.9 奇偶类型和操作。

(2) UARTCn 控制寄存器 1 (UCnCTL1)

关于细节，参见 17.7 (2) UARTCn 控制寄存器 1 (UCnCTL1)。

(3) UARTCn 控制寄存器 2 (UCnCTL2)

关于细节，参见 17.7 (3) UARTCn 控制寄存器 2 (UCnCTL2)。

(4) UARTCn 选项控制寄存器 0（UCnOPT0）

UCnOPT0 寄存器为 8 位寄存器，用于控制 UARTCn 寄存器的串行传输操作。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器设置为 14H。

(1/2)

复位后: 14H R/W 地址: UC0OPT0 FFFFFFFA03H, UC1OPT0 FFFFFFFA13H,
UC2OPT0 FFFFFFFA23H, UC3OPT0 FFFFFFFA33H,
UC4OPT0 FFFFFFFA43H

	<7>	6	5	4	3	2	1	0
UCnOPT0	UCnSRF	UCnSRT	UCnSTT	UCnSLS2	UCnSLS1	UCnSLS0	UCnTDL	UCnRDL
(n = 0 至 4)								

UCnSRF	SBF 接收标志
0	当设置 UCnCTL0.UCnPWR 位 = UCnCTL0.UCnRXE 位 = 0 或 SBF 接收正常结束时。
1	SBF 接收期间
• LIN 通信期间判断 SBF（同步中断域）。 • 发生 SBF 接收错误时 UCnSRF 位保持为 1，然后 SBF 接收重新开始。 • UCnSRF 位为只读位。	

UCnSRT	SBF 接收触发
0	-
1	SBF 接收触发
• 该位为 LIN 通信期间 SBF 的接收触发位，读出值总为 “0”。 • 若要进行 SBF 接收，需设置 UCnSRT 位 (为 1) 来使能 SBF 接收。 • 设置 UCnPWR 位 = UCnRXE 位 = 1 之后，设置 UCnSRT 位。	

UCnSTT	SBF 发送触发
0	-
1	SBF 发送触发
• 该位为 LIN 通信期间 SBF 的发送触发位，读出值总为 “0”。 • 设置 UCnPWR 位 = UCnTXE 位 = 1 之后，设置 UCnSTT 位。	

注意事项 SBF 接收（UCnSRF 位 = 1）期间，不要设置 UCnSRT 和 UCnSTT 位（为 1）。

UCnSLS2	UCnSLS1	UCnSLS0	SBF 发送长度选择
1	0	1	13位输出 (复位值)
1	1	0	14位输出
1	1	1	15位输出
0	0	0	16位输出
0	0	1	17位输出
0	1	0	18位输出
0	1	1	19位输出
1	0	0	20位输出

当 UCnPWR 位 = 0 或当 UCnTXE 位 = 0时可设置该寄存器。

UCnTDL	发送数据电平位
0	传送数据的正常输出
1	传送数据的反向输出

- TXDCn引脚的输出电平可用 UCnTDL 位反向。
- 当 UCnPWR 位 = 0 或当 UCnTXE 位 = 0时可设置该寄存器。

UCnRDL	接收数据电平位
0	传送数据的正常输出
1	传送数据的反向输出

- RXDCn 引脚的输出电平可用 UCnRDL 位反向。
- 当 UCnPWR 位 = 0或当 UCnRXE 位 = 0时可设置该寄存器。

(5) UARTCn 选项控制寄存器 1 (UCnOPT1)

UCnOPT1 寄存器为 8 位寄存器，用来控制 UARTCn 的串行传输操作。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

注意事项 禁止 UARTC 操作时 (UCnCTL0.UCnPWR = 0) 设置 UCnEBE 位。

复位后: 00H R/W 地址: UC0OPT1 FFFFFFFA0AH, UC1OPT1 FFFFFFFA1AH,
UC2OPT1 FFFFFFFA2AH, UC3OPT1 FFFFFFFA3AH,
UC4OPT1 FFFFFFFA4AH

	7	6	5	4	3	2	1	0	
UCnOPT1	0	0	0	0	0	0	0	0	UCnEBE

(n = 0 至 4)

UCnEBE	扩展位允许/禁止
0	禁止扩展位操作。按UCnCTL0.UCnCL位设置的数据长度进行发送/接收。
1	允许扩展位操作。发送/接收可按9位字符长度执行。

- 当设置 UCnEBE 为 1, 按9位数据长度发送时，确保进行以下设置。
如果不进行该设置，则 UCnEBE 位设置无效。
- UCnCTL0.UCnPS1, UCnPS0 = 00 (无奇偶位)
- CnCTL0.UCnCL = 1 (8位字符长度)
- 如果发送或接收按 LIN 通信格式进行，将UCnEBE 设置为 0。

以下显示了寄存器设置值与数据格式之间的关系。

表 17-2. 寄存器设置与数据格式之间的关系

寄存器设置					数据格式				
UCnCTL0				UCnOPT1	D0 至 D6	D7	D8	D9	D10
UCnCL	UCnPS1	UCnPS0	UCnSL	UCnEBE					
0	0	0	0	0	数据	停止	—	—	—
0	非 00				数据	奇偶	停止	—	—
1	0	0			数据	数据	停止	—	—
1	非 00				数据	数据	奇偶	停止	—
0	0	0	1	0	数据	停止	停止	—	—
0	非 00				数据	奇偶	停止	停止	—
1	0	0			数据	数据	停止	停止	—
1	非 00				数据	数据	奇偶	停止	停止
0	0	0	0	1	数据	停止	—	—	—
0	非 00				数据	奇偶	停止	—	—
1	0	0			数据	数据	数据	停止	—
1	非 00				数据	数据	奇偶	停止	—
0	0	0	1	1	数据	停止	停止	—	—
0	非 00				数据	奇偶	停止	停止	—
1	0	0			数据	数据	数据	停止	停止
1	非 00				数据	数据	奇偶	停止	停止

备注 数据：数据位
 停止：停止位
 奇偶：奇偶位

(6) UARTCn 状态寄存器（UCnSTR）

UCnSTR 寄存器为 8 位寄存器，显示 UARTCn 的传输状态和接收错误的内容。

该寄存器可按 8 位或 1 位进行读写，但 UCnTSF 位为只读位，而 UCnPE， UCnFE 和 UCnOVE 位可读可写。这些位只能写入 0 清除，它们不能通过写 1 来置位（即便写入 1，其值仍保持不变）。

初始状态显示如下：

寄存器/位	初始状态
UCnSTR 寄存器	• 复位 • UCnCTL0.UCnPWR = 0
UCnTSF 位	• UCnCTL0.UCnTXE = 0
UCnPE， UCnFE， UCnOVE 位	• 写入 0 • UCnCTL0.UCnRXE = 0

复位后: 00H R/W 地址: UC0STR FFFFFFFA04H, UC1STR FFFFFFFA14H,
UC2STR FFFFFFFA24H, UC3STR FFFFFFFA34H,
UC4STR FFFFFFFA44H

	<7>	6	5	4	3	<2>	<1>	<0>
UCnSTR (n = 0 至 4)	UCnTSF	0	0	0	0	UCnPE	UCnFE	UCnOVE

UCnTSF	传送状态标志
0	- 当已经设置 UCnPWR 位 = 0 或 UCnTXE 位 = 0 时。 - 当发送完成后 UCnTX 寄存器没有要发送的下一个数据时
1	写入 UCnTX 寄存器
执行连续发送时 UCnTSF 位总为 1。当初始化发送单元时，在执行初始化前检查 UCnTSF 位 = 0。当 UCnTSF 位 = 1 时执行初始化，发送数据得不到保证。	

UCnPE	奇偶错误标志
0	- 当已经设置 UCnPWR 位 = 0 或 UCnRXE 位 = 0 时。 - 当已经写入 0 时。
1	当接收期间数据的奇偶与奇偶位不匹配时。
- UCnPE 位的操作由 UCnCTL0.UCnPS1 和 UCnCTL0.UCnPS0 位的设置控制。 - UCnPE 位可读可写，但只能通过向其写入 0 而将其清零，并且向其写入 1 不能将其置位。当向该位写入 1 时，保留该值。	

UCnFE	帧错误标志
0	- 当已经设置 UCnPWR 位 = 0 或 UCnRXE 位 = 0 时。 - 当已经写入 0 时。
1	当接收期间检测不到停止位时。
- 只检查接收数据停止位的第一位，而不考虑 UCnCTL0.UCnSL 位的值。 - UCnFE 位可读可写，但只能通过向其写入 0 而将其清零，并且向其写入 1 不能将其置位。当向该位写入 1 时，保留该值。	

UCnOVE	超限错误标志
0	- 当已经设置 UCnPWR 位 = 0 或 UCnRXE 位 = 0 时。 - 当已经写入 0 时。
1	当接收数据已经设置到 UCnRX 寄存器而在接收数据被读取之前下一个接收操作已经完成时。
- 当发生超限错误时，如果没有下一个接收数据写入接收缓存，该数据被丢弃。 - UCnOVE 位可读可写，但只能通过向其写入 0 而将其清零，并且向其写入 1 不能将其置位。当向该位写入 1 时，保留该值。	

(7) UARTCn 接收数据寄存器 L (UCnRXL) 和 UARTCn 接收数据寄存器 (UCnRX)

UCnRXL 和 UCnRX 寄存器是一个 8 位或 9 位缓冲寄存器，用于存储由接收移位寄存器转换而来的并行数据。

1 字节数据接收完成时，存储于接收移位寄存器的数据传输至 UCnRXL 和 UCnRX 寄存器。

当数据长度指定为 7 位，LSB 接收当中，接收数据传输至 UCnRXL 寄存器的位 6 至位 0，而 MSB 总是为 0。

MSB 接收当中，接收数据传输至 UCnRXL 寄存器的位 7 至位 1，而 LSB 总是为 0。

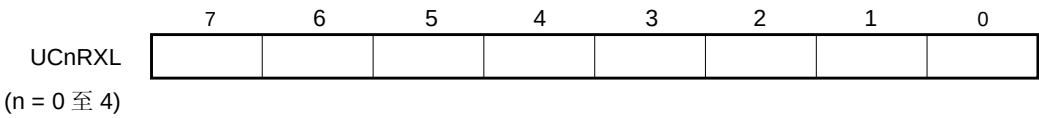
发生超限 (UCnOVE) 错误时，此时的接收数据不能传输至 UCnRXL 寄存器并被丢弃。

访问单位或复位值根据字符长度不同而有所区别。

- 字符长度 7/8 位 (UCnOPT1.UCnEBE = 0)
本寄存器为只读寄存器，可以按 8 位读取。
复位或 UCnCTL0.UCnPWR 位 = 0 将该寄存器设置为 FFH。
- 字符长度 9 位 (UCnOPT1.UCnEBE = 0)
本寄存器为只读寄存器，可以按 16 位读取。
复位或 UCnCTL0.UCnPWR 位 = 0 将该寄存器设置为 01FFH。

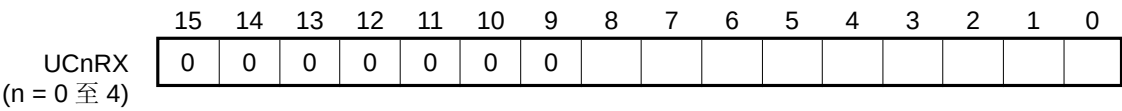
(a) 字符长度 7/8 位 (UCnOPT1.UCnEBE = 0)

复位后: FFH R 地址: UC0RXL FFFFFFFA06H, UC1RXL FFFFFFFA16H,
 UC2RXL FFFFFFFA26H, UC3RXL FFFFFFFA36H,
 UC4RXL FFFFFFFA46H



(b) 字符长度 9 位 (UCnOPT1.UCnEBE = 1)

复位后: 01FFH R 地址: UC0RX FFFFFFFA06H, UC1RX FFFFFFFA16H,
 UC2RX FFFFFFFA26H, UC3RX FFFFFFFA36H,
 UC4RX FFFFFFFA46H



(8) UARTCn 发送数据寄存器 L (UCnTXL) 和 UARTCn 发送数据寄存器 (UCnTX)

UCnTXL 和 UCnTX 寄存器为 8 位寄存器，用来设置发送数据。

当数据长度指定为 7 位，LSB 发送当中，发送数据传输至 UCnRX 寄存器的位 6 至位 0。MSB 发送当中，接收数据传输至 UCnRX 寄存器的位 7 至位 1。

访问单位或复位值根据字符长度不同而有所区别。

- 字符长度 7/8 位 (UCnOPT1.UCnEBE = 0)

该寄存器可按 8 位为单位进行读写。

复位将此寄存器 设置为 FFH。

- 字符长度 9 位 (UCnOPT1.UCnEBE = 0)

该寄存器可按 16 位为单位进行读写。

复位后本寄存器被设置为 01FFH。

注意事项 1. 在发送操作使能状态 (UCnPWR = 1 和 UCnTXE = 1) 下，写入 UCnTXL，UCnTX 寄存器作为发送开始的触发操作，如果写入最近值并将其存储，则在 INTUCnT 中断发生前，相同数据分两次传送。

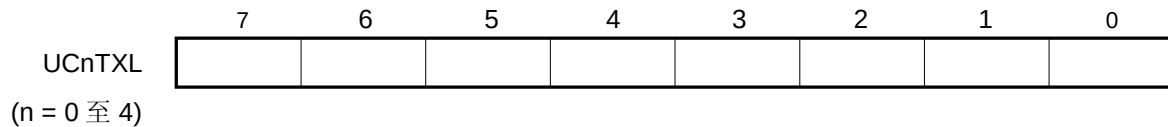
2. 产生 INTUCnT 中断后，为连续发送写入数据。

如果在 INTUCnT 中断发生前写入下一个数据，发送开始处理与写入 UCnTXL，UCnTX 寄存器发生冲突，则可能发生未预料的操作。

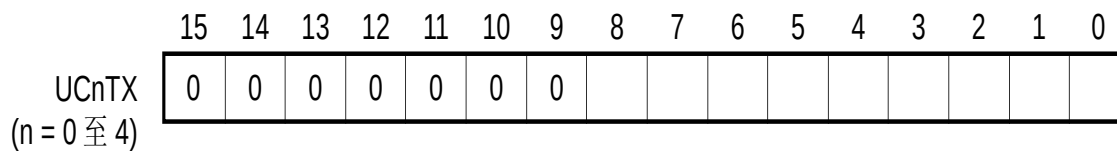
3. 如果在禁止发送操作的寄存器中执行写入 UCnTXL，UCnTXL，则其不能用作发送开始触发。因此，在禁止发送操作状态中执行写入 UCnTXL，UCnTX 寄存器后即使处于发送使能状态，也不能开始发送。

(a) 字符长度 7/8 位 (UCnOPT1.UCnEBE = 0)

复位后: FFH R/W 地址: UC0TXL FFFFFFFA08H, UC1TXL FFFFFFFA18H,
UC2TXL FFFFFFFA28H, UC3TXL FFFFFFFA38H,
UC4TXL FFFFFFFA48H

**(b) 字符长度 9 位 (UCnOPT1.UCnEBE = 1)**

复位后: 01FFH R/W 地址: UC0TX FFFFFFFA08H, UC1TX FFFFFFFA18H,
UC2TX FFFFFFFA28H, UC3TX FFFFFFFA38H,
UC4TX FFFFFFFA48H



17.5 中断请求信号

以下两类中断请求信号产生于 UARTCn:

- 接收完成中断请求信号（INTUCnR）
- 发送允许中断请求信号（INTUCnT）

这两类中断请求信号的默认优先级为接收完成中断请求信号优先，然后为发送使能中断请求信号。

表 17-3. 中断与其默认优先级

中断	优先级
接收完成	高
发送允许	低

(1) 接收完成中断请求信号（INTUCnR）

接收使能状态下，数据转移至接收移位寄存器并发送至 UCnRX 寄存器时，接收完成中断请求信号输出。
发生接收错误时，接收完成中断请求信号也同样输出。因此，响应接收完成中断请求信号并读取数据时，读取 UCnSTR 寄存器并检查接收结果是否错误。
在接收禁止状态没有接收完成中断请求信号产生。

(2) 发送允许中断请求信号（INTUCnT）

在发送允许状态下，当发送数据从 UCnTX 寄存器传输至 UARTCn 发送移位寄存器，产生发送允许中断请求信号。

17.6 操作

17.6.1 数据格式

执行全双工串行数据接收和发送。

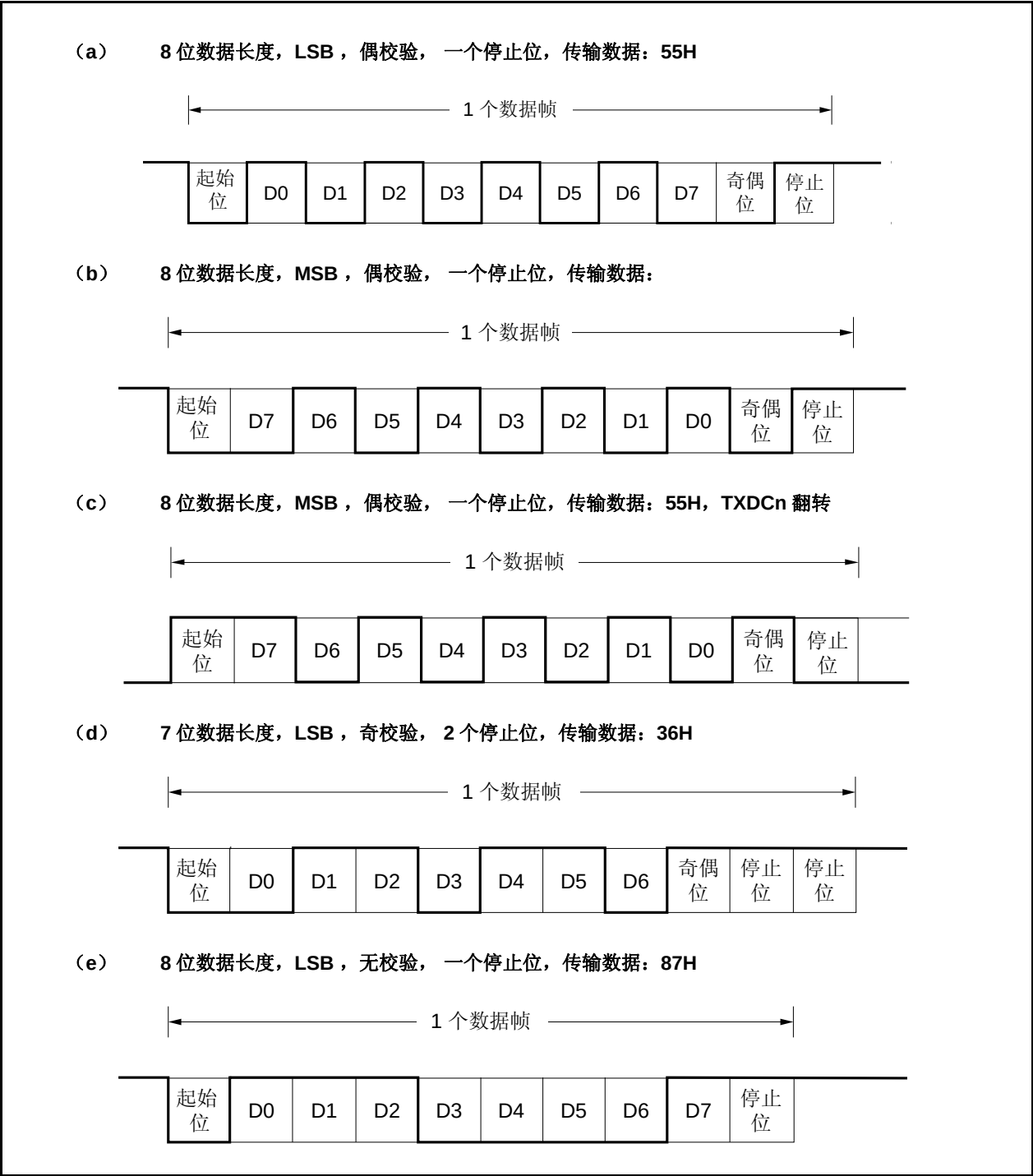
图 17-7 所示，一个发送/接收数据帧的数据由起始位、字符位、校验位以及停止位组成。

数据帧内字符位的说明、奇偶位、停止位长度以及 MSB/LSB 传输的选择均由 UCnCTL0 寄存器来指定。

另外，UART 中关于 TXDCn 位的输出/转换控制由 UCnOPT0.UCnTDL 位来执行。

- 起始位.....1 位
- 字符位.....7 位/8 位
- 奇偶位.....奇检验/偶校验/0 校验/无校验
- 停止位.....1 位/2 位

图 17-7. UARTA 发送/接收数据格式



17.6.2 SBF发送/接收格式

V850ES/JG3-H 和 V850ES/JH3-H 具有 SBF（同步中断域）发送/接收控制功能以允许使用 LIN 功能。

备注 LIN 表示局部互联网，是一种低速（1 至 20 kbps）串行通信协议，旨在于降低汽车网络的成本。LIN 通信为单主机通信，一台主机最多可连接 15 台从机。LIN 从机用于控制开关、传动装置和传感器，这些装置可通过 LIN 网络连接到 LIN 主机上。通常，LIN 主机接入诸如 CAN（控制区域网）之类的网络。另外，LIN 总线使用单线方式且由遵循 ISO9141 规范的收发器连接至各个节点。LIN 协议中，主机以某一波特率信息发送一个数据帧，从机将其接收并校正波特率错误。因此，从机中波特率错误在 $\pm 15\%$ 或更少时，通信可正常进行。

图 17-8 和 17-9 表明 LIN 的发送和接收的处理。

图 17-8. LIN 发送处理概要

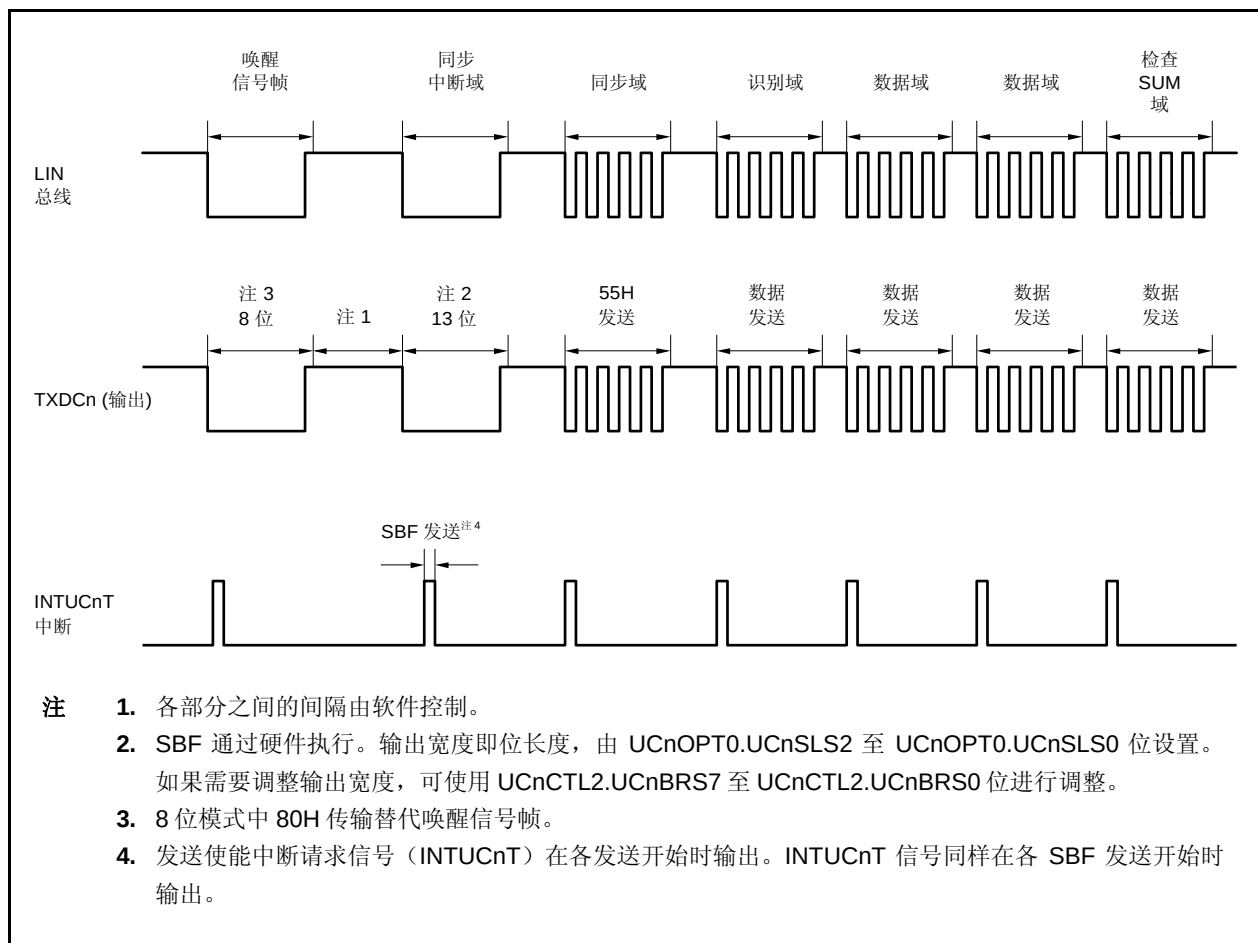
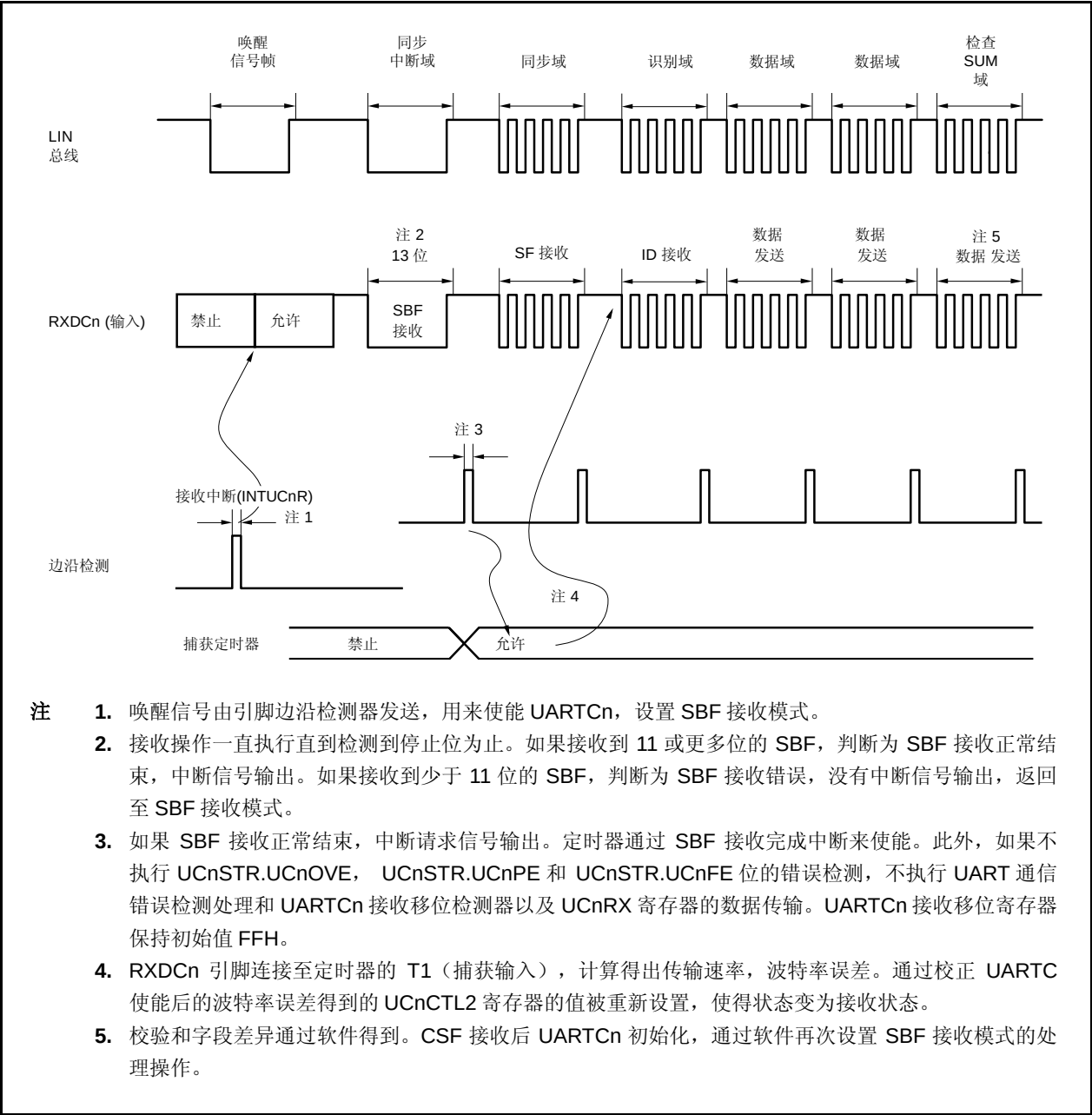


图 17-9. LIN 接收处理概要



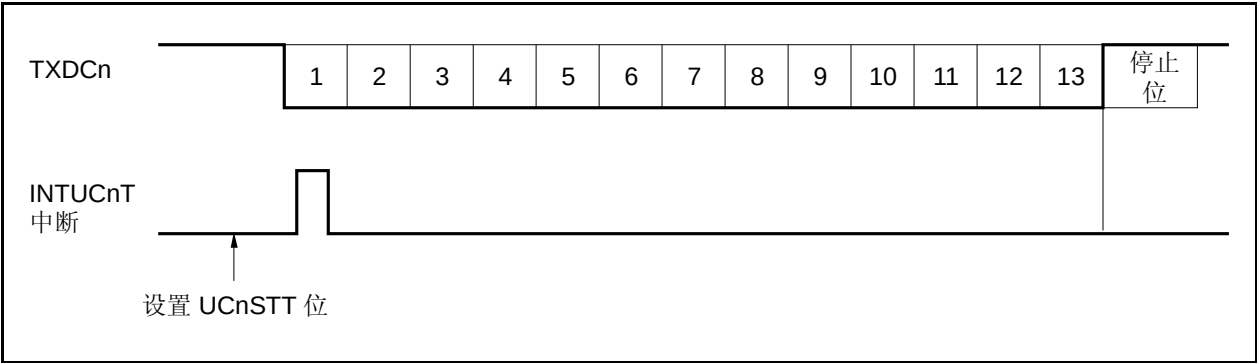
17.6.3 SBF发送

UCnCTL0.UCnPWR 位 = UCnCTL0.UCnTXE 位 = 1 时，进入发送使能状态，通过设置 SBF 发送触发 (UCnOPT0.UCnSTT 位) 为 1 开始 SBF 发送。

之后，由 UCnOPT0.UCnSLS2 至 UCnOPT0.UCnSLS0 位指定的位 13 至 20 宽度的低电平输出。SBF 发送开始时，发送使能中断请求信号 (INTUCnT) 产生。接下来结束 SBF 发送，UCnSTT 位自动清除。在这之后，恢复 UART 发送模式。

发送被暂停直到需要发送的下一个数据被写入 UCnTX 寄存器，或者设置 SBF 发送触发 (UCnSTT 位)。

图 17-10. SBF 发送

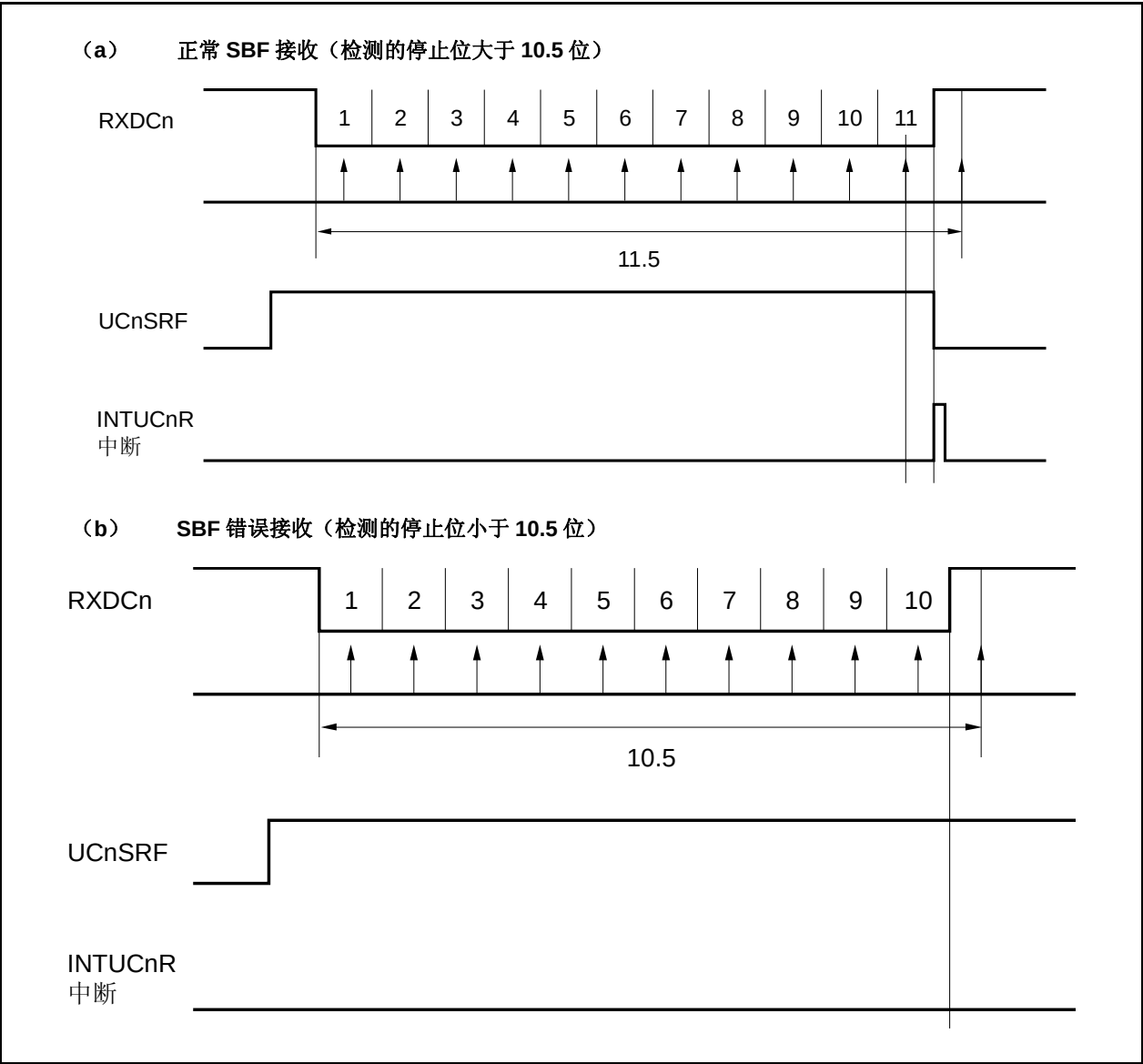


17.6.4 SBF接收

通过设置 UCnCTL0.UCnPWR 位为 1 并随之设置 UCnCTL0.UCnRXE 位为 1 来进入接收等待状态。
通过设置 SBF 接收触发 (UCnOPT0.UCnSRT 位) 为 1 来设置 SBF 接收等待状态。
SBF 接收等待状态下, 类似于 UART 接收等待状态, RXDCn 引脚被监控并且进行起始位的检测。
起始位检测完成后, 根据设置的波特率开始接收并且内部计数器开始向上计数。
接收到一个停止位时, 如果 SBF 宽度为 11 位或更多位, 则判断为正常处理, 接收完成中断请求信号 (INTUCnR) 输出。UCnOPT0.UCnSRF 位自动清除并且 SBF 接收结束。如果不执行 UCnSTR.UCnOVE, UCnSTR.UCnPE 和 UCnSTR.UCnFE 位的错误检测, 不执行 UART 通信错误检测处理。另外, 不执行 UARTCn 接收移位寄存器和 UCnRX 寄存器的数据传输并且保持初始值 FFH。如果 SBF 宽度为 10 位或更少位, 则接收作为错误处理而终止而不输出中断, 并返回 SBF 接收模式。此时, UCnSRF 位不被清除。

- 注意事项 1. 如果数据接收期间 SBF 被发送, 会发生一个帧错误。
2. SBF 接收 (UCnSRF = 1) 期间, 不要设置 SBF 接收触发位 (UCnSRT) 和 SBF 发送触发位 (UCnSTT) 为 1。

图 17-11. SBF 接收



17.6.5 UART 发送

通过设置 UCnCTL0.UCnPWR 位为 1，输入一个高电平至 TXDCn 引脚。

之后，通过设置 UCnCTL0.UCnTXE 位为 1 设定发送允许状态，并且，通过写入发送数据至 UCnTX 寄存器启动发送。起始位、校验位以及停止位自动添加。

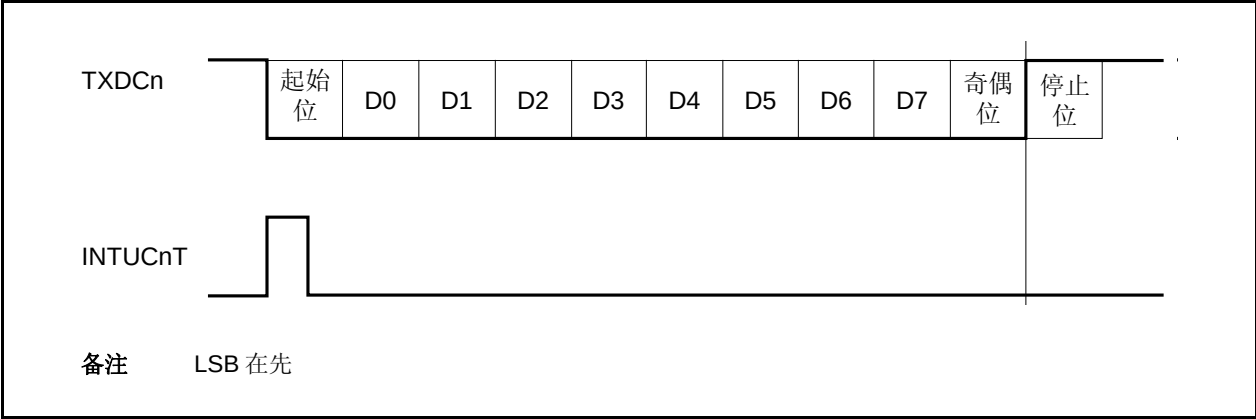
由于 UARTCn 中不提供 CTS（发送允许信号）输入引脚，故而使用一个端口检查发送目的方的接收允许信号。

当发送操作启动之时，UCnTX 寄存器内的数据传输至 UARTCn 发送移位寄存器。

UCnTX 寄存器的数据发送至 UARTCn 发送移位寄存器完成时，发送使能中断请求信号（INTUCnT）产生，之后 UARTCn 发送移位寄存器的内容输出至 TXDCn 引脚。

INTUCnT 信号产生后，允许下一个发送数据写入 UCnTX 寄存器。

图 17-12. UART 发送



17.6.6 连续发送过程

当 UARTCn 发送移位寄存器启动移位操作时，UARTCn 可以写入下一个数据至 UCnTX 寄存器。可以由发送允许中断请求信号 (INTUCnT) 来判断 UARTCn 发送移位寄存器的发送时序。通信速率的效能依赖于在传输中将下一个待发送数据写入 UCnTX 寄存器。

注意事项 连续发送期间进行初始化时，确保 UCnSTR.UCnTSF 位为 0，然后执行初始化。UCnTSF 位为 1 时被初始化的发送数据得不到保证。

图 17-13. 连续发送处理框图

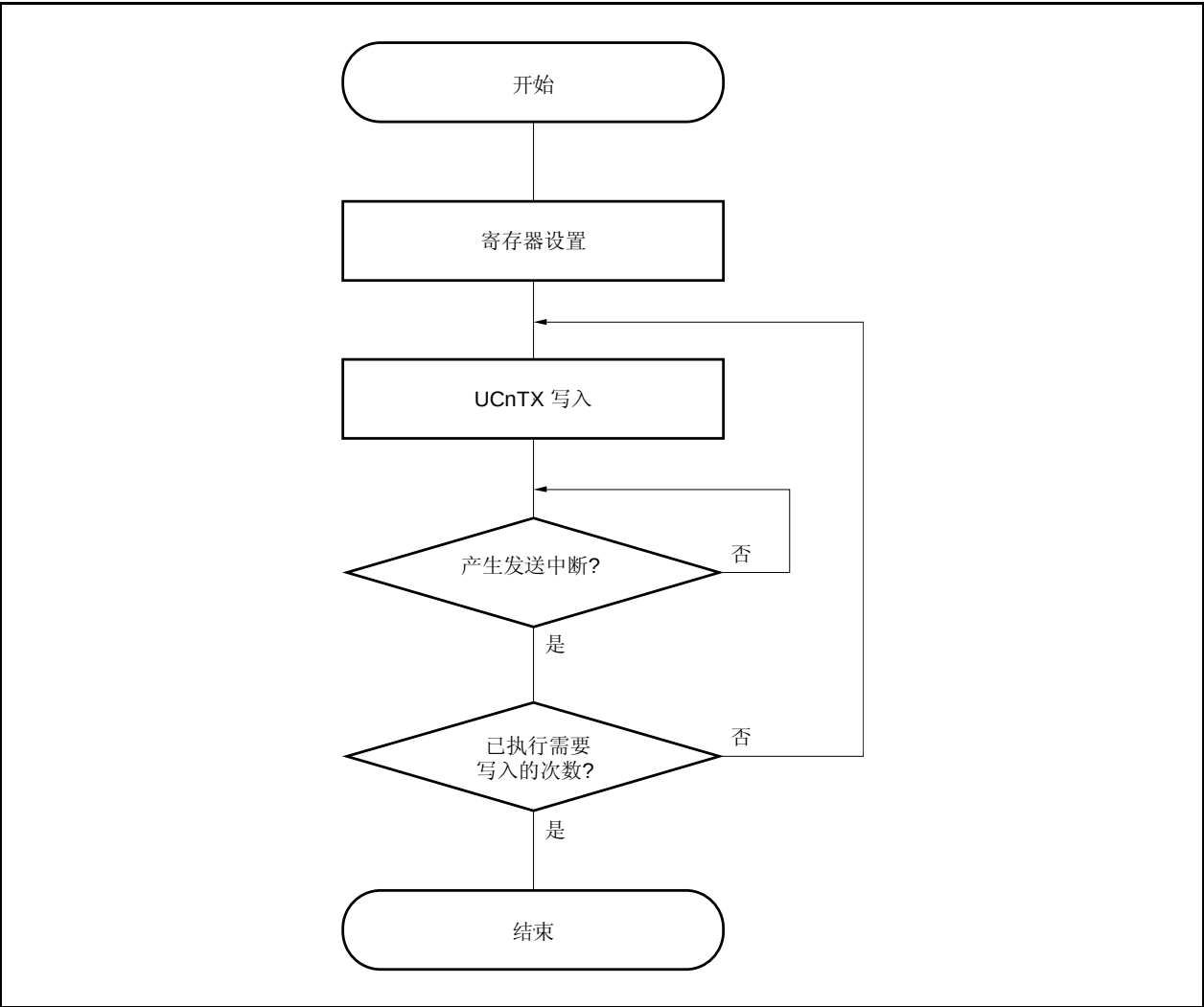
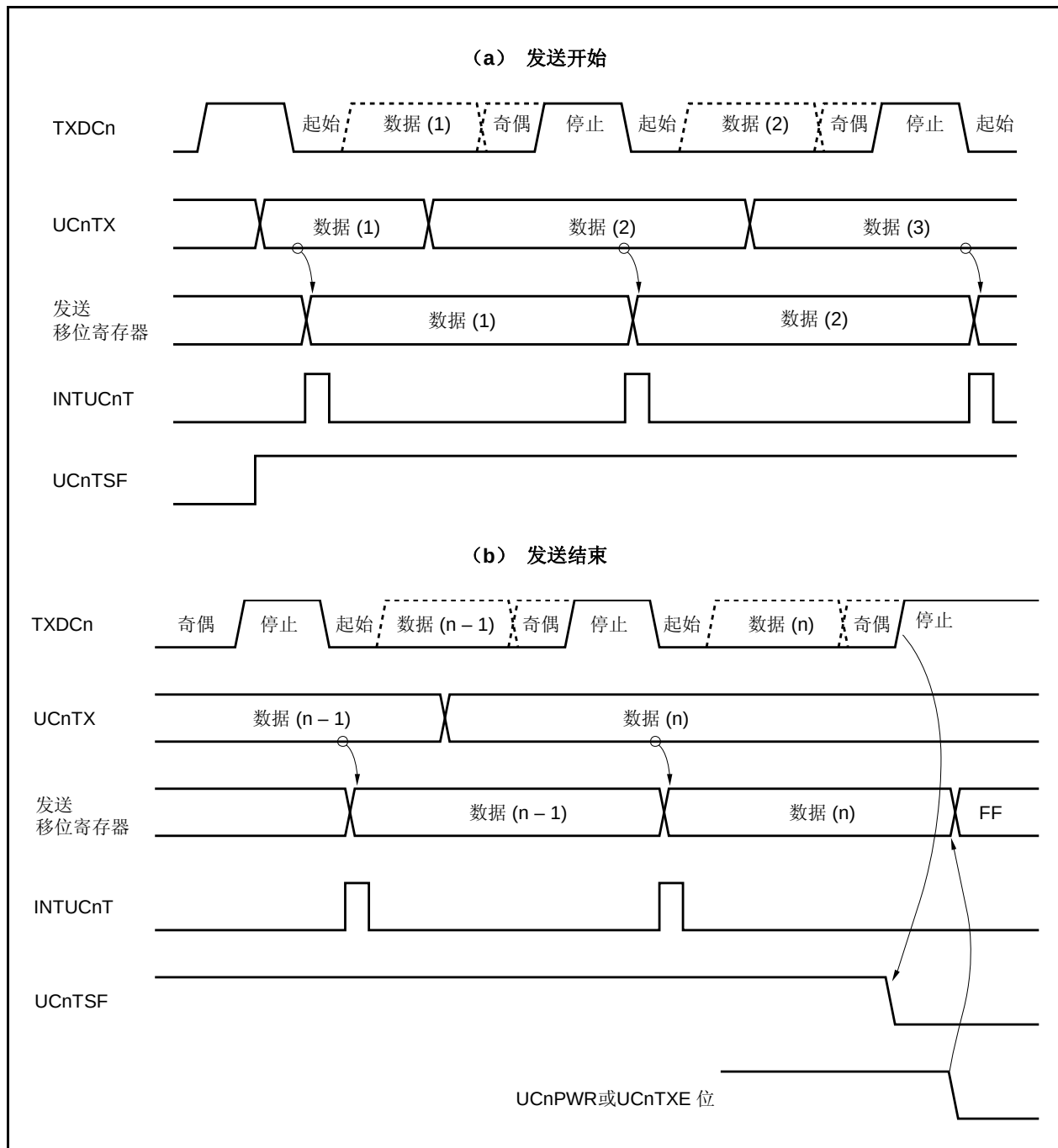


图 17-14. 连续发送操作时序



17.6.7 UART 接收

通过设置 UCnCTL0.UCnPWR 位为 1 然后设置 UCnCTL0.UCnRXE 位为 1 来设置接收等待状态。接收等待状态下，RXDCn 引脚被监控并且进行起始位检测。

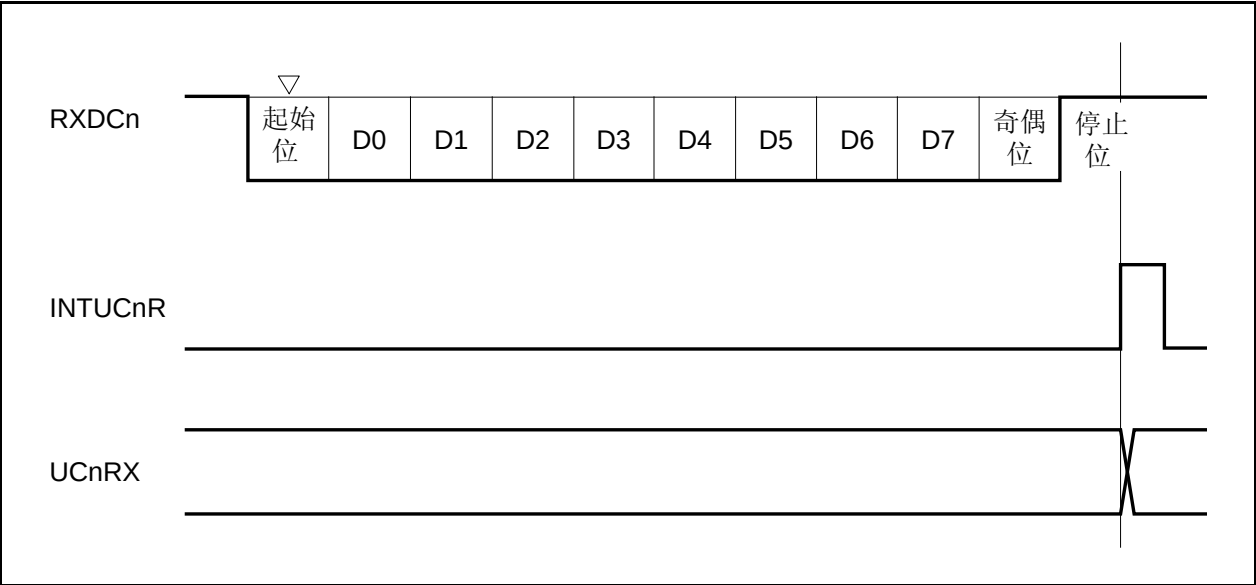
应用两步检测程序进行起始位检测。

第一步检测到 RXDCn 引脚的上升沿并在下降沿开始采样。如果 RXDCn 引脚在起始位采样点上为低电平，确认为起始位。起始位得到确认以后，开始接收操作，按照设定的波特率，串行数据被存入 UARTCn 接收移位寄存器。

基于接收到停止位而输出接收完成中断请求信号（INTUCnR）时，UARTCn 接收移位寄存器的数据写入 UCnRX 寄存器。然而，如果发生超限（UCnSTR.UCnOVE 位）错误，此时的接收数据不能写入 UCnRX 寄存器而被丢弃。

即使接收期间发生奇偶错误（UCnSTR.UCnPE 位）或帧错误（UCnSTR.UCnFE 位），接收会继续进行直到第一个停止位的接收位置，并在接收完成后输出 INTUCnR。

图 17-15. UART 接收



- 注意事项
1. 即使发生接收错误，也务必读取 UCnRX 寄存器。如果不读取 UCnRX 寄存器，则下一个数据接收期间发生超限错误，并且接收错误连续发生不确定。
 2. 接收期间假设只有一个停止位进行操作。第二个停止位被忽略。
 3. 当接收完毕后，在接收结束中断请求信号（INTUCnR）已经产生后读取 UCnRX 寄存器，并将 UCnPWR 或 UCnRXE 位清 0。如果 UCnPWR 或 UCnRXE 位在 INTUCnR 信号产生之前清 0，不能保证对 UCnRX 寄存器值的正确读取。
 4. 如果 UARTCn 的接收结束处理（INTUCnR 信号产生）和 UCnPWR 位 = 0 或 UCnRXE 位 = 0 相冲突，则即使 UCnRX 寄存器内没有存储数据，INTUCnR 信号也可能产生。
若要不等待 INTUCnR 信号产生而完成接收，确保将中断控制寄存器（UCnRIC）的中断屏蔽标志（UCnRMK）设置为 1，将 UCnPWR 位或 UCnRXE 位清除为 0，然后清除 UCnRIC 寄存器的中断请求标志（UCnRIF）。

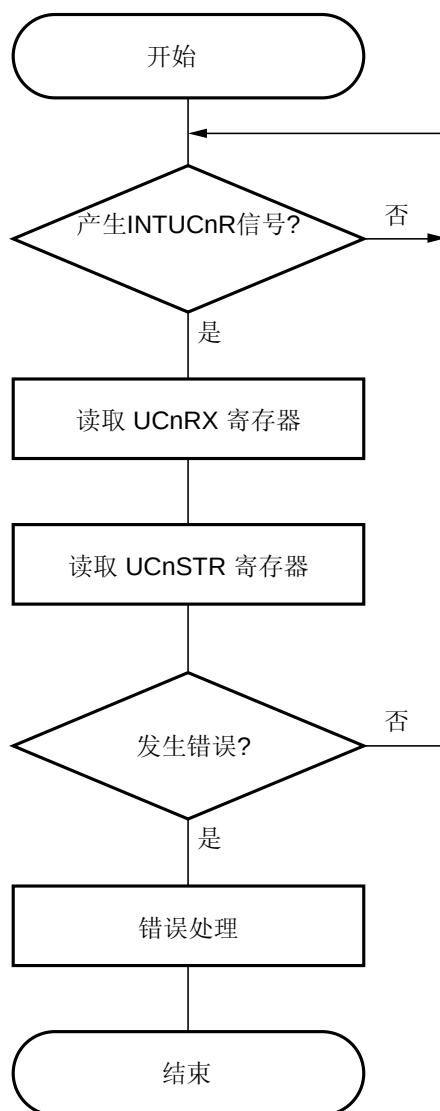
17.6.8 接收错误

接收操作期间的错误有三类：奇偶错误，帧错误和超限错误。数据接收结果错误标志设置在 UCnSTR 寄存器内，当错误发生时，输出接收完成中断请求信号（INTUCnR）。

通过读取 UCnSTR 寄存器的内容，可以确定在接收过程中发生了哪种错误。

读取完毕后，写入 0 来清除接收错误标志。

图 17-16. 接收数据读取流程



注意事项 INTUCnR 信号发生时，必须读取 UCnSTR 寄存器以检查错误。

接收错误原因

错误标志	接收错误	原因
UCnPE	奇偶错误	接收的校验位和设定的不匹配。
UCnFE	帧错误	未检测到停止位。
UCnOVE	超限错误	在数据从接收缓冲读取之前下一个数据接收完成。

发生接收错误时，根据错误类型执行以下步骤：

- 奇偶错误
如果由于类似接收过程中的噪声问题而接收到不正确的数据，则丢弃所接收数据并重新发送。
- 帧错误
接收端和发送端之间可能发生波特率错误或者错误的检测到起始位。因为这对通信格式来说是致命错误，发送端检查操作停止，各自进行初始化处理，然后重新开始通信。
- 超限错误
如果在读取接收数据之前完成下一个数据接收，1 帧数据会丢弃。如果需要该数据，则进行重新发送。

注意事项 如果连续接收期间发生接收错误中断，在下一个数据接收完成前必须读取 UCnSTR 寄存器的内容，然后进行错误处理。

17.6.9 奇偶类型和操作

注意事项 使用 LIN 功能时，将 UCnCTL0.UCnPS1 和 UCnCTL0.UCnPS0 位固定为 00。

奇偶校验位用来检测通信数据中的位错误。通常情况下，发送端和接收端应用相同的检验。
在使用奇校验和偶校验情况下，可以检测到奇数个位错误。而在 0 奇偶校验和无校验情况下，不能检测到错误。

(a) 偶校验。

(i) 在发送期间

发送数据之中值为“1”的比特数，包括奇偶校验位，控制其数目为偶数。奇偶检验位的值计算如下：

- 发送数据中值为“1”的比特数目为奇数：1
- 发送数据中值为“1”的比特数目为偶数：0

(ii) 在接收期间

接收数据中值为“1”的比特数，包括校验位一并计算，如果数目为奇数，则输出奇偶校验错误。

(b) 奇数校验

(i) 在发送期间

与偶校验相对，发送数据之中值为“1”的比特数，包括奇偶校验位，控制其数目为奇数。奇偶检验位的值计算如下：

- 发送数据中值为“1”的比特数目为奇数：0
- 发送数据中值为“1”的比特数目为偶数：1

(ii) 在接收期间

接收数据中值为“1”的比特数，包括校验位一并计算，如果数目为偶数，则输出奇偶校验错误。

(c) 0 奇偶校验

发送过程中，不管传输数据怎样，奇偶校验位总是为 0。

在接收期间，不执行奇偶校验。因而，无论奇偶位为 0 或 1，均不产生奇偶错误。

(d) 无奇偶校验

不向发送数据附加奇偶位。

设定无奇偶位进行接收。因为无奇偶位所以没有奇偶错误发生。

17.6.10 接收数据噪声滤波器

本滤波器使用预分频器输出的基础时钟采样 RXDCn 引脚。

如果相同的采样值被读到两次，则匹配检测器输出改变，并且 RXDCn 信号被采样为输入数据。所以，不超过两个时钟宽度的数据被判定为噪声，而不会提供给内部电路（参见 图 17-18）。有关基础时钟，参见 17.7 (1) (a) 基础时钟。

另外，由于电路如图 17-17 所示，所以，接收操作的内部处理过程与外部信号状态相比，延迟了 3 个时钟周期。

图 17-17. 噪声过滤器电路

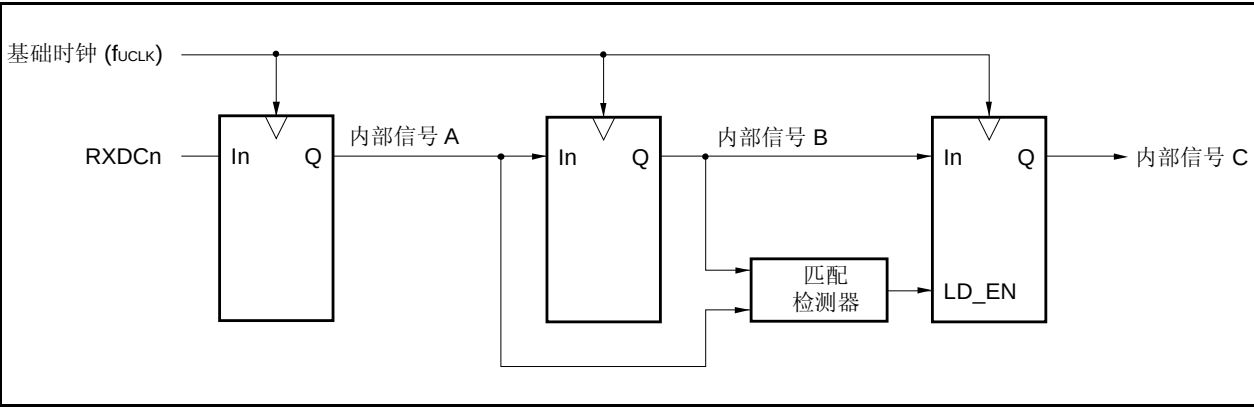
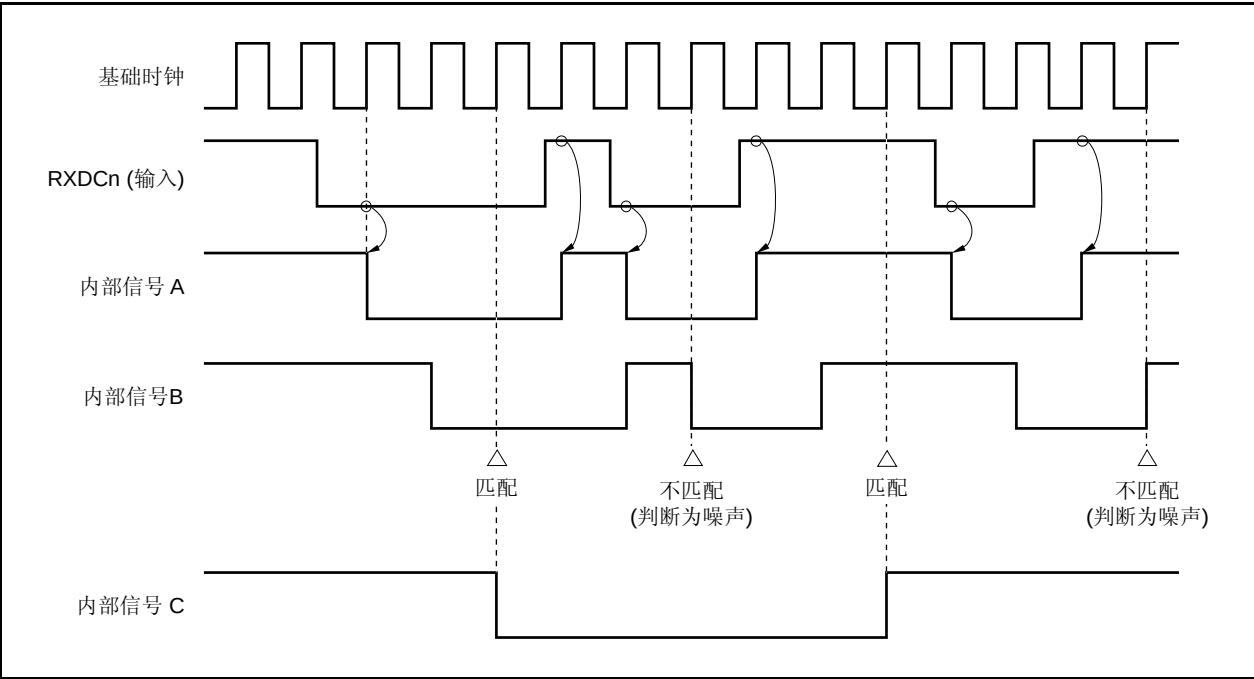


图 17-18. RXDCn 信号判定为噪声的时序



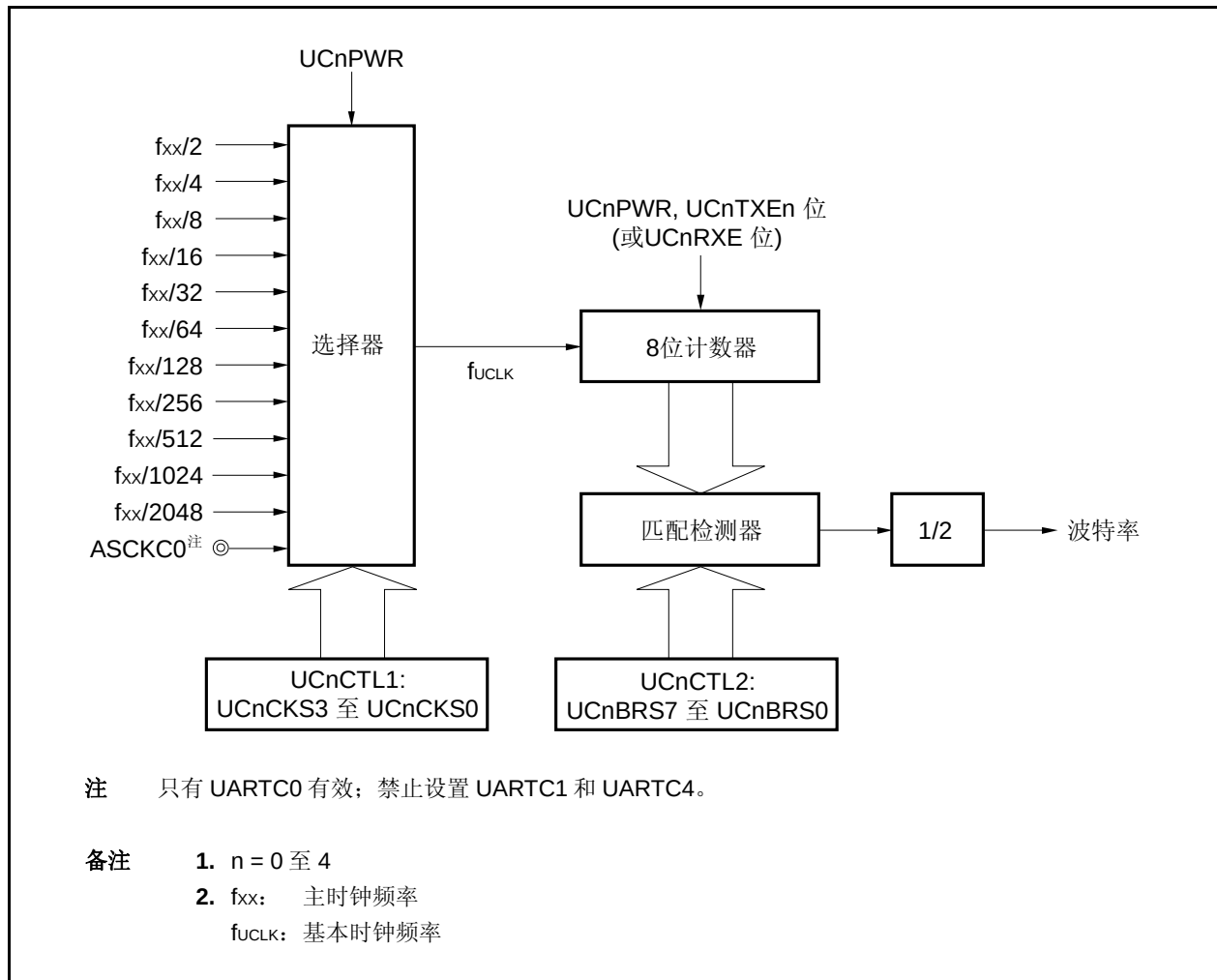
17.7 专用波特率发生器

专用波特率发生器由时钟源选择器模块和一个 8 位可编程计数器构成，它在使用 UARTCn 进行发送和接收时产生一个串行时钟。专用波特率发生器的输出可以为每一个通道选择串行时钟。

一个 8 位计数器用于发送，另一个用于接收。

(1) 波特率发生器的配置

图 17-19. 波特率发生器的配置



(a) 基本时钟

当 UCnCTL0.UCnPWR 位为 1 时，由 UCnCTL1.UCNCKS3 至 UCnCTL1.UCNCKS0 位所选择的时钟提供给 8 位计数器。此时钟称作基本时钟 (f_{UCLK})。

(b) 串行时钟的产生

通过设置 UCnCTL1 寄存器和 UCnCTL2 寄存器来产生一个串行时钟 ($n = 0$ 至 4)。

基础时钟由 UCnCTL1.UCNCKS3 至 UCnCTL1.UCNCKS0 位选择。

8 位计数器的分频值可以使用 UCnCTL2.UCNBRS7 至 UCnCTL2.UCNBRS0 位进行设置。

(2) UARTCn 控制寄存器 1 (UCnCTL1)

UCnCTL1 寄存器为 8 位寄存器，用来选择 UARTCn 基础时钟。

该寄存器可按 8 位为单位进行读写。

复位后本寄存器被设置为 00H。

注意事项 改写 UCnCTL1 寄存器之前将 UCnCTL0.UCnPWR 位清 0。

复位后: 00H R/W 地址: UC0CTL1 FFFFFFFA01H, UC1CTL1 FFFFFFFA11H,
UC2CTL1 FFFFFFFA21H, UC3CTL1 FFFFFFFA31H,
UC4CTL1 FFFFFFFA41H

	7	6	5	4	3	2	1	0
UCnCTL1	0	0	0	0	UCnCKS3	UCnCKS2	UCnCKS1	UCnCKS0

(n = 0 至 4)

UCnCKS3	UCnCKS2	UCnCKS1	UCnCKS0	基础时钟 (f _{CLK}) 选择
0	0	0	0	f _{xx} /2
0	0	0	1	f _{xx} /4
0	0	1	0	f _{xx} /8
0	0	1	1	f _{xx} /16
0	1	0	0	f _{xx} /32
0	1	0	1	f _{xx} /64
0	1	1	0	f _{xx} /128
0	1	1	1	f _{xx} /256
1	0	0	0	f _{xx} /512
1	0	0	1	f _{xx} /1,024
1	0	1	0	f _{xx} /2,048
1	0	1	1	外部时钟 ^注 (ASCKC0 引脚)
其它				禁止设置

注 只有 UARTC0 有效；禁止设置 UARTC1 至 UARTC4。

备注 f_{xx}: 主时钟频率

(3) UARTCn 控制寄存器 2 (UCnCTL2)

UCnCTL2 寄存器为 8 位寄存器，用来选择 UARTCn 的波特率（串行传输速率）时钟。

该寄存器可按 8 位为单位进行读写。

复位将此寄存器 设置为 FFH。

注意事项 改写 UCnCTL2 寄存器之前将 UCnCTL0.UCnPWR 位清除为 0 或将 UCnTXE 和 UCnRXE 位清除为 00。

复位后 FFH R/W 地址: UC0CTL2 FFFFFFFA02H, UC1CTL2 FFFFFFFA12H,
UC2CTL2 FFFFFFFA22H, UC3CTL2 FFFFFFFA32H,
UC4CTL2 FFFFFFFA42H

	7	6	5	4	3	2	1	0
UCnCTL2	UCnBRS7	UCnBRS6	UCnBRS5	UCnBRS4	UCnBRS3	UCnBRS2	UCnBRS1	UCnBRS0

(n = 0 至 4)

UCn BRS7	UCn BRS6	UCn BRS5	UCn BRS4	UCn BRS3	UCn BRS2	UCn BRS1	UCn BRS0	默认值 (k)	串行 时钟
0	0	0	0	0	0	X	X	X	禁止 设置
0	0	0	0	0	1	0	0	4	f _{UCLK} /4
0	0	0	0	0	1	0	1	5	f _{UCLK} /5
0	0	0	0	0	1	1	0	6	f _{UCLK} /6
:	:	:	:	:	:	:	:	:	:
1	1	1	1	1	1	0	0	252	f _{UCLK} /252
1	1	1	1	1	1	0	1	253	f _{UCLK} /253
1	1	1	1	1	1	1	0	254	f _{UCLK} /254
1	1	1	1	1	1	1	1	255	f _{UCLK} /255

备注 f_{UCLK}: 由 UCnCTL1.UCnCKS3 至 UCnCTL1.UCnCKS0 位选择的
时钟频率

(4) 波特率

波特率由以下公式计算得到：

$$\text{波特率} = \frac{f_{\text{CLK}}}{2 \times k} [\text{bps}]$$

使用内部时钟时，按以下公式（用 ASCKC0 引脚作为 UARTC0 处的时钟时，按上式计算）

$$\text{波特率} = \frac{f_{\text{XX}}}{2^{m+1} \times k} [\text{bps}]$$

备注 f_{CLK} = 由 UCnCTL1.UCnCKS3 至 UCnCTL1.UCnCKS0 位选择的基础时钟频率
 f_{XX} : 主时钟频率
 m = 由 UCnCTL1.UCnCKS3 至 UCnCTL1.UCnCKS0 位设置的值 ($m = 0$ 至 10)
 k = 由 UCnCTL2.UCnBRS7 至 UCnCTL2.UCnBRS0 位设置的值 ($k = 4$ 至 255)

波特率误差可以由以下公式计算得到：

$$\begin{aligned} \text{误差} (\%) &= \left[\frac{\text{实际波特率 (有误差的波特率)}}{\text{目标波特率 (校正波特率)}} - 1 \right] \times 100 [\%] \\ &= \left[\frac{f_{\text{CLK}}}{2 \times k \times \text{目标波特率}} - 1 \right] \times 100 [\%] \end{aligned}$$

使用内部时钟时，按以下公式（用 ASCKC0 引脚输入作为 UARTC0 处的时钟时，按上式计算波特率误差）

$$\text{误差} (\%) = \left[\frac{f_{\text{XX}}}{2^{m+1} \times k \times \text{目标波特率}} - 1 \right] \times 100 [\%]$$

注意事项 1. 发送过程中，波特率误差必须限于接收端的误差容限之内。
 2. 接收时的波特率误差必须满足由一定范围，该范围在 (5) 接收中允许的波特率误差范围中指明。

要设置波特率，按以下算式设置 UCnCTL1 和 UCnCTL2 寄存器（使用内部时钟时）。

- <1> 设置 k 等于 $f_{xx}/2/ (2 \times \text{目标波特率})$ ，m 等于 0。
 <2> 如果 k 是 256 或更大 ($k \geq 256$)，将 k 减小一半 ($k/2$)，m 加 1 ($m + 1$)。
 <3> 重复第<2>步，直到 k 小于 256 ($k < 256$)。
 <4> 对 k 的第一位小数进行四舍五入，得到最接近的整数。
 如果四舍五入后 k 为 256，再次执行第<2>步将 k 设置为 128。
 <5> 将 m 的值设置到 UCnCTL1 寄存器，k 的值设置到 UCnCTL2 寄存器。

示例： 如果 $f_{xx} = 48 \text{ MHz}$ 而目标波特率 = 153, 600 bps
 <1> $k = 480,000,000/2/ (2 \times 153,600) = 78.125\dots$ ， $m = 0$
 <2>，<3> $k = 78.125\dots < 256$ ， $m = 0$
 UCnCTL2 寄存器的设置值： $k = 78 = 4\text{EH}$ ，UCnCTL1 寄存器的设置值： $m = 0$

实际波特率 $= 48,000,000/2/ (2 \times 78)$
 $= 153,846 \text{ [bps]}$

波特率误差 $= \{48,000,000/2/ (2 \times 78 \times 153,600) - 1\} \times 100$
 $= 0.160 \text{ [%]}$

以下为波特率设置的典型举例：

表 17-4. 波特率发生器设置数值

波特率 (bps)	$f_{xx} = 48 \text{ MHz}$			$f_{xx} = 32 \text{ MHz}$			$f_{xx} = 24 \text{ MHz}$		
	UCnCTL1	UCnCTL2	ERR (%)	UCnCTL1	UCnCTL2	ERR (%)	UCnCTL1	UCnCTL2	ERR (%)
300	08H	9CH	0.16	07H	D0H	0.16	07H	9CH	-2.3
600	07H	9CH	0.16	06H	D0H	0.16	06H	9CH	0.16
1,200	06H	9CH	0.16	05H	D0H	0.16	05H	9CH	0.16
2,400	05H	9CH	0.16	04H	D0H	0.16	04H	9CH	0.16
4,800	04H	9CH	0.16	03H	D0H	0.16	03H	9CH	0.16
9,600	03H	9CH	0.16	02H	D0H	0.16	02H	9CH	0.16
19,200	02H	9CH	0.16	01H	D0H	0.16	01H	9CH	0.16
31,250	01H	C0H	0.00	01H	80H	0.00	00H	C0H	0.00
38,400	01H	9CH	0.16	00H	D0H	0.16	00H	9CH	0.16
76,800	00H	9CH	0.16	00H	68H	0.16	00H	4EH	0.16
153,600	00H	4EH	0.16	00H	34H	0.16	00H	27H	0.16
312,500	00H	26H	1.05	00H	1AH	-1.54	00H	13H	1.05
625,000	00H	13H	1.05	00H	0DH	-1.54	00H	0AH	-4.00
1,000,000	00H	0CH	0.00	00H	08H	0.00	00H	06H	0.00
1,250,000	00H	0AH	-4.00	禁止设置			00H	05H	-4.00

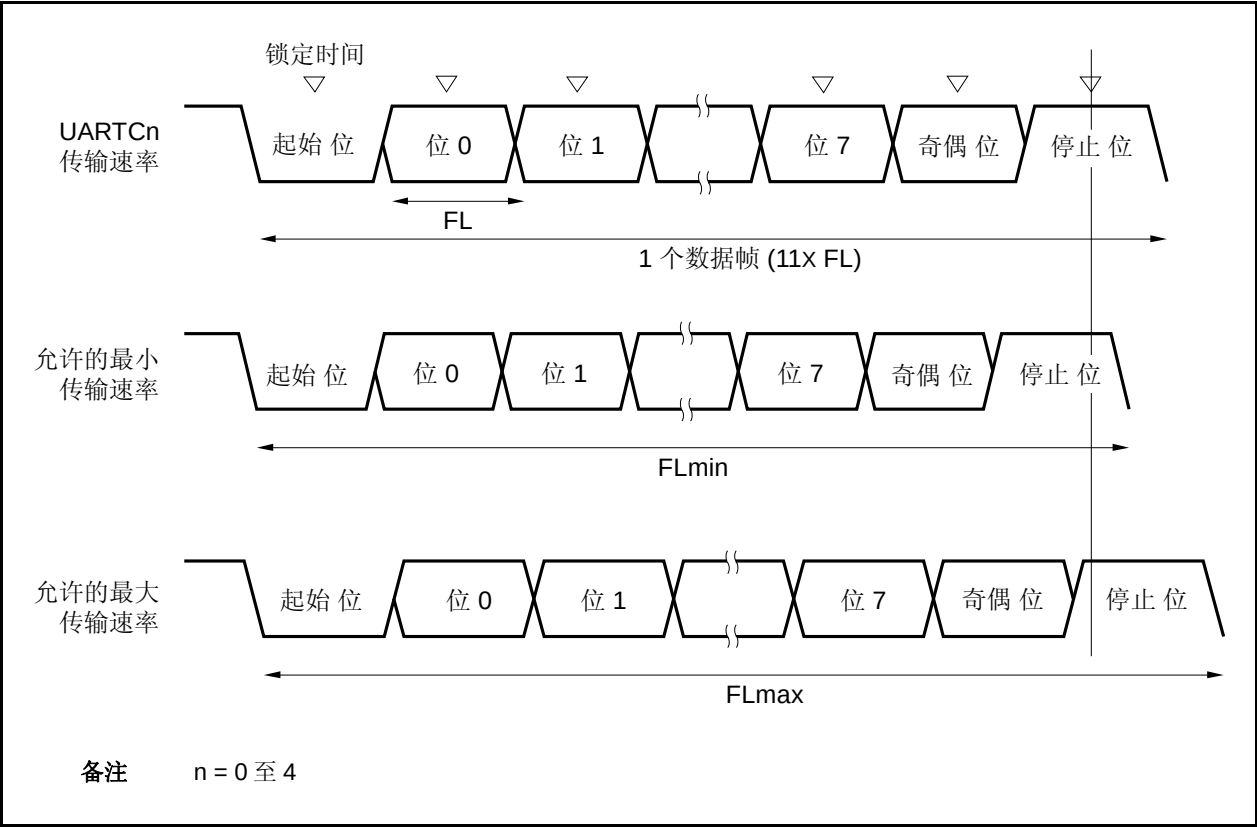
备注 f_{xx} : 主时钟频率
 ERR: 波特率误差 (%)。

(5) 接收中允许的波特率误差范围

目的端接收中允许的波特率误差范围如下所示：

注意事项 接收时的波特率误差必须使用以下等式设置在允许的范围之内。

图 17-20. 接收中允许的波特率范围



如图 17-20 所示，接收数据的锁存时刻取决于检测到起始位后使用 UCnCTL2 寄存器对计数器的设置。如果在此锁存时刻可以一直如期接收到最末数据（停止位），发送数据就可以正常接收。
当应用于 11 位接收时，以下为理论结果：

$$FL = (\text{波特})^{-1}$$

波特：UARTCn 波特率 (n = 0 至 4)
k: UCnCTL2.UCnBRS7 至 UCnCTL2.UCnBRS0 位的设定值 (n = 0 至 4)
FL: 1 位数据长度
锁存时间宽度: 2 个时钟

最小允许传输速率:
$$FL_{min} = 11 \times FL - \frac{k-2}{2k} \times FL = \frac{21k+2}{2k} FL$$

所以，可以由目的端接收的最大波特率如下所示：

$$BR_{\max} = (FL_{\min}/11)^{-1} = \frac{22k}{21k + 2} \text{ 波特}$$

相似地，遵从以下算式计算获得最大许可波特率：

$$\frac{10}{11} \times FL_{\max} = 11 \times FL - \frac{k+2}{2 \times k} \times FL = \frac{21k-2}{2 \times k} FL$$

$$FL_{\max} = \frac{21k-2}{20k} FL \times 11$$

所以，可以由目的端接收的最小波特率如下所示：

$$BR_{\min} = (FL_{\max}/11)^{-1} = \frac{20k}{21k - 2} \text{ 波特}$$

由上述等式获得 UARTCn 以及目的端的允许波特率误差，以此遵从以下表格得到最大波特率值。

表 17-5. 最大/最小允许波特率误差

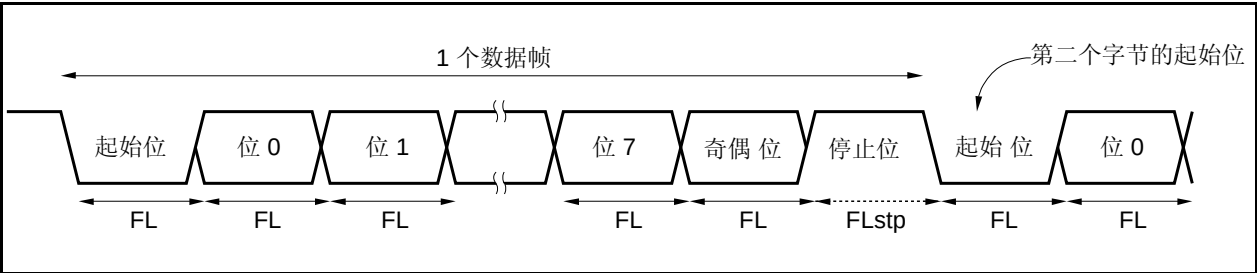
比率 (k)	允许的最大波特率误差	最小允许波特率误差
4	+2.32%	-2.43%
8	+3.52%	-3.61%
20	+4.26%	-4.30%
50	+4.56%	-4.58%
100	+4.66%	-4.67%
255	+4.72%	-4.72%

- 备注**
1. 接收精度依赖于 1 帧内的比特数、输入时钟频率以及分频率（k）。输入时钟频率越高、分频率越大，精度就越高。
 2. k: UCnCTL2.UCnBRS7 至 UCnCTL2.UCnBRS0 位的设定值（n = 0 至 4）

(6) 连续发送时的传输速率

在连续发送之中，从停止位到下一个起始位的转换速度通常为 2 个基础时钟长度。而初始化由接收端通过检测起始位进行，故而这对传输结果没有影响。

图 17-21. 连续传输时的转换速度



假设为 1 位数据长度：FL;停止位长度：FLstp; 基础时钟频率：f_{UCLK}，则可得以下等式。

$$FLstp = FL + 2/f_{UCLK}$$

所以，连续发送时的传输速率如下：

$$\text{传输速率} = 11 \times FL + (2/f_{UCLK})$$

17.8 注意事项

- (1) 当供给 UARTCn 的时钟停止时（如：处于 IDLE1，IDLE2，或 STOP 模式中），则操作停止，每一个寄存器都保留时钟供应停止前的瞬时值。TXDCn 引脚也保持并输出时钟供应停止前的瞬时值。然而，时钟恢复之后不能保证操作正常。所以，时钟恢复之后，应该通过设置 UCnCTL0.UCnPWR，UCnCTL0.UCnRXEn 和 UCnCTL0.UCnTXEn 位为 000，对电路进行初始化。
- (2) RXDC1 和 KR7 引脚不能同时使用。使用 RXDC1 引脚时，不要使用 KR7 引脚功能。若使用 KR7 引脚，就不要使用 RXDC1 引脚功能（建议将 PFC91 位设置为 1 并清除 PFCE91 位为 0）。
- (3) 按以下步骤开始 UARTCn 操作。
 - <1> 将 UCnCTL0.UCnPWR 位设置为 1。
 - <2> 设置端口。
 - <3> 将 UCnCTL0.UCnTXE 位设置为 1，UCnCTL0.UCnRXE 位置 1。
- (4) 按以下步骤停止 UARTCn 操作。
 - <1> 将 UCnCTL0.UCnTXE 位设置为 0，UCnCTL0.UCnRXE 位置 0。
 - <2> 设置端口并设置 UCnCTL0.UCnPWR 位为 0（如果端口设置不改变则不存在该问题）
- (5) 发送模式中（UCnCTL0.UCnPWR 位 = 1 并且 UCnCTL0.UCnTXE 位 = 1），因为写入 UCnTX 寄存器会开始发送，所以不要用软件给该寄存器写入相同的值。若连续发送相同值，则重写相同的值。
- (6) 在连续发送之中，从停止位到下一个起始位的通信速度比通常要延长 2 个基础时钟长度。然而，接收端通过检测起始位进行初始化时序，因此接收结果不受影响。

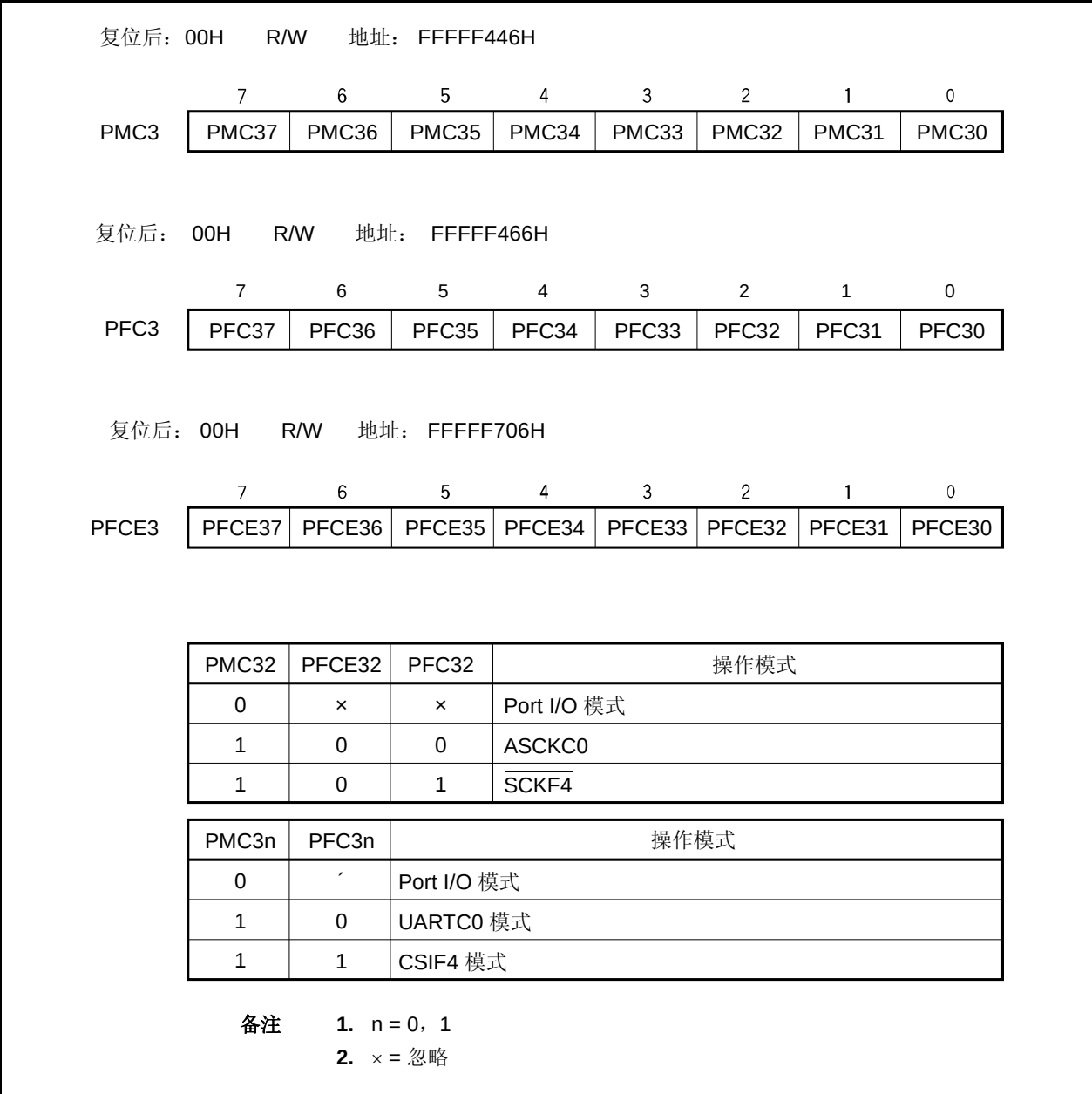
18.1 CSIF和其它串行接口模式转换

18.1.1 CSIF4 和 UARTC0 模式转换

V850ES/JG3-H 和 V850ES/JH3-H 中, CSIF4 和 UARTC0 是相同引脚的复用功能, 因而不能同时使用。 要使用 CSIF4, 必须使用 PMC3、PFC3 和 PFCE3 寄存器设置 CSIF4 的使用。

注意事项 如果发送或接收期间转换这些功能, CSIF4 和 UARTC0 的发送/接收操作得不到保证。确保禁止未使用的模式。

图 18-1. CSIF4 和 UARTC0 模式转换设置



18.1.2 CSIF0、UARTC4 和 I²C01 模式转换

V850ES/JG3-H 和 V850ES/JH3-H 中，CSIF0、UARTC4 以及 I²C01 是相同引脚的复用功能，因而不能同时使用。必须使用 PMC4、PFC4 和 PFCE4 寄存器对 CSIF0、UARTC4 和 I²C01 之间的转换预先设置。

注意事项 如果发送或接收期间转换这些功能，CSIF0、UARTC4 和 I²C01 的发送/接收操作得不到保证。 确保禁止未使用的模式。

图 18-2. CSIF0、UARTC4 和 I²C01 模式转换设置

复位后: 00H R/W 地址: FFFFF448H

	7	6	5	4	3	2	1	0
PMC4	0	0	0	0	0	PMC42	PMC41	PMC40

复位后: 00H R/W 地址: FFFFF468H

	7	6	5	4	3	2	1	0
PFC4	0	0	0	0	0	PFC42	PFC41	PFC40

复位后: 00H R/W 地址: FFFFF708H

	7	6	5	4	3	2	1	0
PFCE4	0	0	0	0	0	0	PFCE41	PFCE40

PMC4n	PFCE4n	PFC4n	操作模式
0	×	×	端口 I/O 模式
1	0	0	CSIF0 模式
1	0	1	UARTC4 模式
1	1	0	I ² C01 模式

备注

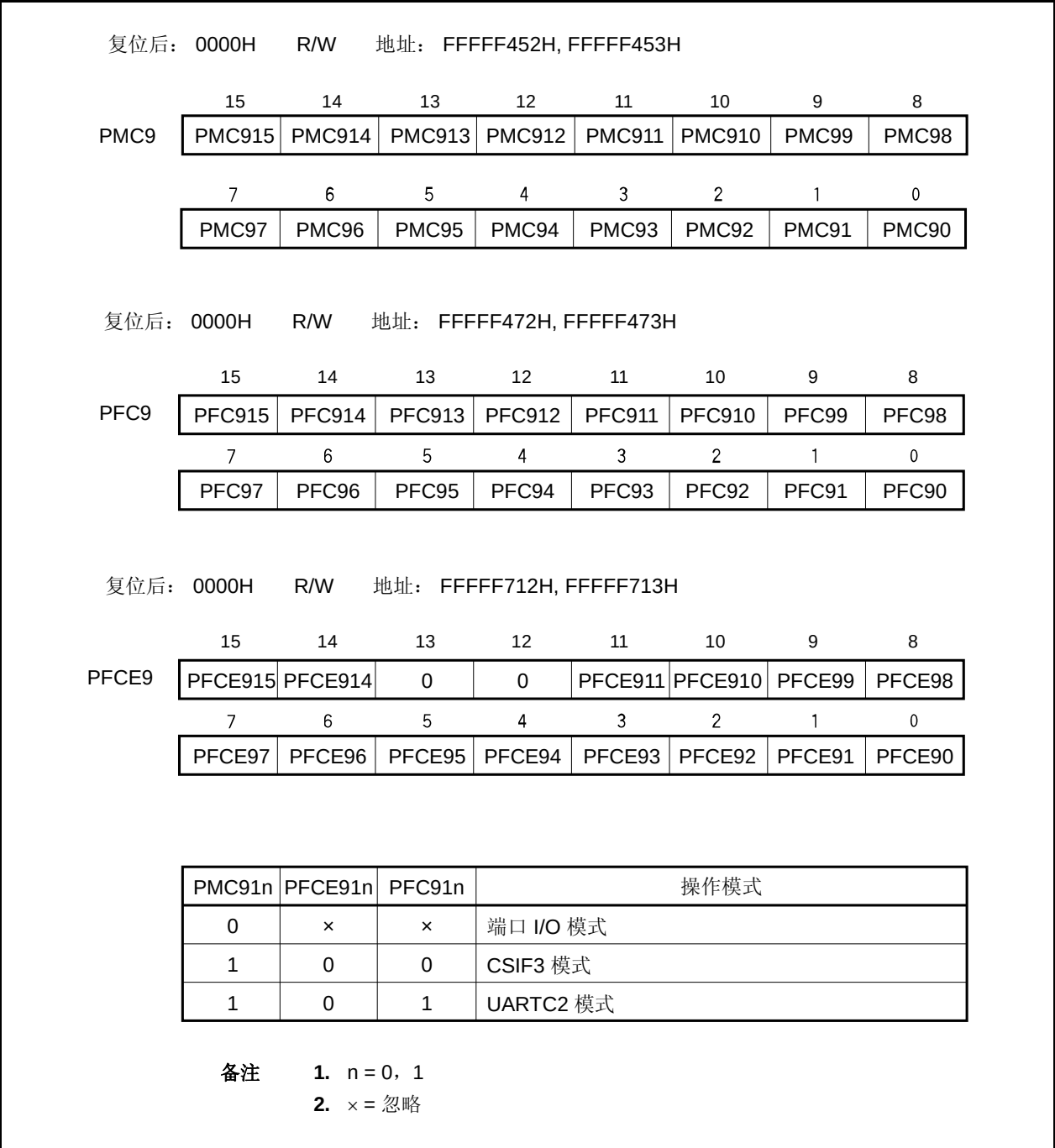
1. n = 0, 1
2. × = 忽略

18.1.3 CSIF3 和UARTC2 模式转换

V850ES/JG3-H 和 V850ES/JH3-H 中，CSIF3 和 UARTC2 是相同引脚的复用功能，因而不能同时使用。 必须使用 PMC9、PFC9 和 PFCE9 寄存器对 CSIF3 和 UARTC2 之间的转换预先设置。

注意事项 如果发送或接收期间转换这些功能，CSIF3 和 UARTC2 的发送/接收操作得不到保证。 确保禁止未使用的模式。

图 18-3. CSIF3 和 UARTC2 模式转换设置



18.2 特性

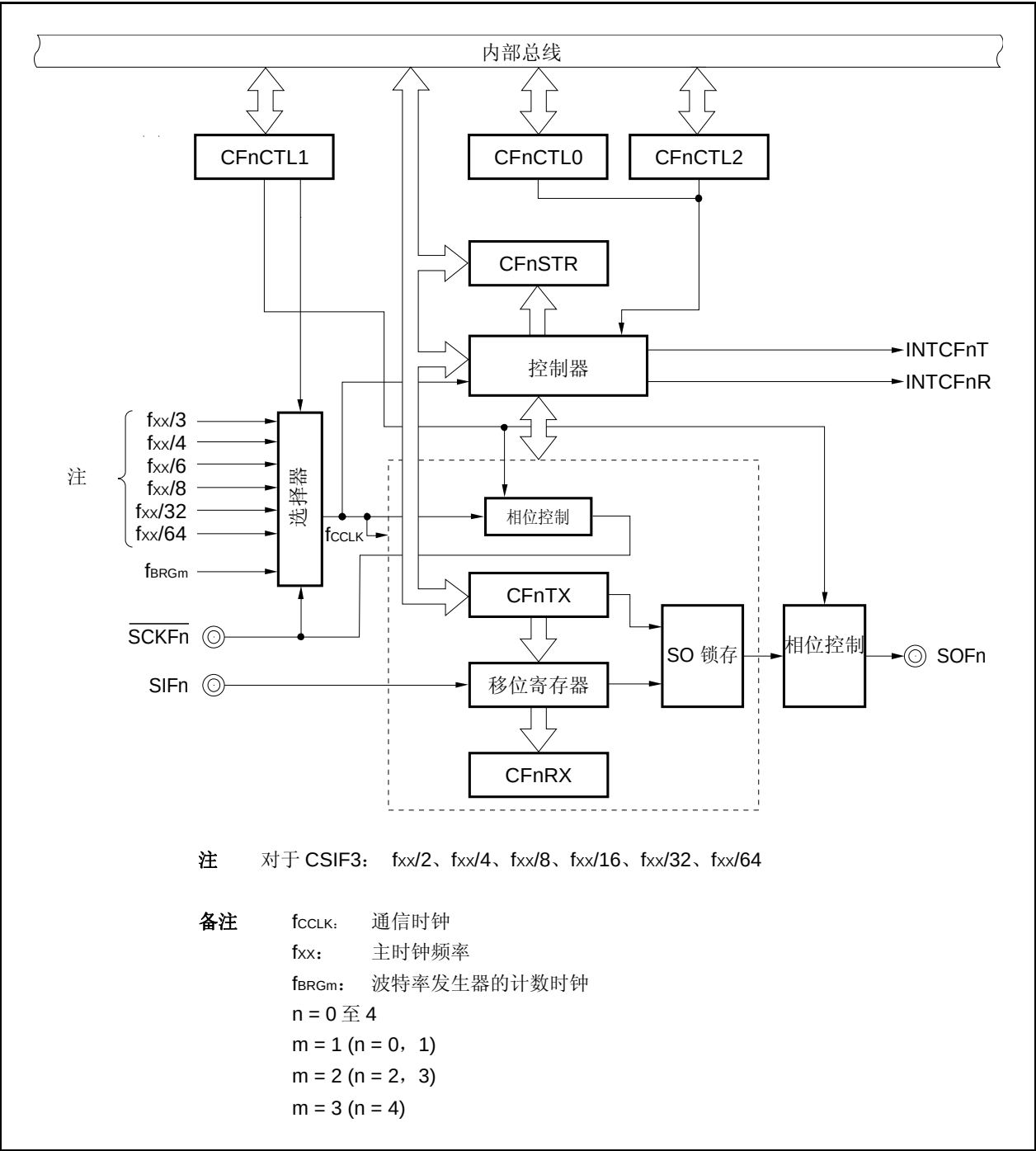
- 传输速率: 最大 12 Mbps ($f_{xx} = 48 \text{ MHz}$, 使用内部时钟, CSIF3)
 8 Mbps ($f_{xx} = 48 \text{ MHz}$, 使用内部时钟, CSIF0 至 CSIF2, CSIF4)
- 主机、从机模式可选
- 8 位至 16 位传输, 3-线串行接口
- 中断请求信号 (INTCFnT, INTCFnR)
- 串行时钟和数据相位可翻转
- 以 1-位为单位, 选择 8 位数据至 16 位之间传输数据
- 可切换 MSB 先行/LSB 先行传输数据
- 3-线传输 SOFn: 串行数据输出
 SIFn: 串行数据输入
 $\overline{\text{SCKFn}}$: 串行时钟 I/O
- 可指定发送模式、接收模式以及发送/接收模式。

备注 n = 0 至 4

18.3 配置

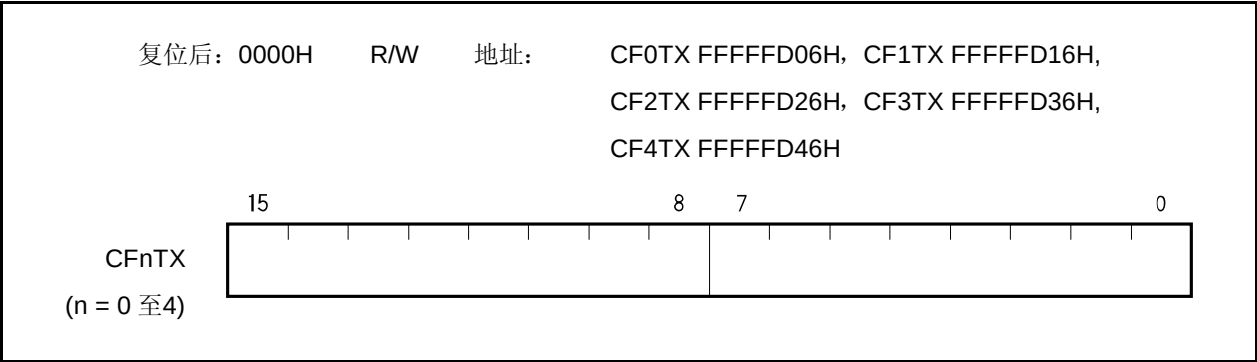
以下显示 CSIFn 的功能框图。

图 18-4. CSIFn 的功能框图



(2) CSIFn 发送数据寄存器(CFnRX)

CFnTX 寄存器为 16 位缓冲寄存器，用来写入 CSIFn 的发送数据。
该寄存器可按 16 位为单位进行读写。
在发送允许的情况下，通过向 CFnTX 寄存器写入数据启动发送操作。
如果传输数据长度为 8 位，本寄存器的低 8 位作为寄存器 CFnTXL，其只能以 8 位为单位进行读取。
复位后本寄存器设置为 0000H。



备注

通信起始状态如下所示。

发送模式 (CFnTXE 位= 1, CFnRXE 位= 0):

写入 CFnTX 寄存器

发送/接收模式(CFnTXE 位= 1, CFnRXE 位= 1):

写入 CFnTX 寄存器

接收模式(CFnTXE 位= 0, CFnRXE 位= 1):

由 CFnRX 寄存器读取

18.4 寄存器

以下寄存器用来控制 CSIFn。

- CSIFn 控制寄存器 0 (CFnCTL0)
- CSIFn 控制寄存器 1 (CFnCTL0)
- CSIFn 控制寄存器 2 (CFnCTL0)
- CSIFn 状态寄存器(CFnSTR)

(1) CSIFn 控制寄存器 0 (CFnCTL0)

CFnCTL0 寄存器控制 CSIFn 串行传输操作。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 01H。

(1/3)

复位后: 01H R/W 地址: CF0CTL0 FFFFFFFD00H, CF1CTL0 FFFFFFFD10H,
CF2CTL0 FFFFFFFD20H, CF3CTL0 FFFFFFFD30H,
CF4CTL0 FFFFFFFD40H

	<7>	<6>	<5>	<4>	3	2	1	<0>
CFnCTL0 (n = 0至4)	CFnPWR	CFnTXE ^注	CFnRXE ^注	CFnDIR ^注	0	0	CFnTMS ^注	CFnSCE

CFnPWR	禁止/允许CSIFn操作的说明
0	禁止操作并复位CFnSTR寄存器使能
1	使能CSIFn操作
• CFnPWR位控制CSIFn操作并使得内部电路复位	

CFnTXE ^注	禁止/允许发送操作的说明
0	禁止发送操作
1	允许发送操作
• 当CFnTXE位为0时SOFn输出低电平	

CFnRXE ^注	禁止/允许接收操作才说明
0	禁止接收操作
1	允许接收操作
• 即使发送指定的数据并更新接收数据 (CFnRX)，也不会输出接收完成中断，因为已经通过将CFnRXE位清为0禁止了接收操作。	

注 当 CFnPWR 位= 0 时，这些位仅可写入。而重写这些位的同时也可设置 CFnPWR 位=1。

注意事项 若要强制挂起发送/接收，将 CFnPWR 位清除为 0 而不是 CFnRXE 和 CFnTXE 位。
此时，时钟输出停止。

CFnDIR ^注	传输方向模式的说明(MSB/LSB)
0	MSB先行传输
1	LSB先行传输

CFnTMS ^注	传输模式说明
0	单次传输模式
1	连续传输模式

[单次传输模式中]
通信完成后会发生接收完成中断（INTCFnR）。
仅是发送使能 (CFnTXE位 = 1)，也不会产生发送使能中断（INTCFnT）。
如果在通信期间（CFnSTR.CFnTSF位= 1）写入下一个待发送的数据，
则将其忽略并开始下一次通信。同样，如果设置了只收通信（CFnTXE位 = 0，
CFnRXE位 = 1），则即使在通信期间读取了接收数据（CFnSTR.CFnTSF 位 = 1），
也并不开始下一次通信操作。

[连续传输模式中]
通过在通信期间（CFnSTR.CFnTSF位= 1）写入下一个要发送的数据使能连续发送操作。
发送使能中断（INTCFnT）产生后，允许写入下一个要发送的数据。
如果在连续传输模式设置了只收通信模式（CFnTXE位 = 0，CFnRXE位 = 1），
则不管是否读取了CFnRX寄存器的内容，都会在接收完成中断（INTCFnR）之
后立即开始下一次接收。
因此，会立即从CFnRX寄存器读取接收数据。如果该读取操作延迟，则会发生超限
错误（CFnOVE位 = 1）。

注 当 CFnPWR 位= 0 时，这些位仅可写入。而重写这些位的同时也可设置 CFnPWR 位 =1。

CFnSCE	禁止/允许传输开始的说明
0	通信开始触发无效
1	通信开始触发有效

• 主机模式
 该位允许或禁止通信开始触发。

(a) 在单次发送模式或发送/接收模式，或连续发送模式，或连续发送/接收模式
 当CFnSCE位为1时可通过将数据写入CFnTX寄存器启动通信操作。
 设置CFnSCE位为1。

(b) 单次接收模式
 禁止在读取最末接收数据之前通过将CFnSCE位清0而开始下一次接收操作，
 因为已经通过读取接收数据（CFnRX寄存器）^{注1}开始了一个接收操作。

(c) 连续接收模式
 在最末数据接收完成之前通过清除CFnSCE位为一个通信时钟，以禁止最末数据
 接收之后^{注2}开始接收操作。

• 从机模式
 该位允许或禁止通信开始触发。
 设置CFnSCE位为1。

[CFnSCE位的用法]

• 单次接收模式中
 <1> 通过INTCFnR中断服务完成最末数据的接收时，在读取CFnRX寄存器
 之前将CFnSCE位清为0。
 <2> 确认CFnSTR.CFnTSF位= 0之后，将CFnRXE位清为0以禁止接收。
 为了连续接收，需将CFnSCE位设为1，以通过CFnRX寄存器的伪读取
 开始下一次接收。

• 连续接收模式中
 <1> 最末数据接收期间通过INTCFnR中断服务将CFnSCE位清为0。
 <2> 读取CFnRX寄存器。
 <3> 响应CFnTIR中断之后，通过读取CFnRX寄存器来读取最末接收的数据。
 <4> 确认CFnSTR.CFnTSF位= 0之后，将CFnRXE位清为0，以禁止接收。
 为了继续接收，将CFnSCE位设为1，以通过伪读取CFnRX寄存器等待下
 一次接收。

- 注**
1. 如果当其值为1时读取CFnSCE位，则开始下一次通信操作。
 2. 最末数据接收结束之前一个通信时钟，CFnSCE位不清为0，下一次通信操作自动开始。

注意事项 一定要将位3和2清为“0”。

(2) CSIFn 控制寄存器 1 (CFnCTL1)

CFnCTL 为 8 位寄存器，控制 CSIFn 串行传输操作。

本寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

注意事项 CFnCTL1 寄存器仅当 CFnCTL0.CFnPWR 位=0 时才可以被改写。

复位后: 00H R/W 地址: CF0CTL1 FFFFFFFD01H, CF1CTL1 FFFFFFFD11H,
CF2CTL1 FFFFFFFD21H, CF3CTL1 FFFFFFFD31H,
CF4CTL1 FFFFFFFD41H

	7	6	5	4	3	2	1	0
CFnCTL1	0	0	0	CFnCKP	CFnDAP	CFnCKS2	CFnCKS1	CFnCKS0

(n = 0至4)

	CFnCKP	CFnDAP	与 SCKFn 相关的数据接收/发送时序的说明
通信类型 1	0	0	
通信类型 2	0	1	
通信类型 3	1	0	
通信类型 4	1	1	

CFnCKS2	CFnCKS1	CFnCKS0	通信时钟 (f _{CCLK})		模式
			n = 0 to 2, 4 ^{注1}	n = 3 ^{注2}	
0	0	0	f _{xx} /3	f _{xx} /2	主机模式
0	0	1	f _{xx} /4	f _{xx} /4	主机模式
0	1	0	f _{xx} /6	f _{xx} /8	主机模式
0	1	1	f _{xx} /8	f _{xx} /16	主机模式
1	0	0	f _{xx} /32	f _{xx} /32	主机模式
1	0	1	f _{xx} /64	f _{xx} /64	主机模式
1	1	0	f _{BRGm}		主机模式
1	1	1	外部时钟 (SCKFn)		从机模式

- 注**
1. 设置通信时钟 (f_{CCLK}) 为 8 MHz 或更低。
 2. 设置通信时钟 (f_{CCLK}) 为 12 MHz 或更低。

备注

当 n = 0, 1、m = 1 时
 当 n = 2, 3、m = 2 时
 当 n = 4, m = 3 时
 关于 f_{BRGm} 详细情况, 参见 18.8 波特率发生器。

<R>

(3) CSIFn 控制寄存器 2 (CFnCTL2)

CFnCTL2 为 8 位寄存器，用来控制 CSIFn 串行传输的位数目。

该寄存器可按 8 位为单位进行读写。

复位后本寄存器被设置为 00H。

注意事项 CFnCTL2 寄存器仅当 CFnCTL0.CFnPWR 位=0 或 CFnTXE 和 CFnRXE 位均为 0 时才可以写入。

复位后: 00H R/W 地址: CF0CTL2 FFFFFFFD02H, CF1CTL2 FFFFFFFD12H,
CF2CTL2 FFFFFFFD22H, CF3CTL2 FFFFFFFD32H,
CF4CTL2 FFFFFFFD42H

	7	6	5	4	3	2	1	0
CFnCTL2	0	0	0	0	CFnCL3	CFnCL2	CFnCL1	CFnCL0

(n = 0至4)

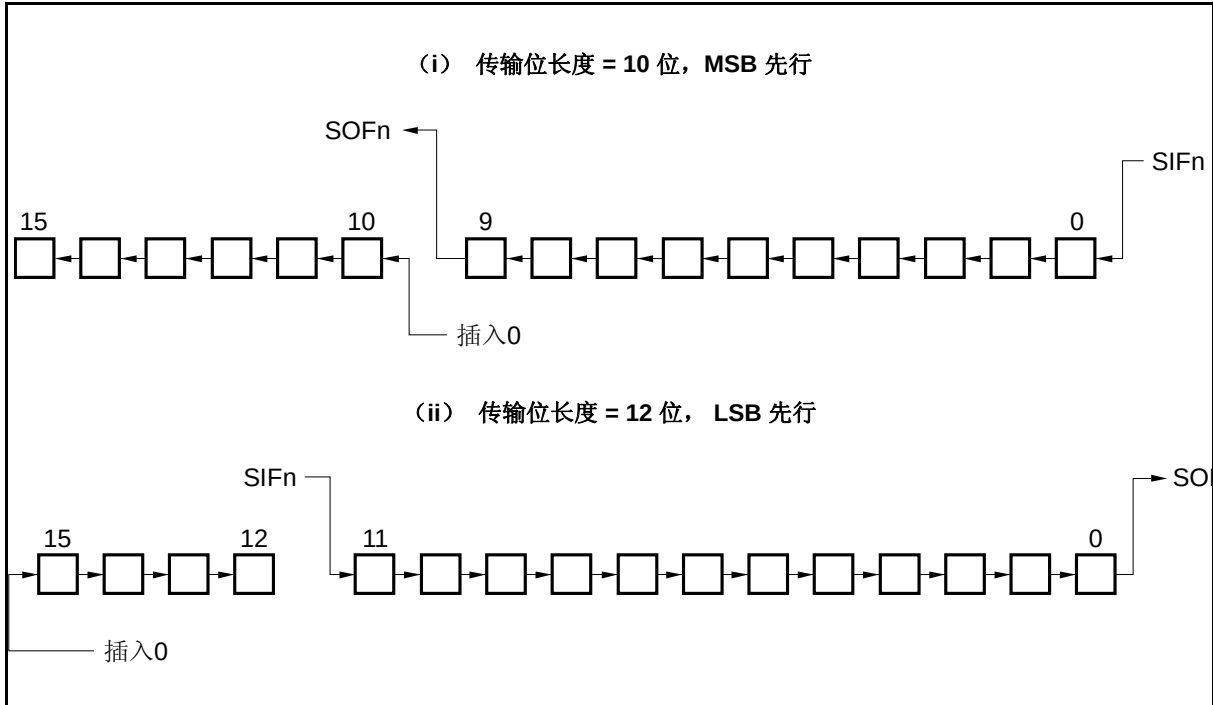
CFnCL3	CFnCL2	CFnCL1	CFnCL0	连续寄存器位长度
0	0	0	0	8位
0	0	0	1	9位
0	0	1	0	10位
0	0	1	1	11位
0	1	0	0	12位
0	1	0	1	13位
0	1	1	0	14位
0	1	1	1	15位
1	×	×	×	16位

- 备注**
1. 如果传输位数不是 8 或 16，请准备并使用填充数据，从 CF0TX 和 CF0RX 寄存器的 LSB 起进行填充。
 2. ×: 忽略

(a) 传输数据长度改变功能

CSIFn 传输数据长度可以使用 CFnCTL2.CFnCL3 至 CFnCTL2.CFnCL0 位，在 8 位和 16 位之间以 1 位为单位进行设置。

当传输位长度设置为非 16 位时，将 CF0TX 或 CF0RX 数据设置为由 LSB 开始，不管传输起始位是 MSB 还是 LSB。没有使用的高位可以设置为任意数据，但是，串行传输之后所接收的相应数据变为 0。



(4) CSIFn 状态寄存器(CFnSTR)

CFnSTR 为一个 8 位寄存器，用来显示 CSIFn 的状态。

本寄存器可以以 8 位或 16 位为单位进行读取或写入，但是，CFnTSF 标志是只读的。

复位后本寄存器被设置为 00H。

除复位输入之外，CFnSTR 可以通过清除 CFnCTL0.CFnPWR 位（为 0）来初始化。

复位后: 00H R/W 地址: CF0STR FFFFD03H, CF1STR FFFFD13H,
CF2STR FFFFD23H, CF3STR FFFFD33H,
CF4STR FFFFD43H

	<7>	6	5	4	3	2	1	<0>
CFnSTR (n = 0至4)	CFnTSF	0	0	0	0	0	0	CFnOVE

CFnTSF	通信状态标志
0	通信停止
1	通信中
- 发送期间，当CFnTX寄存器的数据准备好时设置该寄存器，而接收期间，当进行伪读取CFnRX寄存器时设置该寄存器。 - 传输结束时，在时钟的后边沿将该标志清0。	

CFnOVE	超限错误标志
0	无超限
1	超限
- 在CPU没有读取接收缓冲器值而完成下一个接收的情况下，基于接收操作完成会发生超限错误。 - CFnOVE 标志展示此种情况下超限错误发生的状态。 - CFnOVE位在单次传输模式中同样有效。因此，仅使用发送时，需注意以下各项。 - 不检查 CFnOVE 标志。 - 即使不需读取接收数据也会读取该位。 - 通过将0写入CFnOVE标志将其清除。向其写入1不能对其进行设置。	

18.5 中断请求信号

CSIFn 可产生以下两种中断请求信号：

- 接收完成中断请求信号（INTCFnR）
- 发送允许中断请求信号（INTCFnT）

这两种中断请求信号中，缺省为接收完成中断请求信号有较高优先级，而发送允许中断请求信号优先级较低。

表 18-2. 中断与其默认优先级

中断	优先级
接收完成	高
发送允许	低

(1) 接收完成中断请求信号（INTCFnR）

接收数据传输至 CFnRX 寄存器而接收使能时，产生接收完成中断请求信号。
如果发生溢出错误也同样产生该中断请求信号。
响应接收完成中断请求信号并读取数据时，读取 CFnSTR 寄存器并检查接收结果是否错误。
单次传输模式中，即使只进行发送，INTCFnR 中断请求信号的产生基于发送的完成。

(2) 发送允许中断请求信号（INTCFnT）

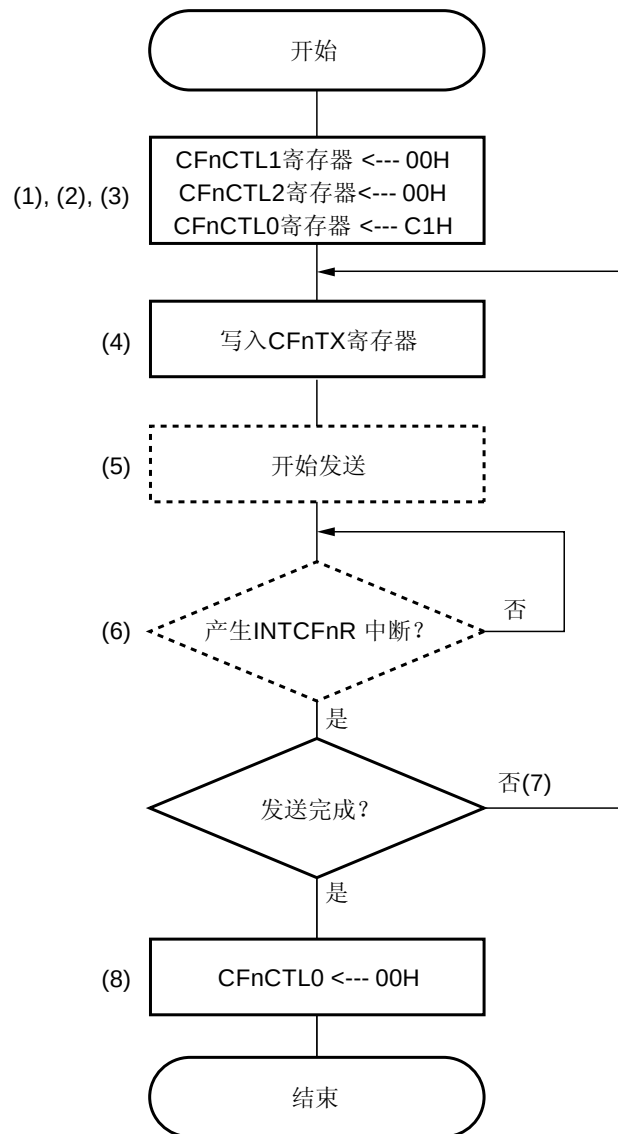
连续发送或连续发送/接收模式中，发送数据从 CFnTX 寄存器移出，并且一旦允许写入 CFnTX 寄存器，即产生发送允许中断请求信号。
在单次发送和单次发送/接收模式中，不产生 INTCFnT 中断。

18.6 操作

18.6.1 单次传输模式（主机模式，发送模式）

MSB 先行（CFnCTL0.CFnDIR 位= 0），通信类型 1（CFnCTL1.CFnCKP 和 CFnCTL1.CFnDAP 位= 00），通信时钟（f_{CCLK}）= f_{xx}/2 或 f_{xx}/3（CFnCTL1.CFnCKS2 至 CFnCTL1.CFnCKS0 位 = 000），传输数据长度= 8 位（CFnCTL2.CFnCL3 至 CFnCTL2.CFnCL0 位= 0000）。

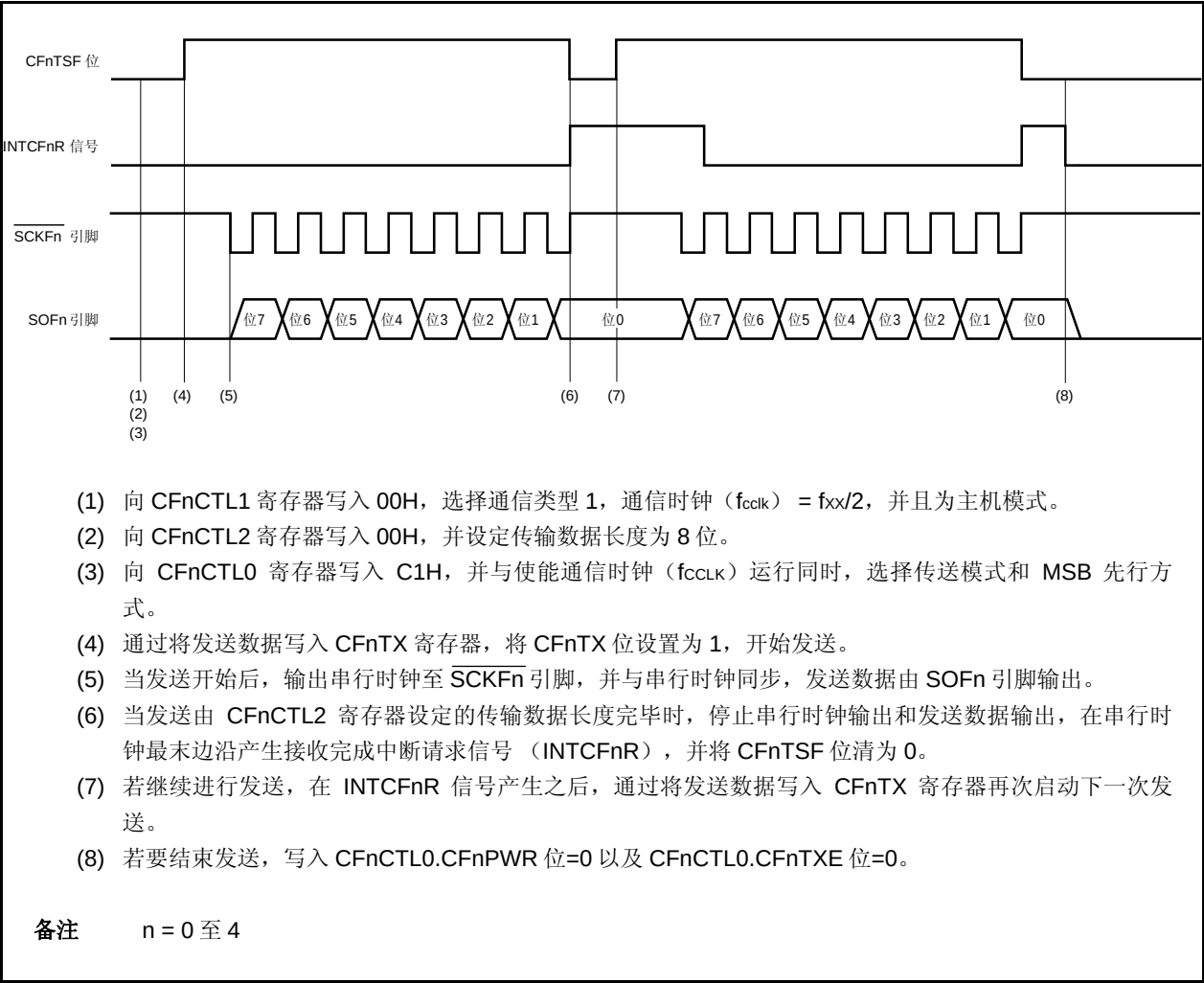
(1) 操作流程



备注

1. 虚线部分表示硬件处理过程。
2. 图中的数字与（2）**操作时序**中的处理数字相对应。
3. n = 0 至 4

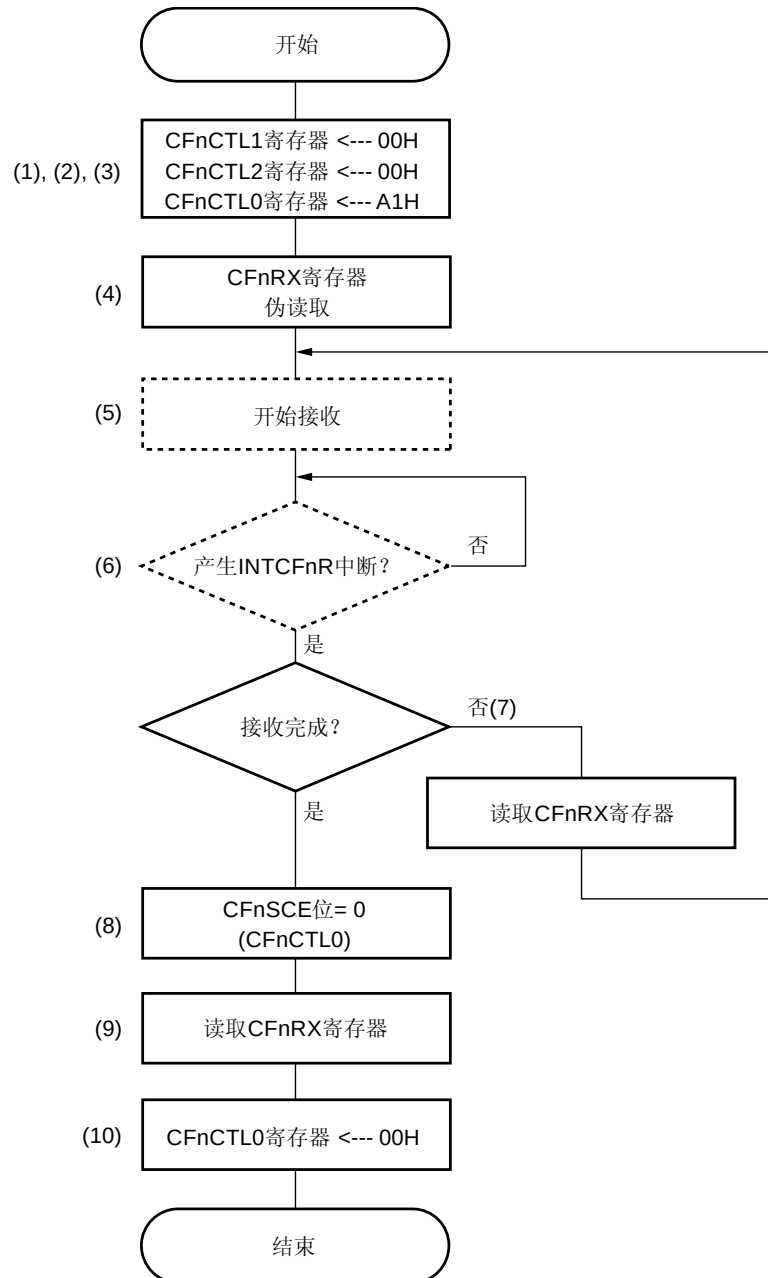
(2) 操作时序



18.6.2 单次传输模式（主机模式，接收模式）

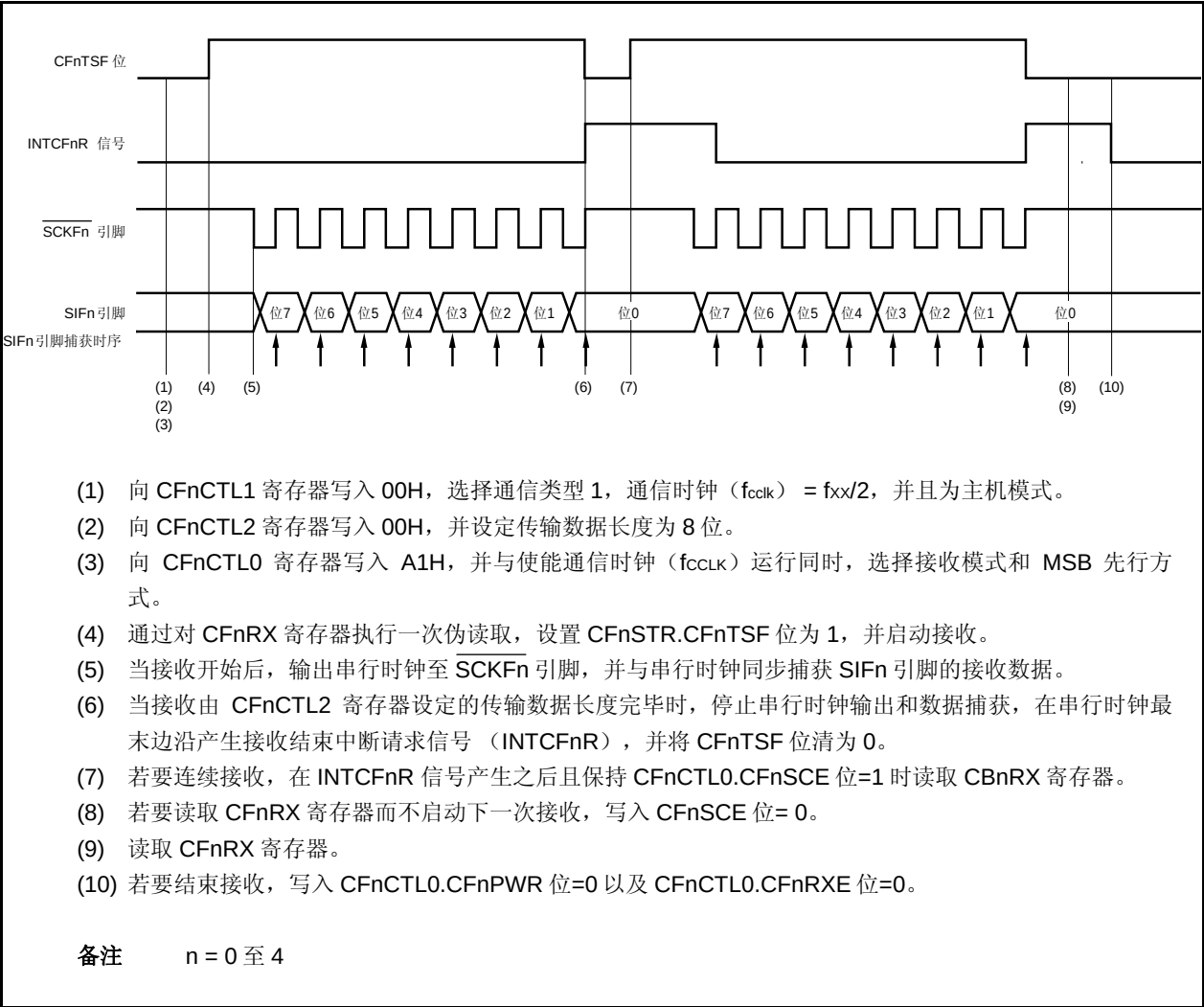
MSB 先行（CFnCTL0.CFnDIR 位= 0），通信类型 1（CFnCTL1.CFnCKP 和 CFnCTL1.CFnDAP 位= 00），通信时钟（f_{CCLK}）= f_{xx}/2 或 f_{xx}/3（CFnCTL1.CFnCKS2 至 CFnCTL1.CFnCKS0 位 = 000），传输数据长度= 8 位（CFnCTL2.CFnCL3 至 CFnCTL2.CFnCL0 位= 0000）。

(1) 操作流程



- 备注
1. 虚线部分表示硬件处理过程。
 2. 图中的数字与（2）操作时序中的处理数字相对应。
 3. n = 0 至 4

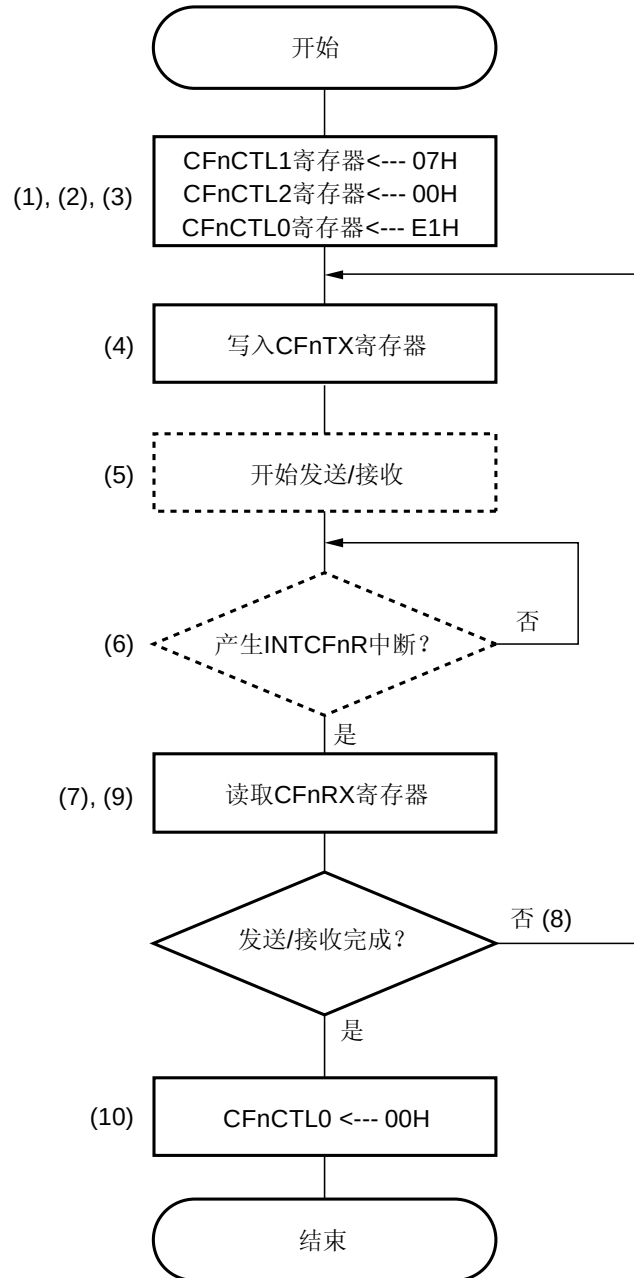
(2) 操作时序



18.6.3 单次传输模式（主机模式，发送/接收模式）

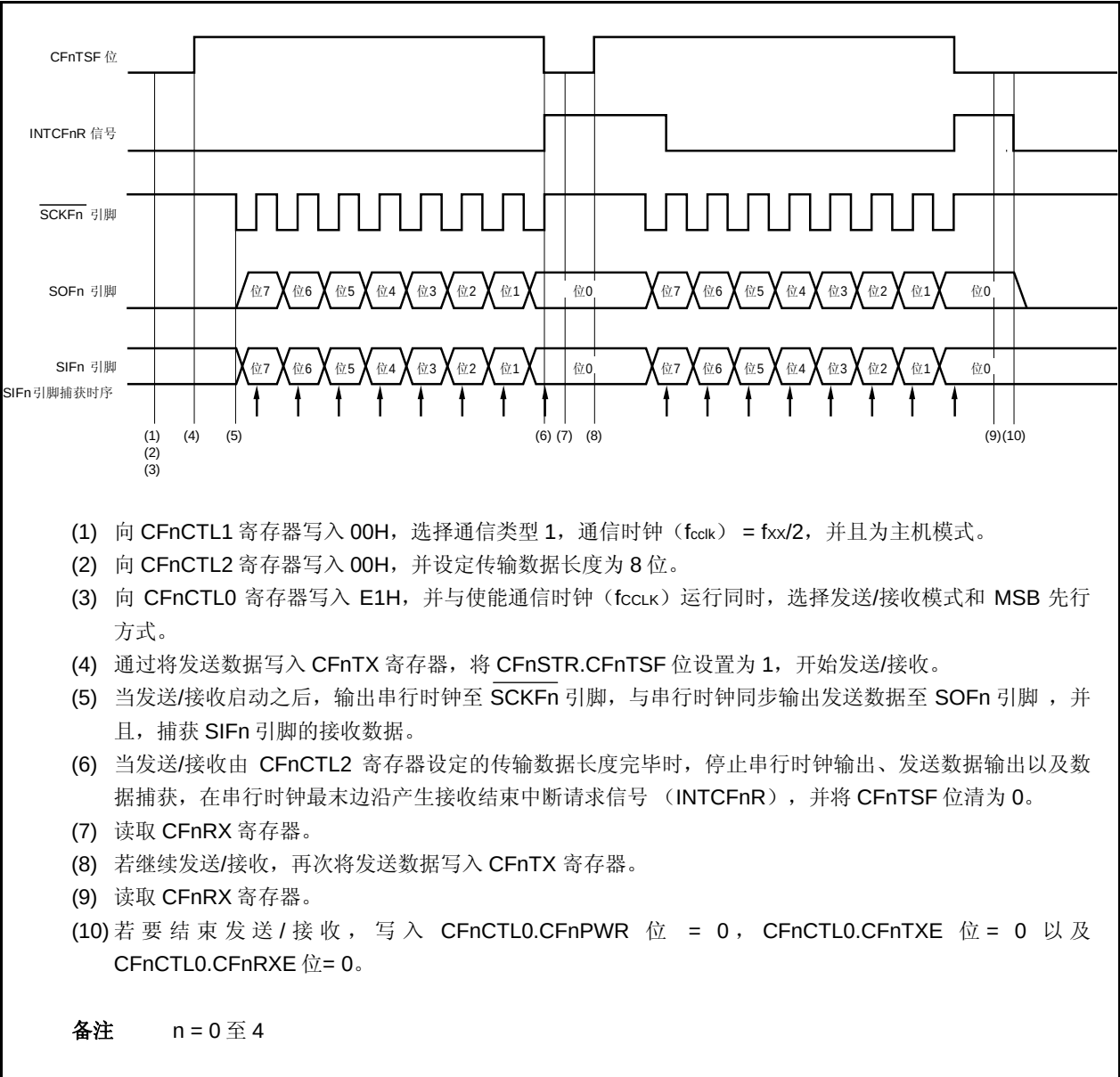
MSB 先行（CFnCTL0.CFnDIR 位= 0），通信类型 1（CFnCTL1.CFnCKP 和 CFnCTL1.CFnDAP 位= 00），通信时钟（f_{CCLK}）= f_{xx}/2 或 f_{xx}/3（CFnCTL1.CFnCKS2 至 CFnCTL1.CFnCKS0 位 = 000），传输数据长度= 8 位（CFnCTL2.CFnCL3 至 CFnCTL2.CFnCL0 位= 0000）。

(1) 操作流程



- 备注
1. 虚线部分表示硬件处理过程。
 2. 图中的数字与（2）**操作时序**中的处理数字相对应。
 3. n = 0 至 4

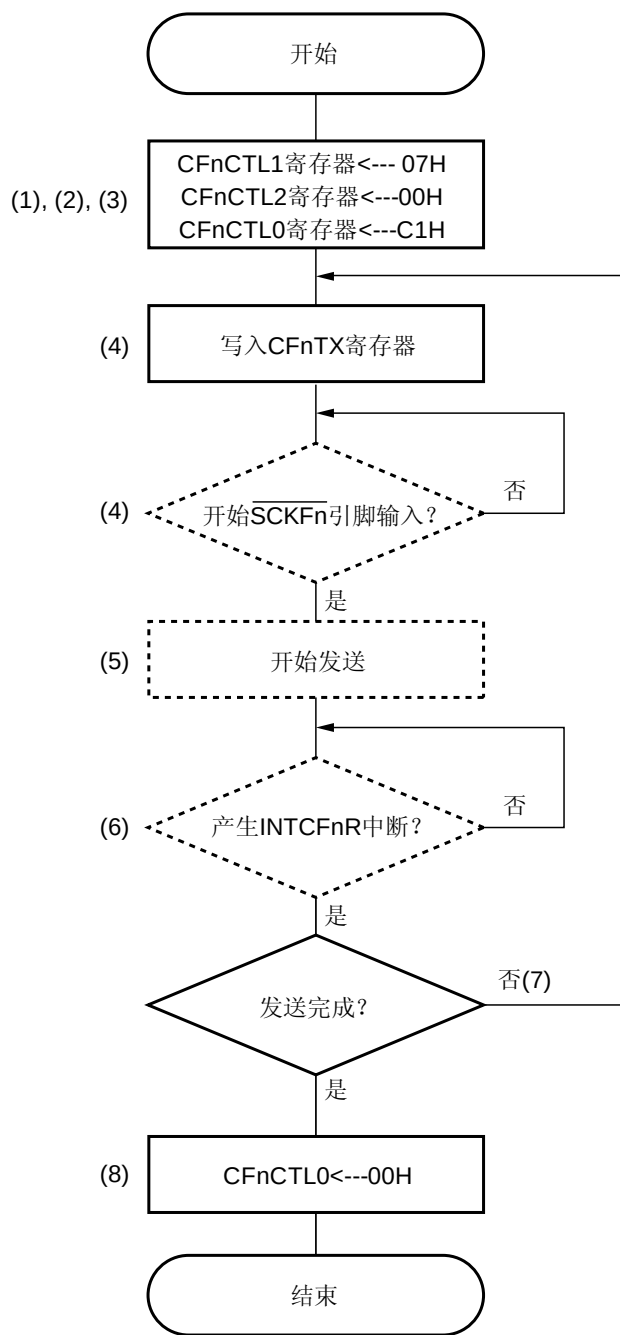
(2) 操作时序



18.6.4 单次传输模式（从机模式，发送模式）

MSB 先行（CFnCTL0.CFnDIR 位= 0），通信类型 1（CFnCTL1.CFnCKP 和 CFnCTL1.CFnDAP 位= 00），通信时钟（f_{CCLK}）=外部时钟（ $\overline{\text{SCKFn}}$ ）（CFnCTL1.CFnCKS2 至 CFnCTL1.CFnCKS0 位 = 111），发送数据长度= 8 位（CFnCTL2.CFnCL3 至 CFnCTL2.CFnCL0 位= 0000）

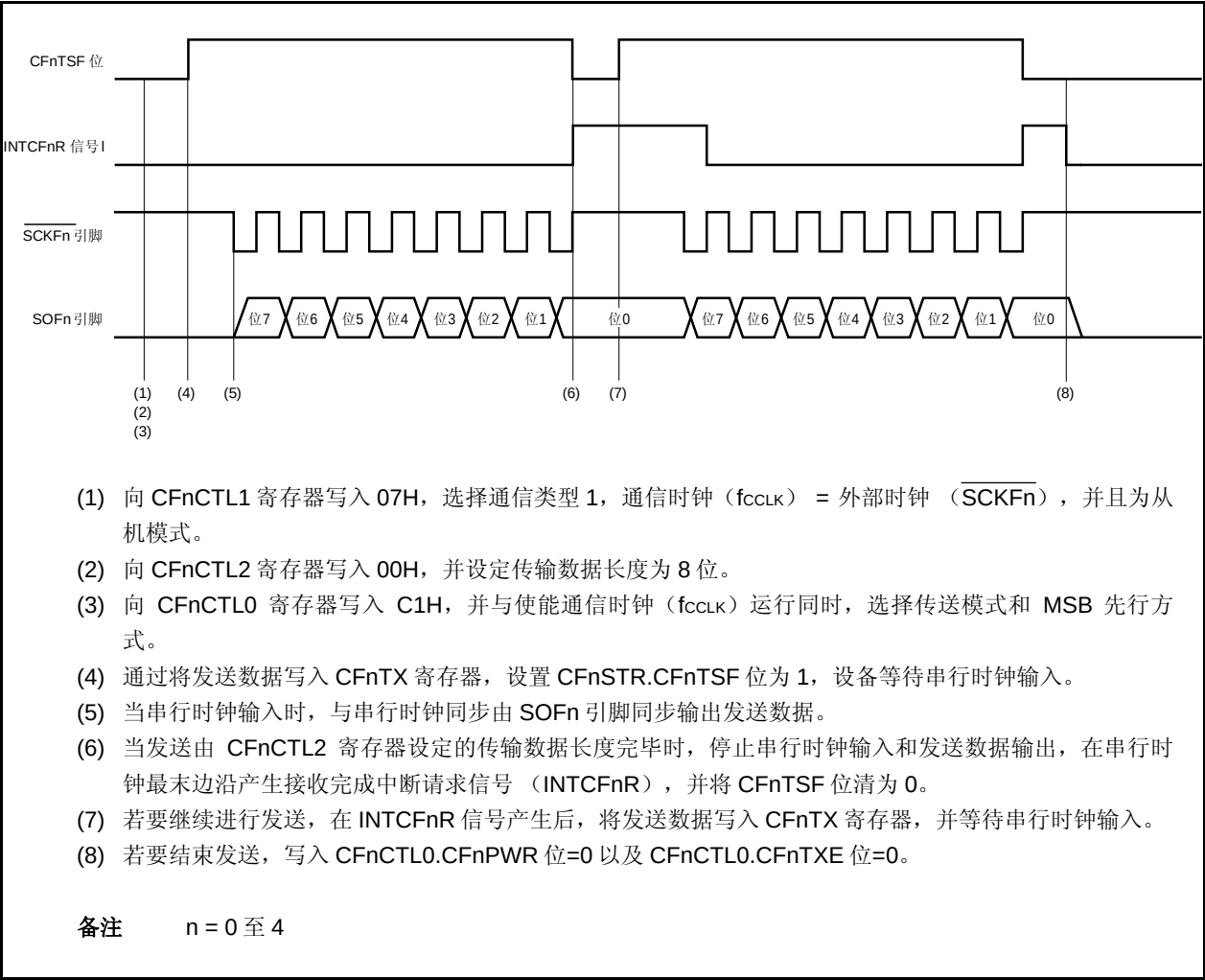
(1) 操作流程



备注

1. 虚线部分表示硬件处理过程。
2. 图中的数字与（2）操作时序中的处理数字相对应。
3. n = 0 至 4

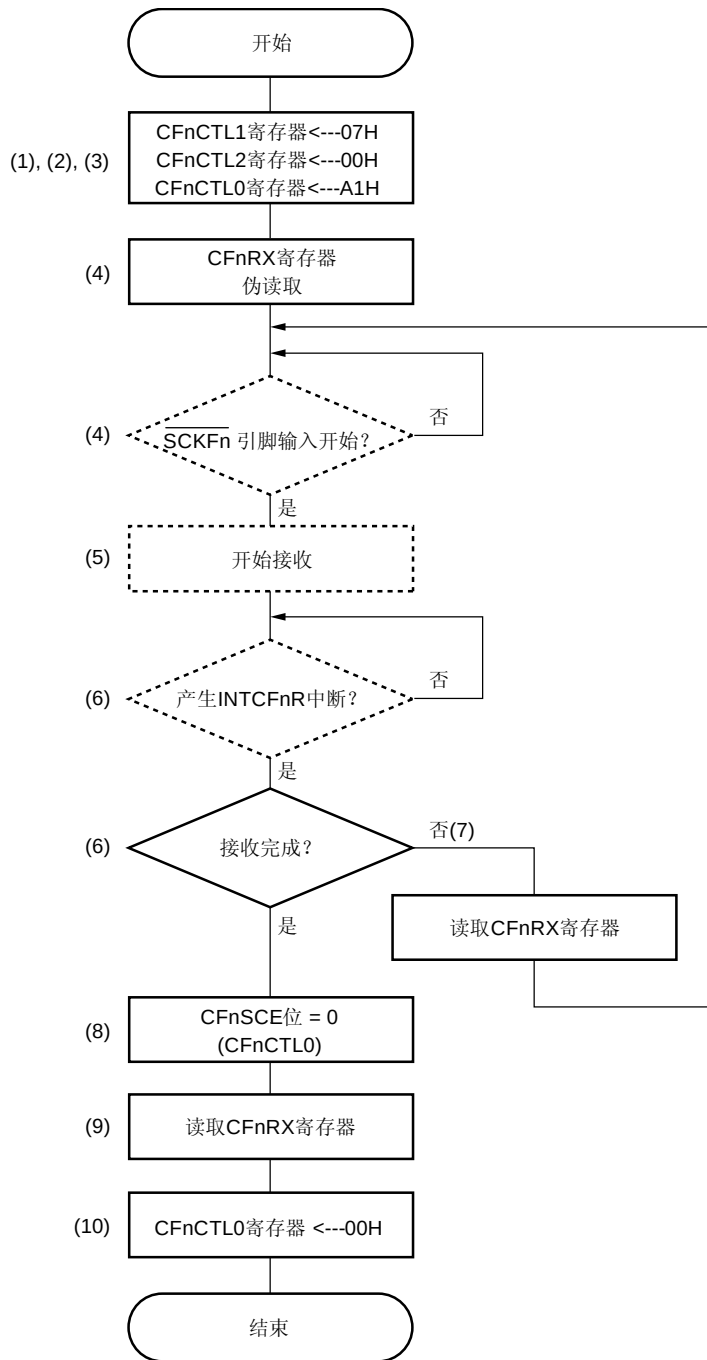
(2) 操作时序



18.6.5 单次传输模式（从机模式，接收模式）

MSB 先行（CFnCTL0.CFnDIR 位= 0），通信类型 1（CFnCTL1.CFnCKP 和 CFnCTL1.CFnDAP 位= 00），通信时钟（f_{CCLK}）=外部时钟（SCKFn）（CFnCTL1.CFnCKS2 至 CFnCTL1.CFnCKS0 位 = 111），发送数据长度= 8 位（CFnCTL2.CFnCL3 至 CFnCTL2.CFnCL0 位= 0000）

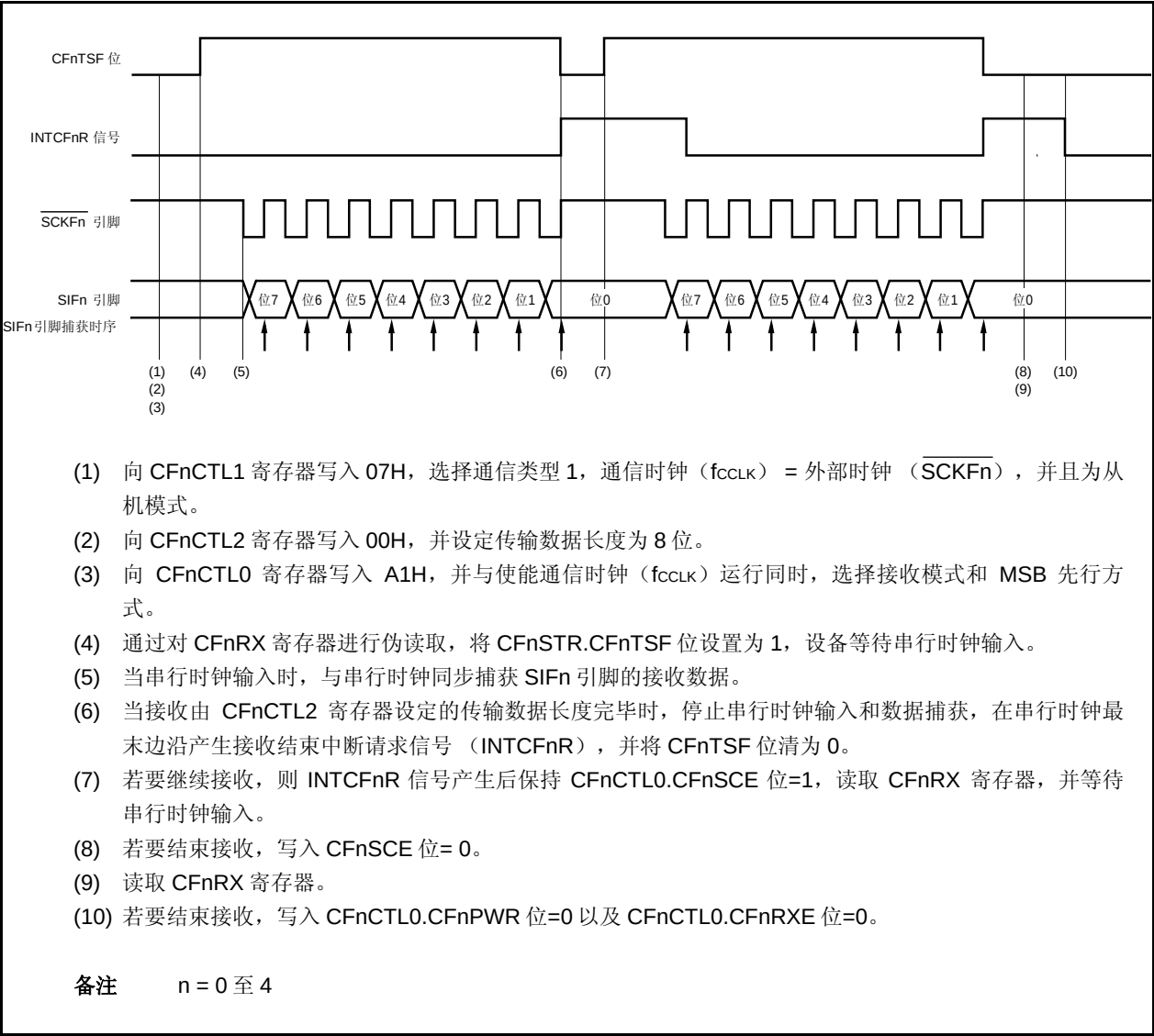
(1) 操作流程



备注

1. 虚线部分表示硬件处理过程。
2. 图中的数字与（2）**操作时序**中的处理数字相对应。
3. n = 0 至 4

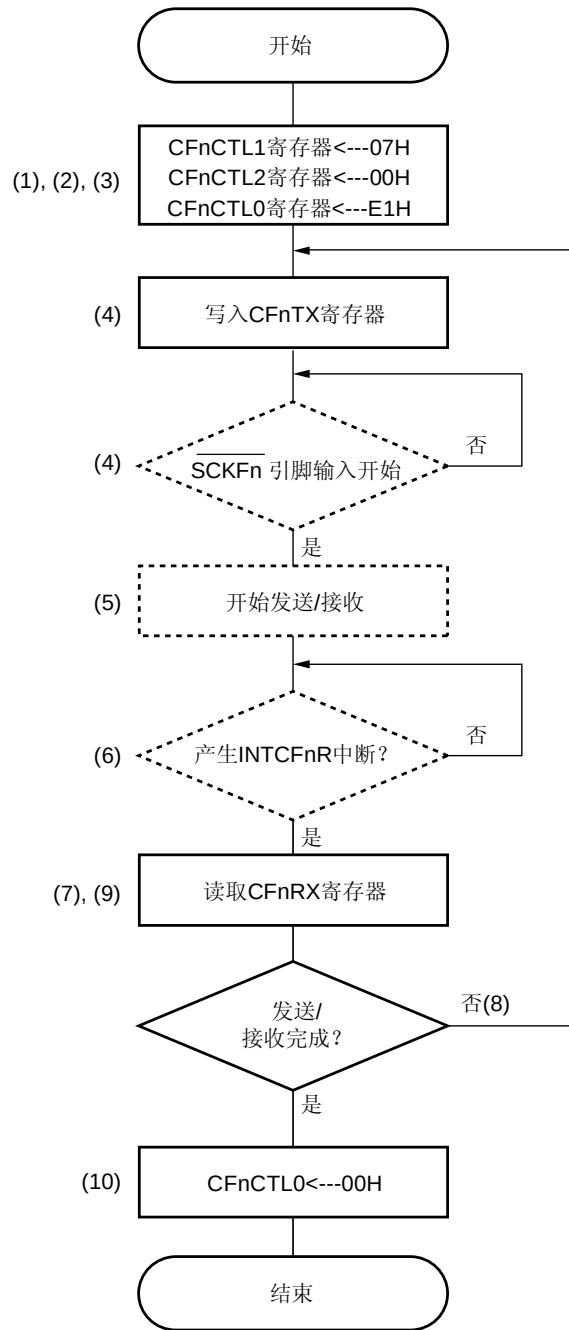
(2) 操作时序



18.6.6 单次传输模式（从机模式，发送/接收模式）

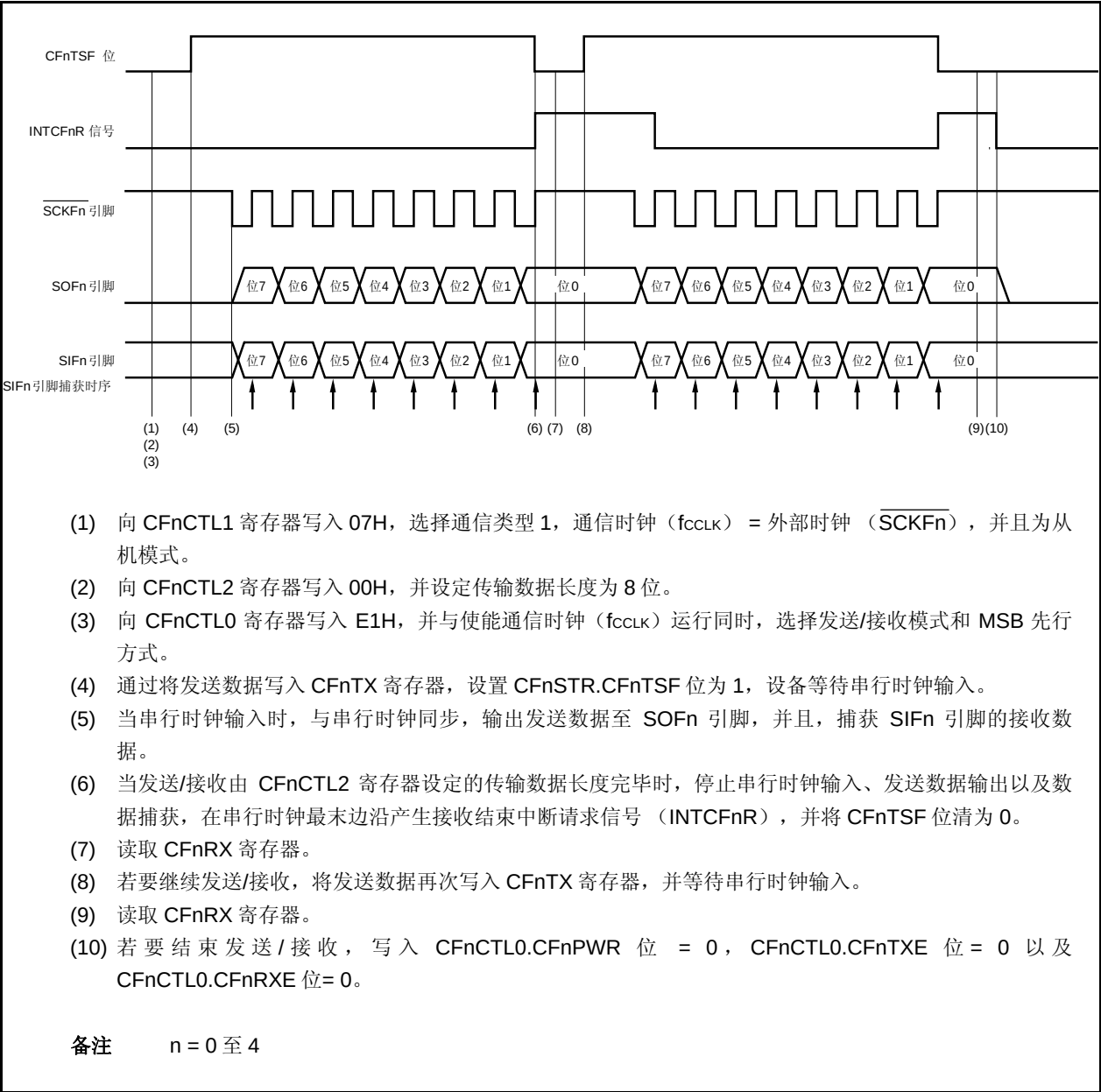
MSB 先行（CFnCTL0.CFnDIR 位= 0），通信类型 1（CFnCTL1.CFnCKP 和 CFnCTL1.CFnDAP 位= 00），通信时钟（f_{CCLK}）=外部时钟（ $\overline{\text{SCKFn}}$ ）（CFnCTL1.CFnCKS2 至 CFnCTL1.CFnCKS0 位 = 111），发送数据长度= 8 位（CFnCTL2.CFnCL3 至 CFnCTL2.CFnCL0 位= 0000）

(1) 操作流程



- 备注
1. 虚线部分表示硬件处理过程。
 2. 图中的数字与（2）操作时序中的处理数字相对应。
 3. n = 0 至 4

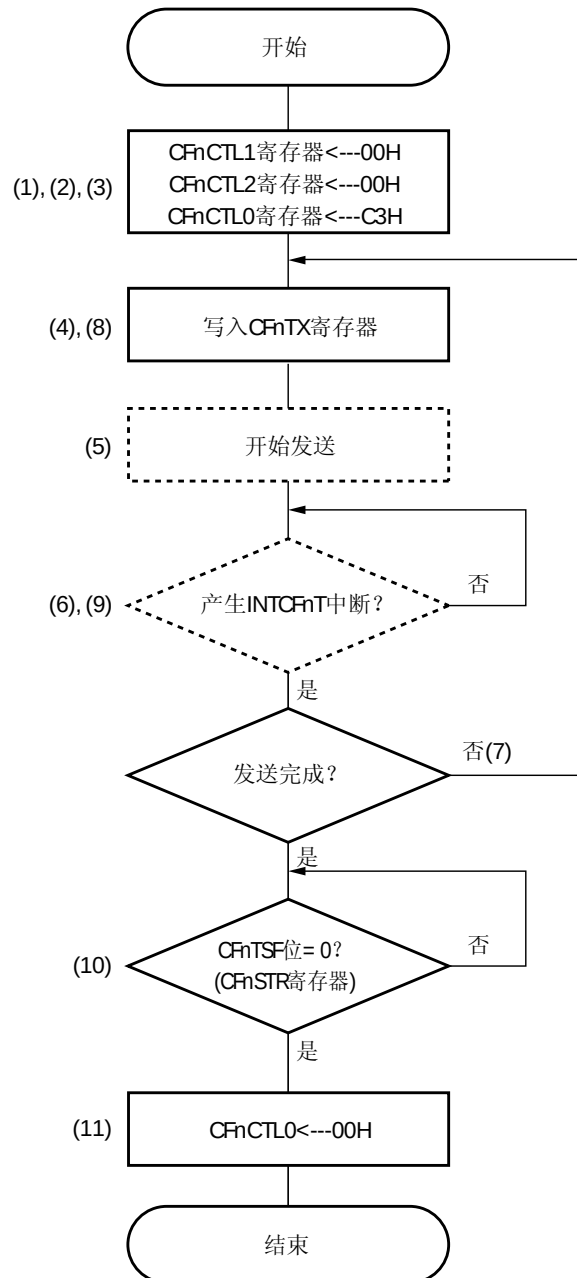
(2) 操作时序



18.6.7 连续传输模式（主机模式，发送模式）

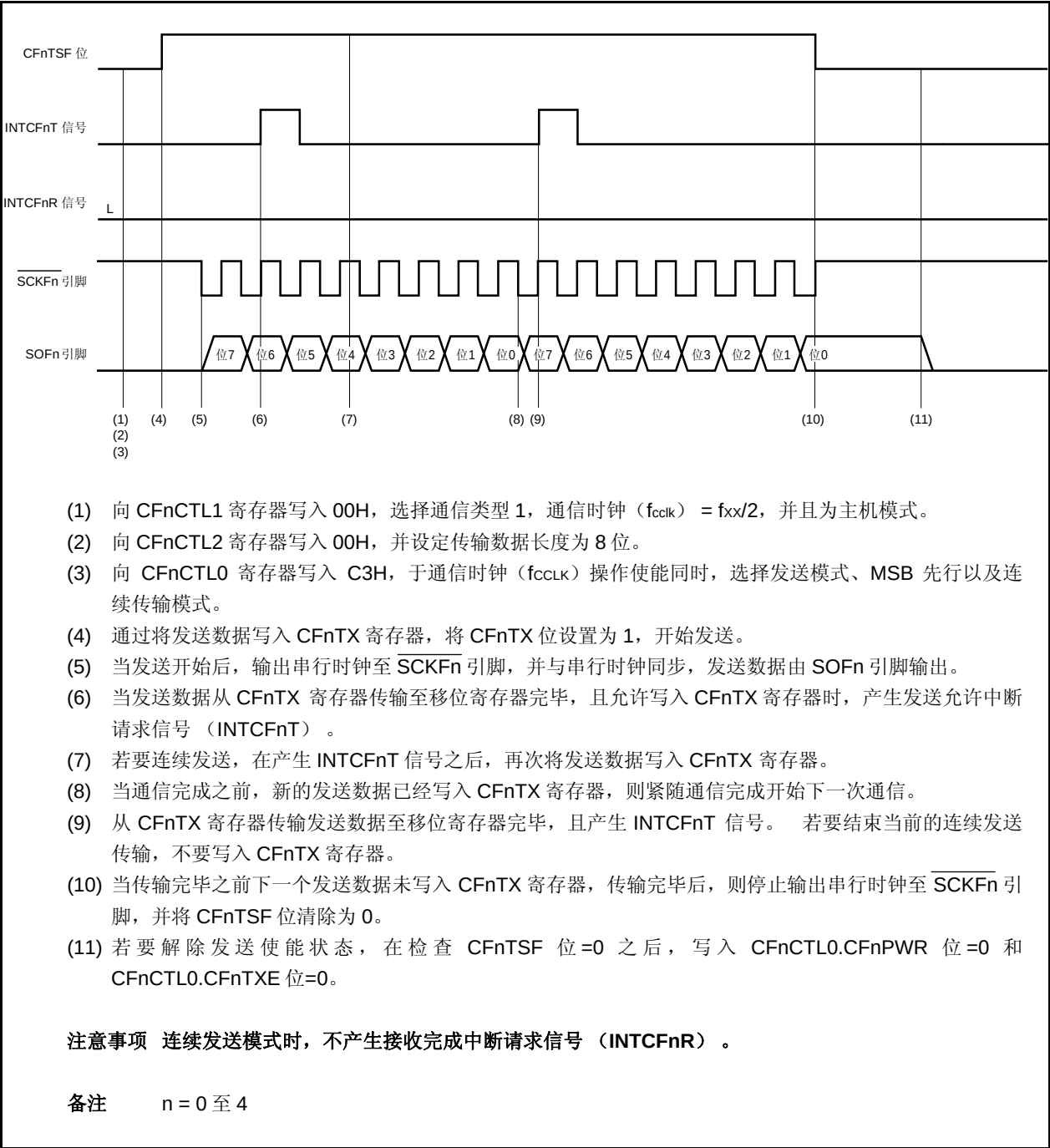
MSB 先行（CFnCTL0.CFnDIR 位= 0），通信类型 1（CFnCTL1.CFnCKP 和 CFnCTL1.CFnDAP 位= 00），通信时钟（f_{CCLK}）= f_{xx}/2 或 f_{xx}/3（CFnCTL1.CFnCKS2 至 CFnCTL1.CFnCKS0 位 = 000），传输数据长度= 8 位（CFnCTL2.CFnCL3 至 CFnCTL2.CFnCL0 位= 0000）。

(1) 操作流程



- 备注
1. 虚线部分表示硬件处理过程。
 2. 图中的数字与（2）操作时序中的处理数字相对应。
 3. n = 0 至 4

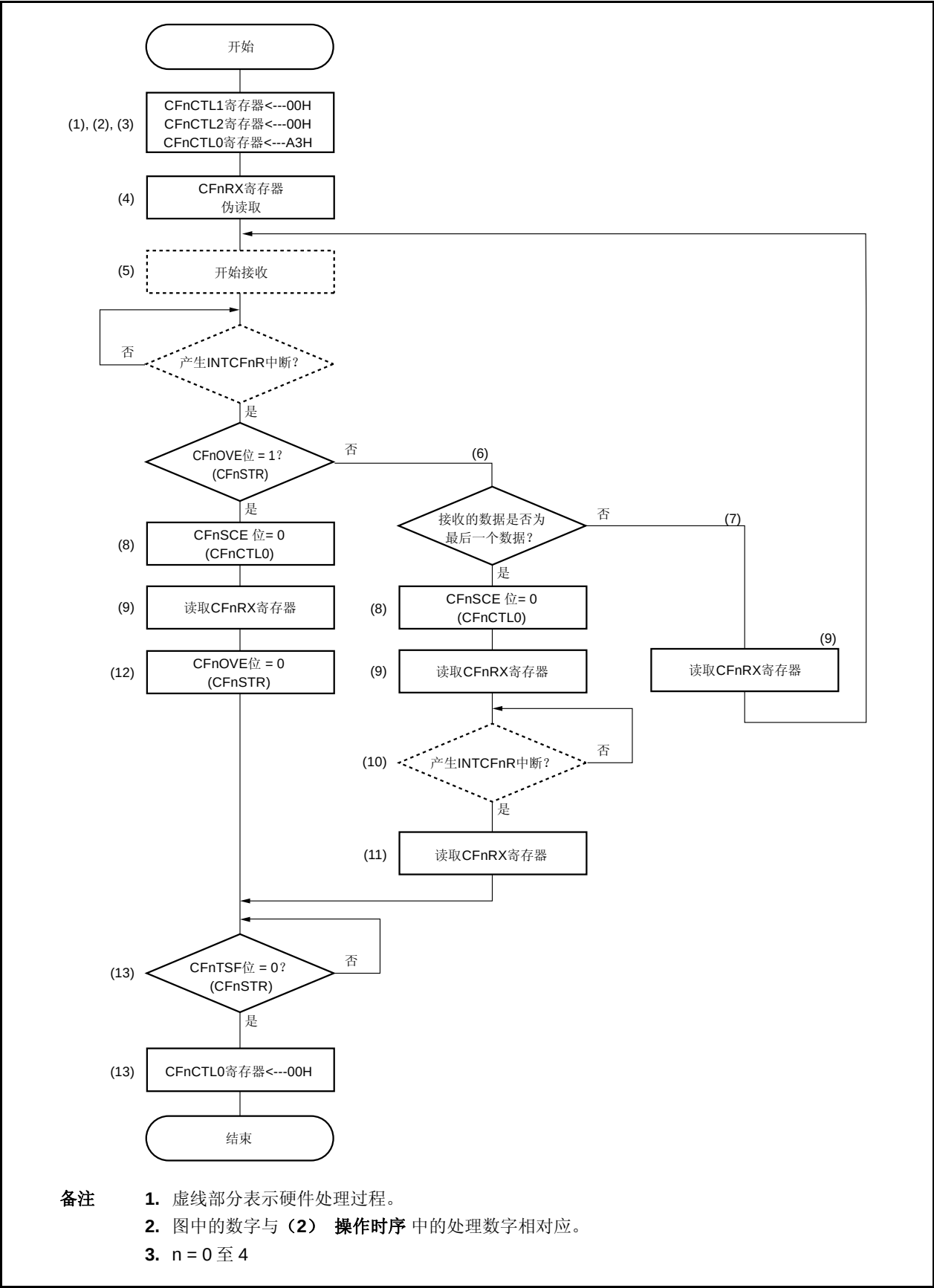
(2) 操作时序



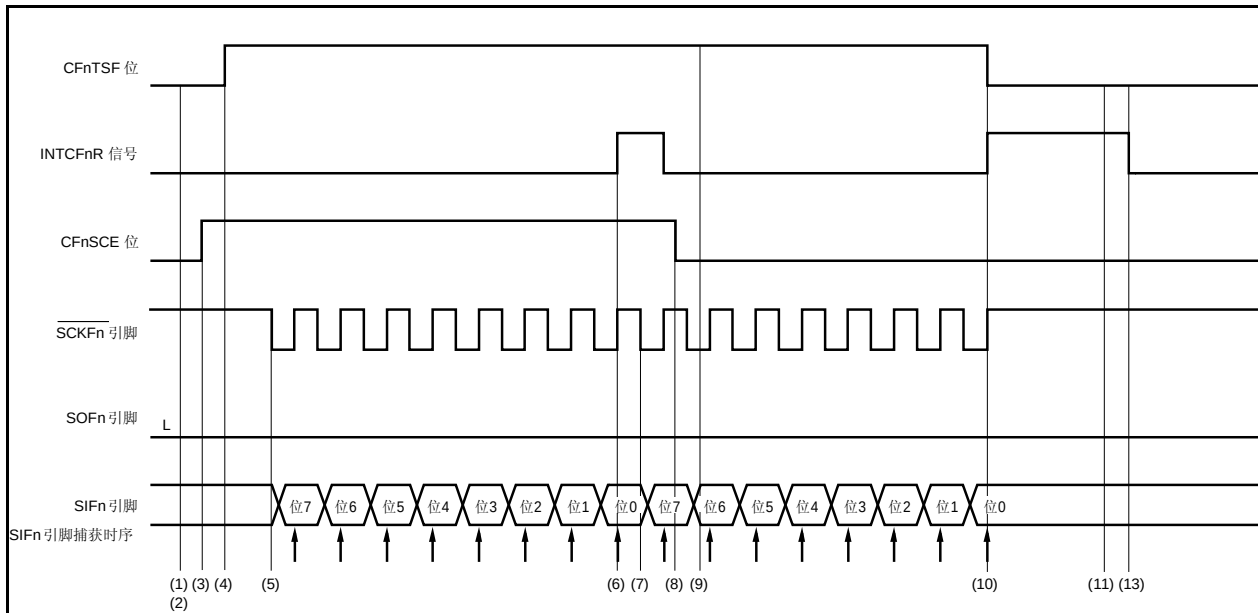
18.6.8 连续传输模式（主机模式，接收模式）

MSB 先行（CFnCTL0.CFnDIR 位= 0），通信类型 1（CFnCTL1.CFnCKP 和 CFnCTL1.CFnDAP 位= 00），通信时钟（f_{CCLK}）= f_{xx}/2 或 f_{xx}/3（CFnCTL1.CFnCKS2 至 CFnCTL1.CFnCKS0 位 = 000），传输数据长度= 8 位（CFnCTL2.CFnCL3 至 CFnCTL2.CFnCL0 位= 0000）。

(1) 操作流程



(2) 操作时序



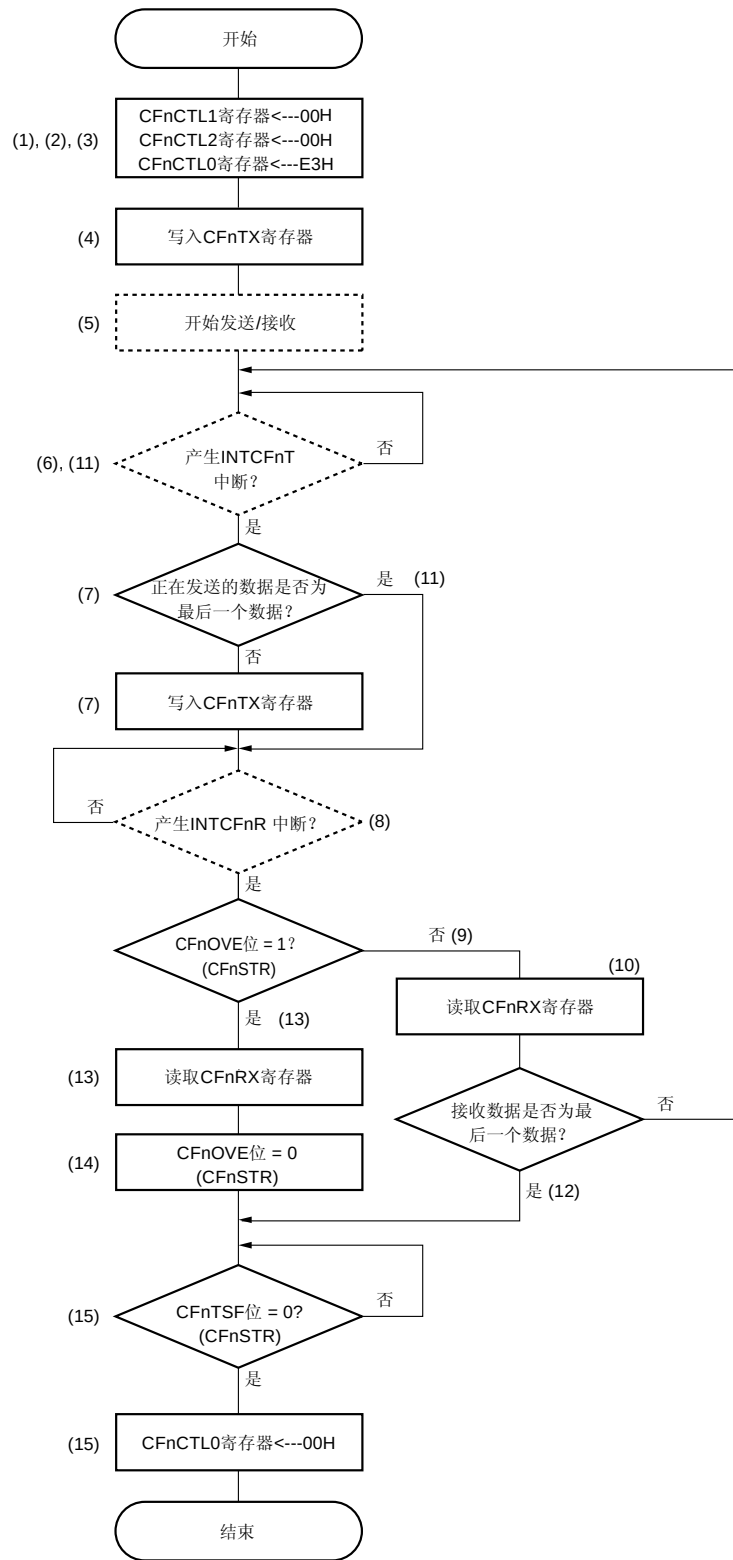
- (1) 向 CFnCTL1 寄存器写入 00H，选择通信类型 1，通信时钟 (f_{CLK}) = $f_{xx}/2$ ，并且为主机模式。
- (2) 向 CFnCTL2 寄存器写入 00H，并设定传输数据长度为 8 位。
- (3) 向 CFnCTL0 寄存器写入 A3H，于通信时钟 (f_{CLK}) 操作使能同时，选择接收模式、MSB 先行以及连续传输模式。
- (4) 通过对 CFnRX 寄存器执行一次伪读取，设置 CFnSTR.CFnTSF 位为 1，并启动接收。
- (5) 当接收开始后，输出串行时钟至 SCKFn 引脚，并与串行时钟同步捕获 SIFn 引脚的接收数据。
- (6) 当接收完毕之后，产生接收完成中断请求信号 (INTCFnR)，并且允许读取 CFnRX 寄存器。
- (7) 当通信结束时候 CFnCTL0.CFnSCE 位= 1，则紧随通信完毕开始下一次通信。
- (8) 若要结束当前连续接收传输，写入 CFnSCE 位= 0。
- (9) 读取 CFnRX 寄存器。
- (10) 当接收完毕时，产生 INTCFnR 信号，并且允许读取 CFnRX 寄存器。当通信结束之前设置 CFnSCE 位= 0，停止输出串行时钟至 SCKFn 引脚，并清除 CFnTSF 为 0，从而结束接收操作。
- (11) 读取 CFnRX 寄存器。
- (12) 如果发生超限错误，写入 CFnSTR.CFnOVE 位= 0，并且，并清除错误标志。
- (13) 若要解除接收使能状态，在检查 CFnTSF 位=0 之后，写入 CFnCTL0.CFnPWR 位=0 和 CFnCTL0.CFnRXE 位=0。

备注 n = 0 至 4

18.6.9 连续传输模式（主机模式，发送/接收模式）

MSB 先行（CFnCTL0.CFnDIR 位= 0），通信类型 1（CFnCTL1.CFnCKP 和 CFnCTL1.CFnDAP 位= 00），通信时钟（f_{CCLK}）= f_{xx}/2 或 f_{xx}/3（CFnCTL1.CFnCKS2 至 CFnCTL1.CFnCKS0 位 = 000），传输数据长度= 8 位（CFnCTL2.CFnCL3 至 CFnCTL2.CFnCL0 位= 0000）。

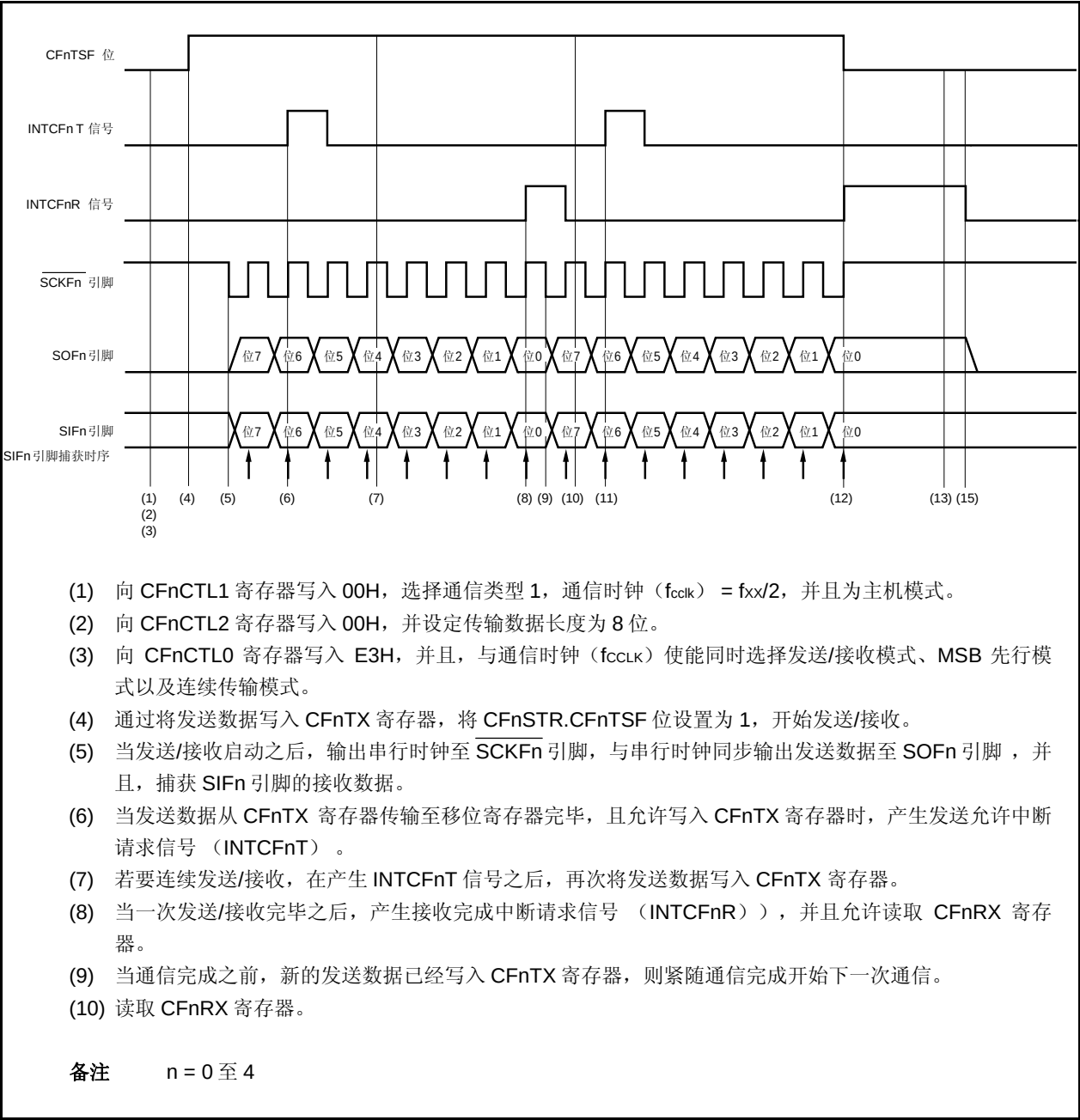
(1) 操作流程



- 备注
1. 虚线部分表示硬件处理过程。
 2. 图中的数字与 (2) 操作时序 中的处理数字相对应。
 3. n = 0 至 4

(2) 操作时序

(1/2)



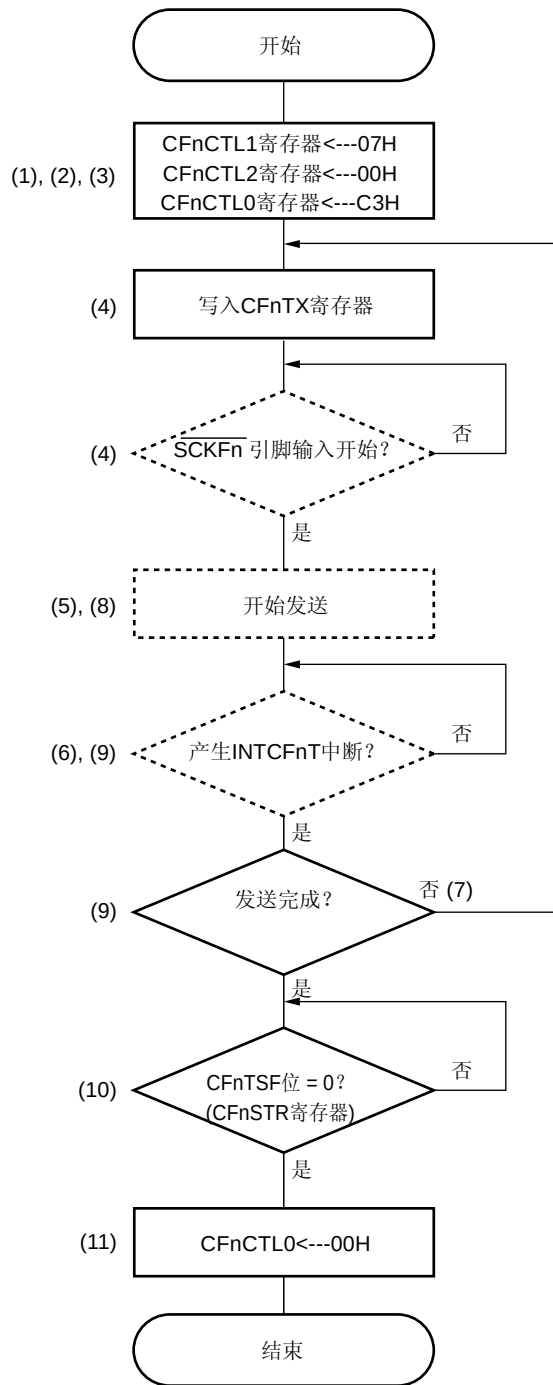
- (11) 从 CFnTX 寄存器传输发送数据至移位寄存器完毕，且产生 INTCFnT 信号。若要结束当前的发送/接收传输，则不写入 CFnTX 寄存器。
- (12) 当传输完毕之前下一个发送数据未写入 CFnTX 寄存器，传输完毕后，则停止输出串行时钟至 $\overline{\text{SCKFn}}$ 引脚，并将 CFnTSF 位清除为 0。
- (13) 发生接收错误中断请求信号 (INTCFnR) 时，读取 CFnRX 寄存器。
- (14) 如果发生超限错误，写入 CFnSTR.CFnOVE 位= 0，并且，并清除错误标志。
- (15) 若要解除发送/接收使能状态，在检查 CFnTSF 位= 0 之后，写入 CFnCTL0.CFnPWR 位= 0，CFnCTL0.CFnTXE 位= 0 以及 CFnCTL0.CFnRXE 位= 0。

备注 n = 0 至 4

18.6.10 连续传输模式（从机模式，发送模式）

MSB 先行（CFnCTL0.CFnDIR 位= 0），通信类型 1（CFnCTL1.CFnCKP 和 CFnCTL1.CFnDAP 位= 00），通信时钟（f_{CLK}）=外部时钟（ $\overline{\text{SCKFn}}$ ）（CFnCTL1.CFnCKS2 至 CFnCTL1.CFnCKS0 位 = 111），发送数据长度= 8 位（CFnCTL2.CFnCL3 至 CFnCTL2.CFnCL0 位= 0000）

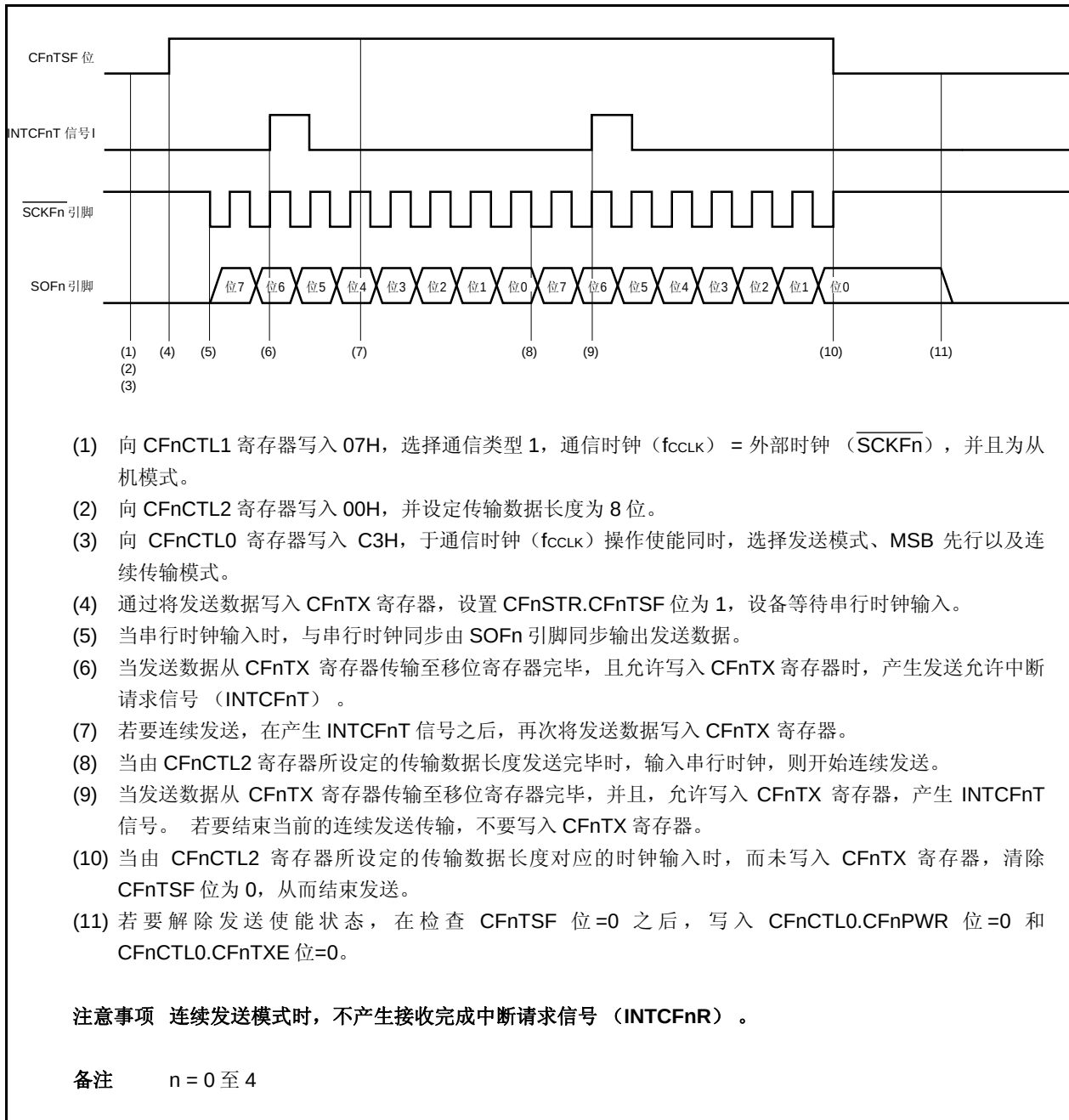
(1) 操作流程



备注

1. 虚线部分表示硬件处理过程。
2. 图中的数字与（2）操作时序中的处理数字相对应。
3. n = 0 至 4

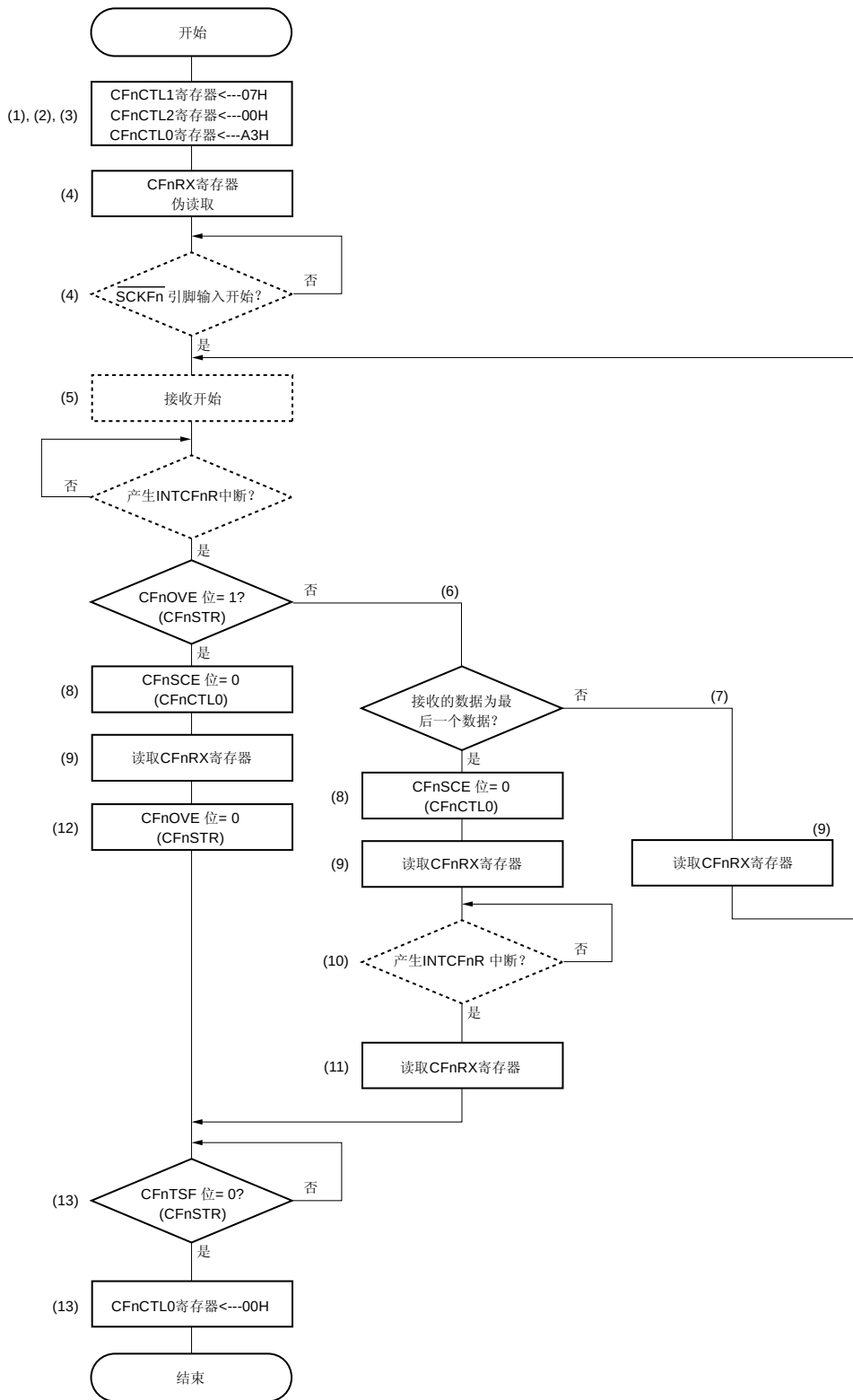
(2) 操作时序



18.6.11 连续传输模式（从机模式，接收模式）

MSB 先行（CFnCTL0.CFnDIR 位= 0），通信类型 1（CFnCTL1.CFnCKP 和 CFnCTL1.CFnDAP 位= 00），通信时钟（f_{CLK}）=外部时钟（ $\overline{\text{SCKFn}}$ ）（CFnCTL1.CFnCKS2 至 CFnCTL1.CFnCKS0 位 = 111），发送数据长度= 8 位（CFnCTL2.CFnCL3 至 CFnCTL2.CFnCL0 位= 0000）

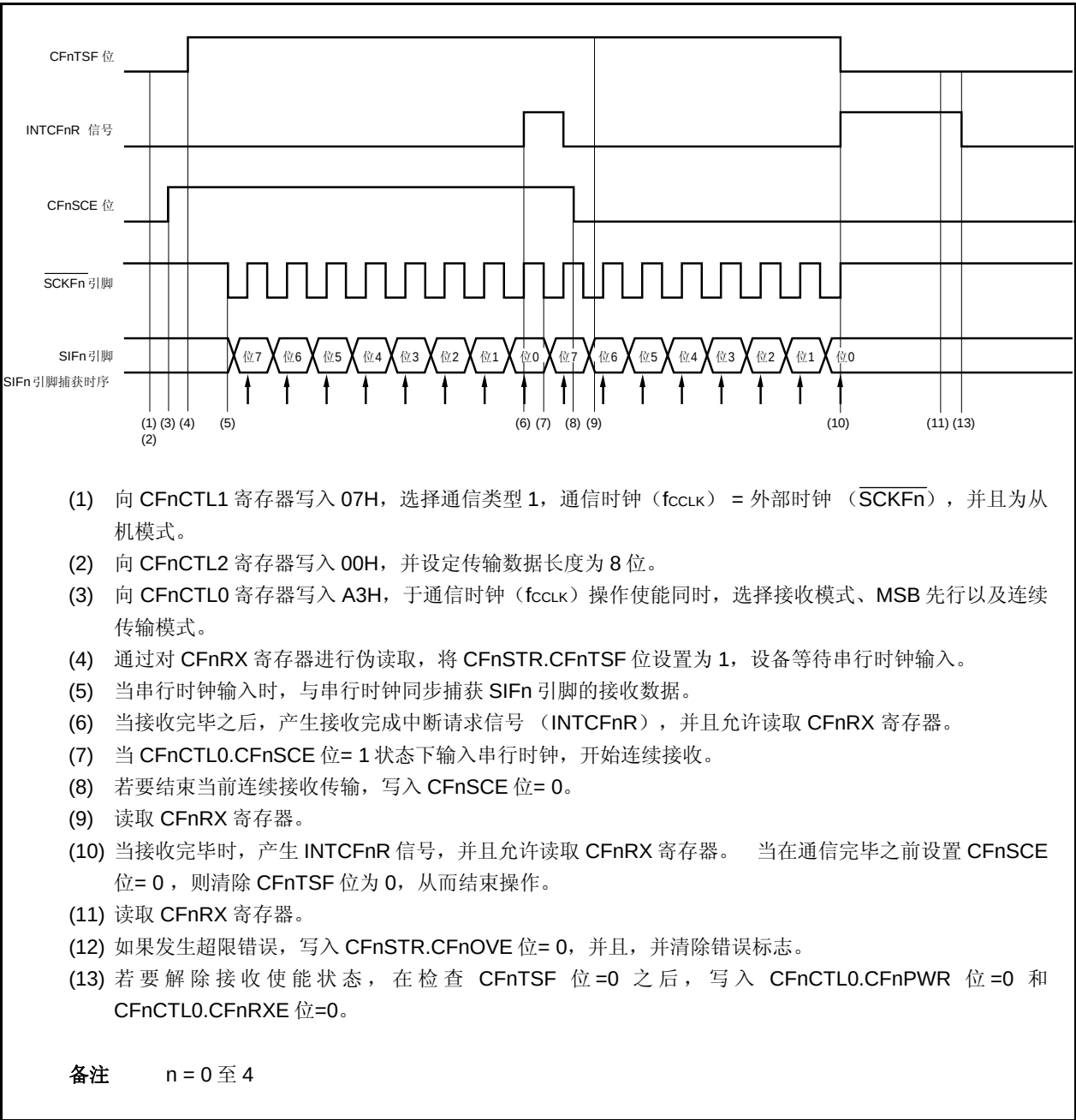
(1) 操作流程



备注

1. 虚线部分表示硬件处理过程。
2. 图中的数字与 (2) 操作时序 中的处理数字相对应。
3. n = 0 至 4

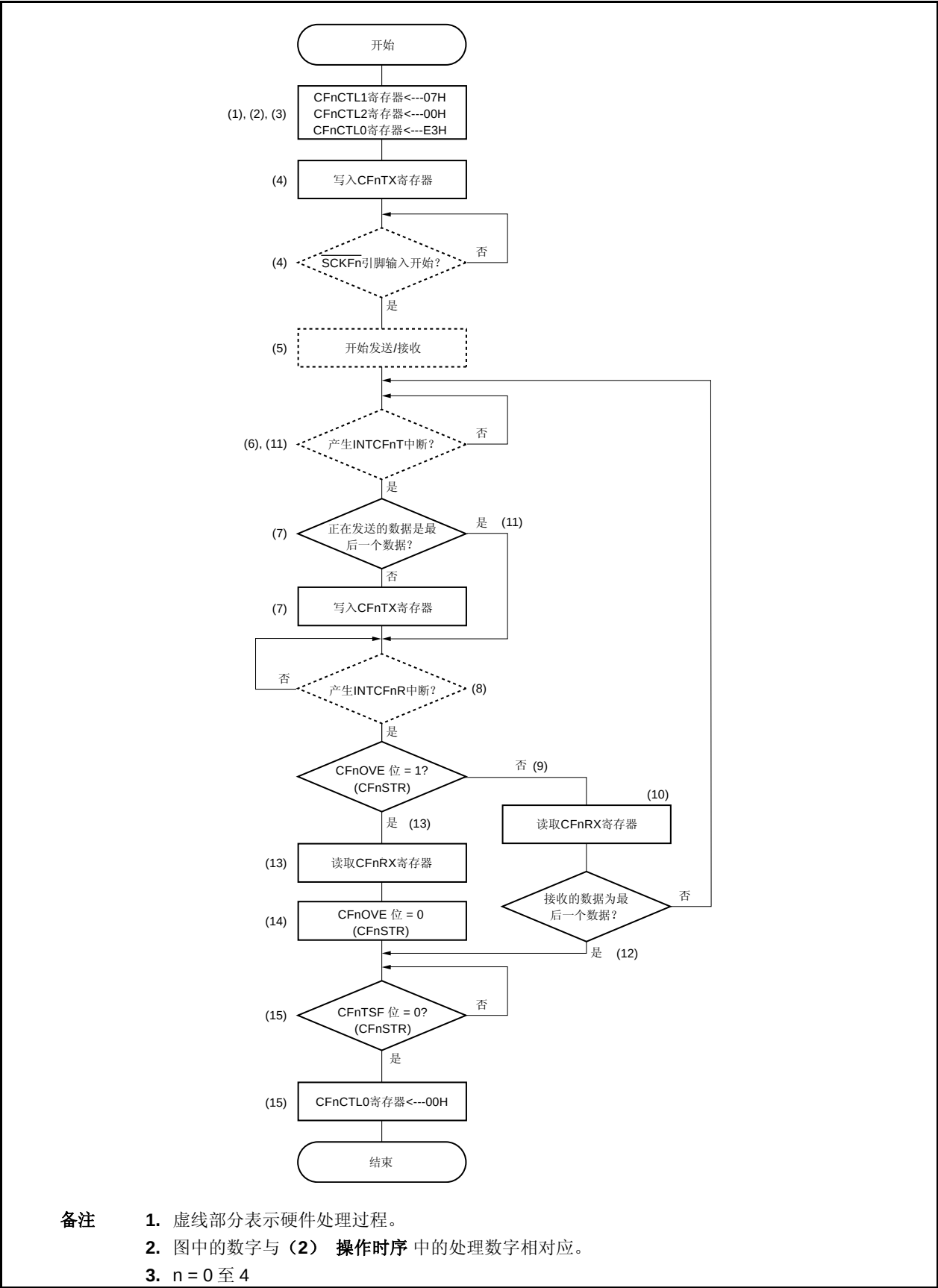
(2) 操作时序



18.6.12 连续传输模式（从机模式，发送/接收模式）

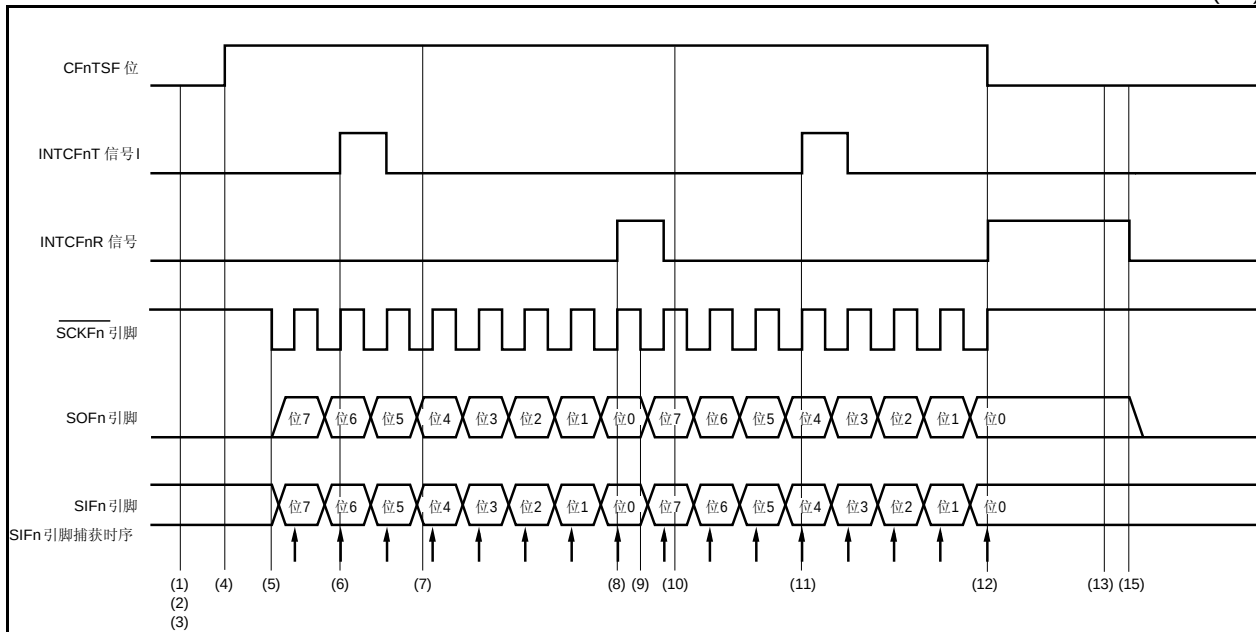
MSB 先行（CFnCTL0.CFnDIR 位= 0），通信类型 1（CFnCTL1.CFnCKP 和 CFnCTL1.CFnDAP 位= 00），通信时钟（f_{CCLK}）=外部时钟（ $\overline{\text{SCKFn}}$ ）（CFnCTL1.CFnCKS2 至 CFnCTL1.CFnCKS0 位 = 111），发送数据长度= 8 位（CFnCTL2.CFnCL3 至 CFnCTL2.CFnCL0 位= 0000）

(1) 操作流程



(2) 操作时序

(1/2)



- (1) 向 CFnCTL1 寄存器写入 07H, 选择通信类型 1, 通信时钟 (f_{CCLK}) = 外部时钟 (SCKFn), 并且为从机模式。
- (2) 向 CFnCTL2 寄存器写入 00H, 并设定传输数据长度为 8 位。
- (3) 向 CFnCTL0 寄存器写入 E3H, 并且, 与通信时钟 (f_{CCLK}) 使能同时选择发送/接收模式、MSB 先行模式以及连续传输模式。
- (4) 通过将发送数据写入 CFnTX 寄存器, 设置 CFnSTR.CFnTSP 位为 1, 设备等待串行时钟输入。
- (5) 当串行时钟输入时, 与串行时钟同步, 输出发送数据至 SOFn 引脚, 并且, 捕获 SIFn 引脚的接收数据。
- (6) 当发送数据从 CFnTX 寄存器传输至移位寄存器完毕, 且允许写入 CFnTX 寄存器时, 产生发送允许中断请求信号 (INTCFnT)。
- (7) 若要连续发送, 在产生 INTCFnT 信号之后, 再次将发送数据写入 CFnTX 寄存器。
- (8) 当由 CFnCTL2 寄存器所设定的传输数据长度接收完毕后, 产生接收完成中断请求信号 (INTCFnR), 并且, 允许读取 CFnRX 寄存器。
- (9) 当串行时钟连续输入, 开始连续发送/接收。
- (10) 读取 CFnRX 寄存器。
- (11) 当发送数据从 CFnTX 寄存器传输至移位寄存器完毕, 并且, 允许写入 CFnTX 寄存器, 产生 INTCFnT 信号。若要结束当前的发送/接收传输, 则不写入 CFnTX 寄存器。

备注 n = 0 至 4

- (12) 当由 CFnCTL2 所设定的传输数据长度相应的时钟输入时，而不写入 CFnTX 寄存器，则产生 INTCFnR 信号。将 CFnTSF 位清除为 0 从而结束发送/接收。
- (13) 产生 INTCFnR 信号时，读取 CFnRX 寄存器。
- (14) 如果发生超限错误，写入 CFnSTR.CFnOVE 位= 0，并且，并清除错误标志。
- (15) 若要解除发送/接收使能状态，在检查 CFnTSF 位= 0 之后，写入 CFnCTL0.CFnPWR 位= 0，CFnCTL0.CFnTXE 位= 0 以及 CFnCTL0.CFnRXE 位= 0。

备注 n = 0 至 4

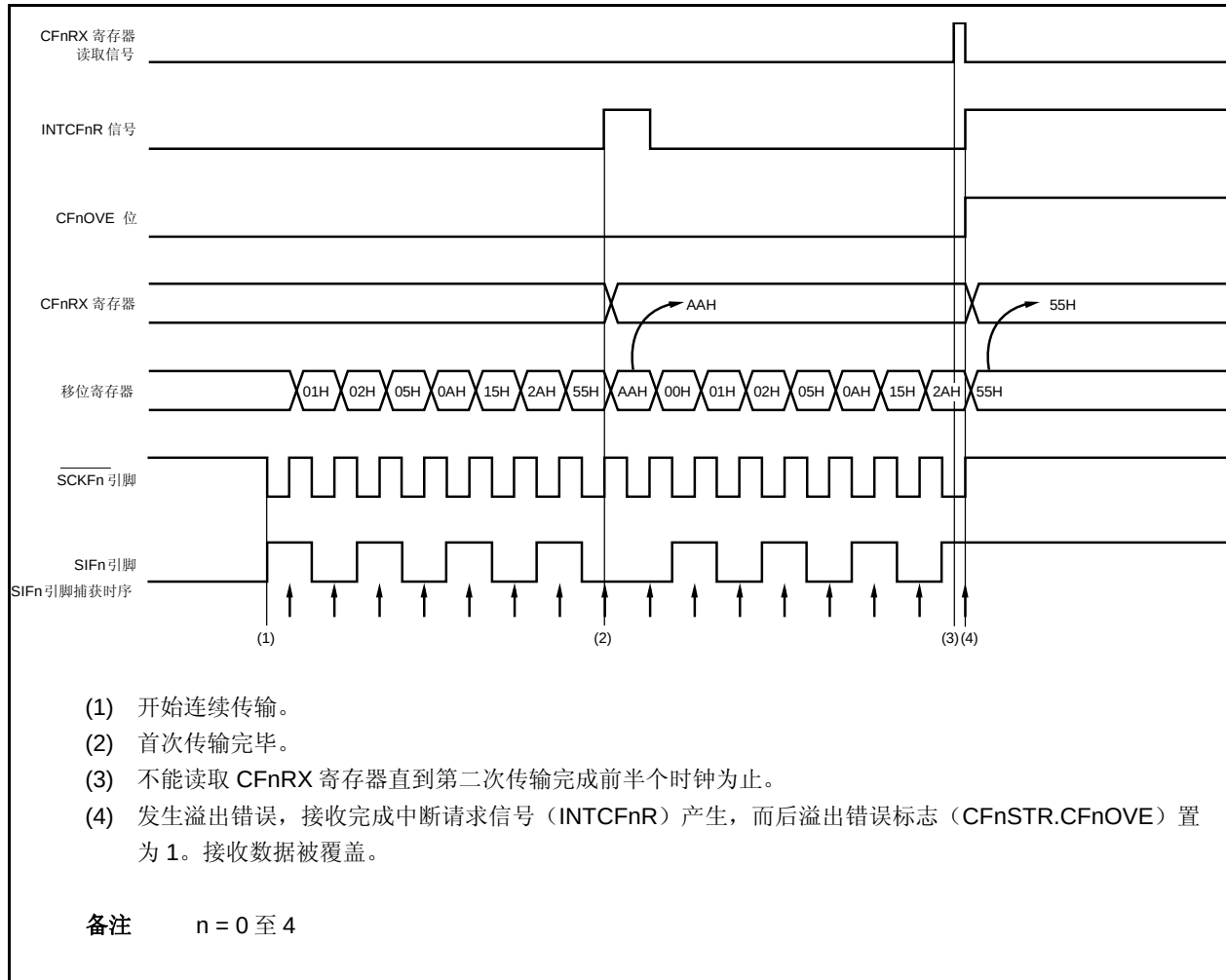
18.6.13 接收错误

连续传输模式中，在接收允许（CFnCTL0.CFnRXE 位= 1）时执行传输，INTCFnR 信号产生后，在读取 CFnRX 寄存器之前完成下一次接收操作时，再次产生接收完成中断请求信号（INTCFnR），并且，溢出错误标志（CFnSTR.CFnOVE）被设置为 1。

即使发生了溢出错误，由于 CFnRX 已更新，因此先前的接收数据丢失。即使发生了超限错误，如果 CFnRX 寄存器没被读取，则下次接收完毕时产生 INTCFnR 信号。

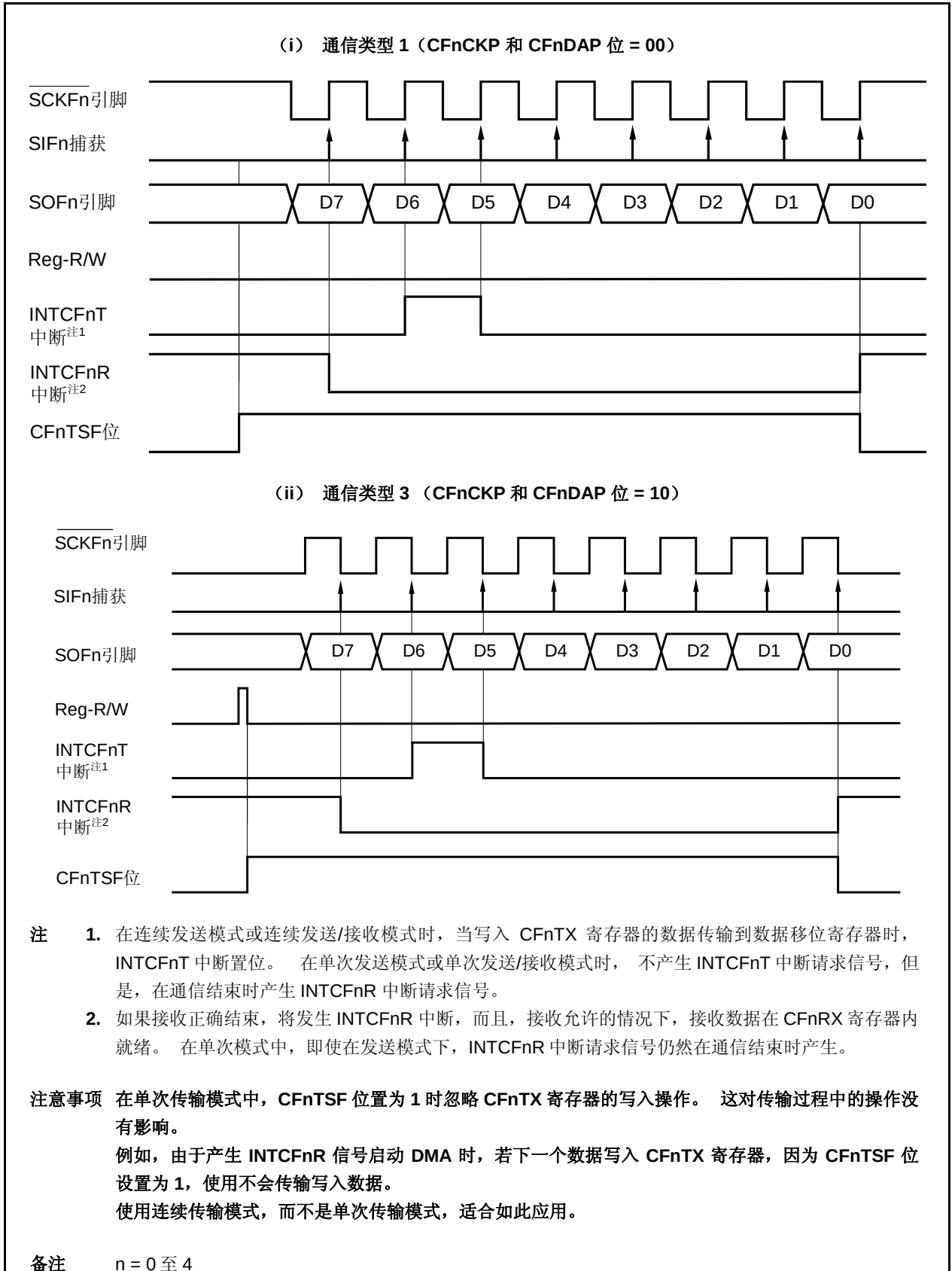
为了避免发生溢出错误，从 INTCFnR 信号产生起至采样下次接收最末位数据前半个时钟，应在此之前完成读取 CFnRX 寄存器。

(1) 操作时序

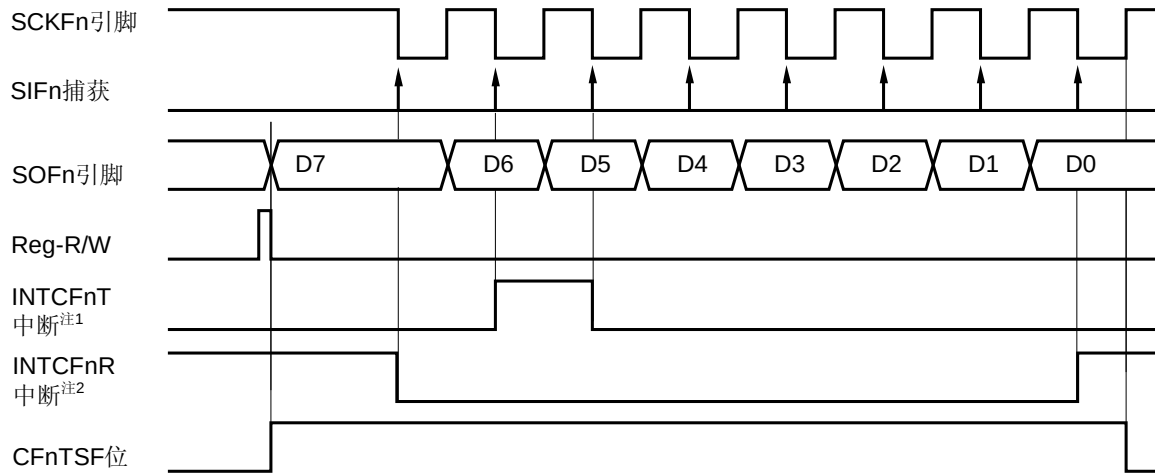


18.6.14 时钟时序

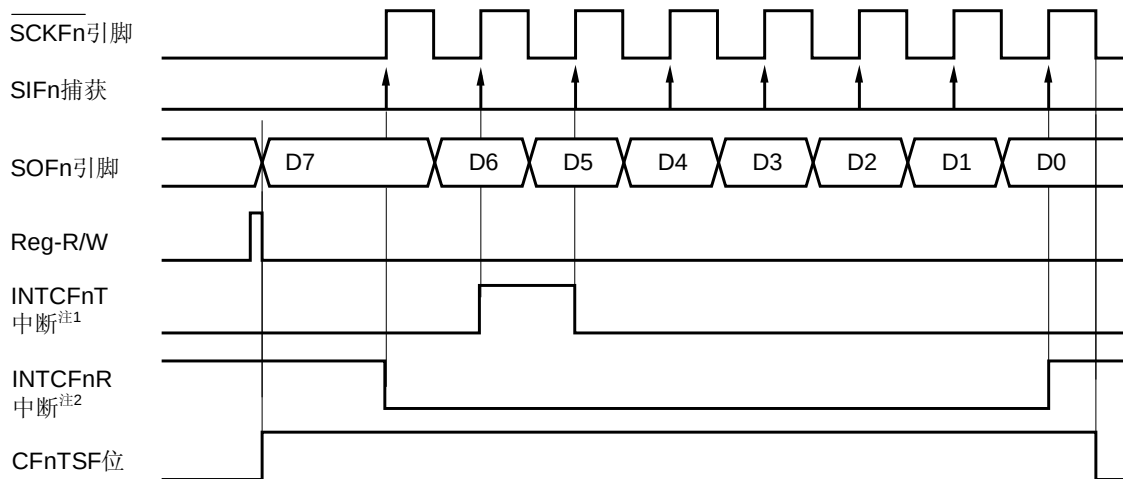
(1/2)



(iii) 通信类型 2 (CFnCKP 和 CFnDAP 位 = 01)



(iv) 通信类型 4 (CFnCKP 和 CFnDAP 位 = 11)



- 注 1. 在连续发送模式或连续发送/接收模式时，当写入 CFnTX 寄存器的数据传输到数据移位寄存器时，INTCFnT 中断置位。在单次发送模式或单次发送/接收模式时，不产生 INTCFnT 中断请求信号，但是，在通信结束时产生 INTCFnR 中断请求信号。
2. 如果接收正确结束，将发生 INTCFnR 中断，而且，接收允许的情况下，接收数据在 CFnRX 寄存器内就绪。在单次模式中，即使在发送模式下，INTCFnR 中断请求信号仍然在通信结束时产生。

注意事项 在单次传输模式中，CFnTSF 位置为 1 时忽略 CFnTX 寄存器的写入操作。这对传输过程中的操作没有影响。

例如，由于产生 INTCFnR 信号启动 DMA 时，若下一个数据写入 CFnTX 寄存器，因为 CFnTSF 位设置为 1，使用不会传输写入数据。

使用连续传输模式，而不是单次传输模式，适合如此应用。

备注 n = 0 至 4

18.7 输出引脚

(1) $\overline{\text{SCKFn}}$ 引脚

当 CSIFn 操作禁止 (CFnCTL0.CFnPWR 位= 0)， $\overline{\text{SCKFn}}$ 引脚输出状态如下所示。

CFnCKP	CFnCKS2	CFnCKS1	CFnCKS0	$\overline{\text{SCKFn}}$ 引脚输出
0	1	1	1	高阻抗状态
	其它			固定为高电平
1	1	1	1	高阻抗状态
	其它			固定为低电平

- 备注**
1. 如果 CFnCTL1.CFnCKP 和 CFnCKS2 至 CFnCKS0 位任何位被重写， $\overline{\text{SCKFn}}$ 引脚输出改变。
 2. n = 0 至 4

(2) SOFn 引脚

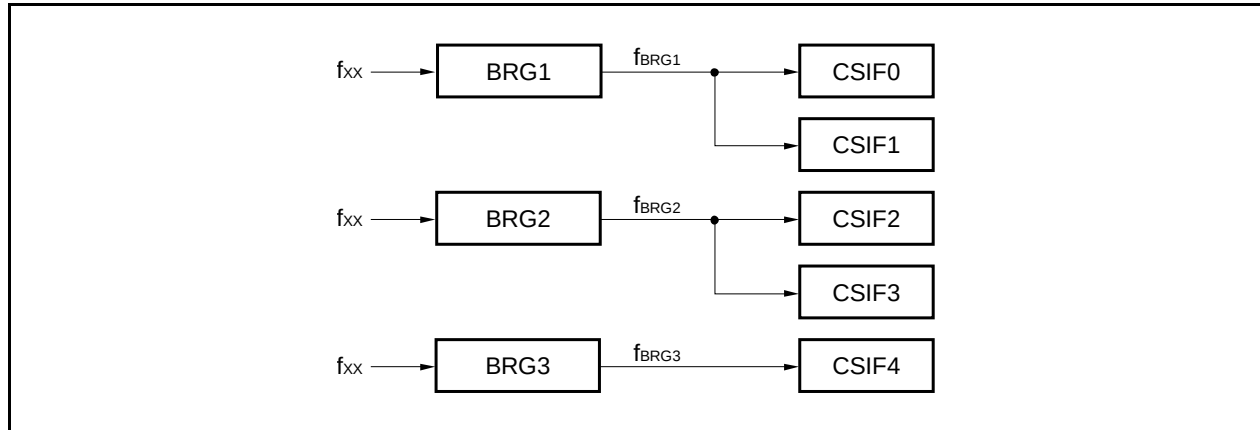
当 CSIFn 操作禁止 (CFnPWR 位= 0)， SOFn 引脚输出状态如下所示。

CFnTXE	CFnDAP	CFnDIR	SOFn 引脚输入
0	x	x	固定为低电平
1	0	x	SOFn 锁存值 (低电平)。
	1	0	CFnTX 值 (MSB)
		1	CFnTX 值 (LSB)

- 备注**
1. 如果 CFnCTL0.CFnTXE，CFnCTL0.CFnDIR 位或 CFnCTL1.CFnDAP 位中任何位被重写， SOFn 引脚输出都会改变。
 2. x: 忽略
 3. n = 0 至 4

18.8 波特率发生器

BRG1 至 BRG3 和 CSIF0 至 CSIF4 波特率发生器连接如下框图所示：



(1) 预分频模式寄存器 1 至 3 (PRSM1 至 PRSM3)

PRSM1 至 PRSM3 为 CSIF 控制波特率信号的发生。

这些寄存器可以按 8 位或按 1 位为单位进行读写。

复位后这些寄存器设置为 00H。

复位后：00H R/W 地址：PRSM1 FFFFF320H, PRSM2 FFFFF324H,
PRSM3 FFFFF328H

	7	6	5	<4>	3	2	1	0
PRSMm (m = 1至3)	0	0	0	BGCEm	0	0	BGCSm1	BGCSm0

BGCEm	波特率输出
0	禁止
1	使能

BGCSm1	BGCSm0	输入时钟选择 (fBGCSm)	设置值 (k)
0	0	fxx/2	0
0	1	fxx/4	1
1	0	fxx/8	2
1	1	fxx/16	3

- 注意事项
1. 操作期间禁止重写 PRSMm 寄存器。
 2. 在设置 BGCEm 位为 1 之前设置 PRSMm 寄存器。
 2. 一定要将位 7 至 5, 3 和 2 设置为 0。

(2) 预分频比较寄存器 1 至 3 (PRSCM1 至 PRSCM3)

PRSCM1 至 PRSCM3 寄存器为 8 位比较寄存器。
这些寄存器可以按 8 位读取或写入。
复位后这些寄存器设置为 00H。

复位后：00H R/W 地址：PRSCM1 FFFF321H, PRSCM2 FFFF325H,
PRSCM3 FFFF329H

PRSCMm

7	6	5	4	3	2	1	0
PRSCMm7	PRSCMm6	PRSCMm5	PRSCMm4	PRSCMm3	PRSCMm2	PRSCMm1	PRSCMm0

注意事项 1. 操作期间禁止重写 PRSCMm 寄存器。
2. 在设置 PRSMm.BGCEm 位为 1 之前设置 PRSCMm 寄存器。

18.8.1 波特率的产生

发送/接收时钟通过分频主时钟产生。 从主时钟产生的波特率由以下公式获得：

$$f_{BRGm} = \frac{f_{xx}}{2^{k+1} \times N}$$

注意事项 将 f_{BRGm} 设为 8 MHz，（CSIF0 至 CSIF2 以及 CSIF4），12 MHz（CSIF3）或更低。

备注 f_{BRGm} = BRGm 计数时钟
f_{xx}: 主时钟振荡频率
k: PRSMm 寄存器设定值 = 0 至 3
N: PRSCMm 寄存器设定值 = 1 至 256
然而，仅当 PRSCMm 寄存器设置为 00H 时 N = 256。
m = 1 至 3

18.9 注意事项

- (1) 使用 DMA 进行发送数据和接收数据时，即使串行传输期间发生超限错误也不执行错误处理。DMA 传输完成后读取 CFnSTR.CFnOVE 位以检查有无超限错误发生。
- (2) 如果在操作期间误将操作时禁止重写的寄存器（CFnCTL0.CFnPWR 位 = 1）进行了重写，则将 CFnCTL0.CFnPWR 位设为 0 然后将 CSIFn 初始化。

操作期间禁止重写的寄存器如下所示：

- CFnCTL0 寄存器：CFnTXE、CFnRXE、CFnDIR、CFnTMS 位
- CFnCTL1 寄存器：CFnCKP、CFnDAP、CFnCKS2 至 CFnCKS0 位
- CFnCTL2 寄存器：CFnCL3 至 CFnCL0 位

- (3) 通信类型 2 或 4（CFnCTL1.CFnDAP 位 = 1）中，接收完成中断（INTCFnR）发生后 $\overline{\text{SCKFn}}$ 时钟的一半时，CFnSTR.CFnTSF 位被清除。

单次传输模式下，忽略通信期间写入的下一个发送数据（CFnTSF 位 = 1），并且不开始下次通信。同样如果设置为只接收通信（CFnCTL0.CFnTXE 位 = 0，CFnCTL0.CFnRXE 位 = 1），即使在通信期间读取接收数据也不开始下一次通信。

因此，使用通信类型 2 或 4（CFnDAP 位 = 1）的单次传输模式时，需特别注意以下情况：

- 若要开始下一次发送，需确认 CFnTSF 位 = 0 而后将发送数据写入 CFnTX 寄存器。
- 设置为只接收通信（CFnTXE 位 = 0，CFnRXE 位 = 1）时，若要进行下一次连续接收，需确认 CFnTSF 位 = 0 而后读取 CFnRX 寄存器。

或者使用连续传输模式代替单次传输模式。用 DMA 传输时特别建议使用连续传输模式。

备注 n = 0 至 4

第十九章 I²C总线

要使用 I²C 总线功能，将 P36/SCL00, P37/SDA00, P40/SDA01, P41/SCL01, P90/SDA02,和 P91/SCL02 引脚设置为复用功能引脚，并将它们设为 N 沟道漏极开路输出。

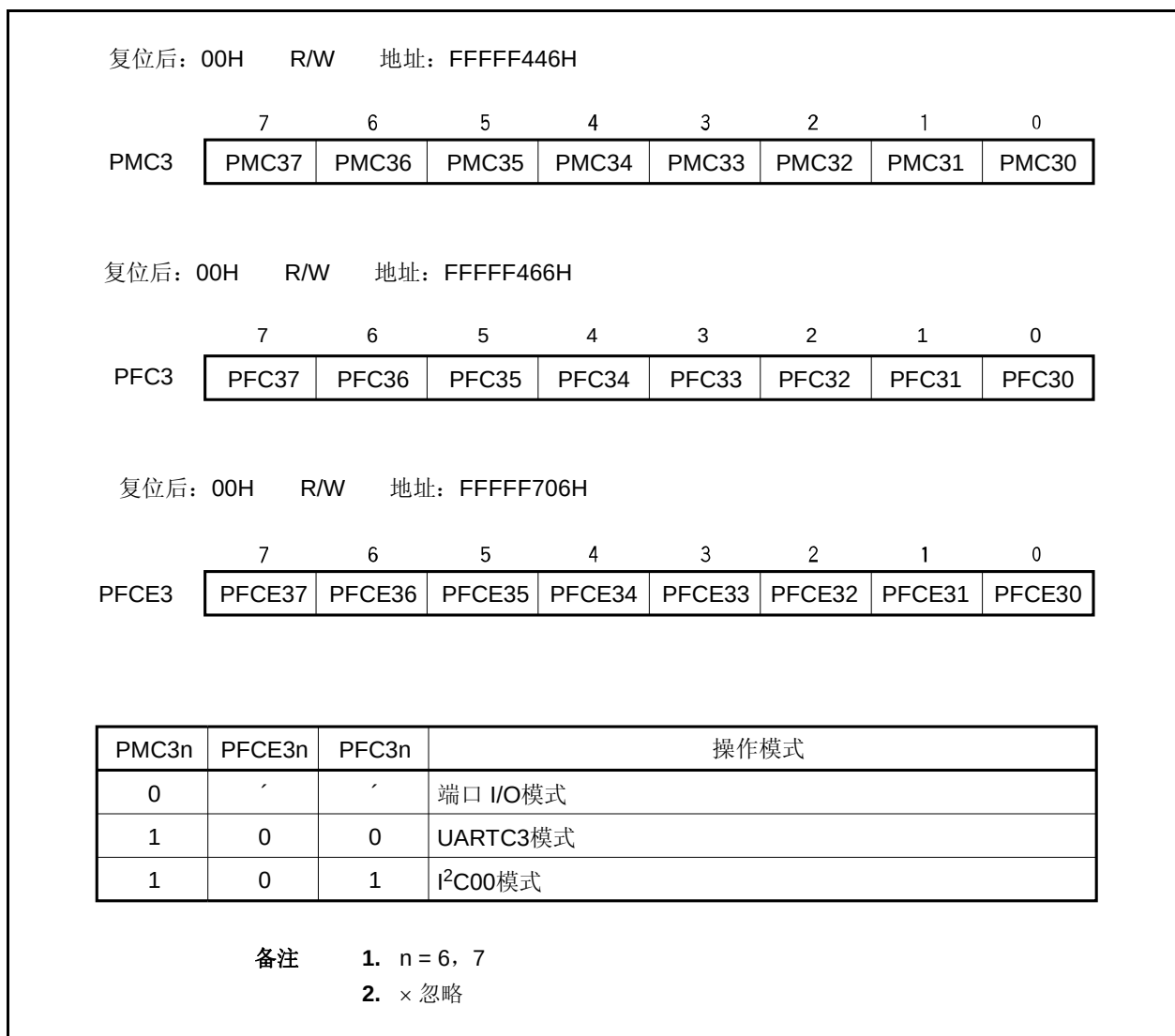
19.1 I²C总线和其它串行接口模式转换

19.1.1 UARTC3 和 I²C00 模式转换

V850ES/JG3-H 和 V850ES/JH3-H 中，UARTC3 和 I²C00 是相同引脚的复用功能，因而不能同时使用。通过使用 PMC3, PFC3 和 PFCE3 寄存器对 UARTC3 和 I²C00 之间的转换预先设置。

注意事项 如果发送或接收期间转换这些功能，UARTC3 和 I²C00 的发送/接收操作得不到保证。确保禁止未使用的模式。

图 19-1. UARTC3 和 I²C00 模式转换设置

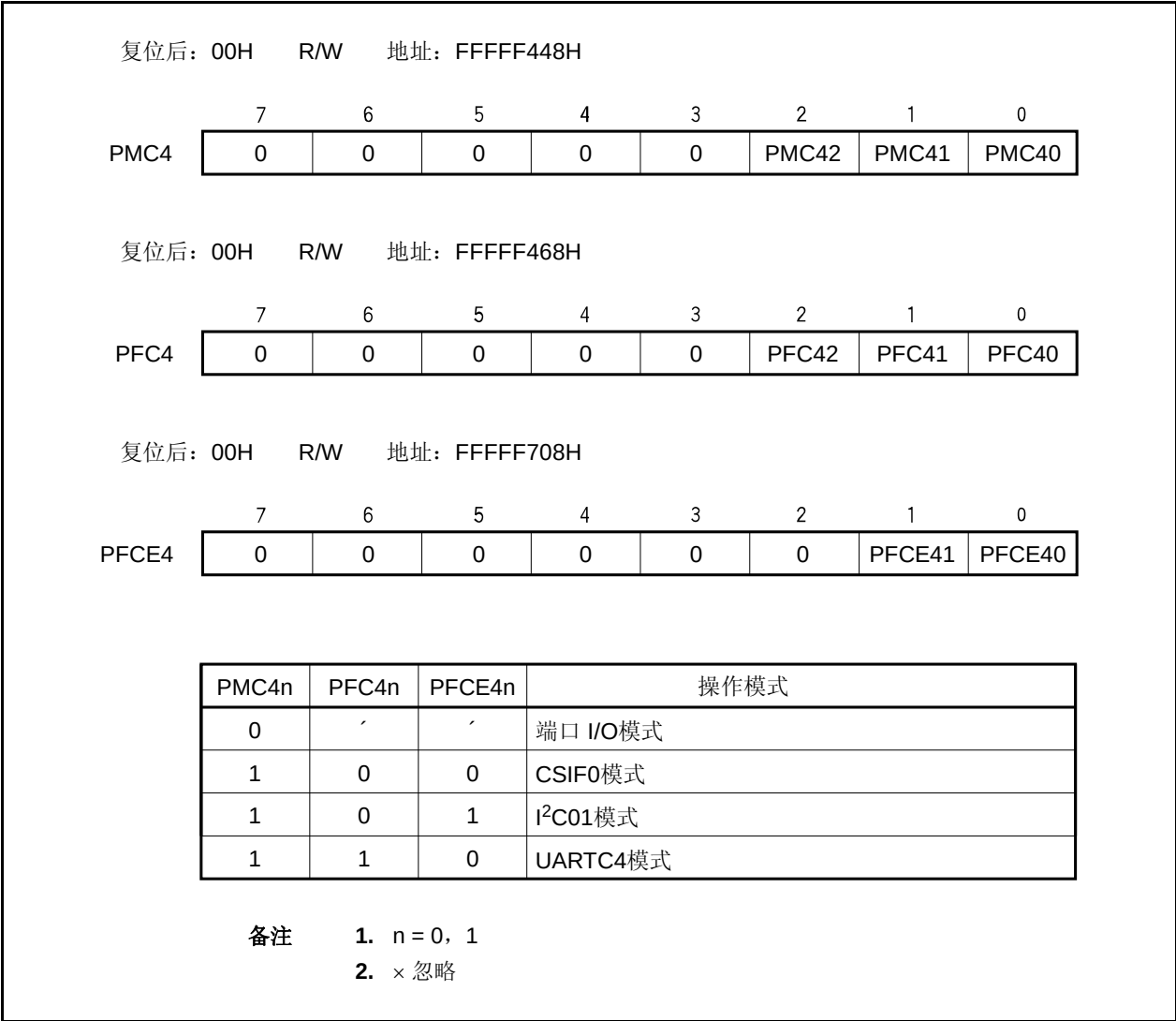


19.1.2 UARTC4, CSIF0 和 I²C01 模式转换

V850ES/JG3-H 和 V850ES/JH3-H 中, UARTC4, CSIF0 以及 I²C01 是相同引脚的复用功能, 因而不能同时使用。必须使用 PMC4、PFC4 和 PFCE4 寄存器对 UARTC4, CSIF0, 和 I²C01 之间的转换预先设置。

注意事项 如果发送或接收期间转换这些功能, UARTC4, CSIF0 和 I²C01 的发送/接收操作得不到保证。 确保禁止未使用的模式。

图 19-2. UARTC4, CSIF0 和 I²C01 模式转换设置



19.2 特性

I²C00 至 I²C02 具有以下两种模式：

- 工作停止模式
- I²C（IC 内部）总线模式（支持多主机通信）

(1) 工作停止模式

该模式下，不执行串行传输，这样能减少功耗。

(2) I²C 总线模式（支持多主机通信）

本模式用作多个设备间进行 8 位数据传输，仅使用两条信号线：一个串行时钟引脚（SCL0n）和一个串行数据总线引脚（SDA0n）。

本模式遵从 I²C 总线格式，主机设备可以通过其串行数据总线，产生“开始条件”、“地址”、“传输方向指示”、“传输数据”和“停止条件”数据信号给从设备。 这种功能可以使 I²C 总线控制应用程序得到简化。

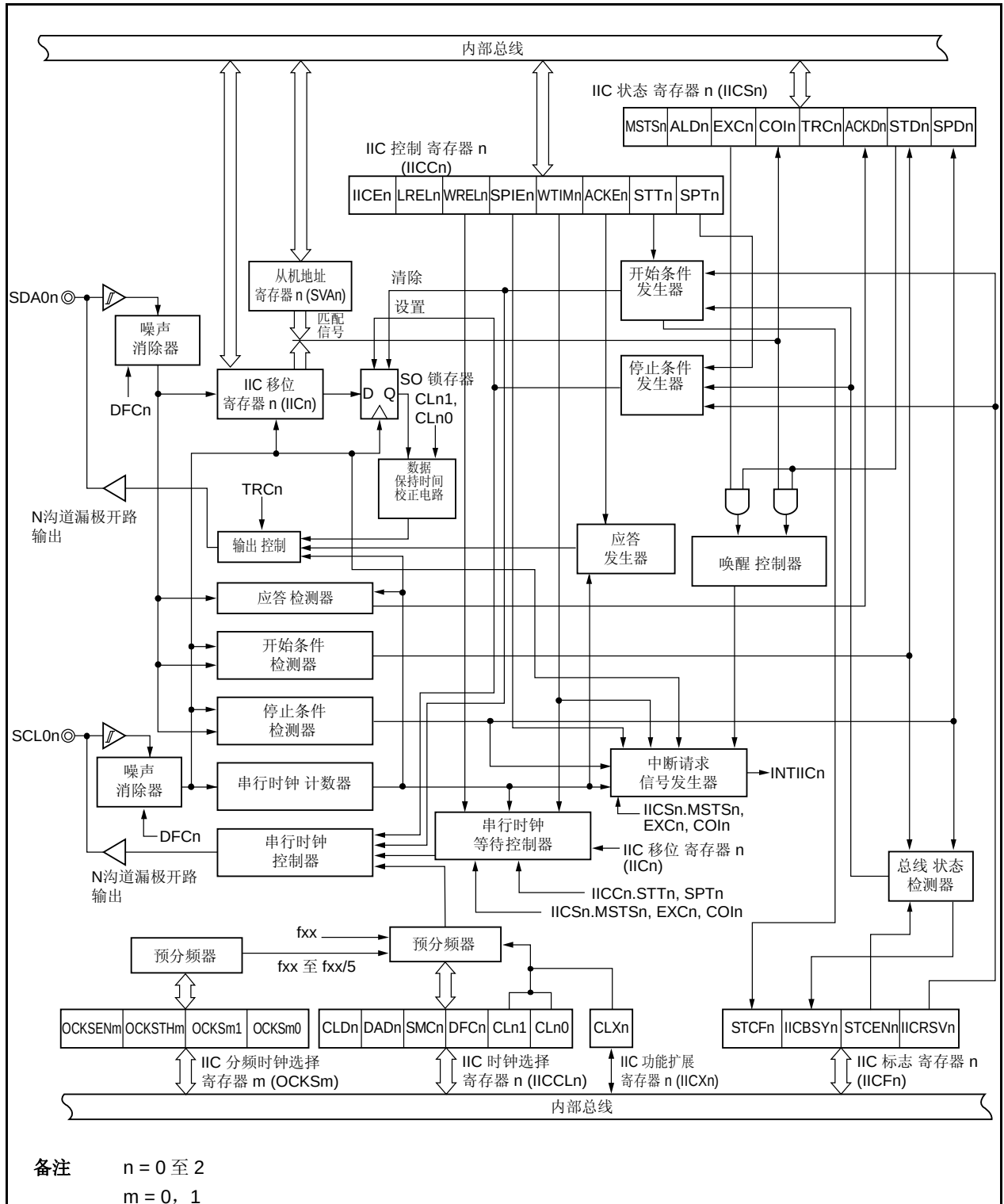
由于 SCL0n 和 SDA0n 引脚用于 N 沟道漏极开路输出，因此，I²C0n 的串行时钟信号线和串行数据信号线都需要上拉电阻。

