

RX210 群

用户手册 硬件篇

瑞萨 32 位单片机
RX 族 / RX200 系列

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

Notice

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

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

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

产品使用时的注意事项

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

1. 未使用的引脚的处理

【注意】将未使用的引脚按照正文的“未使用引脚的处理”进行处理。

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

2. 通电时的处理

【注意】通电时产品处于不定状态。

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

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

3. 禁止存取保留地址（保留区）

【注意】禁止存取保留地址（保留区）

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

4. 关于时钟

【注意】复位时，请在时钟稳定后解除复位。

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

5. 关于产品间的差异

【注意】在变更不同型号的产品时，请对每一个产品型号进行系统评价测试。

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

本手册的使用方法

1 目的和对象

本手册是一本帮助用户理解本单片机的硬件功能和电特性的手册。它以使用本手册来设计应用系统的用户为对象。在使用本手册时，需要具备电路、逻辑电路以及单片机的基础知识。

本手册由产品概要、CPU、系统控制功能、外围功能、电特性、使用时的注意事项几大部分组成。

必须在充分确认注意事项后使用本单片机。注意事项记录在各章的正文中、各章的最后和注意事项章节中。

修订记录归纳了对旧版本记载内容的更正或追加的主要位置。并不是修订内容的全部记载。详情请确认本手册的正文。

RX210 群准备了以下的文献。请使用最新的文献。最新版本刊登在瑞萨电子的主页上。

文献的种类	记载内容	资料名	资料号
数据表	硬件的概要和电特性	RX210 Group Datasheet	R01DS0041EJ
用户手册 硬件篇	硬件规格（引脚排列、存储器映像、外围功能规格、电特性、时序）和运行的说明	RX210 群 用户手册 硬件篇	本硬件手册
用户手册 软件篇	CPU 指令集的说明	RX 族 用户手册 软件篇	R01US0028CJ
应用手册	应用例子参考程序等	—	—
RENESAS TECHNICAL UPDATE	产品规格、文献等的快速公告	—	—

2 寄存器的表示

在各章的“寄存器说明”中，有位的排列图和功能表，说明位的设定内容。以下说明所使用的符号和术语。

X.X.X 寄存器

地址 xxxx xxxxxh

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—

复位后的值 x 0 0 0 0 0 0 0

位	符号	位名	功能	R/W
b00 位 (注2)	0: (注3) 1: 不能设定。	R/W (注1)
b3-b1	—	保留位	读写值都为“0”。	R/W
b44 位	0: 1:	R
b6-b5[1:0] 位	0 0: 0 1: (注3) 上述以外: 不能设定。	R(W) (注1)
b7	—	保留位	读取值为不定值, 写无效。	R

- 注1. R/W : 可读写。
 R(W) : 可读写, 但是有写数据的限制。
 限制内容请参照各寄存器的说明和注释。
 R : 可写, 写数据无效。
- 注2. 保留位。只能写指定值, 如果写的值不是指定的值, 就不保证运行。
- 注3. 不能设定。如果设定, 就不保证运行。

3 省略语及简称的说明

省略 / 简称	全称	备注
ACIA	Asynchronous Communication Interface Adapter	异步通信接口适配器
bps	bits per second	位 / 秒; 每秒传送位数
CRC	Cyclic Redundancy Check	循环冗余校验
DMA	Direct Memory Access	直接存储器存取
DMAC	Direct Memory Access Controller	DMA 控制器
GSM	Global System for Mobile Communications	全球数字移动电话系统
Hi-Z	High Impedance	高阻抗
IEBus	Inter Equipment bus	—
I/O	Input/Output	输入 / 输出
IrDA	Infrared Data Association	红外线数据协会
LSB	Least Significant Bit	最低有效位
MSB	Most Significant Bit	最高有效位
NC	Non-Connection	未连接引脚
PLL	Phase Locked Loop	锁相环路
PWM	Pulse Width Modulation	脉宽调制
SFR	Special Function Registers	特殊功能寄存器
SIM	Subscriber Identity Module	ISO-7816 规定的 IC 卡
UART	Universal Asynchronous Receiver/Transmitter	通用异步收发器
VCO	Voltage Controlled Oscillator	电压控制振荡器

所有商标及注册商标分别归属于其所有者。
本用户手册仅为参考译文，对应的日文版和英文版具有正式效力。

目 录

特点	1
1. 概要	2
1.1 规格概要	2
1.2 产品一览表	7
1.3 框图	9
1.4 引脚功能	10
1.5 引脚排列图	14
2. CPU	27
2.1 特点	27
2.2 CPU 寄存器组	28
2.2.1 通用寄存器 (R0 ~ R15)	29
2.2.2 控制寄存器	29
2.2.3 DSP 功能指令的相关寄存器	32
2.3 处理器模式	33
2.3.1 管理模式	33
2.3.2 用户模式	33
2.3.3 特权指令	33
2.3.4 处理器模式之间的转移	33
2.4 数据类型	34
2.4.1 整数	34
2.4.2 位	34
2.4.3 字符串	35
2.5 字节序	36
2.5.1 字节序的设定	36
2.5.2 I/O 寄存器的存取	39
2.5.3 存取 I/O 寄存器时的注意事项	39
2.5.4 数据排列	40
2.5.5 分配指令码的注意事项	40
2.6 向量表	41
2.6.1 固定向量表	41
2.6.2 可向量表	42
2.7 指令操作	42
2.7.1 RMPA 指令和字符串操作指令的数据预取	42
2.8 流水线	43
2.8.1 概要	43
2.8.2 指令和流水线处理	44
2.8.3 指令处理时间的计算方法	50
2.8.4 中断响应周期数	51
3. 运行模式	52
3.1 运行模式的种类和选择	52
3.2 寄存器说明	53
3.2.1 模式监视寄存器 (MDMONR)	53
3.2.2 模式状态寄存器 (MDSR)	53
3.2.3 系统控制寄存器 0 (SYSCR0)	54
3.2.4 系统控制寄存器 1 (SYSCR1)	55
3.3 运行模式的说明	56
3.3.1 单芯片模式	56

3.3.2	内部 ROM 有效扩展模式	56
3.3.3	内部 ROM 无效扩展模式	56
3.3.4	引导模式	56
3.3.5	用户引导模式	56
3.4	运行模式的转移	57
3.4.1	通过模式引脚进行运行模式的转移	57
3.4.2	通过寄存器进行运行模式的转移	58
4.	地址空间	59
4.1	地址空间	59
4.2	外部地址空间	60
5.	I/O 寄存器	61
5.1	I/O 寄存器地址一览表（地址顺序）	63
6.	复位	85
6.1	概要	85
6.2	寄存器说明	87
6.2.1	复位状态寄存器 0（RSTSR0）	87
6.2.2	复位状态寄存器 1（RSTSR1）	89
6.2.3	复位状态寄存器 2（RSTSR2）	90
6.2.4	软件复位寄存器（SWRR）	91
6.3	运行说明	92
6.3.1	引脚复位	92
6.3.2	上电复位、电压监视 0 复位	92
6.3.3	电压监视 1 复位、电压监视 2 复位	93
6.3.4	深度软件待机复位	94
6.3.5	独立看门狗定时器复位	95
6.3.6	看门狗定时器复位	95
6.3.7	软件复位	95
6.3.8	冷启动 / 热启动判断功能	95
6.3.9	复位发生源的判断	96
7.	选项设定存储器	97
7.1	概要	97
7.2	寄存器说明	98
7.2.1	选项功能选择寄存器 0（OFS0）	98
7.2.2	选项功能选择寄存器 1（OFS1）	102
7.2.3	字节序选择寄存器 B（MDEB）、字节序选择寄存器 S（MDES）	103
7.3	UB 码	104
7.3.1	UB 码 A	104
7.3.2	UB 码 B	104
7.4	使用时的注意事项	104
7.4.1	选项设定存储器的设定例子	104
8.	电压检测电路（LVDAa）	105
8.1	概要	105
8.2	寄存器说明	108
8.2.1	电压监视 1 电路 / 比较器 A1 的控制寄存器 1（LVD1CR1）	108
8.2.2	电压监视 1 电路 / 比较器 A1 的状态寄存器（LVD1SR）	108
8.2.3	电压监视 2 电路 / 比较器 A2 的控制寄存器 1（LVD2CR1）	109
8.2.4	电压监视 2 电路 / 比较器 A2 的状态寄存器（LVD2SR）	109

8.2.5	电压监视电路 / 比较器 A 的控制寄存器 (LVCMPCR)	110
8.2.6	电压检测电平选择寄存器 (LVDLVLR)	111
8.2.7	电压监视 1 电路 / 比较器 A1 的控制寄存器 0 (LVD1CR0)	112
8.2.8	电压监视 2 电路 / 比较器 A2 的控制寄存器 0 (LVD2CR0)	114
8.3	VCC 输入电压的监视	116
8.3.1	Vdet0 的监视	116
8.3.2	Vdet1 的监视	116
8.3.3	Vdet2 的监视	116
8.4	电压监视 0 复位	117
8.5	电压监视 1 中断和电压监视 1 复位	118
8.6	电压监视 2 中断和电压监视 2 复位	121
8.7	事件链接输出功能	124
8.7.1	中断处理和事件链接的关系	124
9.	时钟发生电路	125
9.1	概要	125
9.2	寄存器说明	127
9.2.1	系统时钟控制寄存器 (SCKCR)	127
9.2.2	系统时钟控制寄存器 3 (SCKCR3)	129
9.2.3	PLL 控制寄存器 (PLLCR)	130
9.2.4	PLL 控制寄存器 2 (PLLCR2)	131
9.2.5	外部总线时钟控制寄存器 (BCKCR)	132
9.2.6	主时钟振荡器的控制寄存器 (MOSCCR)	133
9.2.7	副时钟振荡器的控制寄存器 (SOSCCR)	134
9.2.8	低速内部振荡器的控制寄存器 (LOCOCR)	135
9.2.9	IWDT 专用内部振荡器的控制寄存器 (ILOCOCR)	136
9.2.10	高速内部振荡器的控制寄存器 (HOCOCR)	137
9.2.11	高速内部振荡器的控制寄存器 2 (HOCOCR2)	138
9.2.12	振荡停止检测控制寄存器 (OSTDCR)	139
9.2.13	振荡停止检测状态寄存器 (OSTDSR)	140
9.2.14	主时钟振荡器的强制振荡控制寄存器 (MOFCR)	141
9.2.15	高速内部振荡器的电源控制寄存器 (HOCOPCR)	142
9.3	主时钟振荡器	143
9.3.1	连接谐振器的方法	143
9.3.2	输入外部时钟的方法	144
9.3.3	有关输入外部时钟的注意事项	144
9.4	副时钟振荡器	145
9.4.1	连接 32.768kHz 晶体谐振器的方法	145
9.4.2	不使用副时钟情况下的引脚处理	145
9.5	振荡停止检测功能	146
9.5.1	振荡停止检测和检测后的运行	146
9.5.2	振荡停止检测中断	147
9.6	PLL 电路	147
9.7	内部时钟	148
9.7.1	系统时钟	148
9.7.2	外围模块时钟	148
9.7.3	FlashIF 时钟	148
9.7.4	外部总线时钟	148
9.7.5	CAC 时钟	149
9.7.6	RTC 专用时钟	149
9.7.7	IWDT 专用低速时钟	149

9.8	使用时的注意事项	150
9.8.1	有关时钟发生电路的注意事项	150
9.8.2	有关谐振器的注意事项	150
9.8.3	设计电路板时的注意事项	151
9.8.4	有关谐振器连接引脚的注意事项	151
9.8.5	有关副时钟的注意事项	151
10.	时钟频率精度测量电路 (CAC)	152
10.1	概要	152
10.2	寄存器说明	153
10.2.1	CAC 控制寄存器 0 (CACR0)	153
10.2.2	CAC 控制寄存器 1 (CACR1)	154
10.2.3	CAC 控制寄存器 2 (CACR2)	155
10.2.4	CAC 中断控制寄存器 (CAICR)	156
10.2.5	CAC 状态寄存器 (CASTR)	157
10.2.6	CAC 上限值设定寄存器 (CAULVR)	158
10.2.7	CAC 下限值设定寄存器 (CALLVR)	158
10.2.8	CAC 计数器的缓冲寄存器 (CACNTBR)	158
10.3	运行说明	159
10.3.1	以 CACREF 引脚输入为基准测量时钟频率	159
10.3.2	以其他时钟源为基准测量时钟频率	160
10.3.3	CACREF 引脚的数字滤波器功能	161
10.4	中断请求	161
10.5	使用时的注意事项	161
10.5.1	模块停止功能的设定	161
11.	低功耗功能	162
11.1	概要	162
11.2	寄存器说明	165
11.2.1	待机控制寄存器 (SBYCR)	165
11.2.2	模块停止控制寄存器 A (MSTPCRA)	166
11.2.3	模块停止控制寄存器 B (MSTPCRB)	168
11.2.4	模块停止控制寄存器 C (MSTPCRC)	170
11.2.5	运行功率控制寄存器 (OPCCR)	171
11.2.6	睡眠模式返回时钟源转换寄存器 (RSTCKCR)	176
11.2.7	主时钟振荡器的等待控制寄存器 (MOSCWTCR)	177
11.2.8	副时钟振荡器的等待控制寄存器 (SOSCWTCR)	178
11.2.9	PLL 等待控制寄存器 (PLLWTCR)	179
11.2.10	HOCO 等待控制寄存器 2 (HOCOWTCR2)	180
11.2.11	深度待机控制寄存器 (DPSBYCR)	181
11.2.12	深度待机中断允许寄存器 0 (DPSIER0)	182
11.2.13	深度待机中断允许寄存器 2 (DPSIER2)	183
11.2.14	深度待机中断标志寄存器 0 (DPSIFR0)	184
11.2.15	深度待机中断标志寄存器 2 (DPSIFR2)	185
11.2.16	深度待机中断边沿寄存器 0 (DPSIEGR0)	187
11.2.17	深度待机中断边沿寄存器 2 (DPSIEGR2)	188
11.2.18	闪存 HOCO 软件待机控制寄存器 (FHSSBYCR)	189
11.2.19	深度待机备用寄存器 (DPSBKRY) (y=0 ~ 31)	189
11.3	通过转换时钟来降低功耗	190
11.4	模块停止功能	190
11.5	低运行功率功能	190
11.5.1	运行功率控制模式的设定方法	190

11.6	低功耗状态	191
11.6.1	睡眠模式	191
11.6.2	全模块时钟停止模式	192
11.6.3	软件待机模式	193
11.6.4	深度软件待机模式	195
11.7	使用时的注意事项	199
11.7.1	I/O 端口状态	199
11.7.2	DMAC 和 DTC 的模块停止	199
11.7.3	内部外围模块的中断	199
11.7.4	MSTPCRA、MSTPCRB 和 MSTPCRC 寄存器的写操作	199
11.7.5	通过 DIRQnE 位 (n=0 ~ 7) 控制输入缓冲器	199
11.7.6	WAIT 指令的执行时序	199
11.7.7	睡眠模式中通过 DMAC 和 DTC 进行的寄存器改写	199
11.7.8	全模块时钟停止模式的解除	199
11.7.9	将副时钟用作系统时钟的时钟源时的注意事项	199
12.	寄存器写保护功能	200
12.1	寄存器说明	201
12.1.1	保护寄存器 (PRCR)	201
13.	异常处理	202
13.1	异常事件	202
13.1.1	未定义指令异常	202
13.1.2	特权指令异常	202
13.1.3	复位	202
13.1.4	非屏蔽中断	202
13.1.5	中断	202
13.1.6	无条件陷阱	202
13.2	异常处理步骤	203
13.3	异常事件的接受	204
13.3.1	接受时序和被保存的 PC 值	204
13.3.2	向量和 PC、PSW 的保存场所	204
13.4	接受异常 / 从异常返回时的硬件处理	205
13.5	硬件预处理	206
13.5.1	未定义指令异常	206
13.5.2	特权指令异常	206
13.5.3	复位	206
13.5.4	非屏蔽中断	206
13.5.5	中断	206
13.5.6	无条件陷阱	207
13.6	从异常处理程序的返回	207
13.7	异常事件的优先级	207
14.	中断控制器 (ICUb)	208
14.1	概要	208
14.2	寄存器说明	210
14.2.1	中断请求寄存器 n (IRn) (n= 中断向量号)	210
14.2.2	中断请求允许寄存器 m (IERm) (m=02h ~ 1Fh)	211
14.2.3	中断源优先级寄存器 n (IPRn) (n=000 ~ 249)	212
14.2.4	高速中断设定寄存器 (FIR)	213
14.2.5	软件中断启动寄存器 (SWINTR)	214
14.2.6	DTC 启动允许寄存器 n (DTCERn) (n= 中断向量号)	214

14.2.7	DMAC 启动请求选择寄存器 m (DMRSRm) (m=DMAC 通道号)	215
14.2.8	IRQ 控制寄存器 i (IRQCRi) (i=0 ~ 7)	215
14.2.9	IRQ 引脚数字滤波器的允许寄存器 0 (IRQFLTE0)	216
14.2.10	IRQ 引脚数字滤波器的设定寄存器 0 (IRQFLTC0)	217
14.2.11	非屏蔽中断状态寄存器 (NMISR)	218
14.2.12	非屏蔽中断允许寄存器 (NMIER)	220
14.2.13	非屏蔽中断状态清除寄存器 (NMICLR)	222
14.2.14	NMI 引脚中断控制寄存器 (NMICR)	223
14.2.15	NMI 引脚数字滤波器的允许寄存器 (NMIFLTE)	223
14.2.16	NMI 引脚数字滤波器的设定寄存器 (NMIFLTC)	224
14.3	向量表	225
14.3.1	中断向量表	225
14.3.2	高速中断的向量表地址	231
14.3.3	非屏蔽中断的向量表地址	231
14.4	中断的运行说明	232
14.4.1	中断检测	232
14.4.2	中断请求的允许 / 禁止	234
14.4.3	中断请求目标的选择	235
14.4.4	优先级的判断	237
14.4.5	高速中断	237
14.4.6	数字滤波器	237
14.4.7	外部引脚中断	238
14.5	非屏蔽中断的运行说明	238
14.6	从低功耗状态的返回	239
14.6.1	从睡眠模式的返回	239
14.6.2	从全模块时钟停止模式的返回	239
14.6.3	从软件待机模式的返回	239
14.7	使用时的注意事项	239
14.7.1	使用非屏蔽中断时的 WAIT 指令的注意事项	239
15.	总线	240
15.1	概要	240
15.2	总线说明	242
15.2.1	CPU 总线	242
15.2.2	存储器总线	242
15.2.3	内部主总线	243
15.2.4	内部外围总线	243
15.2.5	外部总线	245
15.2.6	并行运行	247
15.2.7	总线的设定	247
15.2.8	限制事项	247
15.3	寄存器说明	248
15.3.1	CSn 控制寄存器 (CSnCR) (n=0 ~ 3)	248
15.3.2	CSn 恢复周期设定寄存器 (CSnREC) (n=0 ~ 3)	250
15.3.3	CS 恢复周期插入允许寄存器 (CSRECEN)	252
15.3.4	CSn 模式寄存器 (CSnMOD) (n=0 ~ 3)	255
15.3.5	CSn 等待控制寄存器 1 (CSnWCR1) (n=0 ~ 3)	257
15.3.6	CSn 等待控制寄存器 2 (CSnWCR2) (n=0 ~ 3)	261
15.3.7	总线错误状态清除寄存器 (BERCLR)	265
15.3.8	总线错误监视允许寄存器 (BEREN)	265
15.3.9	总线错误状态寄存器 1 (BERSR1)	266
15.3.10	总线错误状态寄存器 2 (BERSR2)	266

15.3.11	总线优先级控制寄存器 (BUSPRI)	267
15.4	字节序和数据调整	269
15.4.1	CS 区域的数据调整控制	269
15.5	CS 区域控制器的运行说明	273
15.5.1	分离总线	273
15.5.2	地址 / 数据多路复用总线	282
15.5.3	外部等待功能	285
15.5.4	恢复周期的插入	287
15.5.5	不存取时的状态	290
15.5.6	写缓冲器功能	290
15.6	限制事项	291
15.6.1	使用分离总线接口时的限制事项	291
15.6.2	使用地址 / 数据多路复用总线时的限制事项	291
15.6.3	兼用 A0 引脚和 BC0# 引脚的产品时的限制事项	292
15.6.4	通过 BCLK 引脚输出选择位设定 BCLK 的 2 分频时的限制事项	292
15.6.5	禁止存取跨地址空间的各区域	292
15.6.6	有关 RMPA 指令和字符串操作指令的限制事项	292
15.6.7	有关指令代码的限制事项	292
15.7	总线错误监视部	293
15.7.1	总线错误的种类	293
15.7.2	发生总线错误时的运行	293
15.7.3	总线错误的发生条件	294
16.	DMA 控制器 (DMACA)	295
16.1	概要	295
16.2	寄存器说明	297
16.2.1	DMA 传送源地址寄存器 (DMSAR)	297
16.2.2	DMA 传送目标地址寄存器 (DMDAR)	297
16.2.3	DMA 传送计数寄存器 (DMCRA)	298
16.2.4	DMA 块传送计数寄存器 (DMCRB)	299
16.2.5	DMA 传送模式寄存器 (DMTMD)	300
16.2.6	DMA 中断设定寄存器 (DMINT)	301
16.2.7	DMA 地址模式寄存器 (DMAMD)	302
16.2.8	DMA 偏移寄存器 (DMOFR)	305
16.2.9	DMA 传送允许寄存器 (DMCNT)	305
16.2.10	DMA 软件启动寄存器 (DMREQ)	306
16.2.11	DMA 状态寄存器 (DMSTS)	307
16.2.12	DMA 启动源标志控制寄存器 (DMCSL)	308
16.2.13	DMA 模块启动寄存器 (DMAST)	309
16.3	运行说明	310
16.3.1	传送模式	310
16.3.2	扩展重复区域功能	313
16.3.3	带偏移的地址更新功能	314
16.3.4	启动源	318
16.3.5	运行时序	318
16.3.6	DMAC 的执行周期	319
16.3.7	DMAC 的启动	320
16.3.8	DMA 传送的开始	321
16.3.9	DMA 传送过程中的寄存器	321
16.3.10	通道的优先级	322
16.4	DMA 传送的结束	323
16.4.1	通过设定的总数据传送结束传送	323

16.4.2	通过重复大小结束中断结束传送	323
16.4.3	通过扩展重复区域上溢中断结束传送	323
16.5	中断	324
16.6	事件链接功能	325
16.7	低功耗功能	326
16.8	使用时的注意事项	327
16.8.1	使用外部设备的情况	327
16.8.2	外围模块的 DMA 传送的情况	327
16.8.3	有关 DMA 运行过程中的寄存器存取	327
16.8.4	有关保留区的 DMA 传送	327
16.8.5	因设定 DMA 启动源标志控制寄存器 (DMCSL) 而产生的各传送结束中断请求	327
16.8.6	中断控制器的 DMAC 启动请求寄存器 (ICU.DMRSRm) 的设定	327
16.8.7	保留或者重新开始 DMA 启动的方法	327
17.	数据传送控制器 (DTCa)	328
17.1	概要	328
17.2	寄存器说明	330
17.2.1	DTC 模式寄存器 A (MRA)	330
17.2.2	DTC 模式寄存器 B (MRB)	331
17.2.3	DTC 传送源寄存器 (SAR)	332
17.2.4	DTC 传送目标寄存器 (DAR)	332
17.2.5	DTC 传送计数寄存器 A (CRA)	333
17.2.6	DTC 传送计数寄存器 B (CRB)	334
17.2.7	DTC 控制寄存器 (DTCCR)	334
17.2.8	DTC 向量基址寄存器 (DTCVBR)	335
17.2.9	DTC 地址模式寄存器 (DTCADM0D)	335
17.2.10	DTC 模块启动寄存器 (DTCST)	336
17.2.11	DTC 状态寄存器 (DTCSTS)	337
17.3	启动源	338
17.3.1	传送信息的分配和 DTC 向量表	338
17.4	运行说明	340
17.4.1	传送信息的跳读功能	342
17.4.2	传送信息的回写省略功能	343
17.4.3	正常传送模式	344
17.4.4	重复传送模式	345
17.4.5	块传送模式	346
17.4.6	链传送	347
17.4.7	运行时序	348
17.4.8	DTC 的执行周期	351
17.4.9	DTC 的总线权释放时序	351
17.5	DTC 的设定步骤	352
17.6	DTC 的使用例子	353
17.6.1	正常传送	353
17.6.2	计数器为“0”时的链传送	354
17.7	中断源	355
17.8	事件链接功能	355
17.9	低功耗功能	355
17.10	使用时的注意事项	356
17.10.1	传送信息起始地址 / 传送源地址 / 传送目标地址	356
17.10.2	传送信息的分配	356
17.10.3	中断控制器的 DTC 启动允许寄存器 (ICU.DTCERn) 的设定	356

18. 事件链接控制器 (ELC)	357
18.1 概要	357
18.2 寄存器说明	358
18.2.1 事件链接控制寄存器 (ELCR)	358
18.2.2 事件链接设定寄存器 n (ELSRn) (n=1 ~ 4、7、10、12、15、16、18 ~ 29)	359
18.2.3 事件链接选项设定寄存器 A (ELOPA)	362
18.2.4 事件链接选项设定寄存器 B (ELOPB)	363
18.2.5 事件链接选项设定寄存器 C (ELOPC)	363
18.2.6 事件链接选项设定寄存器 D (ELOPD)	364
18.2.7 端口组指定寄存器 n (PGRn) (n=1、2)	364
18.2.8 端口组控制寄存器 n (PGCn) (n=1、2)	365
18.2.9 端口缓冲寄存器 n (PDBFn) (n=1、2)	366
18.2.10 事件链接端口指定寄存器 n (PELn) (n=0 ~ 3)	367
18.2.11 事件链接的软件事件发生寄存器 (ELSEGR)	368
18.3 运行说明	369
18.3.1 中断处理和事件链接的关系	369
18.3.2 事件链接	370
18.3.3 输入定时器类外围功能的事件时的运行	371
18.3.4 输入 A/D 转换器和 D/A 转换器的事件时的运行	371
18.3.5 端口的事件输入运行和事件发生运行	371
18.3.6 事件链接的运行设定步骤	375
18.4 使用时的注意事项	375
18.4.1 有关 ELSR18 寄存器和 ELSR19 寄存器的设定	375
18.4.2 有关输出端口组的位循环运行的设定	375
18.4.3 使用 DMAC/DTC 传送结束的事件链接时的注意事项	375
18.4.4 有关时钟的设定	375
18.4.5 模块停止功能的设定	375
19. I/O 端口	376
19.1 概要	376
19.2 输入 / 输出端口的结构	378
19.3 寄存器说明	383
19.3.1 端口方向寄存器 (PDR)	383
19.3.2 端口输出数据寄存器 (PODR)	384
19.3.3 端口输入数据寄存器 (PIDR)	385
19.3.4 端口模式寄存器 (PMR)	386
19.3.5 漏极开路控制寄存器 0 (ODR0)	387
19.3.6 漏极开路控制寄存器 1 (ODR1)	388
19.3.7 上拉控制寄存器 (PCR)	389
19.3.8 驱动能力控制寄存器 (DSCR)	390
19.3.9 未使用引脚的处理	390
20. 多功能引脚控制器 (MPC)	391
20.1 概要	391
20.2 寄存器说明	399
20.2.1 写保护寄存器 (PWPR)	399
20.2.2 P0n 引脚功能控制寄存器 (P0nPFS) (n=3、5、7)	400
20.2.3 P1n 引脚功能控制寄存器 (P1nPFS) (n=2 ~ 7)	401
20.2.4 P2n 引脚功能控制寄存器 (P2nPFS) (n=0 ~ 7)	403
20.2.5 P3n 引脚功能控制寄存器 (P3nPFS) (n=0 ~ 4)	405
20.2.6 P4n 引脚功能控制寄存器 (P4nPFS) (n=0 ~ 7)	407
20.2.7 P5n 引脚功能控制寄存器 (P5nPFS) (n=4 ~ 5)	407

20.2.8	PAn 引脚功能控制寄存器 (PAnPFS) (n=0 ~ 7)	408
20.2.9	PBn 引脚功能控制寄存器 (PBnPFS) (n=0 ~ 7)	410
20.2.10	PCn 引脚功能控制寄存器 (PCnPFS) (n=0 ~ 7)	412
20.2.11	PDn 引脚功能控制寄存器 (PDnPFS) (n=0 ~ 7)	413
20.2.12	PEn 引脚功能控制寄存器 (PEnPFS) (n=0 ~ 7)	414
20.2.13	PHn 引脚功能控制寄存器 (PHnPFS) (n=0 ~ 3)	415
20.2.14	PJn 引脚功能控制寄存器 (PJnPFS) (n=1、3)	416
20.2.15	CS 输出允许寄存器 (PFCSE)	417
20.2.16	地址输出允许寄存器 0 (PFAOE0)	418
20.2.17	地址输出允许寄存器 1 (PFAOE1)	419
20.2.18	外部总线控制寄存器 0 (PFBCR0)	420
20.2.19	外部总线控制寄存器 1 (PFBCR1)	421
20.3	外部总线接口的设定方法	422
20.4	使用时的注意事项	424
20.4.1	引脚输入 / 输出功能的设定步骤	424
20.4.2	设定 MPC 寄存器时的注意事项	424
20.4.3	使用模拟功能时的注意事项	424
21.	多功能定时器脉冲单元 2 (MTU2a)	425
21.1	概要	425
21.2	寄存器说明	430
21.2.1	定时器的控制寄存器 (TCR)	430
21.2.2	定时器的模式寄存器 (TMDR)	433
21.2.3	定时器的 I/O 控制寄存器 (TIOR)	435
21.2.4	定时器的比较匹配清除寄存器 (TCNTCMPCLR)	446
21.2.5	定时器的中断允许寄存器 (TIER)	447
21.2.6	定时器的状态寄存器 (TSR)	449
21.2.7	定时器的缓冲运行传送模式寄存器 (TBTM)	450
21.2.8	定时器的输入捕捉控制寄存器 (TICCR)	451
21.2.9	定时器的 A/D 转换开始请求控制寄存器 (TADCR)	452
21.2.10	定时器的 A/D 转换开始请求周期设定寄存器 A、B (TADCORA/B)	453
21.2.11	定时器的 A/D 转换开始请求周期设定缓冲寄存器 A、B (TADCOBRA/B)	453
21.2.12	定时器的计数器 (TCNT)	454
21.2.13	定时器的通用寄存器 (TGR)	454
21.2.14	定时器的启动寄存器 (TSTR)	455
21.2.15	定时器的同步寄存器 (TSYR)	456
21.2.16	定时器的读写允许寄存器 (TRWER)	457
21.2.17	定时器的输出主控允许寄存器 (TOER)	458
21.2.18	定时器的输出控制寄存器 1 (TOCR1)	459
21.2.19	定时器的输出控制寄存器 2 (TOCR2)	461
21.2.20	定时器的输出电平缓冲寄存器 (TOLBR)	463
21.2.21	定时器的门控寄存器 (TGCR)	464
21.2.22	定时器的副计数器 (TCNTS)	465
21.2.23	定时器的死区时间数据寄存器 (TDDR)	465
21.2.24	定时器的周期数据寄存器 (TCDR)	466
21.2.25	定时器的周期缓冲寄存器 (TCBR)	466
21.2.26	定时器的中断减少设定寄存器 (TITCR)	466
21.2.27	定时器的中断减少次数计数器 (TITCNT)	468
21.2.28	定时器的缓冲传送设定寄存器 (TBTER)	469
21.2.29	定时器的死区时间允许寄存器 (TDER)	470
21.2.30	定时器的波形控制寄存器 (TWCR)	471
21.2.31	噪声滤波器的控制寄存器 (NFCR)	472

21.2.32	和总线主控的接口	474
21.3	运行说明	475
21.3.1	概要	475
21.3.2	同步运行	480
21.3.3	缓冲运行	482
21.3.4	级联运行	486
21.3.5	PWM 模式	490
21.3.6	相位计数模式	494
21.3.7	复位同步 PWM 模式	500
21.3.8	互补 PWM 模式	503
21.3.9	A/D 转换开始请求的延迟功能	531
21.3.10	外部脉宽的测量功能	534
21.3.11	死区时间的补偿功能	535
21.3.12	在互补 PWM 的“波峰 / 波谷”进行的 TCNT 捕捉运行	536
21.3.13	噪声滤波器功能	537
21.4	中断源	538
21.4.1	中断源和优先级	538
21.4.2	DTC/DMAC 的启动	539
21.4.3	A/D 转换器的启动	539
21.5	运行时序	541
21.5.1	输入 / 输出时序	541
21.5.2	中断信号的时序	546
21.6	使用时的注意事项	549
21.6.1	模块停止功能的设定	549
21.6.2	输入时钟的制限事项	549
21.6.3	设定周期时的注意事项	549
21.6.4	TCNT 的写和清除的竞争	550
21.6.5	TCNT 的写和递增计数的竞争	550
21.6.6	TGR 的写和比较匹配的竞争	551
21.6.7	缓冲寄存器的写和比较匹配的竞争	551
21.6.8	缓冲寄存器的写和 TCNT 清除的竞争	552
21.6.9	TGR 的读和输入捕捉的竞争	552
21.6.10	TGR 的写和输入捕捉的竞争	553
21.6.11	缓冲寄存器的写和输入捕捉的竞争	554
21.6.12	级联中的 MTU2.TCNT 的写和上溢 / 下溢的竞争	555
21.6.13	互补 PWM 模式停止时的计数器值	556
21.6.14	互补 PWM 模式中的缓冲运行的设定	556
21.6.15	复位同步 PWM 模式的缓冲运行和比较匹配标志	556
21.6.16	复位同步 PWM 模式的上溢标志	557
21.6.17	上溢 / 下溢和计数器清除的竞争	558
21.6.18	TCNT 的写和上溢 / 下溢的竞争	558
21.6.19	从正常运行或者 PWM 模式 1 转移到复位同步 PWM 模式时的注意事项	559
21.6.20	互补 PWM 模式和复位同步 PWM 模式的输出电平	559
21.6.21	模块停止状态时的中断	559
21.6.22	级联中的 MTU1.TCNT 和 MTU2.TCNT 的同时输入捕捉	559
21.6.23	未使用互补 PWM 模式的输出保护功能时的注意事项	559
21.6.24	同步清除互补 PWM 模式时的异常动作的防止	560
21.6.25	比较匹配脉冲中断的连续输出	561
21.7	MTU 输出引脚的初始化方法	562
21.7.1	运行模式	562
21.7.2	因运行过程中的异常而重新设定时的运行	562
21.7.3	因运行过程中的异常而对引脚进行初始化的步骤以及模式转移的概要	563

21.8	通过 ELC 进行的链接运行	578
21.8.1	向 ELC 输出事件信号	578
21.8.2	通过接收 ELC 的事件信号进行的 MTU2 运行	578
21.8.3	通过接收 ELC 的事件信号进行的 MTU2 运行的注意事项	579
22.	端口输出允许 2 (POE2a)	580
22.1	概要	580
22.2	寄存器说明	583
22.2.1	输入电平控制 / 状态寄存器 1 (ICSR1)	583
22.2.2	输出电平控制 / 状态寄存器 1 (OCSR1)	585
22.2.3	输入电平控制 / 状态寄存器 2 (ICSR2)	586
22.2.4	软件端口输出允许寄存器 (SPOER)	587
22.2.5	端口输出允许控制寄存器 1 (POECR1)	588
22.2.6	端口输出允许控制寄存器 2 (POECR2)	588
22.2.7	输入电平控制 / 状态寄存器 3 (ICSR3)	590
22.3	运行说明	591
22.3.1	输入电平的检测	593
22.3.2	输出电平的比较	594
22.3.3	通过寄存器进行的高阻抗控制	594
22.3.4	通过振荡停止检测进行的高阻抗控制	594
22.3.5	通过接收 ELC 的事件信号进行的高阻抗控制	594
22.3.6	高阻抗的解除	594
22.4	中断	595
22.5	使用时的注意事项	595
22.5.1	向软件待机模式或者深度软件待机模式的转移	595
22.5.2	不使用 POE 时的注意事项	595
22.5.3	有关引脚的 MTU 功能的设定	595
22.5.4	通过接收 ELC 的事件信号进行的高阻抗控制的注意事项	595
23.	8 位定时器 (TMR)	596
23.1	概要	596
23.2	寄存器说明	601
23.2.1	定时器的计数器 (TCNT)	601
23.2.2	时间常数寄存器 A (TCORA)	602
23.2.3	时间常数寄存器 B (TCORB)	602
23.2.4	定时器的控制寄存器 (TCR)	603
23.2.5	定时器的计数器控制寄存器 (TCCR)	604
23.2.6	定时器的控制 / 状态寄存器 (TCSR)	606
23.2.7	定时器的计数器启动寄存器 (TCSTR)	608
23.3	运行说明	609
23.3.1	脉冲输出	609
23.3.2	复位输入	609
23.4	运行时序	610
23.4.1	TCNT 计数器的计数时序	610
23.4.2	比较匹配时的中断标志变为“1”的时序	611
23.4.3	比较匹配时的定时器输出时序	611
23.4.4	通过比较匹配进行的计数器清除时序	612
23.4.5	TCNT 计数器的外部复位时序	612
23.4.6	通过上溢使中断标志变为“1”的时序	613
23.5	级联时的运行	613
23.5.1	16 位计数模式	613
23.5.2	比较匹配计数模式	613

23.6	中断源	614
23.6.1	中断源和 DTC 启动	614
23.7	通过 ELC 进行的链接运行	615
23.7.1	向 ELC 输出事件信号	615
23.7.2	通过接收 ELC 的事件信号进行的 TMR 运行	615
23.7.3	通过接收 ELC 的事件信号进行的 TMR 运行的注意事项	615
23.8	使用时的注意事项	616
23.8.1	模块停止功能的设定	616
23.8.2	设定周期时的注意事项	616
23.8.3	TCNT 计数器的写和计数器清除的竞争	616
23.8.4	TCNT 计数器的写和递增计数的竞争	616
23.8.5	TCORA 寄存器、TCORB 寄存器的写和比较匹配的竞争	617
23.8.6	比较匹配 A 和比较匹配 B 的竞争	617
23.8.7	分频时钟的转换和 TCNT 计数器的运行	618
23.8.8	级联时的时钟源设定	619
23.8.9	比较匹配中断的连续输出	619
24.	比较匹配定时器 (CMT)	620
24.1	概要	620
24.2	寄存器说明	621
24.2.1	比较匹配定时器的启动寄存器 0 (CMSTR0)	621
24.2.2	比较匹配定时器的启动寄存器 1 (CMSTR1)	621
24.2.3	比较匹配定时器的控制寄存器 (CMCR)	622
24.2.4	比较匹配定时器的计数器 (CMCNT)	623
24.2.5	比较匹配定时器的常数寄存器 (CMCOR)	623
24.3	运行说明	624
24.3.1	周期计数	624
24.3.2	CMCNT 计数器的计数时序	624
24.4	中断	625
24.4.1	中断源	625
24.4.2	比较匹配中断的发生时序	625
24.5	通过 ELC 进行的链接运行	626
24.5.1	向 ELC 输出事件信号	626
24.5.2	通过接收 ELC 的事件信号进行的 CMT 运行	626
24.5.3	通过接收 ELC 的事件信号进行的 CMT 运行的注意事项	626
24.6	使用时的注意事项	627
24.6.1	模块停止功能的设定	627
24.6.2	CMCNT 计数器的写和比较匹配的竞争	627
24.6.3	CMCNT 计数器的写和递增计数的竞争	627
25.	实时时钟 (RTCb)	628
25.1	概要	628
25.2	寄存器说明	630
25.2.1	64Hz 计数器 (R64CNT)	630
25.2.2	秒计数器 (RSECCNT)	631
25.2.3	分钟计数器 (RMINCNT)	631
25.2.4	小时计数器 (RHRCNT)	632
25.2.5	星期计数器 (RWKCNT)	633
25.2.6	日计数器 (RDAYCNT)	634
25.2.7	月计数器 (RMONCNT)	634
25.2.8	年计数器 (RYRCNT)	635
25.2.9	秒闹钟寄存器 (RSECAR)	635

25.2.10	分钟闹钟寄存器 (RMINAR)	636
25.2.11	小时闹钟寄存器 (RHRAR)	637
25.2.12	星期闹钟寄存器 (RWKAR)	638
25.2.13	日闹钟寄存器 (RDAYAR)	639
25.2.14	月闹钟寄存器 (RMONAR)	639
25.2.15	年闹钟寄存器 (RYRAR)	640
25.2.16	年闹钟允许寄存器 (RYRAREN)	640
25.2.17	RTC 控制寄存器 1 (RCR1)	641
25.2.18	RTC 控制寄存器 2 (RCR2)	642
25.2.19	RTC 控制寄存器 3 (RCR3)	644
25.2.20	时间误差校正寄存器 (RADJ)	645
25.2.21	时间捕捉控制寄存器 y (RTCCRY) (y=0 ~ 2)	646
25.2.22	秒捕捉寄存器 y (RSECCPY) (y=0 ~ 2)	647
25.2.23	分钟捕捉寄存器 y (RMINCPY) (y=0 ~ 2)	648
25.2.24	小时捕捉寄存器 y (RHRCPY) (y=0 ~ 2)	648
25.2.25	日捕捉寄存器 y (RDAYCPY) (y=0 ~ 2)	649
25.2.26	月捕捉寄存器 y (RMONCPY) (y=0 ~ 2)	649
25.3	运行说明	650
25.3.1	接通电源后的寄存器初始设定概要	650
25.3.2	时钟设定步骤	650
25.3.3	时间设定步骤	651
25.3.4	30 秒调整步骤	651
25.3.5	64Hz 计数器和时间的读取步骤	652
25.3.6	闹钟功能	653
25.3.7	闹钟中断的禁止步骤	654
25.3.8	时钟误差校正功能	654
25.4	中断源	657
25.5	事件链接输出功能	658
25.5.1	中断处理和事件链接的关系	658
25.6	使用时的注意事项	659
25.6.1	有关计数运行时的寄存器写操作	659
25.6.2	有关周期中断的使用	659
25.6.3	有关 1Hz 时钟的输出	659
25.6.4	有关在设定寄存器后向低功耗模式的转移	659
25.6.5	读写寄存器时的注意事项	659
26.	看门狗定时器 (WDTA)	660
26.1	概要	660
26.2	寄存器说明	662
26.2.1	WDT 刷新寄存器 (WDTRR)	662
26.2.2	WDT 控制寄存器 (WDTCR)	663
26.2.3	WDT 状态寄存器 (WDTSR)	666
26.2.4	WDT 复位控制寄存器 (WDTRCR)	667
26.2.5	选项功能选择寄存器 0 (OFS0)	667
26.3	运行说明	668
26.3.1	各计数开始条件下的运行	668
26.3.2	WDTCR 寄存器和 WDTRCR 寄存器的写控制	671
26.3.3	刷新	672
26.3.4	状态标志	673
26.3.5	复位输出	673
26.3.6	中断源	674
26.3.7	递减计数器值的读操作	674

26.3.8	选项功能选择寄存器 0 (OFS0) 和 WDT 寄存器的对应	675
26.4	使用时的注意事项	675
26.4.1	有关刷新运行	675
27.	独立看门狗定时器 (IWDTa)	676
27.1	概要	676
27.2	寄存器说明	678
27.2.1	IWDT 刷新寄存器 (IWDTRR)	678
27.2.2	IWDT 控制寄存器 (IWDTCR)	679
27.2.3	IWDT 状态寄存器 (IWDTSR)	682
27.2.4	IWDT 复位控制寄存器 (IWDTRCR)	683
27.2.5	IWDT 计数停止控制寄存器 (IWDTCSTPR)	683
27.2.6	选项功能选择寄存器 0 (OFS0)	683
27.3	运行说明	684
27.3.1	各计数开始条件下的运行	684
27.3.2	IWDTCR 寄存器、IWDTRCR 寄存器和 IWDTCSTPR 寄存器的写控制	687
27.3.3	刷新	688
27.3.4	状态标志	690
27.3.5	复位输出	690
27.3.6	中断源	690
27.3.7	递减计数器值的读操作	691
27.3.8	选项功能选择寄存器 0 (OFS0) 和 IWDT 寄存器的对应	692
27.4	通过 ELC 进行的链接运行	692
27.5	使用时的注意事项	692
27.5.1	刷新	692
28.	串行通信接口 (SC1c、SC1d)	693
28.1	概要	693
28.2	寄存器说明	701
28.2.1	接收移位寄存器 (RSR)	701
28.2.2	接收数据寄存器 (RDR)	701
28.2.3	发送数据寄存器 (TDR)	701
28.2.4	发送移位寄存器 (TSR)	701
28.2.5	串行模式寄存器 (SMR)	702
28.2.6	串行控制寄存器 (SCR)	706
28.2.7	串行状态寄存器 (SSR)	711
28.2.8	智能卡模式寄存器 (SCMR)	715
28.2.9	位速率寄存器 (BRR)	717
28.2.10	串行扩展模式寄存器 (SEMR)	725
28.2.11	噪声滤波器的设定寄存器 (SNFR)	727
28.2.12	I ² C 模式寄存器 1 (SIMR1)	728
28.2.13	I ² C 模式寄存器 2 (SIMR2)	729
28.2.14	I ² C 模式寄存器 3 (SIMR3)	730
28.2.15	I ² C 状态寄存器 (SISR)	732
28.2.16	SPI 模式寄存器 (SPMR)	733
28.2.17	扩展串行模式有效寄存器 (ESMER)	735
28.2.18	控制寄存器 0 (CR0)	735
28.2.19	控制寄存器 1 (CR1)	736
28.2.20	控制寄存器 2 (CR2)	737
28.2.21	控制寄存器 3 (CR3)	738
28.2.22	端口控制寄存器 (PCR)	738
28.2.23	中断控制寄存器 (ICR)	739

28.2.24	状态寄存器 (STR)	740
28.2.25	状态清除寄存器 (STCR)	741
28.2.26	Control Field 0 数据寄存器 (CF0DR)	741
28.2.27	Control Field 0 比较允许寄存器 (CF0CR)	742
28.2.28	Control Field 0 接收数据寄存器 (CF0RR)	742
28.2.29	主 Control Field 1 数据寄存器 (PCF1DR)	742
28.2.30	次 Control Field 1 数据寄存器 (SCF1DR)	743
28.2.31	Control Field 1 比较允许寄存器 (CF1CR)	743
28.2.32	Control Field 1 接收数据寄存器 (CF1RR)	743
28.2.33	定时器的控制寄存器 (TCR)	744
28.2.34	定时器的模式寄存器 (TMR)	744
28.2.35	定时器的预分频寄存器 (TPRE)	745
28.2.36	定时器的计数器 (TCNT)	745
28.3	异步模式的运行	746
28.3.1	串行发送 / 接收格式	747
28.3.2	异步模式的接收数据采样时序和接收容限	748
28.3.3	时钟	749
28.3.4	CTS 和 RTS 功能	749
28.3.5	SCI 的初始化 (异步模式)	750
28.3.6	串行数据的发送 (异步模式)	751
28.3.7	串行数据的接收 (异步模式)	753
28.4	多处理器通信功能	757
28.4.1	多处理器串行数据的发送	758
28.4.2	多处理器串行数据的接收	759
28.5	时钟同步模式的运行	762
28.5.1	时钟	762
28.5.2	CTS 和 RTS 功能	762
28.5.3	SCI 的初始化 (时钟同步模式)	763
28.5.4	串行数据的发送 (时钟同步模式)	764
28.5.5	串行数据的接收 (时钟同步模式)	765
28.5.6	串行数据的同时发送和接收 (时钟同步模式)	768
28.6	智能卡接口模式的运行	769
28.6.1	连接例子	769
28.6.2	数据格式 (块传送模式除外)	770
28.6.3	块传送模式	771
28.6.4	接收数据的采样时序和接收容限	771
28.6.5	SCI 的初始化 (智能卡接口模式)	772
28.6.6	串行数据的发送 (块传送模式除外)	773
28.6.7	串行接收 (块传送模式除外)	776
28.6.8	时钟的输出控制	777
28.7	简易 I ² C 模式的运行	779
28.7.1	开始条件、重新开始条件和停止条件的生成	780
28.7.2	时钟同步	781
28.7.3	SSDA 输出延迟	782
28.7.4	SCI 的初始化 (简易 I ² C 模式)	783
28.7.5	主控发送 (简易 I ² C 模式)	784
28.7.6	主控接收 (简易 I ² C 模式)	786
28.8	简易 SPI 模式的运行	788
28.8.1	主控模式、从属模式和各引脚的状态	789
28.8.2	主控模式中的 SS 功能	789
28.8.3	从属模式中的 SS 功能	789
28.8.4	时钟和发送 / 接收数据的关系	789

28.8.5	SCI 的初始化 (简易 SPI 模式)	790
28.8.6	串行数据的发送和接收 (简易 SPI 模式)	790
28.9	扩展串行模式控制部的运行说明	791
28.9.1	串行通信协议	791
28.9.2	Start Frame 发送	792
28.9.3	Start Frame 接收	795
28.9.4	总线冲突检测功能	801
28.9.5	RXDX12 引脚输入的数字滤波器功能	802
28.9.6	位速率测量功能	803
28.9.7	RXDX12 接收数据的采样时序选择功能	804
28.9.8	定时器	805
28.10	噪声消除功能	807
28.11	中断源	808
28.11.1	TXI 中断和 RXI 中断的缓冲运行	808
28.11.2	串行通信接口模式和简易 SPI 模式的中断	808
28.11.3	智能卡接口模式的中断	809
28.11.4	简易 I ² C 模式的中断	809
28.11.5	扩展串行模式控制部的中断请求	810
28.12	事件链接功能	811
28.13	使用时的注意事项	812
28.13.1	模块停止功能的设定	812
28.13.2	有关中止的检测和处理	812
28.13.3	标记状态和中止的发送	812
28.13.4	有关接收错误标志和发送 (只限于时钟同步模式)	812
28.13.5	有关 TDR 寄存器的写操作	812
28.13.6	时钟同步发送时的限制事项	812
28.13.7	使用 DMAC 或者 DTC 时的限制事项	812
28.13.8	有关开始通信的注意事项	813
28.13.9	有关低功耗状态时的运行	813
28.13.10	时钟同步模式的外部时钟输入	815
28.13.11	简易 SPI 模式的限制事项	816
28.13.12	扩展串行模式控制部的使用限制事项 1	816
28.13.13	扩展串行模式控制部的使用限制事项 2	817
29.	I ² C 总线接口 (RIIC)	818
29.1	概要	818
29.2	寄存器说明	821
29.2.1	I ² C 总线控制寄存器 1 (ICCR1)	821
29.2.2	I ² C 总线控制寄存器 2 (ICCR2)	823
29.2.3	I ² C 总线模式寄存器 1 (ICMR1)	826
29.2.4	I ² C 总线模式寄存器 2 (ICMR2)	827
29.2.5	I ² C 总线模式寄存器 3 (ICMR3)	829
29.2.6	I ² C 总线功能允许寄存器 (ICFER)	831
29.2.7	I ² C 总线状态允许寄存器 (ICSER)	833
29.2.8	I ² C 总线中断允许寄存器 (ICIER)	835
29.2.9	I ² C 总线状态寄存器 1 (ICSR1)	837
29.2.10	I ² C 总线状态寄存器 2 (ICSR2)	840
29.2.11	从属地址寄存器 Ly (SARLy) (y=0 ~ 2)	843
29.2.12	从属地址寄存器 Uy (SARUy) (y=0 ~ 2)	844
29.2.13	I ² C 总线位速率低电平寄存器 (ICBRL)	845
29.2.14	I ² C 总线位速率高电平寄存器 (ICBRH)	846
29.2.15	I ² C 总线发送数据寄存器 (ICDRT)	848

29.2.16	I ² C 总线接收数据寄存器 (ICDRR)	848
29.2.17	I ² C 总线移位寄存器 (ICDRS)	848
29.3	运行说明	849
29.3.1	通信数据的格式	849
29.3.2	初始设定	850
29.3.3	主控发送	851
29.3.4	主控接收	855
29.3.5	从属发送	858
29.3.6	从属接收	861
29.4	SCL 同步电路	863
29.5	SDA 输出延迟功能	864
29.6	数字噪声滤波器电路	865
29.7	地址匹配检测功能	866
29.7.1	从属地址匹配检测功能	866
29.7.2	全呼地址检测功能	868
29.7.3	设备 ID 地址检测功能	868
29.7.4	主机地址检测功能	870
29.8	SCL 的 Low 电平自动保持功能	871
29.8.1	发送数据的误发送防止功能	871
29.8.2	NACK 接收传送中止功能	872
29.8.3	接收数据漏取防止功能	872
29.9	仲裁失败检测功能	874
29.9.1	主控仲裁失败检测功能 (MALE 位)	874
29.9.2	NACK 发送仲裁失败检测功能 (NALE 位)	876
29.9.3	从属仲裁失败检测功能 (SALE 位)	877
29.10	开始条件、重新开始条件和停止条件的发行功能	878
29.10.1	开始条件的发行	878
29.10.2	重新开始条件的发行	878
29.10.3	停止条件的发行	879
29.11	总线意外停机	880
29.11.1	超时检测功能	880
29.11.2	SCL 时钟追加输出功能	881
29.11.3	RIIC 复位 / 内部复位	881
29.12	SMBus 运行	882
29.12.1	SMBus 超时测量	882
29.12.2	数据包错误码 (PEC)	883
29.12.3	SMBus 主机通知协议 /Notify ARP master	883
29.13	中断源	884
29.13.1	ICTXI 中断和 ICRXI 中断的缓冲运行	884
29.14	复位状况	885
29.15	事件链接输出功能	886
29.15.1	中断处理和事件链接的关系	886
29.16	使用时的注意事项	886
29.16.1	模块停止功能的设定	886
29.16.2	有关开始通信的注意事项	886
30.	串行外围接口 (RSPI)	887
30.1	概要	887
30.2	寄存器说明	890
30.2.1	RSPI 控制寄存器 (SPCR)	890
30.2.2	RSPI 从属选择极性寄存器 (SSLP)	891
30.2.3	RSPI 引脚控制寄存器 (SPPCR)	892

30.2.4	RSPI 状态寄存器 (SPSR)	893
30.2.5	RSPI 数据寄存器 (SPDR)	894
30.2.6	RSPI 顺序控制寄存器 (SPSCR)	897
30.2.7	RSPI 顺序状态寄存器 (SPSSR)	898
30.2.8	RSPI 位速率寄存器 (SPBR)	899
30.2.9	RSPI 数据控制寄存器 (SPDCR)	900
30.2.10	RSPI 时钟延迟寄存器 (SPCKD)	901
30.2.11	RSPI 从属选择无效延迟寄存器 (SSLND)	902
30.2.12	RSPI 下次存取延迟寄存器 (SPND)	903
30.2.13	RSPI 控制寄存器 2 (SPCR2)	904
30.2.14	RSPI 命令寄存器 0 ~ 7 (SPCMD0 ~ SPCMD7)	905
30.3	运行说明	908
30.3.1	RSPI 运行概要	908
30.3.2	RSPI 引脚的控制	909
30.3.3	RSPI 系统结构例子	910
30.3.4	数据格式	916
30.3.5	传送格式	925
30.3.6	通信运行模式	927
30.3.7	发送缓冲器空中断 / 接收缓冲器满中断	929
30.3.8	错误检测	930
30.3.9	RSPI 的初始化	933
30.3.10	SPI 运行	934
30.3.11	时钟同步运行	945
30.3.12	主控模式的运行	945
30.3.13	从属模式的运行	950
30.3.14	错误处理	953
30.3.15	环回模式	955
30.3.16	奇偶校验位功能的自诊断	956
30.3.17	中断源	957
30.4	事件链接输出功能	958
30.4.1	接收缓冲器满事件的输出	958
30.4.2	发送缓冲器空事件的输出	958
30.4.3	模式故障 / 溢出 / 奇偶校验错误事件的输出	958
30.4.4	RSPI 空闲事件的输出	959
30.4.5	发送结束事件的输出	959
30.4.6	中断处理和事件链接的关系	959
30.5	使用时的注意事项	960
30.5.1	模块停止功能的设定	960
30.5.2	低功耗功能的注意事项	960
30.5.3	有关开始通信的注意事项	960
31.	CRC 运算器 (CRC)	961
31.1	概要	961
31.2	寄存器说明	962
31.2.1	CRC 控制寄存器 (CRCCR)	962
31.2.2	CRC 数据输入寄存器 (CRCDIR)	962
31.2.3	CRC 数据输出寄存器 (CRCDOR)	963
31.3	CRC 运算器的运行说明	964
31.4	使用时的注意事项	967
31.4.1	模块停止功能的设定	967
31.5	传送时的注意事项	967

32.	12 位 A/D 转换器 (S12ADb)	968
32.1	概要	968
32.2	寄存器说明	972
32.2.1	A/D 数据寄存器 y (ADDRy) (y=0 ~ 15)	972
32.2.2	A/D 数据双重化寄存器 (ADDBLDR)	974
32.2.3	A/D 温度传感器的数据寄存器 (ADTSDR)	975
32.2.4	A/D 内部基准电压的数据寄存器 (ADOCDR)	976
32.2.5	A/D 自诊断数据寄存器 (ADRD)	977
32.2.6	A/D 控制寄存器 (ADCSR)	978
32.2.7	A/D 通道选择寄存器 A (ADANSA)	981
32.2.8	A/D 通道选择寄存器 B (ADANSB)	981
32.2.9	A/D 转换值加法运算模式选择寄存器 (ADADS)	982
32.2.10	A/D 转换值加法运算次数选择寄存器 (ADADC)	983
32.2.11	A/D 控制扩展寄存器 (ADCER)	984
32.2.12	A/D 开始触发选择寄存器 (ADSTRGR)	986
32.2.13	A/D 转换扩展输入控制寄存器 (ADEXICR)	988
32.2.14	A/D 采样状态寄存器 n (ADSSTRn) (n=0 ~ 7、L、T、O)	989
32.2.15	采样 & 保持电路控制寄存器 (ADSHCR)	990
32.2.16	AD 断线检测控制寄存器 (ADDISCR)	991
32.3	运行说明	992
32.3.1	扫描的运行说明	992
32.3.2	单次扫描模式	992
32.3.3	连续扫描模式	998
32.3.4	分组扫描模式	1002
32.3.5	模拟输入的采样和扫描转换时间	1004
32.3.6	寄存器的自动清除功能的使用例子	1005
32.3.7	A/D 转换值加法运算功能	1005
32.3.8	断线检测辅助功能	1006
32.3.9	通过异步触发开始的 A/D 转换	1007
32.3.10	通过外围模块的同步触发开始的 A/D 转换	1007
32.4	中断源和 DMA 传送请求	1008
32.4.1	扫描结束时的中断请求	1008
32.5	事件链接功能	1008
32.5.1	向 ELC 输出事件的运行	1008
32.5.2	通过 ELC 的事件进行的 12 位 A/D 转换器运行	1008
32.5.3	通过 ELC 的事件进行的 12 位 A/D 转换器运行的注意事项	1008
32.6	A/D 转换精度的定义	1008
32.7	使用时的注意事项	1009
32.7.1	读数据寄存器的注意事项	1009
32.7.2	停止 A/D 转换时的注意事项	1009
32.7.3	开始和强制停止 A/D 转换时的运行时序	1009
32.7.4	扫描结束中断处理的注意事项	1009
32.7.5	模块停止功能的设定	1009
32.7.6	向低功耗状态转移时的注意事项	1009
32.7.7	有关容许信号源阻抗	1010
32.7.8	对绝对精度的影响	1010
32.7.9	模拟电源引脚以外的设定范围	1011
32.7.10	电路板设计时的注意事项	1011
32.7.11	噪声对策的注意事项	1012
32.7.12	使用 12 位 A/D 转换器输入时的端口设定	1012
32.7.13	使用断线检测辅助功能时的绝对精度误差	1012

33.	D/A 转换器 (DA)	1013
33.1	概要	1013
33.2	寄存器说明	1014
33.2.1	D/A 数据寄存器 m (DADRm) (m=0、1)	1014
33.2.2	D/A 控制寄存器 (DACR)	1015
33.2.3	DADRm 格式选择寄存器 (DADPR)	1016
33.3	运行说明	1017
33.4	事件链接运行的设定步骤	1017
33.5	事件链接运行时的注意事项	1017
33.6	使用时的注意事项	1018
33.6.1	模块停止功能的设定	1018
33.6.2	模块停止时的 D/A 转换器的运行	1018
33.6.3	软件待机模式时的 D/A 转换器的运行	1018
33.6.4	深度软件待机模式时的注意事项	1018
33.6.5	使用 D/A 转换器输出时的端口设定	1018
34.	温度传感器 (TEMPSa)	1019
34.1	概要	1019
34.2	寄存器说明	1020
34.2.1	温度传感器的控制寄存器 (TSCR)	1020
34.3	温度传感器的使用方法	1021
34.3.1	使用前的准备	1021
34.3.2	12 位 A/D 转换器的设定	1022
34.3.3	温度传感器的 A/D 转换结果	1022
34.3.4	温度传感器的使用步骤	1023
34.3.5	温度传感器输出的 A/D 转换时序	1024
34.4	使用时的注意事项	1024
34.4.1	模块停止功能的设定	1024
35.	比较器 A (CMPA)	1025
35.1	概要	1025
35.2	寄存器说明	1027
35.2.1	电压监视 1 电路 / 比较器 A1 的控制寄存器 1 (LVD1CR1)	1027
35.2.2	电压监视 1 电路 / 比较器 A1 的状态寄存器 (LVD1SR)	1027
35.2.3	电压监视 2 电路 / 比较器 A2 的控制寄存器 1 (LVD2CR1)	1028
35.2.4	电压监视 2 电路 / 比较器 A2 的状态寄存器 (LVD2SR)	1028
35.2.5	电压监视电路 / 比较器 A 的控制寄存器 (LVCMPCR)	1029
35.2.6	电压监视 1 电路 / 比较器 A1 的控制寄存器 0 (LVD1CR0)	1030
35.2.7	电压监视 2 电路 / 比较器 A2 的控制寄存器 0 (LVD2CR0)	1031
35.3	比较结果的监视	1032
35.3.1	比较器 A1 的监视	1032
35.3.2	比较器 A2 的监视	1032
35.4	运行说明	1033
35.4.1	比较器 A1	1033
35.4.2	比较器 A2	1035
35.5	比较器 A1 中断和比较器 A2 中断	1037
35.5.1	非屏蔽中断	1037
35.5.2	可屏蔽中断	1037
35.6	事件链接输出功能	1037
35.7	中断处理和事件链接的关系	1037

36.	比较器 B (CMPB)	1038
36.1	概要	1038
36.2	寄存器说明	1039
36.2.1	比较器 B 控制寄存器 1 (CPBCNT1)	1039
36.2.2	比较器 B 标志寄存器 (CPBFLG)	1039
36.2.3	比较器 B 中断控制寄存器 (CPBINT)	1040
36.2.4	比较器 B 的滤波器选择寄存器 (CPBF)	1041
36.3	运行说明	1042
36.3.1	比较器 Bi 的数字滤波器 (i=0、1)	1043
36.4	比较器 B0 中断和比较器 B1 中断	1044
36.5	事件链接输出功能	1044
36.5.1	中断处理和事件链接的关系	1044
36.6	使用时的注意事项	1044
36.6.1	模块停止功能的设定	1044
37.	数据运算电路 (DOC)	1045
37.1	概要	1045
37.2	寄存器说明	1046
37.2.1	DOC 控制寄存器 (DOCR)	1046
37.2.2	DOC 数据输入寄存器 (DODIR)	1047
37.2.3	DOC 数据设定寄存器 (DODSR)	1047
37.3	运算说明	1048
37.3.1	数据比较模式	1048
37.3.2	数据加法运算模式	1048
37.3.3	数据减法运算模式	1049
37.4	中断请求	1050
37.5	使用时的注意事项	1050
37.5.1	模块停止功能的设定	1050
38.	RAM	1051
38.1	概要	1051
38.2	运行说明	1051
38.2.1	数据的保持	1051
38.2.2	低功耗功能	1051
39.	ROM (保存代码的闪存)	1052
39.1	概要	1052
39.2	寄存器说明	1054
39.2.1	闪存编程 / 擦除保护寄存器 (FWEPROR)	1054
39.2.2	闪存模式寄存器 (FMODR)	1055
39.2.3	闪存存取状态寄存器 (FASTAT)	1056
39.2.4	闪存存取错误中断允许寄存器 (FAEINT)	1058
39.2.5	FCU RAM 允许寄存器 (FCURAME)	1059
39.2.6	闪存状态寄存器 0 (FSTATR0)	1060
39.2.7	闪存状态寄存器 1 (FSTATR1)	1062
39.2.8	闪存就绪中断允许寄存器 (FRDYIE)	1063
39.2.9	闪存 P/E 模式入口寄存器 (FENTRYR)	1064
39.2.10	闪存保护寄存器 (FPROTR)	1065
39.2.11	闪存复位寄存器 (FRESETR)	1066
39.2.12	FCU 命令寄存器 (FCMDR)	1067
39.2.13	FCU 处理转换寄存器 (FCPSR)	1068
39.2.14	闪存 P/E 状态寄存器 (FPESTAT)	1068

39.2.15	外围时钟通知寄存器 (PCKAR)	1069
39.3	ROM 的存储区结构	1070
39.4	块结构	1071
39.5	ROM 相关的运行模式	1072
39.6	ROM 的编程 / 擦除	1073
39.6.1	FCU 的模式	1073
39.6.2	FCU 命令一览表	1075
39.6.3	FCU 的模式和命令的关系	1077
39.6.4	FCU 命令的使用方法	1078
39.7	挂起	1095
39.7.1	编程和擦除时的挂起 (挂起优先模式)	1095
39.7.2	编程和擦除时的挂起 (编程和擦除优先模式)	1096
39.8	保护	1097
39.8.1	软件保护	1097
39.8.2	错误保护	1097
39.9	用户引导模式	1099
39.10	引导模式	1099
39.10.1	系统结构	1099
39.10.2	引导模式的状态转移	1100
39.10.3	位速率的自动调整	1102
39.10.4	ID 码保护 (引导模式)	1103
39.10.5	UB 码	1104
39.10.6	查询设定主机命令的等待状态	1104
39.10.7	ID 码等待状态	1114
39.10.8	编程 / 擦除主机命令等待状态	1115
39.11	On-chip 调试器的 ID 码保护	1123
39.12	ROM 码保护	1123
39.13	使用时的注意事项	1124
40.	E2 数据闪存 (保存数据的闪存)	1125
40.1	概要	1125
40.2	寄存器说明	1127
40.2.1	闪存模式寄存器 (FMODR)	1127
40.2.2	闪存存取状态寄存器 (FASTAT)	1128
40.2.3	闪存存取错误中断允许寄存器 (FAEINT)	1130
40.2.4	E2 数据闪存读允许寄存器 0 (DFLRE0)	1131
40.2.5	E2 数据闪存编程 / 擦除允许寄存器 0 (DFLWE0)	1132
40.2.6	闪存 P/E 模式入口寄存器 (FENTRYR)	1133
40.2.7	E2 数据闪存空白检查控制寄存器 (DFLBCCNT)	1134
40.2.8	E2 数据闪存空白检查状态寄存器 (DFLBCSTAT)	1134
40.3	E2 数据闪存的存储区结构	1135
40.4	块结构	1135
40.5	E2 数据闪存相关的运行模式	1136
40.6	E2 数据闪存的编程 / 擦除	1137
40.6.1	FCU 的模式	1137
40.6.2	FCU 命令一览表	1139
40.6.3	FCU 的模式和命令的关系	1140
40.6.4	FCU 命令的使用方法	1141
40.7	保护	1144
40.7.1	软件保护	1144
40.7.2	错误保护	1144

40.8	引导模式	1146
40.8.1	查询设定主机命令	1146
40.8.2	编程 / 擦除主机命令	1147
40.9	使用时的注意事项	1148
41.	电特性	1149
41.1	绝对最大额定值	1149
41.2	DC 特性	1150
41.3	AC 特性	1155
41.4	时钟时序	1157
41.4.1	复位时序	1161
41.4.2	从低功耗状态恢复的时序	1162
41.4.3	控制信号的时序	1163
41.4.4	总线时序	1164
41.4.5	内部外围模块的时序	1172
41.5	A/D 转换特性	1181
41.6	D/A 转换特性	1183
41.7	温度传感器特性	1184
41.8	比较器特性	1185
41.9	上电复位电路和电压检测电路的特性	1186
41.10	振荡停止检测时序	1190
41.11	ROM（保存代码的闪存）特性	1190
41.12	E2 数据闪存（保存数据的闪存）特性	1192
附录	1195
附录 1.	各运行模式中的端口状态	1195
附录 2.	封装尺寸图	1198

50MHz、32 位 RX MCU、78 DMIPS、最大 512K 字节闪存、12 位 A/D、
10 位 D/A、ELC、MPC、RTC、最多 9 个通信功能、符合 IEC60730 标准的功能

特点

■ 内置 32 位 RX CPU 内核

- 最大工作频率 50MHz
- 78 DMIPS 性能（以 50MHz 运行时）
- 32×32 → 64 位运算结果（1 条指令）的累加器
- 32×32 位乘法、除法器（乘法运算指令为 1 个 CPU 时钟）
- 高速中断
- 5 级流水线的 CISC 哈佛体系结构
- 可变长指令格式：大幅度地缩短代码
- 内部调试电路

■ 低功耗功能

- 以 1.62V ~ 5.5V 运行的单电源
- 能以 1.62V 运行（最大 20MHz）
- 使用 RTC 的深度软件待机模式
- 4 种低功耗模式

■ 内部编码闪存（无等待）

- 以 50MHz 运行、读周期为 20ns
- 当 CPU 全速读时，无等待
- 128K ~ 512K 字节的容量
- SCI 的用户编程
- 能以 1.62V 改写
- 用于指令、操作数

■ 内部数据闪存

- 8K 字节（改写次数：100000 次）
- 不对 CPU 外加负荷的编程 / 擦除

■ 内部 SRAM（无等待）

- 20K ~ 64K 字节的容量

■ DMA

- DMA：内置 4 个通道
- DTC：4 种传送模式

■ ELC

- 能不通过中断通过事件信号使模块运行
- 能在 CPU 睡眠模式下使模块运行

■ 复位和电源电压控制

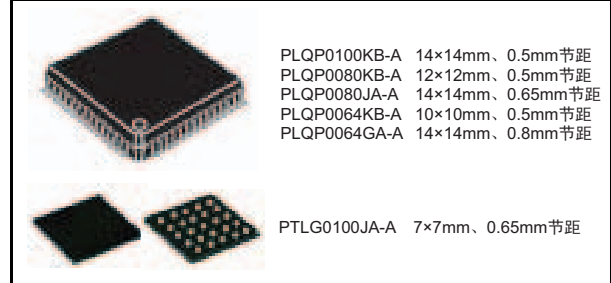
- 支持上电复位（POR）等 9 种复位
- 能设定低电压检测功能（LVD）

■ 时钟功能

- 外部时钟的输入频率：~ 20MHz
- 用于副时钟的振荡器频率：32.768kHz
- PLL 电路输入 4MHz ~ 12.5MHz
- 内置低速内部振荡器、高速内部振荡器、IWDT 专用低速内部振荡器
- 生成 32.768kHz 的 RTC 专用时钟
- 内置时钟频率精度测量电路（CAC）

■ 内置实时时钟

- 校正功能（30 秒、瑞年、误差）
- 事件捕捉功能
- 通过输入外部引脚的事件捕捉时间
- 能通过 RTC 从深度软件待机模式返回



■ 内置独立看门狗定时器

- 125kHz 的 IWDG 专用低速内部振荡器时钟运行

■ 内置符合 IEC60730 标准的功能

- A/D 转换器自诊断功能 / 断线检测辅助功能、时钟频率精度测量电路、独立看门狗定时器、RAM 测试辅助功能等

■ 内置最多 9 个通信功能

- 支持多种功能的 SCI（最多 7ch）异步模式 / 时钟同步模式 / 智能卡接口模式
- I²C 总线接口
最大以 1Mbps 传送、支持 SMBus（1ch）
- RSPI（1ch）

■ 外部地址空间

- 4 个 CS 区域（4×16M 字节）
- 各区域可选择 8/16 位总线空间

■ 最多 14 个扩展定时器功能

- 16 位 MTU2(6ch)：输入捕捉、输出比较、互补 PWM 输出、相位计数模式
- 8 位 TMR(4ch)
- 16 位 CMT(4ch)

■ 内置 12 位 A/D 转换器

- 最小能以 1μs 进行转换
- 内置采样 & 保持电路（3ch）
- 能进行 3ch 同时采样
- 内置自诊断功能 / 模拟输入断线检测辅助功能

■ 内置 10 位 D/A 转换器

■ 内置模拟比较器

■ 内置可编程输入 / 输出端口

- 5V 容许、漏极开路、输入上拉、驱动能力切换功能

■ MPC

- 能从多处选择外围功能的输入 / 输出引脚

■ 内置温度传感器

■ 工作环境温度

- -40 °C ~ +85 °C
- -40 °C ~ +105 °C

1. 概要

1.1 规格概要

规格概要如表 1.1 所示，各封装的功能比较一览表如表 1.2 所示。

在表 1.1 的规格中记载了最大规格，外围模块的通道数根据封装的引脚数而不同。详细内容请参照“表 1.2 各封装的功能比较一览表”。

表 1.1 规格概要 (1/4)

分类	模块 / 功能	说明
CPU	中央处理器	<ul style="list-style-type: none"> 最大工作频率：50MHz 32 位 RX CPU 指令的最短执行时间：1 条指令 1 个时钟 地址空间：4G 字节、线性地址 寄存器 <ul style="list-style-type: none"> 通用寄存器：32 位 ×16 个 控制寄存器：32 位 ×8 个 累加器：64 位 ×1 个 基本指令：73 种 DSP 功能指令：9 种 寻址方式：10 种 数据排列 <ul style="list-style-type: none"> 指令：小端法 数据：可选择小端法或者大端法 32 位乘法器：32 位 ×32 位 →64 位 除法器：32 位 ÷32 位 →32 位 桶式移位器：32 位
存储器	ROM	<ul style="list-style-type: none"> 容量：128K/256K/384K/512K 字节 100MHz、无等待存取 板上编程：3 种 板外编程（并行编程器模式）
	RAM	<ul style="list-style-type: none"> 容量：20K/32K/64K 字节 100MHz、无等待存取
	E2 数据闪存	<ul style="list-style-type: none"> 容量：8K 字节 编程 / 擦除次数：100000 次
MCU 运行模式		单芯片模式、内部 ROM 有效扩展模式、内部 ROM 无效扩展模式（软件切换）
时钟	时钟发生电路	<ul style="list-style-type: none"> 主时钟振荡器、副时钟振荡器、低速和高速内部振荡器、PLL 频率合成器、IWDI 专用内部振荡器 振荡停止检测：有 时钟频率精度测定电路（CAC）：有 能分别设定系统时钟（ICLK）、外围模块时钟（PCLK）、外部总线时钟（BCLK）和 FlashIF 时钟（FCLK）。 CPU 和总线主控器等系统设备与 ICLK 同步：Max 50MHz 外围模块与 PCLK 同步：Max 32MHz 连接外部总线的设备与 BCLK 同步：Max 12.5MHz 闪存外围电路与 FCLK 同步：Max 32MHz
复位		引脚复位、上电复位、电压监视复位、看门狗定时器复位、独立看门狗定时器复位、深度软件待机复位、软件复位
电压检测	电压检测电路（LVDAa）	<ul style="list-style-type: none"> 如果 VCC 低于电压检测电平，就发生内部复位或者内部中断。 电压检测 0 能从 4 个电平中选择检测电压 电压检测 1 能从 16 个电平中选择检测电压 电压检测 2 能从 16 个电平中选择检测电压

表 1.1 规格概要 (2/4)

分类	模块 / 功能	说明
低功耗	低功耗功能	<ul style="list-style-type: none"> 模块停止功能 4 种低功耗模式 睡眠模式、全模块时钟停止模式、软件待机模式、深度软件待机模式
中断	中断控制器 (ICUb)	<ul style="list-style-type: none"> 中断向量数: 117 外部中断: 9 个中断源 (NMI、IRQ0 ~ IRQ7 引脚) 非屏蔽中断: 6 个中断源 (NMI 引脚、振荡停止检测中断、电压监视 1 中断、电压监视 2 中断、WDT 中断、IWDI 中断) 能设定 16 个中断优先级
外部总线扩展		<ul style="list-style-type: none"> 将外部地址空间分为 4 个区域 (CS0 ~ CS3) 进行管理 各区域: 16M 字节 (CS0 ~ CS3) 各区域能输出片选 (CS0# ~ CS3#) 各区域能选择 8 位总线宽度 / 16 位总线宽度。 各区域能设定字节序 (只限于数据)。 总线格式: 能选择分离总线或者多路复用总线 能控制等待。 写缓冲器功能
DMA	DMA 控制器 (DMACA)	<ul style="list-style-type: none"> 4 个通道 传送模式: 正常传送模式、重复传送模式、块传送模式 启动源: 软件触发、外部中断、外围功能中断
	数据传送控制器 (DTCa)	<ul style="list-style-type: none"> 传送模式: 正常传送模式、重复传送模式、块传送模式 启动源: 通过中断源启动 有链传送功能
I/O 端口	可编程输入 / 输出端口	<p>100 引脚 LQFP/80 引脚 LQFP/64 引脚 LQFP</p> <ul style="list-style-type: none"> 输入 / 输出: 84/64/48 输入: 1/1/1 上拉电阻: 85/65/49 漏极开路输出: 54/44/35 5V 容许: 4/4/2
事件链接控制器 (ELC)		<ul style="list-style-type: none"> 能将 59 种事件信号直接链接到模块 定时器类的模块能选择输入事件时的运行 能进行端口 B 和端口 E 的事件链接运行
多功能引脚控制器 (MPC)		能从多个引脚选择输入 / 输出功能
定时器	多功能定时器脉冲单元 2 (MTU2a)	<ul style="list-style-type: none"> (16 位 × 6 个通道) × 1 个单元 以 16 位定时器 6 个通道为基础, 最多能输入 / 输出 16 个脉冲以及输入 3 个脉冲。 各通道可选择 8 种或者 7 种计数时钟 (PCLK/1、PCLK/4、PCLK/16、PCLK/64、PCLK/256、PCLK/1024、MTCLKA、MTCLKB、MTCLKC、MTCLKD) (通道 5 可选择 4 种) 输入捕捉功能 21 个输出比较寄存器兼输入捕捉寄存器 脉冲输出模式 互补 PWM 输出模式 复位同步 PWM 模式 相位计数模式 能生成 A/D 转换器的转换开始触发
	端口输出允许 2 (POE2a)	控制 MTU 波形输出引脚的高阻抗

表 1.1 规格概要 (3/4)

分类	模块 / 功能	说明
定时器	8 位定时器 (TMR)	<ul style="list-style-type: none"> • (8 位 × 2 个通道) × 2 个单元 • 可选择 7 种内部时钟 (PCLK/1、PCLK/2、PCLK/8、PCLK/32、PCLK/64、PCLK/1024、PCLK/8192) 和外部时钟 • 能进行任意占空比的脉冲输出和 PWM 输出 • 能将 2 个通道进行级联运行, 用作 16 位定时器。 • 能生成 SCI5、SCI6、SCI12 的波特率时钟
	比较匹配定时器 (CMT)	<ul style="list-style-type: none"> • (16 位 × 2 个通道) × 2 个单元 • 可选择 4 种时钟 (PCLK/8、PCLK/32、PCLK/128、PCLK/512)
	看门狗定时器 (WDTA)	<ul style="list-style-type: none"> • 14 位 × 1 个通道 • 可选择 6 种计数时钟 (PCLK/4、PCLK/64、PCLK/128、PCLK/512、PCLK/2048、PCLK/8192)
	独立看门狗定时器 (IWDTa)	<ul style="list-style-type: none"> • 14 位 × 1 个通道 • 计数时钟: IWDT 专用内部振荡器 1 分频、16 分频、32 分频、64 分频、128 分频、256 分频
	实时时钟 (RTCb)	<ul style="list-style-type: none"> • 时钟源: 通过副时钟运行 • 时钟 / 日历功能 • 中断源: 闹钟中断、周期中断、进位中断 • 3 个时间值捕捉功能
通信功能	串行通信接口 (SC1c、SC1d)	<ul style="list-style-type: none"> • 7 个通道 (通道 0、1、5、6、8、9: SC1c, 通道 12: SC1d) • 串行通信方式: 异步 / 时钟同步 / 智能卡接口 • 能通过内部波特率发生器选择任意的位速率 • 能选择 LSB first 或者 MSB first • 能输入 TMR 的平均传送率时钟 (SCI5、SCI6、SCI12) • 简易 IIC 功能 • 简易 SPI 功能 • 支持主控 / 从属模式 (只限 SC1d) • 由开始帧和信息帧构成 (只限 SC1d)
	I ² C 总线接口 (RIIC)	<ul style="list-style-type: none"> • 1 个通道 • 通信格式: I²C 总线格式 / SMBus 格式 • 能选择主控或者从属 • 支持优先模式
	串行外围接口 (RSPI)	<ul style="list-style-type: none"> • 1 个通道 • 传送功能 能使用 MOSI (Master Out Slave In)、MISO (Master In Slave Out)、SSL (Slave Select) 信号和 RSPCK (RSPI Clock), 通过 SPI 运行 (4 线式) / 时钟同步运行 (3 线式) 进行串行通信 • 能选择主控 / 从属模式 • 数据格式 • 能选择 LSB first 或者 MSB first • 能选择传送位长 (8 ~ 16、20、24、32 位) • 发送 / 接收缓冲器为 128 位 1 次发送 / 接收最多传送 4 帧 (1 帧最多为 32 位) • 发送 / 接收缓冲器为双缓冲器结构

表 1.1 规格概要 (4/4)

分类	模块 / 功能	说明
12 位 A/D 转换器 (S12ADb)		<ul style="list-style-type: none"> • 12 位 (16 个通道 × 1 个单元) • 分辨率: 12 位 • 最短转换时间: 每个通道 1.0μs (以 ADCLK=50MHz 运行时) • 运行模式 扫描模式 (单次扫描模式 / 连续扫描模式、组扫描模式) • 采样 & 保持功能 • A/D 转换器的自诊断功能 • 模拟输入断线检测辅助功能 • 双触发模式 (A/D 转换数据双重化功能) • A/D 转换开始条件 软件触发、定时器 (MTU) 的触发、外部触发、ELC
温度传感器 (TEMPSa)		<ul style="list-style-type: none"> • 输出根据温度变化的电压 • PGA 增益转换: 能根据电压范围转换 4 个阶段
D/A 转换器 (DA)		<ul style="list-style-type: none"> • 2 个通道 • 分辨率: 10 位 • 输出电压: 0V ~ VREFH
CRC 运算器 (CRC)		<ul style="list-style-type: none"> • 对以 8 位为单位的任意数据长度生成 CRC 码 • 能从 3 个多项式中选择: X^8+X^2+X+1、$X^{16}+X^{15}+X^2+1$、$X^{16}+X^{12}+X^5+1$ • 能选择是生成 LSB first 通信的 CRC 码, 还是生成 MSB first 通信的 CRC 码。
比较器 A (CMPA)		<ul style="list-style-type: none"> • 2 个通道 • 基准电压和模拟输入电压的比较功能
比较器 B (CMPB)		<ul style="list-style-type: none"> • 2 个通道 • 基准电压和模拟输入电压的比较功能
数据运算电路 (DOC)		对 16 位的数据进行比较、加法运算和减法运算的功能
电源电压 / 工作频率		VCC=1.62V ~ 1.8V: 20MHz、VCC=1.8V ~ 2.7V: 32MHz、 VCC=2.7V ~ 5.5V: 50MHz
消耗电流		14mA@50MHz(typ)
工作环境温度		-40 ~ +105°C
封装		100 引脚 TFLGA (PTLG0100JA-A) 100 引脚 LQFP (PLQP0100KB-A) 80 引脚 LQFP (PLQP0080KB-A) 80 引脚 LQFP (PLQP0080JA-A) 64 引脚 LQFP (PLQP0064KB-A) 64 引脚 LQFP (PLQP0064GA-A)
内部调试系统		E1 仿真器 (FINE 接口)

表 1.2 各封装的功能比较一览表

模块功能		RX210 群		
		100 引脚	80 引脚	64 引脚
外部总线	外部总线宽度	16 位	不支持	
中断	外部中断	NMI、IRQ0 ~ IRQ7		
DMA	DMA 控制器	4 个通道 (DMAC0 ~ DMAC3)		
	数据传送控制器	有		
定时器	多功能定时器脉冲单元 2	6 个通道 (MTU0 ~ MTU5)		
	端口输出允许 2	POE0# ~ POE3#、POE8#		
	8 位定时器	2 个通道 × 2 个单元		
	比较匹配定时器	2 个通道 × 2 个单元		
	实时时钟	有		
	看门狗定时器	有		
	独立看门狗定时器	有		
通信功能	串行通信接口 (SC1c)	6 个通道 (SC10、1、5、6、8、9)	5 个通道 (SC11、5、6、8、9)	
	串行通信接口 (SC1d)	1 个通道 (SC112)		
	I ² C 总线接口	1 个通道		
	串行外围接口	1 个通道		
12 位 A/D 转换器		16 个通道 (AN000 ~ AN015)	14 个通道 (AN000 ~ AN013)	12 个通道 (AN000 ~ AN004、 AN006、AN008 ~ AN013)
温度传感器		有		
D/A 转换器		2 个通道		
CRC 运算器		有		
事件链接控制器		有		
比较器 A		2 个通道		
比较器 B		2 个通道		
封装		100 引脚 TFLGA 100 引脚 LQFP	80 引脚 LQFP	64 引脚 LQFP

1.2 产品一览表

产品一览表如表 1.3 所示，产品型号、存储容量和封装图 1.1 所示。

表 1.3 产品一览表 (1/2)

群	型号	封装	ROM 容量	RAM 容量	E2 数据闪存	工作频率 (Max)	工作环境温度			
RX210	R5F52108ADFP	PLQP0100KB-A	512K 字节	64K 字节	8K 字节	50MHz	-40 ~ +85 °C			
	R5F52108ADFN	PLQP0080KB-A								
	R5F52108ADFF	PLQP0080JA-A (注 1)								
	R5F52108ADFM	PLQP0064KB-A								
	R5F52108ADFK	PLQP0064GA-A (注 1)								
	R5F52108ADLJ	PTLG0100JA-A								
	R5F52107ADFP	PLQP0100KB-A	384K 字节							
	R5F52107ADFN	PLQP0080KB-A								
	R5F52107ADFF	PLQP0080JA-A (注 1)								
	R5F52107ADFM	PLQP0064KB-A								
	R5F52107ADFK	PLQP0064GA-A (注 1)								
	R5F52107ADLJ	PTLG0100JA-A								
	R5F52106ADFP	PLQP0100KB-A	256K 字节	32K 字节						
	R5F52106ADFN	PLQP0080KB-A								
	R5F52106ADFF	PLQP0080JA-A (注 1)								
	R5F52106ADFM	PLQP0064KB-A								
	R5F52106ADFK	PLQP0064GA-A (注 1)								
	R5F52106ADLJ	PTLG0100JA-A								
	R5F52105ADFP	PLQP0100KB-A	128K 字节	20K 字节						
	R5F52105ADFN	PLQP0080KB-A								
	R5F52105ADFF	PLQP0080JA-A (注 1)								
	R5F52105ADFM	PLQP0064KB-A								
	R5F52105ADFK	PLQP0064GA-A (注 1)								
	R5F52105ADLJ	PTLG0100JA-A								
	R5F52108AGFP	PLQP0100KB-A	512K 字节	64K 字节				8K 字节	50MHz	-40 ~ +105 °C
	R5F52108AGFN	PLQP0080KB-A								
	R5F52108AGFF	PLQP0080JA-A (注 1)								
	R5F52108AGFM	PLQP0064KB-A								
	R5F52108AGFK	PLQP0064GA-A (注 1)								
	R5F52108AGLJ	PTLG0100JA-A								
R5F52107AGFP	PLQP0100KB-A	384K 字节								
R5F52107AGFN	PLQP0080KB-A									
R5F52107AGFF	PLQP0080JA-A (注 1)									
R5F52107AGFM	PLQP0064KB-A									
R5F52107AGFK	PLQP0064GA-A (注 1)									
R5F52107AGLJ	PTLG0100JA-A									

表 1.3 产品一览表 (2/2)

群	型号	封装	ROM 容量	RAM 容量	E2 数据闪存	工作频率 (Max)	工作环境温度
RX210	R5F52106AGFP	PLQP0100KB-A	256K 字节	32K 字节	8K 字节	50MHz	-40~+105℃
	R5F52106AGFN	PLQP0080KB-A					
	R5F52106AGFF	PLQP0080JA-A (注1)					
	R5F52106AGFM	PLQP0064KB-A					
	R5F52106AGFK	PLQP0064GA-A (注1)					
	R5F52106AGLJ	PTLG0100JA-A					
	R5F52105AGFP	PLQP0100KB-A	128K 字节	20K 字节			
	R5F52105AGFN	PLQP0080KB-A					
	R5F52105AGFF	PLQP0080JA-A (注1)					
	R5F52105AGFM	PLQP0064KB-A					
	R5F52105AGFK	PLQP0064GA-A (注1)					
	R5F52105AGLJ	PTLG0100JA-A					

注 1. 计划中

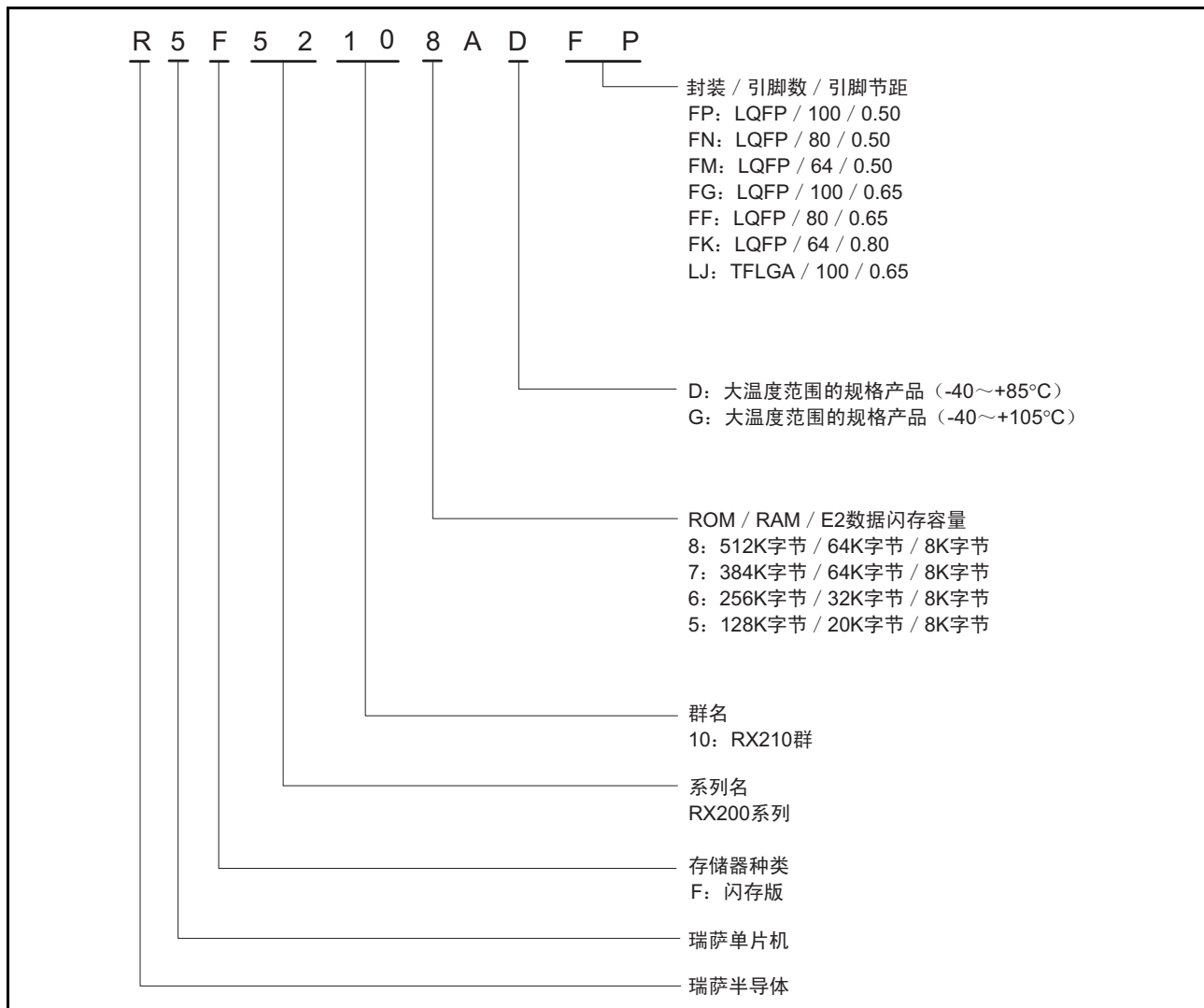


图 1.1 产品型号、存储容量和封装

1.3 框图

框图如图 1.2 所示。

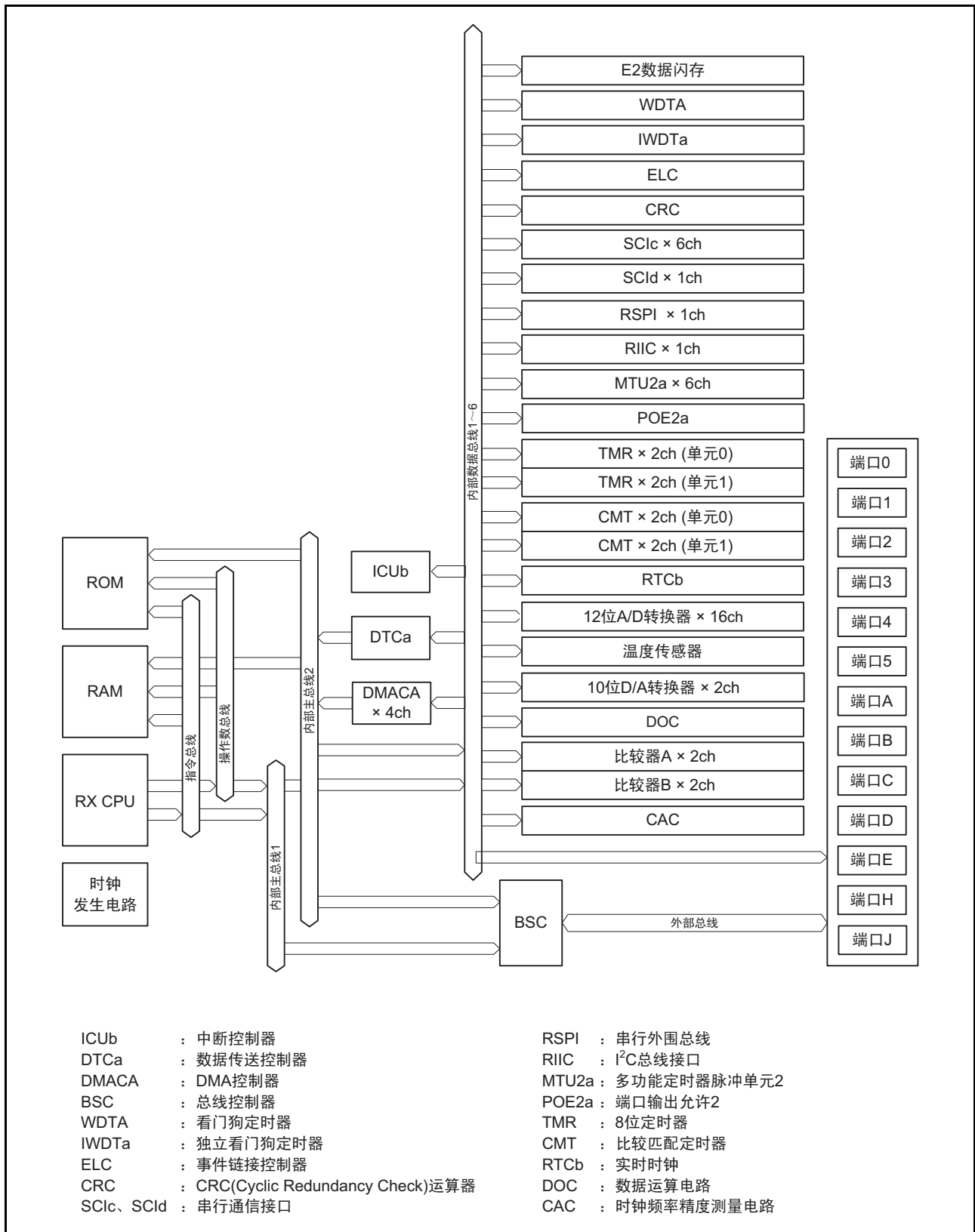


图 1.2 框图

1.4 引脚功能

引脚功能一览如表 1.4 所示。

表 1.4 引脚功能一览表 (1/4)

分类	引脚名	输入 / 输出	功能
电源	VCC	输入	电源引脚 必须连接系统电源。
	VCL	输入	必须通过 0.1 μ F 的电容器连接 VSS，并且使电容器靠近引脚。
	VSS	输入	接地引脚 必须连接系统电源 (0V)。
时钟	XTAL	输出	连接晶体谐振器的引脚
	EXTAL	输入	EXTAL 引脚也能输入外部时钟。
	BCLK	输出	用于外部设备的外部总线时钟的输出引脚
	XCIN	输入	副时钟振荡器的输入 / 输出引脚
	XCOU	输出	必须在 XCOU 和 XCIN 之间连接晶体谐振器。
运行模式的控制	MD	输入	设定运行模式。不能在运行中改变这些引脚。
系统控制	RES#	输入	复位引脚 当此引脚为 Low 电平时，进入复位状态。
CAC	CACREF	输入	时钟频率精度测定电路的输入引脚
内部仿真器	FINED	输入 / 输出	FINE 接口引脚
	FINEC	输入	用于 FINE 接口的时钟引脚
地址总线	A0 ~ A23	输出	地址的输出引脚
数据总线	D0 ~ D15	输入 / 输出	双向数据总线
多路复用总线	A0/D0 ~ A15/D15	输入 / 输出	地址 / 数据多路复用总线
总线控制	RD#	输出	此选通信号表示正在读外部总线接口空间。
	WR#	输出	此选通信号表示在 1 次写选通模式中正在写外部总线接口空间。
	WR0#、WR1#	输出	此选通信号表示在字节选通模式中正在写外部总线接口空间并且数据总线 (D7 ~ D0、D15 ~ D8) 中任意一个有效。
	BC0#、BC1#	输出	此选通信号表示在 1 次写选通模式中正在存取外部总线接口空间并且数据总线 (D7 ~ D0、D15 ~ D8) 中任意一个有效。
	CS0# ~ CS3#	输出	区域 0 ~ 3 的选择信号
	WAIT#	输入	存取外部空间时的等待请求信号
	ALE	输出	选择地址 / 数据多路复用总线时的地址锁存信号
中断	NMI	输入	非屏蔽中断请求引脚
	IRQ0-IRQ7	输入	中断请求引脚

表 1.4 引脚功能一览表 (2/4)

分类	引脚名	输入 / 输出	功能
多功能定时器脉冲单元 2	MTIOC0A、MTIOC0B、MTIOC0C、MTIOC0D	输入 / 输出	TGRA0 ~ TGRD0 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	MTIOC1A、MTIOC1B	输入 / 输出	TGRA1 和 TGRB1 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	MTIOC2A、MTIOC2B	输入 / 输出	TGRA2 和 TGRB2 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	MTIOC3A、MTIOC3B、MTIOC3C、MTIOC3D	输入 / 输出	TGRA3 ~ TGRD3 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	MTIOC4A、MTIOC4B、MTIOC4C、MTIOC4D	输入 / 输出	TGRA4 ~ TGRD4 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	MTIC5U、MTIC5V、MTIC5W	输入	TGRU5、TGRV5 和 TGRW5 的输入捕捉的输入引脚 / 外部脉冲的输入引脚
	MTCLKA、MTCLKB、MTCLKC、MTCLKD	输入	外部时钟的输入引脚
端口输出允许 2	POE0# ~ POE3#、POE8#	输入	将 MTU2 的大电流引脚置为高阻抗状态的请求信号的输入引脚
8 位定时器	TMO0 ~ TMO3	输出	比较匹配的输入引脚
	TMCIO ~ TMCIO3	输入	输入到计数器的外部时钟的输入引脚
	TMRI0 ~ TMRI3	输入	计数器复位的输入引脚
实时时钟	RTCOUT	输出	1Hz 时钟的输出引脚
	RTCIC0 ~ RTCIC2	输入	抗干扰事件的输入引脚
串行通信接口 (SC1c)	• 异步模式 / 时钟同步模式		
	SCK0、SCK1、SCK5、SCK6、SCK8、SCK9	输入 / 输出	时钟的输入 / 输出引脚
	RXD0、RXD1、RXD5、RXD6、RXD8、RXD9	输入	接收数据的输入引脚
	TXD0、TXD1、TXD5、TXD6、TXD8、TXD9	输出	发送数据的输出引脚
	CTS0#、CTS1#、CTS5#、CTS6#、CTS8#、CTS9#	输入	用于控制开始发送 / 接收的输入引脚
	RTS0#、RTS1#、RTS5#、RTS6#、RTS8#、RTS9#	输出	用于控制开始发送 / 接收的输出引脚
	• 简易 I ² C 模式		
	SSCL0、SSCL1、SSCL5、SSCL6、SSCL8、SSCL9	输入 / 输出	I ² C 时钟的输入 / 输出引脚
	SSDA0、SSDA1、SSDA5、SSDA6、SSDA8、SSDA9	输入 / 输出	I ² C 数据的输入 / 输出引脚
	• 简易 SPI 模式		
	SCK0、SCK1、SCK5、SCK6、SCK8、SCK9	输入 / 输出	时钟的输入 / 输出引脚
	SMISO0、SMISO1、SMISO5、SMISO6、SMISO8、SMISO9	输入 / 输出	从属发送数据的输入 / 输出引脚
	SMOSI0、SMOSI1、SMOSI5、SMOSI6、SMOSI8、SMOSI9	输入 / 输出	主控发送数据的输入 / 输出引脚
	SS0# ~ SS11#	输入	片选的输入引脚

表 1.4 引脚功能一览表 (3/4)

分类	引脚名	输入 / 输出	功能
串行通信接口 (SCId)	• 异步模式 / 时钟同步模式		
	SCK12	输入 / 输出	时钟的输入 / 输出引脚
	RXD12	输入	接收数据的输入引脚
	TXD12	输出	发送数据的输出引脚
	CTS12#	输入	用于控制开始发送 / 接收的输入引脚
	RTS12#	输出	用于控制开始发送 / 接收的输出引脚
	• 简易 I ² C 模式		
	SSCL12	输入 / 输出	I ² C 时钟的输入 / 输出引脚
	SSDA12	输入 / 输出	I ² C 数据的输入 / 输出引脚
	• 简易 SPI 模式		
	SCK12	输入 / 输出	时钟的输入 / 输出引脚
	SMISO12	输入 / 输出	从属发送数据的输入 / 输出引脚
	SMOSI12	输入 / 输出	主控发送数据的输入 / 输出引脚
	SS12#	输入	片选的输入引脚
	• 扩展串行模式		
	RXDX12	输入	SCId 接收数据的输入引脚
TXDX12	输出	SCId 发送数据的输出引脚	
SIOX12	输入 / 输出	SCId 的接收 / 发送数据的输入 / 输出引脚	
I ² C 总线接口	SCL	输入 / 输出	I ² C 总线接口的时钟输入 / 输出引脚 能通过 N 沟道漏极开路输出，直接驱动总线。
	SDA	输入 / 输出	I ² C 总线接口的数据输入 / 输出引脚 能通过 N 沟道漏极开路输出，直接驱动总线。
串行外围接口	RSPCKA	输入 / 输出	RSPI 的时钟输入 / 输出引脚
	MOSIA	输入 / 输出	RSPI 的主控发送数据引脚
	MISOA	输入 / 输出	RSPI 的从属发送数据引脚
	SSLA0	输入 / 输出	RSPI 的从属片选输入 / 输出引脚
	SSLA1 ~ SSLA3	输出	RSPI 的从属片选输出引脚
12 位 A/D 转换器	AN000 ~ AN015	输入	A/D 转换器的模拟输入引脚
	ADTRG0#	输入	用于开始 A/D 转换的外部触发输入引脚
D/A 转换器	DA0、DA1	输出	D/A 转换器的模拟输出引脚
比较器 A	CMPA1	输入	用于比较器 A1 的模拟引脚
	CMPA2	输入	用于比较器 A2 的模拟引脚
	CVREFA	输入	用于比较器的基准电压引脚
比较器 B	CMPB0	输入	用于比较器 B0 的模拟引脚
	CVREFB0	输入	用于比较器 B0 的基准电压引脚
	CMPB1	输入	用于比较器 B1 的模拟引脚
	CVREFB1	输入	用于比较器 B1 的基准电压引脚

表 1.4 引脚功能一览表 (4/4)

分类	引脚名	输入 / 输出	功能
模拟电源	AVCC0	输入	12 位 A/D 转换器的模拟电源引脚 在不使用 12 位 A/D 转换器时, 必须连接 VCC。
	AVSS0	输入	12 位 A/D 转换器的模拟接地引脚 在不使用 12 位 A/D 转换器时, 必须连接 VSS。
	VREFH0	输入	12 位 A/D 转换器的基准电源引脚 在不使用 12 位 A/D 转换器时, 必须连接 VCC。
	VREF0L	输入	12 位 A/D 转换器的基准接地引脚 在不使用 12 位 A/D 转换器时, 必须连接 VSS。
	VREFH	输入	D/A 转换器的模拟电源引脚 在不使用 D/A 转换器时, 必须连接 VCC。
	VREFL	输入	D/A 转换器的模拟接地引脚 在不使用 D/A 转换器时, 必须连接 VSS。
I/O 端口	P03、P05、P07	输入 / 输出	3 位输入 / 输出引脚
	P12 ~ P17	输入 / 输出	6 位输入 / 输出引脚
	P20 ~ P27	输入 / 输出	8 位输入 / 输出引脚
	P30 ~ P37	输入 / 输出	8 位输入 / 输出引脚 (P35 为输入引脚)
	P40 ~ P47	输入 / 输出	8 位输入 / 输出引脚
	P50 ~ P55	输入 / 输出	6 位输入 / 输出引脚
	PA0 ~ PA7	输入 / 输出	8 位输入 / 输出引脚
	PB0 ~ PB7	输入 / 输出	8 位输入 / 输出引脚
	PC0 ~ PC7	输入 / 输出	8 位输入 / 输出引脚
	PD0 ~ PD7	输入 / 输出	8 位输入 / 输出引脚
	PE0 ~ PE7	输入 / 输出	8 位输入 / 输出引脚
	PH0 ~ PH3	输入 / 输出	4 位输入 / 输出引脚
	PJ1、PJ3	输入 / 输出	2 位输入 / 输出引脚

1.5 引脚排列图

引脚排列图如图 1.3 ~ 图 1.6 所示。各功能引脚一览表如表 1.5 ~ 表 1.8 所示。

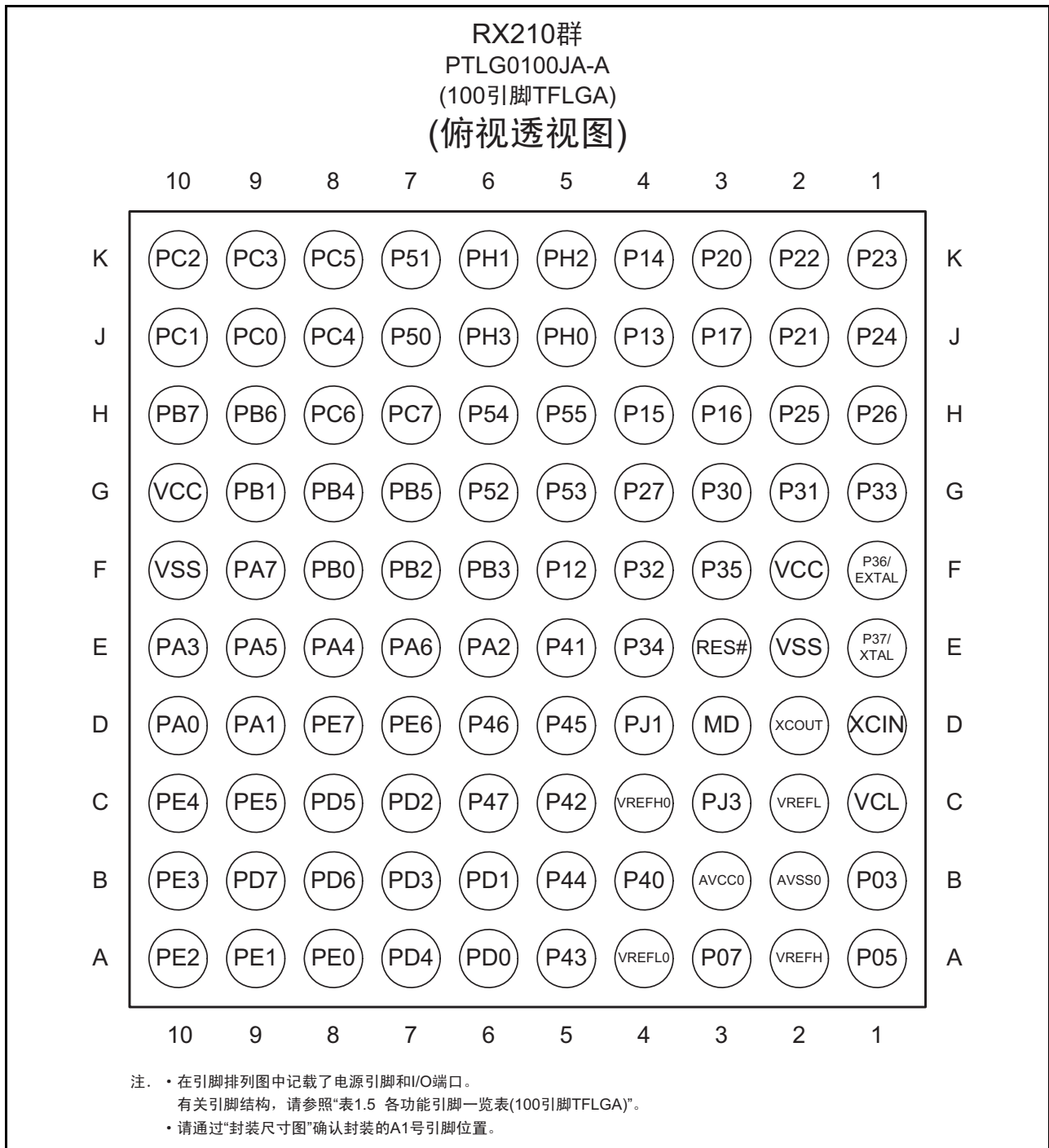


图 1.3 100 引脚 TFLGA 的引脚排列图 (俯视透视图)



图 1.4 100 引脚 LQFP 的引脚排列图

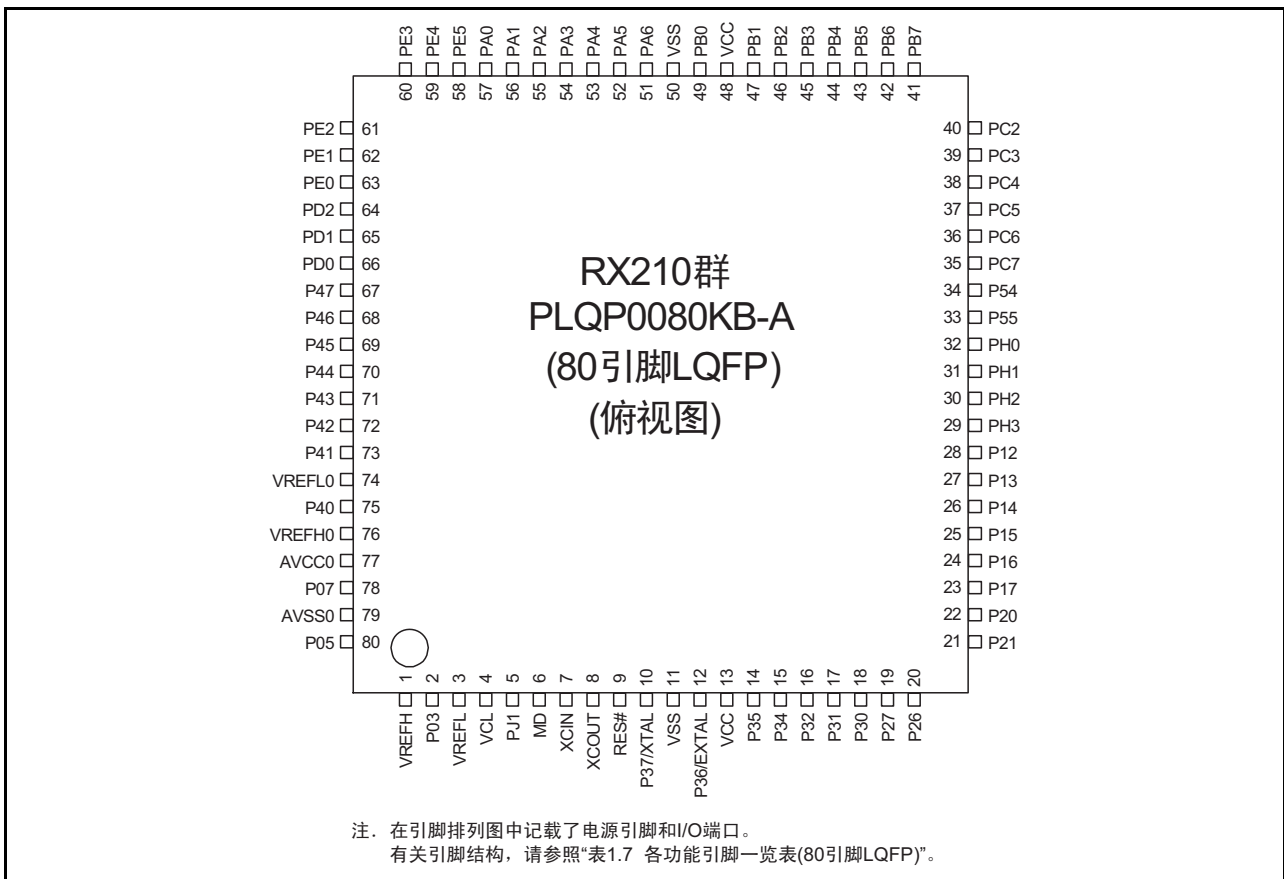


图 1.5 80 引脚 LQFP 的引脚排列图

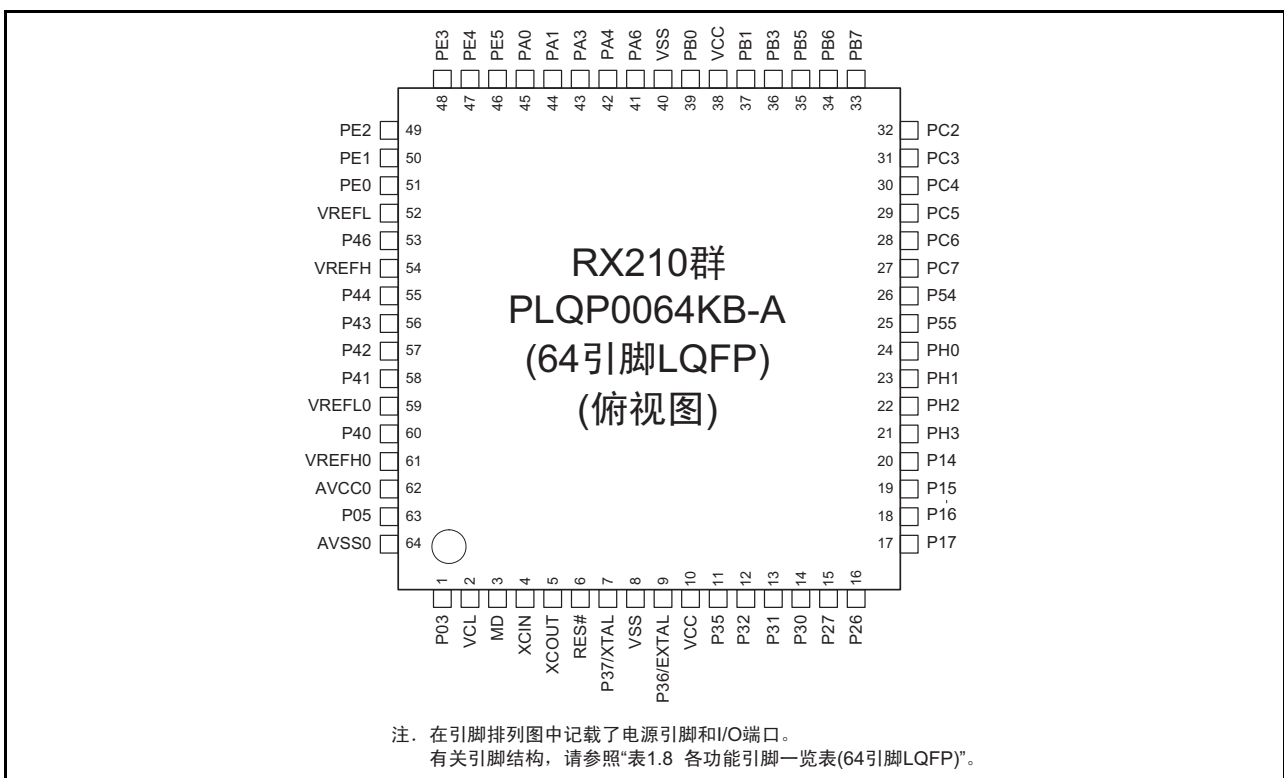


图 1.6 64 引脚 LQFP 的引脚排列图

表 1.5 各更能引脚一览表 (100 引脚 TFLGA) (1/3)

引脚序号	电源、时钟、系统控制	I/O 端口	外部总线	定时器 (MTU、TMR、POE)	通信 (SC1c、SC1d、RSPI、RIIC)	其他
A1		P05				DA1
A2	VREFH					
A3		P07				ADTRG0#
A4	VREFL0					
A5		P43				AN003
A6		PD0	D0[A0/D0]			IRQ0
A7		PD4	D4[A4/D4]	POE3#		IRQ4
A8		PE0	D8[A8/D8]		SCK12	AN008
A9		PE1	D9[A9/D9]	MTIOC4C	TXD12/TXDX12/ SIOX12/SMOSI12/ SSDA12	AN009/CMPB0
A10		PE2	D10[A10/D10]	MTIOC4A	RXD12/RXDX12/ SMISO12/SSCL12	IRQ7-DS/AN010/ CVREFB0
B1		P03				DA0
B2	AVSS0					
B3	AVCC0					
B4		P40				AN000
B5		P44				AN004
B6		PD1	D1[A1/D1]	MTIOC4B		IRQ1
B7		PD3	D3[A3/D3]	POE8#		IRQ3
B8		PD6	D6[A6/D6]	MTIC5V/POE1#		IRQ6
B9		PD7	D7[A7/D7]	MTIC5U/POE0#		IRQ7
B10		PE3	D11[A11/D11]	MTIOC4B/POE8#	CTS12#/RTS12#/ SS12#	AN011/CMPA1
C1	VCL					
C2	VREFL					
C3		PJ3		MTIOC3C	CTS6#/RTS6#/SS6#	
C4	VREFH0					
C5		P42				AN002
C6		P47				AN007
C7		PD2	D2[A2/D2]	MTIOC4D		IRQ2
C8		PD5	D5[A5/D5]	MTIC5W/POE2#		IRQ5
C9		PE5	D13[A13/D13]	MTIOC4C/MTIOC2B		IRQ5/AN013
C10		PE4	D12[A12/D12]	MTIOC4D/MTIOC1A		AN012/CMPA2
D1	XCIN					
D2	XCOUT					
D3	MD					FINED
D4		PJ1		MTIOC3A		
D5		P45				AN005
D6		P46				AN006
D7		PE6	D14[A14/D14]			IRQ6/AN014
D8		PE7	D15[A15/D15]			IRQ7/AN015
D9		PA1	A1	MTIOC0B/MTCLKC	SCK5/SSLA2	CVREFA
D10		PA0	A0/BC0#	MTIOC4A	SSLA1	CACREF
E1	XTAL	P37				

表 1.5 各更能引脚一览表 (100 引脚 TFLGA) (2/3)

引脚序号	电源、时钟、系统控制	I/O 端口	外部总线	定时器 (MTU、TMR、POE)	通信 (SCLc、SCId、RSPI、RIIC)	其他
E2	VSS					
E3	RES#					
E4		P34		MTIOC0A/TMCI3/ POE2#	SCK6	IRQ4
E5		P41				AN001
E6		PA2	A2		RXD5/SMISO5/ SSCL5/SSLA3	
E7		PA6	A6	MTIC5V/MTCLKB/ TMCI3/POE2#	CTS5#/RTS5#/SS5#/ MOSIA	
E8		PA4	A4	MTIC5U/MTCLKA/ TMRI0	TXD5/SMOSI5/ SSDA5/SSLA0	IRQ5-DS/CVREFB1
E9		PA5	A5		RSPCKA	
E10		PA3	A3	MTIOC0D/MTCLKD	RXD5/SMISO5/ SSCL5	IRQ6-DS/CMPB1
F1	EXTAL	P36				
F2	VCC					
F3		P35				NMI
F4		P32		MTIOC0C/TMO3	TXD6/SMOSI6/ SSDA6	IRQ2-DS/RTCOUT/ RTCIC2
F5		P12		TMCI1	SCL	IRQ2
F6		PB3	A11	MTIOC0A/MTIOC4A/ TMO0/POE3#	SCK6	
F7		PB2	A10		CTS6#/RTS6#/SS6#	
F8		PB0	A8	MTIC5W	RXD6/SMISO6/ SSCL6/RSPCKA	
F9		PA7	A7		MISOA	
F10	VSS					
G1		P33		MTIOC0D/TMRI3/ POE3#	RXD6/SMISO6/ SSCL6	IRQ3-DS
G2		P31		MTIOC4D/TMCI2	CTS1#/RTS1#/SS1#	IRQ1-DS/RTCIC1
G3		P30		MTIOC4B/TMRI3/ POE8#	RXD1/SMISO1/ SSCL1	IRQ0-DS/RTCIC0
G4		P27	CS3#	MTIOC2B/TMCI3	SCK1	FINEC
G5	BCLK	P53				
G6		P52	RD#			
G7		PB5	A13	MTIOC2A/MTIOC1B/ TMRI1/POE1#	SCK9	
G8		PB4	A12		CTS9#/RTS9#/SS9#	
G9		PB1	A9	MTIOC0C/MTIOC4C/ TMCI0	TXD6/SMOSI6/ SSDA6	IRQ4-DS
G10	VCC					
H1		P26	CS2#	MTIOC2A/TMO1	TXD1/SMOSI1/ SSDA1	
H2		P25	CS1#	MTIOC4C/MTCLKB		ADTRG0#
H3		P16		MTIOC3C/MTIOC3D/ TMO2	TXD1/SMOSI1/ SSDA1/MOSIA/SCL- DS	IRQ6/RTCOUT/ ADTRG0#
H4		P15		MTIOC0B/MTCLKB/ TMCI2	RXD1/SMISO1/ SSCL1	IRQ5

表 1.5 各更能引脚一览表 (100 引脚 TFLGA) (3/3)

引脚序号	电源、时钟、 系统控制	I/O 端口	外部总线	定时器 (MTU、TMR、POE)	通信 (SC1c、SC1d、 RSPI、RIIC)	其他
H5		P55	WAIT#	MTIOC4D/TMO3		
H6		P54	ALE	MTIOC4B/TMC11		
H7		PC7	A23/CS0#	MTIOC3A/TMO2/ MTCLKB	TXD8/SMOSI8/ SSDA8/MISOA	CACREF
H8		PC6	A22/CS1#	MTIOC3C/MTCLKA/ TMC12	RXD8/SMISO8/ SSCL8/MOSIA	
H9		PB6	A14	MTIOC3D	RXD9/SMISO9/ SSCL9	
H10		PB7	A15	MTIOC3B	TXD9/SMOSI9/ SSDA9	
J1		P24	CS0#	MTIOC4A/MTCLKA/ TMRI1		
J2		P21		MTIOC1B/TMC10	RXD0/SMISO0/ SSCL0	
J3		P17		MTIOC3A/MTIOC3B/ TMO1/POE8#	SCK1/MISOA/ SDA-DS	IRQ7
J4		P13		MTIOC0B/TMO3	SDA	IRQ3
J5		PH0				CACREF
J6		PH3		TMC10		
J7		P50	WR0#/WR#			
J8		PC4	A20/CS3#	MTIOC3D/MTCLKC/ TMC11/POE0#	SCK5/CTS8#/ RTS8#/SS8#/SSLA0	
J9		PC0	A16	MTIOC3C	CTS5#/RTS5#/SS5#/ SSLA1	
J10		PC1	A17	MTIOC3A	SCK5/SSLA2	
K1		P23		MTIOC3D/MTCLKD	CTS0#/RTS0#/SS0#	
K2		P22		MTIOC3B/MTCLKC/ TMO0	SCK0	
K3		P20		MTIOC1A/TMRI0	TXD0/SMOSI0/ SSDA0	
K4		P14		MTIOC3A/MTCLKA/ TMRI2	CTS1#/RTS1#/SS1#	IRQ4
K5		PH2		TMRI0		IRQ1
K6		PH1		TMO0		IRQ0
K7		P51	WR1#/BC1#/ WAIT#			
K8		PC5	A21/CS2#/WAIT#	MTIOC3B/MTCLKD/ TMRI2	SCK8/RSPCKA	
K9		PC3	A19	MTIOC4D	TXD5/SMOSI5/ SSDA5	
K10		PC2	A18	MTIOC4B	RXD5/SMISO5/ SSCL5/SSLA3	

注. 引脚名中附加“-DS”的引脚能用作深度软件待机模式的解除引脚。

表 1.6 各引脚更能一览表 (100 引脚 LQFP) (1/3)

引脚序号	电源、时钟 系统控制	I/O 端口	外部总线	定时器 (MTU、TMR、POE)	通信 (SC1c、SC1d、 RSPI、RIIC)	其他
1	VREFH					
2		P03				DA0
3	VREFL					
4		PJ3		MTIOC3C	CTS6#/RTS6#/SS6#	
5	VCL					
6		PJ1		MTIOC3A		
7	MD					FINED
8	XCIN					
9	XCOUT					
10	RES#					
11	XTAL	P37				
12	VSS					
13	EXTAL	P36				
14	VCC					
15		P35				NMI
16		P34		MTIOC0A/TMCI3/POE 2#	SCK6	IRQ4
17		P33		MTIOC0D/TMRI3/POE 3#	RXD6/SMISO6/ SSCL6	IRQ3-DS
18		P32		MTIOC0C/TMO3	TXD6/SMOSI6/ SSDA6	IRQ2-DS/RTCOUT/ RTCIC2
19		P31		MTIOC4D/TMCI2	CTS1#/RTS1#/SS1#	IRQ1-DS/RTCIC1
20		P30		MTIOC4B/TMRI3/POE 8#	RXD1/SMISO1/ SSCL1	IRQ0-DS/RTCIC0
21		P27	CS3#	MTIOC2B/TMCI3	SCK1	FINEC
22		P26	CS2#	MTIOC2A/TMO1	TXD1/SMOSI1/ SSDA1	
23		P25	CS1#	MTIOC4C/MTCLKB		ADTRG0#
24		P24	CS0#	MTIOC4A/MTCLKA/ TMRI1		
25		P23		MTIOC3D/MTCLKD	CTS0#/RTS0#/SS0#	
26		P22		MTIOC3B/MTCLKC/ TMO0	SCK0	
27		P21		MTIOC1B/TMCI0	RXD0/SMISO0/ SSCL0	
28		P20		MTIOC1A/TMRI0	TXD0/SMOSI0/ SSDA0	
29		P17		MTIOC3A/MTIOC3B/ TMO1/POE8#	SCK1/MISOA/ SDA-DS	IRQ7
30		P16		MTIOC3C/MTIOC3D/ TMO2	TXD1/SMOSI1/ SSDA1/MOSIA/SCL- DS	IRQ6/RTCOUT/ ADTRG0#
31		P15		MTIOC0B/MTCLKB/ TMCI2	RXD1/SMISO1/ SSCL1	IRQ5
32		P14		MTIOC3A/MTCLKA/ TMRI2	CTS1#/RTS1#/SS1#	IRQ4
33		P13		MTIOC0B/TMO3	SDA	IRQ3
34		P12		TMCI1	SCL	IRQ2
35		PH3		TMCI0		

表 1.6 各引脚更能一览表 (100 引脚 LQFP) (2/3)

引脚序号	电源、时钟 系统控制	I/O 端口	外部总线	定时器 (MTU、TMR、POE)	通信 (SC1c、SC1d、 RSPI、RIIC)	其他
36		PH2		TMRI0		IRQ1
37		PH1		TMO0		IRQ0
38		PH0				CACREF
39		P55	WAIT#	MTIOC4D/TMO3		
40		P54	ALE	MTIOC4B/TMCI1		
41	BCLK	P53				
42		P52	RD#			
43		P51	WR1#/BC1#/ WAIT#			
44		P50	WR0#/WR#			
45		PC7	A23/CS0#	MTIOC3A/TMO2/ MTCLKB	TXD8/SMOSI8/ SSDA8/MISOA	CACREF
46		PC6	A22/CS1#	MTIOC3C/MTCLKA/ TMCI2	RXD8/SMISO8/ SSCL8/MOSIA	
47		PC5	A21/CS2#/WAIT#	MTIOC3B/MTCLKD/ TMRI2	SCK8/RSPCKA	
48		PC4	A20/CS3#	MTIOC3D/MTCLKC/ TMCI1/POE0#	SCK5/CTS8#/ RTS8#/SS8#/SSLA0	
49		PC3	A19	MTIOC4D	TXD5/SMOSI5/ SSDA5	
50		PC2	A18	MTIOC4B	RXD5/SMISO5/ SSCL5/SSLA3	
51		PC1	A17	MTIOC3A	SCK5/SSLA2	
52		PC0	A16	MTIOC3C	CTS5#/RTS5#/SS5#/ SSLA1	
53		PB7	A15	MTIOC3B	TXD9/SMOSI9/ SSDA9	
54		PB6	A14	MTIOC3D	RXD9/SMISO9/ SSCL9	
55		PB5	A13	MTIOC2A/MTIOC1B/ TMRI1/POE1#	SCK9	
56		PB4	A12		CTS9#/RTS9#/SS9#	
57		PB3	A11	MTIOC0A/MTIOC4A/ TMO0/POE3#	SCK6	
58		PB2	A10		CTS6#/RTS6#/SS6#	
59		PB1	A9	MTIOC0C/MTIOC4C/ TMCI0	TXD6/SMOSI6/ SSDA6	IRQ4-DS
60	VCC					
61		PB0	A8	MTIC5W	RXD6/SMISO6/ SSCL6/RSPCKA	
62	VSS					
63		PA7	A7		MISOA	
64		PA6	A6	MTIC5V/MTCLKB/ TMCI3/POE2#	CTS5#/RTS5#/SS5#/ MOSIA	
65		PA5	A5		RSPCKA	
66		PA4	A4	MTIC5U/MTCLKA/ TMRI0	TXD5/SMOSI5/ SSDA5/SSLA0	IRQ5-DS/CVREFB1
67		PA3	A3	MTIOC0D/MTCLKD	RXD5/SMISO5/ SSCL5	IRQ6-DS/CMPB1

表 1.6 各引脚更能一览表 (100 引脚 LQFP) (3/3)

引脚序号	电源、时钟 系统控制	I/O 端口	外部总线	定时器 (MTU、TMR、POE)	通信 (SC1c、SC1d、 RSPI、RIIC)	其他
68		PA2	A2		RXD5/SMISO5/ SSCL5/SSLA3	
69		PA1	A1	MTIOC0B/MTCLKC	SCK5/SSLA2	CVREFA
70		PA0	A0/BC0#	MTIOC4A	SSLA1	CACREF
71		PE7	D15[A15/D15]			IRQ7/AN015
72		PE6	D14[A14/D14]			IRQ6/AN014
73		PE5	D13[A13/D13]	MTIOC4C/MTIOC2B		IRQ5/AN013
74		PE4	D12[A12/D12]	MTIOC4D/MTIOC1A		AN012/CMPA2
75		PE3	D11[A11/D11]	MTIOC4B/POE8#	CTS12#/RTS12#/ SS12#	AN011/CMPA1
76		PE2	D10[A10/D10]	MTIOC4A	RXD12/RDX12/ SMISO12/SSCL12	IRQ7-DS/AN010/ CVREFB0
77		PE1	D9[A9/D9]	MTIOC4C	TXD12/TXD12/ SIOX12/SMOS12/ SSDA12	AN009/CMPB0
78		PE0	D8[A8/D8]		SCK12	AN008
79		PD7	D7[A7/D7]	MTIC5U/POE0#		IRQ7
80		PD6	D6[A6/D6]	MTIC5V/POE1#		IRQ6
81		PD5	D5[A5/D5]	MTIC5W/POE2#		IRQ5
82		PD4	D4[A4/D4]	POE3#		IRQ4
83		PD3	D3[A3/D3]	POE8#		IRQ3
84		PD2	D2[A2/D2]	MTIOC4D		IRQ2
85		PD1	D1[A1/D1]	MTIOC4B		IRQ1
86		PD0	D0[A0/D0]			IRQ0
87		P47				AN007
88		P46				AN006
89		P45				AN005
90		P44				AN004
91		P43				AN003
92		P42				AN002
93		P41				AN001
94	VREFL0					
95		P40				AN000
96	VREFH0					
97	AVCC0					
98		P07				ADTRG0#
99	AVSS0					
100		P05				DA1

注. 引脚名中附加“-DS”的引脚能用作深度软件待机模式的解除引脚。

表 1.7 各功能引脚一览表 (80 引脚 LQFP) (1/2)

引脚序号	电源、时钟、 系统控制	I/O 端口	定时器 (MTU、TMR、POE)	通信 (SClC、SClD、RSPI、RIIC)	其他
1	VREFH				
2		P03			DA0
3	VREFL				
4	VCL				
5		PJ1	MTIOC3A		
6	MD				FINED
7	XCIN				
8	XCOUT				
9	RES#				
10	XTAL	P37			
11	VSS				
12	EXTAL	P36			
13	VCC				
14		P35			NMI
15		P34	MTIOC0A/TMCI3/POE2#	SCK6	IRQ4
16		P32	MTIOC0C/TMO3	TXD6/SMOSI6/SSDA6	IRQ2-DS/RTCOUT/ RTCIC2
17		P31	MTIOC4D/TMCI2	CTS1#/RTS1#/SS1#	IRQ1-DS/RTCIC1
18		P30	MTIOC4B/TMRI3/POE8#	RXD1/SMISO1/SSCL1	IRQ0-DS/RTCIC0
19		P27	MTIOC2B/TMCI3	SCK1	FINEC
20		P26	MTIOC2A/TMO1	TXD1/SMOSI1/SSDA1	
21		P21	MTIOC1B/TMCI0	RXD0/SMISO0/SSCL0	
22		P20	MTIOC1A/TMRI0	TXD0/SMOSI0/SSDA0	
23		P17	MTIOC3A/MTIOC3B/TMO1/ POE8#	SCK1/MISOA/ SDA-DS	IRQ7
24		P16	MTIOC3C/MTIOC3D/TMO2	TXD1/SMOSI1/SSDA1/MOSIA/ SCL-DS	IRQ6/RTCOUT/ ADTRG0#
25		P15	MTIOC0B/MTCLKB/TMCI2	RXD1/SMISO1/SSCL1	IRQ5
26		P14	MTIOC3A/MTCLKA/TMRI2	CTS1#/RTS1#/SS1#	IRQ4
27		P13	MTIOC0B/TMO3	SDA	IRQ3
28		P12	TMCI1	SCL	IRQ2
29		PH3	TMCI0		
30		PH2	TMRI0		IRQ1
31		PH1	TMO0		IRQ0
32		PH0			CACREF
33		P55	MTIOC4D/TMO3		
34		P54	MTIOC4B/TMCI1		
35		PC7	MTIOC3A/TMO2/MTCLKB	TXD8/SMOSI8/SSDA8/MISOA	CACREF
36		PC6	MTIOC3C/MTCLKA/TMCI2	RXD8/SMISO8/SSCL8/MOSIA	
37		PC5	MTIOC3B/MTCLKD/TMRI2	SCK8/RSPCKA	
38		PC4	MTIOC3D/MTCLKC/TMCI1/ POE0#	SCK5/CTS8#/RTS8#/SS8#/ SSLA0	
39		PC3	MTIOC4D	TXD5/SMOSI5/SSDA5	
40		PC2	MTIOC4B	RXD5/SMISO5/SSCL5/SSLA3	
41		PB7	MTIOC3B	TXD9/SMOSI9/SSDA9	
42		PB6	MTIOC3D	RXD9/SMISO9/SSCL9	

表 1.7 各功能引脚一览表 (80 引脚 LQFP) (2/2)

引脚序号	电源、时钟、系统控制	I/O 端口	定时器 (MTU、TMR、POE)	通信 (SC1c、SC1d、RSPI、RIIC)	其他
43		PB5	MTIOC2A/MTIOC1B/TMR1/ POE1#	SCK9	
44		PB4		CTS9#/RTS9#/SS9#	
45		PB3	MTIOC0A/MTIOC4A/TMO0/ POE3#	SCK6	
46		PB2		CTS6#/RTS6#/SS6#	
47		PB1	MTIOC0C/MTIOC4C/TMC10	TXD6/SMOSI6/SSDA6	IRQ4-DS
48	VCC				
49		PB0	MTIC5W	RXD6/SMISO6/SSCL6/ RSPCKA	
50	VSS				
51		PA6	MTIC5V/MTCLKB/TMC13/ POE2#	CTS5#/RTS5#/SS5#/MOSIA	
52		PA5		RSPCKA	
53		PA4	MTIC5U/MTCLKA/TMR10	TXD5/SMOSI5/SSDA5/SSLA0	IRQ5-DS/CVREFB1
54		PA3	MTIOC0D/MTCLKD	RXD5/SMISO5/SSCL5	IRQ6-DS/CMPB1
55		PA2		RXD5/SMISO5/SSCL5/SSLA3	
56		PA1	MTIOC0B/MTCLKC	SCK5/SSLA2	CVREFA
57		PA0	MTIOC4A	SSLA1	CACREF
58		PE5	MTIOC4C/MTIOC2B		IRQ5/AN013
59		PE4	MTIOC4D/MTIOC1A		AN012/CMPA2
60		PE3	MTIOC4B/POE8#	CTS12#/RTS12#/SS12#	AN011/CMPA1
61		PE2	MTIOC4A	RXD12/RXD12/SMISO12/ SSCL12	IRQ7-DS/AN010/ CVREFB0
62		PE1	MTIOC4C	TXD12/TXD12/SIOX12/ SMOSI12/SSDA12	AN009/CMPB0
63		PE0		SCK12	AN008
64		PD2	MTIOC4D		IRQ2
65		PD1	MTIOC4B		IRQ1
66		PD0			IRQ0
67		P47			AN007
68		P46			AN006
69		P45			AN005
70		P44			AN004
71		P43			AN003
72		P42			AN002
73		P41			AN001
74	VREFL0				
75		P40			AN000
76	VREFH0				
77	AVCC0				
78		P07			ADTRG0#
79	AVSS0				
80		P05			DA1

注：引脚名中附加“-DS”的引脚能用作深度软件待机模式的解除引脚。

表 1.8 各功能引脚一览表 (64 引脚 LQFP) (1/2)

引脚序号	电源、时钟、 系统控制	I/O 端口	定时器 (MTU、TMR、POE)	通信 (SC1c、SC1d、RSPI、RIIC)	其他
1		P03			DA0
2	VCL				
3	MD				FINED
4	XCIN				
5	XCOUT				
6	RES#				
7	XTAL	P37			
8	VSS				
9	EXTAL	P36			
10	VCC				
11		P35			NMI
12		P32	MTIOC0C/TMO3	TXD6/SMOSI6/SSDA6	IRQ2-DS/RTCCOUT/ RTCIC2
13		P31	MTIOC4D/TMCI2	CTS1#/RTS1#/SS1#	IRQ1-DS/RTCIC1
14		P30	MTIOC4B/TMRI3/POE8#	RXD1/SMISO1/SSCL1	IRQ0-DS/RTCIC0
15		P27	MTIOC2B/TMCI3	SCK1	FINEC
16		P26	MTIOC2A/TMO1	TXD1/SMOSI1/SSDA1	
17		P17	MTIOC3A/MTIOC3B/TMO1/ POE8#	SCK1/MISOA/SDA-DS	IRQ7
18		P16	MTIOC3C/MTIOC3D/TMO2	TXD1/SMOSI1/SSDA1/ MOSIA/SCL-DS	IRQ6/RTCCOUT/ADTRG0#
19		P15	MTIOC0B/MTCLKB/TMCI2	RXD1/SMISO1/SSCL1	IRQ5
20		P14	MTIOC3A/MTCLKA/TMRI2	CTS1#/RTS1#/SS1#	IRQ4
21		PH3	TMCI0		
22		PH2	TMRI0		IRQ1
23		PH1	TMO0		IRQ0
24		PH0			CACREF
25		P55	MTIOC4D/TMO3		
26		P54	MTIOC4B/TMCI1		
27		PC7	MTIOC3A/TMO2/MTCLKB	TXD8/SMOSI8/SSDA8/ MISOA	CACREF
28		PC6	MTIOC3C/MTCLKA/TMCI2	RXD8/SMISO8/SSCL8/ MOSIA	
29		PC5	MTIOC3B/MTCLKD/TMRI2	SCK8/RSPCKA	
30		PC4	MTIOC3D/MTCLKC/TMCI1/ POE0#	SCK5/CTS8#/RTS8#/SS8#/ SSLA0	
31		PC3	MTIOC4D	TXD5/SMOSI5/SSDA5	
32		PC2	MTIOC4B	RXD5/SMISO5/SSCL5/ SSLA3	
33		PB7	MTIOC3B	TXD9/SMOSI9/SSDA9	
34		PB6	MTIOC3D	RXD9/SMISO9/SSCL9	
35		PB5	MTIOC2A/MTIOC1B/TMRI1/ POE1#	SCK9	
36		PB3	MTIOC0A/MTIOC4A/TMO0/ POE3#	SCK6	
37		PB1	MTIOC0C/MTIOC4C/TMCI0	TXD6/SMOSI6/SSDA6	IRQ4-DS
38	VCC				

表 1.8 各功能引脚一览表 (64 引脚 LQFP) (2/2)

引脚序号	电源、时钟、 系统控制	I/O 端口	定时器 (MTU、TMR、POE)	通信 (SC1c、SC1d、RSPI、RIIC)	其他
39		PB0	MTIC5W	RXD6/SMISO6/SSCL6/ RSPCKA	
40	VSS				
41		PA6	MTIC5V/MTCLKB/TMCI3/ POE2#	CTS5#/RTS5#/SS5#/MOSIA	
42		PA4	MTIC5U/MTCLKA/TMRI0	TXD5/SMOSI5/SSDA5/ SSLA0	IRQ5-DS/CVREFB1
43		PA3	MTIOC0D/MTCLKD	RXD5/SMISO5/SSCL5	IRQ6-DS/CMPB1
44		PA1	MTIOC0B/MTCLKC	SCK5/SSLA2	CVREFA
45		PA0	MTIOC4A	SSLA1	CACREF
46		PE5	MTIOC4C/MTIOC2B		IRQ5/AN013
47		PE4	MTIOC4D/MTIOC1A		AN012/CMPA2
48		PE3	MTIOC4B/POE8#	CTS12#/RTS12#/SS12#	AN011/CMPA1
49		PE2	MTIOC4A	RXD12/RDX12/SMISO12/ SSCL12	IRQ7-DS/AN010/ CVREFB0
50		PE1	MTIOC4C	TXD12/TXD12/SIOX12/ SMOSI12/SSDA12	AN009/CMPB0
51		PE0		SCK12	AN008
52	VREFL				
53		P46			AN006
54	VREFH				
55		P44			AN004
56		P43			AN003
57		P42			AN002
58		P41			AN001
59	VREFL0				
60		P40			AN000
61	VREFH0				
62	AVCC0				
63		P05			DA1
64	AVSS0				

注. 引脚名中附加“-DS”的引脚能用作深度软件待机模式的解除引脚。

2. CPU

RX210 群是装载了 RX CPU 的处理器。

RX CPU 采用了可变长指令格式。通过将常用指令分配为较短的指令长度，能开发出存储容量小而且效率高的程序。

RX CPU 有 73 种基本指令、9 种 DSP 功能指令共 82 种指令和 10 种寻址方式，进行寄存器 - 寄存器之间、寄存器 - 存储器之间、立即数 - 寄存器、立即数 - 存储器的运算以及位操作、存储器 - 存储器之间的传送。通过 1 个时钟执行寄存器之间的运算指令和多条复合指令，实现了快速运算处理。因为内置了乘法器和除法器，所以能进行快速的乘除法运算处理。

RX CPU 通过取指令、解码、执行、存储器的存取、回写这 5 个阶段的流水线处理，进行指令的处理。如果因存储器的存取引起流水线的延长，就可能先执行后面的运算。RX CPU 采用“out-of-order-completion”进行节省时钟周期数的指令执行控制。

2.1 特点

- 指令的最短执行时间：1 个时钟执行 1 条指令
- 地址空间：4G 字节、线性地址
- CPU 寄存器组
 - 通用寄存器：32 位×16 个
 - 控制寄存器：32 位×8 个
 - 累加器：64 位×1 个
- 基本指令：73 种（算术/逻辑指令、传送指令、转移指令、位操作指令、字符串操作指令、系统操作指令）
 - 对应转移距离的相对转移指令
 - 可变长指令格式（1 字节～8 字节）
 - 常用指令有短格式
- DSP 功能指令：9 种
 - 对应 16 位×16 位的乘法指令和乘加指令
 - 对应累加器的舍入指令
- 寻址方式：10 种
- 5 个阶段的流水线
 - 采用“out-of-order-completion”
- 处理器模式
 - 管理模式、用户模式
- 数据排列
 - 可选择小端法或者大端法

2.2 CPU 寄存器组

RX CPU 寄存器有 16 个通用寄存器、8 个控制寄存器和 1 个由 DSP 功能指令使用的累加器。

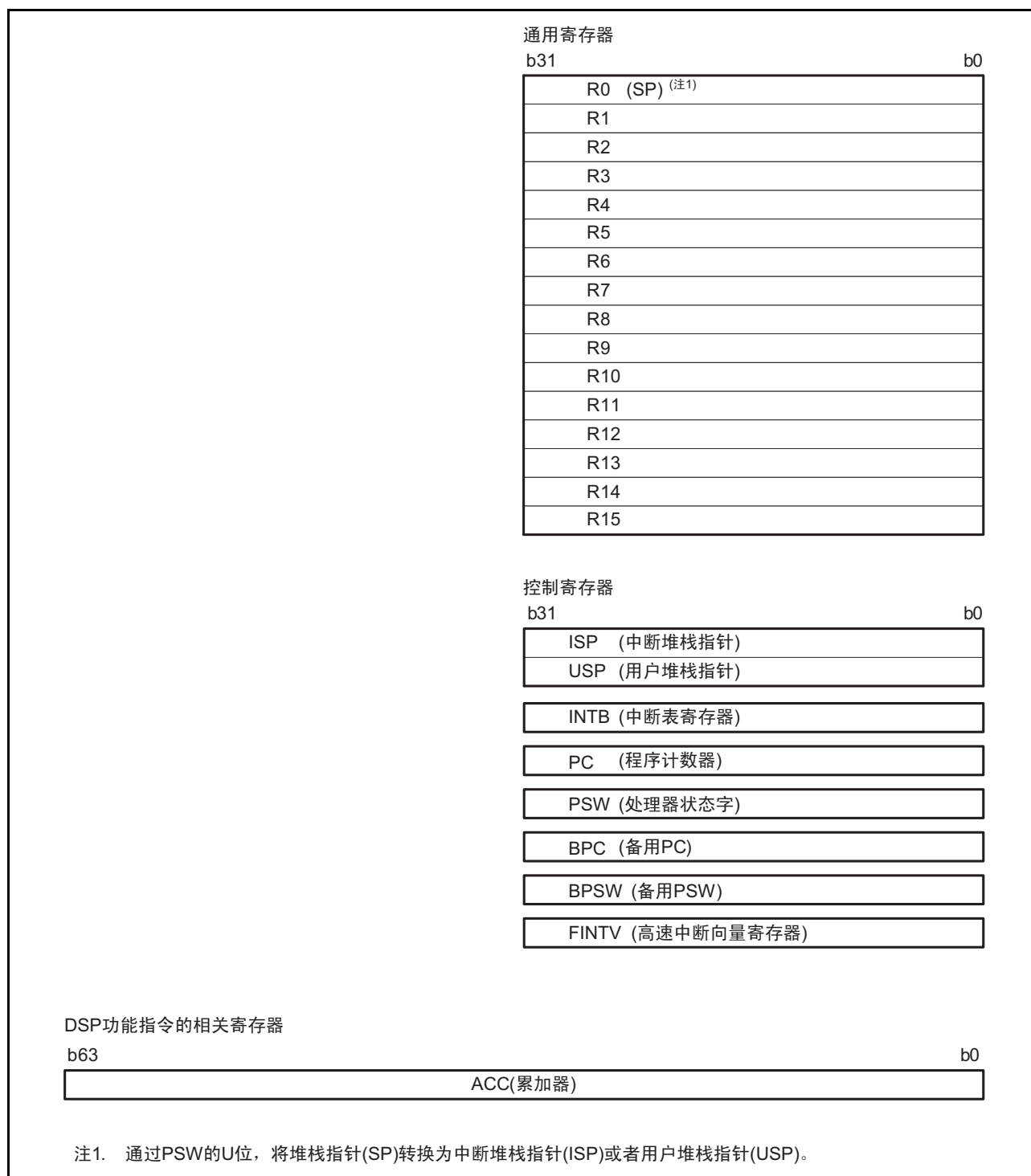


图 2.1 CPU 寄存器组

2.2.1 通用寄存器 (R0 ~ R15)

通用寄存器有 16 个 (R0 ~ R15)，用作数据寄存器和地址寄存器。

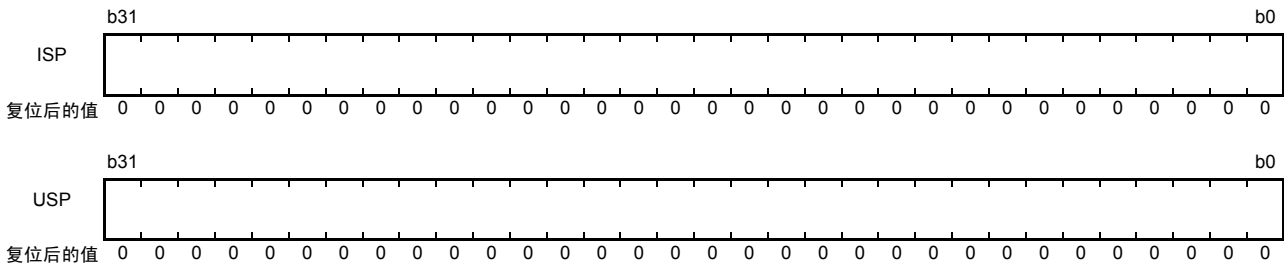
通用寄存器 R0 除了具有通用寄存器的功能以外，还有堆栈指针 (SP) 的功能。通过处理器状态字 (PSW) 的堆栈指针指定位 (U)，将 SP 转换为中断堆栈指针 (ISP) 或者用户堆栈指针 (USP)。

2.2.2 控制寄存器

控制寄存器有以下 8 个：

- 中断堆栈指针 (ISP)
- 用户堆栈指针 (USP)
- 中断表寄存器 (INTB)
- 程序计数器 (PC)
- 处理器状态字 (PSW)
- 备用 PC (BPC)
- 备用 PSW (BPSW)
- 高速中断向量寄存器 (FINTV)

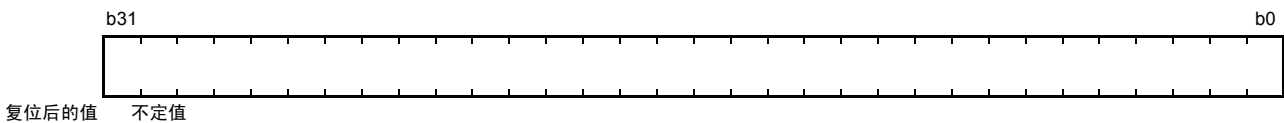
2.2.2.1 中断堆栈指针 (ISP) / 用户堆栈指针 (USP)



堆栈指针 (SP) 有中断堆栈指针 (ISP) 和用户堆栈指针 (USP) 两种，通过处理器状态字 (PSW) 的堆栈指针指定位 (U) 转换要使用的堆栈指针 (ISP/USP)。

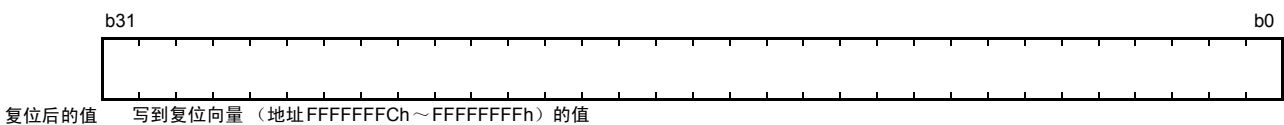
如果给 ISP 和 USP 设定 4 的倍数，带有堆栈操作的指令和中断响应顺序的周期数就会变短。

2.2.2.2 中断表寄存器 (INTB)



必须给中断表寄存器 (INTB) 设定可定向量表的起始地址。

2.2.2.3 程序计数器 (PC)



程序计数器 (PC) 表示正在执行的指令的地址。

2.2.2.4 处理器状态字 (PSW)

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
—	—	—	—	IPL[3:0]				—	—	—	PM	—	—	U	I
复位后的值	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	—	—	—	—	—	O	S	Z	C
复位后的值	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	C	进位标志	0: 未发生进位 1: 发生进位	R/W
b1	Z	零标志	0: 运算结果不为“0” 1: 运算结果为“0”	R/W
b2	S	符号标志	0: 运算结果为正数或者“0” 1: 运算结果为负数	R/W
b3	O	上溢标志	0: 未发生上溢 1: 发生上溢	R/W
b15-b4	—	保留位	读写值都为“0”。	R/W
b16	I (注1)	中断允许位	0: 禁止中断 1: 允许中断	R/W
b17	U (注1)	堆栈指针指定位	0: 指定中断堆栈指针 (ISP) 1: 指定用户堆栈指针 (USP)	R/W
b19-b18	—	保留位	读写值都为“0”。	R/W
b20	PM (注1、注2、注3)	处理器模式设定位	0: 设定为管理模式 1: 设定为用户模式	R/W
b23-b21	—	保留位	读写值都为“0”。	R/W
b27-b24	IPL[3:0] (注1)	处理器中断优先级	b27 b24 0000: 0级 (最低) 0001: 1级 0010: 2级 0011: 3级 0100: 4级 0101: 5级 0110: 6级 0111: 7级 1000: 8级 1001: 9级 1010: 10级 1011: 11级 1100: 12级 1101: 13级 1110: 14级 1111: 15级 (最高)	R/W
b31-b28	—	保留位	读写值都为“0”。	R/W

注 1. 当设定为用户模式时, 忽视用 MVTC 指令和 POPC 指令对 IPL[3:0] 位、PM 位、U 位和 I 位的写操作。另外, 当用 MVTIPL 指令写 IPL[3:0] 位时, 发生特权指令异常。

注 2. 当设定为管理模式时, 忽视用 MVTC 指令和 POPC 指令对 PM 位的写操作, 但是能写除 PM 位以外的其他位。

注 3. 要从管理模式转换为用户模式时, 必须在将被压栈的 PSW.PM 位置“1”后执行 RTE 指令、或者在将 BPSW.PM 位置“1”后执行 RTFI 指令。

处理器状态字（PSW）表示指令的执行结果和 CPU 的状态。

C 标志（进位标志）

表示运算结果发生进位、借位或者移出。

Z 标志（零标志）

表示运算结果为“0”。

S 标志（符号标志）

表示运算结果为负数。

O 标志（上溢标志）

表示运算中发生上溢。

I 位（中断允许位）

此位是允许接受中断请求的位。如果接受异常处理，此位就变为“0”。

U 位（堆栈指针指定位）

此位是指定要使用的堆栈指针（ISP/USP）的位。如果接受异常处理，此位就变为“0”。如果从管理模式转移到用户模式，此位就变为“1”。

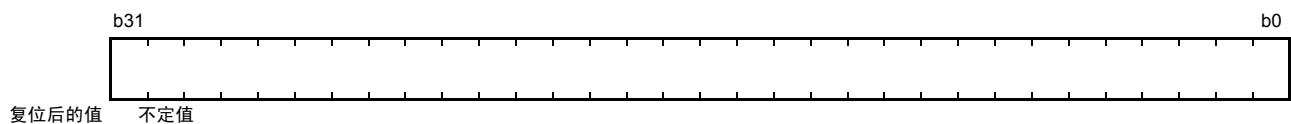
PM 位（处理器模式设定位）

此位是设定处理器模式的位。如果接受异常处理，此位就变为“0”。

IPL[3:0] 位（处理器中断优先级）

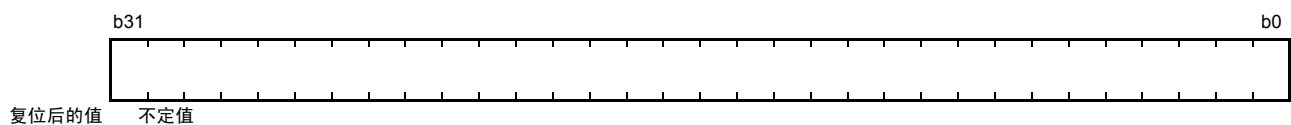
IPL[3:0] 位指定 0 级（最低）～ 15 级（最高）的 16 个处理器中断优先级。如果发生请求的中断的优先级高于处理器中断优先级，就允许该中断。在将 IPL[3:0] 位设定为“15 级”（Fh）时，禁止全部的中断。如果发生非屏蔽中断，IPL[3:0] 位就变为“15 级”（Fh）。如果发生中断，这些位就被设定为所接受中断的优先级。

2.2.2.5 备用 PC（BPC）



备用 PC（BPC）是为实现中断响应高速化而设计的寄存器。如果发生高速中断，就将程序计数器（PC）的内容保存到 BPC。

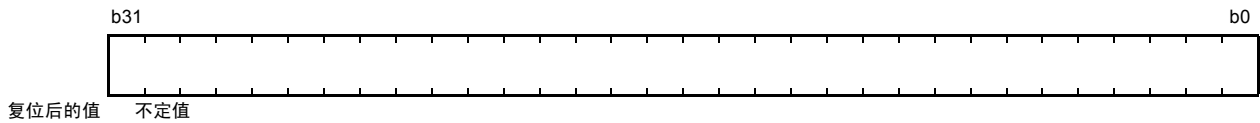
2.2.2.6 备用 PSW（BPSW）



备用 PSW（BPSW）是为实现中断响应高速化而设计的寄存器。

如果发生高速中断，就将处理器状态字（PSW）的内容保存到 BPSW。BPSW 的位配置与 PSW 对应。

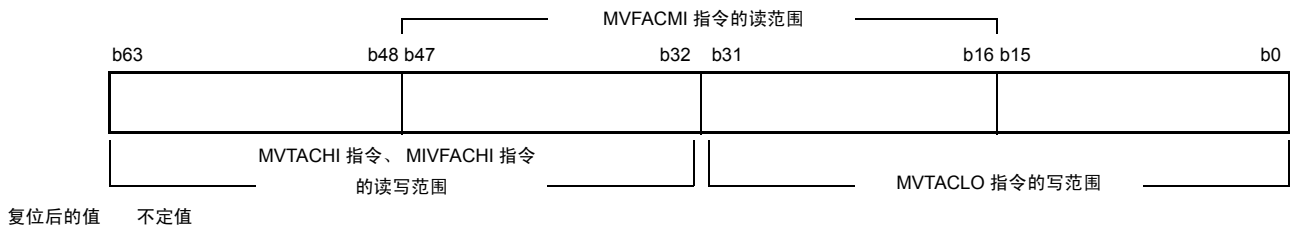
2.2.2.7 高速中断向量寄存器 (FINTV)



高速中断向量寄存器 (FINTV) 是为实现中断响应高速化而设计的寄存器。必须设定发生高速中断时的转移目标地址。

2.2.3 DSP 功能指令的相关寄存器

2.2.3.1 累加器 (ACC)



累加器 (ACC) 是 64 位寄存器, 用于 DSP 功能指令, 也用于乘法指令 (EMUL、EMULU、MUL) 和乘加运算指令 (RMPA)。在执行这些指令时, ACC 的值被更改。

使用 MVTACHI 指令和 MVTACLO 指令写 ACC。通过 MVTACHI 指令将数据写到高 32 位 (b63 ~ b32), 通过 MVTACLO 指令, 将数据写到低 32 位 (b31 ~ b0)。

使用 MVFACHI 指令和 MVFACMI 指令读 ACC。通过 MVFACHI 指令读高 32 位 (b63 ~ b32) 的数据, 通过 MVFACMI 指令读中间 32 位 (b47 ~ b16) 的数据。

2.3 处理器模式

RX CPU 有管理模式和用户模式两种处理器模式。能通过使用处理器模式，实现对 CPU 资源的阶层保护结构。

各处理器模式对能执行的指令以及能存取的 CPU 资源规定了权限，管理模式的权限高于用户模式。复位后，以管理模式运行。

2.3.1 管理模式

在管理模式中，能存取全部的 CPU 资源，还能执行全部的指令。但是，忽视通过 MVTC 指令和 POPC 指令写处理器状态字（PSW）的处理器模式设定位（PM）。有关写 PM 位的方法，请参照“2.2.2.4 处理器状态字（PSW）”。

2.3.2 用户模式

在用户模式中，限制部分 CPU 资源的写存取。被限制写存取的 CPU 资源如下，限制对象为全部指令的存取。

- 处理器状态字（PSW）的部分位（IPL[3:0]、PM、U、I）
- 中断堆栈指针（ISP）
- 中断表寄存器（INTB）
- 备用 PSW（BPSW）
- 备用 PC（BPC）
- 高速中断向量寄存器（FINTV）

2.3.3 特权指令

特权指令是只能在管理模式中执行的指令。如果在用户模式中执行特权指令，就会发生特权指令异常。特权指令有 RTFI、MVTIPL、RTE、WAIT 指令。

2.3.4 处理器模式之间的转移

通过处理器状态字（PSW）的处理器模式设定位（PM）转换处理器模式。但是，通过 MVTC 指令和 POPC 指令对 PM 位的改写无效。必须通过以下所示的方法进行转换。

(1) 用户模式向管理模式的转移

如果发生异常，PSW.PM 位就变为“0”，CPU 转移到管理模式。在管理模式中执行硬件的预处理。被保存的 PSW.PM 位保持发生异常前的处理器模式。

(2) 管理模式向用户模式的转移

在被压栈的 PSW.PM 位为“1”时执行 RTE 指令，或者在被保存到备用 PSW（BPSW）中的 PSW.PM 位为“1”时执行 RTFI 指令，向用户模式转移。一旦转移到用户模式，PSW 的堆栈指针指定位（U）就变为“1”。

2.4 数据类型

RX CPU 能处理整数、位、字符串共 3 种数据。

2.4.1 整数

整数有带符号整数和不带符号整数，带符号整数的负值用 2 的补数表现。

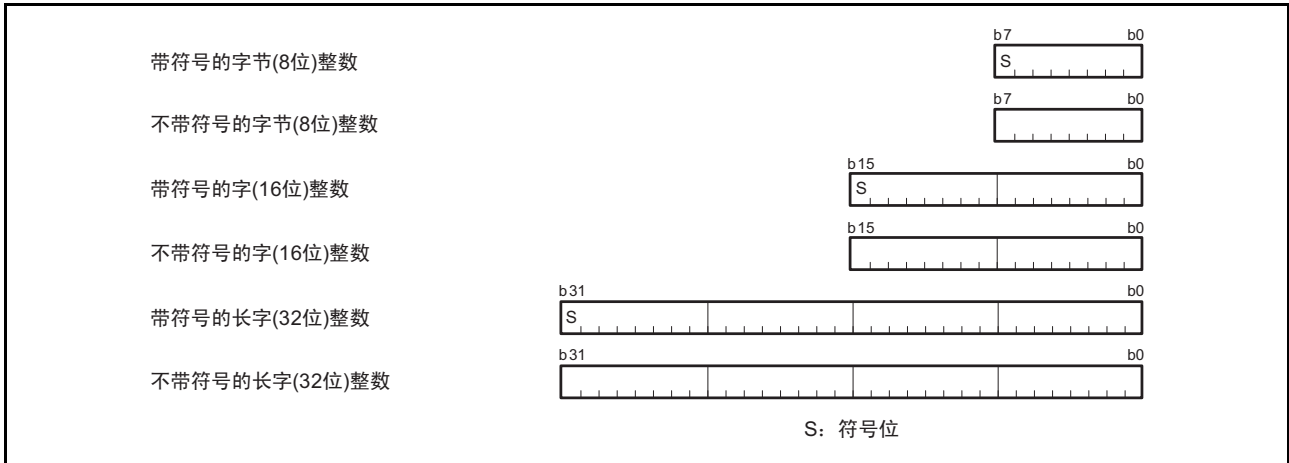


图 2.2 整数

2.4.2 位

位用于位操作指令 BCLR、BMCnd、BNOT、BSET、BTST 共 5 种指令。

通过对象寄存器和 31 ~ 0 的位号指定寄存器的位。

通过对象地址和 7 ~ 0 的位号指定存储器的位。能用于地址指定的寻址方式有寄存器间接和寄存器相对两种。

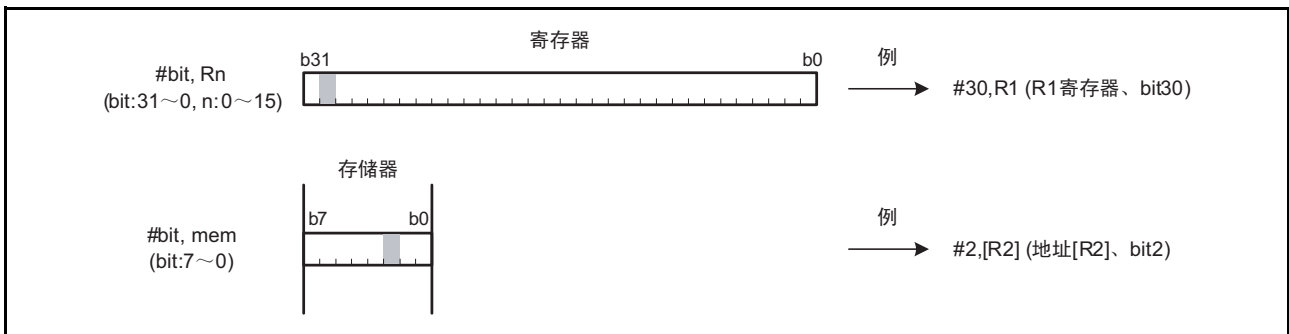


图 2.3 位

2.4.3 字符串

字符串是指只连续排列任意个数的字节（8 位）、字（16 位）或者长字（32 位）数据的数据类型。字符串能用于字符串操作指令 SCMPU、SMOVB、SMOVF、SMOVU、SSTR、SUNTIL、S WHILE 共 7 种指令。

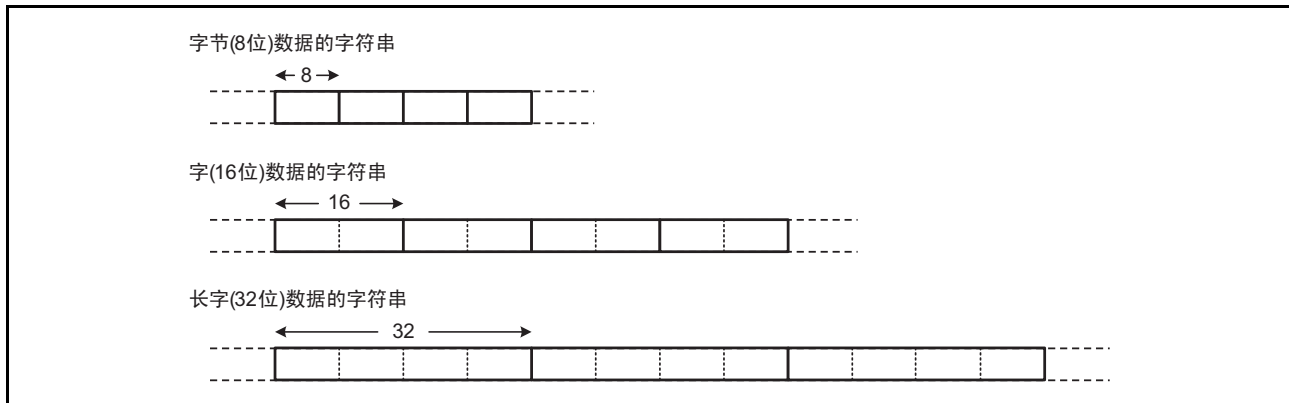


图 2.4 字符串

2.5 字节序

RX CPU 的指令固定为小端法。

数据排列能选择小端法或者大端法。

2.5.1 字节序的设定

在 RX210 群中，能使用高位字节（MSB）为地址 0 的大端法和低位字节（LSB）为地址 0 的小端法两种字节数据的排列方法。

有关字节序的设定，请参照“3. 运行模式”。

使用指令选择 8 位、16 位或者 32 位的存取，并且存取因小端法或者大端法的设定而不同，各存取如表 2.1～表 2.12 所示。

表中的

LL 表示通用寄存器的 D7 ~ D0；

LH 表示通用寄存器的 D15 ~ D8；

HL 表示通用寄存器的 D23 ~ D16；

HH 表示通用寄存器的 D31 ~ D24。

	D31 ~ D24	D23 ~ D16	D15 ~ D8	D7 ~ D0
通用寄存器 Rm	HH	HL	LH	LL

表 2.1 设定为小端法时的 32 位读操作

操作 地址 src	用 32 位读地址 0	用 32 位读地址 1	用 32 位读地址 2	用 32 位读地址 3	用 32 位读地址 4
地址 0	传送到 LL	—	—	—	—
地址 1	传送到 LH	传送到 LL	—	—	—
地址 2	传送到 HL	传送到 LH	传送到 LL	—	—
地址 3	传送到 HH	传送到 HL	传送到 LH	传送到 LL	—
地址 4	—	传送到 HH	传送到 HL	传送到 LH	传送到 LL
地址 5	—	—	传送到 HH	传送到 HL	传送到 LH
地址 6	—	—	—	传送到 HH	传送到 HL
地址 7	—	—	—	—	传送到 HH

表 2.2 设定为大端法时的 32 位读操作

操作 地址 src	用 32 位读地址 0	用 32 位读地址 1	用 32 位读地址 2	用 32 位读地址 3	用 32 位读地址 4
地址 0	传送到 HH	—	—	—	—
地址 1	传送到 HL	传送到 HH	—	—	—
地址 2	传送到 LH	传送到 HL	传送到 HH	—	—
地址 3	传送到 LL	传送到 LH	传送到 HL	传送到 HH	—
地址 4	—	传送到 LL	传送到 LH	传送到 HL	传送到 HH
地址 5	—	—	传送到 LL	传送到 LH	传送到 HL
地址 6	—	—	—	传送到 LL	传送到 LH
地址 7	—	—	—	—	传送到 LL

表 2.3 设定为小端法时的 32 位写操作

操作 地址 dest	用 32 位写地址 0	用 32 位写地址 1	用 32 位写地址 2	用 32 位写地址 3	用 32 位写地址 4
地址 0	从 LL 传送	—	—	—	—
地址 1	从 LH 传送	从 LL 传送	—	—	—
地址 2	从 HL 传送	从 LH 传送	从 LL 传送	—	—
地址 3	从 HH 传送	从 HL 传送	从 LH 传送	从 LL 传送	—
地址 4	—	从 HH 传送	从 HL 传送	从 LH 传送	从 LL 传送
地址 5	—	—	从 HH 传送	从 HL 传送	从 LH 传送
地址 6	—	—	—	从 HH 传送	从 HL 传送
地址 7	—	—	—	—	从 HH 传送

表 2.4 设定为大端法时的 32 位写操作

操作 地址 dest	用 32 位写地址 0	用 32 位写地址 1	用 32 位写地址 2	用 32 位写地址 3	用 32 位写地址 4
地址 0	从 HH 传送	—	—	—	—
地址 1	从 HL 传送	从 HH 传送	—	—	—
地址 2	从 LH 传送	从 HL 传送	从 HH 传送	—	—
地址 3	从 LL 传送	从 LH 传送	从 HL 传送	从 HH 传送	—
地址 4	—	从 LL 传送	从 LH 传送	从 HL 传送	从 HH 传送
地址 5	—	—	从 LL 传送	从 LH 传送	从 HL 传送
地址 6	—	—	—	从 LL 传送	从 LH 传送
地址 7	—	—	—	—	从 LL 传送

表 2.5 设定为小端法时的 16 位读操作

操作 地址 src	用 16 位 读地址 0	用 16 位 读地址 1	用 16 位 读地址 2	用 16 位 读地址 3	用 16 位 读地址 4	用 16 位 读地址 5	用 16 位 读地址 6
地址 0	传送到 LL	—	—	—	—	—	—
地址 1	传送到 LH	传送到 LL	—	—	—	—	—
地址 2	—	传送到 LH	传送到 LL	—	—	—	—
地址 3	—	—	传送到 LH	传送到 LL	—	—	—
地址 4	—	—	—	传送到 LH	传送到 LL	—	—
地址 5	—	—	—	—	传送到 LH	传送到 LL	—
地址 6	—	—	—	—	—	传送到 LH	传送到 LL
地址 7	—	—	—	—	—	—	传送到 LH

表 2.6 设定为大端法时的 16 位读操作

操作 地址 src	用 16 位 读地址 0	用 16 位 读地址 1	用 16 位 读地址 2	用 16 位 读地址 3	用 16 位 读地址 4	用 16 位 读地址 5	用 16 位 读地址 6
地址 0	传送到 LH	—	—	—	—	—	—
地址 1	传送到 LL	传送到 LH	—	—	—	—	—
地址 2	—	传送到 LL	传送到 LH	—	—	—	—
地址 3	—	—	传送到 LL	传送到 LH	—	—	—
地址 4	—	—	—	传送到 LL	传送到 LH	—	—
地址 5	—	—	—	—	传送到 LL	传送到 LH	—
地址 6	—	—	—	—	—	传送到 LL	传送到 LH
地址 7	—	—	—	—	—	—	传送到 LL

表 2.7 设定为小端法时的 16 位写操作

操作 地址 dest	用 16 位 写地址 0	用 16 位 写地址 1	用 16 位 写地址 2	用 16 位 写地址 3	用 16 位 写地址 4	用 16 位 写地址 5	用 16 位 写地址 6
地址 0	从 LL 传送	—	—	—	—	—	—
地址 1	从 LH 传送	从 LL 传送	—	—	—	—	—
地址 2	—	从 LH 传送	从 LL 传送	—	—	—	—
地址 3	—	—	从 LH 传送	从 LL 传送	—	—	—
地址 4	—	—	—	从 LH 传送	从 LL 传送	—	—
地址 5	—	—	—	—	从 LH 传送	从 LL 传送	—
地址 6	—	—	—	—	—	从 LH 传送	从 LL 传送
地址 7	—	—	—	—	—	—	从 LH 传送

表 2.8 设定为大端法时的 16 位写操作

操作 地址 dest	用 16 位 写地址 0	用 16 位 写地址 1	用 16 位 写地址 2	用 16 位 写地址 3	用 16 位 写地址 4	用 16 位 写地址 5	用 16 位 写地址 6
地址 0	从 LH 传送	—	—	—	—	—	—
地址 1	从 LL 传送	从 LH 传送	—	—	—	—	—
地址 2	—	从 LL 传送	从 LH 传送	—	—	—	—
地址 3	—	—	从 LL 传送	从 LH 传送	—	—	—
地址 4	—	—	—	从 LL 传送	从 LH 传送	—	—
地址 5	—	—	—	—	从 LL 传送	从 LH 传送	—
地址 6	—	—	—	—	—	从 LL 传送	从 LH 传送
地址 7	—	—	—	—	—	—	从 LL 传送

表 2.9 设定为小端法时的 8 位读操作

操作 地址 src	用 8 位读地址 0	用 8 位读地址 1	用 8 位读地址 2	用 8 位读地址 3
地址 0	传送到 LL	—	—	—
地址 1	—	传送到 LL	—	—
地址 2	—	—	传送到 LL	—
地址 3	—	—	—	传送到 LL

表 2.10 设定为大端法时的 8 位读操作

操作 地址 src	用 8 位读地址 0	用 8 位读地址 1	用 8 位读地址 2	用 8 位读地址 3
地址 0	传送到 LL	—	—	—
地址 1	—	传送到 LL	—	—
地址 2	—	—	传送到 LL	—
地址 3	—	—	—	传送到 LL

表 2.11 设定为小端法时的 8 位写操作

操作 地址 dest	用 8 位写地址 0	用 8 位写地址 1	用 8 位写地址 2	用 8 位写地址 3
地址 0	从 LL 传送	—	—	—
地址 1	—	从 LL 传送	—	—
地址 2	—	—	从 LL 传送	—
地址 3	—	—	—	从 LL 传送

表 2.12 设定为大端法时的 8 位写操作

操作 地址 dest	用 8 位写地址 0	用 8 位写地址 1	用 8 位写地址 2	用 8 位写地址 3
地址 0	从 LL 传送	—	—	—
地址 1	—	从 LL 传送	—	—
地址 2	—	—	从 LL 传送	—
地址 3	—	—	—	从 LL 传送

2.5.2 I/O 寄存器的存取

I/O 寄存器分配在固定地址，与设定的大端法或者小端法无关。因此，I/O 寄存器的存取不受字节序变更的影响。有关 I/O 寄存器的分配，请参照各章的寄存器说明。

2.5.3 存取 I/O 寄存器时的注意事项

必须按照以下规则存取 I/O 寄存器：

- 在存取 8 位总线宽度的 I/O 寄存器时，必须使用长度说明符（.size）为“B”的指令或者长度扩展说明符（.memex）为“B”或者“UB”的指令。
- 在存取 16 位总线宽度的 I/O 寄存器时，必须使用长度说明符（.size）为“W”的指令或者长度扩展说明符（.memex）为“W”或者“UW”的指令。
- 在存取 32 位总线宽度的 I/O 寄存器时，必须使用长度说明符（.size）为“L”的指令或者长度扩展说明符（.memex）为“L”的指令。

2.5.4 数据排列

2.5.4.1 寄存器的数据排列

寄存器的数据长度和位号的关系如图 2.5 所示。

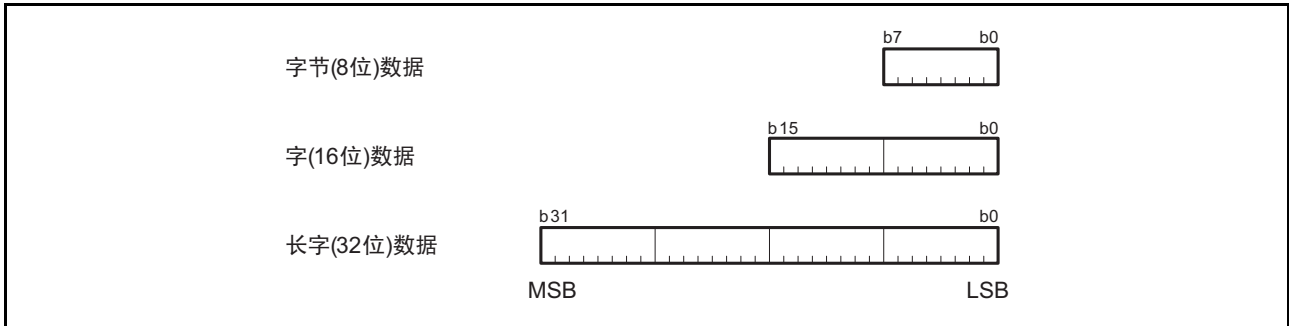


图 2.5 寄存器的数据排列

2.5.4.2 存储器的数据排列

存储器的数据长度有字节（8 位）、字（16 位）和长字（32 位）共 3 种，能选择小端法或者大端法的数据排列方式。存储器的数据排列如图 2.6 所示。

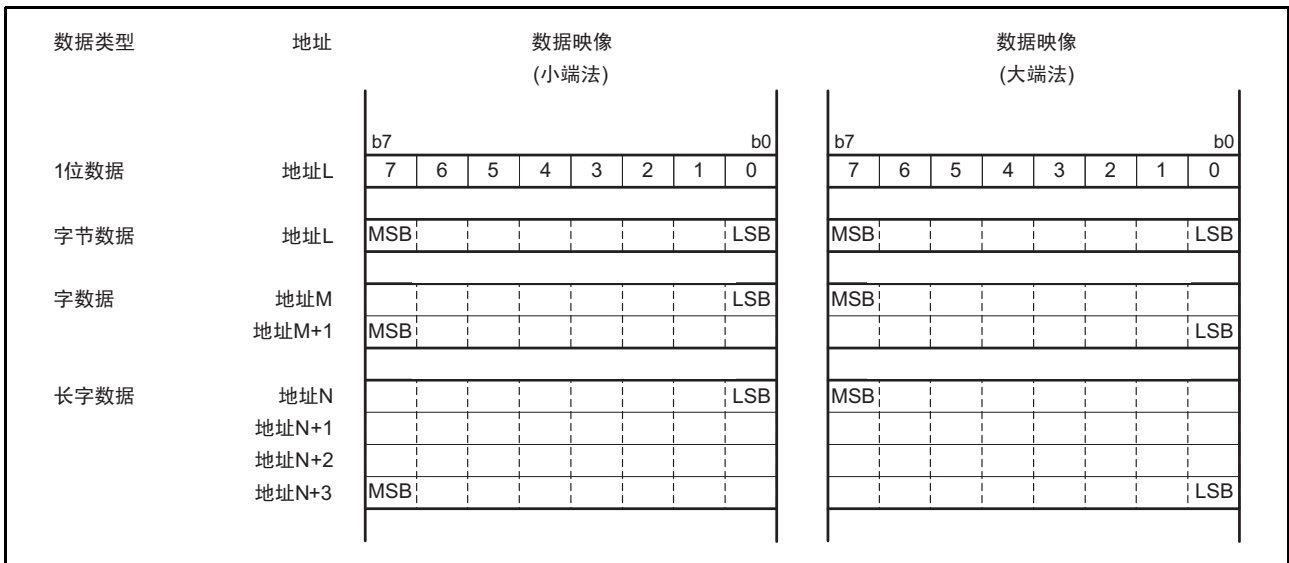


图 2.6 存储器的数据排列

2.5.5 分配指令码的注意事项

不能将指令编码分配到与芯片字节序设定不同的外部空间字节序设定区域。如果要将指令码分配到外部空间，就必须分配到与芯片字节序相同的字节序设定区域。

2.6 向量表

向量表有固定向量表和可变向量表。向量表的 1 个向量由 4 字节构成，各向量设有对应的异常处理程序的起始地址。

2.6.1 固定向量表

固定向量表是表的分配地址被固定的向量表。特权指令异常、未定义指令异常、非屏蔽中断、复位的各向量分配在地址 FFFFFFF80h ~ FFFFFFFFh。固定向量表如图 2.7 所示。

	MSB	LSB
FFFFFFF80h	(保留区)	
:	:	
FFFFFFFCCh	(保留区)	
FFFFFFD0h	特权指令异常	
FFFFFFD4h	(保留区)	
FFFFFFD8h	(保留区)	
FFFFFFDCh	未定义指令异常	
FFFFFFE0h	(保留区)	
FFFFFFE4h	浮点异常	
FFFFFFE8h	(保留区)	
FFFFFFECh	(保留区)	
FFFFFFF0h	(保留区)	
FFFFFFF4h	(保留区)	
FFFFFFF8h	非屏蔽中断	
FFFFFFFCh	复位	

图 2.7 固定向量表

2.6.2 可变向量表

可变向量表是能改变表的分配地址的向量表。无条件陷阱和中断的各向量分配在以中断表寄存器（INTB）的内容所示的值为起始地址（IntBase）的 1024 字节的区域。可变向量表如图 2.8 所示。

可变向量表中的每个向量带有序号（0 ~ 255）。在无条件陷阱发生源的 INT 指令时，分配了与 INT 指令号（0 ~ 255）对应的向量，而在 BRK 指令时，分配了序号 0 的向量。另外，在中断源时，分配了各产品规定的向量号（0 ~ 255）。有关中断的向量号，请参照“14.3.1 中断向量表”。

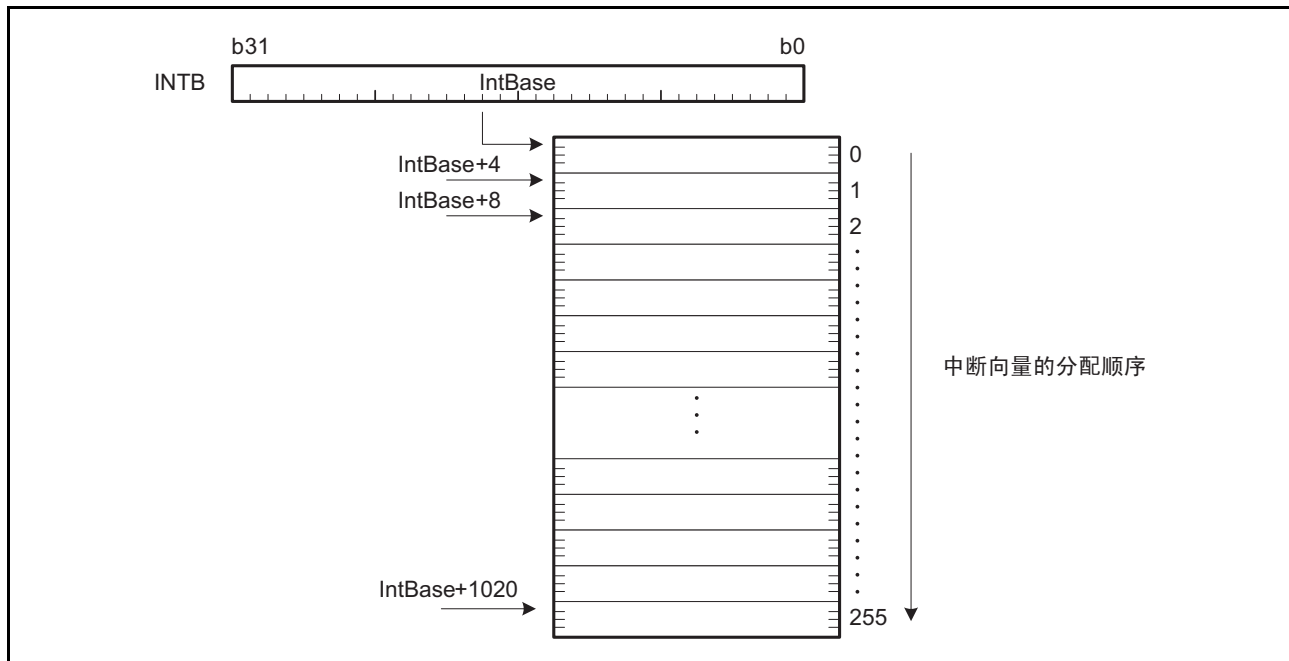


图 2.8 可变向量表

2.7 指令操作

2.7.1 RMPA 指令和字符串操作指令的数据预取

为了从存储器快速读取数据，SSTR 指令以外的字符串操作指令（SCMPU、SMOVB、SMOVF、SMOVU、SUNTIL、SWHILE）和 RMPA 指令有可能进行数据预取。对应数据读取位置，能进行最多到该位置之后的 3 字节为止的数据预取。各指令的数据读取位置如下：

- RMPA 指令：R1 指定的被乘数地址以及 R2 指定的乘数地址
- SCMPU 指令：R1 指定的比较源地址以及 R2 指定的比较目标地址
- SUNTIL 指令和 SWHILE 指令：R1 指定的比较目标地址
- SMOVB、SMOVF、SMOVU 指令：R2 指定的传送源地址

2.8 流水线

2.8.1 概要

RX CPU 由 5 个阶段的流水线构成。RX CPU 的指令转换为 1 个或者多个微操作，RX CPU 对微操作进行流水线处理。流水线 IF 阶段以指令为单位进行操作，D 以后的阶段以微操作为单位进行操作。

流水线的操作和各阶段的概要如下所示：

(1) IF 阶段（取指令阶段）

这是从存储器取指令的阶段。RX CPU 有 4 个 4 字节的指令队列，与 D（解码）阶段的解码处理结束无关，继续取指令，直到指令队列满为止。

(2) D 阶段（解码阶段）

D 阶段进行指令的解码处理（DEC），并且将指令转换为微操作。如果在此阶段读寄存器（RF）并且要参照前面指令的运算结果，就进行旁路处理（BYP）。通过旁路处理，能将运算结果写到寄存器（RW），同时在 D 阶段参照寄存器。

(3) E 阶段（执行阶段）

进行运算和地址计算等（OP）。

(4) M 阶段（存储器存取阶段）

此阶段进行操作数的存储器存取（OA1 和 OA2），只在存储器存取时使用此阶段。此阶段又分为 M1 和 M2 两个子阶段。RX CPU 中 M1 阶段和 M2 阶段各有 1 个存储器存取。

- M1 阶段（存储器存取阶段 1）

此阶段进行操作数的存储器存取（OA1）。

存储时：一旦总线接受写请求，就结束流水线处理。

加载时：一旦总线接受读请求，就进入 M2 阶段。如果请求的接受和加载数据的到达同时（无等待的存储器存取），就进入 WB 阶段。

- M2 阶段（存储器存取阶段 2）

此阶段进行操作数的存储器存取（OA2）。这是等待加载数据到达的阶段。一旦加载数据到达，就进入 WB 阶段。

(5) WB 阶段（回写阶段）

将运算结果以及从存储器读取的数据写到寄存器（RW）。能将存储器读取的数据以及其他运算结果同时（相同周期）写到寄存器。

流水线结构及其运行如图 2.9 所示。

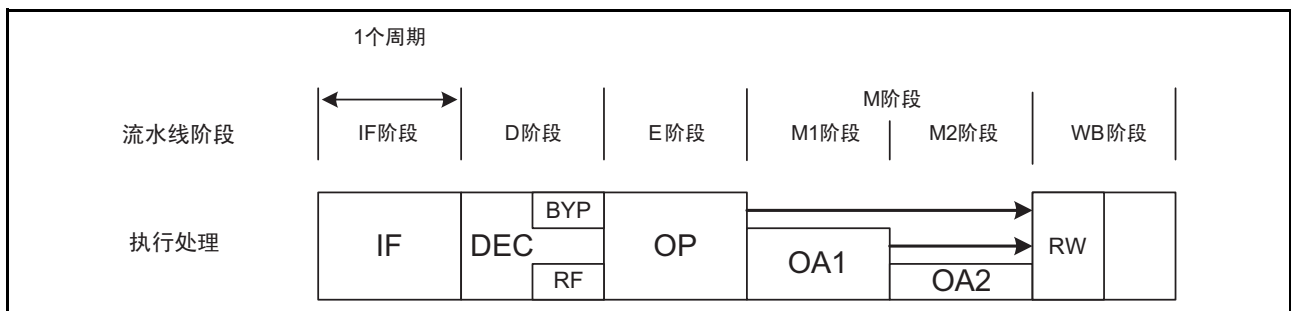


图 2.9 流水线结构及其运行

2.8.2 指令和流水线处理

表中操作数的表示法如下：

#IMM: 立即数

flag: 位、标志

Rs、Rs2、Rd、Rd2、Ri、Rb: 通用寄存器

CR: 控制寄存器

dsp: 位移量

pcdsp: 位移量

2.8.2.1 转换为单一微操作的指令和流水线处理

转换为单一微操作的指令如下所示，周期数表示无等待存储器存取时的周期数。

表 2.13 转换为单一微操作的指令

指令	助记符 (当省略指令长度时, 为全长度共同的操作)	参照图	周期数
算术 / 逻辑运算指令 (寄存器之间、立即数 - 寄存器) DIV、DIVU、EMUL、EMULU、 RMPA 和 SATR 除外	<ul style="list-style-type: none"> {ABS, NEG, NOT} "Rd"/"Rs, Rd" {ADC, MAX, MIN, ROTL, ROTR, XOR} "#IMM, Rd"/"Rs, Rd" ADD "#IMM, Rd"/"Rs, Rd"/"#IMM, Rs, Rd"/"Rs, Rs2, Rd" {AND, MUL, OR, SUB} "#IMM, Rd"/"Rs, Rd"/"Rs, Rs2, Rd" {CMP, TST} "#IMM, Rs"/"Rs, Rs2" NOP {ROL, ROR, SAT} "Rd" SBB "Rs, Rd" {SHAR, SHLL, SHLR} "#IMM, Rd"/"Rs, Rd"/"#IMM, Rs, Rd" 	图 2.10	1
算术 / 逻辑运算指令 (除法)	DIV "#IMM, Rd"/"Rs, Rd"	图 2.10	3 ~ 20 (注 1)
	DIVU "#IMM, Rd"/"Rs, Rd"	图 2.10	2 ~ 18 (注 1)
传送指令 (寄存器之间、立即数 - 寄存器)	<ul style="list-style-type: none"> MOV "#IMM, Rd"/"Rs, Rd" {MOVU, REVL, REVW} "Rs, Rd" SCCnd "Rd" {STNZ, STZ} "#IMM, Rd" 	图 2.10	1
传送指令 (加载)	<ul style="list-style-type: none"> {MOV, MOVU} "[Rs], Rd"/"dsp[Rs], Rd"/"[Rs+], Rd" "[-Rs], Rd"/"[Ri, Rb], Rd" POP "Rd" 	图 2.11	吞吐量: 1 等待时间: 2 (注 2)
传送指令 (存储)	<ul style="list-style-type: none"> MOV "Rs, [Rd]"/"Rs, dsp[Rd]"/"Rs, [Rd+]" "Rs, [-Rd]"/"Rs, [Ri, Rb]"/"#IMM, dsp[Rd]"/"#IMM, [Rd]" PUSH "Rs" PUSHC "CR" SCCnd "[Rd]"/"dsp[Rd]" 	图 2.12	1
位操作指令 (寄存器)	<ul style="list-style-type: none"> {BCLR, BNOT, BSET} "#IMM, Rd"/"Rs, Rd" BMCnd "#IMM, Rd" BTST "#IMM, Rs"/"Rs, Rs2" 	图 2.10	1
转移指令	<ul style="list-style-type: none"> BCnd "pcdsp" {BRA, BSR} "pcdsp"/"Rs" {JMP, JSR} "Rs" 	图 2.20	转移成立: 3 转移不成立: 1
系统控制指令	<ul style="list-style-type: none"> {CLRPSW, SETPSW} "flag" MVTC "#IMM, CR"/"Rs, CR" MVFC "CR, Rd" MVTIPL "#IMM" 	—	1
DSP 功能指令	<ul style="list-style-type: none"> {MACHI, MACLO, MULHI, MULLO} "Rs, Rs2" {MVFACHI, MVFACMI} "Rd" {MVTACHI, MVTACLO} "Rs" RACW "#IMM" 	图 2.10	1

注 1. 除法指令的周期数因除数和被除数的值而不同。

注 2. 有关吞吐量以及表示等待时间的周期数, 请参照“2.8.3 指令处理时间的计算方法”。

转换为单一微操作的指令操作如图 2.10 ~ 图 2.12 所示。

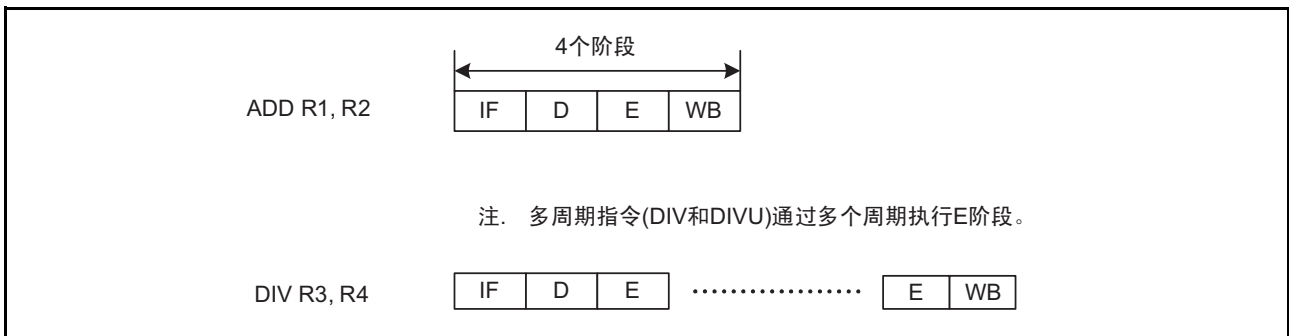


图 2.10 寄存器之间、立即数 - 寄存器的运算

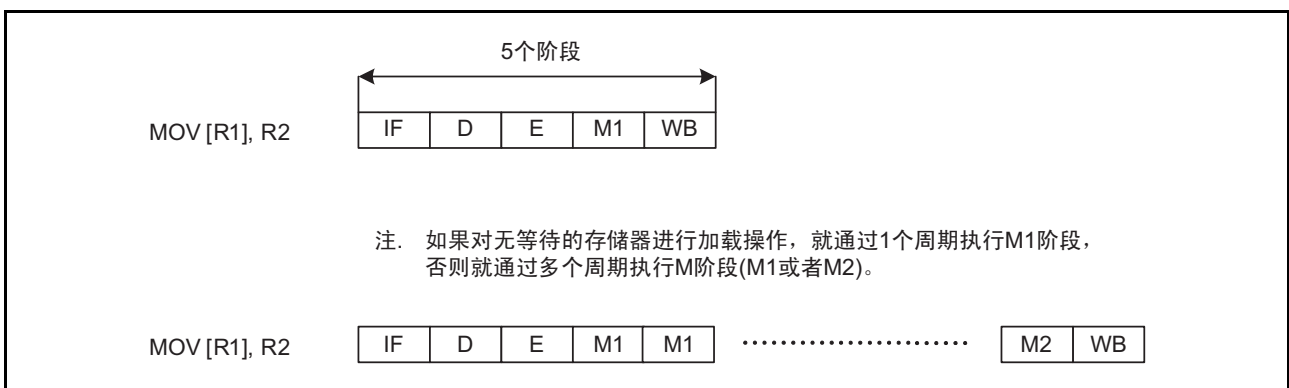


图 2.11 加载操作

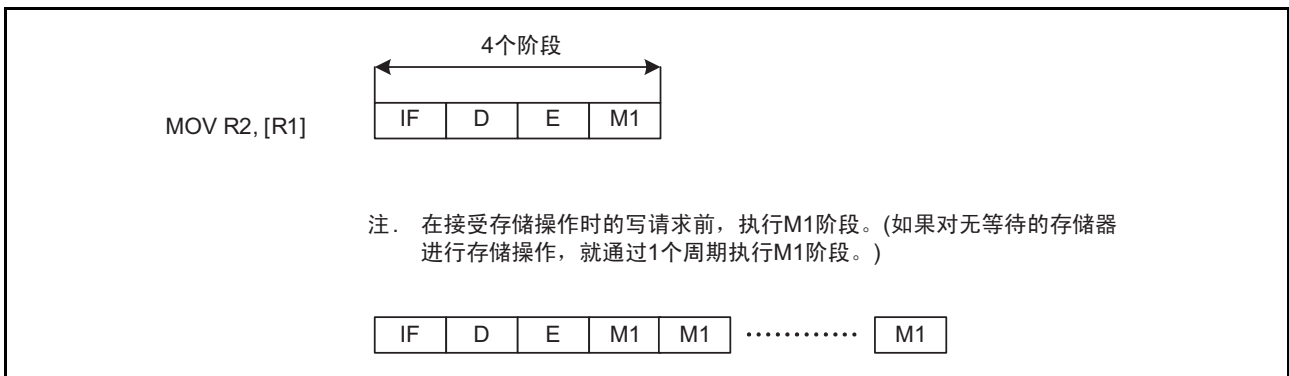


图 2.12 存储操作

2.8.2.2 转换为多个微操作的指令和流水线处理

转换为多个微操作的指令如下所示，周期数表示无等待存储器存取时的周期数。

表 2.14 转换为多个微操作的指令 (1/2)

指令	助记符 (当省略指令长度时, 为全长度共同的操作)	参照图	周期数
算术 / 逻辑运算指令 (存储器源操作数)	<ul style="list-style-type: none"> {ADC, ADD, AND, MAX, MIN, MUL, OR, SBB, SUB, XOR} "[Rs], Rd" / "dsp[Rs], Rd" {CMP, TST} "[Rs], Rs2" / "dsp[Rs], Rs2" 	图 2.13	3
算术 / 逻辑运算指令 (除法)	DIV "[Rs], Rd / dsp[Rs], Rd"	—	5 ~ 22
	DIVU "[Rs], Rd / dsp[Rs], Rd"	—	4 ~ 20
算术 / 逻辑运算指令 (乘法 32bit×32bit→64bit) (寄存器之间、寄存器 - 立即数)	• {EMUL, EMULU} "#IMM, Rd" / "Rs, Rd"	图 2.15	2
算术 / 逻辑运算指令 (乘法 32bit×32bit→64bit) (存储器源操作数)	• {EMUL, EMULU} "[Rs], Rd" / "dsp[Rs], Rd"	—	4
算术 / 逻辑运算指令 (乘加运算)	• RMPA.B	—	6+7×floor(n/4)+4×(n%4) n 为处理字节数 (注1)
	• RMPA.W	—	6+5×floor(n/2)+4×(n%2) n 为处理字数 (注1)
	• RMPA.L	—	6+4n n 为处理长字数 (注1)
算术 / 逻辑运算指令 (RMPA 指令的 64 位带符号的饱和处理)	• SATR	—	3
传送指令 (存储器之间的传送)	<ul style="list-style-type: none"> MOV "[Rs], [Rd]" / "dsp[Rs], [Rd]" / "Rs, dsp[Rd]" / "dsp[Rs], dsp[Rd]" PUSH "[Rs]" / "dsp[Rs]" 	图 2.14	3
位操作指令 (存储器源操作数)	<ul style="list-style-type: none"> {BCLR, BNOT, BSET} "#IMM, [Rd]" / "#IMM, dsp[Rd]" / "Rs, [Rd]" / "Rs, dsp[Rd]" BM<i>Cnd</i> "#IMM, [Rd]" / "#IMM, dsp[Rd]" BTST "#IMM, [Rs]" / "#IMM, dsp[Rs]" / "Rs, [Rs2]" / "Rs, dsp[Rs2]" 	图 2.14	3
传送指令 (加载指令)	• POPC "CR"	—	吞吐量: 3 等待时间: 4 (注2)
传送指令 (多个寄存器的保存)	• PUSHM "Rs-Rs2"	—	n n 为寄存器的个数 (注3)
传送指令 (多个寄存器的恢复)	• POPM "Rs-Rs2"	—	吞吐量: n 等待时间: n+1 n 为寄存器的个数 (注2、注4)
传送指令 (寄存器之间的交换)	• XCHG "Rs, Rd"	图 2.16	2
传送指令 (存储器 - 寄存器的交换)	• XCHG "[Rs], Rd" / "dsp[Rs], Rd"	图 2.17	2
转移指令	• RTS	—	5
	• RTSD "#IMM"	—	5
	• RTSD "#IMM, Rd-Rd2"	—	吞吐量: n<5?5:1+n 等待时间: n<4?5:2+n n 为寄存器的个数 (注2)

表 2.14 转换为多个微操作的指令 (2/2)

指令	助记符 (当省略指令长度时, 为全长度共同的操作)	参照图	周期数
字符串操作指令 (注5)	• SCMPU	—	$2+4 \times \text{floor}(n/4)+4 \times (n\%4)$ n 为比较字节数 (注1)
	• SMOVB	—	n>3? $6+3 \times \text{floor}(n/4)+3 \times (n\%4):$ $2+3n$ n 为传送字节数 (注1)
	• SMOV, SMOVU	—	$2+3 \times \text{floor}(n/4)+3 \times (n\%4)$ n 为传送字节数 (注1)
	• SSTR.B	—	$2+\text{floor}(n/4)+n\%4$ n 为传送字节数 (注1)
	• SSTR.W	—	$2+\text{floor}(n/2)+n\%2$ n 为传送字数 (注1)
	• SSTR.L	—	$2+n$ n 为传送长字数
	• SUNTIL.B, SWHILE.B	—	$3+3 \times \text{floor}(n/4)+3 \times (n\%4)$ n 为比较字节数 (注1)
	• SUNTIL.W, SWHILE.W	—	$3+3 \times \text{floor}(n/2)+3 \times (n\%2)$ n 为比较字数 (注1)
	• SUNTIL.L, SWHILE.L	—	$3+3 \times n$ n 为比较长字数
系统控制指令	• RTE	—	6
	• RTFI	—	3

? : 条件运算符

注 1. floor(x): 小于等于 x 的最大整数

注 2. 有关吞吐量以及表示等待时间的周期数, 请参照“2.8.3 指令处理时间的计算方法”。

注 3. 将 PUSHM 指令转换为多个存储操作。MOV 指令的存储操作和指定寄存器次数的重复是相同的流水线处理。

注 4. 将 POPM 指令转换为多个加载操作。MOV 指令的加载操作和指定寄存器次数的重复是相同的流水线处理。

注 5. 如果在执行 SCMPU、SMOVU、SWHILE、SUNTIL 各指令时满足结束条件, 就结束指令的执行, 与记载的周期无关。

转换为多个微操作的指令操作如图 2.13 ~ 图 2.17 所示。

注. mop: 微操作; stall: 流水线延迟

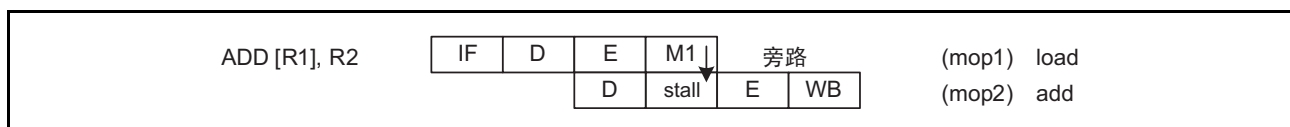


图 2.13 算术 / 逻辑运算指令 (存储器源操作数)

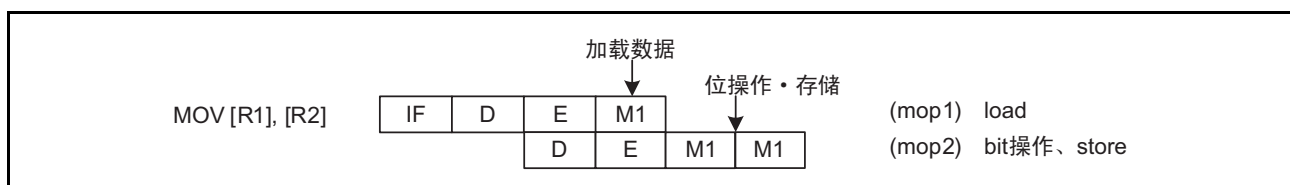


图 2.14 MOV 指令 (存储器之间的传送) 和位操作指令 (存储器源操作数)

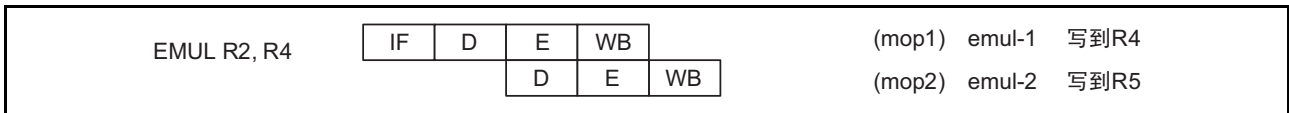


图 2.15 EMUL 指令和 EMULU 指令（寄存器之间、寄存器 - 立即数）

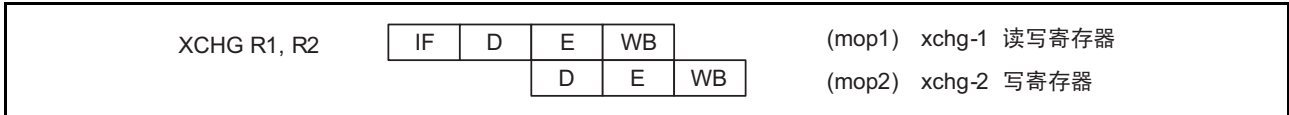


图 2.16 XCHG 指令（寄存器）

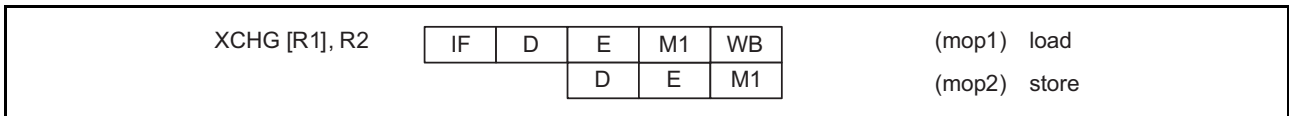


图 2.17 XCHG 指令（存储器源操作数）

2.8.2.3 流水线的基本操作

在理想的流水线处理中，各阶段的执行周期数为“1”，但是流水线处理有可能因各阶段中的处理以及转移执行等而发生混乱。

CPU 在 IF 阶段以指令为单位进行流水线阶段控制，在 D 以后的阶段以微操作为单位进行流水线阶段控制。

典型的流水线处理状况如下所示。

注. mop: 微操作; stall: 流水线延迟

(1) 流水线处理混乱的情况

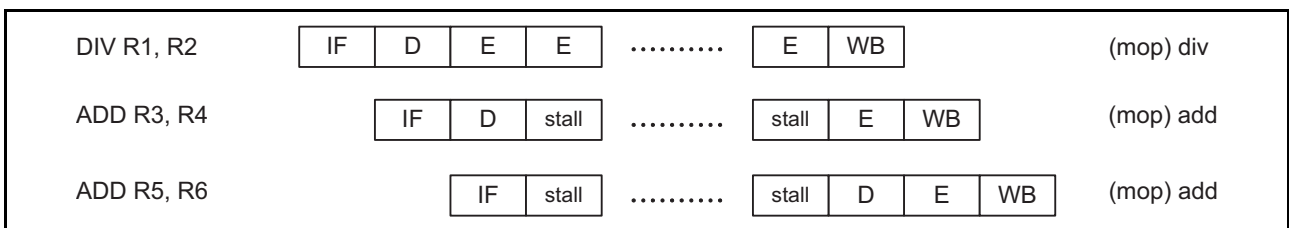


图 2.18 在执行 E 阶段过程中执行需要多个周期的指令的情况

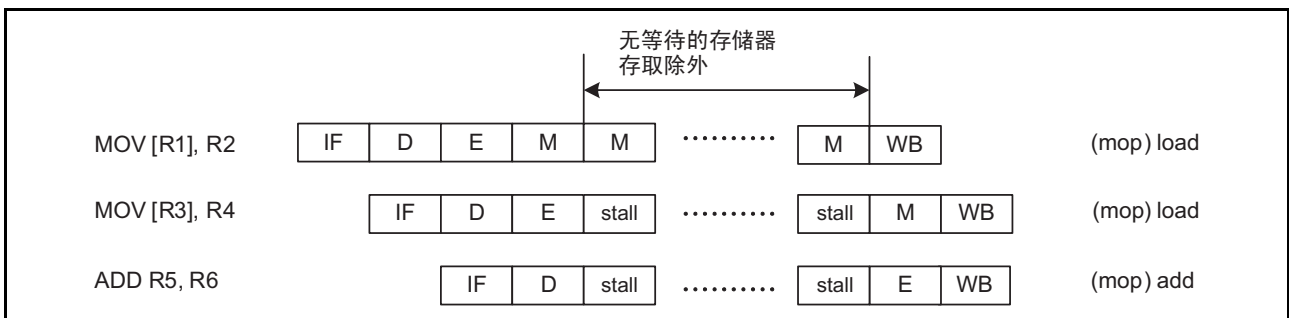


图 2.19 操作数存取没有在 1 个周期中结束的情况

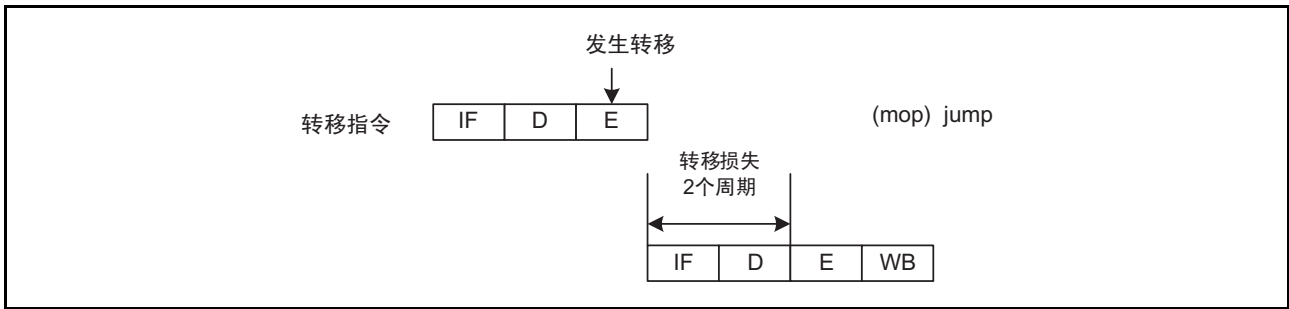


图 2.20 转移（在无转移或者条件转移中条件成立的情况）

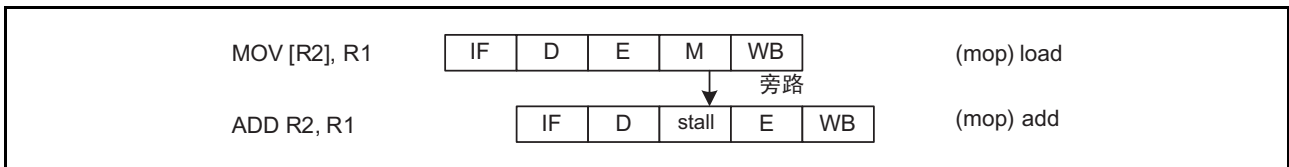


图 2.21 后续指令使用从存储器读取的操作数的情况

(2) 流水线处理不混乱的情况

(a) 旁路

即使后续指令使用前面的指令写的寄存器，在进行寄存器之间的运算时，流水线处理也不会因旁路而发生混乱。

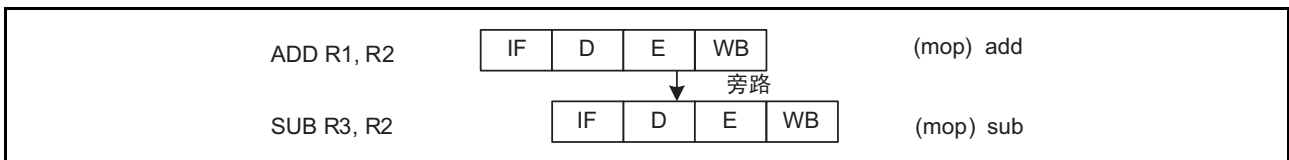


图 2.22 旁路

(b) 存储器加载和运算的 WB 阶段重叠的情况

因为加载数据和运算结果能同时写到寄存器，所以即使存储器加载和运算的 WB 阶段重叠，流水线处理也不会混乱。

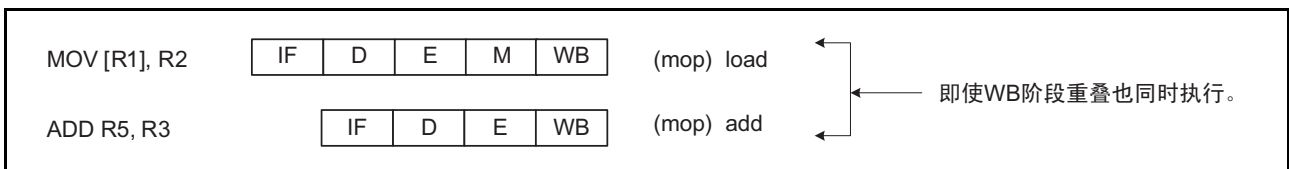


图 2.23 存储器加载和运算的 WB 阶段重叠的情况

(c) 在存储器加载结束前后续指令写相同寄存器的情况

在存储器加载结束前，即使后续指令写相同的寄存器，因为取消了存储器加载的 WB 阶段，所以流水线处理不会混乱。

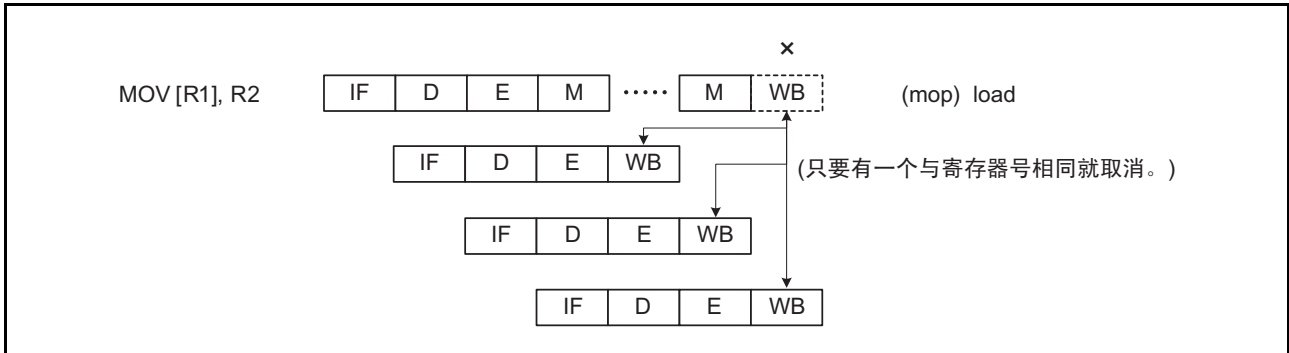


图 2.24 在结束存储器加载前，后续指令写相同寄存器的情况

(d) 后续指令不参照存储器加载的数据的情况

当后续指令不参照存储器加载的数据时，先执行后续的指令，然后结束。
(Out-of-Order Completion)

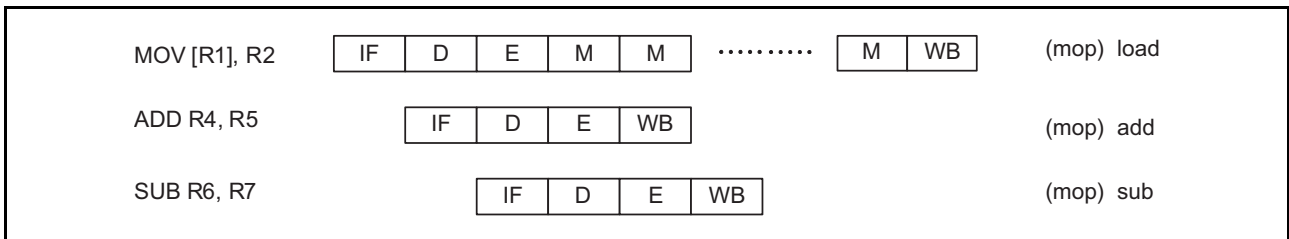


图 2.25 后续指令不参照存储器加载的数据的情况

2.8.3 指令处理时间的计算方法

CPU 的指令处理时间因流水线处理而不同，但是能使用以下的计算方法概算指令的处理时间。

- 计算周期数（参照表 2.13 和表 2.14）。
- 如果后续指令参照存储器加载的结果，存储器加载的指令周期数就为“等待时间”的周期数。否则，就为“吞吐量”的周期数。
- 如果引起取指令的延迟，就再追加周期数。
- 根据系统结构，存储器存取有时需要多个周期。存储器存取周期数取决于机种。

2.8.4 中断响应周期数

中断响应处理的周期数如表 2.15 所示。

表 2.15 中断响应周期数

中断请求的种类 / 处理内容	高速中断	高速中断以外的中断
ICU 判断优先级。	2 个周期	
CPU 从通知中断请求到接受中断的周期数。	N 个周期 (因正在执行的指令而不同。)	
CPU 硬件预处理 保存到 PC 或者 PSW 的 RAM。 (将高速中断保存到控制寄存器。) 读向量。 转移到异常处理程序的起始位置。	4 个周期	6 个周期

表 2.15 是 CPU 的存储器存取全部以无等待处理时的中断响应时间。RX210 群装载了能无等待存取的内部 ROM 和内部 RAM。通过将程序（包括向量）分配到内部 ROM，将堆栈区分配到内部 RAM，能将中断响应周期数缩到最短。而且，异常处理程序的起始地址必须指定 8 字节边界。

有关从通知中断请求到接受中断的周期数 N，请参照“表 2.13 转换为单一微操作的指令”和“表 2.14 转换为多个微操作的指令”。

中断接受时序取决于流水线状态。有关中断的接受时序，请参照“13.3.1 接受时序和被保存的 PC 值”。

3. 运行模式

3.1 运行模式的种类和选择

通过 MD 引脚和系统控制寄存器 0 (SYSCR0) 的内部 ROM 有效位 (ROME)、外部总线有效位 (EXBE) 设定运行模式。

在各运行模式中，能选择字节序。通过选项设定存储器的字节序选择寄存器 (MDEB、MDES) 的字节序选择位 (MDE[2:0]) 选择字节序。使用用户引导模式时需要设定选项设定存储器的 UB 码 A 和 UB 码 B。有关选项设定存储器，请参照“7. 选项设定存储器”。

在 LSI 的运行过程中不能更改 MD 引脚。另外，不能设定表 3.1 中没有的组合。

表 3.1 通过模式引脚选择的运行模式

模式引脚 MD	PC7 引脚	运行模式	SYSCR0 寄存器初始状态		内部 ROM (注 1)	外部总线
			ROME	EXBE		
Low	Low	引导模式 (SCI 模式)	1	0	有效	无效
	High	用户引导模式	1	0	有效	无效
High	—	单芯片模式	1	0	有效	无效

注 1. 内部 ROM 有 ROM 和 E2 数据闪存，详细内容请参照“39. ROM (保存代码的闪存)”和“40. E2 数据闪存 (保存数据的闪存)”。

表 3.2 通过寄存器选择的运行模式

SYSCR0 寄存器		运行模式	内部 ROM (注 1)	外部总线
ROME	EXBE			
0	0	单芯片模式	无效	无效
1	0		有效	无效
0	1	内部 ROM 无效扩展模式	无效	有效
1	1	内部 ROM 有效扩展模式	有效	有效

注 1. 内部 ROM 有 ROM 和 E2 数据闪存，详细内容请参照“39. ROM (保存代码的闪存)”和“40. E2 数据闪存 (保存数据的闪存)”。

表 3.3 字节序的设定方法

运行模式	字节序的设定
用户引导模式	通过选项设定存储器的字节序选择寄存器 (MDEB) 选择
单芯片模式	通过选项设定存储器的字节序选择寄存器 (MDES) 选择

注. 选项设定存储器的字节序选择寄存器 (MDEB、MDES) 存在于 ROM 中。

表 3.4 字节序的选择

MDEB.MDE[2:0] 位或者 MDES.MDS[2:0] 位	字节序
000b	小端法
111b	大端法

3.2 寄存器说明

3.2.1 模式监视寄存器 (MDMONR)

地址 0008 0000h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MD
复位后的值	0	0	0	0	0	0	0	x	0	0	0	0	0	0	0	0/1 (注1)

注 1. 因模式引脚 (MD) 的设定而不同。MD 引脚在 Low 电平时为“0”，在 High 电平时为“1”。

位	符号	位名	说明	R/W
b0	MD	MD 引脚状态标志	0: MD 引脚为“Low 电平” 1: MD 引脚为“High 电平”	R
b7-b1	—	保留位	读取值为“0”，写操作无效。	R
b8	—	保留位	读取值为不定值，写操作无效。	R
b15-b9	—	保留位	读取值为“0”，写操作无效。	R

3.2.2 模式状态寄存器 (MDSR)

地址 0008 0002h

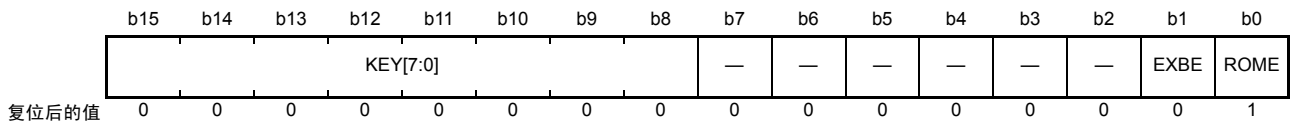
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	UBTS	BOTS	—	—	EXB	IROM
复位后的值	0	0	0	0	0	0	0	0	0	0	0/1 (注1)	0/1 (注1)	0	0	0	1

注 1. 因运行模式而不同。

位	符号	位名	说明	R/W
b0	IROM	内部 ROM 的启动状态标志	0: 启动时，内部 ROM 无效。 1: 启动时，内部 ROM 有效。	R
b1	EXB	外部总线的启动状态标志	0: 启动时，外部总线无效。 1: 启动时，外部总线有效。	R
b3-b2	—	保留位	读取值为“0”，写操作无效。	R
b4	BOTS	引导模式 (SCI 引导) 的启动标志	0: 非引导模式 (SCI 引导) 启动 1: 引导模式 (SCI 引导) 启动	R
b5	UBTS	用户引导模式的启动标志	0: 非用户引导模式启动 1: 用户引导模式启动	R
b15-b6	—	保留位	读取值为“0”，写操作无效。	R

3.2.3 系统控制寄存器 0 (SYSCR0)

地址 0008 0006h



位	符号	位名	说明	R/W
b0	ROME	内部 ROM 有效位	0: 内部 ROM 无效 1: 内部 ROM 有效	R/W
b1	EXBE	外部总线有效	0: 外部总线无效 1: 外部总线有效	R/W
b7-b2	—	保留位	读写值都为“0”。	R/W
b15-b8	KEY[7:0]	SYSCR0 键码	控制是否改写 EXBE 位和 ROME 位	R/W

ROME 位（内部 ROM 有效位）

此位选择内部 ROM（ROM 和 E2 数据闪存）的有效或者无效。

当 ROME 位为“1”时，能写“0”；当 ROME 位为“0”时，不能写“1”。

如果将内部 ROM 从有效改为无效，就不能使用 ROME 位将内部 ROM 置为有效。

不能在存取内部 ROM 时给 ROME 位写“0”。在给 ROME 位写“0”并且将内部 ROM 设定为无效后，必须先确认 ROME 位已被改写为“0”，然后再进行以后的处理。

EXBE 位（外部总线有效位）

此位选择外部总线的有效或者无效。

必须在未执行外部总线周期的状态下给 EXBE 位写“0”。在外部地址空间执行程序的过程中，不能给此位写“0”，

因为外部总线和内部总线有可能同时运行，因此必须注意外部总线是否被置为无效。要改写 EXBE 位时，必须在写寄存器后确认 EXBE 位已被改写，然后再进行总线存取。

需要同时设定 EXBE 位和 I/O 端口，详细内容请参照“20. 多功能引脚控制器（MPC）”。

KEY[7:0] 位（SYSCR0 键码）

这些位选择允许或者禁止写 SYSCR0 寄存器。

在给 ROME 位和 EXBE 位写值时，也必须同时给 KEY[7:0] 位写“5Ah”。如果给 KEY[7:0] 位写“5Ah”以外的值，即使写 SYSCR0 寄存器，ROME 位和 EXBE 位的值也不变。

3.2.4 系统控制寄存器 1 (SYSCR1)

地址 0008 0008h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RAME
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

复位后的值

位	符号	位名	说明	R/W
b0	RAME	RAM 有效位	0: 内部 RAM 无效 1: 内部 RAM 有效	R/W
b15-b1	—	保留位	读写值都为“0”。	R/W

RAME 位 (RAM 有效位)

此位选择内部 RAM 的有效或者无效。

一旦解除复位，RAME 位就被初始化为“1”。

不能在存取内部 RAM 时写“0”。在将内部 RAM 从无效改为有效后紧接着存取内部 RAM 时，必须在确认 RAME 位已被改写为“1”后进行存取。

即使将 RAME 位置“0”，也保持内部 RAM 的值，而且 RAM 待机电压 (VRAM) 也必须保持规定值。详细内容请参照“41. 电特性”。

3.3 运行模式的说明

3.3.1 单芯片模式

在此模式中，内部 ROM 有效或者无效而外部总线无效（SYSCR0.EXBE 位 =0）。全部 I/O 端口都能用作输入 / 输出端口。

启动时的内部 ROM 有效。当内部 ROM 有效（SYSCR0.ROME 位 =1）时，能将内部 ROM 设定为无效（SYSCR0.ROME 位 =0）；当内部 ROM 无效（SYSCR0.ROME 位 =0）时，不能将内部 ROM 设定为有效（SYSCR0.ROME 位 =1）。

如果将 SYSCR0.EXBE 位置“1”（外部总线有效），就能通过转移到内部 ROM 有效扩展模式或者内部 ROM 无效扩展模式使用外部总线。

3.3.2 内部 ROM 有效扩展模式

这是内部 ROM 有效（SYSCR0.ROME 位 =1）并且能使用外部总线的外部扩展模式（SYSCR0.EXBE 位 =1）。能将部分 I/O 端口用作数据总线的输入 / 输出、地址总线的输出和总线控制信号的输入 / 输出，详细内容请参照“20. 多功能引脚控制器（MPC）”。

能通过设定总线的外部总线宽度选择位（CSnCR.BSIZE[1:0] 位（n=0 ~ 3））更改外部总线宽度，详细内容请参照“15. 总线”。

能通过将 SYSCR0.EXBE 位置“0”（外部总线无效）转移到单芯片模式（内部 ROM 有效）。

能通过将 SYSCR0.ROME 位置“0”（内部 ROM 无效）转移到内部 ROM 无效扩展模式。

3.3.3 内部 ROM 无效扩展模式

这是内部 ROM 无效（SYSCR0.ROME 位 =0）并且能使用外部总线的外部扩展模式（SYSCR0.EXBE 位 =1）。能将部分 I/O 端口用作数据总线的输入 / 输出、地址总线的输出和总线控制信号的输入 / 输出，详细内容请参照“20. 多功能引脚控制器（MPC）”。

能通过设定总线的外部总线宽度选择位（CSnCR.BSIZE[1:0] 位（n=0 ~ 3））更改外部总线宽度，详细内容请参照“15. 总线”。

不能将内部 ROM 设定为有效（SYSCR0.ROME 位 =1）。

能通过将 SYSCR0.EXBE 位置“0”（外部总线无效）转移到单芯片模式（内部 ROM 无效）。

3.3.4 引导模式

这是内部 ROM 的引导模式（SCI 引导）。除了内部 ROM 的编程 / 擦除以外，和单芯片模式中的运行相同。详细内容请参照“39. ROM（保存代码的闪存）”和“40. E2 数据闪存（保存数据的闪存）”。

3.3.5 用户引导模式

这是执行用户引导模式的模式。除了内部 ROM 的编程 / 擦除以外，和单芯片模式中的运行相同。详细内容请参照“39. ROM（保存代码的闪存）”和“40. E2 数据闪存（保存数据的闪存）”。

在用户引导模式中不能转移到软件待机模式和深度软件待机模式。

3.4 运行模式的转移

3.4.1 通过模式引脚进行运行模式的转移

通过设定 MD 引脚进行运行模式的转移，如图 3.1 所示，能将运行模式转移到图中箭头所指的方向。

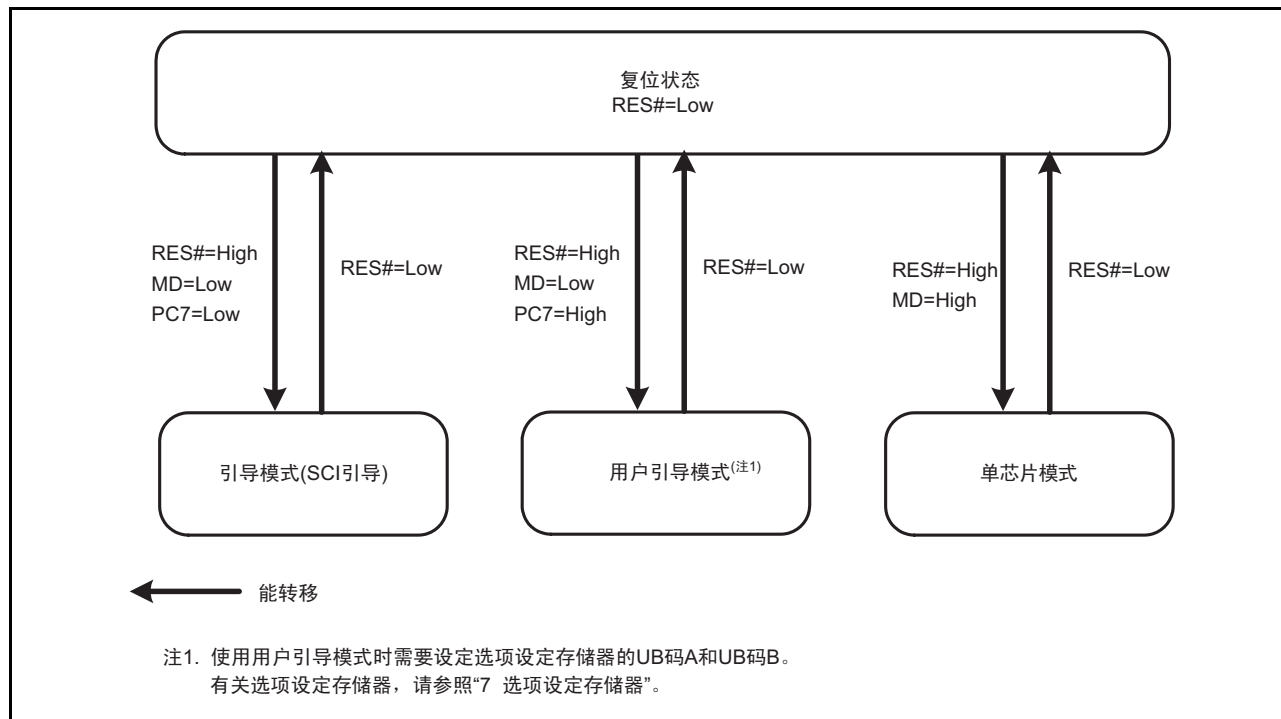


图 3.1 MD 引脚的设定和运行模式

3.4.2 通过寄存器进行运行模式的转移

通过设定 SYSCR0.ROME 位和 SYSCR0.EXBE 位进行运行模式的转移，如图 3.2 所示，能将运行模式转移到图中箭头所指的方向。

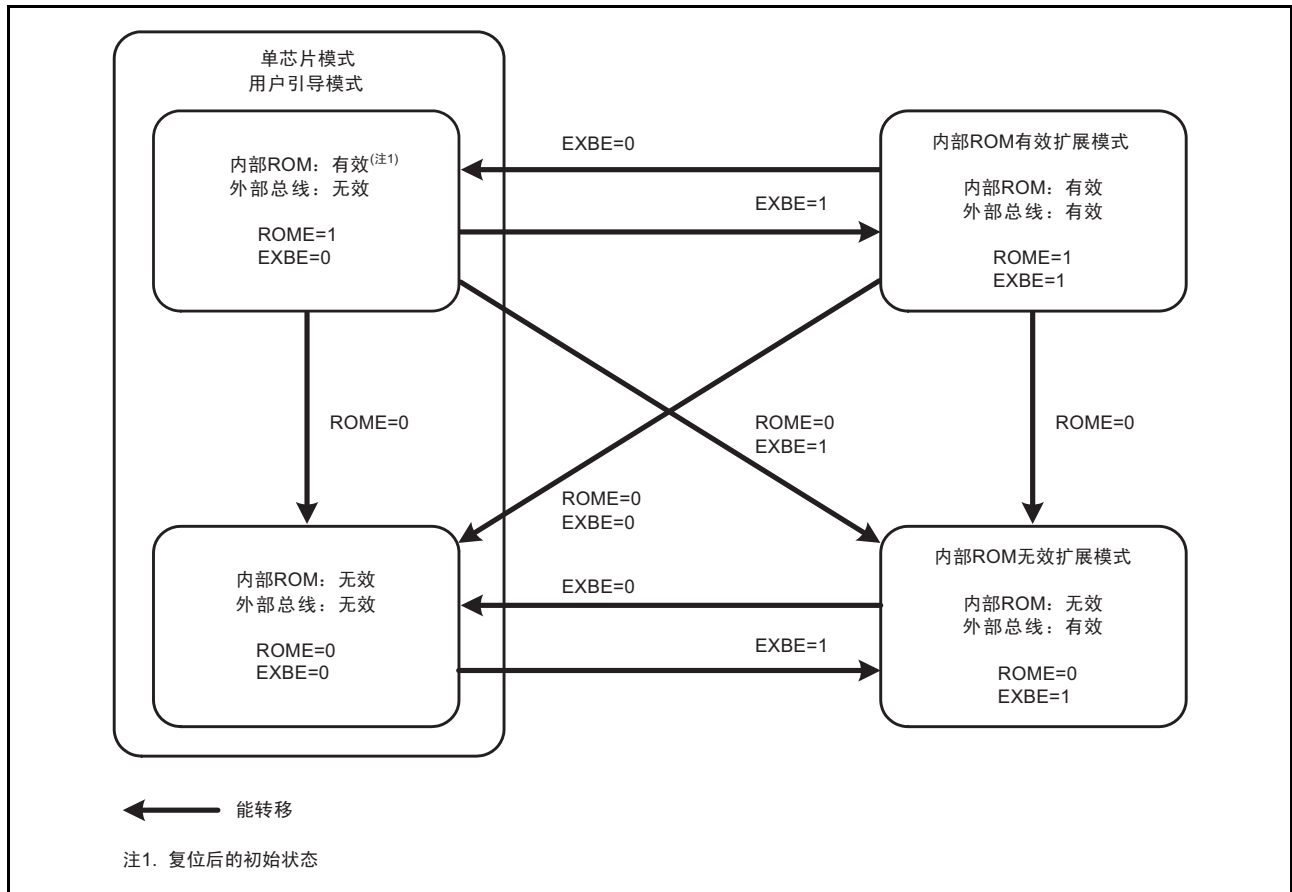


图 3.2 SYSCR0.ROME 位、SYSCR0.EXBE 位的设定和运行模式

4. 地址空间

4.1 地址空间

地址空间有地址 0000 0000h ~ 地址 FFFF FFFFh 的 4G 字节，能对程序区和数据区共计最多 4G 字节的空间进行线性存取。

各运行模式的存储器映像如图 4.1 所示，能存取的区域因运行模式和各控制位的状态而不同。

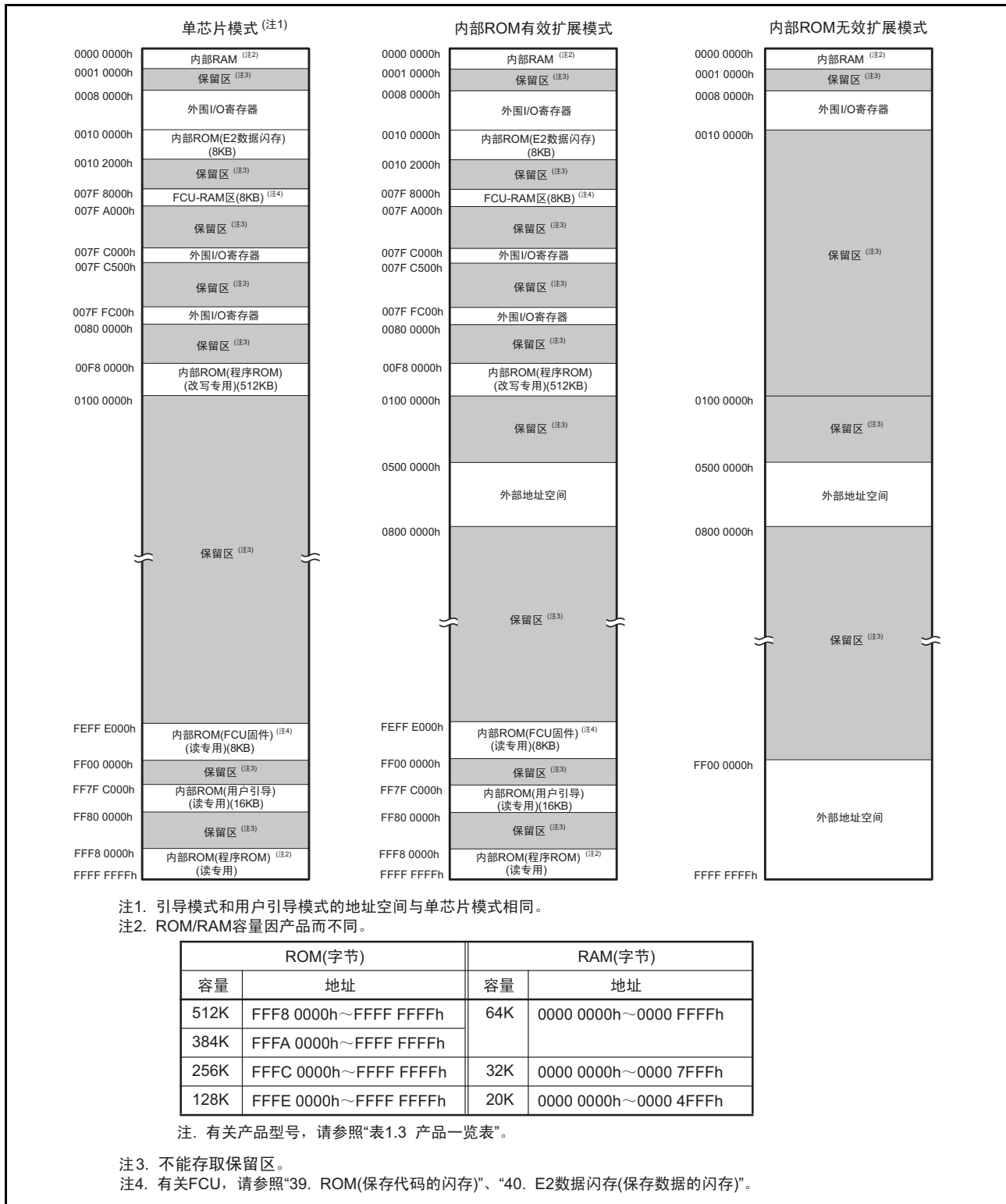


图 4.1 各运行模式的存储器映像

4.2 外部地址空间

外部地址空间通过 CSn# 引脚 (n=0 ~ 3) 输出的 CSn# 信号最多被分为 4 个 CS 区域 (CS0 ~ CS3)。内部 ROM 无效扩展模式中的 CS 区域 (CS0 ~ CS3) 和地址的对应如图 4.2 所示。

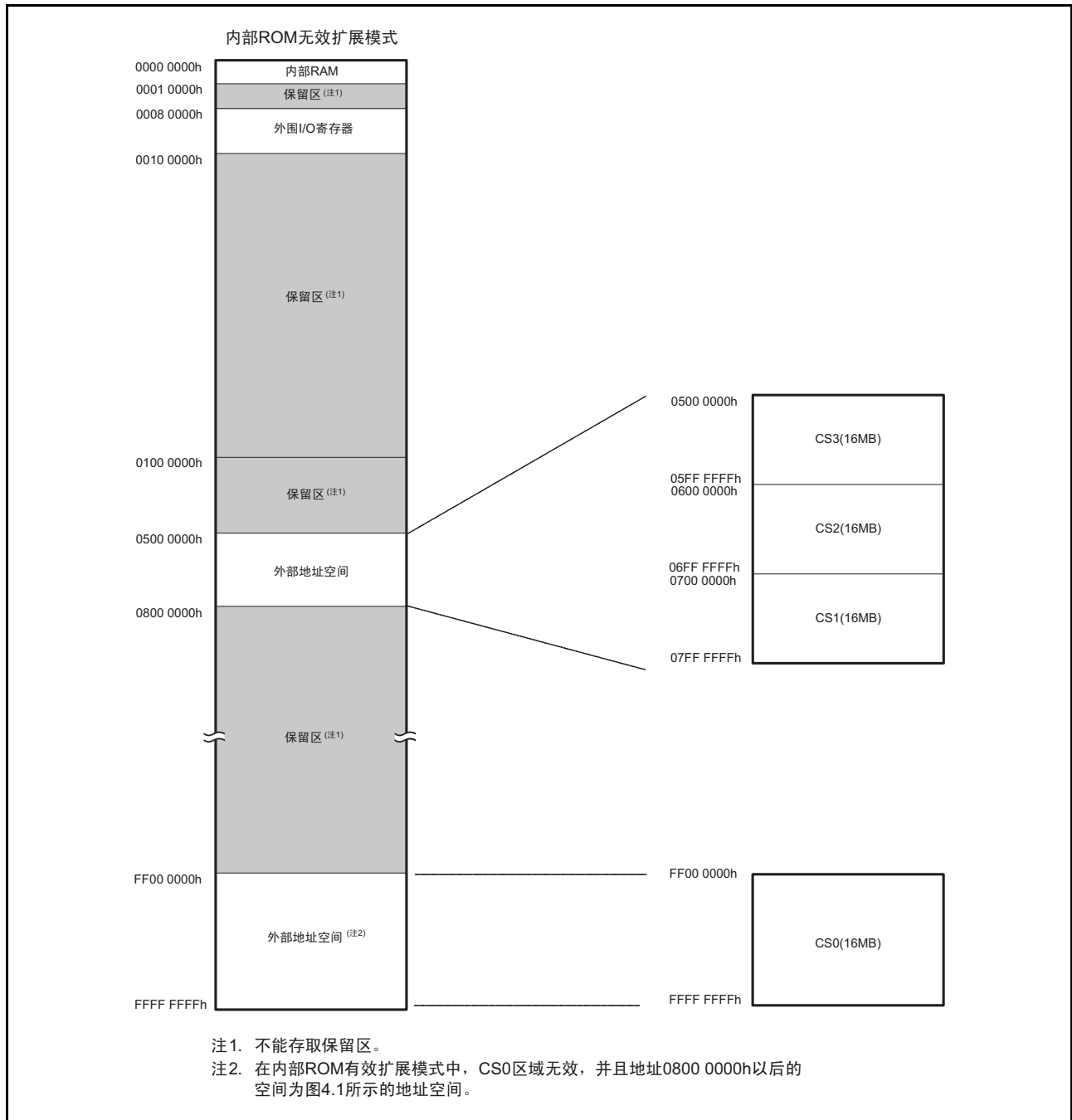


图 4.2 外部地址空间和 CS 区域 (内部 ROM 无效扩展模式的情况)

5. I/O 寄存器

I/O 寄存器一览表汇总了有关内部寄存器的地址和位结构的信息。表中的记述方法以及写寄存器时的注意事项如下所示。

(1) I/O 寄存器地址一览表（地址顺序）

- 从分配地址小的寄存器开始按顺序记载。
- 按模块符号进行分类。
- 存取状态数表示指定的基准时钟的状态数。
- 在I/O寄存器的区域，寄存器一览表中没有记载的地址区为保留区。禁止存取保留区。因为无法保证存取这些寄存器时的运行及其后续运行，所以禁止存取这些寄存器。

(2) 写 I/O 寄存器时的注意事项

在写 I/O 寄存器时，CPU 不等到写结束就执行后续指令。因此，在通过写 I/O 寄存器进行的设定变更被反映到运行前，有可能执行后续指令。

需要注意：在以下例子中，必须在 I/O 寄存器的设定变更被反映后执行后续指令。

[需要注意的运行例子]

- 将中断请求的允许位（ICU.IERm.IENj 位）置“0”，要在禁止中断的状态下执行后续指令时
- 在转移到低功耗状态的预处理后接着执行 WAIT 指令时

在这些情况下，必须在写 I/O 寄存器后按照以下步骤等到写结束，然后执行后续指令。

1. 写 I/O 寄存器。
2. 将所写的 I/O 寄存器的值读到通用寄存器。
3. 使用读取值进行运算。
4. 执行后续指令。

[指令例子]

- I/O 寄存器为字节的情况

```
MOV.L #SFR_ADDR,R1
MOV.B #SFR_DATA,[R1]
CMP [R1].UB, R1
;;以后的处理
```

- I/O 寄存器为字的情况

```
MOV.L #SFR_ADDR,R1
MOV.W #SFR_DATA,[R1]
CMP [R1].W,R1
;;以后的处理
```

- I/O 寄存器为长字的情况

```
MOV.L #SFR_ADDR,R1
MOV.L #SFR_DATA,[R1]
CMP [R1].L,R1
;;以后的处理
```

另外，要在写多个寄存器并且等到写结束后执行后续指令时，必须将最后写的 I/O 寄存器作为对象进行读操作和运算，而不需要将所写的全部寄存器作为对象进行读操作和运算。

(3) I/O 寄存器的存取周期数

有关 I/O 寄存器存取周期数，请参照“表 5.1 寄存器地址一览表”。

用以下表达式表示存取 I/O 寄存器时的存取周期数。(注 1)

$$\begin{aligned} \text{I/O 寄存器的存取周期数} = & \text{内部主总线 1 的总线周期数} + \\ & \text{分频时钟的同步周期数} + \\ & \text{内部外围总线 1} \sim \text{6 的总线周期数} \end{aligned}$$

内部外围总线 1 ~ 6 的总线周期数因存取的目标寄存器而不同。

在存取内部外围总线 2 ~ 6 连接的外围功能以及外部总线控制部的寄存器（总线错误的相关寄存器除外）时，增加分频时钟的同步周期。

分频时钟的同步周期数因 ICLK 和 PCLK（或者 FCLK、BCLK）的频率比以及总线的存取时序而不同。

在外围功能部中，如果 $\text{ICLK} \geq \text{PCLK}$ （或者 FCLK），内部主总线 1 的总线周期数加上分频时钟的同步周期数最多就为 1 个 PCLK（或者 FCLK）周期，因此在表 5.1 中记载为 1 个 PCLK（或者 FCLK）的宽度；如果 $\text{ICLK} < \text{PCLK}$ （或者 FCLK），就在外围功能结束后的第一个 ICLK 周期开始下一个总线存取，因此在表 5.1 中以 ICLK 为单位记载。

在外部总线控制部中，因为内部主总线 1 的总线周期数加上分频时钟的同步周期数最多为 1 个 BCLK 周期，所以在表 5.1 中记载为 1 个 BCLK 的宽度。

注 1. 这是 CPU 存取寄存器时，在和外部存储器的取指令或者不同总线主控器（DMAC、DTC）的总线存取不发生竞争的情况下执行的周期数。

5.1 I/O 寄存器地址一览表（地址顺序）

表 5.1 I/O 寄存器地址一览表 (1/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									ICLK ≥ PCLK 时	ICLK < PCLK 时	
○	○	○	0008 0000h	SYSTEM	模式监视寄存器	MDMONR	16	16	3 个 ICLK		第 3 章
○	○	○	0008 0002h	SYSTEM	模式状态寄存器	MDSR	16	16	3 个 ICLK		
○	○	○	0008 0006h	SYSTEM	系统控制寄存器 0	SYSCR0	16	16	3 个 ICLK		
○	○	○	0008 0008h	SYSTEM	系统控制寄存器 1	SYSCR1	16	16	3 个 ICLK		
○	○	○	0008 000Ch	SYSTEM	待机控制寄存器	SBYCR	16	16	3 个 ICLK		第 11 章
○	○	○	0008 0010h	SYSTEM	模块停止控制寄存器 A	MSTPCRA	32	32	3 个 ICLK		
○	○	○	0008 0014h	SYSTEM	模块停止控制寄存器 B	MSTPCRB	32	32	3 个 ICLK		
○	○	○	0008 0018h	SYSTEM	模块停止控制寄存器 C	MSTPCRC	32	32	3 个 ICLK		
○	○	○	0008 0020h	SYSTEM	系统时钟控制寄存器	SCKCR	32	32	3 个 ICLK		第 9 章
○	○	○	0008 0026h	SYSTEM	系统时钟控制寄存器 3	SCKCR3	16	16	3 个 ICLK		
○	○	○	0008 0028h	SYSTEM	PLL 控制寄存器	PLLCR	16	16	3 个 ICLK		
○	○	○	0008 002Ah	SYSTEM	PLL 控制寄存器 2	PLLCR2	8	8	3 个 ICLK		
○	○	○	0008 0030h	SYSTEM	外部总线时钟控制寄存器	BCKCR	8	8	3 个 ICLK		第 11 章
○	○	○	0008 0032h	SYSTEM	主时钟振荡器的控制寄存器	MOSCCR	8	8	3 个 ICLK		
○	○	○	0008 0033h	SYSTEM	副时钟振荡器的控制寄存器	SOSCCR	8	8	3 个 ICLK		
○	○	○	0008 0034h	SYSTEM	低速内部振荡器的控制寄存器	LOCOCR	8	8	3 个 ICLK		
○	○	○	0008 0035h	SYSTEM	IWDT 专用内部振荡器的控制寄存器	ILOCOCR	8	8	3 个 ICLK		
○	○	○	0008 0036h	SYSTEM	高速内部振荡器的控制寄存器	HOCOCR	8	8	3 个 ICLK		
○	○	○	0008 0037h	SYSTEM	高速内部振荡器的控制寄存器 2	HOCOCR2	8	8	3 个 ICLK		
○	○	○	0008 0040h	SYSTEM	振荡停止检测控制寄存器	OSTDCR	8	8	3 个 ICLK		
○	○	○	0008 0041h	SYSTEM	振荡停止检测状态寄存器	OSTDSR	8	8	3 个 ICLK		
○	○	○	0008 00A0h	SYSTEM	运行功率控制寄存器	OPCCR	8	8	3 个 ICLK		
○	○	○	0008 00A1h	SYSTEM	睡眠模式返回时钟源转换寄存器	RSTCKCR	8	8	3 个 ICLK		
○	○	○	0008 00A2h	SYSTEM	主时钟振荡器的等待控制寄存器	MOSCWTCR	8	8	3 个 ICLK		
○	○	○	0008 00A3h	SYSTEM	副时钟振荡器的等待控制寄存器	SOSCWTCR	8	8	3 个 ICLK		
○	○	○	0008 00A6h	SYSTEM	PLL 等待控制寄存器	PLLWTCR	8	8	3 个 ICLK		
○	○	○	0008 00A9h	SYSTEM	HOCO 等待控制寄存器 2	HOCOWTCR2	8	8	3 个 ICLK		
○	○	○	0008 00C0h	SYSTEM	复位状态寄存器 2	RSTSR2	8	8	3 个 ICLK		第 6 章
○	○	○	0008 00C2h	SYSTEM	软件复位寄存器	SWRR	16	16	3 个 ICLK		
○	○	○	0008 00E0h	SYSTEM	电压监视 1 电路 / 比较器 A1 的控制寄存器 1	LVD1CR1	8	8	3 个 ICLK		第 8 章、 35 章
○	○	○	0008 00E1h	SYSTEM	电压监视 1 电路 / 比较器 A1 的状态寄存器	LVD1SR	8	8	3 个 ICLK		
○	○	○	0008 00E2h	SYSTEM	电压监视 2 电路 / 比较器 A2 的控制寄存器 1	LVD2CR1	8	8	3 个 ICLK		
○	○	○	0008 00E3h	SYSTEM	电压监视 2 电路 / 比较器 A2 的状态寄存器	LVD2SR	8	8	3 个 ICLK		
○	○	○	0008 03FEh	SYSTEM	保护寄存器	PRCR	16	16	3 个 ICLK		第 12 章
○	○	○	0008 1300h	BSC	总线错误状态清除寄存器	BERCLR	8	8	2 个 ICLK		第 15 章
○	○	○	0008 1304h	BSC	总线错误监视允许寄存器	BEREN	8	8	2 个 ICLK		
○	○	○	0008 1308h	BSC	总线错误状态寄存器 1	BERSR1	8	8	2 个 ICLK		
○	○	○	0008 130Ah	BSC	总线错误状态寄存器 2	BERSR2	16	16	2 个 ICLK		
○	○	○	0008 1310h	BSC	总线优先级控制寄存器	BUSPRI	16	16	2 个 ICLK		
○	○	○	0008 2000h	DMAC0	DAM 错误源地址寄存器	DMSAR	32	32	2 个 ICLK		第 16 章
○	○	○	0008 2004h	DMAC0	DAM 传送目标地址寄存器	DMDAR	32	32	2 个 ICLK		
○	○	○	0008 2008h	DMAC0	DMA 传送计数寄存器	DMCRA	32	32	2 个 ICLK		
○	○	○	0008 200Ch	DMAC0	DMA 块传送计数寄存器	DMCRB	16	16	2 个 ICLK		
○	○	○	0008 2010h	DMAC0	DMA 传送模式寄存器	DMTMD	16	16	2 个 ICLK		
○	○	○	0008 2013h	DMAC0	DMA 中断设定寄存器	DMINT	8	8	2 个 ICLK		
○	○	○	0008 2014h	DMAC0	DMA 地址模式寄存器	DMAMD	16	16	2 个 ICLK		
○	○	○	0008 2018h	DMAC0	DMA 偏移寄存器	DMOFR	32	32	2 个 ICLK		
○	○	○	0008 201Ch	DMAC0	DMA 传送允许寄存器	DMCNT	8	8	2 个 ICLK		
○	○	○	0008 201Dh	DMAC0	DMA 软件启动寄存器	DMREQ	8	8	2 个 ICLK		

表 5.1 I/O 寄存器地址一览表 (2/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									ICLK ≥ PCLK 时	ICLK < PCLK 时	
○	○	○	0008 201Eh	DMAC0	DMA 状态寄存器	DMSTS	8	8	2 个 ICLK		第 16 章
○	○	○	0008 201Fh	DMAC0	DMA 启动源标志控制寄存器	DMCSL	8	8	2 个 ICLK		
○	○	○	0008 2040h	DMAC1	DMA 传送源地址寄存器	DMSAR	32	32	2 个 ICLK		
○	○	○	0008 2044h	DMAC1	DMA 传送目标地址寄存器	DMDAR	32	32	2 个 ICLK		
○	○	○	0008 2048h	DMAC1	DMA 传送计数寄存器	DMCRA	32	32	2 个 ICLK		
○	○	○	0008 204Ch	DMAC1	DMA 块传送计数寄存器	DMCRB	16	16	2 个 ICLK		
○	○	○	0008 2050h	DMAC1	DMA 传送模式寄存器	DMTMD	16	16	2 个 ICLK		
○	○	○	0008 2053h	DMAC1	DMA 中断设定寄存器	DMINT	8	8	2 个 ICLK		
○	○	○	0008 2054h	DMAC1	DMA 地址模式寄存器	DMAMD	16	16	2 个 ICLK		
○	○	○	0008 205Ch	DMAC1	DMA 传送允许寄存器	DMCNT	8	8	2 个 ICLK		
○	○	○	0008 205Dh	DMAC1	DMA 软件启动寄存器	DMREQ	8	8	2 个 ICLK		
○	○	○	0008 205Eh	DMAC1	DMA 状态寄存器	DMSTS	8	8	2 个 ICLK		
○	○	○	0008 205Fh	DMAC1	DMA 启动源标志控制寄存器	DMCSL	8	8	2 个 ICLK		
○	○	○	0008 2080h	DMAC2	DMA 传送源地址寄存器	DMSAR	32	32	2 个 ICLK		
○	○	○	0008 2084h	DMAC2	DMA 传送目标地址寄存器	DMDAR	32	32	2 个 ICLK		
○	○	○	0008 2088h	DMAC2	DMA 传送计数寄存器	DMCRA	32	32	2 个 ICLK		
○	○	○	0008 208Ch	DMAC2	DMA 块传送计数寄存器	DMCRB	16	16	2 个 ICLK		
○	○	○	0008 2090h	DMAC2	DMA 传送模式寄存器	DMTMD	16	16	2 个 ICLK		
○	○	○	0008 2093h	DMAC2	DMA 中断设定寄存器	DMINT	8	8	2 个 ICLK		
○	○	○	0008 2094h	DMAC2	DMA 地址模式寄存器	DMAMD	16	16	2 个 ICLK		
○	○	○	0008 209Ch	DMAC2	DMA 传送允许寄存器	DMCNT	8	8	2 个 ICLK		
○	○	○	0008 209Dh	DMAC2	DMA 软件启动寄存器	DMREQ	8	8	2 个 ICLK		
○	○	○	0008 209Eh	DMAC2	DMA 状态寄存器	DMSTS	8	8	2 个 ICLK		
○	○	○	0008 209Fh	DMAC2	DMA 启动源标志控制寄存器	DMCSL	8	8	2 个 ICLK		
○	○	○	0008 20C0h	DMAC3	DMA 传送源地址寄存器	DMSAR	32	32	2 个 ICLK		
○	○	○	0008 20C4h	DMAC3	DMA 传送目标地址寄存器	DMDAR	32	32	2 个 ICLK		
○	○	○	0008 20C8h	DMAC3	DMA 传送计数寄存器	DMCRA	32	32	2 个 ICLK		
○	○	○	0008 20CCh	DMAC3	DMA 块传送计数寄存器	DMCRB	16	16	2 个 ICLK		
○	○	○	0008 20D0h	DMAC3	DMA 传送模式寄存器	DMTMD	16	16	2 个 ICLK		
○	○	○	0008 20D3h	DMAC3	DMA 中断设定寄存器	DMINT	8	8	2 个 ICLK		
○	○	○	0008 20D4h	DMAC3	DMA 地址模式寄存器	DMAMD	16	16	2 个 ICLK		
○	○	○	0008 20DCh	DMAC3	DMA 传送允许寄存器	DMCNT	8	8	2 个 ICLK		
○	○	○	0008 20DDh	DMAC3	DMA 软件启动寄存器	DMREQ	8	8	2 个 ICLK		
○	○	○	0008 20DEh	DMAC3	DMA 状态寄存器	DMSTS	8	8	2 个 ICLK		
○	○	○	0008 20DFh	DMAC3	DMA 启动源标志控制寄存器	DMCSL	8	8	2 个 ICLK		
○	○	○	0008 2200h	DMAC	DMA 模块启动寄存器	DMAST	8	8	2 个 ICLK		
○	○	○	0008 2400h	DTC	DTC 控制寄存器	DTCCR	8	8	2 个 ICLK		第 17 章
○	○	○	0008 2404h	DTC	DTC 向量基址寄存器	DTCVBR	32	32	2 个 ICLK		
○	○	○	0008 2408h	DTC	DTC 地址模式寄存器	DTCADMOD	8	8	2 个 ICLK		
○	○	○	0008 240Ch	DTC	DTC 模块启动寄存器	DTCST	8	8	2 个 ICLK		
○	○	○	0008 240Eh	DTC	DTC 状态寄存器	DTCSTS	16	16	2 个 ICLK		
○	—	—	0008 3002h	BSC	CS0 模式寄存器	CS0MOD	16	16	1 ~ 2 个 BCLK		第 15 章
○	—	—	0008 3004h	BSC	CS0 等待控制寄存器 1	CS0WCR1	32	32	1 ~ 2 个 BCLK		
○	—	—	0008 3008h	BSC	CS0 等待控制寄存器 2	CS0WCR2	32	32	1 ~ 2 个 BCLK		
○	—	—	0008 3012h	BSC	CS1 模式寄存器	CS1MOD	16	16	1 ~ 2 个 BCLK		
○	—	—	0008 3014h	BSC	CS1 等待控制寄存器 1	CS1WCR1	32	32	1 ~ 2 个 BCLK		
○	—	—	0008 3018h	BSC	CS1 等待控制寄存器 2	CS1WCR2	32	32	1 ~ 2 个 BCLK		
○	—	—	0008 3022h	BSC	CS2 模式寄存器	CS2MOD	16	16	1 ~ 2 个 BCLK		
○	—	—	0008 3024h	BSC	CS2 等待控制寄存器 1	CS2WCR1	32	32	1 ~ 2 个 BCLK		
○	—	—	0008 3028h	BSC	CS2 等待控制寄存器 2	CS2WCR2	32	32	1 ~ 2 个 BCLK		
○	—	—	0008 3032h	BSC	CS3 模式寄存器	CS3MOD	16	16	1 ~ 2 个 BCLK		

表 5.1 I/O 寄存器地址一览表 (3/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									$ICLK \geq PCLK$ 时	$ICLK < PCLK$ 时	
○	—	—	0008 3034h	BSC	CS3 等待控制寄存器 1	CS3WCR1	32	32	1 ~ 2 个 BCLK		第 15 章
○	—	—	0008 3038h	BSC	CS3 等待控制寄存器 2	CS3WCR2	32	32	1 ~ 2 个 BCLK		
○	—	—	0008 3802h	BSC	CS0 控制寄存器	CS0CR	16	16	1 ~ 2 个 BCLK		
○	—	—	0008 380Ah	BSC	CS0 恢复周期设定寄存器	CS0REC	16	16	1 ~ 2 个 BCLK		
○	—	—	0008 3812h	BSC	CS1 控制寄存器	CS1CR	16	16	1 ~ 2 个 BCLK		
○	—	—	0008 381Ah	BSC	CS1 恢复周期设定寄存器	CS1REC	16	16	1 ~ 2 个 BCLK		
○	—	—	0008 3822h	BSC	CS2 控制寄存器	CS2CR	16	16	1 ~ 2 个 BCLK		
○	—	—	0008 382Ah	BSC	CS2 恢复周期设定寄存器	CS2REC	16	16	1 ~ 2 个 BCLK		
○	—	—	0008 3832h	BSC	CS3 控制寄存器	CS3CR	16	16	1 ~ 2 个 BCLK		
○	—	—	0008 383Ah	BSC	CS3 恢复周期设定寄存器	CS3REC	16	16	1 ~ 2 个 BCLK		
○	—	—	0008 3880h	BSC	CS 恢复周期插入允许寄存器	CSRECEN	16	16	1 ~ 2 个 BCLK		第 14 章
○	○	○	0008 7010h	ICU	中断请求寄存器 016	IR016	8	8	2 个 ICLK		
○	○	○	0008 7015h	ICU	中断请求寄存器 021	IR021	8	8	2 个 ICLK		
○	○	○	0008 7017h	ICU	中断请求寄存器 023	IR023	8	8	2 个 ICLK		
○	○	○	0008 701Bh	ICU	中断请求寄存器 027	IR027	8	8	2 个 ICLK		
○	○	○	0008 701Ch	ICU	中断请求寄存器 028	IR028	8	8	2 个 ICLK		
○	○	○	0008 701Dh	ICU	中断请求寄存器 029	IR029	8	8	2 个 ICLK		
○	○	○	0008 701Eh	ICU	中断请求寄存器 030	IR030	8	8	2 个 ICLK		
○	○	○	0008 701Fh	ICU	中断请求寄存器 031	IR031	8	8	2 个 ICLK		
○	○	○	0008 7020h	ICU	中断请求寄存器 032	IR032	8	8	2 个 ICLK		
○	○	○	0008 7021h	ICU	中断请求寄存器 033	IR033	8	8	2 个 ICLK		
○	○	○	0008 7022h	ICU	中断请求寄存器 034	IR034	8	8	2 个 ICLK		
○	○	○	0008 702Ch	ICU	中断请求寄存器 044	IR044	8	8	2 个 ICLK		
○	○	○	0008 702Dh	ICU	中断请求寄存器 045	IR045	8	8	2 个 ICLK		
○	○	○	0008 702Eh	ICU	中断请求寄存器 046	IR046	8	8	2 个 ICLK		
○	○	○	0008 702Fh	ICU	中断请求寄存器 047	IR047	8	8	2 个 ICLK		
○	○	○	0008 7039h	ICU	中断请求寄存器 057	IR057	8	8	2 个 ICLK		
○	○	○	0008 703Ah	ICU	中断请求寄存器 058	IR058	8	8	2 个 ICLK		
○	○	○	0008 703Bh	ICU	中断请求寄存器 059	IR059	8	8	2 个 ICLK		
○	○	○	0008 703Fh	ICU	中断请求寄存器 063	IR063	8	8	2 个 ICLK		
○	○	○	0008 7040h	ICU	中断请求寄存器 064	IR064	8	8	2 个 ICLK		
○	○	○	0008 7041h	ICU	中断请求寄存器 065	IR065	8	8	2 个 ICLK		
○	○	○	0008 7042h	ICU	中断请求寄存器 066	IR066	8	8	2 个 ICLK		
○	○	○	0008 7043h	ICU	中断请求寄存器 067	IR067	8	8	2 个 ICLK		
○	○	○	0008 7044h	ICU	中断请求寄存器 068	IR068	8	8	2 个 ICLK		
○	○	○	0008 7045h	ICU	中断请求寄存器 069	IR069	8	8	2 个 ICLK		
○	○	○	0008 7046h	ICU	中断请求寄存器 070	IR070	8	8	2 个 ICLK		
○	○	○	0008 7047h	ICU	中断请求寄存器 071	IR071	8	8	2 个 ICLK		
○	○	○	0008 7058h	ICU	中断请求寄存器 088	IR088	8	8	2 个 ICLK		
○	○	○	0008 7059h	ICU	中断请求寄存器 089	IR089	8	8	2 个 ICLK		
○	○	○	0008 705Ch	ICU	中断请求寄存器 092	IR092	8	8	2 个 ICLK		
○	○	○	0008 705Dh	ICU	中断请求寄存器 093	IR093	8	8	2 个 ICLK		
○	○	○	0008 7066h	ICU	中断请求寄存器 102	IR102	8	8	2 个 ICLK		
○	○	○	0008 7067h	ICU	中断请求寄存器 103	IR103	8	8	2 个 ICLK		
○	○	○	0008 706Ah	ICU	中断请求寄存器 106	IR106	8	8	2 个 ICLK		
○	○	○	0008 706Bh	ICU	中断请求寄存器 107	IR107	8	8	2 个 ICLK		
○	○	○	0008 7072h	ICU	中断请求寄存器 114	IR114	8	8	2 个 ICLK		
○	○	○	0008 7073h	ICU	中断请求寄存器 115	IR115	8	8	2 个 ICLK		
○	○	○	0008 7074h	ICU	中断请求寄存器 116	IR116	8	8	2 个 ICLK		
○	○	○	0008 7075h	ICU	中断请求寄存器 117	IR117	8	8	2 个 ICLK		
○	○	○	0008 7076h	ICU	中断请求寄存器 118	IR118	8	8	2 个 ICLK		

表 5.1 I/O 寄存器地址一览表 (4/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									ICLK ≥ PCLK 时	ICLK < PCLK 时	
○	○	○	0008 7077h	ICU	中断请求寄存器 119	IR119	8	8	2 个 ICLK		第 14 章
○	○	○	0008 7078h	ICU	中断请求寄存器 120	IR120	8	8	2 个 ICLK		
○	○	○	0008 7079h	ICU	中断请求寄存器 121	IR121	8	8	2 个 ICLK		
○	○	○	0008 707Ah	ICU	中断请求寄存器 122	IR122	8	8	2 个 ICLK		
○	○	○	0008 707Bh	ICU	中断请求寄存器 123	IR123	8	8	2 个 ICLK		
○	○	○	0008 707Ch	ICU	中断请求寄存器 124	IR124	8	8	2 个 ICLK		
○	○	○	0008 707Dh	ICU	中断请求寄存器 125	IR125	8	8	2 个 ICLK		
○	○	○	0008 707Eh	ICU	中断请求寄存器 126	IR126	8	8	2 个 ICLK		
○	○	○	0008 707Fh	ICU	中断请求寄存器 127	IR127	8	8	2 个 ICLK		
○	○	○	0008 7080h	ICU	中断请求寄存器 128	IR128	8	8	2 个 ICLK		
○	○	○	0008 7081h	ICU	中断请求寄存器 129	IR129	8	8	2 个 ICLK		
○	○	○	0008 7082h	ICU	中断请求寄存器 130	IR130	8	8	2 个 ICLK		
○	○	○	0008 7083h	ICU	中断请求寄存器 131	IR131	8	8	2 个 ICLK		
○	○	○	0008 7084h	ICU	中断请求寄存器 132	IR132	8	8	2 个 ICLK		
○	○	○	0008 7085h	ICU	中断请求寄存器 133	IR133	8	8	2 个 ICLK		
○	○	○	0008 7086h	ICU	中断请求寄存器 134	IR134	8	8	2 个 ICLK		
○	○	○	0008 7087h	ICU	中断请求寄存器 135	IR135	8	8	2 个 ICLK		
○	○	○	0008 7088h	ICU	中断请求寄存器 136	IR136	8	8	2 个 ICLK		
○	○	○	0008 7089h	ICU	中断请求寄存器 137	IR137	8	8	2 个 ICLK		
○	○	○	0008 708Ah	ICU	中断请求寄存器 138	IR138	8	8	2 个 ICLK		
○	○	○	0008 708Bh	ICU	中断请求寄存器 139	IR139	8	8	2 个 ICLK		
○	○	○	0008 708Ch	ICU	中断请求寄存器 140	IR140	8	8	2 个 ICLK		
○	○	○	0008 708Dh	ICU	中断请求寄存器 141	IR141	8	8	2 个 ICLK		
○	○	○	0008 70AAh	ICU	中断请求寄存器 170	IR170	8	8	2 个 ICLK		
○	○	○	0008 70ABh	ICU	中断请求寄存器 171	IR171	8	8	2 个 ICLK		
○	○	○	0008 70AEh	ICU	中断请求寄存器 174	IR174	8	8	2 个 ICLK		
○	○	○	0008 70AFh	ICU	中断请求寄存器 175	IR175	8	8	2 个 ICLK		
○	○	○	0008 70B0h	ICU	中断请求寄存器 176	IR176	8	8	2 个 ICLK		
○	○	○	0008 70B1h	ICU	中断请求寄存器 177	IR177	8	8	2 个 ICLK		
○	○	○	0008 70B2h	ICU	中断请求寄存器 178	IR178	8	8	2 个 ICLK		
○	○	○	0008 70B3h	ICU	中断请求寄存器 179	IR179	8	8	2 个 ICLK		
○	○	○	0008 70B4h	ICU	中断请求寄存器 180	IR180	8	8	2 个 ICLK		
○	○	○	0008 70B5h	ICU	中断请求寄存器 181	IR181	8	8	2 个 ICLK		
○	○	○	0008 70B6h	ICU	中断请求寄存器 182	IR182	8	8	2 个 ICLK		
○	○	○	0008 70B7h	ICU	中断请求寄存器 183	IR183	8	8	2 个 ICLK		
○	○	○	0008 70B8h	ICU	中断请求寄存器 184	IR184	8	8	2 个 ICLK		
○	○	○	0008 70B9h	ICU	中断请求寄存器 185	IR185	8	8	2 个 ICLK		
○	○	○	0008 70C6h	ICU	中断请求寄存器 198	IR198	8	8	2 个 ICLK		
○	○	○	0008 70C7h	ICU	中断请求寄存器 199	IR199	8	8	2 个 ICLK		
○	○	○	0008 70C8h	ICU	中断请求寄存器 200	IR200	8	8	2 个 ICLK		
○	○	○	0008 70C9h	ICU	中断请求寄存器 201	IR201	8	8	2 个 ICLK		
○	○	—	0008 70D6h	ICU	中断请求寄存器 214	IR214	8	8	2 个 ICLK		
○	○	—	0008 70D7h	ICU	中断请求寄存器 215	IR215	8	8	2 个 ICLK		
○	○	—	0008 70D8h	ICU	中断请求寄存器 216	IR216	8	8	2 个 ICLK		
○	○	—	0008 70D9h	ICU	中断请求寄存器 217	IR217	8	8	2 个 ICLK		
○	○	○	0008 70DAh	ICU	中断请求寄存器 218	IR218	8	8	2 个 ICLK		
○	○	○	0008 70DBh	ICU	中断请求寄存器 219	IR219	8	8	2 个 ICLK		
○	○	○	0008 70DCh	ICU	中断请求寄存器 220	IR220	8	8	2 个 ICLK		
○	○	○	0008 70DDh	ICU	中断请求寄存器 221	IR221	8	8	2 个 ICLK		
○	○	○	0008 70DEh	ICU	中断请求寄存器 222	IR222	8	8	2 个 ICLK		
○	○	○	0008 70DFh	ICU	中断请求寄存器 223	IR223	8	8	2 个 ICLK		

表 5.1 I/O 寄存器地址一览表 (5/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									ICLK \geq PCLK 时	ICLK $<$ PCLK 时	
			0008 70E0h	ICU	中断请求寄存器 224	IR224	8	8	2 个 ICLK		第 14 章
			0008 70E1h	ICU	中断请求寄存器 225	IR225	8	8	2 个 ICLK		
			0008 70E2h	ICU	中断请求寄存器 226	IR226	8	8	2 个 ICLK		
			0008 70E3h	ICU	中断请求寄存器 227	IR227	8	8	2 个 ICLK		
			0008 70E4h	ICU	中断请求寄存器 228	IR228	8	8	2 个 ICLK		
			0008 70E5h	ICU	中断请求寄存器 229	IR229	8	8	2 个 ICLK		
			0008 70E6h	ICU	中断请求寄存器 230	IR230	8	8	2 个 ICLK		
			0008 70E7h	ICU	中断请求寄存器 231	IR231	8	8	2 个 ICLK		
			0008 70E8h	ICU	中断请求寄存器 232	IR232	8	8	2 个 ICLK		
			0008 70E9h	ICU	中断请求寄存器 233	IR233	8	8	2 个 ICLK		
			0008 70EAh	ICU	中断请求寄存器 234	IR234	8	8	2 个 ICLK		
			0008 70EBh	ICU	中断请求寄存器 235	IR235	8	8	2 个 ICLK		
			0008 70ECh	ICU	中断请求寄存器 236	IR236	8	8	2 个 ICLK		
			0008 70EDh	ICU	中断请求寄存器 237	IR237	8	8	2 个 ICLK		
			0008 70EEh	ICU	中断请求寄存器 238	IR238	8	8	2 个 ICLK		
			0008 70EFh	ICU	中断请求寄存器 239	IR239	8	8	2 个 ICLK		
			0008 70F0h	ICU	中断请求寄存器 240	IR240	8	8	2 个 ICLK		
			0008 70F1h	ICU	中断请求寄存器 241	IR241	8	8	2 个 ICLK		
			0008 70F2h	ICU	中断请求寄存器 242	IR242	8	8	2 个 ICLK		
			0008 70F3h	ICU	中断请求寄存器 243	IR243	8	8	2 个 ICLK		
			0008 70F4h	ICU	中断请求寄存器 244	IR244	8	8	2 个 ICLK		
			0008 70F5h	ICU	中断请求寄存器 245	IR245	8	8	2 个 ICLK		
			0008 70F6h	ICU	中断请求寄存器 246	IR246	8	8	2 个 ICLK		
			0008 70F7h	ICU	中断请求寄存器 247	IR247	8	8	2 个 ICLK		
			0008 70F8h	ICU	中断请求寄存器 248	IR248	8	8	2 个 ICLK		
			0008 70F9h	ICU	中断请求寄存器 249	IR249	8	8	2 个 ICLK		
			0008 711Bh	ICU	DTC 启动允许寄存器 027	DT CER027	8	8	2 个 ICLK		
			0008 711Ch	ICU	DTC 启动允许寄存器 028	DT CER028	8	8	2 个 ICLK		
			0008 711Dh	ICU	DTC 启动允许寄存器 029	DT CER029	8	8	2 个 ICLK		
			0008 711Eh	ICU	DTC 启动允许寄存器 030	DT CER030	8	8	2 个 ICLK		
			0008 711Fh	ICU	DTC 启动允许寄存器 031	DT CER031	8	8	2 个 ICLK		
			0008 712Dh	ICU	DTC 启动允许寄存器 045	DT CER045	8	8	2 个 ICLK		
			0008 712Eh	ICU	DTC 启动允许寄存器 046	DT CER046	8	8	2 个 ICLK		
			0008 713Ah	ICU	DTC 启动允许寄存器 058	DT CER058	8	8	2 个 ICLK		
			0008 713Bh	ICU	DTC 启动允许寄存器 059	DT CER059	8	8	2 个 ICLK		
			0008 7140h	ICU	DTC 启动允许寄存器 064	DT CER064	8	8	2 个 ICLK		
			0008 7141h	ICU	DTC 启动允许寄存器 065	DT CER065	8	8	2 个 ICLK		
			0008 7142h	ICU	DTC 启动允许寄存器 066	DT CER066	8	8	2 个 ICLK		
			0008 7143h	ICU	DTC 启动允许寄存器 067	DT CER067	8	8	2 个 ICLK		
			0008 7144h	ICU	DTC 启动允许寄存器 068	DT CER068	8	8	2 个 ICLK		
			0008 7145h	ICU	DTC 启动允许寄存器 069	DT CER069	8	8	2 个 ICLK		
			0008 7146h	ICU	DTC 启动允许寄存器 070	DT CER070	8	8	2 个 ICLK		
			0008 7147h	ICU	DTC 启动允许寄存器 071	DT CER071	8	8	2 个 ICLK		
			0008 7166h	ICU	DTC 启动允许寄存器 102	DT CER102	8	8	2 个 ICLK		
			0008 7167h	ICU	DTC 启动允许寄存器 103	DT CER103	8	8	2 个 ICLK		
			0008 716Ah	ICU	DTC 启动允许寄存器 106	DT CER106	8	8	2 个 ICLK		
			0008 716Bh	ICU	DTC 启动允许寄存器 107	DT CER107	8	8	2 个 ICLK		
			0008 7172h	ICU	DTC 启动允许寄存器 114	DT CER114	8	8	2 个 ICLK		
			0008 7173h	ICU	DTC 启动允许寄存器 115	DT CER115	8	8	2 个 ICLK		
			0008 7174h	ICU	DTC 启动允许寄存器 116	DT CER116	8	8	2 个 ICLK		
			0008 7175h	ICU	DTC 启动允许寄存器 117	DT CER117	8	8	2 个 ICLK		

表 5.1 I/O 寄存器地址一览表 (6/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									ICLK \geq PCLK 时	ICLK $<$ PCLK 时	
○	○	○	0008 7179h	ICU	DTC 启动允许寄存器 121	DTCER121	8	8	2 个 ICLK		第 14 章
○	○	○	0008 717Ah	ICU	DTC 启动允许寄存器 122	DTCER122	8	8	2 个 ICLK		
○	○	○	0008 717Dh	ICU	DTC 启动允许寄存器 125	DTCER125	8	8	2 个 ICLK		
○	○	○	0008 717Eh	ICU	DTC 启动允许寄存器 126	DTCER126	8	8	2 个 ICLK		
○	○	○	0008 7181h	ICU	DTC 启动允许寄存器 129	DTCER129	8	8	2 个 ICLK		
○	○	○	0008 7182h	ICU	DTC 启动允许寄存器 130	DTCER130	8	8	2 个 ICLK		
○	○	○	0008 7183h	ICU	DTC 启动允许寄存器 131	DTCER131	8	8	2 个 ICLK		
○	○	○	0008 7184h	ICU	DTC 启动允许寄存器 132	DTCER132	8	8	2 个 ICLK		
○	○	○	0008 7186h	ICU	DTC 启动允许寄存器 134	DTCER134	8	8	2 个 ICLK		
○	○	○	0008 7187h	ICU	DTC 启动允许寄存器 135	DTCER135	8	8	2 个 ICLK		
○	○	○	0008 7188h	ICU	DTC 启动允许寄存器 136	DTCER136	8	8	2 个 ICLK		
○	○	○	0008 7189h	ICU	DTC 启动允许寄存器 137	DTCER137	8	8	2 个 ICLK		
○	○	○	0008 718Ah	ICU	DTC 启动允许寄存器 138	DTCER138	8	8	2 个 ICLK		
○	○	○	0008 718Bh	ICU	DTC 启动允许寄存器 139	DTCER139	8	8	2 个 ICLK		
○	○	○	0008 718Ch	ICU	DTC 启动允许寄存器 140	DTCER140	8	8	2 个 ICLK		
○	○	○	0008 718Dh	ICU	DTC 启动允许寄存器 141	DTCER141	8	8	2 个 ICLK		
○	○	○	0008 71AEh	ICU	DTC 启动允许寄存器 174	DTCER174	8	8	2 个 ICLK		
○	○	○	0008 71AFh	ICU	DTC 启动允许寄存器 175	DTCER175	8	8	2 个 ICLK		
○	○	○	0008 71B1h	ICU	DTC 启动允许寄存器 177	DTCER177	8	8	2 个 ICLK		
○	○	○	0008 71B2h	ICU	DTC 启动允许寄存器 178	DTCER178	8	8	2 个 ICLK		
○	○	○	0008 71B4h	ICU	DTC 启动允许寄存器 180	DTCER180	8	8	2 个 ICLK		
○	○	○	0008 71B5h	ICU	DTC 启动允许寄存器 181	DTCER181	8	8	2 个 ICLK		
○	○	○	0008 71B7h	ICU	DTC 启动允许寄存器 183	DTCER183	8	8	2 个 ICLK		
○	○	○	0008 71B8h	ICU	DTC 启动允许寄存器 184	DTCER184	8	8	2 个 ICLK		
○	○	○	0008 71C6h	ICU	DTC 启动允许寄存器 198	DTCER198	8	8	2 个 ICLK		
○	○	○	0008 71C7h	ICU	DTC 启动允许寄存器 199	DTCER199	8	8	2 个 ICLK		
○	○	○	0008 71C8h	ICU	DTC 启动允许寄存器 200	DTCER200	8	8	2 个 ICLK		
○	○	○	0008 71C9h	ICU	DTC 启动允许寄存器 201	DTCER201	8	8	2 个 ICLK		
○	○	○	0008 71D7h	ICU	DTC 启动允许寄存器 215	DTCER215	8	8	2 个 ICLK		
○	○	○	0008 71D8h	ICU	DTC 启动允许寄存器 216	DTCER216	8	8	2 个 ICLK		
○	○	○	0008 71DBh	ICU	DTC 启动允许寄存器 219	DTCER219	8	8	2 个 ICLK		
○	○	○	0008 71DCh	ICU	DTC 启动允许寄存器 220	DTCER220	8	8	2 个 ICLK		
○	○	○	0008 71DFh	ICU	DTC 启动允许寄存器 223	DTCER223	8	8	2 个 ICLK		
○	○	○	0008 71E0h	ICU	DTC 启动允许寄存器 224	DTCER224	8	8	2 个 ICLK		
○	○	○	0008 71E3h	ICU	DTC 启动允许寄存器 227	DTCER227	8	8	2 个 ICLK		
○	○	○	0008 71E4h	ICU	DTC 启动允许寄存器 228	DTCER228	8	8	2 个 ICLK		
○	○	○	0008 71E7h	ICU	DTC 启动允许寄存器 231	DTCER231	8	8	2 个 ICLK		
○	○	○	0008 71E8h	ICU	DTC 启动允许寄存器 232	DTCER232	8	8	2 个 ICLK		
○	○	○	0008 71EBh	ICU	DTC 启动允许寄存器 235	DTCER235	8	8	2 个 ICLK		
○	○	○	0008 71ECh	ICU	DTC 启动允许寄存器 236	DTCER236	8	8	2 个 ICLK		
○	○	○	0008 71EFh	ICU	DTC 启动允许寄存器 239	DTCER239	8	8	2 个 ICLK		
○	○	○	0008 71F0h	ICU	DTC 启动允许寄存器 240	DTCER240	8	8	2 个 ICLK		
○	○	○	0008 71F7h	ICU	DTC 启动允许寄存器 247	DTCER247	8	8	2 个 ICLK		
○	○	○	0008 71F8h	ICU	DTC 启动允许寄存器 248	DTCER248	8	8	2 个 ICLK		
○	○	○	0008 7202h	ICU	中断请求允许寄存器 02	IER02	8	8	2 个 ICLK		
○	○	○	0008 7203h	ICU	中断请求允许寄存器 03	IER03	8	8	2 个 ICLK		
○	○	○	0008 7204h	ICU	中断请求允许寄存器 04	IER04	8	8	2 个 ICLK		
○	○	○	0008 7205h	ICU	中断请求允许寄存器 05	IER05	8	8	2 个 ICLK		
○	○	○	0008 7207h	ICU	中断请求允许寄存器 07	IER07	8	8	2 个 ICLK		
○	○	○	0008 7208h	ICU	中断请求允许寄存器 08	IER08	8	8	2 个 ICLK		
○	○	○	0008 720Bh	ICU	中断请求允许寄存器 0B	IER0B	8	8	2 个 ICLK		

表 5.1 I/O 寄存器地址一览表 (7/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									ICLK ≥ PCLK 时	ICLK < PCLK 时	
○	○	○	0008 720Ch	ICU	中断请求允许寄存器 0C	IER0C	8	8	2 个 ICLK		第 14 章
○	○	○	0008 720Dh	ICU	中断请求允许寄存器 0D	IER0D	8	8	2 个 ICLK		
○	○	○	0008 720Eh	ICU	中断请求允许寄存器 0E	IER0E	8	8	2 个 ICLK		
○	○	○	0008 720Fh	ICU	中断请求允许寄存器 0F	IER0F	8	8	2 个 ICLK		
○	○	○	0008 7210h	ICU	中断请求允许寄存器 10	IER10	8	8	2 个 ICLK		
○	○	○	0008 7211h	ICU	中断请求允许寄存器 11	IER11	8	8	2 个 ICLK		
○	○	○	0008 7215h	ICU	中断请求允许寄存器 15	IER15	8	8	2 个 ICLK		
○	○	○	0008 7216h	ICU	中断请求允许寄存器 16	IER16	8	8	2 个 ICLK		
○	○	○	0008 7217h	ICU	中断请求允许寄存器 17	IER17	8	8	2 个 ICLK		
○	○	○	0008 7218h	ICU	中断请求允许寄存器 18	IER18	8	8	2 个 ICLK		
○	○	○	0008 7219h	ICU	中断请求允许寄存器 19	IER19	8	8	2 个 ICLK		
○	○	○	0008 721Ah	ICU	中断请求允许寄存器 1A	IER1A	8	8	2 个 ICLK		
○	○	○	0008 721Bh	ICU	中断请求允许寄存器 1B	IER1B	8	8	2 个 ICLK		
○	○	○	0008 721Ch	ICU	中断请求允许寄存器 1C	IER1C	8	8	2 个 ICLK		
○	○	○	0008 721Dh	ICU	中断请求允许寄存器 1D	IER1D	8	8	2 个 ICLK		
○	○	○	0008 721Eh	ICU	中断请求允许寄存器 1E	IER1E	8	8	2 个 ICLK		
○	○	○	0008 721Fh	ICU	中断请求允许寄存器 1F	IER1F	8	8	2 个 ICLK		
○	○	○	0008 72E0h	ICU	软件中断启动寄存器	SWINTR	8	8	2 个 ICLK		
○	○	○	0008 72F0h	ICU	高速中断设定寄存器	FIR	16	16	2 个 ICLK		
○	○	○	0008 7300h	ICU	中断源优先级寄存器 000	IPR000	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7301h	ICU	中断源优先级寄存器 001	IPR001	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7302h	ICU	中断源优先级寄存器 002	IPR002	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7303h	ICU	中断源优先级寄存器 003	IPR003	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7304h	ICU	中断源优先级寄存器 004	IPR004	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7305h	ICU	中断源优先级寄存器 005	IPR005	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7306h	ICU	中断源优先级寄存器 006	IPR006	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7307h	ICU	中断源优先级寄存器 007	IPR007	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7320h	ICU	中断源优先级寄存器 032	IPR032	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7321h	ICU	中断源优先级寄存器 033	IPR033	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7322h	ICU	中断源优先级寄存器 034	IPR034	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 732Ch	ICU	中断源优先级寄存器 044	IPR044	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7339h	ICU	中断源优先级寄存器 057	IPR057	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 733Ah	ICU	中断源优先级寄存器 058	IPR058	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 733Bh	ICU	中断源优先级寄存器 059	IPR059	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 733Fh	ICU	中断源优先级寄存器 063	IPR063	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7340h	ICU	中断源优先级寄存器 064	IPR064	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7341h	ICU	中断源优先级寄存器 065	IPR065	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7342h	ICU	中断源优先级寄存器 066	IPR066	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7343h	ICU	中断源优先级寄存器 067	IPR067	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7344h	ICU	中断源优先级寄存器 068	IPR068	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7345h	ICU	中断源优先级寄存器 069	IPR069	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7346h	ICU	中断源优先级寄存器 070	IPR070	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7347h	ICU	中断源优先级寄存器 071	IPR071	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7358h	ICU	中断源优先级寄存器 088	IPR088	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7359h	ICU	中断源优先级寄存器 089	IPR089	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 735Ch	ICU	中断源优先级寄存器 092	IPR092	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 735Dh	ICU	中断源优先级寄存器 093	IPR093	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7366h	ICU	中断源优先级寄存器 102	IPR102	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7367h	ICU	中断源优先级寄存器 103	IPR103	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 736Ah	ICU	中断源优先级寄存器 106	IPR106	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 736Bh	ICU	中断源优先级寄存器 107	IPR107	8	8	读时 3 个 ICLK、写时 2 个 ICLK		

表 5.1 I/O 寄存器地址一览表 (8/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									ICLK ≥ PCLK 时	ICLK < PCLK 时	
○	○	○	0008 7372h	ICU	中断源优先级寄存器 114	IPR114	8	8	读时 3 个 ICLK、写时 2 个 ICLK	第 14 章	
○	○	○	0008 7376h	ICU	中断源优先级寄存器 118	IPR118	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7379h	ICU	中断源优先级寄存器 121	IPR121	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 737Bh	ICU	中断源优先级寄存器 123	IPR123	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 737Dh	ICU	中断源优先级寄存器 125	IPR125	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 737Fh	ICU	中断源优先级寄存器 127	IPR127	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7381h	ICU	中断源优先级寄存器 129	IPR129	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7385h	ICU	中断源优先级寄存器 133	IPR133	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7386h	ICU	中断源优先级寄存器 134	IPR134	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 738Ah	ICU	中断源优先级寄存器 138	IPR138	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 738Bh	ICU	中断源优先级寄存器 139	IPR139	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73AAh	ICU	中断源优先级寄存器 170	IPR170	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73ABh	ICU	中断源优先级寄存器 171	IPR171	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73AEh	ICU	中断源优先级寄存器 174	IPR174	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73B1h	ICU	中断源优先级寄存器 177	IPR177	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73B4h	ICU	中断源优先级寄存器 180	IPR180	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73B7h	ICU	中断源优先级寄存器 183	IPR183	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73C6h	ICU	中断源优先级寄存器 198	IPR198	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73C7h	ICU	中断源优先级寄存器 199	IPR199	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73C8h	ICU	中断源优先级寄存器 200	IPR200	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73C9h	ICU	中断源优先级寄存器 201	IPR201	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73D6h	ICU	中断源优先级寄存器 214	IPR214	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73DAh	ICU	中断源优先级寄存器 218	IPR218	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73DEh	ICU	中断源优先级寄存器 222	IPR222	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73E2h	ICU	中断源优先级寄存器 226	IPR226	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73E6h	ICU	中断源优先级寄存器 230	IPR230	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73EAh	ICU	中断源优先级寄存器 234	IPR234	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73EEh	ICU	中断源优先级寄存器 238	IPR238	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73F2h	ICU	中断源优先级寄存器 242	IPR242	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73F3h	ICU	中断源优先级寄存器 243	IPR243	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73F4h	ICU	中断源优先级寄存器 244	IPR244	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73F5h	ICU	中断源优先级寄存器 245	IPR245	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73F6h	ICU	中断源优先级寄存器 246	IPR246	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73F7h	ICU	中断源优先级寄存器 247	IPR247	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73F8h	ICU	中断源优先级寄存器 248	IPR248	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 73F9h	ICU	中断源优先级寄存器 249	IPR249	8	8	读时 3 个 ICLK、写时 2 个 ICLK		
○	○	○	0008 7400h	ICU	DMAC 启动请求选择寄存器 0	DMRSR0	8	8	2 个 ICLK		
○	○	○	0008 7404h	ICU	DMAC 启动请求选择寄存器 1	DMRSR1	8	8	2 个 ICLK		
○	○	○	0008 7408h	ICU	DMAC 启动请求选择寄存器 2	DMRSR2	8	8	2 个 ICLK		
○	○	○	0008 740Ch	ICU	DMAC 启动请求选择寄存器 3	DMRSR3	8	8	2 个 ICLK		
○	○	○	0008 7500h	ICU	IRQ 控制寄存器 0	IRQCR0	8	8	2 个 ICLK		
○	○	○	0008 7501h	ICU	IRQ 控制寄存器 1	IRQCR1	8	8	2 个 ICLK		
○	○	○	0008 7502h	ICU	IRQ 控制寄存器 2	IRQCR2	8	8	2 个 ICLK		
○	○	○	0008 7503h	ICU	IRQ 控制寄存器 3	IRQCR3	8	8	2 个 ICLK		
○	○	○	0008 7504h	ICU	IRQ 控制寄存器 4	IRQCR4	8	8	2 个 ICLK		
○	○	○	0008 7505h	ICU	IRQ 控制寄存器 5	IRQCR5	8	8	2 个 ICLK		
○	○	○	0008 7506h	ICU	IRQ 控制寄存器 6	IRQCR6	8	8	2 个 ICLK		
○	○	○	0008 7507h	ICU	IRQ 控制寄存器 7	IRQCR7	8	8	2 个 ICLK		
○	○	○	0008 7510h	ICU	IRQ 引脚数字滤波器的允许寄存器 0	IRQFLTE0	8	8	2 个 ICLK		
○	○	○	0008 7514h	ICU	IRQ 引脚数字滤波器的设定寄存器 0	IRQFLTC0	16	16	2 个 ICLK		
○	○	○	0008 7580h	ICU	非屏蔽中断状态寄存器	NMISR	8	8	2 个 ICLK		

表 5.1 I/O 寄存器地址一览表 (9/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									ICLK ≥ PCLK 时	ICLK < PCLK 时	
○	○	○	0008 7581h	ICU	非屏蔽中断允许寄存器	NMIER	8	8	2 个 ICLK		第 14 章
○	○	○	0008 7582h	ICU	非屏蔽中断状态清除寄存器	NMICLR	8	8	2 个 ICLK		
○	○	○	0008 7583h	ICU	NMI 引脚中断控制寄存器	NMICR	8	8	2 个 ICLK		
○	○	○	0008 7590h	ICU	NMI 端子数字滤波器的允许寄存器	NMIFLTE	8	8	2 个 ICLK		
○	○	○	0008 7594h	ICU	NMI 端子数字滤波器的设定寄存器	NMIFLTC	8	8	2 个 ICLK		
○	○	○	0008 8000h	CMT	比较匹配定时器的启动寄存器 0	CMSTR0	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	第 24 章
○	○	○	0008 8002h	CMT0	比较匹配定时器的控制寄存器	CMCR	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8004h	CMT0	比较匹配定时器的计数器	CMCNT	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8006h	CMT0	比较匹配定时器的常数寄存器	CMCOR	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8008h	CMT1	比较匹配定时器的控制寄存器	CMCR	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 800Ah	CMT1	比较匹配定时器的计数器	CMCNT	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 800Ch	CMT1	比较匹配定时器的常数寄存器	CMCOR	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8010h	CMT	比较匹配定时器的启动寄存器 1	CMSTR1	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8012h	CMT2	比较匹配定时器的控制寄存器	CMCR	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8014h	CMT2	比较匹配定时器的计数器	CMCNT	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8016h	CMT2	比较匹配定时器的常数寄存器	CMCOR	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8018h	CMT3	比较匹配定时器的控制寄存器	CMCR	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 801Ah	CMT3	比较匹配定时器的计数器	CMCNT	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 801Ch	CMT3	比较匹配定时器的常数寄存器	CMCOR	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8020h	WDT	WDT 刷新寄存器	WDTRR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8022h	WDT	WDT 控制寄存器	WDTCR	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8024h	WDT	WDT 状态寄存器	WDTSR	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8026h	WDT	WDT 复位控制寄存器	WDTRCR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8030h	IWDT	IWDT 刷新寄存器	IWDTRR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	第 27 章
○	○	○	0008 8032h	IWDT	IWDT 控制寄存器	IWDTCR	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8034h	IWDT	IWDT 状态寄存器	IWDTSR	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8036h	IWDT	IWDT 复位控制寄存器	IWDTRCR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8038h	IWDT	IWDT 计数停止控制寄存器	IWDTCSTPR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 80C0h	DA	D/A 数据寄存器 0	DADR0	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	第 33 章
○	○	○	0008 80C2h	DA	D/A 数据寄存器 1	DADR1	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 80C4h	DA	D/A 控制寄存器	DACR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 80C5h	DA	DADRm 格式选择寄存器	DADPR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8200h	TMR0	定时器的控制寄存器	TCR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	第 23 章
○	○	○	0008 8201h	TMR1	定时器的控制寄存器	TCR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8202h	TMR0	定时器的控制 / 状态寄存器	TCSR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8203h	TMR1	定时器的控制 / 状态寄存器	TCSR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8204h	TMR0	时间常数寄存器 A	TCORA	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8205h	TMR1	时间常数寄存器 A	TCORA	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8206h	TMR0	时间常数寄存器 B	TCORB	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8207h	TMR1	时间常数寄存器 B	TCORB	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8208h	TMR0	定时器的计数器	TCNT	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8209h	TMR1	定时器的计数器	TCNT	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 820Ah	TMR0	定时器的计数器控制寄存器	TCCR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 820Bh	TMR1	定时器的计数器控制寄存器	TCCR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 820Ch	TMR0	定时器的计数器启动寄存器	TCSTR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8210h	TMR2	定时器的控制寄存器	TCR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8211h	TMR3	定时器的控制寄存器	TCR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8212h	TMR2	定时器的控制 / 状态寄存器	TCSR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8213h	TMR3	定时器的控制 / 状态寄存器	TCSR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8214h	TMR2	时间常数寄存器 A	TCORA	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8215h	TMR3	时间常数寄存器 A	TCORA	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	

表 5.1 I/O 寄存器地址一览表 (10/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									ICLK ≥ PCLK 时	ICLK < PCLK 时	
○	○	○	0008 8216h	TMR2	时间常数寄存器 B	TCORB	8	8	2~3 个 PCLKB	2 个 ICLK	第 23 章
○	○	○	0008 8217h	TMR3	时间常数寄存器 B	TCORB	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8218h	TMR2	定时器的计数器	TCNT	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8219h	TMR3	定时器的计数器	TCNT	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 821Ah	TMR2	定时器的计数器控制寄存器	TCCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 821Bh	TMR3	定时器的计数器控制寄存器	TCCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 821Ch	TMR2	定时器的计数器启动寄存器	TCSTR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8280h	CRC	CRC 控制寄存器	CRCCR	8	8	2~3 个 PCLKB	2 个 ICLK	第 31 章
○	○	○	0008 8281h	CRC	CRC 数据输入寄存器	CRCDIR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8282h	CRC	CRC 数据输出寄存器	CRCDOR	16	16	2~3 个 PCLKB	2 个 ICLK	第 29 章
○	○	○	0008 8300h	RIIC0	I ² C 总线控制寄存器 1	ICCR1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8301h	RIIC0	I ² C 总线控制寄存器 2	ICCR2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8302h	RIIC0	I ² C 总线模式寄存器 1	ICMR1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8303h	RIIC0	I ² C 总线模式寄存器 2	ICMR2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8304h	RIIC0	I ² C 总线模式寄存器 3	ICMR3	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8305h	RIIC0	I ² C 总线功能允许寄存器	ICFER	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8306h	RIIC0	I ² C 总线状态允许寄存器	ICSER	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8307h	RIIC0	I ² C 总线中断允许寄存器	ICIER	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8308h	RIIC0	I ² C 总线状态寄存器 1	ICSR1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8309h	RIIC0	I ² C 总线状态寄存器 2	ICSR2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 830Ah	RIIC0	从属地址寄存器 L0	SARL0	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 830Bh	RIIC0	从属地址寄存器 U0	SARU0	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 830Ch	RIIC0	从属地址寄存器 L1	SARL1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 830Dh	RIIC0	从属地址寄存器 U1	SARU1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 830Eh	RIIC0	从属地址寄存器 L2	SARL2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 830Fh	RIIC0	从属地址寄存器 U2	SARU2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8310h	RIIC0	I ² C 总线位速率低电平寄存器	ICBRL	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8311h	RIIC0	I ² C 总线位速率高电平寄存器	ICBRH	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8312h	RIIC0	I ² C 总线发送数据寄存器	ICDRT	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8313h	RIIC0	I ² C 总线接收数据寄存器	ICDRR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8380h	RSPI0	RSPI 控制寄存器	SPCR	8	8	2~3 个 PCLKB	2 个 ICLK	第 30 章
○	○	○	0008 8381h	RSPI0	RSPI 从属选择极性寄存器	SSLP	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8382h	RSPI0	RSPI 引脚控制寄存器	SPPCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8383h	RSPI0	RSPI 状态寄存器	SPSR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8384h	RSPI0	RSPI 数据寄存器	SPDR	32	16、32	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8388h	RSPI0	RSPI 顺序控制寄存器	SPSCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8389h	RSPI0	RSPI 顺序状态寄存器	SPSSR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 838Ah	RSPI0	RSPI 位速率寄存器	SPBR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 838Bh	RSPI0	RSPI 数据控制寄存器	SPDCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 838Ch	RSPI0	RSPI 时钟延迟寄存器	SPCKD	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 838Dh	RSPI0	RSPI 从属选择无效延迟寄存器	SSLND	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 838Eh	RSPI0	RSPI 下次存取延迟寄存器	SPND	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 838Fh	RSPI0	RSPI 控制寄存器 2	SPCR2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8390h	RSPI0	RSPI 命令寄存器 0	SPCMD0	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8392h	RSPI0	RSPI 命令寄存器 1	SPCMD1	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8394h	RSPI0	RSPI 命令寄存器 2	SPCMD2	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8396h	RSPI0	RSPI 命令寄存器 3	SPCMD3	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8398h	RSPI0	RSPI 命令寄存器 4	SPCMD4	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 839Ah	RSPI0	RSPI 命令寄存器 5	SPCMD5	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 839Ch	RSPI0	RSPI 命令寄存器 6	SPCMD6	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 839Eh	RSPI0	RSPI 命令寄存器 7	SPCMD7	16	16	2~3 个 PCLKB	2 个 ICLK	

表 5.1 I/O 寄存器地址一览表 (11/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									ICLK ≥ PCLK 时	ICLK < PCLK 时	
○	○	○	0008 8600h	MTU3	定时器的控制寄存器	TCR	8	8	2~3 个 PCLKB	2 个 ICLK	第 21 章
○	○	○	0008 8601h	MTU4	定时器的控制寄存器	TCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8602h	MTU3	定时器的模式寄存器	TMDR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8603h	MTU4	定时器的模式寄存器	TMDR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8604h	MTU3	定时器的 I/O 控制寄存器 H	TIORH	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8605h	MTU3	定时器的 I/O 控制寄存器 L	TIORL	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8606h	MTU4	定时器的 I/O 控制寄存器 H	TIORH	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8607h	MTU4	定时器的 I/O 控制寄存器 L	TIORL	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8608h	MTU3	定时器的中断允许寄存器	TIER	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8609h	MTU4	定时器的中断允许寄存器	TIER	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 860Ah	MTU	定时器的输出主控允许寄存器	TOER	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 860Dh	MTU	定时器的门控寄存器	TGCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 860Eh	MTU	定时器的输出控制寄存器 1	TOCR1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 860Fh	MTU	定时器的输出控制寄存器 2	TOCR2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8610h	MTU3	定时器的计数器	TCNT	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8612h	MTU4	定时器的计数器	TCNT	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8614h	MTU	定时器的周期数据寄存器	TCDR	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8616h	MTU	定时器的死区时间数据寄存器	TDDR	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8618h	MTU3	定时器的通用寄存器 A	TGRA	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 861Ah	MTU3	定时器的通用寄存器 B	TGRB	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 861Ch	MTU4	定时器的通用寄存器 A	TGRA	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 861Eh	MTU4	定时器的通用寄存器 B	TGRB	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8620h	MTU	定时器的副计数器	TCNTS	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8622h	MTU	定时器的周期缓冲寄存器	TGBR	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8624h	MTU3	定时器的通用寄存器 C	TGRC	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8626h	MTU3	定时器的通用寄存器 D	TGRD	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8628h	MTU4	定时器的通用寄存器 C	TGRC	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 862Ah	MTU4	定时器的通用寄存器 D	TGRD	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 862Ch	MTU3	定时器的状态寄存器	TSR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 862Dh	MTU4	定时器的状态寄存器	TSR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8630h	MTU	定时器的中断减少设定寄存器	TITCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8631h	MTU	定时器的中断减少次数计数器	TITCNT	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8632h	MTU	定时器的缓冲传送设定寄存器	TBTER	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8634h	MTU	定时器的死区时间允许寄存器	TDER	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8636h	MTU	定时器的输出电平缓冲寄存器	TOLBR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8638h	MTU3	定时器的缓冲运行传送模式寄存器	TBTM	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8639h	MTU4	定时器的缓冲运行传送模式寄存器	TBTM	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8640h	MTU4	定时器的 A/D 转换开始请求控制寄存器	TADCR	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8644h	MTU4	定时器的 A/D 转换开始请求周期设定寄存器 A	TADCORA	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8646h	MTU4	定时器的 A/D 转换开始请求周期设定寄存器 B	TADCORB	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8648h	MTU4	定时器的 A/D 转换开始请求周期设定缓冲寄存器 A	TADCOBRA	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 864Ah	MTU4	定时器的 A/D 转换开始请求周期设定缓冲寄存器 B	TADCOBRB	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8660h	MTU	定时器的波形控制寄存器	TWCR	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8680h	MTU	定时器的启动寄存器	TSTR	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8681h	MTU	定时器的同步寄存器	TSYR	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8684h	MTU	定时器的读写允许寄存器	TRWER	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8690h	MTU0	噪声滤波器的控制寄存器	NFCR	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8691h	MTU1	噪声滤波器的控制寄存器	NFCR	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8692h	MTU2	噪声滤波器的控制寄存器	NFCR	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8693h	MTU3	噪声滤波器的控制寄存器	NFCR	8	8、16	2~3 个 PCLKB	2 个 ICLK	

表 5.1 I/O 寄存器地址一览表 (12/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									ICLK ≥ PCLK 时	ICLK < PCLK 时	
○	○	○	0008 8694h	MTU4	噪声滤波器的控制寄存器	NFCR	8	8、16	2~3 个 PCLKB	2 个 ICLK	第 21 章
○	○	○	0008 8695h	MTU5	噪声滤波器控制寄存器的	NFCR	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8700h	MTU0	定时器的控制寄存器	TCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8701h	MTU0	定时器的模式寄存器	TMDR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8702h	MTU0	定时器的 I/O 控制寄存器 H	TIORH	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8703h	MTU0	定时器的 I/O 控制寄存器 L	TIORL	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8704h	MTU0	定时器的中断允许寄存器	TIER	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8705h	MTU0	定时器的状态寄存器	TSR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8706h	MTU0	定时器的计数器	TCNT	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8708h	MTU0	定时器的通用寄存器 A	TGRA	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 870Ah	MTU0	定时器的通用寄存器 B	TGRB	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 870Ch	MTU0	定时器的通用寄存器 C	TGRC	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 870Eh	MTU0	定时器的通用寄存器 D	TGRD	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8720h	MTU0	定时器的通用寄存器 E	TGRE	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8722h	MTU0	定时器的通用寄存器 F	TGRF	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8724h	MTU0	定时器的中断允许寄存器 2	TIER2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8726h	MTU0	定时器的缓冲运行传送模式寄存器	TBTM	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8780h	MTU1	定时器的控制寄存器	TCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8781h	MTU1	定时器的模式寄存器	TMDR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8782h	MTU1	定时器的 I/O 控制寄存器	TIOR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8784h	MTU1	定时器的中断允许寄存器	TIER	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8785h	MTU1	定时器的状态寄存器	TSR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8786h	MTU1	定时器的计数器	TCNT	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8788h	MTU1	定时器的通用寄存器 A	TGRA	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 878Ah	MTU1	定时器的通用寄存器 B	TGRB	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8790h	MTU1	定时器的输入捕捉控制寄存器	TICCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8800h	MTU2	定时器的控制寄存器	TCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8801h	MTU2	定时器的模式寄存器	TMDR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8802h	MTU2	定时器的 I/O 控制寄存器	TIOR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8804h	MTU2	定时器的中断允许寄存器	TIER	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8805h	MTU2	定时器的状态寄存器	TSR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8806h	MTU2	定时器的计数器	TCNT	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8808h	MTU2	定时器的通用寄存器 A	TGRA	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 880Ah	MTU2	定时器的通用寄存器 B	TGRB	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8880h	MTU5	定时器的计数器 U	TCNTU	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8882h	MTU5	定时器的通用寄存器 U	TGRU	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8884h	MTU5	定时器的控制寄存器 U	TCRU	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8886h	MTU5	定时器的 I/O 控制寄存器 U	TIORU	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8890h	MTU5	定时器的计数器 V	TCNTV	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8892h	MTU5	定时器的通用寄存器 V	TGRV	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8894h	MTU5	定时器的控制寄存器 V	TCRV	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8896h	MTU5	定时器的 I/O 控制寄存器 V	TIORV	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 88A0h	MTU5	定时器的计数器 W	TCNTW	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 88A2h	MTU5	定时器的通用寄存器 W	TGRW	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 88A4h	MTU5	定时器的控制寄存器 W	TCRW	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 88A6h	MTU5	定时器的 I/O 控制寄存器 W	TIORW	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 88B2h	MTU5	定时器的中断允许寄存器	TIER	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 88B4h	MTU5	定时器的启动寄存器	TSTR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 88B6h	MTU5	定时器的比较匹配清除寄存器	TCNTCMPCLR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 8900h	POE	输入电平控制 / 状态寄存器 1	ICSR1	16	8、16	2~3 个 PCLKB	2 个 ICLK	第 22 章
○	○	○	0008 8902h	POE	输出电平控制 / 状态寄存器 1	OCSR1	16	8、16	2~3 个 PCLKB	2 个 ICLK	

表 5.1 I/O 寄存器地址一览表 (13/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									ICLK ≥ PCLK 时	ICLK < PCLK 时	
○	○	○	0008 8908h	POE	输入电平控制 / 状态寄存器 2	ICSR2	16	8、16	2 ~ 3 个 PCLKB	2 个 ICLK	第 22 章
○	○	○	0008 890Ah	POE	软件端口输出允许寄存器	SPOER	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 890Bh	POE	端口输出允许控制寄存器 1	POECR1	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 890Ch	POE	端口输出允许控制寄存器 2	POECR2	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 890Eh	POE	输入电平控制 / 状态寄存器 3	ICSR3	16	8、16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9000h	S12AD	A/D 控制寄存器	ADCSR	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	第 32 章
○	○	○	0008 9004h	S12AD	A/D 通道选择寄存器 A	ADANSA	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9008h	S12AD	A/D 转换值加法运算模式选择寄存器	ADADS	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 900Ch	S12AD	A/D 转换值加法运算次数选择寄存器	ADADC	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 900Eh	S12AD	A/D 控制扩展寄存器	ADCER	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9010h	S12AD	A/D 开始触发选择寄存器	ADSTRGR	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9012h	S12AD	A/D 转换扩展输入控制寄存器	ADEXICR	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9014h	S12AD	A/D 通道选择寄存器 B	ADANSB	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9018h	S12AD	A/D 数据双重化寄存器	ADBLDR	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 901Ah	S12AD	A/D 温度传感器的数据寄存器	ADTSDR	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 901Ch	S12AD	A/D 内部基准电压的数据寄存器	ADOCDR	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 901Eh	S12AD	A/D 自诊断数据寄存器	ADDRD	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9020h	S12AD	A/D 数据寄存器 0	ADDR0	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9022h	S12AD	A/D 数据寄存器 1	ADDR1	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9024h	S12AD	A/D 数据寄存器 2	ADDR2	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9026h	S12AD	A/D 数据寄存器 3	ADDR3	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9028h	S12AD	A/D 数据寄存器 4	ADDR4	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	—	0008 902Ah	S12AD	A/D 数据寄存器 5	ADDR5	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 902Ch	S12AD	A/D 数据寄存器 6	ADDR6	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	—	0008 902Eh	S12AD	A/D 数据寄存器 7	ADDR7	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9030h	S12AD	A/D 数据寄存器 8	ADDR8	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9032h	S12AD	A/D 数据寄存器 9	ADDR9	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9034h	S12AD	A/D 数据寄存器 10	ADDR10	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9036h	S12AD	A/D 数据寄存器 11	ADDR11	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9038h	S12AD	A/D 数据寄存器 12	ADDR12	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 903Ah	S12AD	A/D 数据寄存器 13	ADDR13	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	—	—	0008 903Ch	S12AD	A/D 数据寄存器 14	ADDR14	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	—	—	0008 903Eh	S12AD	A/D 数据寄存器 15	ADDR15	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9060h	S12AD	A/D 采样状态寄存器 0	ADSSTR0	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9061h	S12AD	A/D 采样状态寄存器 L	ADSSTRL	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9066h	S12AD	采样 & 保持电路控制寄存器	ADSHCR	16	16	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9070h	S12AD	A/D 采样状态寄存器 T	ADSSTRT	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9071h	S12AD	A/D 采样状态寄存器 O	ADSSTRO	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9073h	S12AD	A/D 采样状态寄存器 1	ADSSTR1	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9074h	S12AD	A/D 采样状态寄存器 2	ADSSTR2	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9075h	S12AD	A/D 采样状态寄存器 3	ADSSTR3	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9076h	S12AD	A/D 采样状态寄存器 4	ADSSTR4	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	—	0008 9077h	S12AD	A/D 采样状态寄存器 5	ADSSTR5	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 9078h	S12AD	A/D 采样状态寄存器 6	ADSSTR6	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	—	0008 9079h	S12AD	A/D 采样状态寄存器 7	ADSSTR7	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 907Ah	S12AD	A/D 断线检测控制寄存器	ADDISCR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	—	0008 A000h	SCI0	串行模式寄存器	SMR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	28 章
○	○	—	0008 A001h	SCI0	位速率寄存器	BRR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	—	0008 A002h	SCI0	串行控制寄存器	SCR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	—	0008 A003h	SCI0	发送数据寄存器	TDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	—	0008 A004h	SCI0	串行状态寄存器	SSR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	

表 5.1 I/O 寄存器地址一览表 (14/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									ICLK ≥ PCLK 时	ICLK < PCLK 时	
○	○	—	0008 A005h	SCI0	接收数据寄存器	RDR	8	8	2~3 个 PCLKB	2 个 ICLK	28 章
○	○	—	0008 A006h	SCI0	智能卡模式寄存器	SCMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 A007h	SCI0	串行扩展模式寄存器	SEMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 A008h	SCI0	噪声滤波器的设定寄存器	SNFR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 A009h	SCI0	I ² C 模式寄存器 1	SIMR1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 A00Ah	SCI0	I ² C 模式寄存器 2	SIMR2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 A00Bh	SCI0	I ² C 模式寄存器 3	SIMR3	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 A00Ch	SCI0	I ² C 状态寄存器	SISR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 A00Dh	SCI0	SPI 模式寄存器	SPMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A020h	SCI1	串行模式寄存器	SMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A021h	SCI1	位速率寄存器	BRR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A022h	SCI1	串行控制寄存器	SCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A023h	SCI1	发送数据寄存器	TDR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A024h	SCI1	串行状态寄存器	SSR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A025h	SCI1	接收数据寄存器	RDR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A026h	SCI1	智能卡模式寄存器	SCMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A027h	SCI1	串行扩展模式寄存器	SEMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A028h	SCI1	噪声滤波器的设定寄存器	SNFR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A029h	SCI1	I ² C 模式寄存器 1	SIMR1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A02Ah	SCI1	I ² C 模式寄存器 2	SIMR2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A02Bh	SCI1	I ² C 模式寄存器 3	SIMR3	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A02Ch	SCI1	I ² C 状态寄存器	SISR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A02Dh	SCI1	SPI 模式寄存器	SPMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0A0h	SCI5	串行模式寄存器	SMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0A1h	SCI5	位速率寄存器	BRR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0A2h	SCI5	串行控制寄存器	SCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0A3h	SCI5	发送数据寄存器	TDR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0A4h	SCI5	串行状态寄存器	SSR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0A5h	SCI5	接收数据寄存器	RDR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0A6h	SCI5	智能卡模式寄存器	SCMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0A7h	SCI5	串行扩展模式寄存器	SEMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0A8h	SCI5	噪声滤波器的设定寄存器	SNFR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0A9h	SCI5	I ² C 模式寄存器 1	SIMR1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0AAh	SCI5	I ² C 模式寄存器 2	SIMR2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0ABh	SCI5	I ² C 模式寄存器 3	SIMR3	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0ACh	SCI5	I ² C 状态寄存器	SISR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0ADh	SCI5	SPI 模式寄存器	SPMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0C0h	SCI6	串行模式寄存器	SMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0C1h	SCI6	位速率寄存器	BRR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0C2h	SCI6	串行控制寄存器	SCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0C3h	SCI6	发送数据寄存器	TDR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0C4h	SCI6	串行状态寄存器	SSR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0C5h	SCI6	接收数据寄存器	RDR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0C6h	SCI6	智能卡模式寄存器	SCMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0C7h	SCI6	串行扩展模式寄存器	SEMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0C8h	SCI6	噪声滤波器的设定寄存器	SNFR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0C9h	SCI6	I ² C 模式寄存器 1	SIMR1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0CAh	SCI6	I ² C 模式寄存器 2	SIMR2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0CBh	SCI6	I ² C 模式寄存器 3	SIMR3	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0CCh	SCI6	I ² C 状态寄存器	SISR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A0CDh	SCI6	SPI 模式寄存器	SPMR	8	8	2~3 个 PCLKB	2 个 ICLK	

表 5.1 I/O 寄存器地址一览表 (15/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									ICLK ≥ PCLK 时	ICLK < PCLK 时	
○	○	○	0008 A100h	SCI8	串行模式寄存器	SMR	8	8	2~3 个 PCLKB	2 个 ICLK	第 28 章
○	○	○	0008 A101h	SCI8	位速率寄存器	BRR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A102h	SCI8	串行控制寄存器	SCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A103h	SCI8	发送数据寄存器	TDR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A104h	SCI8	串行状态寄存器	SSR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A105h	SCI8	接收数据寄存器	RDR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A106h	SCI8	智能卡模式寄存器	SCMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A107h	SCI8	串行扩展模式寄存器	SEMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A108h	SCI8	噪声滤波器的设定寄存器	SNFR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A109h	SCI8	I ² C 模式寄存器 1	SIMR1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A10Ah	SCI8	I ² C 模式寄存器 2	SIMR2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A10Bh	SCI8	I ² C 模式寄存器 3	SIMR3	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A10Ch	SCI8	I ² C 状态寄存器	SISR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A10Dh	SCI8	SPI 模式寄存器	SPMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A120h	SCI9	串行模式寄存器	SMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A121h	SCI9	位速率寄存器	BRR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A122h	SCI9	串行控制寄存器	SCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A123h	SCI9	发送数据寄存器	TDR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A124h	SCI9	串行状态寄存器	SSR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A125h	SCI9	接收数据寄存器	RDR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A126h	SCI9	智能卡模式寄存器	SCMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A127h	SCI9	串行扩展模式寄存器	SEMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A128h	SCI9	噪声滤波器的设定寄存器	SNFR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A129h	SCI9	I ² C 模式寄存器 1	SIMR1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A12Ah	SCI9	I ² C 模式寄存器 2	SIMR2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A12Bh	SCI9	I ² C 模式寄存器 3	SIMR3	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A12Ch	SCI9	I ² C 状态寄存器	SISR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 A12Dh	SCI9	SPI 模式寄存器	SPMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B000h	CAC	CAC 控制寄存器 0	CACR0	8	8	2~3 个 PCLKB	2 个 ICLK	第 10 章
○	○	○	0008 B001h	CAC	CAC 控制寄存器 1	CACR1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B002h	CAC	CAC 控制寄存器 2	CACR2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B003h	CAC	CAC 中断控制寄存器	CAICR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B004h	CAC	CAC 状态寄存器	CASTR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B006h	CAC	CAC 上限值设定寄存器	CAULVR	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B008h	CAC	CAC 下限值设定寄存器	CALLVR	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B00Ah	CAC	CAC 计数器的缓冲寄存器	CACNTBR	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B080h	DOC	DOC 控制寄存器	DOCR	8	8	2~3 个 PCLKB	2 个 ICLK	第 37 章
○	○	○	0008 B082h	DOC	DOC 数据输入寄存器	DODIR	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B084h	DOC	DOC 数据设定寄存器	DODSR	16	16	2~3 个 PCLKB	2 个 ICLK	第 18 章
○	○	○	0008 B100h	ELC	事件链接控制寄存器	ELCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B102h	ELC	事件链接设定寄存器 1	ELSR1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B103h	ELC	事件链接设定寄存器 2	ELSR2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B104h	ELC	事件链接设定寄存器 3	ELSR3	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B105h	ELC	事件链接设定寄存器 4	ELSR4	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B108h	ELC	事件链接设定寄存器 7	ELSR7	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B10Bh	ELC	事件链接设定寄存器 10	ELSR10	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B10Dh	ELC	事件链接设定寄存器 12	ELSR12	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B110h	ELC	事件链接设定寄存器 15	ELSR15	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B111h	ELC	事件链接设定寄存器 16	ELSR16	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B113h	ELC	事件链接设定寄存器 18	ELSR18	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B114h	ELC	事件链接设定寄存器 19	ELSR19	8	8	2~3 个 PCLKB	2 个 ICLK	

表 5.1 I/O 寄存器地址一览表 (16/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章	
									ICLK ≥ PCLK 时	ICLK < PCLK 时		
○	○	○	0008 B115h	ELC	事件链接设定寄存器 20	ELSR20	8	8	2~3 个 PCLKB	2 个 ICLK	第 18 章	
○	○	○	0008 B116h	ELC	事件链接设定寄存器 21	ELSR21	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B117h	ELC	事件链接设定寄存器 22	ELSR22	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B118h	ELC	事件链接设定寄存器 23	ELSR23	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B119h	ELC	事件链接设定寄存器 24	ELSR24	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B11Ah	ELC	事件链接设定寄存器 25	ELSR25	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B11Bh	ELC	事件链接设定寄存器 26	ELSR26	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B11Ch	ELC	事件链接设定寄存器 27	ELSR27	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B11Dh	ELC	事件链接设定寄存器 28	ELSR28	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B11Eh	ELC	事件链接设定寄存器 29	ELSR29	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B11Fh	ELC	事件链接选项设定寄存器 A	ELOPA	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B120h	ELC	事件链接选项设定寄存器 B	ELOPB	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B121h	ELC	事件链接选项设定寄存器 C	ELOPC	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B122h	ELC	事件链接选项设定寄存器 D	ELOPD	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B123h	ELC	端口组指定寄存器 1	PGR1	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B124h	ELC	端口组指定寄存器 2	PGR2	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B125h	ELC	端口组控制寄存器 1	PGC1	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B126h	ELC	端口组控制寄存器 2	PGC2	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B127h	ELC	端口缓冲寄存器 1	PDBF1	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B128h	ELC	端口缓冲寄存器 2	PDBF2	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B129h	ELC	事件链接端口指定寄存器 0	PEL0	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B12Ah	ELC	事件链接端口指定寄存器 1	PEL1	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B12Bh	ELC	事件链接端口指定寄存器 2	PEL2	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B12Ch	ELC	事件链接端口指定寄存器 3	PEL3	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B12Dh	ELC	事件链接的软件事件发生寄存器	ELSEGR	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B300h	SCI12	串行模式寄存器	SMR	8	8	2~3 个 PCLKB	2 个 ICLK		第 28 章
○	○	○	0008 B301h	SCI12	位速率寄存器	BRR	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B302h	SCI12	串行控制寄存器	SCR	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B303h	SCI12	发送数据寄存器	TDR	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B304h	SCI12	串行状态寄存器	SSR	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B305h	SCI12	接收数据寄存器	RDR	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B306h	SCI12	智能卡模式寄存器	SCMR	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B307h	SCI12	串行扩展模式寄存器	SEMR	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B308h	SCI12	噪声滤波器的设定寄存器	SNFR	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B309h	SCI12	I ² C 模式寄存器 1	SIMR1	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B30Ah	SCI12	I ² C 模式寄存器 2	SIMR2	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B30Bh	SCI12	I ² C 模式寄存器 3	SIMR3	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B30Ch	SCI12	I ² C 状态寄存器	SISR	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B30Dh	SCI12	SPI 模式寄存器	SPMR	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B320h	SCI12	扩展串行模式有效寄存器	ESMER	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B321h	SCI12	控制寄存器 0	CR0	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B322h	SCI12	控制寄存器 1	CR1	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B323h	SCI12	控制寄存器 2	CR2	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B324h	SCI12	控制寄存器 3	CR3	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B325h	SCI12	端口控制寄存器	PCR	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B326h	SCI12	中断控制寄存器	ICR	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B327h	SCI12	状态寄存器	STR	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B328h	SCI12	状态清除寄存器	STCR	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B329h	SCI12	Control Field 0 数据寄存器	CF0DR	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B32Ah	SCI12	Control Field 0 比较允许寄存器	CF0CR	8	8	2~3 个 PCLKB	2 个 ICLK		
○	○	○	0008 B32Bh	SCI12	Control Field 0 接收数据寄存器	CF0RR	8	8	2~3 个 PCLKB	2 个 ICLK		

表 5.1 I/O 寄存器地址一览表 (17/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									ICLK ≥ PCLK 时	ICLK < PCLK 时	
○	○	○	0008 B32Ch	SCI12	主 Control Field 1 数据寄存器	PCF1DR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	第 28 章
○	○	○	0008 B32Dh	SCI12	次 Control Field 1 数据寄存器	SCF1DR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B32Eh	SCI12	Control Field 1 比较允许寄存器	CF1CR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B32Fh	SCI12	Control Field 1 接收数据寄存器	CF1RR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B330h	SCI12	定时器的控制寄存器	TCR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B331h	SCI12	定时器的模式寄存器	TMR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B332h	SCI12	定时器的预分频器寄存器	TPRE	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 B333h	SCI12	定时器的计数器	TCNT	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C000h	PORT0	端口方向寄存器	PDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	第 19 章
○	○	○	0008 C001h	PORT1	端口方向寄存器	PDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C002h	PORT2	端口方向寄存器	PDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C003h	PORT3	端口方向寄存器	PDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C004h	PORT4	端口方向寄存器	PDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C005h	PORT5	端口方向寄存器	PDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C00Ah	PORTA	端口方向寄存器	PDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C00Bh	PORTB	端口方向寄存器	PDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C00Ch	PORTC	端口方向寄存器	PDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C00Dh	PORTD	端口方向寄存器	PDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C00Eh	PORTE	端口方向寄存器	PDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	第 19 章
○	○	○	0008 C011h	PORTH	端口方向寄存器	PDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C012h	PORTJ	端口方向寄存器	PDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C020h	PORT0	端口输出数据寄存器	PODR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C021h	PORT1	端口输出数据寄存器	PODR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C022h	PORT2	端口输出数据寄存器	PODR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C023h	PORT3	端口输出数据寄存器	PODR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C024h	PORT4	端口输出数据寄存器	PODR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C025h	PORT5	端口输出数据寄存器	PODR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C02Ah	PORTA	端口输出数据寄存器	PODR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C02Bh	PORTB	端口输出数据寄存器	PODR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C02Ch	PORTC	端口输出数据寄存器	PODR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C02Dh	PORTD	端口输出数据寄存器	PODR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C02Eh	PORTE	端口输出数据寄存器	PODR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C031h	PORTH	端口输出数据寄存器	PODR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C032h	PORTJ	端口输出数据寄存器	PODR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C040h	PORT0	端口输入数据寄存器	PIDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C041h	PORT1	端口输入数据寄存器	PIDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C042h	PORT2	端口输入数据寄存器	PIDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C043h	PORT3	端口输入数据寄存器	PIDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C044h	PORT4	端口输入数据寄存器	PIDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C045h	PORT5	端口输入数据寄存器	PIDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C04Ah	PORTA	端口输入数据寄存器	PIDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C04Bh	PORTB	端口输入数据寄存器	PIDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C04Ch	PORTC	端口输入数据寄存器	PIDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C04Dh	PORTD	端口输入数据寄存器	PIDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C04Eh	PORTE	端口输入数据寄存器	PIDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C051h	PORTH	端口输入数据寄存器	PIDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C052h	PORTJ	端口输入数据寄存器	PIDR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C060h	PORT0	端口模式寄存器	PMR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C061h	PORT1	端口模式寄存器	PMR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C062h	PORT2	端口模式寄存器	PMR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C063h	PORT3	端口模式寄存器	PMR	8	8	2 ~ 3 个 PCLKB	2 个 ICLK	

表 5.1 I/O 寄存器地址一览表 (18/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									ICLK ≥ PCLK 时	ICLK < PCLK 时	
○	○	○	0008 C064h	PORT4	端口模式寄存器	PMR	8	8	2~3 个 PCLKB	2 个 ICLK	第 19 章
○	○	○	0008 C065h	PORT5	端口模式寄存器	PMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C06Ah	PORTA	端口模式寄存器	PMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C06Bh	PORTB	端口模式寄存器	PMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C06Ch	PORTC	端口模式寄存器	PMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C06Dh	PORTD	端口模式寄存器	PMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C06Eh	PORTE	端口模式寄存器	PMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C071h	PORTH	端口模式寄存器	PMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C072h	PORTJ	端口模式寄存器	PMR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C082h	PORT1	漏极开路控制寄存器 0	ODR0	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C083h	PORT1	漏极开路控制寄存器 1	ODR1	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C084h	PORT2	漏极开路控制寄存器 0	ODR0	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C085h	PORT2	漏极开路控制寄存器 1	ODR1	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C086h	PORT3	漏极开路控制寄存器 0	ODR0	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C087h	PORT3	漏极开路控制寄存器 1	ODR1	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C094h	PORTA	漏极开路控制寄存器 0	ODR0	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C095h	PORTA	漏极开路控制寄存器 1	ODR1	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C096h	PORTB	漏极开路控制寄存器 0	ODR0	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C097h	PORTB	漏极开路控制寄存器 1	ODR1	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C098h	PORTC	漏极开路控制寄存器 0	ODR0	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C099h	PORTC	漏极开路控制寄存器 1	ODR1	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C09Ch	PORTE	漏极开路控制寄存器 0	ODR0	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C09Dh	PORTE	漏极开路控制寄存器 1	ODR1	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0C0h	PORT0	上拉控制寄存器	PCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0C1h	PORT1	上拉控制寄存器	PCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0C2h	PORT2	上拉控制寄存器	PCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0C3h	PORT3	上拉控制寄存器	PCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0C4h	PORT4	上拉控制寄存器	PCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0C5h	PORT5	上拉控制寄存器	PCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0CAh	PORTA	上拉控制寄存器	PCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0CBh	PORTB	上拉控制寄存器	PCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0CCh	PORTC	上拉控制寄存器	PCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C0CDh	PORTD	上拉控制寄存器	PCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0CEh	PORTE	上拉控制寄存器	PCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0D1h	PORTH	上拉控制寄存器	PCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C0D2h	PORTJ	上拉控制寄存器	PCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0E1h	PORT1	驱动能力控制寄存器	DSCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0E2h	PORT2	驱动能力控制寄存器	DSCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0E3h	PORT3	驱动能力控制寄存器	DSCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0E5h	PORT5	驱动能力控制寄存器	DSCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0EAh	PORTA	驱动能力控制寄存器	DSCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0EBh	PORTB	驱动能力控制寄存器	DSCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0ECh	PORTC	驱动能力控制寄存器	DSCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C0EDh	PORTD	驱动能力控制寄存器	DSCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0EEh	PORTE	驱动能力控制寄存器	DSCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C0F1h	PORTH	驱动能力控制寄存器	DSCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C0F2h	PORTJ	驱动能力控制寄存器	DSCR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	—	—	0008 C100h	MPC	CS 输出允许寄存器	PFCSE	8	8	2~3 个 PCLKB	2 个 ICLK	第 20 章
○	—	—	0008 C104h	MPC	地址输出允许寄存器 0	PFAOE0	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	—	—	0008 C105h	MPC	地址输出允许寄存器 1	PFAOE1	8	8、16	2~3 个 PCLKB	2 个 ICLK	
○	—	—	0008 C106h	MPC	外部总线控制寄存器 0	PFCR0	8	8、16	2~3 个 PCLKB	2 个 ICLK	

表 5.1 I/O 寄存器地址一览表 (19/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									ICLK ≥ PCLK 时	ICLK < PCLK 时	
○	—	—	0008 C107h	MPC	外部总线控制寄存器 1	PFBCR1	8	8、16	2~3 个 PCLKB	2 个 ICLK	第 20 章
○	○	○	0008 C11Fh	MPC	写保护寄存器	PWPR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C143h	MPC	P03 引脚功能控制寄存器	P03PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C145h	MPC	P05 引脚功能控制寄存器	P05PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C147h	MPC	P07 引脚功能控制寄存器	P07PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C14Ah	MPC	P12 引脚功能控制寄存器	P12PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C14Bh	MPC	P13 引脚功能控制寄存器	P13PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C14Ch	MPC	P14 引脚功能控制寄存器	P14PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C14Dh	MPC	P15 引脚功能控制寄存器	P15PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C14Eh	MPC	P16 引脚功能控制寄存器	P16PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C14Fh	MPC	P17 引脚功能控制寄存器	P17PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C150h	MPC	P20 引脚功能控制寄存器	P20PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C151h	MPC	P21 引脚功能控制寄存器	P21PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	—	—	0008 C152h	MPC	P22 引脚功能控制寄存器	P22PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	—	—	0008 C153h	MPC	P23 引脚功能控制寄存器	P23PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	—	—	0008 C154h	MPC	P24 引脚功能控制寄存器	P24PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	—	—	0008 C155h	MPC	P25 引脚功能控制寄存器	P25PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C156h	MPC	P26 引脚功能控制寄存器	P26PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C157h	MPC	P27 引脚功能控制寄存器	P27PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C158h	MPC	P30 引脚功能控制寄存器	P30PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C159h	MPC	P31 引脚功能控制寄存器	P31PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C15Ah	MPC	P32 引脚功能控制寄存器	P32PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	—	—	0008 C15Bh	MPC	P33 引脚功能控制寄存器	P33PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C15Ch	MPC	P34 引脚功能控制寄存器	P34PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C160h	MPC	P40 引脚功能控制寄存器	P40PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C161h	MPC	P41 引脚功能控制寄存器	P41PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C162h	MPC	P42 引脚功能控制寄存器	P42PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C163h	MPC	P43 引脚功能控制寄存器	P43PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C164h	MPC	P44 引脚功能控制寄存器	P44PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C165h	MPC	P45 引脚功能控制寄存器	P45PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C166h	MPC	P46 引脚功能控制寄存器	P46PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C167h	MPC	P47 引脚功能控制寄存器	P47PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C16Ch	MPC	P54 引脚功能控制寄存器	P54PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C16Dh	MPC	P55 引脚功能控制寄存器	P55PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C190h	MPC	PA0 引脚功能控制寄存器	PA0PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C191h	MPC	PA1 引脚功能控制寄存器	PA1PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C192h	MPC	PA2 引脚功能控制寄存器	PA2PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C193h	MPC	PA3 引脚功能控制寄存器	PA3PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C194h	MPC	PA4 引脚功能控制寄存器	PA4PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C195h	MPC	PA5 引脚功能控制寄存器	PA5PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C196h	MPC	PA6 引脚功能控制寄存器	PA6PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	—	—	0008 C197h	MPC	PA7 引脚功能控制寄存器	PA7PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C198h	MPC	PB0 引脚功能控制寄存器	PB0PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C199h	MPC	PB1 引脚功能控制寄存器	PB1PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C19Ah	MPC	PB2 引脚功能控制寄存器	PB2PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C19Bh	MPC	PB3 引脚功能控制寄存器	PB3PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C19Ch	MPC	PB4 引脚功能控制寄存器	PB4PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C19Dh	MPC	PB5 引脚功能控制寄存器	PB5PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C19Eh	MPC	PB6 引脚功能控制寄存器	PB6PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C19Fh	MPC	PB7 引脚功能控制寄存器	PB7PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	—	—	0008 C1A0h	MPC	PC0 引脚功能控制寄存器	PC0PFS	8	8	2~3 个 PCLKB	2 个 ICLK	

表 5.1 I/O 寄存器地址一览表 (20/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									ICLK ≥ PCLK 时	ICLK < PCLK 时	
○	—	—	0008 C1A1h	MPC	PC1 引脚功能控制寄存器	PC1PFS	8	8	2~3 个 PCLKB	2 个 ICLK	第 20 章
○	○	○	0008 C1A2h	MPC	PC2 引脚功能控制寄存器	PC2PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C1A3h	MPC	PC3 引脚功能控制寄存器	PC3PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C1A4h	MPC	PC4 引脚功能控制寄存器	PC4PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C1A5h	MPC	PC5 引脚功能控制寄存器	PC5PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C1A6h	MPC	PC6 引脚功能控制寄存器	PC6PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C1A7h	MPC	PC7 引脚功能控制寄存器	PC7PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C1A8h	MPC	PD0 引脚功能控制寄存器	PD0PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C1A9h	MPC	PD1 引脚功能控制寄存器	PD1PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C1AAh	MPC	PD2 引脚功能控制寄存器	PD2PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	—	—	0008 C1ABh	MPC	PD3 引脚功能控制寄存器	PD3PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	—	—	0008 C1ACh	MPC	PD4 引脚功能控制寄存器	PD4PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	—	—	0008 C1ADh	MPC	PD5 引脚功能控制寄存器	PD5PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	—	—	0008 C1AEh	MPC	PD6 引脚功能控制寄存器	PD6PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	—	—	0008 C1AFh	MPC	PD7 引脚功能控制寄存器	PD7PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C1B0h	MPC	PE0 引脚功能控制寄存器	PE0PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C1B1h	MPC	PE1 引脚功能控制寄存器	PE1PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C1B2h	MPC	PE2 引脚功能控制寄存器	PE2PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C1B3h	MPC	PE3 引脚功能控制寄存器	PE3PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C1B4h	MPC	PE4 引脚功能控制寄存器	PE4PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C1B5h	MPC	PE5 引脚功能控制寄存器	PE5PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	—	—	0008 C1B6h	MPC	PE6 引脚功能控制寄存器	PE6PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	—	—	0008 C1B7h	MPC	PE7 引脚功能控制寄存器	PE7PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C1C8h	MPC	PH0 引脚功能控制寄存器	PH0PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C1C9h	MPC	PH1 引脚功能控制寄存器	PH1PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C1CAh	MPC	PH2 引脚功能控制寄存器	PH2PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C1CBh	MPC	PH3 引脚功能控制寄存器	PH3PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	—	0008 C1D1h	MPC	PJ1 引脚功能控制寄存器	PJ1PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	—	—	0008 C1D3h	MPC	PJ3 引脚功能控制寄存器	PJ3PFS	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C280h	SYSTEM	深度待机控制寄存器	DPSBYCR	8	8	4~5 个 PCLKB	2~3 个 ICLKB	第 11 章
○	○	○	0008 C282h	SYSTEM	深度待机中断允许寄存器 0	DPSIER0	8	8	4~5 个 PCLKB	2~3 个 ICLKB	
○	○	○	0008 C284h	SYSTEM	深度待机中断允许寄存器 2	DPSIER2	8	8	4~5 个 PCLKB	2~3 个 ICLKB	
○	○	○	0008 C286h	SYSTEM	深度待机中断标志寄存器 0	DPSIFR0	8	8	4~5 个 PCLKB	2~3 个 ICLKB	
○	○	○	0008 C288h	SYSTEM	深度待机中断标志寄存器 2	DPSIFR2	8	8	4~5 个 PCLKB	2~3 个 ICLKB	
○	○	○	0008 C28Ah	SYSTEM	深度待机中断边沿寄存器 0	DPSIEGR0	8	8	4~5 个 PCLKB	2~3 个 ICLKB	
○	○	○	0008 C28Ch	SYSTEM	深度待机中断边沿寄存器 2	DPSIEGR2	8	8	4~5 个 PCLKB	2~3 个 ICLKB	
○	○	○	0008 C28Fh	SYSTEM	闪存 HOCO 软件待机控制寄存器	FHSSBYCR	8	8	4~5 个 PCLKB	2~3 个 ICLKB	
○	○	○	0008 C290h	SYSTEM	复位状态寄存器 0	RSTSR0	8	8	4~5 个 PCLKB	2~3 个 ICLKB	第 6 章
○	○	○	0008 C291h	SYSTEM	复位状态寄存器 1	RSTSR1	8	8	4~5 个 PCLKB	2~3 个 ICLKB	
○	○	○	0008 C293h	SYSTEM	主时钟振荡器的强制振荡控制寄存器	MOFCR	8	8	4~5 个 PCLKB	2~3 个 ICLKB	第 9 章
○	○	○	0008 C294h	SYSTEM	高速内部振荡器的电源控制寄存器	HOCOPCR	8	8	4~5 个 PCLKB	2~3 个 ICLKB	
○	○	○	0008 C296h	FLASH	闪存编程 / 擦除保护寄存器	FWEPOR	8	8	4~5 个 PCLKB	2~3 个 ICLKB	第 39 章、 40 章
○	○	○	0008 C297h	SYSTEM	电压监视电路 / 比较器 A 的控制寄存器	LVCMPCR	8	8	4~5 个 PCLKB	2~3 个 ICLKB	
○	○	○	0008 C298h	SYSTEM	电压检测电平选择寄存器	LVDLVL	8	8	4~5 个 PCLKB	2~3 个 ICLKB	第 8 章、 35 章
○	○	○	0008 C29Ah	SYSTEM	电压监视 1 电路 / 比较器 A1 的控制寄存器 0	LVD1CR0	8	8	4~5 个 PCLKB	2~3 个 ICLKB	
○	○	○	0008 C29Bh	SYSTEM	电压监视 2 电路 / 比较器 A2 的控制寄存器 0	LVD2CR0	8	8	4~5 个 PCLKB	2~3 个 ICLKB	
○	○	○	0008 C2A0h ~ 0008 C2BFh	SYSTEM	深度待机备用寄存器 0 ~ 31	DPSBKR0 ~ 31	8	8	4~5 个 PCLKB	2~3 个 ICLKB	第 11 章
○	○	○	0008 C400h	RTC	64Hz 计数器	R64CNT	8	8	2~3 个 PCLKB	2 个 ICLK	第 25 章
○	○	○	0008 C402h	RTC	秒计数器	RSECCNT	8	8	2~3 个 PCLKB	2 个 ICLK	

表 5.1 I/O 寄存器地址一览表 (21/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									ICLK ≥ PCLK 时	ICLK < PCLK 时	
○	○	○	0008 C404h	RTC	分钟计数器	RMINCNT	8	8	2~3 个 PCLKB	2 个 ICLK	第 25 章
○	○	○	0008 C406h	RTC	小时计数器	RHRCNT	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C408h	RTC	星期计数器	RWKCNT	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C40Ah	RTC	日计数器	RDAYCNT	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C40Ch	RTC	月计数器	RMONCNT	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C40Eh	RTC	年计数器	RYRCNT	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C410h	RTC	秒闹钟寄存器	RSECAR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C412h	RTC	分钟闹钟寄存器	RMINAR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C414h	RTC	小时闹钟寄存器	RHRAR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C416h	RTC	星期闹钟寄存器	RWKAR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C418h	RTC	日闹钟寄存器	RDAYAR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C41Ah	RTC	月闹钟寄存器	RMONAR	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C41Ch	RTC	年闹钟寄存器	RYRAR	16	16	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C41Eh	RTC	年闹钟允许寄存器	RYRAREN	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C422h	RTC	RTC 控制寄存器 1	RCR1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C424h	RTC	RTC 控制寄存器 2	RCR2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C426h	RTC	RTC 控制寄存器 3	RCR3	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C42Eh	RTC	时间误差校正寄存器	RADJ	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C440h	RTC	时间捕捉控制寄存器 0	RTCCR0	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C442h	RTC	时间捕捉控制寄存器 1	RTCCR1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C444h	RTC	时间捕捉控制寄存器 2	RTCCR2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C452h	RTC	秒捕捉寄存器 0	RSECCP0	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C454h	RTC	分捕捉寄存器 0	RMINCP0	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C456h	RTC	时捕捉寄存器 0	RHRCP0	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C45Ah	RTC	日捕捉寄存器 0	RDAYCP0	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C45Ch	RTC	月捕捉寄存器 0	RMONCP0	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C462h	RTC	秒捕捉寄存器 1	RSECCP1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C464h	RTC	分捕捉寄存器 1	RMINCP1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C466h	RTC	时捕捉寄存器 1	RHRCP1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C46Ah	RTC	日捕捉寄存器 1	RDAYCP1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C46Ch	RTC	月捕捉寄存器 1	RMONCP1	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C472h	RTC	秒捕捉寄存器 2	RSECCP2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C474h	RTC	分捕捉寄存器 2	RMINCP2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C476h	RTC	时捕捉寄存器 2	RHRCP2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C47Ah	RTC	日捕捉寄存器 2	RDAYCP2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C47Ch	RTC	月捕捉寄存器 2	RMONCP2	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C500h	TEMPS	温度传感器控制寄存器	TSCR	8	8	2~3 个 PCLKB	2 个 ICLK	第 34 章
○	○	○	0008 C580h	CMPB	比较器 B 控制寄存器 1	CPBCNT1	8	8	2~3 个 PCLKB	2 个 ICLK	第 36 章
○	○	○	0008 C582h	CMPB	比较器 B 标志寄存器	CPBFLG	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C583h	CMPB	比较器 B 中断控制寄存器	CPBINT	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	0008 C584h	CMPB	比较器 B 的滤波器选择寄存器	CPBF	8	8	2~3 个 PCLKB	2 个 ICLK	
○	○	○	007F C402h	FLASH	闪存模式寄存器	FMODR	8	8	2~3 个 FCLK	2 个 ICLK	第 39 章、 40 章
○	○	○	007F C410h	FLASH	闪存存取状态寄存器	FASTAT	8	8	2~3 个 FCLK	2 个 ICLK	
○	○	○	007F C411h	FLASH	闪存存取错误中断允许寄存器	FAEINT	8	8	2~3 个 FCLK	2 个 ICLK	
○	○	○	007F C412h	FLASH	闪存就绪中断允许寄存器	FRDYIE	8	8	2~3 个 FCLK	2 个 ICLK	第 40 章
○	○	○	007F C440h	FLASH	E2 数据闪存读允许寄存器 0	DFLRE0	16	16	2~3 个 FCLK	2 个 ICLK	
○	○	○	007F C450h	FLASH	E2 数据闪存编程 / 擦除允许寄存器 0	DFLWE0	16	16	2~3 个 FCLK	2 个 ICLK	
○	○	○	007F C454h	FLASH	FCU RAM 允许寄存器	FCURAME	16	16	2~3 个 FCLK	2 个 ICLK	第 39 章、 40 章
○	○	○	007F FFB0h	FLASH	闪存状态寄存器 0	FSTATR0	8	8	2~3 个 FCLK	2 个 ICLK	
○	○	○	007F FFB1h	FLASH	闪存状态寄存器 1	FSTATR1	8	8	2~3 个 FCLK	2 个 ICLK	
○	○	○	007F FFB2h	FLASH	闪存 P/E 模式入口寄存器	FENTRYR	16	16	2~3 个 FCLK	2 个 ICLK	第 39 章、 40 章

表 5.1 I/O 寄存器地址一览表 (22/22)

100 引脚	80 引脚	64 引脚	地址	模块符号	寄存器名	寄存器符号	位数	存取长度	存取周期数		参照章
									$ICLK \geq PCLK$ 时	$ICLK < PCLK$ 时	
○	○	○	007F FFB4h	FLASH	闪存保护寄存器	FPROTR	16	16	2~3 个 FCLK	2 个 ICLK	第 39 章
○	○	○	007F FFB6h	FLASH	闪存复位寄存器	FRESETR	16	16	2~3 个 FCLK	2 个 ICLK	
○	○	○	007F FFBAh	FLASH	FCU 命令寄存器	FCMDR	16	16	2~3 个 FCLK	2 个 ICLK	
○	○	○	007F FFC8h	FLASH	FCU 处理转换寄存器	FCPSR	16	16	2~3 个 FCLK	2 个 ICLK	
○	○	○	007F FFCAh	FLASH	E2 数据闪存空白检查控制寄存器	DFLBCCNT	16	16	2~3 个 FCLK	2 个 ICLK	第 40 章
○	○	○	007F FFCCh	FLASH	闪存 P/E 状态寄存器	FPESTAT	16	16	2~3 个 FCLK	2 个 ICLK	第 39 章、 40 章
○	○	○	007F FFCEh	FLASH	E2 数据闪存空白检查状态寄存器	DFLBCSTAT	16	16	2~3 个 FCLK	2 个 ICLK	第 40 章
○	○	○	007F FFE8h	FLASH	外围时钟通知寄存器	PCKAR	16	16	2~3 个 FCLK	2 个 ICLK	第 39 章、 40 章

6. 复位

6.1 概要

复位有引脚复位、上电复位、电压监视 0 复位、电压监视 1 复位、电压监视 2 复位、深度软件待机复位、独立看门狗定时器复位、看门狗定时器复位和软件复位。

复位名称和复位源如表 6.1 所示。

表 6.1 复位名称和复位源

复位名称	复位源
引脚复位	RES# 引脚的输入电压为 Low 电平。
上电复位	VCC 上升 (监视电压: VPOR) (注 1)
电压监视 0 复位	VCC 下降 (监视电压: Vdet0)
电压监视 1 复位	VCC 下降 (监视电压: Vdet1) (注 1)
电压监视 2 复位	VCC 下降 (监视电压: Vdet2) (注 1)
深度软件待机复位	通过中断解除深度软件待机模式。
独立看门狗定时器复位	独立看门狗定时器发生下溢或者刷新错误。
看门狗定时器复位	看门狗定时器发生下溢或者刷新错误。
软件复位	设定寄存器

注. 有关监视电压 (VPOR、Vdet0、Vdet1、Vdet2)，请参照“8. 电压检测电路 (LVDAa)”和“41. 电特性”。

通过复位对内部状态进行初始化，引脚为初始状态。

通过各复位进行初始化的对象如表 6.2 所示。

表 6.2 各种复位的初始化对象

复位对象	复位源								
	引脚复位	上电复位	电压监视 0 复位	独立看门狗定时器复位	看门狗定时器复位	电压监视 1 复位	电压监视 2 复位	深度软件待机复位	软件复位
上电复位检测标志 (RSTSR0.PORF)	○	—	—	—	—	—	—	—	—
冷启动 / 热启动判断标志 (RSTSR1.CWSF)	—	○	—	—	—	—	—	—	—
电压监视 0 复位检测标志 (RSTSR0.LVD0RF)	○	○	—	—	—	—	—	—	—
独立看门狗定时器复位检测标志 (RSTSR2.IWDTRF)	○	○	○	—	—	—	—	○	—
独立看门狗定时器的寄存器 (IWDTRR、IWDTCR、IWDTSR、IWDTRCR、IWDTCSTPR、ILOCOCR)	○	○	○	—	—	—	—	○	—
看门狗定时器复位检测标志 (RSTSR2.WDTRF)	○	○	○	○	—	—	—	○	—
看门狗定时器的寄存器 (WDTRR、WDTCSR、WDTSR、WDTRCR)	○	○	○	○	—	—	—	○	—
电压监视 1 复位检测标志 (RSTSR0.LVD1RF)	○	○	○	○	○	—	—	—	—
电压监视功能 1 的寄存器 (LVD1CR0、LVCMPPCR.EXVREFINP1、EXVCCINP1、LVD1E、LVDLVLR.LVD1LVL)	○	○	○	○	○	—	—	—	—
(LVD1CR1、LVD1SR)	○	○	○	○	○	—	—	○	—
电压监视 2 复位检测标志 (RSTSR0.LVD2RF)	○	○	○	○	○	○	—	—	—
电压监视功能 2 的寄存器 (LVD2CR0、LVCMPPCR.EXVREFINP2、EXVCCINP2、LVD2E、LVDLVLR.LVD2LVL)	○	○	○	○	○	○	—	—	—
(LVD2CR1、LVD2SR)	○	○	○	○	○	○	—	○	—
深度软件待机复位检测标志 (RSTSR0.DPSRSTF)	○	○	○	○	○	○	○	—	—
低功耗功能的相关寄存器 (DPSBYCR、DPSIER0、DPSIER2、DPSIFR0、DPSIFR2、DPSIEGR0、DPSIEGR2、FHSSBYCR)	○	○	○	○	○	○	○	—	○
高速内部振荡器的相关寄存器 (HOCOPCR.HOCOPCNT)	○	○	○	○	○	○	○	—	○
引脚状态	○	○	○	○	○	○	○	—	○
软件复位检测标志 (RSTSR2.SWRF)	○	○	○	○	○	○	○	○	—
实时时钟的寄存器 (注 1)	—	—	—	—	—	—	—	—	—
上述以外的寄存器、CPU 以及内部状态	○	○	○	○	○	○	○	○	○

○：被初始化、—：不变化。

注 1. 部分控制位 (RCR1.CIE、RCR2.RTCOE、ADJ30、RESET) 能通过全部复位被初始化。有关对象控制位，请参照“25. 实时时钟 (RTCb)”。

如果解除复位，就开始复位异常处理。有关复位异常处理，请参照“13. 异常处理”。

复位的相关输入 / 输出引脚如表 6.3 所示。

表 6.3 复位的相关输入 / 输出引脚

引脚名	输入 / 输出	功能
RES#	输入	复位引脚

6.2 寄存器说明

6.2.1 复位状态寄存器 0 (RSTSR0)

地址 0008 C290h

	b7	b6	b5	b4	b3	b2	b1	b0
	DPSRSTF	—	—	—	LVD2RF	LVD1RF	LVD0RF	PORF
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	PORF	上电复位检测标志	0: 未检测到上电复位 1: 检测到上电复位	R/W (注 1)
b1	LVD0RF	电压监视 0 复位检测标志	0: 未检测到电压监视 0 复位 1: 检测到电压监视 0 复位	R/W (注 1)
b2	LVD1RF	电压监视 1 复位检测标志	0: 未检测到电压监视 1 复位 1: 检测到电压监视 1 复位	R/W (注 1)
b3	LVD2RF	电压监视 2 复位检测标志	0: 未检测到电压监视 2 复位 1: 检测到电压监视 2 复位	R/W (注 1)
b6-b4	—	保留位	读写值都为“0”。	R/W
b7	DPSRSTF	深度软件待机复位检测标志	0: 无通过中断产生的深度软件待机模式解除请求 1: 有通过中断产生的深度软件待机模式检测请求	R/W (注 1)

注 1. 只能写为了清除标志的“0”。

PORF 标志 (上电复位检测标志)

此标志表示发生上电复位。

[为“1”的条件]

- 发生上电复位时

[为“0”的条件]

- 进行表6.2所示的复位时
- 读“1”后写“0”时

LVD0RF 标志 (电压监视 0 复位检测标志)

此标志表示检测到 VCC 电压低于 Vdet0 电平。

[为“1”的条件]

- 检测到 Vdet0 电平的 VCC 电压时

[为“0”的条件]

- 进行表6.2所示的复位时
- 读“1”后写“0”时

LVD1RF 标志（电压监视 1 复位检测标志）

此标志表示检测到 VCC 电压低于 Vdet1 电平。

[为“1”的条件]

- 检测到 Vdet1 电平的 VCC 电压时

[为“0”的条件]

- 进行表 6.2 所示的复位时
- 读“1”后写“0”时

LVD2RF 标志（电压监视 2 复位检测标志）

此标志表示检测到 VCC 电压低于 Vdet2 电平。

[为“1”的条件]

- 检测到 Vdet2 电平的 VCC 电压时

[为“0”的条件]

- 进行表 6.2 所示的复位时
- 读“1”后写“0”时

DPSRSTF 标志（深度软件待机复位检测标志）

此标志表示在通过中断解除深度软件待机模式的同时发生内部复位（深度软件待机复位）。

[为“1”的条件]

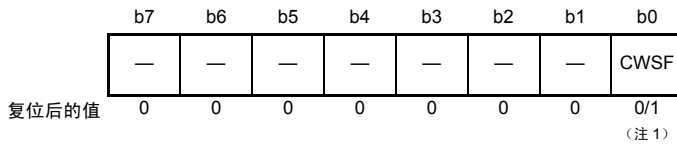
- 通过中断解除深度软件待机模式时
详细内容请参照“11. 低功耗功能”。

[为“0”的条件]

- 进行表 6.2 所示的复位时
- 读“1”后写“0”时

6.2.2 复位状态寄存器 1 (RSTSR1)

地址 0008 C291h



注 1. 复位后的值因复位源而不同。

位	符号	位名	功能	R/W
b0	CWSF	冷启动 / 热启动判断标志	0: 冷启动 1: 热启动	R/W (注1)
b7-b1	—	保留位	读写值都为“0”。	R/W

注 1. 只能写为了清除标志的“0”。

RSTSR1 寄存器判断是接通电源时的复位处理（冷启动）还是运行过程中输入复位信号时的复位处理（热启动）。

CWSF 标志（冷启动 / 热启动判断标志）

此标志表示是冷启动或者热启动。

能通过上电复位对 CWSF 标志进行初始化，不能通过 RES# 引脚复位信号对 CWSF 标志进行初始化。

[为“1”的条件]

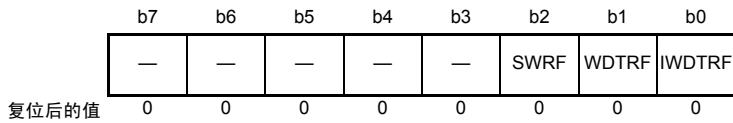
- 通过程序写“1”时
即使写“0”也不发生变化

[为“0”的条件]

- 进行表 6.2 所示的复位时

6.2.3 复位状态寄存器 2 (RSTSR2)

地址 0008 00C0h



位	符号	位名	功能	R/W
b0	IWDTRF	独立看门狗定时器复位检测标志	0: 未检测到独立看门狗定时器复位 1: 检测到独立看门狗定时器复位	R/W (注1)
b1	WDTRF	看门狗定时器复位检测标志	0: 未检测到看门狗定时器复位 1: 检测到看门狗定时器复位	R/W (注1)
b2	SWRF	软件复位检测标志	0: 未检测到软件复位 1: 检测到软件复位	R/W (注1)
b7-b3	—	保留位	读写值都为“0”。	R/W

注 1. 只能写为了清除标志的“0”。

IWDTRF 标志 (独立看门狗定时器复位检测标志)

此标志表示发生独立看门狗定时器复位。

[为“1”的条件]

- 发生独立看门狗定时器复位时

[为“0”的条件]

- 进行表6.2所示的复位时
- 读“1”后写“0”时

WDTRF 标志 (看门狗定时器复位检测标志)

此标志表示发生看门狗定时器复位。

[为“1”的条件]

- 发生看门狗定时器复位时

[为“0”的条件]

- 进行表6.2所示的复位时
- 读“1”后写“0”时

SWRF 标志 (软件复位检测标志)

此标志表示发生软件复位。

[为“1”的条件]

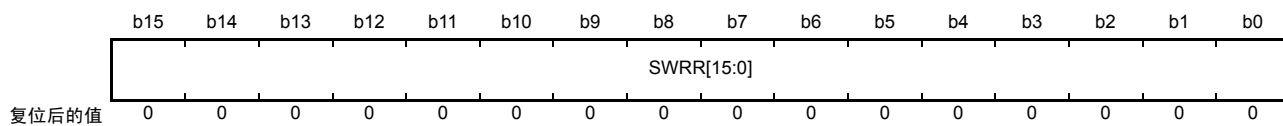
- 未发生软件复位时

[为“0”的条件]

- 进行表6.2所示的复位时
- 读“1”后写“0”时

6.2.4 软件复位寄存器 (SWRR)

地址 0008 00C2h



位	符号	位名	功能	R/W
b15-b3	SWRR[15:0]	软件复位位	如果写“A501h”，LSI 就复位。读取值为“0000h”。	R/W

6.3 运行说明

6.3.1 引脚复位

这是 RES# 引脚引起的复位。

当 RES# 引脚为 Low 电平时，停止正在执行的全部处理，进入复位状态。

为了确保复位，必须在接通电源时根据规定的振荡稳定时间，使 RES# 引脚保持 Low 电平。

在 RES# 引脚从 Low 电平变为 High 电平后，如果经过 RES# 解除后的待机时间 (tRESWT)，就解除内部复位，并且 CPU 开始复位异常处理。

详细内容请参照“41. 电特性”。

6.3.2 上电复位、电压监视 0 复位

上电复位是上电复位电路引起的内部复位。

如果在将 RES# 引脚置为 High 电平的状态下接通电源，就发生上电复位。如果 VCC 高于 VPOR，就在经过一定的时间（上电复位时间）后解除上电复位，并且 CPU 开始复位异常处理。上电复位时间是使外部电源和 LSI 稳定的时间。

如果发生上电复位，RSTSR0.PORF 标志就变为“1”。通过引脚复位对 PORF 标志进行初始化。

电压监视 0 复位是电压监视电路引起的内部复位。

如果在选项功能选择寄存器 1 (OFS1) 的电压检测 0 电平选择位 (LVDAS) 为“0”（复位后，电压监视 0 复位有效）的状态下，VCC 下降到小等于 Vdet0，RSTSR0.LVD0RF 标志就变为“1”，并且电压检测电路发生电压监视 0 复位。如果使用电压监视 0 复位，必须将 OFS1.LVDAS 位置“0”。如果 VCC 上升到大于 Vdet0，就在经过 LVD0 复位时间 (tLVD0) 后解除内部复位，并且 CPU 开始复位异常处理。

上电复位和电压监视 0 复位的运行例子如图 6.1 所示。

电压监视 0 复位的详细内容请参照“8. 电压检测电路 (LVDAa)”。

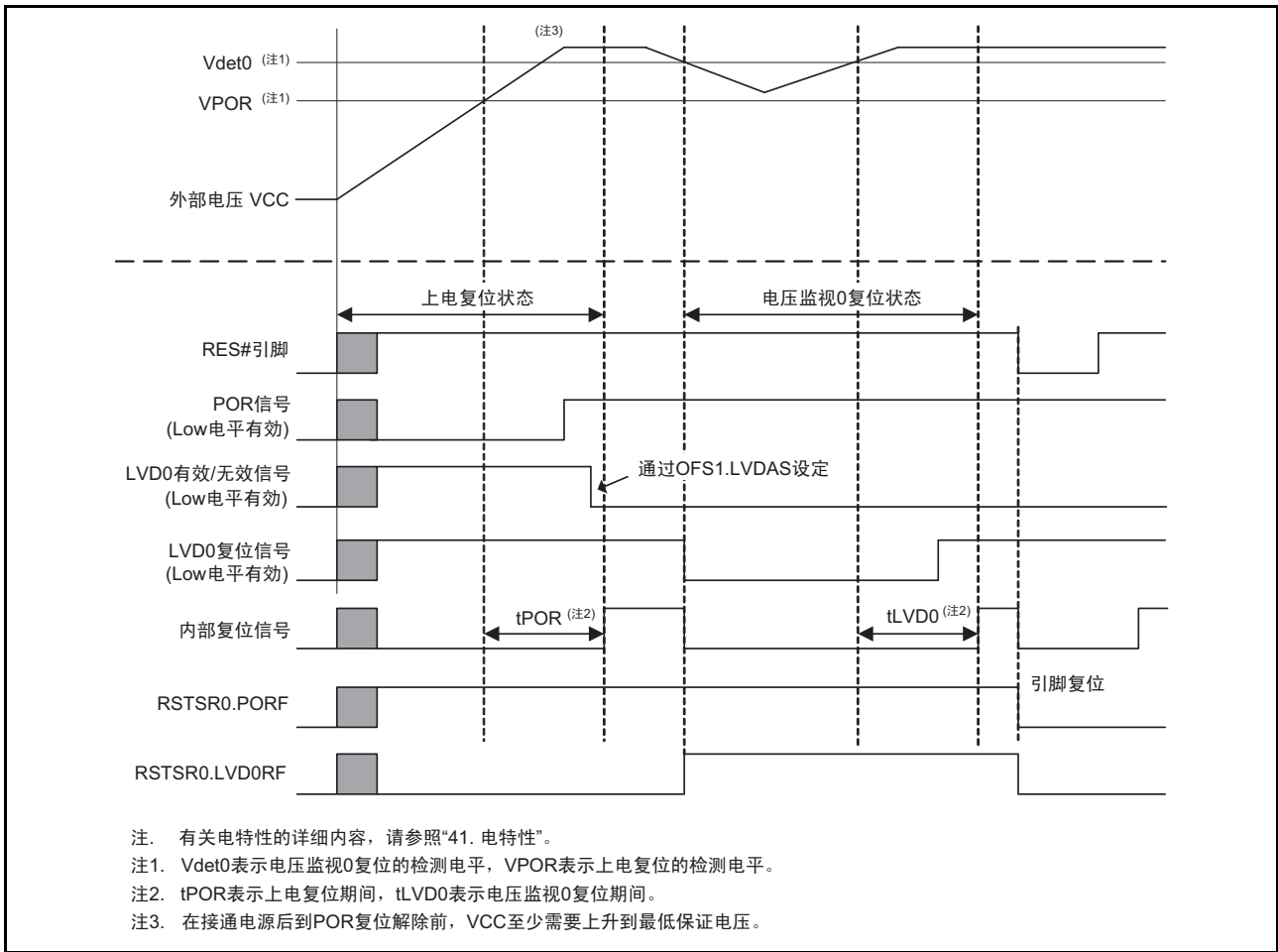


图 6.1 上电复位的运行

6.3.3 电压监视 1 复位、电压监视 2 复位

这是电压检测电路引起的内部复位。

在电压监视 1 电路控制寄存器 0 (LVD1CR0) 的电压监视 1 中断 / 复位允许位 (LVD1RIE) 为“1” (通过电压检测电路引起的复位 / 中断有效), 并且电压监视 1 电路模式选择位 (LVD1RI) 为“1” (在低电压检测时发生复位) 的状态下, 如果 VCC 下降到小于等于 Vdet1, RSTSR0.LVD1RF 标志就变为“1”, 并且电压检测电路发生电压监视 1 复位。

同样, 在电压监视 2 电路控制寄存器 0 (LVD2CR0) 的电压监视 2 中断 / 复位允许位 (LVD2RIE) 为“1” (通过电压检测电路引起的复位 / 中断有效), 并且电压监视 2 电路模式选择位 (LVD2RI) 为“1” (在低电压检测时发生复位) 的状态下, 如果 VCC 下降到小于等于 Vdet2, RSTSR0.LVD2RF 标志就变为“1”, 并且电压检测电路发生电压监视 2 复位。

能通过 LVD1CR0 寄存器的电压监视 1 复位无效选择位 (LVD1RN) 选择电压监视 1 复位的解除时序。在 LVD1CR0.LVD1RN 位为“0”时, 如果 VCC 先下降到小于等于 Vdet1, 然后再从上升到大于 Vdet1 开始又经过 LVD1 复位时间 (tLVD1) 后, 就解除内部复位, 并且 CPU 开始复位异常处理。另外, 在 LVD1CR0.LVD1RN 位为“1”时, 如果 VCC 从下降到小于等于 Vdet1 后开始又经过 LVD1 复位时间 (tLVD1), 就解除内部复位, 并且 CPU 开始复位异常处理。

同样, 能通过设定 LVD2CR0 寄存器的电压监视 2 复位无效选择位 (LVD2RN) 选择电压监视 2 复位的解除时序。

能通过设定电压检测电平选择寄存器 (LVDLVLR) 更改 Vdet1 和 Vdet2 的电压检测电平。

电压监视 1 复位和电压监视 2 复位的运行例子如图 6.2 所示。

电压监视 1 复位和电压监视 2 复位的详细内容请参照“8. 电压检测电路 (LVDAa)”。

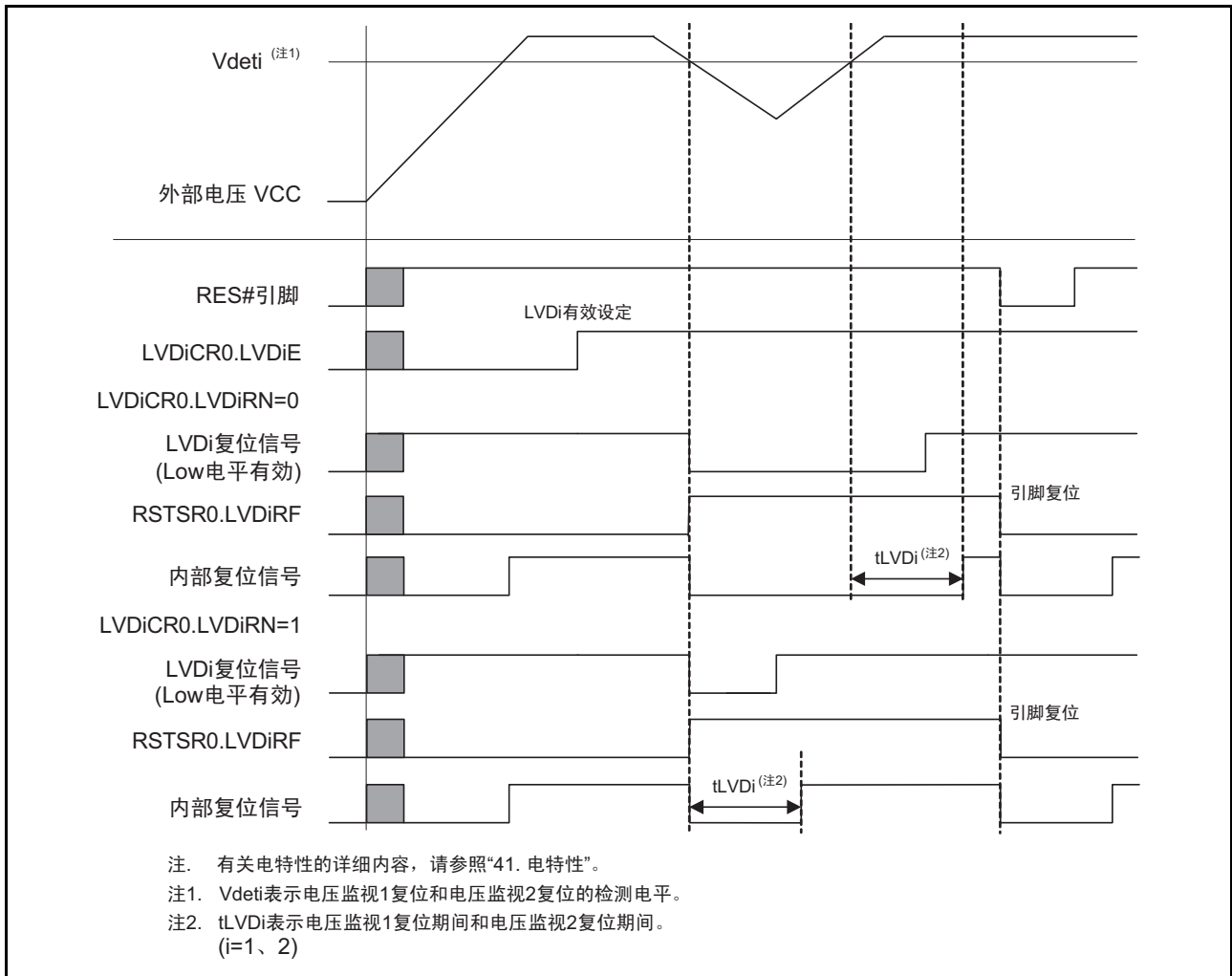


图 6.2 电压监视 1 复位和电压监视 2 复位的运行例子

6.3.4 深度软件待机复位

这是通过中断来解除深度软件待机模式时发生的内部复位。

如果产生深度软件待机模式的解除源, 就发生深度软件待机复位。此后, 在经过深度软件待机解除后的复位时间 (tDSBY) 后, 解除深度软件待机复位。在解除深度软件待机复位的同时, 深度软件待机模式也随之解除。

在解除深度软件待机复位模式后, 如果经过深度软件待机模式解除后的待机时间 (tDSBYWT), 就解除内部复位, 并且 CPU 开始复位异常处理。

深度软件待机复位的详细内容请参照“11. 低功耗功能”。

6.3.5 独立看门狗定时器复位

这是独立看门狗定时器引起的内部复位。

能通过设定 IWDTRCR 复位控制寄存器（IWDTRCR）或者选项功能选择寄存器 0（OFS0），选择是否从独立看门狗定时器输出独立看门狗定时器复位。

在选择输出独立看门狗定时器复位时，如果独立看门狗定时器发生下溢，或者在非刷新允许期间进行写操作，就发生独立看门狗定时器复位。如果在发生独立看门狗定时器复位后，又经过内部复位时间（tRESW2），就解除内部复位，并且 CPU 开始复位异常处理。

独立看门狗定时器复位的详细内容请参照“27. 独立看门狗定时器（IWDtA）”。

6.3.6 看门狗定时器复位

这是看门狗定时器引起的内部复位。

能通过设定 WDT 复位控制寄存器（WDTRCR）或者选项功能选择寄存器 0（OFS0），选择是否从看门狗定时器输出看门狗定时器复位。

在选择输出看门狗定时器复位时，如果看门狗定时器发生下溢，或者在非刷新允许期间进行写操作，就发生看门狗定时器复位。如果在发生看门狗定时器复位后，又经过内部复位时间（tRESW2），就解除内部复位，并且 CPU 开始复位异常处理。

看门狗定时器复位的详细内容请参照“26. 看门狗定时器（WDtA）”。

6.3.7 软件复位

这是软件复位电路引起的内部复位。

如果给 SWRR 寄存器写“A501h”，就发生软件复位。如果在发生软件复位后，又经过内部复位时间（tRESW2），就解除内部复位，并且 CPU 开始异常复位处理。

6.3.8 冷启动 / 热启动判断功能

这是通过 RSTSR1.CWSF 标志判断是接通电源时的复位处理（冷启动）还是运行过程中输入复位信号时的复位处理（热启动）的功能。

如果发生上电复位，RSTSR1.CWSF 标志就变为“0”（冷启动）。即使进行其他复位此标志也不变为“0”。另外，如果通过程序写“1”，此标志就变为“1”。即使写“0”也不发生变化。

冷启动 / 热启动判断功能的运行例子如图 6.3 所示。

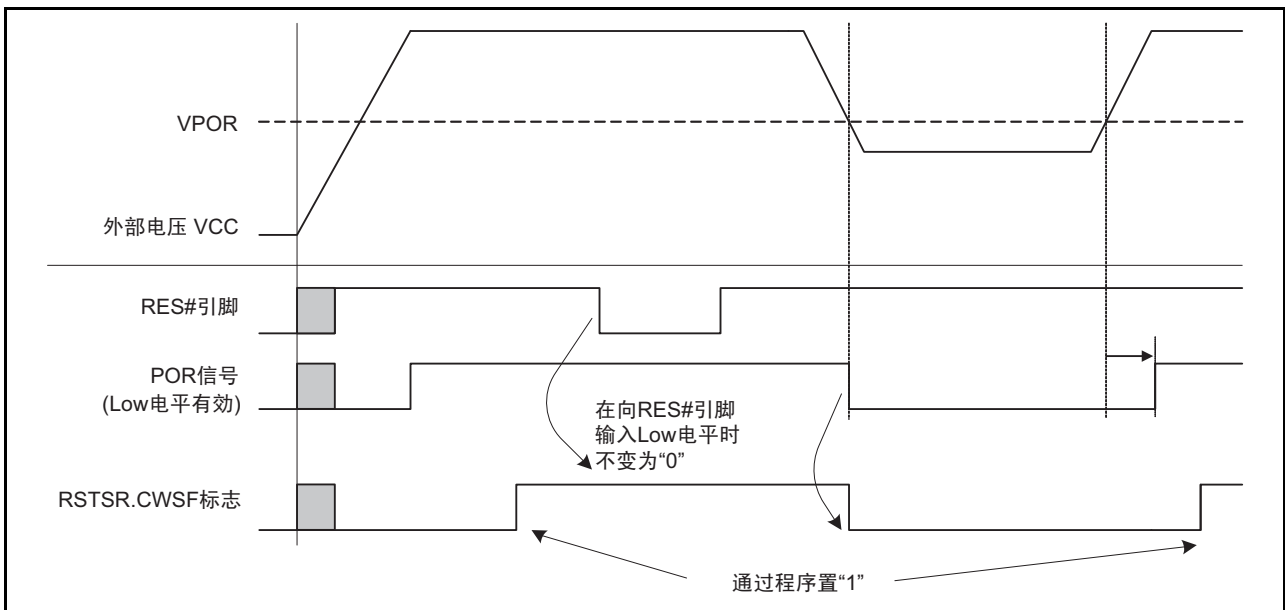


图 6.3 冷启动 / 热启动判断功能的运行例子

6.3.9 复位发生源的判断

能通过读 RSTSR0 寄存器和 RSTSR2 寄存器，确认是因发生哪种复位而执行了复位异常处理。
 复位发生源的判断流程例子如图 6.4 所示。

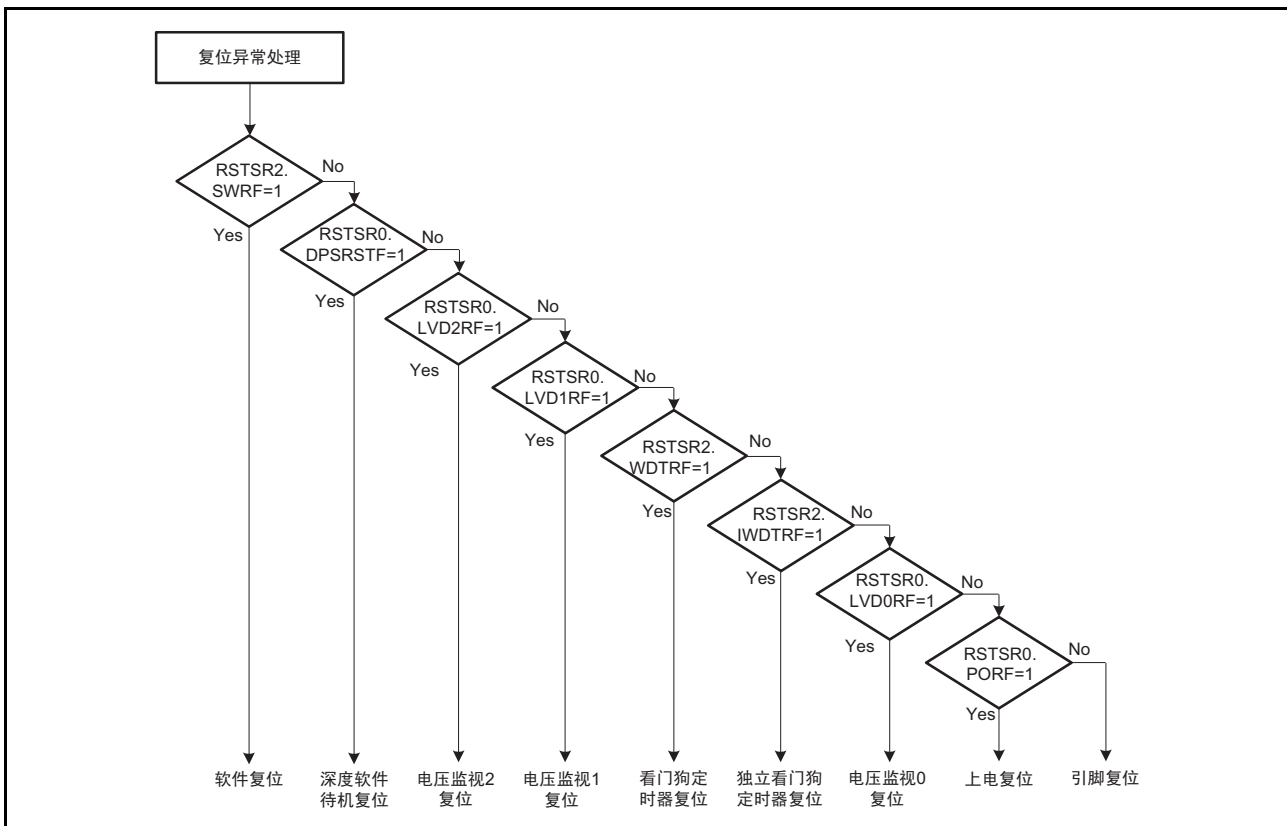


图 6.4 复位发生源的判断流程例子

7. 选项设定存储器

7.1 概要

选项设定存储器在 ROM 内，有选择复位后的单片机状态的寄存器。

选项设定存储区如图 7.1 所示。

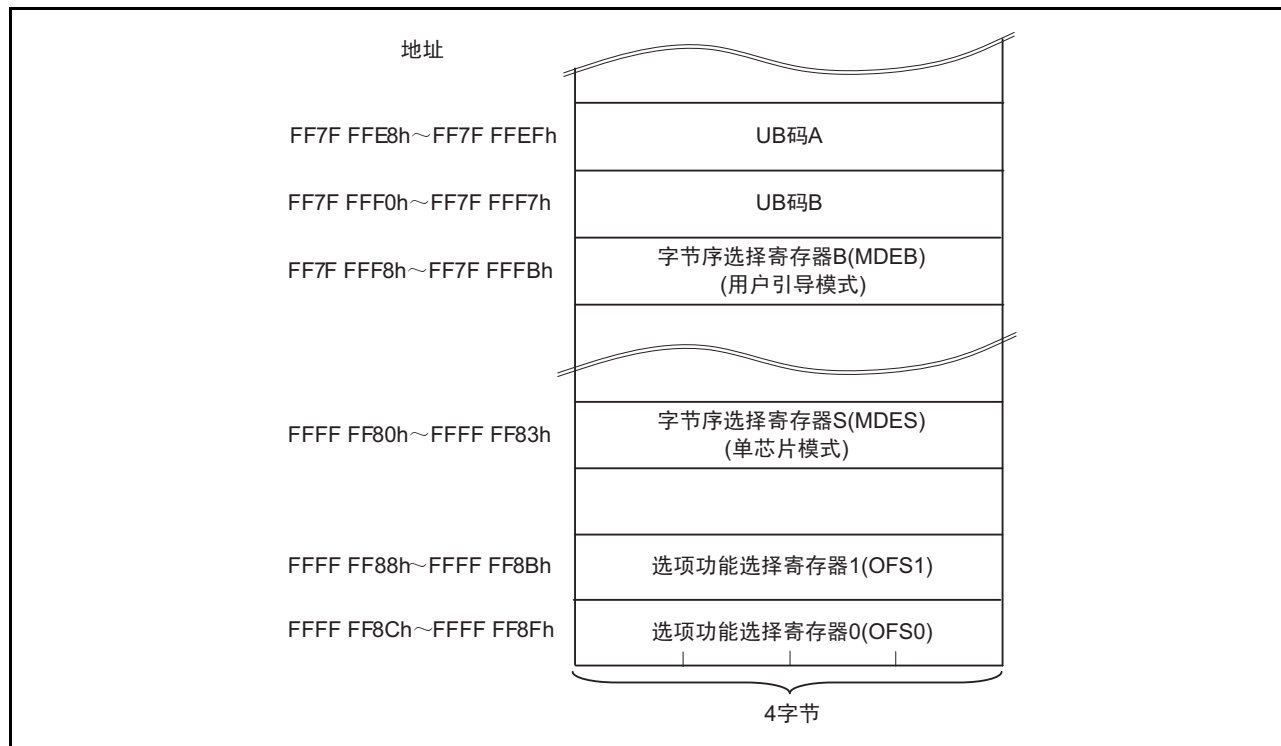


图 7.1 选项设定存储区

7.2 寄存器说明

7.2.1 选项功能选择寄存器 0 (OFS0)

地址 FFFF FF8Ch

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
—	—	—	IWDRST IRQS	WDTRPSS[1:0]	WDTRPES[1:0]	WDTCKS[3:0]			WDTTOPS[1:0]			WDTST RT	—		
复位后的值 用户的设定值 (注1)															
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	IWDTSL CSTP	—	IWDRS TIRQS	WDTRPSS[1:0]	WDTRPES[1:0]	IWDCKS[3:0]			IWDTTOPS[1:0]			IWDTST RT	—		
复位后的值 用户的设定值 (注1)															

注 1. 空白产品为“FFFF FFFFh”，在用户编程后为编程后的值。

位	符号	位名	功能	R/W
b0	—	保留位	读取值为编程后的值。在编程时，必须置“1”。	R
b1	IWDTSTRT	IWDT 启动模式选择位	0: 复位后，IWDT 在自动启动模式中自动启动。 1: 复位后，IWDT 为停止状态。	R
b3-b2	IWDTTOPS[1:0]	IWDT 超时期间选择位	b3 b2 0 0: 1024 个周期 (03FFh) 0 1: 4096 个周期 (0FFFh) 1 0: 8192 个周期 (1FFFh) 1 1: 16384 个周期 (3FFFh)	R
b7-b4	IWDCKS[3:0]	IWDT 时钟频比选择位	b7 b4 0 0 0 0: 1 分频 (周期为 131ms) 0 0 1 0: 16 分频 (周期为 2.10s) 0 0 1 1: 32 分频 (周期为 4.19s) 0 1 0 0: 64 分频 (周期为 8.39s) 1 1 1 1: 128 分频 (周期为 16.8s) 0 1 0 1: 256 分频 (周期为 33.6s) 上述以外: 不能设定	R
b9-b8	IWDTRPES[1:0]	IWDT 窗口结束位置选择位	b9 b8 0 0: 75% 0 1: 50% 1 0: 25% 1 1: 0% (没有设定窗口结束位置)	R
b11-b10	IWDTRPSS[1:0]	IWDT 窗口开始位置选择位	b11 b10 0 0: 25% 0 1: 50% 1 0: 75% 1 1: 100% (没有设定窗口开始位置)	R
b12	IWDRSTIRQS	IWDT 复位中断请求选择位	0: 允许非屏蔽中断请求 1: 允许复位	R
b13	—	保留位	读取值为编程后的值。在编程时，必须置“1”。	R
b14	IWDTSLCSTP	IWDT 睡眠模式计数停止控制位	0: 计数停止无效 1: 在向睡眠模式、软件待机模式、深度软件待机模式和全模块时钟停止模式转移时，计数停止有效。	R
b16-b15	—	保留位	读取值为编程后的值。在编程时，必须置“1”。	R
b17	WDTSTRT	WDT 启动模式选择位	0: 复位后，WDT 在自动启动模式中自动启动。 1: 复位后，WDT 为停止状态。	R

位	符号	位名	功能	R/W
b19-b18	WDTTOPS[1:0]	WDT 超时期间选择位	b19 b18 0 0: 1024 个周期 (03FFh) 0 1: 4096 个周期 (0FFFh) 1 0: 8192 个周期 (1FFFh) 1 1: 16384 个周期 (3FFFh)	R
b23-b20	WDTCKS[3:0]	WDT 时钟分频比选择位	b23 b20 0 0 0 1: PCLK/4 0 1 0 0: PCLK/64 1 1 1 1: PCLK/128 0 1 1 0: PCLK/512 0 1 1 1: PCLK/2048 1 0 0 0: PCLK/8192 上述以外: 不能设定	R
b25-b24	WDRPES[1:0]	WDT 窗口结束位置选择位	b25 b24 0 0: 75% 0 1: 50% 1 0: 25% 1 1: 0% (没有设定窗口结束位置)	R
b27-b26	WDRPSS[1:0]	WDT 窗口开始位置选择位	b27 b26 0 0: 25% 0 1: 50% 1 0: 75% 1 1: 100% (没有设定窗口开始位置)	R
b28	WDRSTIRQS	WDT 复位中断请求选择位	0: 允许非屏蔽中断请求 1: 允许复位	R
b31-29	—	保留位	读取值为编程后的值。在编程时, 必须置“1”。	R

OFS0 寄存器是选择复位后独立看门狗定时器 (IWDT) 和看门狗定时器 (WDT) 的运行的寄存器。

OFS0 寄存器在 ROM 内, 必须和程序一起写, 但是在写后不能对 OFS0 寄存器进行追加写。

如果擦除包括 OFS0 寄存器的块, OFS0 寄存器变为“FFFF FFFFh”。

在用户引导模式中, OFS0 寄存器的设定无效, 为“FFFF FFFFh”。

IWDTSTRT 位 (IWDT 启动模式选择位)

能选择复位后的 IWDT 启动模式 (停止状态或者在自动启动模式中的启动)。在自动启动模式中启动时, IWDT 的设定为 OFS0 寄存器的设定有效。

IWDTTOPS[1:0] 位 (IWDT 超时期间选择位)

以 IWDTCKS[3:0] 位设定的分频时钟为 1 个周期, 从 1024 个周期、4096 个周期、8192 个周期和 16384 个周期中选择递减计数器发生下溢为止的超时期间。

从刷新后到发生下溢为止的时间 (IWDT 的 LOCO 时钟数) 取决于 IWDTCKS[3:0] 位和 IWDTTOPS[1:0] 位的组合。

详细内容请参照“27. 独立看门狗定时器 (IWDTa) ”。

IWDTCKS[3:0] 位 (IWDT 时钟分频比选择位)

对 IWDT 的 LOCO 时钟进行分频的预分频器的分频比可从 1 分频、16 分频、32 分频、64 分频、128 分频和 256 分频中选择。能通过和 IWDTTOPS[1:0] 位的组合，将 IWDT 的计数期间设定在 IWDT 的 LOCO 时钟的 1024 ~ 4194304 个时钟之间。

详细内容请参照“27. 独立看门狗定时器 (IWDTa)”。

IWDRPES[1:0] 位 (IWDT 窗口结束位置选择位)

递减计数器的窗口结束位置可从计数期间的 75%、50%、25% 和 0% 中选择。选择的窗口结束位置必须小于窗口开始位置的值 (窗口开始位置 > 窗口结束位置)。如果将窗口结束位置设定为大于窗口开始位置的值，就只有窗口开始位置的设定有效。

IWDRPSS[1:0] 位和 IWDRPES[1:0] 位设定的窗口开始 / 结束位置的计数器值因 IWDTTOPS[1:0] 位的设定而不同。

详细内容请参照“27. 独立看门狗定时器 (IWDTa)”。

IWDRPSS[1:0] 位 (IWDT 窗口开始位置选择位)

递减计数器的窗口开始位置可从计数期间 (开始计数时为 100%，发生下溢时为 0%) 的 100%、75%、50% 和 25% 中选择。从窗口开始位置到窗口结束位置的期间为刷新允许期间，其他期间为刷新禁止期间。

详细内容请参照“27. 独立看门狗定时器 (IWDTa)”。

IWDRSTIRQS 位 (IWDT 复位中断请求选择位)

此位设定递减计数器发生下溢或者刷新错误时的运行。能选择独立看门狗定时器复位或者非屏蔽中断请求。

详细内容请参照“27. 独立看门狗定时器 (IWDTa)”。

IWDTSLCSTP 位 (IWDT 睡眠模式计数停止控制位)

此位选择向睡眠模式、软件待机模式、深度软件待机模式和全模块时钟停止模式转移时的计数停止。

详细内容请参照“27. 独立看门狗定时器 (IWDTa)”。

WDTSTRT 位 (WDT 启动模式选择位)

此位能选择复位后的 WDT 启动模式 (停止状态或者在自动启动模式中的启动)。在自动启动模式中启动时，WDT 的设定为 OFS0 寄存器的设定有效。

WDTTOPS[1:0] 位 (WDT 超时期间选择位)

以 WDTCKS[3:0] 位设定的分频时钟为 1 个周期，从 1024 个周期、4096 个周期、8192 个周期和 16384 个周期中选择递减计数器发生下溢为止的超时期间。

从刷新后到发生下溢为止的时间 (PCLK) 取决于 WDTCKS[3:0] 位和 WDTTOPS[1:0] 位的组合。

详细内容请参照“26. 看门狗定时器 (WDTA)”。

WDTCKS[3:0] 位 (WDT 时钟分频比选择位)

对 PCLK 进行分频的预分频器的分频比可从 4 分频、64 分频、128 分频、512 分频、2048 分频和 8192 分频中选择。能通过和 WDTTOPS[1:0] 位的组合，将 WDT 的计数期间设定在 PCLK 的 4096 ~ 134217728 个时钟之间。

详细内容请参照“26. 看门狗定时器 (WDTA)”。

WDTRPES[1:0] 位 (WDT 窗口结束位置选择位)

递减计数器的窗口结束位置可从计数期间的 75%、50%、25% 和 0% 中选择。选择的窗口结束位置必须小于窗口开始位置的值 (窗口开始位置 > 窗口结束位置)。如果将窗口结束位置设定为大于窗口开始位置的值, 就只有窗口开始位置的设定有效。

WDTRPSS[1:0] 位和 WDTRPES[1:0] 位设定的窗口开始 / 结束位置的计数器值因 WDTTOPS[1:0] 位的设定而不同。

详细内容请参照“26. 看门狗定时器 (WDTA) ”。

WDTRPSS[1:0] 位 (WDT 窗口开始位置选择位)

递减计数器的窗口开始位置可从计数期间 (开始计数时为 100%, 发生下溢时为 0%) 的 100%、75%、50% 和 25% 中选择。从窗口开始位置到窗口结束位置的期间为刷新允许期间, 其他期间为刷新禁止期间。

详细内容请参照“26. 看门狗定时器 (WDTA) ”。

WDTRSTIRQS 位 (WDT 复位中断请求选择位)

此位设定递减计数器发生下溢或者刷新错误时的运行。能选择看门狗定时器复位或者非屏蔽中断请求。

详细内容请参照“26. 看门狗定时器 (WDTA) ”。

7.2.2 选项功能选择寄存器 1 (OFS1)

地址 FFFF FF88h

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
复位后的值 用户的设定值 (注1)															
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	HOCO EN	—	—	—	—	—	LVDAS	VDSEL[1:0]	—
复位后的值 用户的设定值 (注1)															

注 1. 空白产品为“FFFF FFFFh”，在用户编程后为编程后的值。

位	符号	位名	功能	R/W
b1-b0	VDSEL[1:0]	电压检测 0 电平选择位 (注 1)	b1 b2 0 0: 选择 3.80V 0 1: 选择 2.85V 1 0: 选择 1.90V 1 1: 选择 1.72V	R
b2	LVDAS	电压检测 0 电路启动位	0: 复位后, 电压监视 0 复位有效。 1: 复位后, 电压监视 0 复位无效。	R
b7-b3	—	保留位	读取值为编程后的值。在编程时, 必须置“1”。	R
b8	HOCOEN	HOCO 振荡有效位	0: 复位后, HOCO 振荡有效。 1: 复位后, HOCO 振荡无效。	R
b31-b9	—	保留位	读取值为编程后的值。在编程时, 必须置“1”。	R

注 1. 将通过 VDSEL[1:0] 位选择的电压检测 0 电平设定到电压监视 0 复位功能和上电复位功能。

OFS1 寄存器在 ROM 内, 必须和程序一起写, 但是在写后不能对 OFS1 寄存器进行追加写。

如果擦除包括 OFS1 寄存器的块, OFS1 寄存器就变为“FFFF FFFFh”。

在用户引导模式中, OFS1 寄存器的设定无效, 为“FFFF FFFFh”。

VDSEL[1:0] 位 (电压检测 0 电平选择位)

此位选择电压检测 0 电路的电压检测电平。

LVDAS 位 (电压检测 0 电路启动位)

此位选择在复位后将电压监视 0 复位置为有效或者无效。

通过 VDSEL[1:0] 位选择在电压监视 0 电路监视的 Vdet0 电压。

HOCOEN 位 (HOCO 振荡有效位)

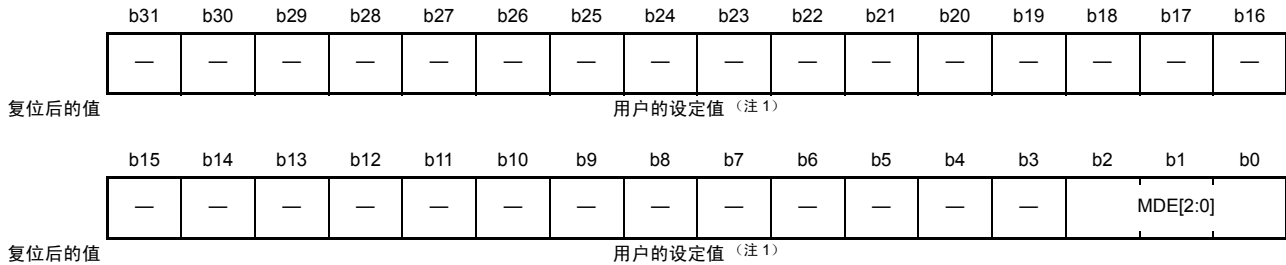
此位选择在复位后将 HOCO 振荡允许位置为有效或者无效。

能通过将 HOCOEN 位置“0”, 使 HOCO 在 CPU 运行前开始振荡并且减少振荡稳定的等待时间。

即使将 HOCOEN 位置“0”, 也不将系统时钟源转换为 HOCO。但是, 能通过 CPU 改写时钟源选择位 (SCKCR3.CKSEL[2:0]) 转换为 HOCO。

7.2.3 字节序选择寄存器 B (MDEB)、字节序选择寄存器 S (MDES)

地址 MDEB (用户引脚模式): FF7F FFF8h
MDES (单芯片模式): FFFF FF80h



注 1. 空白产品为“FFFF FFFFh”，在用户编程后为编程后的值。

位	符号	位名	功能	R/W
b2-b0	MDE[2:0]	字节序选择位	b2 b0 0 0 0: 大端法 1 1 1: 小端法 上述以外: 不能设定	R
b31-b3	—	保留位	读取值为编程后的值。在编程时, 必须置“1”。	R

MDE_n (n=B、S) 寄存器是选择 CPU 字节序的寄存器。在用户引导模式中, 通过地址 FF7F FFF8h 的字节序选择寄存器 B (MDEB) 选择字节序; 在单芯片模式中, 通过地址 FFFF FF80h 的字节序选择寄存器 S (MDES) 选择字节序。

MDE_n 寄存器在 ROM 内, 必须和程序一起写, 但是在写后不能对 MDE_n 寄存器进行追加写。

如果擦除包括 MDE_n 寄存器的块, MDE_n 寄存器就变为“FFFF FFFFh”。

MDE[2:0] 位 (字节序选择位)

这些位选择小端法或者大端法。

在用户引导模式中运行时, 用户引导区的 FF7F FFF8h 的值决定字节序; 在单芯片模式中运行时, 用户区的 FFFF FF80h 的值决定字节序。

7.3 UB 码

UB 码 A 和 UB 码 B 是用户使用用户引导模式时需要的代码。如果在以下的 4 个条件成立时解除复位，就在用户引导模式中启动 MCU。

- UB 码 A 是“5573 6572h、426F 6F74h”。
- UB 码 B 是“FFFF FF07h、0008 C04Ch”。
- 给 MD 引脚输入 Low 电平。
- 给 PC7 引脚输入 High 电平。

7.3.1 UB 码 A

UB 码 A 是 32 位（2 个字）数据。必须给 UB 码 A 设定“5573 6572h、426F 6F74h”，而不能设定其他值。UB 码 A 的结构如图 7.2 所示，必须以 32 位为单位设定 UB 码 A。

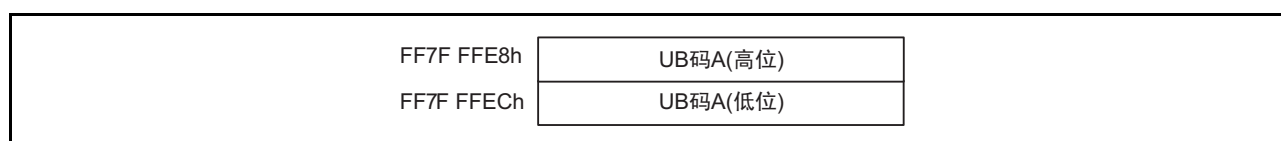


图 7.2 UB 码 A 的结构

7.3.2 UB 码 B

UB 码 B 是 32 位（2 个字）数据。必须给 UB 码 B 设定“FFFF FF07h、0008 C04Ch”，而不能设定其他值。UB 码 B 的结构如图 7.3 所示，必须以 32 位为单位设定 UB 码 B。

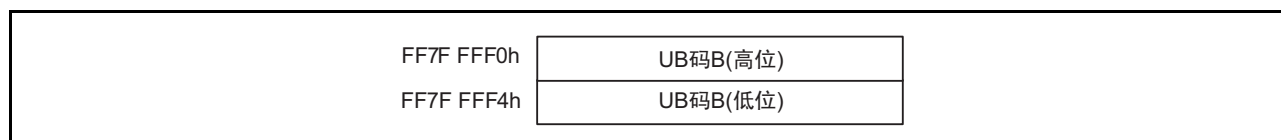


图 7.3 UB 码 B 的结构

7.4 使用时的注意事项

7.4.1 选项设定存储器的设定例子

因为选项设定存储器在 ROM 内，所以不能通过执行指令进行改写，必须在编写程序时写适当的值。设定例子如下所示：

- 给 OFS0 寄存器设定“ffff fff8h”的情况


```
.org    0fff ff8ch
.lword  0fff fff8h
```

注：程序的格式因编译器而不同，必须通过编译器的使用手册进行确认。

8. 电压检测电路 (LVDAa)

电压检测电路是能通过程序监视 VCC 引脚的输入电压的电路。

8.1 概要

电压检测 0 能通过选项功能选择寄存器 1 (OFS1) 从 4 个电平中选择检测电压。

电压检测 1 和电压检测 2 能通过电压检测电平选择寄存器 (LVDLVLR) 从 16 个电平中选择检测电压。

电压检测 2 能进行 VCC 和 CMPA2 引脚输入电压的转换

能使用电压监视 0 复位、电压监视 1 复位 / 中断和电压监视 2 复位 / 中断。

电压监视 1 和电压监视 2 与比较器 A1 和比较器 A2 共用电压检测电路。能选择电压监视 1、电压监视 2 使用电压检测电路或者比较器 A1、比较器 A2 使用电压检测电路。

电压检测电路的规格如表 8.1 所示，电压检测电路的框图、电压监视 1 中断 / 复位发生电路的框图和电压监视 2 中断 / 复位发生电路的框图分别如图 8.1、图 8.2 和图 8.3 所示。

表 8.1 电压检测电路的规格

项目		电压监视 0	电压监视 1	电压监视 2
VCC 监视	监视电压	Vdet0	Vdet1	Vdet2
	检测对象	在下降过程中是否经过 Vdet0。	在上升或者下降过程中是否经过 Vdet1。	在上升或者下降过程中是否经过 Vdet2。 能通过 LVCMPCR.EXVCCINP2 位进行 VCC 和 CMPA2 引脚输入电压的转换。
	检测电压	能通过 OFS1 寄存器从 4 个电平中选择。	能通过 LVDLVLR.LVD1LVL[3:0] 位从 16 个电平中选择。	根据选择 VCC 或者 CMPA2 引脚输入电压而不同。 能通过 LVDLVLR.LVD2LVL[3:0] 位从 16 个电平中选择。
	监视标志	无	LVD1SR.LVD1MON 标志： 监视是高于还是低于 Vdet1。 LVD1SR.LVD1DET 标志： 检测是否经过 Vdet1。	LVD2SR.LVD2MON 标志： 监视是高于还是低于 Vdet2。 LVD2SR.LVD2DET 标志： 检测是否经过 Vdet2。
电压检测时的处理	复位	电压监视 0 复位 当 Vdet0 > VCC 时，复位： 从 VCC > Vdet0 开始经过一定的时间后 CPU 重新开始运行。	电压监视 1 复位 当 Vdet1 > VCC 时，复位： 能选择从 VCC > Vdet1 开始经过一定的时间后 CPU 重新开始运行或者从 Vdet1 > VCC 开始经过一定的时间后 CPU 重新开始运行。	电压监视 2 复位 当 Vdet2 > VCC 时，复位： 能选择从 VCC > Vdet2 开始经过一定的时间后 CPU 重新开始运行或者从 Vdet2 > VCC 开始经过一定的时间后 CPU 重新开始运行。
	中断	无	电压监视 1 中断 能选择非屏蔽中断或者可屏蔽中断 在 Vdet1 > VCC 和 VCC > Vdet1 时都产生中断请求、在 Vdet1 > VCC 时产生中断请求、或者在 VCC > Vdet1 时产生中断请求。	电压监视 2 中断 能选择非屏蔽中断或者可屏蔽中断 在 Vdet2 > VCC 和 VCC > Vdet2 时都产生中断请求、在 Vdet2 > VCC 时产生中断请求、或者在 VCC > Vdet2 时产生中断请求。
数字滤波器	有效 / 无效的转换	无数字滤波器功能	有	有
	采样时间	—	LOCO 的 n 分频 × 2 (n: 1、2、4、8)	LOCO 的 n 分频 × 2 (n: 1、2、4、8)
事件链接功能		无	有 检测到经过 Vdet1 事件的输出	有 检测到经过 Vdet2 事件的输出

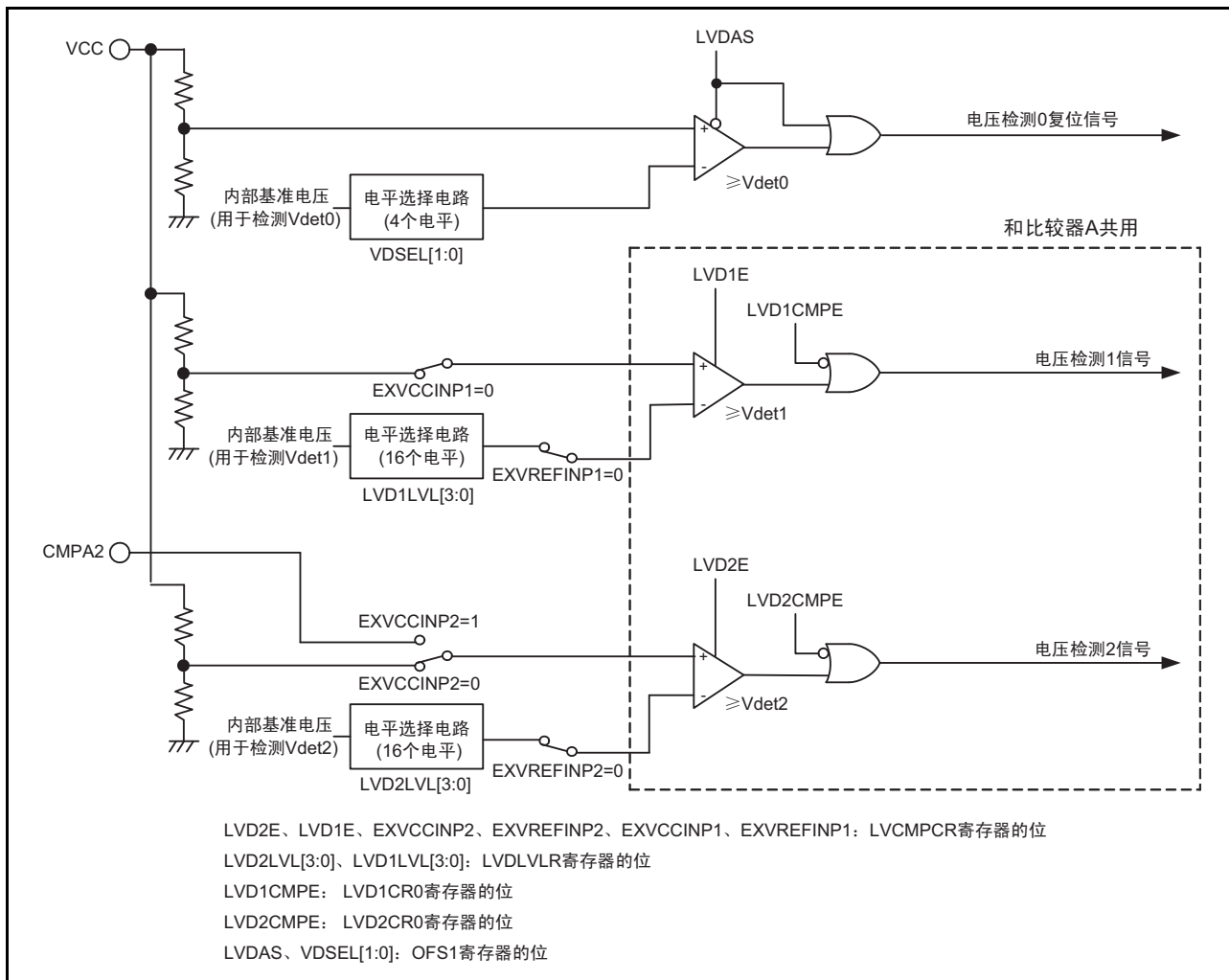


图 8.1 电压检测电路的框图

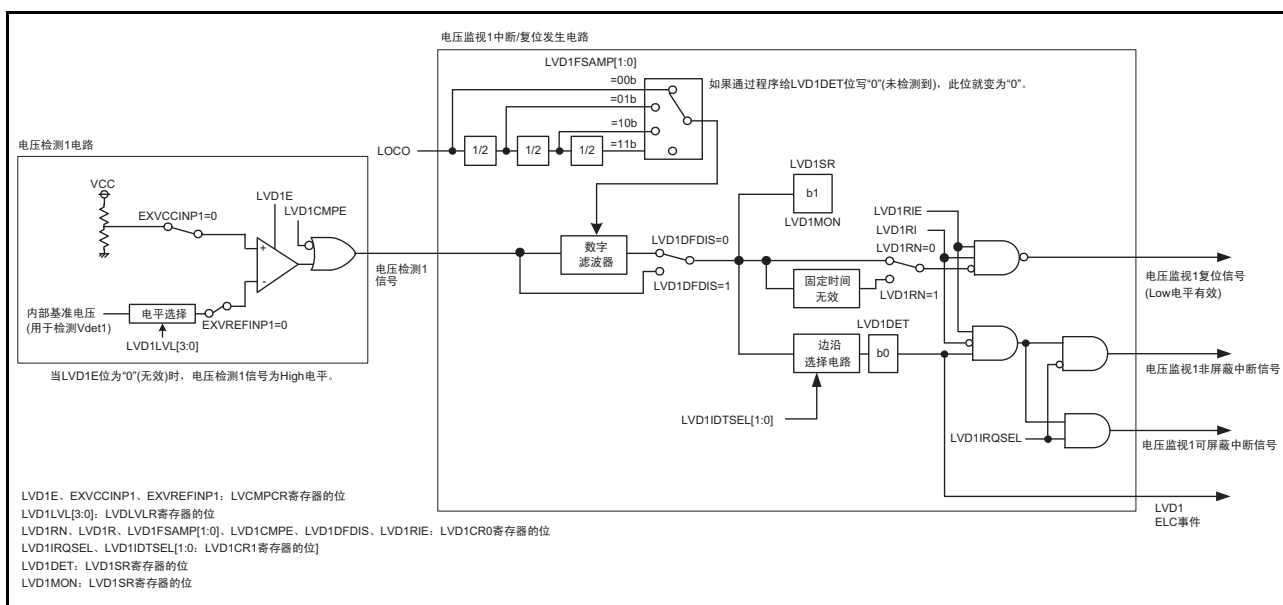


图 8.2 电压监视 1 中断 / 复位发生电路的框图

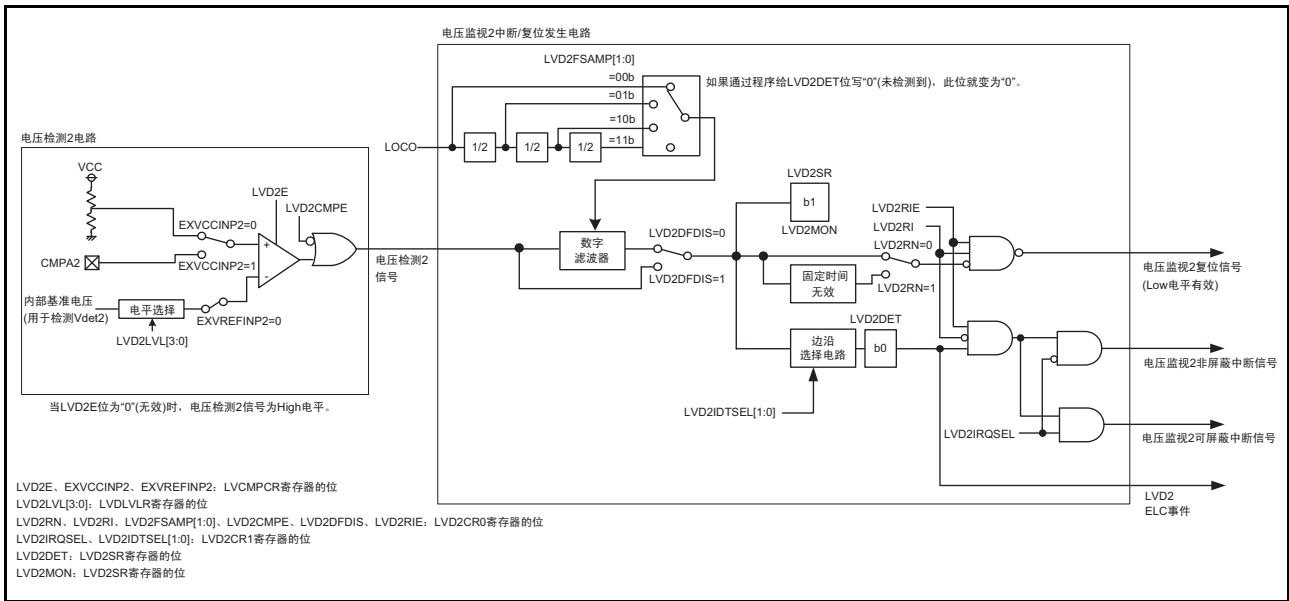


图 8.3 电压监视 2 中断 / 复位发生电路的框图

电压检测电路使用的输入 / 输出引脚如表 8.2 所示。

表 8.2 电压检测电路的输入 / 输出引脚

引脚名	输入 / 输出	功能
CMPA2	输入	用于电压检测 2 的检测对象电压引脚

8.2 寄存器说明

8.2.1 电压监视 1 电路 / 比较器 A1 的控制寄存器 1 (LVD1CR1)

地址 0008 00E0h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	LVD1IRQSEL	LVD1IDTSEL	[1:0]
0	0	0	0	0	0	0	1

复位后的值

位	符号	位名	功能	R/W
b1-b0	LVD1IDTSEL[1:0]	电压监视 1/ 比较器 A1 的中断 ELC 事件发生条件选择位	b1 b0 0 0: 当检测到 $VCC \geq V_{det1}$ (上升) 时 0 1: 当检测到 $VCC < V_{det1}$ (下降) 时 1 0: 当检测到下降和上升时 1 1: 不能设定	R/W
b2	LVD1IRQSEL	电压监视 1/ 比较器 A1 的中断种类选择位	0: 非屏蔽中断 1: 可屏蔽中断	R/W
b7-b3	—	保留位	读写值都为“0”。	R/W

8.2.2 电压监视 1 电路 / 比较器 A1 的状态寄存器 (LVD1SR)

地址 0008 00E1h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	LVD1MON	LVD1DET
0	0	0	0	0	0	1	0

复位后的值

位	符号	位名	功能	R/W
b0	LVD1DET	电压监视 1/ 比较器 A1 的电压变化检测标志	0: 未检测到 1: 检测到经过 V_{det1}	R/(W) (注 1)
b1	LVD1MON	电压监视 1/ 比较器 A1 的信号监视标志	0: $VCC < V_{det1}$ 1: $VCC \geq V_{det1}$ 或者 LVD1MON 无效	R
b7-b2	—	保留位	读写值都为“0”。	R/W

注 1. 只能写“0”。从写“0”后到被反映为 LVD1DET 位的读取值为止，需要 2 个系统时钟周期。

LVD1DET 标志 (电压监视 1/ 比较器 A1 的电压变化检测标志)

当 LVCMPCR.LVD1E 位为“1” (电压检测 1 电路有效) 并且 LVD1CR0.LVD1CMPE 位为“1” (允许输出电压监视 1 电路的比较结果) 时，LVD1DET 标志有效。

要将 LVD1DET 标志置“0”时，必须先将 LVD1CR0.LVD1RIE 位置“0” (禁止)。如果要再次将 LVD1CR0.LVD1RIE 位置“1” (允许)，就必须至少经过 2 个 PCLKB 周期后进行设定。

能通过读由 PCLKB 定义存取周期数的 I/O 寄存器，至少确保 2 个 PCLKB 周期的等待时间。

LVD1MON 标志 (电压监视 1/ 比较器 A1 的信号监视标志)

当 LVCMPCR.LVD1E 位为“1” (电压检测 1 电路有效) 并且 LVD1CR0.LVD1CMPE 位为“1” (允许输出电压监视 1 电路的比较结果) 时，LVD1MON 标志有效。

8.2.3 电压监视 2 电路 / 比较器 A2 的控制寄存器 1 (LVD2CR1)

地址 0008 00E2h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	LVD2IRQSEL	LVD2IDTSEL [1:0]	
0	0	0	0	0	0	0	1

复位后的值

位	符号	位名	功能	R/W
b1-b0	LVD2IDTSEL[1:0]	电压监视 2/ 比较器 A2 的中断 ELC 发生条件选择位	b1 b0 0 0: 当检测到 $VCC \geq V_{det2}$ (上升) 时 0 1: 当检测到 $VCC < V_{det2}$ (下降) 时 1 0: 当检测到下降和上升时 1 1: 不能设定	R/W
b2	LVD2IRQSEL	电压监视 2/ 比较器 A2 的中断种类选择位	0: 非屏蔽中断 1: 可屏蔽中断	R/W
b7-b3	—	保留位	读写值都为“0”。	R/W

8.2.4 电压监视 2 电路 / 比较器 A2 的状态寄存器 (LVD2SR)

地址 0008 00E3h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	LVD2MON	LVD2DET
0	0	0	0	0	0	1	0

复位后的值

位	符号	位名	功能	R/W
b0	LVD2DET	电压监视 2/ 比较器 A2 的电压变化检测标志	0: 未检测到 1: 检测到经过 V_{det2}	R(W) (注 1)
b1	LVD2MON	电压监视 2/ 比较器 A2 的信号监视标志	0: $VCC < V_{det2}$ 1: $VCC \geq V_{det2}$ 或者 LVD2MON 无效	R
b7-b2	—	保留位	读写值都为“0”。	R/W

注 1. 只能写“0”。从写“0”后到被反映为 LVD2DET 位的读取值为止, 需要 2 个系统时钟周期。

LVD2DET 标志 (电压监视 2/ 比较器 A2 的电压变化检测标志)

当 LVCMPCR.LVD2E 位为“1” (电压检测 2 电路有效) 并且 LVD2CR0.LVD2CMPE 位为“1” (允许输出电压监视 2 电路的比较结果) 时, LVD2DET 标志有效。

要将 LVD2DET 标志置“0”时, 必须先将 LVD2CR0.LVD2RIE 位置“0” (禁止)。如果要再次将 LVD2CR0.LVD2RIE 位置“1” (允许), 就必须至少经过 2 个 PCLKB 周期后进行设定。

能通过读由 PCLKB 定义存取周期数的 I/O 寄存器, 至少确保 2 个 PCLKB 周期的等待时间。

LVD2MON 标志 (电压监视 2/ 比较器 A2 的信号监视标志)

当 LVCMPCR.LVD2E 位为“1” (电压检测 2 电路有效) 并且 LVD2CR0.LVD2CMPE 位为“1” (允许输出电压监视 2 电路的比较结果) 时, LVD2MON 标志有效。

8.2.5 电压监视电路 / 比较器 A 的控制寄存器 (LVCMPCR)

地址 0008 C297h

b7	b6	b5	b4	b3	b2	b1	b0
—	LVD2E	LVD1E	—	EXVCCI NP2	EXVREFI NP2	EXVCCI NP1	EXVREFI NP1
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	EXVREFINP1	比较器 A1 的基准电压外部输入选择位	0: 内部基准电压 在 LVD 下使用时, 必须置“0”。	R/W
b1	EXVCCINP1	比较器 A1 的比较电压外部输入选择位	0: 电压电压 (VCC) 在 LVD 下使用时, 必须置“0”。	R/W
b2	EXVREFINP2	比较器 A2 的基准电压外部输入选择位	0: 内部基准电压 在 LVD 下使用时, 必须置“0”。	R/W
b3	EXVCCINP2	比较器 A2 的比较电压外部输入选择位	0: 电压电压 (VCC) 1: CMPA2 引脚输入电压 (注 1)	R/W
b4	—	保留位	读写值都为“0”。	R/W
b5	LVD1E	电压检测 1/ 比较器 A1 允许位	0: 电压检测 1/ 比较器 A1 的电路无效 1: 电压检测 1/ 比较器 A1 的电路有效	R/W
b6	LVD2E	电压检测 2/ 比较器 A2 允许位	0: 电压检测 2/ 比较器 A2 的电路无效 1: 电压检测 2/ 比较器 A2 的电路有效	R/W
b7	—	保留位	读写值都为“0”。	R/W

注 1. 要将 EXVCCINP2 位置“1” (CMPA2 引脚输入电压) 时, 必须先将 LVDLVLRLVD2LVL[3:0] 位设定为“0001b”。

LVD1E 位 (电压检测 1/ 比较器 A1 允许位)

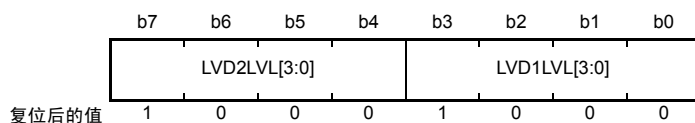
在使用电压检测 1/ 比较器 A1 的中断 / 复位或者 LVD1SR.LVD1MON 位时, 必须将 LVD1E 位置“1”。在将 LVD1E 位从“0”置为“1”并且经过 $t_d(E-A)$ 后, 电压检测 1/ 比较器 A1 电路运行。在深度软件待机模式中使用电压检测 1/ 比较器 A1 电路时, 不能将 DPSBYCR.DEEP CUT1 位置“1”。

LVD2E 位 (电压检测 2/ 比较器 A2 允许位)

在使用电压检测 2/ 比较器 A2 的中断 / 复位或者 LVD2SR.LVD2MON 位时, 必须将 LVD2E 位置“1”。在将 LVD2E 位从“0”置为“1”并且经过 $t_d(E-A)$ 后, 电压检测 2/ 比较器 A2 电路运行。在深度软件待机模式中使用电压检测 2/ 比较器 A2 电路时, 不能将 DPSBYCR.DEEP CUT1 位置“1”。

8.2.6 电压检测电平选择寄存器 (LVDLVLR)

地址 0008 C298h



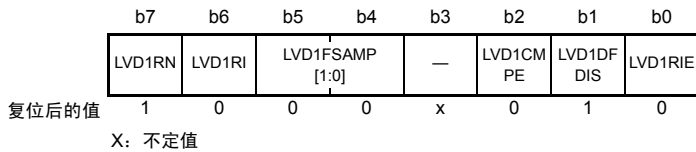
位	符号	位名	功能	R/W
b3-b0	LVD1LVL[3:0]	电压检测 1 电平选择位 (电压下降时的标准电压)	b3 b2 b1 b0 0 0 0 0: 4.15V 0 0 0 1: 4.00V 0 0 1 0: 3.85V 0 0 1 1: 3.70V 0 1 0 0: 3.55V 0 1 0 1: 3.40V 0 1 1 0: 3.25V 0 1 1 1: 3.10V 1 0 0 0: 2.95V 1 0 0 1: 2.80V 1 0 1 0: 2.65V 1 0 1 1: 2.50V 1 1 0 0: 2.35V 1 1 0 1: 2.20V 1 1 1 0: 2.05V 1 1 1 1: 1.90V	R/W
b7-b4	LVD2LVL[3:0]	电压检测 2 电平选择位 (电压下降时的标准电压)	(当 LVCMP2CR.EXVCCINP2=0 (选择 VCC 选择) 时) b7 b6 b5 b4 0 0 0 0: 4.15V 0 0 0 1: 4.00V 0 0 1 0: 3.85V 0 0 1 1: 3.70V 0 1 0 0: 3.55V 0 1 0 1: 3.40V 0 1 1 0: 3.25V 0 1 1 1: 3.10V 1 0 0 0: 2.95V 1 0 0 1: 2.80V 1 0 1 0: 2.65V 1 0 1 1: 2.50V 1 1 0 0: 2.35V 1 1 0 1: 2.20V 1 1 1 0: 2.05V 1 1 1 1: 1.90V (当 LVCMP2CR.EXVCCINP2=1 (选择 CMPA2 引脚) 时) b7 b6 b5 b4 0 0 0 1: 1.33V 上述以外: 不能设定	R/W

只能在 LVCMP2CR.LVD1E 位和 LVCMP2CR.LVD2E 位都为“0” (电压检测 n 电路无效) (n=1、2) 时更改 LVDLVLR 寄存器。

不能在设定相同检测电压电平的情况下使用电压检测 1 电路和电压检测 2 电路。在将电压检测 0 电路的检测电压电平设定为 1.90V 时, 不能将电压检测 1 电路和电压检测 2 电路的检测电压电平设定为 1.90V。

8.2.7 电压监视 1 电路 / 比较器 A1 的控制寄存器 0 (LVD1CR0)

地址 0008 C29Ah



位	符号	位名	功能	R/W
b0	LVD1RIE	电压监视 1/ 比较器 A1 的中断 / 复位允许位	0: 禁止 1: 允许	R/W
b1	LVD1DFDIS	电压监视 1/ 比较器 A1 的数字滤波器无效模式选择位	0: 数字滤波器有效 1: 数字滤波器无效	R/W
b2	LVD1CMPE	电压监视 1 电路 / 比较器 A1 的比较结果输出允许位	0: 禁止输出电压监视 1 电路的比较结果 1: 允许输出电压监视 1 电路的比较结果	R/W
b3	—	保留位	读取值为不定值, 只能写“0”。	R/W
b5-b4	LVD1FSAMP [1:0]	采样时钟选择位	b5 b4 0 0: LOCO 的 1 分频 0 1: LOCO 的 2 分频 1 0: LOCO 的 4 分频 1 1: LOCO 的 8 分频	R/W
b6	LVD1RI	电压监视 1 电路 / 比较器 A1 的模式选择位	0: 在经过 Vdet1 时发生电压监视 1 中断 1: 在下降过程中经过 Vdet1 时发生电压监视 1 复位	R/W
b7	LVD1RN	电压监视 1/ 比较器 A1 的复位无效选择位	0: 从检测到 VCC > Vdet1 开始经过一定的时间 (tLVD1) 后无效 1: 从 LVD1 复位有效开始经过一定的时间 (tLVD1) 后无效	R/W

LVD1RIE 位 (电压监视 1/ 比较器 A1 的中断 / 复位允许位)

不能在闪存的编程 / 擦除过程中发生 LVD1 复位和 LVD1 非屏蔽中断。

LVD1DFDIS 位 (电压监视 1/ 比较器 A1 的数字滤波器无效模式选择位)

要将 LVD1DFDIS 位置“0” (数字滤波器电路有效) 时, 必须将 LOCOCR.LCSTP 位置“0” (LOCO 运行)。

在软件待机模式或者深度软件待机模式中使用电压监视 1 电路时, 必须将 LVD1DFDIS 位置“1” (数字滤波器电路无效)。

LVD1FSAMP[1:0] 位 (采样时钟选择位)

只能在 LVD1DFDIS 位为“1” (数字滤波器电路无效) 时改写 LVD1FSAMP[1:0] 位。当 LVD1DFDIS 位为“0” (数字滤波器电路有效) 时, 不能改写 LVD1FSAMP[1:0] 位。

LVD1RI 位 (电压监视 1 电路 / 比较器 A1 的模式选择位)

当 LVD1RI 位为“1” (选择电压监视 1 复位) 或者 LVD2CR0.LVD2RI 位为“1” (选择电压监视 2 复位) 时, 不能转移到深度软件待机模式, 而转移到软件待机模式。要转移到深度软件待机模式时, 必须将 LVD1RI 位置“0” (选择电压监视 1 中断) 并且将 LVD2CR0.LVD2RI 位置“0” (选择电压监视 2 中断)。

LVD1RN 位 (电压监视 1 / 比较器 A1 的复位无效选择位)

要将 LVD1RN 位置“1” (从 LVD1 复位有效开始经过一定的时间后无效) 时, 必须将 LOCOCR.LCSTP 位置“0” (LOCO 运行)。要转移到软件待机模式或者深度软件待机模式时, 只能将 LVD1RN 位置“0” (从检测到 $VCC > V_{det1}$ 开始经过一定的时间后无效), 而不能将 LVD1RN 位置“1” (从 LVD1 复位有效开始经过一定的时间后无效)。

8.2.8 电压监视 2 电路 / 比较器 A2 的控制寄存器 0 (LVD2CR0)

地址 0008 C29Bh

	b7	b6	b5	b4	b3	b2	b1	b0
	LVD2RN	VD2RI	LVD2FSAMP [1:0]		—	LVD2CMPE	LVD2DFDIS	LVD2RIE
复位后的值	1	0	0	0	x	0	1	0

X: 不定值

位	符号	位名	功能	R/W
b0	LVD2RIE	电压监视 2/ 比较器 A2 的中断 / 复位允许位	0: 禁止 1: 允许	R/W
b1	LVD2DFDIS	电压监视 2/ 比较器 A2 的数字滤波器无效模式选择位	0: 数字滤波器有效 1: 数字滤波器无效	R/W
b2	LVD2CMPE	电压监视 2 电路 / 比较器 A2 的比较结果输出允许位	0: 禁止输出电压监视 2 电路的比较结果 1: 允许输出电压监视 2 电路的比较结果	R/W
b3	—	保留位	读取值为不定值, 只能写“0”。	R/W
b5-b4	LVD2FSAMP [1:0]	采样时钟选择位	b5 b4 0 0: LOCO 的 1 分频 0 1: LOCO 的 2 分频 1 0: LOCO 的 4 分频 1 1: LOCO 的 8 分频	R/W
b6	LVD2RI	电压监视 2 电路 / 比较器 A2 的模式选择位	0: 在经过 Vdet2 时发生电压监视 2 中断 1: 在下降过程中经过 Vdet2 时发生电压监视 2 复位	R/W
b7	LVD2RN	电压监视 2/ 比较器 A2 的复位无效选择位	0: 从检测到 VCC > Vdet2 开始经过一定的时间 (tLVD2) 后无效 1: 从 LVD2 复位有效开始经过一定的时间 (tLVD2) 后无效	R/W

LVD2RIE 位 (电压监视 2/ 比较器 A2 的中断 / 复位允许位)

不能在闪存的编程 / 擦除过程中发生 LVD2 复位和 LVD2 非屏蔽中断。

LVD2DFDIS 位 (电压监视 2/ 比较器 A2 的数字滤波器无效模式选择位)

要将 LVD2DFDIS 位置“0” (数字滤波器电路有效) 时, 必须将 LOCOCR.LCSTP 位置“0” (LOCO 运行)。

在软件待机模式或者深度软件待机模式中使用电压监视 2 电路时, 必须将 LVD2DFDIS 位置“1” (数字滤波器电路无效)。

LVD2FSAMP[1:0] 位 (采样时钟选择位)

只能在 LVD2DFDIS 位为“1” (数字滤波器电路无效) 时改写 LVD2FSAMP[1:0] 位。当 LVD2DFDIS 位为“0” (数字滤波器电路有效) 时, 不能改写 LVD2FSAMP[1:0] 位。

LVD2RI 位 (电压监视 2 电路 / 比较器 A2 的模式选择位)

当 LVD2RI 位为“1” (选择电压监视 2 复位) 或者 LVD1CR0.LVD1RI 位为“1” (选择电压监视 1 复位) 时, 不能转移到深度软件待机模式, 而转移到软件待机模式。要转移到深度软件待机模式时, 必须将 LVD2RI 位置“0” (选择电压监视 2 中断) 并且将 LVD1CR0.LVD1RI 位置“0” (选择电压监视 1 中断)。

LVD2RN 位 (电压监视 2 / 比较器 A2 的复位无效选择位)

要将 LVD2RN 位置“1” (从 LVD2 复位有效开始经过一定的时间后无效) 时, 必须将 LOCOCR.LCSTP 位置“0” (LOCO 运行)。要转移到软件待机模式或者深度软件待机模式时, 只能将 LVD2RN 位置“0” (从检测到 $VCC > V_{det2}$ 开始经过一定的时间后无效), 而不能将 LVD2RN 位置“1” (从 LVD2 复位有效开始经过一定的时间后无效)。

8.3 VCC 输入电压的监视

8.3.1 Vdet0 的监视

不能监视 Vdet0。

8.3.2 Vdet1 的监视

在进行以下设定后，能通过 LVD1SR.LVD1MON 标志来监视电压监视 1 的比较结果。

1. 设定 LVDLVLR.LVD1LVL[3:0] 位（电压检测 1 的检测电压）。
2. 将 LVCMPCR.EXVREFINP1 位置“0”（内部基准电压），将 LVCMPCR.EXVCCINP1 位置“0”（VCC 电压）。
3. 通过 LVD1CR0.LVD1FSAMP[1:0] 位选择数字滤波器的采样时钟。
4. 将 LVCMPCR.LVD1E 位置“1”（电压检测 1 电路有效）。
5. 在等待 $t_d(E-A)$ 后，将 LVD1CR0.LVD1CMPE 位置“1”（允许输出电压监视 1 电路的比较结果）。
6. 至少等待 1 个 LOCO 周期。
7. 将 LVD1CR0.LVD1DFDIS 位置“0”（数字滤波器有效）。
8. 至少等待 $2n+3$ 个 LOCO 周期。
($n=1、2、4、8$: 数字滤波器的采样时钟=LOCO 的 n 分频)

注. 如果在数字滤波器无效时使用，就不需要步骤 3、步骤 6 ~ 8。

8.3.3 Vdet2 的监视

在进行以下设定后，能通过 LVD2SR.LVD2MON 标志来监视电压监视 2 的比较结果。

1. 设定 LVDLVLR.LVD2LVL[3:0] 位（电压检测 2 的检测电压）。
2. 将 LVCMPCR.EXVREFINP2 位置“0”（内部基准电压），将 LVCMPCR.EXVCCINP2 位置“0”（VCC 电压）或者置“1”（CMPA2 引脚输入电压）。
3. 通过 LVD2CR0.LVD2FSAMP[1:0] 位选择数字滤波器的采样时钟。
4. 将 LVCMPCR.LVD2E 位置“1”（电压检测 2 电路有效）。
5. 在等待 $t_d(E-A)$ 后，将 LVD2CR0.LVD2CMPE 位置“1”（允许输出电压监视 2 的电路比较结果）。
6. 至少等待 1 个 LOCO 周期。
7. 将 LVD2CR0.LVD2DFDIS 位置“0”（数字滤波器有效）。
8. 至少等待 $2n+3$ 个 LOCO 周期。
($n=1、2、4、8$: 数字滤波器的采样时钟=LOCO 的 n 分频)

注. 如果在数字滤波器无效时使用，就不需要步骤 3、步骤 6 ~ 8。

8.4 电压监视 0 复位

在使用电压监视 0 复位时，必须将电压检测 0 电路的启动位 (OFS1.LVDAS) 置“0” (复位后，电压监视 0 复位有效)。

电压监视 0 复位的运行例子如图 8.4 所示。

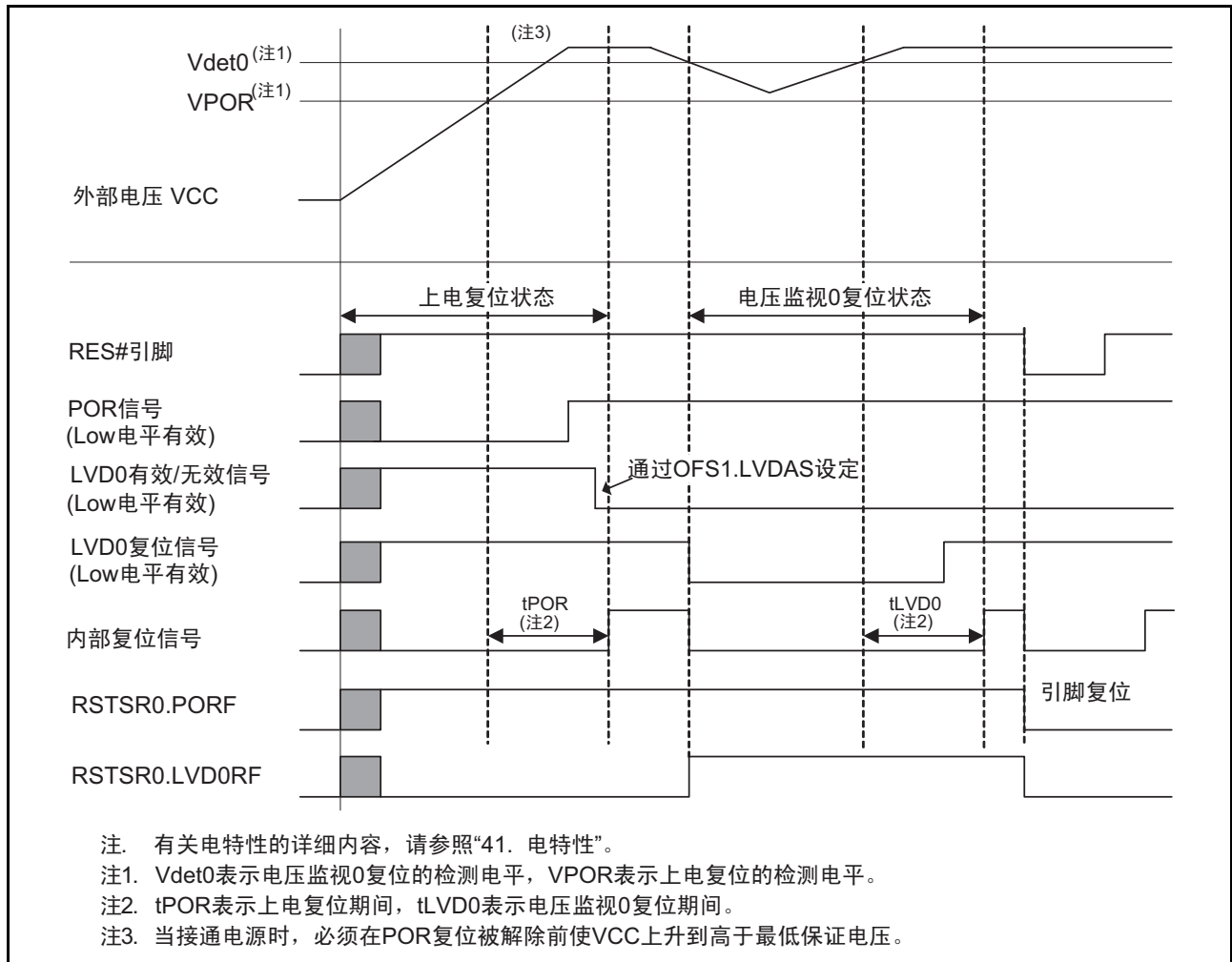


图 8.4 电压监视 0 复位的运行例子

8.5 电压监视 1 中断和电压监视 1 复位

电压监视 1 中断和电压监视 1 复位的相关位的运行设定步骤如表 8.3 所示，电压监视 1 中断和电压监视 1 复位的相关位的停止设定步骤如表 8.4 所示，电压监视 1 中断的运行例子如图 8.5 所示。有关电压监视 1 复位的运行例子，请参照“6. 复位”的图 6.2。

在软件待机模式或者深度软件待机模式中使用电压监视 1 电路时，必须将 LVD1CR0.LVD1DFDIS 位置“1”（数字滤波器无效）。

表 8.3 电压监视 1 中断和电压监视 1 复位的相关位的运行设定步骤

步骤	使用数字滤波器的情况		不使用数字滤波器的情况	
	电压监视 1 中断、 电压监视 1 的 ELC 事件输出	电压监视 1 复位	电压监视 1 中断 电压监视 1 的 ELC 事件输出	电压监视 1 复位
1 (注2)	通过 LVD1LVL.R.LVD1LVL[3:0] 位选择检测电压。			
2 (注2)	将 LVCMPPCR.EXVREFINP1 位置“0”（内部基准电压）， 将 LVCMPPCR.EXVCCINP1 位置“0”（VCC 电压）。			
3 (注1)	通过 LVD1CR0.LVD1FSAMP[1:0]位选择数字滤波器的采样时钟。		将 LVD1CR0.LVD1DFDIS 位置“1”（数字滤波器无效）。	
4 (注1、注2)	将 LVD1CR0.LVD1RI 位置“0” （电压监视 1 中断）。	将 LVD1CR0.LVD1RI 位置 “1”（电压监视 1 复位）。 通过 LVD1CR0.LVD1RN 位 选择复位无效的种类。	将 LVD1CR0.LVD1RI 位置“0” （电压监视 1 中断）。	将 LVD1CR0.LVD1RI 位置 “1”（电压监视 1 复位）。 通过 LVD1CR0.LVD1RN 位选 择复位无效的种类。
5	通过 LVD1CR1.LVD1IDTSEL [1:0] 位选择中断请求时序。 通过 LVD1CR1.LVD1IRQSEL 位 选择中断的种类。	—	通过 LVD1CR1.LVD1IDTSEL [1:0] 位选择中断请求时序。 通过 LVD1CR1.LVD1IRQSEL 位 选择中断的种类。	—
6	—	将 LVD1CR0.LVD1RIE 位置 “1”（允许电压监视 1 中断/ 复位）。	—	将 LVD1CR0.LVD1RIE 位置 “1”（允许电压监视 1 中断/ 复位）。
7 (注2)	将 LVCMPPCR.LVD1E 位置“1”（电压检测 1 电路有效）。			
8 (注2)	至少等待 td(E-A)。			
9	将 LVD1CR0.LVD1CMPE 位置“1”（允许输出电压监视 1 电路的比较结果）。			
10	至少等待 1 个 LOCO 周期。		—	
11	将 LVD1CR0.LVD1DFDIS 位置“0”（数字滤波器有效）。		—	
12	至少等待 $2n+3$ 个 LOCO 周期（ $n=1、2、4、8$ ：数字滤波器的采样时钟 = LOCO 的 n 分频）		—（无等待时间）	
13	将 LVD1SR.LVD1DET 位置 “0”。	—	将 LVD1SR.LVD1DET 位置 “0”。	—
14	将 LVD1CR0.LVD1RIE 位置 “1”（允许电压监视 1 中断/ 复位）。	—	将 LVD1CR0.LVD1RIE 位置 “1”（允许电压监视 1 中断/ 复位）。	—

注 1. 可以用 1 条指令同时执行步骤 3 和步骤 4。

注 2. 在以下的任何一种情况下，不需要步骤 1、2、4、7、8。

在通过设定为电压监视 1 中断（LVD1CR0.LVD1RI=0）而运行时，如果

- 在停止后只更改 LVD1CR0.LVD1FSAMP 位、LVD1CR0.LVD1DFDIS 位或者 LVD1CR1.LVD1IRQSEL 位、LVD1CR1.LVD1IDTSEL 位的设定而重新运行时

- 在停止后不更改电压检测电路的相关设定而重新运行时

对于通过设定为电压监视 1 复位（LVD1CR0.LVD1RI=1）而运行时的更改，必须通过步骤 1~14 进行设定。

表 8.4 电压监视 1 中断和电压监视 1 复位的相关位的停止设定步骤

步骤	电压监视 1 中断、电压监视 1 的 ELC 事件输出	电压监视 1 复位
1	将 LVD1CR0.LVD1RIE 位置“0”（禁止电压监视 1 中断 / 复位）。	—
2	将 LVD1CR0.LVD1CMPE 位置“0”（禁止输出电压监视 1 电路的比较结果）。	
3 (注 1)	将 LVCMPCR.LVD1E 位置“0”（电压检测 1 电路无效）。	
4	—	将 LVD1CR0.LVD1RIE 位置“0”（禁止电压监视 1 中断 / 复位）。
5	更改除 LVCMPCR.LVD1E 位、LVD1CR0.LVD1RIE 位和 LVD1CR0.LVD1CMPE 位以外的电压检测电路相关寄存器的设定。	

注 1. 在以下的任何一种情况下，不需要步骤 3。

在通过设定为电压监视 1 中断 (LVD1CR0.LVD1RI=0) 而运行时，如果

- 在停止后只更改 LVD1CR0.LVD1FSAMP 位、LVD1CR0.LVD1DFDIS 位或者 LVD1CR1.LVD1IRQSEL、LVD1CR1.LVD1IDTSEL 位的设定而重新运行时
- 在停止后不更改电压检测电路的相关设定而重新运行时

对于通过设定为电压监视 1 复位 (LVD1CR0.LVD1RI=1) 而运行时的更改，必须通过步骤 1~5 进行设定。

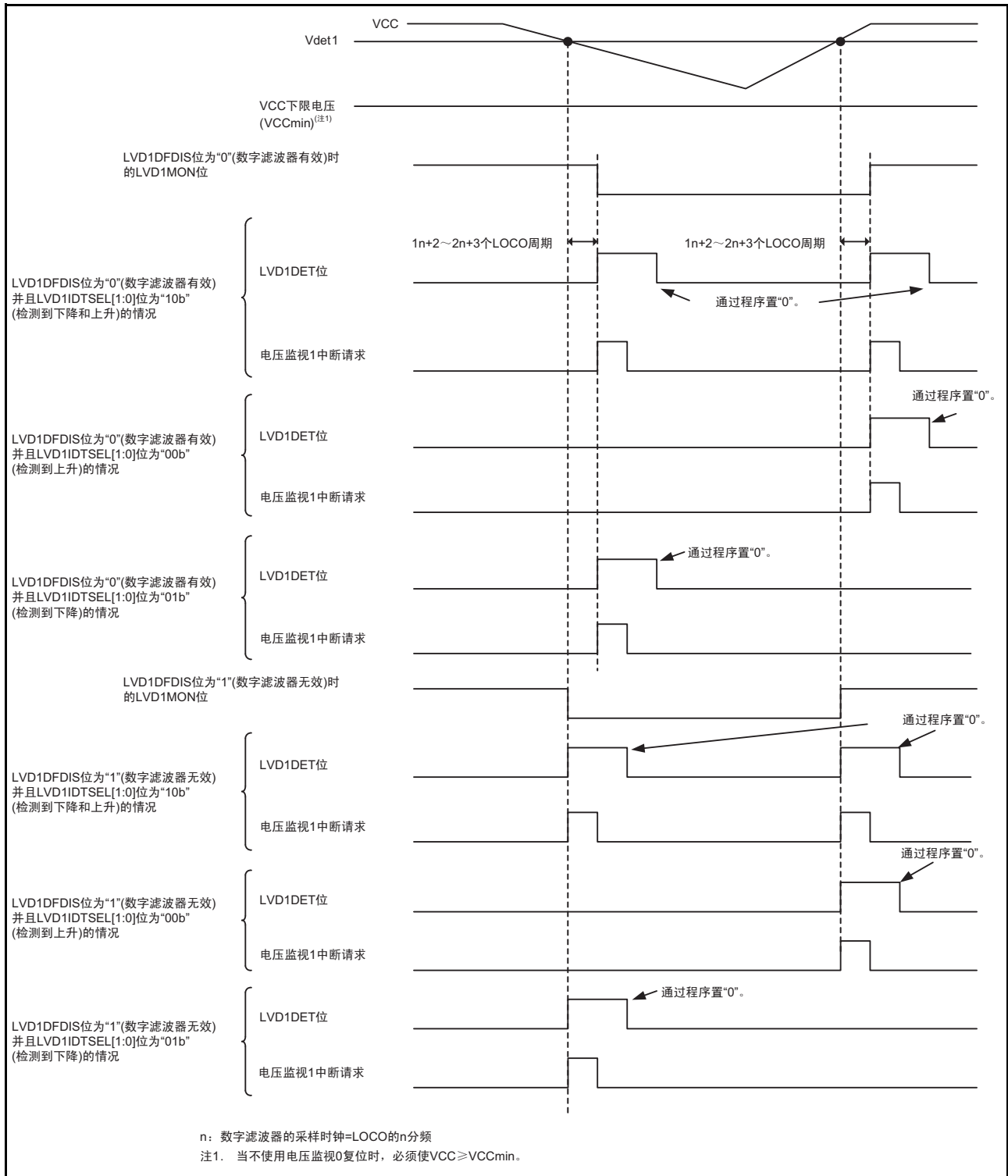


图 8.5 电压监视 1 中断的运行例子

8.6 电压监视 2 中断和电压监视 2 复位

电压监视 2 中断和电压监视 2 复位的相关位的运行设定步骤如表 8.5 所示，电压监视 2 中断和电压监视 2 复位的相关位的停止设定步骤如表 8.6 所示，电压监视 2 中断的运行例子如图 8.6 所示。有关电压监视 2 复位的运行例子，请参照“6. 复位”的图 6.2。

在软件待机模式或者深度软件待机模式中使用电压监视 2 电路时，必须将 LVD2CR0.LVD2DFDIS 位置“1”（数字滤波器无效）。

表 8.5 电压监视 2 中断和电压监视 2 复位的相关位的运行设定步骤

步骤	使用数字滤波器的情况		不使用数字滤波器的情况	
	电压监视 2 中断 电压监视 2 的 ELC 事件输出	电压监视 2 复位	电压监视 2 中断 电压监视 2 的 ELC 事件输出	电压监视 2 复位
1 (注2)	通过 LVLDLVL.R.LVD2LVL[3:0] 位选择检测电压。			
2 (注2)	将 LVCMPCR.EXVREFINP2 位置“0”（内部基准电压）， 将 LVCMPCR.EXVCCINP2 位置“0”（VCC 电压）或者置“1”（CMPA2 引脚输入电压）。			
3 (注1)	通过 LVD2CR0.LVD2FSAMP[1:0]位选择数字滤波器的采样时钟。		将 LVD2CR0.LVD2DFDIS 位置“1”（数字滤波器无效）。	
4 (注1、注2)	将 LVD2CR0.LVD2RI 位置“0” （电压监视 2 中断）。	将 LVD2CR0.LVD2RI 位置 “1”（电压监视 2 复位）。 通过 LVD2CR0.LVD2RN 位 选择复位无效的种类。	将 LVD2CR0.LVD2RI 位置“0” （电压监视 2 中断）。	将 LVD2CR0.LVD2RI 位置 “1”（电压监视 2 复位）。 通过 LVD2CR0.LVD2RN 位 选择复位无效的种类。
5	通过 LVD2CR1.LVD2IDTSEL [1:0] 位选择中断请求时序。 通过 LVD2CR1.LVD2IRQSEL 位选择中断的种类。	—	通过 LVD2CR1.LVD2IDTSEL [1:0] 位选择中断请求时序。 通过 LVD2CR1.LVD2IRQSEL 位选择中断的种类。	—
6	—	将 LVD2CR0.LVD2RIE 位置 “1”（允许电压监视 2 中断/ 复位）。	—	将 LVD2CR0.LVD2RIE 位置 “1”（允许电压监视 2 中断/ 复位）。
7 (注2)	将 LVCMPCR.LVD2E 位置“1”（电压检测 2 电路有效）。			
8 (注2)	至少等待 td(E-A)。			
9	将 LVD2CR0.LVD2CMPE 位置“1”（允许输出电压监视 2 电路的比较结果）。			
10	至少等待 1 个 LOCO 周期。		—	
11	将 LVD2CR0.LVD2DFDIS 位置“0”（数字滤波器有效）。		—	
12	至少等待 2n+3 个 LOCO 周期（n=1、2、4、8：数字滤波器的采样时钟 = LOCO 的 n 分频）		—（无等待时间）	
13	将 LVD2SR.LVD2DET 位置 “0”。	—	将 LVD2SR.LVD2DET 位置 “0”。	—
14	将 LVD2CR0.LVD2RIE 位置 “1”（允许电压监视 2 中断/ 复位）。	—	将 LVD2CR0.LVD2RIE 位置 “1”（允许电压监视 2 中断/ 复位）。	—

注 1. 可以用 1 条指令同时执行步骤 3 和步骤 4。

注 2. 在以下的任何一种情况下，不需要步骤 1、2、4、7 和 8。

在通过设定为电压监视 2 中断（LVD2CR0.LVD2RI=0）而运行时，如果

- 在停止后只更改 LVD2CR0.LVD2DFDIS 位、LVD2CR0.LVD2FSAMP 位或者 LVD2CR1.LVD2IRQSEL 位、LVD2CR1.LVD2IDTSEL 位的设定而重新运行时

- 在停止后不更改电压检测电路的相关设定而重新运行时

对于通过设定为电压监视 2 复位（LVD2CR0.LVD2RI=1）而运行时的更改，必须通过步骤 1~14 进行设定。

表 8.6 电压监视 2 中断和电压监视 2 复位的相关位的停止设定步骤

步骤	电压监视 2 中断、电压监视 2 的 ELC 事件输出	电压监视 2 复位
1	将 LVD2CR0.LVD2RIE 位置“0”（禁止电压监视 2 中断 / 复位）。	—
2	将 LVD2CR0.LVD2CMPE 位置“0”（禁止输出电压监视 2 电路的比较结果）。	
3 (注 1)	将 LVCMPCR.LVD2E 位置“0”（电压检测 2 电路无效）。	
4	—	将 LVD2CR0.LVD2RIE 位置“0”（禁止电压监视 2 中断 / 复位）。
5	更改除 LVCMPCR.LVD2E 位、LVD2CR0.LVD2RIE 位和 LVD2CR0.LVD2CMPE 位以外的电压检测电路相关寄存器的设定。	

注 1. 在以下的任何一种情况下，不需要步骤 3。

在通过设定为电压监视 2 中断 (LVD2CR0.LVD2RI=0) 而运行时，如果

- 在停止后只更改 LVD2CR0.LVD2DFDIS 位、LVD2CR0.LVD2FSAMP 位或者 LVD2CR1.LVD2IRQSEL 位、LVD2CR1.LVD2IDTSEL[1:0] 位的设定而重新运行时
- 在停止后不更改电压检测电路的相关设定而重新运行时

对于通过设定为电压监视 2 复位 (LVD2CR0.LVD2RI=1) 而运行时的更改，必须通过步骤 1 ~ 5 进行设定。

8.7 事件链接输出功能

电压检测电路具有向事件链接控制器 (ELC) 输出以下事件的功能。

1. 检测到经过 Vdet1 事件的输出

在电压检测 1 电路有效并且允许输出电压检测 1 电路的比较结果的状态下检测到经过 Vdet1 时，输出事件。

2. 检测到经过 Vdet2 事件的输出

在电压检测 2 电路有效并且允许输出电压检测 2 电路的比较结果的状态下检测到经过 Vdet2 时，输出事件。

如果要将 LVD 的事件链接输出功能置为有效，就必须在进行 LVD 的有效设定后，将 ELC 侧的 LVD 事件链接功能置为有效。另外，如果要停止 LVD 的事件链接输出功能，就必须在进行 LVD 的停止设定前，将 ELC 侧的 LVD 事件链接功能置为无效。

8.7.1 中断处理和事件链接的关系

电压检测电路有控制允许或者禁止电压监视 1 中断和电压监视 2 中断的中断允许位。如果产生中断源，就在中断允许位为允许的情况下向 CPU 输出中断请求信号。

与此相对，事件链接输出信号与中断允许位的设定无关，一旦产生中断源，就作为事件信号经由 ELC 输出到其他模块。

在软件待机模式和深度软件待机中也能输出电压监视 1 中断和电压监视 2 中断的请求信号，而有关 ELC 的事件信号的输出如下所示。

- 在软件待机模式中检测到经过 Vdet1/Vdet2 时，由于在软件待机模式期间不提供时钟，所以不输出 ELC 的事件信号。但是，因为保持检测经过 Vdet1/Vdet2 的标志，所以一旦从软件待机模式返回并且重新开始提供时钟，就根据检测经过 Vdet1/Vdet2 的标志输出 ELC 的事件信号。
- 在深度软件待机模式中检测到经过 Vdet1/Vdet2 时，不输出 ELC 的事件信号。

9. 时钟发生电路

9.1 概要

RX210 群内置时钟发生电路。

时钟发生电路的规格和框图分别如表 9.1 和图 9.1 所示。

表 9.1 时钟发生电路的规格

项目	规格
用途	<ul style="list-style-type: none"> 生成给 CPU、DMAC、DTC、ROM 和 RAM 提供的系统时钟 (ICLK)。 生成给外围模块提供的外围模块时钟 (PCLKB、PCLKD)。 外围模块时钟 (PCLKD) 是用于 S12AD 的运行时钟, 外围模块时钟 (PCLKB) 是用于 S12AD 以外的外围模块的运行时钟。 生成给 FlashIF 提供的 FlashIF 时钟 (FCLK)。 生成给外部总线提供的外部总线时钟 (BCLK)。 生成给 CAC 提供的 CAC 时钟 (CACCLK)。 生成给 RTC 提供的 RTC 专用副时钟 (RTCSCLK)。 生成给 IWDWT 提供的 IWDWT 专用低速时钟 (IWDWTCLK)。
运行频率 (注2)	<ul style="list-style-type: none"> ICLK: 50MHz(max) PCLKB: 32MHz(max) PCLKD: 50MHz(max) FCLK: 4MHz ~ 32MHz (注1) BCLK: 25MHz(max) BCLK 引脚输出: 12.5MHz(max) CACCLK: 与各振荡器的时钟相同 RTCSCLK: 32.768kHz IWDWTCLK: 125kHz
主时钟振荡器	<ul style="list-style-type: none"> 谐振器频率: 1MHz ~ 20MHz 外部时钟的输入频率: 20MHz(max) 能连接的谐振器或者附加电路: 陶瓷谐振器、晶体谐振器 连接引脚: EXTAL、XTAL 振荡停止检测功能: 这是在检测到主时钟停止振荡时转换到 LOCO 并且将 MTU2 引脚置为高阻抗的功能。 强制振荡功能
副时钟振荡器	<ul style="list-style-type: none"> 谐振器频率: 32.768kHz 能连接的谐振器或者附加电路: 晶体谐振器 连接引脚: XCIN、XCOUT
PLL 电路	<ul style="list-style-type: none"> 输入时钟源: 主时钟 输入分频比: 能选择 1 分频、2 分频或者 4 分频。 输入频率: 4MHz ~ 12.5MHz 倍增比: 能选择递增 8 倍、10 倍、12 倍、16 倍、20 倍、24 倍或者 25 倍。 VCO 振荡频率: 50MHz ~ 100MHz
高速内部振荡器 (HOCO)	<ul style="list-style-type: none"> 振荡频率: 32MHz/36.864MHz/40MHz/50MHz HOCO 电源控制
低速内部振荡器 (LOCO)	振荡频率: 125kHz
IWDWT 专用低速内部振荡器	振荡频率: 125kHz
BCLK 引脚的输出控制功能	<ul style="list-style-type: none"> 能选择输出 BCLK 时钟或者 High 电平。 输出时钟能选择 BCLK 或者 BCLK 的 2 分频时钟。

注 1. 编程 / 擦除 (P/E) 时的下限值为 4MHz。

注 2. 这是在高速运行模式中的最大工作频率。有关其他运行模式中的最大工作频率, 请参照“11.2.5 运行功率控制寄存器 (OPCCR)”。

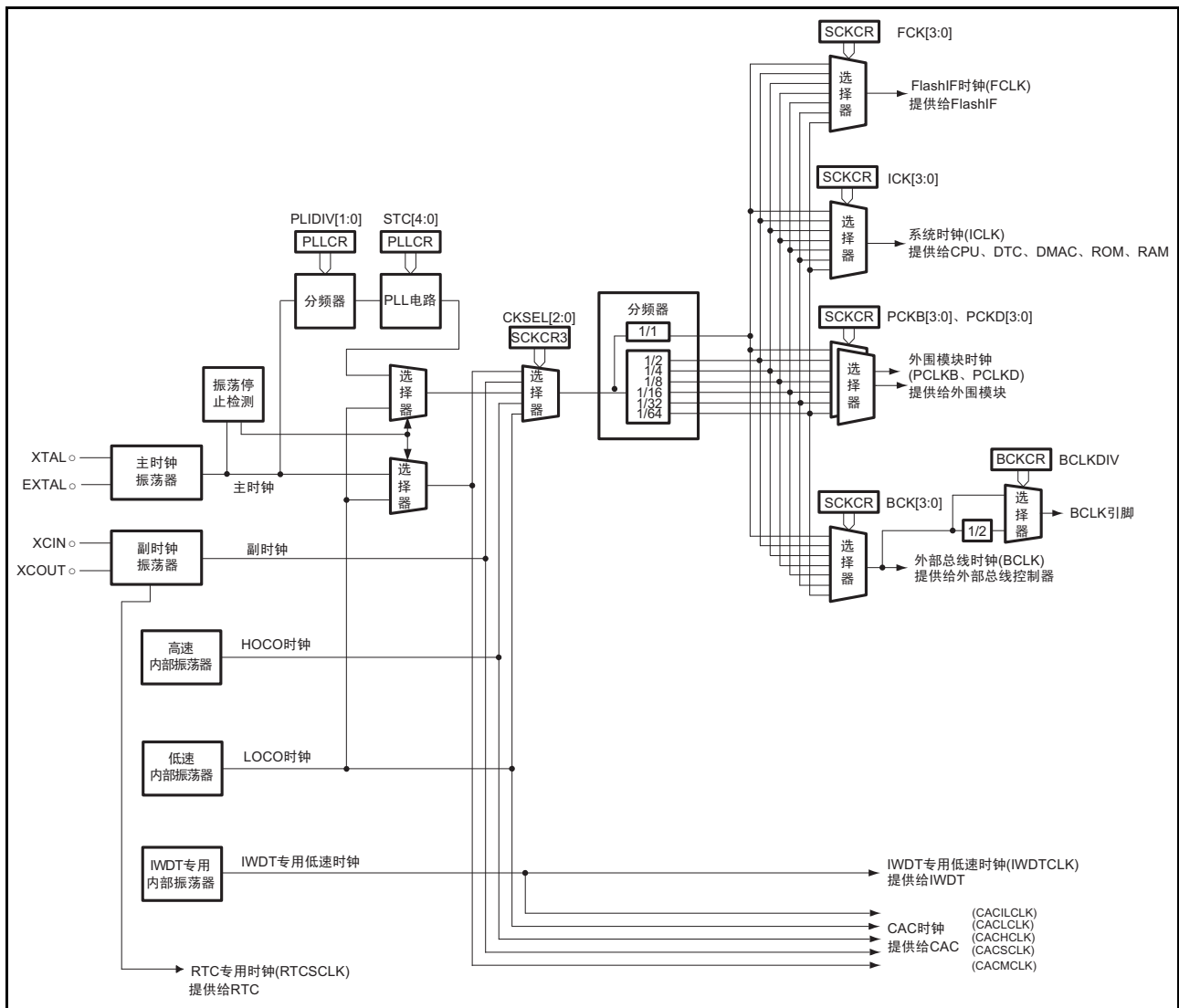


图 9.1 时钟发生电路的框图

时钟发生电路的输入 / 输出引脚如表 9.2 所示。

表 9.2 时钟发生电路的输入 / 输出引脚

引脚名	输入 / 输出	功能
XTAL	输出	连接谐振器的引脚 EXTAL 引脚也能输入外部时钟，详细内容请参照“9.3.2 输入外部时钟的方法”。
EXTAL	输入	
XCIN	输入	连接 32.768kHz 晶体谐振器。
XCOUT	输出	
BCLK	输出	给外部设备提供外部总线时钟（BCLK）。

9.2 寄存器说明

9.2.1 系统时钟控制寄存器 (SCKCR)

地址 0008 0020h

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
FCK[3:0]				ICK[3:0]				PSTOP1	—	—	—	BCK[3:0]			
复位后的值 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0															
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—				PCKB[3:0]				—	—	—	—	PCKD[3:0]			
复位后的值 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0															

位	符号	位名	功能	R/W
b3-b0	PCKD[3:0]	外围模块时钟 D (PCLKD) 选择位 (注 1、注 2)	b3 b0 0000: 1 分频 0001: 2 分频 0010: 4 分频 0011: 8 分频 0100: 16 分频 0101: 32 分频 0110: 64 分频 上述以外: 不能设定	R/W
b7-b4	—	保留位	必须设定“0001b”。	R/W
b11-b8	PCKB[3:0] (注 3)	外围模块时钟 B (PCLKB) 选择位 (注 1)	b11 b8 0000: 1 分频 0001: 2 分频 0010: 4 分频 0011: 8 分频 0100: 16 分频 0101: 32 分频 0110: 64 分频 上述以外: 不能设定	R/W
b15-b12	—	保留位	必须设定“0001b”。	R/W
b19-b16	BCK[3:0]	外部总线时钟 (BCLK) 选择位 (注 1、注 2)	b19 b16 0000: 1 分频 0001: 2 分频 0010: 4 分频 0011: 8 分频 0100: 16 分频 0101: 32 分频 0110: 64 分频 上述以外: 不能设定	R/W
b22-b20	—	保留位	读写值都为“0”。	R/W
b23	PSTOP1	BCLK 引脚输出控制位 (注 3)	0: 从 BCLK 引脚输出 1: 停止从 BCLK 引脚输出 (固定为 High 电平)	R/W
b27-b24	ICK[3:0]	系统时钟 (ICK) 选择位 (注 1、注 2)	b27 b24 0000: 1 分频 0001: 2 分频 0010: 4 分频 0011: 8 分频 0100: 16 分频 0101: 32 分频 0110: 64 分频 上述以外: 不能设定	R/W

位	符号	位名	功能	R/W
b31-b28	FCK[3:0]	FlashIF 时钟 (FCLK) 选择位 (注1)	b31 b28 0 0 0 0: 1 分频 0 0 0 1: 2 分频 0 0 1 0: 4 分频 0 0 1 1: 8 分频 0 1 0 0: 16 分频 0 1 0 1: 32 分频 0 1 1 0: 64 分频 上述以外: 不能设定	R/W

注 1. 当选择 PLL 时, 禁止设定 1 分频。

注 2. ICLK 不能设定低于外部总线时钟的频率。

注 3. 当外部总线有效时, 和 BCLK 引脚兼用的 P53 不能用作 I/O 端口。

注 4. 在系统时钟 (ICLK) 和外部总线时钟 (BCLK) 之间必须满足以下频率关系。

$$ICLK \geq BCLK$$

在不使用 BCLK 时, 必须将 ICLK 和 PCLKB 中设定频率大的一方设定到 SCKCR.BCK 位。但是, 如果该设定使 BCLK 大于 25MHz, 就必须将小于 25MHz 的最大分频比设定大 SCKCR.BCK 位。

如果写 SCKCR 寄存器后的指令为写 SCKCR 寄存器、SCKCR3 寄存器或者 BCKCR 寄存器, 就必须执行以下步骤。

1. 写 SCKCR 寄存器
2. 确认值已写到 SCKCR 寄存器
3. 进入到下一步

在以下情况下禁止改写 SCKCR 寄存器:

- 当运行功率控制寄存器的运行功率控制模式转移状态标志 (OPCCR.OPCMTSF) 为“1” (运行功率控制模式的转移过程中) 时
- 当闪存 P/E 模式入口寄存器的 ROM P/E 模式入口位 0 (FENTRYR.FENTRY0) 为“1” (ROM P/E 模式) 时
- 在从发行用于转移到睡眠模式的 WAIT 指令开始到从睡眠模式返回到正常运行模式为止的期间

PCKD[3:0] 位 (外部模块时钟 D (PCLKD) 选择位)

这些位选择外围模块时钟 (PCLKD) 的频率。

外围模块时钟 (PCLKD) 是用于 S12AD 的运行时钟。

PCKB[3:0] 位 (外部模块时钟 B (PCLKB) 选择位)

这些位选择外围模块时钟 (PCLKB) 的频率。

外围模块时钟 (PCLKB) 是用于 S12AD 以外的外围模块的运行时钟。

BCK[3:0] 位 (外部总线时钟 (BCLK) 选择位)

这些位选择外部总线时钟 (BCLK) 的频率。

PSTOP1 位 (BCLK 引脚输出控制位)

此位控制从 BCLK 引脚输出或者停止输出时钟。停止时输出 High 电平。

ICK[3:0] 位 (系统时钟 (ICLK) 选择位)

这些位选择系统时钟 (ICLK) 的频率。

FCK[3:0] 位 (FlashIF 时钟 (FCLK) 选择位)

这些位选择 FlashIF 时钟 (FCLK) 的频率。

9.2.2 系统时钟控制寄存器 3 (SCKCR3)

地址 0008 0026h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	CKSEL[2:0]		—	—	—	—	—	—	—	—	—
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b7-b0	—	保留位	读写值都为“0”。	R/W
b10-b8	CKSEL[2:0]	时钟源选择位	b10 b8 0 0 0: 选择 LOCO 0 0 1: 选择 HOCO 0 1 0: 选择主时钟振荡器 0 1 1: 选择副时钟振荡器 1 0 0: 选择 PLL 电路 上述以外: 不能设定	R/W
b15-b11	—	保留位	读写值都为“0”。	R/W

在以下情况下禁止改写 SCKCR3 寄存器:

- 当运行功率控制寄存器的运行功率控制模式转移状态标志 (OPCCR.OPCMTSF) 为“1” (运行功率控制模式的转移过程中) 时
- 当闪存 P/E 模式入口寄存器的 ROM P/E 模式入口位 0 (FENTRYR.FENTRY0) 为“1” (ROM P/E 模式) 时
- 在从发行用于转移到睡眠模式的 WAIT 指令开始到从睡眠模式返回到正常运行模式为止的期间

如果在解除复位后以低速内部振荡器进行运行的情况下更改系统时钟的时钟源, 就必须在写此寄存器前, 给地址 0008 0200h 写“00h”。

[写步骤]

1. 允许写 PRCCR.PRC2 位的对象寄存器
给保护寄存器 (PRCCR) 写“A504h”
2. 给地址 0008 0200h 写“00h”
3. 进行写 PRCCR.PRC2 位的对象寄存器
给保护寄存器 (PRCCR) 写“A500h”

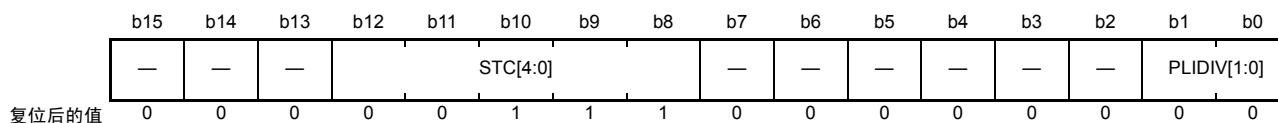
CKSEL[2:0] 位 (时钟源选择位)

从低速内部振荡器 (LOCO)、高速内部振荡器 (HOCO)、主时钟振荡器、副时钟振荡器和 PLL 电路中
选择系统时钟 (ICLK)、外围模块时钟 (PCLKB、PCLKD)、FlashIF 时钟 (FCLK) 和外部总线时钟 (BCLK)
的时钟源。

禁止转换到停止的时钟源。

9.2.3 PLL 控制寄存器 (PLLCR)

地址 0008 0028h



位	符号	位名	功能	R/W
b1-b0	PLIDIV[1:0]	PLL 输入分频比选择位	b1 b0 0 0: 1 分频 0 1: 2 分频 1 0: 4 分频 1 1: 不能设定	R/W
b7-b2	—	保留位	读写值都为“0”。	R/W
b12-b8	STC[5:0]	频率倍增率设定位	b13 b8 0 0 1 1 1: ×8 0 1 0 0 1: ×10 0 1 0 1 1: ×12 0 1 1 1 1: ×16 1 0 0 1 1: ×20 1 0 1 1 1: ×24 1 1 0 0 0: ×25 上述以外: 不能设定	R/W
b15-b13	—	保留位	读写值都为“0”。	R/W

当 PLLCR2.PLEN 位为“0”（PLL 运行）时，禁止写 PLLCR 寄存器。

PLIDIV[1:0] 位（PLL 输入分频比选择位）

这些位选择 PLL 时钟源的输入分频比。

必须将 PLIDIV[1:0] 位设定在 PLL 输入频率（4 ~ 12.5MHz）的范围内。

STC[4:0] 位（频率倍增率设定位）

这些位设定 PLL 的频率倍增率。

必须将 STC[4:0] 位设定在 PLL 的 VCO 振荡频率（50 ~ 100MHz）的范围内。

9.2.4 PLL 控制寄存器 2 (PLLCR2)

地址 0008 002Ah

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	PLLEN
0	0	0	0	0	0	0	1

复位后的值

位	符号	位名	功能	R/W
b0	PLLEN	PLL 停止控制位	0: PLL 运行 1: PLL 停止	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

必须在设定“11. 低功耗功能”中的 PLL 等待控制寄存器后，设定 PLLCR2 寄存器。

PLLEN 位 (PLL 停止控制位)

此位控制 PLL 的运行或者停止。

必须在通过 PLLEN 位将 PLL 设定为运行并且经过 PLL 时钟振荡稳定待机时间 (t_{PLLWT1} 或者 t_{PLLWT2}) 后，开始使用 PLL 时钟。

从设定为运行后到振荡稳定为止，PLL 需要一定的时间；从设定为停止后到振荡停止为止，PLL 也需要一定的时间。因此必须注意：有关运行的开始和停止，有以下限制：

- 要在将 PLL 设定为停止后再次设定为运行时，停止期间必须至少为 5 个 PLL 时钟周期的时间。
- 必须在 PLL 时钟振荡稳定的状态下进行 PLL 的停止设定。
- 与是否选择作为系统时钟无关，在将 PLL 设定为运行后转移到软件待机模式或者深度软件待机模式时，必须在振荡稳定的状态下执行 WAIT 指令。
- 要在将 PLL 设定为停止后转移到软件待机模式或者深度软件待机模式时，必须在将 PLL 设定为停止并且至少等待 2 个 PLL 时钟周期后执行 WAIT 指令。

当通过系统时钟控制寄存器 3 的时钟源选择位 (SCKCR3.CKSEL[2:0]) 选择了 PLL 时，禁止给 PLLEN 位写“1” (PLL 停止运行)。

当通过运行功率控制寄存器的运行功率控制模式选择位 (OPCCR.OPCM[2:0]) 选择了低速运行模式 1 或者低速运行模式 2 时，禁止给 PLLEN 位写“0” (PLL 运行)。

9.2.5 外部总线时钟控制寄存器 (BCKCR)

地址 0008 0030h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	BCLKDIV
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	BCLKDIV	BCLK 引脚输出选择位	0: BCLK 1: BCLK 的 2 分频	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

在以下情况下禁止改写 BCKCR 寄存器:

- 当运行功率控制寄存器的运行功率控制模式转移状态标志 (OPCCR.OPCMTSF) 为“1” (运行功率控制模式的转移过程中) 时
- 当闪存 P/E 模式入口寄存器的 ROM P/E 模式入口位 0 (FENTRYR.FENTRY0) 为“1” (ROM P/E 模式) 时
- 在从发行用于转移到睡眠模式的 WAIT 指令开始到从睡眠模式返回到正常运行模式为止的期间

BCLKDIV 位 (BCLK 引脚输出选择位)

此位选择从 BCLK 引脚输出的时钟。

能选择由 SCKCR.BCK[3:0] 位选择频率的 BCLK 或者 BCLK 的 2 分频时钟。

9.2.6 主时钟振荡器的控制寄存器 (MOSCCR)

地址 0008 0032h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	MOSTP
0	0	0	0	0	0	0	1

复位后的值

位	符号	位名	功能	R/W
b0	MOSTP	主时钟振荡器停止位	0: 主时钟振荡器振荡 1: 主时钟振荡器停止振荡	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

必须在设定“11. 低功耗功能”中的主时钟振荡器等待控制寄存器后，设定 MOSCCR 寄存器。

MOSTP 位 (主时钟振荡器停止位)

此位控制主时钟振荡器的振荡或者停止。

通过 MOSTP 位将振荡器振荡设定为强制振荡制，就能使主时钟振荡器振荡。

要改写 MOSTP 位时，必须在写后通过读该位来确认是否已被改写，然后执行后续的指令（请参照“5. I/O 寄存器”的“(2) 写 I/O 寄存器时的注意事项”）。

在通过连接晶体谐振器来提供主时钟的情况下，必须在通过 MOSTP 位将主时钟振荡器设定为振荡并且经过主时钟振荡稳定待机时间（晶体）(tMAINOSCWT) 后，开始使用主时钟。

在通过输入外部时钟来提供主时钟的情况下，必须在通过 MOSTP 位将主时钟振荡器设定为振荡并且经过 EXTAL 外部时钟输入待机时间 (tEXWT) 后，开始使用主时钟。

从设定为振荡后到振荡稳定为止，主时钟振荡器需要一定的时间；从设定为停止后到振荡停止为止，主时钟振荡器也需要一定的时间。因此必须注意：有关振荡的开始和停止，有以下限制：

- 要在将主时钟振荡器设定为停止后再次设定为振荡时，停止期间必须至少为 5 个主时钟周期的时间。
- 必须在主时钟振荡器振荡稳定的状态下进行主时钟振荡器的停止设定。
- 与是否选择作为系统时钟无关，要在将主时钟振荡器设定为振荡后转移到软件待机模式或者深度软件待机模式时，必须在主时钟振荡器振荡稳定的状态下执行 WAIT 指令。
- 要在将主时钟振荡器设定为停止后转移到软件待机模式或者深度软件待机模式时，必须在将主时钟设定为停止并且至少等待 2 个主时钟周期后执行 WAIT 指令。

当通过系统时钟控制寄存器 3 的时钟源选择位 (SCKCR3.CKSEL[2:0]) 选择了主时钟振荡器或者通过 SCKCR3.CKSEL[2:0] 位选择了 PLL 时，禁止给 MOSTP 位写“1”（主时钟振荡器停止振荡）。

9.2.7 副时钟振荡器的控制寄存器 (SOSCCR)

地址 0008 0033h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	SOSTP
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	SOSTP	副时钟振荡器停止位	0: 副时钟振荡器振荡 1: 副时钟振荡器停止振荡	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

必须在设定“11. 低功耗功能”中的副时钟振荡器等待控制寄存器后，设定 SOSCCR 寄存器。

SOSTP 位 (副时钟振荡器停止位)

此位选择副时钟振荡器的振荡或者停止。

通过 SOSTP 位和 RTC 控制寄存器 3 的副时钟振荡器控制位 (RCR3.RTCEN) 控制副时钟振荡器的振荡或者停止。如果将这两位中的任意一位设定为运行，副时钟振荡器就变为振荡状态。

要改写 SOSTP 位或者 RCR3.RTCEN 位时，必须在写后通过读该位来确认是否已被改写，然后执行后续的指令 (请参照“5. I/O 寄存器”的“(2) 写 I/O 寄存器时的注意事项”)。

必须在通过 SOSTP 位或者 RCR3.RTCEN 位将副时钟振荡器设定为振荡并且经过副时钟振荡稳定待机时间 (tSUBOSCWT) 后，开始使用副时钟。

从设定为振荡后到振荡稳定为止，副时钟振荡器需要一定的时间；从设定为停止后到振荡停止为止，副时钟振荡器也需要一定的时间。因此必须注意：有关振荡的开始和停止，有以下限制：

- 要在将副时钟振荡器设定为停止后再次设定为振荡时，停止期间必须至少为 5 个副时钟周期的时间。
- 必须在副时钟振荡器振荡稳定的状态下进行副时钟振荡器的停止设定。
- 与是否选择作为系统时钟无关，要在将副时钟振荡器设定为振荡后转移到软件待机模式或者深度软件待机模式时，必须在振荡稳定的状态下执行 WAIT 指令。
- 要在将副时钟振荡器设定为停止后转移到软件待机模式或者深度软件待机模式时，必须在将副时钟振荡器设定为停止并且至少等待 2 个副时钟周期后执行 WAIT 指令。

当通过系统时钟控制寄存器 3 的时钟源选择位 (SCKCR3.CKSEL[2:0]) 选择了副时钟振荡器时，禁止给 SOSTP 位写“1” (副时钟振荡器停止振荡)。

9.2.8 低速内部振荡器的控制寄存器 (LOCOCR)

地址 0008 0034h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	LCSTP
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	LCSTP	LOCO 停止位	0: LOCO 振荡 1: LOCO 停止振荡	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

LCSTP 位 (LOCO 停止位)

此位控制 LOCO 的振荡或者停止。

在通过 LCSTP 位将 LOCO 从停止改为振荡后使用 LOCO 时钟的情况下，必须在经过 LOCO 时钟振荡稳定待机时间 (tLOCOWT) 后，开始使用 LOCO 时钟。

从设定为振荡后到振荡稳定为止，LOCO 需要一定的时间；从设定为停止后到振荡停止为止，LOCO 也需要一定的时间。因此必须注意：有关振荡的开始和停止，有以下的限制：

- 要在将 LOCO 设定为停止后再次设定为振荡时，停止期间必须至少为 5 个 LOCO 时钟周期的时间。
- 必须在 LOCO 振荡稳定的状态下进行 LOCO 的停止设定。
- 与是否选择作为系统时钟无关，要在将 LOCO 设定为振荡后转移到软件待机模式或者深度软件待机模式时，必须在 LOCO 振荡稳定的状态下执行 WAIT 指令。
- 要在将 LOCO 设定为停止后转移到软件待机模式或者深度软件待机模式时，必须在将 LOCO 设定为停止并且至少等待 3 个 LOCO 时钟周期后执行 WAIT 指令。

当通过系统时钟控制寄存器 3 的时钟源选择位 (SCKCR3.CKSEL[2:0]) 选择了 LOCO 时，禁止给 LCSTP 位写“1” (LOCO 停止振荡)。

当通过振荡停止检测控制寄存器的振荡停止检测功能允许位 (OSTDCR.OSTDE) 将振荡停止检测功能设定为有效时，禁止给 LCSTP 位写“1” (LOCO 停止振荡)。

9.2.9 IWDT 专用内部振荡器的控制寄存器 (ILOCOCR)

地址 0008 0035h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	ILCSTP
0	0	0	0	0	0	0	1

复位后的值

位	符号	位名	功能	R/W
b0	ILCSTP	IWDT 专用内部振荡器停止位	0: IWDT 专用内部振荡器振荡 1: IWDT 专用内部振荡器停止振荡	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

当选项功能选择寄存器 0 的 IWDT 启动模式选择位 (OFS0.IWDTSTRT) 为“0” (IWDT 振荡) 时, ILOCOCR 寄存器的设定无效; 当 OFS0.IWDTSTRT 位为“1” (IWDT 停止振荡) 时, ILOCOCR 寄存器的设定有效。在 ILOCOCR 寄存器有效并且 ILCSTP 位变为“0” (IWDT 专用内部振荡器振荡) 后, 不能将 ILCSTP 位置“1” (IWDT 专用内部振荡器停止振荡)。

ILCSTP 位 (IWDT 专用内部振荡器停止位)

此位控制 IWDT 专用内部振荡器的振荡或者停止。

如果通过 ILCSTP 位将 IWDTLOCO 从停止改为振荡, 就在经过相当于 LOCO 振荡稳定待机时间 (t_{LOCOWT}) 的一定时间后开始给 LSI 内部提供时钟。如果要使用 IWDT 专用低速时钟, 就必须在经过此待机时间后开始使用。

要在将 IWDT 专用内部振荡器设定为振荡后转移到软件待机模式或者深度软件待机模式时, 必须在振荡稳定的状态下执行 WAIT 指令。

9.2.10 高速内部振荡器的控制寄存器（HOCO CR）

地址 0008 0036h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	HCSTP

复位后的值 0 0 0 0 0 0 0 0/1
(注 1)

注 1. 当选项功能选择寄存器 1 的 HOCO 振荡有效位（OFS1.HOCOEN）位为“0”时，HCSTP 位的复位后的值为“0”。当 OFS1.HOCOEN 位为“1”时，HCSTP 位的复位后的值为“1”。

位	符号	位名	功能	R/W
b0	HCSTP	HOCO 停止位	0: HOCO 振荡 1: HOCO 停止	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

必须在设定“11. 低功耗功能”中的 HOCO 等待控制寄存器 2 后，设定 HOCO CR 寄存器。

HCSTP 位（HOCO 停止位）

此位控制 HOCO 的振荡或者停止。

如果通过 HCSTP 位将 HOCO 从停止改为振荡，就必须在经过 HOCOWTCR2.HSTS2[3:0] 位设定的待机时间后，开始给 LSI 内部提供时钟。如果要使用 HOCO，就必须在经过此待机时间后开始使用。

从设定为振荡后到振荡稳定为止，HOCO 需要一定的时间；从设定为停止后到振荡停止为止，HOCO 也需要一定的时间。因此必须注意：有关振荡的开始和停止，有以下限制：

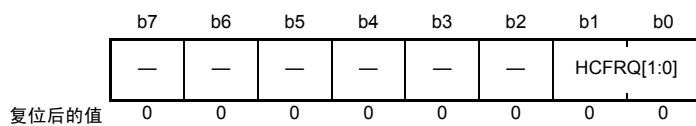
- 要在将 HOCO 设定为停止后再次设定为振荡时，停止期间必须至少为 5 个 HOCO 时钟周期的时间。
- 必须在 HOCO 振荡稳定的状态下进行 HOCO 的停止设定。
- 与是否选择作为系统时钟无关，要在将 HOCO 设定为振荡后转移到软件待机模式或者深度软件待机模式，必须在 HOCO 振荡稳定的状态下执行 WAIT 指令。
- 要在将 HOCO 设定为停止后转移到软件待机模式或者深度软件待机模式，必须在将 HOCO 设定为停止并且至少等待 2 个 HOCO 时钟周期后执行 WAIT 指令。

当通过系统时钟控制寄存器 3 的时钟源选择位（SCKCR3.CKSEL[2:0]）选择了 HOCO 时，禁止给 HCSTP 位写“1”（HOCO 停止振荡）。

当通过运行功率控制寄存器的运行功率控制模式选择位（OPCCR.OPCM[2:0]）选择了低速运行模式 2 时，禁止给 HCSTP 位写“0”（HOCO 振荡）。

9.2.11 高速内部振荡器的控制寄存器 2 (HOCOCR2)

地址 0008 0037h



位	符号	位名	功能	R/W
b1-b0	HCFRQ[1:0]	HOCO 频率设定位	b1 b0 0 0: 32MHz 0 1: 36.864MHz 1 0: 40MHz 1 1: 50MHz	R/W
b7-b2	—	保留位	读写值都为“0”。	R/W

9.2.12 振荡停止检测控制寄存器 (OSTDCR)

地址 0008 0040h

b7	b6	b5	b4	b3	b2	b1	b0
OSTDE	—	—	—	—	—	—	OSTDIE
复位后的值	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	OSTDIE	振荡停止检测中断允许位	0: 禁止振荡停止检测中断并且没有将振荡停止检测通知 POE 1: 允许振荡停止检测中断并且将振荡停止检测通知 POE	R/W
b6-b1	—	保留位	读写值都为“0”。	R/W
b7	OSTDE	振荡停止检测功能允许位	0: 振荡停止检测功能无效 1: 振荡停止检测功能有效	R/W

OSTDCR 寄存器是控制主时钟振荡器的振荡停止检测功能的寄存器。

OSTDIE 位 (振荡停止检测中断允许位)

必须在将 OSTDIE 位置“0”后清除振荡停止检测状态寄存器的振荡停止检测标志 (OSTDSR.OSTDF)。此后, 如果要再次将 OSTDIE 位置“1”, 就必须至少等待 2 个 PCLKB 周期。能通过读由 PCLKB 定义存取周期数的 I/O 寄存器, 至少确保 2 个 PCLKB 周期的等待时间。

OSTDE 位 (振荡停止检测功能允许位)

此位设定振荡停止检测功能的有效或者无效。

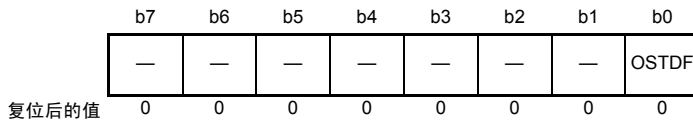
如果将 OSTDE 位置“1” (振荡停止检测功能有效), LOCO 停止位 (LOCOCR.LCSTP) 就变为“0”并且 LOCO 振荡。在振荡停止检测功能有效期间, 不能停止 LOCO 的振荡。即使给 LOCOCR.LCSTP 位写“1” (LOCO 停止振荡), 写操作也无效。

当振荡停止检测状态寄存器的振荡停止检测标志 (OSTDSR.OSTDF) 为“1” (检测到主时钟振荡停止振荡) 时, 即使给 OSTDE 位写“0”, 写操作也无效。

当 OSTDE 位为“1”时, 不能转移到软件待机模式和深度软件待机模式。如果要转移到软件待机模式和深度软件待机模式, 就必须在将 OSTDE 位置“0”后执行 WAIT 指令。

9.2.13 振荡停止检测状态寄存器 (OSTDSR)

地址 0008 0041h



位	符号	位名	功能	R/W
b0	OSTDF	振荡停止检测标志	0: 未检测到主时钟停止振荡 1: 检测到主时钟停止振荡	R(W) (注1)
b7-b1	—	保留位	读取值为“0”，写操作无效	R

注 1. 只能写“0”。

OSTDF 标志 (振荡停止检测标志)

这是表示主时钟状态的标志。当 OSTDF 标志为“1”时，表示检测到主时钟停止振荡。

在检测到主时钟停止振荡后，即使主时钟重新开始振荡，OSTDF 标志也不变为“0”。通过在读“1”后写“0”，使 OSTDF 标志变为“0”。到 OSTDF 为“0”被反映为读取值为止，至少需要等待 3 个 ICLK 周期。如果在主时钟停止振荡的状态下将 OSTDF 标志置“0”，OSTDF 标志就在变为“0”后再次变为“1”。

当通过系统时钟控制寄存器 3 的时钟源选择位 (SCKCR3.CKSEL[2:0]) 选择了主时钟振荡器 (“010b”) 或者 PLL (“100b”) 时，不能将 OSTDF 标志置“0”。必须在将时钟源转换到不是主时钟振荡器和 PLL 后将 OSTDF 标志置“0”。

[为“1”的条件]

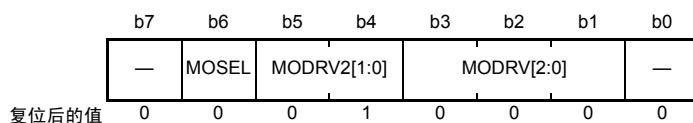
- 在 OSTDCR.OSTDE 位为“1” (振荡停止检测功能有效) 的状态下主时钟停止振荡时

[为“0”的条件]

- 在 SCKCR3.CKSEL[2:0] 位不是“010b”和“100b”的情况下读“1”后写“0”时

9.2.14 主时钟振荡器的强制振荡控制寄存器 (MOFCR)

地址 0008 C293h



位	符号	位名	功能	R/W
b0	—	保留位	读写值都为“0”。	R/W
b3-b1	MODRV[2:0]	主时钟振荡器驱动能力切换位	b3 b2 b1 0 0 0: 非 16MHz ~ 20MHz 陶瓷谐振器引脚产品 1 1 1: 16MHz ~ 20MHz 陶瓷谐振器引脚产品 上述以外: 不能设定	R/W
b5-b4	MODRV2[1:0]	主时钟振荡器驱动能力切换 2 位	b5 b4 0 1: 1MHz ~ 8MHz 1 0: 8.1MHz ~ 15.9MHz 1 1: 16MHz ~ 20MHz 上述以外: 不能设定	R/W
b6	MOSEL	主时钟振荡器切换位	0: 谐振器 1: 外部振荡输入	R/W
b7	—	保留位	读写值都为“0”。	R/W

EXTAL/XTAL 引脚和端口兼用，在初始设定状态下用作端口。

MODRV[2:0] 位（主时钟振荡器驱动能力切换位）

这些位切换主时钟振荡器的驱动能力。

MODRV2[1:0] 位（主时钟振荡器驱动能力切换 2 位）

这些位切换主时钟振荡器的驱动能力。

MOSEL 位（主时钟振荡器切换位）

此位切换主时钟振荡器的振荡源。

9.2.15 高速内部振荡器的电源控制寄存器 (HOCOPCR)

地址 0008 C294h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	HOCOP CNT
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	HOCOPCNT	高速内部振荡器电源控制位	0: HOCO 的电源 ON 1: HOCO 的电源 OFF	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

HOCOPCR 寄存器是控制高速内部振荡器 (HOCO) 电源的寄存器。

HOCOPCNT 位 (高速内部振荡器电源控制位)

这是控制 HOCO 电源的位。

当此位为“0”时，接通 HOCO 的电源，能使高速内部振荡器振荡。

当此位为“1”时，切断 HOCO 的电源，能降低功耗。

要将 HOCOPCNT 位置“1”时，必须将高速内部振荡器控制寄存器的 HOCO 停止位 (HOCOCCR.HCSTP) 置“1” (HOCO 停止振荡)。

在将 HOCOPCNT 位从“1”改为“0”后到将 HOCOCCR.HCSTP 位置“0”为止，需要稳定等待时间。详细内容请参照“41. 电特性”。

在以下情况下不能改写 HOCOPCNT 位的值。

- 当通过系统时钟控制寄存器3的时钟源选择位 (SCKCR3.CKSEL[2:0]) 选择HOCO作为时钟源时
- 当通过运行功率控制寄存器的运行功率控制模式选择位 (OPCCR.OPCM[2:0]) 设定为低速运行模式1或者低速运行模式2时

9.3 主时钟振荡器

给主时钟振荡器提供时钟的方法有连接谐振器和输入外部时钟的方法。

9.3.1 连接谐振器的方法

连接谐振器时的连接例子如图 9.2 所示。

必须根据需要插入阻尼电阻 (R_d)。电阻值因谐振器和振荡驱动能力而不同，因此必须设定为谐振器厂商的推荐值。如果谐振器厂商有外加反馈电阻 (R_f) 的指示，就必须根据该指示，在 EXTAL 和 XTAL 之间插入 R_f 。

在连接谐振器提供时钟的情况下，连接的谐振器必须在表 9.1 的主时钟振荡器的谐振器频率的范围内。

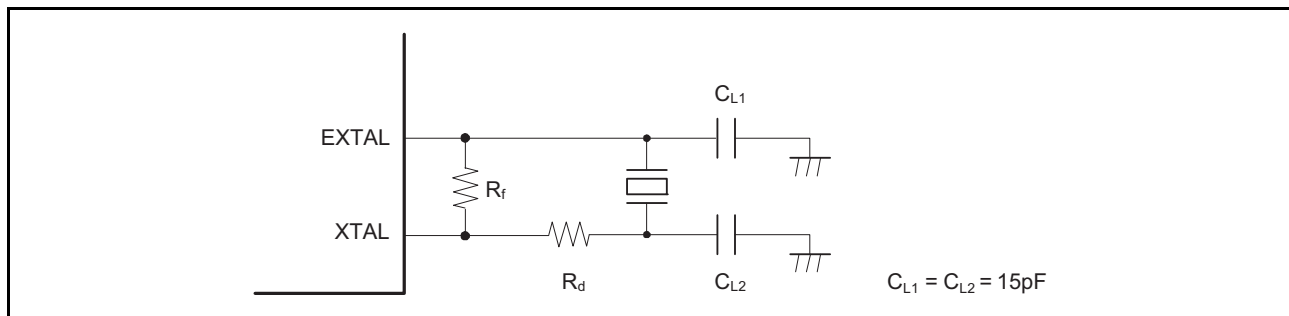


图 9.2 晶体谐振器的连接例子

表 9.3 阻尼电阻 (参考值)

频率 (MHz)	1	8	16	20
R_d (Ω)	750	0	0	0

晶体谐振器的等效电路如图 9.3 所示，必须使用表 9.4 所示特性的晶体谐振器。

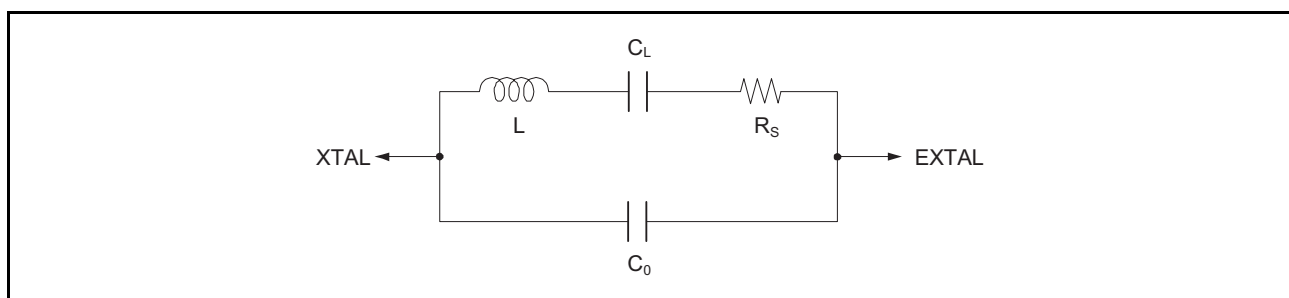


图 9.3 晶体谐振器的等效电路

表 9.4 晶体谐振器的特性 (参考值)

频率 (MHz)	1	8	16	20
R_S max (Ω)	480	14	16.8	8.1
C_0 max (pF)	3	2.2	3	3.5

9.3.2 输入外部时钟的方法

输入外部时钟的连接例子如图 9.4 所示。在将 XTAL 引脚置为开路时，寄生电容必须小于等于 5pF。

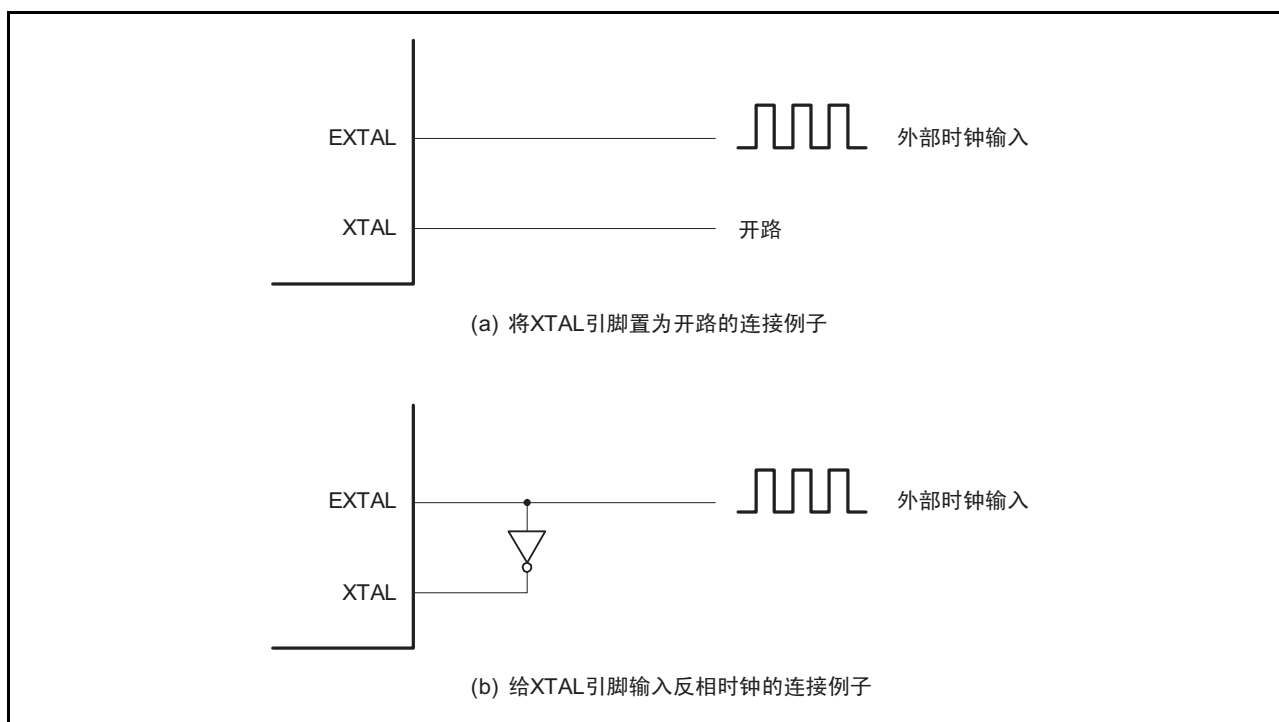


图 9.4 外部时钟的连接例子

9.3.3 有关输入外部时钟的注意事项

只能在主时钟振荡器停止振荡时更改外部时钟的输入频率。在主时钟振荡器停止位 (MOSCCR.MOSTP) 已被置“0” (主时钟振荡器振荡) 的期间，不能更改外部时钟的输入频率。

9.4 副时钟振荡器

给副时钟振荡器提供时钟的方法有连接晶体谐振器的方法。

9.4.1 连接 32.768kHz 晶体谐振器的方法

如图 9.5 所示，为了给副时钟振荡器提供时钟而连接 32.768kHz 晶体谐振器。

必须根据需要插入阻尼电阻 (Rd)。电阻值因谐振器和振荡驱动能力而不同，因此必须设定为谐振器厂商的推荐值。如果谐振器厂商有外加反馈电阻 (Rf) 的指示，就必须根据该指示，在 XCIN 和 XCOU 之间插入 Rf。在连接谐振器提供时钟的情况下，连接的谐振器必须在表 9.1 的副时钟振荡器的谐振器频率的范围内。

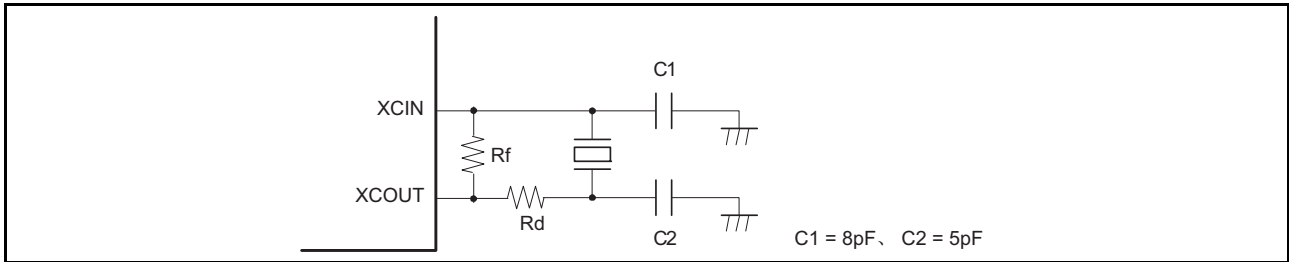


图 9.5 32.768kHz 晶体谐振器的连接例子

32.768kHz 晶体谐振器的等效电路如图 9.6 所示，必须使用表 9.5 所示特性的晶体谐振器。

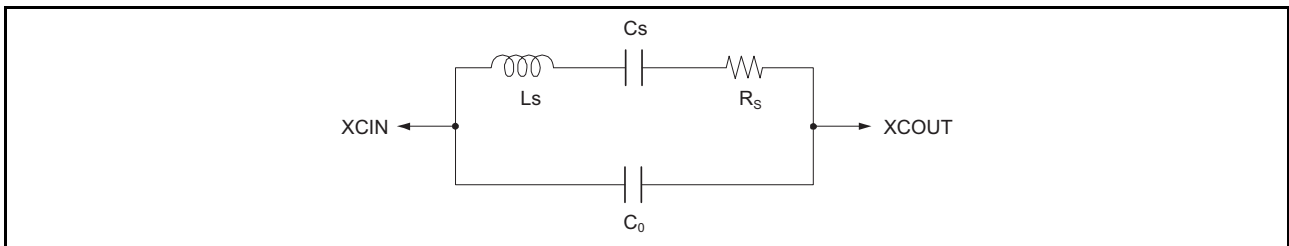


图 9.6 晶体谐振器的等效电路

表 9.5 晶体谐振器的特性 (参考值)

频率 (kHz)	32.768 (低 CL)	32.768 (标准 CL)
R _S max (kΩ)	37	35
C ₀ max (pF)	0.9	2.0

9.4.2 不使用副时钟情况下的引脚处理

如图 9.7 所示，在不使用副时钟的情况下，必须将 XCIN 引脚连接 VSS 并且将 XCOU 引脚置为开路。

在不连接谐振器的情况下，必须将副时钟振荡器停止位 (SOSCCR.SOSTP) 置“1” (停止振荡) 并且将 RTC 控制寄存器 3 的副时钟振荡器控制位 (RCR3.RTCEN) 置“0” (禁止从副时钟振荡器输入时钟)。

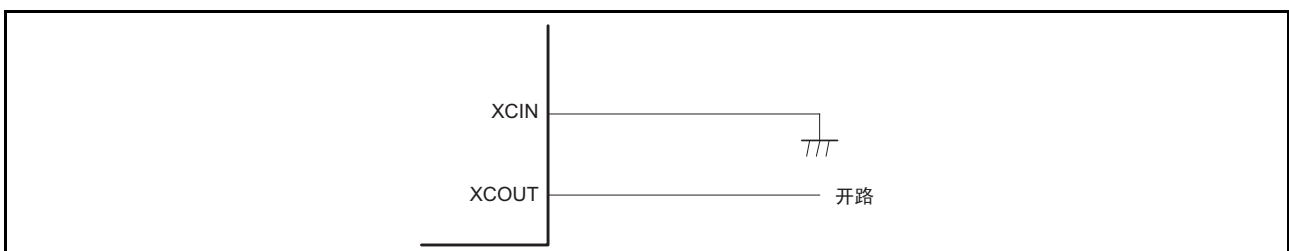


图 9.7 不使用副时钟情况下的引脚处理

9.5 振荡停止检测功能

9.5.1 振荡停止检测和检测后的运行

振荡停止检测功能检测主时钟振荡器的停止，并且作为系统时钟的时钟源，代替主时钟和 PLL 时钟提供低速内部振荡器输出的低速时钟。

能在检测到振荡停止时产生振荡停止检测中断请求，也能在检测到振荡停止时将 MTU2 的输出强制置为高阻抗。详细内容请参照“21. 多功能定时器脉冲单元 2 (MTU2a)”和“22. 端口输出允许 2 (POE2a)”。

如果输入时钟因主时钟振荡器发生异常等而在一定期间内变为“0”或者“1”（参照“表 41.50 振荡停止检测电路特性”），RX210 群就检测到主时钟停止振荡。

一旦检测到振荡停止，时钟源选择位 (SCKCR3.CKSEL[2:0]) 选择的主时钟和 PLL 时钟就通过各自前段的选择器转换到 LOCO 时钟。因此，如果在选择主时钟或者 PLL 时钟作为系统时钟的时钟源的状态下检测到振荡停止，CKSEL[2:0] 位的设定值就不变而将系统时钟的时钟源转换到 LOCO 时钟。

振荡停止检测标志 (OSTDSR.OSTDF) 控制主时钟和 LOCO 时钟的转换以及 PLL 时钟和 LOCO 时钟的转换。如果将 OSTDF 标志置“1”，就转换到 LOCO 时钟；如果将 OSTDF 标志置“0”，就恢复主时钟或者 PLL 时钟。但是，在通过 CKSEL[2:0] 位选择了主时钟或者 PLL 时钟的情况下，不能将 OSTDF 标志置“0”。如果想在检测到振荡停止后将时钟源恢复为主时钟或者 PLL 时钟，就必须在将 CKSEL[2:0] 位的设定更改为不是主时钟和 PLL 时钟后将 OSTDF 标志置“0”。此后，必须确认 OSTDF 标志不是“1”，并且在经过规定的振荡稳定时间后将 CKSEL[2:0] 位的设定更改为主时钟或者 PLL 时钟。

在解除复位后，主时钟振荡器停止振荡，振荡停止检测功能无效。要将振荡停止检测功能置为有效时，必须使主时钟振荡器振荡，并且在经过规定的振荡稳定时间后写振荡停止检测功能允许位 (OSTDCR.OSTDE)。

因为振荡停止检测功能是防止由外部原因导致主时钟停止的功能，所以要通过软件使主时钟振荡器停止振荡或者转移到软件待机模式和深度软件待机模式时，必须预先将振荡停止检测功能置为无效。

通过振荡停止检测而转换到 LOCO 时钟的时钟是作为系统时钟源的主时钟、PLL 时钟和 CAC 主时钟 (CACMCLK)。LOCO 时钟运行时的系统时钟 (ICLK) 的频率取决于 LOCO 振荡频率和系统时钟选择位 (SCKCR.ICK[3:0]) 的分频比的设定。

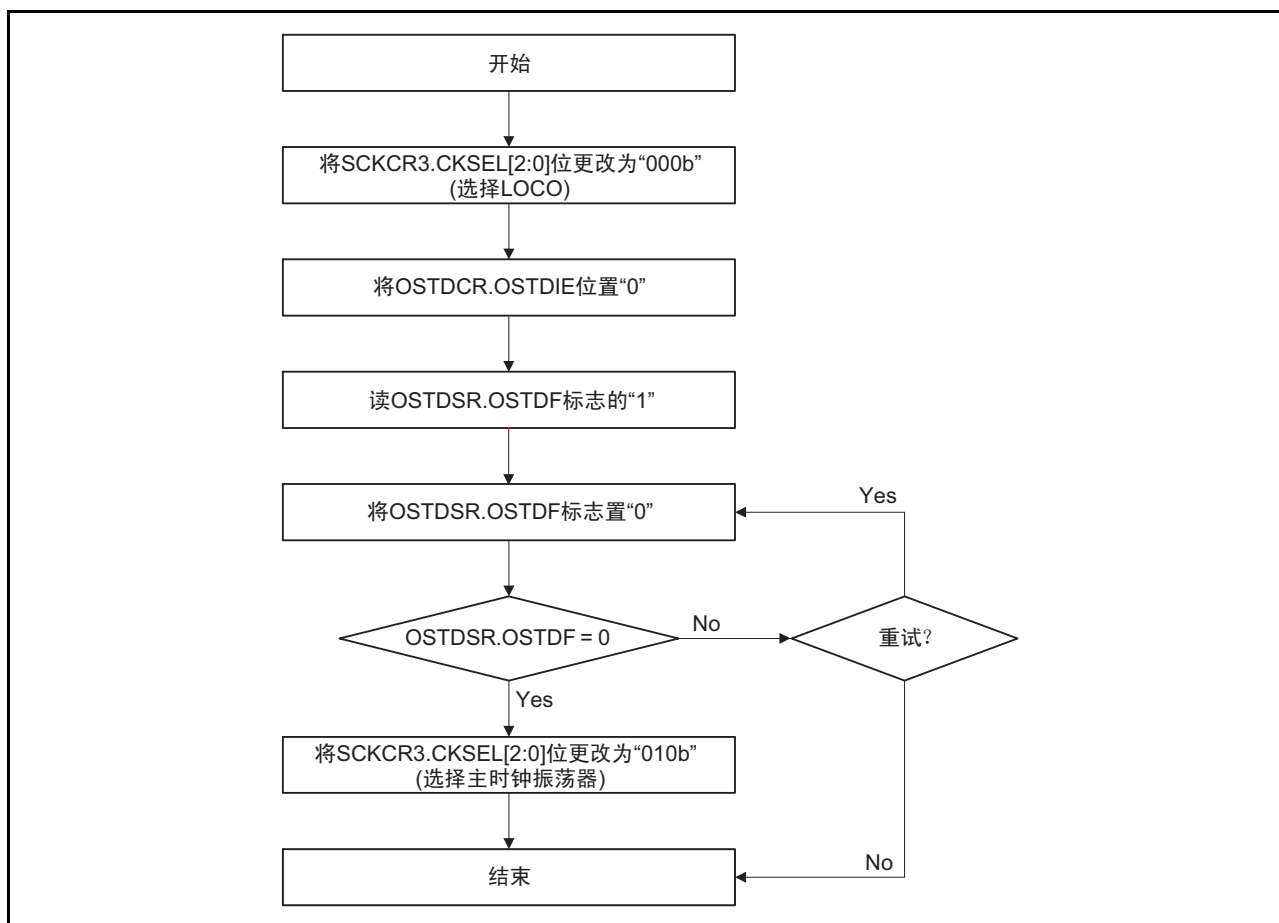


图 9.8 从振荡停止检测恢复的流程图例子

9.5.2 振荡停止检测中断

当振荡停止检测中断允许位（OSTDCR.OSTDIE）为“1”（允许振荡停止检测中断）时，如果振荡停止检测标志（OSTDSR.OSTDF）变为“1”，就产生振荡停止检测中断（OSTDI）请求。此时，将主时钟振荡器停止振荡通知到端口输出允许2（POE）。在POE接受振荡停止的通知后将输入电平控制/状态寄存器3的OSTST高阻抗标志（ICSR3.OSTSTF）置“1”。**必须注意：**在检测到振荡停止后的10个PCLK周期期间，不能写ICSR3.OSTSTF标志。必须在将振荡停止检测中断允许位（OSTDCR.OSTDIE）置“0”后清除OSTDSR.OSTDF标志。此后，要再次将OSTDCR.OSTDIE位置“1”时，必须在至少等待2个PCLKB周期后进行设定。能通过读PCLKB定义的I/O寄存器，确保存取周期数至少为2个PCLKB周期的等待时间。

振荡停止检测中断是非屏蔽中断。因为在解除复位后的初始状态下为“禁止非屏蔽中断”，所以在使用振荡停止检测中断时，必须通过软件将非屏蔽中断置为有效。详细内容请参照“14. 中断控制器（ICUb）”。

9.6 PLL 电路

PLL 电路具有倍增振荡器频率的功能。

9.7 内部时钟

作为内部时钟的时钟源，有主时钟、副时钟、HOCO 时钟、LOCO 时钟、PLL 时钟和 IWDT 专用低速时钟，由这些时钟生成如下所示的 7 种内部时钟。

1. CPU、DMAC、DTC、ROM 和 RAM 的运行时钟：系统时钟（ICLK）
2. 外围模块的运行时钟：外围模块时钟（PCLKB、PCLKD）
3. FlashIF 的运行时钟：FlashIF 时钟（FCLK）
4. 外部总线控制器、外部引脚输出时钟：外部总线时钟（BCLK）
5. CAC 模块的运行时钟：CAC 时钟（CACCLK）
6. RTC 模块的运行时钟：RTC 专用副时钟（RTCSCLK）
7. IWDT 模块的运行时钟：IWDT 专用低速时钟（IWDTCLK）

通过选择分频比的 SCKCR.FCK[3:0] 位、SCKCR.ICK[3:0] 位、SCKCR.BCK[3:0] 位、SCKCR.PCKB[3:0] 位和 SCKCR.PCKD[3:0] 位、选择时钟源的 SCKCR3.CKSEL[2:0] 位以及选择 PLL 电路频率的 PLLCR.STC[4:0] 位和 PLLCR.PLIDIV[1:0] 位的组合，设定内部时钟的频率。在改写各位后，以更改后的频率运行。

9.7.1 系统时钟

系统时钟（ICLK）是 CPU、DMAC、DTC、ROM 和 RAM 的运行时钟。

通过 SCKCR.ICK[3:0] 位、SCKCR3.CKSEL[2:0] 位、PLLCR.STC[4:0] 位和 PLLCR.PLIDIV[1:0] 位设定 ICLK 的频率。

9.7.2 外围模块时钟

外围模块时钟（PCLKB、PCLKD）是外围模块的运行时钟。

通过 SCKCR.PCKB[3:0] 位、SCKCR.PCKD[3:0] 位、SCKCR3.CKSEL[2:0] 位、PLLCR.STC[4:0] 位和 PLLCR.PLIDIV[1:0] 位设定 PCLKB 和 PCLKD 的频率。

9.7.3 FlashIF 时钟

FlashIF 时钟（FCLK）是 FlashIF 的运行时钟。

通过 SCKCR.FCK[3:0] 位、SCKCR3.CKSEL[2:0] 位、PLLCR.STC[4:0] 位和 PLLCR.PLIDIV[1:0] 位设定 FCLK 的频率。

9.7.4 外部总线时钟

外部总线时钟（BCLK）是外部总线控制器的运行时钟。能将 BCLK 从 BCLK 引脚输出到外部，用于外部总线。当外部总线有效时，不能将和 BCLK 引脚兼用的 P53 用作 I/O 端口。

如果将 SCKCR.PSTOP1 位置“0”并且将系统控制寄存器 0 的外部总线允许位（SYSCR0.EXBE）置“1”，就能从 BCLK 引脚输出 BCLK。必须在 SCKCR.PSTOP1 位为“1”的状态下将 SYSCR0.EXBE 位更改为“1”。

如果将 BCKCR.BCLKDIV 位置“1”，就能从 BCLK 引脚输出 BCLK 的 2 分频时钟。

能通过 SCKCR.BCK[3:0] 位、SCKCR3.CKSEL[2:0] 位、PLLCR.STC[4:0] 位和 PLLCR.PLIDIV[1:0] 位设定 BCLK 的频率。

9.7.5 CAC 时钟

CAC 时钟（CACCLK）是 CAC 模块的运行时钟。

CACCLK 有主时钟振荡器生成的 CACMCLK、副时钟振荡器生成的 CACSCLK、高速内部振荡器生成的 CACHCLK、低速内部振荡器生成的 CACLCLK 以及 IWDT 专用内部振荡器生成的 CACILCLK。

9.7.6 RTC 专用时钟

RTC 专用时钟（RTCSCLK）是 RTC 模块的运行时钟。

RTCSCLK 是副时钟振荡器生成的时钟。

9.7.7 IWDT 专用低速时钟

IWDT 专用低速时钟（IWDTCLK）是 IWDT 模块的运行时钟。

IWDTCLK 是 IWDT 专用内部振荡器通过内部振荡生成的时钟。

9.8 使用时的注意事项

9.8.1 有关时钟发生电路的注意事项

1. 通过 SCKCR 寄存器选择提供给各模块的系统时钟（ICLK）、外围模块时钟（PCLKB、PCLKD）、FlashIF 时钟（FCLK）和外部总线时钟（BCLK）的频率。
各频率的设定注意事项如下所示：
各频率必须设定在电特性的 AC 时序的时钟周期时间 t_{cy} 的运行保证范围内。
必须将频率设定在表 9.1 的频率范围内。
外围模块基本上以 PCLKB 和 PCLKD 为基准运行。因此必须注意：在更改频率的前后定时器和 SCI 等的运行速度会发生变化。
2. 在系统时钟（ICLK）和外部总线时钟（BCLK）之间需要满足以下的频率关系：
 $ICLK \geq BCLK$
在不使用 BCLK 时，必须将 ICLK 和 PCLKB 中设定频率大的一方设定到 SCKCR.BCK 位。但是，如果该设定使 BCLK 大于 25MHz，就必须将小于 25MHz 的最大分频比设定大 SCKCR.BCK 位。
3. 不能在存取外部总线的过程中更改时钟频率。如果在更改时钟频率后开始存取外部总线，就必须在确认频率更改结束后开始存取外部总线。
4. 为了在更改时钟频率后确切地执行下一个处理，必须在频率更改的写操作后读同一个寄存器，然后执行下一个处理。
5. 在解除复位后，如果在低速内部振荡器运行的过程中更改系统时钟的时钟源，就必须在写 SCKCR3 寄存器前，给地址 0008 0200h 写“00h”。

[写步骤]

- ①允许写 PRCR.PRC2 位的对象寄存器
给保护寄存器（PRCR）写“A504h”
- ②给地址 0008 0200h 写“00h”
- ③禁止写 PRCR.PRC2 位的对象寄存器
给保护寄存器（PRCR）写“A500h”

9.8.2 有关谐振器的注意事项

因为谐振器的各种特性与用户的电路板设计密接相关，所以请用户参照本章介绍的谐振器的连接例子，在给予充分的评估后使用。谐振器的电路常数因谐振器和安装电路的寄生电容而不同，因此必须和谐振器厂商仔细商谈后决定。外加在振荡引脚的电压不能超过最大额定值。

9.8.3 设计电路板时的注意事项

在使用谐振器时，必须尽量将谐振器和电容器排列在谐振器连接引脚的附近。如图 9.9 所示，在振荡电路附近不能通过信号线，否则就可能因电感而不能正常振荡。

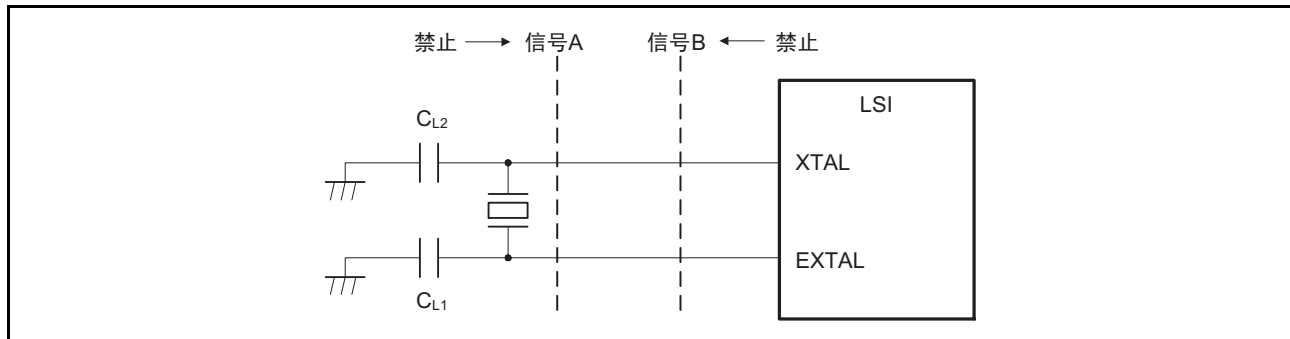


图 9.9 有关振荡电路部的电路板设计注意事项（主时钟振荡器的情况，副时钟振荡器的情况也一样）

9.8.4 有关谐振器连接引脚的注意事项

在不使用主时钟的情况下，能将 EXTAL 引脚和 XTAL 引脚用作通用端口 P36、P37。要用作通用端口时，必须将主时钟设定为停止（将 MOSCCR.MOSTP 位置“1”）的情况下使用。但是，在使用主时钟的系统中，不能将 EXTAL 引脚和 XTAL 引脚用作通用端口。在使用主时钟时，不能将 P36 和 P37 设定为输出。

9.8.5 有关副时钟的注意事项

副时钟既有可能用作系统时钟或者实时时钟的计数源，也有可能同时用作系统时钟和实时时钟的计数源。因此，包括不使用副时钟的情况在内，在设定时有以下的注意事项和限制事项。

- 通过副时钟振荡器控制寄存器的副时钟振荡器停止位（SOSCCR.SOSTP）和 RTC 控制寄存器 3 的副时钟振荡器控制位（RCR3.RTCEN），控制副时钟的运行或者停止。如果将这两位中的任意一位设定为运行，副时钟就为运行状态。
- 如果将副时钟用作系统时钟，就必须通过 SOSCCR.SOSTP 位进行设定；如果将副时钟用作实时时钟的计数源，就必须通过 RCR3.RTCEN 位进行设定。
- 与 RCR3.RTCEN 位的状态无关，如果将 SOSCCR.SOSTP 位更改为“0”（振荡），就必须在等待振荡稳定时间后使用。
- 如果在改写 SOSCCR.SOSTP 位后改写 RCR3.RTCEN 位或者在改写 RCR3.RTCEN 位后改写 SOSCCR.SOSTP 位，就必须确认先改写的位是否已被改写，然后改写后面的位。
- 因为冷启动后副时钟控制电路的状态不稳定，因此，与是否使用副时钟无关，必须进行初始化。通过将 SOSCCR.SOSTP 和 RCR3.RTCEN 位设定为停止振荡来进行初始化。有关 RCR3.RTCEN 位的初始化，请参照“25. 实时时钟（RTCb）”。

10. 时钟频率精度测量电路 (CAC)