备注 n = 0 至 2

19.3 配置

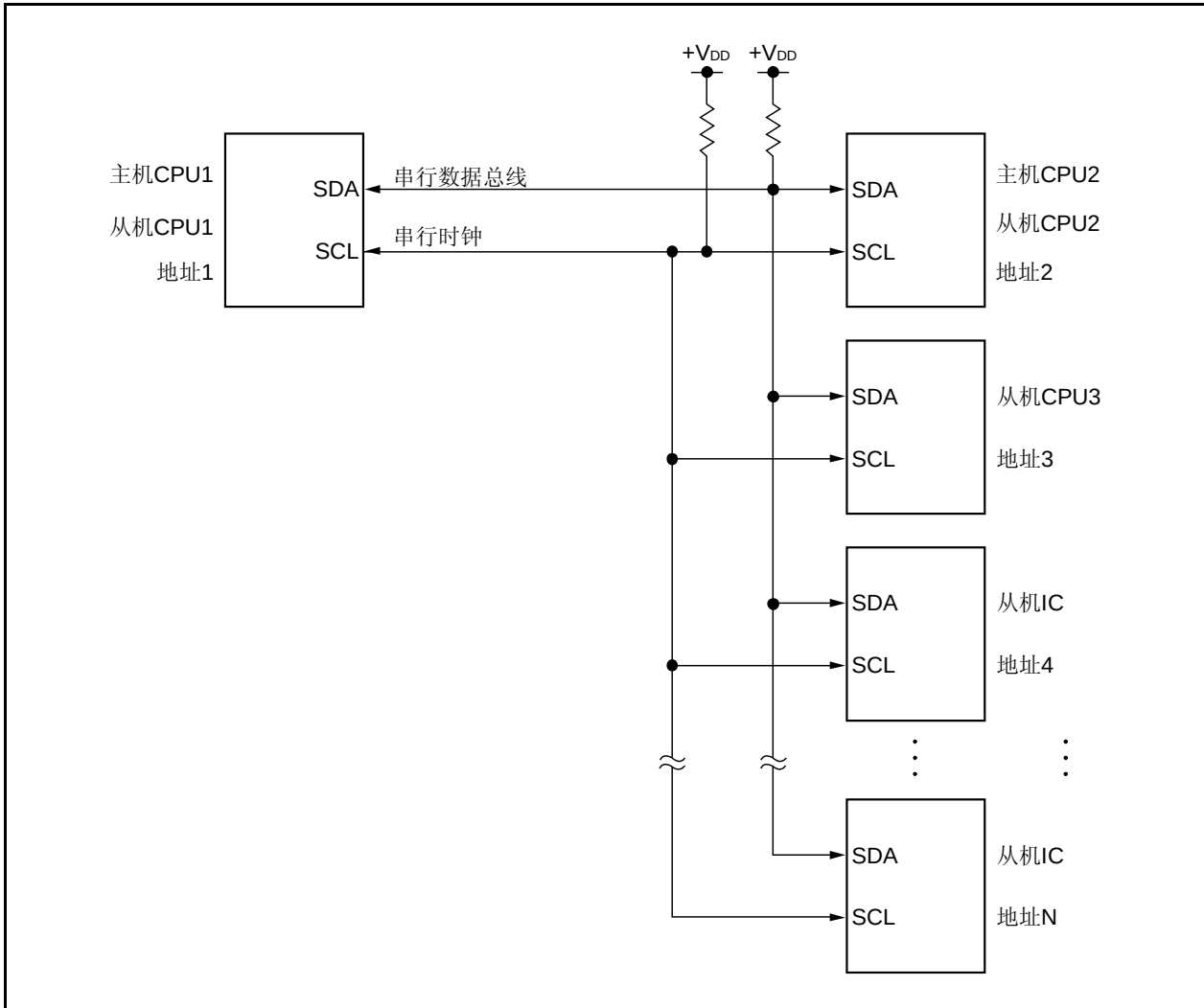
I²C0n 的功能框图如下所示：

图 19-4. I²C0n 的功能框图



以下显示了一个串行总线结构配置示例。

图 19-5. 使用 I²C 总线的串行总线结构示例



I²C0n 包括以下硬件（n = 0 至 2）：

表 19-1. I²C0n 的配置

项目	配置
寄存器	IIC 移位寄存器 n（IICn） 从地址寄存器 n（SVAn）
控制寄存器	IIC 控制寄存器 n（IICCN） IIC 状态寄存器 n（IICSn） IIC 标志寄存器 n（IICF0n） IIC 时钟选择寄存器 n（IICCLn） IIC 功能扩展寄存器 n（IICXn） IIC 时钟分频选择寄存器 0, 1（OCKS0, OCKS1）

(1) IIC 移位寄存器 n（IICn）

IICn 寄存器将 8 位串行数据转换为 8 位并行数据，也可将 8 位并行数据转换为 8 为串行数据，可用于发送和接收（n = 0 至 2）。

IICn 寄存器的读写操作，用来控制实际发送和接收操作。

该寄存器可按 8 位为单位进行读写。

复位后本寄存器被设置为 00H。

(2) 从地址寄存器 n（SVAn）

当 SVAn 寄存器处于从模式时，用来设定本机地址（n = 0 至 2）。

该寄存器可按 8 位为单位进行读写。

复位后本寄存器被设置为 00H。

(3) SO 锁存器

SO 锁存器用于保持 SDA0n 引脚的输出电平（n = 0 至 2）。

(4) 唤醒控制器

当本寄存器收到的地址和 SVAn 寄存器设定的地址匹配时，或收到扩展码时，该电路就会产生一个中断请求信号（INTIICn）（n = 0 至 2）。

(5) 预分频器

预分频器选择使用的采样时钟。

(6) 串行时钟计数器

该计数器对在发送/接收操作期间的串行时钟进行计数，并可用来检验 8 位数据的发送或接收。

(7) 中断请求信号发生器

本电路控制中断请求信号（INTIICn）的产生。

以下两种触发都可以产生一个 I²C 中断请求信号。

- 串行时钟的第 8 个或第 9 个时钟（由 IICn.WTIMn 位设定）下降沿。
- 由于检测到停止条件（由 IICn.SPIEn 位设定）而发生中断。

备注 n = 0 至 2

(8) 串行时钟控制器

主控模式下，本电路将采集到的时钟经由 SCL0n 引脚输出（n = 0 至 2）。

(9) 串行时钟等待控制器

本电路控制等待时间。

(10) $\overline{\text{ACK}}$ 发生器、停止条件检测器、开始条件检测器以及 $\overline{\text{ACK}}$ 监测器。

这些电路用来产生和检测不同的操作状态。

(11) 数据保持时间校正电路

本电路根据相应的 SCL0n 引脚下降沿，为数据产生保持时间。

(12) 开始条件发生器

当 IICn.STTn 位置位时，本电路单元产生一个开始条件信号。

然而，在通信保留禁止状态（IICFn.IICRSVn 位 = 1）下，当总线未被释放（IICFn.IICBSYn 位 = 1）时，该请求被忽略，并且 IICFn.STCFn 位被置为 1。

(13) 停止条件发生器

当 IICn.SPTn 位置位时，该电路单元产生一个停止条件信号。

(14) 总线状态检测器

本电路通过探测总线开始和停止条件，检测总线是否被释放。

然而，总线状态在操作后不能立即被检测到，因此使用 IICFn.STCENn 位将总线状态检测器设置成初始状态。

19.4 寄存器

I²C00 和 I²C02 由下列寄存器进行控制：

- IIC 控制寄存器 0 至 2 (IICC0 至 IICC2)
- IIC 状态寄存器 0 至 2 (IICS0 至 IICS2)
- IIC 标志寄存器 0 至 2 (IICF0 至 IICF2)
- IIC 时钟选择寄存器 0 至 2 (IICCL0 至 IICCL2)
- IIC 功能扩展寄存器 0 至 2 (IICX0 至 IICX2)
- IIC 分频时钟选择寄存器 0, 1 (OCKS0, OCKS1)

以下寄存器也被使用：

- IIC 移位寄存器 0 至 2 (IIC0 至 IIC2)
- 从设备地址寄存器 0 至 2 (SVA0 至 SVA2)

备注 关于复用功能引脚的设置，参见 表 4-20 当端口引脚用作复用功能时的设置。

(1) IIC 控制寄存器 0 至 2 (IICC0 至 IICC2)

IICCn 寄存器允许/停止 I²C0n 操作，设置等待时间并设置其它 I²C 操作 (n = 0 至 2)。

这些寄存器可以按 8 位或按 1 位为单位进行读写。然而，当 IICEn 位为 0 或处于等待期间时，将 SPIEn、WTIMn 和 ACKEn 位置位。当 IICEn 位由“0”设置为“1”时，这些位被同时置位。

复位后这些寄存器设置为 00H。

复位后: 00H R/W 地址: IICC0 FFFFFFFD82H, IICC1 FFFFFFFD92H, IICC2 FFFFFFFDA2H

	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IICCN	IICEn	LRELn	WRELn	SPIEn	WTIMn	ACKEn	STTn	SPTn

(n=0 至 2)

IICEn	I ² Cn 操作允许/禁止的说明
0	操作停止。 IICSn 寄存器复位 ¹ 。内部操作停止。
1	允许操作。
SCL0n 和 SDA0n 线是高电平时确保将该位设置为 1。	
清零条件 (IICEn 位 = 0)	
置位条件 (IICEn 位 = 1)	
- 由指令清除。 - 复位后。	

LRELn ²	退出通信
0	正常工作模式
1	从当前通信操作退出并设置为待机模式。 本设置在被执行之后自动清除。 包括接收到本地无效扩展码的情况下使用。 SCL0n 和 SDA0n 信号线被设为高阻状态。 STTn 和 SPTn 位以及 IICSn 寄存器的 MSTSn, EXCn, COIn, TRCn, ACKDn 和 STDn 位被清除。
从通信中退出的待机状态一直保持有效, 直到遇到下面的通信入口条件为止。 - 检测到停止条件后, 在主机模式下重新开始。 - 在开始条件以后, 发生地址匹配或收到扩展码。	
清零条件 (LRELn 位 = 0)	
置位条件 (LRELn 位 = 1)	
- 执行后自动清除。 - 复位后。	

WRELn ²	等待状态取消控制
0	等待状态不能取消
1	等待状态被取消。 等待取消以后, 本设置自动清除。
清零条件 (WRELn 位 = 0)	
设置状态 (WRELn 位 = 1)	
- 执行后自动清除。 - 复位后。	

- 注 1. IICSn 寄存器、IICFn.STCFn 和 IICFn.IICBSYn 位、IICCLn.CLDn 位和 IICCLn.DADn 位复位。
 2. 当 IICEn 位 = 0 时, 本标志信号无效。

注意事项 当 SCL0n 信号线为高电平而 SDA0n 信号线为低电平时, 如果 I²Cn 操作使能 (IICEn 位 = 1), 则立刻检测到开始条件。 为了避免这种情况, I²Cn 操作使能以后, 立刻使用 1 条位处理指令将 LRELn 位设置为 1。

备注 数据被设置后读取时 LRELn 和 WRELn 位为 0。

SPIEn [※]	当检测到停止条件时，允许/禁止产生中断请求	
0	禁止	
1	允许	
清零条件（SPIEn 位 = 0）		置位条件（SPIEn 位 = 1）
- 由指令清除。 - 复位后。		由指令设置。

WTIMn ^注	等待状态和中断请求发生控制	
0	在第 8 个时钟的下降沿产生中断请求信号。 主控模式：在输出 8 个时钟以后，时钟输出被设置为低电平，并且置位等待状态。 从模式：在输入 8 个时钟以后，时钟被设置为低电平，并为主设备置位等待状态。	
1	在第 9 个时钟信号的下降沿产生中断请求信号。 主控模式：在输出 9 个时钟以后，时钟输出被设置为低电平，并且置位等待状态。 从模式：在输入 9 个时钟以后，时钟被设置为低电平，并为主设备置位等待状态。	
地址传输期间，在第 9 个时钟下降沿发生中断而不考虑该位的设置。地址传输完成时此位设置有效。主控模式下，在地址传输期间的第 9 个时钟下降沿处，插入一个等待状态。在从设备接收到本机地址，发出 $\overline{\text{ACK}}$ 信号之后，在第 9 个时钟的下降沿处插入一个等待状态。然而，当从设备收到扩展码后，在第 8 个时钟的下降沿插入等待状态。		
清零条件（WTIMn 位 = 0）		置位条件（WTIMn 位 = 1）
● 由指令清除。 ● 复位后。		● 由指令设置。

ACKEn ^註	应答控制
0	禁止应答。
1	允许应答。 在第 9 个时钟期间，SDA0n 信号线置位低电平。
ACKEn 位的设置对从设备的地址接收无效， 这种情况下， $\overline{\text{ACK}}$ 信号在地址匹配时产生。 然而，在接收扩展码地址时， ACKEn 位的设置生效。 在接收扩展码系统中设置 ACKEn 位。	
清零条件（ACKEn 位 = 0）	置位条件（ACKEn 位 = 1）
• 由指令清除。 • 复位后。	• 由指令设置。

注 当 IICEn 位 = 0 时，本标志信号无效。

备注 n = 0 至 2

STTn	开始条件触发
0	不产生开始条件。
1	当总线被释放（在 STOP 模式下）： 产生开始条件（主机开始）。当 SCLn 信号线为高电平时，SDA0n 信号线由高电平变为低电平，这样，开始条件就产生了。之后，按比例过了一定的时间之后，SCL0n 信号线变为低电平。 与第三方通信期间： 如果通信保留功能使能（IICFn.IICRSVn 位 = 0）： - 该触发功能作为开始条件保留标志。设置 1 时，其释放总线而后自动产生开始条件。 如果通信保留功能禁止（IICRSVn = 1）： - IICFn.STCFn 位设置为 1，并且 STTn 位的设置信息（1）被清除，此触发不能产生开始条件。 处于等待状态（当使用主机设备时）： 等待状态释放后产生重新开始条件。
有关设置时序的注意事项： 主机接收：传输期间不能设置为 1。只能在 ACKEn 位已经设置 0，而且从设备已经被告知最末接收时，才可以置为 1。 主机发送：在 ACK 发生期间，通常不会产生开始条件。在输出第 9 个时钟的等待期间置 1。 从设备下：即使禁止通信保留功能（IICRSVn 位 = 1）时，也会进入通信保留状态。 - SPTn 位被禁止的同时将其设置为 1。 - 当 STTn 位设置为 1 时，禁止再次将其置 1，直到设置被清 0 为止。	
清零条件（STTn 位 = 0）	设置状态（STTn 位 = 1）
- 在通信保留禁止状态，当 STTn 位设置为 1 时。 - 仲裁失败后清 0。 - 主机产生开始条件后清除。 - 当 LRELn 位 = 1 时（通信从模式）。 - IICEn 位 = 0（操作停止）时。 - 复位后。	由指令设置。

备注

1. 如果在数据设置后立即读取 STTn 位，其值为 0。
2. n = 0 至 2

SPTn	停止条件触发
0	不产生停止条件。
1	产生停止条件（终止主机设备传输）。 SDA0 信号线变为低电平以后，或者将 SCL0n 信号线设置为高电平，或者一直等到 SCL0n 引脚变为高电平。之后，按比例经过一段时间后，SDA0n 信号线由低电平变为高电平，产生了停止条件。
有关设置时序的注意事项： 主机接收： 传输期间不能设置为 1。 只能在 ACKEn 位已经设置 0 而且从设备已经被告知最末接收后的等待期间，才可以置为 1。 主机发送： 在 ACK 接收期间，通常不会产生停止条件。在输出第 9 个时钟的等待期间置 1。 • 不能和 STTn 位同时置 1。 • 只有在主控模式下注，SPTn 位才可以设置为 1。 • 在 WTIMn 位已经设置 0 时，如果在输出 8 个时钟后的等待期间将 SPTn 位设置为 1，注意，停止条件将在第 9 个时钟的高电平期间产生。 如果在输出 8 个时钟后的等待期间，WTIMn 位应该从 0 变为 1，并且，在输出第 9 个时钟的后的等待期间可以将 SPTn 位设置为 1。 • 当 SPTn 位设置为 1 时，禁止再次将 SPT0 位置为 1，直到设置被清 0 为止。	
清零条件（SPTn 位 = 0）	置位条件（SPTn 位 = 1）
• 仲裁失败后清 0。 • 检测到停止条件后自动清 0。 • 当 LRELn 位 = 1 时（通信从模式）清除。 • IICEn 位 = 0（操作停止）时。 • 复位后。	• 由指令设置。

注 只有在主机模式下才可以设置 SPTn 位为 1。但是，当 IICRSVn 位为 0 时，要在操作使能之后检测到第一个停止条件之前执行主控操作的话，必须将 SPTn 位设为 1 并设置停止条件。详细情况参见 19.15 注意事项。

注意事项 TRCn 位 = 1 时，在第 9 个时钟期间 WRELn 位被置 1 并且等待状态被取消，TRCn 位清除为 0 而 SDA0n 信号线进入高阻状态。

备注 1. 如果在数据设置以后立即读取 SPTn 位，其值为 0。
 2. n = 0 至 2

(2) IIC 状态寄存器 0 至 2 (IICS0 至 IICS2)

IICS_n 寄存器表明 I²C0_n 的状态 (n = 0 至 2)。

这些寄存器是以 8 位或 1 位为单位进行访问的只读寄存器。但是, IICS_n 寄存器只能在 IICCN.STTn 位为 1 或等待期间读取。

复位后这些寄存器设置为 00H。

注意事项 在以下状态中, 禁止访问 IICS_n 寄存器。有关详细说明, 参见 3.4.8 (2) 访问片上专用外围 I/O 寄存器。

- 当 CPU 工作于副时钟, 而且主时钟振荡停止时
- 当 CPU 工作于内部振荡时钟时

(1/3)

复位后: 00H R 地址: IICS0 FFFFFD86H, IICS1 FFFFFD96H, IICS2 FFFFFDA6H

	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IICS _n	MSTSn	ALDn	EXCn	COLn	TRCn	ACKDn	STDn	SPDn

(n=0 至 2)

MSTSn	主机设备状态
0	从设备状态或通信待机状态
1	主机设备通信状态
清零条件 (MSTSn 位 = 0)	
• 当检测到停止条件时。 • 当 ALDn 位 = 1 (仲裁失败)。 • 由 LRELn 位 = 1 清除 (通信从模式)。 • 当 IICEn 位由 1 变为 0 (工作停止) 时。 • 复位后。	
置位条件 (MSTSn 位 = 1)	
• 当产生开始条件时。	
ALDn	仲裁失败检测
0	此状态意味着不存在总线仲裁或仲裁结果为'赢'。
1	此状态指示仲裁结果为失败, MSTSn 位被清除为 0。
清零条件 (ALDn 位 = 0)	
• IICS_n 寄存器被读取注后, 自动清 0^注。 • 当 IICEn 位由 1 变为 0 (工作停止) 时。 • 复位后。	
置位条件 (ALDn 位 = 1)	
• 当总线仲裁结果为'失败'。	
EXCn	扩展码接收检测
0	未收到扩展码。
1	收到扩展码。
清零条件 (EXCn 位 = 0)	
• 当检测到开始条件时 • 检测到停止条件时 • 通过 LRELn 位= 1 (保留通信) 清除 • IICEn 位由 1 变为 0 时 (操作停止) • 复位后。	
置位条件 (EXCn 位 = 1)	
• 当接收地址数据的高 4 位为 0000 或 1111 (在第_8 个时钟的上升沿置位)。	

注 当对除了 IICS_n 寄存器外的其它位进行位处理时, 该位也被清 0。

COIn	匹配地址检测	
0	地址不匹配。	
1	地址匹配。	
清零条件（COIn 位 = 0）		置位条件（COIn 位 = 1）
• 当检测到开始条件时 • 检测到停止条件时 • 通过 LRELn 位 = 1（保留通信）清除 • IICEn 位由 1 变为 0 时（操作停止） • 复位后		• 当接收到的地址和本机地址（SVAn 寄存器）匹配时（在第 8 个时钟的上升沿置位）。

TRCn	发送/接收状态检测	
0	接收状态（不同于发送状态）。 SDA0n 信号线设置为高阻态。	
1	发送状态。 SO 锁存器内的数据允许从 SDA0n 信号线输出（在第一个字节的第 9 个时钟下降沿开始有效）。	
清零条件（TRCn 位 = 0）		置位条件（TRCn 位 = 1）
• 当检测到停止条件时。 • 由 LRELn 位 = 1 清除（通信从模式）。 • 当 IICEn 位由 1 变为 0（工作停止）时。 • 由 IICn.WRELn 位 = 1 清除^注。 • 当 ALDn 位从 0 变为 1（仲裁失败） • 复位后。 主机 • 当第一个字节的 LSB（传输方向指示位）输出为_1 时。” 从机 • 当检测到开始条件时。 当不用于通信时		主机 • 当产生开始条件时。 • 当第一个字节的 LSB（传输方向指示位）输出为 0 时。 从机 • 当第一个字节的 LSB（传输方向指示位）输入为 1 时。

ACKDn	ACK 检测	
0	未检测到 ACK。	
1	检测到 ACK。	
清零条件（ACKDn 位 = 0）		置位条件（ACKD 位 = 1）
• 当检测到停止条件时。 • 处于下一字节第一个时钟上升沿。 • 由 LRELn 位 = 1 清除（通信从模式）。 • 当 IICEn 位由 1 变为 0 时（操作停止） • 复位后。		• SDA0n 引脚在 SCL0n 引脚的第 9 个时钟的上升沿设置为低电平后。

注 当 WRELn 位设置为 1，并且等待状态在第九个时钟由 TRCn 位 = 1 而取消时，TRCn 被清 0，而且 SDA0n 信号线变为高阻态。

备注 n = 0 至 2

STDn	开始条件检测	
0	未检测到开始条件。	
1	检测到开始条件。 这表明地址传输段有效。	
清零条件（STDn 位 = 0）		设置状态（STDn 位 = 1）
• 当检测到停止条件时。 • 在地址传输后的下一个字节的第一个时钟的上升沿。 • 由 LRELn 位 = 1 清除（通信从模式）。 • 当 IICEn 位由 1 变为 0 时（操作停止） • 复位后。		• 当检测到开始条件时。

SPDn	停止条件检测	
0	未检测到停止条件。	
1	检测到停止条件。 主机设备的通信终止，并且总线被释放。	
清零条件（SPDn 位 = 0）		置位条件（SPDn 位 = 1）
• 在本位置位和检测到开始条件之后，在地址传输的第一个时钟上升沿。 • 当 IICEn 位由 1 变为 0 时（操作停止） • 复位后。		• 当检测到停止条件时。

备注 n = 0 至 2

(3) IIC 标志寄存器 0 至 2 (IICF0 至 IICF2)

IICFn 寄存器设置 I²C0n 操作模式并表明 I²C 总线状态。

这些寄存器可以按 8 位或按 1 位为单位进行读写。但是，STCFn 和 IICBSYn 位为只读位。

IICRSVn 允许/禁止通信保留功能（参见 19.14 通信保留）。

IICBSYn 位的初始值通过 STCENn 位设置（参见 19.15 注意事项）。

IICRSVn 和 STCENn 位只能在 I²C0n 操作禁止 (IICn.IICEn 位 = 0) 时写入。使能操作后，IICFn 可被读取 (n = 0 至 2)。

复位后这些寄存器设置为 00H。

复位后: 00H R/W[※] 地址: IICF0 FFFFFFFD8AH, IICF1 FFFFFFFD9AH, IICF2 FFFFFFFDAAH

	<7>	<6>	5	4	3	2	<1>	<0>
IICFn	STCFn	IICBSYn	0	0	0	0	STCENn	IICRSVn

(n=0 至 2)

STCFn	STTn 位清除
0	发出开始条件。
1	不发出开始条件, STTn 位被清除。
清零条件 (STCFn 位 = 0)	
置位条件 (STCFn 位 = 1)	
- 由 IICFn.STTn 位 = 1 清除。 - IICFn.IICEn 位 = 0 时 - 复位后。	禁止通信保留 (IICRSVn 位 = 1) 时, 不发出开始条件且 STTn 标志清除为 0 时。

IICBSYn	I ² C0n 总线状态
0	总线释放状态 (STCENn 位 = 1 时缺省为通信状态)。
1	总线通信状态 (STCENn 位 = 0 时缺省为通信状态)。
清零条件 (IICBSYn 位 = 0)	
设置状态 (IICBSYn 位 = 1)	
- 当检测到停止条件时。 - IICEn 位 = 0 时。 - 复位后。	- 当检测到开始条件时。 - 通过 STCENn 位 = 0 时设置 IICEn 位。

STCENn	初始化开始使能触发
0	开始条件不能产生直到操作使能后 (IICEn 位 = 1), 检测到停止条件为止。
1	开始条件产生即使操作使能后 (IICEn 位 = 1) 检测不到停止条件。
清零条件 (STCENn 位 = 0)	
设置状态 (STCENn 位 = 1)	
- 当检测到开始条件时。 - 复位后。	由指令设置。

IICRSVn	通信保留功能禁止位
0	允许通信保留
1	禁止通信保留
清零条件 (IICRSVn 位 = 0)	
设置状态 (IICRSVn 位 = 1)	
- 由指令清除。 - 复位后。	由指令设置。

注 位 6 和位 7 为只读位。

- 注意事项**
1. 仅当操作停止时 (IICEn 位 = 0), 才可以写入 STCENn 位。
 2. STCENn 位=1 时, 紧随 I²Cn 操作使能之后, 不管总线的实际状态如何, 总线释放状态 (IICFn.IICBSYn 位 = 0) 总是能够识别。 因此, 若要产生首个开始条件 (STTn 位 = 1), 有必要确认总线是否已经释放, 以便不干扰其它通信。
 3. 仅当操作停止时 (IICEn 位 = 0), 才可以写入 IICRSVn 位。

(4) IIC 时钟选择寄存器 0 至 2 (IICCL0 至 IICCL2)

IICCLn 寄存器为 I²C0n 设置传输时钟。

这些寄存器可以按 8 位或按 1 位为单位进行读写。但是，CLDn 和 DADn 位为只读位。

当 IICn.IICEn 位 = 0 时，设置 IICCLn 寄存器。

SMCn, CLn1 和 CLn0 位通过联合使用 IICXn.CLXn 位与 OCKSm 寄存器的 OCKSTHm, OCKSm1 和 OCKSm0 位进行设置 (参见 19.4 (6) I²C0n 传输时钟设置方法) (n = 0 至 2, m = 0, 1)。

复位后这些寄存器设置为 00H。

复位后: 00H R/W* 地址: IICCL0 FFFFFD84H, IICCL1 FFFFFD94H, IICCL2 FFFFFDA4H

	7	6	<5>	<4>	3	2	1	0
IICCLn	0	0	CLDn	DADn	SMCn	DFCn	CLn1	CLn0

(n=0 至 2)

CLDn	SCL0n 引脚电平检测 (仅当 IICn.IICEn 位 = 1 时有效)
0	SCL0n 引脚检测为低电平。
1	SCL0n 引脚检测为高电平。
清零条件 (CLDn 位 = 0)	
- 当 SCL0n 引脚为低电平时。 - IICEn 位 = 0 (操作停止) 时。 - 复位后。	
设置状态 (CLDn 位 = 1)	
当 SCL0n 引脚为高电平时。	

DADn	SDA0n 引脚电平检测 (仅当 IICEn 位 = 1 时有效)
0	SDA0n 引脚检测为低电平。
1	SDA0n 引脚检测为高电平。
清零条件 (DADn 位 = 0)	
- 当 SDA0n 引脚为低电平时。 - IICEn 位 = 0 (操作停止) 时。 - 复位后。	
置位条件 (DADn 位 = 1)	
当 SDA0n 引脚为高电平时。	

SMCn	工作模式切换
0	工作于标准模式。
1	工作于高速模式。

DFCn	数字滤波器操作控制
0	数字滤波器关闭。
1	数字滤波器开启。
数字滤波器只能在高速模式下使用。 在高速模式下, 传输时钟都不会根据 DFCn 位的设置 (开启/关闭) 而改变。 数字滤波器用于高速模式下的抑制噪声。	

注 位 4 和位 5 为只读位。

注意事项 一定要将位 7 和 6 清为 0。

备注 IICn.IICEn 位 = 0 时, 若读取 CLDn 和 DADn 位, 其值为 0。

(5) IIC 功能扩展寄存器 0 至 2 (IICX0 至 IICX2)

IICXn 寄存器设置 I²C0n 功能扩展（仅在高速模式下有效）。

这些寄存器可以按 8 位或按 1 位为单位进行读写。

通过联合使用 IICCLn 寄存器的 SMCn, CLn1 和 CLn0 位以及 OCKSm 寄存器的 OCKSTHm, OCKSm1 和 OCKSm0 位对 CLXn 位进行设置（参见 19.4 (6) I²C0n 传输时钟设置方法）（m = 0, 1）。

当 IICCLn.IICEn 位 = 0 时，设置 IICXn 寄存器。

复位后这些寄存器设置为 00H。

复位后：00H R/W 地址：IICX0 FFFFFD85H, IICX1 FFFFFD95H, IICX2 FFFFFDA5H

	7	6	5	4	3	2	1	<0>
IICXn	0	0	0	0	0	0	0	CLXn
(n = 0至2)								

(6) I²C0n 传输时钟设置方式

使用以下算式计算 I²C0n 传输时钟频率（f_{SCL}）（n = 0 至 2）。

$$f_{SCL} = 1/(m \times T + t_r + t_f)$$

m = 24, 48, 72, 96, 108, 120, 144, 172, 192, 240, 264, 344, 352, 396, 440, 516, 688, 860（参见表 19-2 时钟设置）。

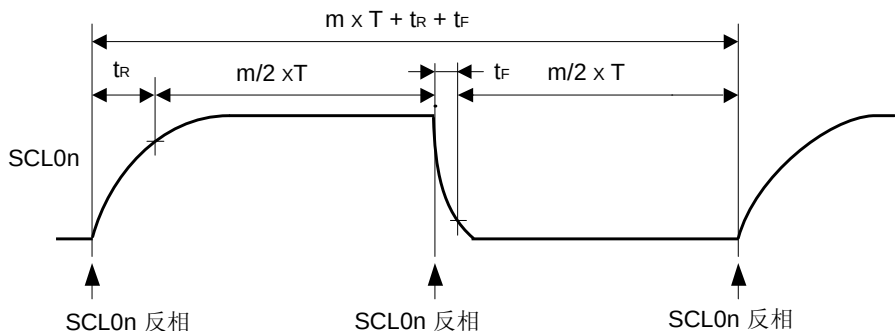
T: 1/f_{xx}

t_r: SCL0n 引脚上升时间

t_f: SCL0n 引脚下降时间

例如，当 f_{xx} = 19.2 MHz, m = 198, t_r = 200 ns, 和 t_f = 50 ns 时，I²C0n 传输时钟频率（f_{SCL}）使用下面算式计算。

$$f_{SCL} = 1/(198 \times 52 \text{ ns} + 200 \text{ ns} + 50 \text{ ns}) \cong 94.7 \text{ kHz}$$



通过联合使用 IICCLn 寄存器的 SMCn, CLn1 和 CLn0 位、IICXn 寄存器的 CLXn 以及 OCKSm 寄存器的 OCKSTHm, OCKSm1 和 OCKSm0 位来设置被选择的时钟（n = 0 至 2, m = 0, 1）。

表 19-2. 时钟设置

IICXn		IICCLn		选择时钟	传输时钟	可设置的主时钟 频率（fxx）范围	传输速度	操作模式
CLXn	SMCn	CLn1	CLn0					
0	0	0	0	fxx/6 (OCKSm = 11H)	fxx/264	24.00 MHz ≤ fxx ≤ 25.14 MHz	90.91 kHz 至 95.23 kHz	标准模式 （SMCn=0）
				fxx/8 (OCKSm= 12H)	fxx/352	24.00 MHz ≤ fxx ≤ 33.52 MHz	68.18 kHz 至 95.23 kHz	
				fxx/10 (OCKSm = 13H)	fxx/440	30.00 MHz ≤ fxx ≤ 41.90 MHz	68.18 kHz 至 95.23 kHz	
0	0	0	1	fxx/4 (OCKSm = 10H)	fxx/344	24.00 MHz ≤ fxx ≤ 33.52 MHz	48.72 kHz 至 97.44 kHz	
				fxx/6 (OCKSm= 11H)	fxx/516	25.14 MHz ≤ fxx ≤ 48.00 MHz	48.72 kHz 至 93.02 kHz	
				fxx/8 (OCKSm = 12H)	fxx/688	33.52 MHz ≤ fxx ≤ 48.00 MHz	48.72 kHz 至 69.77 kHz	
				fxx/10 (OCKSm = 13H)	fxx/860	41.90 MHz ≤ fxx ≤ 48.00 MHz	48.72 kHz 至 55.81 kHz	
0	0	1	1	fxx/4 (OCKSm = 10H)	fxx/264	24.00 MHz ≤ fxx ≤ 25.60 MHz	90.91 kHz 至 96.97 kHz	
				fxx/6 (OCKSm = 11H)	fxx/396	38.40 MHz	96.97 kHz	
0	1	0	X	fxx/4 (OCKSm = 10H)	fxx/96	24.00 MHz ≤ fxx ≤ 33.52 MHz	250.00 kHz 至 349.17 kHz	高速模式 （SMCn=1）
				fxx/6 (OCKSm = 11H)	fxx/144	24.00 MHz ≤ fxx ≤ 48.00 MHz	166.67 kHz 至 333.33 kHz	
				fxx/8 (OCKSm = 12H)	fxx/192	32.00 MHz ≤ fxx ≤ 48.00 MHz	166.67 kHz 至 250.00 kHz	
				fxx/10 (OCKSm = 13H)	fxx/240	40.00 MHz ≤ fxx ≤ 48.00 MHz	166.67 kHz 至 200.00 kHz	
0	1	1	1	fxx/4 (OCKSm = 10H)	fxx/72	24.00 MHz ≤ fxx ≤ 25.60 MHz	333.33 kHz 至 355.56 kHz	
				fxx/6 (OCKSm= 11H)	fxx/108	38.40 MHz	355.56 kHz	
1	1	0	X	fxx/6 (OCKSm = 11H)	fxx/72	24.00 MHz ≤ fxx ≤ 25.14 MHz	333.33 kHz 至 349.17 kHz	
				fxx/8 (OCKSm = 12H)	fxx/96	32.00 MHz ≤ fxx ≤ 33.52 MHz	333.33 kHz 至 349.17 kHz	
				fxx/10 (OCKSm = 13H)	fxx/120	40.00 MHz ≤ fxx ≤ 41.90 MHz	333.33 kHz 至 349.17 kHz	
其它				禁止设置	—	—	—	—

备注 1. n=0 至 2, m=0, 1
2. x忽略

(7) IIC 分频时钟选择寄存器 0, 1 (OCS0, OCS1)

OCSm 寄存器控制 I²C0n 分频时钟 (n = 0 至 2, m = 0, 1)

这些寄存器通过 OCS0 寄存器控制 I²C00 分频时钟, 通过 OCS1 寄存器控制 I²C01 和 I²C02 分频时钟。

这些寄存器可以按 8 位读取或写入。

复位后这些寄存器设置为 00H。

复位后: 00H R/W 地址: OCS0 FFFFF340H, OCS1 FFFFF344H

	7	6	5	4	3	2	1	0
OCSm	0	0	0	OCSm	OCSm	0	OCSm	OCSm

(m = 0, 1)

OCSm	I ² C 分频时钟的工作设置
0	停止 I ² C 分频时钟工作
1	允许 I ² C 分频时钟工作

OCSm	OCSm	OCSm	I ² C 分频时钟的选择
0	0	0	f _{xx} /4
0	0	1	f _{xx} /6
0	1	0	f _{xx} /8
0	1	1	f _{xx} /10
1	0	0	f _{xx} /2

(8) IIC 移位寄存器 0 至 2 (IIC0 至 IIC2)

IICn 移位寄存器用作串行发送/接收 (移位操作), 并和串行时钟同步。 这些移位寄存器可以按 8 位读写, 但是, 在数据传输期间, 不应向 IICn 移位寄存器写入数据。

只有在等待期间, 可以访问 (读/写) IICn 移位寄存器。 在等待期间以外的其它通信状态时, 禁止访问这些寄存器。 然而, 对于主机设备, 只有在传输触发位 (IICn.STTn 位) 被置为 1 后, IICn 移位寄存器才可以写入一次。

通过在等待期间写入 IICn 寄存器释放一个等待状态, 开始时间传输 (n = 0 至 2)

复位后这些寄存器设置为 00H。

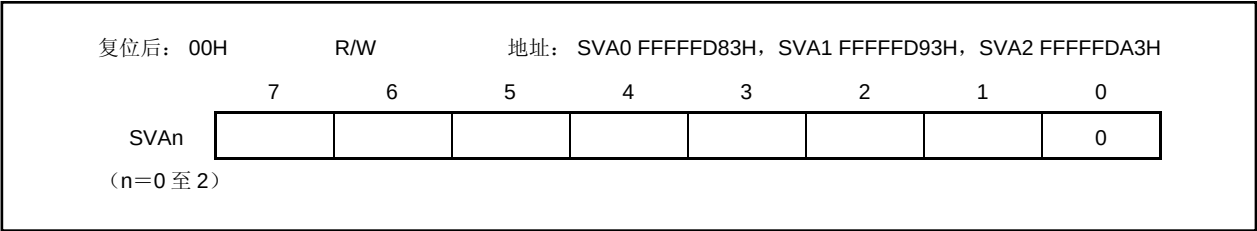
复位后: 00H R/W 地址: IIC0 FFFFFD80H, IIC1 FFFFFD90H, IIC2 FFFFFDA0H

	7	6	5	4	3	2	1	0
IICn								

(n=0 至 2)

(9) 从设备地址寄存器 0 至 2（SVA0 至 SVA2）

SVAn 寄存器保存 I²C 总线的从地址。
这些寄存器可按 8 位进行读写，但位 0 固定为 0。而 IICSn.STDn 位 = 1（开始条件检测）时，禁止重写这些寄存器。
复位后这些寄存器设置为 00H。



19.5 I²C 总线模式功能

19.5.1 引脚配置

串行时钟引脚（SCL0n）和串行数据总线引脚（SDA0n）结构如下（n = 0 至 2）。

SCL0n 本引脚用于串行时钟输入和输出。

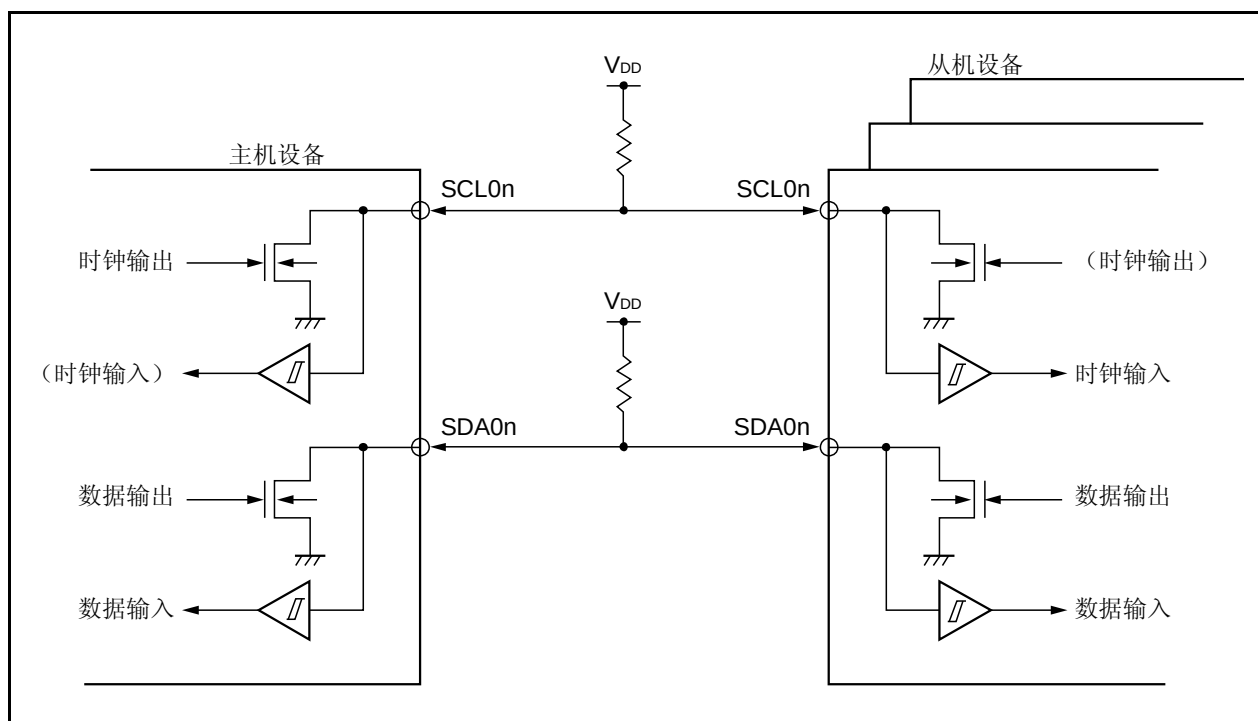
对于主机设备和从设备，本引脚都是 N 沟道漏极开路输出。输入为施密特输入。

SDA0n 本引脚用于串行数据输入和输出。

对于主机设备和从设备，本引脚都是 N 沟道漏极开路输出，输入为施密特输入。

由于串行时钟信号线和串行数据总线输出都是 N 沟道漏极开路输出，所以都需要外接上拉电阻。

图 19-6. 引脚结构配置图

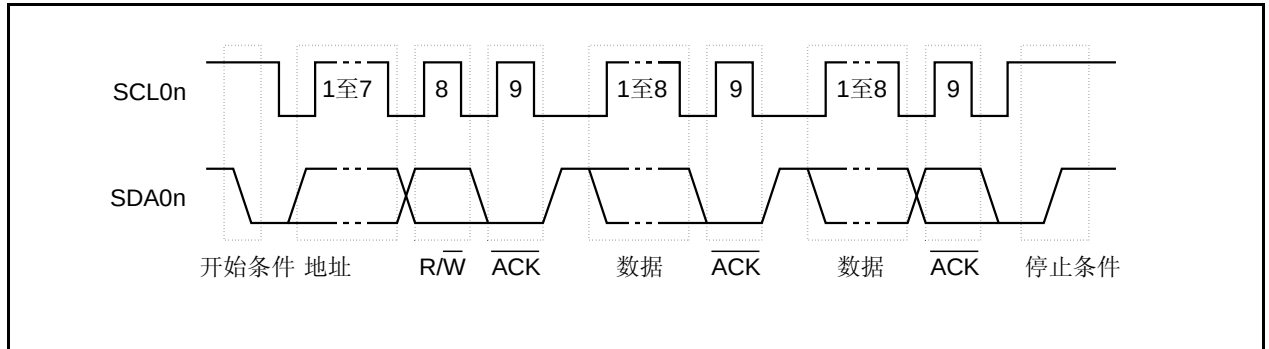


19.6 I²C 总线的定义和控制方法

以下部分描述 I²C 总线的串行数据通信格式和由 I²C 总线使用的信号。

关于 I²C 总线串行数据总线产生的“开始条件”、“地址”、“传输方向”、“数据”以及“停止条件”的传输时序，如下所示。

图 19-7. I²C 总线串行数据传输时序



主机设备产生开始条件、从机地址和停止条件。

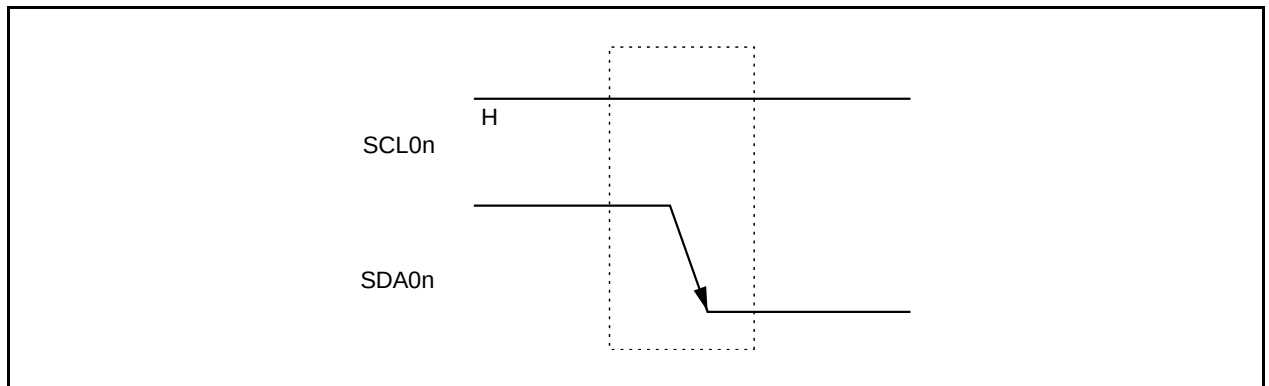
ACK 信号既可由主机产生，也可由从机产生。（通常情况下，它由接收 8 位数据的一方产生）。

串行时钟（SCL0n）由主机设备持续输出。然而，在从机设备 SCL0n 引脚的低电平时段可被延长，插入等待状态（n = 0 至 2）。

19.6.1 开始条件

当 SCL0n 引脚为高电平而且 SDA0n 引脚信号由高电平变为低电平时，形成开始条件。SCL0n 和 SDA0n 引脚的开始条件是开始串行传输时主机设备输出至从机设备的信号。从机设备可检测开始条件（n = 0 至 2）。

图 19-8. 开始条件



当检测到停止条件后（IICSn.SPDn 位 = 1）设置 IICC0.STT0 位为 1 时，输出一次开始条件。当检测到开始条件时，IICSn.STDn 位被置位为 1（n = 0 至 2）。

注意事项 在和其它设备进行通信时，当 V850ES/JG3-H 和 V850ES/JH3-H 的 IICn.IICEn 位被设置为 1 时，开始条件的检测依赖于通信序列的状态，SCL0n 和 SDA0n 信号线是高电平时确保将 IICn.IICEn 位设置为 1。

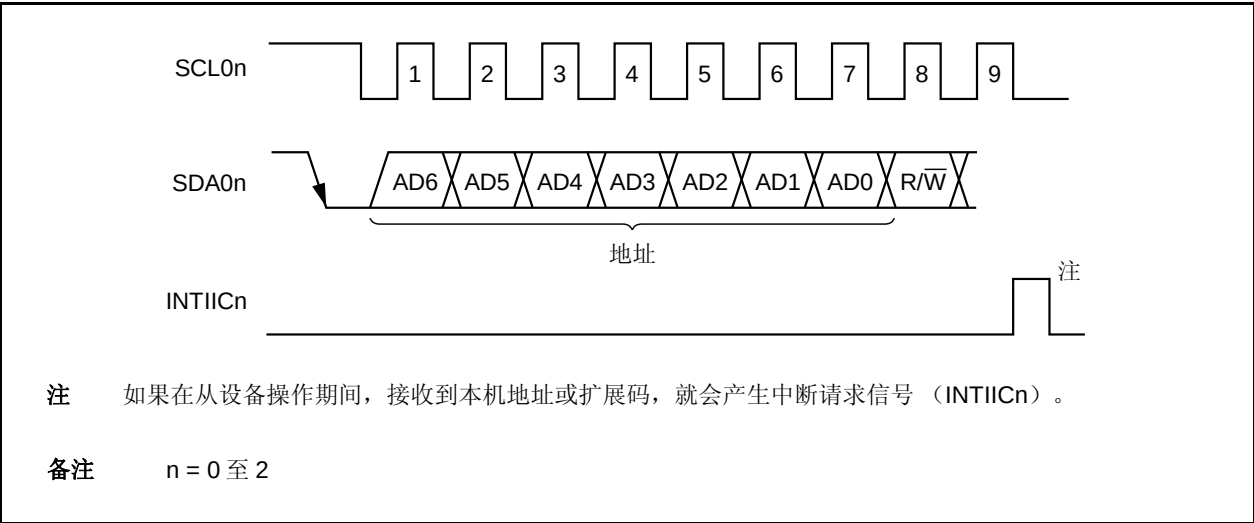
19.6.2 地址

紧随开始条件之后的 7 位数据定义为地址。

地址为 7 位数据字段输出，因此主机设备可以选择连接到总线序列的从机设备。 所以，每一个接入总线的从机设备必须具有一个独一无二的地址。

从机设备通过硬件检测开始条件，并检查 7 位地址数据是否和 SVAn 寄存器内储存的数据值匹配。 如果地址数据和 SVAn 寄存器内储存的数据值相匹配，从设备就被选择并且和主机进行通信，直到主机设备产生一个开始条件或结束条件为止（n = 0 至 2）。

图 19-9. 地址



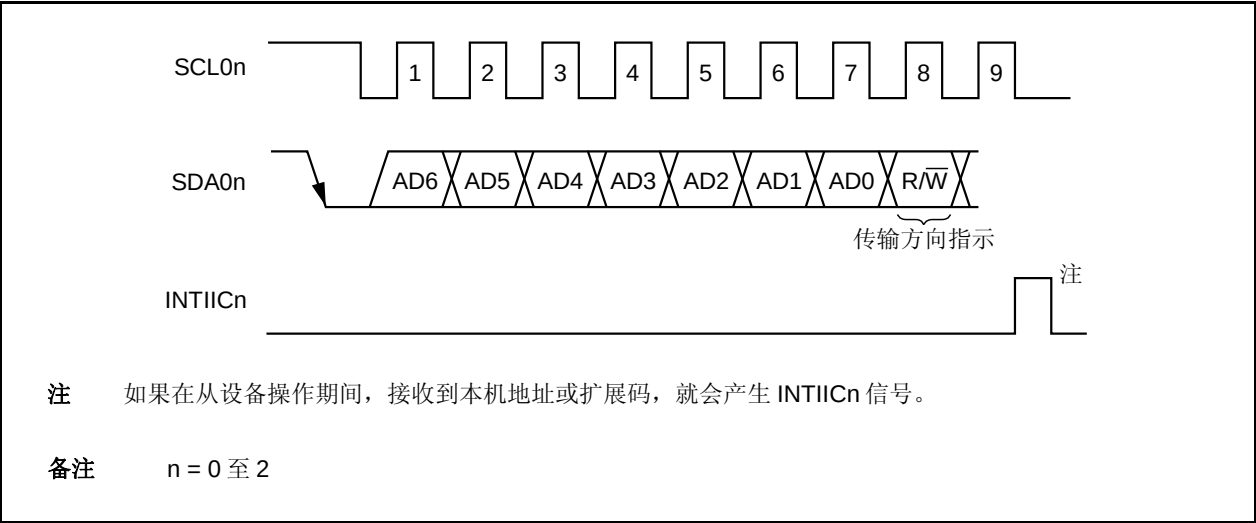
当 19.6.3 传输方向指示 中说明的从机地址和传输方向同时作为 8 位数据写入 IICn 寄存器时，会输出一个地址。 接收到的地址数据被写入 IICn 寄存器（n = 0 至 2）。

从机地址位于 IICn 寄存器的高 7 位。

19.6.3 传输方向指示

除了 7 位地址数据之外，主机设备发送 1 位数据指示传输方向。
当此传输方向指示位为 0 时，指示主机设备向从机设备传输数据。 当此传输方向指示位为 1 时，指示主机设备由从机设备接收数据。

图 19-10. 传输方向指示



19.6.4 ACK

$\overline{\text{ACK}}$ 用来确认发送设备和接收设备的串行数据传输状态。

接收设备对接收到的每 8 位数据应答一次 $\overline{\text{ACK}}$ 信号。

通常情况下，发送设备发送 8 位数据后收到一个 $\overline{\text{ACK}}$ 信号。当 $\overline{\text{ACK}}$ 信号由接收端返回后，被判定为接收正常并且操作继续进行。使用 IICSn.ACKDn 位对 $\overline{\text{ACK}}$ 信号检测进行确认。

当主机为接收设备端时，收到最后一个数据以后，不返回 $\overline{\text{ACK}}$ 信号并产生停止条件。当从机为接收设备而且不返回 $\overline{\text{ACK}}$ 信号时，主机要么产生一个停止条件，要么产生一个重新开始条件，之后，停止当前的通信传输。返回 $\overline{\text{ACK}}$ 失败的情况可由以下因素导致：

- (a) 接收工作异常。
- (b) 接收到最后一个数据。
- (c) 指定的接收设备（从机）地址不存在。

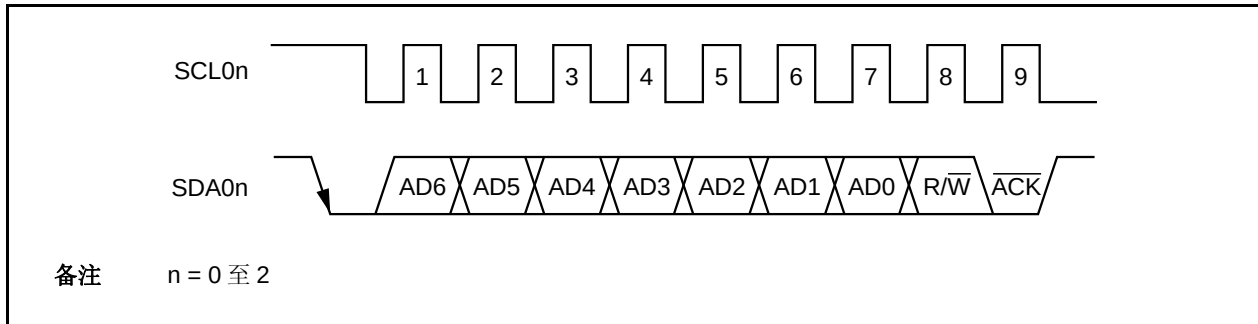
当第 9 个时钟期间接收设备将 SDA0n 信号线设置为低电平，产生 $\overline{\text{ACK}}$ 信号（接收正常）。

当 IICCN.ACKEn 位设置为 1 时，自动 $\overline{\text{ACK}}$ 产生使能。7 位地址之后的第 8 位发送导致 IICSn.TRCn 位被设置。通常情况下，设置 ACKE0 位为 1 作为应答（TRCn 位 = 0）。

当从机设备进行接收（当 TRCn 位 = 0 时）时，如果从机设备不能接收数据或不再需要接收连续的数据，将 ACKEn 位清除为 0，指示主机不能接收更多数据了。

与上述相似，当主机设备进行接收（当 TRCn 位 = 0）时，并且不再需要后续数据，将 ACKEn 位清除为 0 来阻断 $\overline{\text{ACK}}$ 信号产生。通知从机设备（发送设备）数据发送结束（发送停止）。

图 19-11. $\overline{\text{ACK}}$



当本机地址收到后，不管 ACKEn 位的设定值如何， $\overline{\text{ACK}}$ 信号都自动产生。如果收到的地址不是本机地址（NACK），不产生 $\overline{\text{ACK}}$ 信号。

当接收扩展码时，在产生 $\overline{\text{ACK}}$ 信号之前，将 ACKEn 位设置为 1。

数据接收期间 $\overline{\text{ACK}}$ 的产生方式基于等待时序设置，描述如下：

- 当选择 8 时钟等待时（IICCN.WTIMn 位 = 0）：
如果等待状态取消前将 ACKEn 位设置为 1， $\overline{\text{ACK}}$ 信号在 SCL0n 引脚第 8 个时钟下降沿产生的同时产生。
- 当选择 9 时钟等待时（IICCN.WTIMn 位 = 1）：
如果 ACKEn 位预先被设置为 1，产生 $\overline{\text{ACK}}$ 信号。

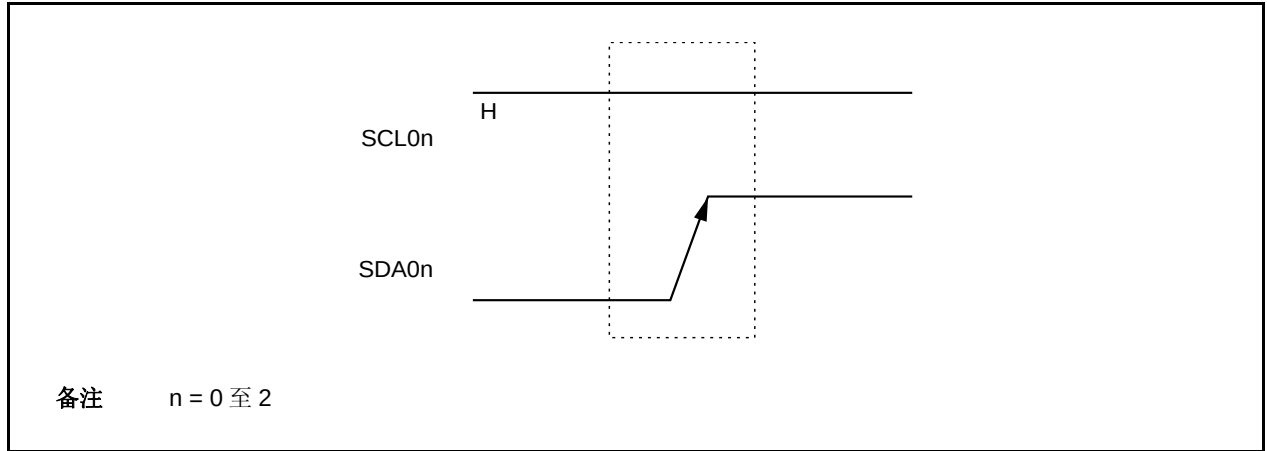
备注 n = 0 至 2

19.6.5 停止条件

当 SCL0n 引脚为高电平时，将 SDA0n 引脚由低电平变为高电平即产生一个停止条件（n = 0 至 2）。

当从主机设备向从机设备的串行传输完毕以后，产生一个停止条件，当 V850ES/JG3-H 或 V850ES/JH3-H 用作从机设备时，不能检测停止条件。

图 19-12. 停止条件



当 IICn.SPTn 位设为 1 时会产生停止条件。检测到停止条件时，IICSn.SPDn 位被设为 1 且在 IICn.SPIEn 位（n = 0 至 2）设为 1 时产生中断请求信号（INTIICn）。

19.6.6 等待状态

等待状态用来通知通信伙伴设备（主机或从机）已经准备就绪（即：处于等待状态），可以进行收发数据。

将 SCL0n 引脚设为低电平通知通信伙伴本机处于等待状态。当主机设备和从机设备的等待状态都被取消时，下一次传输就可以开始了（n = 0 至 2）。

图 19-13. 等待状态（1/2）

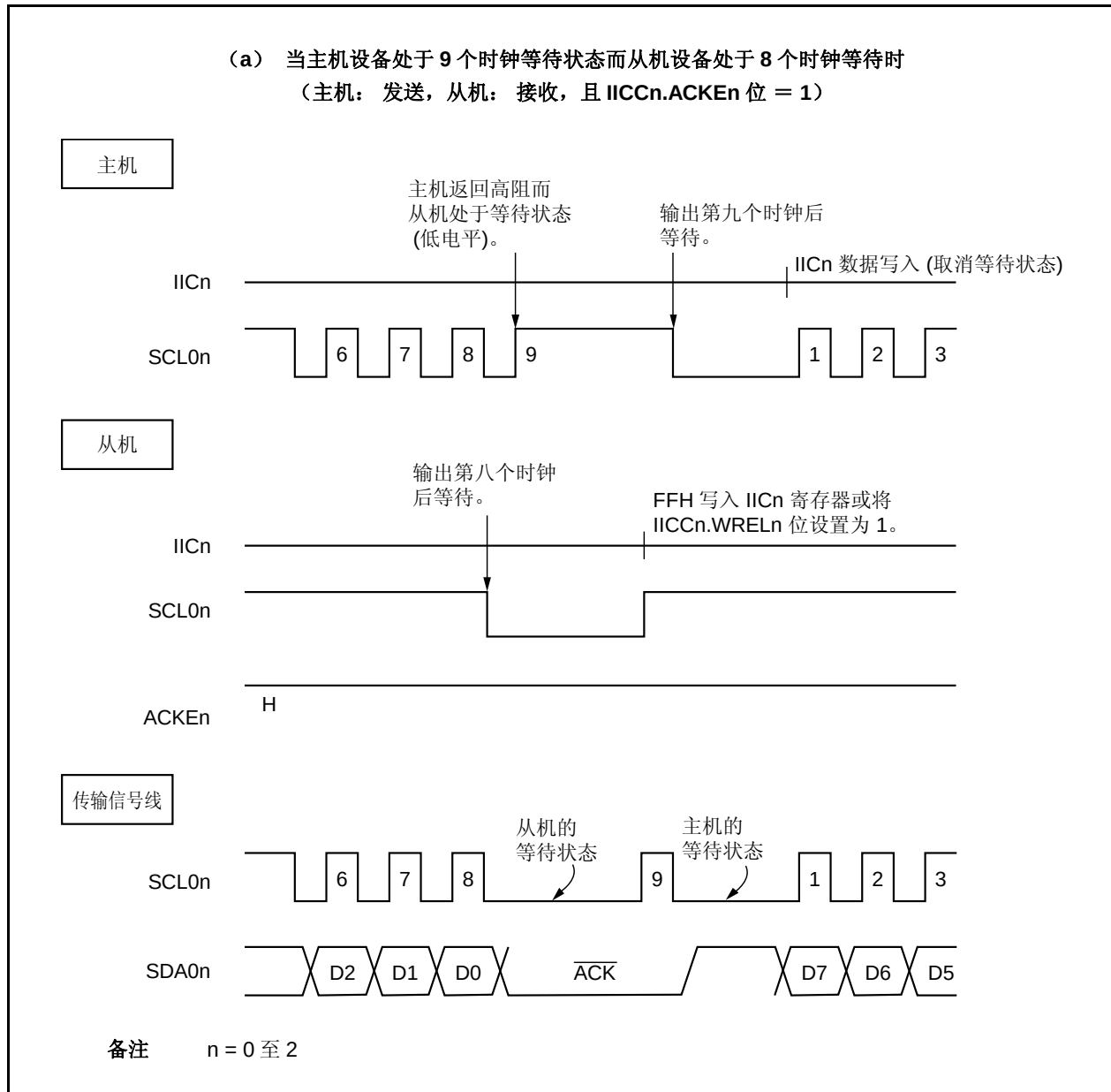
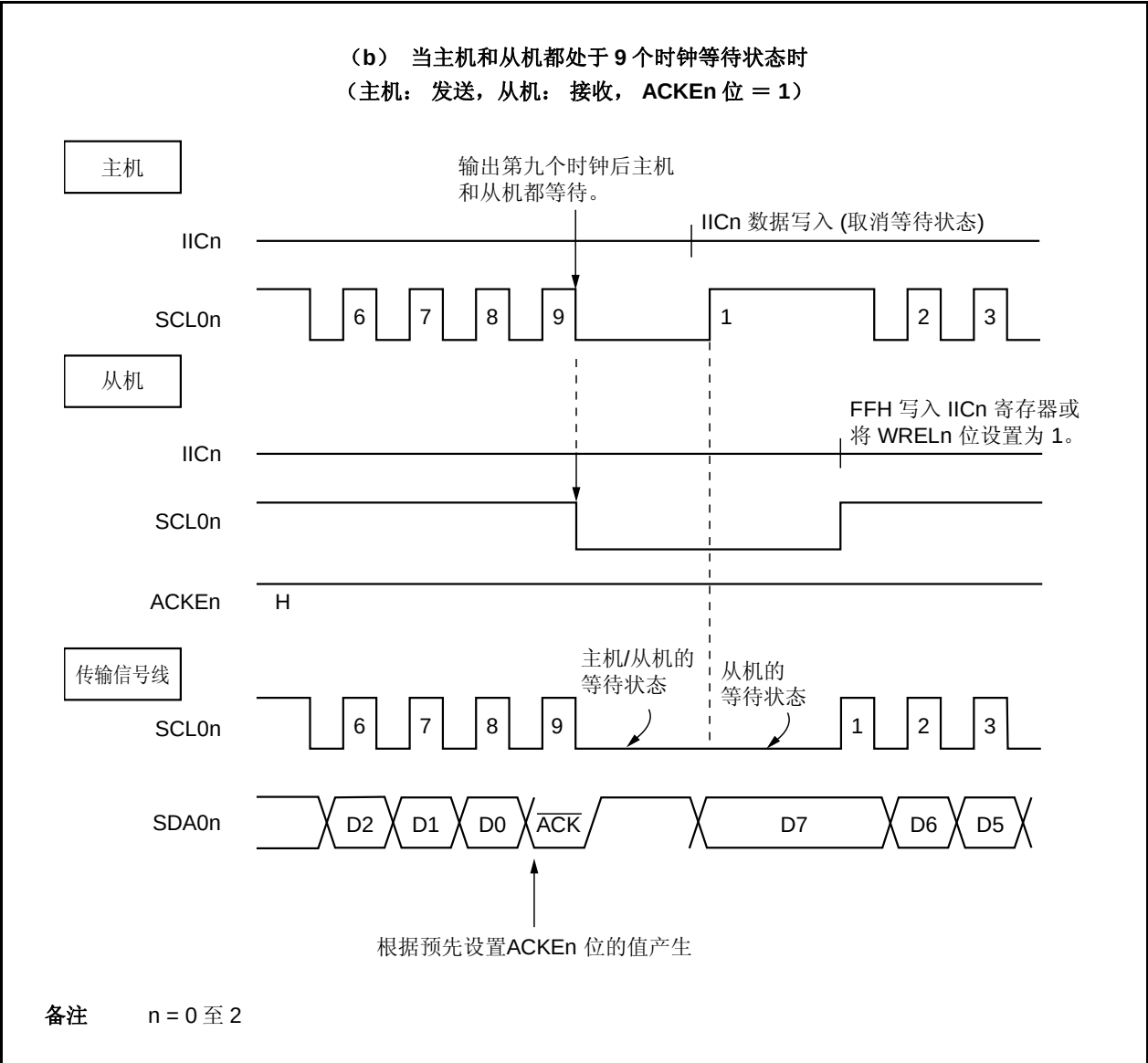


图 19-13. 等待状态 (2/2)



等待状态可依照 IICn.WTIMn 位的设置自动产生 (n = 0 至 2)。

通常, 接收端会在 IICn.WRELn 设为 1 或将 FFh 写入 IICn 寄存器时取消等待状态, 而发送端在将数据写入 IICn 寄存器时取消等待状态。

主机设备可以通过以下途径取消等待状态。

- 将 IICn.STTn 位置为 1。
- 将 IICn.SPTn 位置为 1。

19.6.7 等待状态取消方法

在使用 I²C0n 的情况下，一般使用以下方式取消等待状态（n = 0 至 2）。

- 通过向 IICn 寄存器写入数据
- 通过将 IICCn.WRELn 位置为 1（取消等待状态）
- 通过将 IICCn.STTn 位置为 1（产生开始条件）
- 通过将 IICCn.SPTn 位置为 1（产生停止条件）

如果任何一种等待取消操作被执行，I²C0n 总线将取消等待状态并重新开始通信。

当取消等待状态并开始发送数据（包括地址）时，向 IICn 寄存器写入数据。

为了在等待状态取消后接收数据，或结束数据发送，将 WRELn 位置为 1。

要在等待状态取消后产生一个重新开始条件，将 STTn 位设置为 1。

要在等待状态取消后产生一个停止条件，将 SPTn 位设置为 1。

对每一个等待状态只可执行一次取消操作。

例如，如果通过将 WRELn 位设置为 1 取消等待状态，而后数据写入 IICn 寄存器，SDA0n 信号线的时序变化和 IICn 寄存器的写入时序可能发生冲突，导致 SDA0n 信号线上的输出数据错误。

即便在其它操作情况下，如果通信中途停止，将 IICCn.IICEn 位清 0 也将停止通信，取消等待使能。

如果 I²C 总线因噪声等原因发生死锁，将 IICCn.LRELn 位设置为 1，退出通信操作，取消等待状态使能。

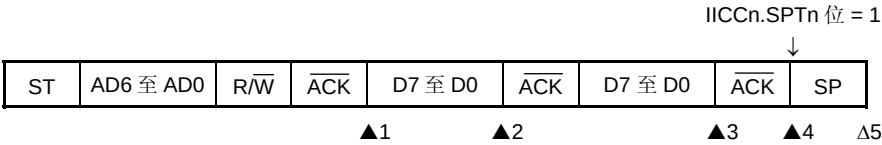
19.7 I²C中断请求信号（INTIICn）

以下显示了 INTIICn 中断请求信号产生时序和 INTIICn 信号产生时序中 IICSn 寄存器的值（n = 0 至 2）。

19.7.1 主机设备操作

(1) 开始 ~ 地址 ~ 数据 ~ 数据 ~ 停止（正常发送/接收）

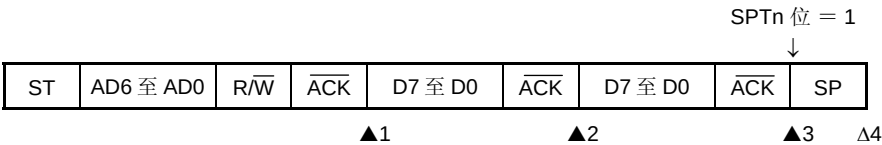
<1> 当 IICn.WTIMn 位 = 0 时



- ▲1: IICSn 寄存器 = 1000X110B
- ▲2: IICSn 寄存器 = 1000X000B
- ▲3: IICSn 寄存器 = 1000X000B (WTIMn 位 = 1)
- ▲4: IICSn 寄存器 = 1000XX00B
- Δ 5: IICSn 寄存器 = 00000001B

备注 1. ▲: 总是产生
 Δ: 仅当 SPIEn 位 = 1 时产生
 X: 忽略
 2. n = 0 至 2

<2> 当 WTIMn 位 = 1 时

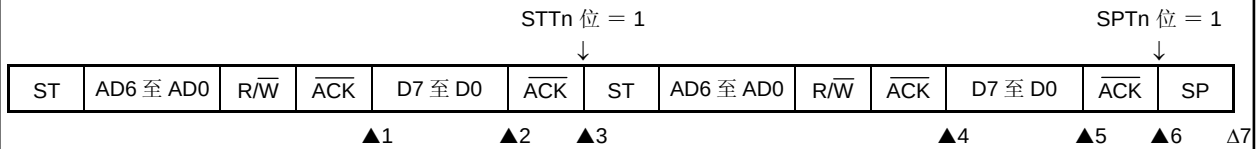


- ▲1: IICSn 寄存器 = 1000X110B
- ▲2: IICSn 寄存器 = 1000X100B
- ▲3: IICSn 寄存器 = 1000XX00B
- Δ 4: IICSn 寄存器 = 00000001B

备注 1. ▲: 总是产生
 Δ: 仅当 SPIEn 位 = 1 时产生
 X: 忽略
 2. n = 0 至 2

(2) 开始 ~ 地址 ~ 数据 ~ 开始 ~ 地址 ~ 数据 ~ 停止 (重新开始)

<1> 当 WTIMn 位 = 0 时

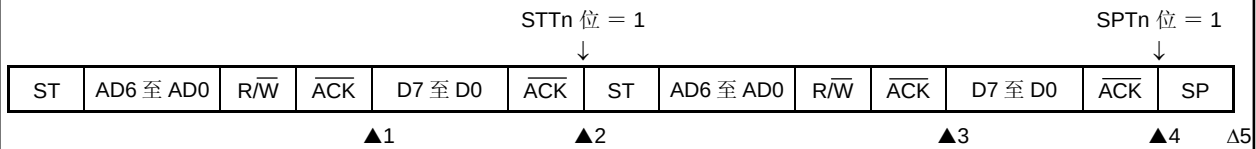


- ▲1: IICSn 寄存器 = 1000X110B
- ▲2: IICSn 寄存器 = 1000X000B (WTIMn 位 = 1)
- ▲3: IICSn 寄存器 = 1000XX00B (WTIMn 位 = 0)
- ▲4: IICSn 寄存器 = 1000X110B (WTIMn 位 = 0)
- ▲5: IICSn 寄存器 = 1000X000B (WTIMn 位 = 1)
- ▲6: IICSn 寄存器 = 1000XX00B
- Δ 7: IICSn 寄存器 = 00000001B

备注

- 1. ▲: 总是产生
Δ: 仅当 SPIEn 位 = 1 时产生
X: 忽略
- 2. n = 0 至 2

<2> 当 WTIMn 位 = 1 时



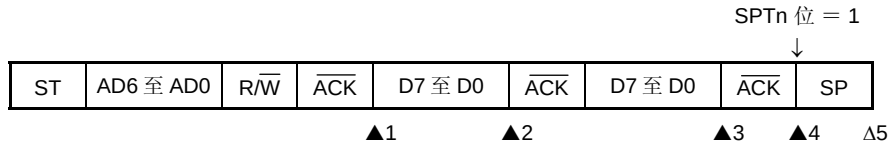
- ▲1: IICSn 寄存器 = 1000X110B
- ▲2: IICSn 寄存器 = 1000XX00B
- ▲3: IICSn 寄存器 = 1000X110B
- ▲4: IICSn 寄存器 = 1000XX00B
- Δ 5: IICSn 寄存器 = 00000001B

备注

- 1. ▲: 总是产生
Δ: 仅当 SPIEn 位 = 1 时产生
X: 忽略
- 2. n = 0 至 2

(3) 开始~扩展码~数据~数据~停止（发送扩展码）

<1> 当 WTIMn 位 = 0 时



▲1: IICSn 寄存器 = 1010X110B

▲2: IICSn 寄存器 = 1010X000B

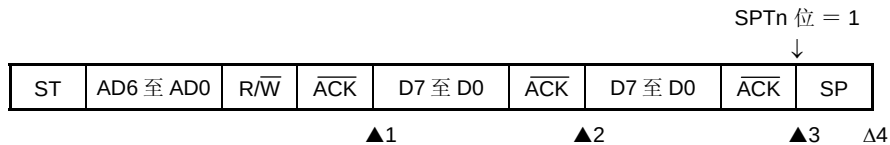
▲3: IICSn 寄存器 = 1010X000B (WTIMn 位 = 1)

▲4: IICSn 寄存器 = 1010XX00B

Δ 5: IICSn 寄存器 = 00000001B

- 备注**
1. ▲: 总是产生
Δ: 仅当 SPIEn 位 = 1 时产生
X: 忽略
 2. n = 0 至 2

<2> 当 WTIMn 位 = 1 时



▲1: IICSn 寄存器 = 1010X110B

▲2: IICSn 寄存器 = 1010X100B

▲3: IICSn 寄存器 = 1010XX00B

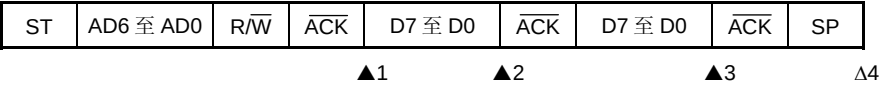
Δ 4: IICSn 寄存器 = 00000001B

- 备注**
1. ▲: 总是产生
Δ: 仅当 SPIEn 位 = 1 时产生
X: 忽略
 2. n = 0 至 2

19.7.2 从设备操作（当接收从机地址数据（地址匹配）时）

(1) 开始 ~ 地址 ~ 数据 ~ 数据 ~ 停止

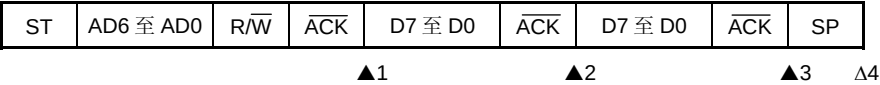
<1> 当 IICn.WTIMn 位 = 0 时



- ▲1: IICSn 寄存器 = 0001X110B
- ▲2: IICSn 寄存器 = 0001X000B
- ▲3: IICSn 寄存器 = 0001X000B
- Δ 4: IICSn 寄存器 = 00000001B

备注 1. ▲: 总是产生
 Δ: 仅当 IICn.SPIEn 位 = 1 时产生
 X: 忽略
 2. n = 0 至 2

<2> 当 WTIMn 位 = 1 时

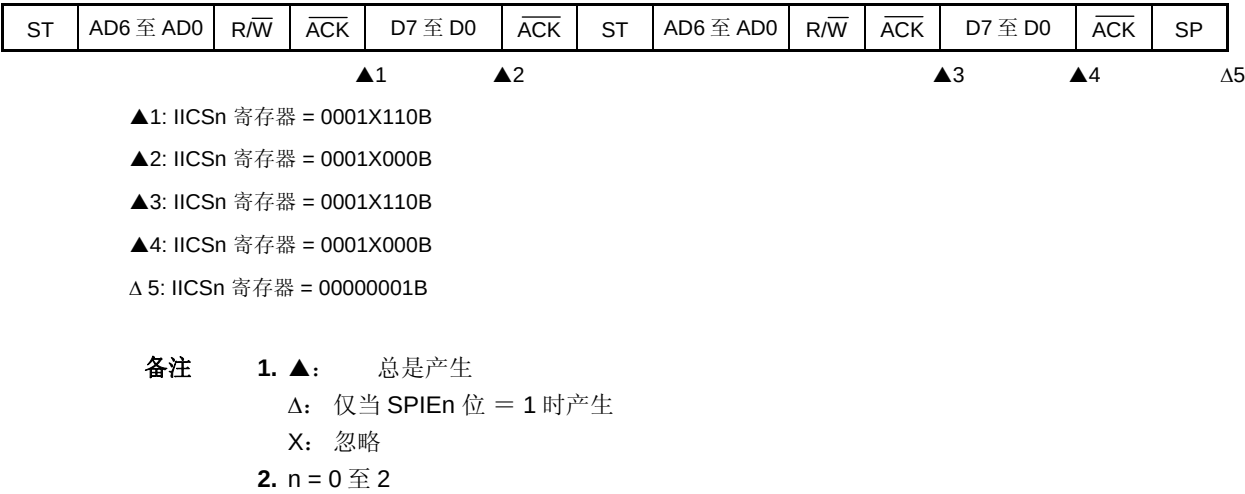


- ▲1: IICSn 寄存器 = 0001X110B
- ▲2: IICSn 寄存器 = 0001X100B
- ▲3: IICSn 寄存器 = 0001XX00B
- Δ 4: IICSn 寄存器 = 00000001B

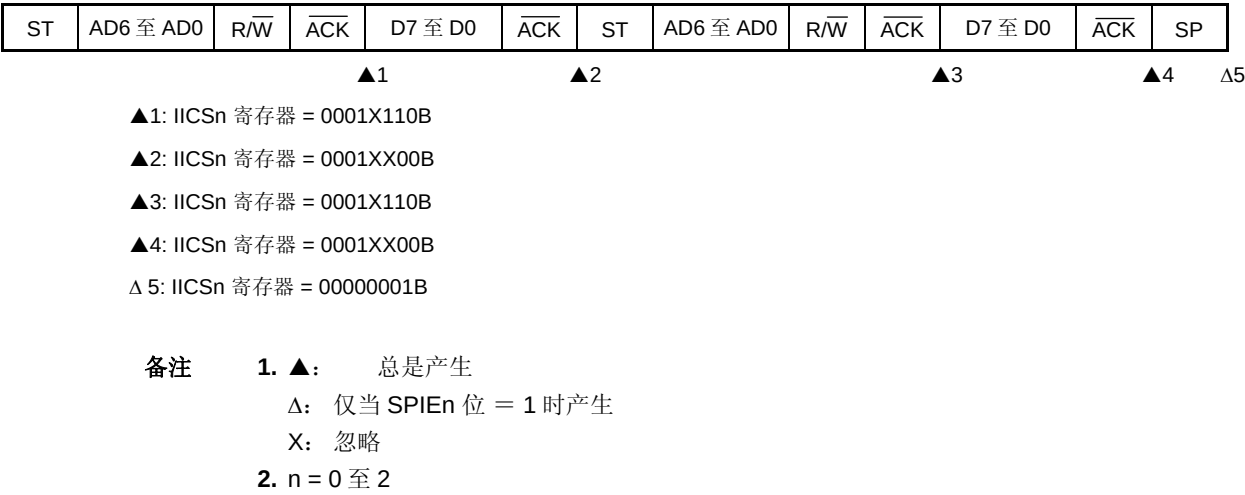
备注 1. ▲: 总是产生
 Δ: 仅当 SPIEn 位 = 1 时产生
 X: 忽略
 2. n = 0 至 2

(2) 开始 ~ 地址 ~ 数据 ~ 开始 ~ 地址 ~ 数据 ~ 停止

<1> 当 WTIMn 位 = 0（重新开始后，地址匹配）时

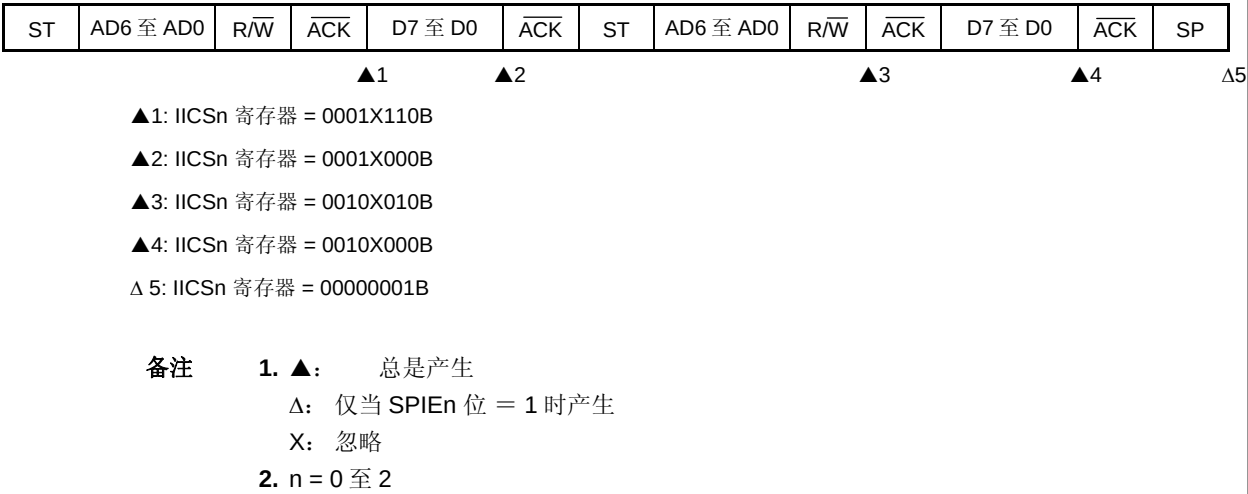


<2> 当 WTIMn 位 = 1（重新开始后，地址匹配）时

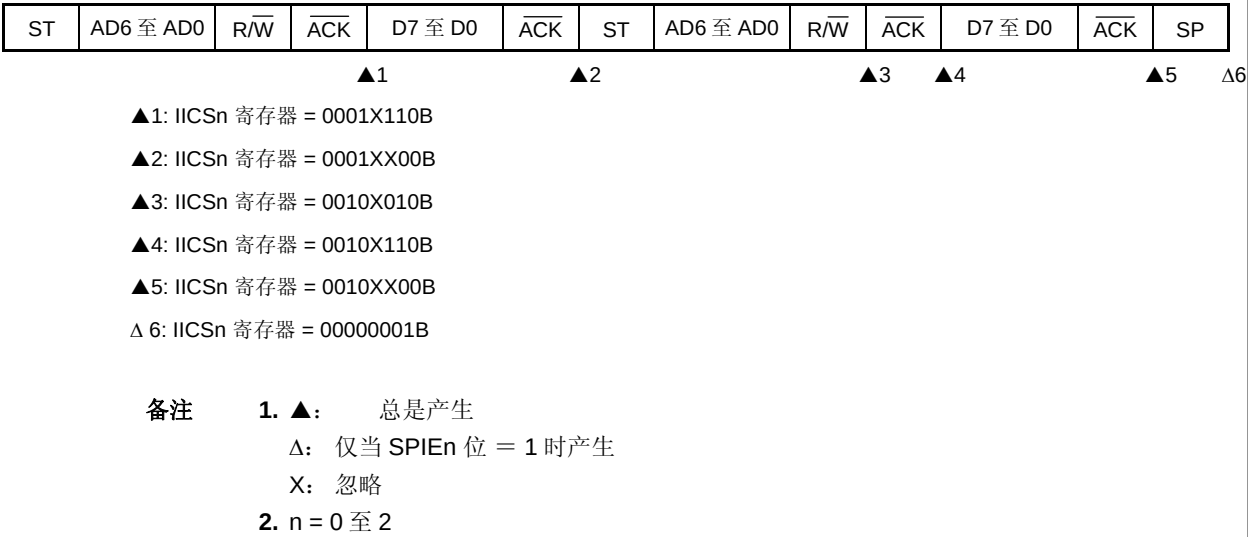


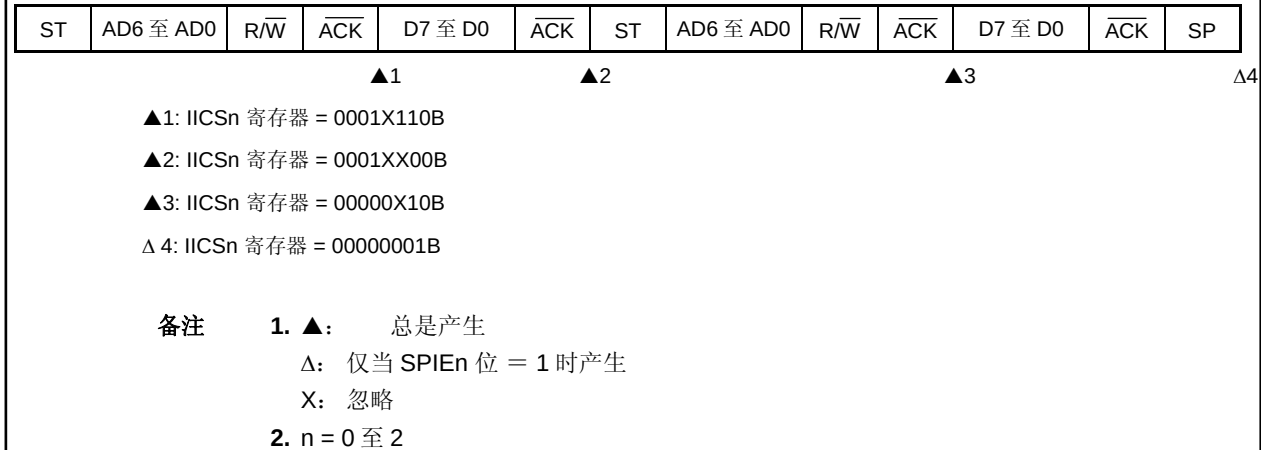
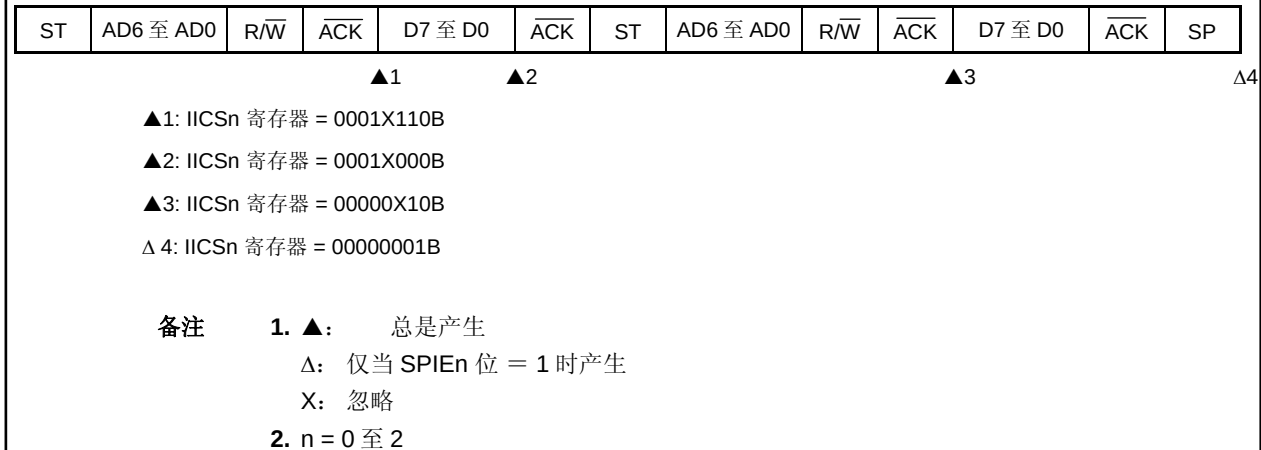
(3) 开始 ~ 地址 ~ 数据 ~ 开始 ~ 扩展码 ~ 数据 ~ 停止

<1> 当 WTIMn 位 = 0（重新开始后，扩展码接收）时



<2> 当 WTIMn 位 = 1（重新开始后，扩展码接收）时

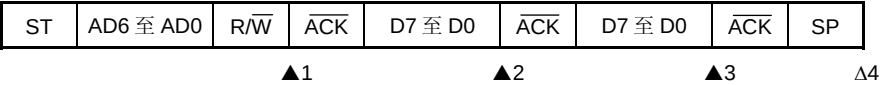




19.7.3 从机设备操作（当接收扩展码时）

(1) 开始 ~ 扩展码 ~ 数据 ~ 数据 ~ 停止

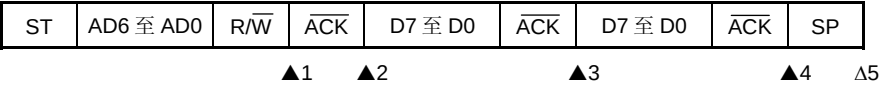
<1> 当 IICn.WTIMn 位 = 0 时



- ▲1: IICSn 寄存器 = 0010X010B
- ▲2: IICSn 寄存器 = 0010X000B
- ▲3: IICSn 寄存器 = 0010X000B
- Δ 4: IICSn 寄存器 = 00000001B

备注 1. ▲: 总是产生
 Δ: 仅当 IICn.SPIEn 位 = 1 时产生
 X: 忽略
 2. n = 0 至 2

<2> 当 WTIMn 位 = 1 时

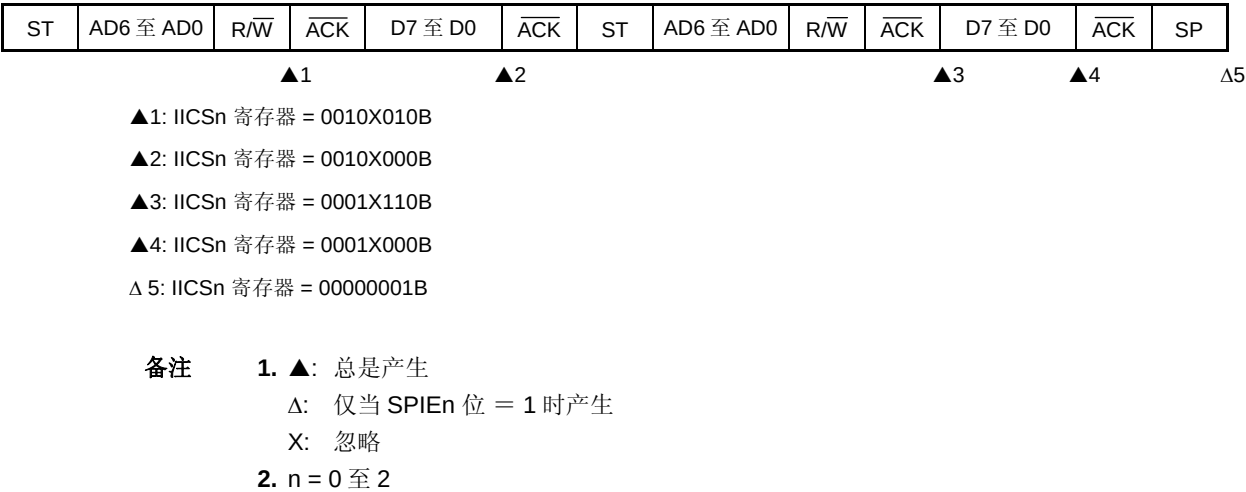


- ▲1: IICSn 寄存器 = 0010X010B
- ▲2: IICSn 寄存器 = 0010X110B
- ▲3: IICSn 寄存器 = 0010X100B
- ▲4: IICSn 寄存器 = 0010XX00B
- Δ 5: IICSn 寄存器 = 00000001B

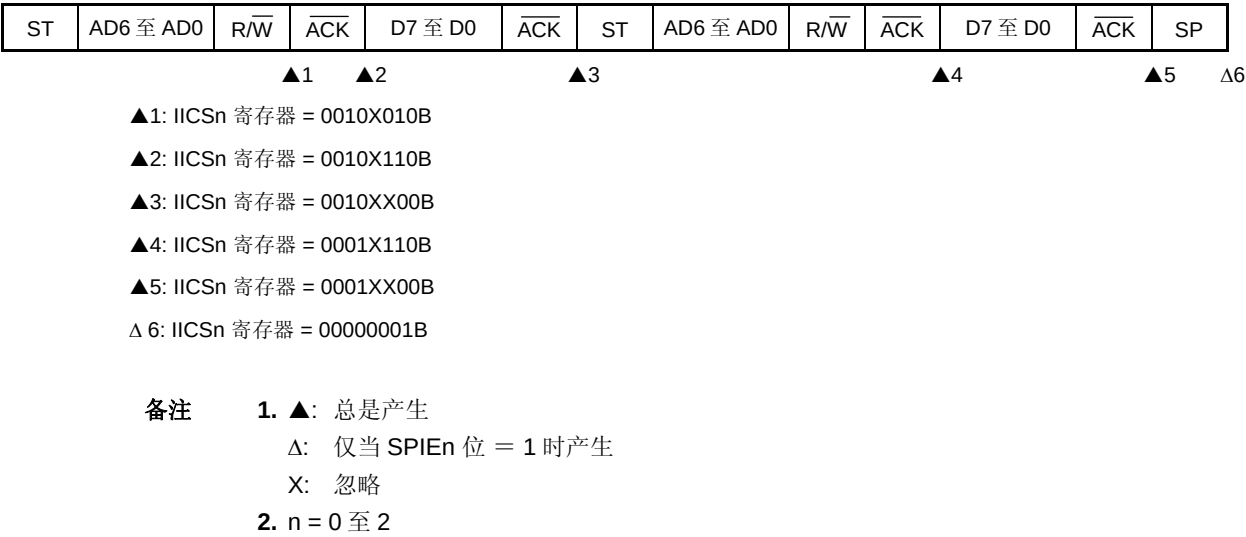
备注 1. ▲: 总是产生
 Δ: 仅当 SPIEn 位 = 1 时产生
 X: 忽略
 2. n = 0 至 2

(2) 开始 ~ 扩展码 ~ 数据 ~ 开始 ~ 地址 ~ 数据 ~ 停止

<1> 当 WTIMn 位 = 0（重新开始后，地址匹配）时

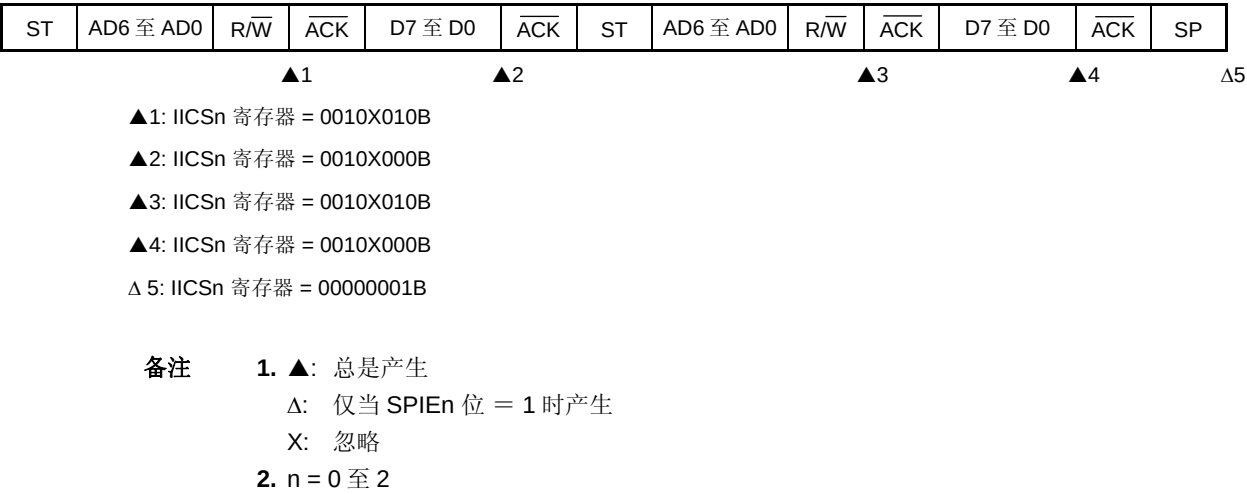


<2> 当 WTIMn 位 = 1（重新开始后，地址匹配）时

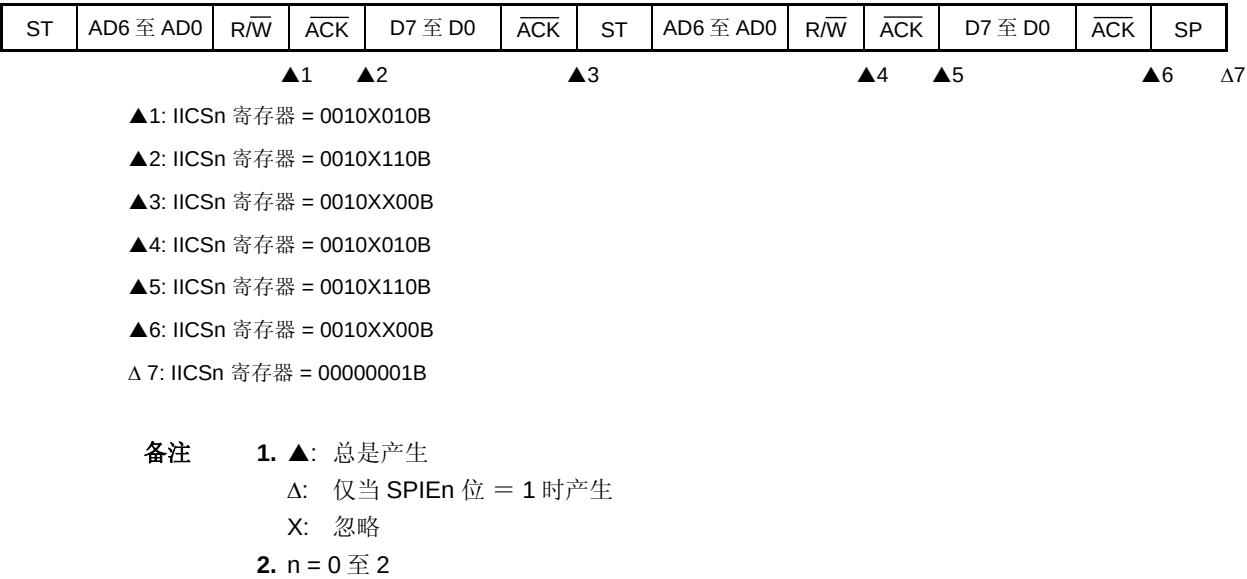


(3) 开始 ~ 扩展码 ~ 数据 ~ 开始 ~ 扩展码 ~ 数据 ~ 停止

<1> 当 WTIMn 位 = 0（重新开始后，扩展码接收）时

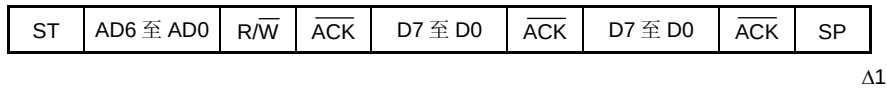


<2> 当 WTIMn 位 = 1（重新开始后，扩展码接收）时



19.7.4 不通信时的操作

(1) 开始 ~ 扩展码 ~ 数据 ~ 数据 ~ 停止



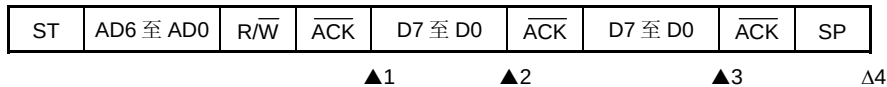
Δ 1: IICSn 寄存器 = 00000001B

- 备注
1. Δ: 仅当 SPIEn 位 = 1 时产生
 2. n = 0 至 2

19.7.5 仲裁失败操作（从设备在仲裁失败后的操作）

(1) 当在从机地址数据传输期间发生仲裁失败

<1> 当 IICn.WTIMn 位 = 0 时



▲1: IICSn 寄存器 = 0101X110B（例如：在中断服务期间读取 IICSn.ALDn 位）

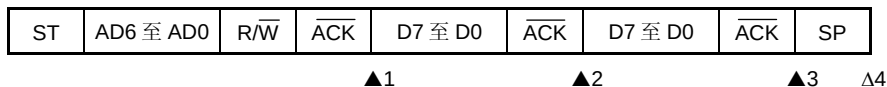
▲2: IICSn 寄存器 = 0001X000B

▲3: IICSn 寄存器 = 0001X000B

Δ 4: IICSn 寄存器 = 00000001B

- 备注
1. ▲: 总是产生
 Δ: 仅当 IICn.SPIEn 位 = 1 时产生
 X: 忽略
 2. n = 0 至 2

<2> 当 WTIMn 位 = 1 时



▲1: IICSn 寄存器 = 0101X110B（例如：在中断服务期间读取 ALDn 位）

▲2: IICSn 寄存器 = 0001X100B

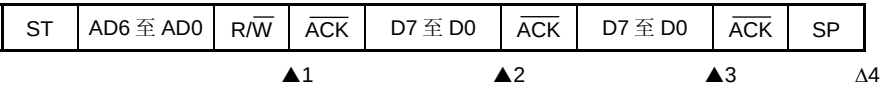
▲3: IICSn 寄存器 = 0001XX00B

Δ 4: IICSn 寄存器 = 00000001B

- 备注
1. ▲: 总是产生
 Δ: 仅当 SPIEn 位 = 1 时产生
 X: 忽略
 2. n = 0 至 2

(2) 当在传输扩展码期间发生仲裁失败

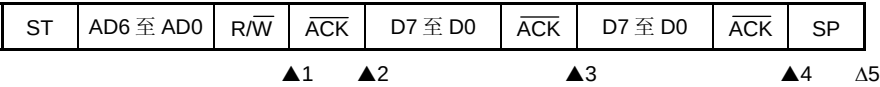
<1> 当 WTIMn 位 = 0 时



- ▲1: IICSn 寄存器 = 0110X010B (例如: 在中断服务期间读取 ALDn 位)
- ▲2: IICSn 寄存器 = 0010X000B
- ▲3: IICSn 寄存器 = 0010X000B
- Δ 4: IICSn 寄存器 = 00000001B

备注 1. ▲: 总是产生
 Δ: 仅当 SPIEn 位 = 1 时产生
 X: 忽略
 2. n = 0 至 2

<2> 当 WTIMn 位 = 1 时



- ▲1: IICSn 寄存器 = 0110X010B (例如: 在中断服务期间读取 ALDn 位)
- ▲2: IICSn 寄存器 = 0010X110B
- ▲3: IICSn 寄存器 = 0010X100B
- ▲4: IICSn 寄存器 = 0010XX00B
- Δ 5: IICSn 寄存器 = 00000001B

备注 1. ▲: 总是产生
 Δ: 仅当 SPIEn 位 = 1 时产生
 X: 忽略
 2. n = 0 至 2

19.7.6 当仲裁失败发生时的操作（在仲裁失败后不进行通信）

(1) 当在从机地址数据传输期间发生仲裁失败



▲1 Δ2

▲1: IICSn 寄存器 = 01000110B（例如：在中断期间读取 IICSn.ALDn 位时）
Δ 2: IICSn 寄存器 = 00000001B

备注 1. ▲：总是产生
 Δ：仅当 IICn.SPIEn 位 = 1 时产生
 2. n = 0 至 2

(2) 当在传输扩展码期间发生仲裁失败



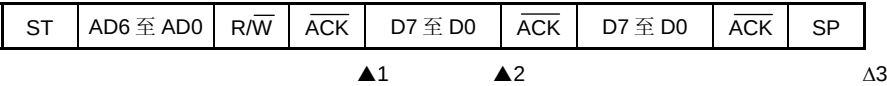
▲1 Δ2

▲1: IICSn 寄存器 = 0110X010B（例如：在中断服务期间读取 ALDn 位时）
通过软件将 IICn.LRELn 位设为 1
Δ 2: IICSn 寄存器 = 00000001B

备注 1. ▲：总是产生
 Δ：仅当 SPIEn 位 = 1 时产生
 X：忽略
 2. n = 0 至 2

(3) 当在数据传输期间发生仲裁失败

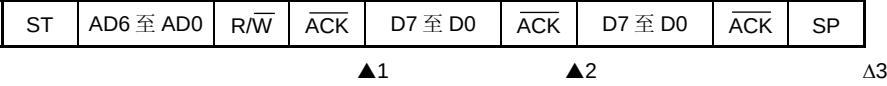
<1> 当 IICSn.WTIMn 位 = 0 时



- ▲1: IICSn 寄存器 = 10001110B
▲2: IICSn 寄存器 = 01000000B (例如: 在中断服务期间读取 ALDn 位时)
Δ 3: IICSn 寄存器 = 00000001B

备注 1. ▲: 总是产生
 Δ: 仅当 SPIEn 位 = 1 时产生
 2. n = 0 至 2

<2> 当 WTIMn 位 = 1 时



- ▲1: IICSn 寄存器 = 10001110B
▲2: IICSn 寄存器 = 01000100B (例如: 在中断服务期间读取 ALDn 位时)
Δ 3: IICSn 寄存器 = 00000001B

备注 1. ▲: 总是产生
 Δ: 仅当 SPIEn 位 = 1 时产生
 2. n = 0 至 2

(4) 当在数据传输期间，由于重新开始条件导致发生仲裁失败

<1> 无扩展码（例如：地址不匹配）

ST	AD6 至 AD0	R/W	ACK	D7 至 Dn	ST	AD6 至 AD0	R/W	ACK	D7 至 D0	ACK	SP

▲1

▲2

Δ3

▲1: IICSn 寄存器 = 1000X110B

▲2: IICSn 寄存器 = 01000110B（例如：在中断服务期间读取 ALDn 位时）

Δ 3: IICSn 寄存器 = 00000001B

- 备注
1. ▲：总是产生
Δ：仅当 SPIEn 位 = 1 时产生
X：忽略
 2. Dn = D6 至 D0
n = 0 至 2

<2> 扩展代码

ST	AD6 至 AD0	R/W	ACK	D7 至 Dn	ST	AD6 至 AD0	R/W	ACK	D7 至 D0	ACK	SP

▲1

▲2

Δ3

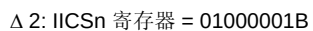
▲1: IICSn 寄存器 = 1000X110B

▲2: IICSn 寄存器 = 0110X010B（例如：在中断服务期间读取 ALDn 位时）

通过软件将 IICn.LRELn 设为 1

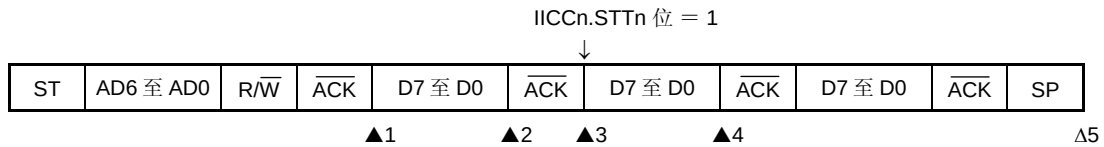
Δ 3: IICSn 寄存器 = 00000001B

- 备注
1. ▲：总是产生
Δ：仅当 SPIEn 位 = 1 时产生
X：忽略
 2. Dn = D6 至 D0
n = 0 至 2



(6) 当试图产生重新开始条件时，由于 SDA0n 引脚为低电平，导致发生仲裁失败

<1> 当 WTIMn 位 = 0 时



▲1: IICSn 寄存器 = 1000X110B

▲2: IICSn 寄存器 = 1000X000B (WTIMn 位 = 1)

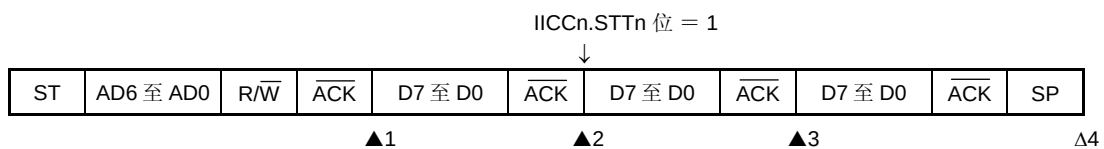
▲3: IICSn 寄存器 = 1000XX00B (WTIMn 位 = 0)

▲4: IICSn 寄存器 = 01000000B (例如：在中断服务期间读取 ALDn 位时)

Δ5: IICSn 寄存器 = 00000001B

- 备注**
1. ▲: 总是产生
 Δ: 仅当 SPIEn 位 = 1 时产生
 X: 忽略
 2. n = 0 至 2

<2> 当 WTIMn 位 = 1 时



▲1: IICSn 寄存器 = 1000X110B

▲2: IICSn 寄存器 = 1000XX00B

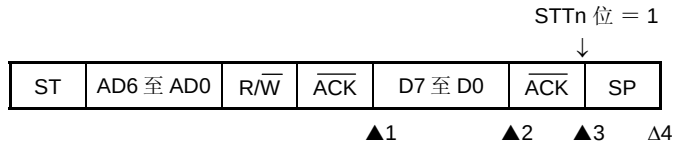
▲3: IICSn 寄存器 = 01000100B (例如：在中断服务期间读取 ALDn 位时)

Δ4: IICSn 寄存器 = 00000001B

- 备注**
1. ▲: 总是产生
 Δ: 仅当 SPIEn 位 = 1 时产生
 X: 忽略
 2. n = 0 至 2

(7) 当试图产生重新开始条件时，由于停止条件导致发生仲裁失败

<1> 当 WTIMn 位 = 0 时



▲1: IICSn 寄存器 = 1000X110B

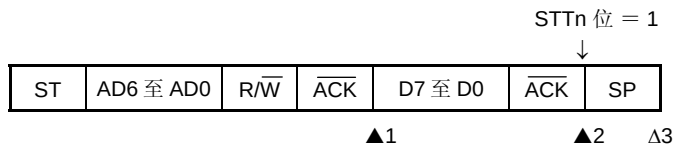
▲2: IICSn 寄存器 = 1000X000B (WTIMn 位 = 1)

▲3: IICSn 寄存器 = 1000XX00B

Δ4: IICSn 寄存器 = 01000001B

- 备注 1. ▲: 总是产生
 Δ: 仅当 SPIEn 位 = 1 时产生
 X: 忽略
 2. n = 0 至 2

<2> 当 WTIMn 位 = 1 时



▲1: IICSn 寄存器 = 1000X110B

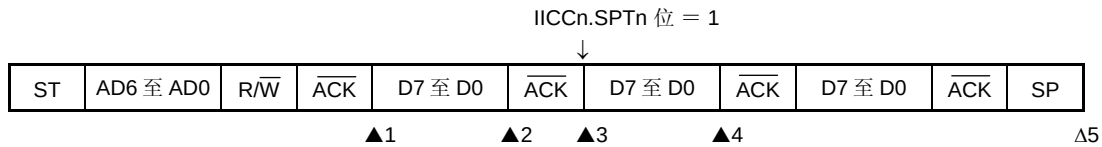
▲2: IICSn 寄存器 = 1000XX00B

Δ3: IICSn 寄存器 = 01000001B

- 备注 1. ▲: 总是产生
 Δ: 仅当 SPIEn 位 = 1 时产生
 X: 忽略
 2. n = 0 至 2

(8) 当试图产生停止条件时，由于 SDA0n 引脚为低电平导致发生仲裁失败

<1> 当 WTIMn 位 = 0 时



▲1: IICSn 寄存器 = 1000X110B

▲2: IICSn 寄存器 = 1000X000B (WTIMn bit = 1)

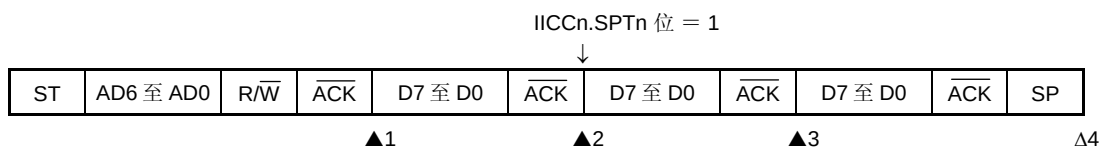
▲3: IICSn 寄存器 = 1000XX00B (WTIMn bit = 0)

▲4: IICSn 寄存器 = 01000000B (例如：在中断服务期间读取 ALDn 位时)

Δ5: IICSn 寄存器 = 00000001B

- 备注**
1. ▲: 总是产生
Δ: 仅当 SPIEn 位 = 1 时产生
X: 忽略
 2. n = 0 至 2

<2> 当 WTIMn 位 = 1 时



▲1: IICSn 寄存器 = 1000X110B

▲2: IICSn 寄存器 = 1000XX00B

▲3: IICSn 寄存器 = 01000000B (例如：在中断服务期间读取 ALDn 位时)

Δ4: IICSn 寄存器 = 00000001B

- 备注**
1. ▲: 总是产生
Δ: 仅当 SPIEn 位 = 1 时产生
X: 忽略
 2. n = 0 至 2

19.8 中断请求信号（INTIICn）的产生时序和等待控制

IICn.WTIMn 位的设定决定了 INTIICn 寄存器的产生时序和相应的等待控制，如下所示。

表 19-3. INTIICn 产生时序和等待控制

WTIMn 位	从设备操作期间			主设备操作期间		
	地址	数据接收	数据发送	地址	数据接收	数据发送
0	9 ^{注 1, 2}	8 ^{注 2}	8 ^{注 2}	9	8	8
1	9 ^{注 1, 2}	9 ^{注 2}	9 ^{注 2}	9	9	9

- 注 1.** 仅当接收的地址和 SVAn 寄存器设定的值相匹配时，从设备的 INTIICn 信号与等待期在第 9 个时钟的下降沿发生。
- 在这一时刻，不论 IICn.ACKEn 位如何设置，都产生 $\overline{\text{ACK}}$ 应答信号，对于已经接收到扩展码的从设备，INTIICn 信号在第 8 个时钟的下降沿发生。
- 当重新开始以后地址不匹配时，INTIICn 信号在第 9 个时钟的下降沿发生，但是，不产生等待期。
- 2.** 如果接收地址和 SVAn 寄存器的内容不匹配，而且不能收到扩展码，则既不产生 INTIICn 信号，也不产生等待期。

- 备注 1.** 表中的数目表示串行时钟信号的时钟数目。中断请求信号和等待控制都和这些时钟信号的下降沿同步。
- 2.** n = 0 至 2

(1) 在地址发送/接收期间

- 从机设备操作：不论 WTIMn 位如何设置，中断请求和等待时序都根据以上的**注 1**和**注 2**说明的条件决定。
- 主机设备操作：不论 WTIMn 位如何设置，中断请求和等待时序在第 9 个时钟的下降沿发生。

(2) 在数据接收期间

- 主机/从机设备操作：中断请求和等待时序由 WTIMn 位的设置决定。

(3) 数据发送期间

- 主机/从机设备操作：中断请求和等待时序由 WTIMn 位的设置决定。

(4) 等待状态取消方法

四种等待取消方法如下所示：

- 通过将 IICn.WRELn 位置为 1。
- 通过向 IICn 寄存器写入数据。
- 通过开始条件的设置 (IICn.STTn 位= 1) ^註。
- 通过停止条件的设置 (IICn.SPTn 位= 1) ^註。

注 只限于主机设备。

当已经选择 8 时钟等待时 (WTIMn 位 = 0)，取消等待状态之前必须确定是否已经产生了 $\overline{\text{ACK}}$ 信号。

备注 n = 0 至 2

(5) 停止条件检测

当检测到停止条件时，就会产生 INTIICn 信号。

备注 n = 0 至 2

19.9 地址匹配检测方法

处于 I²C 总线模式时，主机设备可以通过发送从机地址来选择特定的从机设备。

地址匹配检测由设备硬件自动执行。当本机地址被置入 SVAn 寄存器以后，而且置入 SVAn 寄存器的本机地址和主机发送的从机地址匹配，或当接收到扩展码时，产生一个 INTIICn 信号（n = 0 至 2）。

19.10 错误检测

处于 I²C 总线模式时，串行数据总线（SDA0n）的状态在数据传送期间被发送设备的 IICn 寄存器所捕获，这样，发送以前 IICn 寄存器内的数据就可以和所发送的 IICn 寄存器数据进行比较，从而能够检测传送错误。当比较值不匹配时，判断为错误发生（n = 0 至 2）。

19.11 扩展代码

- (1) 当接收到的地址数据的高 4 位为 0000 或 1111 时，扩展码标志（IICSn.EXCn 位）被置位指示扩展码接收，并且，在第 8 个时钟的下降沿产生一个中断请求信号（INTIICn）（n = 0 至 2）。

存储在 SVAn 寄存器中的本机地址不受影响。

- (2) 如果 SVAn 寄存器通过 10 位地址传输被设置为 11110xx0，11110xx0 数据由主机设备发出，其结果如下所示。注意，INTIICn 信号在第 8 个时钟的下降沿产生（n = 0 至 2）。

- 数据高 4 位匹配：EXCn 位 = 1
- 7 位数据匹配：IICSn.COIn 位 = 1

- (3) 由于根据紧随扩展码之后的数据不同，而中断请求信号发生后的处理也各不相同，因此，这些处理由软件完成。例如，接收到扩展码以后，当不希望从机设备工作时，将 IICn.LRELn 位设置为 1，CPU 将进入下一次通信等待状态。

<R>

表 19-4. 扩展码位定义

从机地址	R/ $\overline{W}$ 位	说明
0000 000	0	通用呼叫地址
1111 0xx	0	10-位从机地址规范（当指定地址时）
1111 0xx	1	10-位从机地址规范（地址匹配之后，产生读取命令）

备注 关于除了上述扩展码之外的扩展码，参见由 NXP 产生的 I²C 总线规范。

19.12 仲裁

当数个主机设备同时产生开始条件时（当 IICCn.STTn 位在 IICSn.STDn 位被置 1 之前设置为 1），在多主机间通信同时判断时钟数目，直到数据不同为止。这种操作称之为仲裁（n = 0 至 2）。

当一个主机设备在仲裁中失败时，仲裁失败发生时序将仲裁失败标志（IICSn.ALDn 位）设置为 1，并且，SCL0n 信号线和 SDA0n 信号线都被设置为高阻状态，从而释放总线。

仲裁失败检测基于下次中断请求信号（INTIICn）的时序（当检测到停止条件时，第 8 或第 9 个时钟）和由软件完成的 ALDn 位 = 1 的设定（n = 0 至 2）。

关于中断请求信号时序的细节，参见 19.7 I²C 中断请求信号（INTIICn）。

图 19-14. 仲裁时序举例

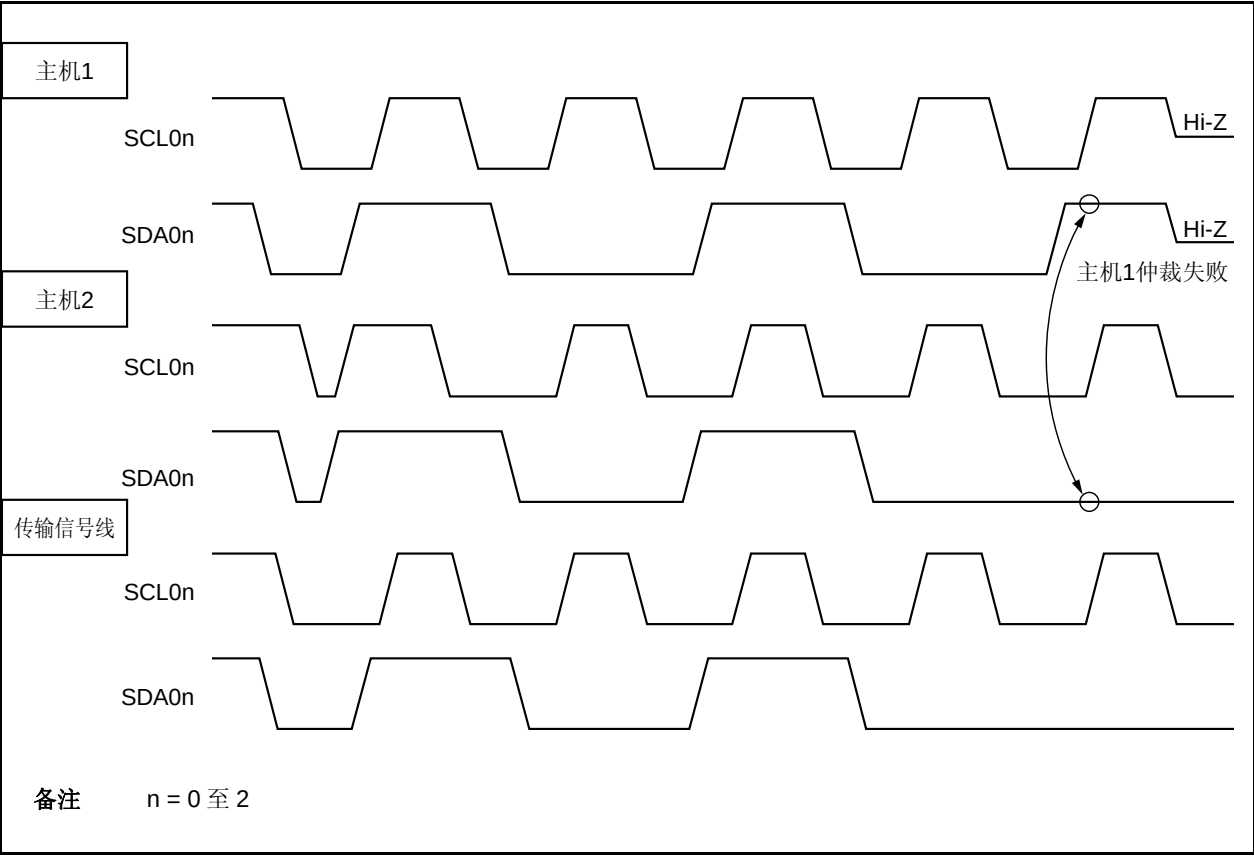


表 19-5. 在仲裁和中断请求信号发生时序期间的状态

仲裁期间的状态	中断请求发生时序
发送地址传输	字节传输之后的第 8 个或第 9 个时钟的下降沿 ^{注 1}
地址发送后读出/写入数据	
发送扩展代码	
扩展码发送后读出/写入数据	
发送数据	
数据接收后 $\overline{\text{ACK}}$ 传输期间	
当数据传输期间检测到重新开始条件	
当数据传输期间检测到停止条件	当停止条件产生时（当 IICn.SPIEn 位 = 1 时） ^{注 2}
SDA0n 引脚为低电平而试图产生重新开始条件时	字节传输之后的第 8 个或第 9 个时钟的下降沿 ^{注 1}
当检测到停止条件时试图产生重新开始条件	当停止条件产生时（当 IICn.SPIEn 位 = 1 时） ^{注 2}
DSA0n 引脚为低电平而试图产生停止条件时	字节传输之后的第 8 个或第 9 个时钟的下降沿 ^{注 1}
SCL0n 引脚为低电平而试图产生重新开始条件时	

- 注
1. 当 IICn.WTIMn 位 = 1 时，INTIICn 信号在第 9 个时钟的下降沿发生。当 WTIMn 位 = 0 并且接收到扩展码的从地址时，INTIICn 信号在第 8 个时钟的下降沿发生（n = 0 至 2）。
 2. 当有可能发生仲裁时，在主机设备操作下，设置 SPIEn 位为 1（n = 0 至 2）。

19.13 唤醒功能

唤醒功能是在通过 I²C 总线的从机功能接收到本机地址或扩展码时产生中断请求信号（INTIICn）的功能。在地址不匹配时，该功能可以通过阻止产生不必要的 INTIICn 信号使处理更为高效。

当检测到开始条件后，待机状态被唤醒。仲裁失败可能将主机设备（已经产生了开始条件）变为从机设备，当这种可能情况下的地址传输时，待机模式唤醒有效。

然而，当检测到停止条件后，IICn.SPIEn 位总是置位，而不会唤醒功能，而这决定了 INTIICn 信号的使能或禁止（n = 0 至 2）。

19.14 通信保留

19.14.1 当通信保留功能使能时 (IICFn.IICRSVn 位 = 0)

在目前不能使用总线的情况下开始主机设备通信时，通信保留功能可以等到总线释放后再产生开始条件。有两种模式下不能使用总线。

- 当仲裁结果为既不是主机操作，又不是从机操作。
- 当收到扩展码后而且从机操作被禁止时（当 IICFn.LRELn 位被置为 1 时， $\overline{ACK}$ 信号未返回且总线释放）（n = 0 至 2）。

如果在不能使用总线时将 IICFn.STTn 位置 1，将自动产生一个开始条件，并且总线释放后（检测到停止条件后）设置为等待状态。

检测到总线释放（检测到停止条件时）时，写入 IICFn 寄存器导致主机地址传输开始。由于这点，IICFn.SPIEn 位应被设置为 1（n = 0 至 2）。

当 STTn 位被设置为 1 后，根据总线状态确定工作模式（开始条件或通信保留状态）（n = 0 至 2）。

如果总线已被释放.....产生开始条件

如果总线未被释放（待机模式）.....通信保留

要检测 STTn 位决定了何种工作模式，将 STTn 位设置为 1，待等待期完毕后，然后检查 IICSn.MSTS_n 位（n = 0 至 2）。

应由软件设定的等待期如表 19-6 所列，这些等待期可以通过 SMCn，CLn1 以及 IICCLn 寄存器的 CLn0 和 IICXn.CLXn 位进行设定（n = 0 至 2）。

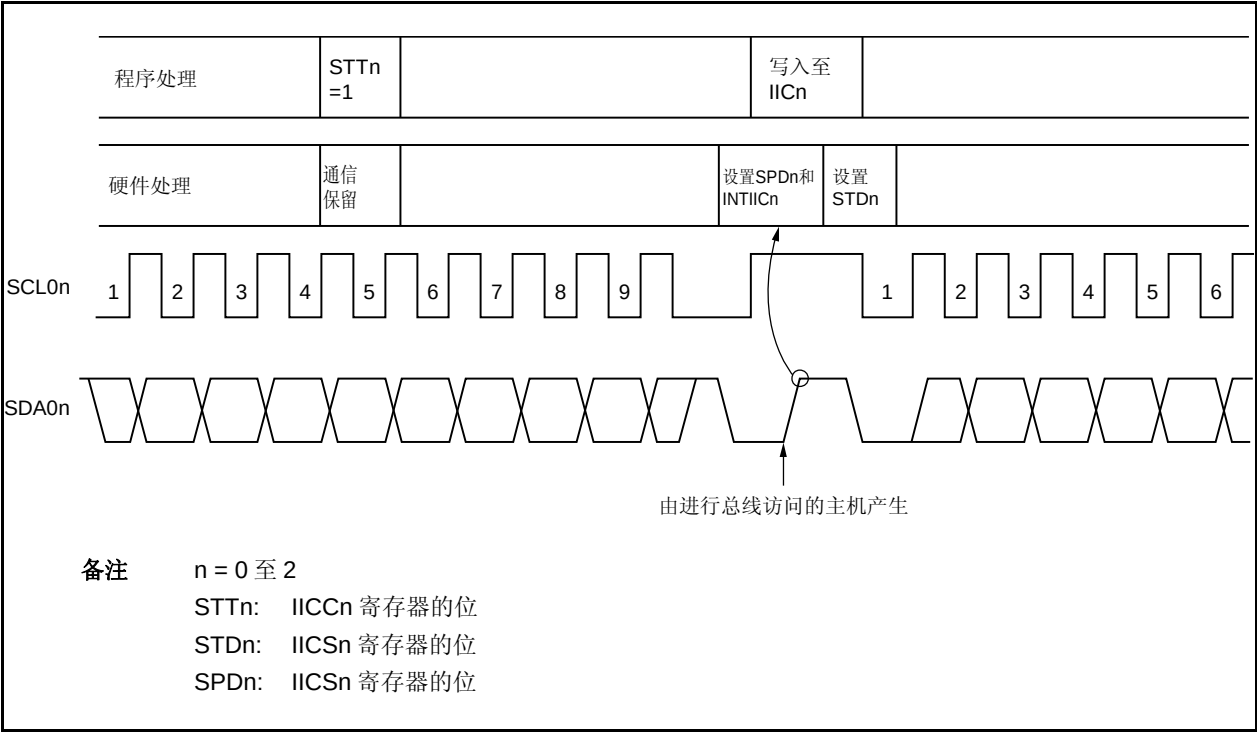
表 19-6. 等待期

时钟选择	CLXn	SMCn	CLn1	CLn0	等待期
f _{xx} /6 (OCKSm = 11H)	0	0	0	0	156 个时钟
f _{xx} /8 (OCKSm = 12H)	0	0	0	0	208 个时钟
f _{xx} /10 (OCKSm = 13H)	0	0	0	0	260 个时钟
f _{xx} /4 (OCKSm = 10H)	0	0	0	1	188 个时钟
f _{xx} /6 (OCKSm = 11H)	0	0	0	1	282 个时钟
f _{xx} /8 (OCKSm = 12H)	0	0	0	1	376 个时钟
f _{xx} /10 (OCKSm = 13H)	0	0	0	1	470 个时钟
f _{xx} /4 (OCKSm = 10H)	0	0	1	1	148 个时钟
f _{xx} /6 (OCKSm = 11H)	0	0	1	1	222 个时钟
f _{xx} /4 (OCKSm = 10H)	0	1	0	×	64 个时钟
f _{xx} /6 (OCKSm = 11H)	0	1	0	×	96 个时钟
f _{xx} /8 (OCKSm = 12H)	0	1	0	×	128 个时钟
f _{xx} /10 (OCKSm = 13H)	0	1	0	×	160 个时钟
f _{xx} /4 (OCKSm = 10H)	0	1	1	1	52 个时钟
f _{xx} /6 (OCKSm = 11H)	0	1	1	1	78 个时钟
f _{xx} /6 (OCKSm = 11H)	1	1	0	×	60 个时钟
f _{xx} /8 (OCKSm = 12H)	1	1	0	×	80 个时钟
f _{xx} /10 (OCKSm = 13H)	1	1	0	×	100 个时钟

备注 1. m = 0 和 1
 n = 0 至 2
 2. × 忽略

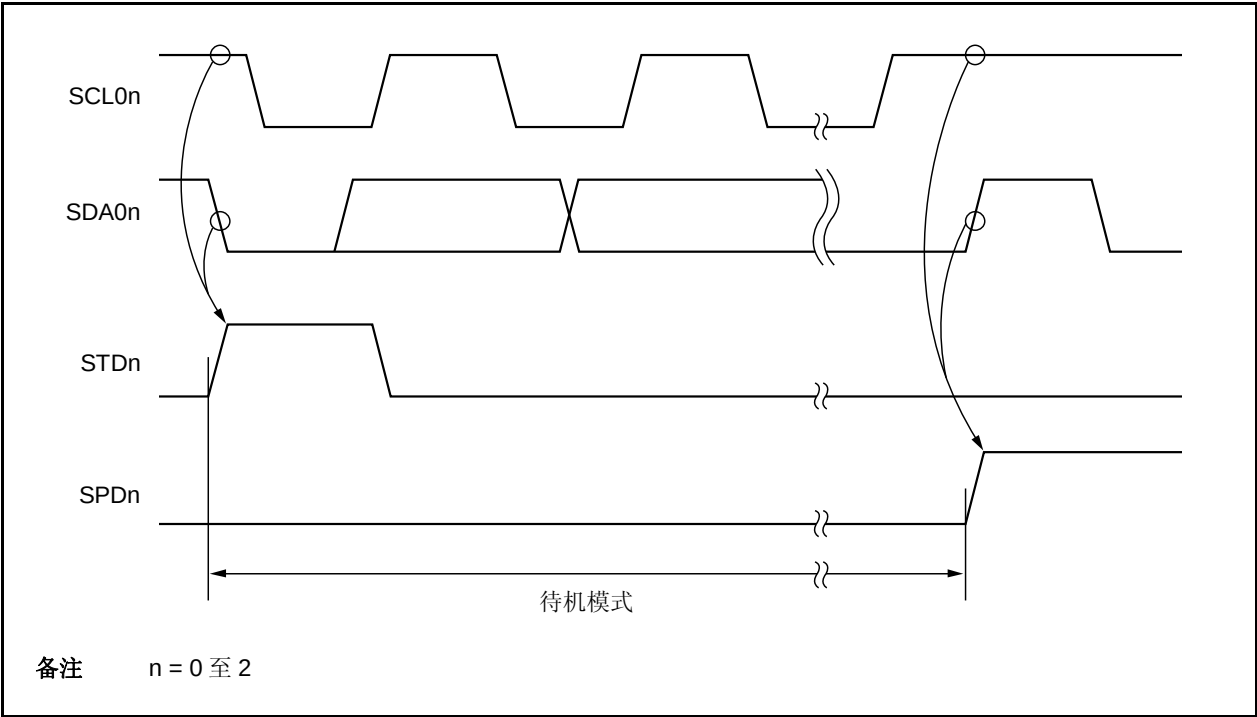
通信保留时序如下所示。

图 19-15. 通信保留时序



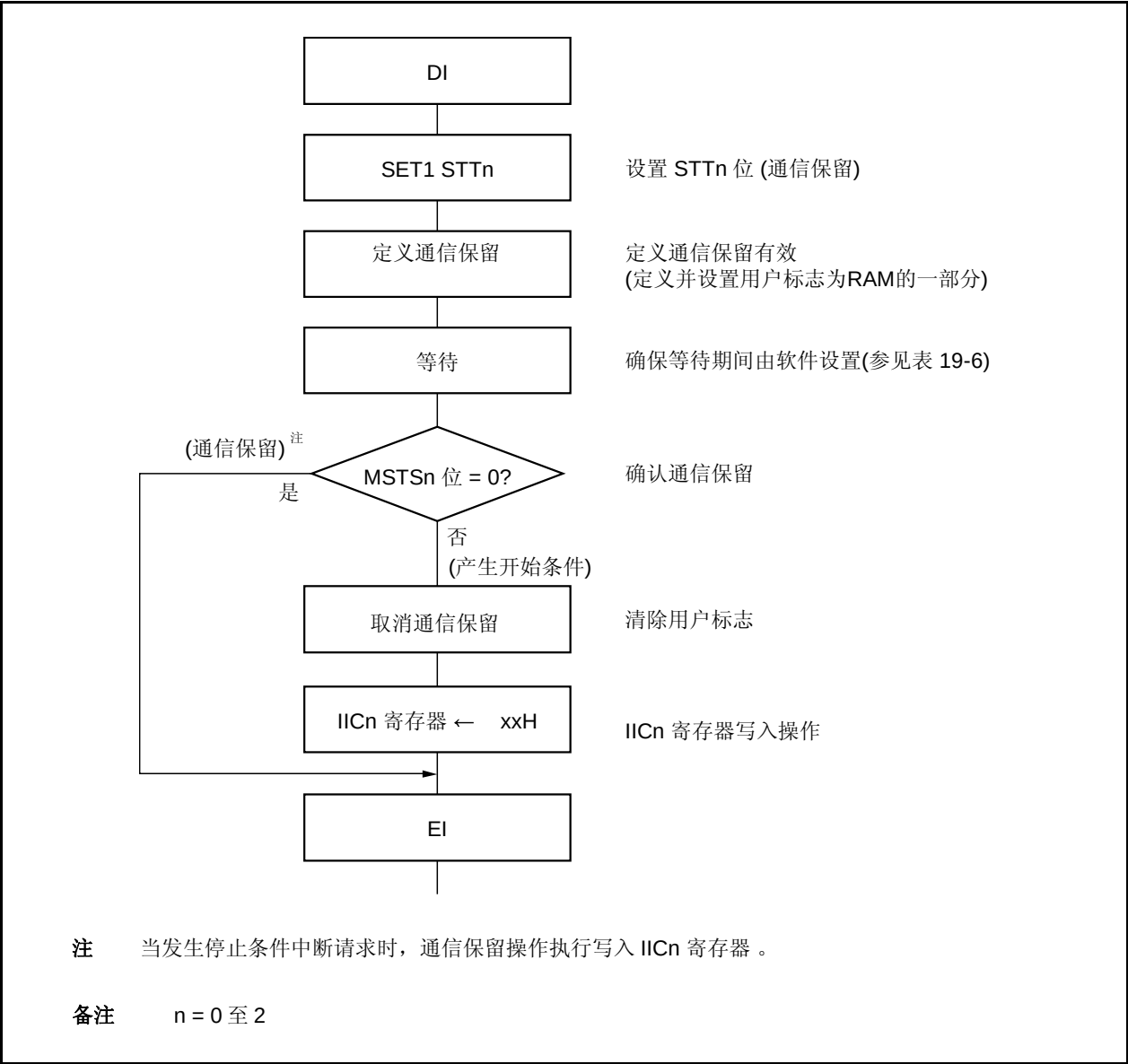
通过以下时序通信保留被受理。IICSn.STDn 位设置为 1 后，在检测到停止条件前，通过设置 IICn.STTn 位为 1 使得一次通信保留（n = 0 至 2）。

图 19-16. 受理通信保留的时序



通信保留流程图说明如下。

图 19-17. 通信保留流程图



19.14.2 当通信保留功能被禁止时（IICFn.IICRSVn 位 = 1）时

在总线通信中当不能使用总线时，在将 IICFn.STTn 位置位的情况下，通信保留申请被拒绝，且不会产生开始条件。有两种模式下不能使用总线。

- 当仲裁结果为既不是主机操作，又不是从机操作。
- 当收到扩展码后而且从机操作被禁止时（当 IICFn.LRELn 位被置为 1 时， $\overline{\text{ACK}}$ 信号未返回且总线释放）（n = 0 至 2）。

为了确认是产生了开始条件还是申请遭到拒绝，检查 IICFn.STCFn 标志。 设置 STTn 位为 1 后直到 STCFn 标志被置位，需要的时间如表 19-7 所显示。

表 19-7. 等待期

OCKSEn _m	OCKSm ₁	OCKSm ₀	CLn ₁	CLn ₀	等待期
1	0	0	0	×	20 个时钟
1	0	1	0	×	30 个时钟
1	1	0	0	×	40 个时钟
1	1	1	0	×	50 个时钟
0	0	0	1	0	10 个时钟

备注 1. ×: 忽略
 2. n = 0 至 2
 m = 0, 1

19.15 注意事项

(1) 当 IICFn.STCENn 位 = 0 时

紧随 I²C0n 操作使能之后，不管总线的实际状态如何，总线通信状态（IICFn.IICBSYn 位 = 1）总是得到认可。若要在尚未检测到停止条件的情况下执行主机通信，需要先产生一个停止条件，使总线在开始主机通信前得到释放。
使用以下次序产生停止条件。

- <1> 设置 IICCLn 寄存器。
- <2> 设置 IICCN.IICEn 位。
- <3> 设置 IICCN.SPTn 位。

(2) 当 IICFn.STCENn 位 = 1 时

紧随 I²C0n 操作使能之后，不管总线的实际状态如何，总线释放状态（IICFn.IICBSYn 位 = 0）总是得到认可。若要产生首个开始条件（IICCN.STTn 位 = 1），有必要确认总线是否已经释放，以便不干扰其它通信。

(3) 在和其它设备进行通信时，当 V850ES/JG3-H 和 V850ES/JH3-H 的 IICCN.IICEn 位被设置为 1 时，开始条件的检测依赖于通信序列的状态，SCL0n 和 SDA0n 信号线是高电平时确保将 IICCN.IICEn 位设置为 1。

(4) 使能操作（IICCN.IICEn 位 = 1）之前，通过 IICCLn，IICXn 和 OCKSm 寄存器确定工作时钟频率。要改变操作时钟频率，需要将 IICCN.IICEn 位清 0 一次。

(5) 当 IICCN.STTn 和 IICCN.SPTn 位已经被设置为 1，若不首先将其清 0，则一定不要对其再次置位。

(6) 如果传送已经被保留，设置 IICCN.SPIEn 位为 1，这样，检测到停止条件后就会产生一个中断请求信号。当一个中断请求信号已经产生后，向 I²Cn 写入通信数据将释放等待状态，之后，传输开始。然而，如果检测到停止条件后不产生中断请求信号，由于请求未产生，传输将暂停在停止条件上。但是，没有必要为了软件检测 IICSn.MSTSn 位而将 SPIEn 位置为 1。

备注 n = 0 至 2
 m = 0, 1

19.16 通信操作

以下显示了 3 种带有流程图操作流程。

(1) 主机工作于单主机系统

当使用 V850ES/JG3-H and V850ES/JH3-H 作为单主机系统中的主机设备时，操作流程如下所述。

操作流程可以宽泛的分为初始设置和通信操作两部分。开始时执行初始设置。如果需要和从机进行通信，进行通信准备后执行通信操作。

(2) 主机工作于多主机系统

在 I²C_{0n} 总线多主机系统中，当总线参与到通信之中后，不能由 I²C 总线规范判定总线是释放还是被使用。这里，当数据线和时钟线都在一定的时间（1 帧）里保持高电平，V850ES/JG3-H 或 V850ES/JH3-H 就认为总线释放并可参与通信。

操作流程可以宽泛的分为初始设置、通信等待和通信处理过程。在 V850ES/JG3-H 或 V850ES/JH3-H 仲裁中失败并被认定为从设备的情况这里予以省略，而只是展示了作为主机的处理操作。通信开始时执行初始设置以便参与通信。之后，作为主机等待通信请求或作为从机设备等待指定。实际通信在通信处理时进行，且实际通信除了包括从机的发送/接收数据之外还包括其他主机数据的只能仲裁。

(3) 从机操作

当 V850ES/JG3-H 或 V850ES/JH3-H 用作 I²C_{0n} 总线的从机设备时例子如下所示。

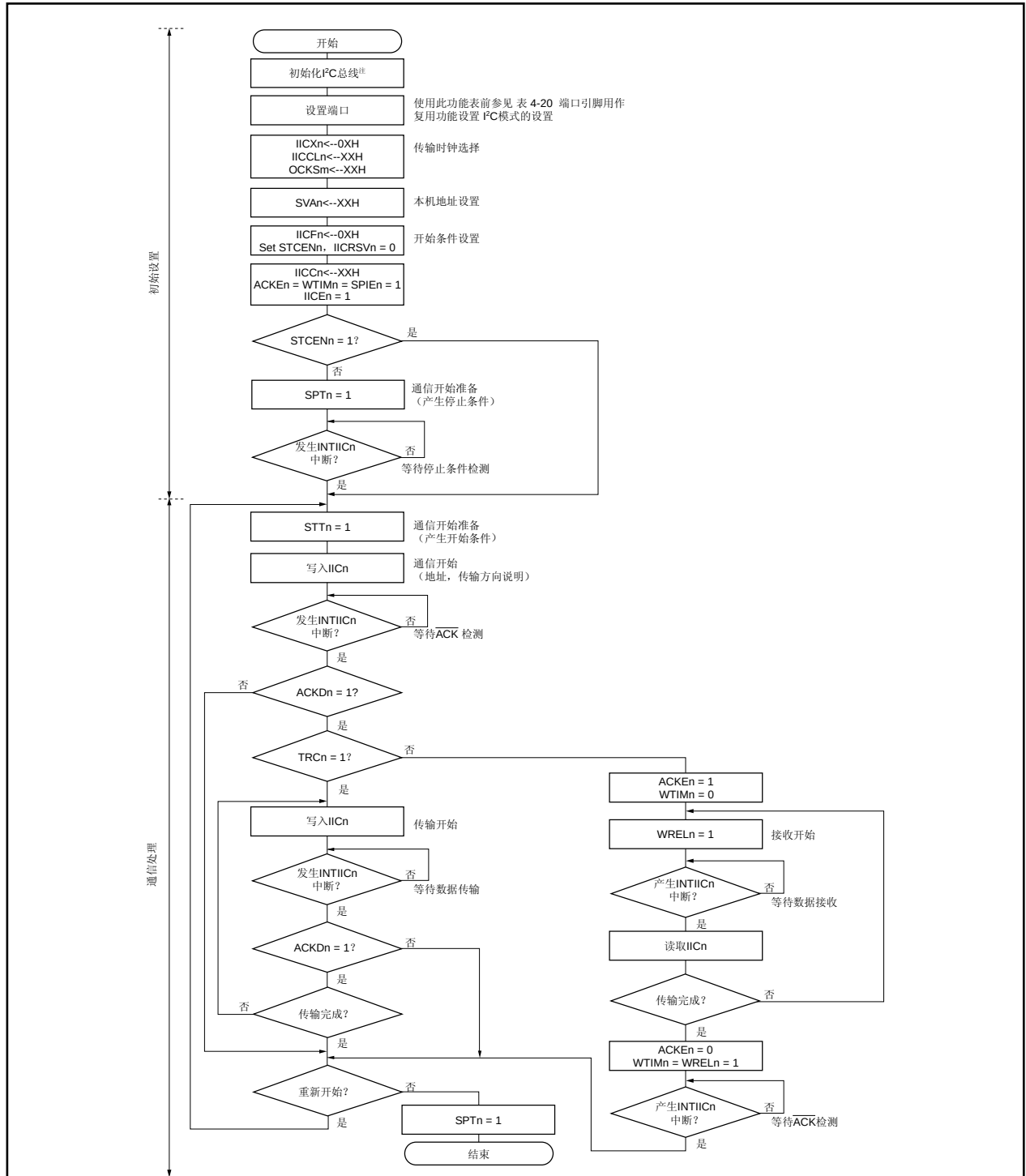
当用作从机时，由一个中断启动操作。在启动时执行初始设置，之后，等待 INTIIC_n 中断信号发生（通信等待）。当 INTIIC_n 中断发生时，通信状态被判定，并且其判决结果作为一个标志传送给主处理程序。

通过检查该标志，必要的通信处理得以执行。

备注 n = 0 至 2

19.16.1 主机工作于单主机系统

图 19-18. 单主机系统的主机操作



注 释放 I²C_{0n} 总线（SCL_{0n}, SDA_{0n} 引脚 = 高电平）与进行通信产品的规格一致。例如，当 EEPROM[™] 输出低电平到 SDA_{0n} 引脚时，设置 SCL_{0n} 引脚为输出端口并从中输出时钟脉冲，直到 SDA_{0n} 引脚为恒定的高电平为止。

备注 1. 至于发送格式和传输格式，与进行通信产品的规格一致。
2. n=0 至 2, m=0, 1

19.16.2 主机工作于多主机系统

图 19-19. 多主机系统中的主机操作 (1/3)

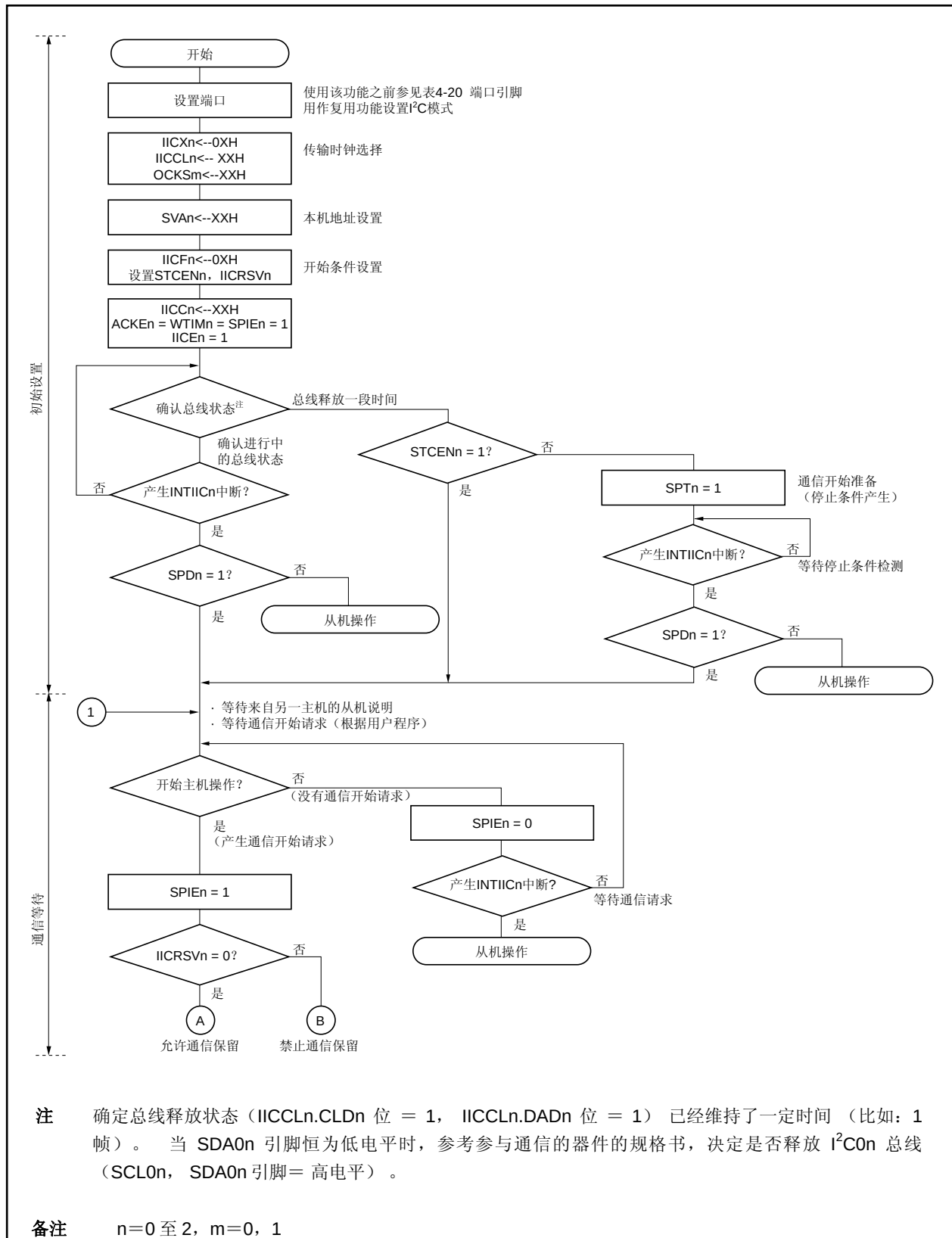


图 19-19. 多主机系统中的主机操作 (2/3)

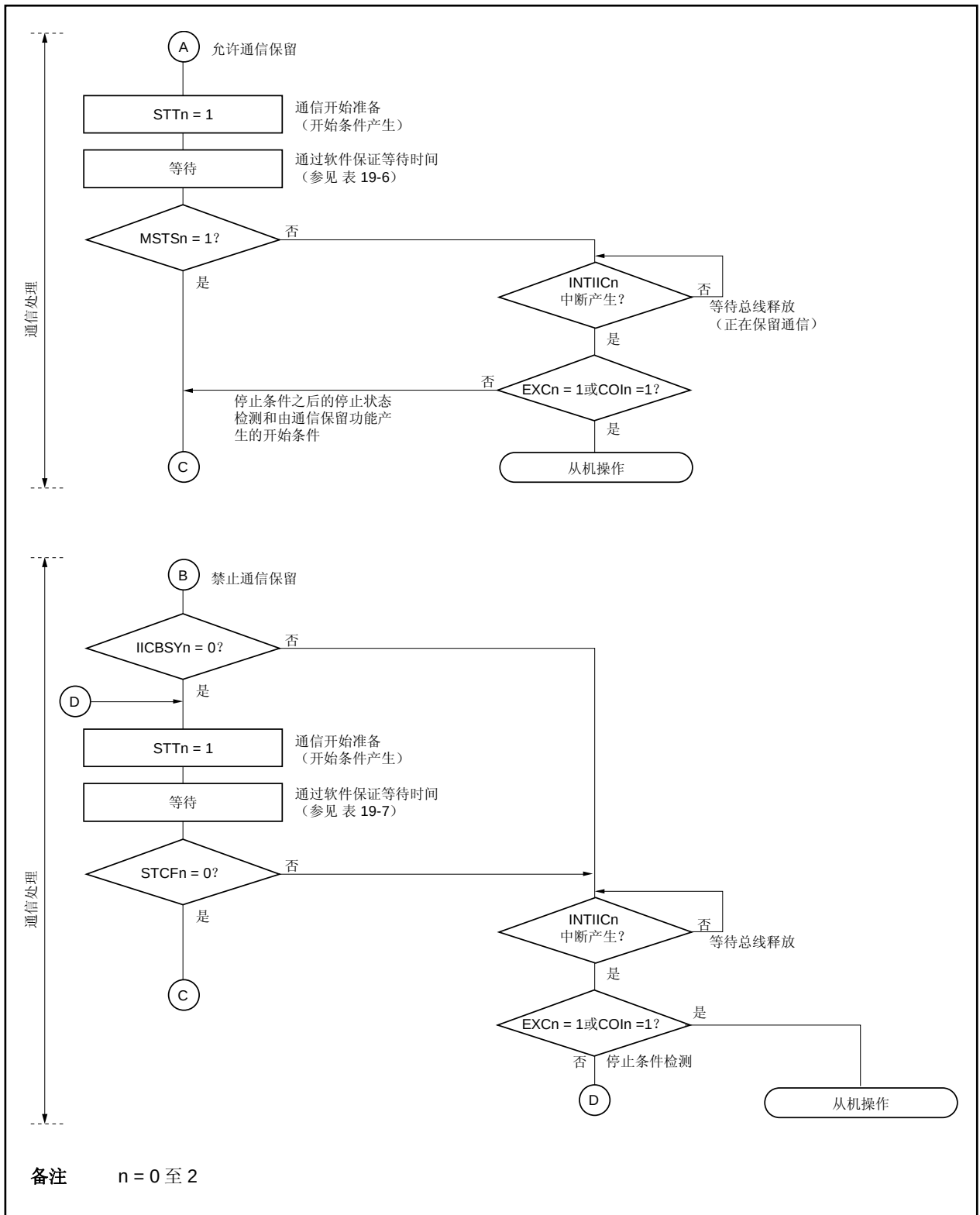
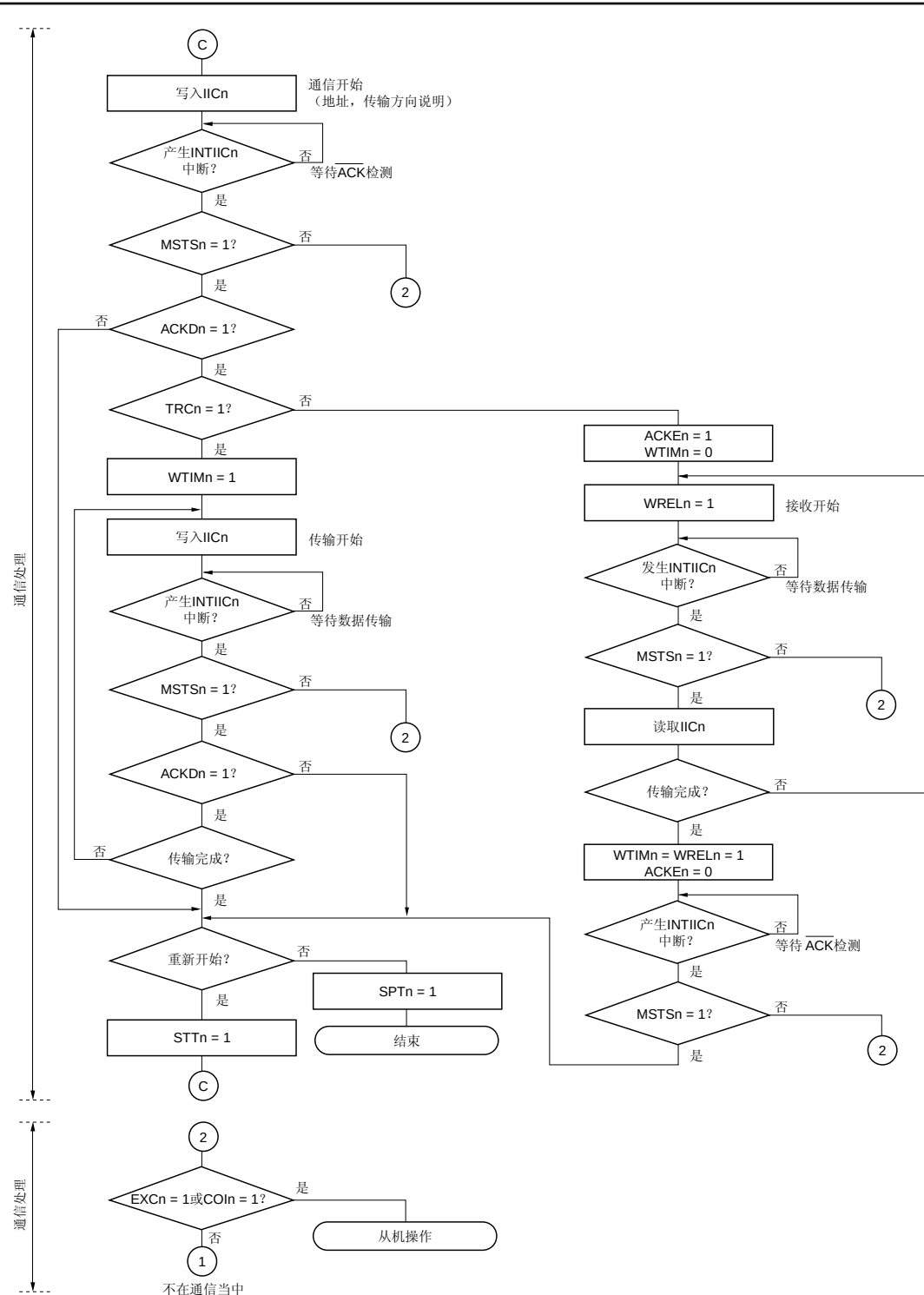


图 19-19. 多主机系统中的主机操作 (3/3)



备注

1. 至于发送格式和传输格式，与进行通信产品的规格一致。
2. 当 V850ES/JG3-H 或 V850ES/JH3-H 在多主机系统中用作主机时，每次 INTIICn 中断发生时都要读取 IICSn.MSTSn 位以确定仲裁结果。
3. 当 V850ES/JG3-H 或 V850ES/JH3-H 在多主机系统中用作从机时，每次 INTIICn 中断发生时都要使用 IICSn 和 IICFn 寄存器以确定下次处理。
4. n = 0 至 2

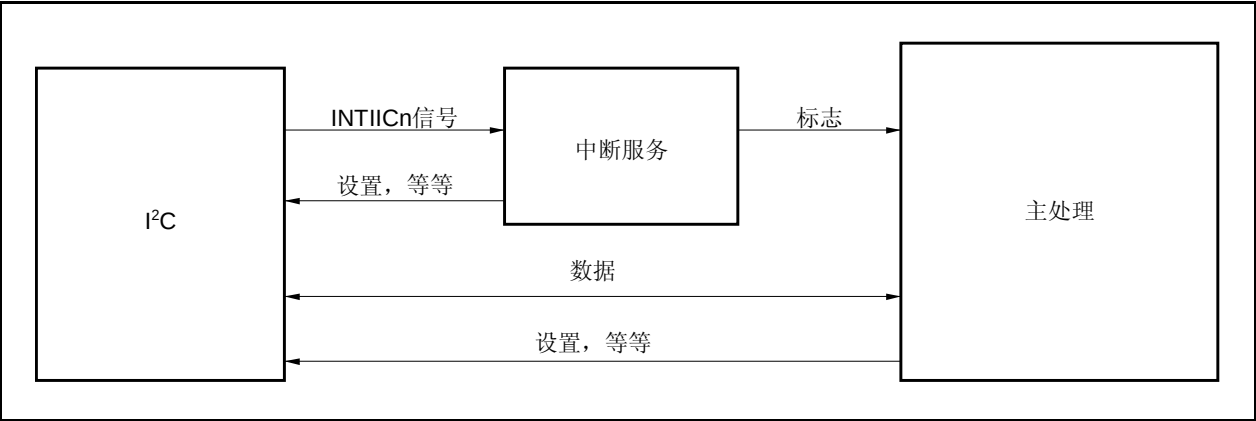
19.16.3 从机操作

以下展示从机操作的处理步骤。

基本而言，从机设备的操作都是由事件驱动执行的。所以，需要由 INTIICn 中断进行处理（处理需要操作状态的有效改变，比如通信过程中检测到停止条件）。

以下描述假定数据通信不支持扩展码，还假设 INTIICn 中断仅服务于通信过程中的状态改变，并且，实际数据通信由主程序执行。

图 19-20. 从机操作期间的软件概况



所以，系统准备了以下 3 个标志，以便将其传送给主程序从而执行数据传输处理，而不使用 INTIICn 信号。

(1) 通信模式标志

本标志指示以下通信状态：

清除模式：	未进行数据通信
通信模式：	正在进行数据通信（检测到有效地址，检测到停止条件，未检测到来自主机的 ACK 以及地址不匹配）

(2) 就绪标志

本标志指示数据通信允许。这和正常数据传输期间发生 INTIICn 中断后的状态相同。本标志在中断处理模块内被置位，而在主程序模块中被清除。第一个数据发送的就绪标志并不在中断处理模块内设置，所以，首个数据传输后不需要清除处理（地址匹配状态被视为申请下一个数据）。

(3) 通信方向标志

本标志指示数据通信的方向，和 IICSn.TRCn 位的值相同。

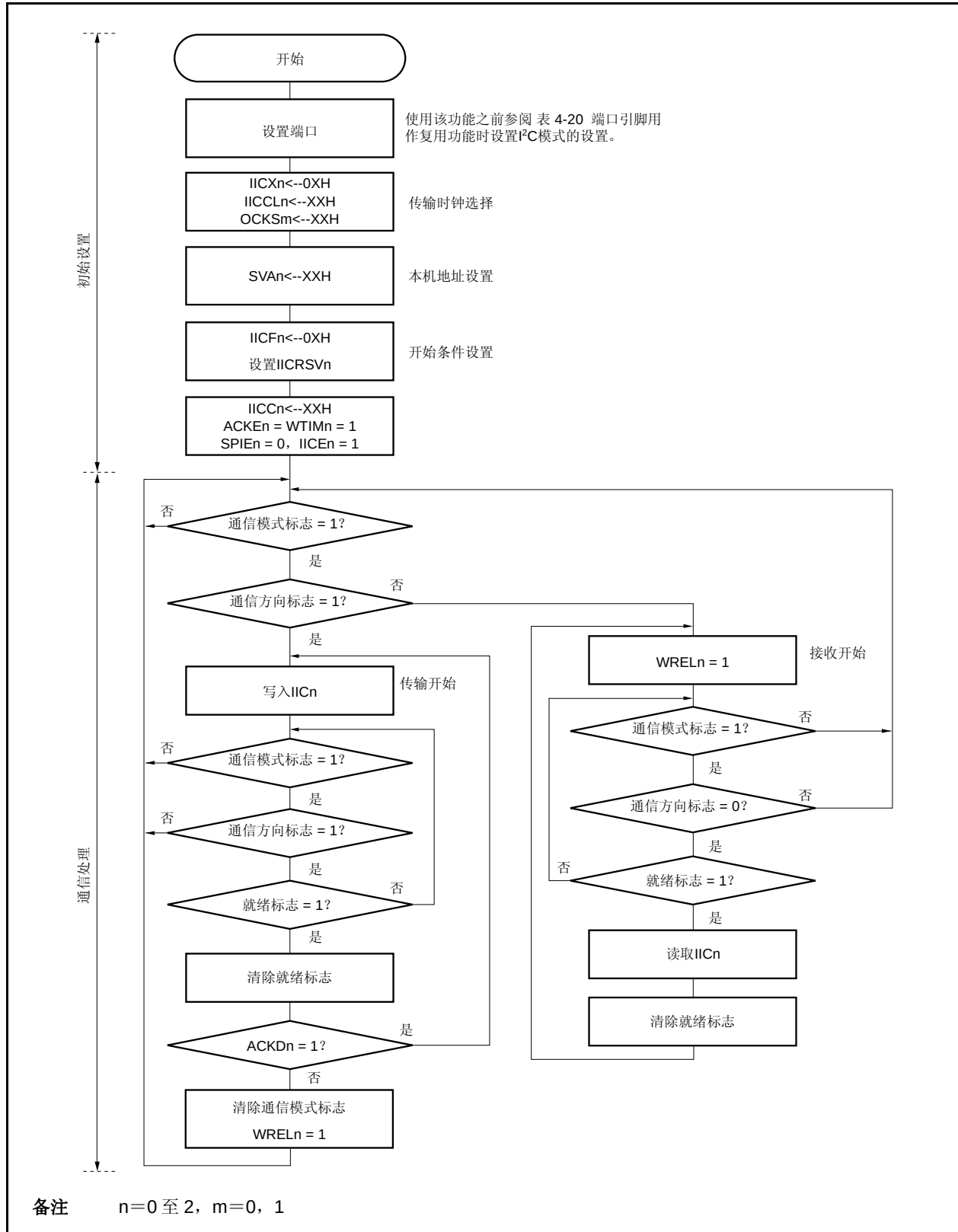
以下显示了从机操作期间主程序模块的操作。

启动 I²C0n 并且等待通信允许状态。当通信被使能后，I²C0n 使用通信模式标志和就绪标志进行通信（停止条件和开始条件的处理由中断处理执行，通信中的状态由标志位确认）。

对于发送操作，重复发送直到主机设备停止返回 ACK 信号，当主机设备停止返回 ACK 信号后，传输完毕。

对于接收操作，接收到所要求数目的数据后不返回 $\overline{\text{ACK}}$ 信号请求下一个数据，紧接着通信完毕。此后，主机设备产生停止条件或重新开始条件。这导致退出通信。

图 19-21. 从机操作流程 (1)

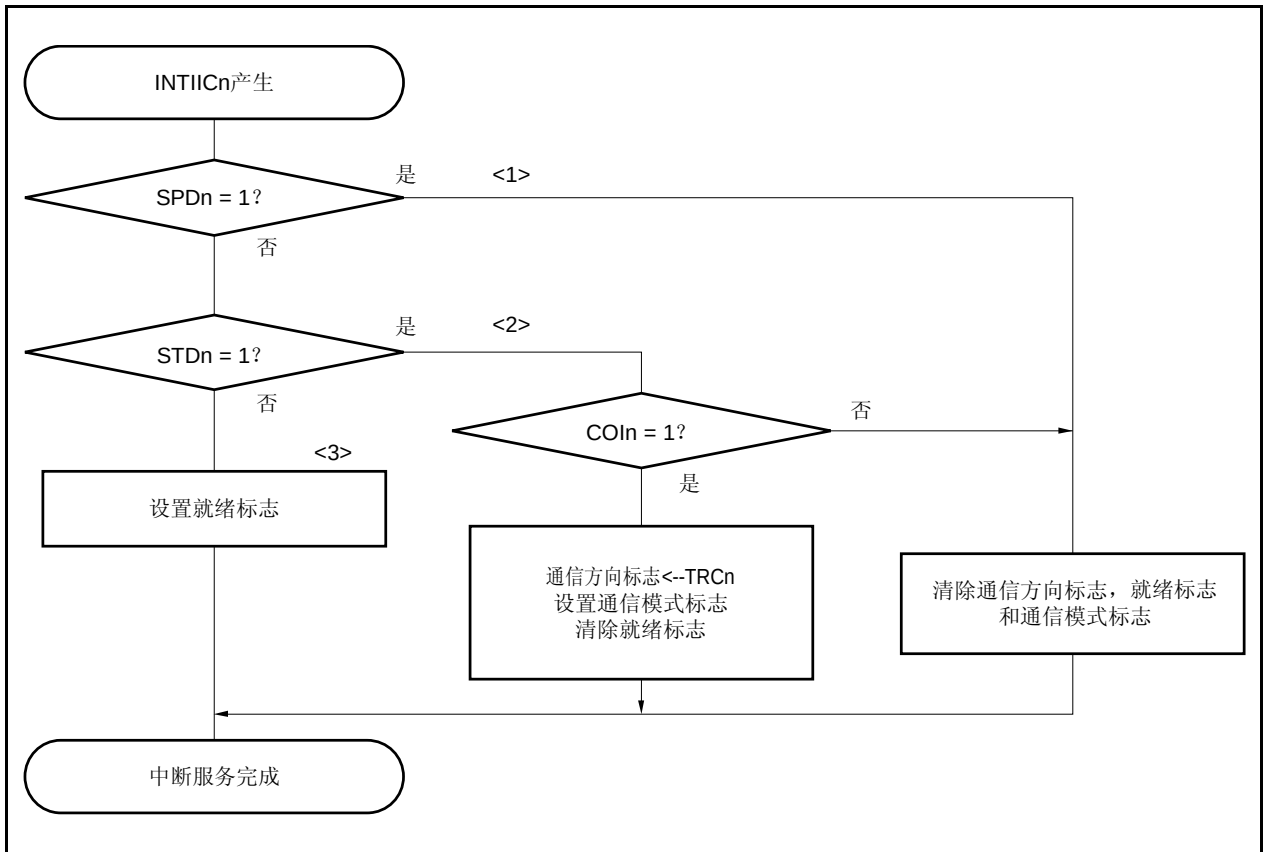


以下显示了从机设备使用 INTIICn 中断进行处理的一个例子（这里，假定不使用扩展码）。在 INTIICn 中断服务期间，通信状态得以确认并且执行以下 3 个步骤。

- <1> 当检测到停止条件时，通信终止。
- <2> 当检测到开始条件后，进行地址确认。如果地址不匹配，则通信终止。如果地址匹配，通信模式被设定并且释放等待状态，处理程序从中断中返回（就绪标志被清除）。
- <3> 对于数据发送/接收过程，当就绪标志置位时，处理程序从中断中返回，此时，I²C0n 总线保持等待状态。

备注 以上<1>至<3>步骤对应着图 19-22 从机操作流程（2）的<1>至<3>步骤。

图 19-22. 从机操作流程（2）



19.17 数据通信时序

当使用 I²C 总线模式时，主机设备通过串行总线输出一个地址，选择多个从机设备之一作为自己的通信伙伴。

输出从机设备地址以后，主机设备发送 IICSn.TRCn 位指明数据传输方向，之后，开始与从机设备的串行通信。

IICn 寄存器的移位操作和串行时钟引脚（SCL0n）信号的下降沿同步。发送数据被传送到 SO 锁存器并经由 SDA0n 引脚输出（MSB 在先）。

由 SDA0n 引脚输入的数据在 SCL0n 引脚信号的上升沿被 IICn 寄存器所捕获。

数据通信时序如下所示。

备注 n = 0 至 2

<R>

图 19-23. 主机向从机传输举例
(当主机设备和从机设备都选择 9 时钟等待时) (1/3)

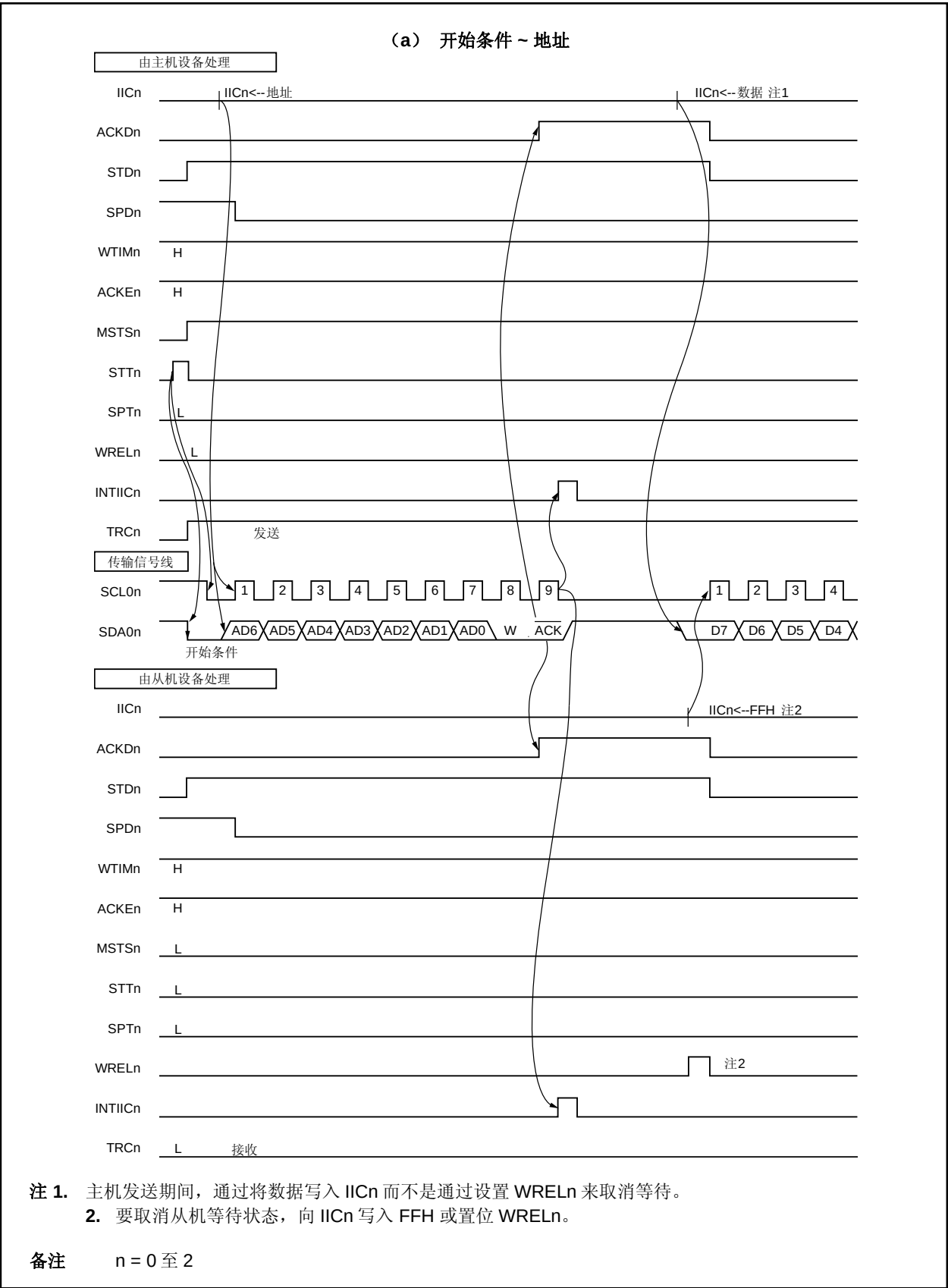


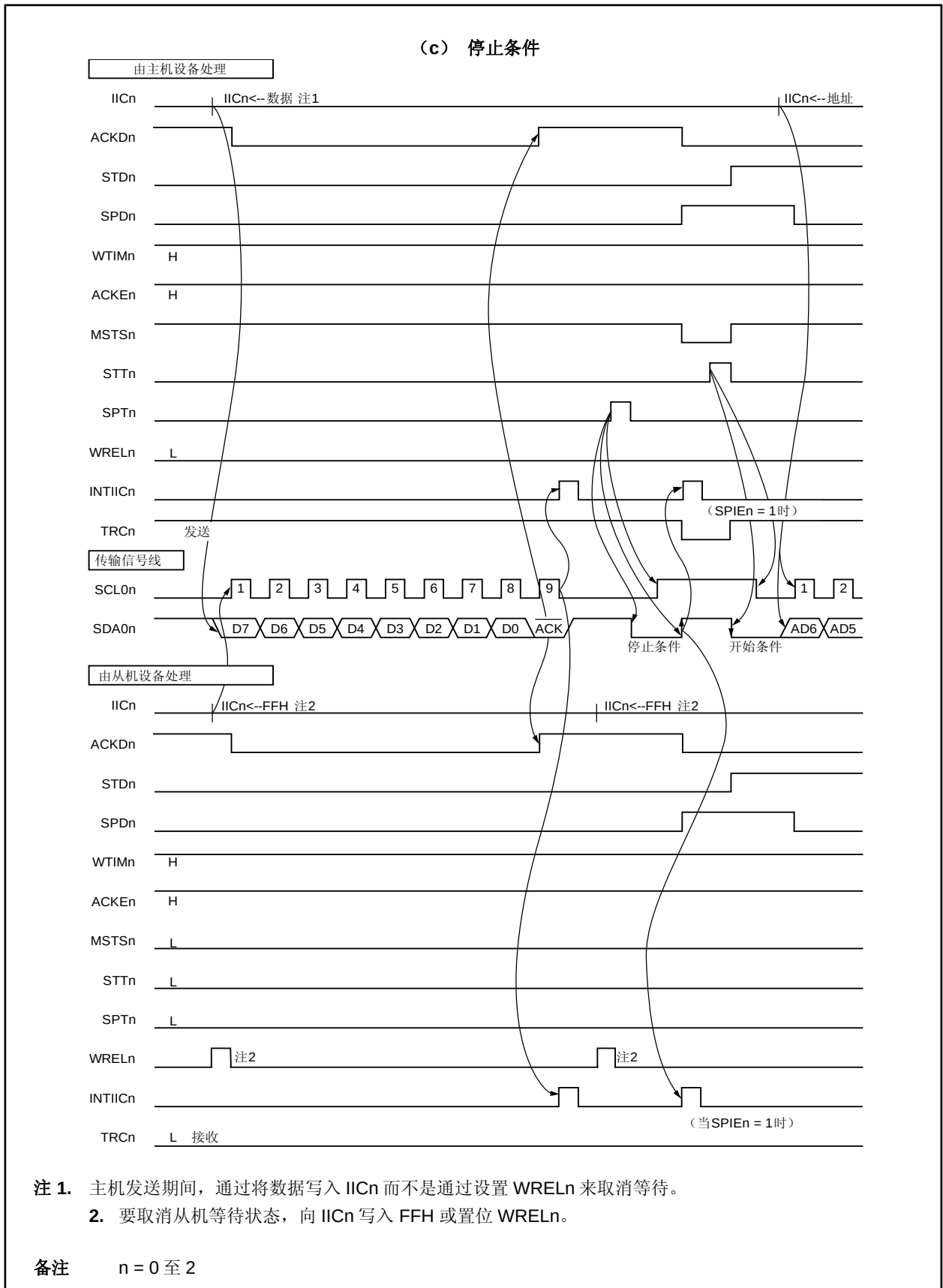
图 19-23. 主机向从机传输举例
(当主机设备和从机设备都选择 9 时钟等待时) (2/3)



用户手册初稿 U19181CA2V0UD

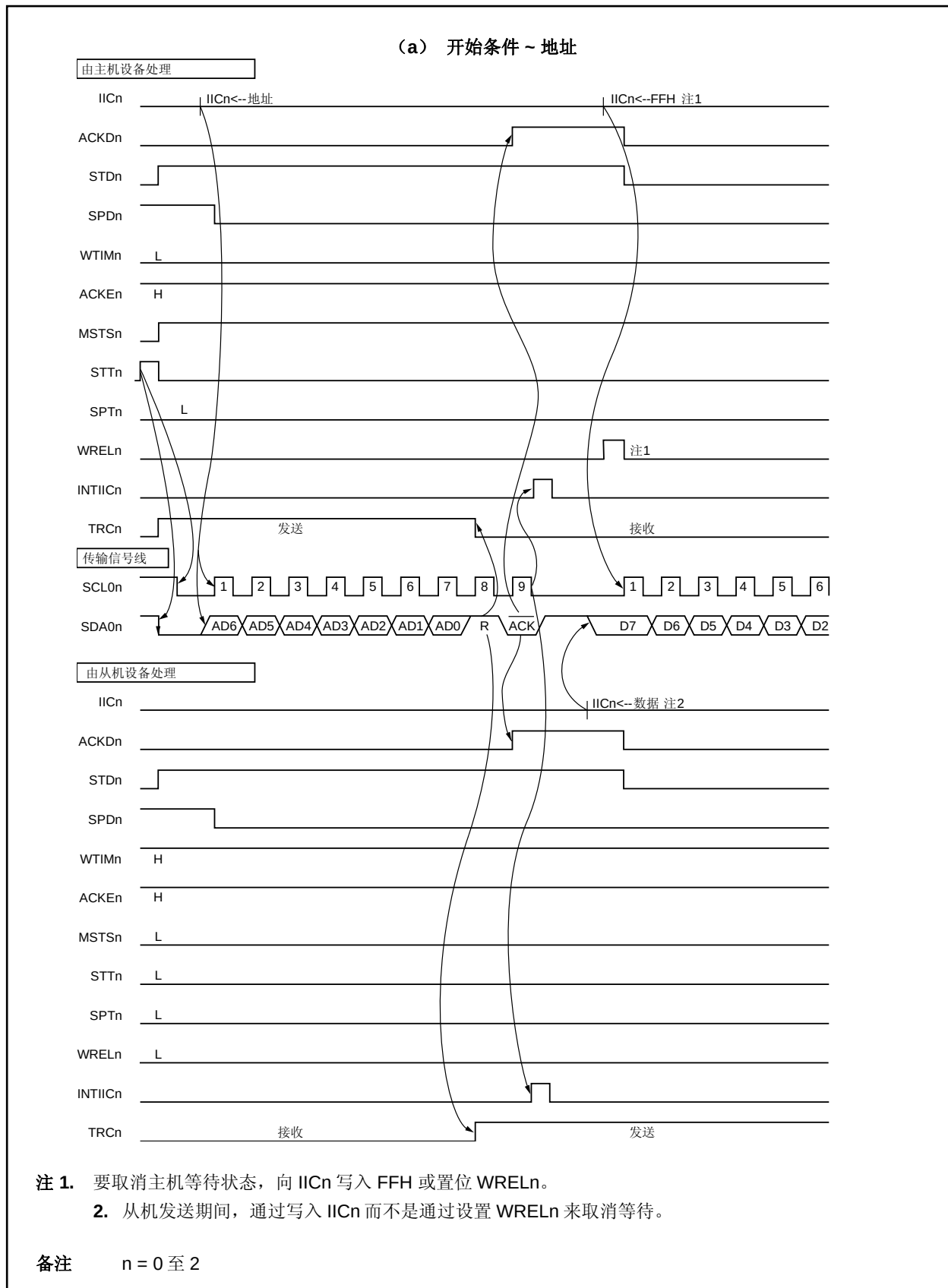
<R>

图 19-23. 主机向从机传输举例
(当主机设备和从机设备都选择 9 时钟等待时) (3/3)



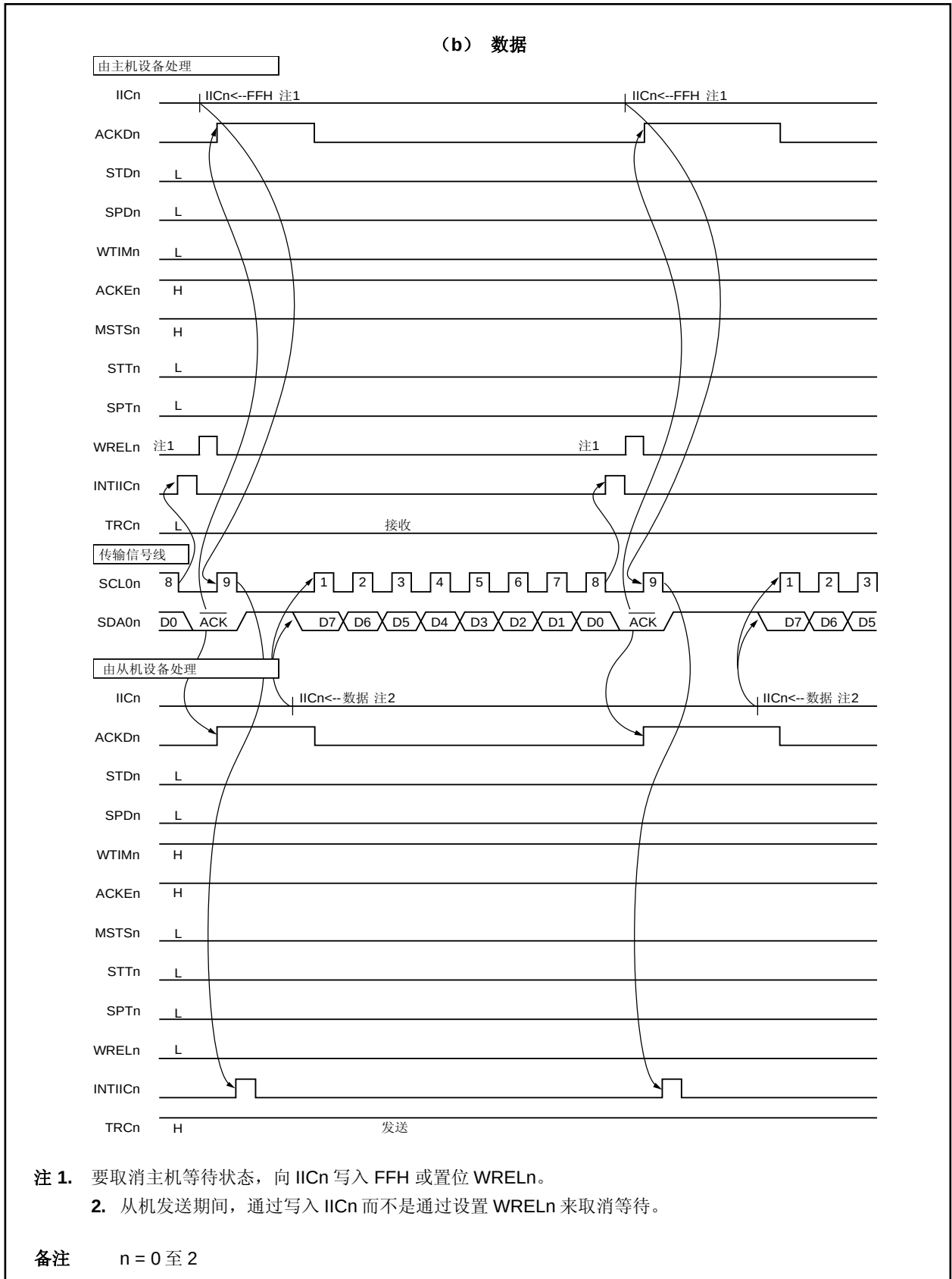
<R>

图 19-24. 从机设备向主机设备传输举例
(当选择主机 8 时钟等待而从机 9 时钟等待时) (1/3)



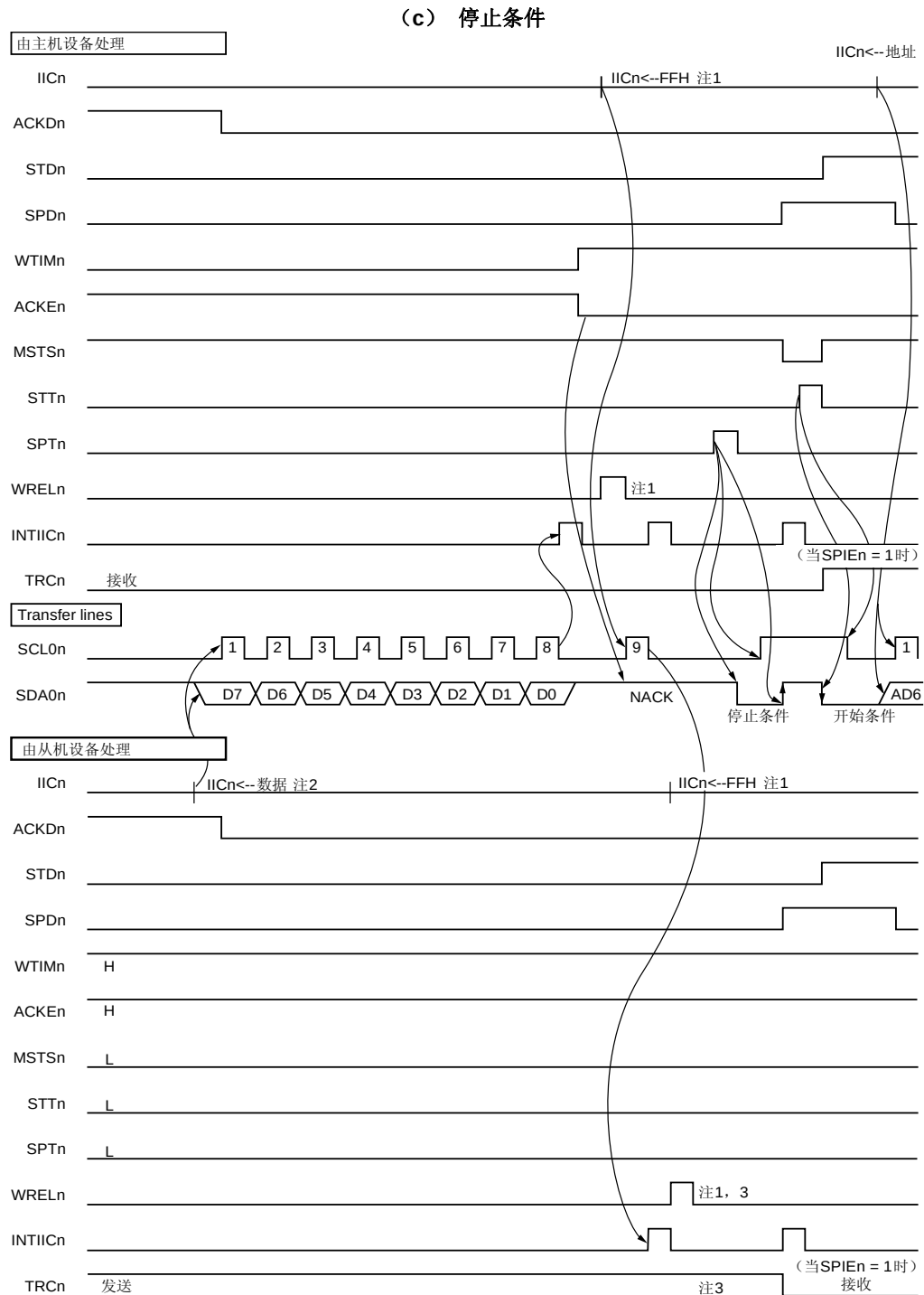
<R>

图 19-24. 从机设备向主机设备传输举例
(当选择主机 8 时钟等待而从机 9 时钟等待时) (2/3)



<R>

图 19-24. 从机设备向主机设备传输举例
(当选择主机 8 时钟等待而从机 9 时钟等待时) (3/3)



- 注 1. 要取消等待状态，向 IICn 写入 FFH 或置位 WRELn。
2. 从机发送期间，通过写入 IICn 而不是通过设置 WRELn 来取消等待。
3. 从机发送期间，当通过设置 WRELn 取消等待时，TRCn 会被清除。

备注 n = 0 至 2

第二十章 CAN控制器

20.1 概述

V850ES/JG3-H 和 V850ES/JH3-H 单片机专有一个片上 1 通道的 CAN(控制器局部网)控制器，它遵守 ISO 11898 中规定的 CAN 协议。

具有一个片上 CAN 控制器的 V850ES/JG3-H 和 V850ES/JH3 单片机产品如下：

- μ PD70F3770, 70F3771

20.1.1 特性

- 遵循 ISO11898 并依照 ISO/DIS 16845 (CAN 一致性测试)测试
- 允许标准帧和扩展帧发送/接收
- 传输速率: 最高 1Mbps(CAN 时钟输入 ≥ 8 MHz)
- 32 个报文缓存器/通道
- 接收/发送记录列表功能
- 自动块传输功能
- 多缓存接收块功能
- 每个通道可设置 4 种屏蔽模式

20.1.2 功能概述

表 20-1 给出了 CAN 控制器的功能概述。

表 20-1. 功能概述

功能	详情
协议	CAN 协议 ISO 11898 (标准帧和扩展帧发送/接收)。
波特率	最高 1 Mbps (CAN 时钟输入 $\geq$ 8 MHz)。
数据存储	报文存储在 RAM 指定的 CAN 中。
报文数目	32 个报文缓存器/通道 - 每个报文缓存器既可以设为发送报文缓存器也可以设为接收报文缓存器。
报文接收	- 每个报文缓存器可设置唯一的 ID。 - 每个通道可设置 4 种屏蔽模式。 - 可为每个报文缓存器禁止或使能接收完成中断。 - 两个或更多的接收报文缓存器可用作 FIFO 接收缓存器(多缓存接收块功能)。 - 可以记录并查找具有接收和存储报文的缓存数目 (接收记录列表功能)。
报文发送	- 每个报文缓存器可设置唯一的 ID。 - 每一个报文缓存器的发送完成中断。 0 至 7 号报文缓存器指定为发送报文缓存器, 可用于自动块传输。 报文发送间隔可编程 (自动块发送功能 (以下称为“ABT”))。 - 可以记录并查找已发送报文的缓存数目 (接收记录列表功能)。
远程帧处理	通过发送缓存器处理远程帧。
时间标记功能	- 可为接收报文设置时间标记功能。 可选择时间标志捕获触发器 (可检测 CAN 报文帧中的 SOF 或 EOF)。 - 可为发送报文设置时间标记功能。 数据域中的指定字节可用捕获时间标记^注代替。
诊断功能	- 可读错误计数器。 - 验证总线连接的“有效协议操作标志”。 - 只收模式 - 单次模式 - CAN 协议错误类型标识 - 自检模式
强制从掉线状态释放	总线关闭时可设为默认模式, 因此总线可以强制从掉线状态释放。
节电模式	- CAN 睡眠模式 (可由 CAN 总线唤醒)。 - CAN 停止模式 (不能由 CAN 总线唤醒)。

注 仅对具有高级时间标记功能的宏有效。

20.1.3 配置

CAN 控制器由以下四部分组成：

(1) NPB 接口

该功能部分提供一个 NPB(NEC 外设 I/O 总线)接口，并提供 CAN 模块和主机 CPU 的接口。

(2) MAC（存储器访问控制器）

该功能部分控制到 CAN 协议层的和 CAN 模块内部 CAN RAM 的访问。

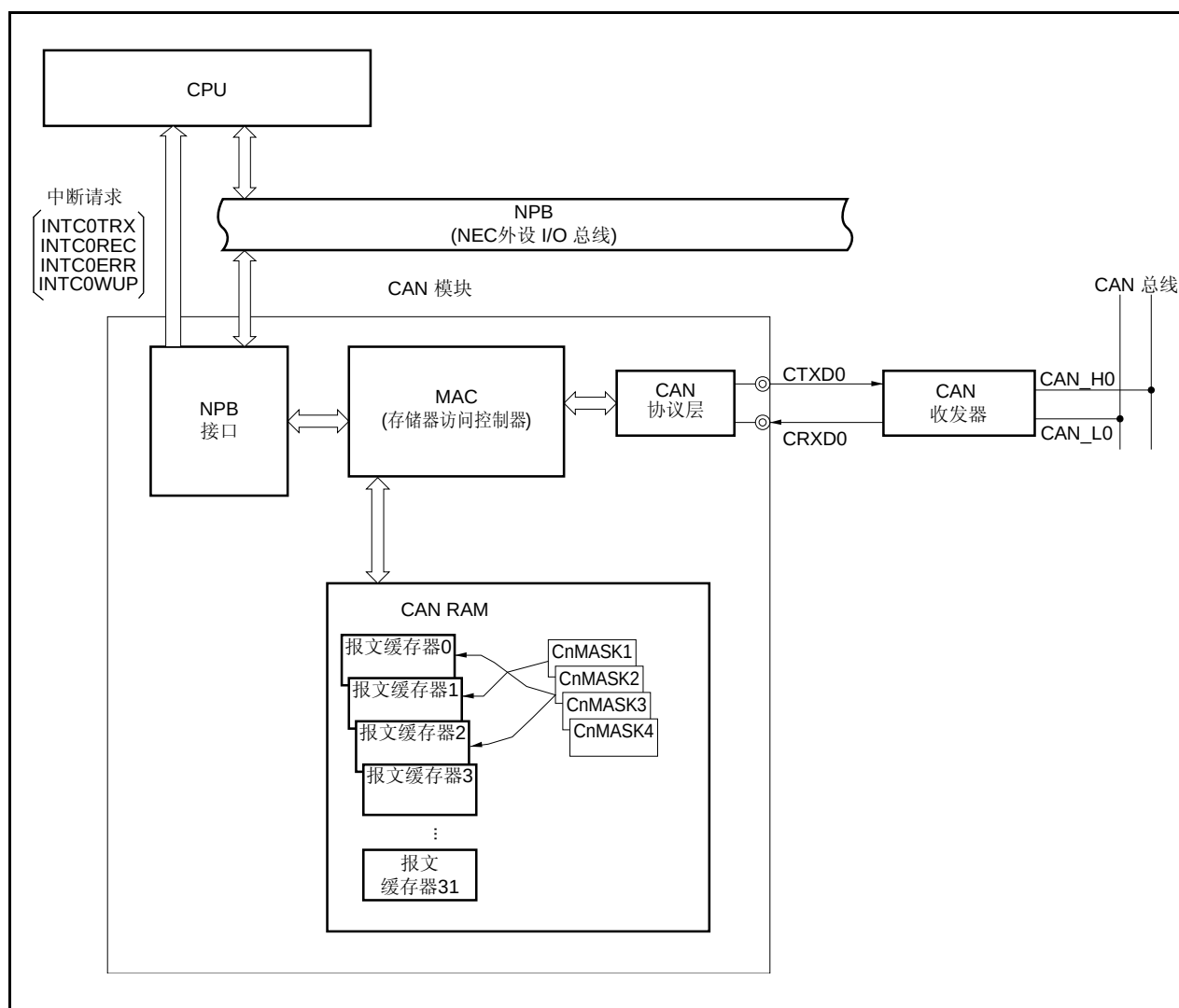
(3) CAN 协议层

该功能部分设置 CAN 协议层。

(4) CAN RAM

此为 CAN 存储器功能部分，用于存储报文 ID，报文数据等等。

图 20-1. CAN 模块框图

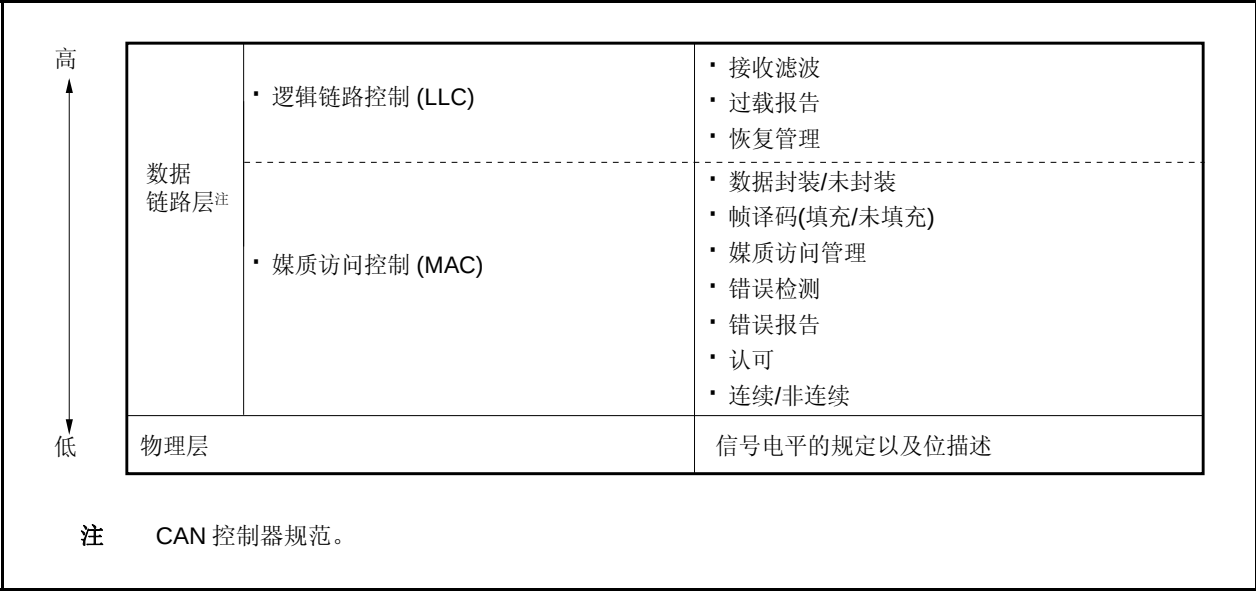


20.2 CAN 协议

CAN(控制器局域网)是应用于汽车实时通信中的高速多路复用通信协议 (C 类)。 CAN 由 ISO 11898 规定。关于详情，请参见 ISO 11898 规范。

CAN 规范通常分为两层： 物理层和数据链路层。 数据链路层包括逻辑链路和媒质访问控制。 这些层的组成图示如下：

图 20-2. 层的组成



20.2.1 帧格式

(1) 标准帧格式

- 标准格式帧使用 11 位标识符，这意味着它可以处理最多为 2048 个的报文。

(2) 扩展帧格式

- 扩展格式帧使用 29 位(11 位 + 18 位)标识符，将可处理的报文数目增加至 $2,048 \times 2^{18}$ 。
- 当仲裁区的 SRR 和 IDE 位均设为“接收电平”（CMOS 电平“1”）时，设置为扩展格式帧。

20.2.2 帧类型

CAN 协议使用下面四种类型的帧。

表 20-2. 帧类型

帧类型	说明
数据帧	用于发送数据的帧
远程帧	用于请求一个数据帧的帧
错误帧	用于报告错误检测的帧
过载帧	用于对下一数据帧或远程帧的延时

(1) 总线值

总线值分为显性和隐性。

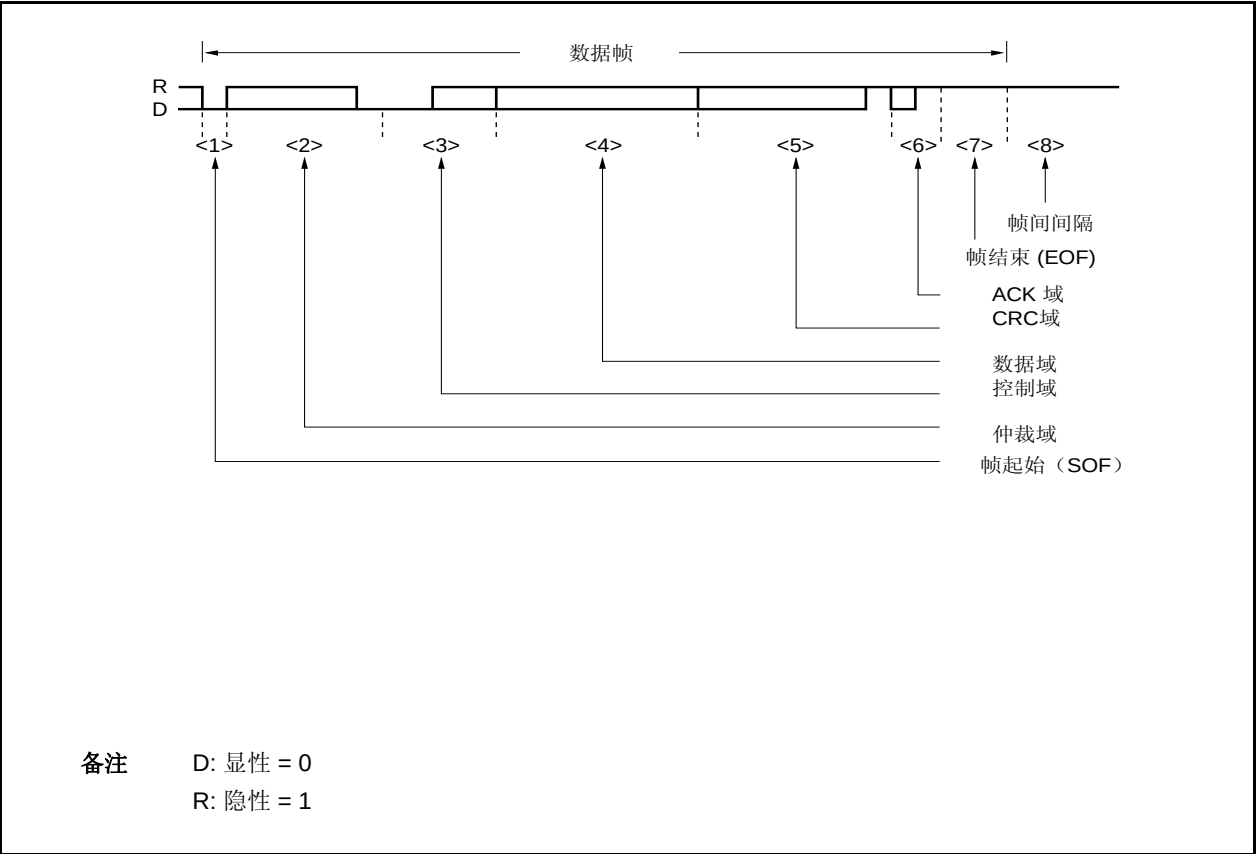
- 显性电平用逻辑 0 表示。
- 隐性电平用逻辑 1 表示。
- 当同时发送显性电平和隐性电平时，总线值变为显性电平。

20.2.3 数据帧和远程帧

(1) 数据帧

一个数据帧包含 7 个域。

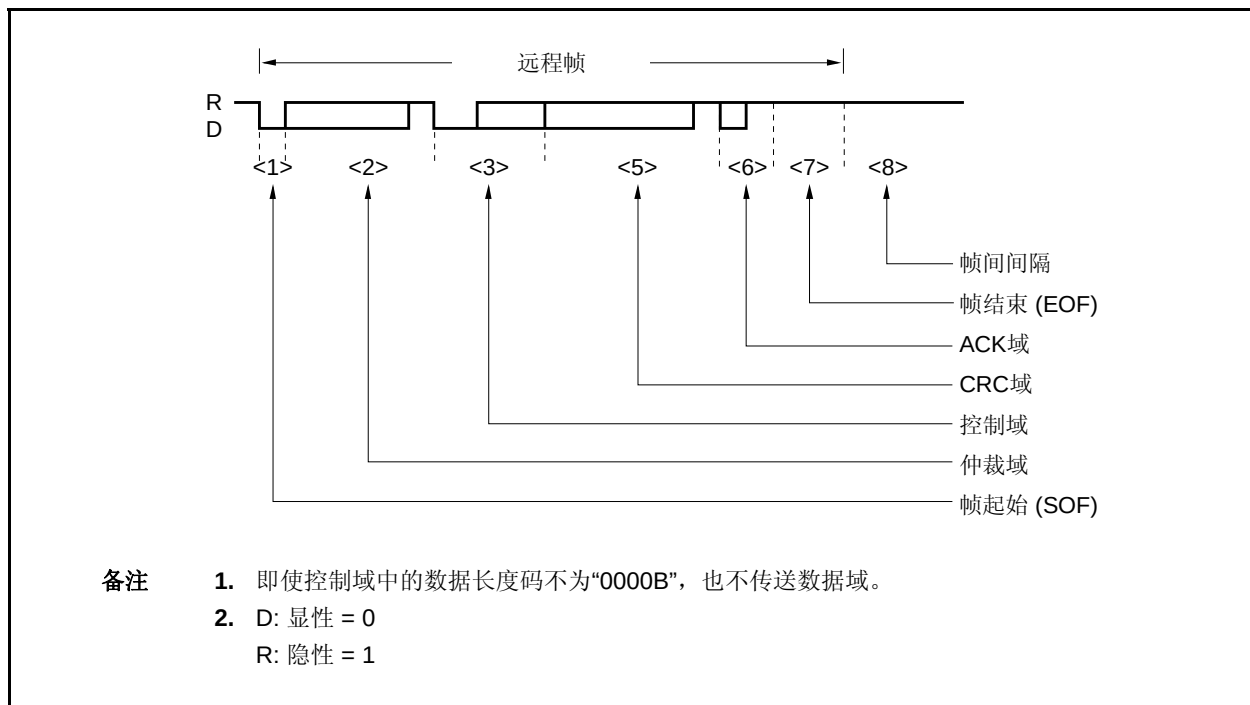
图 20-3. 数据帧



(2) 远程帧

一个远程帧包含 6 个域。

图 20-4. 远程帧

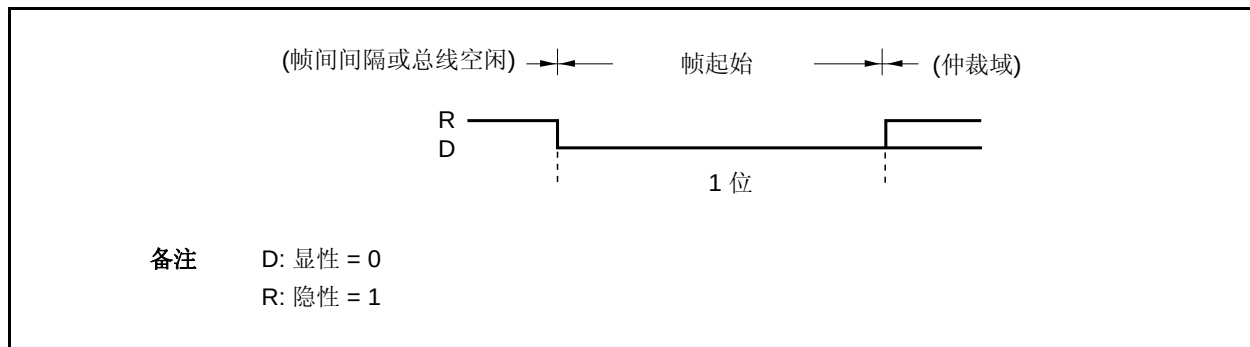


(3) 域的说明

<1> 帧起始 (SOF)

帧起始域表示一个数据帧或远程帧的开始。

图 20-5. 帧起始 (SOF)



- 若在总线空闲状态检测到显性电平，则执行硬件同步(当前 TQ 被赋值为 SYNC 段)。
- 这样的硬件同步之后如果在采样点采样到一个显性电平，则该位赋值为 SOF。如果检测到一个隐性电平，那么协议层将返回总线空闲状态并将前面的显性脉冲视为噪声。在这种情况下不会产生错误帧。

<2> 仲裁域

仲裁域用来设置优先级，数据帧/远程帧和帧格式。

图 20-6. 仲裁域 (标准格式模式)

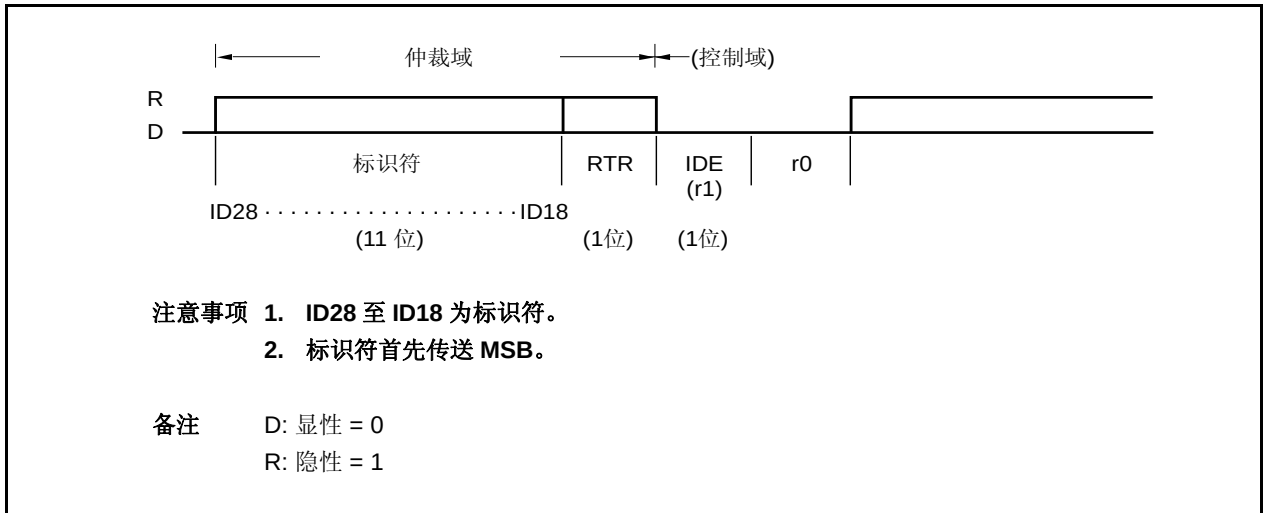


图 20-7. 仲裁域 (扩展格式模式)

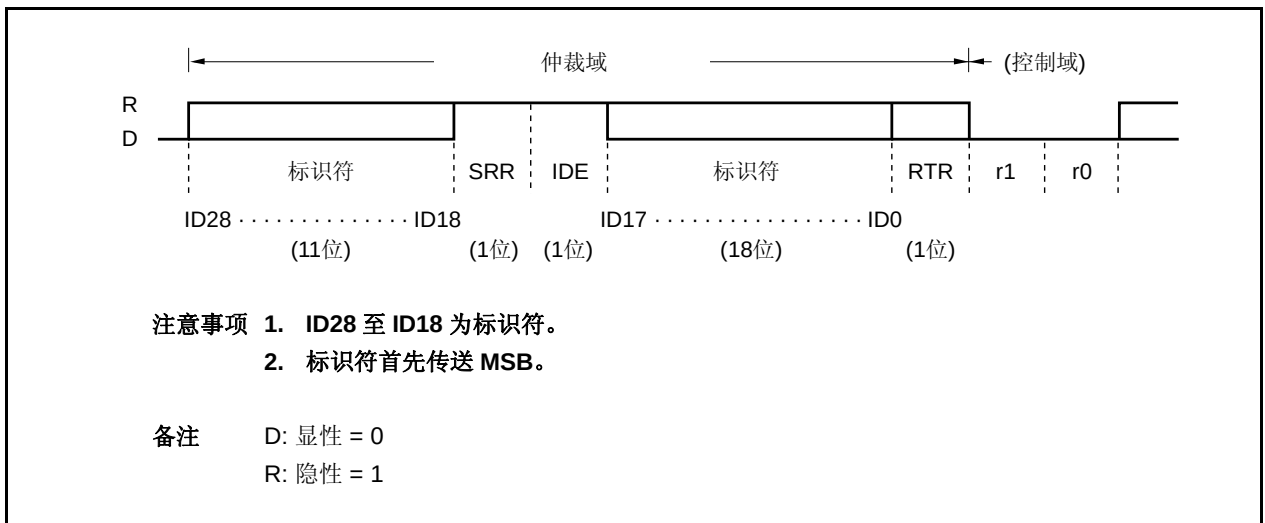


表 20-3. RTR 帧设置

帧类型	RTR 位
数据帧	0 (D)
远程帧	1 (R)

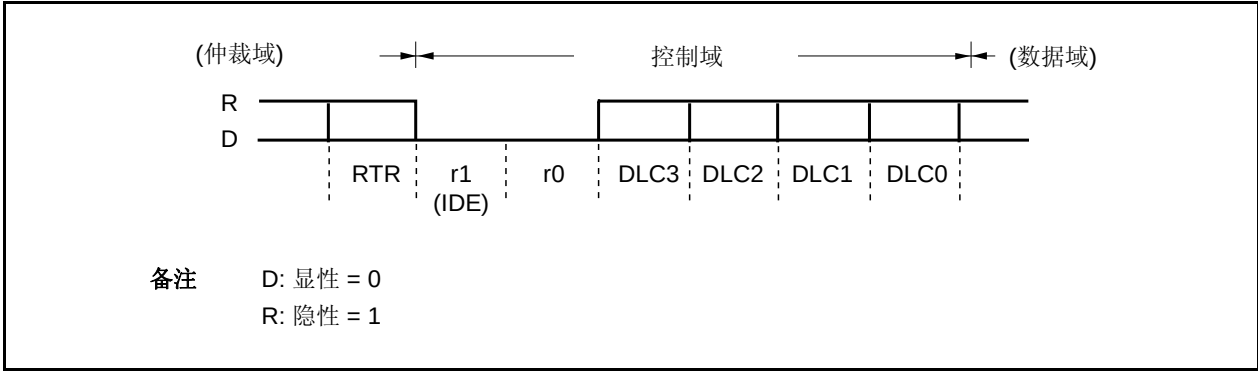
表 20-4. 帧格式设置 (IDE 位)和标识符 (ID)位的个数

帧格式	SRR 位	IDE 位	位个数
标准帧模式	无	0 (D)	11 位
扩展帧模式	1 (R)	1 (R)	29 位

<3> 控制域

控制域将“N”设为数据域的数据字节个数（N = 0 至 8）。

图 20-8. 控制域



在标准格式帧中，控制域的 IDE 位与 r1 位相同。

表 20-5. 数据长度设置

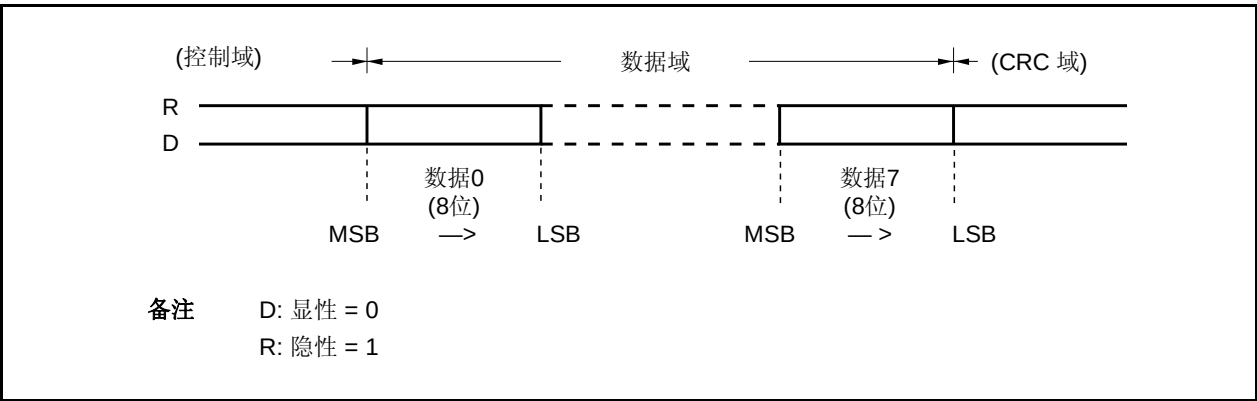
数据长度码				数据字节计算
DLC3	DLC2	DLC1	DLC0	
0	0	0	0	0 个字节
0	0	0	1	1 个字节
0	0	1	0	2 个字节
0	0	1	1	3 个字节
0	1	0	0	4 个字节
0	1	0	1	5 个字节
0	1	1	0	6 个字节
0	1	1	1	7 个字节
1	0	0	0	8 个字节
其它				8 个字节而与 DLC3 至 DLC0 的值无关

注意事项 在远程帧中，即使数据长度码不为 0000B，也没有数据域。

<4> 数据域

数据域包含由控制域设定的数据量（字节单元）。 最多可设定 8 个数据单元。

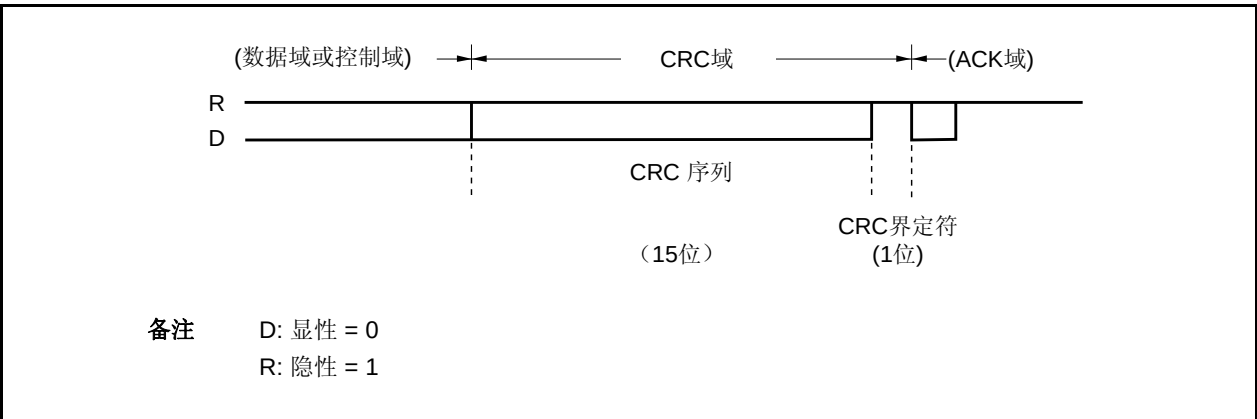
图 20-9. 数据域



<5> CRC 域

CRC 是用于检查发送数据错误的 16 位的域。

图 20-10. CRC 域

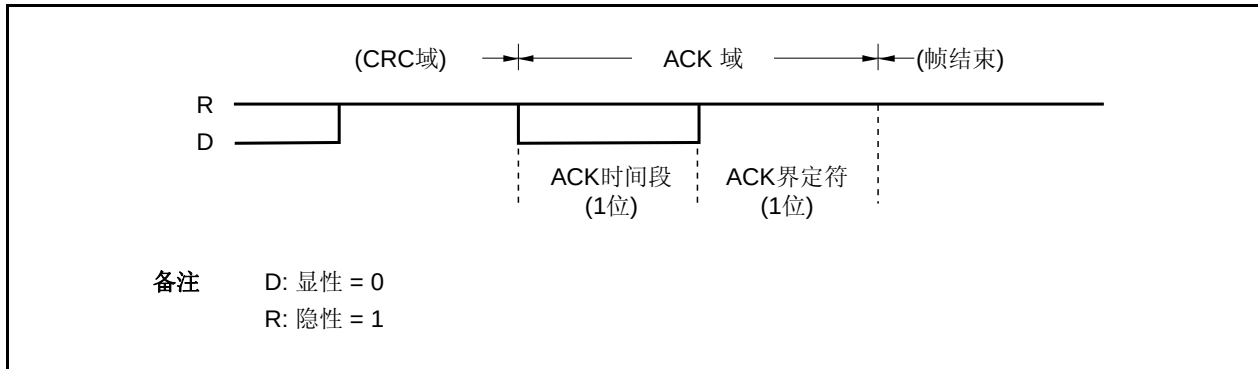


- P(X)多项式用于产生 15 位的 CRC 序列，序列表示如下：
$$P(X) = X^{15} + X^{14} + X^{10} + X^8 + X^7 + X^4 + X^3 + 1$$
- 发送节点：发送 CRC 序列，该序列由帧起始、仲裁域、控制域和数据域中的数据（位填充之前）计算。
- 接收节点：使用不包括填充位的数据位，对接收数据中计算得到的 CRC 序列和 CRC 域中的 CRC 序列进行比较。
如果两个 CRC 序列不匹配，则该节点产生一个错误帧。

<6> ACK 域

ACK 域用来确认正常的接收。

图 20-11. ACK 域

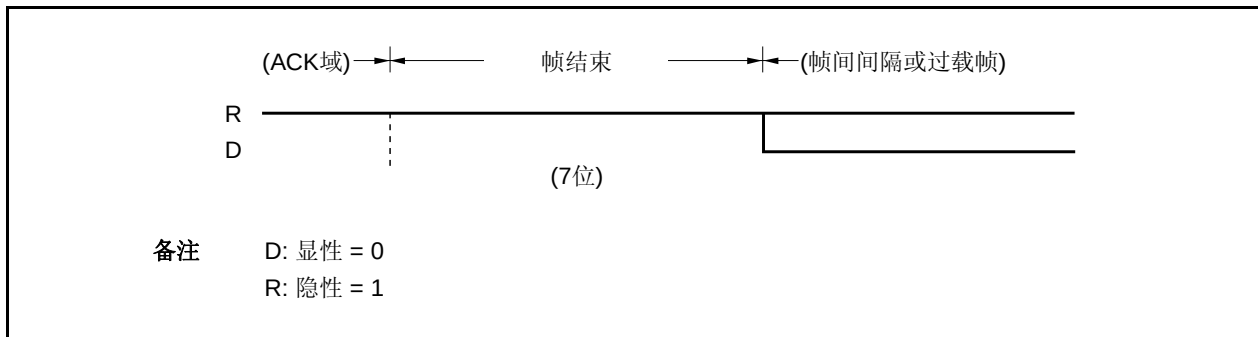


- 如果没有检测到 CRC 错误，则接收节点将 ACK 时间段设为显性电平。
- 发送节点输出两个隐性电平位。

<7> 帧结束(EOF)

帧结束域表示数据帧/远程帧的结束。

图 20-12. 帧结束(EOF)



<8> 帧间间隔

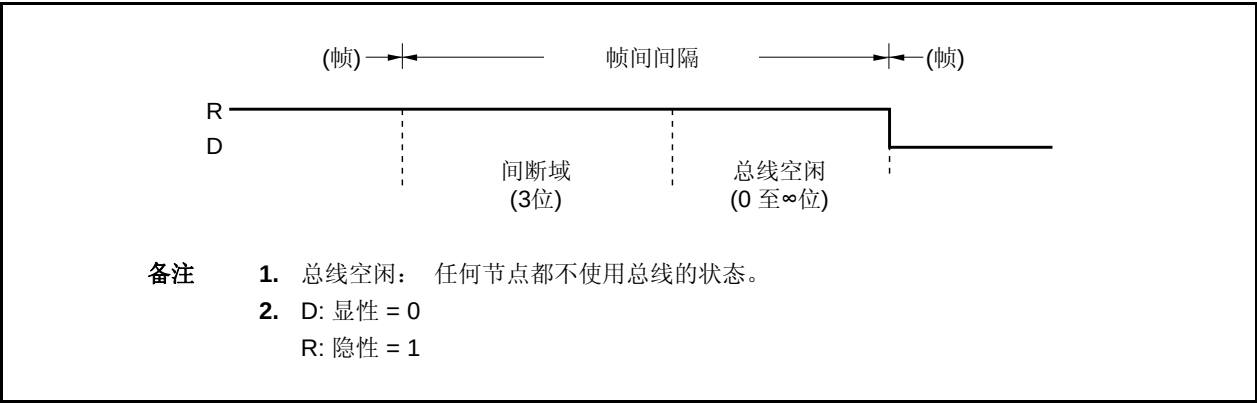
帧间间隔在数据帧、远程帧、错误帧或过载帧之后插入，以使一帧与下一帧隔开。

- 总线状态根据错误状态不同而异。

(a) 错误生成节点

帧间间隔由一个 3 位间断域和一个总线空闲域组成。

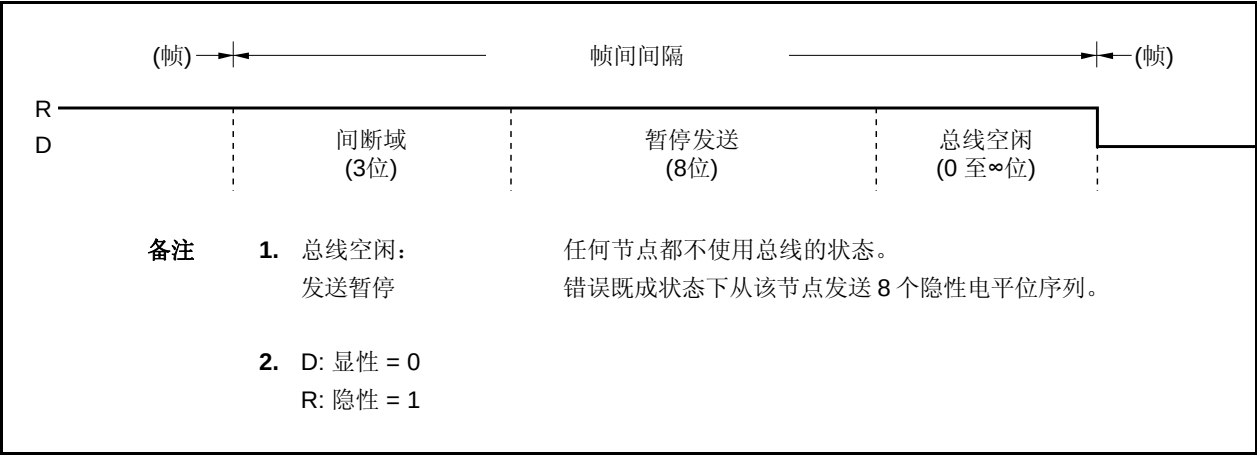
图 20-13. 帧间间隔 (错误生成节点)



(b) 错误既成节点

帧间间隔由间断域、发送暂停域和总线空闲域组成。

图 20-14. 帧间间隔 (错误既成节点)



间断域通常为 3 位。然而，如果发送节点在间断域的第 3 位检测到一个显性电平，它仍然执行发送。

- 错误状态下的操作。

表 20-6. 错误状态下的操作

错误状态	操作
错误生成	该状态下的节点可在 3 位间断之后立刻发送。
错误既成	该状态下的节点可间断之后可发送 8 位。

20.2.4 错误帧

节点检测到错误之后输出错误帧。

图 20-15. 错误帧

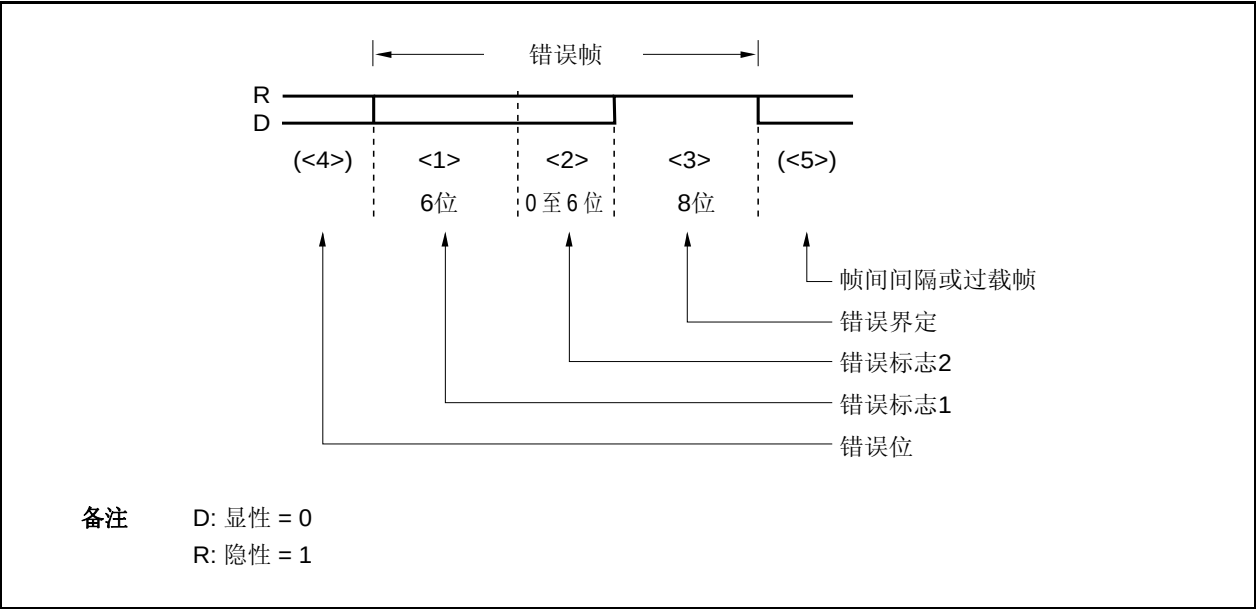


表 20-7. 错误帧域的说明

No.	名称	位计数	说明
<1>	错误标志 1	6	错误生成节点：连续输出 6 个显性电平比特。 错误既成节点：连续输出 6 个隐性电平比特。 如果在一个节点输出既成错误标志时，另外一个节点输出一个显性电平，则既成错误标志不被清除直到在 1 行中检测到相同的 6 比特电平。
<2>	错误标志 2	0 至 6	接收错误标志 1 的节点检测位填充错误并产生此错误标志。
<3>	错误界定符	8	连续输出 8 个隐性电平比特。 如果在第 8 位检测到显性电平，则从下一位发送一个过载帧。
<4>	错误位	—	检测到错误的位。 从错误位的下一位输出错误标志。 在 CRC 错误的情况下，此位在 ACK 界定符之后输出。
<5>	帧间间隔/过载帧	—	由此产生帧间间隔或过载帧。

20.2.5 过载帧

在以下条件下发送过载帧。

- 接收节点没有完成接收操作时注
- 如果间断期间在前两位检测到显性电平
- 如果在帧结束的最末位（第 7 位）或错误界定符/过载界定符的最末位（第 8 位）检测到显性电平

因为该 CAN 控制器中内部处理速度高，因此所有接收帧都可以在不输出过载帧的情况下载入。

图 20-16. 过载帧

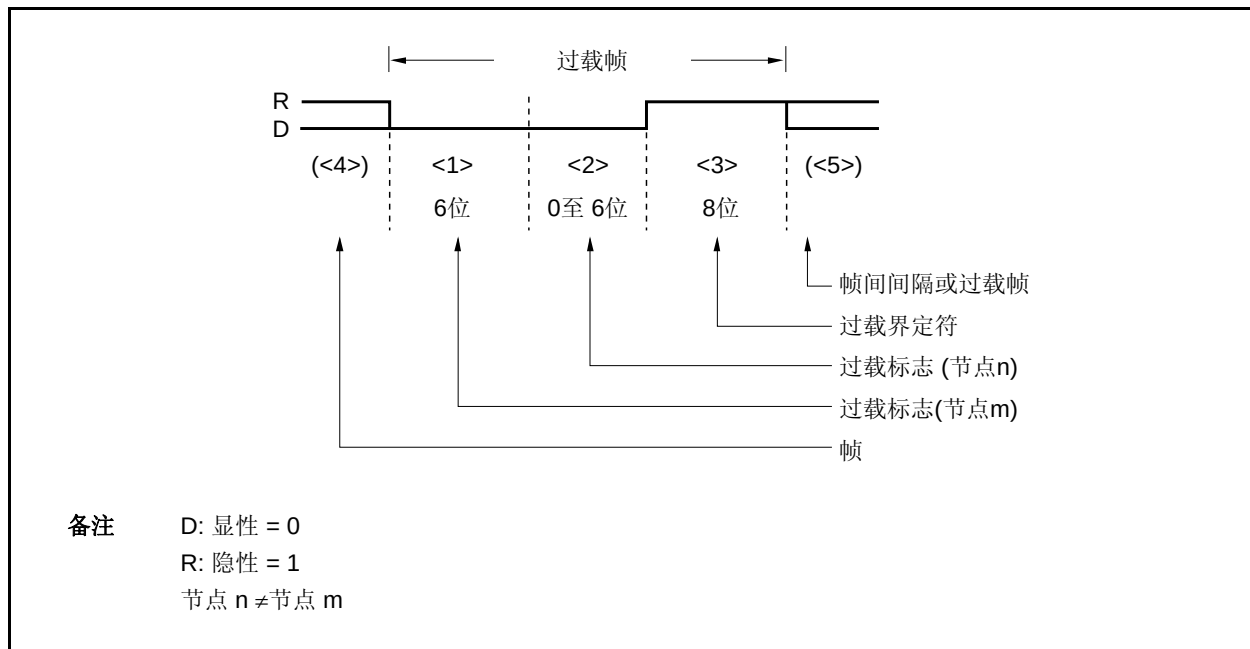


表 20-8. 过载帧域的说明

编号	名称	位计数	说明
<1>	过载标志	6	连续输出 6 个显性电平比特。
<2>	来自其它节点的过载标志	0 至 6	帧间间隔中接收过载标志的节点输出一个过载标志。
<3>	过载界定符	8	连续输出 8 个隐性电平比特。 如果在第 8 位检测到显性电平，则从下一位发送一个过载帧。
<4>	帧	—	在帧结束、错误界定符或过载界定符之后输出。
<5>	帧间间隔/过载帧	—	由此产生帧间间隔或过载帧。

20.3 功能

20.3.1 决定总线优先级

- (1) 当一个节点开始发送时:
- 总线空闲期间, 首先输出数据的节点发送数据。
- (2) 当不止一个节点开始发送时:
- 从仲裁域第 1 位开始连续输出显性电平最长的节点具有总线优先级(如果同时发送显性电平和隐性电平, 则显性电平作为总线的值)。
 - 发送节点比较其输出的仲裁域和总线的数据电平。

表 20-9. 决定总线优先级

电平匹配	连续发送
电平不匹配	停止数据输出后, 在检测到不匹配的位之后的位接收

- (3) 数据帧和远程帧的优先级
- 当数据帧和远程帧在总线上冲突时, 数据帧具有优先级, 因为它的 RTR 位即仲裁域的最后一位承载了一个显性电平。

备注 如果扩展格式数据帧和标准格式远程帧在总线上发生冲突(如果它们的 ID28 至 ID18 全部相同), 标准格式数据帧优先。

20.3.2 位填充

如果相同的电平持续 5 位, 位填充利用增加一位反相电平数据来建立同步, 以防止突发错误。

表 20-10. 位填充

发送	数据帧或远程帧发送期间, 当帧起始和 CRC 域之间的数据连续 5 位为相同电平时, 在下一位之前插入一个反相电平数据比特。
接收	数据帧或远程帧接收期间, 当帧起始和 CRC 域之间的数据连续 5 位为相同电平时, 在删除下一位之后继续接收。

20.3.3 多主机方式

因为总线的优先级(获得发送权的节点)取决于标识符, 因此任何节点都可以作为总线主控。

20.3.4 多播方式

尽管只有一个发送节点, 但是两个或更多的节点可以同时接收相同的数据, 因为可以为两个或更多的节点设置相同的标识符。

20.3.5 CAN 睡眠模式/CAN停止模式功能

CAN 睡眠模式/CAN 停止模式功能可使 CAN 控制器进入等待模式，以获得低功耗。

总线操作可以从 CAN 睡眠模式中唤醒控制器，但是不能从 CAN 停止模式唤醒控制器(CAN 停止模式由 CPU 存取控制)。

20.3.6 错误控制功能

(1) 错误类型

表 20-11. 错误类型

类型	错误的定义		检测状态	
	检测方式	检测条件	发送/接收	域/帧
位错误	比较输出电平和总线电平（除了填充位之外）	电平不匹配	发送/接收节点	总线上从帧起始输出数据至帧结束、错误帧和过载帧的位
填充错误	检查填充位的接收数据	6 个连续相同的输出电平的位	接收节点	帧起始至 CRC 序列
CRC 错误	比较接收数据生成的 CRC 序列和已接收的 CRC 序列	CRC 不匹配	接收节点	CRC 域
格式错误	固定格式的域/帧检查	检测固定格式的破坏	接收节点	• CRC 界定符 • ACK 域 • 帧结束 • 错误帧 • 过载帧
ACK 错误	发送节点检查 ACK 段	ACK 段的隐性电平检测	发送节点	ACK 段