时钟频率精度测量电路 (CAC) 根据从 LSI 外部输入的基准信号和其他时钟源来监视时钟频率，并且在超出设定范围时发生中断。

10.1 概要

CAC 的规格和框图分别如表 10.1 和图 10.1 所示。

表 10.1 CAC 的规格

项目	内容
时钟频率测量	能测量以下时钟的频率 • 主时钟振荡器的输出时钟 (主时钟) • 副时钟振荡器的输出时钟 (副时钟) • 高速内部振荡器的输出时钟 (HOCO 时钟) • 低速内部振荡器的输出时钟 (LOCO 时钟) • IWDT 专用内部振荡器的输出时钟 (IWDTCLK 时钟)
选择功能	数字滤波器功能
中断源	• 测量结束中断 • 频率错误中断 • 上溢
低功耗功能	能设定为模块停止状态

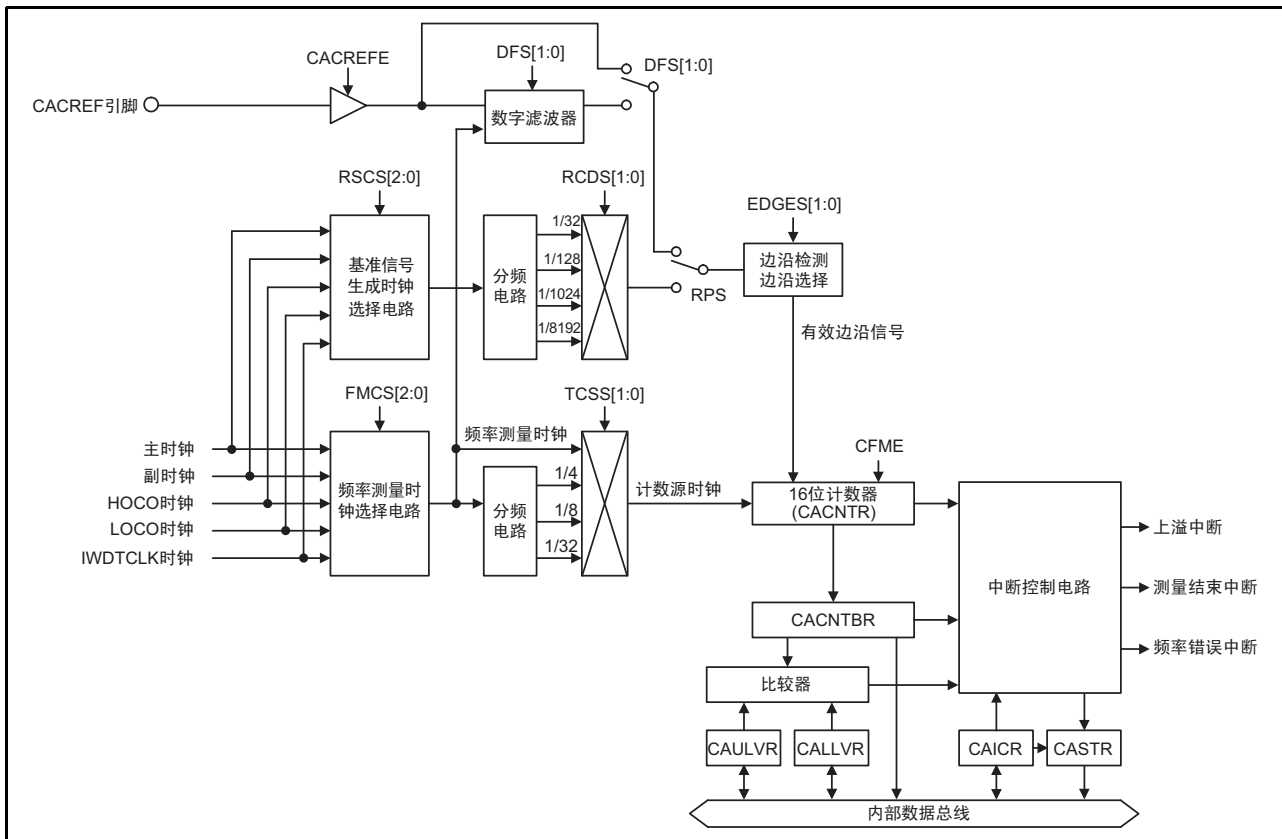


图 10.1 CAC 的框图

CAC 的输入 / 输出引脚如表 10.2 所示。

表 10.2 CAC 的输入 / 输出引脚

引脚名	输入 / 输出	功能
CACREF	输入	时钟频率精度测量电路的输入引脚

10.2 寄存器说明

10.2.1 CAC 控制寄存器 0 (CACR0)

地址 0008 B000h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	CFME
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	CFME	时钟频率测量允许位	0: 时钟频率测量无效 1: 时钟频率测量有效	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

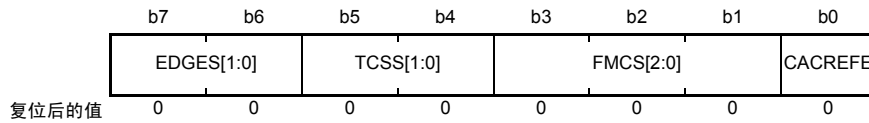
CFME 位 (时钟频率测量允许位)

当此位为“1”时，时钟频率测量有效。

在改写此位后，其结果反映到内部电路需要一些时间。如果在前一个值还未反映到内部电路的状态下改写此位，就忽视此改写。能通过读此位确认改写是否已被反映。

10.2.2 CAC 控制寄存器 1 (CACR1)

地址 0008 B001h



位	符号	位名	功能	R/W																											
b0	CACREFE	CACREF 引脚输入允许位	0: CACREF 引脚输入无效 1: CACREF 引脚输入有效	R/W																											
b3-b1	FMCS[2:0]	频率测量时钟选择位	<table style="width: 100%; border: none;"> <tr> <td style="width: 10%; text-align: right;">b3</td> <td style="width: 10%; text-align: left;">b1</td> <td></td> </tr> <tr> <td>0 0</td> <td>0</td> <td>主时钟振荡器的输出时钟</td> </tr> <tr> <td>0 0</td> <td>1</td> <td>副时钟振荡器的输出时钟</td> </tr> <tr> <td>0 1</td> <td>0</td> <td>高速内部振荡器的输出时钟</td> </tr> <tr> <td>0 1</td> <td>1</td> <td>低速内部振荡器的输出时钟</td> </tr> <tr> <td>1 0</td> <td>0</td> <td>IWDT 专用内部振荡器的输出时钟</td> </tr> <tr> <td>1 0</td> <td>1</td> <td>不能设定</td> </tr> <tr> <td>1 1</td> <td>0</td> <td>不能设定</td> </tr> <tr> <td>1 1</td> <td>1</td> <td>不能设定</td> </tr> </table>	b3	b1		0 0	0	主时钟振荡器的输出时钟	0 0	1	副时钟振荡器的输出时钟	0 1	0	高速内部振荡器的输出时钟	0 1	1	低速内部振荡器的输出时钟	1 0	0	IWDT 专用内部振荡器的输出时钟	1 0	1	不能设定	1 1	0	不能设定	1 1	1	不能设定	R/W
b3	b1																														
0 0	0	主时钟振荡器的输出时钟																													
0 0	1	副时钟振荡器的输出时钟																													
0 1	0	高速内部振荡器的输出时钟																													
0 1	1	低速内部振荡器的输出时钟																													
1 0	0	IWDT 专用内部振荡器的输出时钟																													
1 0	1	不能设定																													
1 1	0	不能设定																													
1 1	1	不能设定																													
b5-b4	TCSS[1:0]	定时器计数时钟源选择位	<table style="width: 100%; border: none;"> <tr> <td style="width: 10%; text-align: right;">b5</td> <td style="width: 10%; text-align: left;">b4</td> <td></td> </tr> <tr> <td>0 0</td> <td></td> <td>无分频时钟</td> </tr> <tr> <td>0 1</td> <td></td> <td>4 分频时钟</td> </tr> <tr> <td>1 0</td> <td></td> <td>8 分频时钟</td> </tr> <tr> <td>1 1</td> <td></td> <td>32 分频时钟</td> </tr> </table>	b5	b4		0 0		无分频时钟	0 1		4 分频时钟	1 0		8 分频时钟	1 1		32 分频时钟	R/W												
b5	b4																														
0 0		无分频时钟																													
0 1		4 分频时钟																													
1 0		8 分频时钟																													
1 1		32 分频时钟																													
b7-b6	EDGES[1:0]	有效边沿选择位	<table style="width: 100%; border: none;"> <tr> <td style="width: 10%; text-align: right;">b7</td> <td style="width: 10%; text-align: left;">b6</td> <td></td> </tr> <tr> <td>0 0</td> <td></td> <td>上升沿</td> </tr> <tr> <td>0 1</td> <td></td> <td>下降沿</td> </tr> <tr> <td>1 0</td> <td></td> <td>双边沿 (上升沿 / 下降沿)</td> </tr> <tr> <td>1 1</td> <td></td> <td>不能设定</td> </tr> </table>	b7	b6		0 0		上升沿	0 1		下降沿	1 0		双边沿 (上升沿 / 下降沿)	1 1		不能设定	R/W												
b7	b6																														
0 0		上升沿																													
0 1		下降沿																													
1 0		双边沿 (上升沿 / 下降沿)																													
1 1		不能设定																													

CACREFE 位 (CACREF 引脚输入允许位)

当此位为“1”时，CACREF 引脚的输入有效。

FMCS[2:0] 位 (频率测量时钟选择位)

能通过设定此位选择要测量频率的时钟。

TCSS[1:0] 位 (定时器计数时钟源选择位)

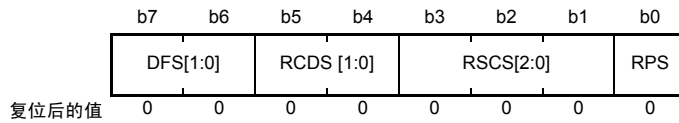
能通过设定此位选择时钟频率精度测量电路的计数时钟源。

EDGES[1:0] 位 (有效边沿选择位)

能通过设定此位选择基准信号的有效边沿。

10.2.3 CAC 控制寄存器 2 (CACR2)

地址 0008 B002h



位	符号	位名	功能	R/W																											
b0	RPS	基准信号选择位	0: CACREF 引脚输入 1: 内部生成信号	R/W																											
b3-b1	RSCS[2:0]	基准信号生成时钟选择位	<table style="width: 100%; border: none;"> <tr> <td style="width: 10%; text-align: right;">b3</td> <td style="width: 10%; text-align: left;">b1</td> <td></td> </tr> <tr> <td>0 0 0:</td> <td></td> <td>主时钟振荡器的输出时钟</td> </tr> <tr> <td>0 0 1:</td> <td></td> <td>副时钟振荡器的输出时钟</td> </tr> <tr> <td>0 1 0:</td> <td></td> <td>高速内部振荡器的输出时钟</td> </tr> <tr> <td>0 1 1:</td> <td></td> <td>低速内部振荡器的输出时钟</td> </tr> <tr> <td>1 0 0:</td> <td></td> <td>IWDT 专用内部振荡器的输出时钟</td> </tr> <tr> <td>1 0 1:</td> <td></td> <td>不能设定</td> </tr> <tr> <td>1 1 0:</td> <td></td> <td>不能设定</td> </tr> <tr> <td>1 1 1:</td> <td></td> <td>不能设定</td> </tr> </table>	b3	b1		0 0 0:		主时钟振荡器的输出时钟	0 0 1:		副时钟振荡器的输出时钟	0 1 0:		高速内部振荡器的输出时钟	0 1 1:		低速内部振荡器的输出时钟	1 0 0:		IWDT 专用内部振荡器的输出时钟	1 0 1:		不能设定	1 1 0:		不能设定	1 1 1:		不能设定	R/W
b3	b1																														
0 0 0:		主时钟振荡器的输出时钟																													
0 0 1:		副时钟振荡器的输出时钟																													
0 1 0:		高速内部振荡器的输出时钟																													
0 1 1:		低速内部振荡器的输出时钟																													
1 0 0:		IWDT 专用内部振荡器的输出时钟																													
1 0 1:		不能设定																													
1 1 0:		不能设定																													
1 1 1:		不能设定																													
b5-b4	RCDS[1:0]	基准信号生成时钟的分频比选择位	<table style="width: 100%; border: none;"> <tr> <td style="width: 10%; text-align: right;">b5</td> <td style="width: 10%; text-align: left;">b4</td> <td></td> </tr> <tr> <td>0 0:</td> <td></td> <td>32 分频时钟</td> </tr> <tr> <td>0 1:</td> <td></td> <td>128 分频时钟</td> </tr> <tr> <td>1 0:</td> <td></td> <td>1024 分频时钟</td> </tr> <tr> <td>1 1:</td> <td></td> <td>8192 分频时钟</td> </tr> </table>	b5	b4		0 0:		32 分频时钟	0 1:		128 分频时钟	1 0:		1024 分频时钟	1 1:		8192 分频时钟	R/W												
b5	b4																														
0 0:		32 分频时钟																													
0 1:		128 分频时钟																													
1 0:		1024 分频时钟																													
1 1:		8192 分频时钟																													
b7-b6	DFS[1:0]	数字滤波器功能选择位	<table style="width: 100%; border: none;"> <tr> <td style="width: 10%; text-align: right;">b7</td> <td style="width: 10%; text-align: left;">b6</td> <td></td> </tr> <tr> <td>0 0:</td> <td></td> <td>数字滤波器功能无效</td> </tr> <tr> <td>0 1:</td> <td></td> <td>频率测量时钟</td> </tr> <tr> <td>1 0:</td> <td></td> <td>频率测量时钟的 4 分频时钟</td> </tr> <tr> <td>1 1:</td> <td></td> <td>频率测量时钟的 16 分频时钟</td> </tr> </table>	b7	b6		0 0:		数字滤波器功能无效	0 1:		频率测量时钟	1 0:		频率测量时钟的 4 分频时钟	1 1:		频率测量时钟的 16 分频时钟	R/W												
b7	b6																														
0 0:		数字滤波器功能无效																													
0 1:		频率测量时钟																													
1 0:		频率测量时钟的 4 分频时钟																													
1 1:		频率测量时钟的 16 分频时钟																													

RPS 位 (基准信号选择位)

能通过设定此位选择是将 CACREF 引脚输入用作基准信号还是将内部生成信号用作基准信号。

RSCS[2:0] 位 (基准信号生成时钟选择位)

能通过设定此位选择生成基准信号的时钟源。

RCDS[1:0] 位 (基准信号生成时钟的分频比选择位)

能通过设定此位选择基准信号生成时钟的分频比。

DFS[1:0] 位 (数字滤波器功能选择位)

能通过设定此位选择数字滤波器的有效或者无效以及采样时钟。

10.2.4 CAC 中断控制寄存器 (CAICR)

地址 0008 B003h

b7	b6	b5	b4	b3	b2	b1	b0
—	OVFFCL	MENDFCL	FERRFCL	—	OVFIE	MENDIE	FERRIE
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	FERRIE	频率错误中断允许位	0: 频率错误中断无效 1: 频率错误中断有效	R/W
b1	MENDIE	测量结束中断允许位	0: 测量结束中断无效 1: 测量结束中断有效	R/W
b2	OVFIE	上溢中断允许位	0: 上溢中断无效 1: 上溢中断有效	R/W
b3	—	保留位	读写值都为“0”。	R/W
b4	FERRFCL	FERRF 标志清除位	如果将此位置“1”，就清除 FERRF 标志。 读取值总是为“0”。	R/W
b5	MENDFCL	MENDF 标志清除位	如果将此位置“1”，就清除 MENDF 标志。 读取值总是为“0”。	R/W
b6	OVFFCL	OVFF 标志清除位	如果将此位置“1”，就清除 OVFF 标志。 读取值总是为“0”。	R/W
b7	—	保留位	读写值都为“0”。	R/W

FERRIE 位 (频率错误中断允许位)

当此位为“1”时，允许频率错误中断。

MENDIE 位 (测量结束中断允许位)

当此位为“1”时，允许测量结束中断。

OVFIE 位 (上溢中断允许位)

当此位为“1”时，允许上溢中断。

FERRFCL 位 (FERRF 标志清除位)

如果将此位置“1”，就清除 FERRF 标志。

MENDFCL 位 (MENDF 标志清除位)

如果将此位置“1”，就清除 MENDF 标志。

OVFFCL 位 (OVFF 标志清除位)

如果将此位置“1”，就清除 OVFF 标志。

10.2.5 CAC 状态寄存器 (CASTR)

地址 0008 B004h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	OVFF	MENDF	FERRF
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	FERRF	频率错误标志	[为“1”的条件] • 当时钟频率不为设定值时 [为“0”的条件] • 给 FERRFCL 写“1”时	R
b1	MENDF	测量结束标志	[为“1”的条件] • 当测量结束时 [为“0”的条件] • 给 MENDFCL 写“1”时	R
b2	OVFF	上溢标志	[为“1”的条件] • 当计数器发生上溢时 [为“0”的条件] • 给 OVFFCL 写“1”时	R
b7-b3	—	保留位	读写值都为“0”。	R/W

FERRF 标志 (频率错误标志)

当时钟频率不为设定值时，此标志变为“1”。

MENDF 标志 (测量结束标志)

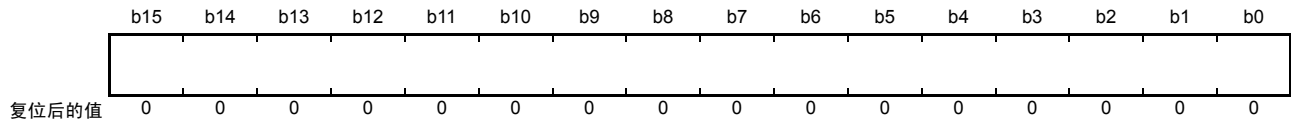
当测量结束时，此标志变为“1”。

OVFF 标志 (上溢标志)

当计数器发生上溢时，此标志变为“1”。

10.2.6 CAC 上限值设定寄存器 (CAULVR)

地址 0008 B006h



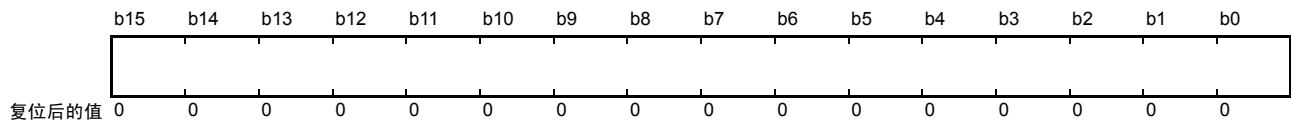
CAULVR 寄存器是保存频率上限值的 16 位可读写寄存器。

在 CACR0.CFME 位为“0”时设定此次寄存器。

由于数字滤波器、边沿检测电路、以及 CACREF 引脚输入的相位差，可能会使保存到 CACNTBR 寄存器的计数器值发生偏差，所以必须设定有宽裕的值。

10.2.7 CAC 下限值设定寄存器 (CALLVR)

地址 0008 B008h



CALLVR 寄存器是保存频率下限值的 16 位可读写寄存器。

在 CACR0.CFME 位为“0”时设定此次寄存器。

由于数字滤波器、边沿检测电路、以及 CACREF 引脚输入的相位差，可能会使保存到 CACNTBR 寄存器的计数器值发生偏差，所以必须设定有宽裕的值。

10.2.8 CAC 计数器的缓冲寄存器 (CACNTBR)

地址 0008 B00Ah



CACNTBR 寄存器是 16 位只读专用寄存器，保存输入基准信号有效边沿时的计数器值。

10.3 运行说明

10.3.1 以 CACREF 引脚输入为基准测量时钟频率

以 CACREF 引脚输入为基准的时钟频率精度测量电路的运行例子如图 10.2 所示。
在测量时钟频率时，时钟频率精度测量电路如下运行。

1. 在将 CACR2.RPS 位置“0”、CACR1.CACREFE 位置“1”的状态下给 CACR0.CFME 位写“1”时，以 CACREF 引脚输入为基准的时钟频率测量有效。
2. 在给 CFME 位写“1”后，如果从 CACREFE 引脚输入 CACR1.EDGES[1:0] 位选择的有效边沿，就开始定时器的递增计数。
3. 在输入下一个有效边沿时，将寄存器值保存到 CACNTBR 寄存器，并且与 CAULVR 寄存器和 CALLVR 寄存器进行比较。当 CACNTBR 寄存器 \leq CAULVR 寄存器并且 CACNTBR 寄存器 \geq CALLVR 寄存器时，时钟频率为正常值，因此只有 CASTR.MENDF 标志被置“1”。另外，如果给 CAICR.MENDIE 位设定“1”，就发生测量结束中断。
4. 在输入下一个有效边沿时，将寄存器值保存到 CACNTBR 寄存器，并且与 CAULVR 寄存器和 CALLVR 寄存器进行比较。当 CACNTBR 寄存器 $>$ CAULVR 寄存器时，时钟频率就为异常值，因此 CASTR.FERRF 标志被置“1”。另外，如果给 CAICR.FERRIE 位设定“1”，就发生频率错误中断，而且 CASTR.MENDF 标志也被置“1”。如果给 CAICR.MENDIE 位设定“1”，就发生测量结束中断。
5. 在输入下一个有效边沿时，将寄存器值保存到 CACNTBR 寄存器，并且与 CAULVR 寄存器和 CALLVR 寄存器进行比较。当 CACNTBR 寄存器 $<$ CALLVR 寄存器时，时钟频率为异常值，因此 CASTR.FERRF 标志被置“1”。另外，如果给 CAICR.FERRIE 位设定“1”，就发生频率错误中断，而且 CASTR.MENDF 标志也被置“1”。如果给 CAICR.MENDIE 位设定“1”，就发生测量结束中断。
6. 在 CACR0.CFME 位为“1”的期间，每当输入有效边沿时，就将计数器值保存到 CACNTBR 寄存器，并且与 CAULVR 寄存器和 CALLVR 寄存器进行比较。如果给 CACR0.CFME 位写“0”，就清除计数器并且停止递增计数。

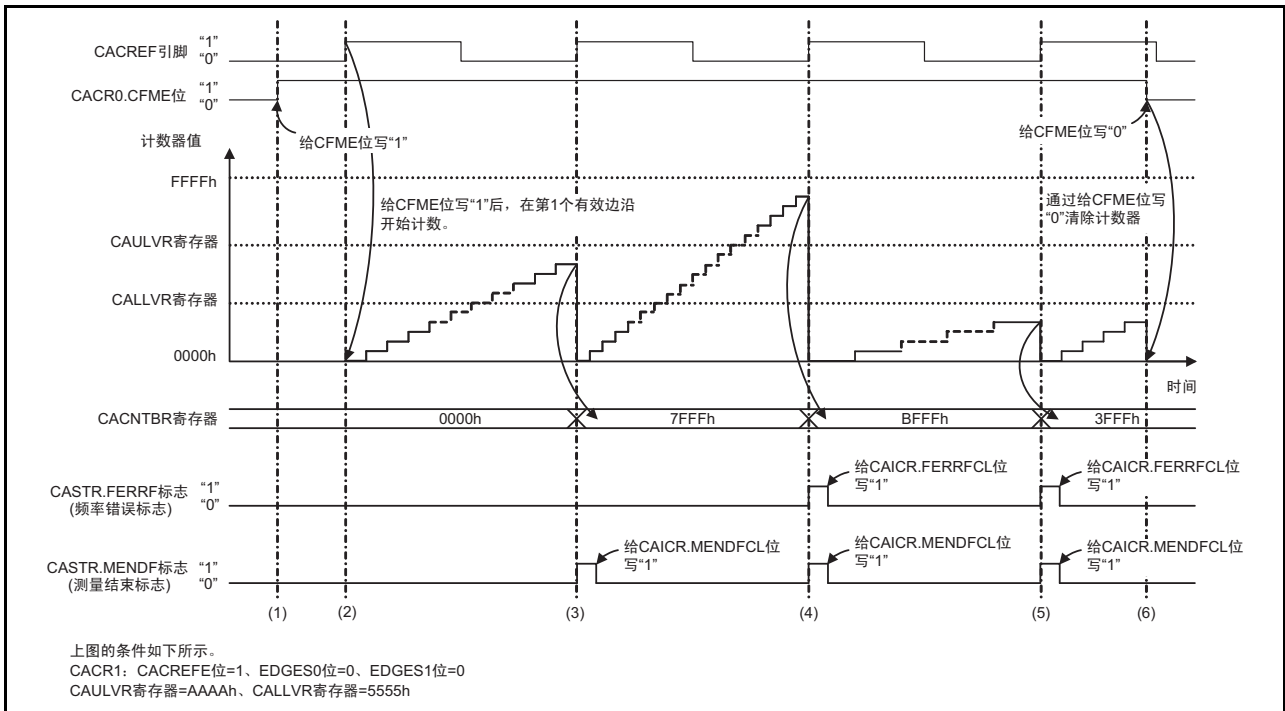


图 10.2 以 CACREF 引脚为基准的时钟频率精度测量电路的运行例子

10.3.2 以其他时钟源为基准测量时钟频率

以其他时钟源为基准的时钟频率精度测量电路的运行例子如图 10.3 所示。
在测量时钟频率时，时钟频率精度测量电路如下运行。

1. 在将 CACR2.RPS 位置“1”状态下给 CACR0.CFME 位写“1”时，以其他时钟源为基准的时钟频率测量有效。
2. 在给 CFME 位写“1”后，如果根据 CACR2.RSCS[2:0] 位选择的时钟源输入 CACR1.EDGES[1:0] 位选择的有效边沿，就开始定时器的递增计数。
3. 在输入下一个有效边沿时，将寄存器值保存到 CACNTBR 寄存器，并且与 CAULVR 寄存器和 CALLVR 寄存器进行比较。当 CACNTBR 寄存器 ≤ CAULVR 寄存器并且 CACNTBR 寄存器 ≥ CALLVR 寄存器时，时钟频率为正常值，因此只有 CASTR.MENDF 标志被置“1”。另外，如果给 CAICR.MENDIE 位设定“1”，就发生测量结束中断。
4. 在输入下一个有效边沿时，将寄存器值保存到 CACNTBR 寄存器，并且与 CAULVR 寄存器和 CALLVR 寄存器进行比较。当 CACNTBR 寄存器 > CAULVR 寄存器时，时钟频率为异常值，因此 CASTR.FERRF 标志被置“1”。另外，如果给 CAICR.FERRIE 位设定“1”，就发生频率错误中断，而且 CASTR.MENDF 标志也被置“1”。如果给 CAICR.MENDIE 位设定“1”，就发生测量结束中断。
5. 在输入下一个有效边沿时，将寄存器值保存到 CACNTBR 寄存器，并且与 CAULVR 寄存器和 CALLVR 寄存器进行比较。当 CACNTBR 寄存器 < CALLVR 寄存器时，时钟频率为异常值，因此 CASTR.FERRF 标志被置“1”。另外，如果给 CAICR.FERRIE 位设定“1”，就发生频率错误中断，而且 CASTR.MENDF 标志也被置“1”。如果给 CAICR.MENDIE 位设定“1”，就发生测量结束中断。
6. 在 CACR0.CFME 位为“1”的期间，每当输入有效边沿时，就将计数器值保存到 CACNTBR 寄存器，并且与 CAULVR 寄存器和 CALLVR 寄存器进行比较。如果给 CACR0.CFME 位写“0”，就清除计数器并且停止递增计数。

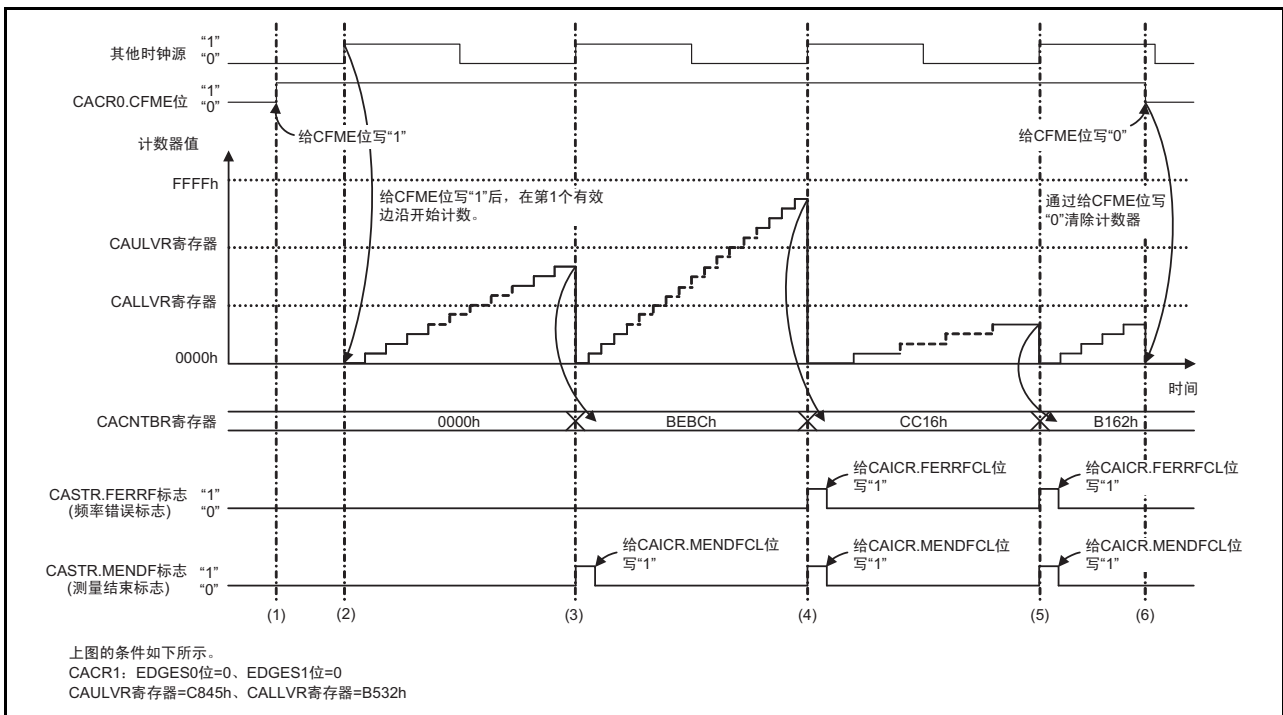


图 10.3 以其他时钟源为基准的时钟频率精度测量电路的运行例子

10.3.3 CACREF 引脚的数字滤波器功能

CACREF 引脚具有数字滤波器功能。数字滤波器功能在跟据设定的采样周期进行 3 次采样的引脚的电平相同时，将此相同的电平传达到内部，直到重新进行 3 次采样的引脚的电平相同为止。

数字滤波器功能可以设定数字滤波器功能的有效或者无效以及采样时钟。

由于数字滤波器和 CACREF 引脚输入信号的相位差，保存到 CACNTBR 寄存器的计数值有可能产生最大 1 个采样时钟周期的误差。

在选择分频时钟作为计数源时钟的情况下，能通过以下计算式表达计数值误差。

计数值误差 = (1 个计数源时钟周期) / (1 个采样时钟周期)

10.4 中断请求

时钟频率精度测量电路生成的中断请求有 3 种，分别是频率错误中断、测量结束中断以及上溢中断。如果产生各中断源，各状态标志就被置“1”。时钟频率精度测量电路的中断请求如表 10.3 所示。

表 10.3 时钟频率精度测量电路的中断请求

中断请求	中断允许位	状态标志	中断源
频率错误中断	CAICR.FERRIE	CASTR.FERRF	将 CACNTBR 寄存器与 CAULVR 寄存器和 CALLVR 寄存器比较，当比较结果为 CACNTBR 寄存器 > CAULVR 寄存器或者 CACNTBR 寄存器 < CALLVR 寄存器时
测量结束中断	CAICR.MENDIE	CASTR.MENDF	当输入基准信号的有效边沿时 但是，如果给 CACR0.CFME 位写“1”，就不能在第 1 个有效边沿发生测量结束中断。
上溢中断	CAICR.OVFIE	CASTR.OVFF	当计数器发生上溢时

10.5 使用时的注意事项

10.5.1 模块停止功能的设定

能通过模块停止控制寄存器 C (MSTPCRC) 设定禁止或者允许时钟频率精度测量电路的运行。初始值为停止时钟频率精度测量电路的运行。能通过解除模块停止状态存取寄存器。详细内容请参照“11. 低功耗功能”。

11. 低功耗功能

11.1 概要

作为低功耗功能，RX210 群有通过转换时钟来降低功耗的功能、BCLK 输出控制功能、模块停止功能、正常运行时的低功耗功能以及向低功耗状态转移的功能。

低功耗功能的规格如表 11.1 所示，向低功耗状态转移的条件、CPU 和外围模块等的状态以及各模式的解除方法如表 11.2 所示。

复位后，在正常的程序运行中，除 DMAC、DTC 和内部 RAM 以外的模块处于停止状态。

表 11.1 低功耗功能的规格

项目	内容
通过转换时钟来降低功耗的功能	能给系统时钟（ICLK）、外围模块时钟（PCLKB）、S12AD0 专用时钟（PCLKD）、外部总线时钟（BCLK）和闪存接口时钟（FCLK）分别设定分频比（注 1）。
BCLK 输出控制功能	可选择 BCLK 输出或者 High 电平输出（注 1）。
模块停止功能	能停止各外围模块的功能。
向低功耗状态转移的功能	<ul style="list-style-type: none"> 能进入使 CPU、外围模块和振荡器停止运行的低功耗状态。
低功耗状态	<ul style="list-style-type: none"> 睡眠模式 全模块时钟停止模式 软件待机模式 深度软件待机模式
低运行功率功能	<ul style="list-style-type: none"> 根据工作频率和工作电压范围选择运行功率控制模式，能降低正常运行时以及睡眠模式和全模块时钟停止模式中的功耗。 运行功率控制状态：5 种 <ul style="list-style-type: none"> 高速运行模式 中速运行模式 A 中速运行模式 B 低速运行模式 1 低速运行模式 2

注 1. 详细内容请参照“9. 时钟发生电路”。

表 11.2 各模式的转移方法、解除方法和运行状态

转移方法、解除方法和运行状态	睡眠模式	全模块时钟停止模式	软件待机模式	深度软件待机模式
转移方法	控制寄存器 + 指令	控制寄存器 + 指令	控制寄存器 + 指令	控制寄存器 + 指令
复位以外的解除方法	中断	中断 (注1)	中断 (注2)	中断 (注3)
解除后的状态 (注4)	程序的执行状态 (中断处理)	程序的执行状态 (中断处理)	程序的执行状态 (中断处理)	程序的执行状态 (复位处理)
主时钟振荡器	能运行	能运行	能运行	能运行
副时钟振荡器	能运行	能运行	能运行 (注5)	能运行 (注5)
高速内部振荡器	能运行	能运行	停止	停止
低速内部振荡器	能运行	能运行	停止	停止
IWDT 专用内部振荡器	能运行 (注6)	能运行 (注6)	能运行 (注6)	停止 (不定值) (注6)
PLL	能运行	能运行	停止	停止
CPU	停止 (保持)	停止 (保持)	停止 (保持)	停止 (不定值)
内部 RAM0 (0000 0000h ~ 0000 FFFFh)	能运行 (保持)	停止 (保持)	停止 (保持)	停止 (不定值)
闪存	运行	停止 (保持)	停止 (保持)	停止 (保持)
看门狗定时器 (WDT)	停止 (保持)	停止 (保持)	停止 (保持)	停止 (不定值)
独立看门狗定时器 (IWDT)	能运行 (注6)	能运行 (注6)	能运行 (注6)	停止 (不定值) (注6)
实时时钟 (RTC)	能运行	能运行	能运行	能运行
8 位定时器 (单元 0 和单元 1) (TMR)	能运行	能运行 (注7)	停止 (保持)	停止 (不定值)
电压检测电路 (LVD)	能运行	能运行	能运行 (注8)	能运行 (注9)
上电复位电路	运行	运行	运行 (注8)	运行 (注8)
外围模块	能运行	停止 (保持)	停止 (保持)	停止 (不定值)
I/O 端口	运行	保持 (注10)	保持 (注11)	保持 (注11)

“能运行”表示能通过设定控制寄存器来控制运行或者停止。

“停止 (保持)”表示保持内部寄存器的值并且内部状态为运行中止状态。

“停止 (不定值)”表示内部寄存器的值为不定值并且内部状态为断电状态。

注 1. 这是外部引脚中断 (NMI、IRQ0 ~ IRQ7) 和外围功能中断 (8 位定时器、RTC 闹钟、RTC 周期、IWDT、电压监视 1、电压监视 2、振荡停止检测)。

注 2. 这是外部引脚中断 (NMI、IRQ0~IRQ7) 和外围功能中断 (RTC 闹钟、RTC 周期、IWDT、电压监视 1、电压监视 2)。

注 3. 这是作为外部引脚中断发生源的部分引脚 (NMI、IRQ0-DS ~ IRQ7-DS、SCL-DS、SDA-DS) 和外围功能中断 (RTC 闹钟、RTC 周期、电压监视 1、电压监视 2)。但是, 这些中断源只在深度待机中断允许寄存器 i (DPSIERi) (i=0、2) 的对应位为“1”时有效。引脚名中附件“-DS”的引脚能用作深度软件待机模式的解除引脚。

注 4. 通过 RES# 引脚复位、上电复位、电压监视复位、独立看门狗定时器复位进行的解除除外。当通过 RES# 引脚复位、上电复位、电压监视复位、独立看门狗定时器复位进行解除时, 转移到复位状态。

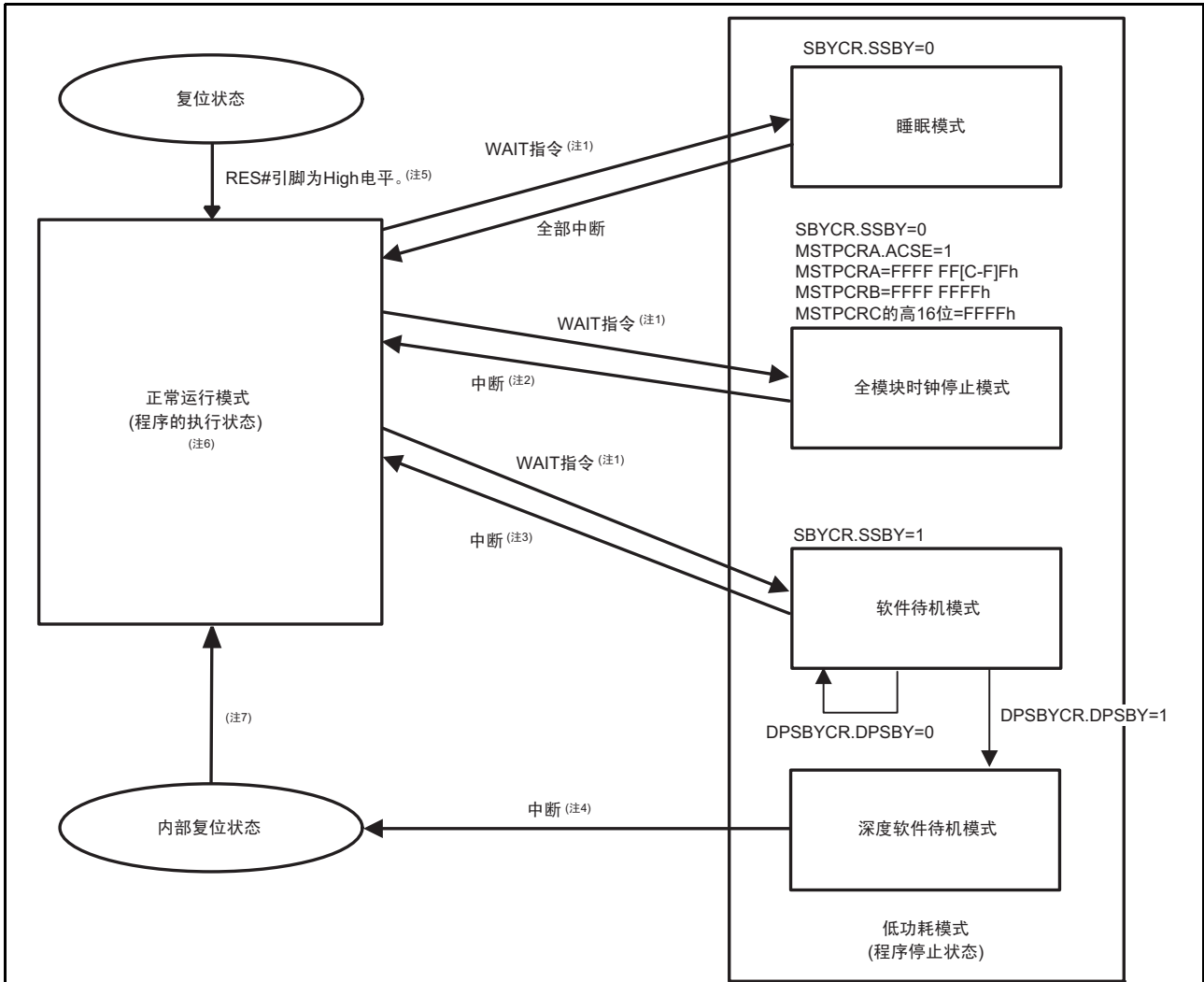
注 5. 能通过设定 RTC 控制寄存器 3 的副时钟振荡器控制位 (RCR3.RTCEN), 选择运行或者停止。

注 6. 在 IWDT 自动启动模式中, 能通过设定选项功能选择寄存器 0 的 IWDT 睡眠模式计数停止控制位 (OFS0.IWDTSLCSTP), 选择运行或者停止。当 OFS0.IWDTSLCSTP 位为“0” (在转移到低功耗模式时继续进行计数) 时, 不转移到深度软件待机模式而转移到软件待机模式。如果不在 IWDT 自动启动模式中, 就能通过设定 IWDT 计数停止控制寄存器的睡眠模式计数停止控制位 (IWDTCSSTPR.SLCSTP), 选择运行或者停止。当 IWDTCSSTPR.SLCSTP 位为“0” (在转移到低功耗模式时继续进行计数) 时, 不转移到深度软件待机模式而转移到软件待机模式。

注 7. 能通过设定模块停止控制寄存器 A 的 8 位定时器 1、0 (单元 0) 的模块停止设定位 (MSTPCRA.MSTPA5) 以及 8 位定时器 3、2 (单元 1) 的模块停止设定位 (MSTPCRA.MSTPA4), 选择运行或者停止。

注 8. 如果通过将闪存 HOCO 软件待机控制寄存器的软切断位 (FHSSBYCR.SOFTCUT2) 设定为“1”而转移到软件待机模式, 或者通过深度待机控制寄存器的深度切断位 (DPSBYCR.DEEPCUT1) 设定为“1”而转移到深度软件待机模式, 电压检测电路就停止运行, 并且上电复位电路为低功耗功能有效。

- 注 9. 当电压监视 1 电路控制寄存器 0 的电压监视 1 电路模式选择位 (LVD1CR0.LVD1RI) 为“1”或者电压监视 2 电路控制寄存器 0 的电压监视 2 电路模式选择位 (LVD2CR0.LVD2RI) 为“1”时, 不转移到深度软件待机模式而转移到软件待机模式。
- 注 10. 如果将 P53 用作 BCLK, 就继续进行 BCLK 输出。当 8 位定时器和 RTC 运行时, 相关引脚继续运行。
- 注 11. 能通过设定待机控制寄存器的输出端口允许位 (SBYCR.OPE), 选择地址总线、总线控制信号 (CS0# ~ CS3#、RD#、WR0#、WR1#、WR#、BC0#、BC1#、ALE) 为保持状态或者高阻抗状态。



- 注1. 在执行WAIT指令后, 如果在向程序停止状态转移过程中接受解除源的中断, 就不转移到程序停止状态而执行中断异常处理。
 - 注2. 这是外部引脚中断(NMI、IRQ0~IRQ7)、外围功能中断(8位定时器、RTC闹钟、RTC周期、IWDT、电压监视1、电压监视2、振荡停止检测)。但是, 8位定时器在模块停止控制寄存器A的8位定时器1、0(单元0)的模块停止设定位(MSTPCRA.MSTPA5)或者8位定时器3、2(单元1)的模块停止设定位(MSTPCRA.MSTPA4)为“0”时有效。
 - 注3. 这是外部引脚中断(NMI、IRQ0~IRQ7)和外围功能中断(RTC闹钟、RTC周期、IWDT、电压监视1、电压监视2)。
 - 注4. 这是作为外部引脚中断发生源的部分引脚(NMI、IRQ0-DS~IRQ7-DS、SCL-DS、SDA-DS)和外围功能中断(RTC闹钟、RTC周期、电压监视1、电压监视2)。但是, 这些中断源只在深度待机中断允许寄存器0、2(DPSIER0、DPSIER2)的对应位为“1”时有效。
 - 注5. 通过正常运行模式的LOCO时钟源从复位状态进行转移。
 - 注6. 通过中断从睡眠模式、全模块时钟停止模式和软件待机模式转移到正常运行模式。此时, 对于睡眠模式, 能选择返回后的时钟源。详细内容请参照“11.2.6 睡眠模式返回时钟源转换寄存器(RSTCKCR)”。对于软件待机模式, 返回时和返回前为相同的时钟源。
 - 注7. 如果发生注4所示的中断源, 就产生一定时间的内部复位(深度软件待机复位)。在解除内部复位的同时解除深度软件待机模式。如果解除深度软件待机模式, 就转移到正常运行模式, 并且将LOCO时钟作为时钟源开始进行复位异常处理。
- 在所有状态中, 如果发生引脚复位、上电复位、电压监视0复位~电压监视2复位、看门狗定时器复位、独立看门狗定时器复位和软件复位, 就转移到复位状态。

图 11.1 模式转移

11.2 寄存器说明

11.2.1 待机控制寄存器 (SBYCR)

地址 0008 000Ch

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SSBY	OPE	—	—	—	—	—	—	—	—	—	—	—	—	—	—
复位后的值	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b13-b0	—	保留位	读写值都为“0”。	R/W
b14	OPE	输出端口允许位	0: 在软件待机模式和深度软件待机模式中, 地址总线和总线控制信号为高阻抗。 1: 在软件待机模式和深度软件待机模式中, 地址总线和总线控制信号保持输出状态。	R/W
b15	SSBY	软件待机位	0: 在执行 WAIT 指令后, 转移到睡眠模式或者全模块时钟停止模式。 1: 在执行 WAIT 指令后, 转移到软件待机模式。	R/W

OPE 位 (输出端口允许位)

在软件待机模式和深度软件待机模式中, 此位选择是将地址总线和总线控制信号 (CS0# ~ CS3#、RD#、WR0#、WR1#、WR#、BC0#、BC1#、ALE) 保持为输出状态还是置为高阻抗。

SSBY 位 (软件待机位)

此位设定 WAIT 指令执行后的转移目标。

如果在 SSBY 位为“1”的状态下执行 WAIT 指令, 就转移到软件待机模式。

如果在通过中断解除软件待机模式后转移到正常模式, SSBY 位就保持“1”。要将 SSBY 位置“0”时, 必须写“0”。

当振荡停止检测控制寄存器的振荡停止检测功能允许位 (OSTDCR.OSTDE) 为“1”时, SSBY 位的设定值无效。当 SSBY 位为“1”时, 也在执行 WAIT 指令后转移到睡眠模式或者全模块时钟停止模式。

11.2.2 模块停止控制寄存器 A (MSTPCRA)

地址 0008 0010h

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
ACSE	—	MSTPA 29	MSTPA 28	MSTPA 27	—	—	MSTPA 24	—	—	—	—	MSTPA 19	—	MSTPA 17	—
复位后的值	0	1	0	0	0	1	1	0	1	1	1	1	1	1	1
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
MSTPA 15	MSTPA 14	—	—	—	—	MSTPA9	—	—	—	MSTPA5	MSTPA4	—	—	—	—
复位后的值	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

位	符号	位名	功能	R/W
b3-b0	—	保留位	读写值都为“1”。	R/W
b4	MSTPA4	8 位定时器 3、2（单元 1）的模块停止设定位	对象模块：TMR3 和 TMR2 0：解除模块停止状态 1：转移到模块停止状态	R/W
b5	MSTPA5	8 位定时器 1、0（单元 0）的模块停止设定位	对象模块：TMR1 和 TMR0 0：解除模块停止状态 1：转移到模块停止状态	R/W
b8-b6	—	保留位	读写值都为“1”。	R/W
b9	MSTPA9	多功能定时器脉冲单元 2A 的模块停止设定位	对象模块：MTU（MTU0 ~ MTU5） 0：解除模块停止状态 1：转移到模块停止状态	R/W
b13-b10	—	保留位	读写值都为“1”。	R/W
b14	MSTPA14	比较匹配定时器（单元 1）的模块停止设定位	对象模块：CMT 单元 1（CMT2 和 CMT3） 0：解除模块停止状态 1：转移到模块停止状态	R/W
b15	MSTPA15	比较匹配定时器（单元 0）的模块停止设定位	对象模块：CMT 单元 0（CMT0 和 CMT1） 0：解除模块停止状态 1：转移到模块停止状态	R/W
b16	—	保留位	读写值都为“1”。	R/W
b17	MSTPA17	12 位 A/D 转换器的模块停止设定位	对象模块：S12AD 0：解除模块停止状态 1：转移到模块停止状态	R/W
b18	—	保留位	读写值都为“1”。	R/W
b19	MSTPA19	D/A 转换器的模块停止设定位	对象模块：DA 0：解除模块停止状态 1：转移到模块停止状态	R/W
b23-b20	—	保留位	读写值都为“1”。	R/W
b24	MSTPA24	模块停止 A24 设定位	读操作和写操作都有效。如果要转移到全模块停止模式，就必须预先给此位写“1”。	R/W
b26-b25	—	保留位	读写值都为“1”。	R/W
b27	MSTPA27	模块停止 A27 设定位	读操作和写操作都有效。如果要转移到全模块停止模式，就必须预先给此位写“1”。	R/W

位	符号	位名	功能	R/W
b28	MSTPA28	DMA 控制器 / 数据传送控制器的 模块停止设定位	对象模块: DMAC/DTC 0: 解除模块停止状态 1: 转移到模块停止状态	R/W
b29	MSTPA29	模块停止 A29 设定位	读操作和写操作都有效。如果要转移到全模块停止 模式, 就必须预先给此位写“1”。	R/W
b30	—	保留位	读写值都为“1”。	R/W
b31	ACSE	全模块时钟停止模式允许位	0: 禁止全模块时钟停止模式 1: 允许全模块时钟停止模式	R/W

ACSE 位 (全模块时钟停止模式允许位)

通过 ACSE 位设定允许或者禁止转移到全模块时钟停止模式。在将 ACSE 位置“1”并且 SBYCR.SSBY 位、MSTPCRA、MSTPCRB、MSTPCRC 寄存器满足规定条件的状态下, 如果 CPU 执行 WAIT 指令, 就转移到全模块时钟停止模式。详细内容请参照“11.6.2 全模块时钟停止模式”。

能通过设定 MSTPA5 位和 MSTPA4 位, 选择 8 位定时器的运行或者停止。

当 SBYCR.SSBY 位和 MSTPCRA.ACSE 位都为“0”时, 在执行 WAIT 指令后转移到睡眠模式。

11.2.3 模块停止控制寄存器 B (MSTPCRB)

地址 0008 0014h

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
MSTPB 31	MSTPB 30	—	—	—	MSTPB 26	MSTPB 25	—	MSTPB 23	—	MSTPB 21	—	—	—	MSTPB 17	—
复位后的值	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	MSTPB 10	MSTPB9	MSTPB8	—	MSTPB6	—	MSTPB4	—	—	—	—
复位后的值	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

位	符号	位名	功能	R/W
b3-b0	—	保留位	读写值都为“1”。	R/W
b4	MSTPB4	串行通信接口 SCId 的模块停止设定位	对象模块: SCId (SCI12) 0: 解除模块停止状态 1: 转移到模块停止状态	R/W
b5	—	保留位	读写值都为“1”。	R/W
b6	MSTPB6	DOC 模块停止设定位	对象模块: DOC 0: 解除模块停止状态 1: 转移到模块停止状态	R/W
b7	—	保留位	读写值都为“1”。	R/W
b8	MSTPB8	温度传感器的模块停止设定位	对象模块: 温度传感器 0: 解除模块停止状态 1: 转移到模块停止状态	R/W
b9	MSTPB9	ELC 模块停止设定位	对象模块: ELC 0: 解除模块停止状态 1: 转移到模块停止状态	R/W
b10	MSTPB10	比较器 B 的模块停止设定位	对象模块: 比较器 B 0: 解除模块停止状态 1: 转移到模块停止状态	R/W
b16-b11	—	保留位	读写值都为“1”。	R/W
b17	MSTPB17	串行外围接口 0 的模块停止设定位	对象模块: RSPI0 0: 解除模块停止状态 1: 转移到模块停止状态	R/W
b20-b18	—	保留位	读写值都为“1”。	R/W
b21	MSTPB21	I ² C 总线接口 0 的模块停止设定位	对象模块: RIIC0 0: 解除模块停止状态 1: 转移到模块停止状态	R/W
b22	—	保留位	读写值都为“1”。	R/W
b23	MSTPB23	CRC 运算器的模块停止设定位	对象模块: CRC 0: 解除模块停止状态 1: 转移到模块停止状态	R/W
b24	—	保留位	读写值都为“1”。	R/W

位	符号	位名	功能	R/W
b25	MSTPB25	串行通信接口 6 的模块停止设定位	对象模块: SCI6 0: 解除模块停止状态 1: 转移到模块停止状态	R/W
b26	MSTPB26	串行通信接口 5 的模块停止设定位	对象模块: SCI5 0: 解除模块停止状态 1: 转移到模块停止状态	R/W
b29-b27	—	保留位	读写值都为“1”。	R/W
b30	MSTPB30	串行通信接口 1 的模块停止设定位	对象模块: SCI1 0: 解除模块停止状态 1: 转移到模块停止状态	R/W
b31	MSTPB31	串行通信接口 0 的模块停止设定位	对象模块: SCI0 0: 解除模块停止状态 1: 转移到模块停止状态	R/W

11.2.4 模块停止控制寄存器 C (MSTPCRC)

地址 0008 0018h

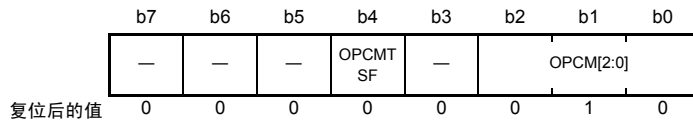
	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	MSTPC 27	MSTPC 26	—	—	—	—	—	—	MSTPC 19	—	—	—
复位后的值	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTPC0
复位后的值	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	MSTPC0	RAM0 模块停止设定位 (注 1)	对象模块: RAM0 (0000 0000h ~ 0000 FFFF) 0: RAM0 运行 1: RAM0 停止	R/W
b15-b1	—	保留位	读写值都为“0”。	R/W
b18-b16	—	保留位	读写值都为“1”。	R/W
b19	MSTPC19	时钟频率精度测量电路的模块停止设定位 (注 2)	对象模块: CAC 0: 解除模块停止状态 1: 转移到模块停止状态	R/W
b25-b20	—	保留位	读写值都为“1”。	R/W
b26	MSTPC26	串行通信接口 9 的模块停止设定位	对象模块: SCI9 0: 解除模块停止状态 1: 转移到模块停止状态	R/W
b27	MSTPC27	串行通信接口 8 的模块停止设定位	对象模块: SCI8 0: 解除模块停止状态 1: 转移到模块停止状态	R/W
b31-b28	—	保留位	读写值都为“1”。	R/W

- 注 1. 不能在存取内部RAM的过程中将对应的MSTPC0位置“1”，而且不能在MSTPC0位为“1”的状态下存取对应的RAM。
- 注 2. 只能在此位控制的时钟为振荡稳定的状态下改写 MSTPC19 位。如果要在改写此位后转移到软件待机模式，就必须在经过2个振荡器（在改写后保持振荡的振荡器中，输出振荡最慢时钟的振荡器）的输出时钟周期后执行WAIT指令。

11.2.5 运行功率控制寄存器 (OPCCR)

地址 0008 00A0h



位	符号	位名	功能	R/W
b2-b0	OPCM[2:0]	运行功率控制模式选择位	b2 b0 0 0 0: 高速运行模式 0 1 0: 中速运行模式 A 0 1 1: 中速运行模式 B 1 1 0: 低速运行模式 1 1 1 1: 低速运行模式 2 上述以外: 不能设定	R/W
b3	—	保留位	读写值都为“0”。	R/W
b4	OPCMTSF	运行功率控制模式转移状态标志	• 读时 0: 转移结束 1: 转移过程中 • 写时 只能写“0”。	R/W
b7-b5	—	保留位	读写值都为“0”。	R/W

OPCCR 寄存器用于降低正常运行模式、睡眠模式和全模块时钟停止模式中的功耗。

能通过设定 OPCCR 寄存器，根据使用的工作频率和工作电压来降低功耗。

在以下情况下禁止改写 OPCCR 寄存器。

- 当运行功率控制模式转移状态标志 (OPCMTSF) 为“1” (运行功率控制模式的转移过程中) 时
- 当闪存 P/E 模式入口寄存器的 ROM P/E 模式入口位 0 (FENTRYR.FENTRY0) 为“1” (ROM P/E 模式) 时
- 在从发行用于转移到睡眠模式的 WAIT 指令开始到从睡眠模式返回到正常运行模式为止的期间

不能在闪存的编程 / 擦除 (P/E) 过程中写 OPCCR 寄存器，写操作无效。

有关向运行功率控制模式转移的步骤，请参照“11.5 低运行功率功能”。

OPCM[2:0] 位 (运行功率控制模式选择位)

这些位选择正常运行模式、睡眠模式和全模块时钟停止模式中的运行功率控制模式。

运行功率控制模式和工作频率范围、工作电压范围、功耗的关系如表 11.3 所示。

表 11.3 运行功率控制模式和运行范围、功耗的关系

运行功率控制模式	OPCM[2:0] 位	工作频率范围						工作电压范围		消耗功率
		读闪存时					闪存 P/E 时	读闪存时	闪存 P/E 时	
		ICLK	FCLK	PCLKD	PCLKB	BCLK	FCLK			
高速运行模式	000b	50MHz max	32MHz max	50MHz max	32MHz max	25MHz max	4MHz ~ 32MHz	2.7V ~ 5.5V	2.7V ~ 5.5V	大 ↓ 小
中速运行模式 A	010b	32MHz max (注 1)	32MHz max (注 1)	32MHz max (注 1)	32MHz max (注 1)	25MHz max (注 1)	4MHz ~ 32MHz	1.62V ~ 5.5V (注 1)	2.7V ~ 5.5V	
中速运行模式 B	011b	32MHz max (注 1)	32MHz max (注 1)	32MHz max (注 1)	32MHz max (注 1)	25MHz max (注 1)	4MHz ~ 32MHz (注 1)	1.62V ~ 5.5V (注 1)	1.62V ~ 3.6V (注 1)	
低速运行模式 1	110b	1MHz max	1MHz max	1MHz max	1MHz max	1MHz max	不能进行 P/E	1.62V ~ 5.5V	不能进行 P/E	
低速运行模式 2	111b	32kHz max	32kHz max	32kHz max	32kHz max	32kHz max	不能进行 P/E	1.62V ~ 5.5V	不能进行 P/E	

注 1. 但是，当 $1.62V \leq \text{电压范围} < 1.8V$ 时，工作频率为 20MHz max。

• 高速运行模式

这是能进行高速运行的模式。

在读闪存 (FLASH) 时，ICLK 和 PCLKD 的最大工作频率是 50MHz，FCLK 和 PCLKB 的最大工作频率是 32MHz，BCLK 的最大工作频率是 25MHz。在对 FLASH 进行编程 / 擦除 (P/E) 时，FCLK 能以 4MHz ~ 32MHz 运行。在读 FLASH 或者对 FLASH 进行编程 / 擦除 (P/E) 时，工作电压范围都是 2.7V ~ 5.5V。如果在 1.62V ~ 2.7V (不包括 2.7V) 的电压范围下使用，就禁止设定此模式。

高速运行模式中的工作电压和工作频率的关系如图 11.2 所示。

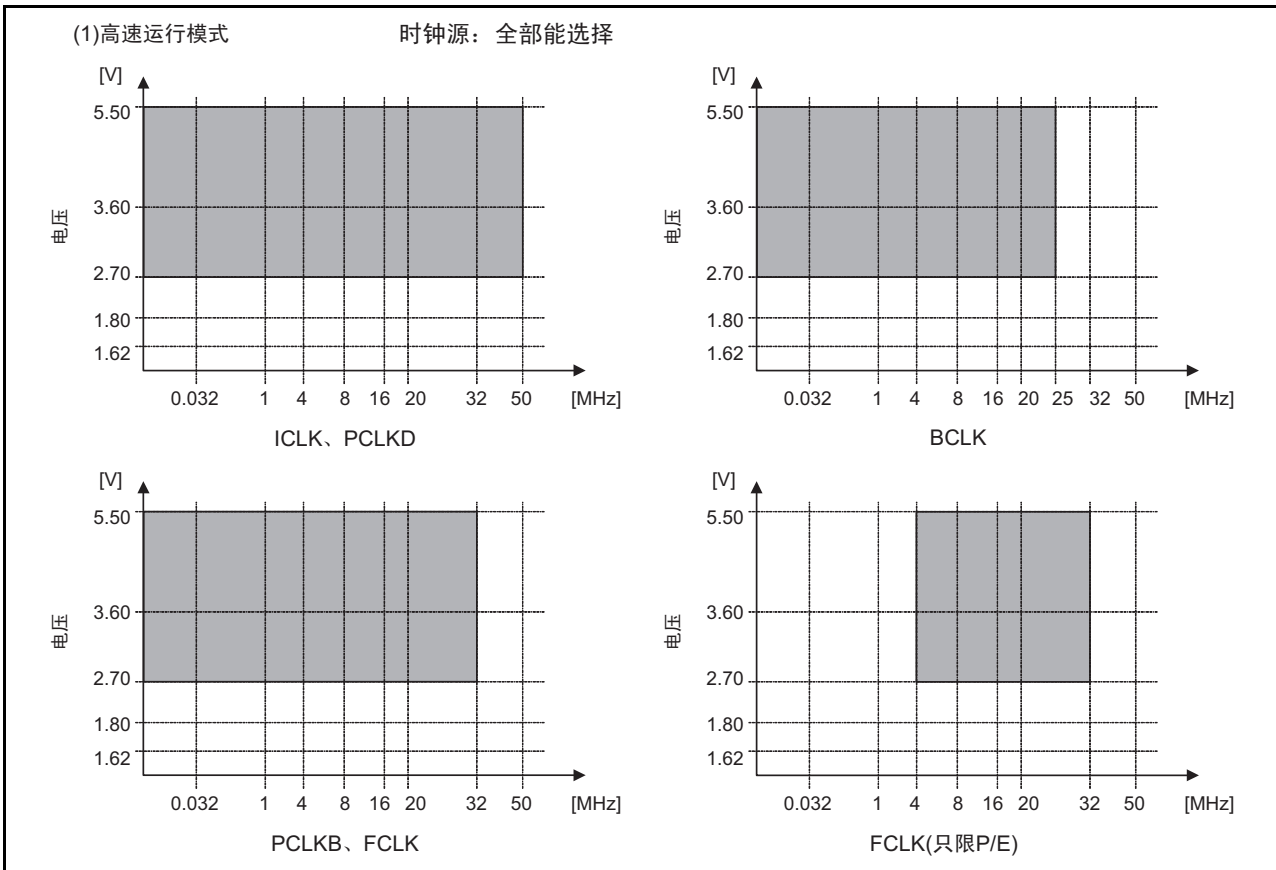


图 11.2 高速运行模式中的工作电压和工作频率的关系

• 中速运行模式 A

这是能在广电压范围下运行的模式。

在读 FLASH 时，ICLK、FCLK、PCLKB 和 PCLKD 的最大工作频率是 32MHz，BCLK 的最大工作频率是 25MHz，工作电压范围是 1.62V ~ 5.5V。但是，如果在 1.62V ~ 1.8V（不包括 1.8V）的电压范围下读 FLASH 时，ICLK、FCLK、PCLKB、PCLKD 和 BCLK 的最大工作频率都被限制为 20MHz。

对 FLASH 进行 P/E 时的工作频率范围是 4MHz ~ 32MHz，工作电压范围是 2.7V ~ 5.5V。

在相同条件下（频率、电压）进行相同运行时，中速运行模式 A 比高速运行模式能降低功耗。在解除复位后，在此模式中启动。

• 中速运行模式 B

这是以中速运行模式 A 为基础，面向在低速低工作电压范围下对 FLASH 进行 P/E 的低功耗模式。

读 FLASH 时的工作频率范围和工作电压范围与中速运行模式 A 相同。

对 FLASH 进行 P/E 时的工作频率范围是 4MHz ~ 32MHz，工作电压范围是 1.62V ~ 3.6V。但是，在 1.62V ~ 1.8V（不包括 1.8V）的电压范围下进行 P/E 时的最大工作频率被限制为 20MHz。

在相同条件下（频率、电压）进行相同运行时，中速运行模式 B 比高速运行模式能降低功耗。中速运行模式 A 和中速运行模式 B 中的工作电压和工作频率的关系如图 11.3 所示。

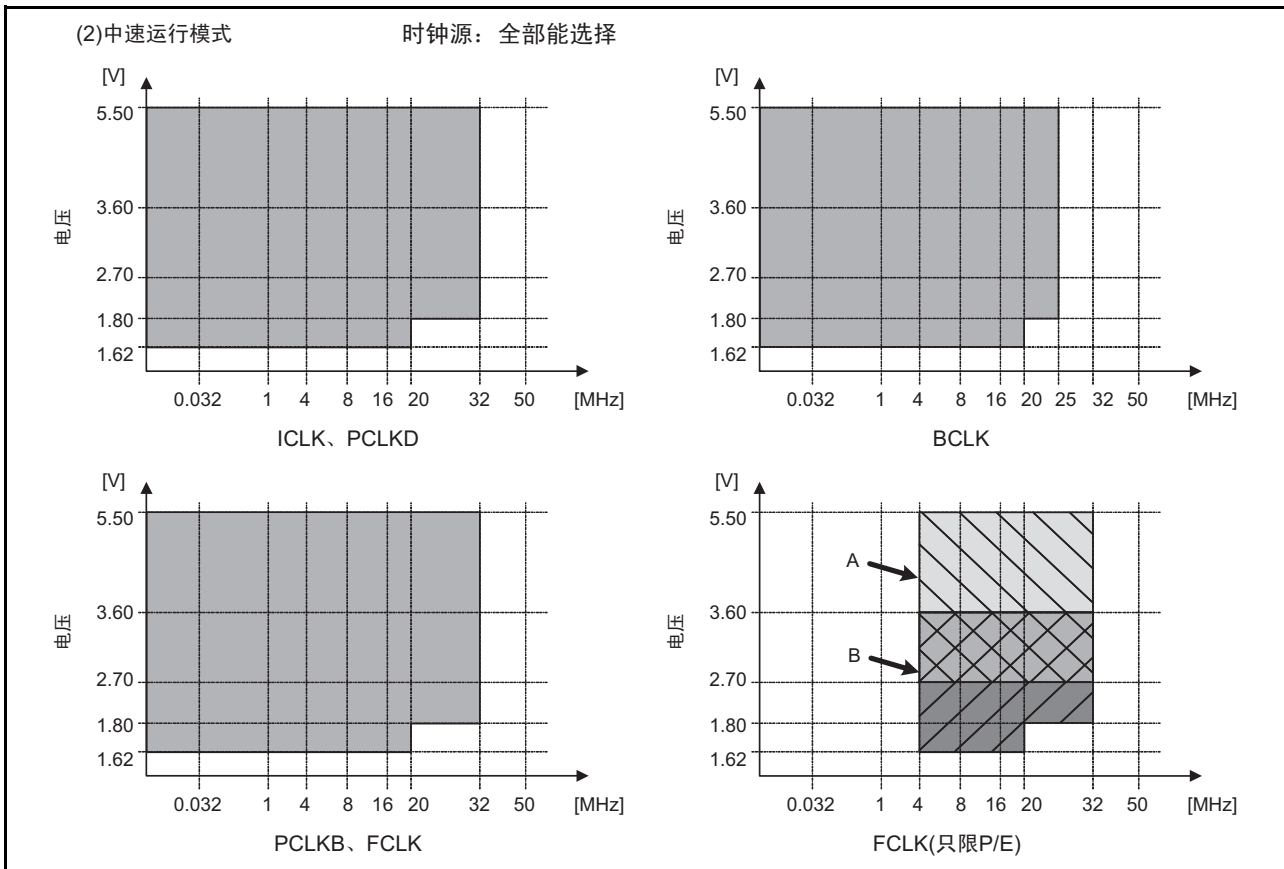


图 11.3 中速运行模式 A 和中速运行模式 B 中的工作电压和工作频率的关系

• 低速运行模式 1

这是面向比中速运行模式 A 和中速运行模式 B 进行更低速运行的低功耗模式。

在读 FLASH 时，ICLK、FCLK、PCLKD、PCLKB 和 BCLK 的最大工作频率都是 1MHz，工作电压范围是 1.62V ~ 5.5V。

不能在低速运行模式 1 中对 FLASH 进行编程/擦除 (P/E)，而且禁止给 PLLCR2.PLEN 位写“0” (PLL 运行)。

在相同条件 (频率、电压) 下进行相同运行时，低速运行模式 1 比中速运行模式 A 和中速运行模式 B 还能降低功耗。

低速运行模式 1 中的工作电压和工作频率的关系如图 11.4 所示。

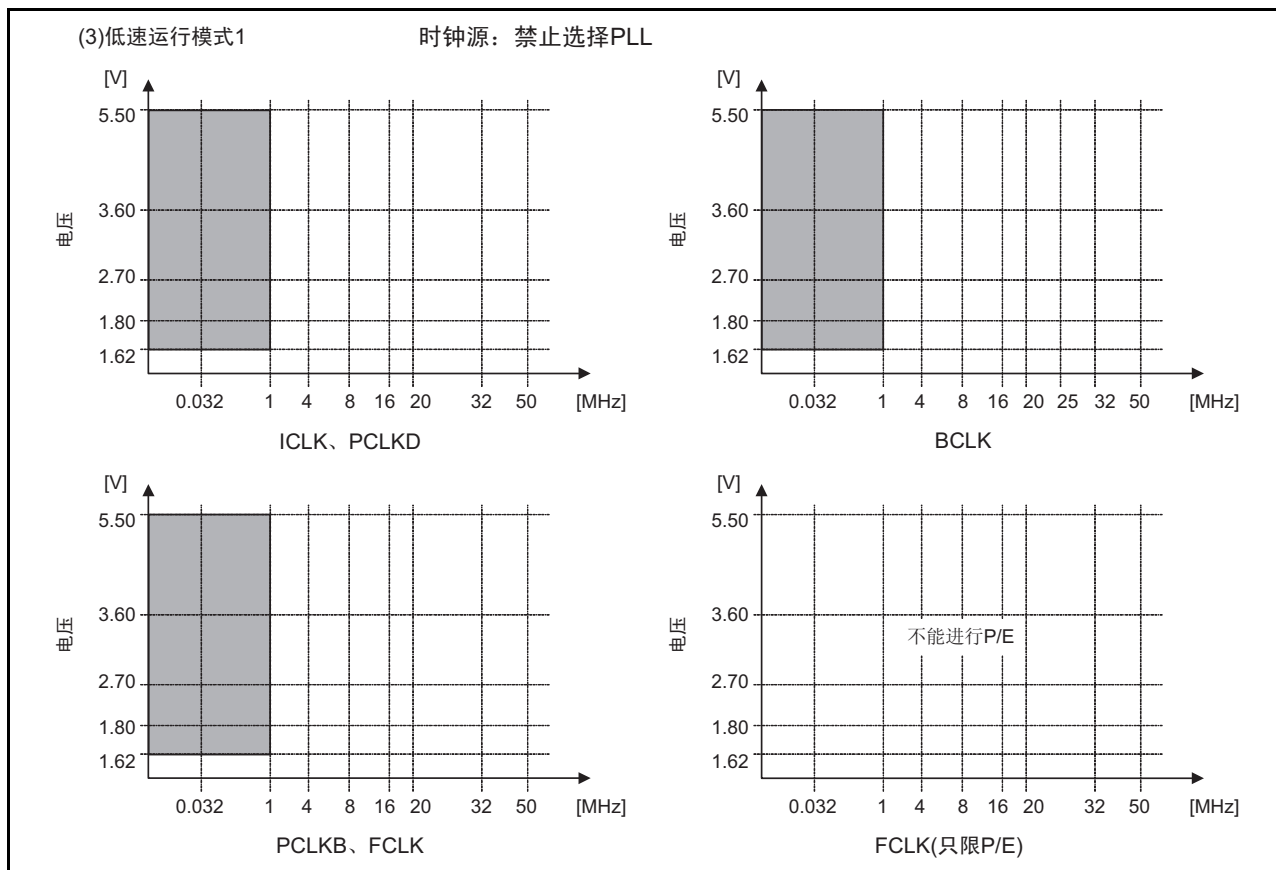


图 11.4 低速运行模式 1 中的工作电压和工作频率的关系

• 低速运行模式 2

这是面向比低速运行模式 1 进行更低速运行的低功耗模式。

在读 FLASH 时, ICLK、FCLK、PCLKD、PCLKB 和 BCLK 的最大工作频率都是 32kHz, 工作电压范围是 1.62V ~ 5.5V。

在选择低速运行模式 2 时, 有以下限制事项:

- 禁止对闪存进行 P/E。
- 禁止读数据闪存。
- 禁止使用 PLL 和 HOCO。
- 禁止使用主时钟振荡器的振荡停止检测功能。
- 禁止使用 ELC 的时钟源转换功能

在相同条件 (频率、电压) 下进行相同运行时, 低速运行模式 2 比低速运行模式 1 还能降低功耗。

注 1. 当 PLLCR2.PLLEN 位为“0” (PLL 运行) 时, 不能给 OPCM[2:0] 位写“110b” (低速运行模式 1) 和“111b” (低速运行模式 2)。

当 HOCOCR.HCSTP 位为“0” (HOCO 运行) 时, 不能给 OPCM[2:0] 位写“111b” (低速运行模式 2),

低速运行模式 2 中的工作电压和工作频率的关系如图 11.5 所示。

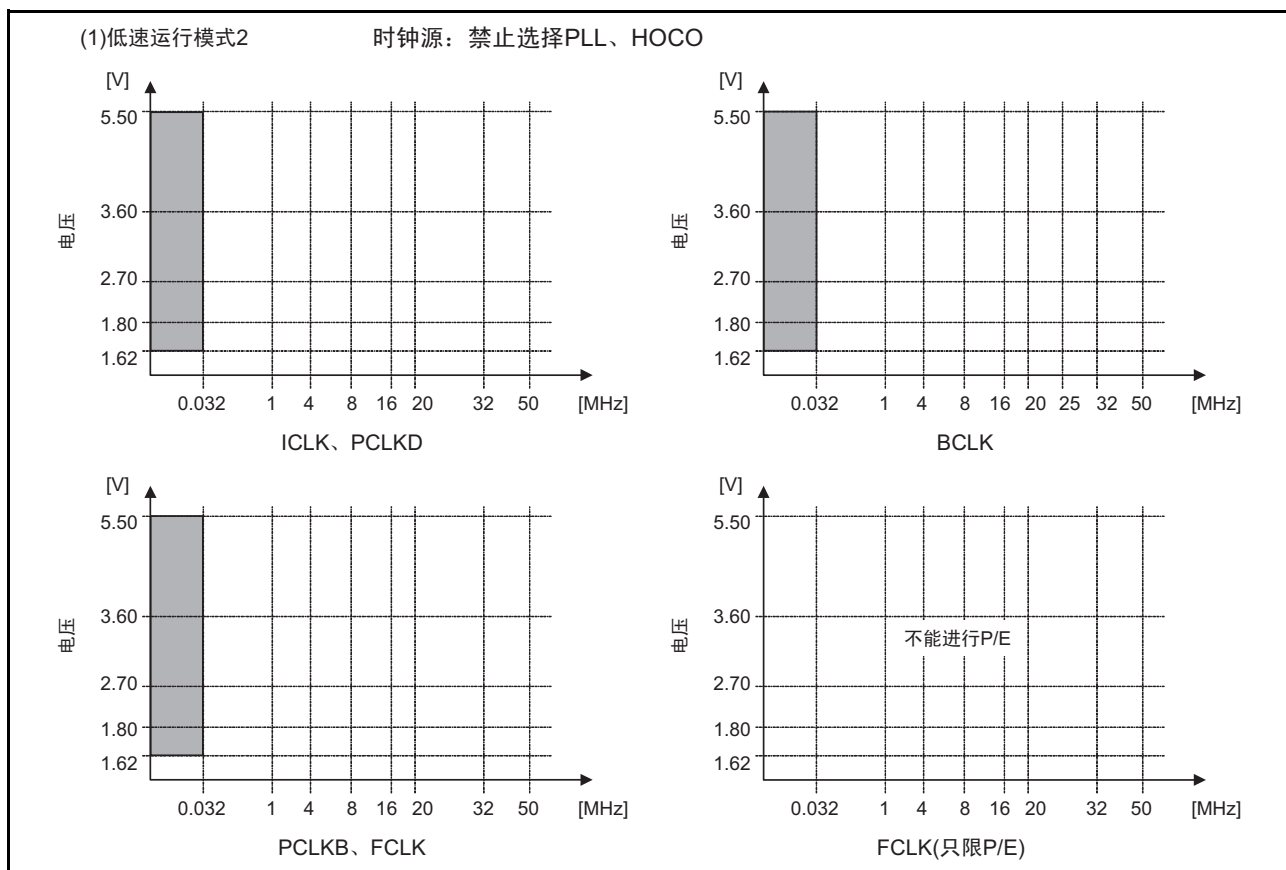


图 11.5 低速运行模式 2 中的工作电压和工作频率的关系

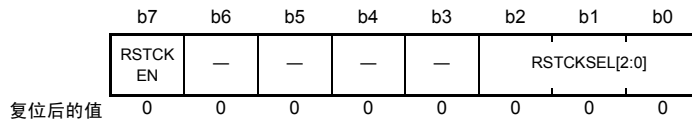
OPCMTSF 标志 (运行功率控制模式转移状态标志)

此标志表示转换运行功率控制模式时的转换控制状态。

如果更改运行功率控制模式, OPCMTSF 标志就变为“1”; 如果已转移到更改后的运行功率控制模式, 此标志就变为“0”。必须在确认 OPCMTSF 标志为“0” (结束运行功率控制模式的转移) 后进行之后的处理。

11.2.6 睡眠模式返回时钟源转换寄存器 (RSTCKCR)

地址 0008 00A1h



位	符号	位名	功能	R/W
b2-b0	RSTCKSEL [2:0]	睡眠模式返回时钟源选择位	b2 b0 0 0 1: 选择 HOCO 0 1 0: 选择主时钟振荡器 当 RSTCKEN 位为“1”时, 不能进行上述以外的设定。	R/W
b6-b3	—	保留位	读写值都为“0”。	R/W
b7	RSTCKEN	睡眠模式返回时钟源转换允许位	0: 在解除睡眠模式时, 时钟源的转换无效。 1: 在解除睡眠模式时, 时钟源的转换有效。	R/W

RSTCKCR 寄存器是控制解除睡眠模式时的时钟源转换的寄存器。

在通过设定 RSTCKCR 寄存器从睡眠模式返回时, 自动将返回时钟源对应的主时钟振荡器控制寄存器的主时钟停止位 (MOSCCR.MOSTP) 和高速内部振荡器控制寄存器的 HOCO 停止位 (HOCOCR.HCSTP) 改写为运行状态, 并且自动将 RSTCKSEL[2:0] 位的值重新加载到系统时钟控制寄存器 3 的时钟源选择位 (SCKCR3.CKSEL[2:0])。

在通过设定 RSTCKCR 寄存器并且通过 HOCO 从睡眠模式返回时, HOCO 电源不会自动接通。要通过 HOCO 返回时, 必须在 HOCO 电源接通的状态下转移到睡眠模式。

从睡眠模式返回时, 根据 SCKCR 寄存器和 RSTCKSEL[2:0] 位的设定, OPCCR.OPCM[2:0] 位自动转换为中速运行模式 A (“010b”) 或者高速运行模式 (“000b”)。

如果给 RSTCKSEL[2:0] 位设定 “001b” (选择 HOCO) 并且允许转换睡眠模式返回时钟源, 就必须在转移到睡眠模式前给 SCKCR 寄存器的 FCK 位、ICK 位、BCK 位、PCKD 位和 PCKB 位设定至少 2 分频。

禁止同时使用睡眠模式返回时钟源转换功能和通过 ELC 进行时钟源转换的功能。

如果将睡眠模式返回时钟源转换功能置为有效, 就必须在通过 ELC 进行时钟源转换的功能无效的状态下将 RSTCKEN 位置 “1”。如果要将通过 ELC 进行时钟源转换的功能置为有效, 就必须在 RSTCKEN 位为 “0” 的状态下置为有效。

RSTCKSEL[2:0] 位 (睡眠模式返回时钟源选择位)

这些位选择解除睡眠模式时的时钟源。

只能在 RSTCKEN 位为 “1” 时通过 RSTCKSEL[2:0] 位选择时钟源。

如果 SCKCR 寄存器的 FCK 位、ICK 位、BCK 位、PCKA 位、PCKB 位、PCKC 位或者 PCKD 位为 “0000b” (1 分频), 就禁止将 RSTCKSEL 位设定为 “001b” (选择 HOCO)。

RSTCKEN 位 (睡眠模式返回时钟源转换允许位)

此位控制解除睡眠模式时的时钟源转换的有效或者无效。

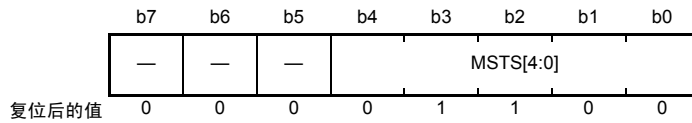
只有在选择 LOCO 和副时钟作为转移到睡眠模式时的时钟的情况下, 才能在解除睡眠模式时转换时钟源。要在选择 HOCO、主时钟或者 PLL 作为时钟源的状态下转移到睡眠模式时, 不能将 RSTCKEN 位置 “1”。

如果在将此位设定为有效的状态下从睡眠模式返回, OPCCR.OPCM[2:0] 位就自动改写为中速运行模式 A。

如果在设定为高速运行模式下转移到睡眠模式, 就无法改写到中速运行模式 A, 在复位后也保持高速运行模式。

11.2.7 主时钟振荡器的等待控制寄存器 (MOSCWTCR)

地址 0008 00A2h



位	符号	位名	功能	R/W
b4-b0	MSTS[4:0]	主时钟振荡器的等待时间设定位	b4 b0 0 0 0 0 0: 待机时间 =2 个周期 0 0 0 0 1: 待机时间 =4 个周期 0 0 0 1 0: 待机时间 =8 个周期 0 0 0 1 1: 待机时间 =16 个周期 0 0 1 0 0: 待机时间 =32 个周期 0 0 1 0 1: 待机时间 =256 个周期 0 0 1 1 0: 待机时间 =512 个周期 0 0 1 1 1: 待机时间 =1024 个周期 0 1 0 0 0: 待机时间 =2048 个周期 0 1 0 0 1: 待机时间 =4096 个周期 0 1 0 1 0: 待机时间 =16384 个周期 0 1 0 1 1: 待机时间 =32768 个周期 0 1 1 0 0: 待机时间 =65536 个周期 0 1 1 0 1: 待机时间 =131072 个周期 0 1 1 1 0: 待机时间 =262144 个周期 0 1 1 1 1: 待机时间 =524288 个周期 上述以外: 不能设定	R/W
b7-b5	—	保留位	读写值都为“0”。	R/W

MOSCWTCR 寄存器是选择主时钟振荡器的振荡稳定等待时间的寄存器。

在对主时钟进行 MOSCWTCR 寄存器所设周期数的计数后, 开始给 LSI 内部提供主时钟。

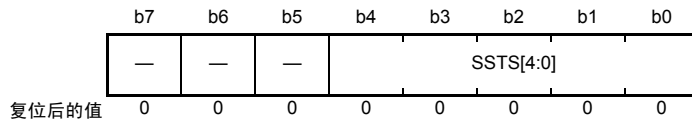
必须将 MSTS[4:0] 位至少设定为主时钟振荡稳定时间 (tMAINOSC) 的待机时间。例如, 当使用的谐振器的振荡频率为 10MHz (周期为 100ns) 时, 如果将 MSTS[4:0] 位置 “01101b”, 待机时间就为 100ns×131072 个周期 ≈ 13.12ms。

在将主时钟用于外部输入时, 不需要待机时间。

只能在 MOSCCR.MOSTP 位为 “1” 时改写 MOSCWTCR 寄存器, 否则就不能进行改写。

11.2.8 副时钟振荡器的等待控制寄存器 (SOSCWTCR)

地址 0008 00A3h



位	符号	位名	功能	R/W
b4-b0	SSTS[4:0]	副时钟振荡器的等待时间设定位	b4 b0 0 0 0 0 0: 待机时间 =2s+2 个周期 0 0 0 0 1: 待机时间 =2s+4 个周期 0 0 0 1 0: 待机时间 =2s+8 个周期 0 0 0 1 1: 待机时间 =2s+16 个周期 0 0 1 0 0: 待机时间 =2s+32 个周期 0 0 1 0 1: 待机时间 =2s+64 个周期 0 0 1 1 0: 待机时间 =2s+512 个周期 0 0 1 1 1: 待机时间 =2s+1024 个周期 0 1 0 0 0: 待机时间 =2s+2048 个周期 0 1 0 0 1: 待机时间 =2s+4096 个周期 0 1 0 1 0: 待机时间 =2s+16384 个周期 0 1 0 1 1: 待机时间 =2s+32768 个周期 0 1 1 0 0: 待机时间 =2s+65536 个周期 0 1 1 0 1: 待机时间 =2s+131072 个周期 0 1 1 1 0: 待机时间 =2s+262144 个周期 0 1 1 1 1: 待机时间 =2s+524288 个周期 上述以外: 不能设定	R/W
b7-b5	—	保留位	读写值都为“0”。	R/W

SOSCWTCR 寄存器是选择副时钟振荡器的振荡稳定等待时间的寄存器。

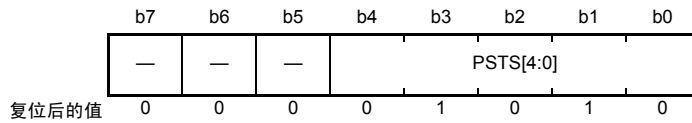
在对副时钟进行 SOSCWTCR 寄存器所设周期数的计数后, 开始给 LSI 内部提供主时钟。

必须将 SSTS[4:0] 位至少设定为副时钟振荡稳定时间 (t_{SUBOSC}) 的待机时间。例如, 当使用的谐振器的振荡频率为 32.768kHz (周期为 30.5μs) 时, 如果将 SSTS[4:0] 位置“01011b”, 待机时间就为 2s+30.5μs×32768 个周期≈ 2s+1s=3s。

只能在 SOSCCR.SOSTP 位为“1”(副时钟振荡器停止)时改写 SOSCWTCR 寄存器, 否则就不能进行改写。

11.2.9 PLL 等待控制寄存器 (PLLWTCR)

地址 0008 00A6h



位	符号	位名	功能	R/W
b4-b0	PSTS[4:0]	PLL 等待时间设定位	b4 b0 00000: 待机时间 =16 个周期 00001: 待机时间 =32 个周期 00010: 待机时间 =64 个周期 00011: 待机时间 =512 个周期 00100: 待机时间 =1024 个周期 00101: 待机时间 =2048 个周期 00110: 待机时间 =4096 个周期 00111: 待机时间 =16384 个周期 01000: 待机时间 =32768 个周期 01001: 待机时间 =65536 个周期 01010: 待机时间 =131072 个周期 01011: 待机时间 =262144 个周期 01100: 待机时间 =524288 个周期 01101: 待机时间 =1048576 个周期 01110: 待机时间 =2097152 个周期 01111: 待机时间 =4194304 个周期 上述以外: 不能设定	R/W
b7-b5	—	保留位	读写值都为“0”。	R/W

PLLWTCR 寄存器是选择 PLL 振荡稳定等待时间的寄存器。

在对 PLL 时钟进行 PLLWTCR 寄存器所设周期数的计数后，开始给 LSI 内部提供 PLL 时钟。

在将晶体谐振器用作谐振器并且不等待主时钟振荡稳定就将 PLL 设定为运行的情况下，必须将 PSTS[4:0] 位至少设定为 PLL 振荡稳定时间（不等主时钟振荡稳定就将 PLL 设定为 ON 的情况）(t_{PLL2}) 的待机时间。例如，当 PLL 振荡频率为 100MHz（周期为 10ns）时，如果将 PSTS[4:0] 位置“01101b”，待机时间就为 10ns×1048576 个周期≈ 10.48ms。

在将晶体谐振器用作谐振器并且在等待主时钟振荡稳定后将 PLL 设定为运行以及将主时钟用于外部输入的情况下，因为不需要等待基准时钟的振荡稳定时间，所以必须将 PSTS[4:0] 位至少设定为 PLL 振荡稳定时间（在主时钟振荡稳定后将 PLL 设定为 ON 的情况）(t_{PLL1}) 的待机时间。

只能在 PLLCR2.PLLEN 位为“1”（PLL 停止）时改写 PLLWTCR 寄存器，否则就不能进行改写。

11.2.10 HOCO 等待控制寄存器 2 (HOCOWTCR2)

地址 0008 00A9h



位	符号	位名	功能	R/W
b3-b0	HSTS2[3:0]	HOCO 等待时间设定 2 位	b3 b0 0 0 0 0: 待机时间 =3072 个周期 0 0 0 1: 待机时间 =5120 个周期 0 0 1 0: 待机时间 =7168 个周期 当 HOCO 时钟的频率为 32MHz、 36.864MHz 或者 40MHz 时, 必须设定为 “0010b” (注 1)。 0 0 1 1: 待机时间 =9216 个周期 当 HOCO 时钟的频率为 50MHz 时, 必须 设定为 “0011b” (注 1)。 0 1 0 0: 待机时间 =11264 个周期 0 1 0 1: 待机时间 =13312 个周期 0 1 1 0: 待机时间 =15360 个周期 0 1 1 1: 待机时间 =17408 个周期 1 0 0 0: 待机时间 =19456 个周期 1 0 0 1: 待机时间 =21504 个周期 1 0 1 0: 待机时间 =23552 个周期 1 0 1 1: 待机时间 =25600 个周期 1 1 0 0: 待机时间 =27648 个周期 1 1 0 1: 待机时间 =29696 个周期 1 1 1 0: 待机时间 =31744 个周期 1 1 1 1: 待机时间 =33792 个周期 上述以外: 不能设定	R/W
b7-b5	—	保留位	读写值都为 “0”。	R/W

注 1. 如果设定此值, 就能确保 HOCO 振荡稳定时间 2 (tHOCO2), 并且在开始提供时钟后立即提供电特性中记载的 HOCO 振荡频率 (fHOCO) 精度的时钟。

即使设定比此设定值少的周期数, 也能提供时钟。但是, 因为不能确保 HOCO 振荡稳定时间 2 (tHOCO2), 所以在一开始提供时钟时就无法保证电特性中记载的 HOCO 频率精度。此时, 从振荡开始到经过 tHOCO2 后, 变为电特性中记载的 HOCO 频率精度。

HOCOWTCR2 寄存器是指定 HOCO 的振荡稳定时间的寄存器。

在对 HOCO 时钟进行 HOCOWTCR2 寄存器所设周期数的计数后, 开始给 LSI 内部提供 HOCO 时钟。

只能在 HOCOCR.HCSTP 位为 “1” (HOCO 停止) 时改写 HOCOWTCR2 寄存器, 否则就不能进行改写。

11.2.11 深度待机控制寄存器 (DPSBYCR)

地址 0008 C280h

	b7	b6	b5	b4	b3	b2	b1	b0
	DPSBY	IOKEEP	—	—	—	—	DEEPC UT1	—
复位后的值	0	0	0	0	0	0	0	1

位	符号	位名	功能	R/W
b0	—	保留位	读写值都为“1”。	R/W
b1	DEEPCUT1	深度切断位	0: 在设定软件待机模式中, LVD、POR 能运行。 1: 在设定软件待机模式中, LVD 停止运行, POR 以低功耗模式运行。	R/W
b5-b2	—	保留位	读写值都为“0”。	R/W
b6	IOKEEP	I/O 端口保持位	0: 在解除深度软件待机模式的同时, 解除 I/O 端口的保持状态。 1: 在解除深度软件待机模式后仍然保持 I/O 端口的状态。此后, 如果给 IOKEEP 位写“0”, 就解除 I/O 端口的保持状态。	R/W
b7	DPSBY	深度软件待机位	SSBY b7 0 0: 在执行 WAIT 指令后转移到睡眠模式或者全模块时钟停止模式 0 1: 在执行 WAIT 指令后转移到睡眠模式或者全模块时钟停止模式 1 0: 在执行 WAIT 指令后转移到软件待机模式 1 1: 在执行 WAIT 指令后转移到深度软件待机模式	R/W

注 1. 有关内部 RAM 的地址空间, 请参照表 11.2。

不能通过作为深度软件待机模式解除源的内部复位信号对 DPSBYCR 寄存器进行初始化, 详细内容请参照“表 6.2 各种复位的初始化对象”。

IOKEEP 位 (I/O 端口保持位)

在深度软件待机模式中, I/O 端口保持和软件待机模式中相同的状态。通过 IOKEEP 位选择在解除深度软件待机模式后是继续保持还是解除在深度软件待机模式中保持的 I/O 端口状态。

DPSBY 位 (深度软件待机位)

此位控制向深度软件待机模式的转移。

如果在 SBYCR.SSBY 位为“1”并且 DPSBY 位为“1”的状态下执行 WAIT 指令, 就经过软件待机模式转移到深度软件待机模式。

如果通过作为外部引脚中断发生源的部分引脚 (NMI、IRQ0-DS ~ IRQ7-DS、SCL-DS、SDA-DS) 或者外围功能中断 (RTC 闹钟、RTC 周期、电压监视 1、电压监视 2) 解除深度软件待机模式, DPSBY 位就保持“1”。要将此位置“0”时, 必须写“0”。

在 IWDT 为自动启动模式并且 OFS0.IWDTSLCSTP 位为“0” (继续进行计数) 时或者在 IWDT 为寄存器启动模式并且 IWDTCTPR.SLCSTP 位为“0”时, DPSBY 位的设定值无效。此时, 即使 SBYCR.SSBY 位为“1”并且 DPSBY 位为“1”, 也在执行 WAIT 指令后转移到软件待机模式。

如果在电压监视 1 电路模式中选择电压监视 1 复位 (LVD1CR0.LVD1RI=1) 或者在电压监视 2 电路模式中选择电压监视 2 复位 (LVD2CR0.LVD2RI=1), DPSBY 位的设定值就无效。此时, 即使 SBYCR.SSBY 位为“1”并且 DPSBY 位为“1”, 也在执行 WAIT 指令后转移到软件待机模式。

11.2.12 深度待机中断允许寄存器 0 (DPSIER0)

地址 0008 C282h

b7	b6	b5	b4	b3	b2	b1	b0
DIRQ7E	DIRQ6E	DIRQ5E	DIRQ4E	DIRQ3E	DIRQ2E	DIRQ1E	DIRQ0E
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	DIRQ0E	IRQ0-DS 引脚允许位	0: 禁止通过 IRQ0-DS 引脚解除深度软件待机模式 1: 允许通过 IRQ0-DS 引脚解除深度软件待机模式	R/W
b1	DIRQ1E	IRQ1-DS 引脚允许位	0: 禁止通过 IRQ1-DS 引脚解除深度软件待机模式 1: 允许通过 IRQ1-DS 引脚解除深度软件待机模式	R/W
b2	DIRQ2E	IRQ2-DS 引脚允许位	0: 禁止通过 IRQ2-DS 引脚解除深度软件待机模式 1: 允许通过 IRQ2-DS 引脚解除深度软件待机模式	R/W
b3	DIRQ3E	IRQ3-DS 引脚允许位	0: 禁止通过 IRQ3-DS 引脚解除深度软件待机模式 1: 允许通过 IRQ3-DS 引脚解除深度软件待机模式	R/W
b4	DIRQ4E	IRQ4-DS 引脚允许位	0: 禁止通过 IRQ4-DS 引脚解除深度软件待机模式 1: 允许通过 IRQ4-DS 引脚解除深度软件待机模式	R/W
b5	DIRQ5E	IRQ5-DS 引脚允许位	0: 禁止通过 IRQ5-DS 引脚解除深度软件待机模式 1: 允许通过 IRQ5-DS 引脚解除深度软件待机模式	R/W
b6	DIRQ6E	IRQ6-DS 引脚允许位	0: 禁止通过 IRQ6-DS 引脚解除深度软件待机模式 1: 允许通过 IRQ6-DS 引脚解除深度软件待机模式	R/W
b7	DIRQ7E	IRQ7-DS 引脚允许位	0: 禁止通过 IRQ7-DS 引脚解除深度软件待机模式 1: 允许通过 IRQ7-DS 引脚解除深度软件待机模式	R/W

不能通过作为深度软件待机模式解除源的内部复位信号对 DPSIER0 寄存器进行初始化，详细内容请参照“表 6.2 各种复位的初始化对象”。

如果更改 DPSIER0 寄存器的设定，就可能根据引脚状态在内部产生边沿，使 DPSIFR0 寄存器变为“1”。必须在向深度软件待机模式转移前将 DPSIFR0 寄存器置“0”。

即使 DPSIER0 寄存器为“0”，也可能在向深度软件待机模式转移时，根据引脚状态在内部产生上升沿，使 DPSIFR0 寄存器变为“1”。因为在 DPSIEGR0 寄存器为“0”时不检测上升沿，所以 DPSIFR0 寄存器不变为“1”。

11.2.13 深度待机中断允许寄存器 2 (DPSIER2)

地址 0008 C284h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	DRIICC IE	DRIICD IE	DNMIE	DRTCA IE	DRTCIE	DLVD2IE	DLVD1IE
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	DLVD1IE	LVD1 深度待机解除信号允许位	0: 禁止通过电压监视 1 信号解除深度软件待机模式 1: 允许通过电压监视 1 信号解除深度软件待机模式	R/W
b1	DLVD2IE	LVD2 深度待机解除信号允许位	0: 禁止通过电压监视 2 信号解除深度软件待机模式 1: 允许通过电压监视 2 信号解除深度软件待机模式	R/W
b2	DRTCIE	RTC 周期中断深度待机解除信号允许位	0: 禁止通过 RTC 周期中断信号解除深度软件待机模式 1: 允许通过 RTC 周期中断信号解除深度软件待机模式	R/W
b3	DRTCAIE	RTC 闹钟中断深度待机解除信号允许位	0: 禁止通过 RTC 闹钟中断信号解除深度软件待机模式 1: 允许通过 RTC 闹钟中断信号解除深度软件待机模式	R/W
b4	DNMIE	NMI 引脚允许位	0: 禁止通过 NMI 引脚解除深度软件待机模式 1: 允许通过 NMI 引脚解除深度软件待机模式	R/W (注 1)
b5	DRIICDIE	SDA-DS 深度待机解除信号允许位	0: 禁止通过 SDA-DS 信号解除深度软件待机模式 1: 允许通过 SDA-DS 信号解除深度软件待机模式	R/W
b6	DRIICDIE	SCL-DS 深度待机解除信号允许位	0: 禁止通过 SCL-DS 信号解除深度软件待机模式 1: 允许通过 SCL-DS 信号解除深度软件待机模式	R/W
b7	—	保留位	读写值都为“0”。	R/W

注 1. 只能写 1 次“1”，以后的写操作无效。

作为深度软件待机模式解除源的内部复位信号不对 DPSIER2 寄存器进行初始化，详细内容请参照“表 6.2 各种复位的初始化对象”。

如果更改 DPSIER2 寄存器的设定，就可能根据引脚状态在内部产生边沿，使 DPSIFR2 寄存器变为“1”。必须在向深度软件待机模式转移前将 DPSIFR2 寄存器置“0”。

即使 DPSIER2 寄存器为“0”，也可能在向深度软件待机模式转移时，根据引脚状态在内部产生上升沿，使 DPSIFR2 寄存器变为“1”。因为在 DPSIEGR2 寄存器为“0”时不检测上升沿，所以 DPSIFR2 寄存器不变为“1”。

11.2.14 深度待机中断标志寄存器 0 (DPSIFR0)

地址 0008 C286h

b7	b6	b5	b4	b3	b2	b1	b0
DIRQ7F	DIRQ6F	DIRQ5F	DIRQ4F	DIRQ3F	DIRQ2F	DIRQ1F	DIRQ0F
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	DIRQ0F	IRQ0-DS 引脚深度待机解除标志	0: 未发生由 IRQ0-DS 引脚引起的解除请求 1: 发生由 IRQ0-DS 引脚引起的解除请求	R(W) (注1)
b1	DIRQ1F	IRQ1-DS 引脚深度待机解除标志	0: 未发生由 IRQ1-DS 引脚引起的解除请求 1: 发生由 IRQ1-DS 引脚引起的解除请求	R(W) (注1)
b2	DIRQ2F	IRQ2-DS 引脚深度待机解除标志	0: 未发生由 IRQ2-DS 引脚引起的解除请求 1: 发生由 IRQ2-DS 引脚引起的解除请求	R(W) (注1)
b3	DIRQ3F	IRQ3-DS 引脚深度待机解除标志	0: 未发生由 IRQ3-DS 引脚引起的解除请求 1: 发生由 IRQ3-DS 引脚引起的解除请求	R(W) (注1)
b4	DIRQ4F	IRQ4-DS 引脚深度待机解除标志	0: 未发生由 IRQ4-DS 引脚引起的解除请求 1: 发生由 IRQ4-DS 引脚引起的解除请求	R(W) (注1)
b5	DIRQ5F	IRQ5-DS 引脚深度待机解除标志	0: 未发生由 IRQ5-DS 引脚引起的解除请求 1: 发生由 IRQ5-DS 引脚引起的解除请求	R(W) (注1)
b6	DIRQ6F	IRQ6-DS 引脚深度待机解除标志	0: 未发生由 IRQ6-DS 引脚引起的解除请求 1: 发生由 IRQ6-DS 引脚引起的解除请求	R(W) (注1)
b7	DIRQ7F	IRQ7-DS 引脚深度待机解除标志	0: 未发生由 IRQ7-DS 引脚引起的解除请求 1: 发生由 IRQ7-DS 引脚引起的解除请求	R(W) (注1)

注 1. 只能写“0”。

如果发生 DPSIEGR0 寄存器设定的解除请求，DPSIFR0 寄存器就变为“1”。

即使在非深度软件待机模式的状态下发生解除请求，DPSIFR0 寄存器也可能变为“1”。也有可能因更改 DPSIER0 寄存器的设定而使 DPSIFR0 寄存器变为“1”。因此必须在将 DPSIFR0 寄存器置“00h”后转移到深度软件待机模式。

要在更改 DPSIER0 寄存器的设定后将 DPSIFR0 寄存器置“00h”时，必须在至少经过 6 个 PCLKB 周期并且读 DPSIFR0 寄存器后，给 DPSIFR0 寄存器写“0”。例如，能通过读 DPSIER0 寄存器，至少确保 6 个 PCLKB 周期。

作为深度软件待机模式解除源的内部复位信号不对 DPSIFR0 寄存器进行初始化，详细内容请参照“表 6.2 各种复位的初始化对象”。

DIRQnF (IRQn 深度待机解除标志) (n=0 ~ 7)

此标志表示发生由 IRQn-DS 引脚引起的解除请求。

[为“1”的条件]

- 当发生由 DPSIEGR0 寄存器所选 IRQn-DS 引脚引起的解除请求时

[为“0”的条件]

- 在读“1”后写“0”时

11.2.15 深度待机中断标志寄存器 2 (DPSIFR2)

地址 0008 C288h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	DRIICC IF	DRIICD IF	DNMIF	DRTCA IF	DRTCIF	DLVD2IF	DLVD1IF
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	DLVD1IF	LVD1 深度待机解除标志	0: 未发生由电压监视 1 信号引起的解除请求 1: 发生由电压监视 1 信号引起的解除请求	R(W) (注 1)
b1	DLVD2IF	LVD2 深度待机解除标志	0: 未发生由电压监视 2 信号引起的解除请求 1: 发生由电压监视 2 信号引起的解除请求	R(W) (注 1)
b2	DRTCIF	RTC 周期中断深度待机解除标志	0: 未发生由 RTC 周期中断信号引起的解除请求 1: 发生由 RTC 周期中断信号引起的解除请求	R(W) (注 1)
b3	DRTCAIF	RTC 闹钟中断深度待机解除标志	0: 未发生由 RTC 闹钟中断信号引起的解除请求 1: 发生由 RTC 闹钟中断信号引起的解除请求	R(W) (注 1)
b4	DNMIF	NMI 深度待机解除标志	0: 未发生由 NMI 引脚引起的解除请求 1: 发生由 NMI 引脚引起的解除请求	R(W) (注 1)
b5	DRIICDIF	SDA-DS 深度待机解除标志	0: 未发生由 SDA-DS 信号引起的解除请求 1: 发生由 SDA-DS 信号引起的解除请求	R(W) (注 1)
b6	DRIICCIF	SCL-DS 深度待机解除标志	0: 未发生由 SCL-DS 信号引起的解除请求 1: 发生由 SCL-DS 信号引起的解除请求	R(W) (注 1)
b7	—	保留位	读写值都为“0”。	R(W)

注 1. 只能写“0”。

如果发生 DPSIEGR2 寄存器设定的解除请求，DPSIFR2 寄存器就变为“1”。

即使在非深度软件待机模式的状态下发生解除请求，DPSIFR2 寄存器也可能变为“1”。也有可能因更改 DPSIER2 寄存器的设定而使 DPSIFR2 寄存器变为“1”。因此必须在将 DPSIFR2 寄存器置“00h”后转移到深度软件待机模式。

要在更改 DPSIER2 寄存器的设定后将 DPSIFR2 寄存器置“00h”时，必须在至少经过 6 个 PCLKB 周期并且读 DPSIFR2 寄存器后，给 DPSIFR2 寄存器写“0”。例如，能通过读 DPSIER2 寄存器，至少确保 6 个 PCLKB 周期。

作为深度软件待机模式解除源的内部复位信号不对 DPSIFR2 寄存器进行初始化，详细内容请参照“表 6.2 各种复位的初始化对象”。

DLVDmIF (LVDm 深度待机解除标志) (m=1、2)

此标志表示发生由电压监视 m 信号引起的解除请求。

[为“1”的条件]

- 当发生由 DPSIEGR2 寄存器所选电压监视 m 信号引起的解除请求时

[为“0”的条件]

- 在读“1”后写“0”时

DRTCIF (RTC 周期中断深度待机解除标志)

此标志表示发生由 RTC 周期中断信号引起的解除请求。

[为 “1” 的条件]

- 当发生由 RTC 周期中断信号引起的解除请求时

[为 “0” 的条件]

- 在读“1”后写“0”时

DRTCAIF (RTC 闹钟中断深度待机解除标志)

此标志表示发生由 RTC 闹钟中断信号引起的解除请求。

[为 “1” 的条件]

- 当发生由 RTC 闹钟中断信号引起的解除请求时

[为 “0” 的条件]

- 在读“1”后写“0”时

DNMIF (NMI 深度待机解除标志)

此标志表示发生由 NMI 引脚引起的解除请求。

[为 “1” 的条件]

- 当发生由 DPSIEGR2 寄存器所设 NMI 引脚引起的解除请求时

[为 “0” 的条件]

- 在读“1”后写“0”时

DRIICDIF (SDA-DS 深度待机解除标志)

此标志表示发生由 SDA-DS 中断信号引起的解除请求。

[为 “1” 的条件]

- 当发生由 DPSIEGR2 寄存器所选 SDA-DS 引脚引起的解除请求时

[为 “0” 的条件]

- 在读“1”后写“0”时

DRIICCIF (SCL-DS 深度待机解除标志)

此标志表示发生由 SCL-DS 中断信号引起的解除请求。

[为 “1” 的条件]

- 当发生由 DPSIEGR2 寄存器所选 SCL-DS 引脚引起的解除请求时

[为 “0” 的条件]

- 在读“1”后写“0”时

11.2.16 深度待机中断边沿寄存器 0 (DPSIEGR0)

地址 0008 C28Ah

	b7	b6	b5	b4	b3	b2	b1	b0
	DIRQ7 EG	DIRQ6 EG	DIRQ5 EG	DIRQ4 EG	DIRQ3 EG	DIRQ2 EG	DIRQ1 EG	DIRQ0 EG
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	DIRQ0EG	IRQ0-DS 引脚边沿选择位	0: 在下降沿产生解除请求 1: 在上升沿产生解除请求	R/W
b1	DIRQ1EG	IRQ1-DS 引脚边沿选择位	0: 在下降沿产生解除请求 1: 在上升沿产生解除请求	R/W
b2	DIRQ2EG	IRQ2-DS 引脚边沿选择位	0: 在下降沿产生解除请求 1: 在上升沿产生解除请求	R/W
b3	DIRQ3EG	IRQ3-DS 引脚边沿选择位	0: 在下降沿产生解除请求 1: 在上升沿产生解除请求	R/W
b4	DIRQ4EG	IRQ4-DS 引脚边沿选择位	0: 在下降沿产生解除请求 1: 在上升沿产生解除请求	R/W
b5	DIRQ5EG	IRQ5-DS 引脚边沿选择位	0: 在下降沿产生解除请求 1: 在上升沿产生解除请求	R/W
b6	DIRQ6EG	IRQ6-DS 引脚边沿选择位	0: 在下降沿产生解除请求 1: 在上升沿产生解除请求	R/W
b7	DIRQ7EG	IRQ7-DS 引脚边沿选择位	0: 在下降沿产生解除请求 1: 在上升沿产生解除请求	R/W

作为深度软件待机模式解除源的内部复位信号不对 DPSIEGR0 寄存器进行初始化，详细内容请参照“表 6.2 各种复位的初始化对象”。

11.2.17 深度待机中断边沿寄存器 2 (DPSIEGR2)

地址 0008 C28Ch

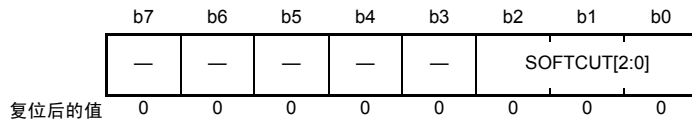
	b7	b6	b5	b4	b3	b2	b1	b0
	—	DRIICC EG	DRIICD EG	DNMIEG	—	—	DLVD2 EG	DLVD1 EG
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	DLVD1EG	LVD1 边沿选择位	0: 在检测到 $VCC < V_{det1}$ (下降) 时产生解除请求 1: 在检测到 $VCC \geq V_{det1}$ (上升) 时产生解除请求	R/W
b1	DLVD2EG	LVD2 边沿选择位	0: 在检测到 $VCC < V_{det2}$ (下降) 时产生解除请求 1: 在检测到 $VCC \geq V_{det2}$ (上升) 时产生解除请求	R/W
b3-b2	—	保留位	读写值都为“0”。	R/W
b4	DNMIEG	NMI 边沿选择位	0: 在下降沿产生解除请求 1: 在上升沿产生解除请求	R/W
b5	DRIICDEG	SDA-DS 边沿选择位	0: 在下降沿产生解除请求 1: 在上升沿产生解除请求	R/W
b6	DRIICCEG	SCL-DS 边沿选择位	0: 在下降沿产生解除请求 1: 在上升沿产生解除请求	R/W
b7	—	保留位	读写值都为“0”。	R/W

作为深度软件待机模式解除源的内部复位信号不对 DPSIEGR2 寄存器进行初始化，详细内容请参照“表 6.2 各种复位的初始化对象”。

11.2.18 闪存 HOCO 软件待机控制寄存器 (FHSSBYCR)

地址 0008 C28Fh



位	符号	位名	功能	R/W
b2-b0	SOFTCUT [2:0]	软切断位	b2 b0 0 0 0: 在软件待机模式中, 不切断向闪存和 HOCO 供电。而且, 电压检测电路运行并且 POR 低功耗功能无效。 0 1 0: 在软件待机模式中, 不切断向闪存供电, 但是切断向 HOCO 供电。而且, 电压检测电路运行并且 POR 低功耗功能无效。 0 1 1: 在软件待机模式中, 切断向闪存和 HOCO 供电。而且, 电压检测电路运行并且 POR 低功耗功能无效。 1 0 0: 在软件待机模式中, 不切断向闪存和 HOCO 供电。而且, 电压检测电路停止运行并且 POR 低功耗功能有效。 1 1 0: 在软件待机模式中, 不切断向闪存供电, 但是切断向 HOCO 供电。而且, 电压检测电路停止运行并且 POR 低功耗功能有效。 1 1 1: 在软件待机模式中, 切断向闪存和 HOCO 供电。而且, 电压检测电路停止运行并且 POR 低功耗功能有效。 上述以外: 不能设定	R/W
b7-b3	—	保留位	读写值都为“0”。	R/W

FHSSBYCR 寄存器在软件待机模式中控制向闪存和 HOCO 提供内部电源, 并且控制软件待机模式中的 LVD 和上电复位电路的状态。

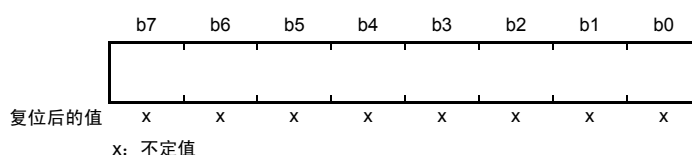
在软件待机模式中使用 LVD 时, 必须将 SOFTCUT[2] 位置“0”。

为了实现低功耗而停止 LVD 的运行, 并且将上电复位电路的低功耗功能置为有效时, 必须将 SOFTCUT[2] 位置“1”。

作为深度软件待机模式解除源的内部复位信号不对 FHSSBYCR 寄存器进行初始化, 详细内容请参照“表 6.2 各种复位的初始化对象”。

11.2.19 深度待机备用寄存器 (DPSBKRY) (y=0 ~ 31)

地址 0008 C2A0h ~ 0008 C2BFh



DPSBKRY 寄存器是用于在深度软件待机模式中保存数据的 32 字节可读写寄存器。

不对 DPSBKRY 寄存器进行初始化。在刚接通电源后, 寄存器的值为不定值。

11.3 通过转换时钟来降低功耗

如果设定 SCKCR.FCK[3:0] 位、SCKCR.ICK[3:0] 位、SCKCR.BCK[3:0] 位、SCKCR.PCKB[3:0] 位和 SCKCR.PCKD[3:0]，就转换时钟频率。CPU、DMAC、DTC、ROM 和 RAM 以 ICK[3:0] 位设定的运行时钟运行。

外围模块以 PCKB[3:0] 位和 PCKD[3:0] 位设定的运行时钟运行。

闪存接口以 FCK[3:0] 位设定的运行时钟运行。

外部总线时钟以 BCK[3:0] 位设定的运行时钟运行。详细内容请参照“9. 时钟发生电路”。

11.4 模块停止功能

能以内部外围模块为单位设定模块停止功能。

如果将 MSTPCRA ~ MSTPCRC 寄存器对应的 MSTPmi 位 (m=A ~ C, i=31 ~ 0) 置“1”，模块就停止运行并且转移到模块停止状态。此时，CPU 继续单独运行。通过将对应的 MSTPmi 位置“0”，解除模块停止状态并且在总线周期结束时模块重新开始运行。在模块停止状态下，保持模块的内部状态。

在解除复位后，除 DMAC、DTC 和内部 RAM 以外的全部模块都处于模块停止状态。

不能读写被设定为模块停止状态的模块的寄存器。但是必须注意：如果在设定为模块停止后立即进行写操作，也许能写。

11.5 低运行功率功能

根据工作频率和工作电压选择运行功率控制模式，能降低正常运行模式、睡眠模式和全模块时钟停止模式中的功耗。

11.5.1 运行功率控制模式的设定方法

运行功率控制模式的转移步骤例子如下所示。

(1) 从大功耗模式向小功耗模式转移的情况

例：从高速运行模式转移到低速运行模式 1

(在转移前的运行功率控制模式中进行高速运行。)

↓

从 HOCO 时钟转换到 LOCO 时钟的设定 (时钟源、分频比)

↓

写 OPCCR 寄存器

↓

(在转移后的运行功率控制模式中进行低速运行。)

(2) 从小功耗模式向大功耗模式转移的情况

例：从低速运行模式 2 转移到高速运行模式

(在转移前的运行功率控制模式中进行低速运行。)

↓

写 OPCCR 寄存器

↓

从 LOCO 时钟转换到 HOCO 时钟的设定 (时钟源、分频比)

↓

(在转移后的运行功率控制模式中进行高速运行。)

11.6 低功耗状态

11.6.1 睡眠模式

11.6.1.1 向睡眠模式的转移

如果在 SBYCR.SSBY 位为“0”的状态下执行 WAIT 指令，就进入睡眠模式。CPU 在睡眠模式中停止运行，但是保持 CPU 内部寄存器的值。CPU 以外的外围功能不停止运行。

如果在使用 WDT 时转移到睡眠模式，WDT 就停止计数。

在自动启动模式中使用 IWDT 的情况下，如果在 OFS0.IWDTSLCSTP 位为“1”时转移到睡眠模式，IWDT 就停止计数。同样，在寄存器启动模式中使用 IWDT 的情况下，如果在 IWDTCSSTPR.SLCSTP 位为“1”时转移到睡眠模式，IWDT 就停止计数。

在自动启动模式中使用 IWDT 的情况下，即使在 OFS0.IWDTSLCSTP 位为“0”（在转移到低功耗模式时，IWDT 继续进行计数）时转移到睡眠模式，IWDT 也继续进行计数。同样，在寄存器启动模式中使用 IWDT 的情况下，即使在 IWDTCSSTPR.SLCSTP 位为“0”时转移到睡眠模式，IWDT 也继续进行计数。

要使用睡眠模式时，必须在进行以下设定后执行 WAIT 指令：

1. 将CPU的PSW.I位（注1）置“0”。
2. 将用于从睡眠模式返回的中断传达目标设定为CPU。
3. 将用于从睡眠模式返回的中断优先级（注2）设定为高于CPU的PSW.IPL[3:0]位（注1）的优先级。
4. 将用于从睡眠模式返回的中断的IERm.IENj位（注2）置“1”。
5. 读最后写的I/O寄存器并且确认写入值已被反映。
6. 执行WAIT指令（通过执行WAIT指令，CPU的PSW.I位（注1）自动变为“1”）。

注 1. 详细内容请参照“2. CPU”。

注 2. 详细内容请参照“14. 中断控制器（ICUb）”。

11.6.1.2 睡眠模式的解除

通过全部中断、RES# 引脚复位、上电复位、电压监视复位或者 IWDT 下溢复位来解除睡眠模式。

- 通过中断进行的解除
如果发生中断，就解除睡眠模式并且开始中断异常处理。如果可屏蔽中断被CPU屏蔽（中断优先级（注1）小于等于CPU的PSW.IPL[3:0]位（注2）的优先级），就不解除睡眠模式。
- 通过RES#引脚进行的解除
如果将RES#引脚置为Low电平，就进入复位状态。如果在经过规定的复位输入期间后将RES#引脚置为High电平，CPU就开始复位异常处理。
- 通过上电复位进行的解除
通过上电复位来解除睡眠模式。
- 通过电压监视复位进行的解除
通过电压检测电路的电压监视复位来解除睡眠模式。
- 通过独立看门狗定时器复位进行的解除
通过IWDT下溢的内部复位来解除睡眠模式。但是在睡眠模式中，因为在IWDT停止计数的条件（OFS0.IWDTSTRT=0 并且 OFS0.IWDTSLCSTP=1 或者 OFS0.IWDTSTRT=1 并且 IWDTCSSTPR.SLCSTP=1）下IWDT停止运行，所以不能通过独立看门狗定时器复位进行解除。

注 1. 详细内容请参照“14. 中断控制器（ICUb）”。

注 2. 详细内容请参照“2. CPU”。

11.6.1.3 睡眠模式返回时钟源的转换功能

为了转换睡眠模式返回时钟源，需要通过睡眠模式返回时钟源转换寄存器（RSTCKCR）设定返回后的时钟以及设定各时钟的等待控制寄存器。如果发生返回中断，就在等待作为返回时钟而设定的振荡器振荡稳定后，自动转换时钟源并且从睡眠模式返回。此时，自动改写时钟源转换的相关寄存器。

详细内容请参照“11.2.6 睡眠模式返回时钟源转换寄存器（RSTCKCR）”。有关振荡稳定等待时间的设定，请参照“11.2.7 主时钟振荡器的等待控制寄存器（MOSCWTCR）”、“11.2.8 副时钟振荡器的等待控制寄存器（SOSCWTCR）”和“11.2.10 HOCO 等待控制寄存器 2（HOCOWTCR2）”。

另外，如果在 SBYCR.SBY 位为“1”时转移到睡眠模式（振荡停止检测功能有效时），本功能（睡眠返回时钟源转换功能）就无效。

11.6.2 全模块时钟停止模式

11.6.2.1 向全模块时钟停止模式的转移

在将 MSTPCRA.ACSE 位置“1”并且将由 MSTPCRA、MSTPCRB、MSTPCRC 寄存器控制的模块置为模块停止状态（MSTPCRA=FFFF FF[C-F]Fh（注1）、MSTPCRB=FFFF FFFFh、MSTPCRC[31:16]=FFFFh）的情况下，如果在将 SBYCR.SBY 位置“0”的状态下执行 WAIT 指令，总线控制器、I/O 端口以及除 8 位定时器（注2）、IWDT、RTC、上电复位电路、电压检测电路以外的全部模块就在总线周期结束时停止运行，转移到全模块时钟停止模式（注3）。

如果在使用 WDT 时转移到全模块时钟停止模式，WDT 就停止计数。

在自动启动模式中使用 IWDT 的情况下，如果在 OFS0.IWDTSLCSTP 位为“1”时转移到全模块时钟停止模式，IWDT 就停止计数。同样，在寄存器启动模式中使用 IWDT 的情况下，如果在 IWDTCSLTPR.SLCSTP 位为“1”时转移到全模块时钟停止模式，IWDT 就停止计数。

在自动启动模式中使用 IWDT 的情况下，即使在 OFS0.IWDTSLCSTP 位为“0”（在转移到低功耗模式时，IWDT 继续进行计数）时转移到全模块时钟停止模式，IWDT 也继续进行计数。同样，在寄存器启动模式中使用 IWDT 的情况下，即使在 IWDTCSLTPR.SLCSTP 位为“0”时转移到全模块时钟停止模式，IWDT 也继续进行计数。

要使用全模块时钟停止模式时，必须在进行以下设定后执行 WAIT 指令：

1. 将 CPU 的 PSW.I 位（注4）置“0”。
2. 将用于从全模块时钟停止模式返回的中断传达目标设定为 CPU。
3. 将用于从全模块时钟停止模式返回的中断优先级（注5）设定为高于 CPU 的 PSW.IPL[3:0] 位（注4）的优先级。
4. 将用于从全模块时钟停止模式返回的中断的 IERm.IENj 位（注4）置“1”。
5. 读最后写的 I/O 寄存器并且确认写入值已被反映。
6. 执行 WAIT 指令（通过执行 WAIT 指令，CPU 的 PSW.I 位（注4）自动变为“1”）。

注 1. 尽管 MSTPCRA.MSTPA15 位和 MSTPCRA.MSTPA14 位在运行状态下也转移到全模块时钟停止模式，但是必须在停止状态下进行转移。

注 2. 能通过 MSTPCRA.MSTPA5 位和 MSTPCRA.MSTPA4 位选择运行或者停止。

注 3. 有可能因 DTC 和 DMAC 的运行状态而无法转移到全模块时钟停止模式。因此必须在将 MSTPCRA.MSTPA28 位置“1”前，将 DMAC 的 DMAST.DMST 位和 DTC 的 DTCST.DTCST 位置“0”，并且在未启动 DTC 和 DMAC 的状态下进行转移。

注 4. 详细内容请参照“2. CPU”。

注 5. 详细内容请参照“14. 中断控制器（ICUb）”。

11.6.2.2 全模块时钟停止模式的解除

通过外部引脚中断（NMI、IRQ0～IRQ7）、外围功能中断（8位定时器（注1）、RTC 闹钟、RTC 周期、IWDTC（注2）、电压监视 1、电压监视 2、振荡停止检测）、RES# 引脚复位、上电复位、电压监视复位或者独立看门狗定时器复位来解除全模块时钟停止模式，在经过异常处理后转移到正常的程序执行状态。如果可屏蔽中断被 CPU 屏蔽（中断优先级（注3）小于等于 CPU 的 PSW.IPL[3:0] 位（注4）的优先级）或者被设定为 DTC 和 DMAC 的启动源，就不解除全模块时钟停止模式。

注 1. 能通过 MSTPCRA.MSTPA5 位和 MSTPCRA.MSTPA4 位选择运行或者停止。

注 2. 在全模块时钟停止模式中，因为在独立看门狗定时器停止计数的条件（OFS0.IWDTCSTRT=0 并且 OFS0.IWDTCSLCSTP =1 或者 OFS0.IWDTCSTRT=1 并且 IWDTCSTPR.SLCSTP=1）下独立看门狗定时器停止运行，所以不能通过独立看门狗定时器复位进行解除。

注 3. 详细内容请参照“14. 中断控制器（ICUb）”。

注 4. 详细内容请参照“2. CPU”。

11.6.3 软件待机模式

11.6.3.1 向软件待机模式的转移

如果在将 SBYCR.SSBY 位置“1”并且将 DPSBYCR.DPSBY 位置“0”的状态下执行 WAIT 指令，就转移到软件待机模式。在此模式中，CPU、内部外围功能和振荡器的全部功能停止运行，但是保持 CPU 内部寄存器的值、内部 RAM 的数据、内部外围功能状态和 I/O 端口状态。能通过 SBYCR.OPE 位选择是将地址总线和总线控制信号设定为高阻抗状态还是保持输出状态。因为在软件待机模式中振荡器停止振荡，所以功耗明显降低。另外，能通过设定 FHSSBYCR 寄存器进一步降低消耗电流。在软件待机模式中，如果 FHSSBYCR.SOFTCUT0 位被置“1”，就停止向内部 ROM 提供内部电源；如果 FHSSBYCR.SOFTCUT1 位被置“1”，就停止向高速内部振荡器提供电源；如果 FHSSBYCR.SOFTCUT2 位被置“1”，就停止电压检测电路（LVD）的运行，同时上电复位电路的低功耗功能变为有效。此时上电复位电路的电压检测特性发生变化，详细内容请参照“41. 电特性”。

必须在执行 WAIT 指令前将 DMAC 的 DMAST.DMST 位和 DTC 的 DTCST.DTCST 位置“0”。

如果在使用 WDT 时转移到软件待机模式，因为振荡器停止振荡，所以 WDT 就停止计数。

在自动启动模式中使用 IWDTC 的情况下，如果在 OFS0.IWDTCSLCSTP 位为“1”时转移到软件待机模式，IWDTC 就停止计数。同样，在寄存器启动模式中使用 IWDTC 的情况下，如果在 IWDTCSTPR.SLCSTP 位为“1”时转移到软件待机模式，IWDTC 就停止计数。

在自动启动模式中使用 IWDTC 的情况下，即使在 OFS0.IWDTCSLCSTP 位为“0”（在转移到低功耗模式时，IWDTC 继续进行计数）时转移到软件待机模式，IWDTC 也继续进行计数。同样，在寄存器启动模式中使用 IWDTC 的情况下，即使在 IWDTCSTPR.SLCSTP 位为“0”时转移到软件待机模式，IWDTC 也继续进行计数。

当振荡停止检测功能有效（OSTDCR.OSTDE=1）时，不能转移到软件待机模式。如果要转移到软件待机模式，就必须在将振荡停止检测功能设定为无效（OSTDCR.OSTDE=0）后执行 WAIT 指令。

要使用软件待机模式时，必须在进行以下设定后执行 WAIT 指令：

1. 将 CPU 的 PSW.I 位（注1）置“0”。
2. 将用于从软件待机模式返回的中断传达目标设定为 CPU。
3. 将用于从软件待机模式返回的中断优先级（注2）设定为高于 CPU 的 PSW.IPL[3:0] 位（注1）的优先级。
4. 将用于从软件待机模式返回的中断的 IERm.IENj 位（注2）置“1”。
5. 读最后写的 I/O 寄存器并且确认写入值已被反映。
6. 执行 WAIT 指令（通过执行 WAIT 指令，CPU 的 PSW.I 位（注1）自动变为“1”）。

注 2. 详细内容请参照“2. CPU”。

注 3. 详细内容请参照“14. 中断控制器（ICUb）”。

11.6.3.2 软件待机模式的解除

通过外部引脚中断（NMI、IRQ0 ~ IRQ7）、外围功能中断（RTC 闹钟、RTC 周期、IWDT、电压监视 1、电压监视 2）、RES# 引脚复位、上电复位、电压监视复位或者独立看门狗定时器复位来解除软件待机模式。如果产生软件待机模式解除源，转移到软件待机模式前运行的各振荡器就重新开始运行。在等待这些全部振荡器振荡稳定后，从软件待机模式返回。

- 通过中断进行的解除
如果产生 NMI、IRQ0 ~ IRQ7、RTC 闹钟、RTC 周期、IWDT、电压监视 1 和电压监视 2 的中断请求，转移到软件待机模式前运行的各振荡器就重新开始运行。在各振荡器经过 MOSCWTCR.MSTS[4:0] 位、SOSCWTCR.SSTS[4:0] 位和 PLLWTCR.PSTS[4:0] 位设定的时间后，给整个 LSI 提供稳定的时钟，解除软件待机模式并且开始中断异常处理。
- 通过 RES# 引脚进行的解除
如果将 RES# 引脚置为 Low 电平，时钟就开始振荡，并且在时钟开始振荡的同时给 LSI 提供时钟。此时，必须将 RES# 引脚的 Low 电平保持到时钟振荡稳定为止。如果将 RES# 引脚置为 High 电平，CPU 就开始复位异常处理。
- 通过上电复位进行的解除
如果因电源电压的降低而发生上电复位，就解除软件待机模式。
- 通过电压监视复位进行的解除
如果因电源电压的降低而发生电压监视复位，就解除软件待机模式。
- 通过独立看门狗定时器复位进行的解除
通过 IWDT 下溢的内部复位来解除软件待机模式。
但是在软件待机模式中，因为在独立看门狗定时器停止计数的条件（OFS0.IWDTSTRT=0 并且 OFS0.IWDTSLCSTP=1 或者 OFS0.IWDTSTRT=1 并且 IWDTCSTPR.SLCSTP=1）下独立看门狗定时器停止运行，所以不能通过独立看门狗定时器复位进行解除。

11.6.3.3 软件待机模式的应用例子

在 IRQn 引脚的下降沿转移到软件待机模式并且在 IRQn 引脚的上升沿解除软件待机模式的例子如图 11.6 所示。

此例中，在 ICU 的 IRQCRi.IRQMD[1:0] 位为“01b”（下降沿）的状态下接受 IRQn 中断后，将 IRQCRi.IRQMD[1:0] 位置“10b”（上升沿）并且将 SBYCR.SSBY 位置“1”，然后执行 WAIT 指令，转移到软件待机模式。此后，在 IRQn 引脚的上升沿解除软件待机模式。

要从软件待机模式返回时，还需要设定中断控制器（ICU）。详细内容请参照“14. 中断控制器（ICU）”。

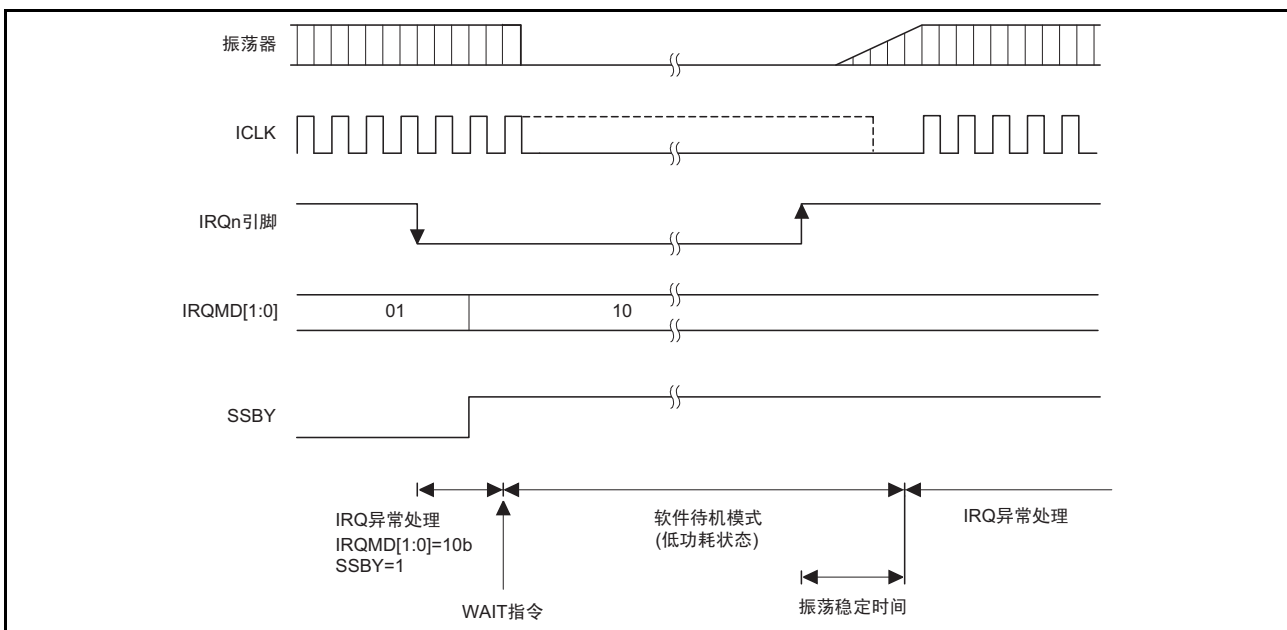


图 11.6 软件待机模式的应用例子

11.6.4 深度软件待机模式

11.6.4.1 向深度软件待机模式的转移

如果在 SBYCR.SSBY 位为“1”的状态下执行 WAIT 指令，就转移到软件待机模式（注1）。此时，如果 DPSBYCR.DPSBY 位为“1”，就转移到深度软件待机模式。

因为在深度软件待机模式中 CPU、内部外围功能（RTC 闹钟、RTC 周期、SCL-DS、SDA-DS 除外）和振荡器的全部功能停止运行并且停止给它们提供内部电源，所以功耗明显降低。此时，CPU、内部外围功能（RTC 闹钟、RTC 周期、SCL-DS、SDA-DS 除外）的寄存器内容和内部 RAM 的数据全部为不定值。

在将 DEEPCUT1 位置“1”的情况下，因为停止电压检测电路的运行并且上电复位电路的低功耗功能有效，所以能进一步降低功耗。此时上电复位电路的电压检测特性发生变化，详细内容请参照“41. 电特性”。

如果在使用 WDT 时转移到深度软件待机模式，振荡器就停止振荡，停止给 WDT 供电并且停止计数。

在自动启动模式中使用 IWDT 的情况下，如果在 OFS0.IWDTSLCSTP 位为“1”时转移到深度软件待机模式，就停止给 IWDT 专用 LOCO 和 IWDT 供电并且停止计数。同样，在寄存器启动模式中使用 IWDT 的情况下，如果在 IWDTCSSTPR.SLCSTP 位为“1”时转移到深度软件待机模式，就停止给 IWDT 专用 LOCO 和 IWDT 供电并且停止计数。

在自动启动模式中使用 IWDT 的情况下，当 OFS0.IWDTSLCSTP 位为“0”（在转移到低功耗模式时，IWDT 继续进行计数）时不转移到深度软件待机模式而转移到软件待机模式，并且继续进行计数。同样，在寄存器启动模式中使用 IWDT 的情况下，当 IWDTCSSTPR.SLCSTP 位为“0”时不转移到深度软件待机模式而转移到软件待机模式，并且继续进行计数。

如果在电压检测电路中选择电压监视 1 复位功能（LVD1CR0.LVD1RI=1）或者电压监视 2 复位功能（LVD2CR0.LVD2RI=1），就不能转移到深度软件待机模式而转移到软件待机模式。

端口保持软件待机模式中的状态。

注 1. 在执行 WAIT 指令前，需要满足转移到软件待机模式时的 DTC、DMAC、IWDT 的相关条件。详细内容请参照“11.6.3 软件待机模式”。

11.6.4.2 深度软件待机模式的解除

通过作为外部引脚中断发生源的部分引脚（NMI、IRQ0-DS ~ IRQ7-DS、SCL-DS、SDA-DS）、外围功能中断（RTC 闹钟、RTC 周期、电压监视 1、电压监视 2）、RES# 引脚复位、上电复位或者电压监视 0 复位来解除深度软件待机模式。

1. 通过外部中断引脚和内部中断信号进行的解除

通过 DPSIERn（n=0、2）寄存器和 DPSIFRn（n=0、2）寄存器控制深度软件待机模式的解除。如果产生能解除深度软件待机模式的中断请求，DPSIFRn 寄存器的对应标志就变为“1”。此时，在 DPSIERn 寄存器允许解除源的情况下解除深度软件待机模式。能通过 DPSIEGRn（n=0、2）选择上升沿或者下降沿。可选择边沿的中断为 NMI、IRQ0-DS ~ IRQ7-DS、SCL-DS、SDA-DS、电压监视 1 和电压监视 2。如果产生深度软件待机模式的解除请求，就提供内部电源，LOCO 时钟开始振荡，并且对整个 LSI 进行内部复位（深度软件待机复位）。此后，给整个 LSI 提供稳定的 LOCO 时钟并且解除深度软件待机复位。如果在解除深度软件待机复位的同时解除深度软件待机模式，就开始复位异常处理。

如果通过外部中断引脚和内部中断信号解除深度软件待机模式，RSTSR0.DPSRSTF 标志就变为“1”。

2. 通过 RES# 引脚进行的解除

如果将 RES# 引脚置为 Low 电平，就解除深度软件待机模式。此时，必须根据“41. 电特性”，将 RES# 引脚保持为 Low 电平。如果将 RES# 引脚置 High 电平，CPU 就开始复位异常处理。

3. 通过上电复位进行的解除

如果因电源电压的降低而发生上电复位，就解除深度软件待机模式。

4. 通过电压监视 0 复位进行的解除

如果因电源电压的降低而发生电压监视 0 复位，就解除深度软件待机模式。

11.6.4.3 解除深度软件待机模式时的引脚状态

在深度软件待机模式中，I/O 端口保持软件待机模式中的状态。通过深度软件待机模式的内部复位对 LSI 内部进行初始化，如果解除深度软件待机模式，就开始复位异常处理。此时的端口状态如下所示。

能通过 DPSBYCR.IOKEEP 位选择是将 I/O 端口设定为初始状态还是保持软件待机模式中的 I/O 端口状态。

- DPSBYCR.IOKEEP 位为“0”的情况

通过解除深度软件待机模式的内部复位，使 I/O 端口变为初始状态。

- DPSBYCR.IOKEEP 位为“1”的情况

通过解除深度软件待机模式的内部复位对 LSI 内部进行初始化，I/O 端口与 LSI 的内部状态无关，保持软件待机模式中的状态。此时，即使设定 I/O 端口、外围模块，也保持软件待机模式中的 I/O 端口状态。此后，通过将 IOKEEP 位置“0”，解除 I/O 端口的状态保持，并且根据内部状态运行。

解除深度软件待机模式的内部复位不对 DPSBYCR.IOKEEP 位进行初始化。

11.6.4.4 深度软件待机模式的应用例子

在 IRQn-DS 引脚的下降沿转移到深度软件待机模式并且在 IRQn-DS 引脚的上升沿解除深度软件待机模式的运行例子如图 11.7 所示。

此例中，在 ICU 的 IRQCRi.IRQMD[1:0] 位为“01b”（下降沿）的状态下接受 IRQn 中断后，将 DPSIEGR0.DIRQnEG 位（n=0～7）置“1”（上升沿）并且将 SBYCR.SSBY 位和 DPSBYCR.DPSBY 位置“1”，然后执行 WAIT 指令，转移到深度软件待机模式。

此后，在 IRQn-DS 引脚的上升沿解除深度软件待机模式。

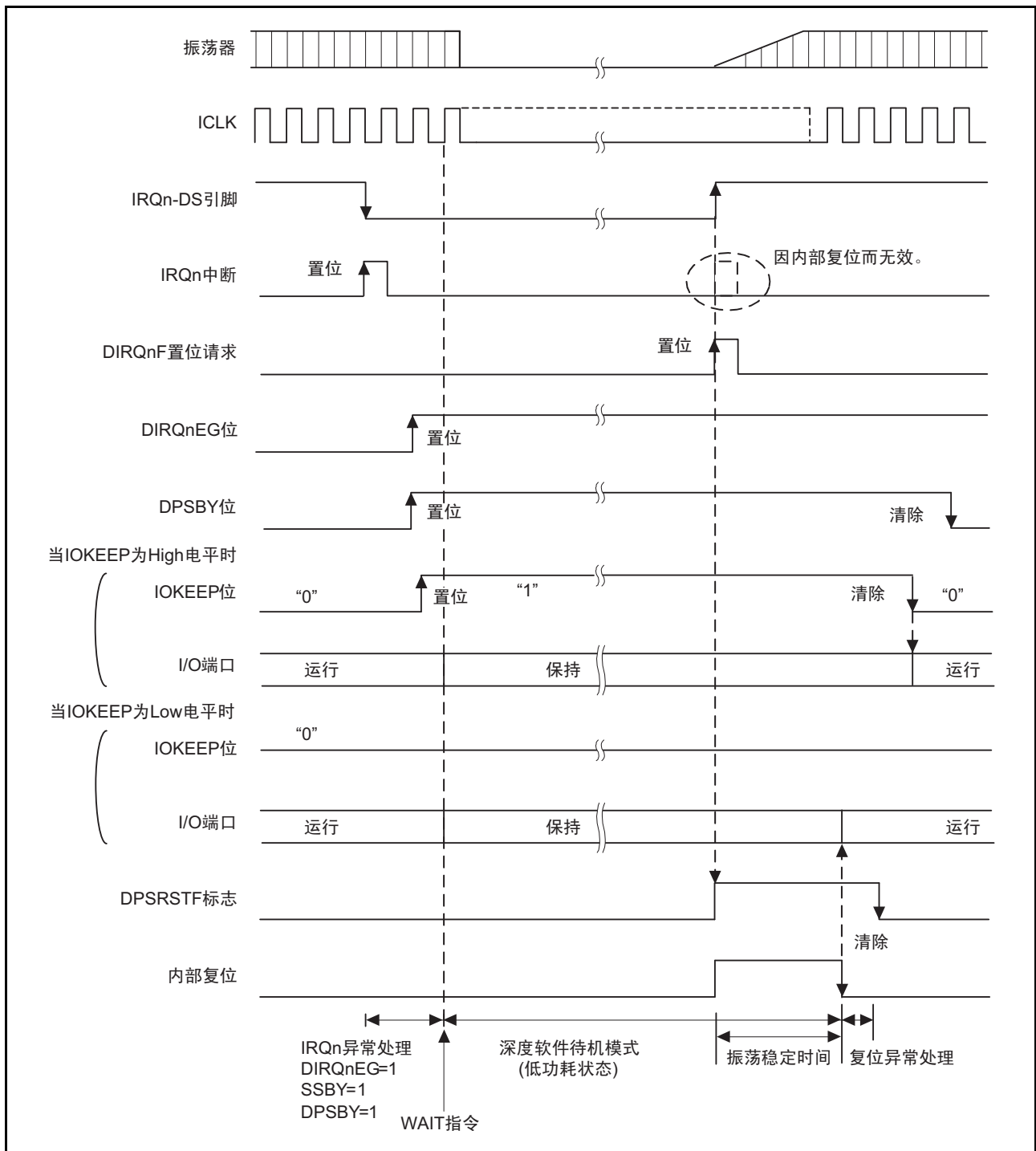


图 11.7 深度软件待机模式的应用例子

11.6.4.5 深度软件待机模式的流程图

使用深度软件待机模式时的流程图例子如图 11.8 所示。

此例中，在进行复位异常处理后，根据复位功能的 RSTSR0.DPSRSTF 标志判断是通过 RES# 引脚进行的复位还是通过解除深度软件待机模式进行的复位。

当通过 RES# 引脚进行复位时，在进行各种设定后转移到深度软件待机模式。

当通过解除深度软件待机模式进行复位时，在设定 I/O 端口后将 DPSBYCR.IOKEEP 位置“0”。

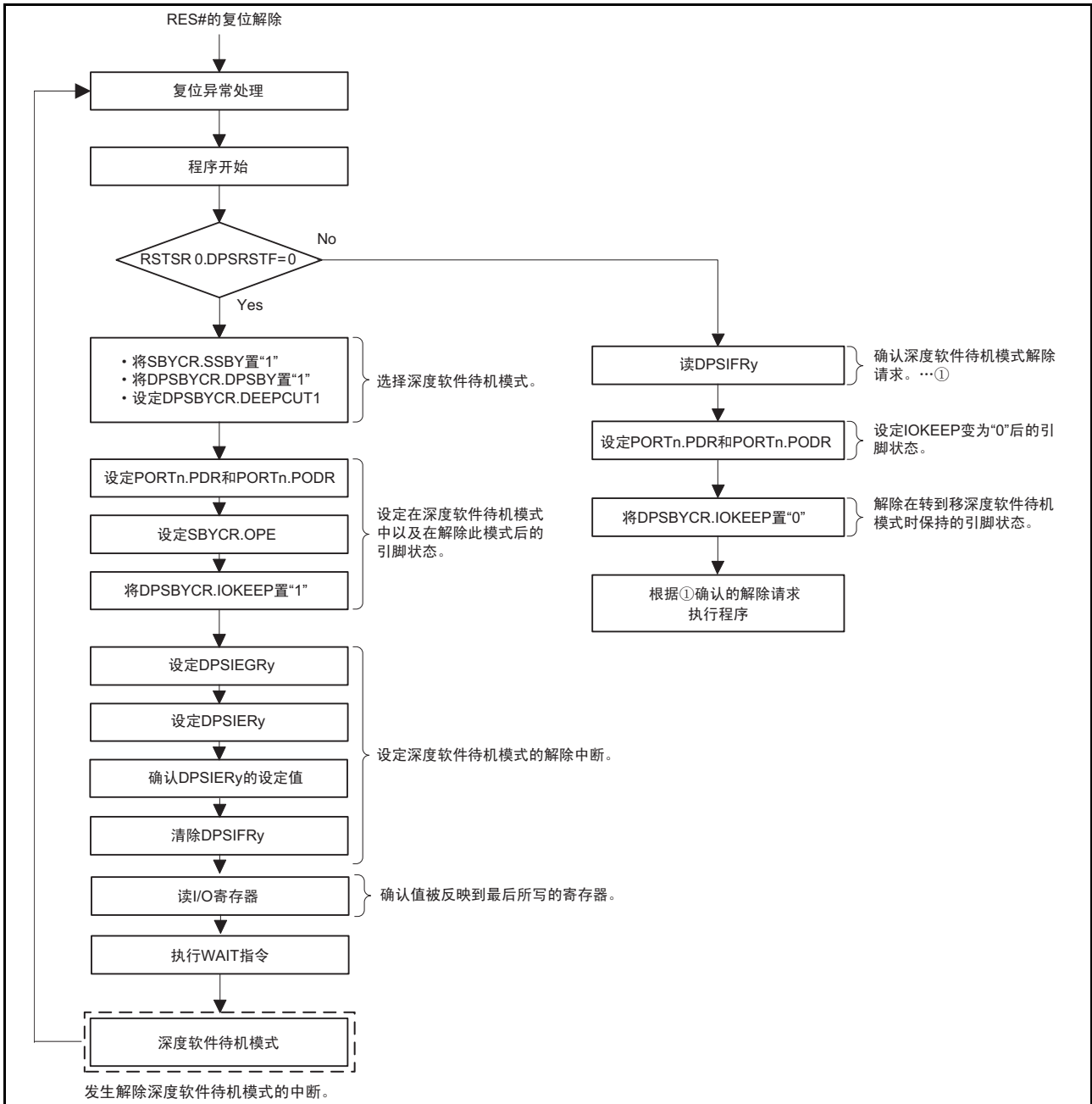


图 11.8 深度软件待机模式的流程图例子

11.7 使用时的注意事项

11.7.1 I/O 端口状态

在软件待机模式和深度软件待机模式中保持 I/O 端口状态。如果输入 High 电平，就无法降低输出电流的消耗电流。

11.7.2 DMAC 和 DTC 的模块停止

必须在将 MSTPCRA.MSTPA28 位置“1”前将 DMAC 的 DMAST.DMST 位和 DTC 的 DTCST.DTCST 位置“0”，并且将 DTC 和 DMAC 设定为未启动的状态。

详细内容请参照“16. DMA 控制器 (DMACA)”和“17. 数据传送控制器 (DTCa)”。

11.7.3 内部外围模块的中断

此中断不能在模块停止状态下运行。如果在产生中断请求的状态下停止模块，就不能清除 CPU 的中断源或者 DMAC、DTC 的启动源。必须事先禁止中断，然后设定为模块停止状态。

11.7.4 MSTPCRA、MSTPCRB 和 MSTPCRC 寄存器的写操作

只能通过 CPU 写 MSTPCRA、MSTPCRB 和 MSTPCRC 寄存器。

11.7.5 通过 DIRQnE 位 (n=0 ~ 7) 控制输入缓冲器

能通过将 DPSIER0.DIRQnE (n=0 ~ 7) 位置“1”，将 IRQ0-DS ~ IRQ7-DS 引脚的输入缓冲器置为有效。因此必须注意：这些引脚的输入被传送到 DPSIFR0.DIRQnF (n=0 ~ 7) 位而不被传送到中断控制器、外围模块和 I/O 端口。

11.7.6 WAIT 指令的执行时序

如果不等到先执行的寄存器的写操作结束而执行 WAIT 指令，就可能在通过写寄存器进行的设定更改被反映前执行 WAIT 指令，导致意想不到的运行。因此必须在确认已写完最后的寄存器后执行 WAIT 指令。

11.7.7 睡眠模式中通过 DMAC 和 DTC 进行的寄存器改写

在睡眠模式中，WDT 停止运行。不能在睡眠模式中通过 DMAC 和 DTC 改写 WDT 的相关寄存器。

在睡眠模式中，通过设定 OFS0.IWDTSLCSTP 位和 IWDTCTPR.SLCSTP 位来停止 IWDT 的运行。此时，不能在睡眠模式中通过 DMAC 和 DTC 改写 IWDT 的相关寄存器。

RSTCKCR 寄存器是睡眠模式返回时钟源转换功能的相关寄存器。如果在睡眠模式中进行改写，就可能导致意想不到的运行，因此不能在睡眠模式中改写 RSTCKCR 寄存器。

11.7.8 全模块时钟停止模式的解除

如果 ICLK 慢于 PCLKB，就不能使用 TMR 中断来解除全模块时钟停止模式。要使用 TMR 中断来解除全模块时钟停止模式时，必须事先将 ICLK 更改为大于等于 PCLKB 的频率，然后转移到全模块时钟停止模式。

11.7.9 将副时钟用作系统时钟的时钟源时的注意事项

在将副时钟用作系统时钟的时钟源的情况下，必须在转移到软件待机模式时使 RTC (RCR3.RTCEN=1) 或者低速内部振荡器 (LOCOCR.LCSTP=0) 运行。

12. 寄存器写保护功能

寄存器写保护功能是在程序失控时保护重要的寄存器不被改写的功能。通过保护寄存器（PRCR）设定要保护的寄存器。

PRCR 寄存器和被保护的寄存器的对应如表 12.1 所示。

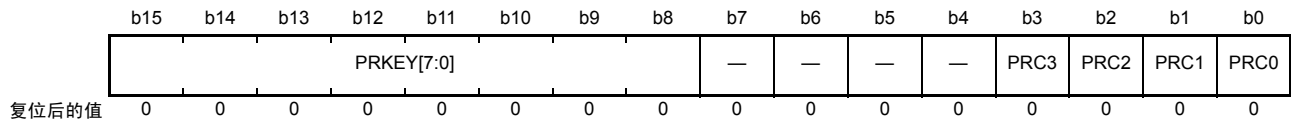
表 12.1 PRCR 寄存器和被保护的寄存器的对应

PRCR 寄存器	被保护的寄存器
PRC0 位	<ul style="list-style-type: none"> • 时钟发生电路的相关寄存器 SCKCR、SCKCR3、PLLCR、PLLCR2、BCKCR、MOSCCR、SOSCCR、LOCOCR、ILOOCR、HOCOOCR、OSTDCR、OSTDSR、HOCOOCR2
PRC1 位	<ul style="list-style-type: none"> • 运行模式的相关寄存器 SYSCR0、SYSCR1 • 低功耗功能的相关寄存器 SBYCR、MSTPCRA、MSTPCRB、MSTPCRC、OPCCR、RSTCKCR、MOSCWTCR、SOSCWTCR、PLLWTCR、DPSBYCR、DPSIER0、DPSIER2、DPSIFR0、DFSIFR2、DPSIEGR0、DPSIEGR2、FHSSBYCR、HOCOWTCR2 • 时钟发生电路的相关寄存器 MOFCR、HOCOPCR • 软件复位寄存器 SWRR
PRC2 位	<ul style="list-style-type: none"> • 地址 0008 0200h
PRC3 位	<ul style="list-style-type: none"> • LVD 的相关寄存器 LVCMPCR、LVDLVLRL、LVD1CR0、LVD1CR1、LVD1SR、LVD2CR0、LVD2CR1、LVD2SR

12.1 寄存器说明

12.1.1 保护寄存器 (PRCR)

地址 0008 03FEh



位	符号	位名	功能	R/W
b0	PRC0	保护位 0	允许写时钟发生电路的相关寄存器。 0: 禁止写 1: 允许写	R/W
b1	PRC1	保护位 1	允许写运行模式、低功耗功能和软件复位的相关寄存器。 0: 禁止写 1: 允许写	R/W
b2	PRC2	保护位 2	允许写地址 0008 0200h 0: 禁止写 1: 允许写	R/W
b3	PRC3	保护位 3	允许写 LVD 的相关寄存器。 0: 禁止写 1: 允许写	R/W
b7-b4	—	保留位	读写值都为“0”。	R/W
b15-b8	PRKEY[7:0]	PRC 键码位	控制是否能改写 PRC0 ~ PRC3 位。	R/W

PRCi 位 (保护位 i) (i=0 ~ 3)

这些位选择允许或者禁止写被保护的寄存器。

当 PRCi 位为“1”时，能写被保护的寄存器；当 PRCi 位为“0”时，不能写被保护的寄存器。

PRKEY[7:0] (PRC 键码位)

这些位选择允许或者禁止写 PRCR 寄存器。

要给 PRCi 位写值时，必须将 PRKEY[7:0] 位置“A5h”。如果 PRKEY[7:0] 位不是“A5h”，即使写 PRCR 寄存器，PRCi 位的值也不变。

13. 异常处理

13.1 异常事件

在 CPU 正常执行程序的过程中，有可能因某个事件的发生而中止正在执行的程序并且需要执行其他程序。此类事件统称为异常事件。

RX CPU 对应 6 种异常，异常事件的种类如图 13.1 所示。

如果发生异常，处理器模式就转移到管理模式。

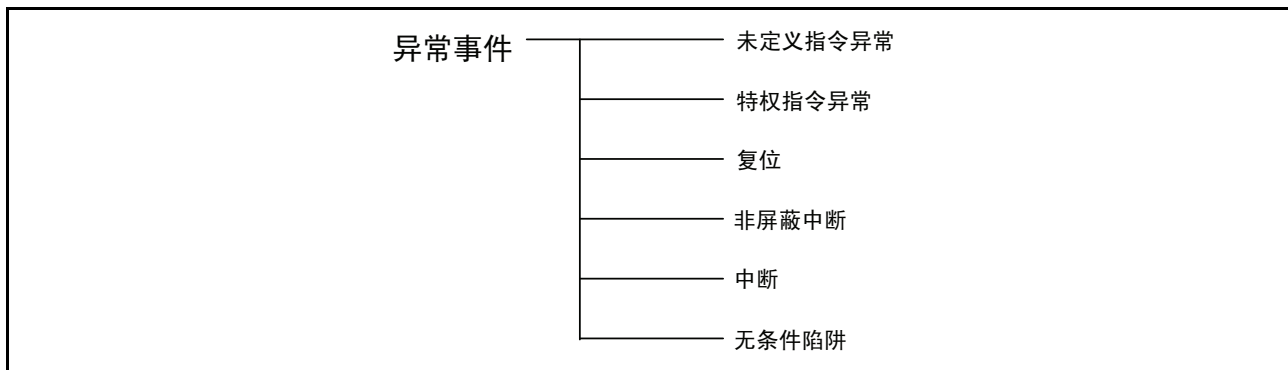


图 13.1 异常事件的种类

13.1.1 未定义指令异常

在检测到执行未定义指令（未安装的指令）时发生未定义指令异常。

13.1.2 特权指令异常

在用户模式中检测到已执行特权指令时发生特权指令异常。只能在管理模式中执行特权指令。

13.1.3 复位

在给 CPU 输入复位信号时发生复位。因为复位的优先级最高，所以随时被接受。

13.1.4 非屏蔽中断

在给 CPU 输入非屏蔽中断信号时发生非屏蔽中断。只在认为是对系统造成致命的故障时使用此中断。使用条件是必须在异常处理程序的处理后不返回到发生异常时正在执行的程序。

13.1.5 中断

在给 CPU 输入中断信号时发生中断。能将中断中的 1 个中断源分配为高速中断，高速中断的硬件预处理和硬件后处理比一般中断快，并且高速中断的优先级为 15（最高）。

当 PSW 的 I 位为“0”时，禁止接受中断。

13.1.6 无条件陷阱

如果执行 INT 指令和 BRK 指令，就产生无条件陷阱。

13.2 异常处理步骤

异常处理包括硬件自动处理的部分以及通过用户记述的程序（异常处理程序）进行处理的部分。除复位以外，接受异常时的处理步骤如图 13.2 所示。

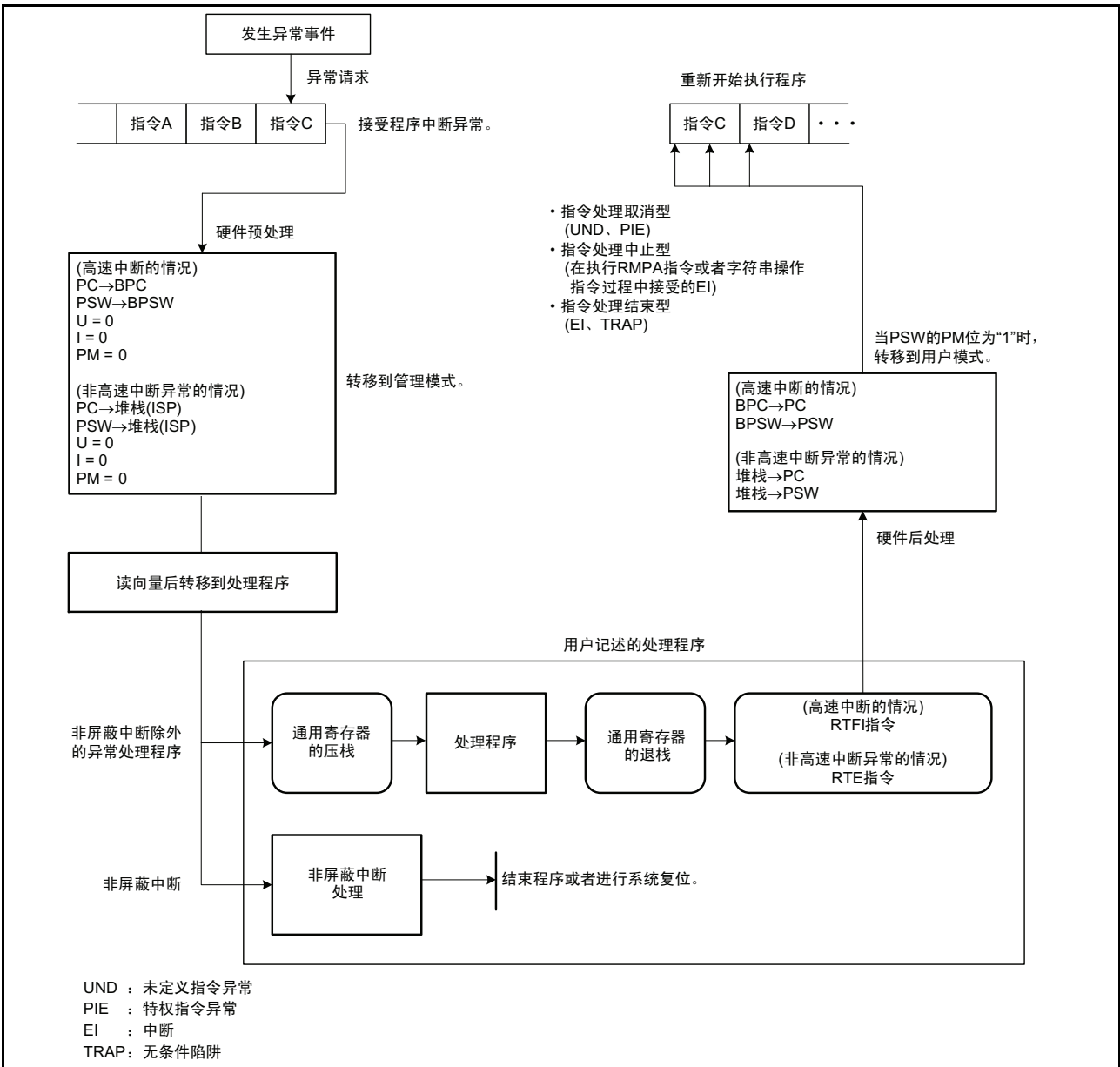


图 13.2 异常处理步骤的概要

一旦接受异常，RX CPU 就在硬件处理后，存取向量并且取得转移目标地址。按各异常给向量分配向量地址，并且写异常处理程序的转移目标地址。

有关 RX CPU 的硬件预处理，在高速中断的情况下，将程序计数器（PC）的内容保存到备用 PC（BPC），处理器状态字（PSW）的内容保存到备用 PSW（BPSW）；在非高速中断异常的情况下，将 PC 和 PSW 保存到堆栈区。对于异常处理程序中使用的通用寄存器以及 PC 和 PSW 以外的控制寄存器，必须在异常处理程序的起始位置，通过用户程序将这些寄存器压栈。

在异常处理程序处理结束后，通过在恢复被压栈的寄存器后执行 RTE 指令，从异常处理返回到原来的程序。只在高速中断的情况下执行 RTFI 指令。但是，在非屏蔽中断的情况下，不返回到原来的程序而必须结束程序或者进行系统复位。

有关 RX CPU 的硬件后处理，在高速中断的情况下，将 BPC 的值返回到 PC，BPSW 的值返回到 PSW。在非高速中断异常的情况下，从堆栈区返回 PC 和 PSW 的值。

13.3 异常事件的接受

如果发生异常事件，就在中止目前执行的程序后转移到异常处理程序。

13.3.1 接受时序和被保存的 PC 值

各异常事件的接受时序以及被保存的程序计数器（PC）的值如表 13.1 所示。

表 13.1 接受时序和被保存的 PC 值

异常事件		处理型	接受时序	被保存到 BPC/ 堆栈的 PC 值
未定义指令异常		指令处理取消型	正在执行指令	发生异常的指令的 PC 值
特权指令异常		指令处理取消型	正在执行指令	发生异常的指令的 PC 值
复位		指令处理放弃型	各周期	无
非屏蔽中断	正在执行 RMPA、SCMPU、SMOVB、SMOVF、SMOVU、SSTR、SUNTIL、SWHILE 各指令	指令处理中止型	正在执行指令	正在执行的指令的 PC 值
	上述以外的状态	指令处理结束型	在指令和指令之间	下一条指令的 PC 值
中断	正在执行 RMPA、SCMPU、SMOVB、SMOVF、SMOVU、SSTR、SUNTIL、SWHILE 各指令	指令处理中止型	正在执行指令	正在执行的指令的 PC 值
	上述以外的状态	指令处理结束型	在指令和指令之间	下一条指令的 PC 值
无条件陷阱		指令处理结束型	在指令和指令之间	下一条指令的 PC 值

13.3.2 向量和 PC、PSW 的保存场所

各异常事件的向量、程序计数器（PC）和处理器状态字（PSW）的保存场所如表 13.2 所示。

表 13.2 向量和 PC、PSW 的保存场所

异常事件		向量	PC 和 PSW 的保存场所
未定义指令异常		固定向量表	堆栈
特权指令异常		固定向量表	堆栈
复位		固定向量表	无
非屏蔽中断		固定向量表	堆栈
中断	高速中断	FINTV	BPC、BPSW
	非高速中断	可变向量表（INTB）	堆栈
无条件陷阱		可变向量表（INTB）	堆栈

13.4 接受异常 / 从异常返回时的硬件处理

以下说明接受异常以及从异常返回时的硬件处理（复位除外）。

(1) 接受异常时的硬件预处理

(a) PSW 的保存

（高速中断的情况）

PSW→BPSW

（非高速中断异常的情况）

PSW→堆栈区

(b) PSW 的 PM 位、U 位和 I 位的更新

I : 置“0”

U : 置“0”

PM : 置“0”

(c) PC 的保存

（高速中断的情况）

PC→BPC

（非高速中断异常的情况）

PC→堆栈区

(d) 给 PC 设定异常处理程序的转移目标地址

通过取得对应各异常的向量，转移到异常处理程序的处理。

(2) 执行 RTE 指令和 RTFI 指令时的硬件后处理

(a) PSW 的恢复

（高速中断的情况）

BPSW→PSW

（非高速中断异常的情况）

堆栈区 →PSW

(b) PC 的恢复

（高速中断的情况）

BPC→PC

（非高速中断异常的情况）

堆栈区 →PC

13.5 硬件预处理

以下说明从接受异常请求到执行异常处理程序的硬件预处理。

13.5.1 未定义指令异常

1. 将处理器状态字（PSW）的内容保存到堆栈区（ISP）。
2. 将PSW的处理器模式设定位（PM）、堆栈指针指定位（U）和中断允许位（I）置“0”。
3. 将程序计数器（PC）的内容保存到堆栈区（ISP）。
4. 从地址FFFFFFDCh取向量。
5. 将取到的向量设定到PC后转移到异常处理程序。

13.5.2 特权指令异常

1. 将处理器状态字（PSW）的内容保存到堆栈区（ISP）。
2. 将PSW的处理器模式设定位（PM）、堆栈指针指定位（U）和中断允许位（I）置“0”。
3. 将程序计数器（PC）的内容保存到堆栈区（ISP）。
4. 从地址FFFFFFD0h取向量。
5. 将取到的向量设定到PC后转移到异常处理程序。

13.5.3 复位

1. 对控制寄存器进行初始化。
2. 从地址FFFFFFFCh取向量。
3. 将取到的向量设定到程序计数器（PC）。

13.5.4 非屏蔽中断

1. 将处理器状态字（PSW）的内容保存到堆栈区（ISP）。
2. 将PSW的处理器模式设定位（PM）、堆栈指针指定位（U）和中断允许位（I）置“0”。
3. 当正在执行RMPA、SCMPU、SMOVB、SMOVF、SMOVU、SSTR、SUNTIL、SWHILE各指令时，将正在执行的指令的程序计数器（PC）内容保存到堆栈区（ISP）；而在其他状态下，将下一条指令的PC内容保存到堆栈区（ISP）。
4. 将PSW的处理器中断优先级（IPL[3:0]）置“Fh”。
5. 从地址FFFFFFF8h取向量。
6. 将取到的向量设定到PC后转移到异常处理程序。

13.5.5 中断

1. 将处理器状态字（PSW）的内容保存到堆栈区（ISP）。在高速中断的情况下，保存到备用PSW（BPSW）。
2. 将PSW的处理器模式设定位（PM）、堆栈指针指定位（U）和中断允许位（I）置“0”。
3. 当正在执行RMPA、SCMPU、SMOVB、SMOVF、SMOVU、SSTR、SUNTIL、SWHILE各指令时，将正在执行的指令的程序计数器（PC）内容保存到堆栈区（ISP）；而在其他状态下，将下一条指令的PC内容保存到堆栈区（ISP）。在高速中断的情况下，保存到备用PC（BPC）。
4. 给PSW的处理器中断优先级（IPL[3:0]）设定已接受中断的中断优先级。
5. 从可变量表取已接受中断源的向量。在高速中断的情况下，从高速中断向量寄存器（FINTV）取向量。
6. 将取到的向量设定到PC后转移到异常处理程序。

13.5.6 无条件陷阱

1. 将处理器状态字（PSW）的内容保存到堆栈区（ISP）。
2. 将PSW的处理器模式设定位（PM）、堆栈指针指定位（U）和中断允许位（I）置“0”。
3. 将下一条指令的程序计数器（PC）内容保存到堆栈区（ISP）。
4. 在使用INT指令时，从可变向量表取对应INT指令号的向量。
在使用BRK指令时，从可变向量表的起始地址取向量。
5. 将取得的向量设定到PC后转移到异常处理程序。

13.6 从异常处理程序的返回

如果在异常处理程序的最后执行表 13.3 所示的指令，就恢复异常处理顺序前保存到堆栈区或者控制寄存器（BPC 和 BPSW）的程序计数器（PC）和处理器状态字（PSW）的内容。

表 13.3 异常处理程序的返回指令

异常事件		返回指令
未定义指令异常		RTE
特权指令异常		RTE
复位		不能返回
非屏蔽中断		不能返回
中断	高速中断	RTFI
	非高速中断	RTE
无条件陷阱		RTE

13.7 异常事件的优先级

异常事件的优先级如表 13.4 所示。如果同时发生多个异常，就先接受优先级高的事件。

表 13.4 中断优先级

优先级		异常事件
高  低	1	复位
	2	非屏蔽中断
	3	中断
	4	未定义指令异常 特权指令异常
	5	无条件陷阱

14. 中断控制器 (ICUb)

14.1 概要

中断控制器接受外围模块中断和外部引脚中断，向 CPU 请求中断以及启动 DTC 和 DMAC。
中断控制器的规格和框图分别如表 14.1 和图 14.1 所示。

表 14.1 中断控制器的规格

项目		内容
中断	外围功能中断	<ul style="list-style-type: none"> • 外围模块的中断 • 中断检测：边沿检测 / 电平检测 连接外围模块的各中断源的检测方法是固定的。
	外部引脚中断	<ul style="list-style-type: none"> • IRQ0 ~ IRQ7 引脚的中断 • 中断源数：8 个 • 中断检测：各中断源能设定 Low 电平、下降沿、上升沿或者双边沿。 • 数字滤波器功能：有
	软件中断	<ul style="list-style-type: none"> • 通过写寄存器来产生中断。 • 中断源数：1 个
	事件链接中断	<ul style="list-style-type: none"> • 通过 ELC 事件发生 ELSR18I 中断和 ELSR19I 中断。
	中断优先级	通过寄存器设定优先级。
	高速中断功能	能实现 CPU 中断处理的高速化。只能给 1 个中断源设定此功能。
	DTC 和 DMAC 的控制	能通过中断源启动 DTC 和 DMAC (注 1)。
非屏蔽中断	NMI 引脚中断	<ul style="list-style-type: none"> • NMI 引脚的中断 • 中断检测：下降沿、上升沿 • 数字滤波器功能：有
	振荡停止检测中断	检测到振荡停止时的中断
	WDT 下溢 / 刷新错误	在间隔定时器模式中计数器发生下溢时的中断
	IWDT 下溢 / 刷新错误	递减计数器发生下溢的中断
	电压监视 1 中断	电压检测电路 1 (LVD1) 的电压监视中断
	电压监视 2 中断	电压检测电路 2 (LVD2) 的电压监视中断
从低功耗状态的返回	<ul style="list-style-type: none"> • 睡眠模式：通过非屏蔽中断或者全部中断源返回。 • 全模块时钟停止模式：通过非屏蔽中断、IRQ0 ~ IRQ7 中断、TMR 中断或者 RTC 闹钟 / 周期中断返回。 • 软件待机模式：通过非屏蔽中断、IRQ0 ~ IRQ7 中断或者 RTC 闹钟 / 周期中断返回。 	

注 1. 有关 DTC 和 DMAC 的启动源，请参照“表 14.3 中断向量表”。

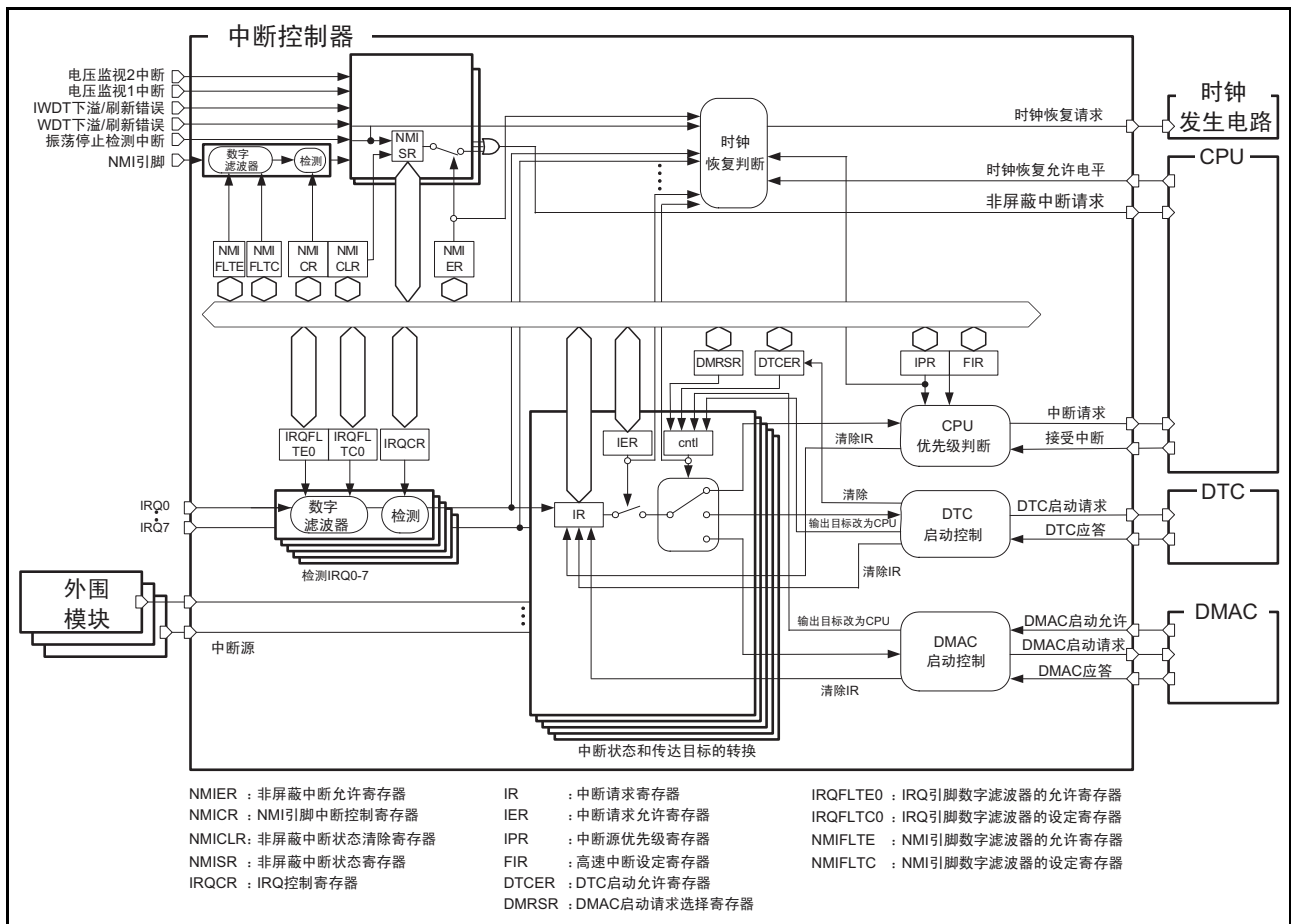


图 14.1 中断控制器的框图

中断控制器使用的输入 / 输出引脚如表 14.2 所示。

表 14.2 中断控制器的输入 / 输出引脚

引脚名	输入 / 输出	功能
NMI	输入	非屏蔽中断请求引脚
IRQ0 ~ IRQ7	输入	外部中断请求引脚

14.2 寄存器说明

14.2.1 中断请求寄存器 n (IRn) (n= 中断向量号)

地址 0008 7010h ~ 0008 70F9h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	IR
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	IR	中断状态标志	0: 无中断请求 1: 有中断请求	R/(W) (注 1)
b7-b1	—	保留位	读写值都为“0”。	R/W

注 1. 在边沿检测源的情况下，只能写“0”而不能写“1”。
在电平检测源的情况下，不能写此位。

各中断源都有 IRn 寄存器，n 对应中断向量号。
中断源和中断向量号的对应请参照“表 14.3 中断向量表”。

IR 标志 (中断状态标志)

这是中断请求的状态标志。如果产生中断请求，此标志就变为“1”。为了检测中断请求，需要通过外围模块的中断允许位允许中断请求的输出。

中断请求的检测方法有边沿检测和电平检测。对于外围模块中断，给各中断源规定了边沿检测或者电平检测。对于 IRQi 引脚中断，能通过设定 IRQCRi.IRQMD[1:0] 位 (i=0 ~ 7)，进行边沿检测和电平检测的转换。有关各中断源的检测方法，请参照“表 14.3 中断向量表”。

(1) 边沿检测的情况

[为“1”的条件]

- 如果产生外围模块中断请求或者 IRQi 引脚中断请求，此标志就变为“1”。有关各外围模块中断请求的产生，请参照各外围模块的章节。

[为“0”的条件]

- 如果中断请求目标接受中断请求，此标志就变为“0”。
- 如果给 IR 标志写“0”，此标志就变为“0”。但是，在将中断请求目标设定为 DTC 或者 DMAC 时，禁止给 IR 标志写“0”。

(2) 电平检测的情况

[为“1”的条件]

- 在产生外围模块中断请求或者 IRQi 引脚中断请求的期间，此标志为“1”。有关各外围模块中断请求的产生，请参照各外围模块的章节。

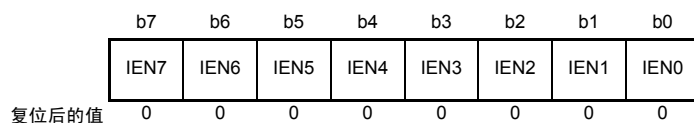
[为“0”的条件]

- 如果清除中断请求的输出源，此标志就变为“0”（即使中断请求目标接受中断请求，此标志也不变为“0”）。有关各外围模块中断请求的清除，请参照各外围模块的章节。

要在通过电平检测使用 IRQi 引脚的情况下取消中断时，必须将 IRQi 引脚置为 High 电平。
在电平检测的情况下，禁止给 IR 标志写“0”和“1”。

14.2.2 中断请求允许寄存器 m (IERm) (m=02h ~ 1Fh)

地址 0008 7202h ~ 0008 721Fh



位	符号	位名	功能	R/W
b0	IEN0	中断请求允许位 0	0: 禁止中断请求 1: 允许中断请求	R/W
b1	IEN1	中断请求允许位 1		R/W
b2	IEN2	中断请求允许位 2		R/W
b3	IEN3	中断请求允许位 3		R/W
b4	IEN4	中断请求允许位 4		R/W
b5	IEN5	中断请求允许位 5		R/W
b6	IEN6	中断请求允许位 6		R/W
b7	IEN7	中断请求允许位 7		R/W

注. 被保留的向量号的对应位的读写值都为“0”。

IENj 位 (中断请求允许位 j) (j = 7 ~ 0)

当 IENj 位为“1”时, 将中断请求输出到中断请求目标。

当 IENj 位为“0”时, 不将中断请求输出到中断请求目标。

IRn.IR 标志不受 IENj 位的影响。即使 IENj 位为“0”, 在“14.2.1 中断请求寄存器 n (IRn) (n= 中断向量号)”所示的条件下, IR 标志也发生变化。

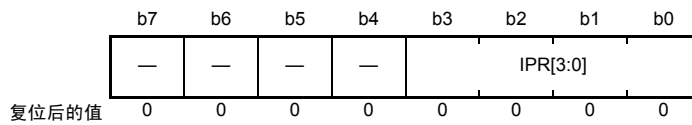
各中断源 (向量号) 都有 IERm.IENj 位。

中断源和 IERm.IENj 位的对应请参照“表 14.3 中断向量表”。

有关选择中断请求目标时的 IERm.IENj 位的设定步骤, 请参照“14.4.3 中断请求目标的选择”。

14.2.3 中断源优先级寄存器 n (IPRn) (n=000 ~ 249)

地址 0008 7300h ~ 0008 73F9h



位	符号	位名	功能	R/W
b3-b0	IPR[3:0]	中断优先级设定位	b3 b0 0000: 0级 (禁止中断) (注1) 0001: 1级 0010: 2级 0011: 3级 0100: 4级 0101: 5级 0110: 6级 0111: 7级 1000: 8级 1001: 9级 1010: 10级 1011: 11级 1100: 12级 1101: 13级 1110: 14级 1111: 15级 (最高)	R/W
b7-b4	—	保留位	读写值都为“0”。	R/W

注 1. 在设定为高速中断的情况下, 即使为 0 级, 也能发行中断。

中断源和 IPRn 寄存器的对应请参照“表 14.3 中断向量表”。

IPR[3:0] 位 (中断优先级设定位)

这些位选择对应的中断源优先级。

IPR[3:0] 位选择的优先级只在判断向 CPU 请求的中断优先级时被参照, 不影响 DTC 和 DMAC 的启动请求。

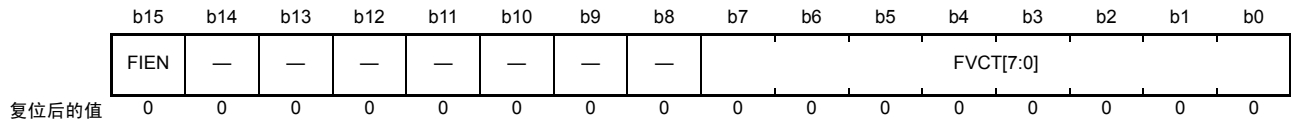
CPU 只接受优先级高于 PSW.IPL[3:0] 位所示优先级的中断请求, 进行中断处理。

如果同时发生多个中断请求, 就通过 IPR[3:0] 位的设定值进行优先级的比较。如果同时产生相同优先级的中断请求, 就优先接受向量号小的中断源。

必须在禁止中断请求 (IERm.IENj 位 =0) 的状态下写这些位。

14.2.4 高速中断设定寄存器 (FIR)

地址 0008 72F0h



位	符号	位名	功能	R/W
b7-b0	FVCT[7:0]	高速中断向量设定位	指定要设定为高速中断的向量号。	R/W
b14-b8	—	保留位	读写值都为“0”。	R/W
b15	FIEN	高速中断允许位	0: 禁止高速中断 1: 允许高速中断	R/W

通过设定 FIR 寄存器实现高速化的功能只对向 CPU 请求的中断有效，不影响 DTC 和 DMAC 的启动请求。必须在禁止中断请求 (IERm.IENj 位 =0) 的状态下写此寄存器。

FVCT[7:0] 位 (高速中断向量设定位)

这些位指定要使用高速中断功能的中断向量号。

FIEN 位 (高速中断允许位)

此位是允许高速中断的位。

如果将 FIEN 位置“1”，FVCT[7:0] 位所设向量号的中断就变为高速中断。

当 FIEN 位为“1”时，如果中断请求目标为 CPU 并且产生 FVCT[7:0] 位所指向量号的中断请求，就作为高速中断向 CPU 输出中断请求，与 IPRn 寄存器的设定无关。有关将高速中断用于从软件待机模式的返回，请参照“14.6.3 从软件待机模式的返回”。

对于通过 IERm.IENj 位 (m=02h ~ 1Fh、j=7 ~ 0) 禁止中断请求的中断源，不将中断请求输出到 CPU。能设定的向量号请参照“表 14.3 中断向量表”。

FVCT[7:0] 位不能指定被保留的向量号。

高速中断的详细内容请参照“13. 异常处理”和“14.4.5 高速中断”。

14.2.5 软件中断启动寄存器 (SWINTR)

地址 0008 72E0h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	SWINT
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	SWINT	软件中断启动位	读取值为“0”。通过写“1”，发行软件中断请求。写“0”无效。	R/(W) (注1)
b7-b1	—	保留位	读写值都为“0”。	R/W

注 1. 只能写“1”。

SWINT 位 (软件中断启动位)

如果给 SWINT 位写“1”，中断请求寄存器 027 (IR027) 就变为“1”。

如果在将 DTC 启动允许寄存器 027 (DTCER027) 置“0”后给 SWINT 位写“1”，就向 CPU 请求中断。

如果在将 DTC 启动允许寄存器 027 (DTCER027) 置“1”后给 SWINT 位写“1”，就发行 DTC 启动请求。

14.2.6 DTC 启动允许寄存器 n (DTCERn) (n= 中断向量号)

地址 0008 711Bh ~ 0008 71F8h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	DTCE
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	DTCE	DTC 启动允许位	0: 禁止启动 DTC 1: 允许启动 DTC	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

对于和 DMAC 启动请求选择的中断源相同的中断源，禁止设定为允许启动 DTC。有关与中断源的对应，请参照“表 14.3 中断向量表”。

DTCE 位 (DTC 启动允许位)

如果将 DTCE 位置“1”，就选择对应的中断源作为 DTC 启动源。

[为“1”的条件]

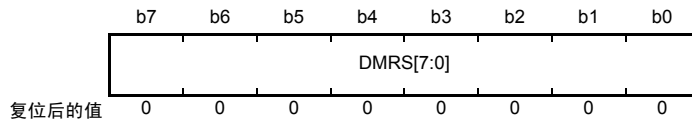
- 给 DTCE 位写“1”时

[为“0”的条件]

- 当指定次数的数据传送结束时 (在链传送的情况下，当最后链传送的指定次数的数据传送结束时)
- 给 DTCE 位写“0”时

14.2.7 DMAC 启动请求选择寄存器 m (DMRSRm) (m=DMAC 通道号)

地址 DMRSR0 0008 7400h、DMRSR1 0008 7404h、DMRSR2 0008 7408h、DMRSR3 0008 740Ch



位	符号	位名	功能	R/W
b7-b0	DMRS[7:0]	DMAC 启动源选择位	设定 DMAC 启动请求的向量号。	R/W

禁止给多个 DMRSRm 寄存器设定相同的中断源。对于和 DMRSRm 寄存器设定的中断源相同的中断源，禁止设定为允许启动 DTC。如果违反这些禁止事项，就不保证运行。

DMRS[7:0] 位 (DMAC 启动源选择位)

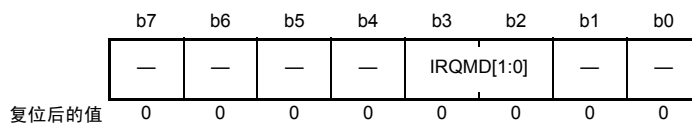
用 8 位指定要启动 DMAC 的中断源的向量号。不能设定未分配为 DMAC 的启动源的向量号。

中断源的向量号请参照“表 14.3 中断向量表”。

必须在 DMA 传送允许寄存器的 DMA 传送允许位 (DMACm.DMCNT.DTE) 为“0”的状态下写 DMRSRm 寄存器。

14.2.8 IRQ 控制寄存器 i (IRQCRi) (i=0 ~ 7)

地址 0008 7500h ~ 0008 7507h



位	符号	位名	功能	R/W
b1-b0	—	保留位	读写值都为“0”。	R/W
b3-b2	IRQMD[1:0]	IRQ 检测设定位	b3 b2 0 0: Low 电平 0 1: 下降沿 1 0: 上升沿 1 1: 双边沿	R/W
b7-b4	—	保留位	读写值都为“0”。	R/W

必须在对应的中断请求允许位为禁止中断请求 (IERm.IENj 位为“0”) 的状态下更改此寄存器的设定。在更改寄存器后，必须清除 IR 标志，然后将中断请求允许位设定为允许。要更改为 Low 电平时，不需要清除 IR 标志。

IRQMD[1:0] 位 (IRQ 检测设定位)

这些位设定外部引脚中断源 (IRQ0 ~ IRQ7) 的检测方法。

外部引脚中断的检测设定步骤请参照“14.4.7 外部引脚中断”。

14.2.9 IRQ 引脚数字滤波器的允许寄存器 0 (IRQFLTE0)

地址 0008 7510h

b7	b6	b5	b4	b3	b2	b1	b0
FLTEN7	FLTEN6	FLTEN5	FLTEN4	FLTEN3	FLTEN2	FLTEN1	FLTEN0
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	FLTEN0	IRQ0 数字滤波器允许位	0: 数字滤波器无效 1: 数字滤波器有效	R/W
b1	FLTEN1	IRQ1 数字滤波器允许位		R/W
b2	FLTEN2	IRQ2 数字滤波器允许位		R/W
b3	FLTEN3	IRQ3 数字滤波器允许位		R/W
b4	FLTEN4	IRQ4 数字滤波器允许位		R/W
b5	FLTEN5	IRQ5 数字滤波器允许位		R/W
b6	FLTEN6	IRQ6 数字滤波器允许位		R/W
b7	FLTEN7	IRQ7 数字滤波器允许位		R/W

FLTENi 位 (IRQi 数字滤波器允许位) (i = 0 ~ 7)

此位允许使用外部引脚中断源 (IRQ0 ~ IRQ7) 的数字滤波器。

当 FLTENi 位为“1”时，数字滤波器有效；当 FLTENi 位为“0”时，数字滤波器功能无效。

在 IRQFLTC0.FCLKSELi[1:0] 位设定的每个采样时钟对 IRQi 引脚电平进行采样，当电平 3 次相同时，更改数字滤波器的输出电平。

数字滤波器的详细内容请参照“14.4.6 数字滤波器”。

14.2.10 IRQ 引脚数字滤波器的设定寄存器 0 (IRQFLTC0)

地址 0008 7514h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
FCLKSEL7[1:0]		FCLKSEL6[1:0]		FCLKSEL5[1:0]		FCLKSEL4[1:0]		FCLKSEL3[1:0]		FCLKSEL2[1:0]		FCLKSEL1[1:0]		FCLKSEL0[1:0]	
复位后的值 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0															

位	符号	位名	功能	R/W
b1-b0	FCLKSEL0[1:0]	IRQ0 数字滤波器的采样时钟设定位	0 0: PCLK	R/W
b3-b2	FCLKSEL1[1:0]	IRQ1 数字滤波器的采样时钟设定位	0 1: PCLK/8	R/W
b5-b4	FCLKSEL2[1:0]	IRQ2 数字滤波器的采样时钟设定位	1 0: PCLK/32	R/W
b7-b6	FCLKSEL3[1:0]	IRQ3 数字滤波器的采样时钟设定位	1 1: PCLK/64	R/W
b9-b8	FCLKSEL4[1:0]	IRQ4 数字滤波器的采样时钟设定位		R/W
b11-b10	FCLKSEL5[1:0]	IRQ5 数字滤波器的采样时钟设定位		R/W
b13-b12	FCLKSEL6[1:0]	IRQ6 数字滤波器的采样时钟设定位		R/W
b15-b14	FCLKSEL7[1:0]	IRQ7 数字滤波器的采样时钟设定位		R/W

FCLKSELi[1:0] 位 (IRQi 数字滤波器的采样时钟设定位) (i = 0 ~ 7)

这些位选择外部引脚中断请求引脚 (IRQ0 ~ IRQ7) 的数字滤波器的采样时钟。

从 PCLK (每个时钟)、PCLK/8 (每 8 个时钟采样 1 次)、PCLK/32 (每 32 个时钟采样 1 次) 或者 PCLK/64 (每 64 个时钟采样 1 次) 中选择采样时钟。

数字滤波器的详细内容请参照“14.4.6 数字滤波器”。

14.2.11 非屏蔽中断状态寄存器 (NMISR)

地址 0008 7580h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	LVD2ST	LVD1ST	IWDTST	WDTST	OSTST	NMIST
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	NMIST	NMI 状态标志	0: 无 NMI 引脚中断请求 1: 有 NMI 引脚中断请求	R
b1	OSTST	振荡停止检测中断状态标志	0: 无振荡停止检测中断请求 1: 有振荡停止检测中断请求	R
b2	WDTST	WDT 下溢 / 刷新错误状态标志	0: 无 WDT 下溢 / 刷新错误中断请求 1: 有 WDT 下溢 / 刷新错误中断请求	R
b3	IWDTST	IWDT 下溢 / 刷新错误状态标志	0: 无 IWDT 下溢 / 刷新错误中断请求 1: 有 IWDT 下溢 / 刷新错误中断请求	R
b4	LVD1ST	电压监视 1 中断状态标志	0: 无电压监视 1 中断请求 1: 有电压监视 1 中断请求	R
b5	LVD2ST	电压监视 2 中断状态标志	0: 无电压监视 2 中断请求 1: 有电压监视 2 中断请求	R
b7-b6	—	保留位	读取值为“0”，写操作无效。	R

NMISR 寄存器是监视非屏蔽中断源状态的寄存器。忽视对 NMISR 寄存器的写操作。

非屏蔽中断允许寄存器 (NMIER) 的设定不影响这些状态标志。

必须在非屏蔽中断处理程序结束前读 NMISR 寄存器并且确认其他非屏蔽中断的发生状况。必须在确认 NMISR 寄存器的全部位都为“0”后结束处理程序。

NMIST 标志 (NMI 状态标志)

此标志表示 NMI 引脚中断请求。

只能读 NMIST 标志，通过 NMICLR.NMICLR 位清除此标志。

[为“1”的条件]

- 将 NMICR.NMIMD 位设定的边沿输入到 NMI 引脚时

[为“0”的条件]

- 给 NMICLR.NMICLR 位写“1”时

OSTST 标志 (振荡停止检测中断状态标志)

此标志表示振荡停止检测中断请求。

只能读 OSTST 标志，通过 NMICLR.OSTCLR 位清除此标志。

[为“1”的条件]

- 发生振荡停止检测中断时

[为“0”的条件]

- 给 NMICLR.OSTCLR 位写“1”时

WDTST 标志 (WDT 下溢 / 刷新错误状态标志)

此标志表示 WDT 下溢 / 刷新错误中断请求。

只能读 WDTST 标志, 通过 NMICLR.WDTCLR 位清除此标志。

[为 “1” 的条件]

- 发生 WDT 下溢 / 刷新错误中断时

[为 “0” 的条件]

- 给 NMICLR.WDTCLR 位写 “1” 时

IWDTST 标志 (IWDT 下溢 / 刷新错误状态标志)

此标志表示 IWDT 下溢 / 刷新错误中断请求。

只能读 IWDTST 标志, 通过 NMICLR.IWDTCLR 位清除此标志。

[为 “1” 的条件]

- 在发生源为允许发生中断的情况下发生 IWDT 下溢 / 刷新错误中断时

[为 “0” 的条件]

- 给 NMICLR.IWDTCLR 位写 “1” 时

LVD1ST 标志 (电压监视 1 中断状态标志)

此标志表示电压监视 1 中断请求。

只能读 LVD1ST 标志, 通过 NMICLR.LVD1CLR 位清除此标志。

[为 “1” 的条件]

- 在发生源为允许发生中断的情况下发生电压监视 1 中断时

[为 “0” 的条件]

- 给 NMICLR.LVD1CLR 位写 “1” 时

LVD2ST 标志 (电压监视 2 中断状态标志)

此标志表示电压监视 2 中断请求。

只能读 LVD2ST 标志, 通过 NMICLR.LVD2CLR 位清除此标志。

[为 “1” 的条件]

- 在发生源为允许发生中断的情况下发生电压监视 2 中断时

[为 “0” 的条件]

- 给 NMICLR.LVD2CLR 位写 “1” 时

14.2.12 非屏蔽中断允许寄存器 (NMIER)

地址 0008 7581h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	LVD2EN	LVD1EN	IWDTEN	WDTEN	OSTEN	NMIEN
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	NMIEN	NMI 引脚中断允许位	0: 禁止 NMI 引脚中断 1: 允许 NMI 引脚中断	R/(W) (注 1)
b1	OSTEN	振荡停止检测中断允许位	0: 禁止振荡停止检测中断 1: 允许振荡停止检测中断	R/(W) (注 1)
b2	WDTEN	WDT 下溢 / 刷新错误允许位	0: 禁止 WDT 下溢 / 刷新错误中断 1: 允许 WDT 下溢 / 刷新错误中断	R/(W) (注 1)
b3	IWDTEN	IWDT 下溢 / 刷新错误允许位	0: 禁止 IWDT 下溢 / 刷新错误中断 1: 允许 IWDT 下溢 / 刷新错误中断	R/(W) (注 1)
b4	LVD1EN	电压监视 1 中断允许位	0: 禁止电压监视 1 中断 1: 允许电压监视 1 中断	R/(W) (注 1)
b5	LVD2EN	电压监视 2 中断允许位	0: 禁止电压监视 2 中断 1: 允许电压监视 2 中断	R/(W) (注 1)
b7-b6	—	保留位	读写值都为“0”。	R/W

注 1. 只能写 1 次“1”，以后的写操作无效。

NMIEN 位 (NMI 引脚中断允许位)

此位是允许使用 NMI 引脚中断的位。
只能写 1 次“1”，以后的写操作无效。
不能写“0”。

OSTEN 位 (振荡停止检测中断允许位)

此位是允许使用振荡停止检测中断的位。
只能写 1 次“1”，以后的写操作无效。
不能写“0”。

WDTEN 位 (WDT 下溢 / 刷新错误允许位)

此位是允许使用 WDT 下溢 / 刷新错误中断的位。
只能写 1 次“1”，以后的写操作无效。
不能写“0”。

IWDTEN 位 (IWDT 下溢 / 刷新错误允许位)

此位是允许使用 IWDT 下溢 / 刷新错误中断的位。
只能写 1 次“1”，以后的写操作无效。
不能写“0”。

LVD1EN 位 (电压监视 1 中断允许位)

此位是允许使用电压监视 1 中断的位。
只能写 1 次“1”，以后的写操作无效。
不能写“0”。

LVD2EN 位 (电压监视 2 中断允许位)

此位是允许使用电压监视 2 中断的位。
只能写 1 次“1”，以后的写操作无效。
不能写“0”。

14.2.13 非屏蔽中断状态清除寄存器 (NMICLR)

地址 0008 7582h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	LVD2CLR	LVD1CLR	IWDTCLR	WDTCLR	OSTCLR	NMICLR
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	NMICLR	NMI 清除位	读取值为“0”。通过写“1”，清除 NMISR.NMIST 标志。写“0”无效。	R/(W) (注1)
b1	OSTCLR	OST 清除位	读取值为“0”。通过写“1”，清除 NMISR.OSTST 标志。写“0”无效。	R/(W) (注1)
b2	WDTCLR	WDT 清除位	读取值为“0”。通过写“1”，清除 NMISR.WDTST 标志。写“0”无效。	R/(W) (注1)
b3	IWDTCLR	IWDT 清除位	读取值为“0”。通过写“1”，清除 NMISR.IWDTST 标志。写“0”无效。	R/(W) (注1)
b4	LVD1CLR	LVD1 清除位	读取值为“0”。通过写“1”，清除 NMISR.LVD1ST 标志。写“0”无效。	R/(W) (注1)
b5	LVD2CLR	LVD2 清除位	读取值为“0”。通过写“1”，清除 NMISR.LVD2ST 标志。写“0”无效。	R/(W) (注1)
b7-b6	—	保留位	读写值都为“0”。	R/W

注 1. 只能写“1”。

NMICLR 位 (NMI 清除位)

如果写“1”，NMISR.NMIST 标志就变为“0”。不保持“1”的状态，读取值为“0”。

OSTCLR 位 (OST 清除位)

如果写“1”，NMISR.OSTST 标志就变为“0”。不保持“1”的状态，读取值为“0”。

WDTCLR 位 (WDT 清除位)

如果写“1”，NMISR.WDTST 标志就变为“0”。不保持“1”的状态，读取值为“0”。

IWDTCLR 位 (IWDT 清除位)

如果写“1”，NMISR.IWDTST 标志就变为“0”。不保持“1”的状态，读取值为“0”。

LVD1CLR 位 (LVD1 清除位)

如果写“1”，NMISR.LVD1ST 标志就变为“0”。不保持“1”的状态，读取值为“0”。

LVD2CLR 位 (LVD2 清除位)

如果写“1”，NMISR.LVD2ST 标志就变为“0”。不保持“1”的状态，读取值为“0”。

14.2.14 NMI 引脚中断控制寄存器 (NMICR)

地址 0008 7583h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	NMIMD	—	—	—
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b2-b0	—	保留位	读写值都为“0”。	R/W
b3	NMIMD	NMI 检测设定位	0: 下降沿 1: 上升沿	R/W
b7-b4	—	保留位	读写值都为“0”。	R/W

必须在允许使用 NMI 引脚中断（将 NMIER.NMIEN 位置“1”）前更改 NMICR 寄存器的设定。

NMIMD 位 (NMI 检测设定位)

此位设定 NMI 引脚中断的检测方法。

14.2.15 NMI 引脚数字滤波器的允许寄存器 (NMIFLTE)

地址 0008 7590h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	NFLTEN
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	NFLTEN	NMI 数字滤波器允许位	0: 数字滤波器无效 1: 数字滤波器有效	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

NFLTEN 位 (NMI 数字滤波器允许位)

此位允许使用 NMI 引脚中断的数字滤波器。

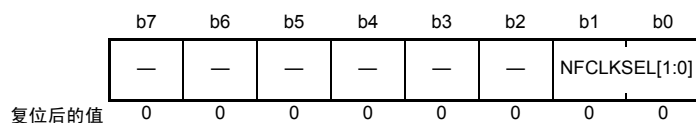
当 NFLTEN 位为“1”时，数字滤波器有效；当 NFLTEN 位为“0”时，数字滤波器功能无效。

在 NMIFLTC.NFCLKSEL[1:0] 位设定的每个采样时钟对 NMI 引脚电平进行采样，当电平 3 次相同时，更改数字滤波器的输出电平。

数字滤波器的详细内容请参照“14.4.6 数字滤波器”。

14.2.16 NMI 引脚数字滤波器的设定寄存器 (NMIFLTC)

地址 0008 7594h



位	符号	位名	功能	R/W
b1-b0	NFCLKSEL[1:0]	NMI 数字滤波器的采样时钟设定位	b1 b0 0 0: PCLK 0 1: PCLK/8 1 0: PCLK/32 1 1: PCLK/64	R/W
b7-b2	—	保留位	读写值都为“0”。	R/W

NFCLKSEL[1:0] 位 (NMI 数字滤波器的采样时钟设定位)

这些位选择 NMI 引脚中断的数字滤波器的采样时钟。

从 PCLK (每个时钟)、PCLK/8 (每 8 个时钟采样 1 次)、PCLK/32 (每 32 个时钟采样 1 次) 或者 PCLK/64 (每 64 个时钟采样 1 次) 中选择采样时钟。

数字滤波器的详细内容请参照“14.4.6 数字滤波器”。

14.3 向量表

通过中断控制器检测的异常事件有中断和非屏蔽中断。

如果 CPU 接受中断或者非屏蔽中断，就从向量表取 4 字节的向量地址。

14.3.1 中断向量表

中断向量表连续分配在从 CPU 的中断表寄存器 (INTB) 设定地址开始的 1024 字节 (4 字节 × 256 个中断源) 的区域。必须在允许中断前设定 INTB 寄存器，并且给 INTB 寄存器设定 4 的倍数。

中断向量表如表 14.3 所示，表 14.3 的各项目的内容如下：

项目	内容
100 引脚	用“○”表示各封装（引脚数）的中断。
80 引脚	
64 引脚	
中断请求发生源	表示中断请求发生源的名称。
名称	表示中断名称。
向量号	表示向量号。
向量地址的偏移	表示向量基址偏移值。
中断的检测方法	用“边沿”和“电平”表示中断的检测方法。
CPU 中断	用“○”表示 CPU 中断源。
DTC 启动	用“○”表示 DTC 启动源。
DMAC 启动	用“○”表示 DMAC 启动源。
sstb 返回	用“○”表示从软件待机模式返回的返回源。
sacs 返回	用“○”表示从全模块时钟停止模式返回的返回源。
IER	表示对应向量号的 IER 寄存器和位名。
IPR	表示对应中断源的 IPR 寄存器。
DTCER	表示对应 DTC 启动源的 DTCER 寄存器。

表 14.3 中断向量表 (1/6)

100 引脚	80 引脚	64 引脚	中断请求发生源	名称	向量号 (注1)	向量地址 的偏移	中断的 检测方法	CPU 中断	DTC 中断	DMAC 中断	sstb 返回	sacs 返回	IER	IPR	DTCER
—	—	—	—	保留	0	0000h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	1	0004h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	2	0008h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	3	000Ch	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	4	0010h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	5	0014h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	6	0018h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	7	001Ch	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	8	0020h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	9	0024h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	10	0028h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	11	002h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	12	0030h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	13	0034h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	14	0038h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	15	003Ch	—	x	x	x	x	x	—	—	—
○	○	○	BSC	BUSERR	16	0040h	电平	○	x	x	x	x	IER02.IEN0	IPR000	—
—	—	—	—	保留	17	0044h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	18	0048h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	19	004Ch	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	20	0050h	—	x	x	x	x	x	—	—	—
○	○	○	FCU	FIFERR	21	0054h	电平	○	x	x	x	x	IER02.IEN5	IPR001	—
—	—	—	—	保留	22	0058h	—	x	x	x	x	x	—	—	—
○	○	○	FCU	FRDYI	23	005Ch	电平	○	x	x	x	x	IER02.IEN7	IPR002	—
—	—	—	—	保留	24	0060h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	25	0064h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	26	0068h	—	x	x	x	x	x	—	—	—
○	○	○	ICU	SWINT	27	006Ch	边沿	○	○	x	x	x	IER03.IEN3	IPR003	DTCER027
○	○	○	CMT0	CMIO	28	0070h	边沿	○	○	○	x	x	IER03.IEN4	IPR004	DTCER028
○	○	○	CMT1	CM11	29	0074h	边沿	○	○	○	x	x	IER03.IEN5	IPR005	DTCER029
○	○	○	CMT2	CM12	30	0078h	边沿	○	○	○	x	x	IER03.IEN6	IPR006	DTCER030
○	○	○	CMT3	CM13	31	007Ch	边沿	○	○	○	x	x	IER03.IEN7	IPR007	DTCER031
○	○	○	CAC	FERRF	32	0080h	电平	○	x	x	x	x	IER04.IEN0	IPR032	—
○	○	○		MENDF	33	0084h	电平	○	x	x	x	x	IER04.IEN1	IPR033	—
○	○	○		OVFF	34	0088h	电平	○	x	x	x	x	IER04.IEN2	IPR034	—
—	—	—	—	保留	35	008Ch	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	36	0090h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	37	0094h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	38	0098h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	39	009Ch	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	40	00A0h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	41	00A4h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	42	00A8h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	43	00ACh	—	x	x	x	x	x	—	—	—
○	○	○	RSPI0	SPEI0	44	00B0h	电平	○	x	x	x	x	IER05.IEN4	IPR044	—
○	○	○		SPRI0	45	00B4h	边沿	○	○	○	x	x	IER05.IEN5		DTCER045
○	○	○		SPTI0	46	00B8h	边沿	○	○	○	x	x	IER05.IEN6		DTCER046
○	○	○		SPII0	47	00BCh	电平	○	x	x	x	x	IER05.IEN7		—

表 14.3 中断向量表 (2/6)

100 引脚	80 引脚	64 引脚	中断请求发生源	名称	向量号 (注1)	向量地址 的偏移	中断的 检测方法	CPU 中断	DTC 中断	DMAC 中断	sstb 返回	sacs 返回	IER	IPR	DTCER
—	—	—	—	保留	48	00C0h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	49	00C4h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	50	00C8h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	51	00CCh	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	52	00D0h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	53	00D4h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	54	00D8h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	55	00DCh	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	56	00E0h	—	x	x	x	x	x	—	—	—
○	○	○	DOC	DOPCF	57	00E4h	电平	○	x	x	x	x	IER07.IEN1	IPR057	—
○	○	○	CMPB	CMPB0	58	00E8h	边沿	○	○	○	x	x	IER07.IEN2	IPR058	DTCER058
○	○	○		CMPB1	59	00ECh	边沿	○	○	○	x	x	IER07.IEN3	IPR059	DTCER059
—	—	—	—	保留	60	00F0h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	61	00F4h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	62	00F8h	—	x	x	x	x	x	—	—	—
○	○	○	RTC	COUNTUP	63	00FCh	边沿	○	x	x	x	x	IER07.IEN7	IPR063	—
○	○	○	ICU	IRQ0	64	0100h	边沿 / 电平	○	○	○	○	○	IER08.IEN0	IPR064	DTCER064
○	○	○		IRQ1	65	0104h	边沿 / 电平	○	○	○	○	○	IER08.IEN1	IPR065	DTCER065
○	○	○		IRQ2	66	0108h	边沿 / 电平	○	○	○	○	○	IER08.IEN2	IPR066	DTCER066
○	○	○		IRQ3	67	010Ch	边沿 / 电平	○	○	○	○	○	IER08.IEN3	IPR067	DTCER067
○	○	○		IRQ4	68	0110h	边沿 / 电平	○	○	x	○	○	IER08.IEN4	IPR068	DTCER068
○	○	○		IRQ5	69	0114h	边沿 / 电平	○	○	x	○	○	IER08.IEN5	IPR069	DTCER069
○	○	○		IRQ6	70	0118h	边沿 / 电平	○	○	x	○	○	IER08.IEN6	IPR070	DTCER070
○	○	○		IRQ7	71	011Ch	边沿 / 电平	○	○	x	○	○	IER08.IEN7	IPR071	DTCER071
—	—	—	—	保留	72	0120h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	73	0124h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	74	0128h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	75	012Ch	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	76	0130h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	77	0134h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	78	0138h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	79	013Ch	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	80	0140h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	81	0144h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	82	0148h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	83	014Ch	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	84	0150h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	85	0154h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	86	0158h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	87	015Ch	—	x	x	x	x	x	—	—	—
○	○	○	LVD/CMPA	LVD1/CMPA1	88	0160h	边沿	○	x	x	○	○	IER0B.IEN0	IPR088	—
○	○	○		LVD2/CMPA2	89	0164h	边沿	○	x	x	○	○	IER0B.IEN1	IPR089	—
—	—	—	—	保留	90	0168h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	91	016Ch	—	x	x	x	x	x	—	—	—
○	○	○	RTC	ALM	92	0170h	边沿	○	x	x	○	○	IER0B.IEN4	IPR092	—
○	○	○		PRD	93	0174h	边沿	○	x	x	○	○	IER0B.IEN5	IPR093	—
—	—	—	—	保留	94	0178h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	95	017Ch	—	x	x	x	x	x	—	—	—

表 14.3 中断向量表 (3/6)

100 引脚	80 引脚	64 引脚	中断请求发生源	名称	向量号 (注1)	向量地址 的偏移	中断的 检测方法	CPU 中断	DTC 中断	DMAC 中断	sstb 返回	sacs 返回	IER	IPR	DTCER
—	—	—	—	保留	96	0180h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	97	0184h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	98	0188h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	99	018Ch	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	100	0190h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	101	0194h	—	x	x	x	x	x	—	—	—
○	○	○	S12AD	S12ADI0	102	0198h	边沿	○	○	○	x	x	IER0C.IEN6	IPR102	DTCER102
○	○	○		GBADI	103	019Ch	边沿	○	○	○	x	x	IER0C.IEN7	IPR103	DTCER103
—	—	—	—	保留	104	01A0h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	105	01A4h	—	x	x	x	x	x	—	—	—
○	○	○	ELC	ELSR18I	106	01A8h	边沿	○	○	○	x	x	IER0D.IEN2	IPR106	DTCER106
○	○	○		ELSR19I	107	01ACh	边沿	○	○	○	x	x	IER0D.IEN3	IPR107	DTCER107
—	—	—	—	保留	108	01B0h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	109	01B4h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	110	01B8h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	111	01BCh	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	112	01C0h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	113	01C4h	—	x	x	x	x	x	—	—	—
○	○	○	MTU0	TGIA0	114	01C8h	边沿	○	○	○	x	x	IER0E.IEN2	IPR114	DTCER114
○	○	○		TGIB0	115	01CCh	边沿	○	○	x	x	x	IER0E.IEN3		DTCER115
○	○	○		TGIC0	116	01D0h	边沿	○	○	x	x	x	IER0E.IEN4		DTCER116
○	○	○		TGID0	117	01D4h	边沿	○	○	x	x	x	IER0E.IEN5		DTCER117
○	○	○		TCIV0	118	01D8h	边沿	○	x	x	x	x	IER0E.IEN6	IPR118	—
○	○	○		TGIE0	119	01DCh	边沿	○	x	x	x	x	IER0E.IEN7		—
○	○	○		TGIF0	120	01E0h	边沿	○	x	x	x	x	IER0F.IEN0		—
○	○	○	MTU1	TGIA1	121	01E4h	边沿	○	○	○	x	x	IER0F.IEN1	IPR121	DTCER121
○	○	○		TGIB1	122	01E8h	边沿	○	○	x	x	x	IER0F.IEN2		DTCER122
○	○	○		TCIV1	123	01ECh	边沿	○	x	x	x	x	IER0F.IEN3	IPR123	—
○	○	○		TCIU1	124	01F0h	边沿	○	x	x	x	x	IER0F.IEN4		—
○	○	○	MTU2	TGIA2	125	01F4h	边沿	○	○	○	x	x	IER0F.IEN5	IPR125	DTCER125
○	○	○		TGIB2	126	01F8h	边沿	○	○	x	x	x	IER0F.IEN6		DTCER126
○	○	○		TCIV2	127	01FCh	边沿	○	x	x	x	x	IER0F.IEN7	IPR127	—
○	○	○		TCIU2	128	0200h	边沿	○	x	x	x	x	IER10.IEN0		—
○	○	○	MTU3	TGIA3	129	0204h	边沿	○	○	○	x	x	IER10.IEN1	IPR129	DTCER129
○	○	○		TGIB3	130	0208h	边沿	○	○	x	x	x	IER10.IEN2		DTCER130
○	○	○		TGIC3	131	020Ch	边沿	○	○	x	x	x	IER10.IEN3		DTCER131
○	○	○		TGID3	132	0210h	边沿	○	○	x	x	x	IER10.IEN4		DTCER132
○	○	○		TCIV3	133	0214h	边沿	○	x	x	x	x	IER10.IEN5	IPR133	—
○	○	○	MTU4	TGIA4	134	0218h	边沿	○	○	○	x	x	IER10.IEN6	IPR134	DTCER134
○	○	○		TGIB4	135	021Ch	边沿	○	○	x	x	x	IER10.IEN7		DTCER135
○	○	○		TGIC4	136	0220h	边沿	○	○	x	x	x	IER11.IEN0		DTCER136
○	○	○		TGID4	137	0224h	边沿	○	○	x	x	x	IER11.IEN1		DTCER137
○	○	○		TCIV4	138	0228h	边沿	○	○	x	x	x	IER11.IEN2	IPR138	DTCER138
○	○	○	MTU5	TGIU5	139	022Ch	边沿	○	○	x	x	x	IER11.IEN3	IPR139	DTCER139
○	○	○		TGIV5	140	0230h	边沿	○	○	x	x	x	IER11.IEN4		DTCER140
○	○	○		TGIW5	141	0234h	边沿	○	○	x	x	x	IER11.IEN5		DTCER141
—	—	—	—	保留	142	0238h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	143	023Ch	—	x	x	x	x	x	—	—	—

表 14.3 中断向量表 (4/6)

100 引脚	80 引脚	64 引脚	中断请求发生源	名称	向量号 (注1)	向量地址 的偏移	中断的 检测方法	CPU 中断	DTC 中断	DMAC 中断	sstb 返回	sacs 返回	IER	IPR	DTCER
—	—	—	—	保留	144	0240h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	145	0244h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	146	0248h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	147	024Ch	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	148	0250h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	149	0254h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	150	0258h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	151	025Ch	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	152	0260h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	153	0264h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	154	0268h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	155	026Ch	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	156	0270h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	157	0274h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	158	0278h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	159	027Ch	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	160	0280h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	161	0284h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	162	0288h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	163	028Ch	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	164	0290h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	165	0294h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	166	0298h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	167	029Ch	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	168	02A0h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	169	02A4h	—	x	x	x	x	x	—	—	—
○	○	○	POE	OE11	170	02A8h	电平	○	x	x	x	x	IER15.IEN2	IPR170	—
○	○	○		OE12	171	02ACh	电平	○	x	x	x	x	IER15.IEN3	IPR171	—
—	—	—	—	保留	172	02B0h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	173	02B4h	—	x	x	x	x	x	—	—	—
○	○	○	TMR0	CMIA0	174	02B8h	边沿	○	○	x	x	○	IER15.IEN6	IPR174	DTCER174
○	○	○		CMIB0	175	02BCh	边沿	○	○	x	x	○	IER15.IEN7		DTCER175
○	○	○		OVI0	176	02C0h	边沿	○	x	x	x	○	IER16.IEN0		—
○	○	○	TMR1	CMIA1	177	02C4h	边沿	○	○	x	x	○	IER16.IEN1	IPR177	DTCER177
○	○	○		CMIB1	178	02C8h	边沿	○	○	x	x	○	IER16.IEN2		DTCER178
○	○	○		OVI1	179	02CCh	边沿	○	x	x	x	○	IER16.IEN3		—
○	○	○	TMR2	CMIA2	180	02D0h	边沿	○	○	x	x	○	IER16.IEN4	IPR180	DTCER180
○	○	○		CMIB2	181	02D4h	边沿	○	○	x	x	○	IER16.IEN5		DTCER181
○	○	○		OVI2	182	02D8h	边沿	○	x	x	x	○	IER16.IEN6		—
○	○	○	TMR3	CMIA3	183	02DCh	边沿	○	○	x	x	○	IER16.IEN7	IPR183	DTCER183
○	○	○		CMIB3	184	02E0h	边沿	○	○	x	x	○	IER17.IEN0		DTCER184
○	○	○		OVI3	185	02E4h	边沿	○	x	x	x	○	IER17.IEN1		—
—	—	—	—	保留	186	02E8h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	187	02ECh	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	188	02F0h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	189	02F4h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	190	02F8h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	191	02FCh	—	x	x	x	x	x	—	—	—

表 14.3 中断向量表 (5/6)

100 引脚	80 引脚	64 引脚	中断请求发生源	名称	向量号 (注1)	向量地址 的偏移	中断的 检测方法	CPU 中断	DTC 中断	DMAC 中断	sstb 返回	sacs 返回	IER	IPR	DTCER
—	—	—	—	保留	192	0300h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	193	0304h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	194	0308h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	195	030Ch	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	196	0310h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	197	0314h	—	x	x	x	x	x	—	—	—
○	○	○	DMAC	DMAC0I	198	0318h	边沿	○	○	x	x	x	IER18.IEN6	IPR198	DTCER198
○	○	○		DMAC1I	199	031Ch	边沿	○	○	x	x	x	IER18.IEN7	IPR199	DTCER199
○	○	○		DMAC2I	200	0320h	边沿	○	○	x	x	x	IER19.IEN0	IPR200	DTCER200
○	○	○		DMAC3I	201	0324h	边沿	○	○	x	x	x	IER19.IEN1	IPR201	DTCER201
—	—	—	—	保留	202	0328h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	203	032Ch	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	204	0330h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	205	0334h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	206	0338h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	207	033Ch	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	208	0340h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	209	0344h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	210	0348h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	211	034Ch	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	212	0350h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	213	0354h	—	x	x	x	x	x	—	—	—
○	○	—	SCI0	ERI0	214	0358h	电平	○	x	x	x	x	IER1A.IEN6	IPR214	—
○	○	—		RXI0	215	035Ch	边沿	○	○	○	x	x	IER1A.IEN7		DTCER215
○	○	—		TXI0	216	0360h	边沿	○	○	○	x	x	IER1B.IEN0		DTCER216
○	○	—		TEI0	217	0364h	电平	○	x	x	x	x	IER1B.IEN1		—
○	○	○	SCI1	ERI1	218	0368h	电平	○	x	x	x	x	IER1B.IEN2	IPR218	—
○	○	○		RXI1	219	036Ch	边沿	○	○	○	x	x	IER1B.IEN3		DTCER219
○	○	○		TXI1	220	0370h	边沿	○	○	○	x	x	IER1B.IEN4		DTCER220
○	○	○		TEI1	221	0374h	电平	○	x	x	x	x	IER1B.IEN5		—
○	○	○	SCI5	ERI5	222	0378h	电平	○	x	x	x	x	IER1B.IEN6	IPR222	—
○	○	○		RXI5	223	037Ch	边沿	○	○	○	x	x	IER1B.IEN7		DTCER223
○	○	○		TXI5	224	0380h	边沿	○	○	○	x	x	IER1C.IEN0		DTCER224
○	○	○		TEI5	225	0384h	电平	○	x	x	x	x	IER1C.IEN1		—
○	○	○	SCI6	ERI6	226	0388h	电平	○	x	x	x	x	IER1C.IEN2	IPR226	—
○	○	○		RXI6	227	038Ch	边沿	○	○	○	x	x	IER1C.IEN3		DTCER227
○	○	○		TXI6	228	0390h	边沿	○	○	○	x	x	IER1C.IEN4		DTCER228
○	○	○		TEI6	229	0394h	电平	○	x	x	x	x	IER1C.IEN5		—
○	○	○	SCI8	ERI8	230	0398h	电平	○	x	x	x	x	IER1C.IEN6	IPR230	—
○	○	○		RXI8	231	039Ch	边沿	○	○	○	x	x	IER1C.IEN7		DTCER231
○	○	○		TXI8	232	03A0h	边沿	○	○	○	x	x	IER1D.IEN0		DTCER232
○	○	○		TEI8	233	03A4h	电平	○	x	x	x	x	IER1D.IEN1		—
○	○	○	SCI9	ERI9	234	03A8h	电平	○	x	x	x	x	IER1D.IEN2	IPR234	—
○	○	○		RXI9	235	03ACh	边沿	○	○	○	x	x	IER1D.IEN3		DTCER235
○	○	○		TXI9	236	03B0h	边沿	○	○	○	x	x	IER1D.IEN4		DTCER236
○	○	○		TEI9	237	03B4h	电平	○	x	x	x	x	IER1D.IEN5		—

表 14.3 中断向量表 (6/6)

100 引脚	80 引脚	64 引脚	中断请求发生源	名称	向量号 (注1)	向量地址 的偏移	中断的 检测方法	CPU 中断	DTC 中断	DMAC 中断	sstb 返回	sacs 返回	IER	IPR	DTCER
○	○	○	SCI12	ERI12	238	03B8h	电平	○	x	x	x	x	IER1D.IEN6	IPR238	—
○	○	○		RXI12	239	03BCh	边沿	○	○	○	x	x	IER1D.IEN7		DTCER239
○	○	○		TXI12	240	03C0h	边沿	○	○	○	x	x	IER1E.IEN0		DTCER240
○	○	○		TEI12	241	03C4h	电平	○	x	x	x	x	IER1E.IEN1	—	
○	○	○		SCIX0	242	03C8h	电平	○	x	x	x	x	IER1E.IEN2	IPR242	—
○	○	○		SCIX1	243	03CCh	电平	○	x	x	x	x	IER1E.IEN3	IPR243	—
○	○	○		SCIX2	244	03D0h	电平	○	x	x	x	x	IER1E.IEN4	IPR244	—
○	○	○		SCIX3	245	03D4h	电平	○	x	x	x	x	IER1E.IEN5	IPR245	—
○	○	○		RIIC0	EEI0	246	03D8h	电平	○	x	x	x	x	IER1E.IEN6	IPR246
○	○	○	RXI0		247	03DCh	边沿	○	○	○	x	x	IER1E.IEN7	IPR247	DTCER247
○	○	○	TXI0		248	03E0h	边沿	○	○	○	x	x	IER1F.IEN0	IPR248	DTCER248
○	○	○	TEI0		249	03E4h	电平	○	x	x	x	x	IER1F.IEN1	IPR249	—
—	—	—	—	保留	250	03E8h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	251	03ECh	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	252	03F0h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	253	03F4h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	254	03F8h	—	x	x	x	x	x	—	—	—
—	—	—	—	保留	255	03FCh	—	x	x	x	x	x	—	—	—

注1. 向量号越小, 优先级越高。

14.3.2 高速中断的向量表地址

被设定为高速中断的中断向量表地址由 CPU 的高速中断向量寄存器 (FINTV) 指定。

14.3.3 非屏蔽中断的向量表地址

非屏蔽中断的向量表地址为“FFFF FFF8h”。

14.4 中断的运行说明

中断控制器进行以下处理:

- 中断检测
- 中断的允许或者禁止控制
- 中断请求目标 (CPU中断、DTC的启动、DMAC的启动) 的选择
- 中断优先级的判断

14.4.1 中断检测

中断请求的检测方法有电平检测和边沿检测两种。

对于 IRQi 引脚 (i=0 ~ 7) 的外部中断请求, 能通过设定 IRQCRi.IRQMD[1:0] 位进行边沿检测和电平检测的转换。

对于外围模块的中断请求, 给各中断源规定了边沿检测或者电平检测。

对应各中断源的检测方法请参照“表 14.3 中断向量表”。

14.4.1.1 边沿检测的中断状态标志

对外围功能中断和外部引脚中断进行边沿检测时的 IRn.IR 标志变化如图 14.2 所示。

在产生中断请求时的中断信号的变化位置, IRn.IR 标志变为“1”。当中断请求目标为 CPU 时, 如果接受中断, IRn.IR 标志就自动变为“0”。当中断请求目标为 DMAC 或者 DTC 时, IRn.IR 标志变“0”的时序因 DMAC/DTC 的传送设定和传送次数而不同。详细内容请参照“表 14.4 启动 DMAC 或者 DTC 时的运行”。不需要通过软件清除 IRn.IR 标志。

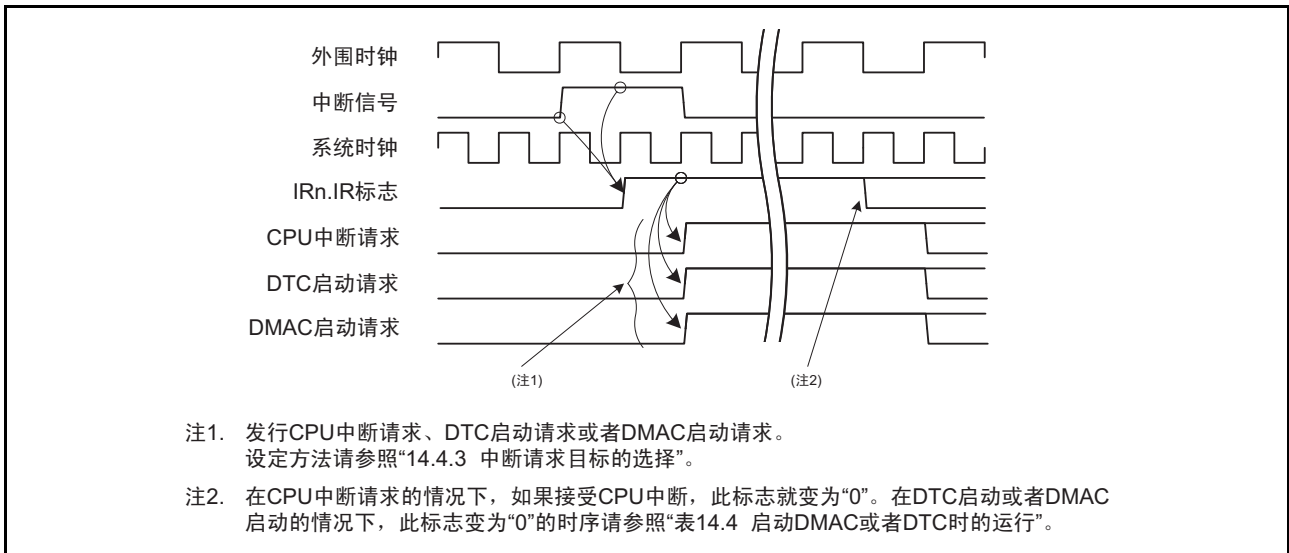


图 14.2 边沿检测时的 IRn.IR 标志变化

图 14.3 ~ 图 14.6 的中断信号是中断控制器的信号。中断向量号 64 ~ 95 的中断时序和其他中断不同。在中断向量号 64 ~ 79 的 IRQ 引脚中断的情况下, 从 IRQ 引脚输入增加内部延迟 +2 个 PCLK 的延迟; 在中断向量号 80 ~ 95 的中断的情况下, 增加 2 个 PCLK 的延迟。

如果在每个周期产生中断信号, 就不能检测后续的中断。对于连续的中断请求, 必须至少空出系统时钟和外围时钟中频率慢的 2 个时钟周期。

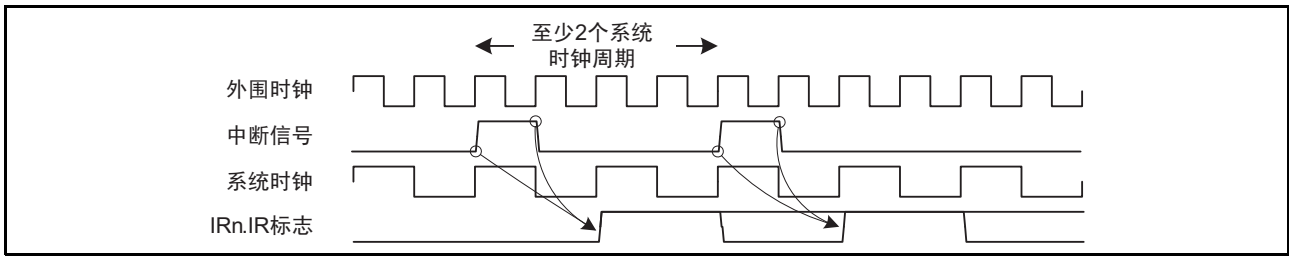


图 14.3 连续中断请求的发行间隔 (系统时钟频率 < 外围时钟频率的情况)

在产生中断请求并且 IRn.IR 标志为“1”的状态下，忽视再次产生的中断请求 (注 1)。IRn.IR 标志再次被置位的时序如图 14.4 所示。

注 1. 但是，在 SCI、RSPI、RIIC 的各发送中断或者接收中断的情况下，保持在 IRn.IR 标志为“1”的状态下产生的中断请求，在 IRn.IR 标志变为“0”后，因被保持的请求使 IRn.IR 标志再次变为“1”。详细内容请参照“28. 串行通信接口 (SC1c、SC1d)”、“29. I²C 总线接口 (RIIC)”和“30. 串行外围接口 (RSPI)”中的各中断的说明。

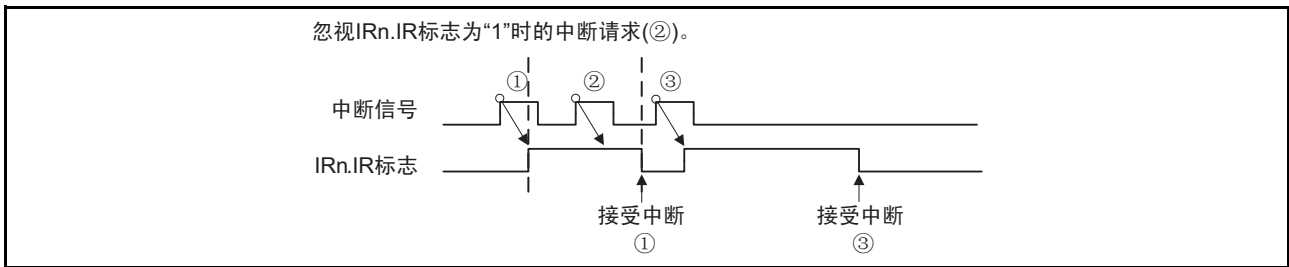


图 14.4 IRn.IR 标志再次被置位的时序

即使在 IRn.IR 标志变为“1”后禁止中断 (通过外围模块的中断允许位禁止中断请求的输出)，IRn.IR 标志也不受影响而保持“1”的状态。禁止中断时的 IRn.IR 标志变化如图 14.5 所示。

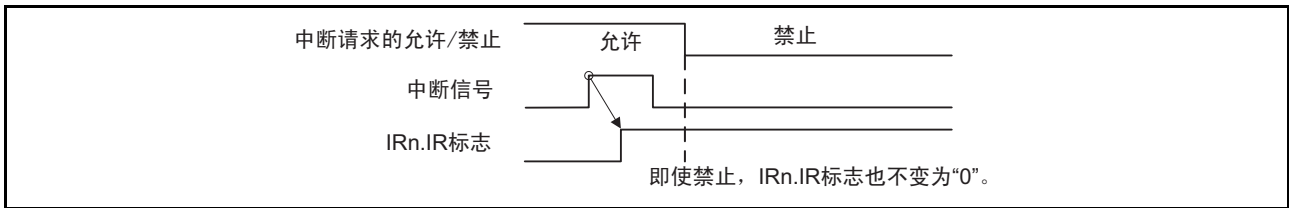


图 14.5 中断请求的禁止和 IRn.IR 标志的关系

14.4.1.2 电平检测的中断状态标志

对外围功能中断和外部引脚中断进行电平检测时的 IRn.IR 标志变化如图 14.6 所示。

在中断信号有效期间，IRn.IR 标志保持“1”。要将 IRn.IR 标志置“0”时，必须将中断发生源的中断请求标志置“0”。必须在确认中断请求发生源的中断请求标志变为“0”并且 IRn.IR 标志变为“0”后结束中断处理程序。

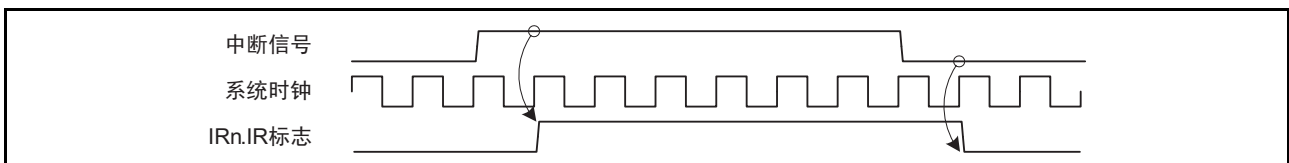


图 14.6 电平检测时的 IRn.IR 标志变化

电平检测的中断处理步骤如图 14.7 所示。

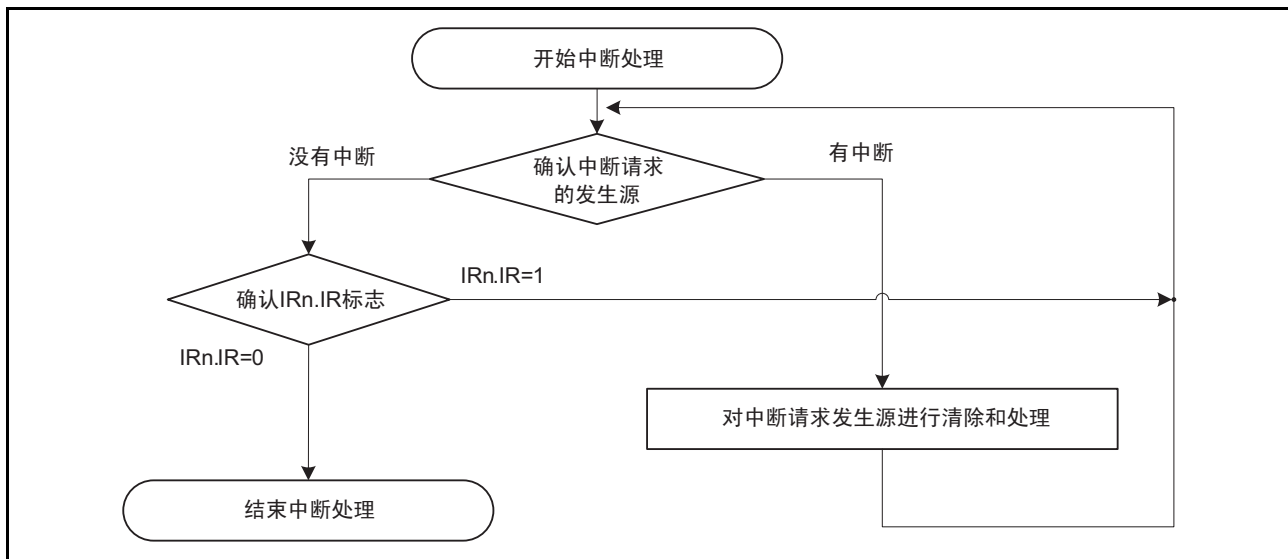


图 14.7 电平检测的中断处理步骤

14.4.2 中断请求的允许 / 禁止

为了允许中断请求，需要进行以下设定：

1. 在外围功能中断的情况下，通过外围模块的中断允许位允许中断请求的输出。
2. 通过 IERm.IENj 位允许中断请求。

如果在中断发生源产生被允许中断输出的中断请求，对应的 IRn.IR 标志就变为“1”。

通过 IERm.IENj 位允许中断请求，将 IRn.IR 标志为“1”的中断请求输出到中断请求目标。通过 IERm.IENj 位禁止中断请求，保留 IRn.IR 标志为“1”的中断请求。

IRn.IR 标志不受 IERm.IENj 位的影响。

禁止中断请求的步骤如下：

1. 将 IERm.IENj 位设定为禁止中断请求。
2. 将外围模块的中断输出允许位设定为禁止，并且通过读已写的寄存器来确认写操作的结束。
3. 根据需要，确认 IRn.IR 标志或者将 IRn.IR 标志置“0”（注1）。

注 1. 如果将 SCI、RSPI、RIIC 的各发送中断或者接收中断从允许状态改为禁止状态，就必须按照上述步骤将 IRn.IR 标志置“0”。详细内容请参照“28. 串行通信接口 (SClC、SCId)”、“29. I²C 总线接口 (RIIC)”和“30. 串行外围接口 (RSPI)”中的各中断的说明。

14.4.3 中断请求目标的选择

给各中断源规定了能设定的中断请求目标，能设定的请求目标如“表 14.3 中断向量表”所示。不能选择表 14.3 中没有记载“○”的中断请求目标。

要通过 IRQ 引脚将 DTC/DMAC 设定为中断请求目标时，必须将 IRQCri.IRQMD[1:0] 位设定为边沿检测。中断请求目标的设定方法如下所示。

(1) DMAC 的启动

必须在 IERm.IENj 位为“0”时对各中断源进行以下设定：

1. 给 DMAC 的各通道准备的 DMAC 启动请求选择寄存器 (DMRSRm) 指定对应的中断源向量号 (注1)。
2. 将 DMAC 对应通道的启动源 (DMACm.DMTMD.DCTG[1:0]) 设定为“01b” (检测中断模块)。
3. 将 DMAC 对应通道的 DMAC 启动允许位 (DMACm.DMCNT.DTE) 置“1”。

必须在上述状态下将 IERm.IENj 位置“1”。

还必须将 DMAC 运行允许位 (DMAST.DMST) 置“1”。各中断源的设定和 DMAC 运行允许位的设定不分先后顺序。

DMAC 的设定步骤请参照“16. DMA 控制器 (DMACA)”的“16.3.7 DMAC 的启动”。

(2) DTC 的启动

必须在 IERm.IENj 位为“0”时对各中断源进行以下设定：

1. 将对应中断源的 DTC 启动允许寄存器的 DTC 传送允许位 (DTCERn.DTCE) 置“1” (注1)。

必须在上述状态下将 IERm.IENj 位置“1”。

还必须将 DTC 模块启动位 (DTCST.DTCST) 置“1”。各中断源的设定和 DTC 模块启动位的设定不分先后顺序。

DTC 的设定步骤请参照“17. 数据传送控制器 (DTCa)”的“17.5 DTC 的设定步骤”。

注 1. 不能给 DTC 启动允许位 (DTCERn.DTCE) 和 DMAC 启动请求选择寄存器 (DMRSRm) 设定相同的中断源，也不能给多个 DMRSRm 设定相同的中断源。

(3) CPU 中断请求

对于中断请求目标既不是 DMAC 也不是 DTC 的中断源，中断请求目标为 CPU。

必须在未设定上述的 DMAC 启动和 DTC 启动的状态下将 IERm.IENj 位置“1”。

将 DMAC 和 DTC 设定为中断请求目标时的运行如表 14.4 所示。

表 14.4 启动 DMAC 或者 DTC 时的运行

中断请求目标	DISEL	剩余的 传送次数	每次请求的运行	IR (注1)	传送后的中断请求目标
DMAC	1	≠0	DMA 传送 → CPU 中断	在接受 CPU 中断时清除。	DMAC
		=0	DMA 传送 → CPU 中断	在接受 CPU 中断时清除。	清除DMACm.DMCNT.DTE位并且转换到 CPU。
	0	≠0	DMA 传送	在开始 DMAC 传送时清除。	DMAC
		=0	DMA 传送 (注2)	在开始 DMAC 传送时清除 (注2)。	清除DMACm.DMCNT.DTE位并且转换到 CPU。
DTC (注3)	1	≠0	DTC 传送 → CPU 中断	在接受 CPU 中断时清除。	DTC
		=0	DTC 传送 → CPU 中断	在接受 CPU 中断时清除。	清除 DTCER.DTCE 位并且转换到 CPU。
	0	≠0	DTC 传送	在读取 DTC 传送信息后开始 DTC 数据传送时清除。	DTC
		=0	DTC 传送 → CPU 中断 (注2)	在接受 CPU 中断时清除 (注2)。	清除 DTCER.DTCE 位并且转换到 CPU。

通过 DMACm.DMCSL.DISEL 位设定 DMAC 的 DISEL，通过 DTC.MRB.DISEL 位设定 DTC 的 DISEL。

注 1. 忽视在 IRn.IR 标志为“1”时再次产生的中断请求 (DTC/DMAC 的启动请求)。

注 2. DISEL 为“0”并且剩余的传送次数为“0”时的运行因 DTC 和 DMAC 而不同。

注 3. 在链传送的情况下，DTC 的传送继续到最后的链传送为止。最后的链传送时的 CPU 中断的有无、IRn.IR 标志的清除、传送后的中断请求目标等，取决于最后的链传送的 DISEL 和剩余的传送次数。有关链传送，请参照“17. 数据传输控制器 (DTCa)”的“表 17.3 链传送的条件”。

要更改中断请求目标时，必须在 IERm.IENj 位为“0”时进行。

要在进行“(1) DMAC 的启动”的设定后传送未结束的状态 (未清除 DMACm.DMCNT.DTE 位的状态) 下更改中断请求目标或者将 DMA 启动源更改为其他中断源时，必须按照以下步骤进行更改：

1. 将要取消的中断源以及作为新启动对象的中断源的 IERm.IENj 位置“0”。
2. 确认 DMAC 的传送状况。如果是在传送过程中，就等待传送结束。
3. 进行“(1) DMAC 的启动”的设定。

要在进行“(2) DTC 的启动”的设定后传送未结束的状态 (未清除 DTCERn.DTCE 位的状态) 下更改中断请求目标或者更改 DTC 的传送设定内容时，就必须按照以下步骤进行更改：

1. 将要取消的中断源以及作为新启动对象的中断源的 IERm.IENj 位置“0”。
2. 确认 DTC 的传送状况。如果是在传送过程中，就等待传送结束。
3. 设定“(2) DTC 的启动”。

14.4.4 优先级的判断

中断控制器判断各中断请求目标的优先级。各中断请求目标的优先级判断方法如下。

(1) 中断请求目标为 CPU 时的优先级判断

被设定为高速中断的中断源的优先级最高，其次优先中断优先级设定位 (IPRn.IPR[3:0]) 中值大的中断源。如果 IPRn.IPR[3:0] 位的值相同并且有多个优先级的中断源，就优先向量号小的中断源。

(2) 中断请求目标为 DTC 时的优先级判断

不受 IPRn.IPR[3:0] 位的影响，优先向量号小的中断源。

(3) 中断请求目标为 DMAC 时的优先级判断

不受 IPRn.IPR[3:0] 位的影响。有关 DMAC 通道的优先级，请参照“16. DMA 控制器 (DMACA)”。

14.4.5 高速中断

高速中断是高速地进行CPU中断处理的功能，只对CPU的中断请求有效，不影响DTC和DMAC的启动请求。

在通过 FIR.FVCT[7:0] 位给要设定为高速中断的中断源设定向量号并且通过 FIR.FIEN 位设定为允许高速中断的状态下，如果产生对应的中断源，就能作为高速中断输出到 CPU。

被设定为高速中断的中断源与 IPRn.IPR[3:0] 位的设定无关，是优先级最高的中断源。

有关高速中断的详细内容，请参照“13. 异常处理”。

14.4.6 数字滤波器

外部中断请求引脚 IRQi (i=0 ~ 7) 和 NMI 引脚中断具有数字滤波器功能。

数字滤波器通过滤波器的采样时钟 (PCLK) 对输入信号进行采样，消除采样周期不足 3 次的脉冲。

当使用 IRQi 引脚的数字滤波器时，必须通过 IRQFLTC0.FCLKSELi[1:0] 位 (i=0 ~ 7) 设定采样频率 (PCLK、PCLK/8、PCLK/32、PCLK/64) 并且将 IRQFLTE0.FLTENi 位置“1” (数字滤波器有效)。

当使用 NMI 引脚中断的数字滤波器时，必须通过 NMIFLTC.NFCLKSEL[1:0] 位设定采样频率 (PCLK、PCLK/8、PCLK/32、PCLK/64) 并且将 NMIFLTE.NFLTEN 位置“1” (数字滤波器有效)。数字滤波器的运行例子如图 14.8 所示。

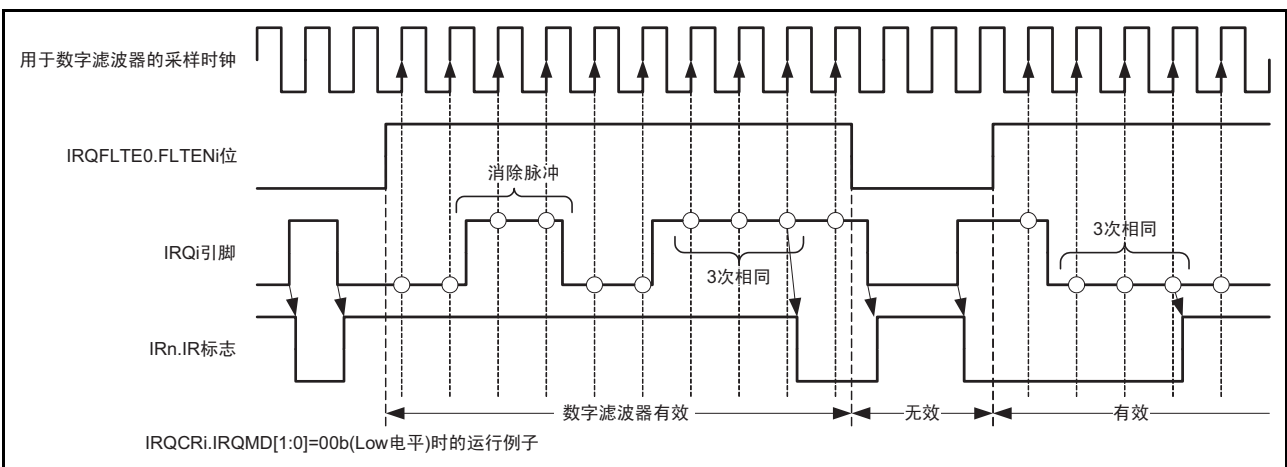


图 14.8 数字滤波器的运行例子

在向软件待机模式转移时，必须将 IRQFLTE0.FLTENi 位和 NMIFLTE.NFLTEN 位都置“0” (数字滤波器无效)。要在从软件待机模式返回后再次使用数字滤波器时，必须将 IRQFLTE0.FLTENi 位或者 NMIFLTE.NFLTEN 位置“1” (数字滤波器有效)。

14.4.7 外部引脚中断

使用外部引脚中断的步骤如下：

1. 将 IERm.IENj 位置“0”（禁止中断请求）。
2. 将 IRQFLTE0.FLTENi 位（i=0~7）置“0”（数字滤波器无效）。
3. 通过 IRQFLTE0.FCLKSELi[1:0] 位设定数字滤波器的采样时钟。
4. 设定 I/O 端口并且进行确认。
5. 通过 IRQCRi.IRQMD[1:0] 位设定检测方法。
6. 将 IRn.IR 标志置“0”（边沿检测的情况）。
7. 将 IRQFLTE0.FLTENi 位置“1”（数字滤波器有效）。
8. 当启动 DMAC 时，设定 DMRSRm.DMRS[7:0] 位；当启动 DTC 时，设定 DTCERn.DTCE 位（在不设定任何位的情况下，为 CPU 中断）。
9. 将 IERm.IENj 位置“1”（允许中断请求）。

14.5 非屏蔽中断的运行说明

非屏蔽中断有 NMI 引脚中断、振荡停止检测中断、WDT 下溢 / 刷新错误中断、IWDT 下溢 / 刷新错误中断、电压监视 1 中断和电压监视 2 中断。非屏蔽中断只限于向 CPU 请求的中断，不能启动 DTC 和 DMAC。在包括高速中断在内的全部中断中，非屏蔽中断是最优先的中断。

非屏蔽中断请求与 CPU 的 PSW.I 位（中断允许位）和 PSW.IPL[3:0] 位（处理器中断优先级）的状态无关，总是被接受。能通过非屏蔽中断状态寄存器（NMISR）确认非屏蔽中断的有无。

必须通过非屏蔽中断处理程序确认 NMISR 寄存器的全部位是否为“0”。

初始状态为“禁止非屏蔽中断”。在使用非屏蔽中断的系统中，必须在程序处理的起始位置按照以下步骤进行设定。

非屏蔽中断的使用步骤：

1. 设定堆栈指针（SP）。
2. 在使用 NMI 引脚时，将 NMIFLTE.NFLTEN 位置“0”（数字滤波器无效）。
3. 在使用 NMI 引脚时，通过 NMIFLTC.NFCLKSEL[1:0] 位设定数字滤波器的采样时钟。
4. 在使用 NMI 引脚时，通过 NMICR.NMIMD 位设定 NMI 引脚的检测方法。
5. 在使用 NMI 引脚时，在给 NMICLR.NMICLR 位写“1”后将 NMISR.NMIST 标志置“0”。
6. 在使用 NMI 引脚时，将 NMIFLTE.NFLTEN 位置“1”（数字滤波器有效）。
7. 通过将非屏蔽中断允许寄存器（NMIER）允许的中断对应位置“1”，允许使用非屏蔽中断。

如果给 NMIER 寄存器写“1”，就忽视以后 NMIER 寄存器的写操作。不能禁止非屏蔽中断，只能通过复位禁止非屏蔽中断。

非屏蔽中断的处理流程请参照“13. 异常处理”。

通过给 NMICLR.NMICLR 位写“1”，NMI 状态标志（NMISR.NMIST）变为“0”。

通过给 NMICLR.OSTCLR 位写“1”，振荡停止检测中断状态标志（NMISR.OSTST）变为“0”。

通过给 NMICLR.WDTCLR 位写“1”，WDT 下溢 / 刷新错误状态标志（NMISR.WDTST）变为“0”。

通过给 NMICLR.IWDTCLR 位写“1”，IWDT 下溢 / 刷新错误状态标志（NMISR.IWDTST）变为“0”。

通过给 NMICLR.LVD1CLR 位写“1”，电压监视 1 中断状态标志（NMISR.LVD1ST）变为“0”。

通过给 NMICLR.LVD2CLR 位写“1”，电压监视 2 中断状态标志（NMISR.LVD2ST）变为“0”。

14.6 从低功耗状态的返回

能用作从睡眠模式、全模块时钟停止模式、软件待机模式返回中断的中断源如“表 14.3 中断向量表”所示。详细内容请参照“11. 低功耗功能”，各低功耗模式中返回对象中断的设定方法如下所示。

14.6.1 从睡眠模式的返回

能通过非屏蔽中断和全部中断源的中断进行返回，返回条件如下：

- 中断
 1. 中断请求目标为 CPU。
 2. 已经通过 IERm.IENj 位允许对应的中断请求。
 3. 高于 CPU.PSW.IPL[3:0] 位的中断优先级。

- 非屏蔽中断

已经通过 NMIER 寄存器允许对应的中断请求。

14.6.2 从全模块时钟停止模式的返回

能通过非屏蔽中断以及能从全模块时钟停止模式返回的中断进行返回，返回条件如下：

- 中断
 1. 是从全模块时钟停止模式返回的中断源。
 2. 中断请求目标为 CPU。
 3. 已经通过 IERm.IENj 位允许对应的中断请求。
 4. 高于 CPU.PSW.IPL[3:0] 位的中断优先级。

- 非屏蔽中断

已经通过 NMIER 寄存器允许对应的中断请求。

14.6.3 从软件待机模式的返回

能通过非屏蔽中断以及能从软件待机模式返回的中断进行返回，返回条件如下：

- 中断
 1. 是从软件待机模式返回的中断源。
 2. 中断请求目标为 CPU。
 3. 已经通过 IERm.IENj 位允许对应的中断请求。
 4. 高于 CPU.PSW.IPL[3:0] 位的中断优先级。

(对于被设定为高速中断的中断源，除了高速中断设定寄存器 (FIR)，还必须将对应的中断优先级 (IPRn) 设定为高于 CPU.PSW.IPL。)

对于不满足上述条件的 IRQ 引脚，在软件待机模式中检测不到时钟停止期间内产生的中断源。

- 非屏蔽中断

已经通过 NMIER 寄存器允许对应的中断请求。

必须按照以下步骤进行软件待机模式的转移或者解除：

1. 必须在向软件待机模式转移前将返回对象的中断源的数字滤波器置为无效 (将 IRQFLTE0.FLTENi 位和 NMIFLTE.NFLTEN 位都置“0”)。
2. 要在从软件待机模式返回后再次使用数字滤波器时，必须将数字滤波器置为有效 (将 IRQFLTE0.FLTENi 位和 NMIFLTE.NFLTEN 位都置“1”)。

14.7 使用时的注意事项

14.7.1 使用非屏蔽中断时的 WAIT 指令的注意事项

必须在确认 NMISR 寄存器的全部状态标志都为“0”后发行 WAIT 指令。

15. 总线

15.1 概要

总线的规格和各种总线的地址对应表分别如表 15.1 和表 15.2 所示，总线结构图如图 15.1 所示。

表 15.1 总线的规格

总线的种类		内容
CPU 总线	指令总线	<ul style="list-style-type: none"> • 连接 CPU（指令）。 • 连接内部存储器（内部 RAM 和内部 ROM）。 • 与系统时钟（ICLK）同步运行。
	操作数总线	<ul style="list-style-type: none"> • 连接 CPU（操作数）。 • 连接内部存储器（内部 RAM 和内部 ROM）。 • 与系统时钟（ICLK）同步运行。
存储器总线	存储器总线 1	• 连接内部 RAM。
	存储器总线 2	• 连接内部 ROM。
内部主总线	内部主总线 1	<ul style="list-style-type: none"> • 连接 CPU。 • 与系统时钟（ICLK）同步运行。
	内部主总线 2	<ul style="list-style-type: none"> • 连接 DMAC 和 DTC。 • 连接内部存储器（内部 RAM 和内部 ROM）。 • 与系统时钟（ICLK）同步运行。
内部外围总线	内部外围总线 1	<ul style="list-style-type: none"> • 连接外围功能（DMAC、中断控制器和总线错误监视部）。 • 与系统时钟（ICLK）同步运行。
	内部外围总线 2	<ul style="list-style-type: none"> • 连接外围功能（内部外围总线 1 以外的外围功能）。 • 与外围模块时钟（PCLKB、PCLKD（注 1））同步运行。
	内部外围总线 6	<ul style="list-style-type: none"> • 连接内部 ROM(P/E) 和 E2 数据闪存。 • 与 FlashIF 时钟（FCLK）同步运行。
外部总线	CS 区域	<ul style="list-style-type: none"> • 连接外部设备。 • 与外部总线时钟（BCLK）同步运行。

注 1. 外围模块时钟（PCLKD）是 S12AD 的运行时钟。

P/E: 编程 / 擦除

BCLK（外部总线时钟）：这是最大 25MHz 的时钟。CSC（CS 区域控制器）与 BCLK 同步运行。

BCLK 引脚输出：默认的频率和 BCLK 的频率相同，能通过外部总线时钟控制寄存器的 BCLK 引脚输出选择位（BCKCR.BCLKDIV）选择 BCLK 的 2 分频，详细内容请参照“9. 时钟发生电路”。

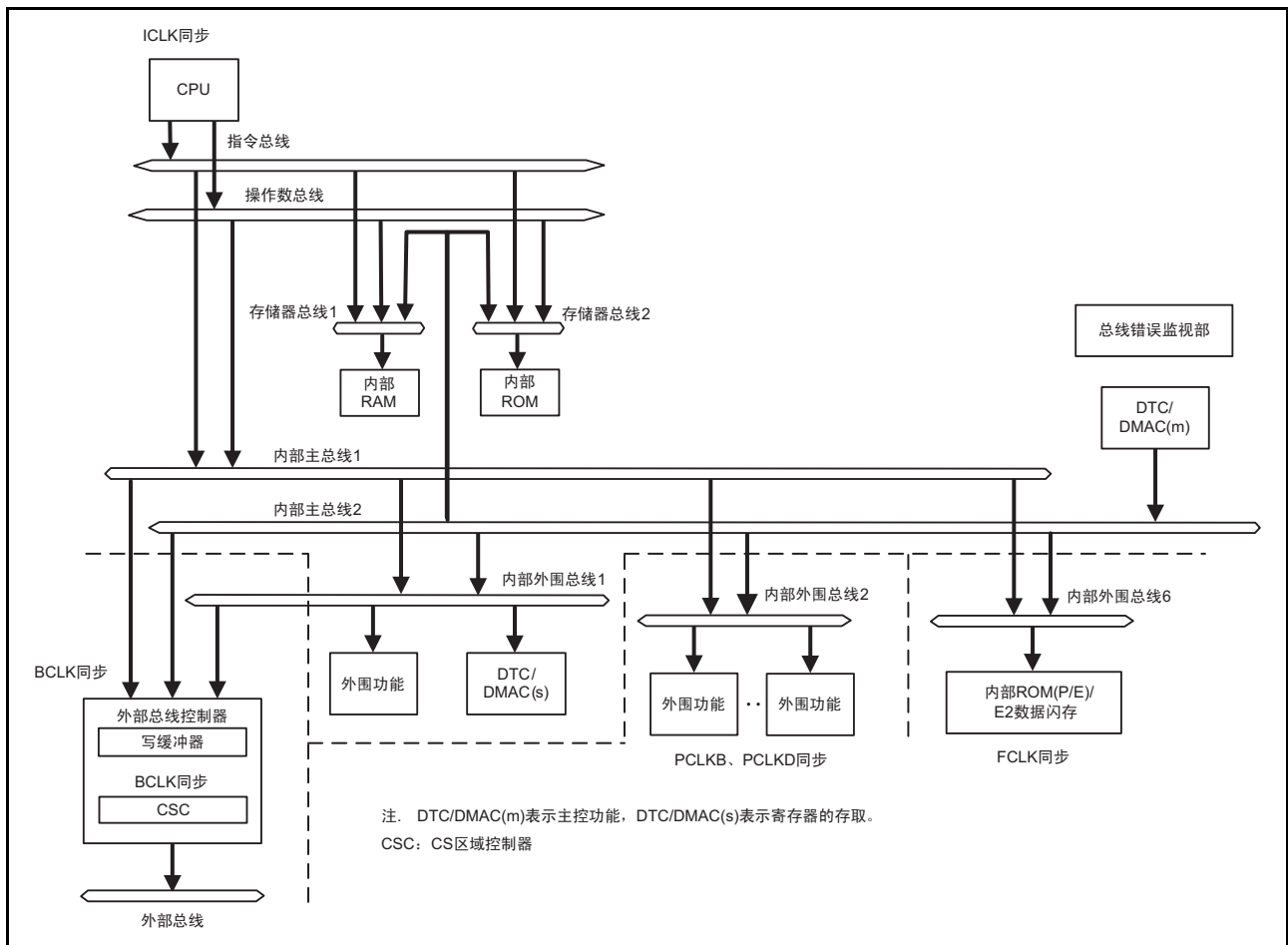


图 15.1 总线结构图

表 15.2 各种总线的地址对应表

地址	总线		内容	
	内部 ROM 模式		内部 ROM 模式	
	有效	无效	有效	无效
0000 0000h ~ 0001 FFFFh	存储器总线 1		内部 RAM	
0002 0000h ~ 0007 FFFFh			保留区	
0008 0000h ~ 0008 7FFFh	内部外围总线 1		外围 I/O 寄存器	
0008 8000h ~ 0009 FFFFh	内部外围总线 2			
0010 0000h ~ 00FF FFFFh	内部外围总线 6	保留区	E2 数据闪存、FCU-RAM、内部 ROM (改写专用)	保留区
0100 0000h ~ 07FF FFFFh	外部总线		外部地址空间 (CS1 ~ CS3)	
0800 0000h ~ 0FFF FFFFh			保留区	
1000 0000h ~ 7FFF FFFFh	保留区		保留区	
8000 0000h ~ FFFF FFFFh	存储器总线 2	保留区	内部 ROM (只读)	保留区
FF00 0000h ~ FFFF FFFFh		外部总线		外部地址空间 (CS0)

15.2 总线说明

15.2.1 CPU 总线

CPU 总线有指令总线和操作数总线，连接内部主总线 1。指令总线用于 CPU 的取指令，操作数总线用于 CPU 的操作数存取。

指令总线和操作数总线连接内部 RAM 和内部 ROM，能不通过内部主总线 1 而由 CPU 直接存取。内部 ROM 为只读区域，能由 CPU 直接存取，但是必须通过内部外围总线进行编程和擦除。

内部主总线 1 对取指令和操作数的总线权请求进行仲裁，优先级为操作数 > 取指令。

如果取指令和操作数存取的请求是针对不同的总线（存储器总线 1、存储器总线 2 和内部主总线 1），就能同时进行各自的总线存取。例如，内部 ROM 和内部 RAM、内部 ROM 和外部存取等能并行运行。

15.2.2 存储器总线

存储器总线有存储器总线 1 和存储器总线 2，存储器总线 1 连接内部 RAM，存储器总线 2 连接内部 ROM。存储器总线 1 和存储器总线 2 对 CPU 总线（取指令和操作数）和内部主总线 2 的总线权请求进行仲裁。

能分别通过总线优先级控制寄存器的存储器总线 1（内部 RAM）的优先级控制位（BUSPRI.BPRA[1:0]）和存储器总线 2（内部 ROM）的优先级控制位（BUSPRI.BPRO[1:0]）设定 2 条总线的优先级。在优先级固定的情况下，2 条总线的优先级为内部主总线 2 > CPU 总线（操作数 > 取指令）；在优先级交替的情况下，内部主总线 2 和 CPU 总线中已接受总线请求的一方的优先级变低。

15.2.3 内部主总线

内部主总线由 CPU 使用的总线（内部主总线 1）以及其他总线主控（DTC 和 DMAC）使用的总线（内部主总线 2）构成。

内部主总线 1 对取指令和操作数的总线权请求进行仲裁，优先级为操作数 > 取指令。

内部主总线 2 对 DTC 和 DMAC 的总线权请求进行仲裁，优先级为 DMAC > DTC，如表 15.3 所示。

对于 DTC 和 DMAC，只有接受了启动请求的一方才能请求总线权。DTC 和 DMAC 的启动请求优先级为 DMAC0 > DMAC1 > DMAC2 > DMAC3 > DTC，与 BUSPRI 寄存器的设定无关。

如果 CPU 和其他总线主控的请求是针对不同的总线（内部存储器、内部外围总线 1、内部外围总线 2、内部外围总线 6 和外部总线），就能同时进行各自的总线存取。

如果通过 CPU 执行 XCHG 指令，就在 XCHG 指令进行的总线存取结束前不接受 CPU 以外的总线存取，与总线优先级控制寄存器（BUSPRI）的设定无关，并且在读以及回写 DTC 的传送信息过程中也不接受 DTC 以外的总线存取。

表 15.3 总线主控优先级

优先级	总线主控
高	DMAC
↑	DTC
低	CPU

15.2.4 内部外围总线

连接内部外围总线的外围功能如表 15.4 所示。

表 15.4 连接内部外围总线的外围功能

总线的种类	外围功能
内部外围总线 1	DMAC、中断控制器、总线错误监视部
内部外围总线 2	内部外围总线 1 以外的外围功能
内部外围总线 6	内部 ROM(P/E)/E2 数据闪存、FCU-RAM

内部外围总线 1、2、6 分别对 CPU（内部主总线 1）和其他总线主控（内部主总线 2）的总线权请求进行仲裁。

能通过总线优先级控制寄存器（BUSPRI）设定 2 条内部总线的优先级，能通过内部外围总线 1 的优先级控制位（BUSPRI.BPIB[1:0]）、内部外围总线 2、3 的优先级控制位（BUSPRI.BPGB[1:0]）以及内部外围总线 6 的优先级控制位（BUSPRI.BPFB[1:0]）设定各总线的优先级。在优先级固定的情况下，优先级为内部主总线 2 > 内部主总线 1；在优先级交替的情况下，内部主总线 1 和内部主总线 2 中已接受总线请求的一方的优先级变低。

必须注意：如果 BUSPRI 寄存器的设定不同，接受请求的顺序就可能发生变化（参照图 15.2）。如图 15.2 所示，当接受的总线请求的优先级低时，优先级不变。

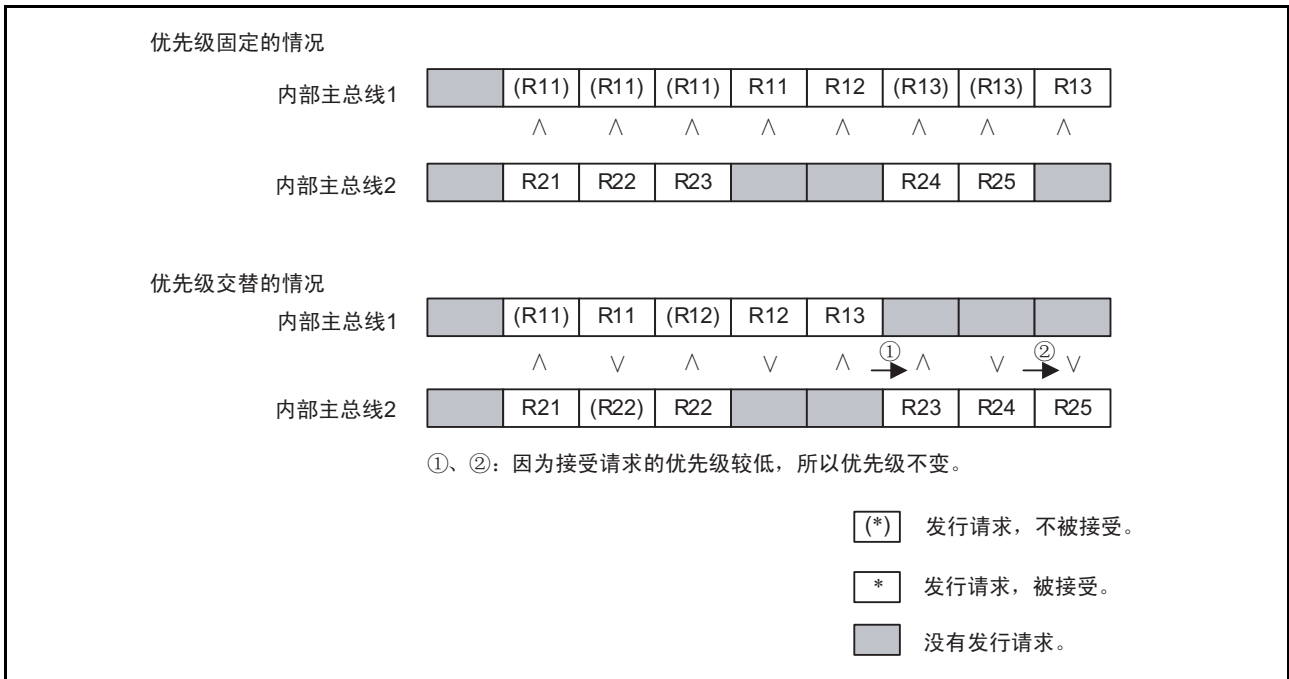


图 15.2 内部外围总线优先级

内部外围总线具有写缓冲器功能，在进行写存取时，能不等运行结束就接受后面的存取。但是，在从相同的总线主控进行存取时，如果写存取后面的存取是针对不同的内部外围总线，后面的存取就要等到写存取结束。必须注意：如果在 CPU 对内部外围总线进行写存取后读内部存储器，因为不等运行结束就接受后面的存取，所以存取的顺序有可能发生变化（参照图 15.3）。

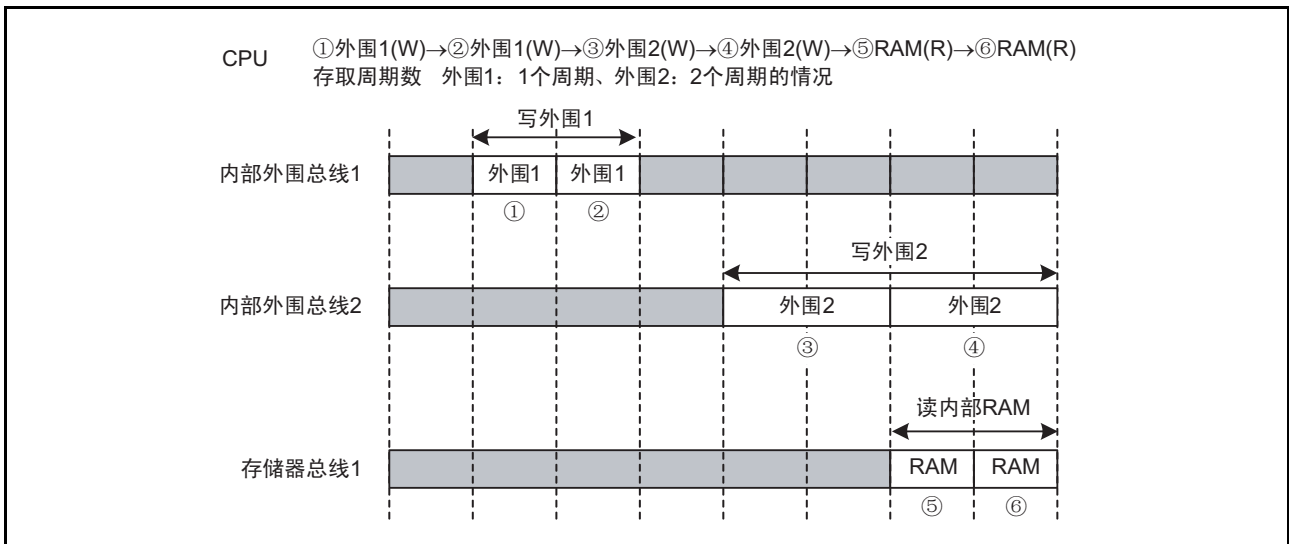


图 15.3 写缓冲器功能

15.2.5 外部总线

外部总线的规格如表 15.5 所示。

外部总线控制器对内部主总线 1 和内部主总线 2 的总线权请求进行仲裁。

能通过总线优先级控制寄存器的外部总线优先级控制位 (BUSPRI.BPEB[1:0]) 设定 2 条总线的优先级。在优先级固定的情况下, 优先级为内部主总线 2 > 内部主总线 1; 在优先级交替的情况下, 内部主总线 1 和内部主总线 2 中已接受总线请求的一方的优先级变低。

必须注意: 如果更改总线优先级控制寄存器的设定, 接受请求的顺序就可能发生变化 (参照图 15.4)。如图 15.4 所示, 当接受的总线请求的优先级低时, 优先级不变。

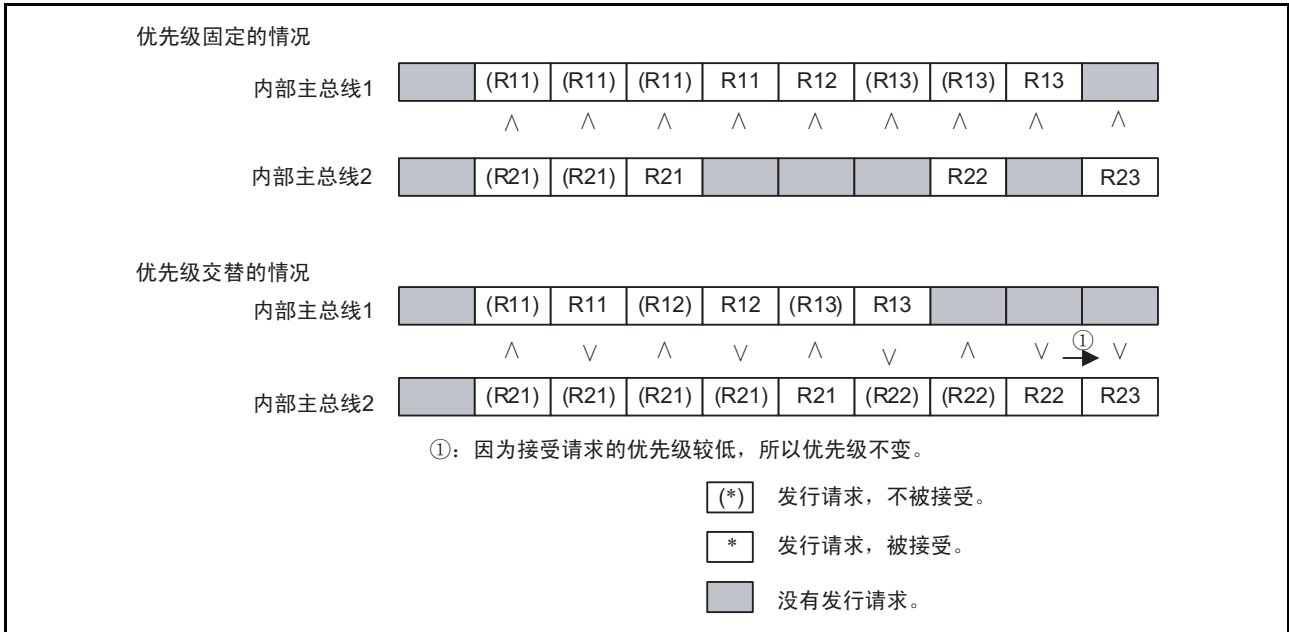


图 15.4 内部外围总线优先级

表 15.5 外部总线的规格

项目	内容
外部地址空间	<ul style="list-style-type: none"> 将外部地址空间分成 4 个 CS 区域 (CS0 ~ CS3) 进行管理。 各区域能输出片选。 各区域能选择总线宽度。 分离总线: 可选择 8 位总线空间或者 16 位总线空间。 地址数据多路复用总线: 可选择 8 位总线空间或者 16 位总线空间。 <ul style="list-style-type: none"> 各区域能设定字节序。
CS 区域控制器	<ul style="list-style-type: none"> 能插入恢复周期。 最多插入 15 个读恢复周期。 最多插入 15 个写恢复周期。 周期等待功能: 最多等待 31 个周期 (页面存取: 最多等待 7 个周期)。 等待控制 能设定片选信号 (CS0# ~ CS3#) 的有效时序或者无效时序。 能设定读信号 (RD#) 和写信号 (WR0#/WR#, WR1#) 的有效时序。 能设定数据输出的开始时序或者结束时序。 写存取模式: 1 次写选通模式或者字节选通模式 各区域能设定分离总线或者地址数据多路复用总线。
写缓冲器功能	在将总线主控的写数据写到写缓冲器时, 结束总线主控的写存取。
频率	CS 区域控制器 (CSC) 与 BCLK 同步运行。

外部总线的输入 / 输出引脚如表 15.6 所示。

表 15.6 外部总线的输入 / 输出引脚

引脚名	输入 / 输出	功能
A23 ~ A0 (注 1)	输出	地址的输出引脚
D15 ~ D0	输入 / 输出	数据的输入 / 输出引脚 当设定 16 位总线空间时, D15 ~ D0 有效; 当设定 8 位总线空间时, D7 ~ D0 有效。
BC0# (注 1)	输出	此选通信号表示在 1 次写选通模式中正在存取外部地址空间 (BC0# 信号为 Low 电平的情况) 并且 D7 ~ D0 有效。 如果设定 8 位总线空间, 就总是输出 Low 电平, 与写存取模式无关。
BC1#	输出	此选通信号表示在 1 次写选通模式中正在存取外部地址空间 (BC1# 信号为 Low 电平的情况) 并且 D15 ~ D8 有效。 如果设定 8 位总线空间, 就不使用此选通信号。
CS0#	输出	区域 0 (CS0) 的片选信号
CS1#	输出	区域 1 (CS1) 的片选信号
CS2#	输出	区域 2 (CS2) 的片选信号
CS3#	输出	区域 3 (CS3) 的片选信号
RD#	输出	此选通信号表示正在读外部地址空间 (CS0 ~ CS3)。
WR0#/WR# (注 2)	输出	WR0# 信号是表示在字节选通模式中正在写外部地址空间 (WR0# 信号为 Low 电平的情况) 并且 D7 ~ D0 有效的选通信号。 WR# 信号是表示在 1 次写选通模式中正在写外部地址空间的选通信号。 如果设定 8 位总线空间, 就在进行写存取时输出 Low 电平, 与写存取模式无关。
WR1#	输出	此选通信号表示在字节选通模式中正在写外部地址空间 (WR1# 信号为 Low 电平的情况) 并且 D15 ~ D8 有效。 在 1 次写选通模式中, 此选通信号无效。 如果设定 8 位总线空间, 就不使用此选通信号。
ALE	输出	选择地址数据多路复用总线时的地址锁存信号
WAIT#	输入	存取外部地址空间 (CS0 ~ CS3) 时的等待请求信号 (Low 电平: 等待请求)

注 1. A0 引脚和 BC0# 引脚兼用, 在字节选通模式中, 各区域的 A0 引脚有效; 在 1 次写选通模式中, 各区域的 BC0# 引脚有效。但是, 在 1 次写选通模式中, 禁止设定 8 位外部总线宽度。有关其他的兼用引脚, 请参照“19. I/O 端口”。

注 2. WR0# 引脚和 WR# 引脚是同一个信号, WR# 引脚表示 1 次写选通模式中的 WR0#。

15.2.6 并行运行

能在各总线主控存取不同的从属模块时并行运行。例如，在 CPU 从内部 ROM 取指令以及从内部 RAM 存取操作数的过程中，DMAC 能进行外围总线和外部总线之间的传送。并行运行的例子如图 15.5 所示。此例中，CPU 能使用指令总线和操作数总线同时存取内部 ROM 和内部 RAM，而且在 CPU 存取内部 ROM 和内部 RAM 的过程中，DMAC 能使用内部主总线 2 同时存取外围总线或者外部总线。

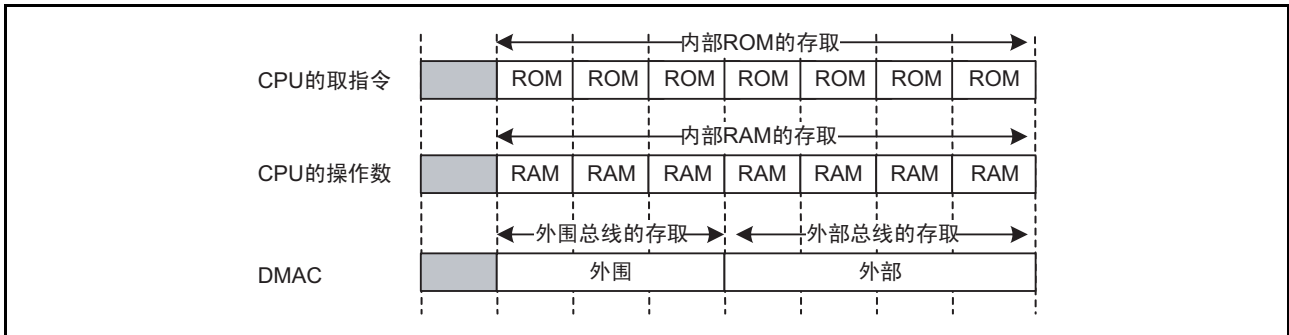


图 15.5 并行运行的例子

15.2.7 总线的设定

1. 通过 CSn 模式寄存器 (CSnMOD)、CSn 等待控制寄存器 1 (CSnWCR1)、CSn 等待控制寄存器 2 (CSnWCR2)、CSn 控制寄存器 (CSnCR)、CSn 恢复周期设定寄存器 (CSnREC)、CS 恢复周期插入允许寄存器 (CSRECEN)、总线错误监视允许寄存器 (BEREN) 和总线优先级控制寄存器 (BUSPRI) 设定外部总线的模式。
2. 通过 CS 输出允许寄存器 (PFCSE)、地址输出允许寄存器 0 (PFAOE0)、地址输出允许寄存器 1 (PFAOE1)、外部总线控制寄存器 0 (PFBCR0) 和外部总线控制寄存器 1 (PFBCR1) 设定引脚。
3. 将使用的引脚设定为输入端口。
4. 将系统控制寄存器 0 (SYSCR0) 的外部总线允许位 (EXBE) 置“1” (外部总线有效)。

15.2.8 限制事项

(1) 禁止存取跨地址空间的多个区域

禁止在 1 次存取中存取跨地址空间的多个区域，否则不保证运行。1 个字或者长字的存取不能隔着地址空间的各区域边界而跨 2 个区域。

(2) 有关 RMPA 指令和字符串操作指令的限制事项

- (a) 外部空间各区域有字节序转换功能 (只限于数据)，但是禁止将 RMPA 指令和字符串操作指令的操作对象数据分配到设定了不同于芯片字节序的区域，否则不保证运行。要将 RMPA 指令和字符串操作指令的操作对象数据分配到外部空间时，必须分配到设定了和芯片字节序相同字节序的区域。
- (b) 禁止将 RMPA 指令和字符串操作指令的操作对象数据分配到 I/O 寄存器，否则不保证运行。

15.3 寄存器说明

15.3.1 CSn 控制寄存器 (CSnCR) (n=0 ~ 3)

地址 CS0CR 0008 3802h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	MPXEN	—	—	—	EMODE	—	—	BSIZE[1:0]	—	—	—	—	EXENB
复位后的值	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1

地址 CS1CR 0008 3812h、CS2CR 0008 3822h、CS3CR 0008 3832h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	MPXEN	—	—	—	EMODE	—	—	BSIZE[1:0]	—	—	—	—	EXENB
复位后的值	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	EXENB	运行允许位	0: 禁止运行 1: 允许运行	R/W (注 1)
b3-b1	—	保留位	读写值都为“0”。	R/W
b5-b4	BSIZE[1:0]	外部总线宽度选择位	b5 b4 0 0: 设定为 16 位总线空间 0 1: 不能设定 1 0: 设定为 8 位总线空间 1 1: 不能设定	R/W (注 2)
b7-b6	—	保留位	读写值都为“0”。	R/W
b8	EMODE	字节序模式指定位	0: 区域 n 的字节序模式和运行模式的字节序相同 (n=0 ~ 3) 1: 区域 n 的字节序模式和运行模式的字节序不同 (n=0 ~ 3)	R/W
b11-b9	—	保留位	读写值都为“0”。	R/W
b12	MPXEN	地址 / 数据多路复用 I/O 接口选择位	0: 区域 n 为分离总线接口 1: 区域 n 为地址 / 数据多路复用 I/O 接口	R/W
b15-b13	—	保留位	读写值都为“0”。	R/W

注 1. CS0CR.EXENB 位的初始值为“1”，CSnCR.EXENB 位 (n=1 ~ 3) 的初始值为“0”。

注 2. CS0CR.BSIZE[1:0] 位的初始值为“10b”。

复位后，只能写 1 次此寄存器。如果写 2 次或者 2 次以上，就不保证运行。

EXENB 位 (运行允许位)

此位设定允许或者禁止各区域的运行。

复位后，只允许 (“1”) 区域 0 (CS0) 的运行，禁止 (“0”) 其他区域的运行。

如果存取被设定为禁止运行的区域，就不存取外部总线。此时，如果总线错误监视允许寄存器的非法地址存取检测允许位 (BEREN.IGAEN=1) 被设定为允许检测，就会发生非法地址错误。

BSIZE[1:0] 位（外部总线宽度选择位）

这些位设定各区域的数据总线宽度。

复位后，区域 0（CS0）的数据总线宽度为 8 位地址空间。

如果通过 MPXEN 位设定地址 / 数据多路复用 I/O 接口，就禁止设定为 32 位总线空间，否则不保证运行。

EMODE 位（字节序模式指定位）

此位设定各区域的字节序。

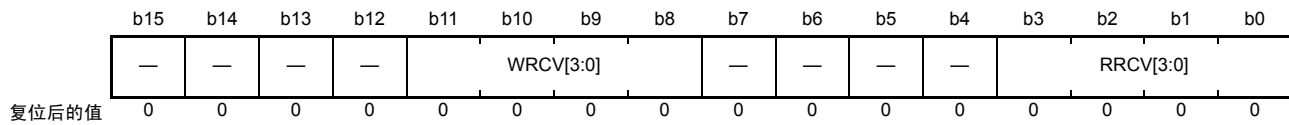
不能将指令代码分配到设定了和芯片字节序不同字节序的各区域。要将指令代码分配到外部空间时，必须分配到设定了和芯片字节序相同字节序的区域。

MPXEN 位（地址 / 数据多路复用 I/O 接口选择位）

此位设定各区域的总线接口。

15.3.2 CSn 恢复周期设定寄存器 (CSnREC) (n=0 ~ 3)

地址 CS0REC 0008 380Ah、CS1REC 0008 381Ah、CS2REC 0008 382Ah、CS3REC 0008 383Ah



位	符号	位名	功能	R/W
b3-b0	RRCV[3:0]	读恢复周期设定位	b3 b0 0000: 不插入恢复周期 0001: 插入 1 个恢复周期 0010: 插入 2 个恢复周期 0011: 插入 3 个恢复周期 0100: 插入 4 个恢复周期 0101: 插入 5 个恢复周期 0110: 插入 6 个恢复周期 0111: 插入 7 个恢复周期 1000: 插入 8 个恢复周期 1001: 插入 9 个恢复周期 1010: 插入 10 个恢复周期 1011: 插入 11 个恢复周期 1100: 插入 12 个恢复周期 1101: 插入 13 个恢复周期 1110: 插入 14 个恢复周期 1111: 插入 15 个恢复周期	R/W
b7-b4	—	保留位	读写值都为“0”。	R/W
b11-b8	WRCV[3:0]	写恢复周期设定位	b11 b8 0000: 不插入恢复周期 0001: 插入 1 个恢复周期 0010: 插入 2 个恢复周期 0011: 插入 3 个恢复周期 0100: 插入 4 个恢复周期 0101: 插入 5 个恢复周期 0110: 插入 6 个恢复周期 0111: 插入 7 个恢复周期 1000: 插入 8 个恢复周期 1001: 插入 9 个恢复周期 1010: 插入 10 个恢复周期 1011: 插入 11 个恢复周期 1100: 插入 12 个恢复周期 1101: 插入 13 个恢复周期 1110: 插入 14 个恢复周期 1111: 插入 15 个恢复周期	R/W
b15-b12	—	保留位	读写值都为“0”。	R/W

CSnREC 寄存器设定外部地址空间各区域的读写存取后的恢复周期数。复位后，只能写 1 次此寄存器。如果写 2 次或者 2 次以上，就不保证运行。

在上次总线存取为分离总线的情况下，当通过 CS 恢复周期插入允许寄存器的分离总线恢复周期插入允许位 (CSRECEN.RCVENj) (j=0 ~ 7) 设定为允许插入恢复周期时，CSnREC 寄存器有效；在上次总线存取为地址 / 数据多路复用总线的情况下，当通过 CS 恢复周期插入允许寄存器的多路复用总线恢复周期插入允许位 (CSRECEN.RCVENMj) 设定为允许插入恢复周期时，CSnREC 寄存器有效。

RRCV[3:0] 位 (读恢复周期设定位)

这些位设定在对外部总线进行读存取后要插入的恢复周期数。

如果设定为允许插入恢复周期并且设定“0000b”以外的值，就在以下情况下插入 1 ~ 15 个恢复周期。

- 在对外部总线进行读存取后有相同区域的外部总线的读存取时
- 在对外部总线进行读存取后有不同区域的外部总线的读存取时
- 在对外部总线进行读存取后有相同区域的外部总线的写存取时
- 在对外部总线进行读存取后有不同区域的外部总线的写存取时

WRCV[3:0] 位 (写恢复周期设定位)

这些位设定在对外部总线进行写存取后要插入的恢复周期数。

如果设定为允许插入恢复周期并且设定“0000b”以外的值，就在以下情况下插入 1 ~ 15 个恢复周期。

- 在对外部总线进行写存取后有相同区域的外部总线的读存取时
- 在对外部总线进行写存取后有不同区域的外部总线的读存取时
- 在对外部总线进行写存取后有相同区域的外部总线的写存取时
- 在对外部总线进行写存取后有不同区域的外部总线的写存取时

15.3.3 CS 恢复周期插入允许寄存器 (CSRECEN)

地址 0008 3880h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RCVEN M7	RCVEN M6	RCVEN M5	RCVEN M4	RCVEN M3	RCVEN M2	RCVEN M1	RCVEN M0	RCVEN7	RCVEN6	RCVEN5	RCVEN4	RCVEN3	RCVEN2	RCVEN1	RCVEN0
复位后的值	0	0	1	1	1	1	1	0	0	0	1	1	1	1	1	0

位	符号	位名	功能	R/W
b0	RCVEN0	分离总线的恢复周期插入允许 0 位	0: 禁止插入恢复周期 1: 允许插入恢复周期	R/W
b1	RCVEN1	分离总线的恢复周期插入允许 1 位	0: 禁止插入恢复周期 1: 允许插入恢复周期	R/W
b2	RCVEN2	分离总线的恢复周期插入允许 2 位	0: 禁止插入恢复周期 1: 允许插入恢复周期	R/W
b3	RCVEN3	分离总线的恢复周期插入允许 3 位	0: 禁止插入恢复周期 1: 允许插入恢复周期	R/W
b4	RCVEN4	分离总线的恢复周期插入允许 4 位	0: 禁止插入恢复周期 1: 允许插入恢复周期	R/W
b5	RCVEN5	分离总线的恢复周期插入允许 5 位	0: 禁止插入恢复周期 1: 允许插入恢复周期	R/W
b6	RCVEN6	分离总线的恢复周期插入允许 6 位	0: 禁止插入恢复周期 1: 允许插入恢复周期	R/W
b7	RCVEN7	分离总线的恢复周期插入允许 7 位	0: 禁止插入恢复周期 1: 允许插入恢复周期	R/W
b8	RCVENM0	多路复用总线的恢复周期插入允许 0 位	0: 禁止插入恢复周期 1: 允许插入恢复周期	R/W
b9	RCVENM1	多路复用总线的恢复周期插入允许 1 位	0: 禁止插入恢复周期 1: 允许插入恢复周期	R/W
b10	RCVENM2	多路复用总线的恢复周期插入允许 2 位	0: 禁止插入恢复周期 1: 允许插入恢复周期	R/W
b11	RCVENM3	多路复用总线的恢复周期插入允许 3 位	0: 禁止插入恢复周期 1: 允许插入恢复周期	R/W
b12	RCVENM4	多路复用总线的恢复周期插入允许 4 位	0: 禁止插入恢复周期 1: 允许插入恢复周期	R/W
b13	RCVENM5	多路复用总线的恢复周期插入允许 5 位	0: 禁止插入恢复周期 1: 允许插入恢复周期	R/W
b14	RCVENM6	多路复用总线的恢复周期插入允许 6 位	0: 禁止插入恢复周期 1: 允许插入恢复周期	R/W
b15	RCVENM7	多路复用总线的恢复周期插入允许 7 位	0: 禁止插入恢复周期 1: 允许插入恢复周期	R/W

CSRECEN 寄存器是设定允许或者禁止在读写存取后插入恢复周期的寄存器。复位后，只能写 1 次此寄存器。如果写 2 次或者 2 次以上，就不保证运行。

RCVEN0 位（分离总线的恢复周期插入允许 0 位）

在对外部总线进行读存取后有相同区域的外部总线的读存取的情况下，此位设定允许（“1”）或者禁止（“0”）插入读恢复周期。

RCVEN1 位（分离总线的恢复周期插入允许 1 位）

在对外部总线进行读存取后有不同区域的外部总线的读存取的情况下，此位设定允许（“1”）或者禁止（“0”）插入读恢复周期。

RCVEN2 位（分离总线的恢复周期插入允许 2 位）

在对外部总线进行读存取后有相同区域的外部总线的写存取的情况下，此位设定允许（“1”）或者禁止（“0”）插入读恢复周期。

RCVEN3 位（分离总线的恢复周期插入允许 3 位）

在对外部总线进行读存取后有不同区域的外部总线的写存取的情况下，此位设定允许（“1”）或者禁止（“0”）插入读恢复周期。

RCVEN4 位（分离总线的恢复周期插入允许 4 位）

在对外部总线进行写存取后有相同区域的外部总线的读存取的情况下，此位设定允许（“1”）或者禁止（“0”）插入写恢复周期。

RCVEN5 位（分离总线的恢复周期插入允许 5 位）

在对外部总线进行写存取后有不同区域的外部总线的读存取的情况下，此位设定允许（“1”）或者禁止（“0”）插入写恢复周期。

RCVEN6 位（分离总线的恢复周期插入允许 6 位）

在对外部总线进行写存取后有相同区域的外部总线的写存取的情况下，此位设定允许（“1”）或者禁止（“0”）插入写恢复周期。

RCVEN7 位（分离总线的恢复周期插入允许 7 位）

在对外部总线进行写存取后有不同区域的外部总线的写存取的情况下，此位设定允许（“1”）或者禁止（“0”）插入写恢复周期。

RCVENM0 位（多路复用总线的恢复周期插入允许 0 位）

在对外部总线进行读存取后有相同区域的外部总线的读存取的情况下，此位设定允许（“1”）或者禁止（“0”）插入读恢复周期。

RCVENM1 位（多路复用总线的恢复周期插入允许 1 位）

在对外部总线进行读存取后有不同区域的外部总线的读存取的情况下，此位设定允许（“1”）或者禁止（“0”）插入读恢复周期。

RCVENM2 位（多路复用总线的恢复周期插入允许 2 位）

在对外部总线进行读存取后有相同区域的外部总线的写存取的情况下，此位设定允许（“1”）或者禁止（“0”）插入读恢复周期。

RCVENM3 位（多路复用总线的恢复周期插入允许 3 位）

在对外部总线进行读存取后有不同区域的外部总线的写存取的情况下，此位设定允许（“1”）或者禁止（“0”）插入读恢复周期。

RCVENM4 位（多路复用总线的恢复周期插入允许 4 位）

在对外部总线进行写存取后有相同区域的外部总线的读存取的情况下，此位设定允许（“1”）或者禁止（“0”）插入写恢复周期。

RCVENM5 位（多路复用总线的恢复周期插入允许 5 位）

在对外部总线进行写存取后有不同区域的外部总线的读存取的情况下，此位设定允许（“1”）或者禁止（“0”）插入写恢复周期。

RCVENM6 位（多路复用总线的恢复周期插入允许 6 位）

在对外部总线进行写存取后有相同区域的外部总线的写存取的情况下，此位设定允许（“1”）或者禁止（“0”）插入写恢复周期。

RCVENM7 位（多路复用总线的恢复周期插入允许 7 位）

在对外部总线进行写存取后有不同区域的外部总线的写存取的情况下，此位设定允许（“1”）或者禁止（“0”）插入写恢复周期。

表 15.7 恢复周期的插入

存取的种类	外部地址空间	恢复周期的插入	对应位 (分离 / 多路复用)
读存取后的读存取	相同区域	插入 RRCV[3:0] 位设定的周期数。	RCVEN0/RCVENM0
	不同区域	插入 RRCV[3:0] 位设定的周期数。	RCVEN1/RCVENM1
读存取后的写存取	相同区域	插入 RRCV[3:0] 位设定的周期数。	RCVEN2/RCVENM2
	不同区域	插入 RRCV[3:0] 位设定的周期数。	RCVEN3/RCVENM3
写存取后的读存取	相同区域	插入 WRCV[3:0] 位设定的周期数。	RCVEN4/RCVENM4
	不同区域	插入 WRCV[3:0] 位设定的周期数。	RCVEN5/RCVENM5
写存取后的写存取	相同区域	插入 WRCV[3:0] 位设定的周期数。	RCVEN6/RCVENM6
	不同区域	插入 WRCV[3:0] 位设定的周期数。	RCVEN7/RCVENM7

15.3.4 CSn 模式寄存器 (CSnMOD) (n=0 ~ 3)

地址 CS0MOD 0008 3002h、CS1MOD 0008 3012h、CS2MOD 0008 3022h、CS3MOD 0008 3032h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PRMOD	—	—	—	—	—	PWENB	PRENB	—	—	—	—	EWENB	—	—	WRMOD
复位后的值	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	WRMOD	写存取模式选择位	0: 字节选通模式 1: 1次写选通模式	R/W
b2-b1	—	保留位	读写值都为“0”。	R/W
b3	EWENB	外部等待允许位	0: 禁止外部等待 1: 允许外部等待	R/W
b7-b4	—	保留位	读写值都为“0”。	R/W
b8	PRENB	页面读存取允许位	0: 禁止页面读存取 1: 允许页面读存取	R/W
b9	PWENB	页面写存取允许位	0: 禁止页面写存取 1: 允许页面写存取	R/W
b14-b10	—	保留位	读写值都为“0”。	R/W
b15	PRMOD	页面读存取模式选择位	0: 正常存取兼容模式 1: 外部数据的读连续有效模式	R/W

WRMOD 位 (写存取模式选择位)

此位选择写存取时的运行模式。

如果将此位置“0”，就为字节选通模式。此时，通过与各字节位置对应的 WRn# 信号 (n=0、1) 控制数据的写操作。

如果将此位置“1”，就为1次写选通模式。此时，通过与各字节位置对应的 BCn# 信号 (n=0、1) 和 WR# 信号控制数据的写操作。但是，在1次写选通模式中，禁止设定8位外部总线宽度。

表 15.8 写存取模式的控制信号

模式	引脚名			
	WR1#	WR0#/WR#	BC1#	BC0#
字节选通模式	○	○	×	×
1次写选通模式	×	○	○	○

○: 有效、×: 无效

EWENB 位 (外部等待允许位)

此位设定允许或者禁止外部等待。

如果将此位置“1”，就允许外部等待，能通过 WAIT# 信号任意控制各周期的等待数。此时，在 WAIT# 信号为 Low 电平期间，插入等待周期。

如果将此位置“0”，WAIT# 信号就无效。

PRENB 位（页面读存取允许位）

此位允许或者禁止页面的读存取。

注. 在通过 CSn 控制寄存器的地址 / 数据多路复用 I/O 接口选择位（CSnCR.MPXEN）选择地址 / 数据多路复用 I/O 接口的情况下，不允许页面的读存取。地址 / 数据多路复用 I/O 接口不支持页面存取。

PWENB 位（页面写存取允许位）

此位设定允许或者禁止页面的写存取。

注. 在通过 CSn 控制寄存器的地址 / 数据多路复用 I/O 接口选择位（CSnCR.MPXEN）选择地址 / 数据多路复用 I/O 接口的情况下，不允许页面的写存取。地址 / 数据多路复用 I/O 接口不支持页面存取。

PRMOD 位（页面读存取模式选择位）

此位选择页面读存取时的运行模式。

如果将此位置“0”，就为正常存取兼容模式。此时，每读 1 个数据，RD# 信号就变为无效并且插入 RD 有效等待。但是，在 RD 有效等待为“0”并且不进行外部总线存取的最后传送时，RD# 信号不变为无效。

如果将此位置“1”，就为外部数据的读连续有效模式。此时，插入 RD 有效等待，但是在此期间 RD# 信号持续有效。

15.3.5 CSn 等待控制寄存器 1 (CSnWCR1) (n=0 ~ 3)

地址 CS0WCR1 0008 3004h、CS1WCR1 0008 3014h、CS2WCR1 0008 3024h、CS3WCR1 0008 3034h

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16	
—	—	—	CSRWAIT[4:0]				—	—	—	CSWWAIT[4:0]				—	—	—
0	0	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1
复位后的值																
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0	
—	—	—	—	—	CSPRWAIT[2:0]			—	—	—	—	—	CSPWWAIT[2:0]			
0	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1	
复位后的值																

位	符号	位名	功能	R/W
b2-b0	CSPWWAIT[2:0]	页面写周期的等待选择位 (注1)	b2 b0 000: 不插入等待 001: 插入1个等待周期 010: 插入2个等待周期 011: 插入3个等待周期 100: 插入4个等待周期 101: 插入5个等待周期 110: 插入6个等待周期 111: 插入7个等待周期	R/W
b7-b3	—	保留位	读写值都为“0”。	R/W
b10-b8	CSPRWAIT[2:0]	页面读周期的等待选择位 (注2)	b10 b8 000: 不插入等待 001: 插入1个等待周期 010: 插入2个等待周期 011: 插入3个等待周期 100: 插入4个等待周期 101: 插入5个等待周期 110: 插入6个等待周期 111: 插入7个等待周期	R/W
b15-b11	—	保留位	读写值都为“0”。	R/W

位	符号	位名	功能	R/W
b20-b16	CSWWAIT[4:0]	正常写周期的等待选择位	b20 b16 00000: 不插入等待 00001: 插入 1 个等待周期 00010: 插入 2 个等待周期 00011: 插入 3 个等待周期 00100: 插入 4 个等待周期 00101: 插入 5 个等待周期 00110: 插入 6 个等待周期 00111: 插入 7 个等待周期 01000: 插入 8 个等待周期 01001: 插入 9 个等待周期 01010: 插入 10 个等待周期 01011: 插入 11 个等待周期 01100: 插入 12 个等待周期 01101: 插入 13 个等待周期 01110: 插入 14 个等待周期 01111: 插入 15 个等待周期 10000: 插入 16 个等待周期 10001: 插入 17 个等待周期 10010: 插入 18 个等待周期 10011: 插入 19 个等待周期 10100: 插入 20 个等待周期 10101: 插入 21 个等待周期 10110: 插入 22 个等待周期 10111: 插入 23 个等待周期 11000: 插入 24 个等待周期 11001: 插入 25 个等待周期 11010: 插入 26 个等待周期 11011: 插入 27 个等待周期 11100: 插入 28 个等待周期 11101: 插入 29 个等待周期 11110: 插入 30 个等待周期 11111: 插入 31 个等待周期	R/W
b23-b21	—	保留位	读写值都为“0”。	R/W

位	符号	位名	功能	R/W
b28-b24	CSRWAIT[4:0]	正常读周期的等待选择位	b28 b24 00000: 不插入等待 00001: 插入 1 个等待周期 00010: 插入 2 个等待周期 00011: 插入 3 个等待周期 00100: 插入 4 个等待周期 00101: 插入 5 个等待周期 00110: 插入 6 个等待周期 00111: 插入 7 个等待周期 01000: 插入 8 个等待周期 01001: 插入 9 个等待周期 01010: 插入 10 个等待周期 01011: 插入 11 个等待周期 01100: 插入 12 个等待周期 01101: 插入 13 个等待周期 01110: 插入 14 个等待周期 01111: 插入 15 个等待周期 10000: 插入 16 个等待周期 10001: 插入 17 个等待周期 10010: 插入 18 个等待周期 10011: 插入 19 个等待周期 10100: 插入 20 个等待周期 10101: 插入 21 个等待周期 10110: 插入 22 个等待周期 10111: 插入 23 个等待周期 11000: 插入 24 个等待周期 11001: 插入 25 个等待周期 11010: 插入 26 个等待周期 11011: 插入 27 个等待周期 11100: 插入 28 个等待周期 11101: 插入 29 个等待周期 11110: 插入 30 个等待周期 11111: 插入 31 个等待周期	R/W
b31-b29	—	保留位	读写值都为“0”。	R/W

注 1. CSPWWAIT[2:0] 位的设定值只在 CSnMOD.PWENB 位为“1”时有效。

注 2. CSPRWAIT[2:0] 位的设定值只在 CSnMOD.PRENB 位为“1”时有效。

CSnWCR1 寄存器是选择外部地址空间各区域的等待周期数的寄存器。复位后，只能写 1 次此寄存器。如果写 2 次或者 2 次以上，就不保证运行。

根据总线接口，必须在“15.6 限制事项 15.6.1 使用分离总线接口时的限制事项”或者“15.6 限制事项 15.6.2 使用地址 / 数据多路复用总线时的限制事项”所示的限制范围内设定各位。

CSPWWAIT[2:0] 位（页面写周期的等待选择位）

这些位设定在第 2 个页面写周期以后的存取中要插入的周期数。CSPWWAIT[2:0] 位的设定在 CSnMOD.PWENB 位为“1”时有效。

注. 必须设定为 $1 \leq \text{CSnWCR2.WDON}[2:0] \text{ 位} \leq \text{CSnWCR2.WRON}[2:0] \text{ 位} \leq \text{CSnWCR1.CSPWWAIT}[2:0] \text{ 位}$ 并且 $\text{CSnWCR2.CSON}[2:0] \text{ 位} \leq \text{CSnWCR2.WRON}[2:0] \text{ 位} \leq \text{CSnWCR1.CSPWWAIT}[2:0] \text{ 位}$ 。

CSPRWAIT[2:0] 位（页面读周期的等待选择位）

这些位设定在第 2 个页面读周期以后的存取中要插入的周期数。CSPRWAIT[2:0] 位的设定在 CSnMOD.PRENB 位为“1”时有效。

注. 必须设定为 CSnWCR2.CSON[2:0] 位 \leq CSnWCR2.RDON[2:0] 位 \leq CSnWCR1.CSPRWAIT[2:0] 位。

CSWWAIT[4:0] 位（正常写周期的等待选择位）

这些位设定在正常写周期和页面写周期的第 1 次存取中要插入的周期数。

注. 必须设定为 $1 \leq$ CSnWCR2.WDON[2:0] 位 \leq CSnWCR2.WRON[2:0] 位 \leq CSnWCR1.CSWWAIT[2:0] 位并且 CSnWCR2.CSON[2:0] 位 \leq CSnWCR2.WRON[2:0] 位 \leq CSnWCR1.CSWWAIT[2:0] 位。

CSRWAIT[4:0] 位（正常读周期的等待选择位）

这些位设定在正常读周期和页面读周期的第 1 次存取中要插入的周期数。

注. 必须设定为 CSnWCR2.CSON[2:0] 位 \leq CSnWCR2.RDON[2:0] 位 \leq CSnWCR1.CSRWAIT[4:0] 位。

15.3.6 CSn 等待控制寄存器 2 (CSnWCR2) (n=0 ~ 3)

地址 CS0WCR2 0008 3008h、CS1WCR2 0008 3018h、CS2WCR2 0008 3028h、CS2WCR2 0008 3038h

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
—	CSON[2:0]			—	WDON[2:0]			—	WRON[2:0]			—	RDON[2:0]		
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
复位后的值															
b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	AWAIT[1:0]		—	WDOFF[2:0]			—	CSWOFF[2:0]			—	CSROFF[2:0]		
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
复位后的值															

位	符号	位名	功能	R/W
b2-b0	CSROFF[2:0]	读取时的 CS 延长周期选择位	b2 b0 000: 不插入等待 001: 插入 1 个等待周期 010: 插入 2 个等待周期 011: 插入 3 个等待周期 100: 插入 4 个等待周期 101: 插入 5 个等待周期 110: 插入 6 个等待周期 111: 插入 7 个等待周期	R/W
b3	—	保留位	读写值都为“0”。	R/W
b6-b4	CSWOFF[2:0]	写入时的 CS 延长周期选择位	b6 b4 000: 不插入等待 001: 插入 1 个等待周期 010: 插入 2 个等待周期 011: 插入 3 个等待周期 100: 插入 4 个等待周期 101: 插入 5 个等待周期 110: 插入 6 个等待周期 111: 插入 7 个等待周期	R/W
b7	—	保留位	读写值都为“0”。	R/W
b10-b8	WDOFF[2:0]	写数据的输出延长周期选择位	b10 b8 000: 不插入等待 001: 插入 1 个等待周期 010: 插入 2 个等待周期 011: 插入 3 个等待周期 100: 插入 4 个等待周期 101: 插入 5 个等待周期 110: 插入 6 个等待周期 111: 插入 7 个等待周期	R/W
b11	—	保留位	读写值都为“0”。	R/W
b13-b12	AWAIT[1:0]	地址周期等待选择位	b13 b12 00: 不插入等待 01: 插入 1 个等待周期 10: 插入 2 个等待周期 11: 插入 3 个等待周期	R/W
b15-b14	—	保留位	读写值都为“0”。	R/W

位	符号	位名	功能	R/W
b18-b16	RDON[2:0]	RD 有效等待选择位	b18 b16 000: 不插入等待 001: 插入 1 个等待周期 010: 插入 2 个等待周期 011: 插入 3 个等待周期 100: 插入 4 个等待周期 101: 插入 5 个等待周期 110: 插入 6 个等待周期 111: 插入 7 个等待周期	R/W
b19	—	保留位	读写值都为“0”。	R/W
b22-b20	WRON[2:0]	WR 有效等待选择位	b22 b20 000: 不插入等待 001: 插入 1 个等待周期 010: 插入 2 个等待周期 011: 插入 3 个等待周期 100: 插入 4 个等待周期 101: 插入 5 个等待周期 110: 插入 6 个等待周期 111: 插入 7 个等待周期	R/W
b23	—	保留位	读写值都为“0”。	R/W
b26-b24	WDON[2:0]	写数据的输出等待选择位	b26 b24 000: 不插入等待 001: 插入 1 个等待周期 010: 插入 2 个等待周期 011: 插入 3 个等待周期 100: 插入 4 个等待周期 101: 插入 5 个等待周期 110: 插入 6 个等待周期 111: 插入 7 个等待周期	R/W
b27	—	保留位	读写值都为“0”。	R/W
b30-b28	CSON[2:0]	CS 有效等待选择位	b30 b28 000: 不插入等待 001: 插入 1 个等待周期 010: 插入 2 个等待周期 011: 插入 3 个等待周期 100: 插入 4 个等待周期 101: 插入 5 个等待周期 110: 插入 6 个等待周期 111: 插入 7 个等待周期	R/W
b31	—	保留位	读写值都为“0”。	R/W

CSnWCR2 寄存器是选择外部地址空间各区域的等待周期数的寄存器。复位后，只能写 1 次此寄存器。如果写 2 次或者 2 次以上，就不保证运行。

根据总线接口，必须在“15.6 限制事项 15.6.1 使用分离总线接口时的限制事项”或者“15.6 限制事项 15.6.2 使用地址 / 数据多路复用总线时的限制事项”所示的限制范围内设定各位。

CSROFF[2:0] 位（读取时的 CS 延长周期选择位）

这些位设定从读存取时的等待周期结束（RD# 信号无效时）到 CSn# 信号（n=0 ~ 3）无效为止的周期数。

CSWOFF[2:0] 位（写入时的 CS 延长周期选择位）

这些位设定从写存取时的等待周期结束（WRn# 信号（n=0～3）无效时）到 CSn# 信号（n=0～7）无效为止的周期数。

注. 必须设定为 WDOFF[2:0] 位 \leq CSWOFF[2:0] 位。

WDOFF[2:0] 位（写数据的输出延长周期选择位）

这些位设定从写存取时的等待周期结束（WRn# 信号（n=0～3）无效时）到写数据输出结束为止的周期数。

注. 必须设定为 WDOFF[2:0] 位 \leq CSWOFF[2:0] 位。

AWAIT[1:0] 位（地址周期等待选择位）

将等待状态插入到地址 / 数据多路复用 I/O 接口的地址输出周期。

注. 必须如下进行设定：

CSnWCR2.CSON[2:0] 位 \leq CSnWCR2.AWAIT[1:0] 位

在进行读存取时，CSnWCR2.AWAIT[1:0] 位 +2 \leq CSnWCR2.RDON[2:0] 位 \leq CSnWCR1.CSRWAIT[4:0] 位

在进行写存取时，CSnWCR2.AWAIT[1:0] 位 +2 \leq CSnWCR2.WRON[2:0] 位 \leq CSnWCR1.CSWWAIT[4:0] 位

并且 CSnWCR2.AWAIT[1:0] 位 +2 \leq CSnWCR2.WDON[2:0] 位 \leq CSnWCR1.CSWWAIT[4:0] 位

RDON[2:0] 位（RD 有效等待选择位）

这些位设定在 RD# 信号有效前要插入的等待周期数。

注. 在进行正常读存取时，必须设定为 CSnWCR2.CSON[2:0] 位 \leq CSnWCR2.RDON[2:0] 位 \leq CSnWCR1.CSRWAIT[4:0] 位。

在进行页面读存取时，必须设定为 CSnWCR2.CSON[2:0] 位 \leq CSnWCR2.RDON[2:0] 位 \leq CSnWCR1.CSPRWAIT[4:0] 位。

在地址 / 数据多路复用 I/O 接口的情况下，必须设定为 CSnWCR2.AWAIT[1:0] 位 +2 \leq CSnWCR2.RDON[2:0] 位 \leq CSnWCR1.CSRWAIT[4:0] 位。

WRON[2:0] 位（WR 有效等待选择位）

这些位设定在 WRn# 信号（n=0～3）有效前要插入的等待周期数。

注. 在进行正常写存取时，必须设定为 1 \leq CSnWCR2.WDON[2:0] 位 \leq CSnWCR2.WRON[2:0] 位 \leq CSnWCR1.CSWWAIT[4:0] 位并且 CSnWCR2.CSON[2:0] 位 \leq CSnWCR2.WRON[2:0] 位 \leq CSnWCR1.CSWWAIT[4:0] 位。

在进行页面写存取时，必须设定为 1 \leq CSnWCR2.WDON[2:0] 位 \leq CSnWCR2.WRON[2:0] 位 \leq CSnWCR1.CSPWAIT[4:0] 位并且 CSnWCR2.CSON[2:0] 位 \leq CSnWCR2.WRON[2:0] 位 \leq CSnWCR1.CSPWAIT[4:0] 位。

在地址 / 数据多路复用 I/O 接口的情况下，必须设定为 CSnWCR2.AWAIT[1:0] 位 +2 \leq CSnWCR2.WRON[2:0] 位 \leq CSnWCR1.CSWWAIT[4:0] 位。

WDON[2:0] 位（写数据的输出等待选择位）

这些位设定在输出写数据前要插入的等待周期数。

- 注 . 在进行正常写存取时，必须设定为 $1 \leq \text{CSnWCR2.WDON}[2:0] \text{ 位} \leq \text{CSnWCR2.WRON}[2:0] \text{ 位} \leq \text{CSnWCR1.CSWWAIT}[4:0] \text{ 位}$ 。
在进行页面写存取时，必须设定为 $1 \leq \text{CSnWCR2.WDON}[2:0] \text{ 位} \leq \text{CSnWCR2.WRON}[2:0] \text{ 位} \leq \text{CSnWCR1.CSPWAIT}[4:0] \text{ 位}$ 。
在地址/数据多路复用 I/O 接口的情况下，必须设定为 $\text{CSnWCR2.AWAIT}[1:0] \text{ 位} + 2 \leq \text{CSnWCR2.WDON}[2:0] \text{ 位} \leq \text{CSnWCR1.CSWWAIT}[4:0] \text{ 位}$ 。

CSON[2:0] 位（CS 有效等待选择位）

这些位设定在 CSn# 信号（n=0 ~ 3）有效前要插入的等待周期数。

- 注 . 在进行正常读存取时，必须设定为 $\text{CSnWCR2.CSON}[2:0] \text{ 位} \leq \text{CSnWCR2.RDON}[2:0] \text{ 位} \leq \text{CSnWCR1.CSRWAIT}[4:0] \text{ 位}$ 。
在进行页面读存取时，必须设定为 $\text{CSnWCR2.CSON}[2:0] \text{ 位} \leq \text{CSnWCR2.RDON}[2:0] \text{ 位} \leq \text{CSnWCR1.CSPRWAIT}[4:0] \text{ 位}$ 。
在进行正常写存取时，必须设定为 $\text{CSnWCR2.CSON}[2:0] \text{ 位} \leq \text{CSnWCR2.WRON}[2:0] \text{ 位} \leq \text{CSnWCR1.CSWWAIT}[4:0] \text{ 位}$ 。
在进行页面写存取时，必须设定为 $\text{CSnWCR2.CSON}[2:0] \text{ 位} \leq \text{CSnWCR2.WRON}[2:0] \text{ 位} \leq \text{CSnWCR1.CSPWAIT}[4:0] \text{ 位}$ 。
在地址/数据多路复用 I/O 接口的情况下，必须设定为 $\text{CSnWCR2.CSON}[2:0] \text{ 位} \leq \text{CSnWCR2.AWAIT}[1:0] \text{ 位}$ 。

15.3.7 总线错误状态清除寄存器 (BERCLR)

地址 0008 1300h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	STSCLR
0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	STSCLR	状态清除位	0: 无效 1: 清除总线错误状态寄存器	(W) (注1)
b7-b1	—	保留位	读写值都为“0”。	R/W

注 1. 只能写“1”，写“0”无效。

STSCLR 位 (状态清除位)

如果给此位写“1”，就清除总线错误状态寄存器 1 (BERSR1) 和总线错误状态寄存器 2 (BERSR2)。
写“0”无效，读取值为“0”。

15.3.8 总线错误监视允许寄存器 (BEREN)

地址 0008 1304h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	TOEN	IGAEN
0	0	0	0	0	0	0	0

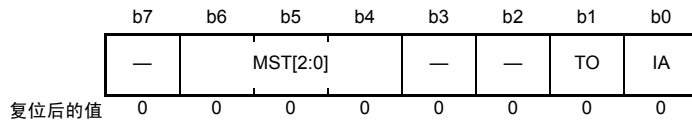
位	符号	位名	功能	R/W
b0	IGAEN	非法地址存取检测允许位	0: 禁止检测非法地址的存取 1: 允许检测非法地址的存取	R/W
b1	TOEN	超时检测允许位 (注 1、注 2)	0: 禁止检测总线超时 1: 允许检测总线超时	R/W
b7-b2	—	保留位	读写值都为“0”。	R/W

注 1. 如果在设定为禁止检测 (TOEN 位 =0) 后存取总线，总线就可能冻结。

注 2. 不能在检测超时错误的过程中将 TOEN 位置“0” (禁止检测)。

15.3.9 总线错误状态寄存器 1 (BERSR1)

地址 0008 1308h



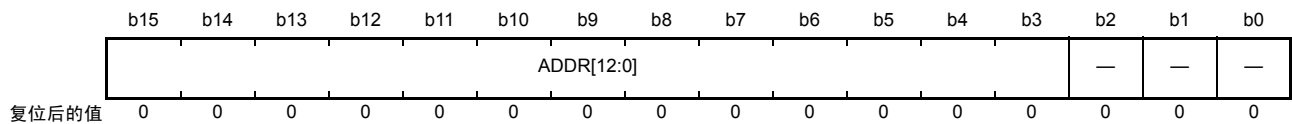
位	符号	位名	功能	R/W
b0	IA	非法地址存取位	0: 未发生非法地址的存取 1: 发生非法地址的存取	R
b1	TO	超时位	0: 未发生超时 1: 发生超时	R
b3-b2	—	保留位	读取值为“0”，写操作无效。	R
b6-b4	MST[2:0]	总线主控代码位	b6 b4 000: CPU 001: 不能设定 010: 不能设定 011: DTC/DMAC 100: 不能设定 101: 不能设定 110: 不能设定 111: 不能设定	R
b7	—	保留位	读取值为“0”，写操作无效。	R

MST[2:0] 位 (总线主控代码位)

这些位表示发生总线错误存取的总线主控。

15.3.10 总线错误状态寄存器 2 (BERSR2)

地址 0008 130Ah



位	符号	位名	功能	R/W
b2-b0	—	保留位	读取值为“0”，写操作无效。	R
b15-b3	ADDR[12:0]	总线错误发生地址位	发生总线错误存取的地址高 13 位 (以 512K 字节为单位)	R

15.3.11 总线优先级控制寄存器 (BUSPRI)

地址 0008 1310h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	BPEB[1:0]	BPFB[1:0]	—	—	BPGB[1:0]	BPIB[1:0]	BPRO[1:0]	BPRA[1:0]						
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b1-b0	BPRA[1:0]	存储器总线 1 (内部 RAM) 的优先级控制位	b1 b0 0 0: 优先级固定 0 1: 优先级交替 1 0: 不能设定 1 1: 不能设定	R(W) (注 1)
b3-b2	BPRO[1:0]	存储器总线 2 (内部 ROM) 的优先级控制位	b3 b2 0 0: 优先级固定 0 1: 优先级交替 1 0: 不能设定 1 1: 不能设定	R(W) (注 1)
b5-b4	BPIB[1:0]	内部外围总线 1 的优先级控制位	b5 b4 0 0: 优先级固定 0 1: 优先级交替 1 0: 不能设定 1 1: 不能设定	R(W) (注 1)
b7-b6	BPGB[1:0]	内部外围总线 2 的优先级控制位	b7 b6 0 0: 优先级固定 0 1: 优先级交替 1 0: 不能设定 1 1: 不能设定	R(W) (注 1)
b9-b8	—	保留位	读写值都为“0”。	R/W
b11-b10	BPFB[1:0]	内部外围总线 6 的优先级控制位	b11 b10 0 0: 优先级固定 0 1: 优先级交替 1 0: 不能设定 1 1: 不能设定	R(W) (注 1)
b13-b12	BPEB[1:0]	外部总线的优先级控制位	b13 b12 0 0: 优先级固定 0 1: 优先级交替 1 0: 不能设定 1 1: 不能设定	R(W) (注 1)
b15-b14	—	保留位	读写值都为“0”。	R/W

注 1. 只能在 DTC 和 DMAC 停止的状态下写 1 次此寄存器。如果写 2 次或者 2 次以上, 就不保证运行。

BPRA[1:0] 位 (存储器总线 1 (内部 RAM) 的优先级控制位)

这些位设定存储器总线 1 (内部 RAM) 的优先级。

在优先级固定的情况下, 优先级为内部主总线 2 > CPU 总线。

在优先级交替的情况下, 内部主总线 2 和 CPU 总线中已接受总线请求的一方的优先级变低。

BPRO[1:0] 位（存储器总线 2（内部 ROM）的优先级控制位）

这些位设定存储器总线 2（内部 ROM）的优先级。

在优先级固定的情况下，优先级为内部主总线 2 > CPU 总线。

在优先级交替的情况下，内部主总线 1 和 CPU 总线中已接受总线请求的一方的优先级变低。

BPIB[1:0] 位（内部外围总线 1 的优先级控制位）

这些位设定内部外围总线 1 的优先级。

在优先级固定的情况下，优先级为内部主总线 2 > 内部主总线 1。

在优先级交替的情况下，内部主总线 1 和内部主总线 2 中已接受总线请求的一方的优先级变低。

BPGB[1:0] 位（内部外围总线 2 的优先级控制位）

这些位设定内部外围总线 2 的优先级。

在优先级固定的情况下，优先级为内部主总线 2 > 内部主总线 1。

在优先级交替的情况下，内部主总线 1 和内部主总线 2 中已接受总线请求的一方的优先级变低。

BPFB[1:0] 位（内部外围总线 6 的优先级控制位）

这些位设定内部外围总线 6 的优先级。

在优先级固定的情况下，优先级为内部主总线 2 > 内部主总线 1。

在优先级交替的情况下，内部主总线 1 和内部主总线 2 中已接受总线请求的一方的优先级变低。

BPEB[1:0] 位（外部总线的优先级控制位）

这些位设定外部总线的优先级。

在优先级固定的情况下，优先级为内部主总线 2 > 内部主总线 1。

在优先级交替的情况下，内部主总线 1 和内部主总线 2 中已接受总线请求的一方的优先级变低。

15.4 字节序和数据调整

外部总线具有数据调整功能，根据存取区域的总线规格（8 位总线空间、16 位总线空间）、数据长度和字节序模式，控制在存取外部地址空间（CS 区域）时使用数据总线 D15 ~ D8 和 D7 ~ D0 中的哪条数据总线。

15.4.1 CS 区域的数据调整控制

(1) 16 位总线空间

如果通过 CSnCR.BSIZE[1:0] 位选择 16 位总线空间，地址总线（A23 ~ A1）就作为以 16 位为单位的地址输出信号并且有效，地址总线（A0）无效（固定为 Low 电平）。

当选择字节选通模式（CSnMOD.WRMOD 位 =0）时，WR0# 引脚和 WR1# 引脚有效，并且不使用 BC0# 引脚和 BC1# 引脚。

当选择 1 次写选通模式（CSnMOD.WRMOD 位 =1）时，只有 WR0# 引脚有效，与数据长度无关，在进行写存取时，WR0# 引脚输出 Low 电平。此时，WR1# 引脚无效（固定为 High 电平）。BC0# 引脚和 BC1# 引脚表示有效的字节位置。

对于 16 位总线空间，在存取 32 位数据时有可能发生页面存取。页面存取对象是存取操作不超过 32 位边界并且 BC0# 信号和 BC1# 信号不变的传送。发生页面存取时的情况如图 15.6 和图 15.7 中的 (p) 所示。

对于 16 位总线空间，芯片的外部数据和控制信号的有效位置因大端法和小端法而不同。

数据长度	存取地址	存取次数	总线周期	数据量	地址	数据总线			
						D15	D8	D7	D0
8bit	4n	1次	第1次	8bit	4n	[7 0]			
	4n+1	1次	第1次	8bit	4n	[7 0]			
	4n+2	1次	第1次	8bit	4n+2	[7 0]			
	4n+3	1次	第1次	8bit	4n+2	[7 0]			
16bit	4n	1次	第1次	16bit	4n	[15 8 7 0]			
	4n+1	2次	第1次	8bit	4n	[7 0]			
			第2次	8bit	4n+2	[15 8]			
	4n+2	1次	第1次	16bit	4n+2	[15 8 7 0]			
	4n+3	2次	第1次	8bit	4n+2	[7 0]			
			第2次	8bit	4n+4	[15 8]			
32bit	4n	2次	第1次	16bit	4n	[15 8 7 0]			
			第2次	16bit	4n+2 (p)	[31 24 23 16]			
	4n+1	3次	第1次	8bit	4n	[7 0]			
			第2次	16bit	4n+2	[23 16 15 8]			
			第3次	8bit	4n+4	[31 24]			
	4n+2	2次	第1次	16bit	4n+2	[15 8 7 0]			
			第2次	16bit	4n+4	[31 24 23 16]			
	4n+3	3次	第1次	8bit	4n+2	[7 0]			
			第2次	16bit	4n+4	[23 16 15 8]			
第3次			8bit	4n+6	[31 24]				

当通过CSnMOD.PREN位和CSnMOD.PWEN位允许页面存取时，(p)表示的存取为页面存取的对象。

图 15.6 16 位总线空间的数据调整（小端法）

数据长度	存取地址	存取次数	总线周期	数据量	地址	WR1#/BC1#	WR0#/BC0#	
						RD#		
						数据总线		
						D15	D8 D7	D0
8bit	4n	1次	第1次	8bit	4n	7	0	
	4n+1	1次	第1次	8bit	4n	7	0	
	4n+2	1次	第1次	8bit	4n+2	7	0	
	4n+3	1次	第1次	8bit	4n+2	7	0	
16bit	4n	1次	第1次	16bit	4n	15	8 7	0
	4n+1	2次	第1次	8bit	4n	15	8	
			第2次	8bit	4n+2	7	0	
	4n+2	1次	第1次	16bit	4n+2	15	8 7	0
	4n+3	2次	第1次	8bit	4n+2	15	8	
			第2次	8bit	4n+4	7	0	
32bit	4n	2次	第1次	16bit	4n	31	24 23	16
			第2次	16bit	4n+2 (p)	15	8 7	0
	4n+1	3次	第1次	8bit	4n	31	24	
			第2次	16bit	4n+2	23	16 15	8
			第3次	8bit	4n+4	7	0	
	4n+2	2次	第1次	16bit	4n+2	31	24 23	16
			第2次	16bit	4n+4	15	8 7	0
	4n+3	3次	第1次	8bit	4n+2	31	24	
			第2次	16bit	4n+4	23	16 15	8
		第3次	8bit	4n+6	7	0		

当通过CSnMOD.PRENB位和CSnMOD.PWENB位允许页面存取时，(p)表示的存取为页面存取的对象。

图 15.7 16 位总线空间的数据调整（大端法）

(2) 8 位总线空间

如果通过 CSnCR.BSIZE[1:0] 位选择 8 位总线空间，地址总线（A23 ~ A0）就作为以字节为单位的地址信号并且有效。

对于 8 位总线空间，与写存取模式的设定无关，只有 WR0# 引脚有效。在进行写存取时，将 Low 电平输出到 WR0# 引脚。不使用 WR1# 引脚、BC0# 引脚和 BC1# 引脚。

对于 8 位总线空间，在存取 16 位或者 32 位数据时有可能发生页面存取。页面存取对象是存取操作不超过 32 位边界的传送。发生页面存取时的情况如图 15.8 和图 15.9 中的 (p) 所示。

对于 8 位总线空间，与字节序无关，芯片的外部数据为 D7 ~ D0，控制信号使用 WR0# 信号。

数据长度	存取地址	存取次数	总线周期	数据量	地址	数据总线			
						D15	D8	D7	D0
8bit	4n	1次	第1次	8bit	4n	7		0	
	4n+1	1次	第1次	8bit	4n+1	7		0	
	4n+2	1次	第1次	8bit	4n+2	7		0	
	4n+3	1次	第1次	8bit	4n+3	7		0	
16bit	4n	2次	第1次	8bit	4n	7		0	
			第2次	8bit	4n+1 (p)	15		8	
	4n+1	2次	第1次	8bit	4n+1	7		0	
			第2次	8bit	4n+2 (p)	15		8	
	4n+2	2次	第1次	8bit	4n+2	7		0	
			第2次	8bit	4n+3 (p)	15		8	
	4n+3	2次	第1次	8bit	4n+3	7		0	
			第2次	8bit	4n+4	15		8	
32bit	4n	4次	第1次	8bit	4n	7		0	
			第2次	8bit	4n+1 (p)	15		8	
			第3次	8bit	4n+2 (p)	23		16	
			第4次	8bit	4n+3 (p)	31		24	
	4n+1	4次	第1次	8bit	4n+1	7		0	
			第2次	8bit	4n+2 (p)	15		8	
			第3次	8bit	4n+3 (p)	23		16	
			第4次	8bit	4n+4	31		24	
	4n+2	4次	第1次	8bit	4n+2	7		0	
			第2次	8bit	4n+3 (p)	15		8	
			第3次	8bit	4n+4	23		16	
			第4次	8bit	4n+5 (p)	31		24	
	4n+3	4次	第1次	8bit	4n+3	7		0	
			第2次	8bit	4n+4	15		8	
			第3次	8bit	4n+5 (p)	23		16	
			第4次	8bit	4n+6 (p)	31		24	

当通过CSnMOD.PRENB位和CSnMOD.PWENB位允许页面存取时，(p)表示的存取为页面存取的对象。

图 15.8 8 位总线空间的数据调整（小端法）

数据长度	存取地址	存取次数	总线周期	数据量	地址	数据总线			
						D15	D8	D7	D0
8bit	4n	1次	第1次	8bit	4n	[7] [0]			
	4n+1	1次	第1次	8bit	4n+1	[7] [0]			
	4n+2	1次	第1次	8bit	4n+2	[7] [0]			
	4n+3	1次	第1次	8bit	4n+3	[7] [0]			
16bit	4n	2次	第1次	8bit	4n	[15] [8]			
			第2次	8bit	4n+1 (p)	[7] [0]			
	4n+1	2次	第1次	8bit	4n+1	[15] [8]			
			第2次	8bit	4n+2 (p)	[7] [0]			
	4n+2	2次	第1次	8bit	4n+2	[15] [8]			
			第2次	8bit	4n+3 (p)	[7] [0]			
	4n+3	2次	第1次	8bit	4n+3	[15] [8]			
			第2次	8bit	4n+4	[7] [0]			
32bit	4n	4次	第1次	8bit	4n	[31] [24]			
			第2次	8bit	4n+1 (p)	[23] [16]			
			第3次	8bit	4n+2 (p)	[15] [8]			
			第4次	8bit	4n+3 (p)	[7] [0]			
	4n+1	4次	第1次	8bit	4n+1	[31] [24]			
			第2次	8bit	4n+2 (p)	[23] [16]			
			第3次	8bit	4n+3 (p)	[15] [8]			
			第4次	8bit	4n+4	[7] [0]			
	4n+2	4次	第1次	8bit	4n+2	[31] [24]			
			第2次	8bit	4n+3 (p)	[23] [16]			
			第3次	8bit	4n+4	[15] [8]			
			第4次	8bit	4n+5 (p)	[7] [0]			
	4n+3	4次	第1次	8bit	4n+3	[31] [24]			
			第2次	8bit	4n+4	[23] [16]			
			第3次	8bit	4n+5 (p)	[15] [8]			
			第4次	8bit	4n+6 (p)	[7] [0]			

当通过CSnMOD.PRENb位和CSnMOD.PWENb位允许页面存取时，(p)表示的存取为页面存取的对象。

图 15.9 8 位总线空间的数据调整（大端法）

15.5 CS 区域控制器的运行说明

15.5.1 分离总线

时序图中记载的各周期的说明如下所示。

CSC (CS 区域控制器) 与外部总线时钟 (BCLK) 同步运行。因此, 通过 BCLK 对 CSC 的寄存器所设等待数等进行计数。在以下内容中, 如果没有特别说明, 就将外部总线时钟 (BCLK) 的频率和 BCLK 引脚的输出频率视为相同。

在 BCLK 引脚输出的上升沿开始存取外部总线。当外部总线时钟 (BCLK) 的频率和 BCLK 引脚的输出频率不同以及总线主控的 1 次传送请求至少发生 2 次外部总线存取时, 根据等待数的设定, 有可能在 BCLK 引脚输出的下降沿开始第 2 次以后的外部总线存取 (参照图 15.15 ~ 图 15.19)。

① Tw1 ~ Twn (正常读周期的等待和正常写周期的等待)

这是从外部总线存取开始到选通信号有效的前 1 个周期为止的周期期间, 能选择 0 ~ 31 个周期。在此期间内, 根据等待数的设定, 将 CSn# 信号、RD# 信号和 WRn# 信号设定为 “Low” 电平有效。能通过 CSn 等待控制寄存器 2 (CSnWCR2) 的 CS 有效等待选择位 (CSOEN)、RD 有效等待选择位 (RDON)、WR 有效等待选择位 (WRON) 和写数据的输出等待选择位 (WDOEN) 控制有效时序。各等待的周期数从外部总线存取的开始周期算起, 可选择 0 ~ 7 个周期, 但是可选择的周期数不超过读 / 写周期等待的周期数。

② Tend (选通信号有效周期)

Tend 表示正常读 / 写周期等待期间或者页面读 / 写周期等待期间结束后的下一个周期。当正常读 / 写周期等待或者页面读 / 写周期等待的各选择位为 “0” 时, 总线存取的开始周期为选通信号有效周期。在选通信号有效周期的下一个周期, RD# 信号和 WRn# 信号无效。在进行读存取时, Tend 为读数据的采样周期。

在允许外部等待的情况下, 在选通信号有效周期对等待信号进行采样。当等待信号为 Low 电平时, 延长总线周期, 一旦等待信号变为 High 电平, 就在下一个周期结束总线周期。(Tend) 表示开始等待信号的采样周期。

除了在页面存取中的第 1 次选通信号有效周期以及设定了写存取时的写数据输出延长周期 (非 “0” 值) 的情况 (④) 以外, 开始下一个周期的第 2 次以后的页面存取 (⑤)。如果 RD 有效等待或者 WR 有效等待的设定值不是 “0”, RD# 信号和 WRn# 信号就在下一个周期无效。如果设定值是 “0”, 各信号就继续有效, 而且 CSn# 信号也继续有效。

③ Tn1 ~ Tnm (CS 延长周期)

在正常存取的情况下, Tn1 ~ Tnm 表示从选通信号有效周期 (Tend) 的下一个周期到 CSn# 信号无效为止的周期期间。在进行读存取时, 能通过读 CSn 等待控制寄存器 2 (CSnWCR2) 时的 CS 延长周期选择位 (CSROFF) 控制无效时序; 在进行写存取时, 能通过写入时的 CS 延长周期选择位 (CSWOFF) 控制无效时序。

周期数从选通信号有效周期的下一个周期开始算起。

在页面存取的情况下, Tn1 ~ Tnm 表示从最后的选通信号有效周期到 CSn# 信号无效为止的周期期间。

在进行写存取时, 通过写数据的输出延长周期选择位 (WDOFF) 延长地址和输出数据。

④ Tdw1 ~ Tdwn (写数据的输出延长周期)

在进行写存取时, 如果写数据的输出延长等待的设定值不是 “0”, 就从选通信号有效周期 (Tend) 的下一个周期插入写数据的输出延长周期。

在正常存取的情况下, 在 CS 延长周期 (③) 期间内插入周期。

在页面存取的情况下, 在选通信号有效周期 (Tend) 和后续的页面存取之间以及 CS 延长周期 (③) 期间内插入周期。在此期间, 地址和输出数据被延迟并且 WRn# 信号无效。

⑤ Tpw1 ~ Tpw_n (页面读周期的等待和页面写周期的等待)

对于页面存取第 2 次以后的总线周期，使用页面读周期等待和页面写周期等待的值而不使用正常读周期等待和正常写周期等待的值。和第 1 次存取一样，WR 有效等待的设定值有效。RD 有效的设定操作因页面读存取模式 (CSnMOD.PRMOD 位) 的设定而不同。

CDnMOD.PRMOD 位 =0: 和第 1 次一样，插入 RD 有效等待并且将 RD# 信号置为无效。

CDnMOD.PRMOD 位 =1: 和正常存取兼容模式一样，插入 RD 有效等待，但是 RD# 信号在该期间继续有效。

⑥ Tr1 ~ Trn (恢复周期)

能从总线周期结束时 (CSn# 信号无效) 开始插入恢复周期。能通过 CSn 恢复周期设定寄存器 (CSnREC) 的读恢复周期设定位 (RRCV) 和写恢复周期设定位 (WRCV) 控制恢复周期数。各恢复周期数从总线周期结束时 (CSn# 信号无效) 开始算起，可选择 0~15 个周期。恢复周期的详细内容请参照“15.5.4 恢复周期的插入”。

(1) 正常存取

如果将 CSnMOD.PRENB 位置“0” (禁止页面读存取) 并且将 CSnMOD.PWENB 位置“0” (禁止页面写存取)，全部总线存取就进行正常存取。

即使将 CSnMOD.PRENB 位置“1” (允许页面读存取) 并且将 CSnMOD.PWENB 位置“1” (允许页面写存取)，与页面存取不对应的总线存取也为正常存取。

正常存取的运行例子如图 15.10 ~ 图 15.12 所示。

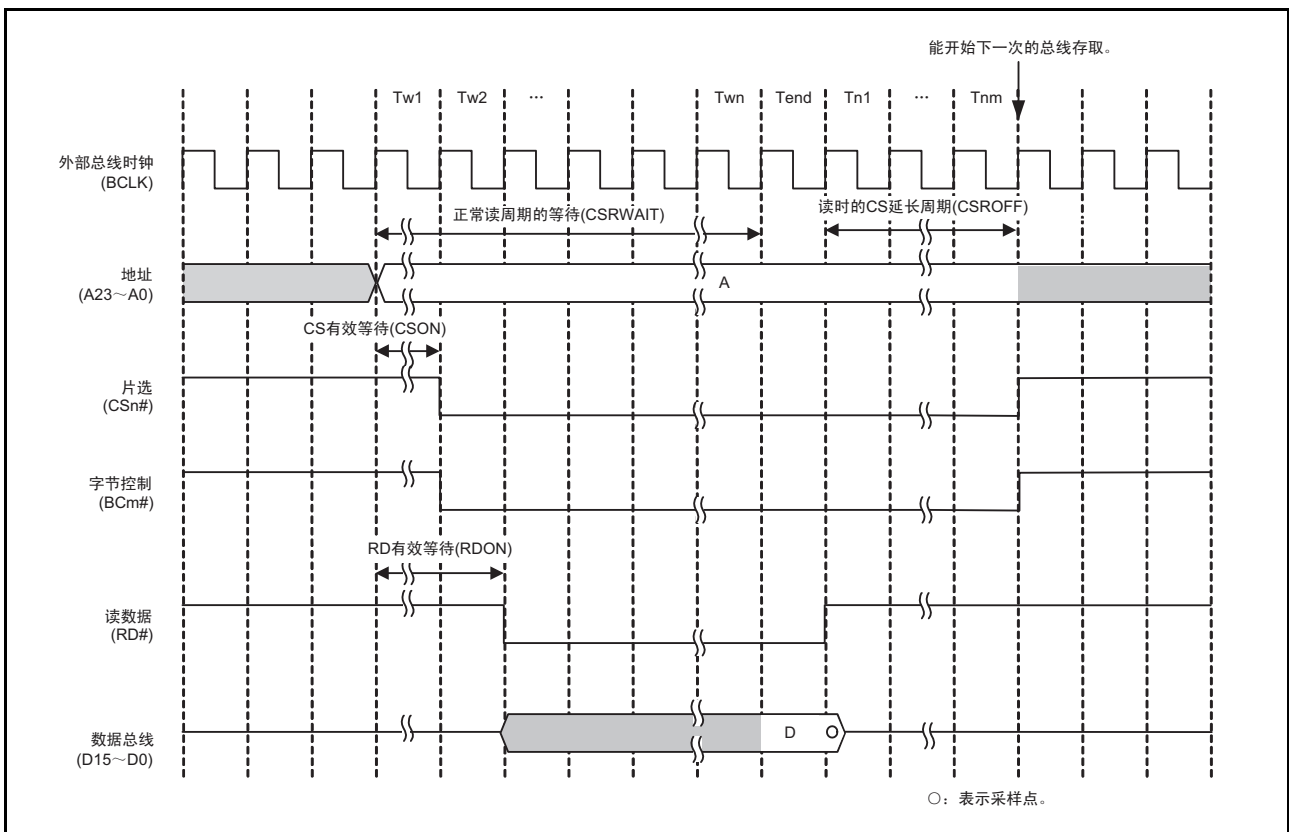


图 15.10 总线时序 (正常读) (n=0 ~ 3、m=0、1)

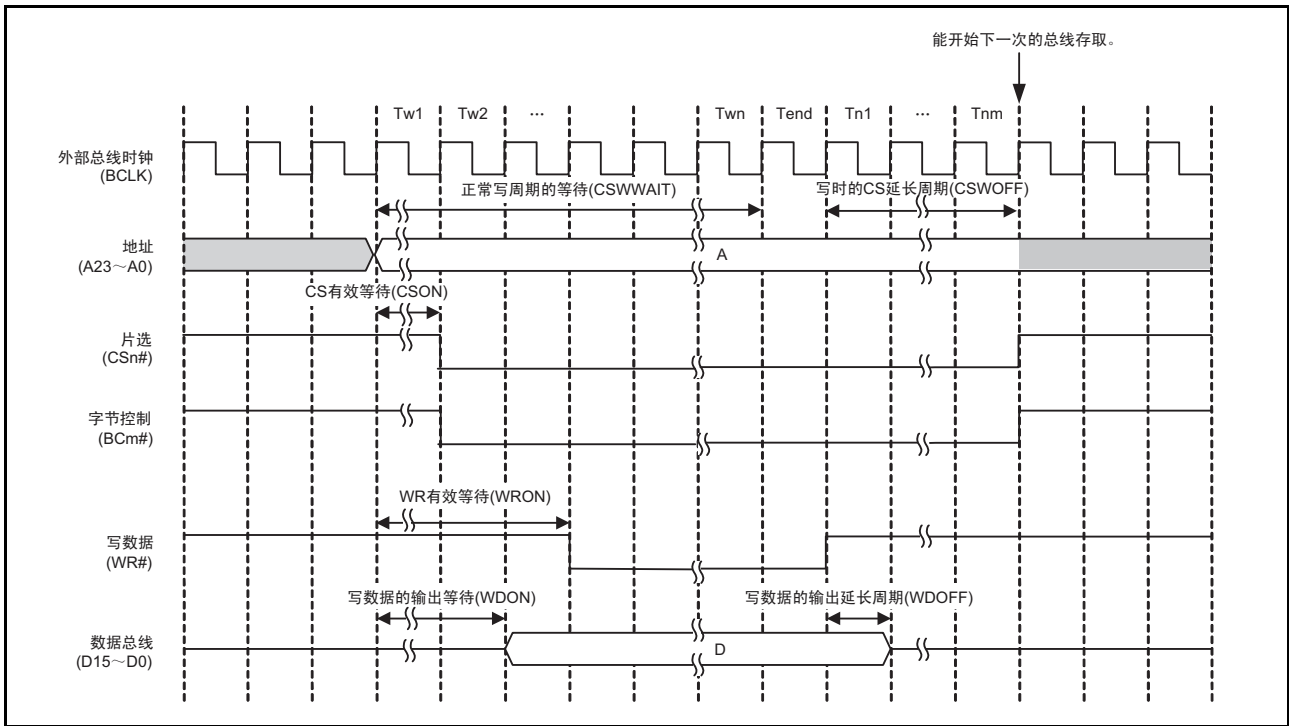


图 15.11 总线时序（正常写，1次写选通模式）（n=0 ~ 3、m=0、1）

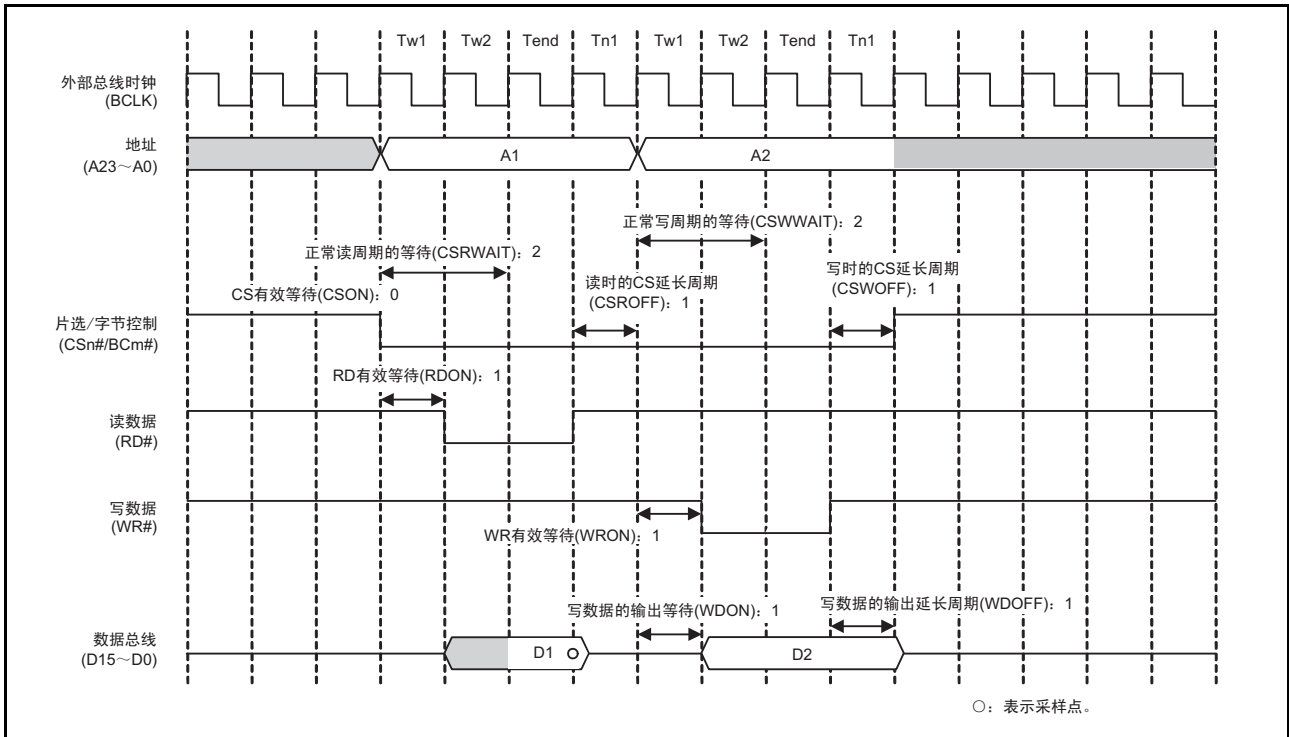


图 15.12 正常存取的运行例子（读写）（n=0 ~ 3、m=0、1）

在总线主控的1次传送请求至少需要2次外部总线存取时，重复正常存取的操作（①~④）。1次传送请求发生2次总线存取时的运行例子如图 15.13 和图 15.14 所示。但是，当恢复周期插入条件成立时，在第2次以后的外部总线存取中也插入恢复周期（⑥）（参照图 15.33）。

各等待控制寄存器的值是设定例子，必须根据要连接的设备规格设定寄存器的值。

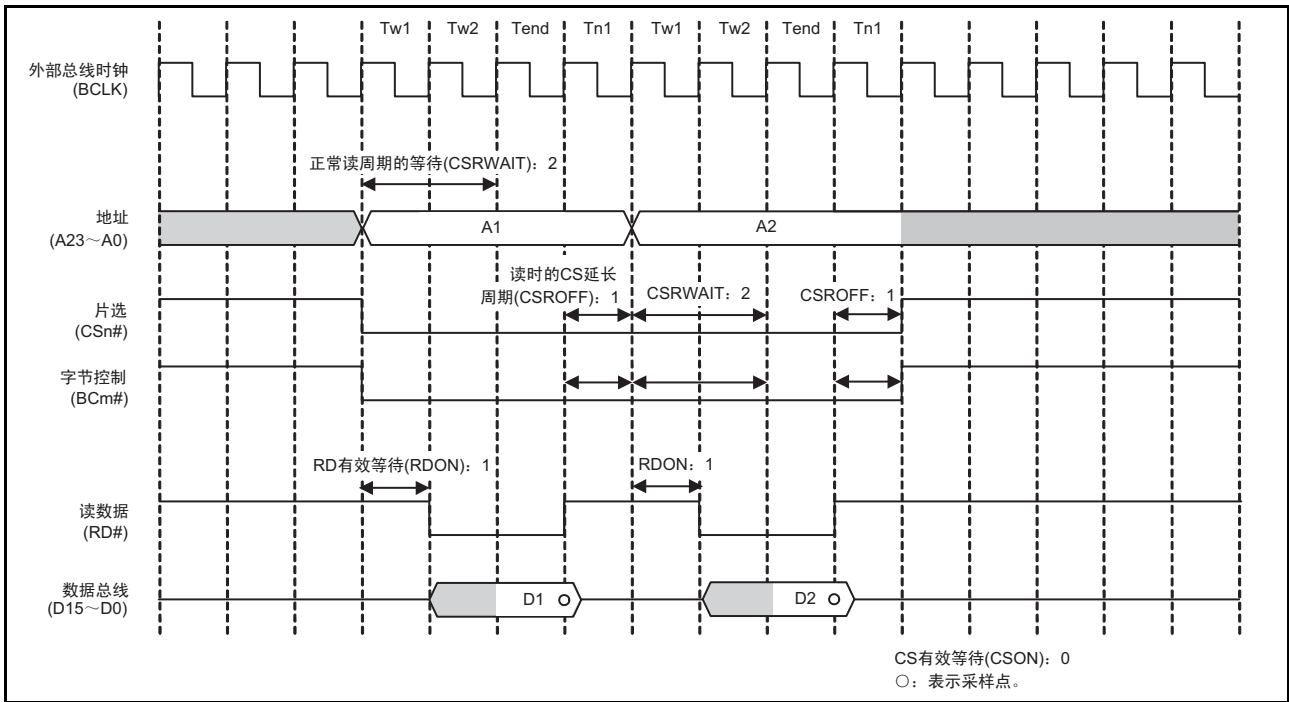


图 15.13 正常读存取的运行例子 (1次传送请求发生2次总线存取的情况) (n=0 ~ 3、m=0、1)

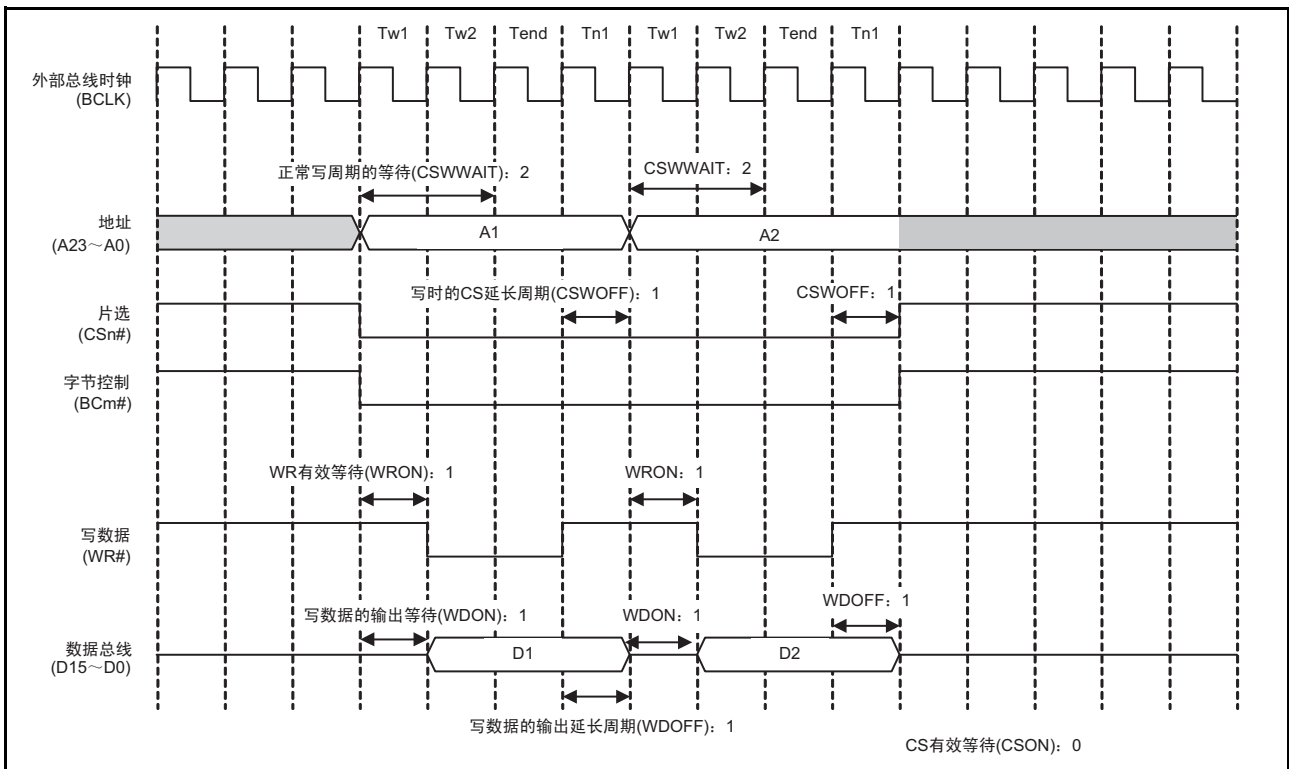


图 15.14 正常写存取的运行例子 (在1次写选通模式中, 1次传送请求发生2次总线存取的情况) (n=0 ~ 3、m=0、1)

通过 BCLK 引脚输出选择位设定 BCLK 的 2 分频时的运行例子如图 15.15 ~ 图 15.19 所示。

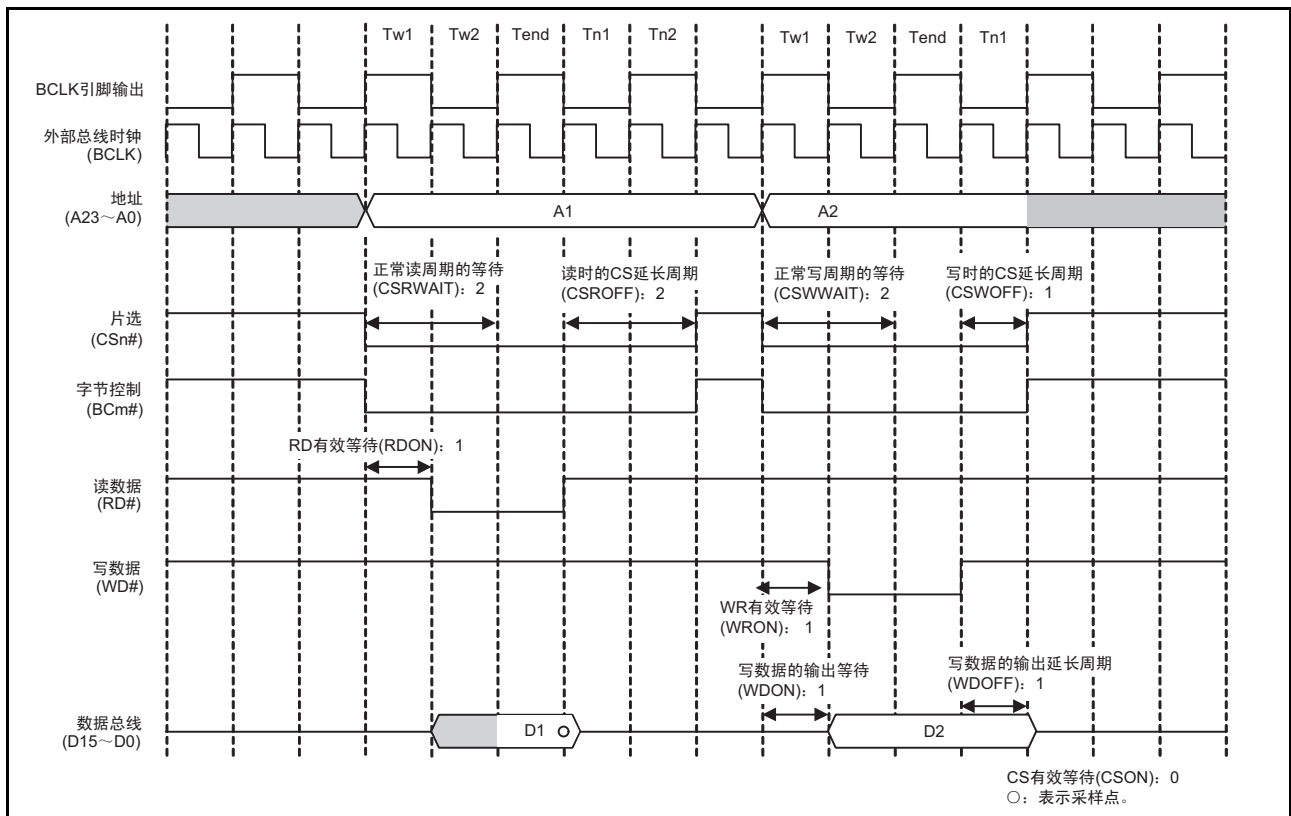


图 15.15 正常存取的运行例子（通过 BCLK 引脚输出选择位设定 BCLK 的 2 分频的情况）
(n=0 ~ 3、m=0、1)

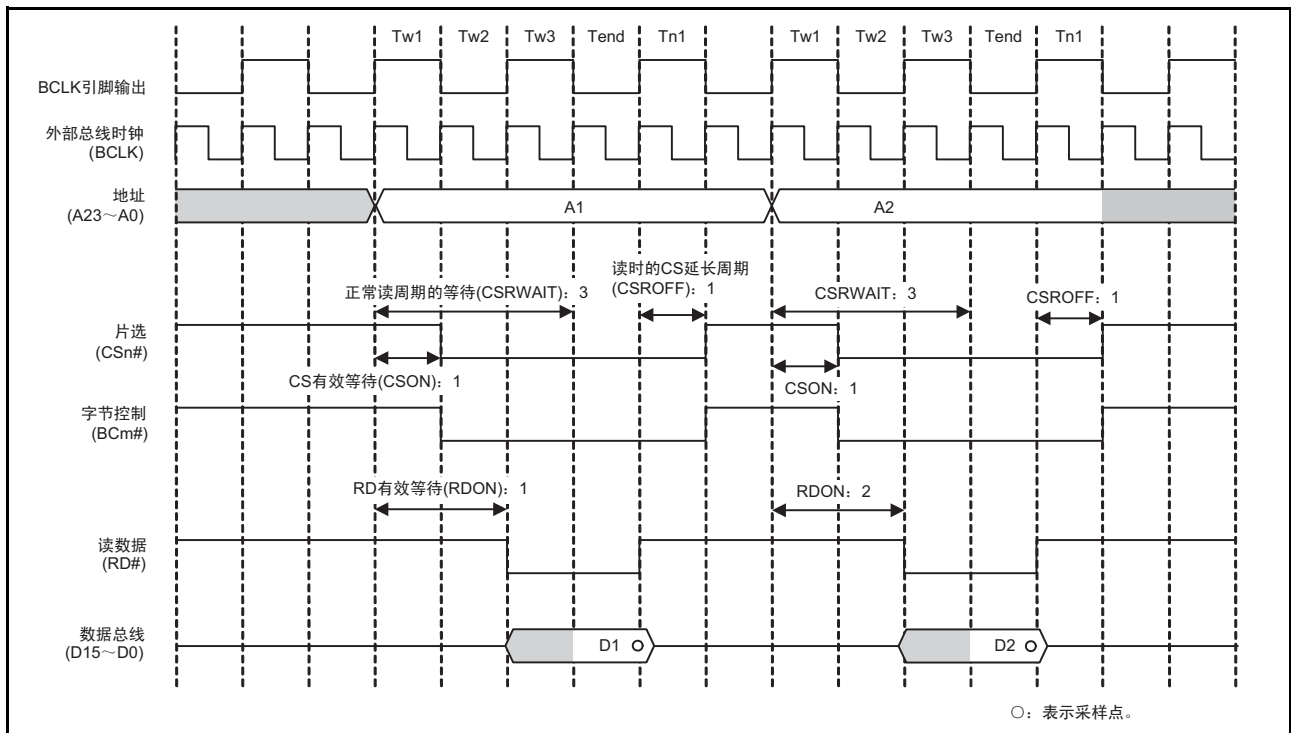


图 15.16 正常读存取的运行例子（通过 BCLK 引脚输出选择位设定 BCLK 的 2 分频的情况）
(n=0 ~ 3、m=0、1)

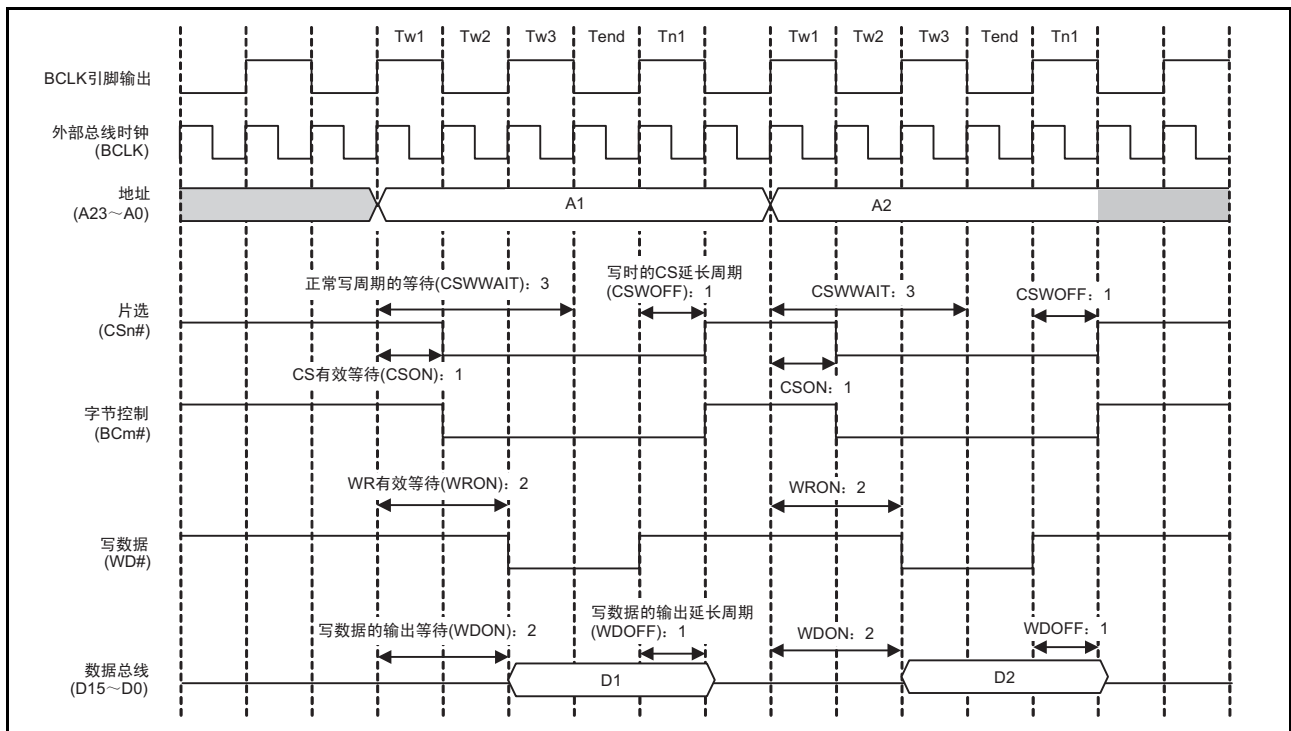


图 15.17 正常写存取的运行例子（通过 BCLK 引脚输出选择位设定 BCLK 的 2 分频的情况）
(n=0 ~ 3、m=0、1)

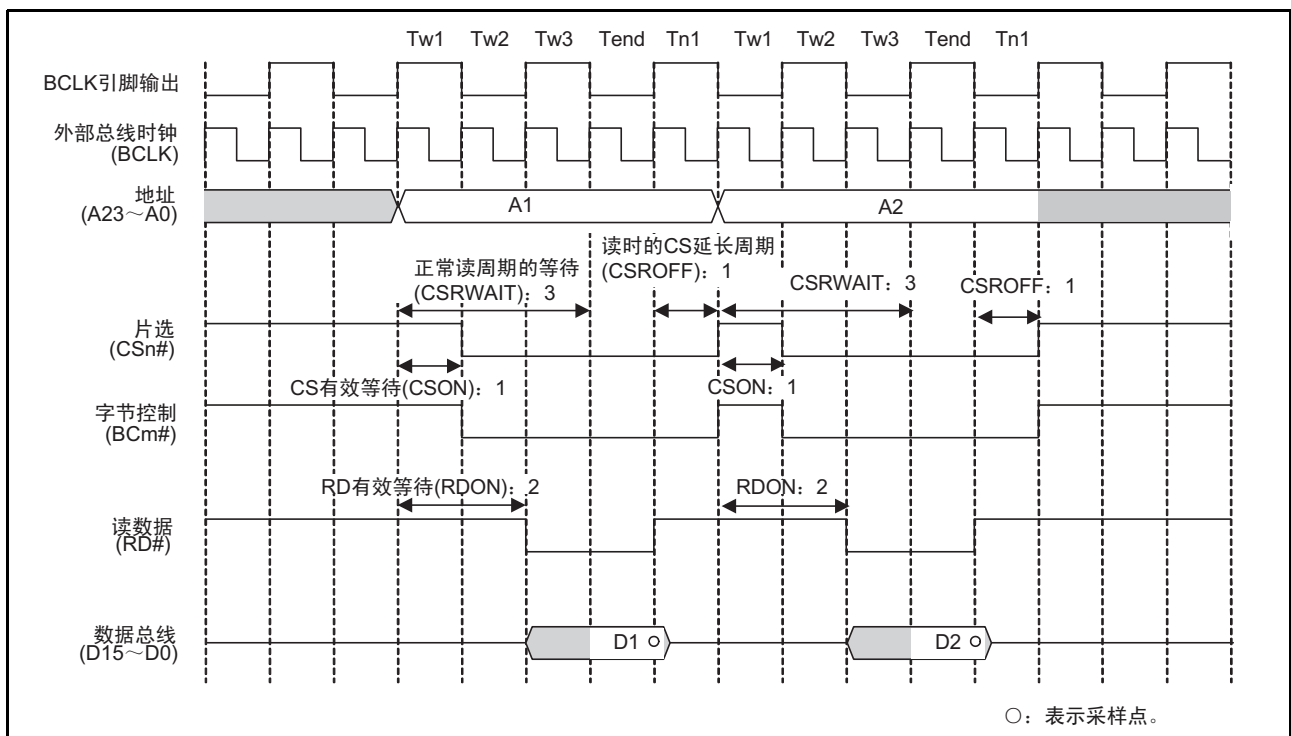


图 15.18 正常读存取的运行例子（通过 BCLK 引脚输出选择位设定 BCLK 的 2 分频并且 1 次传送请求发生 2 次总线存取的情况）(n=0 ~ 3、m=0、1)

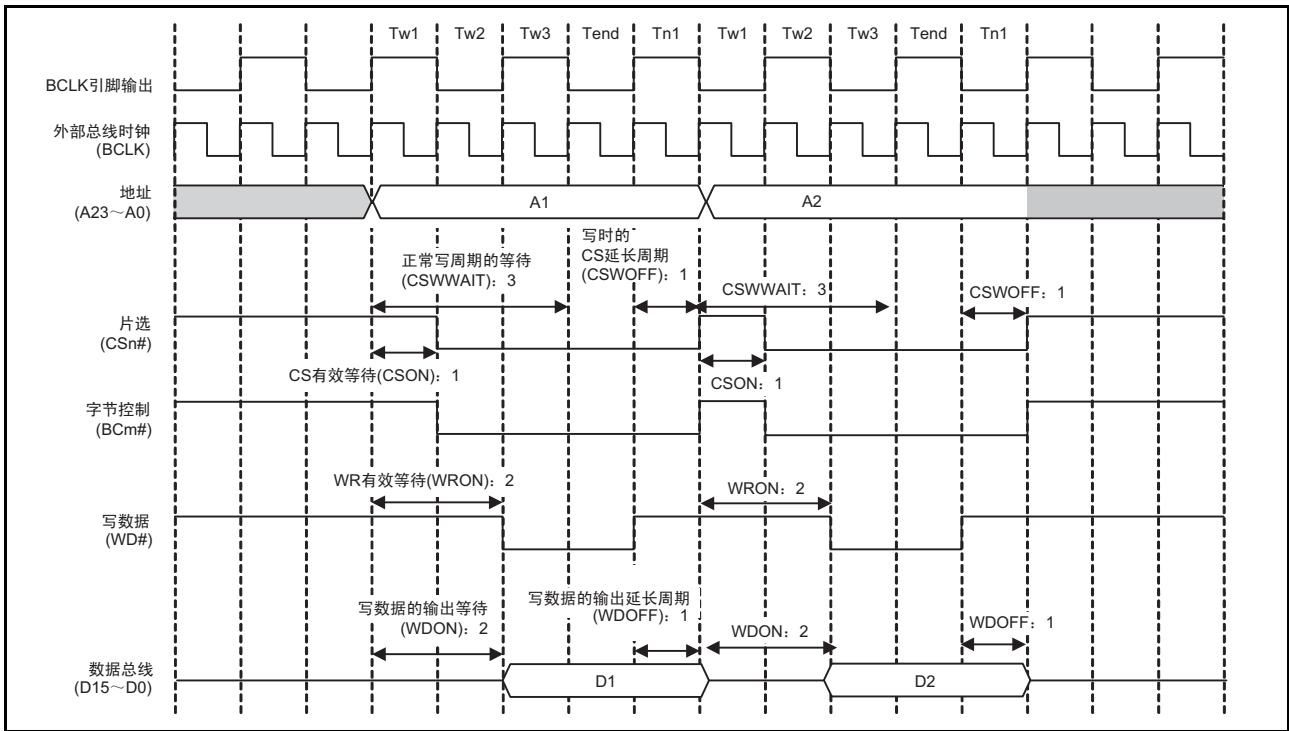


图 15.19 正常写存取的运行例子（通过 BCLK 引脚输出选择位设定 BCLK 的 2 分频并且 1 次传送请求发生 2 次总线存取的情况）(n=0 ~ 3、m=0、1)

(2) 页面存取

如果将 CSnMOD.PRENB 位置“1”（允许页面读存取）并且将 CSnMOD.PWENB 位置“1”（允许页面写存取），与页面存取对应的总线存取就为页面存取。在总线主控的 1 次传送请求至少需要 2 次外部总线存取时，为页面存取对象。如果被分割的存取为非定位存取并且是超过 32 位边界的存取，就不是页面存取而是正常存取的对象。有关发生页面存取的条件的内容，请参照图 15.6 ~ 图 15.9。

页面存取的运行例子如图 15.20 和图 15.21 所示。

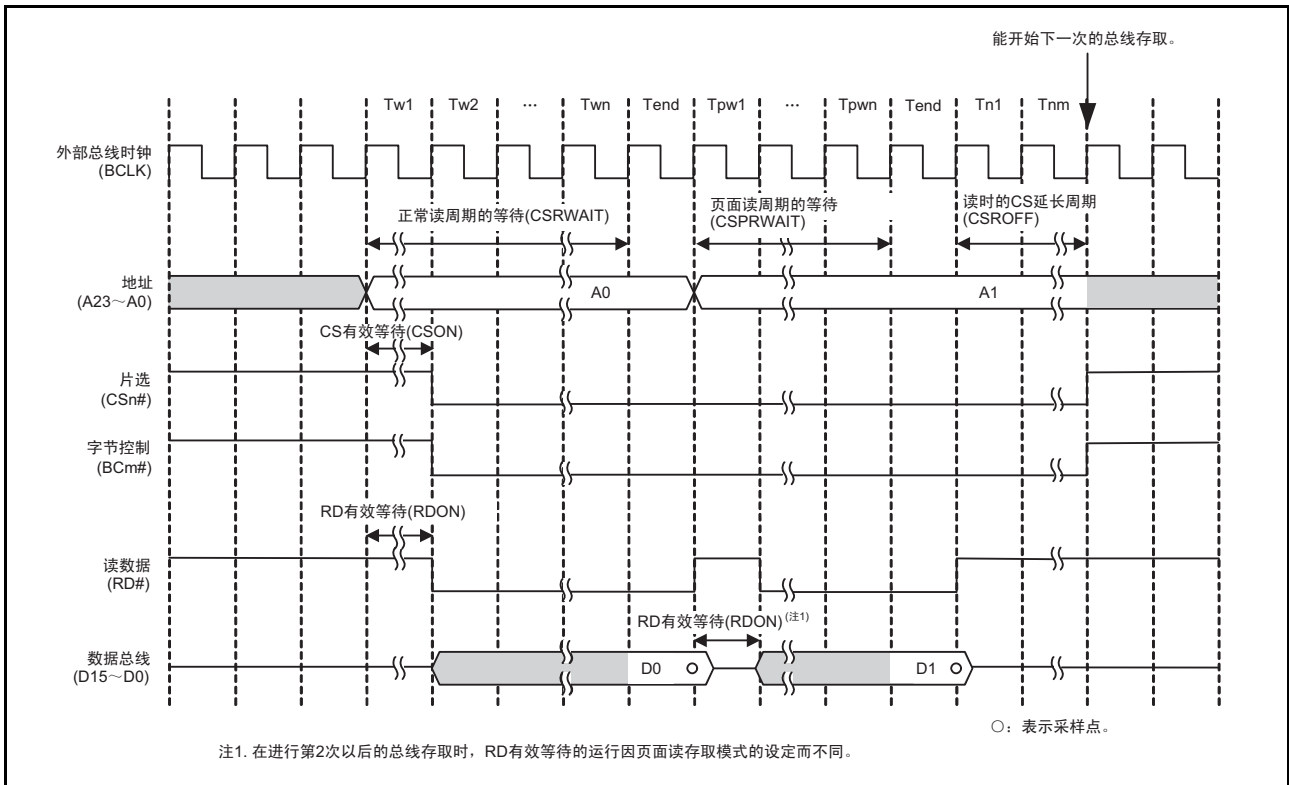


图 15.20 页面读存取时序 (n=0 ~ 3、m=0、1)

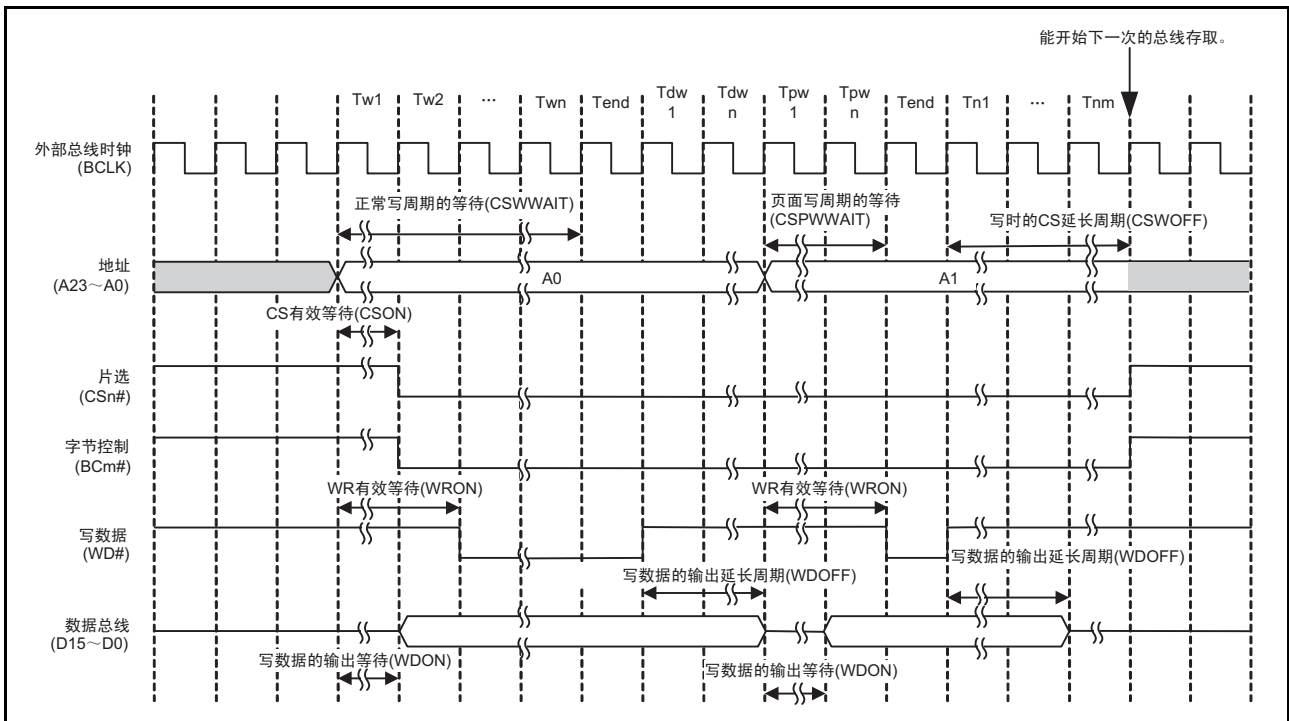


图 15.21 页面写存取时序 (n=0 ~ 3、m=0、1)

通过 BCLK 引脚输出选择位设定 BCLK 的 2 分频时的运行例子如图 15.22 和图 15.23 所示。

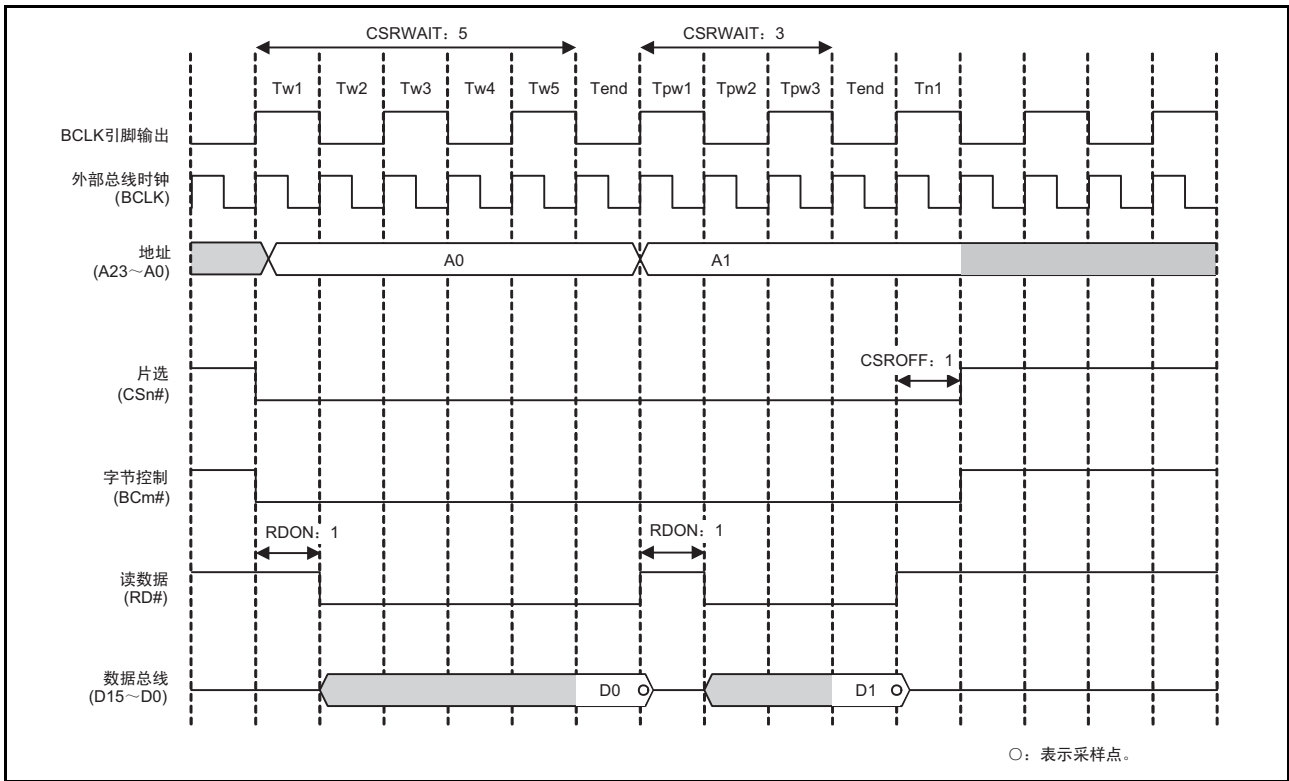


图 15.22 页面读存取的运行例子（通过 BCLK 引脚输出选择位设定 BCLK 的 2 分频并且 1 次传送请求发生 2 次总线存取的情况）（n=0 ~ 3）

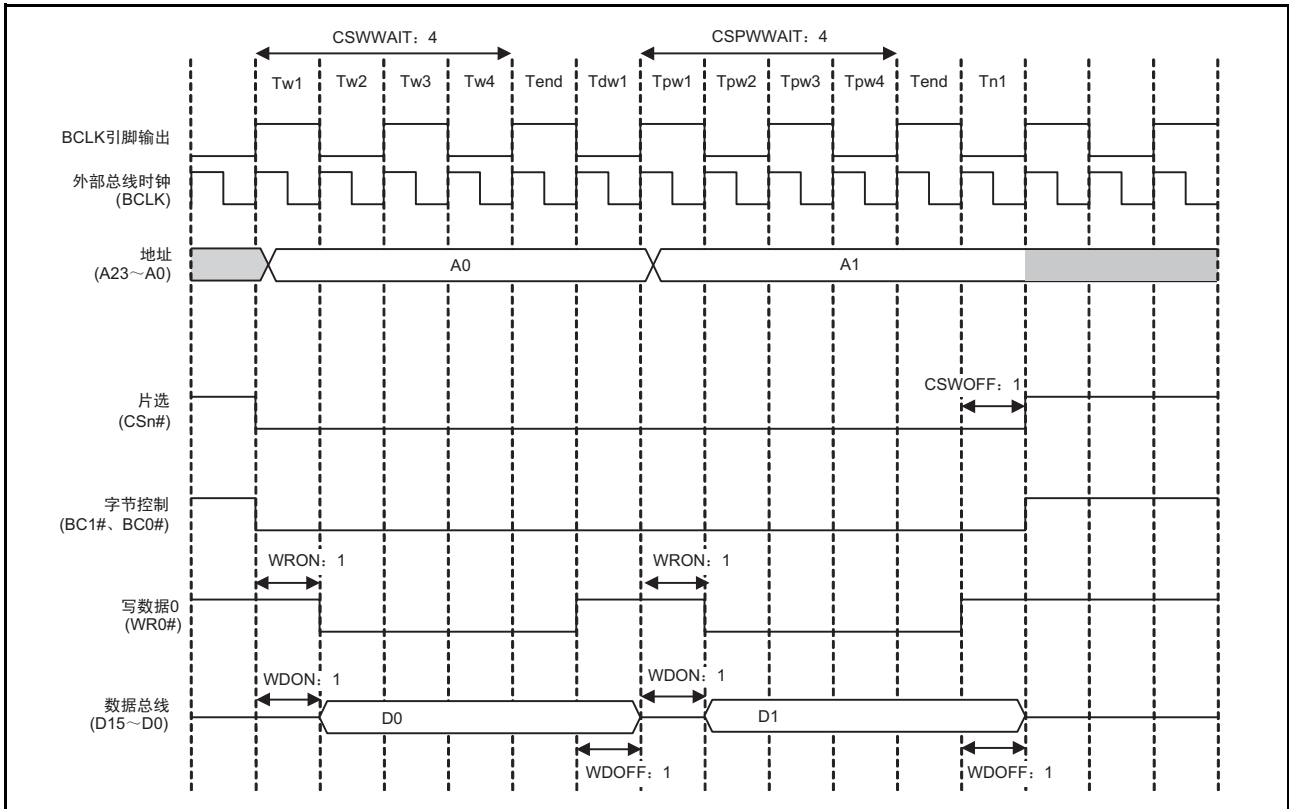


图 15.23 页面写存取的运行例子（在 1 次写选通模式中，通过 BCLK 引脚输出选择位设定 BCLK 的 2 分频并且 1 次传送请求发生 2 次总线存取的情况）（n=0 ~ 3）

15.5.2 地址 / 数据多路复用总线

能通过将 CSn 控制寄存器的地址 / 数据多路复用 I/O 接口选择位 (CS1CR.MPXEN) 置“1”，在对应的区域中将地址和数据进行多路复用后输入或者输出到 D15 ~ D0 引脚。因此，能将地址 / 数据多路复用需要的外围 LSI 直接连接本 LSI。在通过 CSnCR.BSIZE[1:0] 位选择 8 位的区域中，D7 ~ D0 和 A7 ~ A0 被多路复用；在通过 CSnCR.BSIZE[1:0] 位选择 16 位的区域中，D15 ~ D0 和 A15 ~ A0 被多路复用。在地址 / 数据多路复用 I/O 空间中，通过 ALE、RD# 和 WRn# 信号控制存取。

在地址 / 数据多路复用 I/O 空间的存取中，在地址输出周期插入 CSn 等待控制寄存器 2 的地址周期等待选择位 (CSnWCR2.AWAIT[1:0]) 设定的周期后，继续存取数据。

- Ta1 ~ Tan (地址周期等待)

这是从外部总线存取开始到地址锁存 (ALE) 信号无效的前 1 个周期为止的周期期间，此周期只在设定地址 / 数据多路复用 I/O 空间时有效，能选择 0 ~ 3 个周期。在 ALE 信号无效的下一个周期前输出地址 (地址周期)。ALE 信号和 CS# 信号的有效时序相同。地址周期之后是数据周期。为了使地址周期和数据周期不重叠，必须设定 CSnWCR1 寄存器和 CSnWCR2 寄存器的值。

在地址 / 数据多路复用 I/O 空间中，页面存取无效。如果设定为允许页面读存取 (CSnMOD.PRENB 位 =1) 并且允许页面写存取 (CSnMOD.PWENB 位 =1)，就忽视此设定而进行正常存取。

使用地址 / 数据多路复用 I/O 接口时的运行例子如图 15.24 ~ 图 15.26 所示。

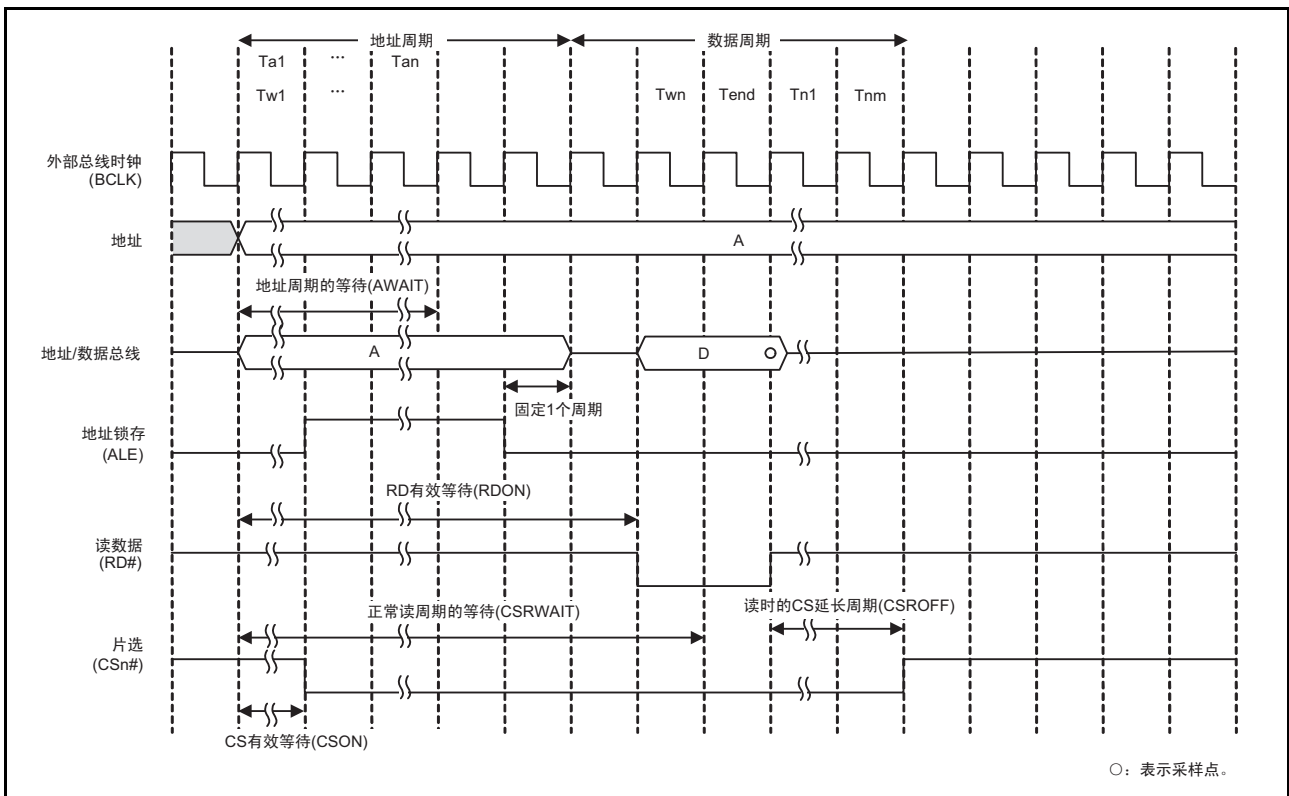


图 15.24 使用地址 / 数据多路复用 I/O 接口时的读存取的运行例子 (n=0 ~ 3)

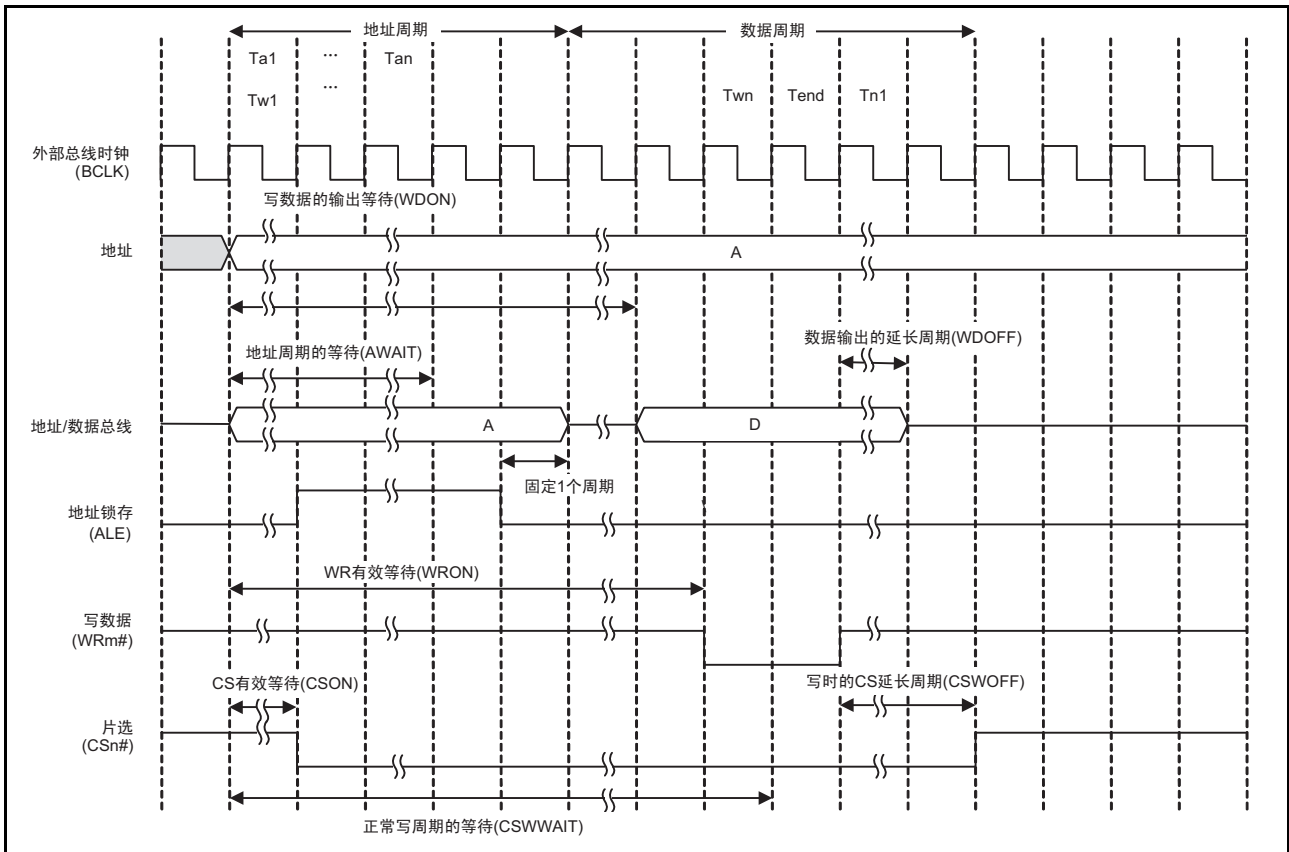


图 15.25 使用地址 / 数据多路复用 I/O 接口时的写存取的运行例子 (n=0 ~ 3、m=0、1)

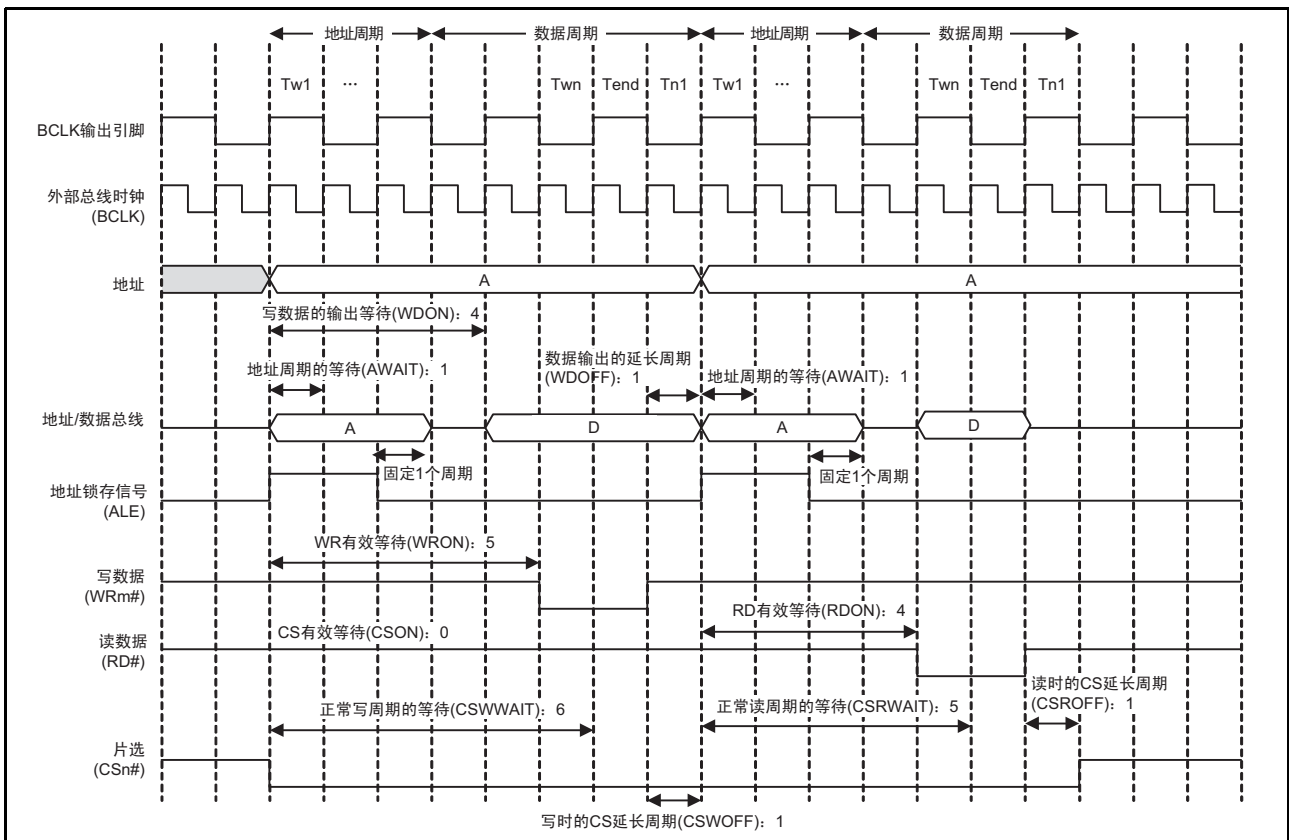


图 15.26 使用地址 / 数据多路复用 I/O 接口时的总线时序例子 (n=0 ~ 3、m=0、1)

当 1 次传送请求至少发生 2 次外部总线存取时，第 2 次以后的外部总线存取和第 1 次一样，重复地址周期和数据周期。详细内容请参照图 15.27 和图 15.28。

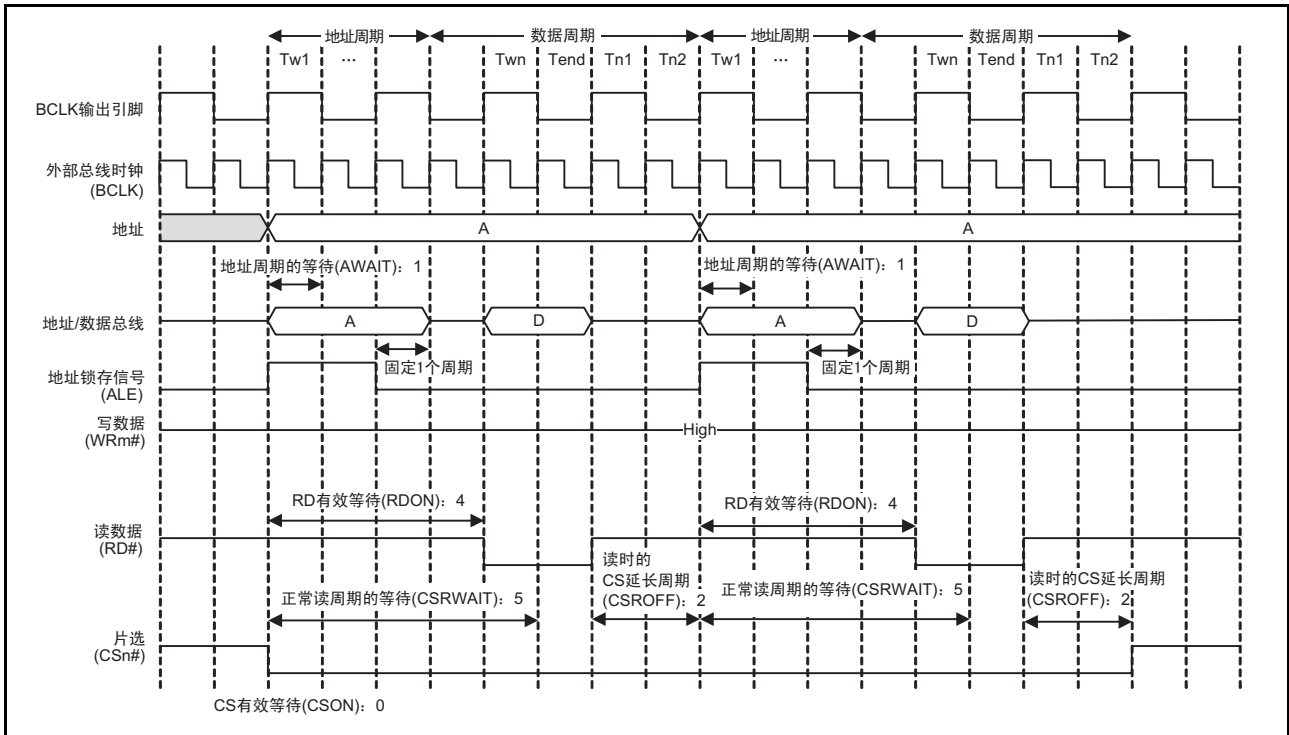


图 15.27 使用地址 / 数据多路复用 I/O 接口并且分割读存取时的运行例子 (n=0 ~ 3、m=0、1)

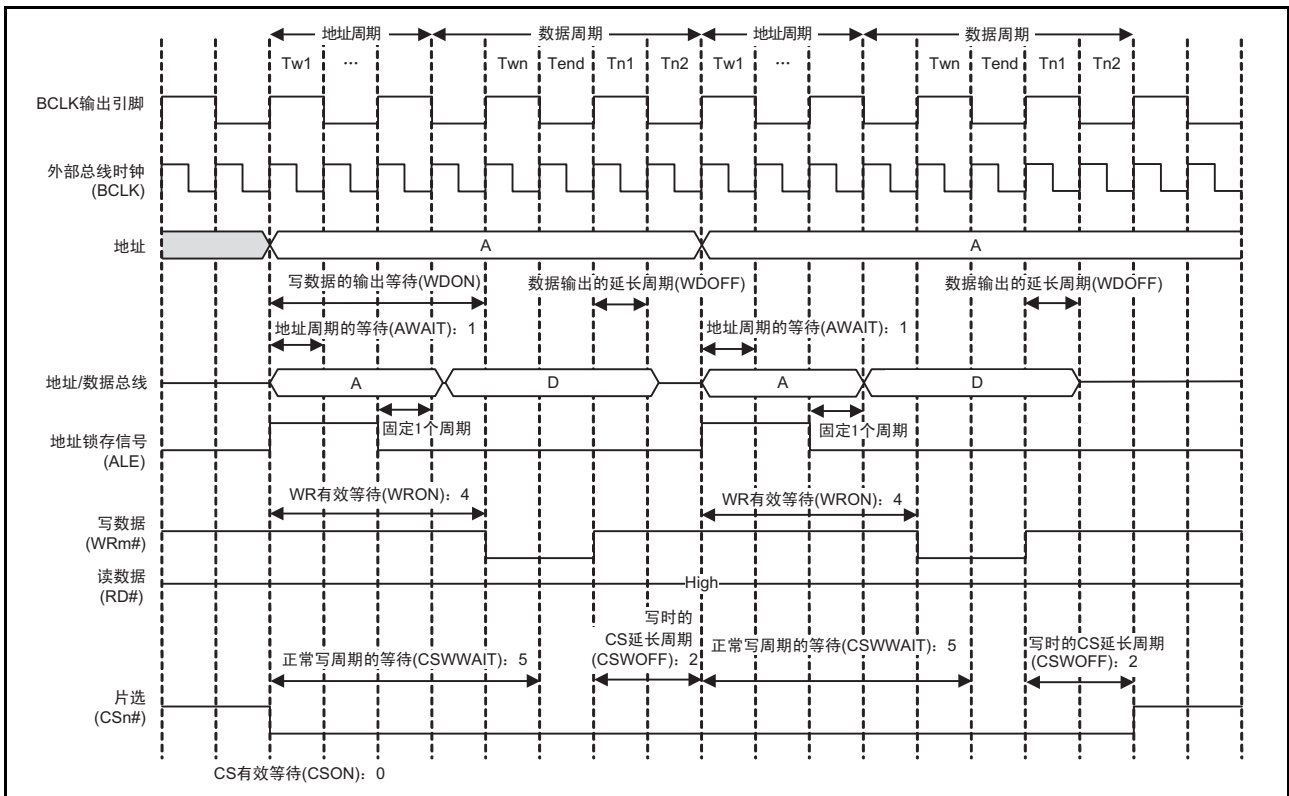


图 15.28 使用地址 / 数据多路复用 I/O 接口并且分割写存取时的运行例子 (n=0 ~ 3、m=0、1)

15.5.3 外部等待功能

能通过 WAIT# 信号将等待周期延长到超过 CSn 等待控制寄存器 1 (CSnWCR1) 设定的正常存取周期等待 (CSRWAIT[4:0] 位和 CSWWAIT[4:0] 位) 和页面存取周期等待 (CSPRWAIT[2:0] 位和 CSPWWAIT[2:0] 位)。

如果设定为允许外部等待 (CSnMOD.EWENB 位 =1)，就在 WAIT# 信号为 Low 电平期间插入等待周期。如果禁止外部等待 (CSnMOD.EWENB 位 =0)，WAIT# 信号就无效。

与 WAIT# 信号无关，插入 CSnWCR1 寄存器设定的各周期等待。

(1) 正常存取

从 CSnWCR1 寄存器设定的周期等待数结束时 (Tend) 开始对 WAIT# 信号进行采样。在 WAIT# 信号为 Low 电平期间延长总线周期，WAIT# 信号变为 High 电平后的下一个周期为等待周期的结束 (Tend)。

(2) 页面存取

最初的存取和正常存取相同。从等待控制寄存器设定的周期等待数结束时 (Tend) 开始对 WAIT# 信号进行采样。在 WAIT# 信号为 Low 电平期间延长总线周期，WAIT# 信号变为 High 电平后的下一个周期为等待周期的结束 (Tend)。

对于第 2 次以后的存取，从页面存取的等待周期结束时 (Tend) 开始对 WAIT# 信号进行采样。在 WAIT# 信号为 Low 电平期间延长页面存取的等待周期，WAIT# 信号变为 High 电平后的下一个周期为等待周期的结束 (Tend)。

使用分离总线接口时的外部等待周期的运行例子如图 15.29 和图 15.30 所示。

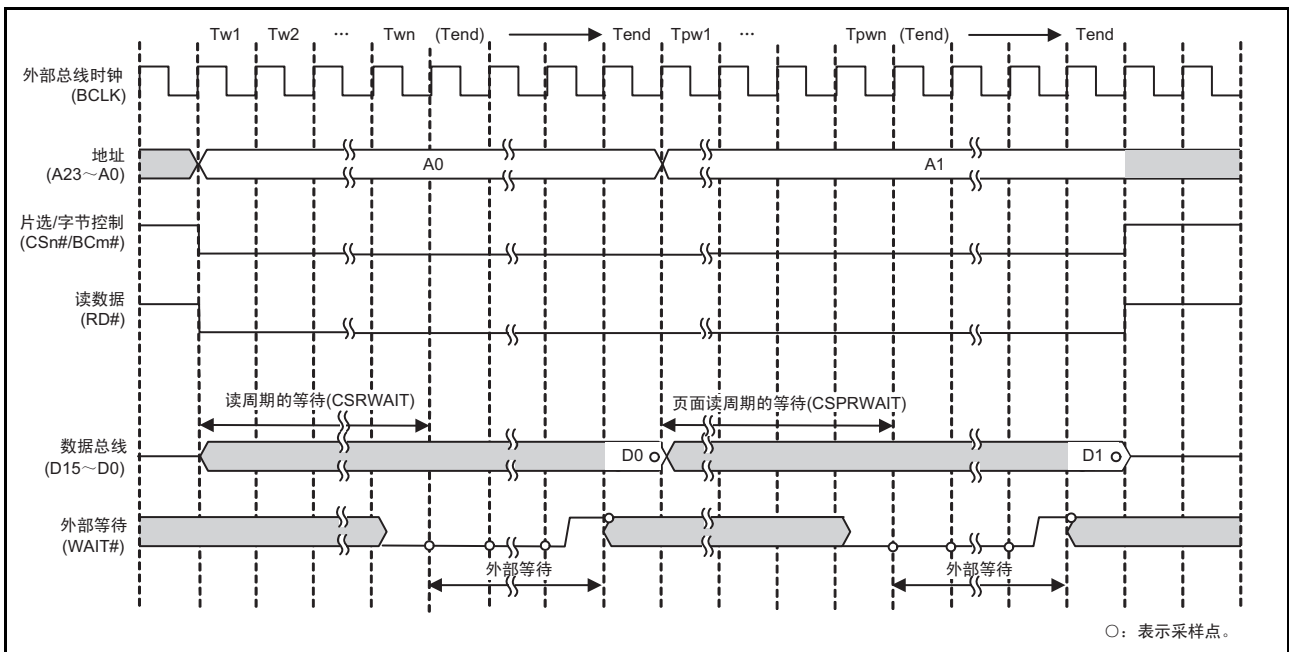


图 15.29 外部等待时序例子 (对 16 位总线空间进行页面读存取) (n=0 ~ 3、m=0、1)

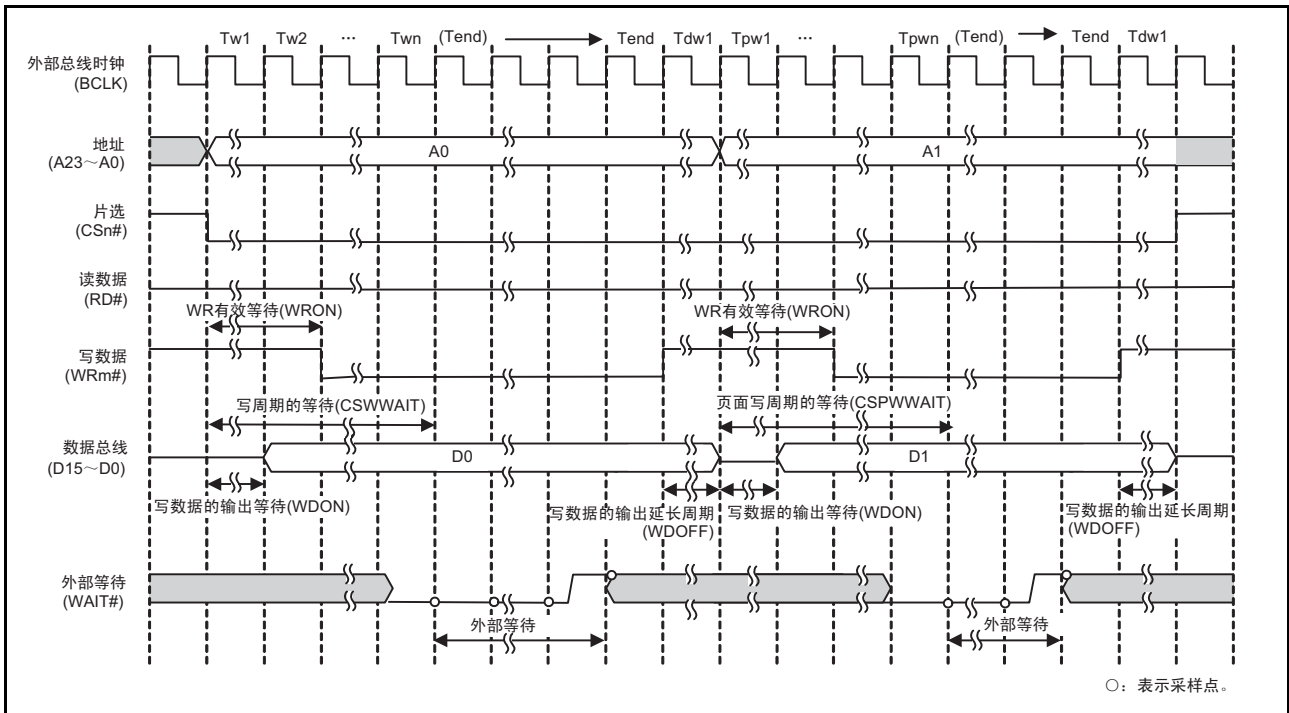


图 15.30 外部等待时序例子（在字节选通模式中，对 16 位总线空间进行页面写存取）（ $n=0 \sim 3, m=0、1$ ）

(3) 地址 / 数据多路复用 I/O 接口

和分离总线接口一样，能在地址 / 数据多路复用 I/O 接口的数据周期插入编程等待，还能通过 WAIT 引脚插入引脚等待。

等待控制的设定不影响地址周期。使用地址 / 数据多路复用 I/O 接口时的外部等待周期的运行例子如图 15.31 所示。

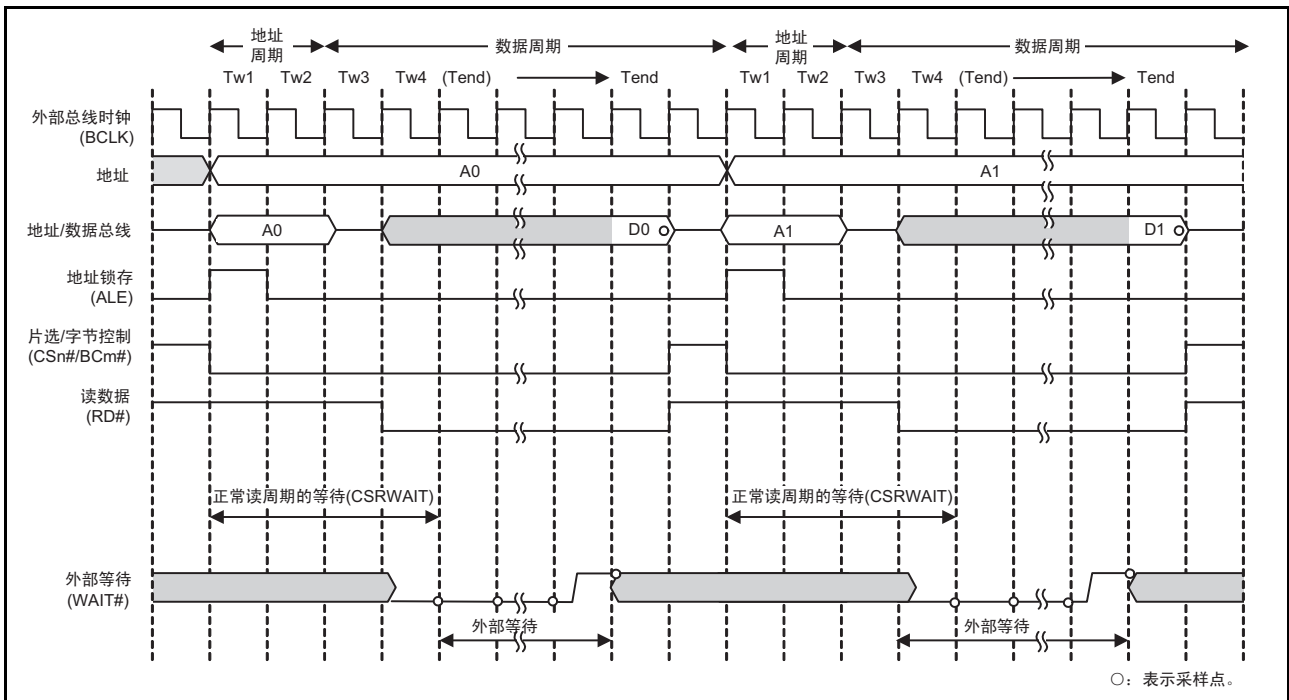


图 15.31 使用地址 / 数据多路复用 I/O 接口时的外部等待插入的运行例子（ $m=0、1$ ）

15.5.4 恢复周期的插入

能通过将 CS 恢复周期插入允许寄存器 (CSRECEN) 的恢复周期插入允许位置“1”，在连续的外部总线存取期间插入恢复周期。

能在写周期后和读周期后通过 CSn 恢复周期设定寄存器 (CSnREC) 设定各区域要插入的恢复周期数。在上一个总线周期为写存取的情况下，通过该区域的写恢复周期设定位 (CSnREC.WRCV[3:0]) 进行设定；在上一个总线周期为读存取的情况下，通过该区域的读恢复周期设定位 (CSnREC.RRCV[3:0]) 进行设定。例如，如果在 CS0 读存取后进行 CS1 读存取，在此期间插入的恢复周期就为 CS0 的 CS0REC.RRCV[3:0] 位设定的周期数。

能插入恢复周期的条件有以下 8 种。在上次总线存取为分离总线的情况下，能通过 CS 恢复周期插入允许寄存器的分离总线恢复周期插入允许位 (CSRECEN.RCVENj) (j=0 ~ 7) 设定允许或者禁止插入恢复周期；在上次总线存取为地址 / 数据多路复用总线的情况下，能通过 CS 恢复周期插入允许寄存器的多路复用总线恢复周期插入允许位 (CSRECEN.RCVENMj) (j=0 ~ 7) 设定允许或者禁止插入恢复周期。

- 在对外部总线进行读存取后有相同区域的外部总线的读存取时
- 在对外部总线进行读存取后有不同区域的外部总线的读存取时
- 在对外部总线进行读存取后有相同区域的外部总线的写存取时
- 在对外部总线进行读存取后有不同区域的外部总线的写存取时
- 在对外部总线进行写存取后有相同区域的外部总线的读存取时
- 在对外部总线进行写存取后有不同区域的外部总线的读存取时
- 在对外部总线进行写存取后有相同区域的外部总线的写存取时
- 在对外部总线进行写存取后有不同区域的外部总线的写存取时

上次总线周期结束时 (CSn# 信号 (n=0 ~ 3) 无效) 是恢复周期的起点。在从结束时开始到设定的恢复周期期间插入 CSn# 信号的 High 电平期间。

最快在恢复周期结束后，下一次的总线存取的片选 CSn# 信号有效。即使在恢复周期期间发生下一个外部地址区的存取请求，也在恢复周期结束后开始下一次外部总线存取。

在总线主控的 1 次传送请求至少需要 2 次外部总线存取并且恢复周期插入条件成立时，在总线存取中途也插入恢复周期。如果设定为允许页面读存取 (CSnMOD.PRENB=1) 或者允许页面写存取 (CSnMOD.PWENB=1)，即使恢复周期插入条件成立，也不在总线存取中途插入恢复周期而只在最后传送的总线存取中途插入恢复周期 (参照图 15.34)。

在设定为允许页面存取并且正常存取的情况下，当使用分离总线接口时，不在总线存取中途插入恢复周期而只在最后传送的总线存取中途插入恢复周期；当使用地址 / 数据多路复用 I/O 接口时，与是否允许页面存取无关，在恢复周期插入条件成立时，即使在总线存取中途也插入恢复周期。

使用分离总线接口时的恢复周期的运行例子如图 15.32 ~图 15.34 所示。

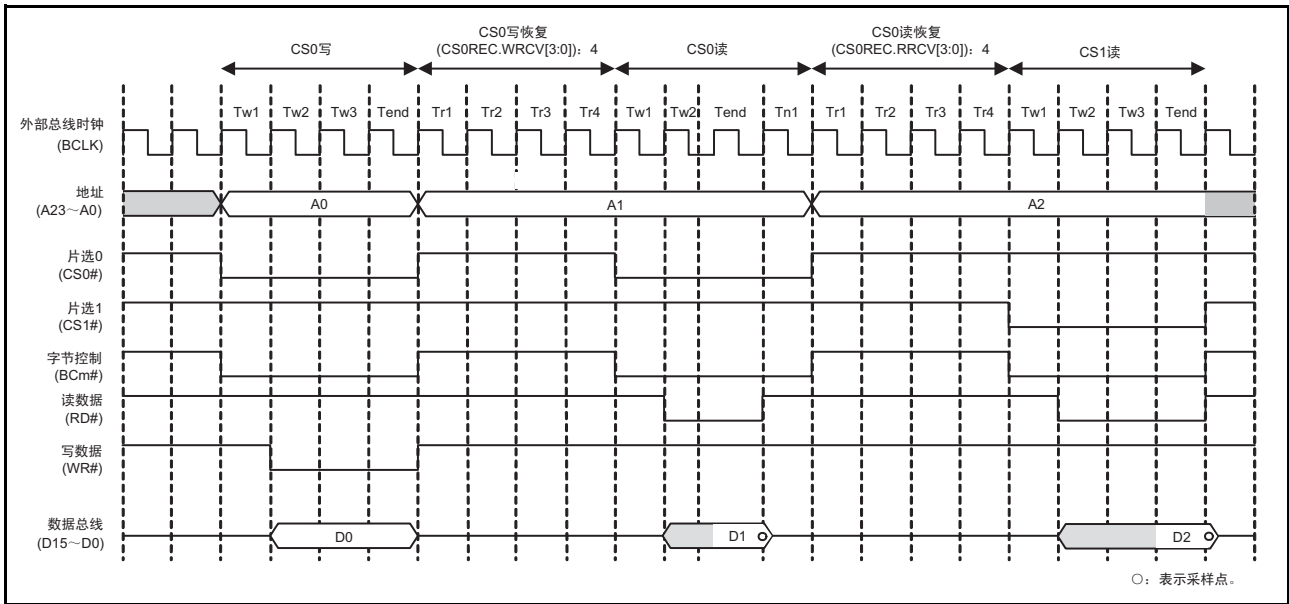


图 15.32 使用分离总线接口时的恢复周期的运行例子 (m=0、1)

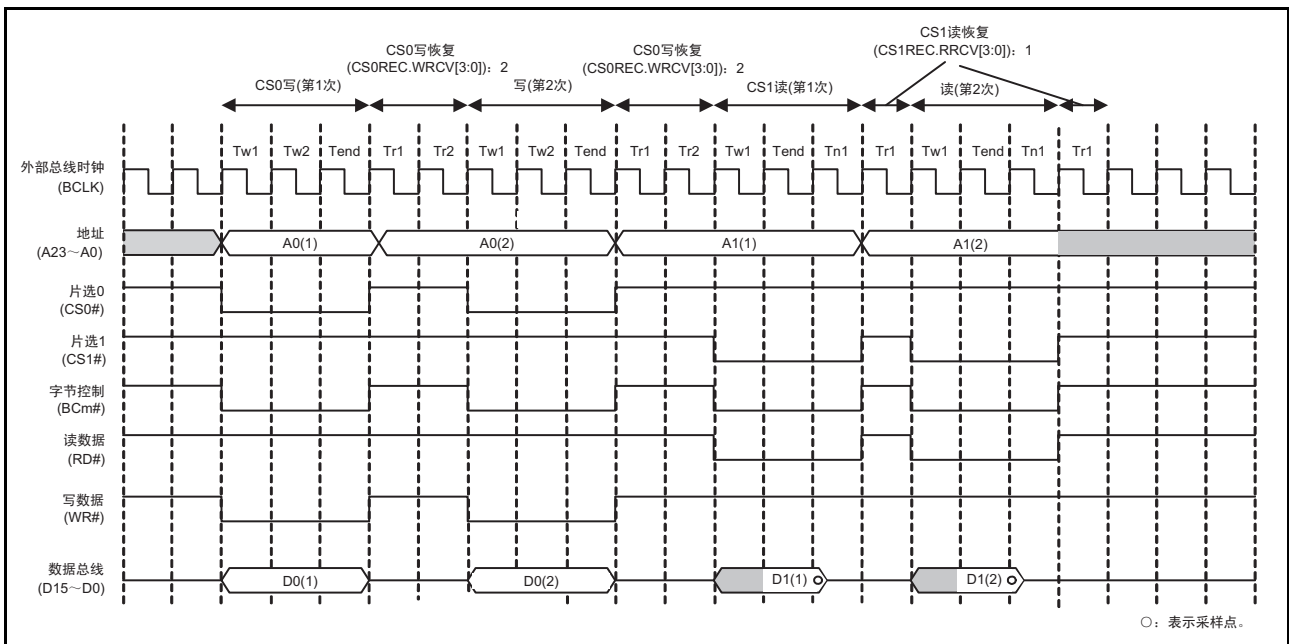


图 15.33 分割总线存取时的恢复周期的运行例子 (分离总线接口, 正常存取) (m=0、1)

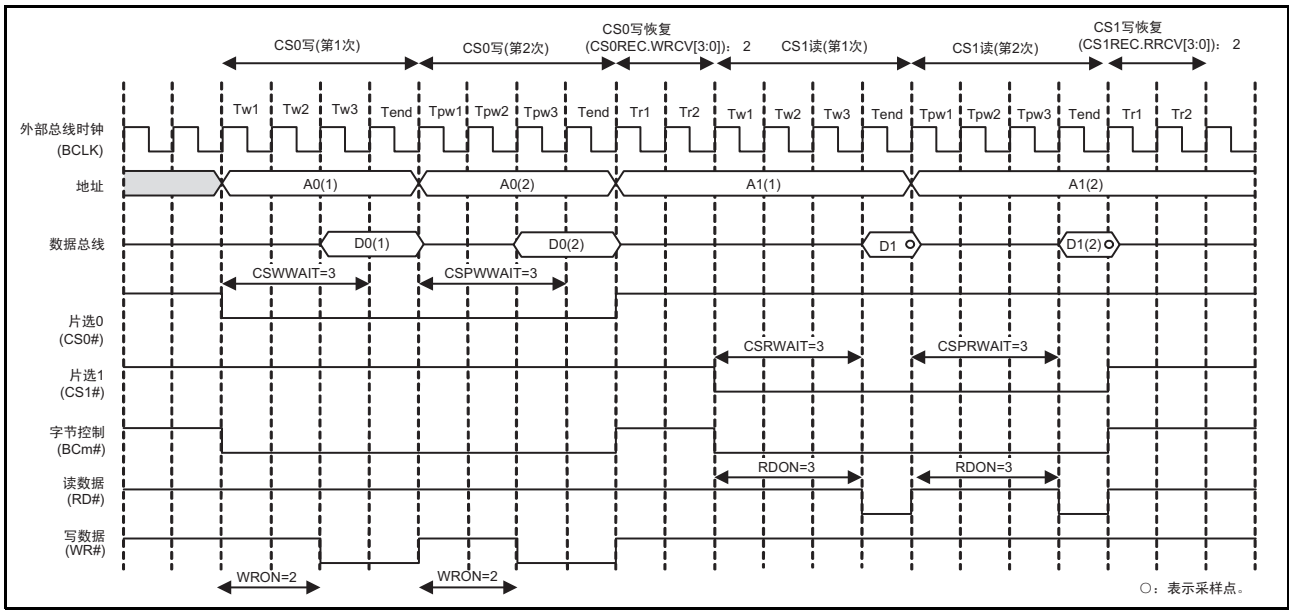


图 15.34 分割总线存取时的恢复周期的运行例子（分离总线接口，页面存取）（m=0、1）

和分离总线接口一样，也插入地址 / 数据多路复用 I/O 接口的恢复周期。使用地址 / 数据多路复用 I/O 接口时的恢复周期的运行例子如图 15.35 和图 15.36 所示。

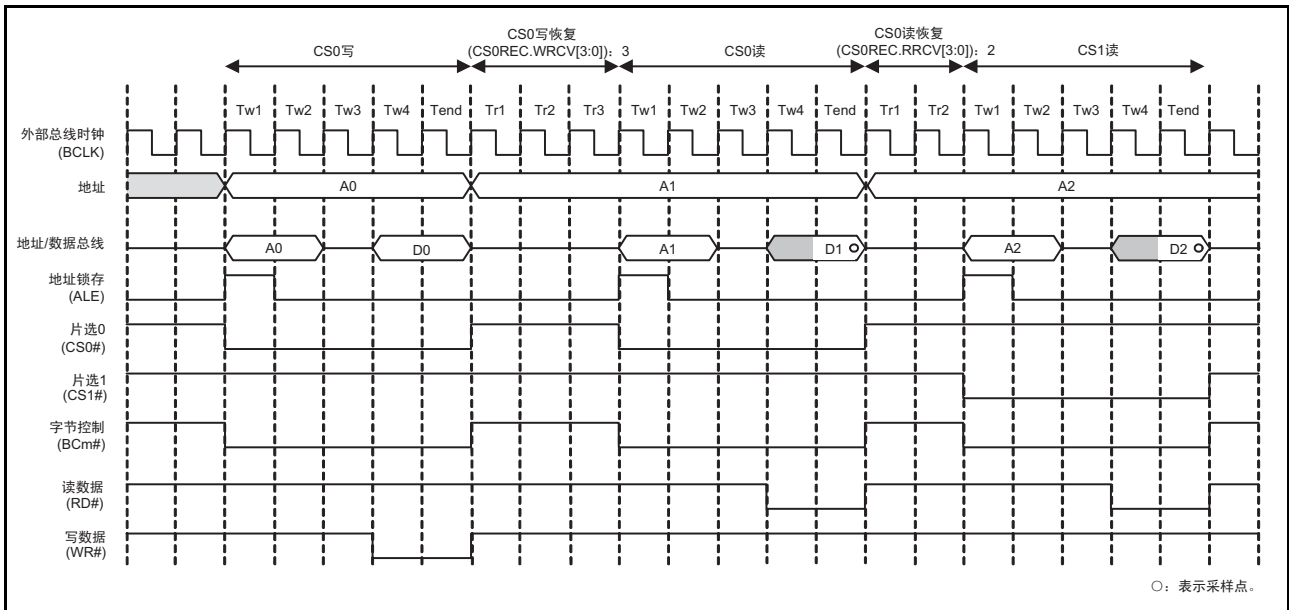


图 15.35 使用地址 / 数据多路复用 I/O 接口时的恢复周期的运行例子（m=0、1）

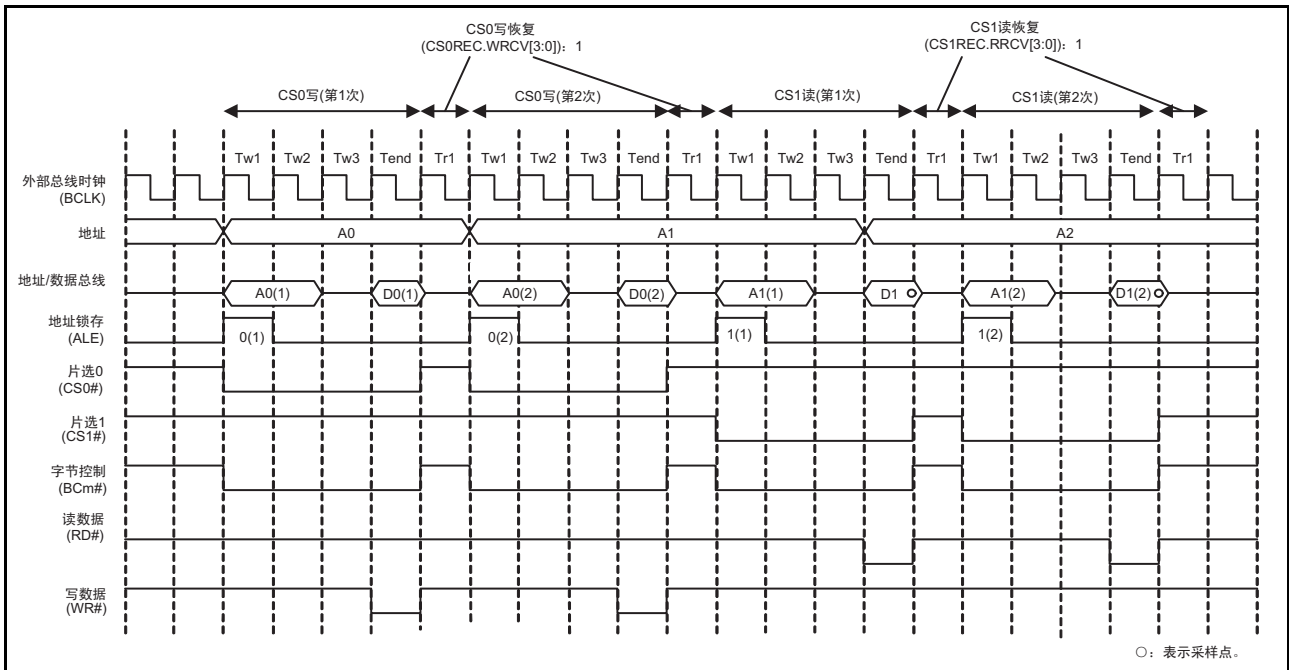


图 15.36 使用地址 / 数据多路复用 I/O 接口并且分割总线存取时的恢复周期的运行例子 (m=0、1)

15.5.5 不存取时的状态

在尚未对外部地址空间进行处理时，CSn# 信号、BCn# 信号、WRn# 信号和 RD# 信号为 High 电平，ALE 信号为 Low 电平，D15 ~ D0 为高阻抗。

15.5.6 写缓冲器功能

在进行写存取时，通过将数据写到写缓冲器来释放内部主总线而不要等到运行结束，因此能开始下一次的总线存取。如果下一次总线存取是存取外部地址空间或者外部总线控制器的寄存器，就要等到上一次外部总线运行结束。

使用写缓冲器功能时的运行例子如图 15.37 所示。如果使用此功能并且在进行外部写操作以后有内部存取，就不等到外部写操作的结束而并行执行内部存取（内部存储器和外围模块的存取）。

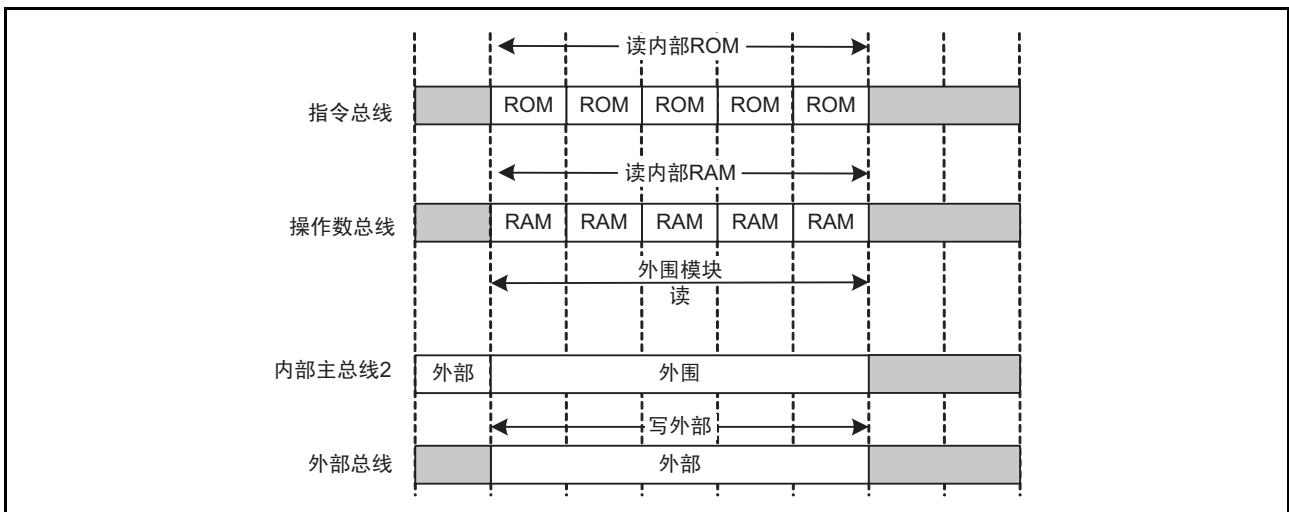


图 15.37 使用写缓冲器功能时的运行例子

15.6 限制事项

15.6.1 使用分离总线接口时的限制事项

1. 正常存取时和页面存取时的CSn等待控制寄存器1（CSnWCR1）和CSn等待控制寄存器2（CSnWCR2）的各位设定值的限制事项如表15.9所示。即使CSn模式寄存器的页面读存取允许位为有效（CSnMOD.PRENB=1）或者页面写存取允许位为有效（CSnMOD.PWENB=1），在进行页面存取的第1次存取或者不作为页面存取对象的存取时，也作为正常存取并且需要满足正常存取的限制条件。

表 15.9 正常存取 / 页面存取时的限制事项

正常存取时的限制条件		页面存取时的限制条件	
读	写	读	写
CSO[2:0] ≤ CSRWAIT RDO[2:0] ≤ CSRWAIT CSO[2:0] ≤ RDO	1 ≤ WDO[2:0] CSO[2:0] ≤ CSWAIT WRO[2:0] ≤ CSWAIT WDO[2:0] ≤ CSWAIT WDOFF[2:0] ≤ CSWOFF WDO[2:0] ≤ WRO CSO[2:0] ≤ WRO	CSO[2:0] ≤ CSPWAIT RDO[2:0] ≤ CSPWAIT CSO[2:0] ≤ RDO	1 ≤ WDO[2:0] CSO[2:0] ≤ CSPWAIT WRO[2:0] ≤ CSPWAIT WDO[2:0] ≤ CSPWAIT WDOFF[2:0] ≤ CSWOFF WDO[2:0] ≤ WRO CSO[2:0] ≤ WRO

2. 在总线主控的1次传送请求至少需要2次外部总线存取并且恢复周期插入条件成立时，如果设定为允许页面读存取（CSnMOD.PRENB=1）或者允许页面写存取（CSnMOD.PWENB=1），就不在总线存取中途插入恢复周期而只在最后传送的总线存取中途插入恢复周期。

15.6.2 使用地址 / 数据多路复用总线时的限制事项

1. 在地址/数据多路复用I/O空间中，页面存取无效。忽视页面存取的设定而进行正常存取。
2. 如果设定地址/数据多路复用I/O接口，就通过CSn控制寄存器的外部总线宽度选择位（CSnCR.BSIZE[1:0]）禁止设定为32位总线空间，否则不保证运行。

表 15.10 正常存取时的限制事项

正常存取时的限制条件	
读	写
CSO[2:0] ≤ CSRWAIT RDO[2:0] ≤ CSRWAIT CSO[2:0] ≤ RDO AWAIT[1:0]+2 ≤ RDO CSO[2:0] ≤ AWAIT	CSO[2:0] ≤ CSWAIT WRO[2:0] ≤ CSWAIT WDO[2:0] ≤ CSWAIT WDOFF[2:0] ≤ CSWOFF WDO[2:0] ≤ WRO CSO[2:0] ≤ WRO AWAIT[1:0]+2 ≤ WRO AWAIT[1:0]+2 ≤ WDO CSO[2:0] ≤ AWAIT

15.6.3 兼用 A0 引脚和 BC0# 引脚的产品时的限制事项

对于兼用 A0 引脚和 BC0# 引脚的产品，禁止对 8 位总线空间设定 1 次写选通模式，否则不保证运行。

15.6.4 通过 BCLK 引脚输出选择位设定 BCLK 的 2 分频时的限制事项

如果通过 BCLK 引脚输出选择位设定 BCLK 的 2 分频，就在 BCLK 引脚输出的上升沿开始存取外部总线。当总线主控的 1 次传送请求至少发生 2 次外部总线存取时，根据等待数的设定，有可能在 BCLK 引脚输出的下降沿开始第 2 次以后的外部总线存取。必须根据要连接的设备规格设定寄存器的值。

15.6.5 禁止存取跨地址空间的各区域

禁止在 1 次存取中存取跨地址空间的多个区域，否则不保证运行。1 个字或者长字的存取不能隔着地址空间的各区域边界而跨 2 个区域。

15.6.6 有关 RMPA 指令和字符串操作指令的限制事项

1. 外部空间的各区域有字节序转换功能（只限于数据），但是禁止将 RMPA 指令和字符串操作指令的操作对象数据分配到设定了不同于芯片字节序的区域，否则不保证运行。要将 RMPA 指令和字符串操作指令的操作对象数据分配到外部空间时，必须分配到设定了和芯片字节序相同字节序的区域。
2. 禁止将 RMPA 指令和字符串操作指令的操作对象数据分配到 I/O 寄存器，否则不保证运行。

15.6.7 有关指令代码的限制事项

不能将指令代码分配到设定了和芯片字节序不同字节序的各区域。要将指令代码分配到外部空间时，必须分配到设定了和芯片字节序相同字节序的区域。

15.7 总线错误监视部

总线错误监视部监视各区域的总线错误并且在发生总线错误时通知总线主控。

15.7.1 总线错误的种类

总线错误有非法地址存取和超时两种。

当存取非法地址时被检测为非法区域的存取；当总线存取在 768 个周期以内不结束时被检测为超时。

15.7.1.1 非法地址的存取

当总线错误监视允许寄存器的非法地址存取检测允许位被设定为有效（BEREN.IGAEN=1）并且发生以下存取时，发生非法地址存取。

- 对外部区域存取了被设定为禁止（CSnCR.EXENB=0）运行的区域时
- 对上述以外的区域存取了非法地址区域时
非法地址存取错误的发生区域如表 15.11 所示。

15.7.1.2 超时

在总线错误监视允许寄存器的超时检测允许位被设定为有效（BEREN.TOEN=1）并且总线存取在 768 个周期以内不结束时，发生超时。

- CS区域（CS0～CS3）：在开始总线存取后，总线存取在外部总线时钟（BCLK）的768个周期以内不结束时（WAIT#信号不变为无效）
如果发生超时，就在BCLK的256个周期内不接受总线主控的存取。在总线主控的1次请求发生多次外部总线存取的传送过程中，即使发生超时也不能中止后续的总线存取。此时，有可能连续发生超时。
- 内部外围总线（2）：在开始总线存取后，总线存取在外围模块时钟（PCLKB）的768个周期以内不结束时
如果发生超时，就在PCLKB的256个周期内不接受总线主控的存取。
- 内部外围总线6：在开始总线存取后，总线存取在FlashIF时钟（FCLK）的768个周期以内不结束时
如果发生超时，就在FCLK的256个周期内不接受总线主控的存取。

15.7.2 发生总线错误时的运行

一旦发生总线错误，就将总线错误通知 CPU。当发生总线错误时，不保证运行。

- 向CPU通知总线错误的发生
发生中断。能通过ICU.IERn寄存器控制是否产生中断。

15.7.3 总线错误的发生条件

地址空间各区域发生的总线错误的种类如表 15.11 所示。

在没有发生总线错误的状态（清除总线错误状态寄存器 n（BERSRn）（n=1、2）的状态）下，如果检测到非法地址存取错误或者超时，就将当时的状态保存到 BERSRn 寄存器。一旦发生总线错误，即使以后还发生总线错误，如果没有清除 BERSRn，就不保存当时的状态。

当 2 个或者 2 个以上的总线主控同时发生总线错误时，只保存 1 个总线主控的信息。在发生总线错误后，状态保持到 BERSRn 寄存器被清除为止。

表 15.11 发生的总线错误种类

地址	内容		种类			
			非法地址存取		超时	
	内部 ROM 模式		内部 ROM 模式		内部 ROM 模式	
有效	无效	有效	无效	有效	无效	
0000 0000h ~ 0007 FFFFh	存储器总线 1		—		—	
0008 0000h ~ 0008 7FFFh	内部外围总线 1		—		—	
0008 8000h ~ 0009 FFFFh	内部外围总线 2		△		—	
000A 0000h ~ 000B FFFFh	保留区		—		—	
000C 0000h ~ 000D FFFFh	保留区		△		—	
000E 0000h ~ 000F FFFFh	保留区		△		—	
0010 0000h ~ 00FF FFFFh	内部外围总线 6	保留区	△	○	—	—
0500 0000h ~ 07FF FFFFh	外部总线（CS1 ~ CS3）		[IA]		[TO]	
0800 0000h ~ 0FFF FFFFh	保留区		—		—	—
1000 0000h ~ 7FFF FFFFh	保留区		○		—	—
8000 0000h ~ FEFF FFFFh	存储器总线 2	保留区	—	○	—	—
FF00 0000h ~ FF7F FFFFh		外部总线（CS0）	—	[IA]	—	[TO]
FF80 0000h ~ FFFF FFFFh			—		—	

—：不发生总线错误。

△：总线错误不确定。

○：发生总线错误。

[IA]：当将这些区域设定为禁止（CSnCR.EXENB=0（n=0 ~ 3））运行时，被检测为总线错误。

[TO]：当总线存取在 768 个周期以内不结束时，被检测为总线错误。

注． 装载的内部 RAM、数据闪存和内部 ROM 的容量因产品而不同。有关各产品的规格，请参照“38. RAM”、“39. ROM（保存代码的闪存）”和“40. E2 数据闪存（保存数据的闪存）”。

16. DMA 控制器 (DMACA)

RX210 群内置 4 个通道的 DMAC (Direct Memory Access Controller)。

DMAC 不通过 CPU 进行数据传送。如果发生传送请求, DMAC 就将传送源地址的数据传送到传送目标地址。

16.1 概要

DMAC 的规格和框图分别如表 16.1 和图 16.1 所示。

表 16.1 DMAC 的规格

项目		内容
通道数		4 个通道 (DMAC _m (m=0 ~ 3))
传送空间		512M 字节 (00000000h ~ 0FFFFFFFh 和 F0000000h ~ FFFFFFFFh 中除保留区以外的区域)
最大传送数据量		1M 数据 (块传送模式的最大总传送量: 1024 个数据 × 1024 块)
DMA 启动源		<ul style="list-style-type: none"> 各通道可选择启动源。 软件触发 外围模块中断请求 / 外部中断输入引脚的触发输入 (注 1)
通道优先级		通道 0 > 通道 1 > 通道 2 > 通道 3 (通道 0 最优先)
传送数据	1 个数据	位长: 8 位、16 位、32 位
	块大小	数据量: 1 ~ 1024 个数据
传送模式	正常传送模式	<ul style="list-style-type: none"> 1 次 DMA 传送请求进行 1 个数据的传送。 能指定总数据传送量 (自由运行模式)。
	重复传送模式	<ul style="list-style-type: none"> 1 次 DMA 传送请求进行 1 个数据的传送。 如果传送由传送源或者传送目标设定的重复大小的数据, 就返回到开始传送时的地址。 重复大小最多能设定 1024 次。
	块传送模式	<ul style="list-style-type: none"> 1 次 DMA 传送请求进行 1 块数据的传送。 块大小最多能设定 1024 个数据。
选择功能	扩展重复区域功能	<ul style="list-style-type: none"> 能通过固定传送地址寄存器的高位的值, 重复设定特定范围的地址。 能分别给传送源和传送目标设定 2 字节到 128M 字节的扩展重复区域。
中断请求	传送结束中断	当传送计数器设定的数据量传送结束时, 发生此中断。
	传送退出结束中断	当重复大小的数据传送结束或者扩展重复区域发生上溢时, 发生此中断。
低功耗功能		能设定为模块停止状态。
事件链接功能		在进行 1 次数据传送后 (块传送时为 1 块传送后), 产生事件链接请求。

注 1. DMAC 的启动源请参照“14. 中断控制器 (ICUb)”的“表 14.3 中断向量表”。

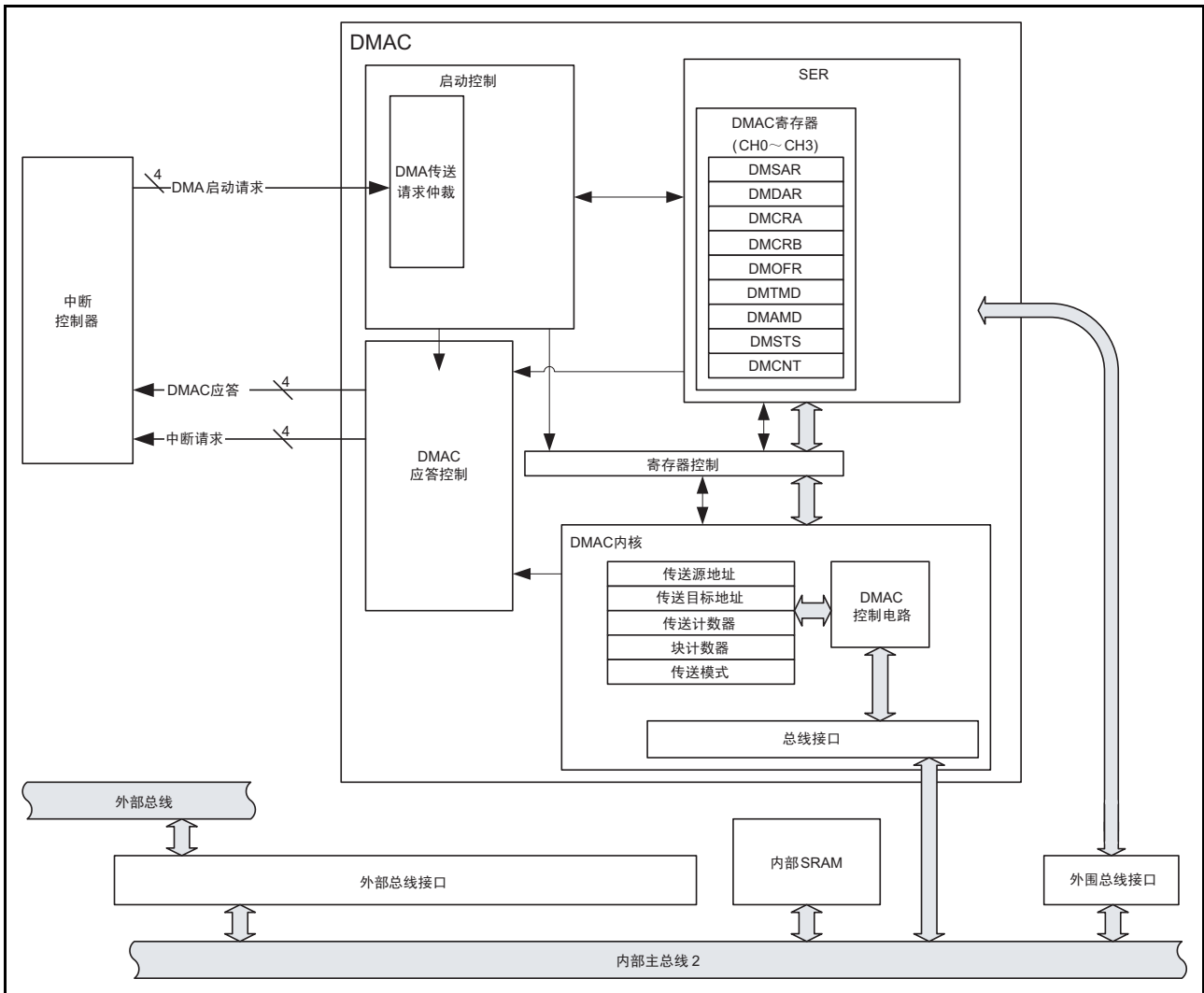
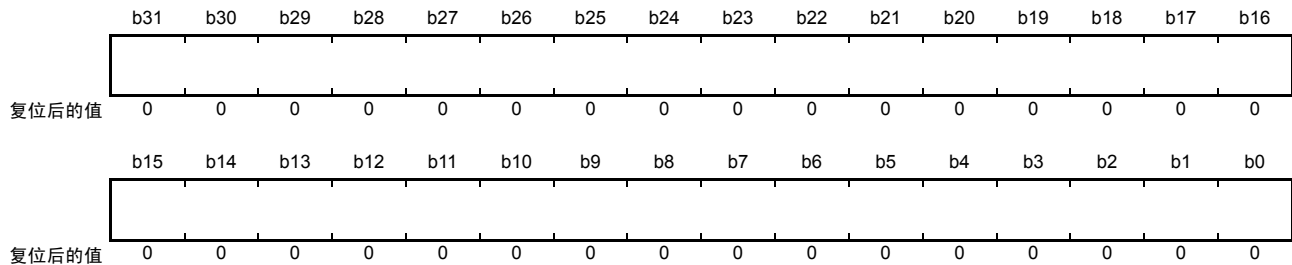


图 16.1 DMAC 的框图

16.2 寄存器说明

16.2.1 DMA 传送源地址寄存器 (DMSAR)

地址 DMAC0.DMSAR 0008 2000h、DMAC1.DMSAR 0008 2040h、DMAC2.DMSAR 0008 2080h、DMAC3.DMSAR 0008 20C0h



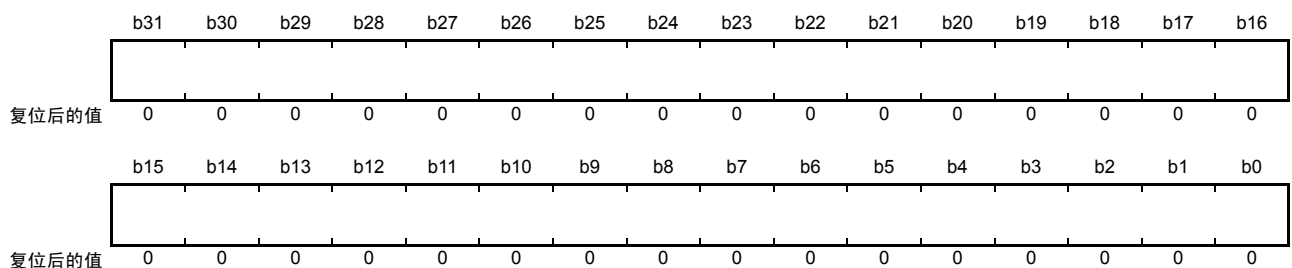
位	功能	设定范围	R/W
b31-b0	设定传送源的起始地址。	00000000h ~ 0FFFFFFFh (256M 字节) F0000000h ~ FFFFFFFFh (256M 字节)	R/W

在设定 DMSAR 寄存器时, 必须在禁止启动 DMAC (DMAST.DMST 位 =0) 或者禁止 DMA 传送 (DMCNT.DTE 位 =0) 时进行写操作。

bit31 ~ bit29 的设定值无效, 将 bit28 的值扩展到 bit31 ~ bit29。如果读 DMSAR 寄存器, 就会读到位扩展后的值。

16.2.2 DMA 传送目标地址寄存器 (DMDAR)

地址 DMAC0.DMDAR 0008 2004h、DMAC1.DMDAR 0008 2044h、DMAC2.DMDAR 0008 2084h、DMAC3.DMDAR 0008 20C4h



位	功能	设定范围	R/W
b31-b0	设定传送目标的起始地址。	00000000h ~ 0FFFFFFFh (256M 字节) F0000000h ~ FFFFFFFFh (256M 字节)	R/W

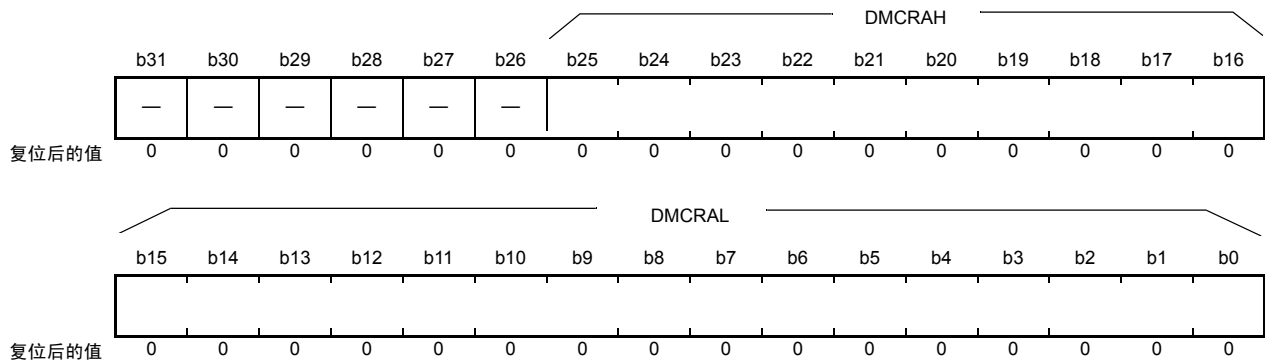
在设定 DMDAR 寄存器时, 必须在禁止启动 DMAC (DMAST.DMST 位 =0) 或者禁止 DMA 传送 (DMCNT.DTE 位 =0) 时进行写操作。

bit31 ~ bit29 的设定值无效, 将 bit28 的值扩展到 bit31 ~ bit29。如果读 DMDAR 寄存器, 就会读到位扩展后的值。

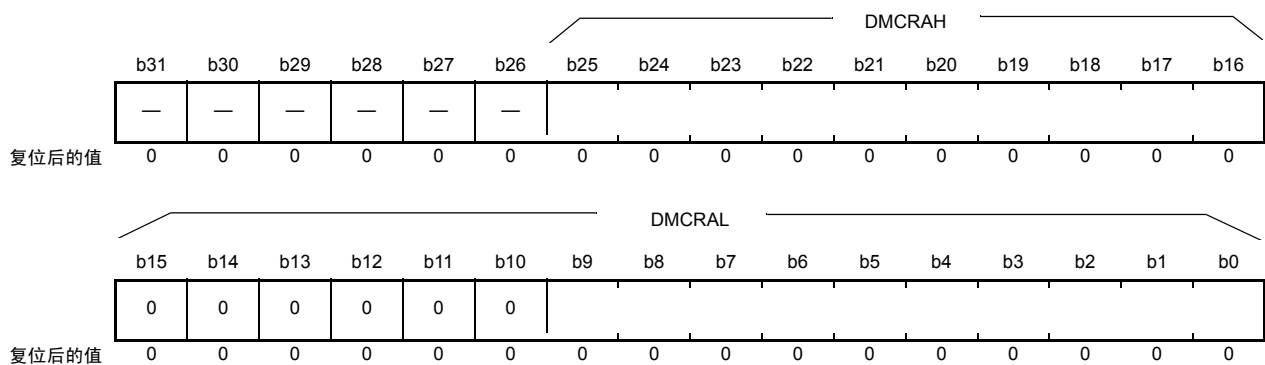
16.2.3 DMA 传送计数寄存器 (DMCRA)

地址 DMAC0.DMCRA 0008 2008h、DMAC1.DMCRA 0008 2048h、DMAC2.DMCRA 0008 2088h、DMAC3.DMCRA 0008 20C8h

• 正常传送模式



• 重复传送模式、块传送模式



符号	位名	功能	R/W
DMCRAL	传送计数的低位	设定传送次数。	R/W
DMCRAH	传送计数的高位		R/W

注. 在重复传送模式和块传送模式中, 必须给 DMCRAH 寄存器和 DMCRAL 寄存器设定相同的值。

(1) 正常传送模式 (DMACm.DMTMD.MD[1:0] 位 =00b)

DMCRAL 寄存器用作 16 位传送计数器。

当设定值是“0001h”时, 传送次数为 1 次; 当设定值是“FFFFh”时, 传送次数为 65535 次。每进行 1 次数据传送, DMCRAL 寄存器就减 1。

当设定值是“0000h”时, 不指定传送次数, 在传送计数器停止计数后进行数据传送 (自由运行模式)。

在正常传送模式中不使用 DMCRAH 寄存器, 必须给 DMCRAH 寄存器写“0000h”。

(2) 重复传送模式 (DMACm.DMTMD.MD[1:0] 位 =01b)

DMCRAH 寄存器保持重复大小, DMCRAL 寄存器用作 10 位传送计数器。

当设定值是“001h”时, 传送次数为 1 次; 当设定值是“3FFh”时, 传送次数为 1023 次; 当设定是“000h”时, 传送次数为 1024 次。在重复传送模式中, DMCRAH 寄存器和 DMCRAL 寄存器的设定范围都是“000h ~ 3FFh” (1 次~ 1024 次)。

DMCRAL 寄存器的 bit15 ~ bit10 的设定值无效, 必须给 DMCRAL 寄存器的 bit15 ~ bit10 写“0”。

每进行 1 次数据传送, DMCRAL 寄存器就减 1。当 DMCRAL 寄存器变为“000h”时, 传送 DMCRAH 寄存器的值。

(3) 块传送模式 (DMACm.DMTMD.MD[1:0] 位 =10b)

DMCRAH 寄存器保持块大小，DMCRAL 寄存器用作 10 位块大小计数器。

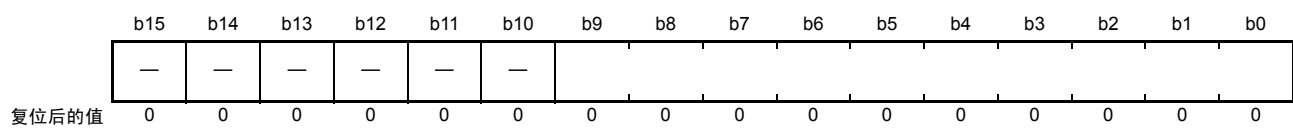
当设定值是“001h”时，块大小为 1；当设定值是“3FFh”时，块大小为 1023；当设定是“000h”时，块大小为 1024 次。在块传送模式中，DMCRAH 寄存器和 DMCRAL 寄存器的设定范围都是“000h ~ 3FFh”。

DMCRAL 寄存器的 bit15 ~ bit10 的设定值无效，必须给 DMCRAL 寄存器的 bit15 ~ bit10 写“0”。

每进行 1 次数据传送，DMCRAL 寄存器就减 1。当 DMCRAL 寄存器变为“000h”时，传送 DMCRAH 寄存器的值。

16.2.4 DMA 块传送计数寄存器 (DMCRB)

地址 DMAC0.DMCRB 0008 200Ch、DMAC1.DMCRB 0008 204Ch、DMAC2.DMCRB 0008 208Ch、DMAC3.DMCRB 0008 20CCh



位	功能	设定范围	R/W
b9-b0	设定块传送次数和重复传送次数。	001h ~ 3FFh (1 ~ 1023 次) 000h (1024 次)	R/W
b15-b10	保留位	读写值都为“0”。	R/W

DMCRB 寄存器指定块传送模式的块传送次数以及指定重复传送模式的重复传送次数。

当设定值是“001h”时，传送次数为 1 次；当设定值是“3FFh”时，传送次数为 1023 次；当设定值是“000h”时，传送次数为 1024 次。

在重复传送模式中传送 1 个重复大小的最后数据时，DMCRB 寄存器减 1。

在块传送模式中传送 1 块大小的最后数据时，DMCRB 寄存器减 1。

在设定为正常传送模式时，不使用 DMCRB 寄存器，设定值无效。

16.2.5 DMA 传送模式寄存器 (DMTMD)

地址 DMAC0.DMTMD 0008 2010h、DMAC1.DMTMD 0008 2050h、DMAC2.DMTMD 0008 2090h、DMAC3.DMTMD 0008 20D0h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
MD[1:0]		DTS[1:0]		—	—	SZ[1:0]		—	—	—	—	—	—	DCTG[1:0]	
复位后的值															
0		0		0		0		0		0		0		0	

位	符号	位名	功能	R/W
b1-b0	DCTG[1:0]	传送请求选择位	b1 b0 0 0: 软件 0 1: 外围模块中断和外部中断输入引脚的中断 (注 1) 1 0: 不能设定 1 1: 不能设定	R/W
b7-b2	—	保留位	读写值都为“0”。	R/W
b9-b8	SZ[1:0]	数据传送长度位	b9 b8 0 0: 8 位传送 0 1: 16 位传送 1 0: 32 位传送 1 1: 不能设定	R/W
b11-b10	—	保留位	读写值都为“0”。	R/W
b13-b12	DTS[1:0]	重复区域选择位	b13 b12 0 0: 传送目标为重复区域或者块区域 0 1: 传送源为重复区域或者块区域 1 0: 不设定重复区域和块区域 1 1: 不能设定	R/W
b15-b14	MD[1:0]	传送模式设定位	b15 b14 0 0: 正常传送 0 1: 重复传送 1 0: 块传送 1 1: 不能设定	R/W

注 1. 通过 ICU.DMRSRm 寄存器设定 DMAC 的启动源, 详细内容请参照“14. 中断控制器 (ICU)”的“表 14.3 中断向量表”。

DTS[1:0] 位 (重复区域选择位)

在重复传送模式或者块传送模式中, 这些位能选择传送源或者传送目标为重复区域; 在正常传送模式中, 这些位的设定值无效。

16.2.6 DMA 中断设定寄存器 (DMINT)

地址 DMAC0.DMINT 0008 2013h、DMAC1.DMINT 0008 2053h、DMAC2.DMINT 0008 2093h、DMAC3.DMINT 0008 20D3h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	DTIE	ESIE	RPTIE	SARIE	DARIE
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	DARIE	传送目标地址的扩展重复区域上溢中断允许位	0: 禁止传送目标地址的扩展重复区域上溢中断 1: 允许传送目标地址的扩展重复区域上溢中断	R/W
b1	SARIE	传送源地址的扩展重复区域上溢中断允许位	0: 禁止传送源地址的扩展重复区域上溢中断 1: 允许传送源地址的扩展重复区域上溢中断	R/W
b2	RPTIE	重复大小结束中断允许位	0: 禁止重复大小结束中断 1: 允许重复大小结束中断	R/W
b3	ESIE	传送退出结束中断允许位	0: 禁止退出中断 1: 允许退出中断	R/W
b4	DTIE	传送结束中断允许位	0: 禁止传送结束中断 1: 允许传送结束中断	R/W
b7-b5	—	保留位	读写值都为“0”。	R/W

DARIE 位 (传送目标地址的扩展重复区域上溢中断允许位)

如果在将 DARIE 位置“1”时传送目标地址的扩展重复区域发生上溢，就将 DMCNT.DTE 位清“0”，同时将 DMSTS.ESIF 标志置“1”，表示发生了传送目标地址的扩展重复区域上溢中断请求。

当和块传送模式并用时，在 1 块数据传送结束后产生中断请求。如果通过中断将传送结束通道的 DMACm.DMCNT.DTE 位置“1”，就能从传送结束状态重新开始传送。

如果没有将扩展重复区域设定为传送目标地址，DARIE 位的设定值就无效。

SARIE 位 (传送源地址的扩展重复区域上溢中断允许位)

如果在将 SARIE 位置“1”时传送源地址的扩展重复区域发生上溢，就将 DMCNT.DTE 位清“0”，同时将 DMSTS.ESIF 标志置“1”，表示发生了传送源地址的扩展重复区域上溢中断请求。

当和块传送模式并用时，在 1 块数据传送结束后产生中断请求。如果通过中断将传送结束通道的 DMACm.DMCNT.DTE 位置“1”，就能从传送结束状态重新开始传送。

如果没有将扩展重复区域设定为传送源地址，SARIE 位的设定值就无效。

RPTIE 位 (重复大小结束中断允许位)

在重复传送模式中，如果将 RPTIE 位置“1”，就在 1 个重复大小的传送结束后将 DMCNT.DTE 位清“0”，同时将 DMSTS.ESIF 标志置“1”，表示发生了重复大小结束中断请求。即使 DMTMD.DTS[1:0] 位为“10b” (不指定重复区域或者块区域)，也能产生重复大小结束中断请求。

在块传送模式中，如果将 DMINT.RPTIE 位置“1”，同样也在 1 块数据传送结束后将 DMCNT.DTE 位清“0”，同时将 DMSTS.ESIF 标志变为“1”，表示发生了重复大小结束中断请求。即使 DMTMD.DTS[1:0] 位为“10b” (不指定重复区域或者块区域)，也能产生重复大小结束中断请求。

ESIE 位 (传送退出结束中断允许位)

此位允许或者禁止 DMA 传送过程中发生的退出中断请求 (重复大小结束中断和扩展重复区域上溢中断)。

如果将 ESIE 位置“1”，就在 DMSTS.ESIF 标志被置“1”时产生传送退出结束中断。如果将 ESIE 位或者 DMSTS.ESIF 标志清“0”，就解除传送退出结束中断。

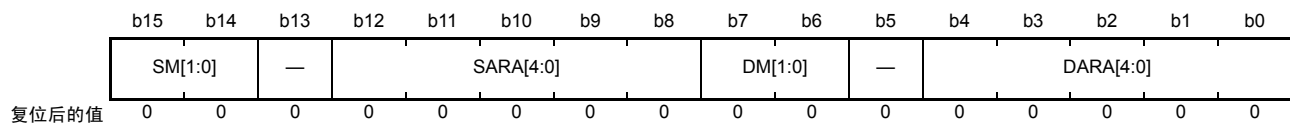
DTIE 位 (传送结束中断允许位)

此位允许或者禁止在指定次数的数据传送结束时发生的传送结束中断请求。

如果将 DTIE 位置“1”，就在 DMSTS.DTIF 标志被置“1”时产生传送结束中断。如果将 DTIE 位或者 DMSTS.DTIF 标志清“0”，就解除传送结束中断。

16.2.7 DMA 地址模式寄存器 (DMAMD)

地址 DMAC0.DMAMD 0008 2014h、DMAC1.DMAMD 0008 2054h、DMAC2.DMAMD 0008 2094h、DMAC3.DMAMD 0008 20D4h



位	符号	位名	功能	R/W
b4-b0	DARA[4:0]	传送目标地址的扩展重复区域设定位	能将扩展重复区域设定为传送目标地址，设定值的详细内容请参照表 16.2。	R/W
b5	—	保留位	读写值都为“0”。	R/W
b7-b6	DM[1:0]	传送目标地址的更新模式设定位	b7 b6 0 0: 地址固定 0 1: 偏移量增加 (注 1) 1 0: 递增 1 1: 递减	R/W
b12-b8	SARA[4:0]	传送源地址的扩展重复区域设定位	能将扩展重复区域设定为传送源地址，设定值的详细内容请参照表 16.2。	R/W
b13	—	保留位	读写值都为“0”。	R/W
b15-b14	SM[1:0]	传送源地址的更新模式设定位	b15 b14 0 0: 地址固定 0 1: 偏移量增加 (注 1) 1 0: 递增 1 1: 递减	R/W

注 1. 只有 DMAC0 能设定偏移量增加。

DARA[4:0] 位 (传送目标地址的扩展重复区域设定位)

这些位能将扩展重复区域设定为传送目标地址。通过将指定的低位地址作为地址更新的对象并且总是给剩余的高位取固定值, 实现扩展重复区域功能。能设定 2 字节~ 128M 字节的扩展重复区域, 设定间隔以 2 的乘方字节为单位。

由地址的增减引起的从扩展重复区域上溢的低位地址会发生以下变化: 当地址增加时, 为扩展重复区域的起始地址; 当地址减小时, 为扩展重复区域的结束地址。

如果将重复区域或者块区域设定为传送目标, 就不能设定传送目标地址的扩展重复区域。如果在进行重复传送或者块传送时将 DMACm.DMTMD.DTS[1:0] 位置 “00b” (传送目标为重复区域或者块区域), 就必须给 DARA[4:0] 位写 “00000b”。

如果 DMINT.DARIE 位为 “1”, 就能在扩展重复区域发生上溢时产生中断。扩展重复区域的设定和范围如表 16.2 所示。

DM[1:0] 位 (传送目标地址的更新模式设定位)

这些位设定传送目标地址的更新模式。

如果选择递增, 就在 DMTMD.SZ[1:0] 位为 “00b” 时加 1; 在 DMTMD.SZ[1:0] 位为 “01b” 时加 2; 在 DMTMD.SZ[1:0] 位为 “10b” 时加 4。

如果选择递减, 就在 DMTMD.SZ[1:0] 位为 “00b” 时减 1; 在 DMTMD.SZ[1:0] 位为 “01b” 时减 2; 在 DMTMD.SZ[1:0] 位为 “10b” 时减 4。

如果选择偏移量增加, 就加上 DMAC0.DMOFR 寄存器的设定值。只有 DMAC0 能设定偏移量增加。

SARA[4:0] 位 (传送源地址的扩展重复区域设定位)

这些位能将扩展重复区域设定为传送源地址。通过将指定的低位地址作为地址更新的对象并且总是给剩余的高位取固定值, 实现扩展重复区域功能。能设定 2 字节~ 128M 字节的扩展重复区域, 设定间隔以 2 的乘方字节为单位。

由地址的增减引起的从扩展重复区域上溢的低位地址会发生以下变化: 当地址增加时, 为扩展重复区域的起始地址; 当地址减小时, 为扩展重复区域的结束地址。

如果将重复区域或者块区域设定为传送源, 就不能设定传送源地址的扩展重复区域。如果在进行重复传送或者块传送时将 DMACm.DMTMD.DTS[1:0] 位置 “01b” (传送源为重复区域或者块区域), 就必须给 SARA[4:0] 位写 “00000b”。

如果 DMINT.SARIE 位为 “1”, 就能在扩展重复区域发生上溢时产生中断。扩展重复区域的设定和范围如表 16.2 所示。

SM 位 (传送源地址的更新模式设定位)

此位设定传送源地址的更新模式。

如果选择递增, 就在 DMTMD.SZ[1:0] 位为 “00b” 时加 1; 在 DMTMD.SZ[1:0] 位为 “01b” 时加 2; 在 DMTMD.SZ[1:0] 位为 “10b” 时加 4。

如果选择递减, 就在 DMTMD.SZ[1:0] 位为 “00b” 时减 1; 在 DMTMD.SZ[1:0] 位为 “01b” 时减 2; 在 DMTMD.SZ[1:0] 位为 “10b” 时减 4。

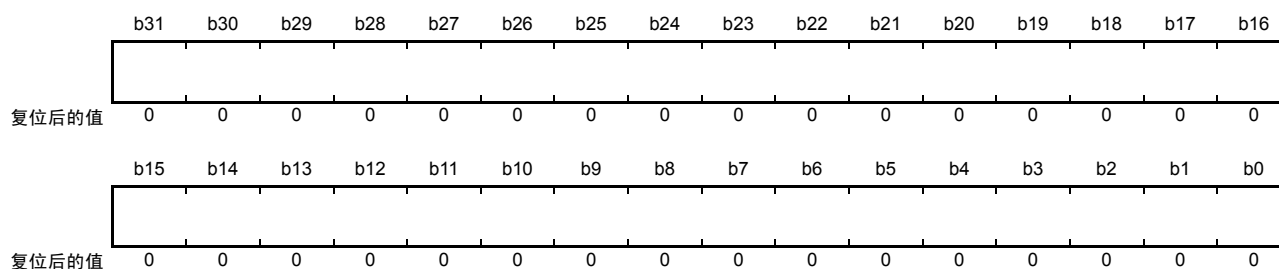
如果选择偏移量增加, 就将 DMAC0.DMOFR 寄存器的设定值相加。只有 DMAC0 能设定偏移量增加。

表 16.2 扩展重复区域的设定和范围

SARA[4:0]/DARA[4:0] 的值	扩展重复区域的范围
00000b	不设定扩展重复区域。
00001b	将该地址的低 1 位 (2 字节) 设定为扩展重复区域。
00010b	将该地址的低 2 位 (4 字节) 设定为扩展重复区域。
00011b	将该地址的低 3 位 (8 字节) 设定为扩展重复区域。
00100b	将该地址的低 4 位 (16 字节) 设定为扩展重复区域。
00101b	将该地址的低 5 位 (32 字节) 设定为扩展重复区域。
00110b	将该地址的低 6 位 (64 字节) 设定为扩展重复区域。
00111b	将该地址的低 7 位 (128 字节) 设定为扩展重复区域。
01000b	将该地址的低 8 位 (256 字节) 设定为扩展重复区域。
01001b	将该地址的低 9 位 (512 字节) 设定为扩展重复区域。
01010b	将该地址的低 10 位 (1K 字节) 设定为扩展重复区域。
01011b	将该地址的低 11 位 (2K 字节) 设定为扩展重复区域。
01100b	将该地址的低 12 位 (4K 字节) 设定为扩展重复区域。
01101b	将该地址的低 13 位 (8K 字节) 设定为扩展重复区域。
01110b	将该地址的低 14 位 (16K 字节) 设定为扩展重复区域。
01111b	将该地址的低 15 位 (32K 字节) 设定为扩展重复区域。
10000b	将该地址的低 16 位 (64K 字节) 设定为扩展重复区域。
10001b	将该地址的低 17 位 (128K 字节) 设定为扩展重复区域。
10010b	将该地址的低 18 位 (256K 字节) 设定为扩展重复区域。
10011b	将该地址的低 19 位 (512K 字节) 设定为扩展重复区域。
10100b	将该地址的低 20 位 (1M 字节) 设定为扩展重复区域。
10101b	将该地址的低 21 位 (2M 字节) 设定为扩展重复区域。
10110b	将该地址的低 22 位 (4M 字节) 设定为扩展重复区域。
10111b	将该地址的低 23 位 (8M 字节) 设定为扩展重复区域。
11000b	将该地址的低 24 位 (16M 字节) 设定为扩展重复区域。
11001b	将该地址的低 25 位 (32M 字节) 设定为扩展重复区域。
11010b	将该地址的低 26 位 (64M 字节) 设定为扩展重复区域。
11011b	将该地址的低 27 位 (128M 字节) 设定为扩展重复区域。
11100b ~ 11111b	不能设定。

16.2.8 DMA 偏移寄存器 (DMOFR)

地址 DMAC0.DMOFR 0008 2018h



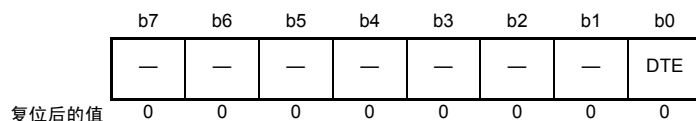
位	功能	设定范围	R/W
b31-b0	设定传送源或者传送目标的地址更新模式为偏移量增加时的偏移值。	00000000h ~ 00FFFFFFh (0 字节~ (16M-1) 字节) FF000000h ~ FFFFFFFFh (-16M 字节~-1 字节)	R/W

在设定 DMOFR 寄存器时, 必须在 DMAC 停止或者禁止 DMA 传送时而不能在数据传送过程中进行写操作。

bit31 ~ bit25 的设定值无效, 将 bit24 的值扩展到 bit31 ~ bit25。如果读 DMOFR 寄存器, 就会读到位扩展后的值。

16.2.9 DMA 传送允许寄存器 (DMCNT)

地址 DMAC0.DMCNT 0008 201Ch、DMAC1.DMCNT 0008 205Ch、DMAC2.DMCNT 0008 209Ch、DMAC3.DMCNT 0008 20DCh



位	符号	位名	功能	R/W
b0	DTE	DMA 传送允许位	0: 禁止 DMA 传送 1: 允许 DMA 传送	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

DTE 位 (DMA 传送允许位)

能在 DMAST.DMST 位为“1” (允许启动 DMAC) 并且 DTE 位为“1” (允许 DMA 传送) 时开始对应通道的 DMA 传送。

[为“1”的条件]

- 当写“1”时

[为“0”的条件]

- 当写“0”时
- 当结束所设定的总传送数据量的传送时
- 当通过重复大小结束中断停止 DMA 传送时
- 当通过扩展重复区域上溢中断停止 DMA 传送时

16.2.10 DMA 软件启动寄存器 (DMREQ)

地址 DMAC0.DMREQ 0008 201Dh、DMAC1.DMREQ 0008 205Dh、DMAC2.DMREQ 0008 209Dh、DMAC3.DMREQ 0008 20DDh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	CLRS	—	—	—	SWREQ
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	SWREQ	DMA 软件启动位	0: 无 DMA 传送请求 1: 有 DMA 传送请求	R/W
b3-b1	—	保留位	读写值都为“0”。	R/W
b4	CLRS	DMA 软件启动位的自动清除选择位	0: 在软件启动后清除 SWREQ 位 1: 在软件启动后不清除 SWREQ 位	R/W
b7-b5	—	保留位	读写值都为“0”。	R/W

SWREQ 位 (DMA 软件启动位)

如果给 SWREQ 位写“1”，就产生 DMA 的传送请求。一旦开始该请求的传送，就在 CLRS 位为“0”时将 SWREQ 位清“0”，而在 CLRS 位为“1”时不将 SWREQ 位清“0”。此时，能在传送结束后再次产生 DMA 传送请求。

但是，SWREQ 位的值只在 DMTMD.DCTG[1:0] 位被置“00b” (DMA 启动源为软件) 时有效，能通过软件启动 DMA 传送。

如果 DMTMD.DCTG[1:0] 位不是“00b”，SWREQ 位的设定值就无效。

如果在 CLRS 位为“0”时要进行软件启动，就必须在确认 SWREQ 位为“0”后给 SWREQ 位写“1”。

[为“1”的条件]

- 当写“1”时

[为“0”的条件]

- 在 CLRS 位为“0” (在软件启动后清除 SWREQ 位) 的情况下接受软件请求并且开始数据传送时
- 当写“0”时

CLRS 位 (DMA 软件启动位的自动清除选择位)

当通过给 SWREQ 位写“1”开始 DMA 传送请求的传送时，此位设定是否将 SWREQ 位清“0”。当 CLRS 位为“0”时，一旦开始传送，就将 SWREQ 位清“0”；当 CLRS 位为“1”时，不将 SWREQ 位清“0”。此时，能在传送结束后再次产生 DMA 传送请求。

16.2.11 DMA 状态寄存器 (DMSTS)

地址 DMAC0.DMSTS 0008 201Eh、DMAC1.DMSTS 0008 205Eh、DMAC2.DMSTS 0008 209Eh、DMAC3.DMSTS 0008 20DEh

	b7	b6	b5	b4	b3	b2	b1	b0
	ACT	—	—	DTIF	—	—	—	ESIF
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	ESIF	传送退出中断标志	0: 未发生传送退出中断 1: 发生传送退出中断	R/W (注1)
b3-b1	—	保留位	读取值为“0”，写操作无效。	R
b4	DTIF	传送结束中断标志	0: 无传送结束中断 1: 有传送结束中断	R/W (注1)
b6-b5	—	保留位	读取值为“0”，写操作无效。	R
b7	ACT	DMA 有效标志	0: DMAC 停止 1: DMAC 正在运行	R

注 1. 只能写“0”。

ESIF 标志 (传送退出中断标志)

这是表示发生了传送退出中断的标志。

[为“1”的条件]

- DMINT.RPTIE 位被置“1”并且在重复传送模式中 1 个重复大小的传送结束后
- DMINT.RPTIE 位被置“1”并且在块传送模式中 1 块数据传送结束后
- 在 DMINT.SARIE 位被置“1”而 DMAMD.SARA[4:0] 位被置的值不是“00000b” (将传送源地址指定为扩展重复区域) 的情况下传送源地址的扩展重复区域发生上溢时
- 在 DMINT.DARIE 位被置“1”而 DMAMD.DARA[4:0] 位被置的值不是“00000b” (将传送目标地址指定为扩展重复区域) 的情况下传送目标地址的扩展重复区域发生上溢时

[为“0”的条件]

- 当写“0”时
- 当给 DMCNT.DTE 位写“1”时

DTIF 标志 (传送结束中断标志)

这是表示发生了传送结束中断的标志。

[为“1”的条件]

- 在正常传送模式中指定次数的传送结束时 (DMCRAL 寄存器变为“0”并且传送结束时)
- 在重复传送模式中指定重复次数的传送结束时 (DMCRB 寄存器变为“0”并且传送结束时)
- 在块传送模式中指定块数的传送结束时 (DMCRB 寄存器变为“0”并且传送结束时)

[为“0”的条件]

- 当写“0”时
- 当给 DMCNT.DTE 位写“1”时

ACT 标志 (DMA 有效标志)

这是表示 DMAC 正在运行或者停止的标志。

[为“1”的条件]

- 当DMAC开始传送时

[为“0”的条件]

- 当1次传送请求的传送全部结束时

16.2.12 DMA 启动源标志控制寄存器 (DMCSL)

地址 DMAC0.DMCSL 0008 201Fh、DMAC1.DMCSL 0008 205Fh、DMAC2.DMCSL 0008 209Fh、DMAC3.DMCSL 0008 20DFh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	DISEL
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	DISEL	中断选择位	0: 在传送结束时, 将启动源的中断标志清“0”。 1: 在传送结束时, 通过启动源的中断标志向 CPU 请求中断。	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

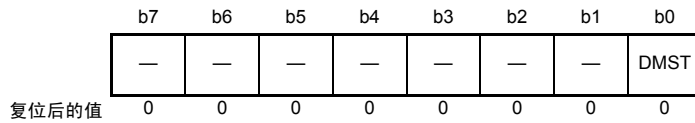
DISEL 位 (中断选择位)

在 DMAC 传送开始时, 此位选择是将启动源的中断标志清“0”还是通过中断标志向 CPU 请求中断。

在将 DMTMD.DCTG[1:0] 位置“00b” (通过软件启动) 时, DISEL 位的设定值无效。

16.2.13 DMA 模块启动寄存器 (DMAST)

地址 0008 2200h



位	符号	位名	功能	R/W
b0	DMST	DMAC 运行允许位	0: 禁止启动 DMAC 1: 允许启动 DMAC	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

DMAST 寄存器是允许或者禁止启动 DMAC 的全部通道的寄存器。

DMST 位 (DMAC 运行允许位)

当 DMST 位为“1”时，允许启动 DMAC 的全部通道。

如果在给多个通道的 DMACm.DMCNT.DTE 位写“1”（允许 DMA 转送）后将 DMST 位置“1”（允许启动 DMAC），就能将多个通道置为可同时接受传送请求的状态。

如果在 DMAC 运行过程中将 DMST 位置“0”，就在正在执行的 1 个传送请求的数据传送结束后暂停 DMA 运行。在此状态下，能通过再次将 DMST 位置“1”继续进行 DMA 传送。

[为“1”的条件]

- 当写“1”时

[为“0”的条件]

- 当写“0”时

16.3 运行说明

16.3.1 传送模式

(1) 正常传送模式

在正常传送模式中，1 次传送请求进行 1 个数据的传送。能通过设定 DMACm.DMCRAL 寄存器，设定最多 65535 次的数据传送次数。如果将 DMACm.DMCRAL 寄存器设定为“0000h”，就不需要指定传送次数，并且在传送计数器停止计数后进行数据传送（自由运行模式）。在正常传送模式中，DMACm.DMCRB 寄存器的设定无效。除自由运行模式以外，能在指定传送次数的传送结束后产生传送结束中断请求。

正常传送模式的寄存器的更新值以及正常传送模式的运行分别如表 16.3 和图 16.2 所示。

表 16.3 正常传送模式的寄存器的更新值

寄存器	功能	1 次传送请求的传送结束后的更新值
DMACm.DMSAR	传送源地址	递增 / 递减 / 固定 / 偏移量增加 (注 1)
DMACm.DMDAR	传送目标地址	递增 / 递减 / 固定 / 偏移量增加 (注 1)
DMACm.DMCRAL	传送计数	减 1 / 不更新 (自由运行模式)
DMACm.DMCRAH	—	不更新 (不用于正常传送模式)
DMACm.DMCRB	—	不更新 (不用于正常传送模式)

注 1. 只有 DMAC0 能指定偏移量增加。

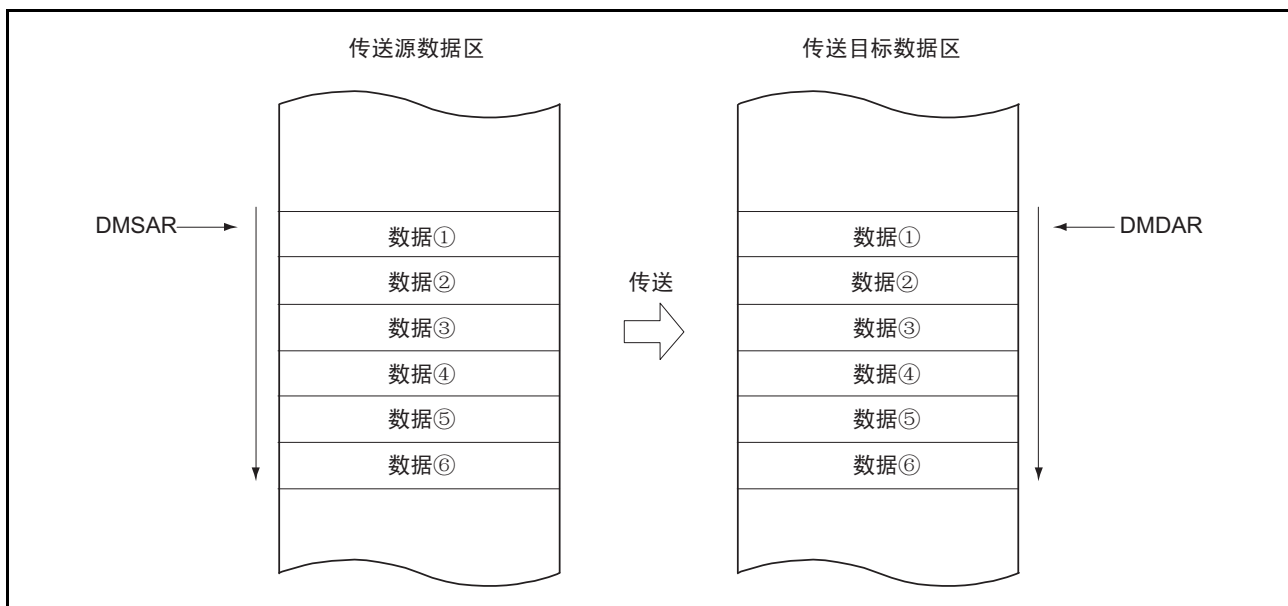


图 16.2 正常传送模式的运行

(2) 重复传送模式

在重复传送模式中，1 次传送请求进行 1 个数据的传送。

能通过 DMACm.DMCRA 寄存器设定最多 1K 数据的重复大小。

还能通过 DMACm.DMCRB 寄存器设定最多 1K 次的指定重复次数，指定最多 1K 数据 × 1K 重复次数 = 1M 数据的总数据传送量。

能将传送源或者传送目标指定为重复区域。一旦重复大小的数据传送结束，被指定为重复区域的地址寄存器 (DMACm.DMSAR 或者 DMACm.DMDAR) 就返回到初始地址。在重复传送模式中，能在重复大小的数据传送结束后停止 DMA 传送并且产生重复大小结束中断请求。如果通过重复大小结束中断处理给 DMACm.DMCNT.DTE 位写“1”，就能重新开始 DMA 传送。

能在指定重复次数的传送结束后产生传送结束中断请求。

重复传送模式的寄存器的更新值以及重复传送模式的运行分别如表 16.4 和图 16.3 所示。

表 16.4 重复传送模式的寄存器的更新值

寄存器	功能	1 次传送请求的传送结束后的更新值	
		DMACm.DMCRAL 寄存器不为“1”	DMACm.DMCRAL 寄存器为“1” (重复大小的最后数据的传送)
DMACm.DMSAR	传送源地址	递增 / 递减 / 固定 / 偏移量增加 (注 1)	<ul style="list-style-type: none"> DMACm.DMTMD.DTS[1:0]=00b 递增 / 递减 / 固定 / 偏移量增加 (注 1) DMACm.DMTMD.DTS[1:0]=01b DMACm.DMSAR 的初始值 DMACm.DMTMD.DTS[1:0]=10b 递增 / 递减 / 固定 / 偏移量增加 (注 1)
DMACm.DMDAR	传送目标地址	递增 / 递减 / 固定 / 偏移量增加 (注 1)	<ul style="list-style-type: none"> DMACm.DMTMD.DTS[1:0]=00b DMACm.DMDAR 的初始值 DMACm.DMTMD.DTS[1:0]=01b 递增 / 递减 / 固定 / 偏移量增加 (注 1) DMACm.DMTMD.DTS[1:0]=10b 递增 / 递减 / 固定 / 偏移量增加 (注 1)
DMACm.DMCRAH	重复大小	保持	保持
DMACm.DMCRAL	传送计数	减 1	DMACm.DMCRAH
DMACm.DMCRB	重复次数的计数	保持	减 1

注 1. 只有 DMAC0 能指定偏移量增加。

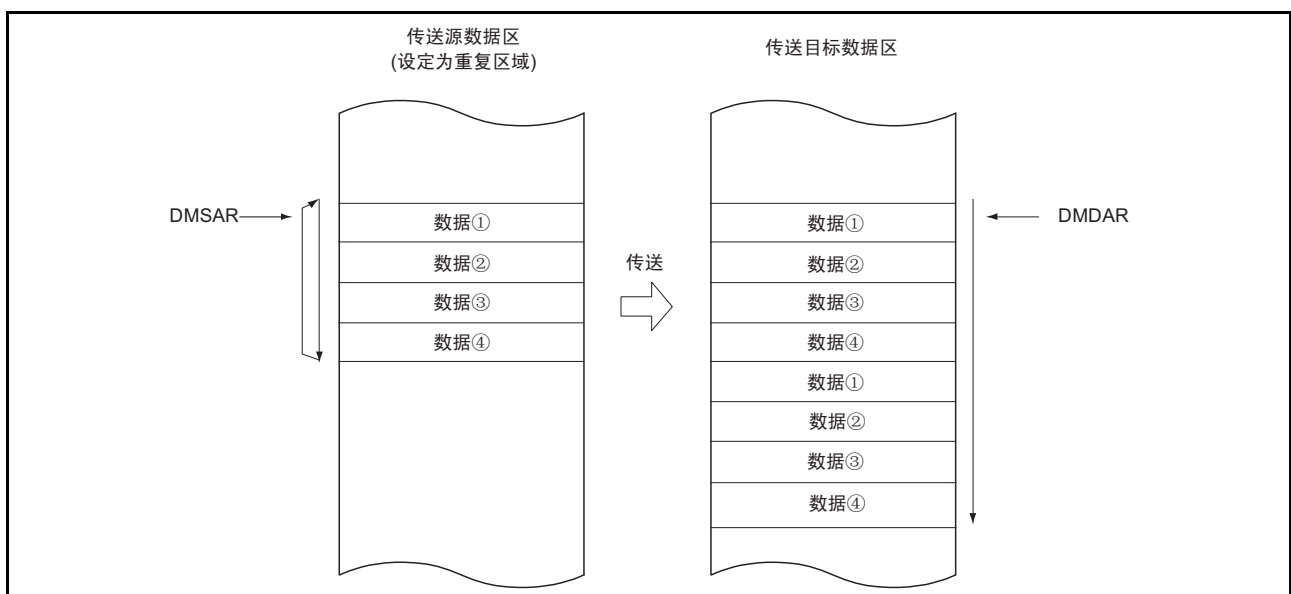


图 16.3 重复传送模式的运行

(3) 块传送模式

在块传送模式中，1 次传送请求进行 1 块数据的传送。

能通过 DMACm.DMCRA 寄存器设定最多 1K 个数据的块大小。

还能通过 DMACm.DMCRB 寄存器设定最多 1K 次的指定块数，指定最多 1K 数据 × 1K 块 = 1M 数据的总数据传送量。能将传送源或者传送目标指定为块区域。一旦 1 块数据传送结束，被指定为块区域的地址寄存器 (DMACm.DMSAR 或者 DMACm.DMDAR) 就返回到初始地址。在块传送模式中，能在 1 块数据传送结束后停止 DMA 传送并且产生重复大小结束中断请求。如果通过重复大小结束中断处理给 DMACm.DMCNT.DTE 位写“1”，就能重新开始 DMA 传送。

能在指定块数的传送结束后产生传送结束中断请求。

块传送模式的寄存器的更新值以及块传送模式的运行分别如表 16.5 和图 16.4 所示。

表 16.5 块传送模式的寄存器的更新值

寄存器	功能	1 次传送请求的 1 块传送结束后的更新值
DMACm.DMSAR	传送源地址	<ul style="list-style-type: none"> DMACm.DMTMD.DTS[1:0]=00b 递增 / 递减 / 固定 / 偏移量增加 (注 1) DMACm.DMTMD.DTS[1:0]=01b DMACm.DMSAR 的初始值 DMACm.DMTMD.DTS[1:0]=10b 递增 / 递减 / 固定 / 偏移量增加 (注 1)
DMACm.DMDAR	传送目标地址	<ul style="list-style-type: none"> DMACm.DMTMD.DTS[1:0]=00b DMACm.DMDAR 的初始值 DMACm.DMTMD.DTS[1:0]=01b 递增 / 递减 / 固定 / 偏移量增加 (注 1) DMACm.DMTMD.DTS[1:0]=10b 递增 / 递减 / 固定 / 偏移量增加 (注 1)
DMACm.DMCRAH	块大小	保持
DMACm.DMCRAL	传送计数	DMACm.DMCRAH
DMACm.DMCRB	块计数	减 1

注 1. 只有 DMAC0 能指定偏移量增加。

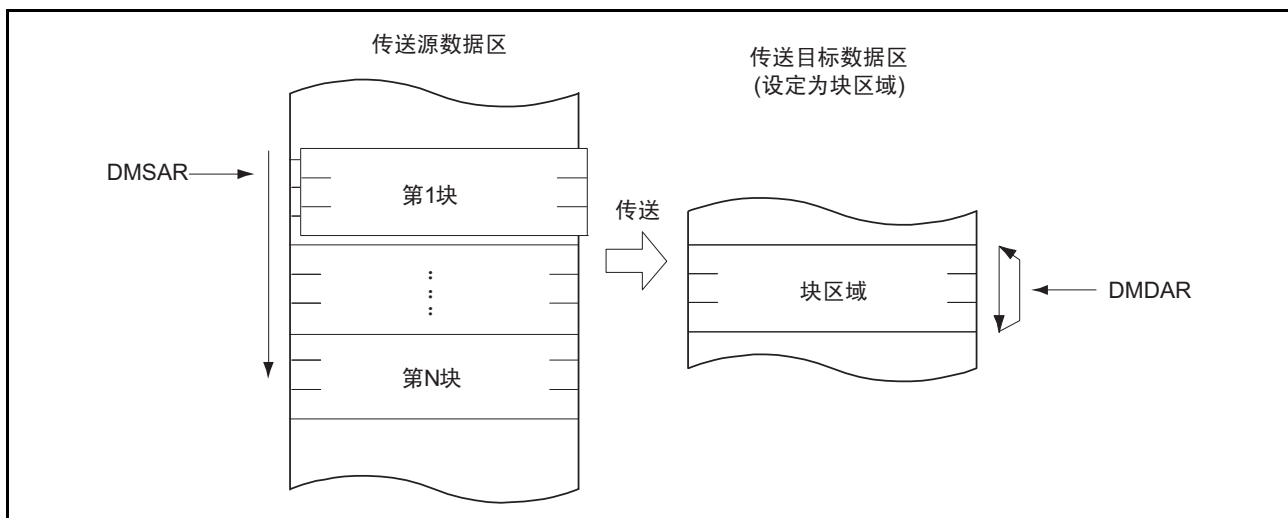


图 16.4 块传送模式的运行

16.3.2 扩展重复区域功能

DMAC 具有给传送源地址或者传送目标地址设定扩展重复区域的功能。如果设定扩展重复区域，地址寄存器就重复被指定为扩展重复区域范围的地址值。

能给 DMACm.DMSAR 寄存器（传送源地址寄存器）和 DMACm.DMDAR 寄存器（传送目标地址寄存器）独立设定扩展重复区域功能。

通过 DMACm.DMAMD.SARA[4:0] 位设定传送源地址的扩展重复区域，通过 DMACm.DMAMD.DARA[4:0] 位设定传送目标地址的扩展重复区域。能独立设定各扩展重复区域的大小，但是不能将被指定为重复区域或者块区域（传送源或者传送目标）的区域指定为扩展重复区域。

如果地址寄存器的值为扩展重复区域的结束地址并且扩展重复区域发生上溢，就暂停 DMA 传送，并且能产生扩展重复区域上溢中断请求。如果将 DMACm.DMINT.SARIE 位置“1”，就在传送源地址的扩展重复区域发生上溢时 DMACm.DMSTS.ESIF 标志变为“1”并且通过将 DMACm.DMCNT.DTE 位置“0”来结束 DMA 传送。此时，如果 DMACm.DMINT.ESIE 位为“1”，就产生扩展重复区域上溢中断请求。

如果将 DMACm.DMINT.DARIE 位置“1”，就以传送目标地址寄存器为对象。

如果通过扩展重复区域上溢中断处理将 DMACm.DMCNT.DTE 位置“1”，就能重新开始 DMA 传送。

扩展重复区域功能的例子如图 16.5 所示。

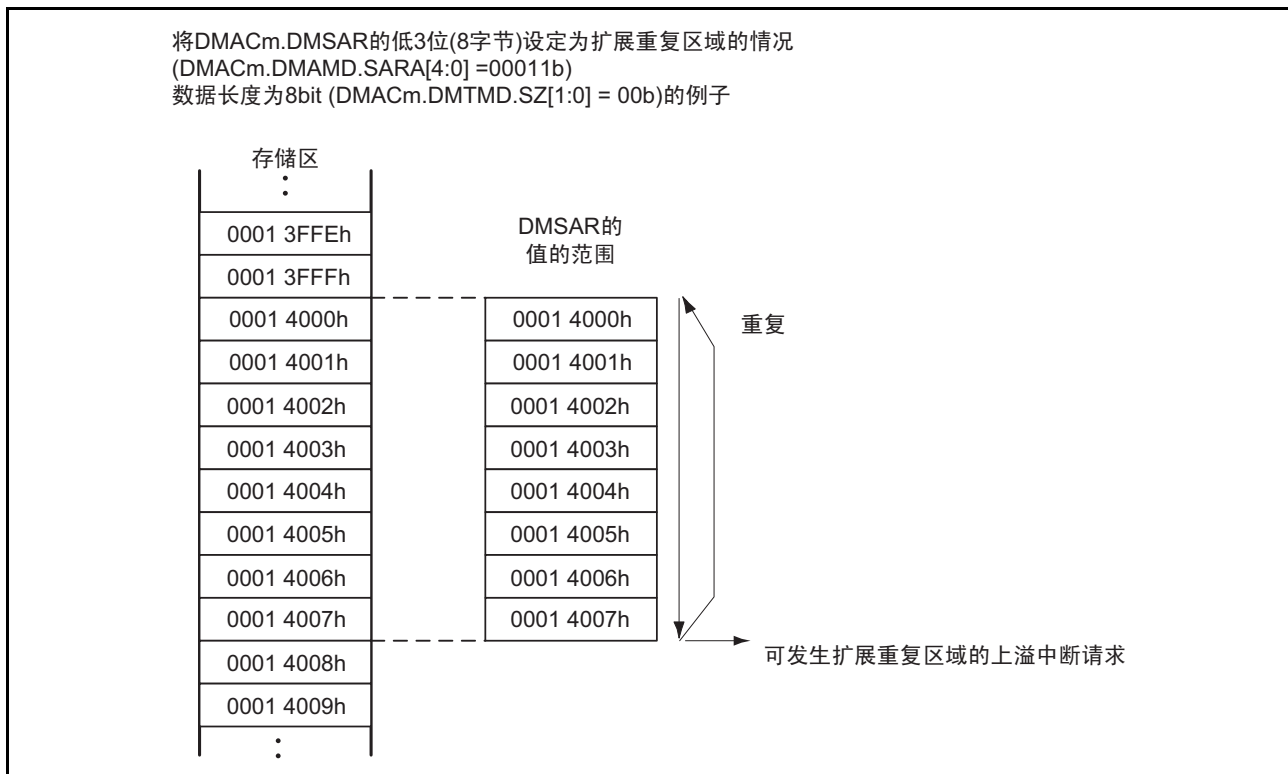


图 16.5 扩展重复区域功能的例子

在块传送模式和扩展重复区域上溢中断并用时，需要注意以下事项。

如果要在扩展重复区域发生上溢时结束传送，就需要将块大小设定为 2 的乘方，或者设定地址寄存器的值，使块大小的边界和扩展重复区域范围的边界相同。如果在传送 1 块数据的过程中扩展重复区域发生上溢，就将扩展重复区域上溢中断请求保留到 1 块数据传送结束，传送发生溢出。

块传送模式和扩展重复区域功能的并用例子如图 16.6 所示。

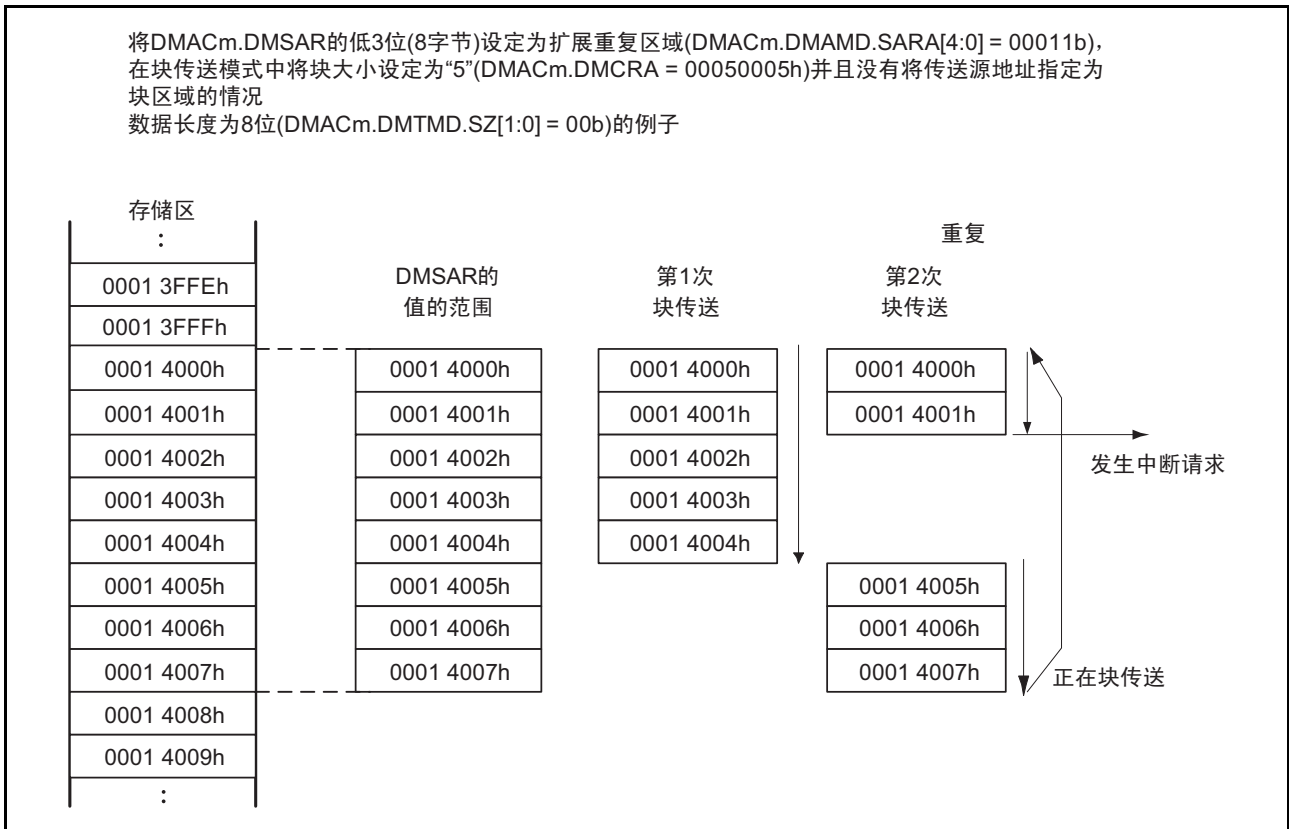


图 16.6 块传送模式和扩展重复区域功能的并用例子

16.3.3 带偏移的地址更新功能

传送源地址和传送目标地址的更新方法有固定、递增、递减和偏移量增加。偏移量增加是指在每进行 1 个数据的传送时加上 DMA 偏移寄存器 (DMAC0.DMOFR) 的设定值。通过此功能, 能跳过中途地址进行数据传送。

如果用 2 的补数给 DMAC0.DMOFR 设定负值, 也能通过偏移实现减法运算。

只有通道 DMAC0 才能使用带偏移的地址更新功能。

各地址更新模式的地址更新方法如表 16.6 所示。

表 16.6 各地址更新模式的地址更新方法

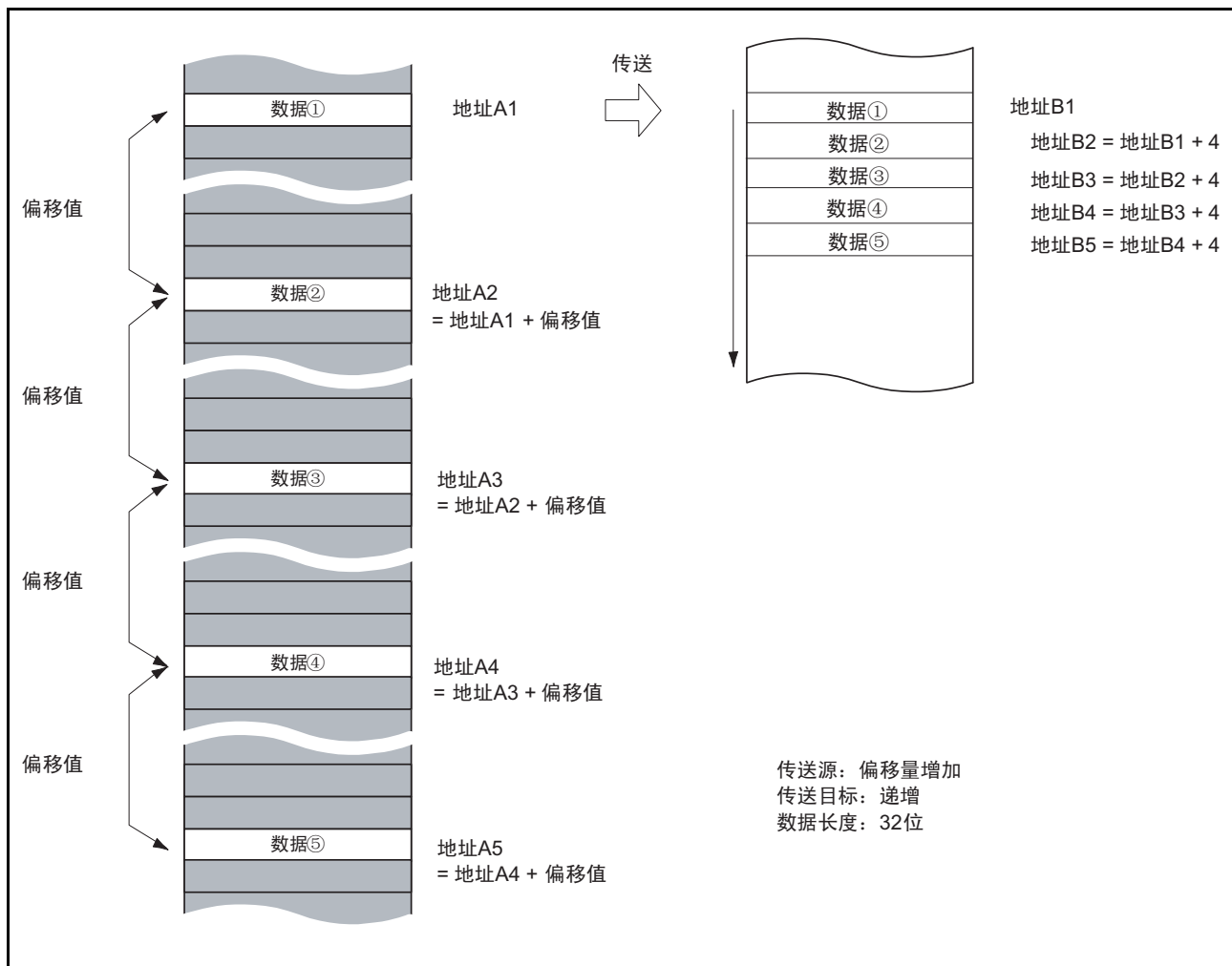
地址更新模式	DMACm.DMAMD.SM[1:0] DMACm.DMAMD.DM[1:0] 地址更新模式的设定值	更新方法 (各 DMACm.DMTMD.SZ[1:0] 设定值的更新方法)		
		SZ[1:0]=00b	SZ[1:0]=01b	SZ[1:0]=10b
地址固定	00b	固定		
偏移量增加	01b	+DMACm.DMOFR (注 1)		
递增	10b	+1	+2	+4
递减	11b	-1	-2	-4

注 1. 必须用 2 的补数给偏移寄存器设定负值。用以下表达式求 2 的补数:

负偏移值的 2 的补数表示 = \sim (偏移值) + 1 (\sim : 位取反)

(1) 使用偏移量增加的基本传送

偏移的地址更新功能的运行例子如图 16.7 所示。



在图 16.7 中，将传送数据长度设定为“32 位”，传送源地址的更新模式设定为“偏移量增加”，传送目标地址的更新模式设定为“递增”。在进行第 2 次以后的传送源地址更新时，从上次传送时的地址加上偏移值后的地址读数据，然后将一定间隔的读数据写到传送目标的连续区域。

(2) 使用偏移量增加的 XY 转换例子

通过重复传送模式和偏移量增加的组合进行 XY 转换时的运行例子如图 16.8 所示。
 设定方法如下：

- DMAC0.DMAMD 寄存器：传送源地址更新模式设定：偏移量增加
- DMAC0.DMAMD 寄存器：传送目标地址更新模式设定：递增
- DMAC0.DMTMD 寄存器：传送数据大小位：32 位传送
- DMAC0.DMTMD 寄存器：传送模式设定位：重复传送
- DMAC0.DMTMD 寄存器：重复区域选择位：传送源侧为重复区域
- DMAC0.DMOFR 寄存器：偏移地址：10h
- DMAC0.DMCRA 寄存器：重复大小：4h
- DMAC0.DMINT 寄存器：设定为重复大小结束中断

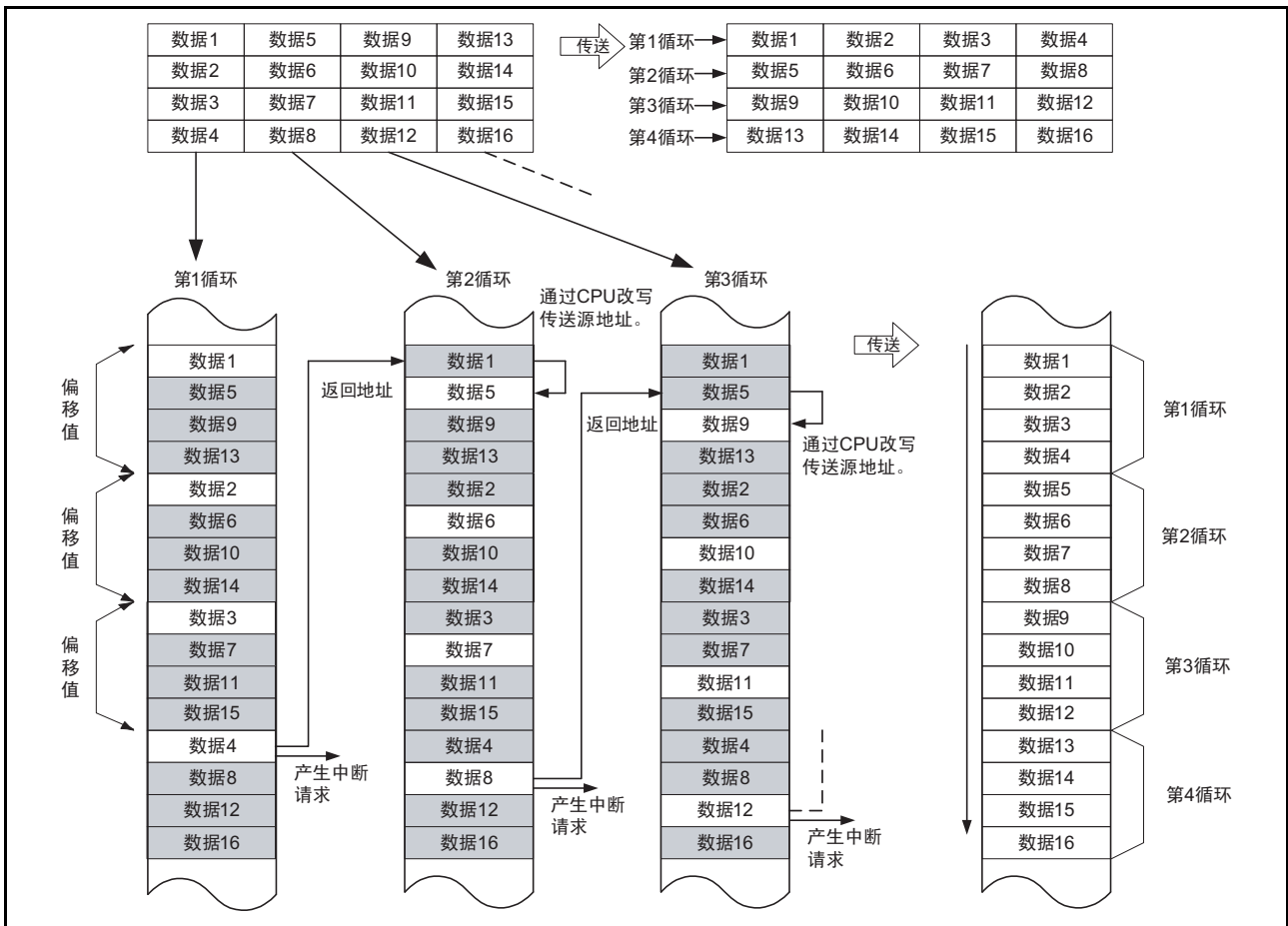


图 16.8 使用重复传送模式和偏移量增加进行 XY 转换时的运行

一旦传送开始，传送源就将地址加上偏移值进行数据传送，传送数据按照传送顺序连续排列在传送目标中。如果传送“数据 4”为止的数据，就结束重复大小数据的传送，DMAC 将传送源地址返回到开始传送时的地址（传送源“数据 1”的地址），同时产生重复大小结束中断请求。通过此中断请求一旦中止传送。必须通过中断进行以下处理：

- DMAC0.DMSAR 寄存器：将 DMA 传送源地址改写为“数据 5”的地址。
 （在上述例子中，改写为“数据 1”的地址+4 的值）。
- DMAC0.DMCNT 寄存器：给 DTE 位写“1”。

从中止 DMA 传送的状态继续开始 DMA 传送。此后，如果重复同样的处理，就能在将传送源数据进行 XY 转换后传送到传送目标。

XY 转换的处理流程如图 16.9 所示。

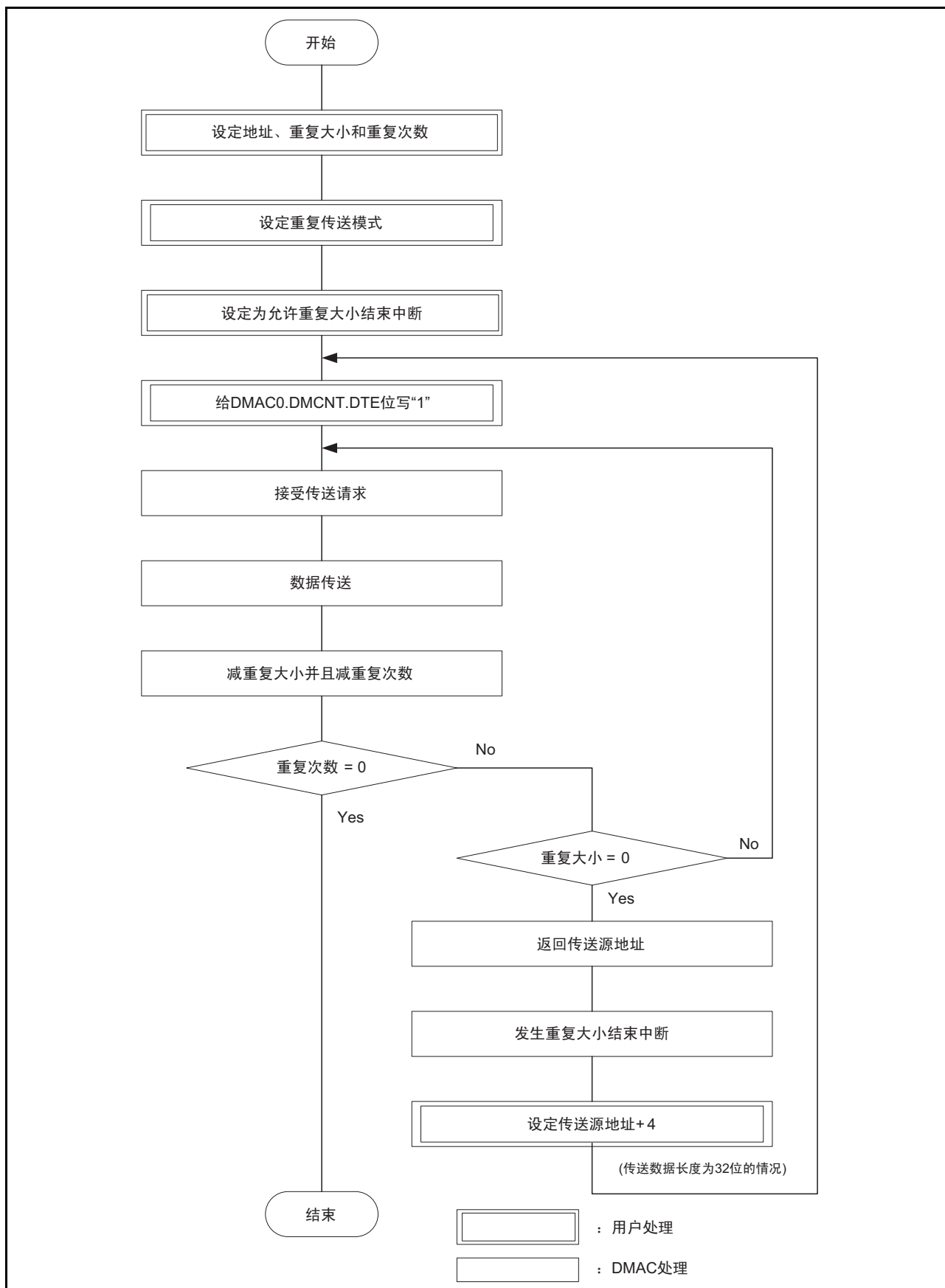


图 16.9 使用重复传送模式和偏移量增加进行 XY 转换的流程

16.3.4 启动源

DMAC 的启动源有软件、外围模块中断请求和外部中断请求，能通过 DMACm.DMTMD.DCTG[1:0] 位选择这些启动源。

(1) 通过软件进行的启动

如果将 DMACm.DMTMD.DCTG[1:0] 位置“00b”，就能通过软件进行启动。

要通过软件启动 DMA 传送时，必须在将 DMACm.DMTMD.DCTG[1:0] 位置“00b”后给 DMACm.DMCNT.DTE 位写“1”（允许 DMA 传送），并且将 DMAST.DMST 位置“1”（允许启动 DMAC）。此后，如果给 DMACm.DMREQ.SWREQ 位写“1”（有 DMA 传送请求），就开始 DMA 运行。

如果在 DMACm.DMREQ.CLR 位为“0”的状态下进行软件启动，就在开始 DMA 传送请求的传送时 DMACm.DMREQ.SWREQ 位变为“0”；如果在 DMACm.DMREQ.CLR 位为“1”的状态下进行软件启动，即使开始传送，DMACm.DMREQ.SWREQ 位也不变为“0”。在该请求的传送结束后再次发生 DMA 传送请求。

(2) 通过外围模块中断请求或者外部中断请求进行的启动

能将外围模块中断请求或者外部中断请求指定为 DMA 传送启动源。通过中断控制器 (ICU) 的 ICU.DMRSRm 寄存器 (m=0 ~ 3) 选择启动源，各通道能独立设定。

当通过外围模块中断请求或者外部中断请求启动 DMA 时，必须在将 DMACm.DMTMD.DCTG[1:0] 位置“01b”（外围模块中断和外部中断引脚的中断）后将 DMACm.DMCNT.DTE 位置“1”（允许 DMA 传送），并且将 DMAST.DMST 位置“1”（允许启动 DMAC）。此后，如果发生中断请求，就开始 DMA 运行。

DMAC 启动源的中断源一览表，请参照“14. 中断控制器 (ICUb)”的“表 14.3 中断向量表”。

16.3.5 运行时序

DMAC 的运行时序例子如图 16.10 和图 16.11 所示。

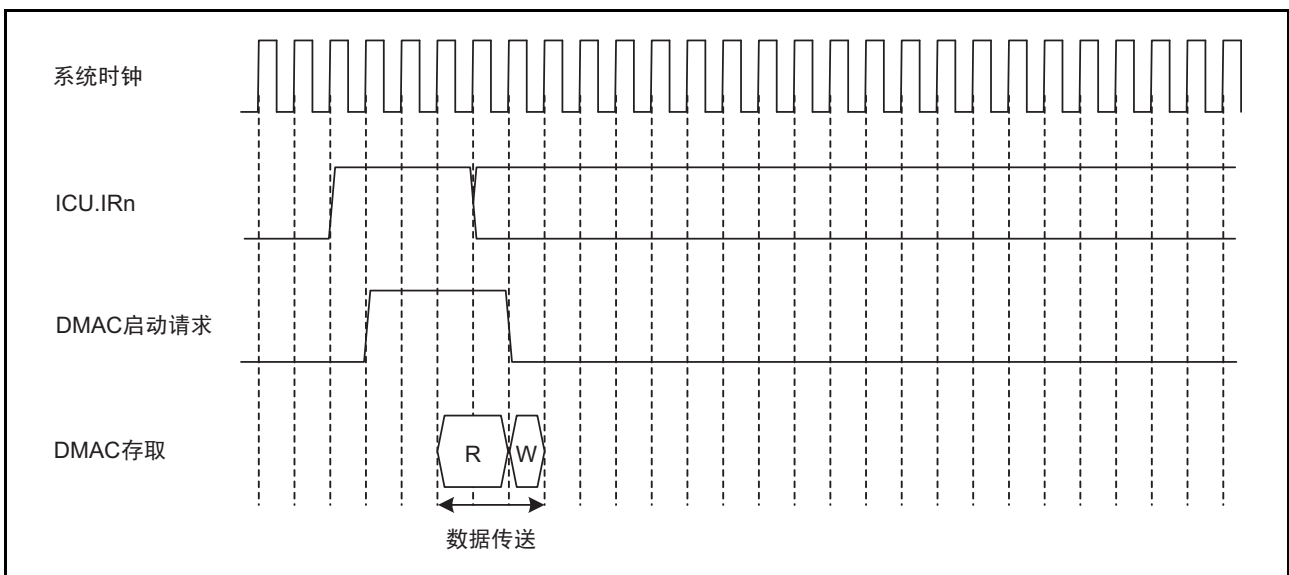


图 16.10 DMAC 的运行时序例子 (1) (在正常传送模式和重复传送模式中，通过外围模块或者外部中断输入引脚的中断启动 DMA 的情况)

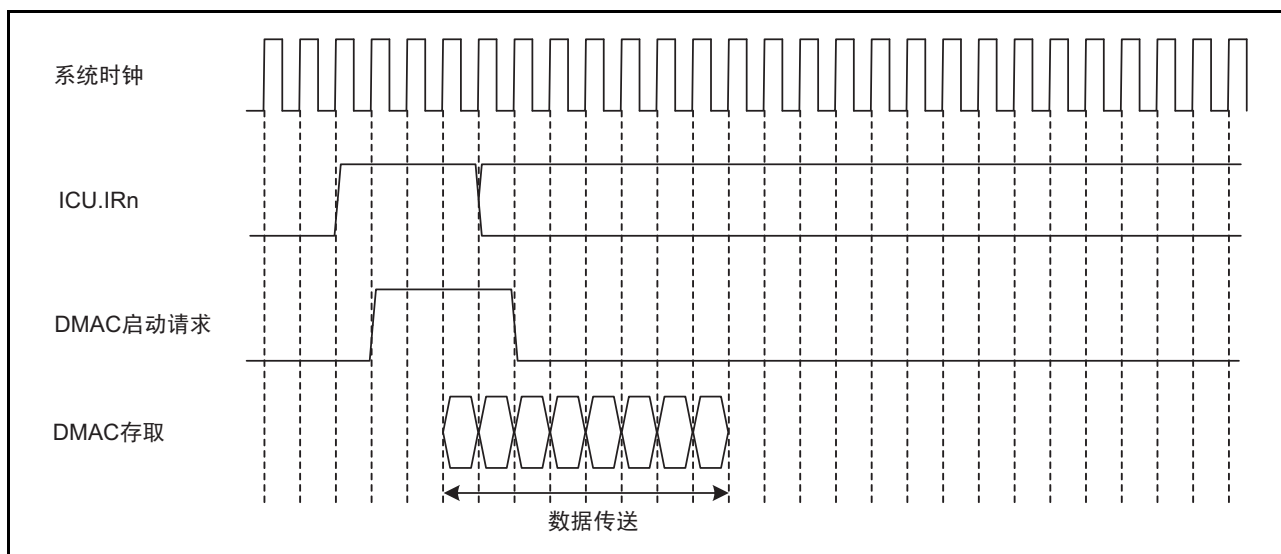


图 16.11 DMAC 的运行时序例子 (2) (在块传送模式中, 块大小为“4”, 通过外围模块或者外部中断输入引脚的中断启动 DMA 的情况)