(2) 错误帧的输出时序

表 20-12. 错误帧的输出时序

类型	输出时序
位错误、填充错误、格式错误、ACK 错误	在检测到错误的下一位开始输出错误帧。
CRC 错误	在 ACK 界定符的下一位开始输出错误帧。

(3) 错误情况的处理

错误帧之后发送节点重新发送数据帧或远程帧。（但是在单脉冲模式下，不重新发送帧。）

(4) 错误状态

(a) 错误状态的类型

CAN 规范定义了下面 3 种类型的错误状态：

- 错误生成
- 错误既成
- “掉线”

通过如表 20-13 所示的 C0ERC.TEC7 至 C0ERC.TEC0 位（发送错误计数器位）和 C0ERC.REC6 至 C0ERC.REC0 位（接收错误计数器位）的值来划分这些错误状态的类型。

由 C0INFO 寄存器指定当前的错误状态。

当各错误计数器的值等于或大于错误告警电平（96）时，将 C0INFO.TECS0 或 C0INFO.RECS0 位设为 1。因为在这种情况下总线被认为具有严重的错误，所以必须测试总线状态。错误计数器的值为 128 或更多则表明为错误既成状态，并且 TECS1 或 RECS1 位设为 1。

- 如果发送错误计数器的值大于或等于 256（实际上，发送错误计数器不能表示一个大于或等于 256 的值），则达到掉线敏状态且将 C0INFO.BOFF 位设为 1。
- 如果在总线启动时仅一个节点在使用中(比如，当总线仅连接本机时)，则即使发送了数据也不返回到 ACK。因此，重复进行错误帧和数据的重传。但是在错误既成状态下，发送错误计数器不增加且不进入掉线敏状态。

表 20-13. 错误状态的类型

类型	操作	错误计数器的值	C0INFO 寄存器显示	错误状态具体操作
错误生成	发送	0 至 95	TECS1, TECS0 = 00	• 检测到错误时输出生成错误标志(6 个连续显性电平比特)。
	接收	0 至 95	RECS1, RECS0 = 00	
	发送	96 至 127	TECS1, TECS0 = 01	
	接收	96 至 127	RECS1, RECS0 = 01	
错误既成	发送	128 至 255	TECS1, TECS0 = 11	• 检测到错误时输出既成错误标志 (6 个连续隐性电平比特)。 • 间断(暂停发送)之后的发送期间发送 8 个隐性电平位。
	接收	128 或更多	RECS1, RECS0 = 11	
掉线	发送	256 或更多 (不显示) ^注	BOFF = 1, TECS1, TECS0 = 11	• 不能通信。 接收到帧时，不存储报文而执行下列操作。 <1> TSOUT 切换。 <2> REC 增量/减量。 <3> 设置 VALID 位。 • 如果设置了初始化模式，且在除了初始化模式之外的操作模式中连续 128 次在一行内生成 11 个隐性电平位，则错误计数器复位为 0 并可恢复错误生成状态。

注 如果设置了 BOFF 位，则发送错误计数器 (TEC) 的值不代表任何意义。
当计数器的值在 248 至 255 范围之间时，如果发生一个会使发送错误计数器的值增 8 的错误，那么计数器不增加并认定为掉线状态。

(b) 错误计数器

发生错误时，错误计数器向上计数，而当发送和接收成功时向下计数。 错误计数器在错误界定符的第一位开始向上或向下计数。

表 20-14. 错误计数器

状态	发送错误计数器 (TEC7 至 TEC0 位)	接收错误计数器 (REC6 至 REC0 位)
接收节点检测到错误(除了生成错误标志或过载标志的位错误)。	不改变	+1 (REPS 位 = 0)
接收节点在错误帧的错误标志后检测到显性电平。	不改变	+8 (REPS 位 = 0)
发送节点发送一个错误标志。 [作为例外，下列情况下错误计数器不改变] <1> 在错误生成状态会检测到 ACK 错误，且正输出生成错误标志时不会检测到显性电平。<2> 在发送隐性电平的仲裁域检测到填充错误作为填充位，但是检测到显性电平位。	+8	不改变
输出生成错误标志或过载标志期间检测到位错误(错误生成发送模式)	+8	不改变
输出生成错误标志或过载标志期间检测到位错误(错误生成接收模式)	不改变	+8 (REPS 位 = 0)
当节点从生成错误标志或过载标志开始检测到 14 个连续显性电平位时，随后检测到 8 个显性电平位。 在既成错误标志之后检测到 8 个连续显性电平时。	+8 (发送)	+8 (接收, REPS 位 = 0)
当发送节点无错误的完成发送时(如果错误计数器 = 0，则±0)。	-1	不改变
当接收节点无错误的完成接收时。	不改变	• -1 ($1 \leq \text{REC6 至 REC0} \leq 127$, REPS 位 = 0) • ±0 (REC6 至 REC0 = 0, REPS 位 = 0) • 设置 119 至 127 的任何值 (REPS 位 = 1)

(c) 间断时发生位错误

产生一个过载帧。

注意事项 如果产生错误，则要根据错误发生之前发送错误计数器和接收错误计数器的内容控制错误。 错误标志输出之后错误计数器的值增加。

(5) 由掉线状态恢复

当 CAN 模块处于掉线状态时，由 CAN 总线切断的发送引脚(CTXD0)总是输出隐性电平。

CAN 模块按照以下的掉线恢复顺序从掉线状态中恢复。

- <1> 请求进入 CAN 初始化模式
- <2> 请求进入 CAN 操作模式
 - (a) 通过正常恢复顺序恢复操作
 - (b) 跳过恢复顺序强行恢复操作

(a) 通过正常恢复顺序从掉线状态恢复

CAN 模块首先产生一个进入初始化模式的请求(参见图 20-17 中时序<1>)。该请求立刻得到响应，并将 C0CTRL.OPMODE 位清除为 000B。诸如分析使用应用软件引起掉线状态的错误，重新定义 CAN 模块和报文缓存器或终止 CAN 模块的操作等处理，可以通过将 C0GMCTRL.GOM 位清为 0 来执行。

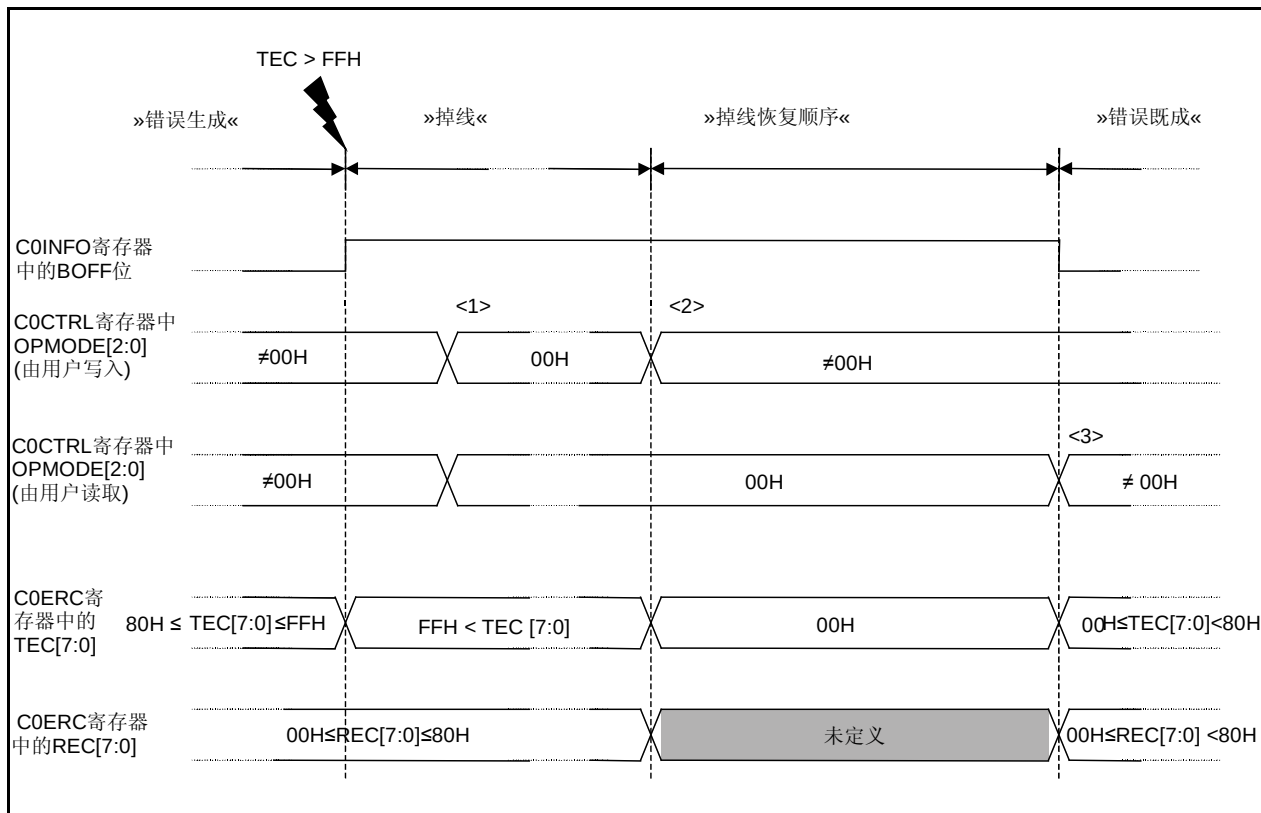
接下来，模块请求从初始化模式变换为一个操作模式(参见图 20-17 中时序<2>)。这会启动一个将 CAN 模块从掉线状态恢复的操作。CAN 协议 ISO 11898 定义了可从掉线状态恢复的条件，并且要求检测 11 个连续隐性电平位 128 次以上。此时，将模式变为一个操作模式的请求被挂起直到满足恢复条件为止。当满足恢复条件时(参见图 20-17 中时序<3>)，CAN 模块可以进入到它申请的操作模式。在 CAN 模块进入该操作模式之前，它一直处于初始化模式。可通过读取 OPMODE 位来确定 CAN 模块是否已进入操作模式。

在掉线期间和掉线恢复顺序中，C0INFO.BOFF 位保持设置(至 1)。在掉线恢复顺序中，接收错误计数器(C0ERC.REC0 至 C0ERC.REC6)对总线商检测到的 11 个连续电平位的次数进行计数。因此，可通过读取 REC0 至 REC6 位检测恢复状态。

注意事项 在掉线恢复序列中，每次检测到 11 个连续隐性电平位时 REC0 至 REC6 位向上计数(+1)。

即使在掉线期间，CAN 模块也能进入 CAN 睡眠模式或 CAN 停止模式。为了从掉线状态解除，模块必须进入一次初始化模式。然而如果模块处于 CAN 睡眠模式或 CAN 停止模式，它不能直接进入初始化模式。这种情况下，即使不切换至初始化模式，也可以在解除 CAN 睡眠模式的同时启动掉线恢复顺序。除了通过软件将 C0CTRL.PSMODE1 和 C0CTRL.PSMODE0 清除之外，也可以根据 CAN 总线上利用显性边沿检测的唤醒功能来启动掉线恢复程序。(当提供 CAN 时钟时，在检测到显性边沿之后必须通过软件清除 C0CTRL.PSMODE0 位)。

图 20-17. 通过正常恢复顺序从掉线状态恢复



(b) 跳过掉线恢复顺序的强行恢复操作

CAN 模块可通过跳过掉线顺序强行从掉线状态中解除，而不考虑总线的状态。这里给出步骤。

首先，CAN 模块请求进入初始化模式。关于操作和此时应该注意的部分，参见 20.3.6 (5) (a) 通过正常恢复顺序从掉线状态恢复。

然后模块请求进入一个操作模式。同时，C0CTRL.CCERC 位必须设为 1。

于是，跳过由 CAN 协议 ISO 11898 定义的掉线敏顺序，然后模块直接进入操作模式。这种情况下，在监测到 11 个连续隐性电平后该模块连接到 CAN 总线。关于细节，参见图 20-53 中的时序。

注意事项 CAN 协议 ISO 11898 不定义该功能。当使用该功能时，应彻底估算其对网络系统的影响。

(6) 在初始化模式中将 CAN 模块错误计数器寄存器 (C0ERC) 初始化

如果有必要将 C0ERC 和 C0INFO 寄存器初始化来调试或评测程序，可以在初始化模式下通过设置 C0CTRL.CCERC 位将它们初始化为默认值。完成初始化之后，CCERC 位自动清除为 0。

注意事项 1. 仅在初始化模式下允许该功能。在 CAN 操作模式下，即使设置 CCERC 位为 1，C0ERC 和 C0INFO 寄存器也不会初始化。

2. 请求进入 CAN 操作模式的同时可以设置 CCERC 位。

20.3.7 波特率控制功能

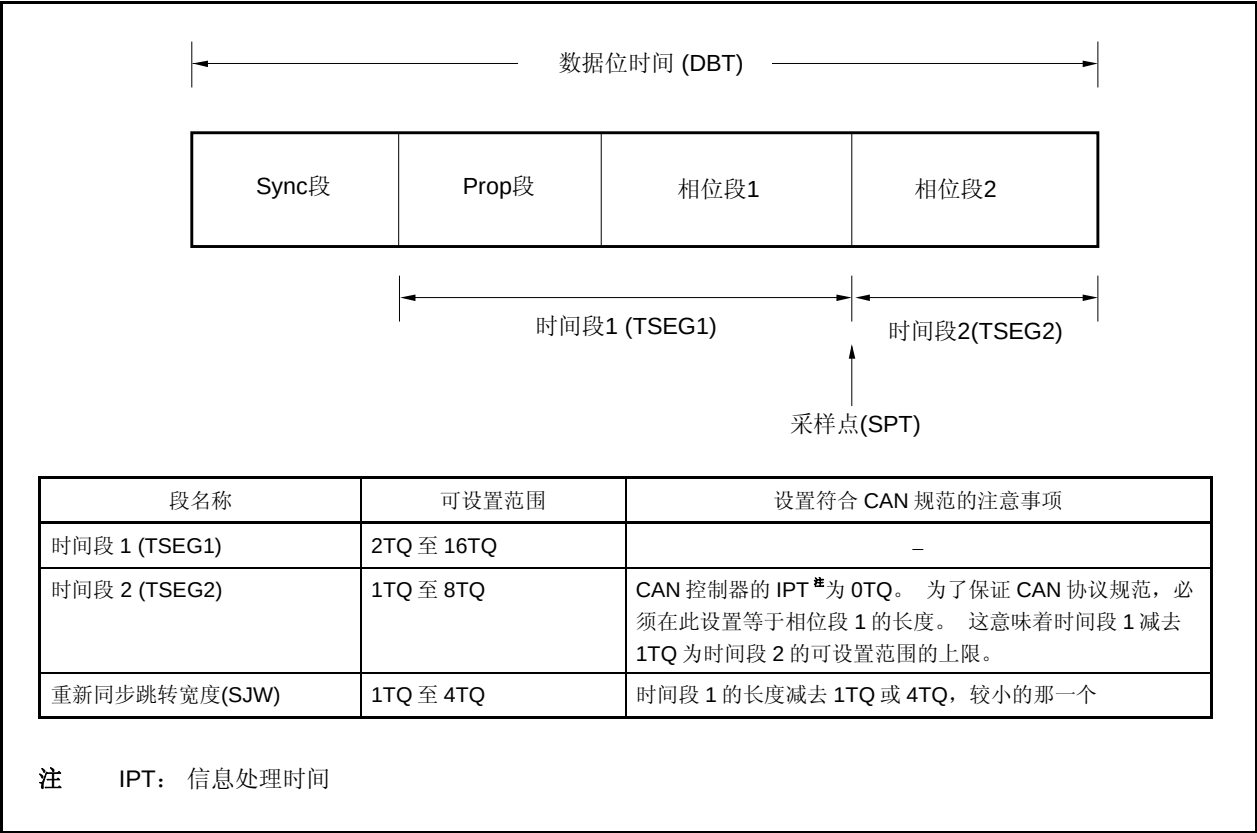
(1) 预分频器

CAN 控制器为 CAN 提供一个将时钟(f_{CAN})分频的预分频器。该预分频器产生一个 CAN 协议层基础时钟(f_{TQ})，它由 CAN 模块系统时钟(f_{CANMOD})经 1 至 256 分频后得到（参见 20.6（12）CAN0 模块位速率预分频器寄存器（C0BRP））。

(2) 数据位时间(8 至 25 个时间量子)

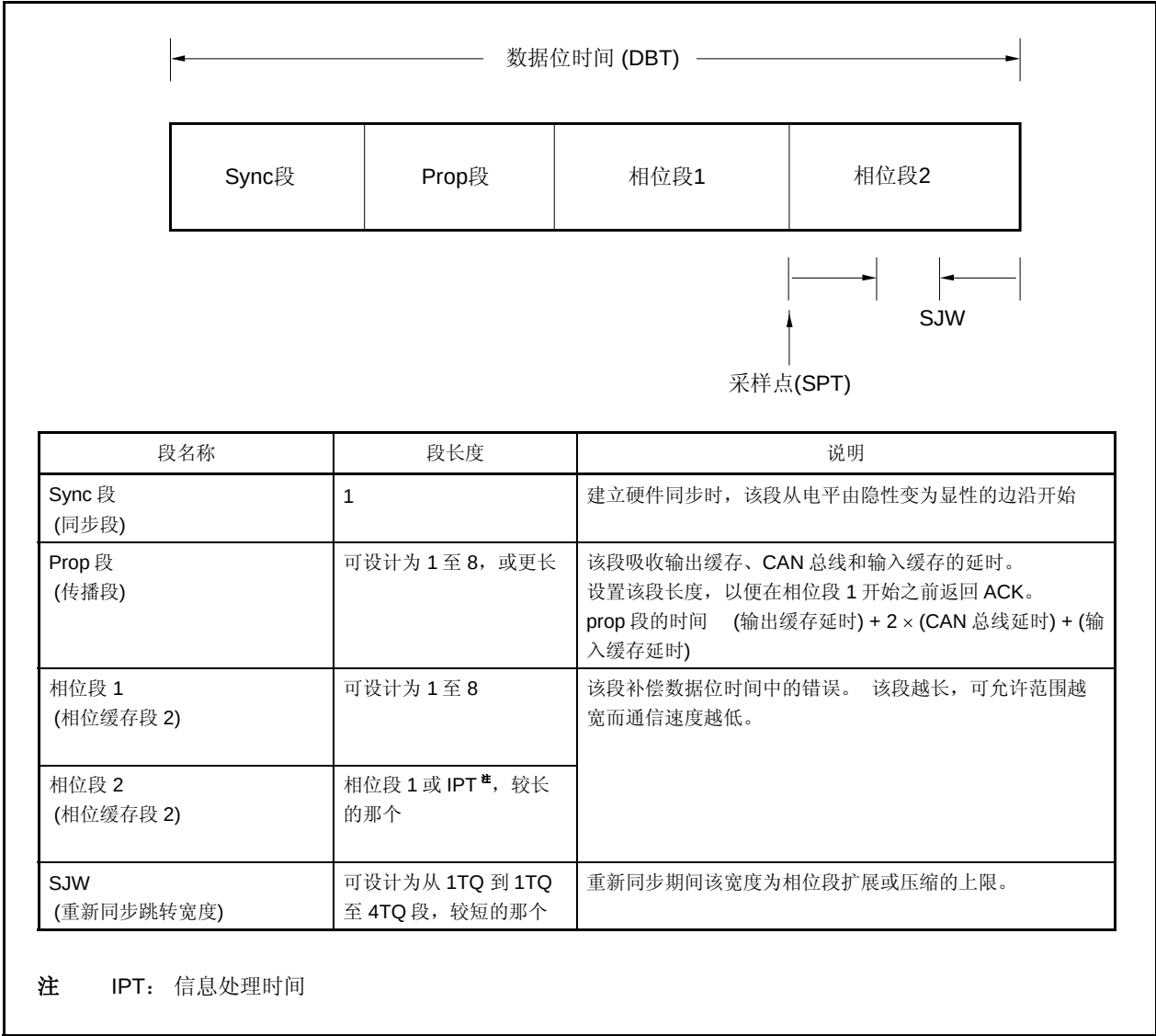
数据位时间如图 20-18 所示进行定义。CAN 控制器将时间段 1、时间段 2 和重新同步跳转宽度（SJW）设为数据位时间，如图 20-18 所示。时间段由 CAN 协议规范定义，与传播段（prop）和相位段 1 的总时间相等。时间段 2 等于相位段 2。

图 20-18. 段设置



备注CAN 协议规范定义的组成数据位时间的段如图 20-19 所示。

图 20-19. 由 CAN 规范定义的数据位时间的构成



(3) 同步数据位

- 因为接收节点没有同步信号，所以它在总线上利用电平的改变建立同步。
- 发送节点与发送节点的位时序同步发送数据。

(a) 硬件同步

当接收节点在帧间间隔检测到帧起始时建立硬件同步。

- 当在总线上检测到下降沿时，TQ 表示 sync 段且下一段为 prop 段。 这时不考虑 SJW 而建立同步。

图 20-20. 数据位同步的调整

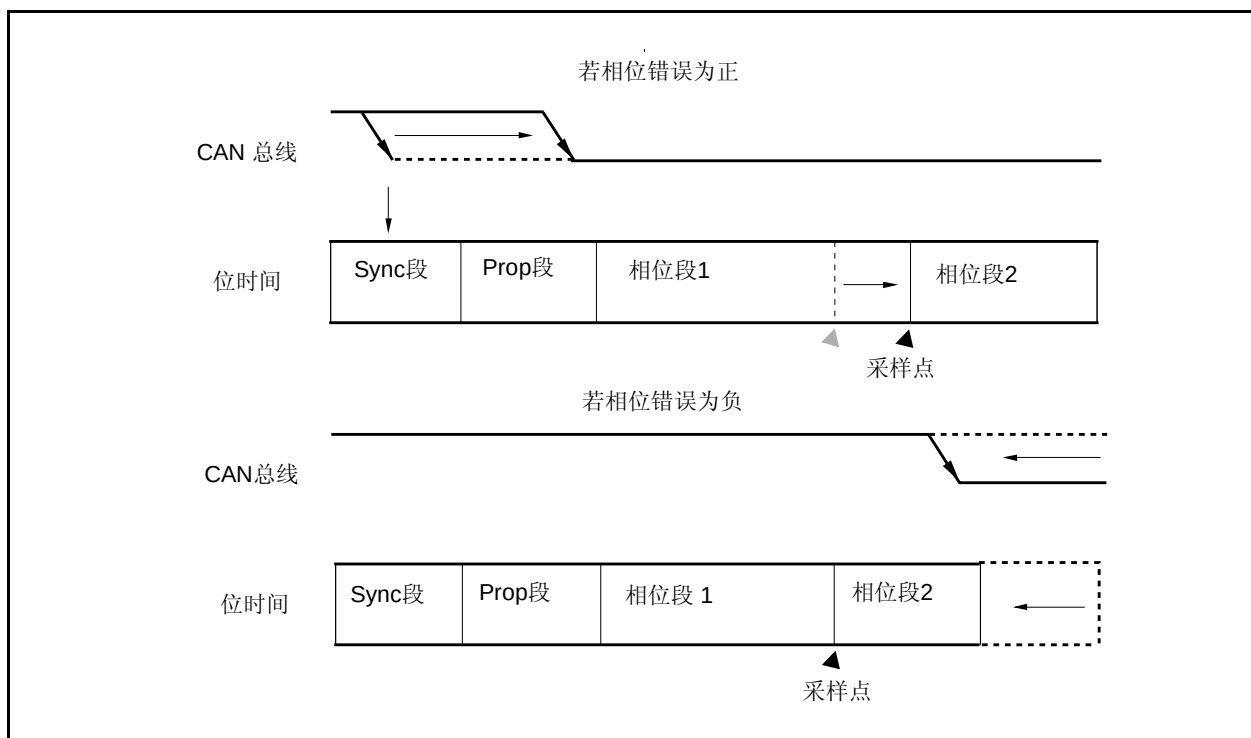


(b) 重新同步

如果接收期间在总线上检测到电平改变则重新建立同步(仅当先前采样了一个隐性电平的情况)。

- 边沿的相位错误由检测边沿和 sync 段的相对位置给出。
<相位错误标记>
0: 如果边沿在 sync 段内
正: 如果边沿在采样点之前(相位错误)
负: 如果边沿在采样点之后(相位错误)
如果相位错误为正: 相位段 1 比指定的 SJW 长。
如果相位错误为负: 相位段 2 比指定的 SJW 段。
• 接收节点的数据采样点随发送节点和接收节点波特率之间的差异而相应移动。

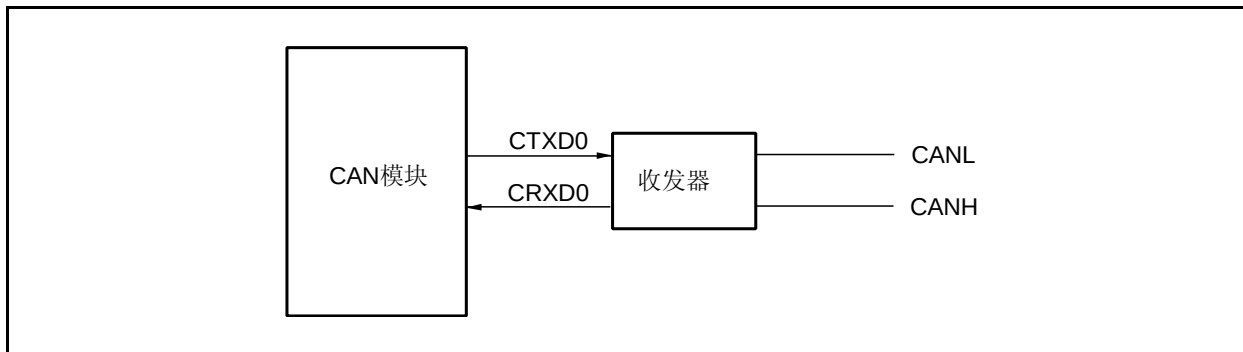
图 20-21. 重新同步



20.4 与目标系统的连接

必须使用外部收发器将 CAN 模块与 CAN 总线进行连接。

图 20-22. 连接至 CAN 总线



20.5 CAN控制器的内部寄存器

20.5.1 CAN控制器配置

表 20-15. 控制器的寄存器列表

项目	寄存器名称
CAN 全局寄存器	CAN0 全局控制寄存器 (C0GMCTRL)
	CAN0 全局时钟寄存器 (C0GMCS)
	CAN0 全局自动块传输控制寄存器 (C0GMABT)
	CAN0 全局自动块传输延迟设置寄存器 (C0GMABTD)
CAN 模块寄存器	CAN0 模块屏蔽 1 寄存器 (C0MASK1L, C0MASK1H)
	CAN0 模块屏蔽 2 寄存器 (C0MASK2L, C0MASK2H)
	CAN0 模块屏蔽 3 寄存器 (C0MASK3L, C0MASK3H)
	CAN0 模块屏蔽 4 寄存器 (C0MASK4L, C0MASK4H)
	CAN0 模块控制寄存器 (C0CTRL)
	CAN0 模块最末错误信息寄存器 (C0LEC)
	CAN0 模块信息寄存器 (C0INFO)
	CAN0 模块错误寄存器 (C0ERC)
	CAN0 模块中断使能寄存器 (C0IE)
	CAN0 模块中断状态寄存器 (C0INTS)
	CAN0 模块位速率预分频器寄存器 (C0BRP)
	CAN0 模块位速率寄存器 (C0BTR)
	CAN0 模块最末输入指针寄存器 (C0LIPT)
	CAN0 模块接收记录列表寄存器 (C0RGPT)
	CAN0 模块最末出指针寄存器 (C0LOPT)
	CAN0 模块发送记录列表寄存器 (C0TGPT)
	CAN0 模块时间标记寄存器 (C0TS)
报文缓存 寄存器	CAN0 报文数据字节 01 寄存器 m (C0MDATA01m)
	CAN0 报文数据字节 0 寄存器 m (C0MDATA0m)
	CAN0 报文数据字节 1 寄存器 m (C0MDATA1m)
	CAN0 报文数据字节 23 寄存器 m (C0MDATA23m)
	CAN0 报文数据字节 2 寄存器 m (C0MDATA2m)
	CAN0 报文数据字节 3 寄存器 m (C0MDATA3m)
	CAN0 报文数据字节 45 寄存器 m (C0MDATA45m)
	CAN0 报文数据字节 4 寄存器 m (C0MDATA4m)
	CAN0 报文数据字节 5 寄存器 m (C0MDATA5m)
	CAN0 报文数据字节 67 寄存器 m (C0MDATA67m)
	CAN0 报文数据字节 6 寄存器 m (C0MDATA6m)
	CAN0 报文数据字节 7 寄存器 m (C0MDATA7m)
	CAN0 报文数据长度寄存器 m (C0MDLCm)
	CAN0 报文结构寄存器 m (C0MCONFm)
	CAN0 报文 ID 寄存器 m (C0MIDLm, C0MIDHm)
	CAN0 报文控制寄存器 m (C0MCTRLm)

- 备注
1. CAN 全局寄存器由 C0GM <寄存器功能>定义。
CAN 模块寄存器由 C0 <寄存器功能>定义。
报文缓存寄存器由 COM <寄存器功能>定义。
 2. m = 00 至 31

20.5.2 寄存器访问类型

表 20-16. 寄存器访问类型(1/17)

地址	寄存器名称	符号	R/W	位操作单元			复位后
				1 位	8 位	16 位	
03FEC000H	CAN0 全局控制寄存器	C0GMCTRL	R/W			√	0000H
03FEC002H	CAN0 全局时钟选择寄存器	C0GMCS			√		0FH
03FEC006H	CAN0 全局块传输寄存器	C0GMABT				√	0000H
03FEC008H	CAN0 全局块传输延时设置寄存器	C0GMABTD			√		00H
03FEC040H	CAN0 模块屏蔽 1 寄存器	C0MASK1L				√	不确定
03FEC042H		C0MASK1H				√	不确定
03FEC044H	CAN0 模块屏蔽 2 寄存器	C0MASK2L				√	不确定
03FEC046H		C0MASK2H				√	不确定
03FEC048H	CAN0 模块屏蔽 3 寄存器	C0MASK3L				√	不确定
03FEC04AH		C0MASK3H				√	不确定
03FEC04CH	CAN0 模块屏蔽 4 寄存器	C0MASK4L				√	不确定
03FEC04EH		C0MASK4H				√	不确定
03FEC050H	CAN0 模块控制寄存器	C0CTRL				√	0000H
03FEC052H	CAN0 模块最末错误代码寄存器	C0LEC			√		00H
03FEC053H	CAN0 模块信息寄存器	C0INFO	R		√		00H
03FEC054H	CAN0 模块错误计数器寄存器	C0ERC				√	0000H
03FEC056H	CAN0 模块中断使能寄存器	C0IE	R/W			√	0000H
03FEC058H	CAN0 模块中断状态寄存器	C0INTS				√	0000H
03FEC05AH	CAN0 模块位-速率预分频器寄存器	C0BRP			√		FFH
03FEC05CH	CAN0 模块位-速率寄存器	C0BTR				√	370FH
03FEC05EH	CAN0 模块最末入指针寄存器	C0LIPT	R		√		不确定
03FEC060H	CAN0 模块接收记录寄存器	C0RGPT	R/W			√	xx02H
03FEC062H	CAN0 模块最末出指针寄存器	C0LOPT	R		√		不确定
03FEC064H	CAN0 模块发送记录列表寄存器	C0TGPT	R/W			√	xx02H
03FEC066H	CAN0 模块时间标记寄存器	C0TS				√	0000H

表 20-16. 寄存器访问类型(2/17)

地址	寄存器名称	符号	R/W	位操作单元			复位后
				1 位	8 位	16 位	
03FEC100H	CAN0 报文数据字节 01 寄存器 00	C0MDATA0100	R/W			√	不确定
03FEC100H	CAN0 报文数据字节 0 寄存器 00	C0MDATA0000			√		不确定
03FEC101H	CAN0 报文数据字节 1 寄存器 00	C0MDATA1000			√		不确定
03FEC102H	CAN0 报文数据字节 23 寄存器 00	C0MDATA2300				√	不确定
03FEC102H	CAN0 报文数据字节 2 寄存器 00	C0MDATA2000			√		不确定
03FEC103H	CAN0 报文数据字节 3 寄存器 00	C0MDATA3000			√		不确定
03FEC104H	CAN0 报文数据字节 45 寄存器 00	C0MDATA4500				√	不确定
03FEC104H	CAN0 报文数据字节 4 寄存器 00	C0MDATA4000			√		不确定
03FEC105H	CAN0 报文数据字节 5 寄存器 00	C0MDATA5000			√		不确定
03FEC106H	CAN0 报文数据字节 67 寄存器 00	C0MDATA6700				√	不确定
03FEC106H	CAN0 报文数据字节 6 寄存器 00	C0MDATA6000			√		不确定
03FEC107H	CAN0 报文数据字节 7 寄存器 00	C0MDATA7000			√		不确定
03FEC108H	CAN0 报文数据长度寄存器 00	C0MDLC00			√		0000xxxxB
03FEC109H	CAN0 报文结构寄存器 00	C0MCONF00			√		不确定
03FEC10AH	CAN0 报文标识符寄存器 00	C0MIDL00				√	不确定
03FEC10CH		C0MIDH00				√	不确定
03FEC10EH	CAN0 报文控制寄存器 00	C0MCTRL00				√	00x00000 000xx000B
03FEC120H	CAN0 报文数据字节 01 寄存器 01	C0MDATA0101				√	不确定
03FEC120H	CAN0 报文数据字节 0 寄存器 01	C0MDATA0001			√		不确定
03FEC121H	CAN0 报文数据字节 1 寄存器 01	C0MDATA1001			√		不确定
03FEC122H	CAN0 报文数据字节 23 寄存器 01	C0MDATA2301				√	不确定
03FEC122H	CAN0 报文数据字节 2 寄存器 01	C0MDATA2001			√		不确定
03FEC123H	CAN0 报文数据字节 3 寄存器 01	C0MDATA3001			√		不确定
03FEC124H	CAN0 报文数据字节 45 寄存器 01	C0MDATA4501				√	不确定
03FEC124H	CAN0 报文数据字节 4 寄存器 01	C0MDATA4001			√		不确定
03FEC125H	CAN0 报文数据字节 5 寄存器 01	C0MDATA5001			√		不确定
03FEC126H	CAN0 报文数据字节 67 寄存器 01	C0MDATA6701				√	不确定
03FEC126H	CAN0 报文数据字节 6 寄存器 01	C0MDATA6001			√		不确定
03FEC127H	CAN0 报文数据字节 7 寄存器 01	C0MDATA7001			√		不确定
03FEC128H	CAN0 报文数据长度寄存器 01	C0MDLC01			√		0000xxxxB
03FEC129H	CAN0 报文结构寄存器 01	C0MCONF01			√		不确定
03FEC12AH	CAN0 报文标识符寄存器 01	C0MIDL01				√	不确定
03FEC12CH		C0MIDH01				√	不确定
03FEC12EH	CAN0 报文控制寄存器 01	C0MCTRL01				√	00x00000 000xx000B

表 20-16. 寄存器访问类型(3/17)

地址	寄存器名称	符号	R/W	位操作单元			复位后
				1 位	8 位	16 位	
03FEC140H	CAN0 报文数据字节 01 寄存器 02	C0MDATA0102	R/W			√	不确定
03FEC140H	CAN0 报文数据字节 0 寄存器 02	C0MDATA002			√		不确定
03FEC141H	CAN0 报文数据字节 1 寄存器 02	C0MDATA102			√		不确定
03FEC142H	CAN0 报文数据字节 23 寄存器 02	C0MDATA2302				√	不确定
03FEC142H	CAN0 报文数据字节 2 寄存器 02	C0MDATA202			√		不确定
03FEC143H	CAN0 报文数据字节 3 寄存器 02	C0MDATA302			√		不确定
03FEC144H	CAN0 报文数据字节 45 寄存器 02	C0MDATA4502				√	不确定
03FEC144H	CAN0 报文数据字节 4 寄存器 02	C0MDATA402			√		不确定
03FEC145H	CAN0 报文数据字节 5 寄存器 02	C0MDATA502			√		不确定
03FEC146H	CAN0 报文数据字节 67 寄存器 02	C0MDATA6702				√	不确定
03FEC146H	CAN0 报文数据字节 6 寄存器 02	C0MDATA602			√		不确定
03FEC147H	CAN0 报文数据字节 7 寄存器 02	C0MDATA702			√		不确定
03FEC148H	CAN0 报文数据长度寄存器 02	C0MDLC02			√		0000xxxxB
03FEC149H	CAN0 报文结构寄存器 02	C0MCONF02			√		不确定
03FEC14AH	CAN0 报文标识符寄存器 02	C0MIDL02				√	不确定
03FEC14CH		C0MIDH02				√	不确定
03FEC14EH	CAN0 报文控制寄存器 02	C0MCTRL02				√	00x00000 000xx000B
03FEC160H	CAN0 报文数据字节 01 寄存器 03	C0MDATA0103				√	不确定
03FEC160H	CAN0 报文数据字节 0 寄存器 03	C0MDATA003			√		不确定
03FEC161H	CAN0 报文数据字节 1 寄存器 03	C0MDATA103			√		不确定
03FEC162H	CAN0 报文数据字节 23 寄存器 03	C0MDATA2303				√	不确定
03FEC162H	CAN0 报文数据字节 2 寄存器 03	C0MDATA203			√		不确定
03FEC163H	CAN0 报文数据字节 3 寄存器 03	C0MDATA303			√		不确定
03FEC164H	CAN0 报文数据字节 45 寄存器 03	C0MDATA4503				√	不确定
03FEC164H	CAN0 报文数据字节 4 寄存器 03	C0MDATA403			√		不确定
03FEC165H	CAN0 报文数据字节 5 寄存器 03	C0MDATA503			√		不确定
03FEC166H	CAN0 报文数据字节 67 寄存器 03	C0MDATA6703				√	不确定
03FEC166H	CAN0 报文数据字节 6 寄存器 03	C0MDATA603			√		不确定
03FEC167H	CAN0 报文数据字节 7 寄存器 03	C0MDATA703			√		不确定
03FEC168H	CAN0 报文数据长度寄存器 03	C0MDLC03			√		0000xxxxB
03FEC169H	CAN0 报文结构寄存器 03	C0MCONF03			√		不确定
03FEC16AH	CAN0 报文标识符寄存器 03	C0MIDL03				√	不确定
03FEC16CH		C0MIDH03				√	不确定
03FEC16EH	CAN0 报文控制寄存器 03	C0MCTRL03				√	00x00000 000xx000B

表 20-16. 寄存器访问类型(4/17)

地址	寄存器名称	符号	R/W	位操作单元			复位后
				1 位	8 位	16 位	
03FEC180H	CAN0 报文数据字节 01 寄存器 04	C0MDATA0104	R/W			√	不确定
03FEC180H	CAN0 报文数据字节 0 寄存器 04	C0MDATA004			√		不确定
03FEC181H	CAN0 报文数据字节 1 寄存器 04	C0MDATA104			√		不确定
03FEC182H	CAN0 报文数据字节 23 寄存器 04	C0MDATA2304				√	不确定
03FEC182H	CAN0 报文数据字节 2 寄存器 04	C0MDATA204			√		不确定
03FEC183H	CAN0 报文数据字节 3 寄存器 04	C0MDATA304			√		不确定
03FEC184H	CAN0 报文数据字节 45 寄存器 04	C0MDATA4504				√	不确定
03FEC184H	CAN0 报文数据字节 4 寄存器 04	C0MDATA404			√		不确定
03FEC185H	CAN0 报文数据字节 5 寄存器 04	C0MDATA504			√		不确定
03FEC186H	CAN0 报文数据字节 67 寄存器 04	C0MDATA6704				√	不确定
03FEC186H	CAN0 报文数据字节 6 寄存器 04	C0MDATA604			√		不确定
03FEC187H	CAN0 报文数据字节 7 寄存器 04	C0MDATA704			√		不确定
03FEC188H	CAN0 报文数据长度寄存器 04	C0MDLC04			√		0000xxxxB
03FEC189H	CAN0 报文结构寄存器 04	C0MCONF04			√		不确定
03FEC18AH	CAN0 报文标识符寄存器 04	C0MIDL04				√	不确定
03FEC18CH		C0MIDH04				√	不确定
03FEC18EH	CAN0 报文控制寄存器 04	C0MCTRL04				√	00x00000 000xx000B
03FEC1A0H	CAN0 报文数据字节 01 寄存器 05	C0MDATA0105				√	不确定
03FEC1A0H	CAN0 报文数据字节 0 寄存器 05	C0MDATA005			√		不确定
03FEC1A1H	CAN0 报文数据字节 1 寄存器 05	C0MDATA105			√		不确定
03FEC1A2H	CAN0 报文数据字节 23 寄存器 05	C0MDATA2305				√	不确定
03FEC1A2H	CAN0 报文数据字节 2 寄存器 05	C0MDATA205			√		不确定
03FEC1A3H	CAN0 报文数据字节 3 寄存器 05	C0MDATA305			√		不确定
03FEC1A4H	CAN0 报文数据字节 45 寄存器 05	C0MDATA4505				√	不确定
03FEC1A4H	CAN0 报文数据字节 4 寄存器 05	C0MDATA405			√		不确定
03FEC1A5H	CAN0 报文数据字节 5 寄存器 05	C0MDATA505			√		不确定
03FEC1A6H	CAN0 报文数据字节 67 寄存器 05	C0MDATA6705				√	不确定
03FEC1A6H	CAN0 报文数据字节 6 寄存器 05	C0MDATA605			√		不确定
03FEC1A7H	CAN0 报文数据字节 7 寄存器 05	C0MDATA705			√		不确定
03FEC1A8H	CAN0 报文数据长度寄存器 05	C0MDLC05			√		0000xxxxB
03FEC1A9H	CAN0 报文结构寄存器 05	C0MCONF05			√		不确定
03FEC1AAH	CAN0 报文标识符寄存器 05	C0MIDL05				√	不确定
03FEC1ACH		C0MIDH05				√	不确定
03FEC1AEH	CAN0 报文控制寄存器 05	C0MCTRL05				√	00x00000 000xx000B

表 20-16. 寄存器访问类型(5/17)

地址	寄存器名称	符号	R/W	位操作单元			复位后
				1 位	8 位	16 位	
03FEC1C0H	CAN0 报文数据字节 01 寄存器 06	C0MDATA0106	R/W			√	不确定
03FEC1C0H	CAN0 报文数据字节 0 寄存器 06	C0MDATA006			√		不确定
03FEC1C1H	CAN0 报文数据字节 1 寄存器 06	C0MDATA106			√		不确定
03FEC1C2H	CAN0 报文数据字节 23 寄存器 06	C0MDATA2306				√	不确定
03FEC1C2H	CAN0 报文数据字节 2 寄存器 06	C0MDATA206			√		不确定
03FEC1C3H	CAN0 报文数据字节 3 寄存器 06	C0MDATA306			√		不确定
03FEC1C4H	CAN0 报文数据字节 45 寄存器 06	C0MDATA4506				√	不确定
03FEC1C4H	CAN0 报文数据字节 4 寄存器 06	C0MDATA406			√		不确定
03FEC1C5H	CAN0 报文数据字节 5 寄存器 06	C0MDATA506			√		不确定
03FEC1C6H	CAN0 报文数据字节 67 寄存器 06	C0MDATA6706				√	不确定
03FEC1C6H	CAN0 报文数据字节 6 寄存器 06	C0MDATA606			√		不确定
03FEC1C7H	CAN0 报文数据字节 7 寄存器 06	C0MDATA706			√		不确定
03FEC1C8H	CAN0 报文数据长度寄存器 06	C0MDLC06			√		0000xxxxB
03FEC1C9H	CAN0 报文结构寄存器 06	C0MCONF06			√		不确定
03FEC1CAH	CAN0 报文标识符寄存器 06	C0MIDL06				√	不确定
03FEC1CCH		C0MIDH06				√	不确定
03FEC1CEH	CAN0 报文控制寄存器 06	C0MCTRL06				√	00x00000 000xx000B
03FEC1E0H	CAN0 报文数据字节 01 寄存器 07	C0MDATA0107				√	不确定
03FEC1E0H	CAN0 报文数据字节 0 寄存器 07	C0MDATA007			√		不确定
03FEC1E1H	CAN0 报文数据字节 1 寄存器 07	C0MDATA107			√		不确定
03FEC1E2H	CAN0 报文数据字节 23 寄存器 07	C0MDATA2307				√	不确定
03FEC1E2H	CAN0 报文数据字节 2 寄存器 07	C0MDATA207			√		不确定
03FEC1E3H	CAN0 报文数据字节 3 寄存器 07	C0MDATA307			√		不确定
03FEC1E4H	CAN0 报文数据字节 45 寄存器 07	C0MDATA4507				√	不确定
03FEC1E4H	CAN0 报文数据字节 4 寄存器 07	C0MDATA407			√		不确定
03FEC1E5H	CAN0 报文数据字节 5 寄存器 07	C0MDATA507			√		不确定
03FEC1E6H	CAN0 报文数据字节 67 寄存器 07	C0MDATA6707				√	不确定
03FEC1E6H	CAN0 报文数据字节 6 寄存器 07	C0MDATA607			√		不确定
03FEC1E7H	CAN0 报文数据字节 7 寄存器 07	C0MDATA707			√		不确定
03FEC1E8H	CAN0 报文数据长度寄存器 07	C0MDLC07			√		0000xxxxB
03FEC1E9H	CAN0 报文结构寄存器 07	C0MCONF07			√		不确定
03FEC1EAH	CAN0 报文标识符寄存器 07	C0MIDL07				√	不确定
03FEC1ECH		C0MIDH07				√	不确定
03FEC1EEH	CAN0 报文控制寄存器 07	C0MCTRL07				√	00x00000 000xx000B

表 20-16. 寄存器访问类型(6/17)

地址	寄存器名称	符号	R/W	位操作单元			复位后
				1 位	8 位	16 位	
03FEC200H	CAN0 报文数据字节 01 寄存器 08	C0MDATA0108	R/W			√	不确定
03FEC200H	CAN0 报文数据字节 0 寄存器 08	C0MDATA008			√		不确定
03FEC201H	CAN0 报文数据字节 1 寄存器 08	C0MDATA108			√		不确定
03FEC202H	CAN0 报文数据字节 23 寄存器 08	C0MDATA2308				√	不确定
03FEC202H	CAN0 报文数据字节 2 寄存器 08	C0MDATA208			√		不确定
03FEC203H	CAN0 报文数据字节 3 寄存器 08	C0MDATA308			√		不确定
03FEC204H	CAN0 报文数据字节 45 寄存器 08	C0MDATA4508				√	不确定
03FEC204H	CAN0 报文数据字节 4 寄存器 08	C0MDATA408			√		不确定
03FEC205H	CAN0 报文数据字节 5 寄存器 08	C0MDATA508			√		不确定
03FEC206H	CAN0 报文数据字节 67 寄存器 08	C0MDATA6708				√	不确定
03FEC206H	CAN0 报文数据字节 6 寄存器 08	C0MDATA608			√		不确定
03FEC207H	CAN0 报文数据字节 7 寄存器 08	C0MDATA708			√		不确定
03FEC208H	CAN0 报文数据长度寄存器 08	C0MDLC08			√		0000xxxxB
03FEC209H	CAN0 报文结构寄存器 08	C0MCONF08			√		不确定
03FEC20AH	CAN0 报文标识符寄存器 08	C0MIDL08				√	不确定
03FEC20CH		C0MIDH08				√	不确定
03FEC20EH	CAN0 报文控制寄存器 08	C0MCTRL08				√	00x00000 000xx000B
03FEC220H	CAN0 报文数据字节 01 寄存器 09	C0MDATA0109				√	不确定
03FEC220H	CAN0 报文数据字节 0 寄存器 09	C0MDATA009			√		不确定
03FEC221H	CAN0 报文数据字节 1 寄存器 09	C0MDATA109			√		不确定
03FEC222H	CAN0 报文数据字节 23 寄存器 09	C0MDATA2309				√	不确定
03FEC222H	CAN0 报文数据字节 2 寄存器 09	C0MDATA209			√		不确定
03FEC223H	CAN0 报文数据字节 3 寄存器 09	C0MDATA309			√		不确定
03FEC224H	CAN0 报文数据字节 45 寄存器 09	C0MDATA4509				√	不确定
03FEC224H	CAN0 报文数据字节 4 寄存器 09	C0MDATA409			√		不确定
03FEC225H	CAN0 报文数据字节 5 寄存器 09	C0MDATA509			√		不确定
03FEC226H	CAN0 报文数据字节 67 寄存器 09	C0MDATA6709				√	不确定
03FEC226H	CAN0 报文数据字节 6 寄存器 09	C0MDATA609			√		不确定
03FEC227H	CAN0 报文数据字节 7 寄存器 09	C0MDATA709			√		不确定
03FEC228H	CAN0 报文数据长度寄存器 09	C0MDLC09			√		0000xxxxB
03FEC229H	CAN0 报文结构寄存器 09	C0MCONF09			√		不确定
03FEC22AH	CAN0 报文标识符寄存器 09	C0MIDL09				√	不确定
03FEC22CH		C0MIDH09				√	不确定
03FEC22EH	CAN0 报文控制寄存器 09	C0MCTRL09				√	00x00000 000xx000B

表 20-16. 寄存器访问类型(7/17)

地址	寄存器名称	符号	R/W	位操作单元			复位后
				1 位	8 位	16 位	
03FEC240H	CAN0 报文数据字节 01 寄存器 10	C0MDATA0110	R/W			√	不确定
03FEC240H	CAN0 报文数据字节 0 寄存器 10	C0MDATA010			√		不确定
03FEC241H	CAN0 报文数据字节 1 寄存器 10	C0MDATA110			√		不确定
03FEC242H	CAN0 报文数据字节 23 寄存器 10	C0MDATA2310				√	不确定
03FEC242H	CAN0 报文数据字节 2 寄存器 10	C0MDATA210			√		不确定
03FEC243H	CAN0 报文数据字节 3 寄存器 10	C0MDATA310			√		不确定
03FEC244H	CAN0 报文数据字节 45 寄存器 10	C0MDATA4510				√	不确定
03FEC244H	CAN0 报文数据字节 4 寄存器 10	C0MDATA410			√		不确定
03FEC245H	CAN0 报文数据字节 5 寄存器 10	C0MDATA510			√		不确定
03FEC246H	CAN0 报文数据字节 67 寄存器 10	C0MDATA6710				√	不确定
03FEC246H	CAN0 报文数据字节 6 寄存器 10	C0MDATA610			√		不确定
03FEC247H	CAN0 报文数据字节 7 寄存器 10	C0MDATA710			√		不确定
03FEC248H	CAN0 报文数据长度寄存器 10	C0MDLC10			√		0000xxxxB
03FEC249H	CAN0 报文结构寄存器 10	C0MCONF10			√		不确定
03FEC24AH	CAN0 报文标识符寄存器 10	C0MIDL10				√	不确定
03FEC24CH		C0MIDH10				√	不确定
03FEC24EH	CAN0 报文控制寄存器 10	C0MCTRL10				√	00x00000 000xx000B
03FEC260H	CAN0 报文数据字节 01 寄存器 11	C0MDATA0111				√	不确定
03FEC260H	CAN0 报文数据字节 0 寄存器 11	C0MDATA011			√		不确定
03FEC261H	CAN0 报文数据字节 1 寄存器 11	C0MDATA111			√		不确定
03FEC262H	CAN0 报文数据字节 23 寄存器 11	C0MDATA2311				√	不确定
03FEC262H	CAN0 报文数据字节 2 寄存器 11	C0MDATA211			√		不确定
03FEC263H	CAN0 报文数据字节 3 寄存器 11	C0MDATA311			√		不确定
03FEC264H	CAN0 报文数据字节 45 寄存器 11	C0MDATA4511				√	不确定
03FEC264H	CAN0 报文数据字节 4 寄存器 11	C0MDATA411			√		不确定
03FEC265H	CAN0 报文数据字节 5 寄存器 11	C0MDATA511			√		不确定
03FEC266H	CAN0 报文数据字节 67 寄存器 11	C0MDATA6711				√	不确定
03FEC266H	CAN0 报文数据字节 6 寄存器 11	C0MDATA611			√		不确定
03FEC267H	CAN0 报文数据字节 7 寄存器 11	C0MDATA711			√		不确定
03FEC268H	CAN0 报文数据长度寄存器 11	C0MDLC11			√		0000xxxxB
03FEC269H	CAN0 报文结构寄存器 11	C0MCONF11			√		不确定
03FEC26AH	CAN0 报文标识符寄存器 11	C0MIDL11				√	不确定
03FEC26CH		C0MIDH11				√	不确定
03FEC26EH	CAN0 报文控制寄存器 11	C0MCTRL11				√	00x00000 000xx000B

表 20-16. 寄存器访问类型(8/17)

地址	寄存器名称	符号	R/W	位操作单元			复位后
				1 位	8 位	16 位	
03FEC280H	CAN0 报文数据字节 01 寄存器 12	C0MDATA0112	R/W			√	不确定
03FEC280H	CAN0 报文数据字节 0 寄存器 12	C0MDATA012			√		不确定
03FEC281H	CAN0 报文数据字节 1 寄存器 12	C0MDATA112			√		不确定
03FEC282H	CAN0 报文数据字节 23 寄存器 12	C0MDATA2312				√	不确定
03FEC282H	CAN0 报文数据字节 2 寄存器 12	C0MDATA212			√		不确定
03FEC283H	CAN0 报文数据字节 3 寄存器 12	C0MDATA312			√		不确定
03FEC284H	CAN0 报文数据字节 45 寄存器 12	C0MDATA4512				√	不确定
03FEC284H	CAN0 报文数据字节 4 寄存器 12	C0MDATA412			√		不确定
03FEC285H	CAN0 报文数据字节 5 寄存器 12	C0MDATA512			√		不确定
03FEC286H	CAN0 报文数据字节 67 寄存器 12	C0MDATA6712				√	不确定
03FEC286H	CAN0 报文数据字节 6 寄存器 12	C0MDATA612			√		不确定
03FEC287H	CAN0 报文数据字节 7 寄存器 12	C0MDATA712			√		不确定
03FEC288H	CAN0 报文数据长度寄存器 12	C0MDLC12			√		0000xxxxB
03FEC289H	CAN0 报文结构寄存器 12	C0MCONF12			√		不确定
03FEC28AH	CAN0 报文标识符寄存器 12	C0MIDL12				√	不确定
03FEC28CH		C0MIDH12				√	不确定
03FEC28EH	CAN0 报文控制寄存器 12	C0MCTRL12				√	00x00000 000xx000B
03FEC2A0H	CAN0 报文数据字节 01 寄存器 13	C0MDATA0113				√	不确定
03FEC2A0H	CAN0 报文数据字节 0 寄存器 13	C0MDATA013			√		不确定
03FEC2A1H	CAN0 报文数据字节 1 寄存器 13	C0MDATA113			√		不确定
03FEC2A2H	CAN0 报文数据字节 23 寄存器 13	C0MDATA2313				√	不确定
03FEC2A2H	CAN0 报文数据字节 2 寄存器 13	C0MDATA213			√		不确定
03FEC2A3H	CAN0 报文数据字节 3 寄存器 13	C0MDATA313			√		不确定
03FEC2A4H	CAN0 报文数据字节 45 寄存器 13	C0MDATA4513				√	不确定
03FEC2A4H	CAN0 报文数据字节 4 寄存器 13	C0MDATA413			√		不确定
03FEC2A5H	CAN0 报文数据字节 5 寄存器 13	C0MDATA513			√		不确定
03FEC2A6H	CAN0 报文数据字节 67 寄存器 13	C0MDATA6713				√	不确定
03FEC2A6H	CAN0 报文数据字节 6 寄存器 13	C0MDATA613			√		不确定
03FEC2A7H	CAN0 报文数据字节 7 寄存器 13	C0MDATA713			√		不确定
03FEC2A8H	CAN0 报文数据长度寄存器 13	C0MDLC13			√		0000xxxxB
03FEC2A9H	CAN0 报文结构寄存器 13	C0MCONF13			√		不确定
03FEC2AAH	CAN0 报文标识符寄存器 13	C0MIDL13				√	不确定
03FEC2ACH		C0MIDH13				√	不确定
03FEC2AEH	CAN0 报文控制寄存器 13	C0MCTRL13				√	00x00000 000xx000B

表 20-16. 寄存器访问类型(9/17)

地址	寄存器名称	符号	R/W	位操作单元			复位后
				1 位	8 位	16 位	
03FEC2C0H	CAN0 报文数据字节 01 寄存器 14	C0MDATA0114	R/W			√	不确定
03FEC2C0H	CAN0 报文数据字节 0 寄存器 14	C0MDATA014			√		不确定
03FEC2C1H	CAN0 报文数据字节 1 寄存器 14	C0MDATA114			√		不确定
03FEC2C2H	CAN0 报文数据字节 23 寄存器 14	C0MDATA2314				√	不确定
03FEC2C2H	CAN0 报文数据字节 2 寄存器 14	C0MDATA214			√		不确定
03FEC2C3H	CAN0 报文数据字节 3 寄存器 14	C0MDATA314			√		不确定
03FEC2C4H	CAN0 报文数据字节 45 寄存器 14	C0MDATA4514				√	不确定
03FEC2C4H	CAN0 报文数据字节 4 寄存器 14	C0MDATA414			√		不确定
03FEC2C5H	CAN0 报文数据字节 5 寄存器 14	C0MDATA514			√		不确定
03FEC2C6H	CAN0 报文数据字节 67 寄存器 14	C0MDATA6714				√	不确定
03FEC2C6H	CAN0 报文数据字节 6 寄存器 14	C0MDATA614			√		不确定
03FEC2C7H	CAN0 报文数据字节 7 寄存器 14	C0MDATA714			√		不确定
03FEC2C8H	CAN0 报文数据长度寄存器 14	C0MDLC14			√		0000xxxxB
03FEC2C9H	CAN0 报文结构寄存器 14	C0MCONF14			√		不确定
03FEC2CAH	CAN0 报文标识符寄存器 14	C0MIDL14				√	不确定
03FEC2CCH		C0MIDH14				√	不确定
03FEC2CEH	CAN0 报文控制寄存器 14	C0MCTRL14				√	00x00000 000xx000B
03FEC2E0H	CAN0 报文数据字节 01 寄存器 15	C0MDATA0115				√	不确定
03FEC2E0H	CAN0 报文数据字节 0 寄存器 15	C0MDATA015			√		不确定
03FEC2E1H	CAN0 报文数据字节 1 寄存器 15	C0MDATA115			√		不确定
03FEC2E2H	CAN0 报文数据字节 23 寄存器 15	C0MDATA2315				√	不确定
03FEC2E2H	CAN0 报文数据字节 2 寄存器 15	C0MDATA215			√		不确定
03FEC2E3H	CAN0 报文数据字节 3 寄存器 15	C0MDATA315			√		不确定
03FEC2E4H	CAN0 报文数据字节 45 寄存器 15	C0MDATA4515				√	不确定
03FEC2E4H	CAN0 报文数据字节 4 寄存器 15	C0MDATA415			√		不确定
03FEC2E5H	CAN0 报文数据字节 5 寄存器 15	C0MDATA515			√		不确定
03FEC2E6H	CAN0 报文数据字节 67 寄存器 15	C0MDATA6715				√	不确定
03FEC2E6H	CAN0 报文数据字节 6 寄存器 15	C0MDATA615			√		不确定
03FEC2E7H	CAN0 报文数据字节 7 寄存器 15	C0MDATA715			√		不确定
03FEC2E8H	CAN0 报文数据长度寄存器 15	C0MDLC15			√		0000xxxxB
03FEC2E9H	CAN0 报文结构寄存器 15	C0MCONF15			√		不确定
03FEC2EAH	CAN0 报文标识符寄存器 15	C0MIDL15				√	不确定
03FEC2ECH		C0MIDH15				√	不确定
03FEC2EEH	CAN0 报文控制寄存器 15	C0MCTRL15				√	00x00000 000xx000B

表 20-16. 寄存器访问类型(10/17)

地址	寄存器名称	符号	R/W	位操作单元			复位后
				1 位	8 位	16 位	
03FEC300H	CAN0 报文数据字节 01 寄存器 16	C0MDATA0116	R/W			√	不确定
03FEC300H	CAN0 报文数据字节 0 寄存器 16	C0MDATA016			√		不确定
03FEC301H	CAN0 报文数据字节 1 寄存器 16	C0MDATA116			√		不确定
03FEC302H	CAN0 报文数据字节 23 寄存器 16	C0MDATA2316				√	不确定
03FEC302H	CAN0 报文数据字节 2 寄存器 16	C0MDATA216			√		不确定
03FEC303H	CAN0 报文数据字节 3 寄存器 16	C0MDATA316			√		不确定
03FEC304H	CAN0 报文数据字节 45 寄存器 16	C0MDATA4516				√	不确定
03FEC304H	CAN0 报文数据字节 4 寄存器 16	C0MDATA416			√		不确定
03FEC305H	CAN0 报文数据字节 5 寄存器 16	C0MDATA516			√		不确定
03FEC306H	CAN0 报文数据字节 67 寄存器 16	C0MDATA6716				√	不确定
03FEC306H	CAN0 报文数据字节 6 寄存器 16	C0MDATA616			√		不确定
03FEC307H	CAN0 报文数据字节 7 寄存器 16	C0MDATA716			√		不确定
03FEC308H	CAN0 报文数据长度寄存器 16	C0MDLC16			√		0000xxxxB
03FEC309H	CAN0 报文结构寄存器 16	C0MCONF16			√		不确定
03FEC30AH	CAN0 报文标识符寄存器 16	C0MIDL16				√	不确定
03FEC30CH		C0MIDH16				√	不确定
03FEC30EH	CAN0 报文控制寄存器 16	C0MCTRL16				√	00x00000 000xx000B
03FEC320H	CAN0 报文数据字节 01 寄存器 17	C0MDATA0117				√	不确定
03FEC320H	CAN0 报文数据字节 0 寄存器 17	C0MDATA017			√		不确定
03FEC321H	CAN0 报文数据字节 1 寄存器 17	C0MDATA117			√		不确定
03FEC322H	CAN0 报文数据字节 23 寄存器 17	C0MDATA2317				√	不确定
03FEC322H	CAN0 报文数据字节 2 寄存器 17	C0MDATA217			√		不确定
03FEC323H	CAN0 报文数据字节 3 寄存器 17	C0MDATA317			√		不确定
03FEC324H	CAN0 报文数据字节 45 寄存器 17	C0MDATA4517				√	不确定
03FEC324H	CAN0 报文数据字节 4 寄存器 17	C0MDATA417			√		不确定
03FEC325H	CAN0 报文数据字节 5 寄存器 17	C0MDATA517			√		不确定
03FEC326H	CAN0 报文数据字节 67 寄存器 17	C0MDATA6717				√	不确定
03FEC326H	CAN0 报文数据字节 6 寄存器 17	C0MDATA617			√		不确定
03FEC327H	CAN0 报文数据字节 7 寄存器 17	C0MDATA717			√		不确定
03FEC328H	CAN0 报文数据长度寄存器 17	C0MDLC17			√		0000xxxxB
03FEC329H	CAN0 报文结构寄存器 17	C0MCONF17			√		不确定
03FEC32AH	CAN0 报文标识符寄存器 17	C0MIDL17				√	不确定
03FEC32CH		C0MIDH17				√	不确定
03FEC32EH	CAN0 报文控制寄存器 17	C0MCTRL17				√	00x00000 000xx000B

表 20-16. 寄存器访问类型(11/17)

地址	寄存器名称	符号	R/W	位操作单元			复位后
				1 位	8 位	16 位	
03FEC340H	CAN0 报文数据字节 01 寄存器 18	C0MDATA0118	R/W			√	不确定
03FEC340H	CAN0 报文数据字节 0 寄存器 18	C0MDATA018			√		不确定
03FEC341H	CAN0 报文数据字节 1 寄存器 18	C0MDATA118			√		不确定
03FEC342H	CAN0 报文数据字节 23 寄存器 18	C0MDATA2318				√	不确定
03FEC342H	CAN0 报文数据字节 2 寄存器 18	C0MDATA218			√		不确定
03FEC343H	CAN0 报文数据字节 3 寄存器 18	C0MDATA318			√		不确定
03FEC344H	CAN0 报文数据字节 45 寄存器 18	C0MDATA4518				√	不确定
03FEC344H	CAN0 报文数据字节 4 寄存器 18	C0MDATA418			√		不确定
03FEC345H	CAN0 报文数据字节 5 寄存器 18	C0MDATA518			√		不确定
03FEC346H	CAN0 报文数据字节 67 寄存器 18	C0MDATA6718				√	不确定
03FEC346H	CAN0 报文数据字节 6 寄存器 18	C0MDATA618			√		不确定
03FEC347H	CAN0 报文数据字节 7 寄存器 18	C0MDATA718			√		不确定
03FEC348H	CAN0 报文数据长度寄存器 18	C0MDLC18			√		0000xxxxB
03FEC349H	CAN0 报文结构寄存器 18	C0MCONF18			√		不确定
03FEC34AH	CAN0 报文标识符寄存器 18	C0MIDL18				√	不确定
03FEC34CH		C0MIDH18				√	不确定
03FEC34EH	CAN0 报文控制寄存器 18	C0MCTRL18				√	00x00000 000xx000B
03FEC360H	CAN0 报文数据字节 01 寄存器 19	C0MDATA0119				√	不确定
03FEC360H	CAN0 报文数据字节 0 寄存器 19	C0MDATA019			√		不确定
03FEC361H	CAN0 报文数据字节 1 寄存器 19	C0MDATA119			√		不确定
03FEC362H	CAN0 报文数据字节 23 寄存器 19	C0MDATA2319				√	不确定
03FEC362H	CAN0 报文数据字节 2 寄存器 19	C0MDATA219			√		不确定
03FEC363H	CAN0 报文数据字节 3 寄存器 19	C0MDATA319			√		不确定
03FEC364H	CAN0 报文数据字节 45 寄存器 19	C0MDATA4519				√	不确定
03FEC364H	CAN0 报文数据字节 4 寄存器 19	C0MDATA419			√		不确定
03FEC365H	CAN0 报文数据字节 5 寄存器 19	C0MDATA519			√		不确定
03FEC366H	CAN0 报文数据字节 67 寄存器 19	C0MDATA6719				√	不确定
03FEC366H	CAN0 报文数据字节 6 寄存器 19	C0MDATA619			√		不确定
03FEC367H	CAN0 报文数据字节 7 寄存器 19	C0MDATA719			√		不确定
03FEC368H	CAN0 报文数据长度寄存器 19	C0MDLC19			√		0000xxxxB
03FEC369H	CAN0 报文结构寄存器 19	C0MCONF19			√		不确定
03FEC36AH	CAN0 报文标识符寄存器 19	C0MIDL19				√	不确定
03FEC36CH		C0MIDH19				√	不确定
03FEC36EH	CAN0 报文控制寄存器 19	C0MCTRL19				√	00x00000 000xx000B

表 20-16. 寄存器访问类型(12/17)

地址	寄存器名称	符号	R/W	位操作单元			复位后
				1 位	8 位	16 位	
03FEC380H	CAN0 报文数据字节 01 寄存器 20	C0MDATA0120	R/W			√	不确定
03FEC380H	CAN0 报文数据字节 0 寄存器 20	C0MDATA020			√		不确定
03FEC381H	CAN0 报文数据字节 1 寄存器 20	C0MDATA120			√		不确定
03FEC382H	CAN0 报文数据字节 23 寄存器 20	C0MDATA2320				√	不确定
03FEC382H	CAN0 报文数据字节 2 寄存器 20	C0MDATA220			√		不确定
03FEC383H	CAN0 报文数据字节 3 寄存器 20	C0MDATA320			√		不确定
03FEC384H	CAN0 报文数据字节 45 寄存器 20	C0MDATA4520				√	不确定
03FEC384H	CAN0 报文数据字节 4 寄存器 20	C0MDATA420			√		不确定
03FEC385H	CAN0 报文数据字节 5 寄存器 20	C0MDATA520			√		不确定
03FEC386H	CAN0 报文数据字节 67 寄存器 20	C0MDATA6720				√	不确定
03FEC386H	CAN0 报文数据字节 6 寄存器 20	C0MDATA620			√		不确定
03FEC387H	CAN0 报文数据字节 7 寄存器 20	C0MDATA720			√		不确定
03FEC388H	CAN0 报文数据长度寄存器 20	C0MDLC20			√		0000xxxxB
03FEC389H	CAN0 报文结构寄存器 20	C0MCONF20			√		不确定
03FEC38AH	CAN0 报文标识符寄存器 20	C0MIDL20				√	不确定
03FEC38CH		C0MIDH20				√	不确定
03FEC38EH	CAN0 报文控制寄存器 20	C0MCTRL20				√	00x00000 000xx000B
03FEC3A0H	CAN0 报文数据字节 01 寄存器 21	C0MDATA0121				√	不确定
03FEC3A0H	CAN0 报文数据字节 0 寄存器 21	C0MDATA021			√		不确定
03FEC3A1H	CAN0 报文数据字节 1 寄存器 21	C0MDATA121			√		不确定
03FEC3A2H	CAN0 报文数据字节 23 寄存器 21	C0MDATA2321				√	不确定
03FEC3A2H	CAN0 报文数据字节 2 寄存器 21	C0MDATA221			√		不确定
03FEC3A3H	CAN0 报文数据字节 3 寄存器 21	C0MDATA321			√		不确定
03FEC3A4H	CAN0 报文数据字节 45 寄存器 21	C0MDATA4521				√	不确定
03FEC3A4H	CAN0 报文数据字节 4 寄存器 21	C0MDATA421			√		不确定
03FEC3A5H	CAN0 报文数据字节 5 寄存器 21	C0MDATA521			√		不确定
03FEC3A6H	CAN0 报文数据字节 67 寄存器 21	C0MDATA6721				√	不确定
03FEC3A6H	CAN0 报文数据字节 6 寄存器 21	C0MDATA621			√		不确定
03FEC3A7H	CAN0 报文数据字节 7 寄存器 21	C0MDATA721			√		不确定
03FEC3A8H	CAN0 报文数据长度寄存器 21	C0MDLC21			√		0000xxxxB
03FEC3A9H	CAN0 报文结构寄存器 21	C0MCONF21			√		不确定
03FEC3AAH	CAN0 报文标识符寄存器 21	C0MIDL21				√	不确定
03FEC3ACH		C0MIDH21				√	不确定
03FEC3AEH	CAN0 报文控制寄存器 21	C0MCTRL21				√	00x00000 000xx000B

表 20-16. 寄存器访问类型(13/17)

地址	寄存器名称	符号	R/W	位操作单元			复位后
				1 位	8 位	16 位	
03FEC3C0H	CAN0 报文数据字节 01 寄存器 22	C0MDATA0122	R/W			√	不确定
03FEC3C0H	CAN0 报文数据字节 0 寄存器 22	C0MDATA022			√		不确定
03FEC3C1H	CAN0 报文数据字节 1 寄存器 22	C0MDATA122			√		不确定
03FEC3C2H	CAN0 报文数据字节 23 寄存器 22	C0MDATA2322				√	不确定
03FEC3C2H	CAN0 报文数据字节 2 寄存器 22	C0MDATA222			√		不确定
03FEC3C3H	CAN0 报文数据字节 3 寄存器 22	C0MDATA322			√		不确定
03FEC3C4H	CAN0 报文数据字节 45 寄存器 22	C0MDATA4522				√	不确定
03FEC3C4H	CAN0 报文数据字节 4 寄存器 22	C0MDATA422			√		不确定
03FEC3C5H	CAN0 报文数据字节 5 寄存器 22	C0MDATA522			√		不确定
03FEC3C6H	CAN0 报文数据字节 67 寄存器 22	C0MDATA6722				√	不确定
03FEC3C6H	CAN0 报文数据字节 6 寄存器 22	C0MDATA622			√		不确定
03FEC3C7H	CAN0 报文数据字节 7 寄存器 22	C0MDATA722			√		不确定
03FEC3C8H	CAN0 报文数据长度寄存器 22	C0MDLC22			√		0000xxxxB
03FEC3C9H	CAN0 报文结构寄存器 22	C0MCONF22			√		不确定
03FEC3CAH	CAN0 报文标识符寄存器 22	C0MIDL22				√	不确定
03FEC3CCH		C0MIDH22				√	不确定
03FEC3CEH	CAN0 报文控制寄存器 22	C0MCTRL22				√	00x00000 000xx000B
03FEC3E0H	CAN0 报文数据字节 01 寄存器 23	C0MDATA0123				√	不确定
03FEC3E0H	CAN0 报文数据字节 0 寄存器 23	C0MDATA023			√		不确定
03FEC3E1H	CAN0 报文数据字节 1 寄存器 23	C0MDATA123			√		不确定
03FEC3E2H	CAN0 报文数据字节 23 寄存器 23	C0MDATA2323				√	不确定
03FEC3E2H	CAN0 报文数据字节 2 寄存器 23	C0MDATA223			√		不确定
03FEC3E3H	CAN0 报文数据字节 3 寄存器 23	C0MDATA323			√		不确定
03FEC3E4H	CAN0 报文数据字节 45 寄存器 23	C0MDATA4523				√	不确定
03FEC3E4H	CAN0 报文数据字节 4 寄存器 23	C0MDATA423			√		不确定
03FEC3E5H	CAN0 报文数据字节 5 寄存器 23	C0MDATA523			√		不确定
03FEC3E6H	CAN0 报文数据字节 67 寄存器 23	C0MDATA6723				√	不确定
03FEC3E6H	CAN0 报文数据字节 6 寄存器 23	C0MDATA623			√		不确定
03FEC3E7H	CAN0 报文数据字节 7 寄存器 23	C0MDATA723			√		不确定
03FEC3E8H	CAN0 报文数据长度寄存器 23	C0MDLC23			√		0000xxxxB
03FEC3E9H	CAN0 报文结构寄存器 23	C0MCONF23			√		不确定
03FEC3EAH	CAN0 报文标识符寄存器 23	C0MIDL23				√	不确定
03FEC3ECH		C0MIDH23				√	不确定
03FEC3EEH	CAN0 报文控制寄存器 23	C0MCTRL23				√	00x00000 000xx000B

表 20-16. 寄存器访问类型(14/17)

地址	寄存器名称	符号	R/W	位操作单元			复位后
				1 位	8 位	16 位	
03FEC400H	CAN0 报文数据字节 01 寄存器 24	C0MDATA0124	R/W			√	不确定
03FEC400H	CAN0 报文数据字节 0 寄存器 24	C0MDATA024			√		不确定
03FEC401H	CAN0 报文数据字节 1 寄存器 24	C0MDATA124			√		不确定
03FEC402H	CAN0 报文数据字节 23 寄存器 24	C0MDATA2324				√	不确定
03FEC402H	CAN0 报文数据字节 2 寄存器 24	C0MDATA224			√		不确定
03FEC403H	CAN0 报文数据字节 3 寄存器 24	C0MDATA324			√		不确定
03FEC404H	CAN0 报文数据字节 45 寄存器 24	C0MDATA4524				√	不确定
03FEC404H	CAN0 报文数据字节 4 寄存器 24	C0MDATA424			√		不确定
03FEC405H	CAN0 报文数据字节 5 寄存器 24	C0MDATA524			√		不确定
03FEC406H	CAN0 报文数据字节 67 寄存器 24	C0MDATA6724				√	不确定
03FEC406H	CAN0 报文数据字节 6 寄存器 24	C0MDATA624			√		不确定
03FEC407H	CAN0 报文数据字节 7 寄存器 24	C0MDATA724			√		不确定
03FEC408H	CAN0 报文数据长度寄存器 24	C0MDLC24			√		0000xxxxB
03FEC409H	CAN0 报文结构寄存器 24	C0MCONF24			√		不确定
03FEC40AH	CAN0 报文标识符寄存器 24	C0MIDL24				√	不确定
03FEC40CH		C0MIDH24				√	不确定
03FEC40EH	CAN0 报文控制寄存器 24	C0MCTRL24				√	00x00000 000xx000B
03FEC420H	CAN0 报文数据字节 01 寄存器 25	C0MDATA0125				√	不确定
03FEC420H	CAN0 报文数据字节 0 寄存器 25	C0MDATA025			√		不确定
03FEC421H	CAN0 报文数据字节 1 寄存器 25	C0MDATA125			√		不确定
03FEC422H	CAN0 报文数据字节 23 寄存器 25	C0MDATA2325				√	不确定
03FEC422H	CAN0 报文数据字节 2 寄存器 25	C0MDATA225			√		不确定
03FEC423H	CAN0 报文数据字节 3 寄存器 25	C0MDATA325			√		不确定
03FEC424H	CAN0 报文数据字节 45 寄存器 25	C0MDATA4525				√	不确定
03FEC424H	CAN0 报文数据字节 4 寄存器 25	C0MDATA425			√		不确定
03FEC425H	CAN0 报文数据字节 5 寄存器 25	C0MDATA525			√		不确定
03FEC426H	CAN0 报文数据字节 67 寄存器 25	C0MDATA6725				√	不确定
03FEC426H	CAN0 报文数据字节 6 寄存器 25	C0MDATA625			√		不确定
03FEC427H	CAN0 报文数据字节 7 寄存器 25	C0MDATA725			√		不确定
03FEC428H	CAN0 报文数据长度寄存器 25	C0MDLC25			√		0000xxxxB
03FEC429H	CAN0 报文结构寄存器 25	C0MCONF25			√		不确定
03FEC42AH	CAN0 报文标识符寄存器 25	C0MIDL25				√	不确定
03FEC42CH		C0MIDH25				√	不确定
03FEC42EH	CAN0 报文控制寄存器 25	C0MCTRL25				√	00x00000 000xx000B

表 20-16. 寄存器访问类型(15/17)

地址	寄存器名称	符号	R/W	位操作单元			复位后
				1 位	8 位	16 位	
03FEC440H	CAN0 报文数据字节 01 寄存器 26	C0MDATA0126	R/W			√	不确定
03FEC440H	CAN0 报文数据字节 0 寄存器 26	C0MDATA026			√		不确定
03FEC441H	CAN0 报文数据字节 1 寄存器 26	C0MDATA126			√		不确定
03FEC442H	CAN0 报文数据字节 23 寄存器 26	C0MDATA2326				√	不确定
03FEC442H	CAN0 报文数据字节 2 寄存器 26	C0MDATA226			√		不确定
03FEC443H	CAN0 报文数据字节 3 寄存器 26	C0MDATA326			√		不确定
03FEC444H	CAN0 报文数据字节 45 寄存器 26	C0MDATA4526				√	不确定
03FEC444H	CAN0 报文数据字节 4 寄存器 26	C0MDATA426			√		不确定
03FEC445H	CAN0 报文数据字节 5 寄存器 26	C0MDATA526			√		不确定
03FEC446H	CAN0 报文数据字节 67 寄存器 26	C0MDATA6726				√	不确定
03FEC446H	CAN0 报文数据字节 6 寄存器 26	C0MDATA626			√		不确定
03FEC447H	CAN0 报文数据字节 7 寄存器 26	C0MDATA726			√		不确定
03FEC448H	CAN0 报文数据长度寄存器 26	C0MDLC26			√		0000xxxxB
03FEC449H	CAN0 报文结构寄存器 26	C0MCONF26			√		不确定
03FEC44AH	CAN0 报文标识符寄存器 26	C0MIDL26				√	不确定
03FEC44CH		C0MIDH26				√	不确定
03FEC44EH	CAN0 报文控制寄存器 26	C0MCTRL26				√	00x00000 000xx000B
03FEC460H	CAN0 报文数据字节 01 寄存器 27	C0MDATA0127				√	不确定
03FEC460H	CAN0 报文数据字节 0 寄存器 27	C0MDATA027			√		不确定
03FEC461H	CAN0 报文数据字节 1 寄存器 27	C0MDATA127			√		不确定
03FEC462H	CAN0 报文数据字节 23 寄存器 27	C0MDATA2327				√	不确定
03FEC462H	CAN0 报文数据字节 2 寄存器 27	C0MDATA227			√		不确定
03FEC463H	CAN0 报文数据字节 3 寄存器 27	C0MDATA327			√		不确定
03FEC464H	CAN0 报文数据字节 45 寄存器 27	C0MDATA4527				√	不确定
03FEC464H	CAN0 报文数据字节 4 寄存器 27	C0MDATA427			√		不确定
03FEC465H	CAN0 报文数据字节 5 寄存器 27	C0MDATA527			√		不确定
03FEC466H	CAN0 报文数据字节 67 寄存器 27	C0MDATA6727				√	不确定
03FEC466H	CAN0 报文数据字节 6 寄存器 27	C0MDATA627			√		不确定
03FEC467H	CAN0 报文数据字节 7 寄存器 27	C0MDATA727			√		不确定
03FEC468H	CAN0 报文数据长度寄存器 27	C0MDLC27			√		0000xxxxB
03FEC469H	CAN0 报文结构寄存器 27	C0MCONF27			√		不确定
03FEC46AH	CAN0 报文标识符寄存器 27	C0MIDL27				√	不确定
03FEC46CH		C0MIDH27				√	不确定
03FEC46EH	CAN0 报文控制寄存器 27	C0MCTRL27				√	00x00000 000xx000B

表 20-16. 寄存器访问类型(16/17)

地址	寄存器名称	符号	R/W	位操作单元			复位后
				1 位	8 位	16 位	
03FEC480H	CAN0 报文数据字节 01 寄存器 28	C0MDATA0128	R/W			√	不确定
03FEC480H	CAN0 报文数据字节 0 寄存器 28	C0MDATA028			√		不确定
03FEC481H	CAN0 报文数据字节 1 寄存器 28	C0MDATA128			√		不确定
03FEC482H	CAN0 报文数据字节 23 寄存器 28	C0MDATA2328				√	不确定
03FEC482H	CAN0 报文数据字节 2 寄存器 28	C0MDATA228			√		不确定
03FEC483H	CAN0 报文数据字节 3 寄存器 28	C0MDATA328			√		不确定
03FEC484H	CAN0 报文数据字节 45 寄存器 28	C0MDATA4528				√	不确定
03FEC484H	CAN0 报文数据字节 4 寄存器 28	C0MDATA428			√		不确定
03FEC485H	CAN0 报文数据字节 5 寄存器 28	C0MDATA528			√		不确定
03FEC486H	CAN0 报文数据字节 67 寄存器 28	C0MDATA6728				√	不确定
03FEC486H	CAN0 报文数据字节 6 寄存器 28	C0MDATA628			√		不确定
03FEC487H	CAN0 报文数据字节 7 寄存器 28	C0MDATA728			√		不确定
03FEC488H	CAN0 报文数据长度寄存器 28	C0MDLC28			√		0000xxxxB
03FEC489H	CAN0 报文结构寄存器 28	C0MCONF28			√		不确定
03FEC48AH	CAN0 报文标识符寄存器 28	C0MIDL28				√	不确定
03FEC48CH		C0MIDH28				√	不确定
03FEC48EH	CAN0 报文控制寄存器 28	C0MCTRL28				√	00x00000 000xx000B
03FEC4A0H	CAN0 报文数据字节 01 寄存器 29	C0MDATA0129				√	不确定
03FEC4A0H	CAN0 报文数据字节 0 寄存器 29	C0MDATA029			√		不确定
03FEC4A1H	CAN0 报文数据字节 1 寄存器 29	C0MDATA129			√		不确定
03FEC4A2H	CAN0 报文数据字节 23 寄存器 29	C0MDATA2329				√	不确定
03FEC4A2H	CAN0 报文数据字节 2 寄存器 29	C0MDATA229			√		不确定
03FEC4A3H	CAN0 报文数据字节 3 寄存器 29	C0MDATA329			√		不确定
03FEC4A4H	CAN0 报文数据字节 45 寄存器 29	C0MDATA4529				√	不确定
03FEC4A4H	CAN0 报文数据字节 4 寄存器 29	C0MDATA429			√		不确定
03FEC4A5H	CAN0 报文数据字节 5 寄存器 29	C0MDATA529			√		不确定
03FEC4A6H	CAN0 报文数据字节 67 寄存器 29	C0MDATA6729				√	不确定
03FEC4A6H	CAN0 报文数据字节 6 寄存器 29	C0MDATA629			√		不确定
03FEC4A7H	CAN0 报文数据字节 7 寄存器 29	C0MDATA729			√		不确定
03FEC4A8H	CAN0 报文数据长度寄存器 29	C0MDLC29			√		0000xxxxB
03FEC4A9H	CAN0 报文结构寄存器 29	C0MCONF29			√		不确定
03FEC4AAH	CAN0 报文标识符寄存器 29	C0MIDL29				√	不确定
03FEC4ACH		C0MIDH29				√	不确定
03FEC4AEH	CAN0 报文控制寄存器 29	C0MCTRL29				√	00x00000 000xx000B

表 20-16. 寄存器访问类型(17/17)

地址	寄存器名称	符号	R/W	位操作单元			复位后
				1 位	8 位	16 位	
03FEC4C0H	CAN0 报文数据字节 01 寄存器 30	C0MDATA0130	R/W			√	不确定
03FEC4C0H	CAN0 报文数据字节 0 寄存器 30	C0MDATA030			√		不确定
03FEC4C1H	CAN0 报文数据字节 1 寄存器 30	C0MDATA130			√		不确定
03FEC4C2H	CAN0 报文数据字节 23 寄存器 30	C0MDATA2330				√	不确定
03FEC4C2H	CAN0 报文数据字节 2 寄存器 30	C0MDATA230			√		不确定
03FEC4C3H	CAN0 报文数据字节 3 寄存器 30	C0MDATA330			√		不确定
03FEC4C4H	CAN0 报文数据字节 45 寄存器 30	C0MDATA4530				√	不确定
03FEC4C4H	CAN0 报文数据字节 4 寄存器 30	C0MDATA430			√		不确定
03FEC4C5H	CAN0 报文数据字节 5 寄存器 30	C0MDATA530			√		不确定
03FEC4C6H	CAN0 报文数据字节 67 寄存器 30	C0MDATA6730				√	不确定
03FEC4C6H	CAN0 报文数据字节 6 寄存器 30	C0MDATA630			√		不确定
03FEC4C7H	CAN0 报文数据字节 7 寄存器 30	C0MDATA730			√		不确定
03FEC4C8H	CAN0 报文数据长度寄存器 30	C0MDLC30			√		0000xxxxB
03FEC4C9H	CAN0 报文结构寄存器 30	C0MCONF30			√		不确定
03FEC4CAH	CAN0 报文标识符寄存器 30	C0MIDL30				√	不确定
03FEC4CCH		C0MIDH30				√	不确定
03FEC4CEH	CAN0 报文控制寄存器 30	C0MCTRL30				√	00x00000 000xx000B
03FEC4E0H	CAN0 报文数据字节 01 寄存器 31	C0MDATA0131				√	不确定
03FEC4E0H	CAN0 报文数据字节 0 寄存器 31	C0MDATA031			√		不确定
03FEC4E1H	CAN0 报文数据字节 1 寄存器 31	C0MDATA131			√		不确定
03FEC4E2H	CAN0 报文数据字节 23 寄存器 31	C0MDATA2331				√	不确定
03FEC4E2H	CAN0 报文数据字节 2 寄存器 31	C0MDATA231			√		不确定
03FEC4E3H	CAN0 报文数据字节 3 寄存器 31	C0MDATA331			√		不确定
03FEC4E4H	CAN0 报文数据字节 45 寄存器 31	C0MDATA4531				√	不确定
03FEC4E4H	CAN0 报文数据字节 4 寄存器 31	C0MDATA431			√		不确定
03FEC4E5H	CAN0 报文数据字节 5 寄存器 31	C0MDATA531			√		不确定
03FEC4E6H	CAN0 报文数据字节 67 寄存器 31	C0MDATA6731				√	不确定
03FEC4E6H	CAN0 报文数据字节 6 寄存器 31	C0MDATA631			√		不确定
03FEC4E7H	CAN0 报文数据字节 7 寄存器 31	C0MDATA731			√		不确定
03FEC4E8H	CAN0 报文数据长度寄存器 31	C0MDLC31			√		0000xxxxB
03FEC4E9H	CAN0 报文结构寄存器 31	C0MCONF31			√		不确定
03FEC4EAH	CAN0 报文标识符寄存器 31	C0MIDL31				√	不确定
03FEC4ECH		C0MIDH31				√	不确定
03FEC4EEH	CAN0 报文控制寄存器 31	C0MCTRL31				√	00x00000 000xx000B

20.5.3 寄存器位配置

表 20-17. CAN 全局寄存器位配置

地址	符号	位 7/15	位 6/14	位 5/13	位 4/12	位 3/11	位 2/10	位 1/9	位 0/8
03FEC000H	C0GMCTRL (W)	0	0	0	0	0	0	0	清除 GOM
03FEC001H		0	0	0	0	0	0	设置 EFSD	设置 GOM
03FEC000H	C0GMCTRL (R)	0	0	0	0	0	0	EFSD	GOM
03FEC001H		MBON	0	0	0	0	0	0	0
03FEC002H	C0GMCS	0	0	0	0	CCP3	CCP2	CCP1	CCP0
03FEC006H	C0GMABT (W)	0	0	0	0	0	0	0	清除 ABTTRG
03FEC007H		0	0	0	0	0	0	设置 ABTCLR	设置 ABTTRG
03FEC006H	C0GMABT (R)	0	0	0	0	0	0	ABTCLR	ABTTRG
03FEC007H		0	0	0	0	0	0	0	0
03FEC008H	C0GMABTD	0	0	0	0	ABTD3	ABTD2	ABTD1	ABTD0

表 20-18. CAN 模块寄存器位配置 (1/2)

地址	符号	位 7/15	位 6/14	位 5/13	位 4/12	位 3/11	位 2/10	位 1/9	位 0/8
03FEC040H	C0MASK1L	CMID7 至 CMID0							
03FEC041H		CMID15 至 CMID8							
03FEC042H	C0MASK1H	CMID23 至 CMID16							
03FEC043H		0	0	0	CMID28 至 CMID24				
03FEC044H	C0MASK2L	CMID7 至 CMID0							
03FEC045H		CMID15 至 CMID8							
03FEC046H	C0MASK2H	CMID23 至 CMID16							
03FEC047H		0	0	0	CMID28 至 CMID24				
03FEC048H	C0MASK3L	CMID7 至 CMID0							
03FEC049H		CMID15 至 CMID8							
03FEC04AH	C0MASK3H	CMID23 至 CMID16							
03FEC04BH		0	0	0	CMID28 至 CMID24				
03FEC04CH	C0MASK4L	CMID7 至 CMID0							
03FEC04DH		CMID15 至 CMID8							
03FEC04EH	C0MASK4H	CMID23 至 CMID16							
03FEC04FH		0	0	0	CMID28 至 CMID24				
03FEC050H	C0CTRL (W)	0	清除 AL	清除 VALID	清除 PSMODE1	清除 PSMODE0	清除 OPMODE2	清除 OPMODE1	清除 OPMODE0
03FEC051H		设置 CCERC	设置 AL	0	设置 PSMODE1	设置 PSMODE0	设置 OPMODE2	设置 OPMODE1	设置 OPMODE0
03FEC050H	C0CTRL (R)	CCERC	AL	VALID	PS MODE1	PS MODE0	OP MODE2	OP MODE1	OP MODE0
03FEC051H		0	0	0	0	0	0	RSTAT	TSTAT
03FEC052H	C0LEC (W)	0	0	0	0	0	0	0	0
03FEC052H	C0LEC (R)	0	0	0	0	0	LEC2	LEC1	LEC0
03FEC053H	C0INFO	0	0	0	BOFF	TECS1	TECS0	RECS1	RECS0
03FEC054H	C0ERC	TEC7 至 TEC0							
03FEC055H		REC7 至 REC0							
03FEC056H	C0IE (W)	0	0	清除 CIE5	清除 CIE4	清除 CIE3	清除 CIE2	清除 CIE1	清除 CIE0
03FEC057H		0	0	设置 CIE5	设置 CIE4	设置 CIE3	设置 CIE2	设置 CIE1	设置 CIE0
03FEC056H	C0IE (R)	0	0	CIE5	CIE4	CIE3	CIE2	CIE1	CIE0
03FEC057H		0	0	0	0	0	0	0	0
03FEC058H	C0INTS (W)	0	0	清除 CINTS5	清除 CINTS4	清除 CINTS3	清除 CINTS2	清除 CINTS1	清除 CINTS0
03FEC059H		0	0	0	0	0	0	0	0
03FEC058H	C0INTS (R)	0	0	CINTS5	CINTS4	CINTS3	CINTS2	CINTS1	CINTS0
03FEC059H		0	0	0	0	0	0	0	0

表 20-18. CAN 模块寄存器位配置 (2/2)

地址	符号	位 7/15	位 6/14	位 5/13	位 4/12	位 3/11	位 2/10	位 1/9	位 0/8
03FEC05AH	C0BRP	TQPRS7 至 TQPRS0							
03FEC05CH	C0BTR	0	0	0	0	TSEG13 至 TSEG10			
03FEC05DH		0	0	SJW1, SJW0		0	TSEG22 至 TSEG20		
03FEC05EH	C0LIPT	LIPT7 至 LIPT0							
03FEC060H	C0RGPT (W)	0	0	0	0	0	0	0	清除 ROVF
03FEC061H		0	0	0	0	0	0	0	0
03FEC060H	C0RGPT (R)	0	0	0	0	0	0	RHPM	ROVF
03FEC061H		RGPT7 至 RGPT0							
03FEC062H	C0LOPT	LOPT7 至 LOPT0							
03FEC064H	C0TGPT (W)	0	0	0	0	0	0	0	清除 TOVF
03FEC065H		0	0	0	0	0	0	0	0
03FEC064H	C0TGPT (R)	0	0	0	0	0	0	THPM	TOVF
03FEC065H		TGPT7 至 TGPT0							
03FEC066H	C0TS (W)	0	0	0	0	0	清除 TSLOCK	清除 TSSEL	清除 TSEN
03FEC067H		0	0	0	0	0	设置 TSLOCK	设置 TSSEL	设置 TSEN
03FEC066H	C0TS (R)	0	0	0	0	0	TSLOCK	TSSEL	TSEN
03FEC067H		0	0	0	0	0	0	0	0
03FEC068H 至 03FEC0FFH	—	禁止访问（反相后将来使用）							

表 20-19. 报文缓存寄存器位配置

地址	符号	位 7/15	位 6/14	位 5/13	位 4/12	位 3/11	位 2/10	位 1/9	位 0/8
03FECxx0H	C0MDATA01m	报文数据(字节 0)							
03FECxx1H		报文数据(字节 1)							
03FECxx0H	C0MDATA0m	报文数据(字节 0)							
03FECxx1H	C0MDATA1m	报文数据(字节 1)							
03FECxx2H	C0MDATA23m	报文数据(字节 2)							
03FECxx3H		报文数据(字节 3)							
03FECxx2H	C0MDATA2m	报文数据(字节 2)							
03FECxx3H	C0MDATA3m	报文数据(字节 3)							
03FECxx4H	C0MDATA45m	报文数据(字节 4)							
03FECxx5H		报文数据(字节 5)							
03FECxx4H	C0MDATA4m	报文数据(字节 4)							
03FECxx5H	C0MDATA5m	报文数据(字节 5)							
03FECxx6H	C0MDATA67m	报文数据(字节 6)							
03FECxx7H		报文数据(字节 7)							
03FECxx6H	C0MDATA6m	报文数据(字节 6)							
03FECxx7H	C0MDATA7m	报文数据(字节 7)							
03FECxx8H	C0MDLCm	0				MDLC3	MDLC2	MDLC1	MDLC0
03FECxx9H	C0MCONFm	OWS	RTR	MT2	MT1	MT0	0	0	MA0
03FECxxAH	C0MIDLm	ID7	ID6	ID5	ID4	ID3	ID2	ID1	ID0
03FECxxBH		ID15	ID14	ID13	ID12	ID11	ID10	ID9	ID8
03FECxxCH	C0MIDHm	ID23	ID22	ID21	ID20	ID19	ID18	ID17	ID16
03FECxxDH		IDE	0	0	ID28	ID27	ID26	ID25	ID24
03FECxxEH	C0MCTRLm (W)	0	0	0	清除 MOW	清除 IE	清除 DN	清除 TRQ	清除 RDY
03FECxxFH		0	0	0	0	设置 IE	0	设置 TRQ	设置 RDY
03FECxxEH	C0MCTRLm (R)	0	0	0	MOW	IE	DN	TRQ	RDY
03FECxxFH		0	0	MUC	0	0	0	0	0
03FECxx0 to 03FECxxFH	—	禁止访问（反相后将来使用）							

备注 m = 00 至 31

xx = 10, 12, 14, 16, 18, 1A, 1C, 1E, 20, 22, 24, 26, 28, 2A, 2C, 2E, 30, 32, 34, 36, 38, 3A, 3C, 3E, 40, 42, 44, 46, 48, 4A, 4C, 4E

20.6 寄存器

注意事项 在以下状态禁止访问 CAN 控制器寄存器。 有关详细说明，参见 3.4.9 (2) 访问片上专用外设 I/O 寄存器。

- 当 CPU 工作于副时钟，而且主时钟振荡停止时
- 当 CPU 工作于内部振荡时钟时

(1) CAN0 全局控制寄存器 (C0GMCTRL)

C0GMCTRL 寄存器用于控制 CAN 模块的操作。

(1/2)

复位后: 0000H R/W 地址: 03FEC000H

(a) 读取

	15	14	13	12	11	10	9	8
C0GMCTRL	MBON	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	EFSD	GOM

(b) 写入

	15	14	13	12	11	10	9	8
C0GMCTRL	0	0	0	0	0	0	设置 EFSD	设置 GOM
	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	清除 r GOM

(a) 读取

MBON	允许访问报文缓存寄存器，发送/接收记录寄存器的位
0	对报文缓存寄存器的读写访问并禁止发送/接收记录列表寄存器
1	对报文缓存寄存器的读写访问并允许发送/接收记录列表寄存器

注意事项 1. 当将 MBON 位清除（为 0）时，禁止对报文缓存器（C0MDATA0m、C0MDATA1m、C0MDATA01m、C0MDATA2m、C0MDATA3m、C0MDATA23m、C0MDATA4m、C0MDATA5m、C0MDATA45m、C0MDATA6m、C0MDATA7m、C0MDATA67m、C0MDLCm、C0MCONFm、C0MIDLm、C0MIDHm 以及 C0MCTRLm）进行软件访问，也禁止访问与发送记录或接收记录相关的寄存器（C0LOPT、C0TGPT、C0LIPT 以及 C0RGPT）。

2. 该位为只读位。 当 MBON 位为 0 时即使向其写入 1，MBON 的值也不会改变，且仍禁止访问报文缓存寄存器和与发送记录或接收记录相关的寄存器。

备注 当进入 CAN 睡眠模式/CAN 停止模式，或 GOM 位清除为 0 时，MBON 位清除为 0。当解除 CAN 睡眠模式/CAN 停止模式或将 GOM 位设为 1 时，将 MBON 位设为 1。

EFSD	允许强制关闭的位
0	GOM 位= 0 时禁止强制关闭
1	GOM 位= 0 时允许强制关闭

注意事项 要请求强制关闭，必须在 EFSD 位设为 1 之后立即将 GOM 位清为 0。如果在 EFSD 位设为 1 之后不清除 GOM 的情况下立即对其它寄存器进行访问（包括读取 C0GMCTRL 寄存器），那么 EFSD 位自动清除为 0，且强制关闭请求无效。

GOM	全局操作模式位
0	禁止 CAN 模块操作
1	允许 CAN 模块操作

注意事项 仅在初始化模式下或 EFSD 位设为 1 时将 GOM 位清除为 0。

(b) 写入

设置 EFSD	EFSD 位设置
0	EFSD 位不改变。
1	EFSD 位设为 1。

设置 GOM	清除 GOM	GOM 位设置
0	1	GOM 位清除为 0。
1	0	GOM 位设置为 1。
其它		GOM 位不改变。

(2) CAN0 全局时钟选择寄存器 (C0GMCS)

C0GMCS 寄存器用于选择 CAN 模块的系统时钟。

复位后: 0FH R/W 地址: 03FEC002H

	7	6	5	4	3	2	1	0
C0GMCS	0	0	0	0	CCP3	CCP2	CCP1	CCP0

CCP3	CCP2	CCP1	CCP1	CAN 模块系统时钟 (f_{CANMOD})
0	0	0	0	$f_{CAN}/1$
0	0	0	1	$f_{CAN}/2$
0	0	1	0	$f_{CAN}/3$
0	0	1	1	$f_{CAN}/4$
0	1	0	0	$f_{CAN}/5$
0	1	0	1	$f_{CAN}/6$
0	1	1	0	$f_{CAN}/7$
0	1	1	1	$f_{CAN}/8$
1	0	0	0	$f_{CAN}/9$
1	0	0	1	$f_{CAN}/10$
1	0	1	0	$f_{CAN}/11$
1	0	1	1	$f_{CAN}/12$
1	1	0	0	$f_{CAN}/13$
1	1	0	1	$f_{CAN}/14$
1	1	1	0	$f_{CAN}/15$
1	1	1	1	$f_{CAN}/16$ (默认值)

备注 f_{CAN} = 提供给 CAN 的时钟 = f_{xx}

(3) CAN0 全局块传输控制寄存器 (C0GMABT)

C0GMABT 寄存器用于控制自动块传输 (ABT) 操作。

(1/2)

复位后: 0000H R/W 地址: 03FEC006H

(a) 读取

	15	14	13	12	11	10	9	8
C0GMABT	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	ABTCLR	ABTTRG

(a) 写入

	15	14	13	12	11	10	9	8
C0GMABT	0	0	0	0	0	0	设置 ABTCLR	设置 ABTTRG
	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	清除 ABTTRG

注意事项 在使用 **ABT** 的正常操作模式改变为初始化模式之前, 务必将 **C0GMABT** 寄存器设为默认值 (0000H)。设置之后, 确认 **C0GMABT** 寄存器初始化为 0000H。

(a) 读取

ABTCLR	自动块传输引擎清除状态位
0	完成自动引擎传输的清除。
1	正在进行自动传输引擎的清除。

- 备注**
1. 在 **ABTTRG** 清为 0 时将 **ABTCLR** 位设为 1。如果在 **ABTTRG** 位设为 1 时也将 **ABTCLR** 位设为 1 则操作得不到保证。
 2. 通过将 **ABTCLR** 设为 1 清除自动块传输引擎时, 已请求的清除处理完成之后, **ABTCLR** 位立即自动清除为 0。

ABTTRG	自动块传输状态位
0	停止自动块传输。
1	执行自动块传输。

- 注意事项**
1. 在初始化模式不要设置 **ABTTRG** 位。
如果在初始化模式下将 **ABTTRG** 位设为 1, 则 **CAN** 模块进入使用 **ABT** 的正常操作模式之后的操作得不到保证。
 2. **C0CTRL.TSTAT** 位设为 1 期间不要将 **ABTTRG** 位设为 1。在 **ABTTRG** 设为 1 之前确保 **TSTAT** 位= 0。

(2/2)

(b) 写入

设置 ABTCLR	自动块传输引擎清除请求位
0	自动块传输引擎处于空闲状态或操作状态。
1	请求清除自动块传输引擎 清除自动块传输引擎之后，通过将 ABTTRG 位设为 1 开始从报文缓存器 0 开始自动块传输。

设置 ABTTRG	清除 ABTTRG	自动块传输起始位
0	1	请求停止自动块传输
1	0	请求开始自动块传输
其它		ABTTRG 位不改变。

(4) CAN0 全局块传输延时设置寄存器 (C0GMABTD)

C0GMABTD 寄存器用于设置间隔，间隔内赋值给 ABT 的报文缓存器的数据将在使用 ABT 的正常操作模式中发送。

复位后: 00H R/W 地址: 03FEC008H

	7	6	5	4	3	2	1	0
C0GMABTD	0	0	0	0	ABTD3	ABTD2	ABTD1	ABTD0

ABTD3	ABTD2	ABTD1	ABTD0	自动块传输期间的数据帧间隔 (单位: 数据位时间 (DBT))
0	0	0	0	0 DBT (默认值)
0	0	0	1	2^5 DBT
0	0	1	0	2^6 DBT
0	0	1	1	2^7 DBT
0	1	0	0	2^8 DBT
0	1	0	1	2^9 DBT
0	1	1	0	2^{10} DBT
0	1	1	1	2^{11} DBT
1	0	0	0	2^{12} DBT
其它				禁止设置

- 注意事项
1. ABTTRG 设为 1 时不要改变 C0GMABTD 寄存器的内容。
 2. ABT 报文实际发送到 CAN 总线的时刻，随其它终端的发送状态，或如何产生发送非 ABT 报文（报文缓存器 8 至 31）的请求不同而异。

(5) CAN0 模块屏蔽控制寄存器 (C0MASKaL, C0MASKaH) (a = 1, 2, 3 或 4)

C0MASKaL 和 C0MASKaH 寄存器通过屏蔽报文的部分标识符 (ID) 并使屏蔽部分无效, 从而用来扩展同一个报文缓存器可接收的报文数目。

(1/2)

• CAN0 模块屏蔽 1 寄存器 (C0MASK1L, C0MASK1H)

复位后: 不确定

R/W

地址: C0MASK1L 03FEC040H, C0MASK1H 03FEC042H

	15	14	13	12	11	10	9	8
C0MASK1L	CMID15	CMID14	CMID13	CMID12	CMID11	CMID10	CMID9	CMID8
	7	6	5	4	3	2	1	0
	CMID7	CMID6	CMID5	CMID4	CMID3	CMID2	CMID1	CMID0
	15	14	13	12	11	10	9	8
C0MASK1H	0	0	0	CMID28	CMID27	CMID26	CMID25	CMID24
	7	6	5	4	3	2	1	0
	CMID23	CMID22	CMID21	CMID20	CMID19	CMID18	CMID17	CMID16

• CAN0 模块屏蔽 2 寄存器 (C0MASK2L, C0MASK2H)

复位后: 未定义

R/W

地址: C0MASK2L 03FEC044H, C0MASK2H 03FEC046H

	15	14	13	12	11	10	9	8
C0MASK2L	CMID15	CMID14	CMID13	CMID12	CMID11	CMID10	CMID9	CMID8
	7	6	5	4	3	2	1	0
	CMID7	CMID6	CMID5	CMID4	CMID3	CMID2	CMID1	CMID0
	15	14	13	12	11	10	9	8
C0MASK2H	0	0	0	CMID28	CMID27	CMID26	CMID25	CMID24
	7	6	5	4	3	2	1	0
	CMID23	CMID22	CMID21	CMID20	CMID19	CMID18	CMID17	CMID16

• CAN0 模块屏蔽 3 寄存器（C0MASK1L, C0MASK1H）

复位后：未定义 R/W 地址： C0MASK3L 03FEC048H, C0MASK3H 03FEC04AH

	15	14	13	12	11	10	9	8
C0MASK3L	CMID15	CMID14	CMID13	CMID12	CMID11	CMID10	CMID9	CMID8
	7	6	5	4	3	2	1	0
	CMID7	CMID6	CMID5	CMID4	CMID3	CMID2	CMID1	CMID0
	15	14	13	12	11	10	9	8
C0MASK3H	0	0	0	CMID28	CMID27	CMID26	CMID25	CMID24
	7	6	5	4	3	2	1	0
	CMID23	CMID22	CMID21	CMID20	CMID19	CMID18	CMID17	CMID16

• CAN0 模块屏蔽 4 寄存器（C0MASK1L, C0MASK1H）

复位后：未定义 R/W 地址： C0MASK4L 03FEC04CH, C0MASK4H 03FEC04EH

	15	14	13	12	11	10	9	8
C0MASK4L	CMID15	CMID14	CMID13	CMID12	CMID11	CMID10	CMID9	CMID8
	7	6	5	4	3	2	1	0
	CMID7	CMID6	CMID5	CMID4	CMID3	CMID2	CMID1	CMID0
	15	14	13	12	11	10	9	8
C0MASK4H	0	0	0	CMID28	CMID27	CMID26	CMID25	CMID24
	7	6	5	4	3	2	1	0
	CMID23	CMID22	CMID21	CMID20	CMID19	CMID18	CMID17	CMID16

CMID28 至 CMID0	ID 位的屏蔽模式设置
0	将 CMID28 至 CMID0 位设置的报文缓存器的 ID 位与已接收报文帧的 ID 位进行比较。
1	不将 CMID28 至 CMID0 位设置的报文缓存器的 ID 位与已接收报文帧的 ID 位进行比较（它们被屏蔽）。

备注 屏蔽总是以 29 位 ID 长度来定义。 如果给具有标准 ID 的报文指定一个屏蔽，则忽略 CMID17 至 CMID0 位。 因此，只屏蔽了已接收 ID 的 CMID28 至 CMID18 位。 标准 ID 和扩展 ID 可使用相同的屏蔽。

(6) CAN0 模块控制寄存器 (C0CTRL)

C0CTRL 寄存器用于控制 CAN 模块的操作模式。

(1/4)

复位后: 0000H R/W 地址: 03FEC050H

(a) 读取

	15	14	13	12	11	10	9	8
C0CTRL	0	0	0	0	0	0	RSTAT	TSTAT
	7	6	5	4	3	2	1	0
	CCERC	AL	VALID	PSMODE 1	PSMODE 0	OPMODE 2	OPMODE 1	OPMODE 0

(a) 写入

	15	14	13	12	11	10	9	8
C0CTRL	设置 CCERC	设置 AL	0	设置 PSMODE 1	设置 PSMODE 0	设置 OPMODE 2	设置 OPMODE 1	设置 OPMODE 0
	7	6	5	4	3	2	1	0
	0	清除 AL	清除 VALID	清除 PSMODE 1	清除 PSMODE 0	清除 OPMODE 2	清除 OPMODE 1	清除 OPMODE 0

(a) 读取

RSTAT	接收状态位
0	停止接收。
1	正在接收。

备注 在下列条件（时刻）下将 RSTAT 位设为 1

- 检测到接收帧的 SOF 位时
- 在一个发送帧期间发生仲裁失败时

在下列条件（时刻）下将 RSTAT 位清为 0

- 在帧间间隔的第 2 位检测到隐性电平时
- 在帧间间隔的第 1 位转换至初始化模式时

TSTAT	发送状态位
0	停止发送。
1	正在发送。

备注 在下列条件（时刻）下将 TSTAT 位设为 1

- 检测到发送帧的 SOF 位时
- 在发送帧中检测到错误标志的第 1 位。

在下列条件（时刻）下将 TSTAT 位清为 0

- 切换至掉线在状态期间
- 在一个发送帧期间发生仲裁失败时
- 在帧间间隔的第 2 位检测到隐性电平
- 在帧间间隔的第 1 位转换至初始化模式时

CCERC	错误计数器清除位
0	初始化模式下不清除 C0ERC 和 C0INFO 寄存器。
1	初始化模式下清除 C0ERC 和 C0INFO 寄存器。

备注 1. CCERC 位用于重新初始化或从掉线状态强制恢复时清除 C0ERC 和 C0INFO 寄存器。该位仅能在初始化模式下设为 1。

2. 清除 C0ERC 和 C0INFO 寄存器之后，CCERC 位也自动清除为 0。

3. 在初始化模式变为操作模式的请求产生的同时可将 CCERC 位设为 1。

4. 如果在自检模式进入到 INTI 模式后随即将 CCERC 位设为 1，则可能会破坏接收数据。

AL	仲裁失败时的操作设置位
0	单次模式下当仲裁失败时不进行重传。
1	单次模式下当仲裁失败时进行重传。

备注 AL 位仅在单次模式下有效。

VALID	有效接收报文帧检测位
0	从 VALID 位上一次清为 0 后未接收到有效报文帧。
1	从 VALID 位上一次清为 0 后已接收到有效报文帧。

- 备注**
1. 有效接收数据帧的检测不依赖于接收报文缓存器（数据帧）或发送报文缓存器（远程帧）是否存有数据。
 2. 在初始化模式变为操作模式之前将 VALID 位清除（0）。
 3. 如果只有两个 CAN 节点连接到 CAN 总线，其中一个节点在正常模式下传送报文而另一个在只收模式下传送报文，所以在发送节点进入错误既成状态前 VALID 位不会置为 1。
 4. 要清除 VALID 位，首先将 Clear VALID 位设为 1 然后确认 VALID 位清除。 如果没有清除该位，则重新进行清除处理。

PSMODE1	PSMODE0	节电模式
0	0	未选择节电模式
0	1	CAN 睡眠模式
1	0	禁止设置
1	1	CAN 停止模式

注意事项 必须通过 CAN 睡眠模式才能进行 CAN 停止模式的转换。 直接到（从）CAN 停止模式的转换请求将被忽略。

OPMODE2	OPMODE1	OPMODE0	操作模式
0	0	0	未选择操作模式（CAN 模块处于初始化模式）。
0	0	1	正常操作模式
0	1	0	使用自动块传输功能的正常操作模式（使用 ABT 的正常操作模式）
0	1	1	只收模式
1	0	0	单次模式
1	0	1	自检模式
其它			禁止设置

备注 在 CAN 睡眠模式或 CAN 停止模式模式下，OPMODE0 至 OPMODE2 位为只读位。

(b) 写入

Set CCERC	CCERC 位的设置
1	CCERC 位设为 1。
其它	CCERC 位不改变。

设置 AL	清除 AL	AL 位的设置
0	1	AL 位清为 0。
1	0	AL 位设为 1。
其它		AL 位不改变。

清除 VALID	VALID 位的设置
0	VALID 位不改变。
1	VALID 位清为 0。

设置 PSMODE0	清除 PSMODE0	PSMODE0 位的设置
0	1	PSMODE0 位清为 0。
1	0	PSMODE0 位设为 1。
其它		PSMODE0 位不改变。

设置 PSMODE1	清除 PSMODE1	PSMODE 位的设置
0	1	PSMODE1 位清为 0。
1	0	PSMODE1 位清为 1。
其它		PSMODE1 位不改变。

设置 OPMODE0	清除 OPMODE0	OPMODE0 位的设置
0	1	PSMODE0 位清为 0。
1	0	PSMODE0 位清为 1。
其它		PSMODE0 位不改变。

设置 OPMODE1	清除 OPMODE1	OPMODE1 位的设置
0	1	OPMODE1 位清为 0。
1	0	OPMODE1 位设为 1。
其它		OPMODE1 位不改变。

设置 OPMODE2	清除 OPMODE2	OPMODE2 位的设置
0	1	OPMODE2 位清为 0。
1	0	OPMODE2 位设为 1。
其它		OPMODE2 位不改变。

(7) CAN0 模块最末错误信息寄存器 (C0LEC)

C0LEC 寄存器提供 CAN 协议的错误信息。

复位后: 00H R/W 地址: 03FEC052H

	7	6	5	4	3	2	1	0
C0LEC	0	0	0	0	0	LEC2	LEC1	LEC0

- 备注**
1. 当 CAN 模块由操作模式变为初始化模式时不清除 C0LEC 寄存器的内容。
 2. 如果试图通过软件将非 00H 的值写入 C0LEC 寄存器，则该访问被忽略。

LEC2	LEC1	LEC0	最末 CAN 协议错误信息
0	0	0	没有错误
0	0	1	填充错误
0	1	0	格式错误
0	1	1	ACK 错误
1	0	0	位错误 (CAN 模块试图将一个隐性电平位作为发送报文 (除了仲裁域) 的一部分来发送, 但是 CAN 总线上的值却为显性电平。)
1	0	1	位错误 (CAN 模块试图将一个隐性电平位作为发送报文、ACK 位或过载帧的一部分来发送, 但是 CAN 总线上的值却为隐性电平。)
1	1	0	CRC 错误
1	1	1	未定义

(8) CAN0 模块信息寄存器 (C0INFO)

C0INFO 寄存器表示 CAN 模块的状态。

复位后: 00H R 地址: 03FEC053H

	7	6	5	4	3	2	1	0
C0INFO	0	0	0	BOFF	TECS1	TECS0	RECS1	RECS0

BOFF	掉线状态位
0	非掉线状态 (发送错误计数器≤255)。(发送计数器的值小于 256)
1	掉线状态 (发送错误计数器>255)。(发送计数器的值为 256 或更多)

TECS1	TECS0	发送错误计数器状态位
0	0	发送错误计数器的值小于告警水平的值 (< 96)。
0	1	发送错误计数器的值在告警水平范围内 (96 至 127)。
1	0	未定义
1	1	发送错误计数器的值在错误既成的范围内或掉线状态 (≥ 128)。

RECS1	RECS0	接收错误计数器状态位
0	0	接收错误计数器的值小于告警水平的值 (< 96)。
0	1	接收错误计数器的值在告警水平范围内 (96 至 127)。
1	0	未定义
1	1	接收计数器的值在错误既成范围内 (≥ 128)。

(9) CAN0 模块错误计数器寄存器 (C0ERC)

C0ERC 寄存器表明发送/接收错误计数器的值。

复位后: 0000H R 地址: 03FEC054H

	15	14	13	12	11	10	9	8
C0ERC	REPS	REC6	REC5	REC4	REC3	REC2	REC1	REC0
	7	6	5	4	3	2	1	0
	TEC7	TEC6	TEC5	TEC4	TEC3	TEC2	TEC1	TEC0

REPS	接收错误既成状态位
0	接收错误计数器的值不在错误既成范围 (< 128)
1	接收错误计数器的值在错误既成范围内 (≥ 128)。

REC6 至 REC0	接收错误计数器位
0 至 127	接收错误数 这些位反映接收错误计数器的状态。 错误数目由 CAN 协议定义。

备注 接收错误计数器的 REC6 至 REC0 位在接收错误既成状态无效 (C0INFO.RECS1, C0INFO.RECS0 位 = 11B)。

TEC7 至 TEC0	发送错误计数器位
0 至 255	发送错误数 这些位反映发送错误计数器的状态。 错误数目由 CAN 协议定义。

备注 发送错误计数器的 TEC7 至 TEC0 位在掉线状态无效 (C0INFO.BOFF 位= 1)

(10) CAN0 模块中断使能寄存器 (C0IE)

C0IE 寄存器用于使能或禁止 CAN 模块的中断。

(1/2)

复位后: 0000H R/W 地址: 03FEC056H

(a) 读取

	15	14	13	12	11	10	9	8
C0IE	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
	0	0	CIE5	CIE4	CIE3	CIE2	CIE1	CIE0

(b) 写入

	15	14	13	12	11	10	9	8
C0IE	0	0	设置 CIE5	设置 CIE4	设置 CIE3	设置 CIE2	设置 CIE1	设置 CIE0
	7	6	5	4	3	2	1	0
	0	0	清除 CIE5	清除 CIE4	清除 CIE3	清除 CIE2	清除 CIE1	清除 CIE0

(a) 读取

CIE5 至 CIE0	CAN 模块中断使能位
0	禁止与中断状态寄存器 CINTSx 对应的中断输出。
1	允许与中断状态寄存器 CINTSx 对应的中断输出。

(b) 写入

设置 CIE5	清除 CIE5	CIE5 位的设置
0	1	CIE5 位清为 0。
1	0	CIE5 位设为 1。
其它		CIE5 位不改变。

设置 CIE4	清除 CIE4	CIE4 位的设置
0	1	CIE4 位清为 0。
1	0	CIE4 位设为 1。
其它		CIE4 位不改变。

设置 CIE3	清除 CIE3	CIE3 位的设置
0	1	CIE3 位清为 0。
1	0	CIE3 位设为 1。
其它		CIE3 位不改变。

设置 CIE2	清除 CIE2	CIE2 位的设置
0	1	CIE 位清为 0。
1	0	CIE2 位设为 1。
其它		CIE 位不改变。

设置 CIE1	清除 CIE1	CIE1 位的设置
0	1	CIE1 位清为 0。
1	0	CIE1 位设为 1。
其它		CIE1 位不改变。

设置 CIE0	清除 CIE0	CIE0 位的设置
0	1	CIE0 位清为 0。
1	0	CIE0 位设为 1。
其它		CIE0 位不改变。

(11) CAN0 模块中断状态寄存器 (C0INTS)

C0INTS 寄存器表示 CAN 模块的中断状态。

复位后: 0000H R/W 地址: 03FEC058H

(a) 读取

	15	14	13	12	11	10	9	8
C0INTS	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
	0	0	CINTS5	CINTS4	CINTS3	CINTS2	CINTS1	CINTS0

(b) 写入

	15	14	13	12	11	10	9	8
C0INTS	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
	0	0	清除 CINTS5	清除 CINTS4	清除 CINTS3	清除 CINTS2	清除 CINTS1	清除 CINTS0

(a) 读取

CINTS5 至 CINTS0	CAN 中断状态位
0	未产生中断源相关事件。
1	产生中断源相关事件。

中断状态位	相关中断源事件
CINTS5	从 CAN 睡眠模式*唤醒中断
CINTS4	仲裁失败中断
CINTS3	CAN 协议错误中断
CINTS2	CAN 错误状态中断
CINTS1	当至缓存器 m 的有效报文帧接收完成时中断
CINTS0	来自报文缓存器 m 的报文帧发送完成时中断

注 仅当通过 CAN 总线操作将 CAN 模块从 CAN 睡眠模式唤醒时将 CINTS5 位设为 1。当 CAN 睡眠模式通过软件解除时不能设置 CINTS5 位。

(b) 写入

清除 CINTS5 至 CINTS0	CINTS5 至 CINTS0 位的设置
0	CINTS5 至 CINTS0 位不改变。
1	CINTS5 至 CINTS0 位清除为 0。

(12) CAN0 模块位速率预分频器寄存器（C0BRP）

C0BRP 寄存器用于选择 CAN 协议层基础时钟（f_{TQ}）。通信波特率设置到 CnBTR 寄存器。

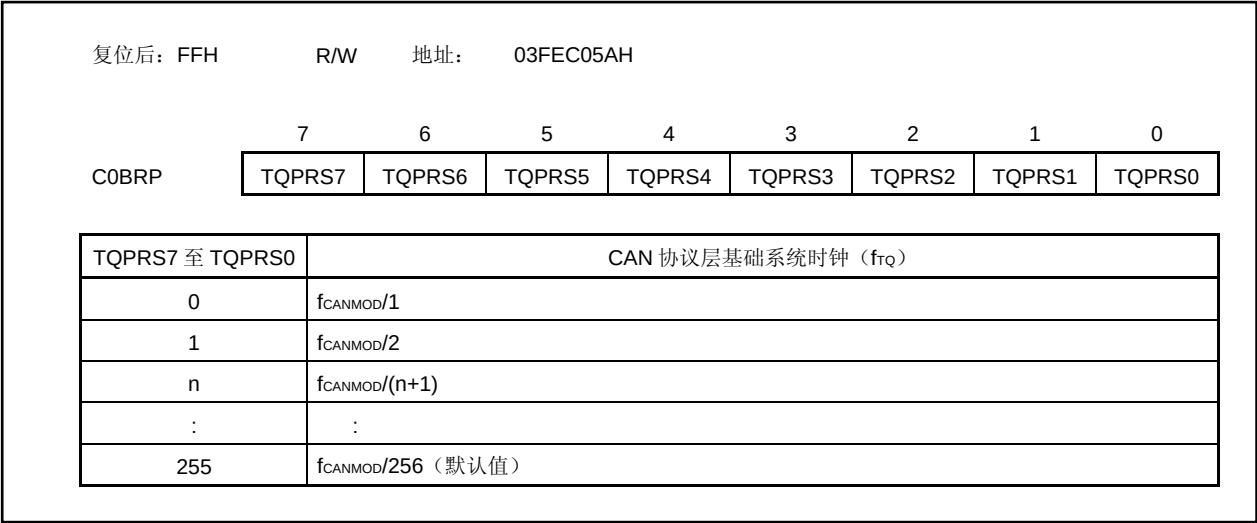
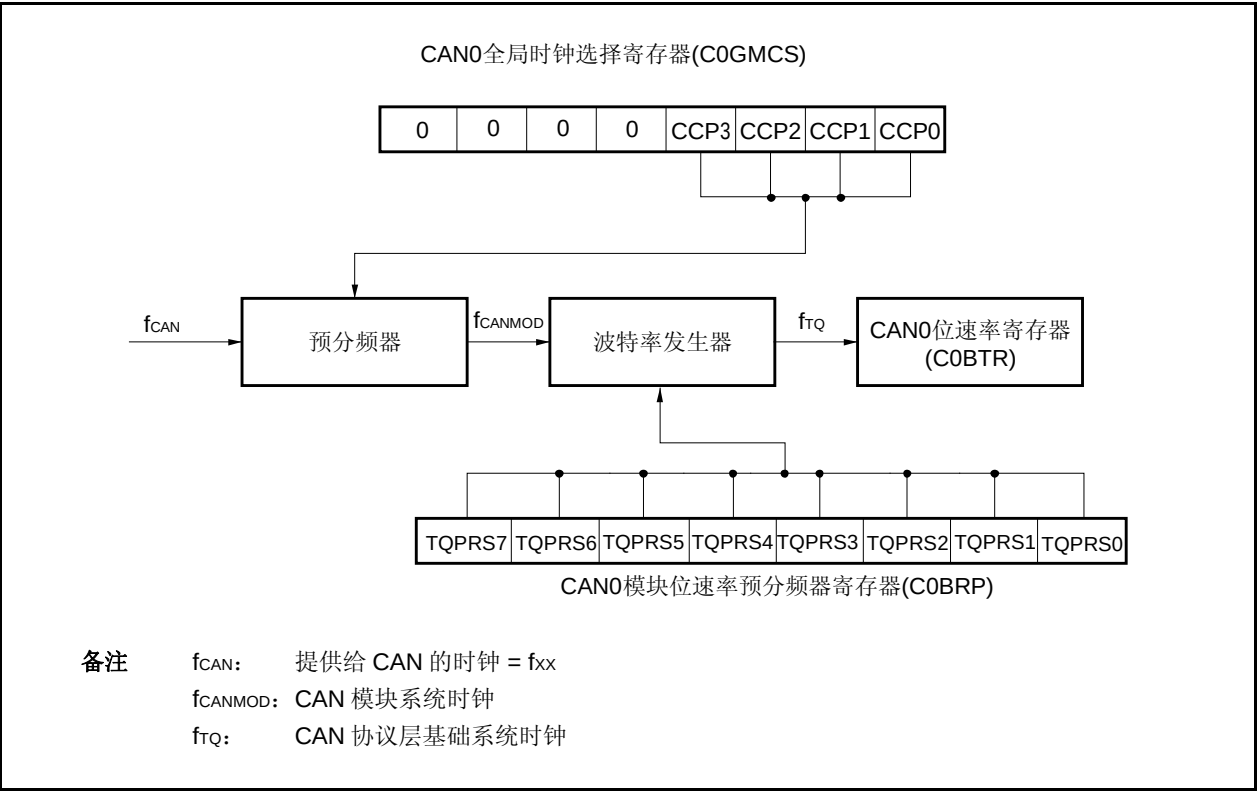


图 20-23. CAN 模块时钟

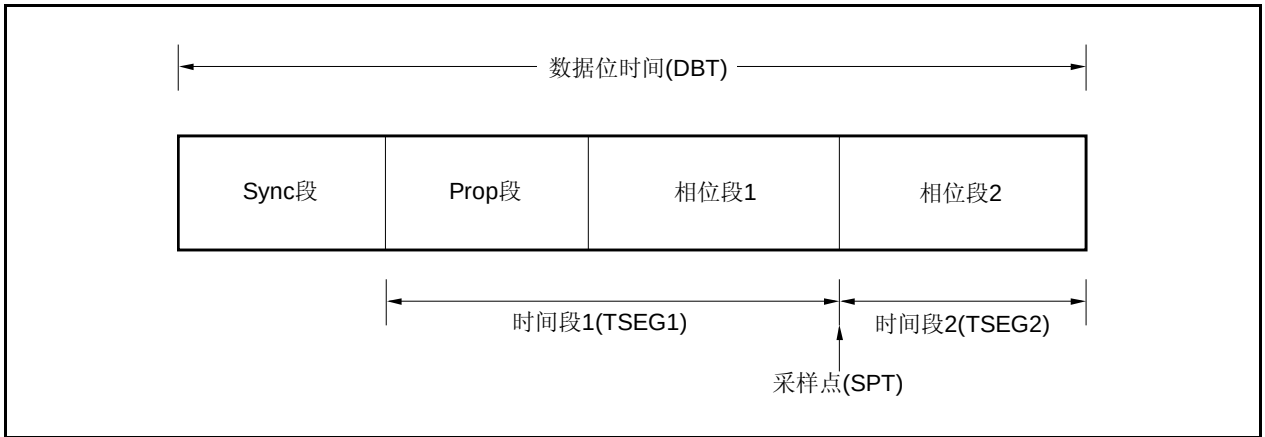


注意事项 C0BRP 寄存器仅在初始化模式下进行写入访问。

(13) CAN0 模块位速率寄存器 (C0BTR)

C0BTR 寄存器用于控制通信波特率的数据位时间。

图 20-24. 数据位时间



复位后: 370FH R/W 地址: 03FEC05CH

	15	14	13	12	11	10	9	8
C0BTR	0	0	SJW1	SJW0	0	TSEG22	TSEG21	TSEG20
	7	6	5	4	3	2	1	0
	0	0	0	0	TSEG13	TSEG12	TSEG11	TSEG10

SJW1	SJW0	同步跳转宽度的长度
0	0	1TQ
0	1	2TQ
1	0	3TQ
1	1	4TQ (默认值)

TSEG22	TSEG21	TSEG20	时间段 2 的长度
0	0	0	1TQ
0	0	1	2TQ
0	1	0	3TQ
0	1	1	4TQ
1	0	0	5TQ
1	0	1	6TQ
1	1	0	7TQ
1	1	1	8TQ (默认值)

TSEG13	TSEG12	TSEG11	TSEG10	时间段 1 的长度
0	0	0	0	禁止设置
0	0	0	1	2TQ [Ⓓ]
0	0	1	0	3TQ [Ⓓ]
0	0	1	1	4TQ
0	1	0	0	5TQ
0	1	0	1	6TQ
0	1	1	0	7TQ
0	1	1	1	8TQ
1	0	0	0	9TQ
1	0	0	1	10TQ
1	0	1	0	11TQ
1	0	1	1	12TQ
1	1	0	0	13TQ
1	1	0	1	14TQ
1	1	1	0	15TQ
1	1	1	1	16TQ (默认值)

注 C0BRP 寄存器= 00H 时务必要不要进行该设置。

备注 TQ = 1/frq (frq: CAN 协议层基础系统时钟)

(14) CAN0 模块最末入指针寄存器（C0LIPT）

C0LIPT 寄存器指出报文缓存器的编号，该缓存器最末存入一个数据帧或远程帧。

复位后：未定义

R

地址：03FEC05EH

7

6

5

4

3

2

1

0

C0LIPT

LIPT7

LIPT6

LIPT5

LIPT4

LIPT3

LIPT2

LIPT1

LIPT0

LIPT7 至 LIPT0	最末入指针寄存器（C0LIPT）
0 至 31	读取 C0LIPT 寄存器时，读取的是由接收记录列表的最末入指针（LIPT）指出的单元的内容项。因此，可以得到最末存储数据帧或远程帧的报文缓存器的编号。

备注

如果报文缓存器从未存储数据帧或远程帧，则 C0LIPT 寄存器的读取值会不确定。 若 CAN 模块从初始化模式变为操作模式之后将 C0RGPT.RHPM 位设为 1，那么 C0LIPT 寄存器的读取值不确定。

(15) CAN0 模块接收记录列表寄存器 (C0RGPT)

C0RGPT 寄存器用于读取接收记录列表。

复位后: xx02H R/W 地址: 03FEC060H

(a) 读取

	15	14	13	12	11	10	9	8
C0RGPT	RGPT7	RGPT6	RGPT5	RGPT4	RGPT3	RGPT2	RGPT1	RGPT0
	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	RHPM	ROVF

(b) 写入

	15	14	13	12	11	10	9	8
C0RGPT	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	清除 ROVF

(a) 读取

RGPT7 至 RGPT0	接收记录列表读取指针
0 至 31	读取 C0RGPT 寄存器时, 读取的是经接收记录列表的接收记录列表取指针 (RGPT) 指出的单元的内容项。因此, 可以得到最末存储数据帧或远程帧的报文缓存器的编号。

RHPM ^注	接收记录列表指针匹配
0	接收记录列表中至少有一个未被读取的报文缓存器编号。
1	接收记录列表没有未被读取的报文缓存器编号。

ROVF	接收记录列表溢出位
0	保留所有未读取的报文缓存器编号。 所有接收到新数据帧或远程帧的报文缓存器的编号都被记录到接收记录列表 (接收记录列表有一个空项)。
1	从上一次主处理器进行 RHL 服务 (例如读取 C0RGPT) 之后, 至少存储了 23 个入口。 前 22 个入口顺序存储而最后一个入口也许被新接收到的报文重写了多次, 这是因为当 ROVF 位设为 1 时, 所有缓存器的编号存储在 LIPT-1 处。结果不能按照它们接收的顺序完全恢复接收。

注 RHPM 位= 1 时, RGPT0 至 RGPT7 位的读取值无效。

(b) 写入

清除 ROVF	ROVF 位的设置
0	ROVF 位不改变。
1	ROVF 位清为 0。

(16) CAN0 模块最末出指针寄存器（C0LOPT）

C0LOPT 寄存器指出报文缓存器的编号，最末发送数据帧或远程帧发送到该报文缓存器。

复位后：未定义 R 地址： 03FEC062H

	7	6	5	4	3	2	1	0
C0LOPT	LOPT7	LOPT6	LOPT5	LOPT4	LOPT3	LOPT2	LOPT1	LOPT0

LOPT7 至 LOPT0	发送记录列表的最末出指针（LOPT）
0 至 31	读取 C0LOPT 寄存器时，读取的是由接收记录列表的最末出指针（LOPT）指出的单元的内容项。 因此，可以得到最末发送数据帧或远程帧的报文缓存器的编号。

备注 如果未曾从报文缓存器发送数据帧或远程帧，则从 C0LOPT 寄存器读取的值不确定。 若 CAN 模块从初始化模式变为操作模式之后将 C0TGPT.RHPM 位设为 1，那么 C0LOPT 寄存器的读取值不确定。

(17) CAN0 模块发送记录列表寄存器 (C0TGPT)

C0TGPT 寄存器用于读取发送记录列表。

复位后: xx02H R/W 地址: 03FEC064H

(a) 读取

	15	14	13	12	11	10	9	8
C0TGPT	TGPT7	TGPT6	TGPT5	TGPT4	TGPT3	TGPT2	TGPT1	TGPT0
	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	THPM	TOVF

(b) 写入

	15	14	13	12	11	10	9	8
C0TGPT	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	清除 TOVF

(a) 读取

TGPT7 至 TGPT0	发送记录列表读取指针
0 至 31	读取 C0TGPT 寄存器时, 读取的是由发送记录列表的读取指针 (TIPT) 指出的单元的内容项。因此, 可以得到最末发送数据帧或远程帧的报文缓冲器的编号。

THPM ^注	发送记录指针匹配
0	发送记录列表中至少有一个未被读取的报文缓冲器编号。
1	发送记录列表没有未被读取的报文缓冲器编号。

TOVF	发送记录列表溢出位
0	保留所有未读取的报文缓冲器编号。 所有发送新数据帧或远程帧的报文缓冲器的编号都被记录到接收记录列表 (发送记录列表有一个空项)。
1	从上一次主处理器进行 THL 服务 (例如读取 C0TGPT) 之后, 至少存储了 7 个入口。 前 6 个入口顺序存储而最后一个入口也许会在每次发送新的数据之前被重写, 这是因为当 TOVF 位设为 1 时, 所有缓冲器的编号存储在 LOPT-1 处。结果不能按照它们接收的顺序完全恢复接收。

注 THPM 位= 1 时, TGPT0 至 TGPT7 位的读取值无效。

备注 使用 ABT 的正常操作模式下, 由报文缓冲器 0 至 7 的发送不记录到发送记录列表。

(b) 写入

清除 TOVF	TOVF 位的设置
0	TOVF 位不改变。
1	TOVF 位清为 0。

(18) CAN0 模块时间标记寄存器 (C0TS)

C0TS 寄存器用于控制时间标记功能。

(1/2)

复位后: 0000H R/W 地址: 03FEC066H

(a) 读取

	15	14	13	12	11	10	9	8
C0TS	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
	0	0	0	0	0	TSLOCK	TSSEL	TSEN

(b) 写入

	15	14	13	12	11	10	9	8
C0TS	0	0	0	0	0	设置 TSLOCK	设置 TSSEL	设置 TSEN
	7	6	5	4	3	2	1	0
	0	0	0	0	0	清除 TSLOCK	清除 TSSEL	清除 TSEN

备注 在使用 ABT 的正常操作模式下，一定不要使用时间标志功能的上锁功能。

(a) 读取

TSLOCK	时间标记上锁功能使能位
0	停止时间标记上锁功能。 每发生一次所选择的时间标志捕获事件，TSOUT 信号来回切换一次。
1	使能时间标记上锁功能。 每发生一次所选择的时间标志捕获事件，TSOUT 信号来回切换一次。但是，当数据帧正确接收至报文缓存器 0 [※] 时，TSOUT 输出信号被锁住。

注 TSEN 位自动清除为 0。

TSSEL	时间标记捕获事件选择位
0	时间标记捕获事件为 SOF。
1	时间标记捕获事件为 EOF 的最后一位。

TSEN	TSOUT 操作设置位
0	禁止 TSOUT 切换操作。
1	允许 TSOUT 切换操作。

(b) 写入

设置 TSLOCK	清除 TSLOCK	TSLOCK 位的设置
0	1	TSLOCK 位清为 0。
1	0	TSLOCK 位设为 1。
其它		TSLOCK 位不改变。

设置 TSSEL	清除 TSSEL	TSSEL 位的设置
0	1	TSSEL 位清为 0。
1	0	TSSEL 位设为 1。
其它		TSSEL 位不改变。

设置 TSEN	清除 TSEN	TSEN 位的设置
0	1	TSEN 位清为 0。
1	0	TSEN 位设为 1。
其它		TSEN 位不改变。

(19) CAN0 报文数据字节寄存器 (C0MDATAxm, C0MDATAy) (x = 0 至 7, y = 01, 23, 45, 67)

C0MDATAxm 和 C0MDATAy 寄存器用于存储发送/接收报文的数据。

C0MDATAxm 寄存器可通过 C0MDATAy 寄存器以 16 位为单位进行存取。

(1/2)

复位后：未定义

R/W

地址：参见 表 20-16.

	15	14	13	12	11	10	9	8
C0MDATA01m	MDATA01	MDATA01	MDATA01	MDATA01	MDATA01	MDATA01	MDATA01	MDATA01
	15	14	13	12	11	10	9	8
	7	6	5	4	3	2	1	0
	MDATA01	MDATA01	MDATA01	MDATA01	MDATA01	MDATA01	MDATA01	MDATA01
	7	6	5	4	3	2	1	0
C0MDATA0m	7	6	5	4	3	2	1	0
	MDATA0	MDATA0	MDATA0	MDATA0	MDATA0	MDATA0	MDATA0	MDATA0
	7	6	5	4	3	2	1	0
C0MDATA1m	7	6	5	4	3	2	1	0
	MDATA1	MDATA1	MDATA1	MDATA1	MDATA1	MDATA1	MDATA1	MDATA1
	7	6	5	4	3	2	1	0
C0MDATA23m	15	14	13	12	11	10	9	8
	MDATA23	MDATA23	MDATA23	MDATA23	MDATA23	MDATA23	MDATA23	MDATA23
	15	14	13	12	11	10	9	8
	7	6	5	4	3	2	1	0
	MDATA23	MDATA23	MDATA23	MDATA23	MDATA23	MDATA23	MDATA23	MDATA23
	7	6	5	4	3	2	1	0
C0MDATA2m	7	6	5	4	3	2	1	0
	MDATA2	MDATA2	MDATA2	MDATA2	MDATA2	MDATA2	MDATA2	MDATA2
	7	6	5	4	3	2	1	0
C0MDATA3m	7	6	5	4	3	2	1	0
	MDATA3	MDATA3	MDATA3	MDATA3	MDATA3	MDATA3	MDATA3	MDATA3
	7	6	5	4	3	2	1	0

(2/2)

C0MDATA45m	15	14	13	12	11	10	9	8
	MDATA45 15	MDATA45 14	MDATA45 13	MDATA45 12	MDATA45 11	MDATA45 10	MDATA45 9	MDATA45 8
	7	6	5	4	3	2	1	0
	MDATA45 7	MDATA45 6	MDATA45 5	MDATA45 4	MDATA45 3	MDATA45 2	MDATA45 1	MDATA45 0
C0MDATA4m	7	6	5	4	3	2	1	0
	MDATA4 7	MDATA4 6	MDATA4 5	MDATA4 4	MDATA4 3	MDATA4 2	MDATA4 1	MDATA4 0
C0MDATA5m	7	6	5	4	3	2	1	0
	MDATA5 7	MDATA5 6	MDATA5 5	MDATA5 4	MDATA5 3	MDATA5 2	MDATA5 1	MDATA5 0
C0MDATA67m	15	14	13	12	11	10	9	8
	MDATA67 15	MDATA67 14	MDATA67 13	MDATA67 12	MDATA67 11	MDATA67 10	MDATA67 9	MDATA67 8
	7	6	5	4	3	2	1	0
	MDATA67 7	MDATA67 6	MDATA67 5	MDATA67 4	MDATA67 3	MDATA67 2	MDATA67 1	MDATA67 0
C0MDATA6m	7	6	5	4	3	2	1	0
	MDATA6 7	MDATA6 6	MDATA6 5	MDATA6 4	MDATA6 3	MDATA6 2	MDATA6 1	MDATA6 0
C0MDATA7m	7	6	5	4	3	2	1	0
	MDATA7 7	MDATA7 6	MDATA7 5	MDATA7 4	MDATA7 3	MDATA7 2	MDATA7 1	MDATA7 0

(20) CAN0 报文数据长度寄存器 m (C0MDLCm)

C0MDLCm 寄存器用于设置报文缓存器数据域的字节个数。

复位后: 0000xxxxB R/W 地址: 参见表 20-16

	7	6	5	4	3	2	1	0
C0MDLCm	0	0	0	0	MDLC3	MDLC2	MDLC1	MDLC0

MDLC3	MDLC2	MDLC1	MDLC0	发送/接收报文的数据长度
0	0	0	0	0 字节
0	0	0	1	1 字节
0	0	1	0	2 字节
0	0	1	1	3 字节
0	1	0	0	4 字节
0	1	0	1	5 字节
0	1	1	0	6 字节
0	1	1	1	7 字节
1	0	0	0	8 字节
1	0	0	1	禁止设置 (如果在发送期间设置这些位, 则当发送数据帧时不管 DLC 值的设置是什么都会发送 8-字节的数据。但是, 实际发送到 CAN 总线的值是设置到该寄存器的 DLC 值) ^注
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

注 实际发送至 CAN 总线的数据和 DLC 的值如下:

发送帧的类型	发送数据的长度	已发送的 DLC
数据帧	DLC 指定的字节个数 (但是如果 DLC ≥ 8, 则为 8 个字节)	MDLC3 至 MDLC0 位
远程帧	0 字节	

注意事项 1. 务必将 7 至 4 位设置为 0000B。

2. 接收数据存储在接收帧的 DLC 值对应的字节个数 (但是, 上限为 8) 一样多的 C0MDATAxm 寄存器中。没有存储数据的 C0MDATAxm 寄存器不确定。

(21) CAN0 报文结构寄存器 m (C0MCONFm)

C0MCONFm 寄存器用于指定报文缓存器的类型和设置屏蔽。

(1/2)

复位后：未定义

R/W

地址：参见 表 20-16

	7	6	5	4	3	2	1	0
C0MCONFm	OVS	RTR	MT2	MT1	MT0	0	0	MA0

OVS	重写控制位
0	已接收数据帧的报文缓存器 ^注 不会被新接收的数据帧重写。 新接收的数据被丢弃。
1	已接收数据帧的报文缓存器被新接收的数据帧重写。

注 “已经接收数据帧的报文缓存器”指其 C0MCTRLm.DN 位已经设为 1 的接收报文缓存器。

备注 不管 OVS 和 DN 位的设置如何，都可接收并存储远程帧。 满足其它条件（ID 匹配、RTR 位=0、C0MCTRLm.TRQ 位=0）的远程帧总是被接收并存储在相应的报文缓存器中（中断产生、DN 标志设置、C0MDLCm.MDLC0 至 C0MDLCm.MDLC3 位更新以及记录至记录列表）。

RTR	远程帧请求位 ^注
0	发送一个数据帧。
1	发送一个远程帧。

注 RTR 位指定由报文缓存器定义为发送报文缓存器的报文帧的类型。

即使接收到有效的远程帧，仍将已接收到帧的发送报文缓存器的 RTR 位清为 0。

即使已经从 CAN 总线接收到 ID 匹配的远程帧，且利用发送报文缓存器的 RTR 位设为 1 来发送远程帧，也不能接收或存储该远程帧（中断产生、DN 标志设置、MDLC0 至 MDLC3 位更新以及记录至接收记录列表）。

MT2	MT1	MT0	报文缓存器类型设置位
0	0	0	发送报文缓存器
0	0	1	接收报文缓存器（没有屏蔽）
0	1	0	接收报文缓存器（屏蔽 1 设置）
0	1	1	接收报文缓存器（屏蔽 2 设置）
1	0	0	接收报文缓存器（屏蔽 3 设置）
1	0	1	接收报文缓存器（屏蔽 4 设置）
其它			禁止设置

(2/2)

MA0	报文缓存器分配位
0	不使用报文缓存器。
1	使用报文缓存器。

注意事项 务必将 0 写入位 2 和 1。

(22) CAN0 报文 ID 寄存器 m (C0MIDLm, C0MIDHm)

C0MIDLm 和 C0MIDHm 寄存器用于设置标识符 (ID)。

复位后: 不确定

R/W

地址: 参见 表 20-16

	15	14	13	12	11	10	9	8
C0MIDLm	ID15	ID14	ID13	ID12	ID11	ID10	ID9	ID8
	7	6	5	4	3	2	1	0
	ID7	ID6	ID5	ID4	ID3	ID2	ID1	ID0

	15	14	13	12	11	10	9	8
C0MIDHm	IDE	0	0	ID28	ID27	ID26	ID25	ID24
	7	6	5	4	3	2	1	0
	ID23	ID22	ID21	ID20	ID19	ID18	ID17	ID16

IDE	格式模式指定位
0	标准格式模式 (ID28 至 ID18: 11 位) [※]
1	扩展格式模式 (ID28 至 ID0: 29 位)

注 不使用 ID17 至 ID0 位。

ID28 至 ID0	报文 ID
ID28 至 ID18	11 位标准 ID 值 (IDE = 0 时)
ID28 至 ID0	29 位扩展 ID 值 (IDE = 1 时)

注意事项 务必将 0 写入 C0MIDHm 寄存器的位 14 和 13。

(23) CAN0 报文控制寄存器 m (C0MCTRLm)

C0MCTRLm 寄存器用于控制报文缓存器的操作。

(1/2)

复位后： 00x000000 R/W 地址：参见 See 表 20-16
000xx000B

(a) 读取

	15	14	13	12	11	10	9	8
C0MCTRLm	0	0	MUC	0	0	0	0	0
	7	6	5	4	3	2	1	0
	0	0	0	MOW	IE	DN	TRQ	RDY

(b) 写入

	15	14	13	12	11	10	9	8
C0MCTRLm	0	0	0	0	设置 IE	0	设置 TRQ	设置 RDY
	7	6	5	4	3	2	1	0
	0	0	0	清除 MOW	清除 IE	清除 DN	清除 TRQ	清除 RDY

(a) 读取

MUC ^注	表示报文缓存器正在更新的位
0	CAN 模块没有更新报文缓存器（接收并存储）。
1	CAN 模块正在更新报文缓存器（接收并存储）。

注 直到执行第一次接收和存储之前，MUC 位不确定。

MOW	报文缓存器重写状态位
0	报文缓存器未被新接收的数据帧重写。
1	报文缓存器被新接收的数据帧重写。

备注 即使在报文寄存器的 DN 位=1 时接收并存储远程帧，也不会将 MOW 位设为 1。

IE	报文缓存器中断请求使能位
0	接收报文缓存器： 禁止有效报文接收完成中断。 发送报文缓存器： 禁止正常报文发送完成中断。
1	接收报文缓存器： 使能有效报文接收完成中断。 发送报文缓存器： 允许正常报文发送完成中断。

DN	报文缓存器数据更新位
0	数据帧或远程帧不存放在报文缓存器。
1	数据帧或远程帧存放在报文缓存器。

TRQ	报文缓存器发送请求位
0	在报文缓存器中没有待决或正在发送的报文帧发送请求。
1	报文缓存器没有保持待决报文帧的发送或正在发送报文帧。

注意事项 不要同时设置 TRQ 位和 RDY 位。 务必在 TRQ 位设为 1 之前设置 RDY 位。

RDY	报文缓存器就绪位
0	报文缓存器可通过软件写入。 CAN 模块不能写入报文缓存器。
1	忽略通过软件写入报文缓存器（除了对 RDY、TRQ、DN 和 MOW 位的写入访问）。 CAN 模块可以写入报文缓存器。

注意事项 报文发送期间不要清除 RDY 位（0）。 为了清除 RDY 位以重新定义，应遵循发送异常中止的步骤。

(b) 写入

清除 MOW	MOW 位的设置
0	MOW 位不改变。
1	MOW 位清为 0。

设置 IE	清除 IE	IE 位的设置
0	1	IE 位清为 0。
1	0	IE 位设为 1。
其它		IE 位不改变。

清除 DN	DN 位的设置
1	DN 位清为 0。
0	DN 位不改变。

注意事项 切勿用软件将 DN 位设置为 1。 确保将位 10 设置为 0。

设置 TRQ	清除 TRQ	TRQ 位的设置
0	1	TRQ 位清为 0。
1	0	TRQ 位设为 1。
其它		TRQ 位不改变。

设置 RDY	清除 RDY	RDY 位的设置
0	1	RDY 位清为 0。
1	0	RDY 位设为 1。
其它		RDY 位不改变。

20.7 位设置/清除功能

CAN 控制寄存器包括可通过 CPU 和 CAN 接口进行位设置或位清除的寄存器。如果直接写入以下寄存器会产生操作错误。不要通过位操作、读取/修改/写入或目标值直接写入来写入任何值。

- CAN0 全局控制寄存器 (C0GMCTRL)
- CAN0 全局自动块传输控制寄存器 (C0GMABT)
- CAN0 模块控制寄存器 (C0CTRL)
- CAN0 模块中断使能寄存器 (C0IE)
- CAN0 模块中断状态寄存器 (C0INTS)
- CAN0 模块接收记录列表寄存器 (C0RGPT)
- CAN0 模块发送记录列表寄存器 (C0TGPT)
- CAN0 模块时间标记寄存器 (C0TS)
- CAN0 报文控制寄存器 (C0MCTRLm)

备注 m = 00 至 31

以上寄存器的 16 位都可以通过一般的方法读取。使用下图 20-25 说明的步骤设置或清除这些寄存器的低 8 位。

以上寄存器低 8 位的设置或清除与高 8 位结合进行（参见图 20-26 中设置/清除操作后的位状态）。图 20-25 显示了相关寄存器中与设置/清除/无改变操作有关的设置位或清除位的值。

图 20-25. 位设置/清除操作举例

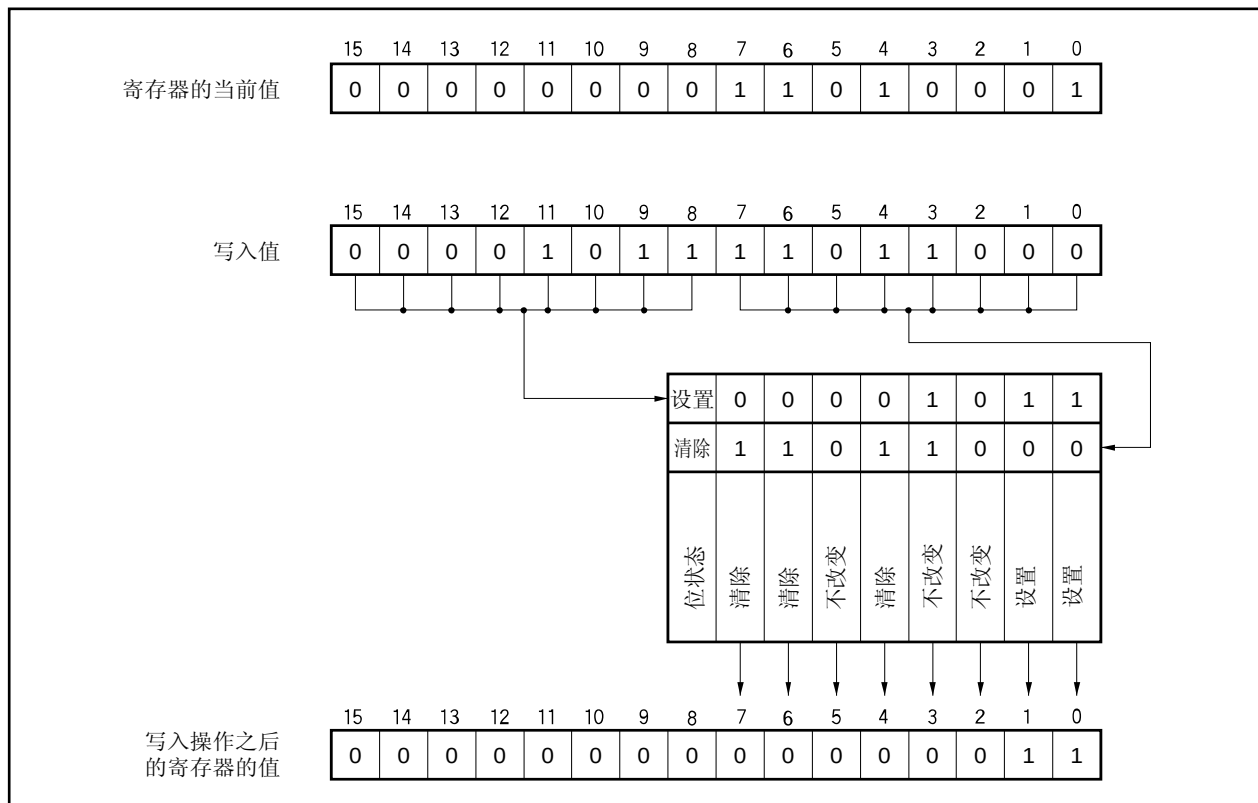


图 20-26. 位设置/清除操作后位状态举例

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
设置 7	设置 6	设置 5	设置 4	设置 3	设置 2	设置 1	设置 0	清除 7	清除 6	清除 5	清除 4	清除 3	清除 2	清除 1	清除 0

设置 n	清除 n	位设置/清除操作之后位 n 的状态
0	0	不改变
0	1	0
1	0	1
1	1	不改变

备注

n = 0 至 7

20.8 CAN控制器初始化

20.8.1 CAN模块的初始化

允许 CAN 模块操作之前，需要通过软件设置 C0GMCS.CCP0 至 C0GMCS.CCP3 位来决定 CAN 模块系统时钟。CAN 模块操作使能之后不要改变 CAN 模块系统时钟的设置。

通过设置 C0GMCTRL.GOM 位使能 CAN 模块。关于 CAN 模块的初始化步骤，参见 20.16 CAN 控制器的操作。

20.8.2 报文缓存器的初始化

CAN 模块使能后，报文缓存器包含不确定的值。在 CAN 模块由初始化模式转换至某一种操作模式之前，必须将所有的报文缓存器初始化。对应用中不使用的缓存器进行以下设置。

- 将 C0MCTRLm.RDY、C0MCTRLm.TRQ 和 C0MCTRLm.DN 位清为 0。
- 将 C0MCONFM.MA0 位清为 0。

备注 m = 00 至 31

20.8.3 报文缓存器的重新定义

重写定义报文缓存器意味着在接收或发送报文时改变报文缓存器的 ID 和控制信息，而不影响其它的发送/接收操作。

(1) 在初始化模式下重新定义报文缓存器

将 CAN 模块置于初始化模式，然后在初始化模式中改变报文缓存器的 ID 和控制信息。改变了 ID 和控制信息后，将 CAN 模块设为操作模式。

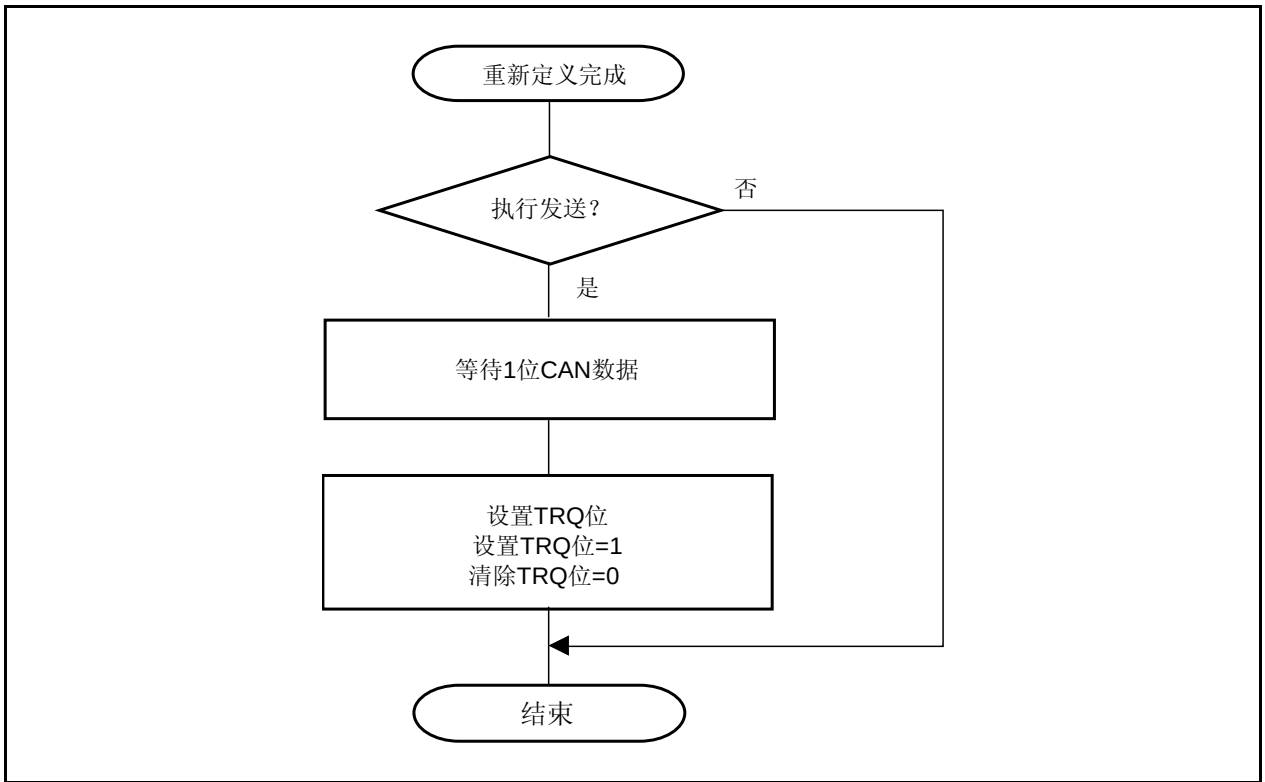
(2) 接收期间重新定义报文缓存器

如图 20-38 中所示进行重新定义。

(3) 发送期间重新定义报文缓存器

为了重写已经设置了发送请求的发送报文缓存器的内容，执行发送中止处理（参见 20.10.4 (1) 利用自动块传输（ABT）的正常操作模式之外的发送中止处理，20.10.4 (2) 利用自动块传输（ABT）的正常操作模式下除了 ABT 发送之外的发送中止处理以及 20.10.4 (3) 使用自动块传输（ABT）的正常操作模式中的发送中止）。确定已经中止或完成了发送，然后重新定义报文缓存器。重新定义发送报文缓存器之后，用如下所述的步骤设置发送请求。然而，当向一个未中止当前发送就重新定义的报文缓存器设置发送请求时，不需要 1-位等待时间。

图 20-27. 报文缓存器重新定义之后发送请求（TRQ）的设置



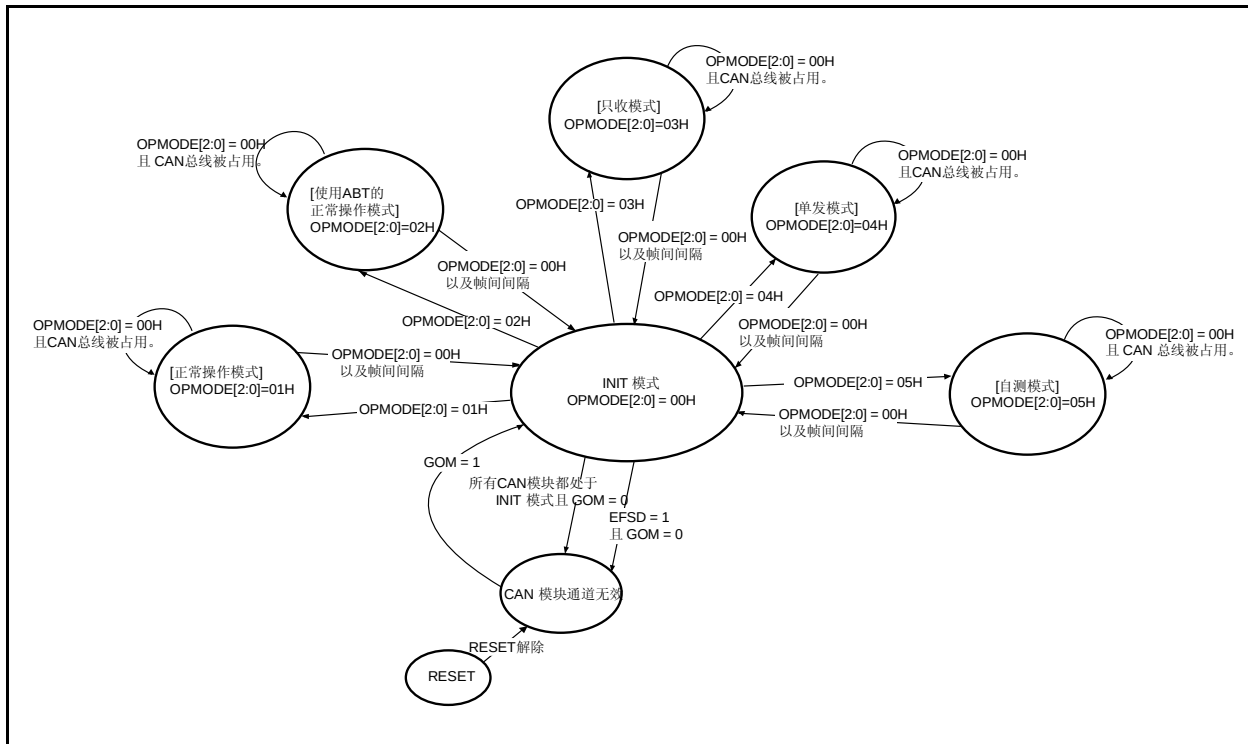
- 注意事项**
1. 接收报文时，按照 ID 和每一个接收缓存器的屏蔽设置进行接收过滤。如果不遵守图 20-39 中的步骤，则已重新定义之后的报文缓存器的内容可能会同接收结果（接收过滤后的结果）相矛盾。若发生这种情况，则检查报文缓存器重新定义后首先接收并存储的 ID 和 IDE 是否为报文缓存器已重新定义之后的 ID 和 IE。如果重新定义之后没有存储 ID 和 IDE，则再次重新定义报文缓存器。
 2. 发送报文时，按照由设定了发送请求的发送报文缓存器设置的 ID，IDE 和 RTR 位检查发送优先级。选择具有最高优先级的发送报文缓存器进行发送。如果不遵守图 20-27 中的步骤，那么重新定义之后可能会发送一个不具有最高优先级 ID 的报文。

20.8.4 从初始化模式到操作模式的转换

CAN 模块可转换到下列操作模式。

- 正常操作模式
- 使用 ABT 的正常操作模式
- 只收模式
- 单次模式
- 自检模式

图 20-28. 至操作模式的转换



C0CTRL.OPMODE2 至 C0CTRL.OPMODE0 位控制从初始化模式到操作模式的转换。

将某一操作模式变为另一操作模式时，需要中间切换为初始化模式。不要直接将操作模式转换为另一操作模式，否则操作得不到保证。

当 CAN 总线不在帧间隙（即：正在进行帧接收或帧发送）时，从操作模式到初始化模式的请求被挂起待决，且 CAN 模块在帧间隙的第 1 位进入初始化模式（OPMODE2 至 OPMODE0 位的值变为 00H）。产生将模式变为初始化模式的请求后，在 OPMODE2 至 OPMODE0 位的值变为 000B 后读取它们的值以确认模块已进入初始化模式（参见图 20-36）。

20.8.5 复位 CAN 模块的错误计数器 C0ERC

重新初始化或强制从掉线状态恢复时，如果有必要复位 C0ERC 和 C0INFO 寄存器，那么在初始化模式中将 C0CTRL.CCERC 位设为 1。当该位设为 1 时，将 C0ERC 和 C0INFO 寄存器清除为它们的默认值。

20.9 报文接收

20.9.1 报文接收

为了存储新接收的报文，在所有操作模式下，在全部报文缓存器区域搜索所有满足下列条件的缓存器。

- 用作报文缓存器
(COMCONFm.MA0 位设为 1。)
- 设置为接收报文缓存器
(COMCONFm.MT2 至 COMCONFm.MT0 位设为 001B, 010B, 011B, 100B,或 101B。)
- 准备接收
(COMCTRLm.RDY 位设为 1。)

备注 m = 00 至 31

当 CAN 模块的两个或更多报文缓存器接收同一个报文时，该报文按照下面说明的优先级存储。通常报文不存储在低优先级的报文缓存器中，而存储在具有最高优先级的报文缓存器中。例如，当非屏蔽接收报文缓存器和连接至屏蔽 1 的接收报文缓存器具有相同的 ID 时，即使非屏蔽接收报文缓存器已接收到报文，接收报文也不会存储在没有接收到报文的与屏蔽 1 相连的报文缓存器。换言之，当设置具有不同优先级的两个或更多的报文缓存器为接收一个报文的状态时，报文总是存储在具有最高优先级的报文缓存器中；而不存储在具有较低优先级的报文缓存器中。这同样适用于具有最高优先级的报文缓存器不能接收并存储报文的情况（例如，DN 位=1 时表明已经接收了报文，但是由于 OWS 位=0 而禁止重写的情况）。这种情况下，实际上报文既不被接收并存储在具有最高优先级的候选缓存器中，也不存储在具有较低优先级的缓存器中。

优先级	设置相同 ID 时的存储状态	
1 (高)	非屏蔽报文缓存器	DN 位 = 0
		DN 位 = 1 且 OWS 位 = 1
2	连接至屏蔽 1 的报文缓存器	DN 位 = 0
		DN 位 = 1 且 OWS 位 = 1
3	连接至屏蔽 2 的报文缓存器	DN 位 = 0
		DN 位 = 1 且 OWS 位 = 1
4	连接至屏蔽 3 的报文缓存器	DN 位 = 0
		DN 位 = 1 且 OWS 位 = 1
5 (低)	连接至屏蔽 4 的报文缓存器	DN 位 = 0
		DN 位 = 1 且 OWS 位 = 1

20.9.2 接收记录列表功能

接收记录列表（RHL）功能在接收记录列表中记录接收并存储各个数据帧或远程帧的接收报文缓存器的编号。RHL 的组成有：最多等于 23 个报文的存储单元、对应 C0LIPT 寄存器的最末报文入指针（LIPT）以及有对应 C0RGPT 寄存器的接收记录列表取指针（RGPT）。

CAN 模块从初始化模式转换到某一种操作模式后的瞬间，RHL 不确定。

C0LIPT 寄存器保持由 LIPT 指针减 1 的值所指示的 RHL 的条目。因此，通过读取 C0LIPT 寄存器，可检查接收并存储数据帧或远程帧的报文缓存器的编号。LIPT 指针用作写入指针，指示报文缓存器的编号被记录到 RHL 的哪一部分。任何时候接收并存储数据帧或远程帧，都将相应的报文缓存器编号记录到由 LIPT 指针指示的 RHL 单元。每次完成到 RHL 的记录时，LIPT 指针自动加 1。如此一来，接收并存储帧的报文缓存器的编号将按照时间顺序进行记录。

RGPT 指针用作读取指针，从 RHL 读取已记录的报文缓存器的编号。该指针表示 CPU 还没有读取的 RHL 的起始单元。通过使用软件读取 C0RGPT 寄存器，可读取到已接收并存储数据帧或远程帧的报文缓存器的编号。每次从 C0TGPT 寄存器读取报文缓存器的编号之后，RGPT 指针自动加 1。

如果 RGPT 指针的值与 LIPT 指针的值匹配，则将 C0RGPT.RHPM 位（接收记录列表指针重合）设为 1。这表示在 RHL 中不存在尚未读取的报文缓存器的编号。如果记录了新的报文缓存器编号，LIPT 指针自动加 1，而由于它的值与 RGPT 的值不再相符，因而 RHPM 位被清除。换言之，在 RHL 中存在未读取的报文缓存器的编号。

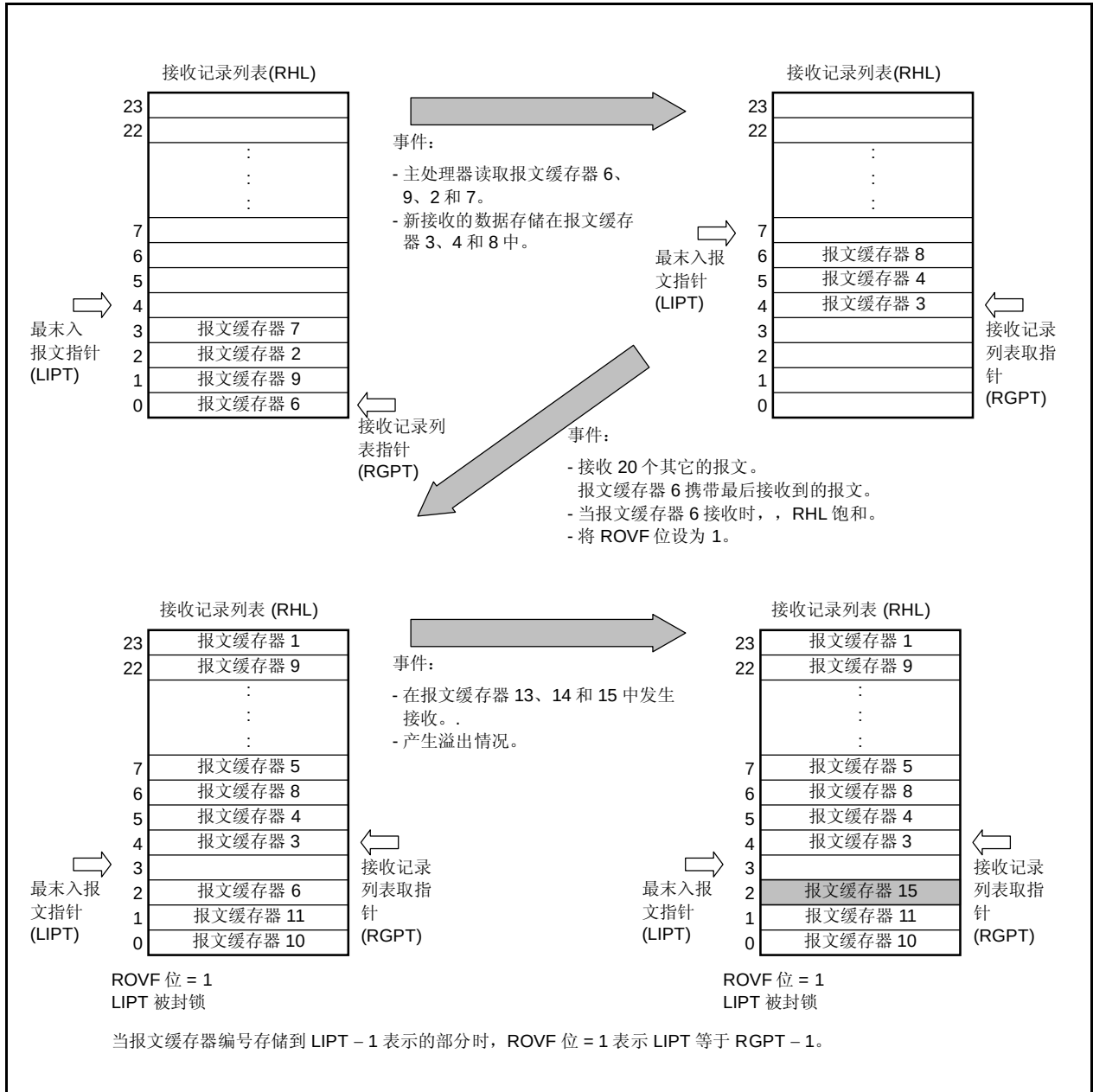
如果 LIPT 指针加 1 并与 RGPT 指针减 1 的值匹配，则将 C0RGPT.ROVF 位（接收记录列表溢出）设为 1。这表示 RHL 中全是未被读取的报文缓存器的编号。发生更多的接收和存储时，最末记录的报文缓存器编号被接收并存储了新报文的报文缓存器编号覆盖。ROVF 位置位（1）之后，RHL 中记录的报文缓存器的编号不能完全反映按时间排列的顺序。

但是接收报文本本身会正确存储。通过执行 CPU 指令查找每个报文缓存器的 DN 位可以指定接收并存储报文的缓存器的编号。

在接收记录列表包含 23 个或更少的入口时，会保持其产生顺序。但是，如果在主处理器没有读取接收记录列表的情况下发生更多的接收，则不能恢复完整的接收序列。

备注 m = 00 至 31

图 20-29. 发送记录列表



20.9.3 屏蔽功能

对于各接收报文缓存器，可定义是否将其连接标识符的屏蔽。

通过使用屏蔽功能，可比较从 CAN 总线接收的报文的标识符和预先设到报文中的标识符。 不管已屏蔽 ID 设为 1 还是 0，接收报文都可存储在既定的报文缓存器中。

当屏蔽功能有效时，不将由接收报文中的有屏蔽定义为 1 的标识符位和报文缓存器中相应的标识符位进行比较。

但是，任何通过屏蔽将值定义为 0 的位都要进行比较。

例如，我们假设有报文将要存储在报文缓存器 14 中，且所有报文都具有标准格式的 ID，其位 ID27 至 ID25 为 0 而位 ID24 和 ID22 为 1。该例的步骤如下所示。

<1> 将存储在报文缓存器中的标识符

ID28	ID27	ID26	ID25	ID24	ID23	ID22	ID21	ID20	ID19	ID18
x	0	0	0	1	x	1	x	x	x	x

备注 忽略

<2> 报文缓存器 14 中配置的标识符（举例）
（使用 COMIDL14 和 COMIDH14 寄存器）

ID28	ID27	ID26	ID25	ID24	ID23	ID22	ID21	ID20	ID19	ID18
x	0	0	0	1	x	1	x	x	x	x
ID17	ID16	ID15	ID14	ID13	ID12	ID11	ID10	ID9	ID8	ID7
x	x	x	x	x	x	x	x	x	x	x
ID6	ID5	ID4	ID3	ID2	ID1	ID0				
x	x	x	x	x	x	x				

将 ID27 至 ID25 位清为 0 且 ID24 和 ID22 位设为 1 的 ID 记录（初始化）至报文缓存器 14。

备注 忽略

备注 报文缓存器 14 设置为连接至屏蔽 1 的标准格式标识符（COMCONF14.MT2 至 COMCONF14.MT0 位设为 010B）。

<3> CAN 模块 0 的屏蔽设置（屏蔽 1）（举例）
（使用 CAN0 模块屏蔽 1 寄存器 L 和 H（C0MASKL1 和 C0MASKH1））

CMID28	CMID27	CMID26	CMID25	CMID24	CMID23	CMID22	CMID21	CMID20	CMID19	CMID18
1	0	0	0	0	1	0	1	1	1	1
CMID17	CMID16	CMID15	CMID14	CMID13	CMID12	CMID11	CMID10	CMID9	CMID8	CMID7
1	1	1	1	1	1	1	1	1	1	1
CMID6	CMID5	CMID4	CMID3	CMID2	CMID1	CMID0				
1	1	1	1	1	1	1				

1: 未比较（屏蔽）
0: 已比较

CMID27 至 CMID24 以及 CMID22 位清为 0，且 CMID28、CMID23 以及 CMID21 至 CMID0 位设为 1。

20.9.4 多缓存接收块功能

多缓存器接收 (MBRB) 功能用于不和 CPU 交互时, 通过将相同的 ID 设置给有相同类型的两个或更多的报文缓存器, 在两个或更多的报文缓存器中连续存储数据块。

例如, 假设将相同的报文缓存器类型设置至 10 个报文缓存器: 报文缓存器 10 至 19, 且给各个报文缓存器设置相同的 ID。如果接收到其 ID 与报文缓存器 ID 匹配的第 1 个报文, 则将它存储在报文缓存器 10。此刻, 报文缓存器 10 的 DN 位置位, 以禁止重写报文缓存器。

当接收到带有匹配 ID 的下一个报文时, 将接收并存入报文缓存器 11, 每收到一次 ID 匹配的报文, 都将它依次 (按照升序) 存储于报文缓存器 12, 13 等等。即使接收由多个报文组成的数据块时, 也可在不重写先前接收的 ID 匹配的数据而存储和接收报文。

通过设置各报文缓存器的 C0MCTRLm.IE 位, 可检查数据块是否已被接收并存储。例如, 如果数据块由 k 个报文, k 个已初始化的用于接收数据块的报文缓存器组成。将报文缓存器 0 至 (k-2) 中的 IE 位清为 0 (中断禁止), 而将报文缓存器 k-1 的 IE 位设为 1 (中断使能)。这种情况下, 当报文缓存器 k-1 接收并存储报文时产生接收完成中断, 表示 MBRB 已满。通过清除报文缓存器 0 至 (k-3) 的 IE 位或设置报文缓存器 k-2 的 IE 位, 二者选一, 可产生 MBRB 将要溢出的警告。

每一个 MBRB 报文缓存器中存储接收数据的基本条件与单个报文缓存器中的数据存储条件相同。

- 注意事项**
1. 可给相同类型的各报文缓存器配置 MBRB。因此, 即使另外的 MBRB 的一个报文缓存器中有空位, 具有空位的报文缓存器的 ID 匹配但报文缓存器的类型不同, 那么接收数据也不存储在那个报文缓存器中, 而是丢弃。
 2. MBRB 不具有环形缓存器结构。所以, MBRB 结构中具有最高编号的报文缓存器接收报文之后, 新接收的报文将不再存储在由最低报文缓存器编号起排序的报文缓存器中。
 3. MBRB 基于接收和存储条件工作; MBRB 没有诸如功能使能位等专用设置。通过给两个或更多的报文缓存器设置相同的类型和 ID, 可自动配置 MBRB。
 4. 关于 MBRB, “ID 匹配”指“屏蔽之后的 ID 匹配”。即使给各个报文缓存器设定的 ID 不同, 如果被屏蔽寄存器屏蔽的 ID 匹配, 则认为匹配 ID 和具有该 ID 的缓存器为报文的存储目的。
 5. 各 MBRB 中的优先级与 20.9.1 报文接收中显示的优先级一致。

备注 m = 00 至 31

20.9.5 远程帧接收

接收远程帧时，在所有操作模式下，在所有满足下列条件的缓存器中查找用于存储远程帧的报文缓存器。

- 用作报文缓存器
(COMCONFm.MA0 位设为 1。)
- 设置为发送报文缓存器
(COMCONFm.MT2 至 COMCONFm.MT0 位设为 000B)
- 准备接收
(COMCTRLm.RDY 位设为 1。)
- 设置至发送报文
(COMCONFm.RTR 位清为 0。)
- 没有设置发送请求
(COMCTRLm.TRQ 位设为 1。)

当接收远程帧时，如果接收远程帧的 ID 与满足以上条件的报文缓存器的 ID 匹配，则执行以下的接收和存储操作。

- C0MDLCm.DLC3 至 C0MDLCm.DLC0 位存储接收的 DLC 值。
- 数据区域的 C0MDATA0m 至 C0MDATA7m 寄存器不更新（报文接收之前的数据）。
- (COMCTRLm.DN 位设为 1。)
- C0INTS.CINTS1 位设为 1（如果接收并存储帧的报文缓存器的 COMCTRLm.IE 位设为 1）。
- 输出接收完成中断（INTC0REC）（如果将接收并存储帧的报文缓存器的 IE 位设为 1 且将 C0IE.CIE1 位设为 1）。
- 将报文缓存器编号记录在接收记录列表中。

注意事项 当查找报文缓存器用于接收并存储远程帧时，不影响由报文缓存器的 COMCONFm.OWS 位控制的重写与 DN 位。忽略 OWS 位的设置并在每种情况下将 DN 位设为 1。

如果不止一个的发送报文缓存器具有相同 ID，且接收远程帧的 ID 与此 ID 匹配，则将该远程帧存储在具有最低报文缓存器编号的发送报文缓存器中。

备注 m = 00 至 31

20.10 报文发送

20.10.1 报文发送

在所有操作模式中，如果将满足以下条件的报文缓存器的 COMCTRLm.TRQ 位设为 1，那么就查找将要发送报文的报文缓存器。

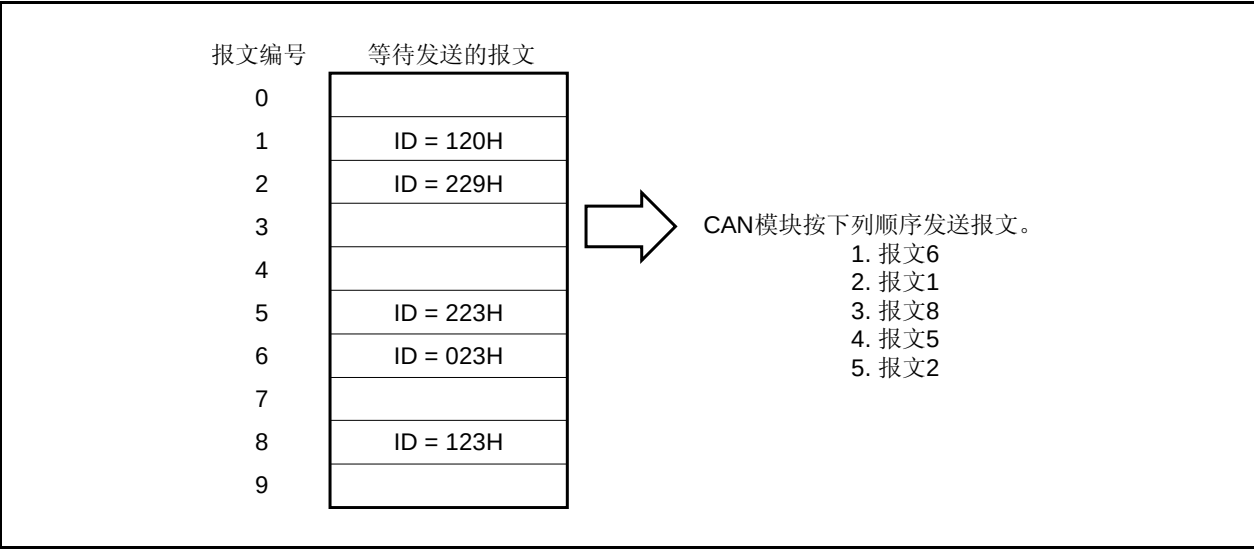
- 用作报文缓存器
(COMCONFm.MA0 位设为 1。)
- 设置为发送报文缓存器
(COMCONFm.MT2 至 COMCONFm.MT0 位设为 000B)
- 准备发送
(COMCTRLm.RDY 位设为 1。)

备注 m = 00 至 31

CAN 系统是多主机通信系统。像这样的系统中，报文发送的优先级基于报文标识符 (IDs) 来决定。 在有多个报文等候发送时，为了便于使用软件进行发送处理，CAN 模块使用硬件检查具有最高优先级的报文的 ID 并自动识别此报文。 这样就不再需要基于软件的优先级控制。

发送优先级由标识符 (ID) 控制。

图 20-30. 报文处理举例



查找发送报文之后，发送其请求被挂起的发送报文缓存器（预先将其 TRQ 位设为 1 的报文缓存器）中优先级最高的发送报文。

如果出现新的发送请求，则将具有新发送请求的发送报文缓存器与请求被挂起的发送报文缓存器进行比较。 如果新发送请求有较高的优先级，则将其发送，除非是已开始了具有低优先级的报文发送。 但是，如果低优先级的报文已经开始发送，则新的发送请求会稍后发送。 依照下面的规则决定最高的优先级。

优先级	条件	说明
1（高）	ID 的前 11 位的值 [ID28 至 ID18]:	首先发送用 ID 的前 11 位表示的具有最低值的报文帧。 如果 11 位标准 ID 的值小于或等于 29-位扩展 ID 的前 11 位的值，11 位标准 ID 相对于具有 29-位 ID 的报文，有较高的优先级。
2	帧类型	相对于具有标准 ID 的远程帧和扩展 ID 的报文帧， 11 位标准 ID 的数据帧（C0MCONFm.RTR 位清为 0）有较高的优先级。
3	ID 类型	标准 ID 的报文帧（C0MIDHm.IDE 位清为 0）相对于扩展 ID 的报文帧有较高的优先级。
4	ID 的低 18 位的值 [ID1 至 ID0]:	如果两个或更多发送被挂起的扩展 ID 报文帧，其 ID 的前 11 位值相等，并具有相同的帧类型（等于 RTR 位的值），那么首先发送其扩展 ID 前 18 位值最低的报文帧。
5（低）	报文缓存器编号	如果两个或更多的报文缓存器请求发送具有相同 ID 的报文，那么首先发送来自具有最低报文缓存器编号的报文缓存器中的报文。

备注 如果在使用 ABT 的正常操作模式中将自动块传输请求位（C0GMABT.ABTTRG 位）设为 1，则仅将 ABT 报文缓存器组中有一个报文缓存器的 TRQ 位设为 1。

如果 ABT 模式由 ABTTRG 位触发（1），则将 ABT 区域（缓存器 0 至 7）中的一个 TRQ 位设为 1。除了此 TRQ 位外，应用软件还可以为不属于 ABT 区域的其它发送-报文缓存器请求发送。 这种情况下，接下来发送由内部发送查找处理（TX 查找）查找的报文。 该处理计算所有 TRQ 置 1 的发送-报文缓存器，并选择具有最高优先级标识符的报文缓存器作为下次发送。 如果有两个或更多的具有最高优先级的标识符（例如，同样的标识符），则首先发送位于最低报文缓存器编号的报文。

当报文帧成功发送时，执行以下操作。

- 相应发送报文缓存器的 TRQ 位自动清为 0。
- 将 C0INTS 寄存器的发送完成状态位 CINTS0 设为 1（如果将相应的发送报文缓存器的中断使能位（IE）设为 1）。
- 如果将 C0IE.CIE0 位设为 1 且将相应发送报文缓存器的中断使能位（IE）设为 1，则输出中断请求信号 INTRRX1。

20.10.2 发送记录列表功能

发送记录列表（THL）功能在发送记录列表中记录发送各个数据帧或远程帧的接收报文缓存器的编号。THL 的组成有：最多等于 7 个报文的存储单元、对应 C0LOPT 寄存器的最末报文出指针（LOPT）以及有对应 C0TGPT 寄存器的发送记录列表取指针（TGPT）。

CAN 模块从初始化模式转换到某一种操作模式后的瞬间，THL 不确定。

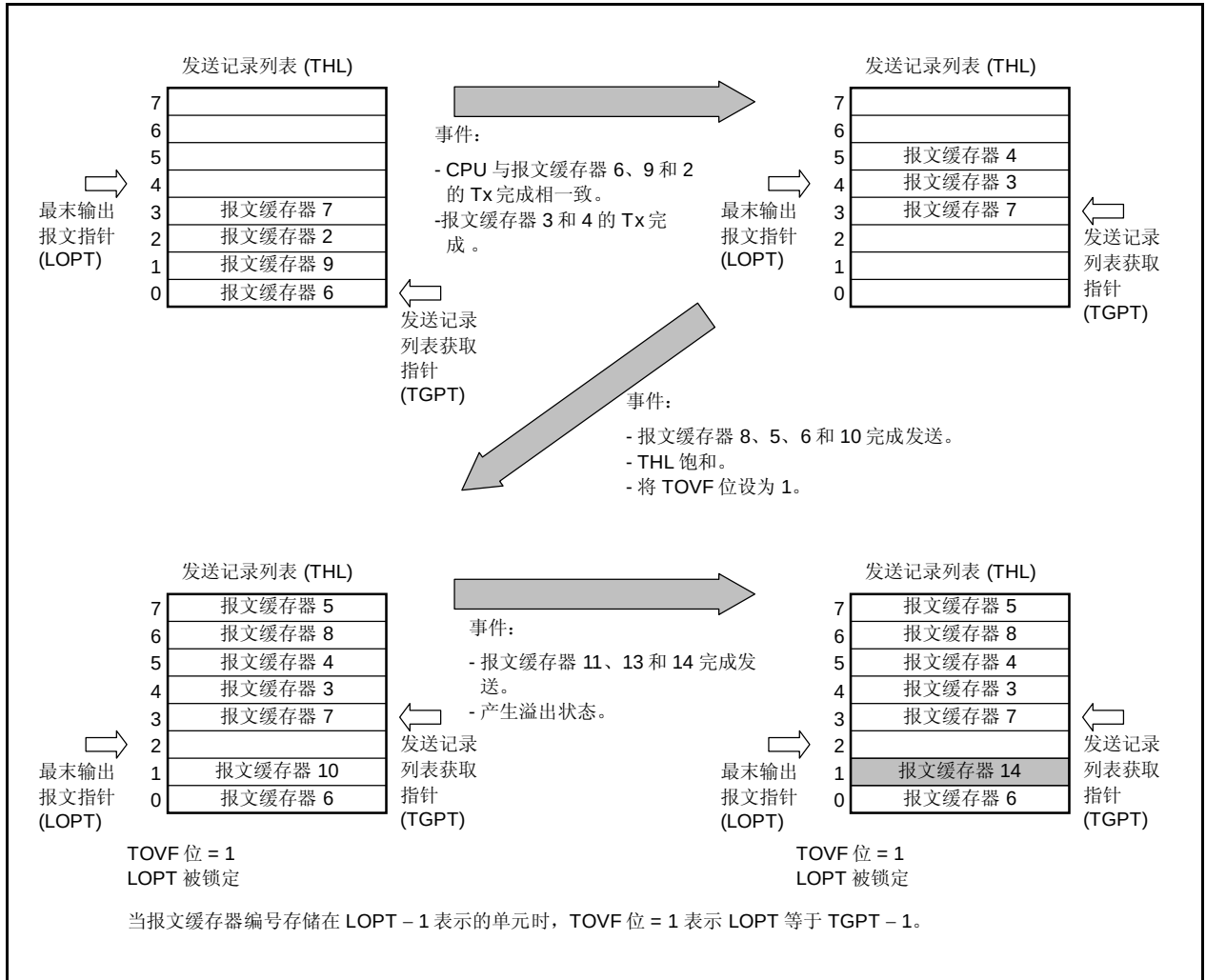
C0LOPT 寄存器保持由 LOPT 指针减 1 的值所指示的 THL 的条目。因此，通过读取 C0LOPT 寄存器，可检查发送数据帧或远程帧的报文缓存器的编号。LOPT 指针用作写入指针，指示报文缓存器的编号被记录到 THL 的哪一部分。任何时候发送数据帧或远程帧，都将相应的报文缓存器编号记录到由 LOPT 指针指示的 THL 单元。每次完成到 THL 的记录时，LOPT 指针自动加 1。如此一来，接收并存储帧的报文缓存器的编号将按照时间顺序进行记录。

TGPT 指针用作读取指针，从 THL 读取已记录的报文缓存器的编号。该指针表示 CPU 还没有读取的 THL 的起始单元。通过用软件读取 C0TGPT 寄存器，可读取已完成发送的报文缓存器的编号。每次从 C0TGPT 寄存器读取报文缓存器的编号之后，TGPT 指针自动加 1。

如果 TGPT 指针的值与 LOPT 指针的值匹配，则将 C0TGPT.RHPM 位（发送记录列表指针重合）设为 1。这表示在 THL 中不存在尚未读取的报文缓存器的编号。如果记录了新的报文缓存器编号，LOPT 指针自动加 1，而由于它的值与 TGPT 的值不再相符，因而 THPM 位被清除。换言之，在 THL 中存在未读取的报文缓存器的编号。

如果 LOPT 指针加 1 并与 TGPT 指针减 1 的值匹配，则将 C0TGPT 寄存器的 TOVF 位（发送记录列表溢出）设为 1。这表示 THL 中全是未被读取的报文缓存器的编号。如果发送新的报文，最末记录的报文缓存器编号被接收并存储了新报文的报文缓存器的编号覆盖。因此，TOVF 位置位（1）之后，THL 中记录的报文缓存器的编号不能完全反映按时间排列的顺序。但是，CPU 可通过查找所有的发送报文缓存器指定发送报文缓存器的编号。（在 CPU 重新设置之前执行。）在 THL 不溢出的前提下，可总共发生多达 6 次的发送完成。

图 20-31. 发送记录列表



20.10.3 自动块传输（ABT）

自动块传输（ABT）功能用于不与 CPU 交互时连续发送两个或更多的数据帧。赋予 ABT 功能的发送报文缓存器的数目固定为 8（报文缓存器编号 0 至 7）。

通过将 C0CTRL.OPMODE2 至 C0CTRL.OPMODE0 位设为 010B，可选择“使用自动块传输功能的正常操作模式”（此后称之为 ABT 模式）。

为了产生 ABT 发送请求，首先通过软件定义报文缓存器。在所有 ABT 用到的报文缓存器中将 C0MCONFm.MA0 位置为（1），并通过将 C0MCONFm.MT2 至 C0MCONFm.MT0 位设为 000B 把所有的缓存器定义为发送报文缓存器。即使当 ID 正在用于所有的报文缓存器时，也务必要将 ABT 所使用的报文缓存器设为相同的 ID。使用 C0MIDLm 和 C0MIDHm 寄存器设置各个报文缓存器的 ID，以便使用两个或更多的 ID。在产生 ABT 的发送请求之前设置 C0MDLCm 以及 C0MDATA0m 至 C0MDATA7m 寄存器。

完成用于 ABT 的报文缓存器的初始化之后，需要将 C0MCTRLm.RDY 位设为（1）。在 ABT 模式中，不必要通过软件操作 C0MCTRLm.TRQ 位。

ABT 报文缓存器的数据准备好之后，将 C0GMABT.ABTTRG 位设为 1，然后开始自动块传输。当开始 ABT 时，第 1 个报文缓存器（报文缓存器 0）中的 TRQ 位自动置 1。报文缓存器 0 的数据传输完成之后，下一个报文缓存器，即报文缓存器 1 的 TRQ 位自动置位。如此这般，连续进行传输。

执行连续发送期间，可通过程序在发送请求（TRQ 位）自动置位的间隔中插入延迟时间。插入的延迟时间由 C0GMABTD 寄存器定义。延迟时间的单位是 DBT（数据位时间）。DBT 取决于 C0BRP 和 C0BTR 寄存器设定的时间。

ABT 期间，不查找 ABT 发送 ID 的优先级。而是连续发送报文缓存器 0 至 7 的数据。当来自报文缓存器 7 的数据帧发送完成时，ABTTRG 位自动清为 0 并结束 ABT 操作。

如果 ABT 期间将 ABT 报文缓存器的 RDY 位清除（0），则该缓存器不发送数据帧，ABT 停止并且 ABTTRG 位被清除。此后，通过用软件将 RDY 和 ABTTRG 位置 1，可将发送从 ABT 停止的报文缓存器处恢复。若不从 ABT 停止的报文缓存器处恢复发送，可在 ABT 模式停止时将 C0GMABT.ABTCLR 位置为 1 并将 ABTTRG 位清为 0，使得内部 ABT 引擎复位。这种情况下，如果 ABTCLR 清为 0 之后将 ABTTRG 位置 1，则发送从报文缓存器 0 开始。

可使用中断来检查所有用于 ABT 的报文缓存器的数据帧是否已经发送。为此，需要将除了最后一个报文缓存器的各报文缓存器的 0MCTRLm.IE 位清除（0）。

如果将非 ABT 功能的报文缓存器（报文缓存器 8 至 31）分配至一个发送报文缓存器，发送报文的优先级由当前发送被挂起的 ABT 报文缓存器的发送报文缓存器的优先级以及非 ABT 功能的报文缓存器的发送报文缓存器的优先级所决定。

发送来自 ABT 报文缓存器的数据帧不记录在发送记录列表（THL）中。

- 注意事项**
1. 为了从报文缓存器 0 恢复使用 ABT 的正常操作模式，在 ABTTRG 清为 0 的同时将 ABTCLR 位设为 1。如果在 ABTTRG 位设为 1 的同时将 ABTCLR 位也设为 1，则不能保证后续操作。
 2. 当通过将 ABTCLR 设为 1 清除自动块传输引擎时，已请求的清除处理完成之后，ABTCLR 位立即自动清除为 0。
 3. 在初始化模式不要设置 ABTTRG 位。如果在初始化模式下设置 ABTTRG 位，那么从初始化模式变为 ABT 模式之后，不能保证正确的操作。
 4. 在使用 ABT 的正常操作模式下，不要使用软件将 ABT 报文缓存器的 TRQ 位置 1。否则，操作得不到保证。
 5. C0GMABTD 寄存器用于设置延迟时间。当 ABT 模式下，发送请求按照各个报文 ABT 模式顺序发送的报文编号发起时，该延迟时间插入到前面的 ABT 报文完成到下一个 ABT 报文的 TRQ 位置位期间。报文实际发送到 CAN 总线的时刻，根据其它站点的状态与非 ABT 报文（报文缓存器 8 至 31）的发送请求所设置状态的不同而异。
 6. 如果产生一个非 ABT 报文的发送请求，且未在 ABT 的发送请求自动置位（C0GMABTD 寄存器=00H）的间隔内插入延迟时间，则可能发送非 ABT 报文的报文，而不考虑其与 ABT 报文的相对优先级。
 7. ABTTRG 位=1 时不要将 RDY 位清为 0。
 8. 如果从 ABT 的正常操作模式的另一个节点接收报文，即使 C0GMABTD 寄存器=00H，报文也可能在一帧的时间过后发送。

备注 m = 00 至 31

20.10.4 发送中止处理

备注 m = 00 至 31

(1) 发送中止处理

用户可将 C0MCTRLm.TRQ 位清为 0 来中止发送请求。如果中止成功则 TRQ 位立刻被清除。可通过指示 CAN 总线发送状态的 C0TGPT 寄存器和 C0CTRL.TSTAT 位来检查发送是否成功中止（关于细节，参见图 20-45 中的处理）。

(2) 使用自动块传输（ABT）的正常操作模式中除 ABT 发送之外的发送中止处理

用户可将 C0GMABT.ABTTRG TRQ 位清为 0 来中止发送请求。检查 C0GMABT 寄存器的 ABTTRG 位=0 之后，将 C0MCTRLm.TRQ 位清为 0。如果中止成功则 TRQ 位立刻被清除。可通过指示 CAN 总线发送状态的 C0TGPT 寄存器和 C0CTRL.TSTAT 位来检查发送是否成功中止（关于细节，参见图 20-46 中的处理）。

(3) 使用自动块传输（ABT）的正常操作模式中发送的中止

为中止已经开始的 ABT，要将 C0GMABT.ABTTRG 清为 0。这种情况下，如果当前正在发送 ABT 报文且直到发送完成（成功或失败），ABTTRG 位保持为 1，一旦发送结束就清除为 0。这会中止 ABT。

如果上一个发送（ABT 之前）成功，那么使用 ABT 的正常操作模式的内部 ABT 指针指向下一个将要发送的报文缓存器。

错误传输的情况下，内部 ABT 指针的位置取决于上一个发送报文缓存器中 TRQ 位的状态。如果在请求清除 ABTTRG 位时将 TRQ 位置 1，那么内部 ABT 指针指向上一个发送报文缓存器（关于细节，参见图 20-47（a）中的处理）。如果在请求清除 ABTTRG 位时将 TRQ 清为 0，那么内部 ABT 指针加 1 并指向位于 ABT 区域的下一个报文缓存器（关于细节，参见图 20-47 中的处理）。

注意事项 务必通过将 ABTTRG 位清为 0 中止 ABT。如果通过清除 RDY 请求中止发送，则操作得不到保证。

当 ABT 中止之后且 ABTTRG 位置为 1 时恢复 ABT 模式的正常操作，由下表决定下一个将要发送的 ABT 报文缓存器。

ABT 报文缓存器 TRQ 位的状态	成功发送后中止	错误发送后中止
置位（1）	ABT 区域 ^注 中下一个报文缓存器	ABT 区域中同一个报文缓存器
清除（0）	ABT 区域 ^注 中下一个报文缓存器	ABT 区域 ^注 中下一个报文缓存器

注 仅在 ABT 区域中存在发送就绪 ABT 的报文缓存器时，才可以进行以上的恢复操作。例如，如果报文缓存器 7 的发送成功完成，那么即使将 ABTTRG 位清为 0，报文缓存器 7 的 ABT 正在进行时产生的中止请求也会被视为 ABT 完成，而不是中止。如果将 ABT 区域中下一个报文缓存器的 C0MCTRLm.RDY 位清为 0，则内部 ABT 指针保持，但是即使将 ABTTRG 位设为 1 也不进行恢复操作，且 ABT 立刻结束。

备注 m = 00 至 31

20.10.5 远程帧发送

远程帧仅能由发送报文缓存器发送。通过 C0MCONFm.RTR 位来设置发送数据帧还是发送远程帧。RTR 位置位（1）时设置为远程帧发送。

备注 m = 00 至 31

20.11 节电模式

20.11.1 CAN睡眠模式

CAN 睡眠模式可将 CAN 控制器设为待机模式，以减少电源消耗。CAN 模块可从所有模式进入到 CAN 睡眠模式。CAN 睡眠模式解除之后返回到进入 CAN 睡眠模式之前同样的模式。

在 CAN 睡眠模式中，即使产生了发送请求或发送请求待决时，CAN 模块也不发送报文。

(1) 进入 CAN 睡眠模式

CPU 通过将 01B 写入到 C0CTRL.PSMODE1 和 C0CTRL.PSMODE0 位产生 CAN 睡眠模式转换请求。

在下列条件下响应此转换请求。

(i) CAN 模块已经处于下列操作模式之一

- 正常操作模式
- 使用 ABT 的正常操作模式
- 只收模式
- 单次模式
- 自检模式
- 以上所有操作模式中的 CAN 停止模式

(ii) CAN 总线状态为总线空闲（帧间间隔的第 4 位为隐性位）^注

注 如果 CAN 总线固定为显性，则将转换至 CAN 睡眠模式的请求挂起。同样地，从 CAN 停止模式到 CAN 睡眠模式的转换也与 CAN 总线状态无关。

(iii) 无正在挂起的发送请求

如果不满足上述条件的任何一条，CAN 模块将操作如下：

- 如果由初始化模式请求进入 CAN 睡眠模式，则忽略 CAN 模式转换请求且 CAN 模式仍然处于初始化模式。
- 如果从一种操作模式请求进入 CAN 睡眠模式时，而 CAN 总线状态不是总线空闲（例如，CAN 总线状态或为发送或为接收），则不可能立即转换至 CAN 睡眠模式。这种情况下，只好将 CAN 睡眠模式转换请求挂起，直到 CAN 总线状态变为总线空闲（帧间间隔的第 4 位为隐性位）为止。从 CAN 睡眠模式请求到成功转换的时间内，PSMODE1 和 PSMODE0 位仍为 00B。当模块进入 CAN 睡眠模式时，PSMODE1 和 PSMODE0 位置为 01B。
- 当 CAN 模块处于某种工作模式时，如果同时产生转换至初始化模式的请求和转换至 CAN 睡眠模式的请求，则使能到初始化模式的请求。CAN 模块在预定的时刻进入初始化模式。此时，CAN 睡眠模式请求不是被挂起而是被忽略。

- 即使不是同时请求初始化模式和睡眠模式时（例如，当产生第 2 个请求时尚未确认第 1 个请求），初始化请求也具有高于 CAN 睡眠模式请求的优先级。例如，CAN 睡眠模式请求和转换到 CAN 睡眠模式的请求被挂起，则产生初始化模式请求。当出现初始化模式的请求产生时，CAN 睡眠模式请求在其提交那一刻就会被取消。同样地，如果初始化模式转换请求和至初始化模式的转换请求挂起时，会产生 CAN 睡眠模式。当 CAN 睡眠模式产生时，初始化请求在其产生那一刻就会被取消。

(2) CAN 睡眠模式的状态

CAN 模块进入睡眠模式之后处于下列状态中的其中一种。

- 内部工作时钟停止并进入低功耗状态。
- 检测 CAN 接收引脚（CRXD0）下降沿功能保持有效，以便从 CAN 总线唤醒 CAN 模块。
- 为了从 CPU 中唤醒 CAN 模块，可向 PSMODE1 和 PSMODE0 位写入数据，但是不能向其它 CAN0 模块寄存器或位写入任何数据。
- 除了 C0LIPT, C0RGPT, C0LOPT 和 C0TGPT 寄存器，可读取 CAN0 模块寄存器。
- 不能读写 CAN0 报文缓存器寄存器。
- 转换至初始化模式的请求不被响应并被忽略。

(3) 解除 CAN 睡眠模式

以下事件可解除 CAN 睡眠模式。

- 将 00B 设到 PSMODE1 和 PSMODE0 位时。
- 当检测到 CAN 接收引脚（CRXD0）的下降沿时（例如，CAN 总线电平由隐性切换为显性）。

注意事项 1. 即使下降沿属于一个接收报文的 SOF，也不会接收并存储该报文。如果 CPU 在 CAN 处于睡眠模式时关掉 CAN 的时钟，那么稍后不能解除 CAN 睡眠模式且 PSMODE1 和 PSMODE0 位将继续为 01B，除非重新给 CAN 提供时钟。除此之外，之后的将不能被接收报文。

2. 如果在提供 CAN 时钟期间在 CAN 接收引脚（CRXD0）检测到下降沿，则必须通过软件清除 PSMODE0 位。（关于详情，参见图 20-52 清除 CAN 睡眠/停止模式。）

解除睡眠模式之后，CAN 模块返回到请求 CAN 睡眠模式前的操作模式并将 PSMODE1 和 PSMODE0 位复位为 00B。如果通过 CAN 总线状态的改变解除 CAN 睡眠模式，则将 C0INTS.CINTS5 位设为 1，而不管 C0IE.CIE 位的值。从 CAN 睡眠模式解除之后，CAN 模块通过在 CAN 总线上自动检测 11 个连续隐性电平位然后重新加入 CAN 总线。

当 CAN 模块处于睡眠模式时，产生转换至初始化模式的请求时会忽略该请求。通过软件解除从 CAN 睡眠模式到初始化模式的转换。

20.11.2 CAN 停止模式

CAN 停止模式可用于将 CAN 控制器设为待机模式，以减少功耗。CAN 模块仅能从 CAN 睡眠模式进入 CAN 停止模式。CAN 停止模式的解除将 CAN 模块置于 CAN 睡眠模式。

CAN 停止模式不能通过 CAN 总线状态的改变解除，而只能通过将 01B 写入 C0CTRL.PSMODE1 和 C0CTRL.PSMODE0 位解除（切换至 CAN 睡眠模式）。即便当发送请求产生或挂起，也没有报文发送。

(1) 进入 CAN 停止模式

通过将 11B 写入 PSMODE1 和 PSMODE0A 位产生 CAN 停止模式转换请求。

仅当 CAN 模块处于 CAN 模式时可响应 CAN 停止模式请求。在所有其它模式中，都忽略该请求。

注意事项 为了将 CAN 模块设为 CAN 停止模式，模块必须处于 CAN 睡眠模式。为了确认模块处于睡眠模式，应检查 PSMODE1 和 PSMODE0 位= 01B，然后再请求进入 CAN 停止模式。正在进行该处理期间，如果在 CAN 接收引脚（CRXD0）发生总线变化，则自动清除 CAN 睡眠模式。这种情况下，不能响应 CAN 停止模式转换请求（但是，提供 CAN 时钟期间，在 CAN 接收引脚（CRXD0）的总线电平改变之后必须用软件将 PSMODE0 清除）。

(2) CAN 停止模式下的状态

CAN 模块进入 CAN 停止模式之后处于下列状态中的其中一种。

- 内部工作时钟停止并进入低功耗状态。
- 为了从 CPU 中唤醒 CAN 模块，可向 PSMODE1 和 PSMODE0 位写入数据，但是不能向其它 CAN0 模块寄存器或位写入任何数据。
- 除了 C0LIPT，C0RGPT，C0LOPT 和 C0TGPT 寄存器，可读取 CAN0 模块寄存器。
- 不能读写 CAN0 报文缓存器寄存器。
- 初始化模式转换请求不被响应并被忽略。

(3) 解除 CAN 停止模式

仅能通过将 01B 写入 PSMODE1 和 PSMODE0 位解除 CAN 停止模式。CAN 停止模式解除之后，CAN 模块进入 CAN 睡眠模式。

当 CAN 模块处于 CAN 停止模式时，如果请求初始化模式，则该请求会被忽略。通过软件解除 CAN 停止模式和 CAN 睡眠模式到初始化模式的请求。在不进入 CAN 睡眠模式的情况下，不可能从 CAN 停止模式直接进入另外的操作模式。此请求将被忽略。

20.11.3 使用节电模式的举例

在某些应用系统中，可能有必要将 CPU 置于节电模式以减少功耗。通过结合使用 CAN 模块指定的节电模式和 CPU 指定的节电模式，可由 CAN 总线将 CPU 从节电状态唤醒。

这里是一个使用节电模式的例子。

首先，将 CAN 模块置于 CAN 睡眠模式（PSMODE1，PSMODE0 位 = 01B）。然后将 CPU 置于节电模式。该状态下，如果在 CRXD0 信号引脚处检测到隐性到显性的电平转换，则将 CAN 模块的 CINTS5 位置 1。如果将 C0CTRL.CIE5 位设为 1，则产生唤醒中断（INTC0WUP）。CAN 模块自动从 CAN 睡眠模式（PSMODE1，PSMODE0 位 = 00B）解除并返回到正常操作模式。（但是，提供 CAN 时钟期间，必须在 CAN 接收引脚（CRXD0）检测到总线电平变化之后用软件将 PSMODE0 清除）。CPU 响应 INTC0WUP，便可解除自身的节电模式并返回到正常操作模式。

为了进一步减少 CPU 的功耗，可以关闭包括 CAN 模块时钟在内的内部时钟。这种情况下，将 CAN 模块置于 CAN 睡眠模式之后关掉 CAN 模块的工作时钟。然后 CPU 便进入 CPU 时钟关闭的节电模式。该状态下如果在 CRXD0 信号引脚检测到从隐性到显性的边沿转换，即使不给 CAN 模块提供时钟，它也可将 CINTS5 位设为 1 并产生唤醒中断（INTC0WUP）。然而，因为 CAN 模块时钟关闭，其它功能不能运行，模块仍然处于 CAN 睡眠模式。CPU 响应 INTC0WUP，解除自身的节电模式，在振荡稳定时间过后恢复包括至 CAN 模块时钟的内部时钟供应，并开始执行指令。当恢复时钟供应时，CAN 模块立刻从 CAN 睡眠模式中解除并返回到正常操作模式（PSMODE1，PSMODE0 位 = 00B）。

20.12 中断功能

CAN 模块提供 6 种不同的中断源。

这些中断源发生存放在中断状态寄存器中。由 6 个中断源可产生 4 个单独的中断请求信号。当对应于两个或更多的中断源的中断请求信号产生时，可使用中断状态寄存器来识别中断源。中断源产生中断之后，必须通过软件将其对应的中断状态位清 0。

表 20-20. CAN 模块中断源列表

No.	中断状态位		中断使能位		中断请求信号	中断源说明
	名称	寄存器	名称	寄存器		
1	CINTS0 ^{注1}	C0INTS	CIE0 ^{注1}	C0IE	INTC0TRX	从报文缓存器 m 成功发送报文帧
2	CINTS1 ^{注1}	C0INTS	CIE1 ^{注1}	C0IE	INTC0REC	到报文缓存器 m 的有效报文帧接收完成中断
3	CINTS2	C0INTS	CIE2	C0IE	INTC0ERR	CAN 模块错误状态中断 ^{注2}
4	CINTS3	C0INTS	CIE3	C0IE		CAN 模块协议错误中断 ^{注3}
5	CINTS4	C0INTS	CIE4	C0IE		CAN 模块仲裁失败中断
6	CINTS5	C0INTS	CIE5	C0IE	INTC0WUP	CAN 模块由 CAN 睡眠模式唤醒中断 ^{注4}

- 注
1. 必须将相应报文缓存器的 C0MCTRL.IE 位（报文缓存器中断使能位）设为 1，以为此相应报文缓存器参与了中断产生处理。
 2. 当发送/接收错误计数器位于告警级，或处于错误既成状态和掉线状态时产生该中断。
 3. 当发生填充错误、格式错误、ACK 错误、位错误或 CRC 错误时产生该中断。
 4. 当由于 CAN 接收引脚检测到下降沿而将 CAN 模块从 CAN 睡眠模式中唤醒时，产生该中断（CAN 总线由隐性转换为显性）。

备注 m = 00 至 31

20.13 诊断功能和特殊操作模式

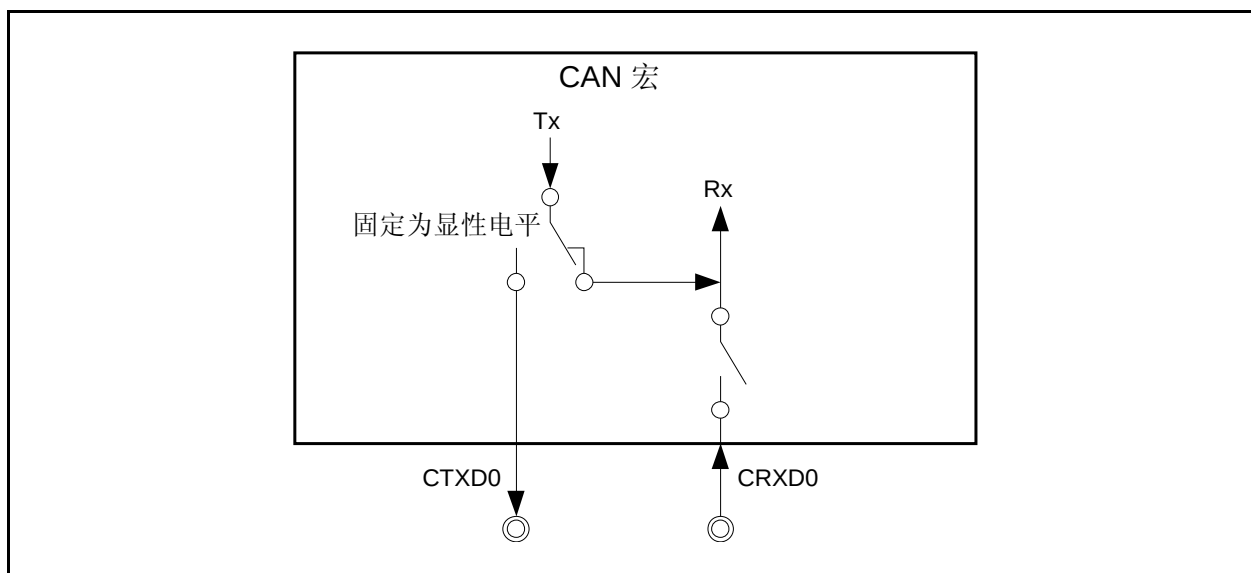
CAN 模块提供只收模式、单次模式和自检模式来支持 CAN 总线诊断功能或特殊 CAN 通信方式的操作。

20.13.1 只收模式

只收模式用于监控接收报文而不对 CAN 总线引起任何干扰，可用于 CAN 总线分析节点。

例如，此模式可用于自动波特率检测。CAN 模块中的波特率在检测到“有效接收”之后才发生变化，因此使得模块中的波特率匹配（“有效接收”指连接在到 CAN 总线的节点间使用适当的 ACK，完成 CAN 协议层面上报文帧的无差错接收）。有效接收不需要将报文帧存储到接收报文缓存器（数据帧）或发送报文缓存器（远程帧）中。有效接收事件通过 C0CTRL.VALID 位置位（1）检查。

图 20-32. 只收模式中 CAN 模块的末端连接



只收模式中，CAN 模块不发送报文到 CAN 总线。将报文缓存器定义位发送报文缓存器的发送请求挂起待决。

在只收模式中，CAN 模块中的 CAN 发送引脚（CTXD0）固定为隐性电平。因此，即使在报文帧接收期间检测到 CAN 总线错误，也不能将生成错误标志从 CAN 模块发送至 CAN 总线。因为 CAN 模块不产生发送，所以发送错误计数器的 C0ERC.TEC7 至 C0ERC.TEC0 位不更新。因此，CAN 模块在只收模式下不进入掉线状态。

此外，该模式中当报文帧有效接收时也不返回 ACK 至 CAN 总线。在内部，本地节点认为其已经发送了 ACK。不能将过载帧发送至 CAN 总线。

注意事项 如果仅有两个 CAN 节点连接到 CAN 总线，且其中一个在只收模式下工作，那么在 CAN 总线上没有 ACK。由于没有了 ACK，发送节点将发送有效错误标志并重复发送报文帧。发送节点发送 16 次报文帧（假设开始时错误计数器为 0 且没有产生其它错误）之后变为错误既成。当第 17 次发送报文帧时，发送节点会产生一个既成错误标志。只收模式下接收节点在该点检测第 1 个有效报文帧，并将 VALID 位在首次就置为 1。

20.13.2 单次模式

在单次模式中，关闭 CAN 协议中定义的自动重发功能。（依照 CAN 协议，由仲裁失败或错误产生中止的报文帧传输，必须在不用软件控制的情况下重传。）单次模式的所有其它操作与正常操作模式下相同。单次模式功能不能与使用 ABT 的正常操作模式结合使用。

单次模式根据 C0CTRL.AL 位的设置禁止已中止报文帧传输的重传。当将 AL 位清为 0 时，禁止基于仲裁失败和基于错误发生的重传。如果将 AL 位设为 1，禁止基于错误产生的重传，但是允许基于仲裁失败的重传。因而，定义为发送报文缓存器的报文缓存器的 C0MCTRLm.TRQ 位通过以下的事件被清除为 0。

- 报文帧的成功发送
- 发送报文帧期间仲裁失败（AL 位 = 0）
- 发送报文帧期间发送错误

仲裁失败事件和错误产生事件可通过检查 C0INTS.CINTS4 和 C0INTS.CINTS3 位来区分，而错误类型可通过读取寄存器的 C0LEC.LEC2 至 C0LEC.LEC0 位来鉴别。

当成功发送报文帧时，发送完成中断将 C0INTS 寄存器的 CINTS0 位设置为 1。如果同时将 C0IE.CIE0 位设为 1，则输出中断请求信号。

单次模式可用来仿真定时触发通信方式（例如，TTCAN 等级 1）。

注意事项 AL 位仅在单次模式下有效。它不影响其它操作模式中仲裁失败时的重传操作。

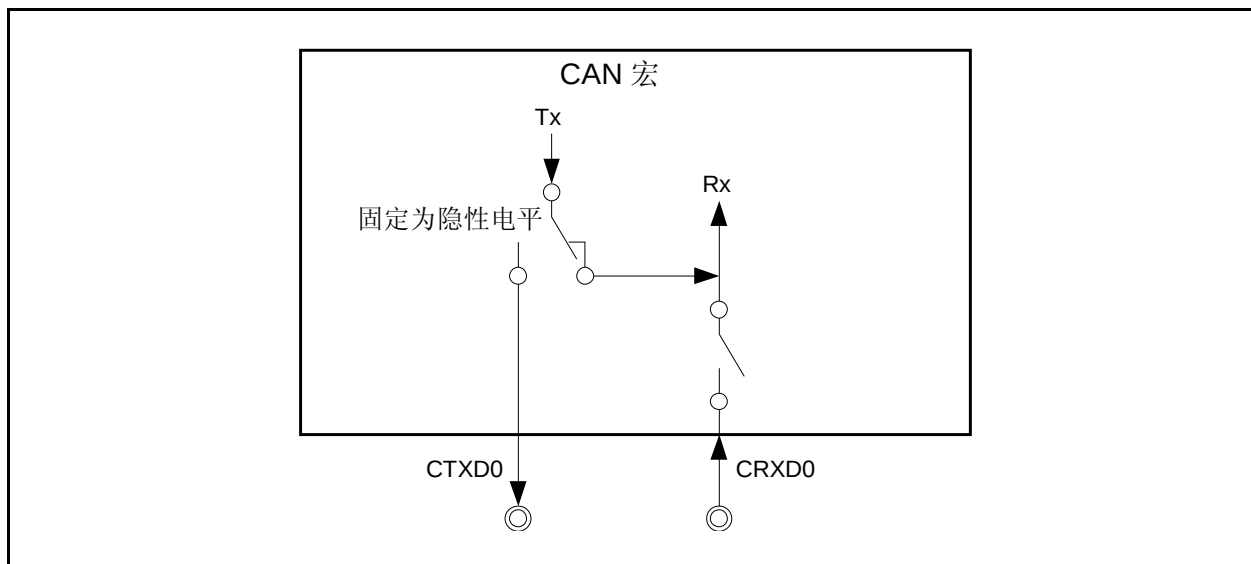
20.13.3 自检模式

自检模式下，可在 CAN 节点不与 CAN 总线连接或不影响 CAN 总线的情况下，对报文帧发送和报文帧接收进行检测。

在自检模式中，CAN 模块与 CAN 总线完全断开，但是发送和接收在内部环回。CAN 发送引脚（CTXD0）固定为隐性电平。

如果 CAN 模块从自检模式进入 CAN 睡眠模式后，在 CAN 接收引脚（CRXD0）上检测到下降沿，无论如何，模块从 CAN 睡眠模式解除时的方法与从其它操作模式解除时相同。（但是当 CAN 时钟供应期间解除睡眠模式时，在 CAN 接收引脚（CRXD0）检测到下降沿之后必须用软件将 PSMODE0 位清除。）为了使模块保持在 CAN 睡眠模式，将 CAN 接收引脚（CRXD0）用作端口引脚。

图 20-33. 自检模式下 CAN 模块的末端连接



注意事项 通过在报文缓存器 0 中接收数据帧，使用 TSLOCK 位的时间标记功能停止 TSOUT 信号的来回切换。因此，必须将报文缓存器 0 设为接收报文缓存器。因为接收报文缓存器不能接收远程帧，所以远程帧的接收不能停止 TSOUT 信号的来回切换。当用除了报文缓存器 0 之外的报文缓存器中接收数据帧时，不停止 TSOUT 信号的来回切换。

由于这些原因，当 CAN 模块处于使用 ABT 的正常操作模式时，报文缓存器 0 不能接收数据帧，因为报文缓存器 0 必须设为发送报文缓存器。因此在该操作模式下，不能使用通过 TSLOCK 位停止 TSOUT 信号来回切换的功能。

20.15 波特率设置

20.15.1 波特率设置条件

务必使设置处于如下限定值范围之内，以确保 CAN 控制器正确工作。

- (a) $5TQ \leq SPT$ (采样点) $\leq 17 TQ$
 $SPT = TSEG1 + 1$
- (b) $8 TQ \leq DBT$ (数据位时间) $\leq 25 TQ$
 $DBT = TSEG1 + TSEG2 + 1TQ = TSEG2 + SPT$
- (c) $1 TQ \leq SJW$ (同步跳转宽度) $\leq 4TQ$
 $SJW \leq DBT - SPT$
- (d) $4 \leq TSEG1 \leq 16$ [$3 \leq TSEG1[3:0]$ 的设定值 ≤ 15]
- (e) $1 \leq TSEG2 \leq 8$ [$0 \leq TSEG2[2:0]$ 的设定值 ≤ 7]

备注 $TQ = 1/frq$ (frq: CAN 协议层基础系统时钟)
 $TSEG1[3:0]$ (C0BTR.TSEG13 至 C0BTR.TSEG10 位)
 $TSEG2[2:0]$ (C0BTR.TSEG22 至 C0BTR.TSEG20 位)

表 20-21 显示了满足以上条件的位速率的组合。

表 20-21. 可设置的位速率组合 (1/3)

有效位速率设置					C0BTR 寄存器设定值		采样点 (单位: %)
DBT 长度	SYNC SEGMENT	PROP SEGMENT	PHASE SEGMENT1	PHASE SEGMENT2	TSEG13 至 TSEG10	TSEG22 至 TSEG20	
25	1	8	8	8	1111	111	68.0
24	1	7	8	8	1110	111	66.7
24	1	9	7	7	1111	110	70.8
23	1	6	8	8	1101	111	65.2
23	1	8	7	7	1110	110	69.6
23	1	10	6	6	1111	101	73.9
22	1	5	8	8	1100	111	63.6
22	1	7	7	7	1101	110	68.2
22	1	9	6	6	1110	101	72.7
22	1	11	5	5	1111	100	77.3
21	1	4	8	8	1011	111	61.9
21	1	6	7	7	1100	110	66.7
21	1	8	6	6	1101	101	71.4
21	1	10	5	5	1110	100	76.2
21	1	12	4	4	1111	011	81.0
20	1	3	8	8	1010	111	60.0
20	1	5	7	7	1011	110	65.0
20	1	7	6	6	1100	101	70.0
20	1	9	5	5	1101	100	75.0
20	1	11	4	4	1110	011	80.0
20	1	13	3	3	1111	010	85.0
19	1	2	8	8	1001	111	57.9
19	1	4	7	7	1010	110	63.2
19	1	6	6	6	1011	101	68.4
19	1	8	5	5	1100	100	73.7
19	1	10	4	4	1101	011	78.9
19	1	12	3	3	1110	010	84.2
19	1	14	2	2	1111	001	89.5
18	1	1	8	8	1000	111	55.6
18	1	3	7	7	1001	110	61.1
18	1	5	6	6	1010	101	66.7
18	1	7	5	5	1011	100	72.2
18	1	9	4	4	1100	011	77.8
18	1	11	3	3	1101	010	83.3
18	1	13	2	2	1110	001	88.9
18	1	15	1	1	1111	000	94.4

表 20-21. 可设置的位速率组合 (2/3)

有效位速率设置					C0BTR 寄存器设定值		采样点 (单位: %)
DBT 长度	SYNC SEGMENT	PROP SEGMENT	PHASE SEGMENT1	PHASE SEGMENT2	TSEG13 至 TSEG10	TSEG22 至 TSEG20	
17	1	2	7	7	1000	110	58.8
17	1	4	6	6	1001	101	64.7
17	1	6	5	5	1010	100	70.6
17	1	8	4	4	1011	011	76.5
17	1	10	3	3	1100	010	82.4
17	1	12	2	2	1101	001	88.2
17	1	14	1	1	1110	000	94.1
16	1	1	7	7	0111	110	56.3
16	1	3	6	6	1000	101	62.5
16	1	5	5	5	1001	100	68.8
16	1	7	4	4	1010	011	75.0
16	1	9	3	3	1011	010	81.3
16	1	11	2	2	1100	001	87.5
16	1	13	1	1	1101	000	93.8
15	1	2	6	6	0111	101	60.0
15	1	4	5	5	1000	100	66.7
15	1	6	4	4	1001	011	73.3
15	1	8	3	3	1010	010	80.0
15	1	10	2	2	1011	001	86.7
15	1	12	1	1	1100	000	93.3
14	1	1	6	6	0110	101	57.1
14	1	3	5	5	0111	100	64.3
14	1	5	4	4	1000	011	71.4
14	1	7	3	3	1001	010	78.6
14	1	9	2	2	1010	001	85.7
14	1	11	1	1	1011	000	92.9
13	1	2	5	5	0110	100	61.5
13	1	4	4	4	0111	011	69.2
13	1	6	3	3	1000	010	76.9
13	1	8	2	2	1001	001	84.6
13	1	10	1	1	1010	000	92.3
12	1	1	5	5	0101	100	58.3
12	1	3	4	4	0110	011	66.7
12	1	5	3	3	0111	010	75.0
12	1	7	2	2	1000	001	83.3
12	1	9	1	1	1001	000	91.7

表 20-21. 可设置的位速率组合 (3/3)

有效位速率设置					C0BTR 寄存器设定值		采样点 (单位: %)
DBT 长度	SYNC SEGMENT	PROP SEGMENT	PHASE SEGMENT1	PHASE SEGMENT2	TSEG13 至 TSEG10	TSEG22 至 TSEG20	
11	1	2	4	4	0101	011	63.6
11	1	4	3	3	0110	010	72.7
11	1	6	2	2	0111	001	81.8
11	1	8	1	1	1000	000	90.9
10	1	1	4	4	0100	011	60.0
10	1	3	3	3	0101	010	70.0
10	1	5	2	2	0110	001	80.0
10	1	7	1	1	0111	000	90.0
9	1	2	3	3	0100	010	66.7
9	1	4	2	2	0101	001	77.8
9	1	6	1	1	0110	000	88.9
8	1	1	3	3	0011	010	62.5
8	1	3	2	2	0100	001	75.0
8	1	5	1	1	0101	000	87.5
7 ^注	1	2	2	2	0011	001	71.4
7 ^注	1	4	1	1	0100	000	85.7
6 ^注	1	1	2	2	0010	001	66.7
6 ^注	1	3	1	1	0011	000	83.3
5 ^注	1	2	1	1	0010	000	80.0
4 ^注	1	1	1	1	0001	000	75.0

注 仅当 C0BRP 寄存器的值不是 00H 时，使用值为 7 或更小的 DBT 的设置有效。

注意事项 表 20-21 中的值不保证网络系统的操作。 应将振荡偏差和 CAN 总线及 CAN 收发器的延时考虑在内，以彻底地检查在网络系统中的效果。

20.15.2 波特率设置的典型示例

表 20-22 和 20-23 显示了波特率设置的典型示例。

表 20-22. 波特率设置的典型示例 (fCANMOD = 8 MHz) (1/2)

设置波特率值 (单位: kbps)	C0BRP 寄存 器的分频系 数	C0BRP 寄存器 设定值 TQPRS[7:0]	有效位速率设置 (单位: kbps)					C0BTR 寄存器设定值		采样点 (单 位: %)
			DBT 的长度	SYNC SEGMENT	PROP SEGMENT	PHASE SEGMENT1	PHASE SEGMENT2	TSEG13 至 TSEG10	TSEG22 至 TSEG20	
1000	1	00000000	8	1	1	3	3	0011	010	62.5
1000	1	00000000	8	1	3	2	2	0100	001	75.0
1000	1	00000000	8	1	5	1	1	0101	000	87.5
500	1	00000000	16	1	1	7	7	0111	110	56.3
500	1	00000000	16	1	3	6	6	1000	101	62.5
500	1	00000000	16	1	5	5	5	1001	100	68.8
500	1	00000000	16	1	7	4	4	1010	011	75.0
500	1	00000000	16	1	9	3	3	1011	010	81.3
500	1	00000000	16	1	11	2	2	1100	001	87.5
500	1	00000000	16	1	13	1	1	1101	000	93.8
500	2	00000001	8	1	1	3	3	0011	010	62.5
500	2	00000001	8	1	3	2	2	0100	001	75.0
500	2	00000001	8	1	5	1	1	0101	000	87.5
250	2	00000001	16	1	1	7	7	0111	110	56.3
250	2	00000001	16	1	3	6	6	1000	101	62.5
250	2	00000001	16	1	5	5	5	1001	100	68.8
250	2	00000001	16	1	7	4	4	1010	011	75.0
250	2	00000001	16	1	9	3	3	1011	010	81.3
250	2	00000001	16	1	11	2	2	1100	001	87.5
250	2	00000001	16	1	13	1	1	1101	000	93.8
250	4	00000011	8	1	3	2	2	0100	001	75.0
250	4	00000011	8	1	5	1	1	0101	000	87.5
125	4	00000011	16	1	1	7	7	0111	110	56.3
125	4	00000011	16	1	3	6	6	1000	101	62.5
125	4	00000011	16	1	5	5	5	1001	100	68.8
125	4	00000011	16	1	7	4	4	1010	011	75.0
125	4	00000011	16	1	9	3	3	1011	010	81.3
125	4	00000011	16	1	11	2	2	1100	001	87.5
125	4	00000011	16	1	13	1	1	1101	000	93.8
125	8	00000111	8	1	3	2	2	0100	001	75.0
125	8	00000111	8	1	5	1	1	0101	000	87.5

注意事项 表 20-22 中的值不保证网络系统的操作。 应将振荡偏差和 CAN 总线及 CAN 收发器的延时考虑在内，以彻底地检查在网络系统中的效果。

表 20-22. 波特率设置的典型示例 (fCANMOD = 8 MHz) (2/2)

设置波特率值 (单位: kbps)	C0BRP 寄存 器的分频系 数	C0BRP 寄存器 设定值 TQPRS[7:0]	有效位速率设置 (单位: kbps)					C0BTR 寄存器设定值		采样点 (单 位: %)
			DBT 的长度	SYNC SEGMENT	PROP SEGMENT	PHASE SEGMENT1	PHASE SEGMENT2	TSEG13 至 TSEG10	TSEG22 至 TSEG20	
100	4	00000011	20	1	7	6	6	1100	101	70.0
100	4	00000011	20	1	9	5	5	1101	100	75.0
100	5	00000100	16	1	7	4	4	1010	011	75.0
100	5	00000100	16	1	9	3	3	1011	010	81.3
100	8	00000111	10	1	3	3	3	0101	010	70.0
100	8	00000111	10	1	5	2	2	0110	001	80.0
100	10	00001001	8	1	3	2	2	0100	001	75.0
100	10	00001001	8	1	5	1	1	0101	000	87.5
83.3	4	00000011	24	1	7	8	8	1110	111	66.7
83.3	4	00000011	24	1	9	7	7	1111	110	70.8
83.3	6	00000101	16	1	5	5	5	1001	100	68.8
83.3	6	00000101	16	1	7	4	4	1010	011	75.0
83.3	6	00000101	16	1	9	3	3	1011	010	81.3
83.3	6	00000101	16	1	11	2	2	1100	001	87.5
83.3	8	00000111	12	1	5	3	3	0111	010	75.0
83.3	8	00000111	12	1	7	2	2	1000	001	83.3
83.3	12	00001011	8	1	3	2	2	0100	001	75.0
83.3	12	00001011	8	1	5	1	1	0101	000	87.5
33.3	10	00001001	24	1	7	8	8	1110	111	66.7
33.3	10	00001001	24	1	9	7	7	1111	110	70.8
33.3	12	00001011	20	1	7	6	6	1100	101	70.0
33.3	12	00001011	20	1	9	5	5	1101	100	75.0
33.3	15	00001110	16	1	7	4	4	1010	011	75.0
33.3	15	00001110	16	1	9	3	3	1011	010	81.3
33.3	16	00001111	15	1	6	4	4	1001	011	73.3
33.3	16	00001111	15	1	8	3	3	1010	010	80.0
33.3	20	00010011	12	1	5	3	3	0111	010	75.0
33.3	20	00010011	12	1	7	2	2	1000	001	83.3
33.3	24	00010111	10	1	3	3	3	0101	010	70.0
33.3	24	00010111	10	1	5	2	2	0110	001	80.0
33.3	30	00011101	8	1	3	2	2	0100	001	75.0
33.3	30	00011101	8	1	5	1	1	0101	000	87.5

注意事项 表 20-22 中的值不保证网络系统的操作。 应将振荡偏差和 CAN 总线及 CAN 收发器的延时考虑在内，以彻底地检查在网络系统中的效果。

表 20-23. 波特率设置的典型示例 (fCANMOD = 16 MHz) (1/2)

设置波特率值 (单位: kbps)	COBRP 寄存 器的分频系 数	COBRP 寄存器 设定值 TQPRS[7:0]	有效位速率设置 (单位: kbps)					COBTR 寄存器设定值		采样点 (单 位: %)
			DBT 的长度	SYNC SEGMENT	PROP SEGMENT	PHASE SEGMENT1	PHASE SEGMENT2	TSEG13 至 TSEG10	TSEG22 至 TSEG20	
1000	1	00000000	16	1	1	7	7	0111	110	56.3
1000	1	00000000	16	1	3	6	6	1000	101	62.5
1000	1	00000000	16	1	5	5	5	1001	100	68.8
1000	1	00000000	16	1	7	4	4	1010	011	75.0
1000	1	00000000	16	1	9	3	3	1011	010	81.3
1000	1	00000000	16	1	11	2	2	1100	001	87.5
1000	1	00000000	16	1	13	1	1	1101	000	93.8
1000	2	00000001	8	1	3	2	2	0100	001	75.0
1000	2	00000001	8	1	5	1	1	0101	000	87.5
500	2	00000001	16	1	1	7	7	0111	110	56.3
500	2	00000001	16	1	3	6	6	1000	101	62.5
500	2	00000001	16	1	5	5	5	1001	100	68.8
500	2	00000001	16	1	7	4	4	1010	011	75.0
500	2	00000001	16	1	9	3	3	1011	010	81.3
500	2	00000001	16	1	11	2	2	1100	001	87.5
500	2	00000001	16	1	13	1	1	1101	000	93.8
500	4	00000011	8	1	3	2	2	0100	001	75.0
500	4	00000011	8	1	5	1	1	0101	000	87.5
250	4	00000011	16	1	3	6	6	1000	101	62.5
250	4	00000011	16	1	5	5	5	1001	100	68.8
250	4	00000011	16	1	7	4	4	1010	011	75.0
250	4	00000011	16	1	9	3	3	1011	010	81.3
250	4	00000011	16	1	11	2	2	1100	001	87.5
250	8	00000111	8	1	3	2	2	0100	001	75.0
250	8	00000111	8	1	5	1	1	0101	000	87.5
125	8	00000111	16	1	3	6	6	1000	101	62.5
125	8	00000111	16	1	7	4	4	1010	011	75.0
125	8	00000111	16	1	9	3	3	1011	010	81.3
125	8	00000111	16	1	11	2	2	1100	001	87.5
125	16	00001111	8	1	3	2	2	0100	001	75.0
125	16	00001111	8	1	5	1	1	0101	000	87.5

注意事项 表 20-23 中的值不保证网络系统的操作。 应将振荡偏差和 CAN 总线及 CAN 收发器的延时考虑在内, 以彻底地检查在网络系统中的效果。

表 20-23. 波特率设置的典型示例 ($f_{\text{CANMOD}} = 16 \text{ MHz}$) (2/2)

设置波特率值 (单位: kbps)	COBRP 寄存 器的分频系 数	COBRP 寄存器 设定值 TQPRS[7:0]	有效位速率设置 (单位: kbps)					COBTR 寄存器设定值		采样点 (单 位: %)
			DBT 的长度	SYNC SEGMENT	PROP SEGMENT	PHASE SEGMENT1	PHASE SEGMENT2	TSEG13 至 TSEG10	TSEG22 至 TSEG20	
100	8	00000111	20	1	9	5	5	1101	100	75.0
100	8	00000111	20	1	11	4	4	1110	011	80.0
100	10	00001001	16	1	7	4	4	1010	011	75.0
100	10	00001001	16	1	9	3	3	1011	010	81.3
100	16	00001111	10	1	3	3	3	0101	010	70.0
100	16	00001111	10	1	5	2	2	0110	001	80.0
100	20	00010011	8	1	3	2	2	0100	001	75.0
83.3	8	00000111	24	1	7	8	8	1110	111	66.7
83.3	8	00000111	24	1	9	7	7	1111	110	70.8
83.3	12	00001011	16	1	7	4	4	1010	011	75.0
83.3	12	00001011	16	1	9	3	3	1011	010	81.3
83.3	12	00001011	16	1	11	2	2	1100	001	87.5
83.3	16	00001111	12	1	5	3	3	0111	010	75.0
83.3	16	00001111	12	1	7	2	2	1000	001	83.3
83.3	24	00010111	8	1	3	2	2	0100	001	75.0
83.3	24	00010111	8	1	5	1	1	0101	000	87.5
33.3	30	00011101	24	1	7	8	8	1110	111	66.7
33.3	30	00011101	24	1	9	7	7	1111	110	70.8
33.3	24	00010111	20	1	9	5	5	1101	100	75.0
33.3	24	00010111	20	1	11	4	4	1110	011	80.0
33.3	30	00011101	16	1	7	4	4	1010	011	75.0
33.3	30	00011101	16	1	9	3	3	1011	010	81.3
33.3	32	00011111	15	1	8	3	3	1010	010	80.0
33.3	32	00011111	15	1	10	2	2	1011	001	86.7
33.3	37	00100100	13	1	6	3	3	1000	010	76.9
33.3	37	00100100	13	1	8	2	2	1001	001	84.6
33.3	40	00100111	12	1	5	3	3	0111	010	75.0
33.3	40	00100111	12	1	7	2	2	1000	001	83.3
33.3	48	00101111	10	1	3	3	3	0101	010	70.0
33.3	48	00101111	10	1	5	2	2	0110	001	80.0
33.3	60	00111011	8	1	3	2	2	0100	001	75.0
33.3	60	00111011	8	1	5	1	1	0101	000	87.5

注意事项 表 20-23 中的值不保证网络系统的操作。 应将振荡偏差和 CAN 总线及 CAN 收发器的延时考虑在内，以彻底地检查在网络系统中的效果。

20.16 CAN控制器的操作

备注 m = 00 至 31

图 20-35. 初始化

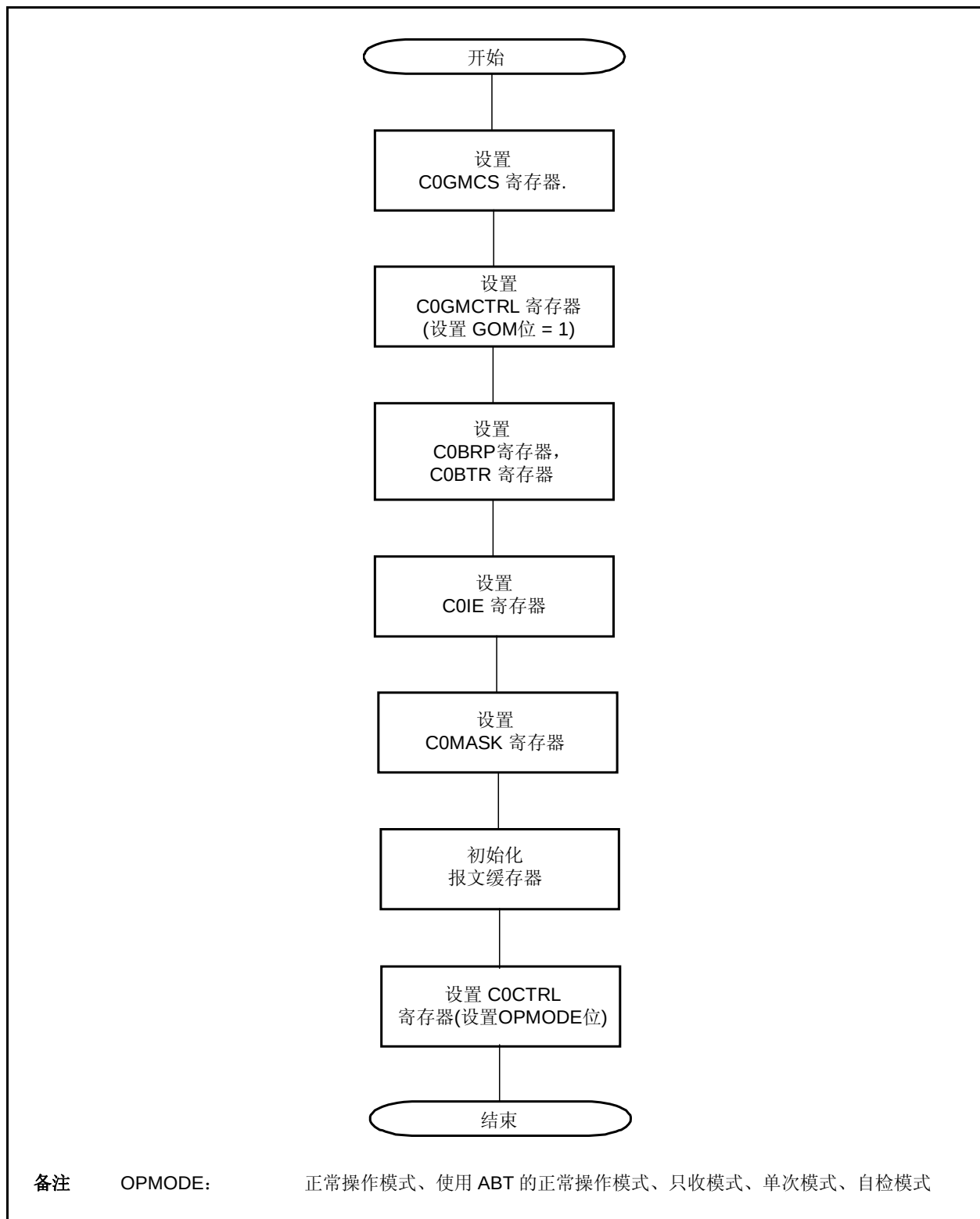
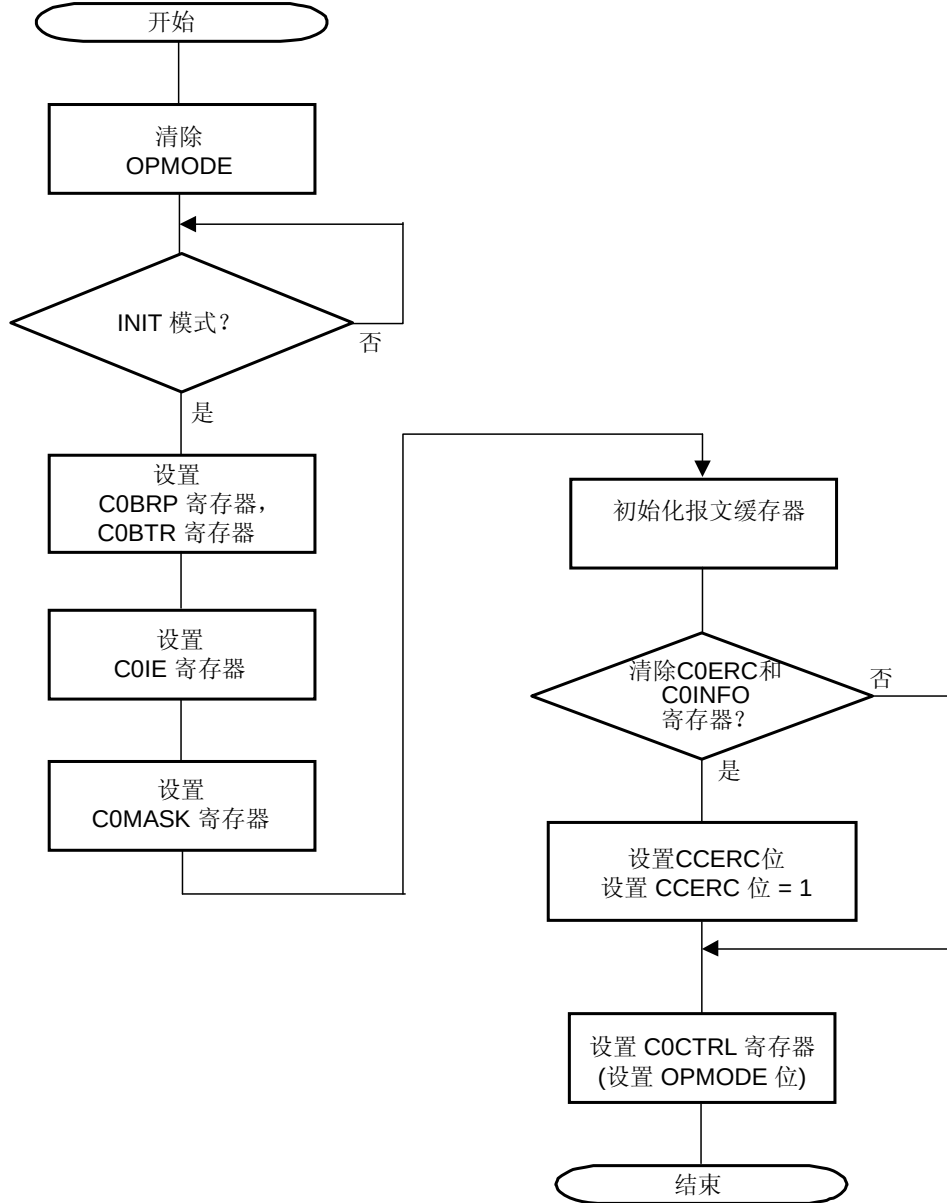


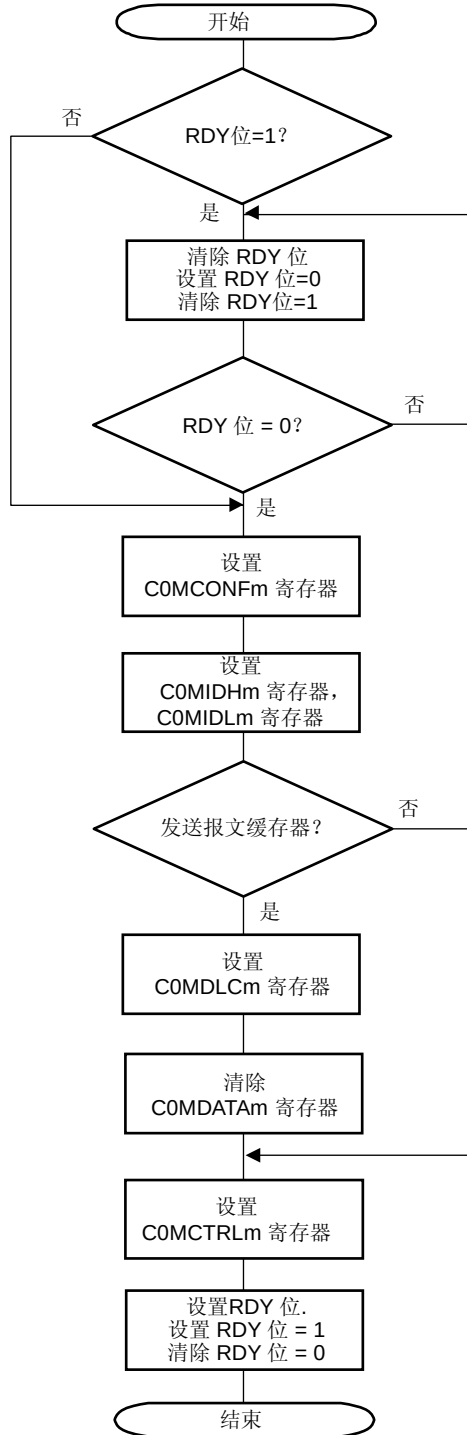
图 20-36. 重新初始化



注意事项 将 CAN 模块设为初始化模式后，应避免之后立刻将模块设为另外的操作模式。如果有必要将模块立即设为其它的操作模式，则务必访问除了 C0CTRL 和 C0GMCTRL 寄存器之外的寄存器（例如，设置报文缓存器）。

备注 OPMODE: 正常操作模式、使用 ABT 的正常操作模式、只收模式、单次模式、自检模式

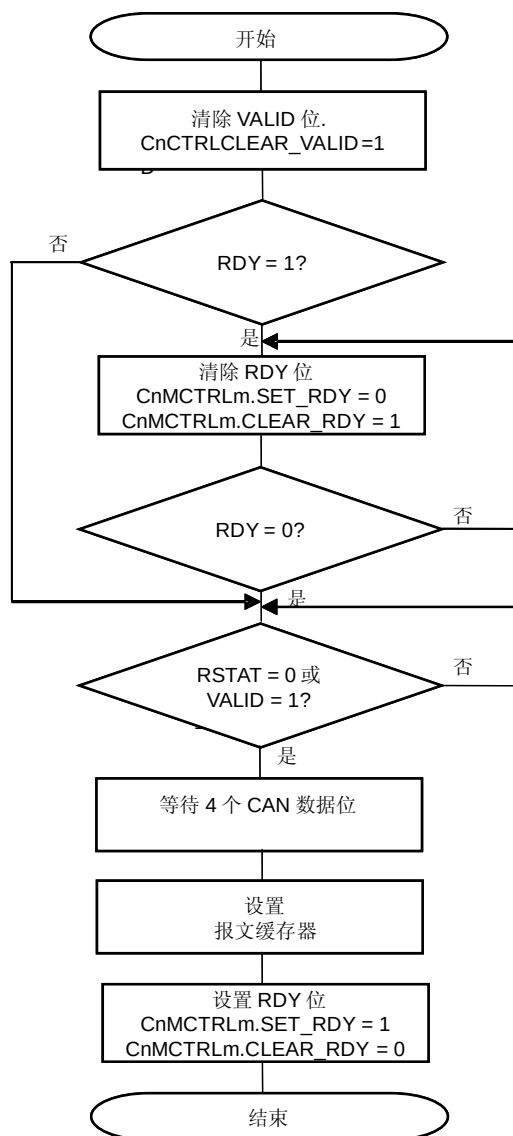
图 20-37. 报文缓存器初始化



- 注意事项
1. 必须在报文缓存器初始化之前清除 RDY 位。
 2. 对应用系统中不使用的报文缓存器进行以下设置。
 - 将 COMCTRLm.RDY、COMCTRLm.TRQ 和 COMCTRLm.DN 位清为 0。
 - 将 COMCONFm.MA0 位清为 0。

图 20-38 显示了接收报文缓存器的处理（COMCONFm.MT2 至 COMCONFm.MT0 位 = 001B 至 101B）。

图 20-38. 报文缓存器重新定义



注意事项 如果在报文接收期间进行重新定义，应确认报文正在进行接收，因为报文完全接收之后必须重新设置 RDY 位（为 1）。

图 20-39 显示了发送报文缓存器发送期间的处理（COMCONFm 寄存器的 MT2 至 MT0 位= 000B）。

图 20-39. 发送期间的报文缓存器重新定义

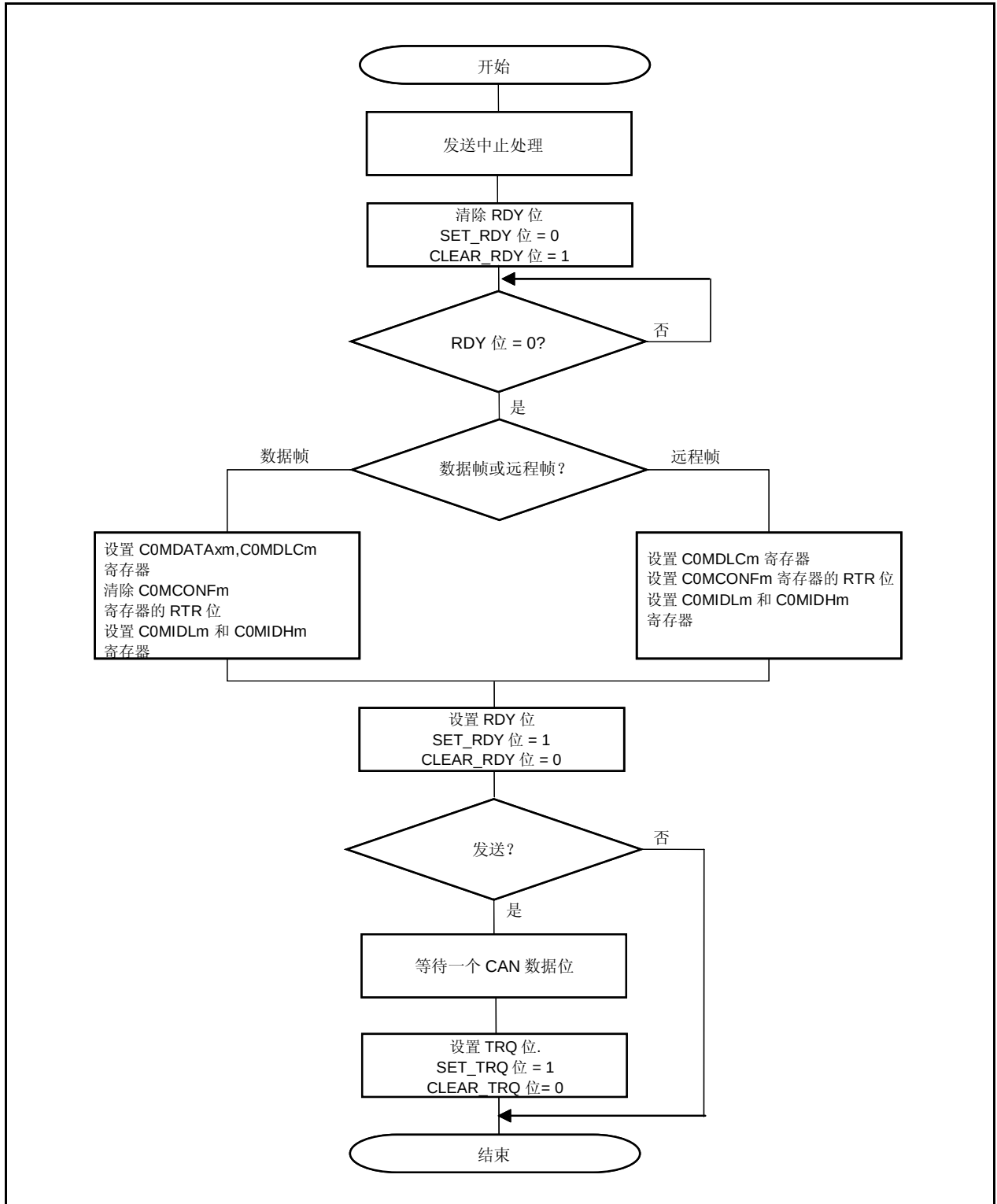


图 20-40 显示了发送报文缓存器的处理（COMCONFm.MT2 至 COMCONFm.MT0 位 = 000B）。

图 20-40. 报文发送处理（正常操作模式）

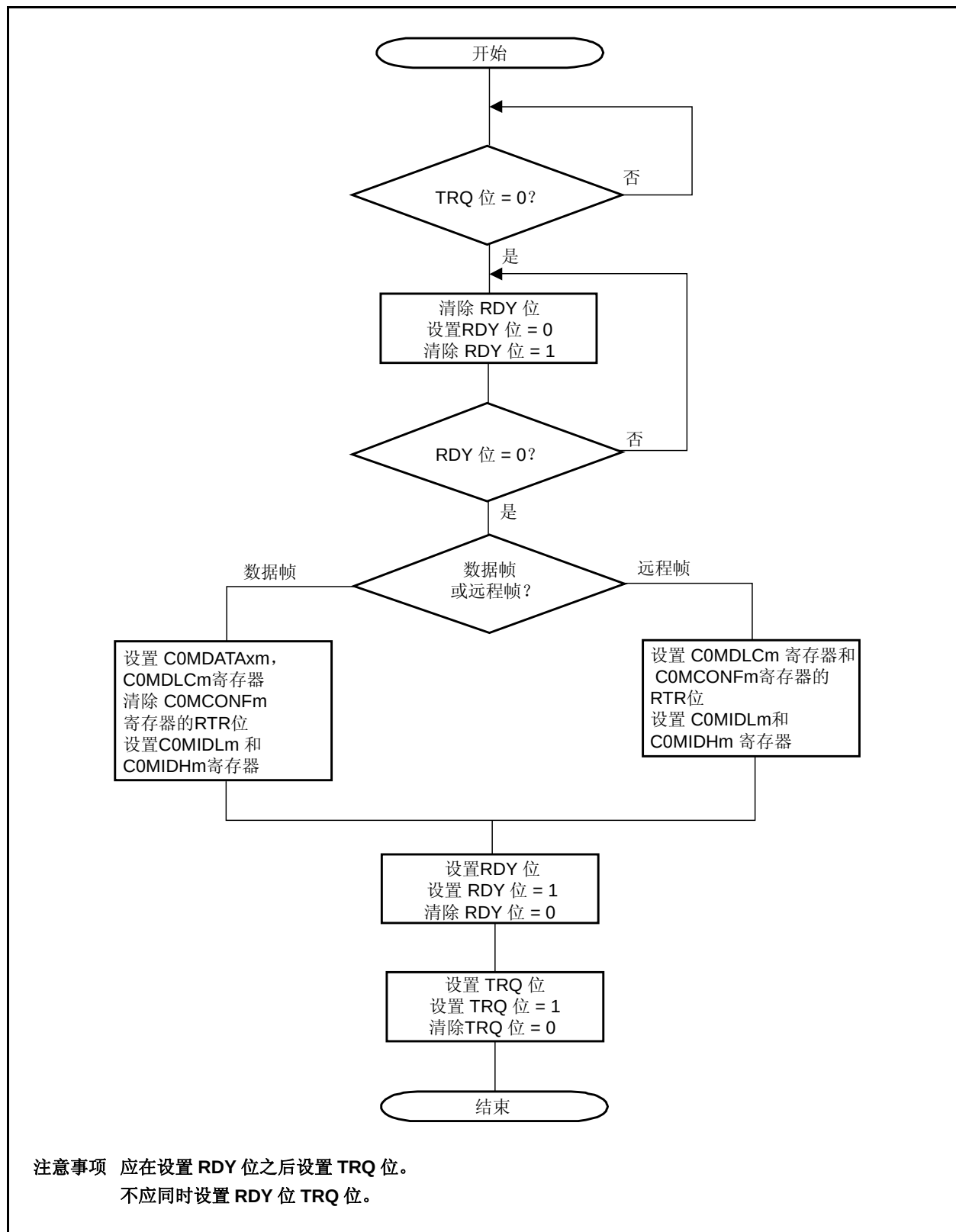


图 20-41 显示了发送报文缓存器的处理（COMCONFm.MT2 至 COMCONFm.MT0 位 = 000B）。

图 20-41. ABT 报文发送处理

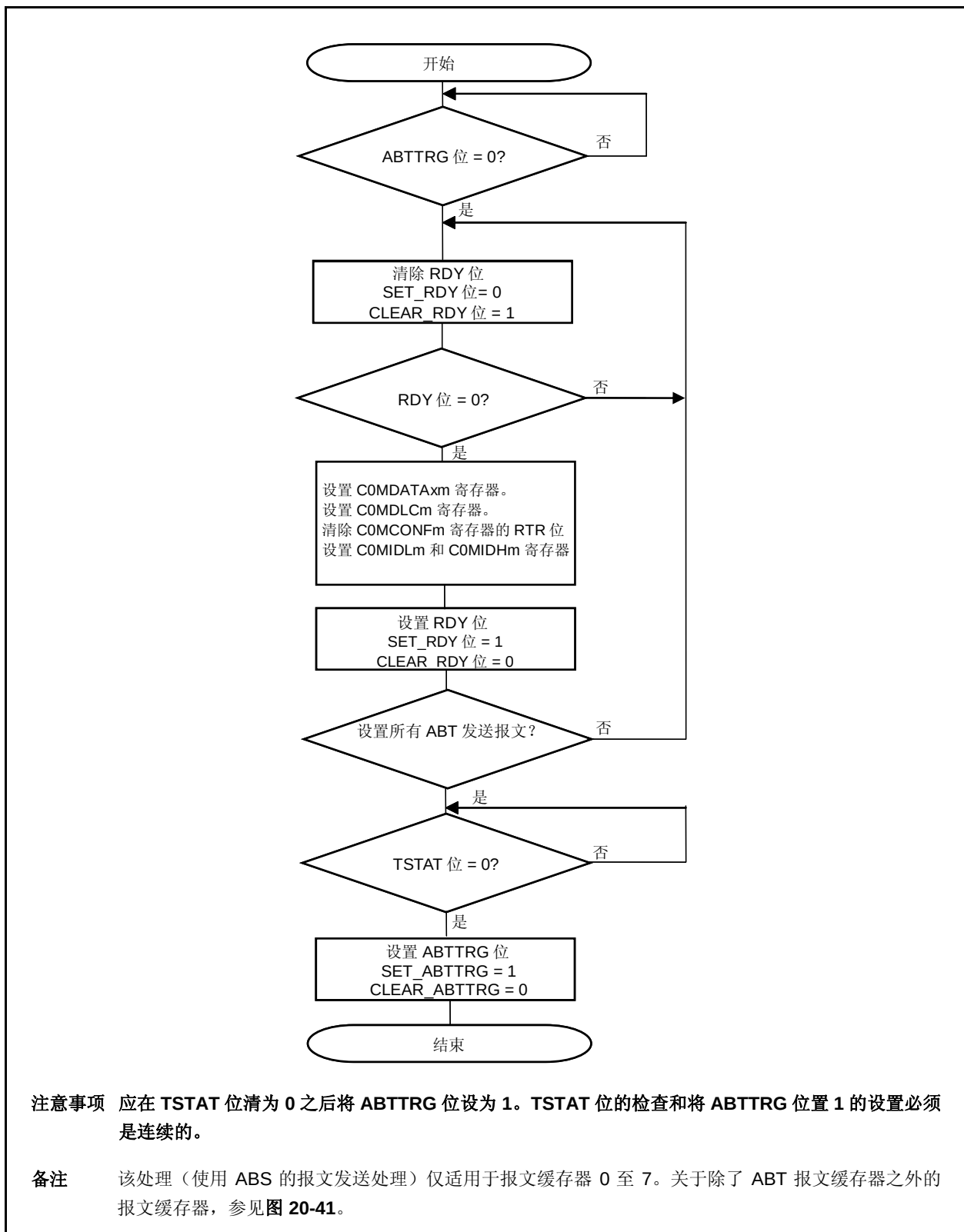


图 20-42. 通过中断发送（使用 CnLOPT 寄存器）

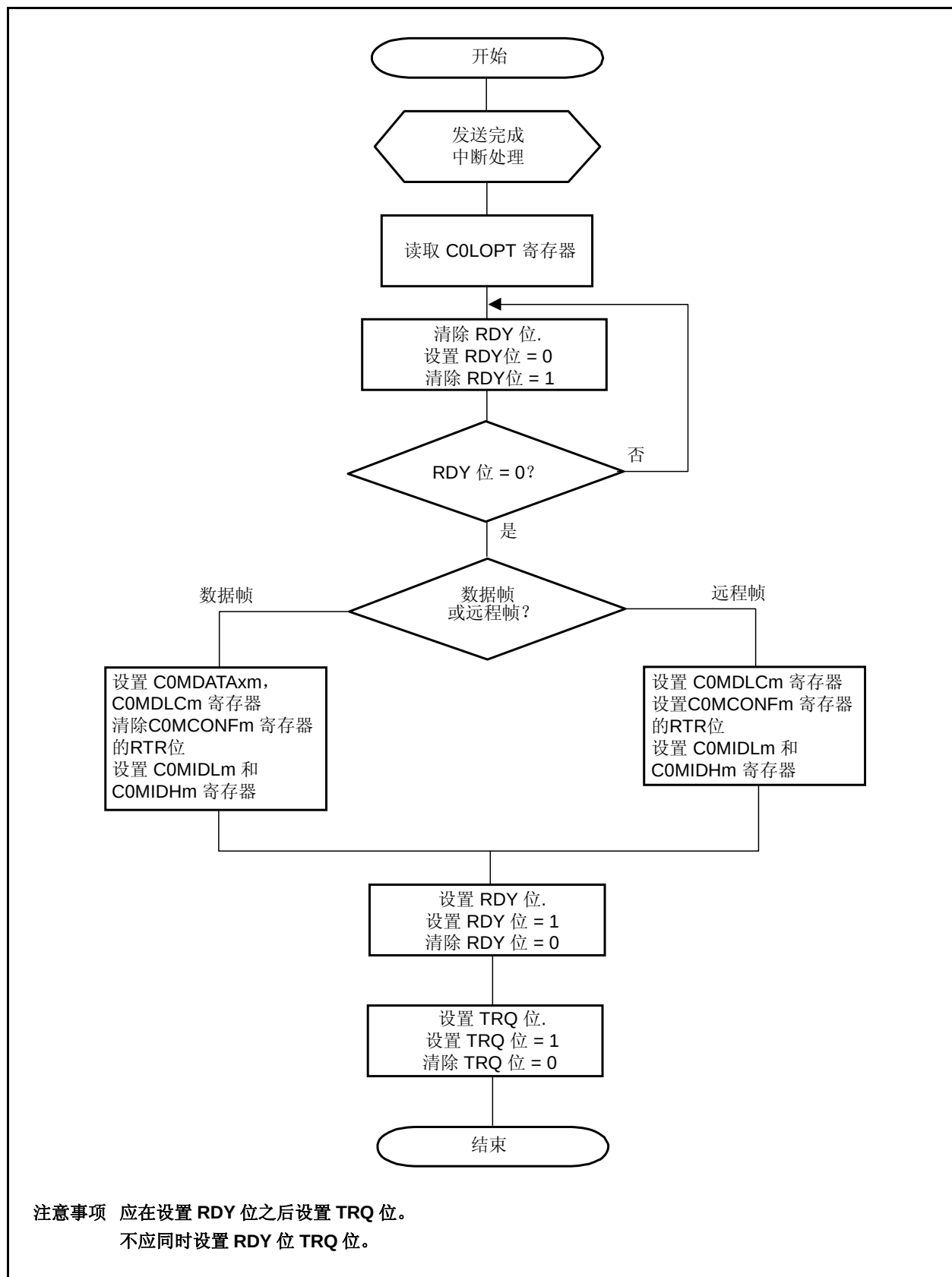


图 20-43. 通过中断发送（使用 CnLOPT 寄存器）

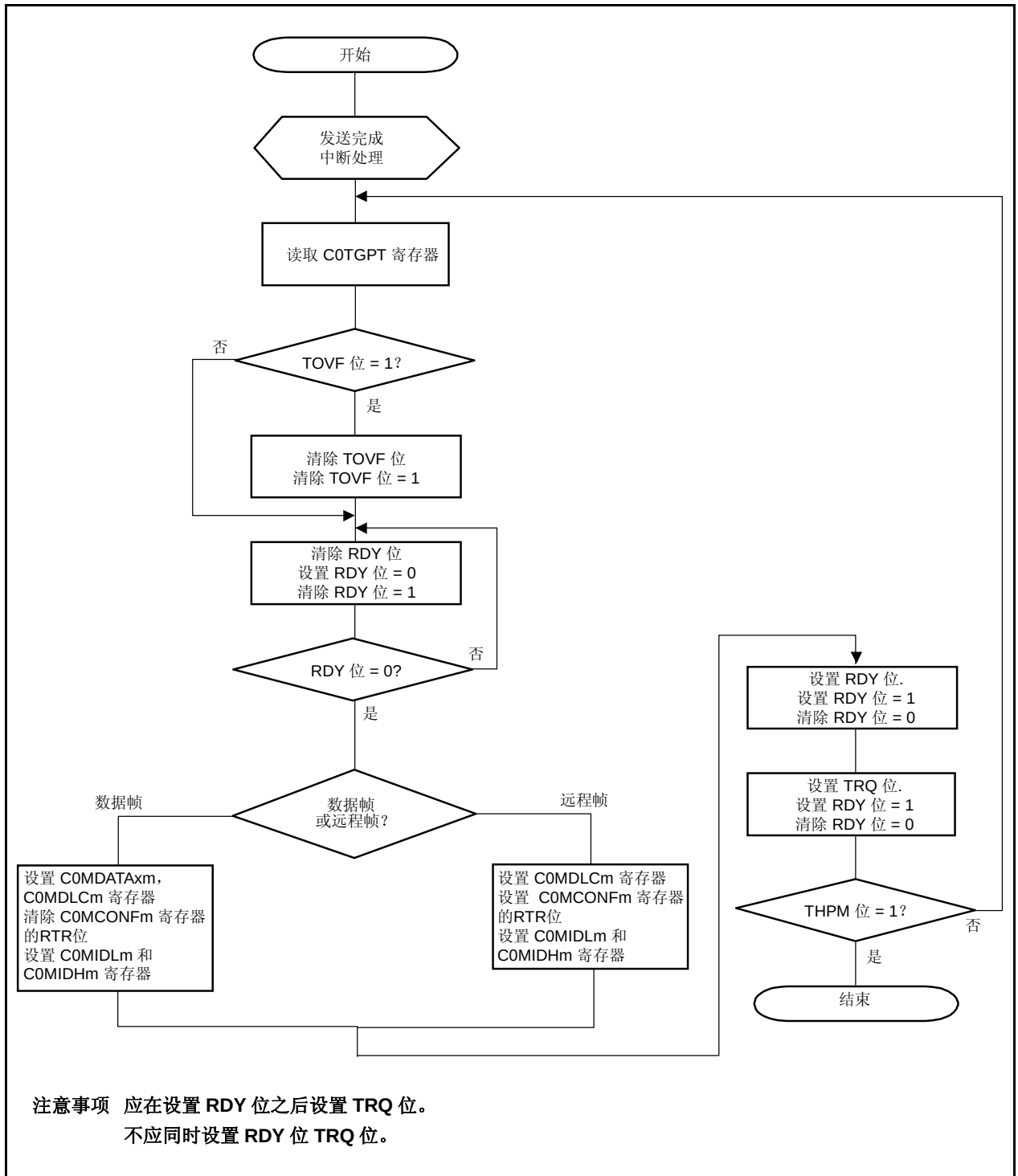


图 20-44. 通过软件查询发送

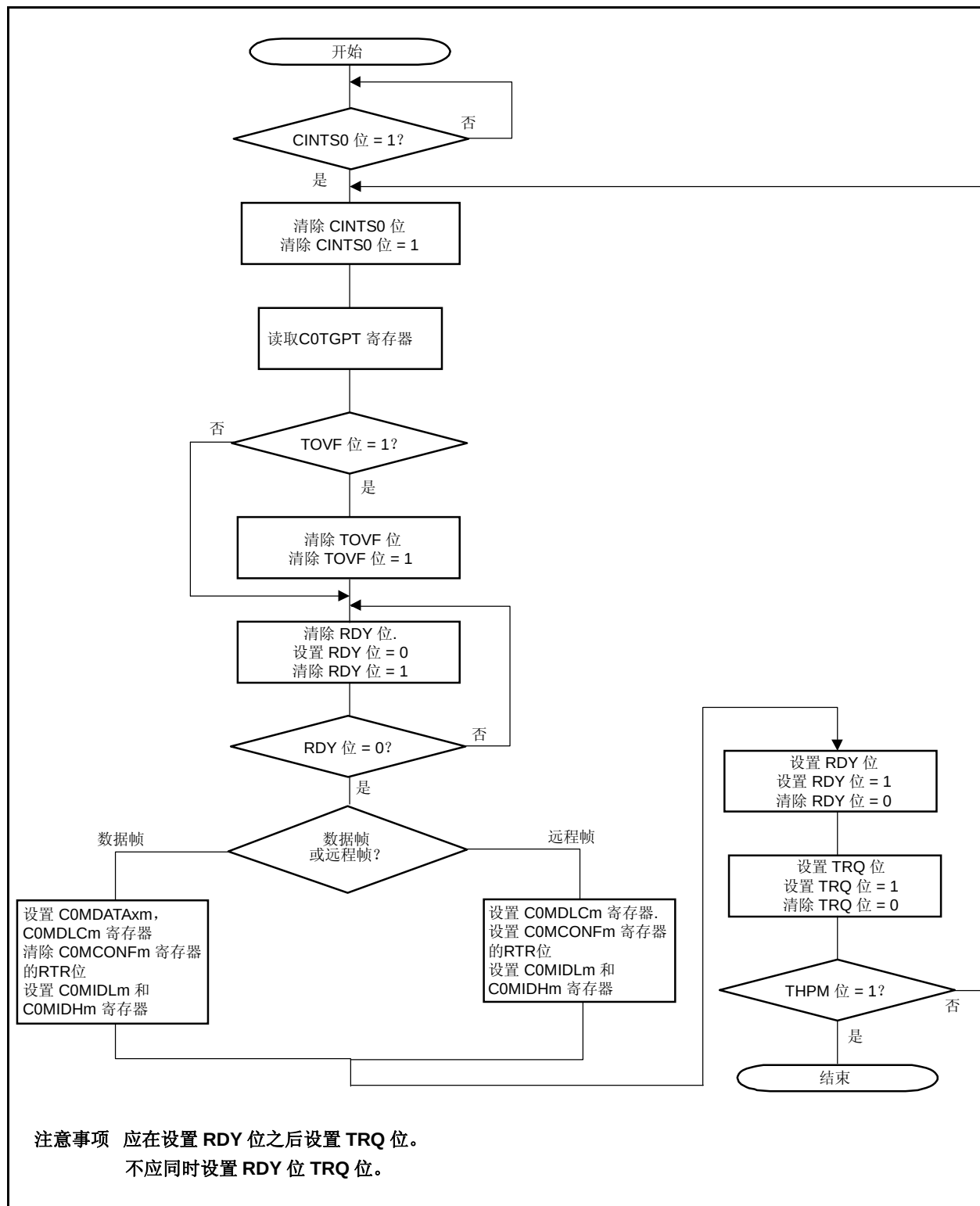


图 20-45. 报文发送处理（正常操作模式）

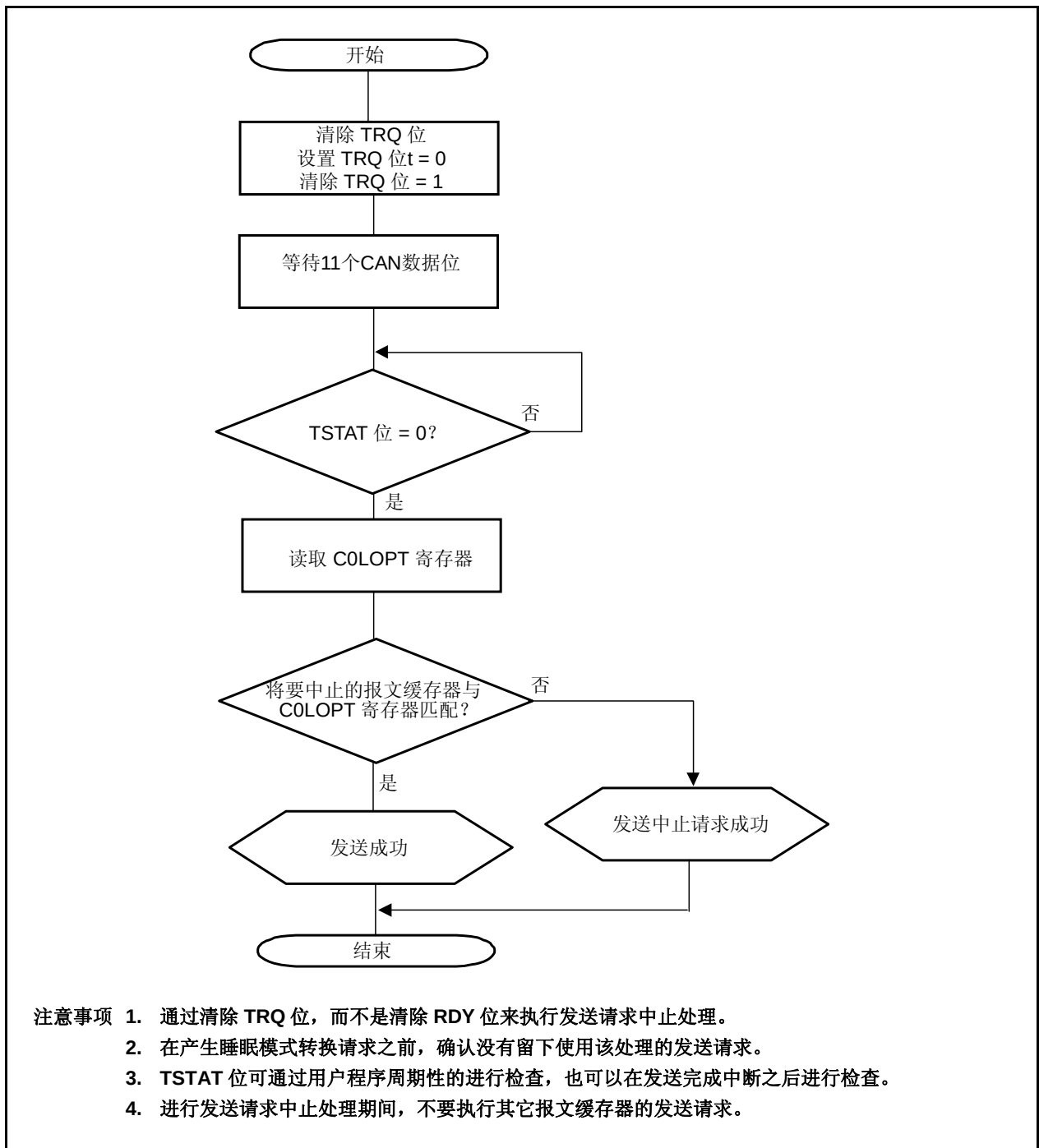
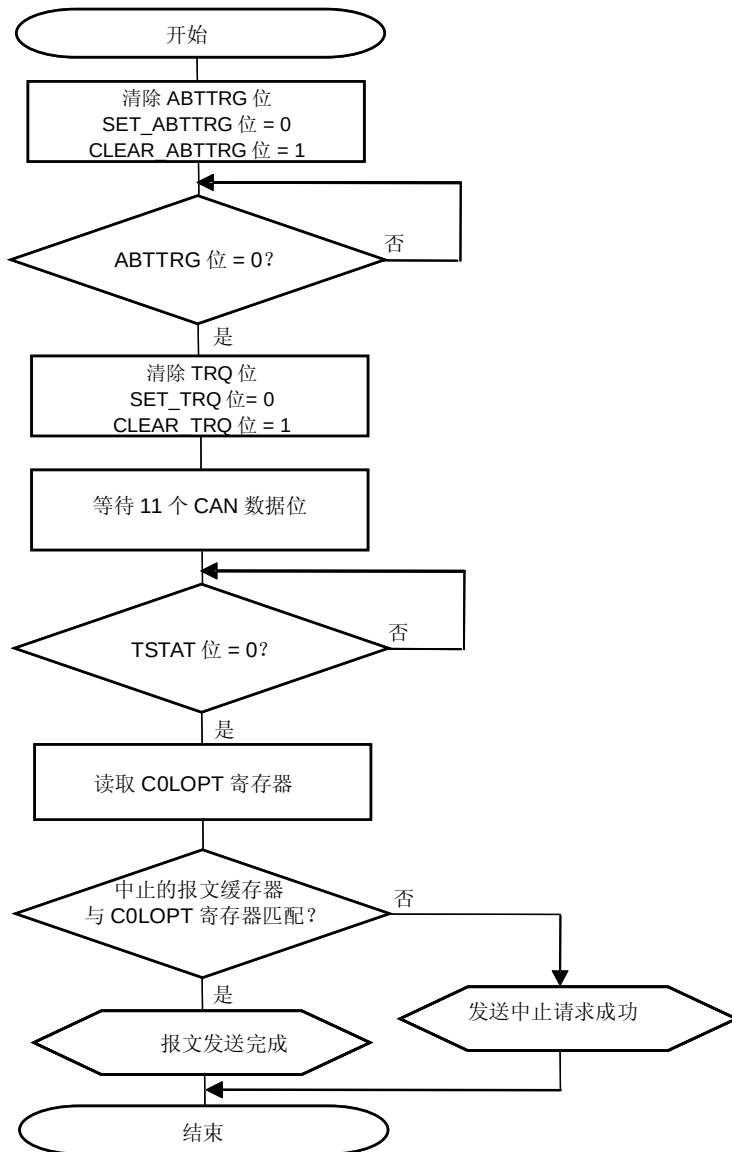


图 20-46. 除了 ABT 发送之外的发送中止处理
(使用 ABT 的正常操作模式)



- 注意事项
1. 通过清除 TRQ 位，而不是清除 RDY 位来执行发送中止处理。
 2. 在产生睡眠模式转换请求之前，确认没有留下使用该处理的发送请求。
 3. TSTAT 位可通过用户程序周期性的进行检查，也可以在发送完成中断之后进行检查。
 4. 进行发送请求中止处理期间，不要执行其它报文缓存器的发送请求。

图 20-47 (a) 显示了 ABT 报文缓存器发送中止时的处理举例，此处理不会跳过被中断的报文发送的继续执行。

图 20-47 (a). 发送中止处理（使用 ABT 的正常操作模式）

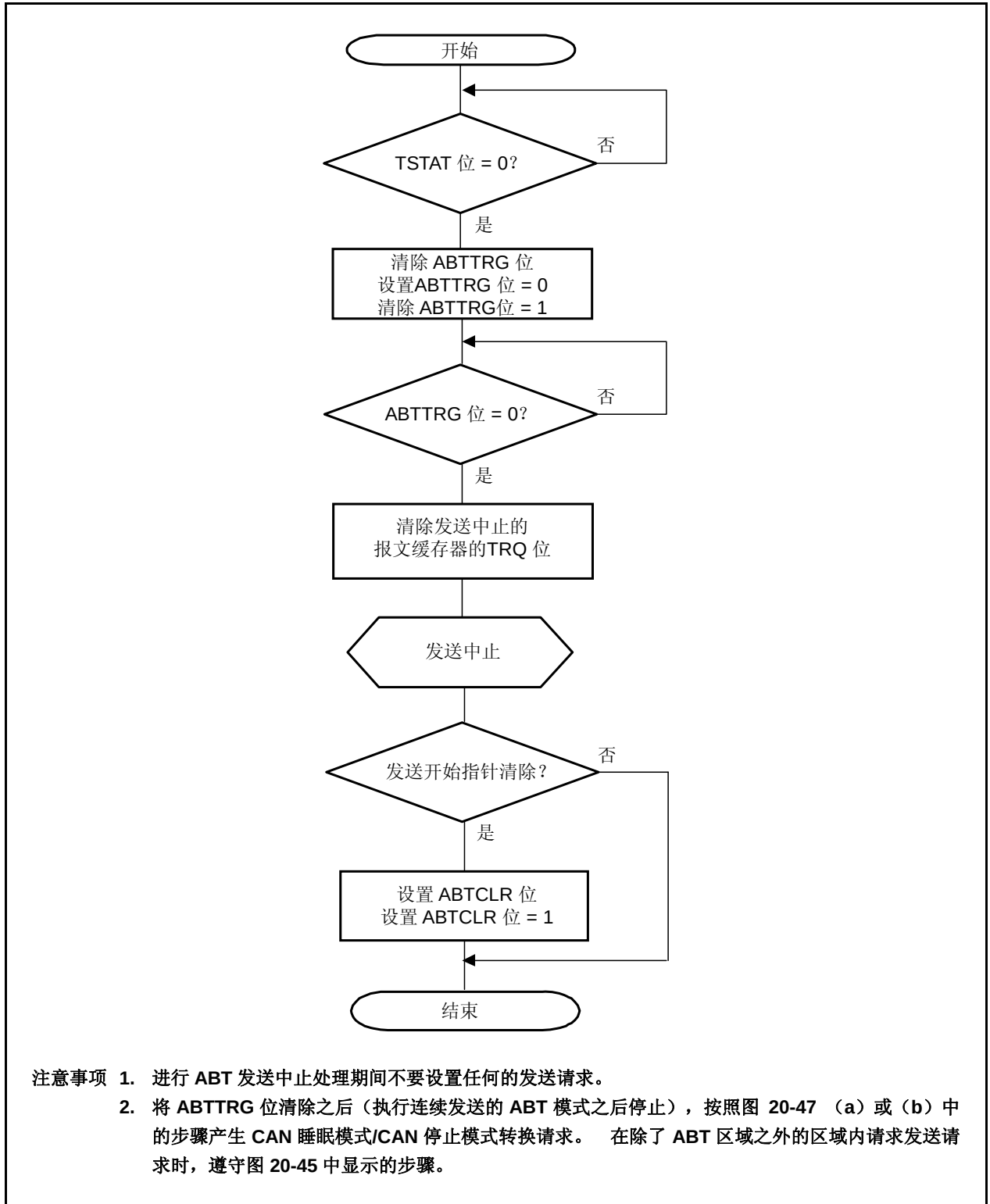
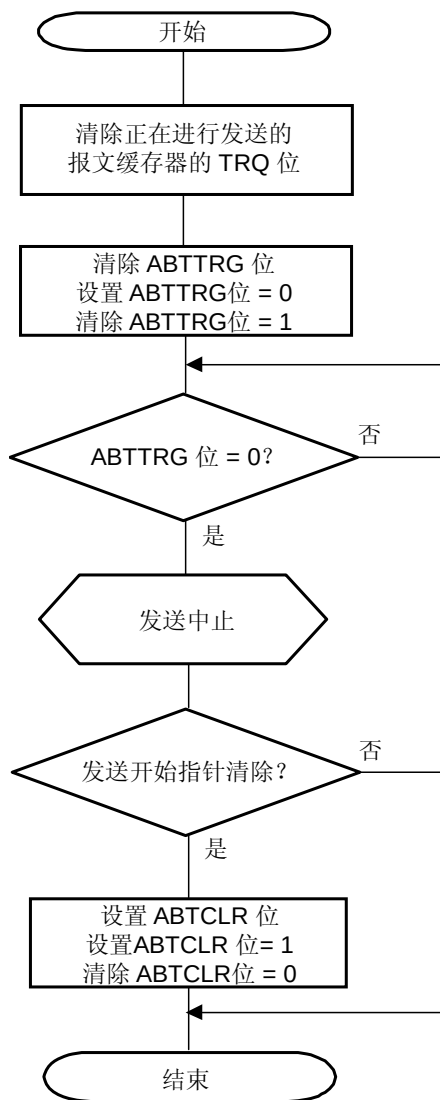


图 20-47 (a) 显示了 ABT 报文缓存器发送中止时的处理举例，此处理不会跳过被中断的报文发送的继续执行。

图 20-47 (b). 发送中止处理（使用 ABT 的正常操作模式）



注意事项 1. 进行 ABT 发送中止处理期间不要设置任何的发送请求。

2. 将 ABTTRG 位清除之后（执行连续发送的 ABT 模式之后停止），按照图 20-47 (a) 或 (b) 中的步骤产生 CAN 睡眠模式/CAN 停止模式转换请求。在除了 ABT 区域之外的区域内请求发送请求时，遵守图 20-45 中显示的步骤。

图 20-48. 通过中断接收（使用 C0LOPT 寄存器）

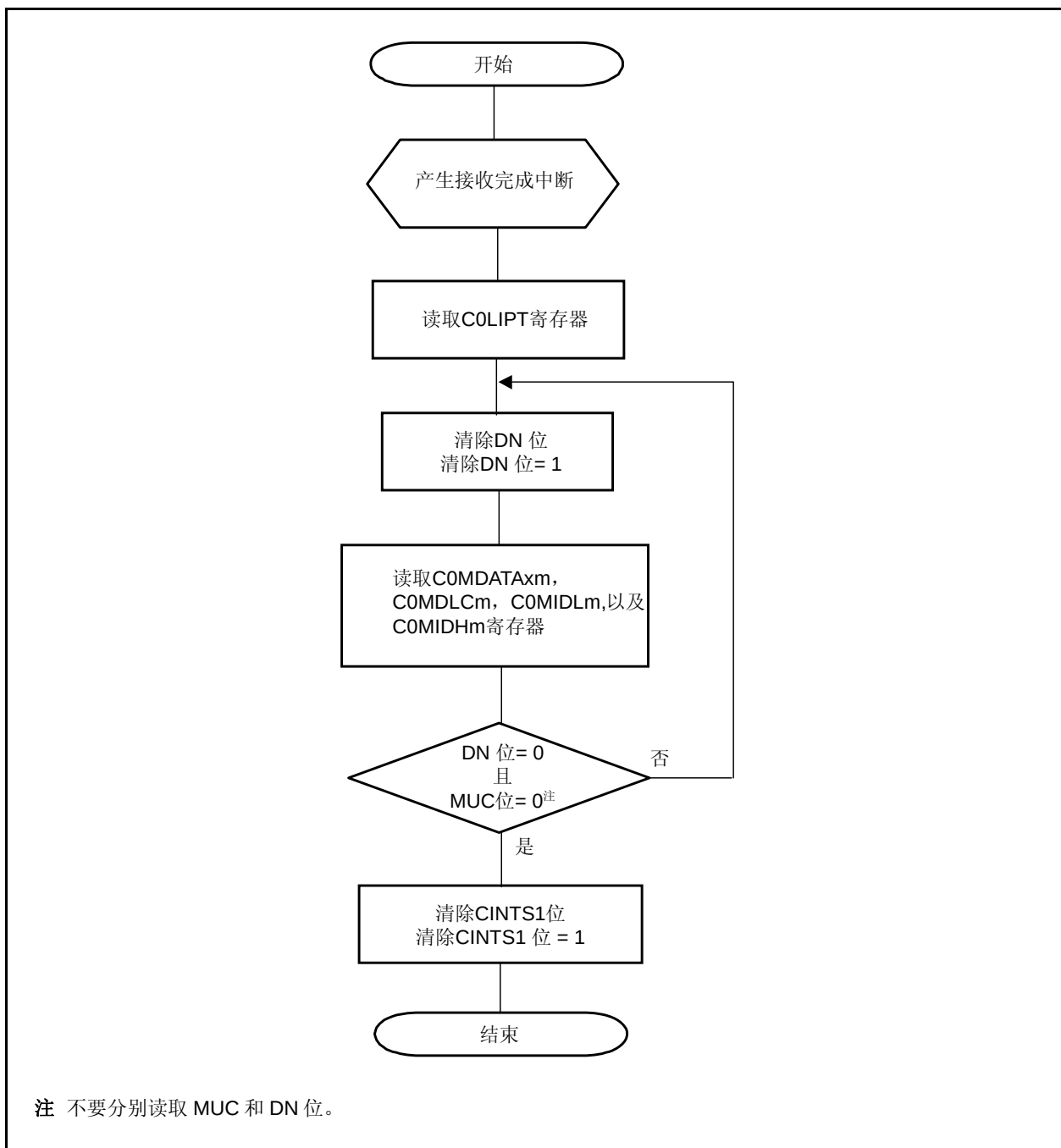


图 20-49. 通过中断接收（使用 C0RGPT 寄存器）

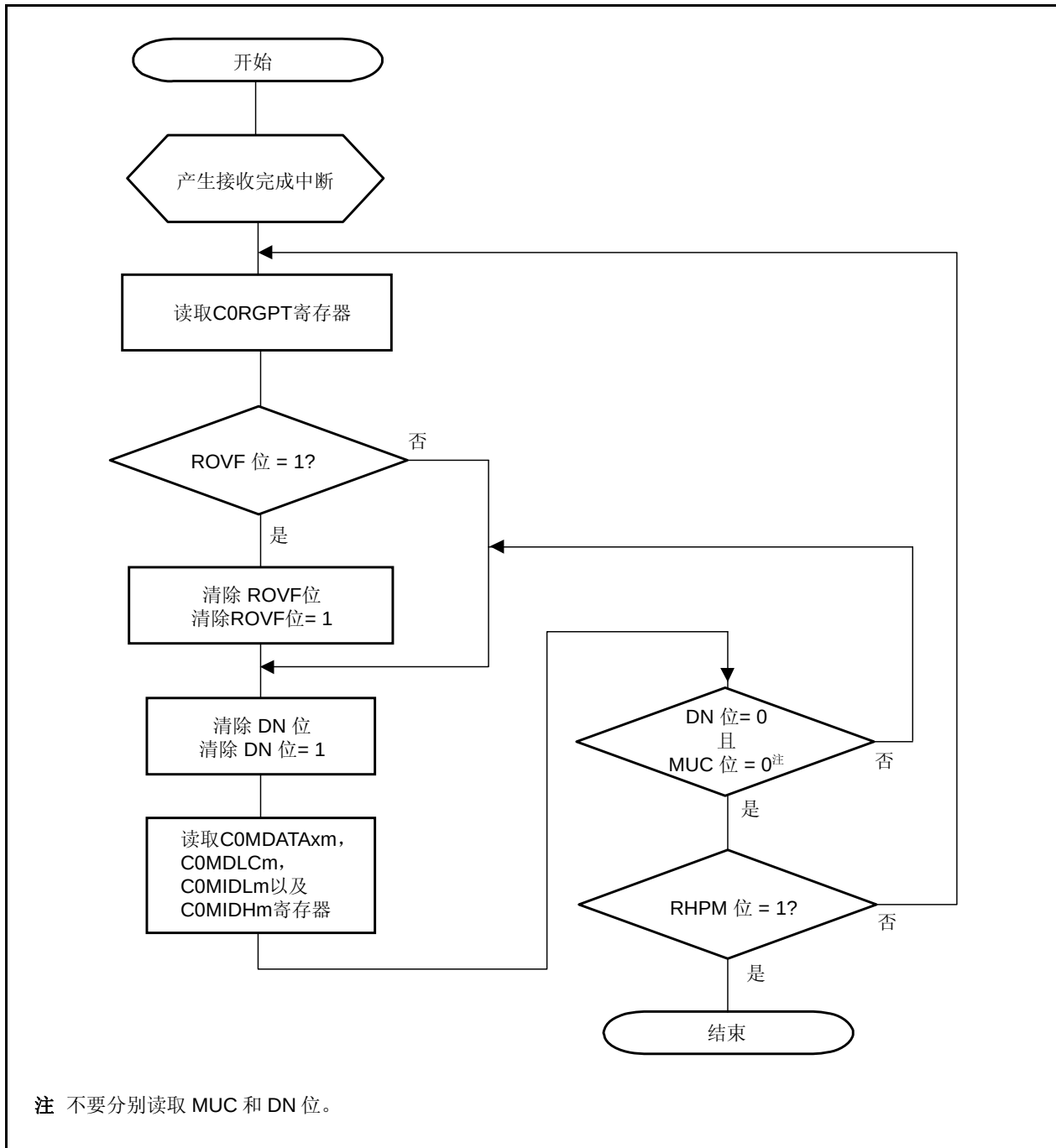


图 20-50. 通过软件查询接收

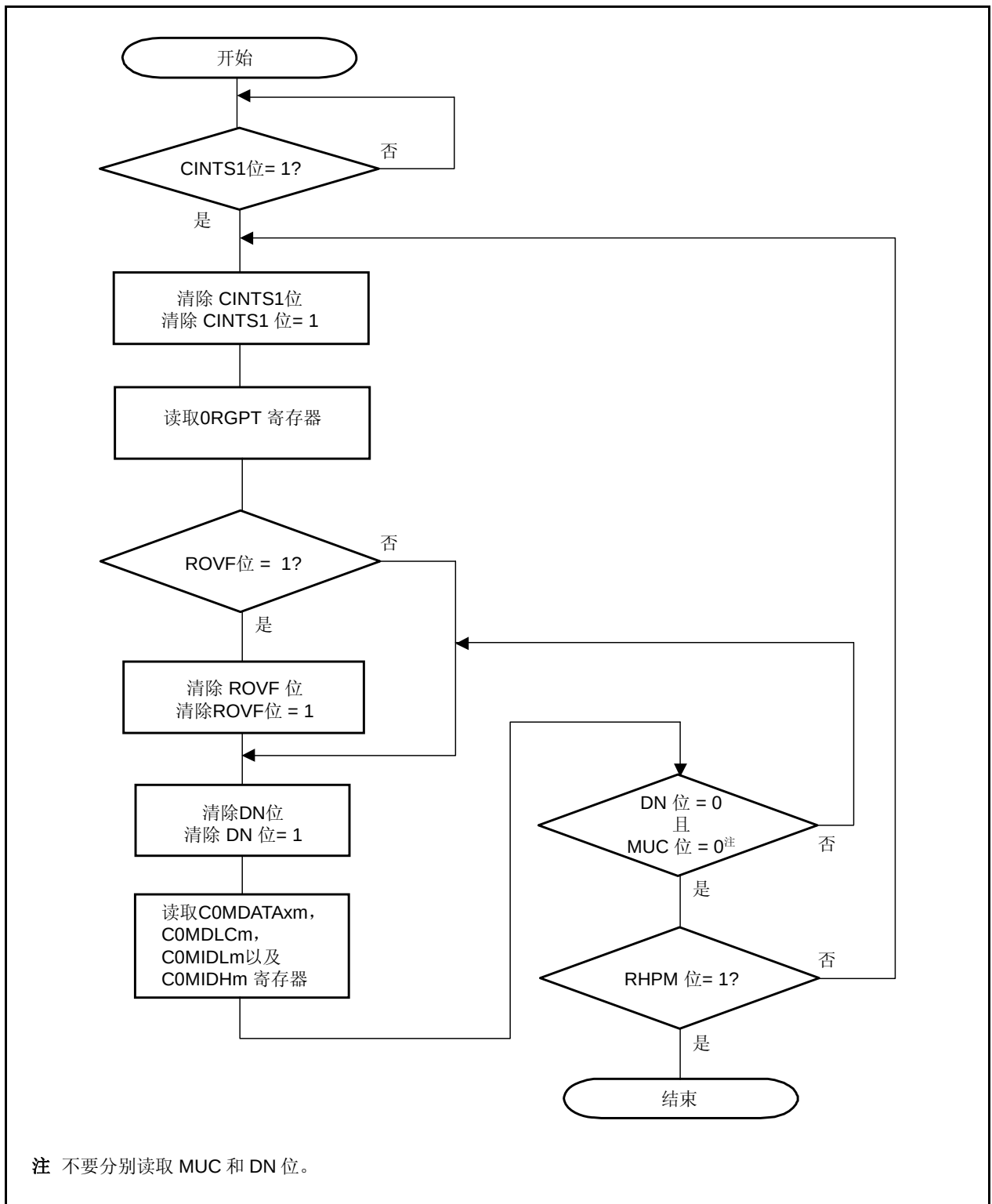
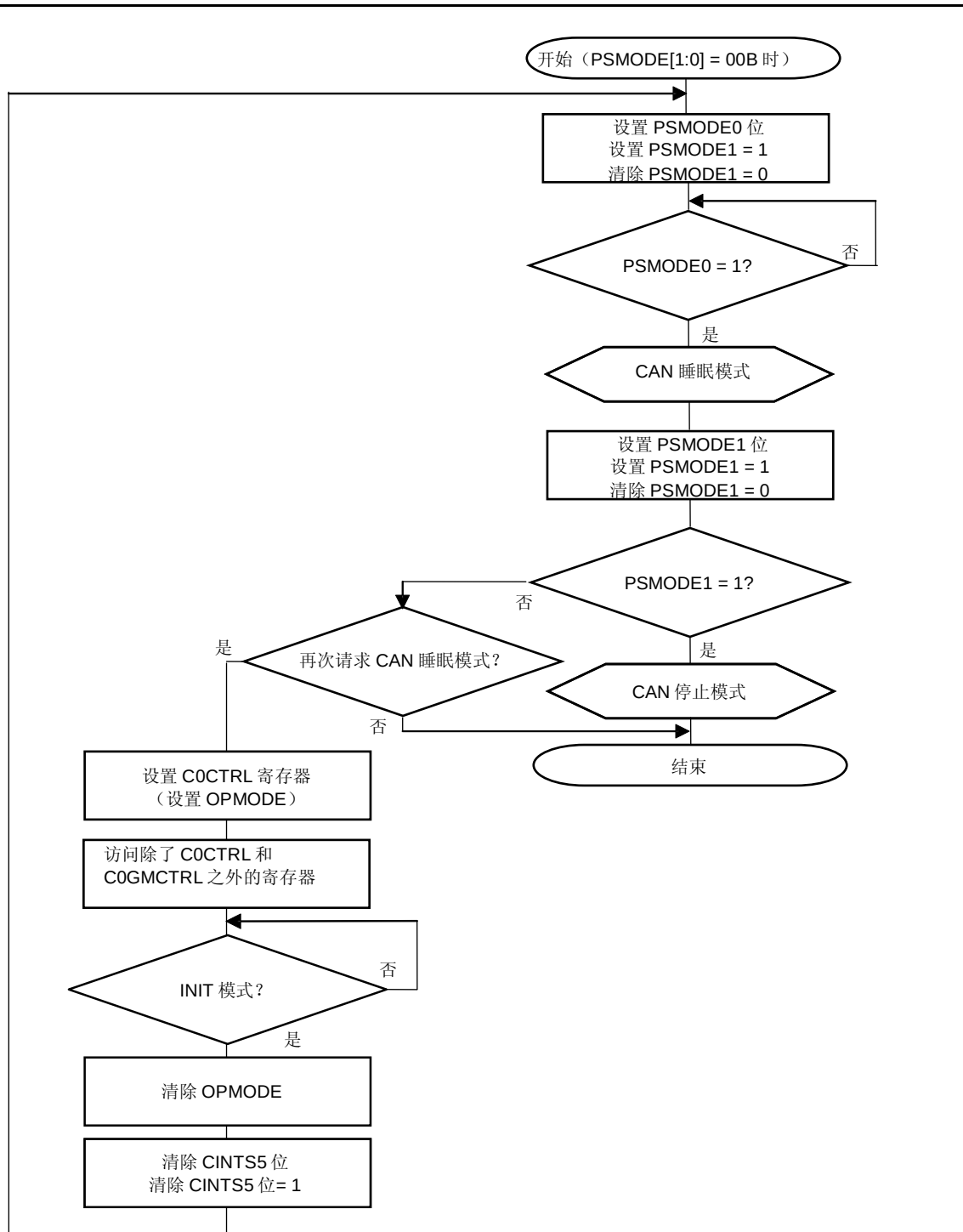


图 20-51. 设置 CAN 睡眠模式/停止模式



- 注意事项
1. 为了在产生 CAN 睡眠模式请求之前中止发送，依照图 20-45 至图 20-47 进行处理。
 2. 要将 CPU 设为节电模式，中断服务需在 CPU 检查是否进入了睡眠模式之前取消。如果没有取消中断服务且在检查是否进入睡眠模式与执行 CPU 待机处理之间解除 CAN 睡眠模式（例如，执行 HALT 指令），则在后来的 CPU 节电模式中不能通过 CAN 总线的改变唤醒 CPU。

图 20-52. 解除 CAN 睡眠/停止模式

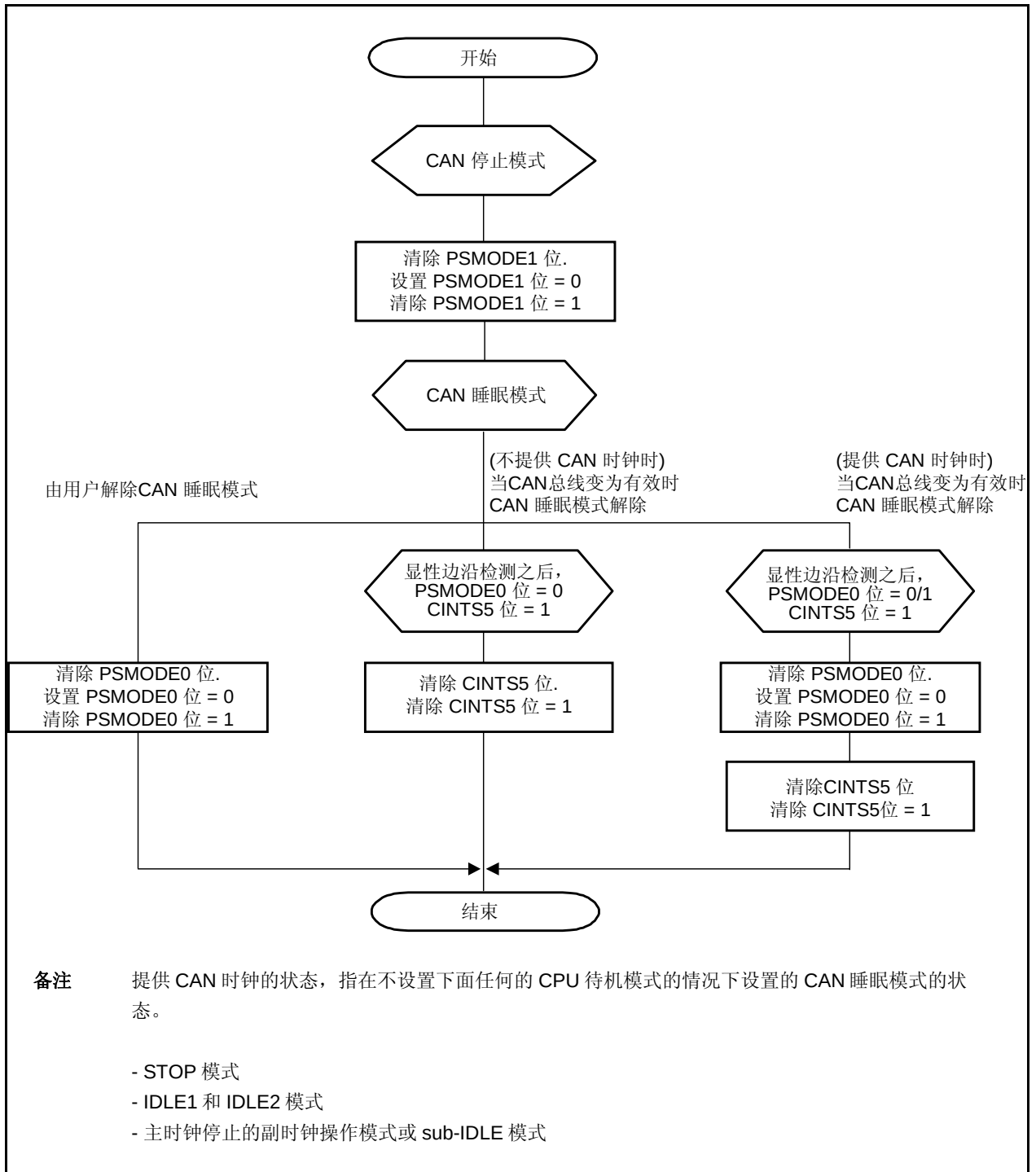


图 20-53. 掉线恢复处理

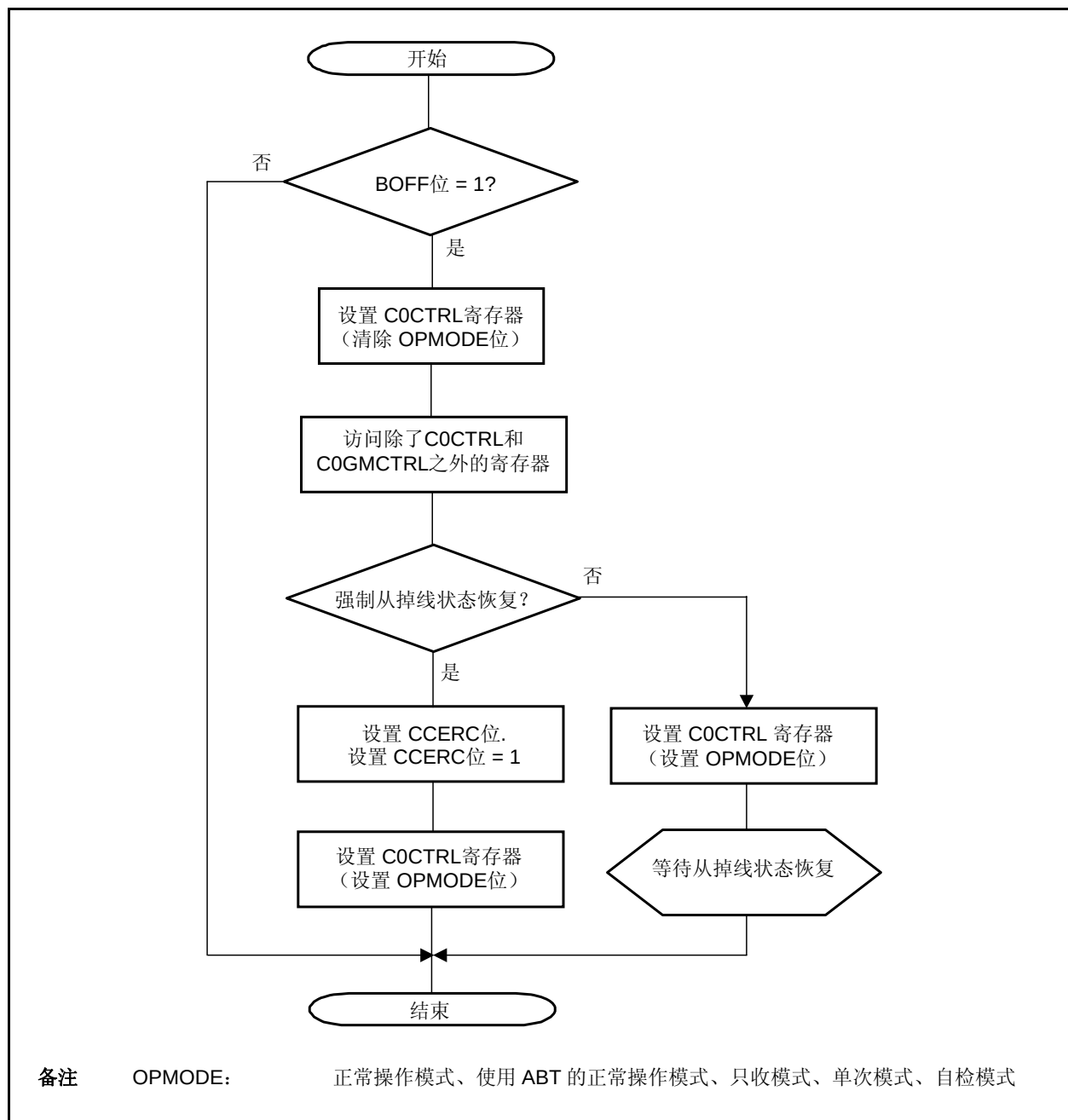


图 20-54. 正常关机处理

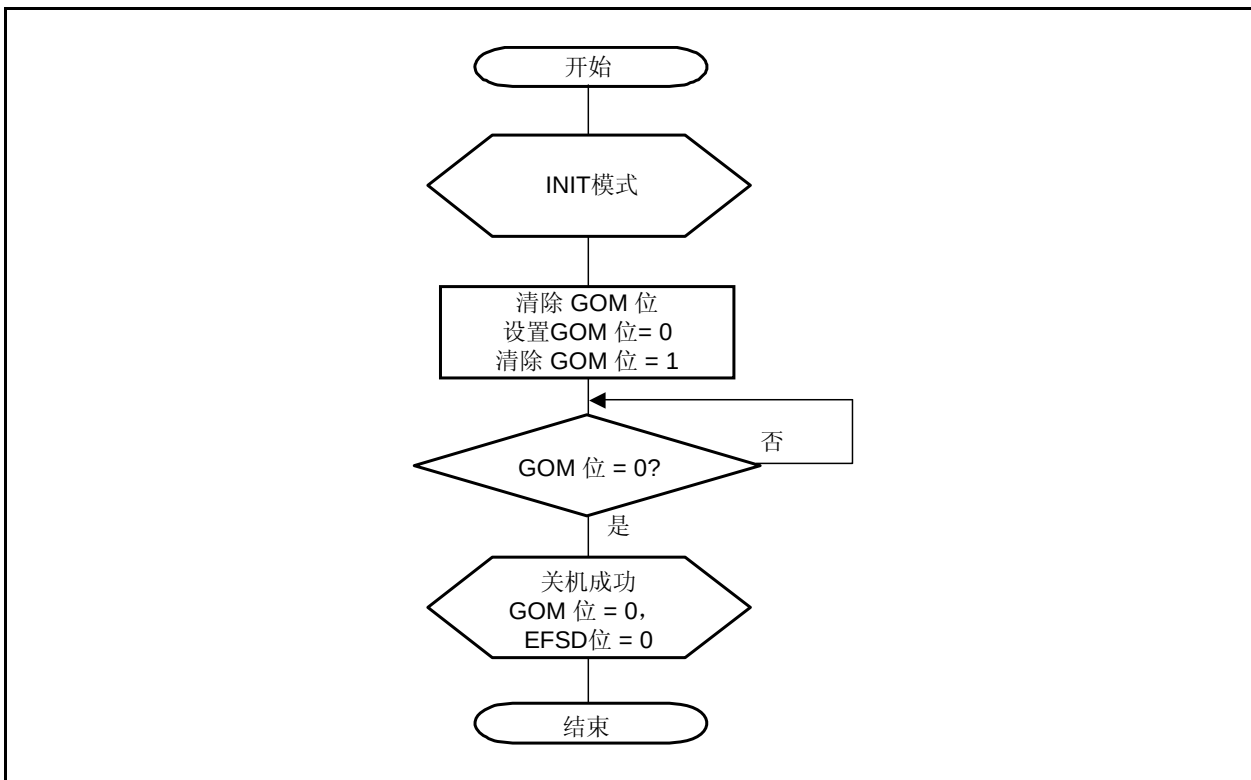


图 20-55. 强制关机处理

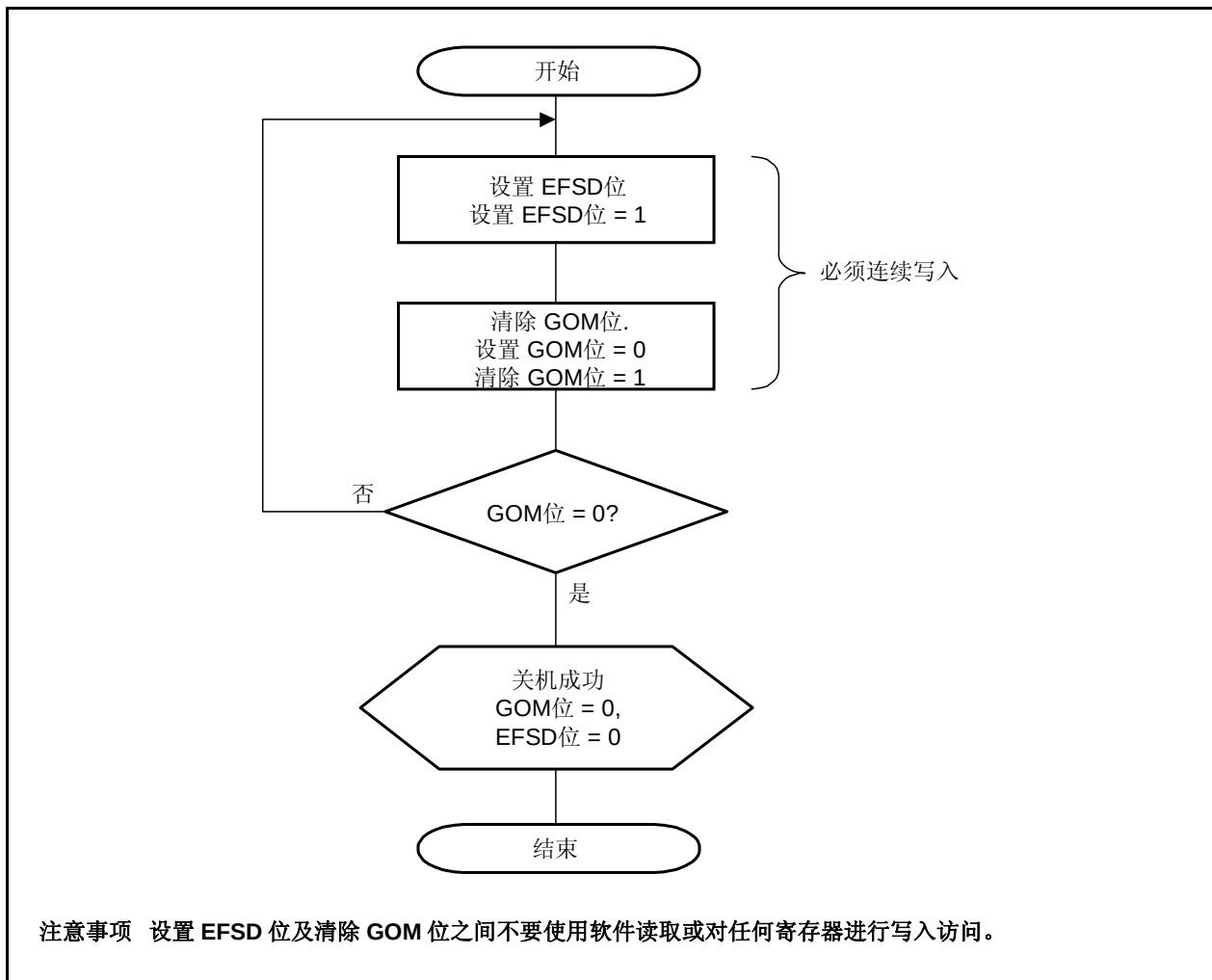


图 20-56. 错误处理

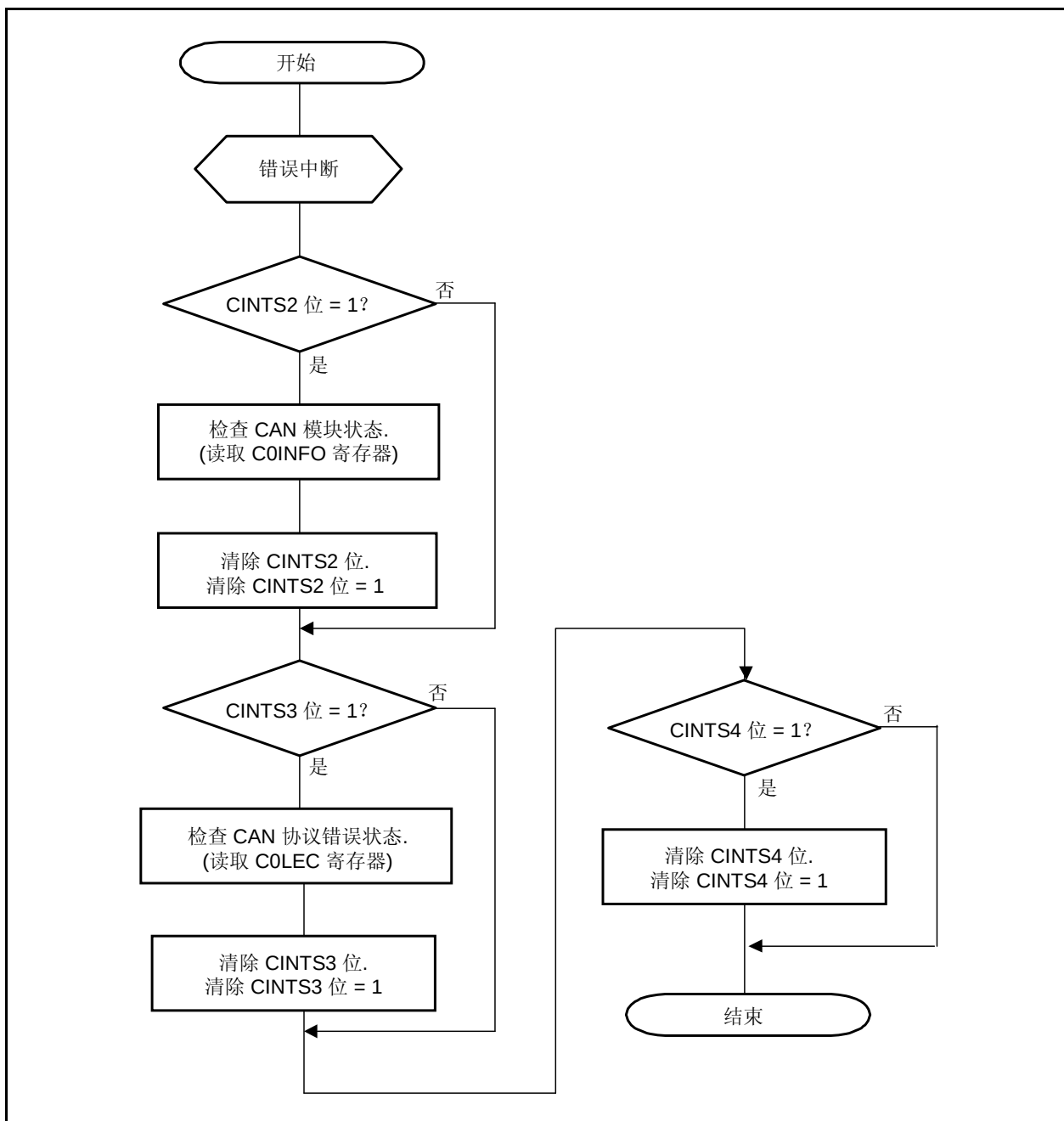


图 20-57. CPU 待机处理（从 CAN 睡眠模式转换）

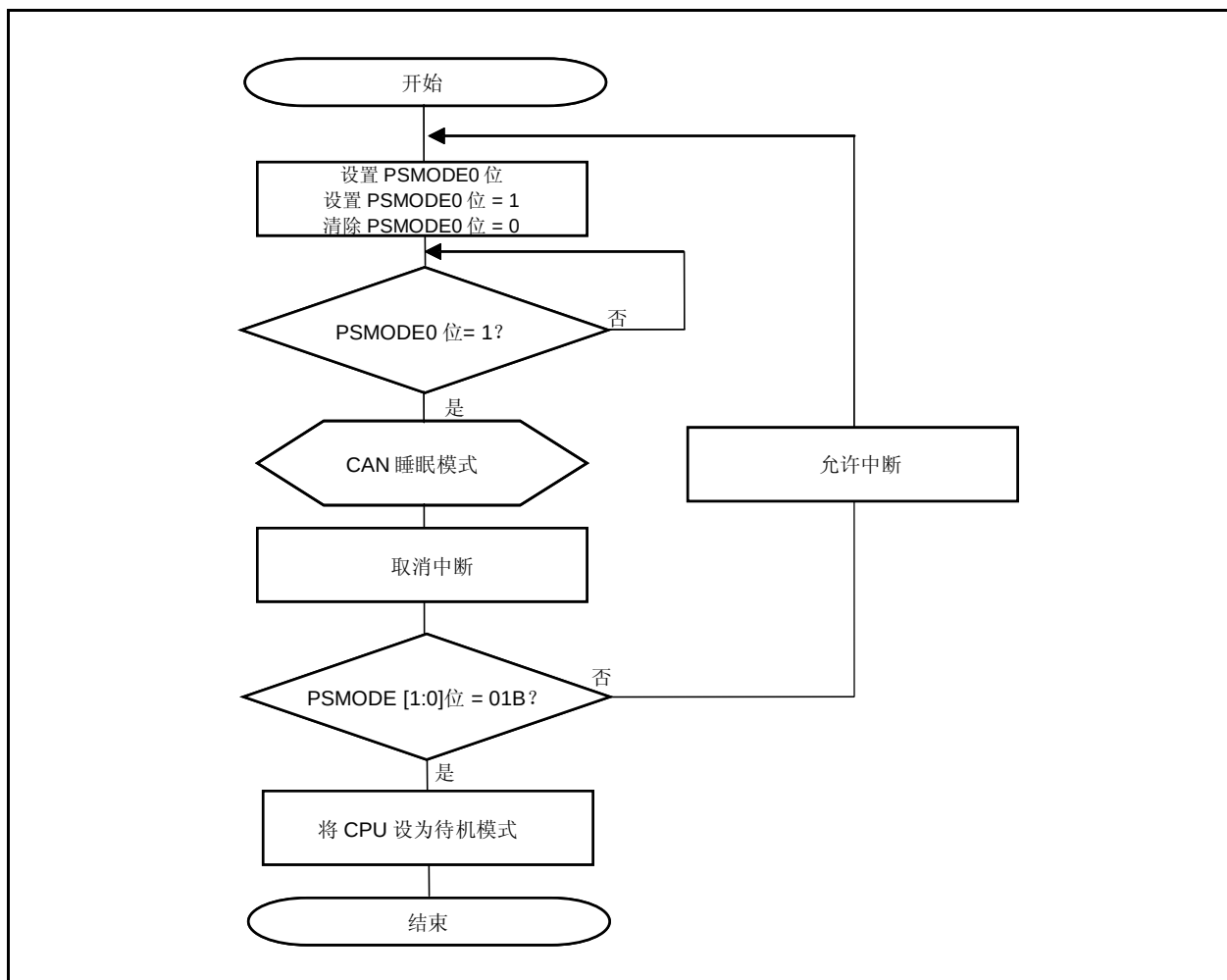
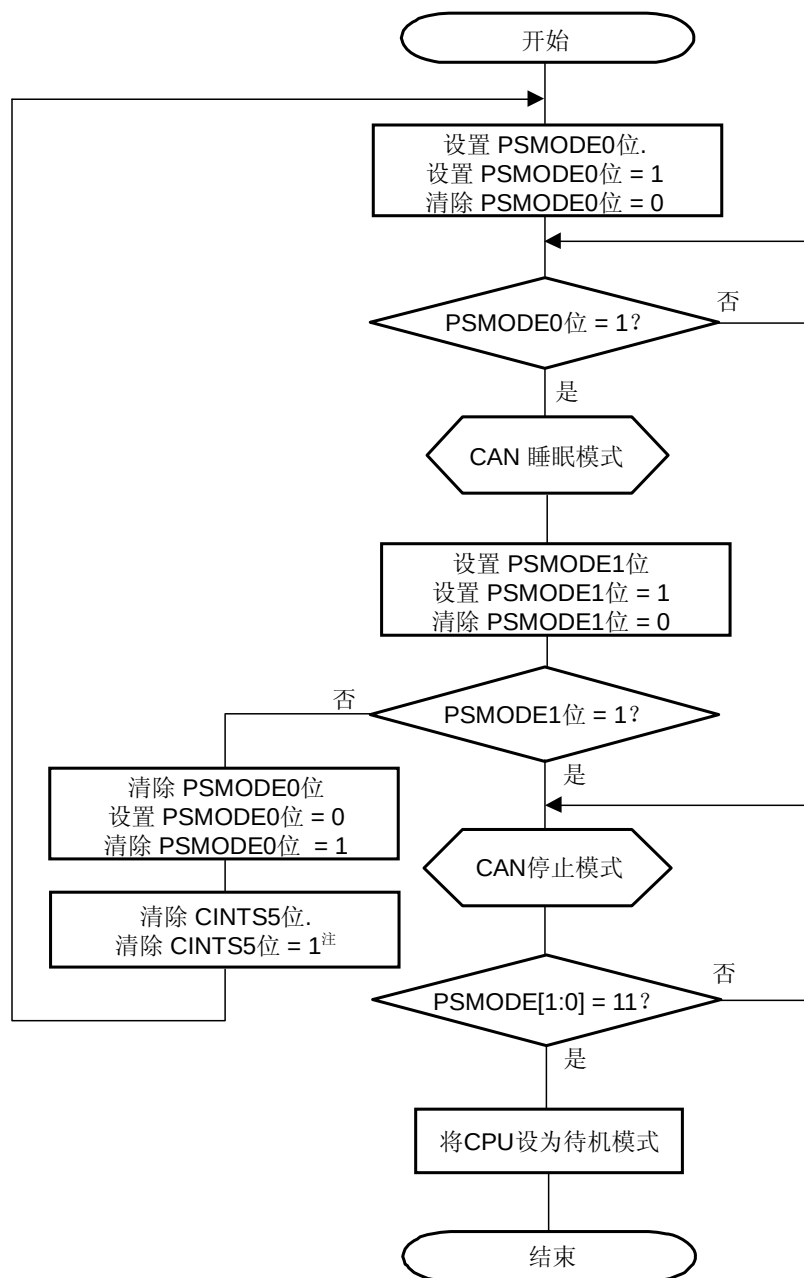


图 20-58. CPU 待机处理（从 CAN 停止模式转换）



注 唤醒中断期间

注意事项 仅能通过将 01 写入 C0CTRL.PSMODE1 和 C0CTRL.PSMODE0 位解除 CAN 停止模式。 不能通过改变 CAN 总线解除。

第二十一章 USB功能控制器（USBF）

V850ES/JG3-H 和 V850ES/JH3-H 单片机具有内部 USB 功能控制器（USBF），该功能寄存器遵循通用串行总线规范。USB 功能控制器和外部主机设备之间，使用基于令牌协议进行轮询数据通信。

21.1 概述

- 遵循通用串行总线规范
- 支持 12 Mbps（全速）传输
- 集成的传输端点

端点名称	FIFO 大小（字节）	传送类型：	备注
端点读取	64	控制传输	—
端点写入	64	控制传输	—
端点 1	64 × 2	Bulk-1 传输（IN）	双缓存配置
端点 2	64 × 2	Bulk-1 传输（OUT）	双缓存配置
端点 3	64 × 2	Bulk-2 传输（IN）	双缓存配置
端点 4	64 × 2	Bulk-2 传输（OUT）	双缓存配置
端点 7	8	中断传输	—

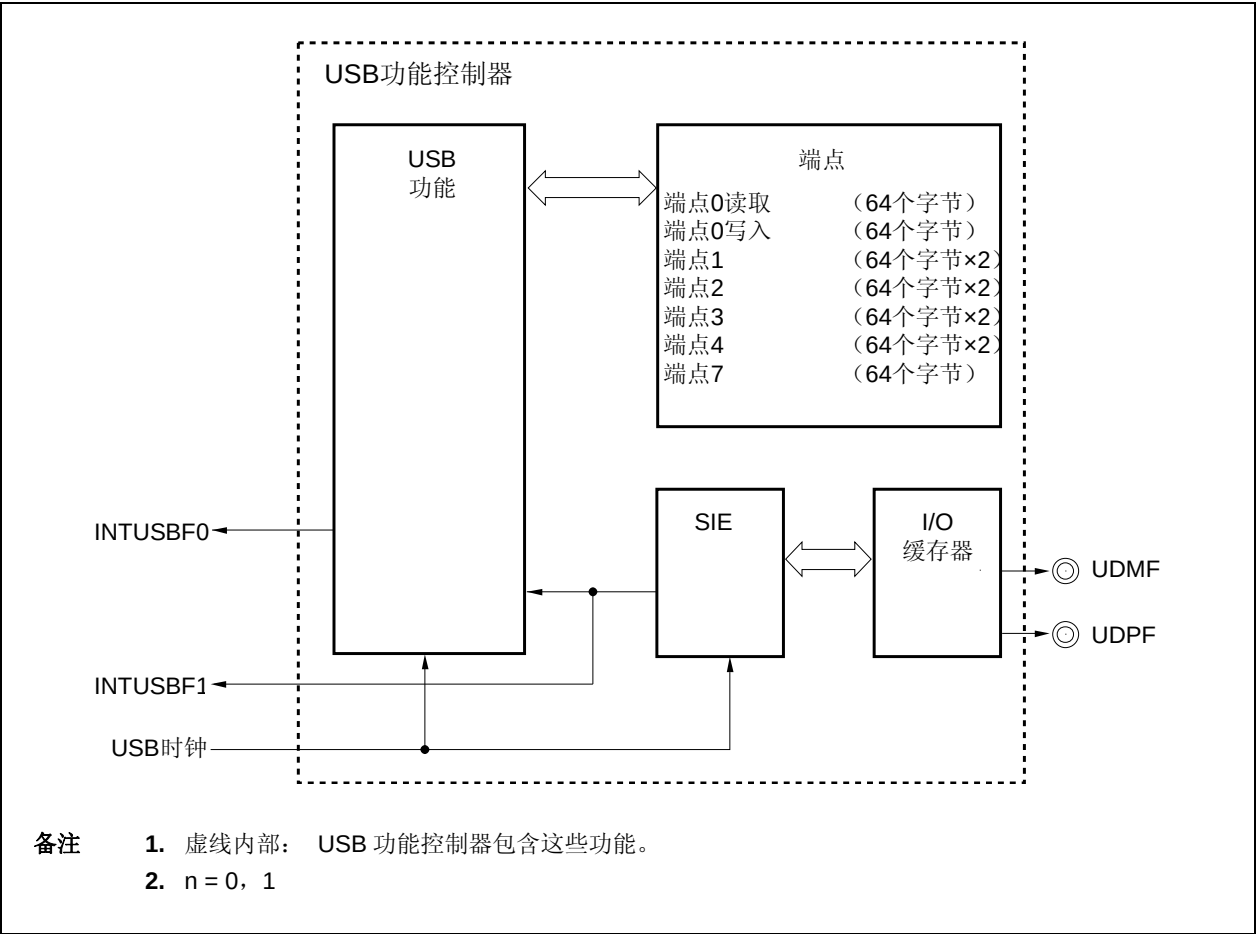
- Bulk 传输（IN/OUT）可以使用 DMA 进行传输（2-周期单次传输模式）
- 时钟： 可选择内部时钟（6 MHz 外部时钟 × 内部时钟 8 倍频 = 48 MHz 内部时钟）或外部时钟（输入至 UCLK 引脚的外部时钟（f_{USB} = 48 MHz））

21.2 配置

21.2.1 框图

<R>

图 21-1. USB 功能控制器框图



21.2.2 USB存储器映射

从 CPU 的角度来看，USB 功能控制器被分配到微控制器的 CS1 空间。 存储空间的划分如下所示。

表 21-1. CPU 存储空间的划分

地址	区域	
00200000H 至 00200092H	EPC 控制寄存器区域	
00200100H 至 00200114H	EPC 数据保持寄存器区域	
00200144H 至 002003C4H	EPC 请求数据寄存器区域	
00200400H 至 00200408H	桥寄存器区域	
00200500H 至 0020050EH	DMA 寄存器区域	
00201000H	Bulk-in 寄存器区域	EP1 (Bulk-IN1)
00202000H		EP3 (Bulk-IN2)
00210000H	Bulk-out 寄存器区域	EP2 (Bulk-Out1)
00220000H		EP4 (Bulk-Out2)
00240000H	外设控制寄存器区域	

21.3 外部电路结构

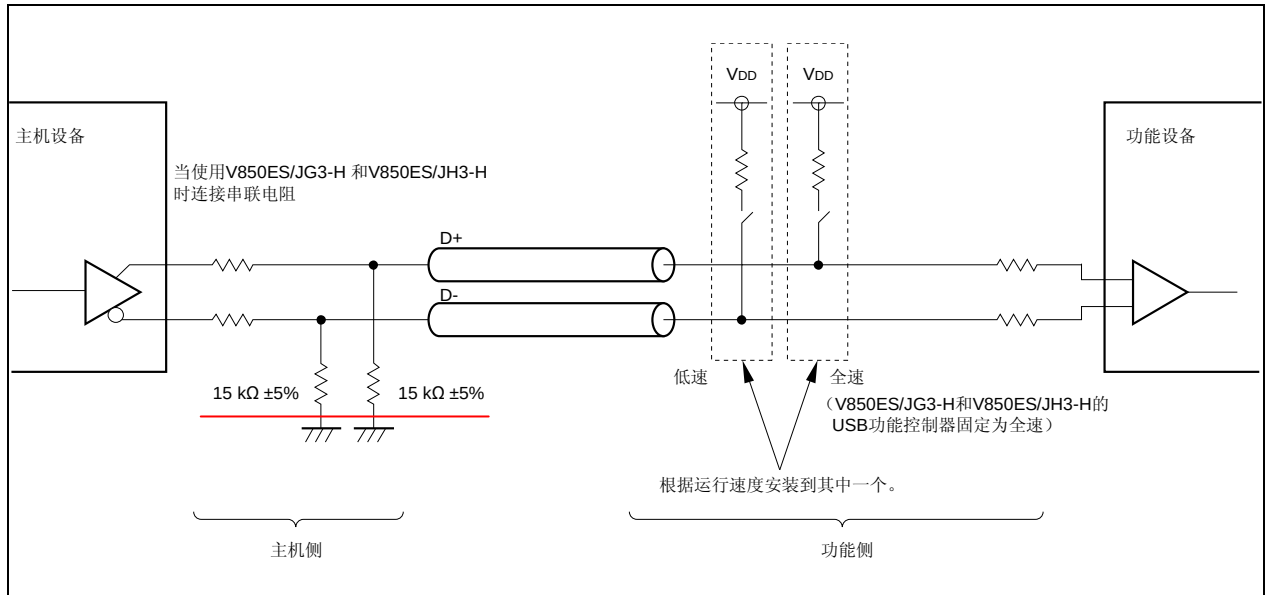
21.3.1 概要

USB 传送当中，当主机控制器和功能控制器之间互相进行通信时，双方必须都在 USB 信号 (D+/D-) 线上连接上拉/下拉电阻。此外，在 V850ES/JG3-H 和 V850ES/JH3-H 中还必须连接串联电阻。

因为 V850ES/JG3-H 和 V850ES/JH3-H 不包含这些上拉/下拉电阻和串联电阻，所以务必在外部连接这些电阻。

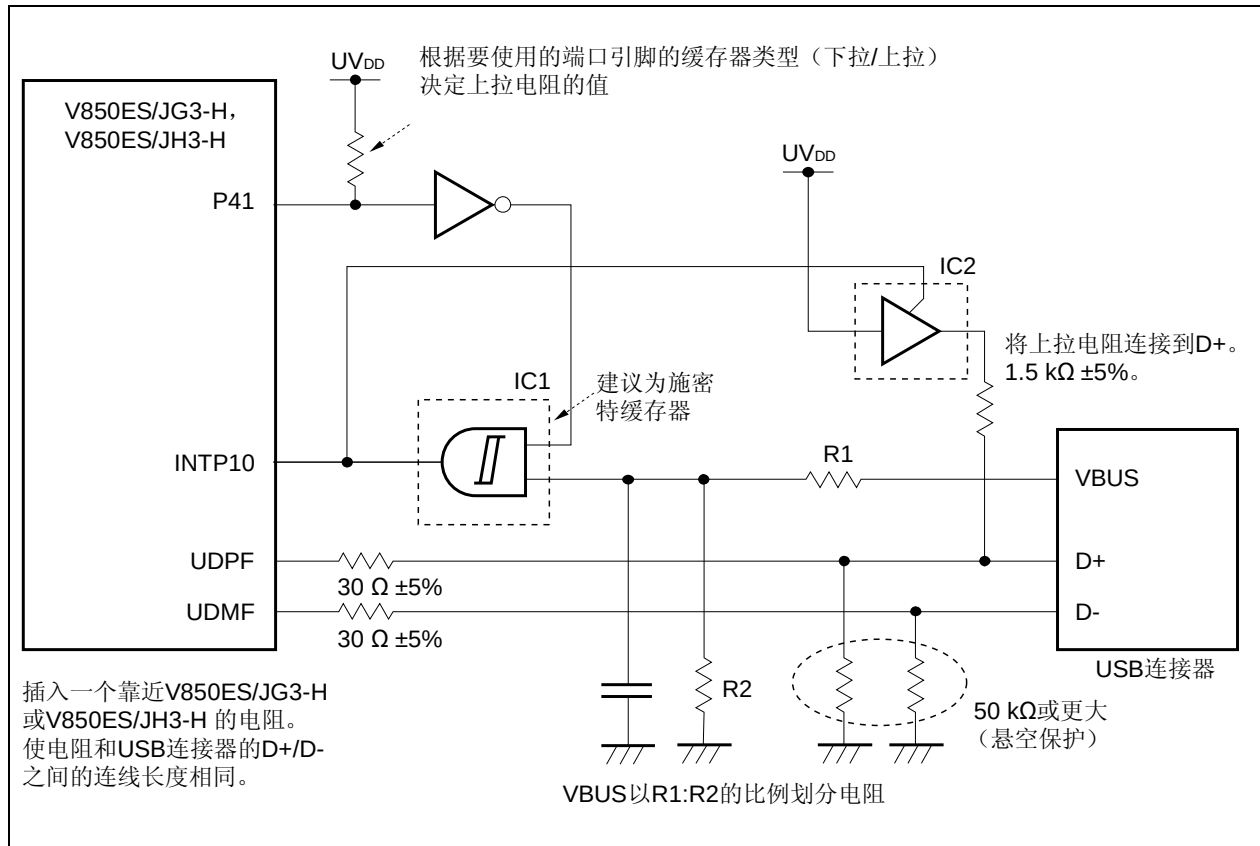
以下显示了 USB 传送信号线的概况。关于外部配置的详情，请参见各章节提供的说明。

图 21-2. USB 传送信号线中上拉、下拉和串联电阻的配置图



21.3.2 连接配置

图 21-3. USB 功能控制器的连接举例



(1) 连接到 D+/D- 的串联电阻

在 V850ES/JG3-H 和 V850ES/JH3-H 中，将 30 Ω ±5% 的串联电阻连接到 USB 功能控制器的 D+/D- 引脚 (UDPF, UDMF)。如果不连接这些电阻，则不能满足阻抗范围且输出波形可能会失常。

将串联电阻置于靠近 V850ES/JG3-H 或 V850ES/JH3-H 的位置，并使其距离和串联电阻到 USB 连接器之间的连线长度相同，以使得 D+ 和 D- 的阻抗相等（建议浮动范围 90 Ω ±5%）。

(2) D+ 的上拉控制

因为 V850ES/JG3-H 和 V850ES/JH3-H 的功能控制器被固定为全速 (FS)，所以一定要通过 1.5 kΩ ±5% 电阻连到 UV_{DD}，以确保上拉 D+ 引脚 (UDPF)。

要禁止向 USB 主机/HUB 连接报告 (D+ 上拉) (例如高优先级服务期间或初始化期间)，则需通过系统的通用端口控制 D+ 的上拉电阻。对于诸如图 21-3 所示的电路，应通过使用通用端口和 USB 线缆 VBUS (AND 电路) 来控制 D+ 引脚的上拉控制信号和 VBUS 输入信号。图 21-3 中，如果通用端口为高电平，则禁止 D+ 上拉 (这种情况下，因为通用端口引脚默认设置为输入端口，所以要通过上拉来保证高电平)。

对于图 21-3 中的 IC2，系统电源关闭时，所用 IC 仍可得到供电。

(3) USB 电缆连接/断开的检测

因为 USBF 的状态由硬件控制，所以 USB 功能控制器 (USB) 需要一个 VBUS 输入信号以辨别 USB 电缆是连接还是未连接。 USBF 电源关闭期间，当 USB 电缆 VBUS 连接到 USB 主机或 HUB 时，将 USB 或 HUB (5V) 的电压提供给 VBUS 作为输入信号。 因此，图 21-3 中的 IC1，当系统电源关闭时，所用 IC 仍可得到供电。 图 21-3 的电路不连接 USB 电缆，当 VBUS 电压下降时，INTP10 的输入信号可能会不稳定。 因此建议在图 21-3 中 IC1 使用施密特缓存器。

(4) 初始化期间或不使用 USBF 时的悬空保护

当初始化或不使用 USB 功能控制器时，为避免悬空状态，使用一个 50 k Ω 或更高的电阻将 D+/ D- 引脚下拉。

21.4 注意事项

(1) 时钟精度

为了使 USB 功能控制器工作，必须将内部时钟（6 MHz 外部时钟 × 内部时钟 8 倍频 = 48 MHz 内部时钟）或外部时钟（输入到 UCLK 引脚的外部时钟（ $f_{\text{USB}} = 48 \text{ MHz}$ ））用作 USB 时钟。当内部时钟用作 USB 时钟时，使用精度为 6 MHz $\pm 500 \text{ ppm}$ （最大）的晶振。使用外部时钟时，给 UCLK 引脚使用精度为 48 MHz $\pm 500 \text{ ppm}$ （最大）的时钟。如果 USB 时钟精度下降，则发送的数据不能满足 USB 标称值。

(2) HUB 连接

建议 HUB 连接时使用外部时钟（晶体振荡器）。如果使用内部时钟，则会因为微控制器的内部电路造成的时钟抖动影响而不能满足 USB 标称值。使用内部时钟时，建议将主机和设备进行 1 对 1 基本连接。

21.5 请求

USB 标准具有一个请求命令，该命令报告由主机设备到功能设备要求执行响应处理的请求。
请求在控制传输的 **SETUP** 阶段接收，且大部分请求可通过 **USB 功能控制器（USBF）** 的硬件自动进行处理。

21.5.1 自动请求

(1) 解码

下表显示了请求格式和请求与解码值之间的对应关系。

表 21-2. 请求格式

偏移量	字段名	
0	bmRequestType	
1	bRequest	
2	wValue	低端
3		高端
4	wIndex	低端
5		高端
6	wLength	低端
7		高端

表 21-3. 请求和解码值之间的对应

请求 \ 偏移量	解码值								响应			数据段
	bmRequestType	bRequest	wValue		wIndex		wLength		Df	Ad	Cf	
	0	1	3	2	5	4	7	6				
GET_INTERFACE	81H	0AH	00H	00H	00H	0nH	00H	01H	STALL	STALL	ACK NAK	√
GET_CONFIGURATION	80H	08H	00H	00H	00H	00H	00H	01H	ACK NAK	ACK NAK	ACK NAK	√
GET_DESCRIPTOR 设备	80H	06H	01H	00H	00H	00H	XXH	XXH ^{Note 1}	ACK NAK	ACK NAK	ACK NAK	√
GET_DESCRIPTOR 结构	80H	06H	02H	00H	00H	00H	XXH	XXH ^{Note 1}	ACK NAK	ACK NAK	ACK NAK	√
GET_STATUS 设备	80H	00H	00H	00H	00H	00H	00H	02H	ACK NAK	ACK NAK	ACK NAK	√
GET_STATUS 端点 0	82H	00H	00H	00H	00H	00H 80H	00H	02H	ACK NAK	ACK NAK	ACK NAK	√
GET_STATUS 端点 X	82H	00H	00H	00H	00H	\$\$H	00H	02H	STALL	STALL	ACK NAK	√
CLEAR_FEATURE 设备 ^{注 2}	00H	01H	00H	01H	00H	00H	00H	00H	ACK NAK	ACK NAK	ACK NAK	×
CLEAR_FEATURE 端点 0 ^{注 2}	02H	01H	00H	00H	00H	00H 80H	00H	00H	ACK NAK	ACK NAK	ACK NAK	×
CLEAR_FEATURE 端点 X ^{注 2}	02H	01H	00H	00H	00H	\$\$H	00H	00H	STALL	STALL	ACK NAK	×
SET_FEATURE 设备 ^{注 3}	00H	03H	00H	01H	00H	00H	00H	00H	ACK NAK	ACK NAK	ACK NAK	×
SET_FEATURE 端点 0 ^{注 3}	02H	03H	00H	00H	00H	00H 80H	00H	00H	ACK NAK	ACK NAK	ACK NAK	×
SET_FEATURE 端点 X ^{注 3}	02H	03H	00H	00H	00H	\$\$H	00H	00H	STALL	STALL	ACK NAK	×
SET_INTERFACE	01H	0BH	00H	0#H	00H	0?H	00H	00H	STALL	STALL	ACK NAK	×
SET_CONFIGURATION ^{注 4}	00H	09H	00H	00H 01H	00H	00H	00H	00H	ACK NAK	ACK NAK	ACK NAK	×
SET_ADDRESS	00H	05H	XXH	XXH	00H	00H	00H	00H	ACK NAK	ACK NAK	ACK NAK	×

备注 √: 数据段
 ×: 无数据段

- 注 1. 如果 wLength 的值小于已有值, 则返回 wLength 的值; 如果 wLength 值等于或大于已有值, 则返回已有值。
2. 当状态阶段接收到 ACK 时, CLEAR_FEATURE 请求将 UF0 设备状态寄存器 L (UF0DSTL) 和 UF0 EPn 状态寄存器 L (UF0EnSL) (n = 0 至 4, 7) 清除。

- 注 3. 当状态阶段接收到 ACK 时, SET_FEATURE 请求设置 UF0 设备状态寄存器 L (UF0DSTL) 和 UF0 EPn 状态寄存器 L (UF0EnSL) (n = 0 至 4, 7)。如果设置 UF0E0SL 寄存器的 E0HALT 位, 则一直到接收 CLEAR_FEATURE 端点 0 请求之前, 会在控制传输的状态段或数据段进行 STALL 响应, 除 GET_STATUS 端点 0 请求、SET_FEATURE 端点 0 请求和由 CPUDEC 中断请求产生的请求之外。对不支持的请求进行 STALL 响应不会将 UF0E0SL 寄存器的 E0HALT 位置 1, 且 STALL 响应会在接收到下一个 SETUP 令牌之后立即清除。
4. 如果 wValue 不是默认值, 则会进行自动 STALL 响应。

注意事项 1. 下列条件下不符合由通用串行总线规范所定义的控制传送序列。此时操作得不到保证。

- 如果不经 SETUP 阶段就突然接收到 IN/OUT 令牌
 - 如果在 SETUP 阶段的数据段发送 DATA PID1
 - 如果接收到 128 个地址或更多地址的令牌
 - 如果 SETUP 阶段发送的请求数据少于 8 个字节
2. 即使当主机在状态段发送空包之外的数据, 也会做出 ACK 响应。
3. FW 处理的控制传输 (读取) 期间, 如果 wLength 的值为 00H, 则控制传输会自动发送一个空包 (不含有数据)。FW 请求不自动发送空包。

- 备注 1. Df: 默认状态, Ad: 寻址状态, Cf: 配置状态
2. n = 0 至 4
接口编号 1 至 4 的请求能否正确得到响应, 由 UF0 动态接口编号寄存器 (UF0AIFN) 的设置决定, 这取决于目标的接口编号有效与否。
3. \$\$: 包含传输方向的有效端点编号
有效端点取决于当前设置的可选设置编号 (参见 21.6.3 (36) UF0 动态可选设置寄存器 (UF0AAS), (38) UF0 端点 1 接口映射寄存器 (UF0E1IM) 至 (42) UF0 端点 7 接口映射寄存器 (UF0E7IM))
4. ?和#: 由主机发送的值 (接口编号 0 至 4 的信息)
对应各接口编号的设置选择请求能否得到正确响应, 由 UF0 动态接口编号寄存器 (UF0AIFN) 和 UF0 动态可选设置寄存器 (UF0AAS) 所决定, 这取决于接口编号和目标的可选设置是否有效。

(2) 处理

默认状态、寻址状态和配置状态下自动请求的处理说明如下。

备注 默认状态： 该状态下，使用默认地址进行操作
 寻址状态： 已指定地址之后的状态
 配置状态： 正确接收到 SET_CONFIGURATION wValue = 1 之后的状态

(a) CLEAR_FEATURE()请求

如果不能清除 CLEAR_FEATURE()请求，或不存在 FEATURE 或目标为一个不存在的接口或端点时，会在状态阶段做出 STALL 响应。 如果 wLength 的值非 0 也会做出 STALL 响应。

- 默认状态： 仅当目标为一个设备或端点 0 的请求时，接收到 CLEAR_FEATURE()请求时会做出正确的响应；否则会在状态阶段做出 STALL 响应。
- 寻址状态： 仅当目标为一个设备或端点 0 的请求时，接收到 CLEAR_FEATURE()请求时会做出正确的响应；否则会在状态段做出 STALL 响应。
- 配置状态： 仅当目标为一个设备或为一个存在的端点请求时，接收到 CLEAR_FEATURE()请求时会做出正确的响应；否则会在状态段做出 STALL 响应。

CLEAR_FEATURE()请求正确处理之后，UF0 CLR 请求寄存器 (UF0CLR) 的相应位被设为 1，UF0 EPn 状态寄存器 L (UF0EnSL) 的 EnHALT 位被清为 0，并产生一个中断 (n = 0 至 4, 7)。 当主体是一个端点时如果接收到 CLEAR_FEATURE()请求，总是将相应端点的切换位 (用来控制 DATA0 和 DATA1 之间的转换) 复位为 DATA0。

(b) GET_CONFIGURATION()请求

如果 wValue, wIndex 或 wLength 不是表 21-3 所示的值，则会在数据段做出 STALL 响应。

- 默认状态： 当接收到 GET_CONFIGURATION() 请求时，返回存储在 UF0 配置寄存器 (UF0CNF) 中的值。
- 寻址状态： 当接收到 GET_CONFIGURATION()请求时，返回存储在 UF0CNF 寄存器中的值。
- 配置状态： 当接收到 GET_CONFIGURATION()请求时，返回存储在 UF0CNF 寄存器中的值。

(c) GET_DESCRIPTOR()请求

如果主体描述符的长度等于 wMaxPacketSize 倍数，则返回一个空包，表示数据段的结束。如果此时的描述符长度小于 wLength 值，则返回整个说明符；如果说明符的长度大于 wLength 的值，则返回多达 wLength 值长度的描述符。

- 默认状态：当接收到 GET_DESCRIPTOR()之后，返回存储在 UF0 设备寄存器 n (UF0DDn) 和 UF0 配置/接口/端点说明符寄存器 m (UF0CIEm) 中的值 (n = 0 至 17, m = 0 至 255)。
- 寻址状态：接收到 GET_DESCRIPTOR()之后，返回存储在 UF0DDn 寄存器和 UF0CIEm 寄存器中的值。
- 配置状态：接收到 GET_DESCRIPTOR()请求之后，返回存储在 UF0DDn 寄存器和 UF0CIEm 寄存器中的值。

在 UF0CIEm 寄存器中可存储多达 256 个字节的描述符。要返回一个大于 256 个字节的说明符，需将 UF0MODC 寄存器的 CDCGDST 位设为 1 并使用 FW 处理 GET_DESCRIPTOR()请求。

将 UF0CIEm 寄存器-1 设置的说明符的总字节数目的值存储在 UF0 说明符长度寄存器 (UF0DSCL) 中。传送数据由该数据+1 的值和 wLength 的值控制。

(d) GET_INTERFACE()请求

如果 wValue 和 wLength 的值都不是表 21-3 所示的值，或 wIndex 不是 UF0 动态接口编号寄存器 (UF0AIFN) 设置的值，则会在数据段做出 STALL 响应。

- 默认状态：接收到 GET_INTERFACE()请求时会在数据段做出 STALL 响应。
- 寻址状态：接收到 GET_INTERFACE()请求时会在数据段做出 STALL 响应。
- 配置状态：接收到 GET_INTERFACE()时，返回存储在 UF0 接口 n 寄存器 (UF0IFn) 中的值，该值对应于 wIndex 的值 (n = 0 至 4)。

(e) GET_STATUS()请求

如果 wValue, wIndex 或 wLength 都不是表 21-3 所示的值, 则会在数据段做出 STALL 响应。如果目标是一个不存在的接口或端点时也会在数据段做出 STALL 响应。

- 默认状态: 仅当接收到 GET_STATUS()请求且该请求是对设备或端点 0 的请求时, 才会返回目标状态寄存器[#]中存储的值; 否则, 在数据段进行 STALL 应答。
- 寻址状态: 仅当接收到 GET_STATUS()请求且该请求用于设备或端点 0 时, 才会返回目标状态寄存器[#]中存储的值; 否则, 在数据段进行 STALL 应答。
- 配置状态: 仅当接收到 GET_STATUS()请求且该请求用于存在的设备或端点 0 时, 才会返回目标状态寄存器[#]中存储的值; 否则, 在数据段进行 STALL 应答。

注 目标状态寄存器如下。

- 如果目标是一个设备: UF0 设备状态寄存器 L (UF0DSTL)
- 如果目标是端点 0: UF0 EP0 状态寄存器 L (UF0E0SL)
- 如果目标是端点 n: UF0 Epn 状态寄存器 L (UF0EnSL) (n = 1 至 4, 7)

(f) SET_ADDRESS()请求

如果 wIndex 或 wLength 不是表 21-3 中所示的值, 则在状态段做出 STALL 响应。如果指定设备的地址大于 127 也会做出 STALL 响应。

- 默认状态: 当接收到 SET_ADDRESS()时, 如果指定的地址不是 0, 则设备进入寻址状态并将待输入到 SIE 的 USB 地址值变为指定地址值。如果指定地址是 0, 则设备保持默认状态。
- 寻址状态: 当接收到 SET_ADDRESS()时, 如果指定的地址是 0, 则设备进入默认状态并将 SIE 的 USB 地址值返回给默认地址。如果指定地址不是 0, 设备保持寻址状态, 并将要输入到 SIE 的 USB 地址值变为一个新的指定地址值。
- 配置状态: 当接收到 SET_ADDRESS()时, 如果指定的地址是 0, 则设备保持配置状态并将 SIE 的 USB 地址值返回给默认地址。这种情况下, 除了端点 0 之外的端点仍然有效, 且除了端点 0 之外的控制传输 (IN)、控制传输 (OUT)、Bulk 传输和中断传输也都会得到响应。如果指定地址不是 0, 设备仍处于配置状态, 并将要输入到 SIE 的 USB 地址值变为一个指定的新地址值。

(g) SET_CONFIGURATION()请求

如果 wValue, wIndex 或 wLength 不是表 21-3 所示的值, 则会在状态阶段做出 STALL 响应。

- 默认状态: 接收到 SET_CONFIGURATION()请求时, 如果指定配置值是 1, 则将 UF0 模式状态寄存器 (UF0MODS) 和 UF0 配置寄存器 (UF0CNF) 的 CONF 位设为 1。如果指定配置值是 0, 则将 UF0MODS 寄存器和 UF0CNF 寄存器的 CONF 位清为 0。换言之, 设备会跳过寻址状态而移到配置状态, 其在该状态下对应默认地址。
- 寻址状态: 接收到 SET_CONFIGURATION()请求时, 如果指定配置值是 1, 则将 UF0MODS 寄存器和 UF0CNF 寄存器的 CONF 位设为 1, 且设备进入配置状态。如果指定配置值是 0, 则设备保持寻址状态。
- 配置状态: 接收到 SET_CONFIGURATION()请求时, 如果指定配置值是 0, 则将 UF0MODS 寄存器和 UF0CNF 寄存器的 CONF 位设为 1, 且设备返回寻址状态。如果指定配置值是 1, 则设备保持配置状态。

如果正确处理了 SET_CONFIGURATION()请求, 则将 UF0 SET 请求寄存器 (UF0SE) 的目标位设为 1 并产生一个中断。SET_CONFIGURATION()完成之后, 即使指定的配置值与当前的配置值相同也会清除 Halt 特性。如果正确处理了 SET_CONFIGURATION()请求, 则总是将所有端点的数据转换再次初始化到 DATA0 (它被定义为默认状态, 从接收到 SET_CONFIGURATION 请求时一直到接收到 SET_INTERFACE 请求时设定为可选设置 0)。

(h) SET_FEATURE()请求

如果 SET_FEATURE()请求用于不可设置或不存在的功能部件, 或者如果目标为一个接口或不存在的端点时, 会在状态阶段做出 STALL 响应。如果 wLength 的值非 0 也会做出 STALL 响应。

- 默认状态: 如果请求用于设备或端点 0, 当接收到 SET_FEATURE()请求时会做出正确的响应; 否则会在状态阶段做出 STALL 响应。
- 寻址状态: 如果请求用于设备或端点 0, 当接收到 SET_FEATURE()请求时会做出正确的响应; 否则会在状态阶段做出 STALL 响应。
- 配置状态: 如果请求用于一个设备或为一个存在的端点, 当接收到 SET_FEATURE()请求时会做出正确的响应; 否则会在状态阶段做出 STALL 响应。

SET_FEATURE()请求正确处理之后, 将 UF0 SET 请求寄存器 (UF0SET) 的目标位和 UF0 EPn 状态寄存器 L (UF0EnSL) 的 EnHALT 位设为 1, 并产生一个中断 (n = 0 至 4, 7)。

(i) SET_INTERFACE()请求

如果 wLength 不是表 21-3 中所示的值，wIndex 不是设置到 UF0 动态接口编号寄存器 (UF0AIFN) 的值，且 wValue 也不是设置到 UF0 动态可选设置寄存器 (UF0AAS) 的值，则会在状态阶段做出 STALL 响应。

- 默认状态: 接收到 SET_INTERFACE()请求时会在状态阶段做出 STALL 响应。
- 寻址状态: 接收到 SET_INTERFACE()请求时会在状态阶段做出 STALL 响应。
- 配置状态: 接收到 SET_INTERFACE()请求时会在状态阶段发送空包。

当正确处理 SET_INTERFACE()时，会产生一个中断。清除 SET_INTERFACE()请求之后，会清除所有连接到目标接口的端点的 Halt 特性。总是将与目标接口编号相关的所有端点的数据转换重新初始化为 DATA0。当通过正确处理 SET_INTERFACE()请求改变当前所选的可选设置时，受影响端点的 FIFO 会被完全清除，且所有相关的中断源都会被初始化。

当完成 SET_INTERFACE()时，连接到目标接口的所有端点的 FIFO 都被清除。与此同时，Halt 特性和 DATA PID 被初始化，且相关的 UF0 INT 状态寄存器 (UF0ISn) 被清除为 0 (n = 0 至 4)。（当完成 SET_CONFIGURATION 请求时，仅清除 Halt 特性和 DATA PID。）

DMA 传输期间，如果 SET_INTERFACE()请求不支持目标端点，则立刻声明 DMA 请求信号，且完全清除在 SET_INTERFACE()请求完成时连接的端点的 FIFO。清除 FIFO 导致 DMA 的数据传输不能正确处理。

21.5.2 其它请求

(1) 响应和处理

下表显示了如何对其它请求做出响应和处理。

表 21-4. 其它请求的响应和处理

请求	响应和处理
GET_DESCRIPTOR 信息串	产生 CPUDEC 中断请求
GET_STATUS 接口	自动 STALL 响应
CLEAR_FEATURE 接口	自动 STALL 响应
SET_FEATURE 接口	自动 STALL 响应
所有的 SET_DESCRIPTOR	产生 CPUDEC 中断请求
所有其它请求	产生 CPUDEC 中断请求

21.6 寄存器配置

21.6.1 USB控制寄存器

(1) USB 时钟选择寄存器（UCKSEL）

UCKSEL 寄存器选择 USB 控制器的工作时钟。

UCKSEL 寄存器可以按字节或按位读写。

复位后本寄存器被设置为 00H。

复位后：00H R/W 地址：FFFFFF40H

	7	6	5	4	3	2	1	0
UCKSEL	0	0	0	0	0	0	UUSEL1	0

UUSEL1	USB 控制器工作时钟的选择
0	由UCLK 引脚输入的外部时钟（f _{USB} = 48 MHz）
1	主时钟（f _{xx} = 48 MHz）

注意事项 一定要将位 7 至 2，和位 0 设为“0”。

(2) USB 功能控制寄存器（UFCKMSK）

UFCKMSK 寄存器控制使能/禁止 USB 功能控制器的操作。

UFCKMSK 寄存器可以按字节或按位读写。

系统复位后该寄存器被置为 03H。

复位后：03H R/W 地址：FFFFFF41H

	7	6	5	4	3	2	1	0
UFCKMSK	0	0	0	0	0	0	UFBUFMSK	UFMSK

UFBUFMSK	UFMSK	USB 功能控制器操作使能/禁止
0	0	使能操作
1	1	禁止操作
其它		禁止设置

(3) USB 功能选择寄存器 (UHCKMSK)

在使用 USB 控制器功能时，UHCKMSK 寄存器用于控制数据 RAM 的操作。
UHCKMSK 寄存器可以按字节或按位读写。
系统复位后该寄存器被置为 03H。

复位后: 03H R/W 地址: FFFFFFF42H

	7	6	5	4	3	2	1	0
UHCKMSK	0	0	0	0	0	0	1	UHMSK

UHMSK	USB 控制器功能选择
0	使能数据 RAM (8 KB) 操作
1	禁止使用 USB 主机控制器/数据 RAM (8 KB)

注意事项 确保将位 7 至 2 设为“0”，并将位 1 设为“1”。

21.6.2 USB功能控制器寄存器列表

(1) EPC 控制寄存器

(1/2)

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
00200000H	UF0 EP0NAK 寄存器	UF0E0N	R/W		√		00H
00200002H	UF0 EP0NAKALL 寄存器	UF0E0NA	R/W		√		00H
00200004H	UF0 EPNAK 寄存器	UF0EN	R/W		√		00H
00200006H	UF0 EPNAK 屏蔽寄存器	UF0ENM	R/W		√		00H
00200008H	UF0 SNDSIE 寄存器	UF0SDS	R/W		√		00H
0020000AH	UF0 CLR 请求寄存器	UF0CLR	R		√		00H
0020000CH	UF0 SET 请求寄存器	UF0SET	R		√		00H
0020000EH	UF0 EP 状态 0 寄存器	UF0EPS0	R		√		00H
00200010H	UF0 EP 状态 1 寄存器	UF0EPS1	R		√		00H
00200012H	UF0 EP 状态 2 寄存器	UF0EPS2	R		√		00H
00200020H	UF0 INT 状态 0 寄存器	UF0IS0	R		√		00H
00200022H	UF0 INT 状态 1 寄存器	UF0IS1	R		√		00H
00200024H	UF0 INT 状态 2 寄存器	UF0IS2	R		√		00H
00200026H	UF0 INT 状态 3 寄存器	UF0IS3	R		√		00H
00200028H	UF0 INT 状态 4 寄存器	UF0IS4	R		√		00H
0020002EH	UF0 INT 屏蔽 0 寄存器	UF0IM0	R/W		√		00H
00200030H	UF0 INT 屏蔽 1 寄存器	UF0IM1	R/W		√		00H
00200032H	UF0 INT 屏蔽 2 寄存器	UF0IM2	R/W		√		00H
00200034H	UF0 INT 屏蔽 3 寄存器	UF0IM3	R/W		√		00H
00200036H	UF0 INT 屏蔽 4 寄存器	UF0IM4	R/W		√		00H
0020003CH	UF0 INT 清除 0 寄存器	UF0IC0	W		√		FFH
0020003EH	UF0 INT 清除 1 寄存器	UF0IC1	W		√		FFH
00200040H	UF0 INT 清除 2 寄存器	UF0IC2	W		√		FFH
00200042H	UF0 INT 清除 3 寄存器	UF0IC3	W		√		FFH
00200044H	UF0 INT 清除 4 寄存器	UF0IC4	W		√		FFH
0020004CH	UF0 INT & DMARQ 寄存器	UF0IDR	R/W		√		00H
0020004EH	UF0 DMA 状态 0 寄存器	UF0DMS0	R		√		00H
00200050H	UF0 DMA 状态 1 寄存器	UF0DMS1	R		√		00H
00200060H	UF0 FIFO 清除 0 寄存器	UF0FIC0	W		√		00H
00200062H	UF0 FIFO 清除 1 寄存器	UF0FIC1	W		√		00H
0020006AH	UF0 数据结束寄存器	UF0DEND	R/W		√		00H
0020006EH	UF0 GPR 寄存器	UF0GPR	W		√		00H
00200074H	UF0 模式控制寄存器	UF0MODC	R/W		√		00H
00200078H	UF0 模式状态寄存器	UF0MODS	R		√		00H

(2/2)

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
00200080H	UF0 动态接口编号寄存器	UF0AIFN	R/W		√		00H
00200082H	UF0 动态可选设置寄存器	UF0AAS	R/W		√		00H
00200084H	UF0 可选设置状态寄存器	UF0ASS	R		√		00H
00200086H	UF0 端点 1 接口映射寄存器	UF0E1IM	R/W		√		00H
00200088H	UF0 端点 2 接口映射寄存器	UF0E2IM	R/W		√		00H
0020008AH	UF0 端点 3 接口映射寄存器	UF0E3IM	R/W		√		00H
0020008CH	UF0 端点 4 接口映射寄存器	UF0E4IM	R/W		√		00H
00200092H	UF0 端点 7 接口映射寄存器	UF0E7IM	R/W		√		00H

(2) EPC 数据保持寄存器

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
00200100 H	UF0 EP0 读寄存器	UF0E0R	R		√		不确定
00200102H	UF0 EP0 长度寄存器	UF0E0L	R		√		00H
00200104H	UF0 EP0 设置寄存器	UF0E0ST	R		√		00H
00200106H	UF0 EP0 写寄存器	UF0E0W	W		√		不确定
00200108H	UF0 bulk-out 1 寄存器	UF0BO1	R		√		不确定
0020010AH	UF0 bulk-out 1 长度寄存器	UF0BO1L	R		√		00H
0020010CH	UF0 bulk-out 2 寄存器	UF0BO2	R		√		不确定
0020010EH	UF0 bulk-out 2 长度寄存器	UF0BO2L	R		√		00H
00200110H	UF0 bulk-in 1 寄存器	UF0BI1	W		√		不确定
00200112H	UF0 bulk-in 2 寄存器	UF0BI2	W		√		不确定
00200114H	UF0 中断 1 寄存器	UF0INT1	W		√		不确定

(3) EPC 请求数据寄存器

(1/13)

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
00200144H	UF0 设备状态寄存器 L	UF0DSTL	R/W		√		00H
00200146H	UF0 设备状态寄存器 H	UF0DSTH	R/W		√		00H
00200148H	UF0 接口状态寄存器 L	UF0ISTL	R/W		√		00H
0020014AH	UF0 接口状态寄存器 H	UF0ISTH	R/W		√		00H
0020014CH	UF0 EP0 状态寄存器 L	UF0E0SL	R/W		√		00H
0020014EH	UF0 EP0 状态寄存器 H	UF0E0SH	R/W		√		00H
00200150H	UF0 EP1 状态寄存器 L	UF0E1SL	R/W		√		00H
00200152H	UF0 EP1 状态寄存器 H	UF0E1SH	R/W		√		00H
00200154H	UF0 EP2 状态寄存器 L	UF0E2SL	R/W		√		00H
00200156H	UF0 EP2 状态寄存器 H	UF0E2SH	R/W		√		00H
00200158H	UF0 EP3 状态寄存器 L	UF0E3SL	R/W		√		00H
0020015AH	UF0 EP3 状态寄存器 H	UF0E3SH	R/W		√		00H
0020015CH	UF0 EP4 状态寄存器 L	UF0E4SL	R/W		√		00H
0020015EH	UF0 EP4 状态寄存器 H	UF0E4SH	R/W		√		00H
00200168H	UF0 EP7 状态寄存器 L	UF0E7SL	R/W		√		00H
0020016AH	UF0 EP7 状态寄存器 H	UF0E7SH	R/W		√		00H
00200180H	UF0 地址寄存器	UF0ADRS	R		√		00H
00200182H	UF0 配置寄存器	UF0CNF	R		√		00H
00200184H	UF0 接口 0 寄存器	UF0IF0	R		√		00H
00200186H	UF0 接口 1 寄存器	UF0IF1	R		√		00H
00200188H	UF0 接口 2 寄存器	UF0IF2	R		√		00H
0020018AH	UF0 接口 3 寄存器	UF0IF3	R		√		00H
0020018CH	UF0 接口 4 寄存器	UF0IF4	R		√		00H
002001A0H	UF0 描述符长度寄存器	UF0DSCL	R/W		√		00H
002001A2H	UF0 设备描述符寄存器 0	UF0DD0	R/W		√		不确定
002001A4H	UF0 设备描述符寄存器 1	UF0DD1	R/W		√		不确定
002001A6H	UF0 设备描述符寄存器 2	UF0DD2	R/W		√		不确定
002001A8H	UF0 设备描述符寄存器 3	UF0DD3	R/W		√		不确定
002001AAH	UF0 设备描述符寄存器 4	UF0DD4	R/W		√		不确定
002001ACH	UF0 设备描述符寄存器 5	UF0DD5	R/W		√		不确定
002001AEH	UF0 设备描述符寄存器 6	UF0DD6	R/W		√		不确定
002001B0H	UF0 设备描述符寄存器 7	UF0DD7	R/W		√		不确定
002001B2H	UF0 设备描述符寄存器 8	UF0DD8	R/W		√		不确定
002001B4H	UF0 设备描述符寄存器 9	UF0DD9	R/W		√		不确定
002001B6H	UF0 设备描述符寄存器 10	UF0DD10	R/W		√		不确定
002001B8H	UF0 设备描述符寄存器 11	UF0DD11	R/W		√		不确定
002001BAH	UF0 设备描述符寄存器 12	UF0DD12	R/W		√		不确定
002001BCH	UF0 设备描述符寄存器 13	UF0DD13	R/W		√		不确定
002001BEH	UF0 设备描述符寄存器 14	UF0DD14	R/W		√		不确定
002001C0H	UF0 设备描述符寄存器 15	UF0DD15	R/W		√		不确定
002001C2H	UF0 设备描述符寄存器 16	UF0DD16	R/W		√		不确定
002001C4H	UF0 设备描述符寄存器 17	UF0DD17	R/W		√		不确定

(2/13)

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
002001C6H	UF0 配置/接口/端点描述符寄存器 0	UF0CIE0	R/W		√		不确定
002001C8H	UF0 配置/接口/端点描述符寄存器 1	UF0CIE1	R/W		√		不确定
002001CAH	UF0 配置/接口/端点描述符寄存器 2	UF0CIE2	R/W		√		不确定
002001CCH	UF0 配置/接口/端点描述符寄存器 3	UF0CIE3	R/W		√		不确定
002001CEH	UF0 配置/接口/端点描述符寄存器 4	UF0CIE4	R/W		√		不确定
002001D0H	UF0 配置/接口/端点描述符寄存器 5	UF0CIE5	R/W		√		不确定
002001D2H	UF0 配置/接口/端点描述符寄存器 6	UF0CIE6	R/W		√		不确定
002001D4H	UF0 配置/接口/端点描述符寄存器 7	UF0CIE7	R/W		√		不确定
002001D6H	UF0 配置/接口/端点描述符寄存器 8	UF0CIE8	R/W		√		不确定
002001D8H	UF0 配置/接口/端点描述符寄存器 9	UF0CIE9	R/W		√		不确定
002001DAH	UF0 配置/接口/端点描述符寄存器 10	UF0CIE10	R/W		√		不确定
002001DCH	UF0 配置/接口/端点描述符寄存器 11	UF0CIE11	R/W		√		不确定
002001DEH	UF0 配置/接口/端点描述符寄存器 12	UF0CIE12	R/W		√		不确定
002001E0H	UF0 配置/接口/端点描述符寄存器 13	UF0CIE13	R/W		√		不确定
002001E2H	UF0 配置/接口/端点描述符寄存器 14	UF0CIE14	R/W		√		不确定
002001E4H	UF0 配置/接口/端点描述符寄存器 15	UF0CIE15	R/W		√		不确定
002001E6H	UF0 配置/接口/端点描述符寄存器 16	UF0CIE16	R/W		√		不确定
002001E8H	UF0 配置/接口/端点描述符寄存器 17	UF0CIE17	R/W		√		不确定
002001EAH	UF0 配置/接口/端点描述符寄存器 18	UF0CIE18	R/W		√		不确定
002001ECH	UF0 配置/接口/端点描述符寄存器 19	UF0CIE19	R/W		√		不确定
002001EEH	UF0 配置/接口/端点描述符寄存器 20	UF0CIE20	R/W		√		不确定
002001F0H	UF0 配置/接口/端点描述符寄存器 21	UF0CIE21	R/W		√		不确定

(3/13)

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
002001F2H	UF0 配置/接口/端点描述符寄存器 22	UF0CIE22	R/W		√		不确定
002001F4H	UF0 配置/接口/端点描述符寄存器 23	UF0CIE23	R/W		√		不确定
002001F6H	UF0 配置/接口/端点描述符寄存器 24	UF0CIE24	R/W		√		不确定
002001F8H	UF0 配置/接口/端点描述符寄存器 25	UF0CIE25	R/W		√		不确定
002001FAH	UF0 配置/接口/端点描述符寄存器 26	UF0CIE26	R/W		√		不确定
002001FCH	UF0 配置/接口/端点描述符寄存器 27	UF0CIE27	R/W		√		不确定
002001FEH	UF0 配置/接口/端点描述符寄存器 28	UF0CIE28	R/W		√		不确定
00200200H	UF0 配置/接口/端点描述符寄存器 29	UF0CIE29	R/W		√		不确定
00200202H	UF0 配置/接口/端点描述符寄存器 30	UF0CIE30	R/W		√		不确定
00200204H	UF0 配置/接口/端点描述符寄存器 31	UF0CIE31	R/W		√		不确定
00200206H	UF0 配置/接口/端点描述符寄存器 32	UF0CIE32	R/W		√		不确定
00200208H	UF0 配置/接口/端点描述符寄存器 33	UF0CIE33	R/W		√		不确定
0020020AH	UF0 配置/接口/端点描述符寄存器 34	UF0CIE34	R/W		√		不确定
0020020CH	UF0 配置/接口/端点描述符寄存器 35	UF0CIE35	R/W		√		不确定
0020020EH	UF0 配置/接口/端点描述符寄存器 36	UF0CIE36	R/W		√		不确定
00200210H	UF0 配置/接口/端点描述符寄存器 37	UF0CIE37	R/W		√		不确定
00200212H	UF0 配置/接口/端点描述符寄存器 38	UF0CIE38	R/W		√		不确定
00200214H	UF0 配置/接口/端点描述符寄存器 39	UF0CIE39	R/W		√		不确定
00200216H	UF0 配置/接口/端点描述符寄存器 40	UF0CIE40	R/W		√		不确定
00200218H	UF0 配置/接口/端点描述符寄存器 41	UF0CIE41	R/W		√		不确定
0020021AH	UF0 配置/接口/端点描述符寄存器 42	UF0CIE42	R/W		√		不确定
0020021CH	UF0 配置/接口/端点描述符寄存器 43	UF0CIE43	R/W		√		不确定

(4/13)

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
0020021EH	UF0 配置/接口/端点描述符寄存器 44	UF0CIE44	R/W		√		不确定
00200220H	UF0 配置/接口/端点描述符寄存器 45	UF0CIE45	R/W		√		不确定
00200222H	UF0 配置/接口/端点描述符寄存器 46	UF0CIE46	R/W		√		不确定
00200224H	UF0 配置/接口/端点描述符寄存器 47	UF0CIE47	R/W		√		不确定
00200226H	UF0 配置/接口/端点描述符寄存器 48	UF0CIE48	R/W		√		不确定
00200228H	UF0 配置/接口/端点描述符寄存器 49	UF0CIE49	R/W		√		不确定
0020022AH	UF0 配置/接口/端点描述符寄存器 50	UF0CIE50	R/W		√		不确定
0020022CH	UF0 配置/接口/端点描述符寄存器 51	UF0CIE51	R/W		√		不确定
0020022EH	UF0 配置/接口/端点描述符寄存器 52	UF0CIE52	R/W		√		不确定
00200230H	UF0 配置/接口/端点描述符寄存器 53	UF0CIE53	R/W		√		不确定
00200232H	UF0 配置/接口/端点描述符寄存器 54	UF0CIE54	R/W		√		不确定
00200234H	UF0 配置/接口/端点描述符寄存器 55	UF0CIE55	R/W		√		不确定
00200236H	UF0 配置/接口/端点描述符寄存器 56	UF0CIE56	R/W		√		不确定
00200238H	UF0 配置/接口/端点描述符寄存器 57	UF0CIE57	R/W		√		不确定
0020023AH	UF0 配置/接口/端点描述符寄存器 58	UF0CIE58	R/W		√		不确定
0020023CH	UF0 配置/接口/端点描述符寄存器 59	UF0CIE59	R/W		√		不确定
0020023EH	UF0 配置/接口/端点描述符寄存器 60	UF0CIE60	R/W		√		不确定
00200240H	UF0 配置/接口/端点描述符寄存器 61	UF0CIE61	R/W		√		不确定
00200242H	UF0 配置/接口/端点描述符寄存器 62	UF0CIE62	R/W		√		不确定
00200244H	UF0 配置/接口/端点描述符寄存器 63	UF0CIE63	R/W		√		不确定
00200246H	UF0 配置/接口/端点描述符寄存器 64	UF0CIE64	R/W		√		不确定
00200248H	UF0 配置/接口/端点描述符寄存器 65	UF0CIE65	R/W		√		不确定

(5/13)

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
0020024AH	UF0 配置/接口/端点描述符寄存器 66	UF0CIE66	R/W		√		不确定
0020024CH	UF0 配置/接口/端点描述符寄存器 67	UF0CIE67	R/W		√		不确定
0020024EH	UF0 配置/接口/端点描述符寄存器 68	UF0CIE68	R/W		√		不确定
00200250H	UF0 配置/接口/端点描述符寄存器 69	UF0CIE69	R/W		√		不确定
00200252H	UF0 配置/接口/端点描述符寄存器 70	UF0CIE70	R/W		√		不确定
00200254H	UF0 配置/接口/端点描述符寄存器 71	UF0CIE71	R/W		√		不确定
00200256H	UF0 配置/接口/端点描述符寄存器 72	UF0CIE72	R/W		√		不确定
00200258H	UF0 配置/接口/端点描述符寄存器 73	UF0CIE73	R/W		√		不确定
0020025AH	UF0 配置/接口/端点描述符寄存器 74	UF0CIE74	R/W		√		不确定
0020025CH	UF0 配置/接口/端点描述符寄存器 75	UF0CIE75	R/W		√		不确定
0020025EH	UF0 配置/接口/端点描述符寄存器 76	UF0CIE76	R/W		√		不确定
00200260H	UF0 配置/接口/端点描述符寄存器 77	UF0CIE77	R/W		√		不确定
00200262H	UF0 配置/接口/端点描述符寄存器 78	UF0CIE78	R/W		√		不确定
00200264H	UF0 配置/接口/端点描述符寄存器 79	UF0CIE79	R/W		√		不确定
00200266H	UF0 配置/接口/端点描述符寄存器 80	UF0CIE80	R/W		√		不确定
00200268H	UF0 配置/接口/端点描述符寄存器 81	UF0CIE81	R/W		√		不确定
0020026AH	UF0 配置/接口/端点描述符寄存器 82	UF0CIE82	R/W		√		不确定
0020026CH	UF0 配置/接口/端点描述符寄存器 83	UF0CIE83	R/W		√		不确定
0020026EH	UF0 配置/接口/端点描述符寄存器 84	UF0CIE84	R/W		√		不确定
00200270H	UF0 配置/接口/端点描述符寄存器 85	UF0CIE85	R/W		√		不确定
00200272H	UF0 配置/接口/端点描述符寄存器 86	UF0CIE86	R/W		√		不确定
00200274H	UF0 配置/接口/端点描述符寄存器 87	UF0CIE87	R/W		√		不确定

(6/13)

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
00200276H	UF0 配置/接口/端点描述符寄存器 88	UF0CIE88	R/W		√		不确定
00200278H	UF0 配置/接口/端点描述符寄存器 89	UF0CIE89	R/W		√		不确定
0020027AH	UF0 配置/接口/端点描述符寄存器 90	UF0CIE90	R/W		√		不确定
0020027CH	UF0 配置/接口/端点描述符寄存器 91	UF0CIE91	R/W		√		不确定
0020027EH	UF0 配置/接口/端点描述符寄存器 92	UF0CIE92	R/W		√		不确定
00200280H	UF0 配置/接口/端点描述符寄存器 93	UF0CIE93	R/W		√		不确定
00200282H	UF0 配置/接口/端点描述符寄存器 94	UF0CIE94	R/W		√		不确定
00200284H	UF0 配置/接口/端点描述符寄存器 95	UF0CIE95	R/W		√		不确定
00200286H	UF0 配置/接口/端点描述符寄存器 96	UF0CIE96	R/W		√		不确定
00200288H	UF0 配置/接口/端点描述符寄存器 97	UF0CIE97	R/W		√		不确定
0020028AH	UF0 配置/接口/端点描述符寄存器 98	UF0CIE98	R/W		√		不确定
0020028CH	UF0 配置/接口/端点描述符寄存器 99	UF0CIE99	R/W		√		不确定
0020028EH	UF0 配置/接口/端点描述符寄存器 100	UF0CIE100	R/W		√		不确定
00200290H	UF0 配置/接口/端点描述符寄存器 101	UF0CIE101	R/W		√		不确定
00200292H	UF0 配置/接口/端点描述符寄存器 102	UF0CIE102	R/W		√		不确定
00200294H	UF0 配置/接口/端点描述符寄存器 103	UF0CIE103	R/W		√		不确定
00200296H	UF0 配置/接口/端点描述符寄存器 104	UF0CIE104	R/W		√		不确定
00200298H	UF0 配置/接口/端点描述符寄存器 105	UF0CIE105	R/W		√		不确定
0020029AH	UF0 配置/接口/端点描述符寄存器 106	UF0CIE106	R/W		√		不确定
0020029CH	UF0 配置/接口/端点描述符寄存器 107	UF0CIE107	R/W		√		不确定
0020029EH	UF0 配置/接口/端点描述符寄存器 108	UF0CIE108	R/W		√		不确定
002002A0H	UF0 配置/接口/端点描述符寄存器 109	UF0CIE109	R/W		√		不确定

(7/13)

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
002002A2H	UF0 配置/接口/端点描述符寄存器 110	UF0CIE110	R/W		√		不确定
002002A4H	UF0 配置/接口/端点描述符寄存器 111	UF0CIE111	R/W		√		不确定
002002A6H	UF0 配置/接口/端点描述符寄存器 112	UF0CIE112	R/W		√		不确定
002002A8H	UF0 配置/接口/端点描述符寄存器 113	UF0CIE113	R/W		√		不确定
002002AAH	UF0 配置/接口/端点描述符寄存器 114	UF0CIE114	R/W		√		不确定
002002ACH	UF0 配置/接口/端点描述符寄存器 115	UF0CIE115	R/W		√		不确定
002002AEH	UF0 配置/接口/端点描述符寄存器 116	UF0CIE116	R/W		√		不确定
002002A0H	UF0 配置/接口/端点描述符寄存器 117	UF0CIE117	R/W		√		不确定
002002A2H	UF0 配置/接口/端点描述符寄存器 118	UF0CIE118	R/W		√		不确定
002002A4H	UF0 配置/接口/端点描述符寄存器 119	UF0CIE119	R/W		√		不确定
002002A6H	UF0 配置/接口/端点描述符寄存器 120	UF0CIE120	R/W		√		不确定
002002B8H	UF0 配置/接口/端点描述符寄存器 121	UF0CIE121	R/W		√		不确定
002002BAH	UF0 配置/接口/端点描述符寄存器 122	UF0CIE122	R/W		√		不确定
002002BCH	UF0 配置/接口/端点描述符寄存器 123	UF0CIE123	R/W		√		不确定
002002BEH	UF0 配置/接口/端点描述符寄存器 124	UF0CIE124	R/W		√		不确定
002002C0H	UF0 配置/接口/端点描述符寄存器 125	UF0CIE125	R/W		√		不确定
002002C2H	UF0 配置/接口/端点描述符寄存器 126	UF0CIE126	R/W		√		不确定
002002C4H	UF0 配置/接口/端点描述符寄存器 127	UF0CIE127	R/W		√		不确定
002002C6H	UF0 配置/接口/端点描述符寄存器 128	UF0CIE128	R/W		√		不确定
002002C8H	UF0 配置/接口/端点描述符寄存器 129	UF0CIE129	R/W		√		不确定
002002CAH	UF0 配置/接口/端点描述符寄存器 130	UF0CIE130	R/W		√		不确定
002002CCH	UF0 配置/接口/端点描述符寄存器 131	UF0CIE131	R/W		√		不确定

(8/13)

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
002002CEH	UF0 配置/接口/端点描述符寄存器 132	UF0CIE132	R/W		√		不确定
002002D0H	UF0 配置/接口/端点描述符寄存器 133	UF0CIE133	R/W		√		不确定
002002D2H	UF0 配置/接口/端点描述符寄存器 134	UF0CIE134	R/W		√		不确定
002002D4H	UF0 配置/接口/端点描述符寄存器 135	UF0CIE135	R/W		√		不确定
002002D6H	UF0 配置/接口/端点描述符寄存器 136	UF0CIE136	R/W		√		不确定
002002D8H	UF0 配置/接口/端点描述符寄存器 137	UF0CIE137	R/W		√		不确定
002002DAH	UF0 配置/接口/端点描述符寄存器 138	UF0CIE138	R/W		√		不确定
002002DCH	UF0 配置/接口/端点描述符寄存器 139	UF0CIE139	R/W		√		不确定
002002DEH	UF0 配置/接口/端点描述符寄存器 140	UF0CIE140	R/W		√		不确定
002002E0H	UF0 配置/接口/端点描述符寄存器 141	UF0CIE141	R/W		√		不确定
002002E2H	UF0 配置/接口/端点描述符寄存器 142	UF0CIE142	R/W		√		不确定
002002E4H	UF0 配置/接口/端点描述符寄存器 143	UF0CIE143	R/W		√		不确定
002002E6H	UF0 配置/接口/端点描述符寄存器 144	UF0CIE144	R/W		√		不确定
002002E8H	UF0 配置/接口/端点描述符寄存器 145	UF0CIE145	R/W		√		不确定
002002EAH	UF0 配置/接口/端点描述符寄存器 146	UF0CIE146	R/W		√		不确定
002002ECH	UF0 配置/接口/端点描述符寄存器 147	UF0CIE147	R/W		√		不确定
002002EEH	UF0 配置/接口/端点描述符寄存器 148	UF0CIE148	R/W		√		不确定
002002F0H	UF0 配置/接口/端点描述符寄存器 149	UF0CIE149	R/W		√		不确定
002002F2H	UF0 配置/接口/端点描述符寄存器 150	UF0CIE150	R/W		√		不确定
002002F4H	UF0 配置/接口/端点描述符寄存器 151	UF0CIE151	R/W		√		不确定
002002F6H	UF0 配置/接口/端点描述符寄存器 152	UF0CIE152	R/W		√		不确定
002002F8H	UF0 配置/接口/端点描述符寄存器 153	UF0CIE153	R/W		√		不确定

(9/13)

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
002002FAH	UF0 配置/接口/端点描述符寄存器 154	UF0CIE154	R/W		√		不确定
002002FCH	UF0 配置/接口/端点描述符寄存器 155	UF0CIE155	R/W		√		不确定
002002FEH	UF0 配置/接口/端点描述符寄存器 156	UF0CIE156	R/W		√		不确定
00200300H	UF0 配置/接口/端点描述符寄存器 157	UF0CIE157	R/W		√		不确定
00200302H	UF0 配置/接口/端点描述符寄存器 158	UF0CIE158	R/W		√		不确定
00200304H	UF0 配置/接口/端点描述符寄存器 159	UF0CIE159	R/W		√		不确定
00200306H	UF0 配置/接口/端点描述符寄存器 160	UF0CIE160	R/W		√		不确定
00200308H	UF0 配置/接口/端点描述符寄存器 161	UF0CIE161	R/W		√		不确定
0020030AH	UF0 配置/接口/端点描述符寄存器 162	UF0CIE162	R/W		√		不确定
0020030CH	UF0 配置/接口/端点描述符寄存器 163	UF0CIE163	R/W		√		不确定
0020030EH	UF0 配置/接口/端点描述符寄存器 164	UF0CIE164	R/W		√		不确定
00200310H	UF0 配置/接口/端点描述符寄存器 165	UF0CIE165	R/W		√		不确定
00200312H	UF0 配置/接口/端点描述符寄存器 166	UF0CIE166	R/W		√		不确定
00200314H	UF0 配置/接口/端点描述符寄存器 167	UF0CIE167	R/W		√		不确定
00200316H	UF0 配置/接口/端点描述符寄存器 168	UF0CIE168	R/W		√		不确定
00200318H	UF0 配置/接口/端点描述符寄存器 169	UF0CIE169	R/W		√		不确定
0020031AH	UF0 配置/接口/端点描述符寄存器 170	UF0CIE170	R/W		√		不确定
0020031CH	UF0 配置/接口/端点描述符寄存器 171	UF0CIE171	R/W		√		不确定
0020031EH	UF0 配置/接口/端点描述符寄存器 172	UF0CIE172	R/W		√		不确定
00200320H	UF0 配置/接口/端点描述符寄存器 173	UF0CIE173	R/W		√		不确定
00200322H	UF0 配置/接口/端点描述符寄存器 174	UF0CIE174	R/W		√		不确定
00200324H	UF0 配置/接口/端点描述符寄存器 175	UF0CIE175	R/W		√		不确定

(10/13)

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
00200326H	UF0 配置/接口/端点描述符寄存器 176	UF0CIE176	R/W		√		不确定
00200328H	UF0 配置/接口/端点描述符寄存器 177	UF0CIE177	R/W		√		不确定
0020032AH	UF0 配置/接口/端点描述符寄存器 178	UF0CIE178	R/W		√		不确定
0020032CH	UF0 配置/接口/端点描述符寄存器 179	UF0CIE179	R/W		√		不确定
0020032EH	UF0 配置/接口/端点描述符寄存器 180	UF0CIE180	R/W		√		不确定
00200330H	UF0 配置/接口/端点描述符寄存器 181	UF0CIE181	R/W		√		不确定
00200332H	UF0 配置/接口/端点描述符寄存器 182	UF0CIE182	R/W		√		不确定
00200334H	UF0 配置/接口/端点描述符寄存器 183	UF0CIE183	R/W		√		不确定
00200336H	UF0 配置/接口/端点描述符寄存器 184	UF0CIE184	R/W		√		不确定
00200338H	UF0 配置/接口/端点描述符寄存器 185	UF0CIE185	R/W		√		不确定
0020033AH	UF0 配置/接口/端点描述符寄存器 186	UF0CIE186	R/W		√		不确定
0020033CH	UF0 配置/接口/端点描述符寄存器 187	UF0CIE187	R/W		√		不确定
0020033EH	UF0 配置/接口/端点描述符寄存器 188	UF0CIE188	R/W		√		不确定
00200340H	UF0 配置/接口/端点描述符寄存器 189	UF0CIE189	R/W		√		不确定
00200342H	UF0 配置/接口/端点描述符寄存器 190	UF0CIE190	R/W		√		不确定
00200344H	UF0 配置/接口/端点描述符寄存器 191	UF0CIE191	R/W		√		不确定
00200346H	UF0 配置/接口/端点描述符寄存器 192	UF0CIE192	R/W		√		不确定
00200348H	UF0 配置/接口/端点描述符寄存器 193	UF0CIE193	R/W		√		不确定
0020034AH	UF0 配置/接口/端点描述符寄存器 194	UF0CIE194	R/W		√		不确定
0020034CH	UF0 配置/接口/端点描述符寄存器 195	UF0CIE195	R/W		√		不确定
0020034EH	UF0 配置/接口/端点描述符寄存器 196	UF0CIE196	R/W		√		不确定

(11/13)

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
00200350H	UF0 配置/接口/端点描述符寄存器 197	UF0CIE197	R/W		√		不确定
00200352H	UF0 配置/接口/端点描述符寄存器 198	UF0CIE198	R/W		√		不确定
00200354H	UF0 配置/接口/端点描述符寄存器 199	UF0CIE199	R/W		√		不确定
00200356H	UF0 配置/接口/端点描述符寄存器 200	UF0CIE200	R/W		√		不确定
00200358H	UF0 配置/接口/端点描述符寄存器 201	UF0CIE201	R/W		√		不确定
0020035AH	UF0 配置/接口/端点描述符寄存器 202	UF0CIE202	R/W		√		不确定
0020035CH	UF0 配置/接口/端点描述符寄存器 203	UF0CIE203	R/W		√		不确定
0020035EH	UF0 配置/接口/端点描述符寄存器 204	UF0CIE204	R/W		√		不确定
00200360H	UF0 配置/接口/端点描述符寄存器 205	UF0CIE205	R/W		√		不确定
00200362H	UF0 配置/接口/端点描述符寄存器 206	UF0CIE206	R/W		√		不确定
00200364H	UF0 配置/接口/端点描述符寄存器 207	UF0CIE207	R/W		√		不确定
00200366H	UF0 配置/接口/端点描述符寄存器 208	UF0CIE208	R/W		√		不确定
00200368H	UF0 配置/接口/端点描述符寄存器 209	UF0CIE209	R/W		√		不确定
0020036AH	UF0 配置/接口/端点描述符寄存器 210	UF0CIE210	R/W		√		不确定
0020036CH	UF0 配置/接口/端点描述符寄存器 211	UF0CIE211	R/W		√		不确定
0020036EH	UF0 配置/接口/端点描述符寄存器 212	UF0CIE212	R/W		√		不确定
00200370H	UF0 配置/接口/端点描述符寄存器 213	UF0CIE213	R/W		√		不确定
00200372H	UF0 配置/接口/端点描述符寄存器 214	UF0CIE214	R/W		√		不确定
00200374H	UF0 配置/接口/端点描述符寄存器 215	UF0CIE215	R/W		√		不确定
00200376H	UF0 配置/接口/端点描述符寄存器 216	UF0CIE216	R/W		√		不确定
00200378H	UF0 配置/接口/端点描述符寄存器 217	UF0CIE217	R/W		√		不确定

(12/13)

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
0020037AH	UF0 配置/接口/端点描述符寄存器 218	UF0CIE218	R/W		√		不确定
0020037CH	UF0 配置/接口/端点描述符寄存器 219	UF0CIE219	R/W		√		不确定
0020037EH	UF0 配置/接口/端点描述符寄存器 220	UF0CIE220	R/W		√		不确定
00200380H	UF0 配置/接口/端点描述符寄存器 221	UF0CIE221	R/W		√		不确定
00200382H	UF0 配置/接口/端点描述符寄存器 222	UF0CIE222	R/W		√		不确定
00200384H	UF0 配置/接口/端点描述符寄存器 223	UF0CIE223	R/W		√		不确定
00200386H	UF0 配置/接口/端点描述符寄存器 224	UF0CIE224	R/W		√		不确定
00200388H	UF0 配置/接口/端点描述符寄存器 225	UF0CIE225	R/W		√		不确定
0020038AH	UF0 配置/接口/端点描述符寄存器 226	UF0CIE226	R/W		√		不确定
0020038CH	UF0 配置/接口/端点描述符寄存器 227	UF0CIE227	R/W		√		不确定
0020038EH	UF0 配置/接口/端点描述符寄存器 228	UF0CIE228	R/W		√		不确定
00200390H	UF0 配置/接口/端点描述符寄存器 229	UF0CIE229	R/W		√		不确定
00200392H	UF0 配置/接口/端点描述符寄存器 230	UF0CIE230	R/W		√		不确定
00200394H	UF0 配置/接口/端点描述符寄存器 231	UF0CIE231	R/W		√		不确定
00200396H	UF0 配置/接口/端点描述符寄存器 232	UF0CIE232	R/W		√		不确定
00200398H	UF0 配置/接口/端点描述符寄存器 233	UF0CIE233	R/W		√		不确定
0020039AH	UF0 配置/接口/端点描述符寄存器 234	UF0CIE234	R/W		√		不确定
0020039CH	UF0 配置/接口/端点描述符寄存器 235	UF0CIE235	R/W		√		不确定
0020039EH	UF0 配置/接口/端点描述符寄存器 236	UF0CIE236	R/W		√		不确定
002003A0H	UF0 配置/接口/端点描述符寄存器 237	UF0CIE237	R/W		√		不确定
002003A2H	UF0 配置/接口/端点描述符寄存器 238	UF0CIE238	R/W		√		不确定

(13/13)

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
002003A4H	UF0 配置/接口/端点描述符寄存器 239	UF0CIE239	R/W		√		不确定
002003A6H	UF0 配置/接口/端点描述符寄存器 240	UF0CIE240	R/W		√		不确定
002003A8H	UF0 配置/接口/端点描述符寄存器 241	UF0CIE241	R/W		√		不确定
002003AAH	UF0 配置/接口/端点描述符寄存器 242	UF0CIE242	R/W		√		不确定
002003ACH	UF0 配置/接口/端点描述符寄存器 243	UF0CIE243	R/W		√		不确定
002003AEH	UF0 配置/接口/端点描述符寄存器 244	UF0CIE244	R/W		√		不确定
002003B0H	UF0 配置/接口/端点描述符寄存器 245	UF0CIE245	R/W		√		不确定
002003B2H	UF0 配置/接口/端点描述符寄存器 246	UF0CIE246	R/W		√		不确定
002003B4H	UF0 配置/接口/端点描述符寄存器 247	UF0CIE247	R/W		√		不确定
002003B6H	UF0 配置/接口/端点描述符寄存器 248	UF0CIE248	R/W		√		不确定
002003B8H	UF0 配置/接口/端点描述符寄存器 249	UF0CIE249	R/W		√		不确定
002003BAH	UF0 配置/接口/端点描述符寄存器 250	UF0CIE250	R/W		√		不确定
002003BCH	UF0 配置/接口/端点描述符寄存器 251	UF0CIE251	R/W		√		不确定
002003BEH	UF0 配置/接口/端点描述符寄存器 252	UF0CIE252	R/W		√		不确定
002003C0H	UF0 配置/接口/端点描述符寄存器 253	UF0CIE253	R/W		√		不确定
002003C2H	UF0 配置/接口/端点描述符寄存器 254	UF0CIE254	R/W		√		不确定
002003C4H	UF0 配置/接口/端点描述符寄存器 255	UF0CIE255	R/W		√		不确定

(4) 桥寄存器

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
00200400H	桥中断控制寄存器	BRGINTT	R/W			√	不确定
00200402H	桥中断使能寄存器	BRGINTE	R/W			√	不确定
00200404H	EPC 宏控制寄存器	EPCCLT	R/W			√	不确定
00200408H	CPU I/F 总线控制寄存器	CPUBCTL	R/W			√	不确定

(5) DMA 寄存器

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
00200500H	EP1 DMA 控制寄存器 1	UF0E1DC1	R/W			√	0000H
00200502H	EP1 DMA 控制寄存器 2	UF0E1DC2	R/W			√	0000H
00200504H	EP2 DMA 控制寄存器 1	UF0E2DC1	R/W			√	0000H
00200506H	EP2 DMA 控制寄存器 2	UF0E2DC2	R/W			√	0000H
00200508H	EP3 DMA 控制寄存器 1	UF0E3DC1	R/W			√	0000H
0020050AH	EP3 DMA 控制寄存器 2	UF0E3DC2	R/W			√	0000H
0020050CH	EP4 DMA 控制寄存器 1	UF0E4DC1	R/W			√	0000H
0020050EH	EP4 DMA 控制寄存器 2	UF0E4DC2	R/W			√	0000H

(6) Bulk-in 寄存器

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
00201000H	UF0 EP1 bulk-in 传输数据寄存器	UF0EP1BI	R/W			√	0000H
00202000H	UF0 EP3 bulk-in 传输数据寄存器	UF0EP3BI	R/W			√	0000H

(7) Bulk-out 寄存器

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
00210000H	UF0 EP2 bulk-out 传输数据寄存器	UF0EP2BO	R			√	0000H
00220000H	UF0 EP4 bulk-out 传输数据寄存器	UF0EP4BO	R			√	0000H

(8) 外设控制寄存器

地址	功能寄存器名称	符号	R/W	可操作的位			默认值
				1	8	16	
00240000H	USBF DMA 请求使能寄存器	UFDRQEN	R/W		√	√	0000H

21.6.3 EPC控制寄存器

(1) UF0 EP0NAK 寄存器 (UF0E0N)

该寄存器控制端点 0 的 NAK（除自动执行的请求之外）。
该寄存器可以按 8 位进行读写（但是位 0 为只读位）。
设置 UF0FIC0 和 UF0FIC1 寄存器之后，需要用五个 USB 时钟来反映该寄存器的状态。 因此，如果要正确读取，访问 UF0FIC0 和 UF0FIC1 寄存器的写入信号和访问 UF0EPS0、UF0EPS1、UF0EPS2、UF0E0N 和 UF0EN 寄存器的读取信号至少需要四个 USB 时钟间隔。
NAK 发送到端点 0 读取、端点 2 和端点 4 期间，EP0NKR 位的写入访问被忽略。

UF0E0N	7	6	5	4	3	2	1	0	地址	复位后
	0	0	0	0	0	0	EP0NKR	EP0NKW	00200000H	00H
位号	位名称		功能							
1	EP0NKR		该位控制端点 0 OUT 令牌的 NAK（除自动执行的请求外）。 当端点 0 正确接收数据时，该位通过硬件自动置 1。 当 FW 读取 UF0E0R 寄存器的数据完毕时（计数器的值 = 1），该位通过硬件清除为 0。 1: 发送 NAK。 0: 不发送 NAK（默认值）。 由于某些原因不应从 USB 总线接收数据时，即使 USBF 准备好接收数据，也会通过 FW 将该位设为 1。 这种情况下，USBF 继续发送 NAK 直到通过 FW 将该位清为 0 为止。 一旦 UF0E0R 寄存器被清除，该位也会被清除为 0。							
0	EP0NKW		该位表示如何控制端点 0 IN 令牌的 NAK（除自动执行请求外）。 当发送端点 0 的数据且主机正确接收到发送数据时，该位通过硬件自动清除为 0。 在该位清除之前保持 UF0E0W 寄存器的值。 因此，即使主机不能正确接收数据，做出重新发送请求的情况下，也不必重写该位。 要发送短包，务必将 UF0DEND 寄存器的 E0DED 位设为 1。当 FIFO 存满时该位自动设为 1。 一旦将 UF0DEND 寄存器的 E0DED 位设为 1，EP0NKW 位也同时自动设为 1。 1: 不发送 NAK。 0: 发送 NAK(默认值)。 在数据阶段不能正确接收 ACK 期间，如果控制传输进入状态阶段，则只要 UF0E0W 寄存器被清除，该位就被清为 0。 当通过 FW 清除 UF0E0W 时该位也会被清为 0。							

使用 IN/OUT 令牌的 SETUP 处理步骤说明如下。

(a) 使用 IN 令牌时（除硬件自动执行的请求外）

接收 CPUDEC 中断之后且在从 UF0E0ST 寄存器读取数据之前，应使用 FW 将 UF0IS1 寄存器的 PROT 位清除为 0。然后，根据请求执行处理，并且如果需要通过 IN 令牌返回数据，则将数据写入到 UF0E0W 寄存器。完成写入之后应确认 UF0IS1 寄存器的 PROT 位为 0，并将 UF0DEND 寄存器的 E0DED 位设为 1。硬件在 EP0NKR 位设为 1 之后的第一个 IN 令牌处发出数据。如果 UF0IS1 寄存器的 PROT 位为 1，则表示在完成控制传输之前又发生了 SETUP 处理。此时，可通过将 UF0IC1 寄存器的 PROTC 位清除为 0 来将 UF0IS1 寄存器的 PROT 清 0，然后再次从 UF0E0ST 寄存器读取数据。稍后可读取接收到的请求。

(b) 使用 OUT 令牌时（除硬件自动执行的请求外）

接收 CPUDEC 中断之后且在从 UF0E0ST 寄存器读取数据之前，应使用 FW 清除 UF0IS1 寄存器的 PROT 位。从 UF0E0R 寄存器读取数据之前应确认 UF0IS1 寄存器的 PROT 位为 0。如果 PROT 位为 1，这意味着保留了无效的数据。通过 FW 清除 FIFO（EP0NKR 位自动清除为 0）。如果 UF0IS1 寄存器的 PROT 位为 0，则读取 UF0E0L 寄存器数据和 UF0E0R 寄存器内所设置的所有数据，当从 UF0E0R 寄存器读取数据完毕时（UF0E0R 寄存器的计数器值清为 0 时），硬件自动将 EP0NKR 位清除为 0。

(2) UF0 EP0NAKALL 寄存器 (UF0E0NA)

该寄存器控制端点 0 所有请求的 NAK。它对自动执行的请求同样有效。
该寄存器可按 8 位为单位进行读写。

								地址	复位后
UF0E0NA	7	6	5	4	3	2	1	0	
	0	0	0	0	0	0	0	EP0NKA	00200002H 00H
位号	位名称		功能						
0	EP0NKA		该位控制除了到端点 0 的 SETUP 处理之外的 NAK（包括自动执行请求）。该位由 FW 操作。 1: 发送 NAK。 0: 不发送 NAK（默认值）。 当要改变用于自动执行请求的数据时，该寄存器用来防止 FW 的写访问和对 SIE 的读取访问之间的冲突。对 SIE 进行访问期间，该位推迟呈现来自于 FW 的写入访问。由 FW 重写请求数据寄存器之前，应确认该位已正确设为 1。 仅在以下情况下该位呈现为 1。 • USBF 复位之后且尚未接收到 SETUP 令牌前之际 • 接收总线复位且尚未接收到 SETUP 令牌前之际 • 已检测到 SETUP 令牌的 PID • 已转变到状态阶段 除正在接收 IN 令牌且正在进行 NAK 响应外，该位都会即刻清除为 0。 端点 0 传输期间，在上述 4 种情况下 EP0NKA 都设置为 1，但是在端点 0 没有传送数据期间，它在数据写入该位后即刻呈现的。						

(3) UF0 EPNAK 寄存器 (UF0EN)

该寄存器控制除端点 0 之外的端点的 NAK。

该寄存器可以按 8 位进行读写（但是位 5，4，1 和 0 为只读位）。

BKO2NK 位仅当 UF0ENM 寄存器的 BKO2NKM 位为 1 时可以写入，而 BKO1NK 位仅当 UF0ENM 寄存器的 BKO1NKM 位为 1 时可以写入。

如果 UF0EnIM 寄存器（n = 1 至 4，7）的设置和接口的当前设置不支持各个端点，则相关位无效。

设置 UF0FIC0 和 UF0FIC1 寄存器之后，需要用五个 USB 时钟来呈现该寄存器的状态。 因此，如果要正确读取，访问 UF0FIC0 和 UF0FIC1 寄存器的写入信号和访问 UF0EPS0、UF0EPS1、UF0EPS2、UF0E0N 和 UF0EN 寄存器的读取信号至少需要四个 USB 时钟间隔。

NAK 发送到端点 0 读取、端点 2 和端点 4 期间，忽略 BKO1NK 和 BKO2NK 位的写入访问。

一定要将位 7 和 6 清为 00。” 如果设置为 1，则操作将不能保障。

(1/4)

UF0EN	7	6	4	3	2	1	0	地址	复位后
	0	0	0	IT1NK	BKO2NK	BKO1NK	BKI2NK	BKI1NK	00200004H 00H

位号	位名称	功能
4	IT1NK	该位控制到端点 7 的 NAK（中断 1 传输） 当 UF0INT1 寄存器由于写入数据变为饱和时，该位自动设为 1 并开始发送。 为了发送不使得 FIFO 饱和的短包，要将 UF0DEND 寄存器的 IT1DEND 位设为 1。只要将 IT1DEND 设为 1，该位也自动设为 1。 1: 不发送 NAK。 0: 发送 NAK(默认值)。 当 UF0INT1 寄存器被清除时，该位也会被清除为 0。

位号	位名称	功能
3	BKO2NK	该位控制到端点 4 的 NAK (bulk 2 传输 (OUT)) 1: 发送 NAK。 0: 不发送 NAK (默认值)。 仅当连接到 UF0BO2 寄存器的 SIE 侧的 FIFO (64-字节的存储体配置的 FIFO) 不能接收数据时, 才将该位设为 1。当执行转换操作时将其清为 0。当满足以下条件时, 会改变存储体 (转换操作)。 • 正确接收到的数据存储在 SIE 侧连接的 FIFO 中。 • 与 CPU 侧相连的 FIFO 计数器的值为 0 (读取完毕)。 当 UF0BO2L 寄存器接收 BLKO2DT 中断请求并从 UF0BO2 寄存器读取与该数据的值一样多的数据时, FW 应用于读取 UF0BO2L 寄存器的值。即使 USBF 准备接收数据, 但由于某些原因不从 USB 总线接收数据时, 通过 FW 将该位设为 1。此时, USBF 继续发送 NAK 直到 FW 将该位清为 0。一旦 UF0BO2 寄存器被清除, 该位也会被立即清为 0。
2	BKO1NK	该位控制到端点 2 的 NAK (bulk 1 传输 (OUT)) 1: 发送 NAK。 0: 不发送 NAK (默认值)。 仅当连接到 UF0BO1 寄存器的 SIE 侧的 FIFO (64-字节配置的 FIFO 区) 不能接收数据时, 才将该位设为 1。当执行转换操作时将其清为 0。当满足以下条件时, 会改变存储区 (转换操作)。 • 正确接收到的数据存储在 SIE 侧连接的 FIFO 中。 • 与 CPU 侧相连的 FIFO 计数器的值为 0 (读取完毕)。 当 UF0BO1L 寄存器接收 BLKO1DT 中断请求并从 UF0BO1 寄存器读取所有数据时, 应使用 FW 读取 UF0BO1L 寄存器内数据。即使 USBF 准备接收数据, 但由于某些原因不从 USB 总线接收数据时, 通过 FW 将该位设为 1。此时, USBF 继续发送 NAK 直到 FW 将该位清为 0。一旦 UF0BO1 寄存器被清除, 该位也会被立即清为 0。

- 注意事项
1. 在 PIO 模式下, 如果从 UF0BO2 寄存器读取数据期间使能 DMA, 则会立即产生一个 DMA 请求。
 2. 如果在 DMA 传输模式读取 CPU 端的 FIFO 的最末一个数据, DMA 请求信号会变为无效。
 3. 如果在 DMA 传输模式接收到 TC 信号, 则 DMA 请求信号变为无效。

位号	位名称	功能
1	BKI2NK	该位控制到端点 3 的 NAK (bulk 2 传输 (IN)) 1: 不发送 NAK。 0: 发送 NAK(默认值)。 仅当连接到 UF0BI2 寄存器的 SIE 侧的 FIFO (64-字节配置的 FIFO 区) 不能接收数据时, 才将该位清为 0。 执行转换操作时将其设为 1 (一直保持 UF0BI2 寄存器的数据直到发送正确完成)。 当满足以下条件时, 会改变存储区 (转换操作)。 • 数据正确写入与 CPU 总线端相连的 FIFO (已完成写入且 FIFO 已满或 UF0DEND 被设置)。 • 与 SIE 侧相连的 FIFO 计数器的值为 0。 当 CPU 端的 FIFO 存满时该位自动设为 1 并开始数据发送, 而且, 写入数据到 FIFO 将导致执行 FIFO 转换操作。 但是, 如果 CPU 端的 FIFO 在 UF0DEND 寄存器的 BKI2T 位清为 0 期间, CPU 侧的 FIFO 被 DMA 写满, 则不执行转换操作, 因为一直到 UF0DEND 寄存器的 BKI2DED 位设为 1 前, 都不满足转换条件。为了发送不使 CPU 侧的 FIFO 存满的短包, 需在数据写入完成之后将 BKI2DED 位设为 1。 将 BKI2DED 位设为 1 时, 执行转换操作且同时自动将该位设为 1。一旦 UF0BI2 寄存器被清除该位也会被清为 0。

- 注意事项**
1. 在 PIO 模式, 如果在向 UF0BI2 寄存器写入数据期间使能 DMA, 则会立即产生一个 DMA 请求。
 2. 如果在 DMA 传输模式接写入 64-字节的数据, 则 DMA 请求信号变为无效。 如果随后将 BKI2NK 位设为 1, 则数据会与 IN 令牌同步发送。 一旦 FIFO 被转换, 只要不屏蔽 DMA 请求则 DMA 请求会重新变为有效。 如果不设置 BKI2NK 位, 则即使接收到 IN 令牌也不会发送数据。 此时, 将 UF0DEND 寄存器的 BKI2DED 位设为 1。
 3. 如果在 DMA 传输模式接收到 TC 信号, 则 DMA 请求信号变为无效。 同时, 屏蔽 DMA 请求。 如果不设置 BKI2NK 位, 则即使接收 IN 令牌也不会发送数据。 当通过 FW 将 UF0DEND 寄存器的 BKI2DED 位设为 1 时, 数据与 IN 令牌同步发送。 要再次进行 DMA 传输, 需解除 DMA 请求的屏蔽。

位号	位名称	功能
0	BKI1NK	该位控制到端点 1 的 NAK (bulk 1 传输 (IN)) 1: 不发送 NAK。 0: 发送 NAK(默认值)。 仅当连接到 UF0BI1 寄存器的 SIE 侧的 FIFO (64-字节配置的 FIFO 区) 不能接收数据时, 才将该位清为 0。 执行转换操作时将其设为 1 (一直保持 UF0BI1 寄存器的数据直到发送正确完成)。 当满足以下条件时, 会改变存储体 (转换操作)。 • 数据正确写入与 CPU 总线端相连的 FIFO (已完成写入且 FIFO 已满或 UF0DEND 被设置)。 • 与 SIE 侧相连的 FIFO 计数器的值为 0。 当 CPU 侧的 FIFO 变满时该位自动设为 1 并开始数据发送, 且由于写入数据到 FIFO 会执行 FIFO 转换操作。 但是, 如果 CPU 侧的 FIFO 在 UF0DEND 寄存器的 BKI1T 位清为 0 期间, 由于向 CPU 端的 FIFO 写入数据使其变为满, 则不执行转换操作, 因为一直到 UF0DEND 寄存器的 BKI1DED 位设为 1 都不满足转换条件。为了发送不使 CPU 侧的 FIFO 存满的短包, 需在数据写入完成之后将 BKI1DED 位设为 1。 将 BKI1DED 位设为 1 时, 执行转换操作且同时自动将该位设为 1。一旦 UF0BI1 寄存器被清除该位也会被清为 0。

- 注意事项
1. 在 PIO 模式, 如果在向 UF0BI1 寄存器写入数据期间使能 DMA, 则会立即产生一个 DMA 请求。
 2. 如果在 DMA 传输模式接写入 64-字节的数据, 则 DMA 请求信号变为无效。 如果随后将 BKI1NK 位设为 1, 则数据会与 IN 令牌同步发送。 一旦 FIFO 被转换, 只要不屏蔽 DMA 请求则 DMA 请求会重新变为有效。 如果不设置 BKI1NK 位, 则即使接收到 IN 令牌也不会发送数据。 此时, 将 UF0DEND 寄存器的 BKI1DED 位设为 1。
 3. 如果在 DMA 传输模式接收到 TC 信号, 则 DMA 请求信号变为无效。 同时, 屏蔽 DMA 请求。 如果不设置 BKI1NK 位, 则即使接收 IN 令牌也不会发送数据。 当通过 FW 将 UF0DEND 寄存器的 BKI1DED 位设为 1 时, 数据与 IN 令牌同步发送。 要再次进行 DMA 传输, 需解除 DMA 请求的屏蔽。

(4) UF0 EPNAK 屏蔽寄存器 (UF0ENM)

该寄存器控制屏蔽对 UF0EN 寄存器的写入访问。
该寄存器可按 8 位为单位进行读写。
一定要将位 7 至 4，位 1 和 0 清为“0”。 如果设置为 1，则操作将不能保障。

	7	6	5	4	3	2	1	0	地址	复位后
UF0ENM	0	0	0	0	BKO2NKM	BKO1NKM	0	0	00200006H	00H
位号	位名称		功能							
3	BKO2NKM		该位表示是否屏蔽了对 UF0EN 寄存器的位 3（BKO2NK）的写入访问。 1: 未屏蔽。 0: 屏蔽（默认值）。							
2	BKO1NKM		该位表示是否屏蔽了对 UF0EN 寄存器的位 2（BKO1NK）的写入访问。 1: 未屏蔽。 0: 屏蔽（默认值）。							

(5) UF0 SNDSIE 寄存器 (UF0SDS)

该寄存器执行诸如‘无握手’的操作。 它可以直接对 SIE 的引脚进行操作。
该寄存器可按 8 位为单位进行读写。
务必将位 2 清为“0”。 如果设置为 1，则操作将不能保障。

UF0SDS	7	6	5	4	3	2	1	0	地址	复位后
	0	0	0	0	SNDSTL	0	0	RSUMIN	00200008H	00H

位号	位名称	功能
3	SNDSTL	该位使得端点 0 产生一个 STALL 握手信号。 如果系统不支持 CPUDEC 处理请求，则将该位设为 1 会导致产生一个 STALL 握手信号响应。 如果由 SET_CONFIGURATION 或 SET_INTERFACE 请求发送了一个不被支持的 wValue，则硬件将该位设为 1。如果在端点 0 由于自动执行请求的溢出发生了问题，则也会将该位设为 1。但是，不会将 UF0E0SL 寄存器的 E0HALT 位设为 1。 1: 响应 STALL 握手信号。 0: 不响应 STALL 握手信号（默认值）。 当接收下一个 SETUP 令牌时，该位被清为 0 且响应总线的握手信号不是 STALL。 为了通过 FW 将 SNDSTL 位设为 1，一定不要将数据写入 UF0E0W 寄存器。 根据该位设置的时刻，STALL 即时做出响应，而可能会在 NAK 响应之后的下一次传输时才做出 STALL 响应。 该位为 1 时，仅当 FW 执行请求执行期间设置该位有效。当接收下一个 SETUP 令牌时该位自动清除为 0。 备注 SNDSTL 位仅对 FW-执行请求有效。
0	RSUMIN	该位将重传信号输出到 USB 总线。除非将 UF0DSL 寄存器的 RMWK 位设为 1，否则写入该位无效。 1: 产生恢复信号。 0: 不产生重传信号（默认值）。 该位设为 1 期间，继续产生重传信号。 指定时间过后通过 FW 将该位清为 0。 因为信号在时钟处被内部采样，仅在提供 CLK 期间可以保证操作。 当系统 CLK 停止时，务必加以注意。

(6) UF0 CLR 请求寄存器 (UF0CLR)

该寄存器表示接收的 CLEAR_FEATURE 请求的目标。
本寄存器为只读寄存器，可以按 8 位读取。
该寄存器仅在产生中断请求时有效。 状态阶段完成之后，各个位被设为 1，读取该寄存器时各位自动清除为 0。
如果 UF0EnIM 寄存器 (n = 1 至 4, 7) 的设置和接口的当前设置不支持各个端点，则相关位无效。

								7	6	5	4	3	2	1	0	地址	复位后
UF0CLR								CLREP8	CLREP7	CLREP4	CLREP3	CLREP2	CLREP1	CLREP0	CLRDEV	0020000AH	00H

位号	位名称	功能
7 至 1	CLREPN	这些位表示接收到 CLEAR_FEATURE 端点 n 请求并自动进行处理。 1: 自动处理 0: 不自动处理（默认值）
0	CLRDEV	该位表示接收到 CLEAR_FEATURE 设备请求并自动进行处理。 1: 自动处理 0: 不自动处理（默认值）

备注

n = 0 至 4, 7

(7) UF0 SET 请求寄存器 (UF0SET)

该寄存器表示自动处理 SET_XXXX (除 SET_INTERFACE 之外) 请求的目标。
本寄存器为只读寄存器, 可以按 8 位读取。
该寄存器仅在产生中断请求时有效。 状态阶段完成之后, 各个位被设为 1, 读取该寄存器时各位自动清除为 0。

	7	6	5	4	3	2	1	0	地址	复位后
UF0SET	SETCON	0	0	0	0	SETEP	0	SETDEV	0020000CH	00H
	位号	位名称	功能							
	7	SETCON	该位表示接收了 SET_CONFIGURATION 请求并自动进行处理。 1: 自动处理 0: 不自动处理 (默认值)							
	2	SETEP	该位表示接收了 SET_FEATURE 端点 n 请求 (n = 0 至 4, 7) 并自动进行处理。 1: 自动处理 0: 不自动处理 (默认值)							
	0	SETDEV	该位表示接收了 SET_FEATURE 设备请求并自动进行处理。 1: 自动处理 0: 不自动处理 (默认值)							

设置 UF0FIC0 和 UF0FIC1 寄存器之后，需要用五个 USB 时钟来反映该寄存器的状态。因此，如果要正确读取，访问 UF0FIC0 和 UF0FIC1 寄存器的写入信号和访问 UF0EPS0、UF0EPS1、UF0EPS2、UF0E0N 和 UF0EN 寄存器的读取信号至少需要四个 USB 时钟间隔。

用户手册初稿 U19181CA2V0UD

位号	位名称	功能
1	EP0W	该位表示数据位于 UF0E0W 寄存器 (FIFO)。即使不向 UF0E0W n 寄存器写入数据 (空数据传输)，通过将 UF0DEND 寄存器的 E0DED 位设为 1，也可以创建 UF0E0W 寄存器中数据所处的状态。一旦将 UF0DEND 寄存器的 E0DED 位设为 1，即使 UF0E0W 寄存器的计数器为 0 时，也会通过硬件将该位设为 1。正确发送之后将其清为 0。 1: 有数据位于寄存器。 0: 寄存器中没有数据 (默认值)。
0	EP0R	该位表示数据位于 UF0E0R 寄存器 (FIFO)。完成对 UF0E0R 寄存器 (FIFO) 的读取时 (计数器值 = 0)，该位通过硬件自动清为 0。如果接收空数据则不将其设为 1。 1: 有数据位于寄存器。 0: 寄存器中没有数据 (默认值)。

(9) UF0 EP 状态 1 寄存器 (UF0EPS1)

该寄存器表示 USB 总线的状态和寄存器数据的存在或缺失。
本寄存器为只读寄存器，可以按 8 位读取。

	7	6	5	4	3	2	1	0	地址	复位后
UF0EPS1	RSUM	0	0	0	0	0	0	0	00200010H	00H
位号	位名称	功能								
7	RSUM	该位表示 USB 总线处于恢复状态。 该位仅在产生中断请求时有效。 1: 中止状态 0: 重传状态（默认值） 因为使用时钟内部进行采样，所以仅在提供 CLK 时可以保证操作。 当系统 CLK 停止时，务必注意。 即使 CLK 停止时 SIE 的 INTUSBF 信号也可以运行。 因此，它可以通过令中断控制寄存器（UFIC1）有效或降低 USBF 的 CLK 获得支持。 当读取该位时，将其自动清零。								

(10) UF0 EP 状态 2 寄存器 (UF0EPS2)

该寄存器表示 USB 总线的状态和寄存器数据是否存在。

本寄存器为只读寄存器，可以按 8 位读取。

如果 UF0EnIM 寄存器 (n = 1 至 4, 7) 的设置和接口的当前设置不支持各个端点，则相关位无效。

								7	6	5	4	3	2	1	0	地址	复位后
UF0EPS2								0	HALT8	HALT7	HALT4	HALT3	HALT2	HALT1	HALT0	00200012H	00H

位号	位名称	功能
6 至 0	HALTn	这些位表示端点 n 当前停止。 当满足诸如发生溢出和未定义请求的接收等停止条件时，将这些位设为 1。 通过硬件将这些位自动设为 1。 1: 端点停止。 0: 端点不停止（默认值）。 一旦由于发生溢出或接收未定义的请求引起 HALT0 位置 1，就将 SNDSTL 位设为 1。 如果该状态下接收下一个 SETUP 令牌，则将 SNDSTL 位清除为 0，于是也将 HALT0 位清除为 0。如果通过 SET_FEATURE 端点 0 请求使端点 0 停止，则一直到接收到 CLEAR_FEATURE 端点 0 请求或使用 FW 清除 Halt 特性之前，该位不被清为 0。 如果接收到 GET_STATUS 端点 0、CLEAR_FEATURE 端点 0 或 SET_FEATURE 端点 0，或由于 CPUDEC 中断请求接收了要使用 FW 处理的请求，则将 HALT0 屏蔽并将其清 0，一直到接收下一个 SETUP 令牌为止。 一直到端点 n 接收 CLEAR_FEATURE 端点请求，通过接口的与端点相连的 SET_INTERFACE 或 SET_CONFIGURATION 请求清除 Halt 特性，或使用 FW 清除 Halt 特性之前，不会将 HALTn 位清除为 0。 正确处理 SET_INTERFACE 或 SET_CONFIGURATION 时，即使 wValue 与当前设置值相同且将这些位清为 0，请求处理之后会清除除了端点 0 之外的所有目标端点的 Halt 特性。由于 SET_INTERFACE 和 SET_CONFIGURATION 请求进行了 STALL 响应，若端点 0 的 Halt 特性被置位，则不能被清除。

备注

n = 0 至 4, 7, 8

(11) UF0 INT 状态 0 寄存器 (UF0IS0)

该寄存器表示中断源。如果改变该寄存器的内容，则 INT0B 信号会变为有效。

本寄存器为只读寄存器，可以按 8 位读取。

如果一个中断请求 (INTUSBF0) 是由 USBF 产生的，则 FW 必须读取该寄存器以识别中断源。

当将 0 写入 UF0IC0 寄存器的对应位时，该寄存器的各位被强制清除为 0。

注意事项 在 USBF 中，诸如总线复位、重传和短路等多重中断源在内部进行或运算并发生一个中断请求 (INTUSBF0)。因此，在发生多重中断源的情况下，它们在内部进行或运算并生成一个 INTUSBF0 中断请求。

例如，如果发生总线复位中断源和重传中断源，则两个中断源或运算并产生一个 INTUSBF0 中断请求信号。

在这些条件下，如果将总线复位中断源清除为 0 (UF0IC0.BUSRSTC = 0)，由于仍存有重传中断源，所以 V850ES/JG3-H 或 V850ES/JH3-H 的内部 INTUSBF0 中断请求仍可能置为 1。因此，可能不会将新的中断请求标志 (US0BIC.US0BIF) 设为 1。

此时，使用 INTUSBF0 中断服务程序对各中断请求执行清除处理之后，应再次确认 UF0IS0 和 UF0IS1 寄存器的标志状态，且如有任何的标志设为 1 的中断源，执行标志清除（仅需对所用位进行清除（不要整体清除））。

(1/2)

	7	6	5	4	3	2	1	0	地址	复位后
UF0IS0	BUSRST	RSUSPD	0	SHORT	DMAED	SETRQ	CLRRQ	EPHALT	00200020H	00H

位号	位名称	功能
7	BUSRST	该位表示发生了总线复位。 1: 发生了总线复位（产生了中断请求）。 0: 非总线复位状态（默认值）。
6	RSUSPD	该位表示发生了重传或中止状态。通过 FW 引用 UF0EPS1 寄存器的位 7。 1: 发生了重传或中止状态（产生了中断请求）。 0: 未发生重传或中止状态（默认值）。
4	SHORT	该位表示数据由 UF0BO1 或 UF0BO2 的 FIFO 区读取，且 USBSPnB 信号 (n = 2, 4) 有效。它仅在 DMA 模式下 FIFO 区存满时有效。 1: USBSPnB 信号有效（产生中断请求）。 0: USBSPnB 信号无效（默认值）。 通过 UF0DMS1 寄存器识别操作哪个端点。即使通过 FW 读取 UF0DMS1 寄存器，该位也不会自动清除为 0。

位号	位名称	功能
3	DMAED	该位表示端点 n (n = 1 至 4, 7) 的 DMA 结束 (TC) 信号有效。 1: 已输入端点 n 的 DMA 结束信号 (产生了中断请求)。 0: 未输入端点 n 的 DMA 结束信号 (默认值)。 当将该位设为 1 时, 端点 n 的 DMA 请求信号变为有效。除非 FW 使能 DMA 传输, 否则端点 n 的 DMA 请求信号不会变为有效。 使用 UF0DMS0 寄存器来确认操作实际在哪个端点执行。但是, 即使通过 FW 读取 UF0DMS0 寄存器, 该位也不会自动清除为 0。
2	SETRQ	该位表示接收了将会自动处理的 SET_XXXX 请求并自动进行处理 (XXXX = CONFIGURATION 或 FEATURE)。 1: 接收了会自动处理的 SET_XXXX 请求 (产生了中断请求)。 0: 未接收将会自动进行处理的 SET_XXXX 请求 (默认值)。 状态阶段完成之后将该位设为 1。参照 UF0SET 寄存器确认请求的目标是什么。即使通过 FW 读取 UF0SET 寄存器, 该位也不会自动清除为 0。 当接收到 SET_FEATURE 端点请求时, EPHALT 位也会设为 1。
1	CLRRQ	该位表示接收了 CLEAR_FEATURE 并自动进行了处理。 1: 已接收了 CLEAR_FEATURE 请求 (产生了中断请求)。 0: 未收到 CLEAR_FEATURE 中断请求 (默认值)。 状态阶段完成之后将该位设为 1。参照 UF0CLR 寄存器确认请求的目标是什么。即使通过 FW 读取 UF0CLR 寄存器, 该位也不会自动清除为 0。
0	EPHALT	该位表示端点已停止。 1: 端点已停止 (已产生中断请求)。 0: 端点未停止 (默认值)。 通过设置 FW 使端点停止时, 该位也会设为 1。 通过参照 UF0EPS2 寄存器确认端点停止。即使当接收到 CLEAR_FEATURE 端点, SET_INTERFACE 或 SET_CONFIGURATION 请求时, 该位不会自动清除为 0。如果在端点 0 溢出的情况下接收了下一个 SETUP 令牌, 它也不会自动清除为 0。 注意事项 即使设置了端点 0 的 Halt 特性并产生了该中断, 也会屏蔽 UF0EPS2 寄存器的位 0, 并在接收 SET_FEATURE 端点 0, CLEAR_FEATURE 端点 0 或 GET_STATUS 端点 0 请求, 或 FW-处理请求与接收除上述之外的 SETUP 令牌之间将其清除为 0。

(12) UF0 INT 状态 1 寄存器 (UF0IS1)

该寄存器表示中断源。如果改变该寄存器的内容，则 INT0B 信号会变为有效。

本寄存器为只读寄存器，可以按 8 位读取。

如果一个中断请求 (INTUSBF0) 是由 USBF 产生的，则 FW 必须读取该寄存器以识别中断源。

当将 0 写入 UF0IC1 寄存器的对应位时，该寄存器的各位被强制清除为 0。但是，接收到下一个 SETUP 令牌时，UF0IS1 寄存器的 SUCES 和 STG 位会自动清除为 0。

注意事项 在 USBF 中，诸如总线复位、重传和短路等多重中断源在内部进行或运算并生成一个中断请求 (INTUSBF0)。因此，在发生多重中断源的情况下，它们被或运算并产生 INTUSBF0 中断请求。

例如，如果发生总线复位中断源和重传中断源，则两个中断源或运算并产生一个 INTUSBF0 中断请求信号。

在这些条件下，如果将总线复位中断源清除为 0 (UF0IC0.BUSRSTC = 0)，则由于仍存有重传中断源，所以 V850ES/JG3-H 或 V850ES/JH3-H 的内部 INTUSBF0 中断请求可能仍然保持设置为 1。因此，可能不会将新的中断请求标志 (US0BIC.US0BIF) 设为 1。

此时，使用 INTUSBF0 中断服务程序对各中断请求执行清除处理之后，应再次确认 UF0IS0 和 UF0IS1 寄存器的标志状态，且如有任何的标志设为 1 的中断源，执行标志清除（仅需对所用位进行清除（不要整体清除））。

(1/2)

								地址	复位后
								00200022H	00H
UF0IS1	7	6	5	4	3	2	1	0	
	0	E0IN	E0INDT	E0ODT	SUCES	STG	PROT	CPU DEC	

位号	位名称	功能
6	E0IN	该位表示接收了端点 0 的 IN 令牌，且硬件自动发送了 NAK。 1: 接收了 IN 令牌并发送 NAK（产生了中断请求）。 0: 未接收 IN 令牌（默认值）。
5	E0INDT	该位表示数据已由 UF0E0W 寄存器正确发送。 1: 完成了 UF0E0W 寄存器的发送（产生了中断请求）。 0: 未完成 UF0E0W 寄存器的发送（默认值）。 数据与将 UF0E0N 寄存器的 EP0NKW 位设为 1 的下一个 IN 令牌同步发送。当主机正确接收数据时，该位通过硬件自动设为 1。即使数据是空包也会将其设为 1。对 UF0E0W 寄存器进行第一次写访问时，会通过硬件自动将该位清除为 0。

位号	位名称	功能
4	E0ODT	该位表示 UF0E0R 寄存器已正确接收数据。 1: 数据位于 UF0E0R 寄存器 (产生了中断请求)。 0: UF0E0R 寄存器中没有数据 (默认值)。 当正确接收数据时, 该位通过硬件自动置 1。同时, UF0EPS0 寄存器的 EP0R 位也置为 1。如果接收到一个空包, 则不将该位置 1。当 FW 读取 UF0E0R 寄存器且 UF0E0L 寄存器的值变为 0 时, 该位通过硬件自动清除为 0。
3	SUCES	该位表示已接收了 FW-处理或硬件处理请求且已正确完成了状态阶段。 1: 已正确完成控制传输 (产生了中断请求)。 0: 没有正确完成控制传输 (默认值)。 状态阶段完成时将该位设为 1。接收到下一个 SETUP 令牌时, 该位通过硬件自动清除为 0。 在控制传输的状态阶段接收到带有 Data PID 为 0 的数据时 (空数据) 时, 也会将该位设为 1。
2	STG	控制传输阶段变为状态阶段时将该位设为 1。它对 FW-处理和硬件处理请求均有效。控制传输阶段 (无数据) 变为状态阶段时将该位也置为 1。 1: 状态阶段 (产生了中断请求) 0: 非状态阶段 (默认值) 接收到下一个 SETUP 令牌时, 该位通过硬件自动清除为 0。 在数据阶段不能正确接收 ACK 期间, 控制传输阶段变为状态阶段时也会将它设为 1。此时, 如果 FW 正在处理控制传输 (读取), 一旦 UF0E0W 寄存器被清除, UF0E0N 寄存器的 EP0NKW 位也会清除为 0。
1	PROT	该位表示正确接收了 SETUP 令牌。它对 FW-处理和硬件处理请求均有效。 1: 正确接收 SETUP 令牌 (产生中断请求)。 0: 未接收 SETUP 令牌 (默认值)。 当 UF0E0ST 寄存器正确接收数据时, 将该位设为 1。当对 UF0E0ST 寄存器做出第一次读取访问时, 通过 FW 将该位清除为 0。如果不通过 FW 将其清除为 0, 则不能正确识别下一个 SETUP 令牌的接收。 该位用于在控制传输期间正确识别 SETUP 处理的再次执行。如果控制传输期间重新执行 SETUP 处理且由硬件执行第二个请求, 则不将 CPUDEC 位设为 1, 但是 PROT 位可用于重新执行时的识别。
0	CPUDEC	该位表示 UF0E0ST 寄存器具有一个需通过 FW 解码的请求。 1: 数据位于 UF0E0ST 寄存器 (产生了中断请求)。 0: UF0E0ST 寄存器中没有数据 (默认值)。 当读取 UF0E0ST 寄存器的所有数据之后, 通过硬件将该位自动清除为 0。

(13) UF0 INT 状态 2 寄存器 (UF0IS2)

该寄存器表示中断源。 如果改变该寄存器的内容，则 INT1B 信号会变为有效。
本寄存器为只读寄存器，可以按 8 位读取。
如果一个中断请求 (INTUSBF0) 是由 USBF 产生的，则 FW 必须读取该寄存器以识别中断源。
当将 0 写入 UF0IC2 寄存器的对应位时，该寄存器的各位被强制清除为 0。
如果 UF0EnIM 寄存器 (n = 1, 3, 7) 的设置和接口的当前设置不支持各个端点，则相关位无效。

								7	6	5	4	3	2	1	0	地址	复位后
UF0IS2								BKl2IN	BKl2DT	BKl1IN	BKl1DT	0	0	IT2DT	IT1DT	00200024H	00H

位号	位名称	功能
7, 5	BKInIN	这些位表示 UF0Bin 寄存器中已接收到一个 IN 令牌（端点 m）且返回 NAK。 1: 接收了 IN 令牌并发送 NAK（产生中断请求）。 0: 未接收 IN 令牌（默认值）。
6, 4	BKInDT	这些位表示 UF0Bin 寄存器的 FIFO 区（端点 m）已切换。 这意味着可将数据写入端点 m。 1: FIFO 已切换（产生中断请求）。 0: FIFO 未切换（默认值）。 写入到端点 m 的数据与将 UF0EN 寄存器的 BKInNK 位设为 1 的下一个 IN 令牌同步发送。当 FIFO 已切换且可从 CPU 写入数据时，通过硬件将该位自动置 1。 当 FIFO 已切换时，即使数据是空包也会将其设为 1。 对 UF0Bin 寄存器进行第一次写访问时，会通过硬件自动将该位清除为 0。
1, 0	ITnDT	这些位表示已从 UF0INTn 寄存器正确接收数据（端点 x）。 1: 完成发送（产生中断请求）。 0: 未完成发送（默认值）。 数据与将 UF0EN 寄存器的 ITnNK 位设为 1 的下一个 IN 令牌同步发送。当主机正确接收数据时，该位通过硬件自动设为 1。 对 UF0INTn 寄存器进行第一次写访问时，会通过硬件自动将该位清除为 0。 即使数据是空包也会将该位设为 1。

备注

n = 1, 2

n = 1 处 m = 1 且 x = 7

n = 2 处 m = 3

(14) UF0 INT 状态 3 寄存器 (UF0IS3)

该寄存器表示中断源。 如果改变该寄存器的内容，则 INT1B 信号会变为有效。
本寄存器为只读寄存器，可以按 8 位读取。
如果一个中断请求 (INTUSBF0) 是由 USBF 产生的，则 FW 必须读取该寄存器以识别中断源。
当将 0 写入 UF0IC3 寄存器的对应位时，该寄存器的各位被强制清除为 0。
如果 UF0EnIM 寄存器 (n = 2, 4) 的设置和接口的当前设置不支持各个端点，则相关位无效。

(1/2)

UF0IS3

7	6	5	4	3	2	1	0	地址	复位后
BKO2FL	BKO2NL	BKO2 NAK	BKO2DT	BKO1FL	BKO1NL	BKO1 NAK	BKO1DT	00200024H	00H

位号	位名称	功能
7, 3	BKOnFL	这些位表示 UF0Bon 寄存器（端点 m）已正确接收数据，且 CPU 和 SIE 的 FIFO 区均保持数据。 1: 接收的数据位于 UF0Bon 寄存器的两个 FIFO（产生中断请求）。 0: 接收的数据不在 UF0Bon 寄存器的 SIE 侧的 FIFO（默认值）。 如果 CPU 和 SIE 的 FIFO 均数保持据，则通过硬件将这些位自动置 1。 当 FIFO 转换时，通过硬件将它们自动清除为 0。
6, 2	BKOnNL	这些位表示 UF0Bon 寄存器（端点 m）中接收到一个空包（长度为 0 的包）。 1: 接收空包（产生中断请求）。 0: 未接收空包（默认值）。 当 FIFO 为空时，空包接收之后立即将这些位设为 1。 如果数据位于 CPU 侧的 FIFO，当完成对此 FIFO 的读取时将它们设为 1。
5, 1	BKOnNAK	这些位表示 UF0Bon 寄存器中已接收到一个 OUT 令牌（端点 m）且返回 NAK。 1: 接收了 OUT 令牌并发送 NAK（产生中断请求）。 0: 未接收 OUT 令牌（默认值）。

备注

n = 1, 2

n = 1 处 m = 2

n = 2 处 m = 4

位号	位名称	功能
4, 0	BKOnDT	这些位表示已 UF0BOn 寄存器中正确接收了数据（端点 m）。 1: 接收已正确完成（产生了中断请求）。 0: 接收未完成（默认值）。 当正确接收数据且 FIFO 已转换时，通过硬件将这些位自动置 1。同时，将 UF0EPS0 寄存器相应的位也设为 1。当数据为空包时不将它们设为 1。当由于通过 FW 读取 UF0Bon 寄存器而致使 UF0BOnL 寄存器的值变为 0 时，通过硬件将这些位自动清除为 0。 当读取了 CPU 侧的 FIFO 区的所有内容时，这些位自动清除为 0。但是，如果此时数据位于 SIE 侧的 FIFO 区则不清除中断请求且 INTUSBF1 信号不会变为无效。如果连续接收数据则该信号保持有效。

备注 n = 1, 2
 n = 1 处 m = 2
 n = 2 处 m = 4

(15) UF0 INT 状态 4 寄存器 (UF0IS41)

该寄存器表示中断源。 如果改变该寄存器的内容，则 INT2B 信号会变为有效。
本寄存器为只读寄存器，可以按 8 位读取。
如果一个中断请求 (INTUSBF0) 是由 USBF 产生的，则 FW 必须读取该寄存器以识别中断源。
当将 0 写入 UF0IC4 寄存器的对应位时，该寄存器的各位被强制清除为 0。
如果 UF0EnIM 寄存器 (n = 1 至 4, 7) 的设置和接口的当前设置不支持各个端点，则相关位无效。

UF0IS4	7	6	5	4	3	2	1	0	地址	复位后
	0	0	SETINT	0	0	0	0	0	00200028H	00H

(16) UF0 INT 屏蔽 0 寄存器 (UF0IM0)

该寄存器控制对 UF0IS0 寄存器指示的中断源的屏蔽。

该寄存器可按 8 位为单位进行读写。

通过将 1 写入该寄存器的相应位，FW 可屏蔽发生来自 USBF 的中断请求 (INTUSBF0)。

UF0IM0	7	6	5	4	3	2	1	0	地址	复位后
	BUS RSTM	RSU SPDM	0	SHORTM	DMA EDM	SET RQM	CLR RQM	EP HALTM	0020002EH	00H

位号	位名称	功能
7	BUSRSTM	该位屏蔽总线复位中断。 1: 屏蔽 0: 不屏蔽 (默认值)。
6	RSUSPDM	该位屏蔽重传/中止中断。 1: 屏蔽 0: 不屏蔽 (默认值)。
4	SHORTM	该位屏蔽短路中断。 1: 屏蔽 0: 不屏蔽 (默认值)。
3	DMAEDM	该位屏蔽 DMA_END 中断。 1: 屏蔽 0: 不屏蔽 (默认值)。
2	SETRQM	该位屏蔽 SET_RQ 中断。 1: 屏蔽 0: 不屏蔽 (默认值)。
1	CLRRQM	该位屏蔽 CLR_RQ 中断。 1: 屏蔽 0: 不屏蔽 (默认值)。
0	EPHALTM	该位屏蔽 EP_Halt 中断。 1: 屏蔽 0: 不屏蔽 (默认值)。

(17) UF0 INT 屏蔽 1 寄存器 (UF0IM1)

该寄存器控制对 UF0IS1 寄存器指示的中断源的屏蔽。
该寄存器可按 8 位为单位进行读写。
通过将 1 写入该寄存器的相应位，FW 可屏蔽发生来自 USBF 的中断请求 (INTUSBF0)。

UF0IM1	7	6	5	4	3	2	1	0	地址	复位后
	0	E0INM	E0 INDTM	E0 ODTM	SUCESM	STGM	PROTM	CPU DECM	00200030H	00H

位号	位名称	功能
6	E0INM	该位屏蔽 EP0IN 中断。 1: 屏蔽 0: 不屏蔽 (默认值)。
5	E0INDTM	该位屏蔽 EP0INDT 中断。 1: 屏蔽 0: 不屏蔽 (默认值)。
4	E0ODTM	该位屏蔽 EP0OUTDT 中断。 1: 屏蔽 0: 不屏蔽 (默认值)。
3	SUCESM	该位屏蔽成功中断。 1: 屏蔽 0: 不屏蔽 (默认值)。
2	STGM	该位屏蔽 Stg 中断。 1: 屏蔽 0: 不屏蔽 (默认值)。
1	PROTM	该位屏蔽保护中断。 1: 屏蔽 0: 不屏蔽 (默认值)。
0	CPUDECM	该位屏蔽 CPUDEC 中断。 1: 屏蔽 0: 不屏蔽 (默认值)。

(19) UF0 INT 屏蔽 3 寄存器 (UF0IM3)

该寄存器控制对 UF0IS3 寄存器指示的中断源的屏蔽。
该寄存器可按 8 位为单位进行读写。
通过将 1 写入该寄存器的相应位，FW 可屏蔽发生来自 USBF 的中断请求 (INTUSBF0)。
如果 UF0EnIM 寄存器 (n = 2, 4) 的设置和接口的当前设置不支持各个端点，则相关位无效。

UF0IM3

7	6	5	4	3	2	1	0	地址	复位后
BKO2 FLM	BKO2 NLM	BKO2 NAKM	BKO2 DTM	BKO1 FLM	BKO1 NLM	BKO1 NAKM	BKO1 DTM	00200034H	00H

位号	位名称	功能
7, 3	BKOnFLM	这些位屏蔽 BLKOnFL 中断。 1: 屏蔽 0: 不屏蔽（默认值）。
6, 2	BKOnNLM	这些位屏蔽 BLKOnN 中断。 1: 屏蔽 0: 不屏蔽（默认值）。
5, 1	BKOnNAKM	这些位屏蔽 BLKOnNK 中断。 1: 屏蔽 0: 不屏蔽（默认值）。
4, 0	BKOnDTM	这些位屏蔽 BLKOnDT 中断。 1: 屏蔽 0: 不屏蔽（默认值）。

备注

n = 1, 2

(21) UF0 INT 清除 0 寄存器 (UF0IC0)

该寄存器控制对 UF0ISO 寄存器所指示的中断源的清除。
该寄存器只能按字节进行写入操作。 如果读取该寄存器，则读取值 FFH。
FW 可通过将 0 写入到该寄存器的相应位来清除中断源。 即使是通过硬件自动清除为 0 的位，在其被硬件清除之前也可使用 FW 清除。 向该寄存器的位写入 0 可自动将该位设为 1。写入 1 无效。

UF0IC0	7	6	5	4	3	2	1	0	地址	复位后
	BUS RSTC	RSU SPDC	1	SHORTC	DMA EDC	SET RQC	CLR RQC	EP HALTC	0020003CH	FFH
位号		位名称		功能						
7		BUSRSTC		该位清除总线复位中断。 0: 清除						
6		RSUSPDC		该位清除重传/中止中断。 0: 清除						
4		SHORTC		该位清除短路中断。 0: 清除						
3		DMAEDC		该位清除 DMA_END 中断。 0: 清除						
2		SETRQC		该位清除 SET_RQ 中断。 0: 清除						
1		CLRRQC		该位清除 CLR_RQ 中断。 0: 清除						
0		EPHALTC		该位清除 EP_Halt 中断。 0: 清除						

(22) UF0 INT 清除 1 寄存器 (UF0IC1)

该寄存器控制对 UF0IS1 寄存器所指示的中断源的清除。
该寄存器只能按字节进行写入操作。 如果读取该寄存器，则读取值 FFH。
FW 可通过将 0 写入到该寄存器的相应位来清除中断源。 即使是通过硬件自动清除为 0 的位，在其被硬件清除之前也可使用 FW 清除。 向该寄存器的位写入 0 可自动将该位设为 1。写入 1 无效。

UF0IC1	7	6	5	4	3	2	1	0	地址	复位后
	1	E0INC	E0 INDTC	E0ODTC	SUCESC	STGC	PROTC	CPU DECC	0020003EH	FFH

位号	位名称	功能
6	E0INC	该位清除 EP0IN 中断。 0: 清除
5	E0INDTC	该位清除 EP0INDT 中断。 0: 清除
4	E0ODTC	该位清除 EP0OUTDT 中断。 0: 清除
3	SUCESC	该位清除成功中断。 0: 清除
2	STGC	该位清除 Stg 中断。 0: 清除
1	PROTC	该位清除保护中断。 0: 清除
0	CPUDECC	该位清除 CPUDEC 中断。 0: 清除

(23) UF0 INT 清除 2 寄存器 (UF0IC2)

该寄存器控制对 UF0IS2 寄存器所指示的中断源的清除。
该寄存器只能按字节进行写入操作。 如果读取该寄存器，则读取值 FFH。
FW 可通过将 0 写入到该寄存器的相应位来清除中断源。 即使是通过硬件自动清除为 0 的位，在其被硬件清除之前也可使用 FW 清除。 向该寄存器的位写入 0 可自动将该位设为 1。写入 1 无效。
如果 UF0EnIM 寄存器 (n = 1, 3, 7) 的设置和接口的当前设置不支持各个端点，则相关位无效。

UF0IC2

7	6	5	4	3	2	1	0	地址	复位后
BKI2INC	BKI2 DTC	BKI1INC	BKI1 DTC	1	1	IT2DTC	IT1DTC	00200040H	FFH

位号	位名称	功能
7, 5	BKInINC	这些位清除 BLKInIN 中断。 0: 清除
6, 4	BKInDTC	这些位清除 BLKInDT 中断。 0: 清除
1, 0	ITnDTC	这些位清除 INTnDT 中断。 0: 清除

备注

n = 1, 2

(24) UF0 INT 清除 3 寄存器 (UF0IC3)

该寄存器控制对 UF0IS3 寄存器所指示的中断源的清除。

该寄存器只能按字节进行写入操作。 如果读取该寄存器，则读取值 FFH。

FW 可通过将 0 写入到该寄存器的相应位来清除中断源。 即使是通过硬件自动清除为 0 的位，在其被硬件清除之前也可使用 FW 清除。 向该寄存器的位写入 0 可自动将该位设为 1。写入 1 无效。

如果 UF0EnIM 寄存器 (n = 2, 4) 的设置和接口的当前设置不支持各个端点，则相关位无效。

UF0IC3	7	6	5	4	3	2	1	0	地址	复位后
	BKO2	BKO2	BKO2	BKO2	BKO1	BKO1	BKO1	BKO1	00200042H	FFH
	FLC	NLC	NAKC	DTC	FLC	NLC	NAKC	DTC		
位号		位名称		功能						
7, 3		BKOnFLC		这些位清除 BLKOnFL 中断。 0: 清除						
6, 2		BKOnNLC		这些位清除 BLKOnNL 中断。 0: 清除						
5, 1		BKOnNAKC		这些位清除 BLKOnNK 中断。 0: 清除						
4, 0		BKOnDTC		这些位清除 BLKOnDT 中断。 0: 清除						
备注		n = 1, 2								

(25) UF0 INT 清除 4 寄存器 (UF0IC4)

该寄存器控制对 UF0IS4 寄存器所指示的中断源的清除。
该寄存器只能按字节进行写入操作。 如果读取该寄存器，则读取值 FFH。
FW 可通过将 0 写入到该寄存器的相应位来清除中断源。 即使是通过硬件自动清除为 0 的位，在其被硬件清除之前也可使用 FW 清除。 向该寄存器的位写入 0 可自动将该位设为 1。写入 1 无效。
如果 UF0EnIM 寄存器 (n = 1 至 4, 7) 的设置和接口的当前设置不支持各个端点，则相关位无效。

UF0IC4	7	6	5	4	3	2	1	0	地址	复位后
	1	1	SETINTC	1	1	1	1	1	00200044H	FFH
位号		位名称	功能							
5		SETINTC	该位清除 SET_INT 中断。 0: 清除							

(26) UF0 INT & DMARQ 寄存器 (UF0IDR)

该寄存器通过中断请求或启动 DMA 进行选择报告。

该寄存器可按 8 位为单位进行读写。

如果在 UF0BO1 或 UF0BO1 寄存器中均有数据，或可将数据写入 UF0BI1 或 UF0BI2 寄存器，则该寄存器选择是通过中断请求将其报告给 FW，还是请求启动 DMA。如果请求启动 DMA，则可根据位 0 和位 1 的设置选择 DMA 传输模式。

如果 UF0EnIM 寄存器 (n = 1 至 4) 的设置和接口的当前设置不支持各个端点，则相关位无效。

一定要将位 3 和 2 清为 0。如果设置为 1，则操作将不能保障。

注意事项 DMA 传输情况下，如果 SET_INTERFACE 请求不支持目标端点，则 DMA 请求信号立即变为无效，且相应的位被硬件自动清除为 0。

(1/2)

UF0IDR	7	6	5	4	3	2	1	0	地址	复位后
	DQBI2 MS	DQBI1 MS	DQBO2 MS	DQBO1 MS	0	0	MODE1	MODE0	0020004CH	00H

位号	位名称	功能
7, 6	DQBI _n MS	这些位使能（屏蔽）一个写入 DMA 传送请求（端点 m 的 DMA 请求信号）至 UF0BI_n 寄存器。当这些位被设为 1 时，端点 m 的 DMA 请求信号变为有效，而写入数据也可得到响应。如果输入端点 m 的 DMA 结束信号（如果 DMA 控制器产生 TC），则由硬件将这些位自动清除为 0。要继续 DMA 传输，需通过 FW 重新将这些位置 1。 1: 使能端点 m 的有效 DMA 请求信号（屏蔽 BKInDT 中断）。 0: 禁止端点 m 的有效 DMA 请求信号（默认值）。
5, 4	DQBO _n MS	这些位使能（屏蔽）一个读取 DMA 传送请求（端点 x 的 DMA 请求信号）至 UF0BO_n 寄存器。当将这些位设为 1 时，如果将要读取的数据在 UF0BO_n 寄存器中就绪，则端点 x 的 DMA 请求信号变为有效。如果输入端点 x 的 DMA 结束信号（如果 DMA 控制器产生 TC），则由硬件将这些位自动清除为 0。当 USBSPxB 信号有效时也将它们清除为 0。要继续 DMA 传输，需通过 FW 重新将这些位置 1。 1: 使能端点 x 的有效 DMA 请求信号（屏蔽 BKOnDT 中断）。 0: 禁止端点 x 的有效 DMA 请求信号（默认值）。

备注	n = 1, 2 n = 1 处 m = 1 且 x = 2 n = 2 处 m = 3 且 x = 4

位号	位名称	功能			
1, 0	MODE1, MODE0	这些位选择 DMA 的传输模式。			
		MODE1	MODE0	模式	备注
		1	1	禁止设置	操作不能得到保障。
		1	0	按需模式	只要存在数据，DMA 请求信号就会变为有效。 如果不再有数据则它就变为无效。
		0	X	单次模式	每次 DMA 传输执行之后，DMA 请求信号变为无效。
		备注 1. X: 忽略 2. 当使用 USB 进行 DMA 传输时，传输仅在单次传输模式下可行。 因此，不管将 MODE1 和 MODE0 位设为按需模式还是设为单次模式，操作没有区别。			

(27) UF0 DMA 状态 0 寄存器 (UF0DMS0)

该寄存器表示端点 1 至端点 4 的 DMA 状态。
本寄存器为只读寄存器，可以按 8 位读取。
如果 UF0EnIM 寄存器 (n = 1 至 4) 的设置和接口的当前设置不支持各个端点，则相关位无效。

								7	6	5	4	3	2	1	0	地址	复位后
UF0DMS0								0	0	DQE4	DQE3	DQE2	DQE1	0	0	0020004EH	00H
位号	位名称	功能															
5	DQE4	该位表示正在产生从端点 4 至存储器的 DMA 读取请求。 1: 正在产生来自端点 4 的 DMA 读取请求。 0: 没有产生来自端点 4 的 DMA 读取请求 (默认值)。															
4	DQE3	该位表示正在产生从存储器至端点 3 的 DMA 写入请求。 注意，即使数据位于端点 3 (当 FIFO 未满足且已将 BKI2DED 位设为 1 之后)，当将 UF0IDR 寄存器的 DQBI2MS 位设为 1 时，DMA 请求信号也会立即变为有效且开始 DMA 传输。 1: 正在产生端点 3 的 DMA 写入请求。 0: 没有产生端点 3 的 DMA 写入请求 (默认值)。															
3	DQE2	该位表示正在产生从端点 2 至存储器的 DMA 读取请求。 1: 正在产生来自端点 2 的 DMA 读取请求。 0: 没有产生来自端点 2 的 DMA 读取请求 (默认值)。															
2	DQE1	该位表示正在产生从存储器至端点 1 的 DMA 写入请求。 注意，即使数据位于端点 1 (当 FIFO 未满足且已将 BKI1DED 位设为 1 之后)，当将 UF0IDR 寄存器的 DQBI1MS 位设为 1 时，DMA 请求信号也会立即变为有效且开始 DMA 传输。 1: 正在产生端点 1 的 DMA 写入请求。 0: 没有产生来自端点 1 的 DMA 写入请求 (默认值)。															

(28) UF0 DMA 状态 1 寄存器 (UF0DMS1)

该寄存器表示端点 1 至端点 4 的 DMA 状态。

本寄存器为只读寄存器，可以按 8 位读取。

如果 UF0EnIM 寄存器 (n = 1 至 4) 的设置和接口的当前设置不支持各个端点，则相关位无效。

当读取该寄存器时，将各位自动清零。但是，即使读取该寄存器时，UF0ISO 寄存器的位 4 和 3 也不会被清为 0。如果 SET_INTERFACE 请求不再支持目标端点，则各位被硬件自动清除为 0（但是，不会清除 DMA_END 请求和 Short 中断请求）。

								地址	复位后

(29) UF0 FIFO 清除 0 寄存器 (UF0FIC0)

该寄存器清除各个 FIFO。

该寄存器只能按字节进行写入操作。 如果读取该寄存器，则读取值为 00H。

FW 可通过将 1 写入到该寄存器的相应位来清空目标 FIFO。 写入 1 的位自动清除为 0。向该位写入 0 无效。

如果 UF0EnIM 寄存器 (n = 1, 3, 7) 的设置和接口的当前设置不支持各个端点，则相关位无效。

								7	6	5	4	3	2	1	0	地址	复位后
UF0FIC0								BKI2SC	BKI2CC	BKI1SC	BKI1CC	ITR2C	ITR1C	EP0WC	EP0RC	00200060H	00H
位号		位名称		功能													
7, 5		BKInSC		这些位仅清除 UF0BIn 寄存器 SIE 侧的 FIFO（复位计数器）。 1: 清除 将 BKInNK 位置 1 对端点 m 的 IN 令牌进行处理期间，对这些位进行写入无效。 通过清除 FIFO 可将 BKInNK 位自动清除为 0。 当使用这些位时应确保 CPU 侧的 FIFO 为空。													
6, 4		BKInCC		这些位仅清除 UF0BIn 寄存器 CPU 侧的 FIFO（复位计数器）。 1: 清除													
3, 2		ITRnC		这些位清除 UF0INTn 寄存器（复位计数器）。 1: 清除 将 ITnNK 位置 1 对端点 x 的 IN 令牌进行处理期间，对这些位进行写入无效。 通过清除 FIFO 可将 ITnNK 位自动清除为 0。													
1		EP0WC		该位清除 UF0E0W 寄存器（复位计数器）。 1: 清除 将 EP0NKW 位置 1 对端点 0 的 IN 令牌进行处理期间，对该位进行写入无效。 通过清除 FIFO 可将 EP0NKW 位自动清除为 0。													
0		EP0RC		该位清除 UF0E0R 寄存器（复位计数器）。 1: 清除 当将 EP0NKR 位设为 1（除该位已通过 FW 置位之外）时，通过清除 FIFO 将 EP0NKR 位自动清除为 0。													

备注

n = 1, 2

n = 1 处 m = 1 且 x = 7

n = 2 处 m = 3 且 x = 8

(30) UF0 FIFO 清除 1 寄存器 (UF0FIC1)

该寄存器清除各个 FIFO。

该寄存器只能按字节进行写入操作。 如果读取该寄存器，则读取值为 00H。

FW 可通过将 1 写入到该寄存器的相应位来目标 FIFO。 写入 1 的位自动清除为 0。向该位写入 0 无效。

如果 UF0EnIM 寄存器 (n = 2, 4) 的设置和接口的当前设置不支持各个端点，则相关位无效。

UF0FIC1	7	6	5	4	3	2	1	0	地址	复位后
	0	0	0	0	BKO2C	BKO2CC	BKO1C	BKO1CC	00200062H	00H

位号	位名称	功能
3, 1	BKOnC	这些位清除 UF0BOn 寄存器 SIE 和 CPU 两侧的 FIFO（复位计数器）。 1: 清除 当将 BKOnNK 位设为 1（除该位已通过 FW 置位之外）时，通过清除 FIFO 将 BKOnNK 位自动清除为 0。
2, 0	BKOnCC	这些位仅清除 UF0BOn 寄存器 CPU 侧的 FIFO（复位计数器）。 1: 清除 当将 BKOnNK 位设为 1（除该位已通过 FW 置位之外）时，通过清除 FIFO 将 BKOnNK 位自动清除为 0。

备注

n = 1, 2

(31) UF0 数据结束寄存器 (UF0DEND)

该寄存器报告至发送系统写入结束。

该寄存器为只写寄存器，可以按 8 位写入（但是位 7 和位 6 可以进行读和写）。如果读取该寄存器，则读取值为 00H。

FW 可通过写入 1 到该寄存器的相应位来启动目标端点的数据传送。写入 1 的位自动清除为 0。向该位写入 0 无效。

如果 UF0EnIM 寄存器（n = 1, 3, 7）的设置和接口的当前设置不支持各个端点，则相关位无效。

(1/2)

		7		6		5		4		3		2		1		0		地址		复位后	
UF0DEND		BKI2T		BKI1T		0		IT2DEND		IT1DEND		BKI2DED		BKI1DED		E0DED		0020006AH		00H	

位号	位名称	功能
7, 6	BKInT	如果 UF0BIn 寄存器 CPU 侧的 FIFO 由于 DMA 而变满，这些位指定是否自动执行 FIFO 的切换。 1: 一旦 FIFO 变满，就会自动执行 FIFO 的切换操作。 0: 即使 FIFO 变满，也不自动执行 FIFO 的切换操作。
4, 3	ITnDEND	将这些位设为 1 以发送 UF0INTn 寄存器的数据。 当这些位设为 1 时，ITnNK 位被设为 1 且进行数据传送。 1: 发送一个短包。 0: 不发送一个短包（默认值）。 如果将 UF0FIC0 寄存器的 ITRnC 位设为 1，之后将这些位也设为 1（UF0INTn 寄存器的计数器 = 0 且 UF0EPS0 寄存器的相应位 = 1），并发送一个空包（数据长度为 0）。 如果 UF0INTn 寄存器中存在数据且将这些位为 1（UF0INTn 寄存器的计数器 ≠ 0 且 UF0EPS0 寄存器的相应位 = 1），则发送一个短包。 当 FIFO 存满时这些位由硬件自动控制。

备注

n = 1, 2

位号	位名称	功能
2, 1	BKInDED	当完成将发送数据写入到 UF0BIn 寄存器时，将这些位设为 1。 当将这些位设为 1 时，FIFO 立刻切换，BKInNK 位设为 1 并发送数据。 1: 发送一个短包。 0: 不发送一个短包（默认值）。 这些位控制 CPU 侧的 FIFO。 如果将 UF0FIC0 寄存器的 BKInCC 位设为 1，之后将这些位也设为 1（UF0BIn 寄存器的计数器 = 0），并发送一个空包（数据长度为 0）。 如果 UF0BIn 寄存器中存在数据且将这些位设为 1（UF0BIn 寄存器的计数器 ≠ 0），且如果 FIFO 未滿，则发送一个短包。 如果 UF0BIn 寄存器 CPU 侧的 FIFO 由于 DMA 变滿，PIO 或 BKInT 为 1，即使不将这些位置 1，硬件也会开始数据发送。 如果 UF0BIn 寄存器 CPU 侧的 FIFO 由于 DMA 变滿，BKInT 位清除为 0，确保将这些位设为 1（参见 21.6.3 (3) UF0 EPNK 寄存器（UF0EN））。
0	E0DED	将该位设为 1 以发送 UF0E0W 寄存器的数据。 将该位设为 1 时，EP0NKW 位为 1 并发送数据。 1: 发送一个短包。 0: 不发送一个短包（默认值）。 如果将 UF0FIC0 寄存器的 EP0WC 位设为 1，之后该位也置为 1（UF0E0W 寄存器的计数器 = 0 且 UF0EPS0 寄存器的位 1 = 1），并发送一个空包（数据长度为 0）。 如果 UF0E0W 寄存器中存在数据且将该位置为 1（UF0E0W 寄存器的计数器 ≠ 0 且 UF0EPS0 寄存器的位 1 = 1），且如果 FIFO 未滿，则发送一个短包。

备注 n = 1, 2

(32) UF0 GPR 寄存器 (UF0GPR)

该寄存器控制 USBF 和 USB 接口。

该寄存器只能按字节进行写入操作。 如果读取该寄存器，则读取值为 00H。 务必将位 7 至位 1 清为“0”。

通过将 1 写入该寄存器的位 0，FW 可将 USBF 复位。 向该位写入 1 之后，该位自动清除为 0。 向该位写入 0 无效。

UF0GPR	7	6	5	4	3	2	1	0	地址	复位后
	0	0	0	0	0	0	0	MRST	0020006EH	00H

位号	位名称	功能
0	MRST	将该位设为 1 以复位 USBF。 1: 复位 实际上，由 FW 将该位置为 1 的两个 USB 之后，USBF 复位，且写入信号变为无效。 系统时钟运行期间通过 MRST 位复位 USBF，与通过 RESET 引脚复位（硬件复位）结果相同（寄存器值返回到默认值）。 但是，MRST 位，不复位 UF0CS 和 UF0BC 寄存器。

(33) UF0 模式控制寄存器 (UF0MODC)

该寄存器控制 CPUDEC 处理。

该寄存器可按 8 位为单位进行读写。

通过设置该寄存器的各位，可改变 UF0MODS 寄存器的设置。 仅在硬件复位和已将 UF0GRP 寄存器的 MRST 设为 1 时，该寄存器的各位才自动清除为 0。

即使通过硬件自动将该寄存器的位设为 1，也得通过 FW 设置优先。

务必将位 7 以及 5 至 0 清为“0”。 如果设置为 1，则操作将不能保障。

注意事项 该寄存器用于调试用途。 通常，除了验证操作或使用特殊模式时不要设置该寄存器。

UF0MODC	7	6	5	4	3	2	1	0	地址	复位后
	0	CDC GDST	0	0	0	0	0	0	00200074H	00H

位号	位名称	功能
6	CDCGDST	将该位设为 1，使 GET_DESCRIPTOR 配置请求转换为 CPUDEC 处理。 通过将该位置 1，可强制将 UF0MODS 寄存器的 CDCGD 设为 1。 1: 强制将 GET_DESCRIPTOR 配置请求变为 CPUDEC 处理（将 UF0MODS 寄存器的 CDCGD 位设为 1）。 0: 自动处理 GET_DESCRIPTOR 配置请求（默认值）。

(34) UF0 模式状态寄存器 (UF0MODS)

该寄存器表示配置状态。

本寄存器为只读寄存器，可以按 8 位读取。

	7	6	5	4	3	2	1	0	地址	复位后
UF0MODS	0	CDCGD	0	MPACK	DFLT	CONF	0	0	00200078H	00H

位号	位名称	功能
6	CDCGD	该位表示是否为 GET_DESCRIPTOR 配置请求进行 CPUDEC 处理。 1: 强制将 GET_DESCRIPTOR 配置请求变为 CPUDEC 处理。 0: 自动处理 GET_DESCRIPTOR 配置请求（默认值）。
4	MPACK	该位表示端点 0 发送包的大小。 1: 发送不是 8 个字节的包。 0: 发送 8 个字节的包（默认值）。 处理 GET_DESCRIPTOR 设备请求之后（状态阶段正常完成时），该位通过硬件自动置 1。一直到 USBF 复位之前，它不被清 0（不能被总线复位清 0）。 如果不将该位设为 1，则硬件仅按 8-字节为单位发送自动执行请求。因此，即使在完成 GET_DESCRIPTOR 设备请求之前由 OUT 令牌发送大于 8 个字节的由 FW 处理的数据，数据也可以被正确接收。 如果端点 0 的大小为 8 个字节，则忽略该位。
3	DFLT	该位表示默认状态（DFLT 位 = 1）。 1: 允许响应。 0: 禁止响应（通常无响应）（默认值）。 该位由总线复位自动置 1。直到该位置为 1 为止，不响应所有端点的处理。
2	CONF	该位表示是否已完成了 SET_CONFIGURATION 请求。 1: 已完成 SET_CONFIGURATION 请求。 0: 未完成 SET_CONFIGURATION 请求（默认值）。 当 SET_CONFIGURATION 请求接收到配置值 = 1 时，该位置为 1。 除非将该位设为 1，否则将忽略对端点 0 之外的端点的访问。 当 SET_CONFIGURATION 请求接收到配置值 = 0 时，该位清为 0。当检测到总线复位时也会清 0。

(35) UF0 动态接口编号寄存器 (UF0AIFN)

该寄存器设置与 GET/SET_INTERFACE 请求正确对应的有效接口编号。因为接口 0 总是有效，故可以选择接口 1 至 4。

该寄存器可按 8 位为单位进行读写。

								7	6	5	4	3	2	1	0	地址	复位后
UF0AIFN								ADDIF	0	0	0	0	0	IFNO1	IFNO0	00200080H	00H

位号	位名称	功能															
7	ADDIF	该位允许使用除 0 之外的接口编号。 1: 支持 IFNO1 和 IFNO0 位指定的接口编号。 0: 仅支持接口 0（默认值）。 不将该位设为 1 时，设置该寄存器的位 1 和 0 无效。															
1, 0	IFNO1, IFNO0	这些位指定所支持的接口编号范围。 <table><tr><th>IFNO1</th><th>IFNO0</th><th>有效接口编号</th></tr><tr><td>1</td><td>1</td><td>0, 1, 2, 3, 4</td></tr><tr><td>1</td><td>0</td><td>0, 1, 2, 3</td></tr><tr><td>0</td><td>1</td><td>0, 1, 2</td></tr><tr><td>0</td><td>0</td><td>0, 1</td></tr></table>	IFNO1	IFNO0	有效接口编号	1	1	0, 1, 2, 3, 4	1	0	0, 1, 2, 3	0	1	0, 1, 2	0	0	0, 1
IFNO1	IFNO0	有效接口编号															
1	1	0, 1, 2, 3, 4															
1	0	0, 1, 2, 3															
0	1	0, 1, 2															
0	0	0, 1															

(36) UF0 动态候选寄存器 (UF0AAS)

该寄存器指定接口编号和候选之间的连接。

该寄存器可按 8 位为单位进行读写。

V850ES/JG3-H 和 V850ES/JH3-H 单片机的 USBF 可为一个接口设置五级候选（可定义候选 0，1，2，3 和 4）和二级候选（可定义候选 0 和 1）。

		7		6		5		4		3		2		1		0		地址		复位后	
UF0AAS		ALT2		IFAL21		IFAL20		ALT2EN		ALT5		IFAL51		IFAL50		ALT5EN		00200082H		00H	

位号	位名称	功能															
7, 3	ALTn	这些位规定接口 0 是否连接 n 级候选。当将这些位设为 1 时，IFALn1 和 IFALn0 位的设置无效。 1: 将 n 级候选连接到接口 0。 0: 不将 n 级候选连接到接口 0（默认值）。															
6, 5, 2, 1	IFALn1, IFALn0	这些位规定要连接 n 级候选的接口的编号。如果连接的接口编号超出了 UF0AIFN 寄存器规定的范围，则 n 级候选无效（ALTnEN 位 = 0）。 <table><tr><th>IFALn1</th><th>IFALn0</th><th>要连接的接口编号</th></tr><tr><td>1</td><td>1</td><td>连接接口 4.</td></tr><tr><td>1</td><td>0</td><td>连接接口 3.</td></tr><tr><td>0</td><td>1</td><td>连接接口 2.</td></tr><tr><td>0</td><td>0</td><td>连接接口 1.</td></tr></table> 不要为同一个接口编号同时连接五级候选和二级候选。	IFALn1	IFALn0	要连接的接口编号	1	1	连接接口 4.	1	0	连接接口 3.	0	1	连接接口 2.	0	0	连接接口 1.
IFALn1	IFALn0	要连接的接口编号															
1	1	连接接口 4.															
1	0	连接接口 3.															
0	1	连接接口 2.															
0	0	连接接口 1.															
4, 0	ALTnEN	这些位使 n 级候选生效。除非将这些位设为 1，否则 ALTn, IFALn1 和 IFALn0 位的设置无效。 1: 使 n 级候选生效。 0: 不使 n 级候选生效（默认值）。															

备注 n = 2, 5

例如，当 UF0AIFN 寄存器设为 82H 且 UF0AAS 寄存器设为 15H 时，接口 0，1，2 和 3 有效。接口 0 和 2 仅支持候选 0。接口 1 支持候选 0 和 1，接口 3 支持候选 0，1，2，3 和 4。基于这些设置，请求 GET_INTERFACE wIndex = 0/1/2/3，SET_INTERFACE wValue = 0 & wIndex = 0/2，SET_INTERFACE wValue = 0/1 & wIndex = 1 和 SET_INTERFACE wValue = 0/1/2/3/4 & wIndex = 3 可自动得到响应，且会给其它 GET/SET_INTERFACE 请求做出一个 STALL 响应。

(37) UF0 候选状态寄存器 (UF0ASS)

该寄存器表示候选的当前状态。

本寄存器为只读寄存器，可以按 8 位读取。

当 SET_INT 中断请求产生时检查该寄存器。SET_INTERFACE 请求接收的值在 UF0IFn 寄存器 (n = 0 至 4) 呈现的同时也会在该寄存器呈现。