16.3.6 DMAC 的执行周期

DMAC 的 1 次数据传送的执行状态如表 16.7 所示。

表 16.7 DMAC 的执行周期

传送模式	数据传送 (读)	数据传送 (写)
正常	Cr+1	Cw
重复	Cr+1	Cw
块 (注 1)	P×Cr	P×Cw

注 1. 这是块大小大于等于“2”的情况。当块大小为“1”时, 为正常传送的周期数。

【符号说明】

P: 块大小 (DMCRAH 寄存器的设定值)

Cr: 数据的读目标存取周期

Cw: 数据的写目标存取周期

Cr 和 Cw 因存取目标而不同。各存取目标的周期数请参照“38. RAM”、“39. ROM (保存代码的闪存)”、“5. I/O 寄存器”和“15.2.5 外部总线”。

数据传送 (读) 的“+1”的单位是系统时钟 (ICLK)。

运行例子请参照“16.3.5 运行时序”。

16.3.7 DMAC 的启动

寄存器的设定步骤如图 16.12 所示。

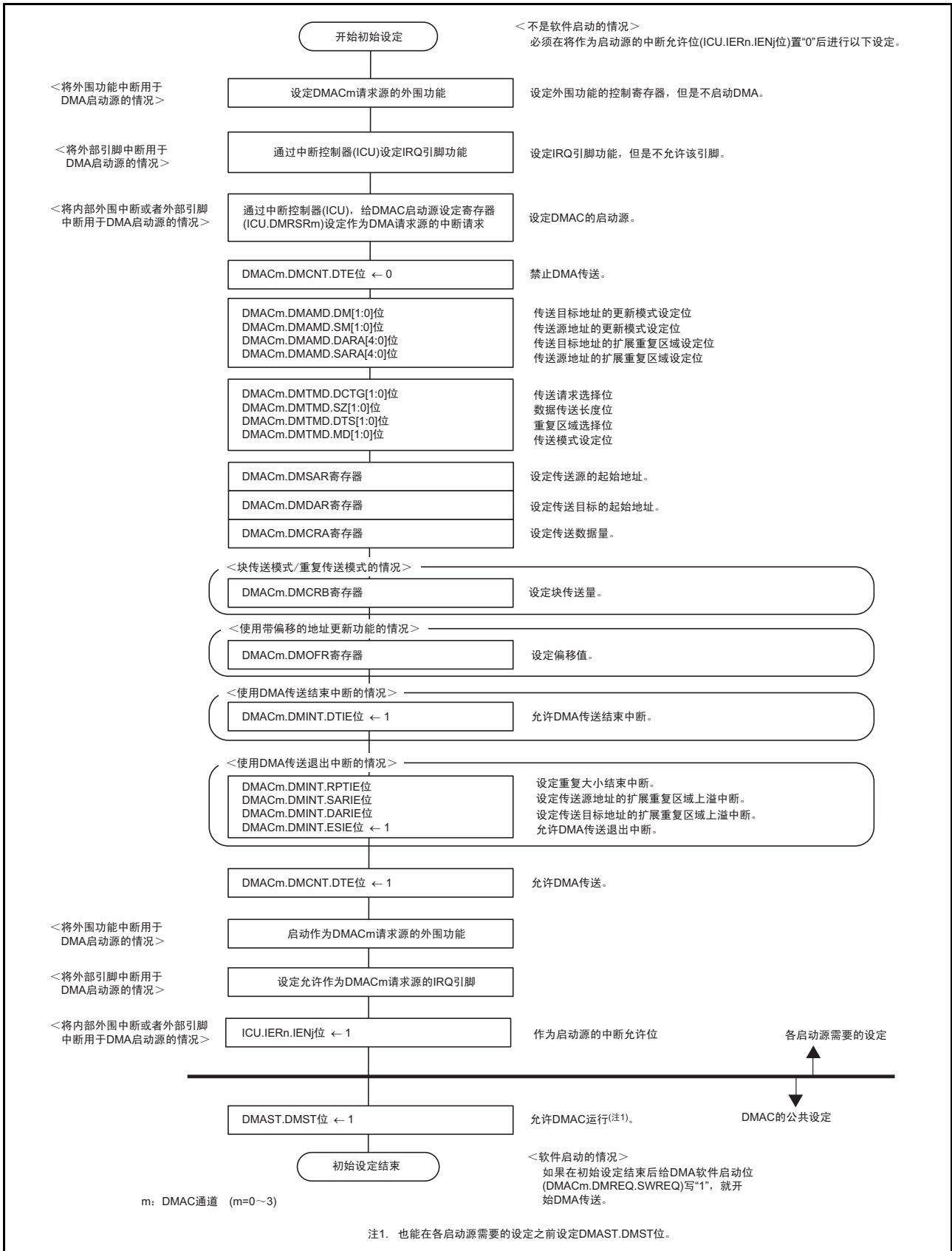


图 16.12 寄存器的设定步骤

16.3.8 DMA 传送的开始

如果在将 DMACm.DMCNT.DTE 位置“1”（允许 DMA 传送）后将 DMAST.DMST 位置“1”（允许启动 DMAC），就能进行通道 m（m=0 ~ 3）的 DMA 传送。

在 DTC 传送过程中，不接受其他 DMAC 通道的新启动请求。在先开始的传送结束时接受优先级最高的通道的 DMA 传送请求，开始 DMA 传送。一旦开始 DMA 传送，DMACm.DMSTS.ACT 位就变为“1”（DMAC 正在运行）。

16.3.9 DMA 传送过程中的寄存器

通过 DMA 传送处理更新 DMAC 寄存器的值，被更新的值因各种设定和传送状态而不同，被更新的寄存器为 DMACm.DMSAR、DMACm.DMDAR、DMACm.DMCRA、DMACm.DMCRB、DMACm.DMCNT 和 DMACm.DMSTS。

(1) DMA 传送源地址寄存器 (DMACm.DMSAR)

如果进行 1 次传送请求的数据传送，就通过下次请求将 DMSAR 寄存器更新为要存取的地址。各模式的寄存器更新请参照表 16.3 ~ 表 16.5。

(2) DMA 传送目标地址寄存器 (DMACm.DMDAR)

如果进行 1 次传送请求的数据传送，就在下次请求时将 DMDAR 寄存器更新为要存取的地址。各模式的寄存器更新请参照表 16.3 ~ 表 16.5。

(3) DMA 传送计数寄存器 (DMACm.DMCRA)

如果进行 1 次传送请求的数据传送，就更新计数值。各模式的更新值不同，各模式的寄存器更新请参照表 16.3 ~ 表 16.5。

(4) DMA 块传送计数寄存器 (DMACm.DMCRB)

如果进行 1 次传送请求的数据传送，就更新计数值。各模式的更新值不同，各模式的寄存器更新请参照表 16.3 ~ 表 16.5。

(5) DMA 传送允许位 (DMACm.DMCNT.DTE)

DMACm.DMCNT.DTE 位通过寄存器的写操作控制允许或者禁止数据传送。根据 DMA 传送状态，如果以下任意一个条件成立，DMACm.DMCNT.DTE 位就自动变为“0”。

- 设定的总传送数据量的传送结束时
- 发生重复大小结束中断请求并且传送结束时
- 发生扩展重复区域上溢中断请求并且传送结束时

禁止写 DMACm.DMCNT.DTE 位为“1”的通道寄存器 (DMACm.DMCNT 寄存器除外)，必须在将 DMACm.DMCNT.DTE 位置“0”的状态下更改各寄存器的设定。

(6) DMA 有效标志 (DMACm.DMSTS.ACT 位)

此标志表示 DMACm 停止或者正在运行。当 DMAC 开始传送时，DMACm.DMSTS.ACT 位为“1”；当结束 1 次传送请求的数据传送时，此标志变为“0”。

即使在 DMA 传送过程中通过给 DMACm.DMCNT.DTE 位写“0”来停止 DMA 传送，DMACm.DMSTS.ACT 位也将“1”的状态保持到 DMA 传送结束为止。

(7) 传送结束中断标志 (DMACm.DMSTS.DTIF 标志)

当通过 DMA 传送结束总传送长度的传送时, 将 DMACm.DMSTS.DTIF 标志置“1”。

如果 DMACm.DMSTS.DTIF 标志为“1”并且 DMACm.DMINT.DTIE 位为“1”, 就产生传送结束中断请求。

在 DMA 传送的总线周期结束后 DMACm.DMSTS.ACT 位变为“0”并且结束传送时, DMACm.DMSTS.DTIF 标志变为“1”。

如果在中断处理过程中给 DMACm.DMCNT.DTE 位写“1”, DMACm.DMSTS.DTIF 标志就自动变为“0”。

(8) 传送退出中断标志 (DMACm.DMSTS.ESIF)

当发生重复大小结束中断请求或者扩展重复区域上溢中断请求时, 将 DMACm.DMSTS.ESIF 标志置“1”。如果 DMACm.DMSTS.ESIF 标志为“1”并且 DMACm.DMINT.ESIE 位为“1”, 就产生传送退出中断请求。

在作为中断请求源的 DMA 传送总线周期结束后 DMACm.DMSTS.ACT 标志变为“0”并且结束传送时, DMACm.DMSTS.ESIF 标志变为“1”。

如果在中断处理过程中给 DMACm.DMCNT.DTE 位写“1”, DMACm.DMSTS.ESIF 标志就自动变为“0”。

要将 DMAC 的中断请求通知 CPU 或者 DTC 时, 需要设定中断控制寄存器, 详细内容请参照“14. 中断控制器 (ICUb)”。

16.3.10 通道的优先级

当有多个 DMA 传送请求时, DMAC 判断有 DMA 传送请求的通道的优先级。通道的优先级固定为通道 0 > 通道 1 > 通道 2 > 通道 3。

如果在数据传送过程中发生 DMA 传送请求, 就在最后的数据传送结束后进行通道仲裁, 开始优先级高的通道的传送。

16.4 DMA 传送的结束

DMA 传送的结束因传送结束条件而不同。一旦 DMA 传送结束，DMACm.DMCNT.DTE 位和 DMACm.DMSTS.ACT 标志就从“1”变为“0”，表示 DMA 传送已经结束。

16.4.1 通过设定的总数据传送结束传送

(1) 正常传送模式 (DMACm.DMTMD.MD[1:0]=00b)

当 DMACm.DMCRAL 寄存器的值从“1”变为“0”时，结束对应通道的 DMA 传送并且将 DMACm.DMCNT.DTE 位清“0”，同时将 DMACm.DMSTS.DTIF 标志置“1”。此时，如果 DMACm.DMINT.DTIE 位为“1”，就向 CPU 或者 DTC 请求传送结束中断。

(2) 重复传送模式 (DMACm.DMTMD.MD[1:0]=01b)

当 DMACm.DMCRB 寄存器的值从“1”变为“0”时，结束对应通道的 DMA 传送并且将 DMACm.DMCNT.DTE 位清“0”，同时将 DMACm.DMSTS.DTIF 标志置“1”。此时，如果 DMACm.DMINT.DTIE 位为“1”，就向 CPU 或者 DTC 请求中断。

(3) 块传送模式 (DMACm.DMTMD.MD[1:0]=10b)

当 DMACm.DMCRB 寄存器的值从“1”变为“0”时，结束对应通道的 DMA 传送并且将 DMACm.DMCNT.DTE 位清“0”，同时将 DMACm.DMSTS.DTIF 标志置“1”。此时，如果 DMACm.DMINT.DTIE 位为“1”，就向 CPU 或者 DTC 请求中断。

要将 DMAC 的中断请求通知 CPU 或者 DTC 时，需要设定中断控制寄存器，详细内容请参照“14. 中断控制器 (ICUb)”。

16.4.2 通过重复大小结束中断结束传送

在重复传送模式中，当 DMACm.DMINT.RPTIE 位为“1”时，在 1 个重复大小的传送结束后发生重复大小结束中断请求。通过中断请求结束 DMA 传送并且将 DMACm.DMCNT.DTE 位清“0”，同时将 DMACm.DMSTS.ESIF 标志置“1”。此时，如果 DMACm.DMINT.ESIE 位为“1”，就向 CPU 或者 DTC 请求中断。如果在此状态下将 DMACm.DMCNT.DTE 位置“1”，就能重新开始传送。

即使在块传送模式中，也能产生重复大小结束中断请求。在块传送模式中，在 1 块数据传送结束后同样发生重复大小结束中断请求。

要将 DMAC 的中断请求通知 CPU 或者 DTC 时，需要设定中断控制寄存器，详细内容请参照“14. 中断控制器 (ICUb)”。

16.4.3 通过扩展重复区域上溢中断结束传送

在指定扩展重复区域并且 DMACm.DMINT.SARIE 位或者 DMACm.DMINT.DARIE 位为“1”时，如果地址的扩展重复区域发生上溢，就产生扩展重复区域上溢中断请求。通过中断请求结束 DMA 传送并且将 DMACm.DMCNT.DTE 位清“0”，同时将 DMACm.DMSTS.ESIF 标志置“1”。此时，如果 DMACm.DMINT.ESIE 位为“1”，就向 CPU 或者 DTC 请求中断。

即使在读周期中发生扩展重复区域上溢中断请求，也执行后续的写周期处理。

在块传送模式中，即使在 1 块数据传送过程中发生扩展重复区域上溢中断请求，也进行 1 块数据的传送。在块大小的边界，通过扩展重复区域上溢中断结束传送。

要将 DMAC 的中断请求通知 CPU 或者 DTC 时，需要设定中断控制寄存器，详细内容请参照“14. 中断控制器 (ICUb)”。

16.5 中断

DMAC 能按各通道，在 1 次请求的传送结束后将中断请求输出到 CPU 或者 DTC。如果传送目标为外部总线或者内部外围总线，就在完成写缓冲器的写操作时而不是在完成实际传送目标的写操作时发行中断请求。

中断源、标志和允许位的关系如表 16.8 所示，中断输出的概略逻辑图如图 16.13 所示。通过 DMAC 中断处理重新开始或者中止 DMA 传送的步骤如图 16.14 所示。

表 16.8 中断源、标志和允许位的关系

中断源	允许位	状态标志	请求输出的允许
传送结束	—	DMACm.DMSTS.DTIF	DMACm.DMINT.DTIE
退出传送结束	重复大小结束	DMACm.DMINT.RPTIE	DMACm.DMINT.ESIE
	传送源地址的扩展重复区域发生上溢	DMACm.DMINT.SARIE	
	传送目标地址的扩展重复区域发生上溢	DMACm.DMINT.DARIE	

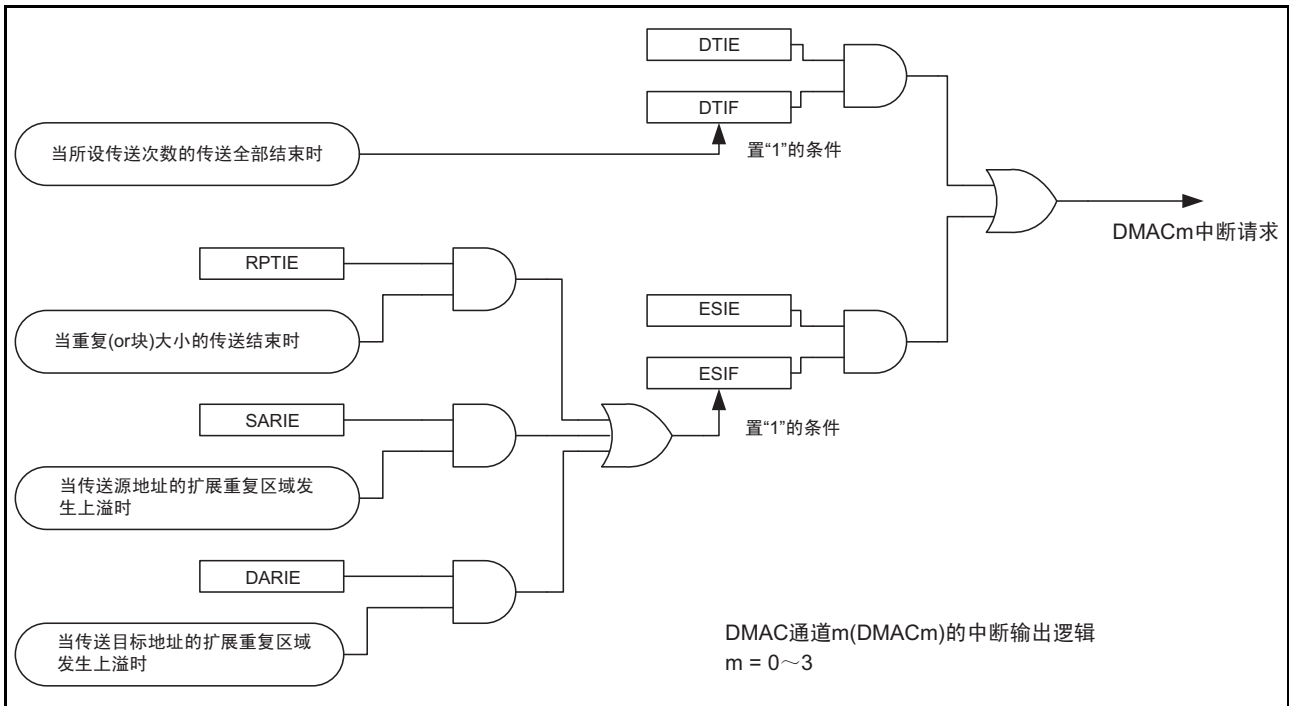


图 16.13 中断输出的概略逻辑图

在结束或者中止 DMA 传送时以及在继续传送时，通过中断处理程序解除中断后重新开始 DMA 传送的方法不同。

(1) 结束或者中止 DMA 传送的情况

在传送结束中断的情况下，必须给 DMACm.DMSTS.DTIF 标志写“0”；在重复大小中断和扩展重复区域上溢中断的情况下，必须给 DMACm.DMSTS.ESIF 标志写“0”。清除中断请求，DMACm 保持停止状态。此后，要进行新的 DMA 传送时，必须给所需寄存器写设定值，并且给 DMACm.DMCNT.DTE 位写“1”（允许 DMA 传送）。

(2) 继续进行 DMA 传送的情况

必须给 DMACm.DMCNT.DTE 位写“1”。自动将 DMACm.DMSTS.ESIF 标志清“0”（清除中断源）并且重新开始传送。

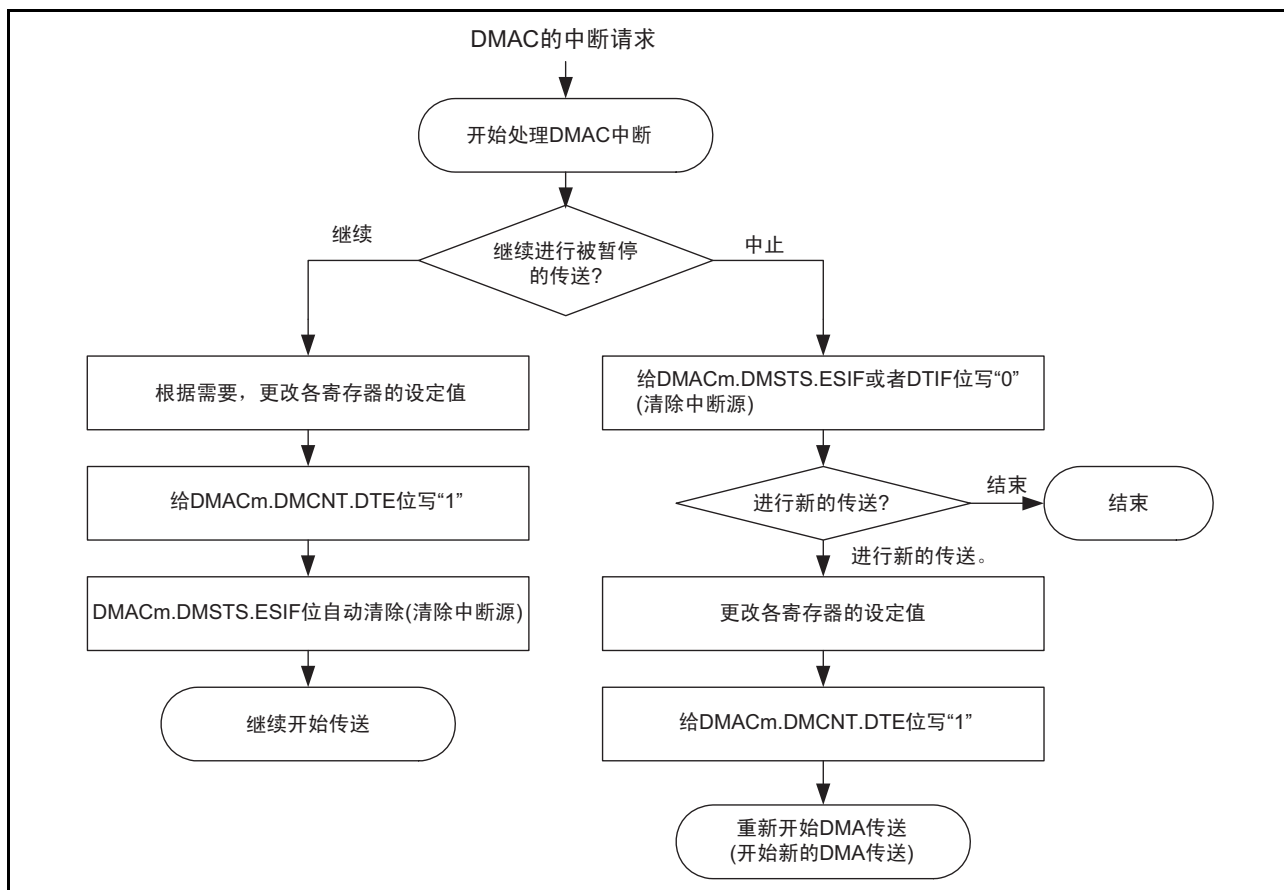


图 16.14 通过 DMAC 中断处理重新开始或者中止 DMA 传送的步骤

16.6 事件链接功能

各 DMAC 通道在进行 1 次数据传送后（块传送时为 1 块传送后）输出事件链接请求。但是，如果传送目标为外部总线或者内部外围总线，就在接受写缓冲器的写操作时，发行事件链接请求。

16.7 低功耗功能

在使用模块停止功能以及向全模块时钟停止模式、软件待机模式或者深度软件待机模式转移时，必须在给 DMAST.DMST 位写“0”（DMAC 模块停止）后分别进行以下的处理。

(1) 模块停止功能

通过给 MSTPCRA.MSTPA28 位写“1”（转移到模块停止状态），使 DMAC 的模块停止功能有效。如果在给 MSTPCRA.MSTPA28 位写“1”时正在进行 DMA 传送，就在 DMA 传送结束后转移到模块停止状态。

不能在 MSTPCRA.MSTPA28 位为“1”时存取 DMAC 的寄存器。

通过给 MSTPCRA.MSTPA28 位写“0”（解除模块停止状态），解除 DMAC 的模块停止状态。

(2) 全模块时钟停止模式

必须按照“11. 低功耗功能”的“11.6.2.1 向全模块时钟停止模式的转移”的步骤进行设定。

如果在执行 WAIT 指令时正在进行 DMA 传送，就在 DMA 传送结束后转移到全模块时钟停止模式。

在从全模块时钟停止模式返回后，通过给 MSTPCRA.MSTPA28 位写“0”，解除 DMAC 的模块停止状态。

(3) 软件待机模式和深度软件待机模式

必须按照“11. 低功耗功能”的“11.6.3.1 向软件待机模式的转移”或者“11.6.4.1 向深度软件待机模式的转移”的步骤进行设定。

如果在执行 WAIT 指令时正在进行 DMA 传送，就在 DMA 传送结束后转移到软件待机模式或者深度软件待机模式。

(4) 使用低功耗功能时的注意事项

有关 WAIT 指令和寄存器的设定步骤，请参照“11. 低功耗功能”的“11.7.6 WAIT 指令的执行时序”。

要在从低功耗模式返回后进行 DMA 传送时，必须再次给 DMAST.DMST 位写“1”。

当全模块时钟停止模式期间或者软件待机模式期间发生的请求不是 DMAC 启动请求而是 CPU 中断请求时，必须按照“14. 中断控制器 (ICUb)”的“14.4.3 中断请求目标的选择”的设定方法，在将中断请求源转换为 CPU 后执行 WAIT 指令。

16.8 使用时的注意事项

16.8.1 使用外部设备的情况

在进行外部设备的 DMA 传送过程中，在从开始写最后数据到外部总线存取结束前，DMACm.DMSTS.ACT 标志有可能变为“0”（DMAC 停止）。

16.8.2 外围模块的 DMA 传送的情况

在进行外围模块的 DMA 传送过程中，在从开始写最后数据到外围总线存取结束前，DMACm.DMSTS.ACT 标志有可能变为“0”（DMAC 停止）。

16.8.3 有关 DMA 运行过程中的寄存器存取

在 DMACm.DMSTS.ACT 标志为“1”（DMAC 正在运行）或者 DMACm.DMCNT.DTE 位为“1”（允许 DMA 传送）的状态下，不能存取相同通道的设定寄存器（DMSAR、DMDAR、DMCRA、DMCRB、DMTMD、DMINT、DMAMD、DMOFR、DMCSL）。

16.8.4 有关保留区的 DMA 传送

因为禁止保留区的 DMA 传送，所以无法保证存取保留区时的传送结果。保留区的详细内容请参照“4. 地址空间”。

16.8.5 因设定 DMA 启动源标志控制寄存器（DMCSL）而产生的各传送结束中断请求

如果将 DMACm.DMCSL.DISEL 位置“1”，就能在每结束 1 次 DMA 启动请求的传送时向 CPU 请求中断。此时发生的中断不同于 DMAC 输出的传送结束中断和退出结束中断，如果在 DMA 传送结束时不将作为 DMAC 启动源的中断标志清“0”，就通过将中断请求目标转换为 CPU，向 CPU 请求中断。在接受 CPU 中断时清除中断标志。

有关中断标志和中断请求目标的变更，请参照“14. 中断控制器（ICUb）”，DMACm.DMCSL.DISEL 位的设定请参照“16.2.12 DMA 启动源标志控制寄存器（DMCSL）”。

16.8.6 中断控制器的 DMAC 启动请求寄存器（ICU.DMRSRm）的设定

必须在 DMA 传送允许位（DMACm.DMCNT.DTE 位）为“0”（禁止 DMA 传送）时设定 DMAC 启动源选择寄存器（ICU.DMRSRm）。不能将和 ICU.DMRSRm 寄存器设定的向量号相同的向量号对应的 DTC 启动允许寄存器（ICU.DTCERn）置“1”。ICU.DTCERn 寄存器和 ICU.DMRSRm 寄存器的详细内容请参照“14. 中断控制器（ICUb）”。

16.8.7 保留或者重新开始 DMA 启动的方法

要保留 DMA 的启动请求时，必须给启动源的中断允许位（ICU.IERn.IENj 位）写“0”；要重新开始 DMA 时，必须在进行“16.3.7 DMAC 的启动”的设定后给 ICU.IERn.IENj 位写“1”。

17. 数据传送控制器 (DTCa)

RX210 群内置数据传送控制器 (DTC)。
能通过中断请求启动 DTC 并且进行数据传送。

17.1 概要

DTC 的规格和框图分别如表 17.1 和图 17.1 所示。

表 17.1 DTC 的规格

项目	内容
传送模式	<ul style="list-style-type: none"> • 正常传送模式 1 次启动进行 1 个数据的传送。 • 重复传送模式 1 次启动进行 1 个数据的传送。 如果传送重复大小的数据, 就返回到开始传送的地址。 重复大小最多能设定 256 个数据。 • 块传送模式 1 次启动进行 1 块数据的传送。 块大小最多能设定 256 个数据。
传送通道	<ul style="list-style-type: none"> • 能进行对应中断源的通道传送 (通过 ICU 的 DTC 启动请求进行传送)。 • 能对 1 个启动源进行多个的数据传送 (链传送)。 • 能选择是在计数器为“0”时进行链传送还是每次都进行链传送。
传送空间	<ul style="list-style-type: none"> • 在短地址模式中为 16M 字节。 (0000 0000h ~ 007F FFFFh 和 FF80 0000h ~ FFFF FFFFh 中保留区除外的区域) • 在全地址模式中为 4G 字节。 (0000 0000h ~ FFFF FFFFh 中保留区除外的区域)
数据传送单位	<ul style="list-style-type: none"> • 1 个数据的位长: 8 位、16 位、32 位 • 块大小的数据量: 1 ~ 256 个数据
CPU 中断请求	<ul style="list-style-type: none"> • 能通过启动 DTC 的中断向 CPU 请求中断。 • 在 1 次数据传送结束后向 CPU 请求中断。 • 在指定数据量的数据传送结束后向 CPU 请求中断。
事件链接功能	在进行 1 次数据传送后 (块传送时为 1 块传送后), 产生事件链接请求。
跳读	能指定传送信息的跳读。
回写的省略	当传送源地址被固定或者传送目标被固定时省略回写。
低功耗功能	能设定为模块停止状态。

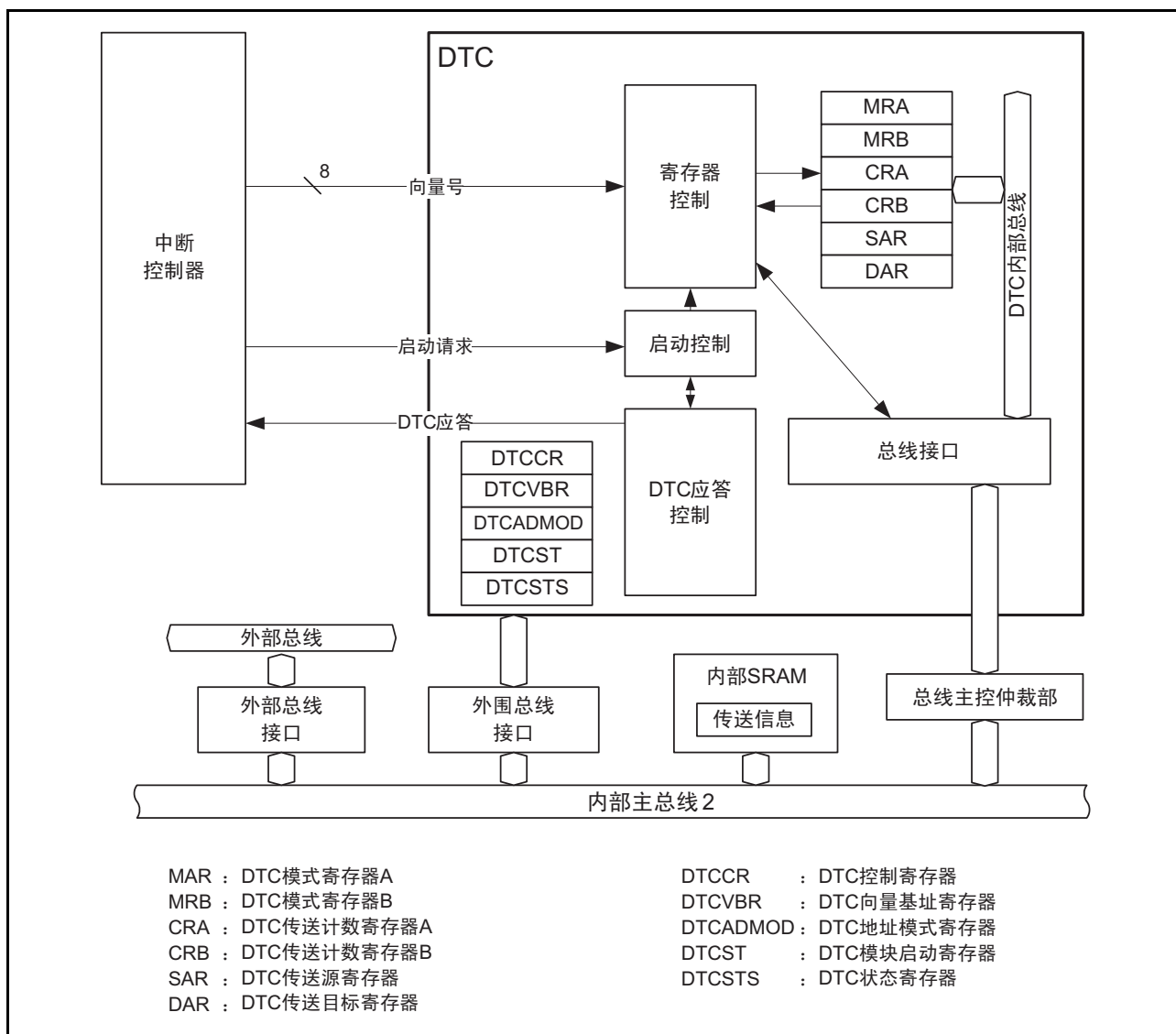


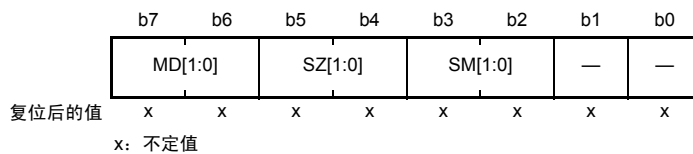
图 17.1 DTC 的框图

17.2 寄存器说明

MRA、MRB、SAR、DAR、CRA、CRB 寄存器是 DTC 的内部寄存器，不能从 CPU 直接存取，这些内部寄存器的设定值作为传送信息被分配到 RAM 区。如果发生 DTC 启动请求，就从 RAM 区读传送信息并且将读到的传送信息设定到内部寄存器。在进行数据传送后，DTC 的内部寄存器的内容作为传送信息被回写到 RAM 区。

17.2.1 DTC 模式寄存器 A (MRA)

地址 (CPU 不能直接存取)



位	符号	位名	功能	R/W
b1-b0	—	保留位	读取值为不定值，只能写“0”。	—
b3-b2	SM[1:0]	传送源地址的寻址方式位	b3 b2 0 0: SAR 寄存器为地址固定 (省略 SAR 寄存器的回写) 0 1: SAR 寄存器为地址固定 (省略 SAR 寄存器的回写) 1 0: 传送后 SAR 寄存器递增 (当 SZ[1:0] 位为“00b”时加 1; 为“01b”时加 2; 为“10b”时加 4) 1 1: 传送后 SAR 寄存器递减 (当 SZ[1:0] 位为“00b”时减 1; 为“01b”时减 2; 为“10b”时减 4)	—
b5-b4	SZ[1:0]	DTC 数据传送长度位	b5 b4 0 0: 8 位 (字节) 传送 0 1: 16 位 (字) 传送 1 0: 32 位 (长字) 传送 1 1: 不能设定	—
b7-b6	MD[1:0]	DTC 传送模式选择位	b7 b6 0 0: 正常传送模式 0 1: 重复传送模式 1 0: 块传送模式 1 1: 不能设定	—

MBA 寄存器是选择 DTC 的运行模式的寄存器。
CPU 不能直接存取 MRA 寄存器。

SM[1:0] 位 (传送源地址的寻址方式位)

这些位指定传送数据后的 SAR 寄存器的运行。

SZ[1:0] 位 (DTC 数据传送长度位)

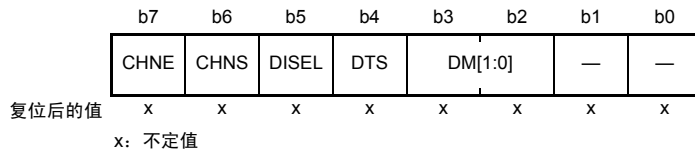
这些位指定传送数据的长度。

MD[1:0] 位 (DTC 传送模式选择位)

这些位指定 DTC 的传送模式。

17.2.2 DTC 模式寄存器 B (MRB)

地址 (CPU 不能直接存取)



位	符号	位名	功能	R/W
b1-b0	—	保留位	读取值为不定值, 只能写“0”。	—
b3-b2	DM[1:0]	传送目标地址的寻址方式位	b3 b2 0 0: DAR 寄存器为地址固定 (省略 DAR 寄存器的回写) 0 1: DAR 寄存器为地址固定 (省略 DAR 寄存器的回写) 1 0: 传送后 DAR 寄存器递增 (当 MAR.SZ[1:0] 位为“00b”时加 1; 为 “01b”时加 2; 为“10b”时加 4) 1 1: 传送后 DAR 寄存器递减 (当 MAR.SZ[1:0] 位为“00b”时减 1; 为 “01b”时减 2; 为“10b”时减 4)	—
b4	DTS	DTC 传送模式选择位	0: 传送目标为重复区域或者块区域 1: 传送源为重复区域或者块区域	—
b5	DISEL	DTC 中断选择位	0: 在指定的数据传送结束时向 CPU 请求中断 1: 在每次 DTC 数据传送时都向 CPU 请求中断	—
b6	CHNS	DTC 链传送选择位	0: 连续进行链传送 1: 只在传送计数器从“0”变为“1”或者从“1”变为 CRAH 时进行链传送	—
b7	CHNE	DTC 链传送允许位	0: 禁止链传送 1: 允许链传送	—

MRB 寄存器是选择 DTC 的运行模式的寄存器。
CPU 不能直接存取 MRB 寄存器。

DM[1:0] 位 (传送目标地址的寻址方式位)

这些位指定传送数据后的 DAR 寄存器的运行。

DTS 位 (DTC 传送模式选择位)

在重复传送模式或者块传送模式中, 此位指定是将传送源还是将传送目标作为重复区域或者块区域。

DISEL 位 (DTC 中断选择位)

此位指定是在每次 DTC 数据传送时都向 CPU 请求中断, 还是只在数据传送结束时向 CPU 请求中断。

CHNS 位 (DTC 链传送选择位)

此位选择链传送的条件。

当 CHNE 位为“0”时，忽视 CHNS 位的设定。有关链传送选择条件的详细内容，请参照“表 17.3 链传送的条件”。

如果下一次传送是链传送，就不判断指定次数的传送结束，并且不清除中断状态标志，也不向 CPU 请求中断。

CHNE 位 (DTC 链传送允许位)

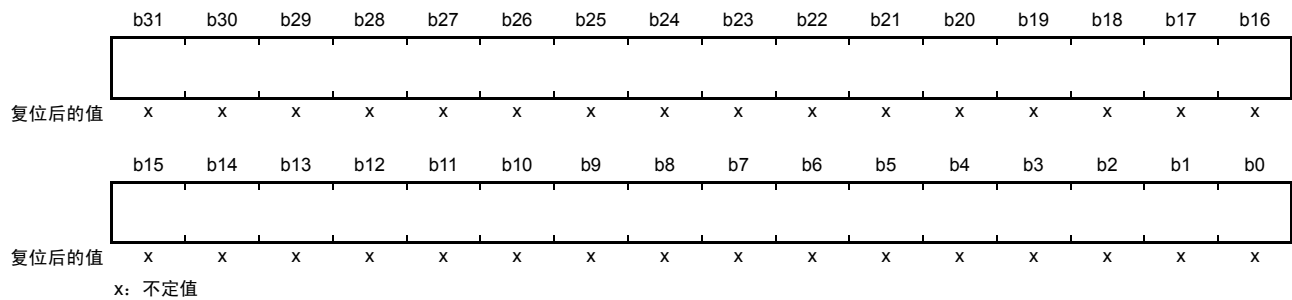
此位指定链传送。

通过 CHNS 位选择链传送的条件。

有关链传送的详细内容，请参照“17.4.6 链传送”。

17.2.3 DTC 传送源寄存器 (SAR)

地址 (CPU 不能直接存取)



SAR 寄存器是设定传送源的起始地址的寄存器。

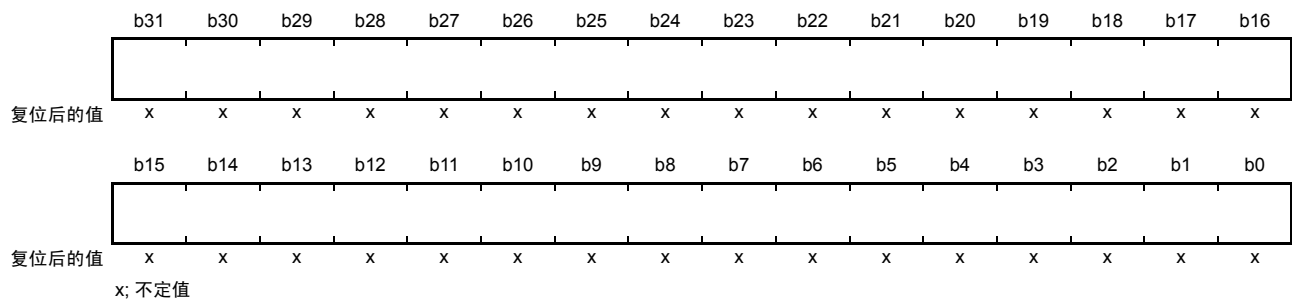
在全地址模式中 32 位有效。

在短地址模式中低 24 位有效，忽视高 8 位 (b31 ~ b24) 的设定，通过 b23 指定的值进行位扩展。

CPU 不能直接存取 SAR 寄存器。

17.2.4 DTC 传送目标寄存器 (DAR)

地址 (CPU 不能直接存取)



DAR 寄存器是设定传送目标的起始地址的寄存器。

在全地址模式中 32 位有效。

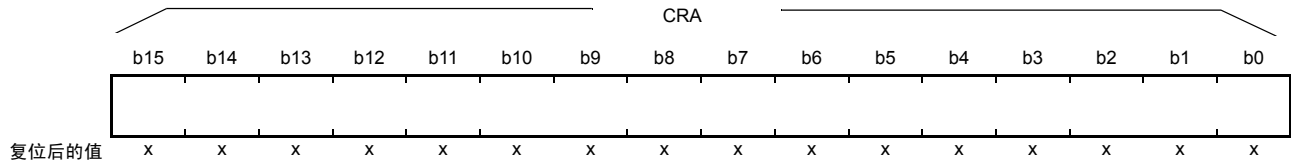
在短地址模式中低 24 位有效，忽视高 8 位 (b31 ~ b24) 的设定，通过 b23 指定的值进行位扩展。

CPU 不能直接存取 DAR 寄存器。

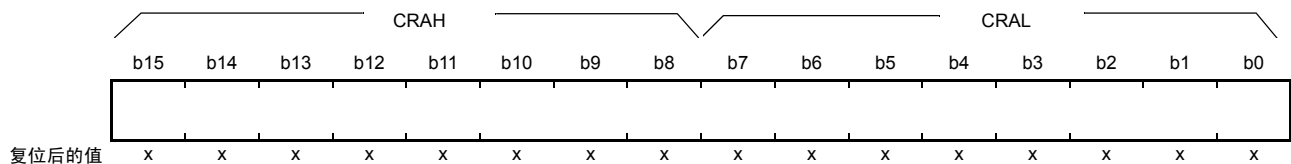
17.2.5 DTC 传送计数器 A (CRA)

地址 (CPU 不能直接存取)

• 正常传送模式



• 重复传送模式、块传送模式



x: 不定值

注: 功能因传送模式而不同。

符号	寄存器名	功能	R/W
CRAL	传送计数器 A 的低位寄存器	设定传送次数。	—
CRAH	传送计数器 A 的高位寄存器		—

注: 在重复传送模式和块传送模式中, 必须给 CRAH 寄存器和 CRAL 寄存器设定相同的值。

CRA 寄存器是指定 DTC 的传送次数的寄存器。

功能因传送模式而不同。

CPU 不能直接存取 CRA 寄存器。

(1) 正常传送模式 (MRA.MD[1:0] 位 =00b)

在正常传送模式中, CRA 寄存器用作 16 位传送计数器。

当设定值为“0001h”时, 传送次数为 1 次; 当设定值为“FFFFh”时, 传送次数为 65535 次; 当设定值为“0000h”时, 传送次数为 65536 次。

每进行 1 次数据传送, CRA 寄存器就减 1。

(2) 重复传送模式 (MRA.MD[1:0] 位 =01b)

CRAH 寄存器保持传送次数, CRAL 寄存器用作 8 位传送计数器。

当设定值为“01h”时, 传送次数为 1 次; 当设定值为“FFh”时, 传送次数为 255 次; 当设定值为“00h”时, 传送次数为 256 次。

每进行 1 次数据传送, CRAL 寄存器就减 1。当 CRAL 寄存器变为“00h”时, 就传送 CRAH 寄存器的值。

(3) 块传送模式 (MRA.MD[1:0] 位 =10b)

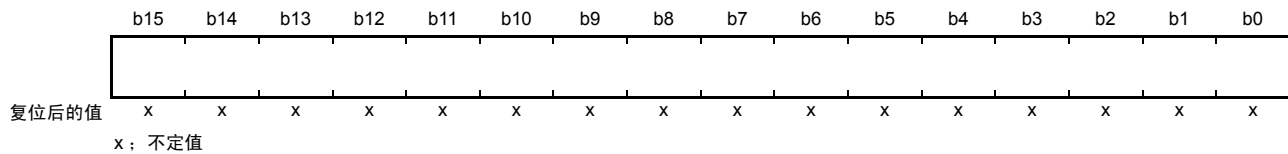
CRAH 寄存器保持块的大小, CRAL 寄存器用作 8 位块大小计数器。

当设定值为“01h”时, 传送次数为 1 次; 当设定值为“FFh”时, 传送次数为 255 次; 当设定值为“00h”时, 传送次数为 256 次。

每进行 1 次数据传送, CRAL 寄存器就减 1。当 CRAL 寄存器变为“00h”时, 传送 CRAH 寄存器的值。

17.2.6 DTC 传送计数器 B (CRB)

地址 (CPU 不能直接存取)



CRB 寄存器是指定块传送模式的块传送次数的寄存器。

当设定值为“0001h”时，传送次数为 1 次；当设定值为“FFFFh”时，传送次数为 65535 次；当设定值为“0000h”时，传送次数为 65536 次。

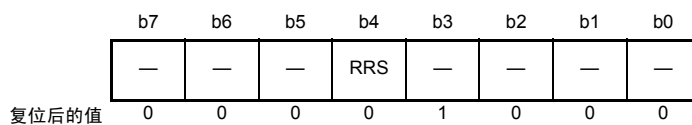
当进行 1 块大小的最后数据传送时，CRB 寄存器减 1。

在设定为正常传送模式和重复传送模式时，不使用 CRB 寄存器，忽视设定值。

CPU 不能直接存取 CRB 寄存器。

17.2.7 DTC 控制寄存器 (DTCCR)

地址 0008 2400h



位	符号	位名	功能	R/W
b2-b0	—	保留位	读写值都为“0”。	R/W
b3	—	保留位	读写值都为“1”。	R/W
b4	RRS	DTC 传送信息的跳读允许位	0: 不跳读传送信息 1: 在向量号相同时跳读传送信息	R/W
b7-b5	—	保留位	读写值都为“0”。	R/W

DTCCR 寄存器是控制 DTC 的寄存器。

RRS 位 (DTC 传送信息的跳读允许位)

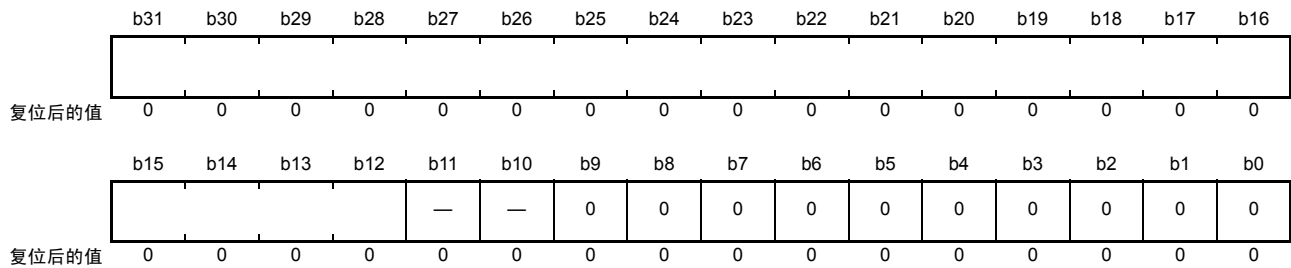
DTC 向量号随时和上次启动的向量号进行比较。

如果向量号相同并且 RRS 位为“1”，就不读传送信息而进行 DTC 的数据传送。如果上次的启动为链传送，就读传送信息，与 RRS 位的值无关。

在上次的传送为正常传送并且传送计数器 (CRA 寄存器) 为“0”时，或者在上次的传送为块传送并且传送计数器 (CRB 寄存器) 为“0”时，都读传送信息，与 RRS 位的值无关。

17.2.8 DTC 向量基址寄存器 (DTCVBR)

地址 0008 2404h



DTCVBR 寄存器设定在计算 DTC 向量表地址时的基址。

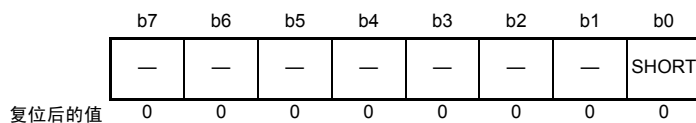
b11 位和 b10 位为保留位。读写值都为“0”。

必须将低 10 位 (b9 ~ b0) 固定为“0”，写操作无效。

忽视高 4 位 (b31 ~ b28) 的写操作，通过 b27 指定的值进行位扩展。

17.2.9 DTC 地址模式寄存器 (DTCADM0D)

地址 0008 2408h



位	符号	位名	功能	R/W
b0	SHORT	短地址模式设定位	0: 全地址模式 1: 短地址模式	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

DTCADM0D 寄存器是设定 DTC 能存取的地域的寄存器。

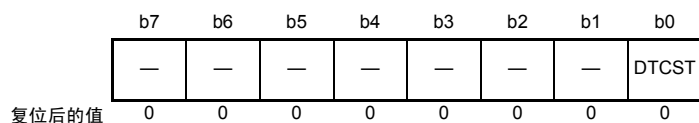
SHORT 位 (短地址模式设定位)

在全地址模式中，能存取 4G 字节空间 (00000000h ~ FFFFFFFFh)。

在短地址模式中，能存取 16M 字节空间 (00000000h ~ 007FFFFFFh 和 FF800000h ~ FFFFFFFFh)。

17.2.10 DTC 模块启动寄存器 (DTCST)

地址 0008 240Ch



位	符号	位名	功能	R/W
b0	DTCST	DTC 模块启动位	0: DTC 模块停止 1: DTC 模块运行	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

DTCST 位 (DTC 模块启动位)

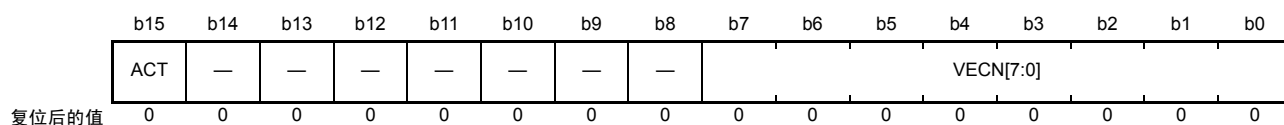
为了使 DTC 能接受传送请求, 必须将 DTCST 位置“1”。如果将 DTCST 位置“0”, 就不能接受新的传送请求。如果在运行过程中将此位改写为“0”, 已接受的传送请求就保持到处理结束为止。

在使用模块停止功能以及向全模块时钟停止模式、软件待机模式、深度软件待机模式转移时, 必须将 DTCST 位置“0”。

有关使用模块停止功能以及向全模块时钟停止模式、软件待机模式、深度软件待机模式的转移, 请参照“17.9 低功耗功能”和“11. 低功耗功能”。

17.2.11 DTC 状态寄存器 (DTCSTS)

地址 0008 240Eh



位	符号	位名	功能	R/W
b7-b0	VECN[7:0]	DTC 有效向量号监视位	在 DTC 传送过程中，用向量号表示该传送的启动源。只在 DTC 传送过程中 (ACT 标志为“1”)，此值为有效值。	R
b14-b8	—	保留位	读取值为“0”，写操作无效。	R
b15	ACT	DTC 有效标志	0: 不在进行 DTC 传送 1: 正在进行 DTC 传送	R

VECN[7:0] 位 (DTC 有效向量号监视位)

在 DTC 传送过程中，用向量号表示该传送的启动源。

如果在读 DTCSTS 寄存器时 ACT 标志为“1” (正在进行 DTC 传送)，读到的 VECN[7:0] 就为有效值；如果在读 DTCSTS 寄存器时 ACT 标志为“0” (不在进行 DTC 传送)，读到的 VECN[7:0] 就为无效值。

DTC 启动源和向量地址的关系请参照“14. 中断控制器 (ICUb)”的“表 14.3 中断向量表”。

ACT 标志 (DTC 有效标志)

表示 DTC 的传送状态。

[为“1”的条件]

- 当通过传送请求进行了 DTC 启动时

[为“0”的条件]

- 当对于 1 次传送请求 DTC 运行结束时

17.3 启动源

通过中断请求启动 DTC。如果将启动 DTC 的中断对应的中断控制器 (ICU) 的 $DTCERn.DTCE$ 位 (n 为中断向量号) 置“1”，该中断就为 DTC 的启动源。

DTC 启动源和向量地址的关系请参照“14. 中断控制器 (ICUb)”的“表 14.3 中断向量表”。软件启动请参照“14. 中断控制器 (ICUb)”的“14.2.5 软件中断启动寄存器 (SWINTR)”。

一旦接受启动请求，DTC 就在该 1 次请求的传送结束前不接受新的启动请求，与优先级无关。如果在 DMAC/DTC 传送过程中产生多个启动源，就在该传送结束时接受最高优先级的请求。如果在 DTC 模块启动位 ($DTCST.DTCST$) 为“0”的状态下产生多个启动请求，就在将 $DTCST.DTCST$ 位置“1”时接受最高优先级的请求。

在开始 1 次数据传送 (在链传送时为连续的最后传送) 或者结束数据传送时，DTC 进行以下的运行：

- 在指定的总传送量的传送结束时，在数据传送后将 $DTCERn.DTCE$ 位置“0”，然后向 CPU 请求中断。
- 当 $MRB.DISEL$ 位为“1”时，在数据传送后向 CPU 请求中断。
- 如果是上述以外的情况，就在开始数据传送或者结束数据传送时将作为启动源的中断状态标志置“0”。

17.3.1 传送信息的分配和 DTC 向量表

DTC 按启动源从向量表读传送信息的起始地址，并且从该起始地址读传送信息。

必须分配向量表的基址 (起始地址)，使低 12 位变为“0”。必须给 DTC 向量基址寄存器 ($DTCVBR$) 设定 DTC 向量表的基址。

传送信息分配在 RAM 区。对于向量表的基址，向量号 n 的传送信息 (n) 的起始地址必须为地址 $+4n$ 。

能通过短地址模式 (3 个长字) 或者全地址模式 (4 个长字) 分配传送信息。通过 $DTCADMOD.SHORT$ 位设定短地址模式 ($SHORT$ 位 = 1) 或者全地址模式 ($SHORT$ 位 = 0)。

DTC 向量表和传送信息的对应如图 17.2 所示。

RAM 区内传送信息的分配如图 17.3 所示。低位地址因分配区域的字节序而不同，详细内容请参照“17.10.2 传送信息的分配”。

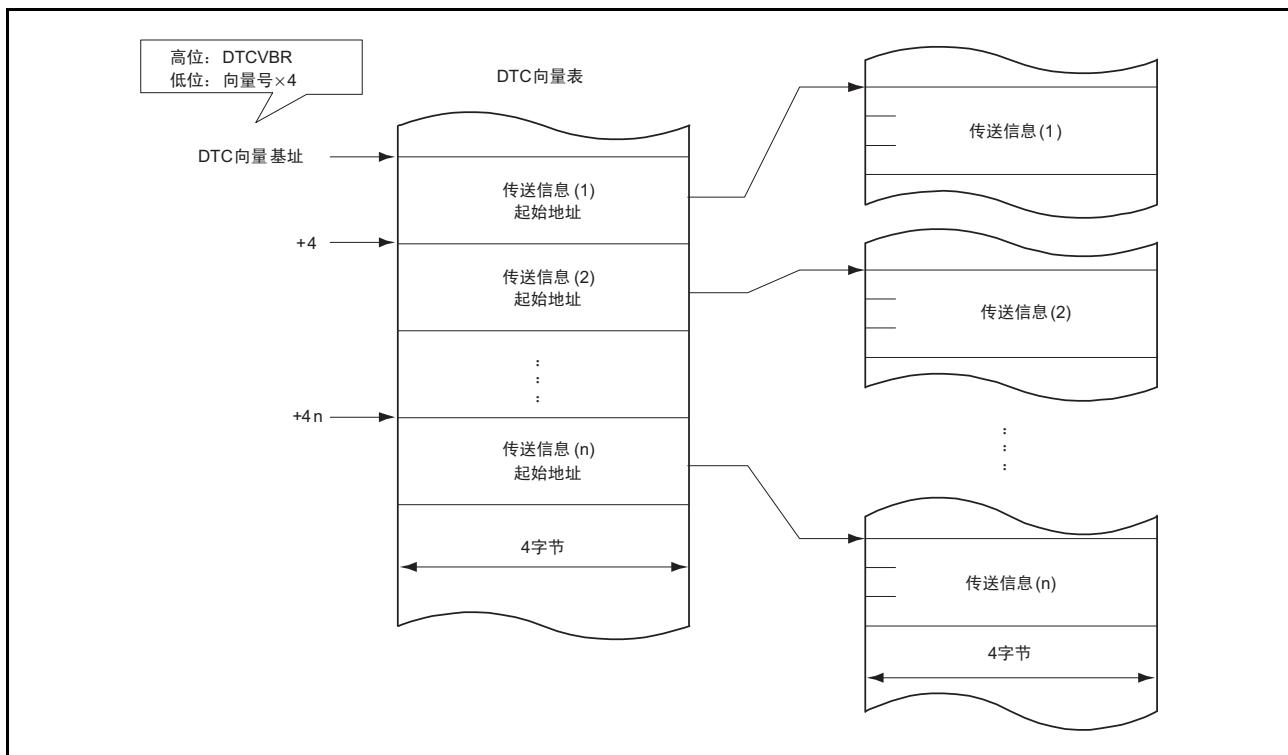


图 17.2 DTC 向量表和传送信息的对应

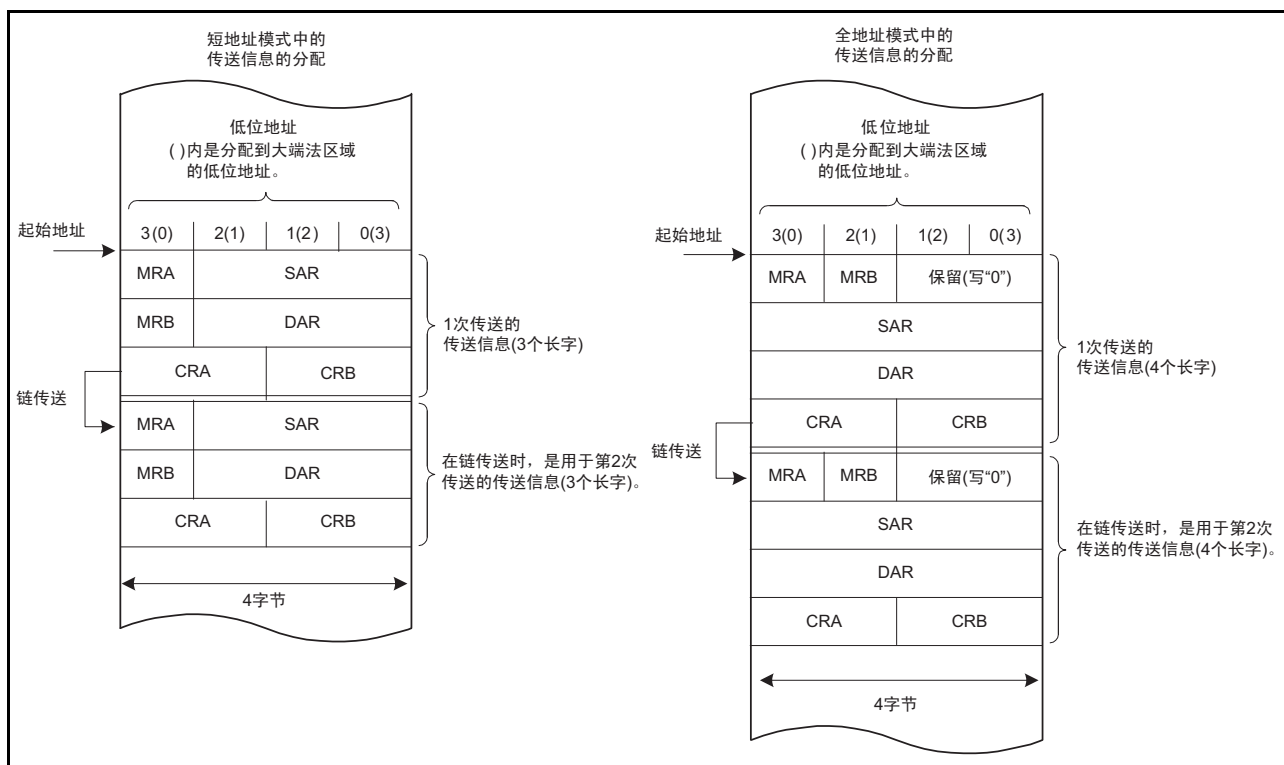


图 17.3 RAM 区内的传送信息的分配

17.4 运行说明

DTC 以传送信息为准进行数据传送。为了使 DTC 运行，需要预先将传送信息保存到 RAM 区。

一旦启动 DTC，就读与向量号对应的 DTC 向量。然后从 DTC 向量所示的传送信息的保存地址读传送信息，并且进行数据传送以及回写数据传送后的传送信息。能通过将传送信息保存到 RAM 区，进行任意个通道的数据传送。

传送模式有正常传送模式、重复传送模式和块传送模式。

DTC 通过 SAR 寄存器指定传送源地址，通过 DAR 寄存器指定传送目标地址。在传送后，SAR 寄存器和 DAR 寄存器分别递增、递减或者为地址固定。

DTC 的传送模式如表 17.2 所示。

表 17.2 DTC 的传送模式

传送模式	1 次传送请求能传送的数据长度	存储器地址的增减	能指定的传送次数
正常传送模式	1 字节 / 字 / 长字	增减 1、2、4 或者地址固定	1 ~ 65536 次
重复传送模式 (注 1)	1 字节 / 字 / 长字	增减 1、2、4 或者地址固定	1 ~ 256 次 (注 3)
块传送模式 (注 2)	CRAH 寄存器指定的块大小 (1 ~ 256 字节 / 字 / 长字)	增减 1、2、4 或者地址固定	1 ~ 65536 次

注 1. 将传送源或者传送目标设定为重复区域。

注 2. 将传送源或者传送目标设定为块区域。

注 3. 在指定次数的传送结束后，恢复初始状态并且继续（重复）运行。

能通过先将 MRB.CHNE 位置“1”，用 1 个启动源进行多次传送（链传送）。也能通过设定 MRB.CHNS 位，在指定的数据传送结束时进行链传送的设定。

DTC 的运行流程图如图 17.4 所示，链传送的条件如表 17.3 所示。

表 17.3 链传送的条件

第 1 次传送				第 2 次传送 (注 3)				DTC 传送
CHNE 位	CHNS 位	DISEL 位	传送计数器 (注 1、注 2)	CHNE 位	CHNS 位	DISEL 位	传送计数器 (注 1、注 2)	
0	—	0	(1→0) 以外	—	—	—	—	到第 1 次传送为止结束。
0	—	0	(1→0)	—	—	—	—	到第 1 次传送为止结束并且 向 CPU 请求中断。
0	—	1	—	—	—	—	—	
1	0	—	—	0	—	0	(1→0) 以外	到第 2 次传送为止结束。
				0	—	0	(1→0)	到第 2 次传送为止结束并且 向 CPU 请求中断。
				0	—	1	—	
1	1	0	(1→*) 以外	—	—	—	—	到第 1 次传送为止结束。
1	1	—	(1→*)	0	—	0	(1→0) 以外	到第 2 次传送为止结束。
				0	—	0	(1→0)	到第 2 次传送为止结束并且 向 CPU 请求中断。
				0	—	1	—	
1	1	1	(1→*) 以外	—	—	—	—	到第 1 次传送为止结束并且 向 CPU 请求中断。

注 1. 传送计数器因各传送模式而不同，各传送模式中的传送计数器如下：

正常传送模式：CRA 寄存器

重复传送模式：CRAL 寄存器

块传送模式：CRB 寄存器

注 2. 在正常传送模式或者块传送模式中，传送结束时的计数器变化为 (1→0)；在重复传送模式中，最后传送时的计数器变化为 (1→CRAH)。表中的 (1→*) 表示两种情况的变化。

注 3. 在第 2 次传送或者第 2 次以后的传送时能选择链传送，但是在第 2 次传送时省略 CHNE 位为“1”的组合。

17.4.1 传送信息的跳读功能

能通过设定 DTCCR.RRS 位，跳读向量地址和传送信息。

在发生 DTC 启动请求时，这次启动的 DTC 向量号随时和上次启动的 DTC 向量号进行比较。如果比较结果相同并且 RRS 位为“1”，就不读向量地址和传送信息而进行 DTC 的数据传送。如果上次启动为链传送，就读向量地址和传送信息。如果上次传送为正常传送并且传送计数器 (CRA 寄存器) 变为“0”或者上次传送为块传送并且传送计数器 (CRB 寄存器) 变为“0”，就读传送信息，与 RRS 位的值无关。跳读传送信息的运行例子如图 17.13 所示。

要更新 DTC 向量表和传送信息时，必须先将 RRS 位置“0”并且在更新 DTC 向量表和传送信息后设定 RRS 位。一旦将 RRS 位置“0”，就取消被保持的向量号，在下次启动时能读被更新的 DTC 向量表和传送信息。

17.4.2 传送信息的回写省略功能

如果将 MRA.SM[1:0] 位或者 MRB.DM[1:0] 位设定为“地址固定”，就不回写部分传送信息，此功能与短地址模式和全地址模式的设定无关。传送信息的回写省略条件以及省略回写的寄存器如表 17.4 所示。

与短地址模式和全地址模式的设定无关，回写 CRA 寄存器和 CRB 寄存器。在全地址模式中，省略 MRA 寄存器和 MRB 寄存器的回写。

表 17.4 传送信息的回写省略条件以及省略回写的寄存器

MRA.SM[1:0] 位		MRB.DM[1:0] 位		SAR 寄存器	DAR 寄存器
b3	b2	b3	b2		
0	0	0	0	省略	省略
0	0	0	1		
0	1	0	0		
0	1	0	1		
0	0	1	0	省略	回写
0	0	1	1		
0	1	1	0		
0	1	1	1		
1	0	0	0	回写	省略
1	0	0	1		
1	1	0	0		
1	1	0	1		
1	0	1	0	回写	回写
1	0	1	1		
1	1	1	0		
1	1	1	1		

17.4.3 正常传送模式

用 1 个启动源进行 1 个字节、1 个字或者 1 个长字的传送，传送次数为 1 ~ 65536 次。

能将传送源地址和传送目标地址分别设定为递增、递减或者固定。一旦指定次数的传送结束，就能向 CPU 请求中断。

正常传送模式的寄存器功能和存储器映像分别如表 17.5 和图 17.5 所示。

表 17.5 正常传送模式的寄存器功能

寄存器	功能	回写传送信息时被回写的值
SAR	传送源地址	递增 / 递减 / 固定 (注 1)
DAR	传送目标地址	递增 / 递减 / 固定 (注 1)
CRA	传送计数器 A	CRA-1
CRB	传送计数器 B	不更新

注 1. 在地址固定的情况下，省略回写。

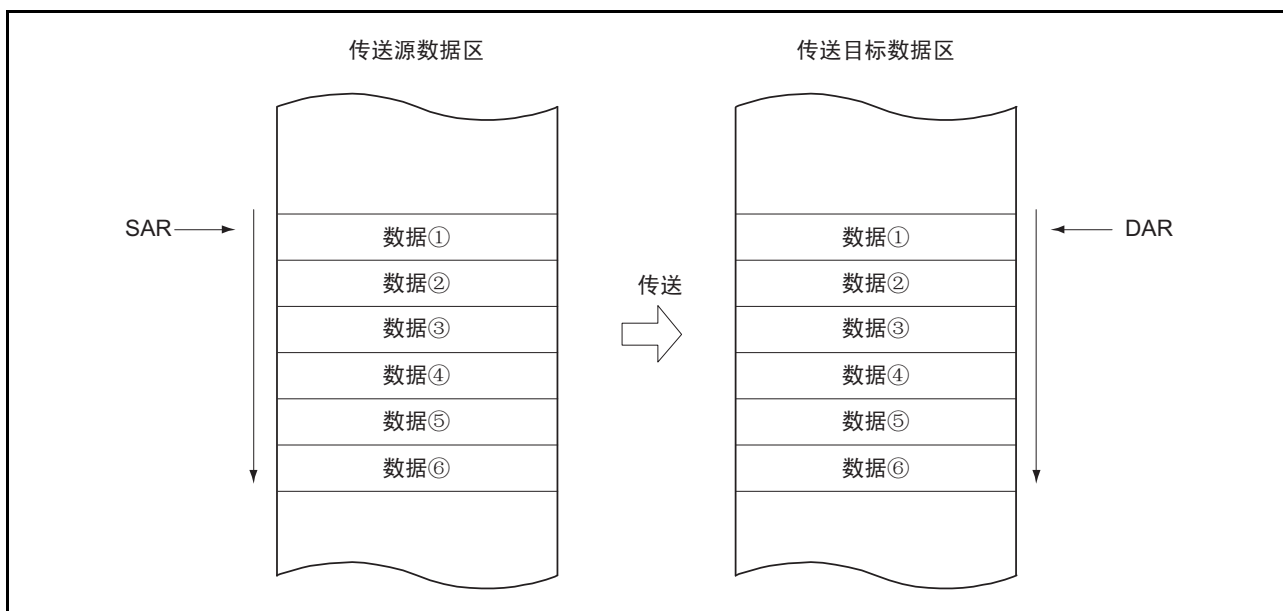


图 17.5 正常传送模式的存储器映像

17.4.4 重复传送模式

用 1 个启动源进行 1 个字节、1 个字或者 1 个长字的传送。

通过 MRB.DTS 位将传送源或者传送目标指定为重复区域，传送次数可指定为 1 ~ 256 次。一旦指定次数的传送结束，传送计数器和被指定为重复区域的地址寄存器就恢复初始状态并且进行重复传送。其他的地址寄存器继续递增、递减或者为地址固定。

在重复传送模式中，如果传送计数器的 CRAL 寄存器的值变为“00h”，CRAL 寄存器的值就被更新为 CRAH 寄存器的设定值。因为传送计数器不变为“00h”，所以在 MRB.DISEL 位为“0”（在指定的数据传送结束时向 CPU 请求中断）时不向 CPU 请求中断。

重复传送模式的寄存器功能和存储器映像分别如表 17.6 和图 17.6 所示。

表 17.6 重复传送模式的寄存器功能

寄存器	功能	写传送信息时被回写的值	
		CRAL 不为“1”	CRAL 为“1”
SAR	传送源地址	递增 / 递减 / 固定 (注 1)	(当 MRB.DTS 位为“0”时) 递增 / 递减 / 固定 (注 1) (当 MRB.DTS 位为“1”时) SAR 寄存器的初始值
DAR	传送目标地址	递增 / 递减 / 固定 (注 1)	(当 MRB.DTS 位为“0”时) DAR 寄存器的初始值 (当 MRB.DTS 位为“1”时) 递增 / 递减 / 固定 (注 1)
CRAH	传送计数器的保持	CRAH	CRAH
CRAL	传送计数器 A	CRAL-1	CRAH
CRB	传送计数器 B	不更新	不更新

注 1. 在地址固定的情况下，省略回写。

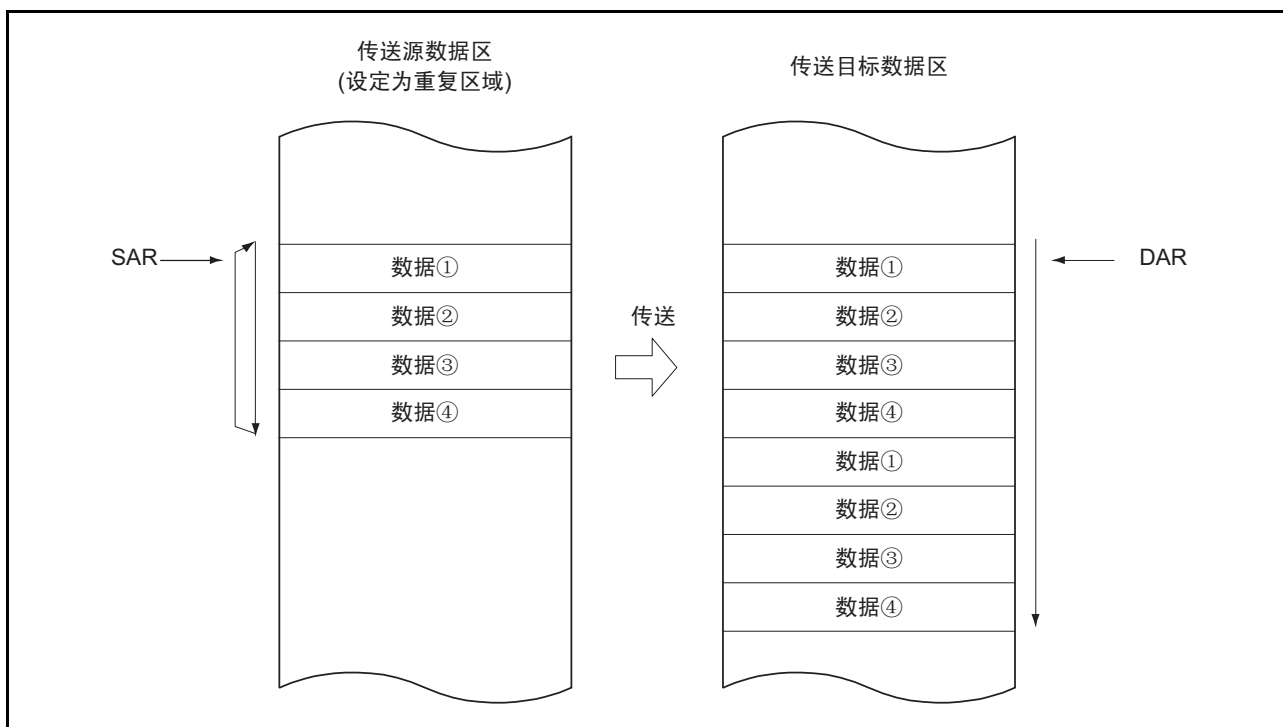


图 17.6 重复传送模式的存储器映像 (将传送源设定为重复区域的情况)

17.4.5 块传送模式

用 1 个启动源进行 1 个块的传送。

通过 MRB.DTS 位将传送源或者传送目标指定为块区域，块大小可指定为 1 ~ 256 字节（或者 1 ~ 256 个字、1 ~ 256 个长字）。

一旦指定的 1 个块的传送结束，块大小计数器的 CRAL 寄存器和被指定为块区域的地址寄存器（当 MRB.DTS 位为“1”时为 SAR 寄存器，当 MRB.DTS 位为“0”时为 DAR 寄存器）就恢复初始状态。其他的地址寄存器继续递增、递减或者为地址固定。

传送次数（块数）能指定为 1 ~ 65536 次。一旦指定次数的块传送结束，就能向 CPU 请求中断。

块传送模式的寄存器功能和存储器映像分别如表 17.7 和图 17.7 所示。

表 17.7 块传送模式的寄存器功能

寄存器	功能	回写传送信息时被回写的值
SAR	传送源地址	（当 MRB.DTS 位为“0”时） 递增 / 递减 / 固定（注 1） （当 MRB.DTS 位为“1”时） SAR 寄存器的初始值
DAR	传送目标地址	（当 MRB.DTS 位为“0”时） DAR 寄存器的初始值 （当 MRB.DTS 位为“1”时） 递增 / 递减 / 固定（注 1）
CRAH	块大小的保持	CRAH
CRAL	块大小计数器	CRAH
CRB	块传送次数计数器	CRB-1

注 1. 在地址固定的情况下，省略回写。

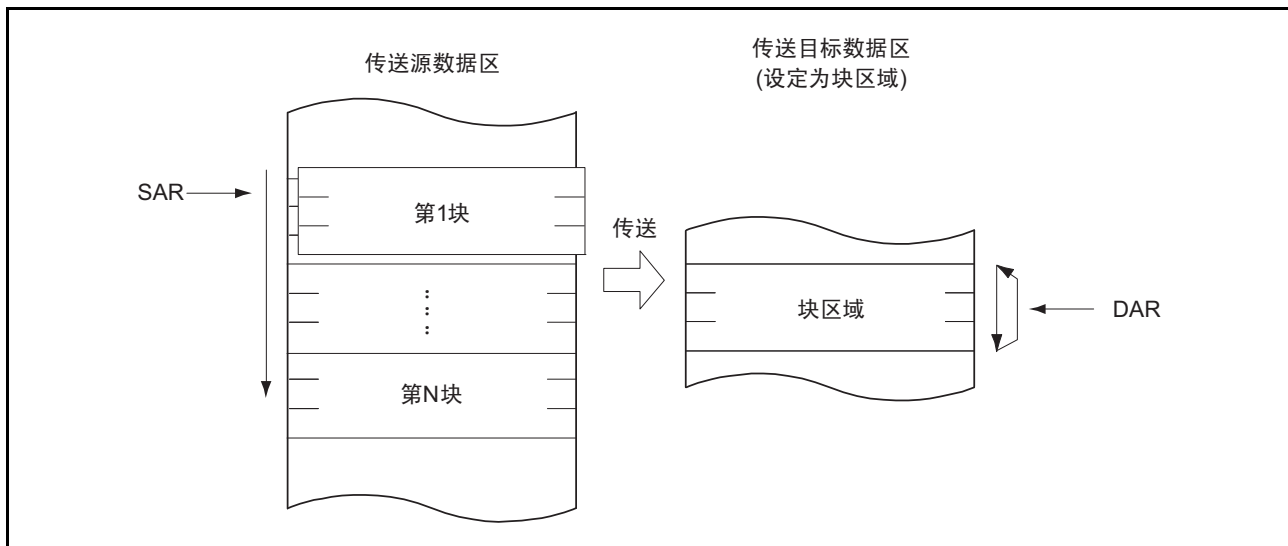


图 17.7 块传送模式的存储器映像（将传送目标指定为块区域的情况）

17.4.6 链传送

如果将 MRB.CHNE 位置“1”，就能进行链传送。链传送用 1 个启动源连续进行多个数据传送。

如果将 MRB.CHNE 位置“1”并且将 MRB.CHNS 位置“0”，就在指定的传送次数结束或者 MRB.DISEL 位为“1”（每次进行 DTC 数据传送时向 CPU 请求中断）时不向 CPU 请求中断，并且也不影响作为启动源的中断状态标志。

能分别设定要定义数据传送的 SAR、DAR、CRA、CRB、MRA、MRB 寄存器。链传送的运行如图 17.8 所示。

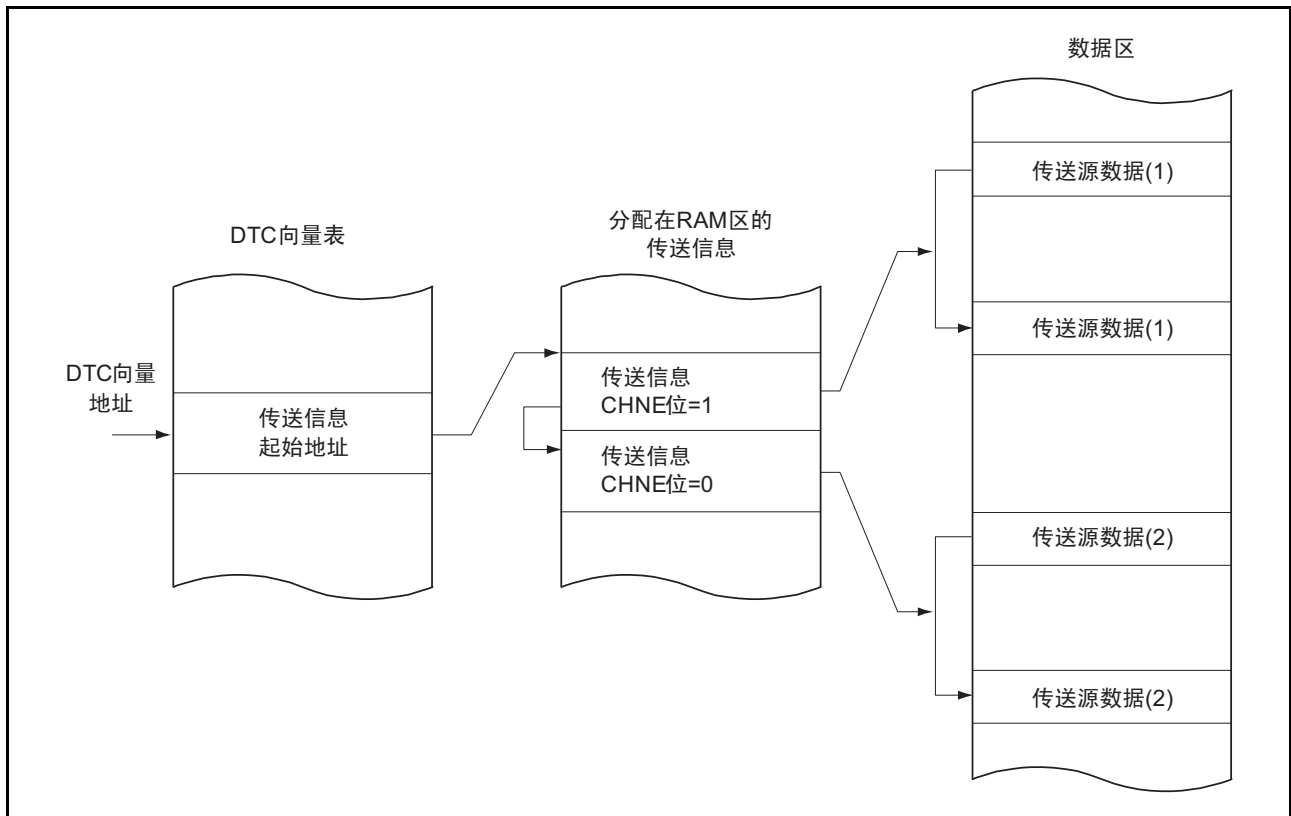


图 17.8 链传送的运行

如果将 MRB.CHNE 位和 MRB.CHNS 位置“1”，就只在指定的数据传送结束时进行链传送。即使在重复传送模式中，也在指定的数据传送结束时进行链传送。

有关链传送条件的详细内容，请参照“表 17.3 链传送的条件”。

17.4.7 运行时序

DTC 的运行时序例子如图 17.9 ~ 图 17.13 所示。

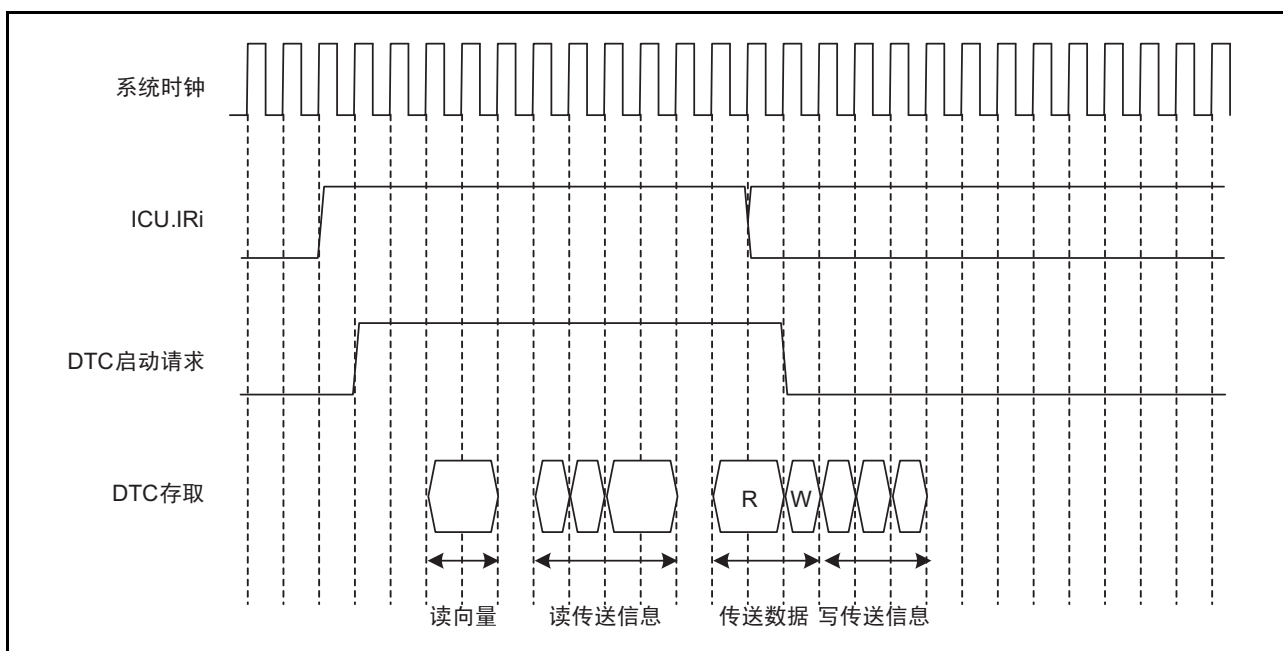


图 17.9 DTC 的运行时序例子 (1) (短地址模式、正常传送模式、重复传送模式的情况)

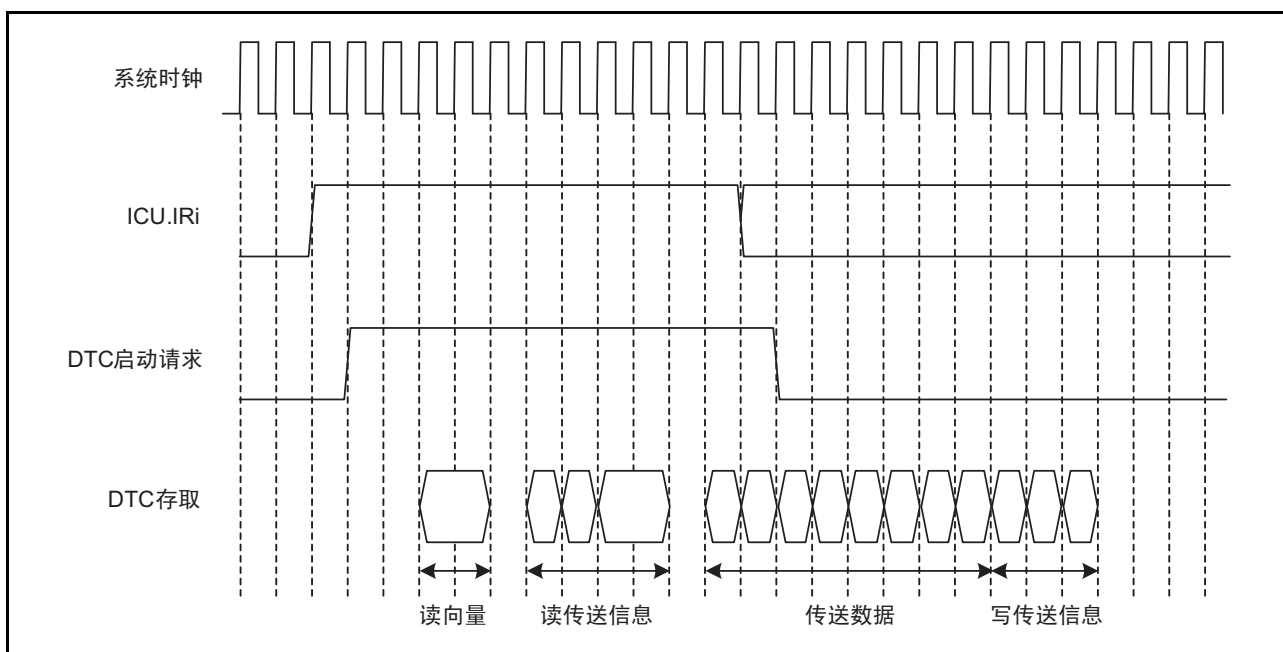


图 17.10 DTC 的运行时序例子 (2) (短地址模式、块传送模式、块大小为“4”的情况)

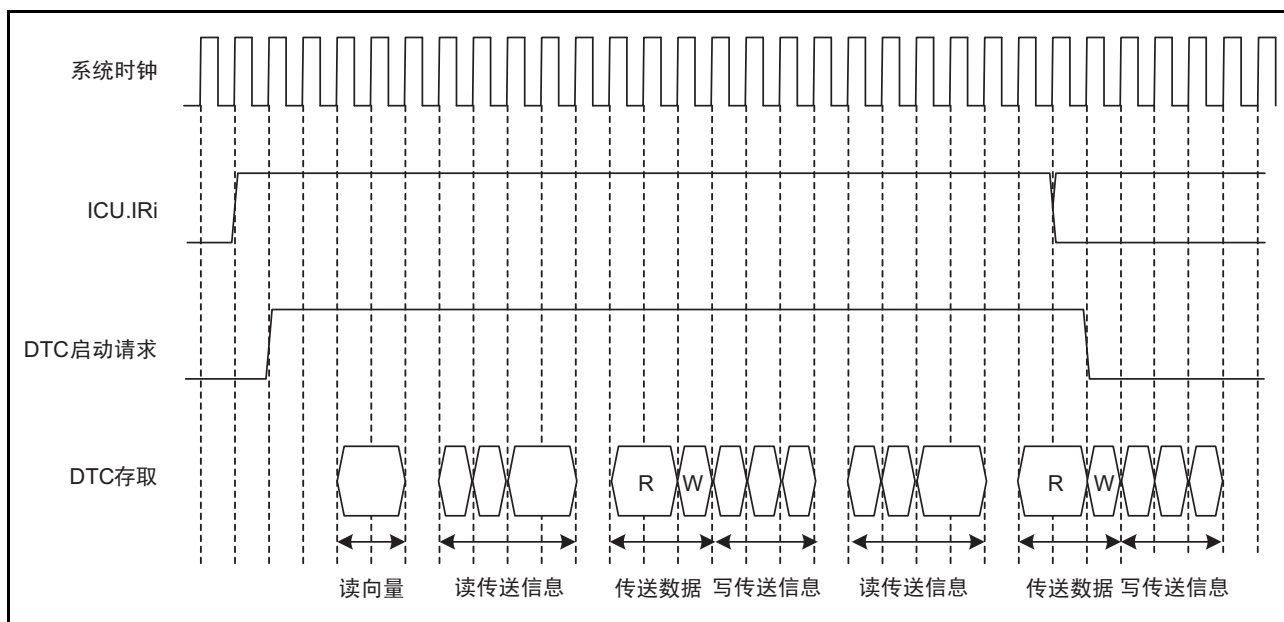


图 17.11 DTC 的运行时序例子 (3) (短地址模式、链传送的情况)

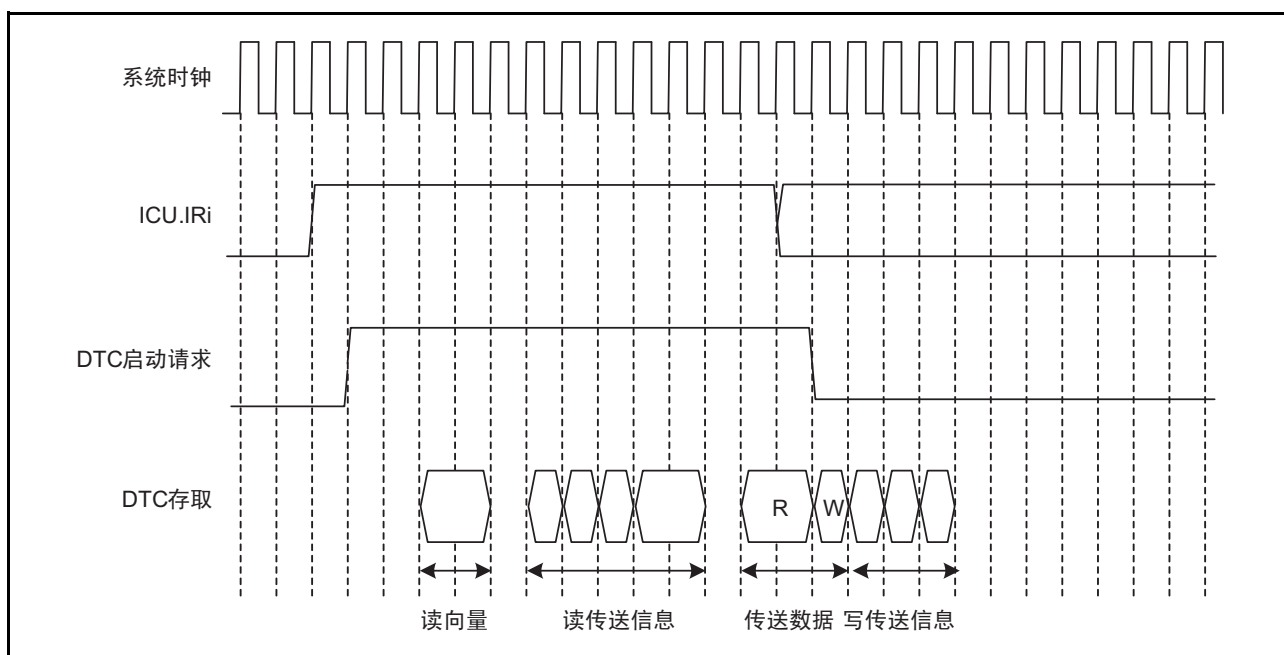


图 17.12 DTC 的运行时序例子 (4) (全地址模式、正常传送模式、重复传送模式的情况)

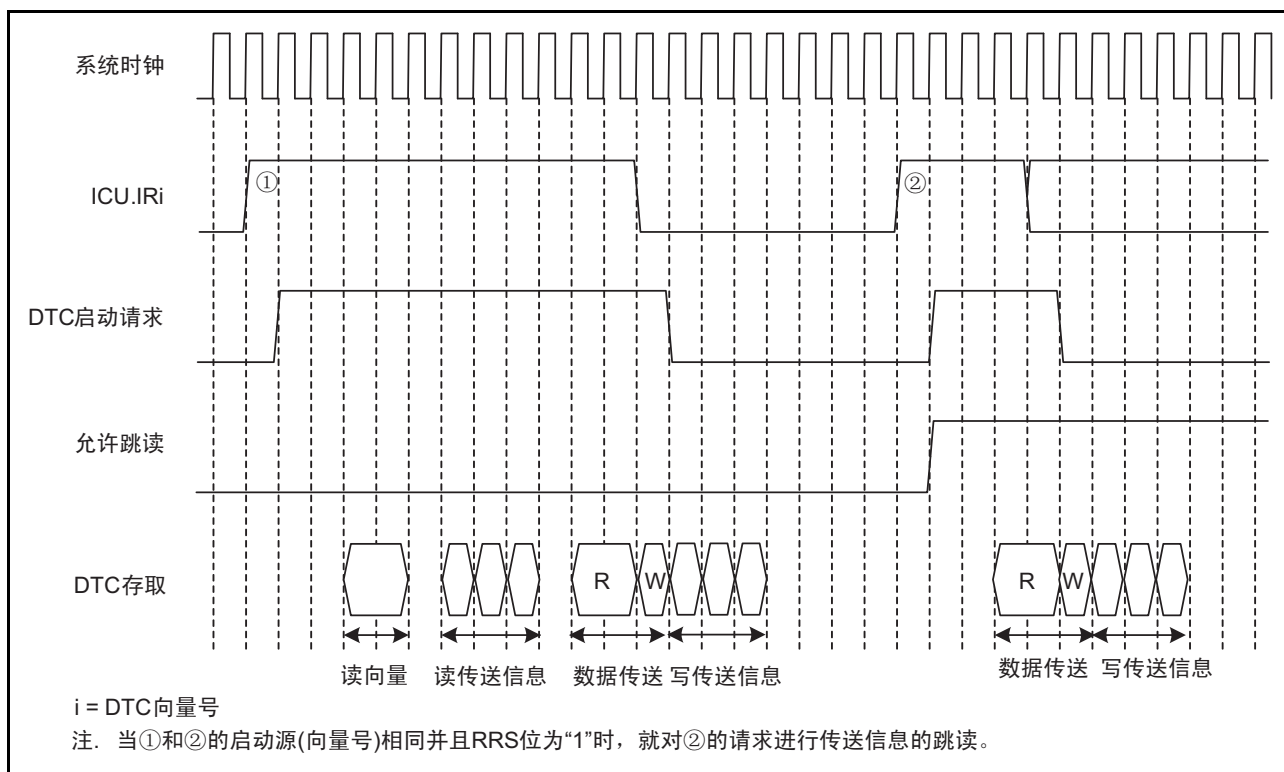


图 17.13 跳读传送信息的运行例子 (向量、传送信息、传送目标为内部 RAM, 传送源为外围模块的情况)

17.4.8 DTC 的执行周期

DTC 的 1 次数据传送的执行周期如表 17.8 所示。
各处理状态的执行顺序请参照“17.4.7 运行时序”。

表 17.8 DTC 的执行周期

传送模式	读向量		读传送信息				写传送信息			数据传送		内部运行	
										读	写		
正常	Cv+1	0 (注 1)	4×Ci+1 (注 2)	3×Ci+1 (注 3)	0 (注 1)	3×Ci (注 4)	2×Ci (注 5)	Ci (注 6)	Cr+1	Cw	2	0 (注 1)	
重复									Cr+1	Cw			
块 (注 7)									P×Cr	P×Cw			

- 注 1. 这是省略传送信息的情况。
 注 2. 这是全地址模式运行的情况。
 注 3. 这是短地址模式运行的情况。
 注 4. SAR 寄存器和 DAR 寄存器都不为地址固定的情况。
 注 5. SAR 寄存器或者 DAR 寄存器为地址固定的情况。
 注 6. SAR 寄存器和 DAR 寄存器为地址固定的情况。
 注 7. 这是块大小大于等于“2”的情况。当块大小为“1”时，为正常传送的周期数。

【符号说明】

P: 块大小 (CRAH 寄存器和 CRAL 寄存器的设定值)

Cv: 向量传送信息保存目标的存取周期

Ci: 传送信息保存目标的存取周期

Cr: 数据读目标的存取周期

Cw: 数据写目标的存取周期

(读向量、读传送信息、读数据传送的“+1”、内部运行的“2”的单位都是系统时钟 (ICLK)。)

(Cv、Ci、Cr 和 Cw 因存取目标而不同。各存取目标的周期数请参照“38. RAM”、“39. ROM (保存代码的闪存)”、“5. I/O 寄存器”和“15.2.5 外部总线”。)

17.4.9 DTC 的总线权释放时序

在读写传送信息过程中，DTC 不释放总线权。在其他时序中，根据总线主控仲裁部决定的优先级进行总线仲裁。有关总线仲裁请参照“15. 总线”。

17.5 DTC 的设定步骤

必须在使用 DTC 前设定 DTC 向量基址寄存器 (DTCVBR)。

启动 DTC 所需的设定步骤如图 17.14 所示。

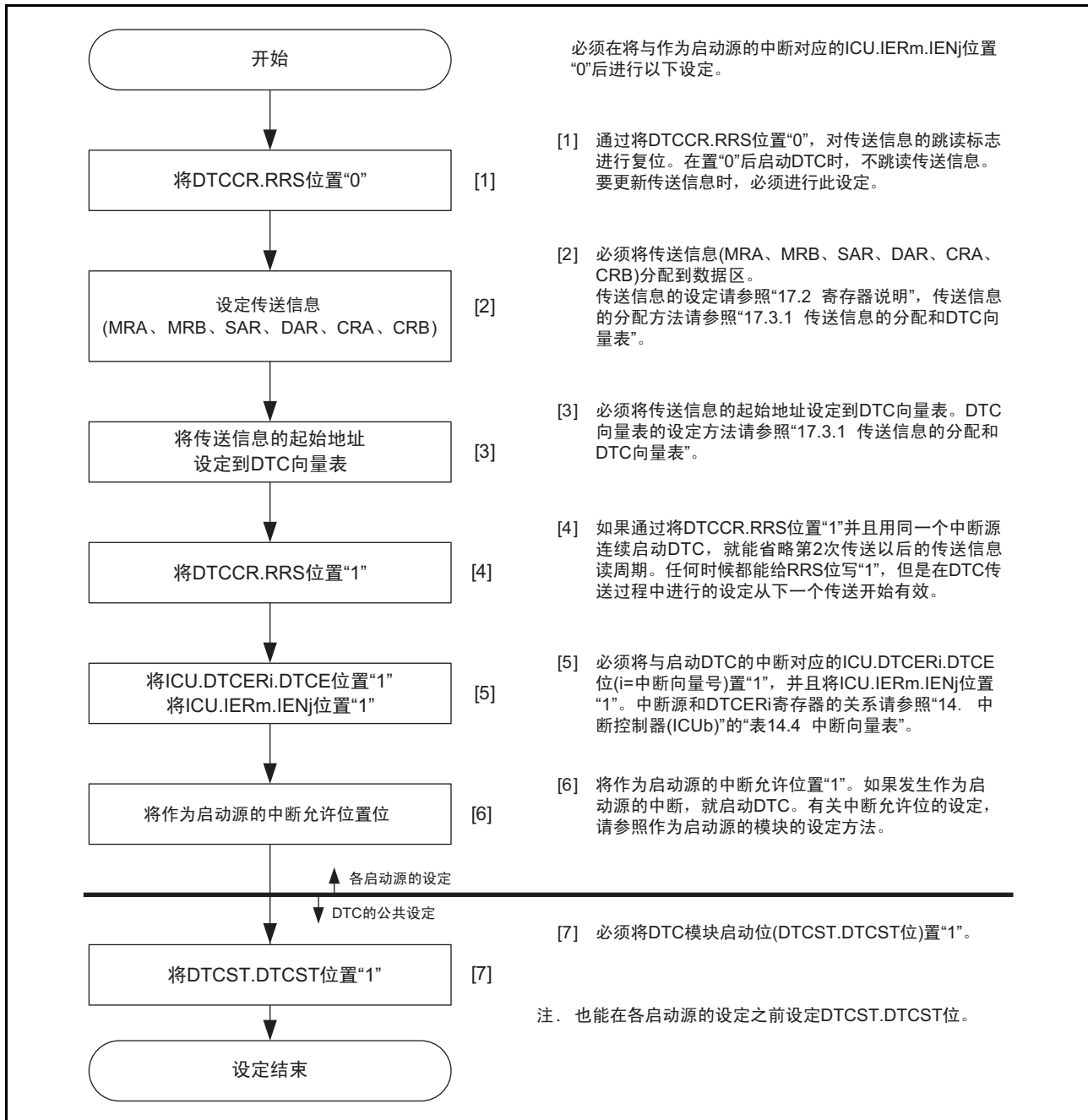


图 17.14 DTC 的设定步骤

17.6 DTC 的使用例子

17.6.1 正常传送

以 SCI 进行的 128 字节数据接收为例说明 DTC 的使用。

(1) 传送信息的设定

给 MRA 寄存器设定传送源地址固定 (MRA.SM[1:0] 位 =00b)、正常传送模式 (MRA.MD[1:0] 位 =00b) 和字节大小 (MRA.SZ[1:0] 位 =00b)；给 MRB 寄存器设定传送目标地址递增 (MRB.DM[1:0] 位 =10b)、1 次中断进行 1 次数据传送 (MRB.CHNE 位 =0 并且 MRB.DISEL 位 =0)，能给 MRB.DTS 位设定任意值；给 SAR 寄存器设定 SCI_m.RDR 寄存器 (m=0 ~ 12) 的地址，给 DAR 寄存器设定用于保存数据的 RAM 起始地址，并且给 CRA 寄存器设定 128 (“0080h”)。CRB 寄存器可以是任意值。

(2) DTC 向量表的设定

将用于 RXI 中断的传送信息起始地址设定到 DTC 向量表。

(3) ICU 的设定和 DTC 模块的启动

将对应的 ICU.DTCER_i.DTCE 位置 “1”，将 ICU.IER_i.IEN_j 位和 DTCST.DTCST 位置 “1”。

(4) SCI 的设定

将 SCI_m.SCR.RIE 位置 “1” 并且允许接收结束 (RXI) 中断。如果在 SCI 接收过程中发生接收错误，就不进行以后的接收，因此必须设定为 CPU 能接受接收错误中断。

(5) DTC 传送

在每次 SCI 的 1 字节数据接收结束时发生 RXI 中断，启动 DTC。通过 DTC 将接收数据从 SCI_m.RDR 寄存器传送到 RAM，并且进行 DAR 寄存器的递增和 CRA 寄存器的递减。

(6) 中断处理

如果在 128 次的传送结束后 CRA 寄存器变为 “0”，就向 CPU 请求 RXI 中断。必须通过中断处理程序进行结束处理。

17.6.2 计数器为“0”时的链传送

只能在计数器变为“0”时进行第2次数据的传送。能通过重新设定第1次数据的传送，至少进行256次的重复传送。

这是构成128K字节输入缓冲器的例子，假设输入缓冲器从低位地址“0000h”开始。计数器为“0”时的链传送如图17.15所示。

1. 作为第1次数据传送，设定输入数据的正常传送模式。传送源地址必须固定，CRA寄存器为“0000h”（65536次），MRB.CHNE位为“1”（允许链传送），MRB.CHNS位为“1”（只在传送计数器为“0”时进行链传送），MRB.DISEL位为“0”（在指定的数据传送结束时向CPU请求中断）。
2. 必须在其他区域（ROM等）准备第1次数据传送的传送目标地址的每65536次的起始地址高8位。例如，当输入缓冲器为“20000h”~“21FFFFh”时，准备“21h”和“20h”。
3. 作为第2次数据传送，设定为用于重新设定第1次数据传送的传送目标地址的重复传送模式（传送源为重复区域），传送目标为第1次传送信息区的DAR寄存器的高8位。此时，必须将MRB.CHNE位置“0”（禁止链传送），并且将MRB.DISEL位置“0”（在指定的数据传送结束时向CPU请求中断）。当上述输入缓冲器为“20000h”~“21FFFFh”时，必须将传送计数器设定为“2”。
4. 通过中断进行65536次的第1次数据的传送。如果第1次数据传送的传送计数器变为“0”，就启动第2次数据传送。将第1次数据传送的传送源地址高8位置“21h”，第1次数据传送的传送目标地址低16位的传送计数器为“0000h”。
5. 通过中断继续进行在第1次数据传送指定的65536次的第1次数据传送。如果第1次数据传送的传送计数器变为“0”，就启动第2次数据传送。将第1次数据传送的传送源地址高8位置“20h”，第1次的数据传送的传送目标地址低16位的传送计数器变为“0000h”。
6. 无限重复上述的步骤4和步骤5。因为第2次数据传送为重复传送模式，所以不向CPU请求中断。

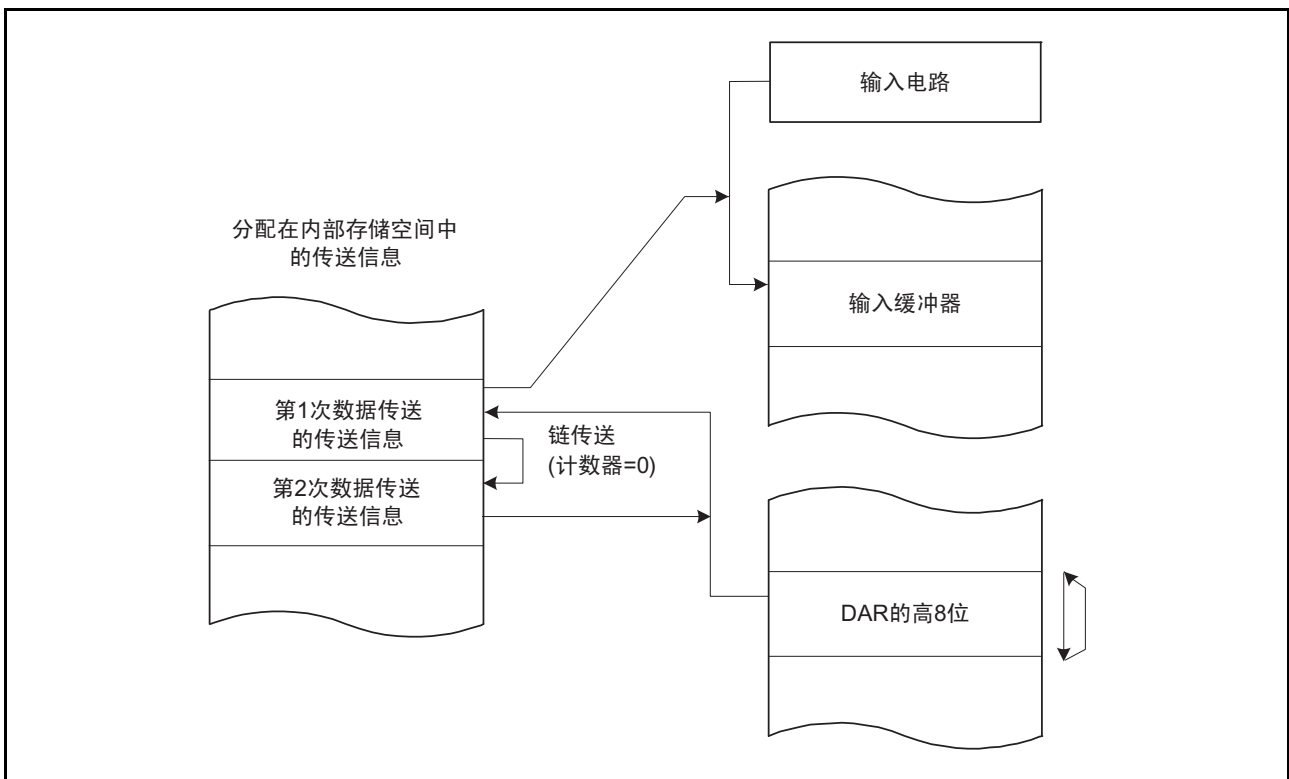


图 17.15 计数器为“0”时的链传送

17.7 中断源

在结束 DTC 指定次数的数据传送以及结束 MRB.DISEL 位为“1”（在每次 DTC 数据传送时向 CPU 请求中断）的数据传送时，通过启动 DTC 的中断源向 CPU 请求中断。这些向 CPU 请求的中断受 CPU 的 PSW.I 位（中断允许位）、PSW.IPL[3:0] 位（处理器中断优先级）和中断控制器优先级的控制。

17.8 事件链接功能

DTC 在 1 个请求的传送结束后输出事件链接请求。但是，如果传送目标为外部总线或者内部外围总线，就在写缓冲器的写操作结束时，发行事件链接请求。

17.9 低功耗功能

在使用模块停止功能以及向全模块时钟停止模式、软件待机模式或者深度软件待机模式转移时，必须在给 DTCST.DTCST 位写“0”（DTC 模块停止）后分别进行以下的设定。

(1) 模块停止功能

通过给 MSTPCRA.MSTPA28 位写“1”（转移到模块停止状态），使 DTC 的模块停止功能有效。如果在给 MSTPCRA.MSTPA28 位写 1 时正在进行 DTC 传送，就在 DTC 传送结束后转移到模块停止状态。

不能在 MSTPCRA.MSTPA28 位为“1”时存取 DTC 的寄存器。

通过给 MSTPCRA.MSTPA28 位写“0”（解除模块停止状态），解除 DTC 模块停止状态。

(2) 全模块时钟停止模式

必须按照“11. 低功耗功能”的“11.6.2.1 向全模块时钟停止模式的转移”的步骤进行设定。

如果在执行 WAIT 指令时正在进行 DTC 传送，就在 DTC 传送结束后转移到全模块时钟停止模式。

在从全模块时钟停止模式返回后，通过给 MSTPCRA.MSTPA28 位写“0”，解除 DTC 的模块停止状态。

(3) 软件待机模式和深度软件待机模式

必须按照“11. 低功耗功能”的“11.6.3.1 向软件待机模式的转移”或者“11.6.4.1 向深度软件待机模式的转移”的步骤进行设定。

如果在执行 WAIT 指令时正在进行 DTC 传送，就在 DTC 传送结束后转移到软件待机模式或者深度软件待机模式。

(4) 低功耗功能的注意事项

有关 WAIT 指令和寄存器的设定步骤，请参照“11. 低功耗功能”的“11.7.6 WAIT 指令的执行时序”。

要在从低功耗模式返回后进行 DTC 传送时，必须再次给 DTCST.DTCST 位写“1”。

当全模块时钟停止模式期间或者软件待机模式期间发生的请求不是 DTC 启动请求而是 CPU 中断请求时，必须按照“14. 中断控制器 (ICUb)”的“14.4.3 中断请求目标的选择”的设定方法，在将中断请求源转换为 CPU 后执行 WAIT 指令。

17.10 使用时的注意事项

17.10.1 传送信息起始地址 / 传送源地址 / 传送目标地址

必须给向量表中指定的传送信息起始地址指定地址 $4n$ ，否则将地址的低 2 位作为“00b”进行存取。

17.10.2 传送信息的分配

如图 17.16 所示，在将传送信息分配到存储器时，必须根据要分配区域的字节序进行分配。

例如，当用 16 位写 CRA 和 CRB 的设定数据时，在大端法的情况下，必须将 CRA 和 CRB 的设定数据分别写到低位地址 0 和低位地址 2；在小端法的情况下，必须将 CRB 和 CRA 的设定数据分别写到低位地址 0 和低位地址 2。当用 32 位写 CRA 和 CRB 的设定数据时，与字节序无关，必须在将 CRA 的设定数据分配到 32 位的 MSB 侧并且将 CRB 的设定数据分配到 LSB 侧后写到低位地址 0。

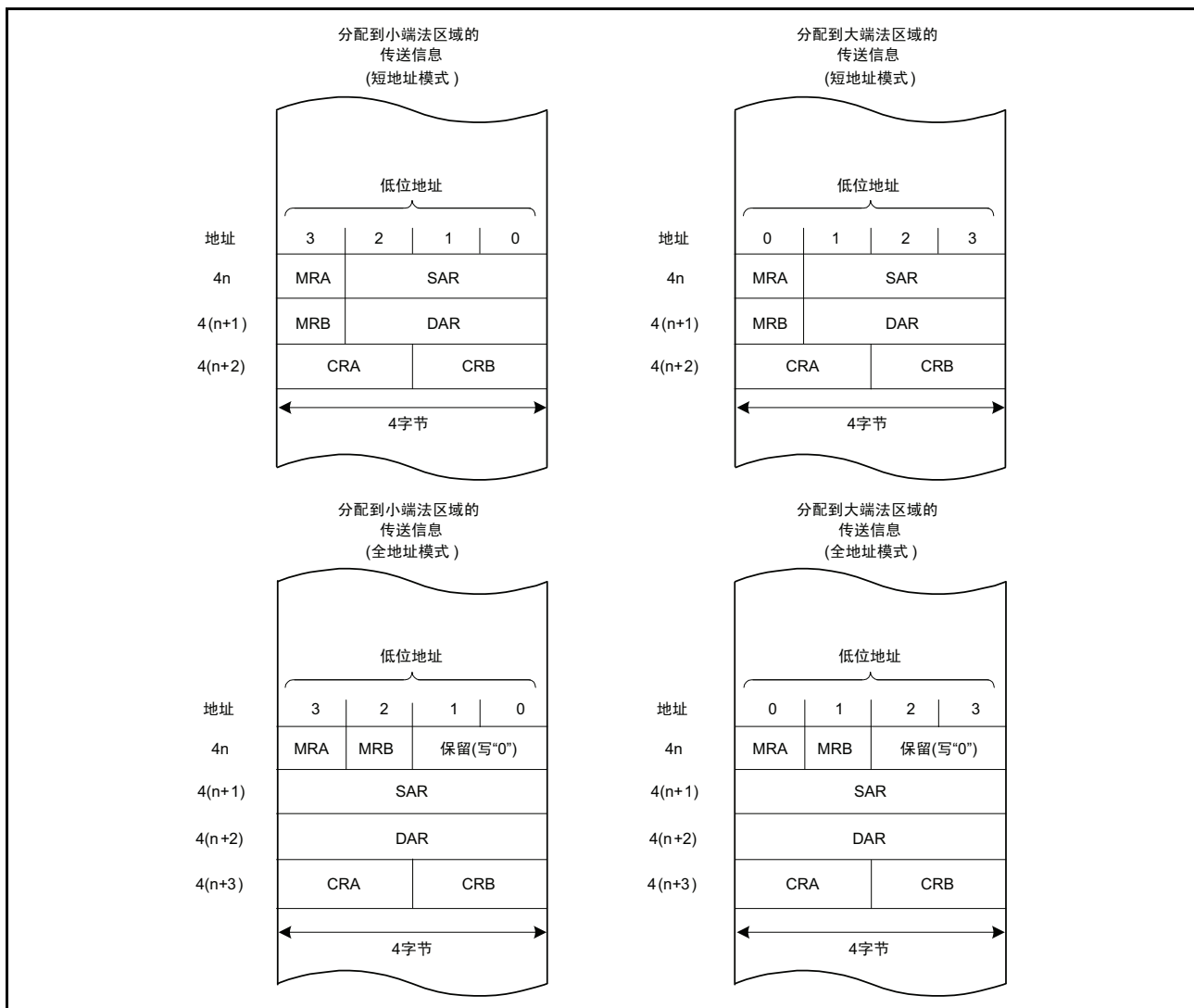


图 17.16 传送信息的分配

17.10.3 中断控制器的 DTC 启动允许寄存器 (ICU.DTCERn) 的设定

必须在 DTCST.DTCST 位为“0” (DTC 模块停止) 时设定 DTC 启动允许寄存器 (ICU.DTCERn 寄存器)。另外，不能在将与通过 ICU.DTCERn 寄存器置“1” (允许 DTC 传送) 的中断向量号相同的中断向量号设定给 DMAC 启动请求选择寄存器 (ICU.DMRSRn ($n=DMAC$ 通道号)) 后启动 DMAC。有关 ICU.DTCERn 寄存器和 ICU.DMRSRn 寄存器的详细内容，请参照“14. 中断控制器 (ICUb)”。

18. 事件链接控制器 (ELC)

18.1 概要

事件链接控制器 (ELC) 能在模块间相互连接 (链接) 各外围模块输出的事件。通过事件链接就能不经 CPU 而直接在模块间进行联合运行。

ELC 的规格和框图分别如表 18.1 和图 18.1 所示。

表 18.1 ELC 的规格

项目	内容
事件链接功能	<ul style="list-style-type: none"> 能将 59 种事件信号直接链接到模块 定时器类的模块能选择输入事件时的运行 能进行端口 B 和端口 E 的事件链接运行 单端口 (注 1): 能给指定的 1 位端口设定事件链接运行。 端口组 (注 1): 在 8 位端口中, 能在将指定的多位组成组后设定事件链接运行。
低功耗功能	能设定为模块停止状态

注 1. 被指定为输入的单端口和端口组根据连接的信号值变化产生事件。

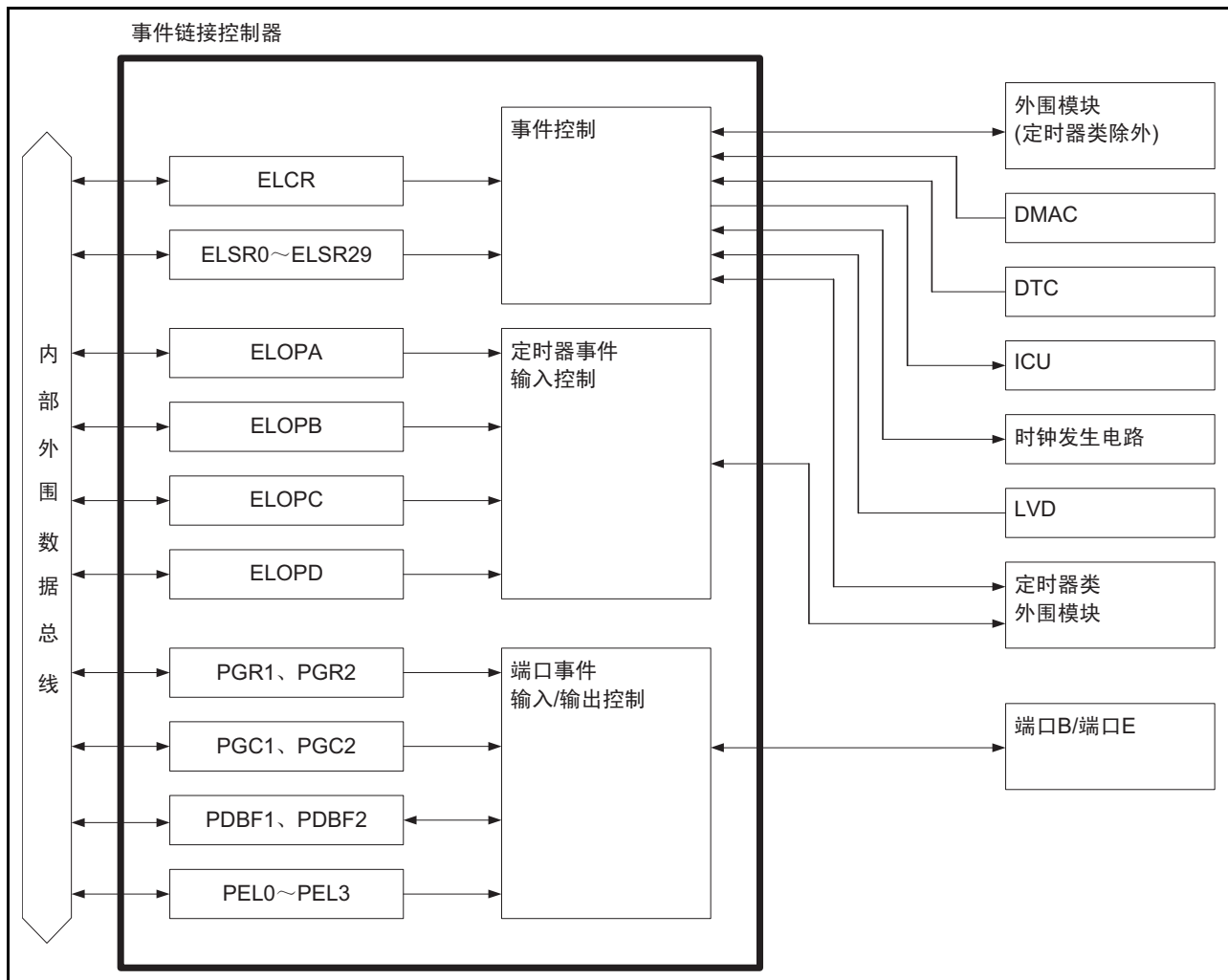


图 18.1 ELC 的框图

18.2 寄存器说明

18.2.1 事件链接控制寄存器 (ELCR)

地址 0008 B100h

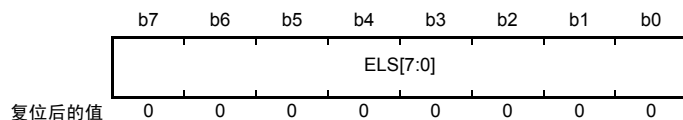
	b7	b6	b5	b4	b3	b2	b1	b0
	ELCON	—	—	—	—	—	—	—
复位后的值	0	1	1	1	1	1	1	1

位	符号	位名	功能	R/W
b6-b0	—	保留位	读写值都为“1”。	R/W
b7	ELCON	全部事件链接允许位	0: 全部事件链接无效 1: 全部事件链接有效	R/W

ELCR 寄存器是控制事件链接控制器 (ELC) 运行的寄存器。

18.2.2 事件链接设定寄存器 n (ELSRn) (n=1 ~ 4、7、10、12、15、16、18 ~ 29)

地址 ELSR1: 0008 B102h、ELSR2: 0008 B103h、ELSR3: 0008 B104h、ELSR4: 0008 B105h、ELSR7: 0008 B108h、
 ELSR10: 0008 B10Bh、ELSR12: 0008 B10Dh、ELSR15: 0008 B110h、ELSR16: 0008 B111h、
 ELSR18: 0008 B113h、ELSR19: 0008 B114h、ELSR20: 0008 B115h、ELSR21: 0008 B116h、
 ELSR22: 0008 B117h、ELSR23: 0008 B118h、ELSR24: 0008 B119h、ELSR25: 0008 B11Ah、
 ELSR26: 0008 B11Bh、ELSR27: 0008 B11Ch、ELSR28: 0008 B11Dh、ELSR29: 0008 B11Eh



位	符号	位名	功能	R/W
b7-b0	ELS[7:0]	事件链接选择位	b7 b0 00000000: 事件链接功能停止 00000001 ~ 01101001: 指定要链接的事件信号号 上述以外: 不能设定	R/W

ELSRn 寄存器是给各外围模块指定要链接的事件信号的寄存器。ELSRn 寄存器和外围模块的对应如表 18.2 所示。给 ELSRn.ELS[7:0] 位设定的事件信号号和事件信号名的对应如所示。

表 18.2 ELSRn 寄存器和外围功能的对应

寄存器名	外围功能 (模块)
ELSR1	MTU1
ELSR2	MTU2
ELSR3	MTU3
ELSR4	MTU4
ELSR7	CMT1
ELSR10	TMR 0
ELSR12	TMR 2
ELSR15	12 位 A/D 转换器
ELSR16	DA0
ELSR18	中断 1
ELSR19	中断 2
ELSR20	输出端口组 1
ELSR21	输出端口组 2
ELSR22	输入端口组 1
ELSR23	输入端口组 2
ELSR24	单端口 0
ELSR25	单端口 1
ELSR26	单端口 2
ELSR27	单端口 3
ELSR28	将时钟源转换为 LOCO
ELSR29	POE

表 18.3 给 ELSRn.ELS[7:0] 位设定的事件信号号和事件信号名的对应 (1/2)

ELS[7:0] 位的值	ELSRn 寄存器设定的事件信号
00001000 (08h)	MTU1 的比较匹配 1A 信号
00001001 (09h)	MTU1 的比较匹配 1B 信号
00001010 (0Ah)	MTU1 的上溢信号
00001011 (0Bh)	MTU1 的下溢信号
00001100 (0Ch)	MTU2 的比较匹配 2A 信号
00001101 (0Dh)	MTU2 的比较匹配 2B 信号
00001110 (0Eh)	MTU2 的上溢信号
00001111 (0Fh)	MTU2 的下溢信号
00010000 (10h)	MTU3 的比较匹配 3A 信号
00010001 (11h)	MTU3 的比较匹配 3B 信号
00010010 (12h)	MTU3 的比较匹配 3C 信号
00010011 (13h)	MTU3 的比较匹配 3D 信号
00010100 (14h)	MTU3 的上溢信号
00010101 (15h)	MTU4 的比较匹配 4A 信号
00010110 (16h)	MTU4 的比较匹配 4B 信号
00010111 (17h)	MTU4 的比较匹配 4C 信号
00011000 (18h)	MTU4 的比较匹配 4D 信号
00011001 (19h)	MTU4 的上溢信号
00011010 (1Ah)	MTU4 的下溢信号
00011111 (1Fh)	CMT1 的比较匹配 1 信号
00100010 (22h)	TMR0 的比较匹配 A0 信号
00100011 (23h)	TMR0 的比较匹配 B0 信号
00100100 (24h)	TMR0 的上溢信号
00101000 (28h)	TMR2 的比较匹配 A2 信号
00101001 (29h)	TMR2 的比较匹配 B2 信号
00101010 (2Ah)	TMR2 的上溢信号
00101110 (2Eh)	RTC 的周期信号
00110001 (31h)	IWDT 的下溢、刷新错误信号
00111010 (3Ah)	SCI5 的错误 (接收错误、错误信号检测) 信号
00111011 (3Bh)	SCI5 的接收数据满信号
00111100 (3Ch)	SCI5 的发送数据空信号
00111101 (3Dh)	SCI5 的发送结束信号
01001110 (4Eh)	RIIC0 的通信错误、事件发生信号
01001111 (4Fh)	RIIC0 的接收数据满信号
01010000 (50h)	RIIC0 的发送数据空信号
01010001 (51h)	RIIC0 发送结束信号
01010010 (52h)	RSPI0 的错误 (模式故障、溢出、奇偶校验错误) 信号
01010011 (53h)	RSPI0 的空闲信号
01010100 (54h)	RSPI0 的接收数据满信号
01010101 (55h)	RSPI0 的发送数据空信号
01010110 (56h)	RSPI0 的发送结束信号

表 18.3 给 ELSRn.ELS[7:0] 位设定的事件信号号和事件信号名的对应 (2/2)

ELS[7:0] 位的值	ELSRn 寄存器设定的事件信号
01011000 (58h)	12 位 A/D 转换器的 A/D 转换结束信号
01011001 (59h)	比较器 B0 的比较结果变化信号
01011010 (5Ah)	比较器 B0、B1 的通用比较结果变化信号
01011011 (5Bh)	LVD1 的电压检测信号
01011100 (5Ch)	LVD2 的电压检测信号
01011101 (5Dh)	DMAC0 的传送结束信号
01011110 (5Eh)	DMAC1 的传送结束信号
01011111 (5Fh)	DMAC2 的传送结束信号
01100000 (60h)	DMAC3 的传送结束信号
01100001 (61h)	DTC 的传送结束信号
01100010 (62h)	时钟发生电路的振荡停止检测信号
01100011 (63h)	输入端口组 1 的输入边沿检测信号
01100100 (64h)	输入端口组 2 的输入边沿检测信号
01100101 (65h)	单输入端口 0 的输入边沿检测信号
01100110 (66h)	单输入端口 1 的输入边沿检测信号
01100111 (67h)	单输入端口 2 的输入边沿检测信号
01101000 (68h)	单输入端口 3 的输入边沿检测信号
01101001 (69h)	软件事件信号
上述以外: 不能设定	

18.2.3 事件链接选项设定寄存器 A (ELOPA)

地址 0008 B11Fh



位	符号	位名	功能	R/W
b1-b0	—	保留位	读写值都为“1”。	R/W
b3-b2	MTU1MD[1:0]	MTU1 运行选择位	b3 b2 0 0: 开始计数 0 1: 重新开始计数 1 0: 输入捕捉 (注 1) 1 1: 事件无效	R/W
b5-b4	MTU2MD[1:0]	MTU2 运行选择位	b5 b4 0 0: 开始计数 0 1: 重新开始计数 1 0: 输入捕捉 (注 2) 1 1: 事件无效	R/W
b7-b6	MTU3MD[1:0]	MTU3 运行选择位	b7 b6 0 0: 开始计数 0 1: 重新开始计数 1 0: 输入捕捉 (注 3) 1 1: 事件无效	R/W

注 1. MTU1.TCNT 计数器的值被捕捉到 MTU1.TGRA 寄存器。

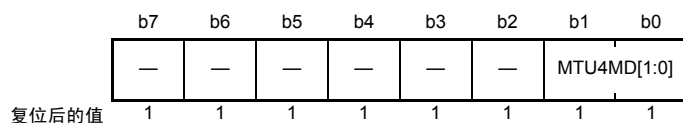
注 2. MTU2.TCNT 计数器的值被捕捉到 MTU2.TGRA 寄存器。

注 3. MTU3.TCNT 计数器的值被捕捉到 MTU3.TGRA 寄存器。

ELOPA 寄存器设定输入 MTU 的 MTU1 ~ MTU3 的事件时的运行。在不使用 ELC 功能时，必须设定为事件无效。

18.2.4 事件链接选项设定寄存器 B (ELOPB)

地址 0008 B120h



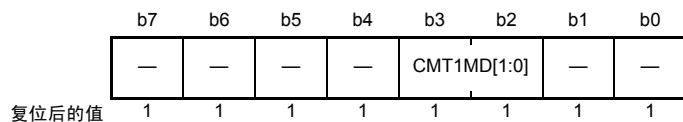
位	符号	位名	功能	R/W
b1-b0	MTU4MD[1:0]	MTU4 运行选择位	b1 b0 0 0: 开始计数 0 1: 重新开始计数 1 0: 输入捕捉 (注1) 1 1: 事件无效	R/W
b7-b2	—	保留位	读写值都为“1”。	R/W

注 1. MTU4.TCNT 计数器的值被捕捉到 MTU4.TGRA 寄存器。

ELOPB 寄存器设定输入 MTU 的 MTU4 的事件时的运行。在不使用 ELC 功能时，必须设定为事件无效。

18.2.5 事件链接选项设定寄存器 C (ELOPC)

地址 0008 B121h



位	符号	位名	功能	R/W
b1-b0	—	保留位	读写值都为“1”。	R/W
b3-b2	CMT1MD[1:0]	CMT1 运行选择位	b3 b2 0 0: 开始计数 0 1: 重新开始计数 1 0: 事件计数器 1 1: 事件无效	R/W
b7-b4	—	保留位	读写值都为“1”。	R/W

ELOPC 寄存器设定输入 CMT 的 CMT1 的事件时的运行。在不使用 ELC 功能时，必须设定为事件无效。

18.2.6 事件链接选项设定寄存器 D (ELOPD)

地址 0008 B122h

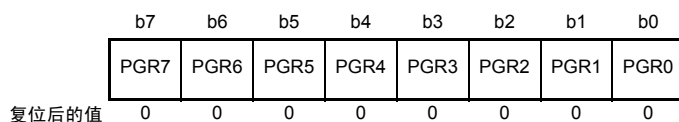


位	符号	位名	功能	R/W
b1-b0	TMR0MD[1:0]	TMR0 运行选择位	b1 b0 0 0: 开始计数 0 1: 重新开始计数 1 0: 事件计数器 1 1: 事件无效	R/W
b3-b2	—	保留位	读写值都为“1”。	R/W
b5-b4	TMR2MD[1:0]	TMR2 运行选择位	b5 b4 0 0: 开始计数 0 1: 重新开始计数 1 0: 事件计数器 1 1: 事件无效	R/W
b7-b6	—	保留位	读写值都为“1”。	R/W

ELOPD 寄存器设定输入 TMR 的 TMR0、TMR2 的事件时的运行。在不使用 ELC 功能时，必须设定为事件无效。

18.2.7 端口组指定寄存器 n (PGRn) (n=1、2)

地址 PGR1: 0008 B123h、PGR2: 0008 B124h

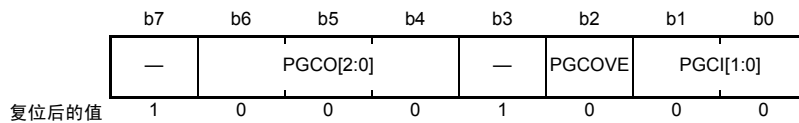


位	符号	位名	功能	R/W
b0	PGR0	端口组指定 0 位	0: 不指定端口组 1: 指定端口组	R/W
b1	PGR1	端口组指定 1 位		R/W
b2	PGR2	端口组指定 2 位		R/W
b3	PGR3	端口组指定 3 位		R/W
b4	PGR4	端口组指定 4 位		R/W
b5	PGR5	端口组指定 5 位		R/W
b6	PGR6	端口组指定 6 位		R/W
b7	PGR7	端口组指定 7 位		R/W

PGRn 寄存器是设定输入 / 输出端口组的寄存器。对 8 位端口中的各个端口 (1 位) 指定组。能将 1 ~ 8 位的任意端口指定到同一组。PGRn 寄存器和端口的对应如表 18.4 所示。

18.2.8 端口组控制寄存器 n (PGCn) (n=1、2)

地址 PGC1: 0008 B125h、PGC2: 0008 B126h



位	符号	位名	功能	R/W
b1-b0	PGCI[1:0]	事件输出边沿选择位	b1 b0 0 0: 在检测到外部输入信号的上升沿后发生事件 0 1: 在检测到外部输入信号的下降沿后发生事件 1 x: 在检测到外部输入信号的双边沿 (上升沿 / 下降沿) 后发生事件	R/W
b2	PGCOVE	PDBF 盖写指定位	0: 盖写 PDBF 寄存器无效 1: 盖写 PDBF 寄存器有效	R/W
b3	—	保留位	读写值都为“1”。	R/W
b6-b4	PGCO[2:0]	端口组运行选择位	b6 b5 b4 0 0 0: 在输入事件时输出 Low 电平 0 0 1: 在输入事件时输出 High 电平 0 1 0: 在输入事件时交替 (取反) 输出 0 1 1: 在输入事件时输出缓冲器值 1 x x: 在输入事件时在组内进行位循环输出 (MSB → LSB 循环)	R/W
b7	—	保留位	读写值都为“1”。	R/W

x: Don't care

PGCn 寄存器给设定为输出的端口组指定从输入事件信号时的端口向外部输出信号的输出形式。另外, 给设定为输入的端口组指定盖写 PDBF 寄存器的有效 / 无效以及设定发生事件的条件 (外部输入信号的变化)。

PGCn 寄存器和端口的对应如表 18.4 所示。

18.2.9 端口缓冲寄存器 n (PDBFn) (n=1、2)

地址 PDBF1: 0008 B127h、PDBF2: 0008 B128h

b7	b6	b5	b4	b3	b2	b1	b0
PDBF7	PDBF6	PDBF5	PDBF4	PDBF3	PDBF2	PDBF1	PDBF0
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	PDBF0	端口缓冲 0 位	在 PODR 和 PDBF 之间，通过输入事件来传送数据。 设定为输入端口组的位的 CPU 写操作无效。 详细内容请参照“18.3 运行说明”。	R/W
b1	PDBF1	端口缓冲 1 位		R/W
b2	PDBF2	端口缓冲 2 位		R/W
b3	PDBF3	端口缓冲 3 位		R/W
b4	PDBF4	端口缓冲 4 位		R/W
b5	PDBF5	端口缓冲 5 位		R/W
b6	PDBF6	端口缓冲 6 位		R/W
b7	PDBF7	端口缓冲 7 位		R/W

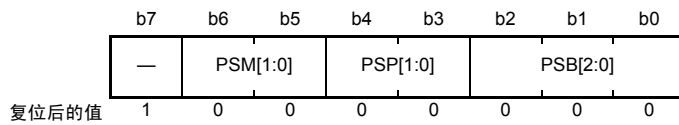
PDBFn 寄存器是与 PGRn 寄存器组成对的 8 位寄存器。有关 PDBFn 寄存器的运行，请参照“18.3 运行说明”。端口组的相关寄存器和端口号的对应如表 18.4 所示。

表 18.4 端口组的相关寄存器和端口号的对应

端口组指定寄存器 (PGR)	端口组控制寄存器 (PGC)	端口缓冲寄存器 (PDBF)	端口号
PGR1 寄存器	PGC1 寄存器	PDBF1 寄存器	端口 B
PGR2 寄存器	PGC2 寄存器	PDBF2 寄存器	端口 E

18.2.10 事件链接端口指定寄存器 n (PELn) (n=0 ~ 3)

地址 PEL0: 0008 B129h、PEL1: 0008 B12Ah、PEL2: 0008 B12Bh、PEL3: 0008 B12Ch



位	符号	位名	功能	R/W
b2-b0	PSB[2:0]	位号指定位	指定 8 位端口的位号	R/W
b4-b3	PSP[1:0]	端口号指定位	b4 b3 0 0: 设定无效 0 1: 端口 B (对应 PGR1 寄存器) 1 0: 端口 E (对应 PGR2 寄存器) 1 1: 不能设定	R/W
b6-b5	PSM[1:0]	事件链接指定位	<ul style="list-style-type: none"> • 当设定为端口输出时: 指定端口输出数据 b6 b5 0 0: 在输入事件时输出 Low 电平 0 1: 在输入事件时输出 High 电平 1 x: 在输入事件时交替 (取反) 输出 • 当设定为端口输入时: 选择事件输出边沿 b6 b5 0 0: 在检测到上升沿后输出事件 0 1: 在检测到下降沿后输出事件 1 x: 在检测到双边沿 (上升沿 / 下降沿) 后输出事件 	R/W
b7	—	保留位	读写值都为“1”。	R/W

x: Don't care

PELn 寄存器指定链接事件的 1 位端口 (以下称单端口), 以及设定输入事件时的运行和发生事件的条件。在 RX210 群的端口 B 和端口 E (8 位端口) 中, 能将其中的任意位设定为单端口 (最多 4 个单端口)。

18.2.11 事件链接的软件事件发生寄存器 (ELSEGR)

地址 0008 B12Dh

	b7	b6	b5	b4	b3	b2	b1	b0
	WI	WE	—	—	—	—	—	SEG
复位后的值	1	0	1	1	1	1	1	0

位	符号	位名	功能	R/W
b0	SEG	软件事件发生位	0: 正常运行 1: 发生软件事件	W
b5-b1	—	保留位	读写值都为“1”。	R/W
b6	WE	SEG 位的写操作允许位	0: 禁止写 SEG 位 1: 允许写 SEG 位	R/W
b7	WI	ELSEGR 寄存器的写操作禁止位	0: 允许写 ELSEGR 寄存器 1: 禁止写 ELSEGR 寄存器	W

必须使用 MOV 指令写 ELSEGR 寄存器。

SEG 位 (软件事件发生位)

如果在 WE 位为“1”的状态下给此位写“1”，就发生软件事件。
此位的读取值为“0”，即使写“1”，也不保存数据。

WE 位 (SEG 位的写操作允许位)

只有在 WE 位为“1”时才能写 SEG 位。
[为“1”的条件]
给 WI 位写“0”并且给 WE 位写“1”时
[为“0”的条件]
给 WI 位和 WE 位都写“0”时

WI 位 (ELSEGR 寄存器的写操作禁止位)

只有在 WI 位的写入值为“0”时才能写 ELSEGR 寄存器。
读取值为“1”。

18.3 运行说明

18.3.1 中断处理和事件链接的关系

在 RX210 群的内置模块中有中断请求状态标志和控制允许或者禁止这些中断的中断允许位。如果在各模块中产生中断请求，中断请求状态标志就被置位，并且在允许该中断请求时向 CPU 请求中断。

与此相对，ELC 将在各模块中产生的中断请求（以下称事件）用作直接启动其他模块的事件信号。即使中断允许控制没有允许中断也能使用事件信号。中断处理和 ELC 的关系如图 18.2 所示。

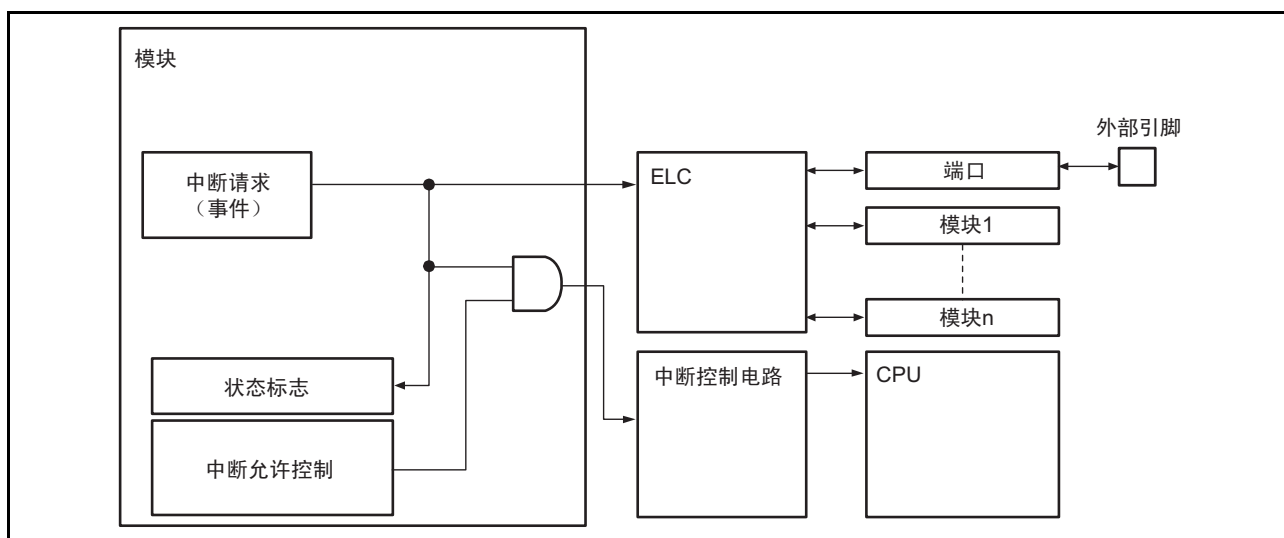


图 18.2 中断处理和 ELC 的关系

18.3.2 事件链接

在通过给 ELSRn 寄存器设定事件发生源来发生设定的事件时，就链接（启动）对应的模块。1 个模块只能链接 1 种事件。必须在启动模块的初始设定结束后通过 ELC 启动模块。输入事件时的各模块运行一览表如表 18.5 所示。

表 18.5 输入事件时的各模块运行一览表

模块	输入事件时的运行		
MTU CMT TMR	通过设定 ELOPA ~ ELOPD 寄存器进行以下运行。 <ul style="list-style-type: none"> 通过输入事件信号开始计数 通过输入事件信号重新开始计数 对输入的事件数进行计数（CMT、TMR） 通过输入事件进行捕捉运行（MTU） 		
A/D 转换器	通过输入事件信号开始 A/D 转换		
D/A 转换器	通过输入事件信号开始 D/A 转换		
输出端口	根据输入的事件信号，PODR 寄存器（输出端口寄存器）的值发生变化。 （外部引脚的信号输出值变化）	端口组	通过设定运行进行以下运行 <ul style="list-style-type: none"> PODR 寄存器的值变为指定值 将 PDBFn 寄存器的值传送到 PODR 寄存器 循环输出
		单端口	PODR 寄存器的值变为指定值
输入端口	输入引脚的信号值变化	端口组	发生事件
		单端口	
	输入事件时	端口组	将外部引脚的信号值传送到 PDBFn 寄存器
		单端口	不能链接事件
时钟振荡器	将时钟源转换为低速内部振荡器		
中断控制	向 CPU 请求中断、开始 DMAC 数据传送、开始 DTC 数据传送		
POE	MTU 互补 PWM 输出引脚和 MTU0 的输出引脚为高阻抗状态		

18.3.3 输入定时器类外围功能的事件时的运行

通过 ELOPA ~ ELOPD 寄存器设定输入事件时的运行。

(1) 开始计数运行

通过输入事件开始定时器的计数，并且各定时器的控制寄存器的计数开始位（注1）被置“1”。在计数开始位为“1”的状态下输入的事件无效。

(2) 重新开始计数运行

通过输入事件对定时器的计数器（注1）进行初始化。因为保持各定时器的控制寄存器的计数开始位（注1），所以在计数开始位为“1”的状态下输入事件时，重新开始计数。

(3) 事件计数器运行

选择输入事件作为定时器的时钟源，因此定时器运行。

(4) 输入捕捉运行

通过输入事件进行捕捉运行。

注 1. 请参照各定时器章节中的位说明。

18.3.4 输入 A/D 转换器和 D/A 转换器的事件时的运行

在 ADCSR.ADST 位和 DACR.DAE 位（注1）被置“1”时开始 A/D 转换或者 D/A 转换。

注 1. 请参照 A/D 转换器和 D/A 转换器章节中的位说明。

18.3.5 端口的事件输入运行和事件发生运行

能将端口运行设定为事件输入运行和事件发生运行。

(1) 单端口和端口组

端口的事件链接有 2 种，分别是对 8 位端口中的任意 1 位端口进行事件链接（单端口的事件链接）和对 8 位端口中的多个任意位组成的组进行事件链接（端口组的事件链接）。

单端口的设定是指通过 PEL0 ~ PEL3 寄存器指定能进行事件链接的端口（注1）中的任意位。端口组的设定是指通过 PGCn 寄存器指定能进行事件链接的端口（注1）中的任意位（至少 1 位）。组的指定是指能在同一端口中能分别设定 1 个输入端口组和 1 个输出端口组。

在将对应的位同时指定为单端口和端口组时，如果为输入端口，单端口和端口组的功能就都有效，如果为输出端口，只有端口组的功能有效。

通过 PDR 寄存器设定端口的输入和输出。

注 1. 是端口 B 和端口 E。

(2) 输入单端口的事件发生运行

被设定为输入的单端口在连接该端口的外部引脚的信号值发生变化时产生事件。通过 PEL0 ~ PEL3 寄存器设定发生事件的条件。单端口的事件链接运行如图 18.3 所示。

(3) 输出单端口的事件输入运行

如果向被设定为输出的单端口输入事件，该端口的 PODR 寄存器的值就发生变化。通过 PEL0 ~ PEL3 寄存器设定 PODR 寄存器值的变化。由此连接该端口的外部引脚的信号值也发生变化。单端口的事件链接运行如图 18.3 所示。

(4) 输入端口组的事件输入运行和事件发生运行

被设定为输入的端口组在连接该端口的任意外部引脚的信号值发生变化时产生事件。通过 PGCn 寄存器设定发生事件的条件。另外，在向输入端口组输入事件时，输入事件时的外部引脚的信号值就被传送到 PDBFn 寄存器。只有指定为输入端口组的位才能被传送。输入端口组的事件链接运行如图 18.4 所示。

(5) 输出端口组的事件输入

如果向输出端口组输入事件，PODR 寄存器的值就变为 PGCn 寄存器指定的值。输出端口组的事件链接运行如图 18.5 所示。

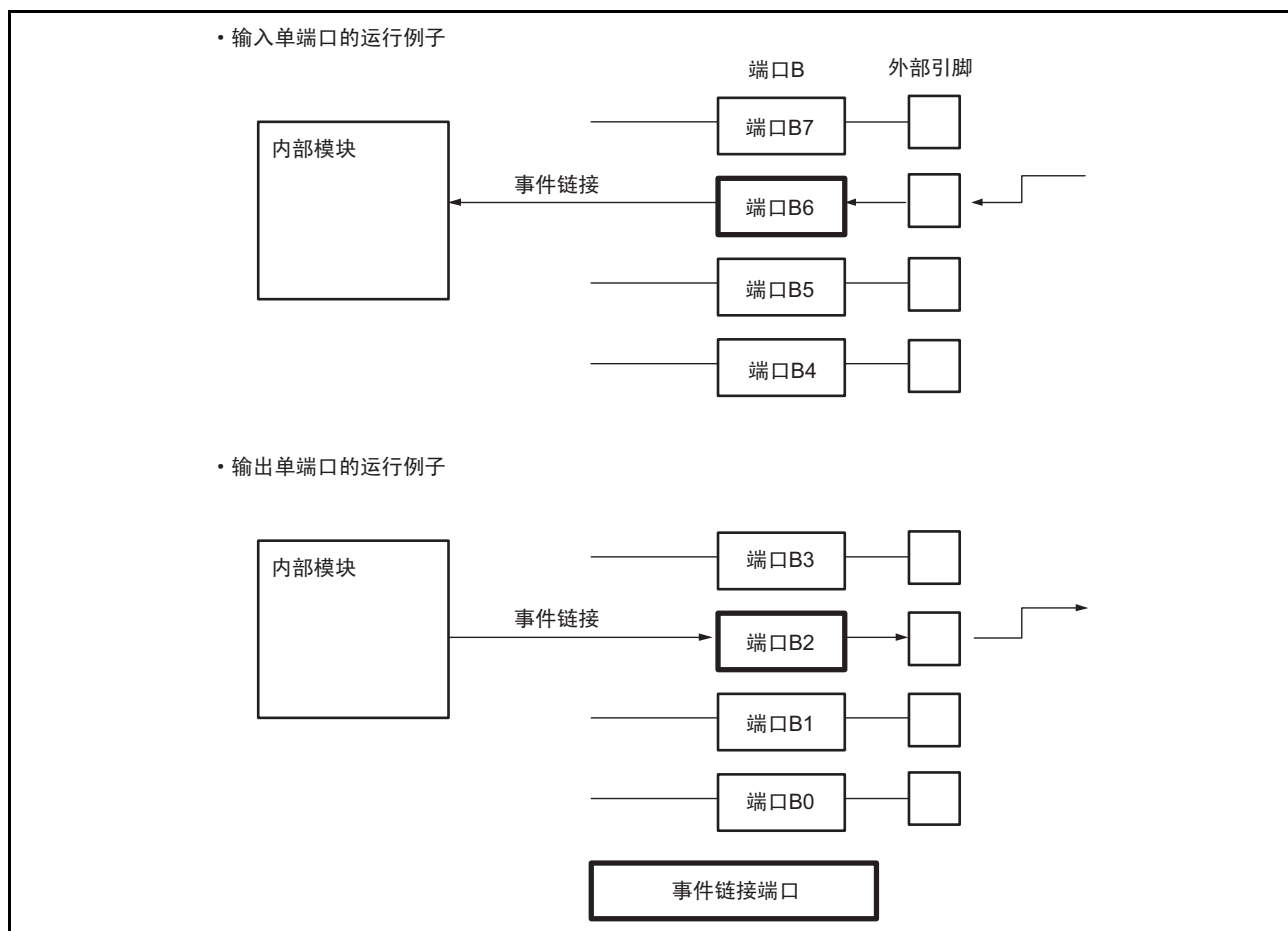


图 18.3 单端口的事件链接运行

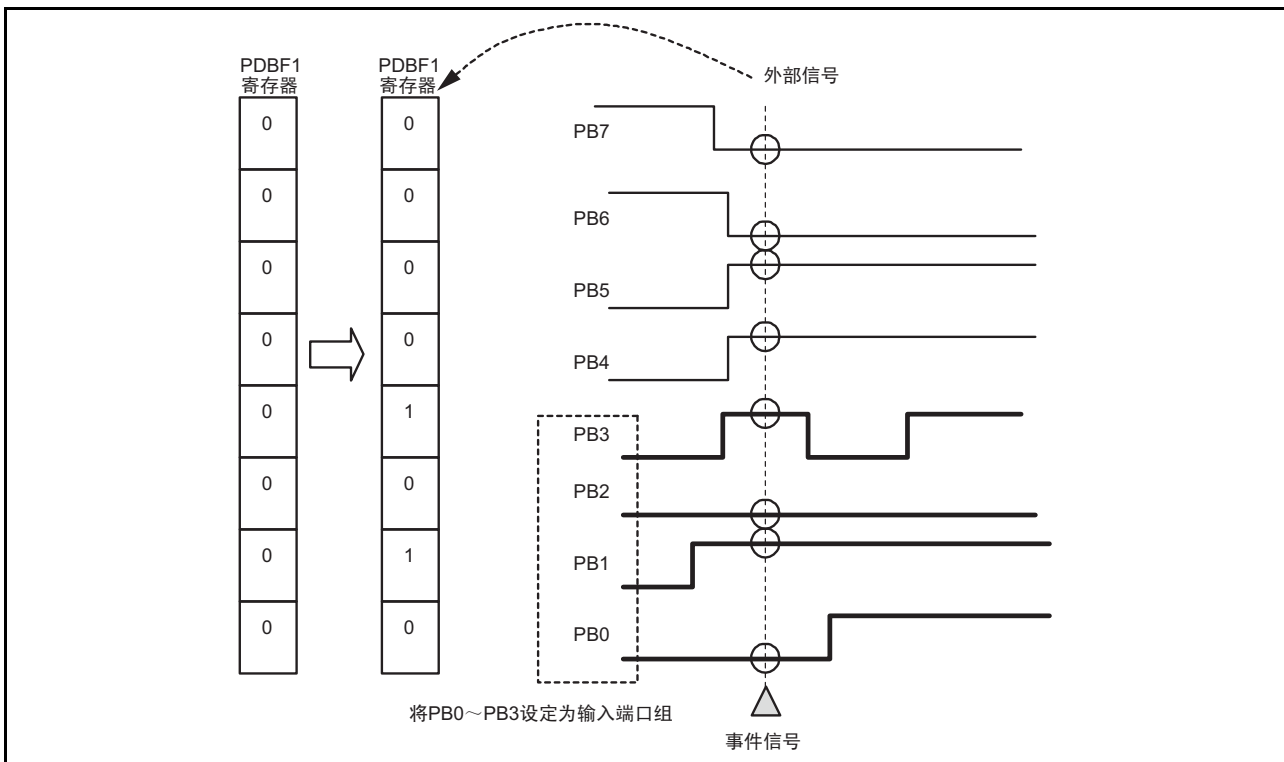


图 18.4 输入端口组的事件链接运行

(6) 端口缓冲寄存器的运行

(a) 输入端口组

如果向输入端口组输入事件，设定为输入端口组的位的外部引脚的信号值就被传送到 PDBFn 寄存器。在此状态下，如果再次向输入端口组输入事件，就通过设定 PGCn.PGCOVE 位进行以下运行。

- 当 PGCn.PGCOVE 位为“0”（盖写无效）时
如果通过 CPU 读取（包括 DTC 传送）到上次输入事件时传送到 PDBFn 寄存器的值，外部引脚的信号值就被传送到 PDBFn 寄存器。如果没有读取到上次输入事件时传送到 PDBFn 寄存器的值，外部引脚的信号值就不被传送到 PDBFn 寄存器，输入的事件无效。
- 当 PGCn.PGCOVE 位为“1”（盖写有效）时
如果向输入端口组输入事件，外部引脚的信号值就被传送到 PDBFn 寄存器。

(b) 输出端口组

当输出端口组设定为“输出 PDBFn 寄存器的值”时，如果向输出端口组输入事件，PDBFn 寄存器的值就被传送到 PODR 寄存器。此时，只有被设定为输出端口组的位被传送到 PODR 寄存器。

当输出端口组设定为“在组内进行位循环输出（PGCn.PGCO[2:0] 位 = 1xx）”时，将数据从 PDBFn 寄存器传送到 PODR 寄存器后，PODR 寄存器的值就在该组内按 MSB → LSB 进行循环。必须将输出到端口的初始值设定到 PDBFn 寄存器。

上述 (a) 和 (b) 的事件链接运行如图 18.5 和图 18.6 所示。

(7) 通过 CPU 写 PODR 寄存器和 PDBF 寄存器的限制事项

当 ELCR.ELCON 位为 “1” 时，以下寄存器的写操作无效。

- 当指定为输入端口组并且设定为事件链接时，设定为 PDBFn 寄存器的输入端口组的位的写操作（通过 CPU 执行）就无效。
- 当指定为输出端口组时，通过 CPU 执行的 PODR 寄存器的对应位的写操作就无效。
- 当指定为输出单端口时，如果对该端口进行事件链接的设定（ELSRn 寄存器的设定），通过 CPU 执行的 PODR 寄存器的对应位的写操作就无效。

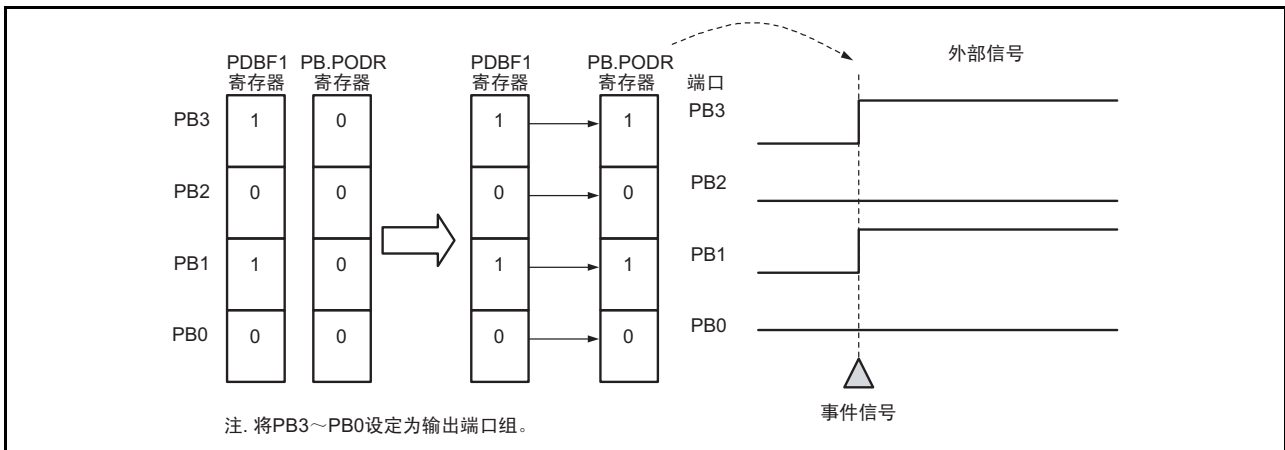


图 18.5 输出端口组的事件链接运行

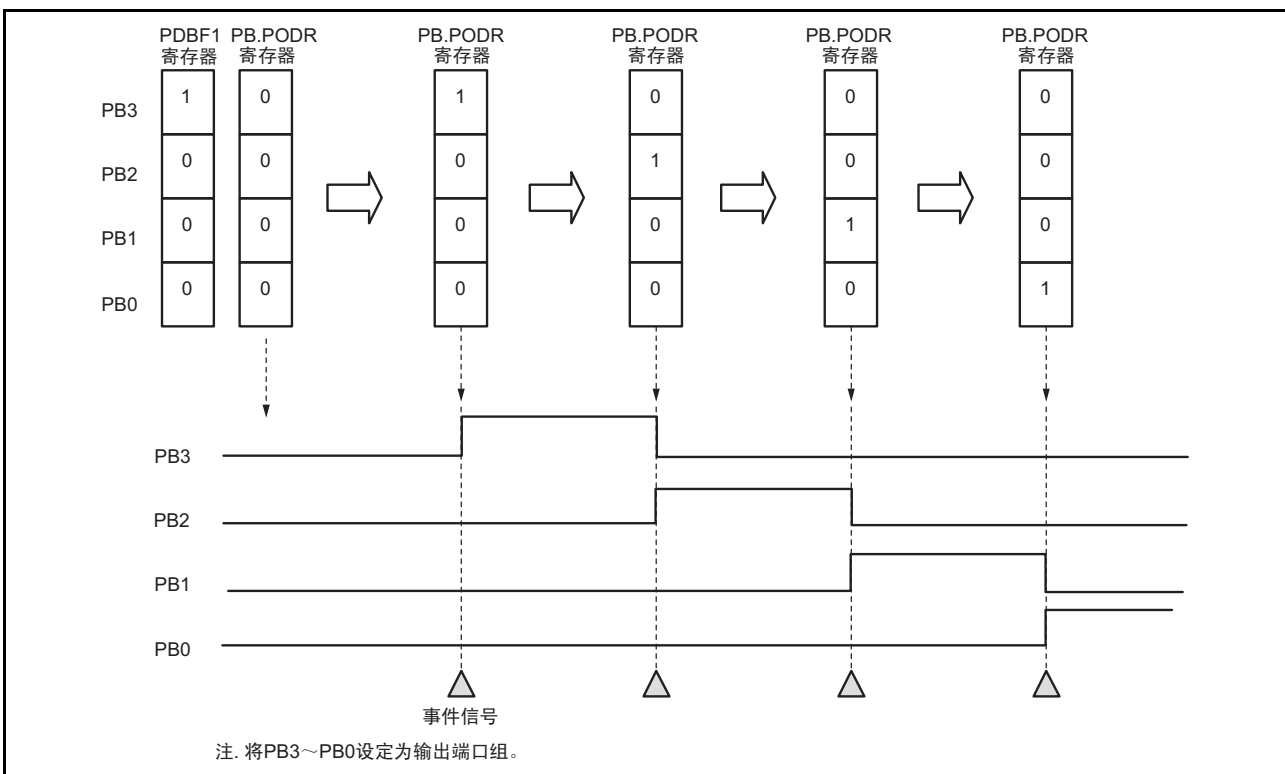


图 18.6 输出端口组的位循环运行

18.3.6 事件链接的运行设定步骤

事件链接运行的设定步骤如下所示。

1. 对事件链接的模块进行初始设定。
2. 在对端口进行事件链接的设定时，设定对应端口的以下寄存器。
PODR 寄存器：设定输出端口的初始值。
PDR 寄存器：设定端口的输入或者输出。
PGRn 寄存器：当作为端口组运行时，设定组成组的对象端口（以位为单位）。
PGCn 寄存器：设定作为端口组运行时的运行。
PELn 寄存器：设定作为单端口运行时的对象端口、输入事件以及发生事件的条件。
3. 给事件链接模块的 ELSRn 寄存器设定链接的事件信号号。
4. 当事件链接模块为定时器时，根据需要设定对应的 ELOPA ~ ELOPD 寄存器。
5. 将 ELCR.ELCON 位置“1”，允许设定了事件链接的全部模块进行事件链接运行。
6. 对事件输出源的模块进行初始设定，并且使其启动。通过从模块输出的事件，使事件链接目标的模块开始规定的运行。
7. 想要停止个别模块的事件链接运行时，必须将对应的 ELSRn.ELS[7:0] 位设定为“0000000b”。另外，能通过将 ELCR.ELCON 位置“0”，停止全部模块的事件链接运行。

注 如果使用 RTC 的事件链接输出功能，就必须在进行 RTC 的设定（初始化、时间设定等）后进行 ELC 的设定。如果在 ELC 的设定后进行 RTC 的设定，就有可能输出意想不到事件。

18.4 使用时的注意事项

18.4.1 有关 ELSR18 寄存器和 ELSR19 寄存器的设定

事件信号必须从“01100011(63h)” ~ “01101001(69)”中指定。上述以外的值，禁止设定。

18.4.2 有关输出端口组的位循环运行的设定

在输出端口组的位循环运行模式中更改 PDBFn 寄存器的值时，必须重新设定 ELSRn 寄存器。用于位循环运行的事件必须空出 1 个 PCLK 周期的事件发生间隔，否则无法正常运行。

18.4.3 使用 DMAC/DTC 传送结束的事件链接时的注意事项

在使用 DMAC/DTC 传送结束的事件链接时，不能将该 DMAC/DTC 传送目标和事件链接启动目标设定到同一外围模块。在外围模块的 DMAC/DTC 传送结束前外围模块有可能启动。

18.4.4 有关时钟的设定

为了让各外围模块输出的事件在模块间相互链接并且运行，除了设定 ELC 以外，ELC 和对象模块还必须处于可运行状态。如果对象模块处于模块停止状态或者处于低功耗状态下的模块停止状态模式（全部模块时钟停止模式、软件待机模式、深度软件待机模式），就无法运行。

18.4.5 模块停止功能的设定

能通过模块停止控制寄存器 B (MSTPCRB) 设定禁止或者允许事件链接控制器的运行。初始值为停止事件链接控制器的运行。能通过解除模块停止状态存取寄存器。详细内容请参照“11. 低功耗功能”。

19. I/O 端口

19.1 概要

RX210 群的 I/O 端口用作可编程输入/输出端口、外围功能的输入/输出、中断输入引脚或者总线控制引脚。

各端口兼用外围模块的输入/输出引脚和中断输入引脚。在复位后立即变为输入端口，通过设定寄存器进行功能的切换。各端口的设定取决于 I/O 端口的寄存器设定和内部外围模块的寄存器设定。

各端口有指定输入/输出的端口方向寄存器 (PDR)、保存输出数据的端口输出数据寄存器 (PODR)、反映引脚状态的端口输入数据寄存器 (PIDR)、选择引脚输出形态的漏极开路控制寄存器 y (ODRy) (y=0、1)、控制输入上拉 MOS 的 ON/OFF 的上拉控制寄存器 (PCR)、控制驱动能力切换的驱动能力控制寄存器 (DSCR) 以及指定功能引脚的端口模式寄存器 (PMR)。有关 PMR 寄存器的详细内容，请参照“20. 多功能引脚控制器 (MPC)”。

I/O 端口的结构因封装而不同，I/O 端口的规格和功能分别如表 19.1 和表 19.2 所示。

表 19.1 I/O 端口的规格

端口符号	封装		封装		封装	
	100 引脚	个数	80 引脚	个数	64 引脚	个数
PORT0	P03、P05、P07	3	P03、P05、P07	3	P03、P05	2
PORT1	P12 ~ P17	6	P12 ~ P17	6	P14 ~ P17	4
PORT2	P20 ~ P27	8	P20、P21、P26、P27	4	P26、P27	2
PORT3	P30 ~ P37	8	P30 ~ P32、P34 ~ P37	7	P30 ~ P32、P35 ~ P37	6
PORT4	P40 ~ P47	8	P40 ~ P47	8	P40 ~ P44、P46	6
PORT5	P50 ~ P55	6	P54、P55	2	P54、P55	2
PORTA	PA0 ~ PA7	8	PA0 ~ PA6	7	PA0、PA1、PA3、PA4、PA6	5
PORTB	PB0 ~ PB7	8	PB0 ~ PB7	8	PB0、PB1、PB3、PB5 ~ PB7	6
PORTC	PC0 ~ PC7	8	PC2 ~ PC7	6	PC2 ~ PC7	6
PORTD	PD0 ~ PD7	8	PD0 ~ PD2	3	无	0
PORTE	PE0 ~ PE7	8	PE0 ~ PE5	6	PE0 ~ PE5	6
PORTH	PH0 ~ PH3	4	PH0 ~ PH3	4	PH0 ~ PH3	4
PORTJ	PJ1、PJ3	2	PJ1	1	无	0
	端口总数	85	端口总数	65	端口总数	49

表 19.2 I/O 端口的功能

端口符号	端口	输入上拉功能	漏极开路输出功能	驱动能力切换功能	5V 容许
PORT0	P03、P05、P07	○	—	通常固定为输出	—
PORT1	P12、P13、P16、P17	○	○	○	○
	P14、P15	○	○	○	—
PORT2	P20 ~ P27	○	○	○	—
PORT3	P30 ~ P34、P36、P37	○	○	○	—
	P35	—	—	—	—
PORT4	P40 ~ P47	○	—	通常固定为输出	—
PORT5	P50 ~ P55	○	—	○	—
PORTA	PA0 ~ PA7	○	○	○	—
PORTB	PB0 ~ PB7	○	○	○	—
PORTC	PC0 ~ PC7	○	○	○	—
PORTD	PD0 ~ PD7	○	—	○	—
PORTE	PE0 ~ PE7	○	○	○	—
PORTH	PH0 ~ PH3	○	—	○	—
PORTJ	PJ1、PJ3	○	—	○	—

输入上拉功能、漏极开路输出功能、驱动能力切换功能和 5V 容许的设定也对通用输入 / 输出端口和引脚复用的其他信号有效。

19.2 输入 / 输出端口的结构

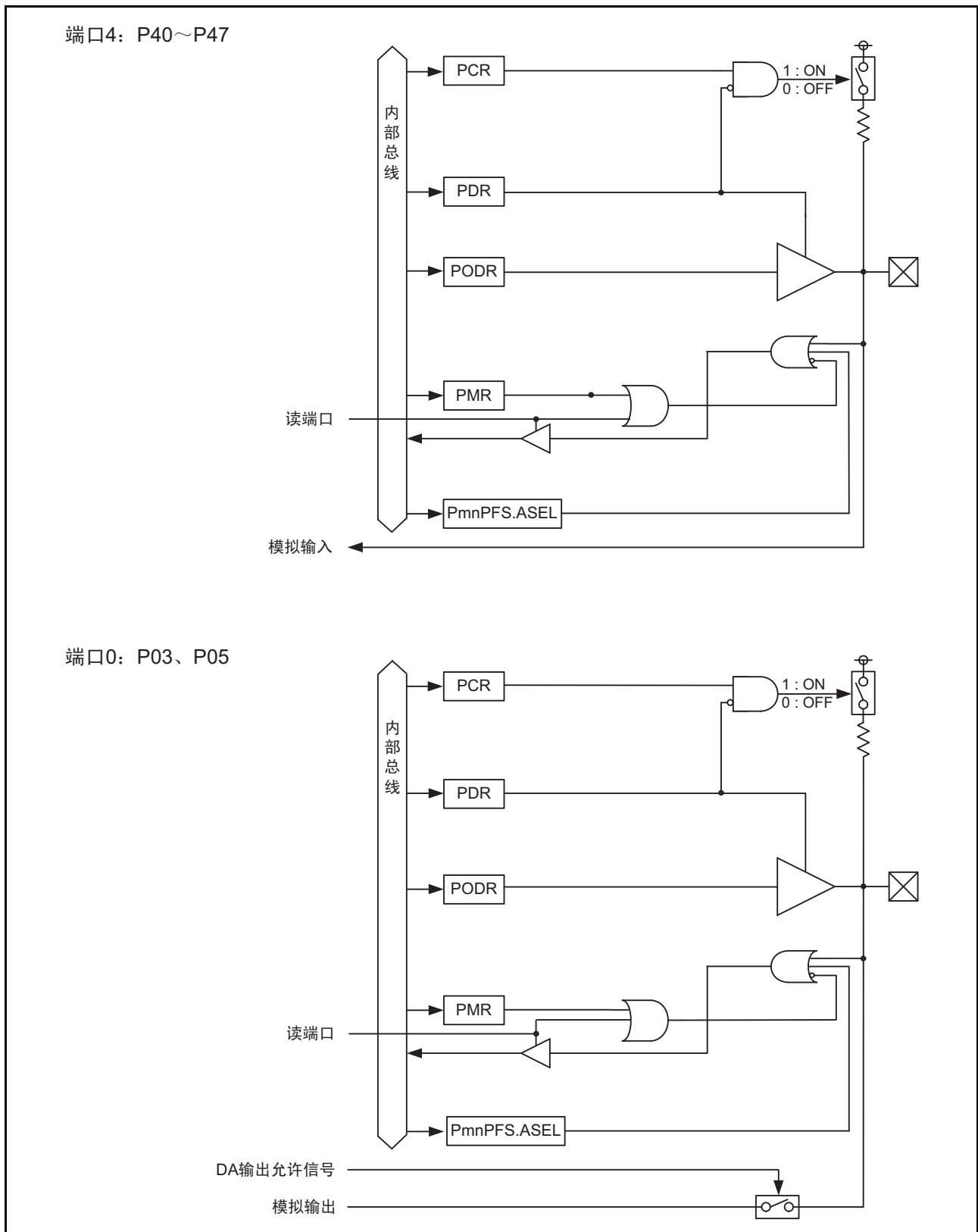


图 19.1 输入 / 输出端口的结构 (1)

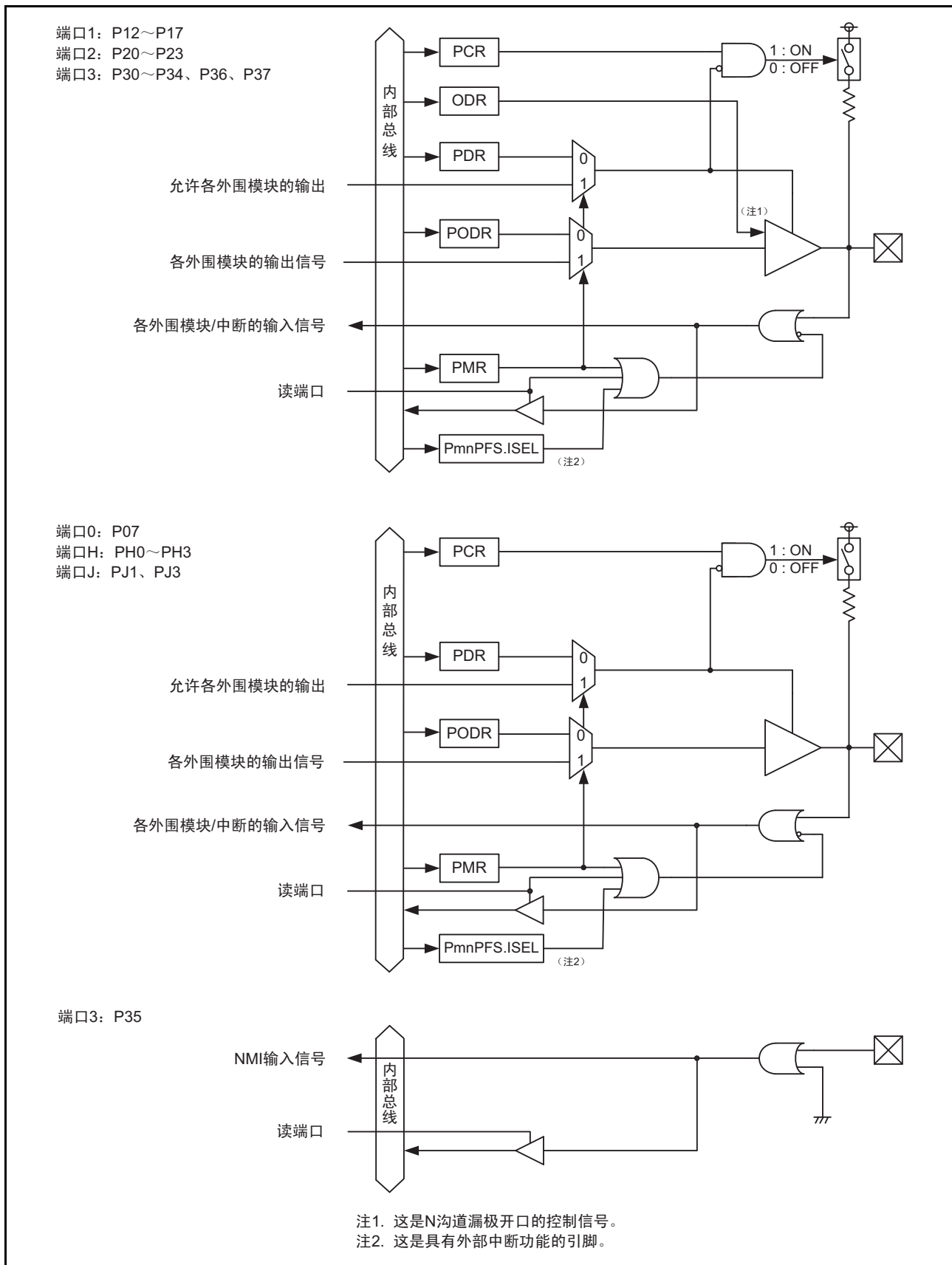


图 19.2 输入 / 输出端口的结构 (2)

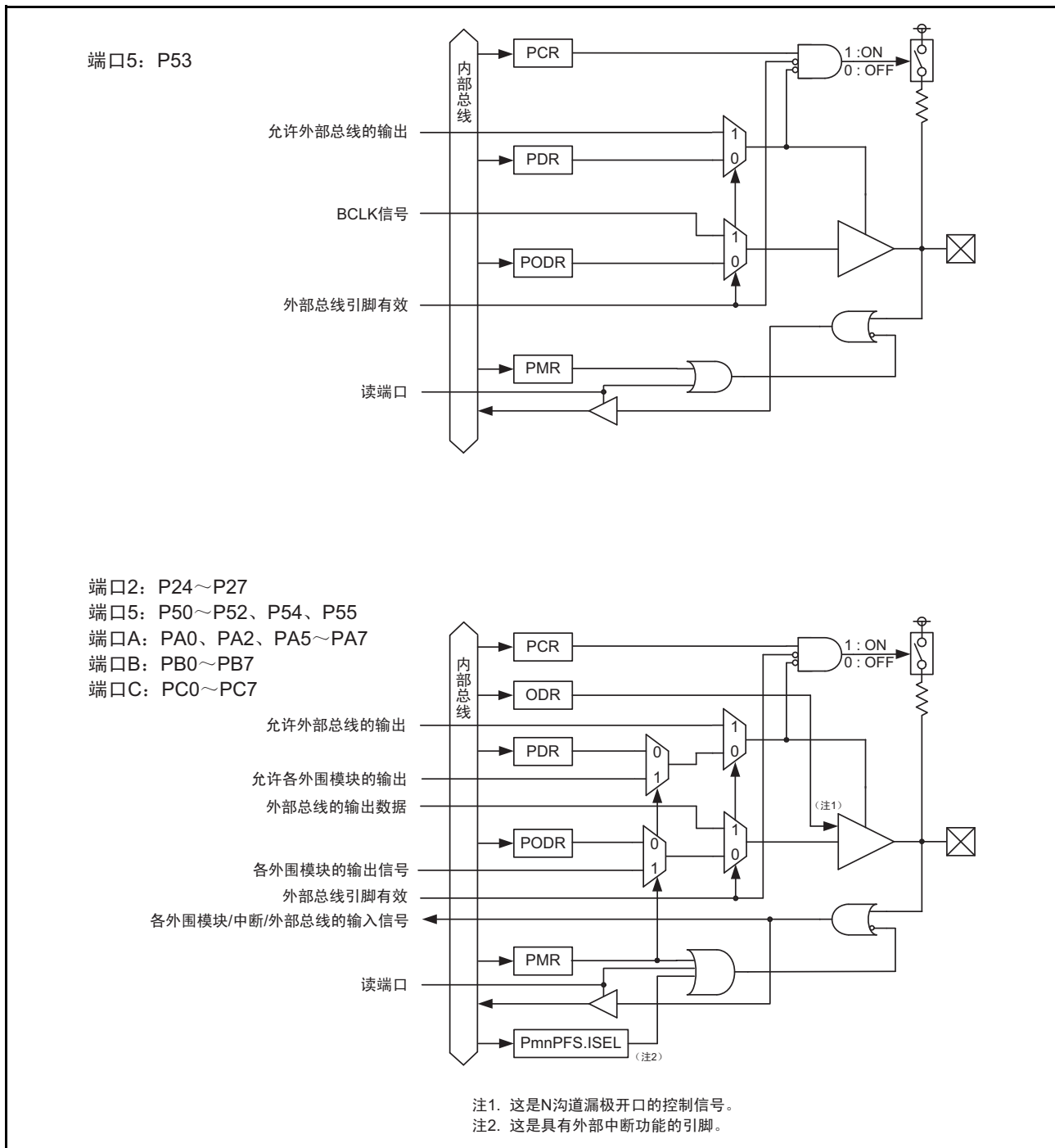


图 19.3 输入 / 输出端口的结构 (3)

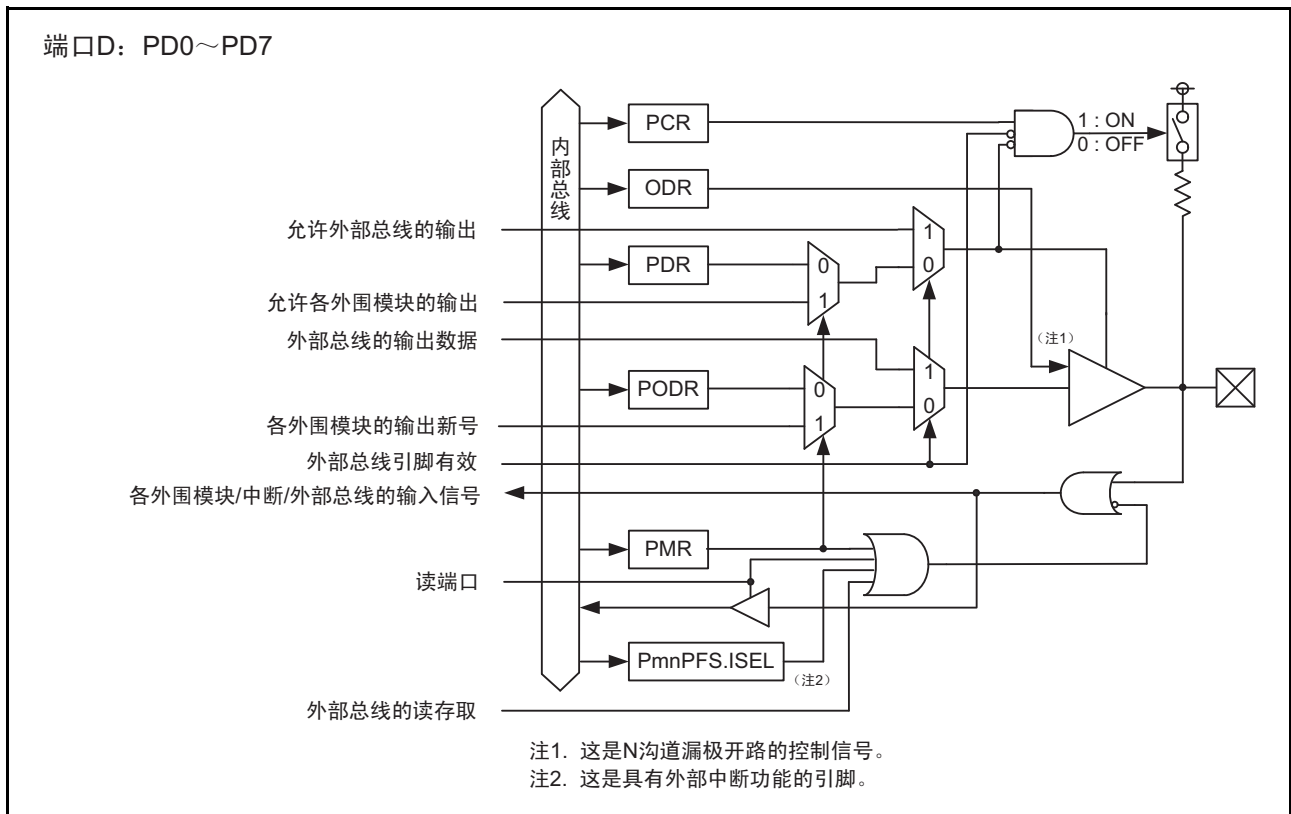


图 19.4 输入 / 输出端口的结构 (4)

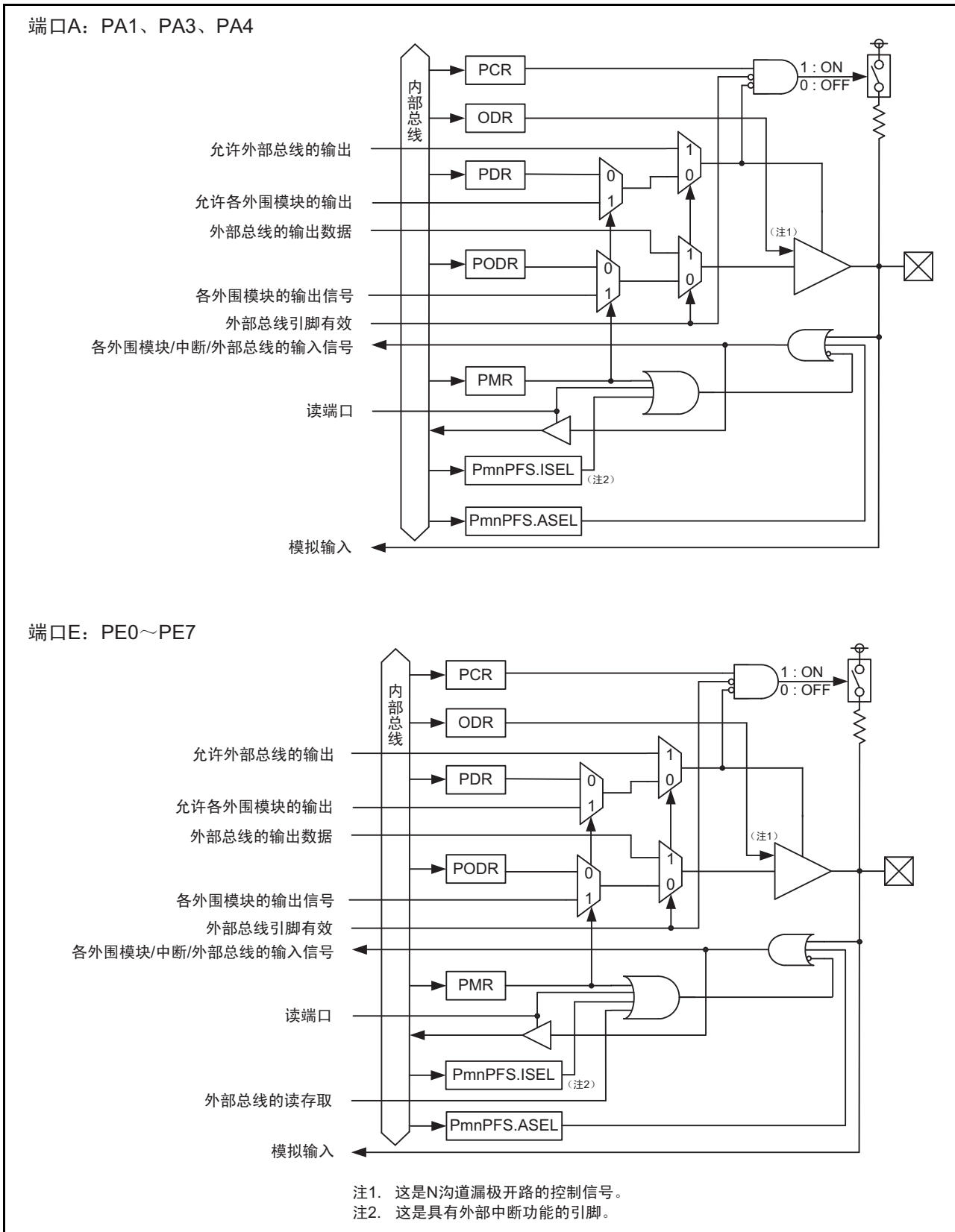


图 19.5 输入 / 输出端口的结构 (5)

19.3 寄存器说明

19.3.1 端口方向寄存器 (PDR)

地址 PORT0.PDR 0008 C000h、PORT1.PDR 0008 C001h、PORT2.PDR 0008 C002h、PORT3.PDR 0008 C003h、PORT4.PDR 0008 C004h、PORT5.PDR 0008 C005h、PORTA.PDR 0008 C00Ah、PORTB.PDR 0008 C00Bh、PORTC.PDR 0008 C00Ch、PORTD.PDR 0008 C00Dh、PORTE.PDR 0008 C00Eh、PORTH.PDR 0008 C011h、PORTJ.PDR 0008 C012h

b7	b6	b5	b4	b3	b2	b1	b0
B7	B6	B5	B4	B3	B2	B1	B0
复位后的值	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	B0	Pm0 方向控制位	0: 输入 (用作输入端口) 1: 输出 (用作输出端口)	R/W
b1	B1	Pm1 方向控制位		R/W
b2	B2	Pm2 方向控制位		R/W
b3	B3	Pm3 方向控制位		R/W
b4	B4	Pm4 方向控制位		R/W
b5	B5	Pm5 方向控制位		R/W
b6	B6	Pm6 方向控制位		R/W
b7	B7	Pm7 方向控制位		R/W

m=0 ~ 5、A ~ E、H、J

PDR 寄存器是在选择了通用输入 / 输出端口功能时指定端口输入 / 输出的寄存器。

PORTm.PDR 寄存器的各位分别对应端口 m 的 1 个引脚，并且以 1 位为单位进行指定。但是，对于不满 100 引脚的产品，不存在的端口 m 的引脚的位为保留位，并且必须写“1”（输出）。

因为 P35 引脚为输入专用引脚，所以 PORT3.PDR.B5 位为保留位。

不存在的引脚的位为保留位，保留位的读写值都为“0”。

19.3.2 端口输出数据寄存器 (PODR)

地址 PORT0.PODR 0008 C020h、PORT1.PODR 0008 C021h、PORT2.PODR 0008 C022h、PORT3.PODR 0008 C023h、
PORT4.PODR 0008 C024h、PORT5.PODR 0008 C025h、PORTA.PODR 0008 C02Ah、PORTB.PODR 0008 C02Bh、
PORTC.PODR 0008 C02Ch、PORTD.PODR 0008 C02Dh、PORTE.PODR 0008 C02Eh、PORTH.PODR 0008 C031h、
PORTJ.PODR 0008 C032h

	b7	b6	b5	b4	b3	b2	b1	b0
	B7	B6	B5	B4	B3	B2	B1	B0
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	B0	Pm0 输出数据保存位	保存输出数据	R/W
b1	B1	Pm1 输出数据保存位		R/W
b2	B2	Pm2 输出数据保存位		R/W
b3	B3	Pm3 输出数据保存位		R/W
b4	B4	Pm4 输出数据保存位		R/W
b5	B5	Pm5 输出数据保存位		R/W
b6	B6	Pm6 输出数据保存位		R/W
b7	B7	Pm7 输出数据保存位		R/W

m=0 ~ 5、A ~ E、H、J

PODR 寄存器保存用作通用输出端口的引脚输出数据。

对于不满 100 引脚的产品，不存在的端口 m 的引脚的位为保留位，并且必须写“0”（输出 Low 电平）。

因为 P35 引脚为输入专用引脚，所以 PORT3.PODR.B5 位为保留位。不存在的引脚的位为保留位，保留位的读写值都为“0”。

19.3.3 端口输入数据寄存器 (PIDR)

地址 PORT0.PIDR 0008 C040h、PORT1.PIDR 0008 C041h、PORT2.PIDR 0008 C042h、PORT3.PIDR 0008 C043h、PORT4.PIDR 0008 C044h、PORT5.PIDR 0008 C045h、PORTA.PIDR 0008 C04Ah、PORTB.PIDR 0008 C04Bh、PORTC.PIDR 0008 C04Ch、PORTD.PIDR 0008 C04Dh、PORTE.PIDR 0008 C04Eh、PORTH.PIDR 0008 C051h、PORTJ.PIDR 0008 C052h

b7	b6	b5	b4	b3	b2	b1	b0
B7	B6	B5	B4	B3	B2	B1	B0

复位后的值 x x x x x x x x

x: 不定值

位	符号	位名	功能	R/W
b0	B0	Pm0 位	反映端口的引脚状态	R
b1	B1	Pm1 位		R
b2	B2	Pm2 位		R
b3	B3	Pm3 位		R
b4	B4	Pm4 位		R
b5	B5	Pm5 位		R
b6	B6	Pm6 位		R
b7	B7	Pm7 位		R

m=0 ~ 5、A ~ E、H、J

PIDR 寄存器是反映端口引脚状态的寄存器。

如果读 PORTm.PIDR 寄存器，就能读到引脚状态，与 PORTm.PDR 寄存器和 PORTm.PMR 的值无关。

P35 引脚能读 NMI 引脚的状态。

不存在的引脚的位为保留位，保留位的读取值为不定值，写操作无效。

19.3.4 端口模式寄存器 (PMR)

地址 PORT0.PMR 0008 C060h、PORT1.PMR 0008 C061h、PORT2.PMR 0008 C062h、PORT3.PMR 0008 C063h、PORT4.PMR 0008 C064h、PORT5.PMR 0008 C065h、PORTA.PMR 0008 C06Ah、PORTB.PMR 0008 C06Bh、PORTC.PMR 0008 C06Ch、PORTD.PMR 0008 C06Dh、PORTE.PMR 0008 C06Eh、PORTH.PMR 0008 C071h、PORTJ.PMR 0008 C072h

b7	b6	b5	b4	b3	b2	b1	b0
B7	B6	B5	B4	B3	B2	B1	B0
复位后的值	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	B0	Pm0 引脚模式控制位	0: 用作通用输入 / 输出端口 1: 用作外围功能	R/W
b1	B1	Pm1 引脚模式控制位		R/W
b2	B2	Pm2 引脚模式控制位		R/W
b3	B3	Pm3 引脚模式控制位		R/W
b4	B4	Pm4 引脚模式控制位		R/W
b5	B5	Pm5 引脚模式控制位		R/W
b6	B6	Pm6 引脚模式控制位		R/W
b7	B7	Pm7 引脚模式控制位		R/W

m=0 ~ 5、A ~ E、H、J

PORTm.PMR 寄存器的各位分别对应端口 m 的 1 个引脚，并且能以 1 位为单位进行指定。但是，对于不满 100 引脚的产品，不存在的端口 m 的引脚的位为保留位，并且必须写“0”（通用输入 / 输出端口）。

不存在的引脚的位为保留位，保留位的读写值都为“0”。

19.3.5 漏极开路控制寄存器 0 (ODR0)

地址 PORT1.ODR0 0008 C082h、PORT2.ODR0 0008 C084h、PORT3.ODR0 0008 C086h、PORTA.ODR0 0008 C094h、
PORTB.ODR0 0008 C096h、PORTC.ODR0 0008 C098h、PORTE.ODR0 0008 C09Ch、

b7	b6	b5	b4	b3	b2	b1	b0
B7	B6	B5	B4	B3	B2	B1	B0

复位后的值 0 0 0 0 0 0 0 0

位	符号	位名	功能	R/W
b0	B0	Pm0 输出形态指定位	0: CMOS 输出 1: N 沟道漏极开路	R/W
b1	B1	保留位	读写值都为“0”。	R/W
b2	B2	Pm1 输出形态指定位	• P21、P31、PA1、PB1、PC1	R/W
b3	B3	保留位	b2 0: CMOS 输出 1: N 沟道漏极开路 b3 读写值都为“0”。 • PE1 0 0: CMOS 输出 0 1: N 沟道漏极开路 1 0: PMOS 漏极开路输出 1 1: Hi-Z	R/W
b4	B4	Pm2 输出形态指定位	0: CMOS 输出 1: N 沟道漏极开路	R/W
b5	B5	保留位	读写值都为“0”。	R/W
b6	B6	Pm3 输出形态指定位	0: CMOS 输出 1: N 沟道漏极开路	R/W
b7	B7	保留位	读写值都为“0”。	R/W

m=1 ~ 3、A ~ C、E

对于不满 100 引脚的产品，不存在的端口 m 的引脚的位为保留位，并且必须写“0”（CMOS 输出）。
不存在的引脚的位为保留位，保留位的读写值都为“0”。

19.3.6 漏极开路控制寄存器 1 (ODR1)

地址 PORT1.ODR1 0008 C083h、PORT2.ODR1 0008 C085h、PORT3.ODR1 0008 C087h、PORTA.ODR1 0008 C095h、
PORTB.ODR1 0008 C097h、PORTC.ODR1 0008 C099h、PORTE.ODR1 0008 C09Dh

b7	b6	b5	b4	b3	b2	b1	b0
B7	B6	B5	B4	B3	B2	B1	B0

复位后的值 0 0 0 0 0 0 0 0

位	符号	位名	功能	R/W
b0	B0	Pm4 输出形态指定位	0: CMOS 输出 1: N 沟道漏极开路	R/W
b1	B1	保留位	读写值都为“0”。	R/W
b2	B2	Pm5 输出形态指定位	0: CMOS 输出 1: N 沟道漏极开路	R/W
b3	B3	保留位	读写值都为“0”。	R/W
b4	B4	Pm6 输出形态指定位	0: CMOS 输出 1: N 沟道漏极开路	R/W
b5	B5	保留位	读写值都为“0”。	R/W
b6	B6	Pm7 输出形态指定位	0: CMOS 输出 1: N 沟道漏极开路	R/W
b7	B7	保留位	读写值都为“0”。	R/W

m=1 ~ 3、A ~ C、E

对于不满 100 引脚的产品，不存在的端口 m 的引脚的位为保留位，并且必须写“0”（CMOS 输出）。

因为 P35 引脚为输入专用引脚，所以 PORT3.ODR1.B2 位为保留位。不存在的引脚的位为保留位，保留位的读写值都为“0”。

19.3.7 上拉控制寄存器 (PCR)

地址 PORT0.PCR 0008 C0C0h、PORT1.PCR 0008 C0C1h、PORT2.PCR 0008 C0C2h、PORT3.PCR 0008 C0C3h、PORT4.PCR 0008 C0C4h、PORT5.PCR 0008 C0C5h、PORTA.PCR 0008 C0CAh、PORTB.PCR 0008 C0CBh、PORTC.PCR 0008 C0CCh、PORTD.PCR 0008 C0CDh、PORTE.PCR 0008 C0CEh、PORTH.PCR 0008 C0D1h、PORTJ.PCR 0008 C0D2h

	b7	b6	b5	b4	b3	b2	b1	b0
	B7	B6	B5	B4	B3	B2	B1	B0
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	B0	Pm0 输入上拉电阻控制位	0: 输入上拉电阻无效 1: 输入上拉电阻有效	R/W
b1	B1	Pm1 输入上拉电阻控制位		R/W
b2	B2	Pm2 输入上拉电阻控制位		R/W
b3	B3	Pm3 输入上拉电阻控制位		R/W
b4	B4	Pm4 输入上拉电阻控制位		R/W
b5	B5	Pm5 输入上拉电阻控制位		R/W
b6	B6	Pm6 输入上拉电阻控制位		R/W
b7	B7	Pm7 输入上拉电阻控制位		R/W

m=0 ~ 5、A ~ E、H、J

在引脚为输入的状态下，与 PORTm.PCR 寄存器为“1”的位对应的引脚的输入上拉电阻有效。

即使在深度软件待机模式中，PCR 寄存器也保持上拉状态。

当用作外部总线引脚、通用端口输出或者外围功能输出时，与 PCR 寄存器的设定值无关，上拉电阻无效。

在复位过程中，上拉电阻也无效。

PORT3.PCR.B5 位为保留位。不存在的引脚的位为保留位，保留位的读写值都为“0”。

19.3.8 驱动能力控制寄存器 (DSCR)

地址 PORT1.DSCR 0008 C0E1h、PORT2.DSCR 0008 C0E2h、PORT3.DSCR 0008 C0E3h、PORT5.DSCR 0008 C0E5h、
PORTA.DSCR 0008 C0EAh、PORTB.DSCR 0008 C0EBh、PORTC.DSCR 0008 C0ECh、PORTD.DSCR 0008 C0EDh、
PORTE.DSCR 0008 C0EEh、PORTH.DSCR 0008 C0F1h、PORTJ.DSCR 0008 C0F2h

b7	b6	b5	b4	b3	b2	b1	b0
B7	B6	B5	B4	B3	B2	B1	B0
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	B0	Pm0 驱动能力控制位	0: 正常输出 1: 高驱动输出	R/W
b1	B1	Pm1 驱动能力控制位		R/W
b2	B2	Pm2 驱动能力控制位		R/W
b3	B3	Pm3 驱动能力控制位		R/W
b4	B4	Pm4 驱动能力控制位		R/W
b5	B5	Pm5 驱动能力控制位		R/W
b6	B6	Pm6 驱动能力控制位		R/W
b7	B7	Pm7 驱动能力控制位		R/W

m=1 ~ 3、5、A ~ E、H、J

DSCR 寄存器按位控制被设定为输出端口的引脚的驱动能力。

能读写固定驱动能力的引脚的对应位，但是不能切换驱动能力。

如果变为深度软件待机模式，全部位就都为正常输出。在解除深度软件待机模式后也为正常输出。

不存在的引脚的位为保留位，保留位的读写值都为“0”。

19.3.9 未使用引脚的处理

未使用引脚的处理内容如表 19.3 所示。

表 19.3 未使用引脚的处理内容

引脚名	处理内容
MD	(用作模式引脚)
RES#	通过电阻连接 VCC (上拉)。
P35/NMI	通过电阻连接 VCC (上拉)。
EXTAL	(用作时钟引脚)
XTAL	将引脚置为开路。
XCIN	通过电阻连接 VCC (上拉) 或者 VSS (下拉)。
XCOUT	将引脚置为开路。
端口 0 ~ 5、A ~ E、H、J	每个引脚通过电阻连接 VCC (上拉) 或者 VSS (下拉)。
VREFH0	连接 VCC。
VREFL0	连接 VSS。
VREFH	连接 VCC。
VREFL	连接 VSS。

20. 多功能引脚控制器 (MPC)

20.1 概要

多功能引脚控制器 (MPC) 具有从多个端口中选择并且分配外围功能输入 / 输出和中断输入信号的功能, 也能分配外部总线相关信号的端口。

多路复用引脚的分配引脚一览表如表 20.1 所示。表中用“○”和“×”表示不同封装的引脚的有无。禁止通过多个引脚将同一功能设定为有效。

表 20.1 多路复用引脚的分配引脚一览表 (1/8)

模块 / 功能	通道	引脚功能	分配端口	封装			
				100 引脚	80 引脚	64 引脚	
中断		NMI (输入)	P35	○	○	○	
中断	IRQ0	IRQ0-DS (输入)	P30	○	○	○	
		IRQ0 (输入)	PD0	○	○	×	
			PH1	○	○	○	
	IRQ1	IRQ1-DS (输入)	P31	○	○	○	
		IRQ1 (输入)	PD1	○	○	×	
			PH2	○	○	○	
	IRQ2	IRQ2-DS (输入)	P32	○	○	○	
		IRQ2 (输入)	P12	○	○	×	
			PD2	○	○	×	
	IRQ3	IRQ3-DS (输入)	P33	○	×	×	
		IRQ3 (输入)	P13	○	○	×	
			PD3	○	×	×	
	IRQ4	IRQ4-DS (输入)	PB1	○	○	○	
		IRQ4 (输入)	P14	○	○	○	
			P34	○	○	×	
			PD4	○	×	×	
	IRQ5	IRQ5-DS (输入)	PA4	○	○	○	
		IRQ5 (输入)	P15	○	○	○	
			PD5	○	×	×	
	PE5	IRQ5 (输入)	PE5	○	○	○	
			IRQ6-DS (输入)	PA3	○	○	○
			IRQ6 (输入)	P16	○	○	○
	PD6	○		×	×		
	PE6	○		×	×		
IRQ7	IRQ7-DS (输入)	PE2	○	○	○		
	IRQ7 (输入)	P17	○	○	○		
		PD7	○	×	×		
		PE7	○	×	×		

表 20.1 多路复用引脚的分配引脚一览表 (2/8)

模块 / 功能	通道	引脚功能	分配端口	封装			
				100 引脚	80 引脚	64 引脚	
多功能定时器单元 2	MTU0	MTIOC0A (输入 / 输出)	P34	○	○	×	
			PB3	○	○	○	
		MTIOC0B (输入 / 输出)	P13	○	○	×	
			P15	○	○	○	
			PA1	○	○	○	
		MTIOC0C (输入 / 输出)	P32	○	○	○	
			PB1	○	○	○	
		MTIOC0D (输入 / 输出)	P33	○	×	×	
			PA3	○	○	○	
		MTU1	MTIOC1A (输入 / 输出)	P20	○	○	×
				PE4	○	○	○
			MTIOC1B (输入 / 输出)	P21	○	○	×
	PB5			○	○	○	
	MTU2	MTIOC2A (输入 / 输出)	P26	○	○	○	
			PB5	○	○	○	
		MTIOC2B (输入 / 输出)	P27	○	○	○	
	PE5		○	○	○		
	MTU3	MTIOC3A (输入 / 输出)	P14	○	○	○	
			P17	○	○	○	
			PC1	○	×	×	
			PC7	○	○	○	
			PJ1	○	○	×	
		MTIOC3B (输入 / 输出)	P17	○	○	○	
			P22	○	×	×	
			PB7	○	○	○	
			PC5	○	○	○	
		MTIOC3C (输入 / 输出)	P16	○	○	○	
			PC0	○	×	×	
			PC6	○	○	○	
			PJ3	○	×	×	
		MTIOC3D (输入 / 输出)	P16	○	○	○	
			P23	○	×	×	
PB6			○	○	○		
PC4	○		○	○			

表 20.1 多路复用引脚的分配引脚一览表 (3/8)

模块 / 功能	通道	引脚功能	分配端口	封装		
				100 引脚	80 引脚	64 引脚
多功能定时器单元 2	MTU4	MTIOC4A (输入 / 输出)	P24	○	×	×
			PA0	○	○	○
			PB3	○	○	○
			PE2	○	○	○
		MTIOC4B (输入 / 输出)	P30	○	○	○
			P54	○	○	○
			PC2	○	○	○
			PD1	○	○	×
		MTIOC4C (输入 / 输出)	PE3	○	○	○
			P25	○	×	×
			PB1	○	○	○
			PE1	○	○	○
		MTIOC4D (输入 / 输出)	PE5	○	○	○
			P31	○	○	○
			P55	○	○	○
			PC3	○	○	○
	MTU5	MTIC5U (输入 / 输出)	PD2	○	○	×
			PE4	○	○	○
		MTIC5V (输入 / 输出)	PA4	○	○	○
			PD7	○	×	×
	MTIC5W (输入 / 输出)	PA6	○	○	○	
		PD6	○	×	×	
	MTU	MTCLKA (输入 / 输出)	PB0	○	○	○
			PD5	○	×	×
			P14	○	○	○
			P24	○	×	×
		MTCLKB (输入 / 输出)	PA4	○	○	○
			PC6	○	○	○
			P15	○	○	○
			P25	○	×	×
		MTCLKC (输入 / 输出)	PA6	○	○	○
			PC7	○	○	○
P22			○	×	×	
MTCLKD (输入 / 输出)		PA1	○	○	○	
		PC4	○	○	○	
		P23	○	×	×	
			PA3	○	○	○
			PC5	○	○	○

表 20.1 多路复用引脚的分配引脚一览表 (4/8)

模块 / 功能	通道	引脚功能	分配端口	封装			
				100 引脚	80 引脚	64 引脚	
端口输出允许 2	POE0	POE0# (输入)	PC4	○	○	○	
			PD7	○	×	×	
	POE1	POE1# (输入)	PB5	○	○	○	
			PD6	○	×	×	
	POE2	POE2# (输入)	P34	○	○	×	
			PA6	○	○	○	
			PD5	○	×	×	
	POE3	POE3# (输入)	P33	○	×	×	
			PB3	○	○	○	
			PD4	○	×	×	
	POE8	POE8# (输入)	P17	○	○	○	
			P30	○	○	○	
			PD3	○	×	×	
			PE3	○	○	○	
	8 位定时器	TMR0	TMO0 (输出)	P22	○	×	×
				PB3	○	○	○
PH1				○	○	○	
TMCIO (输入)			P21	○	○	×	
			PB1	○	○	○	
			PH3	○	○	○	
TMRIO (输入)			P20	○	○	×	
			PA4	○	○	○	
			PH2	○	○	○	
TMR1		TMO1 (输出)	P17	○	○	○	
			P26	○	○	○	
		TMC11 (输入)	P12	○	○	×	
			P54	○	○	○	
		PC4	○	○	○		
			TMRI1 (输入)	P24	○	×	×
PB5		○	○	○			
TMR2		TMO2 (输出)	P16	○	○	○	
			PC7	○	○	○	
		TMC12 (输入)	P15	○	○	○	
			P31	○	○	○	
			PC6	○	○	○	
		TMRI2 (输入)	P14	○	○	○	
PC5			○	○	○		
TMR3		TMO3 (输出)	P13	○	○	×	
	P32		○	○	○		
	P55		○	○	○		
	TMC13 (输入)	P27	○	○	○		
		P34	○	○	×		
		PA6	○	○	○		
	TMRI3 (输入)	P30	○	○	○		
		P33	○	×	×		

表 20.1 多路复用引脚的分配引脚一览表 (5/8)

模块 / 功能	通道	引脚功能	分配端口	封装		
				100 引脚	80 引脚	64 引脚
串行通信接口	SCI0	RXD0 (输入) / SMISO0 (输入 / 输出) / SSCL0 (输入 / 输出)	P21	○	○	×
		TXD0 (输出) / SMOSI0 (输入 / 输出) / SSDA0 (输入 / 输出)	P20	○	○	×
		SCK0 (输入 / 输出)	P22	○	×	×
		CTS0# (输入) / RTS0# (输出) / SS0# (输入)	P23	○	×	×
	SCI1	RXD1 (输入) / SMISO1 (输入 / 输出) / SSCL1 (输入 / 输出)	P15	○	○	○
			P30	○	○	○
		TXD1 (输出) / SMOSI1 (输入 / 输出) / SSDA1 (输入 / 输出)	P16	○	○	○
			P26	○	○	○
		SCK1 (输入 / 输出)	P17	○	○	○
			P27	○	○	○
		CTS1# (输入) / RTS1# (输出) / SS1# (输入)	P14	○	○	○
			P31	○	○	○
	SCI5	RXD5 (输入) / SMISO5 (输入 / 输出) / SSCL5 (输入 / 输出)	PA2	○	○	×
			PA3	○	○	○
			PC2	○	○	○
		TXD5 (输出) / SMOSI5 (输入 / 输出) / SSDA5 (输入 / 输出)	PA4	○	○	○
			PC3	○	○	○
			SCK5 (输入 / 输出)	PA1	○	○
		PC1		○	×	×
		PC4		○	○	○
		CTS5# (输入) / RTS5# (输出) / SS5# (输入)	PA6	○	○	○
			PC0	○	×	×
	SCI6	RXD6 (输入) / SMISO6 (输入 / 输出) / SSCL6 (输入 / 输出)	P33	○	×	×
			PB0	○	○	○
		TXD6 (输出) / SMOSI6 (输入 / 输出) / SSDA6 (输入 / 输出)	P32	○	○	○
			PB1	○	○	○
		SCK6 (输入 / 输出)	P34	○	○	×
			PB3	○	○	○
		CTS6# (输入) / RTS6# (输出) / SS6# (输入)	PB2	○	○	×
			PJ3	○	×	×

表 20.1 多路复用引脚的分配引脚一览表 (6/8)

模块 / 功能	通道	引脚功能	分配端口	封装			
				100 引脚	80 引脚	64 引脚	
串行通信接口	SCI8	RXD8 (输入) / SMISO8 (输入 / 输出) / SSCL8 (输入 / 输出)	PC6	○	○	○	
		TXD8 (输出) / SMOSI8 (输入 / 输出) / SSDA8 (输入 / 输出)	PC7	○	○	○	
		SCK8 (输入 / 输出)	PC5	○	○	○	
		CTS8# (输入) / RTS8# (输出) / SS8# (输入)	PC4	○	○	○	
	SCI9	RXD9 (输入) / SMISO9 (输入 / 输出) / SSCL9 (输入 / 输出)	PB6	○	○	○	
		TXD9 (输出) / SMOSI9 (输入 / 输出) / SSDA9 (输入 / 输出)	PB7	○	○	○	
		SCK9 (输入 / 输出)	PB5	○	○	○	
		CTS9# (输入) / RTS9# (输出) / SS9# (输入)	PB4	○	○	×	
	SCI12	RXD12 (输入) / SMISO12 (输入 / 输出) / SSCL12 (输入 / 输出) / RXDX12 (输入)	PE2	○	○	○	
		TXD12 (输出) / SMOSI12 (输入 / 输出) / SSDA12 (输入 / 输出) / TXDX12 (输出) / SIOX12 (输入 / 输出)	PE1	○	○	○	
		SCK12 (输入 / 输出)	PE0	○	○	○	
		CTS12# (输入) / RTS12# (输出) / SS12# (输入)	PE3	○	○	○	
	I ² C 总线接口	RIIC0	SCL-DS (输入 / 输出)	P16	○	○	○
			SCL (输入 / 输出)	P12	○	○	×
SDA-DS (输入 / 输出)			P17	○	○	○	
SDA (输入 / 输出)			P13	○	○	×	

表 20.1 多路复用引脚的分配引脚一览表 (7/8)

模块 / 功能	通道	引脚功能	分配端口	封装		
				100 引脚	80 引脚	64 引脚
串行外围接口	RSPI0	RSPCKA (输入 / 输出)	PA5	○	○	×
			PB0	○	○	○
			PC5	○	○	○
		MOSIA (输入 / 输出)	P16	○	○	○
			PA6	○	○	○
			PC6	○	○	○
		MISOA (输入 / 输出)	P17	○	○	○
			PA7	○	×	×
			PC7	○	○	○
		SSLA0 (输入 / 输出)	PA4	○	○	○
			PC4	○	○	○
		SSLA1 (输出)	PA0	○	○	○
			PC0	○	×	×
		SSLA2 (输出)	PA1	○	○	○
			PC1	○	×	×
		SSLA3 (输出)	PA2	○	○	×
			PC2	○	○	○
		实时时钟	RTCOUT (输出)	P16	○	○
P32	○			○	○	
RTCIC0 (输入) (注1)	P30			○	○	○
RTCIC1 (输入) (注1)	P31			○	○	○
RTCIC2 (输入) (注1)	P32	○	○	○		
12 位 A/D 转换器	AN000 (输入) (注1)	P40	○	○	○	
		P41	○	○	○	
		P42	○	○	○	
		P43	○	○	○	
		P44	○	○	○	
		P45	○	○	×	
		P46	○	○	○	
		P47	○	○	×	
		PE0	○	○	○	
		PE1	○	○	○	
		PE2	○	○	○	
		PE3	○	○	○	
		PE4	○	○	○	
		PE5	○	○	○	
		PE6	○	×	×	
		PE7	○	×	×	
		ADTRG0# (输入)	P07	○	○	×
			P16	○	○	○
P25	○		×	×		
D/A 转换器	DA0 (输出) (注1)	P03	○	○	○	
		P05	○	○	○	
时钟频率精度测量电路	CACREF (输入)	PA0	○	○	○	
		PC7	○	○	○	
		PH0	○	○	○	

表 20.1 多路复用引脚的分配引脚一览表 (8/8)

模块 / 功能	通道	引脚功能	分配端口	封装		
				100 引脚	80 引脚	64 引脚
比较器 A		CMPA1 (输入) (注 1)	PE3	○	○	○
		CMPA2 (输入) (注 1)	PE4	○	○	○
		CVREFA (输入) (注 1)	PA1	○	○	○
比较器 B		CMPB0 (输入) (注 1)	PE1	○	○	○
		CVREFB0 (输入) (注 1)	PE2	○	○	○
		CMPB1 (输入) (注 1)	PA3	○	○	○
		CVREFB1 (输入) (注 1)	PA4	○	○	○
外部总线 (注 2)	CS0# (输出)		P24	○	×	×
			PC7	○	×	×
	CS1# (输出)		P25	○	×	×
			PC6	○	×	×
	CS2# (输出)		P26	○	×	×
			PC5	○	×	×
	CS3# (输出)		P27	○	×	×
			PC4	○	×	×
	A0 ~ A7 (输出)		PA0 ~ PA7	○	×	×
	A8 ~ A15 (输出)		PB0 ~ PB7	○	×	×
	A16 ~ A23 (输出)		PC0 ~ PC7	○	×	×
	D0 ~ D7 (输入 / 输出)		PD0 ~ PD7	○	×	×
	D8 ~ D15 (输入 / 输出)		PE0 ~ PE7	○	×	×
	BCLK (输出)		P53	○	×	×
	RD# (输出)		P52	○	×	×
	WR# (输出)		P50	○	×	×
	WR0# (输出)		P50	○	×	×
	WR1# (输出)		P51	○	×	×
	BC0# (输出)		PA0	○	×	×
	BC1# (输出)		P51	○	×	×
	WAIT# (输入)		P51	○	×	×
			P55	○	×	×
			PC5	○	×	×
ALE (输出)		P54	○	×	×	

注 1. 在使用此引脚功能时, 必须将对应的引脚设定为通用输入 (将 PORT.PDR.Bm 位和 PORT.PMR.Bm 位置“0”)。

注 2. 有关外部总线的设定, 请参照“20.3 外部总线接口的设定方法”。

20.2 寄存器说明

根据不同的封装，保留没有引脚的寄存器和位。要给对应的位写值时，必须写复位后的值。

20.2.1 写保护寄存器 (PWPR)

地址 0008 C11Fh

	b7	b6	b5	b4	b3	b2	b1	b0
	B0WI	PFSWE	—	—	—	—	—	—
复位后的值	1	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b5-b0	—	保留位	读写值都为“0”。	R/W
b6	PFSWE	PFS 寄存器的写允许位	0: 禁止写 PFS 寄存器 1: 允许写 PFS 寄存器	R/W
b7	B0WI	PFSWE 位的写禁止位	0: 允许写 PFSWE 位 1: 禁止写 PFSWE 位	R/W

PFSWE 位 (PFS 寄存器的写允许位)

只有在 PFSWE 位的写入值为“1”时 PmnPFS 的写操作才有效。

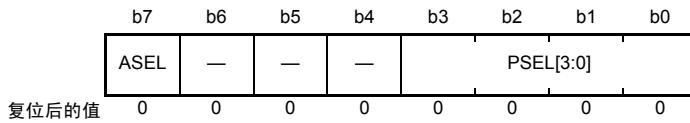
要将 PFSWE 位置“1”时，必须在给 B0WI 位写“0”后将 PFSWE 位置“1”。

B0WI 位 (PFSWE 位的写禁止位)

只有在 B0WI 位的写入值为“0”时 PFSWE 位的写操作才有效。

20.2.2 P0n 引脚功能控制寄存器 (P0nPFS) (n=3、5、7)

地址 P03PFS 0008 C143h、P05PFS 0008 C145h、P07PFS 0008 C147h



位	符号	位名	功能	R/W
b3-b0	PSEL[3:0]	引脚功能选择位	选择外围功能。有关各引脚功能，请参照表 20.2。	R/W
b6-b4	—	保留位	读写值都为“0”。	R/W
b7	ASEL	模拟输入功能选择位	0: 不用作模拟引脚 1: 用作模拟引脚 P03: DA0 (100 引脚、80 引脚、64 引脚) P05: DA1 (100 引脚、80 引脚、64 引脚)	R/W

Pmn 引脚功能控制寄存器 (PmnPFS) 选择引脚功能。

通过 PSEL[3:0] 位设定分配给引脚的外围功能。

在用作 IRQ 输入引脚时设定 ISEL 位，也能和外围功能组合使用。但是，禁止通过 2 个或者 2 个以上的引脚来允许同一序号的 IRQn (外部引脚中断)。

在将引脚用作模拟引脚时设定 ASEL 位。要通过 ASEL 位设定为模拟引脚时，必须通过端口模式寄存器 (PORTm.PMR) 选择通用输入 / 输出端口并且通过端口方向寄存器 (PORTm.PDR) 设定为输入。此时，不能读引脚状态。PmnPFS 寄存器受写保护寄存器 (PWPR) 的保护。要改写时，必须先解除保护。

未将 IRQn 多路复用的引脚的 ISEL 位为保留位，未将模拟输入 / 输出多路复用的引脚的 ASEL 位为保留位。

表 20.2 100 引脚 TFLGA、100 引脚 LQFP、80 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的设定值	引脚		
	P03	P05	P07
0000b (初始值)	Hi-Z		
1001b	—	—	ADTRG0#

—: 不能设定。

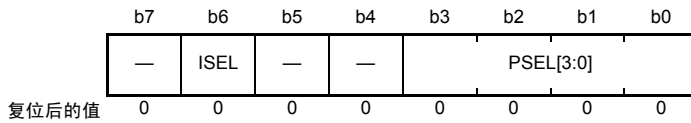
表 20.3 64 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的设定值	引脚	
	P03	P05
0000b (初始值)	Hi-Z	
1001b	—	—

—: 不能设定。

20.2.3 P1n 引脚功能控制寄存器 (P1nPFS) (n=2 ~ 7)

地址 P12PFS 0008 C14Ah、P13PFS 0008 C14Bh、P14PFS 0008 C14Ch、P15PFS 0008 C14Dh、P16PFS 0008 C14Eh、P17PFS 0008 C14Fh



位	符号	位名	功能	R/W
b3-b0	PSEL[3:0]	引脚功能选择位	选择外围功能。有关各引脚功能，请参照表 20.4。	R/W
b5-b4	—	保留位	读写值都为“0”。	R/W
b6	ISEL	中断输入功能选择位	0: 不用作 IRQn 输入引脚 1: 用作 IRQn 输入引脚 P12: IRQ2 (100 引脚、80 引脚) P13: IRQ3 (100 引脚、80 引脚) P14: IRQ4 (100 引脚、80 引脚、64 引脚) P15: IRQ5 (100 引脚、80 引脚、64 引脚) P16: IRQ6 (100 引脚、80 引脚、64 引脚) P17: IRQ7 (100 引脚、80 引脚、64 引脚)	R/W
b7	—	保留位	读写值都为“0”。	R/W

表 20.4 100 引脚 TFLGA、100 引脚 LQFP、80 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的设定值	引脚					
	P12	P13	P14	P15	P16	P17
0000b (初始值)	Hi-Z					
0001b	—	MTIOC0B	MTIOC3A	MTIOC0B	MTIOC3C	MTIOC3A
0010b	—	—	MTCLKA	MTCLKB	MTIOC3D	MTIOC3B
0101b	TMCI1	TMO3	TMRI2	TMCI2	TMO2	TMO1
0111b	—	—	—	—	RTCOUT	POE8#
1001b	—	—	—	—	ADTRG0#	—
1010b	—	—	—	RXD1 SMISO1 SSCL1	TXD1 SMOSI1 SSDA1	SCK1
1011b	—	—	CTS1# RTS1# SS1#	—	—	—
1101b	—	—	—	—	MOSIA	MISOA
1111b	SCL	SDA	—	—	SCL-DS	SDA-DS

—: 不能设定。

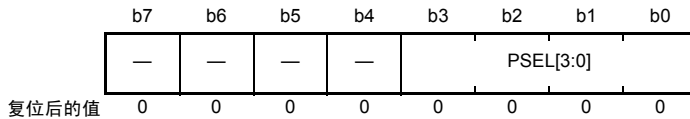
表 20.5 64 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的设定值	引脚			
	P14	P15	P16	P17
0000b (初始值)	Hi-Z			
0001b	MTIOC3A	MTIOC0B	MTIOC3C	MTIOC3A
0010b	MTCLKA	MTCLKB	MTIOC3D	MTIOC3B
0101b	TMRI2	TMCI2	TMO2	TMO1
0111b	—	—	RTCOUT	POE8#
1001b	—	—	ADTRG0#	—
1010b	—	RXD1 SMISO1 SSCL1	TXD1 SMOS1 SSDA1	SCK1
1011b	CTS1# RTS1# SS1#	—	—	—
1101b	—	—	MOSIA	MISOA
1111b	—	—	SCL-DS	SDA-DS

—: 不能设定。

20.2.4 P2n 引脚功能控制寄存器 (P2nPFS) (n=0 ~ 7)

地址 P20PFS 0008 C150h、P21PFS 0008 C151h、P22PFS 0008 C152h、P23PFS 0008 C153h、P24PFS 0008 C154h、P25PFS 0008 C155h、P26PFS 0008 C156h、P27PFS 0008 C157h



位	符号	位名	功能	R/W
b3-b0	PSEL[3:0]	引脚功能选择位	选择外围功能。有关各引脚功能，请参照表 20.6。	R/W
b7-b4	—	保留位	读写值都为“0”。	R/W

表 20.6 100 引脚 TFLGA、100 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚							
	P20	P21	P22	P23	P24	P25	P26	P27
0000b (初始值)	Hi-Z							
0001b	MTIOC1A	MTIOC1B	MTIOC3B	MTIOC3D	MTIOC4A	MTIOC4C	MTIOC2A	MTIOC2B
0010b	—	—	MTCLKC	MTCLKD	MTCLKA	MTCLKB	—	—
0101b	TMRI0	TMCIO	TMO0	—	TMRI1	—	TMO1	TMCI3
1001b	—	—	—	—	—	ADTRG0#	—	—
1010b	TXD0 SMOSI0 SSDA0	RXD0 SMISO0 SSCL0	SCK0	—	—	—	TXD1 SMOSI1 SSDA1	SCK1
1011b	—	—	—	CTS0# RTS0# SS0#	—	—	—	—

—: 不能设定。

表 20.7 80 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚			
	P20	P21	P26	P27
0000b (初始值)	Hi-Z			
0001b	MTIOC1A	MTIOC1B	MTIOC2A	MTIOC2B
0010b	—	—	—	—
0101b	TMRI0	TMCIO	TMO1	TMCI3
1001b	—	—	—	—
1010b	TXD0 SMOSI0 SSDA0	RXD0 SMISO0 SSCL0	TXD1 SMOSI1 SSDA1	SCK1
1011b	—	—	—	—

—: 不能设定。

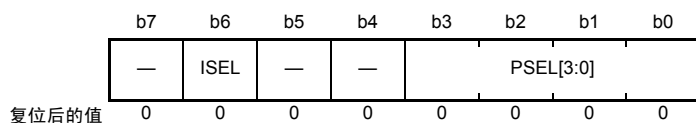
表 20.8 64 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚	
	P26	P27
0000b (初始值)	Hi-Z	
0001b	MTIOC2A	MTIOC2B
0010b	—	—
0101b	TMO1	TMCI3
1001b	—	—
1010b	TXD1 SMOSI1 SSDA1	SCK1
1011b	—	—

—: 不能设定。

20.2.5 P3n 引脚功能控制寄存器 (P3nPFS) (n=0 ~ 4)

地址 P30PFS 0008 C158h、P31PFS 0008 C159h、P32PFS 0008 C15Ah、P33PFS 0008 C15Bh、P34PFS 0008 C15Ch



位	符号	位名	功能	R/W
b3-b0	PSEL[3:0]	引脚功能选择位	选择外围功能。有关各引脚功能，请参照表 20.9。	R/W
b5-b4	—	保留位	读写值都为“0”。	R/W
b6	ISEL	中断输入功能选择位	0: 不用作 IRQn 输入引脚 1: 用作 IRQn 输入引脚 P30: IRQ0-DS (100 引脚、80 引脚、64 引脚) P31: IRQ1-DS (100 引脚、80 引脚、64 引脚) P32: IRQ2-DS (100 引脚、80 引脚、64 引脚) P33: IRQ3-DS (100 引脚) P34: IRQ4 (100 引脚、80 引脚)	R/W
b7	—	保留位	读写值都为“0”。	R/W

表 20.9 100 引脚 TFLGA、100 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚				
	P30	P31	P32	P33	P34
0000b (初期值)	Hi-Z				
0001b	MTIOC4B	MTIOC4D	MTIOC0C	MTIOC0D	MTIOC0A
0101b	TMRI3	TMCI2	TMO3	TMRI3	TMCI3
0111b	POE8#	—	RTCOUT	POE3#	POE2#
1010b	RXD1 SMISO1 SSCL1	—	—	—	—
1011b	—	CTS1# RTS1# SS1#	TXD6 SMOSI6 SSDA6	RXD6 SMISO6 SSCL6	SCK6

—: 不能设定。

表 20.10 80 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚			
	P30	P31	P32	P34
0000b (初始值)	Hi-Z			
0001b	MTIOC4B	MTIOC4D	MTIOC0C	MTIOC0A
0101b	TMRI3	TMCI2	TMO3	TMCI3
0111b	POE8#	—	RTCOUT	POE2#
1010b	RXD1 SMISO1 SSCL1	—	—	—
1011b	—	CTS1# RTS1# SS1#	TXD6 SMOSI6 SSDA6	SCK6

—: 不能设定。

表 20.11 64 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚		
	P30	P31	P32
0000b (初始值)	Hi-Z		
0001b	MTIOC4B	MTIOC4D	MTIOC0C
0101b	TMRI3	TMCI2	TMO3
0111b	POE8#	—	RTCOUT
1010b	RXD1 SMISO1 SSCL1	—	—
1011b	—	CTS1# RTS1# SS1#	TXD6 SMOSI6 SSDA6

—: 不能设定。

20.2.6 P4n 引脚功能控制寄存器 (P4nPFS) (n=0 ~ 7)

地址 P40PFS 0008 C160h、P41PFS 0008 C161h、P42PFS 0008 C162h、P43PFS 0008 C163h、P44PFS 0008 C164h、P45PFS 0008 C165h、P46PFS 0008 C166h、P47PFS 0008 C167h

b7	b6	b5	b4	b3	b2	b1	b0
ASEL	—	—	—	—	—	—	—

复位后的值 0 0 0 0 0 0 0 0

位	符号	位名	功能	R/W
b6-b0	—	保留位	读写值都为“0”。	R/W
b7	ASEL	模拟功能选择位	0: 不用作模拟引脚 1: 用作模拟引脚 P40: AN000 (100 引脚、80 引脚、64 引脚) P41: AN001 (100 引脚、80 引脚、64 引脚) P42: AN002 (100 引脚、80 引脚、64 引脚) P43: AN003 (100 引脚、80 引脚、64 引脚) P44: AN004 (100 引脚、80 引脚、64 引脚) P45: AN005 (100 引脚、80 引脚) P46: AN006 (100 引脚、80 引脚、64 引脚) P47: AN007 (100 引脚、80 引脚)	R/W

20.2.7 P5n 引脚功能控制寄存器 (P5nPFS) (n=4 ~ 5)

地址 P54PFS 0008 C16Ch、P55PFS 0008 C16Dh

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	PSEL[3:0]			

复位后的值 0 0 0 0 0 0 0 0

位	符号	位名	功能	R/W
b3-b0	PSEL[3:0]	引脚功能选择位	选择外围功能。有关各引脚功能，请参照表 20.12。	R/W
b7-b4	—	保留位	读写值都为“0”。	R/W

表 20.12 100 引脚 TFLGA、100 引脚 LQFP、80 引脚 LQFP、64 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚	
	P54	P55
0000b (初始值)	Hi-Z	
0001b	MTIOC4B	MTIOC4D
0101b	TMCI1	TMO3

—: 不能设定。

20.2.8 PAn 引脚功能控制寄存器 (PAnPFS) (n=0 ~ 7)

地址 PA0PFS 0008 C190h、PA1PFS 0008 C191h、PA2PFS 0008 C192h、PA3PFS 0008 C193h、PA4PFS 0008 C194h、PA5PFS 0008 C195h、PA6PFS 0008 C196h、PA7PFS 0008 C197h

b7	b6	b5	b4	b3	b2	b1	b0
ASEL	ISEL	—	—	PSEL[3:0]			
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b3-b0	PSEL[3:0]	引脚功能选择位	选择外围功能。有关各引脚功能，请参照表 20.13。	R/W
b5-b4	—	保留位	读写值都为“0”。	R/W
b6	ISEL	中断输入功能选择位	0: 不用作 IRQn 输入引脚 1: 用作 IRQn 输入引脚 PA3: IRQ6-DS (100 引脚、80 引脚、64 引脚) PA4: IRQ5-DS (100 引脚、80 引脚、64 引脚)	R/W
b7	ASEL	模拟功能选择位	0: 不用作模拟引脚 1: 用作模拟引脚 PA1: CVREFA (100 引脚、80 引脚、64 引脚) PA3: CMPB1 (100 引脚、80 引脚、64 引脚) PA4: CVREFB1 (100 引脚、80 引脚、64 引脚)	R/W

表 20.13 100 引脚 TFLGA、100 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚							
	PA0	PA1	PA2	PA3	PA4	PA5	PA6	PA7
0000b (初始值)	Hi-Z							
0001b	MTIOC4A	MTIOC0B	—	MTIOC0D	MTIC5U	—	MTIC5V	—
0010b	—	MTCLKC	—	MTCLKD	MTCLKA	—	MTCLKB	—
0101b	—	—	—	—	TMRI0	—	TMCI3	—
0111b	CACREF	—	—	—	—	—	POE2#	—
1010b	—	SCK5	RXD5 SMISO5 SSCL5	RXD5 SMISO5 SSCL5	TXD5 SMOSI5 SSDA5	—	—	—
1011b	—	—	—	—	—	—	CTS5# RTS5# SS5#	—
1101b	SSLA1	SSLA2	SSLA3	—	SSLA0	RSPCKA	MOSIA	MISOA

—: 不能设定。

表 20.14 80 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚						
	PA0	PA1	PA2	PA3	PA4	PA5	PA6
0000b (初始值)	Hi-Z						
0001b	MTIOC4A	MTIOC0B	—	MTIOC0D	MTIC5U	—	MTIC5V
0010b	—	MTCLKC	—	MTCLKD	MTCLKA	—	MTCLKB
0101b	—	—	—	—	TMRI0	—	TMCI3
0111b	CACREF	—	—	—	—	—	POE2#
1010b	—	SCK5	RXD5 SMISO5 SSCL5	RXD5 SMISO5 SSCL5	TXD5 SMOSI5 SSDA5	—	—
1011b	—	—	—	—	—	—	CTS5# RTS5# SS5#
1101b	SSLA1	SSLA2	SSLA3	—	SSLA0	RSPCKA	MOSIA

—: 不能设定。

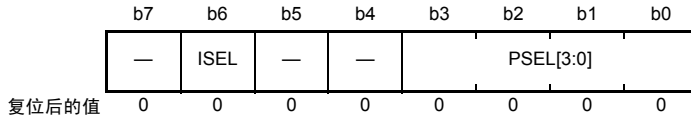
表 20.15 64 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚				
	PA0	PA1	PA3	PA4	PA6
0000b (初始值)	Hi-Z				
0001b	MTIOC4A	MTIOC0B	MTIOC0D	MTIC5U	MTIC5V
0010b	—	MTCLKC	MTCLKD	MTCLKA	MTCLKB
0101b	—	—	—	TMRI0	TMCI3
0111b	CACREF	—	—	—	POE2#
1010b	—	SCK5	RXD5 SMISO5 SSCL5	TXD5 SMOSI5 SSDA5	—
1011b	—	—	—	—	CTS5# RTS5# SS5#
1101b	SSLA1	SSLA2	—	SSLA0	MOSIA

—: 不能设定。

20.2.9 PBn 引脚功能控制寄存器 (PBnPFS) (n=0 ~ 7)

地址 PB0PFS 0008 C198h、PB1PFS 0008 C199h、PB2PFS 0008 C19Ah、PB3PFS 0008 C19Bh、PB4PFS 0008 C19Ch、PB5PFS 0008 C19Dh、PB6PFS 0008 C19Eh、PB7PFS 0008 C19Fh。



位	符号	位名	功能	R/W
b3-b0	PSEL[3:0]	引脚功能选择位	选择外围功能。有关各引脚功能，请参照表 20.16。	R/W
b5-b4	—	保留位	读写值都为“0”。	R/W
b6	ISEL	中断输入功能选择位	0: 不用作 IRQn 输入引脚 1: 用作 IRQn 输入引脚 PB1: IRQ4-DS (100 引脚、80 引脚、64 引脚)	R/W
b7	—	保留位	读写值都为“0”。	R/W

表 20.16 100 引脚 TFLGA、100 引脚 LQFP、80 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚							
	PB0	PB1	PB2	PB3	PB4	PB5	PB6	PB7
0000b (初始值)	Hi-Z							
0001b	MTIC5W	MTIOC0C	—	MTIOC0A	—	MTIOC2A	MTIOC3D	MTIOC3B
0010b	—	MTIOC4C	—	MTIOC4A	—	MTIOC1B	—	—
0101b	—	TMCIO	—	TMO0	—	TMRI1	—	—
0111b	—	—	—	POE3#	—	POE1#	—	—
1010b	—	—	—	—	—	SCK9	RXD9 SMISO9 SSCL9	TXD9 SMOSI9 SSDA9
1011b	RXD6 SMISO6 SSCL6	TXD6 SMOSI6 SSDA6	CTS6# RTS6# SS6#	SCK6	CTS9# RTS9# SS9#	—	—	—
1101b	RSPCKA	—	—	—	—	—	—	—

—: 不能设定。

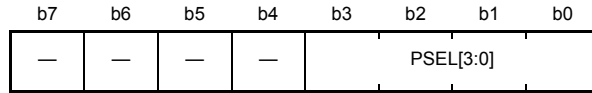
表 20.17 64 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚					
	PB0	PB1	PB3	PB5	PB6	PB7
0000b (初始值)	Hi-Z					
0001b	MTIC5W	MTIOC0C	MTIOC0A	MTIOC2A	MTIOC3D	MTIOC3B
0010b	—	MTIOC4C	MTIOC4A	MTIOC1B	—	—
0101b	—	TMC10	TMO0	TMRI1	—	—
0111b	—	—	POE3#	POE1#	—	—
1010b	—	—	—	SCK9	RXD9 SMISO9 SSCL9	TXD9 SMOSI9 SSDA9
1011b	RXD6 SMISO6 SSCL6	TXD6 SMOSI6 SSDA6	SCK6	—	—	—
1101b	RSPCKA	—	—	—	—	—

—: 不能设定。

20.2.10 PCn 引脚功能控制寄存器 (PCnPFS) (n=0 ~ 7)

地址 PC0PFS 0008 C1A0h、PC1PFS 0008 C1A1h、PC2PFS 0008 C1A2h、PC3PFS 0008 C1A3h、PC4PFS 0008 C1A4h、
PC5PFS 0008 C1A5h、PC6PFS 0008 C1A6h、PC7PFS 0008 C1A7h



复位后的值 0 0 0 0 0 0 0 0

位	符号	位名	功能	R/W
b3-b0	PSEL[3:0]	引脚功能选择位	选择外围功能。有关各引脚功能，请参照表 20.18。	R/W
b7-b4	—	保留位	读写值都为“0”。	R/W

表 20.18 100 引脚 TFLGA、100 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚							
	PC0	PC1	PC2	PC3	PC4	PC5	PC6	PC7
0000b (初始值)	Hi-Z							
0001b	MTIOC3C	MTIOC3A	MTIOC4B	MTIOC4D	MTIOC3D	MTIOC3B	MTIOC3C	MTIOC3A
0010b	—	—	—	—	MTCLKC	MTCLKD	MTCLKA	MTCLKB
0101b	—	—	—	—	TMCI1	TMRI2	TMCI2	TMO2
0111b	—	—	—	—	POE0#	—	—	CACREF
1010b	—	SCK5	RXD5 SMISO5 SSCL5	TXD5 SMOSI5 SSDA5	SCK5	SCK8	RXD8 SMISO8 SSCL8	TXD8 SMOSI8 SSDA8
1011b	CTS5# RTS5# SS5#	—	—	—	CTS8# RTS8# SS8#	—	—	—
1101b	SSLA1	SSLA2	SSLA3	—	SSLA0	RSPCKA	MOSIA	MISOA

—: 不能设定。

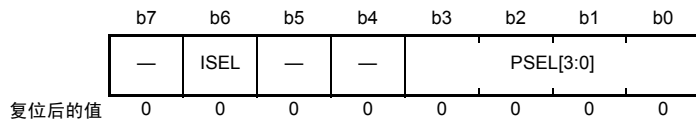
表 20.19 80 引脚 LQFP、64 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚					
	PC2	PC3	PC4	PC5	PC6	PC7
0000b (初始值)	Hi-Z					
0001b	MTIOC4B	MTIOC4D	MTIOC3D	MTIOC3B	MTIOC3C	MTIOC3A
0010b	—	—	MTCLKC	MTCLKD	MTCLKA	MTCLKB
0101b	—	—	TMCI1	TMRI2	TMCI2	TMO2
0111b	—	—	POE0#	—	—	CACREF
1010b	RXD5 SMISO5 SSCL5	TXD5 SMOSI5 SSDA5	SCK5	SCK8	RXD8 SMISO8 SSCL8	TXD8 SMOSI8 SSDA8
1011b	—	—	CTS8# RTS8# SS8#	—	—	—
1101b	SSLA3	—	SSLA0	RSPCKA	MOSIA	MISOA

—: 不能设定。

20.2.11 PDn 引脚功能控制寄存器 (PDnPFS) (n=0 ~ 7)

地址 PD0PFS 0008 C1A8h、PD1PFS 0008 C1A9h、PD2PFS 0008 C1AAh、PD3PFS 0008 C1ABh、PD4PFS 0008 C1ACh、
PD5PFS 0008 C1ADh、PD6PFS 0008 C1AEh、PD7PFS 0008 C1AFh



位	符号	位名	功能	R/W
b3-b0	PSEL[3:0]	引脚功能选择位	选择外围功能。有关各引脚功能，请参照表 20.20。	R/W
b5-b4	—	保留位	读写值都为“0”。	R/W
b6	ISEL	中断输入功能选择位	0: 不用作 IRQn 输入引脚 1: 用作 IRQn 输入引脚 PD0: IRQ0 (100 引脚、80 引脚) PD1: IRQ1 (100 引脚、80 引脚) PD2: IRQ2 (100 引脚、80 引脚) PD3: IRQ3 (100 引脚) PD4: IRQ4 (100 引脚) PD5: IRQ5 (100 引脚) PD6: IRQ6 (100 引脚) PD7: IRQ7 (100 引脚)	R/W
b7	—	保留位	读写值都为“0”。	R/W

表 20.20 100 引脚 TFLGA、100 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚						
	PD1	PD2	PD3	PD4	PD5	PD6	PD7
0000b (初始值)	Hi-Z						
0001b	MTIOC4B	MTIOC4D	—	—	MTIC5W	MTIC5V	MTIC5U
0111b	—	—	POE8#	POE3#	POE2#	POE1#	POE0#

—: 不能设定。

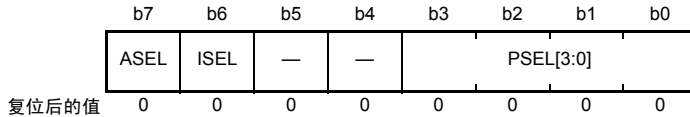
表 20.21 80 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚	
	PD1	PD2
0000b (初始值)	Hi-Z	
0001b	MTIOC4B	MTIOC4D
0111b	—	—

—: 不能设定。

20.2.12 PEn 引脚功能控制寄存器 (PEnPFS) (n=0 ~ 7)

地址 PE0PFS 0008 C1B0h、PE1PFS 0008 C1B1h、PE2PFS 0008 C1B2h、PE3PFS 0008 C1B3h、PE4PFS 0008 C1B4h、
PE5PFS 0008 C1B5h、PE6PFS 0008 C1B6h、PE7PFS 0008 C1B7h



位	符号	位名	功能	R/W
b3-b0	PSEL[3:0]	引脚功能选择位	选择外围功能。有关各引脚功能，请参照表 20.22。	R/W
b5-b4	—	保留位	读写值都为“0”。	R/W
b6	ISEL	中断输入功能选择位	0: 不用作 IRQn 输入引脚 1: 用作 IRQn 输入引脚 PE2: IRQ7-DS (100 引脚、80 引脚、64 引脚) PE5: IRQ5 (100 引脚、80 引脚、64 引脚) PE6: IRQ6 (100 引脚) PE7: IRQ7 (100 引脚)	R/W
b7	ASEL	模拟输入功能选择位	0: 不用作模拟引脚 1: 用作模拟引脚 PE0: AN008 (100 引脚、80 引脚、64 引脚) PE1: AN009、CMPB0 (100 引脚、80 引脚、64 引脚) PE2: AN010、CVREFB0 (100 引脚、80 引脚、64 引脚) PE3: AN011、CMPA1 (100 引脚、80 引脚、64 引脚) PE4: AN012、CMPA2 (100 引脚、80 引脚、64 引脚) PE5: AN013 (100 引脚、80 引脚、64 引脚) PE6: AN014 (100 引脚) PE7: AN015 (100 引脚)	R/W

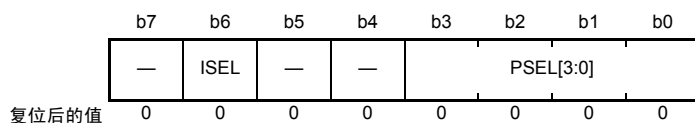
表 20.22 100 引脚 TFLGA、100 引脚 LQFP、80 引脚 LQFP、64 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚					
	PE0	PE1	PE2	PE3	PE4	PE5
0000b (初始值)	Hi-Z					
0001b	—	MTIOC4C	MTIOC4A	MTIOC4B	MTIOC4D	MTIOC4C
0010b	—	—	—	—	MTIOC1A	MTIOC2B
0111b	—	—	—	POE8#	—	—
1100b	SCK12	TXD12 TXDX12 SIOX12 SMOSI12 SSDA12	RXD12 RXDX12 SMISO12 SSCL12	CTS12# RTS12# SS12#	—	—

—: 不能设定。

20.2.13 PHn 引脚功能控制寄存器 (PHnPFS) (n=0 ~ 3)

地址 PH0PFS 0008 C1C8h、PH1PFS 0008 C1C9h、PH2PFS 0008 C1CAh、PH3PFS 0008 C1CBh



位	符号	位名	功能	R/W
b3-b0	PSEL[3:0]	引脚功能选择位	选择外围功能。有关各引脚功能，请参照表 20.23。	R/W
b5-b4	—	保留位	读写值都为“0”。	R/W
b6	ISEL	中断输入功能选择位	0: 不用作 IRQn 输入引脚 1: 用作 IRQn 输入引脚 PH1: IRQ0 (100 引脚、80 引脚、64 引脚) PH2: IRQ1 (100 引脚、80 引脚、64 引脚)	R/W
b7	—	保留位	读写值都为“0”。	R/W

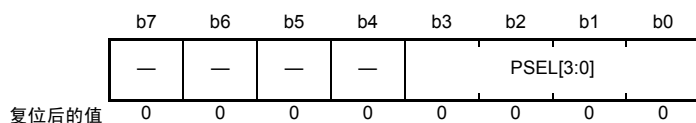
表 20.23 100 引脚 TFLGA、100 引脚 LQFP、80 引脚 LQFP、64 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚			
	PH0	PH1	PH2	PH3
0000b (初始值)	Hi-Z			
0101b	—	TMO0	TMRIO	TMCIO
0111b	CACREF	—	—	—

—: 不能设定。

20.2.14 PJn 引脚功能控制寄存器 (PJnPFS) (n=1、3)

地址 PJ1PFS 0008 C1D1h、PJ3PFS 0008 C1D3h



位	符号	位名	功能	R/W
b3-b0	PSEL[3:0]	引脚功能选择位	选择外围功能。有关各引脚功能，请参照表 20.24。	R/W
b7-b4	—	保留位	读写值都为“0”。	R/W

表 20.24 100 引脚 TFLGA、100 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚	
	PJ1	PJ3
0000b (初始值)	Hi-Z	
0001b	MTIOC3A	MTIOC3C
1011b	—	CTS6# RTS6# SS6#

—: 不能设定。

表 20.25 80 引脚 LQFP 的引脚输入 / 输出功能寄存器的设定

PSEL[3:0] 位的 设定值	引脚
	PJ1
0000b (初始值)	Hi-Z
0001b	MTIOC3A
1011b	—

—: 不能设定。

20.2.15 CS 输出允许寄存器 (PFCSE)

地址 0008 C100h

b7	b6	b5	b4	b3	b2	b1	b0
CS7E	CS6E	CS5E	CS4E	CS3E	CS2E	CS1E	CS0E
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	CS0E	PC7 的 CS0 允许位	0: 将 PC7 设定为 I/O 端口 1: 将 PC7 设定为 CS0# 信号	R/W
b1	CS1E	PC6 的 CS1 允许位	0: 将 PC6 设定为 I/O 端口 1: 将 PC6 设定为 CS1# 信号	R/W
b2	CS2E	P26 的 CS2 允许位	0: 将 P26 设定为 I/O 端口 1: 将 P26 设定为 CS2# 信号	R/W
b3	CS3E	P27 的 CS3 允许位	0: 将 P27 设定为 I/O 端口 1: 将 P27 设定为 CS3# 信号	R/W
b4	CS4E	P24 的 CS0 允许位	0: 将 P24 设定为 I/O 端口 1: 将 P24 设定为 CS0# 信号	R/W
b5	CS5E	P25 的 CS1 允许位	0: 将 P25 设定为 I/O 端口 1: 将 P25 设定为 CS1# 信号	R/W
b6	CS6E	PC5 的 CS2 允许位	0: 将 PC5 设定为 I/O 端口 1: 将 PC5 设定为 CS2# 信号	R/W
b7	CS7E	PC4 的 CS3 允许位	0: 将 PC4 设定为 I/O 端口 1: 将 PC4 设定为 CS3# 信号	R/W

PFCSE 寄存器是选择允许或者禁止输出 CSn# (n=0~3) 的寄存器。

当输出 CSn 信号时, 必须在设定 PFCSE 寄存器后将系统控制寄存器 0 的外部总线允许位 (SYSCR0.EXBE) 位置“1”。有关 SYSCR0.EXBE 位, 请参照“3.2.3 系统控制寄存器 0 (SYSCR0)”。

PC5 的 CS2# 输出和 WAIT# 输入的设置方法如表 20.26 所示。

表 20.26 PC5 的 CS2# 输出和 WAIT# 输入的设置方法

外部总线允许位 (SYSCR0.EXBE) 有效		外部总线控制寄存器 1 (PFBCR1) 的 WAITS[1:0] 位	
		10	10 以外
CS 输出允许寄存器 (PFCSE) 的 CS6E 位	1	(注 1)	CS2# (输出)
	0	WAIT# (输入)	(注 2)

注 1. 不能设定。

注 2. 能进行通用端口和外围模块的转换。

20.2.16 地址输出允许寄存器 0 (PFAOE0)

地址 0008 C104h

b7	b6	b5	b4	b3	b2	b1	b0
A15E	A14E	A13E	A12E	A11E	A10E	A9E	A8E
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	A8E	地址 A8 的输出允许位	0: 禁止输出 A8 1: 允许输出 A8	R/W
b1	A9E	地址 A9 的输出允许位	0: 禁止输出 A9 1: 允许输出 A9	R/W
b2	A10E	地址 A10 的输出允许位	0: 禁止输出 A10 1: 允许输出 A10	R/W
b3	A11E	地址 A11 的输出允许位	0: 禁止输出 A11 1: 允许输出 A11	R/W
b4	A12E	地址 A12 的输出允许位	0: 禁止输出 A12 1: 允许输出 A12	R/W
b5	A13E	地址 A13 的输出允许位	0: 禁止输出 A13 1: 允许输出 A13	R/W
b6	A14E	地址 A14 的输出允许位	0: 禁止输出 A14 1: 允许输出 A14	R/W
b7	A15E	地址 A15 的输出允许位	0: 禁止输出 A15 1: 允许输出 A15	R/W

PFAOE0 寄存器是选择允许或者禁止输出地址的寄存器。

AnE 位 (地址 An 的输出允许位) (n=8 ~ 15)

此位选择允许或者禁止输出地址 (An)。

当输出 An 信号时, 必须在设定 PFAOE0 寄存器后将系统控制寄存器 0 的外部总线允许位 (SYSCR0.EXBE) 位置“1”。有关 SYSCR0.EXBE 位, 请参照“3.2.3 系统控制寄存器 0 (SYSCR0)”。

20.2.17 地址输出允许寄存器 1 (PFAOE1)

地址 0008 C105h

	b7	b6	b5	b4	b3	b2	b1	b0
	A23E	A22E	A21E	A20E	A19E	A18E	A17E	A16E
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	A16E	地址 A16 的输出允许位	0: 禁止输出 A16 1: 允许输出 A16	R/W
b1	A17E	地址 A17 的输出允许位	0: 禁止输出 A17 1: 允许输出 A17	R/W
b2	A18E	地址 A18 的输出允许位	0: 禁止输出 A18 1: 允许输出 A18	R/W
b3	A19E	地址 A19 的输出允许位	0: 禁止输出 A19 1: 允许输出 A19	R/W
b4	A20E	地址 A20 的输出允许位	0: 禁止输出 A20 1: 允许输出 A20	R/W
b5	A21E	地址 A21 的输出允许位	0: 禁止输出 A21 1: 允许输出 A21	R/W
b6	A22E	地址 A22 的输出允许位	0: 禁止输出 A22 1: 允许输出 A22	R/W
b7	A23E	地址 A23 的输出允许位	0: 禁止输出 A23 1: 允许输出 A23	R/W

PFAOE1 寄存器是选择允许或者禁止输出地址的寄存器。

AnE 位 (地址 An 的输出允许位) (n=16 ~ 23)

此位选择允许或者禁止输出地址 (An)。

当输出 An 信号时, 必须在设定 PFAOE1 寄存器后将系统控制寄存器 0 的外部总线允许位 (SYSCR0.EXBE) 位置“1”。有关 SYSCR0.EXBE 位, 请参照“3.2.3 系统控制寄存器 0 (SYSCR0)”。

20.2.18 外部总线控制寄存器 0 (PFBCR0)

地址 0008 C106h

b7	b6	b5	b4	b3	b2	b1	b0
—	WR1BC1E	—	DHE	—	—	—	ADRLE
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	ADRLE	A0 ~ A7 输出允许位	0: 将 PA0 ~ PA7 设定为 I/O 端口 1: 将 PA0 ~ PA7 设定为外部地址总线 A0 ~ A7	R/W
b3-b1	—	保留位	读写值都为“0”。	R/W
b4	DHE	D8 ~ D15 输出允许位	0: 将 PE0 ~ PE7 设定为 I/O 端口 1: 将 PE0 ~ PE7 设定为外部数据总线 D8 ~ D15	R/W
b5	—	保留位	读写值都为“0”。	R/W
b6	WR1BC1E	WR1#/BC1# 输出允许位	0: 将 P51 设定为 I/O 端口 1: 将 P51 设定为 WR1# 或者 BC1#	R/W
b7	—	保留位	读写值都为“0”。	R/W

PFBCR0 寄存器是控制外部总线的输入 / 输出引脚的寄存器。

ADRLE 位 (A0 ~ A7 输出允许位)

此位选择允许或者禁止地址总线 (A0 ~ A7) 的输出。

DHE 位 (D8 ~ D15 输出允许位)

此位选择允许或者禁止数据 (D8 ~ D15) 的输出。

必须设定为 CSi 控制寄存器的外部总线宽度选择位 (CSnCR.BSIZE[1:0]) 设定的外部总线宽度。如果在设定为 16 位外部总线的状态下将 DHE 位置“0”，就无法保证运行。有关 CSnCR.BSIZE[1:0] 位，请参照“15.3.1 CSn 控制寄存器 (CSnCR) (n=0 ~ 3)”。

WR1BC1E 位 (WR1#/BC1# 输出允许位)

此位选择允许或者禁止 WR1#/BC1# 的输出。

如果允许数据 D8 ~ D15 的输出和 WR1#/BC1# 的输出，就必须在设定 PFBCR0 寄存器后将系统控制寄存器 0 的外部总线允许位 (SYSCR0.EXBE) 位置“1”。有关 SYSCR0.EXBE 位，请参照“3.2.3 系统控制寄存器 0 (SYSCR0)”。

P51 的 WR1#/BC1# 输出和 WAIT# 输入的设置方法如表 20.27 所示。

表 20.27 P51 的 WR1#/BC1# 输出和 WAIT# 输入的设置方法

外部总线允许位 (SYSCR0.EXBE) 有效		外部总线控制寄存器 1 (PFBCR1) 的 WAITS[1:0] 位	
		11	11 以外
外部总线控制寄存器 0 (PFBCR0) 的 WR1BC1E 位	1	(注 1)	WR1#/BC1# (输出)
	0	WAIT# (输入)	(注 2)

注 1. 不能设定。

注 2. 能进行通用端口和外围模块的转换。

20.2.19 外部总线控制寄存器 1 (PFBCR1)

地址 0008 C107h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	ALEOE	WAITS[1:0]	
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b1-b0	WAITS[1:0]	WAIT 选择位	b1 b0 0 x: 将 P55 设定为 WAIT# 输入引脚 1 0: 将 PC5 设定为 WAIT# 输入引脚 1 1: 将 P51 设定为 WAIT# 输入引脚	R/W
b2	ALEOE	ALE 输出允许位	0: 将 P54 设定为 I/O 端口 1: 将 P54 设定为 ALE 引脚	R/W
b7-b3	—	保留位	读写值都为“0”。	R/W

x: Don't care

PFBCR1 寄存器是控制外部总线的输入 / 输出引脚的寄存器。

WAITS[1:0] (WAIT 选择位)

这些位选择 WAIT# 的输入引脚。

ALEOE (ALE 输出允许位)

此位选择允许或者禁止 ALE 的输出。

当输入 WAIT# 信号时, 必须在设定 PFBCR1 寄存器后将系统控制寄存器 0 的外部总线允许位 (SYSCR0.EXBE) 置“1”。有关 SYSCR0.EXBE 位, 请参照“3.2.3 系统控制寄存器 0 (SYSCR0)”。

PC5 的 CS2# 输出和 WAIT# 输入的设定方法如表 20.28 所示, P51 输出的 WR1#/BC1# 输出和 WAIT# 输入的设定方法如表 20.29 所示。

表 20.28 PC5 的 CS2# 输出和 WAIT# 输入的设定方法

外部总线允许位 (SYSCR0.EXBE) 有效		外部总线控制寄存器 1 (PFBCR1) 的 WAITS[1:0] 位	
		10	10 以外
CS 输出允许寄存器 (PFCSE) 的 CS6E 位	1	(注 1)	CS2# (输出)
	0	WAIT# (输入)	(注 2)

表 20.29 P51 的 WR1#/BC1# 输出和 WAIT# 输入的设定方法

外部总线允许位 (SYSCR0.EXBE) 有效		外部总线控制寄存器 1 (PFBCR1) 的 WAITS[1:0] 位	
		11	11 以外
外部总线控制寄存器 0 (PFBCR0) 的 WR1BC1E 位	1	(注 1)	WR1#/BC1# (输出)
	0	WAIT# (输入)	(注 2)

注 1. 不能设定。

注 2. 能进行通用端口和外围模块的转换。

20.3 外部总线接口的设定方法

各端口的外部总线接口的设定方法如表 20.30 所示。

对应的外部总线接口控制寄存器的详细内容请参照各寄存器的说明。

在使用外部总线接口时，必须在设定外部总线接口控制寄存器后，将外部总线允许位 (SYSCR0.EXBE) 置“1”。

表 20.30 外部总线接口的设定方法 (1/2)

端口	模块名	输出信号名	外部总线接口控制寄存器	系统控制寄存器 0 (SYSCR0)
P24	外部总线 (CS)	CS0#	PFCSE.CS4E=1	SYSCR0.EXBE=1
P25	外部总线 (CS)	CS1#	PFCSE.CS5E=1	SYSCR0.EXBE=1
P26	外部总线 (CS)	CS2#	PFCSE.CS2E=1	SYSCR0.EXBE=1
P27	外部总线 (CS)	CS3#	PFCSE.CS3E=1	SYSCR0.EXBE=1
P50	外部总线	WR# WR0#		SYSCR0.EXBE=1
P51	外部总线	WR1# BC1#	PFBCR0.WR1BC1E=1 PFBCR1.WAITS[1:0]=00 or 01 or 10	SYSCR0.EXBE=1
	外部总线 (WAIT)	WAIT#	PFBCR0.WR1BC1E=0 PFBCR1.WAITS[1:0]=11	SYSCR0.EXBE=1
P52	外部总线	RD#		SYSCR0.EXBE=1
P53	外部总线	BCLK		SYSCR0.EXBE=1
P54	外部总线	ALE	PFBCR1.ALEOE=1	SYSCR0.EXBE=1
P55	外部总线 (WAIT)	WAIT#	PFBCR1.WAITS[1:0]=00 or 01	SYSCR0.EXBE=1
PA0	外部总线 (地址)	A0 BC0#	PFBCR0.ADRLE=1	SYSCR0.EXBE=1
PA1	外部总线 (地址)	A1	PFBCR0.ADRLE=1	SYSCR0.EXBE=1
PA2	外部总线 (地址)	A2	PFBCR0.ADRLE=1	SYSCR0.EXBE=1
PA3	外部总线 (地址)	A3	PFBCR0.ADRLE=1	SYSCR0.EXBE=1
PA4	外部总线 (地址)	A4	PFBCR0.ADRLE=1	SYSCR0.EXBE=1
PA5	外部总线 (地址)	A5	PFBCR0.ADRLE=1	SYSCR0.EXBE=1
PA6	外部总线 (地址)	A6	PFBCR0.ADRLE=1	SYSCR0.EXBE=1
PA7	外部总线 (地址)	A7	PFBCR0.ADRLE=1	SYSCR0.EXBE=1
PB0	外部总线 (地址)	A8	PFAOE0.A8E=1	SYSCR0.EXBE=1
PB1	外部总线 (地址)	A9	PFAOE0.A9E=1	SYSCR0.EXBE=1
PB2	外部总线 (地址)	A10	PFAOE0.A10E=1	SYSCR0.EXBE=1
PB3	外部总线 (地址)	A11	PFAOE0.A11E=1	SYSCR0.EXBE=1
PB4	外部总线 (地址)	A12	PFAOE0.A12E=1	SYSCR0.EXBE=1
PB5	外部总线 (地址)	A13	PFAOE0.A13E=1	SYSCR0.EXBE=1
PB6	外部总线 (地址)	A14	PFAOE0.A14E=1	SYSCR0.EXBE=1
PB7	外部总线 (地址)	A15	PFAOE0.A15E=1	SYSCR0.EXBE=1
PC0	外部总线 (地址)	A16	PFAOE1.A16E=1	SYSCR0.EXBE=1
PC1	外部总线 (地址)	A17	PFAOE1.A17E=1	SYSCR0.EXBE=1
PC2	外部总线 (地址)	A18	PFAOE1.A18E=1	SYSCR0.EXBE=1
PC3	外部总线 (地址)	A19	PFAOE1.A19E=1	SYSCR0.EXBE=1