								7	6	5	4	3	2	1	0	地址	复位后
UF0ASS								0	0	0	0	AL5ST3	AL5ST2	AL5ST1	AL2ST	00200084H	00H

位号	位名称	功能																								
3 至 1	AL5ST3 至 AL5ST1	这些位表示五级候选的当前状态。																								
		<table><tr><th>AL5ST3</th><th>AL5ST2</th><th>AL5ST1</th><th>被选的候选编号</th></tr><tr><td>1</td><td>0</td><td>0</td><td>候选 4</td></tr><tr><td>0</td><td>1</td><td>1</td><td>候选 3</td></tr><tr><td>0</td><td>1</td><td>0</td><td>候选 2</td></tr><tr><td>0</td><td>0</td><td>1</td><td>候选 1</td></tr><tr><td>0</td><td>0</td><td>0</td><td>候选 0</td></tr></table>	AL5ST3	AL5ST2	AL5ST1	被选的候选编号	1	0	0	候选 4	0	1	1	候选 3	0	1	0	候选 2	0	0	1	候选 1	0	0	0	候选 0
		AL5ST3	AL5ST2	AL5ST1	被选的候选编号																					
		1	0	0	候选 4																					
		0	1	1	候选 3																					
		0	1	0	候选 2																					
		0	0	1	候选 1																					
0	0	0	候选 0																							
0	AL2ST	该位表示二级候选的当前状态（被选的候选编号）。 1: 候选 1 0: 候选 0																								

(38) UF0 端点 1 接口映射寄存器 (UF0E1IM)

该寄存器规定哪个接口和候选端点 1 有效。

该寄存器可按 8 位为单位进行读写。

该寄存器的设置和 SET_INTERFACE 请求所选的候选接口指示端点 1 当前有效与否，而硬件决定如何对 GET_STATUS/CLEAR_FEATURE/SET_FEATURE 端点 1 请求和端点 1 的 IN 处理做出响应，并决定相关位有效还是无效。

	7	6	5	4	3	2	1	0	地址	复位后
UF0E1IM	E1EN2	E1EN1	E1EN0	E12AL1	E15AL4	E15AL3	E15AL2	E15AL1	00200086H	00H

位号	位名称	功能																																			
7 至 5	E1EN2 至 E1EN0	这些位在端点 1 的接口与二/五级候选端点之间设置一个连接。端点与候选端点 0 相连。 与候选端点 0 相连的端点不能包括候选端点 1 至 4。																																			
		<table border="1"> <thead> <tr> <th>E1EN2</th> <th>E1EN1</th> <th>E1EN0</th> <th>连接状态</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1</td> <td>1</td> <td rowspan="2">未与接口相连</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>与接口 4 和候选端点 0 相连</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>与接口 3 和候选端点 0 相连</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>与接口 2 和候选端点 0 相连</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>与接口 1 和候选端点 0 相连</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>与接口 0 和候选端点 0 相连</td> </tr> <tr> <td>0</td> <td>0</td> <td>0</td> <td>不与接口相连（默认值）</td> </tr> </tbody> </table>	E1EN2	E1EN1	E1EN0	连接状态	1	1	1	未与接口相连	1	1	0	1	0	1	与接口 4 和候选端点 0 相连	1	0	0	与接口 3 和候选端点 0 相连	0	1	1	与接口 2 和候选端点 0 相连	0	1	0	与接口 1 和候选端点 0 相连	0	0	1	与接口 0 和候选端点 0 相连	0	0	0	不与接口相连（默认值）
		E1EN2	E1EN1	E1EN0	连接状态																																
		1	1	1	未与接口相连																																
		1	1	0																																	
		1	0	1	与接口 4 和候选端点 0 相连																																
		1	0	0	与接口 3 和候选端点 0 相连																																
		0	1	1	与接口 2 和候选端点 0 相连																																
		0	1	0	与接口 1 和候选端点 0 相连																																
		0	0	1	与接口 0 和候选端点 0 相连																																
0	0	0	不与接口相连（默认值）																																		
当这些位被设为 110 或 111 时，即使将 E12AL1 位清为 0 它们也无效。 如果连接了端点，UF0MODS 寄存器的 CONF 位设为 1 表示端点 1 有效。																																					
4	E12AL1	将二级候选和连接接口的候选设为 1 时，该位使得端点 1 生效。 1: 端点在通过 CONF 位 = 1 设定候选 1 时生效。 0: 即使在 CONF 位 = 1 设定候选 1 时（默认值），也不会使端点有效。 该位在 E15AL4 至 E15AL1 位为 0000 时有效。																																			
3 至 0	E15ALn	当将五级候选和连接接口的候选设为 n 时，这些位使得端点 1 生效。 1: 端点在通过 CONF 位 = 1 设定候选 n 时生效。 0: 即使 CONF 位 = 1 设定候选 n 时，也不会使端点生效（默认值）。																																			

备注 n = 1 至 4

(39) UF0 端点 2 接口映射寄存器 (UF0E2IM)

该寄存器规定端点 2 对哪个接口和候选接口有效。

该寄存器可按 8 位为单位进行读写。

该寄存器的设置和 SET_INTERFACE 请求所选的候选接口表示端点 2 当前有效与否，而硬件决定如何对 GET_STATUS/CLEAR_FEATURE/SET_FEATURE 端点 2 请求和端点 2 的 OUT 处理做出响应，并决定相关位有效还是无效。

		7	6	5	4	3	2	1	0	地址	复位后
UF0E2IM		E2EN2	E2EN1	E2EN0	E22AL1	E25AL4	E25AL3	E25AL2	E25AL1	00200088H	00H

位号	位名称	功能			
7 至 5	E2EN2 至 E2EN0	这些位在端点 2 的接口与二/五级候选之间设置一个连接。端点与候选端点 0 相连。与候选端点 0 相连的端点不能包括候选端点 1 至 4。			
		E2EN2	E2EN1	E2EN0	连接状态
		1	1	1	未与接口相连
		1	1	0	
		1	0	1	与接口 4 和候选端点 0 相连
		1	0	0	与接口 3 和候选端点 0 相连
		0	1	1	与接口 2 和候选端点 0 相连
		0	1	0	与接口 1 和候选端点 0 相连
		0	0	1	与接口 0 和候选端点 0 相连
		0	0	0	不与接口相连（默认值）
当这些位被设为 110 或 111 时，即使将 E22AL1 位清为 0 它们也会无效。 如果连接了端点，将 UF0MODS 寄存器的 CONF 位设为 1 表示端点 2 有效。					
4	E22AL1	将二级候选和连接接口的候选设为 1 时，该位使得端点 2 生效。 1: 端点在通过 CONF 位 = 1 设定候选端点 1 时生效。 0: 即使通过 CONF 位 = 1 设定候选端点 1，也不会使端点生效（默认值）。 该位在 E25AL4 至 E25AL1 位为 0000 时有效。			
3 至 0	E25ALn	当将五级候选和连接接口的候选设为 n 时，这些位使得端点 2 生效。 1: 端点在通过 CONF 位 = 1 设定候选 n 时生效。 0: 即使通过 CONF 位 = 1 设定候选 n，也不会使端点生效（默认值）。			

备注

n = 1 至 4

(40) UF0 端点 3 接口映射寄存器 (UF0E3IM)

该寄存器规定哪个接口和候选端点 3 有效。

该寄存器可按 8 位为单位进行读写。

该寄存器的设置和 SET_INTERFACE 请求所选的候选表示端点 3 当前有效与否，而硬件决定如何对 GET_STATUS/CLEAR_FEATURE/SET_FEATURE 端点 3 请求和端点 3 的 IN 处理做出响应，并决定相关位有效还是无效。

	7	6	5	4	3	2	1	0	地址	复位后
UF0E3IM	E3EN2	E3EN1	E3EN0	E32AL1	E35AL4	E35AL3	E35AL2	E35AL1	0020008AH	00H

位号	位名称	功能																																			
7 至 5	E3EN2 至 E3EN0	这些位在端点 3 的接口与二/五级候选之间设置一个连接。端点与候选端点 0 相连。与候选端点 0 相连的端点不能包括候选端点 1 至 4。																																			
		<table><tr><th>E3EN2</th><th>E3EN1</th><th>E3EN0</th><th>连接状态</th></tr><tr><td>1</td><td>1</td><td>1</td><td rowspan="2">未与接口相连</td></tr><tr><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td><td>与接口 4 和候选端点 0 相连</td></tr><tr><td>1</td><td>0</td><td>0</td><td>与接口 3 和候选 0 端点相连</td></tr><tr><td>0</td><td>1</td><td>1</td><td>与接口 2 和候选端点 0 相连</td></tr><tr><td>0</td><td>1</td><td>0</td><td>与接口 1 和候选端点 0 相连</td></tr><tr><td>0</td><td>0</td><td>1</td><td>与接口 0 和候选端点 0 相连</td></tr><tr><td>0</td><td>0</td><td>0</td><td>不与接口相连（默认值）</td></tr></table>	E3EN2	E3EN1	E3EN0	连接状态	1	1	1	未与接口相连	1	1	0	1	0	1	与接口 4 和候选端点 0 相连	1	0	0	与接口 3 和候选 0 端点相连	0	1	1	与接口 2 和候选端点 0 相连	0	1	0	与接口 1 和候选端点 0 相连	0	0	1	与接口 0 和候选端点 0 相连	0	0	0	不与接口相连（默认值）
		E3EN2	E3EN1	E3EN0	连接状态																																
		1	1	1	未与接口相连																																
		1	1	0																																	
		1	0	1	与接口 4 和候选端点 0 相连																																
		1	0	0	与接口 3 和候选 0 端点相连																																
		0	1	1	与接口 2 和候选端点 0 相连																																
		0	1	0	与接口 1 和候选端点 0 相连																																
		0	0	1	与接口 0 和候选端点 0 相连																																
0	0	0	不与接口相连（默认值）																																		
当这些位被设为 110 或 111 时，即使将 E32AL1 位清为 0 它们也会无效。 如果连接了端点，UF0MODS 寄存器的 CONF 位设为 1 表示端点 3 有效。																																					
4	E32AL1	将二级候选和连接接口的候选设为 1 时，该位使得端点 3 生效。 1: 端点在通过 CONF 位 = 1 设定候选端点 1 时生效。 0: 即使通过 CONF 位 = 1 设定候选端点 1，也不会使端点生效（默认值）。 该位在 E35AL4 至 E35AL1 位为 0000 时有效。																																			
3 to 0	E35ALn	当将五级候选和连接接口的候选设为 n 时，这些位使得端点 3 生效。 1: 端点在通过 CONF 位 = 1 设定候选 n 时生效。 0: 即使通过 CONF 位 = 1 设定候选 n，也不会使端点生效（默认值）。																																			

备注 n = 1 至 4

(41) UF0 端点 4 接口映射寄存器 (UF0E4IM)

该寄存器规定哪个接口和候选端点 4 有效。

该寄存器可按 8 位为单位进行读写。

该寄存器的设置和 SET_INTERFACE 请求所选的候选表示端点 4 当前有效与否，而硬件决定如何对 GET_STATUS/CLEAR_FEATURE/SET_FEATURE 端点 4 请求和端点 4 的 OUT 处理做出响应，并决定相关位有效还是无效。

7		6		5		4		3		2		1		0		地址		复位后	
UF0E4IM		E4EN2		E4EN1		E4EN0		E42AL1		E45AL4		E45AL3		E45AL2		E45AL1		0020008CH 00H	
位号		位名称		功能															
7 至 5		E4EN2 至 E4EN0		这些位在端点 4 的接口与二/五级候选之间设置一个连接。端点与候选端点 0 相连。与候选端点 0 相连的端点不能包括候选端点 1 至 4。															
				E4EN2		E4EN1		E4EN0		连接状态									
				1		1		1		未与接口相连									
				1		1		0											
				1		0		1		与接口 4 和候选端点 0 相连									
				1		0		0		与接口 3 和候选端点 0 相连									
				0		1		1		与接口 2 和候选端点 0 相连									
				0		1		0		与接口 1 和候选端点 0 相连									
				0		0		1		与接口 0 和候选端点 0 相连									
				0		0		0		不与接口相连（默认值）									
				当这些位被设为 110 或 111 时，即使将 E42AL1 位清为 0 它们也会无效。 如果连接了端点，UF0MODS 寄存器的 CONF 位设为 1 表示端点 4 有效。															
4		E42AL1		将二级候选和连接接口的候选设为 1 时，该位使得端点 4 生效。 1: 端点在通过 CONF 位 = 1 设定候选 1 时生效。 0: 即使通过 CONF 位 = 1 设定候选 1，也不会使端点生效（默认值）。 该位在 E45AL4 至 E45AL1 位为 0000 时有效。															
3 至 0		E45ALn		当将五级候选和连接接口的候选设为 n 时，这些位使得端点 4 生效。 1: 端点在通过 CONF 位 = 1 设定候选 n 时生效。 0: 即使通过 CONF 位 = 1 设定候选 n，也不会使端点生效（默认值）。															
备注 n = 1 至 4																			

(42) UF0 端点 7 接口映射寄存器 (UF0E7IM)

该寄存器规定哪个接口和候选端点 7 有效。

该寄存器可按 8 位为单位进行读写。

该寄存器的设置和 SET_INTERFACE 请求所选的候选表示端点 7 当前有效与否，而硬件决定如何对 GET_STATUS/CLEAR_FEATURE/SET_FEATURE 端点 7 请求和端点 7 的 IN 处理做出响应，并决定相关位有效还是无效。

	7	6	5	4	3	2	1	0	地址	复位后
UF0E7IM	E7EN2	E7EN1	E7EN0	E72AL1	E75AL4	E75AL3	E75AL2	E75AL1	00200092H	00H

位号	位名称	功能																																			
7 至 5	E7EN2 至 E7EN0	这些位在端点 7 的接口与二/五级候选之间设置一个连接。端点与候选 0 相连。与候选 0 相连的端点不能包括候选 1 至 4。 <table border="1"> <tr> <th>E7EN2</th> <th>E7EN1</th> <th>E7EN0</th> <th>连接状态</th> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td rowspan="2">未与接口相连</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>与接口 4 和候选 0 相连</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>与接口 3 和候选 0 相连</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>与接口 2 和候选 0 相连</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>与接口 1 和候选 0 相连</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>与接口 0 和候选 0 相连</td> </tr> <tr> <td>0</td> <td>0</td> <td>0</td> <td>不与接口相连（默认值）</td> </tr> </table>	E7EN2	E7EN1	E7EN0	连接状态	1	1	1	未与接口相连	1	1	0	1	0	1	与接口 4 和候选 0 相连	1	0	0	与接口 3 和候选 0 相连	0	1	1	与接口 2 和候选 0 相连	0	1	0	与接口 1 和候选 0 相连	0	0	1	与接口 0 和候选 0 相连	0	0	0	不与接口相连（默认值）
		E7EN2	E7EN1	E7EN0	连接状态																																
		1	1	1	未与接口相连																																
		1	1	0																																	
		1	0	1	与接口 4 和候选 0 相连																																
		1	0	0	与接口 3 和候选 0 相连																																
		0	1	1	与接口 2 和候选 0 相连																																
		0	1	0	与接口 1 和候选 0 相连																																
		0	0	1	与接口 0 和候选 0 相连																																
		0	0	0	不与接口相连（默认值）																																
当这些位被设为 110 或 111 时，即使将 E72AL1 位清为 0 它们也会无效。 如果连接了端点，将 UF0MODS 寄存器的 CONF 位设为 1 表示端点 7 有效。																																					
4	E72AL1	将二级候选和连接接口的候选设为 1 时，该位使得端点 7 生效。 1: 端点在通过 CONF 位 = 1 设定候选 1 时生效。 0: 即使通过 CONF 位 = 1 设定候选 1，也不会使端点生效（默认值）。 该位在 E75AL4 至 E75AL1 位为 0000 时有效。																																			
3 至 0	E75ALn	当将五级候选和连接接口的候选设为 n 时，这些位使得端点 7 生效。 1: 端点在通过 CONF 位 = 1 设定候选 n 时生效。 0: 即使通过 CONF 位 = 1 设定候选 n，也不会使端点生效（默认值）。																																			

备注 n = 1 至 4

21.6.4 数据保持寄存器

(1) UF0 EP0 读取寄存器 (UF0E0R)

UF0E0R 寄存器是一个 64-字节 FIFO，存储在端点 0 往复控制传输的数据阶段来自主机的 OUT 数据。本寄存器为只读寄存器，可以按 8 位读取。该寄存器的写入访问被忽略。

当从主机接收到数据时，硬件自动将数据传送给 UF0E0R 寄存器。当正确接收到数据时，将 UF0IS1 寄存器的 E0ODT 位设为 1。UF0E0L 寄存器保持接收的数据量，并产生一个中断请求 (INTUSBF0)。UF0E0L 总是在接收数据期间更新所接收数据的长度。如果最后的传送被正确接收，则产生中断请求。如果接收异常，则将 UF0E0L 寄存器清为 0 且不产生中断请求。

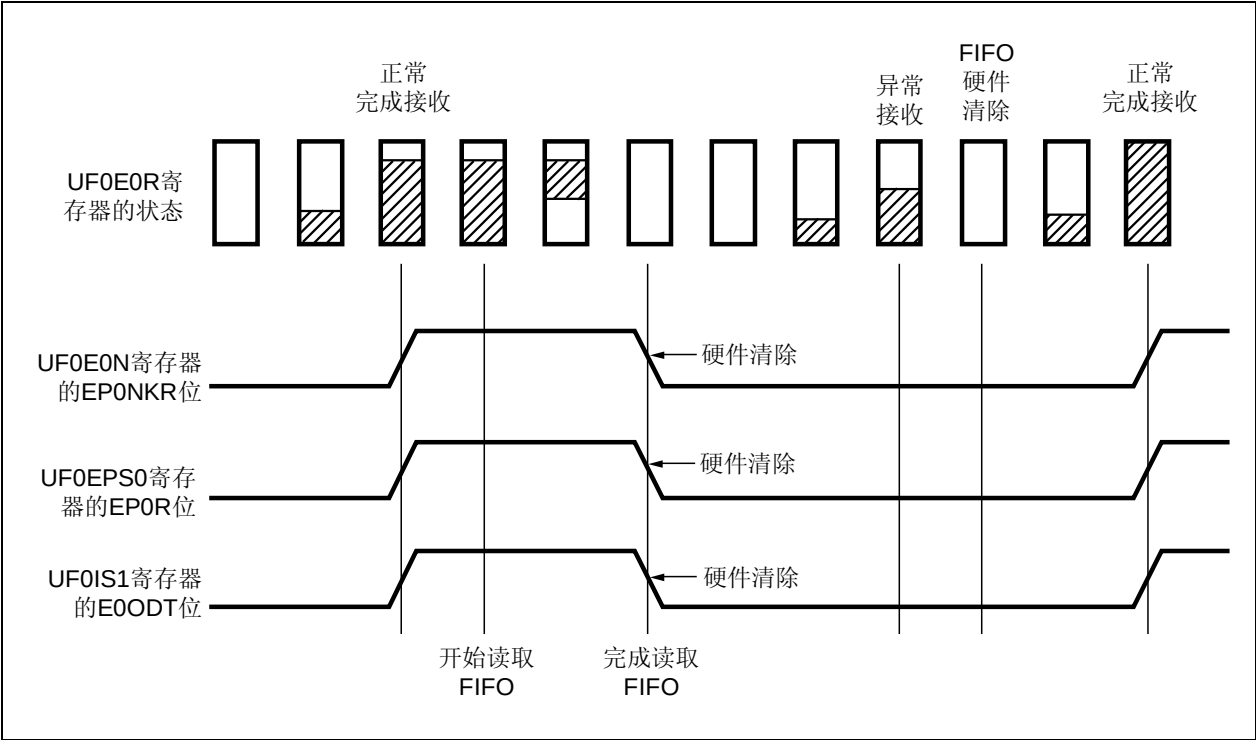
必须通过 FW 读取 UF0E0R 寄存器所保持的且由 UF0E0L 寄存器读取值指定数量的数据。检查所有的数据已通过 UF0EPS0 寄存器的 EP0R 位读取 (当所有的数据被读取时，EP0R 位 = 0)。如果 UF0E0L 寄存器的值为 0，则 UF0E0N 寄存器的 EP0NKR 位被清为 0，且 UF0E0R 寄存器准备好接收。当接收到下一个 SETUP 令牌时，清除 UF0E0R 寄存器。

注意事项 读取所有存储的数据。清除 FIFO 以删除数据。

								7	6	5	4	3	2	1	0	地址	复位后
UF0E0R								E0R7	E0R6	E0R5	E0R4	E0R3	E0R2	E0R1	E0R0	00200100H	不确定
位号		位名称		功能													
7 至 0		E0R7 至 E0R0		这些位存储端点 0 往复控制传输的数据阶段中来自主机的 OUT 数据。													

UF0E0R 寄存器的操作如下图所示。

图 21-4. UF0E0R 寄存器的操作



(2) UF0 EP0 长度寄存器 (UF0E0L)

UF0E0L 寄存器存储 UF0E0R 寄存器所保持的数据的长度。
本寄存器为只读寄存器，可以按 8 位读取。该寄存器的写入访问被忽略。
UF0E0L 总是在接收数据期间更新所接收数据的长度。如果最后的传输接收异常，则 UF0E0L 寄存器清 0 且不产生中断请求。中断请求仅在接收正常时产生，且 FW 可从 UF0E0R 寄存器读取与 UF0E0L 寄存器读取值等值的数据。UF0E0L 寄存器的值在每次读取 UF0E0R 寄存器时递减。

	7	6	5	4	3	2	1	0	地址	复位后
UF0E0L	E0L7	E0L6	E0L5	E0L4	E0L3	E0L2	E0L1	E0L0	00200102H	00H
位号	位名称	功能								
7 至 0	E0L7 至 E0L0	这些位存储 UF0E0R 寄存器所保持的数据的长度。								

(3) UF0 EP0 设置寄存器 (UF0E0ST)

UF0E0ST 寄存器保持主机发送的 SETUP 数据。
本寄存器为只读寄存器，可以按 8 位读取。该寄存器的写入访问被忽略。
UF0E0ST 寄存器通常在接收到 SETUP 处理时写入数据。当正确接收 SETUP 处理时，硬件设置 UF0IS1 寄存器的 PROT 位。在 FW-处理请求的情况下会设置 UF0IS1 寄存器的 CPUDEC 位。然后产生一个中断请求 (INTUSBF0)。FW-处理请求情况下，务必以 8-字节为单位读取该请求。如果不以 8-字节为单位读取，则接下来的请求将不能正确解码。即使接收总线复位时，也不会清除 UF0E0ST 寄存器的读取计数器。不管是否接收了总线复位，总是以 8-字节为单位读取该计数器。
因为 UF0E0ST 寄存器总是使能写入，所以即使在读取寄存器数据期间接收到 SETUP 处理，硬件也会将数据重写至该寄存器。即使不能正确接收 SETUP 处理，也不会产生 CPUDEC 中断请求和保护中断请求，但是之前的数据会被丢弃。但是，如果接收到少于 8 个字节的 SETUP 令牌，则接收到的 SETUP 令牌会被丢弃并保持原来接收的 SETUP 数据。如果在进行一次控制传输时接收到多于一次的 SETUP 令牌，那么务必在下面的条件下检查 UF0IS1 寄存器的 PROT 位。如果 PROT 位 = 1，则因为接收的 SETUP 多于一次，需再次读取 UF0E0ST 寄存器。

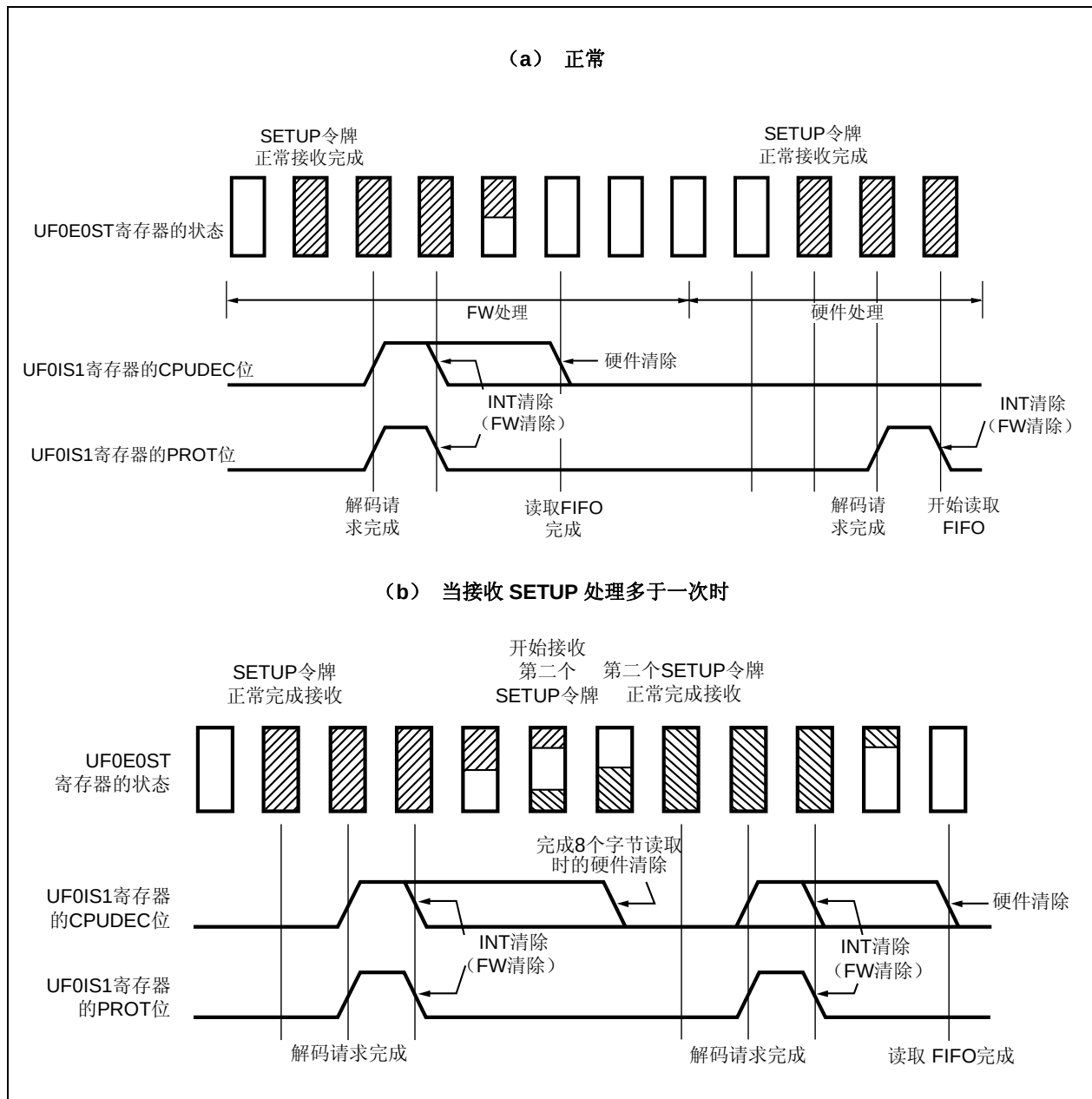
- <1> 如果通过 FW 对请求进行解码且读取 UF0E0R 寄存器或写入 UF0E0W 寄存器
- <2> 准备为解码不符的请求做出 STALL 响应时

注意事项 务必读取存储的所有数据。总是通过 SETUP 中的请求更新 UF0E0ST 寄存器。

UF0E0ST	7	6	5	4	3	2	1	0	地址	复位后
	E0S7	E0S6	E0S5	E0S4	E0S3	E0S2	E0S1	E0S0	00200104H	00H
位号		位名称		功能						
7 至 0		E0S7 至 E0S0		这些位保持从主机发送的 SETUP 数据。						

UF0E0ST 寄存器的操作如下图所示。

图 21-5. UF0E0ST 寄存器的操作



(4) UF0 EP0 写入寄存器 (UF0E0W)

UF0E0W 寄存器是一个 64-字节的 FIFO，它存储在至端点 0 数据阶段中发送给主机的 IN 数据（将它传送给 SIE）。

该寄存器只能按字节进行写入操作。读取该寄存器时，读取值为 00H。

仅当 UF0E0N 寄存器的 EP0NKW 设为 1（不发送 NAK 时）时，硬件才与一个 IN 令牌同步地将数据发送到 USB 总线。当发送数据且主机正确接收数据时，硬件自动将 UF0E0N 寄存器的 EP0NKW 位清为 0。当将数据写入 UF0E0W 寄存器且将 UF0DEND 寄存器的 E0DED 位设为 1 时，会发送一个短包（UF0EPS0 寄存器的 EP0W 位 = 1（数据存在））。当清除 UF0E0W 寄存器且将 UF0DEND 寄存器的 E0DED 位设为 1 时，会发送一个空包（UF0EPS0 寄存器的 EP0W 位 = 1（数据存在））。

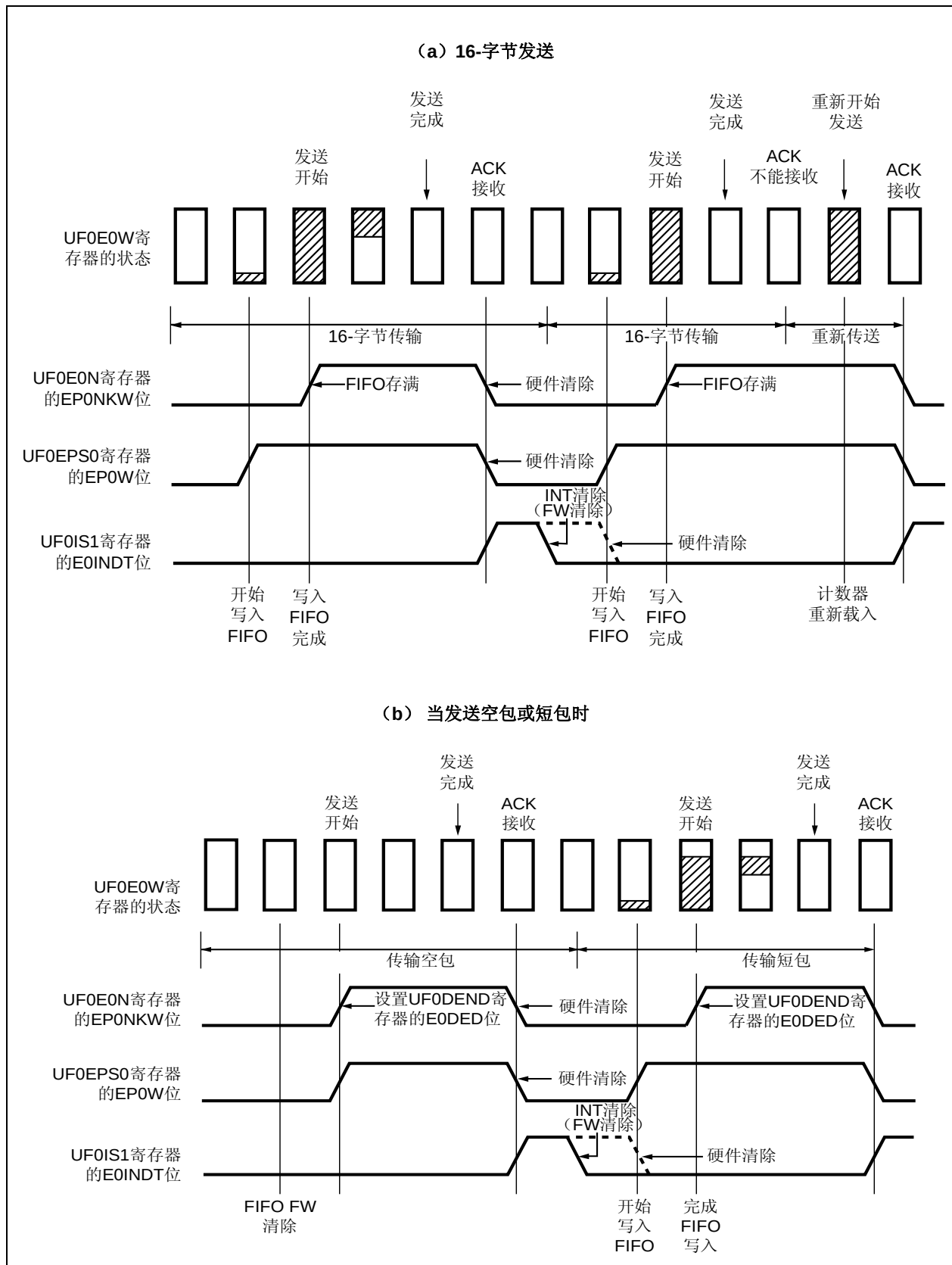
没有完成发送的期间，接收到下一个 SETUP 令牌时会将 UF0E0W 寄存器清除为 0。如果在数据阶段没有正确接收 ACK 期间将控制传输阶段（读取）变为状态阶段，则 UF0E0W 寄存器自动清除为 0。同时，如果 UF0E0N 寄存器的 EP0NKW 位为 1，也会将它清除为 0。

如果在 UF0E0W 寄存器没有数据期间对其进行读取，则会读取 00H。

UF0E0W	7	6	5	4	3	2	1	0	地址	复位后
	E0W7	E0W6	E0W5	E0W4	E0W3	E0W2	E0W1	E0W0	00200106H	不确定
位号		位名称		功能						
7 至 0		E0W7 至 E0W0		这些位存储至端点 0 的数据阶段发送给主机的 IN 数据。						

UF0E0W 寄存器的操作如下图所示。

图 21-6. UF0E0W 寄存器的操作



(5) UF0 bulk-out 1 寄存器 (UF0BO1)

UF0BO1 寄存器是一个为端点 2 存储数据的 64-字节 × 2 的 FIFO。该寄存器由两个 64-字节的 FIFO 存储区组成，各存储区进行切换操作并反复连接到 SIE 和 CPU 侧的总线。当数据位于 SIE 侧的 FIFO 且 CPU 侧的 FIFO 区不存在数据（计数器的值 = 0）时，发生切换操作。

本寄存器为只读寄存器，可以按 8 位读取。该寄存器的写入访问被忽略。

当硬件从主机为端点 2 接收数据时，它自动将数据传送至 UF0BO1 寄存器。当寄存器正确接收数据时，发生 FIFO 切换操作。结果，UF0IS3 寄存器的 BKO1DT 位被设为 1，接收到数据的数量由 UF0BO1L 寄存器保持，且给 CPU 发一个中断请求或 DMA 请求。可通过 UF0IDR 寄存器的 DQBO1MS 位来选择产生中断请求还是 DMA 请求。

通过 FW 读取 UF0BO1 寄存器与 UF0BO1L 寄存器读取值等值的数据。当正确接收的数据由连接到 SIE 侧的 FIFO 保持且 UF0BO1L 寄存器的值达到 0 时，发生 FIFO 的切换，且 UF0EN 寄存器的 BKO1NK 位自动清除为 0。如果读取的数据大于 UF0BO1L 寄存器的值且满足 FIFO 的切换条件，则会发生 FIFO 的切换操作。结果，可能会误读取下一个包。注意，如果不满足切换条件，则会重复读取第一个数据。

如果在数据由 CPU 侧连接的 FIFO 保持的期间接收数据溢出，则停止端点 2，并清除 CPU 侧的 FIFO。

如果在 UF0BO1 寄存器没有数据期间对其进行读取，则会读取不确定的值。

注意事项 务必读取该寄存器中存储的所有数据。

UF0BO1	7	6	5	4	3	2	1	0	地址	复位后
	BKO17	BKO16	BKO15	BKO14	BKO13	BKO12	BKO11	BKO10	00200108H	不确定
位号		位名称		功能						
7 至 0		BKO17 至 BKO10		这些位存储端点 2 的数据。						

UF0BO1 寄存器的操作如下图所示。

图 21-7. UF0BO1 寄存器的操作 (1/2)

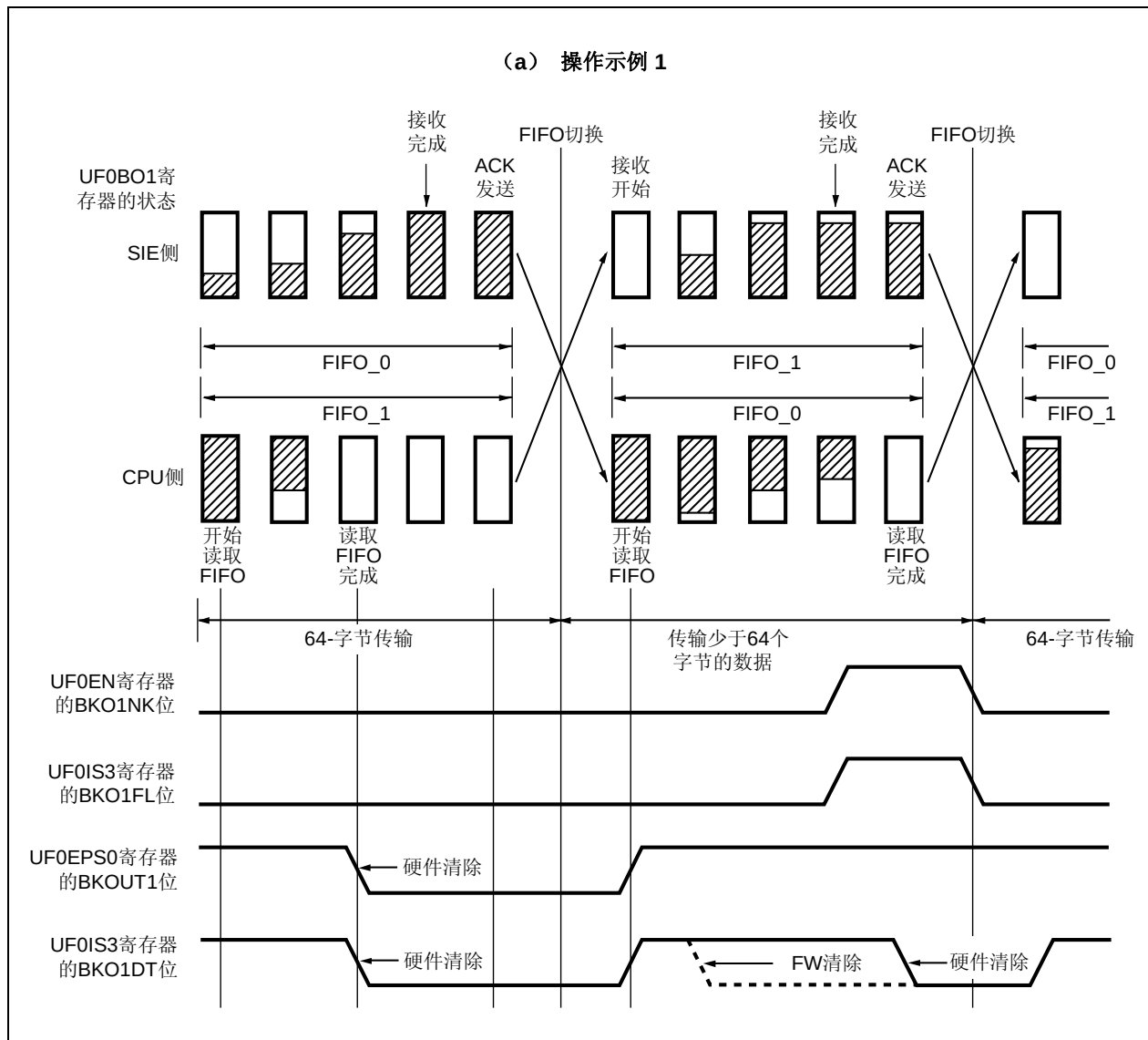
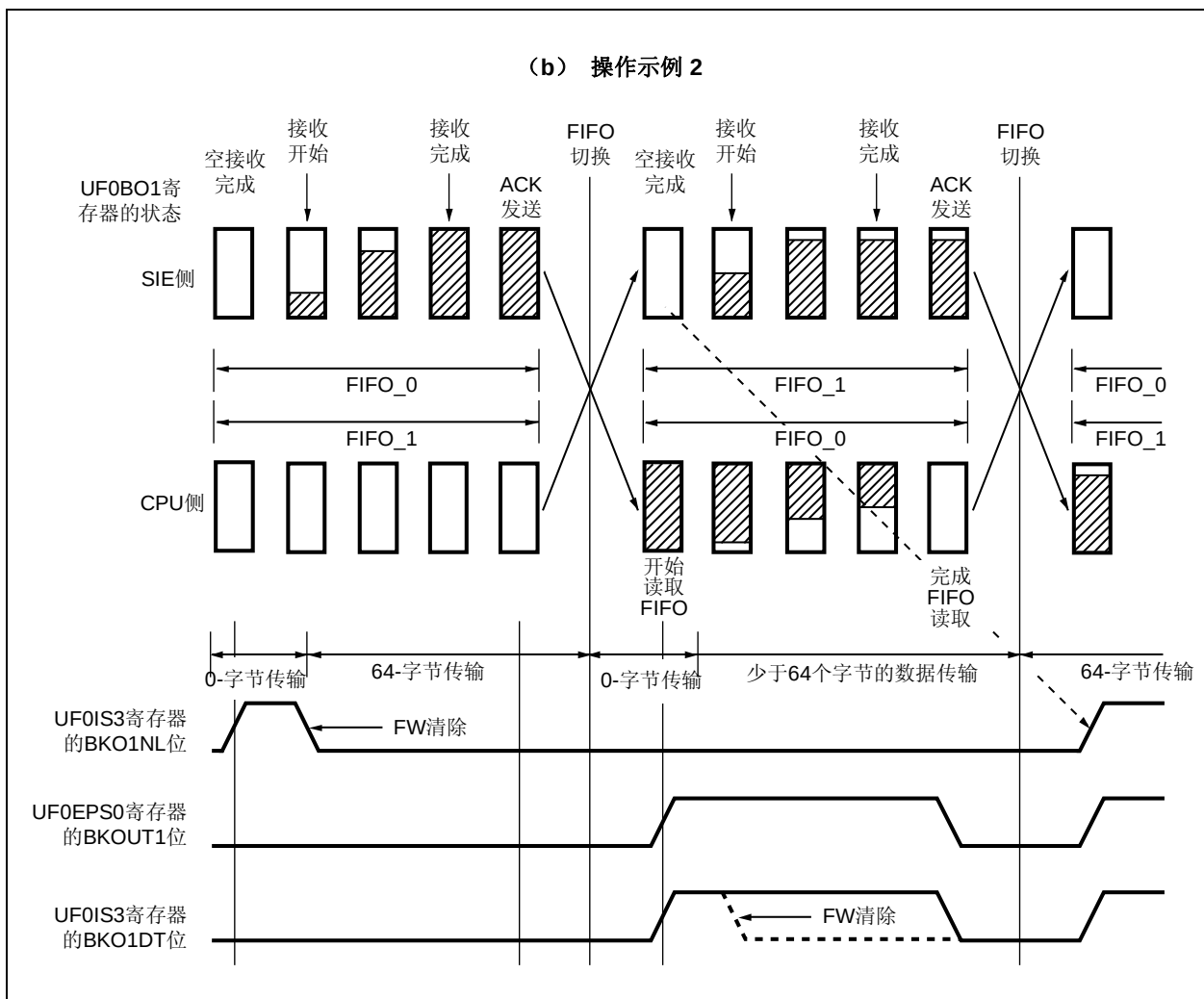


图 21-7. UF0BO1 寄存器的操作 (2/2)



(6) UF0 bulk-out 1 长度寄存器 (UF0BO1L)

UF0BO1L 寄存器存储 UF0BO1 寄存器所保持数据的长度。
本寄存器为只读寄存器，可以按 8 位读取。该寄存器的写入访问被忽略。
UF0BO1L 总是在接收数据期间更新所接收数据的长度。如果最后的传输接收异常，则将 UF0BO1L 寄存器清为 0 且不产生中断请求。中断请求仅在接收正常时产生，且 FW 可从 UF0BO1 寄存器读取的数据量与从 UF0BO1L 寄存器的读取值相等。UF0BO1L 寄存器的值在每次读取 UF0BO1 寄存器时递减。

	7	6	5	4	3	2	1	0	地址	复位后
UF0BO1L	BKO1L7	BKO1L6	BKO1L5	BKO1L4	BKO1L3	BKO1L2	BKO1L1	BKO1L0	0020010AH	00H
位号		位名称		功能						
7 至 0		BKO1L7 至 BKO1L0		这些位存储 UF0BO1 寄存器所保持数据的长度。						

(7) UF0 bulk-out 2 寄存器 (UF0BO2)

UF0BO2 寄存器是一个为端点 4 存储数据的 64-字节 × 2 的 FIFO。该寄存器由两个 64-字节的 FIFO 存储区组成，各存储区执行切换操作并反复连接到 SIE 和 CPU 侧的总线。当数据位于 SIE 侧的 FIFO 区且 CPU 侧的 FIFO 区不存在数据（计数器的值 = 0）时，发生切换操作。

本寄存器为只读寄存器，可以按 8 位读取。该寄存器的写入访问被忽略。

当硬件从主机为端点 4 接收数据时，它自动将数据传送至 UF0BO2 寄存器。当寄存器正确接收数据时，发生 FIFO 切换操作。结果，UF0IS3 寄存器的 BKO2DT 位被设为 1，接收到数据的数量由 UF0BO2L 寄存器保持，且给 CPU 发一个中断请求或 DMA 请求。可通过 UF0IDR 寄存器的 DQBO2MS 位来选择产生中断请求还是 DMA 请求。

在 UF0BO2L 寄存器读取数据数量的值之前，通过 FW 读取 UF0BO2 寄存器所保持的数据。当正确接收的数据由连接到 SIE 侧的 FIFO 保持且 UF0BO2L 寄存器的值达到 0 时，发生 FIFO 的切换操作，且 UF0EN 寄存器的 BKO2NK 位自动清除为 0。如果读取的数据大于 UF0BO2L 寄存器的值且满足 FIFO 的切换条件，则会发生 FIFO 的切换操作。结果，可能会误读取下一个包。注意，如果不满足切换条件，则会重复读取第一个数据。

如果在数据由 CPU 侧连接的 FIFO 保持的期间接收到溢出数据，则停止端点 4，并清除 CPU 侧的 FIFO。

如果在 UF0BO2 寄存器没有数据期间对其进行读取，则会读取不确定的值。

注意事项 务必读取该寄存器中存储的所有数据。

								地址	复位后	
7	6	5	4	3	2	1	0			
UF0BO2	BKO27	BKO26	BKO25	BKO24	BKO23	BKO22	BKO21	BKO20	0020010CH	不确定
位号		位名称		功能						
7 至 0		BKO27 至 BKO20		这些位存储端点 4 的数据。						

UF0BO2 寄存器的操作如下图所示。

图 21-8. UF0BO2 寄存器的操作 (1/2)

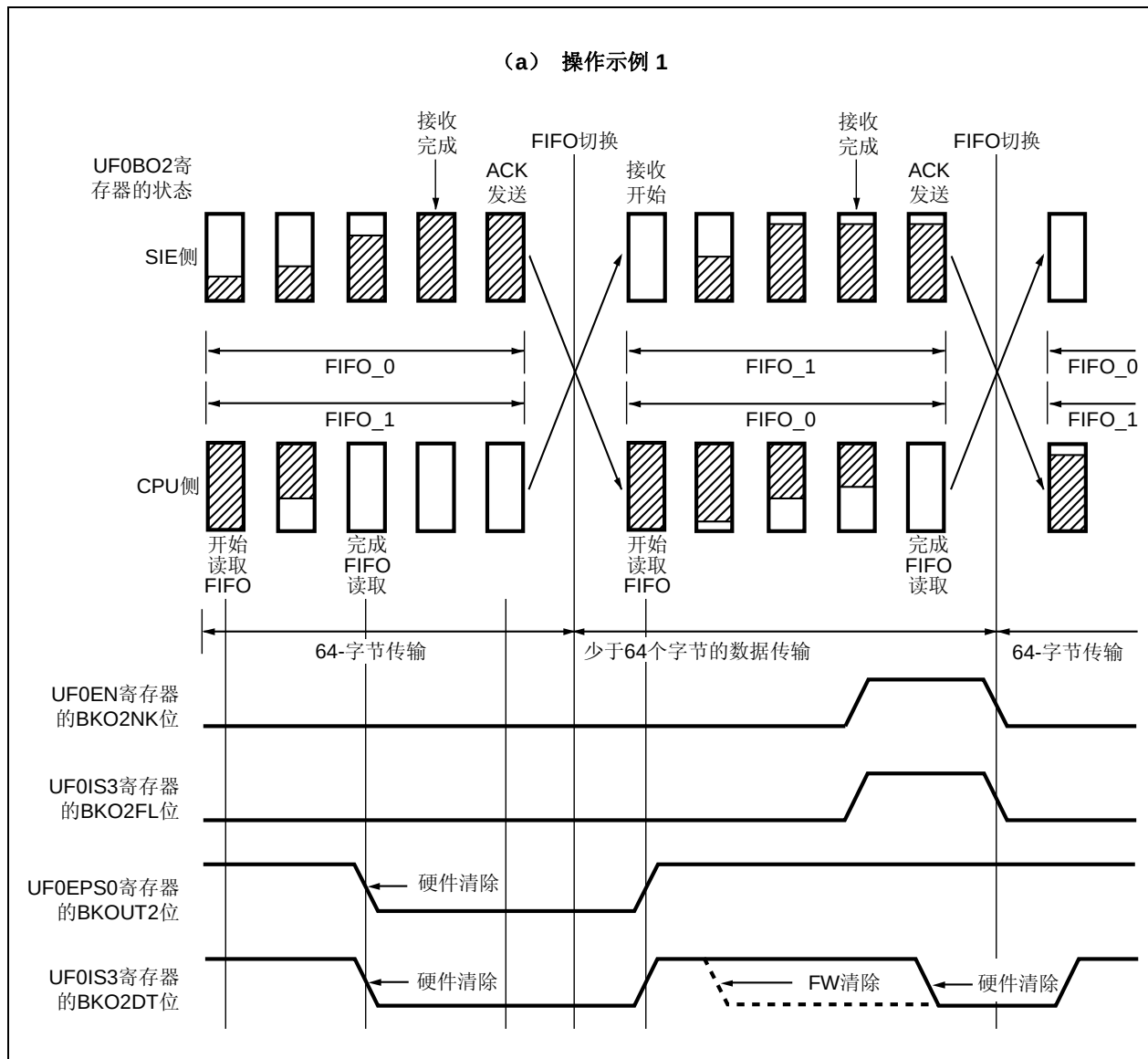
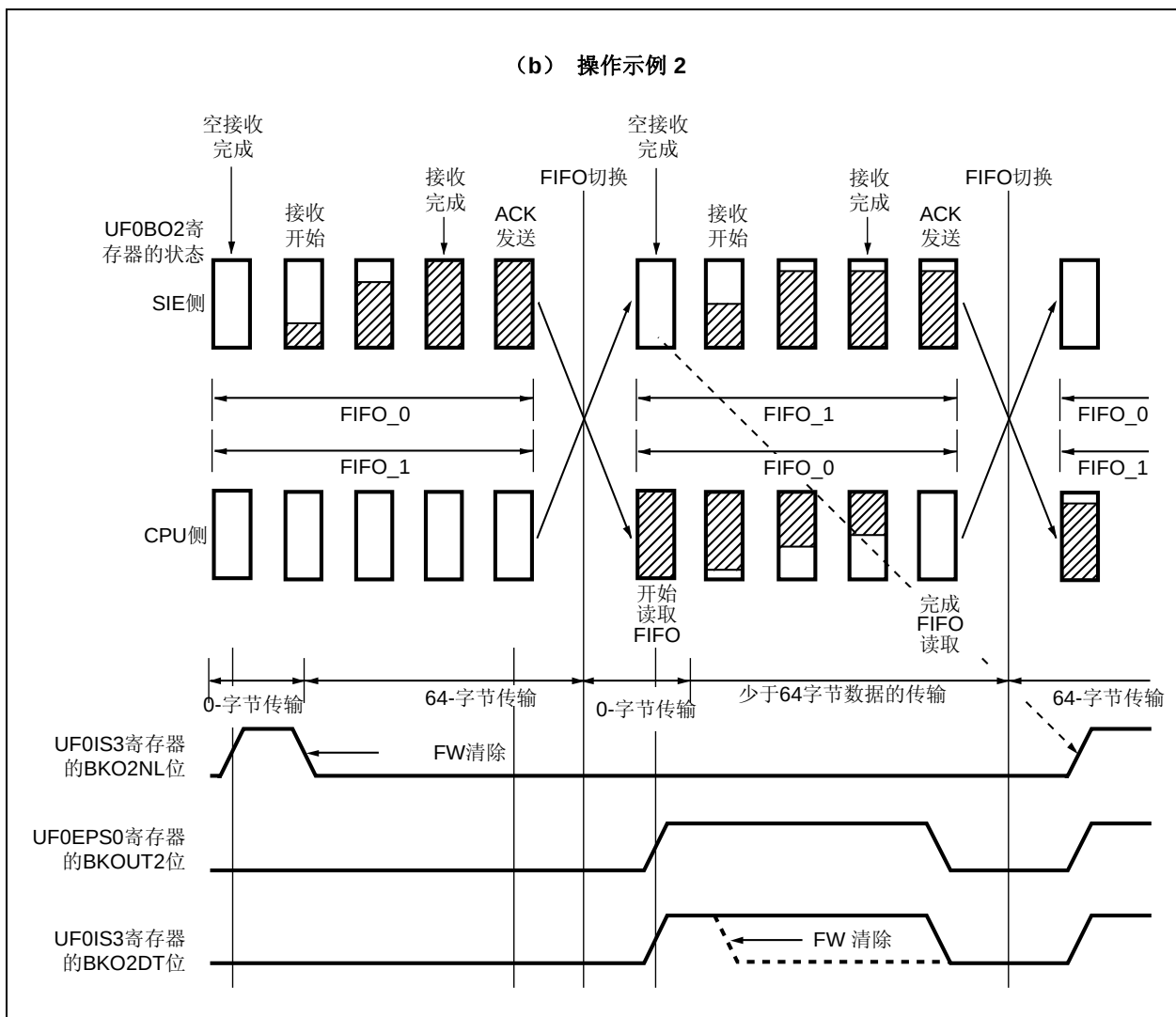


图 21-8. UF0BO2 寄存器的操作 (2/2)



(8) UF0 bulk-out 2 长度寄存器 (UF0BO2L)

UF0BO2L 寄存器存储 UF0BO2 寄存器所保持数据的长度。
本寄存器为只读寄存器，可以按 8 位读取。该寄存器的写入访问被忽略。
UF0BO2L 总是在接收数据期间更新所接收数据的长度。如果最后的传输接收异常，则将 UF0BO2L 寄存器清为 00H 且不产生中断请求。中断请求仅在接收正常时产生，且 FW 可从 UF0BO2 寄存器读取与 UF0BO2L 寄存器读取值等值的数据。UF0BO2L 寄存器的值在每次读取 UF0BO2 寄存器时递减。

								地址	复位后
UF0BO2L	7	6	5	4	3	2	1	0	
	BKO2L7	BKO2L6	BKO2L5	BKO2L4	BKO2L3	BKO2L2	BKO2L1	BKO2L0	0020010EH 00H
位号		位名称		功能					
7 至 0		BKO2L7 至 BKO2L0		这些位存储 UF0BO2 寄存器所保持数据的长度。					

(9) UF0 bulk-in 1 寄存器 (UF0BI1)

UF0BI1 寄存器是一个为端点 1 存储数据的 64-字节 × 2 的 FIFO。该寄存器由两个 64-字节的 FIFO 存储区组成，各存储区执行切换操作并反复连接到 SIE 和 CPU 侧的总线。当 SIE 侧的 FIFO 中没有数据（计数器的值 = 0）且正确写入 CPU 侧的 FIFO 时（FIFO 存满或 BKI1DED 位 = 1），会发生切换操作。
该寄存器只能按字节进行写入操作。读取该寄存器时，会读取到 00H。
仅当 UF0EN 寄存器的 BKI1NK 设为 1（不发送 NAK 时）时，硬件才与端点 1 的一个 IN 令牌同步地将数据发送到 USB 总线。将要写入或读取数据的地址由硬件管理。因此，FW 仅能通过将数据顺序地写入 UF0BI1 寄存器将数据发送到主机。当将数据写入 UF0BI1 寄存器且将 UF0DEND 寄存器的 BKI1DED 位设为 1 时，会发送一个短包（UF0EPS0 寄存器的 BKIN1 位 = 1（数据存在））。当清除 UF0BI1 寄存器且将 UF0DEND 寄存器的 BKI1DED 位设为 1 时，会发送一个空包（UF0EPS0 寄存器的 BKIN1 位 = 1（数据存在））。当数据被正确发送时，发生 FIFO 切换操作。UF0IS2 寄存器的 BKI1DT 位被设为 1，且为 CPU 产生一个中断请求。可通过 UF0IDR 寄存器的 DQBI1MS 位来选择中断请求或 DMA 请求。

								地址	复位后
UF0BI1	7	6	5	4	3	2	1	0	
	BKI17	BKI16	BKI15	BKI14	BKI13	BKI12	BKI11	BKI10	00200110H 不确定
位号		位名称		功能					
7 至 0		BKI17 至 BKI10		这些位存储端点 1 的数据。					

UF0BI1 寄存器的操作如下图所示。

图 21-9. UF0BI1 寄存器的操作 (1/3)

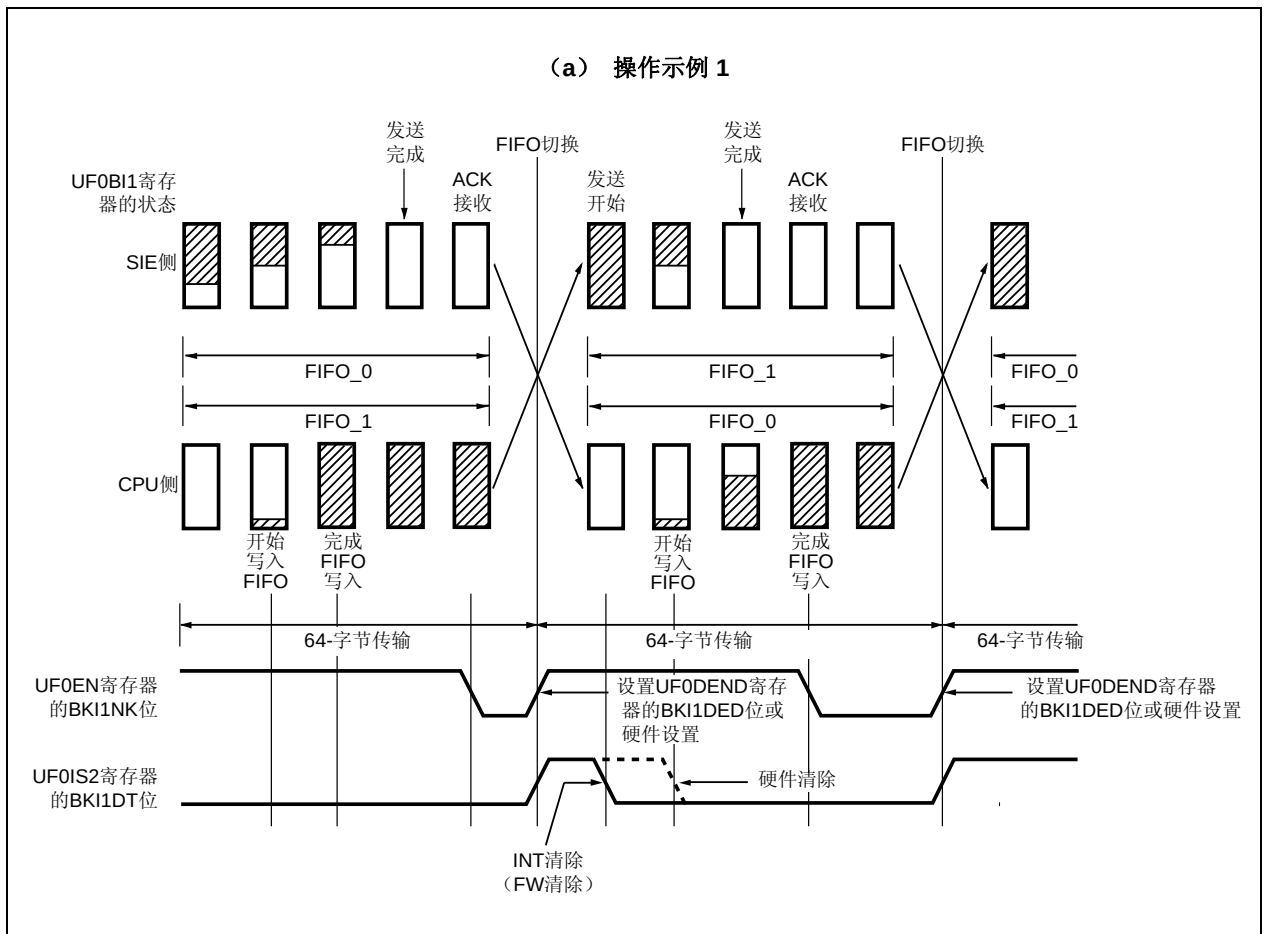


图 21-9. UF0BI1 寄存器的操作 (2/3)

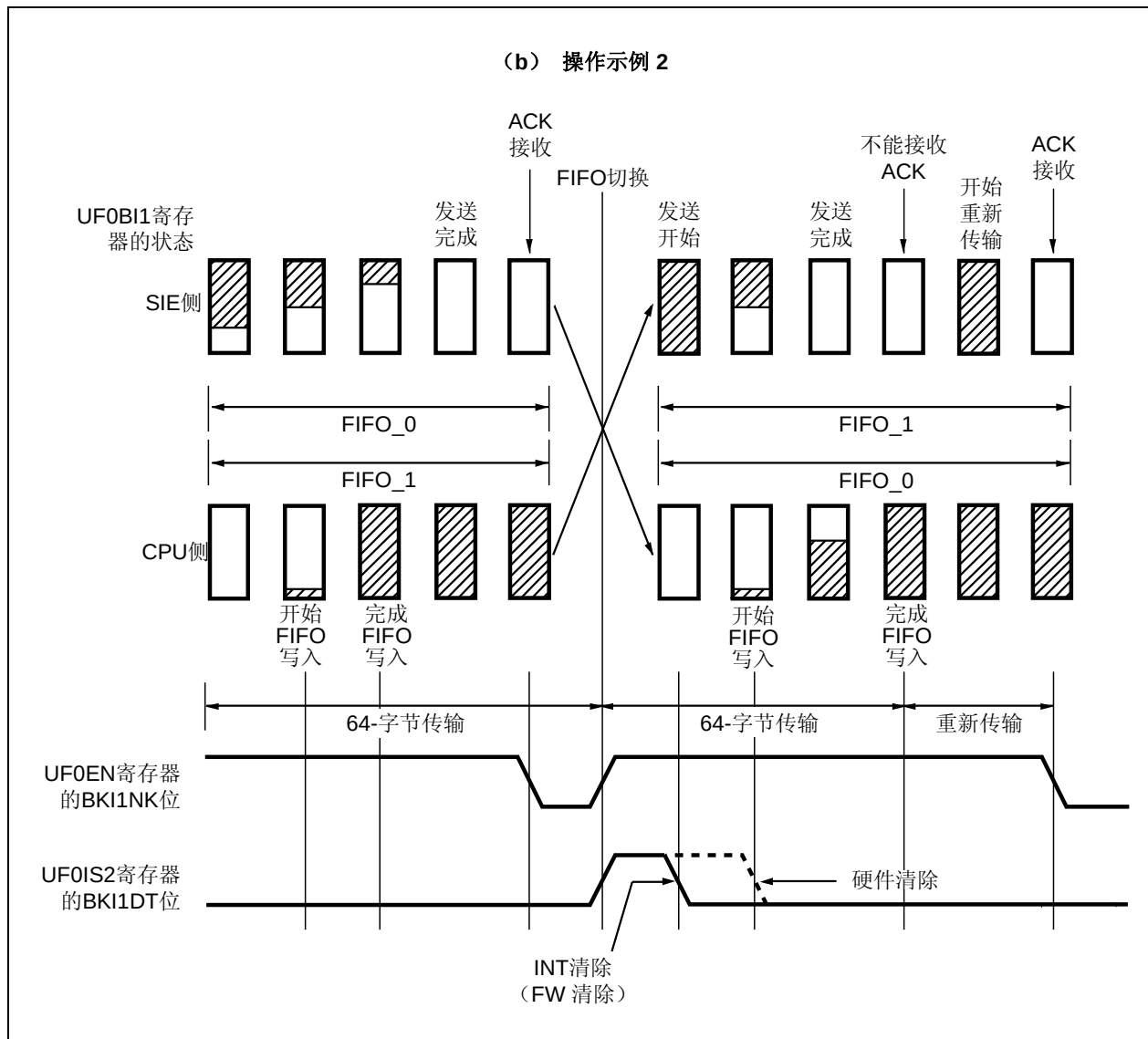
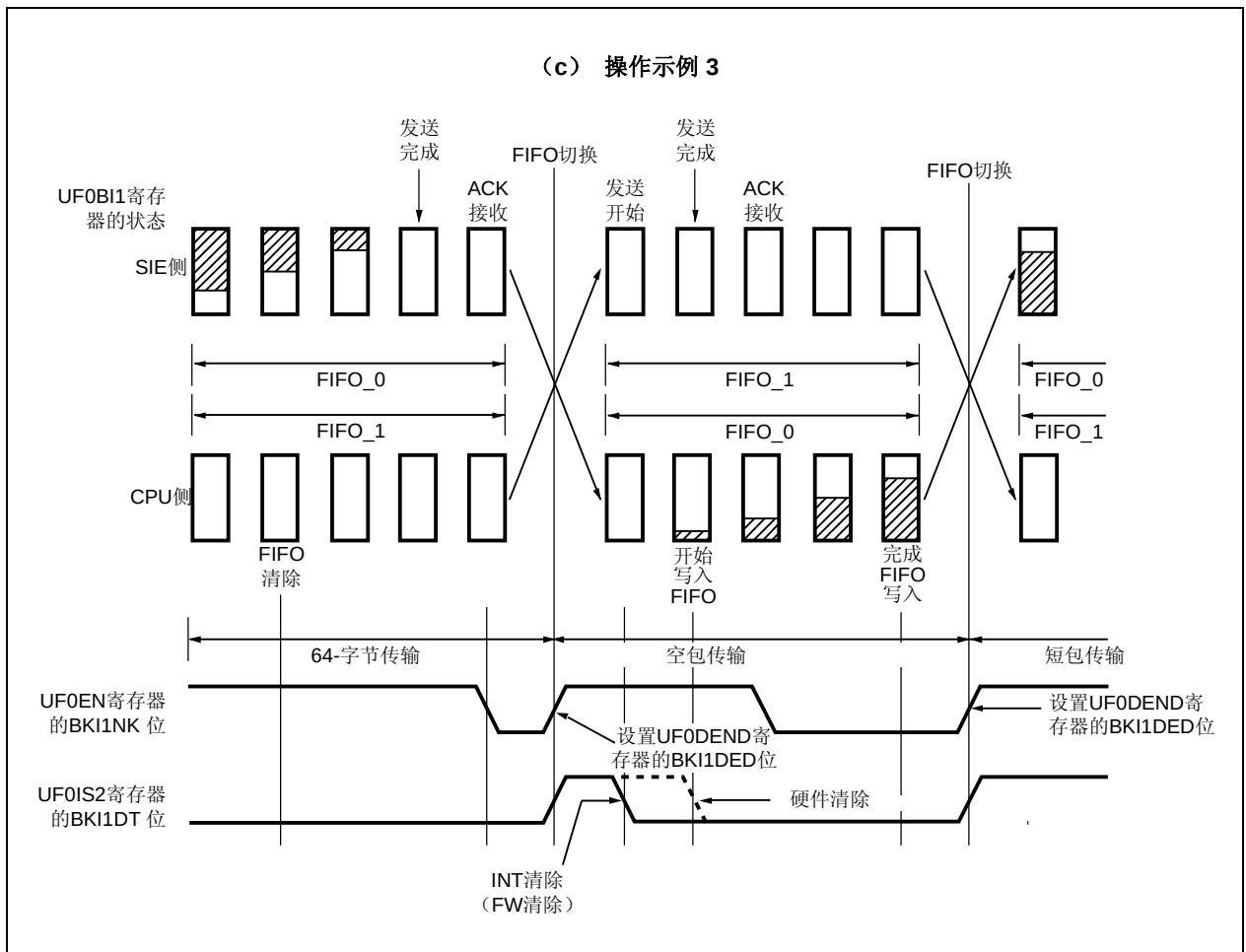


图 21-9. UF0BI1 寄存器的操作 (3/3)



(10) UF0 bulk-in 2 寄存器 (UF0BI2)

UF0BI2 寄存器是一个为端点 3 存储数据的 64-字节 × 2 的 FIFO。该寄存器由两个 64-字节的 FIFO 存储区组成，各存储区执行切换操作并反复连接到 SIE 和 CPU 侧的总线。当 SIE 侧的 FIFO 中没有数据（计数器的值 = 0）且正确写入 CPU 侧的 FIFO 时（FIFO 存满或 BKI2DED 位 = 1），会发生切换操作。

该寄存器只能按字节进行写入操作。读取该寄存器时，会读取到 00H。

仅当 UF0EN 寄存器的 BKI2NK 设为 1（不发送 NAK 时）时，硬件才与端点 3 的 IN 令牌同步地将数据发送到 USB 总线。将要写入或读取数据的地址由硬件管理。因此，FW 仅能通过将数据顺序地写入 UF0BI2 寄存器将数据发送到主机。当将数据写入 UF0BI2 寄存器且将 UF0DEND 寄存器的 BKI2DED 位设为 1（UF0EPS0 寄存器的 BKIN2 位 = 1（数据存在））时，会发送一个短包。当清除 UF0BI2 寄存器且将 UF0DEND 寄存器的 BKI2DED 位设为 1（UF0EPS0 寄存器的 BKIN2 位 = 1（数据存在））时，会发送一个空包。当数据被正确发送时，发生 FIFO 切换操作。UF0IS2 寄存器的 BKI2DT 位被设为 1，且向 CPU 发一个中断请求。可通过 UF0IDR 寄存器的 DQBI2M 位来选择中断请求或 DMA 请求。

	7	6	5	4	3	2	1	0	地址	复位后
UF0BI2	BKI27	BKI26	BKI25	BKI24	BKI23	BKI22	BKI21	BKI20	00200112H	不确定
位号		位名称		功能						
7 至 0		BKI27 至 BKI20		这些位存储端点 3 的数据。						

UF0BI2 寄存器的操作如下图所示。

图 21-10. UF0BI2 寄存器的操作 (1/3)

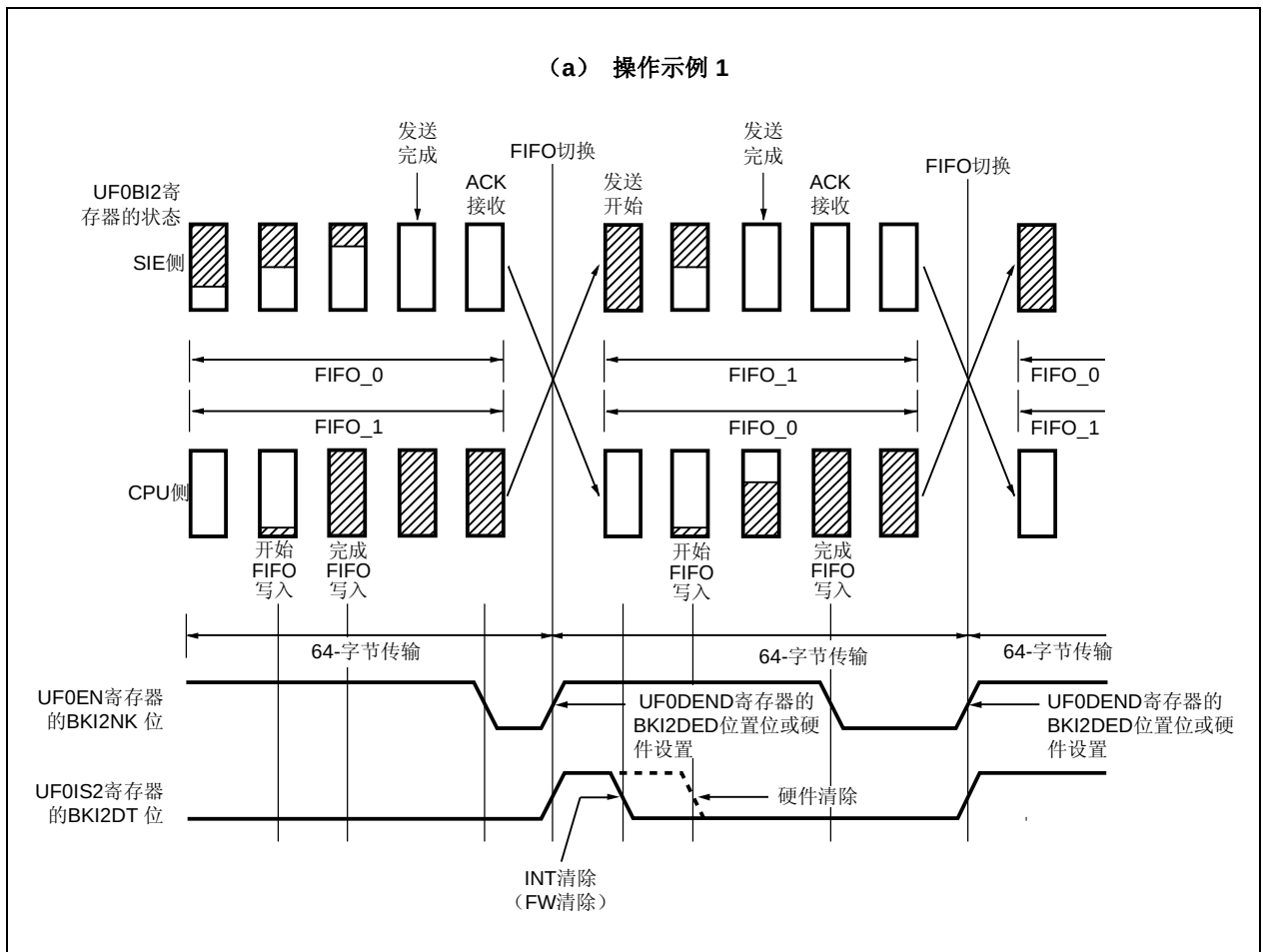


图 21-10. UF0BI2 寄存器的操作 (2/3)

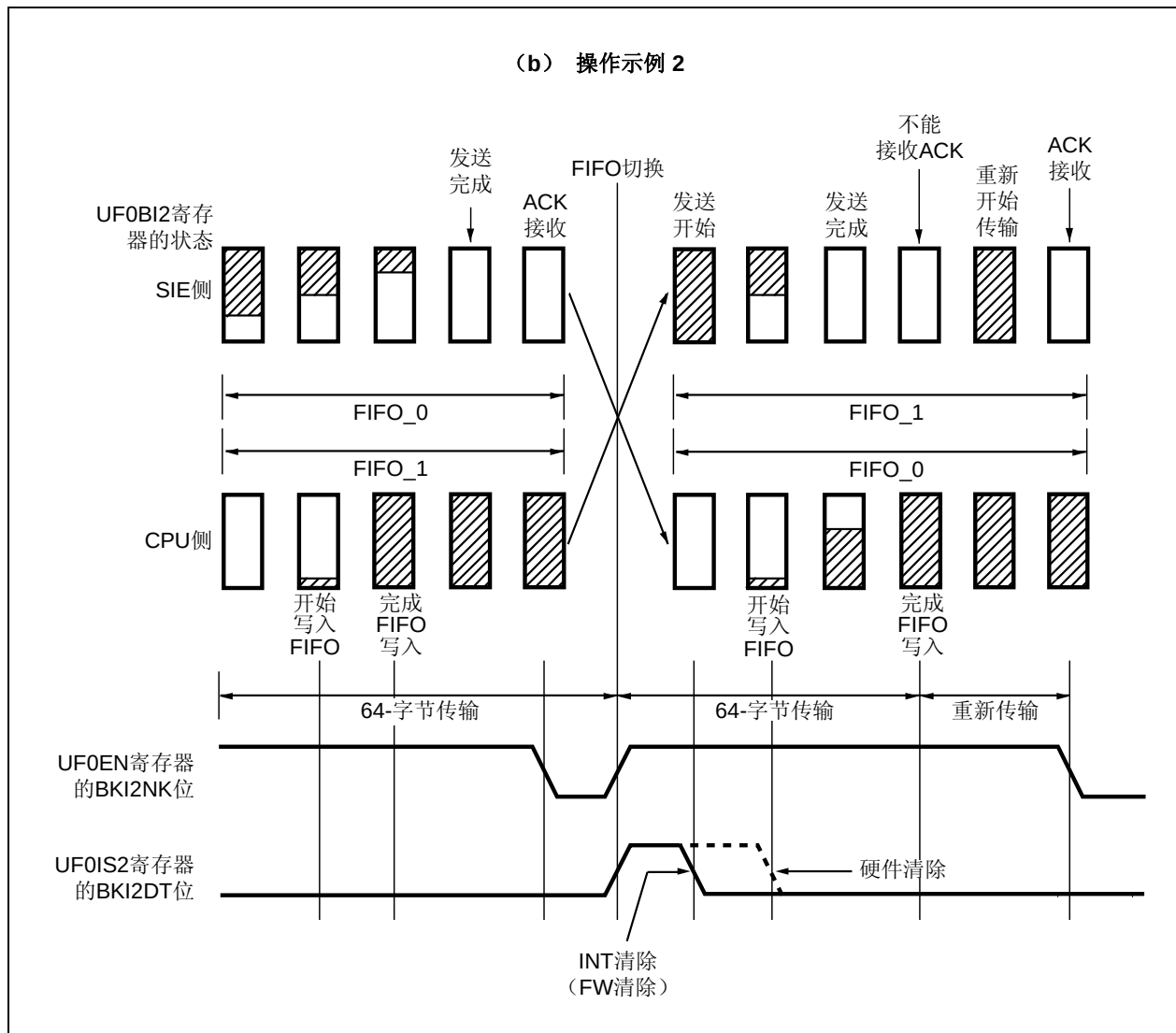
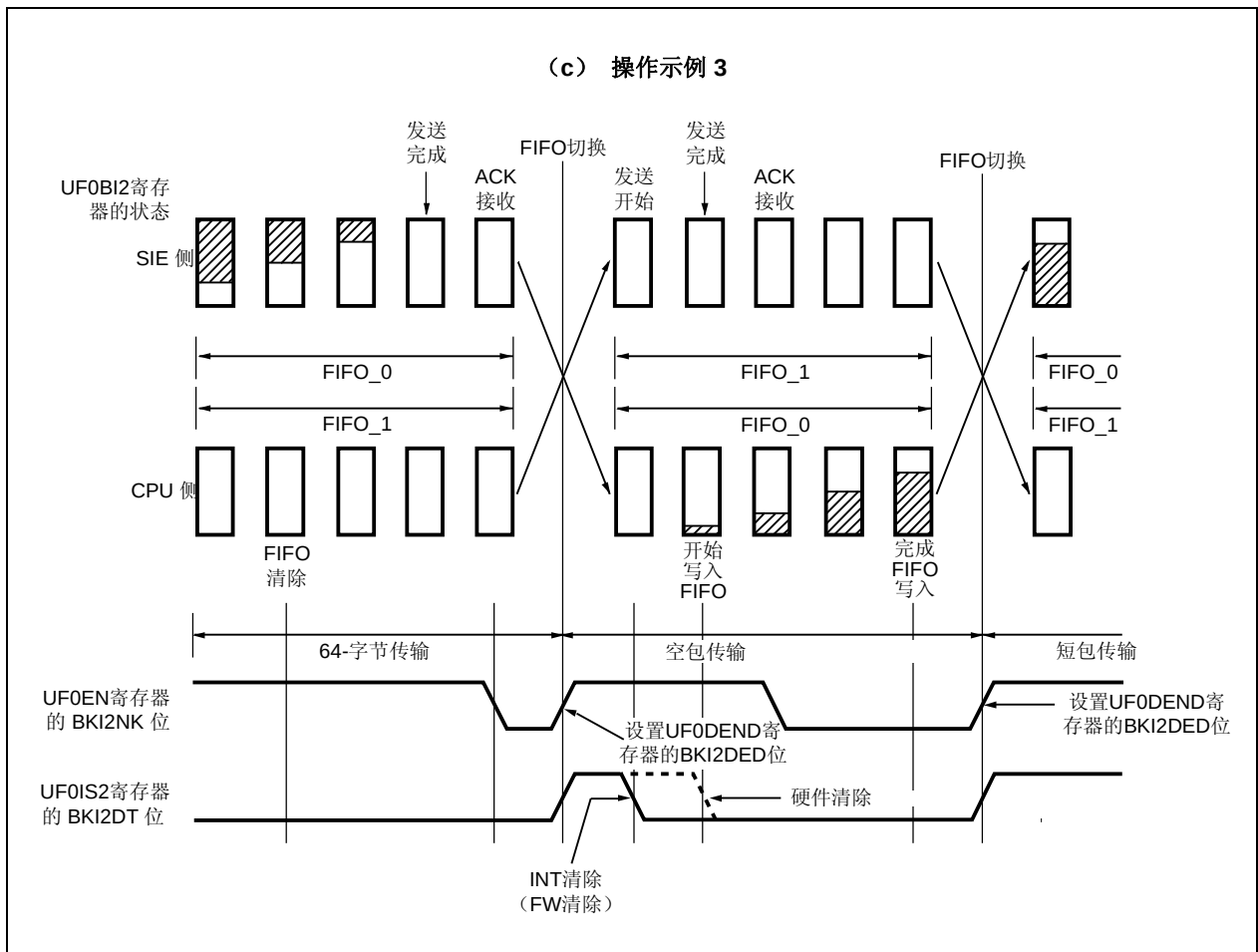


图 21-10. UF0BI2 寄存器的操作 (3/3)



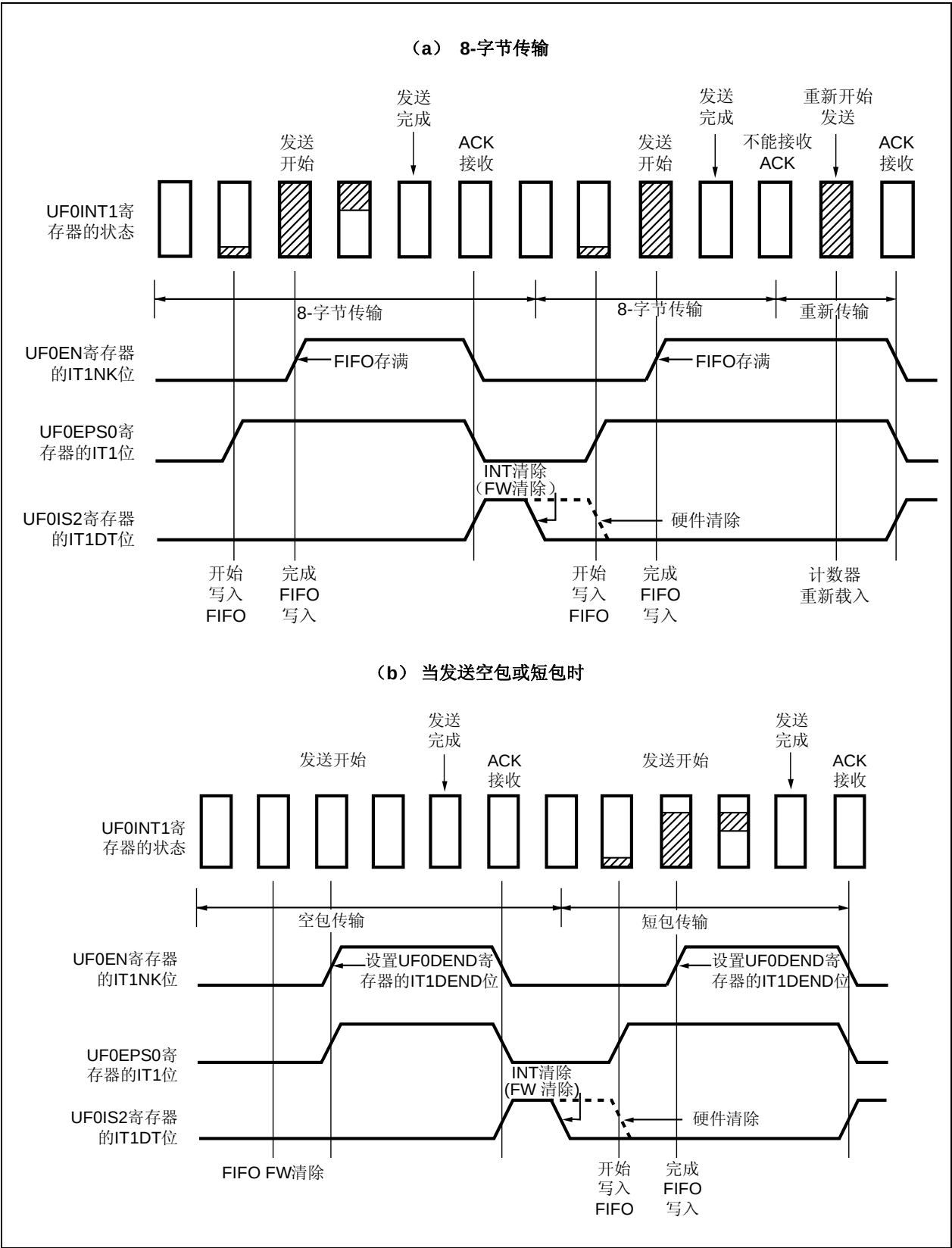
(11) UF0 中断 1 寄存器 (UF0INT1)

UF0INT1 寄存器是一个存储端点 7 数据（将要被传送到 SIE）的 8-字节的 FIFO。
该寄存器只能按字节进行写入操作。读取该寄存器时，会读取到 00H。
仅当 UF0EN 寄存器的 IT1NK 位设为 1（不发送 NAK 时）时，硬件才将数据与端点 7 IN 令牌同步地发送到 USB 总线。当发送数据且主机正确接收数据时，硬件自动将 UF0EN 寄存器的 IT1NK 位清为 0。当将数据写入 UF0INT1 寄存器且将 UF0DEND 寄存器的 IT1DEND 位设为 1 时，会发送一个短包（UF0EPS0 寄存器的 IT1 位 = 1（数据存在））。当清除 UF0INT1 寄存器且将 UF0DEND 寄存器的 IT1DEND 位设为 1（UF0EPS0 寄存器的 IT1 位 = 1（数据存在））时，会发送一个空包。

UF0INT1	7	6	5	4	3	2	1	0	地址	复位后
	IT17	IT16	IT15	IT14	IT13	IT12	IT11	IT10	00200114H	不确定
位号		位名称		功能						
7 至 0		IT17 至 IT10		这些位存储端点 7 的数据。						

UF0INT1 寄存器的操作如下图所示。

图 21-11. UF0INT1 寄存器的操作



21.6.5 EPC请求数据寄存器

(1) UF0 设备状态寄存器 L (UF0DSTL)

该寄存器存储响应 GET_STATUS 设备请求的返回值。
该寄存器可按 8 位为单位进行读写。
接收到 GET_STATUS 设备请求时，硬件会自动将该寄存器的内容发送到主机。

注意事项 要重写该寄存器，需在读取寄存器内容之前将 EP0NKA 位设为 1，且在确认该位已被设置后重写寄存器内容，以防读取访问和写入访问之间冲突。

UF0DSTL	7	6	5	4	3	2	1	0	地址	复位后
	0	0	0	0	0	0	RMWK	SFPW	00200144H	00H
位号	位名称		功能							
1	RMWK		该位规定是否使用设备的远程唤醒功能。 1: 允许 0: 禁止 如果设备支持远程唤醒功能，则在接收到 SET_FEATURE 设备请求时通过硬件将该位设为 1，而在接收到 CLEAR_FEATURE 设备请求时通过硬件将该位清除为 0。如果设备不支持远程唤醒功能，则应确保主机不产生 SET_FEATURE 设备请求。							
0	SFPW		该位表示设备是自供电还是总线供电。 1: 自供电 0: 总线供电							

(2) UF0 EP0 状态寄存器 L (UF0E0SL)

该寄存器存储响应 GET_STATUS 端点 0 请求的返回值。

该寄存器可按 8 位为单位进行读写。但是，注意，仅当 EP0NKA 位设为 1 时可将数据写入该寄存器。

如果在 USBF 中发送错误，则通过 FW 将 E0HALT 位设为 1。接收 USB-侧到端点 0 的访问期间，忽略对该寄存器的写入访问。

当 E0HALT 位由 FW 设为 1 时，如果刚才的控制传输是用于 SET_FEATURE 端点 0、CLEAR_FEATURE 端点 0、GET_STATUS 端点 0 请求或 FW-处理请求的，则直到接收下一个 SETUP 令牌之前，E0HALT 位并不反映。

接收到 GET_STATUS 端点 0 请求时，硬件会自动将该寄存器的内容发送到主机。如果端点 0 停止，则 UF0E0W 和 UF0E0R 寄存器会被清除，且 UF0E0N 寄存器的 EP0NKW 和 EP0NKR 位被清除为 0。

注意事项 要重写该寄存器，需在读取寄存器内容之前将 EP0NKA 位设为 1，且在确认该位已被设置后重写寄存器内容，以防读取访问和写入访问之间冲突。

UF0E0SL	7	6	5	4	3	2	1	0	地址	复位后
	0	0	0	0	0	0	0	E0HALT	0020014CH	00H

位号	位名称	功能
0	E0HALT	该位表示端点 0 的状态。 1: 停止 0: 未停止 当接收到 SET_FEATURE 端点 0 请求时硬件将该位设为 1，而接收到 CLEAR_FEATURE 端点 0 请求时硬件将该位清除为 0。DATA PID 初始化为 DATA0。

(3) UF0 EP1 状态寄存器 L (UF0E1SL)

该寄存器存储响应 GET_STATUS 端点 1 请求的返回值。

该寄存器可按 8 位为单位进行读写。但是，注意，仅当 EP0NKA 位设为 1 时可将数据写入该寄存器。

如果在端点 1 发生错误，则将 E1HALT 位设为 1。接收 USB-侧对端点 1 的访问期间忽略对该寄存器的写入访问。

接收到 GET_STATUS 端点 1 请求时，硬件会自动将该寄存器的内容发送到主机。如果端点 1 停止，则清除 UF0BI1 寄存器并将 BKI1NK 位清除为 0。

因为当执行控制传输之外的传输到端点 1 时，写入该寄存器总是被屏蔽，所以一定要检查该寄存器以查看数据是否正确写入至该寄存器。

注意事项 要重写该寄存器，需在读取寄存器内容之前将 EP0NKA 位设为 1，且在确认该位已置位后重写寄存器内容，以防读取访问和写入访问之间冲突。

	7	6	5	4	3	2	1	0	地址	复位后
UF0E1SL	0	0	0	0	0	0	0	E1HALT	00200150H	00H

位号	位名称	功能
0	E1HALT	该位表示端点 1 的状态。 1: 停止 0: 未停止 当接收到 SET_FEATURE 端点 1 请求时，硬件将该位设为 1。当与端点 1 相连的接口正确接收到 CLEAR_FEATURE 端点 1 请求、SET_CONFIGURATION 请求或 SET_INTERFACE 请求时，硬件将其清除为 0。DATA PID 初始化为 DATA0。

(4) UF0 EP2 状态寄存器 L (UF0E2SL)

该寄存器存储响应 GET_STATUS 端点 2 请求的返回值。

该寄存器可按 8 位为单位进行读写。但是，注意，仅当 EP0NKA 位设为 1 时可将数据写入该寄存器。

如果在端点 2 发生错误，则 E2HALT 位设为 1。接收 USB-侧对端点 2 的访问期间忽略对该寄存器的写入访问。

接收到 GET_STATUS 端点 2 请求时，硬件会自动将该寄存器的内容发送到主机。如果端点 2 停止，则清除 UF0BO1 寄存器并将 BKO1NK 位清除为 0。

因为当执行控制传输之外的传输到端点 2 时，写入该寄存器总是被屏蔽，所以一定要检查该寄存器以查看数据是否正确写入至该寄存器。

注意事项 要重写该寄存器，需在读取寄存器内容之前将 EP0NKA 位设为 1，且在确认该位已置位后重写寄存器内容，以防读取访问和写入访问之间冲突。

	7	6	5	4	3	2	1	0	地址	复位后
UF0E2SL	0	0	0	0	0	0	0	E2HALT	00200154H	00H

位号	位名称	功能
0	E2HALT	该位表示端点 2 的状态。 1: 停止 0: 未停止 当接收到 SET_FEATURE 端点 2 请求时，硬件将该位设为 1。当与端点 2 相连的接口正确接收到 CLEAR_FEATURE 端点 2 请求、SET_CONFIGURATION 请求或 SET_INTERFACE 请求时，硬件将其清除为 0。DATA PID 初始化为 DATA0。

(5) UF0 EP3 状态寄存器 L (UF0E3SL)

该寄存器存储响应 GET_STATUS 端点 3 请求的返回值。

该寄存器可按 8 位为单位进行读写。但是，注意，仅当 EP0NKA 位设为 1 时可将数据写入该寄存器。

如果在端点 3 发生错误，则将 E3HALT 位设为 1。接收 USB-侧对端点 3 的访问期间忽略对该寄存器的写入访问。

接收到 GET_STATUS 端点 3 请求时，硬件会自动将该寄存器的内容发送到主机。如果端点 3 停止，则清除 UF0BI2 寄存器并将 BKI2NK 位清除为 0。

因为当执行控制传输之外的传输到端点 3 时，写入该寄存器总是被屏蔽，所以一定要检查该寄存器以查看数据是否正确写入至该寄存器。

注意事项 要重写该寄存器，需在读取寄存器内容之前将 EP0NKA 位设为 1，且在确认该位已置位后重写寄存器内容，以防读取访问和写入访问之间冲突。

	7	6	5	4	3	2	1	0	地址	复位后
UF0E3SL	0	0	0	0	0	0	0	E3HALT	00200158H	00H

位号	位名称	功能
0	E3HALT	该位表示端点 3 的状态。 1: 停止 0: 未停止 当接收到 SET_FEATURE 端点 3 请求时，硬件将该位设为 1。当与端点 3 相连的接口正确接收到 CLEAR_FEATURE 端点 3 请求、SET_CONFIGURATION 请求或 SET_INTERFACE 请求时，硬件将其清除为 0。DATA PID 初始化为 DATA0。

(6) UF0 EP4 状态寄存器 L (UF0E4SL)

该寄存器存储响应 GET_STATUS 端点 4 请求的返回值。

该寄存器可按 8 位为单位进行读写。但是，注意，仅当 EP0NKA 位设为 1 时可将数据写入该寄存器。

如果在端点 4 发生错误，则将 E4HALT 位设为 1。接收 USB-侧对端点 4 的访问期间忽略对该寄存器的写入访问。

接收到 GET_STATUS 端点 4 请求时，硬件会自动将该寄存器的内容发送到主机。如果端点 4 停止，则清除 UF0BO2 寄存器并将 BKO2NK 位清除为 0。

因为当执行控制传输之外的传输到端点 4 时，写入该寄存器总是被屏蔽，所以一定要检查该寄存器以查看数据是否正确写入至该寄存器。

注意事项 要重写该寄存器，需在读取寄存器内容之前将 EP0NKA 位设为 1，且在确认该位已置位后重写寄存器内容，以防读取访问和写入访问之间冲突。

UF0E4SL	7	6	5	4	3	2	1	0	地址	复位后
	0	0	0	0	0	0	0	E4HALT	0020015CH	00H

位号	位名称	功能
0	E4HALT	该位表示端点 4 的状态。 1: 停止 0: 未停止 当接收到 SET_FEATURE 端点 4 请求时，硬件将该位设为 1。当与端点 4 相连的接口正确接收到 CLEAR_FEATURE 端点 4 请求、SET_CONFIGURATION 请求或 SET_INTERFACE 请求时，硬件将其清除为 0。DATA PID 初始化为 DATA0。

(7) UF0 EP7 状态寄存器 L (UF0E7SL)

该寄存器存储响应 GET_STATUS 端点 7 请求的返回值。

该寄存器可按 8 位为单位进行读写。但是，注意，仅当 EP0NKA 位设为 1 时可将数据写入该寄存器。

如果在端点 7 发生错误，则将 E7HALT 位设为 1。接收 USB-侧对端点 7 的访问期间忽略对该寄存器的写入访问。

接收到 GET_STATUS 端点 7 请求时，硬件会自动将该寄存器的内容发送到主机。如果端点 7 停止，则清除 UF0INT1 寄存器并将 IT1NK 位清除为 0。

因为当执行控制传输之外的传输到端点 7 时，写入该寄存器总是被屏蔽，所以一定要检查该寄存器以查看数据是否正确写入至该寄存器。

注意事项 要重写该寄存器，需在读取寄存器内容之前将 EP0NKA 位设为 1，且在确认该位已置位后重写寄存器内容，以防读取访问和写入访问之间冲突。

	7	6	5	4	3	2	1	0	地址	复位后
UF0E7SL	0	0	0	0	0	0	0	E7HALT	00200168H	00H

位号	位名称	功能
0	E7HALT	该位表示端点 7 的状态。 1: 停止 0: 未停止 当接收到 SET_FEATURE 端点 7 请求时，硬件将该位设为 1。当与端点 7 相连的接口正确接收到 CLEAR_FEATURE 端点 7 请求、SET_CONFIGURATION 请求或 SET_INTERFACE 请求时，硬件将其清除为 0。DATA PID 初始化为 DATA0。

(8) UF0 地址寄存器 (UF0ADRS)

该寄存器存储设备的地址。
本寄存器为只读寄存器，可以按 8 位读取。
由 SET_ADDRESS 请求发送的设备地址解析后将结果自动写入该寄存器。如果 SET_ADDRESS 请求由 FW 处理，则在状态阶段接收 SUCCESS 信号时，将该寄存器的值作为设备地址呈现。

注意事项 不要对该寄存器进行写入访问。如果写入，操作得不到保证。

	7	6	5	4	3	2	1	0	地址	复位后
UF0ADRS	0	ADRS6	ADRS5	ADRS4	ADRS3	ADRS2	ADRS1	ADRS0	00200180H	00H
位号		位名称	功能							
6 至 0		ADRS6 至 ADRS0	这些位保持 SIE 的设备地址。							

(9) UF0 配置寄存器 (UF0CNF)

该寄存器存储将要返回以响应 GET_CONFIGURATION 请求的值。

本寄存器为只读寄存器，可以按 8 位读取。

当接收 SET_CONFIGURATION 请求时，将其 wValue 自动写入到该寄存器。

当检测到该寄存器的值由 00H 变为非 00H 时，将 UF0MODS 寄存器的 CONF 位设为 1。如果 SET_CONFIGURATION 请求由 FW 处理，则一旦将数据写入该寄存器，立即会在 UF0MODS 寄存器中呈现该寄存器的状态（状态阶段完成之前 CONF 位 = 1）。

注意事项 不要对该寄存器进行写入访问。如果写入，操作得不到保证。

UF0CNF	7	6	5	4	3	2	1	0	地址	复位后
	0	0	0	0	0	0	CONF1	CONF0	00200182H	00H
位号		位名称		功能						
1, 0		CONF1, CONF0		这些位保持将要返回以响应 GET_CONFIGURATION 请求的数据。						

(10) UF0 接口 0 寄存器 (UF0IF0)

该寄存器存储响应 GET_INTERFACE wIndex = 0 请求的返回值。

本寄存器为只读寄存器，可以按 8 位读取。

当接收 SET_INTERFACE 请求时，将其 wValue 自动写入到该寄存器。

如果 SET_INTERFACE 请求由 FW 处理，则将 wIndex 和 wValue 解码，并自动改变端点的设置。此时，根据设置将目标端点的状态位和 DPID 自动清除为 0。FIFO 不会被自动清除。

注意事项 不要对该寄存器进行写入访问。如果写入，操作得不到保证。

UF0IF0	7	6	5	4	3	2	1	0	地址	复位后
	0	0	0	0	0	IF02	IF01	IF00	00200184H	00H
位号		位名称		功能						
2 至 0		IF02 至 IF00		这些位保持响应 GET_INTERFACE wIndex = 0 请求的返回数据。						

(11) UF0 接口 1 至 4 寄存器 (UF0IF1 至 UF0IF4)

这些寄存器存储响应 GET_INTERFACE wIndex = n (n = 1 至 4) 请求的返回值。

这些寄存器为只读寄存器，可以按 8 位读取。

当接收 SET_INTERFACE 请求时，将其 wValue 自动写入到这些寄存器。

根据 UF0AIFN 和 UF0AAS 寄存器的设置使得这些寄存器生效。

如果 SET_INTERFACE 请求由 FW 处理，则将 wIndex 和 wValue 解码，并自动改变端点的设置。此时，根据设置将目标端点的状态位和 DPID 自动清除为 0。FIFO 不会被自动清除。

注意事项 不要对该寄存器进行写入访问。如果写入，操作得不到保证。

	7	6	5	4	3	2	1	0	地址	复位后
UF0IF1	0	0	0	0	0	IF12	IF11	IF10	00200186H	00H
UF0IF2	0	0	0	0	0	IF22	IF21	IF20	00200188H	00H
UF0IF3	0	0	0	0	0	IF32	IF31	IF30	0020018AH	00H
UF0IF4	0	0	0	0	0	IF42	IF41	IF40	0020018CH	00H

位号	位名称	功能
2 至 0	IFn2 至 IFn0	这些位保持响应 GET_INTERFACE wIndex = n 请求的返回数据。

备注 n = 1 至 4

(12) UF0 描述符长度寄存器 (UF0DSCL)

该寄存器存储响应 GET_DESCRIPTOR 配置请求的返回值长度。该寄存器的值为 UF0CIEn 寄存器 (n = 0 至 255) 所设置的所有描述符的字节个数减 1。根据该寄存器的值来确定响应 GET_DESCRIPTOR 配置请求的返回全部说明符的总长度。

该寄存器可按 8 位为单位进行读写。但是，仅当 EP0NKA 位设为 1 时可将数据写入该寄存器。

自动控制 wLength 的处理。如果将该寄存器设为 00H，这意味着将要返回的说明符的长度为一个字节。如果将该寄存器设为 FFH，则返回长度为 256 个字节的说明符。当使用长度超过 256 个字节的说明符时，将 UF0MODC 寄存器的 CDCGDST 位设为 1 并通过 FW 处理 GET_DESCRIPTOR 请求（此时，UF0MODS 寄存器的 CDCGD 位也被设为 1）。

注意事项 要重写该寄存器，需在读取寄存器内容之前将 EP0NKA 位设为 1，且在确认该位已置位后重写寄存器内容，以防读取访问和写入访问之间冲突。

UF0DSCL	7	6	5	4	3	2	1	0	地址	复位后
	DPL7	DPL6	DPL5	DPL4	DPL3	DPL2	DPL1	DPL0	002001A0H	00H
位号		位名称		功能						
7 至 0		DPL7 至 DPL0		这些位设置为响应 GET_DESCRIPTOR 配置请求的所有描述符字节数。						

(13) UF0 设备描述符寄存器 0 至 17 (UF0DD0 至 UF0DD17)

这些寄存器存储响应 GET_DESCRIPTOR 设备请求的返回值。
这些寄存器可以按 8 位读取或写入。但是，仅当 EP0NKA 位设为 1 时可将数据写入这些寄存器。

- 注意事项
- 1. 要重写这些寄存器，需在读取寄存器内容之前将 EP0NKA 位设为 1，且在确认该位已名置位后重写寄存器内容，以防读取访问和写入访问之间冲突。
 - 2. 使用 USB 规范 2.0 版本和最新类规范所定义的值作为设置值。

	7	6	5	4	3	2	1	0	地址	复位后
UF0DDn									参见 表 21-5。	不确定
(n = 0 至 17)										

表 21-5. UF0 设备描述符寄存器的映射和数据

符号	地址	字段名	内容
UF0DD0	002001A2H	bLength	该描述符的大小
UF0DD1	002001A4H	bDescriptorType	设备描述符类型
UF0DD2	002001A6H	bcdUSB	USB 规范版本号小数点后的值
UF0DD3	002001A8H		USB 规范版本号小数点前的值
UF0DD4	002001AAH	bDeviceClass	类代码
UF0DD5	002001ACH	bDeviceSubClass	子类代码
UF0DD6	002001AEH	bDeviceProtocol	协议代码
UF0DD7	002001B0H	bMaxPacketSize0	端点 0 最大包的大小
UF0DD8	002001B2H	idVendor	厂家 ID 的低位值
UF0DD9	002001B4H		厂家 ID 的高位值
UF0DD10	002001B6H	idProduct	产品 ID 的低位值
UF0DD11	002001B8H		产品 ID 的高位值
UF0DD12	002001BAH	bcdDevice	设备序列号的低位值
UF0DD13	002001BCH		设备序列号的高位值
UF0DD14	002001BEH	iManufacturer	说明生产商的字符串描述符的指针
UF0DD15	002001C0H	iProduct	说明产品的字符串描述符的目录
UF0DD16	002001C2H	ISerialNumber	说明设备序列号的字符串描述符的指针
UF0DD17	002001C4H	BNumConfigurations	可设置配置的数目

(14) UF0 配置/接口/端点描述符寄存器 0 至 255 (UF0CIE0 至 UF0CIE255)

这些寄存器存储响应 GET_DESCRIPTOR 配置请求的返回值。
这些寄存器可以按 8 位读取或写入。但是，仅当 EP0NKA 位设为 1 时可将数据写入这些寄存器。
在这些寄存器中可存储最多为 256 个字节的描述符信息。按照配置、接口和端点的顺序存储各描述符。如果有两个或两个以上的接口，则在接口描述符之后重复存储数据。

表 21-6. UF0CIEn 寄存器的映射

地址	所存储的描述符
002001C6H	配置描述符（9 个字节）
002001D8H	接口描述符（9 个字节）
002001EAH	端点 1 描述符（7 个字节）
002001F8H	端点 2 描述符（7 个字节）
00200206H	端点 3 描述符（7 个字节）
:	:
002002xxH	接口描述符（9 个字节）
002002xxH+9	端点 1 描述符（7 个字节）
002002xxH+16	端点 2 描述符（7 个字节）
002002xxH+23	端点 3 描述符（7 个字节）
:	:

可设置到这些寄存器的有效数据范围根据 UF0DSCL 寄存器设置的不同而异。除了表 21-7 列出的描述符之外，还可以存储类别和厂商特有的描述符。
如果所有的值是固定的，则可将它们存储在 ROM 中。

- 注意事项
1. 要重写这些寄存器，需在读取寄存器内容之前将 EP0NKA 位设为 1，且在确认该位已设置后重写寄存器内容，以防读取访问和写入访问之间冲突。
 2. 使用 USB 规范 2.0 版本和最新类规范所定义的值作为设置值。

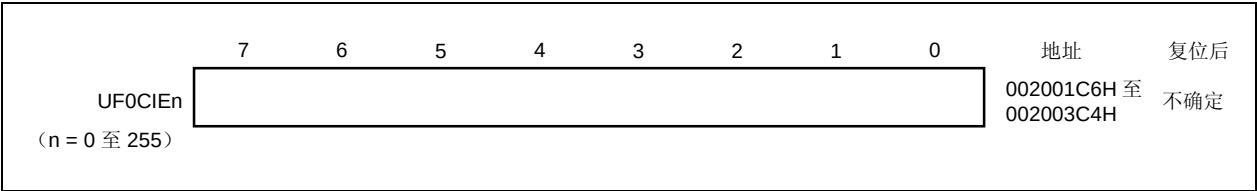


表 21-7. UF0CIEn 寄存器的数据

(a) 配置描述符 (9 个字节)

偏移量	字段名	内容
0	bLength	该描述符的大小
1	bDescriptorType	描述符类型
2	wTotalLength	所有接口和所有端点描述符和配置的总字节个数的低位值
3		所有接口和所有端点描述符和配置的总字节个数的高位值
4	bNumInterface	接口个数
5	bConfigurationValue	用来选择该配置的值
6	iConfiguration	说明该配置的字符串描述符的指针
7	bmAttributes	该配置的特性 (自供电, 不可远程唤醒)
8	MaxPower	该配置下的最大功耗 (单位: mA) [※]

注 以 2mA 为单位显示。(例如: 50 = 100 mA)

(b) 接口描述符 (9 个字节)

偏移量	字段名	内容
0	bLength	该描述符的大小
1	bDescriptorType	描述符类型
2	bInterfaceNumber	该接口的值
3	bAlternateSetting	用来选择接口可选设置的值
4	bNumEndpoints	可用端点的数目
5	bInterfaceClass	类代码
6	bInterfaceSubClass	子类代码
7	bInterfaceProtocol	协议代码
8	Interface	说明该接口的字符串描述符的指针

(c) 端点描述符 (7 个字节)

偏移量	字段名	内容
0	bLength	该描述符的大小
1	bDescriptorType	描述符类型
2	bEndpointAddress	该端点的地址/传输方向
3	bmAttributes	传送类型
4	wMaxPaketSize	传送数据最大数目的低位值
5		传送数据最大数目的高位值
6	bInterval	传送间隔

21.6.6 桥寄存器

(1) 桥中断控制寄存器 (BRGINTT)

BRGINTT 寄存器控制中断生成状态下的 DMA 传输状态，以及从 EPC 至桥电路的各个端点 (EP1 至 EP4)。
BRGINTT 寄存器可以按 16 位进行读写。

复位后： 0000H R/W 地址： 00200400H								
BRGINTT	15	14	13	12	11	10	9	8
	0	0	0	0	EP4INT	EP3INT	EP2INT	EP1INT
	7	6	5	4	3	2	1	0
	0	0	0	0	0	EPCINT2B	EPCINT1B	EPCINT0B
位号	位名称	功能						
11	EP4INT	EP4 中，当 DMA 传输正常终止，或 DMA 传输中错误结束时设置该位。通过写入“1”来清除为“0”。 0: DMA 传输未完成 1: DMA 传输完成						
10	EP3NT	EP3 中，当 DMA 传输正常终止，或 DMA 传输中错误结束时设置该位。通过写入_1 来清除为“0”。 0: DMA 传输未完成 1: DMA 传输完成						
9	EP2NT	EP2 中，当 DMA 传输正常终止，或 DMA 传输中错误结束时设置该位。通过写入_1 来清除为“0”。 0: DMA 传输未完成 1: DMA 传输完成						
8	EP1NT	EP1 中，当 DMA 传输正常终止，或 DMA 传输中错误结束时设置该位。通过写入 01 来清除为“0”。 0: DMA 传输未完成 1: DMA 传输完成						
2	EPCINT2B	显示来自 EPC 的中断信号“EPC_INT2B”的状态。 清除来自 EPC 寄存器请求的控制 0: 不产生中断 1: 产生中断						
1	EPCINT1B	显示来自 EPC 的中断信号“EPC_INT1B 的状态。” 清除来自 EPC 寄存器请求的控制 0: 不产生中断 1: 产生中断						
0	EPCINT0B	显示来自 EPC 的中断信号“EPC_INT0B 的状态。” 清除来自 EPC 寄存器请求的控制 0: 不产生中断 1: 产生中断						

(2) 桥中断使能寄存器 (BRGINTE)

BRGINTE 寄存器控制使能还是禁止桥电路中产生的中断。

BRGINTE 寄存器可以按 16 位进行读写。

复位后: 0000H R/W 地址: 00200402H

	15	14	13	12	11	10	9	8
BRGINTE	0	0	0	0	EP4INTN	EP3INTN	EP2INTN	EP1INTN
	7	6	5	4	3	2	1	0
	0	0	0	0	0	EPC INT2BEN	EPC INT1BEN	EPC INT0BEN

位号	位名称	功能
11	EP4INTN	当设置 EP4INT 位时设置中断发送使能或者禁止 0: 禁止中断 1: 允许中断
10	EP3INTN	当设置 EP3INT 位时设置中断发送使能或者禁止 0: 禁止中断 1: 允许中断
9	EP2INTN	当设置 EP2INT 位时设置中断发送使能或者禁止 0: 禁止中断 1: 允许中断
8	EP1INTN	当设置 EP1INT 位时设置中断发送使能或者禁止 0: 禁止中断 1: 允许中断
2	EPCINT2BEN	当设置 EPCINT2BEN 位时设置中断发送使能或者禁止 0: 禁止中断 1: 允许中断
1	EPCINT1BEN	当设置 EPCINT1BEN 位时设置中断发送使能或者禁止 0: 禁止中断 1: 允许中断
0	EPCINT0BEN	当设置 EPCINT0BEN 位时设置中断发送使能或者禁止 0: 禁止中断 1: 允许中断

(3) EPC 宏控制寄存器 (EPCCLT)

EPCCLT 寄存器控制 EPC 宏的复位发生器。
EPCCLT 寄存器可以按 16 位进行读写。

复位后：0000H R/W 地址：00200404H							
EPCCLT	15	14	13	12	11	10	9
	0	0	0	0	0	0	0
	7	6	5	4	3	2	1
	0	0	0	0	0	0	0
EPCRST							
0	EPCRST	设置 EPC 的复位源。 0: 复位解除 1: 产生复位					

(4) CPU I/F 总线控制寄存器 (CPUBCTL)

CPUBCTL 寄存器控制桥电路和 CPU 之间的接口。

CPUBCTL 寄存器可以按 16 位进行读写。

复位后： 不确定 R/W 地址： 00200408H

	15	14	13	12	11	10	9	8
CPUBCTL	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
	0	0	0	0	0	BULKWAIT	DATAWAIT	NOWAIT

位号	位名称	功能
2	BULKWAIT	当访问 bulk 寄存器时强制插入 1 个等待 (bulk 等待)。 0: 没有强制插入 bulk 等待 [※] (默认值) 1: 强制插入 bulk 等待 注 写入访问时该设置无效, 强制插入 bulk 等待。
1	DATAWAIT	CPU 总线周期之后强制插入 1 个等待 (数据等待)。 0: 没有强制插入数据等待 (默认值) 1: 强制插入数据等待
0	NOWAIT	设置使能/禁止 CPU 总线周期的无等待操作。 0: 禁止无等待 [※] (默认值) 1: 允许无等待 注 插入一个或更多等待。

21.6.7 DMA寄存器

(1) EPn DMA 控制寄存器 1 (UF0E1DC1 至 UF0E4DC1)

UF0E1DC1 至 UF0E4DC1 寄存器控制端点 n (EPn) 的 DMA 传输。(n=1 至 4)

UF0E1DC1 至 UF0E4DC1 寄存器可以按 16 位读出或写入。

(1/2)

复位后： 0000H R/W 地址： 00200500H								
UF0E1DC1	15	14	13	12	11	10	9	8
	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
	0	0	EP1BULK2	EP1BULK1	EP1BULK0	EP1STOP	EP1REQ	EP1DMAEN

复位后： 0000H R/W 地址： 00200504H								
UF0E2DC1	15	14	13	12	11	10	9	8
	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
	0	0	EP2BULK2	EP2BULK1	EP2BULK0	EP2STOP	EP2REQ	EP2DMAEN

复位后： 0000H R/W 地址： 00200508H								
UF0E3DC1	15	14	13	12	11	10	9	8
	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
	0	0	EP3BULK2	EP3BULK1	EP3BULK0	EP3STOP	EP3REQ	EP3DMAEN

复位后： 0000H R/W 地址： 0020050CH								
UF0E4DC1	15	14	13	12	11	10	9	8
	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
	0	0	EP4BULK2	EP4BULK1	EP4BULK0	EP4STOP	EP4REQ	EP4DMAEN

位号	位名称	功能																								
5 至 3	EPnBULK2, EPnBULK1, EPnBULK0	显示内部桥 bulk 传输的状态装置“BIN_STATE”的状态																								
		<table><tr><th>EPnBULK2</th><th>EPnBULK1</th><th>EPnBULK0</th><th>“BIN_STATE”状态</th></tr><tr><td>0</td><td>0</td><td>0</td><td>BIN_IDLE</td></tr><tr><td>0</td><td>0</td><td>1</td><td>BIN_CPU</td></tr><tr><td>0</td><td>1</td><td>0</td><td>BIN_EPC</td></tr><tr><td>0</td><td>1</td><td>1</td><td>BIN_CMP</td></tr><tr><td>1</td><td>0</td><td>0</td><td>BIN_END</td></tr></table>	EPnBULK2	EPnBULK1	EPnBULK0	“BIN_STATE”状态	0	0	0	BIN_IDLE	0	0	1	BIN_CPU	0	1	0	BIN_EPC	0	1	1	BIN_CMP	1	0	0	BIN_END
		EPnBULK2	EPnBULK1	EPnBULK0	“BIN_STATE”状态																					
		0	0	0	BIN_IDLE																					
		0	0	1	BIN_CPU																					
		0	1	0	BIN_EPC																					
		0	1	1	BIN_CMP																					
		1	0	0	BIN_END																					
2	EPnSTOP	显示来自 EPC 的 DMA 传输结束的状态（DMA 传输的结束因素） 0: 通过 EPn_TCNT 值为“0”结束 DMA 传输 1: 通过对“EPC_DMARQ_EPnB”求反结束 DMA 传输 通过将下一个 EP1_DMAEN 设为“1”自动清除（0）。																								
1	EPnREQ	显示来自 EPC 的“EPC_DMARQ_EPnB”信号的状态 0: DMA 请求信号 1: 无 DMA 请求信号																								
0	EPnDMAEN	设置来自 EPC 的 DMA 请求的控制 0: 屏蔽 DMA 请求 1: 允许 DMA 请求 通过完成 EPn_TCNT 中包的传输设置个数自动清除（0），或通过对 DMARQ_EPnB 求反完成 DMA 传输。 注意事项 在强制结束时设置值得不到保证。																								

备注 n = 1 至 4

(2) EPn DMA 控制寄存器 2 (UF0E1DC2 至 UF0E4DC2)

UF0E1DC2 至 UF0E4DC2 寄存器控制端点 n (EPn) 的 DMA 传输。(n=1 至 4)

UF0E1DC2 至 UF0E4DC2 寄存器可以按 16 位读出或写入。

(1/2)

复位后: 0000H R/W 地址: 00200502H

	15	14	13	12	11	10	9	8
UF0E1DC2	EP1	EP1	EP1	EP1	EP1	EP1	EP1	EP1
	TCNT15	TCNT14	TCNT13	TCNT12	TCNT11	TCNT10	TCNT9	TCNT8
	7	6	5	4	3	2	1	0
	EP1	EP1	EP1	EP1	EP1	EP1	EP1	EP1
	TCNT7	TCNT6	TCNT5	TCNT4	TCNT3	TCNT2	TCNT1	TCNT0

复位后: 0000H R/W 地址: 00200506H

	15	14	13	12	11	10	9	8
UF0E2DC2	EP2	EPn	EP2	EP2	EP2	EP2	EP2	EP2
	TCNT15	TCNT14	TCNT13	TCNT12	TCNT11	TCNT10	TCNT9	TCNT8
	7	6	5	4	3	2	1	0
	EP2	EP2	EP2	EP2	EP2	EP2	EP2	EP2
	TCNT7	TCNT6	TCNT5	TCNT4	TCNT3	TCNT2	TCNT1	TCNT0

复位后: 0000H R/W 地址: 0020050AH

	15	14	13	12	11	10	9	8
UF0E3DC2	EP3	EP3	EP3	EP3	EP3	EP3	EP3	EP3
	TCNT15	TCNT14	TCNT13	TCNT12	TCNT11	TCNT10	TCNT9	TCNT8
	7	6	5	4	3	2	1	0
	EP3	EP3	EP3	EP3	EP3	EP3	EP3	EP3
	TCNT7	TCNT6	TCNT5	TCNT4	TCNT3	TCNT2	TCNT1	TCNT0

复位后: 0000H R/W 地址: 0020050EH

	15	14	13	12	11	10	9	8
UF0E4DC2	EP4	EP4	EP4	EP4	EP4	EP4	EP4	EP4
	TCNT15	TCNT14	TCNT13	TCNT12	TCNT11	TCNT10	TCNT9	TCNT8
	7	6	5	4	3	2	1	0
	EP4	EP4	EP4	EP4	EP4	EP4	EP4	EP4
	TCNT7	TCNT6	TCNT5	TCNT4	TCNT3	TCNT2	TCNT1	TCNT0

位号	位名称	功能
15 至 0	EPnTCNT15 至 EPnTCNT0	在 EPn 中设置 DMA 传输的字节个数。 各传输的 EPn_TCNT 的值递减为“0”之后结束 DMA 传输。 注意事项 1. 当 EPn_DMAEN = 0 时设置该寄存器。 2. 禁止将该寄存器设为“0”。 务必将该寄存器+1 的值设置为 DMA 传输计数器寄存器 DBC0 至 DBC3 的值。 3. 该寄存器的值反映桥内部的 bulk 传输计数器 BIN_TCNT 的值。且 BIN_TCNT 的值为“0”，EPn_TCN 也为“0”。 4. 强制终止时，停止更新 bulk 传输计数器 BIN_TCNT 的值。

备注 n = 1 至 4

21.6.8 Bulk-in寄存器

(1) UF0 EP1 bulk-in 传输数据寄存器 (UF0EP1BI)

UF0EP1BI 寄存器写入 EP1 的 bulk-in 传输数据。

UF0EP1BI 寄存器可以按 16 位读出或写入。

复位后： 0000H R/W 地址： 00201000H								
UF0EP1BI	15	14	13	12	11	10	9	8
	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
	EP1BI7	EP1BI6	EP1BI5	EP1BI4	EP1BI3	EP1BI2	EP1BI1	EP1BI0
位号	位名称	功能						
7 至 0	EP1BI7 至 EP1BI0	写入 EP1 的 bulk-out 传输数据。 通过将数据写入该寄存器将数据输出到 EPC。 如果使用该寄存器，则在 DMAC 的 DMA 目的地址寄存器（DDAn（n = 0 至 3））中设置地址（00201000H）。						

(2) UF0 EP3 bulk-in 传输数据寄存器 (UF0EP3BI)

UF0EP3BI 寄存器写入 EP1 的 bulk-in 传输数据。

UF0EP3BI 寄存器可以按 16 位读出或写入。

复位后: 0000H R/W 地址: 00202000H								
UF0EP3BI	15	14	13	12	11	10	9	8
	0	0	0	0	0	0	0	0
	7	6	5	4	3	2	1	0
	EP3BI7	EP3BI6	EP3BI5	EP3BI4	EP3BI3	EP3BI2	EP3BI1	EP3BI0
位号	位名称	功能						
7 至 0	EP3BI7 至 EP3BI0	写入 EP3 的 bulk-out 传输数据。 通过将数据写入该寄存器将数据输出到 EPC。 如果使用该寄存器, 则在 DMAC 的 DMA 目的地址寄存器 (DDAn (n = 0 至 3)) 中设置地址 (00202000H)。						

21.6.9 Bulk-out寄存器

(1) UF0 EP2 bulk-out 传输数据寄存器 (UF0EP2BO)

UF0EP2BO 寄存器写入 EP2 的 bulk-out 传输数据。

UF0EP2BO 寄存器可以按 16 位读出或写入。

复位后: 0000H R 地址: 00210000H

	15	14	13	12	11	10	9	8
UF0EP2BO	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
	EP2BO7	EP2BO6	EP2BO5	EP2BO4	EP2BO3	EP2BO2	EP2BO1	EP2BO0

位号	位名称	功能
7 至 0	EP2BO7 至 EP2BO0	读取 EP2 的 bulk-out 传输数据。 从该寄存器读取来自 EPC 宏的输入数据。 如果使用该寄存器, 则在 DMAC 的 DMA 源地址寄存器 (DSAn (n = 0 至 3)) 中设置地址 (00210000H)。

(2) UF0 EP4 bulk-out 传输数据寄存器 (UF0EP4BO)

UF0EP4BO 寄存器写入 EP4 的 bulk-out 传输数据。

UF0EP4BO 寄存器可以按 16 位读出或写入。

复位后: 0000H R 地址: 00220000H

	15	14	13	12	11	10	9	8
UF0EP4BO	0	0	0	0	0	0	0	0

	7	6	5	4	3	2	1	0
	EP4BO7	EP4BO6	EP4BO5	EP4BO4	EP4BO3	EP4BO2	EP4BO1	EP4BO0

位号	位名称	功能
7 至 0	EP4BO7 至 EP4BO0	读取 EP4 的 bulk-out 传输数据。 从该寄存器读取来自 EPC 宏的输入数据。 如果使用该寄存器, 则在 DMAC 的 DMA 源地址寄存器 (DSAn (n = 0 至 3)) 中设置地址 (00220000H)。

21.6.10 外设控制寄存器

(1) USBF DMA 请求使能寄存器 (UFDRQEN)

UFDRQEN 寄存器指定将要使用的 DMA 通道和将要发送的端点。

UFDRQEN 寄存器可以按字节或按位读写。

(1/2)

复位后： 0000H R/W 地址： 00240000H								
	15	14	13	12	11	10	9	8
UFDRQEN	RQ3UR3E	RQ2UR3E	RQ1UR3E	RQ0UR3E	RQ3UR2E	RQ2UR2E	RQ1UR2E	RQ0UR2E
	7	6	5	4	3	2	1	0
	RQ3UR1E	RQ2UR1E	RQ1UR1E	RQ0UR1E	RQ3UR0E	RQ2UR0E	RQ1UR0E	RQ0UR0E

位号	位名称	功能																														
15, 11, 7, 3	RQ3UR3E, RQ3UR2E, RQ3UR1E, RQ3UR0E	指定将要由 DMA 通道 3 发送的端点 n (EPn)。 (n = 1 至 4)																														
		<table><tr><td>RQ3UR3E</td><td>RQ3UR2E</td><td>RQ3UR1E</td><td>RQ3UR0E</td><td>由 DMA3 发送的 EP</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>EP4</td></tr><tr><td>0</td><td>1</td><td>0</td><td>0</td><td>EP3</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td><td>EP2</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>EP1</td></tr><tr><td colspan="4">其它</td><td>DMA3 不发送 EPn (不使用 DMA3)</td></tr></table>	RQ3UR3E	RQ3UR2E	RQ3UR1E	RQ3UR0E	由 DMA3 发送的 EP	1	0	0	0	EP4	0	1	0	0	EP3	0	0	1	0	EP2	0	0	0	1	EP1	其它				DMA3 不发送 EPn (不使用 DMA3)
		RQ3UR3E	RQ3UR2E	RQ3UR1E	RQ3UR0E	由 DMA3 发送的 EP																										
		1	0	0	0	EP4																										
		0	1	0	0	EP3																										
		0	0	1	0	EP2																										
		0	0	0	1	EP1																										
其它				DMA3 不发送 EPn (不使用 DMA3)																												
14, 10, 6, 2	RQ2UR3E, RQ2UR2E, RQ2UR1E, RQ2UR0E	指定将要由 DMA 通道 2 发送的端点 n (EPn)。 (n = 1 至 4)																														
		<table><tr><td>RQ2UR3E</td><td>RQ3UR2E</td><td>RQ3UR1E</td><td>RQ3UR0E</td><td>由 DMA2 发送的 EP</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>EP4</td></tr><tr><td>0</td><td>1</td><td>0</td><td>0</td><td>EP3</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td><td>EP2</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>EP1</td></tr><tr><td colspan="4">其它</td><td>DMA2 不发送 EPn (不使用 DMA2)</td></tr></table>	RQ2UR3E	RQ3UR2E	RQ3UR1E	RQ3UR0E	由 DMA2 发送的 EP	1	0	0	0	EP4	0	1	0	0	EP3	0	0	1	0	EP2	0	0	0	1	EP1	其它				DMA2 不发送 EPn (不使用 DMA2)
		RQ2UR3E	RQ3UR2E	RQ3UR1E	RQ3UR0E	由 DMA2 发送的 EP																										
		1	0	0	0	EP4																										
		0	1	0	0	EP3																										
		0	0	1	0	EP2																										
		0	0	0	1	EP1																										
其它				DMA2 不发送 EPn (不使用 DMA2)																												

位号	位名称	功能				
14, 10, 6, 2	RQ1UR3E, RQ1UR2E, RQ1UR1E, RQ1UR0E	指定将要由 DMA 通道 1 发送的端点 n (EPn)。 (n = 1 至 4)				
		RQ3UR3E	RQ3UR2E	RQ3UR1E	RQ3UR0E	由 DMA2 发送的 EP
		1	0	0	0	EP4
		0	1	0	0	EP3
		0	0	1	0	EP2
		0	0	0	1	EP1
		其它				DMA1 不发送 EPn (不使用 DMA1)
12, 8, 4, 0	RQ0UR3E, RQ0UR2E, RQ0UR1E, RQ0UR0E	指定将要由 DMA 通道 0 发送的端点 n (EPn)。 (n = 1 至 4)				
		RQ3UR3E	RQ3UR2E	RQ3UR1E	RQ3UR0E	由 DMA0 发送的 EP
		1	0	0	0	EP4
		0	1	0	0	EP3
		0	0	1	0	EP2
		0	0	0	1	EP1
		其它				DMA0 不发送 EPn (不使用 DMA0)

- 注意事项 1. 禁止将同一 DMA 传输目标设置到多个 DMA 通道，且禁止将多个 DMA 传输目的设置到同一个 DMA 通道。
2. 如果使用该寄存器的功能，则设置 DMA 触发因子寄存器 (DTFRn (n = 0 至 3)) 以禁止中断 (00H) 的 DMA 请求。

以下流程图说明了未连接主机且随后重新连接时的程序执行，以及加电时的程序执行。

图 21-12. 未连接主机且随后重新连接时的程序流程图

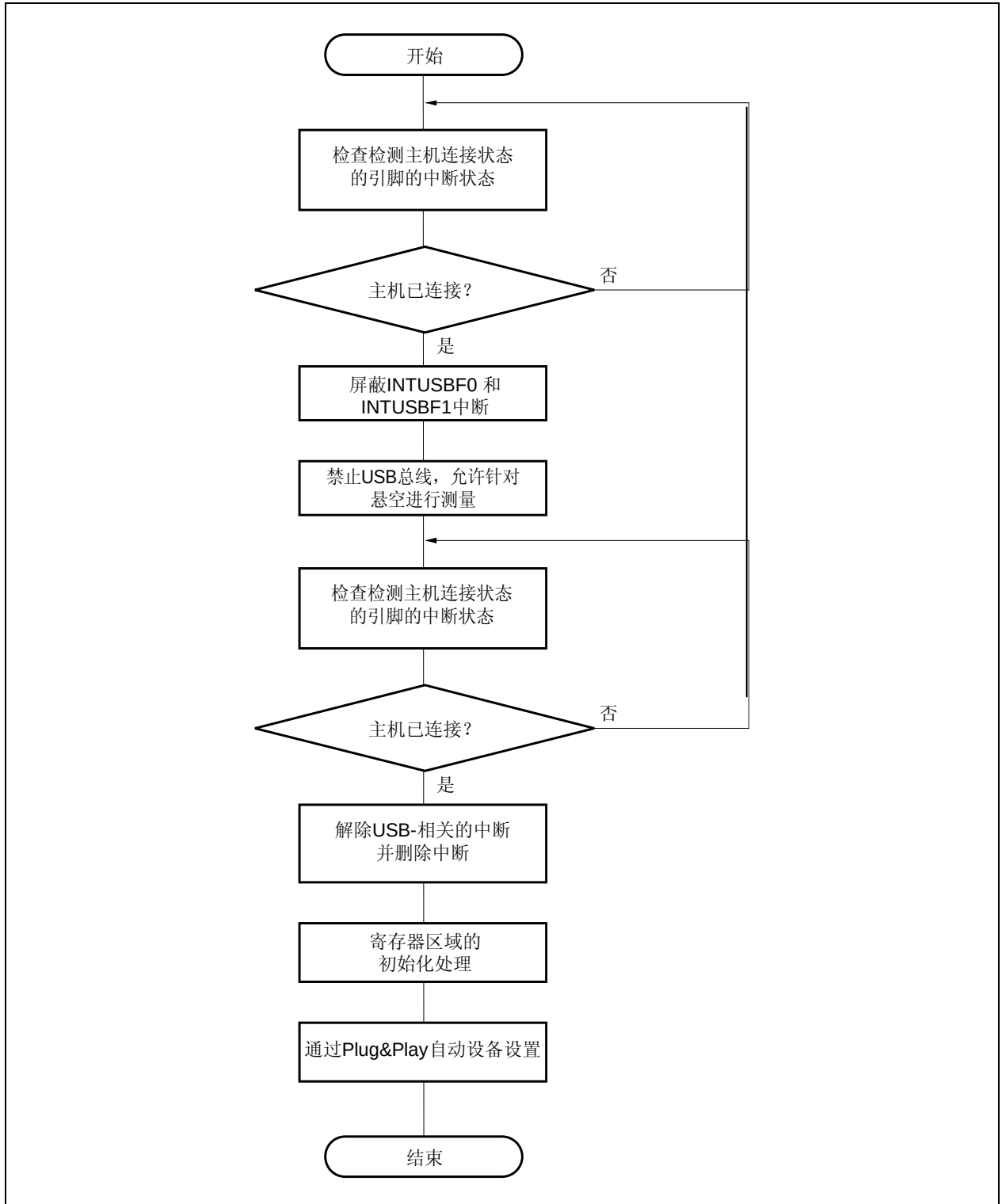
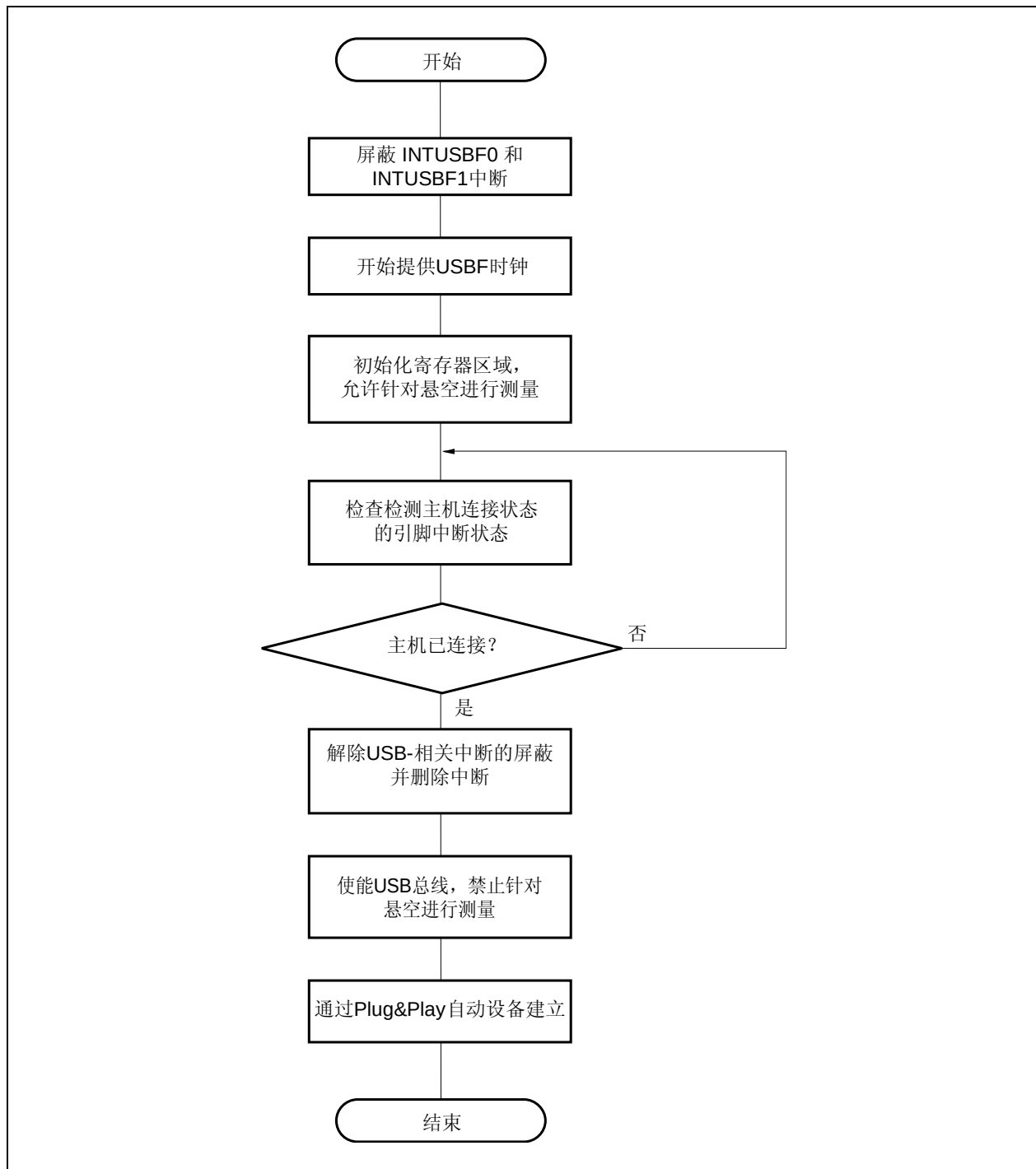


图 21-13. 加电时的程序流程图



21.7 STALL握手信号或不握手信号

USBF 的错误处理规定如下。

传送类型:	处理	目标包	错误类型	响应功能	处理
控制传输/ bulk 传输/ 中断传输	IN/OUT/SETUP	令牌	端点不受支持	无响应	无
			端点传输方向不匹配	无响应	无
			CRC 错误	无响应	无
			位填充错误	无响应	无
控制传输/ bulk 传输	OUT/SETUP	数据	超时	无响应	无
			PID 检查错误	无响应	无
			不支持 PID (除数据 PID 外)	无响应	无
			CRC 错误	无响应	删除接收的数据
			位填充错误	无响应	删除接收的数据
	OUT	数据	数据 PID 不匹配	ACK	删除接收的数据
控制传 (SETUP 阶段)	SETUP	数据	溢出	无响应	删除接收的数据
控制传输 (数据阶段)	OUT	数据	溢出	无响应 ^{注1}	将 UF0SDS 寄存器的 SNDSTL 位设为 1 并删除接收的数据
控制传输 (状态阶段)	OUT	数据	溢出	ACK 或 无响应 ^{注2}	将 UF0SDS 寄存器的 SNDSTL 位设为 1 并删除接收的数据
Bulk 传输	OUT	数据	溢出	无响应 ^{注1}	将 UF0EnSL 寄存器的 EnHALT (n = 0 至 4, 7) 位设为 1
控制传输/ bulk 传输/ 中断传输	IN	握手信号	PID 检查错误	—	保持发送的数据并重新发送数据 ^{注3}
			不支持 PID (除 ACK PID 之外)	—	保持发送的数据并重新发送数据 ^{注3}
			超时	—	保持发送的数据并重新发送数据 ^{注3}

注 1. 主机做出 STALL 响应以重新进行发送。

2. 如果发送的数据小于 MaxPacketSize 则会做出 ACK 响应, 且丢弃在状态阶段接收的数据。如果超出了 MaxPacketSize, 则不做出响应, 将 UF0SDS 寄存器的 SNDSTL 位设为 1 并丢弃所接收的数据。

3. 如果在控制传输期间接收到表示从数据阶段变为状态阶段的 OUT 处理, 则不处理错误并认为接收已正确完成。

注意事项 1. 由当前设置的候选编号判断目标端点有效还是无效。

2. 关于对端点 0 往返控制传输所包括的请求的响应, 参见 21.5 请求。

21.8 特殊状态下的寄存器值

表 21-8. 特殊状态下的寄存器值 (1/2)

寄存器名称	CPU 复位 (RESET) 之后	总线复位之后
UF0E0N 寄存器	00H	值被保持
UF0E0NA 寄存器	00H	值被保持
UF0EN 寄存器	00H	值被保持
UF0ENM 寄存器	00H	值被保持
UF0SDS 寄存器	00H	值被保持
UF0CLR 寄存器	00H	值被保持
UF0SET 寄存器	00H	值被保持
UF0EPS0 寄存器	00H	值被保持
UF0EPS1 寄存器	00H	值被保持
UF0EPS2 寄存器	00H	值被保持
UF0IS0 寄存器	00H	值被保持
UF0IS1 寄存器	00H	值被保持
UF0IS2 寄存器	00H	值被保持
UF0IS3 寄存器	00H	值被保持
UF0IS4 寄存器	00H	值被保持
UF0IM0 寄存器	00H	值被保持
UF0IM1 寄存器	00H	值被保持
UF0IM2 寄存器	00H	值被保持
UF0IM3 寄存器	00H	值被保持
UF0IM4 寄存器	00H	值被保持
UF0IC0 寄存器	FFH	值被保持
UF0IC1 寄存器	FFH	值被保持
UF0IC2 寄存器	FFH	值被保持
UF0IC3 寄存器	FFH	值被保持
UF0IC4 寄存器	FFH	值被保持
UF0IDR 寄存器	00H	值被保持
UF0DMS0 寄存器	00H	值被保持
UF0DMS1 寄存器	00H	值被保持
UF0FIC0 寄存器	00H	值被保持
UF0FIC1 寄存器	00H	值被保持
UF0DEND 寄存器	00H	值被保持
UF0GPR 寄存器	00H	值被保持
UF0MODC 寄存器	00H	值被保持
UF0MODS 寄存器	00H	位 2 (CONF)：清除 (0)， 其它位：值被保持
UF0AIFN 寄存器	00H	值被保持
UF0AAS 寄存器	00H	值被保持
UF0ASS 寄存器	00H	00H
UF0E1IM 寄存器	00H	值被保持
UF0E2IM 寄存器	00H	值被保持

表 21-8. 特殊状态下的寄存器值 (2/2)

寄存器名称	CPU 复位 ($\overline{\text{RESET}}$) 之后	总线复位之后
UF0E3IM 寄存器	00H	值被保持
UF0E4IM 寄存器	00H	值被保持
UF0E7IM 寄存器	00H	值被保持
UF0E0R 寄存器	不确定 ^{注1}	值被保持
UF0E0L 寄存器	00H	值被保持
UF0E0ST 寄存器	00H	00H
UF0E0W 寄存器	不确定 ^{注1}	值被保持
UF0B01 寄存器	不确定 ^{注1}	值被保持
UF0B01L 寄存器	00H	值被保持
UF0B02 寄存器	不确定 ^{注1}	值被保持
UF0B02L 寄存器	00H	值被保持
UF0B11 寄存器	不确定 ^{注1}	值被保持
UF0B12 寄存器	不确定 ^{注1}	值被保持
UF0INT1 寄存器	不确定	值被保持
UF0INT2 寄存器	不确定	值被保持
UF0DSTL 寄存器	00H	00H
UF0E0SL 寄存器	00H	00H
UF0E1SL 寄存器	00H	00H
UF0E2SL 寄存器	00H	00H
UF0E3SL 寄存器	00H	00H
UF0E4SL 寄存器	00H	00H
UF0E7SL 寄存器	00H	00H
UF0ADRS 寄存器	00H	00H
UF0CNF 寄存器	00H	00H
UF0IF0 寄存器	00H	00H
UF0IF1 寄存器	00H	00H
UF0IF2 寄存器	00H	00H
UF0IF3 寄存器	00H	00H
UF0IF4 寄存器	00H	00H
UF0DSCL 寄存器	00H	值被保持
UF0DDn 寄存器 (n = 0 至 17)	注 2	注 2
UF0CIEn 寄存器(n = 0 至 255)	注 2	注 2

- 注
1. 因为当 $\overline{\text{RESET}}$ 信号有效时, 该寄存器的写入指针、计数器和读取指针清除为 0, 所以可通过 $\overline{\text{RESET}}$ 信号将该寄存器清为 0, 同理, 寄存器由 FIFO 控制时, 则由 UF0FICn 寄存器清除。
 2. 该寄存器虽不能被清为 0。但由于数据可通过 FW 写入该寄存器, 所以任何值都可以写入本寄存器 (然而, 写入之前务必将 UF0E0NA 寄存器的 EPONKA 位设为 1)。

21.9 FW处理

执行以下 FW 处理。

- 枚举处理期间，对 SET_CONFIGURATION、SET_INTERFACE、SET_FEATURE 以及 CLEAR_FEATURE 请求在设备侧进行设置。
- XXXXStandard、XXXXClass 和 XXXXVendor 请求的分析和处理不受自动处理影响。
- bulk-发送 OUT 令牌之后从接收缓存器中读取数据。
- 写入响应 bulk-发送 IN 令牌的返回数据。
- 写入响应中断-发送令牌的返回数据。

下表列出了 FW 所支持的请求。

表 21-9. FW 所支持的标准请求

请求	接收侧	处理/ 频率	说明
CLEAR_FEATURE	接口	自动 STALL 响应	尽管该请求为 bmRequestType 所保留，但因为不存在功能选择器的值，所以认为该请求不会达到接口。 当接收到该请求时，硬件自动做出 STALL 响应。
SET_FEATURE	接口	自动 STALL 响应	尽管该请求为 bmRequestType 所保留，但因为不存在功能选择器的值，所以认为该请求不会达到接口。 当接收到该请求时，硬件自动做出 STALL 响应。
GET_DESCRIPTOR	字符串	FW	返回字符串描述符。 当该请求被 SETUP 令牌接收时，硬件会为 FW 生成 CPUDEC 请求。FW 对来自 CPUDEC 中断请求的请求内容进行解码，并将要返回到主机的数据写入 UF0E0W 寄存器。
SET_DESCRIPTOR	设备	FW	重写设备描述符。 当该请求被 SETUP 令牌接收时，硬件会为 FW 生成 CPUDEC 请求。FW 对来自 CPUDEC 中断请求的请求内容解码，并将下一个控制传输 (OUT) 的数据写入到 UF0DDn 寄存器 (n = 0 至 17)。
SET_DESCRIPTOR	配置	FW	重写配置描述符。 当该请求被 SETUP 令牌接收时，硬件会为 FW 生成 CPUDEC 请求。FW 对来自 CPUDEC 中断请求的请求内容解码，并将下一个控制传输 (OUT) 的数据写入到 UF0CIEn 寄存器 (n = 0 至 255)。
SET_DESCRIPTOR	字符串	FW	重写字符串描述符。 当该请求被 SETUP 令牌接收时，硬件会为 FW 生成 CPUDEC 请求。FW 对来自 CPUDEC 中断请求的请求内容解码，并为下一个控制传输 (OUT) 载入数据。
其它	NA	FW	当该请求被 SETUP 令牌接收时，硬件会为 FW 生成 CPUDEC 请求。FW 对来自 CPUDEC 中断请求的请求内容解码，并执行必要的处理。

21.9.1 初始化处理

初始化处理以下面两种方式执行。

- 请求数据寄存器的初始化
- 中断的设置

当初始化请求数据寄存器时，将会写数据至会自动返回值的 GET_XXXX 请求，并将一个端点分配给接口。 在中断设置中，可通过 UF0IMn 寄存器 (n = 0 至 4) 屏蔽不必进行检查的中断源。

下面的流程图说明上述的处理。

图 21-14. 初始化请求数据寄存器

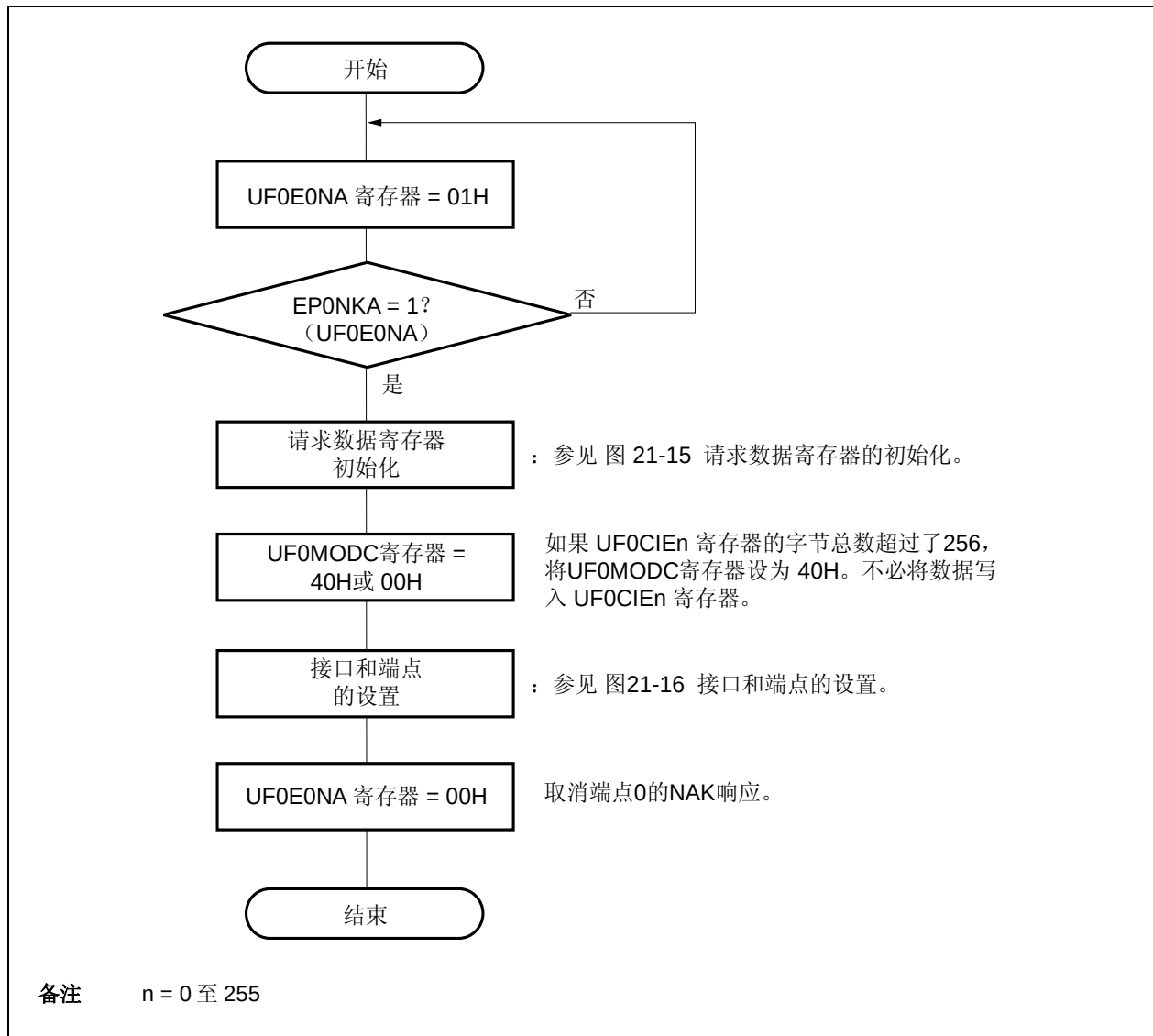


图 21-15. 请求数据寄存器的初始化设置

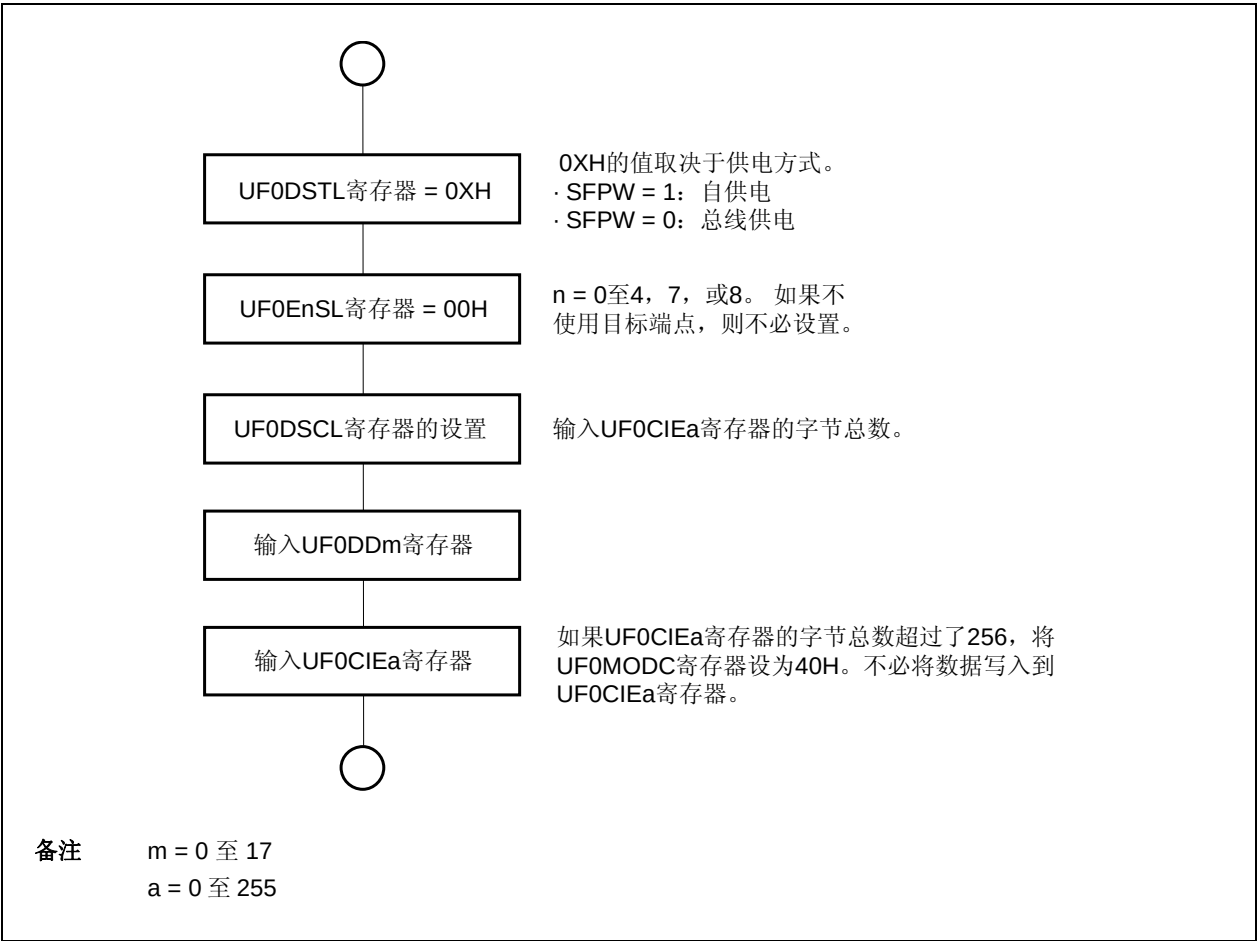


图 21-16. 接口和端点的设置

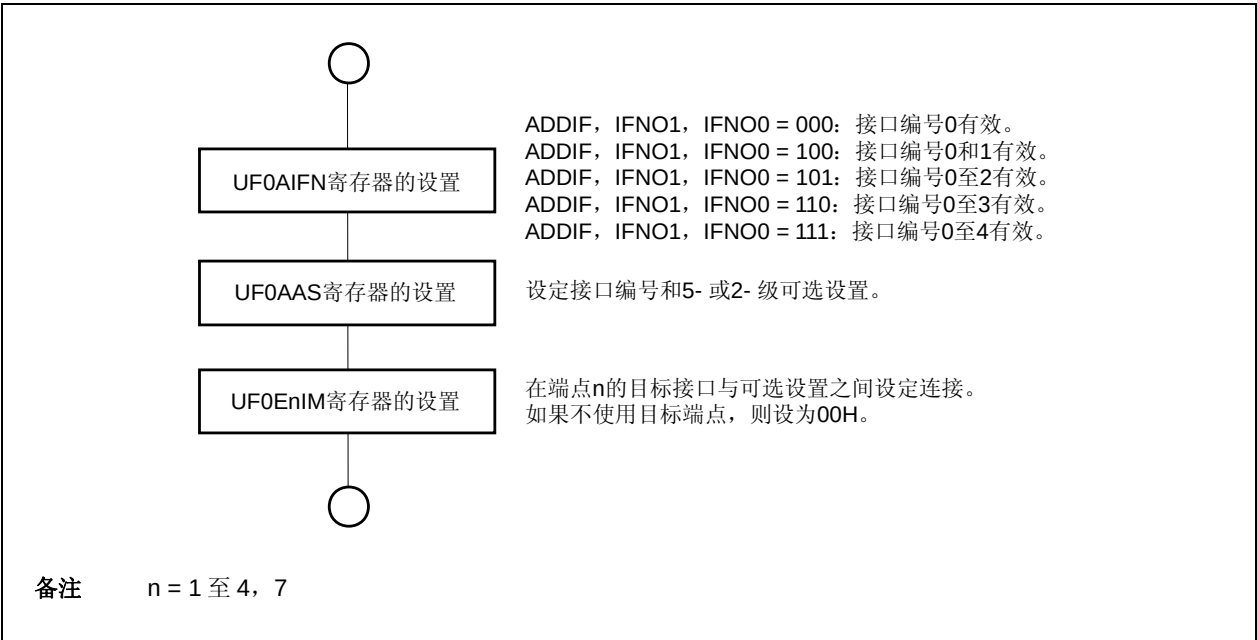
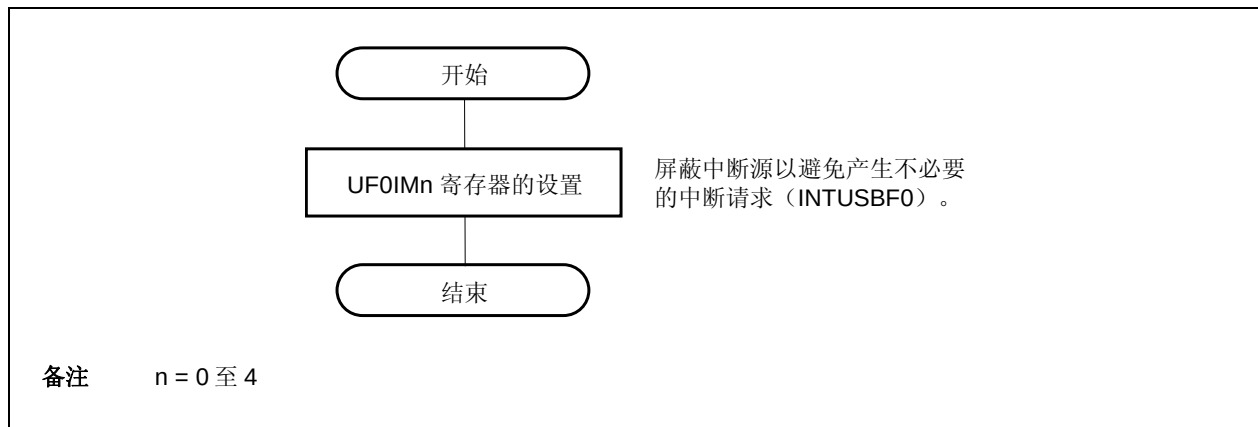


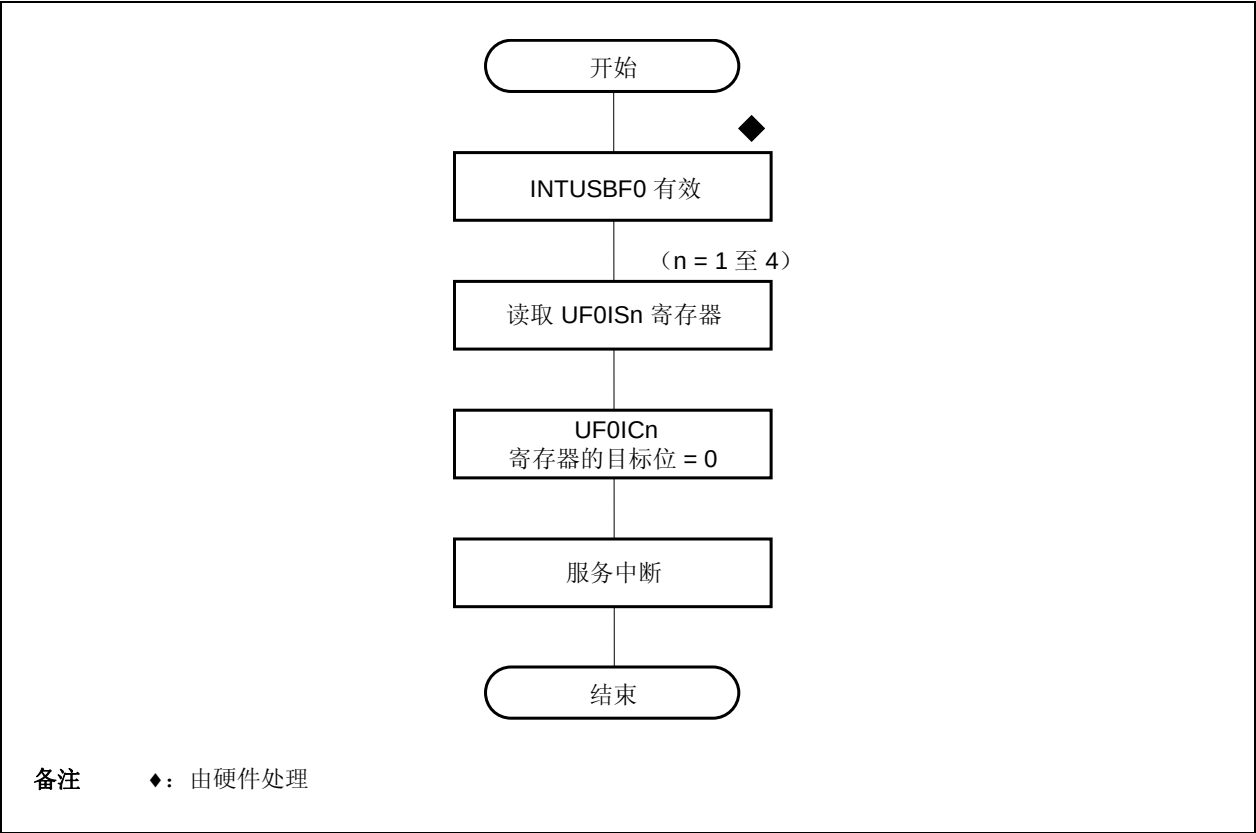
图 21-17. 中断设置



21.9.2 中断服务

下面的流程图说明如何对中断进行服务。

图 21-18. 中断服务



当满足给定条件时，硬件会自动将 UF0ISn 寄存器（n = 1 至 4）的以下各位清除。

- UF0IS1 寄存器的 E0INDT、E0ODT、SUCES、STG 和 CPUDEC 位
- UF0IS2 寄存器的 BKI2DT、BKI1DT、IT2DT 和 IT1DT 位
- UF0IS3 寄存器的 BKO2FL、BKO2DT、BKO1FL 和 BKO1DT 位

因为通过 UF0ICn 寄存器清除中断源的优先级低于硬件设置中断源的优先级，所以中断源可能不会按照时序清除（n = 0 至 4）。

21.9.3 USB主处理

USB 主处理包括处理 USB 事务。 处理的事务类型如下。

- 控制传输的全自动处理请求
- 控制传输的自动处理请求
(SET_CONFIGURATION, SET_INTERFACE, SET_FEATURE, CLEAR_FEATURE)
- 控制传输的 CPUDEC 请求
- Bulk 传输 (IN) 的处理
- Bulk 传输 (OUT) 的处理
- 中断传输 (IN) 的处理

端点 n 的处理包括数据传输的写入或读取。 下面显示的流程图是关于 PIO 的。

(1) 控制传输的全自动处理请求

因为控制传输的全自动处理请求由硬件执行，所以它不能被 FW 引用。 因此，FW 不必为该请求执行任何处理。

(2) 控制传输的自动处理请求

(SET_CONFIGURATION, SET_INTERFACE, SET_FEATURE, CLEAR_FEATURE)

诸如 SET_CONFIGURATION、SET_INTERFACE、SET_FEATURE 和 CLEAR_FEATURE 请求，对控制传输的自动处理请求寄存器的写入处理由硬件自动执行，但是会产生一个用于设备端识别的中断请求。 如果不存在要执行的特殊处理则忽略该处理。

流程图如下所示。

图 21-19. 控制传输的自动处理请求

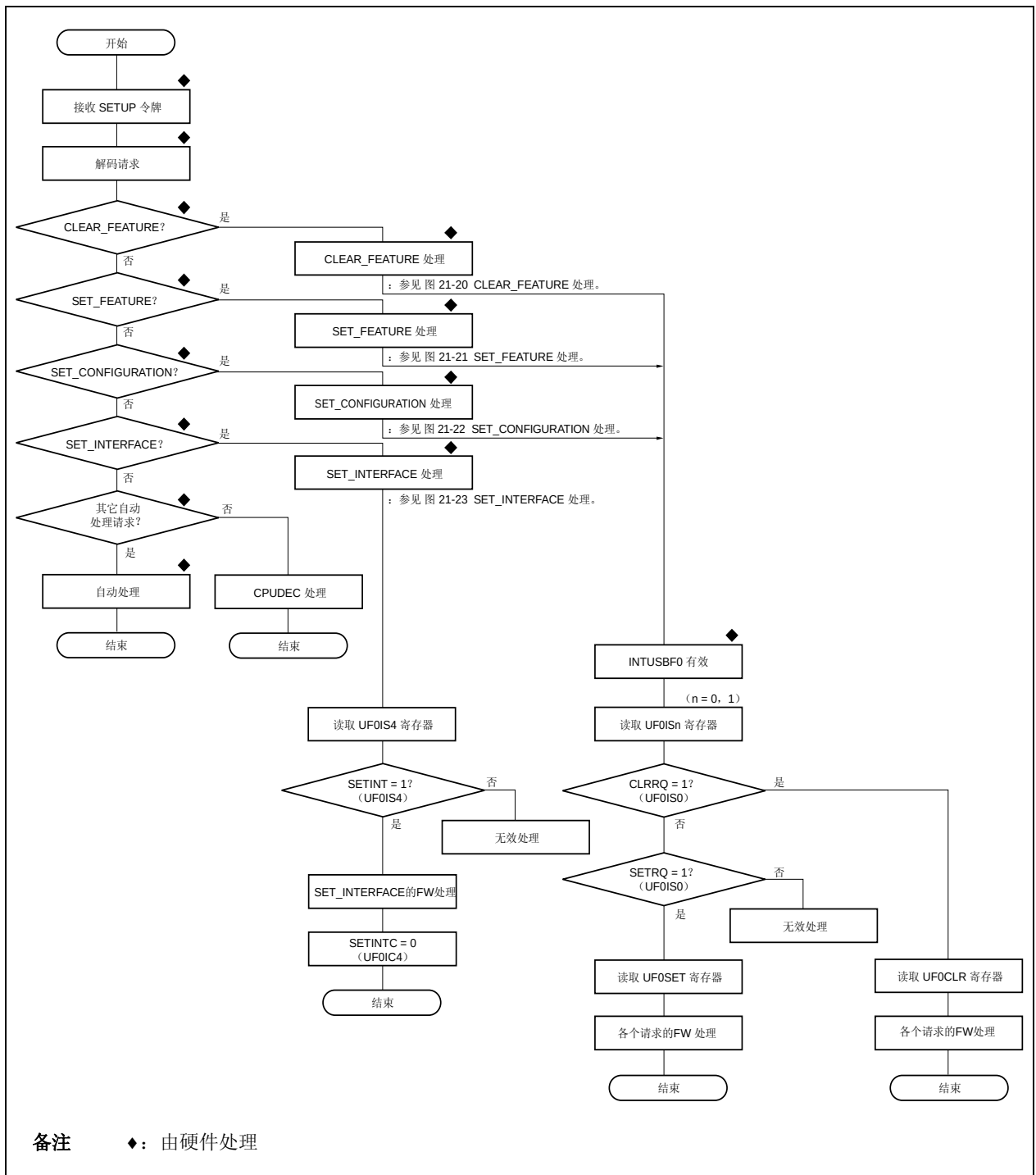


图 21-20. CLEAR_FEATURE 处理

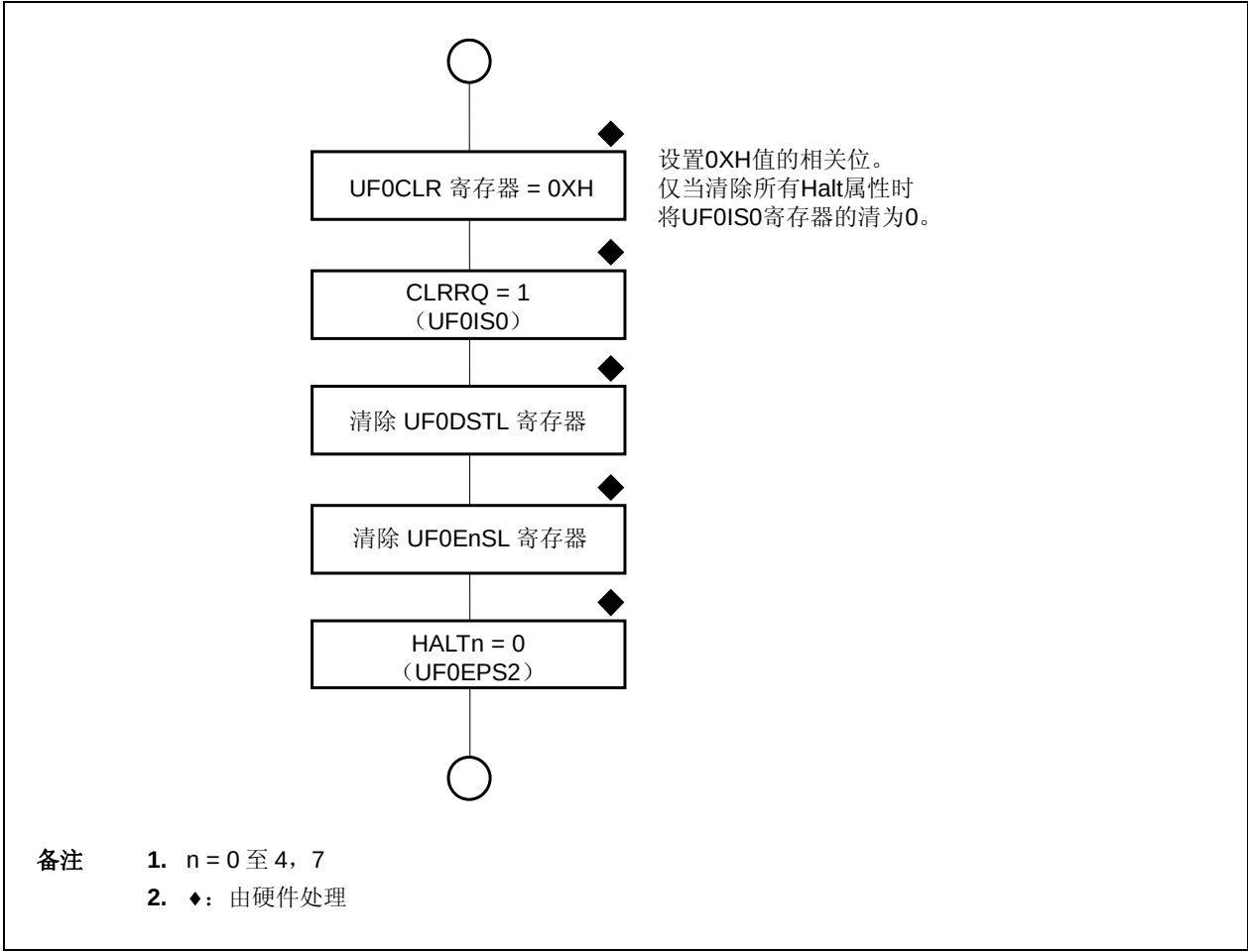


图 21-21. SET_FEATURE 处理

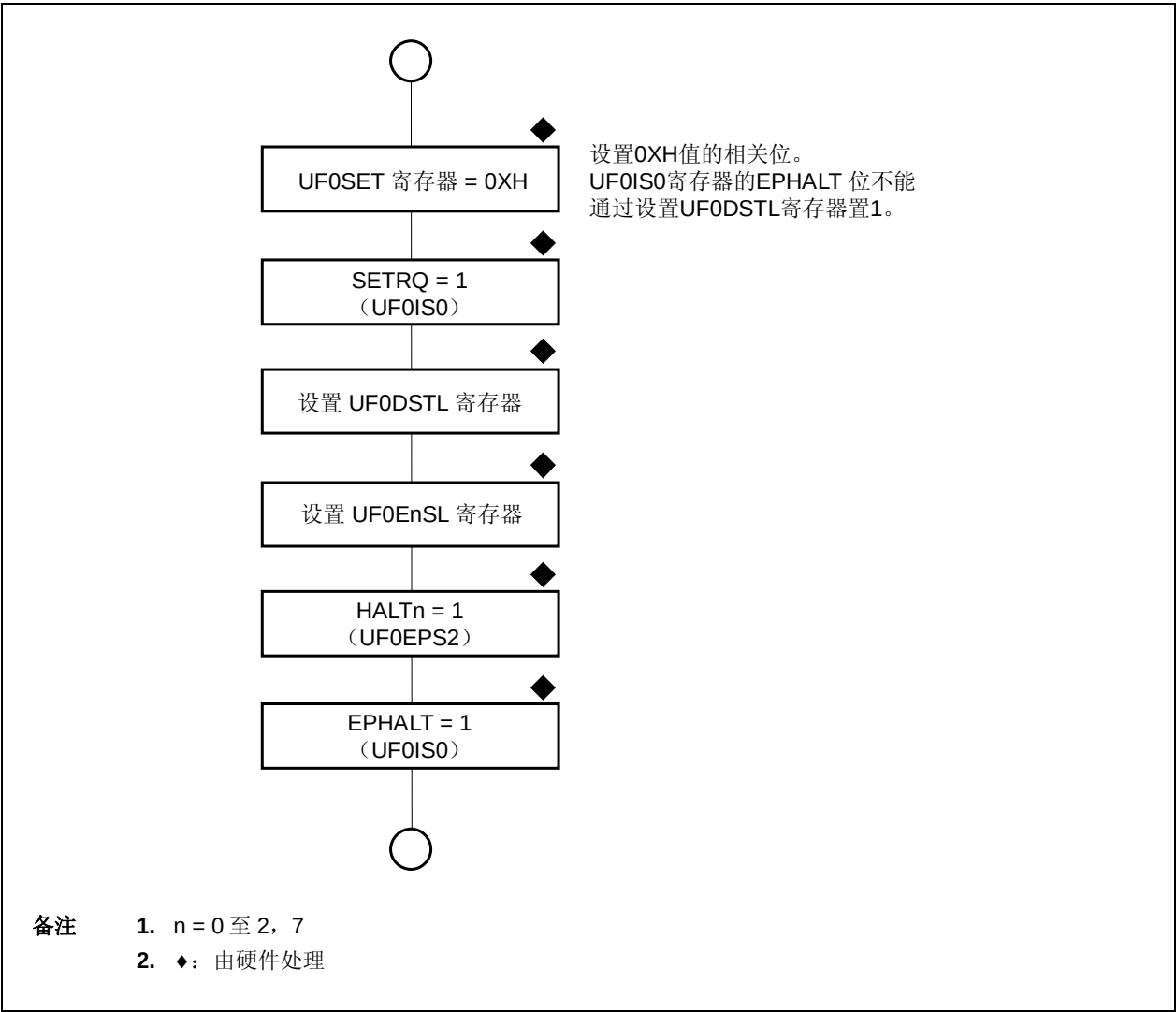


图 21-22. SET_CONFIGURATION 处理

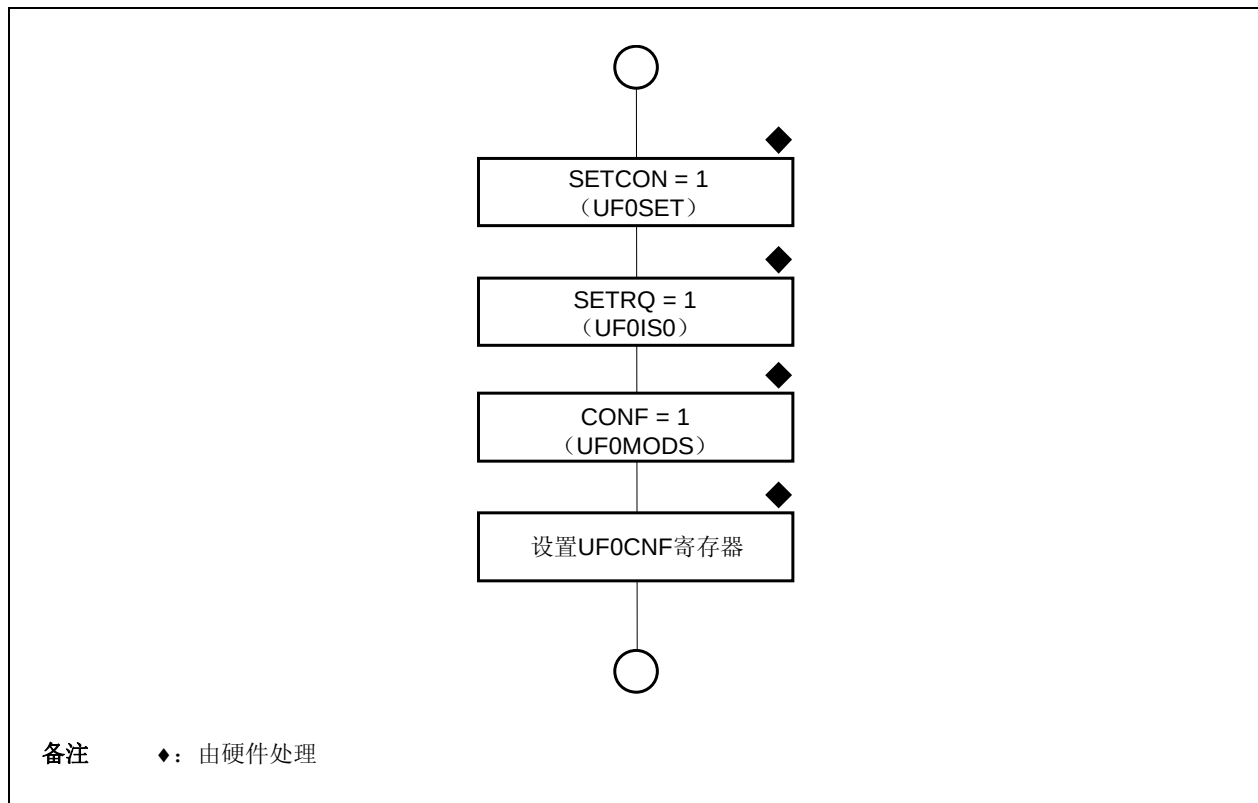
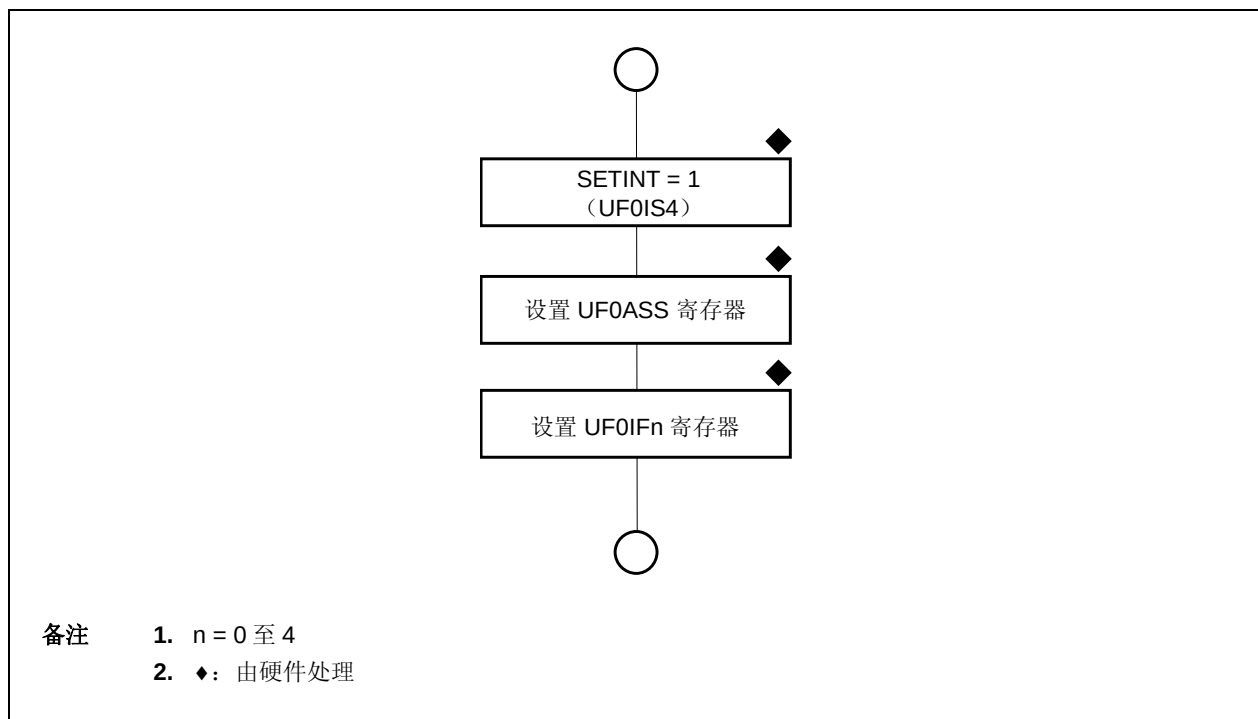


图 21-23. SET_INTERFACE 处理



(3) 控制传输的 CPUDEC 请求

CPUDEC 请求可分为三种处理类型：控制传输（写入），控制传输（读取）和控制传输（无数据）。控制传输（写入）表示在数据阶段使用 OUT 处理的请求（例如，SET_DESCRIPTOR），而控制传输（读取）表示在数据阶段使用 IN 处理的请求（例如，GET_DESCRIPTOR）。控制传输（无数据）表示一个没有数据阶段的请求（例如，SET_CONFIGURATION）。流程图如下所示。

图 21-24. 控制传输的 CPUDEC 请求 (1/12)

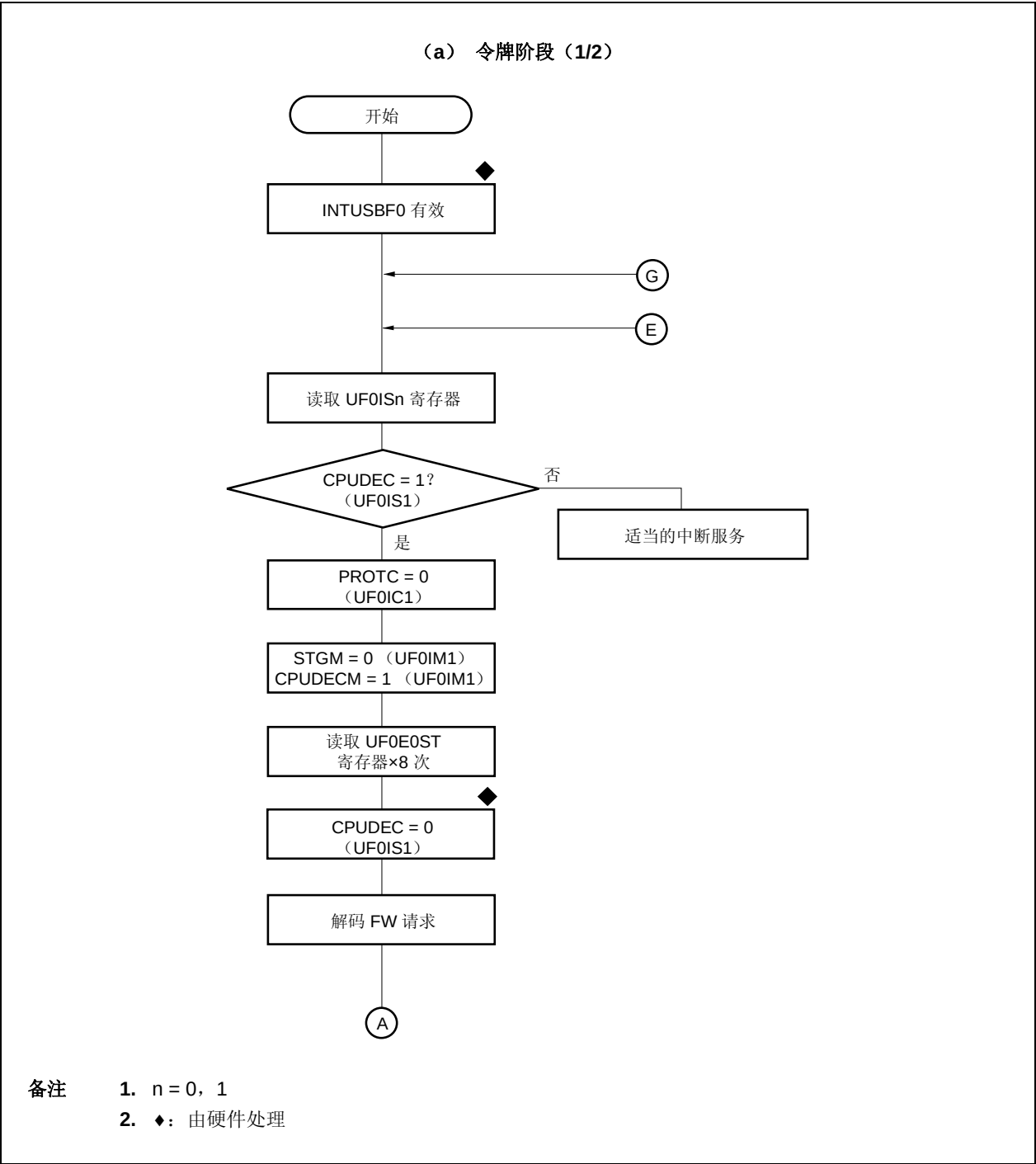


图 21-24. 控制传输的 CPUDEC 请求 (2/12)

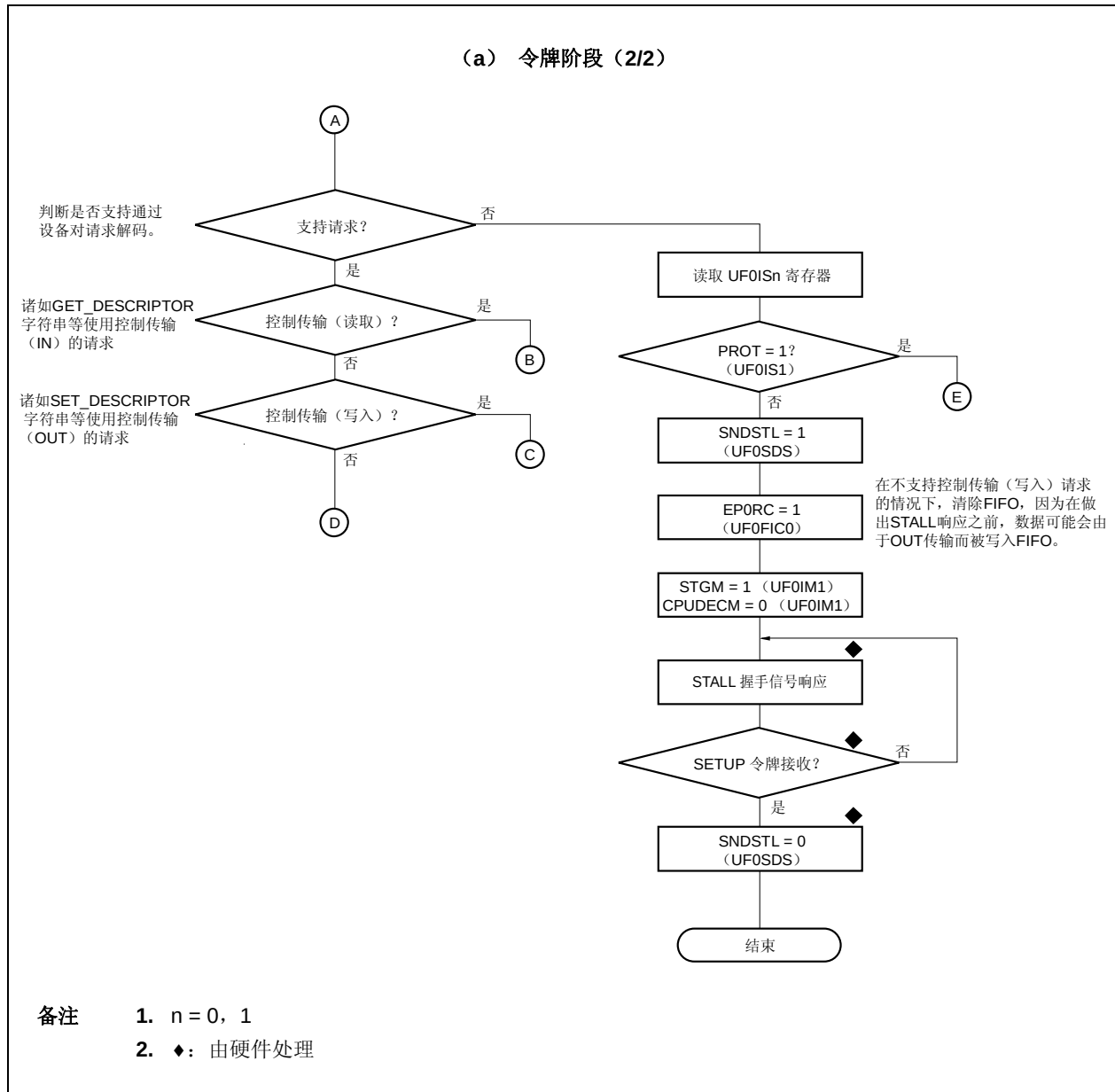


图 21-24. 控制传输的 CPUDEC 请求 (3/12)

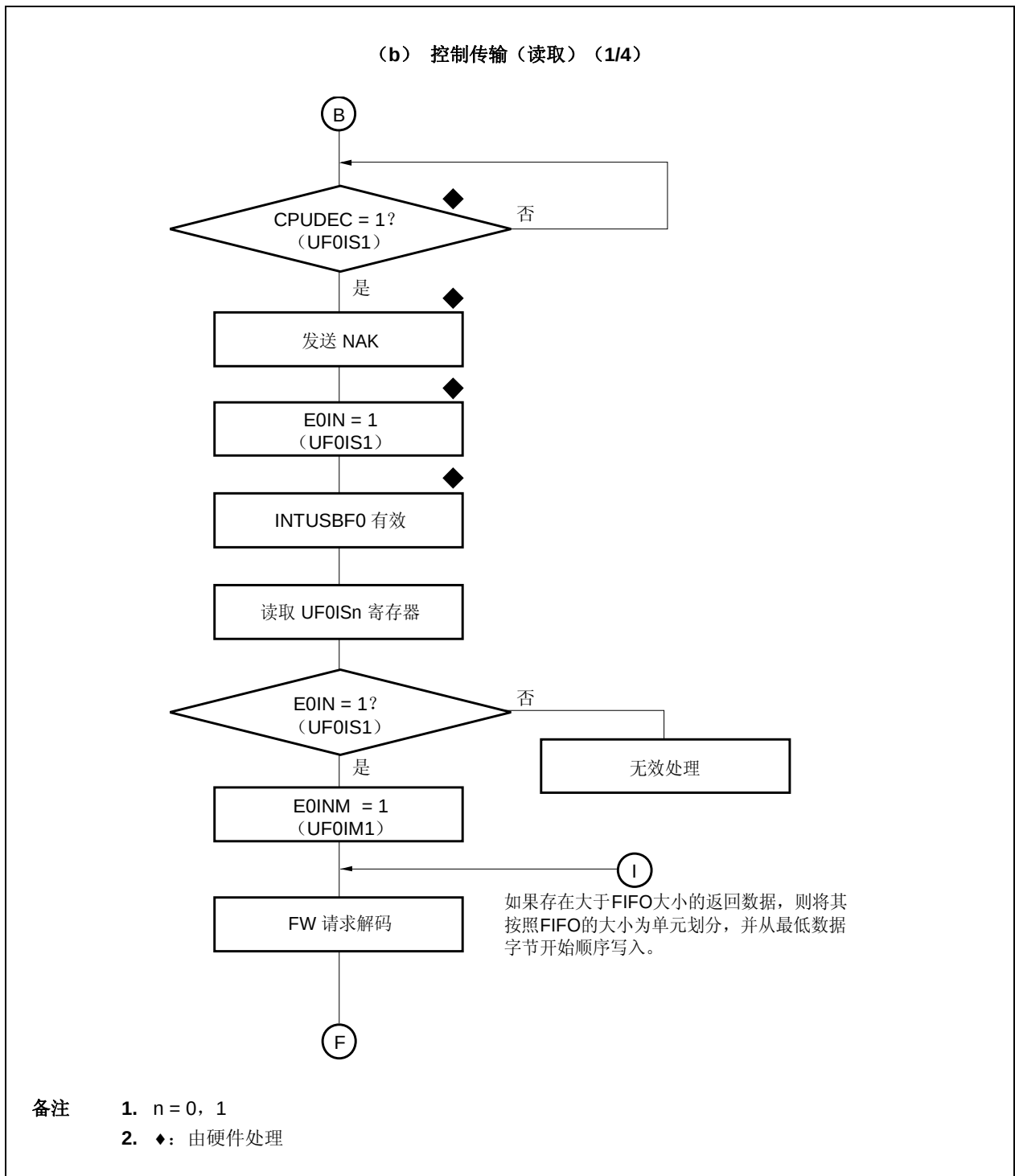


图 21-24. 控制传输的 CPUDEC 请求 (4/12)

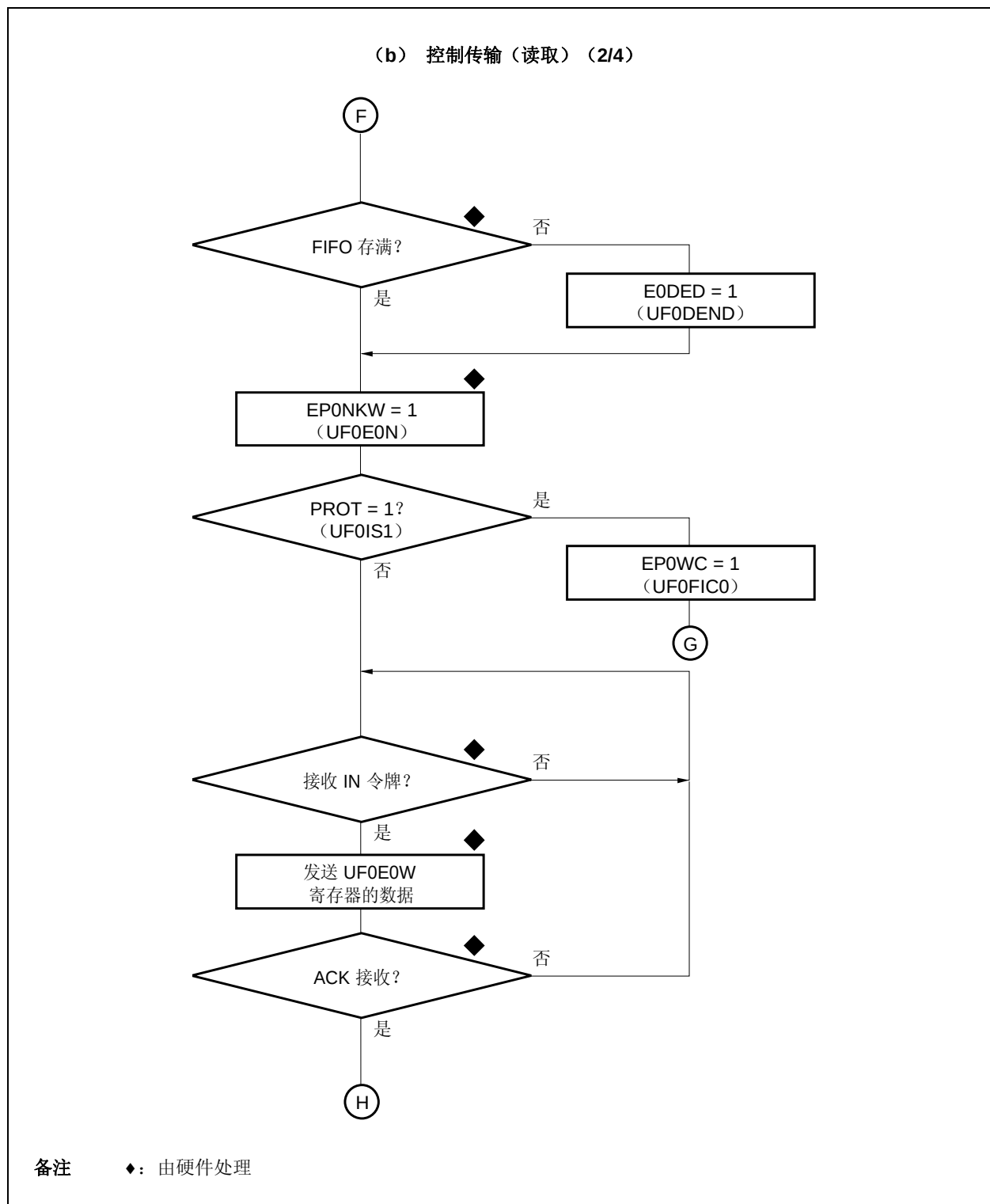


图 21-24. 控制传输的 CPUDEC 请求 (5/12)

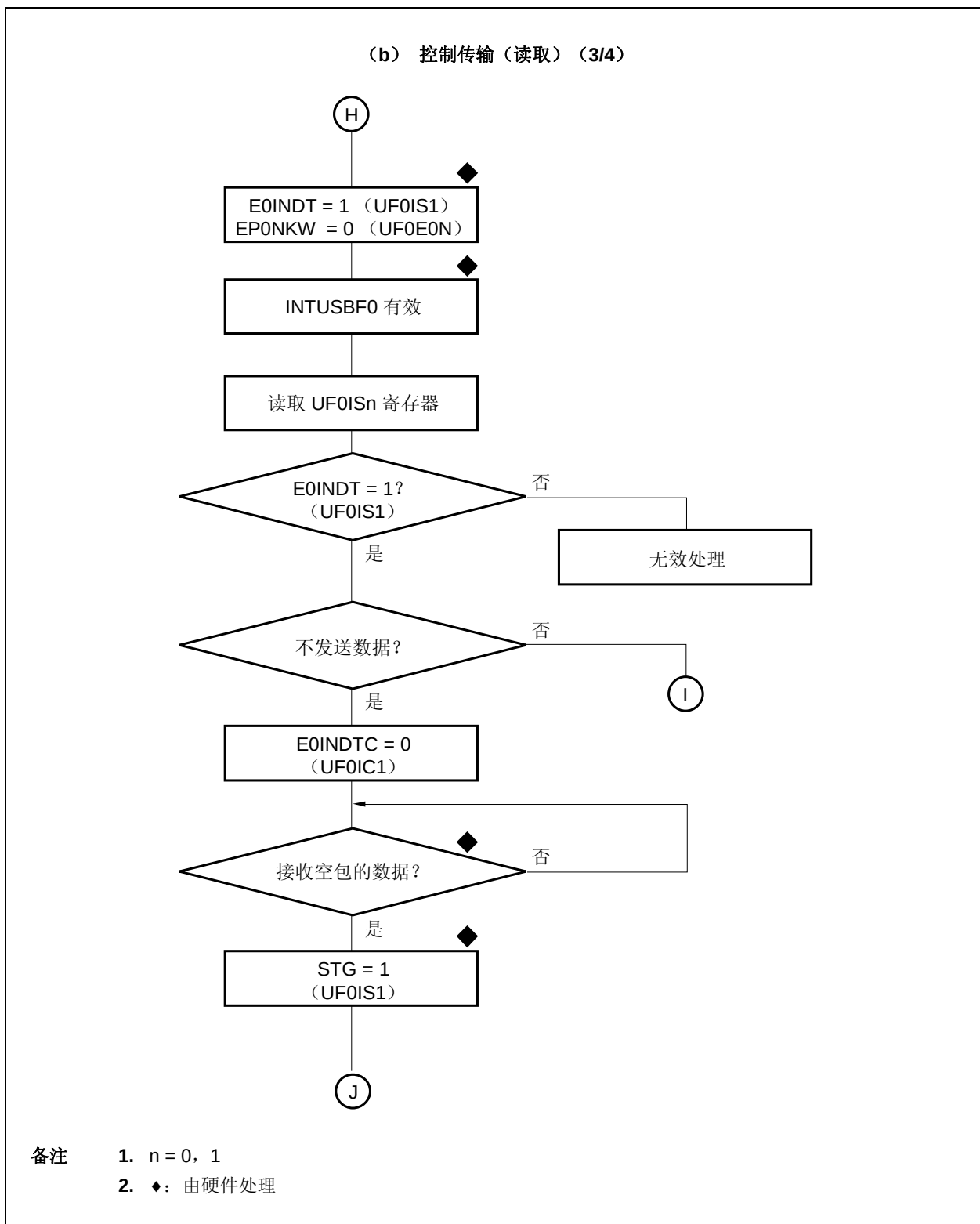
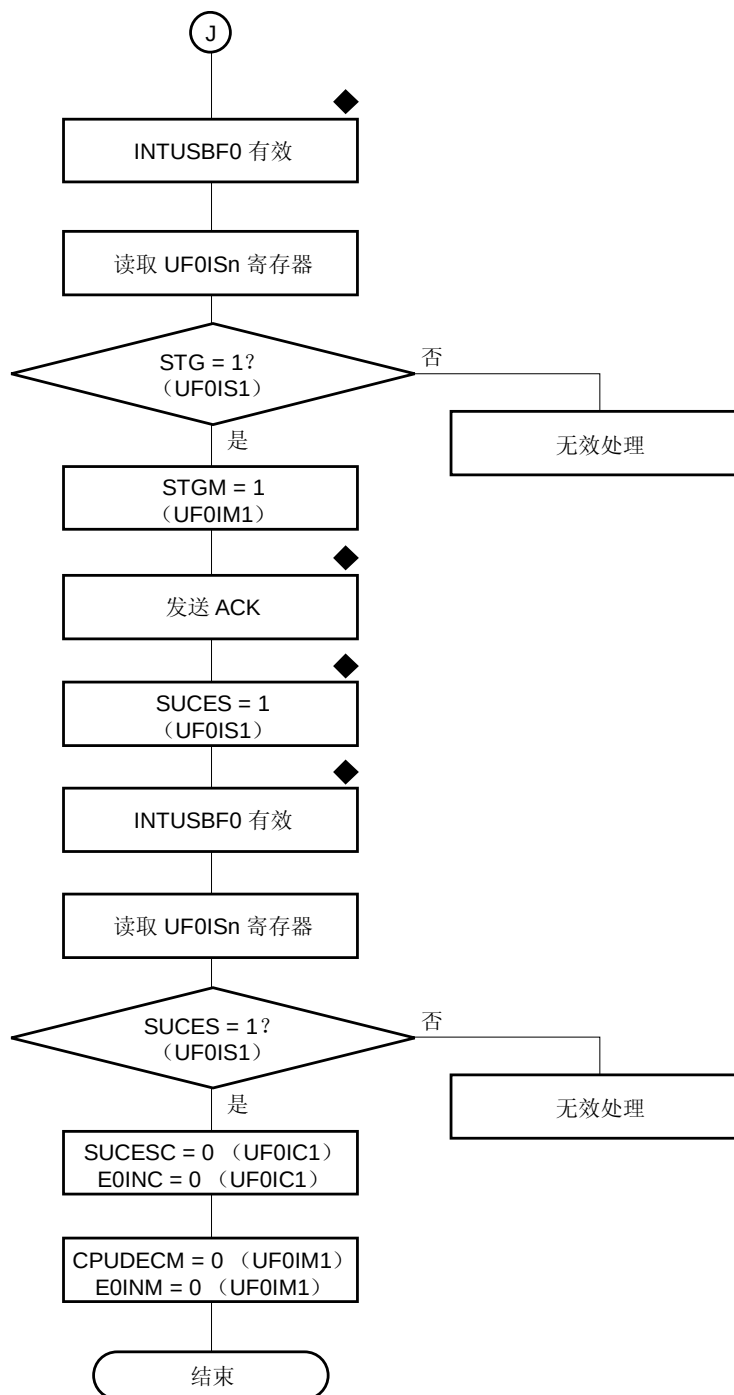


图 21-24. 控制传输的 CPUDEC 请求 (6/12)

(b) 控制传输 (读取) (4/4)



备注

1. n = 0, 1
2. ♦: 由硬件处理

图 21-24. 控制传输的 CPUDEC 请求 (7/12)

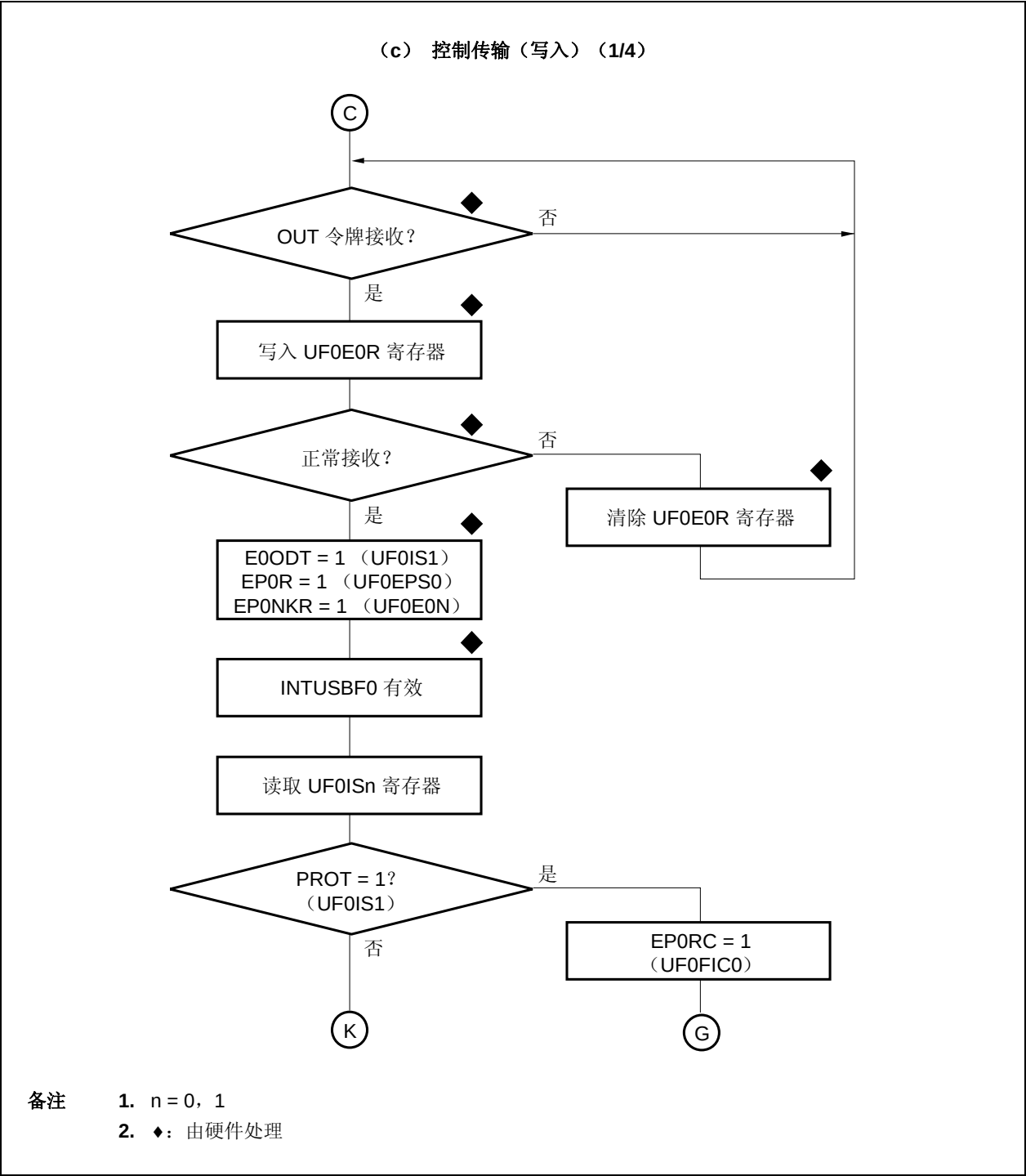


图 21-24. 控制传输的 CPUDEC 请求 (8/12)

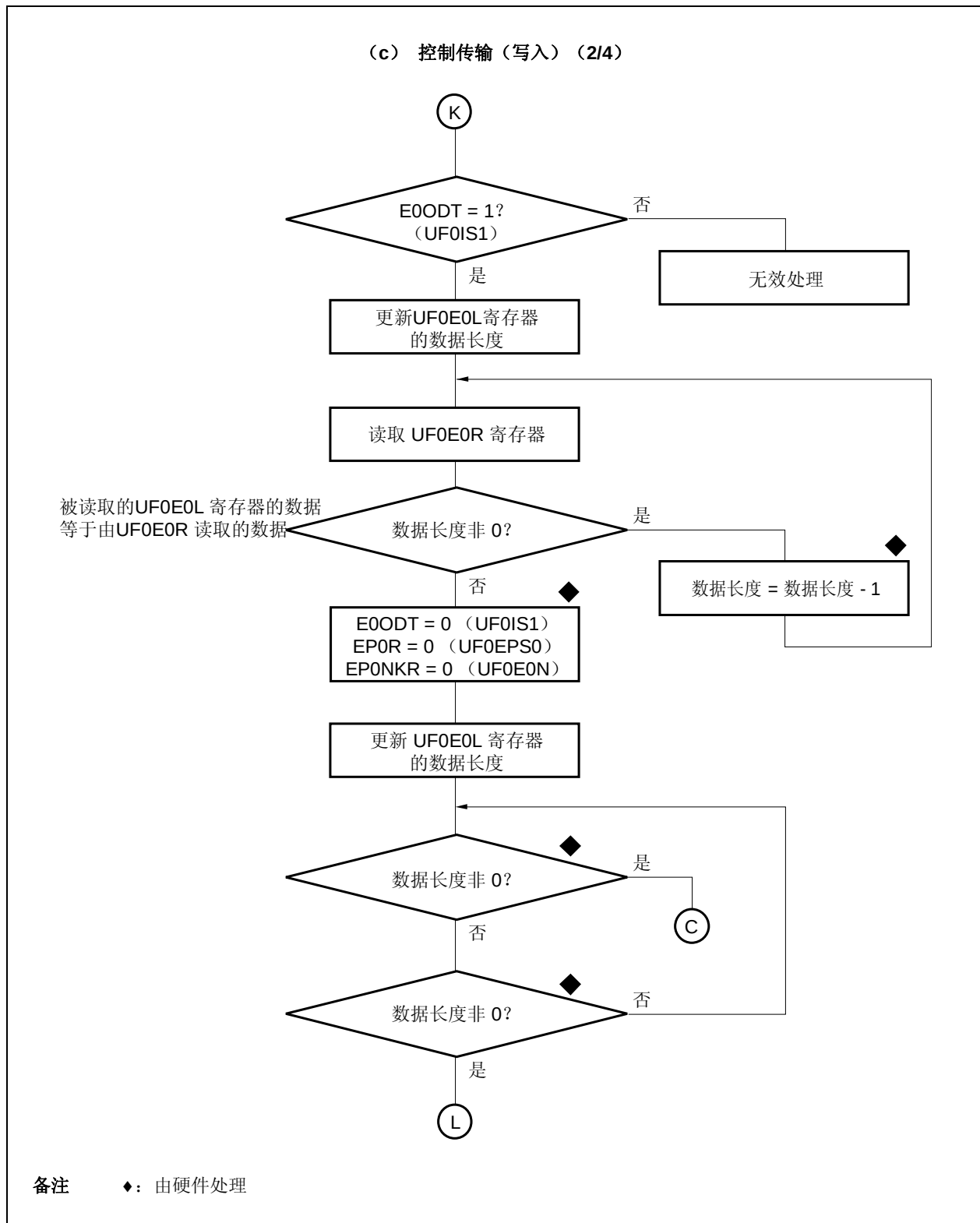


图 21-24. 控制传输的 CPUDEC 请求 (9/12)

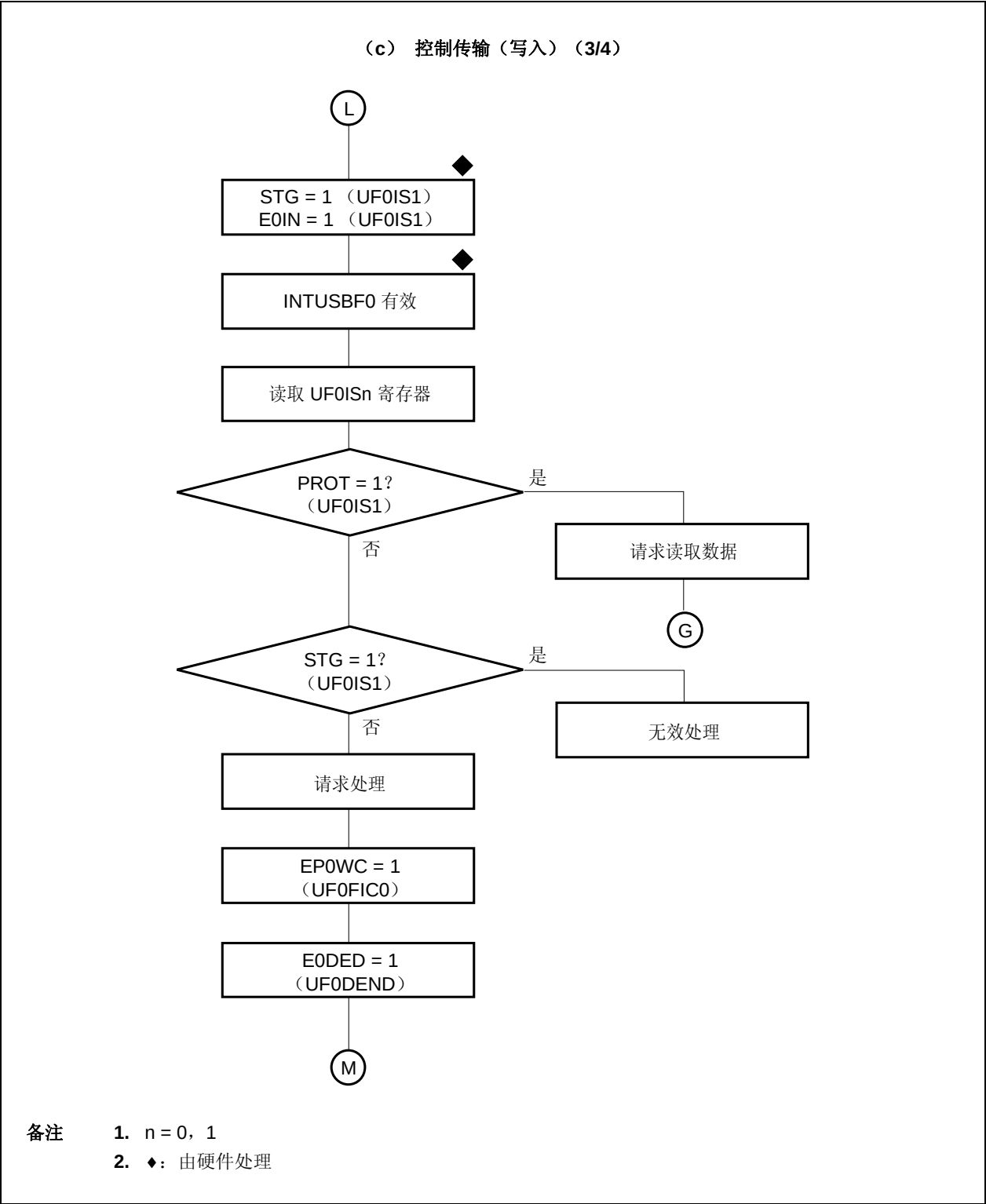
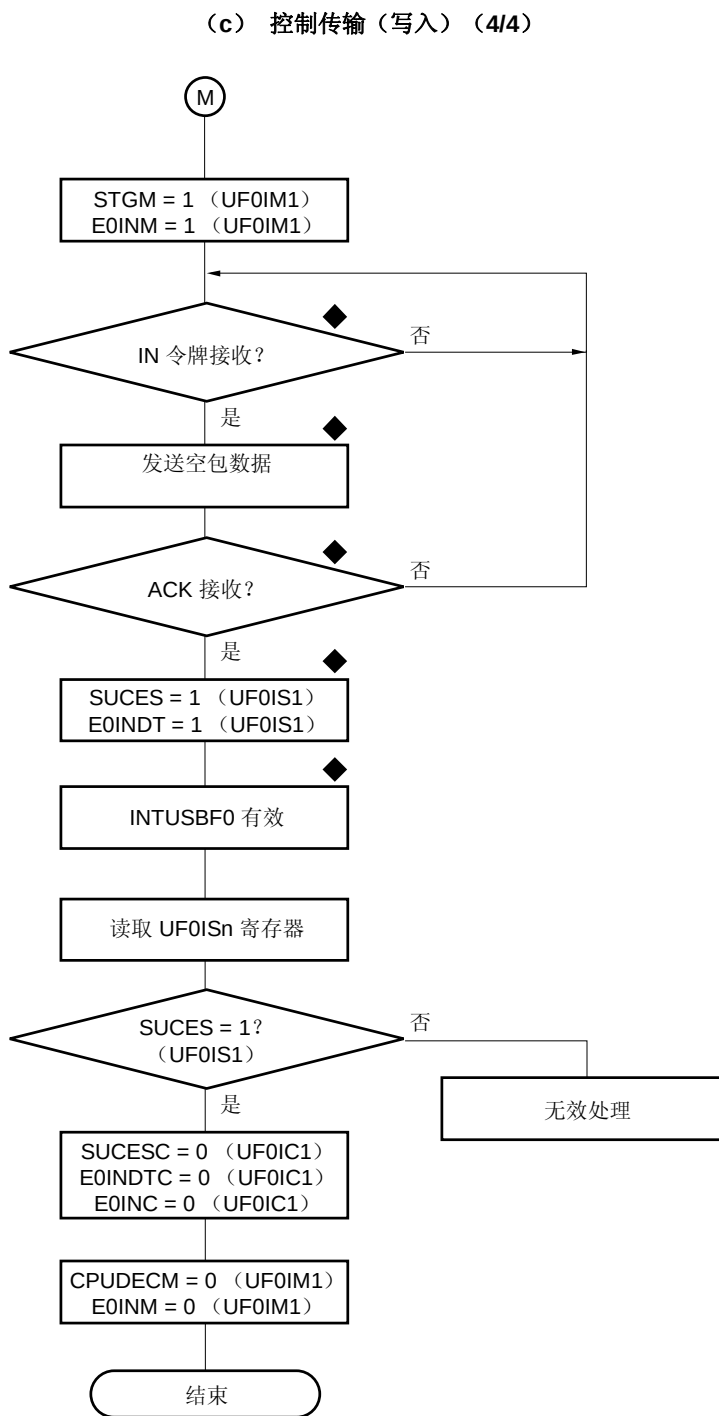


图 21-24. 控制传输的 CPUDEC 请求 (10/12)



备注

1. n = 0, 1
2. ♦: 由硬件处理

图 21-24. 控制传输的 CPUDEC 请求 (11/12)

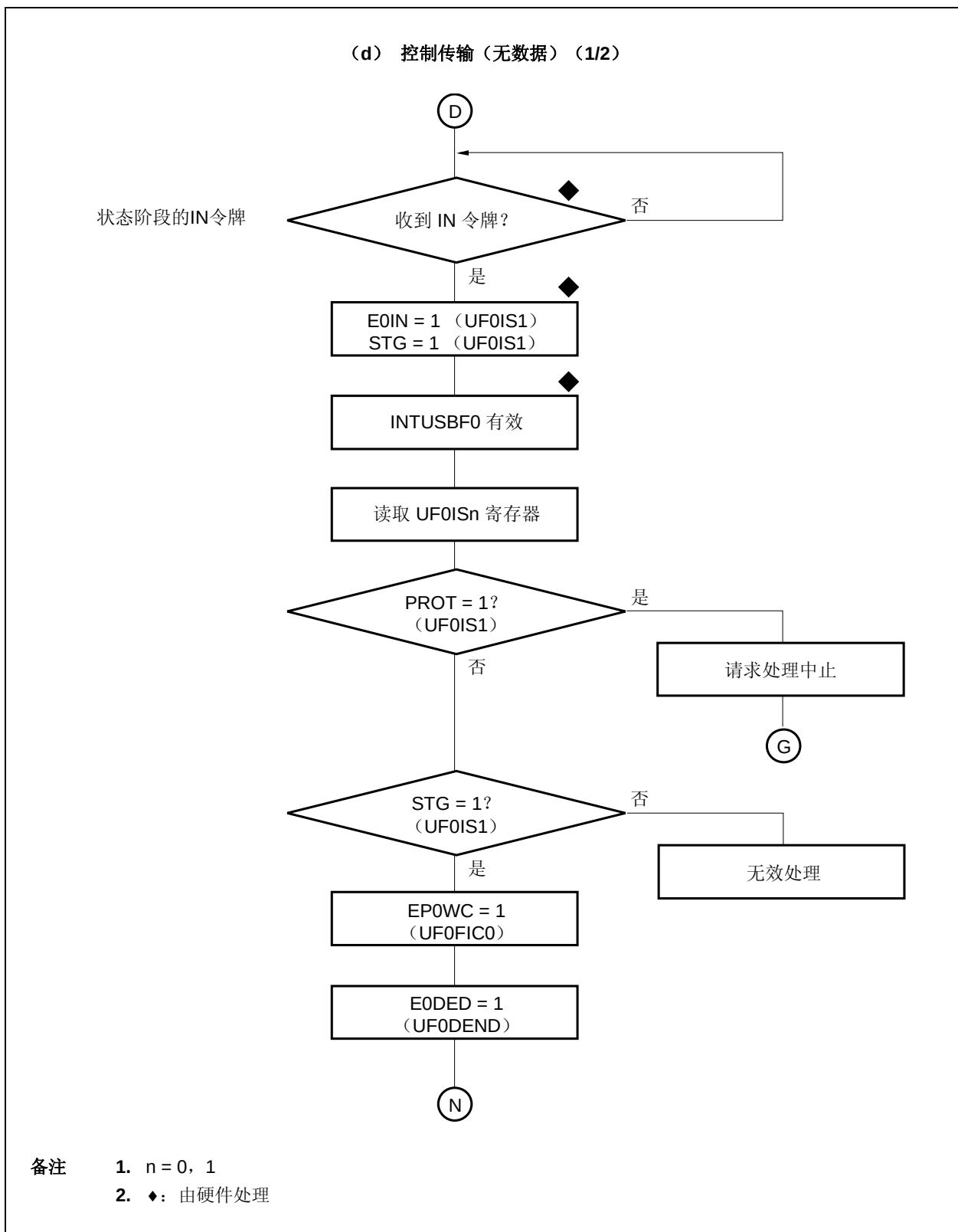
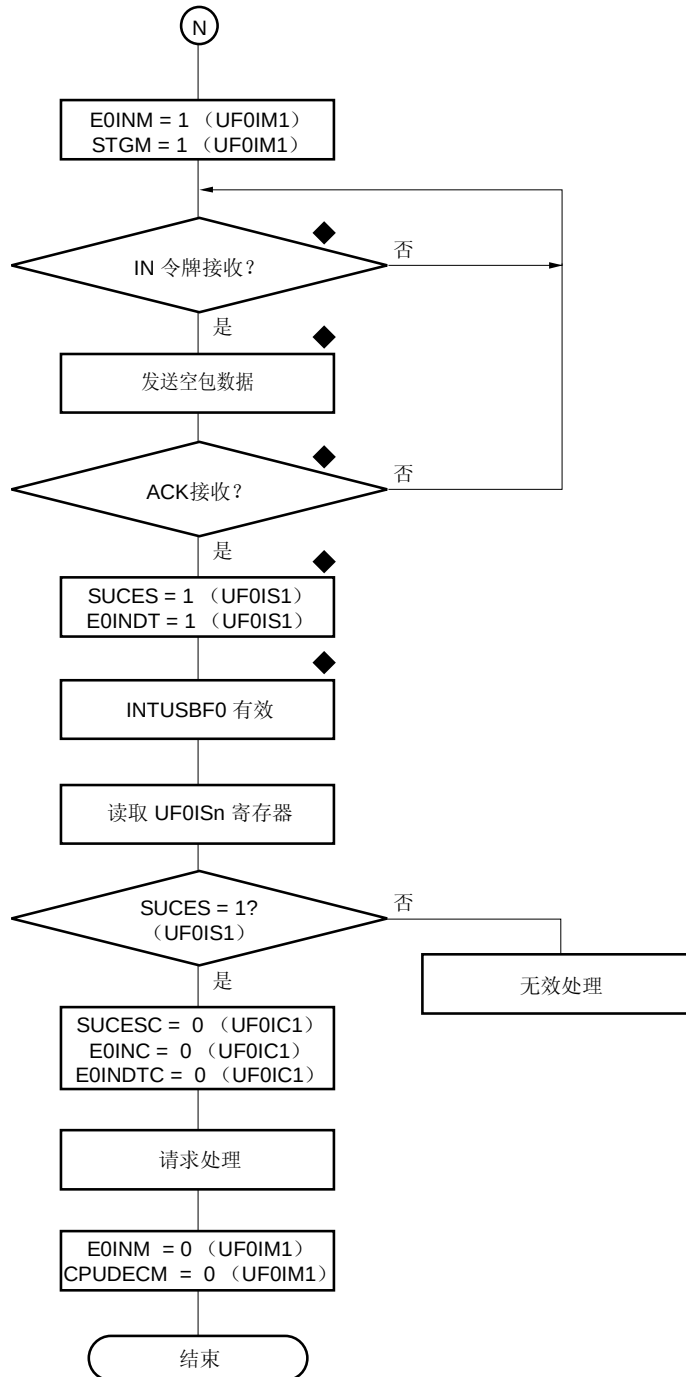


图 21-24. 控制传输的 CPUDEC 请求 (12/12)

(d) 控制传输 (无数据) (2/2)



备注

1. n = 0, 1
2. ◆: 由硬件处理

(4) Bulk 传输 (IN) 的处理

Bulk 传输 (IN) 被分配至端点 1 和端点 3。下面所示的流程图说明了如何控制端点 1。可用同样的顺序对端点 3 进行控制。因此,要使用该流程图作为端点 3 的控制流程,需在流程图将端点 1 的位名称读为端点 3 的位名称。

图 21-25. Bulk 传输 (IN) 的处理 (端点 1)

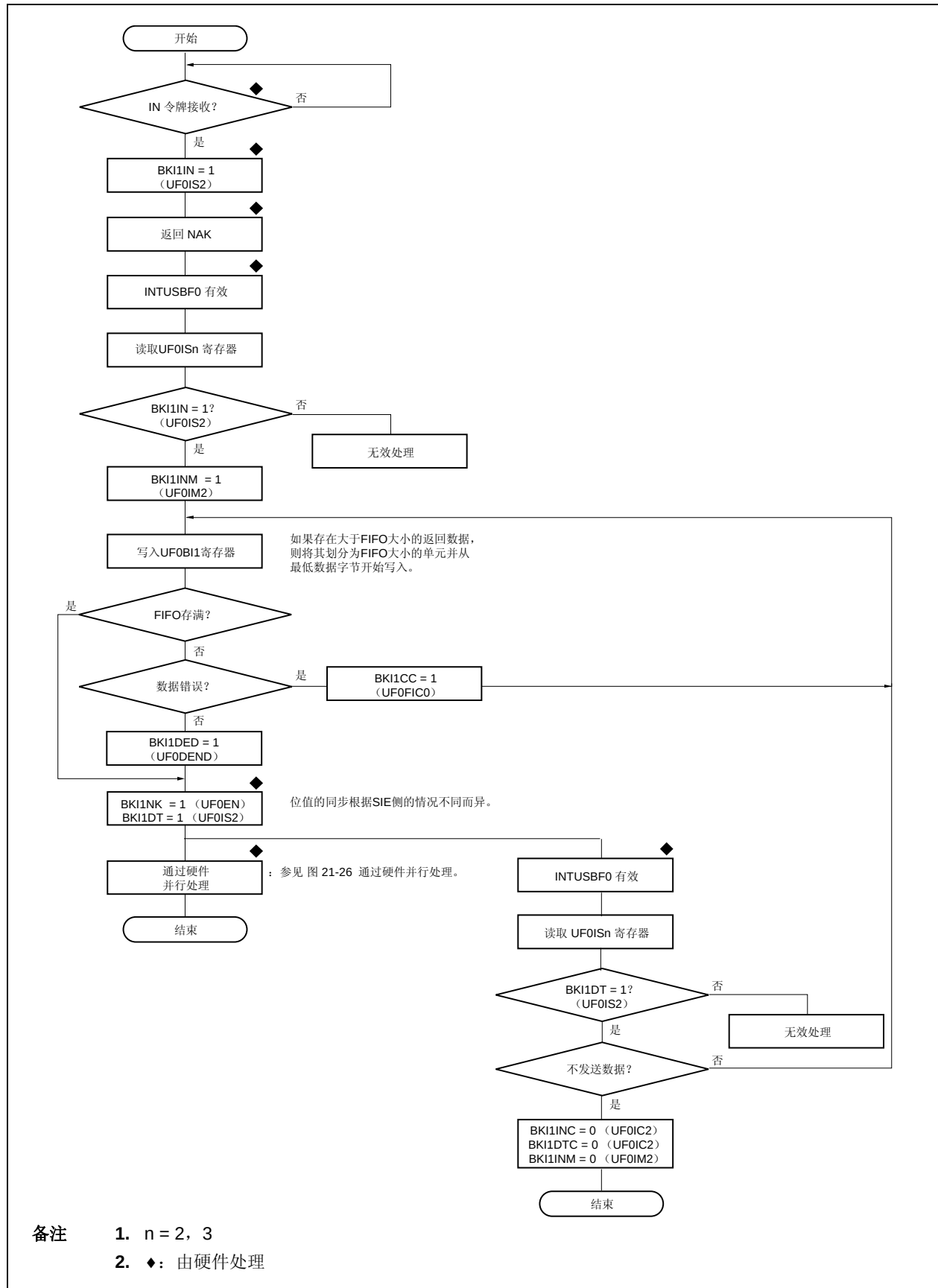
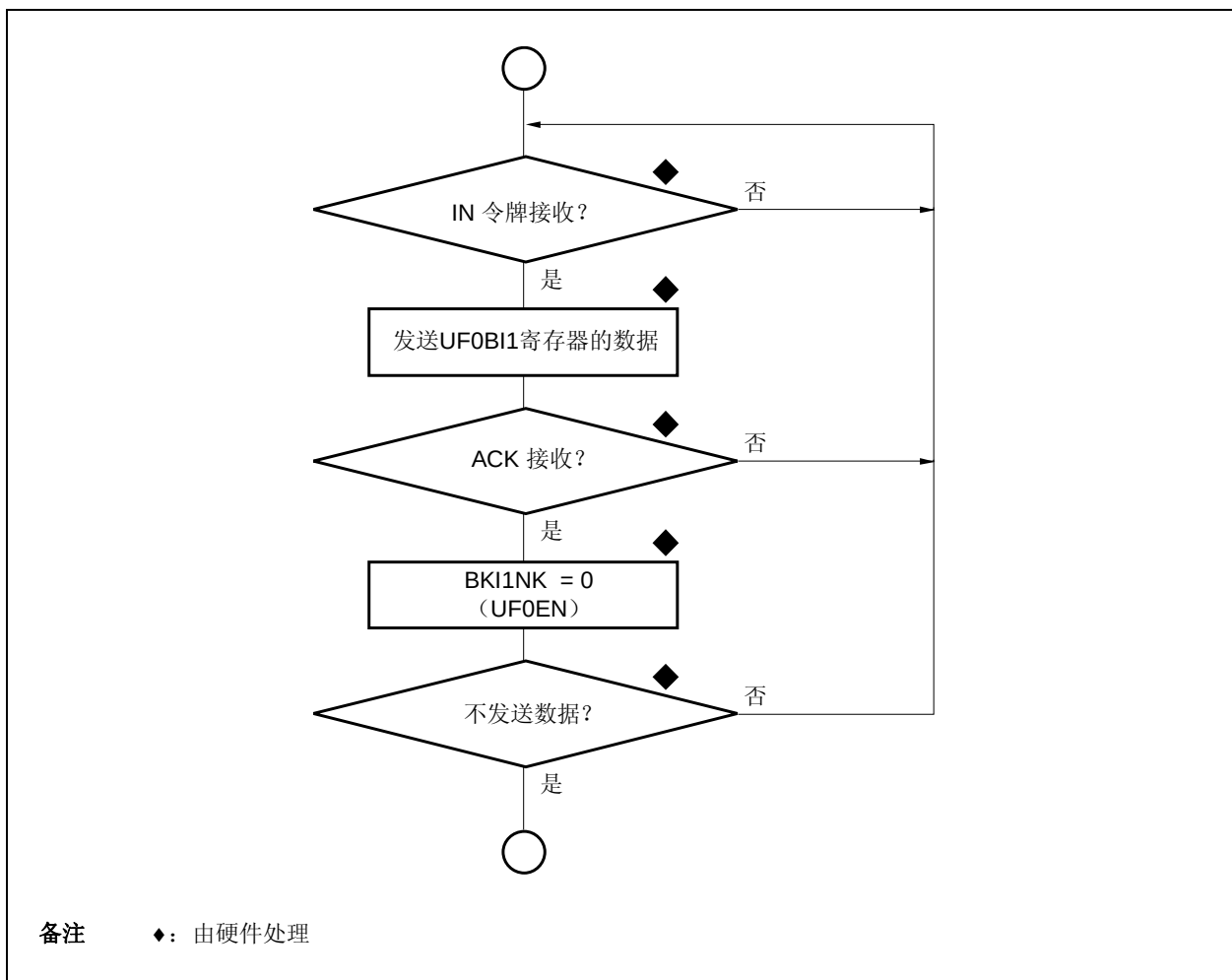


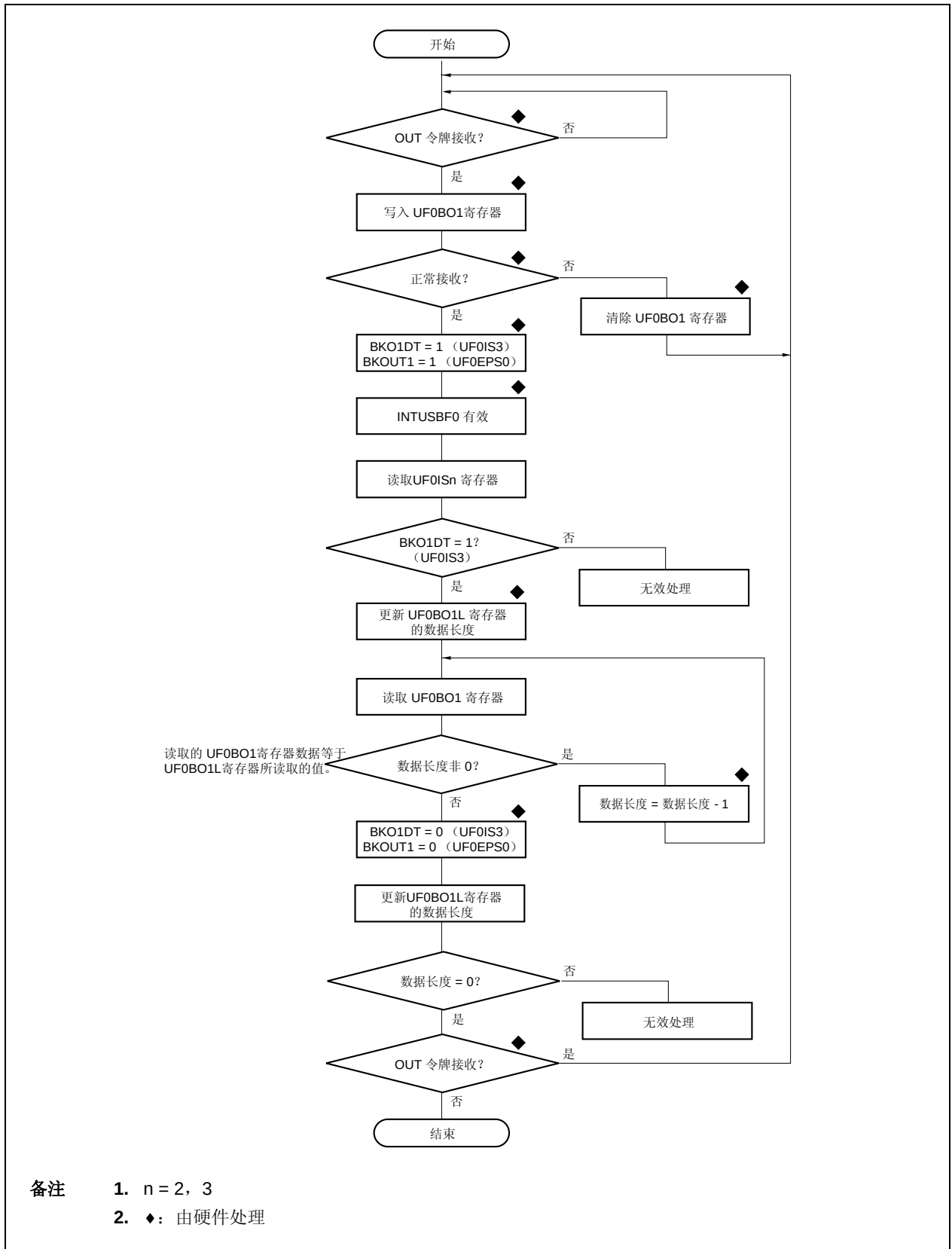
图 21-26. 由硬件并行处理



(5) Bulk 传输 (OUT) 的处理

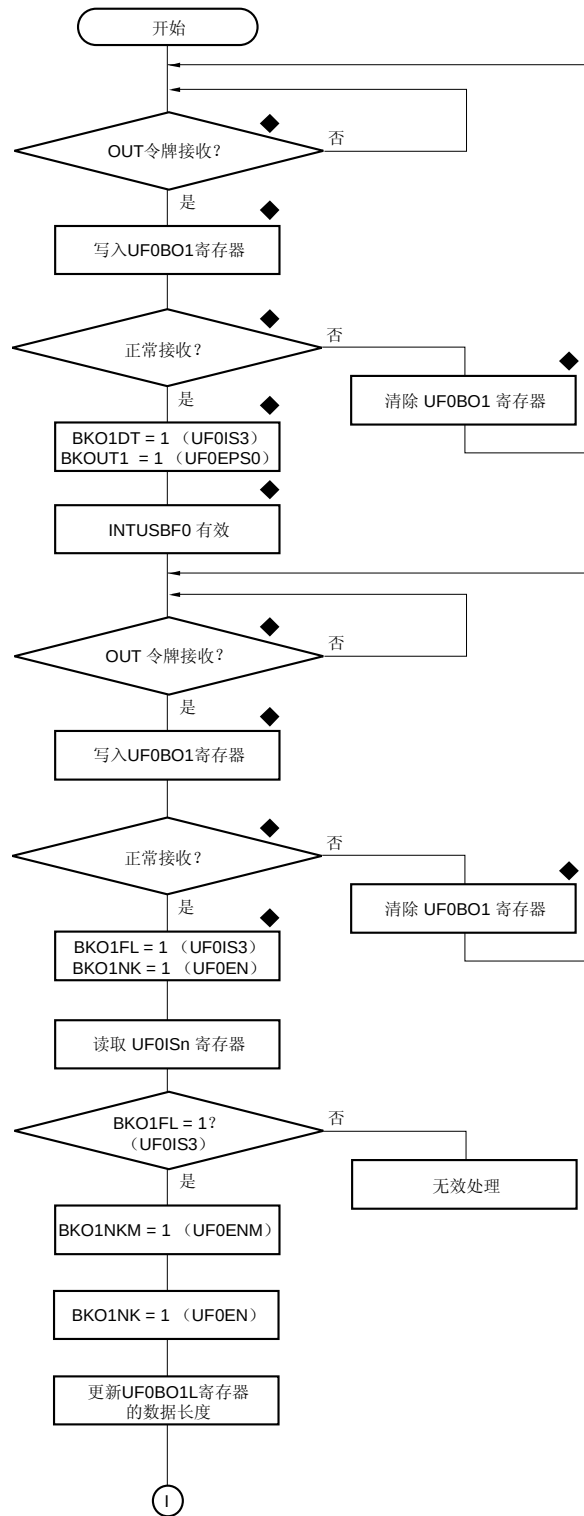
Bulk 传输 (OUT) 被分配至端点 2 和端点 4。下面所示的流程图说明了如何控制端点 2。可用同样的顺序对端点 4 进行控制。因此,要使用该流程图作为端点 4 的控制流程,需在流程图将端点 2 的位名称读作端点 4 的位名称。

图 21-27. Bulk 传输 (OUT) 的正常处理 (端点 2)



Bulk 传输 (OUT) 期间, 由主机发送的数据可能多于系统的预期。 V850ES/JG3-H 和 V850ES/JH3-H 单片机 bulk 传输 (OUT) 的端点 2 和端点 4 由两个 64-字节的缓存器组成, 因此能尽可能的减少 NAK 响应, 并且, 即使在访问总线期间 USB 总线的传输速率增加时也可以从 CPU 侧读取数据。 因此, 如果主机发送多于系统预期的数据, 最差的情况下也可自动接收多达 128 个字节的额外数据。 这种情况下, 当系统预期的数据量减少到两个包时, 将控制流程从端点 2 和端点 4 的正常处理变为下图所示的流程。 该流程图说明如何控制端点 2。 可用同样的顺序对端点 4 进行控制。 因此, 要使用该流程图作为端点 4 的控制流程, 需在流程图中将端点 2 的位名称读作端点 4 的位名称。

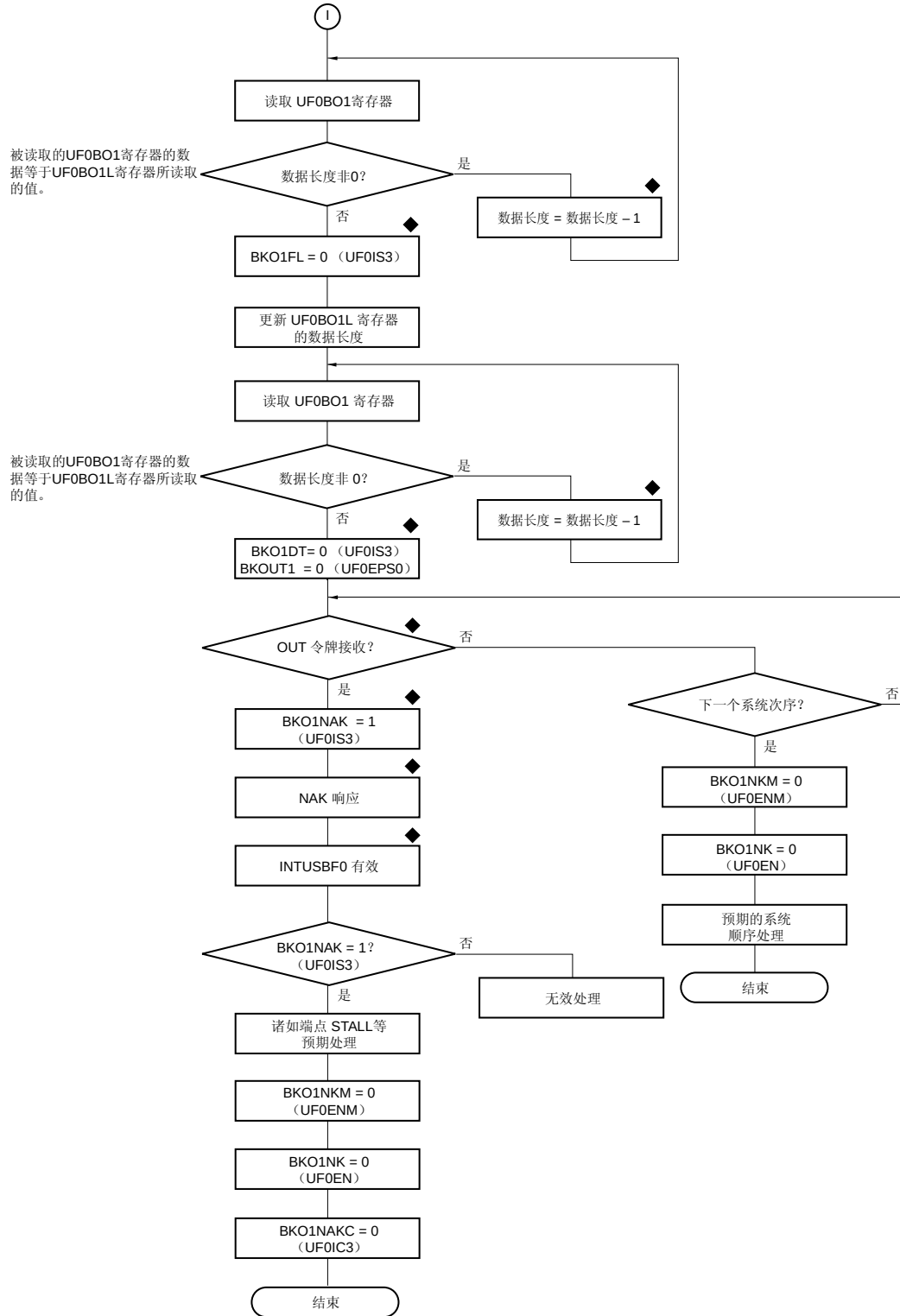
图 21-28. 如果发送多于系统预期的数据时的处理 (端点 2) (1/2)



备注

1. $n = 2, 3$
2. ◆: 由硬件处理

图 21-28. 如果发送多于系统预期的数据时的处理 (端点 2) (2/2)

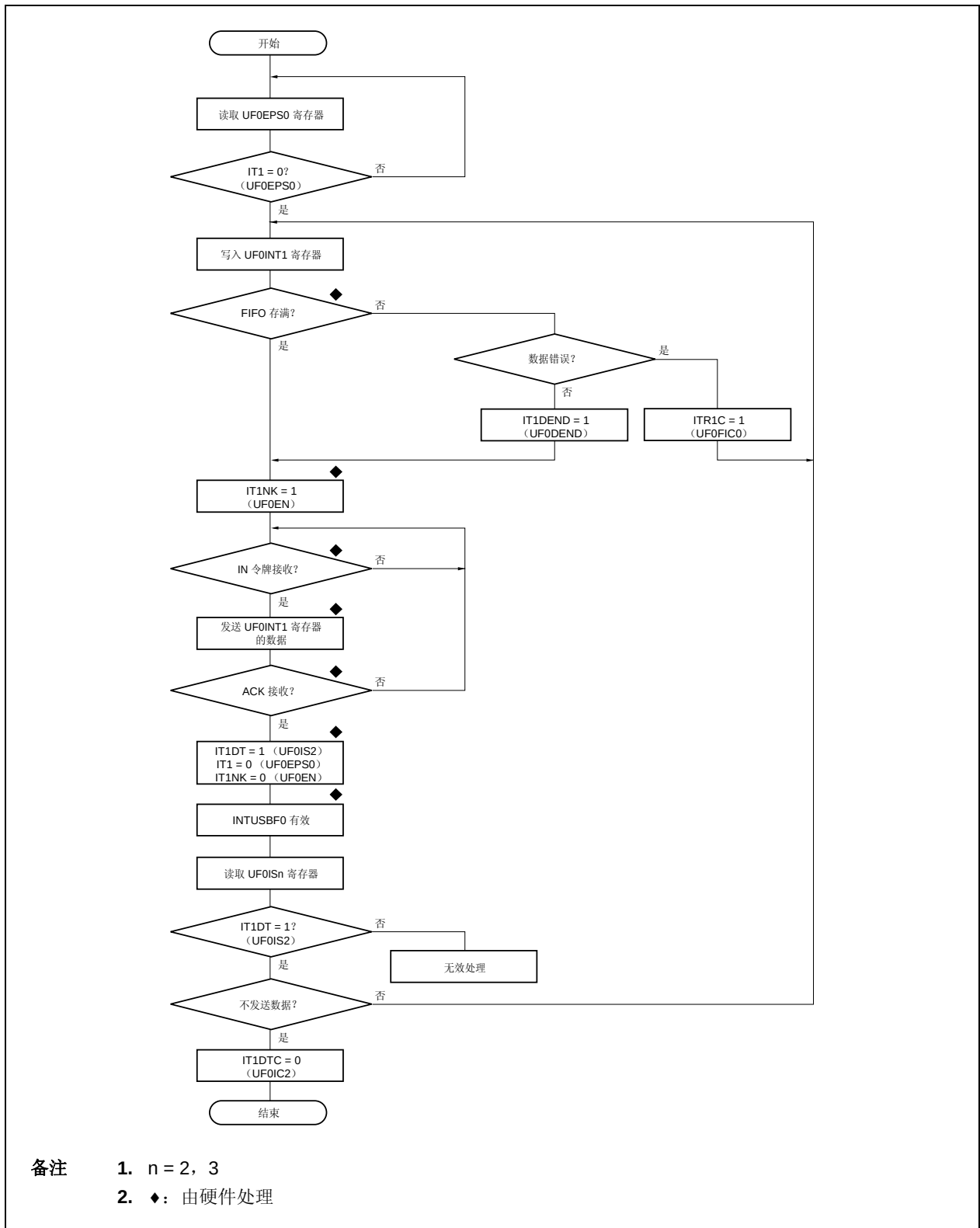


备注 ◆: 由硬件处理

(6) 中断传输 (IN) 的处理

中断传输 (IN) 被分配至端点 7。流程图如图 21-29 所示。

图 21-29. 中断传输 (IN) 的处理 (端点 7)



21.9.4 中止/重传处理

如何进行中止/重传处理根据系统的配置不同而异。下面给出一个示例。

图 21-30. 中止/重传处理的示例 (1/3)

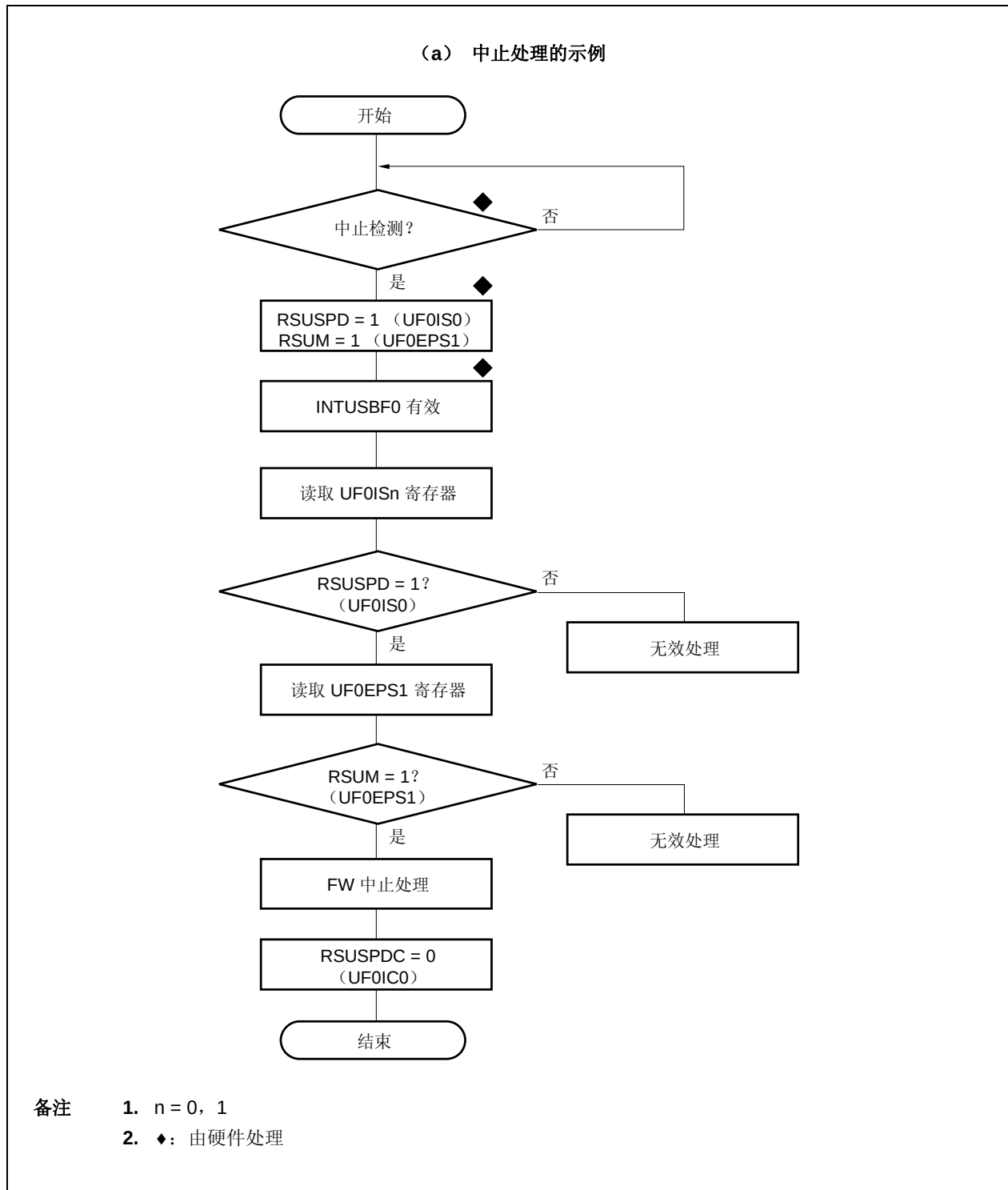


图 21-30. 中止/重传处理的示例 (2/3)

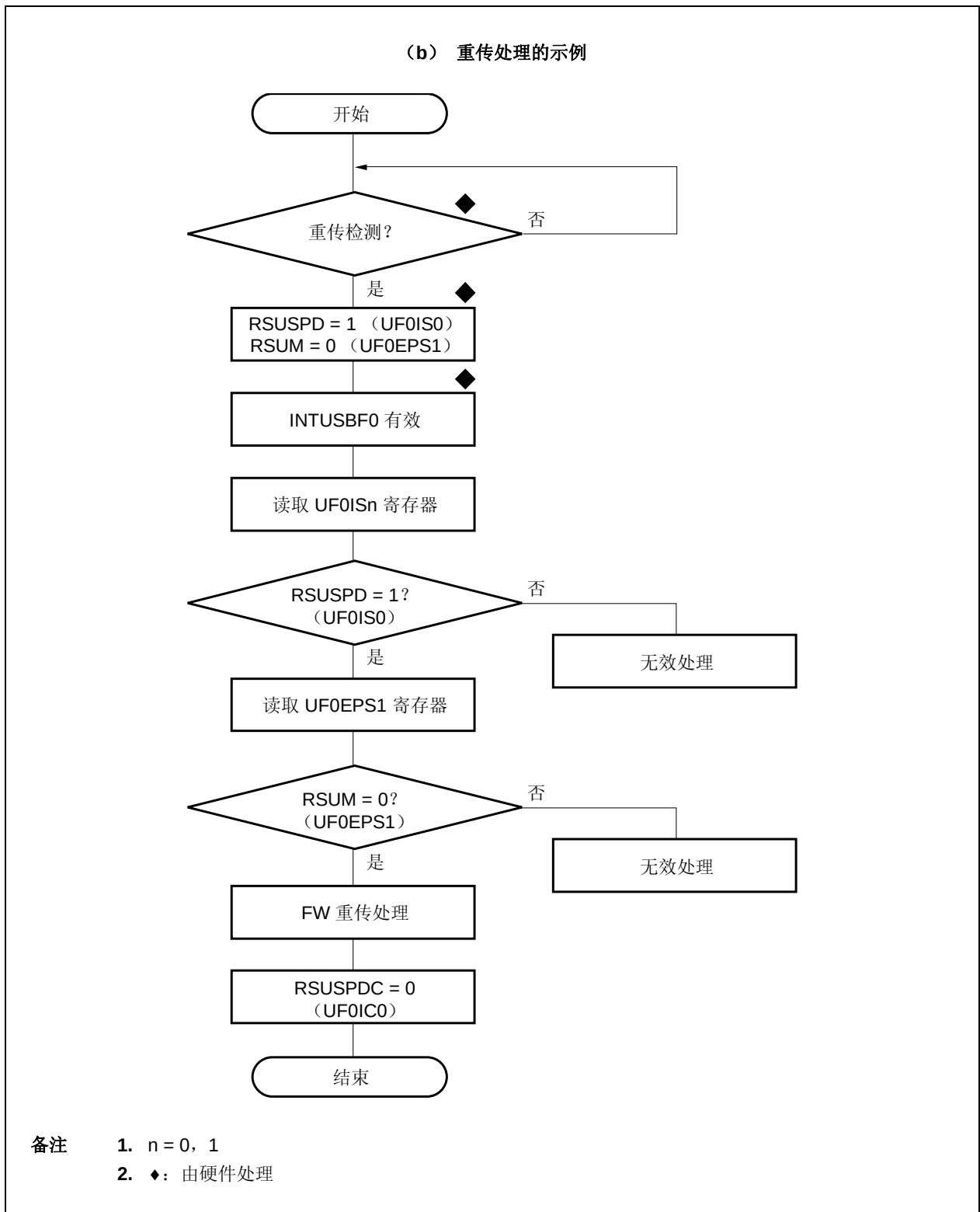
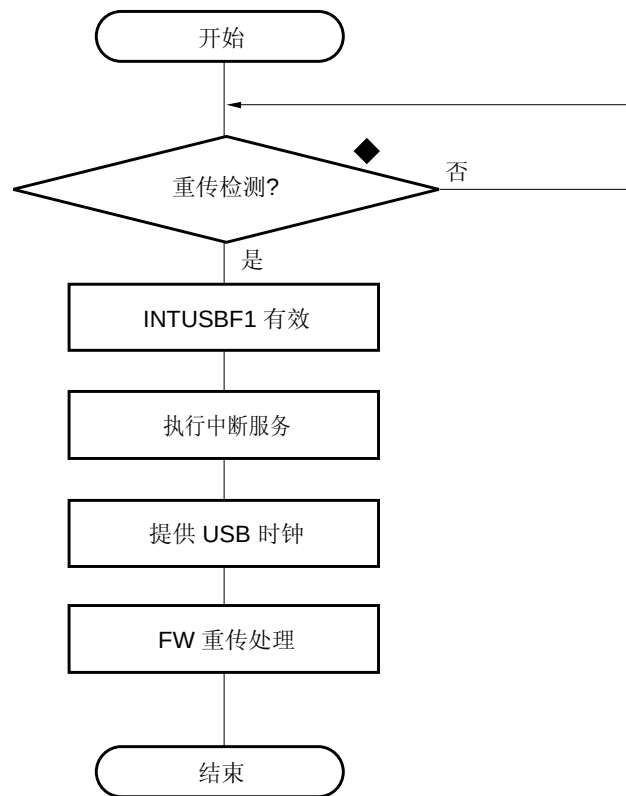


图 21-30. 中止/重传处理的示例 (3/3)

(c) 重传处理的示例 (当停止给 USBF 提供 USB 时钟时)



备注 ♦: 由硬件处理

21.9.5 加电之后的处理

加电之后的执行的处理根据系统的配置不同而异。 下面给出一个示例。

图 21-31. 加电/断电之后的处理示例 (1/3)

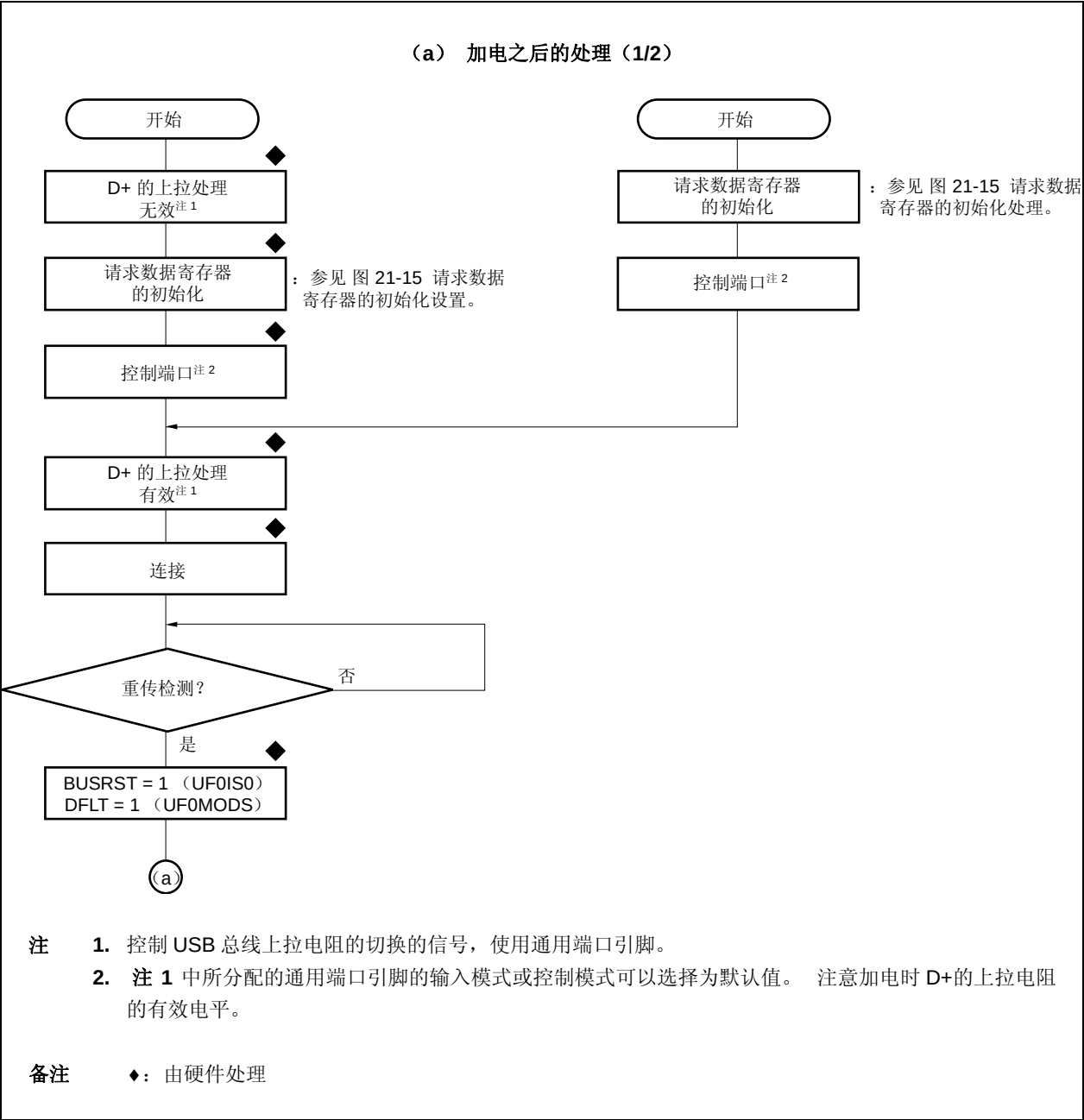
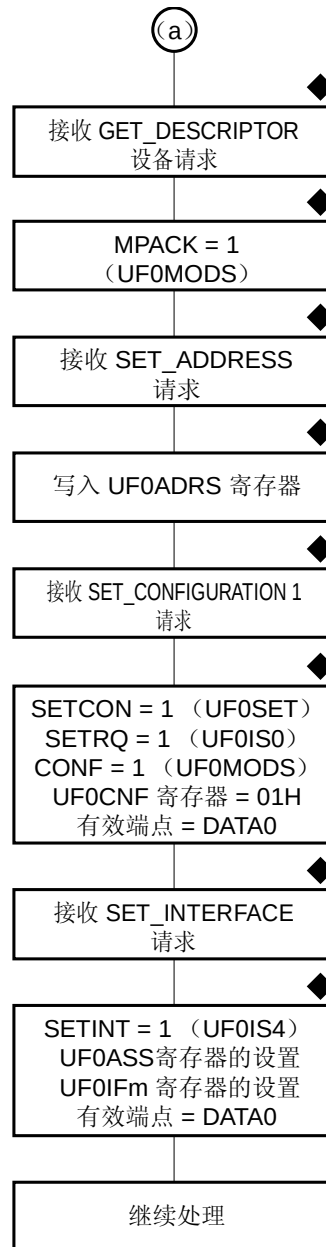


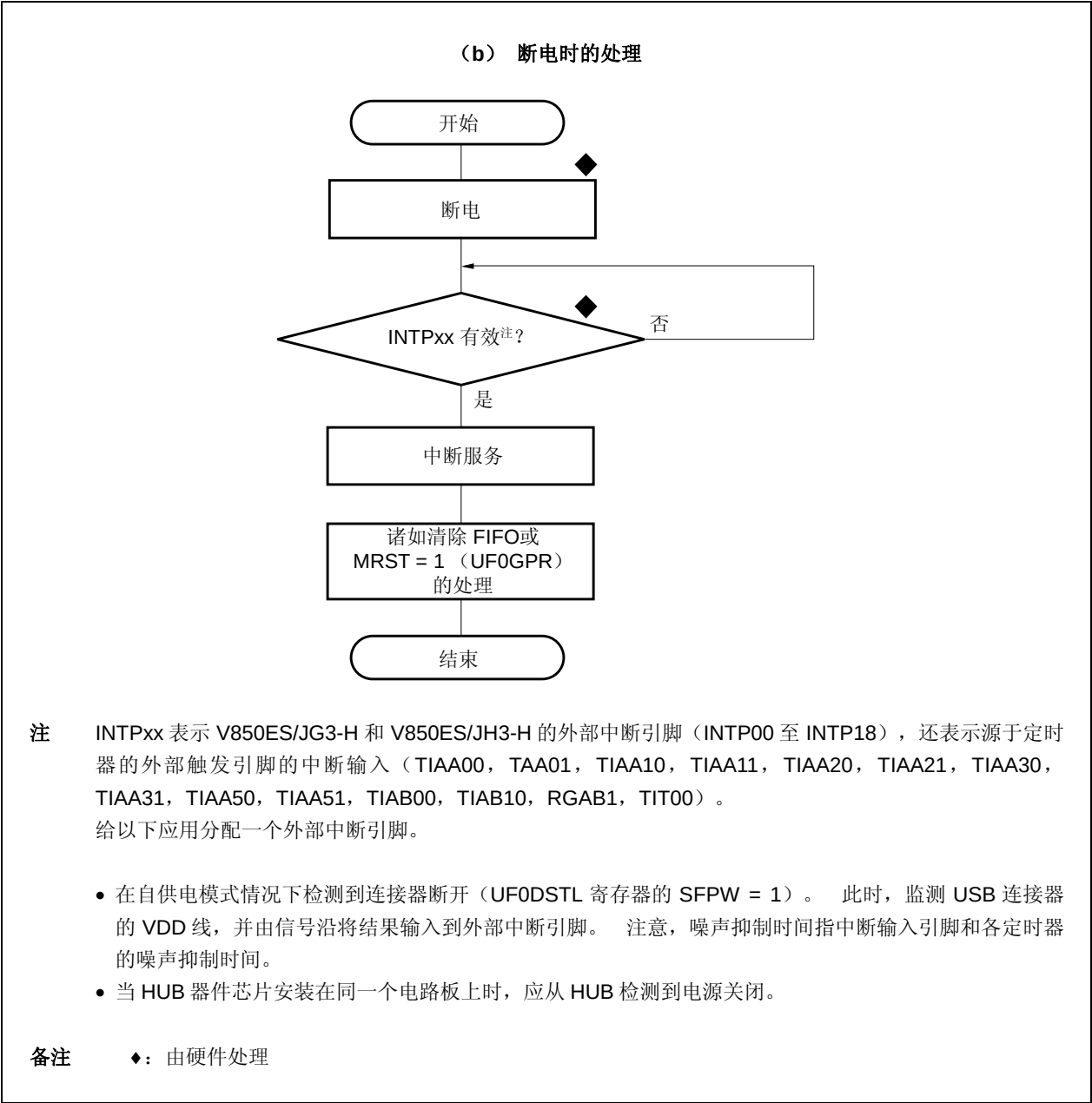
图 21-31. 加电/断电之后的处理示例 (2/3)

(a) 加电之后的处理 (2/2)



- 备注
1. m = 0 至 4
 2. ♦: 由硬件处理

图 21-31. 加电/断电之后的处理示例 (3/3)



21.9.6 在DMA模式下接收bulk传输 (OUT) 数据

Bulk 传输 (OUT) 被分配至端点 2 和端点 4。下面所示的流程图说明了使用 DMA 时如何控制端点 2。可用同样的顺序对端点 4 进行控制。因此,要使用该流程图作为端点 4 的控制流程,需在流程图中将端点 2 的位名称读作端点 4 的位名称。以下显示的控制流程说明 CPU 如何读取剩余的数据。

如果通过将 UF0IDR 寄存器的 DQBO1MS 位设为 1 正确接收了 bulk 传输 (OUT) 的数据,则端点 2 的 DMA 请求信号有效,而不是中断请求 (INTUSBF0) 有效。端点 2 的该 DMA 请求信号根据 UF0IDR 寄存器 MODEn 位的设置工作 (n = 0, 1)。如果 DMA 读取了 UF0BO1 寄存器存储的所有数据,则端点 2 的 DMA 请求信号变为无效。这种情况下,如果正确接收了下一个 bulk 传输 (OUT) 的数据,则端点 2 的 DMA 请求信号再次有效。如果已接收的 bulk 传输 (OUT) 数据等于或小于 FIFO 的大小,则一旦 DMA 完成数据读取,就会产生一个短中断请求且 INTUSBF0 (EP2_ENDINT) 信号有效。要重新由 DMA 读取数据,则需再次将 DQBO1MS 位设为 1。如果通过端点 2 的 DMA 结束信号完成 DMA,则将 UF0IDR 寄存器的 DQBO1MS 位清为 0,且端点 2 的 DMA 请求信号变为无效。同时,产生 DMA_END 中断请求。此时如果数据仍位于 UF0BO1 寄存器,则可通过重新设置 UF0IDR 寄存器的 DQBO1MS 位再次启动 DMA。但是,bulk 传输 (OUT) 的数据总是等于或小于 FIFO 的大小。因此,一旦数据被 DMA 读取,就产生短中断请求,INTUSBF0 (EP2_ENDINT) 变为有效,DQBO1MS 位被清除,且端点 2 的 DMA 请求信号变为无效。

- 注意事项**
1. 在命令模式下,只要存在待发送的数据,端点 n 的 DMA 请求信号 (n = 2, 4) 就会变为有效 (UF0IDR 寄存器的 MODE1 和 MODE0 位 = 10)。
 2. 在单次模式下,如果存在待发送的数据,则端点 n 的 DMA 请求信号 (n = 2, 4) 变为有效 (UF0IDR 寄存器的 MODE1 和 MODE0 位 = 0X (X: 忽略)),但是每发送一个字节该信号就会变为无效。该操作会重复进行一直到没有要发送的数据为止。

图 21-32. Bulk 传输 (OUT) 的 DMA 处理 (1/3)

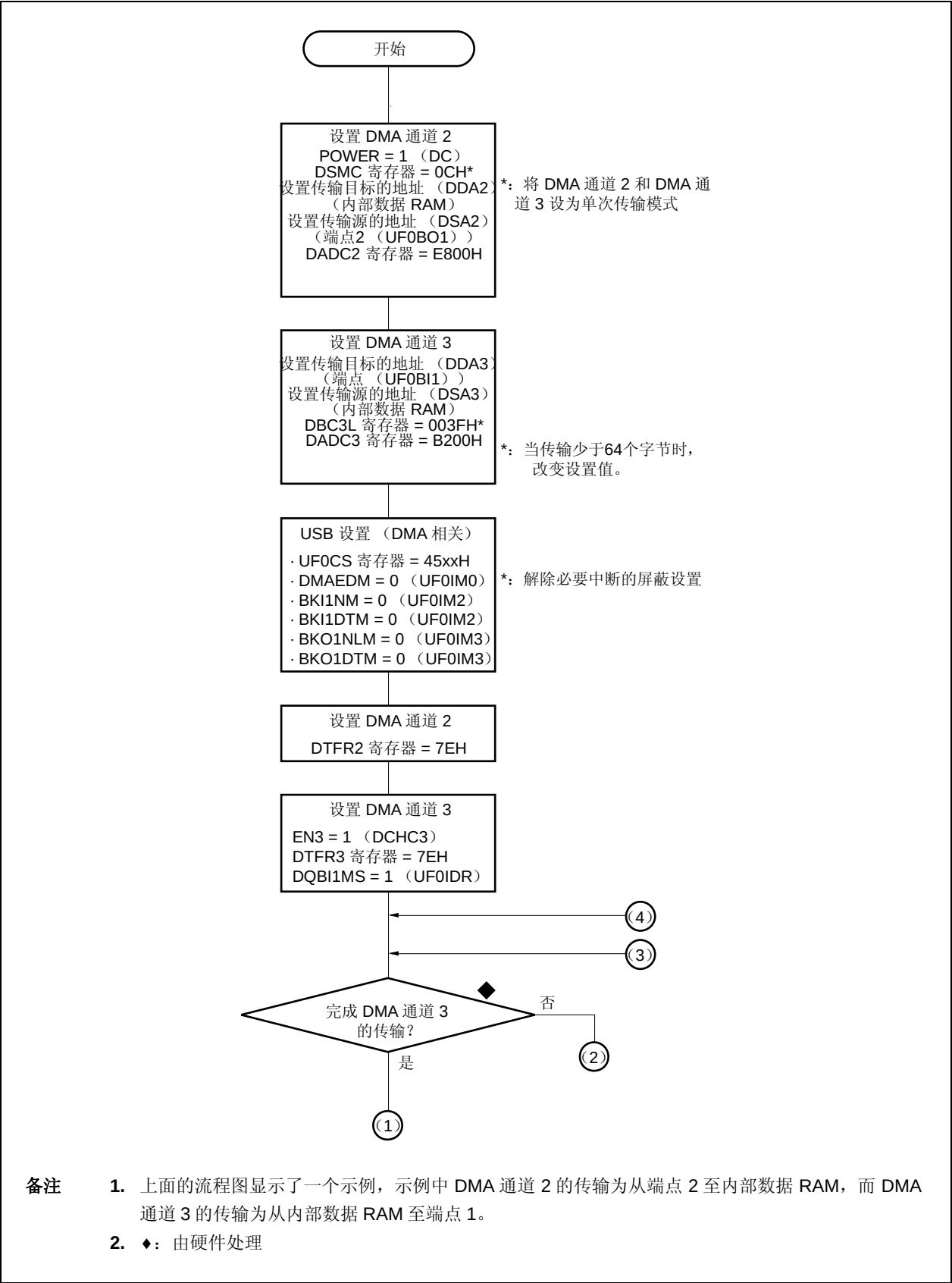
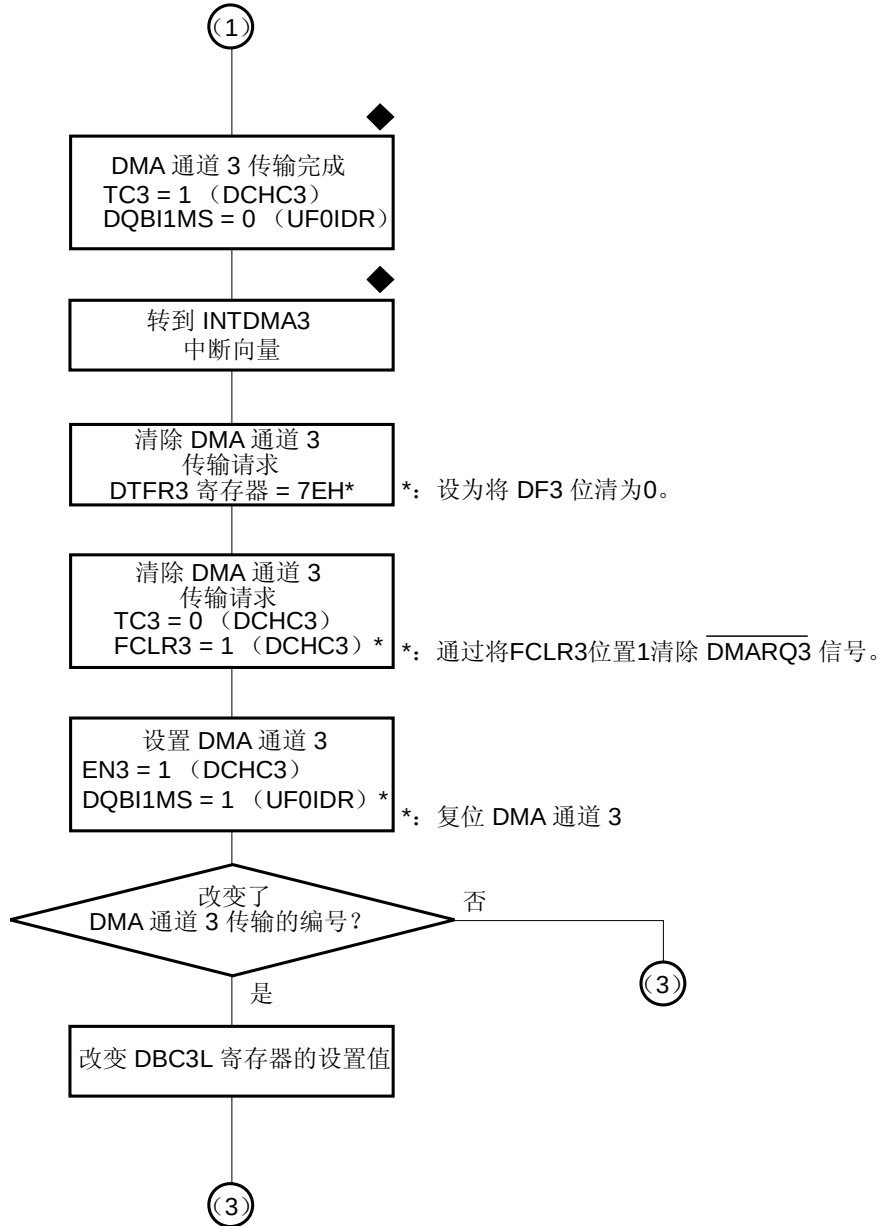


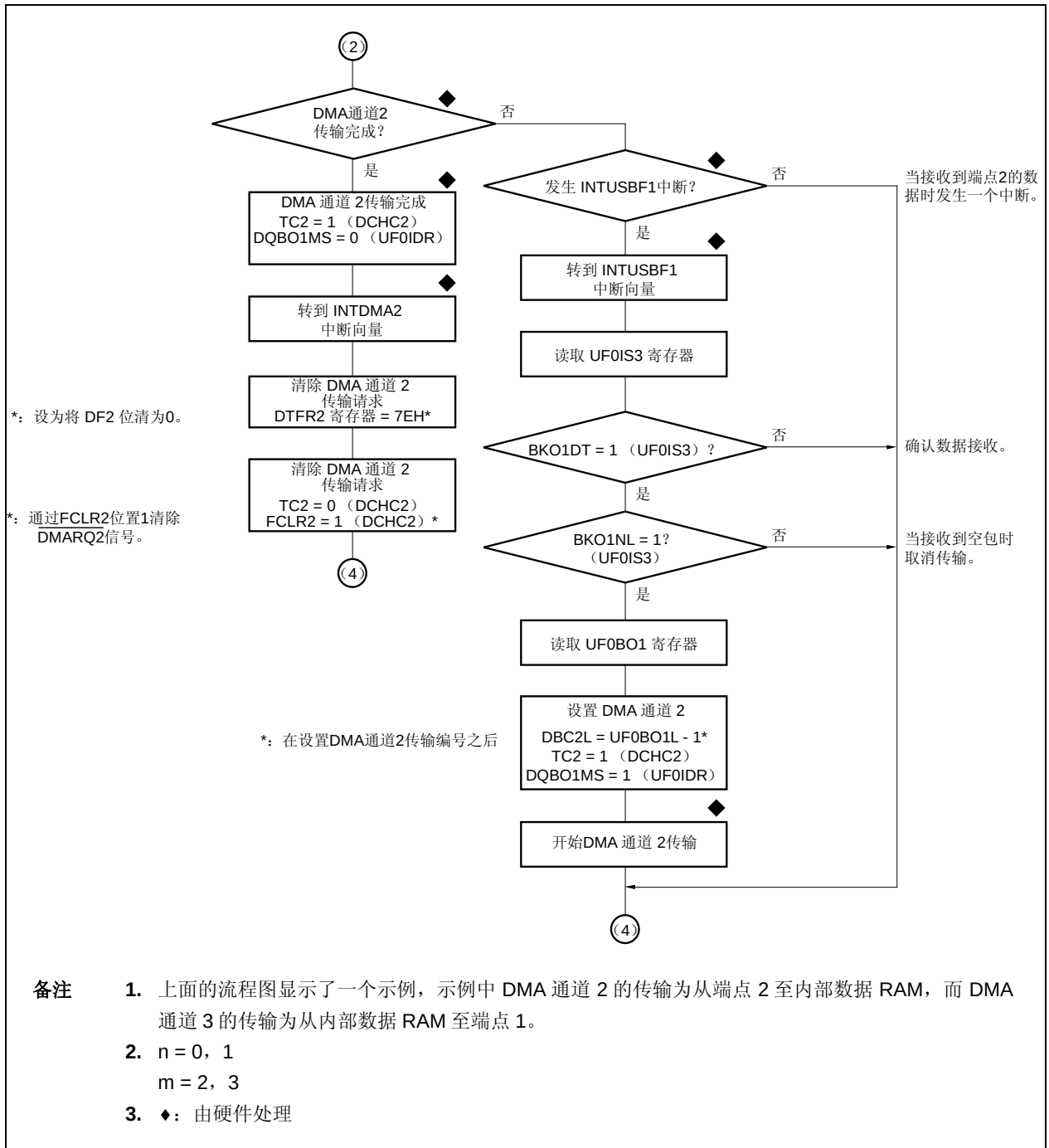
图 21-32. Bulk 传输 (OUT) 的 DMA 处理 (2/3)



备注

1. 上面的流程图显示了一个示例，示例中 DMA 通道 2 的传输为从端点 2 至内部数据 RAM，而 DMA 通道 3 的传输为从内部数据 RAM 至端点 1。
2. $n = 0, 1$
3. ◆：由硬件处理

图 21-32. Bulk 传输 (OUT) 的 DMA 处理 (3/3)



21.9.7 在DMA模式下发送bulk传输 (IN) 数据

Bulk 传输 (IN) 被分配至端点 1 和端点 3。下面所示的流程图说明了使用 DMA 时如何控制端点 1。可用同样的顺序对端点 3 进行控制。因此,要使用该流程图作为端点 3 的控制流程,需在流程图将端点 1 的位名称读作端点 3 的位名称。

如果通过将 UF0IDR 寄存器的 DQBI1MS 位设为 1 写入 bulk 传输 (IN) 数据,则端点 1 的 DMA 请求信号有效,而不是中断请求 (INTUSBF0) 变为有效。此端点 1 的 DMA 请求信号根据 UF0IDR 寄存器 MODEn 位的设置工作 (n = 0, 1)。如果由 DMA 已将所有可写入 UF0BI1 寄存器的所有数据写入,则端点 1 的 DMA 请求信号变为无效。这种情况下,发生 FIFO 的切换操作,而且,如果能写入 bulk 传输 (IN) 的数据,则端点 1 的 DMA 请求信号会重新变为有效。即使 FIFO 由于 DMA 传输变满也不会执行 FIFO 的自动切换操作,除非将 UF0DEND 寄存器的 BKI1T 位设为 1。因此,要传输数据务必将 UF0DEND 寄存器的 BKI1DE 位设为 1。如果由端点 1 的 DMA 结束信号完成 DMA,则将 UF0IDR 寄存器的 DQBI1MS 位清为 0,且端点 1 的 DMA 请求信号变为无效。同时,产生 DMA_END 中断请求。要在 FIFO 未存满时发送一个短包,则需将 UF0DEND 寄存器的 BKI1DED 位设为 1。

- 注意事项**
1. 在请求模式下,只要数据可被发送,端点 n 的 DMA 请求信号 (n = 1, 3) 就会变为有效 (UF0IDR 寄存器的 MODE1 和 MODE0 位 = 10)。
 2. 在单次模式下,如果可发送数据,则端点 n 的 DMA 请求信号 (n = 1, 3) 变为有效 (UF0IDR 寄存器的 MODE1 和 MODE0 位 = 0X (X: 忽略)),但是每发送一个字节该信号就会变为无效。该操作会重复进行一直到没有要发送的数据为止。

图 21-33. Bulk 传输 (IN) 的 DMA 处理 (1/4)

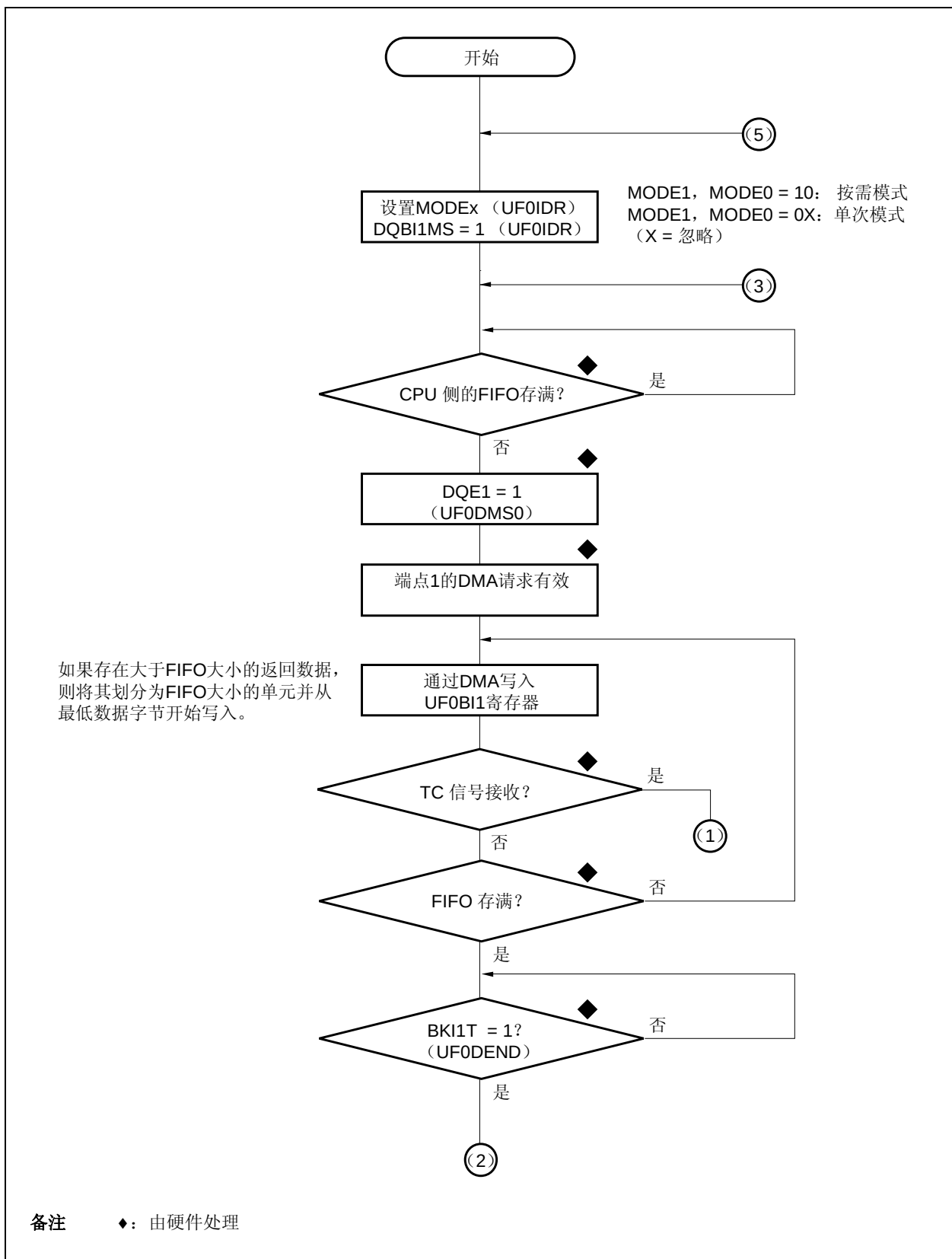


图 21-33. Bulk 传输 (IN) 的 DMA 处理 (2/4)

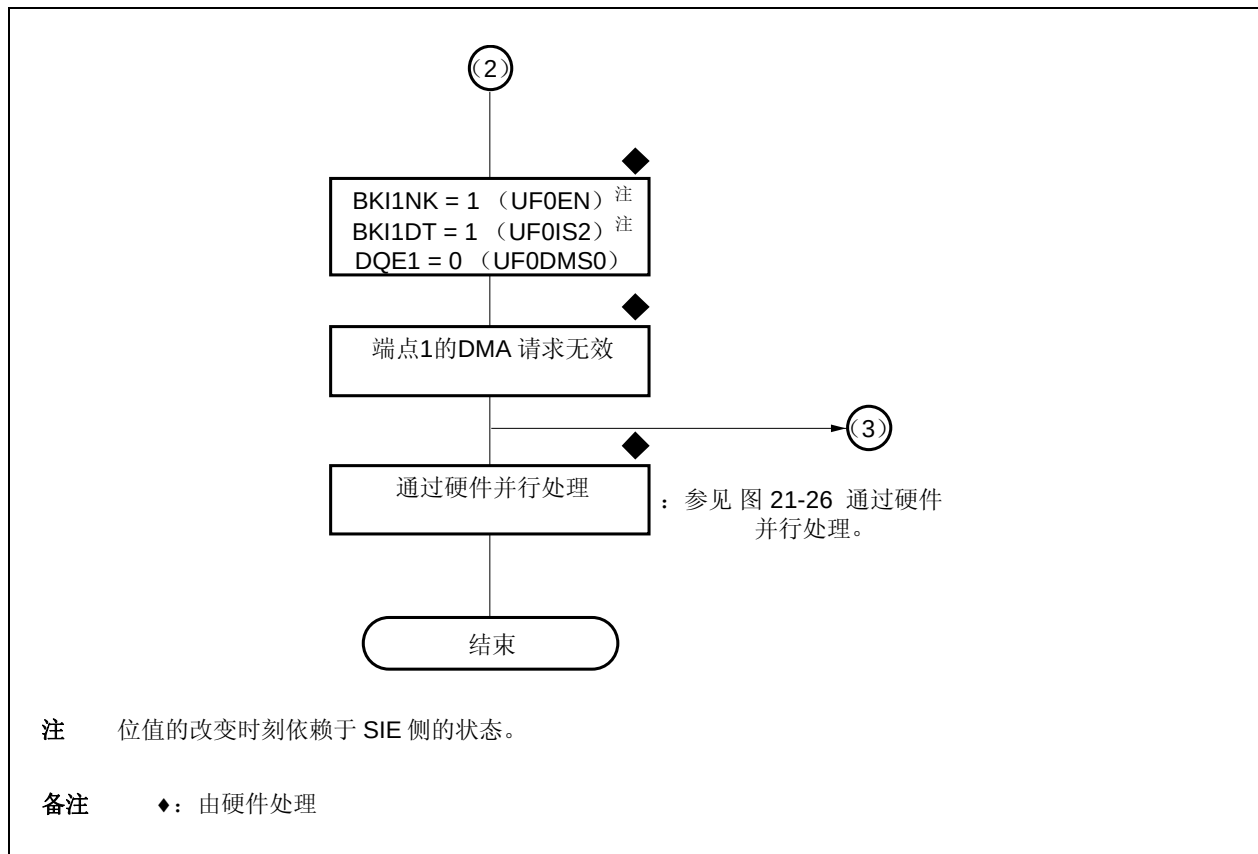


图 21-33. Bulk 传输 (IN) 的 DMA 处理 (3/4)

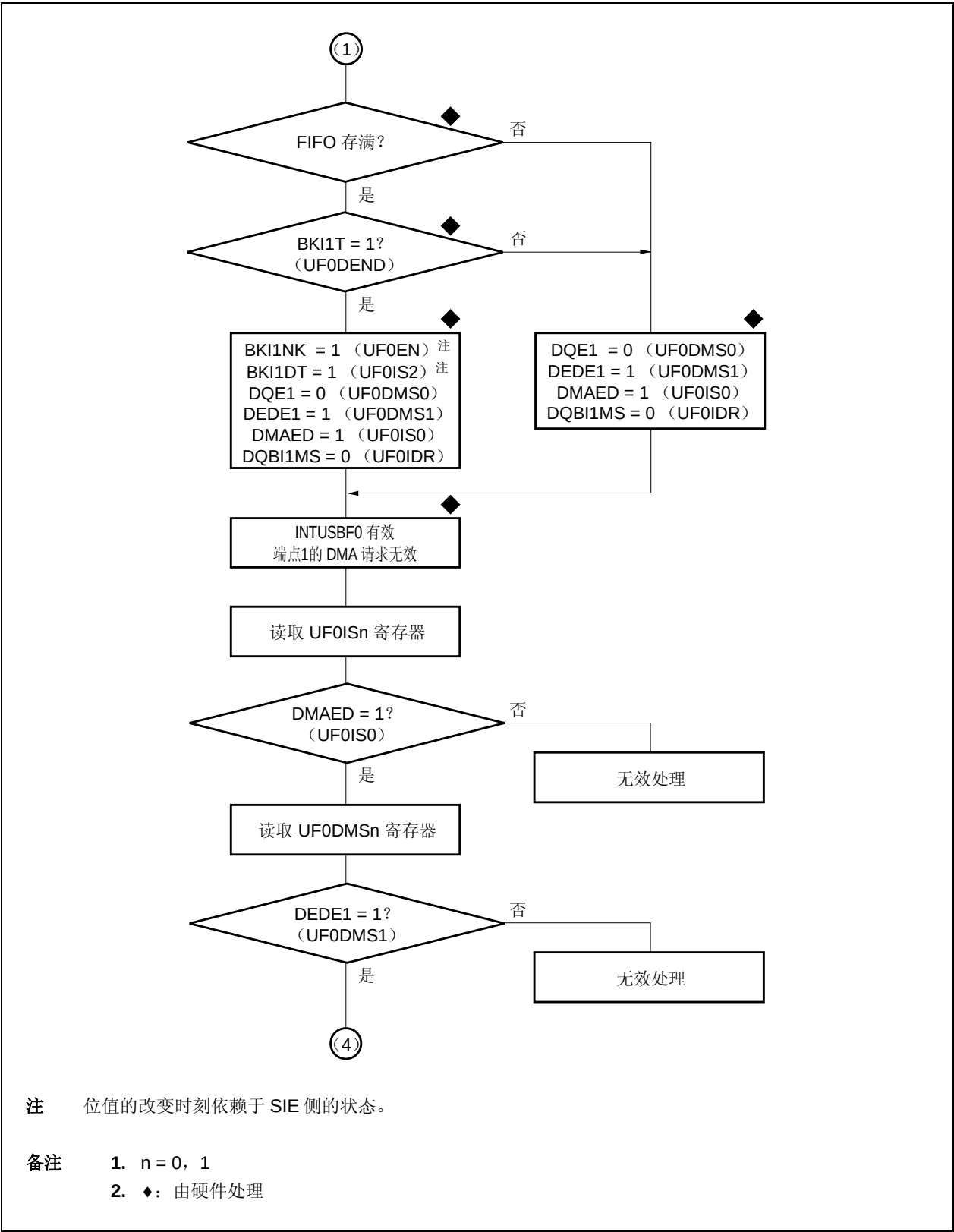
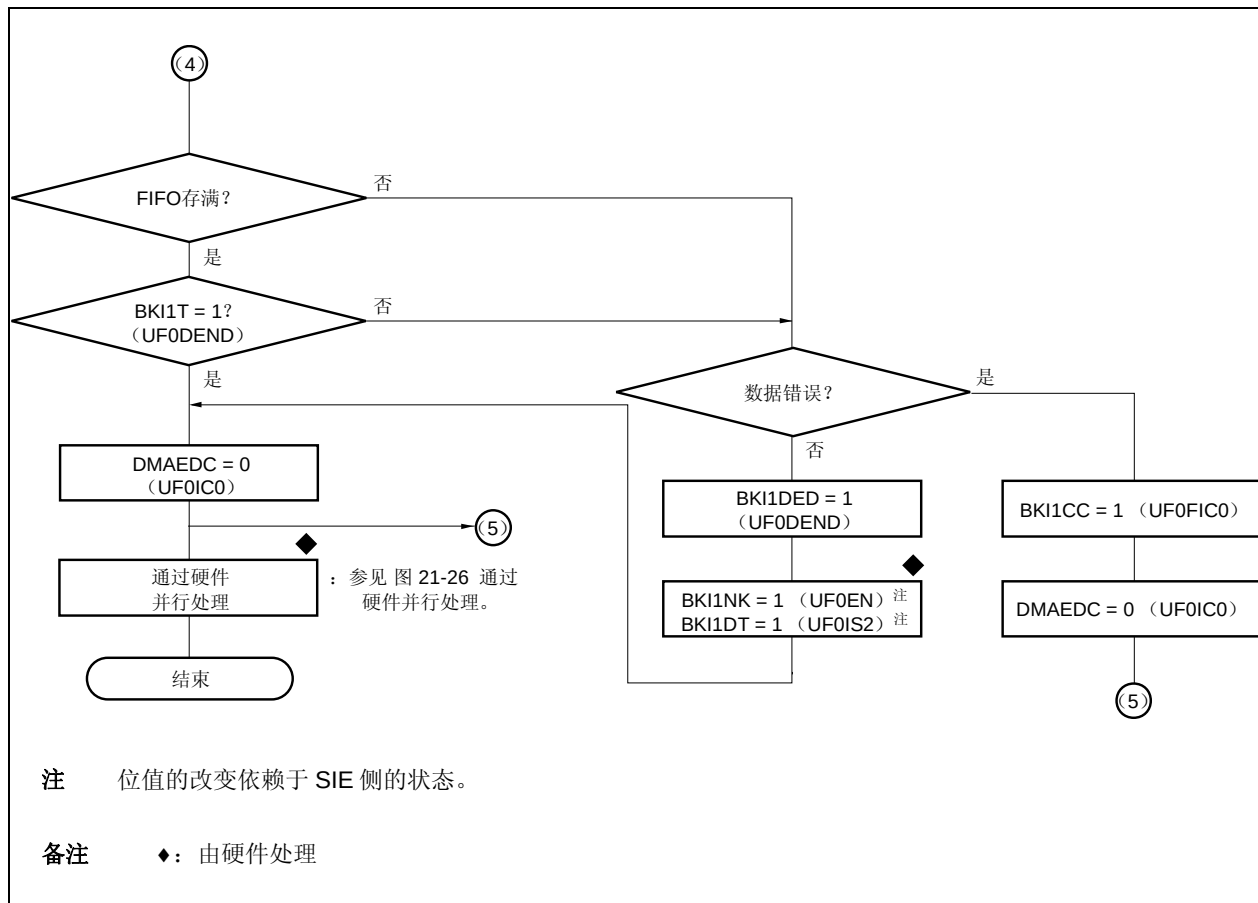


图 21-33. Bulk 传输 (IN) 的 DMA 处理 (4/4)



第二十二章 DMA功能（DMA控制器）

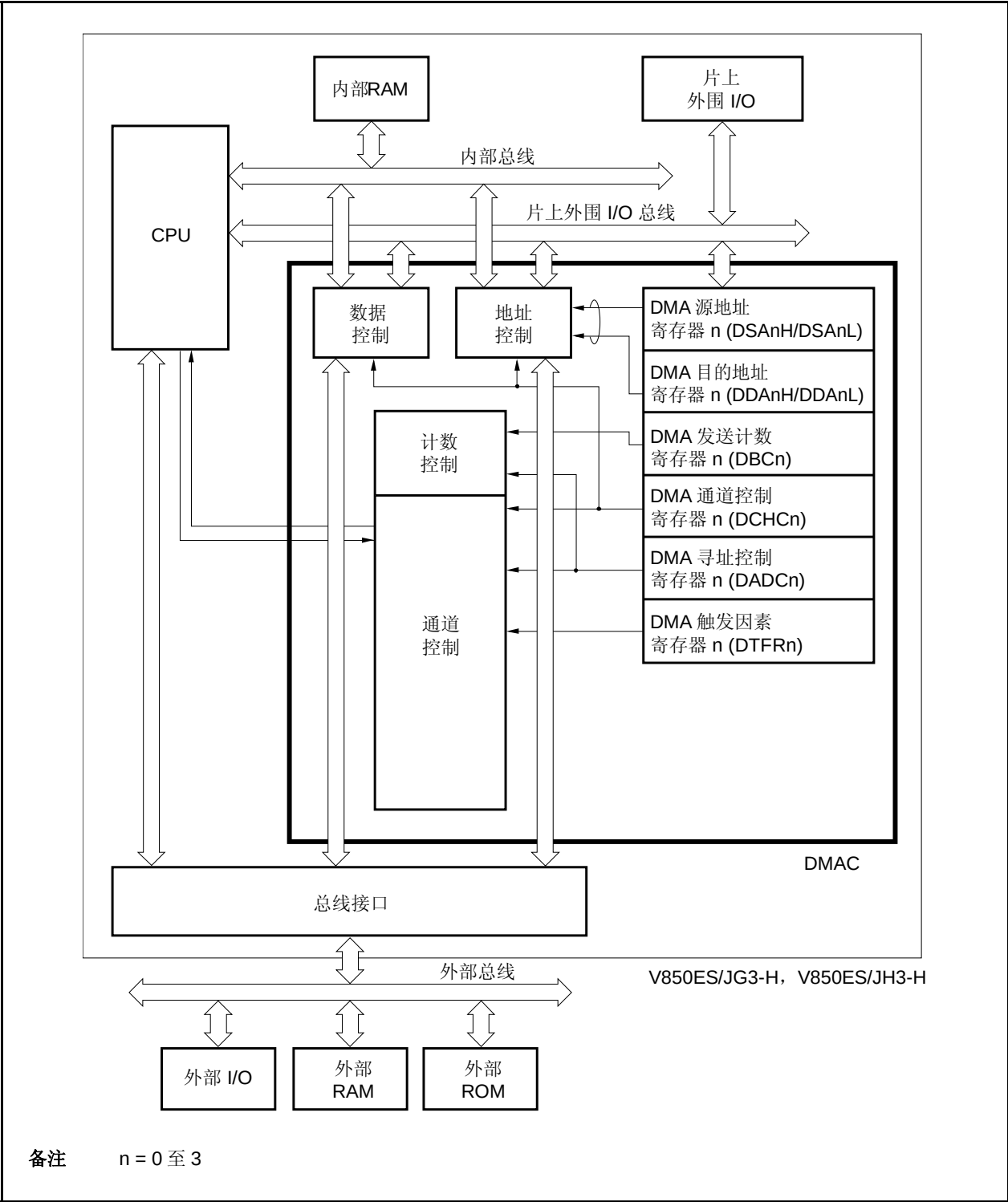
V850ES/JG3-H 和 V850ES/JH3-H 单片机包括一个直接存储器访问（DMA）控制器（DMAC），用于执行和控制 DMA 传送。

DMAC 根据片上外围 I/O（串行接口、定时器/计数器和 A/D 转换器）发出的 DMA 请求、来自外部输入引脚的中断或软件触发（存储器是指内部 RAM 或外存储器），控制存储器与 I/O 之间、存储器之间或 I/O 之间的数据传送。

22.1 特性

- 4 个独立 DMA 通道
- 传送单位：8/16 位
- 最大传送计数：65,536 (2^{16})
- 传送类型：双周期传送
- 传送模式：单次传送模式
- 传送请求
 - 来自片上外围 I/O（串行接口、定时器/计数器和 A/D 转换器）或来自外部输入引脚的中断发出请求
 - 软件触发发出的请求
- 传送目标
 - 内部 RAM ↔ 外围 I/O
 - 外围 I/O ↔ 外围 I/O
 - 内部 RAM ↔ 外存储器
 - 外存储器 ↔ 外围 I/O
 - 外存储器 ↔ 外存储器

22.2 配置



22.3 寄存器

(1) DMA 源地址寄存器 0 至 3 (DSA0 至 DSA3)

DSA0 至 DSA3 寄存器用于为 DMA 通道 n 设置 DMA 源地址 (每个 26 位) (n = 0 至 3)。

这些寄存器被分为两个 16 位寄存器, 即 DSA_nH 和 DSA_nL。

这些寄存器可以按 16 位读取或写入。

复位后: 不确定 R/W 地址: DSA0H FFFFF082H, DSA1H FFFFF08AH,
DSA2H FFFFF092H, DSA3H FFFFF09AH,
DSA0L FFFFF080H, DSA1L FFFFF088H,
DSA2L FFFFF090H, DSA3L FFFFF098H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DSA _n H (n = 0 至 3)	IR	0	0	0	0	0	SA25	SA24	SA23	SA22	SA21	SA20	SA19	SA18	SA17	SA16

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DSA _n L (n = 0 至 3)	SA15	SA14	SA13	SA12	SA11	SA10	SA9	SA8	SA7	SA6	SA5	SA4	SA3	SA2	SA1	SA0

IR	DMA 传送源分类
0	外部存储或片上外围 I/O
1	内部RAM

SA25 至 SA16	设定DMA传送源地址(A25 至 A16) (缺省值未定义)。 DMA传送时, 下一个DMA传送源地址保持。 DMA传送完成后, DMA地址设定首先保持。

SA15 至 SA0	设定DMA传送源地址(A15 至 A0) (缺省值未定义)。 DMA传送时, 下一个DMA传送源地址保持。 DMA传送完成后, DMA地址设定首先保持。

- 注意事项**
1. 确保将 DSA_nH 寄存器的位 14 至位 10 清零。
 2. 当 DMA 传送禁止 (DCHCn.Enn 位 = 0) 时, 按以下时序设置 DSA_nH 和 DSA_nL 寄存器。
 - 从复位后到第一个 DMA 传送开始期间
 - 从通道被 DCHCn.INITn 位初始化后到 DMA 传送开始期间
 - 从 DMA 传送完成后 (DCHCn.TCn 位 = 1) 到下一个 DMA 传送开始期间
 3. 当 DSA_n 寄存器的值被读时, 两个 16 位寄存器, 即 DSA_nH 和 DSA_nL 被读出。 如果读取与更新发生冲突, 则可能读取正在被更新的值 (请参见 22.13 注意事项)
 4. 复位之后, 在开始 DMA 发送之前, 设置 DSA_nH, DSA_nL, DDA_nH, DDA_nL 和 DBCn 寄存器。 如果没有设置这些寄存器, 当 DMA 传输开始时操作不能得到保证。

(2) DMA 目的地址寄存器 0 至 3 (DDA0 至 DDA3)

DDA0 至 DDA3 寄存器用于为 DMA 通道 n 设置 DMA 目的地址 (每个 26 位) (n = 0 至 3)。这些寄存器被分为两个 16 位寄存器, 即 DDAnH 和 DDAnL。

这些寄存器可以按 16 位读取或写入。

复位后: 不确定 R/W 地址: DDA0H FFFFF086H, DDA1H FFFFF08EH,
DA2H FFFFF096H, DDA3H FFFFF09EH,
DDA0L FFFFF084H, DDA1L FFFFF08CH,
DDA2L FFFFF094H, DDA3L FFFFF09CH

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DDAnH (n = 0 至 3)	IR	0	0	0	0	0	DA25	DA24	DA23	DA22	DA21	DA20	DA19	DA18	DA17	DA16

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DDAnL (n = 0 至 3)	DA15	DA14	DA13	DA12	DA11	DA10	DA9	DA8	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0

IR	DMA 传送目的说明
0	外部存储器或片上外围 I/O
1	内部 RAM

DA25 至 DA16	设定一个DMA传送目的地 (A25 至 A16) (缺省值不确定)。 DMA传送期间, 下一个DMA传送目的地保持。 当DMA传送完成时, 首先设定的DMA传送源地址保持。

DA15 至 DA0	设定一个DMA传送目的地 (A15 至 A0) (缺省值不确定)。 DMA传送期间, 下一个DMA传送目的地保持。 当DMA传送完成时, 首先设定的DMA传送源地址保持。

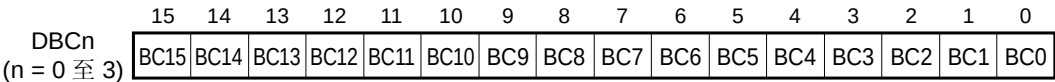
- 注意事项**
- 务必将 DDAnH 寄存器的位 14 至位 10 清零。
 - 当 DMA 传送禁止 (DCHCn.Enn 位 = 0) 时, 按以下时序设置 DDAnH 和 DDAnL 寄存器。
 - 从复位后到第一个 DMA 传送开始期间
 - 从通道被 DCHCn.INITn 位初始化后到 DMA 传送开始期间
 - 从 DMA 传送完成后 (DCHCn.TCn 位 = 1) 到下一个 DMA 传送开始期间
 - 当 DDAn 寄存器的值被读时, 两个 16 位寄存器, 即 DDAnH 和 DDAnL 被读出。如果读取与更新发生冲突, 则可能读取正在被更新的值 (请参见 22.13 注意事项)
 - 复位之后, 在开始 DMA 发送之前, 设置 DSAH, DSAH, DDAnH, DDAnL 和 DBCn 寄存器。如果没有设置这些寄存器, 当 DMA 传输开始时操作不能得到保证。

(3) DMA 传输计数寄存器 0 至 3 (DBC0 至 DBC3)

DBC0 至 DBC3 寄存器是 16 位寄存器，用于设置 DMA 通道 n 的传送计数 (n = 0 至 3)。在 DMA 传送过程中，这些寄存器保存剩余的传送计数。

无论传送数据的单位是什么 (8/16 位)，每传送一次，这些寄存器就递减 1，当发生借位时，传送终止。这些寄存器可以按 16 位读取或写入。

复位后: 不确定 R/W 地址: DBC0 FFFFF0C0H, DBC1 FFFFF0C2H,
DBC2 FFFFF0C4H, DBC3 FFFFF0C6H



BC15 至 BC0	传输计数设置或保持 DMA 传输时传送计数
0000H	传送计数 1 或保持字节传送计数
0001H	传送计数 2 或保持字节传送计数
:	:
FFFFH	传送计数 65,536 (2 ¹⁶) 或保持字节传送计数
DMA 传送完成时，传送数据的数目首先保持。	

- 注意事项 1. 当 DMA 传送禁止 (DCHCn.Enn 位 = 0) 时，按以下时序设置 DBCn 寄存器。
- 从复位后到第一个 DMA 传送开始期间
 - 从通道被 DCHCn.INITn 位初始化后到 DMA 传送开始期间
 - 从 DMA 传送完成后 (DCHCn.TCn 位 = 1) 到下一个 DMA 传送开始期间
2. 复位之后，在开始 DMA 发送之前，设置 DSAnH, DSAnL, DDAnH, DDAnL 和 DBCn 寄存器。如果没有设置这些寄存器，当 DMA 传输开始时操作不能得到保证。

(4) DMA 寻址控制寄存器 0 至 3 (DADC0 至 DADC3)

DADC0 至 DADC3 寄存器是 16 位寄存器，用于控制 DMA 通道 n 的 DMA 传送模式 (n = 0 至 3)。

这些寄存器可以按 16 位读取或写入。

系统复位将这些寄存器设置为 0000H。

复位后: 0000H R/W 地址: DADC0 FFFFF0D0H, DADC1 FFFFF0D2H,
DADC2 FFFFF0D4H, DADC3 FFFFF0D6H

	15	14	13	12	11	10	9	8
DADCn	0	DS0	0	0	0	0	0	0
(n = 0 至 3)	7	6	5	4	3	2	1	0
	SAD1	SAD0	DAD1	DAD0	0	0	0	0

DS0	发送数据大小的设置
0	8 位
1	16 位

SAD1	SAD0	发送源地址计数方向的设置
0	0	增加
0	1	减少
1	0	固定
1	1	设置禁止

DAD1	DAD0	地址计数方向的设置
0	0	增加
0	1	减少
1	0	固定
1	1	设置禁止

- 注意事项**
1. 确保将 DADCn 寄存器的位 15、位 13 至位 8 及位 3 至位 0 清零。
 2. 当 DMA 传送禁止 (DCHCn.Enn 位 = 0) 时，按以下时序设置 DADCn 寄存器。
 - 从复位后到第一个 DMA 传送开始期间
 - 从通道被 DCHCn.INITn 位初始化后到 DMA 传送开始期间
 - 从 DMA 传送完成后 (DCHCn.TCn 位 = 1) 到下一个 DMA 传送开始期间
 3. DS0 位指定传输数据长度，但是不控制总线的宽度。如果设置了 8 位数据长度 (DS0 位 = 0)，则较低的数据总线将不被使用。
 4. 如果传送数据长度被设置为 16 位 (DS0 位 = 1)，则不能从奇地址开始传送。传送始终从这样的地址开始，即低位地址的第一位为 0。
 5. 如果 DMA 传送是在一个片上外围 I/O 寄存器 (作为传送源或目的地) 上执行的，请确保指定与寄存器单位相同的传送长度。例如，要在一个 8 位寄存器上执行 DMA 传送，务必指定 8 位长度传输。

(5) DMA 通道控制寄存器 0 至 3 (DCHC0 至 DCHC3)

DCHC0 至 DCHC3 寄存器是 8 位寄存器，用于控制 DMA 通道 n 的 DMA 传送操作模式。

这些寄存器可以按 8 位或按 1 位为单位进行读写。（不过，位 7 是只读的，而位 1 和位 2 是只写的。如果读取位 1 或位 2，则被读出的值将始终是 0。

复位后这些寄存器设置为 00H。

复位后: 00H R/W 地址: DCHC0 FFFFF0E0H, DCHC1 FFFFF0E2H,
DCHC2 FFFFF0E4H, DCHC3 FFFFF0E6H

	<7>	6	5	4	3	<2>	<1>	<0>
DCHCn	TCn ^{注 1}	0	0	0	0	INITn ^{注 2}	STGn ^{注 2}	Enn

(n = 0 至 3)

TCn ^{注 1}	通过DMA通道n是否完成 状态标志指示DMA是否传送
0	DMA传送未完成。
1	DMA传送完成。
DMA传送最后设定为1，当读取时清为0。	

INITn ^{注 2}	DMA传送禁止(Enn位= 0)时，如果INITn位设定为1， 能够初始化DMA传送状态。 在DMA传送完成(TCn位设定为1之前)之前重新设定DMA 传送状态(重新设定DDAnH, DDAnL, DSAnH, DSAnL, DBCn, 以及DADCn寄存器)，务必初始化DMA通道。 当初始化DMA控制器时，然而，务必阅读22.13 注意事项 中的步骤描述。

STGn ^{注 2}	这是DMA传送的软件开始触发。 如果在DMA传送允许状态(TCn 位 = 0, Enn 位 = 1)时此位被置为1， DMA传送开始。

Enn	通过DMA通道n允许或禁止 设定DMA是否传送
0	DMA 传送禁止
1	DMA 传送允许
当Enn位被设置为1时，DMA传送允许。 当DMA传送完成时(产生一个终端计数)，此位自动清为0。 通过软件将Enn位清为0，中断DMA传送。如要恢复，则重新将Enn位设置为1。 当终止或恢复DMA传送时，务必阅读22.13 注意事项 中的步骤描述。	

注 1. TCn 位只读。

2. INITn 和 STGn 位只写。

注意事项 1. 务必将 DCHCn 寄存器的位 6 至位 3 清零。

2. 当 DMA 传输完成（产生终端计数时）时，Enn 位被清为 0 且随后将 TCn 位设为 1。如果 DCHCn 寄存器更新期间读取它的各位，则可能会读取表示“传输未完成并禁止传输”的值（TCn 位 = 0 且 Enn 位 = 0）。

(6) DMA 触发因素寄存器 0 至 3 (DTFR0 至 DTFR3)

DTFR0 至 DTFR3 寄存器是 8 位寄存器，它通过来自片上外围 I/O 的中断请求信号控制 DMA 传送开始触发。

这些寄存器设置的中断请求信号用作 DMA 传送开始因素。

这些寄存器可以按 8 位读取或写入。不过，仅 DFn 位可以按 1 位为单位进行读写。

复位后这些寄存器设置为 00H。

复位后: 00H R/W 地址: DTFR0 FFFFF810H, DTFR1 FFFFF812H,
DTFR2 FFFFF814H, DTFR3 FFFFF816H

	<7>	6	5	4	3	2	1	0
DTFRn	DFn	0	IFCn5	IFCn4	IFCn3	IFCn2	IFCn1	IFCn0

(n = 0 至 3)

DFn ^注	DMA 发送请求状态标志
0	无DMA 发送请求
1	DMA 发送请求

注 切勿用软件将 DFn 位设置为 1。当 DMA 传送被禁止时，如果发生了一个被指定是导致开始 DMA 传送的原因的中断，则将 0 写入该位，以清除 DMA 传送请求。

注意事项 1. 当 DMA 传送禁止 (DCHCn.Enn 位 = 0) 时，按以下时序设置 IFCn5 至 IFCn0 位。

- 从复位后到第一个 DMA 传送开始期间
- 从通道被 DCHCn.INITn 位初始化后到 DMA 传送开始期间
- 从 DMA 传送完成后 (DCHCn.TCn 位 = 1) 到下一个 DMA 传送开始期间

2. 在待机模式 (IDEL1, IDLE2, STOP 或 sub-IDLE 模式) 中产生的中断请求信号不能启动 DMA 传送循环 (也不将 DFn 位置 1)。

3. 通过 IFCn5 至 IFCn0 位选定 DMA 开始因素，当选定的片上外围 I/O 中断产生时，将 DFn 位置 1，而不论 DMA 传输是允许还是禁止。如果这种状态下 DMA 使能，则 DMA 传送立即开始。

备注 关于 IFCn5 至 IFCn0 位，请参见表 22-1 DMA 开始因素

表 22-1. DMA 开始因素 (1/2)

IFCn5	IFCn4	IFCn3	IFCn2	IFCn1	IFCn0	中断源
0	0	0	0	0	0	禁止中断产生 DMA 请求
0	0	0	0	0	1	INTP02
0	0	0	0	1	0	INTP05
0	0	0	0	1	1	INTP09
0	0	0	1	0	0	INTP10
0	0	0	1	0	1	INTP13
0	0	0	1	1	0	INTP16
0	0	0	1	1	1	INTTAB0OV
0	0	1	0	0	0	INTTAB0CC0
0	0	1	0	0	1	INTTAB0CC1
0	0	1	0	1	0	INTTAB0CC2
0	0	1	0	1	1	INTTAB0CC3
0	0	1	1	0	0	INTTAB1OV_BASE [※]
0	0	1	1	0	1	INTTAB1CC0
0	0	1	1	1	0	INTTAB1CC1
0	0	1	1	1	1	INTTAB1CC2
0	1	0	0	0	0	INTTAB1CC3
0	1	0	0	0	1	INTTT0OV
0	1	0	0	1	0	INTTT0CC0
0	1	0	0	1	1	INTTT0CC1
0	1	0	1	0	0	INTTAA0OV
0	1	0	1	0	1	INTTAA0CC0
0	1	0	1	1	0	INTTAA0CC1
0	1	0	1	1	1	INTTAA1OV
0	1	1	0	0	0	INTTAA1CC0
0	1	1	0	0	1	INTTAA1CC1
0	1	1	0	1	0	INTTAA2CC0
0	1	1	0	1	1	INTTAA2CC1
0	1	1	1	0	0	INTTAA3CC0
0	1	1	1	0	1	INTTAA3CC1
0	1	1	1	1	0	INTTAA4CC0
0	1	1	1	1	1	INTTAA4CC1
1	0	0	0	0	0	INTTAA5CC0
1	0	0	0	0	1	INTTAA5CC1
1	0	0	0	1	0	INTTM0EQ0
1	0	0	0	1	1	INTTM1EQ0
1	0	0	1	0	0	INTTM2EQ0
1	0	0	1	0	1	INTTM3EQ0
1	0	0	1	1	0	INTCF0R/INTIIC1
1	0	0	1	1	1	INTCF0T
1	0	1	0	0	0	INTCF1R
1	0	1	0	0	1	INTCF1T

注 INTTAB1OV_BASE 为中断信号，它通过 TMQOP 在 TAB1 的溢出中断 (INTTAB1OV) 之前提取。

表 22-1. DMA 开始因素 (2/2)

IFCn5	IFCn4	IFCn3	IFCn2	IFCn1	IFCn0	中断源
1	0	1	0	1	0	INTCF2R
1	0	1	0	1	1	INTCF2T
1	0	1	1	0	0	INTCF3R
1	0	1	1	0	1	INTCF3T
1	0	1	1	1	0	INTCF4R
1	0	1	1	1	1	INTCF4T
1	1	0	0	0	0	INTUC0R
1	1	0	0	0	1	INTUC0T
1	1	0	0	1	0	INTUC1R/INTIIC2
1	1	0	0	1	1	INTUC1T
1	1	0	1	0	0	INTUC2R
1	1	0	1	0	1	INTUC2T
1	1	0	1	1	0	INTUC3R/INTIIC0
1	1	0	1	1	1	INTUC3T
1	1	1	0	0	0	INTUC4R
1	1	1	0	0	1	INTUC4T
1	1	1	0	1	0	INTAD
1	1	1	0	1	1	INTKR
1	1	1	1	0	0	INTRTC1

备注 n = 0 至 3

<R>

(7) 外部 DMA 请求使能寄存器 (EXDRQEN)

当通过 $\overline{\text{UDMARQm}}/\overline{\text{UDMAAKm}}$ 引脚 ($m = 0, 1$) 连接到外部 USB 设备时, EXDRQEN 寄存器将 DMA 请求设置到各 DMA 通道。

该寄存器可按 8 位为单位进行读写。

复位后本寄存器被设置为 00H。

复位后: 00H R/W 地址: FFFFFFF60H

	7	6	5	4	3	2	1	0
EXDRQEN	RQ3EX1E	RQ2EX1E	RQ1EX1E	RQ0EX1E	RQ3EX0E	RQ2EX0E	RQ1EX0E	RQ0EX0E

RQnEX1E	DMA通道n的分配 ($n = 0$ 至3)
0	不要将DMA通道n分配给 $\overline{\text{UDMARQ1}}/\overline{\text{UDMAAK1}}$ 引脚
1	将DMA通道n分配给 $\overline{\text{UDMARQ1}}/\overline{\text{UDMAAK1}}$ 引脚

RQnEX0E	DMA 通道n的分配 ($n = 0$ 至3)
0	不要将DMA通道n分配给 $\overline{\text{UDMARQ0}}/\overline{\text{UDMAAK0}}$ 引脚
1	将DMA通道n分配给 $\overline{\text{UDMARQ0}}/\overline{\text{UDMAAK0}}$ 引脚

注意事项 1. 禁止将多 DMA 通道分配给 $\overline{\text{UDMARQ1}}/\overline{\text{UDMAAK1}}$ 引脚 (禁止同时将 RQ3EX1E, RQ2EX1E, RQ1EX1E 和 RQ0EX1E 位设置到 $\overline{\text{UDMARQ1}}/\overline{\text{UDMAAK1}}$ 引脚)。

2. 禁止将多 DMA 通道分配给 $\overline{\text{UDMARQ0}}/\overline{\text{UDMAAK0}}$ 引脚 (禁止同时将 RQ3EX0E, RQ2EX0E, RQ1EX0E 和 RQ0EX0E 位设置到 $\overline{\text{UDMARQ0}}/\overline{\text{UDMAAK0}}$ 引脚)。

3. 禁止将 $\overline{\text{UDMARQ1}}/\overline{\text{UDMAAK1}}$ 引脚和 $\overline{\text{UDMARQ0}}/\overline{\text{UDMAAK0}}$ 引脚分配到同一 DMA 通道 (禁止同时设置 RQ3EX1E 和 RQ3EX0E, RQ2EX1E 和 RQ2EX0E, RQ1EX1E 和 RQ1EX0E 以及 RQ0EX1E 和 RQ0EX0E 位)。

4. 通过设置 EXDRQEN 寄存器使用来自外源的 DMA 请求时, 禁止将 DTFRn.IFCn5-IFCn0 位设为 000000 (以通过中断禁止 DMA 请求)。

关于详情, 参见 22.3 (6) DMA 触发因素寄存器 0 至 3 (DTFR0 至 DTFR3)。

22.4 传送目标

表 22-2 说明了传送目标之间的关系（√：允许传送，×：禁止传送）

表 22-2 传送目标间的关系

		传送目的地			
		内部 ROM	片上外围 I/O	内部 RAM	外部存储器
传送源	片上外围 I/O	×	√	√	√
	内部 RAM	×	√	×	√
	外部存储器	×	√	√	√
	内部 ROM	×	×	×	×

注意事项 在表 22-2 中，用标有“×”的传送目的地和传送源组合的操作将得不到保证。

22.5 传送模式

支持单传送模式。

在单传送模式下，在每次字节/半字传送时总线被释放。 如果有一个后续的 DMA 传送请求，则将再执行一次传送。该操作一直进行下去，直到发生一个终止计数。

当 DMAC 释放总线后，如果发出了另一个更高优先级的 DMA 传送请求，则较高优先级的 DMA 请求将优先被处理。

如果在一个传送周期中产生了同一通道的一个新传送请求和另一个低优先级通道的传送请求，则当总线被释放给 CPU（同一通道的新传送请求在传送周期中被忽略）后，具有低优先级的通道的 DMA 传送被执行。

22.6 传送类型

支持双周期传送。

在双周期传送中，数据传送在两个周期中完成，即读周期和写周期。

在读周期中，传送源地址被输出，并执行从源地址向 DMAC 的读操作。 在写周期中，传送目的地址被输出，并执行从 DMAC 向目的地的写操作。

在读周期和写周期之间总是插入一个时钟空闲周期。 在进行双周期的 DMA 传送时，如果传送源与目的地的数据总线宽度不同，则执行以下操作。

<16 位数据传输>

- <1> 从 32 位总线 → 16 位总线传送
先产生一个读周期（高 16 位处于高阻状态），接着产生一个写周期（16 位）。
- <2> 从 16/32 位总线向 8 位总线传送
产生一次 16 位读周期，然后产生两次 8 位写周期。
- <3> 从 8 位总线向 16/32 位总线传送
先产生两次 8 位读周期，然后产生一次 16 位写周期。
- <4> 16 位总线 与 32 位总线之间传送
产生一次 16 位读周期，然后产生一次 16 位写周期。

对于在一个片上外围 I/O 寄存器（传送源或目的地）上的 DMA 传送上执行的，请务必指定与寄存器单位相同的传送单位。 例如，对于向 8 位寄存器发送的 DMA 传送，务必指定字节（8 位）传送。

- 备注** 每个传送目标（传送源/目的地）的总线宽度如下
- 片上外围 I/O: 16 位总线宽度
 - 内部 RAM: 32 位总线宽度
 - 外部存储器: 8 位或 16 位总线宽度

22.7 DMA通道优先级

DMA 通道优先级固定如下所示：

DMA 通道 0 > DMA 通道 1 > DMA 通道 2 > DMA 通道 3

对于每个传送周期，这些优先级都要检查。

22.8 与DMA传送有关的时间

响应 DMA 请求所需的时间、DMA 传送需要的最少时钟数如下所示：

单次传送： DMA 响应时间（<1>）+ 传送源存储器访问（<2>）+ 1^{注1} + 传输目的地存储器访问（<2>）

DMA 周期		最少执行时钟数
<1> DMA 请求响应时间		4 时钟（最少）+ 噪声消除时间 ^{注2}
<2> 存储器访问	外部存储器访问	取决于连接的存储器
	内部 RAM 访问	2 个时钟 ^{注3}
	外围 I/O 寄存器访问	3 个时钟+VSWC 寄存器规定的等待周期数 ^{注4}

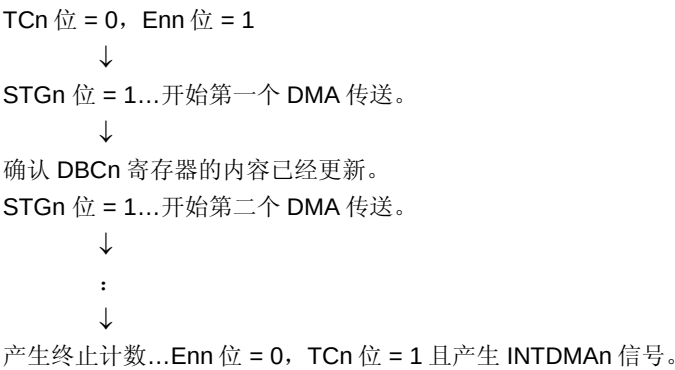
- 注
1. 在 DMA 传送时，在读周期和写周期之间总是插入一个时钟。
 2. 如果指定一个外部中断（INTPn）作为开始 DMA 传送的触发条件，则增加噪声消除时间（n = 00 至 18）。
 3. 一个 DMA 周期需要两个时钟。
 4. 访问某个特定的外围 I/O 寄存器，需要更多等待周期（详细情况，请参见 3.4.9（2））。

22.9 DMA 传送开始条件

DMA 传送开始条件有两种，介绍如下：

(1) 软件请求

当 DCHCn.TCn 位 = 1 且 Enn 位 = 1（允许 DMA 传送）时，如果将 STGn 位设置为 1，则开始 DMA 传送。
如果紧接着前一个传送请求下一个 DMA 传送周期，则应使用 DBCn 寄存器，确认前一个 DMA 传送周期已经完成，并再次将 STGn 位设置为 1（n = 0 至 3）。



(2) 片上外围 I/O 请求

当 DCHCn.TCn 位 = 0 且 Enn 位 = 1（允许 DMA 传送）时，如果从 DTFRn 寄存器设置的片上外围 I/O 产生一个中断请求，则 DMA 传送将开始。

- 注意事项**
1. 两个开始条件（软件触发和硬件触发）不能同时用于一个 DMA 通道。 如果同时产生两个开始因素，则只有其中一个有效。 且不能确认有效的开始条件。
 2. 在前一个 DMA 传送请求产生后产生或在前一个 DMA 传送周期中间产生的新传送请求将被忽略（清除）。
 3. 同一个 DMA 通道的传送请求时间间隔根据 DMA 传送周期中的总线等待的设置、其他通道的开始状态或外部总线保持请求的情况而有所变化。 特别是，正如注意事项 2 中所述，在 DMA 传送周期前或在 DMA 传送周期中间产生的新传送请求将被忽略。 因此，系统的同一 DMA 通道的传送请求时间间隔必须足够长。 当使用软件触发时，可以通过更新 DBCn 寄存器来检查之前产生的 DMA 传送周期的完成情况。

22.10 DMA 中止条件

如果发生总线保持，则 DMA 传送将中止。

如果在内存/片上外围 I/O 和内存/片上外围 I/O 之间传送，也会发生 DMA 传送中止。

当总线保持被清除，则 DMA 传送将被恢复。

22.11 DMA 传送结束

当 DMA 传送已经完成 DBCn 寄存器设置的次数，且 DCHCn.Enn 位被清零、TCn 位被置 1 后，将向中断控制器 (INTC) 产生一个 DMA 传送结束中断请求信号 (INTDMAn) (n = 0 至 3)。

V850ES/JG3-H 和 V850ES/JH3-H 不将终止计数信号输出到外部设备。因此，要使用 DMA 传送结束中断或轮流检测 TCn 位来确认 DMA 传送的完成。

22.12 操作时序

图 22-1 至 22-4 显示了 DMA 操作时序。

图 22-1. DMA 的优先级 (1)

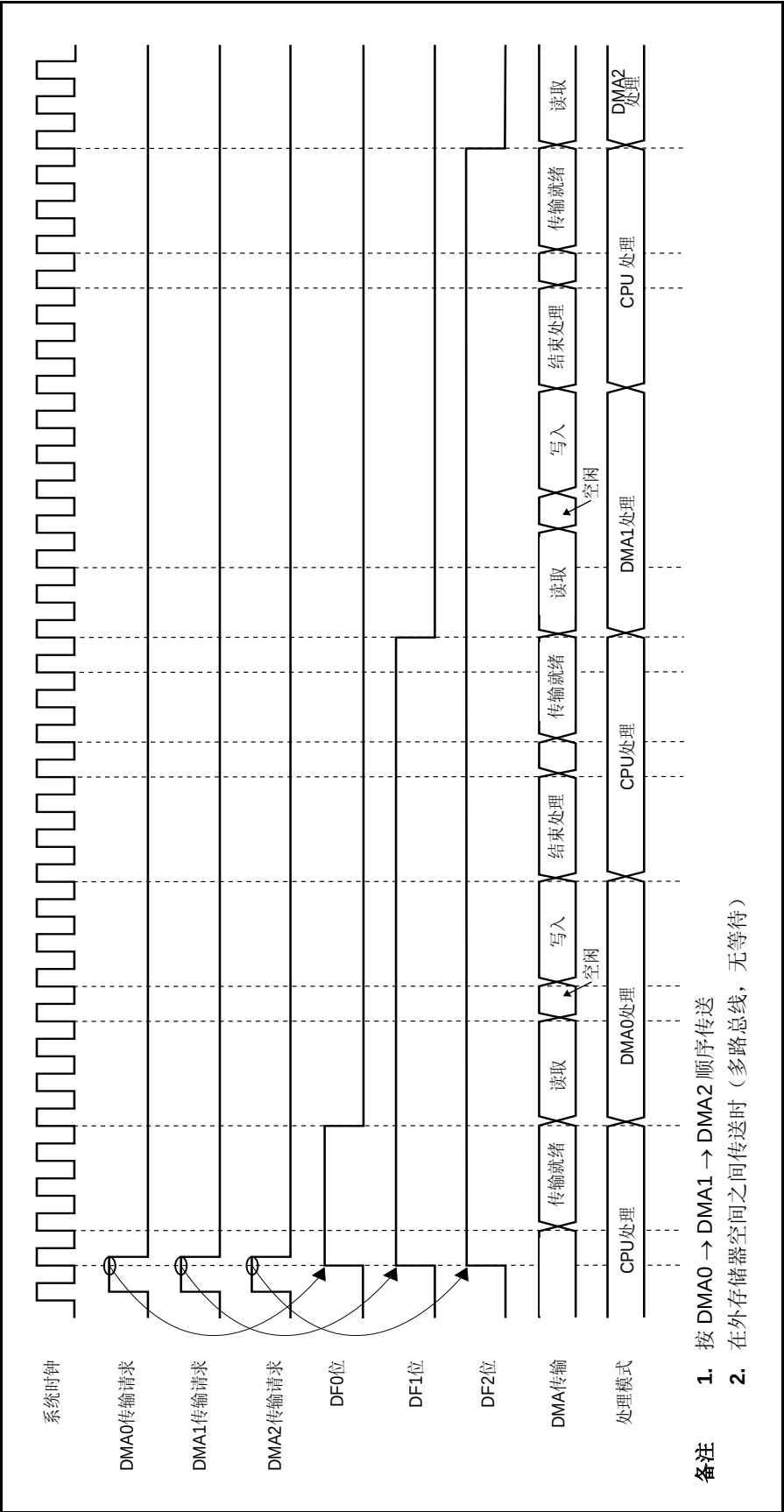


图 22-2. DMA 的优先级 (2)

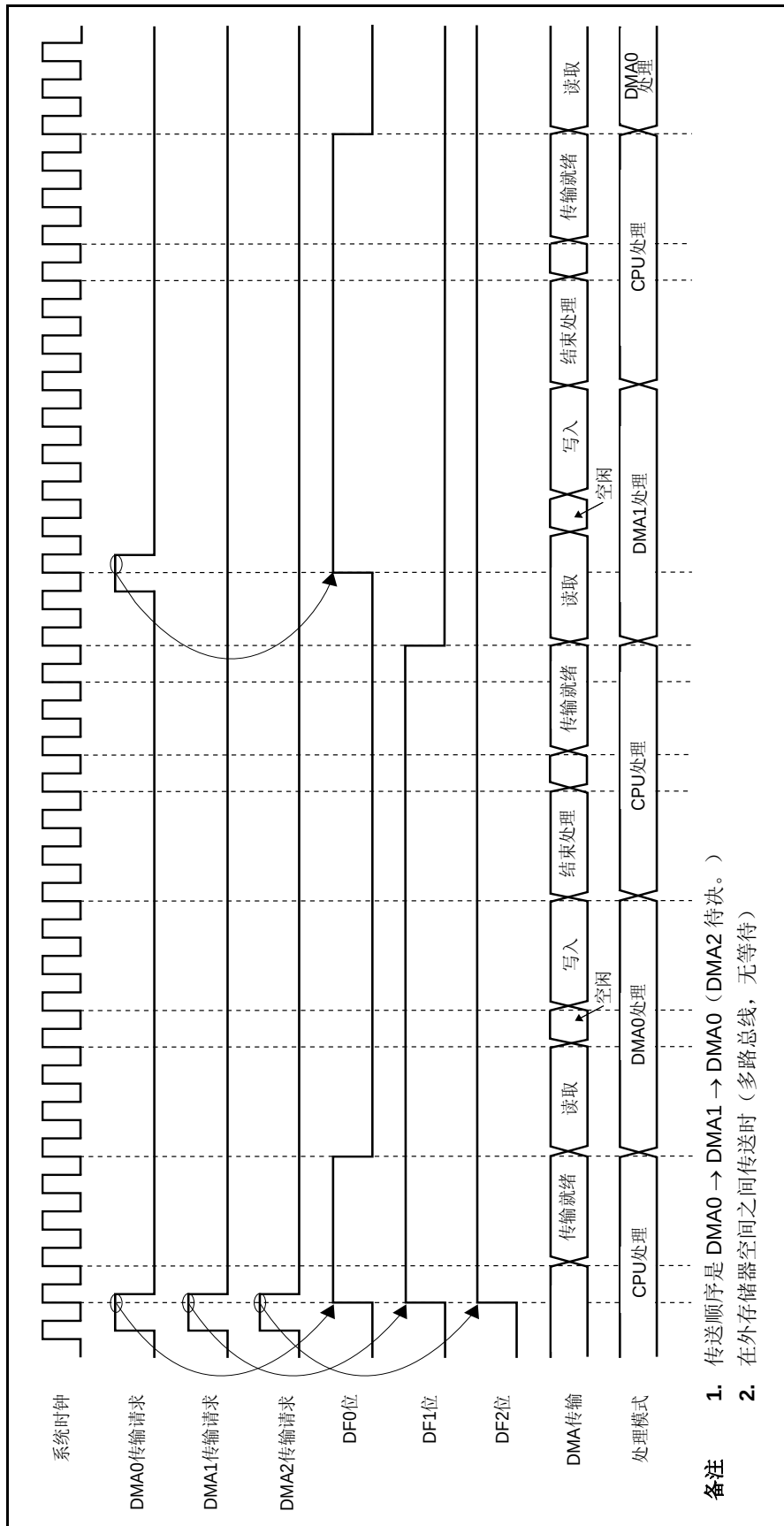


图 22-3 DMA 传送请求被忽略的周期（1）

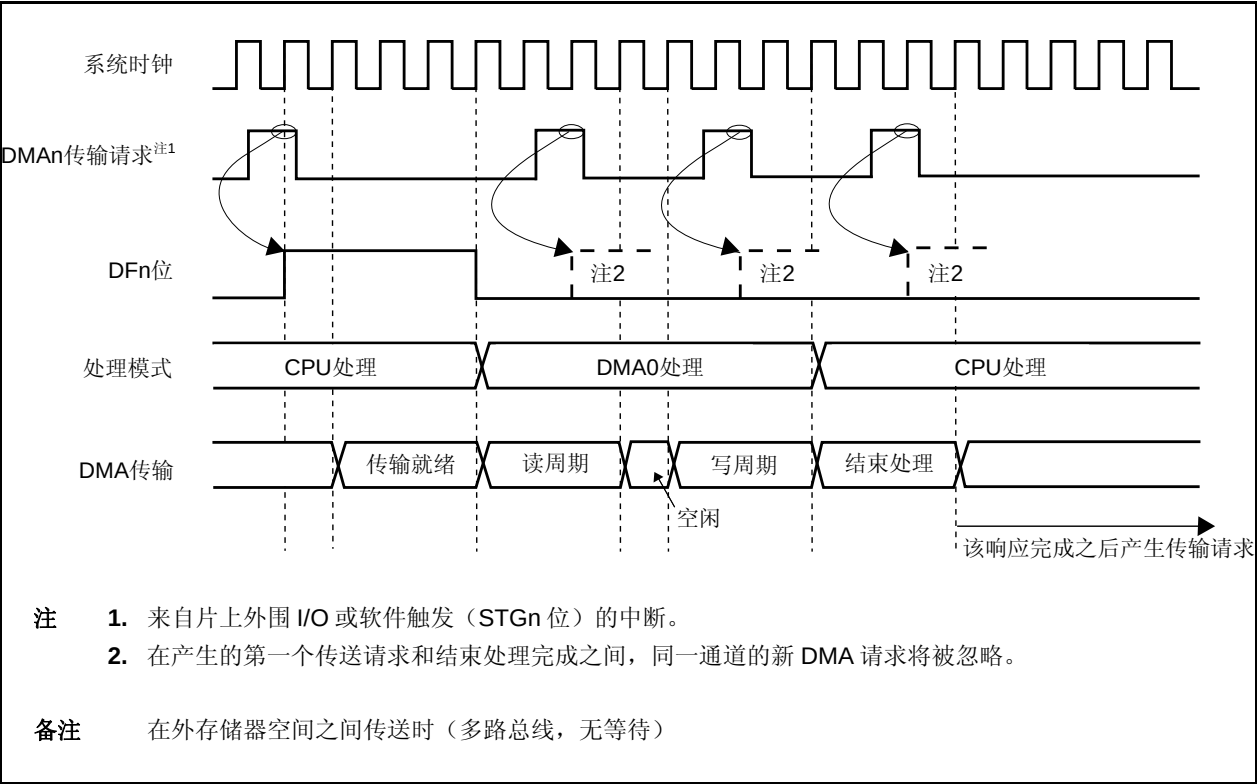
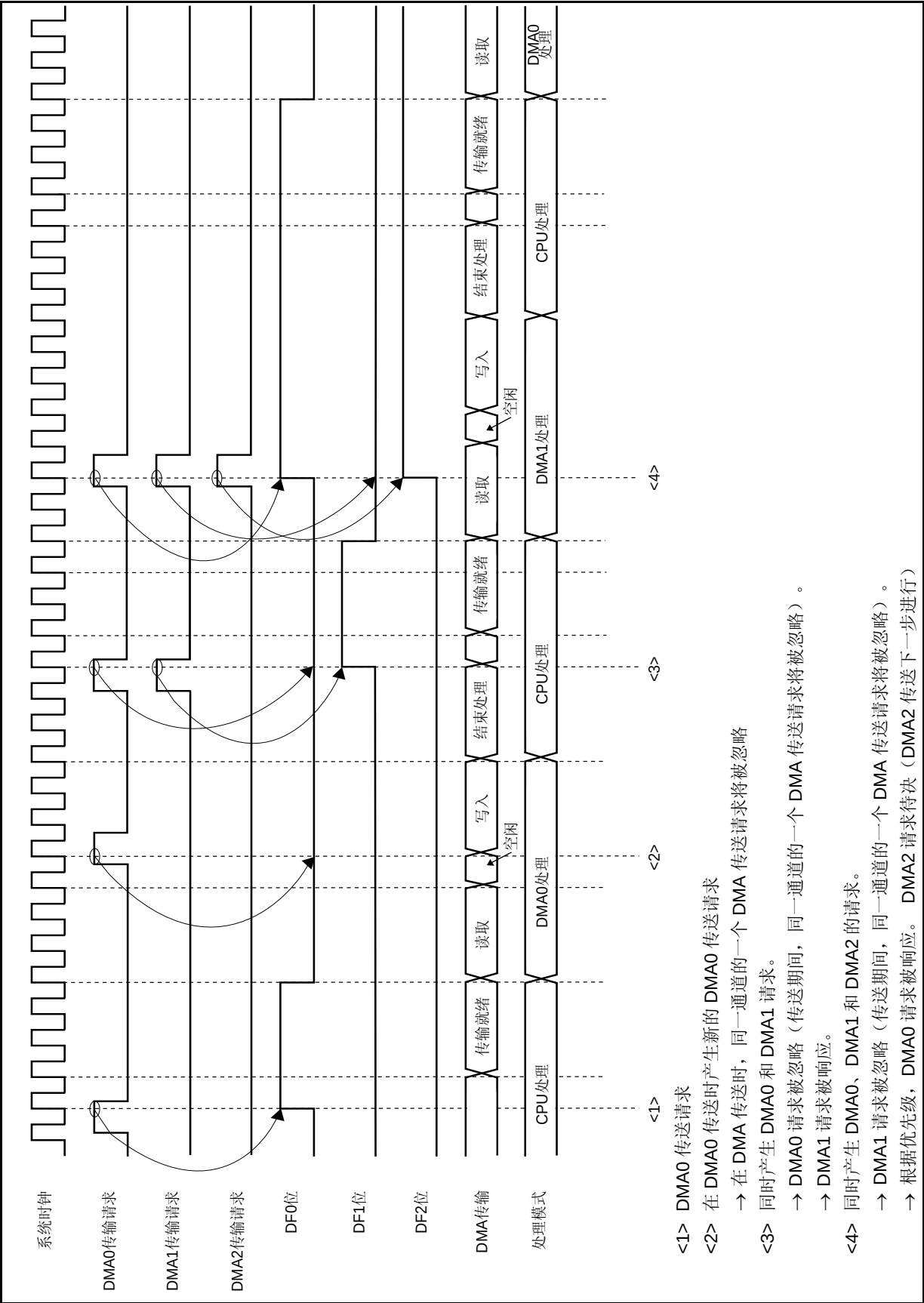


图 22-4 DMA 传送请求被忽略的周期 (2)



22.13 注意事项

(1) VSWC 寄存器注意事项

在使用 DMAC 时，确保根据工作频率为 VSWC 寄存器设置一个合适的值。

如果使用了 VSWC 寄存器的一个默认值 (77H)，或者给 VSWC 寄存器设置了一个不适当的值，则操作将不能正确执行 (关于寄存器的详细情况，请参见 3.4.9 (1) (a) 系统等待控制寄存器 (VSWC))。

(2) 在内部 RAM 中执行的 DMA 传送注意事项

当在内部 RAM 中执行以下指令时，切勿执行与内部 RAM (传送源/目的地) 之间进行数据往来的 DMA 传送，因为其后 CPU 可能会发生工作错误。

- 位于内部 RAM 中的位操作指令 (SET1, CLR1, 或 NOT1)。
- 内部 RAM 中访问错位地址的数据访问指令。

反之，当执行与内部 RAM (传送源/目的地) 间数据往来的 DMA 传送时，切勿执行以上两条指令。

(3) 读 DCHCn.TCn 位 (n = 0 至 3) 注意事项

当读取 TCn 位时会将其清除为 0，但是即使在指定时刻读取，该位也不会自动清除。要准确地将 TCn 位清零，请增加以下处理操作

(a) 通过轮询 TCn 位等待 DMA 传送完成时

确认 TCn 位已经被设置为 1 (在 TCn 位 = 1 被读后)，然后读 TCn 位三次以上。

(b) 当读取中断服务例程中的 TCn 位时

执行读 TCn 位三次。

(4) DMA 传送初始化方法 (将 DCHCn.INITn 位设置为 1)

当执行 DMA 传送的通道要进行初始化时, 即使 INITn 位被设置为 1, 通道也不可以被初始化。要准确地初始化通道, 执行以下两种方法之一。

(a) 暂时停止所有 DMA 通道的传送

使用以下<1>至<7>步, 将执行 DMA 传送的通道初始化。

不过, 请注意, 当第<5>步被执行时, TCn 位将被清零。确保其他处理程序不要求 TCn 位为 1。

<1> 禁止中断 (DI)。

<2> 读非强制终止的 DMA 通道的 DCHCn.Enn 位, 然后将值传送到通用寄存器。

<3> 将所用 DMA 通道 (包括强制终止的通道) 的 Enn 位清除为 0。要清除最后一个 DMA 通道的 Enn 位, 需执行两次清除指令。如果 DMA 传送目标 (传送源/目的地) 是内部 RAM, 则执行指令三次。

示例: 如果通道 0, 1, 2 被使用, 则按以下顺序执行指令 (如果传送目标不是内部 RAM)。

- 将 DCHC0.E00 位清 0
- 将 DCHC1.E11 位清 0
- 将 DCHC2.E22 位清 0
- 将 DCHC2.E22 位再次清 0

<4> 将被强制终止的通道的 INITn 位设置为 1。

<5> 读未强制终止的每个通道的 TCn 位。如果在第<2>步中读的 TCn 位和 Enn 位均为 1 (逻辑乘 (AND) 结果是 1), 将被保存的 Enn 位清零。

<6> 完成第<5>步的操作后, 将 Enn 位值写入 DCHCn 寄存器。

<7> 启用中断 (EI)。

注意事项 务必执行上述的第<5>步, 以防止非法设置在第<2>和<3>步之间正常完成 DMA 传送的通道的 Enn 位。

(b) 重复执行 INITn 位设置，直到传送被正确地强制终止

- <1> 抑制来自待强制终止通道的 DMA 请求源的请求（停止片上外围 I/O 的操作）。
- <2> 使用 DTFRn.DFn 位，确认待强制终止的通道的 DMA 传送请求未处于待决状态。如果 DMA 传输请求处于待决状态，请等待直到其执行处理完成。
- <3> 确认待强制终止的通道的 DMA 传送请求未处于待决状态后，将 Enn 位清零。
- <4> 再次将待强制终止的通道的 Enn 位清零。
如果待强制终止的通道的传送目标（传送源/目的地）是内部 RAM，则再次执行该操作。
- <5> 将强制终止的通道的传送的初始数目拷贝到通用寄存器。
- <6> 将被强制终止的通道的 INITn 位设置为 1。
- <7> 读待强制终止的通道的 DBCn 寄存器的值，然后将其与在第<5>步中拷贝的值进行比较。如果两个值不匹配，重复操作第<6>和<7>步。

- 备注**
- 1. 当在第<7>步中读 DBCn 寄存器的值时，如果正确完成强制终止，则读传送的初始数目。否则，读传送的剩余数目。
 - 2. 注意，如果应用软件频繁地使用非强制终止的 DMA 通道的一个通道的 DMA 传送，则方法 (b) 会消耗较长时间。

(5) 暂时停止 DMA 传送的步骤（清除 Enn 位）

使用以下步骤停止和恢复被处理的 DMA 传送：

- <1> 抑制来自 DMA 请求源的传送请求（停止片上外围 I/O 的操作）。
- <2> 使用 DFn 位（检查 DFn 位是否等于 0），确认 DMA 传送请求未处于待决状态。
如果请求处于待决状态，请等待直到待决 DMA 传送请求处理完成。
- <3> 如果确认没有 DMA 传送请求处于待决状态，将 Enn 位清零（该操作停止 DMA 传送）。
- <4> 将 Enn 位设置为 1，以恢复 DMA 传送。
- <5> 恢复已被停止的 DMA 请求源的操作（开始片上外围 I/O 的操作）。

(6) 存储边界

在 DMA 传送过程中，如果传送源或目的地的地址超出了 DMA 目标（外存储器、内部 RAM 或片上外围 I/O）的区域，则操作将得不到保证。

(7) 传送错位数据

不支持采用 16 位总线进行错位数据的 DMA 传送。

如果一个奇地址被指定为传送源或目的地，则地址的最低有效位被强制假定为 0。

(8) CPU 总线仲裁

因为 DMA 控制器拥有比 CPU 更高优先级的总线权限，在 DMA 传送时，发生的 CPU 访问将保持待决状态直到 DMA 周期完成，总线被释放给 CPU 为止。

不过，CPU 可以访问不参与 DMA 传送的外存储器、内部外设 I/O 及内部 RAM。

- 当外存储器与片上外围 I/O 之间执行 DMA 传送时，CPU 可以访问 ROM 和内部 RAM。
- 当在外部存储器之间进行 DMA 传输时，CPU 可以访问内部 ROM 和内部外设 I/O。

(9) DMA 操作时不可重写的寄存器/位

当未进行 DMA 操作时，按以下时序设置以下寄存器。

[寄存器]

- DSAnH, DSAnL, DDA nH, DDA nL, DBCn 和 DADCn 寄存器
- DTFRn.IFCn5 至 DTFRn.IFCn0 位

[可设置的时序]

- 从复位后到第一个 DMA 传送开始期间
- 从通道初始化后到 DMA 传送开始期间
- 从 DMA 传送完成后 (TCn 位 = 1) 到下一个 DMA 传送开始期间

(10) 务必将以下寄存器位设置为 0。

- DSAnH 寄存器的位 14 至位 10
- DDA nH 寄存器的位 14 至位 10
- 寄存器 DADCn 的位 15, 13 至 8 和位 3 至 0。
- DCHCn 寄存器的位 6 至位 3

(11) DMA 开始条件

不要使用相同开始条件开启两个或更多 DMA 通道。如果使用相同的开始因素启动了两个或更多的 DMA 通道，那么已经设定的 DMA 通道会再启动，或具有较低优先级的 DMA 通道将在具有较高优先级的 DMA 通道之前被应答， 这种操作不能得到保障。

(12) 读 DSA_n 和 DDA_n 寄存器的值

在 DMA 传送过程中，可以从 DSA_n 和 DDA_n 寄存器中读正在被更新当中的值 (n = 0 至 3)。

例如，当 DMA 传送源地址 (DSA_n 寄存器) 为 0000FFFFH，且计数方向是递增 (DADC_n.SAD1 和 DADC_n.SAD0 位 = 00) 时，如果先读 DSA_nH 寄存器，然后读 DSA_n 寄存器，则根据是否在读 DSA_nH 寄存器后立即执行 DMA 传送，DSA_nL 寄存器的值将有以下不同。

(a) 读 DSA_n 寄存器时不发生 DMA 传送

<1> 读 DSA_nH 寄存器的值: DSA_nH = 0000H

<2> 读 DSA_nL 寄存器的值: DSA_nL = FFFFH

(b) 读 DSA_n 寄存器时发生 DMA 传送

<1> 读 DSA_nH 寄存器的值: DSA_nH = 0000H

<2> 发生 DMA 传送

<3> 递增 DSA_n 寄存器: DSA_n = 00100000H

<4> 读 DSA_nL 寄存器的值: DSA_nL = 0000H

第二十三章 中断/异常处理功能

V850ES/JG3-H 和 V850ES/JH3-H 单片机为中断服务提供专用的中断控制器（INTC），可以处理总数达 86 至 93 个中断请求。

所谓中断，是其发生独立于程序执行的事件，而异常则是其发生与程序执行相关的事件。

V850ES/JG3-H 和 V850ES/JH3-H 单片机能够处理来自于片上外围硬件设备和来自于外部中断源的中断请求信号。不仅如此，还可以通过 TRAP 指令（软件异常）或产生一个异常事件（也就是取非法操作码）（异常陷阱）来启动异常处理。

23.1 特性

○ 中断

表 23-1. V850ES/JG3-H 和 V850ES/JH3-H 的中断

		内部			外部		
		不可屏蔽中断	可屏蔽中断	总计	不可屏蔽中断	可屏蔽中断	总计
V850ES/JG3-H	μPD70F3760	1	68	69	1	16	17
	μPD70F3761	1	68	69	1	16	17
	μPD70F3762	1	68	69	1	16	17
	μPD70F3770	1	72	73	1	16	17
V850ES/JH3-H	μPD70F3765	1	68	69	1	19	20
	μPD70F3766	1	68	69	1	19	20
	μPD70F3767	1	68	69	1	19	20
	μPD70F3771	1	72	73	1	19	20

- 8 级可编程优先级（可屏蔽中断）。
- 依照优先级多重中断控制。
- 可以为每一个可屏蔽中断请求指定屏蔽。
- 噪声抑制、边沿检测以及为外部中断请求信号指定有效沿。

○ 异常

- 软件异常：32 个产生源
- 异常陷阱：2 个产生源（非法操作码异常）

V850ES/JG3-H 和 V850ES/JH3-H 的中断/异常源分别在 表 23-2 和 23-3 中列出。

表 23-2. V850ES/JG3-H 中断源 (1/4)

类型	类别	默认优先级	名称	触发	产生单元	异常码	处理程序地址	恢复 PC	中断控制寄存器
复位	中断	—	RESET	RESET 引脚输入 由内部源复位	RESET	0000H	00000000H	未定义	—
不可屏蔽	中断	—	NMI	NMI 引脚有效沿输入	引脚	0010H	00000010H	下一 PC 值	—
		—	INTWDT2	WDT2 溢出	WDT2	0020H	00000020H	注 1	—
软件异常	异常	—	TRAP0n ^{注 2}	TRAP 指令	—	004nH ^{注 2}	00000040H	下一 PC 值	—
		—	TRAP1n ^{注 2}	TRAP 指令	—	005nH ^{注 2}	00000050H	下一 PC 值	—
异常陷阱	异常	—	ILGOP/ DBG0	非法操作码/DBTRAP 指令	—	0060H	00000060H	下一 PC 值	—
可屏蔽	中断	0	INTLVI	低电压检测	POCLVI	0080H	00000080H	下一 PC 值	LVIIC
		3	INTP02	外部中断引脚输入边沿检测 (INTP02)	引脚	00B0H	000000B0H	下一 PC 值	PIC02
		4	INTP03	外部中断引脚输入边沿检测 (INTP03)	引脚	00C0H	000000C0H	下一 PC 值	PIC03
		5	INTP04	外部中断引脚输入边沿检测 (INTP04)	引脚	00D0H	000000D0H	下一 PC 值	PIC04
		6	INTP05	外部中断引脚输入边沿检测 (INTP05)	引脚	00E0H	000000E0H	下一 PC 值	PIC05
		8	INTP07	外部中断引脚输入边沿检测 (INTP07)	引脚	0100H	00000100H	下一 PC 值	PIC07
		9	INTP08	外部中断引脚输入边沿检测 (INTP08)	引脚	0110H	00000110H	下一 PC 值	PIC08
		10	INTP09	外部中断引脚输入边沿检测 (INTP09)	引脚	0120H	00000120H	下一 PC 值	PIC09
		11	INTP10	外部中断引脚输入边沿检测 (INTP10)	引脚	0130H	00000130H	下一 PC 值	PIC10
		12	INTP11	外部中断引脚输入边沿检测 (INTP11)	引脚	0140H	00000140H	下一 PC 值	PIC11
		13	INTP12	外部中断引脚输入边沿检测 (INTP12)	引脚	0150H	00000150H	下一 PC 值	PIC12
		14	INTP13	外部中断引脚输入边沿检测 (INTP16)	引脚	0160H	00000160H	下一 PC 值	PIC13
		15	INTP14	外部中断引脚输入边沿检测 (INTP14)	引脚	0170H	00000170H	下一 PC 值	PIC14
		16	INTP15	外部中断引脚输入边沿检测 (INTP15)	引脚	0180H	00000180H	下一 PC 值	PIC15
		17	INTP16	外部中断引脚输入边沿检测 (INTP16)	引脚	0190H	00000190H	下一 PC 值	PIC16

注 1. 关于 INTWDT2 中断恢复的情况, 请参见 23.2.2 (2) INTWDT2 的信号。

2. n = 0 至 FH

表 23-2. V850ES/JG3-H 中断源 (2/4)

类型	类别	默认优先级	名称	触发	产生单元	异常码	处理程序地址	恢复 PC	中断控制寄存器
可屏蔽	中断	18	INTP17	外部中断引脚输入边沿检测 (INTP17)	引脚	01A0H	000001A0H	下一 PC 值	PIC17
		19	INTP18	外部中断引脚输入边沿检测 (INTP18)	引脚	01B0H	000001B0H	下一 PC 值	PIC18
		20	INTTAB0OV	TAB0 溢出	TAB0	01C0H	000001C0H	下一 PC 值	TAB0OVIC
		21	INTTAB0CC0	TAB0 捕获 0/ 比较 0 匹配	TAB0	01D0H	000001D0H	下一 PC 值	TAB0CCIC0
		22	INTTAB0CC1	TAB0 捕获 1/ 比较 1 匹配	TAB0	01E0H	000001E0H	下一 PC 值	TAB0CCIC1
		23	INTTAB0CC2	TAB0 捕获 2/ 比较 2 匹配	TAB0	01F0H	000001F0H	下一 PC 值	TAB0CCIC2
		24	INTTAB0CC3	TAB0 捕获 3/ 比较 3 匹配	TAB0	0200H	00000200H	下一 PC 值	TAB0CCIC3
		25	INTTAB1OV ^{注 1}	TAB1 溢出	TAB1	0210H	00000210H	下一 PC 值	TAB1OVIC
		26	INTTAB1CC0 ^{注 2}	TAB1 捕获 0/ 比较 0 匹配	TAB1	0220H	00000220H	下一 PC 值	TAB1CCIC0
		27	INTTAB1CC1	TAB1 捕获 1/ 比较 1 匹配	TAB1	0230H	00000230H	下一 PC 值	TAB1CCIC1
		28	INTTAB1CC2	TAB1 捕获 2/ 比较 2 匹配	TAB1	0240H	00000240H	下一 PC 值	TAB1CCIC2
		29	INTTAB1CC3	TAB1 捕获 3/ 比较 3 匹配	TAB1	0250H	00000250H	下一 PC 值	TAB1CCIC3
		30	INTTT0OV	TMT0 溢出	TMT0	0260H	00000260H	下一 PC 值	TT0OVIC
		31	INTTT0CC0	TMT0 捕获 0/ 比较 0 匹配	TMT0	0270H	00000270H	下一 PC 值	TT0CCIC0
		32	INTTT0CC1	TMT0 捕获 1/ 比较 1 匹配	TMT0	0280H	00000280H	下一 PC 值	TT0CCIC1
		33	INTTT0EC	TMT0 编码器输入	TMT0	0290H	00000290H	下一 PC 值	TT0ECIC
		34	INTTAA0OV	TAA0 溢出	TAA0	02A0H	000002A0H	下一 PC 值	TAA0OVIC
		35	INTTAA0CC0	TAA0 捕获 0/ 比较 0 匹配	TAA0	02B0H	000002B0H	下一 PC 值	TAA0CCIC0
		36	INTTAA0CC1	TAA0 捕获 1/ 比较 1 匹配	TAA0	02C0H	000002C0H	下一 PC 值	TAA0CCIC1
		37	INTTAA1OV	TAA1 溢出	TAA1	02D0H	000002D0H	下一 PC 值	TAA1OVIC
		38	INTTAA1CC0	TAA1 捕获 0/ 比较 0 匹配	TAA1	02E0H	000002E0H	下一 PC 值	TAA1CCIC0
		39	INTTAA1CC1	TAA1 捕获 1/ 比较 1 匹配	TAA1	02F0H	000002F0H	下一 PC 值	TAA1CCIC1
		40	INTTAA2OV	TAA2 溢出	TAA2	0300H	00000300H	下一 PC 值	TAA2OVIC

- 注
1. 在 6-段 PWM 输出模式中使用 TAB1 时, 将其用作来自 TMQOP 的 0 匹配中断 (TAB1TIOD) 请求。
 2. 在 6-段 PWM 输出模式中使用 TAB1 时, 将其用作来自 TMQOP 的比较匹配中断 (TAB1TICD0) 请求。

表 23-2. V850ES/JG3-H 中断源 (3/4)

类型	类别	默认优先级	名称	触发	产生单元	异常码	处理程序地址	恢复 PC	中断控制寄存器
可屏蔽	中断	41	INTTAA2CC0	TAA2 捕获 0/ 比较 0 匹配	TAA2	0310H	00000310H	下一 PC 值	TAA2CCIC0
		42	INTTAA2CC1	TAA2 捕获 1/ 比较 1 匹配	TAA2	0320H	00000320H	下一 PC 值	TAA2CCIC1
		43	INTTAA3OV	TAA3 溢出	TAA3	0330H	00000330H	下一 PC 值	TAA3OVIC
		44	INTTAA3CC0	TAA3 捕获 0/ 比较 0 匹配	TAA3	0340H	00000340H	下一 PC 值	TAA3CCIC0
		45	INTTAA3CC1	TAA3 捕获 1/ 比较 1 匹配	TAA3	0350H	00000350H	下一 PC 值	TAA3CCIC1
		46	INTTAA4OV	TAA4 溢出	TAA4	0360H	00000360H	下一 PC 值	TAA4OVIC
		47	INTTAA4CC0	TAA4 比较 0 匹配	TAA4	0370H	00000370H	下一 PC 值	TAA4CCIC0
		48	INTTAA4CC1	TAA4 比较 1 匹配	TAA4	0380H	00000380H	下一 PC 值	TAA4CCIC1
		49	INTTAA5OV	TAA5 溢出	TAA5	0390H	00000390H	下一 PC 值	TAA5OVIC
		50	INTTAA5CC0	TAA5 捕获 0/ 比较 0 匹配	TAA5	03A0H	000003A0H	下一 PC 值	TAA5CCIC0
		51	INTTAA5CC1	TAA5 捕获 1/ 比较 1 匹配	TAA5	03B0H	000003B0H	下一 PC 值	TAA5CCIC1
		52	INTTM0EQ0	TMM0 比较匹配	TMM0	03C0H	000003C0H	下一 PC 值	TM0EQIC0
		53	INTTM1EQ0	TMM1 比较匹配	TMM1	03D0H	000003D0H	下一 PC 值	TM1EQIC0
		54	INTTM2EQ0	TMM2 比较匹配	TMM2	03E0H	000003E0H	下一 PC 值	TM2EQIC0
		55	INTTM3EQ0	TMM3 比较匹配	TMM3	03F0H	000003F0H	下一 PC 值	TM3EQIC0
		56	INTCF0R /INTIIC1	CSIF0 接收完毕/ CSIF0 接收错误/ IIC1 发送完毕	CSIF0/ IIC1	0400H	00000400H	下一 PC 值	CF0RIC/ IICIC1
		57	INTCF0T	CSIF0 连续发送写使能	CSIF0	0410H	00000410H	下一 PC 值	CF0TIC
		58	INTCF1R	CSIF1 接收完毕/ CSIF1 接收错误/	CSIF1	0420H	00000420H	下一 PC 值	CF1RIC
		59	INTCF1T	CSIF1 连续发送写使能	CSIF1	0430H	00000430H	下一 PC 值	CF1TIC
		60	INTCF2R	CSIF2 接收完毕/ CSIF2 接收错误/	CSIF2	0440H	00000440H	下一 PC 值	CF2RIC
		61	INTCF2T	CSIF2 连续发送写使能	CSIF2	0450H	00000450H	下一 PC 值	CF2TIC
		62	INTCF3R	CSIF3 接收完毕/ CSIF3 接收错误/	CSIF3	0460H	00000460H	下一 PC 值	CF3RIC
		63	INTCF3T	CSIF3 连续发送写使能	CSIF3	0470H	00000470H	下一 PC 值	CF3TIC
		64	INTCF4R	CSIF4 接收完毕/ CSIF4 接收错误/	CSIF4	0480H	00000480H	下一 PC 值	CF4RIC
		65	INTCF4T	C SIF4 连续发送写使能	CSIF4	0490H	00000490H	下一 PC 值	CF4TIC

表 23-2. V850ES/JG3-H 中断源 (4/4)

类型	类别	默认优先级	名称	触发	产生单元	异常码	处理程序地址	恢复 PC	中断控制寄存器
可屏蔽	中断	66	INTUC0R	UARTC0 接收完成/UARTC0 接收错误	UARTC0	04A0H	000004A0H	下一 PC 值	UC0RIC
		67	INTUC0T	UARTC0 连续发送使能	UARTC0	04B0H	000004B0H	下一 PC 值	UC0TIC
		68	INTUC1R/ INTIIC2	UARTC1 接收完毕/ UARTC1 接收错误/ IIC2 发送完毕	UARTC1/ IIC2	04C0H	000004C0H	下一 PC 值	UC1RIC/ IICIC2
		69	INTUC1T	UARTC1 连续发送使能	UARTC1	04D0H	000004D0H	下一 PC 值	UC1TIC
		70	INTUC2R	UARTC2 接收完成/ UARTC2 接收错误	UARTC2	04E0H	000004E0H	下一 PC 值	UC2RIC
		71	INTUC2T	UARTC2 连续发送使能	UARTC2	04F0H	000004F0H	下一 PC 值	UC2TIC
		72	INTUC3R/ INTIIC0	UARTC3 接收完成/UARTC0 接收错误/IIC0 传送完成	UARTC3/ IIC0	0500H	00000500H	下一 PC 值	UC3RIC/ IICIC0
		73	INTUC3T	UARTC3 连续发送使能	UARTC3	0510H	00000510H	下一 PC 值	UC3TIC
		74	INTUC4R	UARTC4 接收完成/ UARTC4 接收错误	UARTC4	0520H	00000520H	下一 PC 值	UC4RIC
		75	INTUC4T	UARTC4 连续发送使能	UARTC4	0530H	00000530H	下一 PC 值	UC4TIC
		76	INTAD	A/D 转换完毕	A/D	0540H	00000540H	下一 PC 值	ADIC
		77	INTDMA0	DMA0 发送完毕	DMA	0550H	00000550H	下一 PC 值	DMAIC0
		78	INTDMA1	DMA1 发送完毕	DMA	0560H	00000560H	下一 PC 值	DMAIC1
		79	INTDMA2	DMA2 发送完毕	DMA	0570H	00000570H	下一 PC 值	DMAIC2
		80	INTDMA3	DMA3 发送完毕	DMA	0580H	00000580H	下一 PC 值	DMAIC3
		81	INTKR	按键返回中断	KR	0590H	00000590H	下一 PC 值	KRIC
		82	INTRTC0	RTC 持续周期信号	RTC	05A0H	000005A0H	下一 PC 值	RTC0IC
		83	INTRTC1	RTC 告警匹配	RTC	05B0H	000005B0H	下一 PC 值	RTC1IC
		84	INTRTC2	RTC 间隔信号	RTC	05C0H	000005C0H	下一 PC 值	RTC2IC
		85	INTC0ERR [※]	CAN0 错误	CAN0	05D0H	000005D0H	下一 PC 值	ERRIC0
		86	INTC0WUP1 [※]	CAN0 唤醒	CAN0	05E0H	000005E0H	下一 PC 值	WUPIC0
		87	INTC0REC [※]	CAN0 异常	CAN0	05F0H	000005F0H	下一 PC 值	RECIC0
		88	INTC0TRX [※]	CAN0 发送	CAN0	0600H	00000600H	下一 PC 值	TRXIC0
		92	INTUSBF0	USBF 中断	USBF	0640H	00000640H	下一 PC 值	UFIC0
		93	INTUSBF1	USBF 恢复中断	USBF	0650H	00000650H	下一 PC 值	UFIC1

注 仅限于 μ PD70F3770

- 备注

1. 默认优先级:

当两个或两个以上可屏蔽中断请求同时发生时, 优先级的次序。 最高优先级是 0。
不可屏蔽中断的优先级次序是 INTWDT2 > NMI。
- 恢复 PC:

当中断服务开始时, 程序计数器 (PC) 的值被存入 EIPC、FEPC 或 DBPC。 然而, 需要注意的是: 在以下指令之一执行时, 程序响应一个不可屏蔽中断或可屏蔽中断, 则恢复 PC 不是下一 PC 值 (nextPC)。(如果在中断执行期间响应一个中断, 执行停止, 而之后在中断服务完成后恢复执行。)

• Load 指令 (SLD.B, SLD.BU, SLD.H, SLD.HU, SLD.W)。

• 除法指令 (DIV, DIVH, DIVU, DIVHU)

• PREPARE, DISPOSE 指令 (仅当在堆栈指针更新前产生中断)
- 下一 PC 值:

中断/异常处理后随之由此进行处理的 PC 值。
2. 当发生非法操作码异常时, 非法指令的执行地址计算为: (PC 恢复值-4)。

表 23-3. V850ES/JH3-H 中断源 (1/4)

类型	类别	默认优先级	名称	触发	产生单元	异常码	处理程序地址	恢复 PC	中断控制寄存器
复位	中断	—	RESET	RESET 引脚输入 由内部源复位	RESET	0000H	00000000H	未定义	—
不可屏蔽	中断	—	NMI	NMI 引脚有效沿输入	引脚	0010H	00000010H	下一 PC 值	—
		—	INTWDT2	WDT2 溢出	WDT2	0020H	00000020H	注 1	—
软件异常	异常	—	TRAP0n ^{Note 2}	TRAP 指令	—	004nH ^{Note 2}	00000040H	下一 PC 值	—
		—	TRAP1n ^{Note 2}	TRAP 指令	—	005nH ^{Note 2}	00000050H	下一 PC 值	—
异常陷阱	异常	—	ILGOP/DBG0	非法操作码/ DBTRAP 指令	—	0060H	00000060H	下一 PC 值	—
可屏蔽	中断	0	INTLVI	低电压检测	POCLVI	0080H	00000080H	下一 PC 值	LVIIIC
		1	INTP00	外部中断引脚输入边沿检测 (INTP00)	引脚	0090H	00000090H	下一 PC 值	PIC00
		2	INTP01	外部中断引脚输入边沿检测 (INTP01)	引脚	00A0H	000000A0H	下一 PC 值	PIC01
		3	INTP02	外部中断引脚输入边沿检测 (INTP02)	引脚	00B0H	000000B0H	下一 PC 值	PIC02
		4	INTP03	外部中断引脚输入边沿检测 (INTP03)	引脚	00C0H	000000C0H	下一 PC 值	PIC03
		5	INTP04	外部中断引脚输入边沿检测 (INTP04)	引脚	00D0H	000000D0H	下一 PC 值	PIC04
		6	INTP05	外部中断引脚输入边沿检测 (INTP05)	引脚	00E0H	000000E0H	下一 PC 值	PIC05
		7	INTP06	外部中断引脚输入边沿检测 (INTP06)	引脚	00F0H	000000F0H	下一 PC 值	PIC06
		8	INTP07	外部中断引脚输入边沿检测 (INTP07)	引脚	0100H	00000100H	下一 PC 值	PIC07
		9	INTP08	外部中断引脚输入边沿检测 (INTP08)	引脚	0110H	00000110H	下一 PC 值	PIC08
		10	INTP09	外部中断引脚输入边沿检测 (INTP09)	引脚	0120H	00000120H	下一 PC 值	PIC09
		11	INTP10	外部中断引脚输入边沿检测 (INTP10)	引脚	0130H	00000130H	下一 PC 值	PIC10
		12	INTP11	外部中断引脚输入边沿检测 (INTP11)	引脚	0140H	00000140H	下一 PC 值	PIC11
		13	INTP12	外部中断引脚输入边沿检测 (INTP12)	引脚	0150H	00000150H	下一 PC 值	PIC12
		14	INTP13	外部中断引脚输入边沿检测 (INTP16)	引脚	0160H	00000160H	下一 PC 值	PIC13
		15	INTP14	外部中断引脚输入边沿检测 (INTP14)	引脚	0170H	00000170H	下一 PC 值	PIC14

注 1. 关于 INTWDT2 中断恢复的情况, 请参见 23.2.2 (2) INTWDT2 的信号。

2. n = 0 至 FH

表 23-3. V850ES/JH3-H 中断源 (2/4)

类型	类别	默认优先级	名称	触发	产生单元	异常码	处理程序地址	恢复 PC	中断控制寄存器
可屏蔽	中断	16	INTP15	外部中断引脚输入边沿检测 (INTP15)	引脚	0180H	00000180H	下一 PC 值	PIC15
		17	INTP16	外部中断引脚输入边沿检测 (INTP16)	引脚	0190H	00000190H	下一 PC 值	PIC16
		18	INTP17	外部中断引脚输入边沿检测 (INTP17)	引脚	01A0H	000001A0H	下一 PC 值	PIC17
		19	INTP18	外部中断引脚输入边沿检测 (INTP18)	引脚	01B0H	000001B0H	下一 PC 值	PIC18
		20	INTTAB0OV	TAB0 溢出	TAB0	01C0H	000001C0H	下一 PC 值	TAB0OVIC
		21	INTTAB0CC0	TAB0 捕获 0/ 比较 0 匹配	TAB0	01D0H	000001D0H	下一 PC 值	TAB0CCIC0
		22	INTTAB0CC1	TAB0 捕获 1/ 比较 1 匹配	TAB0	01E0H	000001E0H	下一 PC 值	TAB0CCIC1
		23	INTTAB0CC2	TAB0 捕获 2/ 比较 2 匹配	TAB0	01F0H	000001F0H	下一 PC 值	TAB0CCIC2
		24	INTTAB0CC3	TAB0 捕获 3/ 比较 3 匹配	TAB0	0200H	00000200H	下一 PC 值	TAB0CCIC3
		25	INTTAB1OV ^{#1}	TAB1 溢出	TAB1	0210H	00000210H	下一 PC 值	TAB1OVIC
		26	INTTAB1CC0 ^{#2}	TAB1 捕获 0/ 比较 0 匹配	TAB1	0220H	00000220H	下一 PC 值	TAB1CCIC0
		27	INTTAB1CC1	TAB1 捕获 1/ 比较 1 匹配	TAB1	0230H	00000230H	下一 PC 值	TAB1CCIC1
		28	INTTAB1CC2	TAB1 捕获 2/ 比较 2 匹配	TAB1	0240H	00000240H	下一 PC 值	TAB1CCIC2
		29	INTTAB1CC3	TAB1 捕获 3/ 比较 3 匹配	TAB1	0250H	00000250H	下一 PC 值	TAB1CCIC3
		30	INTTT0OV	TMT0 溢出	TMT0	0260H	00000260H	下一 PC 值	TT0OVIC
		31	INTTT0CC0	TMT0 捕获 0/ 比较 0 匹配	TMT0	0270H	00000270H	下一 PC 值	TT0CCIC0
		32	INTTT0CC1	TMT0 捕获 1/ 比较 1 匹配	TMT0	0280H	00000280H	下一 PC 值	TT0CCIC1
		33	INTTT0EC	TMT0 编码器输入	TMT0	0290H	00000290H	下一 PC 值	TT0ECIC
		34	INTTAA0OV	TAA0 溢出	TAA0	02A0H	000002A0H	下一 PC 值	TAA0OVIC
		35	INTTAA0CC0	TAA0 捕获 0/ 比较 0 匹配	TAA0	02B0H	000002B0H	下一 PC 值	TAA0CCIC0
		36	INTTAA0CC1	TAA0 捕获 1/ 比较 1 匹配	TAA0	02C0H	000002C0H	下一 PC 值	TAA0CCIC1
		37	INTTAA1OV	TAA1 溢出	TAA1	02D0H	000002D0H	下一 PC 值	TAA1OVIC

- 注
1. 在 6-段 PWM 输出模式中使用 TAB1 时, 将其用作来自 TMQOP 的 0 匹配中断 (TAB1TOD) 请求。
 2. 在 6-段 PWM 输出模式中使用 TAB1 时, 将其用作来自 TMQOP 的比较匹配中断 (TAB1TICD0) 请求。

表 23-3. V850ES/JH3-H 中断源 (3/4)

类型	类别	默认优先级	名称	触发	产生单元	异常码	处理程序地址	恢复 PC	中断控制寄存器
可屏蔽	中断	38	INTTAA1CC0	TAA1 捕获 0/ 比较 0 匹配	TAA1	02E0H	000002E0H	下一 PC 值	TAA1CCIC 0
		39	INTTAA1CC1	TAA1 捕获 1/ 比较 1 匹配	TAA1	02F0H	000002F0H	下一 PC 值	TAA1CCIC 1
		40	INTTAA2OV	TAA2 溢出	TAA2	0300H	00000300H	下一 PC 值	TAA2OVIC
		41	INTTAA2CC0	TAA2 捕获 0/ 比较 0 匹配	TAA2	0310H	00000310H	下一 PC 值	TAA2CCIC 0
		42	INTTAA2CC1	TAA2 捕获 1/ 比较 1 匹配	TAA2	0320H	00000320H	下一 PC 值	TAA2CCIC 1
		43	INTTAA3OV	TAA3 溢出	TAA3	0330H	00000330H	下一 PC 值	TAA3OVIC
		44	INTTAA3CC0	TAA3 捕获 0/ 比较 0 匹配	TAA3	0340H	00000340H	下一 PC 值	TAA3CCIC 0
		45	INTTAA3CC1	TAA3 捕获 1/ 比较 1 匹配	TAA3	0350H	00000350H	下一 PC 值	TAA3CCIC 1
		46	INTTAA4OV	TAA4 溢出	TAA4	0360H	00000360H	下一 PC 值	TAA4OVIC
		47	INTTAA4CC0	TAA4 比较 0 匹配	TAA4	0370H	00000370H	下一 PC 值	TAA4CCIC 0
		48	INTTAA4CC1	TAA4 比较 1 匹配	TAA4	0380H	00000380H	下一 PC 值	TAA4CCIC 1
		49	INTTAA5OV	TAA5 溢出	TAA5	0390H	00000390H	下一 PC 值	TAA5OVIC
		50	INTTAA5CC0	TAA5 捕获 0/ 比较 0 匹配	TAA5	03A0H	000003A0H	下一 PC 值	TAA5CCIC 0
		51	INTTAA5CC1	TAA5 捕获 1/ 比较 1 匹配	TAA5	03B0H	000003B0H	下一 PC 值	TAA5CCIC 1
		52	INTTM0EQ0	TMM0 比较匹配	TMM0	03C0H	000003C0H	下一 PC 值	TM0EQIC 0
		53	INTTM1EQ0	TMM1 比较匹配	TMM1	03D0H	000003D0H	下一 PC 值	TM1EQIC 0
		54	INTTM2EQ0	TMM2 比较匹配	TMM2	03E0H	000003E0H	下一 PC 值	TM2EQIC 0
		55	INTTM3EQ0	TMM3 比较匹配	TMM3	03F0H	000003F0H	下一 PC 值	TM3EQIC 0
		56	INTCF0R/ INTIIC1	CSIF0 接收完毕/ CSIF0 接收错误/ IIC1 发送完毕	CSIF0/ IIC1	0400H	00000400H	下一 PC 值	CF0RIC/ IIC1C1
		57	INTCF0T	CSIF0 连续发送写使能	CSIF0	0410H	00000410H	下一 PC 值	CF0TIC
		58	INTCF1R	CSIF1 接收完毕/ CSIF1 接收错误	CSIF1	0420H	00000420H	下一 PC 值	CF1RIC
		59	INTCF1T	CSIF1 连续发送写使能	CSIF1	0430H	00000430H	下一 PC 值	CF1TIC
		60	INTCF2R	CSIF2 接收完毕/ CSIF2 接收错误	CSIF2	0440H	00000440H	下一 PC 值	CF2RIC
		61	INTCF2T	CSIF2 连续发送写使能	CSIF2	0450H	00000450H	下一 PC 值	CF2TIC
		62	INTCF3R	CSIF3 接收完毕/ CSIF3 接收错误	CSIF3	0460H	00000460H	下一 PC 值	CF3RIC

表 23-3. V850ES/JH3-H 中断源 (4/4)

类型	类别	默认优先级	名称	触发	产生单元	异常码	处理程序地址	恢复 PC	中断控制寄存器
可屏蔽	中断	63	INTCF3T	CSIF3 连续发送写使能	CSIF3	0470H	00000470H	下一 PC 值	CF3TIC
		64	INTCF4R	CSIF4 接收完毕/ CSIF4 接收错误	CSIF4	0480H	00000480H	下一 PC 值	CF4RIC
		65	INTCF4T	CSIF4 连续发送写使能	CSIF4	0490H	00000490H	下一 PC 值	CF4TIC
		66	INTUC0R	UARTC0 接收完成/UARTC0 接收错误	UARTC0	04A0H	000004A0H	下一 PC 值	UC0RIC
		67	INTUC0T	UARTC0 连续发送使能	UARTC0	04B0H	000004B0H	下一 PC 值	UC0TIC
		68	INTUC1R /INTIIC2	UARTC1 接收完成/ UARTC1 接收错误/IIC2 传送完成	UARTC1/ IIC2	04C0H	000004C0H	下一 PC 值	UC1RIC/ IICIC2
		69	INTUC1T	UARTC1 连续发送使能	UARTC1	04D0H	000004D0H	下一 PC 值	UC1TIC
		70	INTUC2R	UARTC2 接收完成/UARTC2 接收错误	UARTC2	04E0H	000004E0H	下一 PC 值	UC2RIC
		71	INTUC2T	UARTC2 连续发送使能	UARTC2	04F0H	000004F0H	下一 PC 值	UC2TIC
		72	INTUC3R/ INTIIC0	UARTC3 接收完成/UARTC0 接收错误/IIC0 传送完成	UARTC3/ IIC0	0500H	00000500H	下一 PC 值	UC3RIC/ IICIC0
		73	INTUC3T	UARTC3 连续发送使能	UARTC3	0510H	00000510H	下一 PC 值	UC3TIC
		74	INTUC4R	UARTC4 接收完成/UARTC4 接收错误	UARTC4	0520H	00000520H	下一 PC 值	UC4RIC
		75	INTUC4T	UARTC4 连续发送使能	UARTC4	0530H	00000530H	下一 PC 值	UC4TIC
		76	INTAD	A/D 转换完毕	A/D	0540H	00000540H	下一 PC 值	ADIC
		77	INTDMA0	DMA0 发送完毕	DMA	0550H	00000550H	下一 PC 值	DMAIC0
		78	INTDMA1	DMA1 发送完毕	DMA	0560H	00000560H	下一 PC 值	DMAIC1
		79	INTDMA2	DMA2 发送完毕	DMA	0570H	00000570H	下一 PC 值	DMAIC2
		80	INTDMA3	DMA3 发送完毕	DMA	0580H	00000580H	下一 PC 值	DMAIC3
		81	INTKR	按键返回中断	KR	0590H	00000590H	下一 PC 值	KRIC
		82	INTRTC0	RTC 持续周期信号	RTC	05A0H	000005A0H	下一 PC 值	RTC0IC
		83	INTRTC1	RTC 告警匹配	RTC	05B0H	000005B0H	下一 PC 值	RTC1IC
		84	INTRTC2	RTC 间隔信号	RTC	05C0H	000005C0H	下一 PC 值	RTC2IC
		85	INTC0ERR [※]	CAN0 错误	CAN0	05D0H	000005D0H	下一 PC 值	ERRIC0
		86	INTC0WUP1 [※]	CAN0 唤醒	CAN0	05E0H	000005E0H	下一 PC 值	WUPIC0
		87	INTC0REC [※]	CAN0 接收	CAN0	05F0H	000005F0H	下一 PC 值	RECIC0
		88	INTC0TRX [※]	CAN0 发送	CAN0	0600H	00000600H	下一 PC 值	TRXIC0
		92	INTUSBF0	USB 中断	USBF	0640H	00000640H	下一 PC 值	UFIC0
		93	INTUSBF1	USB 恢复中断	USBF	0650H	00000650H	下一 PC 值	UFIC1

注 仅限于 μ PD70F3771

- 备注**
1. 默认优先级: 当两个或两个以上可屏蔽中断请求同时发生时, 优先级的次序。 最高优先级是 0。
不可屏蔽中断的优先级次序是 INTWDT2 > NMI。
- 恢复 PC: 当中断服务开始时, 程序计数器 (PC) 的值被存入 EIPC、FEPC 或 DBPC。 然而, 需要注意的是: 在以下指令之一执行时, 程序响应一个不可屏蔽中断或可屏蔽中断, 则恢复 PC 不是下一 PC 值 (nextPC)。(如果在中断执行期间响应一个中断, 执行停止, 而之后在中断服务完成后恢复执行。)
- Load 指令 (SLD.B, SLD.BU, SLD.H, SLD.HU, SLD.W)。
 - 除法指令 (DIV, DIVH, DIVU, DIVHU)
 - PREPARE, DISPOSE 指令 (仅当在堆栈指针更新前产生中断)
- 下一 PC 值: 中断/异常处理后随之由此进行处理的 PC 值。
2. 当发生非法操作码异常时, 非法指令的执行地址计算为: (PC 恢复值-4)。

23.2 不可屏蔽中断

即便处于中断禁止（DI）状态，不可屏蔽中断请求信号也无条件地得到相应。不可屏蔽中断并不遵从优先级控制，而且，比所有其他中断请求信号更为优先。

此产品有以下两种不可屏蔽中断请求信号。

- NMI 引脚输入（NMI）
- 因看门狗定时器溢出产生的不可屏蔽中断请求信号（INTWDT2）。

NMI 引脚的有效沿可以选择为以下 4 种类型：“上升沿”、“下降沿”、“双边沿”和“无边沿检测”。

当 WDTM2.WDM21 和 WDTM2.WDM20 位被设置为“01”时，因看门狗定时器 2（INTWDT2）功能溢出产生不可屏蔽中断请求信号。

当两个或两个以上不可屏蔽中断请求信号同时发生时，较高优先级的中断被服务，如下所示（低优先级的中断请求信号被忽略）：

INTWDT2 > NMI

在 NMI 处理时，新的 NMI 或 INTWDT2 请求信号发生，则按以下处理。

(1) 如果 NMI 服务期间产生新的 NMI 请求信号

不管 PSW.NP 位的值如何，新的 NMI 请求信号都会被挂起。待决的 NMI 请求信号在当前正在处理的 NMI 请求完毕后得到响应（在执行完 RETI 指令之后）。

(2) 如果 NMI 服务期间产生 INTWDT2 请求信号

在 NMI 处理期间，如果 NP 位保持置位状态（1），INTWDT2 请求信号被挂起。待决的 NMI 请求信号在当前正在处理的 INTWDT2 请求完毕后得到响应（在执行完 RETI 指令之后）。

在 NMI 请求处理期间，如果 NP 位被清除为 0，执行新产生的 INTWDT2 请求信号（NMI 处理服务停止）。

注意事项 关于不可屏蔽中断请求信号（INTWDT2）的不可屏蔽中断服务的执行，请参见 23.2.2（2）INTWDT2 信号。

图 23-1. 不可屏蔽中断请求信号的响应操作（1/2）

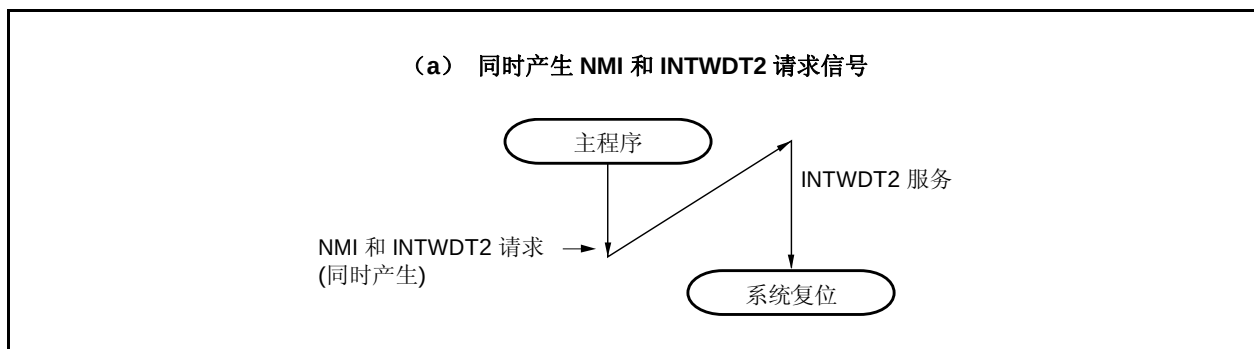
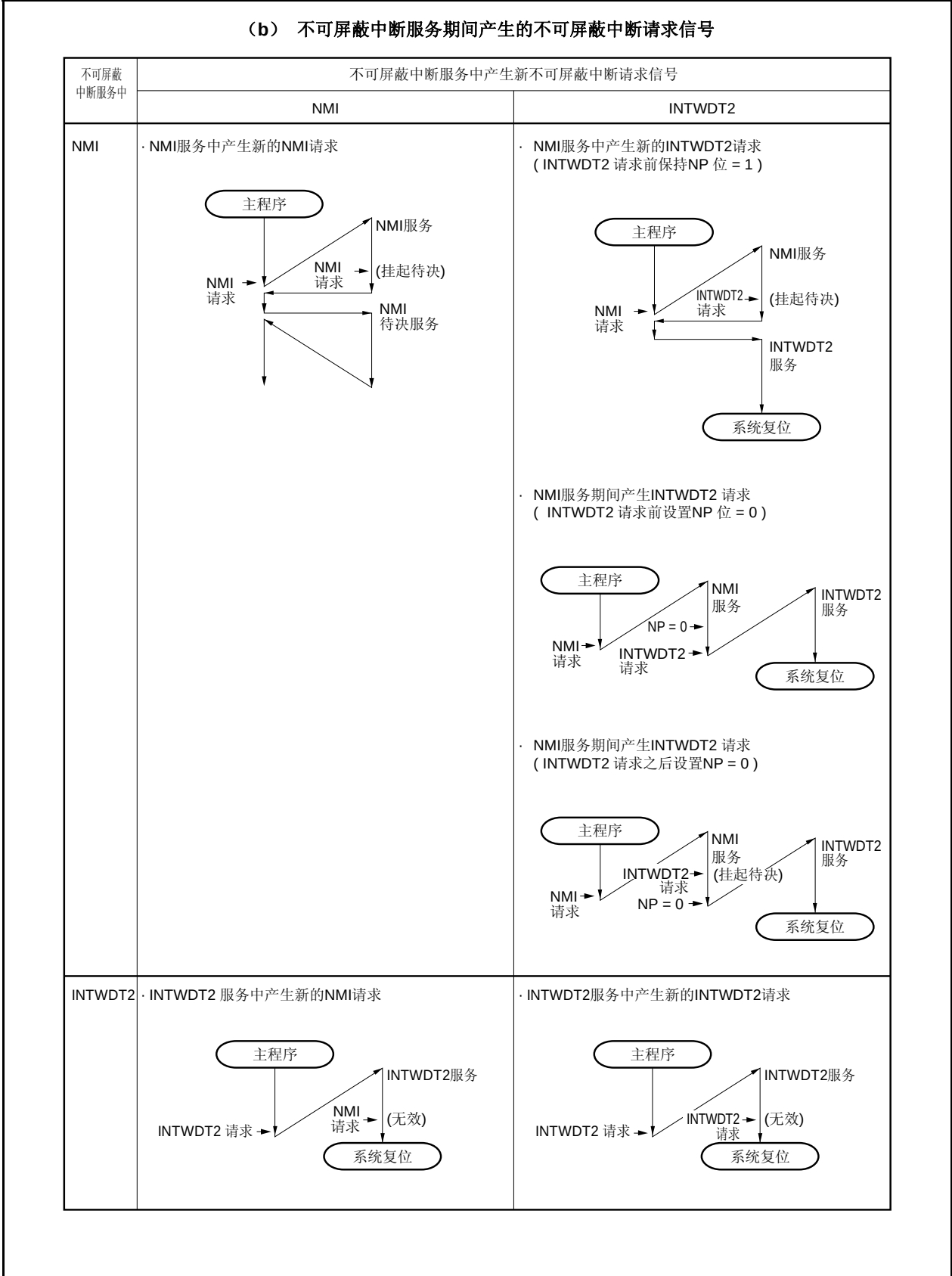


图 23-1. 不可屏蔽中断请求信号的响应操作 (2/2)



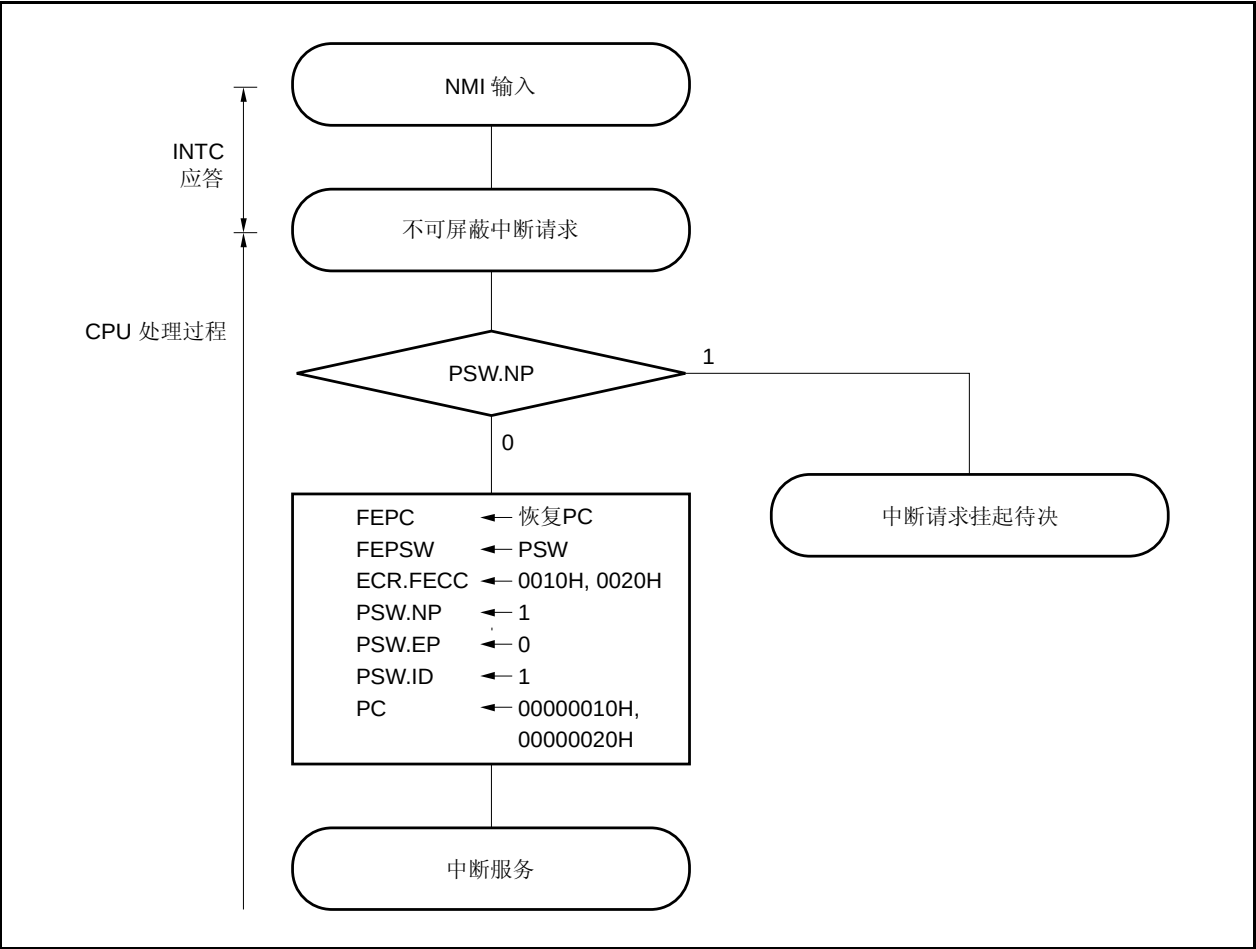
23.2.1 操作

如果一个不可屏蔽中断请求信号产生，则 CPU 执行以下处理过程，并控制转移至一个处理程序。

- <1> 将恢复 PC 保存至 FEPC。
- <2> 将当前 PSW 保存至 FEPSW。
- <3> 将异常码（0010H，0020H）写入 ECR 的高半字（FECC）。
- <4> 将 PSW.NP 和 PSW.ID 位设置为 1 并将 PSW.EP 位清除为 0。
- <5> 设置对应不可屏蔽中断的处理程序地址（00000010H，00000020H）至 PC，并且，控制转移。

不可屏蔽中断的服务流程如下所示：

图 23-2. 不可屏蔽中断服务流程



23.2.2 恢复

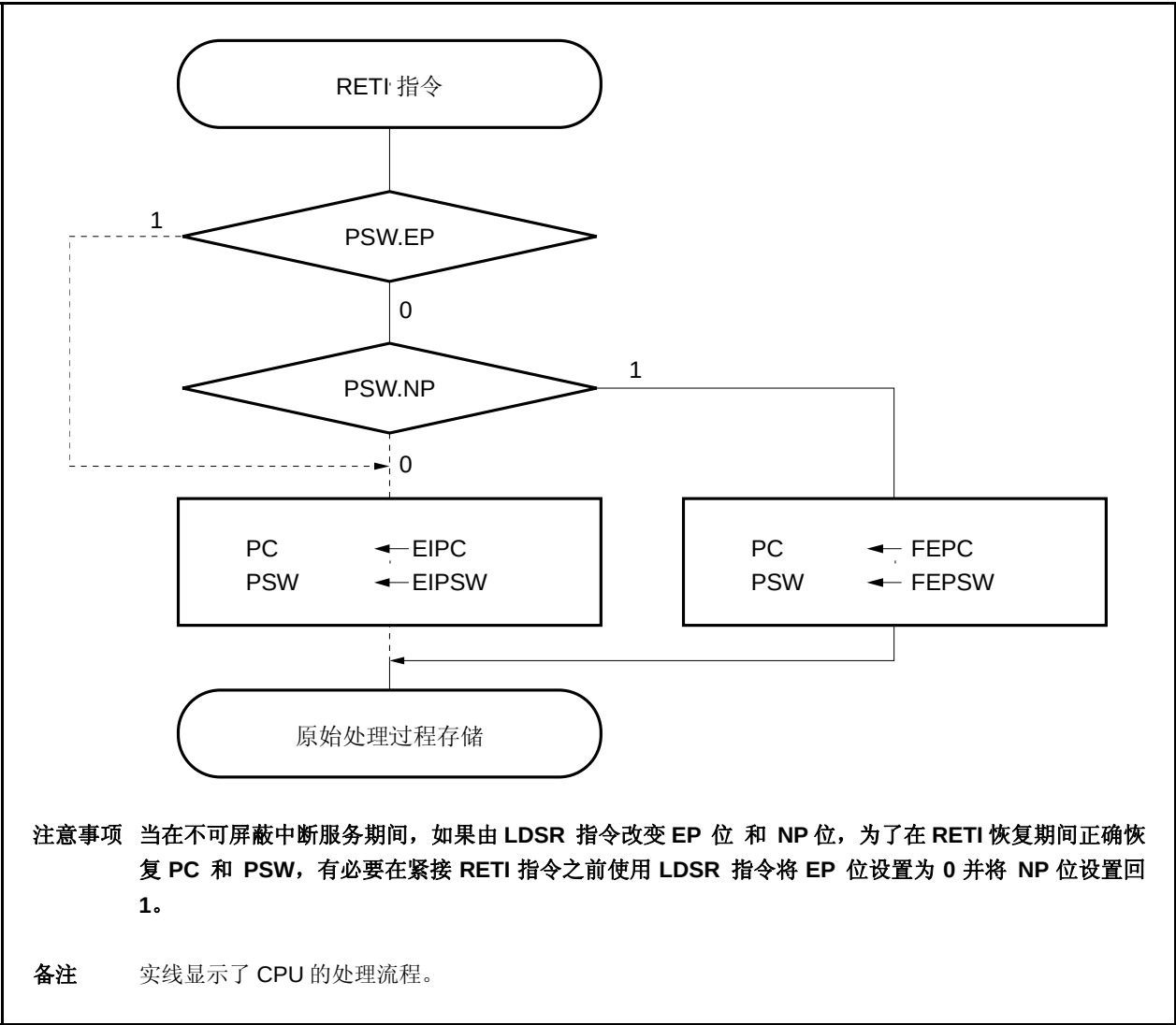
(1) 由 NMI 引脚输入

通过 RETI 指令执行从 NMI 服务的恢复处理。
当执行 RETI 指令时，CPU 执行以下处理过程，并且，控制转移到恢复 PC 的地址。

- <1> 因为 PSW.EP 位是 0，而且 PSW.NP 位是 1，所以，由 FEPC 和 FEPSW 恢复相应的 PC 和 PSW 的恢复值。
- <2> 控制转移返回到恢复 PC 和 PSW 的地址。

RETI 指令的处理过程如下所示：

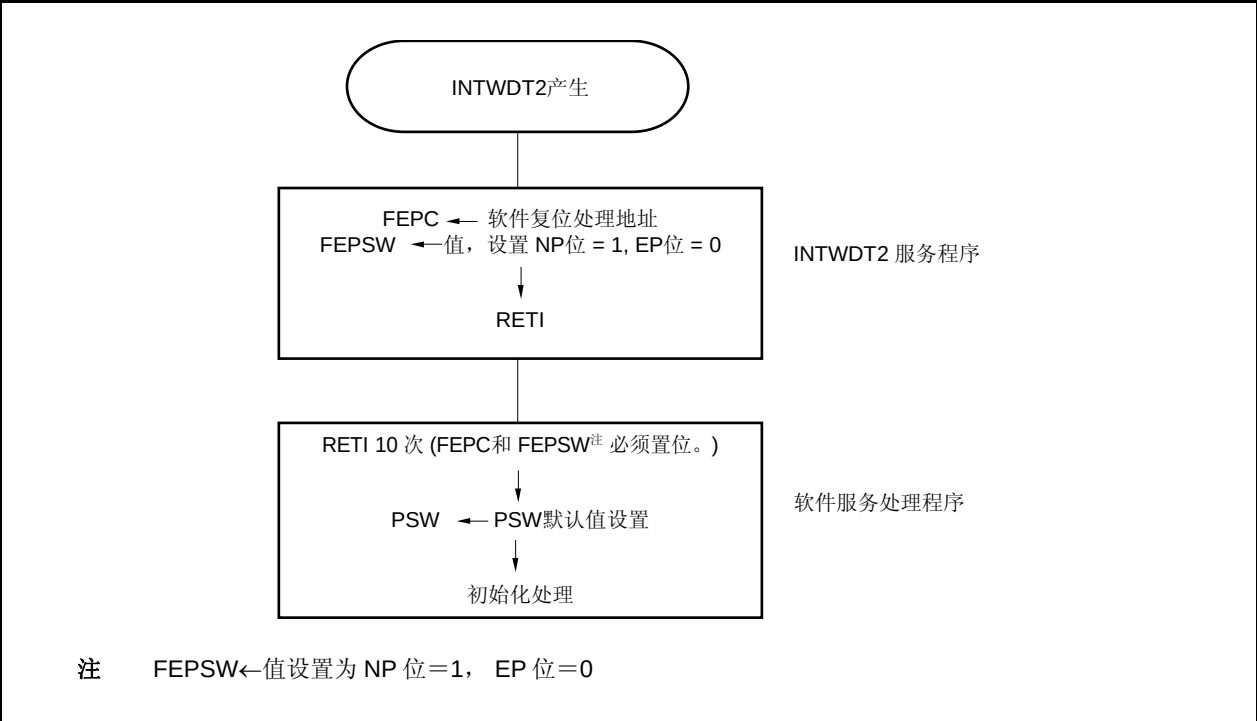
图 23-3. RETI 指令处理过程



(2) 从 INTWDT2 信号

禁止通过使用 RETI 指令执行不可屏蔽中断请求（INTWDT2）从不可屏蔽中断服务中恢复。 执行下面的软件复位处理过程。

图 23-4. 软件复位处理



23.2.3 NP 标志

NP 标志为一个状态标志，指示不可屏蔽中断服务正在进行。
当不可屏蔽中断请求信号被响应之后，本标志置位，并且，屏蔽不可屏蔽中断请求，以便禁止响应多重中断。

复位后：00000020H

	31																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																	

23.3 可屏蔽中断

可屏蔽中断请求信号可由中断控制寄存器进行屏蔽。 V850ES/JG3-H 和 V850ES/JH3-H 具有 84 至 91 个可屏蔽中断源。

如果同时发生两个或更多的可屏蔽中断请求信号，他们按照默认的优先级进行响应。除了默认优先级之外，使用中断控制寄存器（可编程优先控制）可以指定 8 个优先等级

当一个中断请求信号已经得到响应，则禁止响应其他可屏蔽中断请求信号并且中断禁止（DI）状态置位。

当在中断服务程序中执行 EI 指令，中断使能（EI）状态置位，它使得其它中断服务比正在处理的中断请求信号（由中断控制寄存器指定）具有更高的优先级。注意，只有具有较高优先级的中断才有这种能力；具有相同优先级等级的中断不能嵌套。

但是，为了允许多重中断（中断嵌套），在执行 EI 指令之前有必要将 EIPC 和 EIPSW 保存入存储器或通用寄存器中，并在使用 RETI 指令恢复 EIPC 和 EIPSW 原始值之前执行 DI 指令。

23.3.1 操作

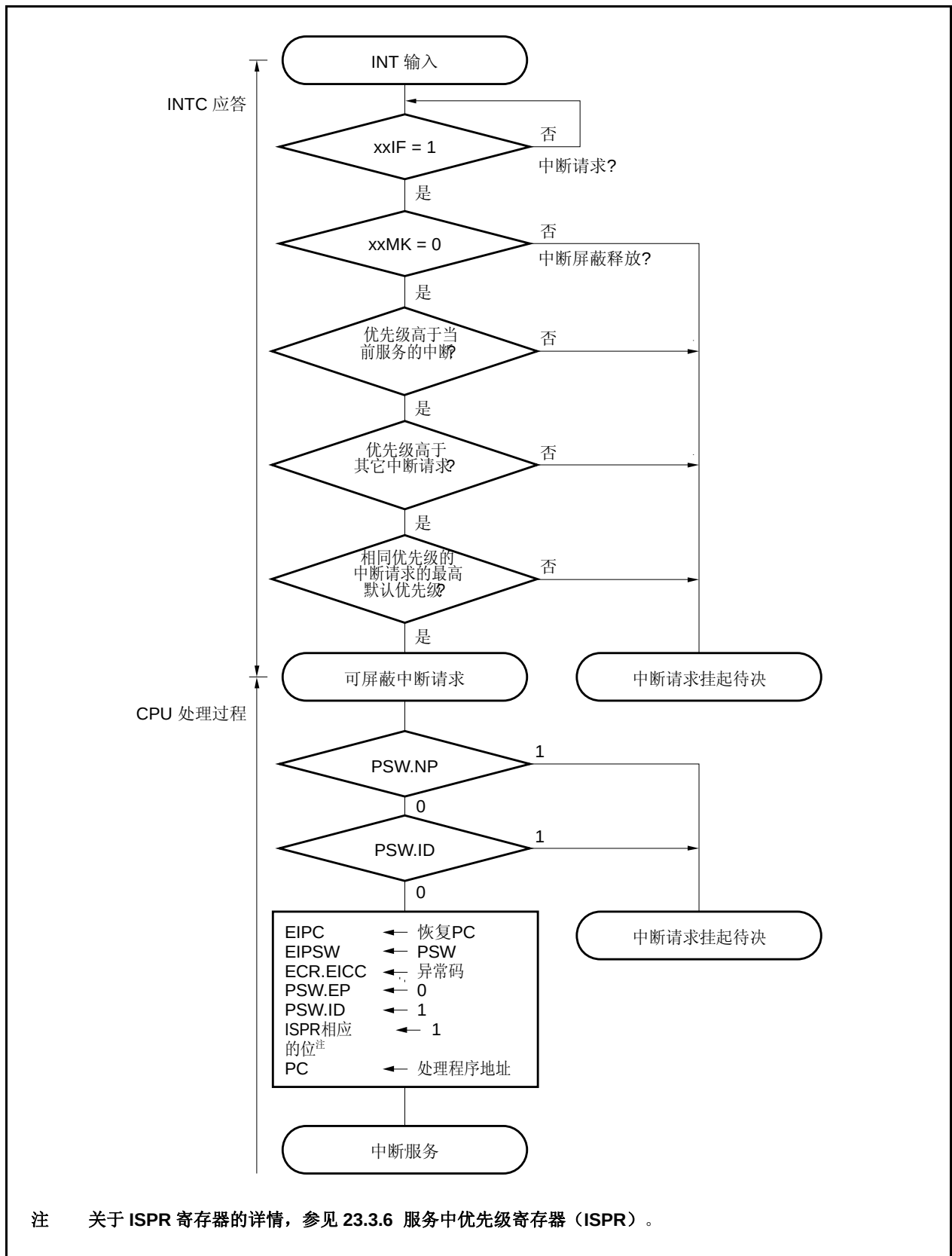
如果发生可屏蔽中断，CPU 执行以下处理过程，并控制转移至处理程序。

- <1> 将 PC 保存至 EIPC。
- <2> 将 PSW 保存至 EIPSW。
- <3> 将异常码写入 ECR（EICC）的低半字。
- <4> 设置 PSW。将 ID 位设为 1 并清除 PSW。EP 位为 0。
- <5> 设置对应于 PC 各中断的处理程序，并进行控制转移。

由中断控制寄存器（INTC）所屏蔽的可屏蔽中断请求信号，和于另外中断服务之中（当 PSW.NP 位=1 或 PSW.ID 位=1）时产生的可屏蔽中断请求信号，在 INTC 内保持挂起状态。这种情况下，如果可屏蔽中断取消屏蔽或使用 RETI 或 LDSR 指令将 NP 或 ID 设为 0，将根据挂起的可屏蔽中断请求之优先级来启动一个新的可屏蔽中断。

以下描述了可屏蔽中断的服务过程：

图 23-5. 可屏蔽中断服务过程



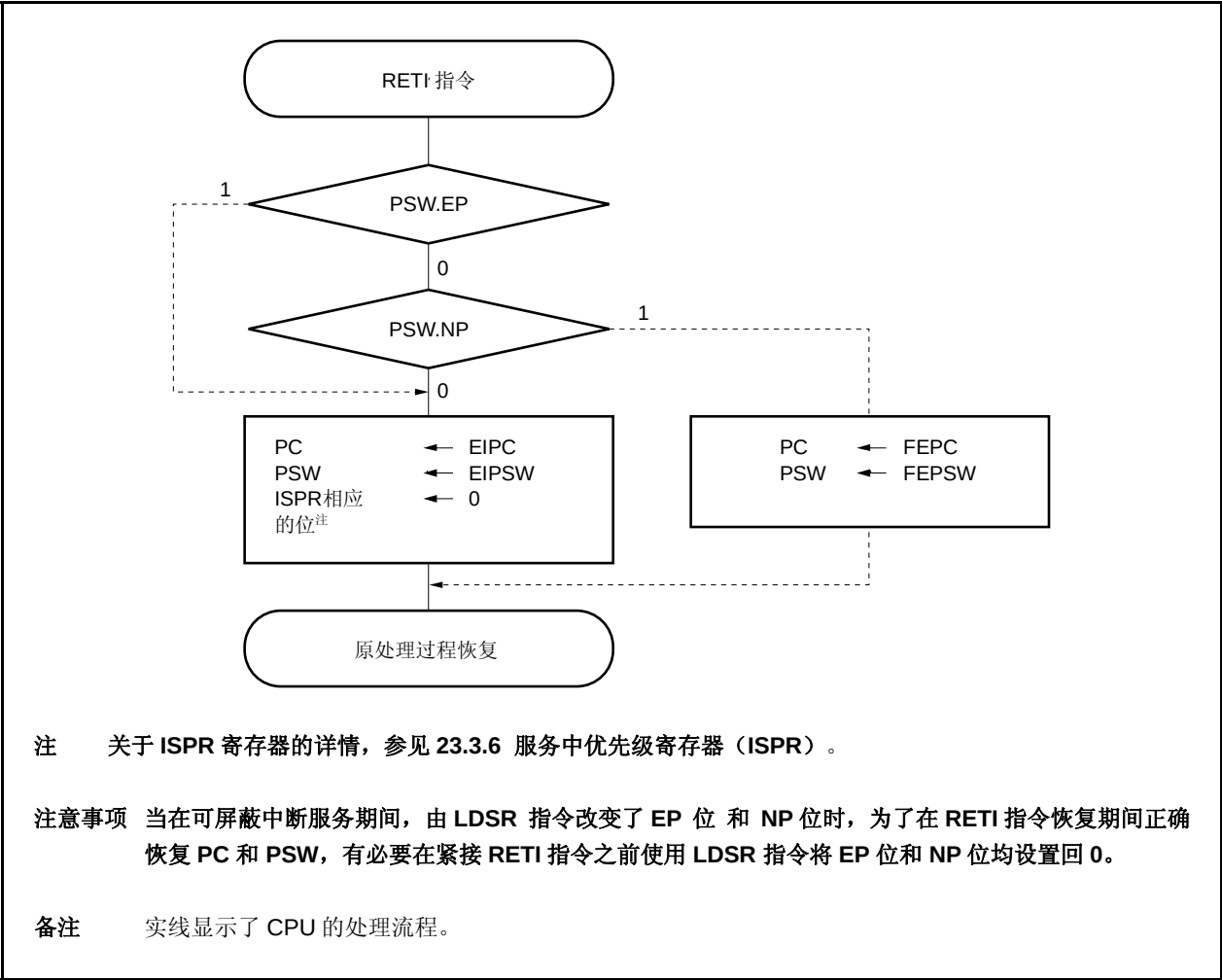
23.3.2 恢复

从可屏蔽中断服务中恢复，由 RETI 指令完成。
当执行 RETI 指令时，CPU 执行以下处理过程，并且，控制转移到恢复 PC 的地址。

- <1> 因为 PSW.EP 位是 0，而且 PSW.NP 位是 0，所以，由 EIPC 和 EIPSW 恢复 PC 和 PSW 的恢复值。
- <2> 控制转移返回到恢复 PC 和 PSW 的地址。

RETI 指令的处理过程如下所示：

图 23-6. RETI 指令处理过程



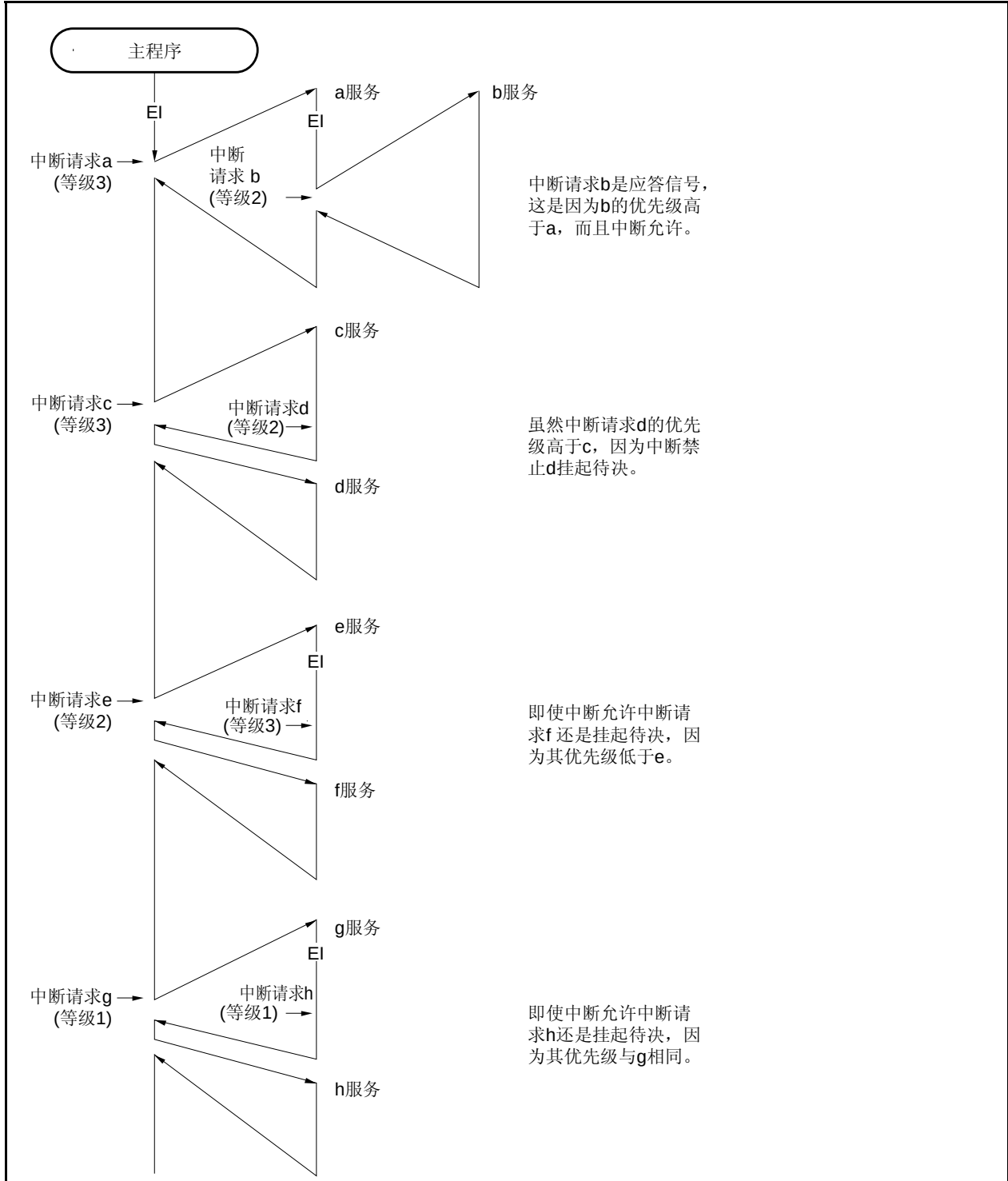
23.3.3 可屏蔽中断的优先级

INTC 执行多重中断服务，多重中断中在一个中断处于服务之际响应另一个中断。多重中断可由优先等级进行控制。

有两种类型的优先级控制方式：基于默认优先级控制，以及基于由中断控制寄存器（xxICn）的中断优先级指定位（xxPRn）确定的可编程优先级控制。当两个或两个以上由 xxPRn 位指定的具有相同优先级的中断同时发生，中断请求信号依照每个中断请求类型（默认优先级）预先设定的优先级次序进行处理。有关更多信息，请参见表 23-2 和表 23-3。通过设置优先级声明标志，可将可编程优先级控制中断请求信号定为 8 个级别。

注意，当中断请求信号被响应时，.ID 标志自动被置为 1。所以，当使用多重中断时，应该预先将 ID 标志清为 0（比如，通过在中断服务程序中安排一条 EI 指令）从而设置中断允许模式。

备注 xx: 每一个外设单元的标志名称（参见表 23-4 中断控制寄存器（xxICn））。
 n: 外设单元编号（参见表 23-4 中断控制寄存器（xxICn））。



注意事项 为了执行多重中断服务，EIPC 和 EIPSW 寄存器的值必须在执行 EI 指令前进行保存。当从多重中断服务中返回时，执行 DI 指令后恢复 EIPC 和 EIPSW 寄存器的值。

备注	1. 为了说明问题，图中 a 至 u 是中断请求信号的暂行名。 2. 图中的默认优先级表明两个中断请求信号之间的相对优先级。

图 23-7. 一个中断服务当中而另一个中断请求发生的处理举例 (2/2)

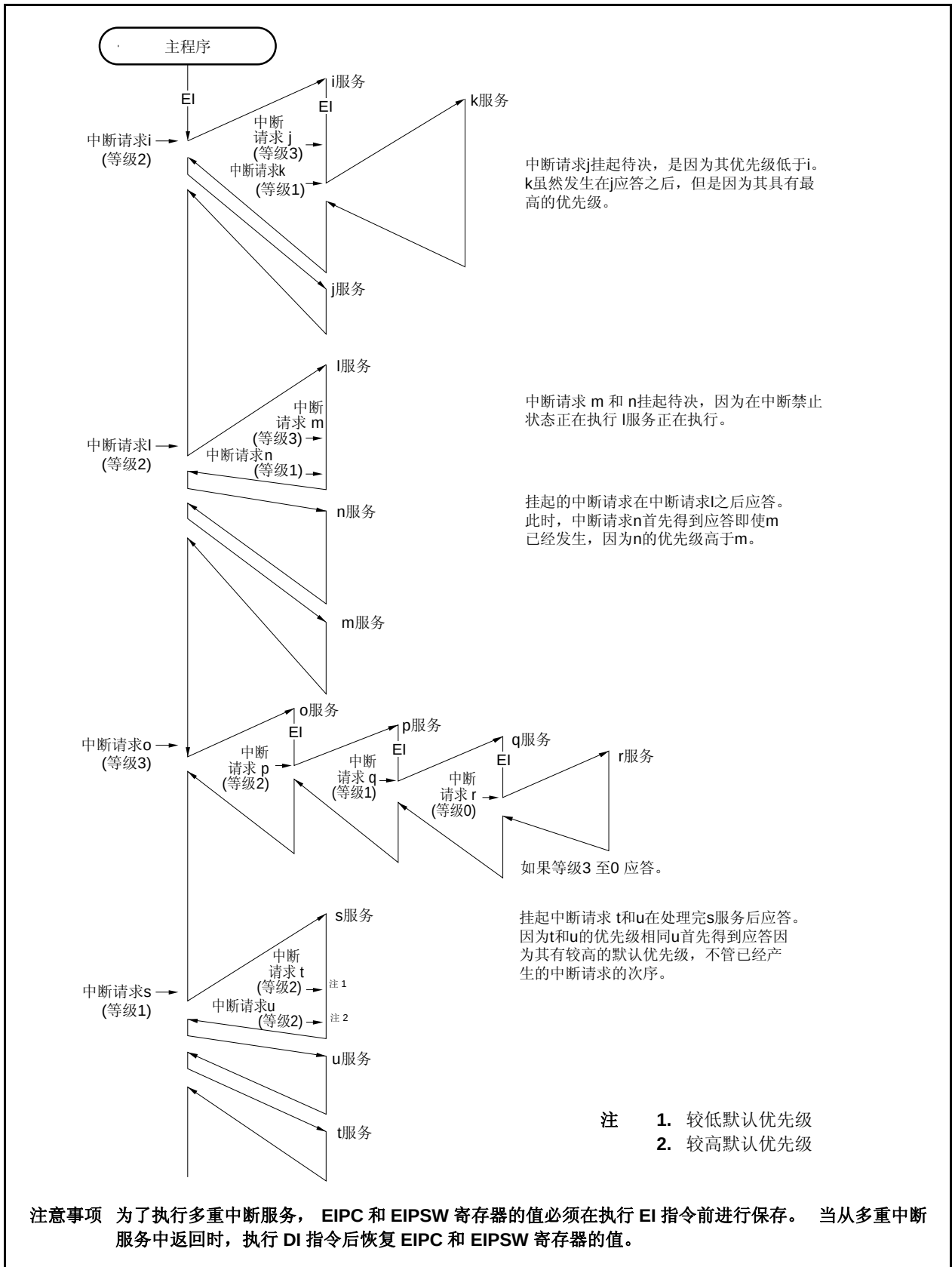
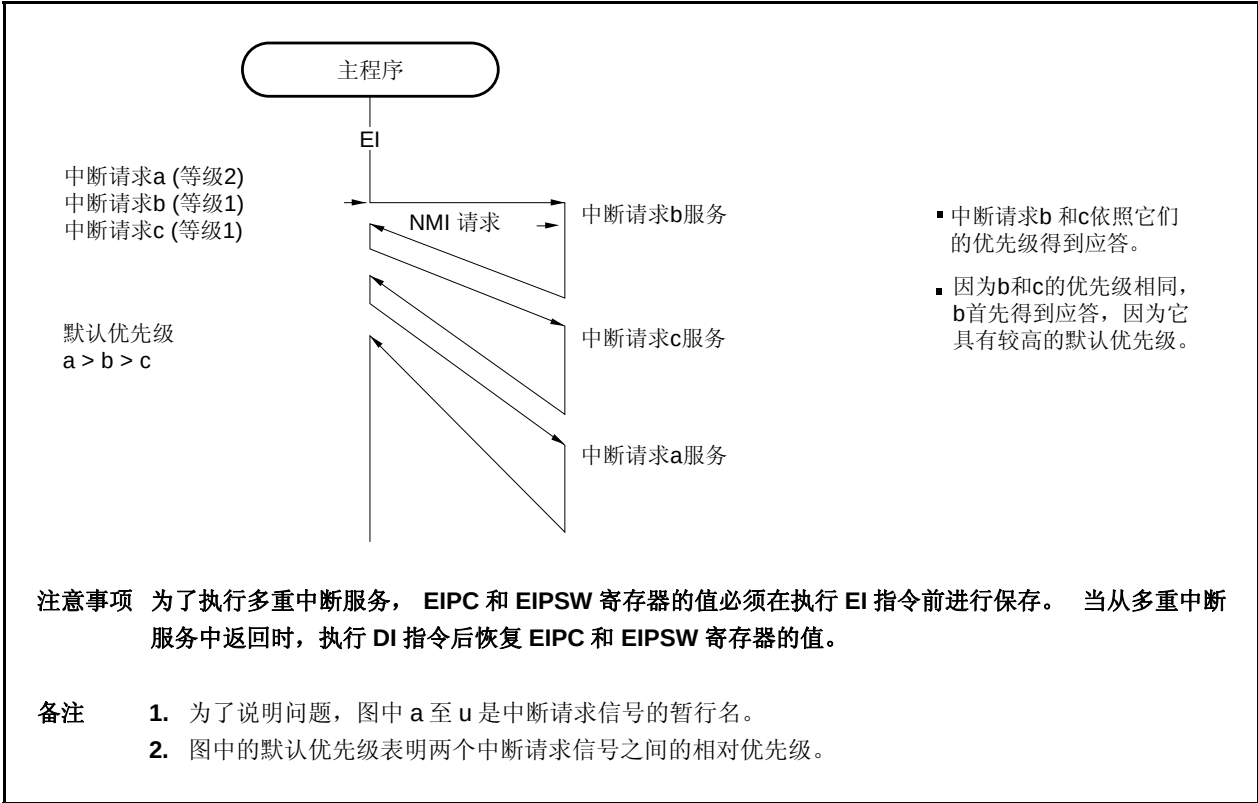


图 23-8. 同时产生中断请求信号服务的举例



23.3.4 中断控制寄存器（xxICn）

每一个中断请求信号（可屏蔽中断）都被赋予一个 xxICn 寄存器，用来设置每一个可屏蔽中断请求的控制条件。
本寄存器可按 8 位或 1 位读写。
系统复位后，寄存器被置为 47H。

注意事项 读取 xxICn.xxIFn 位时禁止中断（DI）或屏蔽中断。 如果在中断允许（EI）或未屏蔽中断的情况下读取 xxIFn 位，当响应一个中断与读取该位相冲突时，可能读不出正确值。

复位后：47H R/W 地址：FFFFF112H至FFFFF184H

	<7>	<6>	5	4	3	2	1	0
xxICn	xxIFn	xxMKn	0	0	0	xxPRn2	xxPRn1	xxPRn0

xxIFn	中断请求标志 ^注
0	不产生中断请求
1	产生中断请求

xxMKn	中断屏蔽标志
0	允许中断服务
1	禁止中断服务（挂起待决）

xxPRn2	xxPRn1	xxPRn0	中断优先级指定位
0	0	0	指定等级0（最高）
0	0	1	指定等级1.
0	1	0	指定等级2.
0	1	1	指定等级3.
1	0	0	指定等级4.
1	0	1	指定等级5.
1	1	0	指定等级6.
1	1	1	指定等级7（最低）

注 如果一个中断请求信号得到响应，xxIFn 标志由硬件自动复位。

备注 xx: 每一个外设单元的标志名称（参见 表 23-4 中断控制寄存器（xxICn））。
 n: 外设单元编号（参见 表 23-4 中断控制寄存器（xxICn））。

中断控制寄存器的地址及各位如下所示：

表 23-4. 中断控制寄存器 (xxlCn) (1/3)

地址	寄存器	位							
		<7>	<6>	5	4	3	2	1	0
FFFFF110H	LVIIC	LVIIIF	LVIMK	0	0	0	LVIPR2	LVIPR1	LVIPR0
FFFFF112H	PIC00 [※]	PIF00	PMK00	0	0	0	PPR002	PPR001	PPR000
FFFFF114H	PIC01 [※]	PIF01	PMK01	0	0	0	PPR012	PPR011	PPR010
FFFFF116H	PIC02	PIF02	PMK02	0	0	0	PPR022	PPR021	PPR020
FFFFF118H	PIC03	PIF03	PMK03	0	0	0	PPR032	PPR031	PPR030
FFFFF11AH	PIC04	PIF04	PMK04	0	0	0	PPR042	PPR041	PPR040
FFFFF11CH	PIC05	PIF05	PMK05	0	0	0	PPR052	PPR051	PPR050
FFFFF11EH	PIC06 [※]	PIF06	PMK06	0	0	0	PPR062	PPR061	PPR060
FFFFF120H	PIC07	PIF07	PMK07	0	0	0	PPR072	PPR071	PPR070
FFFFF122H	PIC08	PIF08	PMK08	0	0	0	PPR082	PPR081	PPR080
FFFFF124H	PIC09	PIF09	PMK09	0	0	0	PPR092	PPR091	PPR090
FFFFF126H	PIC10	PIF10	PMK10	0	0	0	PPR102	PPR101	PPR100
FFFFF128H	PIC11	PIF11	PMK11	0	0	0	PPR112	PPR111	PPR110
FFFFF12AH	PIC12	PIF12	PMK12	0	0	0	PPR122	PPR121	PPR120
FFFFF12CH	PIC13	PIF13	PMK13	0	0	0	PPR132	PPR131	PPR130
FFFFF12EH	PIC14	PIF14	PMK14	0	0	0	PPR142	PPR141	PPR140
FFFFF130H	PIC15	PIF15	PMK15	0	0	0	PPR152	PPR151	PPR150
FFFFF132H	PIC16	PIF16	PMK16	0	0	0	PPR162	PPR161	PPR160
FFFFF134H	PIC17	PIF17	PMK17	0	0	0	PPR172	PPR171	PPR170
FFFFF136H	PIC18	PIF18	PMK18	0	0	0	PPR182	PPR181	PPR180
FFFFF138H	TAB0OVIC	TAB0OVIF	TAB0OVMK	0	0	0	TAB0OVPPR2	TAB0OVPPR1	TAB0OVPPR0
FFFFF13AH	TAB0CCIC0	TAB0CCIF0	TAB0CCMK0	0	0	0	TAB0CCPPR02	TAB0CCPPR01	TAB0CCPPR00
FFFFF13CH	TAB0CCIC1	TAB0CCIF1	TAB0CCMK1	0	0	0	TAB0CCPPR12	TAB0CCPPR11	TAB0CCPPR10
FFFFF13EH	TAB0CCIC2	TAB0CCIF2	TAB0CCMK2	0	0	0	TAB0CCPPR22	TAB0CCPPR21	TAB0CCPPR20
FFFFF140H	TAB0CCIC3	TAB0CCIF3	TAB0CCMK3	0	0	0	TAB0CCPPR32	TAB0CCPPR31	TAB0CCPPR30
FFFFF142H	TAB1OVIC	TAB1OVIF	TAB1OVMK	0	0	0	TAB1OVPPR2	TAB1OVPPR1	TAB1OVPPR0
FFFFF144H	TAB1CCIC0	TAB1CCIF0	TAB1CCMK0	0	0	0	TAB1CCPPR02	TAB1CCPPR01	TAB1CCPPR00
FFFFF146H	TAB1CCIC1	TAB1CCIF1	TAB1CCMK1	0	0	0	TAB1CCPPR12	TAB1CCPPR11	TAB1CCPPR10
FFFFF148H	TAB1CCIC2	TAB1CCIF2	TAB1CCMK2	0	0	0	TAB1CCPPR22	TAB1CCPPR21	TAB1CCPPR20
FFFFF14AH	TAB1CCIC3	TAB1CCIF3	TAB1CCMK3	0	0	0	TAB1CCPPR32	TAB1CCPPR31	TAB1CCPPR30
FFFFF14CH	TT0OVIC	TT0OVIF	TT0OVMK	0	0	0	TT0OVPPR2	TT0OVPPR1	TT0OVPPR0
FFFFF14EH	TT0CCIC0	TT0CCIF0	TT0CCMK0	0	0	0	TT0CCPPR02	TT0CCPPR01	TT0CCPPR00
FFFFF150H	TT0CCIC1	TT0CCIF1	TT0CCMK1	0	0	0	TT0CCPPR12	TT0CCPPR11	TT0CCPPR10
FFFFF152H	TT0IECIC	TT0IECIF	TT0IECMK	0	0	0	TT0IECPPR2	TT0IECPPR1	TT0IECPPR0
FFFFF154H	TAA0OVIC	TAA0OVIF	TAA0OVMK	0	0	0	TAA0OVPPR2	TAA0OVPPR1	TAA0OVPPR0
FFFFF156H	TAA0CCIC0	TAA0CCIF0	TAA0CCMK0	0	0	0	TAA0CCPPR02	TAA0CCPPR01	TAA0CCPPR00
FFFFF158H	TAA0CCIC1	TAA0CCIF1	TAA0CCMK1	0	0	0	TAA0CCPPR12	TAA0CCPPR11	TAA0CCPPR10
FFFFF15AH	TAA1OVIC	TAA1OVIF	TAA1OVMK	0	0	0	TAA1OVPPR2	TAA1OVPPR1	TAA1OVPPR0
FFFFF15CH	TAA1CCIC0	TAA1CCIF0	TAA1CCMK0	0	0	0	TAA1CCPPR02	TAA1CCPPR01	TAA1CCPPR00
FFFFF15EH	TAA1CCIC1	TAA1CCIF1	TAA1CCMK1	0	0	0	TAA1CCPPR12	TAA1CCPPR11	TAA1CCPPR10

注 仅限于 V850ES/JH3-H

表 23-4. 中断控制寄存器 (xxlCn) (2/3)

地址	寄存器	位							
		<7>	<6>	5	4	3	2	1	0
FFFFF160H	TAA2OVIC	TAA2OVIF	TAA2OVMK	0	0	0	TAA2OVPPR2	TAA2OVPPR1	TAA2OVPPR0
FFFFF162H	TAA2CCIC0	TAA2CCIF0	TAA2CCMK0	0	0	0	TAA2CCPPR02	TAA2CCPPR01	TAA2CCPPR00
FFFFF164H	TAA2CCIC1	TAA2CCIF1	TAA2CCMK1	0	0	0	TAA2CCPPR12	TAA2CCPPR11	TAA2CCPPR10
FFFFF166H	TAA3OVIC	TAA3OVIF	TAA3OVMK	0	0	0	TAA3OVPPR2	TAA3OVPPR1	TAA3OVPPR0
FFFFF168H	TAA3CCIC0	TAA3CCIF0	TAA3CCMK0	0	0	0	TAA3CCPPR02	TAA3CCPPR01	TAA3CCPPR00
FFFFF16AH	TAA3CCIC1	TAA3CCIF1	TAA3CCMK1	0	0	0	TAA3CCPPR12	TAA3CCPPR11	TAA3CCPPR10
FFFFF16CH	TAA4OVIC	TAA4OVIF	TAA4OVMK	0	0	0	TAA4OVPPR2	TAA4OVPPR1	TAA4OVPPR0
FFFFF16EH	TAA4CCIC0	TAA4CCIF0	TAA4CCMK0	0	0	0	TAA4CCPPR02	TAA4CCPPR01	TAA4CCPPR00
FFFFF170H	TAA4CCIC1	TAA4CCIF1	TAA4CCMK1	0	0	0	TAA4CCPPR12	TAA4CCPPR11	TAA4CCPPR10
FFFFF172H	TAA5OVIC	TAA5OVIF	TAA5OVMK	0	0	0	TAA5OVPPR2	TAA5OVPPR1	TAA5OVPPR0
FFFFF174H	TAA5CCIC0	TAA5CCIF0	TAA5CCMK0	0	0	0	TAA5CCPPR02	TAA5CCPPR01	TAA5CCPPR00
FFFFF176H	TAA5CCIC1	TAA5CCIF1	TAA5CCMK1	0	0	0	TAA5CCPPR12	TAA5CCPPR11	TAA5CCPPR10
FFFFF178H	TM0EQIC0	TM0EQIF0	TM0EQMK0	0	0	0	TM0EQPR02	TM0EQPR01	TM0EQPR00
FFFFF17AH	TM1EQIC0	TM1EQIF0	TM1EQMK0	0	0	0	TM1EQPR02	TM1EQPR01	TM1EQPR00
FFFFF17CH	TM2EQIC0	TM2EQIF0	TM2EQMK0	0	0	0	TM2EQPR02	TM2EQPR01	TM2EQPR00
FFFFF17EH	TM3EQIC0	TM3EQIF0	TM3EQMK0	0	0	0	TM3EQPR02	TM3EQPR01	TM3EQPR00
FFFFF180H	CF0RIC/ IICIC1	CF0RIF/ IICIF1	CF0RMK/ IICMK1	0	0	0	CF0RPPR2/ IICPPR12	CF0RPPR1/ IICPPR11	CF0RPPR0/ IICPPR10
FFFFF182H	CF0TIC	CF0TIF	CF0TMK	0	0	0	CF0TPPR2	CF0TPPR1	CF0TPPR0
FFFFF184H	CF1RIC	CF1RIF	CF1RMK	0	0	0	CF1RPPR2	CF1RPPR1	CF1RPPR0
FFFFF186H	CF1TIC	CF1TIF	CF1TMK	0	0	0	CF1TPPR2	CF1TPPR1	CF1TPPR0
FFFFF188H	CF2RIC	CF2RIF	CF2RMK	0	0	0	CF2RPPR2	CF2RPPR1	CF2RPPR0
FFFFF18AH	CF2TIC	CF2TIF	CF2TMK	0	0	0	CF2TPPR2	CF2TPPR1	CF2TPPR0
FFFFF18CH	CF3RIC	CF3RIF	CF3RMK	0	0	0	CF3RPPR2	CF3RPPR1	CF3RPPR0
FFFFF18EH	CF3TIC	CF3TIF	CF3TMK	0	0	0	CF3TPPR2	CF3TPPR1	CF3TPPR0
FFFFF190H	CF4RIC	CF4RIF	CF4RMK	0	0	0	CF4RPPR2	CF4RPPR1	CF4RPPR0
FFFFF192H	CF4TIC	CF4TIF	CF4TMK	0	0	0	CF4TPPR2	CF4TPPR1	CF4TPPR0
FFFFF194H	UC0RIC	UC0RIF	UC0RMK	0	0	0	UC0RPPR2	UC0RPPR1	UC0RPPR0
FFFFF196H	UC0TIC	UC0TIF	UC0TMK	0	0	0	UC0TPPR2	UC0TPPR1	UC0TPPR0
FFFFF198H	UC1RIC/ IICIC2	UC1RIF/ IICIF2	UC1RMK/ IICMK2	0	0	0	UC1RPPR2/ IICPPR22	UC1RPPR1/ IICPPR21	UC1RPPR0/ IICPPR20
FFFFF19AH	UC1TIC	UC1TIF	UC1TMK	0	0	0	UC1TPPR2	UC1TPPR1	UC1TPPR0
FFFFF19CH	UC2RIC	UC2RIF	UC2RMK	0	0	0	UC2RPPR2	UC2RPPR1	UC2RPPR0
FFFFF19EH	UC2TIC	UC2TIF	UC2TMK	0	0	0	UC2TPPR2	UC2TPPR1	UC2TPPR0
FFFFF1A0H	UC3RIC/ IICIC0	UC3RIF/ IICIF0	UC3RMK/ IICMK0	0	0	0	UC3RPPR2/ IICPPR02	UC3RPPR1/ IICPPR01	UC3RPPR0/ IICPPR00
FFFFF1A2H	UC3TIC	UC3TIF	UC3TMK	0	0	0	UC3TPPR2	UC3TPPR1	UC3TPPR0
FFFFF1A4H	UC4RIC	UC4RIF	UC4RMK	0	0	0	UC4RPPR2	UC4RPPR1	UC4RPPR0
FFFFF1A6H	UC4TIC	UC4TIF	UC4TMK	0	0	0	UC4TPPR2	UC4TPPR1	UC4TPPR0
FFFFF1A8H	ADIC	ADIF	ADMK	0	0	0	ADPPR2	ADPPR1	ADPPR0
FFFFF1AAH	DMAIC0	DMAIC0	DMAMK0	0	0	0	DMAPPR02	DMAPPR01	DMAPPR00
FFFFF1ACH	DMAIC1	DMAIC1	DMAMK1	0	0	0	DMAPPR12	DMAPPR11	DMAPPR10

表 23-4. 中断控制寄存器 (xxICn) (3/3)

地址	寄存器	位							
		<7>	<6>	5	4	3	2	1	0
FFFFF1AEH	DMAIC2	DMAIC2	DMAMK2	0	0	0	DMAPPR22	DMAPPR21	DMAPPR20
FFFFF1B0H	DMAIC3	DMAIC3	DMAMK3	0	0	0	DMAPPR32	DMAPPR31	DMAPPR30
FFFFF1B2H	KRIC	KRIF	KRMK	0	0	0	KRPPR2	KRPPR1	KRPPR0
FFFFF1B4H	RTC0IC	RTC0IF	RTC0MK	0	0	0	RTC0PPR2	RTC0PPR1	RTC0PPR0
FFFFF1B6H	RTC1IC	RTC1IF	RTC1MK	0	0	0	RTC1PPR2	RTC1PPR1	RTC1PPR0
FFFFF1B8H	RTC2IC	RTC2IF	RTC2MK	0	0	0	RTC2PPR2	RTC2PPR1	RTC2PPR0
FFFFF1BAH	ERRIC0 ^注	ERRIF0	ERRMK0/	0	0	0	ERRPPR02	ERRPPR01	ERRPPR00
FFFFF1BCH	WUPIC0 ^注	WUPIF0	WUPMK0	0	0	0	WUPPPR02	WUPPPR01	WUPPPR00
FFFFF1BEH	RECIC0 ^注	RECIF0	RECMK0	0	0	0	RECPPR02	RECPPR01	RECPPR00
FFFFF1C0H	TRXIC0 ^注	TRXIF0	TRXMK0/ IEMK2	0	0	0	TRXPPR02/ IEPPR22	TRXPPR01/ IEPPR21	TRXPPR00/ IEPPR20
FFFFF1C8H	UFIC0	UFIC0	UFMK0	0	0	0	UFPPR02	UFPPR01	UFPPR00
FFFFF1CAH	UFIC1	UFIC1	UFMK1	0	0	0	UFPPR12	UFPPR11	UFPPR10

注 仅限于 μ PD70F3770, 70F3771

23.3.5 中断屏蔽寄存器 0 至 5 (IMR0 至 IMR5)

IMR0 至 IMR5 寄存器设置可屏蔽中断的中断屏蔽状态。IMR0 至 IMR5 寄存器的 xxMKn 位等同于 xxCn.xxMKn 位。

IMRm 寄存器可以按 16 位读出或写入。

如果 IMRm 寄存器的高 8 位用作 IMRmH 寄存器，而其低 8 位用作 IMRmL 寄存器（m = 0 至 5），则这些寄存器可以按 8 位或按 1 位为单位进行读写。

复位后这些寄存器设置为 FFFFH。

注意事项 设备文件中将 xxCn.xxMKn 位定义为保留字。如果使用 xxMKn 处理一位，将是 xxCn 寄存器的内容而不是 IMRm 寄存器被重写（结果，IMRm 寄存器的内容也被重写）。

(1/2)

复位后: FFFFH R/W 地址: IMR5 FFFFF10AH, IMR5L FFFFF10AH, IMR5H FFFFF10BH							
IMR5 (IMR5H ^{注1})	15	14	13	12	11	10	9
	1	1	UFMK1	UFMK0	1	1	1
IMR5L	7	6	5	4	3	2	1
	RECMK0 ^{注2}	WUPMK0 ^{注2}	ERRMK0 ^{注2}	RTC2MK	RTC1MK	RTC0MK	KRMK
复位后: FFFFH R/W 地址: IMR4 FFFFF108H, IMR4L FFFFF108H, IMR4H FFFFF109H							
IMR4 (IMR4H ^{注1})	15	14	13	12	11	10	9
	DMAMK2	DMAMK1	DMAMK0	ADMK	UC4TMK	UC4RMK	UC3TMK
IMR4L	7	6	5	4	3	2	1
	UC2TMK	UC2RMK	UC1TMK	UC1RMK/ IICMK2	UC0TMK	UC0RMK	CF3TMK
复位后: FFFFH R/W 地址: IMR3 FFFFF106H, IMR3L FFFFF106H, IMR3H FFFFF107H							
IMR3 (IMR3H ^{注1})	15	14	13	12	11	10	9
	CF3TMK	CF3RMK	CF2TMK	CF2RMK	CF1TMK	CF1RMK	CF0TMK
IMR3L	7	6	5	4	3	2	1
	TM3EQMK0	TM2EQMK0	TM1EQMK0	TM0EQMK0	TAA5CCMK1	TAA5CCMK0	TAA5OVMK
注 1. 若要按字节或按位读取 IMR3 至 IMR5 寄存器的位 8 至 15，需指定这些位作为 IMR3H 至 IMR5H 寄存器的位 0 至 7。							
2. 仅 μ PD70F3770, 70F3771							
注意事项 将 IMR5 寄存器的位 9 至 11, 14, 15 设置为 1。如果这些位的设置改变，将不能保证操作。							
备注	xx: 每一个外设单元的标志名称（参见表 23-4 中断控制寄存器（xxICn））。						
	n: 外设单元编号（参见表 23-4 中断控制寄存器（xxICn））。						

复位后： FFFFH R/W 地址： IMR2 FFFFF104H,
IMR2L FFFFF104H, IMR2H FFFFF105H

	15	14	13	12	11	10	9	8
IMR2 (IMR2H ^{注1})	TAA4CCMK0	TAA4OVMK	TAA3CCMK1	TAA3CCMK0	TAA3OVMK	TAA2CCMK1	TAA2CCMK0	TAA2OVMK
	7	6	5	4	3	2	1	0
IMR2L	TAA1CCMK1	TAA1CCMK0	TAA1OVMK	TAA0CCMK1	TAA0CCMK0	TAA0OVMK	TMTIECMK	TMT0CCMK1

复位后： FFFFH R/W 地址： IMR1 FFFFF102H,
IMR1L FFFFF102H, IMR1H FFFFF103H

	15	14	13	12	11	10	9	8
IMR1 (IMR1H ^{注1})	TT0CCMK0	TT0OVMK	TAB1CCMK3	TAB1CCMK2	TAB1CCMK1	TAB1CCMK0	TAB1OVMK	TAB0CCMK3
	7	6	5	4	3	2	1	0
IMR1L	TAB0CCMK2	TAB0CCMK1	TAB0CCMK0	TAB0OVMK	PMK18	PMK17	PMK16	PMK15

复位后： FFFFH R/W 地址： IMR0 FFFFF100H,
IMR0L FFFFF100H, IMR0H FFFFF101H

	15	14	13	12	11	10	9	8
IMR0 (IMR0H ^{注1})	PMK14	PMK13	PMK12	PMK11	PMK10	PMK09	PMK08	PMK07
	7	6	5	4	3	2	1	0
IMR0L	PMK06 ^{注2}	PMK05	PMK04 ^{注2}	PMK03 ^{注2}	PMK02	PMK01 ^{注2}	PMK00 ^{注2}	LVIMK

xxMKn	中断屏蔽标志的设置
0	允许中断服务
1	禁止中断服务

- 注
1.

若要按字节或按位读取 IMR0 至 IMR2 寄存器的位 8 至 15，需指定这些位作为 IMR0H 至 IMR2H 寄存器的位 0 至位 7。
2.