表 20.30 外部总线接口的设定方法 (2/2)

端口	模块名	输出信号名	外部总线接口控制寄存器	系统控制寄存器 0 (SYSCR0)
PC4	外部总线 (地址)	A20	PFAOE1.A20E=1 PFCSE.CS7E=0	SYSCR0.EXBE=1
	外部总线 (CS)	CS3#	PFAOE1.A20E=0 PFCSE.CS7E=1	SYSCR0.EXBE=1
PC5	外部总线 (地址)	A21	PFAOE1.A21E=1 PFCSE.CS6E=0 PFBCR1.WAITS[1:0]=00 or 10 or 11	SYSCR0.EXBE=1
	外部总线 (CS)	CS2#	PFAOE1.A21E=0 PFCSE.CS6E=1 PFBCR1.WAITS[1:0]=00 or 10 or 11	SYSCR0.EXBE=1
	外部总线 (WAIT)	WAIT#	PFAOE1.A21E=0 PFCSE.CS6E=0 PFBCR1.WAITS[1:0]=10	SYSCR0.EXBE=1
PC6	外部总线 (地址)	A22	PFAOE1.A22E=1 PFCSE.CS1E=0	SYSCR0.EXBE=1
	外部总线 (CS)	CS1#	PFCSE.CS1E=1 PFAOE1.A22E=0	SYSCR0.EXBE=1
PC7	外部总线 (地址)	A23	PFAOE1.A23E=1 PFCSE.CS0E=0	SYSCR0.EXBE=1
	外部总线 (CS)	CS0#	PFCSE.CS0E=1 PFAOE1.A23E=0	SYSCR0.EXBE=1
PD0	外部总线 (数据)	D0		SYSCR0.EXBE=1
PD1	外部总线 (数据)	D1		SYSCR0.EXBE=1
PD2	外部总线 (数据)	D2		SYSCR0.EXBE=1
PD3	外部总线 (数据)	D3		SYSCR0.EXBE=1
PD4	外部总线 (数据)	D4		SYSCR0.EXBE=1
PD5	外部总线 (数据)	D5		SYSCR0.EXBE=1
PD6	外部总线 (数据)	D6		SYSCR0.EXBE=1
PD7	外部总线 (数据)	D7		SYSCR0.EXBE=1
PE0	外部总线 (数据)	D8	PFBCR0.DHE=1	SYSCR0.EXBE=1
PE1	外部总线 (数据)	D9	PFBCR0.DHE=1	SYSCR0.EXBE=1
PE2	外部总线 (数据)	D10	PFBCR0.DHE=1	SYSCR0.EXBE=1
PE3	外部总线 (数据)	D11	PFBCR0.DHE=1	SYSCR0.EXBE=1
PE4	外部总线 (数据)	D12	PFBCR0.DHE=1	SYSCR0.EXBE=1
PE5	外部总线 (数据)	D13	PFBCR0.DHE=1	SYSCR0.EXBE=1
PE6	外部总线 (数据)	D14	PFBCR0.DHE=1	SYSCR0.EXBE=1
PE7	外部总线 (数据)	D15	PFBCR0.DHE=1	SYSCR0.EXBE=1

20.4 使用时的注意事项

20.4.1 引脚输入 / 输出功能的设定步骤

必须按照以下步骤设定引脚输入 / 输出功能。

1. 通过将对应引脚的端口方向寄存器 (PDR) 和端口模式寄存器 (PMR) 置“0”，设定为通用输入功能。
2. 通过设定写保护寄存器 (PWPR)，将 Pmn 引脚功能控制寄存器 (PmnPFS) 的写操作置为有效 (m=0~5、A~E、H、J、n=0~7)。
3. 通过 PmnPFS.PSEL[3:0] 位设定引脚输入/输出功能。
4. 必须通过将 PWPR.PFSWE 位置“0”，禁止写 PmnPFS 寄存器。
5. 根据需要，将 PMR 寄存器置“1”，切换到所选择的引脚输入/输出功能。

20.4.2 设定 MPC 寄存器时的注意事项

1. 必须在对应引脚的 PMR 寄存器为“0”的状态下设定 Pmn 引脚功能控制寄存器 (PmnPFS)。如果在 PMR 寄存器为“1”的状态下设定 PmnPFS 寄存器，就有可能在输入功能的情况下输入意外的边沿，在输出功能的情况下输出意外的脉冲。
2. 必须设定 PmnPFS 寄存器能设定的功能。如果设定没有指定的功能，就不保证运行。
3. 不能设定为通过 MPC 给多个引脚分配相同的功能。
4. 端口 4、E 也兼用 A/D 转换器的模拟输入引脚功能。当用作模拟输入引脚时，为了不降低精度，必须在将端口模式寄存器 (PMR) 和端口方向寄存器 (PDR) 的对应位都置“0”后，将对应引脚设定为通用输入并且将 PmnPFS.ASEL 位置“1”。
5. 复位后，时间捕捉控制寄存器 y (RTCCRy) (y=0~2) 的时间捕捉事件输入引脚允许位 (TCEN) 的初始值为不定值。为了禁止不需要的输入，必须将此位置“0”。
6. 被多路复用的引脚的端口模式寄存器 (PMR)、端口方向寄存器 (PDR) 和 Pmn 引脚功能控制寄存器 (PmnPFS) 的设定和注意事项如表 20.31 所示。

在 ASEL 位为“0”时读引脚的状态。

在 PMR.Bn 位为“0”时更改 PSEL[3:0] 位。

表 20.31 寄存器的设定

项目	PMR.Bn	PDR.Bn	PmnPFS			注意事项
			ASEL	ISEL	PSEL[3:0]	
解除复位后	0	0	0	0	0000b	在解除复位后，用作通用输入端口。
通用输入端口	0	0	0	0/1	x	当与中断输入并用时，必须将 ISEL 位置“1”。
通用输出端口	0	1	0	0	x	
外围功能	1	x	0	0/1	外围功能 (参照表 20.3 ~ 表 20.19)	当与中断输入并用时，必须将 ISEL 位置“1”。
中断输入	0	0	0	1	x	
NMI	x	x	x	x (注1)	x	不需要设定寄存器。
模拟输入 / 输出	0	0	1	x (注1)	x	为了将输出缓冲器置为 OFF，必须设定为通用输入端口。
RTC 的时间捕捉事件输入	0	0	x	0/1	x	为了将输出缓冲器置为 OFF，必须设定为通用输入端口。
外部总线	0	x	0	0	x	为了防止信号发生冲突，必须将 PMB.Bj 位置“0”，并且将 RIIC 专用缓冲器置为 OFF。
EXTAL/XTAL	0	0	x	x (注1)	x	为了将输出缓冲器置为 OFF，必须设定为通用输入端口。
XCIN/XCOUT	0	0	x	x (注1)	x	为了将输出缓冲器置为 OFF，必须设定为通用输入端口。

x: 不需要设定。

注 1. 即使将 PmnPFS.ISEL 位置“1”，也不用作 IRQn 输入引脚。

20.4.3 使用模拟功能时的注意事项

1. 在使用模拟功能时，必须将端口模式寄存器 (PMR) 和端口方向寄存器 (PDR) 的对应位都置“0”并且将对应的引脚设定为通用输入端口，然后将 Pmn 引脚功能控制寄存器 (PmnPFS) 的 ASEL 位置“1”。

21. 多功能定时器脉冲单元 2 (MTU2a)

21.1 概要

RX210 群内置由 6 个通道 (MTU0 ~ MTU5) 的 16 位定时器构成的多功能定时器脉冲单元 2 (MTU)。MTU 的规格和功能一览表分别如表 21.1 和表 21.2 所示, MTU 的框图如图 21.1 所示。

表 21.1 MTU 的规格

项目	内容
脉冲输入 / 输出	最多 16 个
脉冲输入	3 个
计数时钟	各通道有 8 种或者 7 种 (MTU5 有 4 种)。
能设定的运行	<p>【MTU0 ~ 4】</p> <ul style="list-style-type: none"> 通过比较匹配进行波形输出。 输入捕捉功能 (噪声滤波器设定功能) 计数器清除运行 同时写多个定时器的计数器 (TCNT)。 通过比较匹配或者输入捕捉进行同时清除。 通过计数器的同步运行进行各寄存器的同步输入 / 输出。 通过和同步运行的组合进行最多 12 相的 PWM 输出。 <p>【MTU0、3、4】</p> <ul style="list-style-type: none"> 能设定缓冲运行。 能设定使用互补 PWM 或者复位同步 PWM 的 AC 同步马达 (无刷 DC 马达) 驱动模式, 并且可选择 2 种 (斩波、电平) 波形输出。 <p>【MTU1 和 MTU2】</p> <ul style="list-style-type: none"> 能分别设定相位计数模式。 级联运行 <p>【MTU3 和 MTU4】</p> <ul style="list-style-type: none"> 能通过联动运行输出互补 PWM 或者复位 PWM 的正负 3 相共 6 相。 <p>【MTU5】</p> <ul style="list-style-type: none"> 用于补偿死区时间的计数器功能。
互补 PWM 模式	<ul style="list-style-type: none"> 计数器波峰 / 波谷的中断 A/D 转换器的转换开始触发减少功能
中断源	28 种
缓冲运行	寄存器数据的自动传送
触发生成	能生成 A/D 转换器的转换开始触发。
低功耗功能	能设定为模块停止状态。

表 21.2 MTU 的功能一览表 (1/2)

项目	MTU0	MTU1	MTU2	MTU3	MTU4	MTU5
计数时钟	PCLK/1 PCLK/4 PCLK/16 PCLK/64 MTCLKA MTCLKB MTCLKC MTCLKD	PCLK/1 PCLK/4 PCLK/16 PCLK/64 PCLK/256 MTCLKA MTCLKB	PCLK/1 PCLK/4 PCLK/16 PCLK/64 PCLK/1024 MTCLKA MTCLKB MTCLKC	PCLK/1 PCLK/4 PCLK/16 PCLK/64 PCLK/256 PCLK/1024 MTCLKA MTCLKB	PCLK/1 PCLK/4 PCLK/16 PCLK/64 PCLK/256 PCLK/1024 MTCLKA MTCLKB	PCLK/1 PCLK/4 PCLK/16 PCLK/64
通用寄存器 (TGR)	TGRA TGRB TGRE	TGRA TGRB	TGRA TGRB	TGRA TGRB	TGRA TGRB	TGRU TGRV TGRW
通用寄存器 / 缓冲寄存器	TGRC TGRD TGRF	—	—	TGRC TGRD	TGRC TGRD	—
输入 / 输出引脚	MTIOC0A MTIOC0B MTIOC0C MTIOC0D	MTIOC1A MTIOC1B	MTIOC2A MTIOC2B	MTIOC3A MTIOC3B MTIOC3C MTIOC3D	MTIOC4A MTIOC4B MTIOC4C MTIOC4D	输入引脚 MTIC5U MTIC5V MTIC5W
计数器清除功能	TGR 的比较匹 配或者输入捕捉	TGR 的比较匹 配或者输入捕捉	TGR 的比较匹 配或者输入捕捉	TGR 的比较匹 配或者输入捕捉	TGR 的比较匹 配或者输入捕捉	TGR 的比较匹 配或者输入捕捉
比较匹配 输出	Low 电平输出	○	○	○	○	—
	High 电平输出	○	○	○	○	—
	交替输出	○	○	○	○	—
输入捕捉功能	○	○	○	○	○	○
同步运行	○	○	○	○	○	—
PWM 模式 1	○	○	○	○	○	—
PWM 模式 2	○	○	○	—	—	—
互补 PWM 模式	—	—	—	○	○	—
复位同步 PWM	—	—	—	○	○	—
AC 同步马达驱动模式	○	—	—	○	○	—
相位计数模式	—	○	○	—	—	—
缓冲运行	○	—	—	○	○	—
用于补偿死区时间的 计数器功能	—	—	—	—	—	○
DMAC 的启动	TGRA 的比较匹 配或者输入捕捉	TGRA 的比较匹 配或者输入捕捉	TGRA 的比较匹 配或者输入捕捉	TGRA 的比较匹 配或者输入捕捉	TGRA 的比较匹 配或者输入捕捉	—
DTC 的启动	TGR 的比较匹 配或者输入捕捉	TGR 的比较匹 配或者输入捕捉	TGR 的比较匹 配或者输入捕捉	TGR 的比较匹 配或者输入捕捉	TGR 的比较匹 配或者输入捕捉、 TCNT 的上溢 / 下 溢	TGR 的比较匹 配或者输入捕捉
A/D 转换开始触发	TGRA 的比较匹 配或者输入捕 捉、TGRB 的 比较匹配或者输 入捕捉、TGRE 的比较匹配、 TGRF 的比较匹 配	TGRA 的比较匹 配或者输入捕捉	TGRA 的比较匹 配或者输入捕捉	TGRA 的比较匹 配或者输入捕捉	TGRA 的比较匹 配或者输入捕捉、互 补 PWM 模式中的 TCNT 下溢 (波 谷)	—

表 21.2 MTU 的功能一览表 (2/2)

项目	MTU0	MTU1	MTU2	MTU3	MTU4	MTU5
中断源	7 个源 <ul style="list-style-type: none"> • 比较匹配 / 输入捕捉 0A • 比较匹配 / 输入捕捉 0B • 比较匹配 / 输入捕捉 0C • 比较匹配 / 输入捕捉 0D • 比较匹配 0E • 比较匹配 0F • 上溢 	4 个源 <ul style="list-style-type: none"> • 比较匹配 / 输入捕捉 1A • 比较匹配 / 输入捕捉 1B • 上溢 • 下溢 	4 个源 <ul style="list-style-type: none"> • 比较匹配 / 输入捕捉 2A • 比较匹配 / 输入捕捉 2B • 上溢 • 下溢 	5 个源 <ul style="list-style-type: none"> • 比较匹配 / 输入捕捉 3A • 比较匹配 / 输入捕捉 3B • 比较匹配 / 输入捕捉 3C • 比较匹配 / 输入捕捉 3D • 上溢 	5 个源 <ul style="list-style-type: none"> • 比较匹配 / 输入捕捉 4A • 比较匹配 / 输入捕捉 4B • 比较匹配 / 输入捕捉 4C • 比较匹配 / 输入捕捉 4D • 上溢 / 下溢 	3 个源 <ul style="list-style-type: none"> • 比较匹配 / 输入捕捉 5U • 比较匹配 / 输入捕捉 5V • 比较匹配 / 输入捕捉 5W
事件链接功能 (输出)	—	4 个源 <ul style="list-style-type: none"> • 比较匹配 1A • 比较匹配 1B • 上溢 • 下溢 	4 个源 <ul style="list-style-type: none"> • 比较匹配 2A • 比较匹配 2B • 上溢 • 下溢 	6 个源 <ul style="list-style-type: none"> • 比较匹配 3A • 比较匹配 3B • 比较匹配 3C • 比较匹配 3D • 上溢 • 下溢 	6 个源 <ul style="list-style-type: none"> • 比较匹配 4A • 比较匹配 4B • 比较匹配 4C • 比较匹配 4D • 上溢 • 下溢 	—
事件链接功能 (输入)	—	1. 开始计数 2. 输入捕捉运行 (捕捉到 TRGA) 3. 重新开始计数	1. 开始计数 2. 输入捕捉运行 (捕捉到 TRGA) 3. 重新开始计数	1. 开始计数 2. 输入捕捉运行 (捕捉到 TRGA) 3. 重新开始计数	1. 开始计数 2. 输入捕捉运行 (捕捉到 TRGA) 3. 重新开始计数	—
A/D 转换的开始请求延迟功能	—	—	—	—	• 在 TADCORA 和 TCNT 匹配时请求开始 A/D 转换或者 在 TADCORB 和 TCNT 匹配时请求开始 A/D 转换	—
中断减少功能	—	—	—	• 减少 TGRA 比较匹配中断	• 减少 TCIV 中断	—
模块停止	MSTPCRA.MSTPA9 (注 1)					

○: 能

—: 不能

注 1. 有关模块停止的详细内容, 请参照“11. 低功耗功能”。

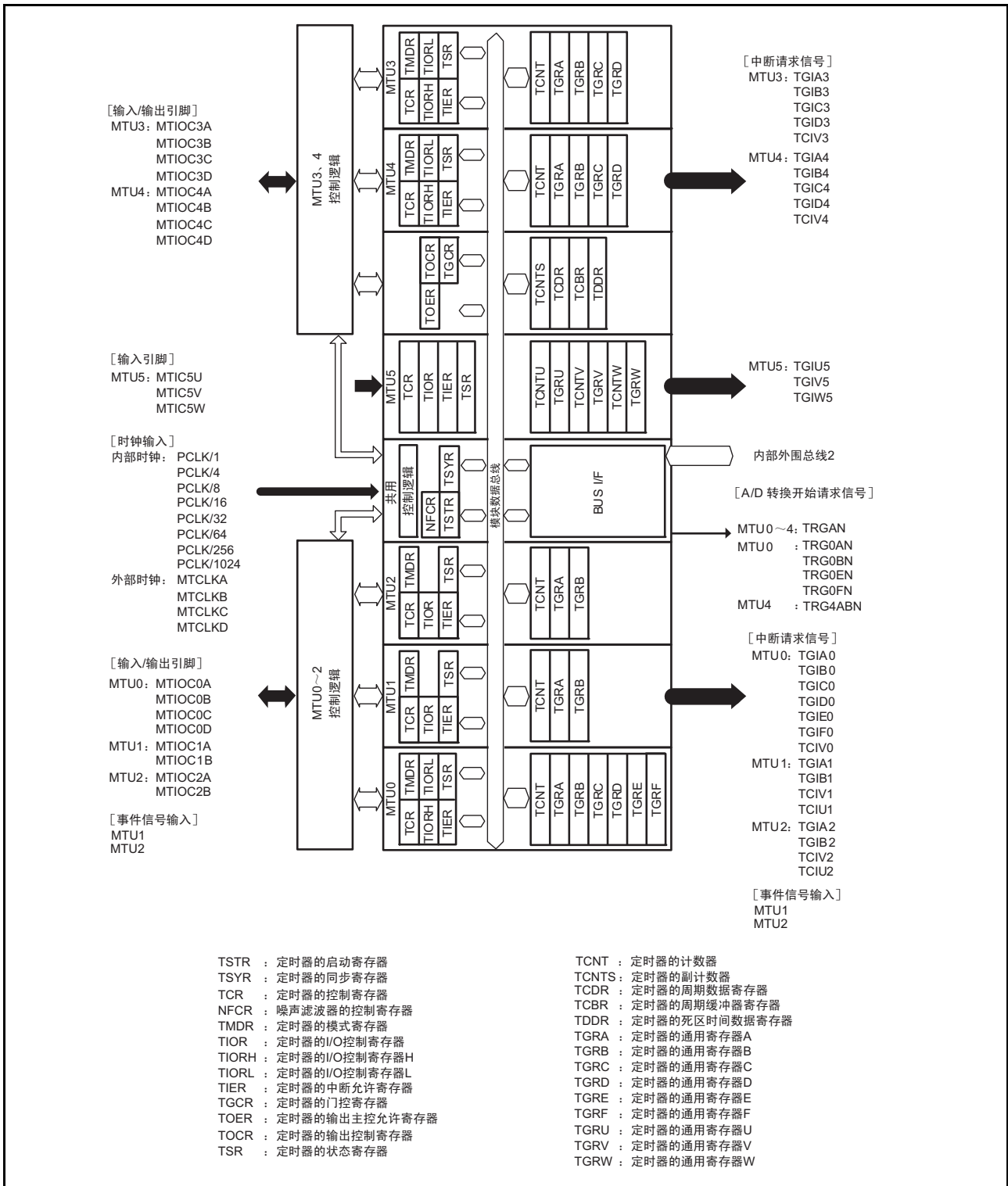


图 21.1 MTU 的框图

MTU 使用的输入 / 输出引脚如表 21.3 所示。

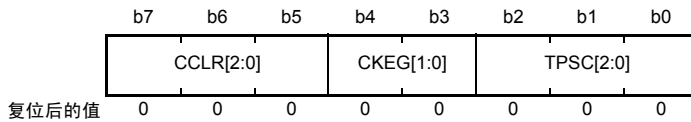
表 21.3 MTU 的输入 / 输出引脚

模块符号	引脚名	输入 / 输出	功能
MTU	MTCLKA	输入	外部时钟 A 的输入引脚 (MTU1 的相位计数模式的 A 相输入)
	MTCLKB	输入	外部时钟 B 的输入引脚 (MTU1 的相位计数模式的 B 相输入)
	MTCLKC	输入	外部时钟 C 的输入引脚 (MTU2 的相位计数模式的 A 相输入)
	MTCLKD	输入	外部时钟 D 的输入引脚 (MTU2 的相位计数模式的 B 相输入)
MTU0	MTIOC0A	输入 / 输出	TGRA0 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	MTIOC0B	输入 / 输出	TGRB0 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	MTIOC0C	输入 / 输出	TGRC0 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	MTIOC0D	输入 / 输出	TGRD0 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
MTU1	MTIOC1A	输入 / 输出	TGRA1 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	MTIOC1B	输入 / 输出	TGRB1 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
MTU2	MTIOC2A	输入 / 输出	TGRA2 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	MTIOC2B	输入 / 输出	TGRB2 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
MTU3	MTIOC3A	输入 / 输出	TGRA3 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	MTIOC3B	输入 / 输出	TGRB3 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	MTIOC3C	输入 / 输出	TGRC3 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	MTIOC3D	输入 / 输出	TGRD3 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
MTU4	MTIOC4A	输入 / 输出	TGRA4 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	MTIOC4B	输入 / 输出	TGRB4 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	MTIOC4C	输入 / 输出	TGRC4 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
	MTIOC4D	输入 / 输出	TGRD4 的输入捕捉的输入引脚 / 输出比较的输出引脚 / PWM 的输出引脚
MTU5	MTIC5U	输入	TGRU5 的输入捕捉的输入引脚 / 外部脉冲的输入引脚
	MTIC5V	输入	TGRV5 的输入捕捉的输入引脚 / 外部脉冲的输入引脚
	MTIC5W	输入	TGRW5 的输入捕捉的输入引脚 / 外部脉冲的输入引脚

21.2 寄存器说明

21.2.1 定时器的控制寄存器 (TCR)

地址 MTU0.TCR 0008 8700h、MTU1.TCR 0008 8780h、MTU2.TCR 0008 8800h、MTU3.TCR 0008 8600h、MTU4.TCR 0008 8601h、MTU5.TCRU 0008 8884h、MTU5.TCRV 0008 8894h、MTU5.TCRW 0008 88A4h



位	符号	位名	功能	R/W
b2-b0	TPSC[2:0]	定时器的预分频器选择位	请参照表 21.6 ~ 表 21.10。	R/W
b4-b3	CKEG[1:0]	时钟边沿选择位	b4 b3 0 0: 在上升沿进行计数 0 1: 在下降沿进行计数 1 x: 在双边沿进行计数	R/W
b7-b5	CCLR[2:0]	计数器清除位	请参照表 21.4 和表 21.5。	R/W

x: Don't care

在 MTU 中，MTU0 ~ MTU4 各有 1 个，MTU5 有 3 个 TCRU/V/W，共计 8 个 TCR 寄存器。

TCR 寄存器是控制各通道 TCNT 的 8 位可读写寄存器。必须在 TCNT 停止计数的状态下设定 TCR 寄存器。

TPSC[2:0] 位 (定时器的预分频器选择位)

这些位选择 TCNT 计数器的时钟。各通道能独立选择时钟源，详细内容请参照表 21.6 ~ 表 21.10。

CKEG[1:0] 位 (时钟边沿选择位)

这些位选择输入时钟的边沿。如果在双边沿对内部时钟进行计数，输入时钟的周期就为 1/2 (例如: PCLK/4 的双边沿 = PCLK/2 的上升沿)。在 MTU1 和 MTU2 中使用相位计数模式时，忽视此设定而优先设定相位计数模式。内部时钟的边沿选择在输入时钟为 PCLK/4 或者慢于 PCLK/4 时有效。如果选择 PCLK/1 或者其他通道的上溢/下溢作为输入时钟，虽然能写值，但是在运行时为初始值。

CCLR[2:0] 位 (计数器清除位)

这些位选择 TCNT 计数器的清除源，详细内容请参照表 21.4 和表 21.5。

表 21.4 CCLR[2:0] (MUT0、MUT3 和 MUT4)

通道	bit7	bit6	bit5	说明
	CCLR2	CCLR1	CCLR0	
MTU0	0	0	0	禁止清除 TCNT。
MTU3	0	0	1	在发生 TGRA 的比较匹配 / 输入捕捉时清除 TCNT。
MTU4	0	1	0	在发生 TGRB 的比较匹配 / 输入捕捉时清除 TCNT。
	0	1	1	在清除正在进行同步清除或者同步运行的其他通道的计数器时清除 TCNT (注 1)。
	1	0	0	禁止清除 TCNT。
	1	0	1	在发生 TGRC 的比较匹配 / 输入捕捉时清除 TCNT (注 2)。
	1	1	0	在发生 TGRD 的比较匹配 / 输入捕捉时清除 TCNT (注 2)。
	1	1	1	在清除正在进行同步清除或者同步运行的其他通道的计数器时清除 TCNT (注 1)。

注 1. 通过将 TSYR.SYNC 位置“1”，设定为同步运行。

注 2. 在将 TGRC 或者 TGRD 用作缓冲寄存器时，因为优先设定缓冲寄存器而且不发生比较匹配 / 输入捕捉，所以不清除 TCNT。

表 21.5 CCLR[2:0] (MTU1 和 MTU2)

通道	bit7	bit6	bit5	说明
	保留位 (注2)	CCLR1	CCLR0	
MTU1	0	0	0	禁止清除 TCNT。
MTU2	0	0	1	在发生 TGRA 的比较匹配 / 输入捕捉时清除 TCNT。
	0	1	0	在发生 TGRB 的比较匹配 / 输入捕捉时清除 TCNT。
	0	1	1	在清除正在进行同步清除或者同步运行的其他通道的计数器时清除 TCNT (注 1)。

注 1. 通过将 TSYR.SYNC 位置“1”，设定为同步运行。

注 2. 在 MTU1 和 MTU2 中，b7 为保留位，读写值都为“0”。

表 21.6 TPSC[2:0] (MTU0)

通道	bit2	bit1	bit0	说明
	TPSC2	TPSC1	TPSC0	
MTU0	0	0	0	内部时钟：通过 PCLK/1 进行计数
	0	0	1	内部时钟：通过 PCLK/4 进行计数
	0	1	0	内部时钟：通过 PCLK/16 进行计数
	0	1	1	内部时钟：通过 PCLK/64 进行计数
	1	0	0	外部时钟：通过 MTCLKA 引脚输入进行计数
	1	0	1	外部时钟：通过 MTCLKB 引脚输入进行计数
	1	1	0	外部时钟：通过 MTCLKC 引脚输入进行计数
	1	1	1	外部时钟：通过 MTCLKD 引脚输入进行计数

表 21.7 TPSC[2:0] (MTU1)

通道	bit2	bit1	bit0	说明
	TPSC2	TPSC1	TPSC0	
MTU1	0	0	0	内部时钟: 通过 PCLK/1 进行计数
	0	0	1	内部时钟: 通过 PCLK/4 进行计数
	0	1	0	内部时钟: 通过 PCLK/16 进行计数
	0	1	1	内部时钟: 通过 PCLK/64 进行计数
	1	0	0	外部时钟: 通过 MTCLKA 引脚输入进行计数
	1	0	1	外部时钟: 通过 MTCLKB 引脚输入进行计数
	1	1	0	内部时钟: 通过 PCLK/256 进行计数
	1	1	1	通过 MTU2.TCNT 的上溢 / 下溢进行计数。

注. 在 MTU1 为相位计数模式时, 此设定无效。

表 21.8 TPSC[2:0] (MTU2)

通道	bit2	bit1	bit0	说明
	TPSC2	TPSC1	TPSC0	
MTU2	0	0	0	内部时钟: 通过 PCLK/1 进行计数
	0	0	1	内部时钟: 通过 PCLK/4 进行计数
	0	1	0	内部时钟: 通过 PCLK/16 进行计数
	0	1	1	内部时钟: 通过 PCLK/64 进行计数
	1	0	0	外部时钟: 通过 MTCLKA 引脚输入进行计数
	1	0	1	外部时钟: 通过 MTCLKB 引脚输入进行计数
	1	1	0	外部时钟: 通过 MTCLKC 引脚输入进行计数
	1	1	1	内部时钟: 通过 PCLK/1024 进行计数

注. 在 MTU2 为相位计数模式时, 此设定无效。

表 21.9 TPSC[2:0] (MTU3 和 MTU4)

通道	bit2	bit1	bit0	说明
	TPSC2	TPSC1	TPSC0	
MTU3 MTU4	0	0	0	内部时钟: 通过 PCLK/1 进行计数
	0	0	1	内部时钟: 通过 PCLK/4 进行计数
	0	1	0	内部时钟: 通过 PCLK/16 进行计数
	0	1	1	内部时钟: 通过 PCLK/64 进行计数
	1	0	0	内部时钟: 通过 PCLK/256 进行计数
	1	0	1	内部时钟: 通过 PCLK/1024 进行计数
	1	1	0	外部时钟: 通过 MTCLKA 引脚输入进行计数
	1	1	1	外部时钟: 通过 MTCLKB 引脚输入进行计数

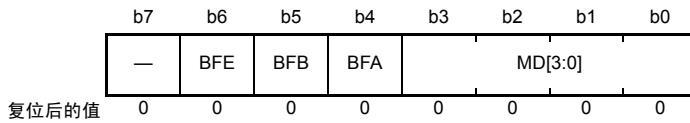
表 21.10 TPSC[1:0] (MTU5)

通道	bit1	bit0	说明
	TPSC1	TPSC0	
MTU5	0	0	内部时钟: 通过 PCLK/1 进行计数
	0	1	内部时钟: 通过 PCLK/4 进行计数
	1	0	内部时钟: 通过 PCLK/16 进行计数
	1	1	内部时钟: 通过 PCLK/64 进行计数

注. 在 MTU5 中, b7-b2 为保留位, 读写值都为“0”。

21.2.2 定时器的模式寄存器 (TMDR)

地址 MTU0.TMDR 0008 8701h、MTU1.TMDR 0008 8781h、MTU2.TMDR 0008 8801h、MTU3.TMDR 0008 8602h、MTU4.TMDR 0008 8603h



位	符号	位名	功能	R/W
b3-b0	MD[3:0]	模式选择位	设定定时器的运行模式，请参照表 21.11。	R/W
b4	BFA	缓冲运行 A 位	0: TGRA 寄存器和 TGRC 寄存器进行正常运行 1: TGRA 寄存器和 TGRC 寄存器进行缓冲运行	R/W
b5	BFB	缓冲运行 B 位	0: TGRB 寄存器和 TGRD 寄存器进行正常运行 1: TGRB 寄存器和 TGRD 寄存器进行缓冲运行	R/W
b6	BFE	缓冲运行 E 位	0: MTU0.TGRE 寄存器和 MTU0.TGRF 寄存器进行正常运行 1: MTU0.TGRE 寄存器和 MTU0.TGRF 寄存器进行缓冲运行	R/W
b7	—	保留位	读写值都为“0”。	R/W

TMDR 寄存器是 8 位可读写寄存器，设定各通道的运行模式。必须在 TCNT 停止计数的状态下设定 TMDR 寄存器。

表 21.11 通过 MD[3:0] 位设定的运行模式

bit3	bit2	bit1	bit0	说明
MD3	MD2	MD1	MD0	
0	0	0	0	正常运行
0	0	0	1	不能设定。
0	0	1	0	PWM 模式 1
0	0	1	1	PWM 模式 2 (注 1)
0	1	0	0	相位计数模式 1 (注 2)
0	1	0	1	相位计数模式 2 (注 2)
0	1	1	0	相位计数模式 3 (注 2)
0	1	1	1	相位计数模式 4 (注 2)
1	0	0	0	复位同步 PWM 模式 (注 3)
1	0	0	1	不能设定。
1	0	1	x	不能设定。
1	1	0	0	不能设定。
1	1	0	1	互补 PWM 模式 1 (在波峰进行传送) (注 3)
1	1	1	0	互补 PWM 模式 2 (在波谷进行传送) (注 3)
1	1	1	1	互补 PWM 模式 3 (在波峰或者波谷进行传送) (注 3)

x: Don't care

注 1. 在 MTU3 和 MTU4 中，不能设定 PWM 模式 2。

注 2. 在 MTU0、MTU3 和 MTU4 中，不能设定相位计数模式。

注 3. 只能在 MTU3 中设定复位同步 PWM 模式或者互补 PWM 模式。

如果将 MTU3 设定为复位同步 PWM 模式或者互补 PWM 模式，MTU4 的设定就无效并且自动服从 MTU3 的设定。必须给 MTU4 设定初始值。

在 MTU0、MTU1 和 MTU2 中，不能设定复位同步 PWM 模式或者互补 PWM 模式。

BFA 位 (缓冲运行 A 位)

此位设定是使 TGRA 进行正常运行还是组合 TGRA 和 TGRC 进行缓冲运行。如果将 TGRC 用作缓冲寄存器，除了互补 PWM 模式以外，不产生 TGRC 的输入捕捉 / 输出比较，但是在互补 PWM 模式中产生 TGRC 的比较匹配。如果在互补 PWM 模式的 Tb 区间发生 MTU4 的比较匹配，就必须将定时器的中断允许寄存器 (MTU4.TIER) 的 TGIEC 位置“0”。

复位同步 PWM 模式和互补 PWM 模式的 MFT3 和 MFT4 的缓冲运行服从 MFT3 的设定。必须给 MFT4.TMDR 的 BFA 位写“0”。

在没有 TGRC 的 MTU1 和 MTU2 中，此位为保留位，读写值都为“0”。有关互补 PWM 模式的 Tb 区间，请参照图 21.40。

BFB 位 (缓冲运行 B 位)

此位设定是使 TGRB 进行正常运行还是组合 TGRB 和 TGRD 进行缓冲运行。如果将 TGRD 用作缓冲寄存器，除了互补 PWM 模式以外，不产生 TGRD 的输入捕捉 / 输出比较，但是在互补 PWM 模式中产生 TGRD 的比较匹配。如果在互补 PWM 模式的 Tb 区间发生比较匹配，就必须将定时器的中断允许寄存器 3、4 (MTU3.TIER 和 MTU4.TIER) 的 TGIED 位置“0”。

复位同步 PWM 模式和互补 PWM 模式的 MTU3 和 MTU4 的缓冲运行服从 MTU3 的设定。必须将 MTU4 的 TMDR.BFB 位置“0”。

在没有 TGRD 的 MTU1 和 MTU2 中，此位为保留位，读写值都为“0”。有关互补 PWM 模式的 Tb 区间，请参照图 21.40。

BFE 位 (缓冲运行 E 位)

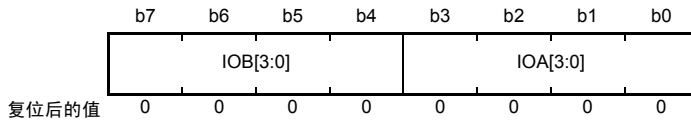
此位选择是否使 MTU0.TGRE 和 MTU0.TGRF 进行正常运行或者缓冲运行。即使将 TGRF 用作缓冲寄存器，也产生 TGRF 的比较匹配。

在 MTU1 ~ MTU4 中，此位为保留位，读写值都为“0”。

21.2.3 定时器的 I/O 控制寄存器 (TIOR)

- MTU0.TIORH、MTU1.TIOR、MTU2.TIOR、MTU3.TIORH、MTU4.TIORH

地址 MTU0.TIORH 0008 8702h、MTU1.TIOR 0008 8782h、MTU2.TIOR 0008 8802h、MTU3.TIORH 0008 8604h、MTU4.TIORH 0008 8606h

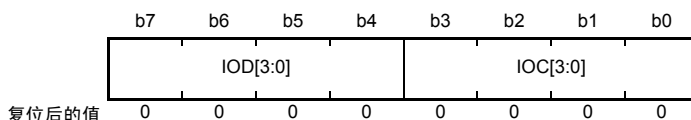


位	符号	位名	功能	R/W
b3-b0	IOA[3:0]	I/O 控制 A 位	请参照下表 (注 1): MTU0.TIORH: 表 21.20 MTU1.TIOR: 表 21.22 MTU2.TIOR: 表 21.23 MTU3.TIORH: 表 21.24 MTU4.TIORH: 表 21.26	R/W
b7-b4	IOB[3:0]	I/O 控制 B 位	请参照下表 (注 1): MTU0.TIORH: 表 21.12 MTU1.TIOR: 表 21.14 MTU2.TIOR: 表 21.15 MTU3.TIORH: 表 21.16 MTU4.TIORH: 表 21.18	R/W

注 1. 在通过比较匹配输出 Low 电平或者 /High 电平或者进行交替输出的过程中, 如果将 IOn[3:0] 位 (n=A、B) 的值变为输出禁止 (“0000b” 或者 “0100b”), 就变为 Hi-Z。

- MTU0.TIORL、MTU3.TIORL、MTU4.TIORL

地址 MTU0.TIORL 0008 8703h、MTU3.TIORL 0008 8605h、MTU4.TIORL 0008 8607h

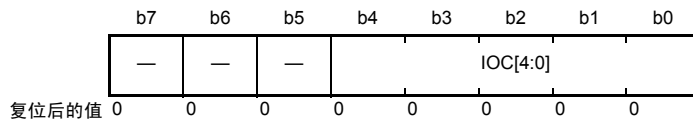


位	符号	位名	功能	R/W
b3-b0	IOC[3:0]	I/O 控制 C 位	请参照下表 (注 1): MTU0.TIORL: 表 21.21 MTU3.TIORL: 表 21.25 MTU4.TIORL: 表 21.27	R/W
b7-b4	IOD[3:0]	I/O 控制 D 位	请参照下表 (注 1): MTU0.TIORL: 表 21.13 MTU3.TIORL: 表 21.17 MTU4.TIORL: 表 21.19	R/W

注 1. 在通过比较匹配输出 Low 电平或者 High 电平或者进行交替输出的过程中, 如果将 IOn[3:0] 位 (n=C、D) 的值变为输出禁止 (“0000b” 或者 “0100b”), 就变为 Hi-Z。

- MTU5.TIORU、MTU5.TIORV、MTU5.TIORW

地址 MTU5.TIORU 0008 8886h、MTU5.TIORV 0008 8896h、MTU5.TIORW 0008 88A6h



位	符号	位名	功能	R/W
b4-b0	IOC[4:0]	I/O 控制 C 位	请参照下表： MTU5.TIORU、MTU5.TIORV、MTU5.TIORW： 表 21.28	R/W
b7-b5	—	保留位	读写值都为“0”。	R/W

在 MTU 中，MTU0、MTU3 和 MTU4 各有 2 个，MTU1 和 MTU2 各有 1 个，MTU5 有 3 个 MTU5.TIORU/V/W，共计 11 个 TIOR 寄存器。

在 TMDR 寄存器的设定为正常运行、PWM 模式或者相位计数模式时设定 TIOR 寄存器。

在计数器停止计数（将 TSTR.CST 位置“0”）的状态下，TIOR 寄存器指定的初始输出有效。另外，在 PWM 模式 2 的情况下，指定计数器为“0”时的输出。

如果将 TGRC 或者 TGRD 设定为缓冲运行，此设定就无效而用于缓冲寄存器的运行。

表 21.12 TIORH (MTU0)

bit7	bit6	bit5	bit4	说明	
IOB3	IOB2	IOB1	IOB0	MTU0.TGRB 的功能	MTIOC0B 引脚的功能
0	0	0	0	输出比较寄存器	禁止输出。
0	0	0	1		初始输出为 Low 电平，在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平，在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平，在比较匹配时进行交替输出。
0	1	0	0		禁止输出。
0	1	0	1		初始输出为 High 电平，在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平，在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平，在比较匹配时进行交替输出。
1	0	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	0	0	1		在下降沿进行输入捕捉。
1	0	1	x		在双边沿进行输入捕捉。
1	1	x	x		捕捉输入源为 MTU1/ 计数时钟。 在 MTU1.TCNT 进行递增计数或者递减计数时进行输入捕捉。

x: Don't care

表 21.13 TIORL (MTU0)

bit7	bit6	bit5	bit4	说明	
IOD3	IOD2	IOD1	IOD0	MTU0.TGRD 的功能	MTIOC0D 引脚的功能
0	0	0	0	输出比较寄存器 (注 1)	禁止输出。
0	0	0	1		初始输出为 Low 电平, 在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平, 在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平, 在比较匹配时进行交替输出。
0	1	0	0		禁止输出。
0	1	0	1		初始输出为 High 电平, 在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平, 在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平, 在比较匹配时进行交替输出。
1	0	0	0	输入捕捉寄存器 (注 1)	在上升沿进行输入捕捉。
1	0	0	1		在下降沿进行输入捕捉。
1	0	1	x		在双边沿进行输入捕捉。
1	1	x	x		捕捉输入源为 MTU1/ 计数时钟。 在 MTU1.TCNT 进行递增计数或者递减计数时进行输入捕捉。

x: Don't care

注 1. 如果将 MTU0.TMDR.BFB 位置“1”使 MTU0.TGRD 用作缓冲寄存器, 此设定就无效并且不发生输入捕捉/输出比较。

表 21.14 TIOR (MTU1)

bit7	bit6	bit5	bit4	说明	
IOB3	IOB2	IOB1	IOB0	MTU1.TGRB 的功能	MTIOC1B 引脚的功能
0	0	0	0	MTU1.TGRB 为输出比较寄存器。	禁止输出。
0	0	0	1		初始输出为 Low 电平, 在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平, 在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平, 在比较匹配时进行交替输出。
0	1	0	0		禁止输出。
0	1	0	1		初始输出为 High 电平, 在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平, 在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平, 在比较匹配时进行交替输出。
1	0	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	0	0	1		在下降沿进行输入捕捉。
1	0	1	x		在双边沿进行输入捕捉。
1	1	x	x		在发生 MTU0.TGRC 的比较匹配 / 输入捕捉时进行输入捕捉。

x: Don't care

表 21.15 TIOR (MTU2)

bit7	bit6	bit5	bit4	说明	
IOB3	IOB2	IOB1	IOB0	MTU2.TGRB 的功能	MTIOC2B 引脚的功能
0	0	0	0	MTU2.TGRB 为输出比较寄存器。	禁止输出。
0	0	0	1		初始输出为 Low 电平, 在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平, 在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平, 在比较匹配时进行交替输出。
0	1	0	0		禁止输出。
0	1	0	1		初始输出为 High 电平, 在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平, 在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平, 在比较匹配时进行交替输出。
1	x	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

x: Don't care

表 21.16 TIORH (MTU3)

bit7	bit6	bit5	bit4	说明	
IOB3	IOB2	IOB1	IOB0	MTU3.TGRB 的功能	MTIOC3B 引脚的功能
0	0	0	0	输出比较寄存器	禁止输出。
0	0	0	1		初始输出为 Low 电平, 在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平, 在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平, 在比较匹配时进行交替输出。
0	1	0	0		禁止输出。
0	1	0	1		初始输出为 High 电平, 在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平, 在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平, 在比较匹配时进行交替输出。
1	x	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

x: Don't care

表 21.17 TIORL (MTU3)

bit7	bit6	bit5	bit4	说明	
IOD3	IOD2	IOD1	IOD0	MTU3.TGRD 的功能	MTIOC3D 引脚的功能
0	0	0	0	输出比较寄存器 (注 1)	禁止输出。
0	0	0	1		初始输出为 Low 电平, 在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平, 在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平, 在比较匹配时进行交替输出。
0	1	0	0		禁止输出。
0	1	0	1		初始输出为 High 电平, 在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平, 在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平, 在比较匹配时进行交替输出。
1	x	0	0	输入捕捉寄存器 (注 1)	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

x: Don't care

注 1. 如果将 MTU3.TMDR.BFB 位置“1”使 MTU3.TGRD 用作缓冲寄存器, 此设定就无效并且不发生输入捕捉/输出比较。

表 21.18 TIORH (MTU4)

bit7	bit6	bit5	bit4	说明	
IOB3	IOB2	IOB1	IOB0	MTU4.TGRB 的功能	MTIOC4B 引脚的功能
0	0	0	0	MTU4.TGRB 为输出比较寄存器。	禁止输出。
0	0	0	1		初始输出为 Low 电平, 在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平, 在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平, 在比较匹配时进行交替输出。
0	1	0	0		禁止输出。
0	1	0	1		初始输出为 High 电平, 在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平, 在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平, 在比较匹配时进行交替输出。
1	x	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

x: Don't care

表 21.19 TIORL (MTU4)

bit7	bit6	bit5	bit4	说明	
IOD3	IOD2	IOD1	IOD0	MTU4.TGRD 的功能	MTIOC4D 引脚的功能
0	0	0	0	输出比较寄存器 (注 1)	禁止输出。
0	0	0	1		初始输出为 Low 电平, 在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平, 在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平, 在比较匹配时进行交替输出。
0	1	0	0		禁止输出。
0	1	0	1		初始输出为 High 电平, 在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平, 在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平, 在比较匹配时进行交替输出。
1	x	0	0	输入捕捉寄存器 (注 1)	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

x: Don't care

注 1. 如果将 MTU4.TMDR.BFB 位置“1”使 MTU4.TGRD 用作缓冲寄存器, 此设定就无效并且不发生输入捕捉/输出比较。

表 21.20 TIORH (MTU0)

bit3	bit2	bit1	bit0	说明	
IOA3	IOA2	IOA1	IOA0	MTU0.TGRA 的功能	MTIOC0A 引脚的功能
0	0	0	0	输出比较寄存器	禁止输出。
0	0	0	1		初始输出为 Low 电平, 在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平, 在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平, 在比较匹配时进行交替输出。
0	1	0	0		禁止输出。
0	1	0	1		初始输出为 High 电平, 在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平, 在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平, 在比较匹配时进行交替输出。
1	0	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	0	0	1		在下降沿进行输入捕捉。
1	0	1	x		在双边沿进行输入捕捉。
1	1	x	x		捕捉输入源为 MTU1/ 计数时钟。 在 MTU1.TCNT 进行递增计数或者递减计数时进行输入捕捉。

x: Don't care

表 21.21 TIORL (MTU0)

bit3	bit2	bit1	bit0	说明	
IOC3	IOC2	IOC1	IOC0	MTU0.TGRC 的功能	MTIOC0C 引脚的功能
0	0	0	0	输出比较寄存器 (注 1)	禁止输出。
0	0	0	1		初始输出为 Low 电平, 在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平, 在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平, 在比较匹配时进行交替输出。
0	1	0	0		禁止输出。
0	1	0	1		初始输出为 High 电平, 在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平, 在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平, 在比较匹配时进行交替输出。
1	0	0	0	输入捕捉寄存器 (注 1)	在上升沿进行输入捕捉。
1	0	0	1		在下降沿进行输入捕捉。
1	0	1	x		在双边沿进行输入捕捉。
1	1	x	x		捕捉输入源为 MTU1/ 计数时钟。 在 MTU1.TCNT 进行递增计数或者递减计数时进行输入捕捉。

x: Don't care

注 1. 如果将 MTU0.TMDR.BFA 位置“1”使 MTU0.TGRC 用作缓冲寄存器, 此设定就无效并且不发生输入捕捉/输出比较。

表 21.22 TIOR (MTU1)

bit3	bit2	bit1	bit0	说明	
IOA3	IOA2	IOA1	IOA0	MTU1.TGRA 的功能	MTIOC1A 引脚的功能
0	0	0	0	输出比较寄存器	禁止输出。
0	0	0	1		初始输出为 Low 电平, 在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平, 在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平, 在比较匹配时进行交替输出。
0	1	0	0		禁止输出。
0	1	0	1		初始输出为 High 电平, 在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平, 在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平, 在比较匹配时进行交替输出。
1	0	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	0	0	1		在下降沿进行输入捕捉。
1	0	1	x		在双边沿进行输入捕捉。
1	1	x	x		在发生 MTU0.TGRA 的比较匹配 / 输入捕捉时进行输入捕捉。

x: Don't care

表 21.23 TIOR (MTU2)

bit3	bit2	bit1	bit0	说明	
IOA3	IOA2	IOA1	IOA0	MTU2.TGRA 的功能	MTIOC2A 引脚的功能
0	0	0	0	输出比较寄存器	禁止输出。
0	0	0	1		初始输出为 Low 电平, 在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平, 在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平, 在比较匹配时进行交替输出。
0	1	0	0		禁止输出。
0	1	0	1		初始输出为 High 电平, 在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平, 在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平, 在比较匹配时进行交替输出。
1	x	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

x: Don't care

表 21.24 TIORH (MTU3)

bit3	bit2	bit1	bit0	说明	
IOA3	IOA2	IOA1	IOA0	MTU3.TGRA 的功能	MTIOC3A 引脚的功能
0	0	0	0	输出比较寄存器	禁止输出。
0	0	0	1		初始输出为 Low 电平, 在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平, 在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平, 在比较匹配时进行交替输出。
0	1	0	0		禁止输出。
0	1	0	1		初始输出为 High 电平, 在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平, 在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平, 在比较匹配时进行交替输出。
1	x	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

x: Don't care

表 21.25 TIORL (MTU3)

bit3	bit2	bit1	bit0	说明	
IOC3	IOC2	IOC1	IOC0	MTU3.TGRC 的引脚	MTIOC3C 引脚的功能
0	0	0	0	输出比较寄存器 (注 1)	禁止输出。
0	0	0	1		初始输出为 Low 电平, 在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平, 在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平, 在比较匹配时进行交替输出。
0	1	0	0		禁止输出。
0	1	0	1		初始输出为 High 电平, 在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平, 在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平, 在比较匹配时进行交替输出。
1	x	0	0	输入捕捉寄存器 (注 1)	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

x: Don't care

注 1. 如果将 MTU3.TMDR.BFA 位置“1”使 MTU3.TGRC 用作缓冲寄存器, 此设定就无效并且不发生输入捕捉/输出比较。

表 21.26 TIORH (MTU4)

bit3	bit2	bit1	bit0	说明	
IOA3	IOA2	IOA1	IOA0	MTU4.TGRA 的功能	MTIOC4A 引脚的功能
0	0	0	0	输出比较寄存器	禁止输出。
0	0	0	1		初始输出为 Low 电平, 在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平, 在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平, 在比较匹配时进行交替输出。
0	1	0	0		禁止输出。
0	1	0	1		初始输出为 High 电平, 在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平, 在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平, 在比较匹配时进行交替输出。
1	x	0	0	输入捕捉寄存器	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

x: Don't care

表 21.27 TIORL (MTU4)

bit3	bit2	bit1	bit0	说明	
IOC3	IOC2	IOC1	IOC0	MTU4.TGRC 的功能	MTIOC4C 引脚的功能
0	0	0	0	输出比较寄存器 (注 1)	禁止输出。
0	0	0	1		初始输出为 Low 电平, 在比较匹配时输出 Low 电平。
0	0	1	0		初始输出为 Low 电平, 在比较匹配时输出 High 电平。
0	0	1	1		初始输出为 Low 电平, 在比较匹配时进行交替输出。
0	1	0	0		禁止输出。
0	1	0	1		初始输出为 High 电平, 在比较匹配时输出 Low 电平。
0	1	1	0		初始输出为 High 电平, 在比较匹配时输出 High 电平。
0	1	1	1		初始输出为 High 电平, 在比较匹配时进行交替输出。
1	x	0	0	输入捕捉寄存器 (注 1)	在上升沿进行输入捕捉。
1	x	0	1		在下降沿进行输入捕捉。
1	x	1	x		在双边沿进行输入捕捉。

x: Don't care

注 1. 如果将 MTU4.TMDR.BFA 位置“1”使 MTU4.TGRC 用作缓冲寄存器, 此设定就无效并且不发生输入捕捉/输出比较。

表 21.28 TIORU、TIO RV、TIO RW (MTU5)

bit4	bit3	bit2	bit1	bit0	说明	
IOC4	IOC3	IOC2	IOC1	IOC0	MTU5.TGRU、MTU5.TGRV、 MTU5.TGRW 的功能	MTIC5U、MTIC5V、MTIC5W 引脚的功能
0	0	0	0	0	比较匹配寄存器	比较匹配
0	0	0	0	1		不能设定。
0	0	0	1	x		不能设定。
0	0	1	x	x		不能设定。
0	1	x	x	x		不能设定。
1	0	0	0	0	输入捕捉寄存器	不能设定。
1	0	0	0	1		在上升沿进行输入捕捉。
1	0	0	1	0		在下降沿进行输入捕捉。
1	0	0	1	1		在双边沿进行输入捕捉。
1	0	1	x	x		不能设定。
1	1	0	0	0		不能设定。
1	1	0	0	1		用于测量外部输入信号的 Low 电平脉宽。 在互补 PWM 模式的波谷进行捕捉。
1	1	0	1	0		用于测量外部输入信号的 Low 电平脉宽。 在互补 PWM 模式的波峰进行捕捉。
1	1	0	1	1		用于测量外部输入的信号 Low 电平脉宽。 在互补 PWM 模式的波峰和波谷进行捕捉。
1	1	1	0	0		不能设定。
1	1	1	0	1		用于测量外部输入信号的 High 电平脉宽。 在互补 PWM 模式的波谷进行捕捉。
1	1	1	1	0		用于测量外部输入信号的 High 电平脉宽。 在互补 PWM 模式的波峰进行捕捉。
1	1	1	1	1		用于测量外部输入信号的 High 电平脉宽。 在互补 PWM 模式的波峰和波谷进行捕捉。

x: Don't care

21.2.4 定时器的比较匹配清除寄存器 (TCNTCMPCLR)

地址 MTU5.TCNTCMPCLR 0008 88B6h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	CMPCLR 5U	CMPCLR 5V	CMPCLR 5W
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	CMPCLR5W	TCNT 比较清除 5W 位	0: 禁止通过 MTU5.TCNTW 和 MTU5.TGRW 的比较匹配 / 输入捕捉将 MTU5.TCNTW 清 “0000h” 1: 允许通过 MTU5.TCNTW 和 MTU5.TGRW 的比较匹配 / 输入捕捉将 MTU5.TCNTW 清 “0000h”	R/W
b1	CMPCLR5V	TCNT 比较清除 5V 位	0: 禁止通过 MTU5.TCNTV 和 MTU5.TGRV 的比较匹配 / 输入捕捉将 MTU5.TCNTV 清 “0000h” 1: 允许通过 MTU5.TCNTV 和 MTU5.TGRV 的比较匹配 / 输入捕捉将 MTU5.TCNTV 清 “0000h”	R/W
b2	CMPCLR5U	TCNT 比较清除 5U 位	0: 禁止通过 MTU5.TCNTU 和 MTU5.TGRU 的比较匹配 / 输入捕捉将 MTU5.TCNTU 清 “0000h” 1: 允许通过 MTU5.TCNTU 和 MTU5.TGRU 的比较匹配 / 输入捕捉将 MTU5.TCNTU 清 “0000h”	R/W
b7-b3	—	保留位	读写值都为 “0”。	R/W

TCNTCMPCLR 寄存器是 8 位可读写寄存器，能设定 MTU5.TCNTU、MTU5.TCNTV 和 MTU5.TCNTW 的清除请求。

21.2.5 定时器的中断允许寄存器 (TIER)

• TIER (MTU0~MTU4)

地址 MTU0.TIER 0008 8704h、MTU1.TIER 0008 8784h、MTU2.TIER 0008 8804h、MTU3.TIER 0008 8608h、MTU4.TIER 0008 8609h

	b7	b6	b5	b4	b3	b2	b1	b0
	TTGE	TTGE2	TCIEU	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	TGIEA	TGR 中断允许 A 位	0: 禁止中断请求 (TGIA) 1: 允许中断请求 (TGIA)	R/W
b1	TGIEB	TGR 中断允许 B 位	0: 禁止中断请求 (TGIB) 1: 允许中断请求 (TGIB)	R/W
b2	TGIEC	TGR 中断允许 C 位	0: 禁止中断请求 (TGIC) 1: 允许中断请求 (TGIC)	R/W
b3	TGIED	TGR 中断允许 D 位	0: 禁止中断请求 (TGID) 1: 允许中断请求 (TGID)	R/W
b4	TCIEV	上溢中断允许位	0: 禁止中断请求 (TCIV) 1: 允许中断请求 (TCIV)	R/W
b5	TCIEU	下溢中断允许位	0: 禁止中断请求 (TCIU) 1: 允许中断请求 (TCIU)	R/W
b6	TTGE2	A/D 转换开始请求允许 2 位	0: 禁止因 MTU4.TCNT 的下溢 (波谷) 而产生的 A/D 转换请求 1: 允许因 MTU4.TCNT 的下溢 (波谷) 而产生的 A/D 转换请求	R/W
b7	TTGE	A/D 转换开始请求允许位	0: 禁止产生 A/D 转换开始请求 1: 允许产生 A/D 转换开始请求	R/W

在 MTU 中，MTU0 有 2 个，MTU1 ~ MTU5 各有 1 个，共计 7 个 TIER 寄存器。
TIER 寄存器是 8 位可读写寄存器，控制允许或者禁止各通道的中断请求。

TGIEA 位、TGIEB 位 (TGR 中断允许 A 位、B 位)

这些位允许或者禁止中断请求 (TGIn) (n=A、B)。

TGIEC 位、TGIED 位 (TGR 中断允许 C 位、D 位)

在 MTU0、MTU3 和 MTU4 中，这些位允许或者禁止中断请求 (TGIn) (n=C、D)。
在 MTU1 和 MTU2 中，这些位为保留位，读写值都为“0”。

TCIEV 位 (上溢中断允许位)

此位允许或者禁止中断请求 (TCIV)。

TCIEU 位 (下溢中断允许位)

在 MTU1 和 MTU2 中，此位允许或者禁止中断请求 (TCIU)。
在 MTU0、MTU3 和 MTU4 中，此位为保留位，读写值都为“0”。

TTGE2 位 (A/D 转换开始请求允许 2 位)

在互补 PWM 模式中，此位允许或者禁止因 MTU4.TCNT 的下溢（波谷）而产生的 A/D 转换请求。
在 MTU0 ~ MTU3 中，此位为保留位，读写值都为“0”。

TTGE 位 (A/D 转换开始请求允许位)

此位允许或者禁止因 TGRA 寄存器的输入捕捉 / 比较匹配而产生的 A/D 转换器开始请求。

• **TIER2 (MTU0)**

地址 MTU0.TIER2 0008 8724h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	TGIEF	TGIEE
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	TGIEE	TGR 中断允许 E 位	0: 禁止中断请求 (TGIE) 1: 允许中断请求 (TGIE)	R/W
b1	TGIEF	TGR 中断允许 F 位	0: 禁止中断请求 (TGIF) 1: 允许中断请求 (TGIF)	R/W
b7-b2	—	保留位	读写值都为“0”。	R/W

TGIEE 位、TGIEF 位 (TGR 中断允许 E 位、F 位)

此位允许或者禁止因 MTU0.TCNT 和 MTU0.TGRm 的比较匹配而产生的中断请求 (m=E、F)。

• **TIER (MTU5)**

地址 MTU5.TIER 0000 88B2h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	TGIE5U	TGIE5V	TGIE5W
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	TGIE5W	TGR 中断允许 5W 位	0: 禁止 TGI5W 中断请求 1: 允许 TGI5W 中断请求	R/W
b1	TGIE5V	TGR 中断允许 5V 位	0: 禁止 TGI5V 中断请求 1: 允许 TGI5V 中断请求	R/W
b2	TGIE5U	TGR 中断允许 5U 位	0: 禁止 TGI5U 中断请求 1: 允许 TGI5U 中断请求	R/W
b7-b3	—	保留位	读写值都为“0”。	R/W

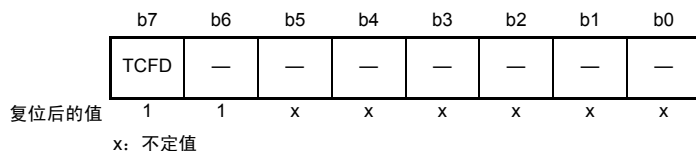
TGIE5W 位、TGIE5V 位、TGIE5U 位 (TGR 中断允许 5m 位)

这些位允许或者禁止中断请求 (TGIm5) (m=W、V、U)。

21.2.6 定时器的状态寄存器 (TSR)

- TSR (MTU0 ~ MTU4)

地址 MTU0.TSR 0008 8705h、MTU1.TSR 0008 8785h、MTU2.TSR 0008 8805h、MTU3.TSR 0008 862Ch、MTU4.TSR 0008 862Dh



位	符号	位名	功能	R/W
b5-b0	—	保留位	读取值为不定值，只能写“1”。	R/W
b6	—	保留位	读写值都为“1”。	R/W
b7	TCFD	计数方向标志	0: TCNT 进行递减计数 1: TCNT 进行递增计数	R

在 MTU 中，MTU0 ~ MTU4 各有 1 个，共计 5 个 TSR 寄存器。

TSR 寄存器是 8 位可读写寄存器，表示各通道的状态。

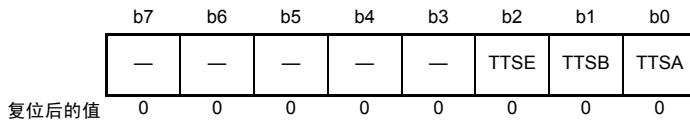
TCFD 标志 (计数方向标志)

这是表示 MTU1 ~ MTU4 的 TCNT 计数方向的状态标志。

在 MTU0 中，此位为保留位，读写值都为“1”。

21.2.7 定时器的缓冲运行传送模式寄存器 (TBTM)

地址 MTU0.TBTM 0008 8726h、MTU3.TBTM 0008 8638h、MTU4.TBTM 0008 8639h



位	符号	位名	功能	R/W
b0	TTSA	时序选择 A 位	0: 在发生各通道的比较匹配 A 时从 TGRC 传送到 TGRA 1: 在清除各通道的 TCNT 时从 TGRC 传送到 TGRA	R/W
b1	TTSB	时序选择 B 位	0: 在发生各通道的比较匹配 B 时从 TGRD 传送到 TGRB 1: 在清除各通道的 TCNT 时从 TGRD 传送到 TGRB	R/W
b2	TTSE	时序选择 E 位	0: 在发生各通道 MTU0 的比较匹配 E 时从 MTU0.TGRF 传送到 MTU0.TGRE 1: 在清除各通道的 MTU0.TCNT 时从 MTU0.TGRF 传送到 MTU0.TGRE	R/W
b7-b3	—	保留位	读写值都为“0”。	R/W

在 MTU 中，MTU0、MTU3 和 MTU4 各有 1 个，共计 3 个 TBTM 寄存器。

TBTM 寄存器设定在 PWM 模式中缓冲寄存器到定时器的通用寄存器的传送时序。

TTSA 位 (时序选择 A 位)

此位设定在各通道进行缓冲运行时 TGRC 到 TGRA 的传送时序。在用于其他模式 (非 PWM 模式) 的通道中，不能将 TTSA 位置“1”。

TTSB 位 (时序选择 B 位)

此位设定在各通道进行缓冲运行时 TGRD 到 TGRB 的传送时序。在用于其他模式 (非 PWM 模式) 的通道中，不能将 TTSB 位置“1”。

TTSE 位 (时序选择 E 位)

此位设定在进行缓冲运行时 MTU0.TGRF 到 MTU0.TGRE 的传送时序。在 MTU3 和 MTU4 中，此位为保留位，读写值都为“0”。在将 MTU0 用于非 PWM 模式时，不能将 TTSE 位置“1”。

21.2.8 定时器的输入捕捉控制寄存器 (TICCR)

地址 MTU1.TICCR 0008 8790h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	I2BE	I2AE	I1BE	I1AE
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	I1AE	输入捕捉允许位	0: 不将 MTIOC1A 引脚追加到 MTU2.TGRA 的输入捕捉条件 1: 将 MTIOC1A 引脚追加到 MTU2.TGRA 的输入捕捉条件	R/W
b1	I1BE	输入捕捉允许位	0: 不将 MTIOC1B 引脚追加到 MTU2.TGRB 的输入捕捉条件 1: 将 MTIOC1B 引脚追加到 MTU2.TGRB 的输入捕捉条件	R/W
b2	I2AE	输入捕捉允许位	0: 不将 MTIOC2A 引脚追加到 MTU1.TGRA 的输入捕捉条件 1: 将 MTIOC2A 引脚追加到 MTU1.TGRA 的输入捕捉条件	R/W
b3	I2BE	输入捕捉允许位	0: 不将 MTIOC2B 引脚追加到 MTU1.TGRB 的输入捕捉条件 1: 将 MTIOC2B 引脚追加到 MTU1.TGRB 的输入捕捉条件	R/W
b7-b4	—	保留位	读写值都为“0”。	R/W

在 MTU 中，MTU1 有 1 个 TICCR 寄存器。

TICCR 寄存器控制 MTU1.TCNT 和 MTU2.TCNT 级联时的输入捕捉条件。

21.2.9 定时器的 A/D 转换开始请求控制寄存器 (TADCR)

地址 MTU4.TADCR 0008 8640h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
BF[1:0]	—	—	—	—	—	—	—	UT4AE	DT4AE	UT4BE	DT4BE	ITA3AE	ITA4VE	ITB3AE	ITB4VE
0	0	0	0	0	0	0	0	0	0 (注1)	0 (注1)	0 (注1)	0 (注1)	0 (注1)	0 (注1)	0 (注1)

复位后的值

注 1. 除了互补 PWM 模式以外, 不能置“1”。

位	符号	位名	功能	R/W
b0	ITB4VE	TCIV4 中断减少联动允许位	0: 不联动 TCI4V 中断减少功能 1: 联动 TCI4V 中断减少功能	R/W
b1	ITB3AE	TGIA3 中断减少联动允许位	0: 不联动 TGI3A 中断减少功能 1: 联动 TGI3A 中断减少功能	R/W
b2	ITA4VE	TCIV4 中断减少联动允许位	0: 不联动 TCI4V 中断减少功能 1: 联动 TCI4V 中断减少功能	R/W
b3	ITA3AE	TGIA3 中断减少联动允许位	0: 不联动 TGI3A 中断减少功能 1: 联动 TGI3A 中断减少功能	R/W
b4	DT4BE	递减计数 TRG4BN 允许位	0: 禁止在 MTU4.TCNT 进行递减计数时发生 A/D 转换的开始请求 (TRG4BN) 1: 允许在 MTU4.TCNT 进行递减计数时发生 A/D 转换的开始请求 (TRG4BN)	R/W
b5	UT4BE	递增计数 TRG4BN 允许位	0: 禁止在 MTU4.TCNT 进行递增计数时发生 A/D 转换的开始请求 (TRG4BN) 1: 允许在 MTU4.TCNT 进行递增计数时发生 A/D 转换的开始请求 (TRG4BN)	R/W
b6	DT4AE	递减计数 TRG4AN 允许位	0: 禁止在 MTU4.TCNT 进行递减计数时发生 A/D 转换的开始请求 (TRG4AN) 1: 允许在 MTU4.TCNT 进行递减计数时发生 A/D 转换的开始请求 (TRG4AN)	R/W
b7	UT4AE	递增计数 TRG4AN 允许位	0: 禁止在 MTU4.TCNT 进行递增计数时发生 A/D 转换的开始请求 (TRG4AN) 1: 允许在 MTU4.TCNT 进行递增计数时发生 A/D 转换的开始请求 (TRG4AN)	R/W
b13-b8	—	保留位	读写值都为“0”。	R/W
b15-b14	BF[1:0]	MTU4.TADCOBRA/B 传送时序选择位	详细内容请参照表 21.29。	R/W

注. 禁止以 8 位为单位存取 TADCR, 必须以 16 位为单位进行存取。

当禁止中断减少功能时 (将定时器的中断减少设定寄存器 (TITCR) 的 T3AEN 位和 T4VEN 位置“0”, 或者将 TITCR 的减少次数设定位 (T3ACOR 和 T4VCOR) 置“0”时), 必须设定为不联动中断减少功能 (将定时器的 A/D 转换开始请求控制寄存器 (TADCR) 的 ITA3AE、ITA4VE、ITB3AE、ITB4VE 位置“0”)。

在禁止中断减少功能时, 如果设定为联动中断减少功能, 就不进行 A/D 转换的开始请求。

除了互补 PWM 模式以外, 不能将 b6~b0 的初始值设定为“1”。

TADCR 寄存器是 16 位可读写寄存器, 允许或者禁止 A/D 转换的开始请求以及设定 A/D 转换开始请求是否联动中断减少功能。

表 21.29 通过 BF0[1:0] 位进行传送时序的设定

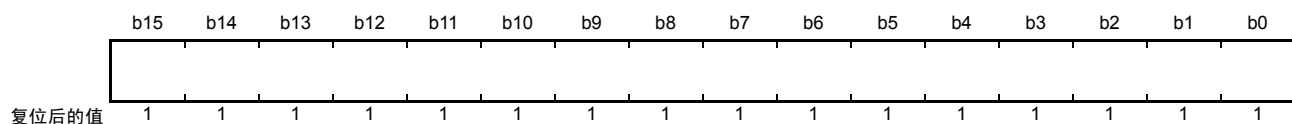
bit15	bit14	说明
BF1	BF0	
0	0	不从周期设定缓冲寄存器传送到周期设定寄存器。
0	1	在 MTU4.TCNT 的波峰从周期设定缓冲寄存器传送到周期设定寄存器 (注 1)。
1	0	在 MTU4.TCNT 的波谷从周期设定缓冲寄存器传送到周期设定寄存器 (注 2)。
1	1	在 MTU4.TCNT 的波峰和波谷从周期设定缓冲寄存器传送到周期设定寄存器 (注 2)。

注 1. 在互补 PWM 模式中，在 MTU4.TCNT 的波峰从周期设定缓冲寄存器传送到周期设定寄存器；在复位同步 PWM 模式中，当发生 MTU3.TCNT 和 MTU3.TGRA 的比较匹配时，从周期设定缓冲寄存器传送到周期设定寄存器；在 PWM 模式 1/ 正常运行模式中，当发生 MTU4.TCNT 和 MTU4.TGRA 的比较匹配时，从周期设定缓冲寄存器传送到周期设定寄存器。

注 2. 除了互补 PWM 模式以外，禁止此设定。

21.2.10 定时器的 A/D 转换开始请求周期设定寄存器 A、B (TADCORA/B)

地址 MTU4.TADCORA 0008 8644h、MTU4.TADCORB 0008 8646h



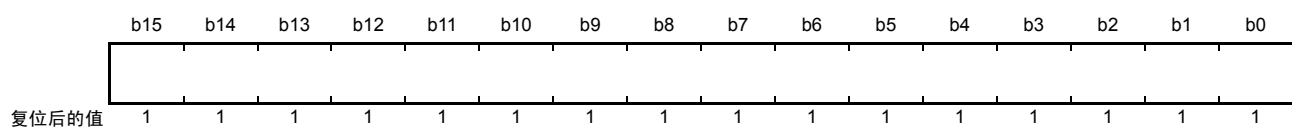
注. 禁止以 8 位为单位存取 MTU4.TADCORA/B，必须以 16 位为单位进行存取。

TADCORA/B 寄存器是 16 位可读写寄存器。当 TADCORA/B 和 MTU4.TCNT 相同时，产生对应的 A/D 转换开始请求。

复位后，TADCORA/B 寄存器的值为“FFFFh”。

21.2.11 定时器的 A/D 转换开始请求周期设定缓冲寄存器 A、B (TADCOBRA/B)

地址 MTU4.TADCOBRA 0008 8648h、MTU4.TADCOBRB 0008 864Ah



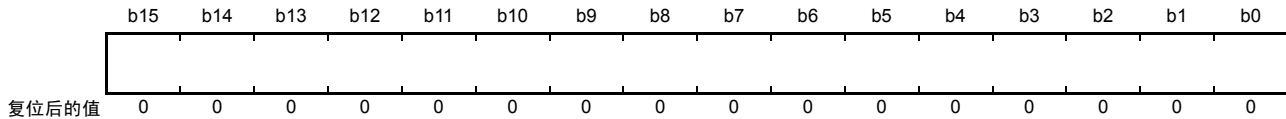
注. 禁止以 8 位为单位存取 MTU4.TADCOBRA/B，必须以 16 位为单位进行存取。

TADCOBRA/B 寄存器是 16 位可读写寄存器，在波峰或者波谷从 TADCORA/B 的缓冲寄存器传送到 TADCORA/B。

复位后，TADCOBRA/B 寄存器的值为“FFFFh”。

21.2.12 定时器的计数器 (TCNT)

地址 MTU0.TCNT 0008 8706h、MTU1.TCNT 0008 8786h、MTU2.TCNT 0008 8806h、MTU3.TCNT 0008 8610h、MTU4.TCNT 0008 8612h、MTU5.TCNTU 0008 8880h、MTU5.TCNTV 0008 8890h、MTU5.TCNTW 0008 88A0h



注：禁止以 8 位为单位存取 TCNT，必须以 16 位为单位进行存取。

在 MTU 中，MTU0 ~ MTU4 各有 1 个，MTU5 有 3 个 MTU5.TCNTU/V/W，共计 8 个 TCNT。TCNT 是 16 位可读写计数器。在复位时，TCNT 被初始化为“0000h”。

21.2.13 定时器的通用寄存器 (TGR)

地址 MTU0.TGRA 0008 8708h、MTU0.TGRB 0008 870Ah、MTU0.TGRC 0008 870Ch、MTU0.TGRD 0008 870Eh、MTU0.TGRE 0008 8720h、MTU0.TGRF 0008 8722h、MTU1.TGRA 0008 8788h、MTU1.TGRB 0008 878Ah、MTU2.TGRA 0008 8808h、MTU2.TGRB 0008 880Ah、MTU3.TGRA 0008 8618h、MTU3.TGRB 0008 861Ah、MTU3.TGRC 0008 8624h、MTU3.TGRD 0008 8626h、MTU4.TGRA 0008 861Ch、MTU4.TGRB 0008 861Eh、MTU4.TGRC 0008 8628h、MTU4.TGRD 0008 862Ah、MTU5.TGRU 0008 8882h、MTU5.TGRV 0008 8892h、MTU5.TGRW 0008 88A2h



注：禁止以 8 位为单位存取 TGR，必须以 16 位为单位进行存取。TGR 的初始值为“FFFFh”。

在 MTU 中，MTU0 有 6 个，MTU1 和 MTU2 各有 2 个，MTU3 和 MTU4 各有 4 个，MTU5 有 3 个，共计 21 个通用寄存器。

TGR 是 16 位可读写寄存器。TGRA、TGRB、TGRC 和 TGRD 是输出比较 / 输入捕捉兼用的寄存器。能将 MTU0、MTU3 和 MTU4 的 TGRC 和 TGRD 用作缓冲寄存器。TGR 和缓冲寄存器的组合为 TGRA-TGRC、TGRB-TGRD。

MTU0.TGRE 和 MTU0.TGRF 用作比较寄存器，当 MTU0.TCNT 和 MTU0.TGRE 相同时，能产生 A/D 转换开始请求。能将 TGRF 用作缓冲寄存器。TGR 和缓冲寄存器的组合为 TGRE-TGRF。

MTU5.TGRU、MTU5.TGRV 和 MTU5.TGRW 是比较匹配 / 输入捕捉 / 外部脉宽测量兼用的寄存器。

21.2.14 定时器的启动寄存器 (TSTR)

• TSTR (MTU0~MTU4)

地址 MTU.TSTR 0008 8680h

	b7	b6	b5	b4	b3	b2	b1	b0
	CST4	CST3	—	—	—	CST2	CST1	CST0
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	CST0	计数器开始 0 位	0: MTU0.TCNT 停止计数 1: MTU0.TCNT 计数运行	R/W
b1	CST1	计数器开始 1 位	0: MTU1.TCNT 停止计数 1: MTU1.TCNT 计数运行	R/W
b2	CST2	计数器开始 2 位	0: MTU2.TCNT 停止计数 1: MTU2.TCNT 计数运行	R/W
b5-b3	—	保留位	读写值都为“0”。	R/W
b6	CST3	计数器开始 3 位	0: MTU3.TCNT 停止计数 1: MTU3.TCNT 计数运行	R/W
b7	CST4	计数器开始 4 位	0: MTU4.TCNT 停止计数 1: MTU4.TCNT 计数运行	R/W

TSTR 寄存器选择 MTU0 ~ MTU4 的 TCNT 的运行或者停止。

在给 TMDR 寄存器设定运行模式或者给 TCR 寄存器设定 TCNT 的计数时钟时，必须在停止 TCNT 计数器的运行后再进行设定。

CSTn 位 (计数开始 n 位) (n=0 ~ 4)

此位选择各通道 TCNT 的运行或者停止。

在 MTIOC 引脚为输出状态下运行时，如果给 CSTn 位写“0”，计数器就停止计数，但是保持 MTIOC 引脚的输出比较的输出电平。如果在 CSTn 位为“0”的状态下写 TIOR 寄存器，就将引脚的输出电平更新为所设定的初始输出值。

• TSTR (MTU5)

地址 MTU5.TSTR 0008 88B4h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	CSTU5	CSTV5	CSTW5
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	CSTW5	计数器开始 W5 位	0: MTU5.TCNTW 停止计数 1: MTU5.TCNTW 计数运行	R/W
b1	CSTV5	计数器开始 V5 位	0: MTU5.TCNTV 停止计数 1: MTU5.TCNTV 计数运行	R/W
b2	CSTU5	计数器开始 U5 位	0: MTU5.TCNTU 停止计数 1: MTU5.TCNTU 计数运行	R/W
b7-b3	—	保留位	读写值都为“0”。	R/W

21.2.15 定时器的同步寄存器 (TSYR)

地址 MTU.TSYR 0008 8681h

b7	b6	b5	b4	b3	b2	b1	b0
SYNC4	SYNC3	—	—	—	SYNC2	SYNC1	SYNC0
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	SYNC0	定时器同步 0 位	0: MTU0.TCNT 独立运行 (TCNT 的预置 / 清除与其他通道无关) 1: MTU0.TCNT 进行同步运行 能进行 TCNT 的同步预置 / 同步清除。	R/W
b1	SYNC1	定时器同步 1 位	0: MTU1.TCNT 独立运行 (TCNT 的预置 / 清除与其他通道无关) 1: MTU1.TCNT 进行同步运行 能进行 TCNT 的同步预置 / 同步清除。	R/W
b2	SYNC2	定时器同步 2 位	0: MTU2.TCNT 独立运行 (TCNT 的预置 / 清除与其他通道无关) 1: MTU2.TCNT 进行同步运行 能进行 TCNT 的同步预置 / 同步清除。	R/W
b5-b3	—	保留位	读写值都为“0”。	R/W
b6	SYNC3	定时器同步 3 位	0: MTU3.TCNT 独立运行 (TCNT 的预置 / 清除与其他通道无关) 1: MTU3.TCNT 进行同步运行 能进行 TCNT 的同步预置 / 同步清除。	R/W
b7	SYNC4	定时器同步 4 位	0: MTU4.TCNT 独立运行 (TCNT 的预置 / 清除与其他通道无关) 1: MTU4.TCNT 进行同步运行 能进行 TCNT 的同步预置 / 同步清除。	R/W

TSYR 寄存器选择 MTU0 ~ MTU4 的 TCNT 进行独立运行或者同步运行。
对应位为“1”的通道进行同步运行。

SYNCn 位 (定时器同步 n 位) (n=0 ~ 4)

此位选择是与其他通道同步运行还是独立运行。

如果选择同步运行, 就能对多个 TCNT 进行同步预置并且通过清除其他通道的计数器进行同步清除。

要设定同步运行时, 需要至少将 2 个通道的 SYNCn 位置“1”; 要设定同步清除时, 除了 SYNCn 位以外, 还需要通过 TCR.CCLR[2:0] 位设定 TCNT 的清除源。

21.2.16 定时器的读写允许寄存器 (TRWER)

地址 MTU.TRWER 0008 8684h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	RWE
0	0	0	0	0	0	0	1

复位后的值

位	符号	位名	功能	R/W
b0	RWE	读写允许位	0: 禁止读写寄存器 1: 允许读写寄存器	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

TRWER 寄存器设定允许或者禁止存取 MTU3 和 MTU4 的误写防止对象寄存器 / 计数器。

RWE 位 (读写允许位)

此位设定允许或者禁止读写误写防止寄存器。

[为“0”的条件]

- 在 RWE 位为“1”的状态下读 RWE 位后给 RWE 位写“0”时
- 误写防止对象寄存器 / 对象计数器
MTUm.TCR、MTUm.TMDR、MTUm.TIORH、MTUm.TIORL、MTUm.TIER、MTUm.TGRA、
MTUm.TGRB、TOER、TOCR1、TOCR2、TGCR、TCDR、TDDR 和 MTUm.TCNT，共计 22 个寄存器
(m=3、4)。

21.2.17 定时器的输出主控允许寄存器 (TOER)

地址 MTU.TOER 0008 860Ah

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	OE4D	OE4C	OE3D	OE4B	OE4A	OE3B
复位后的值	1	1	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	OE3B	主控允许 MTIOC3B 位	0: 禁止 MTU 输出 (无效电平) (注 1) 1: 允许 MTU 输出	R/W
b1	OE4A	主控允许 MTIOC4A 位	0: 禁止 MTU 输出 (无效电平) (注 1) 1: 允许 MTU 输出	R/W
b2	OE4B	主控允许 MTIOC4B 位	0: 禁止 MTU 输出 (无效电平) (注 1) 1: 允许 MTU 输出	R/W
b3	OE3D	主控允许 MTIOC3D 位	0: 禁止 MTU 输出 (无效电平) (注 1) 1: 允许 MTU 输出	R/W
b4	OE4C	主控允许 MTIOC4C 位	0: 禁止 MTU 输出 (无效电平) (注 1) 1: 允许 MTU 输出	R/W
b5	OE4D	主控允许 MTIOC4D 位	0: 禁止 MTU 输出 (无效电平) (注 1) 1: 允许 MTU 输出	R/W
b7-b6	—	保留位	读写值都为“1”。	R/W

注 1. 无效电平取决于定时器的输出控制寄存器 1/2 (TOCR1/2) 的设置。

详细内容请参照“21.2.18 定时器的输出控制寄存器 1 (TOCR1)”和“21.2.19 定时器的输出控制寄存器 2 (TOCR2)”。除了互补 PWM 模式和复位同步 PWM 模式以外, 在进行 MTU 输出时必须置“1”。如果置“0”, 就根据定时器的输出控制寄存器 1/2 (TOCR1/2) 的设置输出无效电平。

TOER 寄存器允许或者禁止输出引脚的 MTIOC4D、MTIOC4C、MTIOC3D、MTIOC4B、MTIOC4A、MTIOC3B 的输出设定。

如果不设定 TOER 寄存器的各位, 就无法正确地输出这些引脚。在 MTU3 和 MTU4 中, 必须在设定 MTU3 和 MTU4 的 TIOR 寄存器前给 TOER 寄存器设定值。

21.2.18 定时器的输出控制寄存器 1 (TOCR1)

地址 MTU.TOCR1 0008 860Eh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	PSYE	—	—	TOCL	TOCS	OLSN	OLSP
复位后的值	0	0	0	0	0 (注1)	0	0	0

注 1. 复位后, 只能写 1 次 “1” 并且不能在写 “1” 后写 “0”。

位	符号	位名	功能	R/W
b0	OLSP	输出电平选择 P 位 (注 2)	请参照表 21.30。	R/W
b1	OLSN	输出电平选择 N 位 (注 2)	请参照表 21.31。	R/W
b2	TOCS	TOC 选择位	0: TOCR1 的设定有效 1: TOCR2 的设定有效	R/W
b3	TOCL	TOC 寄存器的写禁止位 (注 1)	0: 允许写 TOCS 位、OLSN 位和 OLSP 位 1: 禁止写 TOCS 位、OLSN 位和 OLSP 位	R/W (注 3)
b5-b4	—	保留位	读写值都为 “0”。	R/W
b6	PSYE	PWM 同步输出允许位	0: 禁止交替输出 1: 允许交替输出	R/W
b7	—	保留位	读写值都为 “0”。	R/W

注 1. 能通过将 TOCR1.TOCL 位置 “1”, 防止 CPU 失控时的误写。

注 2. 通过将 TOCR1.TOCS 位置 “0”, 使此设定变为有效。

注 3. 复位后, 只能写 1 次 “1” 并且不能在写 “1” 后写 “0”。

TOCR1 寄存器是 8 位可读写寄存器, 允许或者禁止与互补 PWM 模式 / 复位同步 PWM 模式的 PWM 周期同步进行交替输出以及对 PWM 输出电平进行反相控制。

OLSP 位 (输出电平选择 P 位)

在复位同步 PWM 模式 / 互补 PWM 模式中, 此位选择正相的输出电平。

OLSN 位 (输出电平选择 N 位)

在复位同步 PWM 模式 / 互补 PWM 模式中, 此位选择反相的输出电平。

TOCS 位 (TOC 选择位)

此位选择互补 PWM 模式 / 复位同步 PWM 模式的输出电平是 TOCR1 的设定有效还是 TOCR2 的设定有效。

TOCL 位 (TOC 寄存器的写禁止位)

此位设定允许或者禁止写 TOCR1 寄存器的 TOCS 位、OLSN 位和 OLSP 位。

PSYE 位 (PWM 同步输出允许位)

此位设定允许或者禁止与 PWM 周期同步进行交替输出。

表 21.30 输出电平的选择功能

bit0	功能			
OLSP	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	High 电平	Low 电平	Low 电平	High 电平
1	Low 电平	High 电平	High 电平	Low 电平

表 21.31 输出电平的选择功能

bit1	功能			
OLSN	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	High 电平	Low 电平	High 电平	Low 电平
1	Low 电平	High 电平 </td <td>Low 电平</td> <td>High 电平</td>	Low 电平	High 电平

注. 在开始计数并且经过死区时间后，反相波形的初始输出值变为有效电平。

OLSN 为“1”并且 OLSP 为“1”时的互补 PWM 模式的输出例子（1 相）如图 21.2 所示。

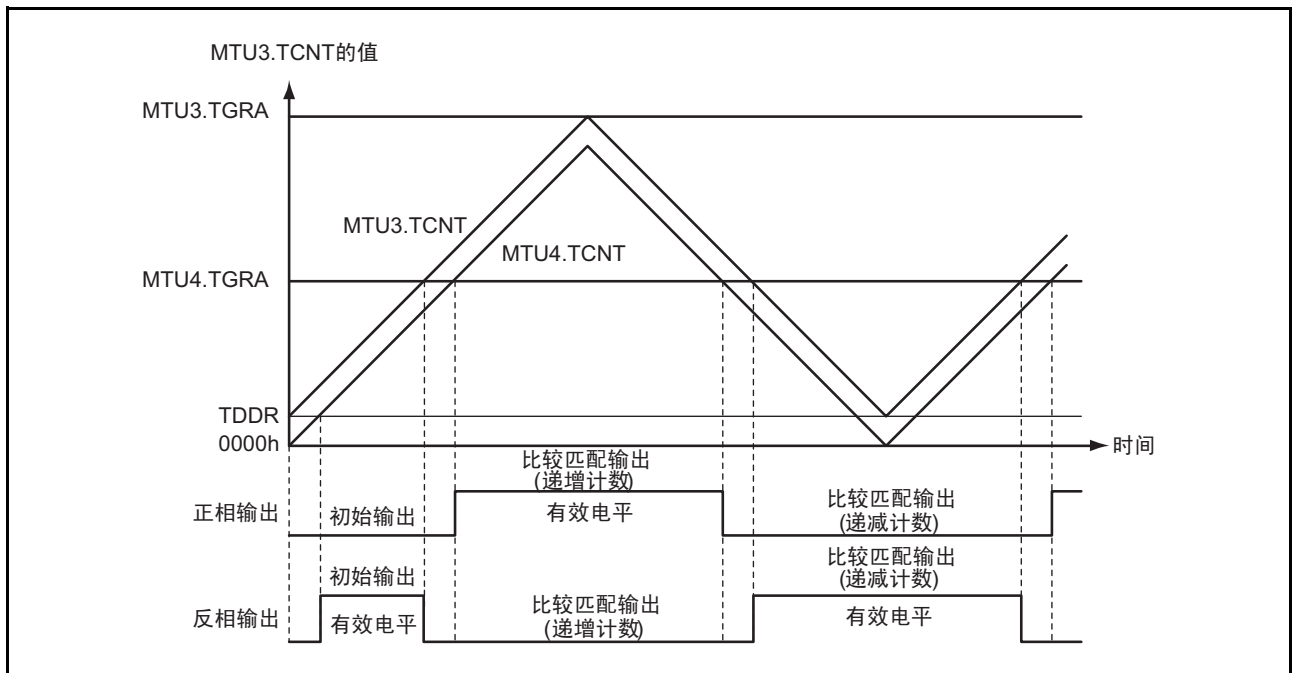
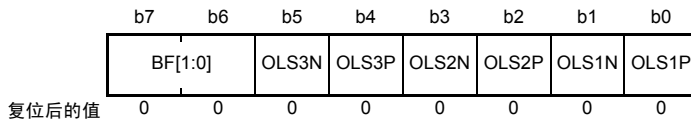


图 21.2 互补 PWM 模式的输出电平例子

21.2.19 定时器的输出控制寄存器 2 (TOCR2)

地址 MTU.TOCR2 0008 860Fh



位	符号	位名	功能	R/W
b0	OLS1P	输出电平选择 1P 位 (注 1)	选择在复位同步 PWM 模式 / 互补 PWM 模式中的 MTIOC3B 的输出电平, 请参照表 21.32。	R/W
b1	OLS1N	输出电平选择 1N 位 (注 1)	选择在复位同步 PWM 模式 / 互补 PWM 模式中的 MTIOC3D 的输出电平, 请参照表 21.33。	R/W
b2	OLS2P	输出电平选择 2P 位 (注 1)	选择在复位同步 PWM 模式 / 互补 PWM 模式中的 MTIOC4A 的输出电平, 请参照表 21.34。	R/W
b3	OLS2N	输出电平选择 2N 位 (注 1)	选择在复位同步 PWM 模式 / 互补 PWM 模式中的 MTIOC4C 的输出电平, 请参照表 21.35。	R/W
b4	OLS3P	输出电平选择 3P 位 (注 1)	选择在复位同步 PWM 模式 / 互补 PWM 模式中的 MTIOC4B 的输出电平, 请参照表 21.36。	R/W
b5	OLS3N	输出电平选择 3N 位 (注 1)	选择在复位同步 PWM 模式 / 互补 PWM 模式中的 MTIOC4D 的输出电平, 请参照表 21.37。	R/W
b7-b6	BF[1:0]	TOLBR 缓冲传送时序选择位	选择 TOLBR 到 TOCR2 的缓冲传送时序, 详细内容请参照表 21.38。	R/W

注 1. 通过将 TOCR1.TOCS 位置“1”, 使此设定变为有效。

TOCR2 寄存器寄存器对互补 PWM 模式 / 复位同步 PWM 模式的 PWM 输出电平进行反相控制。

表 21.32 MTIOC3B 输出电平的选择功能

bit0	功能			
OLS1P	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	High 电平	Low 电平	Low 电平	High 电平
1	Low 电平	High 电平	High 电平	Low 电平

表 21.33 MTIOC3D 输出电平的选择功能

bit1	功能			
OLS1N	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	High 电平	Low 电平	High 电平	Low 电平
1	Low 电平	High 电平	Low 电平	High 电平

注. 在开始计数并且经过死区时间后, 反相波形的初始输出值变为有效电平。

表 21.34 MTIOC4A 输出电平的选择功能

bit2	功能			
OLS2P	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	High 电平	Low 电平	Low 电平	High 电平
1	Low 电平	High 电平	High 电平	Low 电平

表 21.35 MTIOC4C 输出电平的选择功能

bit3	功能			
OLS2N	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	High 电平	Low 电平	High 电平	Low 电平
1	Low 电平	High 电平	Low 电平	High 电平

注. 在开始计数并且经过死区时间后, 反相波形的初始输出值变为有效电平。

表 21.36 MTIOC4B 输出电平的选择功能

bit4	功能			
OLS3P	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	High 电平	Low 电平	Low 电平	High 电平
1	Low 电平	High 电平	High 电平	Low 电平

表 21.37 MTIOC4D 输出电平的选择功能

bit5	功能			
OLS3N	初始输出	有效电平	比较匹配输出	
			递增计数	递减计数
0	High 电平	Low 电平	High 电平	Low 电平
1	Low 电平	High 电平	Low 电平	High 电平

注. 在开始计数并且经过死区时间后, 反相波形的初始输出值变为有效电平。

表 21.38 TOCR2.BF[1:0] 位的设定

bit7	bit6	说明	
BF1	BF0	互补 PWM 模式	复位 PWM 模式
0	0	不从缓冲寄存器 (TOLBR) 传送到 TOCR2。	不从缓冲寄存器 (TOLBR) 传送到 TOCR2。
0	1	在 MTU4.TCNT 的波峰从缓冲寄存器 (TOLBR) 传送到 TOCR2。	在清除 MTU4.TCNT 计数器和 MTU3.TCNT 计数器时从缓冲寄存器 (TOLBR) 传送到 TOCR2。
1	0	在 MTU4.TCNT 的波谷从缓冲寄存器 (TOLBR) 传送到 TOCR2。	不能设定。
1	1	在 MTU4.TCNT 的波峰和波谷从缓冲寄存器 (TOLBR) 传送到 TOCR2。	不能设定。

21.2.20 定时器的输出电平缓冲寄存器 (TOLBR)

地址 MTU.TOLBR 0008 8636h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	OLS1P	输出电平选择 1P 位	必须给 TOCR2 的 OLS1P 位设定缓冲传送的值。	R/W
b1	OLS1N	输出电平选择 1N 位	必须给 TOCR2 的 OLS1N 位设定缓冲传送的值。	R/W
b2	OLS2P	输出电平选择 2P 位	必须给 TOCR2 的 OLS2P 位设定缓冲传送的值。	R/W
b3	OLS2N	输出电平选择 2N 位	必须给 TOCR2 的 OLS2N 位设定缓冲传送的值。	R/W
b4	OLS3P	输出电平选择 3P 位	必须给 TOCR2 的 OLS3P 位设定缓冲传送的值。	R/W
b5	OLS3N	输出电平选择 3N 位	必须给 TOCR2 的 OLS3N 位设定缓冲传送的值。	R/W
b7-b6	—	保留位	读写值都为“0”。	R/W

TOLBR 寄存器是 TOCR2 的缓冲寄存器，设定互补 PWM 模式 / 复位同步 PWM 模式的 PWM 输出电平。TOLBR 寄存器是 8 位可读写寄存器。

在进行缓冲运行的过程中设定 PWM 输出电平时的设定步骤例子如图 21.3 所示。

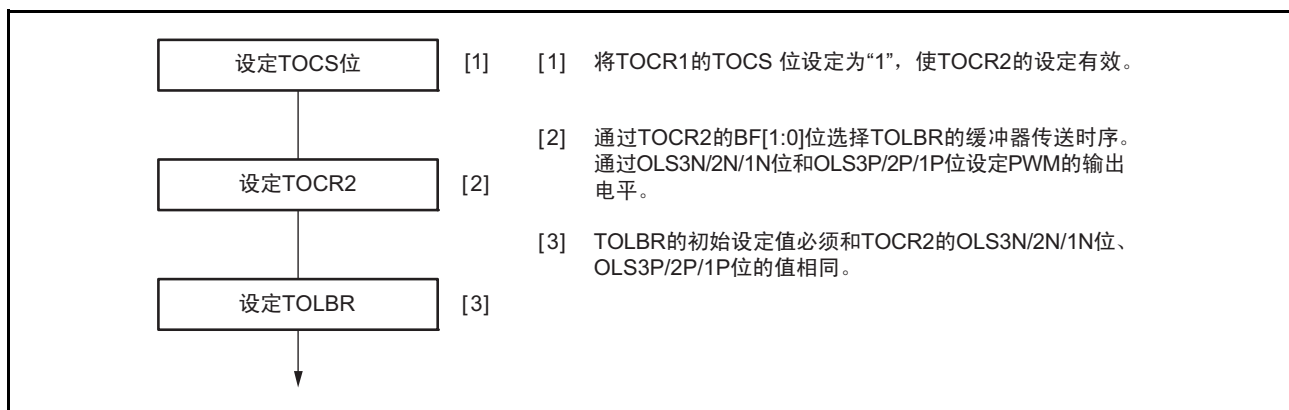


图 21.3 在进行缓冲运行的过程中设定 PWM 输出电平时的设定步骤例子

21.2.21 定时器的门控寄存器 (TGCR)

地址 MTU.TGCR 0008 860Dh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	BDC	N	P	FB	WF	VF	UF
复位后的值	1	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	UF	输出相转换位	设定正相 / 反相输出相的 ON/OFF。这些位的设定只在 TGCR.FB 位为“1”时有效。此时，b0 ~ b2 的设定取代外部输入，请参照表 21.39。	R/W
b1	VF			R/W
b2	WF			R/W
b3	FB	外部反馈信号允许位	0: 通过外部输入进行输出转换 (输入源为 MTU0 的 TGRA、TGRB、TGRC 的输入捕捉信号) 1: 通过软件进行输出转换 (TGCR 的 UF、VF 和 WF 的设定值)	R/W
b4	P	正相输出 (P) 控制位	0: 电平输出 1: 复位同步 PWM/ 互补 PWM 输出	R/W
b5	N	反相输出 (N) 控制位	0: 电平输出 1: 复位同步 PWM/ 互补 PWM 输出	R/W
b6	BDC	无刷 DC 马达位	0: 正常输出 1: 此寄存器的功能有效	R/W
b7	—	保留位	读写值都为“1”。	R/W

TGCR 寄存器在复位同步 PWM 模式 / 互补 PWM 模式中对控制无刷 DC 马达所需的波形输出进行控制。除了互补 PWM 模式 / 复位同步 PWM 模式以外，TGCR 寄存器的设定无效。

UF 位、VF 位、WF 位 (输出相转换位)

这些位的设定只在 TGCR.FB 位为“1”时有效。此时，bit0 ~ 2 的设定取代外部输入，请参照表 21.39。

FB 位 (外部反馈信号允许位)

此位选择是通过 MTU0.TGRA、MTU0.TGRB、MTU0.TGRC 的输入捕捉信号自动进行正反相输出的转换，还是通过给 TGCR 寄存器的 bit2 ~ 0 写“0”或者“1”进行正反相输出的转换。

P 位 (正相输出 (P) 控制位)

在正相引脚 (MTIOC3B 引脚、MTIOC4A 引脚、MTIOC4B 引脚) 输出时，此位选择是进行电平输出还是进行复位同步 PWM/ 互补 PWM 输出。

N 位 (反相输出 (N) 控制位)

在反相引脚 (MTIOC3D 引脚、MTIOC4C 引脚、MTIOC4D 引脚) 输出时，此位选择是进行电平输出还是进行复位同步 PWM/ 互补 PWM 输出。

BDC 位 (无刷 DC 马达位)

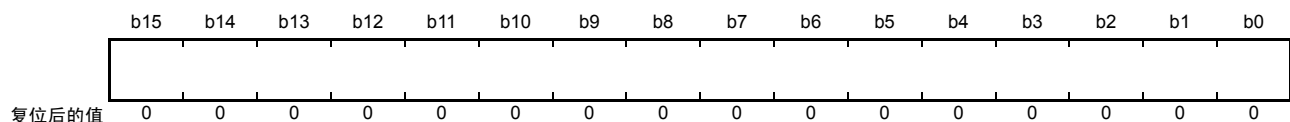
此位选择 TGCR 寄存器功能的有效或者无效。

表 21.39 输出电平的选择功能

bit2	bit1	bit0	功能					
			MTIOC3B	MTIOC4A	MTIOC4B	MTIOC3D	MTIOC4C	MTIOC4D
WF	VF	UF	U 相	V 相	W 相	U 相	V 相	W 相
0	0	0	OFF	OFF	OFF	OFF	OFF	OFF
0	0	1	ON	OFF	OFF	OFF	OFF	ON
0	1	0	OFF	ON	OFF	ON	OFF	OFF
0	1	1	OFF	ON	OFF	OFF	OFF	ON
1	0	0	OFF	OFF	ON	OFF	ON	OFF
1	0	1	ON	OFF	OFF	OFF	ON	OFF
1	1	0	OFF	OFF	ON	ON	OFF	OFF
1	1	1	OFF	OFF	OFF	OFF	OFF	OFF

21.2.22 定时器的副计数器 (TCNTS)

地址 MTU.TCNTS 0008 8620h



注：禁止以 8 位为单位存取 TCNTS 寄存器，必须以 16 位为单位进行存取。

TCNTS 寄存器是只用于互补 PWM 模式的 16 位只读计数器。复位后，TCNTS 寄存器的值为“0000h”。

21.2.23 定时器的死区时间数据寄存器 (TDDR)

地址 MTU.TDDR 0008 8616h

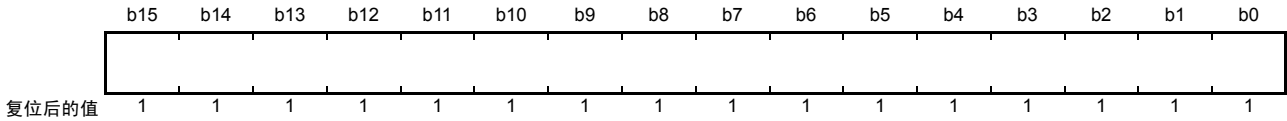


注：禁止以 8 位为单位存取 TDDR 寄存器，必须以 16 位为单位进行存取。

TDDR 寄存器是只用于互补 PWM 模式的 16 位寄存器，在互补 PWM 模式中设定 MTU3.TCNT 计数器和 MTU4.TCNT 计数器的偏移值。在互补 PWM 模式中清除 MTU3.TCNT 计数器和 MTU4.TCNT 计数器后重新开始计数时，将 TDDR 寄存器的值装入到 MTU3.TCNT 计数器并且开始计数。复位后，TDDR 寄存器的值为“FFFFh”。

21.2.24 定时器的周期数据寄存器 (TCDR)

地址 MTU.TCDR 0008 8614h

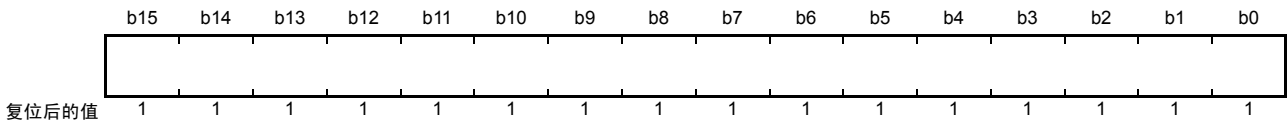


注：禁止以 8 位为单位存取 TCDR 寄存器，必须以 16 位为单位进行存取。

TCDR 寄存器是只用于互补 PWM 模式的 16 位寄存器，必须给 TCDR 寄存器设定 1/2 个 PWM 载波周期的值。在互补 PWM 模式中，TCDR 寄存器随时和 TCNTS 计数器进行比较，如果两者的值相同，TCNTS 计数器就转换计数方向（递减计数 → 递增计数）。复位后，TCDR 寄存器的值为“FFFFh”。

21.2.25 定时器的周期缓冲寄存器 (TCBR)

地址 MTU.TCBR 0008 8622h

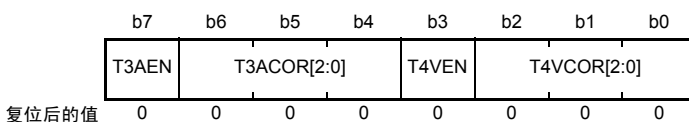


注：禁止以 8 位为单位存取 TCBR 寄存器，必须以 16 位为单位进行存取。

TCBR 寄存器是只用于互补 PWM 模式的 16 位寄存器，用作 TCDR 寄存器的缓冲寄存器。在 TMDR 寄存器设定的传送时将 TCBR 寄存器的值传送到 TCDR 寄存器。复位后，TCBR 寄存器的值为“FFFFh”。

21.2.26 定时器的中断减少设定寄存器 (TITCR)

地址 MTU.TITCR 0008 8630h



位	符号	位名	功能	R/W
b2-b0	T4VCOR[2:0]	TCIV4 中断减少次数设定位	将 TCIV4 的中断减少次数设定为 0 ~ 7 次，详细内容请参照表 21.40。	R/W
b3	T4VEN	T4VEN 位	0: 禁止 TCIV4 中断的减少 1: 允许 TCIV4 中断的减少	R/W
b6-b4	T3ACOR[2:0]	TGIA3 中断减少次数设定位	将 TGIA3 的中断减少次数设定为 0 ~ 7 次 (注 1)，详细内容请参照表 21.41。	R/W
b7	T3AEN	T3AEN 位	0: 禁止 TGIA3 中断的减少 1: 允许 TGIA3 中断的减少	R/W

注 1. 如果将中断减少次数设定为“0”，就不减少中断。

另外，必须在更改中断减少次数前，通过将 TITCR.T3AEN 位和 TITCR.T4VEN 位置“0”，清除减少次数计数器 (TITCNT)。

T4VCOR[2:0] 位 (TCIV4 的中断减少次数设定位)

T3ACOR[2:0] 位 (TGIA3 的中断减少次数设定位)

将 TCIV3 和 TGIA4 的中断减少次数设定为 0 ~ 7 次, 详细内容请参照表 21.40 和表 21.41。

表 21.40 通过 T4VCOR[2:0] 位设定的中断减少次数

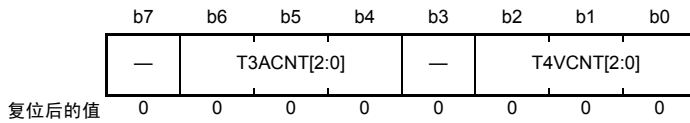
bit2	bit1	bit0	说明
T4VCOR2	T4VCOR1	T4VCOR0	
0	0	0	不减少 TCIV4 的中断。
0	0	1	将 TCIV4 的中断减少次数设定为 1 次。
0	1	0	将 TCIV4 的中断减少次数设定为 2 次。
0	1	1	将 TCIV4 的中断减少次数设定为 3 次。
1	0	0	将 TCIV4 的中断减少次数设定为 4 次。
1	0	1	将 TCIV4 的中断减少次数设定为 5 次。
1	1	0	将 TCIV4 的中断减少次数设定为 6 次。
1	1	1	将 TCIV4 的中断减少次数设定为 7 次。

表 21.41 通过 T3ACOR[2:0] 位设定的中断减少次数

bit6	bit5	bit4	说明
T3ACOR2	T3ACOR1	T3ACOR0	
0	0	0	不减少 TGIA3 的中断。
0	0	1	将 TGIA3 的中断减少次数设定为 1 次。
0	1	0	将 TGIA3 的中断减少次数设定为 2 次。
0	1	1	将 TGIA3 的中断减少次数设定为 3 次。
1	0	0	将 TGIA3 的中断减少次数设定为 4 次。
1	0	1	将 TGIA3 的中断减少次数设定为 5 次。
1	1	0	将 TGIA3 的中断减少次数设定为 6 次。
1	1	1	将 TGIA3 的中断减少次数设定为 7 次。

21.2.27 定时器的中断减少次数计数器 (TITCNT)

地址 MTU.TITCNT 0008 8631h



位	符号	位名	功能	R/W
b2-b0	T4VCNT[2:0]	TCIV4 中断计数位	如果将 TITCR 的 T4VEN 位置“1”，就在发生 TCIV4 中断源时递增 1。	R
b3	—	保留位	读取值为“0”，写操作无效。	R
b6-b4	T3ACNT[2:0]	TGIA3 中断计数位	如果将 TITCR 的 T3AEN 位置“1”，就在发生 TGIA3 中断源时递增 1。	R
b7	—	保留位	读取值为“0”，写操作无效。	R

注. 要清除 TITCNT 的值时，必须将 TITCR.T3AEN 位和 TITCR.T4VEN 位置“0”。

TITCNT 是 8 位可读计数器。在 MTU3.TCNT 和 MTU4.TCNT 停止计数后，TITCNT 仍保持原来的值。

T4VCNT[2:0] 位 (TCIV4 中断计数位)

[为“0”的条件]

- 当 TITCR 的 T4VCOR[2:0]位和 TITCNT 的 T4VCNT[2:0]位相同时
- 当 TITCR 的 T4VEN 位为“0”时
- 当 TITCR 的 T4VCOR[2:0]位为“000b”时

T3ACNT[2:0] 位 (TGIA3 中断计数位)

[为“0”的条件]

- 当 TITCR 的 T3ACOR[2:0]位和 TITCNT 的 T3ACNT[2:0]位相同时
- 当 TITCR 的 T3AEN 位为“0”时
- 当 TITCR 的 T3ACOR[2:0]位为“000b”时

21.2.28 定时器的缓冲传送设定寄存器 (TBTER)

地址 MTU.TBTER 0008 8632h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	BTE[1:0]	
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b1-b0	BTE[1:0]	缓冲传送抑止和中断减少联动设定位	此位设定是否抑止用于互补 PWM 模式的缓冲寄存器到暂存器的传送以及是否联动中断减少功能，详细内容请参照表 21.42。	R/W
b7-b2	—	保留位	读写值都为“0”。	R/W

TBTER 寄存器是 8 位可读写寄存器，设定是否抑止用于互补 PWM 模式的缓冲寄存器到暂存器的传送以及是否联动中断减少功能。

表 21.42 TBTER.BTE[1:0] 位的设定

bit1	bit0	说明
BTE1	BTE0	
0	0	不抑止缓冲寄存器到暂存器的传送 (注 1)，也不联动中断减少功能。
0	1	抑止缓冲寄存器到暂存器的传送。
1	0	缓冲寄存器到暂存器的传送联动中断减少功能 (注 2)。
1	1	不能设定。

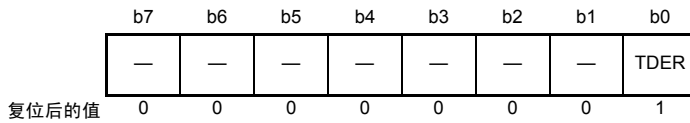
注 1. 根据 TMDR 的 MD[3:0] 位的设定进行传送，详细内容请参照“21.3.8 互补 PWM 模式”。

注 2. 在禁止中断减少功能时 (将定时器的中断减少设定寄存器 (TITCR) 的 T3AEN 位和 T4VEN 位置“0”，或者将 TITCR 的减少次数设定位 (T3ACOR 和 T4VCOR) 置“0”时)，必须设定为缓冲传送不联动中断减少功能 (将定时器缓冲传送寄存器 (TBTER) 的 BTE1 位置“0”)。

在禁止中断减少功能时，如果设定为缓冲传送联动中断减少功能，就不进行缓冲传送。

21.2.29 定时器的死区时间允许寄存器 (TDER)

地址 MTU.TDER 0008 8634h



位	符号	位名	功能	R/W
b0	TDER	死区时间允许寄存器位	0: 不生成死区时间 1: 生成死区时间 (注1)	R/(W) (注2)
b7-b1	—	保留位	读写值都为“0”。	R/W

注 1. 必须设定为 $TDDR \geq 1$ 。注 2. 在互补 PWM 模式中，当写“1”时，需要事先读到“0”；当写“0”时，没有限制。
在非互补 PWM 模式中，禁止写“1”，只能写“0”。

TDER 寄存器是 8 位可读写寄存器，MTU3 有 1 个 TDER 寄存器，能控制互补 PWM 模式的死区时间的生成。必须在 TCNT 停止计数的状态下设定 TDER 寄存器。

TDER 位 (死区时间允许寄存器位)

此位设定是否生成死区时间。

[为“0”的条件]

- 在 TDER 为“1”的状态下读 TDER 位后给 TDER 位写“0”时

21.2.30 定时器的波形控制寄存器 (TWCR)

地址 MTU.TWCR 0008 8660h

	b7	b6	b5	b4	b3	b2	b1	b0
	CCE	—	—	—	—	—	—	WRE
复位后的值	0 (注1)	0	0	0	0	0	0	0

注 1. 除了互补 PWM 模式以外，不能置“1”。

位	符号	位名	功能	R/W
b0	WRE	初始输出抑止允许位	0: 输出 TOCR 寄存器设定的初始输出值 1: 抑止初始输出	R/(W) (注 1)
b6-b1	—	保留位	读写值都为“0”。	R/W
b7	CCE	比较匹配清除允许位	0: 不通过 MTU3.TGRA 的比较匹配清除计数器 1: 通过 MTU3.TGRA 的比较匹配清除计数器	R/(W)

注 1. 在互补 PWM 模式中，当写“1”时，需要事先读到“0”；当写“0”时，没有限制。
在非互补 PWM 模式中，禁止写“1”，只能写“0”。

TWCR 寄存器是 8 位可读写寄存器。TWCR 寄存器控制在互补 PWM 模式中发生 MTU3.TNCT 和 MTU4.TNCT 的同步计数清除时的输出波形以及设定是否通过 MTU3.TGRA 的比较匹配清除计数器。

必须在 TCNT 停止计数的状态下设定 TWCR.CCE 位和 TWCR.WRE 位。

WRE 位 (初始输出抑止允许位)

此位选择在互补 PWM 模式中发生同步计数清除时的输出波形。

只有在互补 PWM 模式的波谷 Tb 区间发生同步清除时，才能通过此功能抑止初始输出。如果在其他区间发生同步清除，就输出 TOCR 寄存器设定的初始值，与 TWCR.WRE 位的设定无关。如果在 MTU3.TCNT 和 MTU4.TCNT 开始计数后的波谷 Tb 区间发生同步清除，也输出 TOCR 寄存器设定的初始值。

有关互补 PWM 模式的波谷 Tb 区间，请参照图 21.40。

[为“1”的条件]

- 在 TWCR.WRE 为“0”的状态下读 TWCR.WRE 位后给 TWCR.WRE 位写“1”时

CCE 位 (比较匹配清除允许位)

此位设定在互补 PWM 模式中是否通过 TGRA3 的比较匹配清除计数器。

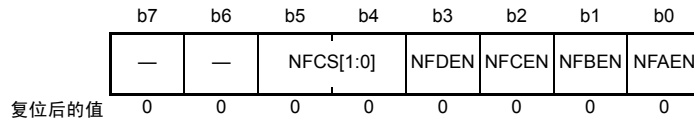
[为“1”的条件]

- 在 CCE 为“0”的状态下读 CCE 位后给 CCE 位写“1”时

21.2.31 噪声滤波器的控制寄存器 (NFCR)

• NFCR (MTU0 ~ MTU4)

地址 MTU0.NFCR 0008 8690h、MTU1.NFCR 0008 8691h、MTU2.NFCR 0008 8692h、MTU3.NFCR 0008 8693h、MTU4.NFCR 0008 8694h



位	符号	位名	功能	R/W
b0	NFAEN	噪声滤波器 A 允许位	0: 停止 MTIOCnA 引脚的噪声滤波器 1: 允许 MTIOCnA 引脚的噪声滤波器	R/W
b1	NFBEN	噪声滤波器 B 允许位	0: 停止 MTIOCnB 引脚的噪声滤波器 1: 允许 MTIOCnB 引脚的噪声滤波器	R/W
b2	NFCEN	噪声滤波器 C 允许位	0: 停止 MTIOCnC 引脚的噪声滤波器 1: 允许 MTIOCnC 引脚的噪声滤波器	R/(W) (注 1)
b3	NFDEN	噪声滤波器 D 允许位	0: 停止 MTIOCnD 引脚的噪声滤波器 1: 允许 MTIOCnD 引脚的噪声滤波器	R/(W) (注 1)
b5-b4	NFCS[1:0]	噪声滤波器时钟选择位	0 0: PCLK/1 0 1: PCLK/8 1 0: PCLK/32 1 1: 计数源	R/W
b7-b6	—	保留位	读写值都为“0”。	R/W

注 1. 在 MTU1.NFCR 寄存器和 MTU2.NFCR 寄存器中，为保留位，读取值为“0”，写入值无效。

MTUn.NFCR 寄存器 (n=0 ~ 4) 是 8 位可读写寄存器，控制允许或者停止 MTIOCnm 引脚的噪声滤波器，并且设定噪声滤波器的采样时钟 (n=0 ~ 4、m=A ~ D)。

NFAEN 位 (噪声滤波器 A 允许位)

此位设定允许或者停止 MTIOCnA 引脚的输入噪声滤波器。因为在更改 NFAEN 位时可能发生意想不到的内部边沿，所以必须在将定时器的 I/O 控制寄存器的该引脚功能设定为输出比较功能的状态下或者在通过 TMDR.MD[3:0] 位设定为非正常模式 (“0000b” 以外的值) 的状态下更改 NFAEN 位。

NFBEN 位 (噪声滤波器 B 允许位)

此位设定允许或者停止 MTIOCnB 引脚的输入噪声滤波器。因为在更改 NFBEN 位时可能发生意想不到的内部边沿，所以必须在将定时器的 I/O 控制寄存器的该引脚功能设定为输出比较功能的状态下或者在通过 TMDR.MD[3:0] 位设定为非正常模式 (“0000b” 以外的值) 的状态下更改 NFBEN 位。

NFCEN 位 (噪声滤波器 C 允许位)

此位设定允许或者停止 MTIOCnC 引脚的输入噪声滤波器。因为在更改 NFCEN 位时可能发生意想不到的内部边沿，所以必须在将定时器的 I/O 控制寄存器的该引脚功能设定为输出比较功能的状态下或者在通过 TMDR.MD[3:0] 位设定为非正常模式 (“0000b” 以外的值) 的状态下更改 NFCEN 位。

NFDEN 位 (噪声滤波器 D 允许位)

此位设定允许或者停止 MTIOcND 引脚的输入噪声滤波器。因为在更改 NFDEN 位时可能发生意想不到的内部边沿, 所以必须在将定时器的 I/O 控制寄存器的该引脚功能设定为输出比较功能的状态下或者在通过 TMDR.MD[3:0] 位设定为非正常模式 (“0000b” 以外的值) 的状态下更改 NFDEN 位。

NFCS[1:0] 位 (噪声滤波器时钟选择位)

这些位设定噪声滤波器的采样周期。必须在设定 NFCS 位后等待 2 个所设采样周期, 然后设定为输入捕捉功能。如果通过将 NFCS[1:0] 位置 “11b” 使计数源为外部时钟, 就必须在设定 NFCS 位后等待 2 次外部时钟的输入, 然后设定为输入捕捉功能。

• **NFCR (MTU5)**

地址 MTU5.NFCR 0008 8695h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	NFCS[1:0]	—	NFWEN	NFVEN	NFUEN	
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	NFUEN	噪声滤波器 U 允许位	0: 停止 MTIC5U 引脚的噪声滤波器 1: 允许 MTIC5U 引脚的噪声滤波器	R/W
b1	NFVEN	噪声滤波器 V 允许位	0: 停止 MTIC5V 引脚的噪声滤波器 1: 允许 MTIC5V 引脚的噪声滤波器	R/W
b2	NFWEN	噪声滤波器 W 允许位	0: 停止 MTIC5W 引脚的噪声滤波器 1: 允许 MTIC5W 引脚的噪声滤波器	R/W
b3	—	保留位	读写值都为“0”。	R/W
b5-b4	NFCS[1:0]	噪声滤波器时钟选择位	0 0: PCLK/1 0 1: PCLK/8 1 0: PCLK/32 1 1: 计数源	R/W
b7-b6	—	保留位	读写值都为“0”。	R/W

MTU5.NFCR 寄存器是 8 位可读写寄存器, 控制允许或者停止 MTIC5m 引脚的噪声滤波器, 并且设定噪声滤波器的采样时钟 (m=U、V、W)。

NFUEN 位 (噪声滤波器 U 允许位)

此位设定允许或者停止 MTIC5U 引脚的输入噪声滤波器。因为在更改 NFUEN 位时可能发生意想不到的内部边沿, 所以必须在将定时器的 I/O 控制寄存器的该引脚功能设定为比较匹配功能的状态下更改 NFUEN 位。

NFVEN 位 (噪声滤波器 V 允许位)

此位设定允许或者停止 MTIC5V 引脚的输入噪声滤波器。因为在更改 NFVEN 位时可能发生意想不到的内部边沿, 所以必须在将定时器的 I/O 控制寄存器的该引脚功能设定为比较匹配功能的状态下更改 NFVEN 位。

NFWEN 位 (噪声滤波器 W 允许位)

此位设定允许或者停止 MTIC5W 引脚的输入噪声滤波器。因为在更改 NFWEN 位时可能发生意想不到的内部边沿, 所以必须在将定时器的 I/O 控制寄存器的该引脚功能设定为比较匹配功能的状态下更改 NFWEN 位。

NFCS[1:0] 位 (噪声滤波器时钟选择位)

这些位设定噪声滤波器的采样周期。必须在设定 NFCS[1:0] 位后等待 2 个所设采样周期, 然后设定为输入捕捉功能。

21.2.32 和总线主控的接口

定时器的计数器 (TCNT)、通用寄存器 (TGR)、定时器的副计数器 (TCNTS)、定时器的周期缓冲寄存器 (TCBR)、定时器的死区时间数据寄存器 (TDDR)、定时器的周期数据寄存器 (TCDR)、定时器的 A/D 转换开始请求控制寄存器 (TADCR)、定时器的 A/D 转换开始请求周期设定寄存器 (TADCORA/B) 和定时器的 A/D 转换开始请求周期设定缓冲寄存器 (TADCOBRA/B) 都是 16 位寄存器。因为和总线主控之间的数据总线宽度为 16 位, 所以能以 16 位为单位进行读写, 而不能以 8 位为单位进行读写。必须总是以 16 位为单位进行存取。

上述以外的寄存器是 8 位寄存器, 必须以 8 位为单位进行读写。

21.3 运行说明

21.3.1 概要

各通道有 TCNT 和 TGR。TCNT 进行递增计数，能进行自由运行、周期计数器运行或者外部事件计数运行。TGR 能分别用作输入捕捉寄存器或者输出比较寄存器。

(1) 计数器的运行

如果将 TSTR 的 CST0 ~ CST4 位以及 MTU5.TSTR 的 CSTU5 位、CSTV5 位和 CSTW5 位置“1”，对应通道的 TCNT 就开始计数。能用作自由运行计数器和周期计数器等。

(a) 计数运行的设定步骤例子

计数器运行的设定步骤例子如图 21.4 所示。

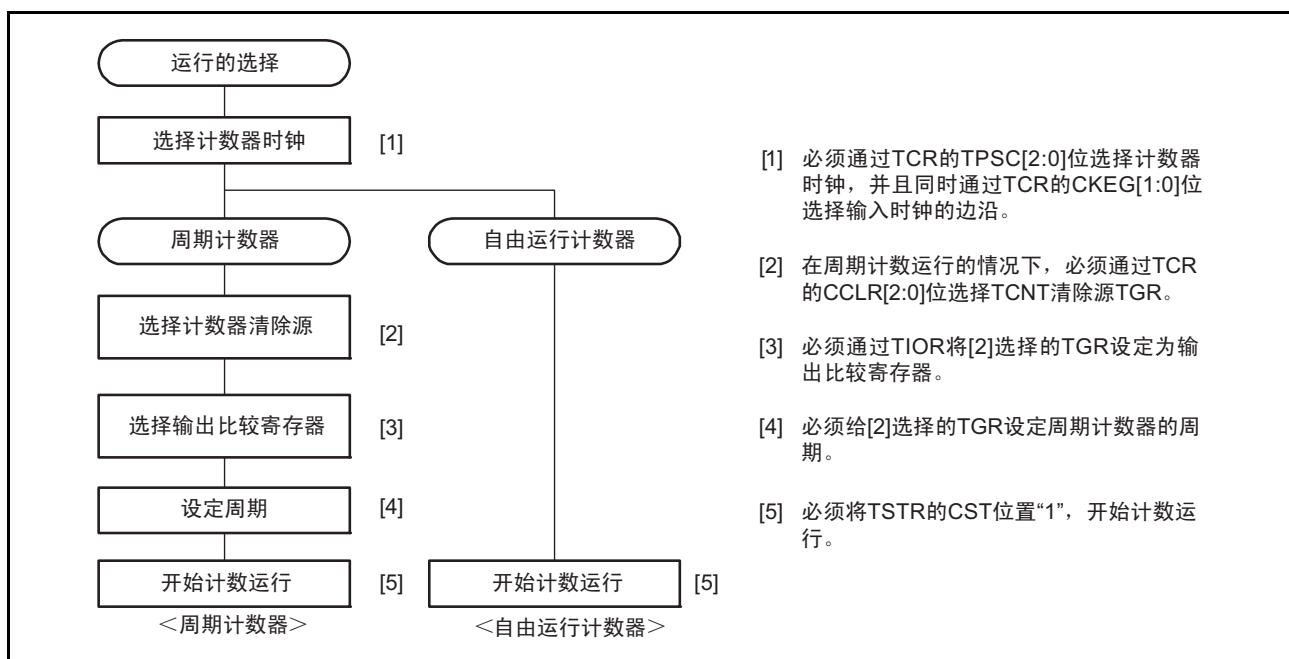


图 21.4 计数器运行的设定步骤例子

(b) 自由运行计数的运行和周期计数的运行

MTU 的 TCNT 在复位后立即被全部设定为自由运行计数器。如果将 TSTR 的对应位置“1”，就作为自由运行计数器开始递增计数。如果 TCNT 发生上溢 (“FFFFh”→“0000h”)并且对应的 TIER 的 TCIEV 位为“1”，MTU 就请求中断。TCNT 在发生上溢后从“0000h”开始继续递增计数。

自由运行计数器的运行如图 21.5 所示。

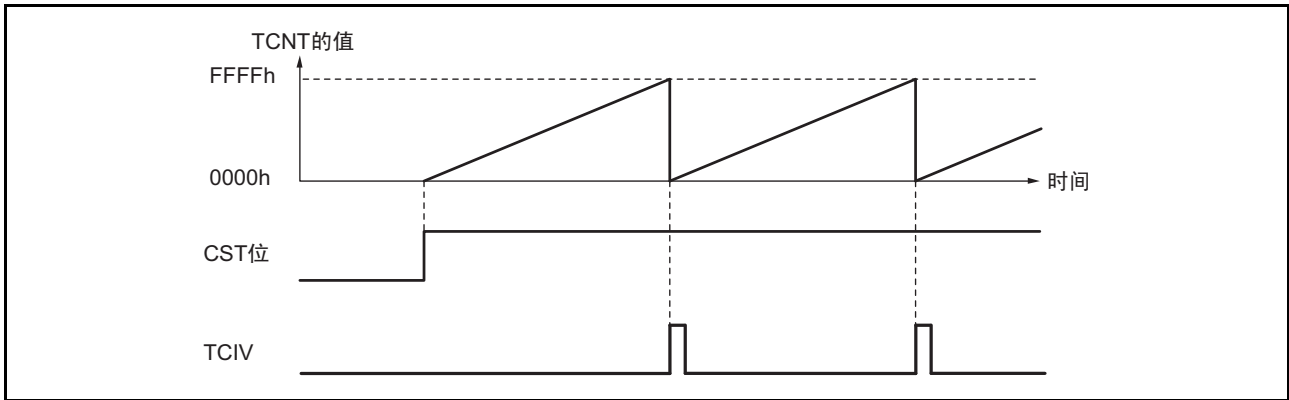


图 21.5 自由运行计数器的运行

当选择比较匹配作为 TCNT 的清除源时，对应通道的 TCNT 进行周期计数。将用于设定周期的 TGR 设定为输出比较寄存器，并且由 TCR 的 CCLR[2:0] 位选择通过比较匹配进行计数器清除。如果在设定后将 TSTR 的对应位置“1”，就作为周期计数器开始递增计数。如果计数值和 TGR 的值相同，TCNT 就变为“0000h”。

此时，如果对应的 TIER 的 TGIE 位为“1”，MTU 就请求中断。TCNT 在比较匹配后从“0000h”开始继续递增计数。

周期计数器的运行如图 21.6 所示。

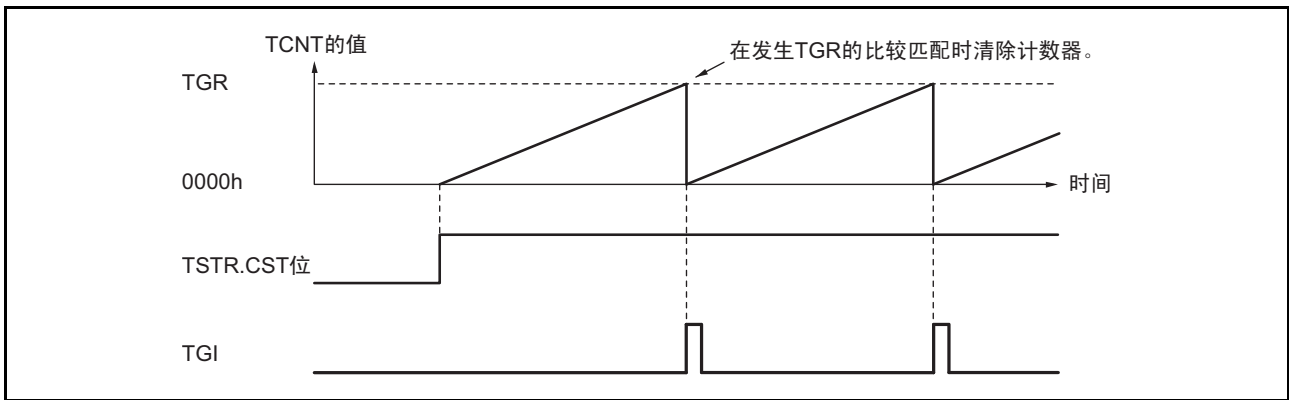


图 21.6 周期计数器的运行

(2) 通过比较匹配进行的波形输出功能

MTU 能通过比较匹配从对应的输出引脚输出 Low 电平、High 电平或者进行交替输出。

(a) 通过比较匹配进行波形输出的设定步骤例子

通过比较匹配进行波形输出的设定步骤例子如图 21.7 所示。

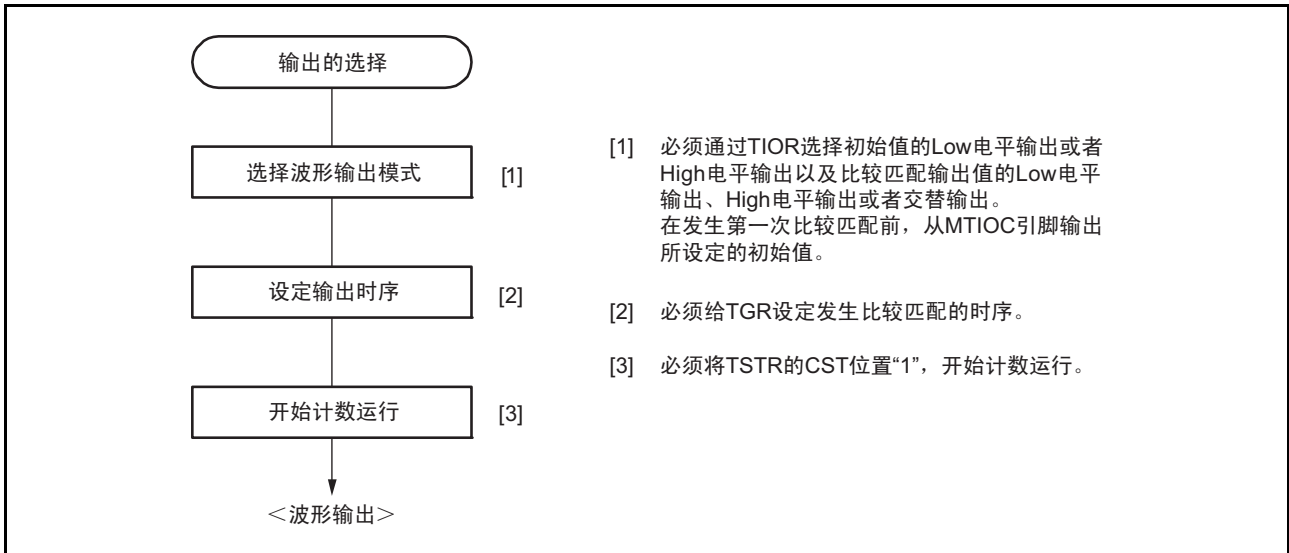


图 21.7 通过比较匹配进行波形输出的运行例子

(b) 波形输出的运行例子

输出 Low 电平 /High 电平的运行例子如图 21.8 所示。

在此例子中，假设 TCNT 进行自由运行的计数，在比较匹配 A 时输出 High 电平，在比较匹配 B 时输出 Low 电平。如果设定的电平和引脚的电平相同，引脚的电平就不变。

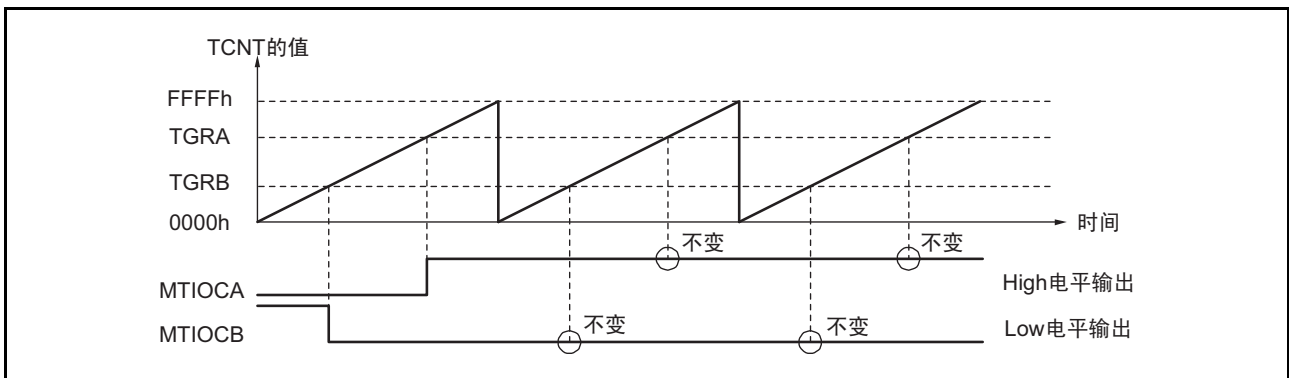


图 21.8 输出 Low 电平 /High 电平的运行例子

交替输出的运行例子如图 21.9 所示。

在此例子中，假设 TCNT 进行周期计数（通过比较匹配 B 进行计数器清除），并且比较匹配 A 和比较匹配 B 都进行交替输出。

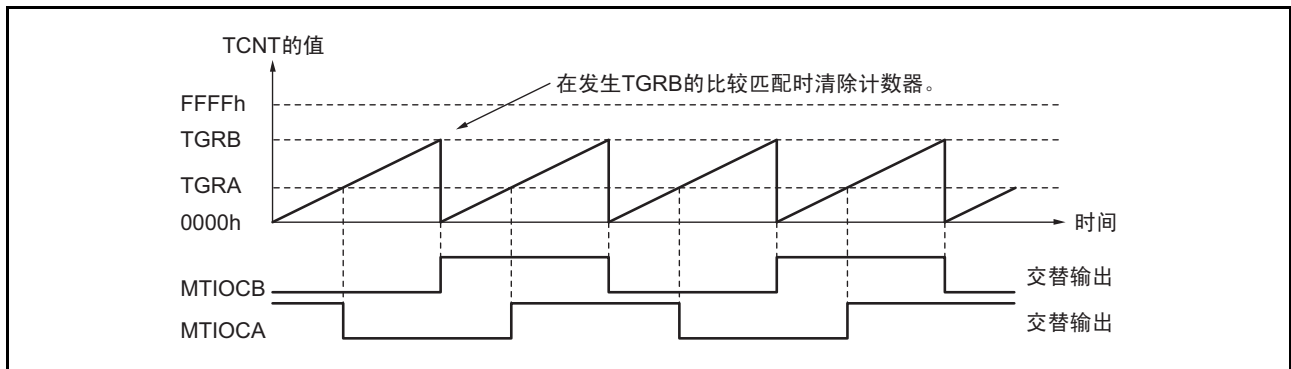


图 21.9 交替输出的运行例子

(3) 输入捕捉功能

能在检测到 MTIOC 引脚的输入边沿后将 TCNT 的值传送到 TGR。

检测边沿可选择上升沿、下降沿或者双边沿，而在 MTU0 和 MTU1 中也能将其他通道的计数器输入时钟或者比较匹配信号作为输入捕捉源。

注 . 如果在 MTU0 和 MTU1 中将其他通道的计数器输入时钟作为输入捕捉的输入引脚，就不能选择 PCLK/1 作为输入捕捉输入的计数器输入时钟。如果选择 PCLK/1，就不产生输入捕捉。

(a) 输入捕捉的设定步骤例子

输入捕捉的设定步骤例子如图 21.10 所示。

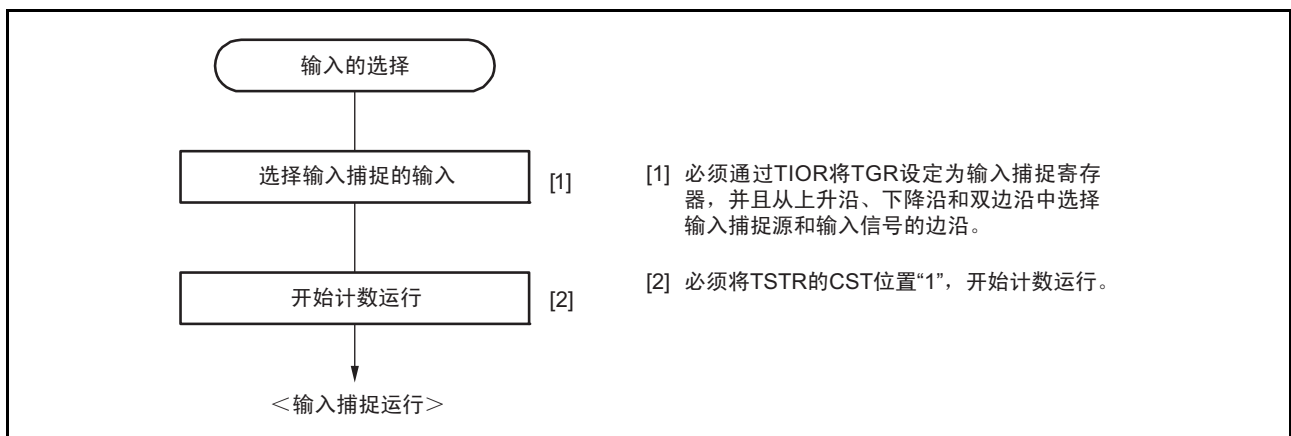


图 21.10 输入捕捉的设定例子

(b) 输入捕捉的运行例子

输入捕捉的运行例子如图 21.11 所示。

在此例子中，假设选择上升沿 / 下降沿的双边沿作为 MTIOCnA 引脚的输入捕捉的输入边沿，选择下降沿作为 MTIOCnB 引脚的输入捕捉的输入边沿，并且在发生 TGRB 的输入捕捉时清除 TCNT 计数器。

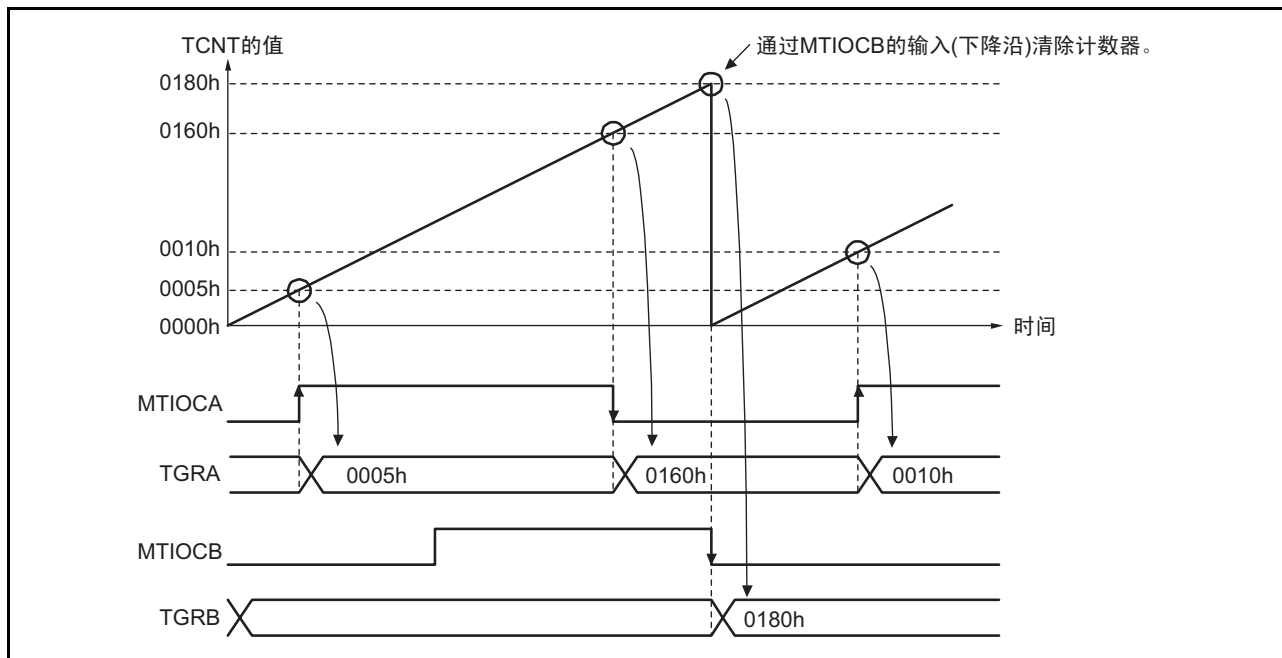


图 21.11 输入捕捉的运行例子

21.3.2 同步运行

同步运行能同时改写多个 TCNT 的值 (同步预置), 还能通过设定 TCR 同时清除多个 TCNT (同步清除)。

能通过同步运行对 1 个时基增加要运行的 TGR 个数。

MTU0 ~ MTU4 都能设定为同步运行。

MTU5 不能设定为同步运行。

(1) 同步运行的设定步骤例子

同步运行的设定步骤例子如图 21.12 所示。

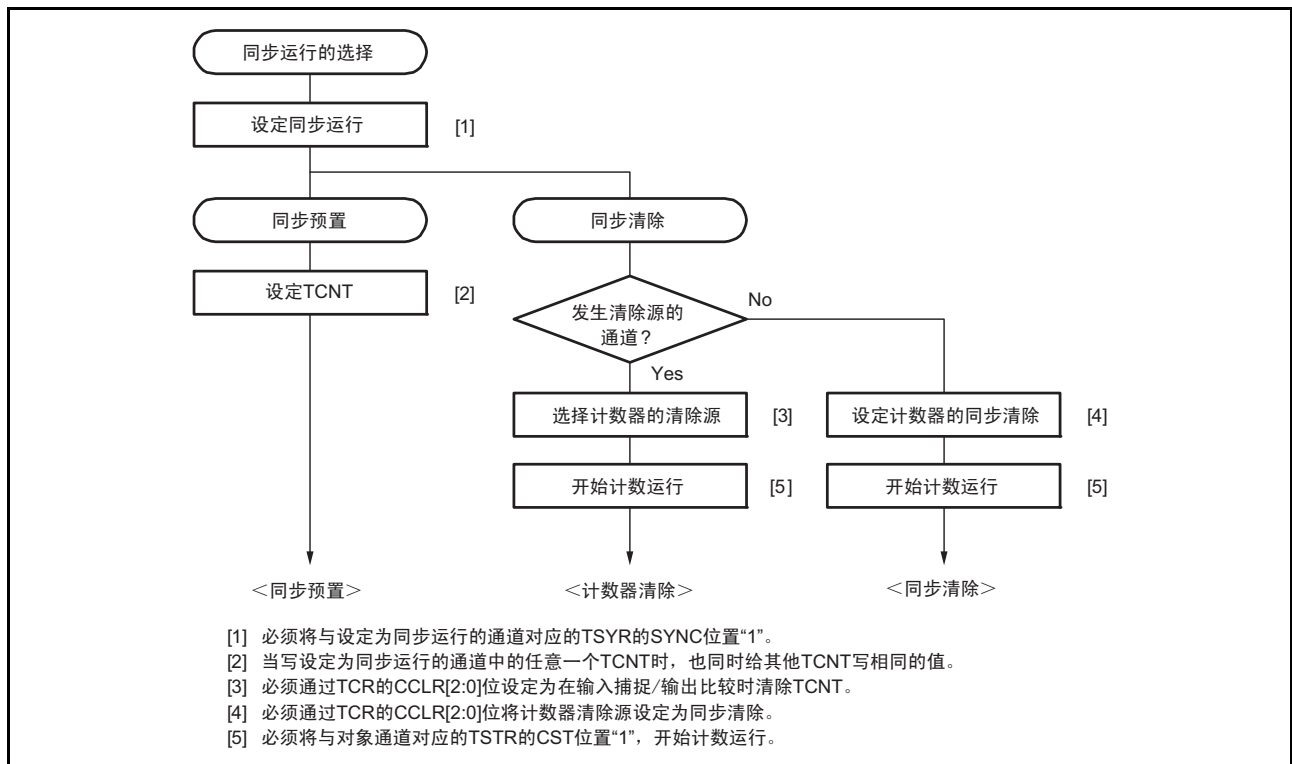


图 21.12 同步运行的设定步骤例子

(2) 同步运行的例子

同步运行的例子如图 21.13 所示。

在此例子中，假设将 MTU0~2 设定为同步运行和 PWM 模式 1，将 MTU0 的计数器清除源设定为 MTU0.TGRB 的比较匹配，将 MTU1 和 MTU2 的计数器清除源设定为同步清除。

从 MTIOC0A、MTIOC1A、MTIOC2A 引脚输出 3 相 PWM 波形。此时，MTU0 ~ MTU2 的 TCNT 进行同步预置并且通过 MTU0.TGRB 的比较匹配进行同步清除，MTU0.TGRB 设定的数据为 PWM 周期。

有关 PWM 模式，请参照“21.3.5 PWM 模式”。

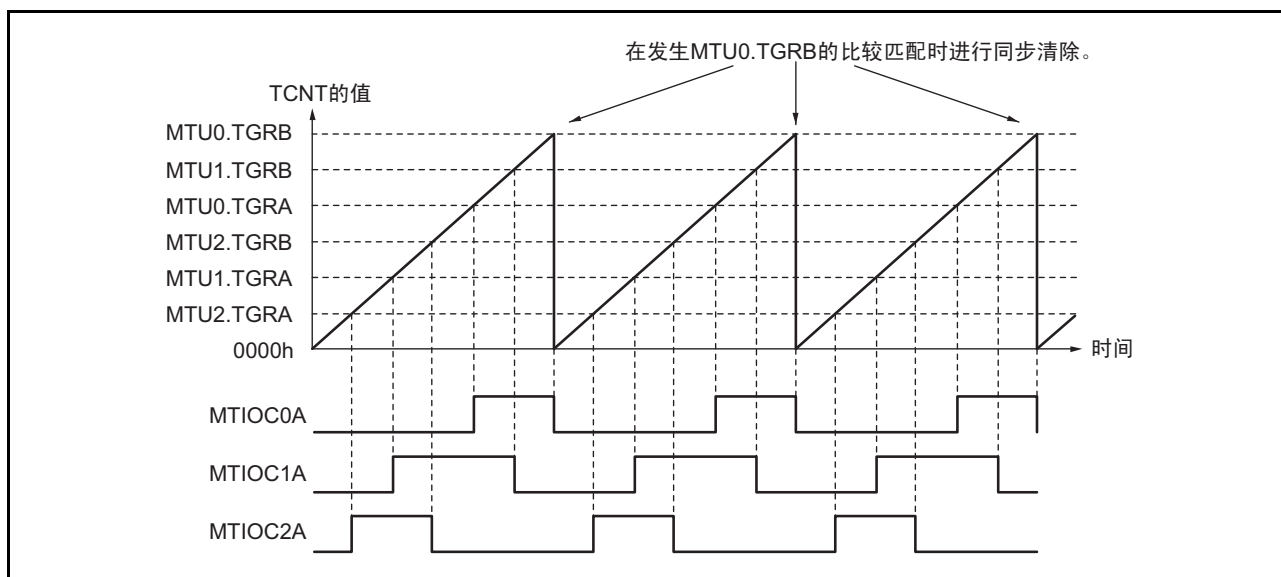


图 21.13 同步运行的例子

21.3.3 缓冲运行

缓冲运行是 MTU0、MTU3 和 MTU4 具有的功能，能将 TGRC 和 TGRD 用作缓冲寄存器，在 MTU0 中也能将 TGRF 用作缓冲寄存器。

将 TGR 设定为输入捕捉寄存器和比较匹配寄存器时的缓冲运行内容不同。

注. MTU0.TGRE 不能被设定为输入捕捉寄存器，而只能用作比较匹配寄存器。

缓冲运行时的寄存器组合如表 21.43 所示。

表 21.43 寄存器的组合

通道	定时器的通用寄存器	缓冲寄存器
MTU0	TGRA	TGRC
	TGRB	TGRD
	TGRE	TGRF
MTU3	TGRA	TGRC
	TGRB	TGRD
MTU4	TGRA	TGRC
	TGRB	TGRD

- TGR为输出比较寄存器的情况

如果发生比较匹配，就将对应通道的缓冲寄存器的值传送到定时器的通用寄存器。此运行如图 21.14 所示。

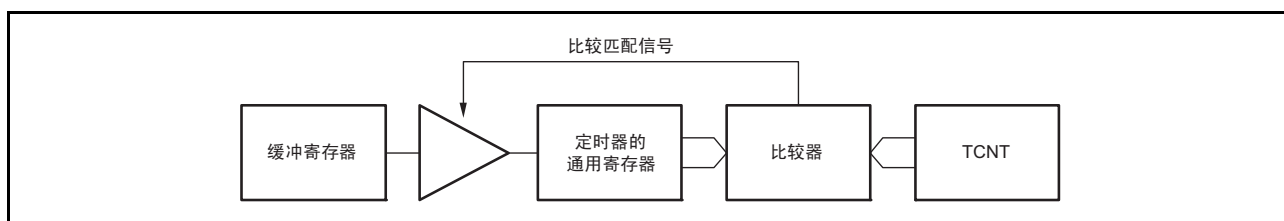


图 21.14 比较匹配的缓冲运行

- TGR为输入捕捉寄存器的情况

如果发生输入捕捉，就在将 TCNT 的值传送到 TGR 的同时，将以前保存的 TGR 值传送到缓冲寄存器。此运行如图 21.15 所示。

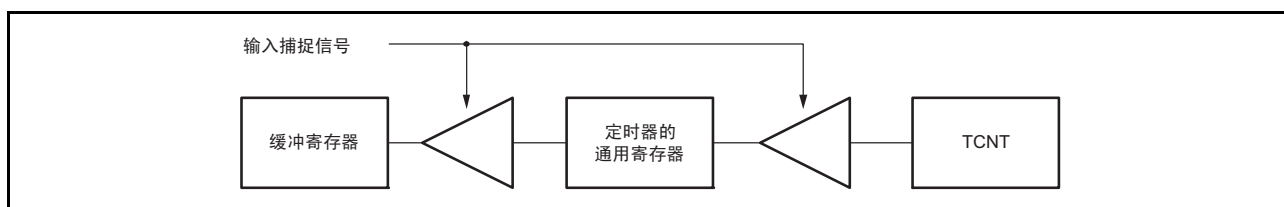


图 21.15 输入捕捉的缓冲运行

(1) 缓冲运行的设定步骤例子

缓冲运行的设定步骤例子如图 21.16 所示。

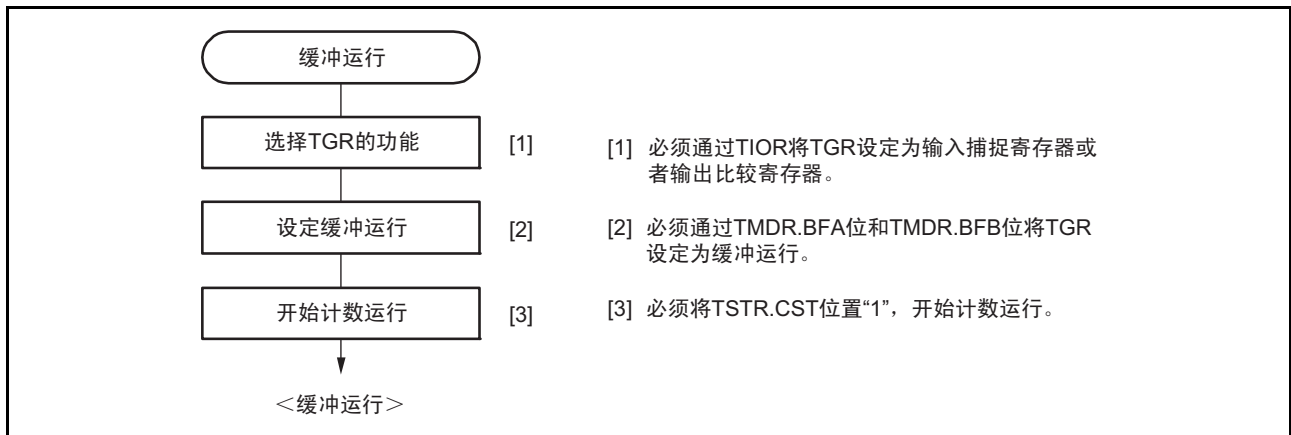


图 21.16 缓冲运行的设定步骤例子

(2) 缓冲运行的例子

(a) TGR 为输出比较寄存器的情况

将 MTU0 设定为 PWM 模式 1 并且将 TGRA 和 TGRC 设定为缓冲运行时的运行例子如图 21.17 所示。在此例子中，假设通过比较匹配 B 进行 TCNT 的清除，在比较匹配 A 时输出 High 电平，在比较匹配 B 时输出 Low 电平，将 TBTM 的 TTSA 位置“0”。

因为设定了缓冲运行，所以当发生比较匹配 A 时，就在输出发生变化的同时将缓冲寄存器 TGRC 的值传送到定时器的通用寄存器 TGRA。每当发生比较匹配 A 时，重复此操作。

有关 PWM 模式，请参照“21.3.5 PWM 模式”。

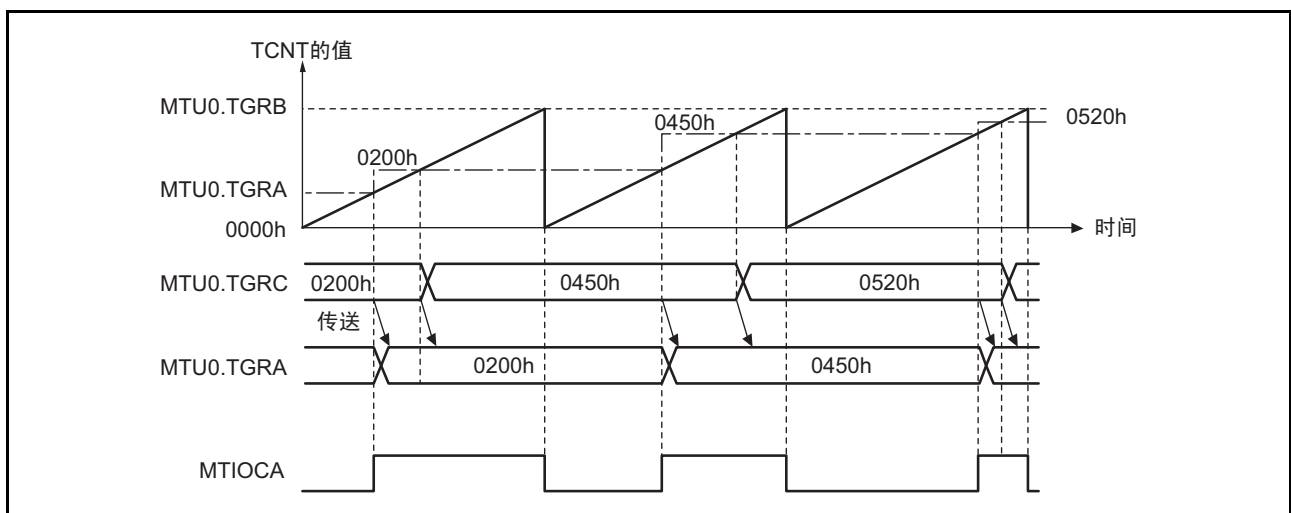


图 21.17 缓冲运行的例子 (1)

(b) TGR 为输入捕捉寄存器的情况

将 TGRA 设定为输入捕捉寄存器并且将 TGRA 和 TGRC 设定为缓冲运行时的运行例子如图 21.18 所示。

在 TGRA 的输入捕捉时清除 TCNT 计数器，选择上升沿 / 下降沿的双边沿作为 MTIOCA 引脚的输入捕捉的输入边沿。

因为设定了缓冲运行，所以在通过输入捕捉 A 将 TCNT 的值保存到 TGRA 的同时，将以前保存在 TGRA 的值传送到 TGRC。

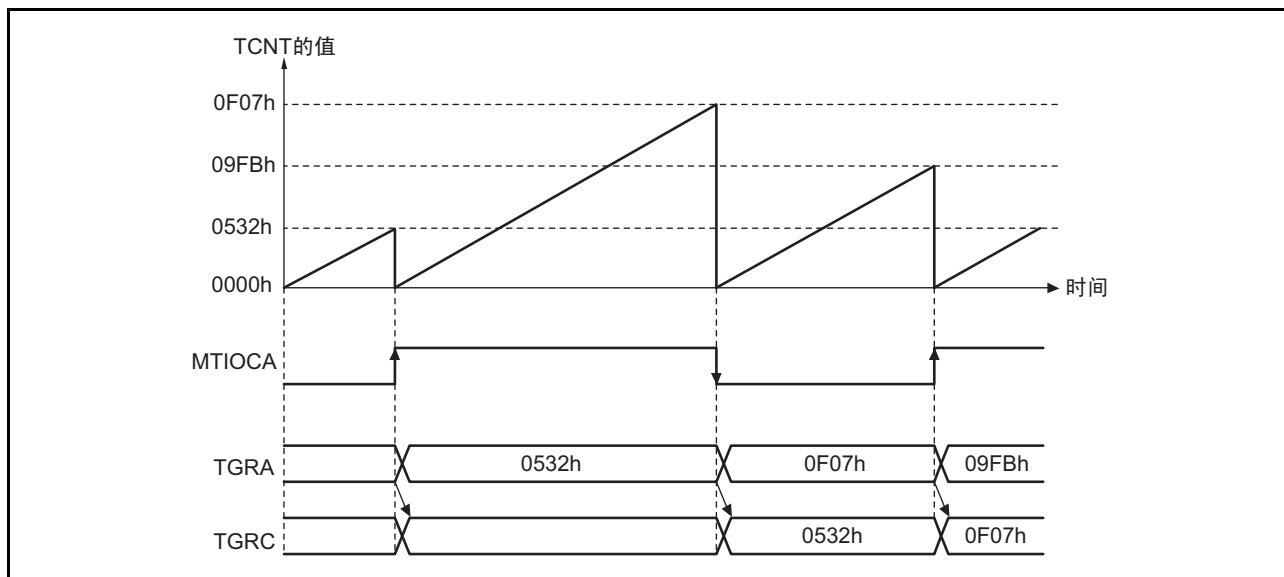


图 21.18 缓冲运行的例子 (2)

(3) 缓冲运行时的缓冲寄存器到定时器的通用寄存器的传送时序选择

能通过设定缓冲运行传送模式寄存器 (MTU0.TBTM、MTU3.TBTM 和 MTU4.TBTM)，选择 MTU0 为 PWM 模式 1 和 PWM 模式 2 时以及 MTU3 和 MTU4 为 PWM 模式 1 时的缓冲寄存器到定时器的通用寄存器的传送时序。能选择在发生比较匹配时 (初始值) 或者在清除 TCNT 时进行缓冲传送。在此，清除 TCNT 时是指当以下的任意一个条件成立时。

- TCNT 发生上溢时 (“FFFFh”→“0000h”)
- 在计数器运行过程中给 TCNT 写“0000h”时
- 通过 TCR 的 CCLR[2:0] 位设定的清除源使 TCNT 变为“0000h”时

注. 必须在 TCNT 停止计数的状态下设定 TBTM 寄存器。

将 MTU0 设定为 PWM 模式 1 并且将 MTU0.TGRA 和 MTU0.TGRC 设定为缓冲运行时的运行例子如图 21.19 所示。通过比较匹配 B 进行 MTU0.TCNT 的清除，在比较匹配 A 时输出 High 电平，在比较匹配 B 时输出 Low 电平，将 MTU0.TBTM 的 TTSA 位置“1”。

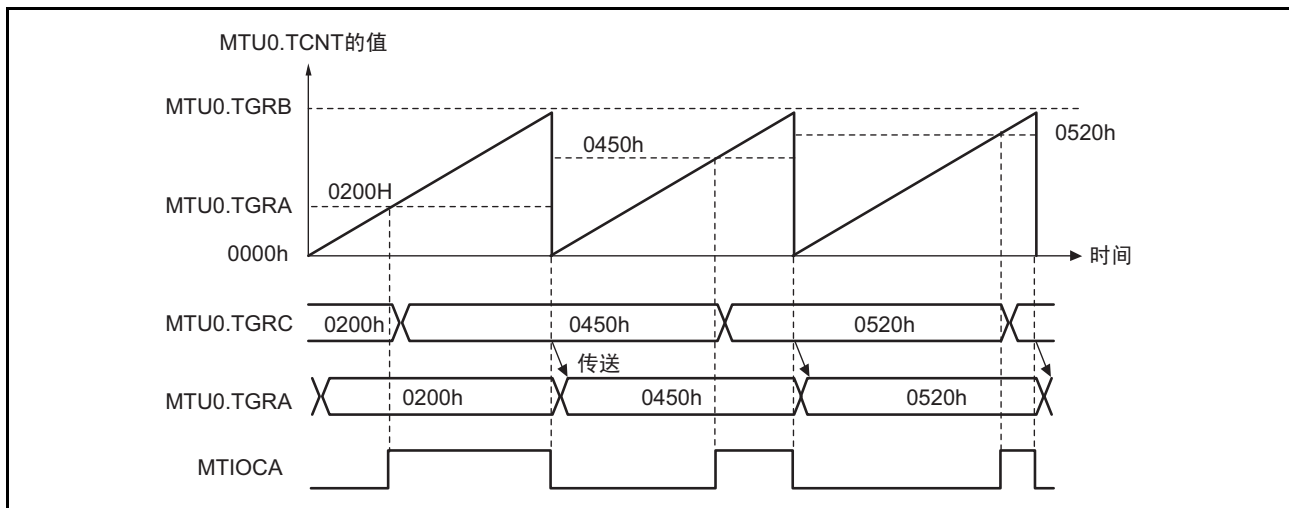


图 21.19 选择在清除 MTU0.TCNT 时进行 MTU0.TGRC 到 MTU0.TGRA 的缓冲传送的运行例子

21.3.4 级联运行

级联运行是将 2 个通道的 16 位计数器连接为 32 位计数器的功能。

通过 TCR 寄存器的 TPSC[2:0] 位将 MTU1 的计数器时钟设定为通过 (MTU2.TCNT 的) 上溢 / 下溢进行计数, 实现级联运行的功能。

只在低 16 位的 TCNT 为相位计数模式时发生下溢。

级联的组合如表 21.44 所示。

注. 如果将 MTU1 和 MTU2 设定为相位计数模式, 计数器时钟的设定就无效, 并且在相位计数模式中独立运行。

表 21.44 级联的组合

组合	高 16 位	低 16 位
MTU1 和 MTU2	MTU1.TCNT	MTU2.TCNT

在进行级联运行时, 如果 MTU1.TCNT 和 MTU2.TCNT 同时进行输入捕捉, 就能通过输入捕捉控制寄存器 (TICCR) 进行设定, 将输入引脚追加到输入捕捉条件。有关级联时的输入捕捉, 请参照“21.6.22 级联中的 MTU1.TCNT 和 MTU2.TCNT 的同时输入捕捉”。

TICCR 寄存器的设定值和输入捕捉的输入引脚的对应如表 21.45 所示。

表 21.45 TICCR 寄存器的设定值和输入捕捉的输入引脚的对应

对象输入捕捉	TICCR 寄存器的设定值	输入捕捉的输入引脚
MTU1.TCNT 到 MTU1.TGRA 的 输入捕捉	I2AE 位 = 0 (初始值)	MTIOC1A
	I2AE 位 = 1	MTIOC1A、MTIOC2A
MTU1.TCNT 到 MTU1.TGRB 的 输入捕捉	I2BE 位 = 0 (初始值)	MTIOC1B
	I2BE 位 = 1	MTIOC1B、MTIOC2B
MTU2.TCNT 到 MTU2.TGRA 的 输入捕捉	I1AE 位 = 0 (初始值)	MTIOC2A
	I1AE 位 = 1	MTIOC2A、MTIOC1A
MTU2.TCNT 到 MTU2.TGRB 的 输入捕捉	I1BE 位 = 0 (初始值)	MTIOC2B
	I1BE 位 = 1	MTIOC2B、MTIOC1B

(1) 级联运行的设定步骤例子

级联运行的设定步骤例子如图 21.20 所示。

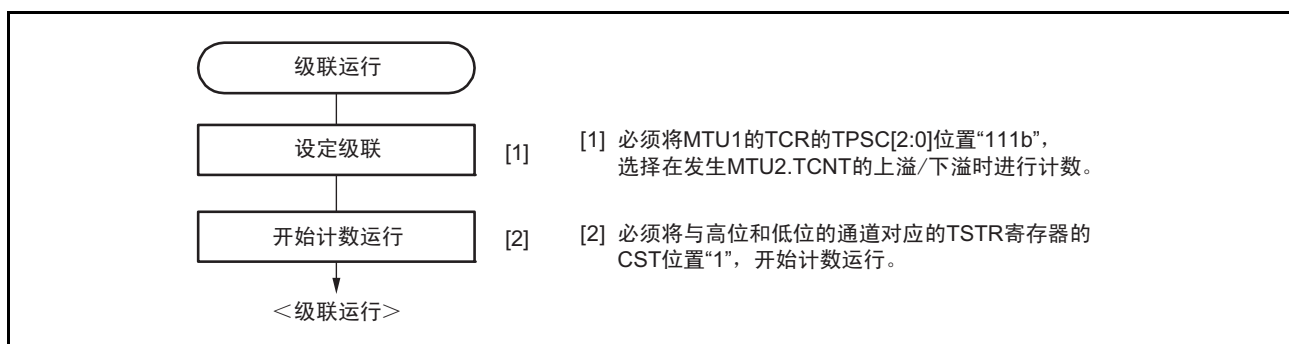


图 21.20 级联运行的设定步骤例子

(2) 级联运行的例子 (a)

MTU1.TCNT 通过 MTU2.TCNT 的上溢 / 下溢进行计数并且将 MTU2 设定为相位计数模式时的运行如图 21.21 所示。

MTU1.TCNT 通过 MTU2.TCNT 的上溢进行递增计数，通过 MTU2.TCNT 的下溢进行递减计数。

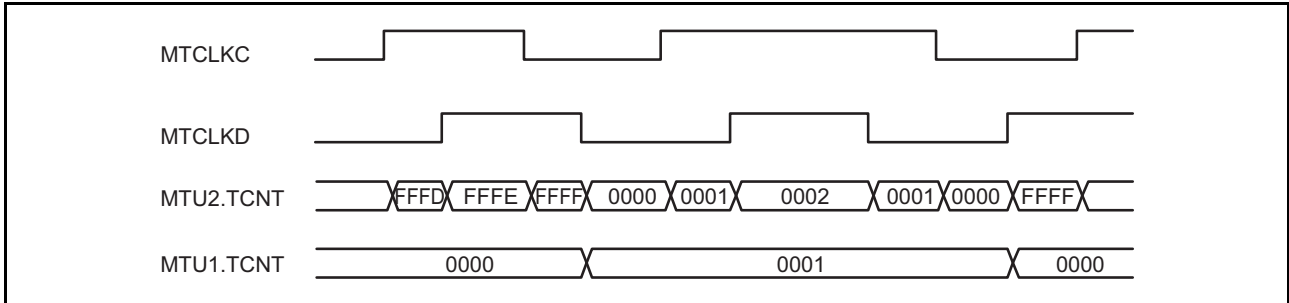


图 21.21 级联运行的例子 (a)

(3) 级联运行的例子 (b)

在将 MTU1.TCNT 和 MTU2.TCNT 进行级联，并且通过将 TICCR.I2AE 位置“1”使 MTIOC2A 引脚追加到 MTU1.TGRA 输入捕捉条件时的运行如图 21.22 所示。在此例子中，将 MTU1.TIOR 的 IOA[3:0] 设定为在 (MTIOC1A 的) 上升沿进行输入捕捉，将 MTU2.TIOR 的 IOA[3:0] 设定为在 (MTIOC2A 的) 上升沿进行输入捕捉。

此时，MTIOC1A 和 MTIOC2A 的上升沿被设定为 MTU1.TGRA 的输入捕捉条件，MTIOC2A 的上升沿被设定为 MTU2.TGRA 的输入捕捉条件。

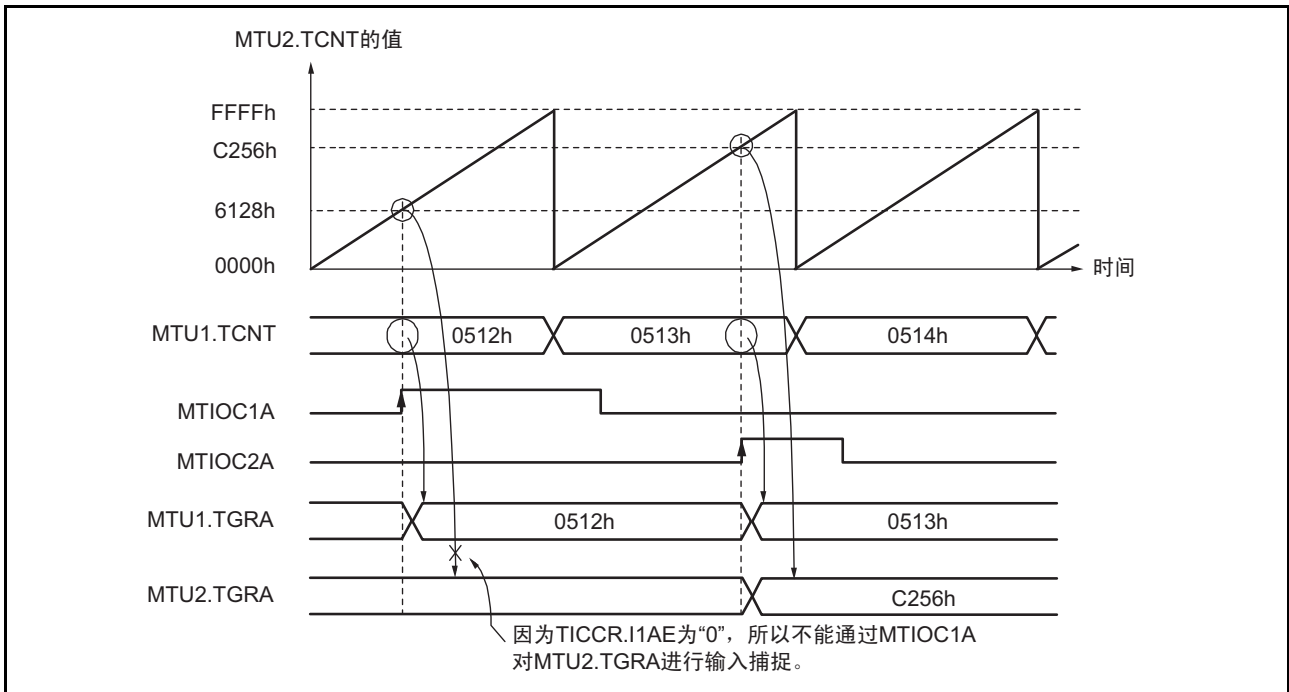


图 21.22 级联运行的例子 (b)

(4) 级联运行的例子 (c)

将 MTU1.TCNT 和 MTU2.TCNT 进行级联，并且通过将 TICCR.I2AE 位和 TICCR.I1AE 位置“1”使 MTIOC2A 引脚追加到 MTU1.TGRA 的输入捕捉条件以及使 MTIOC1A 引脚追加到 MTU2.TGRA 的输入捕捉条件时的运行如图 21.23 所示。在此例子中，将 MTU1.TIOR 和 MTU2.TIOR 的 IOA[3:0] 位都设定为在双边沿进行输入捕捉。此时，MTIOC1A 输入和 MTIOC2A 输入的 OR 为 MTU1.TGRA 和 MTU2.TGRA 的输入捕捉条件。

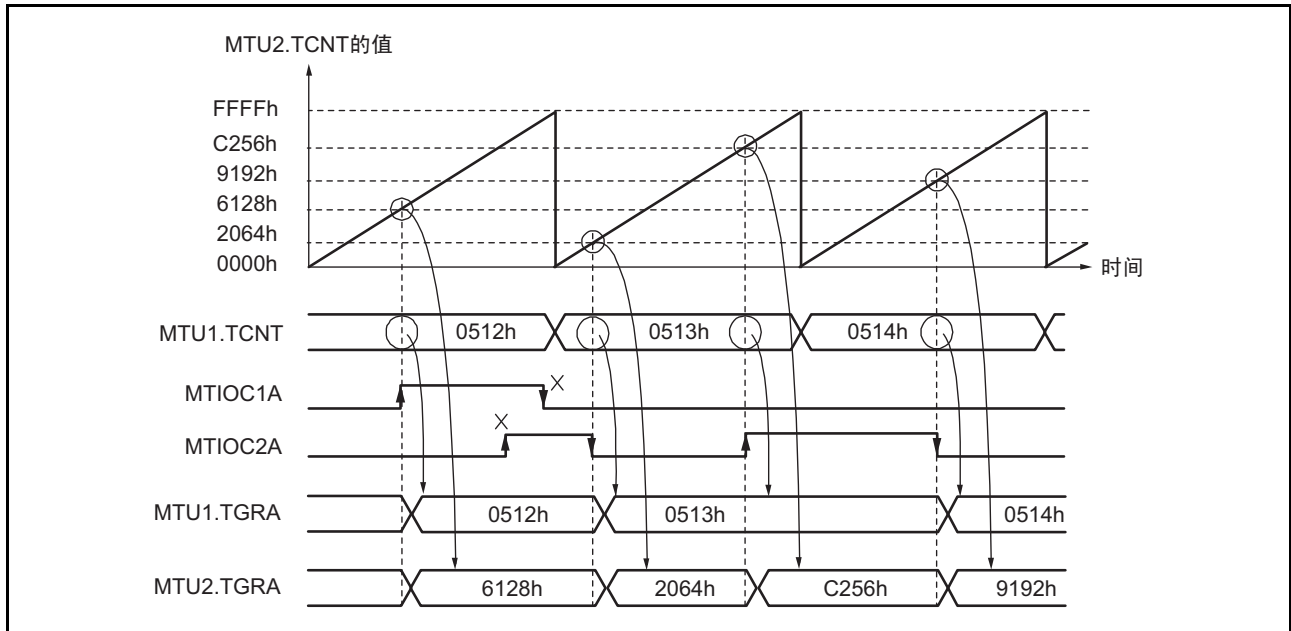


图 21.23 级联运行的例子 (c)

(5) 级联运行的例子 (d)

将 MTU1.TCNT 和 MTU2.TCNT 进行级联，并且通过将 TICCRR 寄存器的 I2AE 位置“1”使 MTIOC2A 引脚追加到 MTU1.TGRA 的输入捕捉条件时的运行如图 21.24 所示。在此例子中，将 MTU1.TIOR 的 IOA[3:0] 位设定为在发生 MTU0.TGRA 的比较匹配 / 输入捕捉时进行输入捕捉，将 MTU2.TIOR 的 IOA[3:0] 位设定为在 (MTIOC2A 的) 上升沿进行输入捕捉。

此时，因为将 MTU1.TIOR 设定为在发生 MTU0.TGRA 的比较匹配 / 输入捕捉时进行输入捕捉，所以即使将 TICCRR 寄存器的 I2AE 位置“1”，MTIOC2A 的边沿也不会成为 MTU1.TGRA 的输入捕捉条件。

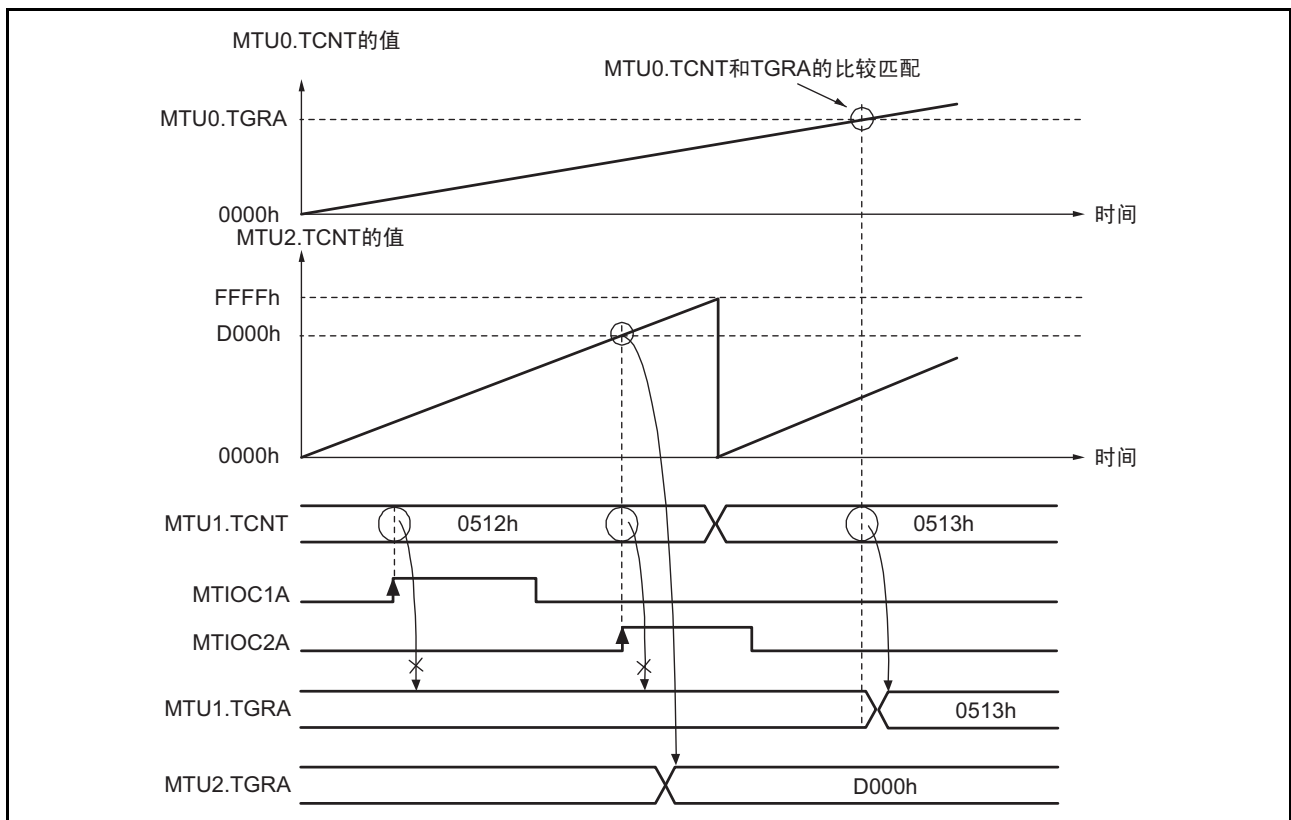


图 21.24 级联运行的例子 (d)

21.3.5 PWM 模式

PWM 模式是从输出引脚分别输出 PWM 波形的模式。各 TGR 比较匹配的输出电平可选择 Low 电平输出、High 电平输出或者交替输出。

能通过设定各 TGR，输出 0 ~ 100% 占空比的 PWM 波形。

能通过将 TGR 的比较匹配作为计数器的清除源，给该寄存器设定周期。能将全部通道分别设定为 PWM 模式，也能进行同步运行。

PWM 模式有以下 2 种模式。

(a) PWM 模式 1

TGRA 和 TGRB、TGRC 和 TGRD 配对使用，生成从 MTIOcNA 引脚和 MTIOcNC 引脚的 PWM 输出波形。通过比较匹配 A 和比较匹配 C 从 MTIOcNA 引脚和 MTIOcNC 引脚进行 TIOR 的 IOA[3:0] 位和 IOC[3:0] 位指定的输出，并且通过比较匹配 B 和比较匹配 D 从 MTIOcNA 引脚和 MTIOcNC 引脚进行 TIOR 的 IOB[3:0] 位和 IOD[3:0] 位指定的输出，初始输出值为 TGRA 和 TGRC 的设定值。当配对使用的 TGR 的设定值相同时，即使发生比较匹配，输出值也不变。

在 PWM 模式 1 中，能进行最多 8 相的 PWM 波形输出。

(b) PWM 模式 2

将 1 个 TGR 用作周期寄存器而将其他 TGR 用作占空比寄存器，生成 PWM 输出波形。通过比较匹配进行 TIOR 指定的输出，并且通过同步寄存器的比较匹配进行计数器清除，各引脚的输出值为 TIOR 设定的初始值。如果周期寄存器和占空比寄存器的设定值相同，即使发生比较匹配，输出值也不变。

在 PWM 模式 2 中，能通过和同步运行的并用，进行最多 8 相的 PWM 波形输出。

PWM 的输出引脚和寄存器的对应如表 21.46 所示。

表 21.46 各 PWM 输出的寄存器和输出引脚

通道	寄存器	输出引脚	
		PWM 模式 1	PWM 模式 2
MTU0	MTU0.TGRA	MTIOC0A	MTIOC0A
	MTU0.TGRB		MTIOC0B
	MTU0.TGRC	MTIOC0C	MTIOC0C
	MTU0.TGRD		MTIOC0D
MTU1	MTU1.TGRA	MTIOC1A	MTIOC1A
	MTU1.TGRB		MTIOC1B
MTU2	MTU2.TGRA	MTIOC2A	MTIOC2A
	MTU2.TGRB		MTIOC2B
MTU3	MTU3.TGRA	MTIOC3A	不能设定。
	MTU3.TGRB		
	MTU3.TGRC	MTIOC3C	
	MTU3.TGRD		
MTU4	MTU4.TGRA	MTIOC4A	
	MTU4.TGRB		
	MTU4.TGRC	MTIOC4C	
	MTU4.TGRD		

注. 在 PWM 模式 2 中，不能对已设定周期的 TGR 进行 PWM 波形输出。

(1) PWM 模式的设定步骤例子

PWM 模式的设定步骤例子如图 21.25 所示。

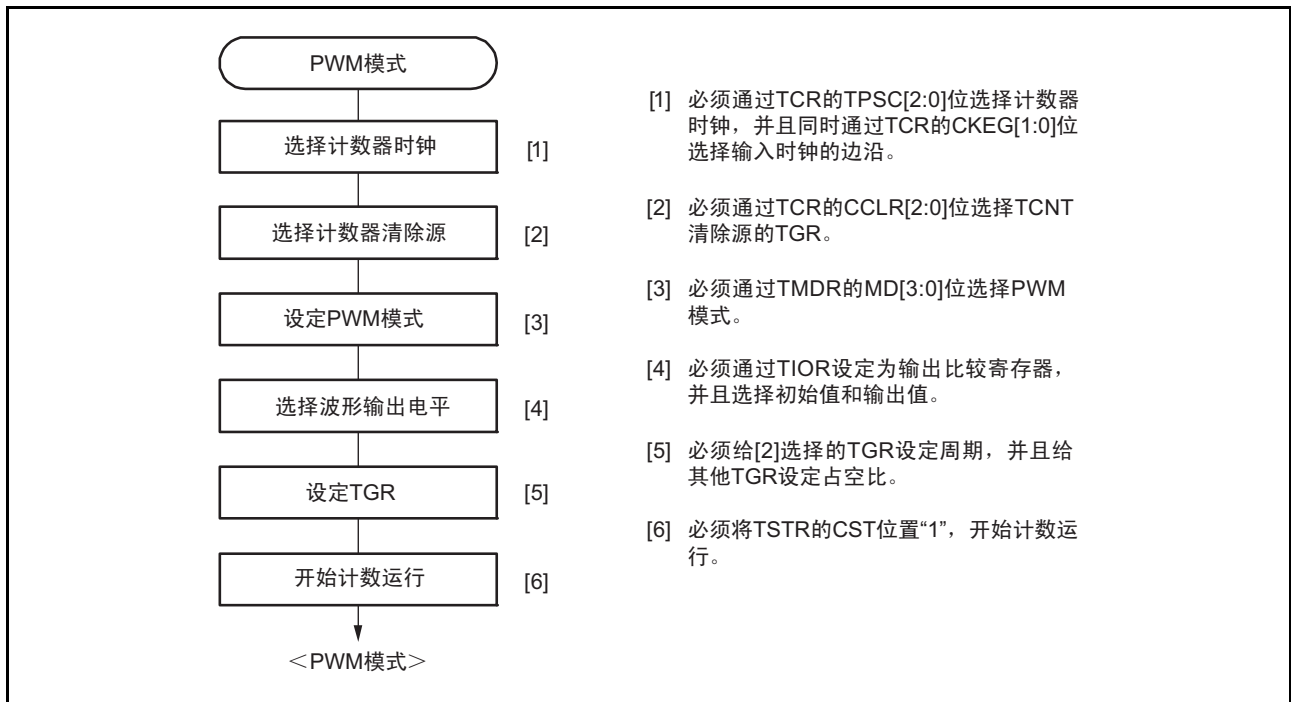


图 21.25 PWM 模式的设定步骤例子

(2) PWM 模式的运行例子

PWM 模式 1 的运行例子如图 21.26 所示。

在此图中，假设将 TGRA 的比较匹配设定为 TCNT 的清除源，将 TGRA 的初始输出值和输出值设定为“0”，将 TGRB 的输出值设定为“1”。

此时，TGRA 的设定值为周期，TGRB 的设定值为占空比。

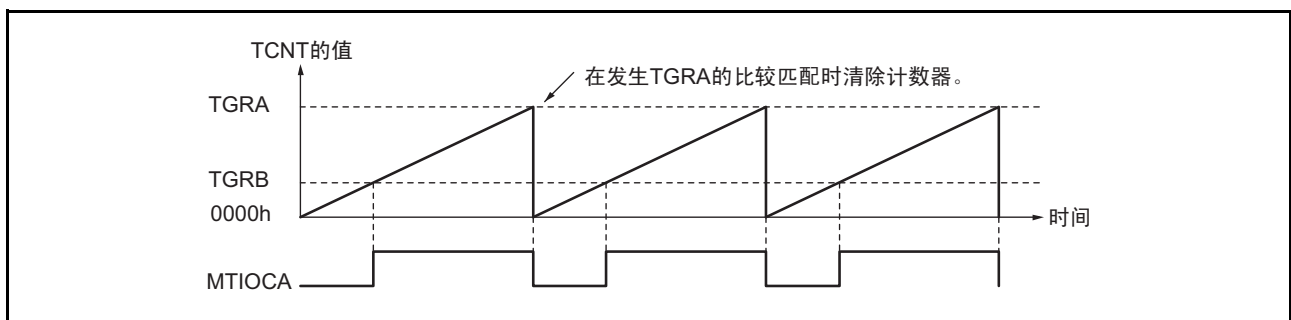


图 21.26 PWM 模式的运行例子

PWM 模式 2 的运行例子如图 21.27 所示。

在此图中，假设使 MTU0 和 MTU1 同步运行，将 MTU1.TGRB 的比较匹配设定为 TCNT 的清除源，将其他 TGR (MTU0.TGRA ~ MTU0.TGRD 和 MTU1.TGRA) 的初始输出值设定为“0”，将输出值设定为“1”，输出 5 相 PWM 波形。

此时，TGR1B 的设定值为周期，其他的 TGR 设定值为占空比。

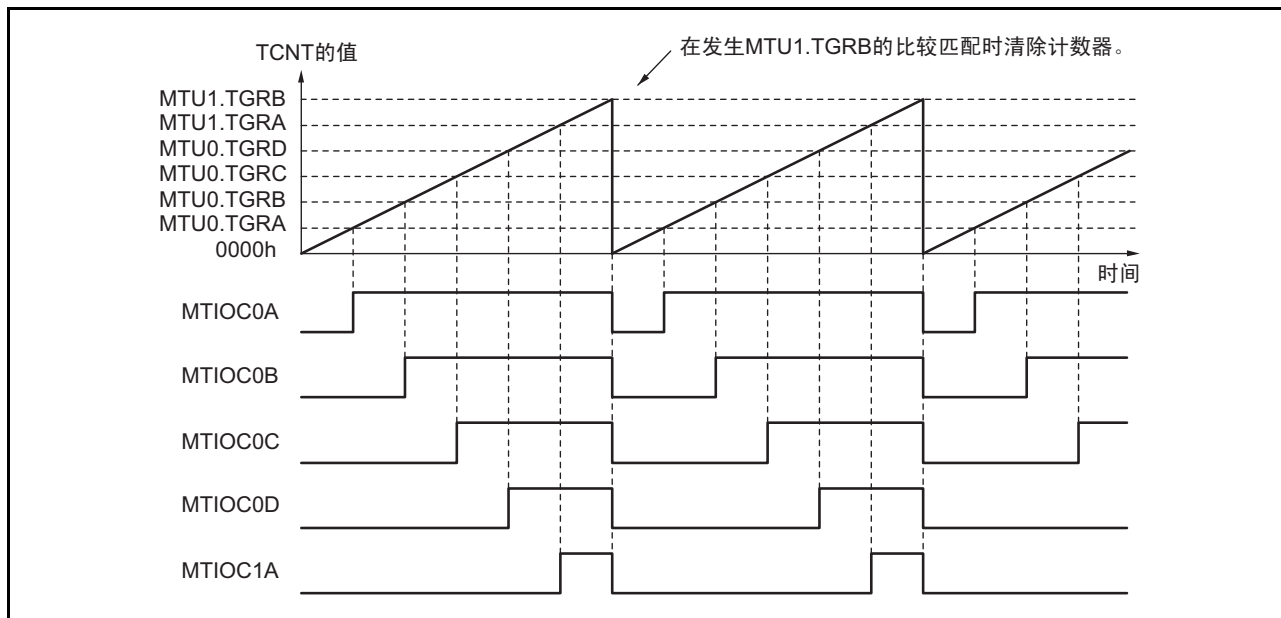


图 21.27 PWM 模式的运行例子

在 PWM 模式中，占空比为 0% 和 100% 的 PWM 波形输出例子如图 21.28 所示。

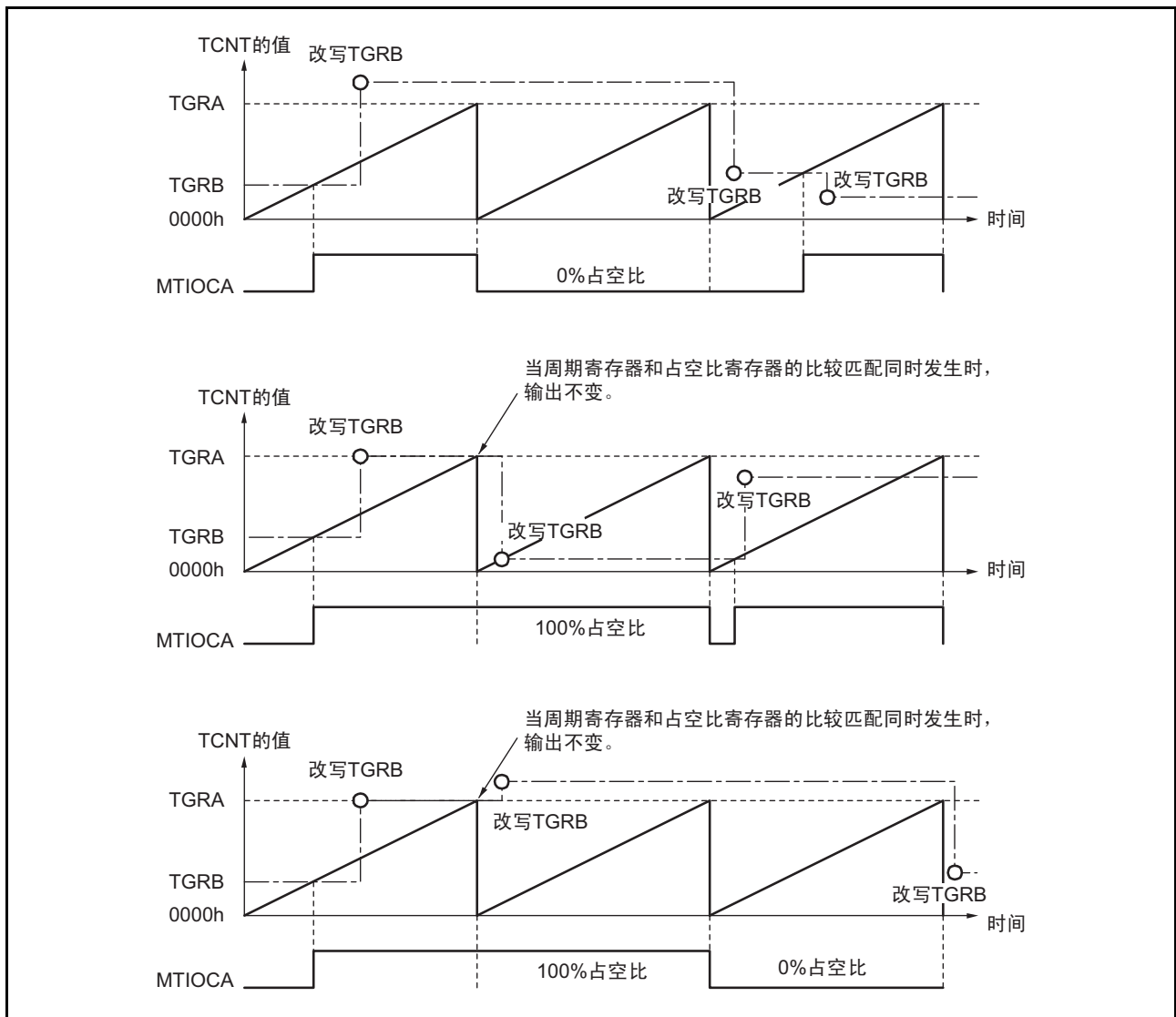


图 21.28 PWM 模式的运行例子

21.3.6 相位计数模式

相位计数模式通过设定 MTU1 和 MTU2，检测 2 个外部时钟输入的相位差，并且 TCNT 进行递增/递减计数。

如果设定为相位计数模式，就选择外部时钟作为计数器输入时钟，并且 TCNT 作为递增/递减计数器运行，与 TCR 寄存器的 TPSC[2:0] 位和 CKEG[1:0] 位的设定无关。但是，因为 TCR 寄存器的 CCLR[1:0] 位、TIOR、TIER、TGR 寄存器的功能有效，所以能使用输入捕捉/比较匹配功能和中断功能。

相位计数模式能用作 2 相编码器脉冲的输入。

如果在 TCNT 进行递增计数时发生上溢并且对应的 TIER 寄存器的 TCIEV 位为“1”，就产生 TCIV 中断。如果在 TCNT 进行递减计数时发生下溢并且对应的 TIER 寄存器的 TCIEU 位为“1”，就产生 TCIU 中断。

TSR 寄存器的 TCFD 位是计数方向标志。能通过读 TCFD 标志，确认 TCNT 是在进行递增计数还是在进行递减计数。

外部时钟引脚和通道的对应如表 21.47 所示。

表 21.47 相位计数模式的时钟输入引脚

通道	外部时钟引脚	
	A 相	B 相
MTU1	MTCLKA	MTCLKB
MTU2	MTCLKC	MTCLKD

(1) 相位计数模式的设定步骤例子

相位计数模式的设定步骤例子如图 21.29 所示。

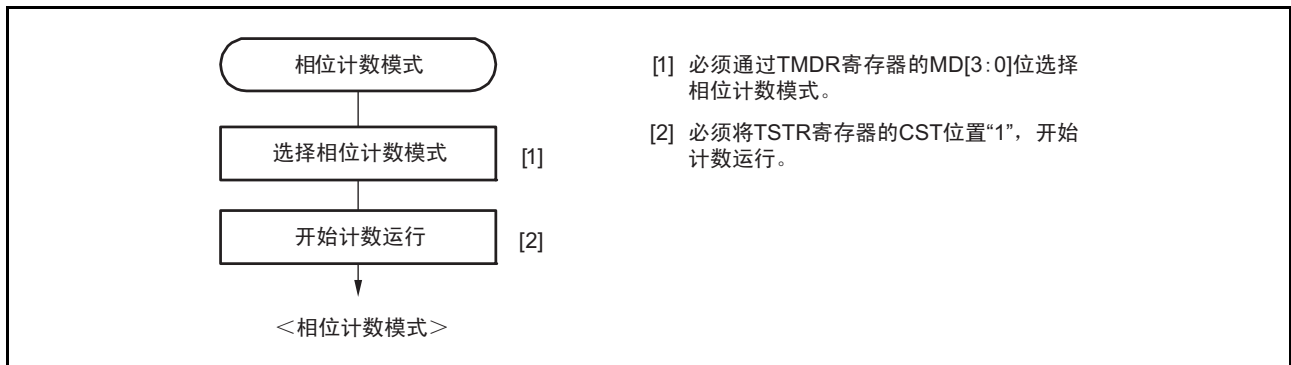


图 21.29 相位计数模式的设定步骤例子

(2) 相位计数模式的运行例子

在相位计数模式中，TCNT 通过 2 个外部时钟的相位差进行递增计数或者递减计数。计数条件有 4 种模式。

(a) 相位计数模式 1

相位计数模式 1 的运行例子如图 21.30 所示，TCNT 的递增 / 递减计数条件如表 21.48 所示。

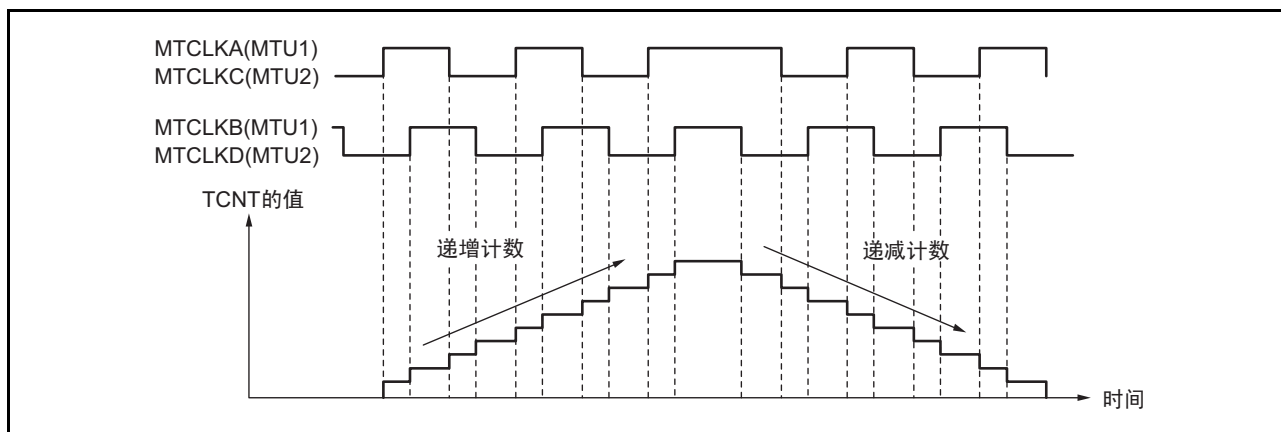


图 21.30 相位计数模式 1 的运行例子

表 21.48 相位计数模式 1 的递增 / 递减计数条件

MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	运行内容
High 电平	上升沿	递增计数
Low 电平	下降沿	
上升沿	Low 电平	
下降沿	High 电平	
High 电平	下降沿	递减计数
Low 电平	上升沿	
上升沿	High 电平	
下降沿	Low 电平	

上升沿
下降沿

(b) 相位计数模式 2

相位计数模式 2 的运行例子如图 21.31 所示，TCNT 的递增 / 递减计数条件如表 21.49 所示。

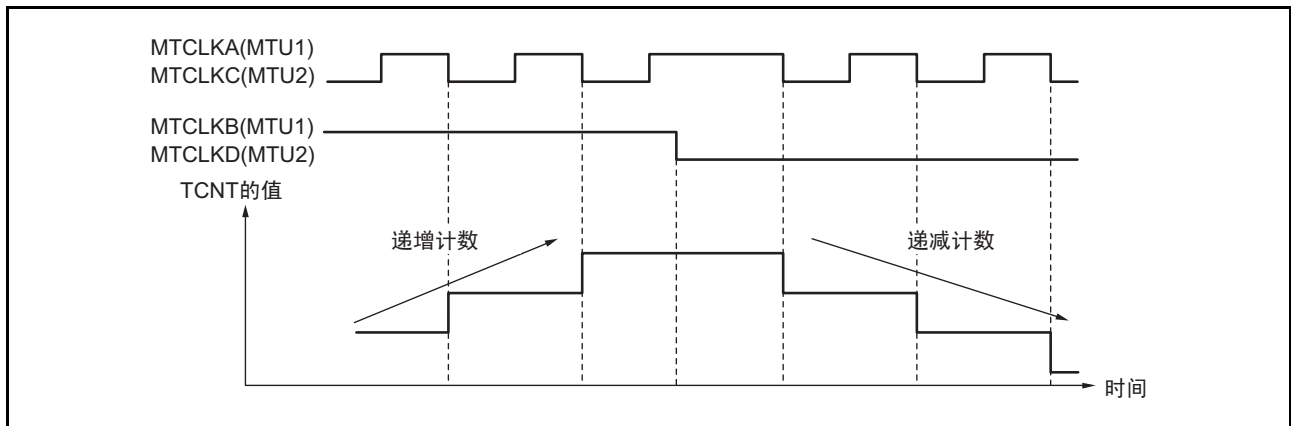


图 21.31 相位计数模式 2 的运行例子

表 21.49 相位计数模式 2 的递增 / 递减计数条件

MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	运行内容
High 电平		不计数 (Don't care)
Low 电平		不计数 (Don't care)
	Low 电平	不计数 (Don't care)
	High 电平	递增计数
High 电平		不计数 (Don't care)
Low 电平		不计数 (Don't care)
	High 电平	不计数 (Don't care)
	Low 电平	递减计数

: 上升沿

: 下降沿

(c) 相位计数模式 3

相位计数模式 3 的运行例子如图 21.32 所示，TCNT 的递增 / 递减计数条件如表 21.50 所示。

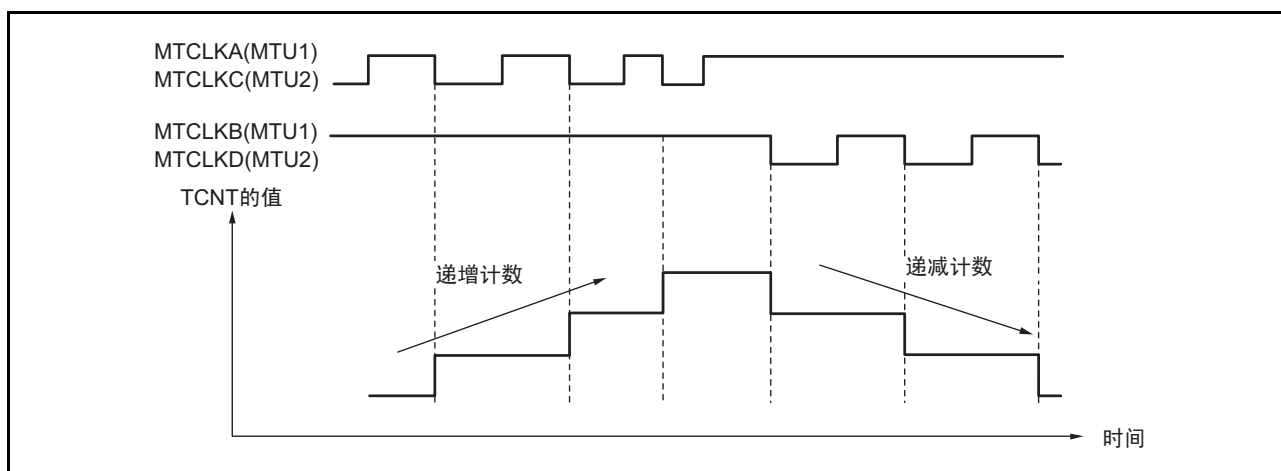


图 21.32 相位计数模式 3 的运行例子

表 21.50 相位计数模式 2 的递增 / 递减计数条件

MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	运行内容
High 电平	↑	不计数 (Don't care)
Low 电平	↓	不计数 (Don't care)
↑	Low 电平	不计数 (Don't care)
↓	High 电平	递增计数
High 电平	↓	递减计数
Low 电平	↑	不计数 (Don't care)
↑	High 电平	不计数 (Don't care)
↓	Low 电平	不计数 (Don't care)

↑ : 上升沿
↓ : 下降沿

(d) 相位计数模式 4

相位计数模式 4 的运行例子如图 21.33 所示，TCNT 的递增 / 递减计数条件如表 21.51 所示。

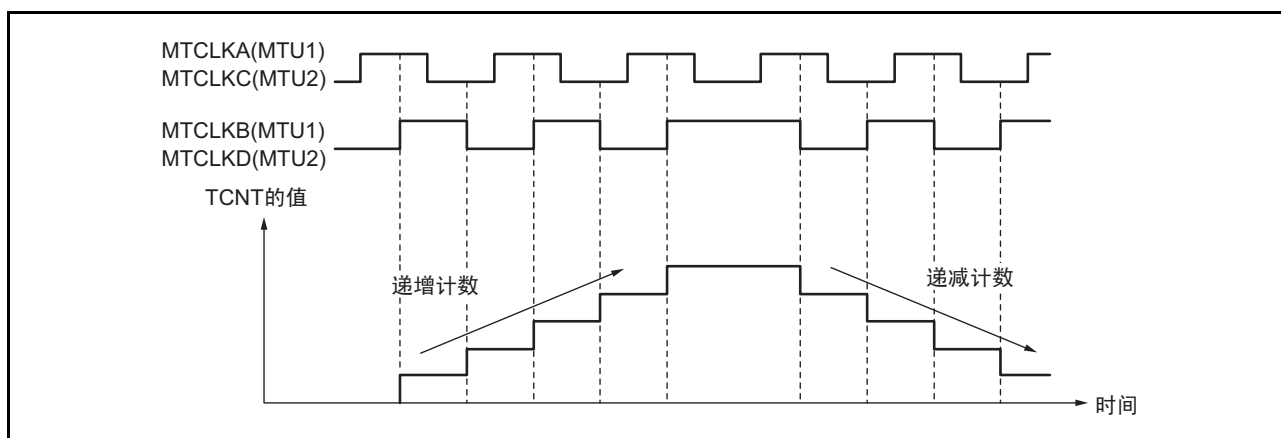


图 21.33 相位计数模式 4 的运行例子

表 21.51 相位计数模式 2 的递增 / 递减计数条件

MTCLKA (MTU1) MTCLKC (MTU2)	MTCLKB (MTU1) MTCLKD (MTU2)	运行内容
High 电平	↑	递增计数
Low 电平	↓	
↑	Low 电平	不计数 (Don't care)
↓	High 电平	
High 电平	↓	递减计数
Low 电平	↑	
↑	High 电平	不计数 (Don't care)
↓	Low 电平	

↑ : 上升沿
↓ : 下降沿

(3) 相位计数模式的应用例子

将 MTU1 设定为相位计数模式，并且在与 MTU0 组合输入伺服马达 2 相编码器脉冲后检测位置或者速度的例子如图 21.34 所示。

将 MTU1 设定为相位计数模式 1，给 MTCLKA 和 MTCLKB 输入编码器脉冲的 A 相和 B 相。

MTU0 通过 MTU0.TGRC 的比较匹配进行 TCNT 计数器清除，MTU0.TGRA 和 MTU0.TGRC 用于比较匹配功能，设定速度控制周期和位置控制周期。MTU0.TGRB 用于输入捕捉功能，使 MTU0.TGRB 和 MTU0.TGRD 进行缓冲运行。将 MTU1 的计数器输入时钟作为 MTU0.TGRB 的输入捕捉源，检测 2 相编码器的 4 倍频脉冲的脉宽。

将 MTU1 的 MTU1.TGRA 和 MTU1.TGRB 设定为输入捕捉功能，选择 MTU0 的 MTU0.TGRA 和 MTU0.TGRC 的比较匹配作为输入捕捉源，保存各控制周期时的递增 / 递减计数器的值。

能用此方法检测正确的位置和速度。

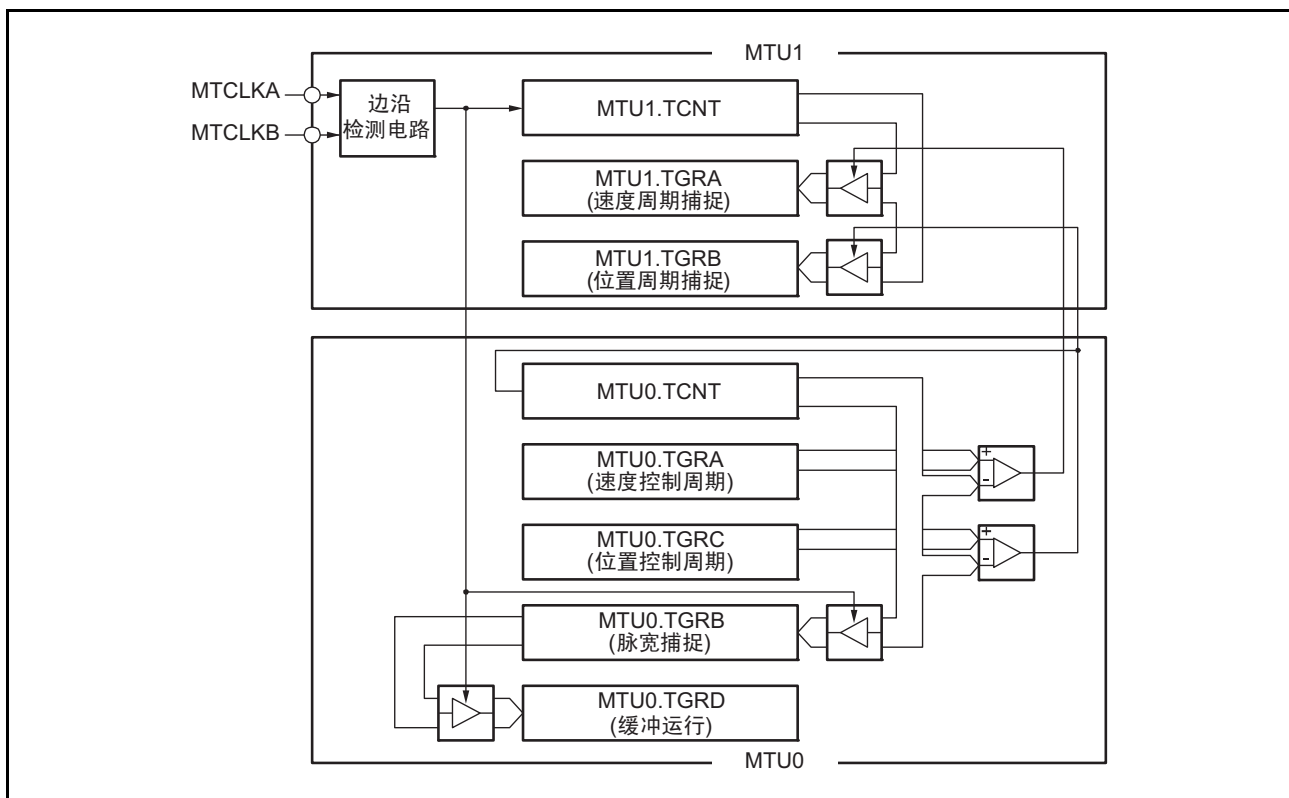


图 21.34 相位计数模式的应用例子

21.3.7 复位同步 PWM 模式

在复位同步 PWM 模式中，通过组合 MTU3 和 MTU4，将一方的波形变化点作为共同关系的 PWM 波形（正相和反相）进行 3 相输出。

如果设定为复位同步 PWM 模式，MTIOC3B、MTIOC3D、MTIOC4A、MTIOC4C、MTIOC4B 和 MTIOC4D 引脚就为 PWM 输出引脚，定时器的计数器 3 (MTU3.TCNT) 用作递增计数器。

PWM 输出引脚和寄存器的设定分别如表 21.52 和表 21.53 所示。

表 21.52 复位同步 PWM 模式中的输出引脚

通道	输出引脚	说明
MTU3	MTIOC3B	PWM 的输出引脚 1
	MTIOC3D	PWM 的输出引脚 1' (PWM 输出 1 的反相波形)
MTU4	MTIOC4A	PWM 的输出引脚 2
	MTIOC4C	PWM 的输出引脚 2' (PWM 输出 2 的反相波形)
	MTIOC4B	PWM 的输出引脚 3
	MTIOC4D	PWM 的输出引脚 3' (PWM 输出 3 的反相波形)

表 21.53 复位同步 PWM 模式中的寄存器设定

寄存器	设定内容
MTU3.TCNT	初始设定“0000h”。
MTU4.TCNT	初始设定“0000h”。
MTU3.TGRA	设定 MTU3.TCNT 的计数周期。
MTU3.TGRB	设定从 MTIOC3B 引脚和 MTIOC3D 引脚输出的 PWM 波形变化点。
MTU4.TGRA	设定从 MTIOC4A 引脚和 MTIOC4C 引脚输出的 PWM 波形变化点。
MTU4.TGRB	设定从 MTIOC4B 引脚和 MTIOC4D 引脚输出的 PWM 波形变化点。

(1) 复位同步 PWM 模式的设定步骤例子

复位同步 PWM 模式的设定步骤例子如图 21.35 所示。

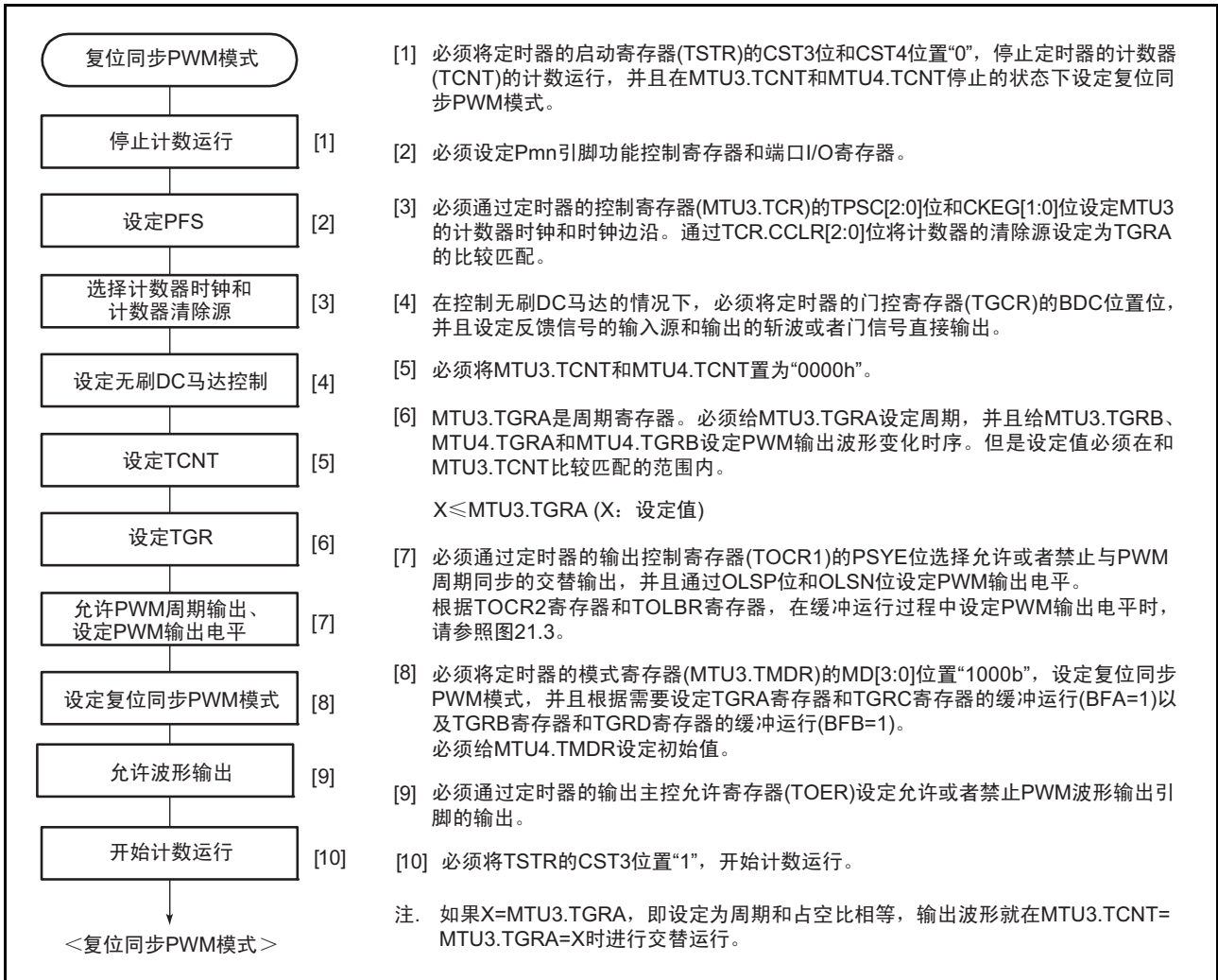


图 21.35 复位同步 PWM 模式的设定步骤例子

(2) 复位同步 PWM 模式的运行例子

复位同步 PWM 模式的运行例子如图 21.36 所示。

在复位同步 PWM 模式中，MTU3.TCNT 和 MTU4.TCNT 作为递增计数器运行。如果 MTU3.TCNT 和 MTU3.TGRA 比较匹配，就清除计数器，并且从“0000h”重新开始递增计数。每当各 MTU3.TGRB、MTU4.TGRA 和 MTU4.TGRB 发生比较匹配并且进行计数器清除时，PWM 输出引脚进行交替输出。

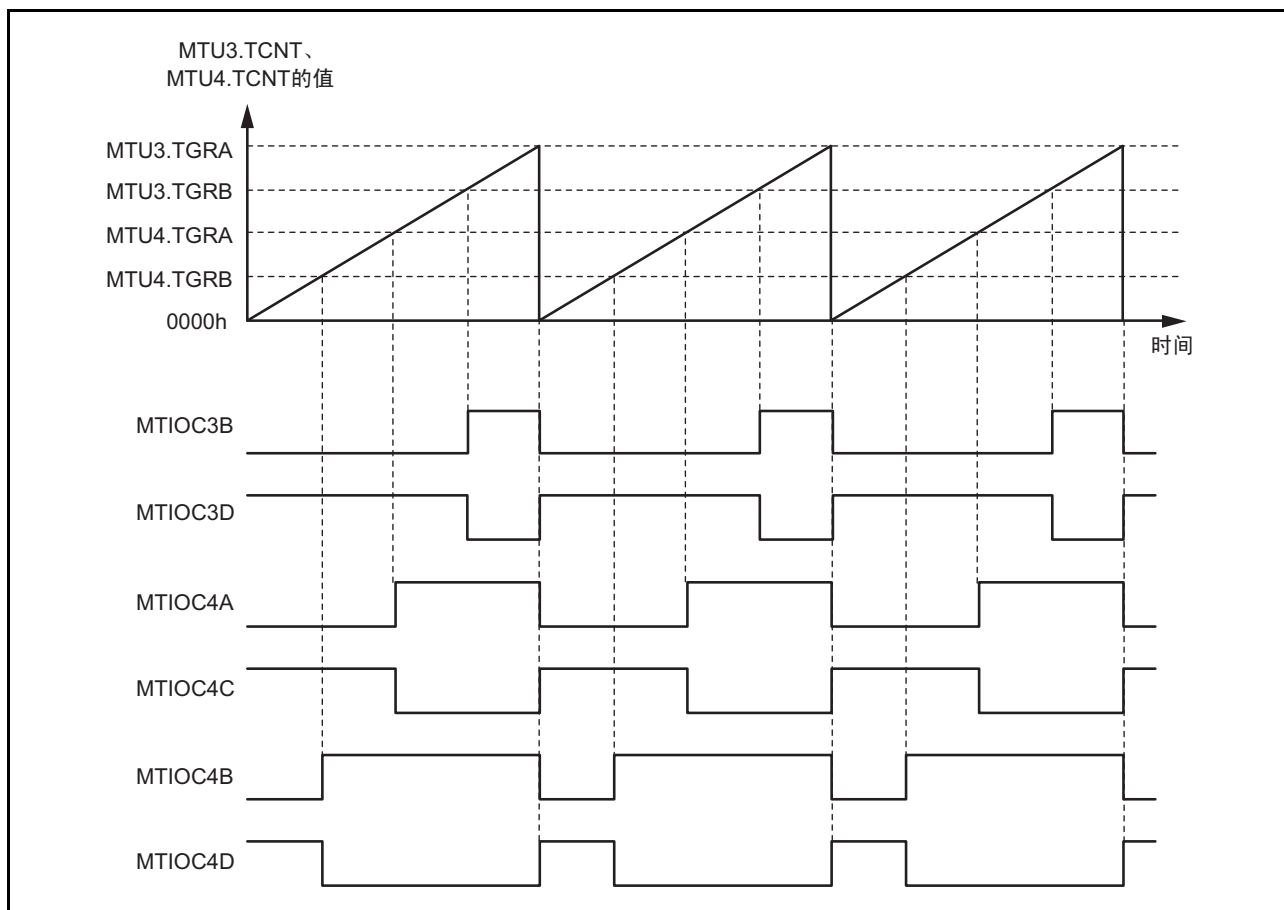


图 21.36 复位同步 PWM 模式的运行例子 (设定为 TOCR 的 OLSN=1 并且 OLSP=1 的情况)

21.3.8 互补 PWM 模式

在互补 PWM 模式中，通过组合 MTU3 和 MTU4，将正相和反相为非重叠关系的 PWM 波形进行 3 相输出，也能设定为没有重叠时间。

如果设定为互补 PWM 模式，MTIOC3B、MTIOC3D、MTIOC4A、MTIOC4B、MTIOC4C 和 MTIOC4D 引脚就为 PWM 输出引脚，也能将 MTIOC3A 引脚设定为与 PWM 周期同步的交替输出。

MTU3.TCNT 和 MTU4.TCNT 用作递增 / 递减计数器。

使用的 PWM 输出引脚和寄存器设定分别如表 21.54 和表 21.55 所示。

作为端口功能，支持通过外部信号直接关闭 PWM 输出的功能。

表 21.54 互补 PWM 模式的输出引脚

通道	输出引脚	说明
MTU3	MTIOC3A	与 PWM 周期同步的交替输出（或者输入 / 输出端口）
	MTIOC3B	PWM 输出引脚 1
	MTIOC3C	输入 / 输出端口（注 1）
	MTIOC3D	PWM 输出引脚 1'（与 PWM 输出 1 有非重叠关系的反相波形，也能设定为没有非重叠时间）
MTU4	MTIOC4A	PWM 输出引脚 2
	MTIOC4C	PWM 输出引脚 2'（与 PWM 输出 2 有非重叠关系的反相波形，也能设定为没有非重叠时间）
	MTIOC4B	PWM 输出引脚 3
	MTIOC4D	PWM 输出引脚 3'（与 PWM 输出 3 有非重叠关系的反相波形，也能设定为没有非重叠时间）

注 1. 不能在互补 PWM 模式中将 MTIOC3C 引脚设定为定时器的输入 / 输出引脚。

表 21.55 互补 PWM 模式的寄存器设定

通道	计数器 / 寄存器	说明	由 CPU 进行读写
MTU3	MTU3.TCNT	从死区时间寄存器的设定值开始递增计数。	能通过设定 TRWER 寄存器(注1)进行屏蔽。
	MTU3.TGRA	设定 MTU3.TCNT 的上限值 (1/2 个载波周期 + 死区时间)。	能通过设定 TRWER 寄存器(注1)进行屏蔽。
	MTU3.TGRB	PWM 输出 1 的比较寄存器	能通过设定 TRWER 寄存器(注1)进行屏蔽。
	MTU3.TGRC	MTU3.TGRA 的缓冲寄存器	随时能读写。
	MTU3.TGRD	PWM 输出 1/MTU3.TGRB 的缓冲寄存器	随时能读写。
MTU4	MTU4.TCNT	初始设定“0000h”并且开始递增计数。	能通过设定 TRWER 寄存器(注1)进行屏蔽。
	MTU4.TGRA	PWM 输出 2 的比较寄存器	能通过设定 TRWER 寄存器(注1)进行屏蔽。
	MTU4.TGRB	PWM 输出 3 的比较寄存器	能通过设定 TRWER 寄存器(注1)进行屏蔽。
	MTU4.TGRC	PWM 输出 2/MTU4.TGRA 的缓冲寄存器	随时能读写。
	MTU4.TGRD	PWM 输出 3/MTU4.TGRB 的缓冲寄存器	随时能读写。
定时器的死区时间数据寄存器 (TDDR)	设定 MTU4.TCNT 和 MTU3.TCNT 的偏移值 (死区时间的值)。	能通过设定 TRWER 寄存器(注1)进行屏蔽。	
定时器的周期数据寄存器 (TCDR)	设定 MTU4.TCNT 的上限值 (1/2 个载波周期)。	能通过设定 TRWER 寄存器(注1)进行屏蔽。	
定时器的周期缓冲寄存器 (TCBR)	TCDR 寄存器的缓冲寄存器	随时能读写。	
副计数器 (TCNTS)	用于生成死区时间的副计数器	只能读。	
暂存器 1 (TEMP1)	PWM 输出 1/MTU3.TGRB 的暂存器	不能读写。	
暂存器 2 (TEMP2)	PWM 输出 2/MTU4.TGRA 的暂存器	不能读写。	
暂存器 3 (TEMP3)	PWM 输出 3/MTU4.TGRB 的暂存器	不能读写。	

注 1. 能通过设定 TRWER 寄存器 (定时器的读写允许寄存器), 允许或者禁止存取。

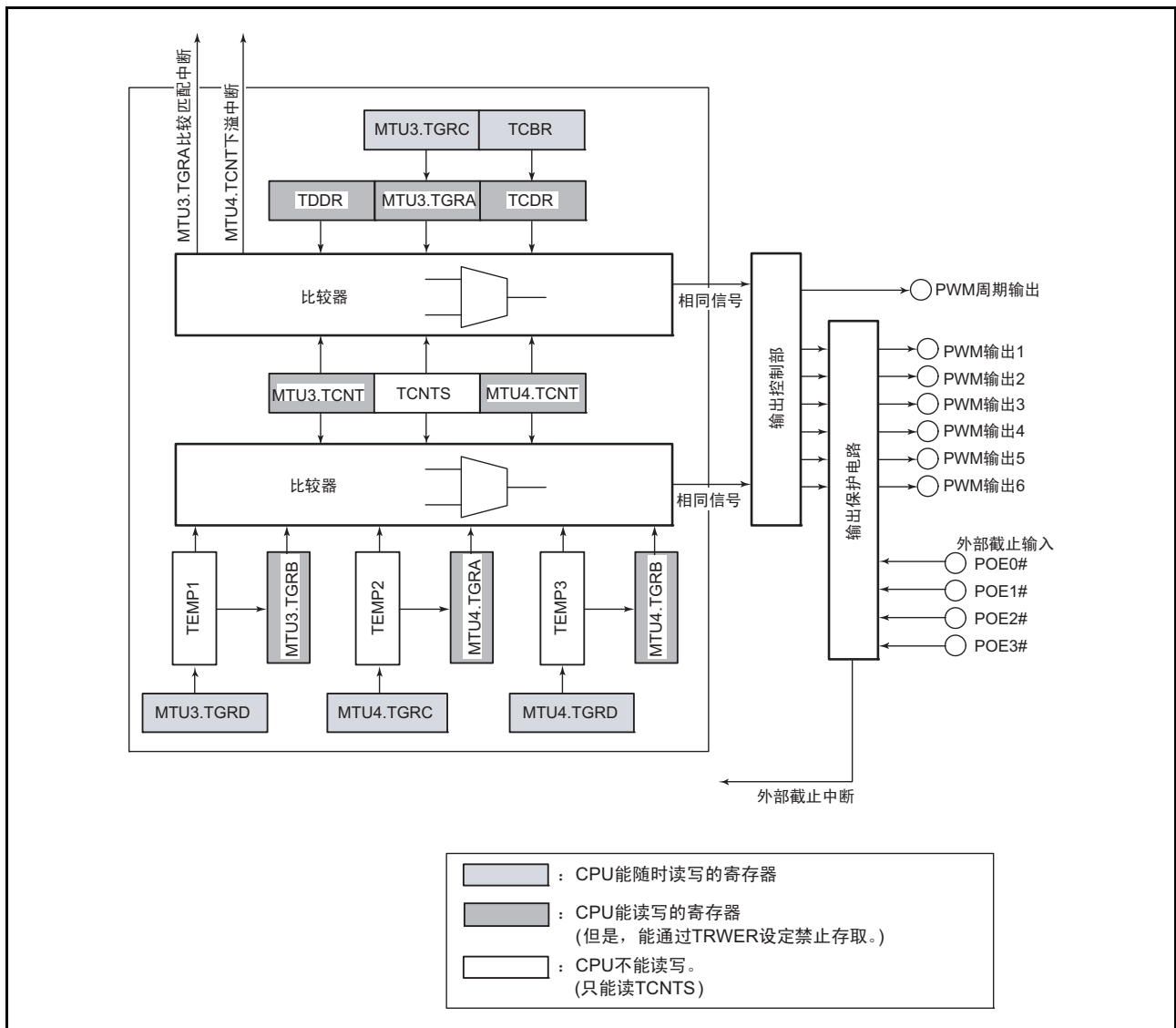


图 21.37 互补 PWM 模式中的 MTU3 和 MTU4 的框图

(1) 互补 PWM 模式的设定步骤例子

互补 PWM 模式的设定步骤例子如图 21.38 所示。



图 21.38 互补 PWM 模式的设定步骤例子

(2) 互补 PWM 模式的运行概要

在互补 PWM 模式中能进行 6 相的 PWM 输出。互补 PWM 模式的计数器运行和运行例子分别如图 21.39 和图 21.40 所示。

(a) 计数器的运行

在互补 PWM 模式中，MTU3.TCNT、MTU4.TCNT 和 TCNTS 这 3 个计数器进行递增计数或者递减计数。

在设定为互补 PWM 模式并且 TSTR 寄存器的 CST 位为“0”时，将 TDDR 寄存器的设定值作为初始值，自动设定给 MTU3.TCNT。

如果将 CST 位置“1”，MTU3.TCNT 就进行递增计数，直到 MTU3.TGRA 的设定值为止。一旦 MTU3.TCNT 的值和 MTU3.TGRA 的值相同，MTU3.TCNT 就改为递减计数。以后，一旦 MTU3.TCNT 的值和 TDDR 寄存器的值相同，MTU3.TCNT 就改为递增计数，重复此运行。

将 MTU4.TCNT 的初始值置为“0000h”。

如果将 CST 位置“1”，就与 MTU3.TCNT 同步运行并且进行递增计数。一旦 MTU4.TCNT 的值和 TCDR 寄存器的值相同，MTU4.TCNT 就改为递减计数。此后，一旦计数到“0000h”，MTU4.TCNT 就改为递增计数，重复此运行。

TCNTS 是只读计数器，不需要设定初始值。

当 MTU3 和 MTU4 的 TCNT 进行递增计数或者递减计数时，一旦 MTU3.TCNT 的值和 TCDR 寄存器的值相同，就开始递减计数。一旦 TCNTS 的值和 TCDR 寄存器的值相同，TCNTS 就改为递增计数。一旦计数器的值和 MTU3.TGRA 的值相同，TCNTS 就变为“0000h”。

在 MTU3.TCNT 和 MTU4.TCNT 进行递减计数时，一旦 MTU4.TCNT 的值和 TDDR 寄存器的值相同，就开始递增计数。一旦 TCNTS 的值和 TDDR 寄存器的值相同，TCNTS 就改为递减计数。一旦计数到“0000h”，TCNTS 就被设定为 MTU3.TGRA 的值。

TCNTS 只在计数运行期间与已设定 PWM 占空比的比较寄存器和暂存器进行比较。

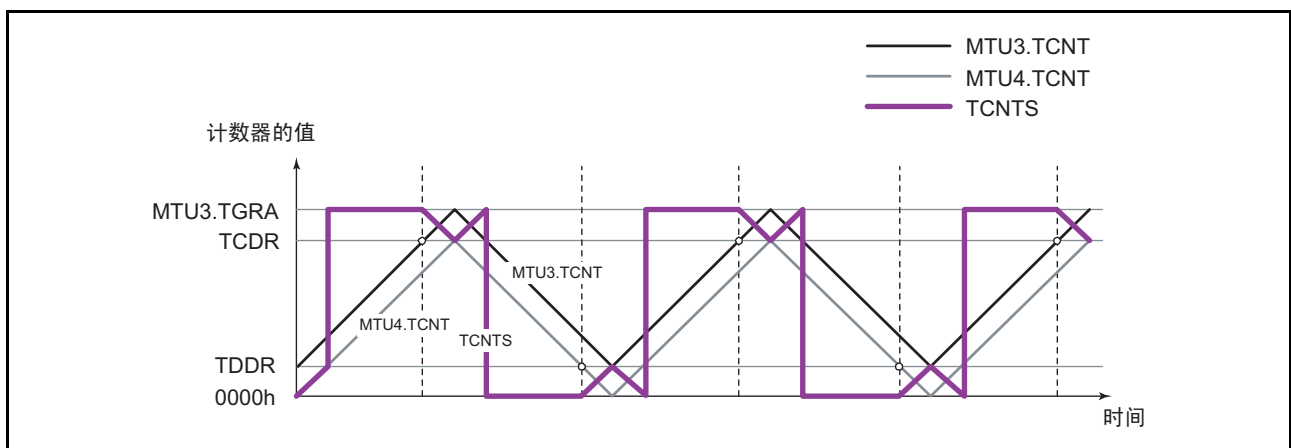


图 21.39 互补 PWM 模式的计数器运行

(b) 寄存器的运行

在互补 PWM 模式中，使用比较寄存器、缓冲寄存器和暂存器等 9 个寄存器。互补 PWM 模式的运行例子如图 21.40 所示。

为了进行 PWM 输出，MTU3.TGRB、MTU4.TGRA、MTU4.TGRB 寄存器随时和计数器进行比较。如果这些寄存器的值和计数器的值相同，就输出定时器的输出控制寄存器 (TOCR) 的 OLSN 位和 OLSP 位的设定值。

这些比较寄存器的缓冲寄存器为 MTU3.TGRD、MTU4.TGRC 和 MTU4.TGRD。在缓冲寄存器和比较寄存器之间有暂存器，但是 CPU 不能存取暂存器。

要更改比较寄存器的数据时，必须给对应的缓冲寄存器写要更改的数据。随时能读写缓冲寄存器。

在 Ta 区间总是将写到缓冲寄存器的数据传送到暂存器，而在 Tb 区间不传送到暂存器。在 Tb 区间结束后，将在此区间写到缓冲寄存器的数据传送到暂存器。

如果 Tb 区间结束的 TCNTS 在进行递增计数时计数值和 MTU3.TGRA 的值相同，或者在递减计数时计数值为“0000h”，就将传送到暂存器的值传送到比较寄存器。能通过定时器的模式寄存器 (TMDR) 的 MD[3:0] 位选择暂存器到比较寄存器的传送时序。选择在波谷更改模式的例子如图 21.40 所示。

在不向暂存器传送数据的 Tb (图 21.40 为 Tb1) 区间，暂存器具有和比较寄存器相同的功能，和计数器进行比较。在此区间，1 相输出有 2 个比较匹配寄存器，比较寄存器保存变更前数据，暂存器保存要变更的新数据。在此区间，MTU3.TCNT、MTU4.TCNT 和 TCNTS 共 3 个计数器和比较寄存器、暂存器 (2 个寄存器) 进行比较，控制 PWM 输出。

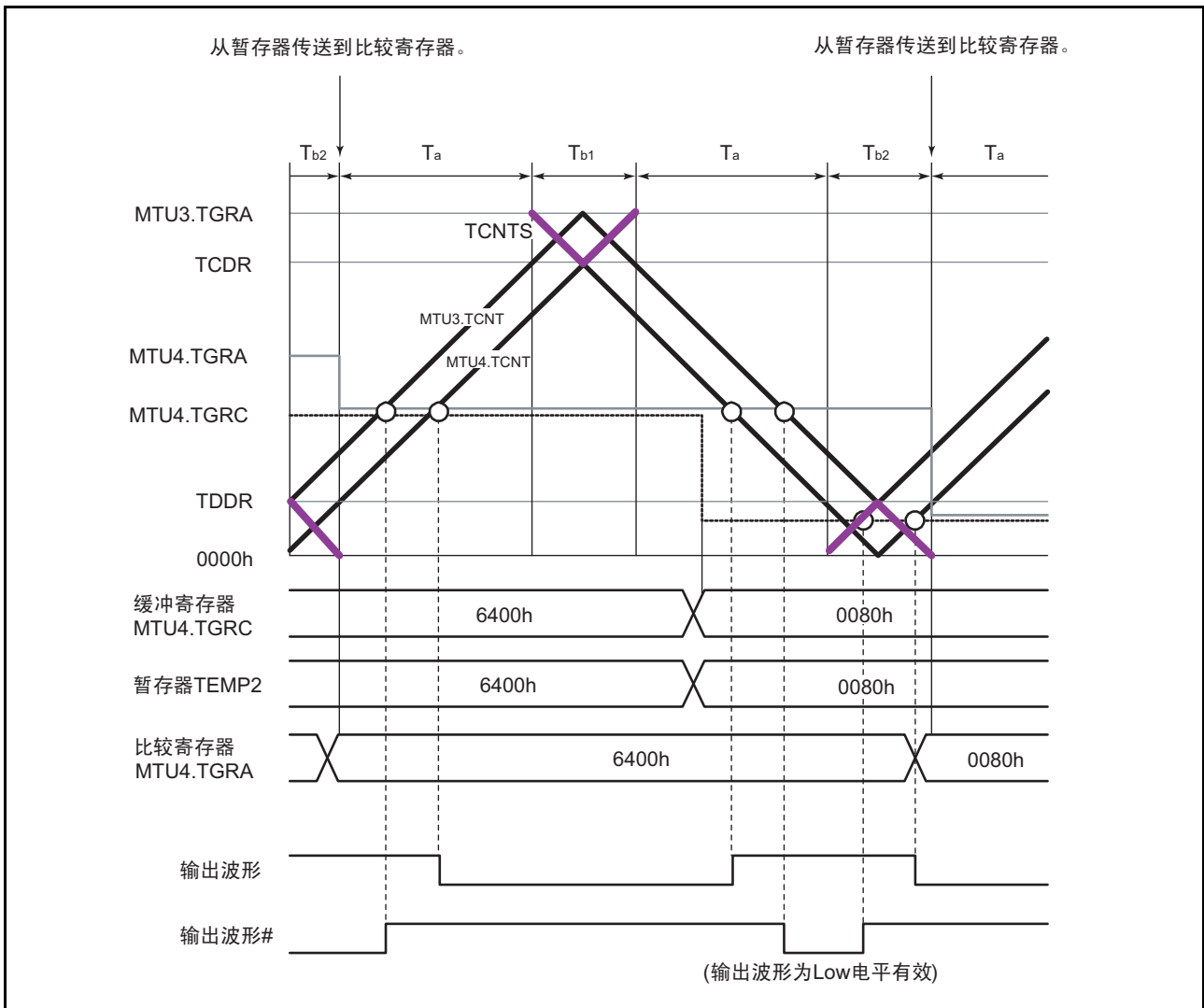


图 21.40 互补 PWM 模式的运行例子

(c) 初始设定

在互补 PWM 模式中，有 6 个需要初始设定的寄存器，还有 1 个用于设定是否生成死区时间的寄存器（只在不生成死区时间的情况下进行设定）。

在通过定时器的模式寄存器（TMDR）的 MD[3:0] 位设定为互补 PWM 模式前，必须设定以下寄存器的初始值。

MTU3.TGRC 用作 MTU3.TGRA 的缓冲寄存器，设定 1/2 个 PWM 载波周期 + 死区时间 Td。定时器的周期缓冲寄存器（TCBR）用作定时器的周期数据寄存器（TCDR）的缓冲寄存器，设定 1/2 个 PWM 载波周期，并且给定时器的死区时间数据寄存器（TDDR）设定死区时间 Td。

如果不生成死区时间，就将定时器的死区时间允许寄存器（TDER）的 TDER 位置“0”，给 MTU3.TGRC 和 MTU3.TGRA 设定 1/2 个 PWM 载波周期 +1，并且将 TDDR 寄存器置“1”。

给 3 个缓冲寄存器 MTU3.TGRD、MTU4.TGRC 和 MTU4.TGRD 分别设定 PWM 占空比的初始值。

在设定为互补 PWM 模式的同时，将 TDDR 寄存器除外的 5 个缓冲寄存器的设定值分别传送到对应的比较寄存器。

必须在设定互补 PWM 模式前将 MTU4.TCNT 设定为“0000h”。

表 21.56 需要初始设定的寄存器和计数器

寄存器 / 计数器	设定值
MTU3.TGRC	1/2 个 PWM 载波周期 + 死区时间 Td (如果通过 TDER 设定为不生成死区时间，就为 1/2 个 PWM 载波周期 +1)
TDDR	死区时间 Td (如果通过 TDER 设定为不生成死区时间，就为“1”)
TCBR	1/2 个 PWM 载波周期
MTU3.TGRD、 MTU4.TGRC、 MTU4.TGRD	各相 PWM 占空比的初始值
MTU4.TCNT	0000h

注. MTU3.TGRC 的设定值必须为 TCBR 寄存器设定的 1/2 个 PWM 载波周期的值和 TDDR 寄存器设定的死区时间 Td 值的和。如果通过 TDER 寄存器设定为不生成死区时间，就为 1/2 个 PWM 载波周期 +1。

(d) PWM 输出电平的设定

在互补 PWM 模式中，通过定时器的输出控制寄存器 1（TOCR1）的 OLSN 位和 OLSP 位或者定时器的输出控制寄存器 2（TOCR2）的 OLS1P ~ OLS3P 位和 OLS1N ~ OLS3N 位，设定 PWM 脉冲的输出电平。

能按 6 相输出中的 3 相正相和 3 相反相设定输出电平。

必须在解除互补 PWM 模式的状态下设定或者更改输出电平。

(e) 死区时间的设定

在互补 PWM 模式中，输出正相和反相为非重叠关系的 PWM 脉冲，此非重叠时间称为死区时间。

将非重叠时间设定到定时器的死区时间数据寄存器（TDDR）。TDDR 寄存器的设定值为 MTU3.TCNT 的计数器开始值，生成 MTU3.TCNT 和 MTU4.TCNT 的非重叠关系。必须在解除互补 PWM 模式的状态下更改 TDDR 寄存器的内容。

(f) 不生成死区时间的设定

通过将定时器的死区时间允许寄存器 (TDER) 的 TDER 位置“0”，设定为不生成死区时间。只在 TDER 位为“1”的状态下读 TDER 位后给 TDER 位写“0”时，能将 TDER 位设定为“0”。

给 MTU3.TGRA 和 MTU3.TGRC 设定 1/2 个 PWM 载波周期 +1，并且将定时器的死区时间数据寄存器 (TDDR) 置“1”。

如果设定为不生成死区时间，就能输出无死区时间的 PWM 波形。不生成死区时间的运行例子如图 21.41 所示。

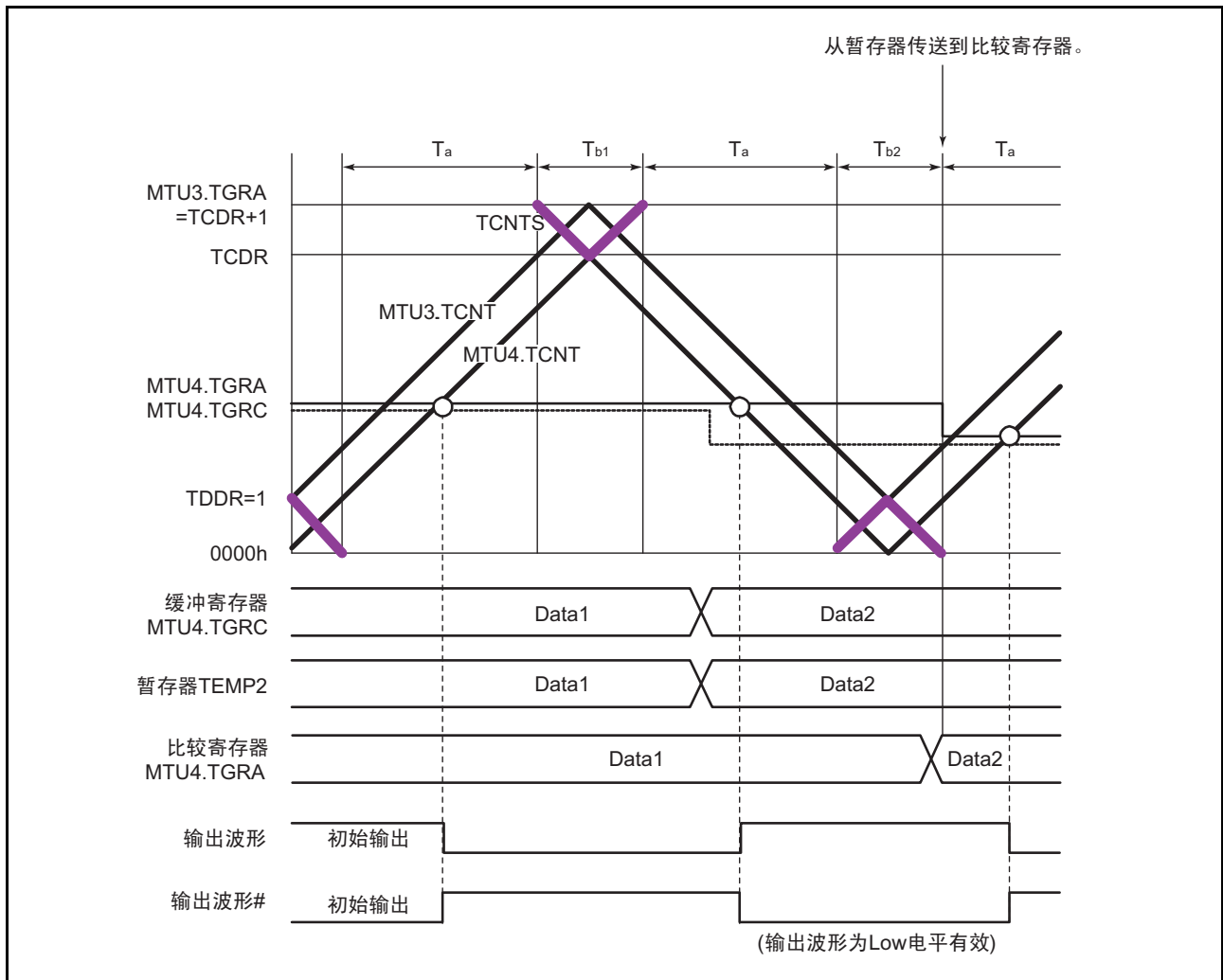


图 21.41 不生成死区时间的运行例子

(g) PWM 周期的设定

在互补 PWM 模式中，将 PWM 脉冲周期设定到 MTU3.TGRA (设定 MTU3.TCNT 的上限值) 和 MTU3.TCDR (设定 MTU4.TCNT 的上限值) 的 2 个寄存器。必须将这 2 个寄存器设定为如下的关系。

生成死区时间: $MTU3.TGRA$ 的设定值 = $TCDR$ 的设定值 + $TDDR$ 的设定值

不生成死区时间: $MTU3.TGRA$ 的设定值 = $TCDR$ 的设定值 + 1

必须通过给缓冲寄存器的 MTU3.TGRC 或者 MTU3.TCBR 设定值，设定 MTU3.TGRA 寄存器和 MTU3.TCDR 寄存器。在定时器的模式寄存器 (TMDR) 的 MD[3:0] 位选择的传送时序，将 MTU3.TGRC 寄存器和 MTU3.TCBR 寄存器的设定值同时传送到 MTU3.TGRA 寄存器和 MTU3.TCDR 寄存器。

如果在波峰更新数据，就从下一个周期反映更改的 PWM 周期；如果在波谷更新数据，就从该周期反映被更改的 PWM 周期。在波峰更改 PWM 周期时的运行例子如图 21.42 所示。

有关各缓冲寄存器数据的更新方法，请参照下面“(h) 寄存器数据的更新”。

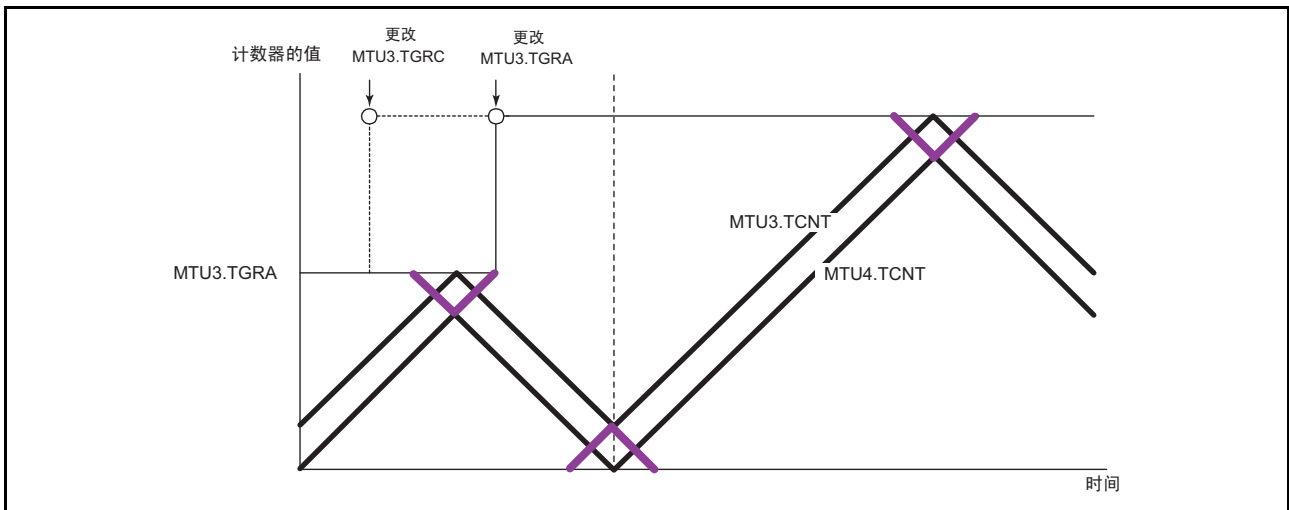


图 21.42 PWM 周期的更改例子

(h) 寄存器数据的更新

在互补 PWM 模式中更新比较寄存器的数据时，使用缓冲寄存器。更新的数据随时能写到缓冲寄存器。另外，能在有缓冲寄存器的运行过程中更改的寄存器为 5 个用于 PWM 占空比和载波周期的寄存器。

在这些寄存器和缓冲寄存器之间有各自的暂存器。如果在副计数器 TCNTS 不进行计数的期间更新寄存器的数据，也会改写暂存器的值。在 TCNTS 计数过程中，不进行缓冲寄存器到暂存器的传送，而在 TCNTS 停止计数后传送缓冲寄存器的写入值。

在定时器的模式寄存器 (TMDR) 的 MD[3:0] 位设定的数据更新时序，将暂存器的值传送到比较寄存器。互补 PWM 模式的数据更新例子如图 21.43 所示，此图是在计数器的波峰和波谷更新数据的模式例子。

在改写缓冲寄存器的数据时，最后必须写 MTU4.TGRD。在写 MTU4.TGRD 后，5 个寄存器的数据同时从缓冲寄存器传送到暂存器。

即使不全部更新 5 个寄存器或者不更新 MTU4.TGRD 的数据，也必须在写要更新的寄存器数据后写 MTU4.TGRD。此时，写到 MTU4.TGRD 的数据必须和写之前的数据相同。

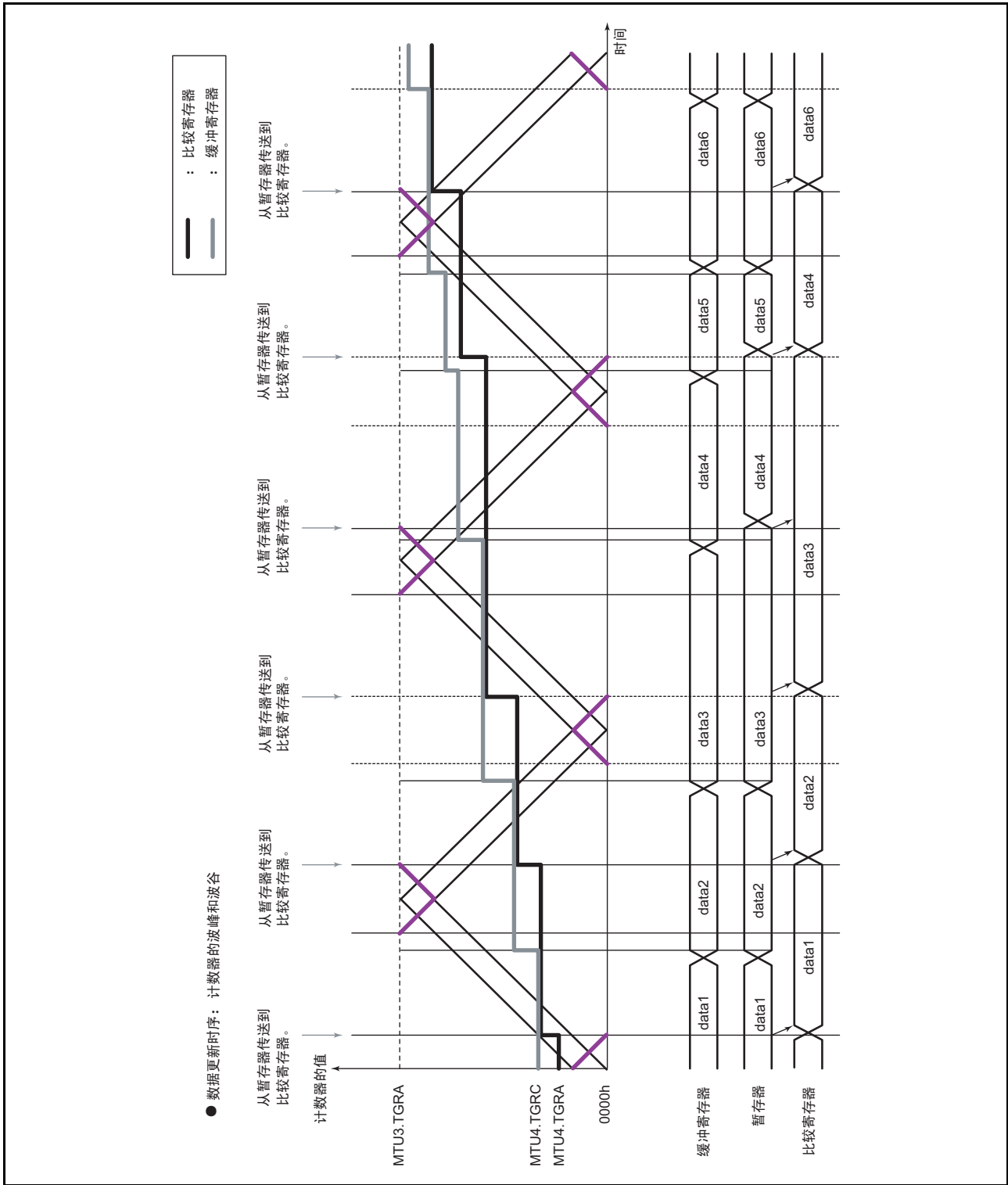


图 21.43 互补 PWM 模式的数据更新例子

(i) 互补 PWM 模式的初始输出

在互补 PWM 模式中，通过设定定时器的输出控制寄存器 1 (TOCR1) 的 OLSN 位和 OLSP 位或者定时器的输出控制寄存器 2 (TOCR2) 的 OLS1N ~ OLS3N 位和 OLS1P ~ OLS3P 位，决定初始输出。

此初始输出为 PWM 脉冲的无效电平，从通过定时器的模式寄存器 (TMDR) 设定互补 PWM 模式开始到 MTU4.TCNT 大于死区时间寄存器 (TDDR) 设定的值前，输出此初始输出。互补 PWM 模式的初始输出例子如图 21.44 所示。

PWM 占空比的初始值小于 TDDR 寄存器的值时的波形例子如图 21.45 所示。

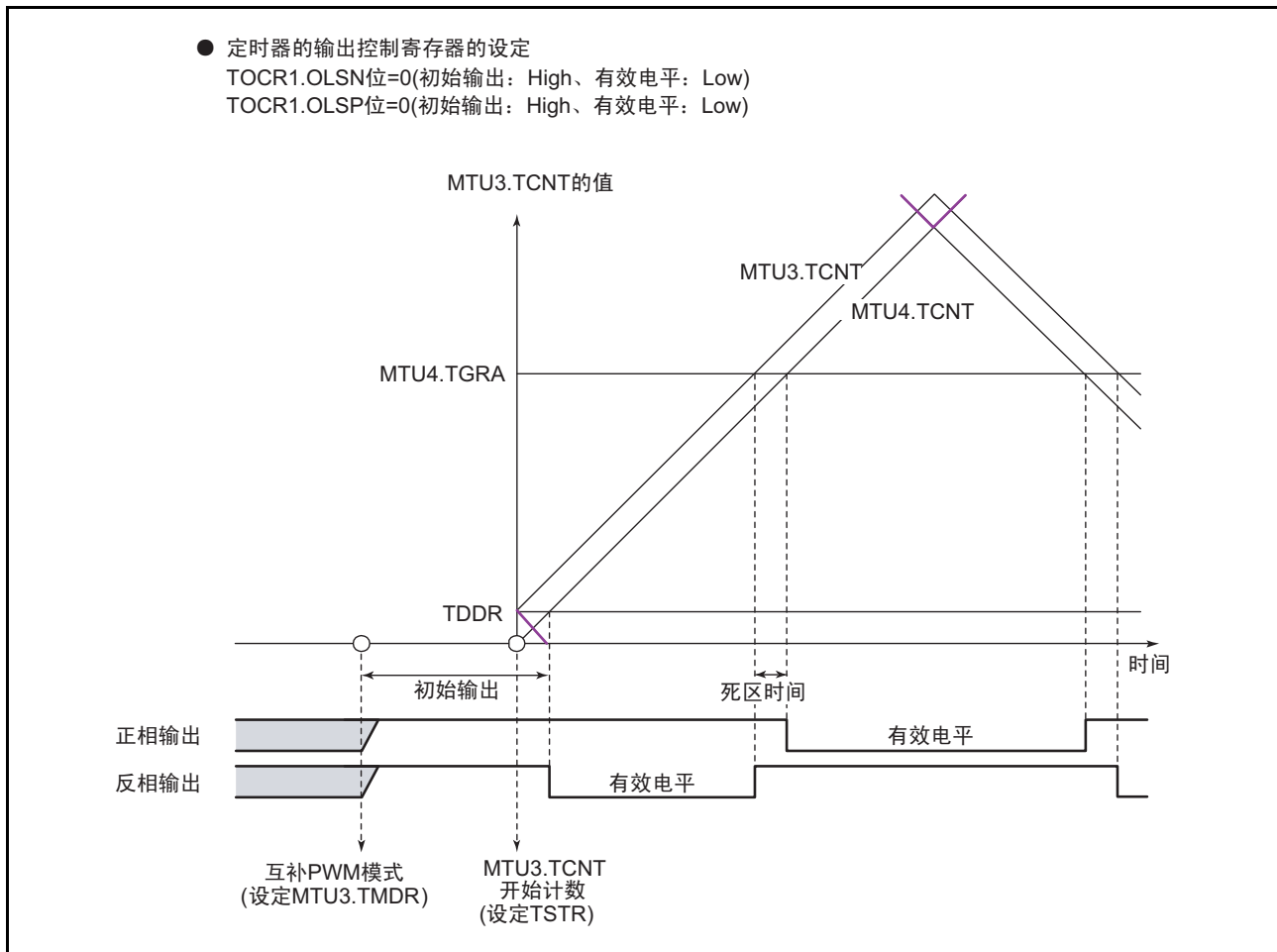


图 21.44 互补 PWM 模式的初始输出例子 (1)

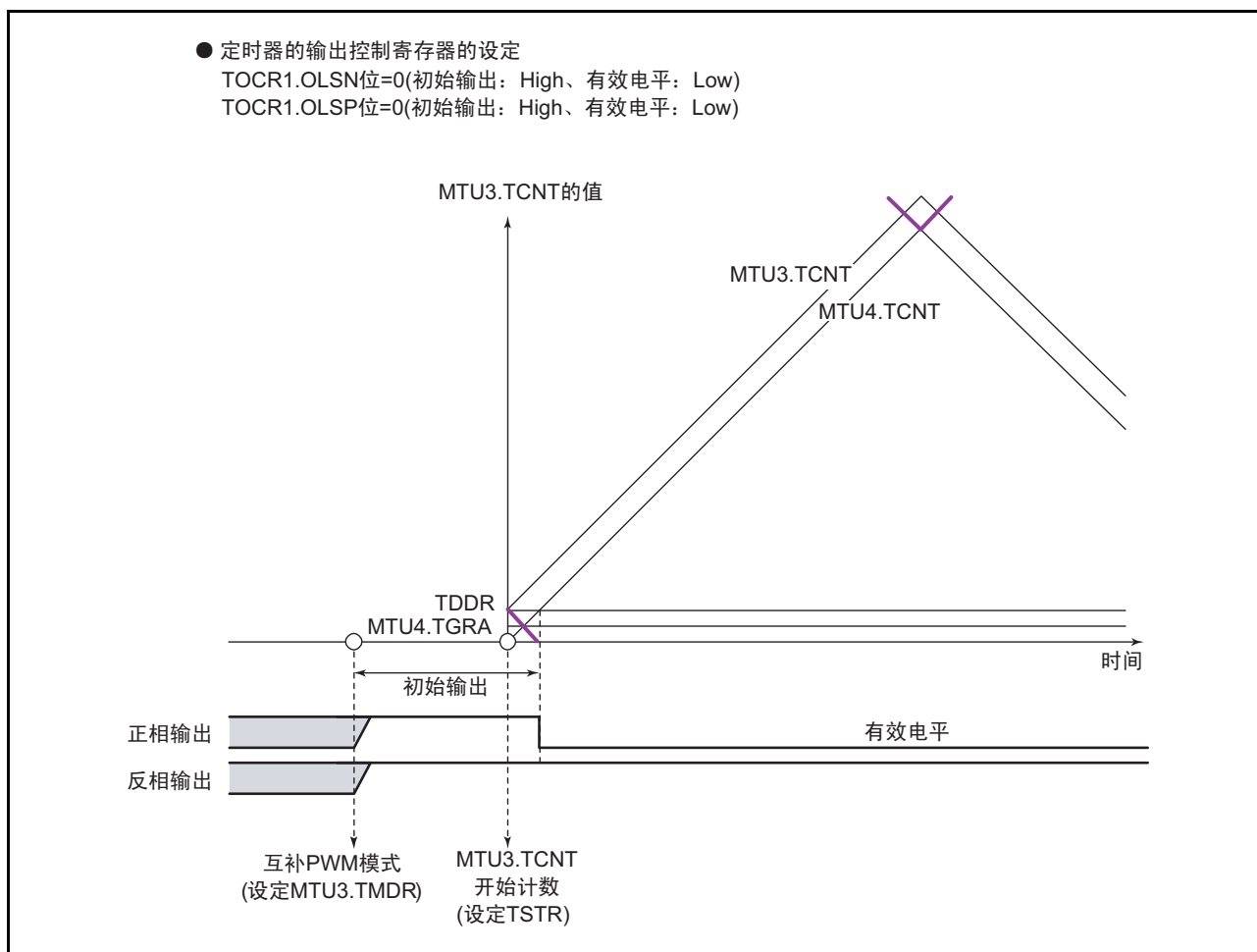


图 21.45 互补 PWM 模式的初始输出例子 (2)