仅限于 V850ES/JH3-H。

备注

xx:

每一个外设单元的标志名称（参见 表 23-4 中断控制寄存器（xxICn））。

n:

外设单元编号（参见 表 23-4 中断控制寄存器（xxICn））。

23.3.6 服务中优先级寄存器（ISPR）

ISPR 寄存器保存当前响应的可屏蔽中断的优先级。 当响应一个中断请求信号时，本寄存器与该中断请求信号的优先级相对应的位设置为 1，并且，在中断服务期间一直保持。

当执行 RETI 指令时，对应中断请求信号具有最高优先级的位被硬件自动清为 0。 但是，当执行从不可屏蔽中断服务或异常处理中返回时，并不复位为 0。

本寄存器为只读寄存器，可以按 8 位或按位读出。

复位后本寄存器被设置为 00H。

注意事项 在中断允许（EI）状态下，如果读取 ISPR 寄存器同时响应一个中断，可能读取的值为响应中断后其对应置位后的 ISPR 寄存器的值。 为了在中断响应前准确读出 ISPR 寄存器的值，应在中断（DI）禁止读取的状态下读取。

复位后：00H R 地址：FFFFFF1FAH

ISPR

<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
ISPR7	ISPR6	ISPR5	ISPR4	ISPR3	ISPR2	ISPR1	ISPR0

ISPRn	当前应答的中断优先级
0	优先级为n的中断请求信号无应答
1	优先级为n的中断请求信号应答

备注 n = 0 至 7（优先级等级）

23.3.7 ID 标志

该标志控制可屏蔽中断的操作状态，并且存储有关允许或禁止中断请求信号的控制信息。 中断禁止标志（ID）位于 PSW。

系统复位以后，设置该标志为 00000020H。

复位后: 00000020H

	31	8	7	6	5	4	3	2	1	0				
PSW	0						NP	EP	ID	SAT	CY	OV	S	Z

ID	可屏蔽中断服务指定 ^注
0	允许可屏蔽中断请求信号应答
1	禁止可屏蔽中断请求信号应答

注 中断禁止标志（ID）功能

该位通过 DI 指令设为 1，且通过 EI 指令清为 0。 其值也可由 RETI 指令或涉及 PSW 的 LDSR 指令所改变。

不管该标志如何，不可屏蔽中断请求信号和异常事件都得到响应。 当一个可屏蔽中断请求信号被响应时，ID 标志由硬件自动置 1。

当 xxiCn.xxIFn 位设为 1 时，在中断响应禁止期间（ID 标志 = 1）产生的中断请求信号可以得到响应，并且 ID 标志清除为 0。

23.3.8 看门狗定时器模式寄存器 2（WDTM2）

本寄存器可以按字节进行读写（关于详情，参见 第十三章 看门狗定时器 2 的功能）。

复位将该寄存器设置为 67H。

复位后: 67H R/W 地址: FFFFF6D0H

	7	6	5	4	3	2	1	0
WDTM2	0	WDM21	WDM20	0	0	0	0	0

WDM21	WDM20	看门狗定时器操作模式选择
0	0	停止操作
0	1	不可屏蔽中断请求模式
1	×	复位模式 (初始值)

23.4 软件异常

当 CPU 执行 TRAP 指令时，产生软件异常，并且总能够得到响应。

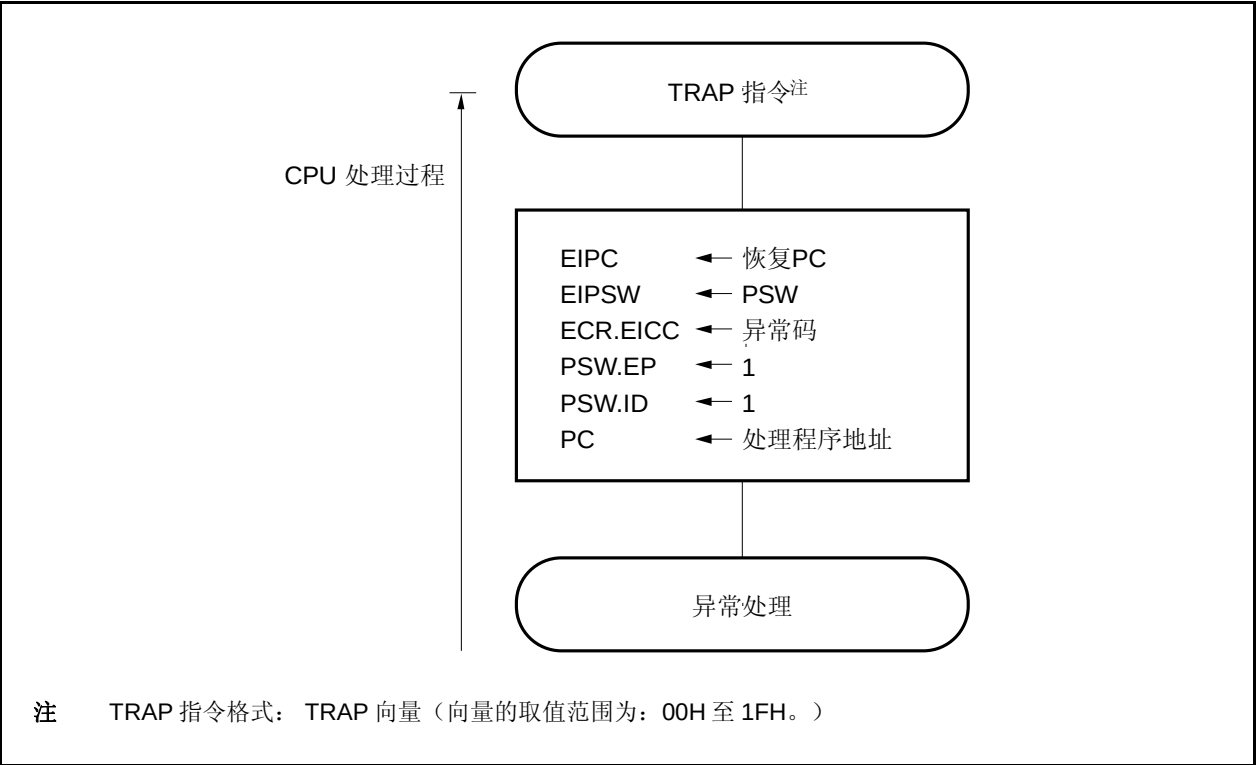
23.4.1 操作

如果发生软件异常，CPU 执行以下处理，并控制转移到处理程序。

- <1> 将 PC 保存至 EIPC。
- <2> 将 PSW 保存至 EIPSW。
- <3> 写入异常码到 ECR（中断源）的低 16 位（EICC）。
- <4> 将 PSW.EP 和 PSW.ID 位设为 1。
- <5> 设置 PC 软件异常的处理程序地址（00000040H 或 00000050H），并进行控制转移。

软件异常的处理过程如下所示：

图 23-9. 软件异常的处理过程



处理程序地址由 TRAP 指令的操作数（向量）决定。如果向量为 00H 至 0FH，它成为 00000040H，而如果向量为 10H 至 1FH，它成为 00000050H。

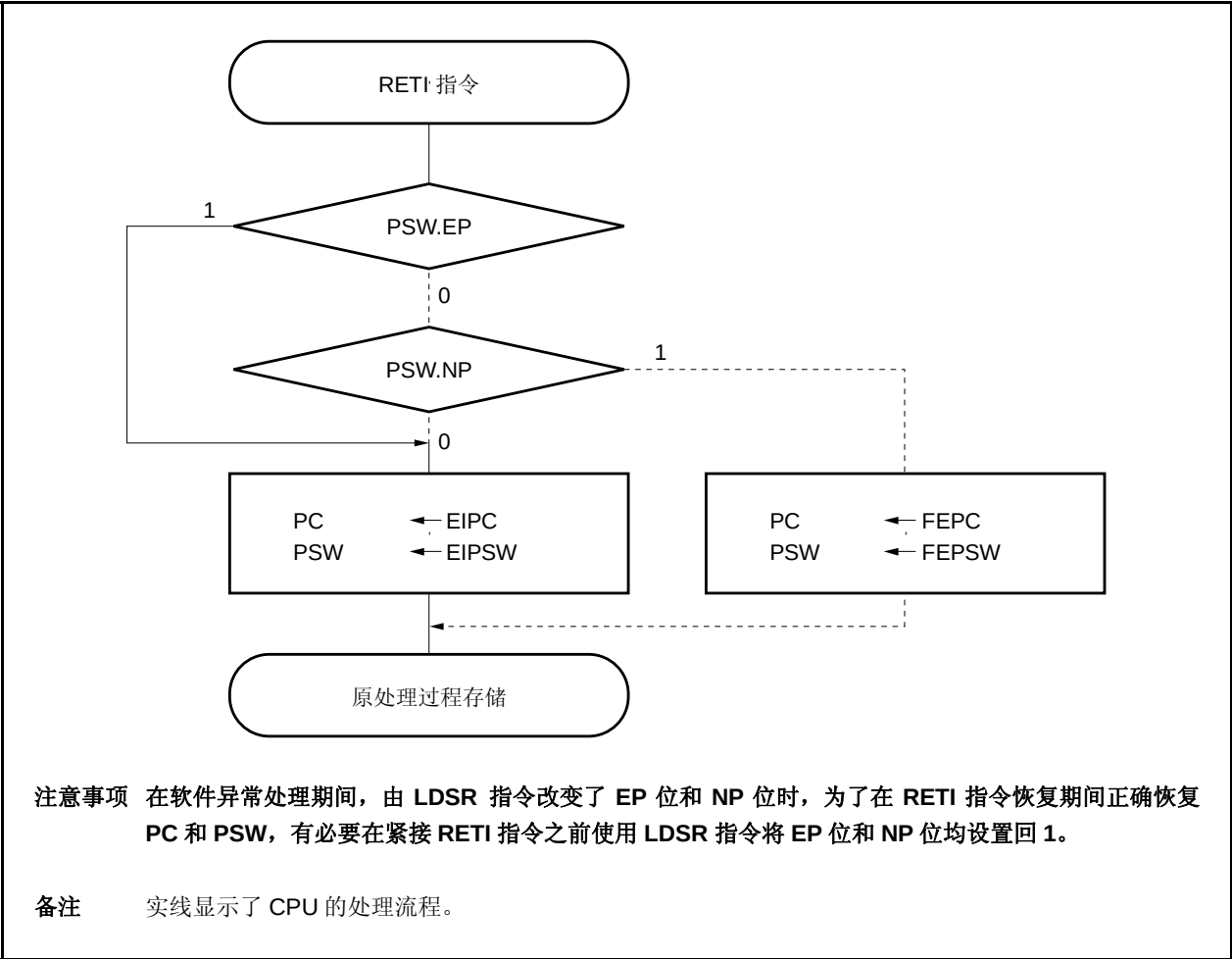
23.4.2 恢复

由 RETI 指令负责从软件异常处理之中进行恢复。
通过执行 RETI 指令，CPU 执行以下处理并控制转移至恢复 PC 的地址。

- <1> 由于 PSW.EP 位为 1，由 EIPC 和 EIPSW 载入恢复 PC 和 PSW。
- <2> 控制转移至恢复 PC 和 PSW 的地址。

RETI 指令的处理过程如下所示：

图 23-10. RETI 指令处理过程



23.4.3 EP标志

EP 为一个状态标志，用来指示异常处理正在处理当中。 当异常发生时被置位。

复位后: 00000020H

PSW

31876543210

0NP EP ID SAT CY OV S Z

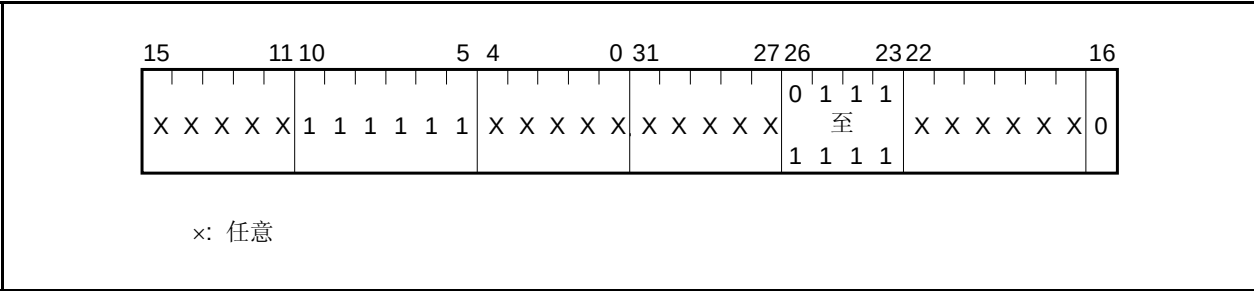
EP	异常处理状态
0	没有进行异常处理。
1	正在进行异常处理。

23.5 异常陷阱

异常陷阱为一个中断，当执行非法指令发生时产生中断请求。 V850ES/JG3-H 和 V850ES/JH3-H 单片机中，将非法操作码异常（ILGOP：非法操作码陷阱）视为异常陷阱。

23.5.1 非法操作码

非法操作码定义为这样一个指令，其指令操作码（位 10 至 5）=11111B，次操作码（位 26 至 23）=0111B 至 1111B，以及次操作码（位 16）=0B。 当这样一个指令被执行时，就产生一个异常陷阱。



注意事项 建议不要使用非法操作码，因为在将来这些指令可能被用作新用途。

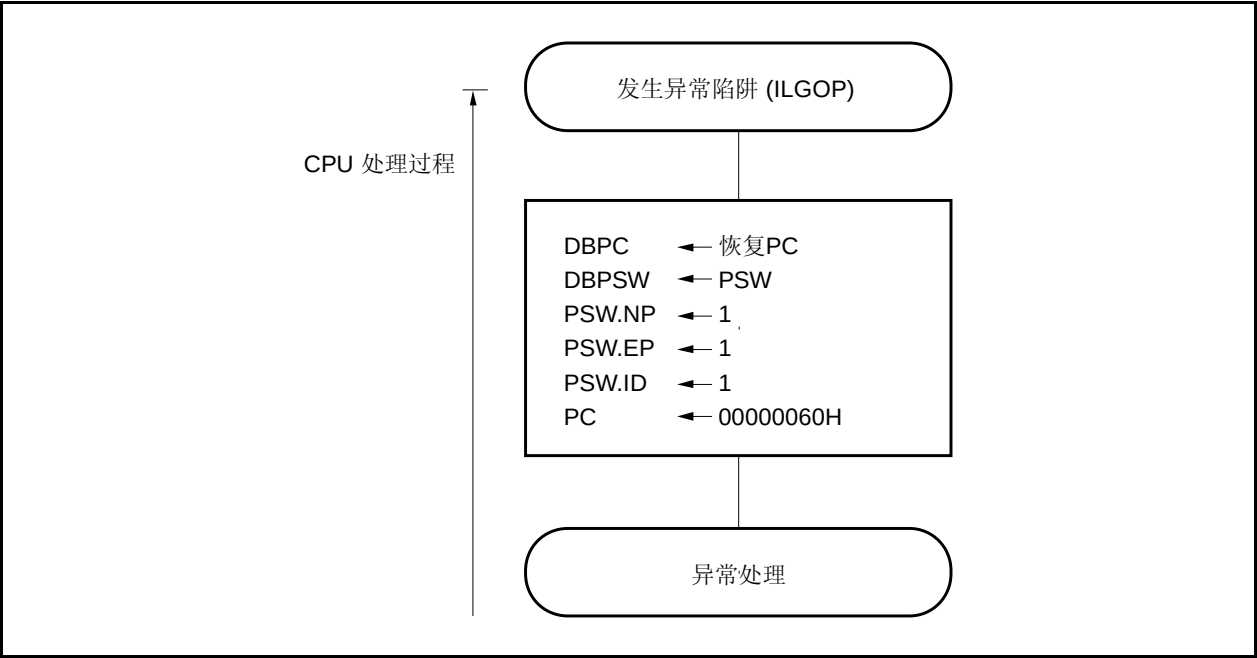
(1) 操作

当发生异常陷阱时， CPU 执行以下处理， 并控制转移到处理程序：

- <1> 将恢复 PC 保存至 DBPC。
- <2> 将当前 PSW 保存至 DBPSW。
- <3> 将 PSW.NP， PSW.EP 和 PSW.ID 位设为 1。
- <4> 设置对应于 PC 异常陷阱的句柄地址（00000060H）， 并进行控制转移。

异常陷阱的处理过程如下所示：

图 23-11. 异常陷阱处理过程



(2) 恢复

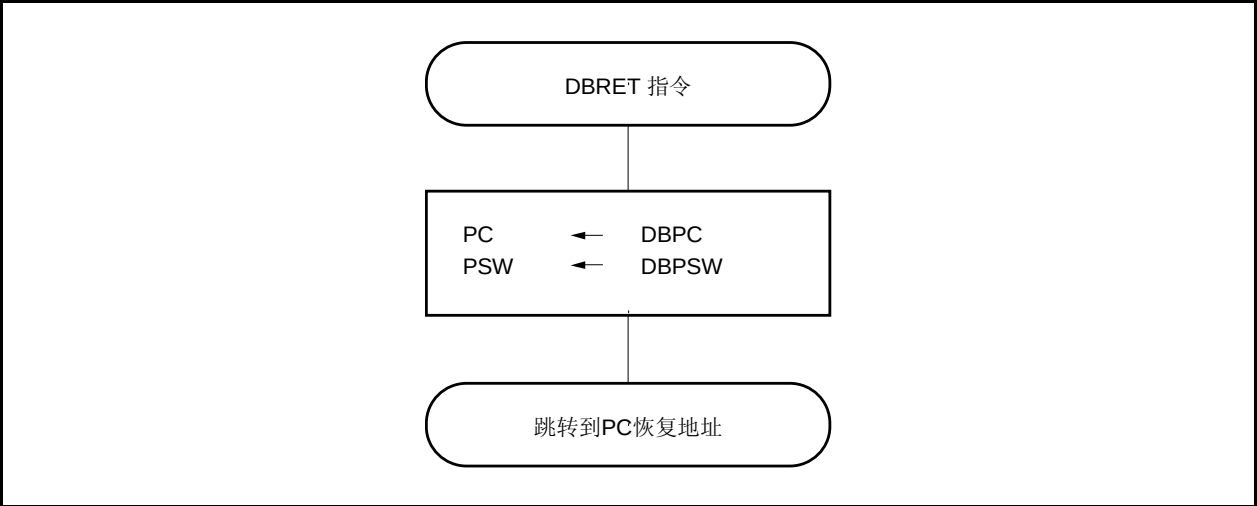
由 DBRET 指令负责从异常陷阱中进行恢复。通过执行 DBRET 指令，CPU 进行以下处理并控制转移至恢复 PC 的地址。

- <1> 从 DBPC 和 DBPSW 中载入恢复 PC 和 PSW。
- <2> 控制转移至恢复 PC 和 PSW 所指定的地址。

注意事项 仅当在处理非法操作码和 DBRET 指令之间的时间间隔内，才可以访问 DBPC 和 DBPSW。

从异常陷阱之中的恢复处理过程如下所示：

图 23-12. 从异常陷阱恢复的处理



23.5.2 调试陷阱

调试陷阱是一个异常，由执行 DBTRAP 指令产生，可以在任何时候都得到响应。

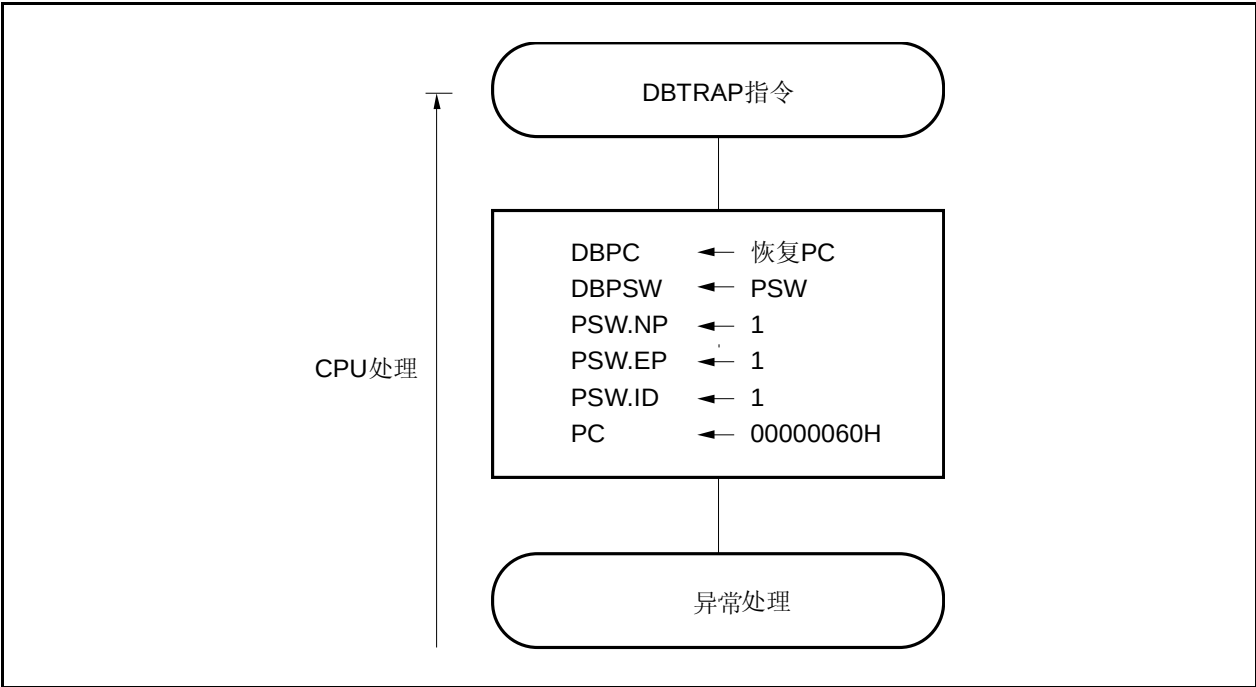
(1) 操作

发生调试陷阱时，CPU 会执行以下处理。

- <1> 将恢复 PC 保存至 DBPC。
- <2> 将当前 PSW 保存至 DBPSW。
- <3> 将 PSW.NP，PSW.EP 和 PSW.ID 位设为 1。
- <4> 将调试陷阱的处理程序地址（00000060H）设置到 PC 并传输控制。

调试陷阱的处理格式如下所示。

图 23-13. 调试陷阱处理格式



(2) 恢复

通过 DBRET 指令执行从调试陷阱中恢复。

使用 DBRET 指令时，CPU 执行以下处理过程并控制转移到恢复 PC 地址。

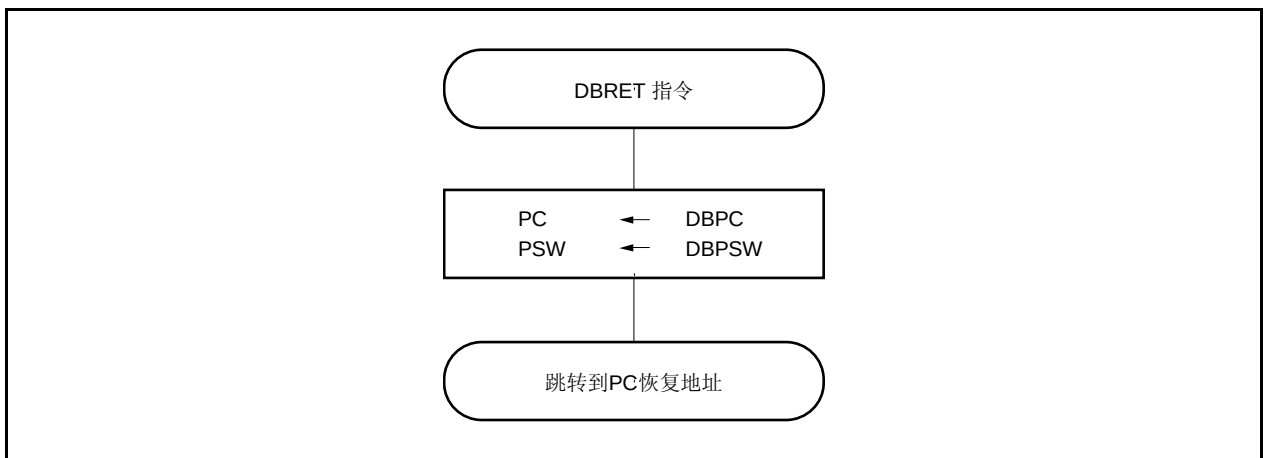
<1> 从 DBPC 和 DBPSW 中读取恢复 PC 和 PSW。

<2> 将控制发送到恢复 PC 和 PSW 的读取地址。

注意事项 仅当在处理非法操作码和 DBRET 指令之间的时间间隔内，才可以访问 DBPC 和 DBPSW。

从调试陷阱中恢复处理的格式如下所示：

图 23-14. 从调试陷阱中恢复的处理流程



23.6 外部中断请求输入引脚（NMI和INTP00至INTP18）

23.6.1 噪声抑制

(1) NMI 引脚的噪声抑制

NMI 引脚具有应用模拟延迟的内部噪声抑制电路。因此，NMI 引脚的输入电平需保持一定的或很长的时间，若非如此，将不能检测到其边沿。因而，经过一定的时间之后，才检测到边沿。

NMI 引脚能用来释放 STOP 模式。在 STOP 模式中，使用系统时钟噪音无法得到抑制，因为此时内部系统时钟是停止的。

(2) INTP00, INTP01 以及 INTP03 至 INTP18 引脚的噪声抑制

INTP00, INTP01 以及 INTP03 至 INTP18 引脚具有应用模拟延迟的内部噪声抑制电路。因此，NMI 引脚的输入电平需保持一定的或很长的时间，若非如此，将不能检测到其边沿。因而，经过一定的时间之后，才检测到边沿。

(3) INTP02 的噪声抑制

INTP02 引脚具有一个使用模拟延迟的内部噪声抑制电路和一个内部数字噪声抑制电路。两个电路都可以使用噪声抑制控制寄存器（INTNFC）选择（参见 23.6.2（7））。

23.6.2 边沿检测

NMI 和 INTP00 至 INTP18 各引脚输入信号的有效边沿可以选择为以下 4 种类型：

- 上升沿
- 下降沿
- 双边沿
- 无检测边沿

在复位后，不检测 NMI 引脚的边沿。因此，除非通过 INTF0 和 INTR0 寄存器允许有效边沿，否则不响应中断请求信号（NMI 引脚用作正常端口引脚）。

(1) 外部中断下降沿和上升沿指定寄存器 0 (INTF0, INTR0)

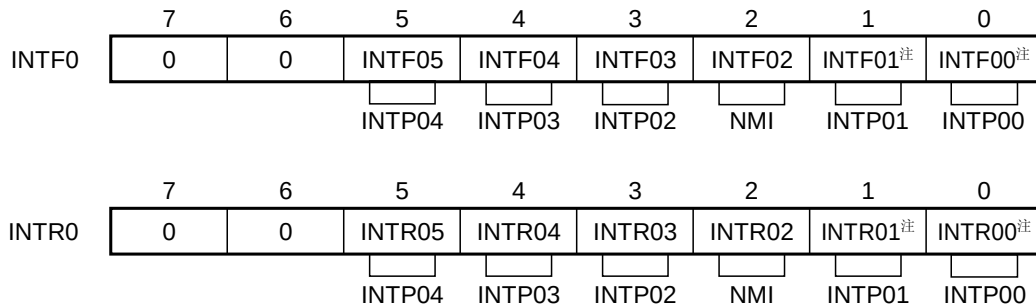
INTF0 和 INTR0 寄存器为 8 位寄存器，用来指明通过位 2 的 NMI 引脚和通过位 0, 1, 3 至 5 的外部中断引脚 (INTP00 至 INTP04) 的下降沿和上升沿检测。

这些寄存器可以按 8 位或按 1 位为单位进行读写。

复位后这些寄存器设置为 00H。

注意事项 当引脚功能由外部中断功能（复用功能）变为端口功能，可能会检测到一个边沿。所以，将 INTF0n 和 INTR0n 位清为 00，然后设置端口模式。

复位后：00H R/W 地址：INTF0 FFFFFFFC00H, INTR0 FFFFFFFC20H



注 仅限于 V850ES/JH3-H。

备注 关于如何指定有效边沿，请参见 表 23-5。

表 23-5. 有效边沿设定

INTF0n	INTR0n	有效边沿设定 (n = 0 至 5)
0	0	无检测边沿
0	1	上升沿
1	0	下降沿
1	1	双边沿

注意事项 当这些寄存器不用作 NMI 或 INTP00 至 INTP04 引脚时，务必将 INTF0n 和 INTR0n 位设为 00。

备注 n = 0, 1: INTP00 和 INTP01 引脚的控制
 n = 2: NMI 引脚控制
 n = 3 至 5: INTP02 至 INTP04 引脚的控制

(2) 外部中断下降沿和上升沿指定寄存器 2 (INTF2, INTR2) (仅限于 V850ES/JH3-H)

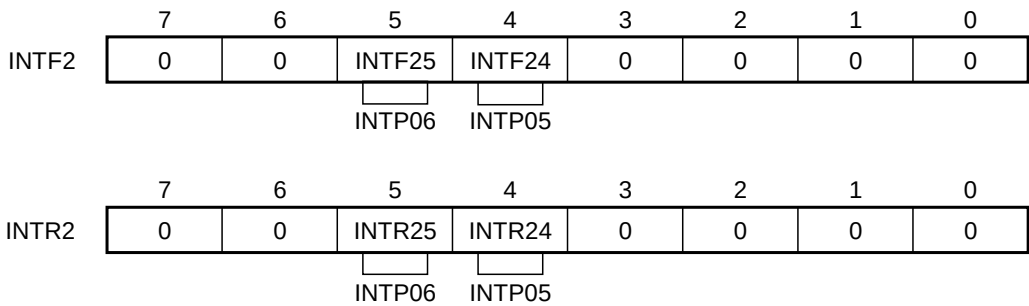
INTF2H 和 INTR2H 寄存器为 8 位寄存器，用来指定外部中断引脚 (INTP05 至 INTP06) 的下降沿和上升沿检测。

这些寄存器可以按 8 位或按 1 位为单位进行读写。

复位后这些寄存器设置为 00H。

注意事项 当引脚功能由外部中断功能 (复用功能) 变为端口功能，可能会检测到一个边沿。所以，将 INTF2n 和 INTR2n 位清为 00，然后设置端口模式。

复位后: 00H R/W 地址: INTF2 FFFFFFFC04H, INTR2 FFFFFFFC24H



备注 关于如何指定有效边沿，请参见 表 23-6。

表 23-6. 有效边沿设定

INTF3n	INTR3n	有效边沿设定
0	0	无检测边沿
0	1	上升沿
1	0	下降沿
1	1	双边沿

注意事项 当不将这些寄存器用作 INTP05 和 INTP06 引脚时，务必将 INTF3n 和 INTR3n 位清为 00。

备注 n = 4, 5: INTP05 和 INTP06 引脚的控制

(3) 外部中断下降沿和上升沿指定寄存器 3 (INTF3, INTR3)

INTF3H 和 INTR3H 寄存器为 8 位寄存器，用来指定外部中断引脚 (INTP07 至 INTP09) 的下降沿和上升沿检测。

这些寄存器可以按 8 位或按 1 位为单位进行读写。

复位后这些寄存器设置为 00H。

注意事项 当引脚功能由外部中断功能（复用功能）变为端口功能，可能会检测到一个边沿。 所以，将 INTF3n 和 INTR3n 位清为 00，然后设置端口模式。

复位后: 00H R/W 地址: INTF3 FFFFFFFC06H, INTR3 FFFFFFFC26H

INTF3

7	6	5	4	3	2	1	0
0	0	0	INTF34	0	0	INTF31	INTF30

INTP09

INTP08

INTP07

INTR3

7	6	5	4	3	2	1	0
0	0	0	INTR34	0	0	INTR31	INTR30

INTP09

INTP08

INTP07

备注 关于如何指定有效边沿，请参见 表 23-7。

表 23-7. 有效边沿设定

INTF3n	INTR3n	有效边沿设定
0	0	无检测边沿
0	1	上升沿
1	0	下降沿
1	1	双边沿

注意事项 不将这些寄存器用作 INTP07 和 INTP09 引脚时，务必将 INTF3n 和 INTR3n 位清为 00。

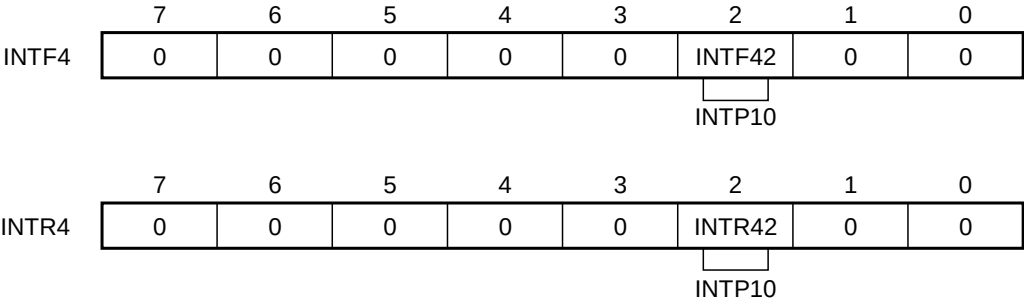
备注 n = 0, 1, 4: INTP07 和 INTP09 引脚的控制

(4) 外部中断下降沿和上升沿指定寄存器 4（INTF4，INTR4）

INTF4 和 INTR4 寄存器为 8 位寄存器，用来指定外部中断引脚（INTP4）的下降沿和上升沿检测。
这些寄存器可以按 8 位或按 1 位为单位进行读写。
复位后这些寄存器设置为 00H。

注意事项 当引脚功能由外部中断功能（复用功能）变为端口功能，可能会检测到一个边沿。 所以，将 INTF42 和 INTR42 位清为 00，然后设置端口模式。

复位后：00H R/W 地址：INTF4 FFFFFC08H, INTR4 FFFFFC28H



备注 关于有效边沿的设定组合，参见 表 23-8。

表 23-8. 有效边沿设定

INTF42	INTR42	有效边沿设定
0	0	无检测边沿
0	1	上升沿
1	0	下降沿
1	1	双边沿

注意事项 如果 INTF42 和 INTR42 位对应的引脚不用作 INTP10 引脚，则务必将它们设为 00。

(5) 外部中断下降沿和上升沿指定寄存器 5（INTF5，INTR5）（仅限于 V850ES/JG3-H）

INTF5 和 INTR5 寄存器为 8 位寄存器，用来指定外部中断引脚（INTP10）的下降沿和上升沿检测。
这些寄存器可以按 8 位或按 1 位为单位进行读写。
复位后这些寄存器设置为 00H。

注意事项 当引脚功能由外部中断功能（复用功能）变为端口功能，可能会检测到一个边沿。 所以，将 INTF56 和 INTR56 位清为 00，然后设置端口模式。

复位后： 00H R/W 地址： INTF5 FFFF00AH, INTR5 FFFF002AH

INTF5

7	6	5	4	3	2	1	0
0	INTF56	0	0	0	0	0	0

INTP05

INTR5

7	6	5	4	3	2	1	0
0	INTR56	0	0	0	0	0	0

INTP05

备注 关于有效边沿的设定组合，参见 表 23-9。

表 23-9. 有效边沿设定

INTF56	INTR56	有效边沿设定
0	0	无检测边沿
0	1	上升沿
1	0	下降沿
1	1	双边沿

注意事项 如果 INTF56 和 INTR56 位对应的引脚不用作 INTP05 引脚，则务必将它们设为 00。

(6) 外部中断下降沿和上升沿指定寄存器 9H (INTF9H, INTR9H)

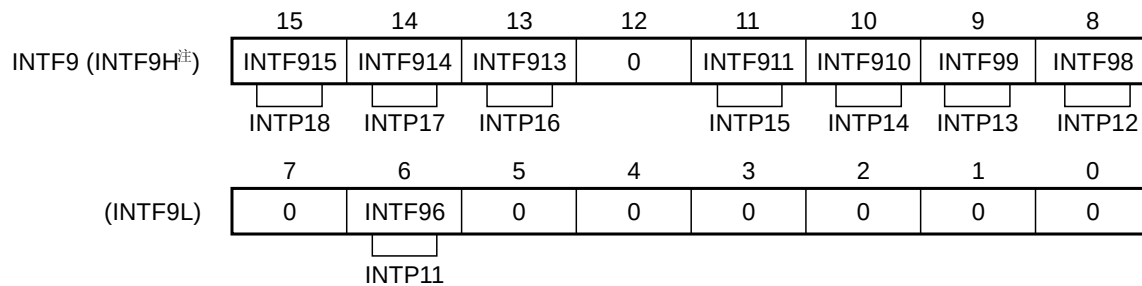
INTF9H 和 INTR9H 寄存器为 8 位寄存器，用来指定外部中断引脚 (INTP11 至 INTP18) 的下降沿和上升沿检测。

这些寄存器可以按 8 位或按 1 位为单位进行读写。

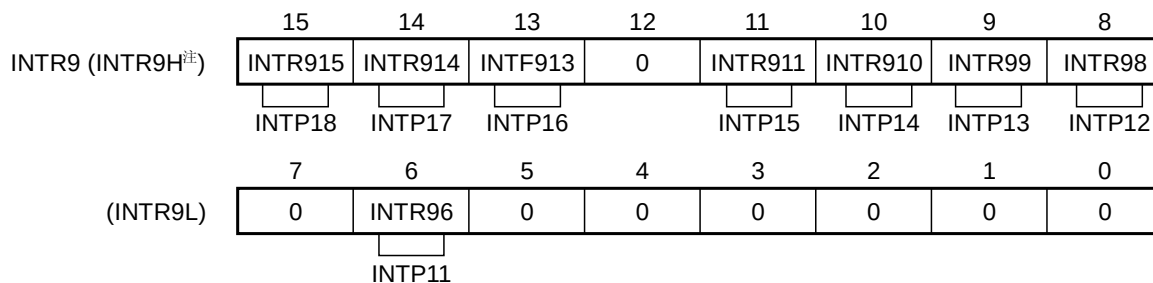
复位后这些寄存器设置为 00H。

注意事项 当引脚功能由外部中断功能（复用功能）变为端口功能，可能会检测到一个边沿。所以，将 INTF9n 和 INTR9n 位清为 00，然后再设置端口模式。

复位后: 00H R/W 地址: INTF9 FFFFC12H,
INTF9H FFFFC12H, INTF9L FFFFC13H



复位后: 00H R/W 地址: INTR9 FFFFC32H,
INTR9H FFFFC32H, INTR9L FFFFC33H



注 若要按字节或按位读取或写入 INTF9 和 INTR9 寄存器的位 8 至 15，需指定这些位为 INTF9H 和 INTR9H 寄存器的位 0 至 7。

备注 关于如何指定有效边沿，请参见 表 23-10。

表 23-10. 有效边沿设定

INTF9n	INTR9n	有效边沿设定 (n = 11 至 18)
0	0	无检测边沿
0	1	上升沿
1	0	下降沿
1	1	双边沿

注意事项 当不将这些寄存器用作 INTP11 和 INTP18 引脚时，务必将 INTF9n 和 INTR9n 位清为 00。

备注 n = 6, 8 至 11, 13 至 15: INTP11 至 INTP18 引脚控制

(7) 噪声抑制控制寄存器 (INTNFC)

有模拟噪声抑制和数字噪声抑制可供 INTP02 引脚选择。使用 INTNFC 寄存器来执行噪声抑制设置。

当选择模拟噪声抑制时，通过将引脚的输入电平保持一段指定时间或更长的时间来检测其是否为边沿。

当选择数字噪声抑制时，数字采样的采样时钟可从以下范围选择： $f_{xx}/64$ ， $f_{xx}/128$ ， $f_{xx}/256$ ， $f_{xx}/512$ ， $f_{xx}/1024$ ，和 f_{XT} 。执行 3 次采样。

即使选择数字噪声抑制，使用 f_{XT} 作为采样时钟也可以使用 INTP02 中断请求信号来释放 IDLE1，IDLE2，以及 STOP 模式成为可能。

该寄存器可按 8 位为单位进行读写。

复位后本寄存器被设置为 00H。

注意事项 在采样时钟改变后，需 3 个采样时钟来初始化数字噪声消除器。因此，在采样时钟改变后，如果一个 INTP02 有效沿在这 3 个采样时钟内输入，可能会产生一个中断请求信号。因此，使用中断和 DMA 功能时要注意以下各点。

- 当使用中断功能时，在 3 个采样时钟过后，中断请求标志 (PIC2.PIF2 位) 被清零后使能中断。
- 当使用 DMA 功能 (由 INTP02 启动) 时，在 3 个采样时钟过后使能 DMA。

复位后: 00H R/W 地址: FFFFFFF28H

	7	6	5	4	3	2	1	0
INTNFC	INTNFEN	0	0	0	0	INTNFC2	INTNFC1	INTNFC0

INTNFEN	INTP02引脚噪声抑制的设置
0	模拟噪声抑制 (60 ns (TYP.))
1	数字噪声抑制

INTNFC2	INTNFC1	INTNFC0	数字采样时钟
0	0	0	$f_{xx}/64$
0	0	1	$f_{xx}/128$
0	1	0	$f_{xx}/256$
0	1	1	$f_{xx}/512$
1	0	0	$f_{xx}/1,024$
1	0	1	f_{XT} (副时钟)
其它			禁止设置

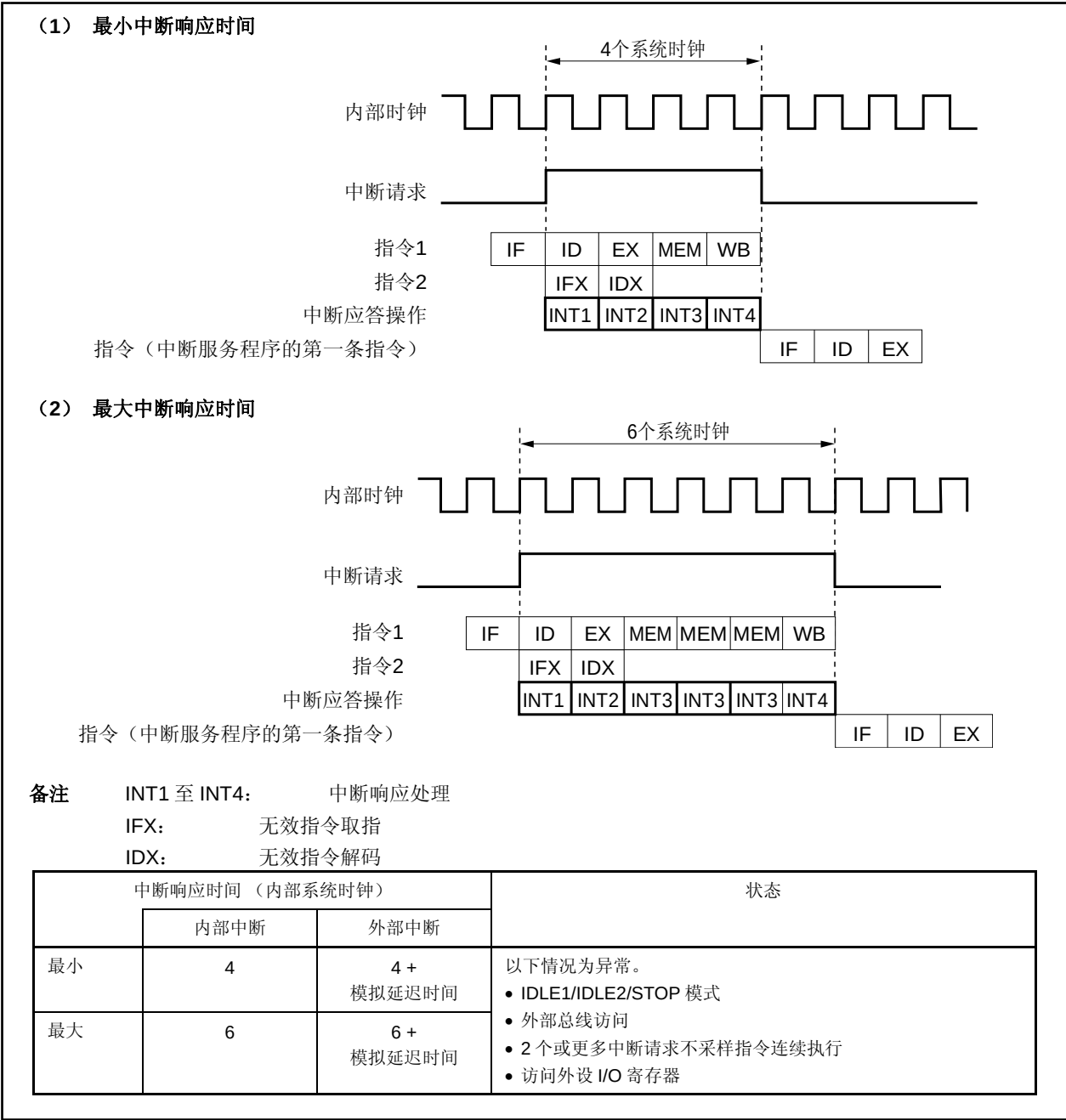
- 备注**
1. 因为采样执行了 3 次，可靠的消除噪音宽度是 2 个采样时钟。
 2. 在噪声宽度小于 2 个采样时钟的情况下，如果采样时钟输入与噪声同步就会产生一个中断请求信号。

23.7 CPU的中断响应时间

除了以下情况之外，CPU 的中断响应时间最小为 4 个时钟周期。若要连续输入中断请求信号，至少需要在前一中断之后 5 个时钟周期输入下一中断请求信号。

- IDLE1/IDLE2/STOP 模式
- 当访问外部总线时
- 此时中断请求不采样指令连续执行（参见 23.8 中断不被 CPU 响应期间。）
- 当访问中断控制寄存器时

图 23-15. 中断请求信号响应的流水作业（概要）



23.8 中断不被CPU响应期间

当指令正在执行时，CPU 可以响应中断。然而，在中断请求不采样指令和下一条指令之间不会响应中断（中断挂起待决）。

中断请求不采样指令如下所示：

- EI 指令
- DI 指令
- LDSR reg2, 0x5 指令（对于 PSW）
- PRCMD 寄存器的存储指令
- 对下列寄存器的存储指令和 SET1, NOT1, CLR1 指令
 - 中断有关寄存器：
中断控制寄存器（xxICn），中断屏蔽寄存器 0 至 5（IMR0 至 IMR5）
 - 节电控制寄存器（PSC）
 - 片上调试模式寄存器（OCDM）

备注 xx: 每一个外设单元的标志名称（参见 表 23-4 中断控制寄存器（xxICn））。
 n: 外设单元编号（参见 表 23-4 中断控制寄存器（xxICn））。

23.9 注意事项

NMI 引脚复用为 P02 引脚，而且在复位后功能用作正常端口引脚。要使能 NMI 引脚，需使 NMI 引脚与 PMC0 寄存器生效。NMI 引脚的初始设置是“无边沿检测”。使用 INTF0 和 INTRO 寄存器选择 NMI 引脚有效沿。

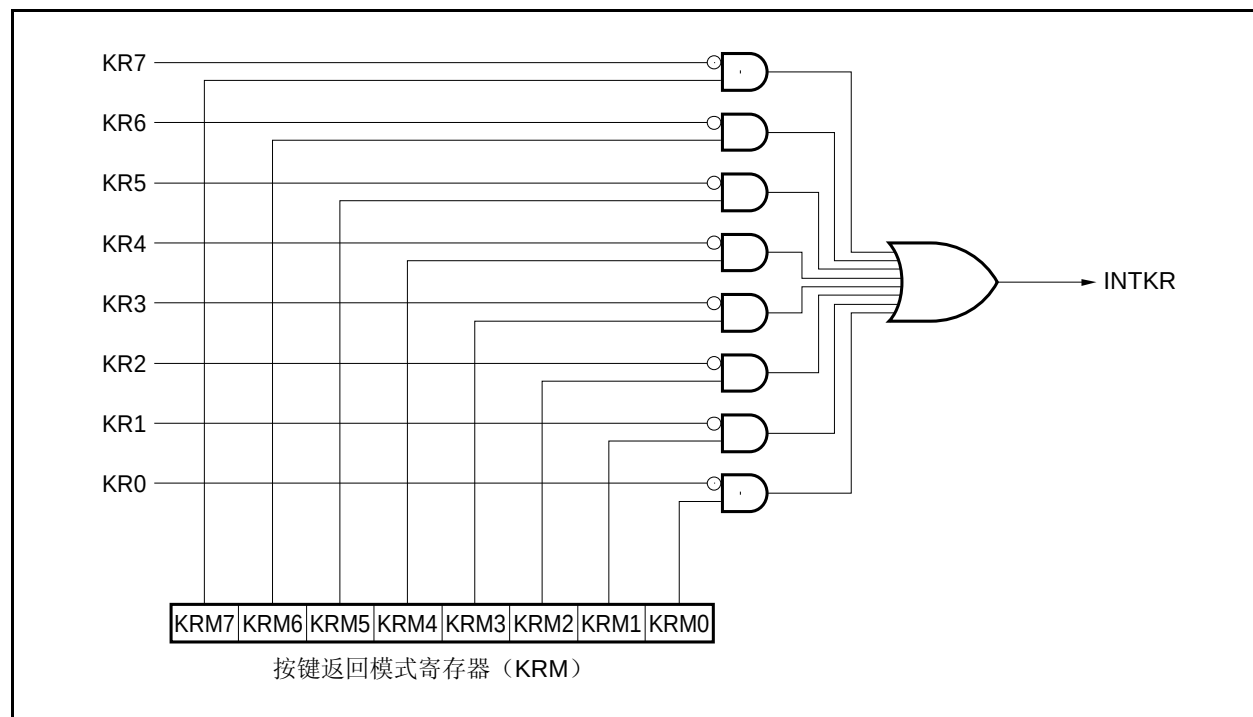
24.1 功能

通过设置 KRM 寄存器，向八个按键输入引脚输入一个下降沿（KR0 至 KR7），可产生一个按键中断请求信号（INTKR）。

表 24-1. 按键返回检测引脚的分配

标志	引脚说明
KRM0	用 1 位单元控制 KR0 信号
KRM1	用 1 位单元控制 KR1 信号
KRM2	用 1 位单元控制 KR2 信号
KRM3	用 1 位单元控制 KR3 信号
KRM4	用 1 位单元控制 KR4 信号
KRM5	用 1 位单元控制 KR5 信号
KRM6	用 1 位单元控制 KR6 信号
KRM7	用 1 位单元控制 KR7 信号

图 24-1. 按键返回框图



24.2 寄存器

(1) 按键返回模式寄存器（KRM）

KRM 寄存器通过使用 KR0 至 KR7 信号来控制 KRM0 至 KRM7 位。
本寄存器可按 8 位或 1 位读写。
复位后寄存器被设置为 00H。

复位后：00H R/W 地址：FFFFFF300H

KRM

7	6	5	4	3	2	1	0
KRM7	KRM6	KRM5	KRM4	KRM3	KRM2	KRM1	KRM0

KRMn	按键返回模式的控制
0	不检查按键返回信号
1	检查按键返回信号

注意事项 将 KRM 寄存器进行一次设置为 00H 后再重写。

备注 关于复用功能引脚的设置， 参见 表 4-20 当端口引脚用作复用功能时的设置。

24.3 注意事项

- (1) 如果 KR0 至 KR7 中任意一引脚输入低电平，则即使其它引脚输入下降沿，也不产生 INTKR 信号。
- (2) RXDC1 和 KR7 引脚不能同时使用。 使用 RXDC1 引脚时，不要使用 KR7 引脚功能。 若使用 KR7 引脚，就不要使用 RXDC1 引脚功能（建议将 PFC91 位设置为 1 并清除 PFCE91 位为 0）。
- (3) 如果改变 KRM 寄存器，就会产生一个中断请求信号（INTKR）。 为防止此种情况，在禁止中断（DI）或屏蔽之后改变 KRM 寄存器，然后，将中断请求标志（KRIC.KRIF 位）清零后允许中断（EI）或取消屏蔽。
- (4) 若要使用按键中断功能，需确保将端口引脚设置为按键返回引脚，然后允许 KRM 寄存器操作。 若要从按键返回引脚转换至端口引脚，需禁止 KRM 寄存器操作而后设置端口引脚。

第二十五章 待机功能

25.1 概述

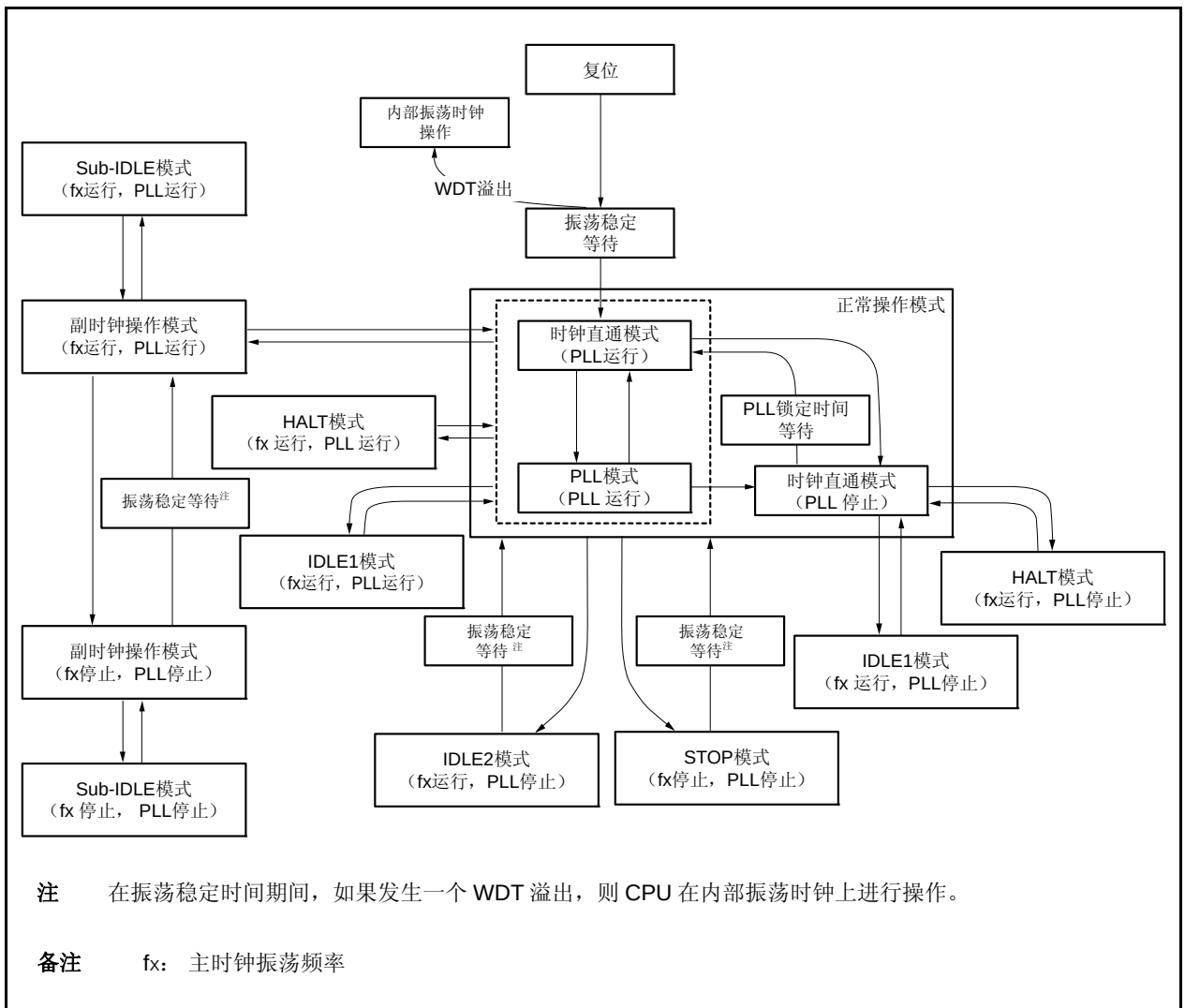
通过结合运用待机模式并为应用程序选择恰当模式，可以有效地减少系统的功耗。 可用待机模式列于表 25-1 中：

表 25-1. 待机模式

模式	功能概述
HALT 模式	此模式下，仅停止 CPU 的工作时钟
IDLE1 模式	此模式下，除振荡器外，PLL 和 flash 存储器之外，停止所有其它内部电路的操作。
IDLE2 模式	此模式下，除振荡器外，停止所有其他内部电路的操作。
STOP 模式	此模式下，除副时钟振荡器外，停止所有其他内部电路的操作。
副时钟工作模式	此模式下，副时钟用作内部系统时钟。
Sub-IDLE 模式	此模式下，除振荡器外，停止副时钟工作模式下的所有内部电路的操作。

注 PLL 保持原来的工作状态。

图 25-1. 状态转换



25.2 寄存器

(1) 节电控制寄存器（PSC）

PSC 寄存器是控制待机功能的 8 位寄存器。该寄存器的 STP 位用来指定 STOP 模式。该寄存器为专用寄存器，只能通过特定的序列组合来写入（参见 3.4.8 专用寄存器）。
本寄存器可按 8 位或 1 位读写。
复位后本寄存器被设置为 00H。

复位后：00H R/W 地址：FFFFFF1FEH

	7	<6>	<5>	<4>	3	2	<1>	0
PSC	0	NMI1M	NMI0M	INTM	0	0	STP	0

NMI1M	通过INTWDT2信号解除待机模式的控制
0	允许通过INTWDT2信号解除待机模式
1	禁止通过INTWDT2信号解除待机模式

NMI0M	通过NMI引脚输入解除待机模式的控制
0	允许通过NMI引脚输入解除待机模式
1	禁止通过NMI引脚输入解除待机模式

INTM	通过可屏蔽中断请求信号解除待机模式的控制
0	允许通过可屏蔽中断请求信号解除待机模式
1	禁止通过可屏蔽中断请求信号解除待机模式

STP	待机模式 ^注 设置
0	正常模式
1	待机模式

注 待机模式由 STP 位设置 IDLE1，IDLE2，STOP 或 sub-IDLE 模式

- 注意事项
1. 设置 IDLE1，IDLE2，STOP，或 sub-IDLE 模式之前，设置 PSMR.PSM1 和 PSMR.PSM0 位，然后设置 STP 位。
 2. 当解除 HALT 模式时，NMI1M，NMI0M，和 INTM 位的设置无效。
 3. 如果在把 NMI1M、NMI0M、或 INTM 位设为 1 的同时把 STP 位也设为 1，则对 NMI1M、NMI0M、或 INTM 位的设置就是无效的。若在设置了 IDLE1/IDLE2/STOP 模式时有未屏蔽中断请求信号处于挂起状态，则应设置相应的中断请求信号位（NMI1M、NMI0M 或 INTM）为 1，再将 STP 位设为 1。

(2) 节电模式寄存器 (PSMR)

PSMR 寄存器为 8 位寄存器，控制节电模式下的操作状态及时钟操作。
本寄存器可按 8 位或 1 位读写。
复位后本寄存器被设置为 00H。

复位后：00H R/W 地址：FFFFF820H

	7	6	5	4	3	2	<1>	<0>
PSMR	0	0	0	0	0	0	PSM1	PSM0

PSM1	PSM0	软件待机模式中的操作说明
0	0	IDLE1, sub-IDLE 模式
0	1	STOP 模式
1	0	IDLE2, sub-IDLE 模式
1	1	STOP 模式

- 注意事项 1. 务必将 2 至 7 位设置为 0。”
2. PSM0 和 PSM1 位仅在 PSC.STP 位为 1 时有效。

备注 IDLE1: 此模式下，除振荡器操作和其它一些电路（flash 存储器和 PLL）之外，停止所有的操作。
在解除 IDLE1 模式后，正常操作模式恢复无需振荡稳定时间的保证，像 HALT 模式一样。

IDLE2: 此模式下，除振荡器操作之外，停止所有的操作。
解除 IDLE2 模式后，在通过 OSTS 寄存器（flash 存储器和 PLL）指定的设置时间过后，正常操作模式恢复。

STOP: 此模式下，除副时钟振荡器操作之外，停止所有的操作。
解除 STOP 模式后，在通过 OSTS 寄存器指定的振荡稳定时间过后，正常操作模式恢复。

Sub-IDLE: 此模式下，除振荡器之外，暂停所有的操作。 在 IDLE 模式通过中断请求信号解除后，在 12 个副时钟周期后，副时钟操作模式将会被恢复。

(3) 振荡稳定时间选择寄存器（OSTS）

解除 STOP 模式后至振荡稳定前的等待时间或在解除 IDLE2 模式后至片上 flash 存储器稳定前的等待时间由 OSTS 寄存器控制。

该寄存器可按 8 位为单位进行读写。

复位设置将寄存器清为 06H。

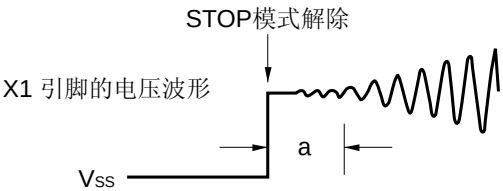
复位后：06H R/W 地址：FFFFF6C0H

	7	6	5	4	3	2	1	0
OSTS	0	0	0	0	0	OSTS2	OSTS1	OSTS0

OSTS2	OSTS1	OSTS0	振荡稳定时间/建立时间的选择 ^注	fx	
				3 MHz	6 MHz
0	0	0	$2^{10}/f_x$	0.341 ms	0.171 ms
0	0	1	$2^{11}/f_x$	0.683 ms	0.341 ms
0	1	0	$2^{12}/f_x$	1.365 ms	0.683 ms
0	1	1	$2^{13}/f_x$	2.730 ms	1.365 ms
1	0	0	$2^{14}/f_x$	5.461 ms	2.731 ms
1	0	1	$2^{15}/f_x$	10.923 ms	5.461 ms
1	1	0	$2^{16}/f_x$	21.85 ms	10.92 ms
1	1	1	禁止设置		

注 当分别解除 STOP 模式和 IDLE2 模式时，振荡稳定时间和建立时间是必须的。

注意事项 1. 不管解除 STOP 模式是因为复位输入还是因为出现了中断请求信号，解除 STOP 模式之后的等待时间不包括解除 STOP 模式之后直到时钟振荡开始之前（下图的“a”）的那段时间。”



- 2. 务必将 3 至 7 位设置为 0。
- 3. 复位之后的振荡稳定时间为 $2^{16}/f_x$ （因为 OSTS 寄存器的初始值= 06H）。

备注 f_x = 主时钟振荡器频率

25.3 HALT 模式

25.3.1 设置和操作状态

当在正常操作模式下执行一个专用指令（HALT）时，HALT 模式被设置。

在 HALT 模式下，时钟振荡器继续工作。只是停止 CPU 的时钟供给，其它片上外围功能的时钟供给照旧。

因此，程序停止执行，内部 RAM 的内容保持设置为 HALT 模式之前的内容。独立于 CPU 指令处理的片上外围功能继续工作。

表 25-3 显示了 HALT 模式下的操作状态。

通过在 HALT 模式下结合使用正常操作模式作为间歇，可以减少系统的平均电流消耗。

注意事项 1. 应在 HALT 指令后插入五个或更多 NOP 指令。

2. 如果有一个未屏蔽的中断请求信号被保持等待响应时执行 HALT 指令，则状态切换至 HALT 模式，但是 HALT 模式立即被此等待响应的中断请求解除。

25.3.2 解除 HALT 模式

在 HALT 模式下来自可操作的外设功能的不可屏蔽中断请求信号（NMI 引脚输入，INTWDT2 信号）、未屏蔽外部中断请求信号（INTP00 至 INTP18 引脚输入）、未屏蔽的内部中断请求信号，或复位信号（由 $\overline{\text{RESET}}$ 引脚输入复位、WDT2RES 信号，低电压检测器（LVI），或时钟监视器），可解除 HALT 模式。

HALT 模式释放后，正常操作模式恢复。

(1) 通过不可屏蔽中断请求信号或未屏蔽的可屏蔽中断请求信号解除 HALT 模式

通过不可屏蔽中断请求信号或未屏蔽的可屏蔽中断请求信号，可解除 HALT 模式，且与中断请求信号的优先级无关。但是，若 HALT 模式被设置在一个中断服务程序中，则之后发出的中断请求信号按如下所述来执行：

- (a) 若发出的中断请求信号优先级低于当前正在接受服务的中断请求，则仅 HALT 模式解除，并且那个中断请求信号未被响应。该中断请求信号被保持。
- (b) 若发出的中断请求信号优先级高于当前正在接受服务的中断请求（包括不可屏蔽中断请求信号），则 HALT 模式解除，且那个中断请求信号被响应。

表 25-2. 用中断请求信号解除 HALT 模式后的操作

解除源	中断允许（EI）状态	中断禁止（DI）状态
不可屏蔽中断请求信号	执行对处理地址的跳转。	
可屏蔽中断请求信号	执行对处理地址的跳转或执行下一条指令。	执行下一条指令。

(2) 通过复位解除 HALT 模式

执行与正常的复位操作相同的操作。

表 25-3. HALT 模式下的操作状态

项目		HALT 模式的设置		操作状态	
				不使用副时钟时	使用副时钟时
主时钟振荡器 (fx)				振荡允许	
副时钟振荡器 (fxT)				—	振荡允许
内部振荡器 (fR)				振荡允许	
PLL				可运行	
CPU				停止运行	
DMA 控制器				可运行	
中断控制器				可运行	
定时器	TAA0 至 TAA5			可运行	
	TAB0, TAB1			可运行	
	TMM0 至 TMM3			当选择非 fxT 时钟为计数时钟时可操作	可运行
	TMT0			可运行	
实时计数器 (RTC)				当 fx (BRG 分频) 选择为计数时钟时可操作	可运行
看门狗定时器 2 (WDT2)				当选择非 fxT 时钟为计数时钟时可操作	可运行
串行接口	CSIF0 至 CSIF4			可运行	
	I ² C00 至 I ² C02			可运行	
	UARTC0 至 UARTC4			可运行	
A/D 转换器				可运行	
D/A 转换器				可运行	
实时输出功能 (RTO)				可运行	
按键中断功能 (KR)				可运行	
CRC 算法电路				可运行 (因为 CPU 停止所以没有数据输入到 CRCIN 寄存器)	
外部总线接口				参见 第五章 总线控制功能	
端口功能				保持设置为 HALT 模式前的状态	
内部数据				CPU 寄存器、状态、数据及其它内部数据, 例如内部 RAM 的内容, 被保持为设置 HALT 模式之前的状态。	
CAN [※]				可运行	
USB 功能				可运行	

注 仅限于 μ PD70F3770, 70F3771

25.4 IDLE1 模式

25.4.1 设置和操作状态

在正常操作模式下，通过将 PSMR.PSM1 和 PSMR.PSM0 位设为 00 并将 PSC.STP 位设为 1 来设置 IDLE1 模式。

在 IDLE1 模式下，时钟振荡器，PLL 和 flash 存储器会继续操作但停止给 CPU 和其它片上外围功能的时钟供应。

因此，程序停止执行，且内部 RAM 的内容保持与设置为 IDLE1 模式前的内容一致。CPU 及其它片上外围功能停止工作。而可以在副时钟或外部时钟下工作的片上外围功能继续工作。

表 25-5 显示了 IDLE1 模式下的操作状态。

与 HALT 模式相比，IDLE1 模式更能减少功耗，因为停止了片上外围功能的运行。由于主时钟振荡器不停止，因此，IDLE1 模式解除后，无需等待振荡稳定时间就可以恢复到正常操作模式，HALT 模式解除时亦如此。

注意事项 1. 应在 PSC 寄存器存储数据指令后插入五个或更多 NOP 指令来设置 IDLE1 模式。

2. 如果有一个未屏蔽的中断请求信号被保持等待响应时设置 IDLE1 模式，则 IDLE1 模式立即被此等待响应的中断请求解除。

25.4.2 解除IDLE1 模式

在 IDLE1 模式下来自可操作的外设功能的不可屏蔽中断请求信号（NMI 引脚输入，INTWDT2 信号）、未屏蔽外部中断请求信号（INTP00 至 INTP18 引脚输入）、未屏蔽的内部中断请求信号，或复位信号（由 $\overline{\text{RESET}}$ 引脚输入复位、WDT2RES 信号，低电压检测器（LVI），或时钟监视器），可解除 IDLE1 模式。

IDLE1 模式解除后，恢复正常操作模式。

(1) 通过不可屏蔽中断请求信号或未屏蔽的可屏蔽中断请求信号解除 IDLE1 模式

通过不可屏蔽中断请求信号或未屏蔽的可屏蔽中断请求信号，可解除 IDLE1 模式，且与中断请求信号的优先级无关。如果在某个中断服务程序中设置 IDLE1 模式，则之后产生的中断请求信号按如下所述来进行处理：

- (a) 若发出的中断请求信号优先级低于当前正在接受服务的中断请求，则仅 IDLE1 模式解除，并且那个中断请求信号未被响应。该中断请求信号被保持。
- (b) 若发出的中断请求信号优先级高于当前正在接受服务的中断请求（包括不可屏蔽中断请求信号），则 IDLE1 模式解除，且那个中断请求信号被响应。

注意事项 通过设置 PSC.NMI1M、PSC.NMI0M、PSC.INTM 位为 1 而禁止的中断请求信号变为无效，因此 IDLE1 模式不会解除。

表 25-4.通过中断请求信号解除 IDLE1 模式后的操作

解除源	中断允许（EI）状态	中断禁止（DI）状态
不可屏蔽中断请求信号	确定指定的建立时间过后，执行对处理地址的跳转。	
可屏蔽中断请求信号	在确保建立时间之后，执行对处理地址的跳转或执行下一条指令。	在确保建立时间之后执行下一条指令。

(2) 通过复位解除 IDLE1 模式

执行与正常的复位操作相同的操作。

表 25-5. IDLE1 模式下的操作状态

IDLE1 模式的设置 项目		操作状态	
		不使用副时钟时	使用副时钟时
主时钟振荡器 (f _x)		振荡允许	
副时钟振荡器 (f _{xT})		—	振荡允许
内部振荡器 (f _R)		振荡允许	
PLL		可运行	
CPU		停止运行	
DMA 控制器		停止运行	
中断控制器		停止运行（但可能会解除待机模式）	
定时器	TAA0 至 TAA5	停止运行	
	TAB0, TAB1	停止运行	
	TMM0 至 TMM3	当 f _R /8 选择为计数时钟时可操作	当 f _R /8 或 f _{xT} 选择为计数时钟时可操作
	TMT0	停止运行	
实时计数器 (RTC)		当 f _x (BRG 分频) 选择为计数时钟时可操作	可运行
看门狗定时器 (WDT2)		当 f _R 选择为计数时钟时可操作	当 f _R 或 f _{xT} 选择为计数时钟时可操作
串行接口	CSIF0 至 CSIF4	SCKFn 输入时钟被选为计数时钟时可运行 (n = 0 至 4)	
	I ² C00 至 I ² C02	停止运行	
	UARTC0 至 UARTC4	停止运行（但是当选择 ASCKC0 输入时钟时，UARTC0 可运行）	
A/D 转换器		保持运行（保持转换结果） ^{※1}	
D/A 转换器		保持运行（保持输出 ^{※1} ）	
实时输出功能 (RTO)		停止运行（保持输出）	
按键中断功能 (KR)		可运行	
CRC 算法电路		停止运行	
外部总线接口		参见 第五章 总线控制功能	
端口功能		保持设置为 IDLE1 模式前的状态	
内部数据		CPU 寄存器、状态、数据及其它内部数据，例如内部 RAM 的内容，被保持为设置 IDLE1 模式之前的状态。	
CAN ^{※2}		停止运行	
USB 功能		停止运行	

注 1. 为了实现降低功率损耗，在切换到 IDLE1 模式前，停止 A/D 转换器和 D/A 转换器。

2. 仅限于 μ PD70F3770, 70F3771

25.5 IDLE2 模式

25.5.1 设置和操作状态

在正常操作模式下，通过将 PSMR.PSM1 和 PSMR.PSM0 位设为 10 并将 PSC.STP 位设为 1 来设置 IDLE2 模式。

在 IDLE2 模式下，时钟振荡器会继续操作但停止给 CPU，PLL 和 flash 存储器以及其它片上外围功能的时钟供应。

因此，程序停止执行，且内部 RAM 的内容保持与设置为 IDLE2 模式前的内容一致。CPU，PLL 及其它片上外围功能停止工作。但是，可以在副时钟或外部时钟下工作的片上外围功能继续工作。

表 25-7 显示了 IDLE2 模式下的操作状态。

与 IDLE1 模式相比，IDLE2 模式更能减少功耗，因为停止了片上外围功能、PLL 和 flash 存储器的运行。但是，因为 PLL 和 flash 存储器停止，当解除 IDLE2 模式时，PLL 和 flash 存储器需要一段建立时间。

注意事项 1. 应在 PSC 寄存器存储数据指令后插入五个或更多 NOP 指令来设置 IDLE2 模式。

2. 如果有一个未屏蔽的中断请求信号被保持等待响应时设置 IDLE2 模式，则 IDLE2 模式立即被此等待响应的中断请求解除。

25.5.2 解除IDLE2 模式

在 IDLE2 模式下来自可操作的外设功能的不可屏蔽中断请求信号（NMI 引脚输入，INTWDT2 信号）、未屏蔽外部中断请求信号（INTP0 至 INTP18 引脚输入）、未屏蔽的内部中断请求信号，或复位信号（由 $\overline{\text{RESET}}$ 引脚输入复位、WDT2RES 信号，低电压检测器（LVI），或时钟监视器），可解除 IDLE2 模式。PLL 返回到设置为 IDLE2 模式前的操作状态。

IDLE2 模式解除后，恢复正常操作模式。

(1) 通过不可屏蔽中断请求信号或未屏蔽的可屏蔽中断请求信号解除 IDLE2 模式

通过不可屏蔽中断请求信号或未屏蔽的可屏蔽中断请求信号，可解除 IDLE2 模式，且与中断请求信号的优先级无关。如果在某个中断服务程序中设置 IDLE2 模式，则之后产生的中断请求信号按如下所述来进行处理：

(a) 若发出的中断请求信号优先级低于当前正在接受服务的中断请求，则仅 IDLE2 模式解除，并且那个中断请求信号未被响应。该中断请求信号被保持。

(b) 若发出的中断请求信号优先级高于当前正在接受服务的中断请求（包括不可屏蔽中断请求信号），则 IDLE2 模式解除，且那个中断请求信号被响应。

注意事项 通过设置 PSC.NMI1M、PSC.NMI0M、PSC.INTM 位为 1 而禁止的中断请求信号变为无效，因此 IDLE2 模式不会解除。

表 25-6.通过中断请求信号解除 IDLE2 模式后的操作

解除源	中断允许 (EI) 状态	中断禁止 (DI) 状态
不可屏蔽中断请求信号	确定指定的建立时间过后，执行对处理地址的跳转。	
可屏蔽中断请求信号	在确保建立时间之后，执行对处理地址的跳转或执行下一条指令。	在确保建立时间之后执行下一条指令。

(2) 通过复位解除 IDLE2 模式

执行与正常的复位操作相同的操作。

表 25-7. IDLE2 模式下的操作状态

IDLE2 模式的设置 项目		操作状态	
		不使用副时钟时	使用副时钟时
主时钟振荡器 (f_x)		振荡允许	
副时钟振荡器 (f_{XT})		—	振荡允许
内部振荡器 (f_R)		振荡允许	
PLL		停止运行	
CPU		停止运行	
DMA 控制器		停止运行	
中断控制器		停止运行	
定时器	TAA0 至 TAA5	停止运行	
	TAB0, TAB1	停止运行	
	TMM0 至 TMM3	当 $f_R/8$ 选择为计数时钟时可操作	当 $f_R/8$ 或 f_{XT} 选择为计数时钟时可操作
	TMT0	停止运行	
实时计数器 (RTC)		当 f_x (BRG 分频) 选择为计数时钟时可操作	可运行
看门狗定时器 (WDT2)		当 f_R 选择为计数时钟时可操作	当 f_R 或 f_{XT} 选择为计数时钟时可操作
串行接口	CSIF0 至 CSIF4	SCKFn 输入时钟被选为计数时钟时可运行 ($n = 0$ 至 4)	
	I ² C00 至 I ² C02	停止运行	
	UARTC0 至 UARTC4	停止运行 (但是当选择 ASCKC0 输入时钟时, UARTC0 可运行)	
A/D 转换器		保持运行 (保持转换结果) ^{注1}	
D/A 转换器		保持运行 (保持输出 ^{注1})	
实时输出功能 (RTO)		停止运行 (保持输出)	
按键中断功能 (KR)		可运行	
CRC 算法电路		停止运行	
外部总线接口		参见 第五章 总线控制功能	
端口功能		保持设置为 IDLE2 模式前的状态	
内部数据		CPU 寄存器、状态、数据及其它内部数据, 例如内部 RAM 的内容, 被保持为设置 IDLE2 模式之前的状态。	
CAN ^{注2}		停止运行	
USB 功能		停止运行	

注 1. 为了实现降低功率损耗, 在切换到 IDLE2 模式前, 停止 A/D 转换器和 D/A 转换器。

2. 仅限于 μ PD70F3770, 70F3771。

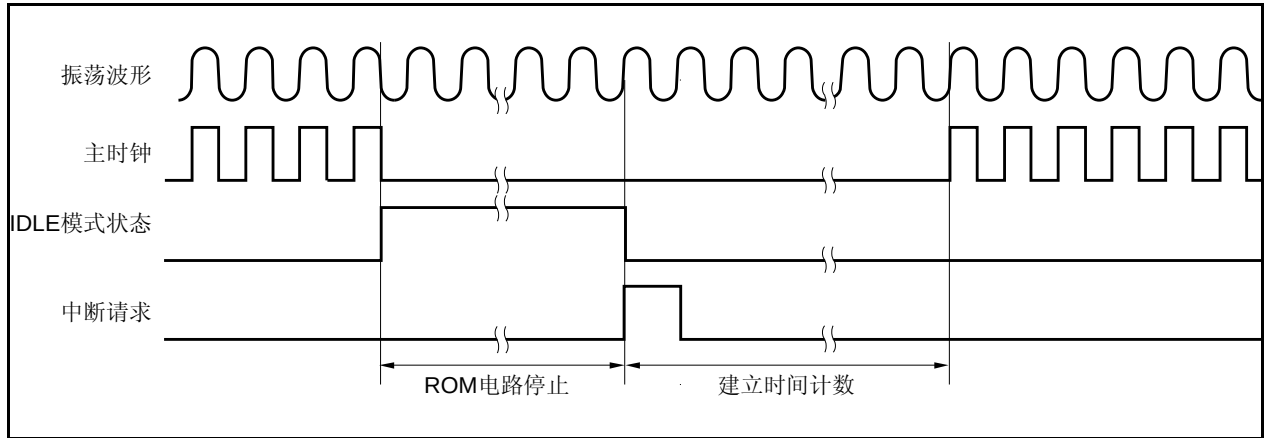
25.5.3 解除IDLE2 模式时确保建立时间

在解除 IDLE2 模式后确保 flash 存储器的建立时间，这是因为在设置 IDLE2 模式后，除了主时钟振荡之外的模块操作停止。

(1) 通过不可屏蔽中断请求信号或未屏蔽的可屏蔽中断请求信号解除 IDLE2 模式

通过设置 OSTS 寄存器确保指定建立时间。

当产生释放源时，依据 OSTS 寄存器的设置专用内部定时器开始计数。 当其溢出时，恢复正常操作模式。



(2) 由复位解除（ $\overline{\text{RESET}}$ 引脚输入，WDT2RES 产生）

此操作与正常复位相同。

振荡稳定时间是 OSTS 寄存器的初始值， $2^{16}/f_x$ 。

25.6 STOP 模式

25.6.1 设置和操作状态

在正常操作模式下，通过将 PSMR.PSM1 和 PSMR.PSM0 位设为 01 或 11 并将 PSC.STP 位设为 1 来设置 STOP 模式。

在 STOP 模式下，副时钟振荡器继续工作但主时钟振荡器停止。停止对 CPU 和片上外围功能的时钟供给。

因此，程序停止执行，且内部 RAM 的内容保持与设置为 STOP 模式前的内容一致。使用副时钟振荡器振荡产生的时钟或外部时钟来运行的片上外围功能继续工作。

表 25-9 显示了 STOP 模式下的操作状态。

因为 STOP 模式使主时钟振荡器的运行停止，所以可将功耗减少至低于 IDLE2 模式时的功耗级别。若不使用副时钟振荡器，内部振荡器和外部时钟，功耗可最小化至仅有漏电流。

注意事项 1. 应在 PSC 寄存器存储数据指令后插入五个或更多 NOP 指令来设置为 STOP 模式。

2. 如果有一个未屏蔽的中断请求信号被保持等待响应时设置 STOP 模式，则 STOP 模式立即被此等待响应的中断请求解除。

25.6.2 解除STOP 模式

在 STOP 模式下来自可操作的外设功能的不可屏蔽中断请求信号（NMI 引脚输入，INTWDT2 信号）、未屏蔽外部中断请求信号（INTP00 至 INTP18 引脚输入）、未屏蔽的内部中断请求信号，或复位信号（由 RESET 引脚输入复位、WDT2RES 信号，或低电压检测器（LVI）），可解除 STOP 模式。

在 STOP 模式解除后，正常操作模式在振荡稳定时间得到保证后恢复。

(1) 通过不可屏蔽中断请求信号或未屏蔽的可屏蔽中断请求信号解除 STOP 模式

通过不可屏蔽中断请求信号或未屏蔽的可屏蔽中断请求信号，可解除 STOP 模式，且与中断请求信号的优先级无关。但是，若 STOP 模式被设置在一个中断服务程序中，则之后发出的中断请求信号按如下所述来执行：

- (a) 若发出的中断请求信号优先级低于当前正在接受服务的中断请求，则仅 STOP 模式解除，并且那个中断请求信号未被响应。该中断请求信号被保持。
- (b) 若发出的中断请求信号优先级高于当前正在接受服务的中断请求（包括不可屏蔽中断请求信号），则 STOP 模式解除，且那个中断请求信号被响应。

注意事项 通过设置 PSC.NMI1M、PSC.NMI0M、PSC.INTM 位为 1 而禁止的中断请求变为无效，因此 STOP 模式不会解除。

表 25-8. 通过中断请求信号解除 STOP 模式的操作

解除源	中断允许 (EI) 状态	中断禁止 (DI) 状态
不可屏蔽中断请求信号	确定振荡稳定时间过后，执行对处理地址的跳转。	
可屏蔽中断请求信号	在保证振荡稳定时间之后执行对处理地址的跳转或执行下一条指令。	在保证振荡稳定时间之后执行下一条指令。

(2) 通过复位解除 STOP 模式

执行与正常的复位操作相同的操作。

表 25-9. STOP 模式下的操作状态

STOP 模式的设置 项目		操作状态	
		不使用副时钟时	使用副时钟时
主时钟振荡器 (f _x)		停止振荡	
副时钟振荡器 (f _{XT})		—	振荡允许
内部振荡器 (f _R)		振荡允许	
PLL		停止运行	
CPU		停止运行	
DMA 控制器		停止运行	
中断控制器		停止运行	
定时器	TAA0 至 TAA5	停止运行	
	TAB0, TAB1	停止运行	
	TMM0 至 TMM3	当 f _R /8 选择为计数时钟时可操作	当 f _R /8 或 f _{XT} 选择为计数时钟时可操作
	TMT0	停止运行	
实时计数器 (RTC)		停止运行	当 f _{XT} 选择为计数时钟时可操作
看门狗定时器 (WDT2)		当 f _R 选择为计数时钟时可操作	当 f _R 或 f _{XT} 选择为计数时钟时可操作
串行接口	CSIF0 至 CSIF4	SCKFn 输入时钟被选为计数时钟时可运行 (n = 0 至 4)	
	I ² C00 至 I ² C02	停止运行	
	UARTC0 至 UARTC4	停止运行 (但是当选择 ASCKC0 输入时钟时, UARTC0 可运行)	
A/D 转换器		停止运行 (未定义转换结果) ^{注 1, 2}	
D/A 转换器		停止运行 ^{注 3, 4} (输出高阻抗)	
实时输出功能 (RTO)		停止运行 (保持输出)	
按键中断功能 (KR)		可运行	
CRC 算法电路		停止运行	
外部总线接口		参见 第五章 总线控制功能	
端口功能		保持设置为 STOP 模式前的状态	
内部数据		CPU 寄存器、状态、数据及其它内部数据, 例如内部 RAM 的内容, 被保持为设置 STOP 模式之前的状态。	
CAN ^{注 5}		停止运行	
USB 功能		停止运行	

- 注
1. A/D 转换器进行操作时如果设置 STOP 模式, 则 A/D 转换器自动停止并且 STOP 模式释放后再次开始操作。但是这种情况下, STOP 模式释放后的 A/D 转换结果无效。STOP 模式设置前的所有 A/D 转换值都无效。
 2. 即使在 A/D 转换器操作时设置 STOP 模式, 功耗减少到设置 STOP 模式前停止 A/D 转换器时的功耗。
 3. D/A 转换器工作时, 如果设置 STOP 模式, 那么 D/A 转换器自动停止并且引脚状态变为高阻。STOP 模式解除后, D/A 转换继续工作, 设置时间过后, 状态返回到设置 STOP 模式之前的输出电平。
 4. 即使在 D/A 转换器操作时设置 STOP 模式, 功耗减少到设置 STOP 模式前停止 D/A 转换器时的功耗。
 5. 仅限于 μ PD70F3770, 70F3771。

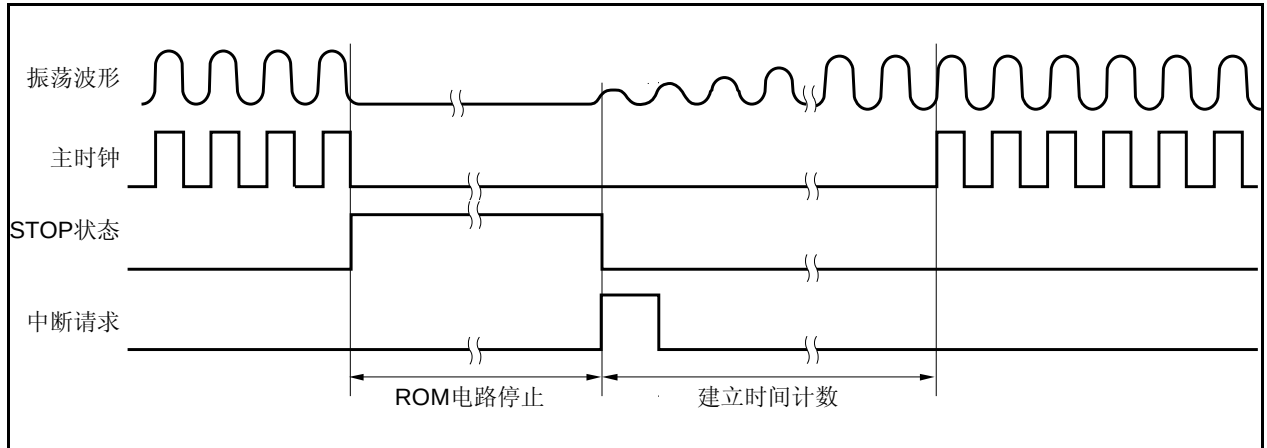
25.6.3 在解除STOP模式时确保振荡稳定时间

在解除 STOP 模式后确保主时钟振荡器的振荡稳定时间，因为在设置 STOP 模式后主时钟振荡器的操作停止。

(1) 通过不可屏蔽中断请求信号或未屏蔽的可屏蔽中断请求信号解除 STOP 模式

通过设置 OSTS 寄存器确保振荡稳定时间。

当产生释放源时，依据 OSTS 寄存器的设置专用内部定时器开始计数。 当其溢出时，恢复正常操作模式。



(2) 通过复位解除

此操作与正常复位相同。

振荡稳定时间是 OSTS 寄存器的初始值， $2^{16}/f_x$ 。

25.7 副时钟工作模式

25.7.1 设置和操作状态

通过在正常操作模式下把 PCC.CK3 位设置为 1，可设置为副时钟工作模式。

当设置为副时钟工作模式时，内部系统时钟从主时钟变为副时钟。通过使用 PCC.CLS 位来检查时钟是否已经被转换。

当 PCC.MCK 位设置为 1 时，主时钟振荡器的运行停止。由此，系统仅工作在副时钟下。

在副时钟工作模式下，功耗可减少至低于正常操作模式时的功耗级别，因为副时钟用作内部系统时钟。此外，通过停止主时钟振荡器的运行，功耗可进一步减少至 STOP 模式时的功耗级别。

表 25-10 显示了副时钟工作模式的操作状态。

注意事项 1. 在对 CK3 位进行操作时，不要改变 PCC.CK2 至 PCC.CK0 位的设置值（建议用位操作指令对此位进行操作）。有关 PCC 寄存器的详情，参见 6.3（1）处理器时钟控制寄存器（PCC）。

2. 若不满足下列条件，应改变 CK2 至 CK0 位使条件满足并设置为副时钟工作模式。

内部系统时钟（f_{CLK}）> 副时钟（f_{XT} = 32.768 kHz）×4

备注 内部系统时钟（f_{CLK}）：依据 CK2 至 CK0 位的设置，由主时钟（f_{xx}）产生时钟。

25.7.2 解除副时钟工作模式

当 CK3 位被设置为 0 时，通过复位信号来解除副时钟操作模式（通过 $\overline{\text{RESET}}$ 引脚输入、WDT2RES 信号、低电压检测器（LVI）或时钟监视器（CLM）复位）。

若主时钟停止（MCK 位= 1），应把 MCK 位设置为 1，用软件保证主时钟的振荡稳定时间，并将 CK3 位设为 0。

当副时钟工作模式解除时正常操作模式恢复。

注意事项 在对 CK3 位进行操作时，不要改变 CK2 至 CK0 位的设置值（建议用位操作指令对此位进行操作）。

有关 PCC 寄存器的详情，参见 6.3（1）处理器时钟控制寄存器（PCC）。

表 25-10. 副时钟工作模式的操作状态

副时钟工作模式的设置 项目		操作状态	
		主时钟振荡时	主时钟停止时
副时钟振荡器 (f _{XT})		振荡允许	
内部振荡器 (f _R)		振荡允许	
PLL		可运行	停止运行 ^{注1}
CPU		可运行	
DMA 控制器		可运行	
中断控制器		可运行	
定时器	TAA0 至 TAA5	可运行	停止运行
	TAB0, TAB1	可运行	停止运行
	TMM0 至 TMM3	可运行	当 f _R /8 或 f _{XT} 选择为计数时钟时可操作
	TMT0		
实时计数器 (RTC)		可运行	当 f _{XT} 选择为计数时钟时可操作
看门狗定时器 (WDT2)		可运行	当 f _R 或 f _{XT} 选择为计数时钟时可操作
串行接口	CSIF0 至 CSIF4	可运行	SCKFn 输入时钟被选为计数时钟时可运行 (n = 0 至 4)
	I ² C00 至 I ² C02	可运行	停止运行
	UARTC0 至 UARTC4	可运行	停止运行 (但是当选择 ASCKC0 输入时钟时, UARTC0 可运行)
A/D 转换器		可运行	停止运行
D/A 转换器		可运行	
实时输出功能 (RTO)		可运行	停止运行 (保持输出)
按键中断功能 (KR)		可运行	
CRC 算法电路		可运行	
外部总线接口		参见 第五章 总线控制功能	
端口功能		可设置	
内部数据		可设置	
CAN ^{注2}		停止运行	
USB 功能		停止运行	

注 1, 在停止主时钟前, 一定要停止锁相环 (PLLCTL.PLLON 位 = 0)。

2. 仅限于 μ PD70F3770, 70F3771

注意事项 当 CPU 在副时钟上操作而且主时钟振荡停止时, 在发生等待时访问寄存器是禁止的。 如果等待已经发生, 只能通过复位解除 (参见 3.4.9 (2))。

<R>

<R>

25.8 Sub-IDLE模式

25.8.1 设置和操作状态

在副时钟操作模式下，通过将 PSMR.PSM1 和 PSMR.PSM0 位设为 00 或 10 并将 PSC.STP 位设为 1 来设置 sub-IDLE 模式。

该模式下，时钟振荡器继续工作但对 CPU，flash 存储器及其它片上外围功能的时钟供给停止。

因此，程序停止执行，且内部 RAM 的内容保持与设置为 Sub-IDLE 模式前的内容一致。CPU 及其它片上外围功能停止。而可以在副时钟或外部时钟下工作的片上外围功能继续工作。

因为 Sub-IDLE 模式停止了 CPU，flash 存储器及其它片上外围功能的运行，所以比副时钟工作模式更能减少功耗。若在主时钟停止后设置为 Sub-IDLE 模式，则该模式下电流消耗可减少至与 STOP 模式同样低的水平。

表 25-12 显示了 Sub-IDLE 模式的操作状态。

- 注意事项**
1. 应在设置 Sub-IDLE 模式的指令存储到 PSC 寄存器之后，插入五个或更多 NOP 指令。
 2. 如果有一个未屏蔽的中断请求信号被保持等待响应时设置 sub-IDLE 模式，则 sub-IDLE 模式立即被此等待响应的中断请求解除。

25.8.2 解除sub-IDLE模式

在 sub-IDLE 模式下来自可操作的外设功能的不可屏蔽中断请求信号（NMI 引脚输入，INTWDT2 信号）、未屏蔽外部中断请求信号（INTP0 至 INTP18 引脚输入）、未屏蔽的内部中断请求信号，或复位信号（由 $\overline{\text{RESET}}$ 引脚输入复位、WDT2RES 信号，低电压检测器（LVI），或时钟监视器），可解除 sub-IDLE 模式。PLL 返回到设置为 sub-IDLE 模式前的操作状态。

当通过中断请求信号解除 sub-IDLE 模式时，设置为副时钟工作模式。

(1) 通过不可屏蔽中断请求信号或未屏蔽的可屏蔽中断请求信号解除 Sub-IDLE 模式

通过不可屏蔽中断请求信号或未屏蔽的可屏蔽中断请求信号，可解除 sub-IDLE 模式，且与中断请求信号的优先级无关。

但是，若 sub-IDLE 模式设置在一个中断服务程序中，则之后发出的中断请求信号接受如下处理：

- (a) 若发出的中断请求信号优先级低于当前正在接受服务的中断请求，则仅 sub-IDLE 模式解除，并且那个中断请求信号未被响应。该中断请求信号被保持。
- (b) 若发出的中断请求信号优先级高于当前正在接受服务的中断请求（包括不可屏蔽中断请求信号），则 Sub-IDLE 模式解除，且那个中断请求信号被响应。

- 注意事项**
1. 通过设置 PSC.NMI1M、PSC.NMI0M、PSC.INTM 位为 1 而禁止的中断请求信号变为无效，因此 sub-IDLE 模式不会解除。
 2. sub-IDLE 模式解除时，从解除 sub-IDLE 模式的中断请求信号产生至模式解除需经过 12 个副时钟周期（大约 366 μs ）。

表 25-11. 通过中断请求信号解除 Sub-IDLE 模式后的操作

解除源	中断允许 (EI) 状态	中断禁止 (DI) 状态
不可屏蔽中断请求信号	执行对处理地址的跳转。	
可屏蔽中断请求信号	执行对处理地址的跳转或执行下一条指令。	执行下一条指令。

(2) 通过复位解除 Sub-IDLE 模式

执行与正常的复位操作相同的操作。

表 25-12. Sub-IDLE 模式的操作状态

Sub-IDLE 模式的设置 项目		操作状态	
		主时钟振荡时	主时钟停止时
副时钟振荡器 (f _{XT})		振荡允许	
内部振荡器 (f _R)		振荡允许	
PLL		可运行	停止运行 ^{‡1}
CPU		停止运行	
DMA 控制器		停止运行	
中断控制器		停止运行	
定时器	TAA0 至 TAA5	停止运行	
	TAB0, TAB1	停止运行	
	TMM0 至 TMM3	当 f _R /8 或 f _{XT} 选择为计数时钟时可操作	
	TMT0	停止运行	
实时计数器 (RTC)		可运行	当 f _{XT} 选择为计数时钟时可操作
看门狗定时器 (WDT2)		当 f _R 或 f _{XT} 选择为计数时钟时可操作	
串行接口	CSIF0 至 CSIF4	SCKFn 输入时钟被选为计数时钟时可运行 (n = 0 至 4)	
	I ² C00 至 I ² C02	停止运行	
	UARTC0 至 UARTC4	停止运行 (但是当选择 ASCKC0 输入时钟时, UARTC0 可运行)	
A/D 转换器		保持运行 (保持转换结果) ^{‡2}	
D/A 转换器		保持运行 (保持输出) ^{‡2}	
实时输出功能 (RTO)		停止运行 (保持输出)	
按键中断功能 (KR)		可运行	
CRC 算法电路		停止运行	
外部总线接口		参见 第五章 总线控制功能 (操作状态与 IDLE 模式相同)	
端口功能		保持设置为 Sub-IDLE 模式前的状态	
内部数据		CPU 寄存器、状态、数据及其它内部数据 (如内部 RAM 的内容) 与设置为 Sub-IDLE 模式前相比保持原样。	
CAN ^{‡3}		停止运行	
USB 功能		停止运行	

- 注
1. 在停止主时钟前, 一定要停止锁相环 (PLLCTL.PLLON 位 = 0)。
 2. 为了实现降低功率损耗, 在切换到 sub-IDLE 模式前, 停止 A/D 转换器和 D/A 转换器。
 3. 仅限于 μ PD70F3770, 70F3771。

26.1 概述

有下列复位功能：

(1) 四种复位源：

- 通过 $\overline{\text{RESET}}$ 引脚输入外部复位
- 通过看门狗定时器 2 (WDT2) 溢出复位 (WDT2RES)
- 通过比较低压检测器 (LVI) 供应电压与检测到的电压进行系统复位
- 通过检测时钟监视器 (CLM) 振荡停止进行系统复位

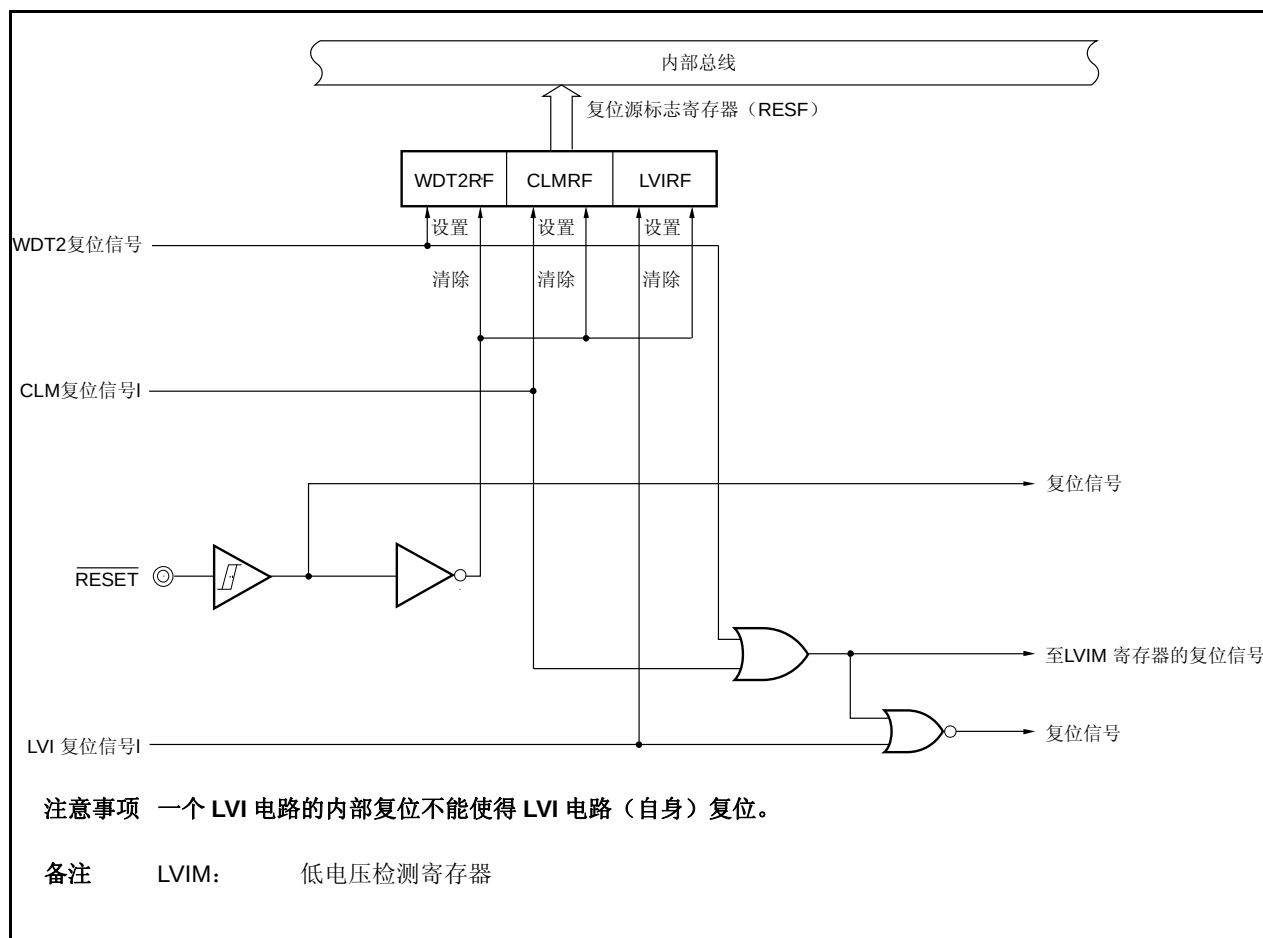
复位解除后，复位源可由复位源标志检测器 (RESF) 确定。

(2) 紧急工作模式

复位之后在插入的主时钟振荡稳定时间内如果发生 WDT2 的溢出，将认定为一个主时钟振荡异常，并且 CPU 开始工作于内部振荡时钟。

注意事项 在紧急工作模式下，除了那些能够在内部时钟振荡工作的寄存器，诸如中断功能、端口功能、“WDT2 或定时器 M”之类以外，不要访问片上其它外设 I/O 寄存器。此外，此模式下禁止使用外部输入时钟对 CSIF0 至 CSIF4 以及 UARTC0 进行操作。

图 26-1. 复位功能框图



26.2 检测复位源的寄存器

V850ES/JG3-H 和 V850ES/JH3-H 具有 4 种复位源。 复位解除后，发生的复位源可由复位源标志检测器（RESF）进行检查。

(1) 复位源标志寄存器（RESF）

RESF 寄存器是一个特殊的寄存器，只能够通过特殊的顺序来写入（参见 3.4.8 特殊寄存器）。

RESF 寄存器表示复位信号由哪个复位源产生。

本寄存器可按 8 位或 1 位读写。

RESET 输入引脚将该寄存器设为 00H。 如果复位源不同于 RESET 引脚信号，初始值会有所不同。

复位后：00H^注 R/W 地址：FFFFF888H

	7	6	5	4	3	2	1	0
RESF	0	0	0	WDT2RF	0	0	CLMRF	LVIRF

WDT2RF	WDT2的复位信号
0	不产生
1	产生

CLMRF	CLM的复位信号
0	不产生
1	产生

LVIRF	LVI的复位信号
0	不产生
1	产生

注 当通过 RESET 引脚执行复位时，RESF 寄存器的值被设为 00H。 当通过看门狗定时器 2（WDT2）、低电压检测器（LVI）、或时钟监视器（CLM）执行复位时，该寄存器的复位标志（WDT2RF 位，CLMRF 位和 LVIRF 位）被置位。 而其它复位源保持不变。

注意事项 只有“0”能写入该寄存器的各位。 如果写“0”与标志的设置冲突（发生复位），标志的设置会优先。

26.3 操作

26.3.1 通过 $\overline{\text{RESET}}$ 引脚的复位操作

当 $\overline{\text{RESET}}$ 引脚输入一个低电平时，系统复位，且所有硬件将被初始化。

当 $\overline{\text{RESET}}$ 引脚的电平由低变高时，复位状态将被解除。

表 26-1. $\overline{\text{RESET}}$ 引脚输入时的硬件状态

项目	复位期间	复位后
主时钟振荡器 (f_x)	振荡停止	振荡开始
副时钟振荡器 (f_{XT})	振荡继续	
内部振荡器	振荡停止	振荡开始
外围时钟 (f_x 至 $f_x/1, 024$)	停止运行	在振荡稳定时间得到保证后开始运行
内部系统时钟 (f_{CLK}) , CPU 时钟 (f_{CPU})	停止运行	在振荡稳定时间得到保证后开始运行 (初始化为 $f_x/8$)
CPU	初始化	在振荡稳定时间得到保证后开始运行
看门狗定时器 2	操作停止 (初始化为 0)	将内部振荡时钟作为时钟源从 0 向上计数
内部 RAM	若加电复位或 CPU 访问与复位输入冲突 (数据被损坏) 则不确定。 否则, 则保持复位输入后一时刻的值。	
I/O 线路 (端口/复用功能引脚)	高阻 [※]	
片上外围 I/O 寄存器	初始化为指定状态, OCDM 寄存器被设置为 01H。	
其它片上外围功能	停止运行	在振荡稳定时间得到保证后开始运行

注 电源开启时在复位期间以下引脚可能暂时输出不确定电平。

- P10/ANO0 引脚
- P11/ANO1 引脚
- DDO 引脚 (仅限于 V850ES/JH3-H)
- P53/SIF2/TIAB00/KR3/TOAB00/RTP03/DDO 引脚 (仅限于 V850ES/JG3-H)

注意事项 OCDM 寄存器通过 $\overline{\text{RESET}}$ 引脚输入进行初始化。 所以, 必须予以注意的是: 复位解除后在 OCDM.OCDM0 位被清除之前, 如果 P56/INTP05/ $\overline{\text{DRST}}$ 引脚输入高电平, 则进入片上调试模式。 关于细节, 请参见 第四章 端口功能。

图 26-2. RESET 引脚输入引起的复位操作的时序

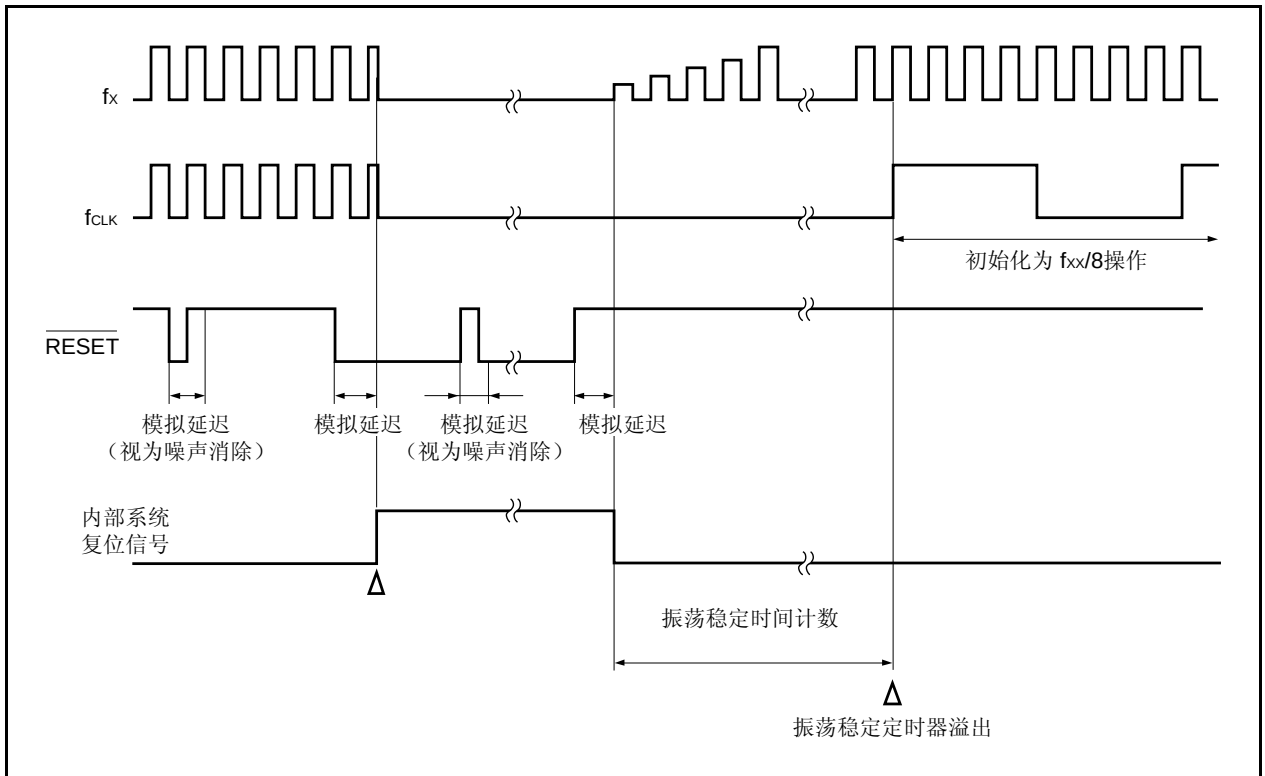
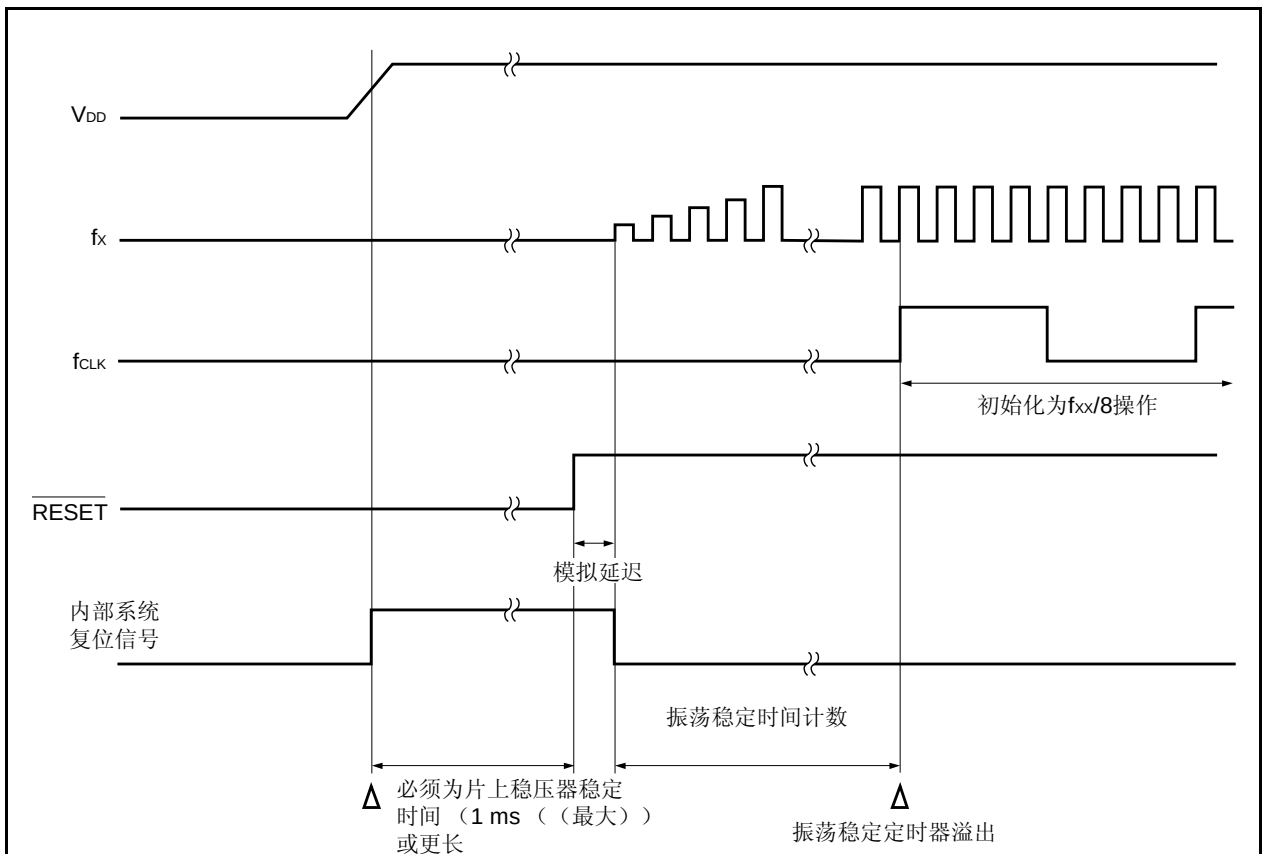


图 26-3. 上电复位操作时序



26.3.2 通过看门狗定时器 2 的复位操作

当看门狗定时器 2 由于溢出被设置为复位操作模式时，基于看门狗定时器溢出（产生 WDT2RES 信号），将执行系统复位且硬件被初始化。

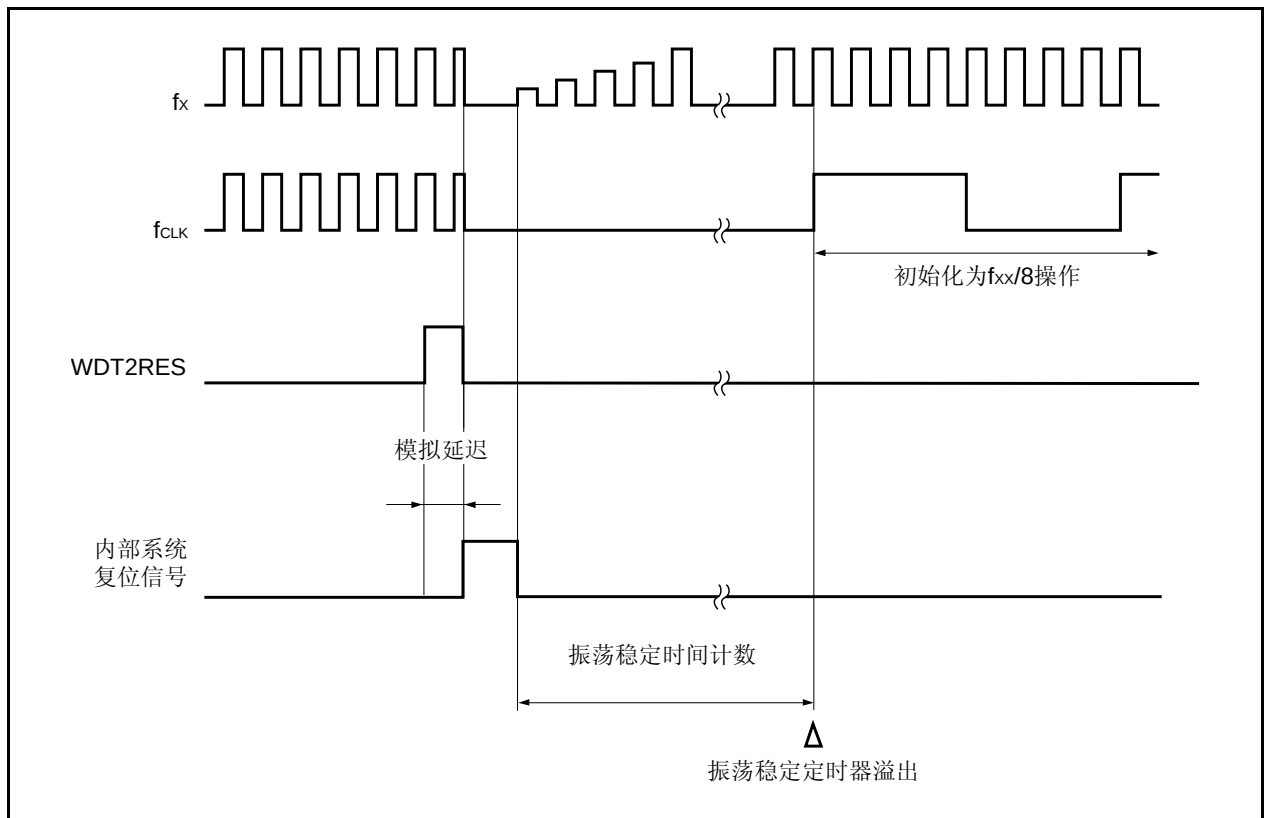
在看门狗定时器 2 溢出后，会进入复位状态并且持续到预定时间（模拟延迟），然后复位状态将会自动解除。

复位期间主时钟振荡器停止。

表 26-2. 看门狗定时器 2 复位操作时的硬件状态

项目	复位期间	复位后
主时钟振荡器（fx）	振荡停止	振荡开始
副时钟振荡器（fxT）	振荡继续	
内部振荡器	振荡停止	振荡开始
外设时钟（fxx 至 fxx/1, 024）	停止运行	在振荡稳定时间得到保证后开始运行
内部系统时钟（fxx）， CPU 时钟（fCPU）	停止运行	在振荡稳定时间得到保证后开始运行（初始化为 fxx/8）
CPU	初始化	在振荡稳定时间得到保证后执行程序
看门狗定时器 2	操作停止（初始化为 0）	将内部振荡时钟作为时钟源从 0 向上计数
内部 RAM	若加电复位或 CPU 访问与复位输入冲突（数据被损坏）则不确定。 否则，则保持复位输入后一时刻的值。	
I/O 线路（端口/复用功能引脚）	高阻抗状态	
片上外围 I/O 寄存器	初始化为指定状态，OCDM 寄存器保持原值	
其它片上外围功能	停止运行	在振荡稳定时间得到保证后开始运行

图 26-4. WDT2RES 信号生成所引起的复位操作时序



26.3.3 通过低电压检测器进行复位操作

如果在允许 LVI 操作时供应电压低于通过低电压检测器检测到的电压，那么将会对系统进行复位（LVIM.LVIMD 位被设为 1 时），且硬件将会被初始化。

复位状态会从检测到供应电压降低开始一直持续到供应电压高于 LVI 检测电压为止。

复位期间主时钟振荡器停止。

当 LVIMD 位 = 0 时，如果检测到低电压则产生中断请求信号（INTLVI）。

表 26-3. 由低电压检测器复位操作时的硬件状态

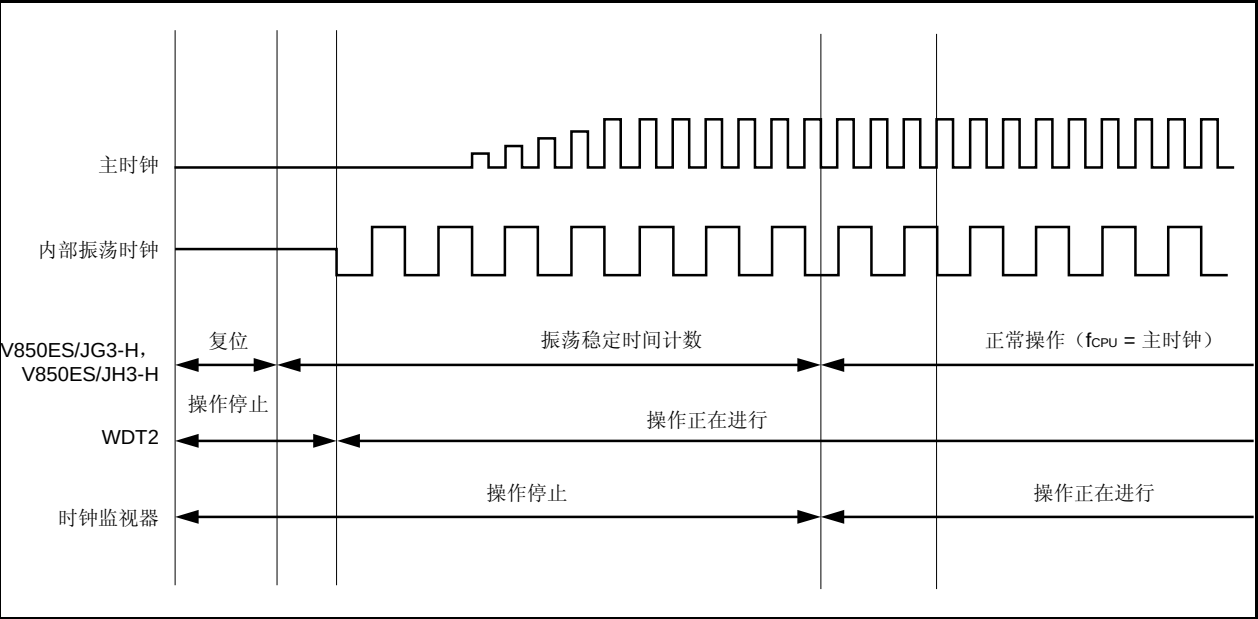
项目	复位期间	复位后
主时钟振荡器（fx）	振荡停止	振荡开始
副时钟振荡器（fxT）	振荡继续	
内部振荡器	振荡停止	振荡开始
外围时钟（fx 至 fx/1, 024）	停止运行	在振荡稳定时间得到保证后开始运行
内部系统时钟（fxx）， CPU 时钟（fcpu）	停止运行	在振荡稳定时间得到保证后开始运行（初始化为 fxx/8）
CPU	初始化	在振荡稳定时间得到保证后开始运行
WDT2	操作停止（初始化为 0）	将内部振荡时钟作为时钟源从 0 向上计数
内部 RAM	若加电复位或 CPU 访问与复位输入冲突（数据被损坏）则不确定。 否则，则保持复位输入后一时刻的值。	
I/O 线路（端口/复用功能引脚）	高阻抗状态	
片上外围 I/O 寄存器	初始化为指定状态，OCDM 寄存器保持原值	
LVI	停止运行	
其它片上外围功能	停止运行	在振荡稳定时间得到保证后开始运行

备注 关于低电压检测器的复位时序，参见**第二十八章 低电压检测器（LVI）**。

26.3.4 复位解除后的操作

复位解除之后，主时钟开始振荡并保证振荡稳定时间（OSTS 寄存器初始值： $2^{16}/f_x$ ），且开始 CPU 程序的执行。
使用内部振荡时钟作为源时钟时，在复位解除之后 WDT2 立刻开始运行。

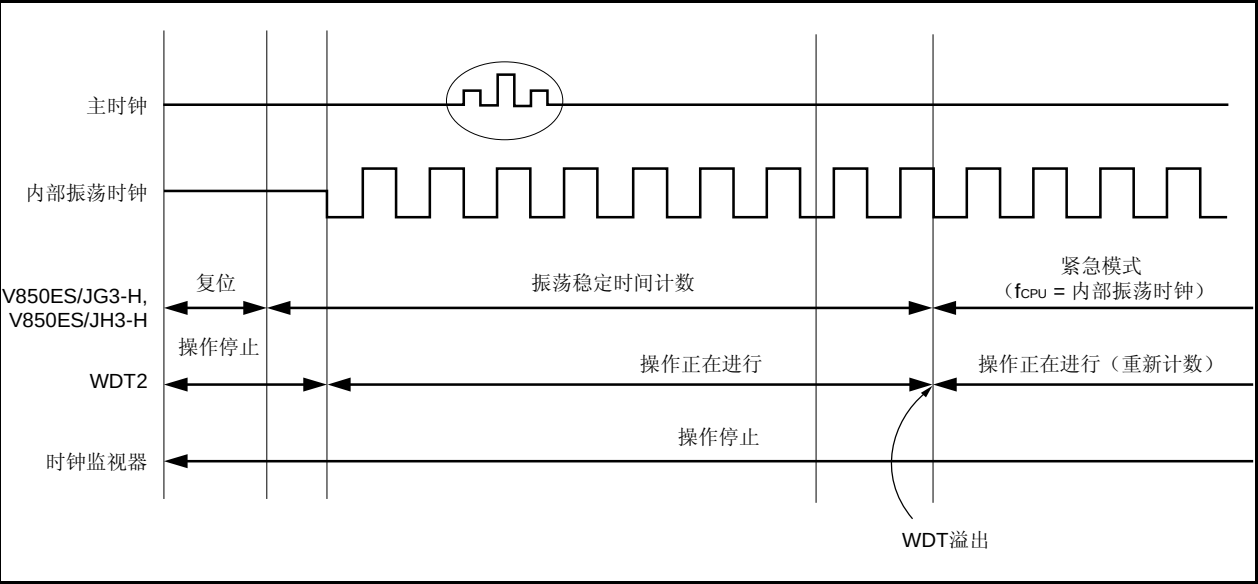
图 26-5. 复位解除后的操作



(1) 紧急操作模式

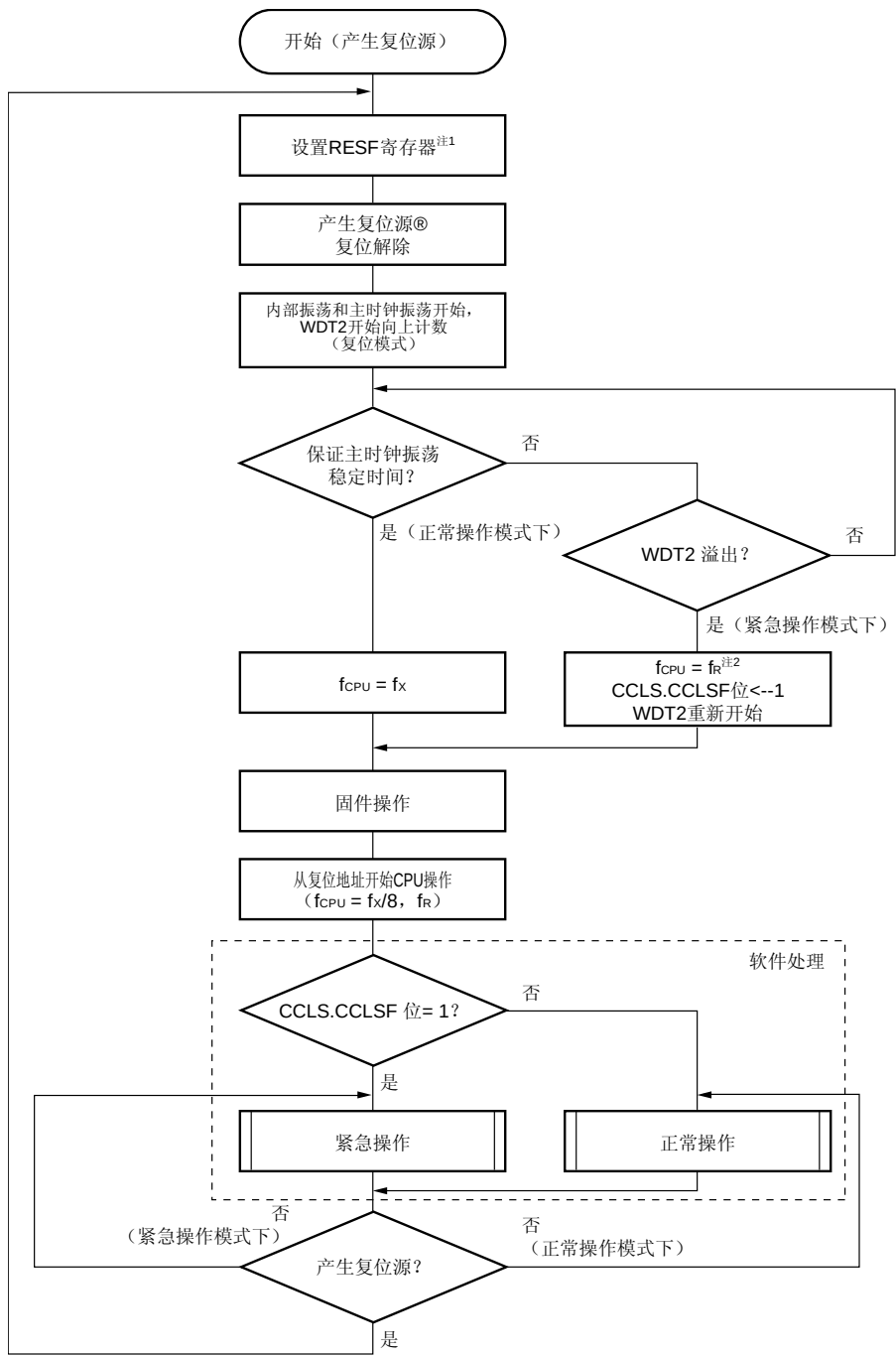
如果在振荡稳定时间保护期之前主时钟发生异常，则 WDT2 在执行 CPU 程序之前溢出。此时，CPU 通过使用内部振荡时钟作为源时钟开始执行程序。

图 26-6. 复位解除后的操作



CPU 操作时钟状态可由 CPU 操作时钟状态寄存器 (CCLS) 来检查。

26.3.5 复位功能操作流程



注 1. 位的设定随复位源不同而改变。

复位源	WDT2RF 位	CRMRF 位	LVIRF 位
RESET 引脚	0	0	0
WDT2	1	保持复位前的值	保持复位前的值
CLM	保持复位前的值	1	保持复位前的值
LVI	保持复位前的值	保持复位前的值	1

2. 不能停止内部振荡器。

27.1 功能

时钟监控器利用内部振荡器时钟采样主时钟，并且在主时钟停止时产生一个复位信号。

一旦通过操作使能标志允许了时钟监控器的操作，就不能通过除了复位之外的其它手段将其清为 0。

当时钟监控器产生的复位发生时，RESF.CLMRF 位置位。 关于 RESF 寄存器的细节，参见 26.2 检查复位源寄存器。

在以下条件下时钟监控器自动停止。

- 解除 STOP 模式之后的振荡稳定时间
- 主时钟停止时（从副时钟操作期间 PCC.MCK 位 = 1 时开始，一直到主时钟操作期间 PCC.CLS 位 = 0 为止）
- 采样时钟（内部振荡时钟）停止时
- 当 CPU 工作于内部振荡时钟时

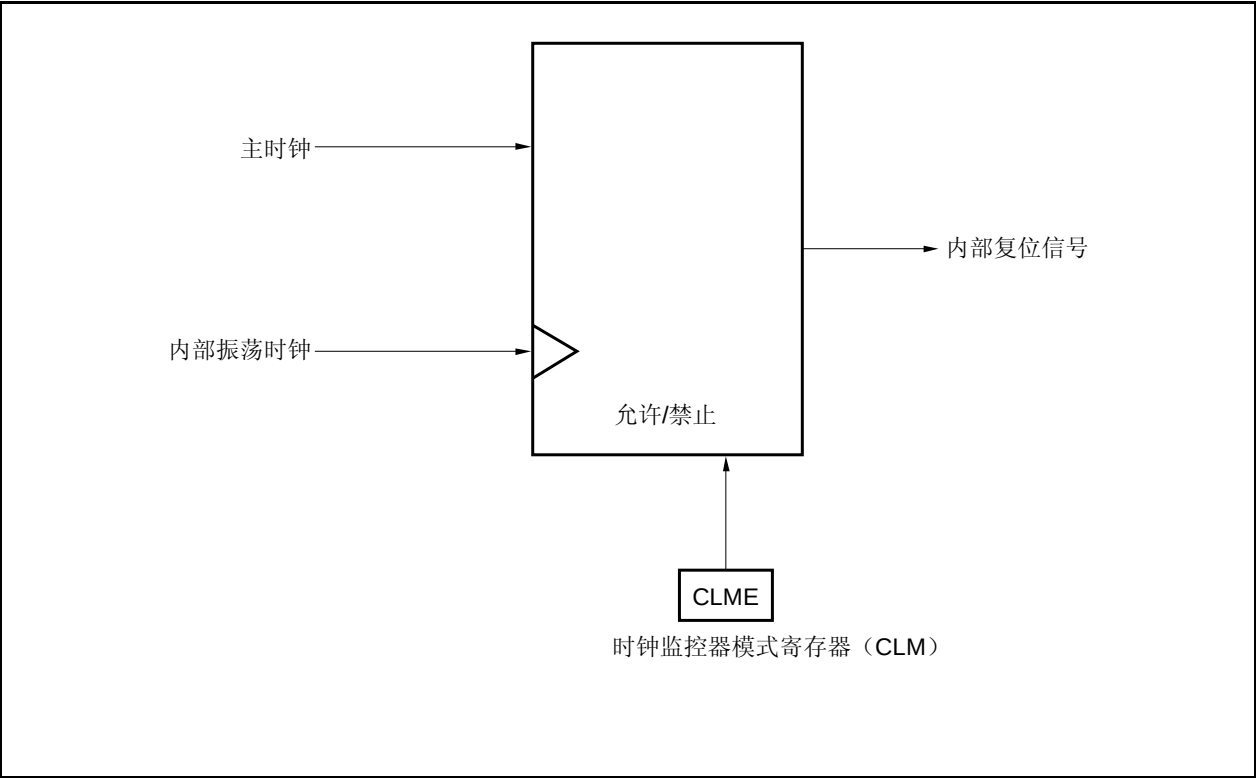
27.2 配置

时钟监控器包含以下硬件：

表 27-1. 时钟监控器的配置

项目	配置
控制寄存器	时钟监视器模式寄存器（CLM）

图 27-1. 通过 RESET 引脚输入复位的时序



27.3 寄存器

时钟监控器由时钟监控器模式寄存器（CLM）控制。

(1) 时钟监视器模式寄存器（CLM）

CLM 寄存器是一个特殊寄存器。 只有通过特定的次序组合才能写入该寄存器（参见 3.4.8 特殊寄存器）。

该寄存器用于设置时钟监控器的操作模式。

该寄存器可按 8 位或 1 位读写。

复位后本寄存器被设置为 00H。

复位后： 00H R/W 地址： FFFFF870H							
	7	6	5	4	3	2	1 <0>
CLM	0	0	0	0	0	0	CLME
CLME		时钟监控器操作使能或禁止					
0		禁止时钟监控器操作					
1		允许时钟监控器操作					

- 注意事项
- 一旦将 CLME 位设为 1，除了复位不能使用任何其它方式将其清 0。
 - 当发生时钟监控器复位时，CLME 位被清为 0 且 RESF.CLMRF 位被设为 1。
 - 务必将 7 至 1 位设置为“0”。

27.4 操作

本节说明了时钟监视器的功能。 开始条件和停止条件如下所示。

<开始条件>

通过将 CLM.CLME 位设为 1 允许操作

<停止条件>

- STOP 模式被释放后对振荡稳定时间进行计数时。
- 主时钟停止时（从副时钟操作期间 PCC.MCK 位 = 1 时开始，一直到主时钟操作期间 PCC.CLS 位 = 0 为止）
- 采样时钟（内部振荡时钟）停止时
- 当 CPU 使用内部振荡时钟工作时

表 27-2. 时钟监控器的操作状态
(内部振荡时钟工作, CLM.CLME 位 = 1 时)

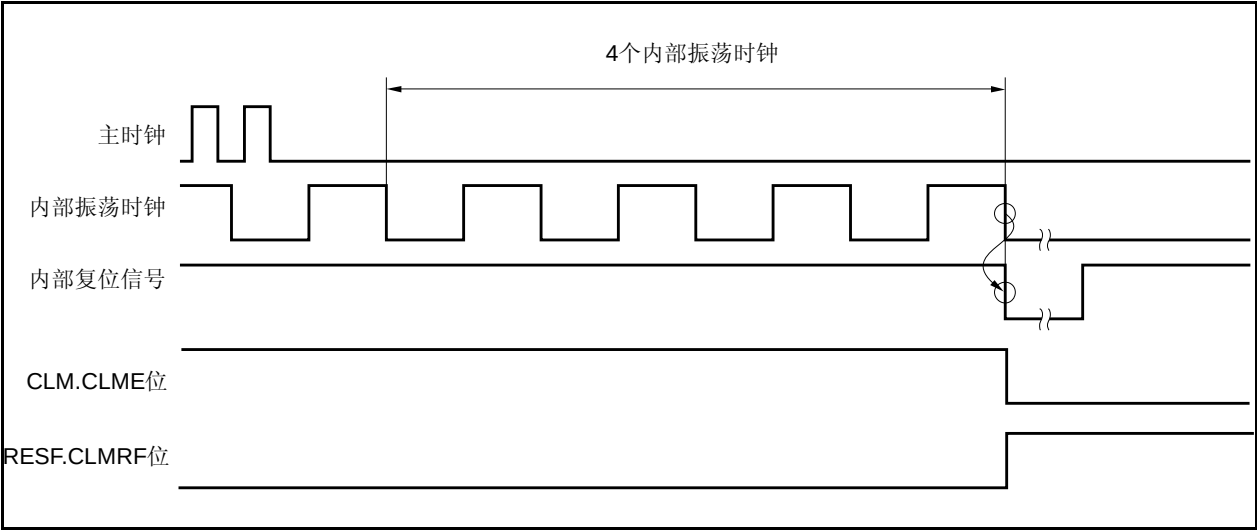
CPU 工作时钟	操作模式	主时钟的状态	内部振荡时钟的状态	时钟监控器的状态
主时钟	HALT 模式	振荡	振荡 ^{注1}	运行 ^{注2}
	IDLE1, IDLE2 模式	振荡	振荡 ^{注1}	运行 ^{注2}
	STOP 模式	停止	振荡 ^{注1}	停止
副时钟 (PCC.MCK = 0)	Sub-IDLE 模式	振荡	振荡 ^{注1}	运行 ^{注2}
副时钟 (PCC.MCK = 1)	Sub-IDLE 模式	停止	振荡 ^{注1}	停止
内部振荡时钟	—	停止	振荡 ^{注3}	停止
复位期间	—	停止	停止	停止

- 注
1. 内部振荡器可通过设置 RCM.RSTOP 位为 1 来停止。
 2. 当内部振荡器停止，时钟监控器同时停止。
 3. 内部振荡器不能通过软件停止。

(1) 主时钟振荡停止时的操作（CLME 位 = 1）

如果 CLME 位=1 时主时钟停止振荡，则如图 27-2 所示产生一个内部复位信号。

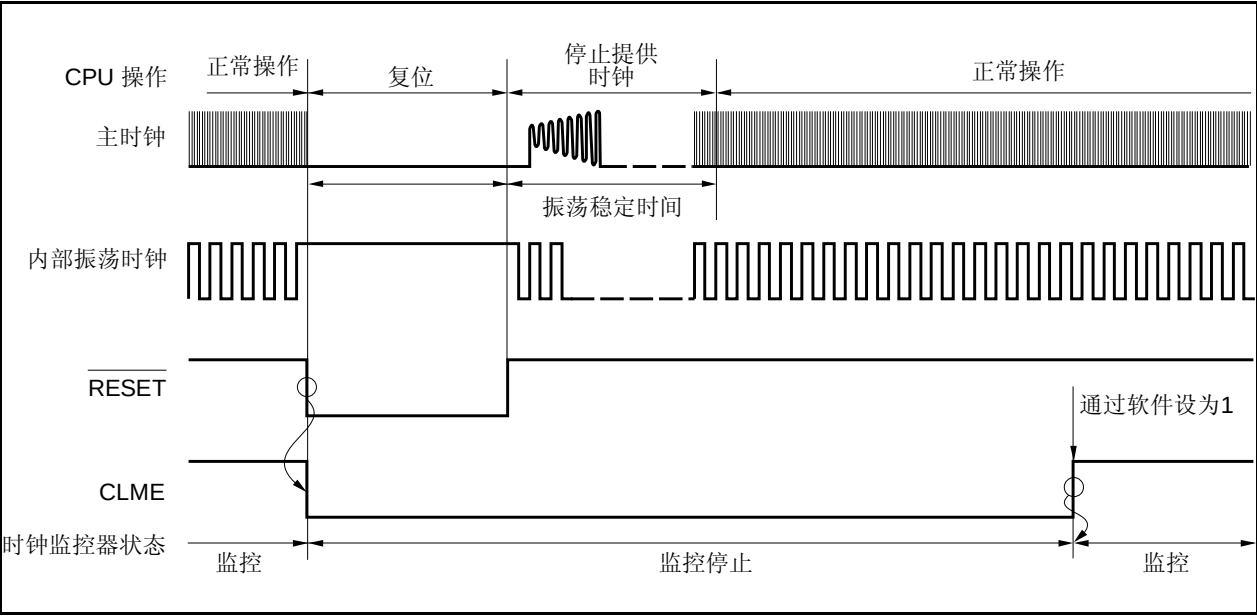
图 27-2. 由主时钟振荡停止引起的复位期间



(2) $\overline{\text{RESET}}$ 输入之后的时钟监控器状态

$\overline{\text{RESET}}$ 输入将 CLM.CLME 位清为 0 并停止时钟监控器的操作。 当在主时钟振荡稳定时间的结束处通过软件将 CLME 位设为 1 时，开始监控。

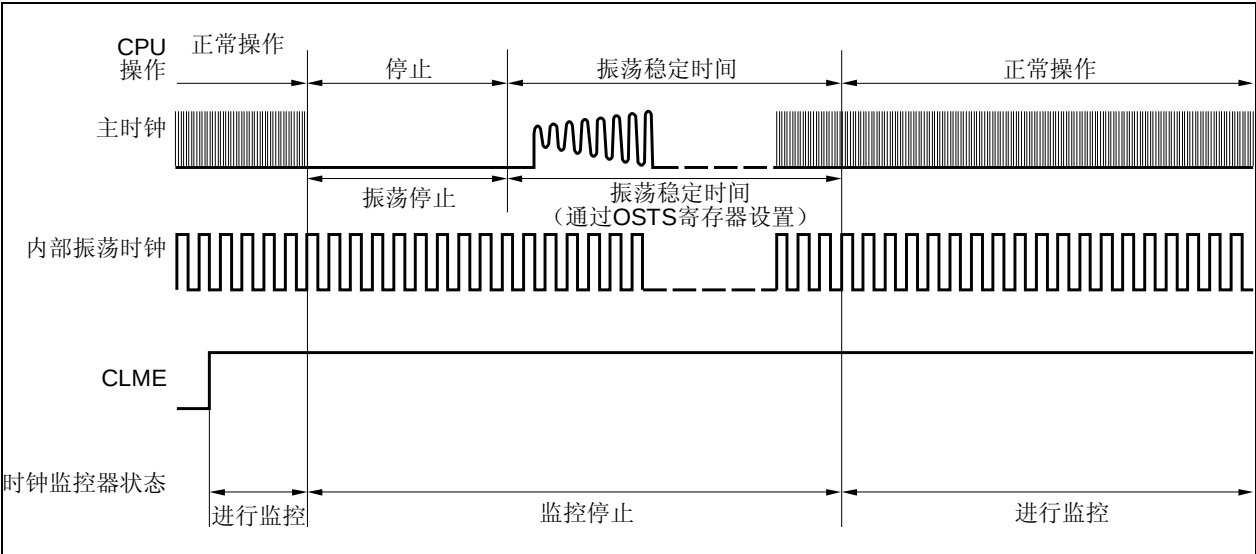
图 27-3. $\overline{\text{RESET}}$ 输入后的时钟监控器状态
($\overline{\text{RESET}}$ 输入后并在主时钟振荡稳定时间完成时设置 CLM.CLME 位 = 1)



(3) 在 STOP 模式时被或 STOP 模式被释放后的操作

如果在 CLM.CLME 位 = 1 时设为 STOP 模式，那么监视器会在 STOP 模式中停止操作且此时会对振荡稳定时间进行计数。 振荡稳定时间过后监视器将会自动开始工作。

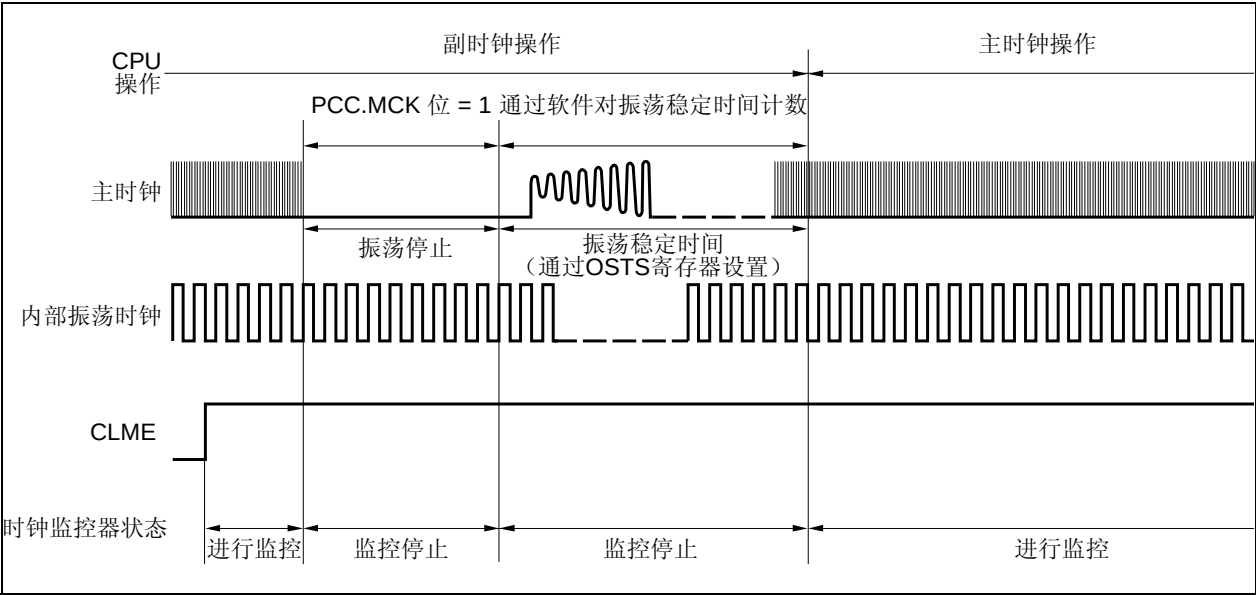
图 27-4. 在 STOP 模式时被或 STOP 模式释放后的操作



(4) 主时钟停止时的操作（任意）

副时钟操作期间（PCC.CLS 位 = 1），或通过设置 PCC.MCK 位为 1 停止主时钟时，监控器停止工作，直到主时钟开始工作（PCC.CLS 位 = 0）为止。 当主时钟开始工作时，时钟监控器随之自动开始工作。

图 27-5. 主时钟停止时的操作（任意）



(5) CPU 基于内部振荡时钟工作时的操作（CCLS.CCLSIF 位 = 1）

CCLSIF 位为 1 时，即使将 CLME 位设为 1 监控器操作也不会停止。

第二十八章 低电压检测器 (LVI)

28.1 功能

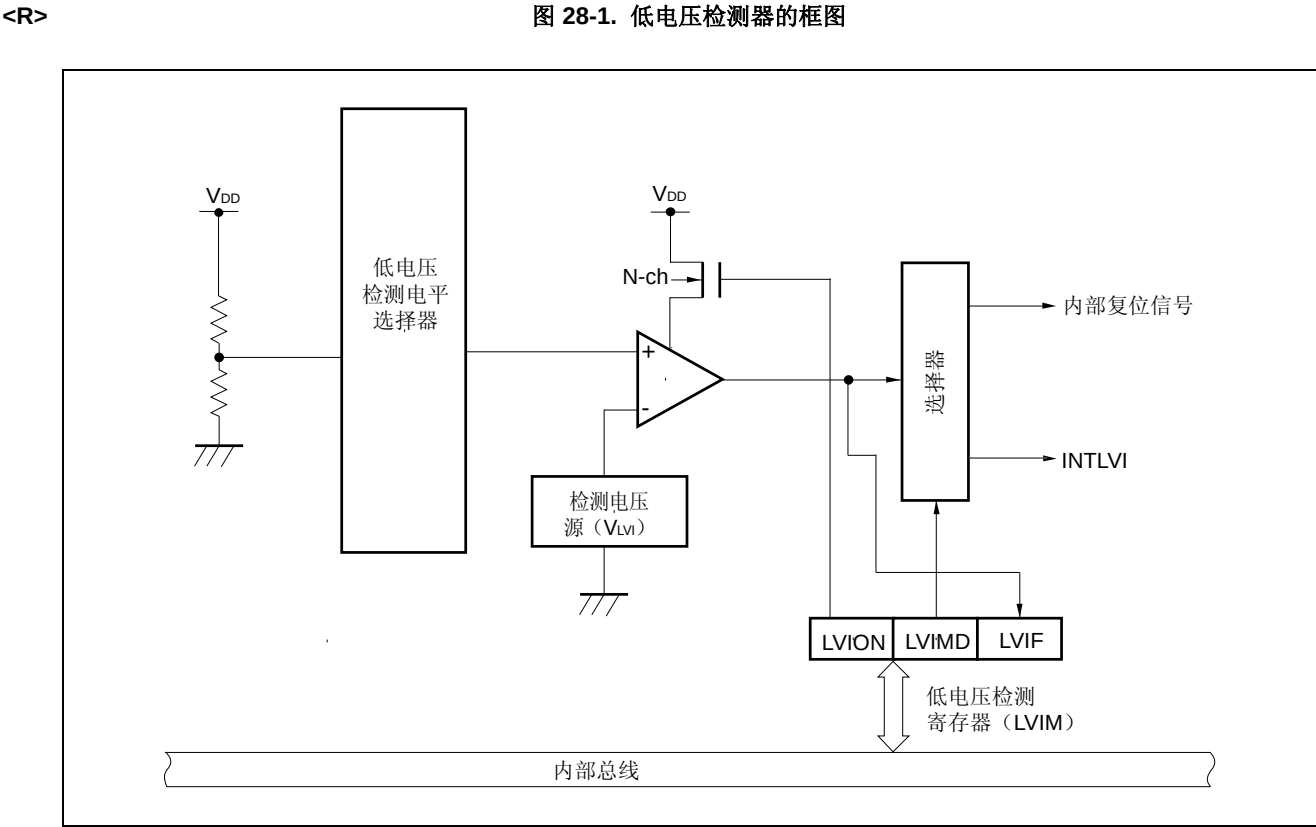
低电压监测器 (LVI) 具有以下功能。

- 如果在选择低电压检测时发生中断，低电压检测器连续地比较供电电压 (V_{DD}) 和检测电压 (V_{LVI})，并且，在供电电压下降或上升越过检测电压时，生成一个内部中断信号。
- 如果在选择低电压检测时发生复位，当供电电压 (V_{DD}) 下降越过检测电压 (V_{LVI}) 时，低电压检测器生成一个中断复位信号。
- 可由软件选择中断或复位信号。
- 可以在 STOP 模式中工作。

如果低电压检测器用于生成复位信号，则当生成复位信号时 RESF.LVIRF 位置为 1。有关 RESF 寄存器的详细内容，请参见 26.2 检测复位源的寄存器。

28.2 配置

低电压检测器的功能框图如下所示。



28.3 寄存器

低电压检测器由下列寄存器控制。

- 低电压检测寄存器 (LVIM)
- 内部 RAM 数据状态寄存器 (RAMS)

(1) 低电压检测寄存器 (LVIM)。

LVIM 寄存器是一个特殊寄存器。只有通过特定的次序组合才能写入该寄存器（参见 3.4.8 特殊寄存器）。
LVIM 寄存器用于允许或禁止低电压检测，并用于设置低电压检测器的操作模式。
该寄存器可按 8 位或 1 位为单元进行读写。但是，LVIF 位为只读位。

复位后: 00H ^{注1}		R/W	地址: FFFFF890H					
<7>		6	5	4	3	2	<1>	<0>
LVIM	LVION	0	0	0	0	0	LVIMD	LVIF
LVION		允许或禁止低电压检测操作						
0		禁止操作。						
1		允许操作。						
LVIMD		选择低电压检测的操作模式						
0		当供电电压下降或上升越过检测电压值时，生成中断信号 INTLVI。						
1		当供电电压下降越过检测电压值时，生成内部复位信号 LVIRE。						
LVIF ^{注2}		低电压检测标志						
0		当供电电压 > 检测电压，或当操作禁止时						
1		接入电源的供电电压 < 检测电压时						

- 注
1. 通过低电压检测复位: 82H
由其它复位源复位: 00H
 2. 在 LVI 开始工作 (LVION 位=1) 或当发生 INTLVI 之后，使用 LVIF 位确认供电电压的状态。

- 注意事项
1. 当 LVION 和 LVIMD 位为 1 时，直到由其它非低电压检测复位源生成复位请求时，低电压检测器才会停止操作。
 2. 当 LVION 位置为 1 时，LVI 电路内的比较器开始工作。LVION 位置位后，在 LVIF 位检测电压前，由软件控制等待至少 0.2ms。
 3. 务必将 6 至 2 位设置为“0”。

(2) 内部 RAM 数据状态寄存器 (RAMS)

RAMS 寄存器是一个特殊寄存器。只有通过特定的次序组合才能写入该寄存器（参见 3.4.8 特殊寄存器）。

该寄存器是一个用于指示内部 RAM 是否有效的标志寄存器。

该寄存器可按 8 位或 1 位为单元进行读写。

RAMF 位的设置/清除条件如下所示。

- 设置条件：检测电压低于指定电平
由指令设置
- 清除条件：以特定顺序写入 0

复位后：01H ^註		R/W		地址：FFFFF892H					
		7	6	5	4	3	2	1	<0>
RAMS		0	0	0	0	0	0	0	RAMF
	RAMF	内部 RAM 电压检测							
	0	不检测低于 RAM 保持电压的电压。							
	1	检测低于 RAM 保持电压的电压。							

注 仅当检测到的电压降到 RAM 保持电压以下时，该寄存器才会进行复位。

28.4 操作

依据 LVIM.VIMD 位的设置，来决定生成中断信号 (INTLVI) 或内部复位信号。
关于如何指定每个操作，连同其时序图，一并描述如下。

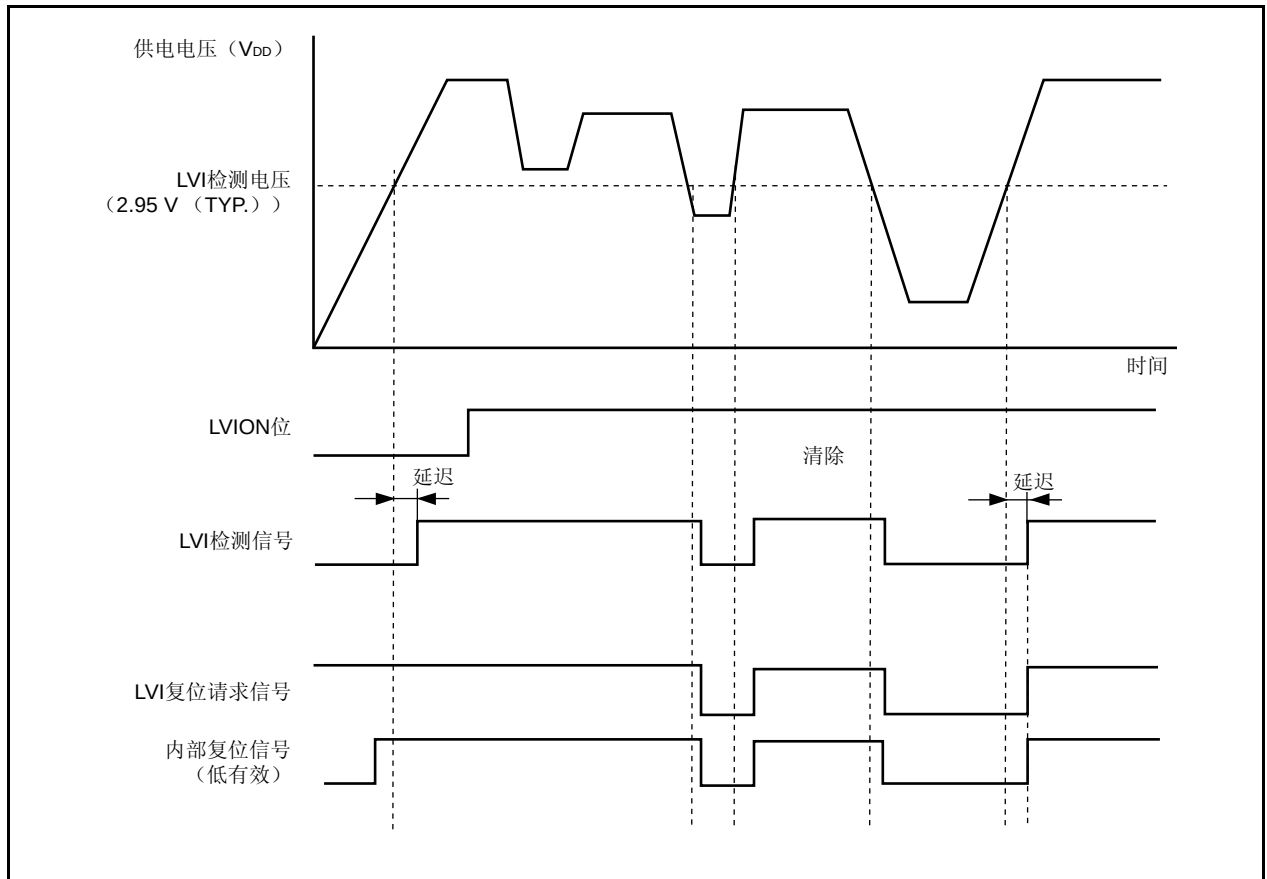
28.4.1 用作内部复位信号

<准备开始操作>

- <1> 屏蔽 LVI 中断。
- <2> 设置 LVIM.LVION 位为 1 (以允许操作)。
- <3> 插入一个 0.2ms (最大值) 的等待时间或通过软件插入更多。
- <4> 通过使用 LVIM.LVIF 位，检验供应电压是否大于检测电压。
- <5> 设置 LVIM.LVIMD 位为 1 (以生成内部复位信号)。

注意事项 如果 LVIMD 位置为 1，那么直到生成 LVI 之外的复位请求时，LVIM 寄存器的内容才能改变。

图 28-2. 低电压检测器的操作时序 (LVIMD 位=1)



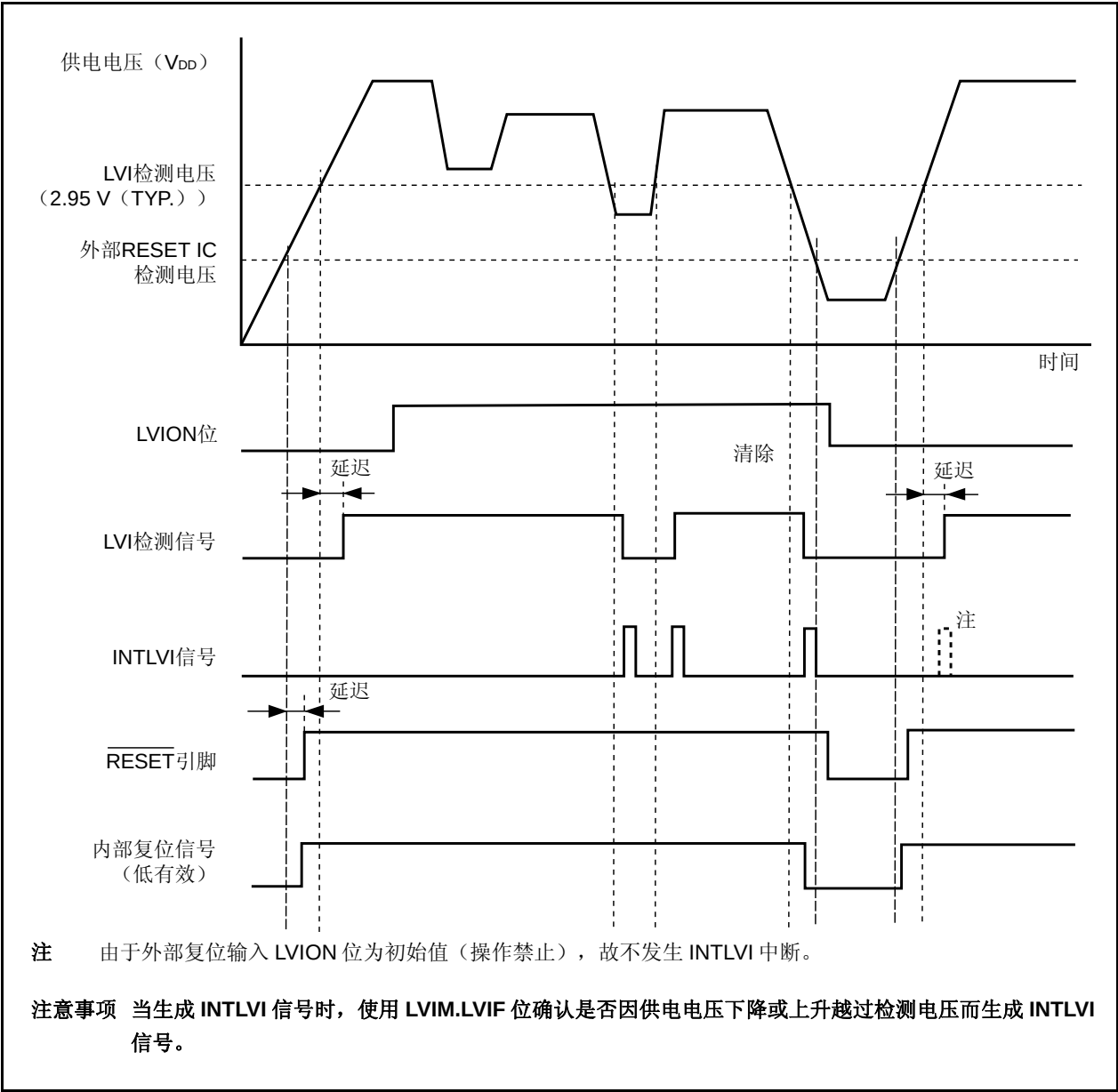
28.4.2 用作中断

- <准备开始操作>
- <1> 屏蔽 LVI 中断。
 - <2> 设置 LVIM.LVION 位为 1（以允许操作）。
 - <3> 插入一个 0.2ms（最大值）的等待时间或通过软件插入更多。
 - <4> 通过使用 LVIM.LVIF 位，检验供应电压是否大于检测电压。
 - <5> 清除 LVI 的中断请求标志。
 - <6> 不屏蔽 LVI 中断。

<准备停止操作>

将 LVION 位清除为 0。

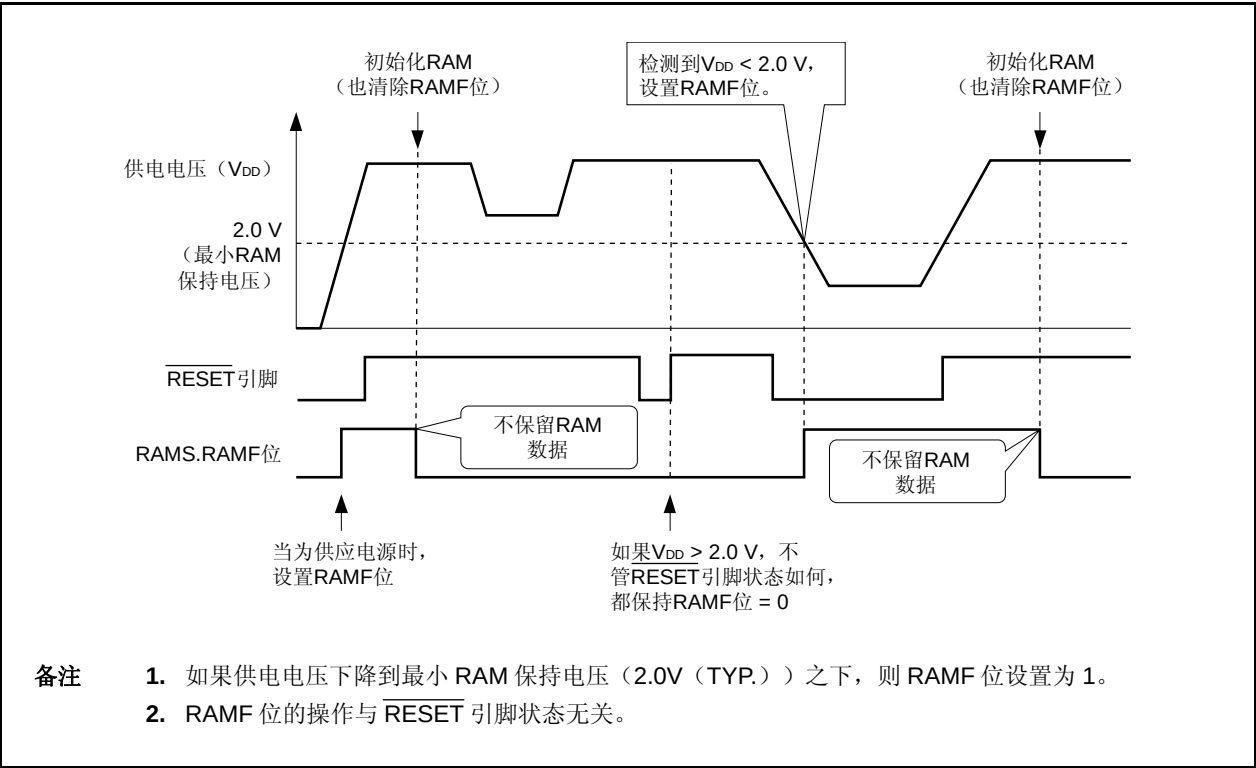
图 28-3. 低电压检测器的操作时序 (LVIMD 位=0)



28.5 RAM保持电压检测操作

比较供电电压和检测电压。当供电电压下降到低于检测电压时（包括应用电源），RAMS.RAMF 位置为 1。

图 28-4. RAM 保持电压检测功能的操作时序



29.1 功能

- CRC 操作电路用于数据块错误检测。
- 使用 CRC-CCITT ($X^{16} + X^{12} + X^5 + 1$) 生成多项式，为以 8 位为单位的任何长度的数据块生成 16 位 CRC 代码。
- 在初始值设置给 CRCD 寄存器之后，每当将 1 个字节数据传输到 CRCIN 寄存器时，就将 CRC 代码设置给 CRCD 数据寄存器。

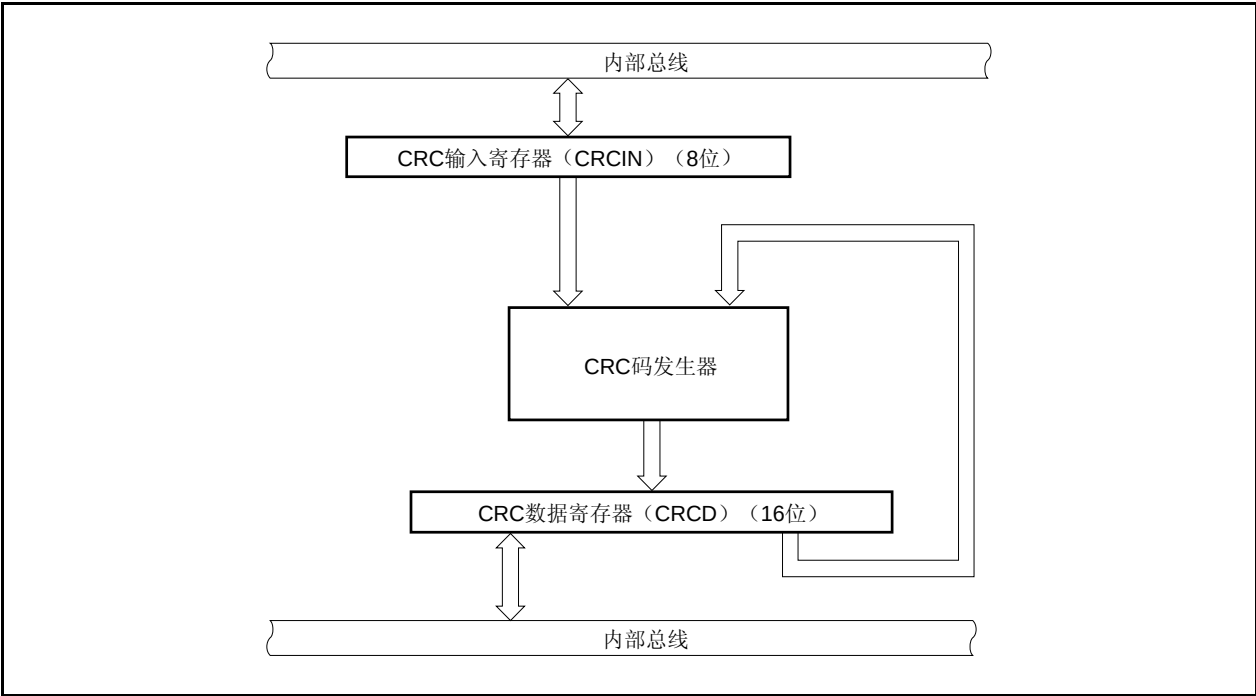
29.2 配置

CRC 功能包含以下硬件。

表 29-1. CRC 配置

项目	配置
控制寄存器	CRC 输入寄存器 (CRCIN) CRC 数据寄存器 (CRCD)

图 29-1. CRC 寄存器框图



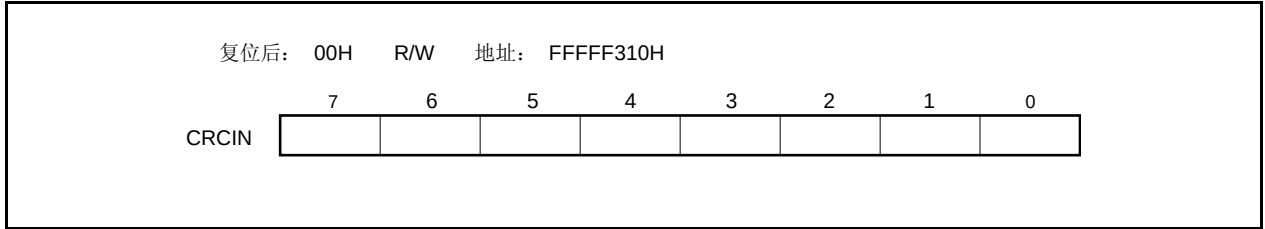
29.3 寄存器

(1) CRC 输入寄存器 (CRCIN)

CRCIN 寄存器是一个用于设置数据的 8 位寄存器。

该寄存器可按 8 位为单位进行读写。

复位后，该寄存器置为 00H。



(2) CRC 数据寄存器 (CRCD)

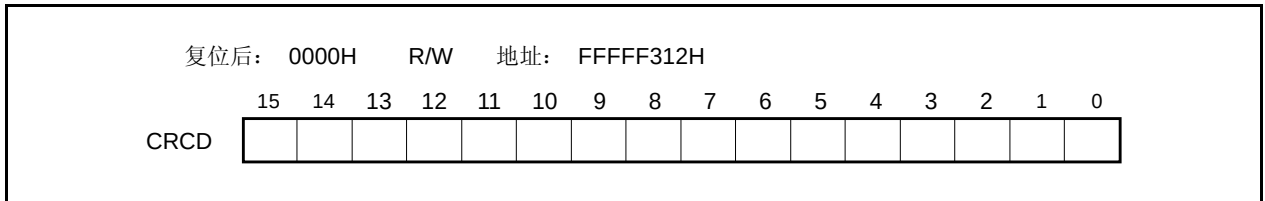
CRCD 寄存器是一个 16 位寄存器，用于存储 CRC-CCITT 的操作结果。

该寄存器可按 16 位为单位进行读写。

复位后，该寄存器置为 0000H。

注意事项 在下列状态中禁止访问 CRCD 寄存器。有关详细说明，参见 3.4.9 (2) 访问专用片上外围 I/O 寄存器。

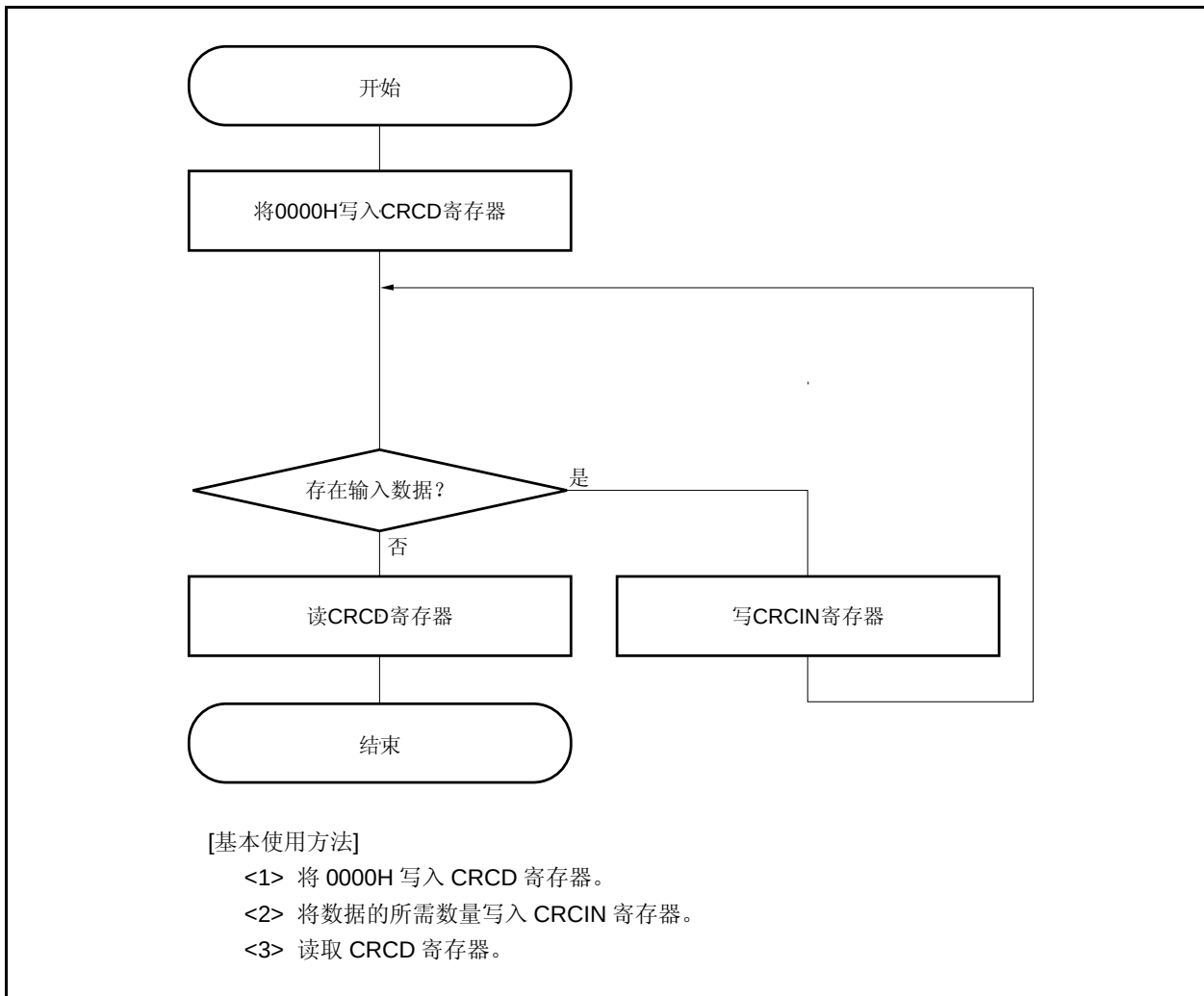
- 当 CPU 在副时钟下工作，且主时钟振荡停止时
- 当 CPU 工作于内部振荡时钟时



29.5 使用方法

下面描述了如何使用 CRC 逻辑电路。

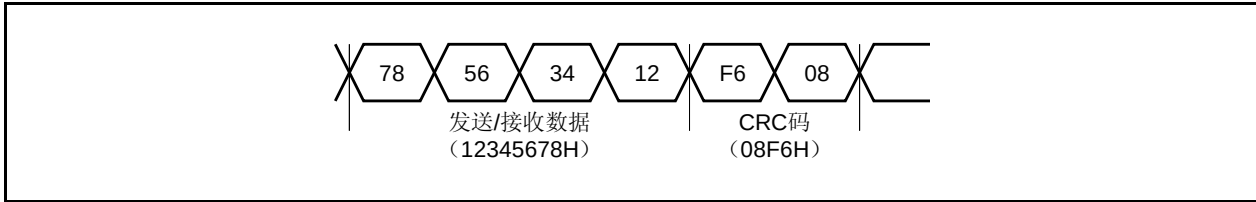
图 29-3. CRC 操作流程



当发送/接收数据由多个字节组成时，如果 CRC 码同发送/接收数据一起被发送/接收，那么就很容易检测到通信错误。

下面以发送 12345678H（0001 0010 0011 0100 0101 0110 0111 1000B）LSB 先行为例进行描述说明。

图 29-4. CRC 发送举例



发送端的设置步骤

- <1> 将初始值 0000H 写入 CRCD 寄存器。
- <2> 将首先发送的 1 字节的数据写入传输缓冲寄存器。（与此同时，也将相同数据写入到 CRCIN 寄存器中。）
- <3> 当发送多个字节的数据时，每次将发送数据写入发送缓冲寄存器时，就写入相同的数据到 CRCIN 寄存器。
- <4> 在所有的数据发送后，将 CRCD 寄存器（CRC 码）的内容写入发送缓冲寄存器并进行发送。（由于是 LSB 先行，发送数据从低字节开始，然后是高字节。）

接收端的设置步骤

- <1> 将初始值 0000H 写入 CRCD 寄存器。
- <2> 当第一个 1 字节数据接收完毕时，将接收数据写入 CRCIN 寄存器。
- <3> 如果接收多个字节的数据，每次接收完毕时都将接收数据写入 CRCIN 寄存器。（在正常接收的情况下，当所有的接收数据都已写入 CPCIN 寄存器，接收端 CRCD 寄存器的内容和发送端 CRCD 寄存器的内容相同。）
- <4> 接下来，CRC 码从发送端发送，与接收数据类似，将此数据写入 CRCIN 寄存器，。
- <5> 当所有数据接收（包括 CRC 码在内）完成后，如果 CRCD 寄存器的内容是 0000H，则接收正常。如果 CRCD 寄存器的内容不是 0000H，这表示发生了通信错误，因此，向发送端发出一个重新发送请求。

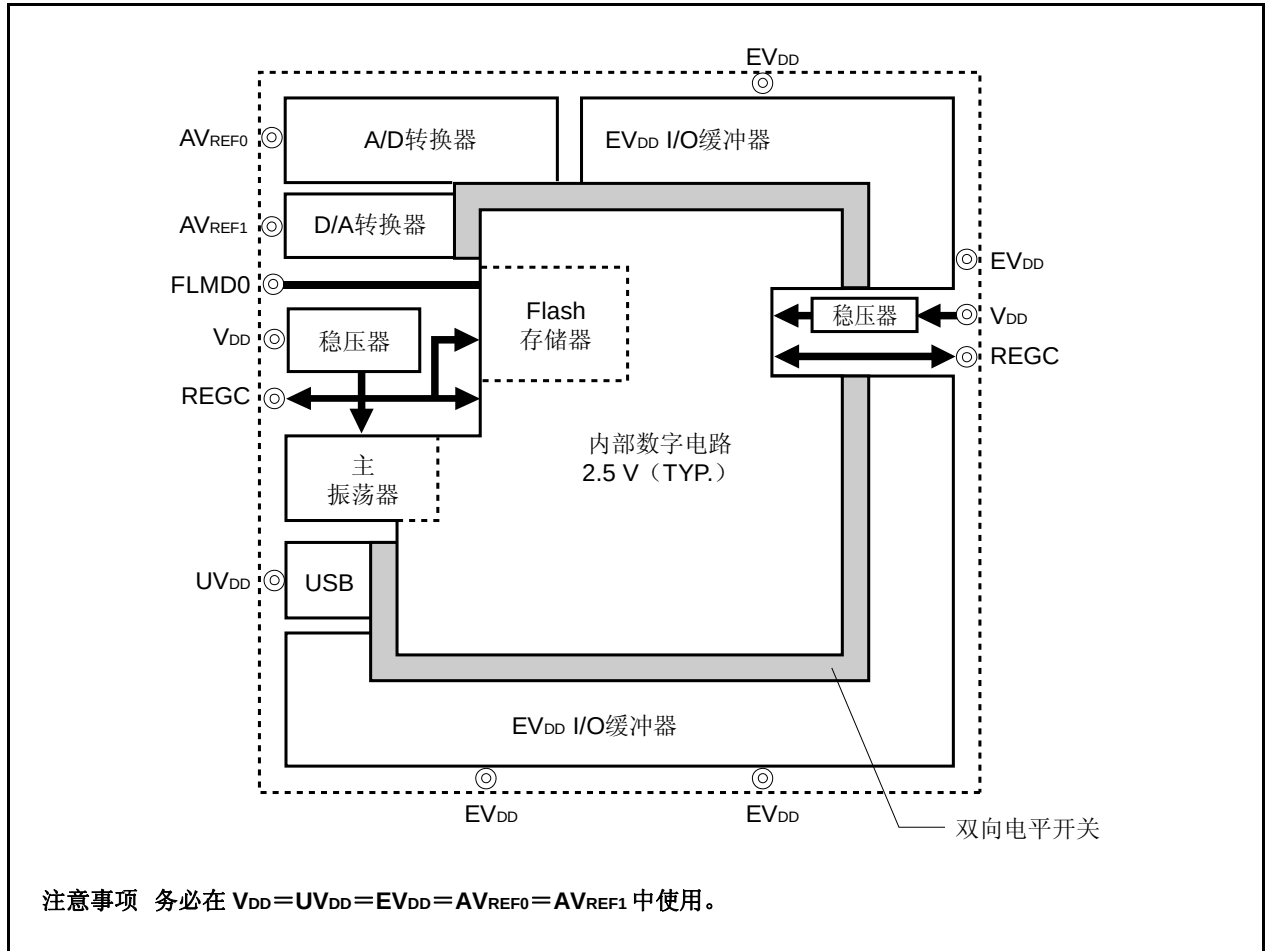
第三十章 稳压器

30.1 概述

V850ES/JG3-H 和 V850ES/JH3-H 单片机包括一个用来减少功率损耗和噪声的稳压器。

该稳压器为振荡器模块和内部逻辑电路（A/D 转换器，D/A 转换器以及输出缓冲器除外）提供一个阶梯减弱的 V_{DD} 电源供电电压。

图 30-1. 稳压器



30.2 操作

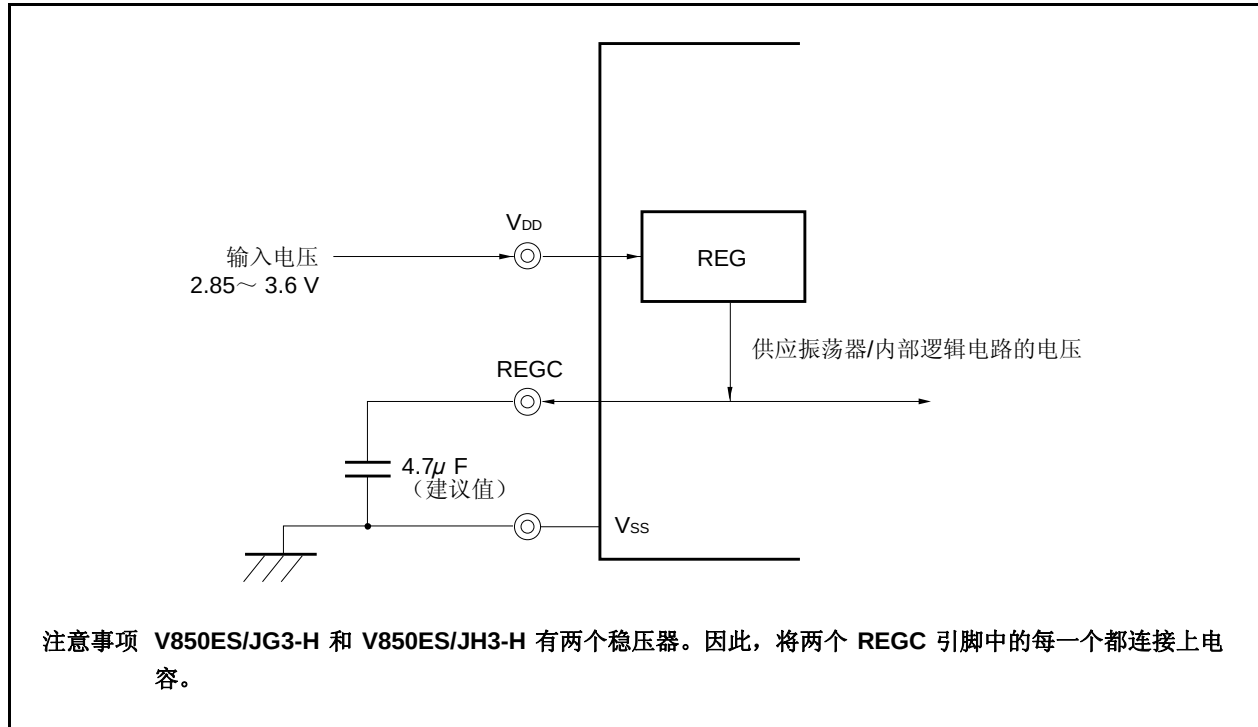
V850ES/JG3-H 和 V850ES/JH3-H 的稳压器总能在任何模式（正常工作模式、HALT 模式、IDLE1 模式、IDLE2 模式、STOP 模式、副时钟操作模式、副 IDLE 模式或复位期间）中操作。

务必要给 REGC 引脚^注连接一个电容（4.7 μF （初步值））来稳定稳压器输出。

稳压器引脚的连接方式图如下所示。

注 共有两个 REGC 引脚。

图 30-2. REGC 引脚连接



第三十一章 FLASH存储器

V850ES/JG3-H 和 V850ES/JH3-H 中并入了一个 flash 存储器。

• μ PD70F3760、70F3765、70F3770、70F3771:	256KB (flash 存储器)
• μ PD70F3761、70F3766:	384KB (flash 存储器)
• μ PD70F3762、70F3767:	512KB (flash 存储器)

Flash 存储器版本为开发环境和批量生产应用提供以下优势。

- 在 V850ES/JG3-H 和 V850ES/JH3-H 焊接到目标系统后改变软件。
- 开始批量生产时进行数据调整。
- 根据不同模型的小批量生产的要求区分软件。
- 便于库存管理。
- 出厂后更新软件。

31.1 特性

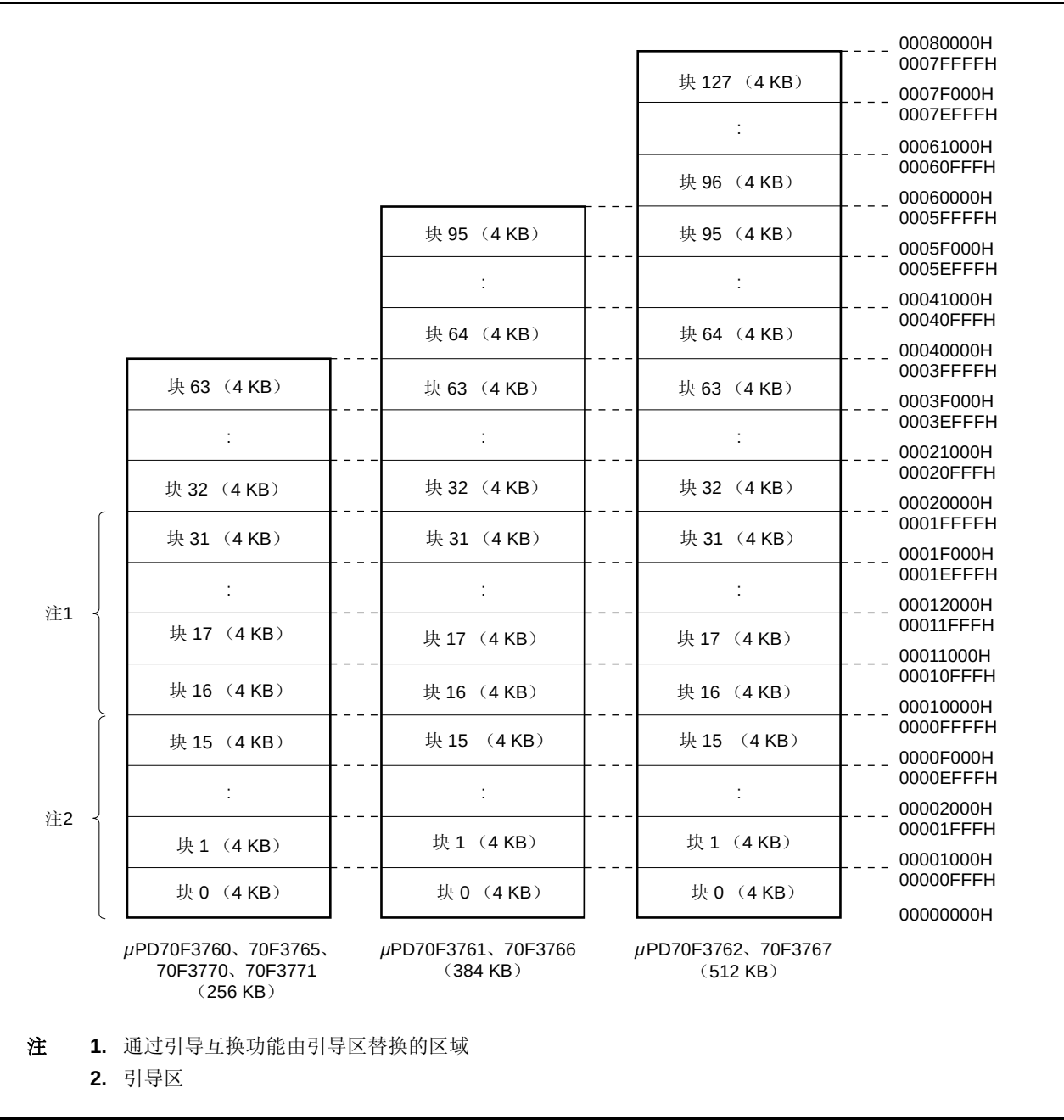
- 4 字节/1 时钟访问 (取指令时)
- 容量: 256/384/512 KB
- 重写电压: 用一个单供电电源擦除 / 写入
- 重写方法
 - 通过串行接口跟专用flash编程器通信进行重写 (在线/离线编程)
 - 通过用户程序重写 flash 存储器 (自编程)
- 支持禁止 Flash 存储器重写的功能 (安全功能)
- 通过自编程使用引导交换功能实现对整个 flash 存储器区域的安全重写
- 自编程期间可以响应中断。

31.2 存储器配置

V850ES/JG3-H 和 V850ES/JH3-H 的内部 flash 存储器区域被划分为 64、96 或 128 块，且能以块为单元进行编程/擦除。也可以一次性擦除所有模块。

当使用引导交换功能时，位于模块 0 至 15 地址处的物理存储区被位于模块 16 至 31 地址处的物理存储区替换。有关引导交换功能的细节，参见 31.5 自编程重写。

图 31-1. Flash 存储器映射



31.3 功能概述

不管 V850ES/JG3-H 和 V850ES/JH3-H 单片机是不是已经安装到目标系统之上（离线/在线编程），V850ES/JG3-H 和 V850ES/JH3-H 的内部 flash 存储器都可以使用专用 flash 程序器的重写功能进行重写。

另外，也支持安全功能，该安全功能禁止重写被写入内部 flash 存储器中的用户程序，这样，程序就不会被未经授权的人更改。

对于目标系统在生产/出厂后改写程序的应用，使用用户程序（自编程）的重写功能将是一个理想的选择。也支持安全重写整个 flash 存储器的引导交换功能。另外，在自编程期间支持中断服务，这样，flash 存储器可以在多种条件下进行重写，比如当与一个外部设备进行通信时。

表 31-1. 重写方法

重写方法	功能概述	操作模式
在线编程	在设备安装到目标系统上之后，通过使用专用 flash 编程器可对 Flash 存储器进行重写。	Flash 存储器编程模式
离线编程	设备安装到目标系统上之前，通过使用专用 flash 编程器和专用程序适配板（FA 系列）可对 Flash 存储器进行重写。	
自编程	通过执行由离线/在线编程的方式预先写入 flash 存储器的一个用户程序，可对 Flash 存储器进行重写。（自编程期间，不能从内部 flash 存储器区中获取指令，也不能对该区域进行数据访问。因此，重写程序必须预先发送到内部 RAM 或外部存储器）。	正常操作模式

备注 FA 系列是 Naito Densei Machida Mfg. Co., Ltd.的产品。

表 31-2. 基本功能

功能	功能概述	支持 (√: 支持, ×: 不支持)	
		在线/离线编程	自编程
块擦除	擦除指定存储模块的内容。	√	√
片擦除	一次性擦除整个存储区的内容。	√	×
写入	写入指定地址, 并且执行校验, 检查写入等级是否安全。	√	√
校验/校验和	比较从 flash 存储器读出的数据和从 flash 编程器传输的数据。	√	× (可由用户程序读取)
空白检查	检查整个存储区的擦除状态。	√	√
安全设置	禁止使用块擦除命令、片擦除命令、编程命令、读命令, 并禁止重写引导区。	√	× (仅当设置从允许变为禁止时支持)

下表列举了安全功能。在出厂后, 默认块擦除命令禁止、片擦除命令禁止以及编程命令禁止, 可以通过在线/离线编程进行重写来完成安全设置。每一项安全功能都可以同时与其它安全功能组合使用。

表 31-3. 安全功能

功能	功能概述
禁止块擦除命令	禁止在所有块上执行块擦除命令。通过执行片擦除命令来初始化禁止设置。
禁止片擦除命令	禁止在所有块上执行块擦除命令和片擦除命令。一旦该禁止功能被设置, 禁止设置就无法再次初始化, 因为不能执行片擦除命令。
禁止编程命令	执行在所有块上执行编程和块擦除命令。可以通过执行片擦除命令来初始化禁止设置。
禁止读取命令	禁止在所有块上执行读取命令。可以通过执行片擦除命令来初始化禁止设置。
禁止引导区重写	禁止在引导区执行写入命令、块擦除命令以及片擦除命令。一旦设置了禁止功能后, 就不能初始化禁止重写引导区的设置。

表 31-4. 安全设置

功能	当设置各项安全功能时的擦除、写入、读取操作 (√: 可执行, ×: 不可执行, -: 不支持)		安全设置的注意事项	
	在线/离线编程	自编程	在线/离线编程	自编程
禁止块擦除命令	块擦除命令: × 片擦除命令: √ 编程命令: √ 读取命令: √	块擦除 (flash 块擦除): √ 片擦除: - 写入 (flash 字写入): √ 读取 (flash 字读取): √	通过片擦除命令可使禁止设置初始化。	仅当设置从允许变为禁止受支持
禁止片擦除命令	块擦除命令: × 片擦除命令: × 编程命令: √ ¹ 读取命令: √	块擦除 (flash 块擦除): √ 片擦除: - 写入 (flash 字写入): √ 读取 (flash 字读取): √	不能被初始化禁止设置。	
禁止编程命令	块擦除命令: × 片擦除命令: √ 编程命令: × 读取命令: √	块擦除 (flash 块擦除): √ 片擦除: - 写入 (flash 字写入): √ 读取 (flash 字读取): √	通过片擦除命令可使禁止设置初始化。	
禁止读取命令	块擦除命令: √ 片擦除命令: √ 编程命令: √ 读取命令: ×	块擦除 (flash 块擦除): √ 片擦除: - 写入 (flash 字写入): √ 读取 (flash 字读取): √	通过片擦除命令可使禁止设置初始化。	
禁止引导区重写	块擦除命令: × ² 片擦除命令: × 编程命令: × ² 读取命令: √	块擦除 (flash 块擦除): × ² 片擦除: - 写入 (flash 字写入): × ² 读取 (flash 字读取): √	不能初始化禁止设置。	仅当设置从允许变为禁止时受支持 ³

- 注
1. 在这种情况下，由于擦除命令无效，不能写入与已经写入 flash 存储器的数据不同的数据。
 2. 除引导区之外可执行。
 3. 复位输入后引导区重写禁止功能有效。

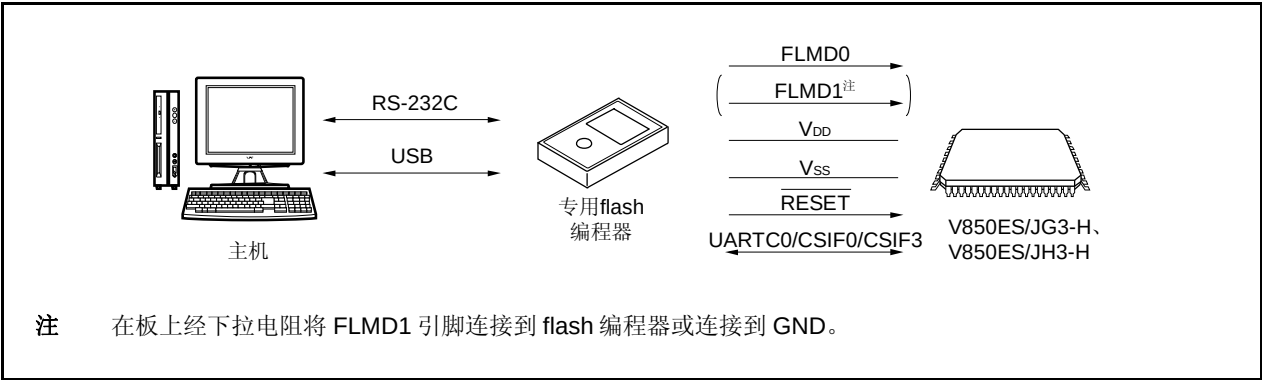
31.4 通过专用Flash编程器重写

V850ES/JG3-H 和 V850ES/JH3-H 安装到目标系统上后（在线编程），可以使用专用 flash 编程器对 flash 存储器进行重写。Flash 存储器也可以使用专用的编程适配器（FA 系列）在设备安装到目标系统（离线编程）上之前被重写。

31.4.1 编程环境

以下显示的是将程序写入 V850ES/JG3-H 和 V850ES/JH3-H 的 flash 存储器时所需的环境。

图 31-2. 将程序写入 Flash 存储器所需的环境



需要一台主机来控制专用 flash 编程器。

UARTC0、CSIF0 或 CSIB3 用作专用 flash 编程器和 V850ES/JG3-H 及 V850ES/JH3-H 之间的接口，通过它执行对单片机的写入、擦除等。离线写入则需要一个专用的编程适配器（FA 系列）。

备注 FA 系列是 Naito Densei Machida Mfg. Co., Ltd.的产品。

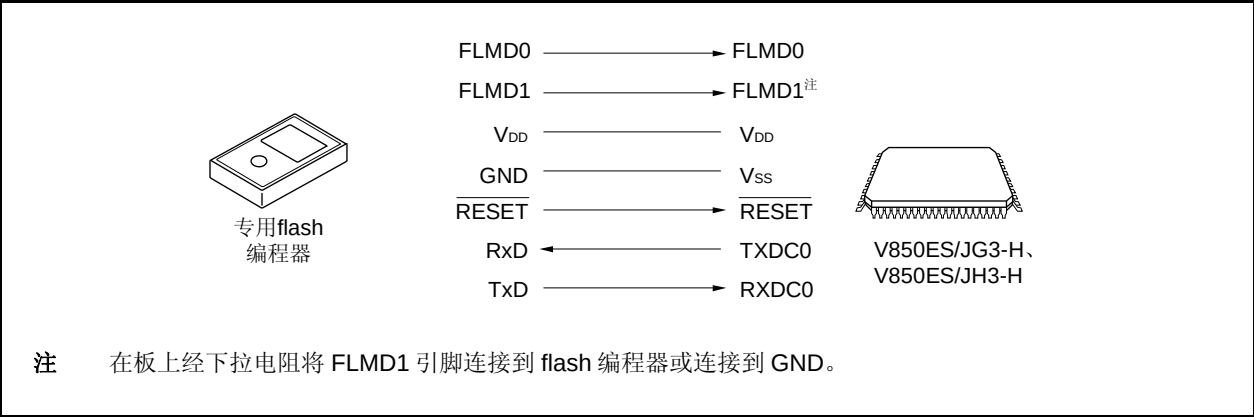
31.4.2 通信模式

使用 V850ES/JG3-H 或 V850ES/JH3-H 的 UARTC0、CSIF0 或 CSIF3 接口，通过串行通信的方式执行专用 flash 编程器与 V850ES/JG3-H 或 V850ES/JH3-H 间的通信。

(1) UARTC0

传输速率： 9, 600 至 153, 600 bps

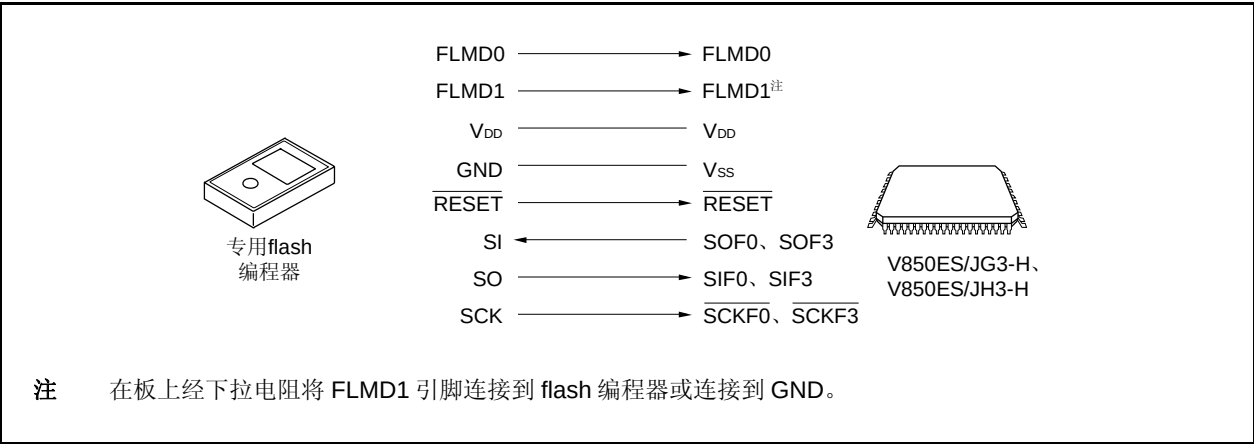
图 31-3. 与专用 Flash 编程器的通信（UARTC0）



(2) CSIF0、CSIF3

串行时钟：5 MHz 或更少（MSB 先行）

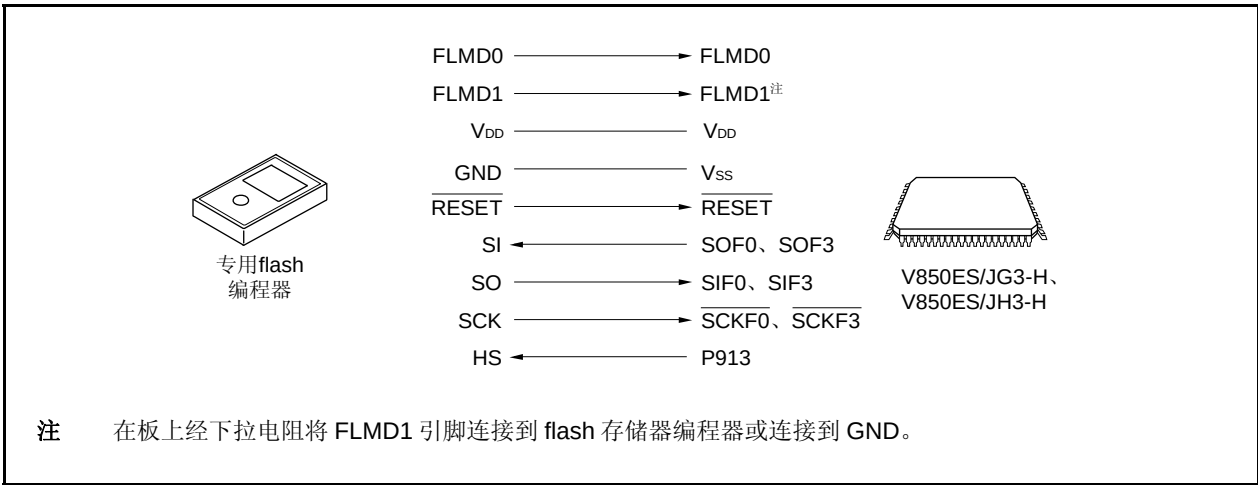
图 31-4. 与专用 Flash 编程器的通信（CSIF0、CSIF3）



(3) CSIF0 + HS、CSIF3 + HS

串行时钟：5 MHz 或更少（MSB 先行）

图 31-5. 与专用 Flash 编程器的通信（CSIF0 + HS、CSIF3 + HS）



专用 flash 编程器输出传输时钟，且 V850ES/JG3-H 和 V850ES/JH3-H 作为从机工作。

当 PG-FP5 用作专用 flash 编程器时，会产生以下信号提供给 V850ES/JG3-H 和 V850ES/JH3-H。有关详细内容，参见 PG-FP5 用户手册（U18865E）。

表 31-5. 专用 Flash 编程器（PG-FP5）的信号连接

PG-FP5			V850ES/JG3-H, V850ES/JH3-H	连接处理		
信号名称	I/O	引脚功能	引脚名称	UARTC0	CSIF0、 CSIF3	CSIF0+HS、 CSIF3+HS
FLMD0	输出	允许/禁止写入	FLMD0	⊙	⊙	⊙
FLMD1	输出	允许/禁止写入	FLMD1	⊙ ^{注 1}	⊙ ^{注 1}	⊙ ^{注 1}
VDD	—	V _{DD} 电压生产/电压监控	V _{DD}	⊙	⊙	⊙
GND	—	地	V _{SS}	⊙	⊙	⊙
CLK	输出	输出到 V850ES/JG3-H 和 V850ES/JH3-H 的时钟	X1, X2	× ^{注 2}	× ^{注 2}	× ^{注 2}
RESET	输出	复位信号	RESET	⊙	⊙	⊙
SI/RxD	输入	接收信号	SOF0、SOF3/ TXDC0	⊙	⊙	⊙
SO/TxD	输出	发送信号	SIF0、SIF3/ RXDC0	⊙	⊙	⊙
SCK	输出	传输时钟	SCKF0, SCKF3	×	⊙	⊙
HS	输入	用于 CSIF0 + HS、CSIF3 + HS 通信的握手信号	P913	×	×	⊙

注 1. 如图 31-6 和 31-7 所示连接这些引脚，或在板上通过下拉电阻将它们连接到 GND。
 2. 不能通过 Flash 编程器的 CLK 引脚提供时钟。在板上创建一个振荡器并提供时钟。

备注 ⊙：必须连接。
 ×：不必连接。

表 31-6. V850ES/JG3-H Flash 写入适配器的连线 (FA-100GC-UEU-B) (1/2)

Flash 编程器 (FG-FP5) 连接引脚			FA 板引脚名称	使用 CSIF0 + HS		使用 CSIF0		使用 UARTC0	
信号名称	I/O	引脚功能		引脚名称	引脚编号	引脚名称	引脚编号	引脚名称	引脚编号
SI/RxD	输入	接收信号	SI	P41/SOF0/ SCL01	23	P41/SOF0/ SCL01	23	P30/TXDC0/ SOF4	25
SO/TxD	输出	发送信号	SO	P40/SIF0/ SDA01	22	P40/SIF0/ SDA01	22	P31/RXDC0/ INTP07/SIF4	26
SCK	输出	传输时钟	SCK	P42/ $\overline{\text{SCKF0}}$	24	P42/ $\overline{\text{SCKF0}}$	24	不需要	—
CLK	输出	至 V850ES/JG3-H 的时钟	X1	不需要	—	不需要	—	不需要	—
			X2	不需要	—	不需要	—	不需要	—
/RESET	输出	复位信号	/RESET	$\overline{\text{RESET}}$	14	$\overline{\text{RESET}}$	14	$\overline{\text{RESET}}$	14
FLMD0	输出	写入电压	FLMD0	FLMD0	8	FLMD0	8	FLMD0	8
FLMD1	输出	写入电压	FLMD1	PDL5/AD5/ FLMD1	76	PDL5/AD5/ FLMD1	76	PDL5/AD5/ FLMD1	76
HS	输入	用于 CSIF0 + HS 通信的握手信号	RESERVE/ HS	P913	55	不需要	—	不需要	—
VDD	—	VDD 电压生成/ 电压监控	VDD	V _{DD}	9、60	V _{DD}	9、60	V _{DD}	9、60
				EV _{DD}	34, 63, 80	EV _{DD}	34, 63, 80	EV _{DD}	34, 63, 80
				UV _{DD}	19	UV _{DD}	19	UV _{DD}	19
				AV _{REF0}	1	AV _{REF0}	1	AV _{REF0}	1
				AV _{REF1}	5	AV _{REF1}	5	AV _{REF1}	5
GND	—	接地	GND	V _{SS}	11, 33, 62, 7	V _{SS}	11, 33, 62, 79	V _{SS}	11, 33, 62, 79
				AV _{SS}	2	AV _{SS}	2	AV _{SS}	2

- 注意事项
1. 务必通过 4.7 μF (初步值) 的电容将 REGC 引脚连接至 GND。
 2. 不能从 flash 编程器的 CLK 引脚提供时钟。
在板上创建一个振荡器并提供时钟。

表 31-6. V850ES/JG3-H Flash 写入适配器的连线（FA-100GC-UEU-B）（2/2）

Flash 编程器（FG-FP5）连接引脚			FA 板引脚名称	使用 CSIF3 + HS		使用 CSIF3	
信号名称	I/O	引脚功能		引脚名称	引脚编号	引脚名称	引脚编号
SI/RxD	输入	接收信号	SI	P911/A11/SOF3	53	P911/A11/SOF3	53
SO/TxD	输出	发送信号	SO	P910/A10/SIF3	52	P910/A10/SIF3	52
SCK	输出	传输时钟	SCK	P912/A12/ $\overline{\text{SCKF3}}$	54	P912/A12/ $\overline{\text{SCKF3}}$	54
CLK	输出	至 V850ES/JG3-H 的时钟	X1	不需要	–	不需要	–
			X2	不需要	–	不需要	–
/RESET	输出	复位信号	/RESET	$\overline{\text{RESET}}$	14	$\overline{\text{RESET}}$	14
FLMD0	输出	写入电压	FLMD0	FLMD0	8	FLMD0	8
FLMD1	输出	写入电压	FLMD1	PDL5/AD5/FLMD1	76	PDL5/AD5/FLMD1	76
HS	输入	用于 CSIO + HS 通信的握手信号	RESERVE/HS	P913	55	不需要	–
VDD	–	VDD 电压生成/电压监控	VDD	V _{DD}	9、60	V _{DD}	9、60
				EV _{DD}	34, 63, 80	EV _{DD}	34, 63, 80
				UV _{DD}	19	UV _{DD}	19
				AV _{REF0}	1	AV _{REF0}	1
				AV _{REF1}	5	AV _{REF1}	5
GND	–	地	GND	V _{SS}	11, 33, 62, 79	V _{SS}	11, 33, 62, 79
				AV _{SS}	2	AV _{SS}	2

- 注意事项
1. 务必通过 4.7 μF （初步值）的电容将 REGC 引脚连接至 GND。
 2. 不能从 Flash 编程器的 CLK 引脚提供时钟。
在板上创建一个振荡器并提供时钟。

图 31-6. V850ES/JG3-H Flash 写入适配器 (FA-100GC-UEU-B) 的连线举例
(CSIF3 + HS 模式下) (1/2)

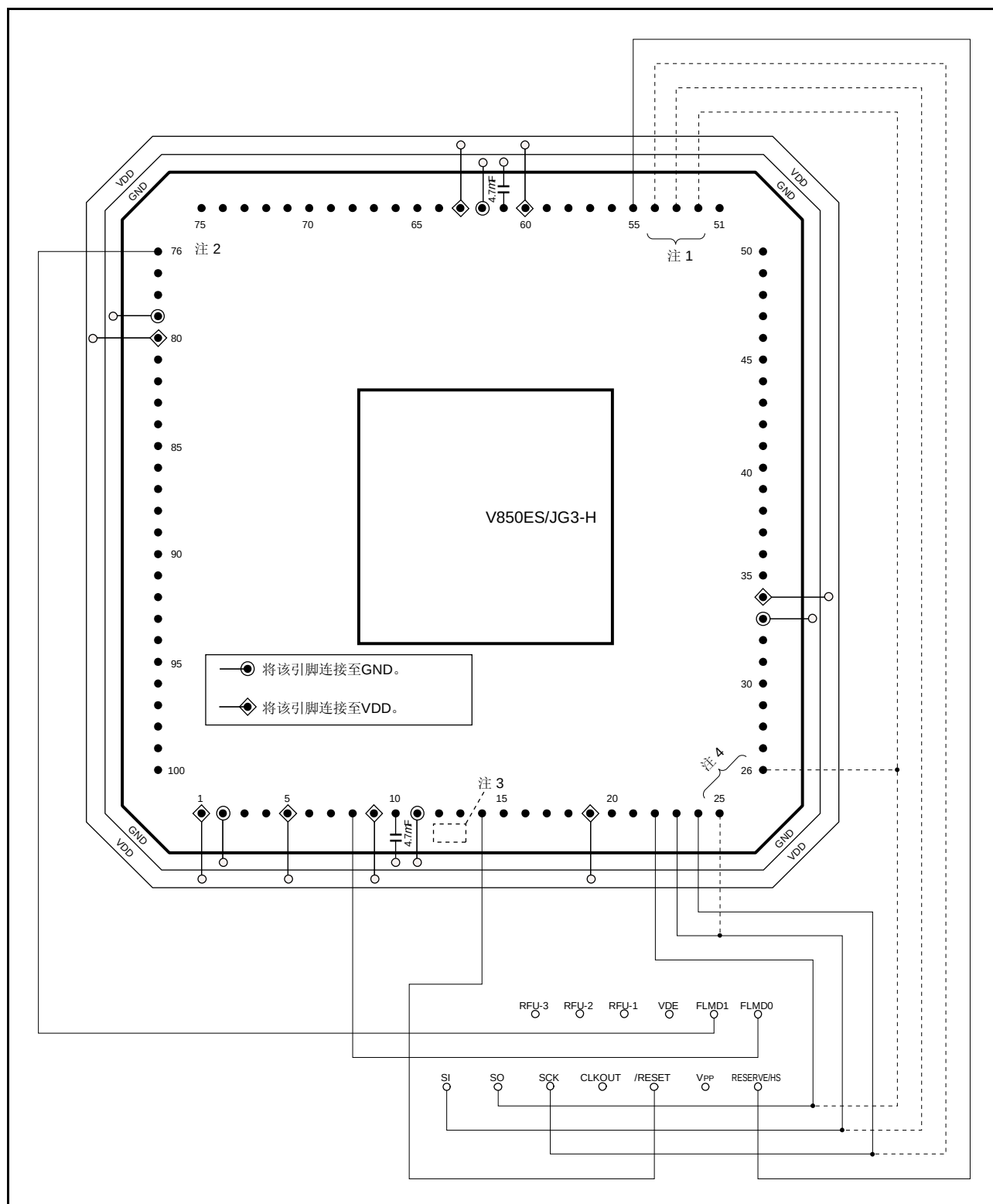
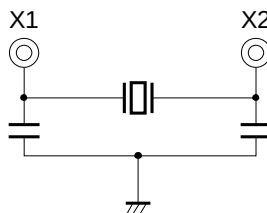


图 31-6. V850ES/JG3-H Flash 写入适配器 (FA-100GC-UEU-B) 的连线举例
(CSIF3 + HS 模式下) (2/2)

- 注
1. 使用 CSIF3 时的相应引脚。
 2. 如下图中所示连接 FLMD1 引脚，或在板上通过一个下拉电阻将它连接到 GND。
 3. 在 flash 写入适配器（以虚线表示）上创建一个振荡器并提供时钟。
这里给出一个振荡器的例子。

示例：



4. 使用 UARTC0 时的相应引脚。

注意事项 禁止将高电平输入 $\overline{\text{DRST}}$ 引脚。

- 备注
1. 处理没有显示的引脚时，与未使用引脚的处理一致（参见 2.3 引脚 I/O 电路类型，I/O 缓冲器电源供应以及未使用引脚的连接）。
 2. 该适配器用于 100-引脚 LQFP 塑料封装。

表 31-7. V850ES/JH3-H Flash 写入适配器的连线 (FA-128GF-GAT-B) (1/2)

Flash 编程器 (FG-FP5) 连接引脚			FA 板引脚名称	使用 CSIF0 + HS		使用 CSIF0		使用 UARTC0	
信号名称	I/O	引脚功能		引脚名称	引脚编号	引脚名称	引脚编号	引脚名称	引脚编号
SI/RxD	输入	接收信号	SI	P41/SOF0/ SCL01	30	P41/SOF0/ SCL01	30	P30/TXDC0/ SOF4	37
SO/TxD	输出	发送信号	SO	P40/SIF0/ SDA01	29	P40/SIF0/ SDA01	29	P31/RXDC0/ INTP07/SIF4	38
SCK	输出	传输时钟	SCK	P42/ $\overline{\text{SCKF0}}$	31	P42/ $\overline{\text{SCKF0}}$	31	不需要	—
CLK	输出	至 V850ES/JH3-H 的时钟	X1	不需要	—	不需要	—	不需要	—
			X2	不需要	—	不需要	—	不需要	—
/RESET	输出	复位信号	/RESET	$\overline{\text{RESET}}$	18	$\overline{\text{RESET}}$	18	$\overline{\text{RESET}}$	18
FLMD0	输出	写入电压	FLMD0	FLMD0	12	FLMD0	12	FLMD0	12
FLMD1	输出	写入电压	FLMD1	PDL5/AD5/ FLMD1	103	PDL5/AD5/ FLMD1	103	PDL5/AD5/ FLMD1	103
HS	输入	用于 CSIF0+HS 通信的握手信号	RESERVE/ HS	P913	69	不需要	—	不需要	—
VDD	—	VDD 电压生成/ 电压监控	VDD	V _{DD}	13, 82	V _{DD}	13, 82	V _{DD}	13, 82
				EV _{DD}	46, 61, 85, 107	EV _{DD}	46, 61, 85, 107	EV _{DD}	46, 61, 85, 107
				UV _{DD}	23	UV _{DD}	23	UV _{DD}	23
				AV _{REF0}	1	AV _{REF0}	1	AV _{REF0}	1
				AV _{REF1}	5	AV _{REF1}	5	AV _{REF1}	5
GND	—	地	GND	V _{SS}	15, 45, 60, 84, 106	V _{SS}	15, 45, 60, 84, 106	V _{SS}	15, 45, 60, 84, 106
				AV _{SS}	2	AV _{SS}	2	AV _{SS}	2

注意事项 1. 务必通过 4.7 μF (初步值) 的电容器将 REGC 引脚连接至 GND。

2. 不能从 Flash 编程器的 CLK 引脚提供时钟。

在板上创建一个振荡器并提供时钟。

表 31-7. V850ES/JH3-H Flash 写入适配器的连线 (FA-128GF-GAT-B) (2/2)

Flash 编程器 (FG-FP5) 连接引脚			FA 板引脚名称	使用 CSIF3 + HS		使用 CSIF3	
信号名称	I/O	引脚功能		引脚名称	引脚编号	引脚名称	引脚编号
SI/RxD	输入	接收信号	SI	P911/A11/SOF3	67	P911/A11/SOF3	67
SO/TxD	输出	发送信号	SO	P910/A10/SIF3	66	P910/A10/SIF3	66
SCK	输出	传输时钟	SCK	P912/A12/SCKF3	68	P912/A12/SCKF3	68
CLK	输出	至 V850ES/JH3-H 的时钟	X1	不需要	–	不需要	–
			X2	不需要	–	不需要	–
/RESET	输出	复位信号	/RESET	RESET	16	RESET	16
FLMD0	输出	写入电压	FLMD0	FLMD0	10	FLMD0	10
FLMD1	输出	写入电压	FLMD1	PDL5/AD5/FLMD1	103	PDL5/AD5/FLMD1	78
HS	输入	用于 CSIO + HS 通信的握手信号	RESERVE/HS	P913	69	不需要	–
VDD	–	VDD 电压生成/电压监控	VDD	V _{DD}	13, 82	V _{DD}	13, 82
				EV _{DD}	46, 61, 85, 107	EV _{DD}	46, 61, 85, 107
				UV _{DD}	23	UV _{DD}	23
				AV _{REF0}	1	AV _{REF0}	1
				AV _{REF1}	5	AV _{REF1}	5
GND	–	地	GND	V _{SS}	15, 45, 60, 84, 106	V _{SS}	15, 45, 60, 84, 106
				AV _{SS}	2	AV _{SS}	2

注意事项 1. 务必通过 4.7 μ F (初步值) 的电容器将 REGC 引脚连接至 GND。

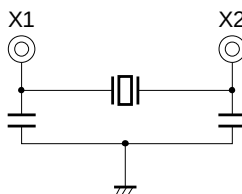
2. 不能从 Flash 编程器的 CLK 引脚提供时钟。

在板上创建一个振荡器并提供时钟。

图 31-7. V850ES/JH3-H Flash 写入适配器 (FA-128GF-GAT-B) 的连线举例
(CSIF0 + HS 模式下) (2/2)

- 注
1. 使用 CSIF3 时的相应引脚。
 2. 如下图中所示连接 FLMD1 引脚，或在板上通过一个下拉电阻将它连接到 GND。
 3. 在 flash 写入适配器（以虚线表示）上创建一个振荡器并提供时钟。
这里是一个振荡器的例子。

示例：



4. 使用 UARTC0 时的相应引脚。

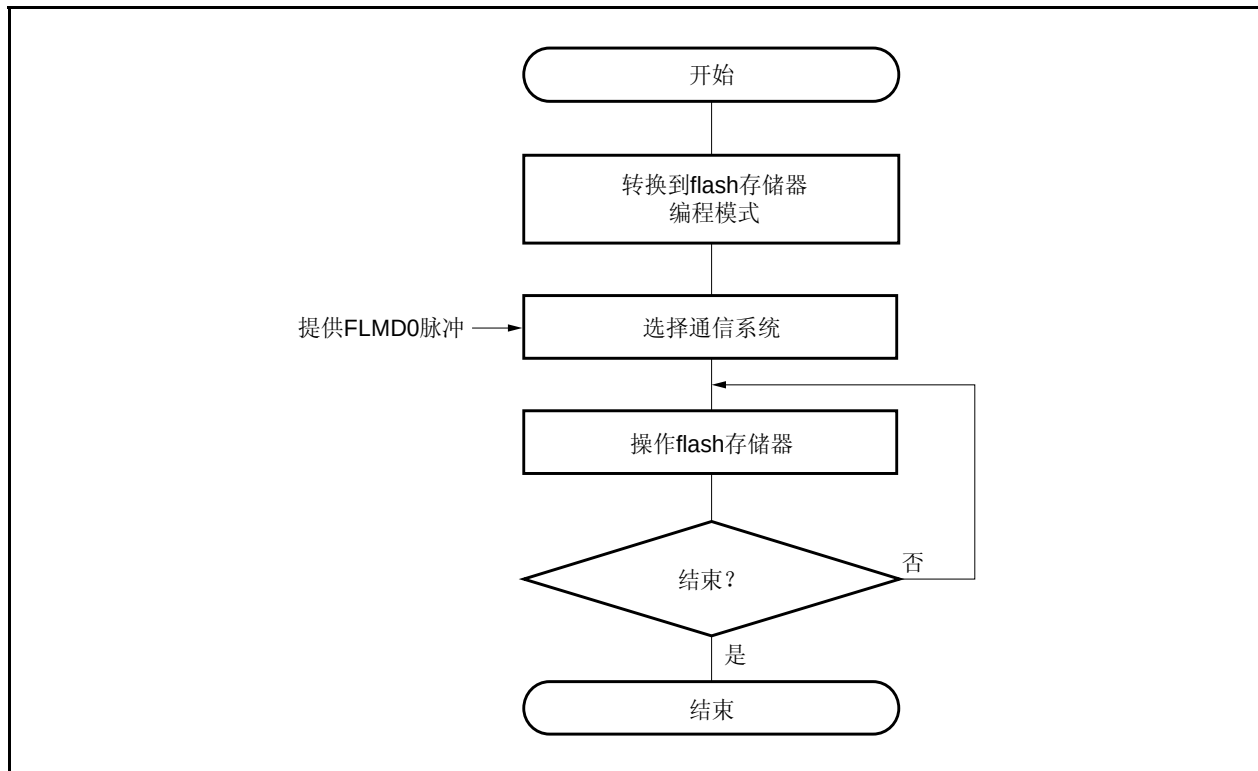
注意事项 禁止将高电平输入 $\overline{\text{DRST}}$ 引脚。

- 备注
1. 处理没有显示的引脚时，与未使用引脚的处理一致（参见 2.3 引脚 I/O 电路类型，I/O 缓冲器电源供应以及未使用引脚的连接）。
 2. 该适配器用于 128-引脚 LQFP 封装。

31.4.3 Flash存储器控制

下图显示了操作 flash 存储器的步骤:

图 31-8. 操作 Flash 存储器的步骤

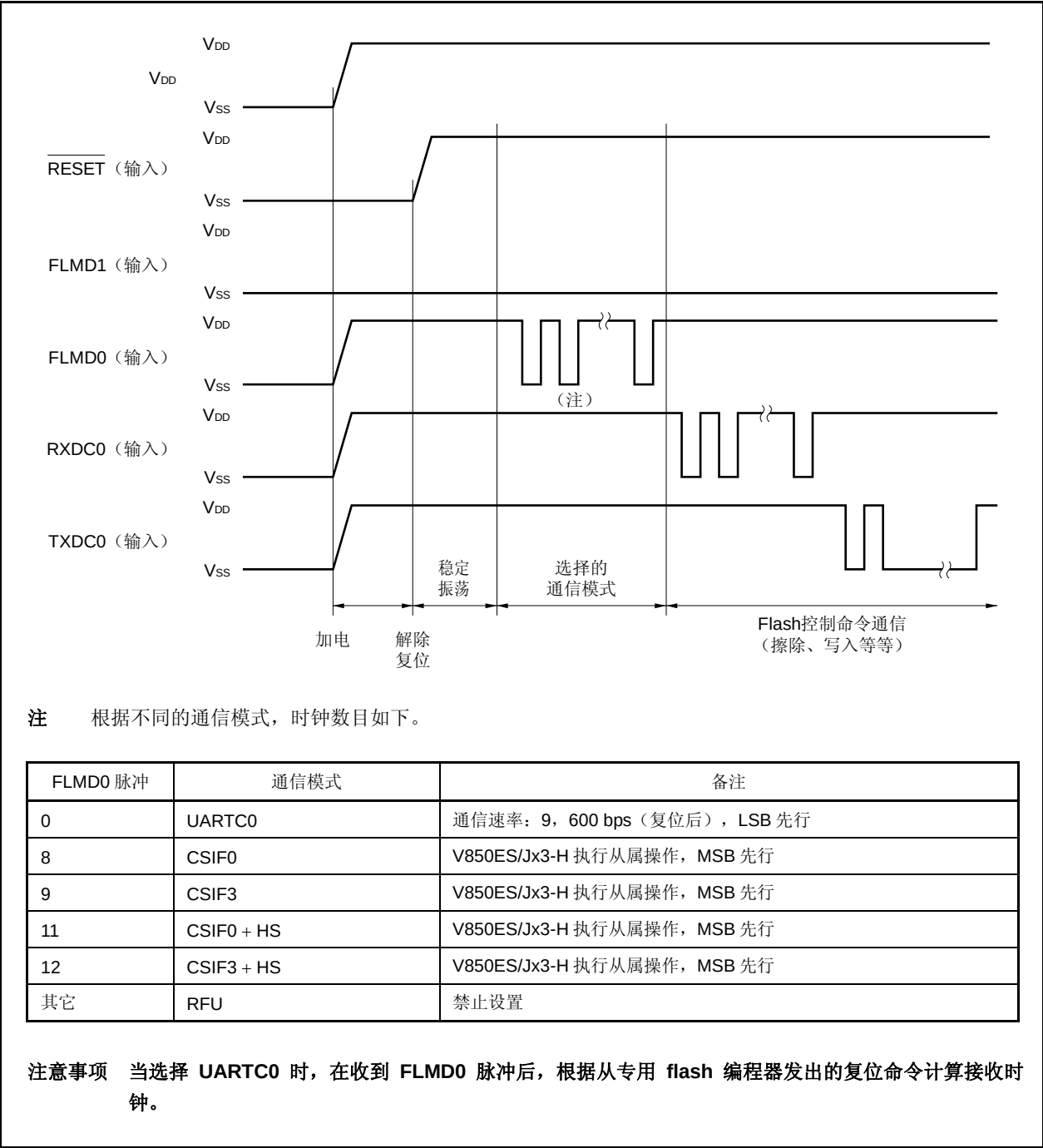


31.4.4 通信模式的选择

在 V850ES/JG3-H 和 V850ES/JH3-H 中，转换为 flash 存储器编程模式后，通过将脉冲（最多为 12 个脉冲）输入到 FLMD0 引脚，来选择通信模式。由专用 flash 编程器生成 FLMD0 脉冲。

下面显示了脉冲数目和通信模式之间的关系。

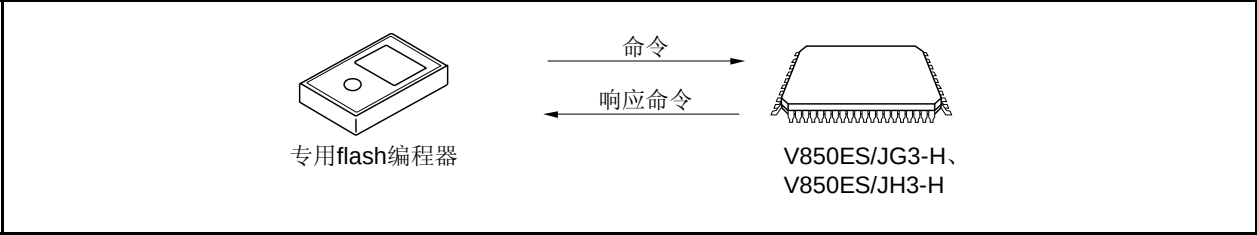
图 31-9. 通信模式的选择



31.4.5 通信命令

V850ES/JG3-H 和 V850ES/JH3-H 通过命令的方式与专用 flash 编程器进行通信。从专用 flash 编程器发送到 V850ES/JG3-H 和 V850ES/JH3-H 中的信号被称作“命令”。从 V850ES/JG3-H 和 V850ES/JH3-H 发送到专用 flash 编程器中的响应信号被称作“响应命令”。

图 31-10. 通信命令



以下显示了 V850ES/JG3-H 和 V850ES/JH3-H 单片机中用于 flash 存储器控制的命令。所有这些命令都由专用 flash 编程器发出，V850ES/JG3-H 和 V850ES/JH3-H 执行相应命令的处理。

表 31-7. Flash 存储器控制命令

分类	命令名称	支持			功能
		CSIF0、CSIF3	CSIF0 + HS, CSIF3 + HS	UARTC0	
空白检查	块空白检查命令	√	√	√	检查指定块的存储内容是否已被正确擦除。
擦除	片擦除命令	√	√	√	擦除整个存储器的内容。
	块擦除命令	√	√	√	擦除指定块的存储内容。
写入	编程命令	√	√	√	写入指定的地址范围并执行内容校验检查。
校验	校验命令	√	√	√	将指定地址范围的存储内容与从 flash 编程器传输的数据进行比较。
	校验和命令	√	√	√	读取指定地址范围中的校验和。
系统设置、控制	硅签字命令	√	√	√	读取硅签字信息。
	安全设置命令	√	√	√	禁止片擦除命令，块擦除命令，编程命令，读取命令以及引导区重写。

31.4.6 引脚连接

进行在线写入操作时，在目标系统上安装一个连接器用来连接专用 flash 编程器。同时，还包括将正常操作模式转换到 flash 存储器编程模式的功能。

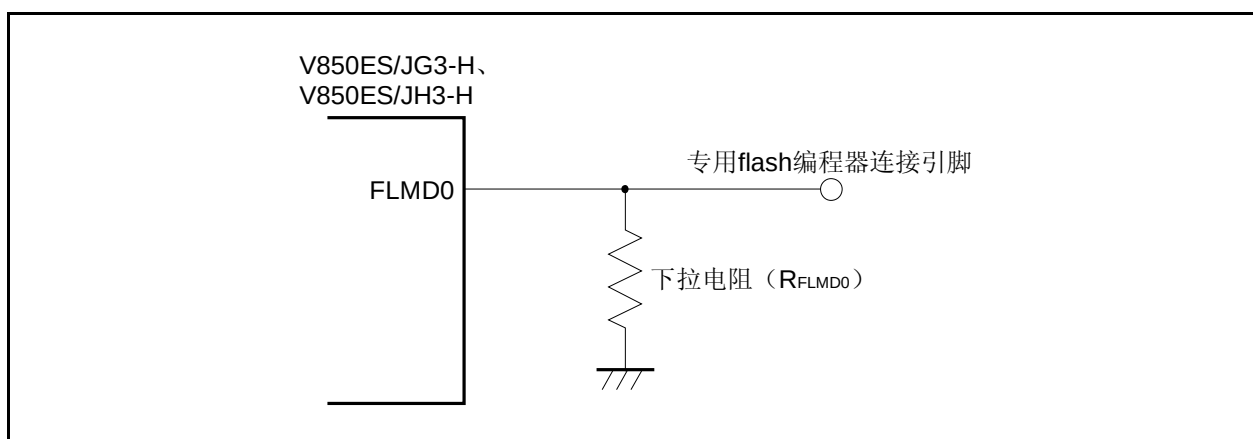
在 flash 存储器编程模式中，所有那些不用于 flash 存储器编程的引脚状态变为复位后的瞬时状态。因此，当外部器件不认可复位后的瞬时状态时需要进行引脚处理。

(1) FLMD0 引脚

在正常操作模式中，将 V_{SS} 电压输入至 FLMD0 引脚。在 flash 存储器编程模式中，为 FLMD0 引脚提供一个值为 V_{DD} 的写入电压。

因为 FLMD0 引脚在自编程模式下作为写保护引脚，所以在写入 flash 存储器之前，必须通过端口控制等方式为 FLMD0 引脚提供一个值为 V_{DD} 的电压。有关细节，参见 31.5.5 (1) FLMD0 引脚。

图 31-11. FLMD0 引脚连接举例



(2) FLMD1 引脚

当在 FLMD0 引脚输入 0 V 时，FLMD1 引脚不起作用。当给 FLMD0 引脚提供 V_{DD} 的电压时，进入 flash 存储器编程模式，因而必须在 FLMD1 引脚输入 0 V 电压。下面显示了 FLMD1 引脚连接的例子。

图 31-12. FLMD1 引脚连接举例

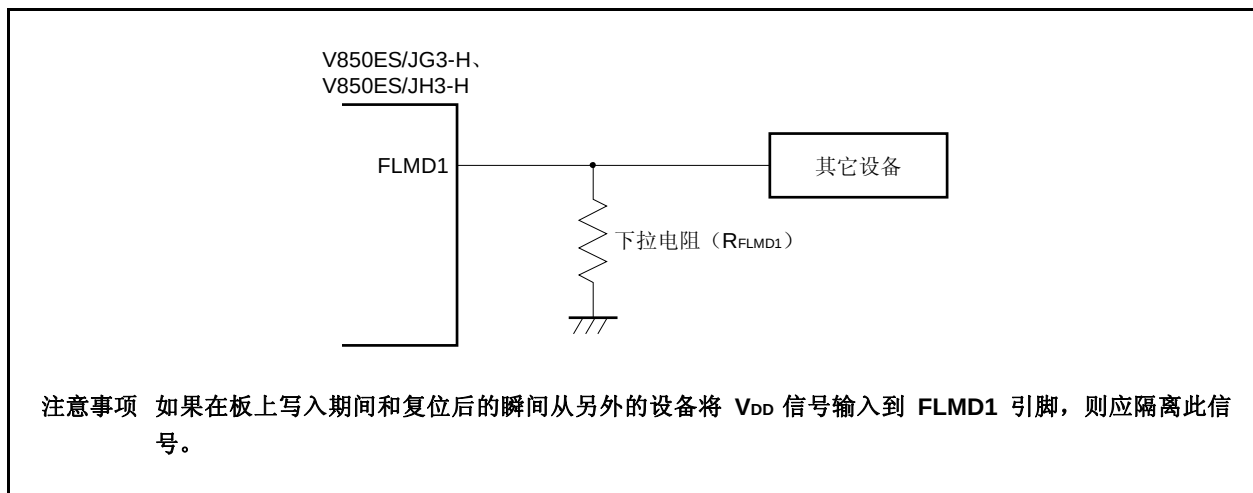


表 31-8. FLMD0 和 FLMD1 引脚间的关系以及复位解除后的操作模式

FLMD0	FLMD1	操作模式
0	可忽略	正常操作模式
V _{DD}	0	Flash 存储器编程模式
V _{DD}	V _{DD}	禁止设置

(3) 串行接口引脚

以下显示了各串行接口使用的引脚。

表 31-9. 串行接口使用的引脚

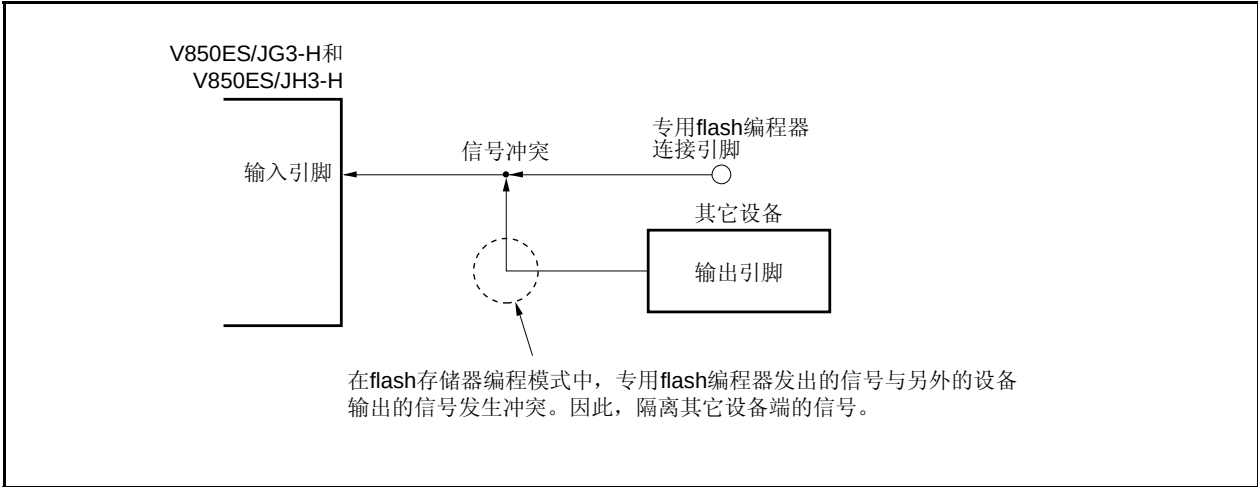
串行接口	使用的引脚
UARTC0	TXDC0, RXDC0
CSIF0	SOF0, SIF0, $\overline{\text{SCKF0}}$
CSIF3	SOF3, SIF3, $\overline{\text{SCKF3}}$
CSIF0 + HS	SOF0, SIF0, $\overline{\text{SCKF0}}$, P913
CSIF3 + HS	SOF3, SIF3, $\overline{\text{SCKF3}}$, P913

当将专用 flash 编程器连接到一个串行接口引脚，而该串行接口已连接到板上另外的设备时，必须注意避免信号冲突以及其它设备的故障。

(a) 信号冲突

当专用 flash 编程器（输出）连接一个已连接至另外设备（输出）上的串行接口引脚（输入）时，会发生信号冲突。为了避免信号冲突，应该隔离与其它设备的连接或将其它设备设置为输出高阻状态。

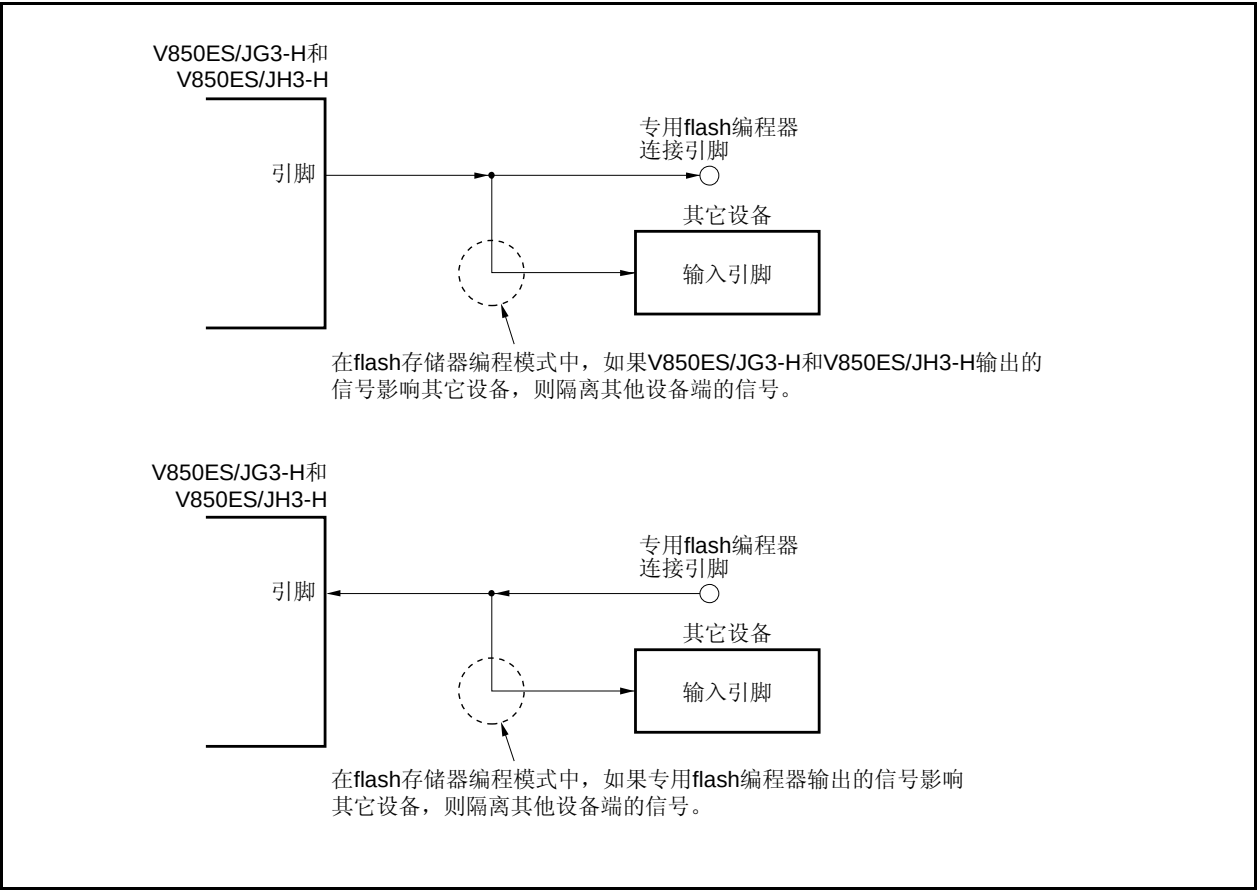
图 31-13. 信号冲突（串行接口输入引脚）



(b) 其它设备的故障

当将专用 flash 编程器（输出或输入）连接到与另外的设备（输入）连接的串行接口引脚（输入或输出）时，信号会输出到其它设备，并引起设备故障。为了避免这种情况发生，应隔离与其它设备的连接。

表 31-14. 其它设备的故障

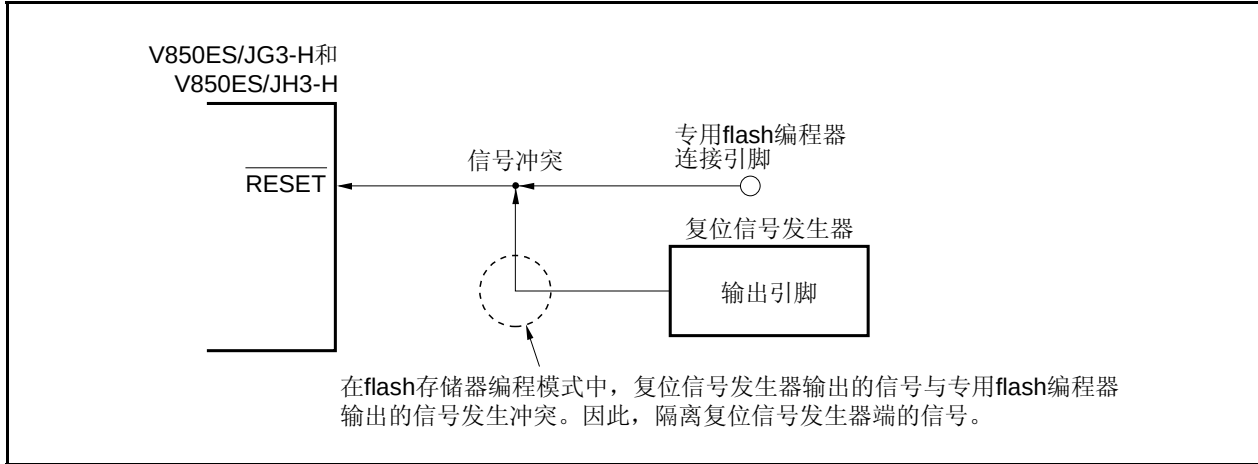


(4) $\overline{\text{RESET}}$ 引脚

当专用 flash 编程器的复位信号连接到 $\overline{\text{RESET}}$ 引脚，而该引脚已连接到板上的复位信号发生器时，就会发生信号冲突。为了避免信号冲突，应隔离与复位信号发生器的连接。

在 flash 存储器编程模式下，当从用户系统输入一个复位信号时，将不会正确执行编程操作。因此，除了来自于专用 flash 编程器的复位信号以外，不要输入其它信号。

图 31-15. 信号冲突 ($\overline{\text{RESET}}$ 引脚)

**(5) 端口引脚（包括 NMI）**

当系统转换到 flash 存储器编程模式时，所有不用于 flash 存储器编程的引脚处于与复位后瞬间状态相同的状态。如果连接各端口的外部设备不能识别复位后瞬间端口的状态，则需要对引脚进行适当地处理，比如通过一个电阻连接至 V_{DD} 或通过一个电阻连接至 V_{SS} 。

(6) 其它信号引脚

在与正常操作模式一样的状态中连接 X1、X2、XT1、XT2 和 REGC 引脚。

在 flash 存储器编程期间，向 $\overline{\text{DRST}}$ 引脚输入低电平或将其保持开路状态。禁止输入高电平。

(7) 电源供应

提供与正常操作模式中相同的电源 (V_{DD} 、 V_{SS} 、 EV_{DD} 、 UV_{DD} 、 AV_{REF0} 、 AV_{REF1} 、 AV_{SS})。

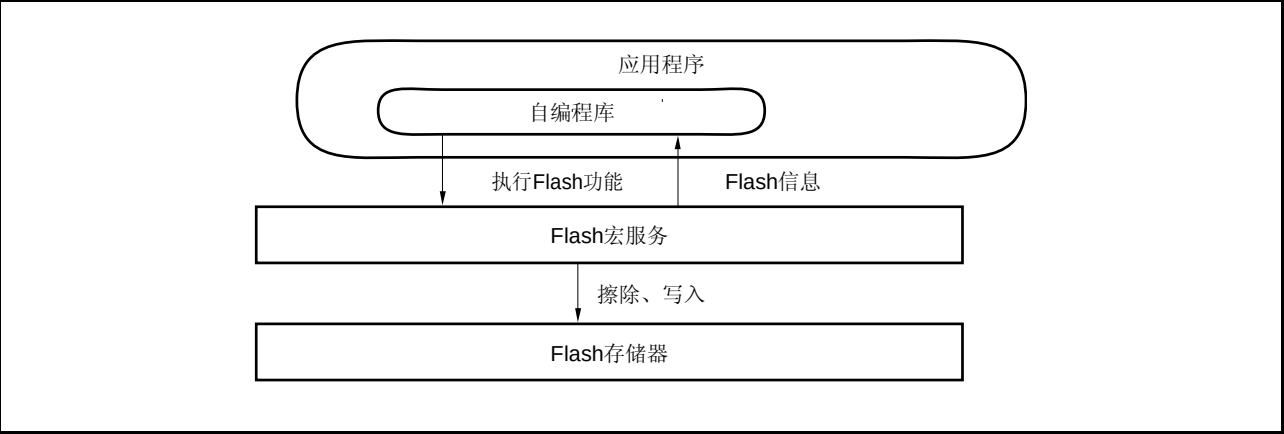
31.5 通过自编程重写

31.5.1 概述

V850ES/JG3-H 和 V850ES/JH3-H 支持 flash 宏服务，该服务允许用户程序自身重写内部 flash 存储器。通过使用此接口和一个使用用户应用程序来重写 flash 存储器的自编程库，flash 存储器可以通过预先传送到内部 RAM 或外部存储器中的用户应用程序进行重写。因此，在这个区域里可以更新用户程序，并且可以重写常量数据^注。

注 务必不要将程序代码分配至已分配了重写目标的常量块中。块配置参见 31.2 存储器配置。

图 31-16. 自编程的概念

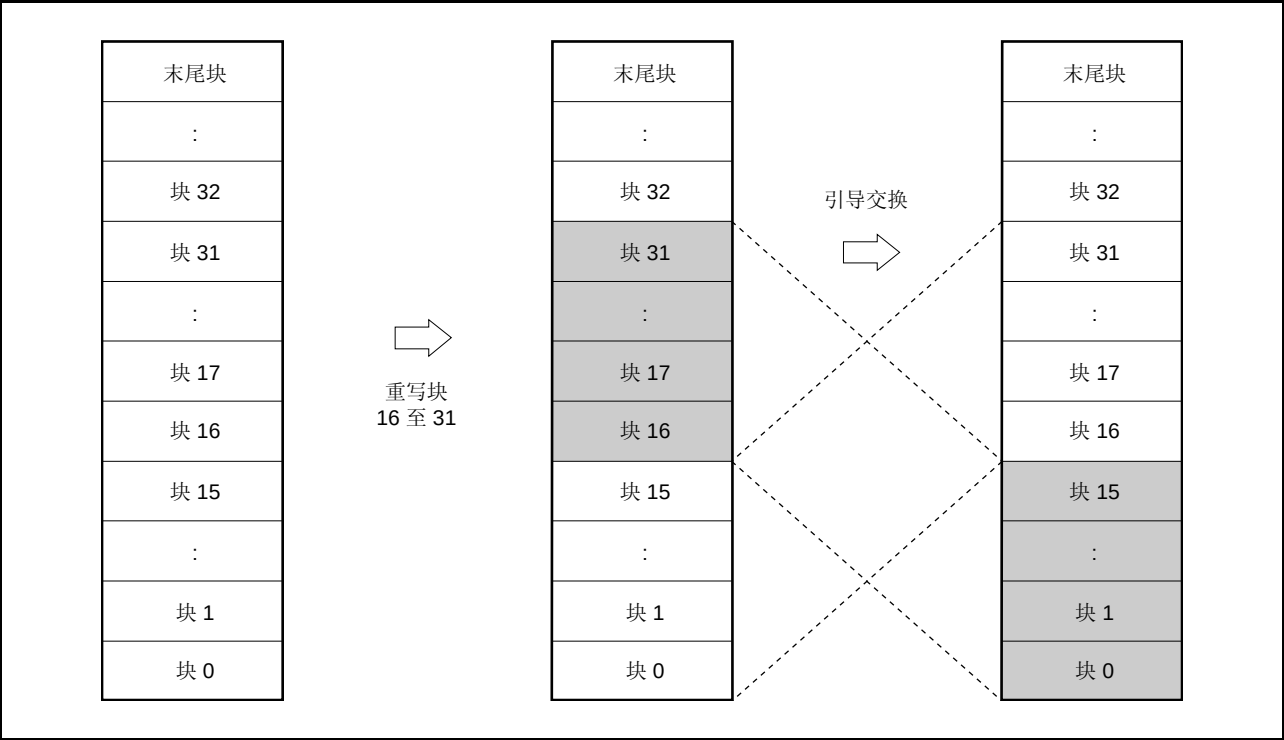


31.5.2 特性

(1) 自编程安全（引导交换功能）

V850ES/JG3-H 和 V850ES/JH3-H 单片机支持引导交换功能，该功能可以互换模块 0 至 15 的物理存储区和模块 16 至 31 的物理存储区。通过预先将用于重写的启动程序写入模块 16 至 31 然后交换物理存储区，那么即使在重写时发生掉电，整个存储区域仍能被安全重写，这是因为在模块 0 至 15 中总是存在正确的用户程序。

图 31-17. 重写整个存储区域（引导交换）



(2) 中断支持

自编程期间不能从 flash 存储器中获取指令。因此，即使发生中断也不能使用写入 flash 存储器的用户处理程序。

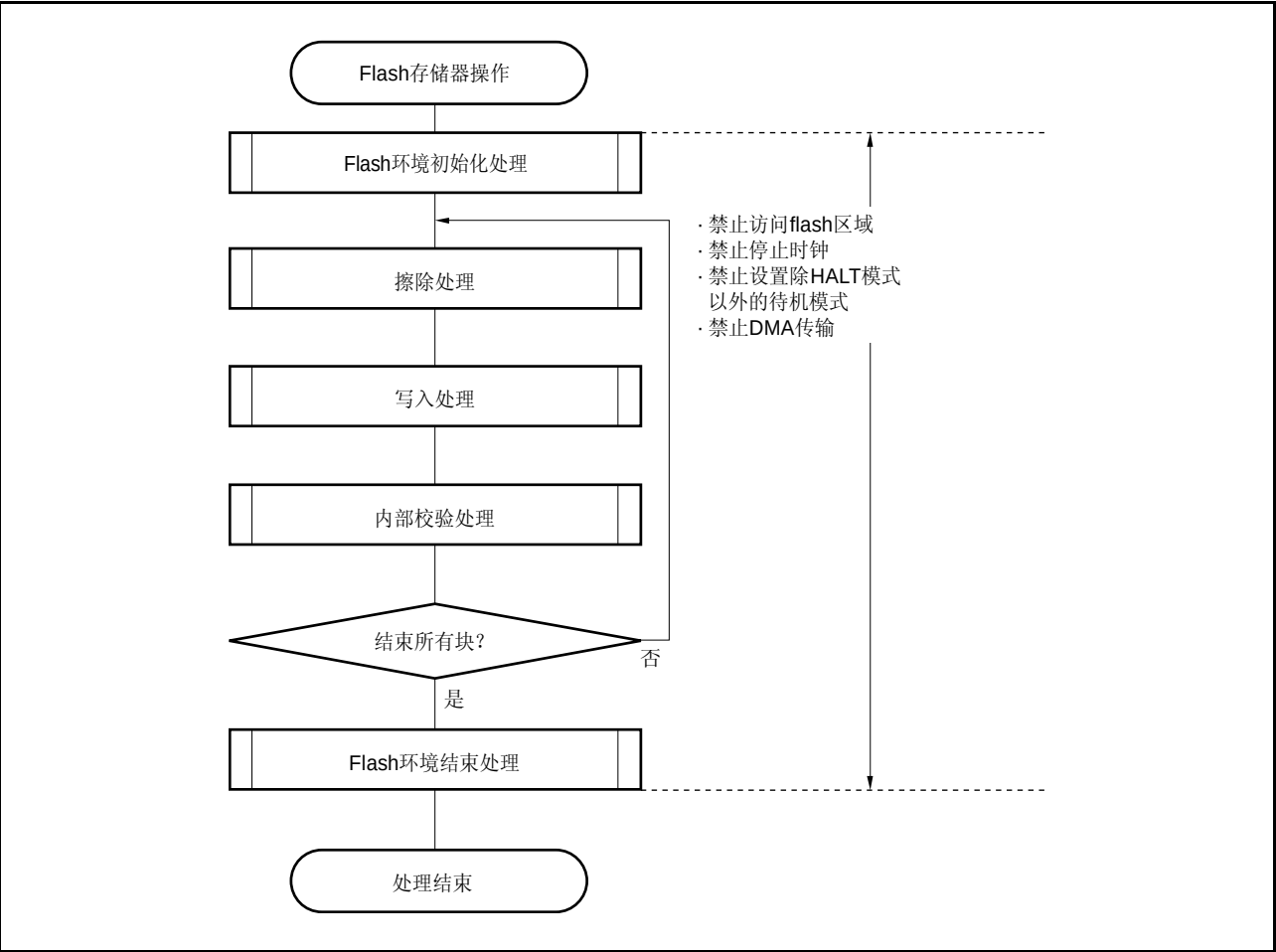
所以在 V850ES/JG3-H 和 V850ES/JH3-H 单片机中，要在自编程期间使用中断，处理移动到内部 RAM 中的特殊地址^注。分配跳转指令，从而将处理转移至内部 RAM 特殊地址^注处的用户中断服务。

注 NMI 中断： 内部 RAM 的起始地址
 可屏蔽中断： 内部 RAM 起始地址+4

31.5.3 标准自编程流程

使用 flash 自编程对 flash 存储器进行重写的整个处理过程说明如下。

图 31-18. 标准自编程流程



31.5.4 Flash功能

表 31-10. Flash 功能列表

功能名称	概述	支持
FlashInit	自编程库初始化	√
FlashEnv	Flash 环境开始/结束	√
FlashFLMDCheck	FLMD 引脚检查	√
FlashStatusCheck	硬件处理执行状态检查	√
FlashBlockErase	块擦除	√
FlashWordWrite	数据写入	√
FlashBlockIVerify	模块的内部校验	√
FlashBlockBlankCheck	模块的空白检查	√
FlashSetInfo	Flash 信息设置	√
FlashGetInfo	Flash 信息采集	√
FlashBootSwap	执行引导交换	√

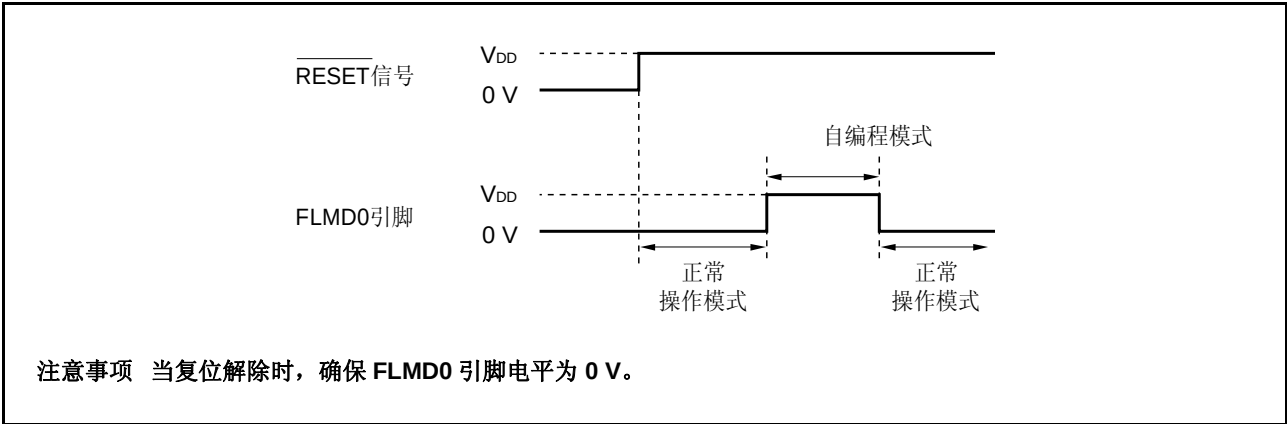
31.5.5 引脚处理

(1) FLMD0 引脚

当系统复位解除后，FLMD0 引脚用来设置操作模式并避免在自动重写期间写入 flash 存储器。因此当复位解除和执行正常工操作时，有必要保持 FLMD0 引脚电压为 0 V。重写存储器前，在自编程模式期间也有必要通过端口控制为 FLMD0 引脚提供值为 V_{DD} 的电压。

当自编程结束后，FLMD0 引脚的电压必须返回到 0 V。

图 31-19. 模式变化时序



31.5.6 使用的内部资源

下表所列为用于自编程的内部资源。这些内部资源除了自编程外还可以自由地用作其他用途。

表 31-11. 使用的内部资源

资源名称	描述
堆栈区	用户使用的堆栈的扩展被库使用（可以用于内部 RAM 和外部 RAM）。
库代码 ^注	库程序实体（除了处理的 flash 存储器模块以外，可用于任何其它地方）。
应用程序	作为用户应用程序执行。 调用 flash 函数。
可屏蔽中断	可以在用户应用执行状态或自编程状态。因为处理程序转移到内部 RAM 起始地址 +4 处，要在自编程状态下使用该中断，所以要预先在内部 RAM 起始地址 + 4 处分配一个跳转指令，将处理程序转移到用户中断服务程序。
NMI 中断	可以在用户应用执行状态或自编程状态。因为处理程序转移到内部 RAM 起始地址处，要在自编程状态下使用该中断功能，所以要预先在内部 RAM 起始地址处分配一个跳转指令，将处理程序转移到用户中断服务程序。

注 有关使用的资源，参见 **Flash 存储器自编程库用户手册**。

第三十二章 片上调试功能

V850ES/JG3-H 和 V850ES/JH3-H 的片上调试功能可以通过以下两种方法实现。

- 使用 DCU（调试控制单元）
在 V850ES/JG3-H 和 V850ES/JH3-H 中，由片上 DCU 使用 $\overline{\text{DRST}}$ 、DCK、DMS、DDI 和 DDO 引脚作为调试接口引脚来实现片上调试功能。
- 不使用 DCU
使用用户资源代替 DCU 由 MINICUBE2 或类似 MINICUBE2 来实现片上调试功能。

下表显示了两种片上调试功能的特性。

表 32-1. 片上调试功能特性

		使用 DCU 调试	不使用 DCU 调试
调试接口引脚		$\overline{\text{DRST}}$ 、DCK、DMS、DDI、DDO	• 使用 UARTC0 时 RXDC0、TXDC0 • 使用 CSIF0 时 SIF0、SOF0、$\overline{\text{SCKF0}}$、HS(P913) • 使用 CSIF3 时 SIF3、SOF3、$\overline{\text{SCKF3}}$、HS(P913)
用户资源保障		不需要	需要
硬件中断功能		2 点	2 点
软件中断功能	内部 ROM 区	4 点	4 点
	内部 RAM 区	2000 点	2000 点
实时 RAM 监控功能 ^{注1}		可用	可用
动态内存修改（DMM）功能 ^{注2}		可用	可用
屏蔽功能		$\overline{\text{Reset}}$ 、 $\overline{\text{NMI}}$ 、 $\overline{\text{INTWDT2}}$ 、 $\overline{\text{HLDRQ}}$ 、 $\overline{\text{WAIT}}$	$\overline{\text{RESET}}$ 引脚
ROM 安全功能		10-字节 ID 代码鉴别	10-字节 ID 代码鉴别
所用硬件		NINICUBE 等。	NINICUBE2 等。
跟踪功能		不支持。	不支持。
调试中断接口功能（DBINT）		不支持。	不支持。

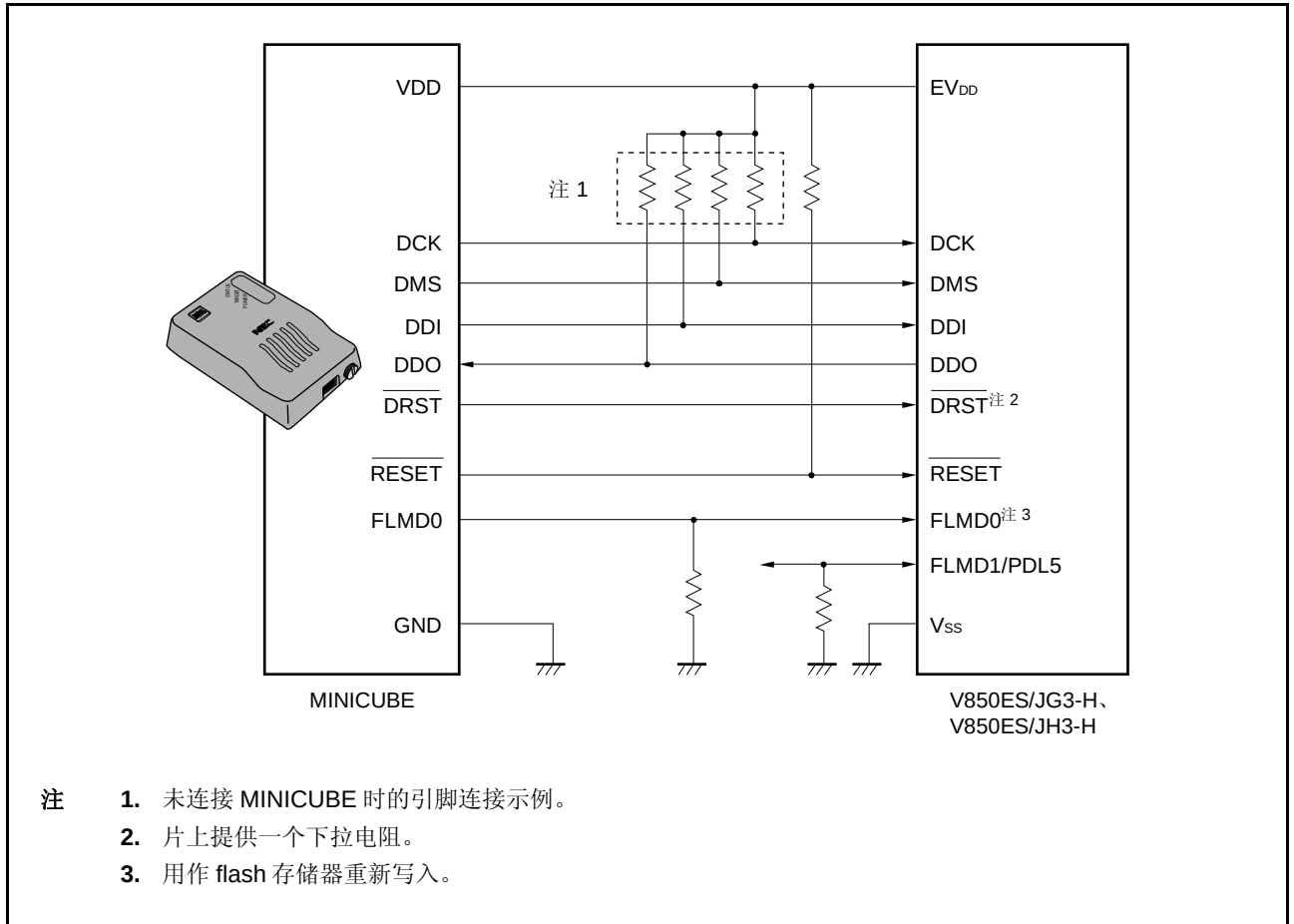
- 注
1. 这是在程序执行时读出存储器内容的功能。
 2. 这是在程序执行期间重写 RAM 内容的功能。

32.1 用DCU调试

使用调试接口引脚 ($\overline{\text{DRST}}$ 、DCK、DMS、DDI 和 DDO) 连接片上调试仿真器 (MINICUBE) 来调试程序。

32.1.1 连接电路示例

图 32-1. 调试接口引脚用作通信接口时的电路连接示例



32.1.2 接口信号

接口信号说明如下。

(1) $\overline{\text{DRST}}$

它是片上调试单元的复位输入信号。它是负逻辑信号，用来异步初始化调试控制单元。

集成调试器启动后，当检测到目标系统的 VDD 时，MINICUBE 使 $\overline{\text{DRST}}$ 上升，并启动设备的片上调试单元。

当 $\overline{\text{DRST}}$ 信号升高时，也会在 CPU 中生成一个复位信号。

通过启动集成调试器来开始调试时，总生成一个 CPU 复位。

(2) DCK

它是时钟输入信号。它由 MINICUBE 提供 20 MHz 或 10 MHz 的时钟。在片上调试单元，在 DCK 信号的上升沿对 DMS 和 DDI 信号进行采样，而在其下降沿输出数据 DDO。

(3) DMS

它是传输模式选择信号。调试单元的传输状态随 DMS 信号的电平变化。

(4) DDI

这是数据输入信号。在片上调试单元内，在 DCK 的上升沿对其进行采样。

(5) DDO

这是数据输出信号。在 DCK 信号的下降沿从片上调试单元输出。

(6) EV_{DD}

该信号用于检测目标系统的 VDD。如果没有检测到目标系统的 VDD，则从 MINICUBE 输出的信号 ($\overline{\text{DRST}}$ 、DCK、DMS、DDI、FLMD0 和 $\overline{\text{RESET}}$) 将进入到高阻状态。

(7) FLMD0

flash 自编程功能用作通过集成调试器把数据下载到 flash 存储器。在 flash 自编程期间，FLMD0 引脚必须保持高位。此外，在 FLMD0 引脚接一个下拉电阻。

FLMD0 引脚可通过下面两种方式中的任意一种进行控制。

<1> 由 MINICUBE 控制

将 MINICUBE 的 FLMD0 信号连接到 FLMD0 引脚。

在正常模式下，MINICUBE 不做任何驱动（高阻）。

中断期间，当执行集成调试器的下载功能时 MINICUBE 将 FLMD0 引脚上升为高电平。

<2> 由端口控制：

将设备的任一端口连接到 FLMD0 引脚。

可能会用到由用户程序使用来实现 flash 自编程功能的同一个端口。

在集成调试器的控制台上，执行下载功能前进行设置使得端口引脚上升为高电平，或执行下载功能后降低端口功能引脚电平。

关于细节，参见 ID850QB Ver.3.40 集成调试器操作用户手册（U18604E）。

(8) $\overline{\text{RESET}}$

它是系统复位输入引脚。如果通过用户程序设置 OCDM 寄存器 OCDM1 位的值使得 $\overline{\text{DRST}}$ 引脚无效，则不能执行片上调试。因此，使用 $\overline{\text{RESET}}$ 引脚，由 MINICUBE 实现复位，从而使 $\overline{\text{DRST}}$ 引脚有效（初始化）。