(j) 互补 PWM 模式的 PWM 输出生成方法

在互补 PWM 模式中，将正相和反相有非重叠时间关系的 PWM 波形进行 3 相输出，此非重叠时间称为死区时间。

在发生计数器和数据寄存器的比较匹配时，通过输出定时器的输出控制寄存器选择的输出电平，生成 PWM 波形。在 TCNTS 进行计数的期间，因为产生 0 ~ 100% 的连续 PWM 脉冲，所以同时比较数据寄存器的值和暂存器的值。此时，ON/OFF 比较匹配的产生时序会有前后，为了确保死区时间并且使正相 / 反相的 ON 时间不重叠，必须优先使各相 OFF 的比较匹配。互补 PWM 模式的波形生成例子如图 21.46 ~ 图 21.48 所示。

通过和实线计数器的比较匹配，生成正相 / 反相的 OFF 时序；通过和点线计数器（比实线计数器迟死区时间）的比较匹配，生成 ON 时序。在 T1 期间，最优先使反相 OFF 的 a 的比较匹配，忽视比 a 先产生的比较匹配。在 T2 期间，最优先使正相 OFF 的 c 的比较匹配，忽视比 c 先产生的比较匹配。

如图 21.46 所示，通常按照 a→b→c→d（或者 c→d→a'→b'）的顺序产生比较匹配。

当不按 a→b→c→d 的顺序产生比较匹配时，因为反相的 OFF 时间短于 2 倍的死区时间，所以表示正相不为 ON；当不按 c→d→a'→b' 的顺序产生比较匹配时，因为正相的 OFF 时间短于 2 倍的死区时间，所以表示反相不为 ON。

如图 21.47 所示，如果在 a 的比较匹配之后先产生 c 的比较匹配，就忽视 b 的比较匹配，而通过 d 的比较匹配使反相 OFF。这是因为比 b 的比较匹配（正相 ON 时序）先产生正相 OFF 的 c 的比较匹配而优先使正相 OFF（因为正相从 OFF 变为 OFF，所以波形不变）。

同样地，在图 21.48 所示的例子中，比 c 的比较匹配先产生和暂存器的新数据比较匹配的 a'，但是在产生使正相 OFF 的 c 前忽视其他比较匹配，因此不使反相 ON。

如此，在互补 PWM 模式中，优先 OFF 时序的比较匹配，即使 ON 时序的比较匹配比 OFF 先产生，也被忽视。

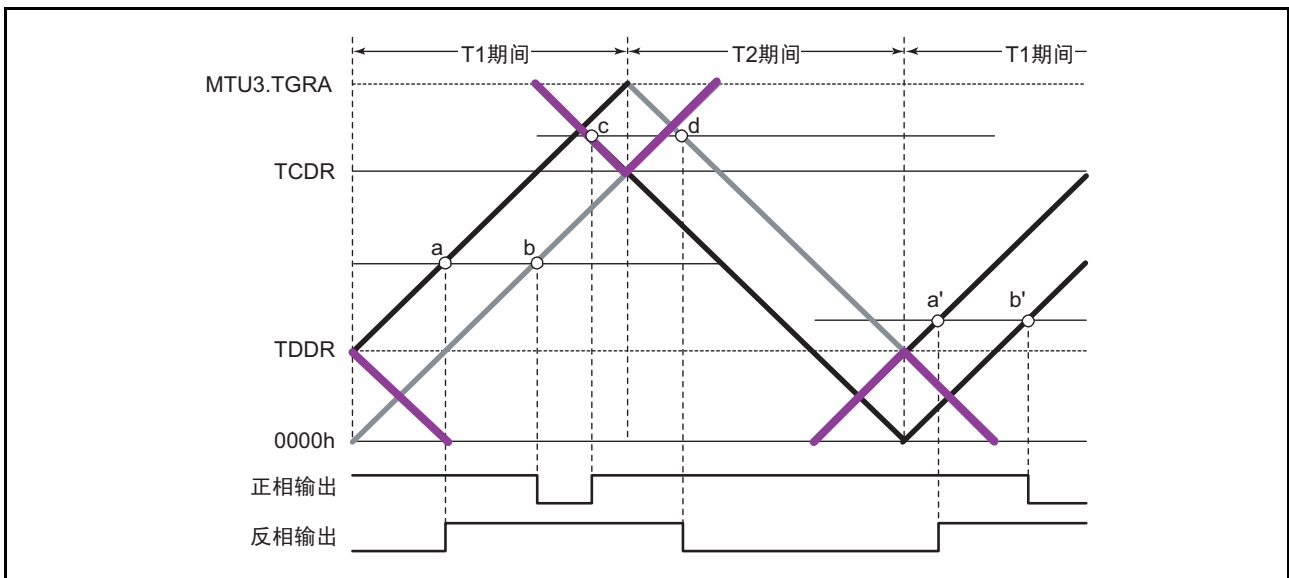


图 21.46 互补 PWM 模式的波形输出例子 (1)

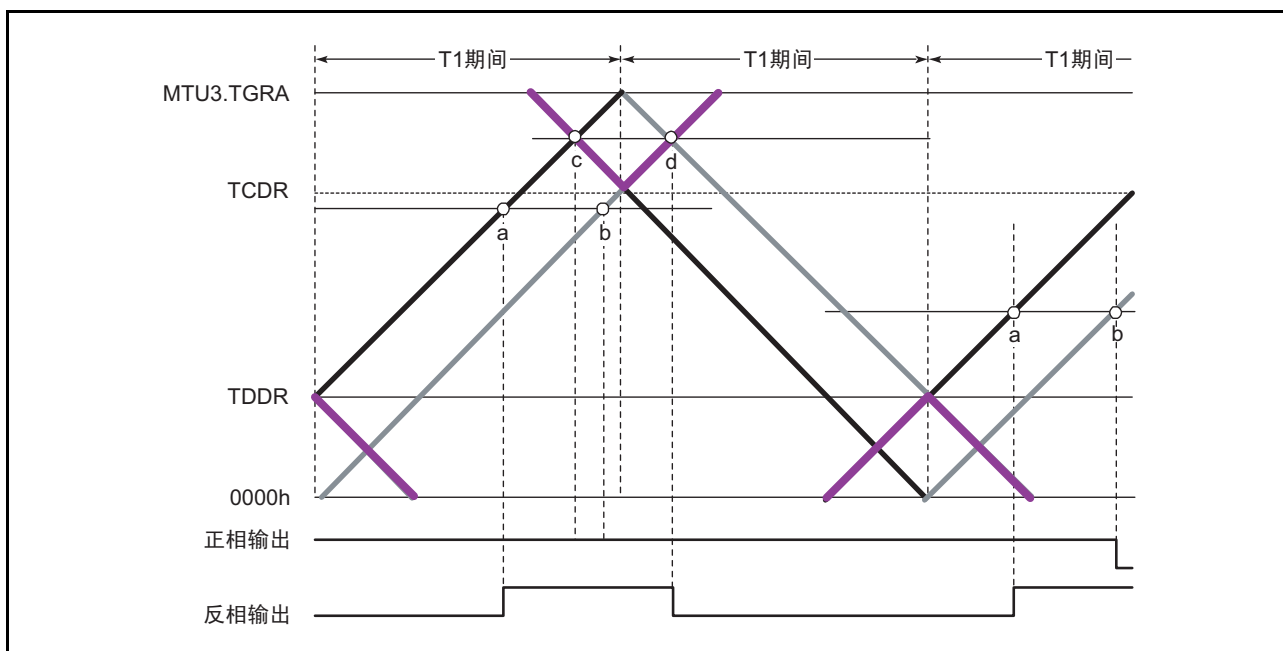


图 21.47 互补 PWM 模式的波形输出例子 (2)

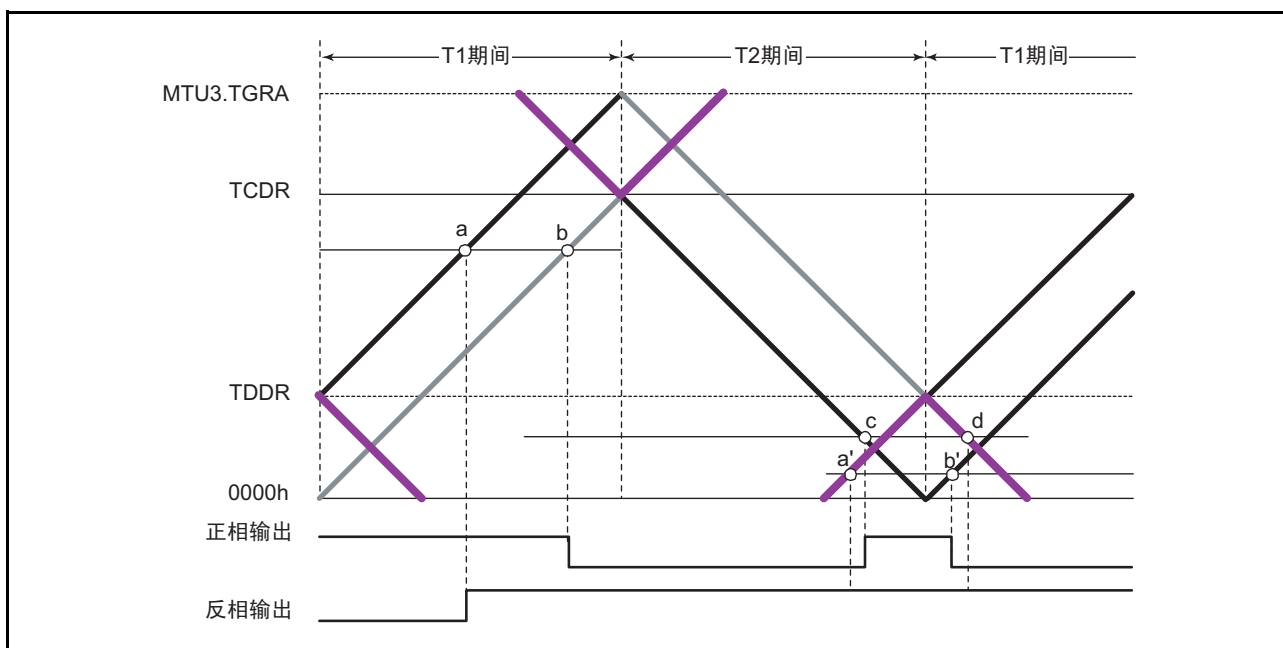


图 21.48 互补 PWM 模式的波形输出例子 (3)

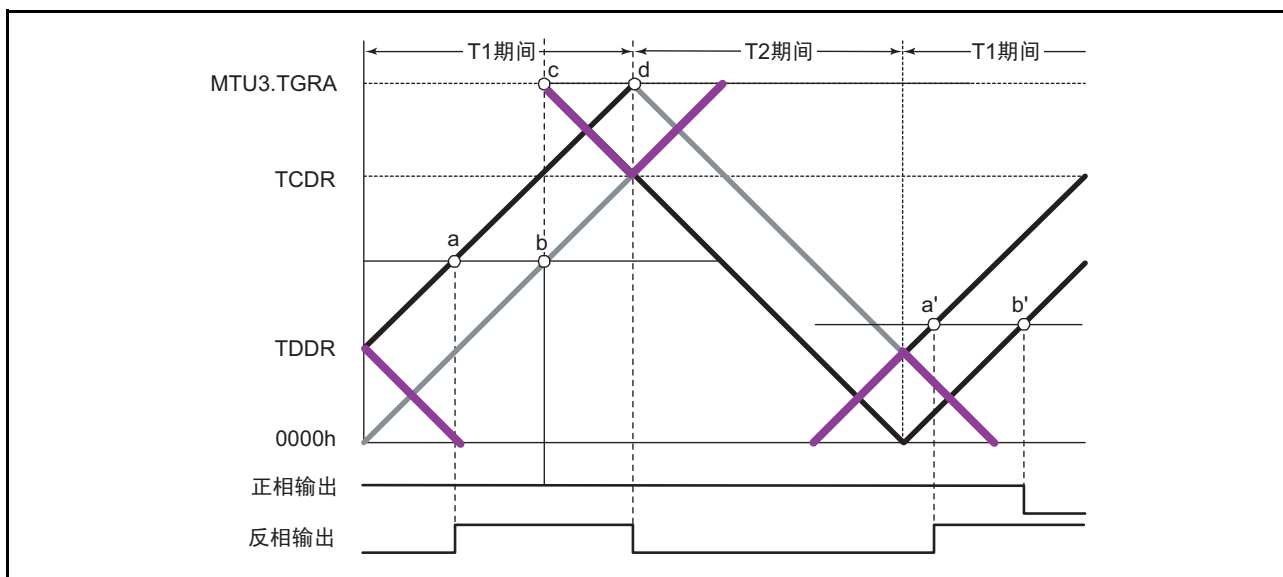


图 21.49 互补 PWM 模式的 0%、100% 波形输出例子 (1)

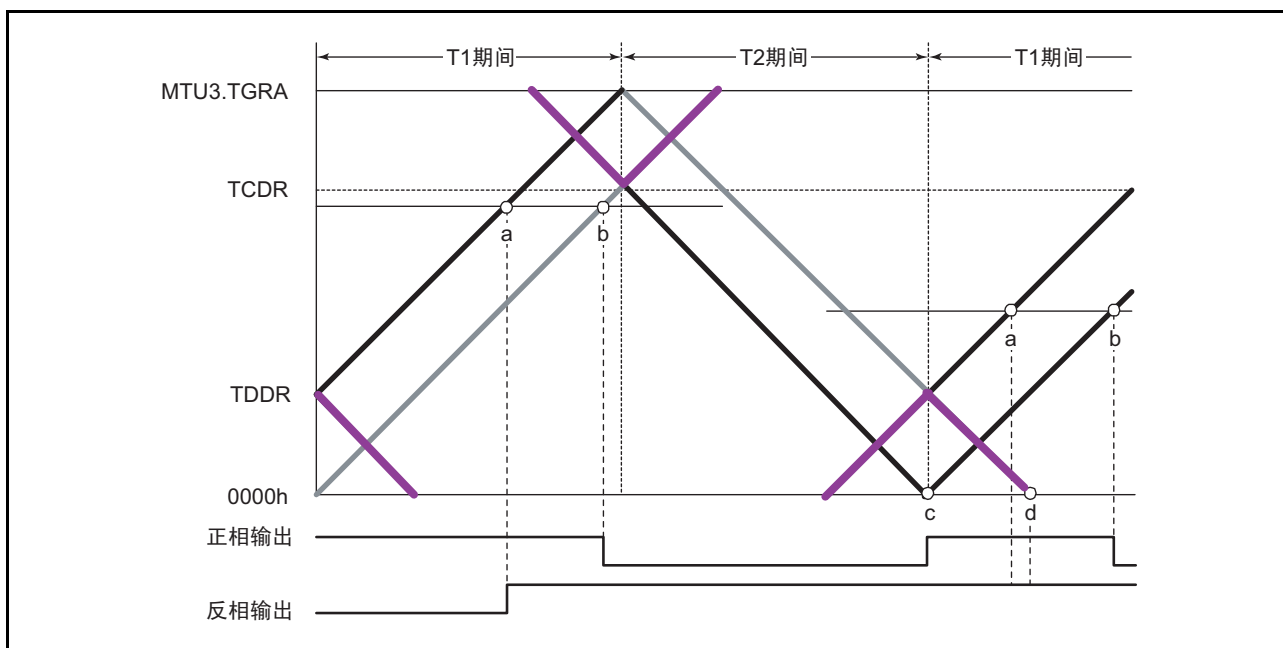


图 21.50 互补 PWM 模式的 0%、100% 波形输出例子 (2)

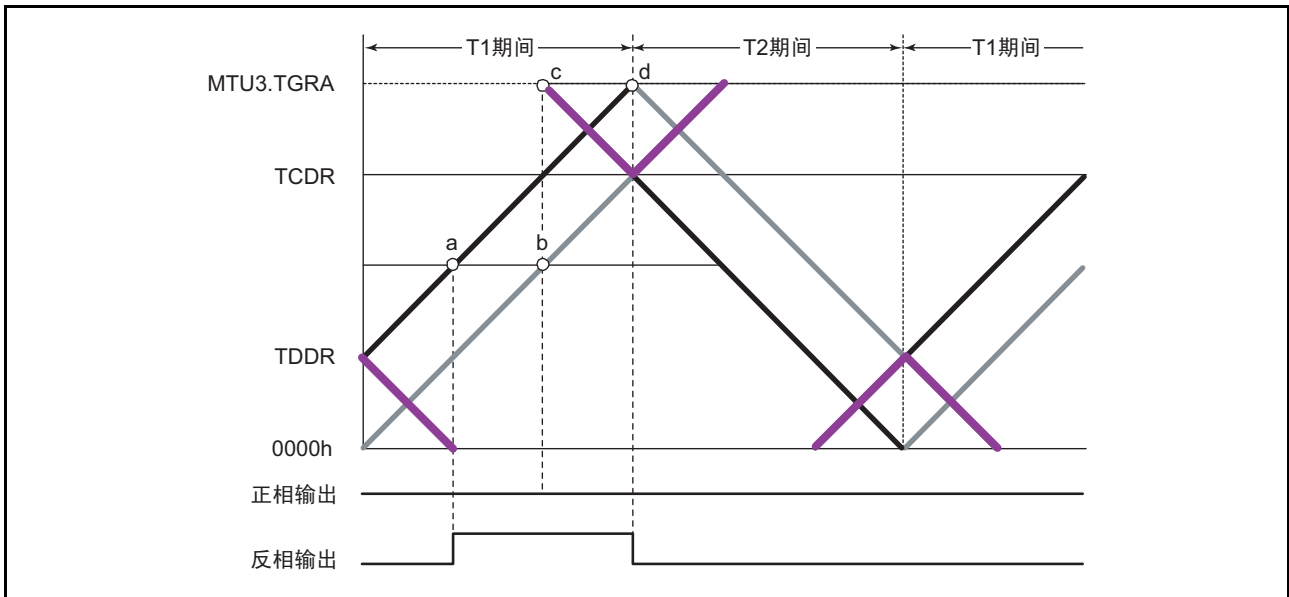


图 21.51 互补 PWM 模式的 0%、100% 波形输出例子 (3)

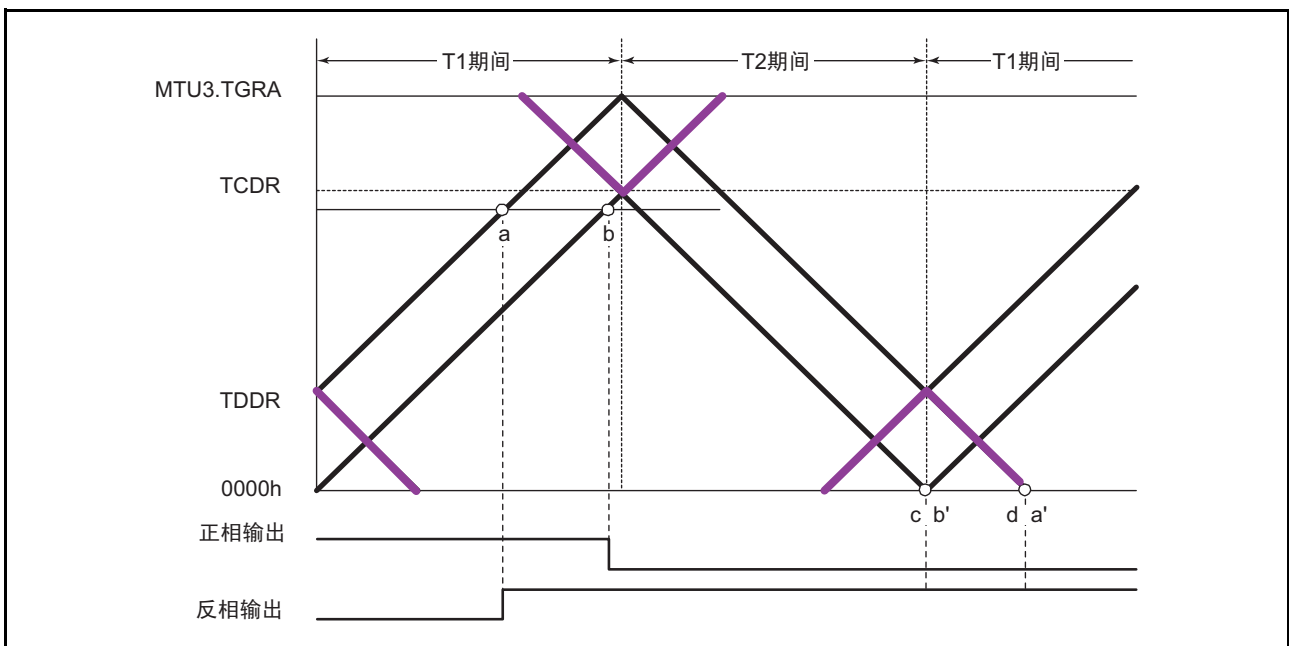


图 21.52 互补 PWM 模式的 0%、100% 波形输出例子 (4)

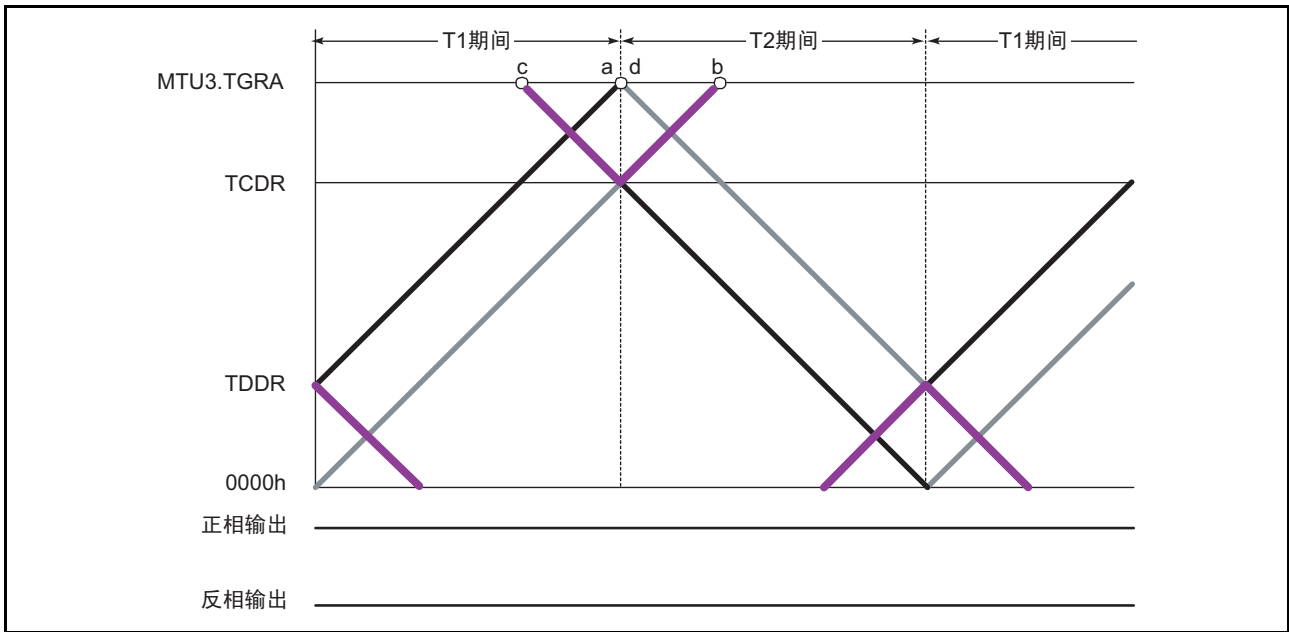


图 21.53 互补 PWM 模式的 0%、100% 波形输出例子 (5)

(k) 互补 PWM 模式的 0%、100% 占空比输出

在互补 PWM 模式中，能任意输出 0%、100% 的占空比，输出例子如图 21.49 ~ 图 21.52 所示。

如果将数据寄存器的值设定为“0000h”，就输出 100% 的占空比。此时的波形是正相为 100%ON 状态的波形。如果将数据寄存器的值设定为和 MTU3.TGRA 相同的值，就输出 0% 的占空比。此时的波形是正相为 100% OFF 状态的波形。

此时，同时产生 ON 和 OFF 的比较匹配，如果同相的 ON 比较匹配和 OFF 比较匹配同时产生，双方的比较匹配就都被忽视而波形不变。

(l) 与 PWM 周期同步的交替输出

在互补 PWM 模式中，能通过将定时器的输出控制寄存器 (TOCR) 的 PSYE 位置“1”，进行与 PWM 载波周期同步的交替输出。交替输出的波形例子如图 21.54 所示。

通过 MTU3.TCNT 和 MTU3.TGRA 的比较匹配以及 MTU4.TCNT 和“0000h”的比较匹配进行交替输出。

此交替输出的输出引脚为 MTIOC3A 引脚，初始输出为 High 电平输出。

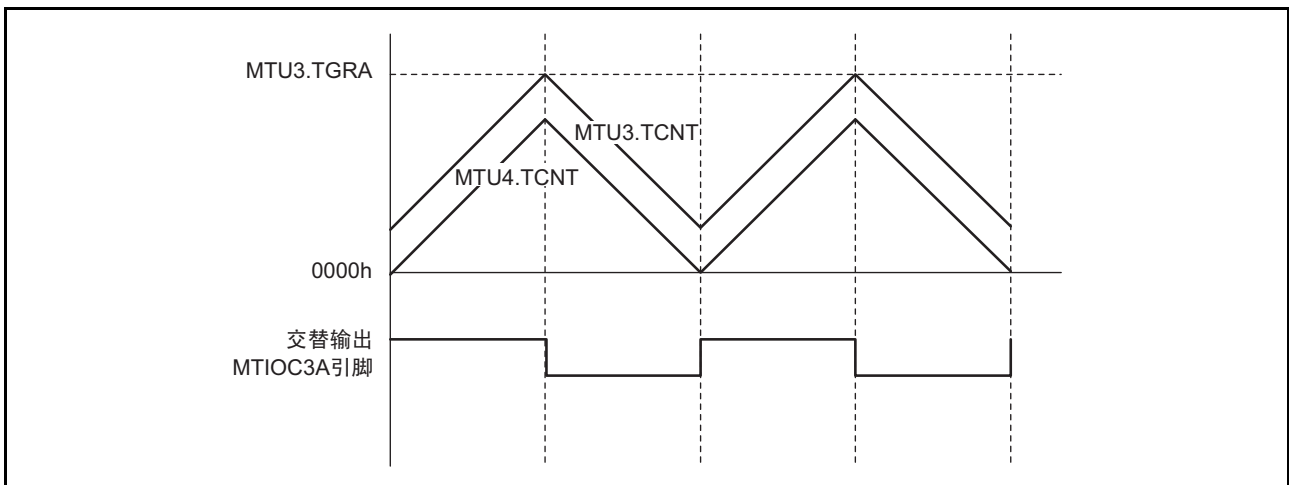


图 21.54 与 PWM 输出同步的交替输出波形例子

(m) 其他通道的计数器清除

在互补 PWM 模式中，当通过定时器的同步寄存器 (TSYR) 设定为与其他通道同步的模式并且通过定时器的控制寄存器 (TCR) 的 CCLR[2:0] 位选择同步清除时，能由其他通道进行 MTU3.TCNT、MTU4.TCNT 和 TCNTS 的清除。

运行例子如图 21.55 所示。

使用此功能，能通过外部信号进行计数器清除和重新开始。

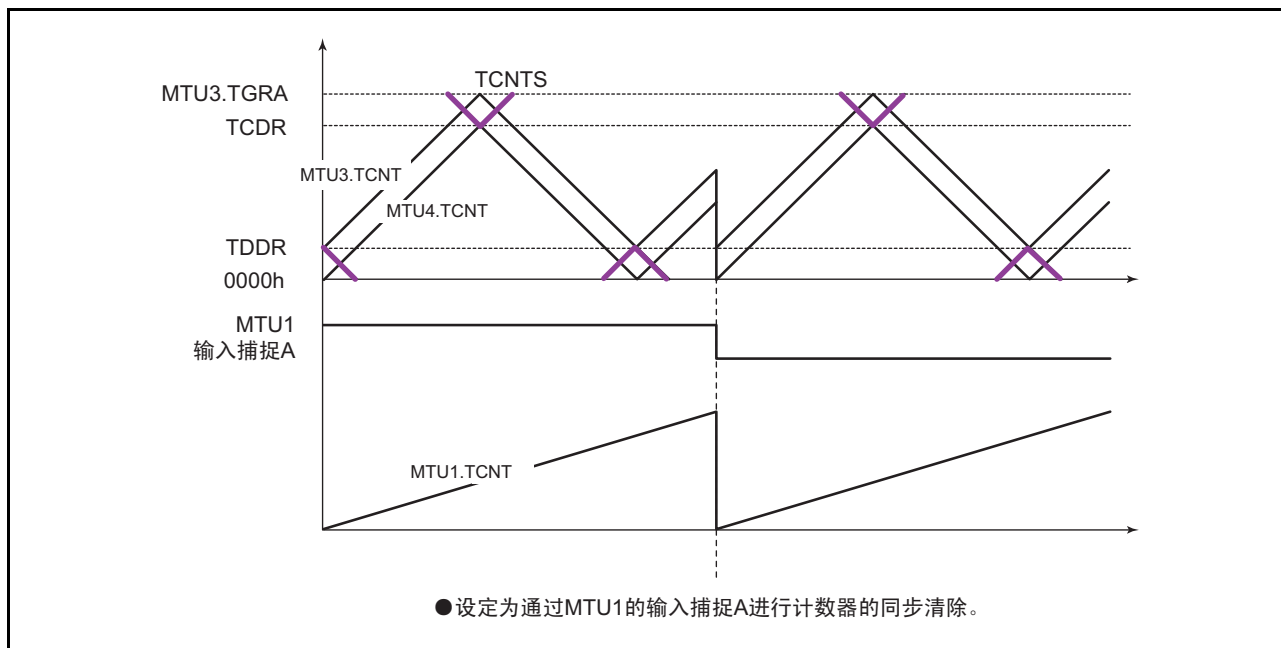


图 21.55 与其他通道同步的计数器清除

(n) 互补 PWM 模式的同步计数器清除时的输出波形控制

能通过将 TWCR 寄存器的 WRE 位置“1”，抑止在互补 PWM 模式的波谷 Tb 区间发生同步计数器清除时的初始输出。因此，能抑止同步计数器清除时占空比的急剧变化。

只有在如图 21.56 的⑩、⑪的波谷 Tb 区间进行同步清除时，才能通过将 WRE 位置“1”来抑止初始输出。如果在其他时序中发生同步清除，就输出 TOCR 寄存器的 OLSN 位和 OLSP 位设定的初始值。即使在波谷的 Tb 区间，如果在图 21.56 的①所示的计数器开始计数后的初始输出期间发生同步清除，也不抑止初始输出。

MTU 的计数器清除源为 MTU0 ~ 2 的同步清除。

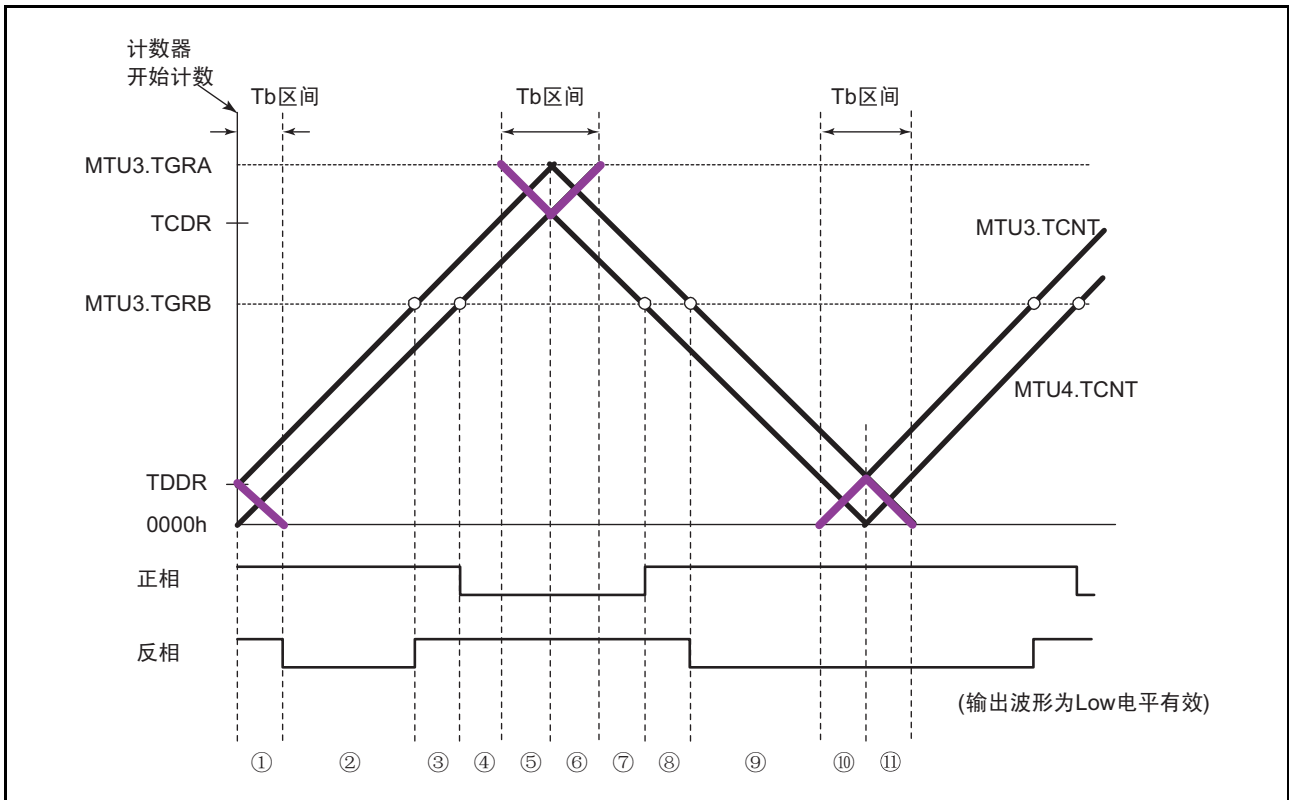


图 21.56 同步计数器的清除时序

- 互补 PWM 模式的同步计数器清除时输出波形控制的设定步骤例子
互补 PWM 模式的同步计数器清除时输出波形控制的设定步骤例子如图 21.57 所示。

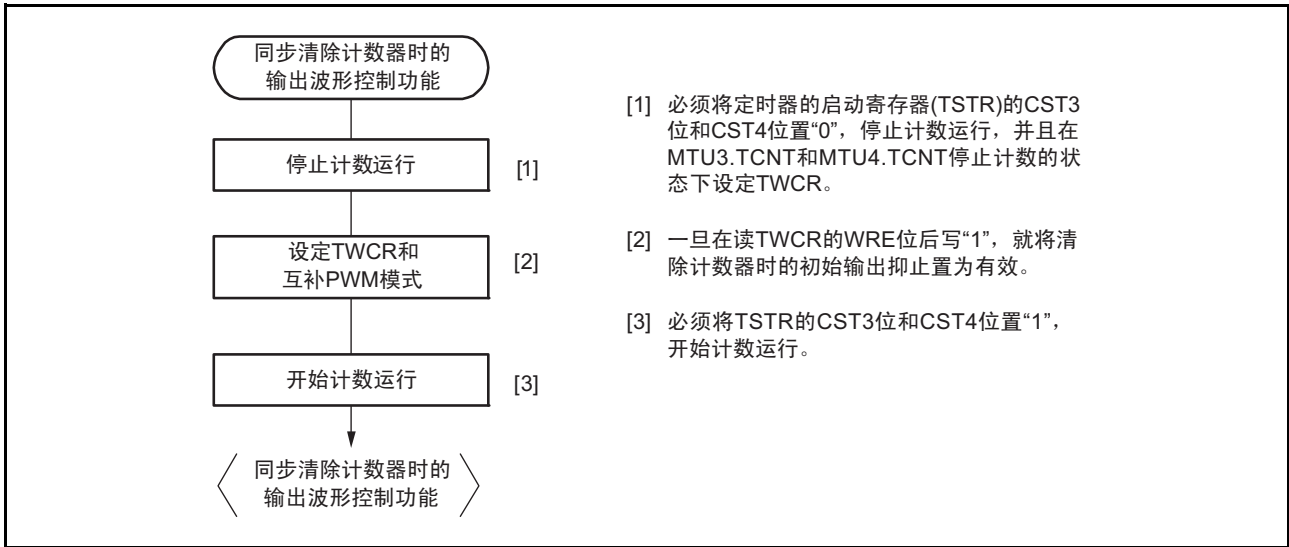


图 21.57 互补 PWM 模式的同步计数器清除时输出波形控制的设定步骤例子

- 互补 PWM 模式的同步计数器清除时输出波形控制的运行例子
在将 TWCR 的 WRE 位置“1”的状态下使 MTU 进行互补 PWM 运行并且进行同步计数器清除时的运行例子如图 21.58 ~ 图 21.61 所示。在此，图 21.58 ~ 图 21.61 的同步计数器清除时序分别是图 21.56 的③、⑥、⑧、⑩所示的时序。

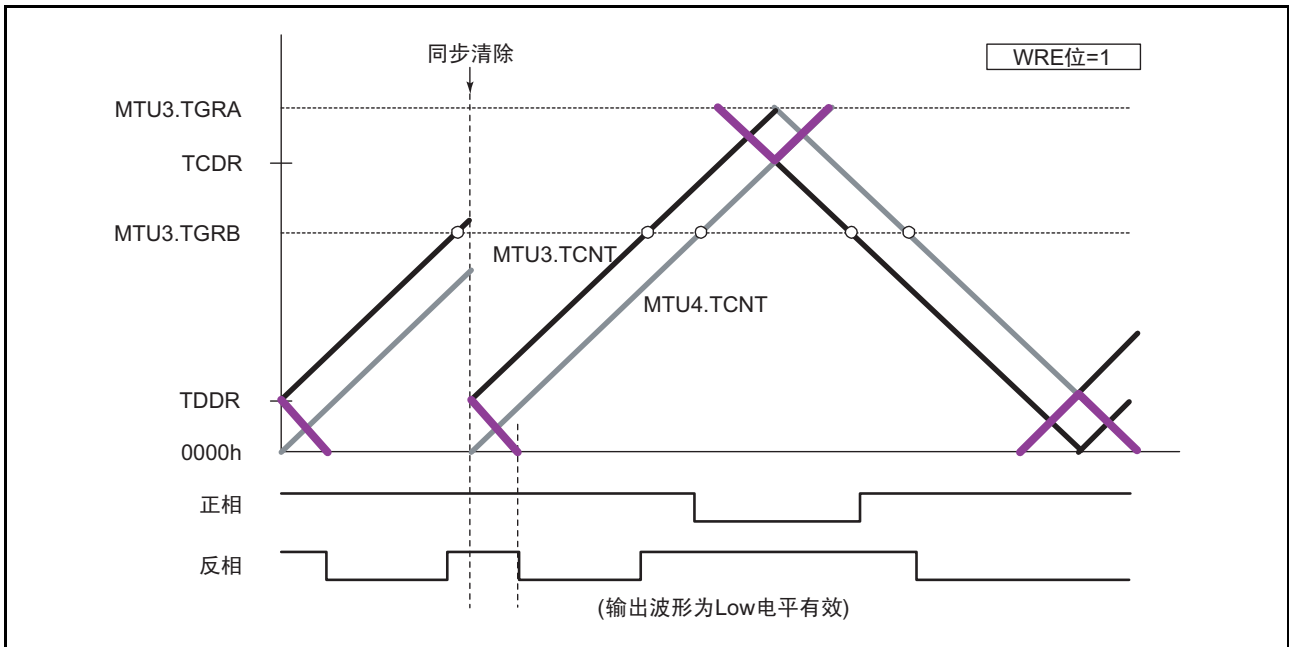


图 21.58 在递增计数过程中的死区时间发生同步清除的情况
(图 21.56 的时序③、MTU 的 TWCR 寄存器的 WRE 位 = 1)

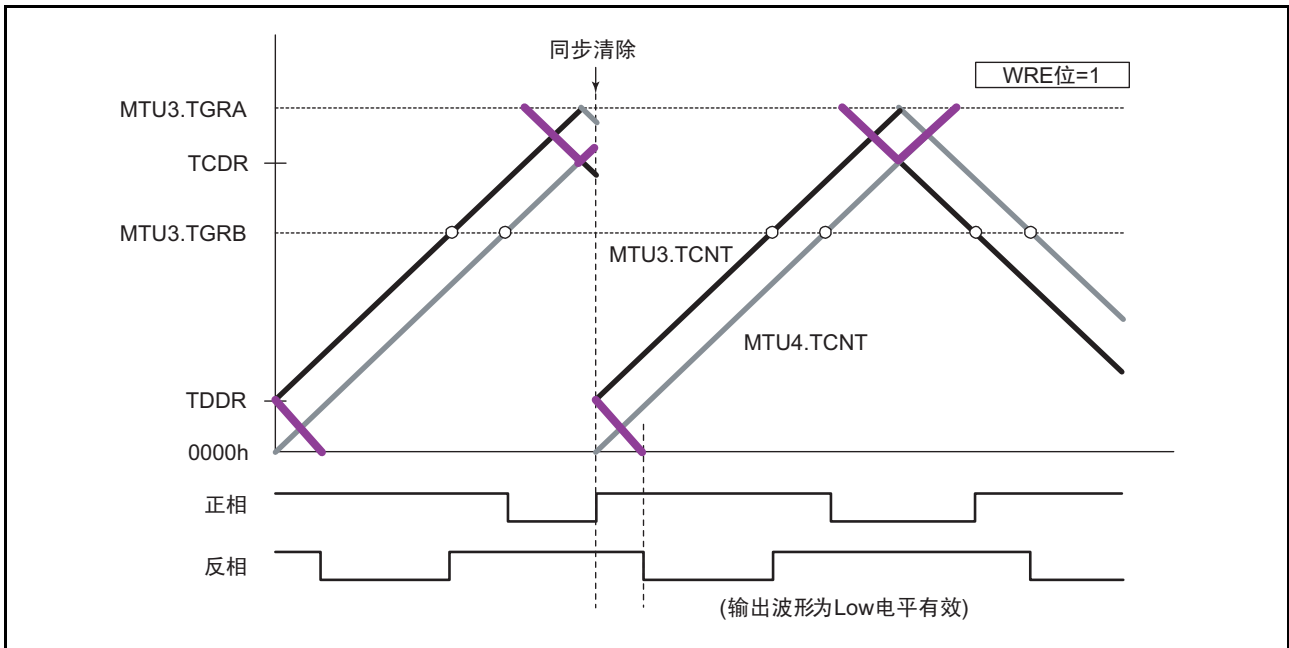


图 21.59 在波峰 Tb 区间发生同步清除的情况
 (图 21.56 的时序⑥、MTU 的 TWCR 寄存器的 WRE 位 =1)

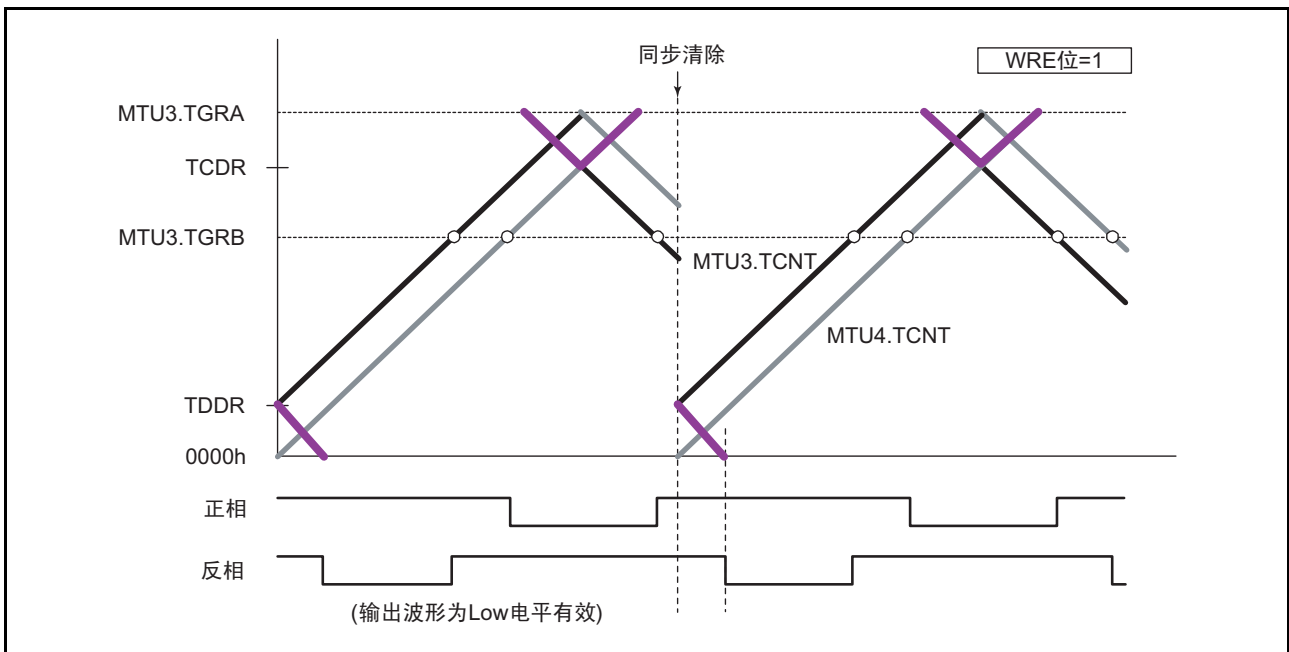


图 21.60 在递减计数过程中的死区时间发生同步清除的情况
 (图 21.56 的时序⑧、TWCR 寄存器的 WRE 位 =1)

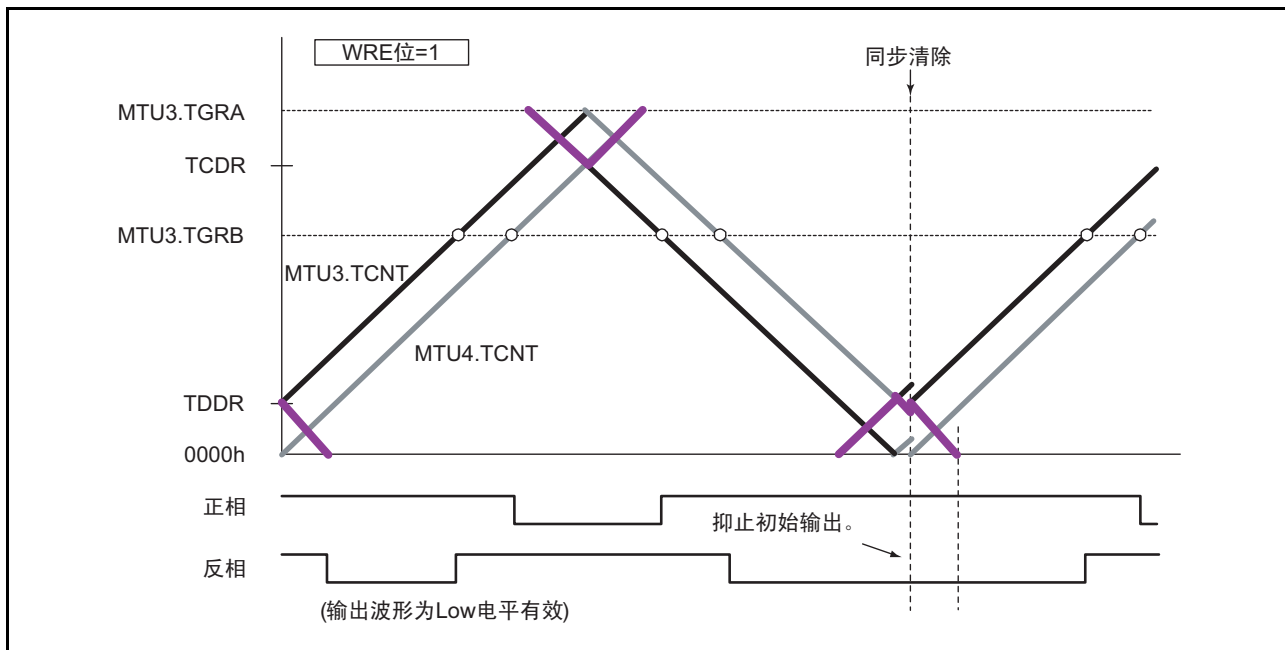


图 21.61 在波谷 Tb 区间发生同步清除的情况 (图 21.56 的时序 ①、TWCR 寄存器的 WRE 位 =1)

(o) 通过 MTU3.TGRA 的比较匹配进行的计数器清除

在互补 PWM 模式中，能通过设定定时器的波形控制寄存器 (TWCR) 的 CCE 位，在发生 MTU3.TGRA 的比较匹配时清除 MTU3.TCNT、MTU4.TCNT 和 TCNTS，运行例子如图 21.62 所示。

- 注 . 只能在互补 PWM 模式 1 (在波峰进行传送) 中使用。
- 不能设定为与其他通道同步清除功能 (不能将定时器的同步寄存器 (TSYR) 的 SYNC0 ~ SYNC4 位置“1”)。
- 不能将 PWM 占空比设定为“0000h”。
- 不能将定时器的输出控制寄存器 1 (TOCR1) 的 PSYE 位置“1”。

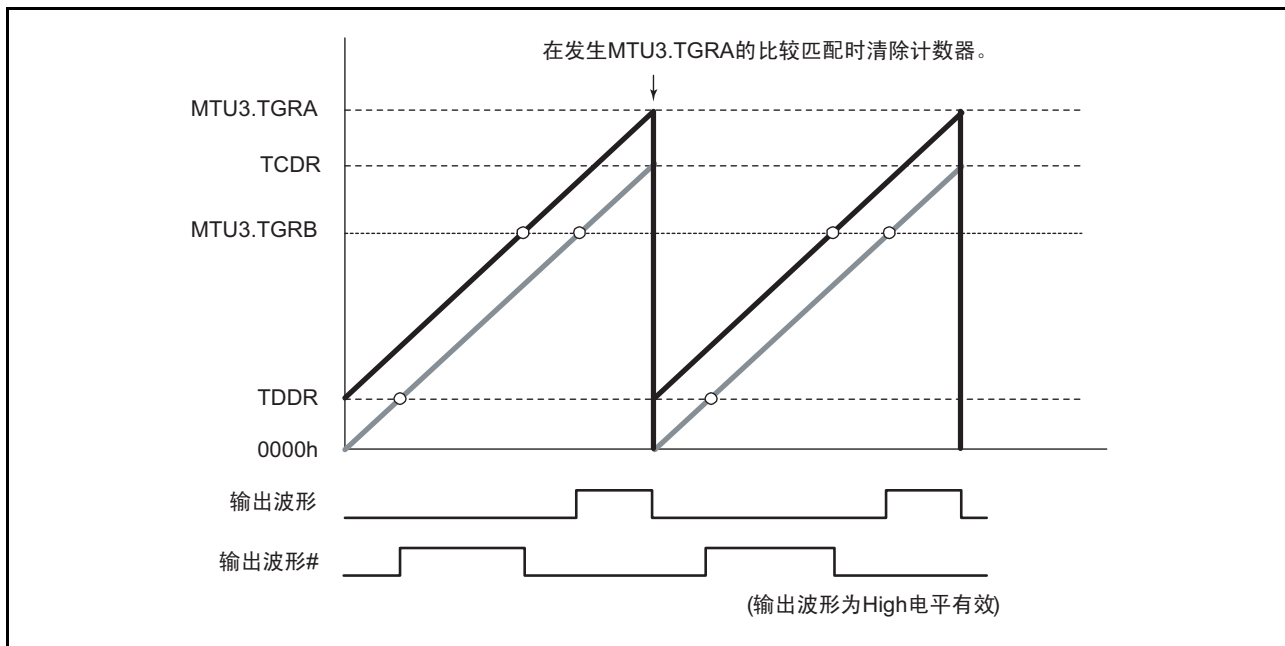


图 21.62 MTU3.TGRA 的比较匹配时的计数器清除的运行例子

(p) AC 同步马达 (无刷 DC 马达) 的驱动波形输出例子

在互补 PWM 模式中, 能使用定时器的门控寄存器 (TGCR), 简单地控制无刷 DC 马达。使用 TGCR 无刷 DC 马达的驱动波形例子如图 21.63 ~ 图 21.66 所示。

在通过使用霍尔元件等检测到的外部信号进行 3 相无刷 DC 马达的输出相转换时, 将 TGCR 寄存器的 FB 位置“0”。此时, 将表示磁极位置的外部信号输入到 MTU0 的定时器输入引脚 MTIOC0A、MTIOC0B、MTIOC0C 引脚 (必须通过 PFS 寄存器进行设定)。如果在 MTIOC0A、MTIOC0B、MTIOC0C 引脚产生边沿, 输出的 ON/OFF 就自动进行转换。

在 FB 位为“1”的情况下, 如果将 TGCR 的 UF 位、VF 位和 WF 位置“0”或者“1”, 输出的 ON/OFF 就进行转换。

从互补 PWM 模式的 6 相输出引脚输出驱动波形。

对于此 6 相输出, 能通过将 N 位或者 P 位置“1”, 在 ON 输出时使用互补 PWM 模式的输出, 进行斩波输出。如果 N 位或者 P 位为“0”, 就为电平输出。

6 相输出的有效电平 (ON 输出时的电平) 与 N 位和 P 位的设定无关, 能通过定时器的输出控制寄存器 (TOCR) 的 OLSN 位和 OLSP 位进行设定。

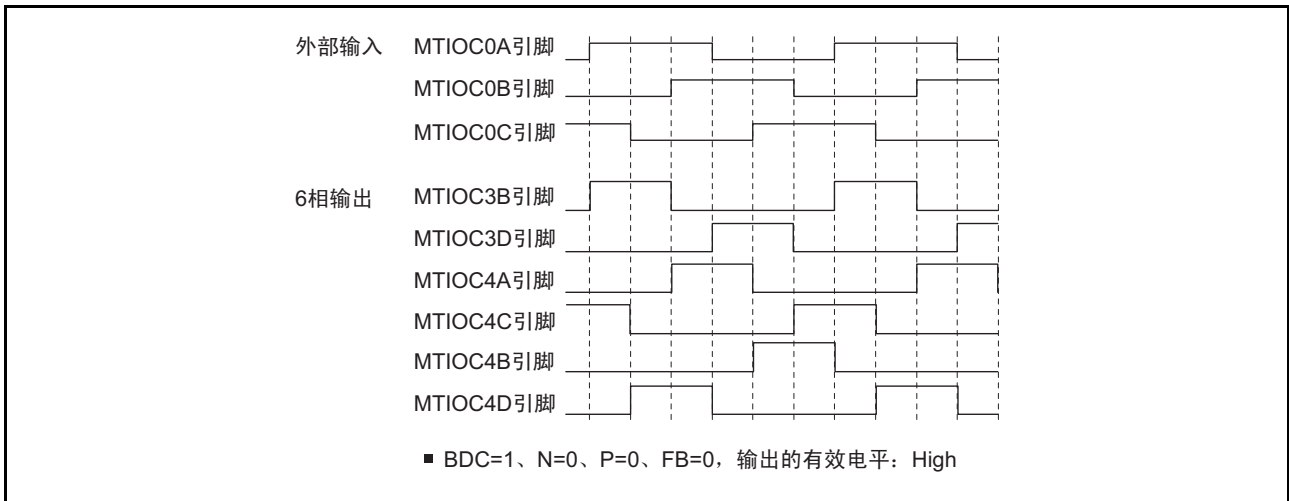


图 21.63 通过外部输入进行输出相转换的运行例子 (1)

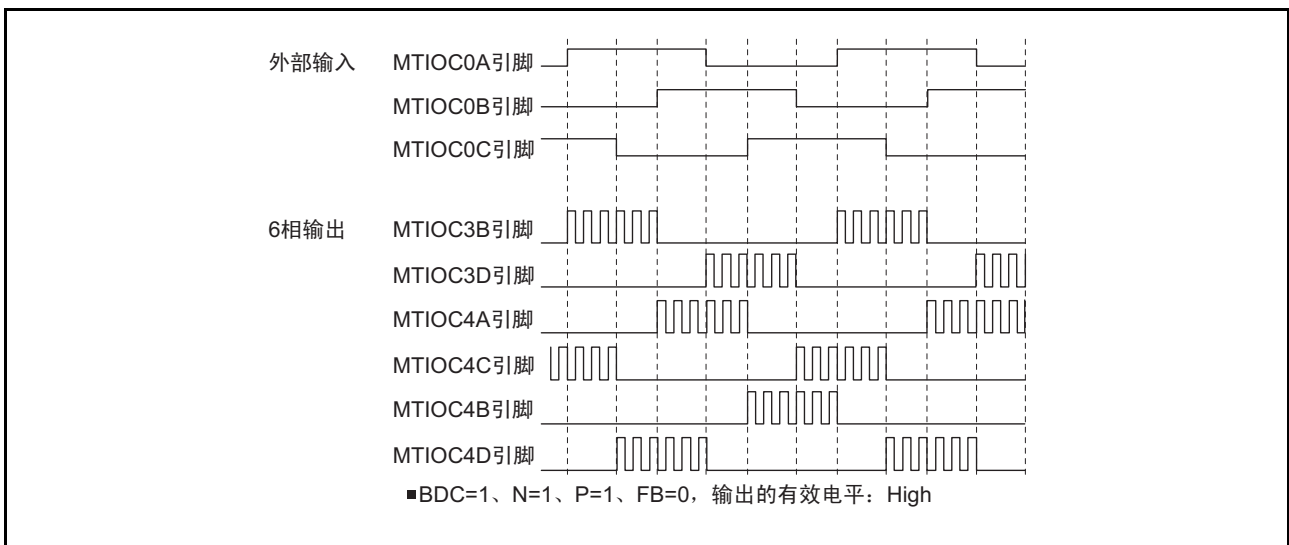


图 21.64 通过外部输入进行输出相转换的运行例子 (2)

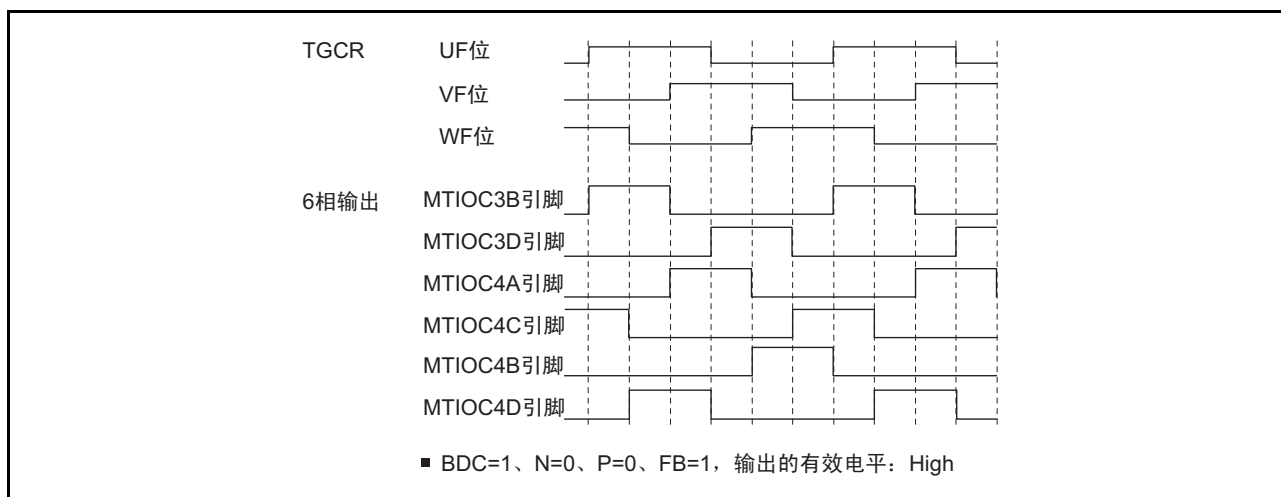


图 21.65 通过设定 UF 位、VF 位和 WF 位进行输出相转换的运行例子 (1)

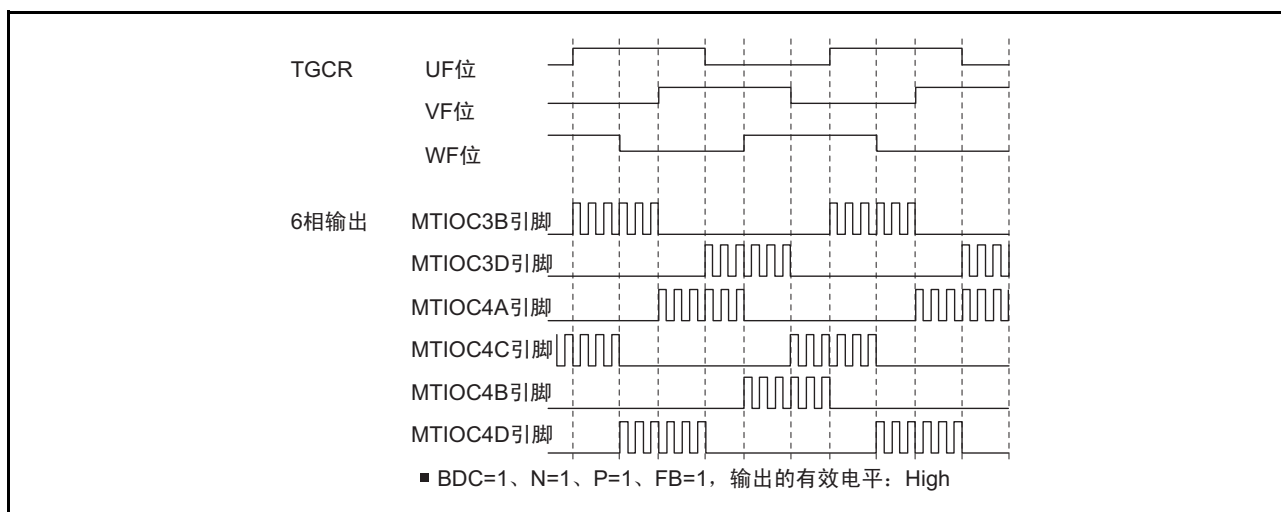


图 21.66 通过设定 UF 位、VF 位和 WF 位进行输出相转换的运行例子 (2)

(q) A/D 转换开始请求的设定

在互补 PWM 模式中，能使用 MTU3.TGRA 的比较匹配、MTU4.TCNT 的下溢（波谷）、MTU3 和 MTU4 以外通道的比较匹配，请求开始 A/D 转换。

如果使用 MTU3.TGRA 的比较匹配设定开始请求，就能在 MTU3.TCNT 的波峰开始 A/D 转换。

能通过将定时器的中断允许寄存器（TIER）的 TTGE 位置“1”，设定 A/D 转换的开始请求，并且能通过将 MTU4.TIER 的 TTGE2 位置“1”，设定 MTU4.TCNT 下溢（波谷）的 A/D 转换开始请求。

(3) 互补 PWM 模式的中断减少功能

能通过设定定时器的中断减少设定寄存器 (TITCR)，使 MTU3 和 MTU4 的 TGIA3 (波峰的中断) 以及 TCIV4 (波谷的中断) 最多减少 7 次中断。

能通过设定定时器的缓冲传送寄存器 (TBTER)，联动缓冲寄存器到暂存器 / 比较寄存器的传送来减少中断。有关和缓冲寄存器的联动，请参照“(c) 联动中断减少功能的缓冲传送控制”。

能通过设定定时器的 A/D 转换请求控制寄存器 (TADCR)，联动 A/D 转换开始请求延迟功能的 A/D 转换开始请求来减少中断。有关和 A/D 转换开始请求延迟功能的联动，请参照“21.3.9 A/D 转换开始请求的延迟功能”。

必须通过设定 MTU3.TIER 寄存器和 MTU4.TIER 寄存器，在禁止 TGIA3 和 TCIV4 中断请求的状态下并且在不发生比较匹配以及不发生由比较匹配产生的 TGIA3 中断请求和 TGIA4 中断请求的状态下，设定定时器的中断减少设定寄存器 (TITCR)。在更改减少次数前，必须将 T3AEN 位和 T4VEN 位置“0”，清除减少计数器。

(a) 中断减少功能的设定步骤例子

中断减少功能的设定步骤例子和中断减少次数的可变更期间分别如图 21.67 和图 21.68 所示。

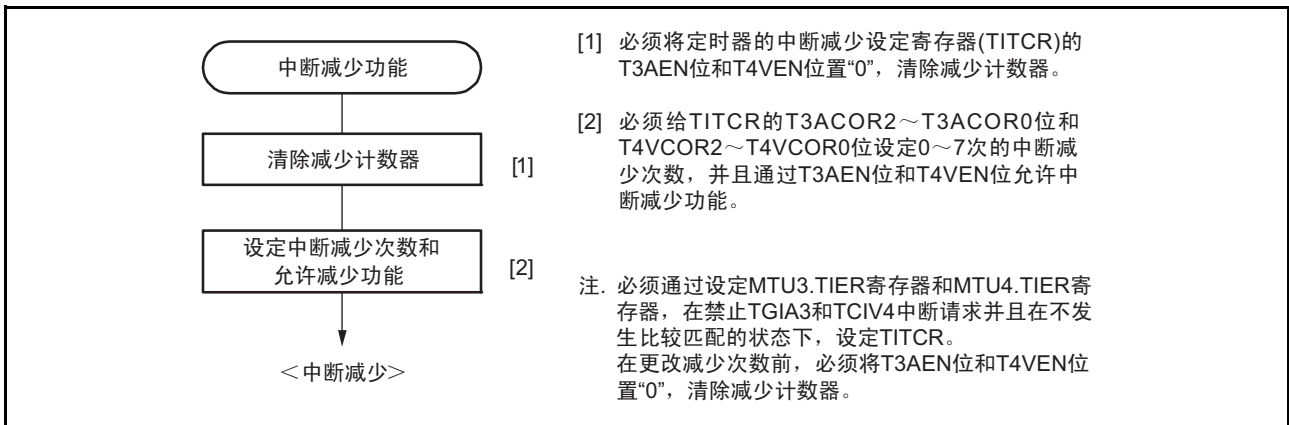


图 21.67 中断减少功能的设定步骤例子

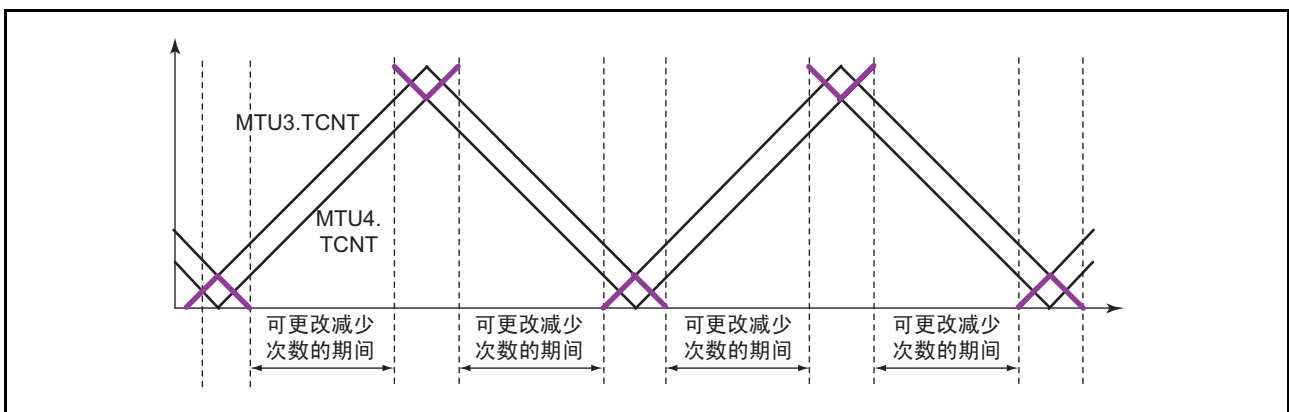


图 21.68 中断减少次数的可变更期间

(b) 中断减少功能的运行例子

通过定时器的中断减少设定寄存器 (TITCR) 的 T3ACOR 位将中断减少次数设定为 3 次并且将 T3AEN 位置“1”时的 MTU3.TGIA 中断减少的运行例子如图 21.69 所示。

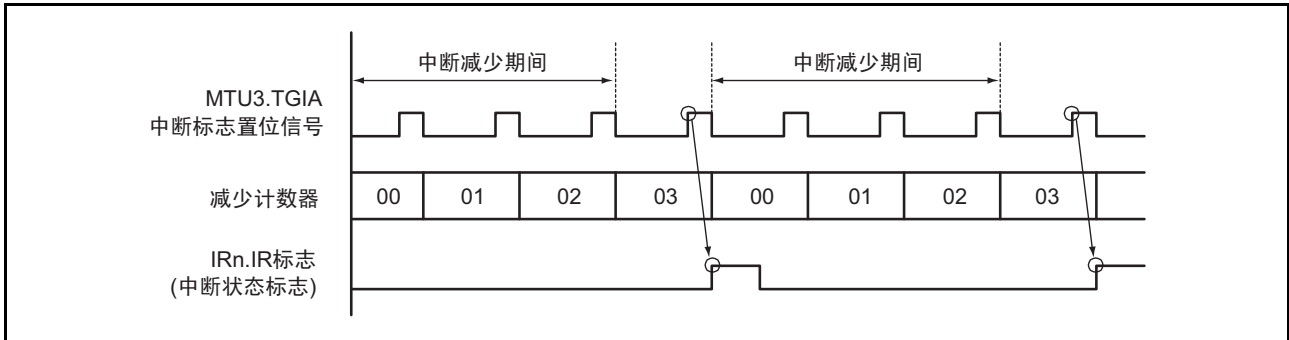


图 21.69 中断减少功能的运行例子

(c) 联动中断减少功能的缓冲传送控制

能通过设定定时器的缓冲传送设定寄存器 (TBTER) 的 BTE1 位和 BTE0 位, 选择在互补 PWM 模式中是否进行缓冲寄存器到暂存器的缓冲传送以及是否联动中断减少功能。

设定为抑止缓冲传送 (BTE1=0 并且 BTE0=1) 时的运行例子如图 21.70 所示。在此设定期间中, 不进行缓冲寄存器到暂存器的传送。

设定为缓冲传送联动中断减少功能 (BTE1=1 并且 BTE0=0) 时的运行例子如图 21.71 所示。此时, 除了缓冲传送允许期间以外, 不进行缓冲寄存器到暂存器的传送。

如果将定时器的中断减少设定寄存器 (TITCR) 的 T3AEN 位置“1”, 或者将 T4VEN 位置“1”, 或者将 T3AEN/T4VEN 位置“1”, 各缓冲传送允许期间就不同。TITCR 寄存器的 T3AEN 位、T4VEN 位的设定和缓冲传送允许期间的关系如图 21.72 所示。

注. 此功能必须和中断减少功能配合使用。

在禁止中断减少 (将定时器的中断减少设定寄存器 (TITCR) 的 T3AEN 位和 T4VEN 位置“0”, 或者将 TITCR 的减少次数设定位 (T3ACOR 和 T4VCOR) 置“0”) 时, 必须设定为缓冲传送不联动中断减少功能 (将定时器的缓冲传送设定寄存器 (TBTER) 的 BTE1 位置“0”)。

在禁止中断减少功能时, 如果设定为缓冲传送联动中断减少功能, 就不进行缓冲传送。

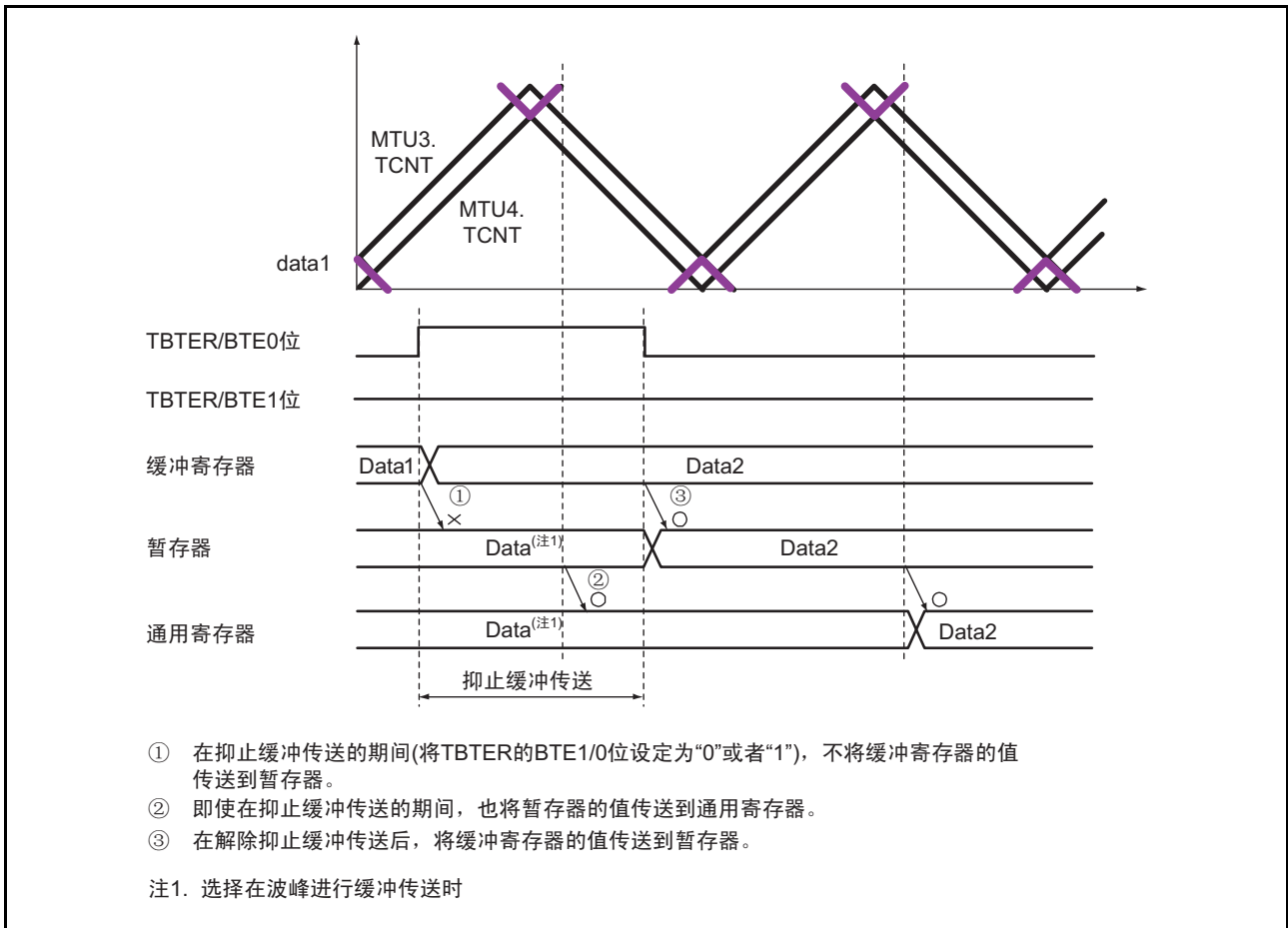


图 21.70 设定为抑制缓冲传送 (BTE1=0 并且 BTE0=1) 时的运行例子

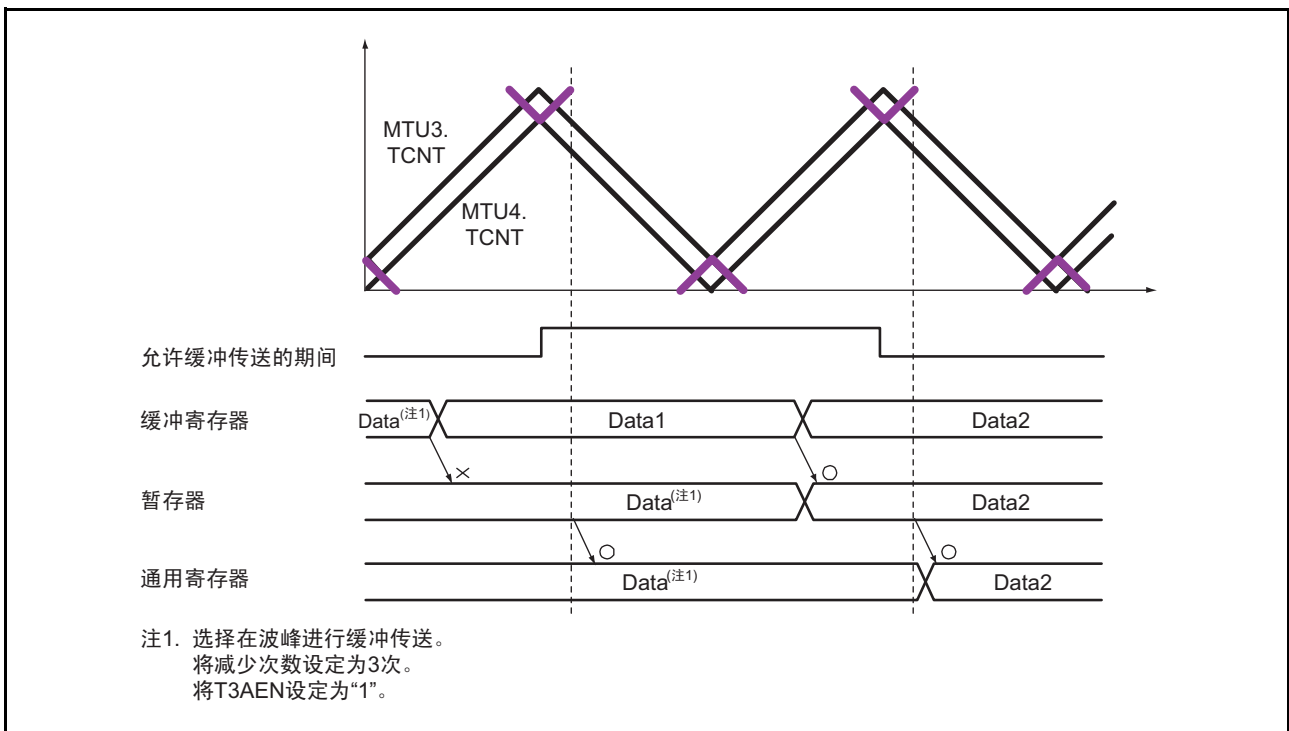


图 21.71 设定为缓冲传送联动中断减少功能 (BTE1=1 并且 BTE0=0) 时的运行例子

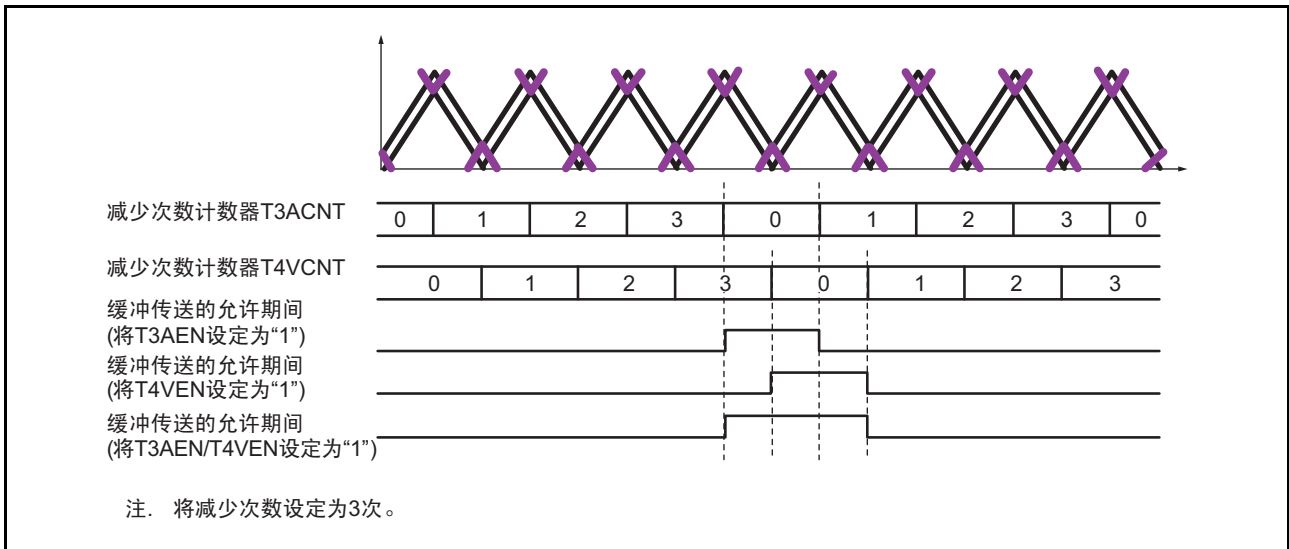


图 21.72 定时器的中断减少设定寄存器 (TITCR) 的 T3AEN 位、T4VEN 位的设定和缓冲传送允许期间的关系

(4) 互补 PWM 模式的输出保护功能

互补 PWM 模式的输出具有以下保护功能。

(a) 寄存器和计数器的误写防止功能

在互补 PWM 模式中使用的寄存器和计数器当中，除了随时能改写的缓冲寄存器以外，能通过设定定时器的读写允许寄存器 (TRWER) 的 RWE 位，选择允许或者禁止 CPU 存取模式寄存器、控制寄存器、比较寄存器和计数器。对象寄存器为 MTU3 和 MTU4 的一部分寄存器，适用于以下寄存器：

MTU3.TCR 和 MTU4.TCR、MTU3.TMDR 和 MTU4.TMDR、MTU3.TIORH 和 MTU4.TIORH、MTU3.TIORL 和 MTU4.TIORL、MTU3.TIER 和 MTU4.TIER、MTU3.TCNT 和 MTU4.TCNT、MTU3.TGRA 和 MTU4.TGRA、MTU3.TGRB 和 MTU4.TGRB、TOER、TOCR1、TOCR2、TGCR、TCDR、TDDR 共计 22 个寄存器。

通过此功能，能设定为禁止 CPU 存取模式寄存器、控制寄存器和计数器，以防止因 CPU 失控而产生的误写。在禁止存取的状态下对象寄存器的读取值为不定值，写操作无效。

(b) 通过外部信号进行的 PWM 输出停止功能

能通过输入指定的外部信号，使 6 相 PWM 输出引脚自动变为高阻抗状态。

详细内容请参照“22. 端口输出允许 2 (POE2a)”。

(c) 振荡停止时的 PWM 输出停止功能

在检测到输入到 RX210 的时钟停止后，6 相 PWM 输出引脚自动变为高阻抗状态。但是，一旦时钟重新开始振荡，就无法保证此引脚状态。

详细内容请参照“9.5 振荡停止检测功能”。

21.3.9 A/D 转换开始请求的延迟功能

能通过设定 MTU4 的定时器的 A/D 转换开始请求控制寄存器 (TADCR)、定时器的 A/D 启动请求周期寄存器 (MTU4.TADCORA 和 MTU4.TADCORB) 或者定时器的 A/D 启动请求周期缓冲寄存器 (MTU4.TADCOBRA 和 MTU4.TADCOBRB)，请求开始 A/D 转换。

A/D 转换开始请求的延迟功能是将 MTU4.TCNT 和 MTU4.TADCORA、MTU4.TADCORB 进行比较，如果 MTU4.TCNT 和 MTU4.TADCORA、MTU4.TADCORB 相同，就请求开始各自的 A/D 转换 (TRG4AN 和 TRG4BN)。

能通过设定 TADCR 的 ITA3AE、ITA4VE、ITB3AE、ITB4VE 位，联动中断减少功能减少 A/D 转换的开始请求 (TRG4AN 和 TRG4BN)。

(1) A/D 转换开始请求延迟功能的设定步骤例子

A/D 转换开始请求延迟功能的设定步骤例子如图 21.73 所示。

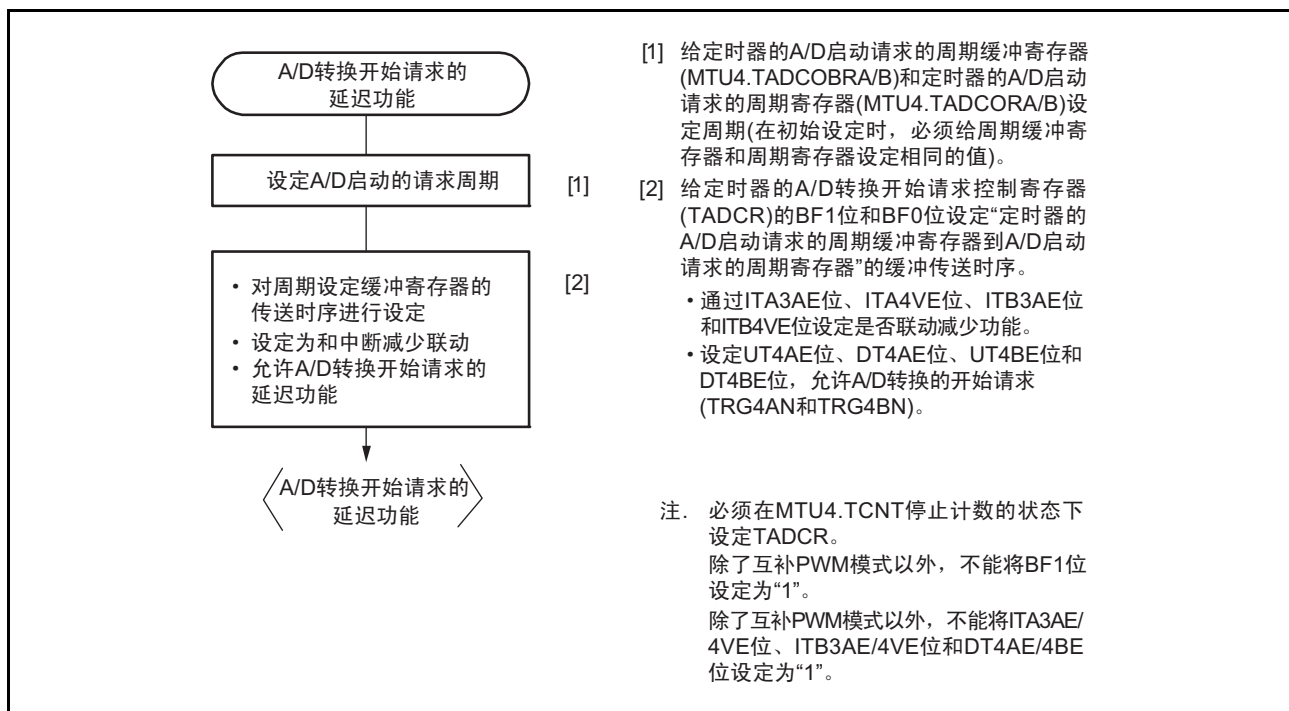


图 21.73 A/D 转换开始请求延迟功能的设定步骤例子

(2) A/D 转换开始请求延迟功能的基本运行例子

将缓冲传送时序设定为 MTU4.TCNT 的波谷并且在 MTU4.TCNT 进行递减计数时输出 A/D 转换开始请求信号 (TRG4AN) 时的 A/D 转换开始请求信号 (TRG4AN) 的基本运行例子如图 21.74 所示。

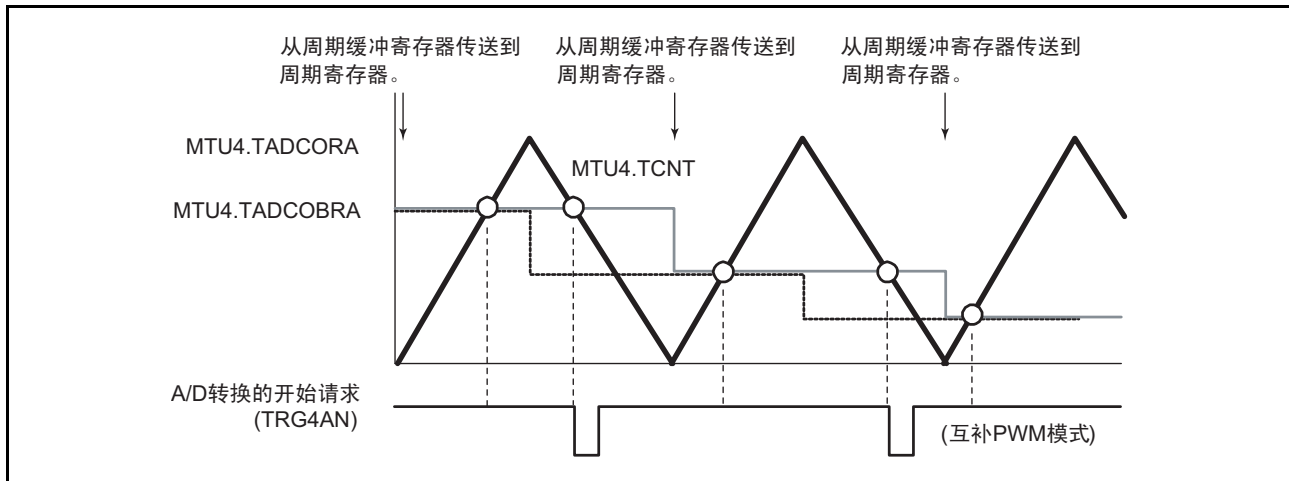


图 21.74 A/D 转换开始请求信号 (TRG4AN) 的基本运行例子

(3) 缓冲传送

通过将数据写到定时器的 A/D 启动请求周期设定缓冲寄存器 (MTU4.TADCOBRA/B)，更新定时器的 A/D 启动请求周期设定寄存器 (MTU4.TADCORA/B) 的数据。能通过设定定时器的 A/D 转换开始请求控制寄存器 (MTU4.TADCR) 的 BF1 位和 BF0 位，选择定时器 A/D 的启动请求周期设定缓冲寄存器到定时器的 A/D 启动请求周期设定寄存器的传送时序。

(4) 联动中断减少功能的 A/D 转换开始请求延迟功能

能通过设定过定时器的 A/D 转换开始请求控制寄存器 (TADCR) 的 ITA3AE、ITA4VE、ITB3AE、ITB4VE 位，联动中断减少功能请求开始 A/D 转换 (TRG4AN 和 TRG4BN)。在 MTU4.TCNT 进行递增计数和递减计数时允许 TRG4AN 输出并且联动中断减少功能时的 A/D 转换开始请求信号 (TRG4AN) 的运行例子如图 21.75 所示。

在 MTU4.TCNT 进行递增计数时，允许 TRG4AN 输出并且联动中断减少功能时的 A/D 转换开始请求信号 (TRG4AN) 的运行例子如图 21.76 所示。

注：此功能必须和中断减少功能配合使用。

在禁止中断减少 (将定时器的中断减少设定寄存器 (TITCR) 的 T3AEN 位和 T4VEN 位置“0”，或者将 TITCR 的减少次数设定位 (T3ACOR 和 T4VCOR) 置“0”) 时，必须设定为不联动中断减少功能 (将定时器的 A/D 转换开始请求控制寄存器 (TADCR) 的 ITA3AE、ITA4VE、ITB3AE 和 ITB4VE 位置“0”)。

必须注意：A/D 转换器的转换请求信号可能变为 TRG4ABN (TRG4AN 或者 TRG4BN)。

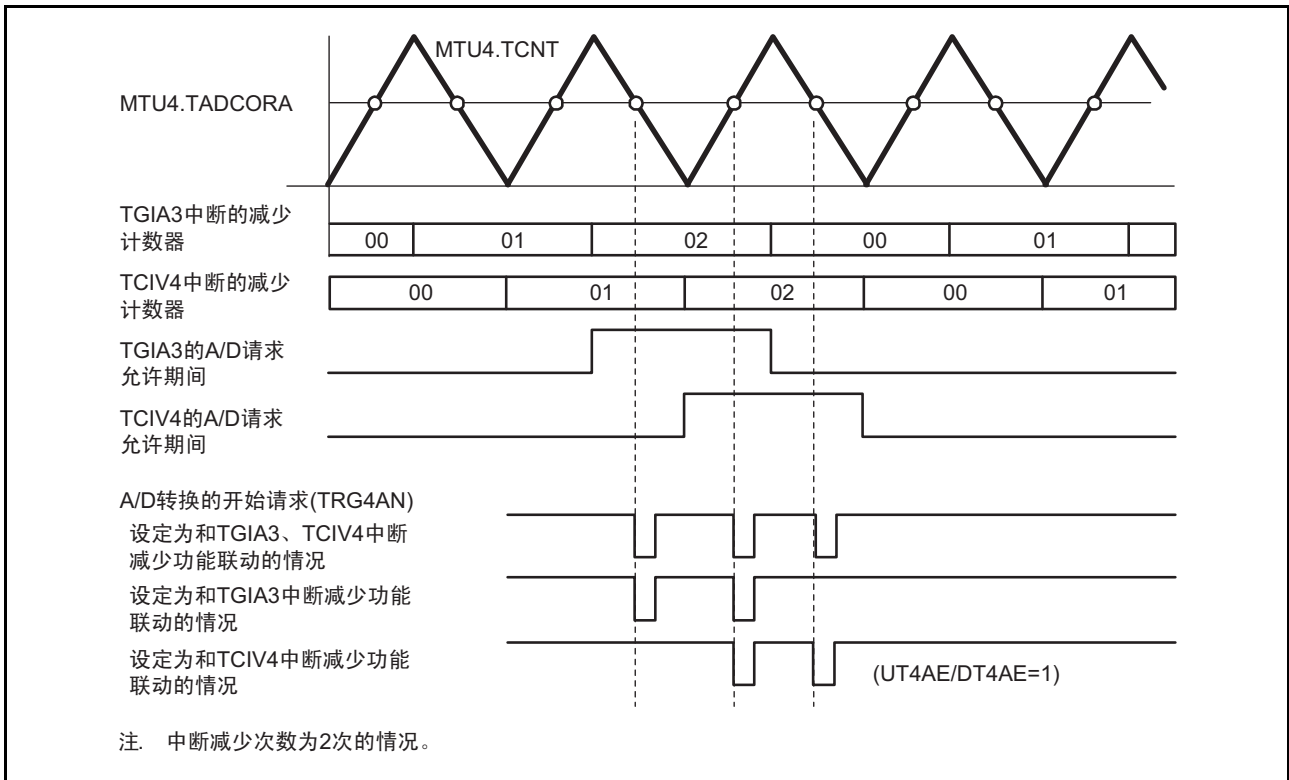


图 21.75 联动中断减少功能时的 A/D 转换开始请求信号 (TRG4AN) 的运行例子 (在 TCNT 进行递增计数以及递减计数时允许 TRG4AN 输出的情况)

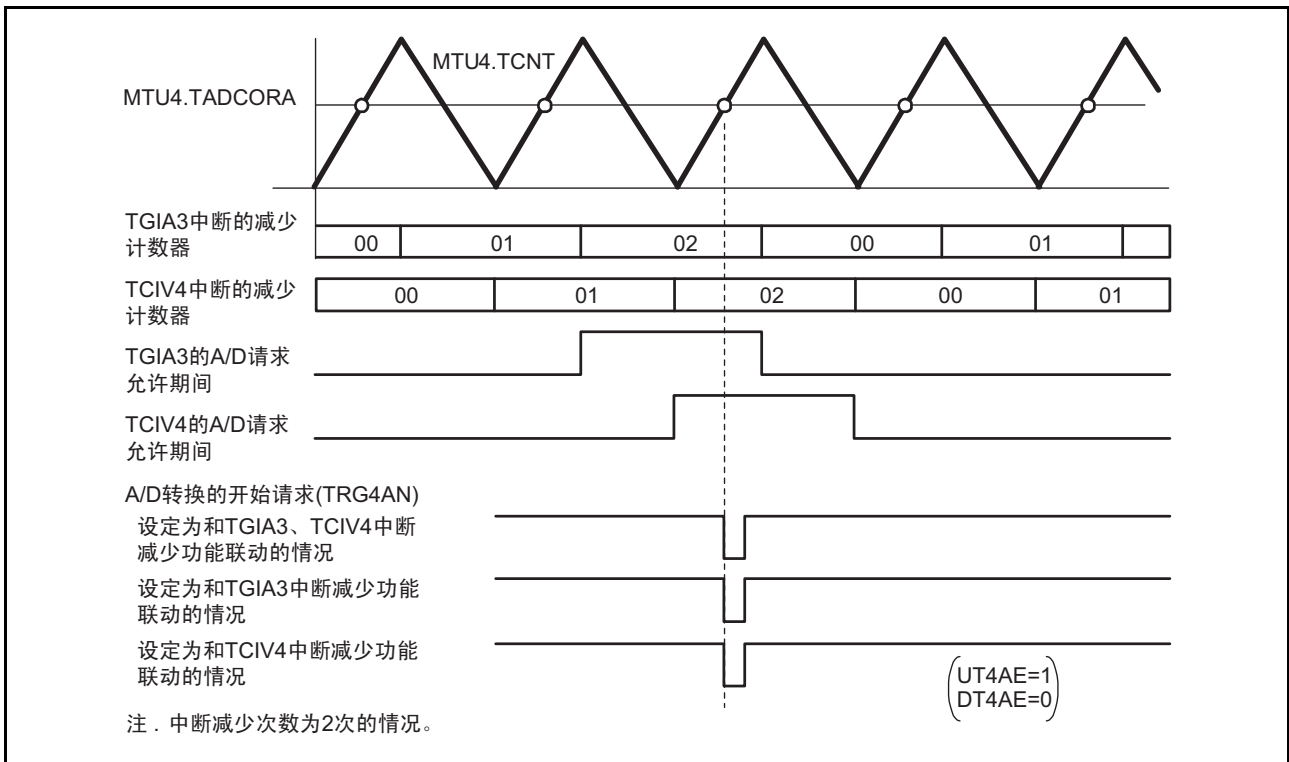


图 21.76 联动中断减少功能时的 A/D 转换开始请求信号 (TRG4AN) 的运行例子 (在 TCNT 进行递增计数时允许 TRG4AN 输出的情况)

21.3.10 外部脉宽的测量功能

MTU5 最多能测量 3 个外部脉宽。

(1) 测量外部脉宽的设定步骤例子

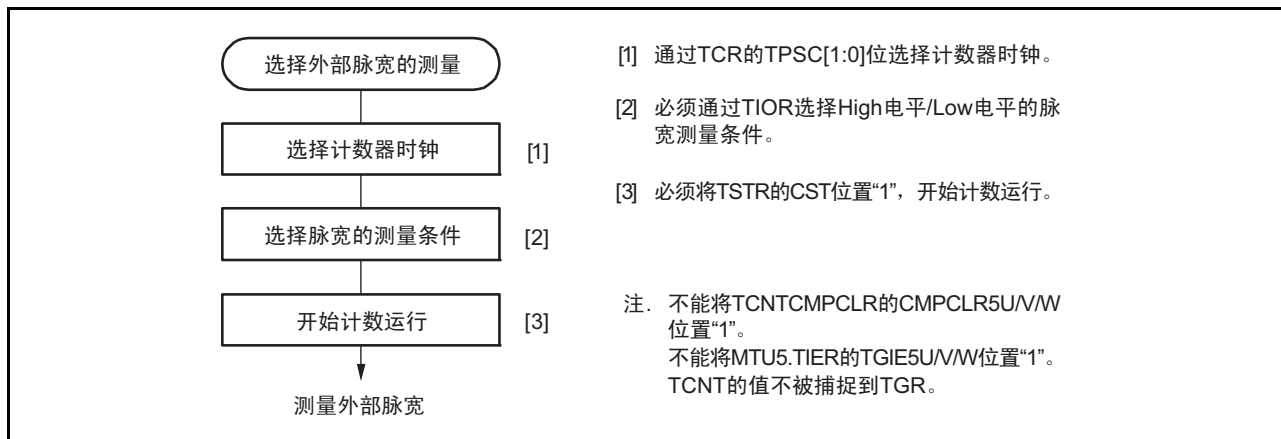


图 21.77 测量外部脉宽的设定步骤例子

(2) 测量外部脉宽的运行例子

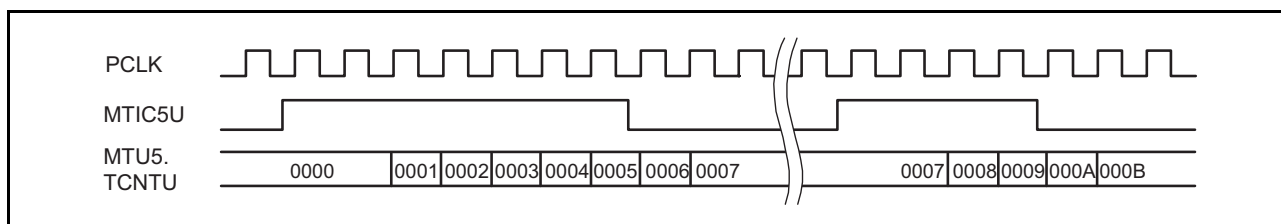


图 21.78 测量外部脉宽的运行例子 (测量 High 电平的宽度)

21.3.11 死区时间的补偿功能

能通过测量输出波形的延迟并且将此延迟反映到占空比, 将外部脉宽测量功能用作互补PWM运行时的PWM输出波形的死区时间补偿功能。

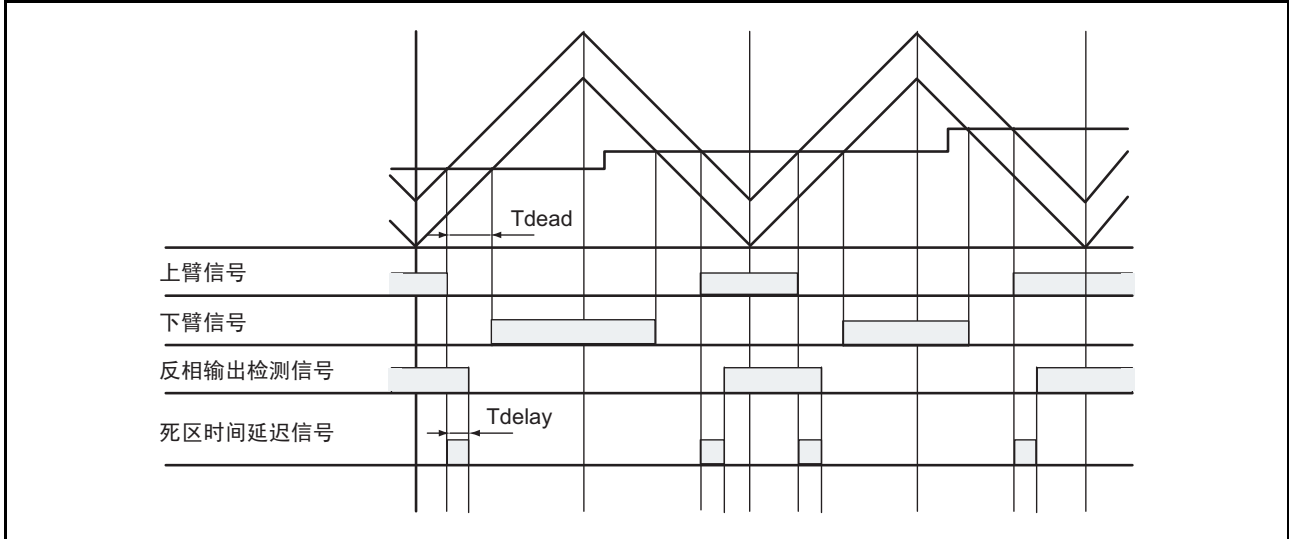


图 21.79 互补 PWM 运行时的死区时间延迟

(1) 死区时间补偿功能的设定步骤例子

使用 MTU5 的 3 个计数器的死区时间补偿功能的设定步骤例子如图 21.80 所示。

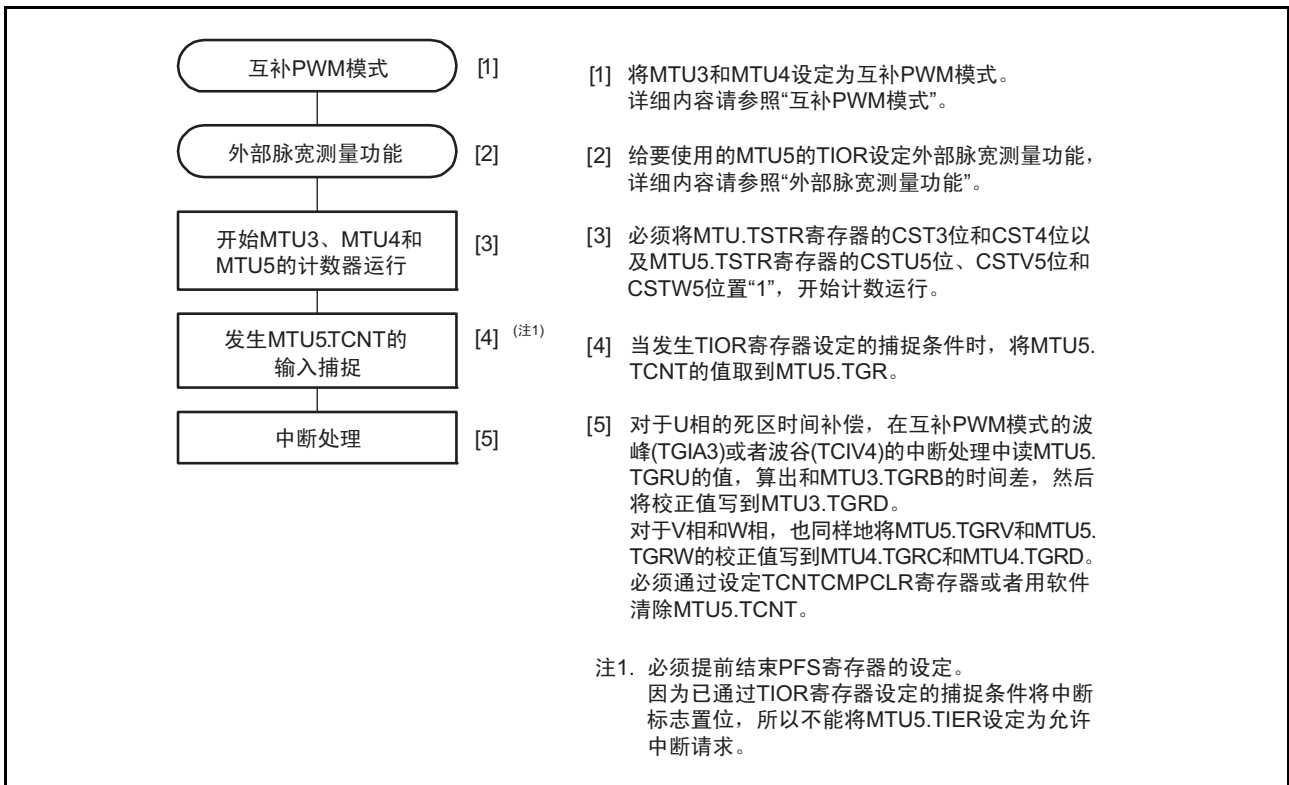


图 21.80 死区时间补偿功能的设定步骤例子

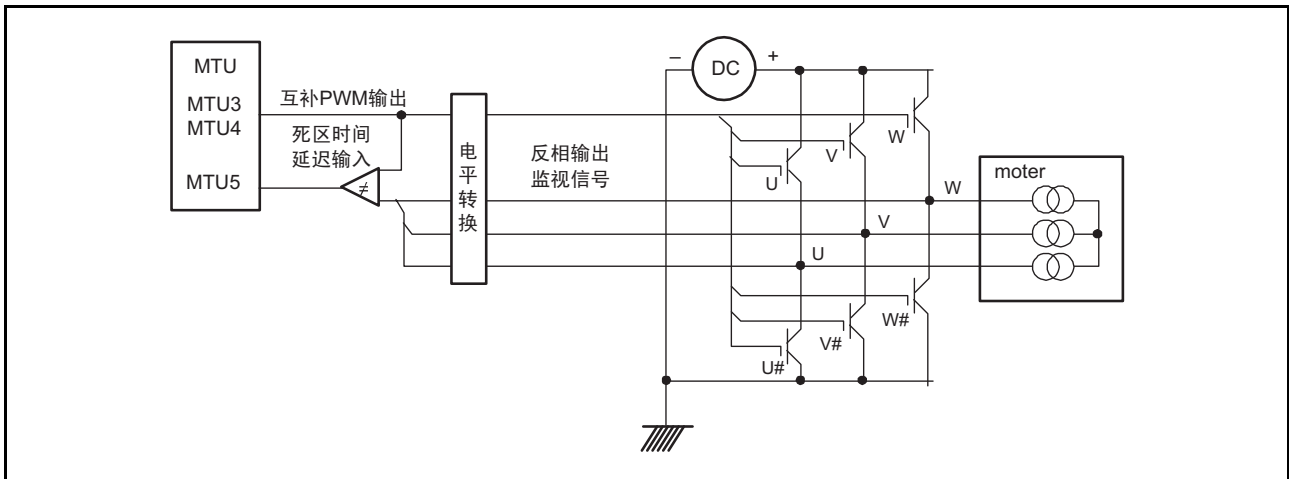


图 21.81 马达控制电路的结构例子

21.3.12 在互补 PWM 的“波峰 / 波谷”进行的 TCNT 捕捉运行

在互补 PWM 运行时，在“波峰、波谷、波峰和波谷”将 TCNT 的值保存到 TGR，通过 TIOR 寄存器选择要保存到 TGR 的时序转换。

TCNT 用作自由运行计数器（不被清除）并且在设定的“波峰、波谷”对 TGR 进行捕捉的运行例子如图 21.82 所示。

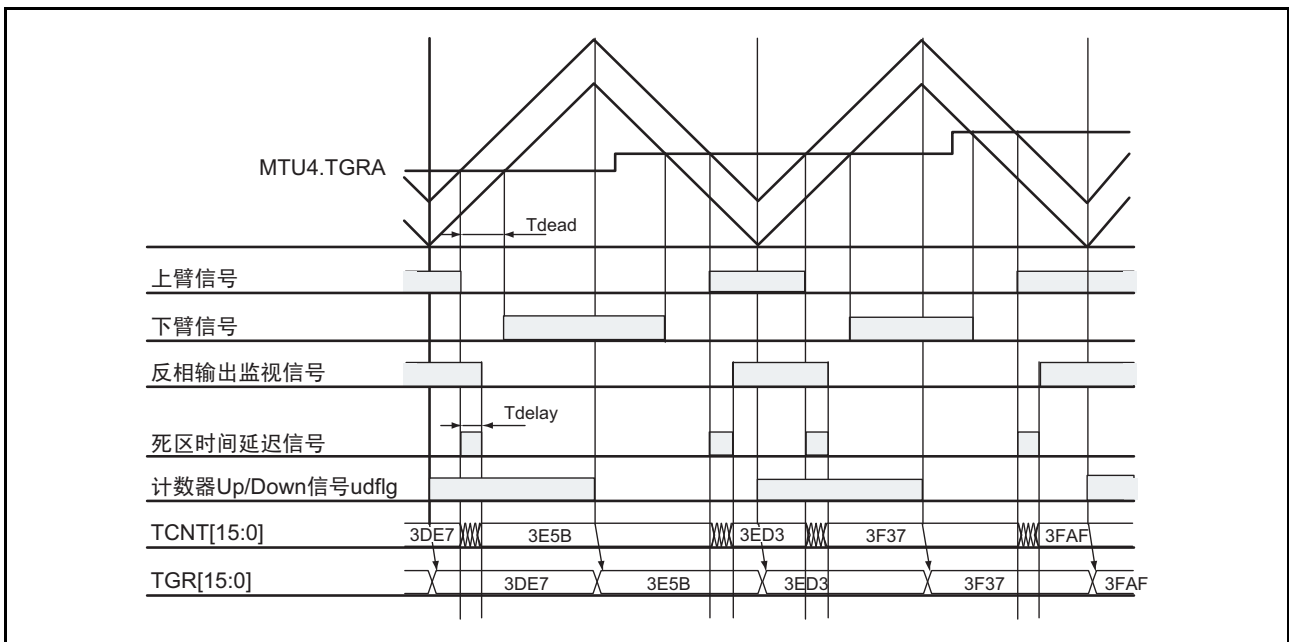


图 21.82 在互补 PWM 的“波峰 / 波谷”进行的 TCNT 捕捉运行

21.3.13 噪声滤波器功能

MTU 的输入捕捉的输入引脚或者外部脉冲输入引脚具有噪声滤波器功能。噪声滤波器功能按设定的采样周期对引脚进行采样，如果 3 次采样的引脚电平相同，就将相同的电平传送到内部并且持续到 3 次采样的引脚电平再次相同为止。

能按引脚设定允许或者停止噪声滤波器功能，并且能按通道设定采样时钟。噪声滤波器的时序如图 21.83 所示。

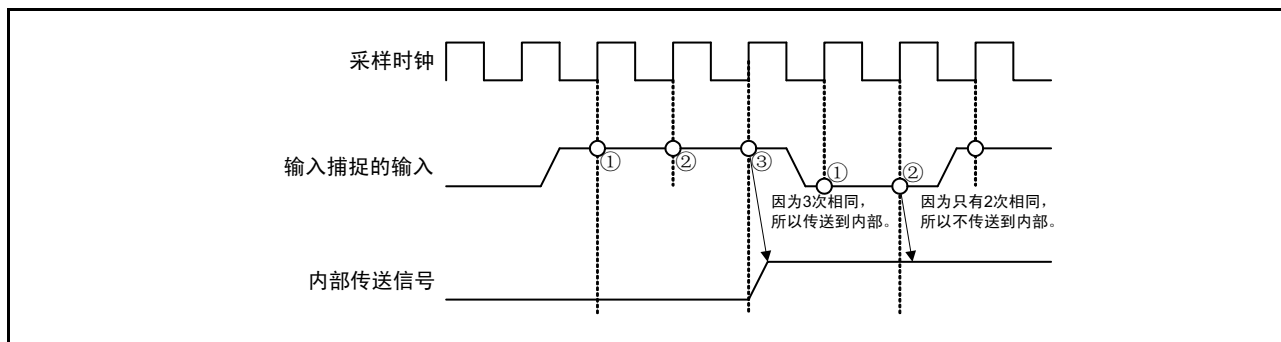


图 21.83 噪声滤波器的时序

21.4 中断源

21.4.1 中断源和优先级

MTU 的中断源有 TGR 的输入捕捉 / 比较匹配、TCNT 的上溢和 TCNT 的下溢共 3 种。因为各中断源有各自专用的状态标志和允许 / 禁止位，所以能独立允许或者禁止中断请求信号的产生。

如果发生中断源并且 TIER 寄存器对应的允许 / 禁止位为“1”，就请求中断。能通过中断控制器更改通道之间的优先级，但是通道内的优先级固定，详细内容请参照“14. 中断控制器 (ICUb)”。

MTU 的中断源一览表如表 21.57 所示。

表 21.57 MTU 中断源 (1)

通道	名称	中断源	DMAC 的启动	DTC 的启动	优先级
MTU0	TGIA0	MTU0.TGRA 的输入捕捉 / 比较匹配	能	能	高 ↑
	TGIB0	MTU0.TGRB 的输入捕捉 / 比较匹配	不能	能	
	TGIC0	MTU0.TGRC 的输入捕捉 / 比较匹配	不能	能	
	TGID0	MTU0.TGRD 的输入捕捉 / 比较匹配	不能	能	
	TCIV0	MTU0.TCNT 的上溢	不能	不能	
	TGIE0	MTU0.TGRE 的比较匹配	不能	不能	
	TGIF0	MTU0.TGRF 的比较匹配	不能	不能	
MTU1	TGIA1	MTU1.TGRA 的输入捕捉 / 比较匹配	能	能	↑ 低
	TGIB1	MTU1.TGRB 的输入捕捉 / 比较匹配	不能	能	
	TCIV1	MTU1.TCNT 的上溢	不能	不能	
	TCIU1	MTU1.TCNT 的下溢	不能	不能	
MTU2	TGIA2	MTU2.TGRA 的输入捕捉 / 比较匹配	能	能	
	TGIB2	MTU2.TGRB 的输入捕捉 / 比较匹配	不能	能	
	TCIV2	MTU2.TCNT 的上溢	不能	不能	
	TCIU2	MTU2.TCNT 的下溢	不能	不能	
MTU3	TGIA3	MTU3.TGRA 的输入捕捉 / 比较匹配	能	能	
	TGIB3	MTU3.TGRB 的输入捕捉 / 比较匹配	不能	能	
	TGIC3	MTU3.TGRC 的输入捕捉 / 比较匹配	不能	能	
	TGID3	MTU3.TGRD 的输入捕捉 / 比较匹配	不能	能	
	TCIV3	MTU3.TCNT 的上溢	不能	不能	
MTU4	TGIA4	MTU4.TGRA 的输入捕捉 / 比较匹配	能	能	
	TGIB4	MTU4.TGRB 的输入捕捉 / 比较匹配	不能	能	
	TGIC4	MTU4.TGRC 的输入捕捉 / 比较匹配	不能	能	
	TGID4	MTU4.TGRD 的输入捕捉 / 比较匹配	不能	能	
	TCIV4	MTU4.TCNT 的上溢 / 下溢	不能	能	
MTU5	TGIU5	MTU5.TGRU 的输入捕捉 / 比较匹配	不能	能	
	TGIV5	MTU5.TGRV 的输入捕捉 / 比较匹配	不能	能	
	TGIW5	MTU5.TGRW 的输入捕捉 / 比较匹配	不能	能	

注. 表示复位后的初始状态。能通过中断控制器更改通道之间的优先级。

(1) 输入捕捉 / 比较匹配中断

如果在各通道的 TGR 发生输入捕捉 / 比较匹配时 TIER 寄存器的 TGIE 位为“1”，就请求中断。在 MTU 中，MTU0 有 6 个，MTU3 和 MTU4 各有 4 个，MTU1 和 MTU2 各有 2 个，MTU5 有 3 个，共计 21 个的输入捕捉 / 比较匹配中断。

(2) 上溢中断

如果在各通道的 TCNT 发生上溢时 TIER 寄存器的 TCIEV 位为“1”，就请求中断。在 MTU 中，各通道有 1 个，共计 5 个上溢中断。

(3) 下溢中断

如果在各通道的 TCNT 发生下溢时 TIER 寄存器的 TCIEU 位为“1”，就请求中断。在 MTU 中，MTU1 和 MTU2 各有 1 个，共计 2 个下溢中断。

21.4.2 DTC/DMAC 的启动

(1) DTC 的启动

能通过各通道 TGR 的输入捕捉 / 比较匹配中断或者 MTU4 的上溢中断来启动 DTC，详细内容请参照“17. 数据传送控制器 (DTCa) ”。

在 MTU 中，MTU0 和 MTU3 各有 4 个，MTU1 和 MTU2 各有 2 个，MTU4 有 5 个，MTU5 有 3 个，共计 20 个输入捕捉 / 比较匹配中断和上溢中断，能将这输入捕捉 / 比较匹配中断和上溢中断作为 DTC 的启动源。

(2) DMAC 的启动

能通过各通道 TGRA 的输入捕捉 / 比较匹配中断来启动 DMAC，详细内容请参照“16. DMA 控制器 (DMACA) ”。

在 MTU 中，MTU0 ~ MTU4 各有 1 个，共计 5 个 TGRA 寄存器的输入捕捉 / 比较匹配中断，能将这输入捕捉 / 比较匹配中断作为 DMAC 的启动源。

当通过 MTU 启动 DMAC 时，在 DMAC 请求内部总线权时清除启动源。因此根据内部总线的状态，即使清除启动源，也会产生 DMAC 传送进入开始等待状态的期间。

21.4.3 A/D 转换器的启动

在 MTU 中，能通过以下 5 种方法启动 A/D 转换器。

各中断源和 A/D 转换开始请求的对应如表 21.58 所示。

(1) 在 TGRA 的输入捕捉 / 比较匹配时以及在互补 PWM 模式的 MTU4.TCNT 波谷进行的 A/D 启动

能通过各通道 TGRA 的输入捕捉 / 比较匹配来启动 A/D 转换器。另外，如果在将 MTU4.TIER 的 TTGE2 位置“1”的状态下进行互补 PWM 运行，就能在 MTU4.TCNT 为波谷 (MTU4.TCNT=0000h) 时启动 A/D 转换器。

在以下所示的条件下，对 A/D 转换器产生 A/D 转换开始请求 TRGAN。

- 在发生各通道的 TGRA 输入捕捉 / 比较匹配的情况下 TIER 的 TTGE 位为“1”时
- 在将 MTU4.TIER 的 TTGE2 位置“1”的状态下进行互补 PWM 运行并且 MTU4.TCNT 变为波谷时 (MTU4.TCNT=0000h)

在上述条件下，如果已在 A/D 转换器侧选择 MTU 的转换开始触发 TRGAN，就开始进行 A/D 转换。

(2) 通过 MTU0.TCNT 和 MTU0.TGRE 的比较匹配进行的 A/D 转换器启动

能通过 MTU0.TCNT 和 MTU0.TGRE 的比较匹配来启动 A/D 转换器。

能通过 MTU0.TCNT 和 MTU0.TGRE 的比较匹配，产生 A/D 转换开始请求 TRG0EN。此时，如果已在 A/D 转换器侧选择 MTU 的转换开始触发 TRG0EN，就开始进行 A/D 转换。

(3) 通过 MTU0.TCNT 和 MTU0.TGRF 的比较匹配进行的 A/D 转换器启动

能通过 MTU0.TCNT 和 MTU0.TGRF 的比较匹配来启动 A/D 转换器。

能通过 MTU0.TCNT 和 MTU0.TGRF 的比较匹配，产生 A/D 转换开始请求 TRG0FN。此时，如果已在 A/D 转换器侧选择 MTU 的转换开始触发 TRG0FN，就开始进行 A/D 转换。

(4) 通过 MTU0.TGRA 或者 MTU0.TGRB 的输入捕捉 / 比较匹配进行的 A/D 转换器启动

能通过 MTU0.TCNT 和 MTU0.TGRA 或者 MTU0.TGRB 的输入捕捉 / 比较匹配，产生 A/D 转换开始请求 TRG0AN 或者 TRG0BN。此时，如果已在 A/D 转换器侧选择 MTU 的转换开始触发 TRG0AN 或者 TRG0BN，就开始进行 A/D 转换。

(5) 通过 A/D 转换开始请求的延迟功能进行的 A/D 转换器启动

如果给 A/D 转换开始请求控制寄存器 (TADCR) 的 UT4AE、DT4AE、UT4BE、DT4BE 位写“1”，就能在 TADCORA、TADCORB 和 MTU4.TCNT 相同时产生 TRG4AN 或者 TRG4BN，启动 A/D 转换器。详细内容请参照“21.3.9 A/D 转换开始请求的延迟功能”。

如果发生 TRG4AN 或者 TRG4BN，就产生 TRG4ABN。如果已在 A/D 转换器侧选择 MTU 的转换开始触发 TRG4ABN，就开始进行 A/D 转换。

表 21.58 各中断源和 A/D 转换开始请求的对应

对象	A/D 转换器的启动源	A/D 转换开始请求
MTU0.TGRA 和 MTU0.TCNT	输入捕捉 / 比较匹配	TRGAN
MTU1.TGRA 和 MTU1.TCNT		
MTU2.TGRA 和 MTU2.TCNT		
MTU3.TGRA 和 MTU3.TCNT		
MTU4.TGRA 和 MTU4.TCNT		
MTU4.TCNT	互补 PWM 模式的 MTU4.TCNT 波谷	
MTU0.TGRA 和 MTU0.TCNT	输入捕捉 / 比较匹配	TRG0AN
MTU0.TGRB 和 MTU0.TCNT		TRG0BN
MTU0.TGRE 和 MTU0.TCNT	比较匹配	TRG0EN
MTU0.TGRF 和 MTU0.TCNT		TRG0FN
TADCORA 和 MTU4.TCNT 或者 TADCORB 和 MTU4.TCNT		TRG4ABN

21.5 运行时序

21.5.1 输入 / 输出时序

(1) TCNT 的计数时序

内部时钟运行时的 TGI 中断的计数时序如图 21.84 和图 21.85 所示，外部时钟运行（正常模式）和外部时钟运行（相位计数模式）时的 TCNT 计数时序分别如图 21.86 和图 21.87 所示。

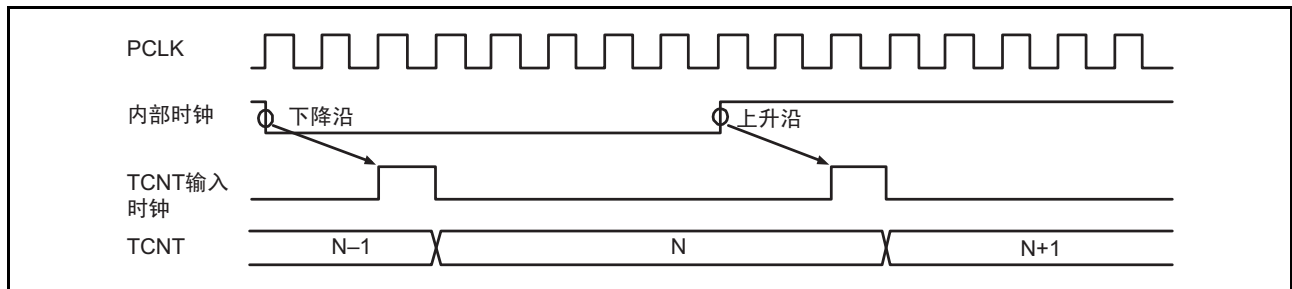


图 21.84 内部时钟运行时的计数时序 (MTU0 ~ MTU4)

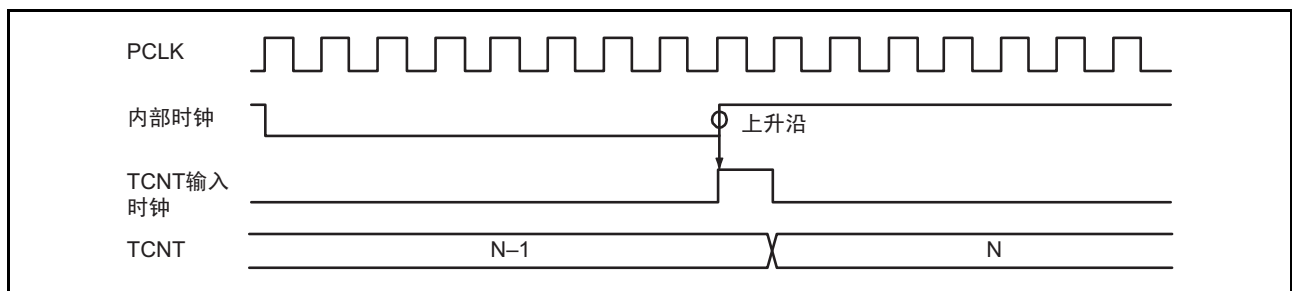


图 21.85 内部时钟运行时的计数时序 (MTU5)

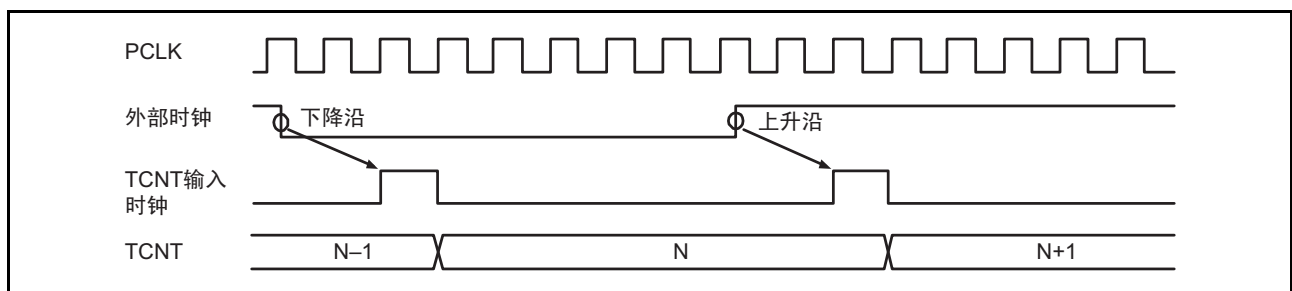


图 21.86 外部时钟运行时的计数时序 (MTU0 ~ MTU4)

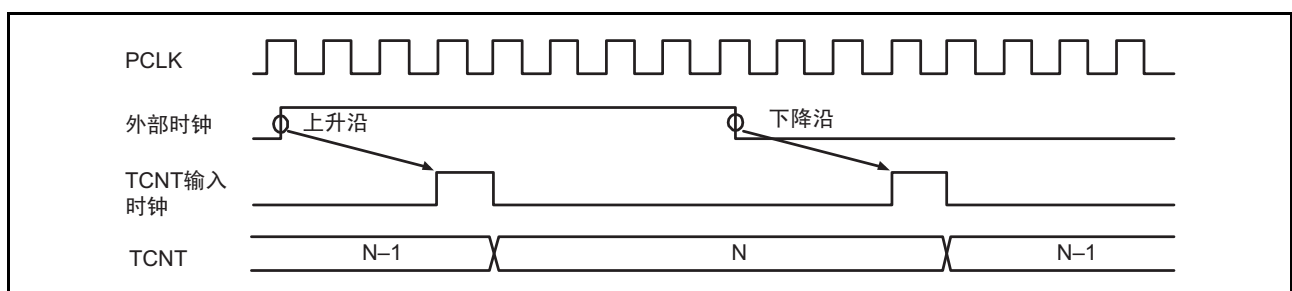


图 21.87 外部时钟运行时的计数时序 (相位计数模式)

(2) 输出比较的输出时序

在 TCNT 和 TGR 的值相同的最后状态（在更新 TCNT 相同后的计数值时）产生比较匹配信号。当产生比较匹配信号时，将 TIOR 寄存器设定的输出值输出到输出比较的输出引脚（MTIOC 引脚）。在 TCNT 和 TGR 的值相同后到产生 TCNT 输入时钟前，不产生比较匹配信号。

输出比较的输出时序（正常模式、PWM 模式）和输出比较的输出时序（互补 PWM 模式、复位同步 PWM 模式）分别如图 21.88 和图 21.89 所示。

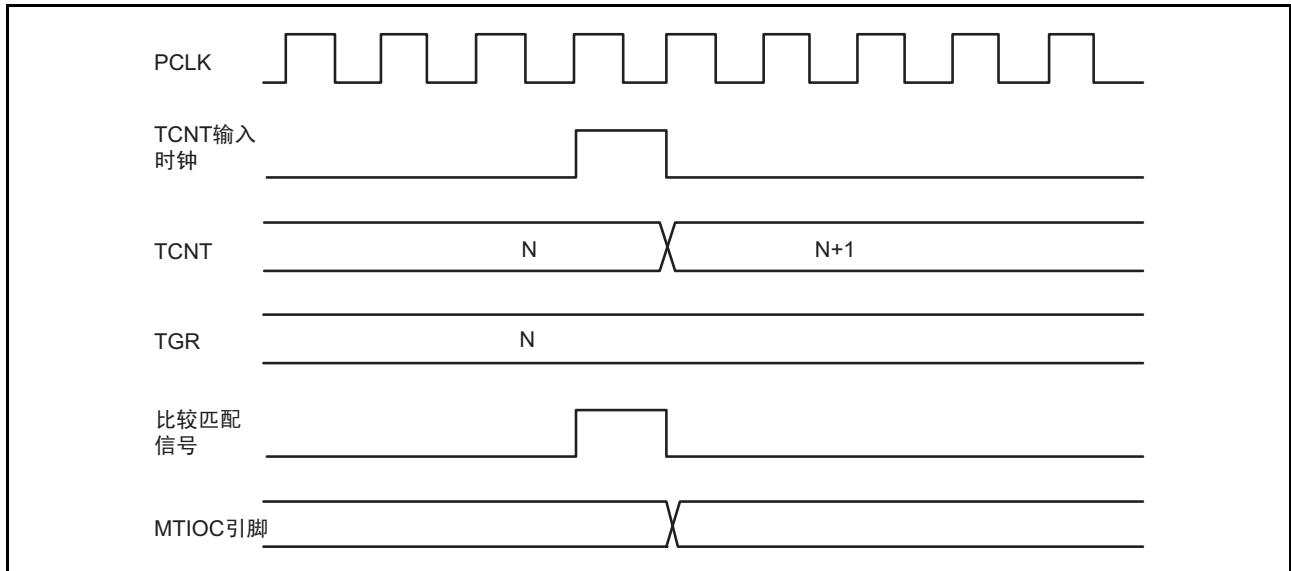


图 21.88 输出比较的输出时序（正常模式、PWM 模式）

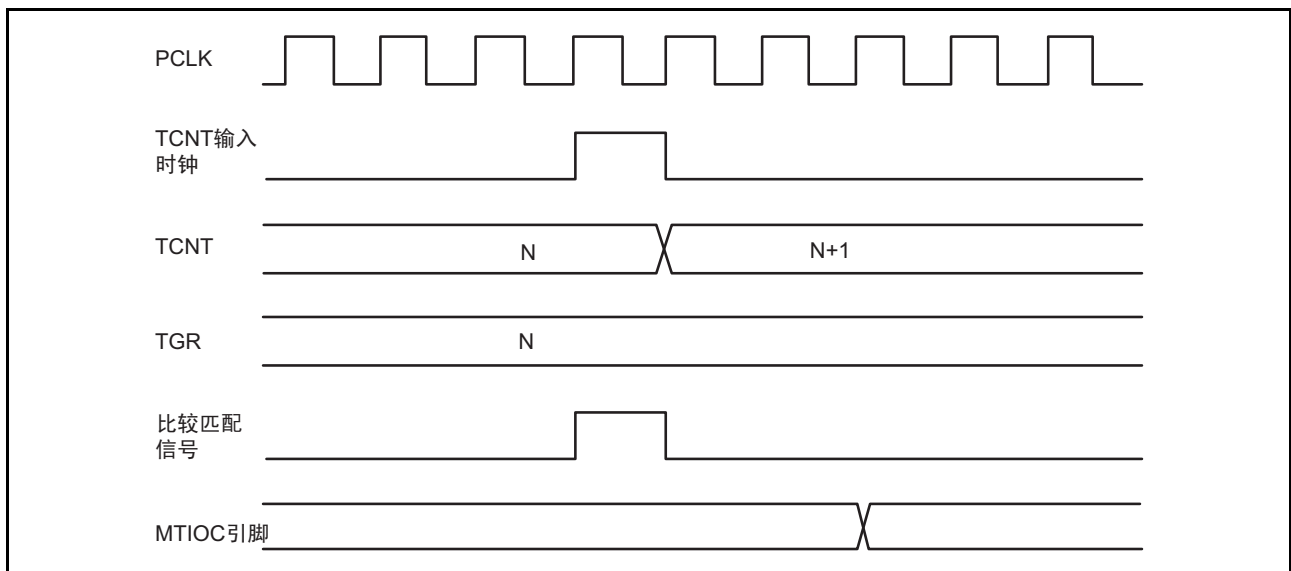


图 21.89 输出比较的输出时序（互补 PWM 模式、复位同步 PWM 模式）

(3) 输入捕捉信号的时序

输入捕捉的时序如图 21.90 所示。

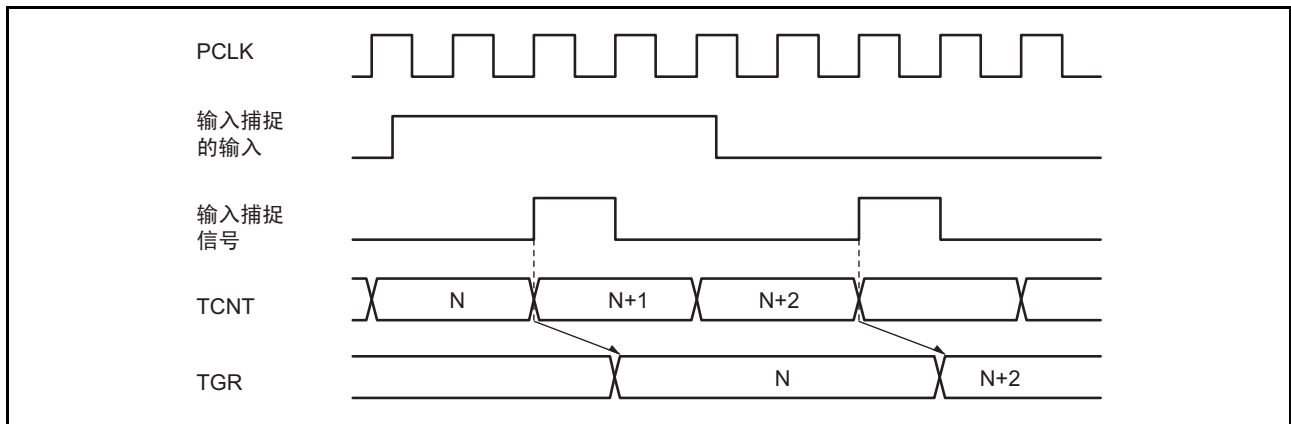


图 21.90 输入捕捉的输入信号时序

(4) 通过比较匹配 / 输入捕捉进行的计数器清除时序

指定通过比较匹配进行计数器清除时的时序如图 21.91 和图 21.92 所示，指定通过输入捕捉进行计数器清除时的时序如图 21.93 所示。

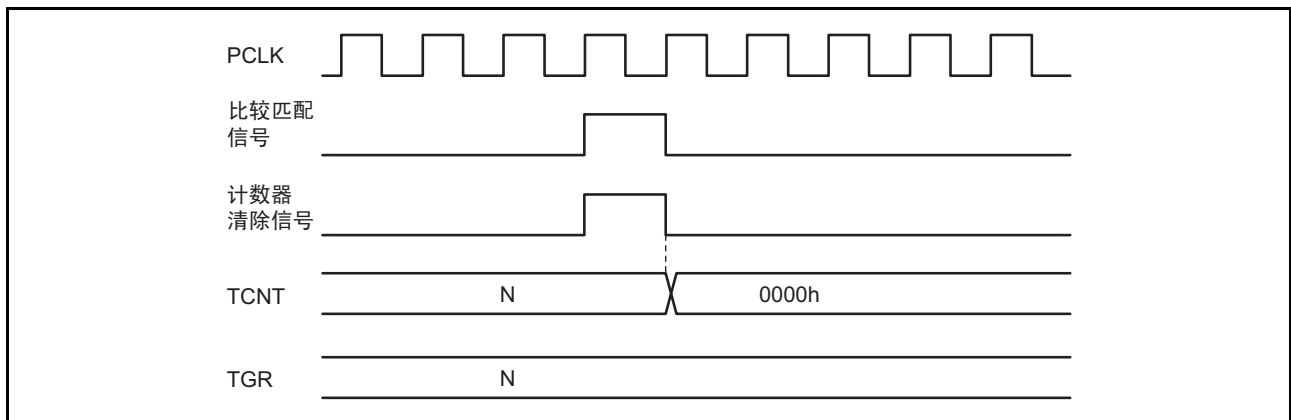


图 21.91 计数器清除的时序 (比较匹配) (MTU0 ~ MTU4)

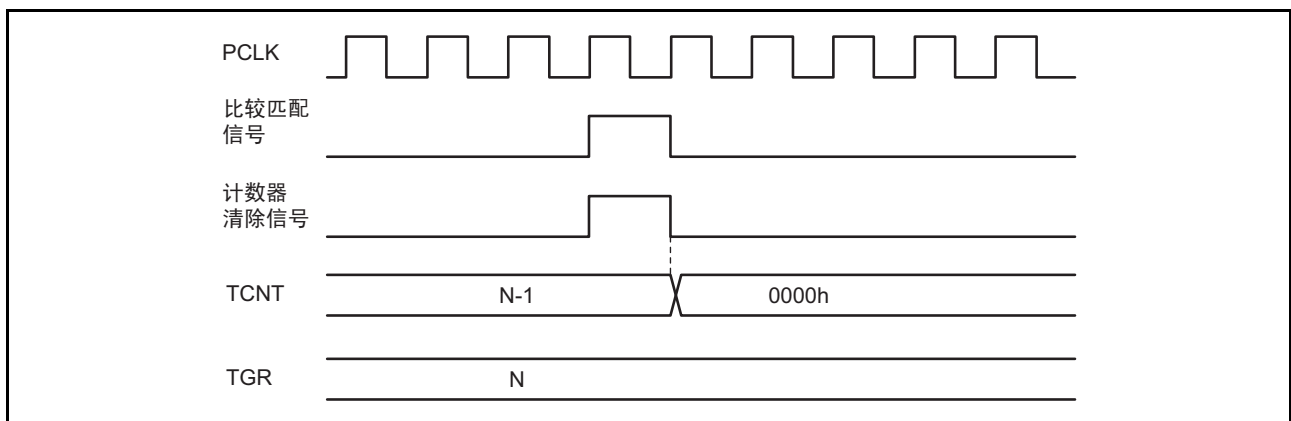


图 21.92 计数器清除的时序 (比较匹配) (MTU5)

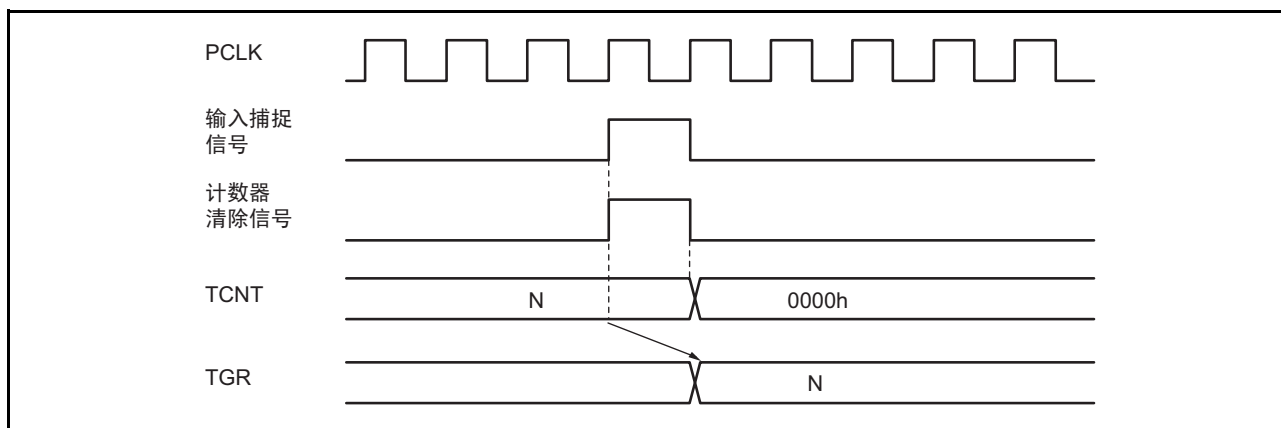


图 21.93 计数器清除的时序 (输入捕捉) (MTU0 ~ MTU5)

(5) 缓冲运行的时序

缓冲运行的时序如图 21.94 ~图 21.96 所示。

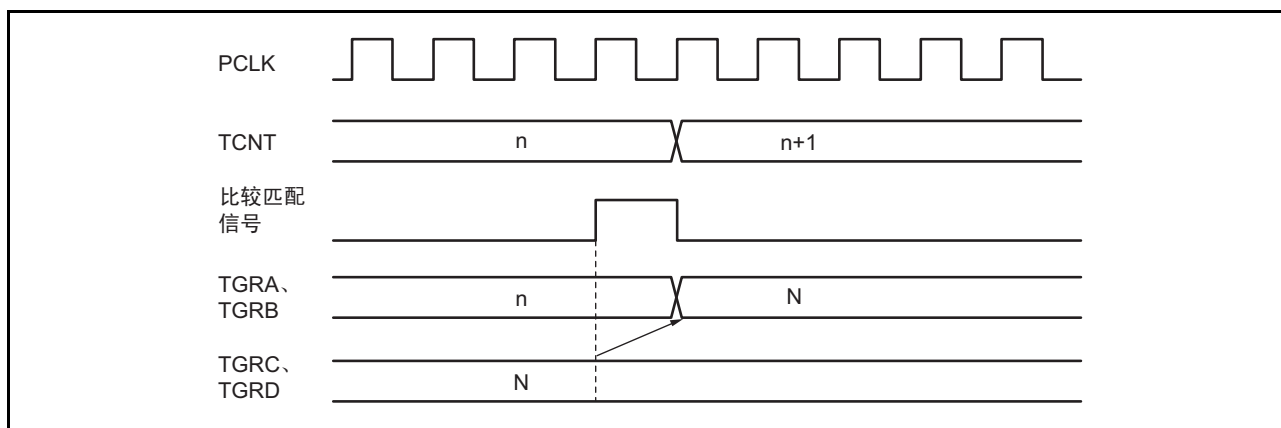


图 21.94 缓冲运行的时序 (比较匹配)

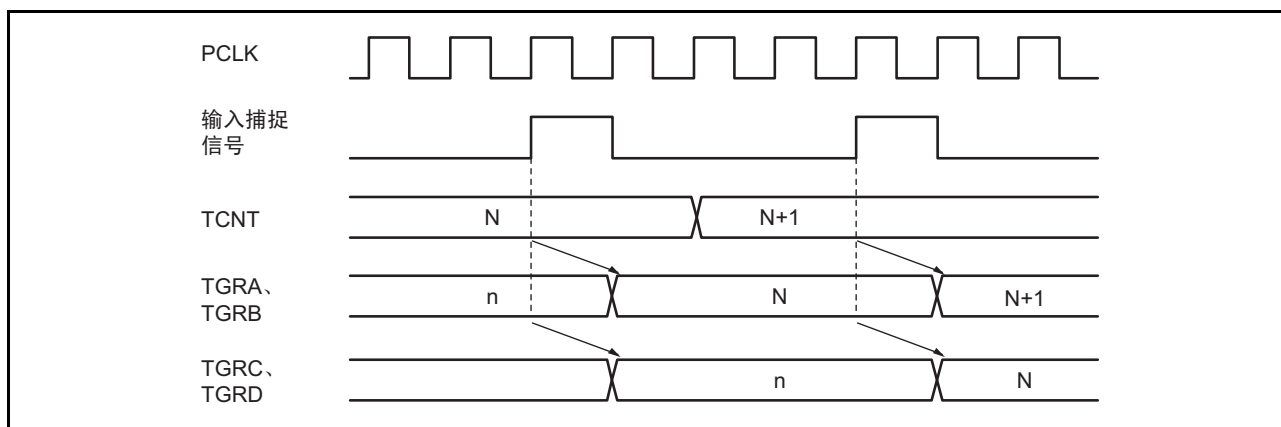


图 21.95 缓冲运行的时序 (输入捕捉)

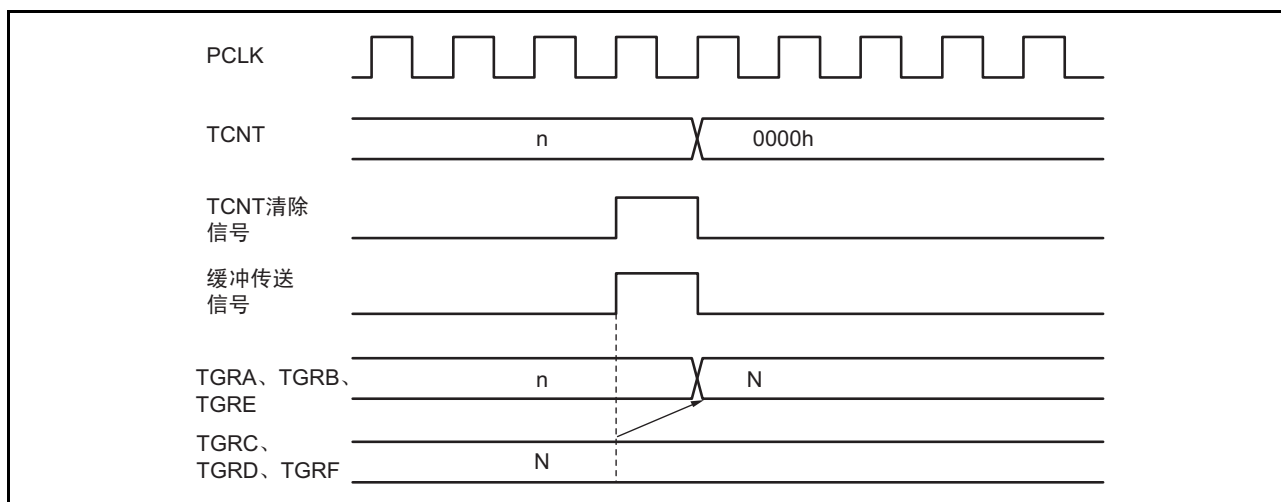


图 21.96 缓冲运行的时序 (清除 TCNT 的情况)

(6) 缓冲传送的时序 (互补 PWM 模式)

互补 PWM 模式的缓冲传送时序如图 21.97 ~ 图 21.99 所示。

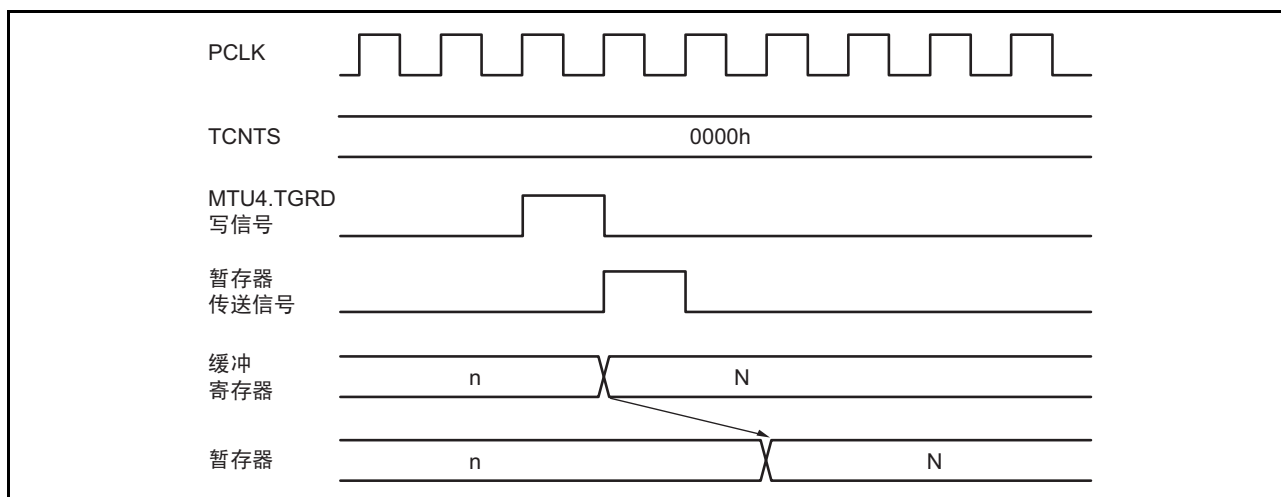


图 21.97 缓冲寄存器到暂存器的传送时序 (TCNTS 停止)

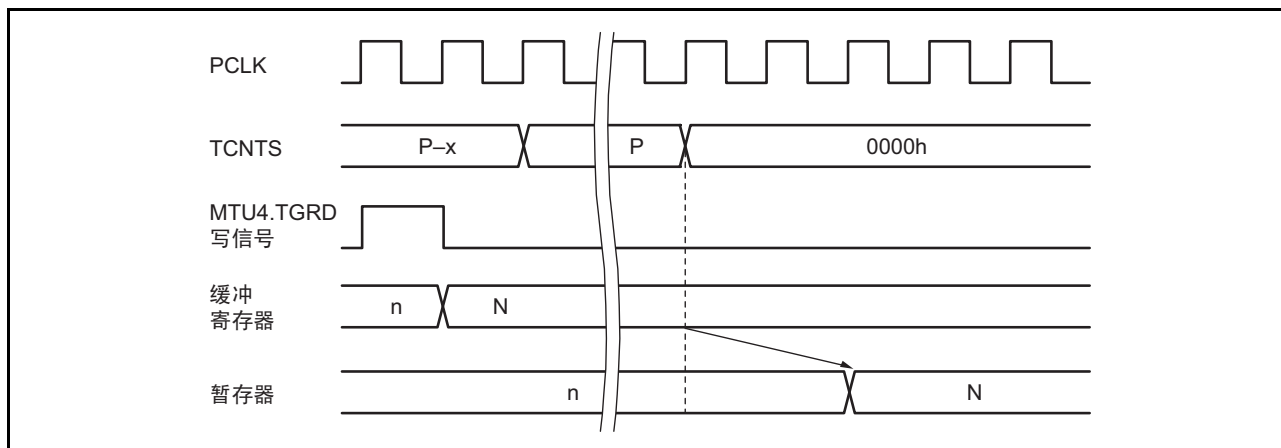


图 21.98 缓冲寄存器到暂存器的传送时序 (TCNTS 正在运行)

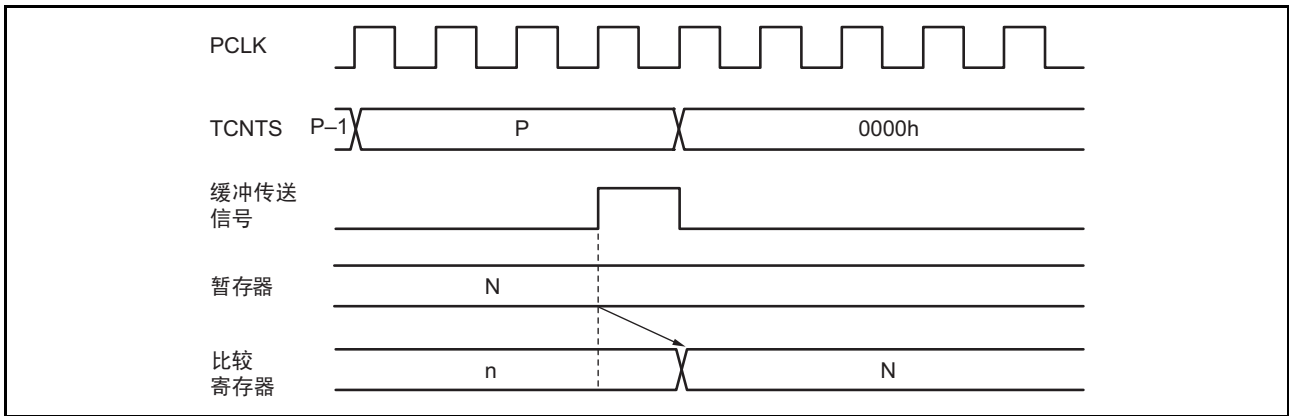


图 21.99 暂存器到比较寄存器的传送时序

21.5.2 中断信号的时序

(1) 比较匹配时的 TGI 中断的置位时序

由比较匹配产生的 TGI 中断请求信号的时序如图 21.100 和图 21.101 所示。

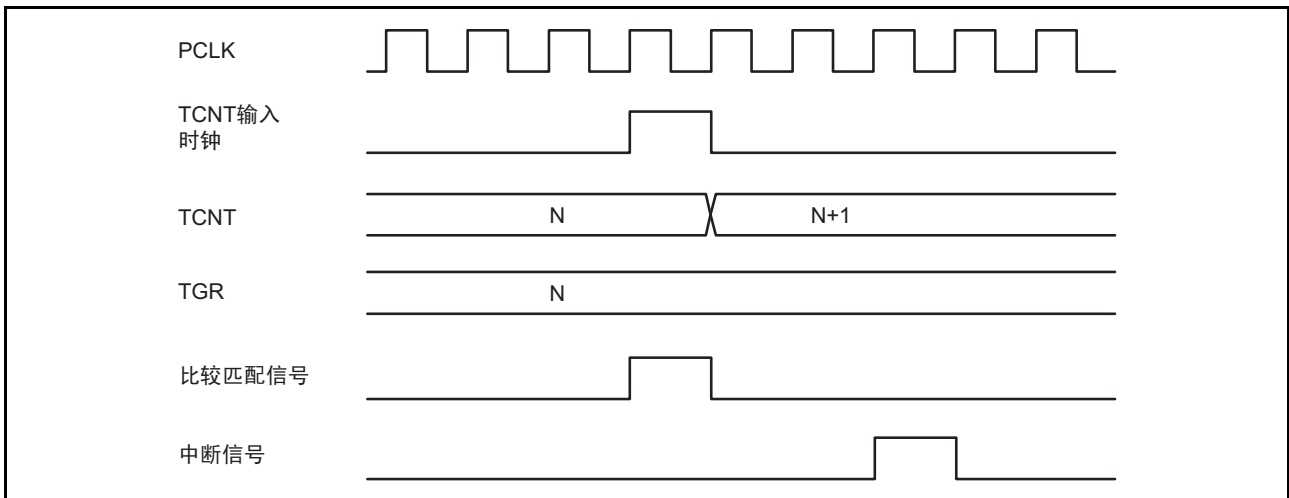


图 21.100 TGI 中断时序 (比较匹配) (MTU0 ~ MTU4)

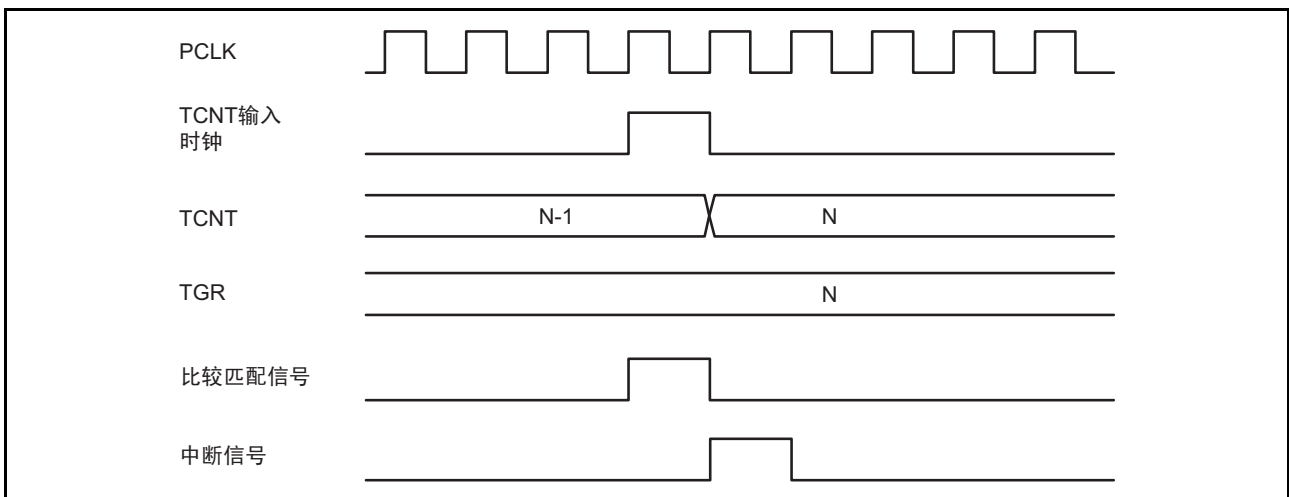


图 21.101 TGI 中断时序 (比较匹配) (MTU5)

(2) 输入捕捉时的 TGI 中断的置位时序

由输入捕捉产生的 TGI 中断请求信号的时序如图 21.102 和图 21.103 所示。

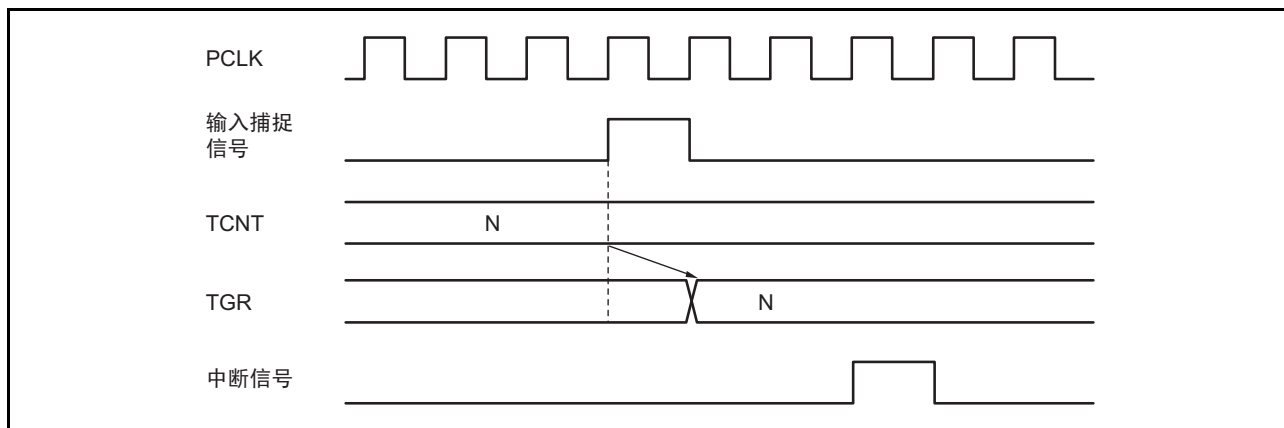


图 21.102 TGI 中断的时序 (输入捕捉) (MTU0 ~ MTU4)

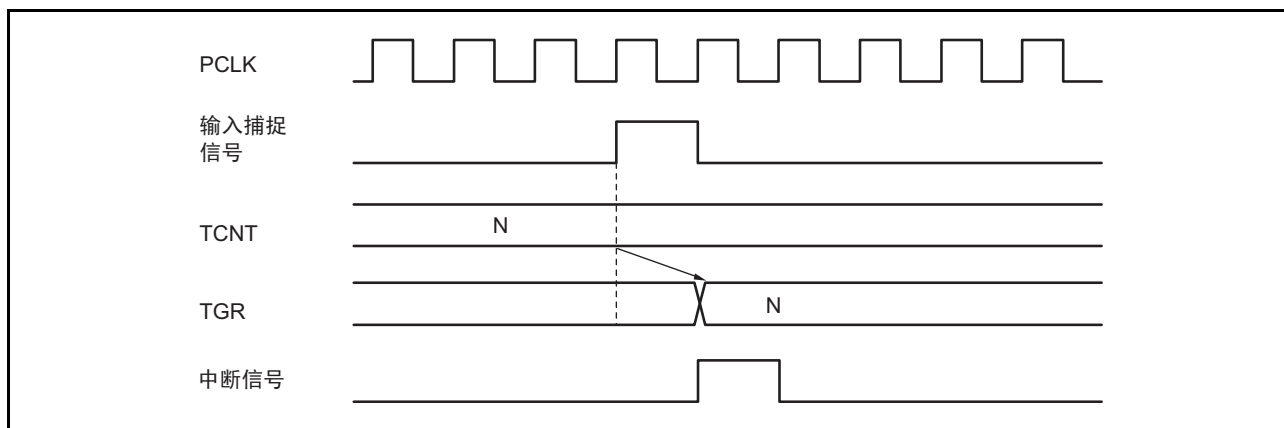


图 21.103 TGI 中断的时序 (输入捕捉) (MTU5)

(3) TCIV/TCIU 中断标志的置位时序

由上溢产生的 TCIV 中断请求信号的时序如图 21.104 所示。

由下溢产生的 TCIU 中断请求信号的时序如图 21.105 所示。

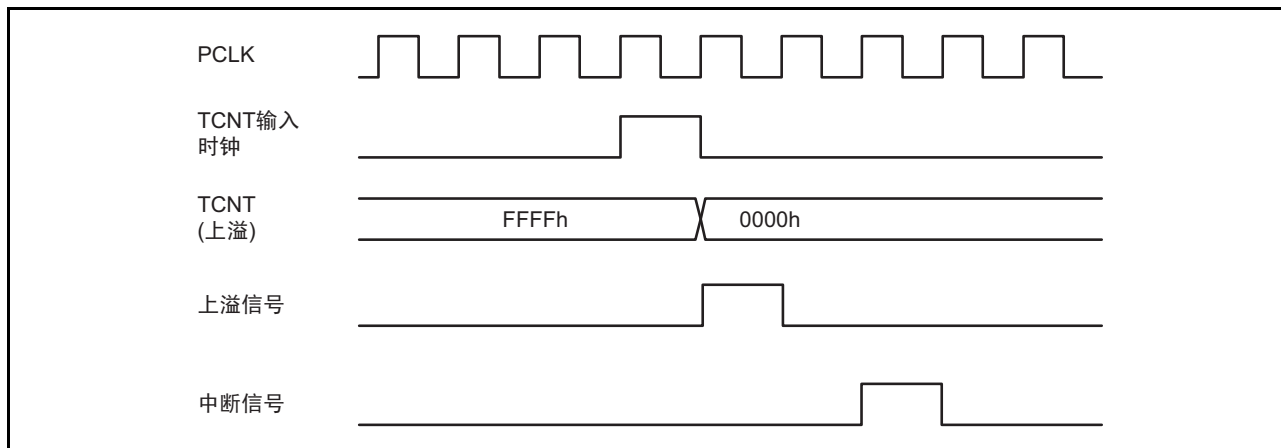


图 21.104 TCIV 中断的置位时序

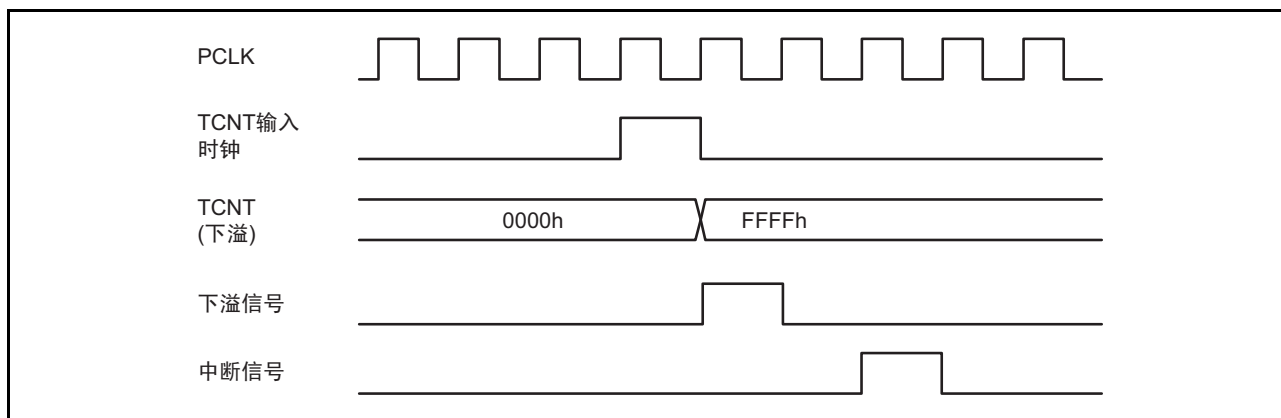


图 21.105 TCIU 中断的置位时序

21.6 使用时的注意事项

21.6.1 模块停止功能的设定

能通过模块停止控制寄存器设定允许或者禁止 MTU 的运行，初始值为停止 MTU 的运行。能通过解除模块时钟停止模式，使寄存器变为可存取的状态，详细内容请参照“11. 低功耗功能”。

21.6.2 输入时钟的制限事项

必须注意：在单边沿的情况下，输入时钟的脉宽至少为 1.5 个状态；在双边沿的情况下，输入时钟的脉宽至少为 2.5 个状态。否则，就不能正常运行。

在相位计数模式中，2 个输入时钟的相位差和重叠都至少为 1.5 个状态，脉宽至少为 2.5 个状态。相位计数模式的输入时钟条件如图 21.106 所示。

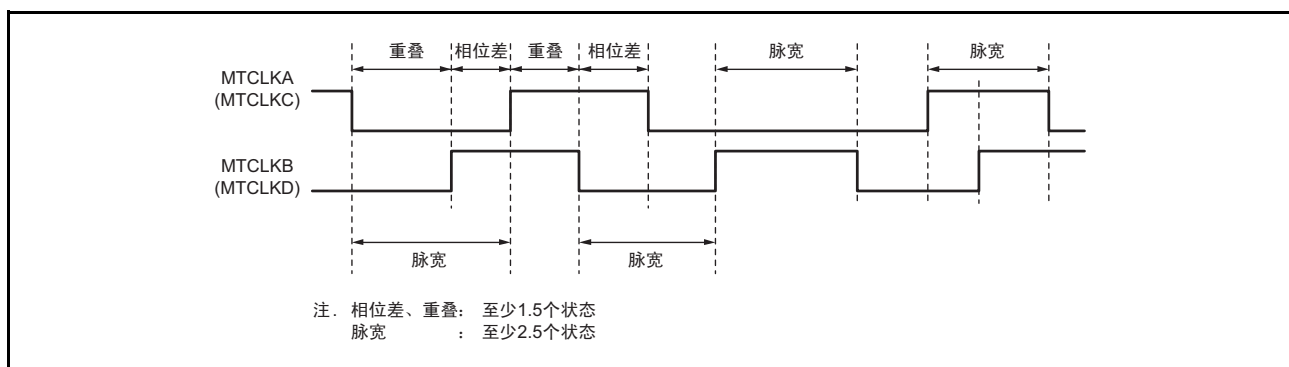


图 21.106 相位计数模式的相位差、重叠和脉宽

21.6.3 设定周期时的注意事项

如果设定为通过比较匹配清除计数器，就在 TCNT 和 TGR 的值相同的最后状态（在更新 TCNT 相同后的计数值时）清除 TCNT 计数器。因此，实际的计数器频率用以下表达式表示：

- MTU0~4 的情况

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

- MTU5 的情况

$$f = \frac{PCLK}{N}$$

- f : 计数器的频率
PCLK : MTU 时钟的工作频率
N : TGR 的设定值

21.6.4 TCNT 的写和清除的竞争

如果在 TCNT 的写周期中产生计数器清除信号，就不写 TCNT 而优先清除 TCNT。
 此时序如图 21.107 所示。

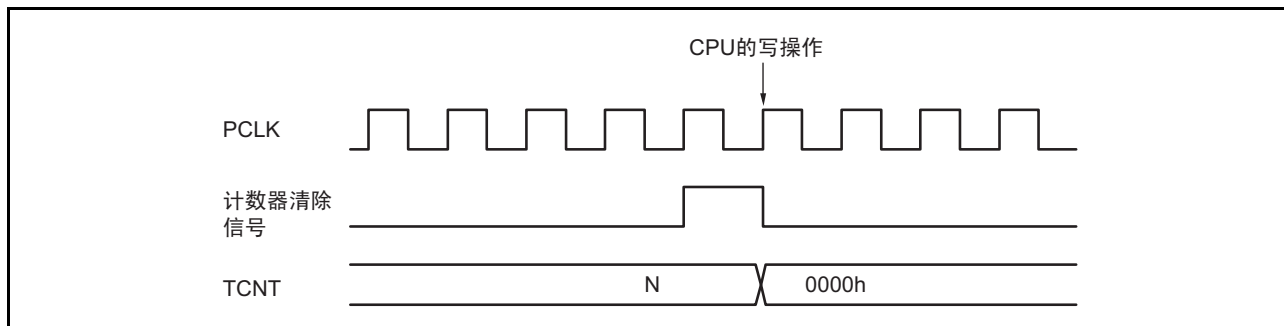


图 21.107 TCNT 的写和清除的竞争

21.6.5 TCNT 的写和递增计数的竞争

即使在 TCNT 的写周期中发生递增计数，也不进行递增计数而优先写 TCNT。
 此时序如图 21.108 所示。

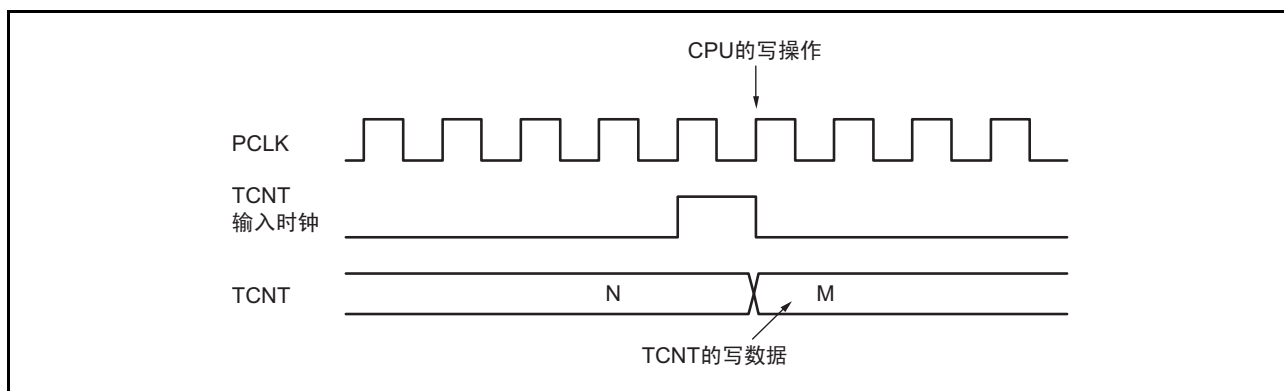


图 21.108 TCNT 的写和递增计数的竞争

21.6.6 TGR 的写和比较匹配的竞争

如果在 TGR 的写周期中发生比较匹配，就写 TGR 并且也产生比较匹配信号。
此时序如图 21.109 所示。

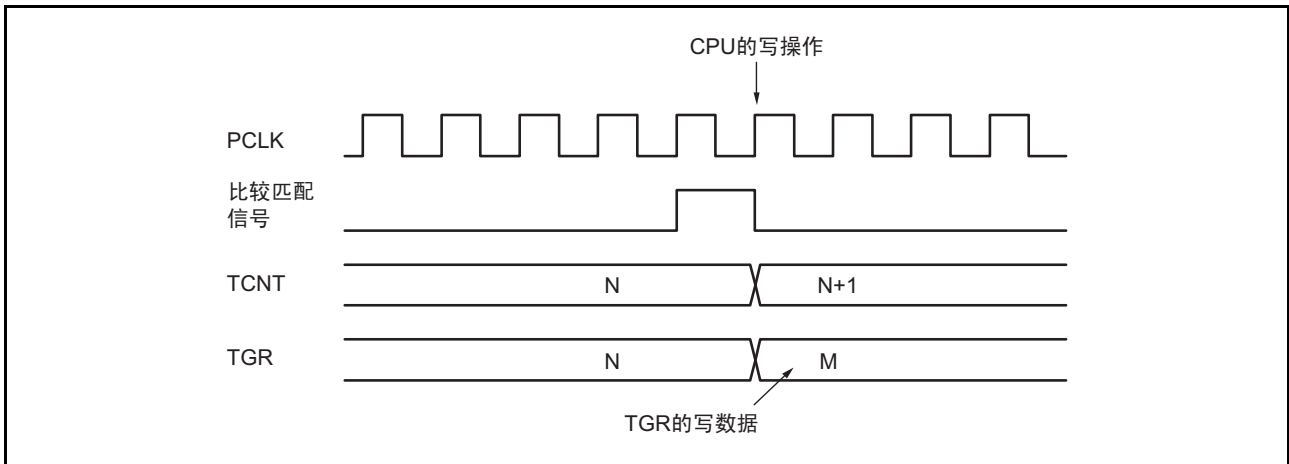


图 21.109 TGR 的写和比较匹配的竞争

21.6.7 缓冲寄存器的写和比较匹配的竞争

如果在 TGR 的写周期中发生比较匹配，通过缓冲运行传送到 TGR 的数据就为写之前的数据。
此时序如图 21.110 所示。

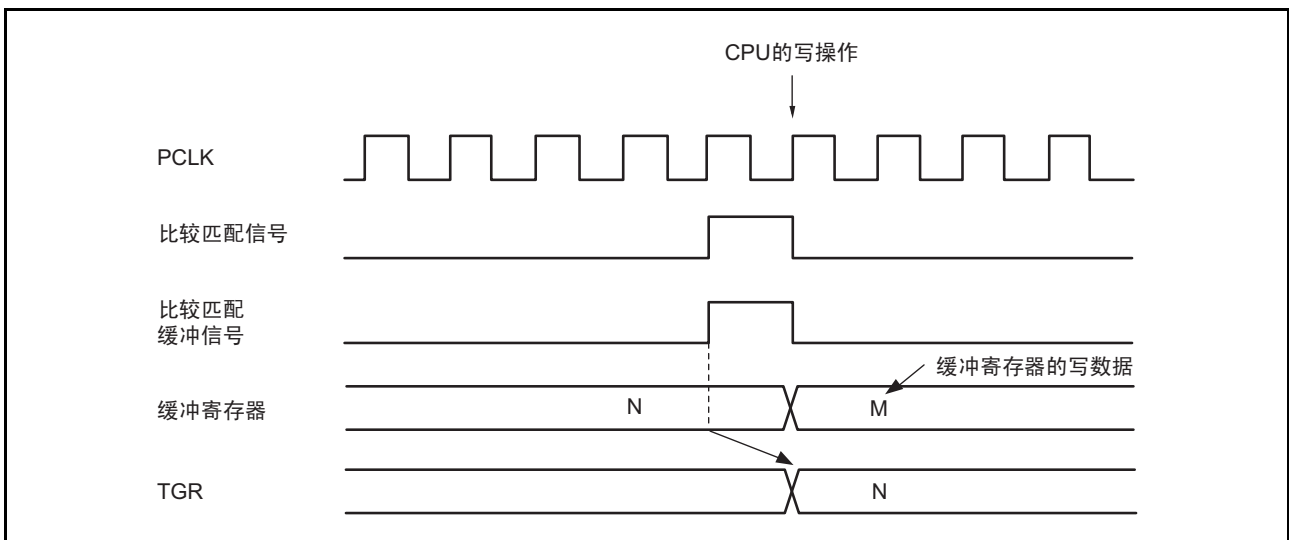


图 21.110 缓冲寄存器的写和比较匹配的竞争

21.6.8 缓冲寄存器的写和 TCNT 清除的竞争

当通过缓冲传送模式寄存器 (TBTM) 将缓冲传送时序设定为清除 TCNT 时, 如果在 TGR 的写周期中产生 TCNT 的清除信号, 通过缓冲运行传送到 TGR 的数据就为写之前的数据。

此时序如图 21.111 所示。

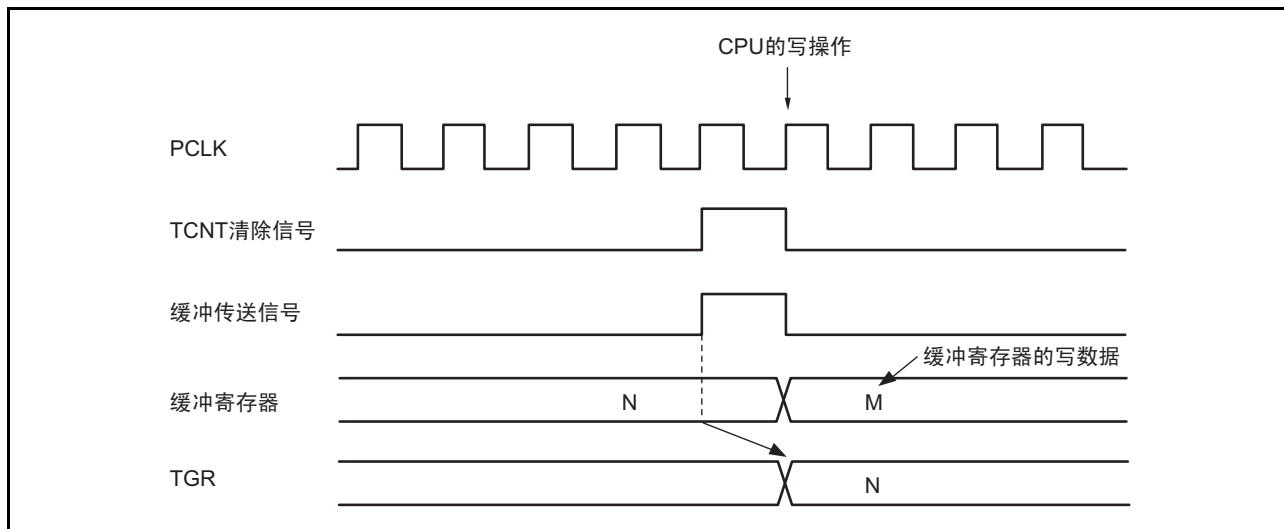


图 21.111 缓冲寄存器的写和 TCNT 清除的竞争

21.6.9 TGR 的读和输入捕捉的竞争

如果在 TGR 的读周期中产生输入捕捉信号, 读出的数据就在 MTU0 ~ MTU4 时为输入捕捉传送前的数据而在 MTU5 时为输入捕捉传送后的数据。

此时序如图 21.112 和图 21.113 所示。

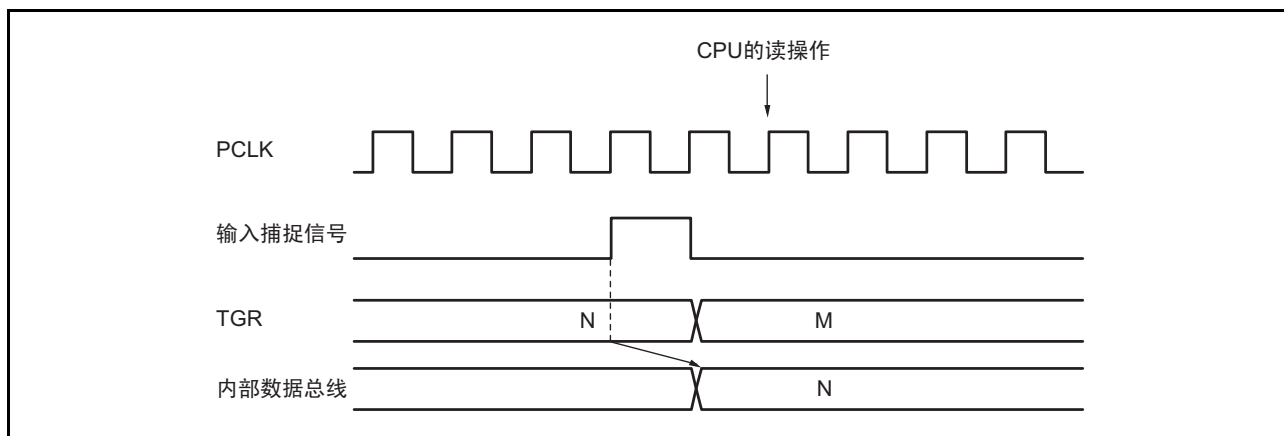


图 21.112 TGR 的读和输入捕捉的竞争 (MTU0 ~ MTU4)

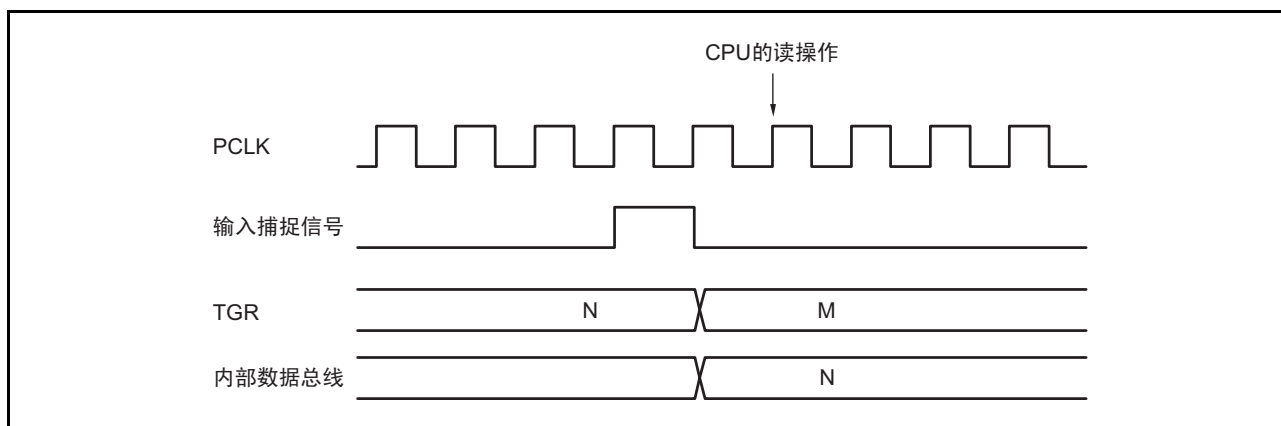


图 21.113 TGR 的读和输入捕捉的竞争 (MTU5)

21.6.10 TGR 的写和输入捕捉的竞争

如果在 TGR 的写周期中产生输入捕捉信号，就在 MTU0 ~ MTU4 时不写 TGR 而优先输入捕捉，但是在 MTU5 时写 TGR 并且也产生输入捕捉信号。

此时序如图 21.114 和图 21.115 所示。

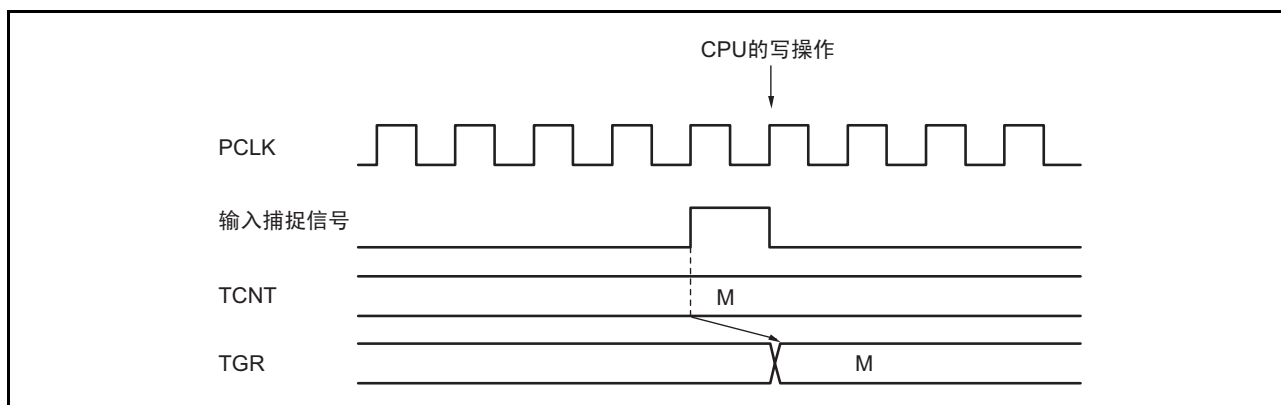


图 21.114 TGR 的写和输入捕捉的竞争 (MTU0 ~ MTU4)

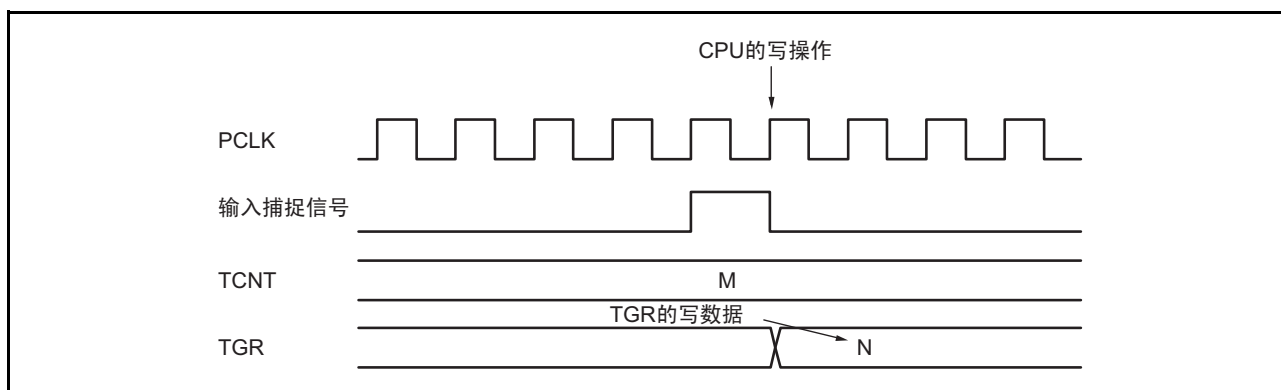


图 21.115 TGR 的写和输入捕捉的竞争 (MTU5)

21.6.11 缓冲寄存器的写和输入捕捉的竞争

如果在缓冲器写周期中产生输入捕捉信号，就不写缓冲寄存器而优先缓冲运行。
此时序如图 21.116 所示。

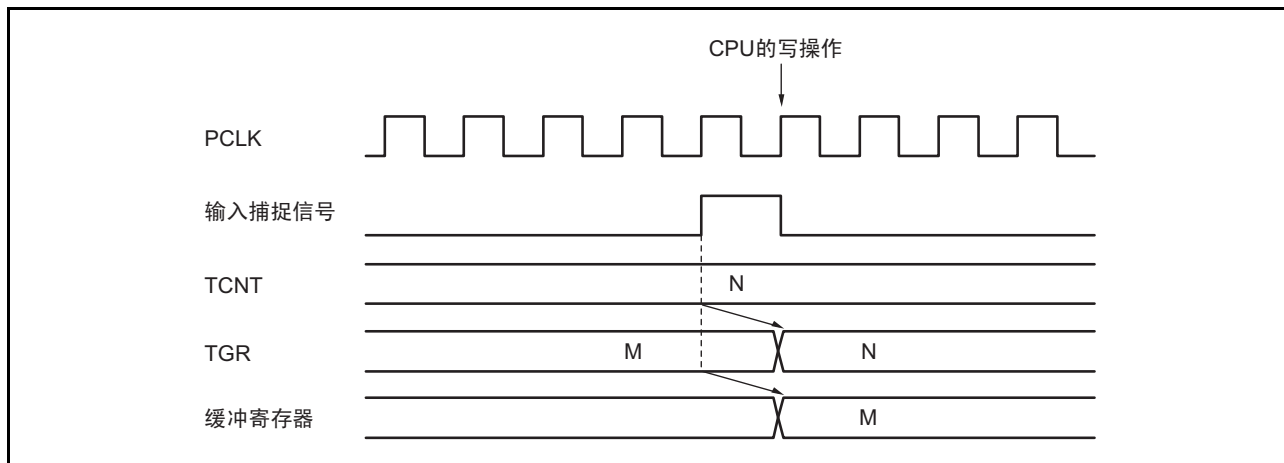


图 21.116 缓冲寄存器的写和输入捕捉的竞争

21.6.12 级联中的 MTU2.TCNT 的写和上溢 / 下溢的竞争

如果级联定时器的计数器 (MTU1.TCNT 和 MTU2.TCNT) 并且在 MTU1.TCNT 计数的瞬间 (MTU2.TCNT 上溢 / 下溢的瞬间) 和 MTU2.TCNT 的写周期发生竞争, 就写 MTU2.TCNT 而禁止 MTU1.TCNT 的计数信号。此时, MTU1.TGRA 作为比较匹配寄存器运行, 如果和 MTU1.TCNT 的值相同, 就产生比较匹配信号。

如果选择 MTU1.TCNT 计数时钟作为 MTU0 的输入捕捉源, MTU0.TGRA ~ TGRD 就进行输入捕捉运行。如果选择 MTU0.TGRC 的比较匹配 / 输入捕捉作为 MTU1.TGRB 的输入捕捉源, MTU1.TGRB 就进行输入捕捉运行。

此时序如图 21.117 所示。

如果在级联运行时设定清除 TCNT, 就必须进行 MTU1 和 MTU2 的同步设定。

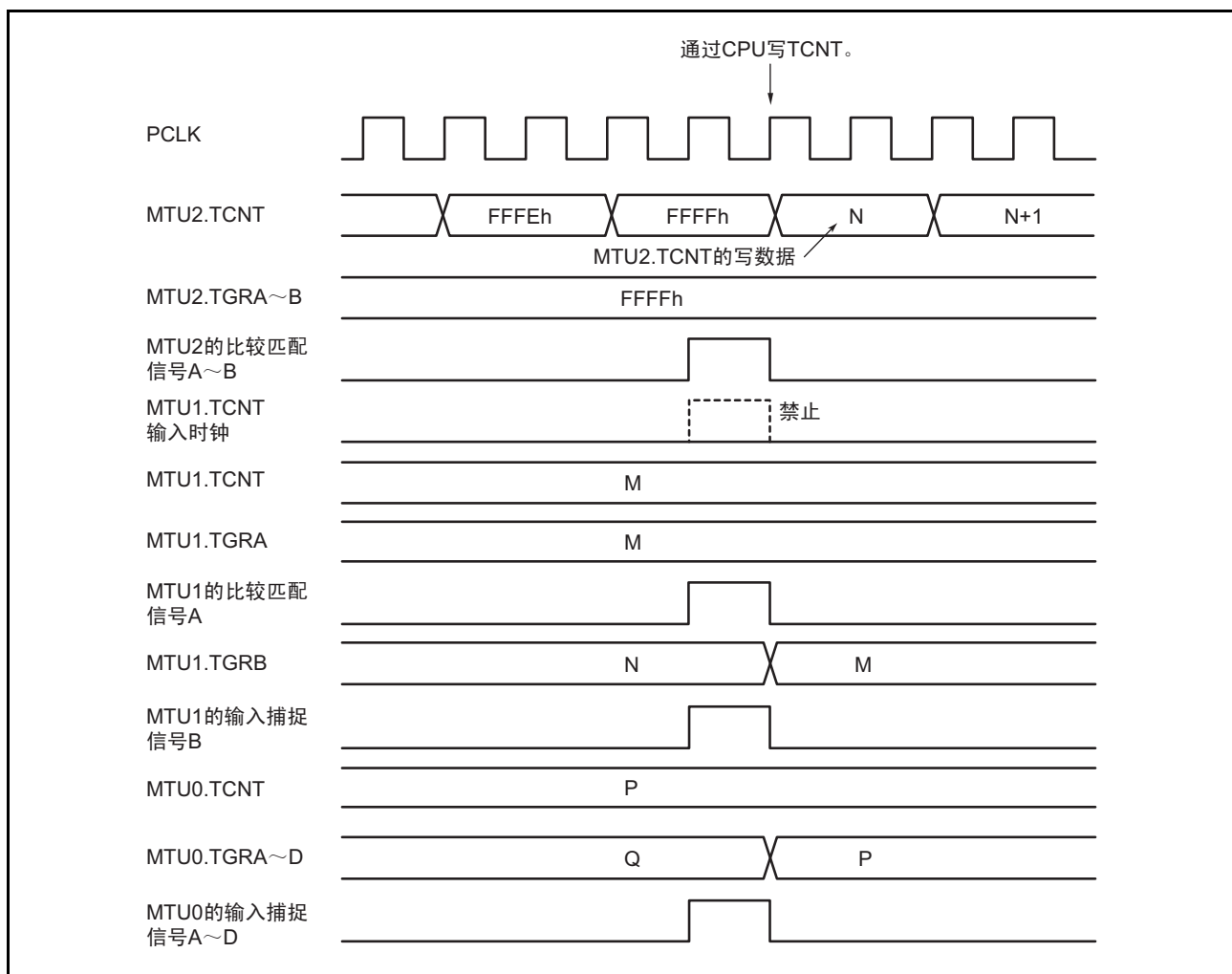


图 21.117 级联中的 MTU2.TCNT 的写和上溢 / 下溢的竞争

21.6.13 互补 PWM 模式停止时的计数器值

如果 MTU3.TCNT 和 MTU4.TCNT 在互补 PWM 运行时停止计数运行，MTU3.TCNT 就变为定时器的死区时间寄存器 (TDDR) 的值，MTU4.TCNT 变为“0000h”。

一旦重新开始互补 PWM 运行，计数器就自动从初始状态开始计数。

此说明图如图 21.118 所示。

如果要在其他运行模式中开始计数，就必须给 MTU3.TCNT 和 MTU4.TCNT 设定计数初始值。

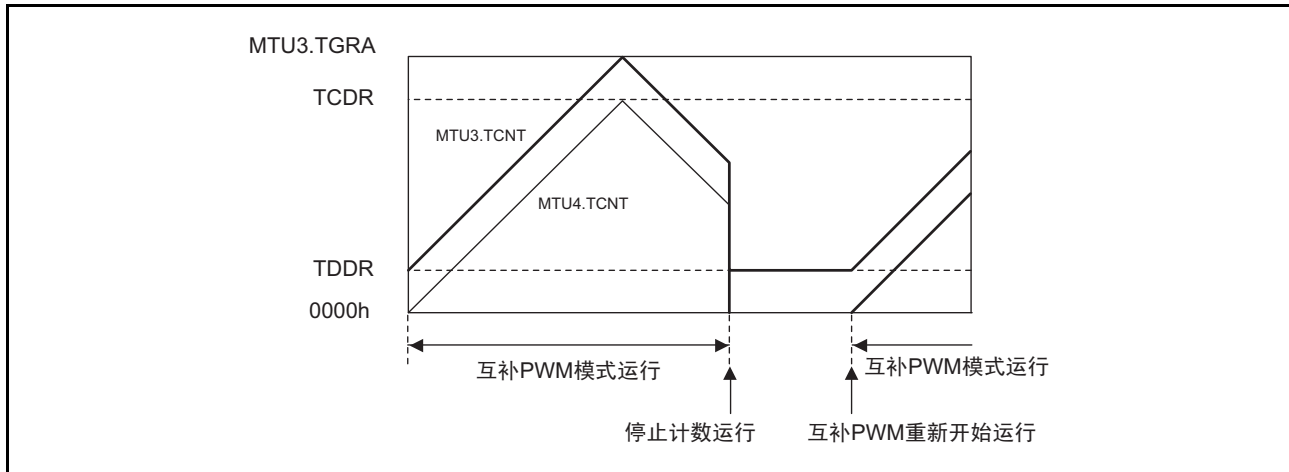


图 21.118 互补 PWM 模式停止时的计数器值 (MTU3 和 MTU4 运行)

21.6.14 互补 PWM 模式中的缓冲运行的设定

在互补 PWM 模式中，必须通过缓冲运行改写 PWM 周期设定寄存器 (MTU3.TGRC)、定时器的周期数据寄存器 (TCDR) 和占空比设定寄存器 (MTU3.TGRC、MTU4.TGRC、MTU4.TGRC)，并且将 MTU4.TMDR 的 BFA 位和 BFB 位置“0”。如果将 MTU4.TMDR 的 BFA 位置“1”，就无法进行 MTIOC4C 引脚的波形输出。同样，如果将 MTU4.TMDR 的 BFB 位置“1”，就无法进行 MTIOC4D 引脚的波形输出。

按照 MTU3.TMDR 寄存器的 BFA 位和 BFB 位的设定，进行互补 PWM 模式的 MTU3 和 MTU4 的缓冲运行。如果将 MTU3.TMDR 寄存器的 BFA 位置“1”，就在 MTU3.TGRC 用作 MTU3.TGRC 的缓冲寄存器的同时，MTU4.TGRC 用作 MTU4.TGRC 的缓冲寄存器，并且 TCBR 用作 TCDR 的缓冲寄存器。

21.6.15 复位同步 PWM 模式的缓冲运行和比较匹配标志

要设定为在复位同步 PWM 模式中进行缓冲运行时，必须将 MTU4.TMDR 的 BFA 位和 BFB 位置“0”。如果将 MTU4.TMDR 的 BFA 位置“1”，就不能进行 MTIOC4C 引脚的波形输出。同样，如果将 MTU4.TMDR 的 BFB 位置“1”，就不能进行 MTIOC4D 引脚的波形输出。

按照 MTU3.TMDR 的 BFA 位和 BFB 位的设定，进行复位同步 PWM 模式的 MTU3 和 MTU4 的缓冲运行。例如，如果将 MTU3.TMDR 的 BFA 位置“1”，就在 MTU3.TGRC 用作 MTU3.TGRC 的缓冲寄存器的同时，MTU4.TGRC 用作 MTU4.TGRC 的缓冲寄存器。

当 MTU3.TGRC 和 MTU3.TGRC 用作缓冲寄存器时，不产生对应的 TGIC 和 TGID 的中断请求。

将 MTU3.TMDR 的 BFA 位和 BFB 位置“1”并且将 MTU4.TMDR 的 BFA 位和 BFB 位置“0”时的 MTU3.TGRC、MTU4.TGRC、MTIOC3m、MTIOC4m 的运行例子如图 21.119 所示。

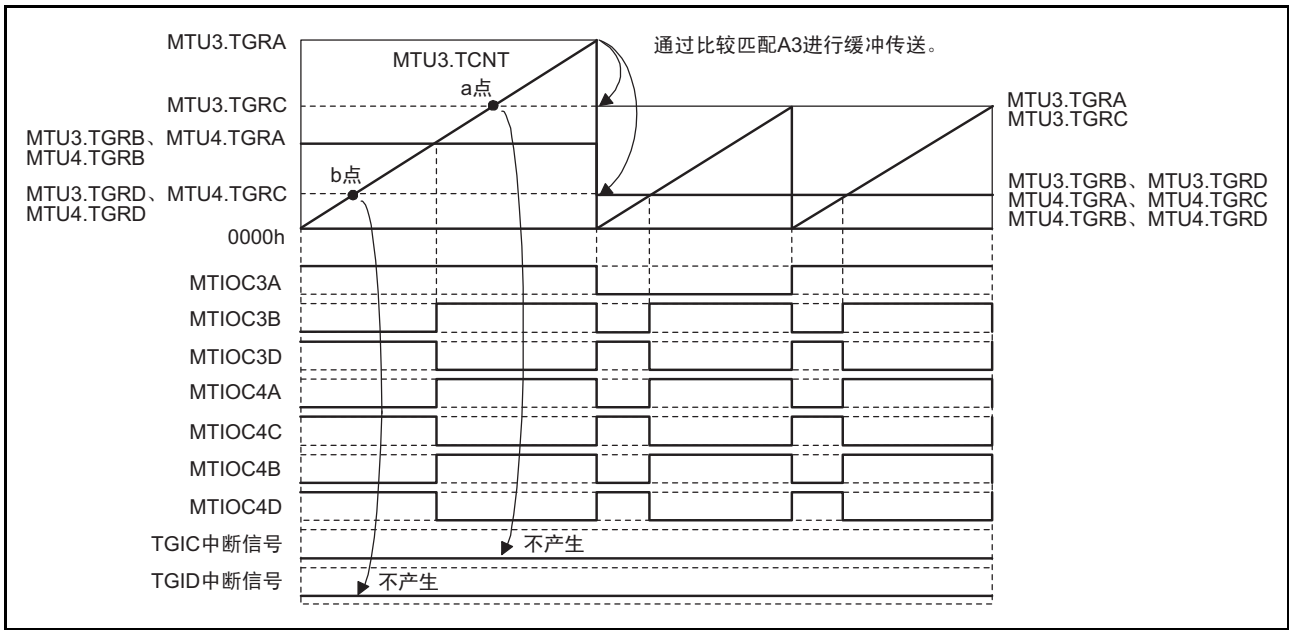


图 21.119 复位同步 PWM 模式的缓冲运行和比较匹配标志

21.6.16 复位同步 PWM 模式的上溢标志

如果设定复位同步 PWM 模式并且将 TSTR 寄存器的 CST3 位置“1”，就开始 MTU3.TCNT 和 MTU4.TCNT 的计数运行。此时，MTU4.TCNT 的计数时钟源和计数边沿服从 MTU3.TCR 的设定。

在复位同步 PWM 模式中，当周期寄存器 MTU3.TGRA 的设定值为“FFFFh”并且指定 MTU3.TGRA 的比较匹配为计数器清除源时，如果 MTU3.TCNT 和 MTU4.TCNT 递增计数到“FFFFh”，就产生和 MTU3.TGRA 的比较匹配，并且清除 MTU3.TCNT 和 MTU4.TCNT 的计数。此时，不产生对应的 TCIV 中断请求。

在复位同步 PWM 模式中，当周期寄存器 MTU3.TGRA 的设定值为“FFFFh”并且指定 MTU3.TGRA 的比较匹配为计数器清除源而不进行同步设定时的运行例子如图 21.120 所示。

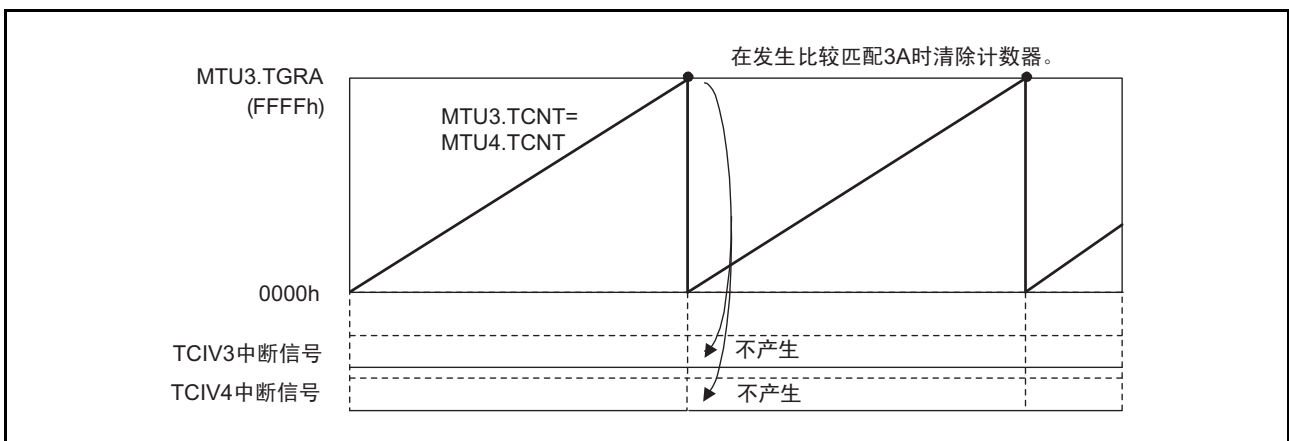


图 21.120 复位同步 PWM 模式的上溢标志

21.6.17 上溢 / 下溢和计数器清除的竞争

如果同时发生上溢 / 下溢和计数器清除，就优先清除 TCNT 而不产生对应的 TCIV 中断。
将 TGR 的比较匹配作为清除源并且给 TGR 设定“FFFFh”时的运行时序如图 21.121 所示。

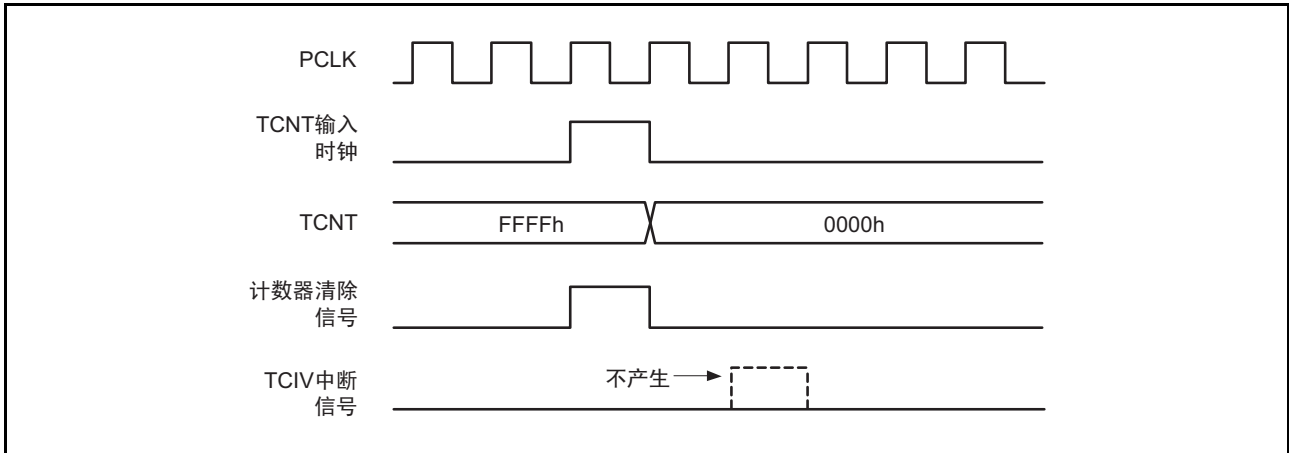


图 21.121 上溢和计数器清除的竞争

21.6.18 TCNT 的写和上溢 / 下溢的竞争

即使在 TCNT 的写周期中发生递增计数 / 递减计数和上溢 / 下溢，也优先写 TCNT 而不产生对应的中断。
TCNT 的写和上溢竞争时的运行时序如图 21.122 所示。

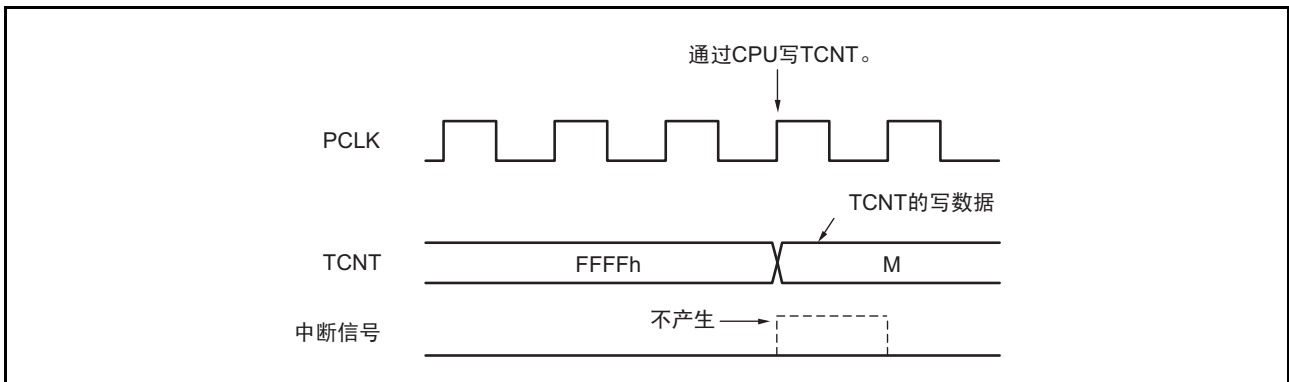


图 21.122 TCNT 的写和上溢的竞争

21.6.19 从正常运行或者 PWM 模式 1 转移到复位同步 PWM 模式时的注意事项

必须注意：要从 MTU3 和 MTU4 的正常运行或者 PWM 模式 1 转移到复位同步 PWM 模式时，如果在将输出引脚 (MTIOC3B、MTIOC3D、MTIOC4A、MTIOC4C、MTIOC4B、MTIOC4D) 置为 High 电平的状态下停止计数器并且在转移到复位同步 PWM 模式后进行运行，就不能正确地进行引脚的初始输出。

在从正常运行转移到复位同步 PWM 模式时，必须在给 MTU3.TIORH、MTU3.TIORL、MTU4.TIORH、MTU4.TIORL 寄存器写“11h”并且将输出引脚初始化为 Low 电平后设定寄存器的初始值“00h”，然后进行模式转移。

在从 PWM 模式 1 转移到复位同步 PWM 模式时，必须暂且转移到正常运行，在将输出引脚初始化为 Low 电平后设定寄存器的初始值“00h”，然后转移到复位同步 PWM 模式。

21.6.20 互补 PWM 模式和复位同步 PWM 模式的输出电平

当 MTU3 和 MTU4 为互补 PWM 模式或者复位同步 PWM 模式时，通过定时器的输出控制寄存器 1 (TOCR1) 的 OLSF 位和 OLSN 位设定 PWM 波形的输出电平。在互补 PWM 模式或者复位同步 PWM 模式中，必须将 TIOR 寄存器置“00h”。

21.6.21 模块停止状态时的中断

如果在请求中断的状态下变为模块停止状态，就不能清除 CPU 的中断源或者 DMAC/DTC 的启动源。必须事先将中断置为无效，然后设定为模块停止状态。

21.6.22 级联中的 MTU1.TCNT 和 MTU2.TCNT 的同时输入捕捉

在将定时器的计数器 1 和计数器 2 (MTU1.TCNT/MTU2.TCNT) 级联用作 32 位计数器时，即使 MTIOC1A 和 MTIOC2A 或者 MTIOC1B 和 MTIOC2B 同时进行输入捕捉的输入，也会与内部时钟同步将输入到 MTU1.TCNT 和 MTU2.TCNT 的外部输入捕捉信号取到内部，此时可能因 MTIOC1A 和 MTIOC2A 或者 MTIOC1B 和 MTIOC2B 的取时序产生偏差而不能正确地捕捉级联计数器的值。

例如，MTU1.TCNT (高 16 位的计数器) 应该捕捉由 MTU2.TCNT (低 16 位的计数器) 的上溢产生的递增计数值，却会捕捉递增计数前的计数值。此时，应该将 MTU1.TCNT 为“FFF1h”以及 MTU2.TCNT 为“0000h”的值传送到 MTU1.TGRA 和 MTU2.TGRA 或者 MTU1.TGRB 和 MTU2.TGRB，却会误传送 MTU1.TCNT 为“FFF0h”以及 MTU2.TCNT 为“0000h”的值。

在 MTU 中，能通过 1 个输入捕捉的输入，追加可同时捕捉 MTU1.TCNT 和 MTU2.TCNT 的功能。如果使用此功能，就不会产生 MTU1.TCNT 和 MTU2.TCNT 的捕捉时序偏差而能取到 32 位计数器的值。详细内容请参照“21.2.8 定时器的输入捕捉控制寄存器 (TICCR)”。

21.6.23 未使用互补 PWM 模式的输出保护功能时的注意事项

互补 PWM 模式的输出保护功能在初始状态下有效。如果不使用此功能，就必须给 POE.POECR2 寄存器写“00h”。

21.6.24 同步清除互补 PWM 模式时的异常动作的防止

在互补 PWM 模式中，如果在同步计数器清除时输出波形控制有效 (TWCR.WRE 位 =1) 的状态下满足条件 1 或者条件 2，就会出现以下现象：

- PWM 输出引脚的死区时间变短 (或者消失)。
- 在非有效电平输出期间，从 PWM 反相输出引脚输出有效电平。

条件 1: 在初始输出的抑止期间⑩，当 PWM 输出在死区时间的期间中进行同步清除时 (参照图 21.123)

条件 2: 在初始输出的抑止期间⑩和 ⑪，当在满足 $MTU3.TGRB \leq TDDR$ 、 $MTU4.TGRA \leq TDDR$ 或者 $MTU4.TGRB \leq TDDR$ 的状态下进行同步清除时 (参照图 21.124)

能通过以下方法回避这种现象：

- 在将比较寄存器 MTU3.TGRB、MTU4.TGRA 和 MTU4.TGRB 全部设定为不小于死区时间数据寄存器 (TDDR) 的 2 倍的状态下进行同步清除。

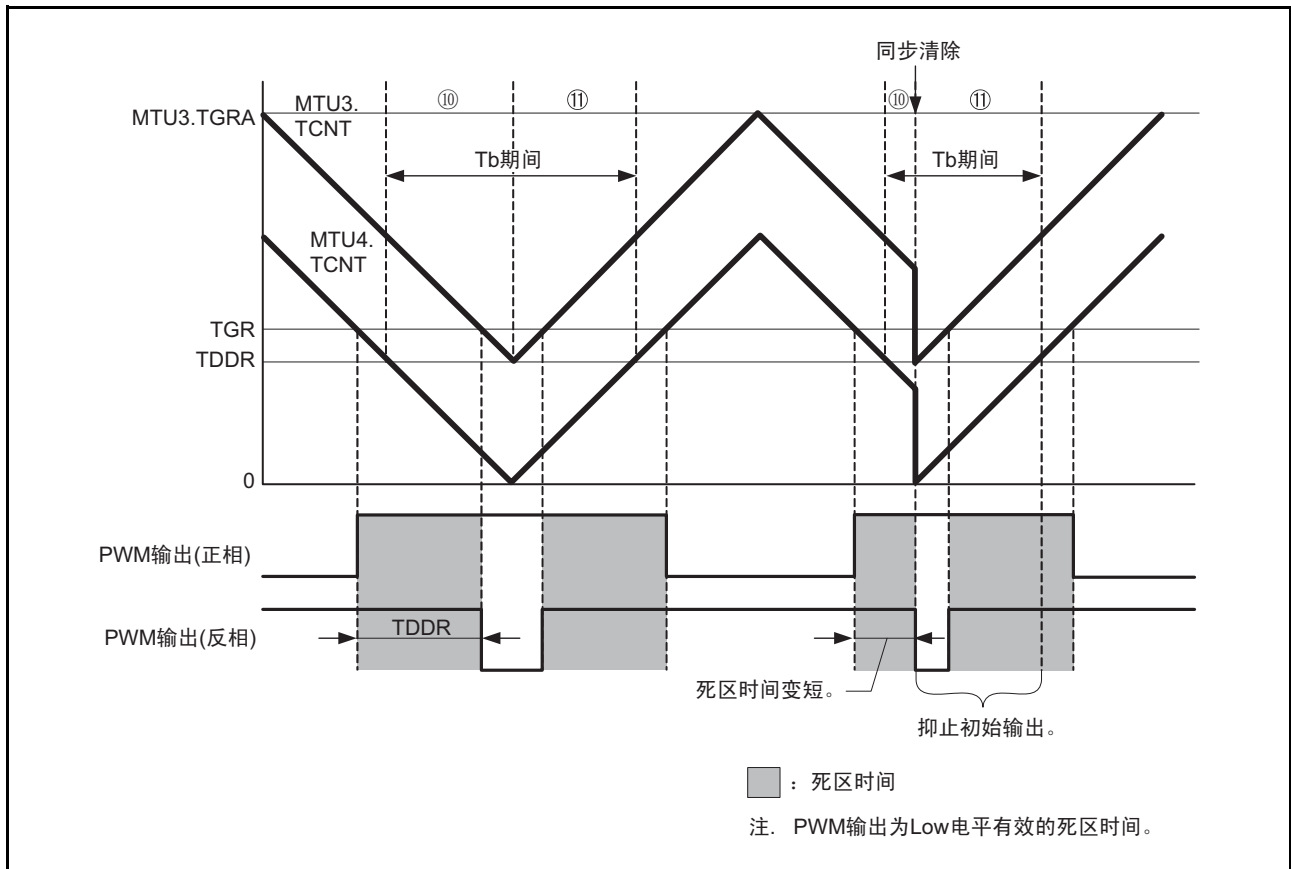


图 21.123 同步清除的例子 (条件 1 的情况)

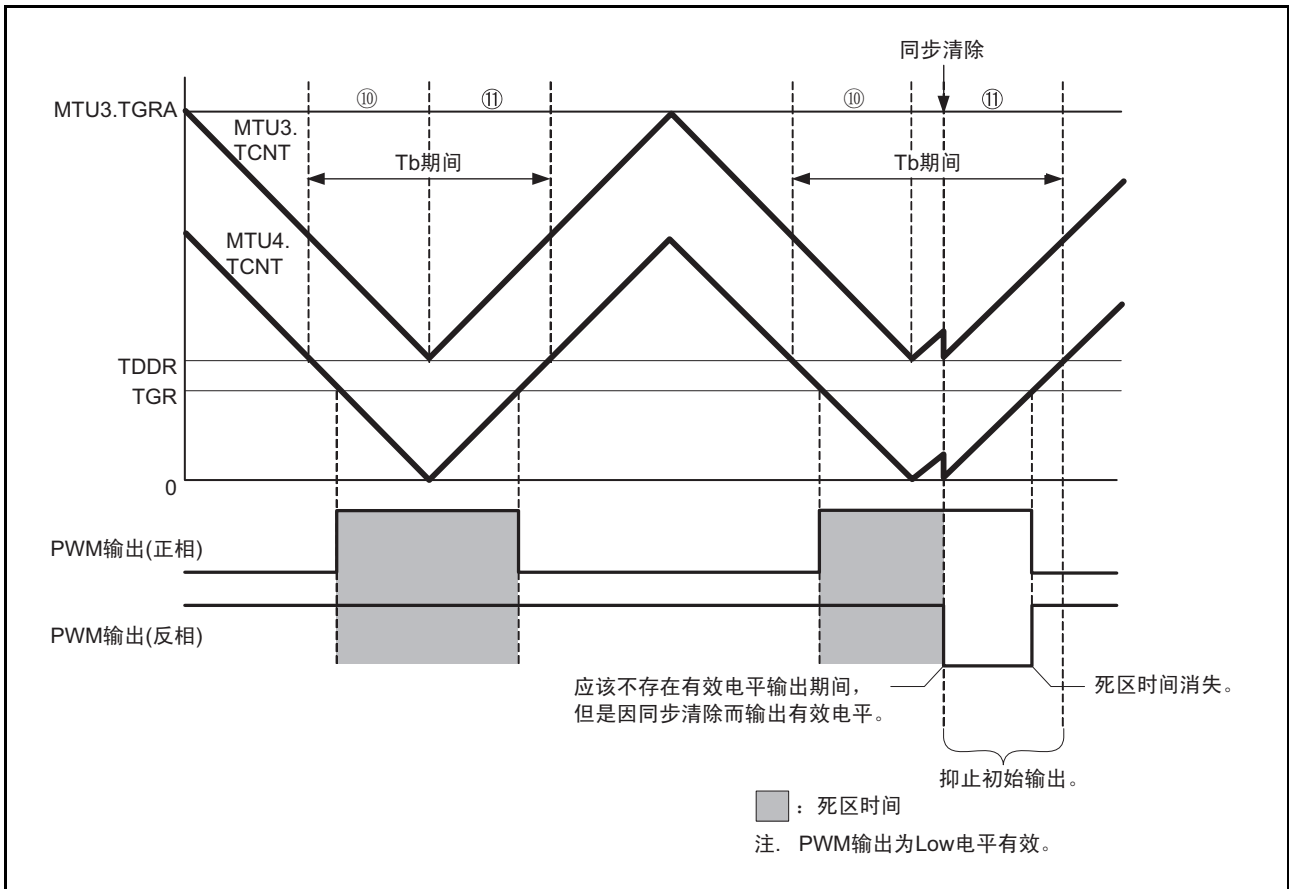


图 21.124 同步清除的例子 (条件 2 的情况)

21.6.25 比较匹配脉冲中断的连续输出

如果将 TGR 寄存器置“0”并且将计数器时钟设定为 PCLK/1 以及设定为通过比较匹配清除计数器，TCNT 计数器就保持“0000h”而不更新，并且将脉冲的比较匹配中断连续输出为电平状态。

在使用脉冲中断的情况下，中断控制器无法检测到第 3 个以后的中断。

比较匹配脉冲中断连续输出的运行时序如图 21.125 所示。

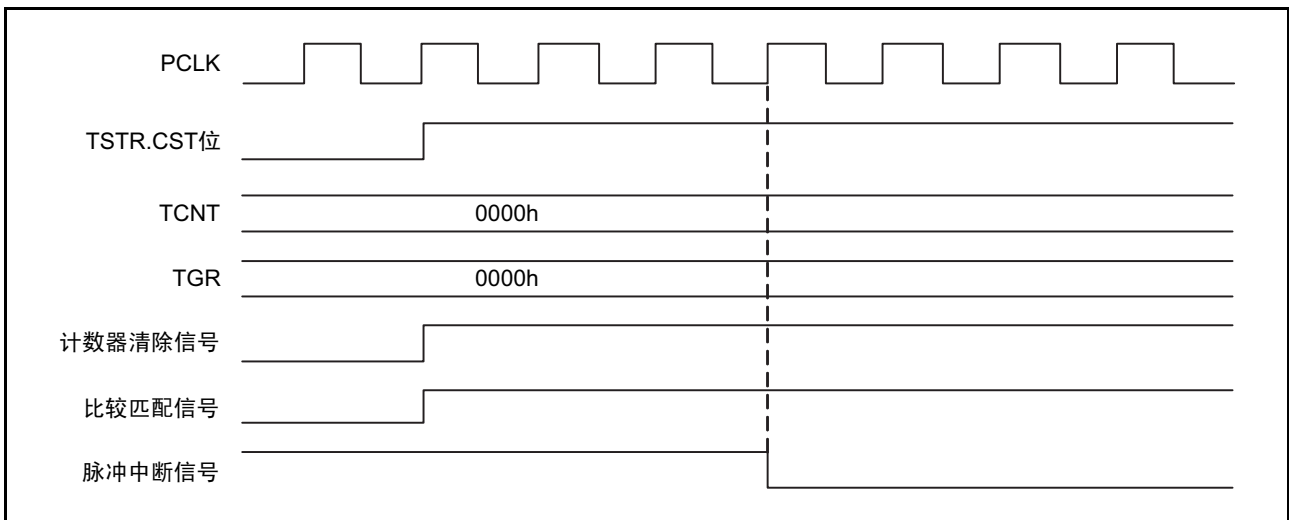


图 21.125 比较匹配脉冲中断的连续输出

21.7 MTU 输出引脚的初始化方法

21.7.1 运行模式

MTU 有以下 6 种运行模式，能在任意的模式中进行波形输出。

- 正常模式 (MTU0 ~ MTU4)
- PWM 模式 1 (MTU0 ~ MTU4)
- PWM 模式 2 (MTU0 ~ MTU2)
- 相位计数模式 1 ~ 4 (MTU1 和 MTU2)
- 互补 PWM 模式 (MTU3 和 MTU4)
- 复位同步 PWM 模式 (MTU3 和 MTU4)

在此说明各模式中的 MTU 输出引脚的初始化方法。

21.7.2 因运行过程中的异常而重新设定时的运行

如果在 MTU 运行过程中发生异常，就必须通过系统切断 MTU 的输出。要切断输出时，将引脚的输出切换为端口并且反相输出有效电平。对于马达驱动引脚，也能使用端口输出允许位 (POE)，通过硬件来切断输出。因运行过程中的异常等而重新设定时的引脚初始化步骤以及重新设定后在其他运行模式中重新开始时的步骤如下所示。

如上所述，因为 MTU 有 6 个运行模式，所以有 36 种模式转移的组合，但是有通道和模式的组合中不存在的转移。模式转移的组合一览表如表 21.59 所示。

表中使用下述的符号表示：

Normal: 正常模式

PWM1: PWM 模式 1

PWM2: PWM 模式 2

PCM: 相位计数模式 1 ~ 4

CPWM: 互补 PWM 模式

RPWM: 复位同步 PWM 模式

表 21.59 模式转移的组合

	Normal	PWM1	PWM2	PCM	CPWM	RPWM
Normal	(1)	(2)	(3)	(4)	(5)	(6)
PWM1	(7)	(8)	(9)	(10)	(11)	(12)
PWM2	(13)	(14)	(15)	(16)	none	none
PCM	(17)	(18)	(19)	(20)	none	none
CPWM	(21)	(22)	none	none	(23) (24)	(25)
RPWM	(26)	(27)	none	none	(28)	(29)

21.7.3 因运行过程中的异常而对引脚进行初始化的步骤以及模式转移的概要

- 在转移到定时器的 I/O 控制寄存器 (TIOR) 选择的引脚输出电平模式 (Normal、PWM1、PWM2、PCM) 时, 必须通过设定 TIOR 对引脚进行初始化。
- 因为在 PWM 模式 1 中不将波形输出到 MTIOCnB (MTIOCnD) 引脚, 所以即使设定 TIOR 也不对引脚进行初始化。要进行初始化时, 必须在正常模式中进行初始化后转移到 PWM 模式 1。
- 因为在 PWM 模式 2 中不将波形输出到周期寄存器的引脚, 所以即使设定 TIOR 也不对引脚进行初始化。要进行初始化时, 必须在正常模式中进行初始化后转移到 PWM 模式 2。
- 在正常模式或者 PWM 模式 2 中, 如果 TGRC 和 TGRD 都用作缓冲寄存器, 即使设定 TIOR 也不对缓冲寄存器的引脚进行初始化。要进行初始化时, 必须在解除缓冲器模式进行初始化后重新设定缓冲器模式。
- 在 PWM 模式 1 中, 如果 TGRC 或者 TGRD 用作缓冲寄存器, 即使设定 TIOR 也不对 TGRC 的引脚进行初始化。要对 TGRC 的引脚进行初始化时, 必须在解除缓冲器模式进行初始化后重新设定缓冲器模式。
- 在转移到定时器的输出控制寄存器 (TOCR) 选择的引脚输出电平模式 (CPWM、RPWM) 时, 必须转移到正常模式并且通过 TIOR 进行初始化, 在将 TIOR 恢复为初始值后通过定时器的输出主控允许寄存器 (TOER) 暂时禁止 MTU3 和 MTU4 的输出, 然后按照模式的设定步骤 (TOCR 的设定、TMDR 的设定和 TOER 的设定) 运行。

注. 在没有特别要求的情况下, 各项记述中的 n 为通道号。

根据表 21.59 的组合 No. 对引脚进行初始化的步骤如下所示, 有效电平为 Low 电平。

(1) 在正常模式的运行过程中发生异常并且在正常模式中重新开始时的运行

在正常模式中发生异常并且重新设定后在正常模式中重新开始时的说明图如图 21.126 所示。

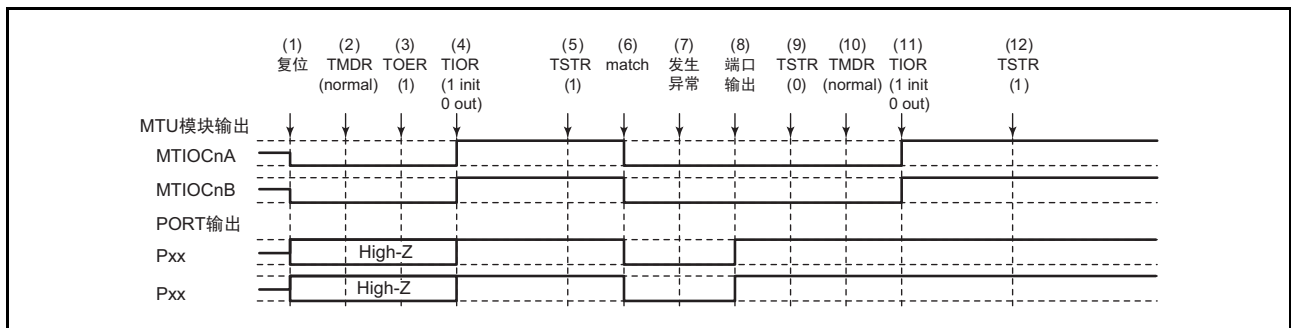


图 21.126 在正常模式中发生异常并且在正常模式中恢复的情况

- 复位后, MTU 的输出为 Low 电平, 端口为高阻抗。
- 复位后, TMDR 被设定为正常模式。
- 在 MTU3 和 MTU4 中, 必须在通过 TIOR 对引脚进行初始化前用 TOER 允许输出。
- 必须通过 TIOR 对引脚进行初始化 (这是初始输出为 High 电平并且在比较匹配时输出 Low 电平的例子)。
- 通过 TSTR 开始计数运行。
- 在比较匹配时输出 Low 电平。
- 发生异常。
- 必须通过 TIOR 禁止引脚输出, 而作为端口输出, 反相输出有效电平。
- 通过 TSTR 停止计数运行。
- 在正常模式中重新开始的情况下不需要此步骤。
- 必须通过 TIOR 对引脚进行初始化。
- 通过 TSTR 重新开始。

(2) 在正常模式的运行过程中发生异常并且在 PWM 模式 1 中重新开始时的运行

在正常模式中发生异常并且重新设定后在 PWM 模式 1 中重新开始时的说明图如图 21.127 所示。

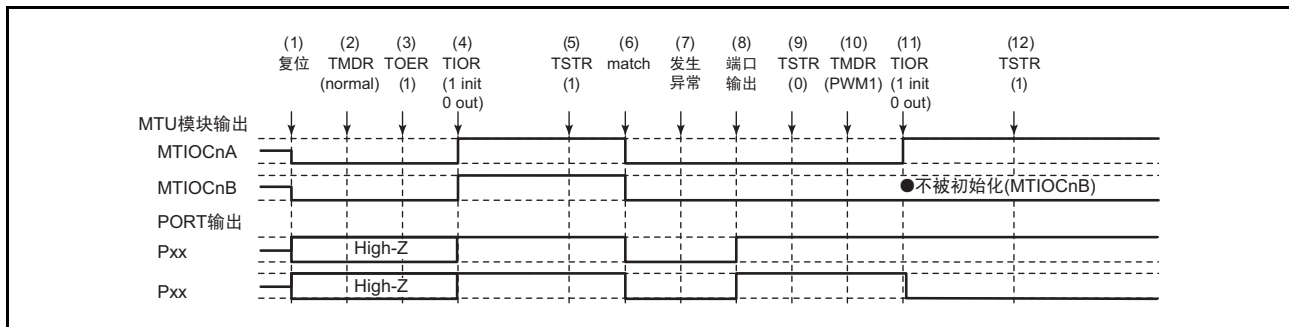


图 21.127 在正常模式中发生异常并且在 PWM 模式 1 中恢复的情况

(1) ~ (9) 和图 21.126 通用。

(10) 设定 PWM 模式 1。

(11) 必须通过 TIOR 对引脚进行初始化（在 PWM 模式 1 中不对 MTIOCnB 进行初始化。如果要进行初始化，就必须在正常模式中进行初始化后转移到 PWM 模式 1）。

(12) 通过 TSTR 重新开始。

(3) 在正常模式的运行过程中发生异常并且在 PWM 模式 2 中重新开始时的运行

在正常模式中发生异常并且重新设定后在 PWM 模式 2 中重新开始时的说明图如图 21.128 所示。

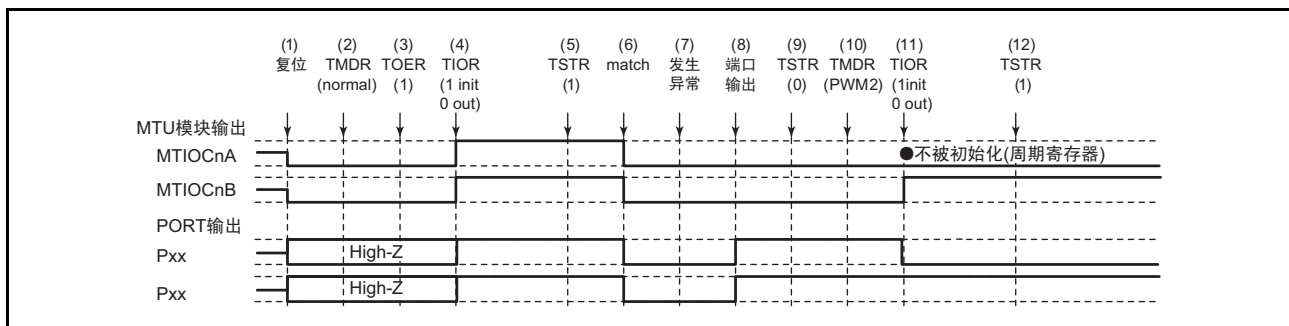


图 21.128 在正常模式中发生异常并且在 PWM 模式 2 中恢复的情况

(1) ~ (9) 和图 21.126 通用。

(10) 设定 PWM 模式 2。

(11) 必须通过 TIOR 对引脚进行初始化（在 PWM 模式 2 中不对周期寄存器的引脚进行初始化。如果要进行初始化，就必须在正常模式中进行初始化后转移到 PWM 模式 2）。

(12) 通过 TSTR 重新开始。

注. 只有 MTU0 ~ 2 能设定 PWM 模式 2，因此不需要设定 TOER。

(4) 在正常模式的运行过程中发生异常并且在相位计数模式中重新开始时的运行

在正常模式中发生异常并且重新设定后在相位计数模式中重新开始时的说明图如图 21.129 所示。

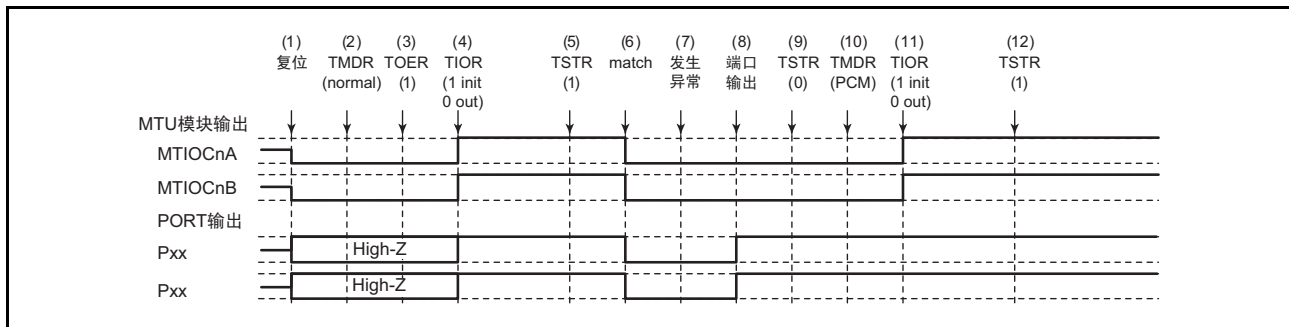


图 21.129 在正常模式中发生异常并且在相位计数模式中恢复的情况

- (1) ~ (9) 和图 21.126 通用。
- (10) 设定相位计数模式。
- (11) 必须通过 TIOR 对引脚进行初始化。
- (12) 通过 TSTR 重新开始。

注. 只有 MTU1 和 MTU2 能设定相位计数模式，因此不需要设定 TOER。

(5) 在正常模式的运行过程中发生异常并且在互补 PWM 模式中重新开始时的运行

在正常模式中发生异常并且重新设定后在互补 PWM 模式中重新开始时的说明图如图 21.130 所示。

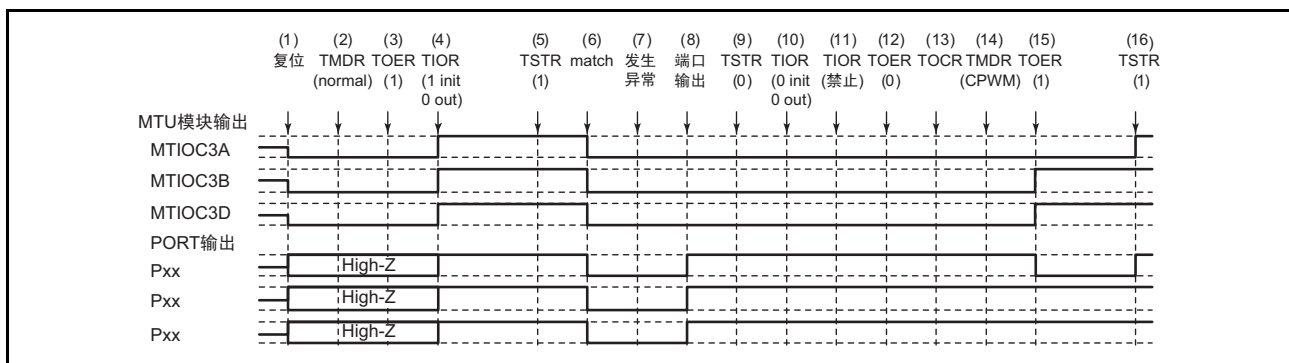


图 21.130 在正常模式中发生异常并且在互补 PWM 模式中恢复的情况

- (1) ~ (9) 和图 21.126 通用。
- (10) 必须通过 TIOR 对正常模式的波形生成部进行初始化。
- (11) 必须通过 TIOR 禁止正常模式的波形生成部的运行。
- (12) 必须通过 TOER 禁止 MTU3 和 MTU4 的输出。
- (13) 必须通过 TOCR 选择互补 PWM 的输出电平以及允许或者禁止周期输出。
- (14) 设定互补 PWM。
- (15) 必须通过 TOER 允许 MTU3 和 MTU4 的输出。
- (16) 通过 TSTR 重新开始。

(6) 在正常模式的运行过程中发生异常并且在复位同步 PWM 模式中重新开始时的运行

在正常模式中发生异常并且重新设定后在复位同步 PWM 模式中重新开始的说明图如图 21.131 所示。

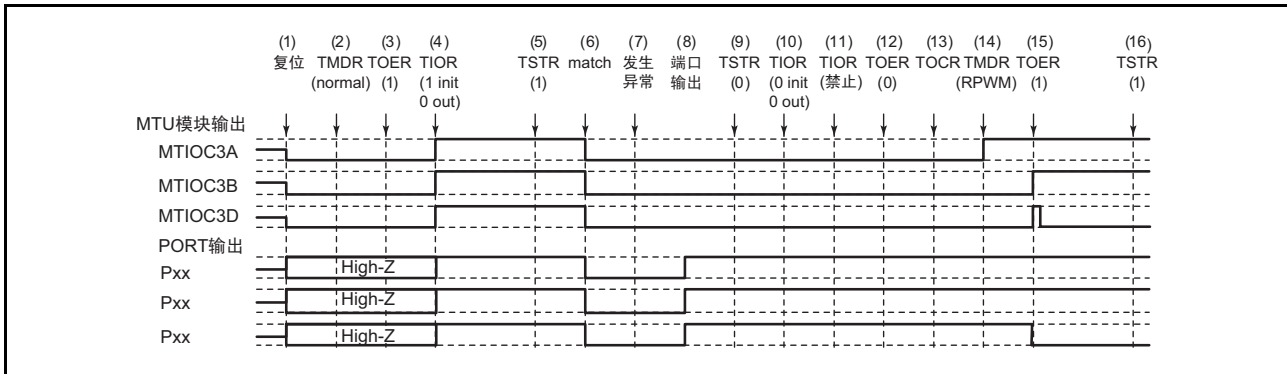


图 21.131 在正常模式中发生异常并且在复位同步 PWM 模式中恢复的情况

- (1) ~ (12) 和图 21.126 通用。
- (13) 必须通过 TOCR 选择复位同步 PWM 的输出电平以及允许或者禁止周期输出。
- (14) 设定复位同步 PWM。
- (15) 必须通过 TOER 允许 MTU3 和 MTU4 的输出。
- (16) 通过 TSTR 重新开始。

(7) 在 PWM 模式 1 的运行过程中发生异常并且在正常模式中重新开始时的运行

在 PWM 模式 1 中发生异常并且重新设定后在正常模式中重新开始的说明图如图 21.132 所示。

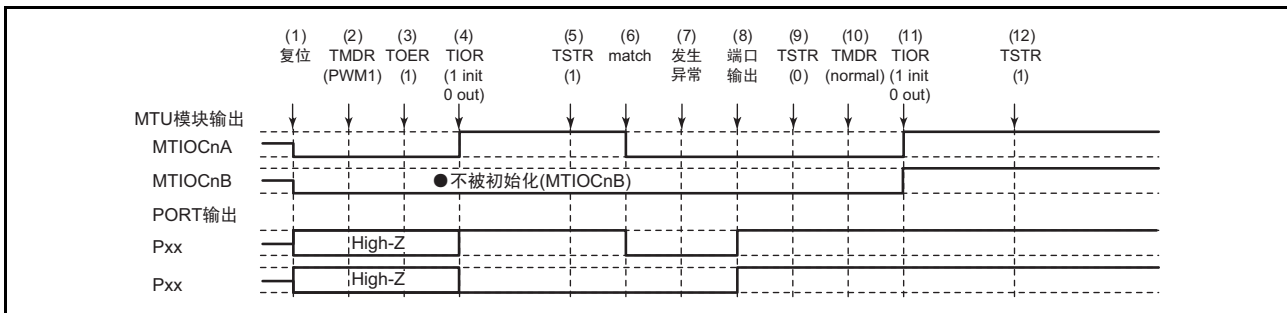


图 21.132 在 PWM 模式 1 中发生异常并且在正常模式中恢复的情况

- (1) 复位后，MTU 的输出为 Low 电平，端口为高阻抗。
- (2) 必须设定 PWM 模式 1。
- (3) 在 MTU3 和 MTU4 中，必须在通过 TIOR 对引脚进行初始化前用 TOER 允许输出。
- (4) 必须通过 TIOR 对引脚进行初始化（这是初始输出为 High 电平并且在比较匹配时输出 Low 电平的例子。在 PWM 模式 1 中不对 MTIOCnB 进行初始化）。
- (5) 通过 TSTR 开始计数运行。
- (6) 在比较匹配时输出 Low 电平。
- (7) 发生异常。
- (8) 必须通过 TIOR 禁止引脚输出，而作为端口输出，反相输出有效电平。
- (9) 通过 TSTR 停止计数运行。
- (10) 必须设定正常模式。
- (11) 必须通过 TIOR 对引脚进行初始化。
- (12) 通过 TSTR 重新开始。

(8) 在 PWM 模式 1 的运行过程中发生异常并且在 PWM 模式 1 中重新开始时的运行

在 PWM 模式 1 中发生异常并且重新设定后在 PWM 模式 1 中重新开始时的说明图如图 21.133 所示。

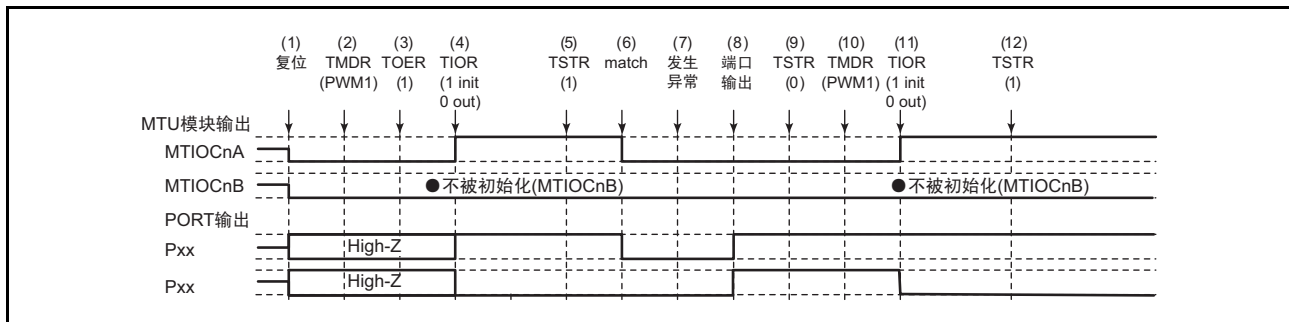


图 21.133 在 PWM 模式 1 中发生异常并且在 PWM 模式 1 中恢复的情况

(1) ~ (9) 和图 21.132 通用。

(10) 在 PWM 模式 1 中重新开始的情况下不需要此步骤。

(11) 必须通过 TIOR 对引脚进行初始化（在 PWM 模式 1 中不对 MTIOCnB 进行初始化）。

(12) 通过 TSTR 重新开始。

(9) 在 PWM 模式 1 的运行过程中发生异常并且在 PWM 模式 2 中重新开始时的运行

在 PWM 模式 1 中发生异常并且重新设定后在 PWM 模式 2 中重新开始时的说明图如图 21.134 所示。

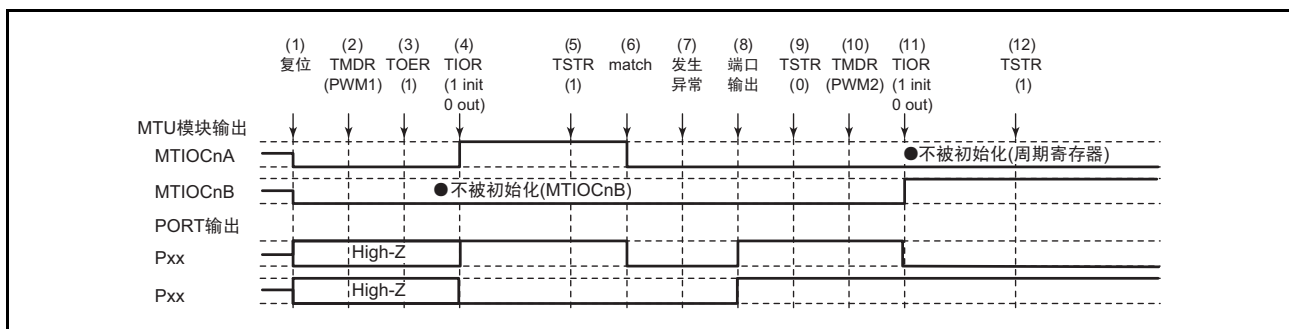


图 21.134 在 PWM 模式 1 中发生异常并且在 PWM 模式 2 中恢复的情况

(1) ~ (9) 和图 21.132 通用。

(10) 设定 PWM 模式 2。

(11) 必须通过 TIOR 对引脚进行初始化（在 PWM 模式 2 中不对周期寄存器的引脚进行初始化）。

(12) 通过 TSTR 重新开始。

注. 只有 MTU0 ~ 2 能设定 PWM 模式 2，因此不需要设定 TOER。

(10) 在 PWM 模式 1 的运行过程中发生异常并且在相位计数模式中重新开始时的运行

在 PWM 模式 1 中发生异常并且重新设定后在相位计数模式中重新开始时的说明图如图 21.135 所示。

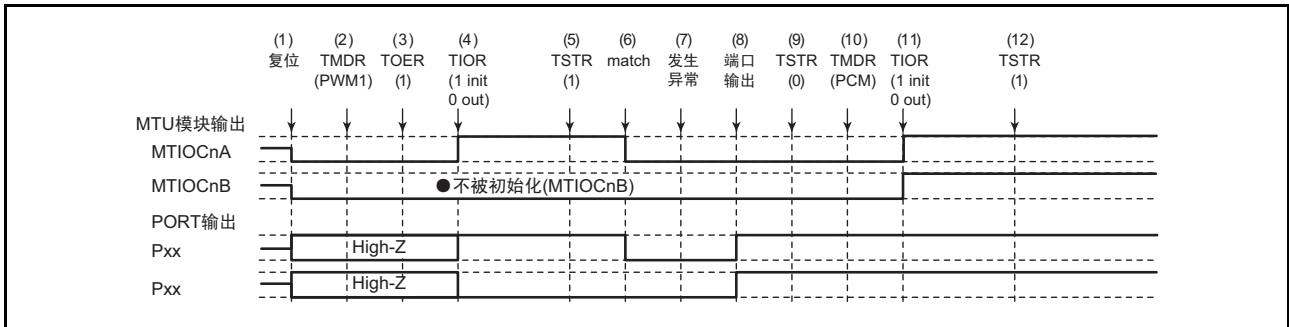


图 21.135 在 PWM 模式 1 中发生异常并且在相位计数模式中恢复的情况

- (1) ~ (9) 和图 21.132 通用。
- (10) 设定相位计数模式。
- (11) 必须通过 TIOR 对引脚进行初始化。
- (12) 通过 TSTR 重新开始。

注： 只有 MTU1 和 MTU2 能设定相位计数模式，因此不需要设定 TOER。

(11) 在 PWM 模式 1 的运行过程中发生异常并且在互补 PWM 模式中重新开始时的运行

在 PWM 模式 1 中发生异常并且重新设定后在互补 PWM 模式中重新开始时的说明图如图 21.136 所示。

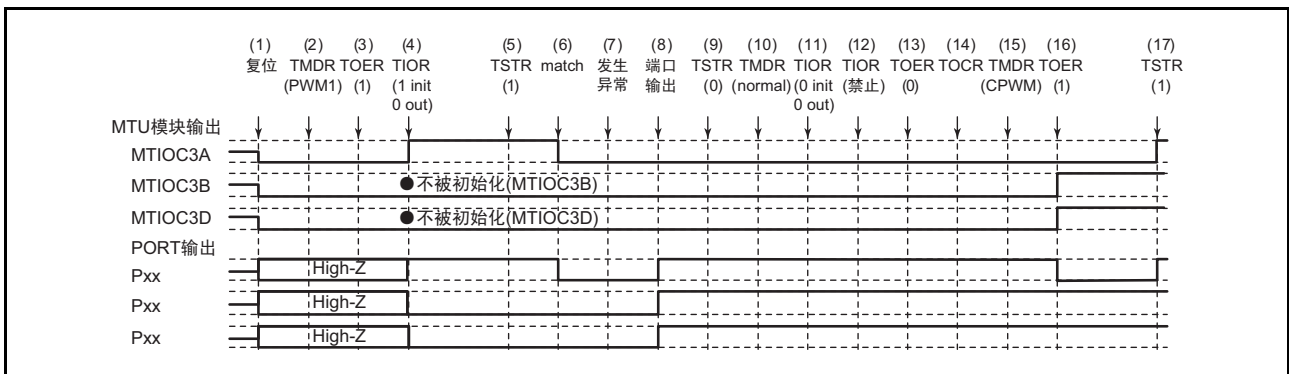


图 21.136 在 PWM 模式 1 中发生异常并且在互补 PWM 模式中恢复的情况

- (1) ~ (9) 和图 21.132 通用。
- (10) 为了对波形生成部进行初始化，必须设定正常模式。
- (11) 必须通过 TIOR 对 PWM 模式 1 的波形生成部进行初始化。
- (12) 必须通过 TIOR 禁止 PWM 模式 1 的波形生成部的运行。
- (13) 必须通过 TOER 禁止 MTU3 和 MTU4 的输出。
- (14) 必须通过 TOCR 选择互补 PWM 的输出电平以及允许或者禁止周期输出。
- (15) 设定互补 PWM。
- (16) 必须通过 TOER 允许 MTU3 和 MTU4 的输出。
- (17) 通过 TSTR 重新开始。

(12) 在 PWM 模式 1 的运行过程中发生异常并且在复位同步 PWM 模式中重新开始时的运行

在 PWM 模式 1 中发生异常并且重新设定后在复位同步 PWM 模式中重新开始时的说明图如图 21.137 所示。

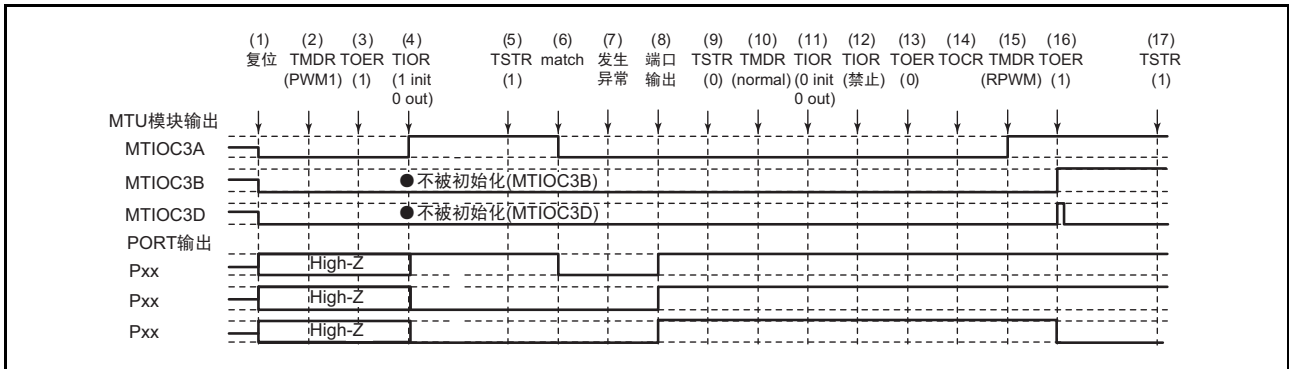


图 21.137 在 PWM 模式 1 中发生异常并且在复位同步 PWM 模式中恢复的情况

- (1) ~ (13) 和图 21.136 通用。
- (14) 必须通过 TOCR 选择复位同步 PWM 的输出电平以及允许或者禁止周期输出。
- (15) 设定复位同步 PWM。
- (16) 必须通过 TOER 允许 MTU3 和 MTU4 的输出。
- (17) 通过 TSTR 重新开始。

(13) 在 PWM 模式 2 的运行过程中发生异常并且在正常模式中重新开始时的运行

在 PWM 模式 2 中发生异常并且重新设定后在正常模式中重新开始时的说明图如图 21.138 所示。

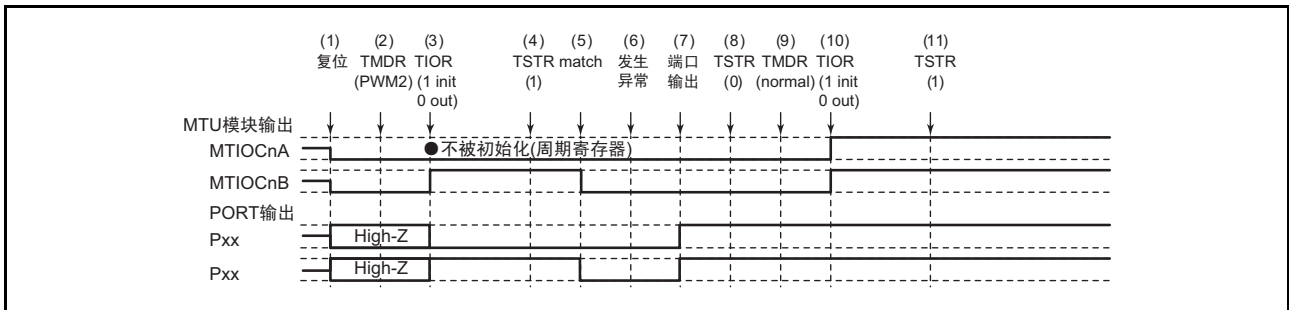


图 21.138 在 PWM 模式 2 中发生异常并且在正常模式中恢复的情况

- (1) 复位后，MTU 的输出为 Low 电平，端口为高阻抗。
- (2) 必须设定 PWM 模式 2。
- (3) 必须通过 TIOR 对引脚进行初始化（这是初始输出为 High 电平并且在比较匹配输出时 Low 电平的例子。在 PWM 模式 2 中不对周期寄存器的引脚进行初始化，这是 MTIOCnA 为周期寄存器的例子）。
- (4) 通过 TSTR 开始计数运行。
- (5) 在比较匹配时输出 Low 电平。
- (6) 发生异常。
- (7) 必须通过 TIOR 禁止引脚输出，而作为端口输出，反相输出有效电平。
- (8) 通过 TSTR 停止计数运行。
- (9) 必须设定正常模式。
- (10) 必须通过 TIOR 对引脚进行初始化。
- (11) 通过 TSTR 重新开始。

(14) 在 PWM 模式 2 的运行过程中发生异常并且在 PWM 模式 1 中重新开始时的运行

在 PWM 模式 2 中发生异常并且重新设定后在 PWM 模式 1 中重新开始时的说明图如图 21.139 所示。

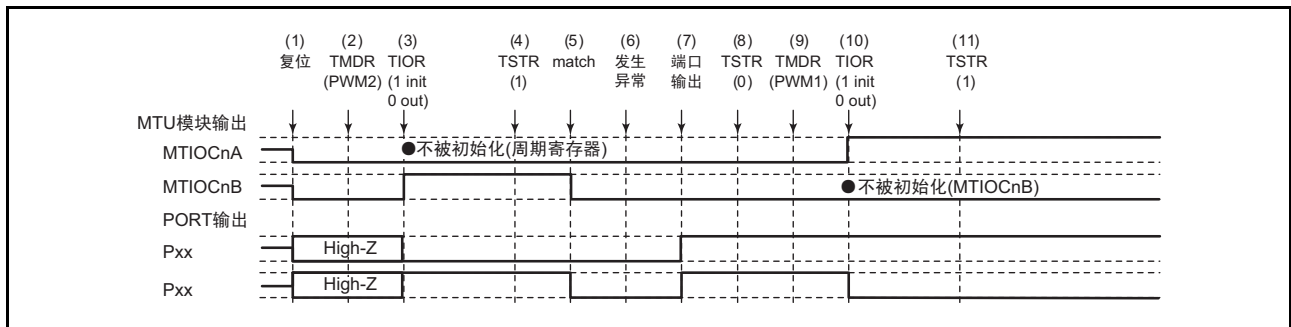


图 21.139 在 PWM 模式 2 中发生异常并且在 PWM 模式 1 中恢复的情况

- (1) ~ (8) 和图 21.138 通用。
- (9) 设定 PWM 模式 1。
- (10) 必须通过 TIOR 对引脚进行初始化 (在 PWM 模式 1 中不对 MTIOcNB 进行初始化)。
- (11) 通过 TSTR 重新开始。

(15) 在 PWM 模式 2 的运行过程中发生异常并且在 PWM 模式 2 中重新开始时的运行

在 PWM 模式 2 中发生异常并且重新设定后在 PWM 模式 2 中重新开始时的说明图如图 21.140 所示。

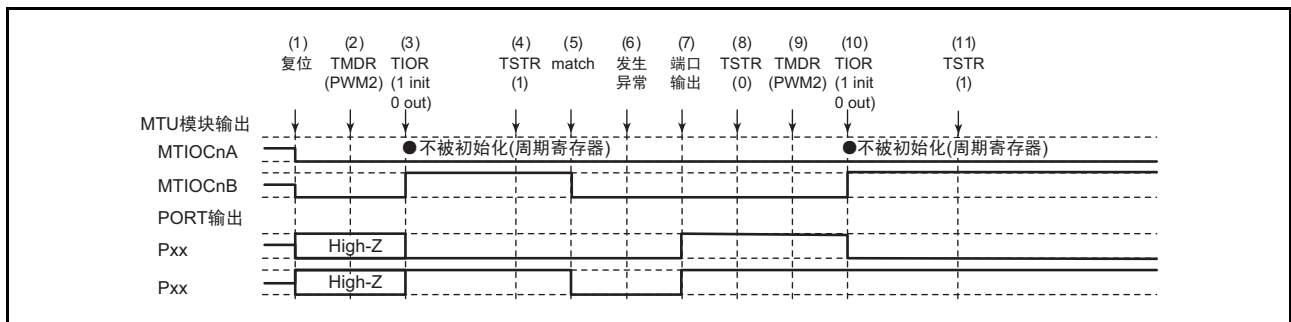


图 21.140 在 PWM 模式 2 中发生异常并且在 PWM 模式 2 中恢复的情况

- (1) ~ (8) 和图 21.138 通用。
- (9) 在 PWM 模式 2 中重新开始的情况下不需要此步骤。
- (10) 必须通过 TIOR 对引脚进行初始化 (在 PWM 模式 2 中不对周期寄存器的引脚进行初始化)。
- (11) 通过 TSTR 重新开始。

(16) 在 PWM 模式 2 的运行过程中发生异常并且在相位计数模式中重新开始时的运行

在 PWM 模式 2 中发生异常并且重新设定后在相位计数模式中重新开始时的说明图如图 21.141 所示。

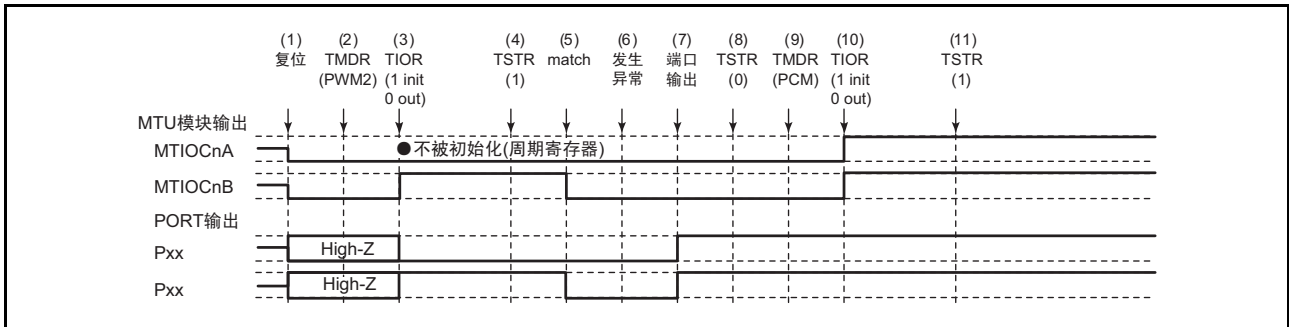


图 21.141 在 PWM 模式 2 中发生异常并且在相位计数模式中恢复的情况

- (1) ~ (8) 和图 21.138 通用。
- (9) 设定相位计数模式。
- (10) 必须通过 TIOR 对引脚进行初始化。
- (11) 通过 TSTR 重新开始。

(17) 在相位计数模式的运行过程中发生异常并且在正常模式中重新开始时的运行

在相位计数模式中发生异常并且重新设定后在正常模式中重新开始时的说明图如图 21.142 所示。

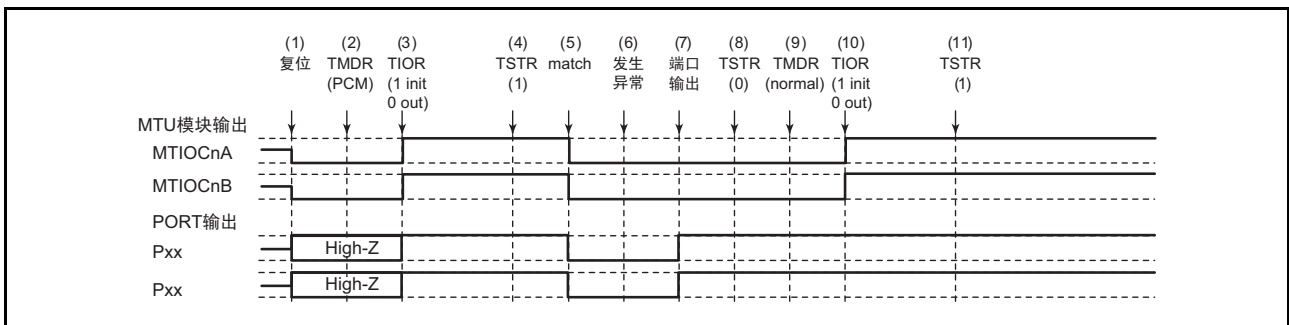


图 21.142 在相位计数模式中发生异常并且在正常模式中恢复的情况

- (1) 复位后，MTU 的输出为 Low 电平，端口为高阻抗。
- (2) 必须设定相位计数模式。
- (3) 必须通过 TIOR 对引脚进行初始化（这是初始输出为 High 电平并且在比较匹配时输出 Low 电平的例子）。
- (4) 通过 TSTR 开始计数运行。
- (5) 在比较匹配时输出 Low 电平。
- (6) 发生异常。
- (7) 必须作为端口输出，反相输出有效电平。
- (8) 通过 TSTR 停止计数运行。
- (9) 必须在正常模式中进行设定。
- (10) 必须通过 TIOR 对引脚进行初始化。
- (11) 通过 TSTR 重新开始。