32.1.3 可屏蔽功能

可以屏蔽 Reset、NMI、INTWDT2、 $\overline{\text{WAIT}}$ 和 $\overline{\text{HLDRQ}}$ 信号。
调试器（ID850QB）的可屏蔽功能和相应的 V850ES/JG3-H 和 V850ES/JH3-H 的功能列举如下。

表 32-2. 可屏蔽功能

ID850QB 的可屏蔽功能	相应的 V850ES/JG3-H 和 V850ES/JH3-H 功能
NMI0	NMI 引脚输入
NMI2	生成不可屏蔽中断请求信号（INTWDT2）
STOP	—
HOLD	$\overline{\text{HLDRQ}}$ 引脚输入
RESET	由 $\overline{\text{RESET}}$ 引脚输入、低电压检测器、时钟监控器或看门狗定时器（WDT2）上溢出生成复位信号
WAIT	$\overline{\text{WAIT}}$ 引脚输入

32.1.4 寄存器

(1) 片上调试模式寄存器（OCDM）（仅适用于 V850ES/JG3-H）

OCDM 寄存器用来选择正常工作模式或片上调试模式。该寄存器是一个专用寄存器，且只能在特定序列组合中写入（参见 3.4.8 特殊寄存器）。

该寄存器还用来指定片上调试功能中给出的引脚是用作片上调试引脚，还是用作普通端口/外设功能引脚。它也可用来断开 P56/INTP05/ $\overline{\text{DRST}}$ 引脚的内部下拉电阻。

只有在 $\overline{\text{DRST}}$ 引脚输入低电平时 OCDM 寄存器才能够写入。

该寄存器可按 8 位或 1 位为单元进行读写。

复位后：01H^注 R/W 地址： FFFFF9FCH

	7	6	5	4	3	2	1	<0>
OCDM	0	0	0	0	0	0	0	OCDM0

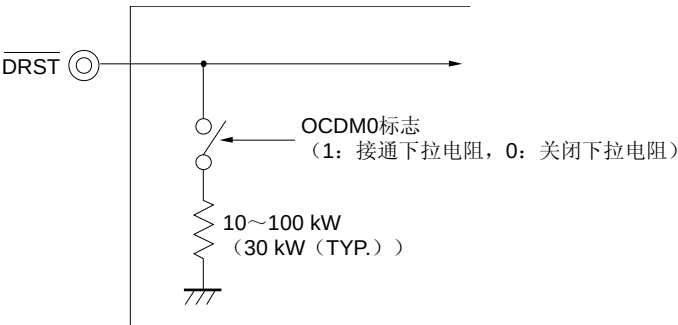
OCDM0	操作模式
0	选择正常操作模式（复用为片上调试功能引脚的引脚用作端口/外设功能引脚）和断开P56/INTP05/DRST引脚的片上下拉电阻。
1	当 $\overline{\text{DRST}}$ 引脚为低电平时： 正常操作模式（复用为片上调试功能引脚的引脚用作端口/外设功能引脚） 当 $\overline{\text{DRST}}$ 引脚为高电平时： 片上调试模式（复用为片上调试功能引脚的引脚用作片上调试模式引脚）

注 $\overline{\text{RESET}}$ 输入将寄存器设为 01H。但是，由 WDT2RES 信号、时钟监控器（CLM）或低电压检测器（LVI）复位后，仍保持 OCDM 寄存器的值。

注意事项 1. 外部复位后，当不将 DDI、DDO、DCK 以及 DMS 引脚用作片上调试引脚而用作端口引脚时，必须执行下列任一操作。

- 给 P56/INTP05/ $\overline{\text{DRST}}$ 引脚输入一个低电平。
- 设置 OCDM0 位。在这种情况下，进行以下列操作。
 - <1> 将 OCDM0 位清除为 0。
 - <2> 在<1>未完成之前将 P56/INTP05/ $\overline{\text{DRST}}$ 引脚固定为低电平。

2. $\overline{\text{DRST}}$ 引脚有一个片上下拉电阻。当 OCDM0 标志置为 0 时断开该电阻。



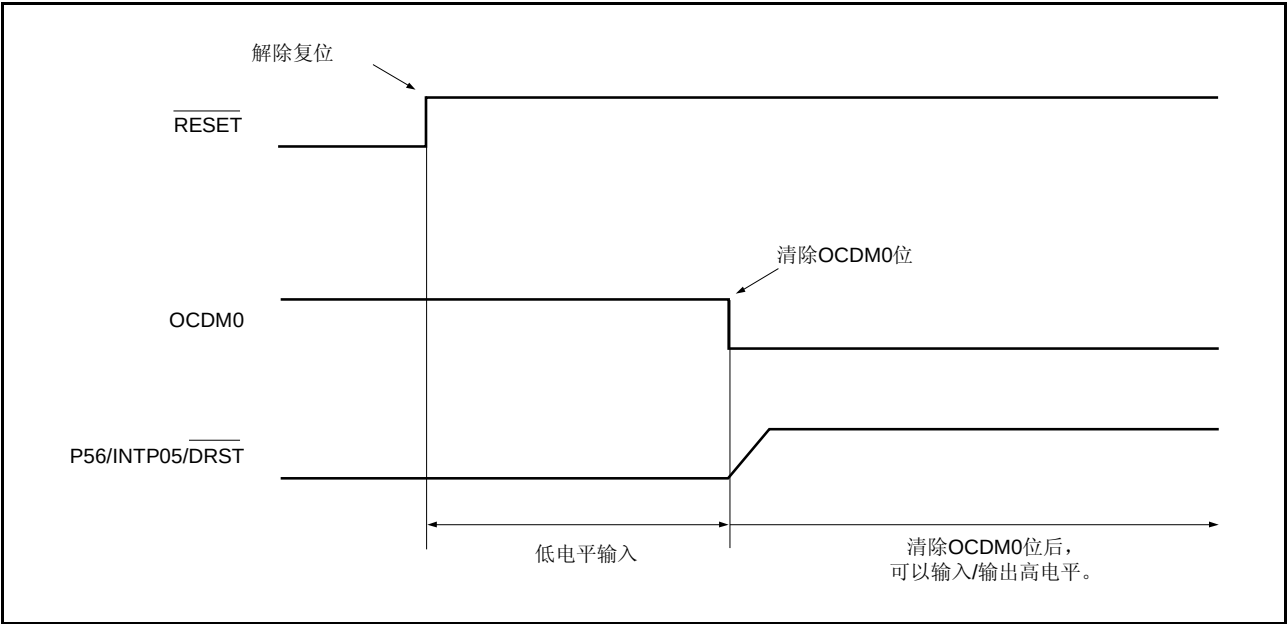
32.1.5 操作

在下表中的条件下片上调试功能无效。
当不使用此功能时，在 OCDM.OCDM0 标志清除为 0 之前一直将 $\overline{\text{DRST}}$ 引脚保持为低电平。

OCDM0 标志 $\overline{\text{DRST}}$ 引脚	0	1
L	无效	无效
H	无效	有效

备注 L: 低电平输入
 H: 高电平输入

图 32-2. 不使用片上调试功能时的时序



32.1.6 注意事项

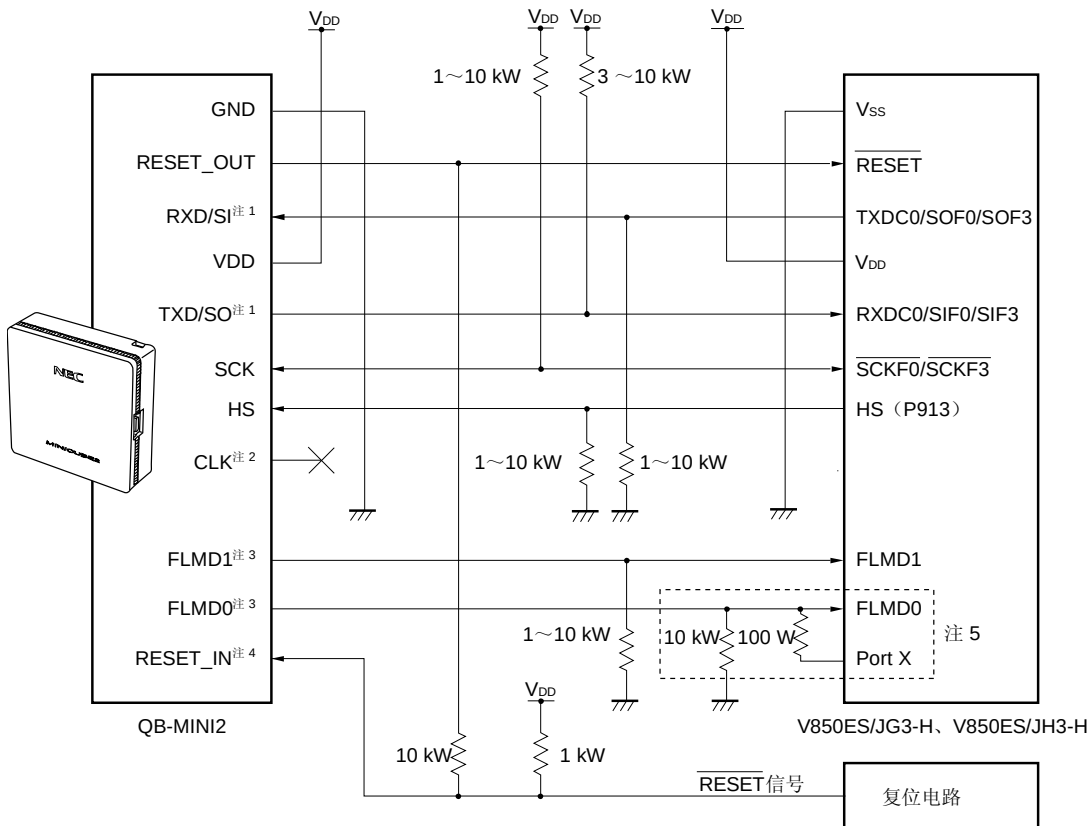
- (1) 如果在 RUN（程序执行）期间输入一个复位信号（来自目标系统或来自内部复位源的复位信号），则中断功能可能会出现误操作。
- (2) 即使由屏蔽功能屏蔽复位信号，如果由引脚输入一个复位信号也可能使 I/O 缓冲器（端口引脚）复位。
- (3) 屏蔽中断期间引脚复位而 CPU 和外设 I/O 不复位。用户程序正在执行中时，如果 flash 存储器由 DMM 重写或由 RAM 监控功能读取，就产生引脚复位或内部复位，那么 CPU 和外设 I/O 可能不能正确复位。
- (4) 在片上调试模式中，将 DDO 引脚强制设置为高电平输出。

32.2 不使用DCU调试

下面说明如何不用 DCU 来实现片上调试功能。即使用 MINICUBE2 的 UARTC0 (RXDC0 和 TXDC0) 引脚或 CSIF0 (SIF0、SOF0、 $\overline{\text{SCKF0}}$ 和 HS (P913)) 引脚或 CSIF3 (SIF3、SOF3、 $\overline{\text{SCKF3}}$ 和 HS (P913)) 引脚作为调试接口。

32.2.1 电路连接举例

图 32-3. 当使用 UARTC0/CSIF0/CSIF3 作为通信接口时的电路连接举例



- 注
1. 将 V850ES/JG3-H 或 V850ES/JH3-H 单片机的 TXDC0/SOF0/SOF3 (发送端) 连接到目标连接器的 RXD/SI (接收端)，并将目标连接器的 TXD/SO (发送端) 连接到 V850ES/JG3-H 或 V850ES/JH3-H 单片机的 RXDC0/SIF0/SIF3 (接收端)。
 2. 在 flash 编程期间，该引脚可能用来由 MINICUBE2 提供一个时钟。关于细节，参见第三十一章 FLASH 存储器。
 3. 除了当存储器在调试中的中断期间写入的情况，连接到该引脚 (FLMD0, FLMD1) 的 V850ES/JG3-H 或 V850ES/JH3-H 端的引脚可用作复用功能引脚，因为该引脚处于高阻状态。
 4. 本连接是在假设 $\overline{\text{RESET}}$ 信号由 N 沟道漏极开路缓冲输出的情况下 (输出电阻: 100 Ω 或更小) 设计的。
 5. 虚线内的电路是为 flash 自编程设计的，该部分电路通过端口控制 FLMD0 引脚。使用端口输入或输出高电平。当不执行 flash 自编程时，接到 FLMD0 引脚的下拉电阻可以在 1 至 10 k Ω 之间。

备注

当 UARTC0、CSIF0 或 CSIF3 用作通信接口时所用的引脚，参见表 32-3。

表 32-3. V850ES/JG3-H 和 MINICUBE2 间的接线

MINICUBE2 (QB-MINI2) 的引脚配置			使用 CSIF0-HS		使用 CSIF3-HS		使用 UARTC0	
信号名称	I/O	引脚功能	引脚名称	引脚编号	引脚名称	引脚编号	引脚名称	引脚编号
SI/RxD	输入	接收来自 V850ES/JG3-H 的命令和数据的引脚	P41/SOF0	23	P911/SOF3	53	P30/TXDC0	25
SO/TxD	输出	发送命令和数据至 V850ES/JG3-H 的引脚	P40/SIF0	22	P910/SIF3	52	P31/RXDC0	26
SCK	输出	3-线串行通信的时钟输出引脚	P42/SCKF0	24	P912/SCKF3	54	不需要	–
CLK ^注	输出	至 V850ES/JG3-H 的时钟输出引脚	不需要 ^注	–	不需要 ^注	–	不需要 ^注	–
RESET_OUT	输出	到 V850ES/JG3-H 的复位输出引脚	RESET	14	RESET	14	RESET	14
FLMD0	输出	将 V850ES/JG3-H 设置为调试模式或编程模式的输出引脚	FLMD0	8	FLMD0	8	FLMD0	8
FLMD1	输出	设置编程模式的输出引脚	PDL5/FLMD1	76	PDL5/FLMD1	76	PDL5/FLMD1	76
HS	输入	CSIO + HS 通信的握手信号	P913	85	P913	85	不需要	–
GND	–	地	V _{SS}	11, 33, 62, 79	V _{SS}	11, 33, 62, 79	V _{SS}	11, 33, 62, 79
			AV _{SS}	2	AV _{SS}	2	AV _{SS}	2
RESET_IN	输入	目标系统的复位输入引脚						

注 它用作 MINICUBE2 flash 编程器的时钟输出。关于细节，参见第三十一章 FLASH 存储器。

表 32-4. V850ES/JH3-H 和 MINICUBE2 间的接线

MINICUBE2 (QB-MINI2) 的引脚配置			使用 CSIF0-HS		使用 CSIF3-HS		使用 UARTC0	
信号名称	I/O	引脚功能	引脚名称	引脚编号	引脚名称	引脚编号	引脚名称	引脚编号
SI/RXD	输入	接收来自 V850ES/JH3-H 的命令和数据的引脚	P41/SOF0	30	P911/SOF3	67	P30/TXDC0	37
SO/TXD	输出	发送命令和数据至 V850ES/JH3-H 的引脚	P40/SIF0	29	P910/SIF3	66	P31/RXDC0	31
SCK	输出	3-线串行通信的时钟输出引脚	P42/SCKF0	31	P912/SCKF3	68	Not needed	–
CLK ^注	输出	至 V850ES/JH3-H 的时钟输出引脚	不需要 ^注	–	不需要 ^注	–	不需要 ^注	–
RESET_OUT	输出	至 V850ES/JH3-H 的复位输出引脚	RESET	18	RESET	18	RESET	18
FLMD0	输出	将 V850ES/JH3-H 设置为调试模式或编程模式的输出引脚	FLMD0	12	FLMD0	12	FLMD0	12
FLMD1	输出	设置编程模式的输出引脚	PDL5/FLMD1	103	PDL5/FLMD1	103	PDL5/FLMD1	103
HS	输入	CSI0 + HS 通信的握手信号	P913	69	P913	69	不需要	–
GND	–	接地	V _{SS}	15, 45, 84, 106	V _{SS}	15, 45, 84, 106	V _{SS}	15, 45, 84, 106
			AV _{SS}	2	AV _{SS}	2	AV _{SS}	2
RESET_IN	输入	目标系统的复位输入引脚						

注 它用作 MINICUBE2 flash 编程器的时钟输出。关于细节，参见第三十一章 FLASH 存储器。

32.2.2 可屏蔽功能

只能屏蔽复位信号。
在调试器（ID850QB）中可以屏蔽的功能和相应的 V850ES/JG3-H 和 V850ES/JH3-H 的功能如下表所示。

表 32-5. 可屏蔽功能

在 ID850QB 中的可屏蔽功能	相应的 V850ES/JG3-H 和 V850ES/JH3-H 的功能
NMI0	—
NMI1	—
NMI2	—
STOP	—
HOLD	—
RESET	由 RESET 引脚输入生成复位信号
WAIT	—

32.2.3 用户资源安全性

用户必须做以下准备来实现 MINICUBE2 和目标器件间的通信功能，并完成各调试功能。这些项目需要在用户程序中设置或使用编译器选项设置。

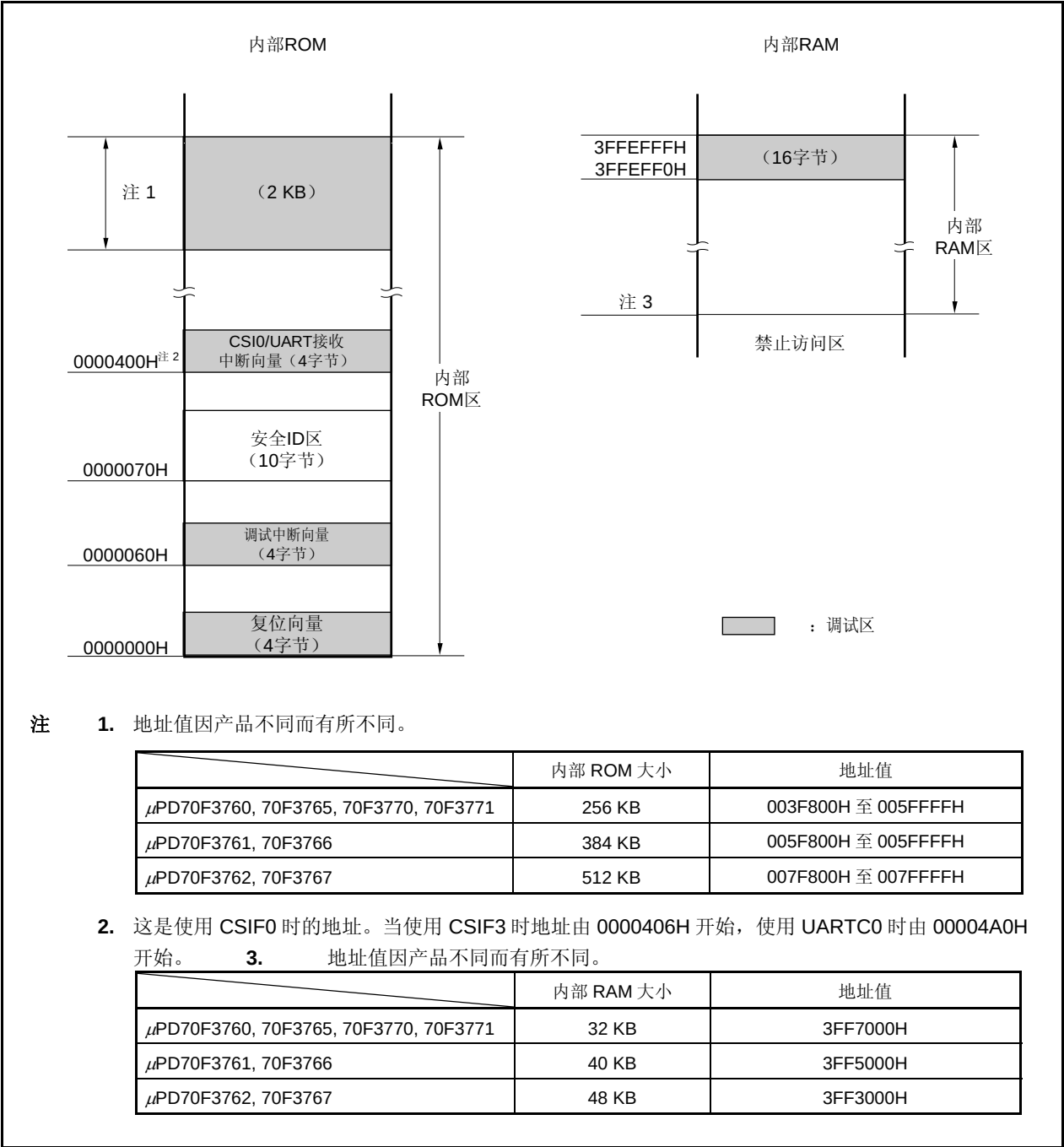
(1) 存储空间的安全性

图 32-4 中的阴影部分是用来放置调试监控程序的预留区，因此不能在这些空间分配用户程序和数据。必须保护这些空间不被用户程序使用。

(2) 安全 ID 设置

ID 代码必须嵌入图 32-4 中 0000070H 至 0000079H 间的区域，以避免存储内容被未授权者读取。关于细节参见 **32.3 ROM 安全功能**。

图 32-4. 分配调试监控程序的存储空间



(3) 复位向量

复位向量包括用于调试监控程序的跳转指令。

[如何保护区域]

没有必要特意保护该区域。当下载一个程序时，调试器对应于下列情况重写复位向量。若重写的模式与下列情况不一致，则调试器生成错误（使用 ID850QB 时为 F0C34）。

(a) 从 0 地址开始连续放置两个空指令时

重写前		重写后
0x0 nop	→	跳转到 0x0 处的调试监控程序
0x2 nop		0x4 xxxx
0x4 xxxx		

(b) 从 0 地址开始连续放置两个 0xFFFF 时（已擦除设备）

重写前		重写后
0x0 0xFFFF	→	跳转到 0x0 处的调试监控程序
0x2 0xFFFF		0x4 xxxx
0x4 xxxx		

(c) 在 0 地址处放置 *jr* 指令（使用 CA850 时）

重写前		重写后
0x0 jr disp22	→	跳转到 0x0 处的调试监控程序
		0x4 jr disp22 - 4

(d) 从 0 地址开始连续放置 *mov32* 和 *jmp*（使用 IAR 编译器 ICCV850 时）

重写前		重写后
0x0 mov imm32,reg1	→	跳转到 0x0 处的调试监控程序
0x6 jmp [reg1]		0x4 mov imm32,reg1
		0xa jmp [reg1]

(e) 在 0 地址处放置用于调试监控程序的跳转指令

重写前		重写后
跳转到 0x0 的处调试监控程序	→	不改变

(4) 调试监控程序区域的安全

图 32-4 中的阴影部分是分配调试监控程序的区域。监控程序执行初始化处理，来进行通信接口调试和 RUN 或 CPU 的中断处理。内部 ROM 区必须填充 0xFF。该区域禁止被用户程序重写。

[如何保护区域]

若用户程序不使用该区域则没有必要保护该区。

然而，为了避免调试器启动期间可能出现的问题，建议预先使用编译器保护该区域。

下面给出保护区域的范例，它们使用 NEC 的电子编译器 CA850。如下所示，添加汇编源文件和连接指令代码。

- 汇编源文件（将下列代码添加为汇编源文件。）

```
-- Secures 2 KB space for monitor ROM section
.section "MonitorROM", const
.space    0x800, 0xff

-- Secures interrupt vector for debugging
.section "DBG0"
.space   4, 0xff

-- Secures interrupt vector for serial communication
-- Change the section name according to the serial communication mode used
.section "INTCF0R"
.space   4, 0xff

-- Secures 16-byte space for monitor RAM section
.section "MonitorRAM", bss
.lcomm  monitorramsym, 16, 4    -- defines symbol monitorramsym
```

- 连接指令（将下列代码添加到连接伪指令文件。）

下面给出了一个内部 ROM 为 512 KB（末地址为 007FFFFH）且内部 RAM 为 56 KB（末地址为 3FFEFFFH）的例子。

```
MROMSEG      : !LOAD ?R V0x07f800{
               MonitorROM      = $PROGBITS    ?A MonitorROM;
};

MRAMSEG       : !LOAD ?RW V0x03ffef0{
               MonitorRAM      = $NOBITS      ?AW MonitorRAM;
};
```

(5) 通信串行接口的安全性

UARTC0、CSIF0 或 CSIF3 用于 MINICUBE2 和目标系统间的通信。与串行接口模式相关的设置通过调试监控程序来完成，但如果设置被用户程序改变，则会发生通信错误。
为了防止发生这样的问题，必须在用户程序中对通信串行接口进行保护。

[如何保护通信串行接口]

- 片上调试模式寄存器（OCDM）
使用 UARTC0、CSIF0 或 CSIF3 的片上调试功能时，将 OCDM 寄存器功能设置为正常模式。务必按照下面进行设置。
 - P05/INTP02/ $\overline{\text{DRST}}$ 引脚输入低电平。
 - 将 OCDM0 位设置如下。
 - <1> 将 OCDM0 位清除为 0。
 - <2> 在<1>完成之前将 P05/INTP02/ $\overline{\text{DRST}}$ 引脚输入固定为低电平。
- 串行接口寄存器
在用户程序中不要设置与 CSIF0、CSIF3 或 UARTC0 相关的寄存器。
- 中断屏蔽寄存器
使用 CSIF0 时，不要屏蔽传送结束中断（INTCF0R）。使用 CSIF3 时，不要屏蔽传送结束中断（INTCF3R）。使用 UARTC0 时，不要屏蔽接收完成中断（INTUC0R）。

(a) 使用 CSIF0 时

	7	6	5	4	3	2	1	0
CF0RIC	x	0	x	x	x	x	x	x

(b) 使用 CSIF3 时

	7	6	5	4	3	2	1	0
CF3RIC	x	0	x	x	x	x	x	x

(c) 使用 UARTC0 时

	7	6	5	4	3	2	1	0
UC0RIC	x	0	x	x	x	x	x	x

备注

x: 忽略

• 使用 UARTC0 时的端口寄存器

使用 UARTC0 时，通过调试监控程序设置端口寄存器，从而使 TXDC0 和 RXDC0 引脚有效。在调试期间不要由用户程序改变下列寄存器的设置。（同样的值可被覆盖写入。）

	7	6	5	4	3	2	1	0
PFC3	x	x	x	x	x	x	0	0
	7	6	5	4	3	2	1	0
PFCE3	x	x	x	x	x	x	0	0
	7	6	5	4	3	2	1	0
PMC3	x	x	x	x	x	x	1	1

备注 x: 忽略

• 使用 CSIF0 时的端口寄存器

使用 CSIF0 时，通过调试监控程序设置端口寄存器，以使 SIF0、SOF0、 $\overline{\text{SCKF0}}$ 和 HS（P913）引脚生效。在调试期间不要由用户程序改变下列寄存器的设置。（同样的值可被覆盖写入。）

（a）SIF0、SOF0 和 $\overline{\text{SCKF0}}$ 的设置								
	7	6	5	4	3	2	1	0
PMC4	x	x	x	x	x	1	1	1
	7	6	5	4	3	2	1	0
PFC4	x	x	x	x	x	x	0	0
	7	6	5	4	3	2	1	0
PFCE4	x	x	x	x	x	x	0	0
（b）HS（P913 引脚）设置								
	15	14	13	12	11	10	9	8
PM9H	x	x	0	x	x	x	x	x
	15	14	13	12	11	10	9	8
P9H	x	x	注	x	x	x	x	x

注 禁止写入该位。

根据调试器的状态由监控程序改变与 HS 引脚对应的端口值。为了按照 8 位为单位执行端口寄存器设置，用户程序通常可利用读取—修改—写入方法。然而，如果在写入之前发生一个调试中断，就可能会执行一个意外的操作。

备注 x: 忽略

- 使用 CSIF3 的端口寄存器
使用 CSIF3 用于通信时，通过调试监控程序设置端口寄存器，以使 SIF3、SOF3、 $\overline{\text{SCKF3}}$ 和 HS（P913）引脚有效。在调试期间不要由用户程序改变下列寄存器的设置。（同样的值可被覆盖写入。）

(a) SIF3、SOF3 和 $\overline{\text{SCKF3}}$ 设置

	15	14	13	12	11	10	9	8
PMC9H	x	x	x	1	1	1	x	x

	15	14	13	12	11	10	9	8
PFC9H	x	x	x	x	0	0	x	x

	15	14	13	12	11	10	9	8
PFCE9H	x	x	x	x	0	0	x	x

(b) HS（P913 引脚）设置

	15	14	13	12	11	10	9	8
PM9H	x	x	0	x	x	x	x	x

	15	14	13	12	11	10	9	8
P9H	x	x	注	x	x	x	x	x

注 禁止写入该位。
根据调试器的状态由监控程序改变与 HS 引脚对应的端口值。为了按照 8 位为单位执行端口寄存器设置，用户程序通常可利用读取—修改—写入方法。然而，如果在写入之前发生一个调试中断，就可能会执行一个意外的操作。

备注 x：忽略

32.2.4 注意事项

(1) 用于调试的器件的处理

不要将用于调试的器件安装在一个批量生产的产品上，因为在调试期间 flash 存储器被重写过，故不能保证 flash 存储器的重写次数。此外，不要将调试监控程序嵌入到批量生产的产品中。

(2) 当中断不能执行时

如果满足下列条件之一，则不能执行强制中断。

- 禁止中断（DI）
- 串行接口发生的中断被屏蔽，该中断用于 MINICUBE2 和目标器件之间的通信。
- 禁止由可屏蔽中断来解除待机时进入待机模式。
- 用于 MINICUBE2 和目标器件间通信的模式为 UARTC0，并且主时钟已经停止。

(3) 当不执行伪实时监控（RRM）功能和 DMM 功能时

如果满足下列条件之一将不运行伪 RRM 功能和 DMM 功能。

- 禁止中断（DI）
- 串行接口发生的中断被屏蔽，该中断用于 MINICUBE2 和目标器件之间的通信。
- 禁止由可屏蔽中断解除待机时进入待机模式。
- MINICUBE2 和目标器件间的通信模式为 UARTC0，且主时钟已经停止。
- MINICUBE2 和目标器件之间的通信模式为 UARTC0，并且用来通信的时钟不同于在调试器中指定的时钟。

(4) 允许由伪 RRM 和 DMM 功能解除待机模式

如果满足下列条件之一则可由伪 RRM 功能和 DMM 功能解除待机模式。

- MINICUBE2 和目标器件之间的通信模式为 CSIF0 或 CSIF3。
- MINICUBE2 和目标器件的通信模式为 UARTC0，且已供应主时钟。

(5) 使用 DMM 功能重新写入需要特定序列的外设 I/O 寄存器。

需要特定序列的外设 I/O 寄存器不能用 DMM 功能重新写入。

(6) Flash 自编程

如果分配给调试监控程序的空间被 flash 自编程重写，那么调试功能将不能再正常运行。

32.3 ROM安全功能

32.3.1 安全ID

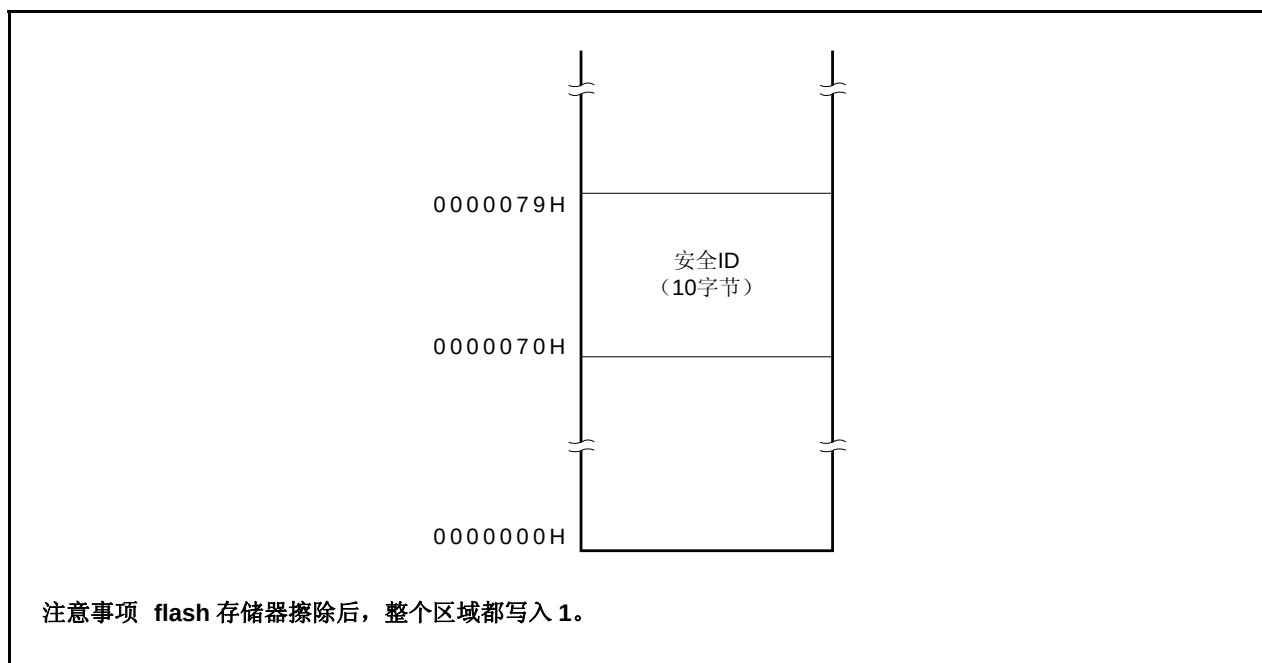
V850ES/JG3-H 和 V850ES/JH3-H 的 flash 存储器版本执行鉴别，使用一个 10 字节的 ID 代码，避免在片上调试仿真器进行片上调试时由未经授权者读取 flash 存储器的内容。

设置片上 flash 存储区 0000070H 至 0000079H 中的 10 字节 ID 代码，允许调试器执行 ID 鉴别。

如果 ID 匹配，则安全性解除且允许读取 flash 存储器和使用片上调试仿真器。

- 在 0000070H 至 0000079H 之间设置 10 字节的 ID 代码。
- 0000079H 的第 7 位为片上调试仿真器的使能标志。
(0: 禁止, 1: 允许)
- 当启动片上调试仿真器时，调试器需要输入 ID。当调试器输入的 ID 与在 0000070H 至 0000079H 间设置的 ID 代码匹配时，启动调试器。
- 若片上调试仿真器的使能标志为 0，即使 ID 码匹配，也不能进行调试。

图 32-5. 安全 ID 区



32.3.2 设置

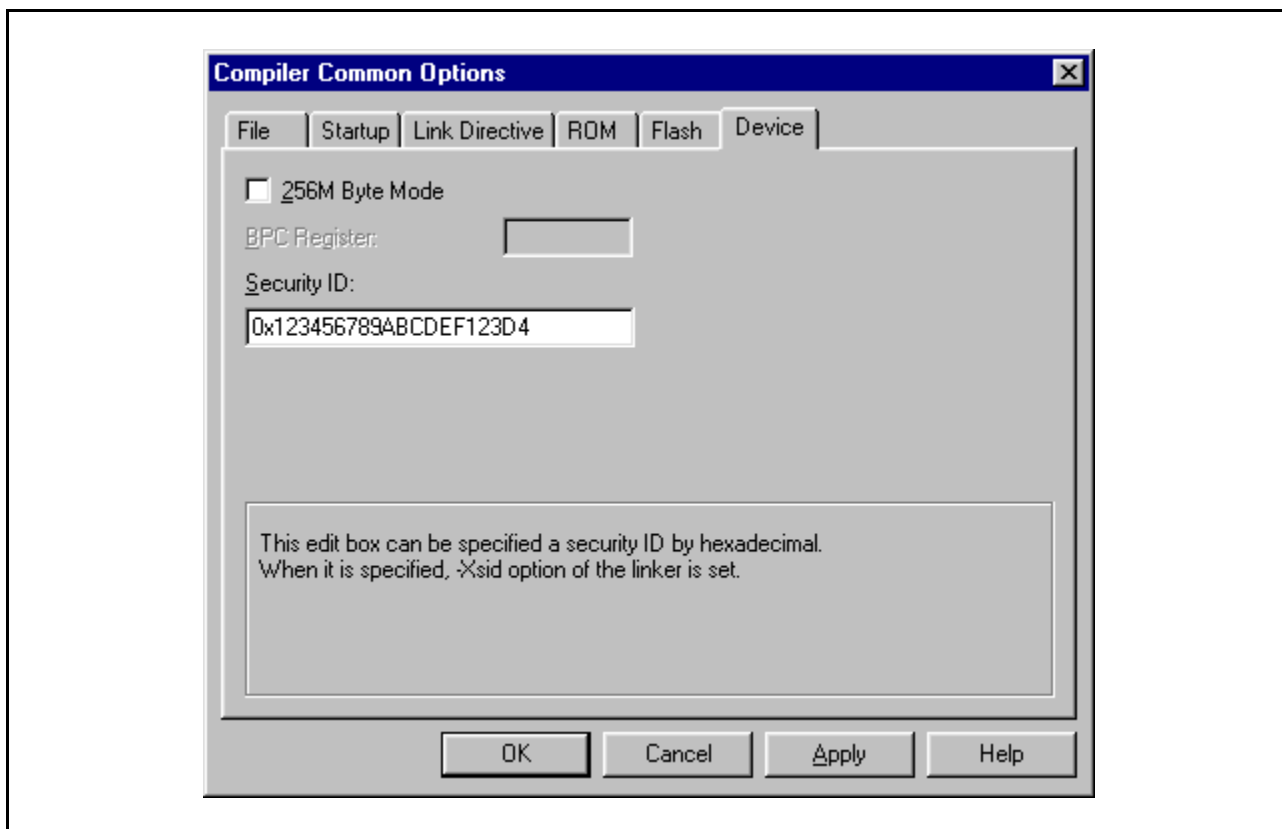
下面显示了如何设置 ID 代码，如表 32-6 所示。

当 ID 代码按照表 32-6 进行设置时，在 ID850QB 配置对话框中输入的 ID 码为“123456789ABCDEF123D4”（ID 码不区分大小写）。

表 32-6. ID 码

地址	值
0x70	0x12
0x71	0x34
0x72	0x56
0x73	0x78
0x74	0x9A
0x75	0xBC
0x76	0xDE
0x77	0xF1
0x78	0x23
0x79	0xD4

可对支持 CA850 Ver. 3.10（或更高版本）的设备文件和使用 PM+编译器的通用选项设置的安全 ID 指定 ID 码。



[程序举例（使用 CA850 Ver. 3.10 或更高版本时）]

```
#-----  
#      SECURITYID  
#-----  
      .section      "SECURITY_ID"      --Interrupt handler address 0x70  
      .word          0x78563412        --0-3 byte code  
      .word          0xF1DEBC9A        --4-7 byte code  
      .hword         0xD423            --8-9 byte code
```

备注 将以上程序例子添加到启动文件中。

第三十三章 电气规范（目标）

33.1 最大绝对额定值

($T_A=25^{\circ}\text{C}$) (1/2)

参数	符号	条件	额定值	单位
供电电压	V_{DD}	$V_{DD}=EV_{DD}=UV_{DD}=AV_{REF0}=AV_{REF1}$	-0.5 至 +4.6	V
	EV_{DD}	$V_{DD}=EV_{DD}=UV_{DD}=AV_{REF0}=AV_{REF1}$	-0.5 至 +4.6	V
	UV_{DD}	$V_{DD}=EV_{DD}=UV_{DD}=AV_{REF0}=AV_{REF1}$	-0.5 至 +4.6	V
	AV_{REF0}	$V_{DD}=EV_{DD}=UV_{DD}=AV_{REF0}=AV_{REF1}$	-0.5 至 +4.6	V
	AV_{REF1}	$V_{DD}=EV_{DD}=UV_{DD}=AV_{REF0}=AV_{REF1}$	-0.5 至 +4.6	V
	V_{SS}	$V_{SS}=AV_{SS}$	-0.5 至 +0.5	V
	AV_{SS}	$V_{SS}=AV_{SS}$	-0.5 至 +0.5	V
输入电压	V_{I1}	P60 至 P65、P90 至 P915、PCM0、PCM1、PCS0、PCS2、PCS3、PCT0、PCT1、PCT4、PCT6、PDL0 至 PDL15、PDH0 至 PDH7、RESET、FLMD0	-0.5 至 $EV_{DD} + 0.5^{\#1}$	V
	V_{I2}	UDMF、UDPF	-0.5 至 $UV_{DD} + 0.5^{\#1}$	V
	V_{I3}	P10、P11	-0.5 至 $AV_{REF1} + 0.5^{\#1}$	V
	V_{I4}	X1、X2、XT1、XT2	-0.5 至 $V_{RO}^{\#2} + 0.5^{\#1}$	V
	V_{I5}	P00 至 P05、P20 至 P25、P30 至 P37、P40 至 P42、P50 至 P56、PCM2、PCM3	-0.5 至 +6.0	V
模拟输入电压	V_{IAN}	P70 至 P711	-0.5 至 $AV_{REF0} + 0.5^{\#1}$	V

- 注意事项**
1. 不要将 IC 产品的输出（或输入/输出）引脚直接互连，或直接连接至 V_{DD} 、 V_{CC} 和 GND 。但是，漏极开路引脚或集电极开路引脚可以直接互连。如果输出引脚能够设置为高阻状态，并且外部电路的输出时序设计为避免输出冲突形式，则 IC 产品和外部电路之间的输出引脚才可以直接相连。
 2. 任何参数，哪怕瞬间超出其最大绝对额定值，都可能使产品性能受损。就是说，最大绝对额定值是指产品处于物理损坏边缘的额定值。所以，产品必须在不超出其最大绝对额定值的条件下使用。指示给 DC 和 AC 特性的额定值和工作条件代表正常工作期间性能得到保障的范围。

- 注**
1. 确保不超出各个供电电压的最大额定值（最大值）。
 2. 片上稳压器输出电压（2.5 V（TYP.））

备注 除非另外说明，复用功能引脚的特性与端口引脚的特性相同。

(T_A=25°C) (2/2)

参数	符号	条件		额定值	单位		
输出电流，低	I _{OL}	P00 至 P05、P20 至 P25、P30 至 P37 P40 至 P42、P50 至 P56、P60 至 P65 P90 至 P915 PCM0 至 PCM3、PCS0、PCS2、 PCS3 PCT0、PCT1、PCT4、PCT6 PDL0 至 PDL15、PDH0 至 PDH7	每个引脚	4	mA		
			所有引脚的总和	100	mA		
			UDMF、UDPF	每个引脚	4	mA	
			所有引脚的总和	16	mA		
		P10、P11	每个引脚	4	mA		
			所有引脚的总和	8	mA		
		P70 至 P711	每个引脚	4	mA		
			所有引脚的总和	20	mA		
		输出电流，高	I _{OH}	P00 至 P05、P20 至 P25、P30 至 P37 P40 至 P42、P50 至 P56、P60 至 P65 P90 至 P915、PCM0 至 PCM3、PCS0 PCS2、PCS3、PCT0、PCT1、PCT4 PCT6、PDL0 至 PDL15、PDH0 至 PDH7	每个引脚	−4	mA
					所有引脚的总和	−100	mA
UDMF、UDPF	每个引脚			−4	mA		
	所有引脚的总和			−16	mA		
P10、P11	每个引脚			−4	mA		
	所有引脚的总和			−8	mA		
P70 至 P711	每个引脚			−4	mA		
	所有引脚的总和			−20	mA		
运行环境温度	T _A			正常操作模式中		−40 至+85	°C
				Flash 存储器编程模式中		−40 至+85	°C
储存温度	T _{stg}			−40 至+125	°C		

- 注意事项**
1. 不要将 IC 产品的输出（或输入/输出）引脚直接互连，或直接连接至 V_{DD}、V_{CC} 和 GND。但是，漏极开路引脚或集电极开路引脚可以直接互连。如果输出引脚能够设置为高阻状态，并且外部电路的输出时序设计为避免输出冲突形式，则 IC 产品和外部电路之间的输出引脚才可以直接相连。
 2. 任何参数，哪怕瞬间超出其最大绝对额定值，都可能使产品性能受损。就是说，最大绝对额定值是指产品处于物理损坏边缘的额定值，所以，产品必须在不超出其最大额定值的条件下使用。指示给 DC 和 AC 特性的额定值和工作条件代表正常工作期间性能得到保障的范围。

备注 除非另外说明，复用功能引脚的特性与端口引脚的特性相同。

33.2 电容

($T_A=25^{\circ}\text{C}$ 、 $V_{DD}=EV_{DD}=UV_{DD}=AV_{REF0}=AV_{REF1}$ 、 $V_{SS}=AV_{SS}=0\text{ V}$)

参数	符号	条件	MIN.	TYP.	MAX.	单位
I/O 电容	C_{IO}	$f_x=1\text{ MHz}$ 量过的引脚返回到 0 V			10	pF

33.3 工作条件

($T_A=-40\text{ 至 }+85^{\circ}\text{C}$ 、 $V_{DD}=EV_{DD}=UV_{DD}=AV_{REF0}=AV_{REF1}$ 、 $V_{SS}=AV_{SS}=0\text{ V}$ 、 $C_L=50\text{ pF}$)

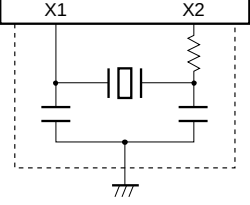
内部系统时钟频率	条件	供电电压				单位
		V_{DD}	EV_{DD}	UV_{DD}	AV_{REF0} 、 AV_{REF1}	
$f_{xx}=3\text{ 至 }6\text{ MHz}$ （时钟直通操作期间） $f_{xx}=24\text{ 至 }48\text{ MHz}$ （PLL 操作期间）	$C=4.7\text{ }\mu\text{F}$ ， A/D 转换器停止 D/A 转换器停止 USB 停止	2.85 至 3.6	2.85 至 3.6	2.85 至 3.6	2.85 至 3.6	V
	$C=4.7\text{ }\mu\text{F}$ ， A/D 转换器工作 D/A 转换器工作 USB 工作	3.0 至 3.6	3.0 至 3.6	3.0 至 3.6	3.0 至 3.6	V
$f_{XT}=32.768\text{ kHz}$	$C=4.7\text{ }\mu\text{F}$ ， 注	2.85 至 3.6	2.85 至 3.6	2.85 至 3.6	2.85 至 3.6	V

注 当系统在副时钟（ $f_{XT}=32.768\text{ kHz}$ ）下工作时，A/D 转换器、D/A 转换器和 USB 控制器不工作。

33.4 振荡器特性

33.4.1 主时钟振荡器特性

($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = E_{VDD} = U_{VDD} = A_{VREF0} = A_{VREF1}$ 、 $V_{SS} = A_{VSS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

谐振器	电路举例	参数	条件	MIN.	TYP.	MAX.	单位
陶瓷谐振器 / 晶体谐振器		振荡频率 (f_x) ^{注 1}		3		6	MHz
		振荡稳定时间 ^{注 2}	复位解除后		$2^{16}/f_x$		s
			STOP 模式解除后		注 3		ms
			IDLE2 模式解除后		注 3		μs

注 1. 上面显示的振荡频率仅指示振荡器特性。如此使用 V850ES/JG3-H 和 V850ES/JH3-H，以便内部工作条件不超出 **AC 特性**和 **DC 特性**中所示的额定值。

2. 从振荡开始到谐振器稳定所需的时间。

3. 该值随 OSTS 寄存器的设置不同而不同。

注意事项 1. 使用 **USB 控制器**时，务必在将内部时钟用作 **USB 时钟**时使用一个精确度小于 **6 MHz ±500 ppm** 的陶瓷谐振器或晶体谐振器。

通过 **UCLK** 引脚使用外部时钟输入时，务必提供一个精确度小于 **48 MHz ±500 ppm** 的时钟。

如果 **USB 时钟**精度下降，则发送/接收的数据不能满足 **USB 规范**。

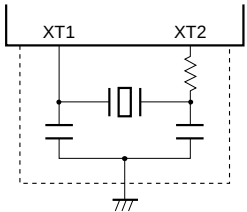
2. 当使用主时钟振荡器时，为了避免导线电容引起的负作用，应该按上图虚线框内的方式布线。。

- 保证连线尽可能短。
- 不要将该连线和其它信号线交叉。
- 布线时不要将该连线靠近有大波动电流通过的信号线。
- 保证振荡器电容的接地点和 V_{SS} 有相同的电位。
- 不要将电容连接到有大电流通过的接地点。
- 不要从振荡器获取信号。

3. 当主时钟停止，设备工作于副时钟时，在由副时钟切换回主时钟前，由程序实现的振荡稳定时间得到保证前一直处于等待状态。

33.4.2 副时钟振荡器特性

($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = EV_{DD} = UV_{DD} = AV_{REF0} = AV_{REF1}$ 、 $V_{SS} = AV_{SS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

谐振器	电路举例	参数	条件	MIN.	TYP.	MAX.	单位
晶体谐振器		振荡频率 (f_{XT}) ^{注 1}		32	32.768	35	kHz
		振荡稳定时间 ^{注 2}				10	s

注 1. 上面显示的振荡频率仅表示振荡器特性。如此使用 V850ES/JG3-H 和 V850ES/JH3-H，以便内部工作条件不超出 **AC 特性**和 **DC 特性**中所示的额定值。

2. 从 V_{DD} 达到振荡电压范围（2.85 V（最小值））起，到晶体振荡器稳定的所需时间。

注意事项 1. 当使用副时钟振荡器时，为了避免连接电容引起的负作用，应该如下连接上图虚线中的电路。

- 保证连线尽可能短。
 - 不要将该连线和其它信号线交叉。
 - 布线时不要将该连线靠近有大波动电流通过的信号线。
 - 保证振荡器电容的接地点和 V_{SS} 有相同的电位。
 - 不要将电容连接到有大电流通过的接地点。
 - 不要从振荡器获取信号。
2. 为了减少功耗，副时钟振荡器被设计为低振幅电路，而比起主时钟振荡器，其更容易受由于噪声引起的干扰。所以，使用副时钟时需要特别注意布线的方式。
3. 有关谐振器的选择和振荡器常量，需要用户自行评估振荡特性或者向谐振器厂商申请谐振器评估。

33.4.3 PLL特性

($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = E_{VDD} = U_{VDD} = A_{VREF0} = A_{VREF1}$ 、 $V_{SS} = A_{VSS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

参数	符号	条件	MIN.	TYP.	MAX.	单位
输入频率	f_x		3		6	MHz
输出频率	f_{xx}	时钟直通模式	3		6	MHz
		PLL 模式 (×8)	24		48	MHz
锁定时间	t_{PLL}				800	μs

33.4.4 内部振荡器特性

($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = E_{VDD} = U_{VDD} = A_{VREF0} = A_{VREF1}$ 、 $V_{SS} = A_{VSS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

参数	符号	条件	MIN.	TYP.	MAX.	单位
输出频率	f_R		100	220	400	kHz

33.5 DC特性

33.5.1 I/O电平

($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = EV_{DD} = UV_{DD} = AV_{REF0} = AV_{REF1}$ 、 $V_{SS} = AV_{SS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

参数	符号	条件	MIN.	TYP.	MAX.	单位
输入电压，高	V_{IH1}	$\overline{\text{RESET}}$ 、FLMD0、P60 至 P65 P90 至 P915	$0.8EV_{DD}$		EV_{DD}	V
	V_{IH2}	P00 至 P05、P20 至 P25、P30 至 P35、 P42、P50 至 P56	$0.8EV_{DD}$		5.5	V
	V_{IH3}	P36、P37、P40、P41、PCM2、PCM3	$0.7EV_{DD}$		5.5	V
	V_{IH4}	PDL0 至 PDL15、PDH0 至 PDH7、 PCM0、 PCM1、PCS0、PCS2、PCS3、PCT0、 PCT1、PCT4、PCT6	$0.7EV_{DD}$		EV_{DD}	V
	V_{IH5}	UDPF、UDMF	2.0		UV_{DD}	V
	V_{IH6}	P70 至 P711	$0.7AV_{REF0}$		AV_{REF0}	V
	V_{IH7}	P10、P11	$0.7AV_{REF1}$		AV_{REF1}	V
输入电压，低	V_{IL1}	$\overline{\text{RESET}}$ 、FLMD0、P60 至 P65、 P90 至 P915	V_{SS}		$0.2EV_{DD}$	V
	V_{IL2}	P00 至 P05、P20 至 P25、P30 至 P35、 P42、P50 至 P56	V_{SS}		$0.2EV_{DD}$	V
	V_{IL3}	P36、P37、P40、P41、PCM2、PCM3	V_{SS}		$0.3EV_{DD}$	V
	V_{IL4}	PDL0 至 PDL15、PDH0 至 PDH7、 PCM0、 PCM1、PCS0、PCS2、PCS3、PCT0、 PCT1、PCT4、PCT6	V_{SS}		$0.3EV_{DD}$	V
	V_{IL5}	UDPF、UDMF	V_{SS}		0.8	V
	V_{IL6}	P70 至 P711	AV_{SS}		$0.3AV_{REF0}$	V
	V_{IL7}	P10、P11	AV_{SS}		$0.3AV_{REF1}$	V
输入漏电流，高	I_{LIH}	$V_i = V_{DD} = EV_{DD} = UV_{DD} = AV_{REF0} = AV_{REF1}$			5	μA
输入漏电流，低	I_{LIL}	$V_i = 0\text{ V}$			-5	μA
输出漏电流，高	I_{LOH}	$V_o = V_{DD} = EV_{DD} = UV_{DD} = AV_{REF0} = AV_{REF1}$			5	μA
输出漏电流，低	I_{LOL}	$V_o = 0\text{ V}$			-5	μA

备注 除非另外说明，复用功能引脚的特性与端口引脚的特性相同。

(TA=-40 至+85°C、VDD=EVDD=UVDD=AVREF0=AVREF1、VSS=AVSS=0 V、CL=50 pF)

参数	符号	条件		MIN.	TYP.	MAX.	单位
<R> 输出电压，高	VOH1	注 1	每个引脚 IOH=-1.0 mA	EVDD - 1.0		EVDD	V
			每个引脚 IOH=-100 μ A	EVDD - 0.5		EVDD	V
	VOH2	P70 至 P711	每个引脚 IOH=-0.4 mA	AVREF0 - 1.0		AVREF0	V
			每个引脚 IOH=-100 μ A	AVREF0 - 0.5		AVREF0	V
	VOH3	P10、P11	每个引脚 IOH=-0.4 mA	AVREF1 - 1.0		AVREF1	V
			每个引脚 IOH=-100 μ A	AVREF1 - 0.5		AVREF1	V
	VOH4	UDPF、UDMF	RL=1.5 k Ω (VSS 连接)	2.8			V
			每个引脚 IOH=-100 μ A	UVDD - 0.5		UVDD	V
<R> 输出电压，低	VOL1	注 2	每个引脚 IOL=1.0 mA	0		0.4	V
	VOL2	P36、P37、P40、 P41、P90、P91	每个引脚 IOL=3.0 mA	0		0.4	V
	VOL3	P70 至 P711	每个引脚 IOL=1.0 mA	0		0.4	V
	VOL4	P10、P11	每个引脚 IOL=0.4 mA	0		0.4	V
	VOL5	UDPF、UDMF	RL=1.5 k Ω (UVDD 连接)	0		0.3	V
软件下拉电阻器	R1	P56	VI=VDD	10	30	100	k Ω

- 注 1. P00 至 P05、P20 至 P25、P30 至 P37、P40 至 P42、P50 至 P56、P60 至 P65、P90 至 P915、PCM0 至 PCM3、PCS0、PCS2、PCS3
PCT0、PCT1、PCT4、PCT6、PDH0 至 PDH7、PDL0 至 PDL15
2. P00 至 P05、P20 至 P25、P30 至 P35、P42、P50 至 P56、P60 至 P65、P92 至 P915、PCM0 至 PCM3、PCS0、PCS2、PCS3、PCT0、PCT1、PCT4、PCT6、PDH0 至 PDH7、PDL0 至 PDL15

- 备注 1. 除非另外说明，复用功能引脚的特性与端口引脚的特性相同。
2. 当 IOH 和 IOL 条件没有满足单个引脚但满足所有引脚的总值时，则仅该引脚视为不满足 DC 特性。

33.5.2 供应电流

($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = EV_{DD} = UV_{DD} = AV_{REF0} = AV_{REF1}$ 、 $V_{SS} = AV_{SS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

参数	符号	条件		MIN.	TYP.	MAX.	单位
供应电流 ^{注 1, 2}	I_{DD1}	正常工作模式	$f_{xx} = 48\text{ MHz}$ ($f_x = 6\text{ MHz}$) 外围功能运行		60	120	mA
	I_{DD2}	HALT 模式	$f_{xx} = 48\text{ MHz}$ ($f_x = 6\text{ MHz}$) 外围功能运行		42	84	mA
	I_{DD3}	IDLE1 模式	$f_{xx} = 48\text{ MHz}$ ($f_x = 6\text{ MHz}$) PLL 接通		4	10	mA
	I_{DD4}	IDLE2 模式	$f_{xx} = 6\text{ MHz}$ ($f_x = 6\text{ MHz}$) PLL 关闭		0.5	1	mA
	I_{DD5}	副时钟工作模式	$f_{xt} = 32.768\text{ kHz}$, 主时钟停止, 内部振荡器停止		120	600	μA
	I_{DD6}	副 IDLE 模式	$f_{xt} = 32.768\text{ kHz}$, 主时钟停止, 内部振荡器停止		13	150	μA
	I_{DD7}	STOP 模式	副时钟停止, 内部振荡器停止		10	130	μA
			副时钟运行, 内部振荡器停止		13	150	μA
	I_{DD8}	Flash 存储器编程模式	$f_{xx} = 48\text{ MHz}$ ($f_x = 6\text{ MHz}$)		65	130	mA

注 1. V_{DD} 、 EV_{DD} 和 UV_{DD} 电流的总值。 不包括流过输出缓冲器、A/D 转换器、D/A 转换器和片上下拉电阻的电流。

2. V_{DD} 的 TYP. 值为 3.3 V。

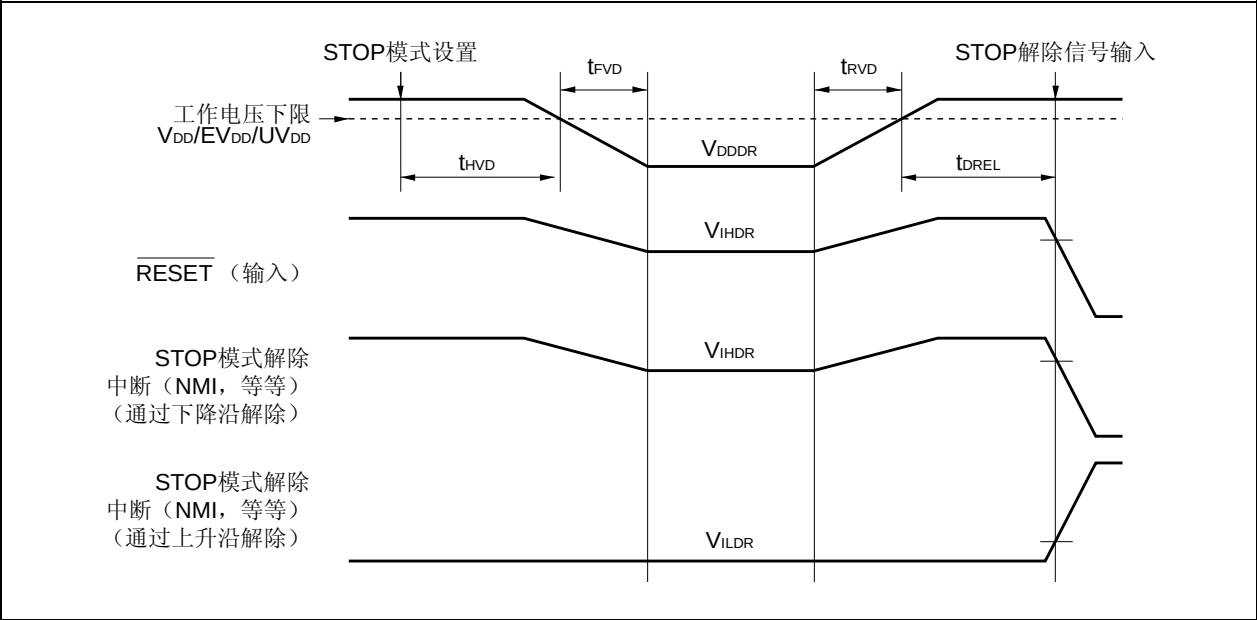
33.6 数据保持特性

(1) STOP 模式中

($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = EV_{DD} = UV_{DD} = AV_{REF0} = AV_{REF1}$ 、 $V_{SS} = AV_{SS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

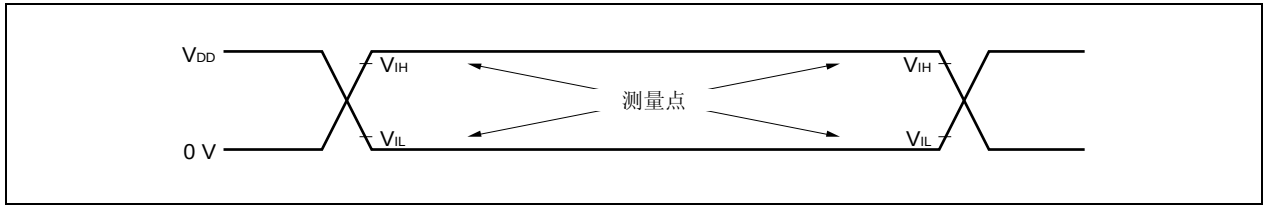
参数	符号	条件	MIN.	TYP.	MAX.	单位
数据保持电压	V_{DDDR}	STOP 模式（所有功能停止）	1.9		3.6	V
数据保持电流	I_{DDDR}	STOP 模式（所有功能停止）， $V_{DDDR} = 2.0\text{ V}$		10	130	μA
供电电压上升时间	t_{rVD}		200			μs
供电电压下降时间	t_{fVD}		200			μs
供电电压保持时间	t_{HVD}	STOP 模式设置后	0			ms
STOP 解除信号输入时间	t_{dREL}	V_{DD} 达到 2.85 V（最小值）后	0			ms
数据保持输入电压，高	V_{IHDR}	$V_{DD} = EV_{DD} = UV_{DD} = V_{DDDR}$	$0.9V_{DDDR}$		V_{DDDR}	V
数据保持输入电压，低	V_{ILDR}	$V_{DD} = EV_{DD} = UV_{DD} = V_{DDDR}$	0		$0.1V_{DDDR}$	V

注意事项 转换到 STOP 模式以及从 STOP 模式中恢复都必须在额定的工作范围内进行。

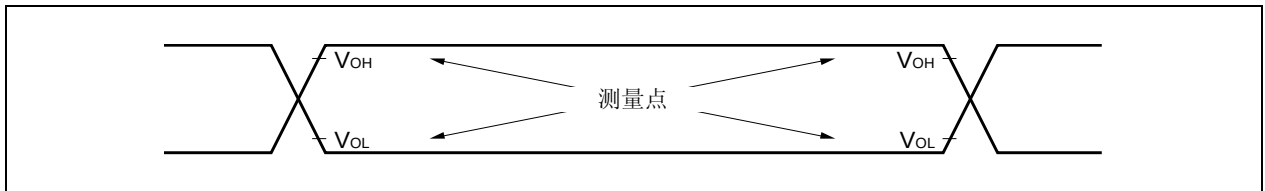


33.7 AC特性

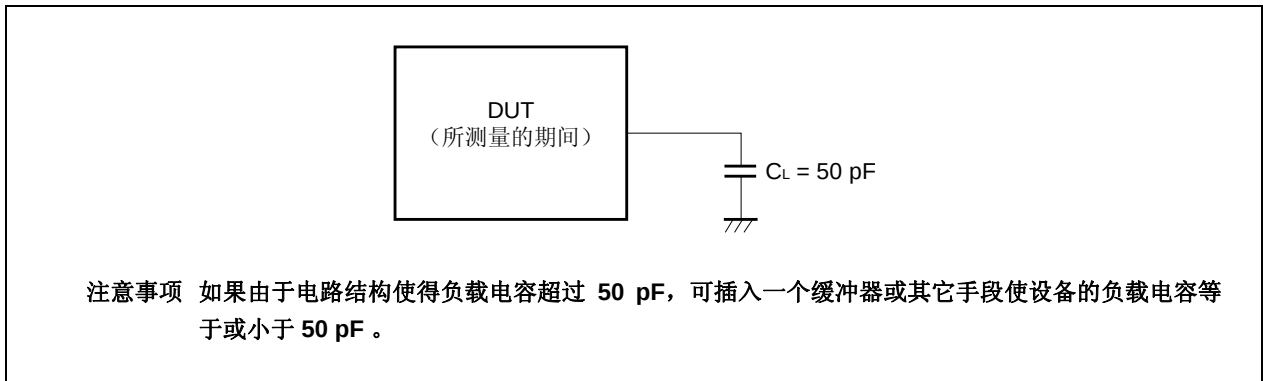
(1) AC 测试输入测量点 (V_{DD} 、 AV_{REF0} 、 AV_{REF1} 、 EV_{DD})



(2) AC 测试输出测量点



(3) 负载条件

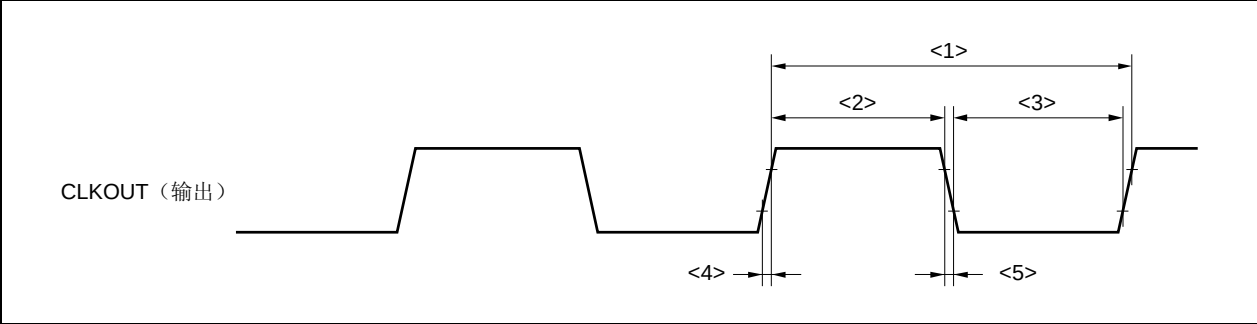


33.7.1 CLKOUT输出时序

($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = E_{VDD} = U_{VDD} = A_{VREF0} = A_{VREF1}$ 、 $V_{SS} = A_{VSS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

参数	符号		条件	MIN.	MAX.	单位
输出周期	t_{CYK}	<1>		20.83 ns	31.25 μs	
高电平宽度	t_{WKH}	<2>		$t_{CYK}/2 - 6$		ns
低电平宽度	t_{WKL}	<3>		$t_{CYK}/2 - 6$		ns
上升时间	t_{KR}	<4>			6	ns
下降时间	t_{KF}	<5>			6	ns

时钟时序



33.7.2 总线时序

(1) 复用总线模式/独立总线模式中

<R>

(a) 读/写周期（CLKOUT 异步）

(TA=-40 至+85°C、VDD=EVDD=UVDD=AVREF0=AVREF1、VSS=AVSS=0 V、CL=50 pF)

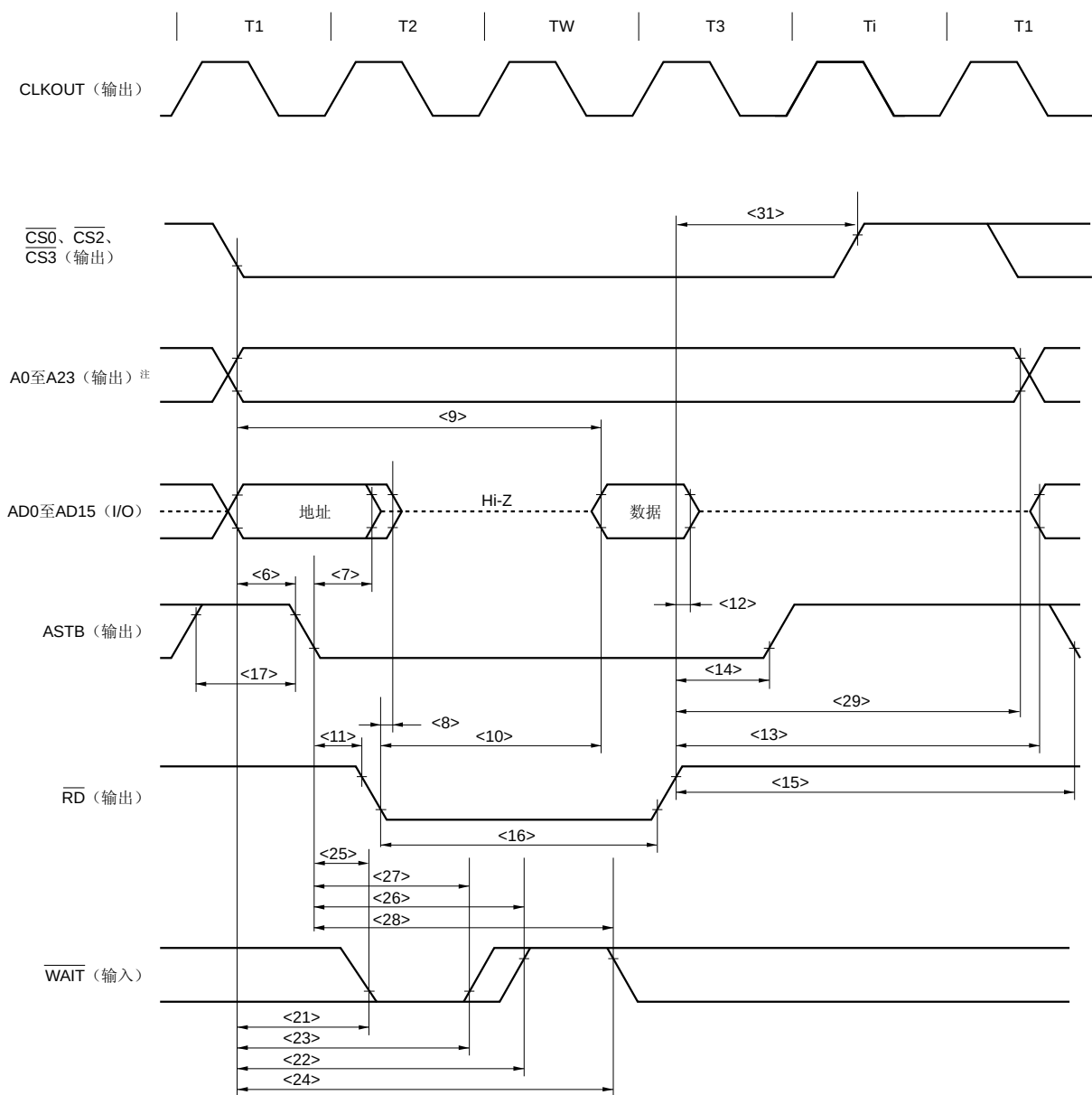
参数	符号	条件	MIN.	MAX.	单位
地址建立时间（至 ASTB↓）	tDAST	<6>	$(0.5 + t_{ASw}) T - 9$		ns
地址保持时间（自 ASTB↓起）	tHSTA	<7>	$(0.5 + t_{AHw}) T - 8$		ns
自 RD↓至地址悬空的延迟时间	tFRDA	<8>		5	ns
自地址起的数据输入建立时间	tDAID	<9>		$(2+n+ t_{ASw} + t_{AHw}) T - 25$	ns
自 RD↓起的数据输入建立时间	tDRDID2	<10>		$(1 + n) T - 15$	ns
自 ASTB↓至 RD↓的延时	tDSTRD	<11>	$(0.5 + t_{AHw}) T - 4$		ns
自 ASTB↓至 WRm↓的延迟时间	tDSTWR				
数据输入保持时间（自 RD↑起）	tHRDID	<12>	0		ns
自 RD↑起的地址输出保持时间	tDRDOD	<13>	$(1 + i) T - 3$		ns
自 RD↑至 ASTB↑的延迟时间	tDRDST	<14>	0.5T - 5		ns
自 WRm↑至 ASTB↑的延迟时间	tDWRST				
自 RD↑至 ASTB↓的延迟时间	tDRDST	<15>	$(1.5 + i + t_{ASw}) T - 4$		ns
RD 低电平宽度	tWRDL	<16>	$(1 + n) T - 10$		ns
WRm 低电平宽度	tWWRL				
ASTB 高电平宽度	tWSTH	<17>	$(1 + i + t_{ASw}) T - 10$		ns
自 WRm↓起的数据输出延迟时间	tDWRDOD	<18>		9	ns
数据输出延迟时间（自 WRm↑起）	tDODWR	<19>	$(1 + n) T - 11$		ns
数据输出保持时间（自 WRm↑起）	tHWROD	<20>	T - 3		ns
WAIT 建立时间（至地址）	tSAWT1	<21>	n ≥ 1	$(1.5+t_{ASw}+t_{AHw}) T - 25$	ns
	tSAWT2	<22>		$(1.5+n+t_{ASw}+t_{AHw}) T - 25$	ns
WAIT 保持时间（自地址起）	tHAWT1	<23>	n ≥ 1	$(0.5+n+ t_{ASw} + t_{AHw}) T$	ns
	tHAWT2	<24>		$(1.5+n + t_{ASw} + t_{AHw}) T$	ns
WAIT 建立时间（至 ASTB↓）	tSSTWT1	<25>	n ≥ 1	$(1 + t_{AHw}) T - 15$	ns
	tSSTWT2	<26>		$(1 + n + t_{AHw}) T - 15$	ns
WAIT 保持时间（自 ASTB↓起）	tHSTWT1	<27>	n ≥ 1	$(n + t_{AHw}) T$	ns
	tHSTWT2	<28>		$(1 + n + t_{AHw}) T$	ns
自 RD↑起的地址保持时间	tHRDA2	<29>	$(1 + i) T - 5$		ns
自 WRm↑起的地址保持时间	tHWRA2	<30>	T - 5		ns
自 RD↑至 CSn 的保持时间	tHRDC2	<31>	i ≥ 1	T - 5	ns
自 WRm↑至 CSn 的保持时间	tHWRC2	<32>		T - 5	ns

备注

1. tASW: 地址建立等待时钟的数目
tAHw: 地址保持等待时钟的数目
2. $T = 1/f_{CPU}$ (fCPU: CPU 工作时钟频率)
3. n: 在总线周期中插入的等待时钟数目
当插入可编程等待时, 采样时序改变。
4. m=0、1
5. i: 读周期后插入的空闲状态数目 (0 或 1)
6. 以上规格中的值为当 1: 1 占空比的时钟从 X1 输入时的值。

读周期（CLKOUT 异步）： 复用总线模式/独立总线模式中

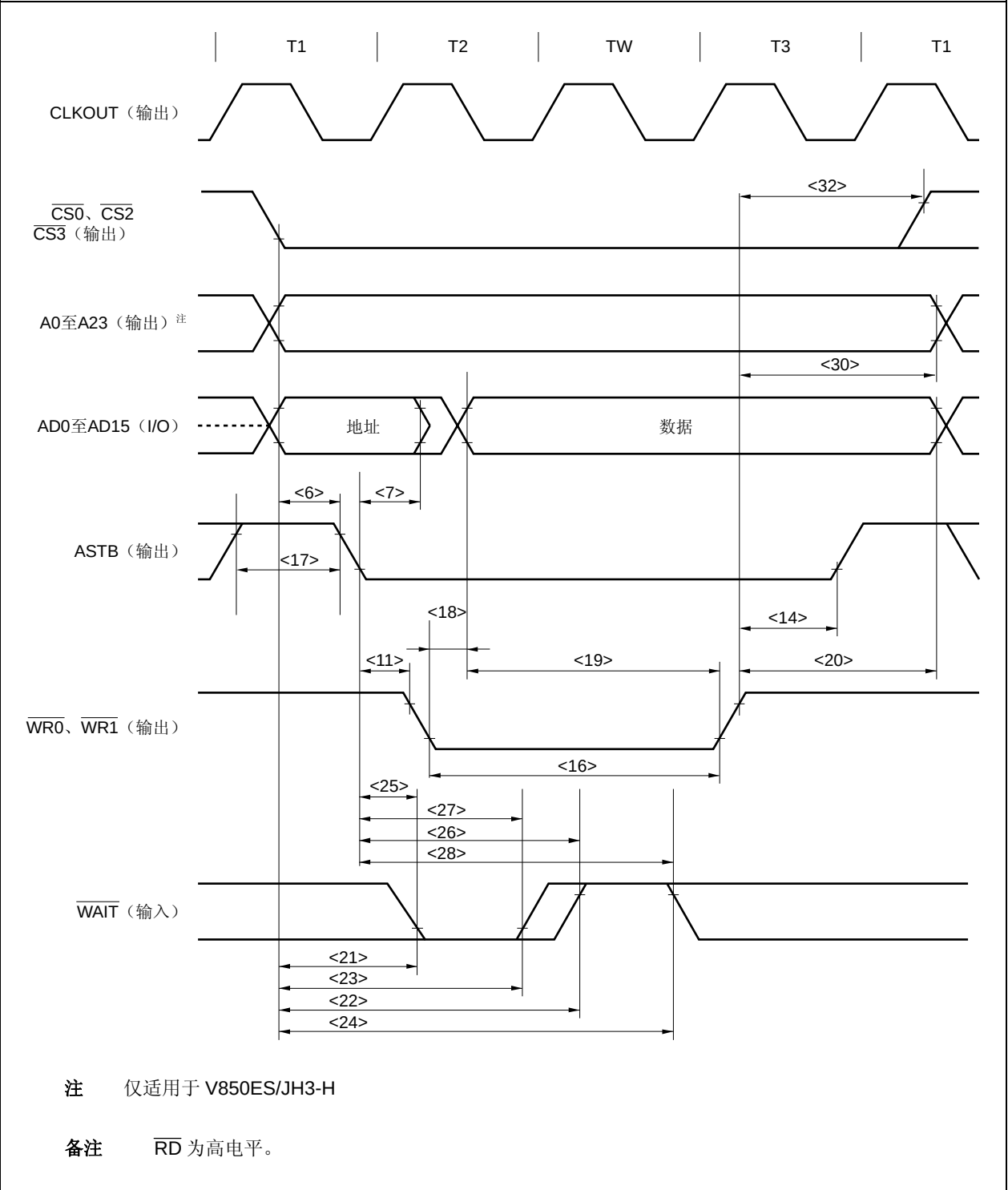
<R>



注 仅适用于 V850ES/JH3-H

备注 $\overline{WR0}$ 和 $\overline{WR1}$ 为高电平。

写周期（CLKOUT 异步）： 复用总线模式/独立总线模式中



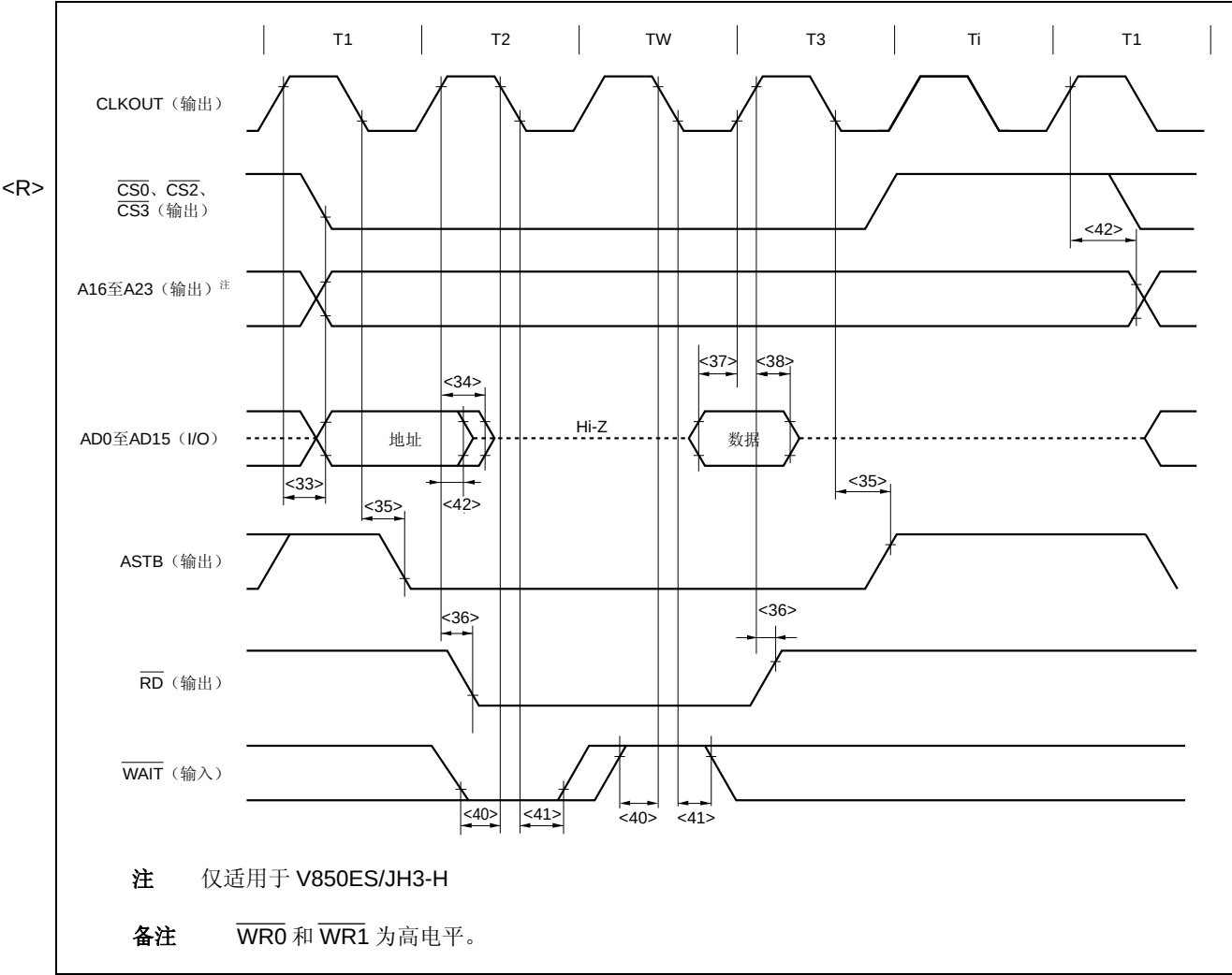
<R> (b) 读/写周期（CLKOUT 同步）： 复用总线模式/独立总线模式中

($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = EV_{DD} = UV_{DD} = AV_{REF0} = AV_{REF1}$ 、 $V_{SS} = AV_{SS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

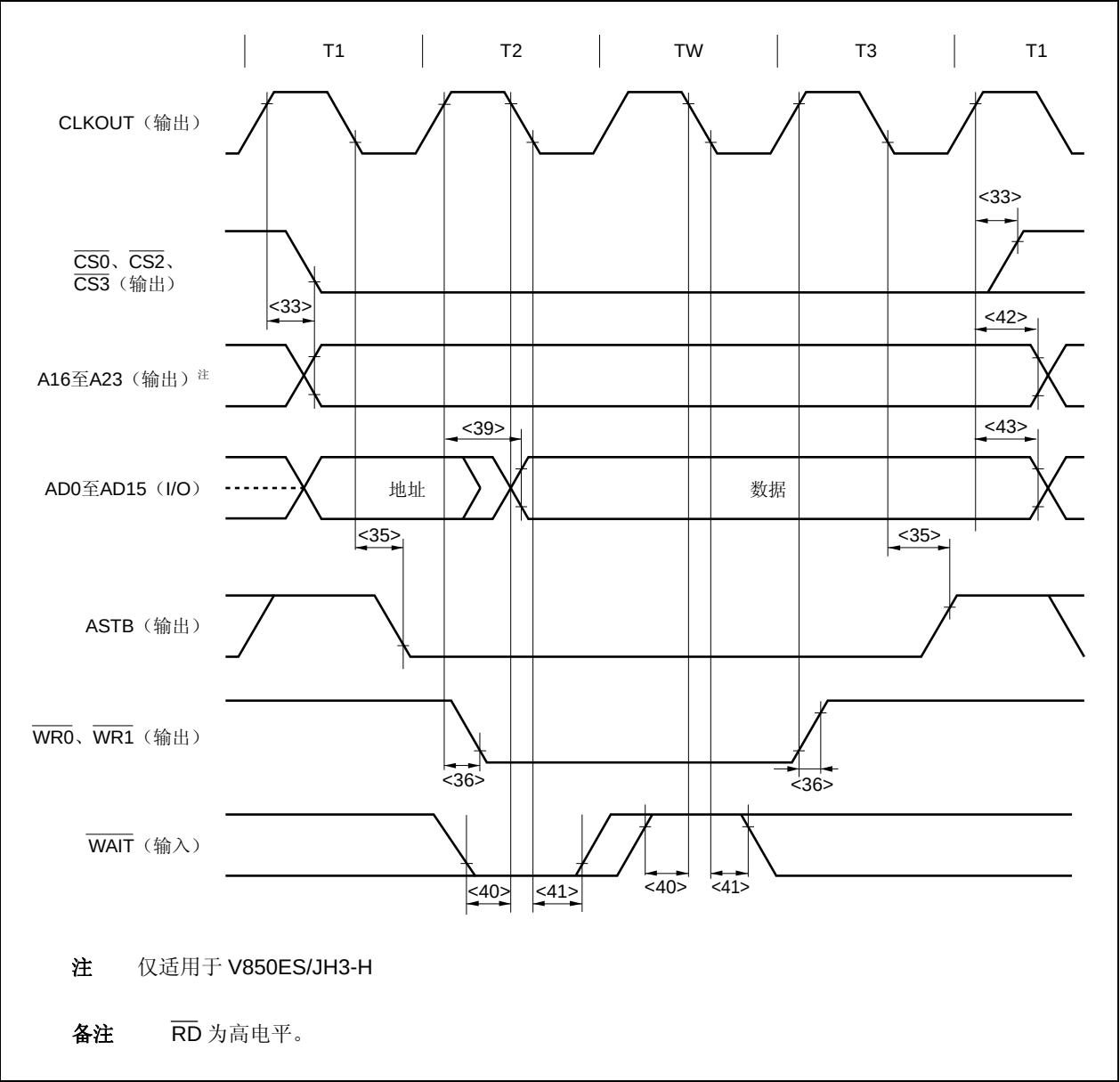
参数	符号	条件	MIN.	MAX.	单位
自 CLKOUT↑至地址的延迟时间	t_{DKA}	<33>	0	17	ns
自 CLKOUT↑至地址悬空的延迟时间	t_{FKA}	<34>	0	15	ns
自 CLKOUT↓至 ASTB 的延迟时间	t_{DKST}	<35>	0	12	ns
自 CLKOUT↑至 $\overline{RD}$ 、 $\overline{WRm}$ 的延迟时间	t_{DKRDWR}	<36>	0	12	ns
数据输入建立时间（至 CLKOUT↑）	t_{SIDK}	<37>	16		ns
数据输入保持时间（自 CLKOUT↑）	t_{HKID}	<38>	0		ns
自 CLKOUT↑起的数据输出延迟时间	t_{DKOD}	<39>		17	ns
$\overline{WAIT}$ 建立时间（至 CLKOUT↓）	t_{SWTK}	<40>	16		ns
$\overline{WAIT}$ 保持时间（自 CLKOUT↓）	t_{HKWT}	<41>	0		ns
自 CLKOUT↑起的地址保持时间	t_{HKA2}	<42>	0		ns
自 CLKOUT↑起的数据输出保持时间	t_{HKOD2}	<43>	0		ns

备注 1. $m=0、1$
2. 以上规格中的值为当 1：1 占空比的时钟从 X1 输入时的值。

读周期（CLKOUT 同步）： 复用总线模式/独立总线模式中



写周期（CLKOUT 同步）： 复用总线模式/独立总线模式中



(2) 总线保持期间（仅 V850ES/JH3-H）

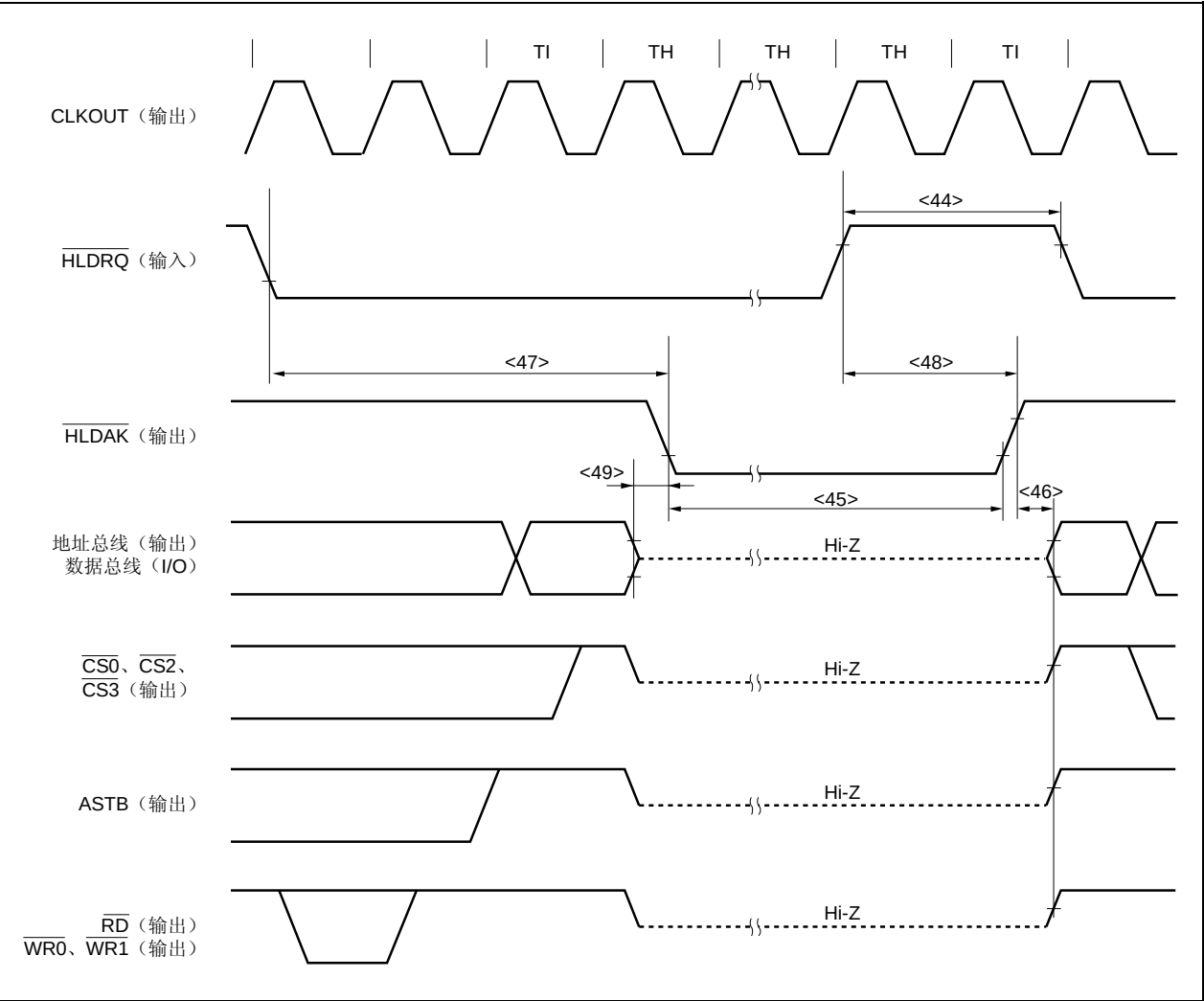
(a) CLKOUT 异步

($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = EV_{DD} = UV_{DD} = AV_{REF0} = AV_{REF1}$ 、 $V_{SS} = AV_{SS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

参数	符号	条件	MIN.	MAX.	单位
$\overline{\text{HLDRQ}}$ 高电平宽度	t_{WHQH}	<44>	$T + 16$		ns
$\overline{\text{HLD\!AK}}$ 低电平宽度	t_{WHAL}	<45>	$T - 10$		ns
自 $\overline{\text{HLD\!AK}}\uparrow$ 至总线输出的延时	t_{DHAC}	<46>	-7		ns
自 $\overline{\text{HLDRQ}}\downarrow$ 至 $\overline{\text{HLD\!AK}}\downarrow$ 的延时	t_{DHQHA1}	<47>	$2.5T$		ns
自 $\overline{\text{HLDRQ}}\uparrow$ 至 $\overline{\text{HLD\!AK}}\uparrow$ 的延时	t_{DHQHA2}	<48>	$0.5T + 17$	$1.5T + 31$	ns
自总线悬空至 $\overline{\text{HLD\!AK}}\downarrow$ 的延时	t_{DFHA}	<49>	0		ns

- 备注
- 1. $T = 1/f_{\text{CPU}}$ (f_{CPU} : CPU 工作时钟频率)
 - 2. n: 在总线周期中插入的等待时钟数目
当插入可编程等待时, 采样时序改变。
 - 3. 以上规格中的值为当 1: 1 占空比的时钟从 X1 输入时的值。

总线保持（CLKOUT 异步）



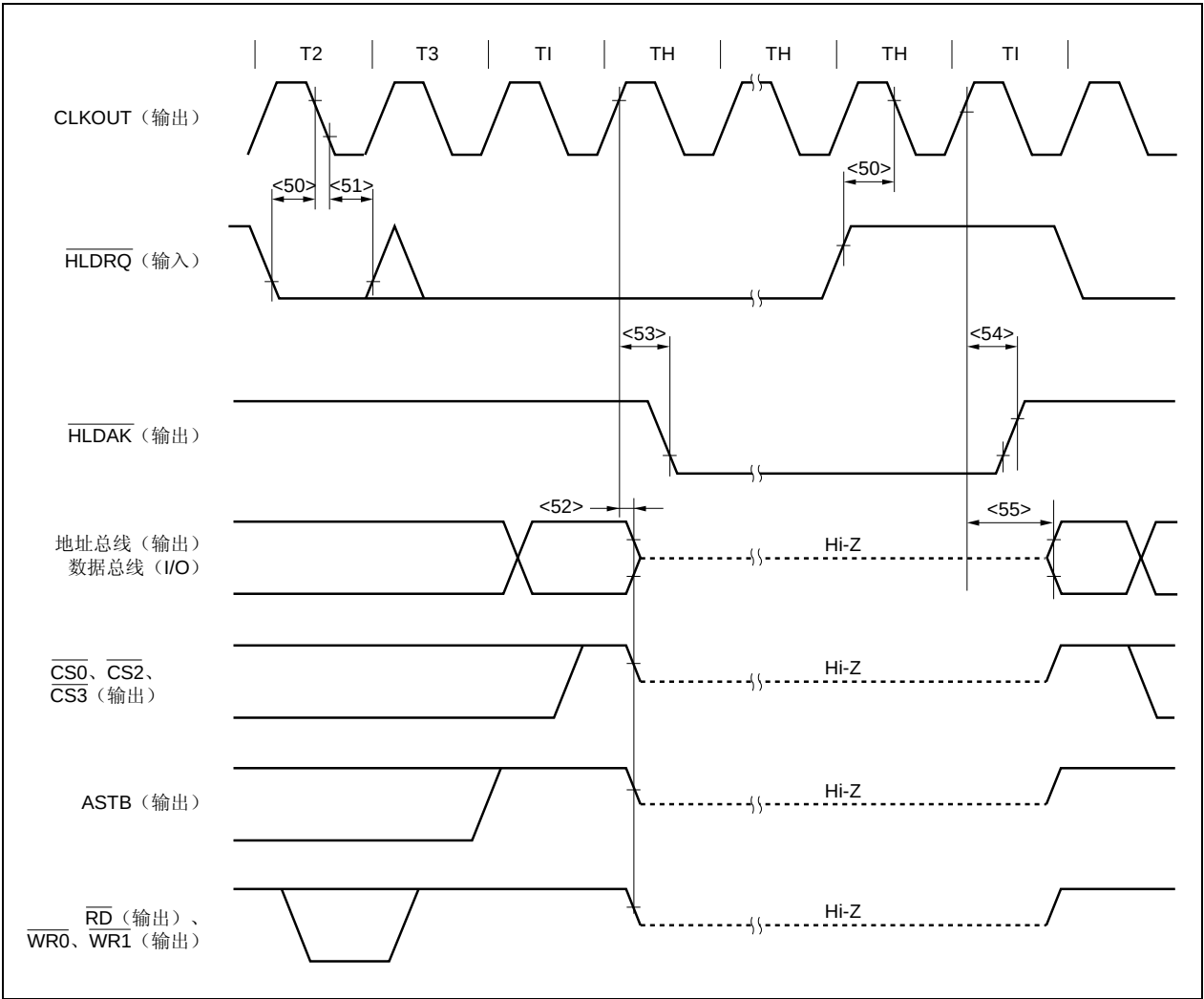
<R> (b) CLKOUT 同步

(TA=−40 至+85°C、VDD=EVDD=UVDD=AVREF0=AVREF1、VSS=AVSS=0 V、CL=50 pF)

参数	符号		条件	MIN.	MAX.	单位
HLD $\overline{\text{RQ}}$ 建立时间（至 CLKOUT↓）	t _{SHQK}	<50>		16		ns
HLD $\overline{\text{RQ}}$ 保持时间（自 CLKOUT↓）	t _{HKHQ}	<51>		0		ns
自 CLKOUT↑至总线悬空的延时	t _{DKF}	<52>			15	ns
自 CLKOUT↑至 HLD $\overline{\text{AK}}$ ↓的延时	t _{DKHA1}	<53>		1	15	ns
自 CLKOUT↑至 HLD $\overline{\text{AK}}$ ↑的延时	t _{DKHA2}	<54>		1	15	ns
自 CLKOUT↑至数据输出的延迟时间	t _{DKBO}	<55>		1	17	ns

备注 以上规格中的值为当 1: 1 占空比的时钟从 X1 输入时的值。

总线保持（CLKOUT 同步）



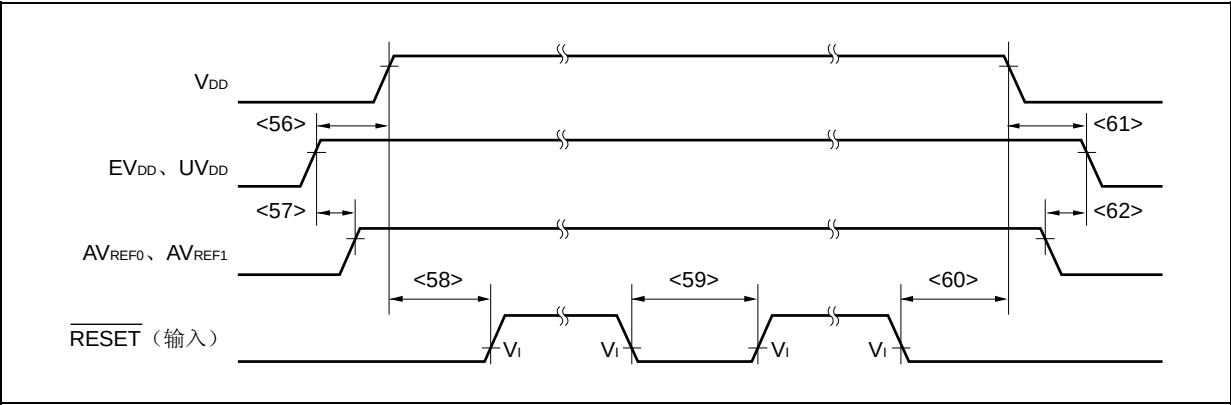
33.8 基本操作

(1) 电源开启/关闭/复位时序

($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = EV_{DD} = UV_{DD} = AV_{REF0} = AV_{REF1}$ 、 $V_{SS} = AV_{SS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

参数	符号		条件	MIN.	MAX.	单位
自 EV_{DD} 、 UV_{DD} ↑至 V_{DD} ↑的时间	t_{REL}	<56>		0		ns
自 EV_{DD} 、 UV_{DD} ↑至 AV_{REF0} 、 AV_{REF1} ↑的时间	t_{REA}	<57>		0	t_{REL}	ns
自 V_{DD} ↑至 $\overline{RESET}$ ↑的时间	t_{RER}	<58>		$500 + t_{REG}^{\#}$		ns
$\overline{RESET}$ 低电平宽度	t_{WRSL}	<59>	模拟噪声抑制（flash 擦除/写入期间）	500		ns
			模拟噪声抑制	500		ns
自 $\overline{RESET}$ ↓至 V_{DD} ↓的时间	t_{FRE}	<60>		500		ns
自 V_{DD} ↓至 EV_{DD} 、 UV_{DD} ↓的时间	t_{FEL}	<61>		0		ns
自 AV_{REF0} 、 AV_{REF1} ↓至 EV_{DD} 、 UV_{DD} ↓的时间	t_{FEA}	<62>		0	t_{FEL}	ns

注 根据片上稳压器的特性。



(2) 复位、中断时序

($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = EV_{DD} = UV_{DD} = AV_{REF0} = AV_{REF1}$ 、 $V_{SS} = AV_{SS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

参数	符号	条件	MIN.	MAX.	单位
RESET 输入低电平宽度	t_{WRSL}		500		ns
NMI 高电平宽度	t_{WNIH}	模拟噪声抑制	500		ns
NMI 低电平宽度	t_{WNIL}	模拟噪声抑制	500		ns
INTPn 高电平宽度	t_{WITH}	n=0 至 18（模拟噪声抑制）	500		ns
		n=2（数字噪声抑制）	$3T_{SMP} + 20$		ns
INTPn 低电平宽度	t_{WITL}	n=0 至 18（模拟噪声抑制）	500		ns
		n=2（数字噪声抑制）	$3T_{SMP} + 20$		ns

备注 T_{SMP} ：由噪声抑制控制寄存器（INTNFC）设置。 可从 fxx/64、fxx/128、fxx/256、fxx/512 和 fxx/1024 中选择。

($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = EV_{DD} = UV_{DD} = AV_{REF0} = AV_{REF1}$ 、 $V_{SS} = AV_{SS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

参数	符号	条件	MIN.	MAX.	单位
KRn 高电平宽度	t_{WKRH}	模拟噪声抑制	500		ns
KRn 低电平宽度	t_{WKRL}	模拟噪声抑制	500		ns

备注 n = 0 至 7

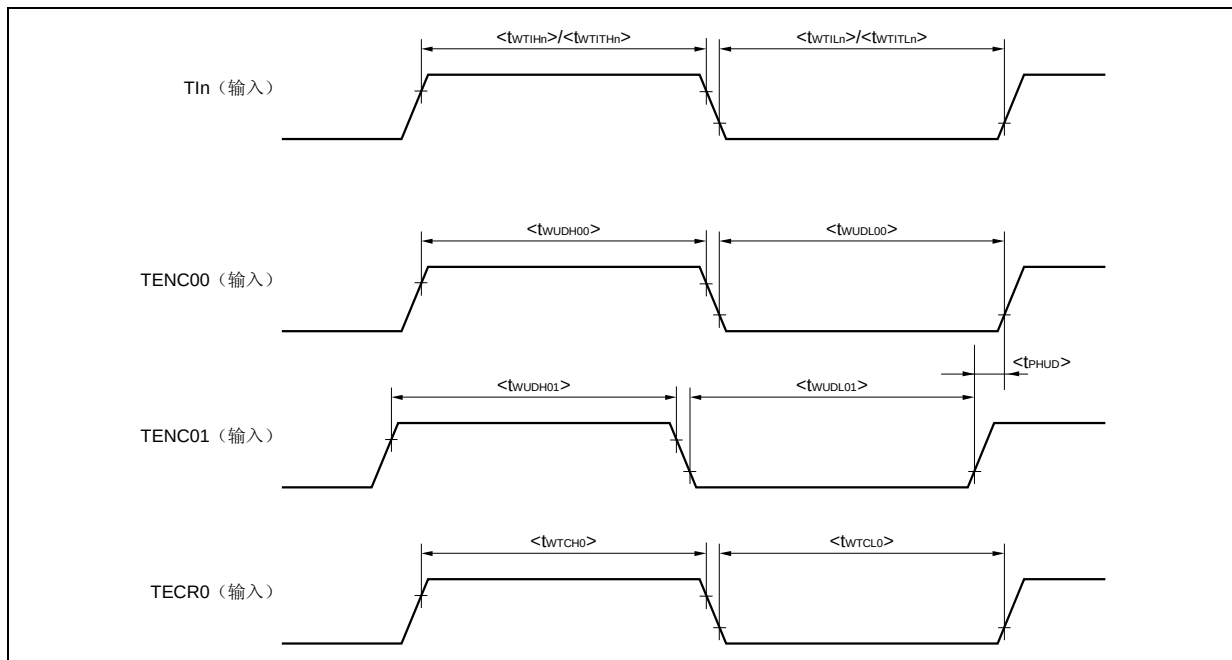
(4) 定时器时序

($T_A = -40$ 至 $+85^\circ\text{C}$ 、 $V_{DD} = EV_{DD} = UV_{DD} = AV_{REF0} = AV_{REF1}$ 、 $V_{SS} = AV_{SS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

参数	符号	条件	MIN.	MAX.	单位
TI 高电平宽度	t_{TIH}	TAB00 至 TAB03、TAB10 至 TAB13、 EVTAB1、TRGAB1	$12T + 20$		ns
		TIAA00、TIAA01、TIAA10、 TIAA11、TIAA20、TIAA21、 TIAA30、TIAA31、TIAA50、TIAA51,	$3T_{SMP1} + 20$		ns
TI 低电平宽度	t_{TIL}	TAB00 至 TAB03、TAB10 至 TAB13、 EVTAB1、TRGAB1	$12T + 20$		ns
		TIAA00、TIAA01、TIAA10、 TIAA11、TIAA20、TIAA21、 TIAA30、TIAA31、TIAA50、TIAA51,	$3T_{SMP1} + 20$		ns
TENCn 高电平宽度	t_{WENCHn}	$n=0, 1$	$3T_{SMP2} + 20$		ns
TENCn 低电平宽度	t_{WENCLn}	$n=0, 1$	$3T_{SMP2} + 20$		ns
TECR0 高电平宽度	t_{WCRH0}		$3T_{SMP2} + 20$		ns
TECR0 低电平宽度	t_{WCRL0}		$3T_{SMP2} + 20$		ns
TItn 高电平宽度	t_{WTITHn}	$n=0, 1$	$3T_{SMP2} + 20$		ns
TItn 低电平宽度	t_{WTITLn}	$n=0, 1$	$3T_{SMP2} + 20$		ns
EVTt0 高电平宽度	t_{WTITH0}		$3T_{SMP2} + 20$		ns
EVTt0 低电平宽度	t_{WTITL0}		$3T_{SMP2} + 20$		ns
TENCn 输入时间差	t_{PHUD}	$n=0, 1$	$3T_{SMP2} + 20$		ns

备注

1. $T = 1/f_{xx}$
2. T_{SMP1} : 由噪声抑制控制寄存器 (TANFC) 设置。 可以从 f_{xx} 和 $f_{xx}/4$ 中选择。
3. T_{SMP2} : 由噪声抑制控制寄存器 (TTNFC) 设置。 可从 $f_{xx}/4$ 、 $f_{xx}/8$ 、 $f_{xx}/16$ 、 $f_{xx}/32$ 和 $f_{xx}/64$ 中选择。
4. 上面的规格中显示了能作为有效边沿被精确检测的脉冲宽度。 因此, 即使输入的脉冲宽度小于上面规定的, 也可能会将其作为有效沿检测。



(5) UARTC 时序

($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = E_{VDD} = U_{VDD} = A_{VREF0} = A_{VREF1}$ 、 $V_{SS} = A_{VSS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

参数	符号	条件	MIN.	MAX.	单位
传输速率				1.25	Mbps
ASCK0 周期时间				10	MHz

(6) CSIF 时序

(a) 主机模式

[使用 CSI0 至 CSIF2、或 CSIF4 时]

<R> ($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = EV_{DD} = UV_{DD} = AV_{REF0} = AV_{REF1}$ 、 $V_{SS} = AV_{SS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

参数	符号	条件	MIN.	MAX.	单位
$\overline{\text{SCKFn}}$ 周期时间	t_{KCY1}	<63>	125		ns
$\overline{\text{SCKFn}}$ 高电平宽度	t_{KH1}	<64>	$t_{\text{KCY1}}/2 - 8$		ns
$\overline{\text{SCKFn}}$ 低电平宽度	t_{KL1}		$t_{\text{KCY1}}/2 - 8$		ns
SIFn 建立时间（至 $\overline{\text{SCKFn}}\uparrow$ ）	t_{SIK1}	<65>	27		ns
SIFn 建立时间（至 $\overline{\text{SCKFn}}\downarrow$ ）			27		ns
SIFn 保持时间（自 $\overline{\text{SCKFn}}\uparrow$ 起）	t_{SI1}	<66>	27		ns
SIFn 保持时间（自 $\overline{\text{SCKFn}}\downarrow$ 起）			27		ns
SOFn 输出延迟时间（自 $\overline{\text{SCKFn}}\uparrow$ 起）	t_{SO1}	<67>		27	ns
SOFn 输出延迟时间（自 $\overline{\text{SCKFn}}\downarrow$ 起）				27	ns
SOFn 输出保持时间（自 $\overline{\text{SCKFn}}\uparrow$ 起）	t_{HSO1}	<68>	$t_{\text{KCY1}}/2 - 10$		ns
SOFn 输出保持时间（自 $\overline{\text{SCKFn}}\downarrow$ 起）			$t_{\text{KCY1}}/2 - 10$		ns

备注 n=0 至 2、4

[使用 CSI3 时]

<R> ($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = EV_{DD} = UV_{DD} = AV_{REF0} = AV_{REF1}$ 、 $V_{SS} = AV_{SS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

参数	符号	条件	MIN.	MAX.	单位
$\overline{\text{SCKF3}}$ 周期时间	t_{KCYM}	<63>	83.3		ns
$\overline{\text{SCKF3}}$ 高电平宽度	t_{KHM}	<64>	$t_{\text{KCYM}}/2 - 8$		ns
$\overline{\text{SCKF3}}$ 低电平宽度			$t_{\text{KCYM}}/2 - 8$		ns
SIF3 建立时间（至 $\overline{\text{SCKF3}}\uparrow$ ）	t_{SIKM}	<65>	16		ns
SIF3 建立时间（至 $\overline{\text{SCKF3}}\downarrow$ ）			16		ns
SIF3 保持时间（自 $\overline{\text{SCKF3}}\uparrow$ 起）	t_{SIM}	<66>	16		ns
SIF3 保持时间（自 $\overline{\text{SCKF3}}\downarrow$ 起）			16		ns
SOF3 输出延迟时间（自 $\overline{\text{SCKF3}}\uparrow$ 起）	t_{SOM}	<67>		16	ns
SOF3 输出延迟时间（自 $\overline{\text{SCKF3}}\downarrow$ 起）				16	ns
SOF3 输出保持时间（自 $\overline{\text{SCKF3}}\uparrow$ 起）	t_{HSOM}	<68>	$t_{\text{KCYM}}/2 - 10$		ns
SOF3 输出保持时间（自 $\overline{\text{SCKF3}}\downarrow$ 起）			$t_{\text{KCYM}}/2 - 10$		ns

(b) 从机模式

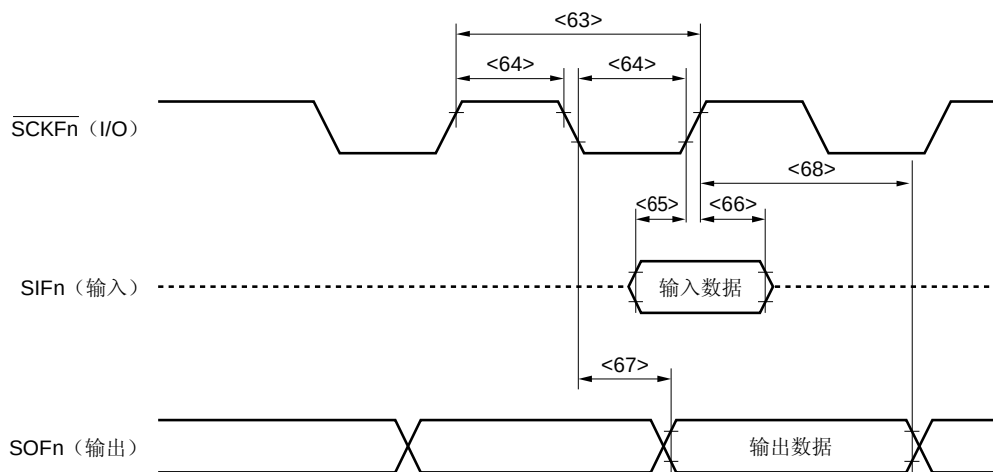
[使用 CSIO 至 CSIF2、或 CSIF4 时]

<R> (TA=−40 至+85°C、VDD=EVDD=UVDD=AVREF0=AVREF1、VSS=AVSS=0 V、CL=50 pF)

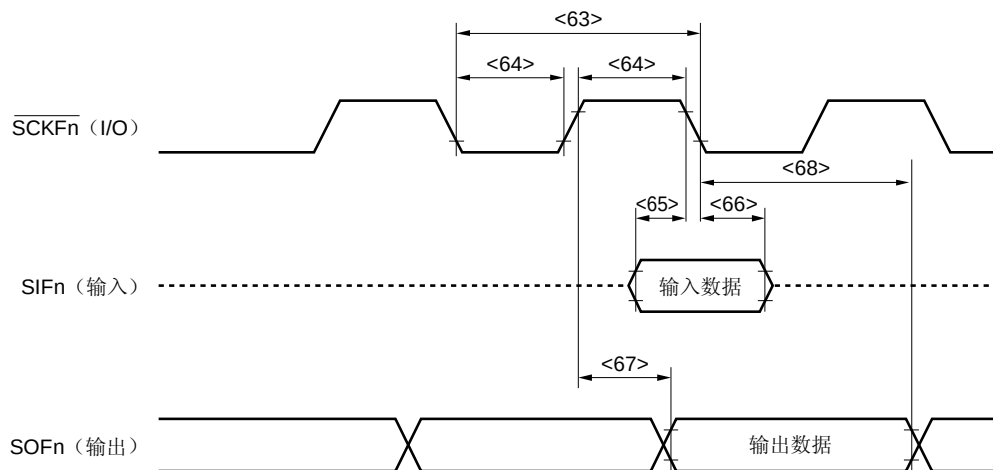
参数	符号		条件	MIN.	MAX.	单位
SCKFn 周期时间	tKCY2	<63>		125		ns
SCKFn 高电平宽度	tKH2	<64>		tKCYn/2 − 8		ns
SCKFn 低电平宽度	tKL2			tKCYn/2 − 8		ns
SIFn 建立时间（至 SCKFn↑）	tSIK2	<65>		27		ns
SIFn 建立时间（自 SCKFn↓起）				27		ns
SIFn 保持时间（至 SCKFn↑）	tKSI2	<66>		27		ns
SIFn 保持时间（自 SCKFn↓起）				27		ns
SOFn 输出延迟时间（至 SCKFn↑）	tKSO2	<67>			27	ns
SOFn 输出延迟时间（自 SCKFn↓起）					27	ns
SOFn 输出延迟时间（至 SCKFn↑）	tHSO2	<68>		tKCYn/2 − 10		ns
SOFn 输出延迟时间（自 SCKFn↓起）				tKCYn/2 − 10		ns

备注 n=0 至 4

(a) CFnCTL1.CFnCKP、CFnDAP 位=00 或 11



(b) CFnCTL1.CFnCKP、CFnDAP 位=10 或 01



备注 n=0 至 4

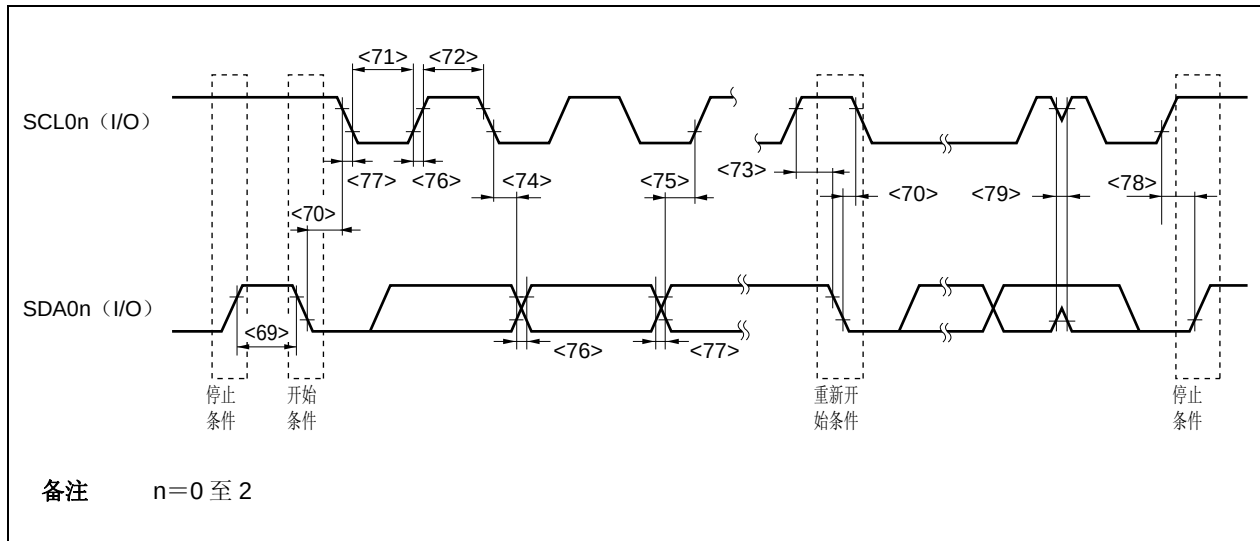
(7) I²C 总线模式(T_A = -40 至 +85°C、V_{DD} = E_{VDD} = U_{VDD} = A_{VREF0} = A_{VREF1}、V_{SS} = A_{VSS} = 0 V)

参数	符号		正常模式		高速模式		单位
			MIN.	MAX.	MIN.	MAX.	
SCL0n 时钟频率	f _{CLK}		0	100	0	400	kHz
总线空闲时间 (开始和停止状态之间)	t _{BUF}	<69>	4.7	—	1.3	—	μs
保持时间 ^{注 1}	t _{HD:STA}	<70>	4.0	—	0.6	—	μs
SCL0n 时钟低电平宽度	t _{LOW}	<71>	4.7	—	1.3	—	μs
SCL0n 时钟高电平宽度	t _{HIGH}	<72>	4.0	—	0.6	—	μs
开始/重新开始状态的建立时间	t _{SU:STA}	<73>	4.7	—	0.6	—	μs
数据保持 时间	CBUS 兼容主机	t _{HD:DAT}	<74>	5.0	—	—	μs
	I ² C 模式		0 ^{注 2}	—	0 ^{注 2}	0.9 ^{注 3}	μs
数据建立时间	t _{SU:DAT}	<75>	250	—	100 ^{注 4}	—	ns
SDA0n 和 SCL0n 信号上升时间	t _R	<76>	—	1000	20 + 0.1Cb ^{注 5}	300	ns
SDA0n 和 SCL0n 信号下降时间	t _F	<77>	—	300	20 + 0.1Cb ^{注 5}	300	ns
停止状态建立时间	t _{SU:STO}	<78>	4.0	—	0.6	—	μs
由输入滤波器抑制的毛刺脉冲宽度	t _{SP}	<79>	—	—	0	50	ns
各总线的电容负载	C _b		—	400	—	400	pF

- 注
1. 当满足开始条件时，在保持时间后生成第一个时钟脉冲。
 2. 用于 SDA0n（在 SCL0n 信号的 V_{IHmin} 处）信号的系统内部需要最短 300 ns 的保持时间，以占用 SCL0n 信号下降沿的不确定区域。
 3. 如果系统不延长 SCL0n 信号低电平保持时间（t_{LOW}），则只需要满足最大数据保持时间（t_{HD: DAT}）。
 4. 高速模式 I²C 总线能够用在正常模式 I²C 总线系统中。这种情况下，设置高速模式 I²C 总线使其满足以下条件。
 - 如果系统不延长 SCL0n 信号低电平保持时间：

$$t_{SU:DAT} \geq 250 \text{ ns}$$
 - 如果系统延长 SCL0n 信号低电平保持时间：
 在 SCL0n 信号线释放前，将下一个数据位输出到 SDA0n 信号线（t_{Rmax} + t_{SU: DAT} = 1, 000 + 250 = 1, 250 ns：指定正常模式 I²C 总线）。
 5. C_b：每条总线的总电容（单位：pF）。

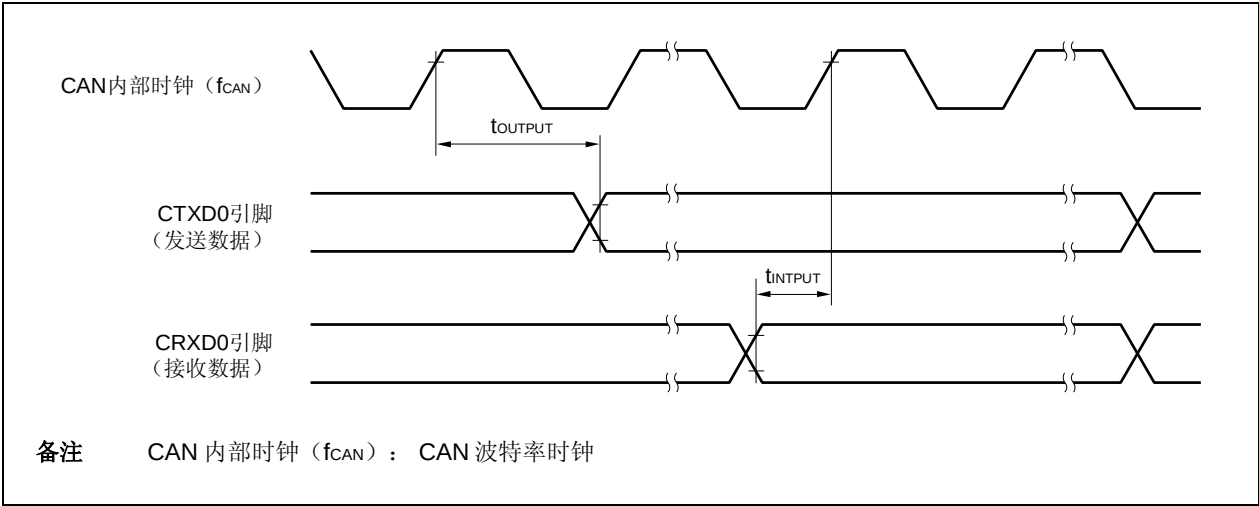
备注 n=0 至 2



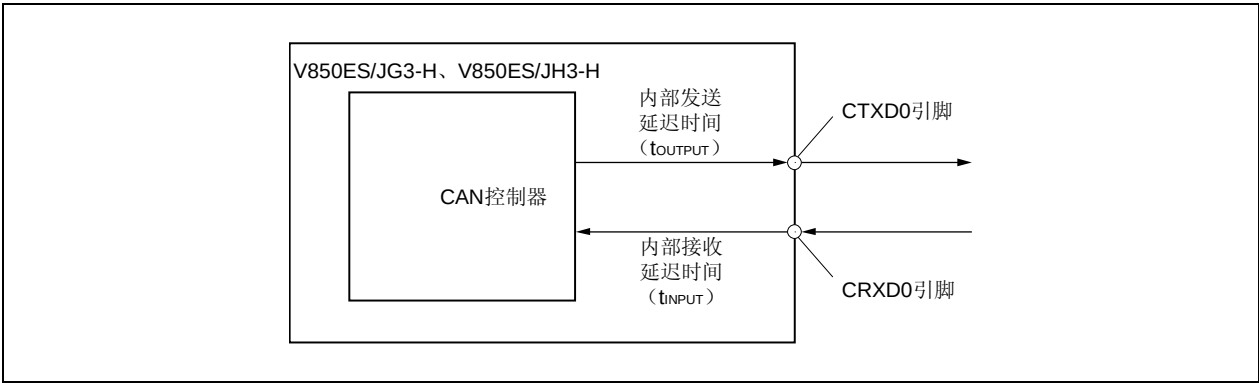
(8) CAN 时序（仅 CAN 控制器版本）

($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = E V_{DD} = U V_{DD} = A V_{REF0} = A V_{REF1}$ 、 $V_{SS} = A V_{SS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

参数	符号	条件	MIN.	MAX.	单位
传输速率				1	Mbps
内部延迟时间	t_{NODE}			100	ns



内部延迟时间 (t_{NODE}) = 内部发送延迟时间 (t_{OUTPUT}) + 内部接收延迟时间 (t_{INPUT})



(9) 高阻抗控制时序

($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = E_{VDD} = U_{VDD} = A_{VREF0} = A_{VREF1}$ 、 $V_{SS} = A_{VSS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

参数	符号	条件	MIN.	MAX.	单位
从振荡器停止到定时器输出高阻抗的时间	t_{CLM}	时钟监控运行中		65	μs
从 TOAB1OFF 输入→定时器输出高阻抗的时间	t_{HTQn}			300	ns
从 TOAA1OFF 输入→定时器输出高阻抗的时间	t_{HTP2}			300	ns

(10) A/D 转换器

($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = E_{VDD} = U_{VDD} = A_{VREF0} = A_{VREF1}$ 、 $3.0\text{ V} \leq A_{VREF0} \leq 3.6\text{ V}$ 、 $V_{SS} = A_{VSS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

参数	符号	条件	MIN.	TYP.	MAX.	单位
分辨率					10	位
总误差 ^注		$3.0 \leq A_{VREF0} \leq 3.6\text{ V}$			± 0.6	%FSR
<R> 转换时间	t_{CONV}		2.17		24	μs
零刻度误差					± 0.5	%FSR
满刻度误差					± 0.5	%FSR
非线性误差					± 4.0	LSB
微分线性误差					± 4.0	LSB
模拟输入电压	V_{IAN}		A_{VSS}		A_{VREF0}	V
参考电压	A_{VREF0}		3.0		3.6	V
A _{VREF0} 电流	A_{IREF0}	正常转换模式		3	6.5	mA
		高速转换模式		4	10	mA
		未使用 A/D 转换器时			5	μA

注 量化误差除外 ($\pm 0.05\%$ FSR)。

注意事项 A/D 转换期间，不要设置（读/写）复用功能端口，否则可能会使转换精度下降。

备注 LSB：最低有效位
 FSR：满刻度范围

(11) D/A 转换器

(TA=−40 至+85℃、VDD=EVDD=UVDD=AVREF0=AVREF1、3.0 V ≤ AVREF1 ≤ 3.6 V、VSS=AVSS=0 V、CL=50pF)

参数	符号	条件	MIN.	TYP.	MAX.	单位
分辨率					8	位
总误差 ^{注 1}		R=2 MΩ			±1.2	%FSR
设置时间		C=20 pF			3	μs
输出电阻器	Ro	输出数据 55H		6.42		kΩ
参考电压	AVREF1		3.0		3.6	V
AVREF1 电流 ^{注 2}	AIREF1	D/A 转换进行中		1	2.5	mA
		D/A 转换停止			5	μA

注 1. 量化误差除外（±0.5 %LSB）。

2. D/A 转换器 1 通道的值

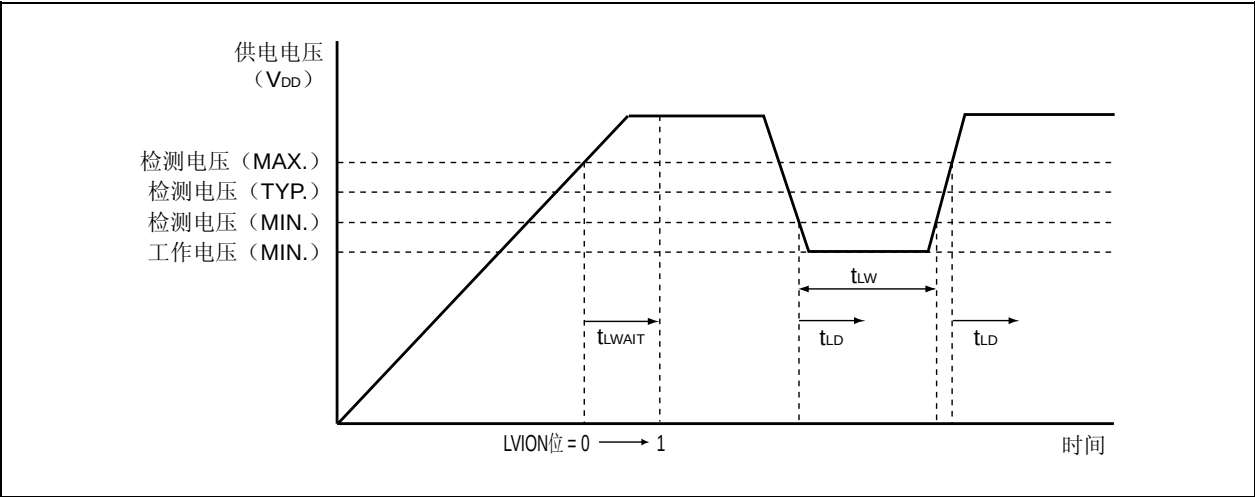
备注 R 是输出引脚负载电阻，C 是输出引脚负载电容。

(12) LVI 电路特性

(TA=−40 至+85℃、VDD=EVDD=UVDD=AVREF0=AVREF1、VSS=AVSS=0 V、CL=50 pF)

参数	符号	条件	MIN.	TYP.	MAX.	单位
检测电压	VLVI0		2.85	2.95	3.05	V
响应时间 ^注	tLD	VDD 达到 VLVI0（MAX.）后，或 VDD 下降到 VLVI0（MAX.）后		0.2	2.0	ms
最小脉冲宽度	tLW		0.2			ms
参考电压稳定等待时间	tLWAIT	VDD 达到 2.85 V（最小值）后		0.1	0.2	ms

注 检测到检测电压和输出一个中断或复位信号所需的时间。

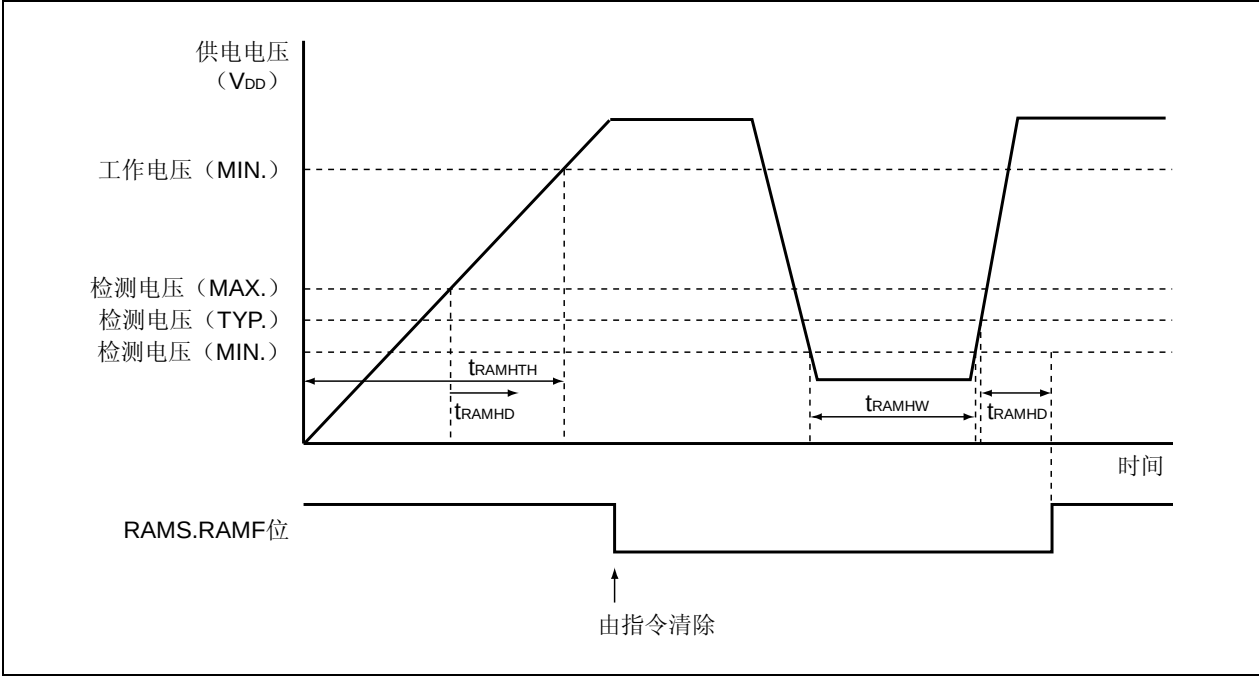


(13) RAM 保持检测

($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = EV_{DD} = UV_{DD} = AV_{REF0} = AV_{REF1}$ 、 $V_{SS} = AV_{SS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

参数	符号	条件	MIN.	TYP.	MAX.	单位
检测电压	V_{RAMH}		1.9	2.0	2.1	V
供电电压上升时间	t_{RAMHTH}	$V_{DD} = 0$ 至 2.85 V	0.002			ms
响应时间 ^注	t_{RAMHD}	V_{DD} 达到 2.1 V 后		0.2	3.0	ms
最小脉冲宽度	t_{RAMHW}		0.2			ms

注 检测到检测电压和设置 RAMS.RAMF 位所需时间。



33.9 Flash存储器编程特性

($T_A = -40$ 至 $+85^{\circ}\text{C}$ 、 $V_{DD} = EV_{DD} = UV_{DD} = AV_{REF0} = AV_{REF1}$ 、 $V_{SS} = AV_{SS} = 0\text{ V}$ 、 $C_L = 50\text{ pF}$)

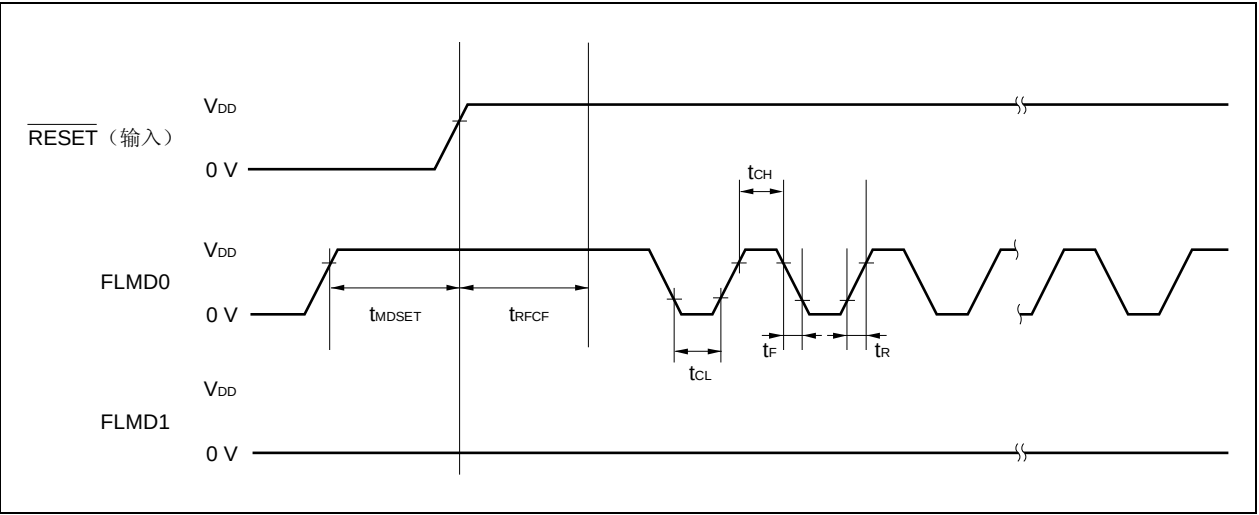
(1) 基本特性

参数	符号	条件	MIN.	TYP.	MAX.	单位
工作频率	f_{CPU}		24		48	MHz
供电电压	V_{DD}		2.85		3.6	V
重写次数	C_{WRT}				100	次数
编程温度	t_{PRG}		-40		+85	$^{\circ}\text{C}$

(2) 串行写入操作特性

参数	符号	条件	MIN.	TYP.	MAX.	单位
FLMD0、FLMD1 建立时间	t_{MDSET}		2		3000	ms
自 $\overline{\text{RESET}}\uparrow$ 的 FLMD0 计数开始时间	t_{RFCF}	$f_x = 3$ 至 6 MHz	800			μs
FLMD0 计数器高电平宽度/低电平宽度	$t_{\text{CH}}/t_{\text{CL}}$		10		100	μs
FLMD0 计数器上升时间/下降时间	$t_{\text{r}}/t_{\text{f}}$				1	μs

Flash 写入模式设置时序



(3) 编程特性

参数	符号	条件	MIN.	TYP.	MAX.	单位
片擦除时间		f _{xx} =48 MHz、批擦除		105		ms
每 256 字节的写入时间		f _{xx} =48 MHz		2.0		ms
块内部校验时间		f _{xx} =48 MHz		10		ms
块空白检查时间		f _{xx} =48 MHz		0.5		ms
Flash 存储器信息设置时间		f _{xx} =48 MHz		30		ms

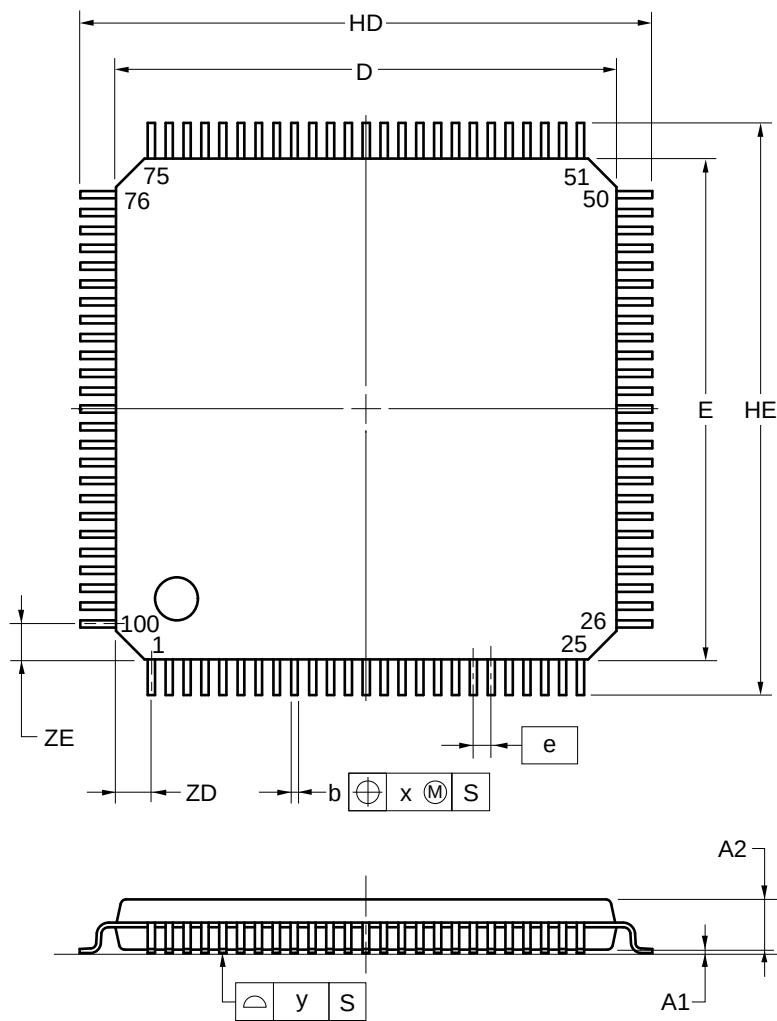
注意事项 当最初对出厂的产品进行写入时，“擦除到写入”操作和“只写入”操作，都按一次重写计数。

例如（P：写入，E：擦除）

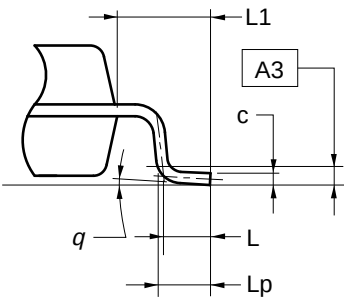
出厂产品 → P → E → P → E → P: 3 次重写
出厂产品 → E → P → E → P → E → P: 3 次重写

备注 块的尺寸为 4 KB。

100-引脚LQFP塑料封装（密间距）（14x14）



引线末端细节

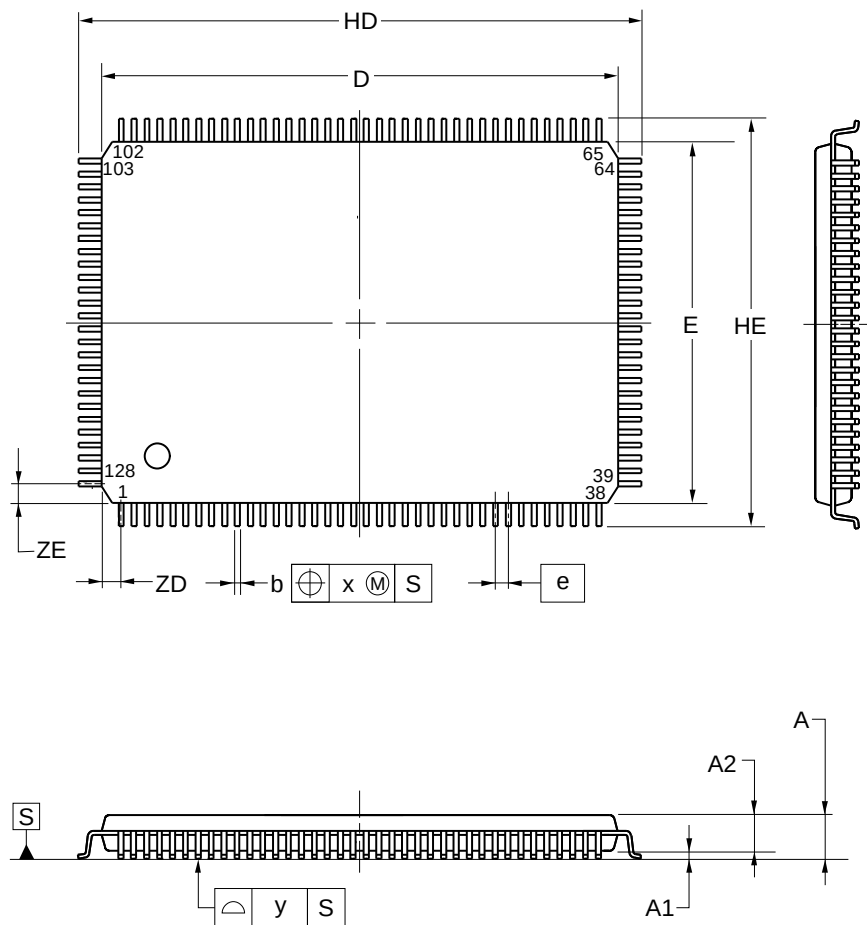


(单位: mm)	
项目	尺寸
D	14.00±0.20
E	14.00±0.20
HD	16.00±0.20
HE	16.00±0.20
A	1.60 MAX.
A1	0.10±0.05
A2	1.40±0.05
A3	0.25
b	0.20 ^{+0.07} _{-0.03}
c	0.125 ^{+0.075} _{-0.025}
L	0.50
Lp	0.60±0.15
L1	1.00±0.20
q	3° ^{+5°} _{-3°}
e	0.50
x	0.08
y	0.08
ZD	1.00
ZE	1.00

P100GC-50-UEU-1

© NEC Electronics Corporation 2007

128-引脚LQFP塑料封装（密间距）（14x20）



注
在最大限度材料条件下，每条引线中心线
位于其真实位置的0.08mm内。

(单位:mm)	
项目	尺寸
D	20.00±0.20
E	14.00±0.20
HD	22.00±0.20
HE	16.00±0.20
A	1.60 MAX.
A1	0.10±0.05
A2	1.40±0.05
A3	0.25
b	0.20 ^{+0.07} _{-0.03}
c	0.125 ^{+0.075} _{-0.025}
L	0.50
Lp	0.60±0.15
L1	1.00±0.20
q	3° ^{+5°} _{-3°}
e	0.50
x	0.08
y	0.08
ZD	0.75
ZE	0.75
P128GF-50-GAT	

© NEC Electronics Corporation 2006

附录 A 开发工具

以下开发工具可用来开发使用 V850ES/JG3-H 或 V850ES/JH3-H 单片机的系统。

图 A-1 显示了开发工具的构成配置。

- **支持 PC98-NX 系列**

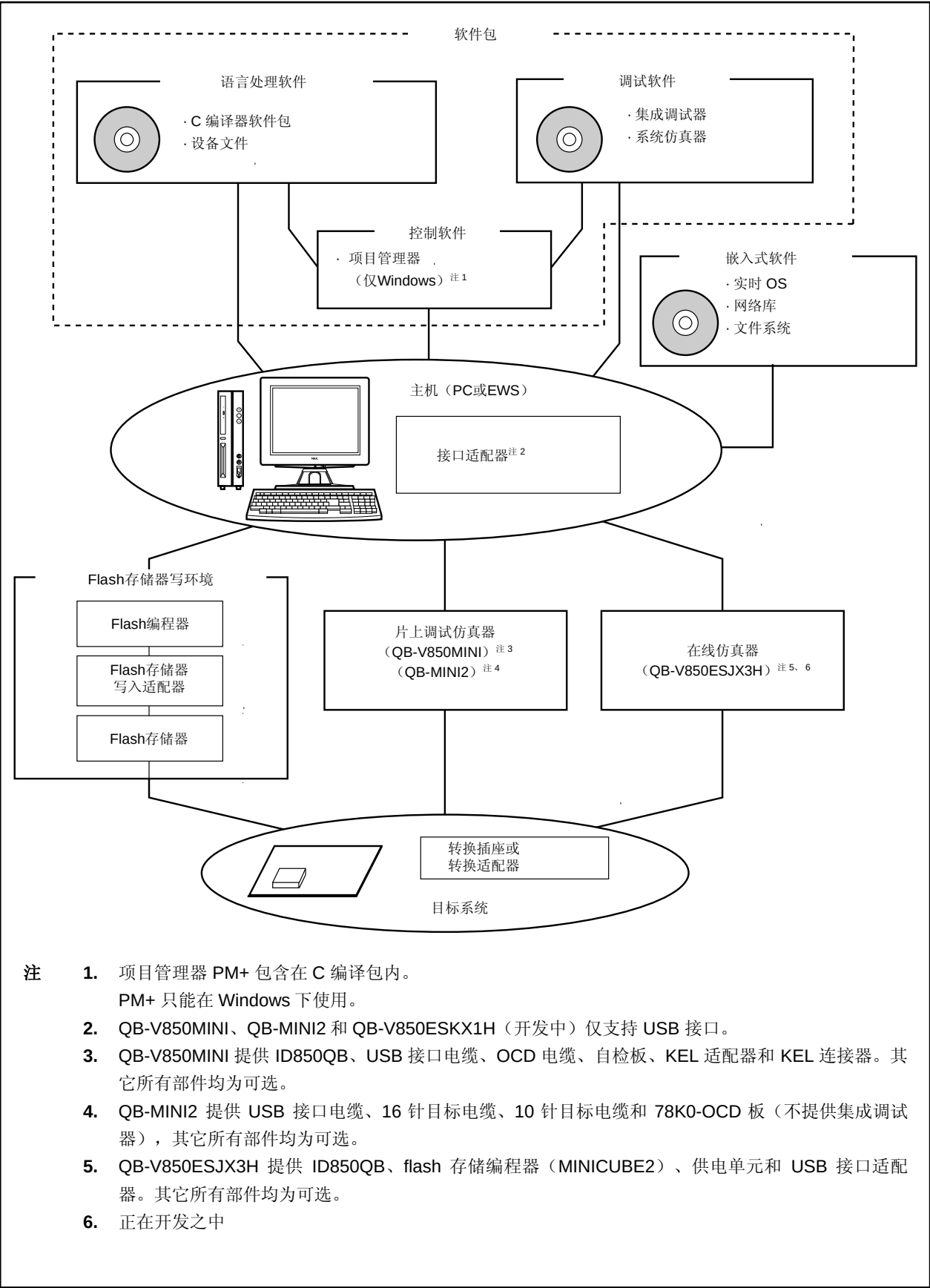
除非另行说明，IBM PC/AT™ 支持兼容的产品兼容于 PC98-NX 系列计算机。当使用 PC98-NX 系列计算机时，可参见 IBM PC/AT 兼容机说明。

- **Windows™**

除非另行说明，“Windows”是指以下操作系统：

- Windows 98、2000
- Windows Me
- Windows XP
- Windows NT™ Ver. 4.0

图 A-1. 开发工具配置



A.1 软件包

V850 微控制器的软件包 SP850	用于 V850 微控制器的通用开发工具（软件）包含在本软件包内。
	型号： μ SxxxxSP850

备注 xxxx 型号随使用的主机和操作系统的不同而异。

μ SxxxxSP850

xxxx	主机	操作系统	提供形式
AB17	PC-9800 系列	Windows（日文版）	CD-ROM
BB17	IBM PC/AT 兼容机	Windows（英文版）	

A.2 语言处理软件

CA850 C 编译包	该编译器将用 C 语言写的程序转换为微控制器执行的目标代码。本编译器由项目管理器 PM+ 启动。
	型号： μ SxxxxCA703000
DF703771 设备文件	本文件包含器件的特有信息。 该设备文件应和开发工具（CA850 或 ID850QB）配合使用。 使用的工具不同，相应使用的操作系统和主机也不同。

备注 xxxx 型号随使用的主机和操作系统的不同而异。

μ SxxxxCA703000

xxxx	主机	操作系统	提供形式
AB17	PC-9800 系列	Windows（日文版）	CD-ROM
BB17	IBM PC/AT 兼容机	Windows（英文版）	
3K17	SPARCstation™	SunOS™（Rel. 4.1.4） Solaris™（Rel. 2.5.1）	

A.3 控制软件

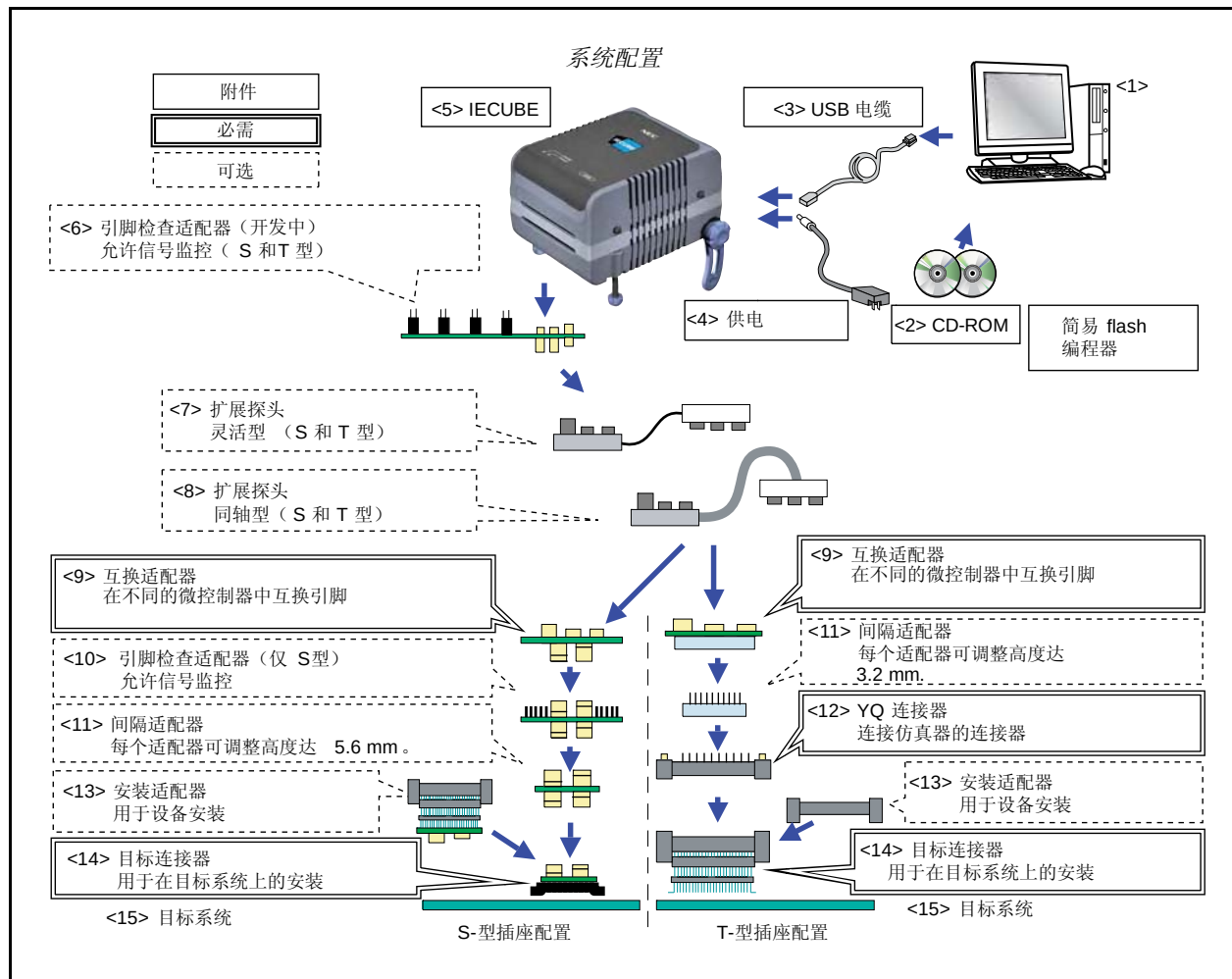
PM+ 项目管理器	这是一个控制软件，使用户可以在 Windows 环境下有效地开发用户程序。所有用户程序开发过程中的操作，诸如启动编辑器、构建以及进行调试等，都可以在 PM+ 中进行。 〈注意事项〉 PM+ 包含在 C 编译包 CA850 内。 它只能在 Windows 下工作。

A.4 调试工具（硬件）

A.4.1 当使用 IECUBE QB-V850ESJX3H^注时

当将 QB-V850ESJX3H^注 连接到主机（PC-9821 系列，PC/AT 兼容机）时，系统配置如下。即使不备有可选部件，照样可以完成连接。

图 A-2. 系统构成（使用 QB-V850ESJX3H^注时）（1/2）



注 正在开发之中

图 A-2. 系统构成（使用 QB-V850ESJX3H^{※1}时）（2/2）

- <1> 主机（PC-9821 系列，IBM-PC/AT 兼容机）
- <2> 调试器、USB 驱动、手册等等。（ID850QB 光碟、附属光碟^{※2}）
- <3> USB 接口电缆
- <4> AC 适配器。
- <5> 在线仿真器（QB-V850ESJX3H^{※1}）。
- <6> 引脚检查适配器（S 和 T 类型）（QB-144-CA-01^{※3}）（可选）
- <7> 扩展探头（灵活型）（S 和 T 型）（QB-144-EP-02S）（可选）
- <8> 扩展探头（同轴型）（S 和 T 型）（QB-144-EP-01S）（可选）
- <9> 互换适配器^{※4}（S 型：QB-100GC-EA-04S（GC 程序包），QB-128GF-EA-01S（GF 程序包），T 型：QB-100GC-EA-05T（GC 程序包），QB-128GF-EA-02T（GF 程序包））
- <10> 引脚检查适配器^{※5}（S 型：QB-100-CA-01S（GC 程序包），QB-128GF-CA-01S（GF 程序包））（可选）
- <11> 间隔适配器^{※5}（S 型：QB-100-SA-01S（GC 程序包），QB-144-SA-01S（GF 程序包），T 型：QB-100GC-YS-01T（GC 程序包），QB-128GF-YS-01T（GF 程序包））（可选）
- <12> YQ 连接器注 4（仅 T 型）（QB-100GC-YQ-01T）（GC 程序包），QB-128GF-YQ-01T（GF 程序包）
- <13> 安装适配器（S 型：QB-100GC-MA-01S（GC 程序包），QB-128GF-MA-01S（GF 程序包），T 型：QB-100GC-HQ-01T（GC 程序包），QB-128GF-HQ-01T（GF 程序包））（可选）
- <14> 目标连接器^{※4}（S 型：QB-100GC-TC-01S（GC 程序包），QB-128GF-TC-01S（GF 程序包），T 型：QB-100GC-NQ-01T（GC 程序包），QB-128GF-NQ-01T（GF 程序包））
- <15> 目标系统

- 注
- 1. 正在开发之中。
 - 2. 从 NEC 电子学网站上下载设备文件。
<http://www.necel.com/micro/ods/eng/>
 - 3. 正在开发之中。
 - 4. 根据订货码决定设备供应。
 - 当订货为 QB-V850ESJX3H-ZZZ 时
不提供互换适配器和目标连接器。
 - 当订货为 QB-V850ESJX3H-S100GC 时
提供 QB-100GC-EA-04S 和 QB-100GC-TC-01S。
 - 当订货为 QB-V850ESJX3H-S128GF 时
提供 QB-128GF-EA-01S 和 QB-128GF-TC-01S。
 - 当订货为 QB-V850ESJX3H-T100GC 时
提供 QB-100GC-EA-05T、QB-100GC-YQ-01T 和 QB-100GC-NQ-01T。
 - 当订货为 QB-V850ESJX3H-T128GF 时
提供 QB-1028GF-EA-02T、QB-128GF-YQ-01T 和 QB-128GF-NQ-01T。
 - 5. 当都使用<9>和<10>时，不必考虑<9>和<10>的次序。

<5> QB-V850ESJX3H ^注 在线仿真器	当使用 V850ES/JG3-H 或 V850ES/JH3-H 开发应用系统时，利用在线仿真器调试其硬件和软件。它支持集成调试器 ID850QB。该仿真器应该和一个供电单元和仿真探头配合使用。使用 USB 接口连接仿真器和主机。
<3> USB 接口电缆	连接主机和 QB-V850ESJX3H 的电缆。
<4> AC 适配器	替代 AC 插头，可支持 100 至 240V 的供电电源。
<9> QB-100GC-EA-04S QB-128GF-EA-01S QB-100GC-EA-05T QB-128GF-EA-02T 互换适配器	完成引脚转换的适配器。 • QB-100GC-EA-04S: 100-引脚 LQFP 塑料封装 (GC-UEU 型) • QB-128GF-EA-01S: 128-引脚 LQFP 塑料封装 (GF-GAT 型) • QB-100GC-EA-05T: 100-引脚 LQFP 塑料封装 (GC-UEU 型) • QB-128GF-EA-02T: 128-引脚 LQFP 塑料封装 (GF-GAT 型)
<10> QB-100-CA-01S QB-128-CA-01S (仅 S 型) 引脚检查适配器	使用了示波器等仪器在观察波形时使用的适配器。 • QB-100-CA-01S: 100-引脚 LQFP 塑料封装 (GC-UEU 型) • QB-128-CA-01S: 128-引脚 LQFP 塑料封装 (GF-GAT 型)
<11> QB-100-SA-01S QB-144-SA-01S QB-100GC-YS-01T QB-128GF-YS-01T 间隔适配器	调节高度的适配器。 • QB-100-SA-01S: 100-引脚 LQFP 塑料封装 (GC-UEU 型) • QB-144-SA-01S: 128-引脚 LQFP 塑料封装 (GF-GAT 型) • QB-100GC-YS-01T: 100-引脚 LQFP 塑料封装 (GC-UEU 型) • QB-128GF-YS-01T: 128-引脚 LQFP 塑料封装 (GF-GAT 型)
<12> QB-100GC-YQ-01T QB-128GF-YQ-01T (仅 T 型) YQ 连接器	转换适配器用于连接目标连接器和互换适配器。 • QB-100GC-YQ-01T: 100-引脚 LQFP 塑料封装 (GC-UEU 型) • QB-128GF-YQ-01T: 128-引脚 LQFP 塑料封装 (GF-GAT 型)
<13> QB-100GC-MA-01S QB-128GF-MA-01S QB-100GC-HQ-01T QB-128GF-HQ-01T 安装适配器	在插座上安装 V850ES/JG3-H 或 V850ES/JH3-H 的适配器。 • QB-100GC-MA-01S: 100-引脚 LQFP 塑料封装 (GC-UEU 型) • QB-128GF-MA-01S: 128-引脚 LQFP 塑料封装 (GF-GAT 型) • QB-100GC-HQ-01T: 100-引脚 LQFP 塑料封装 (GC-UEU 型) • QB-128GF-HQ-01T: 128-引脚 LQFP 塑料封装 (GF-GAT 型)
<14> QB-100GC-TC-01S QB-128GF-TC-01S QB-100GC-NQ-01T QB-128GF-NQ-01T 目标连接器	用于焊接到目标系统上的连接器。 • QB-100GC-TC-01S: 100-引脚 LQFP 塑料封装 (GC-UEU 型) • QB-128GF-TC-01S: 128-引脚 LQFP 塑料封装 (GF-GAT 型) • QB-100GC-NQ-01T: 100-引脚 LQFP 塑料封装 (GC-UEU 型) • QB-128GF-NQ-01T: 128-引脚 LQFP 塑料封装 (GF-GAT 型)

注 QB-V850ESJX3H 提供一个供电单元、USB 接口电缆、Flash 存储编程器 (MINICUBE2)。同时也提供集成调试器 ID850QB 作为控制软件。

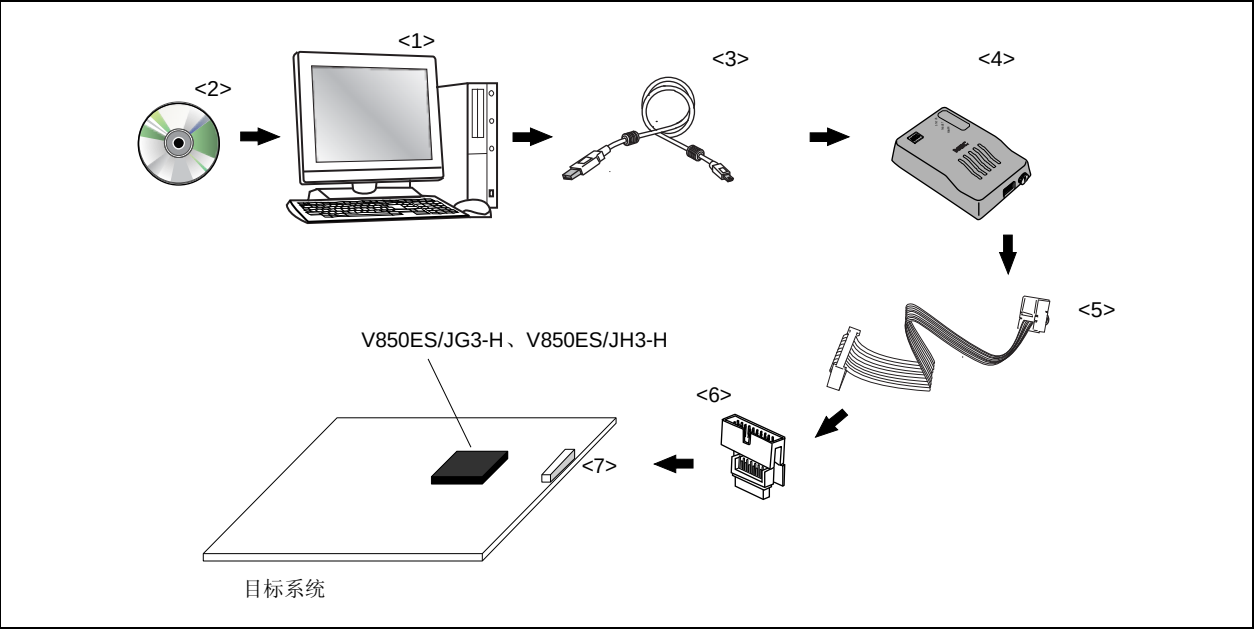
备注 尖括号内的数字对应图 A-2 中的数字。

A.4.2 使用MINICUBE QB-V850MINI时

(1) 使用 MINICUBE 进行片上仿真

当将 MINICUBE 连接到主机（PC-9821 系列、PC/AT 兼容机）时，系统构成如下所示。

图 A-3. 片上仿真系统构成



<1> 主机	配有 USB 端口的 PC
<2> CD-ROM ^{※1}	在 CD-ROM 内有诸如集成调试器 ID850QB、N-Wire 检查器、设备驱动和文档等内容。与 MINICUBE 一并提供。
<3> USB 接口电缆	USB 电缆用来连接主机和 MINICUBE。与 MINICUBE 一并提供。电缆的长度大约 2m。
<4> MINICUBE 片上调试仿真器	当使用 V850ES/JG3-H 或 V850ES/JH3-H 开发应用系统时，利用该片上调试仿真器调试其硬件和软件。它支持集成调试器 ID850QB。
<5> OCD 电缆	连接 MINICUBE 和目标系统的电缆。 它和 MINICUBE 一并提供。电缆的长度大约 20cm。
<6> 连接器转换板 KEL 适配器	该转换板和 MINICUBE 一并提供。
<7> MINICUBE 连接器 KEL 连接器 ^{※2}	8830E-026-170S（与 MINICUBE 一并提供） 8830E-026-170L（单独出售）

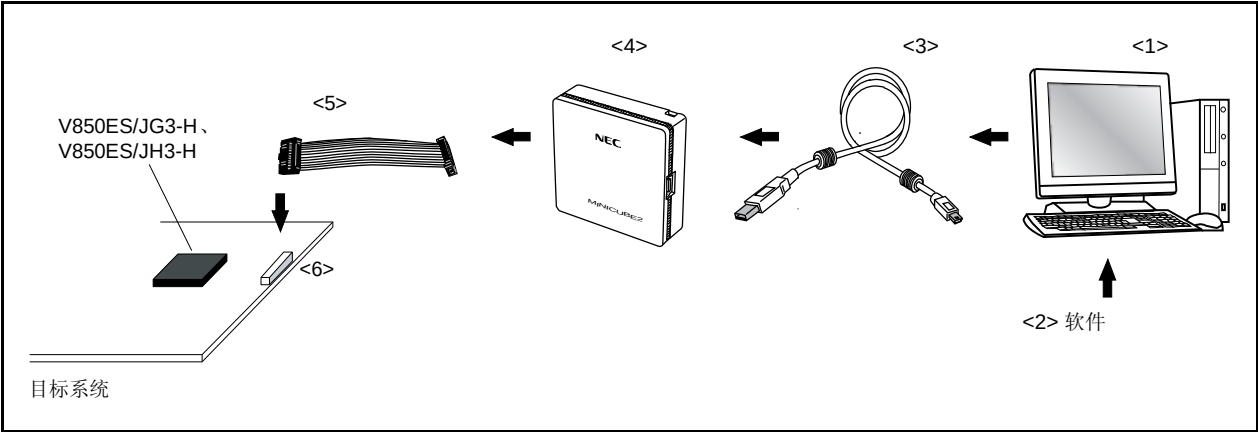
注 1. 从 NEC 电子学网站上下载设备文件。
<http://www.necel.com/micro/ods/eng/index.html>
2. KEL 公司产品

备注 尖括号内的数字对应图 A-3 中的数字。

A.4.3 使用MINICUBE2 QB-MINI2 时

当将 MINICUBE2 连接到主机（PC-9821 系列， PC/AT 兼容机）时，系统配置如下。

图 A-4. 在线仿真系统的系统构成



<1> 主机	配有 USB 端口的 PC
<2> 软件	集成调试器 ID850QB、设备文件等。 从 NEC 电子学网站下载设备文件。 http://www.necel.com/micro/ods/eng/
<3> USB 接口电缆	USB 电缆用来连接主机和 MINICUBE。它和 MINICUBE 一并提供。电缆的长度大约是 2m。
<4> MINICUBE2 片上调试仿真器	当使用 V850ES/JG3-H 或 V850ES/JH3-H 开发应用系统时，使用该片上调试仿真器调试其硬件和软件。它支持集成调试器 ID850QB。
<5> 16-针目标电缆	连接 MINICUBE2 和目标系统的电缆。 它和 MINICUBE 一并提供。电缆的长度大约为 15cm。
<6> 目标连接器（单独出售）	使用一个 2.54mm 间距的 16-引脚通用连接器。

备注 尖括号内的数字对应图 A-4 中的数字。

A.5 调试工具（软件）

ID850QB 集成调试器	本调试器支持 V850 微控制器的在线仿真。ID850QB 是基于 Windows 的软件。它具有改进的 C-兼容调试功能，并使用一个集成窗口功能显示源程序追踪执行结果，该功能与源程序、反汇编显示和追踪结果的存储显示有关。 该调试器应该配合设备文件使用。
	型号： μ Sxxxx ID703000-QB（ID850QB）
V850ES/Jx2 的 SM+（开发中） 系统模拟器	此为 V850 微控制器的系统模拟器。V850ES/Jx2 的 SM+ 是基于 Windows 的软件。当在主机上模拟目标系统的操作时，用来实现在 C 源程序层面或汇编源程序层面的调试。 V850ES/Jx2 的 SM+ 允许在独立于硬件开发的基础上进行应用逻辑测试和性能测试，从而提供较高的开发效率和软件质量。 SM+ 应该配合设备文件使用（单独出售）。
	型号： μ SxxxxSM703724-B

备注 xxxx 型号随使用的主机和操作系统的不同而异。

μ SxxxxID703000-QB
 μ SxxxxSM703724-B

xxxx	主机	操作系统	提供形式
AB17	PC-9800 系列	Windows（日文版）	CD-ROM
BB17	IBM PC/AT 兼容机	Windows（英文版）	

A.6 嵌入式软件

RX850、RX850 Pro 实时 OS	RX850 和 RX850 Pro 为实时操作系统，遵从 μ ITRON 3.0 规范。 提供一个工具（配置器），用来产生多重信息表。 相比 RX850，RX850 Pro 具有更多功能。
	型号： μ SxxxxRX703000- $\Delta\Delta\Delta\Delta$ (RX850) μ SxxxxRX703100- $\Delta\Delta\Delta\Delta$ (RX850 Pro)
Applilet [®] （开发中）	这是一个驱动配置器，可以自动地为 V850ES/JG3-H 和 V850ES/JH3-H 生成示例程序。
RX-FS850 （文件系统）	这是 FAT 文件系统功能。 该文件系统支持 CD-ROM 文件系统功能。 本文件系统为实时操作系统 OS RX850 Pro 所用。

注意事项 欲购买 RX850 或 RX850 Pro 实时操作系统，首先填写购买申请表并签署授权许可文件。

备注 型号中的xxxx和 $\Delta\Delta\Delta\Delta$ 随使用的主机和操作系统的不同而异。

μ SxxxxRX703000- $\Delta\Delta\Delta\Delta$

μ SxxxxRX703100- $\Delta\Delta\Delta\Delta$

$\Delta\Delta\Delta\Delta$	产品概要	批量生产的最大数目
001	评估对象	不要用于批量生产的产品
100K	批量生产对象	0.1 百万单位
001M		1 百万单位
010M		10 百万单位
S01	源程序	批量生产的目标源程序

xxxx	主机	操作系统	提供形式
AB17	PC-9800 系列	Windows（日文版）	CD-ROM
BB17	IBM PC/AT 兼容机	Windows（英文版）	
3K17	SPARCstation	Solaris（Rel. 2.5.1）	

A.7 Flash存储器写入工具

Flashpro IV（型号：PG-FP4） Flashpro V（型号：PG-FP5） Flash 编程器	专用于内部 Flash 存储器微控制器的 Flash 编程器。
QB-MINI2（MINICUBE2）	具有可编程功能的片上调试仿真器。
FA-100GC-UEU-B FA-128GF-GAT-B Flash 存储器写入适配器	Flash 存储器写入适配器（无线）用来连接 Flashpro IV、Flashpro V 等等。 • FA-100GC-UEU-B：100-引脚 LQFP 塑料封装（GC-UEU 型） • FA-128GF-GAT-B：100-引脚 LQFP 塑料封装（GF-GAT 型）

备注 FA-100GC-UEU-B 和 FA-128GF-GAT-B 为 Naito Densei Machida Mfg. Co., Ltd.的产品。
电话：+81-42-750-4172

附录B V850ES/Jx3-H和V850ES/Jx3 之间的主要区别

表 B-1. V850ES/Jx3-H 和 V850ES/Jx3 之间的主要区别

主要区别	V850ES/Jx3-H	V850ES/Jx3
最短指令执行时间	20.8 ns（在 48 MHz 下工作）	31.25 ns（在 32 MHz 下工作）
16-位定时器	TAA（TMP 的高性能型）	TMP
	TAB（TMQ 的高性能型）	TMQ
	TMT（编码器定时器）	无
钟表定时器功能	RTC （内含硬件计数器）	WT （不含硬件计数器）
马达控制功能	可用	无
USB 接口	功能	无
异步串行接口	UARTC（UARTA 的高性能型）	UARTA
封装	100-引脚 LQFP 封装 128-引脚 LQFP 封装	100-引脚 LQFP 封装 144-引脚 LQFP 封装

附录 C 寄存器索引

(1/14)

符号	名称	单元	页码
ADA0CR0	A/D转换结果寄存器 0	ADC	693
ADA0CR0H	A/D转换结果寄存器 0H	ADC	693
ADA0CR1	A/D转换结果寄存器 1	ADC	693
ADA0CR1H	A/D转换结果寄存器 1H	ADC	693
ADA0CR2	A/D转换结果寄存器 2	ADC	693
ADA0CR2H	A/D转换结果寄存器 2H	ADC	693
ADA0CR3	A/D转换结果寄存器 3	ADC	693
ADA0CR3H	A/D转换结果寄存器 3H	ADC	693
ADA0CR4	A/D转换结果寄存器 4	ADC	693
ADA0CR4H	A/D转换结果寄存器 4H	ADC	693
ADA0CR5	A/D转换结果寄存器 5	ADC	693
ADA0CR5H	A/D转换结果寄存器 5H	ADC	693
ADA0CR6	A/D转换结果寄存器 6	ADC	693
ADA0CR6H	A/D转换结果寄存器 6H	ADC	693
ADA0CR7	A/D转换结果寄存器 7	ADC	693
ADA0CR7H	A/D转换结果寄存器 7H	ADC	693
ADA0CR8	A/D转换结果寄存器 8	ADC	693
ADA0CR8H	A/D转换结果寄存器 8H	ADC	693
ADA0CR9	A/D转换结果寄存器 9	ADC	693
ADA0CR9H	A/D转换结果寄存器 9H	ADC	693
ADA0CR10	A/D转换结果寄存器 10	ADC	693
ADA0CR10H	A/D转换结果寄存器 10H	ADC	693
ADA0CR11	A/D转换结果寄存器 11	ADC	693
ADA0CR11H	A/D转换结果寄存器 11H	ADC	693
ADA0M0	A/D转换器模式寄存器 0	ADC	686
ADA0M1	A/D转换器模式寄存器 1	ADC	688
ADA0M2	A/D转换器模式寄存器 2	ADC	691
ADA0PFM	掉电比较模式寄存器	ADC	695
ADA0PFT	掉电比较阈值寄存器	ADC	696
ADA0S	A/D转换器通道指定寄存器	ADC	692
ADIC	中断控制寄存器	INTC	1273
AWC	地址等待控制寄存器	BCU	195
BCC	总线周期控制寄存器	BCU	196
BPC	外围I/O区域选择控制寄存器	BCU	94
BSC	总线宽度配置寄存器	BCU	184
CCLS	CPU工作时钟状态寄存器	CG	210
CF0CTL0	CSIF0控制寄存器 0	CSIF	772
CF0CTL1	CSIF0控制寄存器 1	CSIF	775
CF0CTL2	CSIF0控制寄存器 2	CSIF	776

(2/14)

符号	名称	单元	页码
CF0RIC	中断控制寄存器	INTC	1273
CF0RX	CSIF0接收数据寄存器	CSIF	770
CF0RXL	CSIF0接收数据寄存器 L	CSIF	770
CF0STR	CSIF0状态寄存器	CSIF	778
CF0TIC	中断控制寄存器	INTC	1273
CF0TX	CSIF0发送数据寄存器	CSIF	771
CF0TXL	CSIF0发送数据寄存器 L	CSIF	771
CF1CTL0	CSIF1控制寄存器 0	CSIF	772
CF1CTL1	CSIF1控制寄存器 1	CSIF	775
CF1CTL2	CSIF1控制寄存器 2	CSIF	776
CF1RIC	中断控制寄存器	INTC	1273
CF1RX	CSIF1接收数据寄存器	CSIF	770
CF1RXL	CSIF1接收数据寄存器 L	CSIF	770
CF1STR	CSIF1状态寄存器	CSIF	778
CF1TIC	中断控制寄存器	INTC	1273
CF1TX	CSIF1发送数据寄存器	CSIF	771
CF1TXL	CSIF1发送数据寄存器 L	CSIF	771
CF2CTL0	CSIF2控制寄存器 0	CSIF	772
CF2CTL1	CSIF2控制寄存器 1	CSIF	775
CF2CTL2	CSIF2控制寄存器 2	CSIF	776
CF2RIC	中断控制寄存器	INTC	1273
CF2RX	CSIF2接收数据寄存器	CSIF	770
CF2RXL	CSIF2接收数据寄存器 L	CSIF	770
CF2STR	CSIF2状态寄存器	CSIF	778
CF2TIC	中断控制寄存器	INTC	1273
CF2TX	CSIF2发送数据寄存器	CSIF	771
CF2TXL	CSIF2发送数据寄存器 L	CSIF	771
CF3CTL0	CSIF3控制寄存器 0	CSIF	772
CF3CTL1	CSIF3控制寄存器 1	CSIF	775
CF3CTL2	CSIF3控制寄存器 2	CSIF	776
CF3RIC	中断控制寄存器	INTC	1273
CF3RX	CSIF3接收数据寄存器	CSIF	770
CF3RXL	CSIF3接收数据寄存器 L	CSIF	770
CF3STR	CSIF3状态寄存器	CSIF	778
CF3TIC	中断控制寄存器	INTC	1273
CF3TX	CSIF3发送数据寄存器	CSIF	771
CF3TXL	CSIF3发送数据寄存器 L	CSIF	771
CF4CTL0	CSIF4控制寄存器 0	CSIF	772
CF4CTL1	CSIF4控制寄存器 1	CSIF	775
CF4CTL2	CSIF4控制寄存器 2	CSIF	776
CF4RIC	中断控制寄存器	INTC	1273
CF4RX	CSIF4接收数据寄存器	CSIF	770
CF4RXL	CSIF4接收数据寄存器 L	CSIF	770
CF4STR	CSIF4状态寄存器	CSIF	778

(3/14)

符号	名称	单元	页码
CF4TIC	中断控制寄存器	INTC	1273
CF4TX	CSIF4发送数据寄存器	CSIF	771
CF4TXL	CSIF4发送数据寄存器 L	CSIF	771
CKC	时钟控制寄存器	CG	213
CLM	时钟监控模式寄存器	CLM	1332
CRCD	CRC数据寄存器	CRC	1343
CRCIN	CRC输入寄存器	CRC	1343
CTBP	CALLT基址指针	CPU	65
CTPC	CALLT执行状态保存寄存器	CPU	64
CTPSW	CALLT执行状态保存寄存器	CPU	64
DA0CS0	D/A转换值设定寄存器 0	DAC	720
DA0CS1	D/A转换值设定寄存器 1	DAC	720
DA0M	D/A转换器模式寄存器	DAC	719
DADC0	DMA地址控制寄存器 0	DMAC	1230
DADC1	DMA地址控制寄存器 1	DMAC	1230
DADC2	DMA地址控制寄存器 2	DMAC	1230
DADC3	DMA地址控制寄存器 3	DMAC	1230
DBC0	DMA转移计数寄存器 0	DMAC	1229
DBC1	DMA转移计数寄存器 1	DMAC	1229
DBC2	DMA转移计数寄存器 2	DMAC	1229
DBC3	DMA转移计数寄存器 3	DMAC	1229
DBPC	异常/调试陷阱保存寄存器	CPU	65
DBPSW	异常/调试陷阱保存寄存器	CPU	65
DCHC0	DMA通道控制寄存器 0	DMAC	1231
DCHC1	DMA通道控制寄存器 1	DMAC	1231
DCHC2	DMA通道控制寄存器 2	DMAC	1231
DCHC3	DMA通道控制寄存器 3	DMAC	1231
DDA0H	DMA目的地址寄存器 0H	DMAC	1228
DDA0L	DMA目的地址寄存器 0L	DMAC	1228
DDA1H	DMA目的地址寄存器 1H	DMAC	1228
DDA1L	DMA目的地址寄存器 1L	DMAC	1228
DDA2H	DMA目的地址寄存器 2H	DMAC	1228
DDA2L	DMA目的地址寄存器 2L	DMAC	1228
DDA3H	DMA目的地址寄存器 3H	DMAC	1228
DDA3L	DMA目的地址寄存器 3L	DMAC	1228
DMAIC0	中断控制寄存器	INTC	1273
DMAIC1	中断控制寄存器	INTC	1273
DMAIC2	中断控制寄存器	INTC	1273
DMAIC3	中断控制寄存器	INTC	1273
DSA0H	DMA源地址寄存器 0H	DMAC	1227
DSA0L	DMA源地址寄存器 0L	DMAC	1227
DSA1H	DMA源地址寄存器 1H	DMAC	1227
DSA1L	DMA源地址寄存器 1L	DMAC	1227

(4/14)

符号	名称	单元	页码
DSA2H	DMA源地址寄存器 2H	DMAC	1227
DSA2L	DMA源地址寄存器 2L	DMAC	1227
DSA3H	DMA源地址寄存器 3H	DMAC	1227
DSA3L	DMA源地址寄存器 3L	DMAC	1227
DTFR0	DMA触发因子寄存器 0	DMAC	1232
DTFR1	DMA触发因子寄存器 1	DMAC	1232
DTFR2	DMA触发因子寄存器 2	DMAC	1232
DTFR3	DMA触发因子寄存器 3	DMAC	1232
DWC0	数据等待控制寄存器 0	BCU	191
ECR	中断源寄存器	CPU	62
EIPC	中断状态保存寄存器	CPU	61
EIPSW	中断状态保存寄存器	CPU	61
ERRIC0	中断控制寄存器	INTC	1273
EXDRQEN	外部DMA请求允许寄存器	DMA	1235
FEPC	NMI状态保存寄存器	CPU	62
FEPSW	NMI状态保存寄存器	CPU	62
HZA0CTL0	高阻抗输出控制寄存器 0	马达	589
HZA0CTL1	高阻抗输出控制寄存器 1	马达	589
IIC0	IIC移位寄存器 0	I ² C	838
IIC1	IIC移位寄存器 1	I ² C	838
IIC2	IIC移位寄存器 2	I ² C	838
IICC0	IIC控制寄存器 0	I ² C	825
IICC1	IIC控制寄存器 1	I ² C	825
IICC2	IIC控制寄存器 2	I ² C	825
IICCL0	IIC时钟选择寄存器 0	I ² C	835
IICCL1	IIC时钟选择寄存器 1	I ² C	835
IICCL2	IIC时钟选择寄存器 2	I ² C	835
IICF0	IIC标志寄存器 0	I ² C	833
IICF1	IIC标志寄存器 1	I ² C	833
IICF2	IIC标志寄存器 2	I ² C	833
IICIC0	中断控制寄存器	INTC	1273
IICIC1	中断控制寄存器	INTC	1273
IICIC2	中断控制寄存器	INTC	1273
IICS0	IIC状态寄存器 0	I ² C	830
IICS1	IIC状态寄存器 1	I ² C	830
IICS2	IIC状态寄存器 2	I ² C	830
IICX0	IIC功能扩展寄存器 0	I ² C	836
IICX1	IIC功能扩展寄存器 1	I ² C	836
IICX2	IIC功能扩展寄存器 2	I ² C	836
IMR0	中断屏蔽寄存器 0	INTC	1277
IMR0H	中断屏蔽寄存器 0H	INTC	1277
IMR0L	中断屏蔽寄存器 0L	INTC	1277
IMR1	中断屏蔽寄存器 1	INTC	1277
IMR1H	中断屏蔽寄存器 1H	INTC	1277

(5/14)

符号	名称	单元	页码
IMR1L	中断屏蔽寄存器 1L	INTC	1277
IMR2	中断屏蔽寄存器 2	INTC	1277
IMR2H	中断屏蔽寄存器 2H	INTC	1277
IMR2L	中断屏蔽寄存器 2L	INTC	1277
IMR3	中断屏蔽寄存器 3	INTC	1277
IMR3H	中断屏蔽寄存器 3H	INTC	1277
IMR3L	中断屏蔽寄存器 3L	INTC	1277
IMR4	中断屏蔽寄存器 4	INTC	1277
IMR4H	中断屏蔽寄存器 4H	INTC	1277
IMR4L	中断屏蔽寄存器 4L	INTC	1277
IMR5	中断屏蔽寄存器 5	INTC	1277
IMR5H	中断屏蔽寄存器 5H	INTC	1277
IMR5L	中断屏蔽寄存器 5L	INTC	1277
INTF0	外部下降沿指定寄存器 0	INTC	1289
INTF2	外部下降沿指定寄存器 2	INTC	1290
INTF3	外部下降沿指定寄存器 3	INTC	1291
INTF4	外部下降沿指定寄存器 4	INTC	1292
INTF5	外部下降沿指定寄存器 5	INTC	1293
INTF9	外部下降沿指定寄存器 9	INTC	1294
INTF9H	外部下降沿指定寄存器 9H	INTC	1294
INTF9L	外部下降沿指定寄存器 9L	INTC	1294
INTNFC	噪声抑制控制寄存器	INTC	1295
INTR0	外部上升沿指定寄存器 0	INTC	1289
INTR2	外部上升沿指定寄存器 2	INTC	1290
INTR3	外部上升沿指定寄存器 3	INTC	1291
INTR4	外部上升沿指定寄存器 4	INTC	1292
INTR5	外部上升沿指定寄存器 5	INTC	1293
INTR9	外部上升沿指定寄存器 9	INTC	1294
INTR9H	外部上升沿指定寄存器 9H	INTC	1294
INTR9L	外部上升沿指定寄存器 9L	INTC	1294
ISPR	服务中优先级寄存器	INTC	1279
KRIC	中断控制寄存器	INTC	1273
KRM	按键返回模式寄存器	KR	1299
LOCKR	锁定寄存器	CG	214
LVIIC	中断控制寄存器	INTC	1273
LVIM	低电压检测寄存器	LVI	1337
OCDM	片上调试模式寄存器	DCU	1381
OCKS0	IIC时钟分频选择寄存器 0	I ² C	838
OCKS1	IIC时钟分频选择寄存器 1	I ² C	838
OSTS	振荡稳定时间选择寄存器	待机	1304
P0	端口0寄存器	端口	111
P1	端口1寄存器	端口	116
P2	端口2寄存器	端口	117
P3	端口3寄存器	端口	121

(6/14)

符号	名称	单元	页码
P4	端口4寄存器	端口	126
P5	端口5寄存器	端口	130
P6	端口6寄存器	端口	136
P7H	端口7寄存器 H	端口	140
P7L	端口7寄存器 L	端口	140
P9	端口9寄存器	端口	143
P9H	端口9寄存器 H	端口	143
P9L	端口9寄存器 L	端口	143
PC	程序计数器	CPU	59
PCC	处理器时钟控制寄存器	CG	206
PCM	端口CM寄存器	端口	152
PCS	端口CS寄存器	端口	155
PCT	端口CT寄存器	端口	157
PDH	端口DH寄存器	端口	161
PDL	端口DL寄存器	端口	163
PDLH	端口DL寄存器 H	端口	163
PDLL	端口DL寄存器 L	端口	163
PF0	端口0功能寄存器	端口	115
PF2	端口2功能寄存器	端口	120
PF3	端口3功能寄存器	端口	125
PF4	端口4功能寄存器	端口	128
PF5	端口5功能寄存器	端口	135
PF9	端口9功能寄存器	端口	151
PF9L	端口9功能寄存器 L	端口	151
PFC0	端口0功能控制寄存器	端口	114
PFC2	端口2功能控制寄存器	端口	118
PFC3	端口3功能控制寄存器	端口	123
PFC4	端口4功能控制寄存器	端口	127
PFC5	端口5功能控制寄存器	端口	133
PFC6	端口6功能控制寄存器	端口	138
PFC9	端口9功能控制寄存器	端口	146
PFC9H	端口9功能控制寄存器 H	端口	146
PFC9L	端口9功能控制寄存器 L	端口	146
PFCE0	端口0功能控制扩展寄存器	端口	115
PFCE2	端口2功能控制扩展寄存器	端口	119
PFCE3	端口3功能控制扩展寄存器	端口	123
PFCE4	端口4功能控制扩展寄存器	端口	127
PFCE5	端口5功能控制扩展寄存器	端口	133
PFCE6	端口6功能控制扩展寄存器	端口	138
PFCE9	端口9功能控制扩展寄存器	端口	147
PFCE9H	端口9功能控制扩展寄存器 H	端口	147
PFCE9L	端口9功能控制扩展寄存器 L	端口	147
PIC00	中断控制寄存器	INTC	1273
PIC01	中断控制寄存器	INTC	1273

(7/14)

符号	名称	单元	页码
PIC02	中断控制寄存器	INTC	1273
PIC03	中断控制寄存器	INTC	1273
PIC04	中断控制寄存器	INTC	1273
PIC05	中断控制寄存器	INTC	1273
PIC06	中断控制寄存器	INTC	1273
PIC07	中断控制寄存器	INTC	1273
PIC08	中断控制寄存器	INTC	1273
PIC09	中断控制寄存器	INTC	1273
PIC10	中断控制寄存器	INTC	1273
PIC11	中断控制寄存器	INTC	1273
PIC12	中断控制寄存器	INTC	1273
PIC13	中断控制寄存器	INTC	1273
PIC14	中断控制寄存器	INTC	1273
PIC15	中断控制寄存器	INTC	1273
PIC16	中断控制寄存器	INTC	1273
PIC17	中断控制寄存器	INTC	1273
PIC18	中断控制寄存器	INTC	1273
PLLCTL	PLL控制寄存器	CG	212
PLLS	PLL锁定时间指定寄存器	CG	215
PM0	端口0模式寄存器	端口	112
PM1	端口1模式寄存器	端口	116
PM2	端口2模式寄存器	端口	117
PM3	端口3模式寄存器	端口	121
PM4	端口4模式寄存器	端口	126
PM5	端口5模式寄存器	端口	131
PM6	端口6模式寄存器	端口	137
PM7H	端口7模式寄存器 H	端口	141
PM7L	端口7模式寄存器 L	端口	141
PM9	端口9模式寄存器	端口	143
PM9H	端口9模式寄存器 H	端口	143
PM9L	端口9模式寄存器 L	端口	143
PMC0	端口0模式控制寄存器	端口	113
PMC2	端口2模式控制寄存器	端口	118
PMC3	端口3模式控制寄存器	端口	122
PMC4	端口4模式控制寄存器	端口	127
PMC5	端口5模式控制寄存器	端口	132
PMC6	端口6模式控制寄存器	端口	137
PMC9	端口9模式控制寄存器	端口	144
PMC9H	端口9模式控制寄存器 H	端口	144
PMC9L	端口9模式控制寄存器 L	端口	144
PMCCM	端口CM模式控制寄存器	端口	154
PMCCS	端口CS模式控制寄存器	端口	156
PMCCT	端口CT模式控制寄存器	端口	159
PMCDH	端口DH模式控制寄存器	端口	161

(8/14)

符号	名称	单元	页码
PMCDL	端口DL模式控制寄存器	端口	164
PMCDLH	端口DL模式控制寄存器 H	端口	164
PMCDLL	端口DL模式控制寄存器 L	端口	164
PMCM	端口CM模式寄存器	端口	153
PMCS	端口CS模式寄存器	端口	155
PMCT	端口CT模式寄存器	端口	158
PMDH	端口DH模式寄存器	端口	161
PMDL	端口DL模式寄存器	端口	163
PMDLH	端口DL模式寄存器 H	端口	163
PMDLL	端口DL模式寄存器 L	端口	163
PRCMD	命令寄存器	CPU	97
PRSCM0	预分频器比较寄存器 0	BRG	657
PRSCM1	预分频器比较寄存器 1	BRG	815
PRSCM2	预分频器比较寄存器 2	BRG	815
PRSCM3	预分频器比较寄存器 3	BRG	815
PRSM0	预分频器模式寄存器 0	BRG	656
PRSM1	预分频器模式寄存器 1	BRG	814
PRSM2	预分频器模式寄存器 2	BRG	814
PRSM3	预分频器模式寄存器 3	BRG	814
PSC	节电控制寄存器	CG	1302
PSMR	节电模式寄存器	CG	1303
PSW	程序状态字	CPU	63
r0-r31	通用寄存器	CPU	59
RAMS	内部RAM数据状态寄存器	LVI	1338
RC1ALH	报警小时设定寄存器	RTC	654
RC1ALM	报警分钟设定寄存器	RTC	654
RC1ALW	报警星期设定寄存器	RTC	655
RC1CC0	实时计数控制寄存器 0	RTC	643
RC1CC1	实时计数控制寄存器 1	RTC	643
RC1CC2	实时计数控制寄存器 2	RTC	645
RC1CC3	实时计数控制寄存器 3	RTC	646
RC1DAY	日计数寄存器	RTC	650
RC1HOUR	时计数寄存器	RTC	648
RC1MIN	分计数寄存器	RTC	648
RC1MONTH	月计数寄存器	RTC	652
RC1SEC	秒计数寄存器	RTC	647
RC1SUBC	副-计数寄存器	RTC	647
RC1SUBU	钟表错误校正寄存器	RTC	653
RC1WEEK	星期计数寄存器	RTC	651
RC1YEAR	年计数寄存器	RTC	652
RCM	内部振荡模式寄存器	CG	210
RECIC0	中断控制寄存器	INTC	1273
RESF	复位源标志寄存器	复位	1323
RTBH0	实时输出缓冲寄存器 0H	RTO	677

(9/14)

符号	名称	单元	页码
RTBL0	实时输出缓冲寄存器 0L	RTO	677
RTC0IC	中断控制寄存器	INTC	1273
RTC1IC	中断控制寄存器	INTC	1273
RTC2IC	中断控制寄存器	INTC	1272
RTPC0	实时输出端口控制寄存器 0	RTO	679
RTPM0	实时输出端口模式寄存器 0	RTO	678
SELCNT0	选择器操作控制寄存器 0	定时器	331
SVA0	从地址寄存器 0	I ² C	839
SVA1	从地址寄存器 1	I ² C	839
SVA2	从地址寄存器 2	I ² C	839
SYS	系统状态寄存器	CPU	98
TAA0CCIC0	中断控制寄存器	INTC	1273
TAA0CCIC1	中断控制寄存器	INTC	1273
TAA0CCR0	TAA0捕获/比较寄存器 0	定时器	230
TAA0CCR1	TAA0捕获/比较寄存器 1	定时器	232
TAA0CNT	TAA0计数器读缓冲寄存器	定时器	234
TAA0CTL0	TAA0控制寄存器 0	定时器	221
TAA0CTL1	TAA0控制寄存器 1	定时器	222
TAA0IOC0	TAA0 I/O控制寄存器 0	定时器	224
TAA0IOC1	TAA0 I/O控制寄存器 1	定时器	225
TAA0IOC2	TAA0 I/O控制寄存器 2	定时器	226
TAA0IOC4	TAA0 I/O控制寄存器 4	定时器	227
TAA0OPT0	TAA0选项寄存器 0	定时器	228
TAA0OPT1	TAA0选项寄存器 1	定时器	229
TAA0OVIC	中断控制寄存器	INTC	1273
TAA1CCIC0	中断控制寄存器	INTC	1273
TAA1CCIC1	中断控制寄存器	INTC	1273
TAA1CCR0	TAA1捕获/比较寄存器 0	定时器	230
TAA1CCR1	TAA1捕获/比较寄存器 1	定时器	232
TAA1CNT	TAA1计数器读缓冲寄存器	定时器	234
TAA1CTL0	TAA1控制寄存器 0	定时器	221
TAA1CTL1	TAA1控制寄存器 1	定时器	222
TAA1IOC0	TAA1 I/O控制寄存器 0	定时器	224
TAA1IOC1	TAA1 I/O控制寄存器 1	定时器	225
TAA1IOC2	TAA1 I/O控制寄存器 2	定时器	226
TAA1IOC4	TAA1 I/O控制寄存器 4	定时器	227
TAA1OPT0	TAA1选项寄存器 0	定时器	228
TAA1OVIC	中断控制寄存器	INTC	1273
TAA2CCIC0	中断控制寄存器	INTC	1273
TAA2CCIC1	中断控制寄存器	INTC	1273
TAA2CCR0	TAA2捕获/比较寄存器 0	定时器	230
TAA2CCR1	TAA2捕获/比较寄存器 1	定时器	232
TAA2CNT	TAA2计数器读缓冲寄存器	定时器	234
TAA2CTL0	TAA2控制寄存器 0	定时器	221

(10/14)

符号	名称	单元	页码
TAA2CTL1	TAA2控制寄存器 1	定时器	222
TAA2IOC0	TAA2 I/O控制寄存器 0	定时器	224
TAA2IOC1	TAA2 I/O控制寄存器 1	定时器	225
TAA2IOC2	TAA2 I/O控制寄存器 2	定时器	226
TAA2IOC4	TAA2 I/O控制寄存器 4	定时器	227
TAA2OPT0	TAA2选项寄存器 0	定时器	228
TAA2OPT1	TAA2选项寄存器 1	定时器	229
TAA2OVIC	中断控制寄存器	INTC	1273
TAA3CCIC0	中断控制寄存器	INTC	1273
TAA3CCIC1	中断控制寄存器	INTC	1273
TAA3CCR0	TAA3捕获/比较寄存器 0	定时器	230
TAA3CCR1	TAA3捕获/比较寄存器 1	定时器	232
TAA3CNT	TAA3计数器读缓冲寄存器	定时器	234
TAA3CTL0	TAA3控制寄存器 0	定时器	221
TAA3CTL1	TAA3控制寄存器 1	定时器	222
TAA3IOC0	TAA3 I/O控制寄存器 0	定时器	224
TAA3IOC1	TAA3 I/O控制寄存器 1	定时器	225
TAA3IOC2	TAA3 I/O控制寄存器 2	定时器	226
TAA3IOC4	TAA3 I/O控制寄存器 4	定时器	227
TAA3OPT0	TAA3选项寄存器 0	定时器	228
TAA3OVIC	中断控制寄存器	INTC	1273
TAA4CCIC0	中断控制寄存器	INTC	1273
TAA4CCIC1	中断控制寄存器	INTC	1273
TAA4CCR0	TAA4捕获/比较寄存器 0	定时器	230
TAA4CCR1	TAA4捕获/比较寄存器 1	定时器	232
TAA4CNT	TAA4计数器读缓冲寄存器	定时器	234
TAA4CTL0	TAA4控制寄存器 0	定时器	221
TAA4CTL1	TAA4控制寄存器 1	定时器	222
TAA4OVIC	中断控制寄存器	INTC	1273
TAA5CCIC0	中断控制寄存器	INTC	1273
TAA5CCIC1	中断控制寄存器	INTC	1273
TAA5CCR0	TAA5捕获/比较寄存器 0	定时器	230
TAA5CCR1	TAA5捕获/比较寄存器 1	定时器	232
TAA5CNT	TAA5计数器读缓冲寄存器	定时器	234
TAA5CTL0	TAA5控制寄存器 0	定时器	221
TAA5CTL1	TAA5控制寄存器 1	定时器	222
TAA5IOC0	TAA5 I/O控制寄存器 0	定时器	224
TAA5IOC1	TAA5 I/O控制寄存器 1	定时器	225
TAA5IOC2	TAA5 I/O控制寄存器 2	定时器	226
TAA5IOC4	TAA5 I/O控制寄存器 4	定时器	227
TAA5OPT0	TAA5选项寄存器 0	定时器	228
TAA5OVIC	中断控制寄存器	INTC	1273
TAB0CCIC0	中断控制寄存器	INTC	1273
TAB0CCIC1	中断控制寄存器	INTC	1273

(11/14)

符号	名称	单元	页码
TAB0CCIC2	中断控制寄存器	INTC	1273
TAB0CCIC3	中断控制寄存器	INTC	1273
TAB0CCR0	TAB0捕获/比较寄存器 0	定时器	345
TAB0CCR1	TAB0捕获/比较寄存器 1	定时器	347
TAB0CCR2	TAB0捕获/比较寄存器 2	定时器	349
TAB0CCR3	TAB0捕获/比较寄存器 3	定时器	351
TAB0CNT	TAB0计数器读缓冲寄存器	定时器	353
TAB0CTL0	TAB0控制寄存器 0	定时器	338
TAB0CTL1	TAB0控制寄存器 1	定时器	339
TAB0IOC0	TAB0 I/O控制寄存器 0	定时器	340
TAB0IOC1	TAB0 I/O控制寄存器 1	定时器	341
TAB0IOC2	TAB0 I/O控制寄存器 2	定时器	342
TAB0OIC4	TAB0 I/O控制寄存器 4	定时器	343
TAB0OPT0	TAB0选项寄存器 0	定时器	344
TAB0OVIC	中断控制寄存器	INTC	1273
TAB1CCIC0	中断控制寄存器	INTC	1273
TAB1CCIC1	中断控制寄存器	INTC	1273
TAB1CCIC2	中断控制寄存器	INTC	1273
TAB1CCIC3	中断控制寄存器	INTC	1273
TAB1CCR0	TAB1捕获/比较寄存器 0	定时器	345
TAB1CCR1	TAB1捕获/比较寄存器 1	定时器	347
TAB1CCR2	TAB1捕获/比较寄存器 2	定时器	349
TAB1CCR3	TAB1捕获/比较寄存器 3	定时器	351
TAB1CNT	TAB1计数器读缓冲寄存器	定时器	353
TAB1CTL0	TAB1控制寄存器 0	定时器	338
TAB1CTL1	TAB1控制寄存器 1	定时器	339
TAB1DTC	TAB1死区时间比较寄存器 1	定时器	583
TAB1IOC0	TAB1 I/O控制寄存器 0	定时器	340
TAB1IOC1	TAB1 I/O控制寄存器 1	定时器	341
TAB1IOC2	TAB1 I/O控制寄存器 2	定时器	342
TAB1IOC3	TAB1 I/O控制寄存器 3	定时器	587
TAB1OIC4	TAB1 I/O控制寄存器 4	定时器	343
TAB1OPT0	TAB1选项寄存器 0	定时器	344
TAB1OPT1	TAB1选项寄存器 1	定时器	584
TAB1OPT2	TAB1选项寄存器 2	定时器	585
TAB1OVIC	中断控制寄存器	INTC	1273
TANFC	噪声抑制控制寄存器	定时器	235
TM0CMP0	TMM0比较寄存器 0	定时器	571
TM0CTL0	TMM0控制寄存器 0	定时器	572
TM0EQIC0	中断控制寄存器	INTC	1273
TM1CMP0	TMM1比较寄存器 0	定时器	571
TM1CTL0	TMM1控制寄存器 0	定时器	572
TM1EQIC0	中断控制寄存器	INTC	1273
TM2CMP0	TMM2比较寄存器 0	定时器	571

(12/14)

符号	名称	单元	页码
TM2CTL0	TMM2控制寄存器 0	定时器	572
TM2EQIC0	中断控制寄存器	INTC	1273
TM3CMP0	TMM3比较寄存器 0	定时器	571
TM3CTL0	TMM3控制寄存器 0	定时器	572
TM3EQIC0	中断控制寄存器	INTC	1273
TTNFC	噪声抑制控制寄存器	定时器	461
TRXIC0	中断控制寄存器	INTC	1273
TT0CCIC0	中断控制寄存器	INTC	1273
TT0CCIC1	中断控制寄存器	INTC	1273
TT0CCR0	TMT0捕获/比较寄存器 0	定时器	456
TT0CCR1	TMT0捕获/比较寄存器 1	定时器	458
TT0CNT	TMT0计数器读缓冲寄存器	定时器	460
TT0CTL0	TMT0控制寄存器 0	定时器	442
TT0CTL1	TMT0控制寄存器 1	定时器	443
TT0CTL2	TMT0控制寄存器 2	定时器	445
TT0IECIC	中断控制寄存器	INTC	1273
TT0IOC0	TMT0 I/O控制寄存器 0	定时器	447
TT0IOC1	TMT0 I/O控制寄存器 1	定时器	449
TT0IOC2	TMT0 I/O控制寄存器 2	定时器	450
TT0IOC3	TMT0 I/O控制寄存器 3	定时器	587
TT0OPT0	TMT0选项寄存器 0	定时器	453
TT0OPT1	TMT0选项寄存器 1	定时器	454
TT0OVIC	中断控制寄存器	INTC	1273
TT0TCW	TMT0计数写入寄存器	定时器	460
UC0CTL0	UARTC0控制寄存器 0	UARTC	731
UC0CTL1	UARTC0控制寄存器 1	UARTC	757
UC0CTL2	UARTC0控制寄存器 2	UARTC	758
UC0OPT0	UARTC0选项控制寄存器 0	UARTC	733
UC0OPT1	UARTC0选项控制寄存器 1	UARTC	735
UC0RIC	中断控制寄存器	INTC	1273
UC0RX	UARTC0接收数据寄存器	UARTC	739
UC0RXL	UARTC0接收数据寄存器 L	UARTC	739
UC0STR	UARTC0状态寄存器	UARTC	737
UC0TIC	中断控制寄存器	INTC	1273
UC0TX	UARTC0发送数据寄存器	UARTC	740
UC0TXL	UARTC0发送数据寄存器 L	UARTC	740
UC1CTL0	UARTC1控制寄存器 0	UARTC	731
UC1CTL1	UARTC1控制寄存器 1	UARTC	757
UC1CTL2	UARTC1控制寄存器 2	UARTC	758
UC1OPT0	UARTC1选项控制寄存器 0	UARTC	733
UC1OPT1	UARTC1选项控制寄存器 1	UARTC	735
UC1RIC	中断控制寄存器	INTC	1273
UC1RX	UARTC1接收数据寄存器	UARTC	739
UC1RXL	UARTC1接收数据寄存器 L	UARTC	739

(13/14)

符号	名称	单元	页码
UC1STR	UARTC1状态寄存器	UARTC	737
UC1TIC	中断控制寄存器	INTC	1273
UC1TX	UARTC1发送数据寄存器	UARTC	740
UC1TXL	UARTC1发送数据寄存器 L	UARTC	740
UC2CTL0	UARTC2控制寄存器 0	UARTC	731
UC2CTL1	UARTC2控制寄存器 1	UARTC	757
UC2CTL2	UARTC2控制寄存器 2	UARTC	758
UC2OPT0	UARTC2选项控制寄存器 0	UARTC	733
UC2OPT1	UARTC2选项控制寄存器 1	UARTC	735
UC2RIC	中断控制寄存器	INTC	1273
UC2RX	UARTC2接收数据寄存器	UARTC	739
UC2RXL	UARTC2接收数据寄存器 L	UARTC	739
UC2STR	UARTC2状态寄存器	UARTC	737
UC2TIC	中断控制寄存器	INTC	1273
UC2TX	UARTC2发送数据寄存器	UARTC	740
UC2TXL	UARTC2发送数据寄存器 L	UARTC	740
UC3CTL0	UARTC3控制寄存器 0	UARTC	731
UC3CTL1	UARTC3控制寄存器 1	UARTC	757
UC3CTL2	UARTC3控制寄存器 2	UARTC	758
UC3OPT0	UARTC3选项控制寄存器 0	UARTC	733
UC3OPT1	UARTC3选项控制寄存器 1	UARTC	735
UC3RIC	中断控制寄存器	INTC	1273
UC3RX	UARTC3接收数据寄存器	UARTC	739
UC3RXL	UARTC3接收数据寄存器 L	UARTC	739
UC3STR	UARTC3状态寄存器	UARTC	737
UC3TIC	中断控制寄存器	INTC	1273
UC3TX	UARTC3发送数据寄存器	UARTC	740
UC3TXL	UARTC3发送数据寄存器 L	UARTC	740
UC4CTL0	UARTC4控制寄存器 0	UARTC	731
UC4CTL1	UARTC4控制寄存器 1	UARTC	757
UC4CTL2	UARTC4控制寄存器 2	UARTC	758
UC4OPT0	UARTC4选项控制寄存器 0	UARTC	733
UC4OPT1	UARTC4选项控制寄存器 1	UARTC	735
UC4RIC	中断控制寄存器	INTC	1273
UC4RX	UARTC4接收数据寄存器	UARTC	739
UC4RXL	UARTC4接收数据寄存器 L	UARTC	739
UC4STR	UARTC4状态寄存器	UARTC	737
UC4TIC	中断控制寄存器	INTC	1273
UC4TX	UARTC4发送数据寄存器	UARTC	740
UC4TXL	UARTC4发送数据寄存器 L	UARTC	740
UCKSEL	USB时钟选择寄存器	USBF	1053
UFCKMSK	USB功能控制寄存器	USBF	1053
UFIC0	中断控制寄存器	INTC	1273
UFIC1	中断控制寄存器	INTC	1273

(14/14)

符号	名称	单元	页码
UHCKMSK	USB功能选择寄存器	USBF	1054
VSWC	系统等待控制寄存器	CPU	99
WDTE	看门狗定时器使能寄存器	WDT	674
WDTM2	看门狗定时器模式寄存器 2	WDT	672
WUPIC0	中断控制寄存器	INTC	1273

附录 D 指令集列表

D.1 习惯用法

(1) 用于描述操作数的寄存器符号

寄存器符号	解释
reg1	通用寄存器：用作源寄存器。
reg2	通用寄存器：主要用作目的寄存器，在某指令里也用作源寄存器。
reg3	通用寄存器：主要用作存储除法结果的余数，以及乘法结果的高 32 位。
bit#3	指明位号的 3 位数据。
immX	X 位立即数。
dispX	X 位置换数据。
regID	系统寄存器号。
vector	用来指明陷阱向量的 5 位数据 (00H 至 1FH)。
cccc	用来显示状态码的 4 位数据。
sp	堆栈指针 (r3)。
ep	元指针 (r30)。
listX	X 项寄存器列表。

(2) 用于描述操作码的寄存器符号

寄存器符号	解释
R	指定 reg1 或 regID 的 1 位代码数据。
r	指定 reg2 的 1 位代码数据。
w	指定 reg3 的 1 位代码数据。
d	1 位置换数据。
l	1 位立即数（指示立即数的高位）。
i	1 位立即数。
cccc	显示条件码的 4 位数据。
CCCC	显示 Bcond 指令条件码的 4 位数据。
bbb	指定位号的 3 位数据。
L	在寄存器列表中指示编程寄存器的 1 位数据。

(3) 用于操作的寄存器符号

寄存器符号	解释
←	为之输入
GR []	通用寄存器
SR []	系统寄存器
zero-extend (n)	用零扩展 n 直至满字长。
sign-extend (n)	用符号扩展 n 直至满字长。
load-memory (a, b)	从地址 a 起读取存储长度为 b 的数据。
store-memory (a, b, c)	向地址 c 起存储长度为 a 的字节写入数据 b。
load-memory-bit (a, b)	从地址 a 载入位 b。
store-memory-bit (a, b, c)	写入 c 至地址 a 的 b 位。
saturated (n)	执行 n 的饱和运算 (n 为 2 的补码)。 如此的话, 结果如下处理: n ≥ 7FFFFFFFH, 令其为 7FFFFFFFH。 n ≤ 80000000H, 令其为 80000000H。
result	反映标志的结果。
Byte	字节 (8 位)
Halfword	半字 (16 位)
Word	字 (32 位)
+	加
−	减
	位并拼
×	乘
÷	除
%	除法结果的余数
AND	逻辑乘
OR	逻辑加
XOR	逻辑异或
NOT	逻辑非
logically shift left by	逻辑左移
logically shift right by	逻辑右移
arithmetically shift right by	算术右移

(4) 用于执行时钟的寄存器符号

寄存器符号	解释
i	如果执行第一条指令后紧接着执行另一条指令 (流式执行)。
r	如果执行第一条指令后紧接着重复执行相同指令 (重复执行)。
l	如果紧接之后的指令执行使用当前指令执行结果 (等待执行)。

(5) 用于标志操作的寄存器符号

标识符	说明
(Blank)	无改变。
0	清为 0。
X	根据结果置位或清 0。
R	恢复先前的保存值。

(6) 条件码

条件码 (cccc)	条件公式	说明
0 0 0 0	$OV = 1$	溢出
1 0 0 0	$OV = 0$	无溢出
0 0 0 1	$CY = 1$	进位 小(相比小)
1 0 0 1	$CY = 0$	无进位 不小(相比大于或等于)
0 0 1 0	$Z = 1$	0
1 0 1 0	$Z = 0$	非 0
0 0 1 1	$(CY \text{ or } Z) = 1$	不高(相比小于或等于 l)
1 0 1 1	$(CY \text{ or } Z) = 0$	高(相比大于)
0 1 0 0	$S = 1$	负
1 1 0 0	$S = 0$	正
0 1 0 1	—	恒为(无条件的)
1 1 0 1	$SAT = 1$	饱和
0 1 1 0	$(S \text{ xor } OV) = 1$	小于有符号数。
1 1 1 0	$(S \text{ xor } OV) = 0$	大于或等于有符号数。
0 1 1 1	$((S \text{ xor } OV) \text{ or } Z) = 1$	小于或等于有符号数。
1 1 1 1	$((S \text{ xor } OV) \text{ or } Z) = 0$	大于有符号数。

D.2 指令集 (按字母先后次序)

(1/6)

助记符	操作数	操作码	操作	执行时钟			标志				
				i	r	l	CY	OV	S	Z	SAT
ADD	reg1,reg2	rrrrr001110RRRRR	GR[reg2]←GR[reg2]+GR[reg1]	1	1	1	×	×	×	×	
	imm5,reg2	rrrrr010010iiii	GR[reg2]←GR[reg2]+sign-extend(imm5)	1	1	1	×	×	×	×	
ADDI	imm16,reg1,reg2	rrrrr110000RRRRR iiiiiiiiiiiiiiii	GR[reg2]←GR[reg1]+sign-extend(imm16)	1	1	1	×	×	×	×	
AND	reg1,reg2	rrrrr001010RRRRR	GR[reg2]←GR[reg2]AND GR[reg1]	1	1	1		0	×	×	
ANDI	imm16,reg1,reg2	rrrrr110110RRRRR iiiiiiiiiiiiiiii	GR[reg2]←GR[reg1]AND zero-extend(imm16)	1	1	1		0	×	×	
Bcond	disp9	dddd1011dddc 注 1	如果条件满足 则: PC←PC+sign-extend(disp9)	2	2	2					
			如果条件不满足	注 2	注 2	注 2					
BSH	reg2,reg3	rrrrr11111100000 www01101000010	GR[reg3]←GR[reg2] (23 : 16) GR[reg2] (31 : 24) GR[reg2] (7 : 0) GR[reg2] (15 : 8)	1	1	1	×	0	×	×	
BSW	reg2,reg3	rrrrr11111100000 www01101000000	GR[reg3]←GR[reg2] (7 : 0) GR[reg2] (15 : 8) GR [reg2] (23 : 16) GR[reg2] (31 : 24)	1	1	1	×	0	×	×	
CALLT	imm6	0000001000iiii	CTPC←PC+2(return PC) CTPSW←PSW adr←CTBP+zero-extend(imm6 logically shift left by 1) PC←CTBP+zero-extend(Load-memory(adr,Halfword))	4	4	4					
CLR1	bit#3,disp16[reg1]	10bbb111110RRRRR ddddddddd	adr←GR[reg1]+sign-extend(disp16) Z flag←Not(Load-memory-bit(adr,bit#3)) Store-memory-bit(adr,bit#3,0)	3	3	3				×	
	reg2,[reg1]	rrrrr111111RRRRR 0000000011100100	adr←GR[reg1] Z flag←Not(Load-memory-bit(adr,reg2)) Store-memory-bit(adr,reg2,0)	注 3	注 3	注 3				×	
CMOV	cccc,imm5,reg2,reg3	rrrrr111111iiii www011000cccc0	如果条件满足 则 GR[reg3]←sign-extended(imm5) 否则 GR[reg3]←GR[reg2]	1	1	1					
	cccc,reg1,reg2,reg3	rrrrr111111RRRRR www011001cccc0	如果条件满足 则 GR[reg3]←GR[reg1] 否则 GR[reg3]←GR[reg2]	1	1	1					
CMP	reg1,reg2	rrrrr001111RRRRR	result←GR[reg2]-GR[reg1]	1	1	1	×	×	×	×	
	imm5,reg2	rrrrr010011iiii	result←GR[reg2]-sign-extend(imm5)	1	1	1	×	×	×	×	
CTRET		000001111100000 0000000101000100	PC←CTPC PSW←CTPSW	3	3	3	R	R	R	R	R
DBRET		000001111100000 0000000101000110	PC←DBPC PSW←DBPSW	3	3	3	R	R	R	R	R

(2/6)

助记符	操作数	操作码	操作	执行时钟			标志				
				i	r	l	CY	OV	S	Z	SAT
DBTRAP		1111100001000000	DBPC←PC+2 (restored PC) DBPSW←PSW PSW.NP←1 PSW.EP←1 PSW.ID←1 PC←00000060H	3	3	3					
DI		0000011111100000 0000000101100000	PSW.ID←1	1	1	1					
DISPOSE	imm5,list12	0000011001iiiiL LLLLLLLLLLLL00000	sp←sp+zero-extend(imm5 logically shift left by 2) GR[reg in list12]←Load-memory(sp,Word) sp←sp+4 重复以上两个步骤直到表 12 中所有寄存器被载入。	n+1 注 4	n+1 注 4	n+1 注 4					
	imm5,list12,[reg1]	0000011001iiiiL LLLLLLLLLLLLRRRRR 注 5	sp←sp+zero-extend(imm5 logically shift left by 2) GR[reg in list12]←Load-memory(sp,Word) sp←sp+4 重复以上两个步骤直到表 12 中所有寄存器载入 PC←GR[reg1]	n+3 注 4	n+3 注 4	n+3 注 4					
DIV	reg1,reg2,reg3	rrrrr11111RRRRR wwwww01011000000	GR[reg2]←GR[reg2]÷GR[reg1] GR[reg3]←GR[reg2]%GR[reg1]	35	35	35		×	×	×	
DIVH	reg1,reg2	rrrrr000010RRRRR	GR[reg2]←GR[reg2]÷GR[reg1]注 6	35	35	35		×	×	×	
	reg1,reg2,reg3	rrrrr11111RRRRR wwwww01010000000	GR[reg2]←GR[reg2]÷GR[reg1]注 6 GR[reg3]←GR[reg2]%GR[reg1]	35	35	35		×	×	×	
DIVHU	reg1,reg2,reg3	rrrrr11111RRRRR wwwww01010000010	GR[reg2]←GR[reg2]÷GR[reg1]注 6 GR[reg3]←GR[reg2]%GR[reg1]	34	34	34		×	×	×	
DIVU	reg1,reg2,reg3	rrrrr11111RRRRR wwwww01011000010	GR[reg2]←GR[reg2]÷GR[reg1] GR[reg3]←GR[reg2]%GR[reg1]	34	34	34		×	×	×	
EI		1000011111100000 0000000101100000	PSW.ID←0	1	1	1					
HALT		0000011111100000 0000000100100000	停止	1	1	1					
HSW	reg2,reg3	rrrrr11111100000 wwwww01101000100	GR[reg3]←GR[reg2](15 : 0) GR[reg2] (31 : 16)	1	1	1	×	0	×	×	
JARL	disp22,reg2	rrrrr11110ddddd dddddddddddddd0 注 7	GR[reg2]←PC+4 PC←PC+sign-extend(disp22)	2	2	2					
JMP	[reg1]	00000000011RRRRR	PC←GR[reg1]	3	3	3					
JR	disp22	0000011110ddddd dddddddddddddd0 注 7	PC←PC+sign-extend(disp22)	2	2	2					
LD.B	disp16[reg1],reg2	rrrrr111000RRRRR ddddddddddddddd	adr←GR[reg1]+sign-extend(disp16) GR[reg2]←sign-extend(Load-memory(adrs,Byte))	1	1	注 11					
LD.BU	disp16[reg1],reg2	rrrrr11110bRRRRR dddddddddddddd1 注 8, 10	adr←GR[reg1]+sign-extend(disp16) GR[reg2]←zero-extend(Load-memory(adrs,Byte))	1	1	注 11					

(3/6)

助记符	操作数	操作码	操作	执行时钟			标志				
				i	r	l	CY	OV	S	Z	SAT
LD.H	disp16[reg1],reg2	rrrrr111001RRRRR dddddddddddddd0 注 8	adr←GR[reg1]+sign-extend(disp16) GR[reg2]←sign-extend(Load-memory(adrr,Halfword))	1	1	注 11					
LDSR	reg2,regID	rrrrr111111RRRRR 0000000000100000 注 12	SR[regID]←GR[reg2] 除非 regID = PSW	1	1	1					
			regID = PSW	1	1	1	×	×	×	×	×
LD.HU	disp16[reg1],reg2	rrrrr111111RRRRR dddddddddddddd1 注 8	adr←GR[reg1]+sign-extend(disp16) GR[reg2]←zero-extend(Load-memory(adrr,Halfword))	1	1	注 11					
LD.W	disp16[reg1],reg2	rrrrr111001RRRRR dddddddddddddd1 注 8	adr←GR[reg1]+sign-extend(disp16) GR[reg2]←Load-memory(adrr,Word)	1	1	注 11					
MOV	reg1,reg2	rrrrr000000RRRRR	GR[reg2]←GR[reg1]	1	1	1					
	imm5,reg2	rrrrr010000iiii	GR[reg2]←sign-extend(imm5)	1	1	1					
	imm32,reg1	00000110001RRRRR iiiiiiiiiiiiiiii iiiiiiiiiiiiiiii	GR[reg1]←imm32	2	2	2					
MOVEA	imm16,reg1,reg2	rrrrr110001RRRRR iiiiiiiiiiiiiiii	GR[reg2]←GR[reg1]+sign-extend(imm16)	1	1	1					
MOVHI	imm16,reg1,reg2	rrrrr110010RRRRR iiiiiiiiiiiiiiii	GR[reg2]←GR[reg1]+(imm16 0 ¹⁶)	1	1	1					
MUL	reg1,reg2,reg3	rrrrr111111RRRRR wwwww01000100000 注 14	GR[reg3] GR[reg2]←GR[reg2]xGR[reg1]	1	4	5					
	imm9,reg2,reg3	rrrrr111111iiii wwwww01001111100 注 13	GR[reg3] GR[reg2]←GR[reg2]xsign-extend(imm9)	1	4	5					
MULH	reg1,reg2	rrrrr000111RRRRR	GR[reg2]←GR[reg2] ⁶ xGR[reg1] ⁶	1	1	2					
	imm5,reg2	rrrrr010111iiii	GR[reg2]←GR[reg2] ⁶ xsign-extend(imm5)	1	1	2					
MULHI	imm16,reg1,reg2	rrrrr110111RRRRR iiiiiiiiiiiiiiii	GR[reg2]←GR[reg1] ⁶ ximm16	1	1	2					
MULU	reg1,reg2,reg3	rrrrr111111RRRRR wwwww01000100010 注 14	GR[reg3] GR[reg2]←GR[reg2]xGR[reg1]	1	4	5					
	imm9,reg2,reg3	rrrrr111111iiii wwwww0100111110 注 13	GR[reg3] GR[reg2]←GR[reg2]xzero-extend(imm9)	1	4	5					
NOP		0000000000000000	至少一个时钟周期内不作任何操作。	1	1	1					
NOT	reg1,reg2	rrrrr000001RRRRR	GR[reg2]←NOT(GR[reg1])	1	1	1		0	×	×	
NOT1	bit#3,disp16[reg1]	01bbb11110RRRRR dddddddddddddd	adr←GR[reg1]+sign-extend(disp16) Z flag←Not(Load-memory-bit(adrr,bit#3)) Store-memory-bit(adrr,bit#3,Z flag)	3 注 3	3 注 3	3 注 3				×	
	reg2,[reg1]	rrrrr111111RRRRR 0000000011100010	adr←GR[reg1] Z flag←Not(Load-memory-bit(adrr,reg2)) Store-memory-bit(adrr,reg2,Z flag)	3 注 3	3 注 3	3 注 3				×	

(4/6)

助记符	操作数	操作码	操作	执行时钟			标志				
				i	r	l	CY	OV	S	Z	SAT
OR	reg1,reg2	rrrrr001000RRRRR	GR[reg2]←GR[reg2]OR GR[reg1]	1	1	1		0	×	×	
ORI	imm16,reg1,reg2	rrrrr110100RRRRR iiiiiiiiiiiiiiii	GR[reg2]←GR[reg1]OR zero-extend(imm16)	1	1	1		0	×	×	
PREPARE	list12,imm5	0000011110iiiiL LLLLLLLLLLLL00001	Store-memory(sp−4,GR[reg in list12],Word) sp←sp−4 重复上述步骤直到列表 12 中的所有寄存器被保存 sp←sp-zero-extend(imm5)	n+1 注4	n+1 注4	n+1 注4					
	list12,imm5, sp/imm ^{⋆15}	0000011110iiiiL LLLLLLLLLLLLff011 imm16/imm32 注 16	Store-memory(sp−4,GR[reg in list12],Word) sp←sp+4 重复上述步骤直到列表 12 中的所有寄存器被保存 sp←sp-zero-extend (imm5) ep←sp/imm	n+2 注4 注17	n+2 注4 注17	n+2 注4 注17					
RETI		000001111100000 0000000101000000	如果 PSW.EP=1 那么 PC ←EIPC PSW ←EIPSW 否则, 如果 PSW.NP=1 则 PC ←FEPC PSW ←FEPSW 否则 PC ←EIPC PSW ←EIPSW	3	3	3	R	R	R	R	R
SAR	reg1,reg2	rrrrr11111RRRRR 0000000010100000	GR[reg2]←GR[reg2] 算术右移 GR[reg1]	1	1	1	×	0	×	×	
	imm5,reg2	rrrrr010101iiii	GR[reg2]←GR[reg2]算术右移 zero-extend (imm5)	1	1	1	×	0	×	×	
SASF	cccc,reg2	rrrrr111110cccc 0000001000000000	如果条件满足 则 GR[reg2]←(GR[reg2]逻辑左移 1) OR 00000001H 否则 GR[reg2]←(GR[reg2]逻辑左移 1) OR 00000000H	1	1	1					
SATADD	reg1,reg2	rrrrr000110RRRRR	GR[reg2]←saturated(GR[reg2]+GR[reg1])	1	1	1	×	×	×	×	×
	imm5,reg2	rrrrr010001iiii	GR[reg2]←saturated(GR[reg2]+sign-extend(imm5))	1	1	1	×	×	×	×	×
SATSUB	reg1,reg2	rrrrr000101RRRRR	GR[reg2]←saturated(GR[reg2]−GR[reg1])	1	1	1	×	×	×	×	×
SATSUBI	imm16,reg1,reg2	rrrrr110011RRRRR iiiiiiiiiiiiiiii	GR[reg2]←saturated(GR[reg1]−sign-extend(imm16))	1	1	1	×	×	×	×	×
SATSUBR	reg1,reg2	rrrrr000100RRRRR	GR[reg2]←saturated(GR[reg1]−GR[reg2])	1	1	1	×	×	×	×	×
SETF	cccc,reg2	rrrrr111110cccc 0000000000000000	如果条件满足 则 GR[reg2]←00000001H 否则 GR[reg2]←00000000H	1	1	1					

(5/6)

助记符	操作数	操作码	操作	执行时钟		标志				
				i	r	l	CY	OV	S	Z SAT
SET1	bit#3,disp16[reg1]	00bbb11110RRRRR dddddddddddddd	adr←GR[reg1]+sign-extend(disp16) Z flag←Not (Load-memory-bit(adr,bit#3)) Store-memory-bit(adr,bit#3,1)	3 注3	3 注3	3 注3				×
	reg2,[reg1]	rrrrr11111RRRRR 0000000011100000	adr←GR[reg1] Z flag←Not(Load-memory-bit(adr,reg2)) Store-memory-bit(adr,reg2,1)	3 注3	3 注3	3 注3				×
SHL	reg1,reg2	rrrrr11111RRRRR 0000000011100000	GR[reg2]←GR[reg2]逻辑左移 GR[reg1]	1	1	1	×	0	×	×
	imm5,reg2	rrrrr010110iiii	GR[reg2]←GR[reg2]逻辑左移 zero-extend(imm5)	1	1	1	×	0	×	×
SHR	reg1,reg2	rrrrr11111RRRRR 0000000010000000	GR[reg2]←GR[reg2]逻辑右移 GR[reg1]	1	1	1	×	0	×	×
	imm5,reg2	rrrrr010100iiii	GR[reg2]←GR[reg2]逻辑右移 zero-extend(imm5)	1	1	1	×	0	×	×
SLD.B	disp7[ep],reg2	rrrrr0110dddddd	adr←ep+zero-extend(disp7) GR[reg2]←sign-extend(Load-memory(adr,Byte))	1	1	注9				
SLD.BU	disp4[ep],reg2	rrrrr0000110dddd 注 18	adr←ep+zero-extend(disp4) GR[reg2]←zero-extend(Load-memory(adr,Byte))	1	1	注9				
SLD.H	disp8[ep],reg2	rrrrr1000dddddd 注 19	adr←ep+zero-extend(disp8) GR[reg2]←sign-extend(Load-memory(adr,Halfword))	1	1	注9				
SLD.HU	disp5[ep],reg2	rrrrr0000111dddd 注 18, 20	adr←ep+zero-extend(disp5) GR[reg2]←zero-extend(Load-memory(adr,Halfword))	1	1	注9				
SLD.W	disp8[ep],reg2	rrrrr1010dddddd0 注 21	adr←ep+zero-extend(disp8) GR[reg2]←Load-memory(adr,Word)	1	1	注9				
SST.B	reg2,disp7[ep]	rrrrr0111dddddd	adr←ep+zero-extend(disp7) Store-memory(adr,GR[reg2],Byte)	1	1	1				
SST.H	reg2,disp8[ep]	rrrrr1001dddddd 注 19	adr←ep+zero-extend(disp8) Store-memory(adr,GR[reg2],Halfword)	1	1	1				
SST.W	reg2,disp8[ep]	rrrrr1010dddddd1 注 21	adr←ep+zero-extend(disp8) Store-memory(adr,GR[reg2],Word)	1	1	1				
ST.B	reg2,disp16[reg1]	rrrrr111010RRRRR dddddddddddddd	adr←GR[reg1]+sign-extend(disp16) Store-memory(adr,GR[reg2],Byte)	1	1	1				
ST.H	reg2,disp16[reg1]	rrrrr111011RRRRR dddddddddddddd0 注 8	adr←GR[reg1]+sign-extend(disp16) Store-memory (adr,GR[reg2], Halfword)	1	1	1				
ST.W	reg2,disp16[reg1]	rrrrr111011RRRRR dddddddddddddd1 注 8	adr←GR[reg1]+sign-extend(disp16) Store-memory (adr,GR[reg2], Word)	1	1	1				
STSR	regID,reg2	rrrrr11111RRRRR 000000001000000	GR[reg2]←SR[regID]	1	1	1				

(6/6)

助记符	操作数	操作码	操作	执行时钟		标志				
				i	r	l	CY	OV	S	Z SAT
SUB	reg1,reg2	rrrrr001101RRRRR	GR[reg2]←GR[reg2]−GR[reg1]	1	1	1	x	x	x	x
SUBR	reg1,reg2	rrrrr001100RRRRR	GR[reg2]←GR[reg1]−GR[reg2]	1	1	1	x	x	x	x
SWITCH	reg1	0000000010RRRRR	adr←(PC+2) + (GR [reg1] logically shift left by 1) PC←(PC+2) + (sign-extend (Load-memory (adr,Halfword)) 逻辑左移 1	5	5	5				
SXB	reg1	00000000101RRRRR	GR[reg1]←sign-extend (GR[reg1] (7 : 0))	1	1	1				
SXH	reg1	00000000111RRRRR	GR[reg1]←sign-extend (GR[reg1] (15 : 0))	1	1	1				
TRAP	vector	000001111111iiii 0000000100000000	EIPC ←PC+4 (恢复 PC) EIPSW ←PSW ECR.EICC ←中断代码 PSW.EP ←1 PSW.ID ←1 PC ←00000040H (当矢量为 00H 到 0FH) 00000050H (当矢量为 10H 到 1FH)	3	3	3				
TST	reg1,reg2	rrrrr001011RRRRR	result←GR[reg2] AND GR[reg1]	1	1	1		0	x	x
TST1	bit#3,disp16[reg1]	11bbb111110RRRRR ddddddddddddddd	adr←GR[reg1]+sign-extend(disp16) Z flag←Not (Load-memory-bit (adr,bit#3))	3 注3	3 注3	3 注3				x
	reg2, [reg1]	rrrrr111111RRRRR 0000000011100110	adr←GR[reg1] Z flag←Not (Load-memory-bit (adr,reg2))	3 注3	3 注3	3 注3				x
XOR	reg1,reg2	rrrrr001001RRRRR	GR[reg2]←GR[reg2] XOR GR[reg1]	1	1	1		0	x	x
XORI	imm16,reg1,reg2	rrrrr110101RRRRR iiiiiiiiiiiiiiii	GR[reg2]←GR[reg1] XOR zero-extend (imm16)	1	1	1		0	x	x
ZXB	reg1	00000000100RRRRR	GR[reg1]←zero-extend (GR[reg1] (7 : 0))	1	1	1				
ZXH	reg1	00000000110RRRRR	GR[reg1]←zero-extend (GR[reg1] (15 : 0))	1	1	1				

- 注
1. dddddddd: disp9 的高 8 位。
 2. 如果之前紧邻一条指令改写 PSW 的内容, 则为 3。
 3. 如果没有等待状态 (3 + 读取访问等待状态数)。
 4. n 为列表 12 中加载寄存器的总数。(根据等待状态数目, n 为列表 12 中加载寄存器的总数。如果 n = 0, 和 n = 1 时的操作相同)
 5. RRRRR: 除了 00000 之外。
 6. 只有低半字有效。
 7. dddddddddddddddddddd: disp22 的高 21 位。
 8. dddddddddddddddd: disp16 的高 15 位。
 9. 依照等待状态的数目(如果没有等待状态, 则为 1)。
 10. b: disp16 的位 0。
 11. 依照等待状态的数目(如果没有等待状态, 则为 2)。

- 注 12. 本指令中，为了助记符描述方便，使源寄存器为 reg2，而操作码位置上使用 reg1 寄存器。所以，助记符描述中寄存器的含义和其它指令有所不同。
- rrrrr = 指定 reg1D
RRRRR = 指定 reg2
13. iiii: imm9 的低 5 位。
IIII: imm9 的高 4 位。
14. 不要将为通用寄存器 reg1 和 reg3 指定同一寄存器。
15. sp/imm: 由次操作码的位 19 和位 20 指定。
16. ff = 00: 将 sp 载入 ep。
01: 将符号扩展的 16 位立即数(位 47 至 32)载入 ep。
10: 将经过 16 位逻辑左移的 16 位立即数 (位 47 至 32) 载入 ep。
11: 将 32 位立即数 (位 63 至 32) 载入 ep。
17. 如果 imm = imm32, n + 3 时钟。
18. rrrrr: 非 00000。
19. ddddddd: disp8 的高 8 位。
20. dddd: disp5 的高 4 位。
21. ddddddd: disp8 的高 6 位。
22. V850ES/SJ2-H 的时钟+1。

E.1 本辑主要修订之处

页码	描述说明
p. 75	修改 3.4.4 (2) (d) 仅数据 RAM (8 KB)
p.75	添加 3.4.4 (2) (d) 仅数据 RAM (8 KB) 中的注意事项
p.75	修改图 3-10. 仅数据 RAM (8 KB)
p.76	修改 3.4.4 (4) 外部存储区域
pp.80, 88 至 91, 93	修改 3.4.6 外围 I/O 寄存器
p.263	修改图 7-23. (d) TAAAn I/O 控制寄存器 2 (TAAAnIOC2)
p.275	修改图 7-27. (d) TAAAn I/O 控制寄存器 2 (TAAAnIOC2)
p.325	修改 7.8 级联连接
p.640	修改表 12-1 实时计数器的配置
p.641	修改图 12-1. 实时计数器框图
p.656	添加 12.3 (17) 预分频器模式寄存器 0 (PRSM0)
p.657	添加 12.3 (18) 预分频比较寄存器 0 (PRSCM0)
p.775	添加 18.4 (2) CSIFn 控制寄存器 1 (CFnCTL1) 中的注 2
p.871	修改表 19-4. 扩展码位定义
p.889 至 891	修改图 19-23. 主机向从机传输举例 (当主机设备和从机设备都选择 9 时钟等待时)
p.892 至 894	修改图 19-24. 从机向主机传输举例 (当主机选择 8 时钟等待而从机选择 9 时钟等待时)
p.1039	修改图 21-1. USB 功能控制器框图
p.1235	添加 22.3 (7) 外部 DMA 请求使能寄存器 (EXDRQEN)
p.1319	修改表 25-10. 副时钟工作模式下的工作状态
p.1336	修改图 28-1. 低电压检测器框图
p.1405, 1406	修改 33.5.1 I/O 电平
p.1411,1412	修改 33.7.2 (1) (a) 读/写周期 (CLKOUT 异步)
p.1414	修改 33.7.2 (1) (b) 读/写周期 (CLKOUT 同步): 在复用总线模式/独立总线模式
p.1416	修改 33.7.2 (2) (a) CLKOUT 异步
p.1417	33.7.2 (2) (b) CLKOUT 同步
p.1422	修改 33.7.2 (6) (a) 主机模式
p.1423	修改 33.7.2 (6) (b) 从机模式
p.1428	修改 33.8 (10) A/D 转换器
p.1470	添加附录 E 修订历史

详细信息请联系：

中国区

MCU 技术支持热线：

电话：+86-400-700-0606 (普通话)

服务时间：9:00-12:00, 13:00-17:00 (不含法定节假日)

网址：

<http://www.cn.necel.com/> (中文)

<http://www.necel.com/> (英文)

[北京]

日电电子（中国）有限公司

中国北京市海淀区知春路 27 号

量子芯座 7, 8, 9, 15 层

电话：(+86) 10-8235-1155

传真：(+86) 10-8235-7679

[深圳]

日电电子（中国）有限公司深圳分公司

深圳市福田区益田路卓越时代广场大厦 39 楼

3901, 3902, 3909 室

电话：(+86) 755-8282-9800

传真：(+86) 755-8282-9899

[上海]

日电电子（中国）有限公司上海分公司

中国上海市浦东新区银城中路 200 号

中银大厦 2409-2412 和 2509-2510 室

电话：(+86) 21-5888-5400

传真：(+86) 21-5888-5230

[香港]

香港日电电子有限公司

香港九龙旺角太子道西 193 号新世纪广场

第 2 座 16 楼 1601-1613 室

电话：(+852) 2886-9318

传真：(+852) 2886-9022

2886-9044

上海恩益禧电子国际贸易有限公司

中国上海市浦东新区银城中路 200 号

中银大厦 2511-2512 室

电话：(+86) 21-5888-5400

传真：(+86) 21-5888-5230

[成都]

日电电子（中国）有限公司成都分公司

成都市二环路南三段 15 号天华大厦 7 楼 703 室

电话：(+86)28-8512-5224

传真：(+86)28-8512-5334

[长春]

日电电子（中国）有限公司长春分公司

吉林省长春市朝阳区

西安大路 727 号中银大厦 A 座 1609 室

电话：(+86)431-8859-7533 / 8859-8533

传真：(+86)431-8680-2944

[大连]

日电电子（中国）有限公司长春分公司

大连市中山路 88 号天安国际大厦 2701 室

电话：(+86)411-8230-8815 / 8230-8825

传真：(+86)411-8230-8835