(18) 在相位计数模式的运行过程中发生异常并且在 PWM 模式 1 中重新开始时的运行

在相位计数模式中发生异常并且重新设定后在 PWM 模式 1 中重新开始时的说明图如图 21.143 所示。

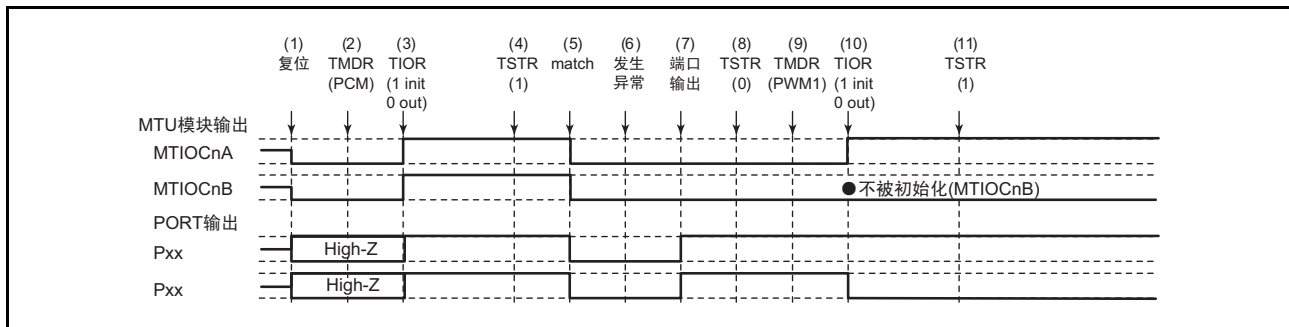


图 21.143 在相位计数模式中发生异常并且在 PWM 模式 1 中恢复的情况

- (1) ~ (8) 和图 21.142 通用。
- (9) 设定 PWM 模式 1。
- (10) 必须通过 TIOR 对引脚进行初始化（在 PWM 模式 1 中不对 MTIOcNB 进行初始化）。
- (11) 通过 TSTR 重新开始。

(19) 在相位计数模式的运行过程中发生异常并且在 PWM 模式 2 中重新开始时的运行

在相位计数模式中发生异常并且重新设定后在 PWM2 模式中重新开始时的说明图如图 21.144 所示。

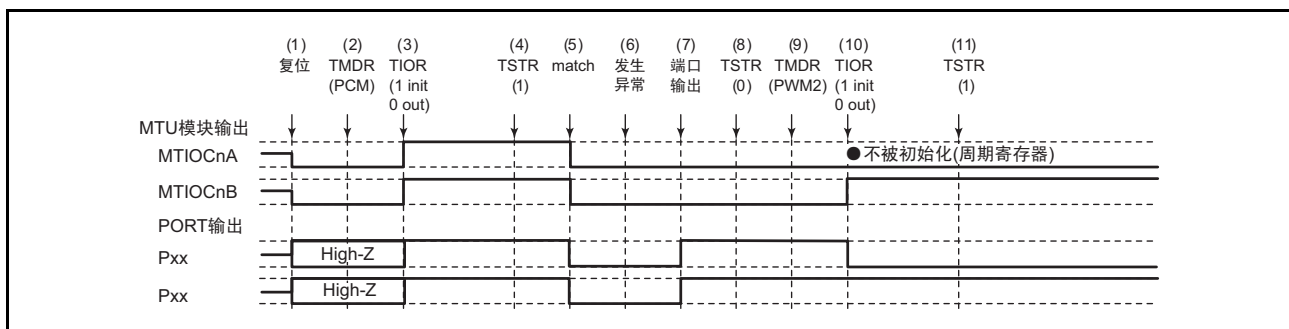


图 21.144 在相位计数模式中发生异常并且在 PWM 模式 2 中恢复的情况

- (1) ~ (8) 和图 21.142 通用。
- (9) 设定 PWM 模式 2。
- (10) 必须通过 TIOR 对引脚进行初始化（在 PWM 模式 2 中不对周期寄存器的引脚进行初始化）。
- (11) 通过 TSTR 重新开始。

(20) 在相位计数模式的运行过程中发生异常并且在相位计数模式中重新开始时的运行

在相位计数模式中发生异常并且重新设定后在相位计数模式中重新开始时的说明图如图 21.145 所示。

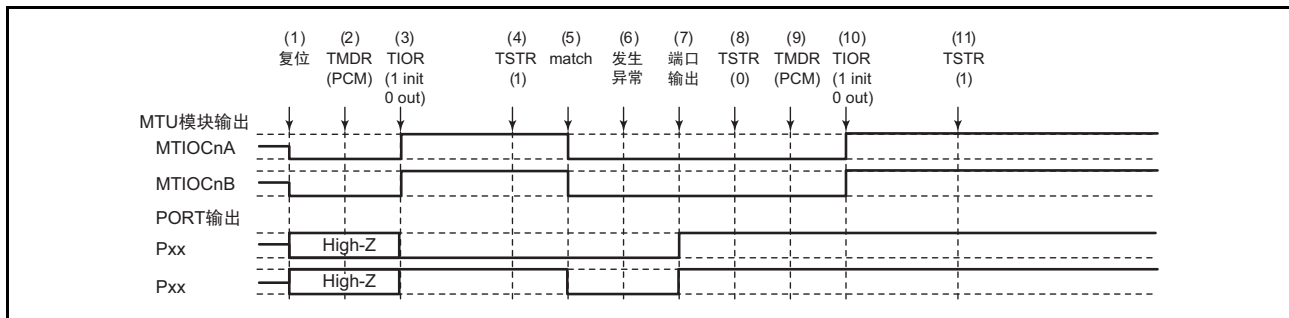


图 21.145 在相位计数模式中发生异常并且在相位计数模式中恢复的情况

- (1) ~ (8) 和图 21.142 通用。
- (9) 在相位计数模式重新开始的情况下不需要此步骤。
- (10) 必须通过 TIOR 对引脚进行初始化。
- (11) 通过 TSTR 重新开始。

(21) 在互补 PWM 模式的运行过程中发生异常并且在正常模式中重新开始时的运行

在互补 PWM 模式中发生异常并且重新设定后在正常模式中重新开始时的说明图如图 21.146 所示。

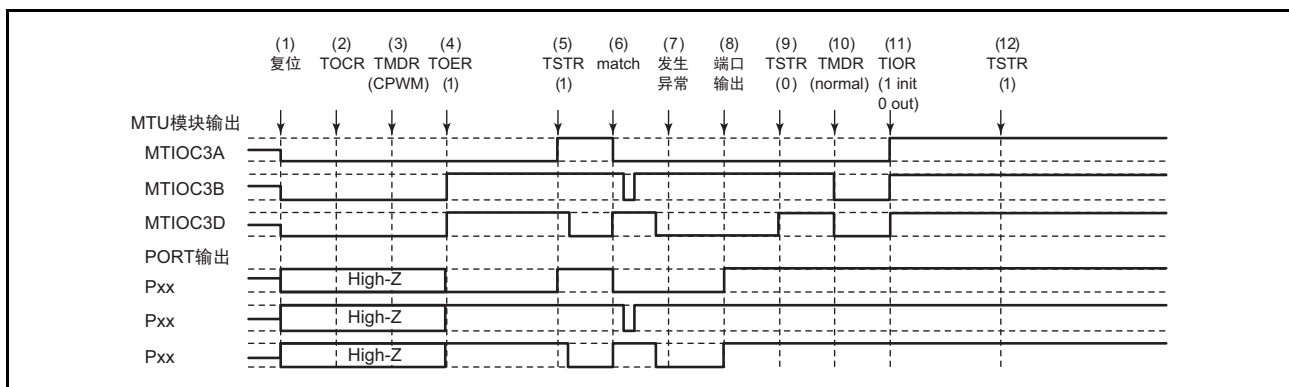


图 21.146 在互补 PWM 模式中发生异常并且在正常模式中恢复的情况

- (1) 复位后，MTU 的输出为 Low 电平，端口为高阻抗。
- (2) 必须通过 TOCR 选择互补 PWM 的输出电平以及允许或者禁止周期输出。
- (3) 设定互补 PWM。
- (4) 必须通过 TOER 允许 MTU3 和 MTU4 的输出。
- (5) 通过 TSTR 开始计数运行。
- (6) 在比较匹配时输出互补 PWM 波形。
- (7) 发生异常。
- (8) 必须通过 TIOR 禁止引脚输出，而作为端口输出，反相输出有效电平。
- (9) 通过 TSTR 停止计数运行（MTU 输出为互补 PWM 输出的初始值）。
- (10) 必须设定正常模式（MTU 的输出为 Low 电平）。
- (11) 必须通过 TIOR 对引脚进行初始化。
- (12) 通过 TSTR 重新开始。

(22) 在互补 PWM 模式的运行过程中发生异常并且在 PWM 模式 1 中重新开始时的运行

在互补 PWM 模式中发生异常并且重新设定后在 PWM 模式 1 中重新开始时的说明图如图 21.147 所示。

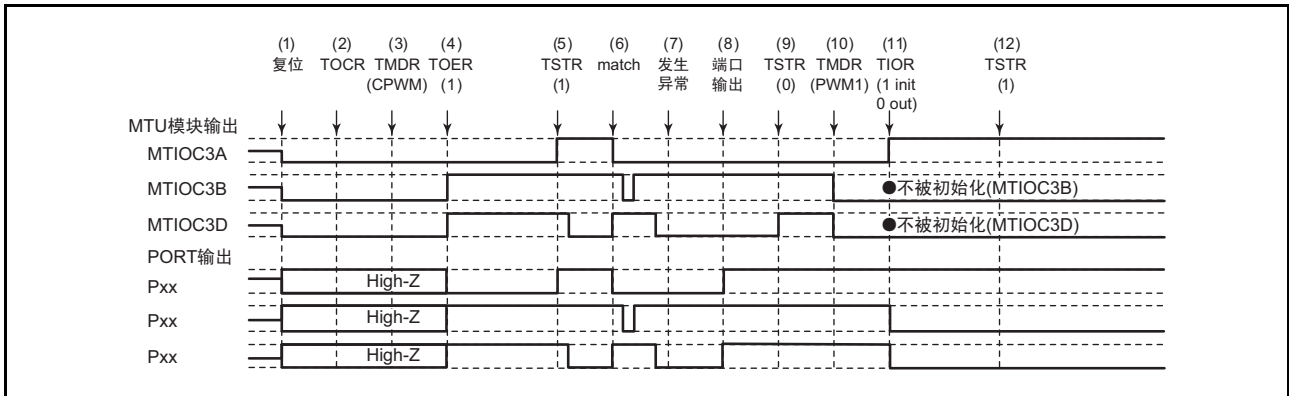


图 21.147 在互补 PWM 模式中发生异常并且在 PWM 模式 1 中恢复的情况

- (1) ~ (9) 和图 21.146 通用。
- (10) 必须设定 PWM 模式 1 (MTU 的输出为 Low 电平)。
- (11) 必须通过 TIOC3A 对引脚进行初始化 (在 PWM 模式 1 中不对 MTIOCnB 进行初始化)。
- (12) 通过 TSTR 重新开始。

(23) 在互补 PWM 模式的运行过程中发生异常并且在互补 PWM 模式中重新开始时的运行

在互补 PWM 模式中发生异常并且重新设定后在互补 PWM 模式中重新开始时的说明图如图 21.148 所示 (从停止对周期和占空比进行计数时的值重新开始的情况)。

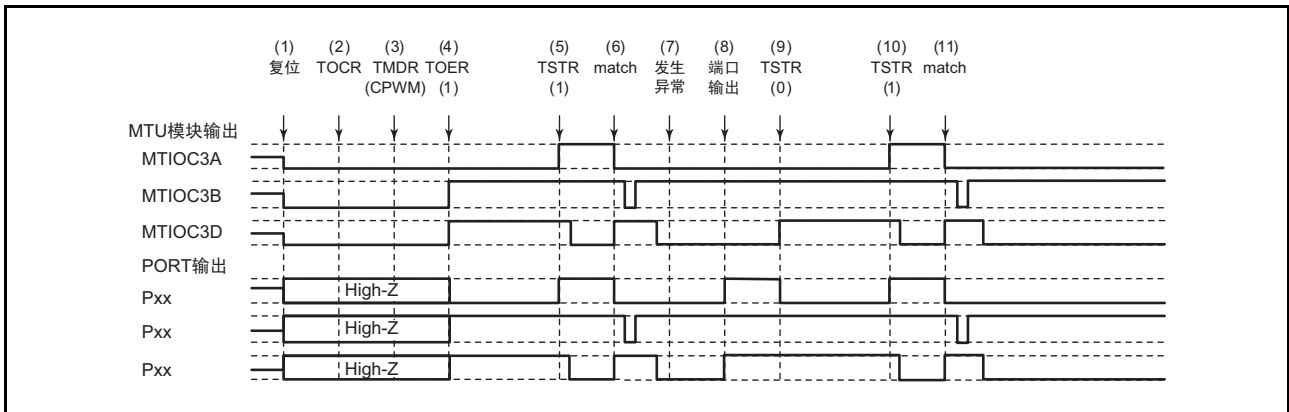


图 21.148 在互补 PWM 模式中发生异常并且在互补 PWM 模式中恢复的情况

- (1) ~ (9) 和图 21.146 通用。
- (10) 通过 TSTR 重新开始。
- (11) 在比较匹配时输出互补 PWM 波形。

(24) 在互补 PWM 模式的运行过程中发生异常并且在互补 PWM 模式中重新开始时的运行

在互补 PWM 模式中发生异常并且重新设定后在互补 PWM 模式中重新开始时的说明图如图 21.149 所示（从重新设定的周期和占空比的值重新开始的情况）。

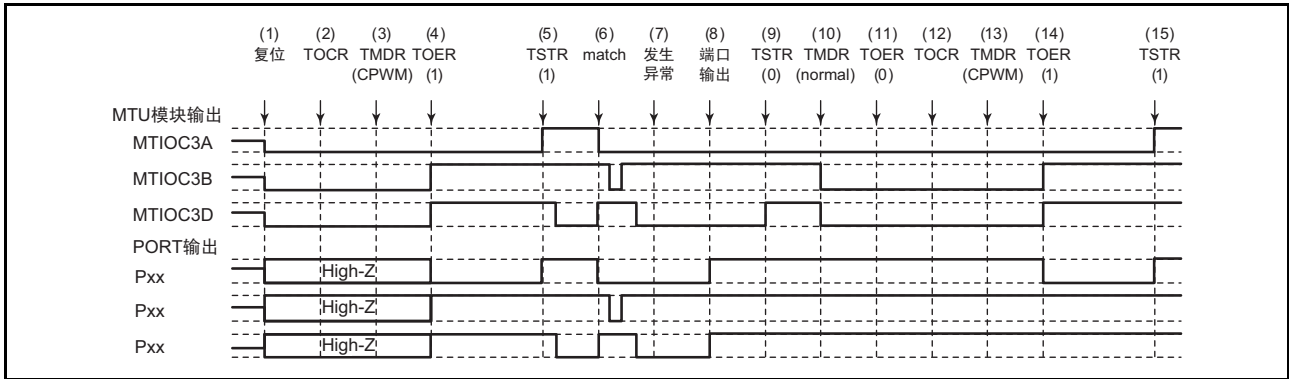


图 21.149 在互补 PWM 模式中发生异常并且在互补 PWM 模式中恢复的情况

- (1) ~ (9) 和图 21.146 通用。
- (10) 必须设定正常模式并且设定新的设定值（MTU 的输出为 Low 电平）。
- (11) 必须通过 TOER 禁止 MTU3 和 MTU4 的输出。
- (12) 必须通过 TOCR 选择互补 PWM 模式的输出电平以及允许或者禁止周期输出。
- (13) 设定互补 PWM。
- (14) 必须通过 TOER 允许 MTU3 和 MTU4 的输出。
- (15) 通过 TSTR 重新开始。

(25) 在互补 PWM 模式的运行过程中发生异常并且在复位同步 PWM 模式中重新开始时的运行

在互补 PWM 模式中发生异常并且重新设定后在复位同步 PWM 模式中重新开始时的说明图如图 21.150 所示。

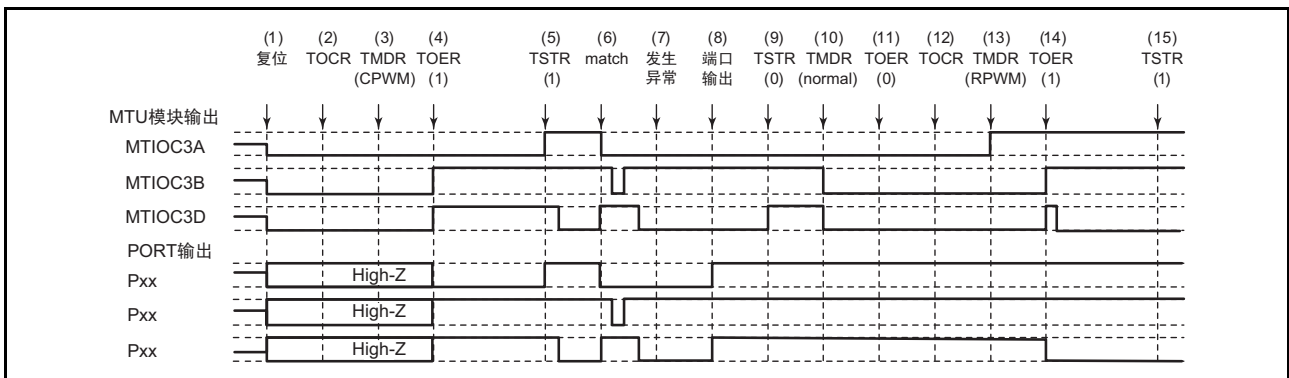


图 21.150 在互补 PWM 模式中发生异常并且在复位同步 PWM 模式中恢复的情况

- (1) ~ (9) 和图 21.146 通用。
- (10) 必须设定正常模式（MTU 的输出为 Low 电平）。
- (11) 必须通过 TOER 禁止 MTU3 和 MTU4 的输出。
- (12) 必须通过 TOCR 选择复位同步 PWM 模式的输出电平以及允许或者禁止周期输出。
- (13) 设定复位同步 PWM。
- (14) 必须通过 TOER 允许 MTU3 和 MTU4 的输出。
- (15) 通过 TSTR 重新开始。

(26) 在复位同步 PWM 模式的运行过程中发生异常并且在正常模式中重新开始时的运行

在复位同步 PWM 模式中发生异常并且重新设定后在正常模式中重新开始时的说明图如图 21.151 所示。

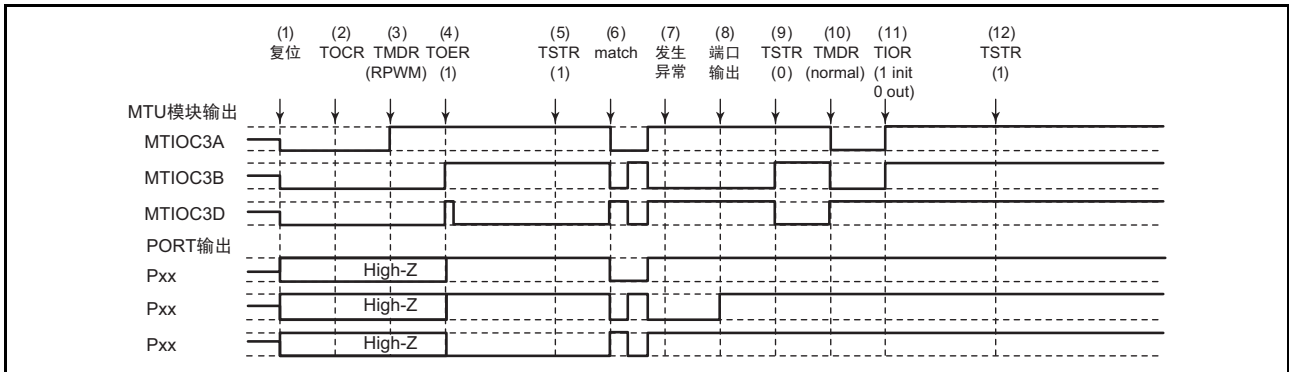


图 21.151 在复位同步 PWM 模式中发生异常并且在正常模式中恢复的情况

- (1) 复位后，MTU 的输出为 Low 电平，端口为高阻抗。
- (2) 必须通过 TOCR 选择复位同步 PWM 的输出电平以及允许或者禁止周期输出。
- (3) 设定复位同步 PWM。
- (4) 必须通过 TOER 允许 MTU3 和 MTU4 的输出。
- (5) 通过 TSTR 开始计数运行。
- (6) 在比较匹配时输出复位同步 PWM 波形。
- (7) 发生异常。
- (8) 必须通过 TIOR 禁止引脚输出，而作为端口输出，反相输出有效电平。
- (9) 通过 TSTR 停止计数运行 (MTU 的输出为复位同步 PWM 输出的初始值)。
- (10) 必须设定正常模式 (MTU 输出的正相侧为 Low 电平，反相侧为 High 电平)。
- (11) 必须通过 TIOR 对引脚进行初始化。
- (12) 通过 TSTR 重新开始。

(27) 在复位同步 PWM 模式的运行过程中发生异常并且在 PWM 模式 1 中重新开始时的运行

在复位同步 PWM 模式中发生异常并且重新设定后在 PWM 模式 1 中重新开始时的说明图如图 21.152 所示。

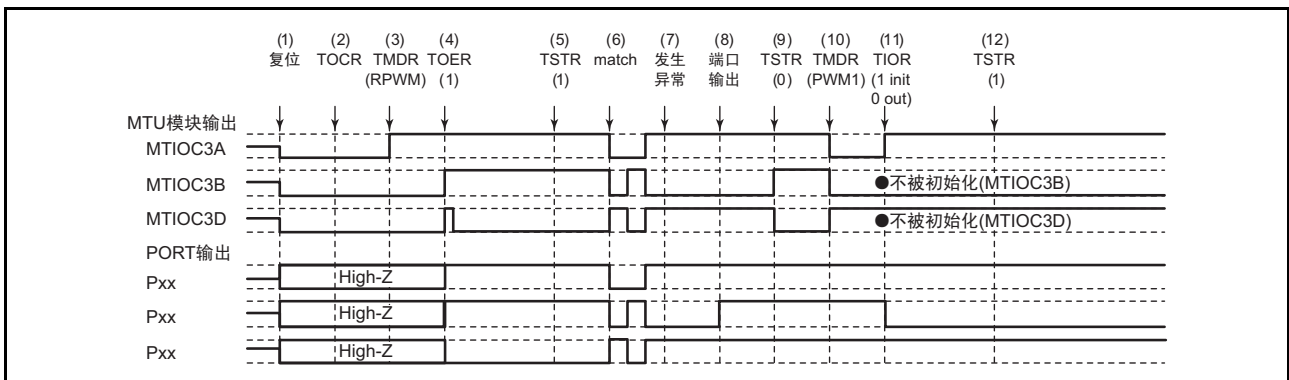


图 21.152 在复位同步 PWM 模式中发生异常并且在 PWM 模式 1 中恢复的情况

- (1) ~ (9) 和图 21.151 通用。
- (10) 必须设定 PWM 模式 1 (MTU 输出的正相侧为 Low 电平，反相侧为 High 电平)。
- (11) 必须通过 TIOR 对引脚进行初始化 (在 PWM 模式 1 中不对 MTIOC3B 进行初始化)。
- (12) 通过 TSTR 重新开始。

(28) 在复位同步 PWM 模式的运行过程中发生异常并且在互补 PWM 模式中重新开始时的运行

在复位同步 PWM 模式中发生异常并且重新设定后在互补 PWM 模式中重新开始时的说明图如图 21.153 所示。

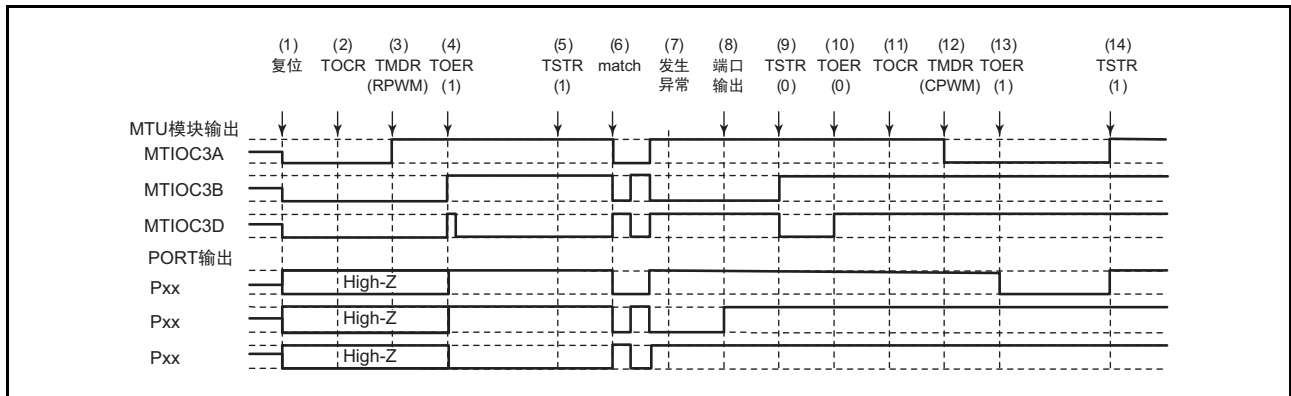


图 21.153 在复位同步 PWM 模式中发生异常并且在互补 PWM 模式中恢复的情况

- (1) ~ (9) 和图 21.151 通用。
- (10) 必须通过 TOER 禁止 MTU3 和 MTU4 的输出。
- (11) 必须通过 TOCR 选择互补 PWM 的输出电平以及允许或者禁止周期输出。
- (12) 设定互补 PWM (MTU 的周期输出引脚为 Low 电平)。
- (13) 必须通过 TOER 允许 MTU3 和 MTU4 的输出。
- (14) 通过 TSTR 重新开始。

(29) 在复位同步 PWM 模式的运行过程中发生异常并且在复位同步 PWM 模式中重新开始时的运行

在复位同步 PWM 模式中发生异常并且重新设定后在复位同步 PWM 模式中重新开始时的说明图如图 21.154 所示。

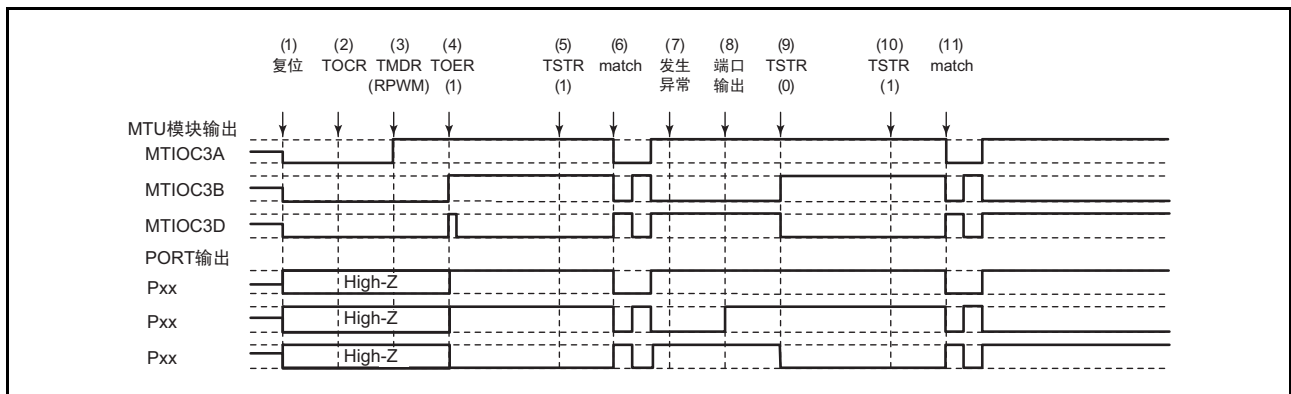


图 21.154 在复位同步 PWM 模式中发生异常并且在复位同步 PWM 模式中恢复的情况

- (1) ~ (9) 和图 21.151 通用。
- (10) 通过 TSTR 重新开始。
- (11) 在比较匹配时输出复位同步 PWM 波形。

21.8 通过 ELC 进行的链接运行

21.8.1 向 ELC 输出事件信号

通过事件链接控制器 (ELC)，MTU 将中断请求信号用作事件信号，并且对事先设定的模块进行链接运行。

与对应的中断请求允许位的设定无关，能输出事件信号。

21.8.2 通过接收 ELC 的事件信号进行的 MTU2 运行

MTU 能通过事先设定的事件（根据 ELC 的 ELSRn 寄存器的设定）进行以下运行。

(1) 开始计数运行

通过 ELC 的 ELOPA 寄存器和 ELOPB 寄存器选择开始 MTU 的计数运行。ELOPA 寄存器用于通道 1 ~ 通道 3，ELOPB 寄存器用于通道 4。必须将设定 MTU 的通道运行模式的 TMDR 寄存器设定为复位后的值 (“00h”)。如果发生 ELSRn 寄存器指定的事件，表 21.60 中所示的 TSTR.CSTn 位就变为 “1”，并且开始 MTU 的计数运行。

但是，如果在 TSTR.CSTn 位为 “1” 的状态下发生指定的事件，该事件就变为无效。对各通道使用的 TSTR 寄存器的位名如表 21.60 所示。

有关开始计数运行的设定步骤的详细内容，请参照 “21.3.1(1) 计数器的运行”。

表 21.60 与 ELC 进行链接运行的定时器的启动寄存器

通道号	定时器的启动寄存器
通道 1	TSTR.CST1 位
通道 2	TSTR.CST2 位
通道 3	TSTR.CST3 位
通道 4	TSTR.CST4 位

(2) 输入捕捉运行

通过 ELC 的 ELOPA 寄存器和 ELOPB 寄存器选择 MTU2 的输入捕捉运行。ELOPA 寄存器用于通道 1 ~ 通道 3，ELOPB 寄存器用于通道 4。必须将设定 MTU 的通道运行模式的 TMDR 寄存器设定为复位后的值 (“00h”)。如果发生 ELSRn 寄存器指定的事件，就将 TCNT 计数器的值捕捉到 TGR 寄存器。如果使用通过事件链接进行的输入捕捉运行，就将 MTU 的 TIOR 寄存器的位设定为输入捕捉，并且将 TSTR.CSTn 位置 “1”，开始计数器的计数运行。

此时，TIOcNA 引脚（输入捕捉引脚）的输入变为无效。

对各通道使用的 TGR 寄存器和 TIOR 寄存器的位名如表 21.61 所示。

有关输入捕捉的设定步骤的详细内容，请参照 “21.3.1(3) 输入捕捉功能”。

表 21.61 在通过 ELC 进行的输入捕捉运行中使用的各通道的定时器的通用寄存器和定时器的 I/O 控制寄存器

通道号	寄存器名	TIOR 寄存器的位名
通道 1	TGRA 寄存器	TIOR.IOA[3:0] 位
通道 2	TGRA 寄存器	TIOR.IOA[3:0] 位
通道 3	TGRA 寄存器	TIORH.IOA[3:0] 位
通道 4	TGRA 寄存器	TIORH.IOA[3:0] 位

(3) 重新开始计数运行

通过 ELC 的 ELOPA 寄存器和 ELOPB 寄存器选择重新开始 MTU2 的计数运行。ELOPA 寄存器用于通道 1~通道 3，ELOPB 寄存器用于通道 4。必须将设定 MTU 的通道运行模式的 TMDR 寄存器设定为复位后的值 (“00h”)。如果发生 ELSRn 寄存器指定的事件，TCNT (定时器的计数器) 的值就被改写为初始值。如果将 TSTR 寄存器的 CSTn 位置 “1”，就能继续计数运行。对应的 TSTR.CSTn 位请参照表 21.60。

21.8.3 通过接收 ELC 的事件信号进行的 MTU2 运行的注意事项

通过事件链接进行的 MTU2 运行的注意事项如下所示。

(1) 开始计数运行

如果在写 TSTR.CSTn 位的过程中发生 ELSRn 寄存器指定的事件，就不写 TSTR.CSTn 位，而优先通过发生的事件将 TSTR.CSTn 位置 “1”。

(2) 重新开始计数运行

如果在写 TCNT 计数器的过程中发生 ELSRn 寄存器指定的事件，就不写 TCNT 计数器，而优先通过发生的事件对计数值进行初始化。

22. 端口输出允许 2 (POE2a)

根据 POE0# ~ POE3# 引脚和 POE8# 引脚的输入变化、MTU 互补 PWM 输出引脚 (MTIOC3B、MTIOC3D、MTIOC4A、MTIOC4B、MTIOC4C、MTIOC4D) 的输出状态、时钟发生电路的振荡停止检测、寄存器的设定 (SPOER 寄存器) 或者事件链接控制器 (ELC) 的事件信号输入, 端口输出允许 2 (POE) 能将 MTU 的互补 PWM 输出引脚和 MTU0 的输出引脚 (MTIOC0A、MTIOC0B、MTIOC0C、MTIOC0D) 置为高阻抗。

另外, 能同时发行中断请求。

22.1 概要

POE 的规格和框图分别如表 22.1 和图 22.1 所示。

表 22.1 POE 的规格

项目	内容
通过输入电平检测进行的高阻抗控制	<ul style="list-style-type: none"> 能给 POE0# ~ POE3# 和 POE8# 的各输入引脚设定下降沿、PCLK 的 8 分频 ×16 次、PCLK 的 16 分频 ×16 次、PCLK 的 128 分频 ×16 次的 Low 电平采样。 能通过 POE0# ~ POE3# 引脚的下降沿或者 Low 电平采样, 将 MTU 的互补 PWM 输出引脚置为高阻抗。 能通过 POE8# 引脚的下降沿或者 Low 电平采样, 将 MTU0 的输出引脚置为高阻抗。
通过输出电平比较进行的高阻抗控制	<ul style="list-style-type: none"> 如果在将 MTU 互补 PWM 输出引脚的输出电平进行比较的同时, 有效电平至少持续输出了 1 个周期, 就能将 MTU 的互补 PWM 输出引脚置为高阻抗。
通过振荡停止检测进行的高阻抗控制	<ul style="list-style-type: none"> 如果时钟发生电路停止振荡, 就能将 MTU 的互补 PWM 输出引脚和 MTU0 的输出引脚置为高阻抗。
通过软件 (寄存器) 进行的高阻抗控制	<ul style="list-style-type: none"> 能通过写 POE 的寄存器, 将 MTU 的互补 PWM 输出引脚和 MTU0 的输出引脚置为高阻抗。
通过事件信号进行的高阻抗控制	<ul style="list-style-type: none"> 能通过事件链接控制器 (ELC) 的事件信号, 将 MTU 的互补 PWM 输出引脚和 MTU0 的输出引脚置为高阻抗。
中断	<ul style="list-style-type: none"> 根据 POE0# ~ POE3# 和 POE8# 的输入电平检测结果或者 MTU 的互补 PWM 输出引脚的输出电平比较结果, 产生各种中断。

如图 22.1 的框图所示, POE 由输入电平检测电路、输出电平比较电路、时钟发生电路的振荡停止检测信号的输入和高阻抗请求 / 中断请求生成电路构成。

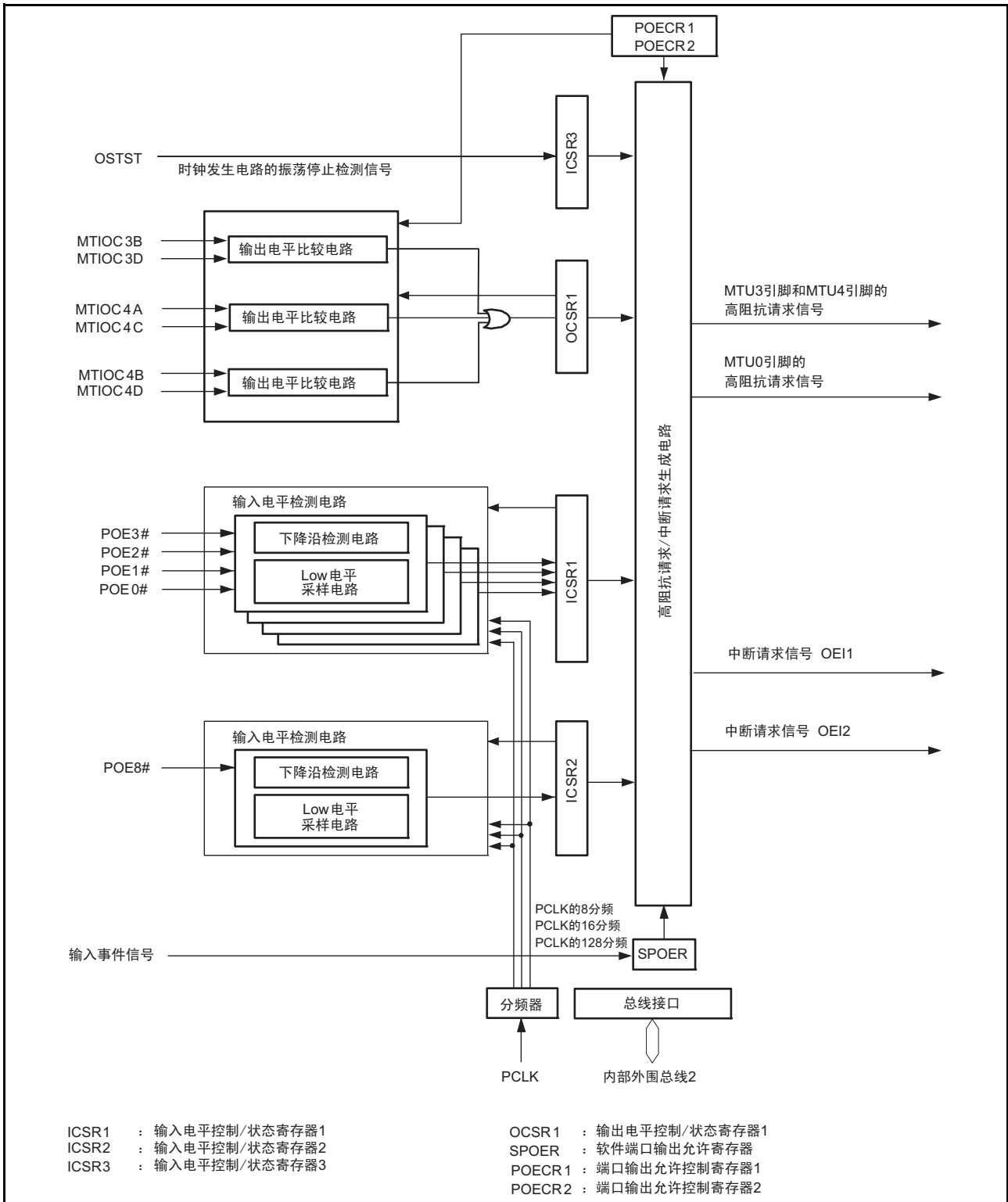


图 22.1 POE 的框图

POE 使用的输入 / 输出引脚如表 22.2 所示。

表 22.2 POE 的输入 / 输出引脚

引脚名	输入 / 输出	功能
POE0# ~ POE3#	输入	将 MTU 的互补 PWM 输出引脚置为高阻抗的请求信号
POE8#	输入	将 MTU0 的引脚置为高阻抗的请求信号
MTIOC3B	输出	MTU3 的互补 PWM 输出引脚
MTIOC3D	输出	MTU3 的互补 PWM 输出引脚
MTIOC4A	输出	MTU4 的互补 PWM 输出引脚
MTIOC4B	输出	MTU4 的互补 PWM 输出引脚
MTIOC4C	输出	MTU4 的互补 PWM 输出引脚
MTIOC4D	输出	MTU4 的互补 PWM 输出引脚
MTIOC0A	输出	MTU0 的输出引脚
MTIOC0B	输出	MTU0 的输出引脚
MTIOC0C	输出	MTU0 的输出引脚
MTIOC0D	输出	MTU0 的输出引脚

通过表 22.3 所示的引脚组合进行输出电平的比较。

表 22.3 引脚组合

引脚组合	输入 / 输出	功能
MTIOC3B 和 MTIOC3D	输出	能通过 POE 的寄存器设定对哪个组合进行输出电平比较和高阻抗控制。当有效电平至少同时持续输出 1 个 PCLK 周期时，将 MTU 的互补 PWM 输出引脚置为高阻抗。 (在 MTU.TOCR1.TOCS 位为“0”的情况下，MTU.TOCR1.OLSP 位、MTU.TOCR1.OLSN 位是“0”时为 Low 电平输出，是“1”时为 High 电平输出。 在 MTU.TOCR1.TOCS 位为“1”的情况下，MTU.TOCR2.OLS3N 位、MTU.TOCR2.OLS3P 位、MTU.TOCR2.OLS2N 位、MTU.TOCR2.OLS2P 位、MTU.TOCR2.OLS1N 或者 MTU.TOCR2.OLS1P 位是“0”时为 Low 电平输出，是“1”时为 High 电平输出)。
MTIOC4A 和 MTIOC4C	输出	
MTIOC4B 和 MTIOC4D	输出	

22.2 寄存器说明

22.2.1 输入电平控制 / 状态寄存器 1 (ICSR1)

地址 0008 8900h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
POE3F	POE2F	POE1F	POE0F	—	—	—	PIE1	POE3M[1:0]	POE2M[1:0]	POE1M[1:0]	POE0M[1:0]				
复位后的值	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b1-b0	POE0M[1:0]	POE0 模式选择位	b1 b0 0 0: 在 POE0# 输入的下降沿接受请求 0 1: 以每个 PCLK 的 8 分频时钟对 POE0# 输入的 Low 电平进行 16 次采样, 当全部为 Low 电平时, 接受请求。 1 0: 以每个 PCLK 的 16 分频时钟对 POE0# 输入的 Low 电平进行 16 次采样, 当全部为 Low 电平时, 接受请求。 1 1: 以每个 PCLK 的 128 分频时钟对 POE0# 输入的 Low 电平进行 16 次采样, 当全部为 Low 电平时, 接受请求。	R/W (注 1)
b3-b2	POE1M[1:0]	POE1 模式选择位	b3 b2 0 0: 在 POE1# 输入的下降沿接受请求 0 1: 以每个 PCLK 的 8 分频时钟对 POE1# 输入的 Low 电平进行 16 次采样, 当全部为 Low 电平时, 接受请求。 1 0: 以每个 PCLK 的 16 分频时钟对 POE1# 输入的 Low 电平进行 16 次采样, 当全部为 Low 电平时, 接受请求。 1 1: 以每个 PCLK 的 128 分频时钟对 POE1# 输入的 Low 电平进行 16 次采样, 当全部为 Low 电平时, 接受请求。	R/W (注 1)
b5-b4	POE2M[1:0]	POE2 模式选择位	b5 b4 0 0: 在 POE2# 输入的下降沿接受请求 0 1: 以每个 PCLK 的 8 分频时钟对 POE2# 输入的 Low 电平进行 16 次采样, 当全部为 Low 电平时, 接受请求。 1 0: 以每个 PCLK 的 16 分频时钟对 POE2# 输入的 Low 电平进行 16 次采样, 当全部为 Low 电平时, 接受请求。 1 1: 以每个 PCLK 的 128 分频时钟对 POE2# 输入的 Low 电平进行 16 次采样, 当全部为 Low 电平时, 接受请求。	R/W (注 1)
b7-b6	POE3M[1:0]	POE3 模式选择位	b7 b6 0 0: 在 POE3# 输入的下降沿接受请求 0 1: 以每个 PCLK 的 8 分频时钟对 POE3# 输入的 Low 电平进行 16 次采样, 当全部为 Low 电平时, 接受请求。 1 0: 以每个 PCLK 的 16 分频时钟对 POE3# 输入的 Low 电平进行 16 次采样, 当全部为 Low 电平时, 接受请求。 1 1: 以每个 PCLK 的 128 分频时钟对 POE3# 输入的 Low 电平进行 16 次采样, 当全部为 Low 电平时, 接受请求。	R/W (注 1)
b8	PIE1	端口中断允许 1 位	0: 禁止通过输入电平的检测进行 OEI1 中断请求 1: 允许通过输入电平的检测进行 OEI1 中断请求	R/W
b11-b9	—	保留位	读写值都为“0”。	R/W
b12	POE0F	POE0 标志	0: POE0# 引脚无高阻抗请求 1: POE0# 引脚有高阻抗请求	R/(W) (注 2)
b13	POE1F	POE1 标志	0: POE1# 引脚无高阻抗请求 1: POE1# 引脚有高阻抗请求	R/(W) (注 2)
b14	POE2F	POE2 标志	0: POE2# 引脚无高阻抗请求 1: POE2# 引脚有高阻抗请求	R/(W) (注 2)
b15	POE3F	POE3 标志	0: POE3# 引脚无高阻抗请求 1: POE3# 引脚有高阻抗请求	R/(W) (注 2)

注 1. 复位后, 只能写 1 次。

注 2. 要将此标志置“0”时, 只能在读“1”后写“0”。

在通过 POE0M[1:0] ~ POE3M[1:0] 位设定了 Low 电平采样的情况下要给 POE0F ~ POE3F 标志写“0”时，需要将 High 电平输入到 POE0# ~ POE3# 引脚。

详细内容请参照“22.3.6 高阻抗的解除”。

POE0M[1:0] 位 (POE0 模式选择位)

这些位选择 POE0# 引脚的输入模式。

POE1M[1:0] 位 (POE1 模式选择位)

这些位选择 POE1# 引脚的输入模式。

POE2M[1:0] 位 (POE2 模式选择位)

这些位选择 POE2# 引脚的输入模式。

POE3M[1:0] 位 (POE3 模式选择位)

这些位选择 POE3# 引脚的输入模式。

PIE1 位 (端口中断允许 1 位)

当 POE3F ~ POE0F 标志中只要有 1 位为“1”时，此位指定是否请求 OEI1 中断。

POE0F 标志 (POE0 标志)

这是表示给 POE0# 引脚输入了高阻抗请求的标志。

[为“0”的条件]

- 在读“1”后写“0”时

[为“1”的条件]

- 当 POE0# 引脚有 POE0M[1:0] 位设定的输入时

POE1F 标志 (POE1 标志)

这是表示给 POE1# 引脚输入了高阻抗请求的标志。

[为“0”的条件]

- 在读“1”后写“0”时

[为“1”的条件]

- 当 POE1# 引脚有 POE1M[1:0] 位设定的输入时

POE2F 标志 (POE2 标志)

这是表示给 POE2# 引脚输入了高阻抗请求的标志。

[为“0”的条件]

- 在读“1”后写“0”时

[为“1”的条件]

- 当 POE2# 引脚有 POE2M[1:0] 位设定的输入时

POE3F 标志 (POE3 标志)

这是表示给 POE3# 引脚输入了高阻抗请求的标志。

[为“0”的条件]

- 在读“1”后写“0”时

[为“1”的条件]

- 当 POE3# 引脚有 POE3M[1:0] 位设定的输入时

22.2.2 输出电平控制 / 状态寄存器 1 (OCSR1)

地址 0008 8902h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
OSF1	—	—	—	—	—	OCE1	OIE1	—	—	—	—	—	—	—	—
复位后的值	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b7-b0	—	保留位	读写值都为“0”。	R/W
b8	OIE1	输出短路的中断允许 1 位	0: 禁止通过输出电平的比较进行 OIE1 中断请求 1: 允许通过输出电平的比较进行 OIE1 中断请求	R/W
b9	OCE1	输出短路的高阻抗允许 1 位	0: 不将引脚置为高阻抗 1: 将引脚置为高阻抗	R/W (注 1)
b14-b10	—	保留位	读写值都为“0”。	R/W
b15	OSF1	输出短路标志 1	0: 不同时为有效电平 1: 同时为有效电平	R/(W) (注 2)

注 1. 复位后, 只能写 1 次。

注 2. 要将此标志置“0”时, 只能在读“1”后写“0”。

OIE1 位 (输出短路的中断允许 1 位)

当 OSF1 标志为“1”时, 此位指定是否请求 OIE1 中断。

OCE1 位 (输出短路的高阻抗允许 1 位)

当 OSF1 标志为“1”时, 此位指定是否将 MTU 的互补 PWM 输出引脚置为高阻抗。

OSF1 标志 (输出短路标志 1)

此标志表示表 22.3 的引脚组合和 MTU 互补 PWM 输出引脚要比较的 3 组 2 相输出中至少有 1 组同时为有效电平。当 POE2R2.PnCZEA (n=1、2、3) 位为“0”或者没将 MTU 的输出比较功能置为有效时, 即使对应的 MTU 互补 PWM 输出引脚同时变为有效电平, OSF1 标志也不变为“1”。

[为“0”的条件]

- 在读“1”后写“0”时
当写“0”时, 需要从 MTU 的互补 PWM 输出引脚输出无效电平。
详细内容请参照“22.3.6 高阻抗的解除”。

[为“1”的条件]

- 当 3 组 2 相输出中至少有 1 组同时变为有效电平时

22.2.3 输入电平控制 / 状态寄存器 2 (ICSR2)

地址 0008 8908h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	POE8F	—	—	POE8E	PIE2	—	—	—	—	—	—	POE8M[1:0]	—
复位后的值	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b1-b0	POE8M[1:0]	POE8 模式选择位	b1 b0 0 0: 在 POE8# 输入的下降沿接受请求 0 1: 以每个 PCLK 的 8 分频时钟对 POE8# 输入的 Low 电平进行 16 次采样, 当全部为 Low 电平时, 接受请求。 1 0: 以每个 PCLK 的 16 分频时钟对 POE8# 输入的 Low 电平进行 16 次采样, 当全部为 Low 电平时, 接受请求。 1 1: 以每个 PCLK 的 128 分频时钟对 POE8# 输入的 Low 电平进行 16 次采样, 当全部为 Low 电平时, 接受请求。	R/W (注 1)
b7-b2	—	保留位	读写值都为“0”。	R/W
b8	PIE2	端口中断允许 2 位	0: 禁止 OEI2 中断请求 1: 允许 OEI2 中断请求	R/W
b9	POE8E	POE8 高阻抗允许位	0: 不将 MTIOC0A、MTIOC0B、MTIOC0C 和 MTIOC0D 引脚置为高阻抗 1: 将 MTIOC0A、MTIOC0B、MTIOC0C 和 MTIOC0D 引脚置为高阻抗	R/W (注 1)
b11-b10	—	保留位	读写值都为“0”。	R/W
b12	POE8F	POE8 标志	0: POE8# 引脚无高阻抗请求 1: POE8# 引脚有高阻抗请求	R/(W) (注 2)
b15-b13	—	保留位	读写值都为“0”。	R/W

注 1. 复位后, 只能写 1 次。

注 2. 要将此标志置“0”时, 只能在读“1”后写“0”。

POE8M[1:0] 位 (POE8 模式选择位)

这些位选择 POE8# 引脚的输入模式。

PIE2 位 (端口中断允许 2 位)

当 POE8F 标志为“1”时, 此位指定是否请求 OEI2 中断。

POE8E 位 (POE8 高阻抗允许位)

当 POE8F 标志为“1”时, 此位指定是否将 MTU0 的引脚置为高阻抗。

POE8F 标志 (POE8 标志)

这是表示给 POE8# 引脚输入了高阻抗请求的标志。

[为“0”的条件]

在读“1”后写“0”时

在通过 POE8M[1:0] 位设定了 Low 电平采样的情况下要写“0”时, 需要给 POE8# 引脚输入 High 电平。

详细内容请参照“22.3.6 高阻抗的解除”。

[为“1”的条件]

- 当 POE8# 引脚有 POE8M[1:0] 位设定的输入时

22.2.4 软件端口输出允许寄存器 (SPOER)

地址 0008 890Ah

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	CH0HIZ	CH34HIZ
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	CH34HIZ	MTU3、MTU4 的输出高阻抗允许位	0: 不置为高阻抗 1: 置为高阻抗	R/W
b1	CH0HIZ	MTU0 的输出高阻抗允许位	0: 不置为高阻抗 1: 置为高阻抗	R/W
b7-b2	—	保留位	读写值都为“0”。	R/W

CH34HIZ 位 (MTU3、MTU4 的输出高阻抗允许位)

此位对 MTU 的互补 PWM 输出引脚 (MTIOC3B/MTIOC3D/MTIOC4A/MTIOC4B/MTIOC4C/MTIOC4D) 进行高阻抗控制。

[为“0”的条件]

- 在读“1”后写“0”时

[为“1”的条件]

- 当写“1”时
- 在接收到事件链接控制器 (ELC) 的事件信号时

CH0HIZ 位 (MTU0 的输出高阻抗允许位)

此位对 MTU0 的引脚 (MTIOC0A/MTIOC0B/MTIOC0C/MTIOC0D) 进行高阻抗控制。

[为“0”的条件]

- 在读“1”后写“0”时

[为“1”的条件]

- 当写“1”时
- 在接收到事件链接控制器 (ELC) 的事件信号时

22.2.5 端口输出允许控制寄存器 1 (POECR1)

地址 0008 890Bh

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	PE3ZE	PE2ZE	PE1ZE	PE0ZE
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	PE0ZE	MTIOC0A 的高阻抗允许位	0: 不置为高阻抗 1: 置为高阻抗	R/W (注 1)
b1	PE1ZE	MTIOC0B 的高阻抗允许位	0: 不置为高阻抗 1: 置为高阻抗	R/W (注 1)
b2	PE2ZE	MTIOC0C 的高阻抗允许位	0: 不置为高阻抗 1: 置为高阻抗	R/W (注 1)
b3	PE3ZE	MTIOC0D 的高阻抗允许位	0: 不置为高阻抗 1: 置为高阻抗	R/W (注 1)
b7-b4	—	保留位	读写值都为“0”。	R/W

注 1. 复位后, 只能写 1 次。

22.2.6 端口输出允许控制寄存器 2 (POECR2)

地址 0008 890Ch

b7	b6	b5	b4	b3	b2	b1	b0
—	P1CZEA	P2CZEA	P3CZEA	—	—	—	—
0	1	1	1	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b3-b0	—	保留位	读写值都为“0”。	R/W
b4	P3CZEA	MTU 端口 3 的高阻抗允许位	0: 不进行输出电平的比较并且不置为高阻抗 1: 置为高阻抗	R/W (注 1)
b5	P2CZEA	MTU 端口 2 的高阻抗允许位	0: 不进行输出电平的比较并且不置为高阻抗 1: 置为高阻抗	R/W (注 1)
b6	P1CZEA	MTU 端口 1 的高阻抗允许位	0: 不进行输出电平的比较并且不置为高阻抗 1: 置为高阻抗	R/W (注 1)
b7	—	保留位	读写值都为“0”。	R/W

注 1. 复位后, 只能写 1 次。

在不使用此功能时, 必须写“00h”。

P3CZEA 位 (MTU 端口 3 的高阻抗允许位)

此位设定是否允许将MTU互补PWM输出引脚的MTIOC4B和MTIOC4D置为高阻抗以及是否允许MTIOC4B和MTIOC4D的输出电平的比较。

P2CZEA 位 (MTU 端口 2 的高阻抗允许位)

此位设定是否允许将MTU互补PWM输出引脚的MTIOC4A和MTIOC4C置为高阻抗以及是否允许MTIOC4A和MTIOC4C的输出电平的比较。

P1CZEA 位 (MTU 端口 1 的高阻抗允许位)

此位设定是否允许将MTU互补PWM输出引脚的MTIOC3B和MTIOC3D置为高阻抗以及是否允许MTIOC3B和MTIOC3D的输出电平的比较。

22.2.7 输入电平控制 / 状态寄存器 3 (ICSR3)

地址 0008 890Eh

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	OSTSTF	—	—	OSTSTE	—	—	—	—	—	—	—	—	—
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b8-b0	—	保留位	读写值都为“0”。	R/W
b9	OSTSTE	OSTST 的高阻抗允许位	0: 不将 MTIOC0A、MTIOC0B、MTIOC0C、MTIOC0D、MTIOC3B、MTIOC3D、MTIOC4A、MTIOC4B、MTIOC4C 和 MTIOC4D 引脚置为高阻抗 1: 将 MTIOC0A、MTIOC0B、MTIOC0C、MTIOC0D、MTIOC3B、MTIOC3D、MTIOC4A、MTIOC4B、MTIOC4C 和 MTIOC4D 引脚置为高阻抗	R/W (注 1)
b11-b10	—	保留位	读写值都为“0”。	R/W
b12	OSTSTF	OSTST 高阻抗标志	0: 无振荡停止高阻抗请求 1: 有振荡停止高阻抗请求	R/(W) (注 2)
b15-b13	—	保留位	读写值都为“0”。	R/W

注 1. 复位后，只能写 1 次。

注 2. 要将此标志置“0”时，只能在读“1”后写“0”。

OSTSTE 位 (OSTST 的高阻抗允许位)

此位设定是否允许在检测到振荡停止时将 MTU 的互补 PWM 输出引脚和 MTU0 的引脚置为高阻抗。

OSTSTF 标志 (OSTST 高阻抗标志)

OSTSTF 标志表示振荡停止高阻抗请求。一旦进入振荡停止状态，此标志就变为“1”。要将 OSTSTF 标志置“0”时，必须在振荡停止检测信号无效的状态下给此标志写“0”。在振荡停止检测信号有效期间，即使给 OSTSTF 标志写“0”，此标志也不变为“0”。有效期间是指从检测到振荡停止后到经过 10 个 PCLK 周期为止的期间。

[为“0”的条件]

- 在读“1”后写“0”时

[为“1”的条件]

- 当检测到振荡停止状态时

22.3 运行说明

高阻抗的对象引脚和条件如下所示。

1. MTU0 的引脚 (MTIOC0A)

当以下的任意一个条件成立时，将引脚置为高阻抗。

- POE8# 引脚的输入电平的检测
在 POECSR1.PE0ZE 位和 ICSR2.POE8E 位为“1”的状态下 ICSR2.POE8F 标志变为“1”时
- SPOER 寄存器的设定
在 POECSR1.PE0ZE 位为“1”的状态下将 SPOER.CH0HIZ 位置“1”时
- 振荡停止的检测
在 POECSR1.PE0ZE 位和 ICSR3.OSTSTE 位为“1”的状态下 OSTSTF 标志变为“1”时
- 接收来自 ELC 的事件

2. MTU0 的引脚 (MTIOC0B)

当以下的任意一个条件成立时，将引脚置为高阻抗。

- POE8# 引脚的输入电平的检测
在 POECSR1.PE1ZE 位和 ICSR2.POE8E 位为“1”的状态下 ICSR2.POE8F 标志变为“1”时
- SPOER 寄存器的设定
在 POECSR1.PE1ZE 位为“1”的状态下将 SPOER.CH0HIZ 位置“1”时
- 振荡停止的检测
在 POECSR1.PE1ZE 位和 ICSR3.OSTSTE 位为“1”的状态下 OSTSTF 标志变为“1”时
- 接收来自 ELC 的事件

3. MTU0 的引脚 (MTIOC0C)

当以下的任意一个条件成立时，将引脚置为高阻抗。

- POE8# 引脚的输入电平的检测
在 POECSR1.PE2ZE 位和 ICSR2.POE8E 位为“1”的状态下 ICSR2.POE8F 标志变为“1”时
- SPOER 寄存器的设定
在 POECSR1.PE2ZE 位为“1”的状态下将 SPOER.CH0HIZ 位置“1”时
- 振荡停止的检测
在 POECSR1.PE2ZE 位和 ICSR3.OSTSTE 位为“1”的状态下 OSTSTF 标志变为“1”时
- 接收来自 ELC 的事件

4. MTU0 的引脚 (MTIOC0D)

当以下的任意一个条件成立时，将引脚置为高阻抗。

- POE8# 引脚的输入电平的检测
在 POECSR1.PE3ZE 位和 ICSR2.POE8E 位为“1”的状态下 ICSR2.POE8F 标志变为“1”时
- SPOER 寄存器的设定
在 POECSR1.PE3ZE 位为“1”的状态下将 SPOER.CH0HIZ 位置“1”时
- 振荡停止的检测
在 POECSR1.PE3ZE 位和 ICSR3.OSTSTE 位为“1”的状态下 OSTSTF 标志变为“1”时
- 接收来自 ELC 的事件

5. MTU3 的引脚 (MTIOC3B、MTIOC3D)

当以下的任意一个条件成立时，将引脚置为高阻抗。

- POE0# ~ POE3# 引脚的输入电平的检测
在 POE2R.P1CZEA 位为“1”的状态下 ICSR1.POE3F、ICSR1.POE2F、ICSR1.POE1F 或者 ICSR1.POE0F 标志变为“1”时
- MTIOC3B 引脚和 MTIOC3D 引脚的输出电平的比较
在 POE2R.P1CZEA 位和 OCSR1.OCE1 位为“1”的状态下 OCSR1.OSF1 标志变为“1”时
- SPOER 寄存器的设定
在 POE2R.P1CZEA 位为“1”的状态下将 SPOER.CH34HIZ 位置“1”时
- 振荡停止的检测
在 POE2R.P1CZEA 位和 ICSR3.OSTSTE 位为“1”的状态下 ICSR3.OSTSTF 标志变为“1”时
- 接收来自 ELC 的事件

6. MTU4 的引脚 (MTIOC4A、MTIOC4C)

当以下的任意一个条件成立时，将引脚置为高阻抗。

- POE0# ~ POE3# 引脚的输入电平的检测
在 POE2R.P2CZEA 位为“1”的状态下 ICSR1.POE3F、ICSR1.POE2F、ICSR1.POE1F 或者 ICSR1.POE0F 标志变为“1”时
- MTIOC4A 引脚和 MTIOC4C 引脚的输出电平的比较
在 POE2R.P2CZEA 位和 OCSR1.OCE1 位为“1”的状态下 OCSR1.OSF1 标志变为“1”时
- SPOER 寄存器的设定
在 POE2R.P2CZEA 位为“1”的状态下将 SPOER.CH34HIZ 位置“1”时
- 振荡停止的检测
在 POE2R.P2CZEA 位和 ICSR3.OSTSTE 位为“1”的状态下 ICSR3.OSTSTF 标志变为“1”时
- 接收来自 ELC 的事件

7. MTU4 的引脚 (MTIOC4B、MTIOC4D)

当以下的任意一个条件成立时，将引脚置为高阻抗。

- POE0# ~ POE3# 引脚的输入电平的检测
在 POE2R.P3CZEA 位为“1”的状态下 ICSR1.POE3F、ICSR1.POE2F、ICSR1.POE1F 或者 ICSR1.POE0F 标志变为“1”时
- MTIOC4B 引脚和 MTIOC4D 引脚的输出电平的比较
在 POE2R.P3CZEA 位和 OCSR1.OCE1 位为“1”的状态下 OCSR1.OSF1 标志变为“1”时
- SPOER 寄存器的设定
在 POE2R.P3CZEA 位为“1”的状态下将 SPOER.CH34HIZ 位置“1”时
- 振荡停止的检测
在 POE2R.P3CZEA 位和 ICSR3.OSTSTE 位为“1”的状态下 ICSR3.OSTSTF 标志变为“1”时
- 接收来自 ELC 的事件

22.3.1 输入电平的检测

当 POE0# ~ POE3# 引脚和 POE8# 引脚产生 ICSR1 寄存器和 ICSR2 寄存器设定的输入条件时，将 MTU 的互补 PWM 输出引脚和 MTU0 的引脚置为高阻抗。

(1) 下降沿检测

当 POE0# ~ POE3# 引脚和 POE8# 引脚从 High 电平变为 Low 电平时，将 MTU 的互补 PWM 输出引脚和 MTU0 的引脚置为高阻抗。

在通过 PCLK 进行采样后检测下降沿。如果给 POE0# ~ POE3# 引脚和 POE8# 引脚输入了不满 1 个 PCLK 周期的 Low 电平，就无法保证是否能检测到下降沿。

从 POE0# ~ POE3# 和 POE8# 的引脚输入到将引脚置为高阻抗为止的时序例子如图 22.2 所示。

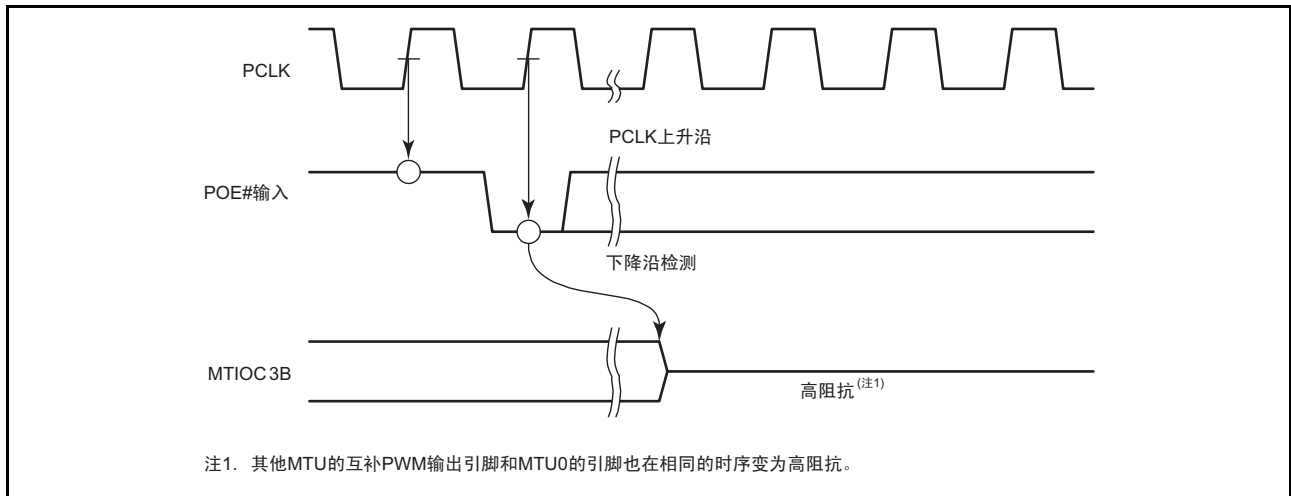


图 22.2 下降沿检测

(2) Low 电平检测

Low 电平检测如图 22.3 所示。通过 ICSR1 寄存器和 ICSR2 寄存器设定的采样时钟连续对 Low 电平进行 16 次采样。此时，只要有 1 次检测到 High 电平，就不接受请求。在未输出采样时钟的期间，POE0# ~ POE3# 引脚和 POE8# 引脚即使发生变化也被忽视。

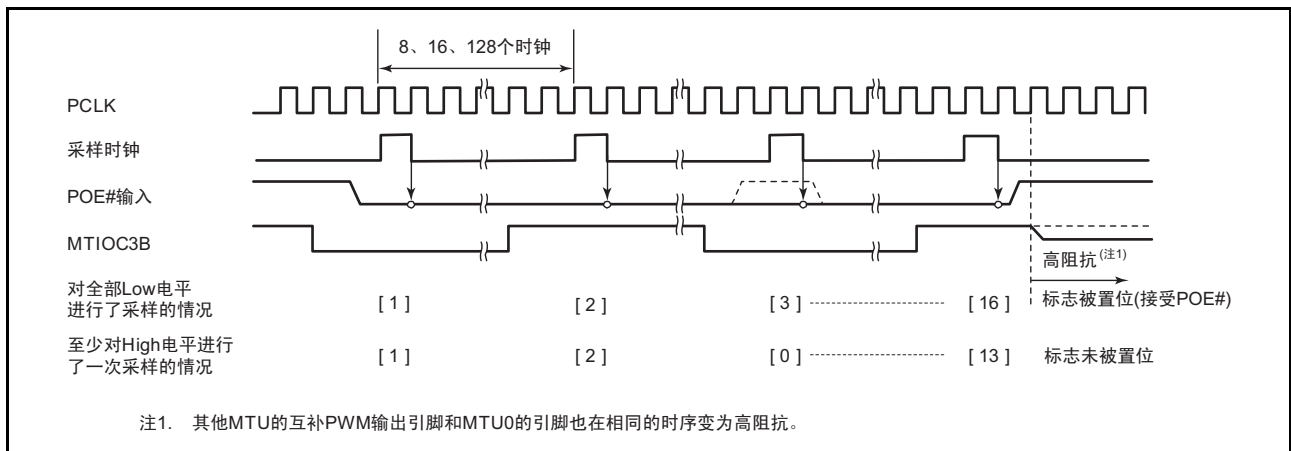


图 22.3 Low 电平检测

22.3.2 输出电平的比较

以 MTIOC3B 和 MTIOC3D 的组合为例，MTU 互补 PWM 输出引脚的输出电平比较如图 22.4 所示，其他引脚的组合也一样。

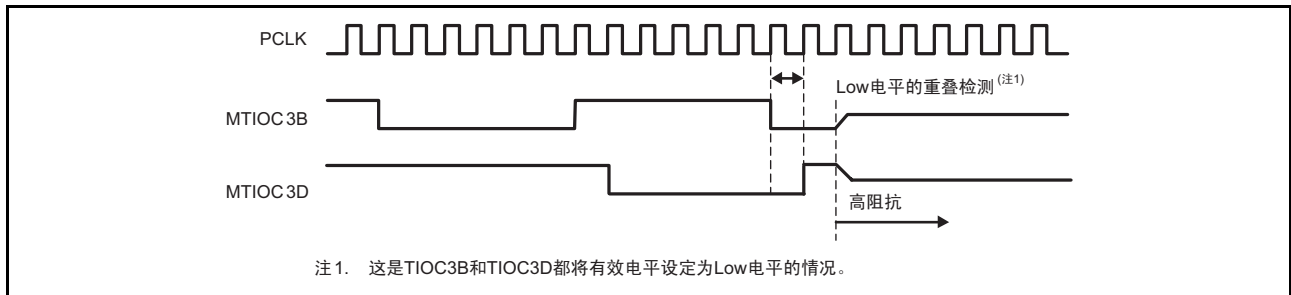


图 22.4 输出电平的比较

22.3.3 通过寄存器进行的高阻抗控制

通过写软件端口输出允许寄存器 (SPOER)，对 MTU 的互补 PWM 输出引脚和 MTU0 的引脚进行高阻抗控制。

通过将 SPOER.CH34HIZ 位置“1”，将端口输出允许控制寄存器 2 (POECR2) 所设 MTU 的互补 PWM 输出引脚 (MTU3、MTU4) 置为高阻抗。

通过将 SPOER.CH0HIZ 位置“1”，将 POECR1 寄存器设定的 MTU0 的输出引脚置为高阻抗。

22.3.4 通过振荡停止检测进行的高阻抗控制

如果通过时钟发生电路的振荡停止检测功能检测到振荡停止，就将 POECR2 寄存器设定的 MTUA 的互补 PWM 输出引脚和 POECR1 寄存器设定的 MTU0 的引脚置为高阻抗。

22.3.5 通过接收 ELC 的事件信号进行的高阻抗控制

能通过接收来自 ELC 的事件信号，将 MTU 的互补 PWM 输出引脚和 MTU0 的引脚置为高阻抗。

想要通过 ELC 对 MTU 的互补 PWM 输出引脚和 MTU0 的引脚进行高阻抗控制时，必须事先将对应的寄存器 (POECR1 或者 POECR2) 设定为允许高阻抗。如果接收到来自 ELC 的事件信号，对应的位 (SPOER.CH0HIZ 或者 SPOER.CH34HIZ) 就变为“1”，并且 MTU 的互补 PWM 输出引脚或者 MTU0 的引脚变为高阻抗。

22.3.6 高阻抗的解除

因输入电平检测而变为高阻抗的 MTU 的互补 PWM 输出引脚和 MTU0 的引脚通过复位恢复到初始状态，或者通过将 ICSR1.POE3F 标志、ICSR1.POE2F 标志、ICSR1.POE1F 标志、ICSR1.POE0F 标志和 ICSR2.POE8F 标志置“0”，解除高阻抗状态。但是，当通过 ICSR1.POE3M[1:0] 位、ICSR1.POE2M[1:0] 位、ICSR1.POE1M[1:0] 位、ICSR1.POE0M[1:0] 位和 ICSR2.POE8M[1:0] 位设定为 Low 电平采样时，如果不是在 POE0# ~ POE3# 引脚和 POE8# 引脚输入 High 电平后对 High 电平进行采样以后，即使给此标志写“0”也无效，此标志不变为“0”。

因输出电平比较而变为高阻抗的 MTU 的互补 PWM 输出引脚通过复位恢复到初始状态，或者通过将 OCSR1.OSF1 标志置“0”，解除高阻抗状态。但是，如果不是在从 MTU 的互补 PWM 输出引脚输出无效电平以后，即使给此标志写“0”也无效，此标志不变为“0”。能通过设定 MTU 的寄存器，输出无效电平。

对于因时钟发生电路的振荡停止而变为高阻抗的 MTU 的互补 PWM 输出引脚和 MTU0 的引脚，通过将 ICSR3.OSTSTF 位或者 ICSR3.OSTSTE 位置“0”，解除高阻抗状态。

对于因 SPOER.CH34HIZ 位或者 SPOER.CH0HIZ 位而变为高阻抗的 MTU 的互补 PWM 输出引脚或者 MTU0 的引脚，通过将引脚的对应位 (SPOER.CH34HIZ 位、SPOER.CH0HIZ 位) 置“0”，解除高阻抗状态。

22.4 中断

如果在进行输入电平的检测、输出电平的比较或者时钟发生电路的振荡停止时满足条件，POE 就能在发出中断请求后产生中断。中断的种类和产生中断请求的条件如表 22.4 所示。在接受 OEI1 中断和 OEI2 中断后，必须确认在对应中断的异常处理程序的起始位置对应标志已被置“1”。

表 22.4 中断请求的种类和条件

名称	中断源	对应标志	条件
OEI1	输出允许中断 1	POE0F、POE1F、POE2F、POE3F、OSF1	在 ICSR1.PIE1 位为“1”的状态下 ICSR1.POE0F、ICSR1.POE1F、ICSR1.POE2F 或者 ICSR1.POE3F 标志变为“1”时，或者在 OCSR1.OIE1 位为“1”的状态下 OCSR1.OSF1 标志变为“1”时
OEI2	输出允许中断 2	POE8F	在 ICSR2.PIE2 位为“1”的状态下 ICSR2.POE8F 标志变为“1”时

22.5 使用时的注意事项

22.5.1 向软件待机模式或者深度软件待机模式的转移

在使用 POE 时，不能向软件待机模式和深度软件待机模式转移。因为在软件待机模式和深度软件待机模式中 POE 停止运行，所以不能对引脚进行高阻抗控制。

22.5.2 不使用 POE 时的注意事项

在不使用 POE 时，必须分别给端口输出允许控制寄存器 1 (POECR1) 和端口输出允许控制寄存器 2 (POECR2) 写“00h”。

22.5.3 有关引脚的 MTU 功能的设定

只有在通过端口模式寄存器 (PMR) 选择引脚作为 MTU 的对应引脚时，才能通过 POE 进行高阻抗控制。在选择引脚作为通用输入 / 输出端口时，不进行高阻抗控制。

22.5.4 通过接收 ELC 的事件信号进行的高阻抗控制的注意事项

当给 SOPER.CH34HIZ 位或者 SPOER.CH0HIZ 位写“0”与接收事件信号发生竞争时，优先接收事件信号，并且对应的位变为“1”。通过接收来自 ELC 的事件信号使 MTU 的互补 PWM 输出引脚和 MTU0 的引脚变为高阻抗时，不产生中断请求。

23. 8 位定时器 (TMR)

RX210 群内置以 8 位计数器为基础的 2 个通道的 8 位定时器 (TMR)，有 2 个单元 (单元 0 和单元 1)，共计 4 个通道。除了能对外部事件进行计数以外，还能通过和 2 个寄存器的比较匹配信号，作为多功能定时器应用于计数器复位、中断请求、任意占空比的脉冲输出等方面。

单元 0 和单元 1 的功能相同，能生成 SCI 的波特率时钟。

23.1 概要

TMR 的规格如表 23.1 所示。

单元 0 和单元 1 的框图分别如图 23.1 和图 23.2 所示。

表 23.1 TMR 的规格

项目	规格
计数时钟	<ul style="list-style-type: none"> 分频时钟: PCLK/1、PCLK/2、PCLK/8、PCLK/32、PCLK/64、PCLK/1024、PCLK/8192 外部时钟
通道数	(8 位 × 2 个通道) × 2 个单元
比较匹配	<ul style="list-style-type: none"> 8 位模式 (比较匹配 A 和比较匹配 B) 16 位模式 (比较匹配 A 和比较匹配 B)
计数器清除	能选择比较匹配 A、比较匹配 B 或者外部复位信号。
定时器输出	任意占空比的脉冲输出和 PWM 输出。
2 个通道的级联	<ul style="list-style-type: none"> 16 位计数模式 将 TMR0 作为高位并且将 TMR1 作为低位 (或者将 TMR2 作为高位并且将 TMR3 作为低位) 的 16 位定时器。 比较匹配计数模式 TMR1 对 TMR0 的比较匹配进行计数 (或者 TMR3 对 TMR2 的比较匹配进行计数)。
中断源	比较匹配 A、比较匹配 B 或者上溢。
事件链接功能 (输出)	比较匹配 A、比较匹配 B 或者上溢 (TMR0、TMR2)
事件链接功能 (输入)	<ol style="list-style-type: none"> 开始计数运行 (TMR0、TMR2) 事件计数器运行 (TMR0、TMR2) 重新开始计数运行 (TMR0、TMR2)
DTC 的启动	能通过比较匹配 A 中断或者比较匹配 B 中断启动 DTC。
SCI 波特率时钟的生成	生成 SCI 的波特率时钟 (注 1)。
低功耗功能	各单元能设定为模块停止状态。

注 1. 详细内容请参照“28. 串行通信接口 (SC1c、SC1d)”。

表 23.2 TMR 的功能一览表

项目		单元 0			单元 1		
		8 位		16 位	8 位		16 位
计数器模式		TMR0	TMR1	TMR0+TMR1	TMR2	TMR3	TMR2+TMR3
通道		TMR0	TMR1	TMR0+TMR1	TMR2	TMR3	TMR2+TMR3
计数时钟		PCLK/1 PCLK2 PCLK/8 PCLK/32 PCLK/64 PCLK/1024 PCLK/8192 TMCIO	PCLK/1 PCLK2 PCLK/8 PCLK/32 PCLK/64 PCLK/1024 PCLK/8192 TMCI1	PCLK/1 PCLK2 PCLK/8 PCLK/32 PCLK/64 PCLK/1024 PCLK/8192 TMCI1	PCLK/1 PCLK2 PCLK/8 PCLK/32 PCLK/64 PCLK/1024 PCLK/8192 TMC2	PCLK/1 PCLK2 PCLK/8 PCLK/32 PCLK/64 PCLK/1024 PCLK/8192 TMC3	PCLK/1 PCLK2 PCLK/8 PCLK/32 PCLK/64 PCLK/1024 PCLK/8192 TMC3
计数器清除		TMR0.TCORA TMR0.TCORAB TMRIO	TMR0.TCORA TMR0.TCORAB TMR1 TMRIO	TMR0.TCORA+ TMR1.TCORA TMR0.TCORB+ TMR1.TCORB TMRIO	TMR2.TCORA TMR2.TCORAB TMRIO	TMR3.TCORA TMR3.TCORAB TMR1	TMR2.TCORA+ TMR3.TCORA TMR2.TCORB+ TMR3.TCORB TMRIO
比较匹配	比较匹配 A	○	○	○	○	○	○
	比较匹配 B	○	○	○	○	○	○
定时器输出	输出 Low 电平	○	○	○	○	○	○
	输出 High 电平	○	○	○	○	○	○
	交替输出	○	○	○	○	○	○
DTC 的启动	比较匹配 A	○	○	○	○	○	○
	比较匹配 B	○	○	○	○	○	○
	TCNT 的上溢	—	—	—	—	—	—
中断	比较匹配 A	CMIA0	CMIA1	CMIA0	CMIA2	CMIA3	CMIA2
	比较匹配 B	CMIB0	CMIB1	CMIB0	CMIB2	CMIB3	CMIB2
	TCNT 的上溢	OVI0	OVI1	OVI0	OVI2	OVI3	OVI2
级联		TMR1 的上溢	TMR0 的比较匹配 A	—	TMR3 的上溢	TMR2 的比较匹配 A	—
SCI 波特率时钟的生成 (注 1)		○		—	○		—
模块停止的设定 (注 2)		(单元 0) MSTPCRA.MSTPA5 位和 (单元 1) MSTPCRA.MSTPA4 位					

○：能

—：不能

注 1. 详细内容请参照“28. 串行通信接口 (SCIC、SCID)”。

注 2. 详细内容请参照“11. 低功耗功能”。

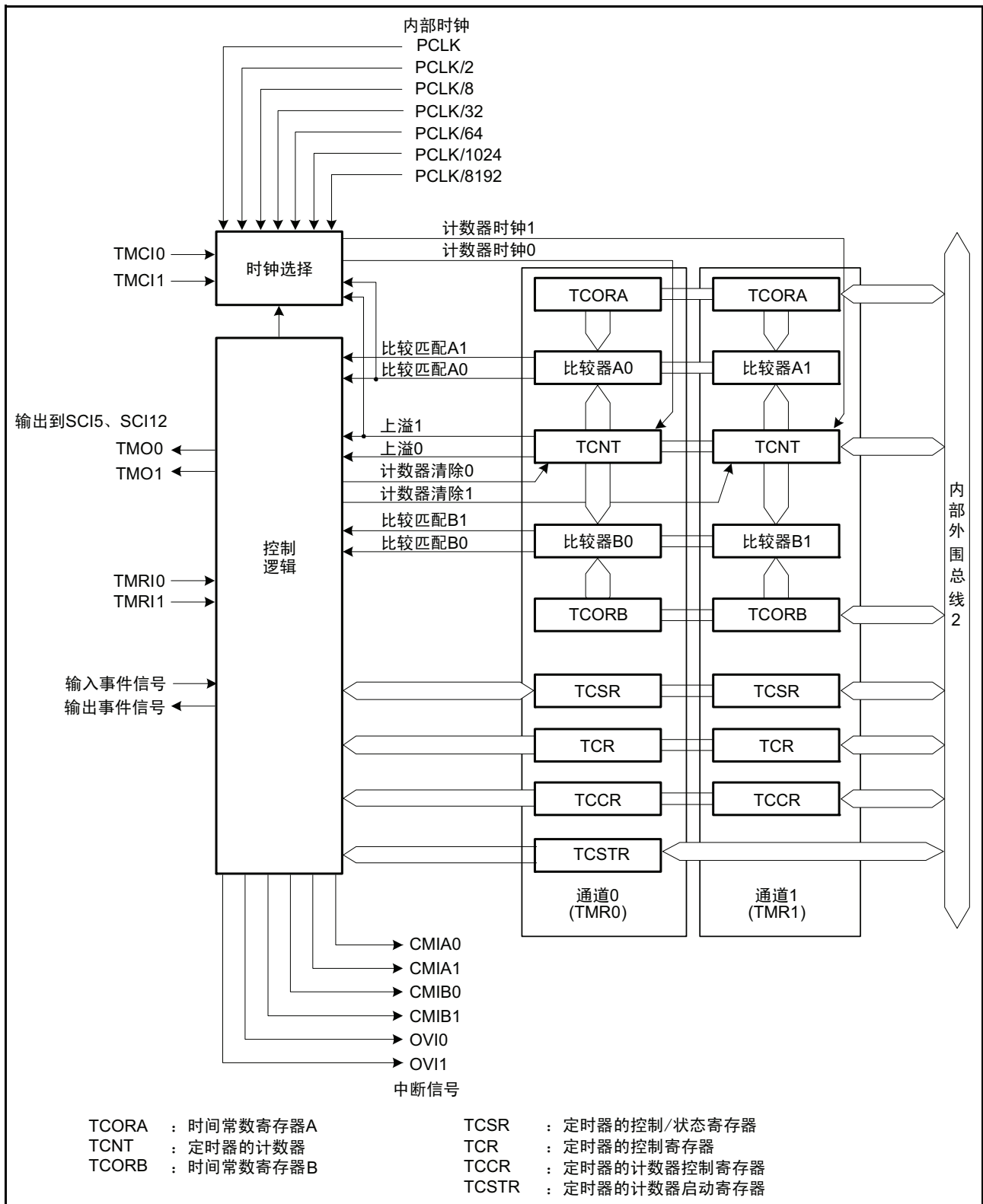


图 23.1 TMR (单元 0) 的框图

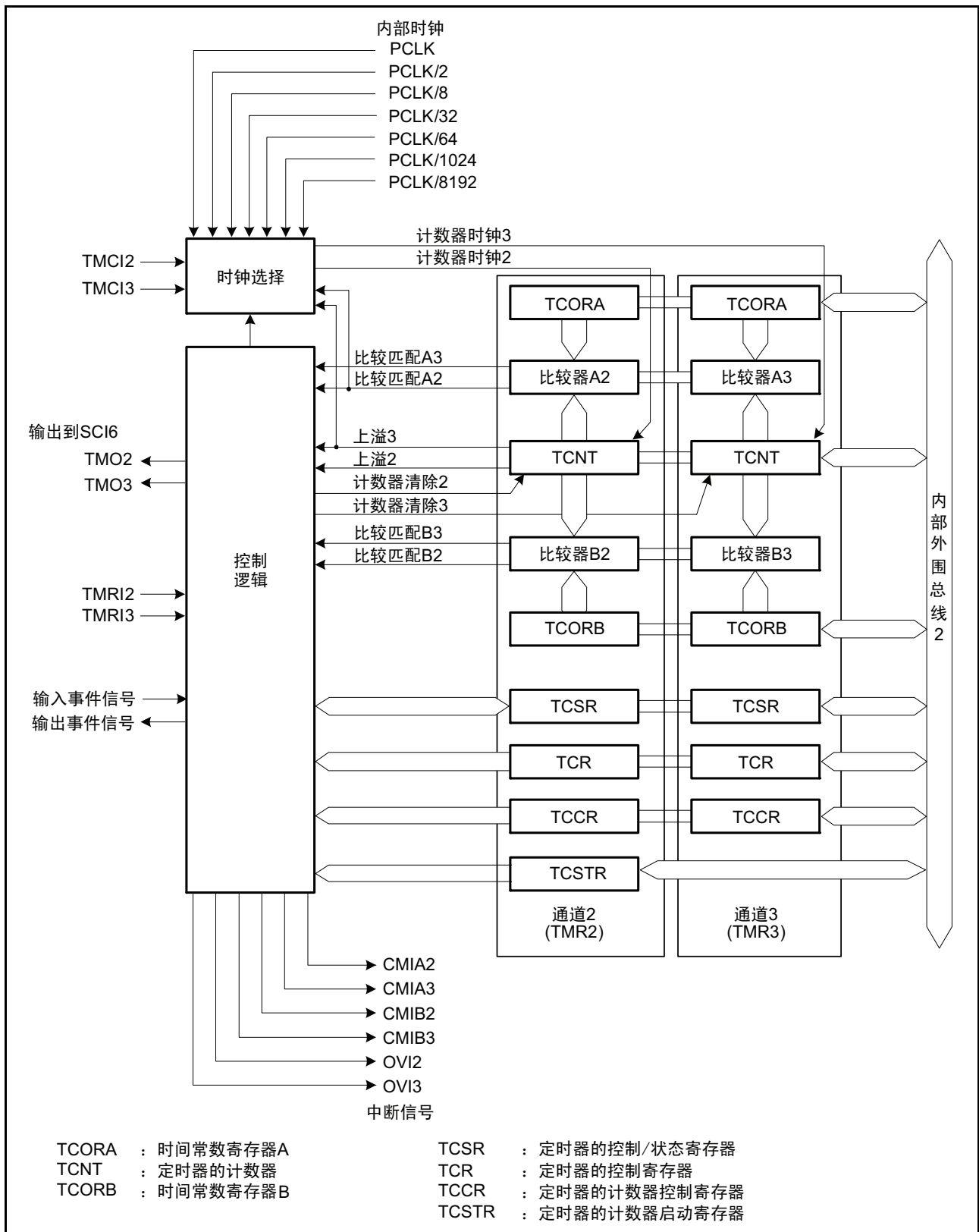


图 23.2 TMR (单元 1) 的框图

TMR 使用的输入 / 输出引脚如表 23.3 所示。

表 23.3 TMR 的输入 / 输出引脚

单元	通道	引脚名	输入 / 输出	功能
单元 0	TMR0	TMO0	输出	比较匹配的输出生
		TMCI0	输入	计数器外部时钟的输入
		TMRI0	输入	计数器外部复位的输入
	TMR1	TMO1	输出	比较匹配的输出生
		TMCI1	输入	计数器外部时钟的输入
		TMRI1	输入	计数器外部复位的输入
单元 1	TMR2	TMO2	输出	比较匹配的输出生
		TMCI2	输入	计数器外部时钟的输入
		TMRI2	输入	计数器外部复位的输入
	TMR3	TMO3	输出	比较匹配的输出生
		TMCI3	输入	计数器外部时钟的输入
		TMRI3	输入	计数器外部复位的输入

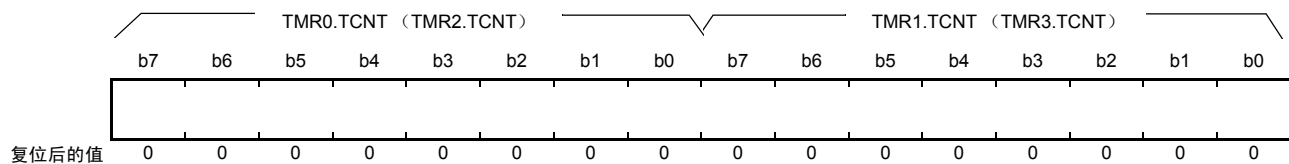
23.2 寄存器说明

表 23.4 16 位存取寄存器分配

地址	高 8 位	低 8 位
0008 8208h	TMR0.TCNT	TMR1.TCNT
0008 8204h	TMR0.TCORA	TMR1.TCORA
0008 8206h	TMR0.TCORB	TMR1.TCORB
0008 820Ah	TMR0.TCCR	TMR1.TCCR
0008 8218h	TMR2.TCNT	TMR3.TCNT
0008 8214h	TMR2.TCORA	TMR3.TCORA
0008 8216h	TMR2.TCORB	TMR3.TCORB
0008 821Ah	TMR2.TCCR	TMR3.TCCR

23.2.1 定时器的计数器 (TCNT)

地址 TMR0.TCNT 0008 8208h、TMR1.TCNT 0008 8209h、TMR2.TCNT 0008 8218h、TMR3.TCNT 0008 8219h



TCNT 计数器是 8 位可读写递增计数器。

也能将 TMR0.TCNT 计数器和 TMR1.TCNT 计数器（或者 TMR2.TCNT 计数器和 TMR3.TCNT 计数器）作为 16 位计数器进行字存取。

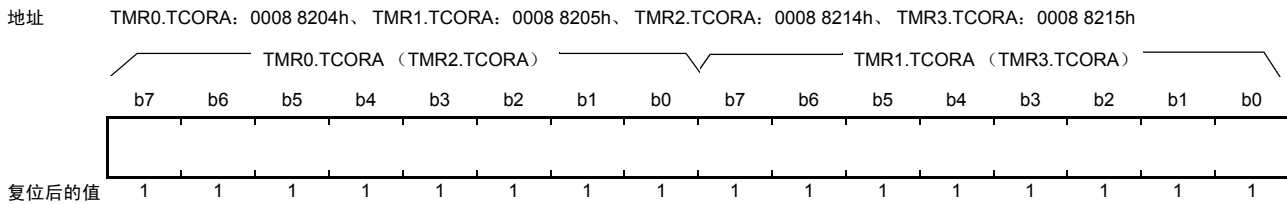
能通过 TCCR.CSS[1:0] 位和 TCCR.CKS[2:0] 位选择时钟。

能通过外部复位输入信号、比较匹配 A 信号或者比较匹配 B 信号清除 TCNT 计数器，并且通过 TCR.CCLR[1:0] 位选择用哪个信号进行清除。

如果 TCNT 计数器发生上溢（“FFh”→“00h”），中断标志就变为“1”。

对应的中断向量号请参照“14. 中断控制器 (ICUb)”和“表 23.6 TMR 的中断源”。

23.2.2 时间常数寄存器 A (TCORA)

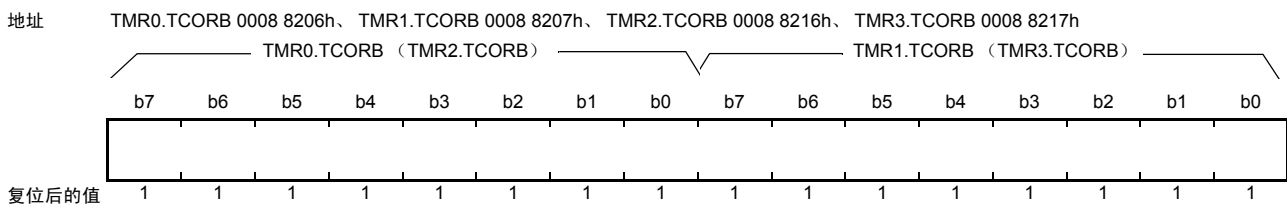


TCORA 寄存器是 8 位可读写寄存器。

也能将 TMR0.TCORA 寄存器和 TMR1.TCORA 寄存器（或者 TMR2.TCORA 寄存器和 TMR3.TCORA 寄存器）作为 16 位寄存器进行字存取。

TCORA 寄存器的值随时和 TCNT 计数器进行比较，如果两者的值相同，比较匹配 A 信号就变为 High 电平。但是，在写 TCORA 寄存器时不进行比较。还能通过 TCSR.OSA[1:0] 位的设定和此比较匹配 A 信号，控制 TMO_n 引脚的定时器输出。

23.2.3 时间常数寄存器 B (TCORB)



TCORB 寄存器是 8 位可读写寄存器。

也能将 TMR0.TCORB 寄存器和 TMR1.TCORB 寄存器（或者 TMR2.TCORB 寄存器和 TMR3.TCORB 寄存器）作为 16 位寄存器进行字存取。

TCORB 寄存器的值随时和 TCNT 计数器进行比较。如果两者的值相同，比较匹配 B 信号就变为 High 电平。但是，在写 TCORB 寄存器时不进行比较。还能通过 TCSR.OSB[1:0] 位的设定和此比较匹配 B 信号，控制 TMO_n 引脚的定时器输出。

23.2.4 定时器的控制寄存器 (TCR)

地址 TMR0.TCR 0008 8200h、TMR1.TCR 0008 8201h、TMR2.TCR 0008 8210h、TMR3.TCR 0008 8211h

b7	b6	b5	b4	b3	b2	b1	b0
CMIEB	CMIEA	OVIE	CCLR[1:0]	—	—	—	
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b2-b0	—	保留位	读写值都为“0”。	R/W
b4-b3	CCLR[1:0]	计数器清除位 (注 1)	b4 b3 0 0: 禁止清除 0 1: 通过比较匹配 A 进行清除 1 0: 通过比较匹配 B 进行清除 1 1: 通过外部复位输入进行清除 (通过 TCCR.TMRIS 位选择边沿或者电平)	R/W
b5	OVIE	定时器的上溢中断允许位	0: 禁止由上溢引起的中断请求 (OVIn) 1: 允许由上溢引起的中断请求 (OVIn)	R/W
b6	CMIEA	比较匹配中断允许 A 位	0: 禁止由比较匹配 A 引起的中断请求 (CMIAAn) 1: 允许由比较匹配 A 引起的中断请求 (CMIAAn)	R/W
b7	CMIEB	比较匹配中断允许 B 位	0: 禁止由比较匹配 B 引起的中断请求 (CMIBn) 1: 允许由比较匹配 B 引起的中断请求 (CMIBn)	R/W

注 1. 在使用计数器外部复位时, 必须将对应引脚的 PORTn.PDR.Bn 位置“0”并且将 PORTn.PMR.Bn 位置“1”, 详细内容请参照“19. I/O 端口”。

CCLR[1:0] 位 (计数器清除位)

这些位指定 TCNT 计数器的清除条件。

OVIE 位 (定时器的上溢中断允许位)

此位允许或者禁止由 TCNT 计数器的上溢引起的中断请求 (OVIn)。

CMIEA 位 (比较匹配中断允许 A 位)

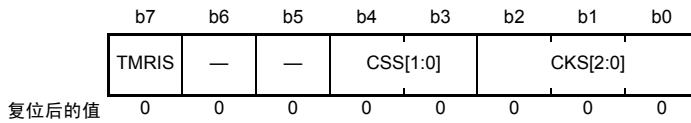
此位选择允许或者禁止在 TCORA 寄存器和 TCNT 计数器的值相同时输出的比较匹配 A 引起的中断请求 (CMIAAn)。

CMIEB 位 (比较匹配中断允许 B 位)

此位选择允许或者禁止在 TCORB 寄存器和 TCNT 计数器的值相同时输出的比较匹配 B 引起的中断请求 (CMIBn)。

23.2.5 定时器的计数器控制寄存器 (TCCR)

地址 TMR0.TCCR 0008 820Ah、TMR1.TCCR 0008 820Bh、TMR2.TCCR 0008 821Ah、TMR3.TCCR 0008 821Bh



位	符号	位名	功能	R/W
b2-b0	CKS[2:0]	时钟选择位 (注 1)	请参照表 23.5。	R/W
b4-b3	CSS[1:0]	时钟源选择位	请参照表 23.5。	R/W
b6-b5	—	保留位	读写值都为“0”。	R/W
b7	TMRIS	定时器的复位检测条件选择位	0: 在外部复位的上升沿进行清除 1: 通过外部复位的 High 电平进行清除	R/W

注 1. 在使用计数器外部复位时, 必须将对应引脚的 PORTn.PDR.Bn 位置“0”并且将 PORTn.PMR.Bn 位置“1”, 详细内容请参照“19. I/O 端口”。

CKS[2:0] 位 (时钟选择位)

CSS[1:0] 位 (时钟源选择位)

CKS[2:0] 位和 CSS[1:0] 位选择时钟, 详细内容请参照表 23.5。

TMRIS 位 (定时器的复位检测条件选择位)

此位在 TCR.CCLR[1:0] 位为“11b” (通过外部复位输入进行清除) 时有效, 用于选择外部复位检测条件 (电平或者边沿)。

表 23.5 TCNT 计数器的输入时钟和计数条件

通道	TCCR 寄存器					功能		
	CSS[1:0]		CKS[2:0]					
	b4	b3	b2	b1	b0			
TMR0 (TMR2)	0	0	—	0	0	禁止时钟输入。		
					1	在外部时钟的上升沿进行计数 (注1)。		
				1	0	在外部时钟的下降沿进行计数 (注1)。		
					1	在外部时钟的双边沿进行计数 (注1)。		
	0	1	0	0	0	分频时钟: 通过 PCLK 进行计数。		
					1	分频时钟: 通过 PCLK/2 进行计数。		
					1	0	分频时钟: 通过 PCLK/8 进行计数。	
						1	分频时钟: 通过 PCLK/32 进行计数。	
				1	0	0	分频时钟: 通过 PCLK/64 进行计数。	
						1	分频时钟: 通过 PCLK/1024 进行计数。	
						1	0	分频时钟: 通过 PCLK/8192 进行计数。
							1	禁止时钟输入。
	1	0	—	—	—	不能设定。		
	1	1	—	—	—	通过 TMR1.TCNT (TMR3.TCNT) 的上溢信号进行计数 (注2)。		
TMR1 (TMR3)	0	0	—	0	0	禁止时钟输入。		
					1	在外部时钟的上升沿进行计数 (注1)。		
				1	0	在外部时钟的下降沿进行计数 (注1)。		
					1	在外部时钟的双边沿进行计数 (注1)。		
	0	1	0	0	0	分频时钟: 通过 PCLK 进行计数。		
					1	分频时钟: 通过 PCLK/2 进行计数。		
					1	0	分频时钟: 通过 PCLK/8 进行计数。	
						1	分频时钟: 通过 PCLK/32 进行计数。	
				1	0	0	分频时钟: 通过 PCLK/64 进行计数。	
						1	分频时钟: 通过 PCLK/1024 进行计数。	
						1	0	分频时钟: 通过 PCLK/8192 进行计数。
							1	禁止时钟输入。
	1	0	—	—	—	不能设定。		
	1	1	—	—	—	通过 TMR0.TCNT (TMR2.TCNT) 的比较匹配 A 进行计数 (注2)。		

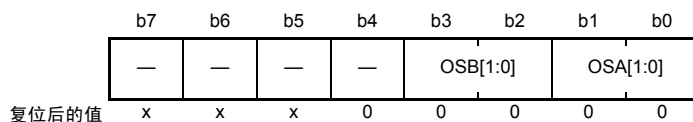
注 1. 在使用计数器外部复位时, 必须将对应引脚的 PORTn.PDR.Bn 位置“0”并且将 PORTn.PMR.Bn 位置“1”, 详细内容请参照“19. I/O 端口”。

注 2. 如果将 TMR0 (或者 TMR2) 的时钟输入用作 TMR1.TCNT (或者 TMR3.TCNT) 计数器的上溢信号并且将 TMR1 (或者 TMR3) 的时钟输入用作 TMR0.TCNT (或者 TMR2.TCNT) 计数器的比较匹配信号, 就不产生递增计数时钟。不能进行此设定。

23.2.6 定时器的控制 / 状态寄存器 (TCSR)

- TMR0.TCSR 寄存器和 TMR2.TCSR 寄存器

地址 TMR0.TCSR 0008 8202h、TMR2.TCSR 0008 8212h



位	符号	位名	功能	R/W
b1-b0	OSA[1:0]	输出选择位 A (注 1)	b1 b0 0 0: 不变 0 1: Low 电平输出 1 0: High 电平输出 1 1: 反相输出 (交替输出)	R/W
b3-b2	OSB[1:0]	输出选择位 B (注 1)	b3 b2 0 0: 不变 0 1: Low 电平输出 1 0: High 电平输出 1 1: 反相输出 (交替输出)	R/W
b4	—	保留位	读写值都为“0”。	R/W
b7-b5	—	保留位	读取值为不定值, 只能写“1”。	R/W

注 1. 当 OSA[1:0] 位和 OSB[1:0] 位都为“0”时, 将 TMRn 引脚对应的输出允许置为无效并且向 I/O 端口请求高阻抗输出。如果将 OSA[1:0] 位或者 OSB[1:0] 位置“1”, 在发生复位后的第 1 个比较匹配前, 定时器的输出就为 Low 电平。

OSA[1:0] 位 (输出选择位 A)

这些位选择通过 TCORA 寄存器和 TCNT 计数器的比较匹配 A 进行的 TMRn 引脚的输出方法。

OSB[1:0] 位 (输出选择位 B)

这些位选择通过 TCORB 寄存器和 TCNT 计数器的比较匹配 B 进行的 TMRn 引脚的输出方法。

- TMR1.TCSR 寄存器和 TMR3.TCSR 寄存器

地址 TMR1.TCSR 0008 8203h、TMR3.TCSR 0008 8213h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	OSB[1:0]		OSA[1:0]	
复位后的值	x	x	x	1	0	0	0

位	符号	位名	功能	R/W
b1-b0	OSA[1:0]	输出选择位 A (注 1)	b1 b0 0 0: 不变 0 1: Low 电平输出 1 0: High 电平输出 1 1: 反相输出 (交替输出)	R/W
b3-b2	OSB[1:0]	输出选择位 B (注 1)	b3 b2 0 0: 不变 0 1: Low 电平输出 1 0: High 电平输出 1 1: 反相输出 (交替输出)	R/W
b4	—	保留位	读写值都为“1”。	R/W
b7-b5	—	保留位	读取值为不定值, 只能写“1”。	R/W

注 1. 当 OSA[1:0] 位和 OSB[1:0] 位都为“0”时, 将 TMO_n 引脚对应的输出允许置为无效并且向 I/O 端口请求高阻抗输出。如果将 OSA[1:0] 位或者 OSB[1:0] 位置“1”, 在发生复位后的第 1 个比较匹配前, 定时器的输出就为 Low 电平。

OSA[1:0] 位 (输出选择位 A)

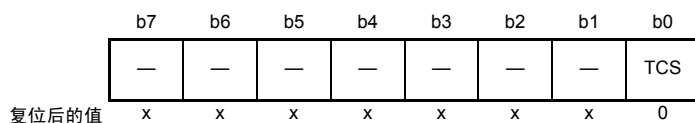
这些位选择通过 TCORA 寄存器和 TCNT 计数器的比较匹配 A 进行 TMO_n 引脚输出的方法。

OSB[1:0] 位 (输出选择位 B)

这些位选择通过 TCORB 寄存器和 TCNT 计数器的比较匹配 B 进行 TMO_n 引脚输出的方法。

23.2.7 定时器的计数器启动寄存器 (TCSTR)

地址 TMR0.TCSTR 0008 820Ch、TMR2.TCSTR 0008 821Ch



位	符号	位名	功能	R/W
b0	TCS	定时器的计数器状态位	0: 通过 ELC 停止计数的状态 1: 通过 ELC 开始计数的状态	R/W
b7-b1	—	(保留位)	读取值为不定值, 只能写“0”。	R/W

TCS 位 (定时器的计数器状态位)

此位能确认通过 ELC 进行的定时器计数的状态。

当此位的读取值为“1”时, 为通过 ELC 开始定时器计数的状态; 读取值为“0”时, 为通过 ELC 停止定时器计数的状态。

必须通过写“0”清除此位。写“1”无效。

只有在通过事件链接控制器 (ELC) 的 ELOPD 寄存器选择开始计数运行时 TCS 位有效。

详细内容请参照“23.7 通过 ELC 进行的链接运行”和“18. 事件链接控制器 (ELC)”。

23.3 运行说明

23.3.1 脉冲输出

任意占空比的脉冲输出例子如图 23.3 所示。

1. 为了通过 TCORA 寄存器的比较匹配清除 TCNT 计数器，将 TCR.CCLR[1:0] 位置“01b”（通过比较匹配 A 进行清除）。
2. 为了通过 TCORA 寄存器的比较匹配进行 High 电平输出并且通过 TCORB 寄存器的比较匹配进行 Low 电平输出，将 TCSR.OSA[1:0] 位置“10b”（High 电平输出）并且将 TCSR.OSB[1:0] 位置“01b”（Low 电平输出）。

通过上述设定，能不通过软件而输出周期由 TCORA 寄存器、脉宽由 TCORB 寄存器决定的波形。

在设定 TCSR.OSA[1:0] 位或者 TCSR.OSB[1:0] 位后，到发生复位后的第 1 个比较匹配前，定时器的输出为 Low 电平。

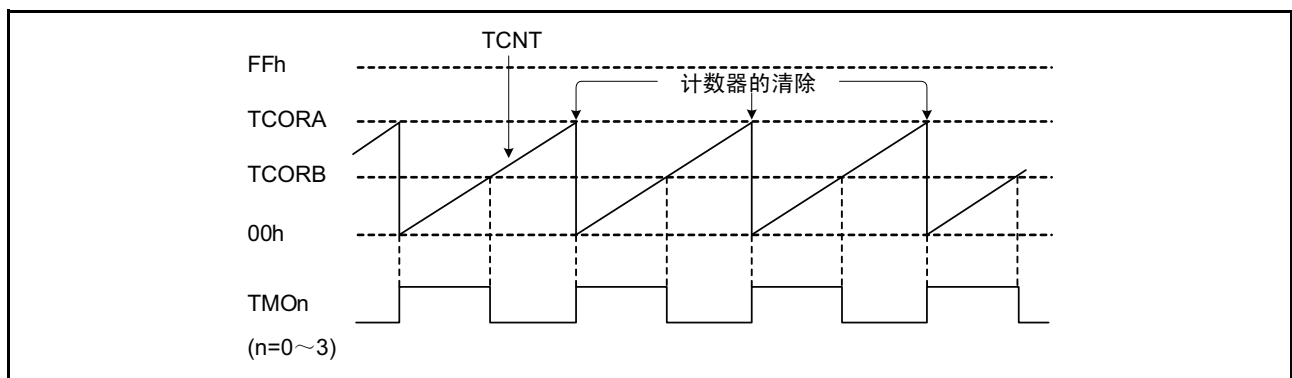


图 23.3 脉冲输出例子

23.3.2 复位输入

TMRIn 输入的任意延迟时间的脉冲输出例子如图 23.4 所示。

1. 为了通过 TMRIn 输入的 High 电平清除 TCNT 计数器，将 TCR.CCLR[1:0] 位置“11b”（通过外部复位输入进行清除）并且将 TCCR.TMRIS 位置“1”（通过外部复位的 High 电平进行清除）。
2. 为了通过 TCORA 寄存器的比较匹配进行 High 电平输出并且通过 TCORB 寄存器的比较匹配进行 Low 电平输出，将 TCSR.OSA[1:0] 位置“10b”（High 电平输出）并且将 TCSR.OSB[1:0] 位置“01b”（Low 电平输出）。

通过上述设定，能输出延迟（从 TMRIn 输入开始的延迟）由 TCORA 寄存器、脉宽由 (TCORB-TCORA) 决定的波形。

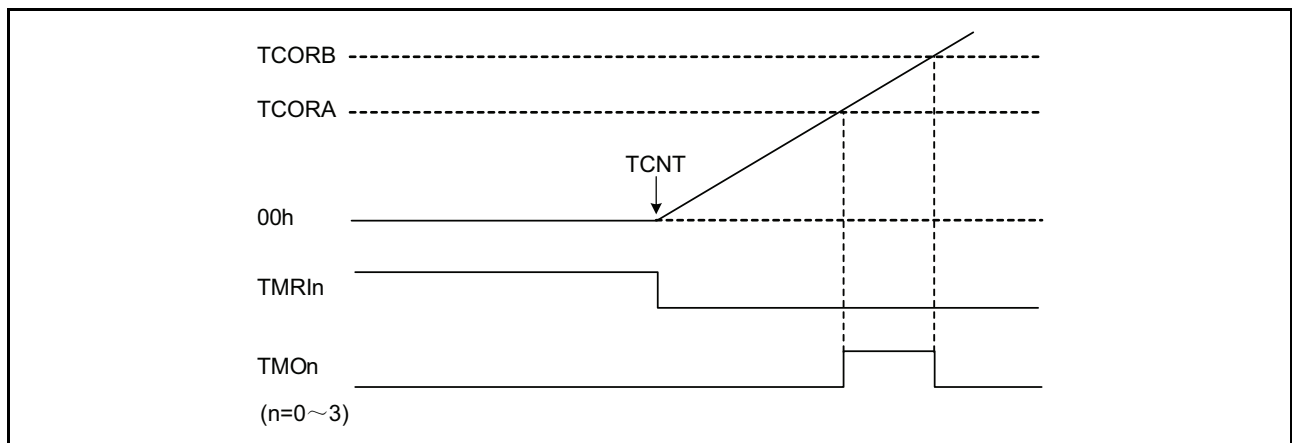


图 23.4 复位输入例子

23.4 运行时序

23.4.1 TCNT 计数器的计数时序

分频时钟和外部时钟运行时的 TCNT 计数器的计数时序分别如图 23.5 和图 23.6 所示。

必须注意：在单边沿的情况下，外部时钟的脉宽至少需要 1.5 个 PCLK；在双边沿的情况下，外部时钟的脉宽至少需要 2.5 个 PCLK。否则，就不能正常运行。

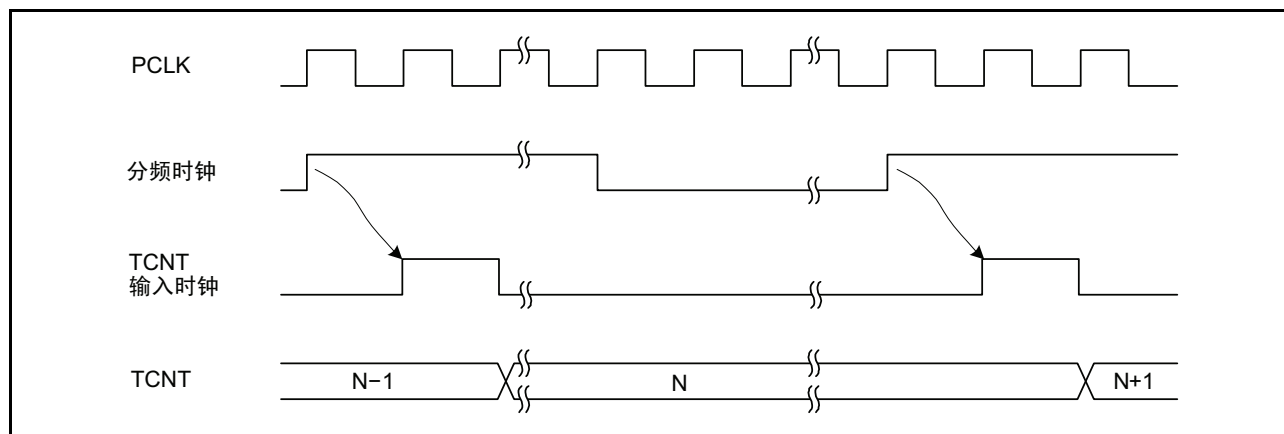


图 23.5 分频时钟运行时的计数时序

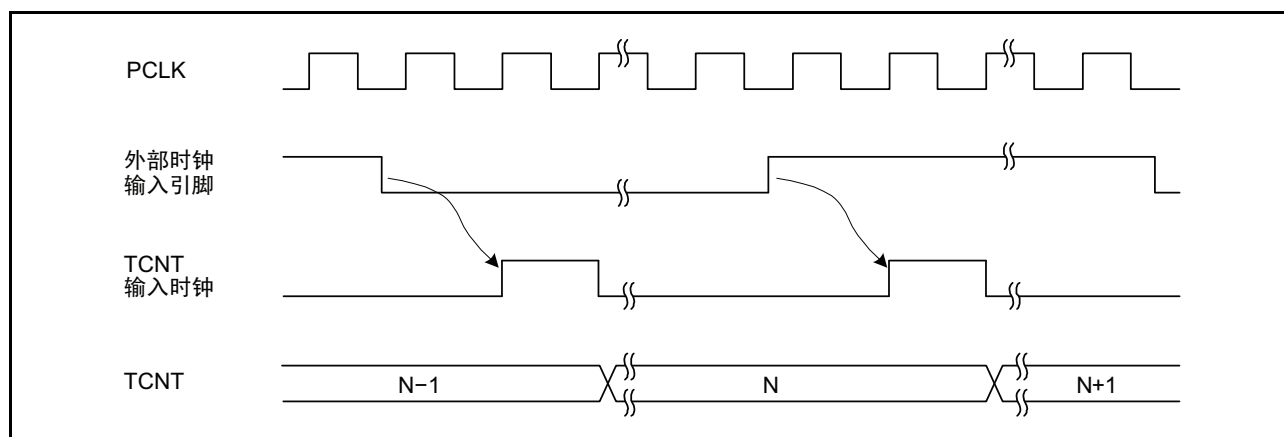


图 23.6 外部时钟运行时的计数时序（双边沿的情况）

23.4.2 比较匹配时的中断标志变为“1”的时序

如果TCORA寄存器、TCORB寄存器和TCNT计数器的值相同并且输出比较匹配信号，中断标志就变为“1”。

在TCORA寄存器、TCORB寄存器和TCNT计数器的值相同的最后PCLK（在TCNT计数器更新相同后的计数值时）产生比较匹配信号。因此，在TCNT计数器和TCORA寄存器、TCORB寄存器的值相同后，到产生TCNT计数器的输入时钟前，不产生比较匹配信号。

中断标志变为“1”的时序如图23.7所示。

对应的中断向量号请参照“14. 中断控制器 (ICUb)”和“表23.6 TMR的中断源”。

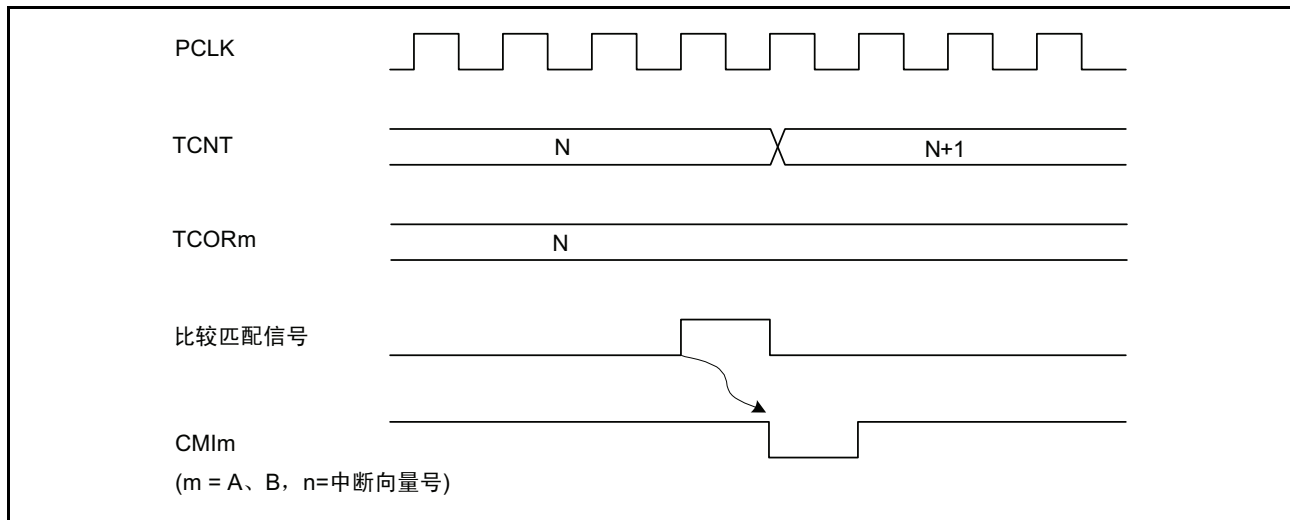


图 23.7 比较匹配时的中断标志变为“1”的时序

23.4.3 比较匹配时的定时器输出时序

在产生比较匹配信号时，将TCSR.OSA[1:0]位和TCSR.OSB[1:0]位设定的输出值输出到定时器的输出引脚。

通过比较匹配A信号进行交替输出时的定时器输出时序如图23.8所示。

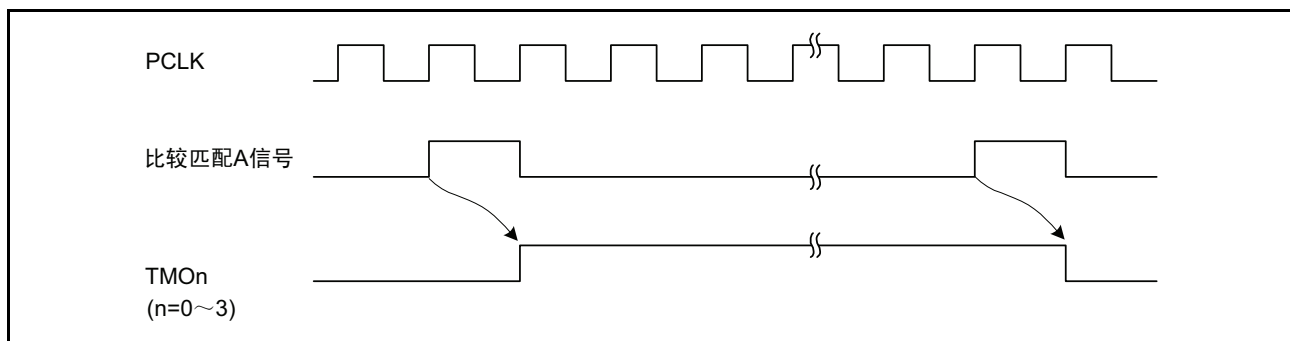


图 23.8 通过比较匹配 A 信号进行的定时器输出时序

23.4.4 通过比较匹配进行的计数器清除时序

根据 TCR.CCLR[1:0] 位的选择，通过比较匹配 A 或者比较匹配 B 进行 TCNT 计数器的清除。
通过比较匹配进行的计数器清除时序如图 23.9 所示。

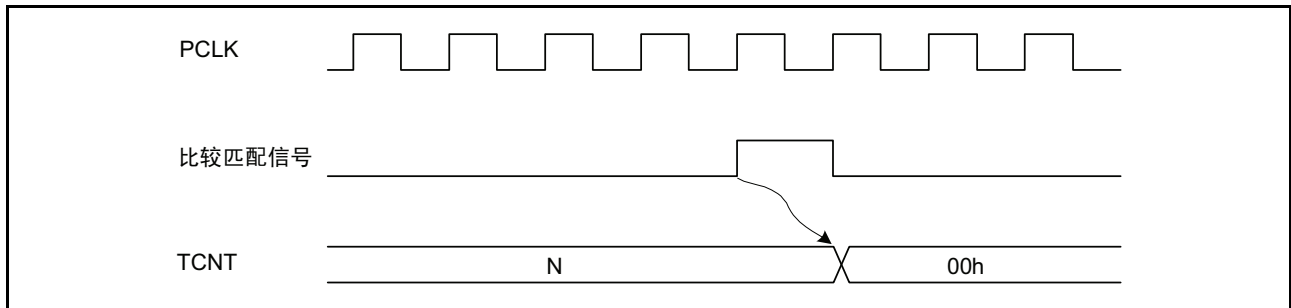


图 23.9 通过比较匹配进行的计数器清除时序

23.4.5 TCNT 计数器的外部复位时序

根据 TCRn.CCLR[1:0] 位的选择，在外部复位输入的上升沿或者通过 High 电平清除 TCNT 计数器。从输入外部复位到清除 TCNT 计数器前，至少需要 2 个 PCLK。

通过外部复位输入进行的清除时序如图 23.10 和图 23.11 所示。

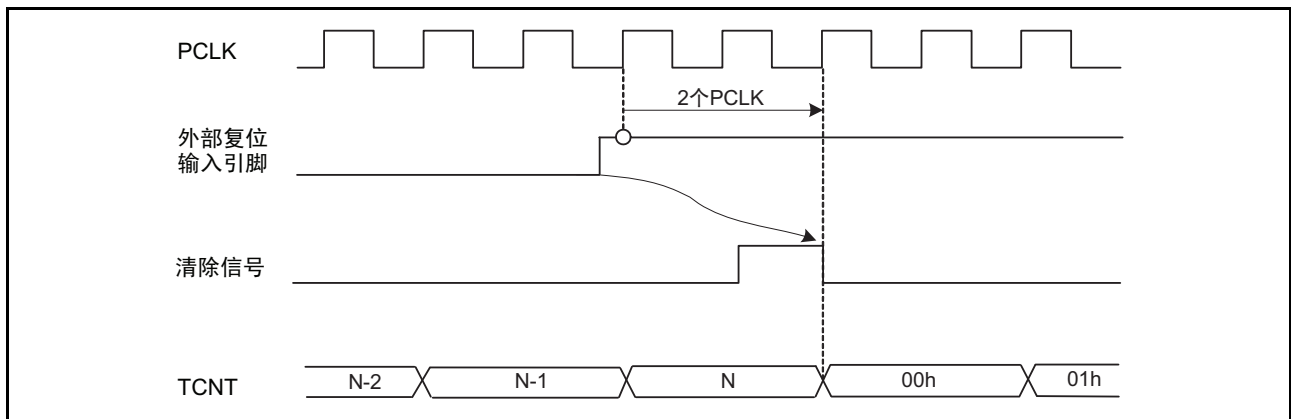


图 23.10 通过外部复位输入进行的清除时序 (上升沿)

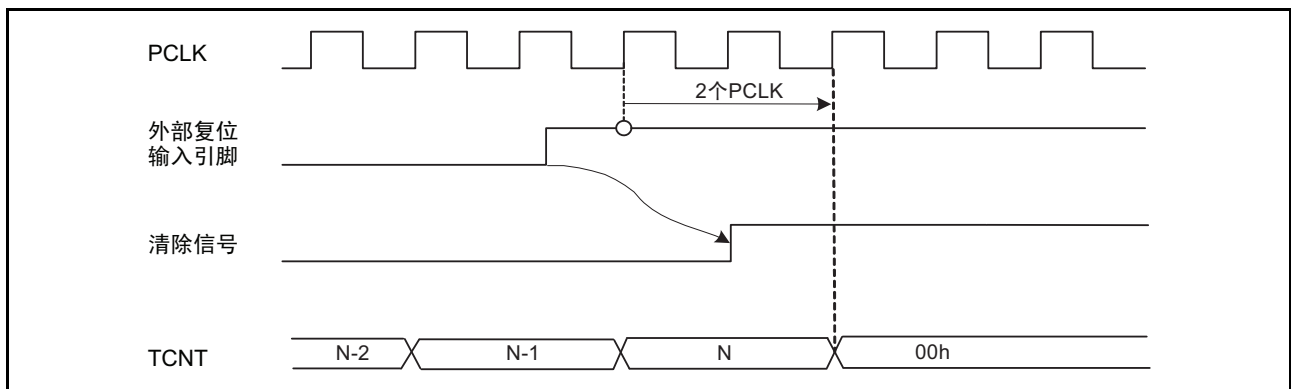


图 23.11 通过外部复位输入进行的清除时序 (High 电平)

23.4.6 通过上溢使中断标志变为“1”的时序

通过 TCNT 计数器发生上溢 (“FFh”→“00h”) 时输出的上溢信号使中断标志变为“1”。

中断标志的置位时序如图 23.12 所示。

对应的中断向量号请参照“14. 中断控制器 (ICUb)”和“表 23.6 TMR 的中断源”。

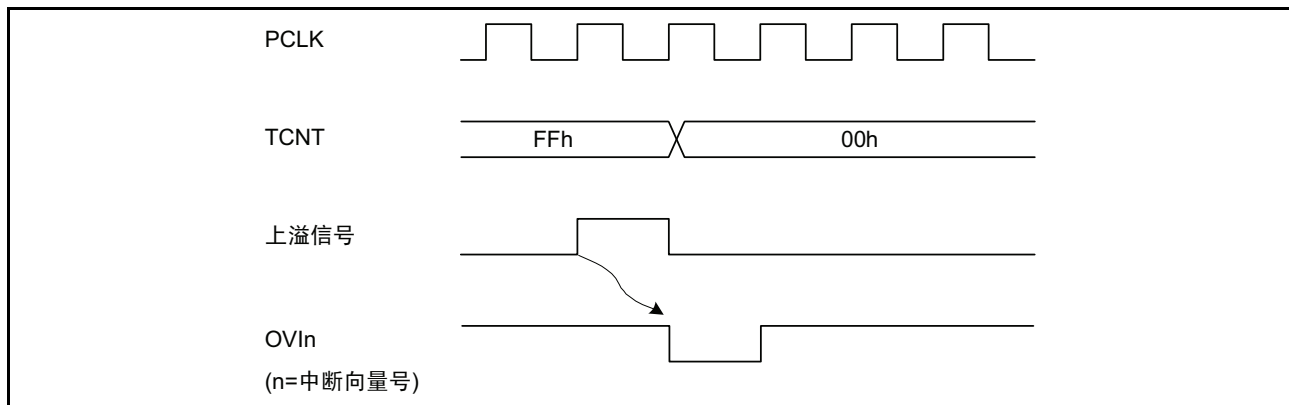


图 23.12 通过上溢信号使中断标志变为“1”的时序

23.5 级联时的运行

如果将 TMR0.TCCR 寄存器或者 TMR1.TCCR 寄存器的 CSS[1:0] 位置“11b”，就将 2 个通道的 TMR 级联。此时，可设定为 16 位计数模式（用作 1 个 16 位定时器）或者比较匹配计数模式（通过 TMR1 对 TMR0 的比较匹配进行计数）。

【补充】“23.5 级联时的运行”说明单元 0，单元 1 级联时的运行和单元 0 相同。

23.5.1 16 位计数模式

当 TMR0.TCCR.CSS[1:0] 位为“11b”时，作为 1 个通道的 16 位定时器（TMR0 为高 8 位并且 TMR1 为低 8 位）运行。

(1) 计数器清除的指定

- TMR0.TCR.CCLR[1:0] 位的设定对 16 位计数器有效。
如果在通过 TMR0.TCR.CCLR[1:0] 位设定为通过比较匹配进行计数器清除时产生 16 位比较匹配，就清除 16 位计数器（TMR0.TCNT 计数器和 TMR1.TCNT 计数器）。如果设定为通过 TMRIO 引脚进行计数器清除，也能清除 16 位计数器（TMR0.TCNT 计数器和 TMR1.TCNT 计数器）。
- TMR1.TCR.CCLR[1:0] 位的设定无效。

(2) 引脚输出

- 根据 16 位的比较匹配条件，通过 TMR0.TCSR.OSA[1:0] 位和 TMR0.TCSR.OSB[1:0] 位控制 TMO0 引脚的输出。
- 根据低 8 位的比较匹配条件，通过 TMR1.TCSR.OSA[1:0] 位和 TMR1.TCSR.OSB[1:0] 位控制 TMO1 引脚的输出。

23.5.2 比较匹配计数模式


当 TMR1.TCCR.CSS[1:0] 位为“11b”时，TMR1.TCNT 计数器对 TMR0 比较匹配 A 的发生次数进行计数。分别控制 TMR0 和 TMR1，并且根据各通道的设定，控制中断的发生、TMO_n (n=0、1) 引脚的输出和计数器的清除等。

23.6 中断源

23.6.1 中断源和 DTC 启动

TMRn 的中断源有 CMIA_n、CMIB_n 和 OVI_n 共 3 种，各中断源和优先级如表 23.6 所示。
能通过 CMIA_n 中断或者 CMIB_n 中断启动 DTC，而不能通过 TMRn 的中断源启动 DMAC。

表 23.6 TMR 的中断源

名称	中断源	DTC 的启动	优先级
CMIA0	TMR0.TCORA 的比较匹配	能	高  低
CMIB0	TMR0.TCORB 的比较匹配	能	
OVI0	TMR0.TCNT 的上溢	不能	
CMIA1	TMR1.TCORA 的比较匹配	能	
CMIB1	TMR1.TCORB 的比较匹配	能	
OVI1	TMR1.TCNT 的上溢	不能	
CMIA2	TMR2.TCORA 的比较匹配	能	
CMIB2	TMR2.TCORB 的比较匹配	能	
OVI2	TMR2.TCNT 的上溢	不能	
CMIA3	TMR3.TCORA 的比较匹配	能	
CMIB3	TMR3.TCORB 的比较匹配	能	
OVI3	TMR3.TCNT 的上溢	不能	

23.7 通过 ELC 进行的链接运行

23.7.1 向 ELC 输出事件信号

通过事件链接控制器 (ELC)，TMR 将中断请求信号用作事件信号，并且对事先设定的模块进行链接运行。TMR 输出比较匹配 A、比较匹配 B 和上溢的事件信号。对应通道为 TMR0 和 TMR2。

与对应的中断请求允许位 (TMR0.TCR.OVIE / TMR2.TCR.OVIE、TMR0.TCR.CMIEA / TMR2.TCR.CMIEA、TMR0.TCR.CMIEB / TMR2.TCR.CMIEB) 的设定无关，能输出事件信号。详细内容请参照“18. 事件链接控制器 (ELC) ”。

级联运行也支持事件输出功能。

23.7.2 通过接收 ELC 的事件信号进行的 TMR 运行

TMR 能通过事先设定的事件 (根据 ELC 的 ELSRn 寄存器的设定) 进行如下运行。但是，级联运行不支持 ELC。

(1) 开始计数运行

通过 ELC 的 ELOPD 寄存器选择开始 TMR 的计数运行。如果产生 ELSRn 寄存器指定的事件，TCSTR.TCS 位就被置“1”，并且开始 TMR 的计数。在通过 ELC 的 ELOPD 寄存器选择开始 TMR 的计数运行后，必须通过设定 TCCR.CKS[2:0] 位和 TCCR.CSS[1:0] 位选择计数源。

在 TCS 位被置“1”的状态下产生指定的事件时，该事件就变为无效。

如果要停止计数，就必须给 TCSTR.TCS 位写“0”。

如果在停止计数的状态下输入开始计数事件，就重新根据 CKS[2:0] 位和 CSS[1:0] 位进行计数。

只有在 ELC 的 ELOPD.TMR0MD 位和 ELOPD.TMR2MD 位选择开始计数运行时 TCS 位才有效。

(2) 事件计数器运行

通过 ELC 的 ELOPD 寄存器选择 TMR 的事件计数器运行。如果产生 ELSRn 寄存器指定的事件，与 TCCR.CKS[2:0] 位和 TCCR.CSS[1:0] 位的设定无关，将该事件作为计数源，进行事件计数运行。读取的计数值为实际输入的事件数。

(3) 重新开始计数运行

通过 ELC 的 ELOPD 寄存器选择重新开始 TMR 的计数运行。如果产生 ELSRn 寄存器指定的事件，TCNT 计数器的值就改写为初始值。如果 CKS[2:0] 位和 CSS[1:0] 位的设定不为“禁止时钟输入”，就能继续计数运行。

23.7.3 通过接收 ELC 的事件信号进行的 TMR 运行的注意事项

通过事件链接进行的 TMR 运行的注意事项如下所示。

(1) 开始计数运行

如果在写 TCSTR.TCS 位的过程中发生 ELSRn 寄存器指定的事件，就不写 TCSTR.TCS 位，而优先通过发生的事件将 TCSTR.TCS 位置“1”。

(2) 事件计数器运行

如果在写 TCNT 计数器的过程中发生 ELSRn 寄存器指定的事件，就不写 TCNT 计数器，而优先通过发生的事件进行计数。

(3) 重新开始计数运行

如果在写 TCNT 计数器的过程中发生 ELSRn 寄存器指定的事件，就不写 TCNT 计数器，而优先通过发生的事件对计数值进行初始化。

23.8 使用时的注意事项

23.8.1 模块停止功能的设定

能通过模块停止控制寄存器设定为禁止或者允许 TMR 的运行，初始值为停止 TMR 的运行。能通过解除模块停止状态，使寄存器变为可存取的状态。详细内容请参照“11. 低功耗功能”。

23.8.2 设定周期时的注意事项

如果设定为通过比较匹配进行计数器清除，就在 TCNT 计数器和 TCORA 寄存器、TCORB 寄存器的值相同的最后 PCLK（在 TCNT 计数器更新相同后的计数值时）清除 TCNT 计数器。因此，计数器的频率用以下表达式表示（f: 计数器的频率，PCLK: 工作频率，N: TCORA 寄存器或者 TCORB 寄存器的设定值）。

$$f = \text{PCLK} / (N + 1)$$

23.8.3 TCNT 计数器的写和计数器清除的竞争

如图 23.13 所示，如果在通过 CPU 写 TCNT 计数器的同时发生计数器清除，就不写计数器而优先清除计数器。

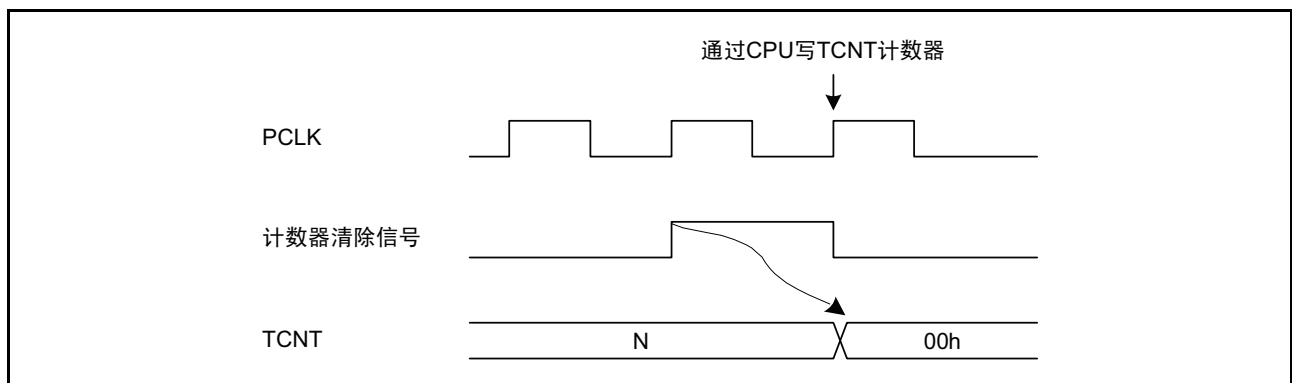


图 23.13 TCNT 计数器的写和计数器清除的竞争

23.8.4 TCNT 计数器的写和递增计数的竞争

如图 23.14 所示，即使在通过 CPU 写 TCNT 计数器的同时发生递增计数，也不进行递增计数而优先写 TCNT 计数器。

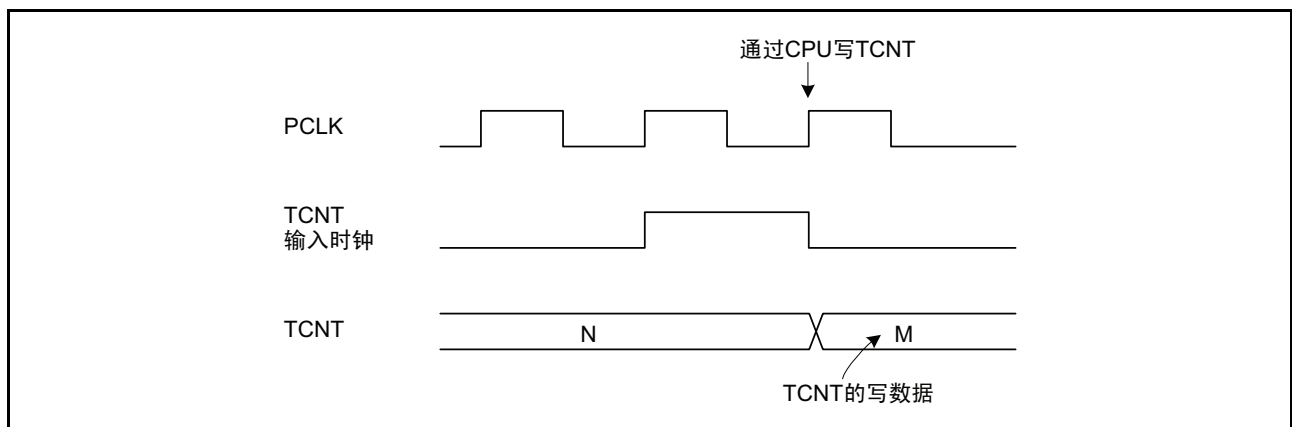


图 23.14 TCNT 计数器的写和递增计数的竞争

23.8.5 TCORA 寄存器、TCORB 寄存器的写和比较匹配的竞争

如图 23.15 所示，即使在通过 CPU 写 TCORA 寄存器或者 TCORB 寄存器的同时发生比较匹配，也优先写 TCORA 寄存器或者 TCORB 寄存器，而比较匹配信号不变为 High 电平。

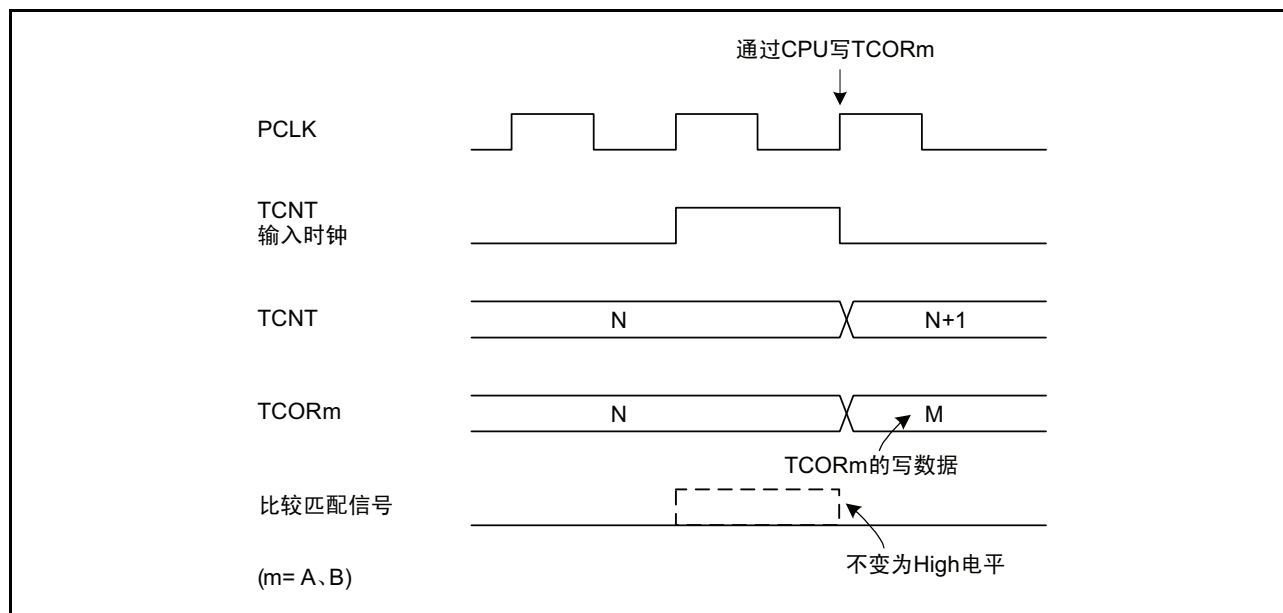


图 23.15 TCORA 寄存器、TCORB 寄存器的写和比较匹配的竞争

23.8.6 比较匹配 A 和比较匹配 B 的竞争

如果同时产生比较匹配 A 和比较匹配 B，就从比较匹配 A 和比较匹配 B 的设定运行中，进行如表 23.7 所示的优先级高的定时器输出。

表 23.7 定时器输出的优先级

输出的设定	优先级
交替输出	高 ↑ 低
High 电平输出	
Low 电平输出	
不变	

23.8.7 分频时钟的转换和 TCNT 计数器的运行

在进行分频时钟的转换时，TCNT 计数器有可能进行了递增计数。分频时钟的转换时序 (TCCR.CKS[2:0] 位的改写) 和 TCNT 计数器运行的关系如表 23.8 所示。

在分频时钟生成 TCNT 计数器的时钟的情况下，检测分频时钟的上升沿。因此，如表 23.8 的 No.2 所示，如果进行 Low 电平 →High 电平的时钟转换，就将转换时序视为边沿，产生 TCNT 计数器时钟并且 TCNT 计数器进行了递增计数。

在进行分频时钟和外部时钟的转换时，TCNT 计数器也有可能进行递增计数。

表 23.8 分频时钟的转换和 TCNT 计数器的运行 (1/2)

No	TCCR.CKS[2:0] 位的改写时序	TCNT 时钟的运行
1	Low 电平 →Low 电平 (注 1) 的转换	<p>转换前的时钟</p> <p>转换后的时钟</p> <p>TCNT 输入时钟</p> <p>TCNT</p> <p>改写TCCR.CKS[2:0]位。</p>
2	Low 电平 →High 电平 (注 2) 的转换	<p>转换前的时钟</p> <p>转换后的时钟</p> <p>TCNT 输入时钟 (注3)</p> <p>TCNT</p> <p>改写TCCR.CKS[2:0]位。</p>
3	High 电平 →Low 电平 (注 4) 的转换	<p>转换前的时钟</p> <p>转换后的时钟</p> <p>TCNT 输入时钟</p> <p>TCNT</p> <p>改写TCCR.CKS[2:0]位。</p>

表 23.8 分频时钟的转换和 TCNT 计数器的运行 (2/2)

No	TCCR.CKS[2:0] 位的改写时序	TCNT 时钟的运行
4	High 电平 → High 电平的转换	

- 注 1. 包括 Low 电平 → 停止和停止 → Low 电平的转换。
- 注 2. 包括停止 → High 电平的转换。
- 注 3. 因为将转换时序视为边沿，所以产生了 TCNT 输入时钟，并且 TCNT 进行了递增计数。
- 注 4. 包括 High 电平 → 停止的转换。

23.8.8 级联时的时钟源设定

如果同时设定 16 位计数模式和比较匹配计数模式，因为不产生 TMR0.TCNT 计数器和 TMR1.TCNT 计数器（或者 TMR2.TCNT 计数器和 TMR3.TCNT 计数器）的输入时钟，所以计数器停止后不再运行。不能进行此设定。

23.8.9 比较匹配中断的连续输出

如果将 TCORA 寄存器或者 TCORB 寄存器置“00h”并且将分频时钟设定为 PCLK/1 以及设定为通过比较匹配清除计数器，TCNT 计数器就保持“00h”而不更新，并且将比较匹配中断连续输出为电平状态。

此时，中断控制器无法检测到第 2 个以后的中断。
比较匹配中断连续输出的运行时序如图 23.16 所示。

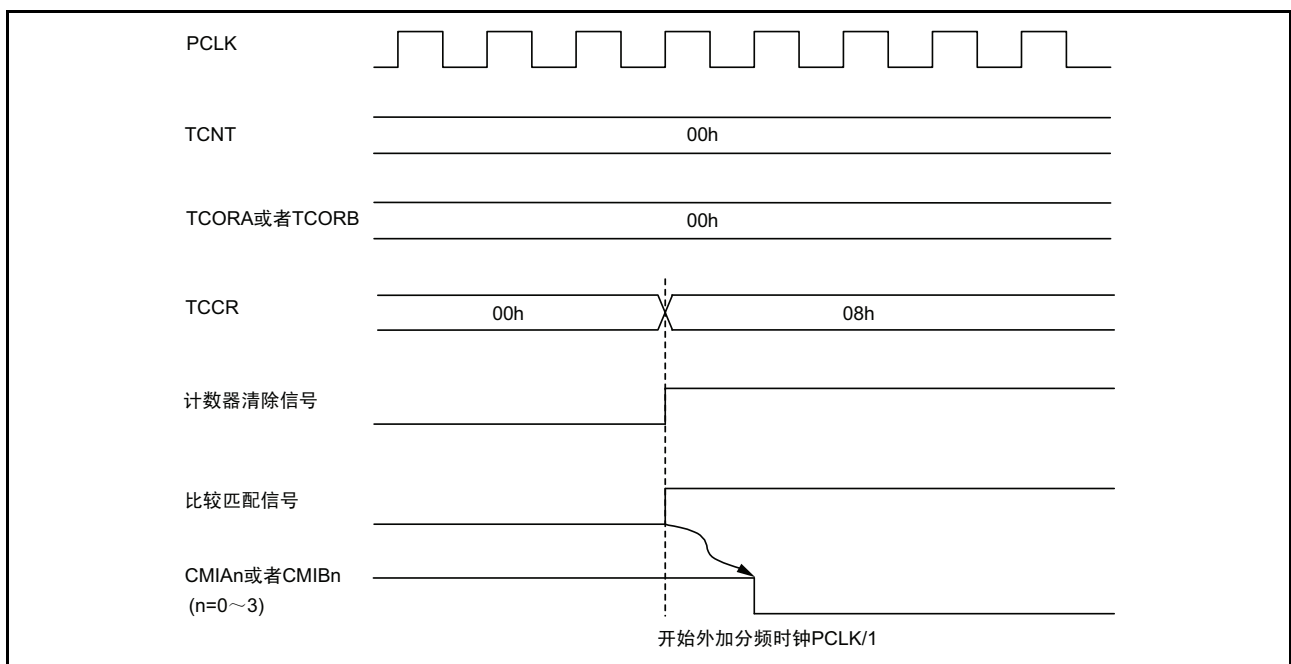


图 23.16 比较匹配中断的连续输出

24. 比较匹配定时器 (CMT)

RX210 群内置由 2 个通道的 16 位定时器构成的比较匹配定时器 (CMT)，有 2 个单元 (单元 0 和单元 1)，共计 4 个通道。CMT 有 16 位计数器，能按设定的各周期发生中断。

24.1 概要

CMT 的规格如表 24.1 所示。

CMT (单元 0) 的框图如图 24.1 所示。2 个通道的 CMT 构成 1 个单元，单元 0 和单元 1 的规格相同。

表 24.1 CMT 的规格

项目	功能
计数时钟	<ul style="list-style-type: none"> 4 种内部时钟 各通道可分别从 PCLK/8、PCLK/32、PCLK/128、PCLK/512 中选择。
中断	能分别向各通道请求比较匹配中断。
事件链接功能 (输出)	通过 CMT1 的比较匹配输出事件信号。
事件链接功能 (输入)	能对设定的模块进行链接运行。 能通过设定的事件以下运行： <ol style="list-style-type: none"> 1. 开始计数运行 2. 事件计数器运行 3. 重新开始计数运行
低功耗功能	各单元能设定为模块停止状态。

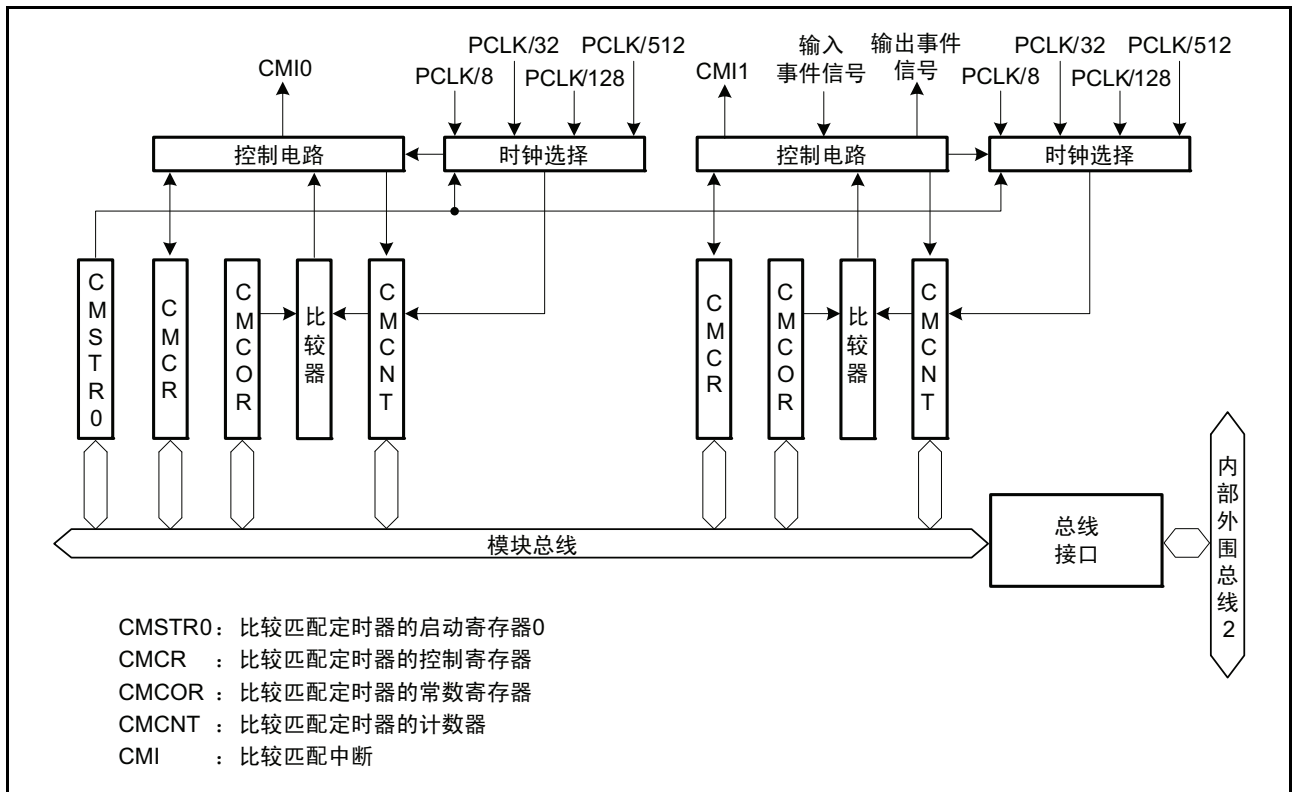


图 24.1 CMT (单元 0) 的框图

24.2 寄存器说明

24.2.1 比较匹配定时器的启动寄存器 0 (CMSTR0)

地址 0008 8000h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STR1	STR0
复位后的值	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	STR0	计数开始 0 位	0: CMT0.CMCNT 计数器停止计数 1: CMT0.CMCNT 计数器开始计数	R/W
b1	STR1	计数开始 1 位	0: CMT1.CMCNT 计数器停止计数 1: CMT1.CMCNT 计数器开始计数	R/W
b15-b2	—	保留位	读写值都为“0”。	R/W

24.2.2 比较匹配定时器的启动寄存器 1 (CMSTR1)

地址 0008 8010h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STR3	STR2
复位后的值	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	STR2	计数开始 2 位	0: CMT2.CMCNT 计数器停止计数 1: CMT2.CMCNT 计数器开始计数	R/W
b1	STR3	计数开始 3 位	0: CMT3.CMCNT 计数器停止计数 1: CMT3.CMCNT 计数器开始计数	R/W
b15-b2	—	保留位	读写值都为“0”。	R/W

24.2.3 比较匹配定时器的控制寄存器 (CMCR)

地址 CMT0.CMCR 0008 8002h、CMT1.CMCR 0008 8008h、CMT2.CMCR 0008 8012h、CMT3.CMCR 0008 8018h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	—	—	CMIE	—	—	—	—	CKS[1:0]	
0	0	0	0	0	0	0	0	x	0	0	0	0	0	0	0

复位后的值

x: 不定值

位	符号	位名	功能	R/W
b1-b0	CKS[1:0]	时钟选择位	b1 b0 0 0: PCLK/8 0 1: PCLK/32 1 0: PCLK/128 1 1: PCLK/512	R/W
b5-b2	—	保留位	读写值都为“0”。	R/W
b6	CMIE	比较匹配中断允许位	0: 禁止比较匹配中断 (CMIn) 1: 允许比较匹配中断 (CMIn)	R/W
b7	—	保留位	读取值为不定值, 只能写“1”。	R/W
b15-b8	—	保留位	读写值都为“0”。	R/W

CKS[1:0] 位 (时钟选择位)

这些位从外围模块时钟 (PCLK) 分频后得到的 4 种分频时钟中选择输入到 CMCNT 计数器的计数时钟。

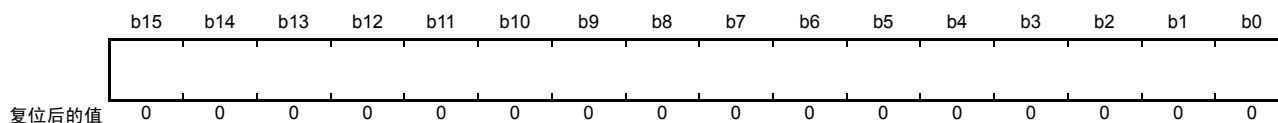
如果将 CMSTRm.STRn 位 (m=0、1, n=0~3) 置“1”, 对应的 CMCNT 计数器就通过 CKS[1:0] 位选择的时钟开始递增计数。

CMIE 位 (比较匹配中断允许位)

在 CMCNT 和 CMCOR 的值相同时, 此位选择允许或者禁止比较匹配中断 (CMIn) (n=0~3) 的发生。

24.2.4 比较匹配定时器的计数器 (CMCNT)

地址 CMT0.CMCNT 0008 8004h、CMT1.CMCNT 0008 800Ah、CMT2.CMCNT 0008 8014h、CMT3.CMCNT 0008 801Ah



CMCNT 计数器是用于产生中断请求的可读写递增计数器。

如果通过 CMCR.CKS[1:0] 位选择分频时钟并且将 CMSTRm.STRn 位 (m=0、1, n=0 ~ 3) 置“1”，CMCNT 计数器就通过该时钟开始递增计数。

如果 CMCNT 计数器的值和 CMCOR 寄存器的值相同，CMCNT 计数器就变为“0000h”，并且产生比较匹配中断 (CMI_n) (n=0 ~ 3)。

24.2.5 比较匹配定时器的常数寄存器 (CMCOR)

地址 CMT0.CMCOR 0008 8006h、CMT1.CMCOR 0008 800Ch、CMT2.CMCOR 0008 8016h、CMT3.CMCOR 0008 801Ch



CMCOR 寄存器是设定和 CMCNT 计数器的比较匹配周期的可读写寄存器。

24.3 运行说明

24.3.1 周期计数

如果通过 `CMCR.CKS[1:0]` 位选择分频时钟并且将 `CMSTRm.STRn` 位 ($m=0、1$, $n=0 \sim 3$) 置“1”，`CMCNT` 计数器就通过所选的时钟开始递增计数。

如果 `CMCNT` 计数器的值和 `CMCOR` 寄存器的值相同，`CMCNT` 计数器就变为“0000h”，并且发生比较匹配中断 (`CMIn`) ($n=0 \sim 3$)。`CMCNT` 计数器从“0000h”重新开始递增计数，`CMCNT` 计数器的运行如图 24.2 所示。

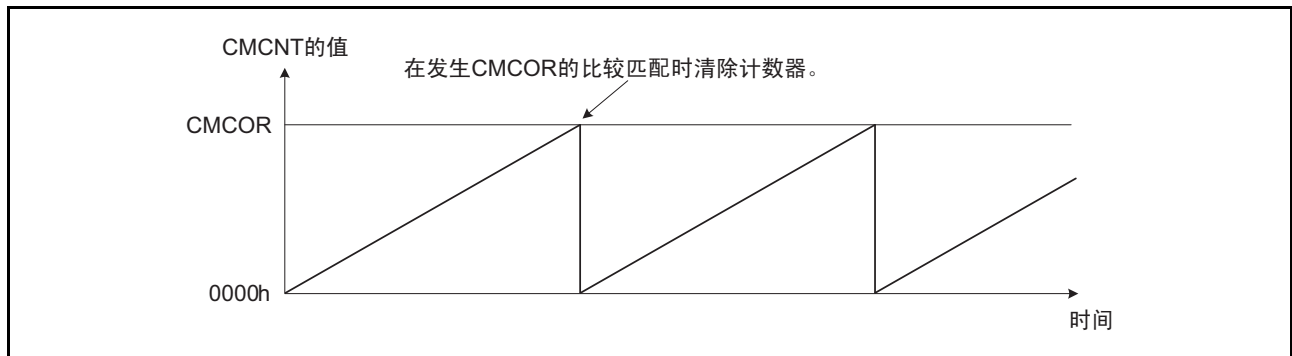


图 24.2 CMCNT 计数器的运行

24.3.2 CMCNT 计数器的计数时序

能通过 `CMCR.CKS[1:0]` 位从外围模块时钟 (`PCLK`) 分频后的 4 种分频时钟 (`PCLK/8`、`PCLK/32`、`PCLK/128`、`PCLK/512`) 中选择输入到 `CMCNT` 计数器的计数时钟，此时的 `CMCNT` 计数器的计数时序如图 24.3 所示。

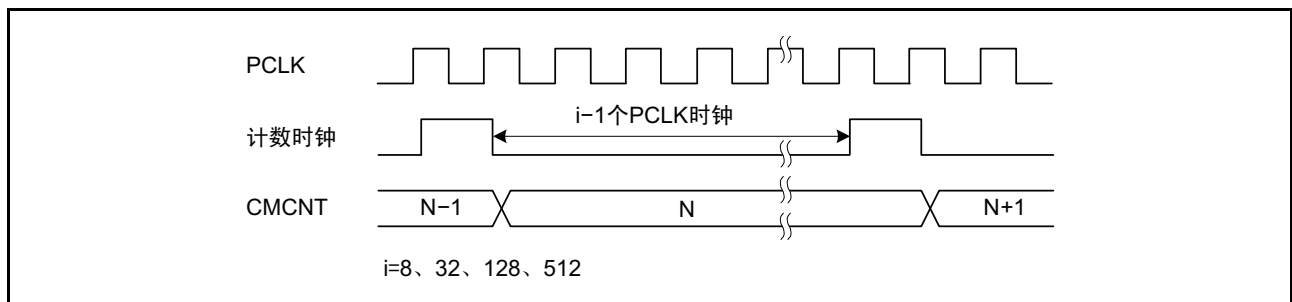


图 24.3 CMCNT 计数器的计数时序

24.4 中断

24.4.1 中断源

CMT 的各通道有比较匹配中断 (CMI_n) (n=0 ~ 3)，各中断分配有向量地址。如果发生比较匹配中断，就输出对应的中断请求。

在通过中断请求启动 CPU 中断时，能通过设定中断控制器更改通道之间的优先级，详细内容请参照“14. 中断控制器 (ICUb)”。

表 24.2 CMT 的中断源

名称	中断源	DTC 的启动	DMAC 的启动
CMI0	CMT0.CMCNT 和 CMT0.CMCOR 的比较匹配	能	能
CMI1	CMT1.CMCNT 和 CMT1.CMCOR 的比较匹配	能	能
CMI2	CMT2.CMCNT 和 CMT2.CMCOR 的比较匹配	能	能
CMI3	CMT3.CMCNT 和 CMT3.CMCOR 的比较匹配	能	能

24.4.2 比较匹配中断的发生时序

在 CMCNT 计数器的值和 CMCOR 寄存器的值相同时，发生比较匹配中断 (CMI_n) (n=0 ~ 3)。

在与 CMCNT 计数器相同的最后状态 (CMCNT 计数器更新相同的计数值时)，产生比较匹配信号。因此，从 CMCNT 计数器和 CMCOR 寄存器的值相同后到产生 CMCNT 输入时钟前，不产生比较匹配信号。

中断标志变为“1”的时序如图 24.4 所示。

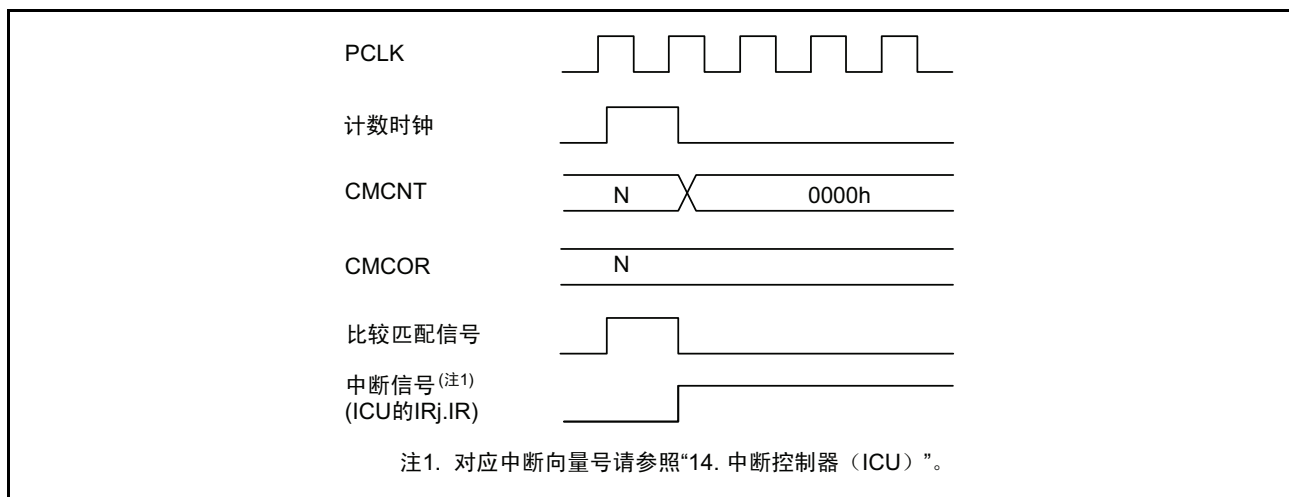


图 24.4 比较匹配中断标志变为“1”的时序

24.5 通过 ELC 进行的链接运行

24.5.1 向 ELC 输出事件信号

通过事件链接控制器 (ELC)，CMT 将中断请求信号用作事件信号，并且对事先设定的模块进行链接运行。通过 CMT1 的比较匹配输出事件信号。

与对应的中断请求允许位 (CMTn.CMCR.CMIE 位) 的设定无关，能输出事件信号。

24.5.2 通过接收 ELC 的事件信号进行的 CMT 运行

CMT 能通过事先设定的事件 (根据 ELC 的 ELSR7 寄存器的设定) 进行如下运行。

(1) 开始计数运行

通过 ELC 的 ELOPC 寄存器选择开始 CMT 的计数运行。如果产生 ELSR7 寄存器指定的事件，CMSTR0.STR1 位就变为“1”，并且开始 CMT 的计数。

但是，如果在 CMSTR0.STR1 位为“1”的状态下产生指定的事件，该事件就变为无效。

(2) 事件计数器运行

通过 ELC 的 ELOPC 寄存器选择 CMT 的事件计数器运行。如果在 CMSTR0.STR1 位为“1”的状态下产生 ELSR7 寄存器指定的事件，与 CMT1.CMCR.CKS[1:0] 位的设定无关，就将该事件作为计数源，进行事件计数运行。如果读计数值，就读取实际输入的事件数。

(3) 重新开始计数运行

通过 ELC 的 ELOPC 寄存器选择重新开始 CMT 的计数运行。如果产生 ELSR7 寄存器指定的事件，就将 CMT1.CMCNT 计数器的值改写为初始值。如果 CMSTR0.STR1 位为“1”，就能继续计数运行。

24.5.3 通过接收 ELC 的事件信号进行的 CMT 运行的注意事项

通过事件链接进行的 CMT 运行的注意事项如下所示。

(1) 开始计数运行

如果在写 CMSTR0.STR1 位的过程中发生 ELSR7 寄存器指定的事件，就不写 CMSTR0.STR1 位，而优先通过发生的事件将 CMSTR0.STR1 位置“1”。

(2) 事件计数器运行

如果在写 CMT1.CMCNT 计数器的过程中发生 ELSR7 指定的事件，就不写 CMT1.CMCNT 寄存器，而优先通过发生的事件进行计数。

(3) 重新开始计数运行

如果在写 CMT1.CMCNT 计数器的过程中发生 ELSR7 指定的事件，就不写 CMT1.CMCNT 计数器，而优先通过发生的事件对计数值进行初始化。

24.6 使用时的注意事项

24.6.1 模块停止功能的设定

能通过模块停止控制寄存器设定禁止或者允许 CMT 的运行，初始值为停止 CMT 的运行。能通过解除模块停止状态，使寄存器变为可存取的状态。详细内容请参照“11. 低功耗功能”。

24.6.2 CMCNT 计数器的写和比较匹配的竞争

如果在写 CMCNT 计数器的过程中产生比较匹配信号，就不写 CMCNT 计数器而优先清除 CMCNT 计数器。此时序如图 24.5 所示。

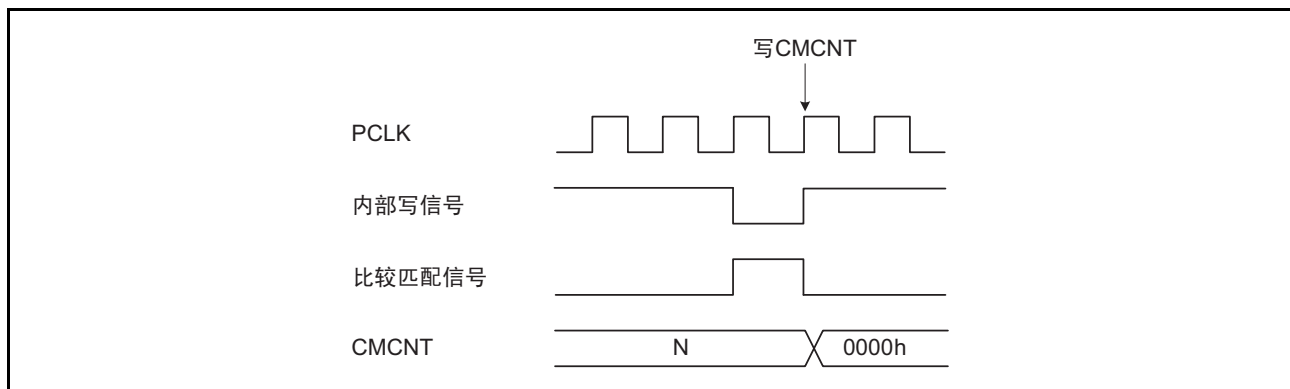


图 24.5 CMCNT 计数器的写和比较匹配的竞争

24.6.3 CMCNT 计数器的写和递增计数的竞争

即使在写 CMCNT 计数器的过程中发生递增计数，CMCNT 计数器也不进行递增计数而优先写 CMCNT 计数器。此时序如图 24.6 所示。

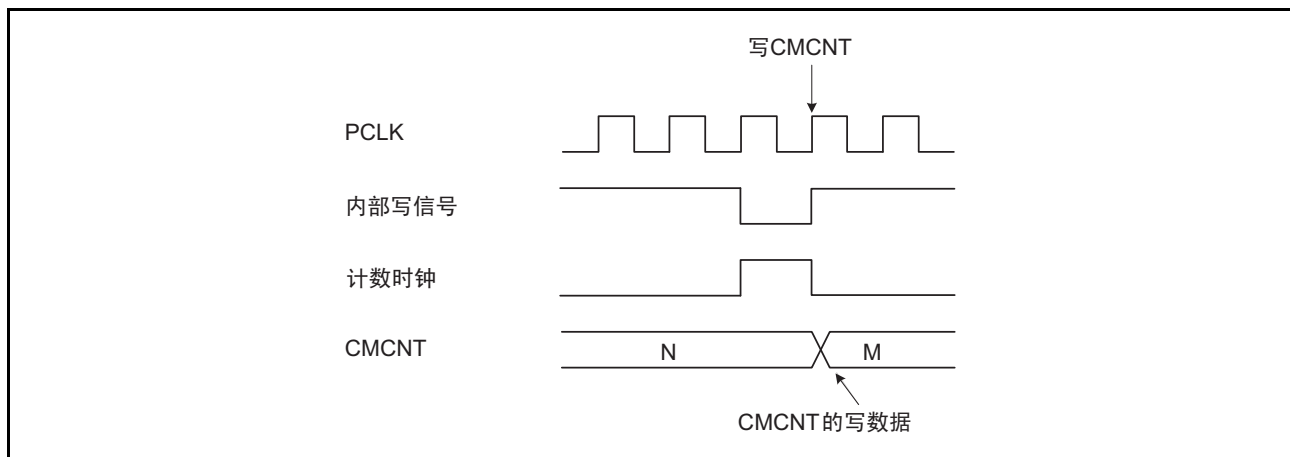


图 24.6 CMCNT 计数器的写和递增计数的竞争

25. 实时时钟 (RTCb)

25.1 概要

RTC 是能对 00 年～99 年的 100 年进行计数的时钟计数器。将年的千位和百位视为“20”，自动校正 2000 年～2099 年的闰年并且进行计数。

RTC 以通过预分频器将计数源分频后的 128Hz 时钟为基准时钟，并且以年、月、日、星期、上午 / 下午（12 小时模式时）、小时、分钟、秒和 1/128 秒为单位进行计数。

RTC 的规格和输入 / 输出引脚分别如表 25.1 和表 25.2 所示，RTC 的框图如图 25.1 所示。

表 25.1 RTC 的规格

项目	内容
计数源	副时钟 (XCIN)
时钟 / 日历功能	<ul style="list-style-type: none"> • 对年、月、日、星期、小时、分钟、秒进行计数和 BCD 显示。 • 用二进制显示 1Hz、2Hz、4Hz、8Hz、16Hz、32Hz、64Hz 的状态。 • 12 小时 / 24 小时模式转换功能 • 开始 / 停止功能 • 30 秒调整功能（在秒数小于 30 秒的情况下，舍去秒数，为 00 秒；在秒数大于等于 30 秒的情况下，向 1 分钟进位） • 闰年自动校正功能 • 1Hz 时钟输出 • 时钟误差校正功能
中断	<ul style="list-style-type: none"> • 闹钟中断 (ALM) 闹钟中断条件可选择与年、月、日、星期、小时、分钟或者秒进行比较。 • 周期中断 (PRD) 中断周期可选择 2 秒、1 秒、1/2 秒、1/4 秒、1/8 秒、1/16 秒、1/32 秒、1/64 秒、1/128 秒或者 1/256 秒周期。 • 进位中断 (CUP) 表示向秒计数器进位或者在读 64Hz 计数器时向 64Hz 计数器进位。 • 能通过闹钟中断或者周期中断从软件待机或者深度软件待机返回。
时间捕捉功能	<ul style="list-style-type: none"> • 能通过 3 个事件输入来捕捉时间。 按每个事件的输入，捕捉月、日、小时、分钟和秒。
事件链接功能	周期事件的输出

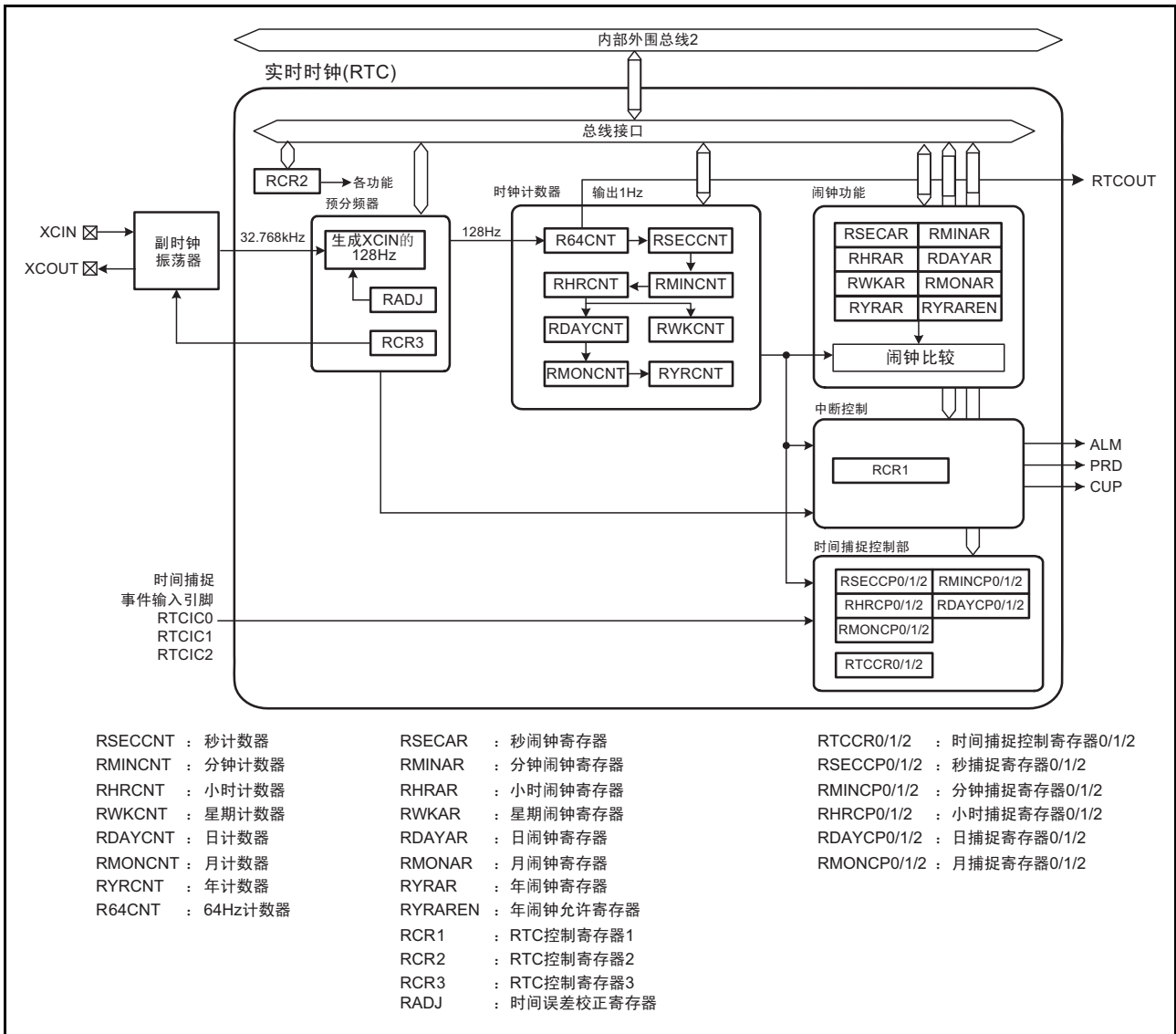


图 25.1 RTC 的框图

表 25.2 RTC 的输入 / 输出引脚

引脚名	输入 / 输出	功能
XCIN	输入	连接用于 RTC 的 32.768kHz 晶体谐振器，XCIN 引脚还能输入外部时钟。
XCOUT	输出	
RTCOUT	输出	输出 1Hz 时钟，但是不在深度软件待机时进行输出。
RTCIC0	输入	时间捕捉的事件输入引脚
RTCIC1	输入	
RTCIC2	输入	

25.2 寄存器说明

必须遵循“25.6.5 读写寄存器时的注意事项”读写 RTC 的寄存器。

在复位或者深度软件待机模式中，不对 RTC 寄存器的位中复位后的值为 x（不定值）的位进行初始化。在计数过程中（RCR2.START 位 =1）转移到复位状态或者低功耗状态时，年、月、星期、日、小时、分钟、秒和 64Hz 的计数器继续运行。但是，必须注意：如果在写寄存器以及对寄存器进行更新处理的过程中发生复位，就可能破坏寄存器的值。另外，不能在设定寄存器后立即转移到软件待机模式和深度软件待机模式，详细内容请参照“25.6.4 有关在设定寄存器后向低功耗模式的转移”。

25.2.1 64Hz 计数器 (R64CNT)

地址 0008 C400h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	F1HZ	F2HZ	F4HZ	F8HZ	F16HZ	F32HZ	F64HZ
复位后的值	0	x	x	x	x	x	x	x

x: 不定值

位	符号	位名	功能	R/W
b0	F64HZ	64Hz 位	表示 1Hz ~ 64Hz 的状态。	R
b1	F32HZ	32Hz 位		R
b2	F16HZ	16Hz 位		R
b3	F8HZ	8Hz 位		R
b4	F4HZ	4Hz 位		R
b5	F2HZ	2Hz 位		R
b6	F1HZ	1Hz 位		R
b7	—	保留位	读取值为“0”，写操作无效。	R

R64CNT 计数器通过 128Hz 时钟进行递增计数，生成秒周期。

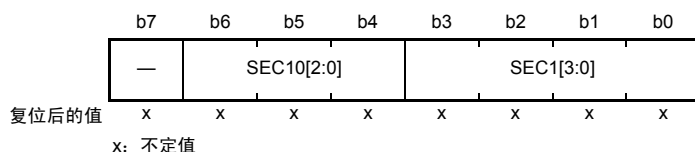
能通过读 R64CNT 计数器，确认秒以下的状态。

如果执行 RTC 软件复位或者 30 秒调整，R64CNT 就变为“00h”。

在读 R64CNT 时，必须遵循“25.3.5 64Hz 计数器和时间的读取步骤”。

25.2.2 秒计数器 (RSECCNT)

地址 0008 C402h



位	符号	位名	功能	R/W
b3-b0	SEC1[3:0]	秒的个位计数位	秒的个位按秒进行 0 ~ 9 的计数。如果发生进位，秒的十位就加 1。	R/W
b6-b4	SEC10[2:0]	秒的十位计数位	秒的十位进行 0 ~ 5 的计数，是 60 秒计数。	R/W
b7	—	保留位	必须置“0”，读取值为“0”。	R/W

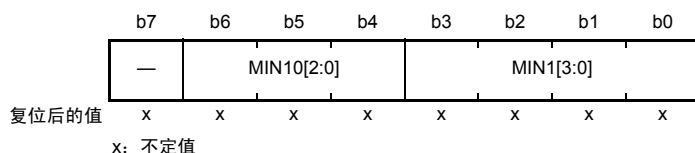
RSECCNT 计数器是对秒部分 (BCD 码) 进行设定和计数的计数器，通过 64Hz 计数器按秒进位进行计数。

可设定的范围是 10 进制 (BCD) 的 00 ~ 59。如果设定其他的值，就不正常运行。另外，必须在通过开始位 (RCR2.START) 停止计数后写此计数器。

在读 RSECCNT 时，必须遵循“25.3.5 64Hz 计数器和时间的读取步骤”。

25.2.3 分钟计数器 (RMINCNT)

地址 0008 C404h



位	符号	位名	功能	R/W
b3-b0	MIN1[3:0]	分钟个位计数位	分钟个位按分钟进行 0 ~ 9 的计数。如果发生进位，分钟的十位就加 1。	R/W
b6-b4	MIN10[2:0]	分钟的十位计数位	分钟的十位进行 0 ~ 5 的计数，是 60 分钟计数。	R/W
b7	—	保留位	必须置“0”，读取值为“0”。	R/W

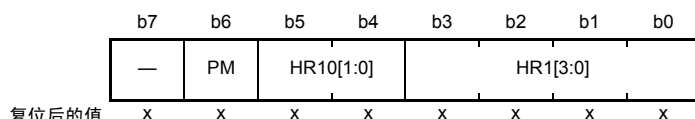
RMINCNT 计数器是对分钟部分 (BCD 码) 进行设定和计数的计数器，通过秒计数器按分钟进位进行计数。

可设定的范围是 10 进制 (BCD) 的 00 ~ 59。如果设定其他的值，就不正常运行。另外，必须在通过开始位 (RCR2.START) 停止计数后写此计数器。

在读 RMINCNT 时，必须遵循“25.3.5 64Hz 计数器和时间的读取步骤”。

25.2.4 小时计数器 (RHRCNT)

地址 0008 C406h



复位后的值

x: 不定值

位	符号	位名	功能	R/W
b3-b0	HR1[3:0]	小时的个位计数位	小时的个位按小时进行 0 ~ 9 的计数。如果发生进位、小时的十位就加 1。	R/W
b5-b4	HR10[1:0]	小时的十位计数位	小时的个位每发生一次进位, 小时的十位就进行 0 ~ 2 的计数。	R/W
b6	PM	PM 位	设定小时计数器的 AM/PM。 0: 上午 1: 下午	R/W
b7	—	保留位	必须置“0”, 读取值为“0”。	R/W

RHRCNT计数器是对小时部分(BCD码)进行设定和计数的计数器,通过分钟计数器按小时进位进行计数。

根据时间模式位(RCR2.HR24),小时的可设定范围分别如下:

当RCR2.HR24位为“0”时,为10进制(BCD)的00~11。

当RCR2.HR24位为“1”时,为10进制(BCD)的00~23。

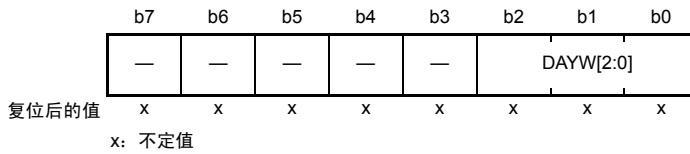
如果设定上述以外的值,就不正常运行。另外,必须在通过开始位(RCR2.START)停止计数后写此计数器。

在读RHRCNT计数器时,PM位只在RCR2.HR24位为“0”时有效。当RCR2.HR24位为“1”时,必须忽视PM位的值。

在读RHRCNT时,必须遵循“25.3.5 64Hz计数器和时间的读取步骤”。

25.2.5 星期计数器 (RWKCNT)

地址 0008 C408h



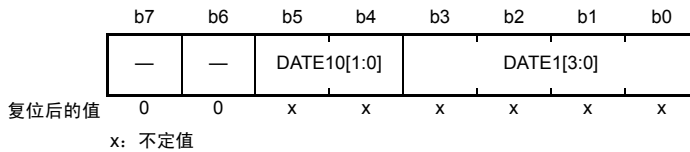
位	符号	位名	功能	R/W
b2-b0	DAYW[2:0]	星期计数位	b2 b0 0 0 0: 星期日 0 0 1: 星期一 0 1 0: 星期二 0 1 1: 星期三 1 0 0: 星期四 1 0 1: 星期五 1 1 0: 星期六 1 1 1: 不能设定	R/W
b7-b3	—	保留位	必须置“0”，读取值为“0”。	R/W

RWKCNT 计数器是对星期部分 (BCD 码) 进行设定和计数的计数器，通过小时计数器按日进位进行计数。可设定的范围是 10 进制 (BCD) 的 0 ~ 6。如果设定其他的值，就不正常运行。另外，必须在通过开始位 (RCR2.START) 停止计数后写此计数器。

在读 RWKCNT 时，必须遵循“25.3.5 64Hz 计数器和时间的读取步骤”。

25.2.6 日计数器 (RDAYCNT)

地址 0008 C40Ah



位	符号	位名	功能	R/W
b3-b0	DATE1[3:0]	日的个位计数位	日的个位按日进行 0 ~ 9 的计数。如果发生进位，日的十位就加 1。	R/W
b5-b4	DATE10[1:0]	日的十位计数位	日的个位每发生一次进位，日的十位就进行 0 ~ 3 的计数。	R/W
b7-b6	—	保留位	读写值都为“0”。	R/W

RDAYCNT 计数器是对日部分 (BCD 码) 进行设定和计数的计数器，通过小时计数器按日进位进行计数，还进行与闰年和月对应的计数。

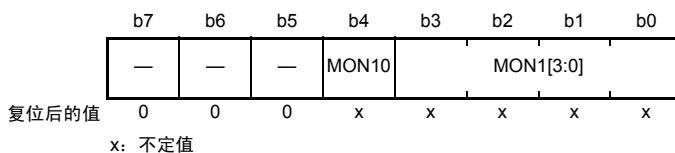
闰年是将年计数器 (RYRCNT) 的 00 视为 2000 年，根据是否能被 400、100 或者 4 整除，计算 2000 年~2099 年的闰年。

可设定的范围是 10 进制 (BCD) 的 01 ~ 31。如果设定其他的值，就不正常运行 (可设定的范围因月和闰年而不同，必须在确认后设定)。另外，必须在通过开始位 (RCR2.START) 停止计数后写此计数器。

在读 RDAYCNT 时，必须遵循“25.3.5 64Hz 计数器和时间的读取步骤”。

25.2.7 月计数器 (RMONCNT)

地址 0008 C40Ch



位	符号	位名	功能	R/W
b3-b0	MON1[3:0]	月的个位计数位	月的个位按月进行 0 ~ 9 的计数。如果发生进位，月的十位就加 1。	R/W
b4	MON10	月的十位计数位	月的个位每发生一次进位，月的十位就进行 0 ~ 1 的计数。	R/W
b7-b5	—	保留位	读写值都为“0”。	R/W

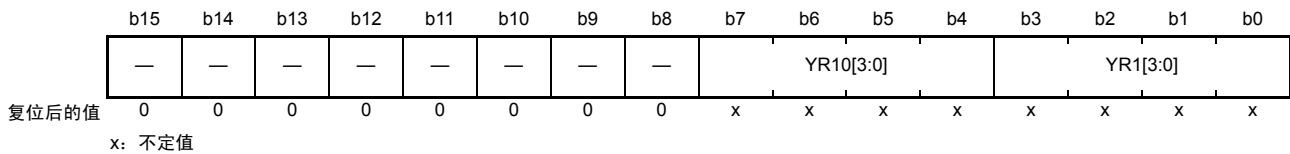
RMONCNT 计数器是对月部分 (BCD 码) 进行设定和计数的计数器，通过日计数器按月进位进行计数。

可设定的范围是 10 进制 (BCD) 的 01 ~ 12。如果设定其他的值，就不正常运行。另外，必须在通过开始位 (RCR2.START) 停止计数后写此计数器。

在读 RMONCNT 时，必须遵循“25.3.5 64Hz 计数器和时间的读取步骤”。

25.2.8 年计数器 (RYRCNT)

地址 0008 C40Eh



位	符号	位名	功能	R/W
b3-b0	YR1[3:0]	年的个位计数位	年的个位按年进行 0 ~ 9 的计数。如果发生进位, 年的十位就加 1。	R/W
b7-b4	YR10[3:0]	年的十位计数位	年的个位每发生一次进位, 年的十位就进行 0 ~ 9 的计数。	R/W
b15-b8	—	保留位	读写值都为“0”。	R/W

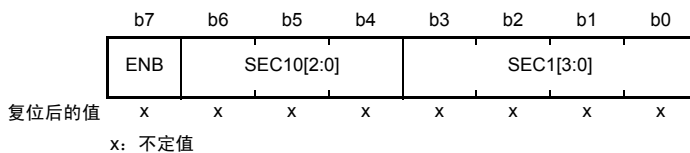
RYRCNT 计数器是对年部分 (BCD 码) 进行设定和计数的计数器, 通过月计数器按年进位进行计数。

可设定的范围是 10 进制 (BCD) 的 00 ~ 99。如果设定其他的值, 就不正常运行。另外, 必须在通过开始位 (RCR2.START) 停止计数后写此计数器。

在读 RYRCNT 时, 必须遵循“25.3.5 64Hz 计数器和时间的读取步骤”。

25.2.9 秒闹钟寄存器 (RSECAR)

地址 0008 C410h



位	符号	位名	功能	R/W
b3-b0	SEC1[3:0]	1 秒位	秒的个位设定值	R/W
b6-b4	SEC10[2:0]	10 秒位	秒的十位设定值	R/W
b7	ENB	ENB 位	如果为“1”, 就和 RSECCNT 计数器的值进行比较。	R/W

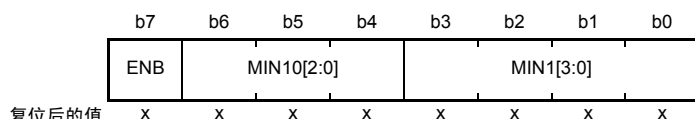
RSECAR 寄存器是与秒部分 (BCD 码) 的计数器 (RSECCNT) 对应的闹钟寄存器。如果 ENB 位为“1”, 就将 RSECAR 寄存器的值和 RSECCNT 计数器的值进行比较。只对 ENB 位为“1”的闹钟寄存器 (RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RYRAREN) 进行计数器和闹钟寄存器的比较, 当各自值全部相同时, ICU 的 IR92.IR 标志变为“1”。

秒的可设定范围是 10 进制 (BCD) 的 00 ~ 59。如果设定其他的值, 就不正常运行。

如果执行 RTC 软件复位, RSECAR 寄存器就变为“00h”。

25.2.10 分钟闹钟寄存器 (RMINAR)

地址 0008 C412h



复位后的值 x x x x x x x x

x: 不定值

位	符号	位名	功能	R/W
b3-b0	MIN1[3:0]	1 分钟位	分钟个位设定值	R/W
b6-b4	MIN10[2:0]	10 分钟位	分钟十位设定值	R/W
b7	ENB	ENB 位	如果为“1”，就和 RMINCNT 计数器的值进行比较。	R/W

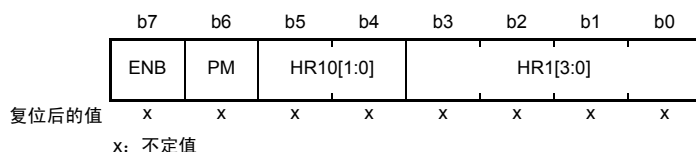
RMINAR 寄存器是与分钟部分 (BCD 码) 的计数器 (RMINCNT) 对应的闹钟寄存器。如果 ENB 位为“1”，就将 RMINAR 寄存器的值和 RMINCNT 计数器的值进行比较。只对 ENB 位为“1”的闹钟寄存器 (RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RYRAREN) 进行计数器和闹钟寄存器的比较，当各自的值全部相同时，ICU 的 IR92.IR 标志变为“1”。

分钟的可设定范围是 10 进制 (BCD) 的 00 ~ 59 位。如果设定其他的值，就不正常运行。

如果执行 RTC 软件复位，RMINAR 寄存器就变为“00h”。

25.2.11 小时闹钟寄存器 (RHRAR)

地址 0008 C414h



位	符号	位名	功能	R/W
b3-b0	HR1[3:0]	1 小时位	小时的个位设定值	R/W
b5-b4	HR10[1:0]	10 小时位	小时的十位设定值	R/W
b6	PM	PM 位	设定小时闹钟的 AM/PM。 0: 上午 1: 下午	R/W
b7	ENB	ENB 位	如果为“1”，就和 RHCNT 计数器的值进行比较。	R/W

RHRAR 寄存器是与小时部分 (BCD 码) 的计数器 (RHCNT) 对应的闹钟寄存器。如果 ENB 位为“1”，就将 RHRAR 寄存器的值和 RHCNT 计数器的值进行比较。只对 ENB 位为“1”的闹钟寄存器 (RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RYRAREN) 进行计数器和闹钟寄存器的比较，当各自的值全部相同时，ICU 的 IR92.IR 标志变为“1”。

根据时间模式位 (RCR2.HR24)，小时的可设定范围分别如下：

当 RCR2.HR24 位为“0”时，为 10 进制 (BCD) 的 00 ~ 11。

当 RCR2.HR24 位为“1”时，为 10 进制 (BCD) 的 00 ~ 23。

如果设定上述以外的值，就不正常运行。

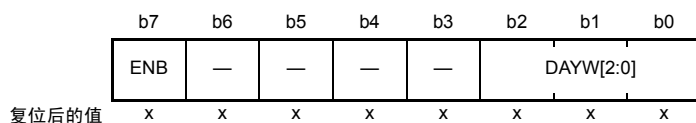
当 RCR2.HR24 位为“0”时，也必须设定 PM 位。

当 RCR2.HR24 位为“1”时，PM 位的值无效。

如果将执行 RTC 软件复位，RHRAR 寄存器就变为“00h”。

25.2.12 星期闹钟寄存器 (RWKAR)

地址 0008 C416h



复位后的值

x: 不定值

位	符号	位名	功能	R/W
b2-b0	DAYW[2:0]	星期的设定值位	b2 b0 0 0 0: 星期日 0 0 1: 星期一 0 1 0: 星期二 0 1 1: 星期三 1 0 0: 星期四 1 0 1: 星期五 1 1 0: 星期六 1 1 1: 不能设定	R/W
b6-b3	—	保留位	必须置“0”，读取值为“0”。	R/W
b7	ENB	ENB 位	如果为“1”，就和 RWKCNT 计数器的值进行比较。	R/W

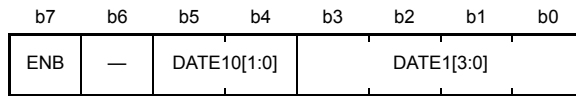
RWKAR 寄存器是与星期部分 (BCD 码) 的计数器 (RWKCNT) 对应的闹钟寄存器。如果 ENB 位为“1”，就将 RWKAR 寄存器的值和 RWKCNT 计数器的值进行比较。只对 ENB 位为“1”的闹钟寄存器 (RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RYRAREN) 进行计数器和闹钟寄存器的比较，当各自的值全部相同时，ICU 的 IR92.IR 标志变为“1”。

星期的可设定范围是 10 进制 (BCD) 的 0 ~ 6。如果设定其他的值，就不正常运行。

如果执行 RTC 软件复位，RWKAR 寄存器就变为“00h”。

25.2.13 日闹钟寄存器 (RDAYAR)

地址 0008 C418h



复位后的值

x: 不定值

位	符号	位名	功能	R/W
b3-b0	DATE1[3:0]	1 日位	日的个位设定值	R/W
b5-b4	DATE10[1:0]	10 日位	日的十位设定值	R/W
b6	—	保留位	必须置“0”，读取值为“0”。	R/W
b7	ENB	ENB 位	如果为“1”，就和 RDAYCNT 计数器的值进行比较。	R/W

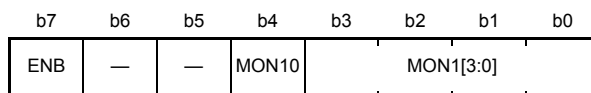
RDAYAR 寄存器是与日部分 (BCD 码) 的计数器 (RDAYCNT) 对应的闹钟寄存器。如果 ENB 位为“1”，就将 RDAYAR 寄存器的值和 RDAYCNT 计数器的值进行比较。只对 ENB 位为“1”的闹钟寄存器 (RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RYRAREN) 进行计数器和闹钟寄存器的比较，当各自的值全部相同时，ICU 的 IR92.IR 标志变为“1”。

日的可设定范围是 10 进制 (BCD) 的 01 ~ 31。如果设定其他的值，就不正常运行。

如果执行 RTC 软件复位，RDAYAR 寄存器就变为“00h”。

25.2.14 月闹钟寄存器 (RMONAR)

地址 0008 C41Ah



复位后的值

x: 不定值

位	符号	位名	功能	R/W
b3-b0	MON1[3:0]	1 月位	月的个位设定值	R/W
b4	MON10	10 月位	月的十位设定值	R/W
b6-b5	—	保留位	必须置“0”，读取值为“0”。	R/W
b7	ENB	ENB 位	如果为“1”，就和 RMONCNT 计数器的值进行比较。	R/W

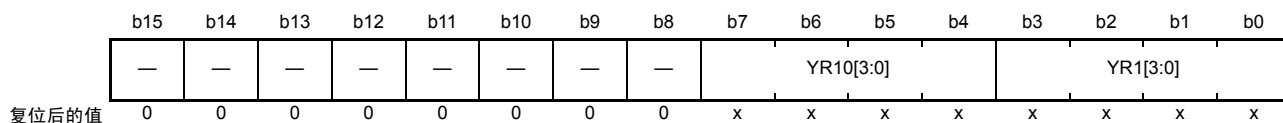
RMONAR 寄存器是与月部分 (BCD 码) 的计数器 (RMONCNT) 对应的闹钟寄存器。如果 ENB 位为“1”，就将 RMONAR 寄存器的值和 RMONCNT 计数器的值进行比较。只对 ENB 位为“1”的闹钟寄存器 (RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RYRAREN) 进行计数器和闹钟寄存器的比较，当各自的值全部相同时，ICU 的 IR92.IR 标志变为“1”。

月的可设定范围是 10 进制 (BCD) 的 01 ~ 12。如果设定其他的值，就不正常运行。

如果执行 RTC 软件复位，RMONAR 寄存器就变为“00h”。

25.2.15 年闹钟寄存器 (RYRAR)

地址 0008 C41Ch



复位后的值

x: 不定值

位	符号	位名	功能	R/W
b3-b0	YR1[3:0]	1 年位	年的个位设定值	R/W
b7-b4	YR10[3:0]	10 年位	年的十位设定值	R/W
b15-b8	—	保留位	读写值都为“0”。	R/W

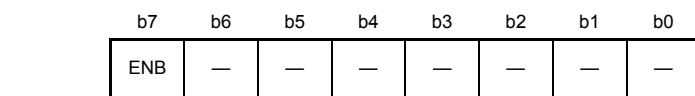
RYRAR 寄存器是与年部分 (BCD 码) 的计数器 (RYRCNT) 对应的闹钟寄存器。

年的可设定范围是 10 进制 (BCD) 的 00 ~ 99。如果设定其他的值, 就不正常运行。

如果执行 RTC 软件复位, RYRAR 寄存器就变为“0000h”。

25.2.16 年闹钟允许寄存器 (RYRAREN)

地址 0008 C41Eh



复位后的值

x: 不定值

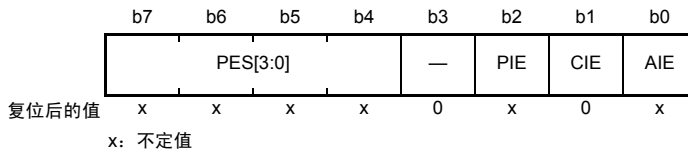
位	符号	位名	功能	R/W
b6-b0	—	保留位	必须置“0”, 读取值为“0”。	R/W
b7	ENB	ENB 位	如果为“1”, 就和 RYRCNT 计数器的值进行比较。	R/W

如果 RYRAREN 寄存器的 ENB 位为“1”, 就将 RYRAR 寄存器的值和 RYRCNT 计数器的值进行比较。只对 ENB 位为“1”的闹钟寄存器 (RSECAR、RMINAR、RHRAR、RWKAR、RDAYAR、RMONAR、RYRAREN) 进行计数器和闹钟寄存器的比较, 当各自的值全部相同时, ICU 的 IR92.IR 标志变为“1”。

如果执行 RTC 软件复位, RYRAREN 寄存器就变为“00h”。

25.2.17 RTC 控制寄存器 1 (RCR1)

地址 0008 C422h



位	符号	位名	功能	R/W
b0	AIE	闹钟中断允许位	0: 不允许闹钟中断请求 1: 允许闹钟中断请求	R/W
b1	CIE	进位中断允许位	0: 不允许进位中断请求 1: 允许进位中断请求	R/W
b2	PIE	周期中断允许位	0: 不允许周期中断请求 1: 允许周期中断请求	R/W
b3	—	保留位	读写值都为“0”。	R/W
b7-b4	PES[3:0]	周期中断选择位	b7 b4 0 1 1 0: 将周期中断的发生周期设定为每 1/256 秒 0 1 1 1: 将周期中断的发生周期设定为每 1/128 秒 1 0 0 0: 将周期中断的发生周期设定为每 1/64 秒 1 0 0 1: 将周期中断的发生周期设定为每 1/32 秒 1 0 1 0: 将周期中断的发生周期设定为每 1/16 秒 1 0 1 1: 将周期中断的发生周期设定为每 1/8 秒 1 1 0 0: 将周期中断的发生周期设定为每 1/4 秒 1 1 0 1: 将周期中断的发生周期设定为每 1/2 秒 1 1 1 0: 将周期中断的发生周期设定为每 1 秒 1 1 1 1: 将周期中断的发生周期设定为每 2 秒 上述以外: 不发生周期中断	R/W

与计数源同步更新 AIE 位、PIE 位和 PES[3:0] 位。如果改写 RCR1 寄存器，就必须在确认全部位的值已被更新后执行下一个处理。

AIE 位 (闹钟中断允许位)

此位选择允许或者禁止闹钟中断请求。

如果在深度软件待机中计数器和闹钟时间相同，就从深度软件待机返回，与 AIE 位的设定无关。

CIE 位 (进位中断允许位)

此位选择允许或者禁止在向秒计数器 (RSECCNT) 进位或者在读 64Hz 计数器 (R64CNT) 时向 64Hz 计数器进位的情况下产生的中断请求。

PIE 位 (周期中断允许位)

此位选择允许或者禁止周期中断请求。

如果在深度软件待机时和 PES[3:0] 位所选周期相同，就从深度软件待机返回，与 PIE 位的设定无关。

PES[3:0] 位 (周期中断选择位)

这些位设定周期中断的周期。根据 PES[3:0] 位设定的周期，定期产生周期中断 (PRD) 源。

25.2.18 RTC 控制寄存器 2 (RCR2)

地址 0008 C424h

b7	b6	b5	b4	b3	b2	b1	b0
—	HR24	AADJP	AADJE	RTCOE	ADJ30	RESET	START
x	x	x	x	0	0	0	x

复位后的值
x: 不定值

位	符号	位名	功能	R/W
b0	START	开始位	0: 年、月、星期、日、小时、分钟、秒、64Hz 计数器和预分频器停止运行。 1: 年、月、星期、日、小时、分钟、秒、64Hz 计数器和预分频器正常运行。	R/W
b1	RESET	RTC 软件复位的位	• 写时 0: 写操作无效 1: 对预分频器和 RTC 软件复位对象寄存器进行复位 • 读时 0: 时钟正常运行或者 RTC 软件复位结束 1: 正在进行 RTC 软件复位	R/W
b2	ADJ30	30 秒调整位	• 写时 0: 写操作无效 1: 进行 30 秒调整 • 读时 0: 时钟正常运行或者 30 秒调整结束 1: 正在进行 30 秒调整	R/W
b3	RTCOE	RTCOOUT 输出控制位	0: 不从引脚输出 RTCOUT 1: 从引脚输出 RTCOUT	R/W
b4	AADJE	自动校正功能允许位	0: 禁止自动校正功能 1: 允许自动校正功能	R/W
b5	AADJP	自动校正周期选择位	0: 每分钟从预分频器的计数值加减 RADJ.ADJ[5:0] 位的值 1: 每 10 秒从预分频器的计数值加减 RADJ.ADJ[5:0] 位的值	R/W
b6	HR24	时间模式位	0: RTC 以 12 小时模式运行 1: RTC 以 24 小时模式运行	R/W
b7	—	保留位	必须置“0”，读取值为“0”。	R/W

START 位 (开始位)

此位控制预分频器和计数器 (时钟) 的停止或者运行。

与计数源同步更新 START 位。如果改写 START 位，就必须在确认此值已被更新后执行下一个处理。

RESET 位 (RTC 软件复位的位)

此位对预分频器和 RTC 软件复位对象寄存器进行初始化。

如果给 RESET 位写“1”，就与计数源同步进行初始化，一旦初始化结束，RESET 位自动变为“0”。

如果 RESET 位已被写“1”，就必须在确认此位已变为“0”后执行下一个处理。

ADJ30 位 (30 秒调整位)

这是进行 30 秒调整的位。

在给 ADJ30 位写“1”的情况下，如果 RSECCNT 计数器的值小于 30 秒，就舍去秒数据，为 00 秒；如果大于等于 30 秒，就向 1 分钟进位。

与计数源同步进行 30 秒调整。在给 ADJ30 位写“1”的情况下，一旦 30 秒调整结束，ADJ30 位自动变为“0”，必须在确认 ADJ30 位已变为“0”后执行下一个处理。

如果进行 30 秒调整，也对预分频器和 R64CNT 计数器进行复位。

如果执行 RTC 软件复位，ADJ30 位就变为“0”。

RTCOE 位 (RTCOUT 输出控制位)

此位控制 RTCOUT (1Hz 时钟) 的输出。

必须在通过 START 位停止计数后改写 RTCOE 位。不能在停止计数 (给 START 位写“0”) 的同时改写 RTCOE 位的值。

要将 1Hz 时钟输出到外部引脚时，必须将 RTCOE 位和端口控制都置为有效。

AADJE 位 (自动校正功能允许位)

此位控制禁止或者允许自动校正功能。

必须在将加减位 (RADJ.PMADJ[1:0]) 置“00h” (不校正) 后改写 AADJE 位。

如果执行 RTC 软件复位，AADJE 位就变为“0”。

AADJP 位 (自动校正周期选择位)

此位选择自动校正周期。

必须在将加减位 (RADJ.PMADJ[1:0]) 置“00h” (不校正) 后改写 AADJP 位。

如果执行 RTC 软件复位，AADJP 位就变为“0”。

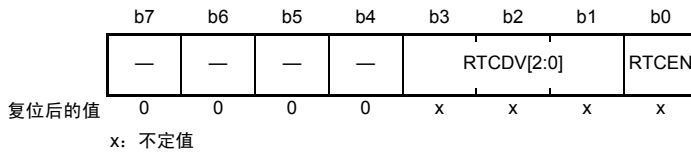
HR24 位 (时间模式位)

此位指定 RTC 运行模式是 12 小时模式还是 24 小时模式。

必须在通过 START 位停止计数后改写 HR24 位，而且不能和 START 位同时改写 HR24 位的值。

25.2.19 RTC 控制寄存器 3 (RCR3)

地址 0008 C426h



位	符号	位名	功能	R/W
b0	RTCEN	副时钟振荡器控制位	0: 副时钟振荡器停止振荡 1: 副时钟振荡器振荡	R/W
b3-b1	RTCDV[2:0]	副时钟振荡器的驱动能力控制位	b3 b1 0 0 0: 不能设定 0 0 1: 用于低 CL 的驱动能力 0 1 0: 不能设定 0 1 1: 不能设定 1 0 0: 不能设定 1 0 1: 不能设定 1 1 0: 用于标准 CL 的驱动能力 1 1 1: 不能设定	R/W
b7-b4	—	保留位	读写值都为“0”。	R/W

RTCEN 位 (副时钟振荡器控制位)

通过 RTCEN 位和时钟发生电路的寄存器控制副时钟振荡器的运行或者停止。如果任意一个位被设定为运行，副时钟振荡器就变为运行状态。

如果要将副时钟用作 RTC 的计数源，就必须通过 RTCEN 位进行副时钟振荡器的运行设定。

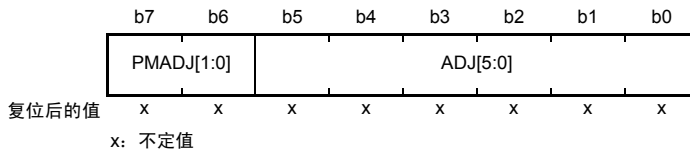
RTCDV[2:0] 位 (副时钟振荡器的驱动能力设定位)

这些位控制副时钟振荡器的驱动能力。

必须注意：如果在连接内部调试仿真器的状态下将 RTCDV[2:0] 位设定为“001b”（用于低 CL 的驱动能力），有可能对副时钟振荡器的振荡精度造成影响。如果将 RCR3.RTCDV[2:0] 位设定为“110b”（用于标准 CL 的驱动能力），就不会对振荡精度造成影响。

25.2.20 时间误差校正寄存器 (RADJ)

地址 0008 C42Eh



位	符号	位名	功能	R/W
b5-b0	ADJ[5:0]	校正值位	设定误差校正值。	R/W
b7-b6	PMADJ[1:0]	加减位	b7 b6 0 0: 不进行校正 0 1: 调快时钟 1 0: 调慢时钟 1 1: 不能设定	R/W

根据误差校正值将时钟调快或者调慢来进行校正。

当自动校正功能允许位 (RCR2.AADJE) 为“0”时，在写 RADJ 寄存器时进行校正。

当 RCR2.AADJE 位为“1”时，通过自动校正周期选择位 (RCR2.AADJP) 设定的间隔进行校正。

在通过软件进行校正时，如果在设定寄存器后的 320 个计数源周期以内设定下一个校正值，上次的校正设定就可能无效。因此，要连续进行校正时，必须在设定寄存器并且等待 320 个计数源周期后再次进行设定。

与计数源同步更新 RADJ 寄存器。如果改写 RADJ 寄存器，就必须在确认全部位的值已被更新后执行下一个处理。

如果执行 RTC 软件复位，RADJ 就变为“00h”。

ADJ[5:0] 位 (校正值位)

根据时钟误差设定校正值 (副时钟的时钟周期数)。

PMADJ[1:0] 位 (加减位)

根据 ADJ[5:0] 位设定的误差校正值选择调快或者调慢时钟。

25.2.21 时间捕捉控制寄存器 y (RTCCRy) (y=0 ~ 2)

地址 RTCCR0 0008 C440h、RTCCR1 0008 C442h、RTCCR2 0008 C444h

	b7	b6	b5	b4	b3	b2	b1	b0
	TCEN	—	TCNF[1:0]	—	TCST	TCCT[1:0]		
复位后的值	x	0	x	x	0	x	x	x

x: 不定值

位	符号	位名	功能	R/W
b1-b0	TCCT[1:0]	时间捕捉控制位	b1 b0 0 0: 不进行事件检测 0 1: 上升沿检测 1 0: 下降沿检测 1 1: 双边沿检测	R/W
b2	TCST	时间捕捉状态位	表示检测到事件。 能通过写“0”将此位置“0”。 0: 未检测到事件 1: 检测到事件 (注1)	R/W
b3	—	保留位	读写值都为“0”。	R/W
b5-b4	TCNF[1:0]	时间捕捉噪声滤波器控制位	b5 b4 0 0: 噪声滤波器 OFF 0 1: 不能设定 1 0: 噪声滤波器 ON (计数源) 1 1: 噪声滤波器 ON (计数源的 32 分频)	R/W
b6	—	保留位	读写值都为“0”。	R/W
b7	TCEN	时间捕捉事件输入引脚允许位	0: RTCICn 引脚作为时间捕捉事件输入引脚无效 1: RTCICn 引脚作为时间捕捉事件输入引脚有效 (n=0 ~ 2)	R/W

注 1. 写“1”无效。

RTCCR0 寄存器控制 RTCIC0 引脚, RTCCR1 寄存器控制 RTCIC1 引脚, RTCCR2 寄存器控制 RTCIC2 引脚。

与计数源同步更新 RTCCRy 寄存器。如果改写 RTCCRy 寄存器, 就必须在确认除 TCST 位以外的全部位的值已被更新后执行下一个处理。

如果执行 RTC 软件复位, RTCCRy 就变为“00h”。

TCCT[1:0] 位 (时间捕捉控制位)

这些位控制时间捕捉事件输入引脚 (RTCIC0、RTCIC1、RTCIC2) 的边沿检测。

可选择要检测的边沿。必须在 TCEN 位为“1”的状态下设定 TCCT[1:0] 位。

TCST 位 (时间捕捉状态位)

此位表示检测到时间捕捉事件输入引脚 (RTCIC0、RTCIC1、RTCIC2) 的事件。

当 TCST 位为“0”时, 表示未检测到事件。

当 TCST 位为“1”时, 表示检测到对应引脚的事件, 并且对应的捕捉寄存器有效。如果多次检测到事件, 就保持第一次的捕捉时间。

如果在停止计数时 (RCR2.START 位为“0”) 检测到事件, 就不能保证捕捉到的值。为了放弃捕捉到的值, 必须在将 TCST 位置“0”后再使用。

能通过给 TCST 位写“0”, 将 TCST 位置“0”。如果写“0”以外的值, 写操作就无效。

必须在将 TCCT[1:0] 位置“00b” (不进行事件检测) 的状态下将 TCST 位置“0”。

TCST 位与计数源同步变为“0”。如果将 TCST 位置“0”, 就必须在确认此值已被更新后执行下一个处理。

TCNF[1:0] 位 (时间捕捉噪声滤波器控制位)

这些位控制时间捕捉事件输入引脚 (RTCIC0、RTCIC1、RTCIC2) 的噪声滤波器。

当噪声滤波器为 ON 时, 可选择计数源的 1 分频或者 32 分频。此时, 如果时间捕捉事件输入引脚的输入电平在选择采样间隔中 3 次相同, 就决定输入电平。

必须在将 TCCT[1:0] 位置“00b” (不进行事件检测) 的状态下设定 TCNF[1:0] 位。在使用噪声滤波器时, 必须在设定 TCNF[1:0] 位并且等待 3 个所设采样周期后设定 TCCT[1:0] 位。必须在 TCEN 位为“1”的状态下设定 TCNF[1:0] 位。

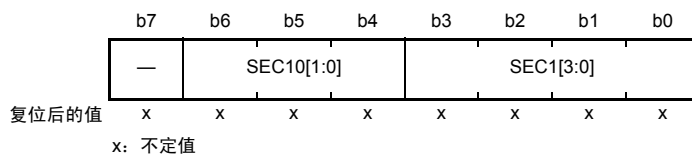
TCEN 位 (时间捕捉事件输入引脚允许位)

此位控制时间捕捉事件输入引脚 (RTCIC0、RTCIC1、RTCIC2) 的有效或者无效。

在多路复用时间捕捉事件输入引脚 (RTCIC0、RTCIC1、RTCIC2) 时, 必须将端口控制和 TCEN 位都置为有效。此时, 必须先设定端口控制。要将 TCEN 位置“0”时, 也必须将 TCCT[1:0] 位置“00h”。

25.2.22 秒捕捉寄存器 y (RSECCPy) (y=0 ~ 2)

地址 RSECCP0 0008 C452h、RSECCP1 0008 C462h、RSECCP2 0008 C472h



位	符号	位名	功能	R/W
b3-b0	SEC1[3:0]	秒的个位捕捉位	表示秒的个位捕捉值。	R
b6-b4	SEC10[1:0]	秒的十位捕捉位	表示秒的十位捕捉值。	R
b7	—	保留位	必须置“0”, 读取值为“0”。	R/W

RSECCPy 寄存器是在检测到时间捕捉事件时捕捉 RSECCNT 计数器值的只读寄存器。

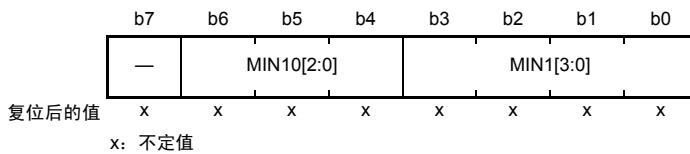
当通过 RTCIC0 引脚检测到事件时, 将事件检测时间保存到 RSECCP0 寄存器; 当通过 RTCIC1 引脚检测到事件时, 将事件检测时间保存到 RSECCP1 寄存器; 当通过 RTCIC2 引脚检测到事件时, 将事件检测时间保存到 RSECCP2 寄存器。

如果执行 RTC 软件复位, RSECCPy 就变为“00h”。

必须在通过 RTCCRy.TCCT[1:0] 位停止时间捕捉事件的检测后读此寄存器。

25.2.23 分钟捕捉寄存器 y (RMINCPy) (y=0 ~ 2)

地址 RMINCP0 0008 C454h、RMINCP1 0008 C464h、RMINCP2 0008 C474h



位	符号	位名	功能	R/W
b3-b0	MIN1[3:0]	分钟个位捕捉位	表示分钟个位捕捉值。	R
b6-b4	MIN10[2:0]	分钟十位捕捉位	表示分钟十位捕捉值。	R
b7	—	保留位	必须置“0”，读取值为“0”。	R/W

RMINCPy 寄存器是在检测到时间捕捉事件时捕捉 RMINCNT 计数器值的只读寄存器。

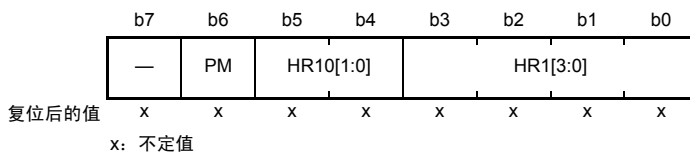
当通过 RTCIC0 引脚检测到事件时，将事件检测时间保存到 RMINCP0 寄存器；当通过 RTCIC1 引脚检测到事件时，将事件检测时间保存到 RMINCP1 寄存器；当通过 RTCIC2 引脚检测到事件时，将事件检测时间保存到 RMINCP2 寄存器。

如果执行 RTC 软件复位，RMINCPy 就变为“00h”。

必须在通过 RTCCRy.TCCT[1:0] 位停止时间捕捉事件的检测后读此寄存器。

25.2.24 小时捕捉寄存器 y (RHRCPy) (y=0 ~ 2)

地址 RHRCP0 0008 C456h、RHRCP1 0008 C466h、RHRCP2 0008 C476h



位	符号	位名	功能	R/W
b3-b0	HR1[3:0]	小时的个位捕捉位	表示小时的个位捕捉值。	R
b5-b4	HR10[1:0]	小时的十位捕捉位	表示小时的十位捕捉值。	R
b6	PM	PM 位	0: 上午 1: 下午	R
b7	—	保留位	必须置“0”，读取值为“0”。	R/W

RHRCPy 寄存器是在检测到时间捕捉事件时捕捉 RHRCNT 计数器值的只读寄存器。

当通过 RTCIC0 引脚检测到事件时，将事件检测时间保存到 RHRCP0 寄存器；当通过 RTCIC1 引脚检测到事件时，将事件检测时间保存到 RHRCP1 寄存器；当通过 RTCIC2 引脚检测到事件时，将事件检测时间保存到 RHRCP2 寄存器。

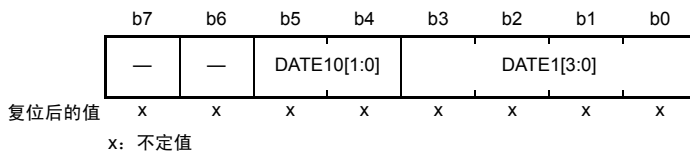
PM 位只在 RCR2.HR24 位为“0”（以 12 小时模式运行）时有效。

如果执行 RTC 软件复位，RHRCPy 就变为“00h”。

必须在通过 RTCCRy.TCCT[1:0] 位停止时间捕捉事件的检测后读此寄存器。

25.2.25 日捕捉寄存器 y (RDAYCPy) (y=0 ~ 2)

地址 RDAYCP0 0008 C45Ah、RDAYCP1 0008 C46Ah、RDAYCP2 0008 C47Ah



位	符号	位名	功能	R/W
b3-b0	DATE1[3:0]	日的个位捕捉位	表示日的个位捕捉值。	R
b5-b4	DATE10[1:0]	日的十位捕捉位	表示日的十位捕捉值。	R
b7-b6	—	保留位	必须置“0”，读取值为“0”。	R/W

RDAYCPy 寄存器是在检测到时间捕捉事件时捕捉 RDAYCNT 计数器值的只读寄存器。

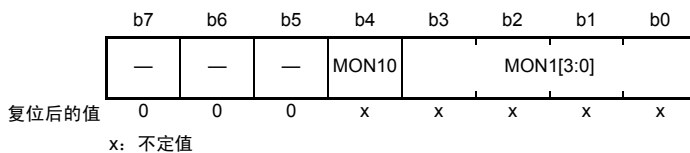
当通过 RTCIC0 引脚检测到事件时，将事件检测时间保存到 RDAYCP0 寄存器；当通过 RTCIC1 引脚检测到事件时，事件检测时间保存到 RDAYCP1 寄存器；当通过 RTCIC2 引脚检测到事件时，将事件检测时间保存到 RDAYCP2 寄存器。

如果执行 RTC 软件复位，RDAYCPy 就变为“00h”。

必须在通过 RTCCRy.TCCT[1:0] 位停止时间捕捉事件的检测后读此寄存器。

25.2.26 月捕捉寄存器 y (RMONCPy) (y=0 ~ 2)

地址 RMONCP0 0008 C45Ch、RMONCP1 0008 C46Ch、RMONCP2 0008 C47Ch



位	符号	位名	功能	R/W
b3-b0	MON1[3:0]	月的个位捕捉位	表示月的个位捕捉值。	R
b4	MON10	月的十位捕捉位	表示月的十位捕捉值。	R
b7-b5	—	保留位	读取值为“0”，写操作无效。	R

RMONCPy 寄存器是在检测到时间捕捉事件时捕捉 RMONCNT 计数器值的只读寄存器。

当通过 RTCIC0 引脚检测到事件时，将事件检测时间保存到 RMONCP0 寄存器；当通过 RTCIC1 引脚检测到事件时，将事件检测时间保存到 RMONCP1 寄存器；当通过 RTCIC2 引脚检测到事件时，将事件检测时间保存到 RMONCP2 寄存器。

如果执行 RTC 软件复位，RMONCPy 就变为“00h”。

必须在通过 RTCCRy.TCCT[1:0] 位停止时间捕捉事件的检测后读此寄存器。

25.3 运行说明

25.3.1 接通电源后的寄存器初始设定概要

必须在接通电源后进行时钟、时间、时钟误差校正、闹钟、中断和时间捕捉控制寄存器的初始设定。

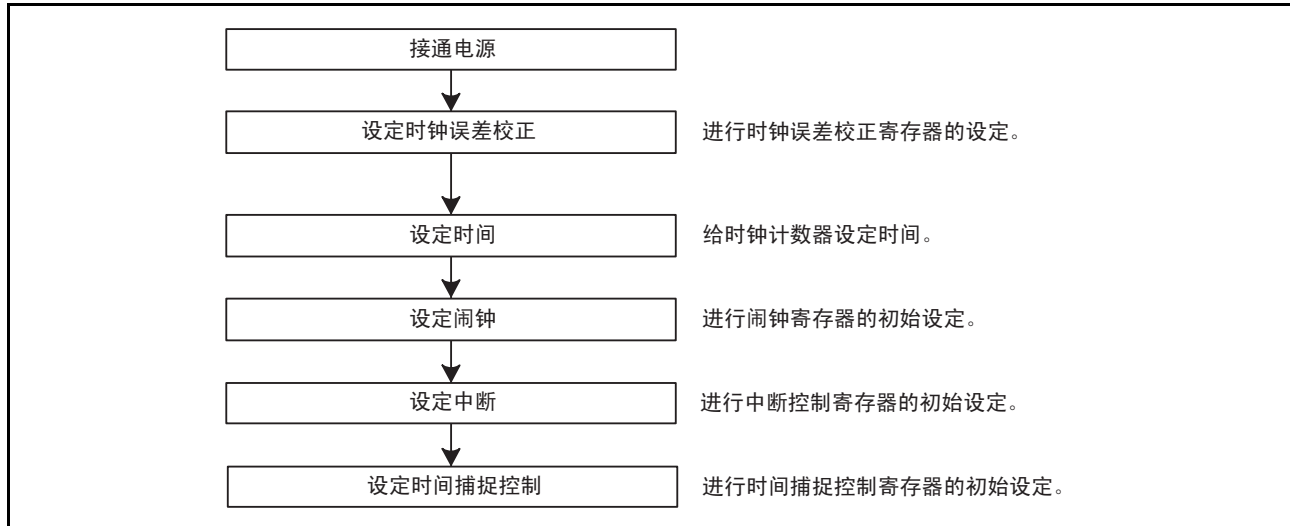


图 25.2 接通电源后的初始设定概要

25.3.2 时钟设定步骤

时钟设定步骤如图 25.3 所示。

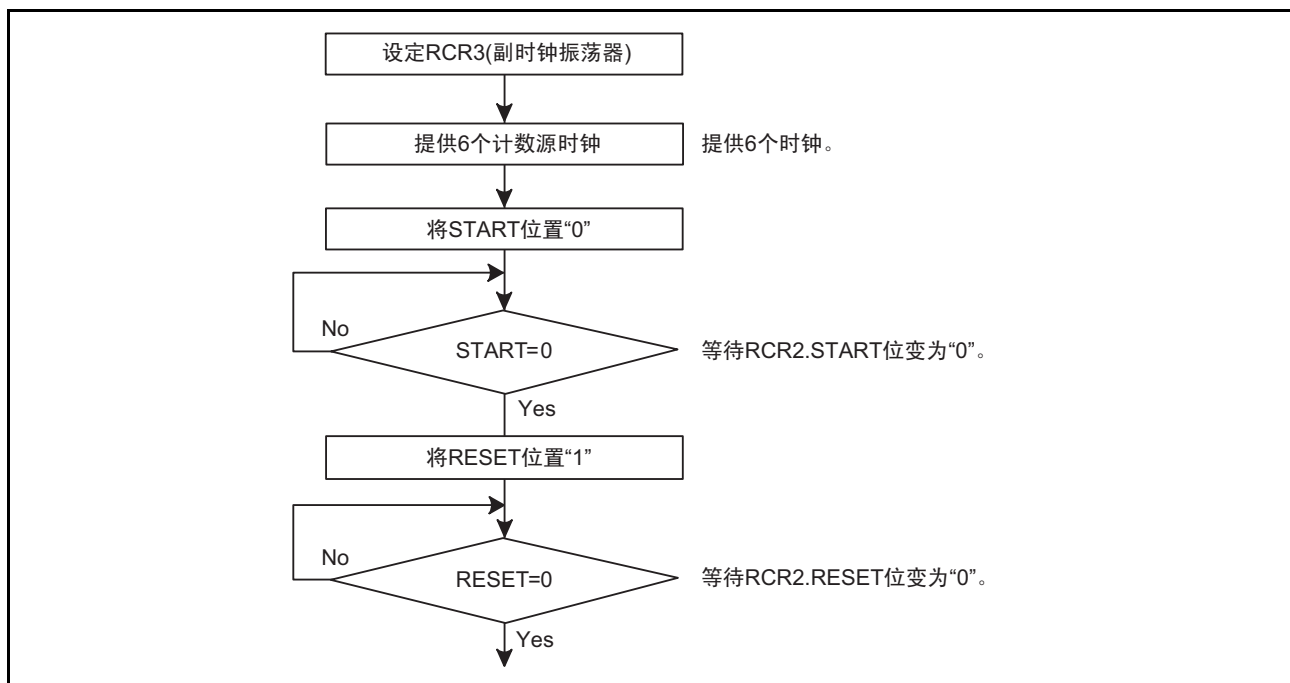


图 25.3 时钟设定步骤

25.3.3 时间设定步骤

时间设定步骤如图 25.4 所示。

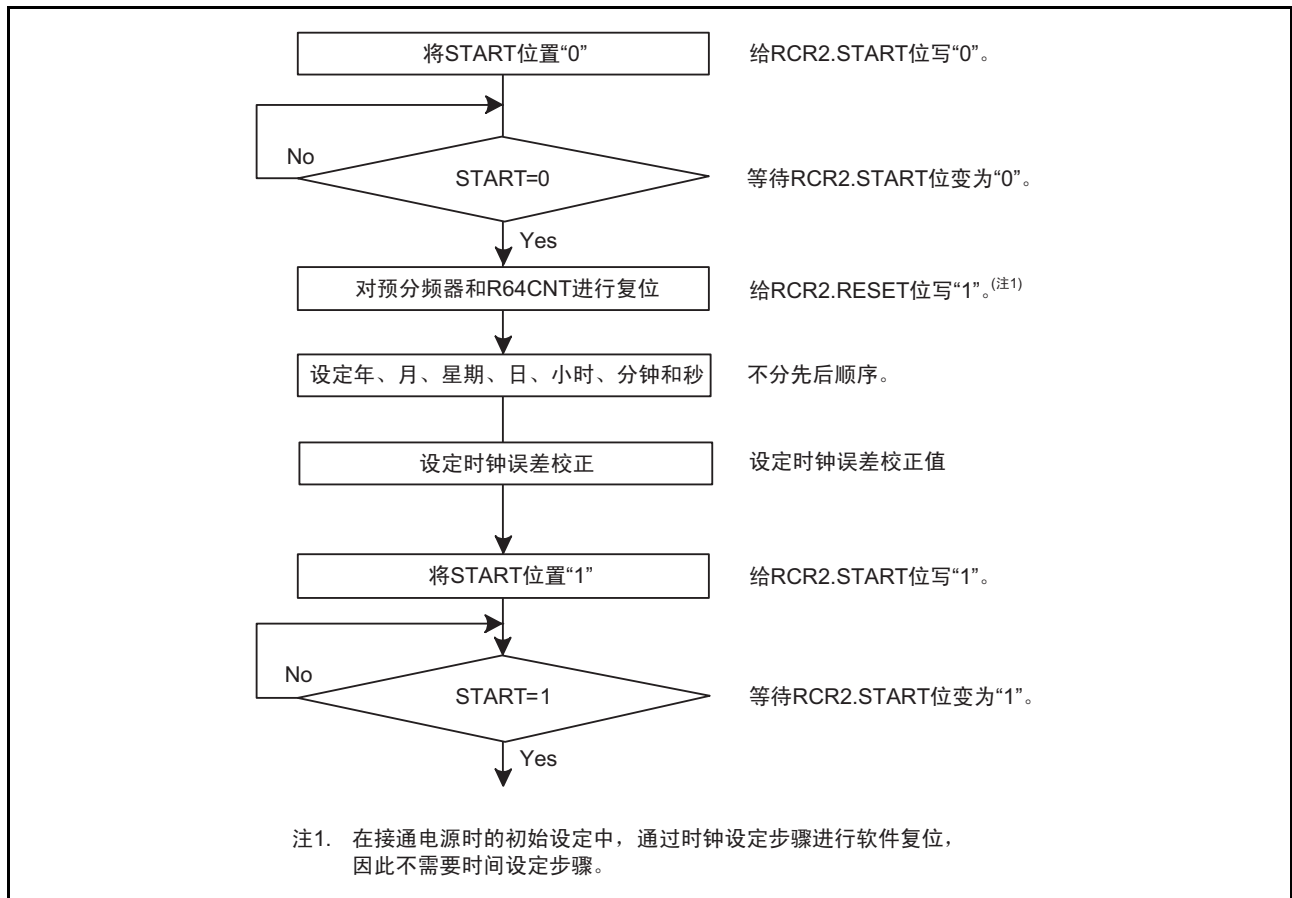


图 25.4 时间设定步骤

25.3.4 30 秒调整步骤

30 秒调整步骤如图 25.5 所示。

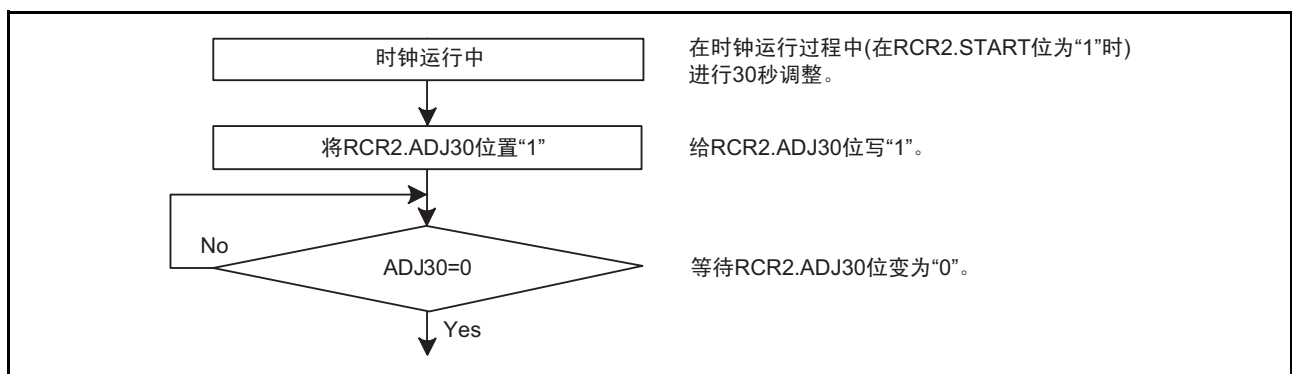


图 25.5 30 秒调整步骤

25.3.5 64Hz 计数器和时间的读取步骤

64Hz 计数器和时间的读取步骤如图 25.6 所示。

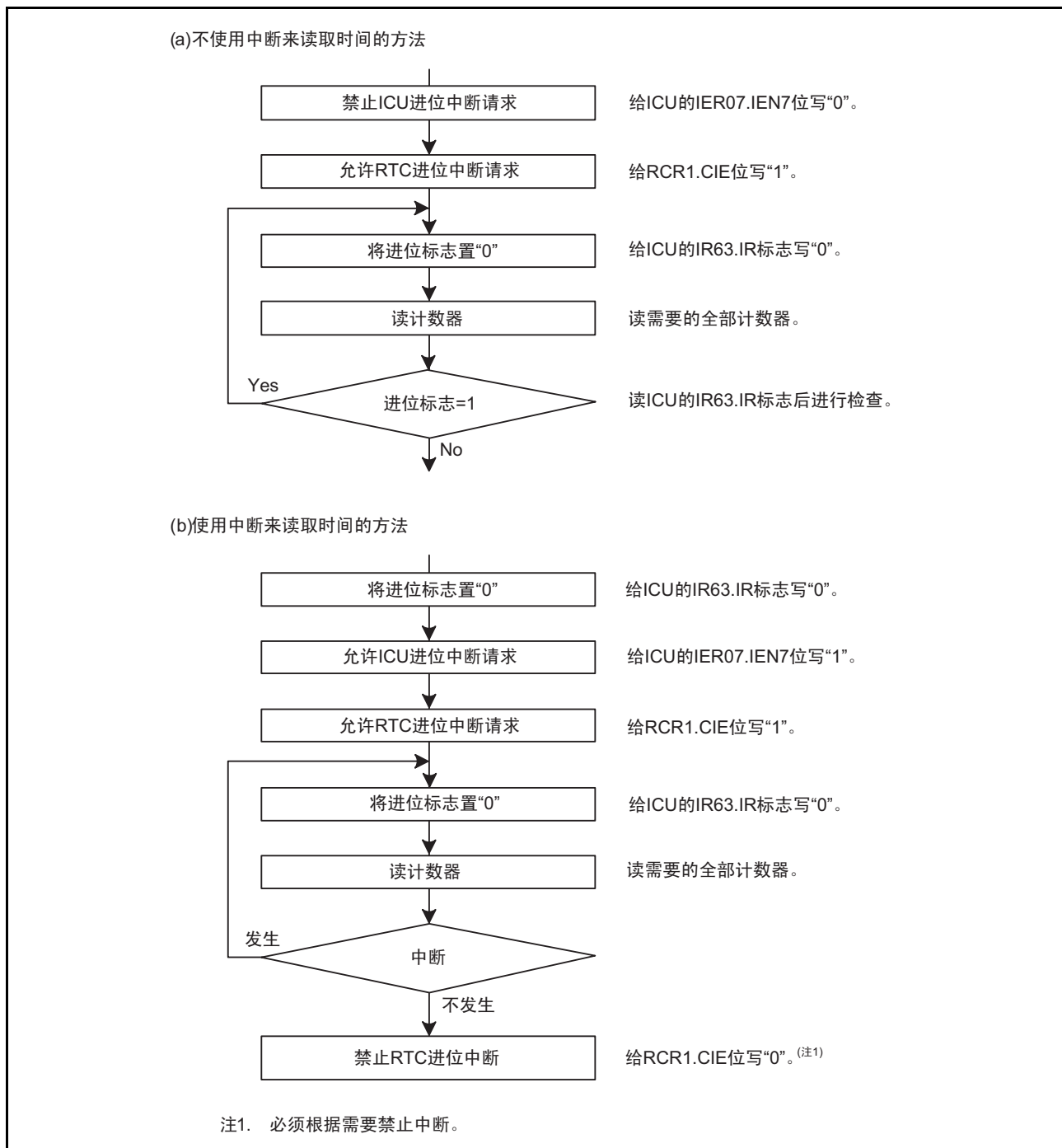
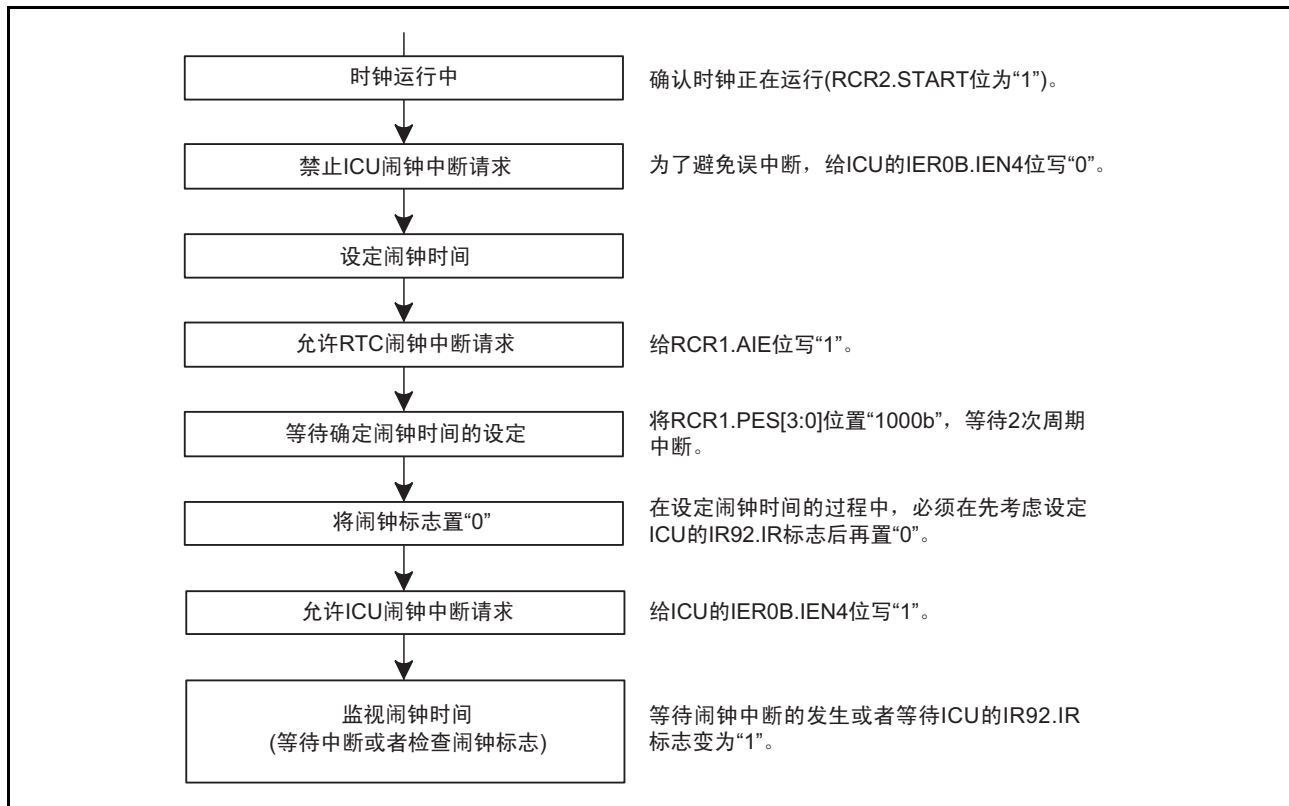


图 25.6 时间读取步骤

如果在读 64Hz 计数器和时间的过程中发生进位，就读不到正确的时间，因此有可能需要重新读。不使用中断的方法以及使用进位中断的方法分别如图 25.6 的 (a) 和图 25.6 的 (b) 所示。为了便于编程，通常利用不使用中断的方法。

25.3.6 闹钟功能

闹钟功能的使用方法如图 25.7 所示。



能通过年、月、日、星期、小时、分钟和秒中的任意一个或者组合产生闹钟。给作为闹钟对象的各闹钟寄存器的 ENB 位写“1”，将闹钟时间设定到低位；给不作为闹钟对象的寄存器的 ENB 位写“0”。

如果计数器和闹钟时间相同，ICU 的 IR92.IR 标志就变为“1”。能通过读此标志来确认闹钟的检测，但是通常通过中断进行确认。能在 ICU 的 IER0B.IEN4 位已被写“1”时产生闹钟中断并且检测到闹钟。

如果给 ICU 的 IR92.IR 标志写“0”，此标志就变为“0”。

在低功耗状态下，如果计数器和闹钟时间相同，就从低功耗状态返回。

25.3.7 闹钟中断的禁止步骤

允许状态的闹钟中断请求的禁止步骤如图 25.8 所示。

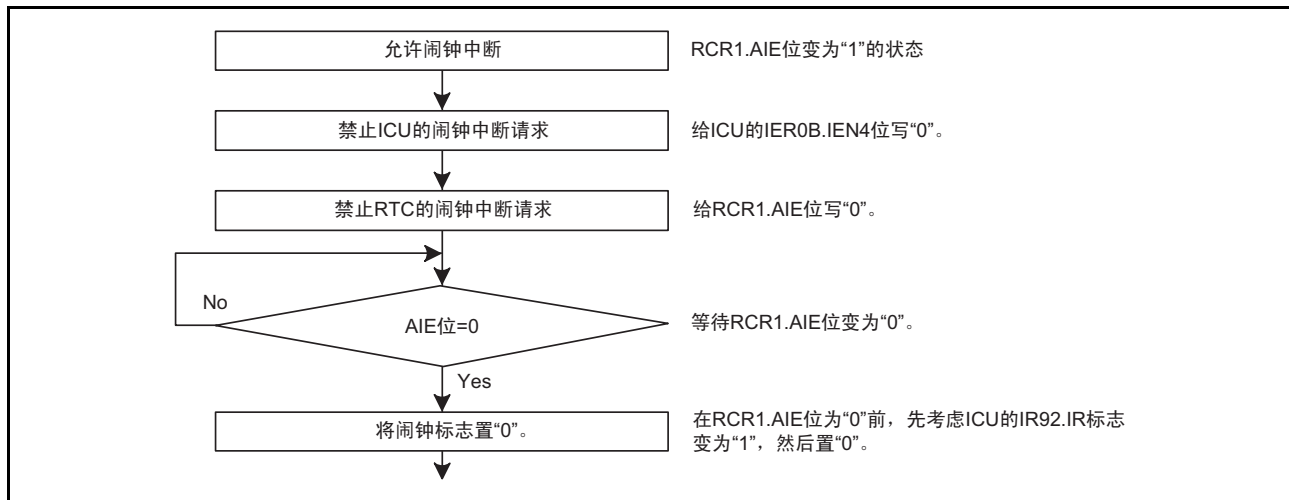


图 25.8 闹钟中断请求的禁止步骤

25.3.8 时钟误差校正功能

时钟误差校正功能校正因副时钟的振荡精度引起的时钟误差（慢 / 快）。因为以 32768 个时钟周期为 1 秒运行，所以副时钟频率高时钟就变快，副时钟频率低时钟就变慢。此功能通过调快或者调慢时钟来校正误差。

时钟误差校正功能有自动校正和软件校正两种。

必须通过 RCR2.AADJE 位选择自动校正或者软件校正。

25.3.8.1 自动校正功能

当 RCR2.AADJE 位为“1”时，自动校正功能有效。

自动校正功能在 RCR2.AADJP 位选择的每个校正周期，根据 RADJ 寄存器的设定将时钟调快或者调慢。例子如下所示：

例 1) 副时钟 = 32.769kHz

校正方法：

当副时钟频率为 32.769kHz 时，以 32769 个时钟周期为 1 秒。因为 RTC 以 32768 个时钟周期为 1 秒运行，所以时钟每秒快 1 个时钟周期。如果是 1 分钟，时钟就快 60 个时钟周期，因此能通过将时钟每分钟调慢 60 个时钟周期来进行校正。

寄存器的设定内容：

- RCR2.AADJP 位 = 0（按分钟进行校正）
- RADJ.PMADJ[1:0] 位 = 10b（调慢）
- RADJ.ADJ[5:0] 位 = 60（3Ch）

例 2) 副时钟 =32.766kHz

校正方法:

当副时钟频率为 32.766kHz 时, 以 32766 个时钟周期为 1 秒, 因为 RTC 以 32768 个时钟周期为 1 秒运行, 所以时钟每秒慢 2 个时钟周期。如果是 10 秒, 时钟就慢 20 个时钟周期, 因此能通过将时钟每 10 秒调快 20 个时钟周期来进行校正。

寄存器的设定内容:

- RCR2.AADJP 位=1 (每 10 秒校正)
- RADJ.PMADJ[1:0] 位=01b (调快)
- RADJ.ADJ[5:0] 位=20 (14h)

25.3.8.2 通过软件进行的校正

当 RCR2.AADJE 位为“0”时, 软件校正有效。

对于软件校正, 在执行写 RADJ 寄存器的指令时, 根据 RADJ 寄存器的设定将时钟调快或者调慢。

例 1) 副时钟 =32.769kHz

校正方法:

当副时钟频率为 32.769kHz 时, 以 32769 个时钟周期为 1 秒, 因为 RTC 以 32768 个时钟周期为 1 秒运行, 所以时钟每秒快 1 个时钟周期。因为时钟每秒快 1 个时钟周期, 所以能通过将时钟每秒调慢 1 个时钟周期来进行校正。

寄存器的设定内容:

- RADJ.PMADJ[1:0] 位=10b (调慢)
 - RADJ.ADJ[5:0] 位=1 (01h)
- 在发生每秒的中断时写 RADJ 寄存器。

25.3.8.3 校正模式的更改步骤

要更改校正模式时, 必须在将 RADJ.PMADJ[1:0] 位置“00b” (不校正) 后更改 RCR2.AADJE 位。

从软件校正更改为自动校正的情况:

1. 将 RADJ.PMADJ[1:0] 位置“00b” (不校正)。
2. 将 RCR2.AADJE 位置“1” (允许自动校正功能)。
3. 通过 RCR2.AADJP 位选择校正周期。
4. 给 RADJ.PMADJ[1:0] 位设定加减算法并且通过 RADJ.ADJ[5:0] 位设定时钟误差校正值。

从自动校正更改为软件校正的情况:

1. 将 RADJ.PMADJ[1:0] 位置“00b” (不校正)。
2. 将 RCR2.AADJE 位置“0” (软件校正功能有效)。
3. 如果在任意时刻给 RADJ.PMADJ[1:0] 位设定加减算法并且通过 RADJ.ADJ[5:0] 位写时钟误差校正值, 就进行校正。此后, 在每次写 RADJ 寄存器时进行校正。

25.3.8.4 校正功能的停止步骤

要停止校正功能时, 必须将 RADJ.PMADJ[1:0] 位置“00b” (不校正)。

25.3.8.5 时钟捕捉功能

RTC 通过时间捕捉事件输入引脚的边沿检测来保存月、日、小时、分钟和秒的值。

能将噪声滤波器用于 RTC 的时间捕捉事件输入引脚。如果根据设定的采样周期采样到的引脚电平 3 次相同，噪声滤波器就将相同的电平传送到 RTC 内部，并且将 RTC 的内部电平维持到采样的引脚电平 3 次相同为止。

时间捕捉事件输入引脚能按各引脚设定噪声滤波器的 ON/OFF。

噪声滤波器 OFF 和 ON 时的运行如图 25.9 和图 25.10 所示。

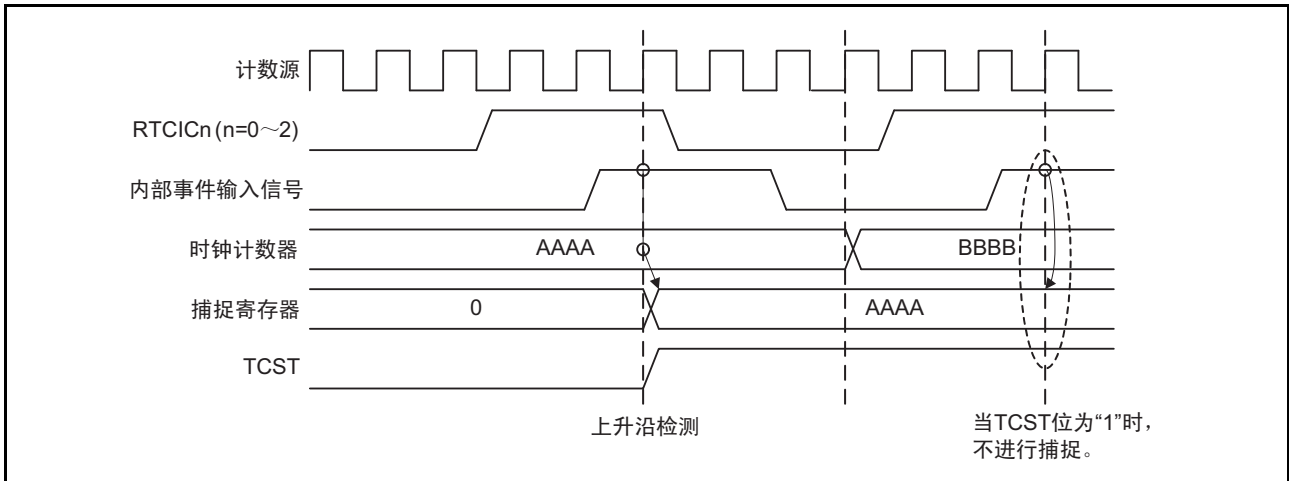


图 25.9 时间捕捉功能的运行时序 (滤波器 OFF)

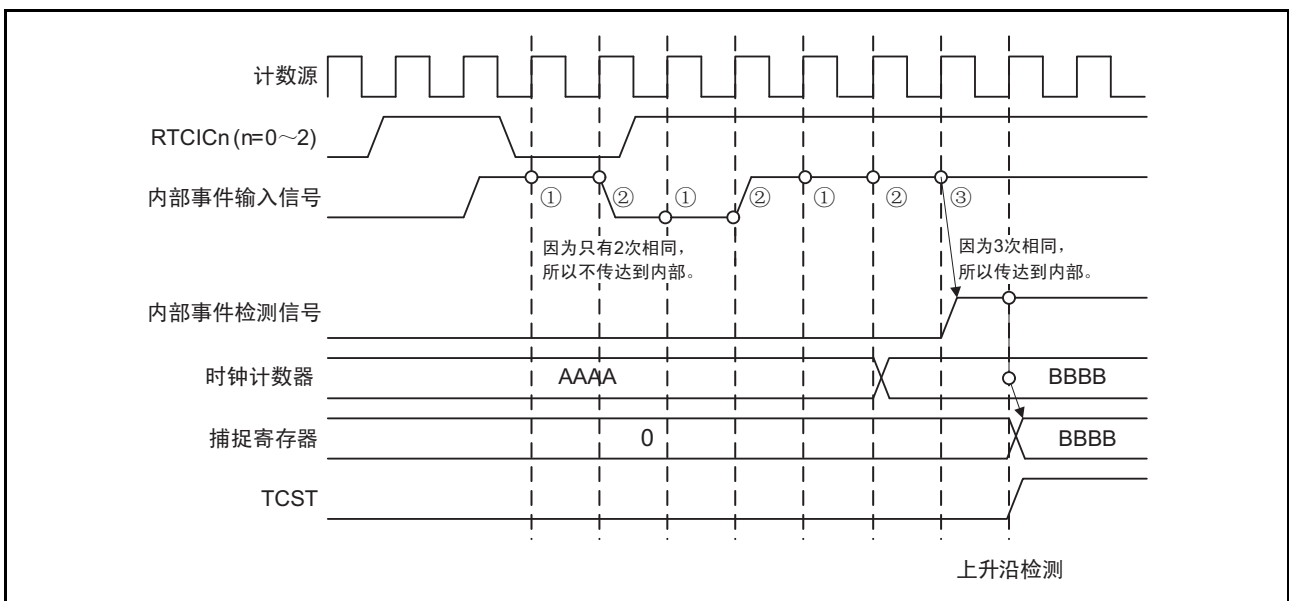


图 25.10 时间捕捉功能的运行时序 (滤波器 ON)

25.4 中断源

RTC 的中断源有以下 3 种，RTC 的中断源如表 25.3 所示。

表 25.3 RTC 的中断源

名称	中断源
ALM	闹钟中断
PRD	周期中断
CUP	进位中断

(1) 闹钟中断 (ALM)

根据闹钟寄存器和时钟计数器的比较结果产生中断（详细内容请参照各闹钟寄存器的说明）。

在设定闹钟寄存器的过程中，可能因闹钟寄存器和时钟计数器的值相同而使中断标志被置位，所以必须在更改闹钟寄存器后清除 1 次该中断的 IR92.IR 标志。一旦闹钟中断的中断标志被清“0”，就在闹钟寄存器和时钟计数器再次变为不同的状态后到再次相同或者重新设定闹钟为止，不将中断标志置位。

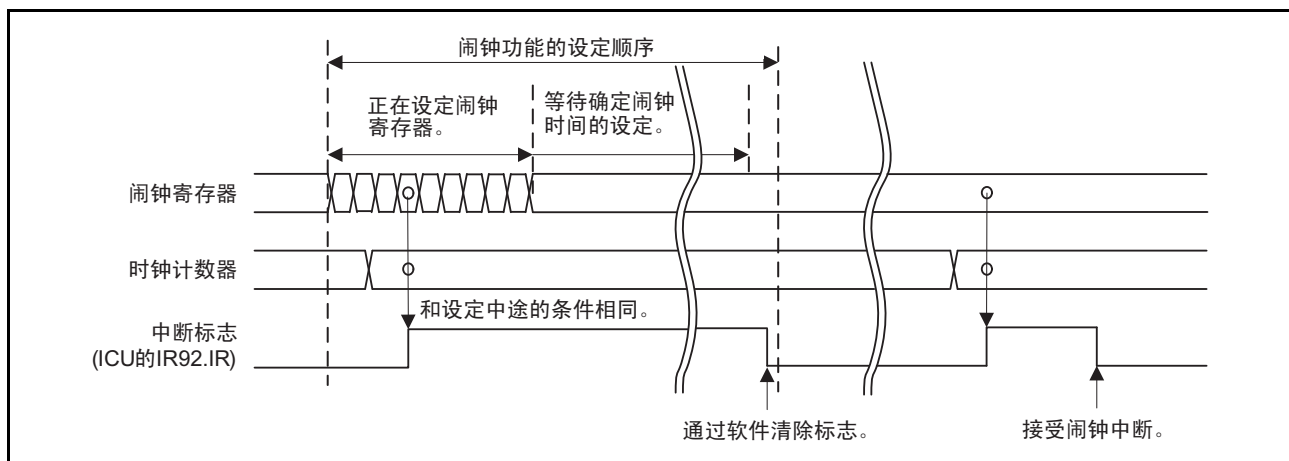


图 25.11 闹钟中断 (ALM) 的时序图

(2) 周期中断 (PRD)

这是在 2 秒、1 秒、1/2 秒、1/4 秒、1/8 秒、1/16 秒、1/32 秒、1/64 秒、1/128 秒、1/256 秒周期产生的中断，可通过 RCR1.PES[3:0] 位选择周期。

(3) 进位中断 (CUP)

这是在向秒计数器进位或者在读 64Hz 计数器时向 R64CNT 计数器进位的情况下产生的有效中断。

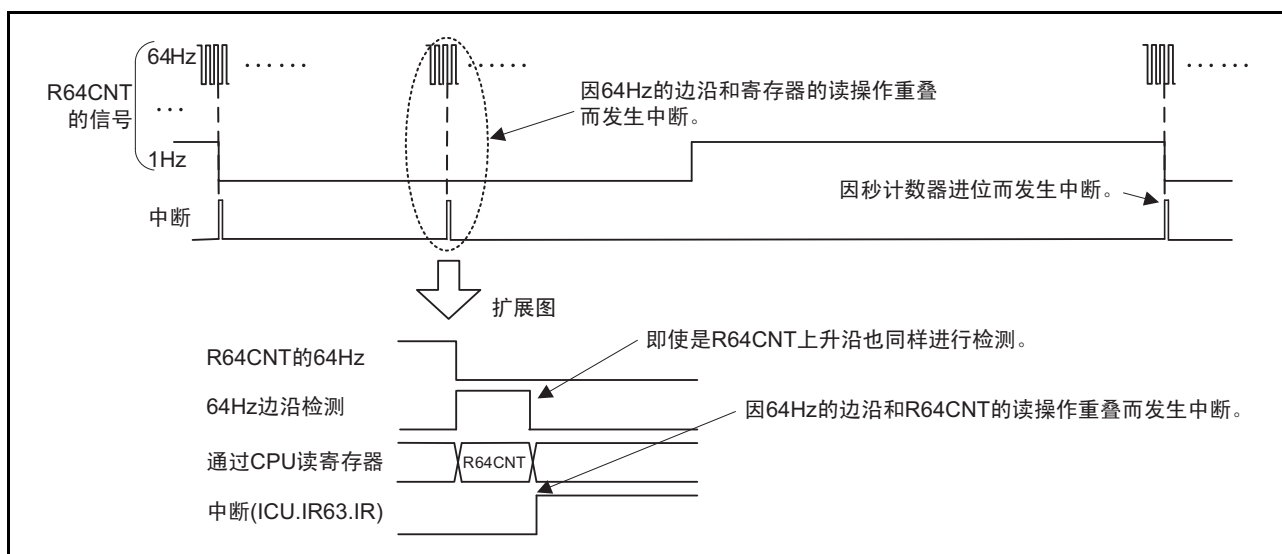


图 25.12 进位中断 (CUP) 的时序图

25.5 事件链接输出功能

RTC 向事件链接控制器 (ELC) 输出以下事件，使事先设定的模块运行。

(1) 周期事件的输出

通过设定 RCR1.PES[3:0] 位，能以从 2 秒、1 秒、1/2 秒、1/4 秒、1/8 秒、1/16 秒、1/32 秒、1/64 秒、1/128 秒、1/256 秒周期中选择的周期输出事件。

注 . 使用 RTC 的事件链接输出功能时，必须在进行了 RTC 的设定（初始化、时刻设定等）后对 ELC 进行设定。如果在设定 ELC 后设定 RTC，就有可能输出意想不到的事件。

25.5.1 中断处理和事件链接的关系

RTC 有控制允许或者禁止周期中断的中断允许位。如果产生中断源，就在中断允许位为允许的情况下向 CPU 输出中断请求信号。

与此相对，事件链接输出信号的输出与中断允许位的设定无关，一旦产生中断源，就作为事件信号经由 ELC 输出到其他模块。

注 . 在软件待机模式和深度软件待机模式中也能输出闹钟中断或者周期中断，但是不能输出 ELC 的周期事件信号。

25.6 使用时的注意事项

25.6.1 有关计数运行时的寄存器写操作

在进行计数时 (RCR2.START 位 =1) 不能写以下寄存器:

RSECCNT、RMINCNT、RHRCNT、RDAYCNT、RWKCNT、RMONCNT、RYRCNT、RCR2.RTCOE、RCR2.HR24

要写上述寄存器时, 必须在暂停计数后写这些寄存器。

25.6.2 有关周期中断的使用

周期中断的使用方法如图 25.13 所示。

对于周期中断, 能通过设定 RCR1.PES[3:0] 位来更改中断的发生周期。因为在发生中断时使用预分频器、R64CNT 和 RSECCNT 计数器, 所以不保证设定 RCR1.PES[3:0] 位后的中断发生周期。

如果通过 RCR2 寄存器进行计数的停止 / 运行、RTC 软件复位或者 30 秒调整, 就会影响中断的发生周期。如果使用时钟误差校正功能, 校正后的中断发生周期就只加减校正值。

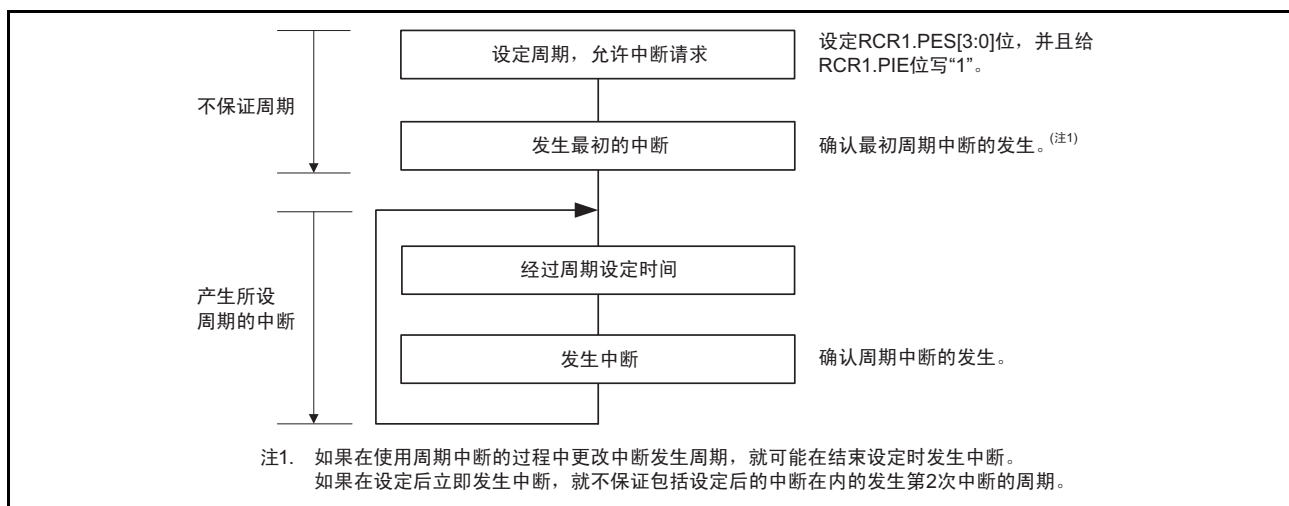


图 25.13 周期中断功能的使用方法

25.6.3 有关 1Hz 时钟的输出

如果通过 RCR2 寄存器进行计数的停止 / 运行、RTC 软件复位或者 30 秒调整, 就会影响 1Hz 时钟的输出周期。如果使用时钟误差校正功能, 校正后的 1Hz 时钟输出周期就只加减校正值。

25.6.4 有关在设定寄存器后向低功耗模式的转移

如果在写 RTC 内的寄存器或者进行寄存器更新处理的过程中转移到低功耗状态 (软件待机模式 / 深度软件待机模式), 就可能破坏寄存器的值。在设定寄存器后, 必须在确认寄存器已被设定后转移到低功耗状态。

25.6.5 读写寄存器时的注意事项

- 在读秒计数器等计数寄存器时, 必须遵循“25.3.5 64Hz 计数器和时间的读取步骤”。
- 如果在写计数寄存器、闹钟寄存器、年闹钟允许寄存器、RCR2.AADJE 位、RCR2.AADJP 位、RCR2.HR24 位、RCR3 寄存器后读这些寄存器, 就从空读 3 次后的读操作开始反映写入值。
- 对于 RCR1.CIE 位和 RCR2.RTCOE 位, 能在写后立即读到写入值。
- 要在从复位或者从软件待机模式、深度软件待机模式返回后读时钟计数器的值时, 必须在时钟运行过程中 (RCR2.START 位 =1) 等待 1/128 秒后读计数器。

26. 看门狗定时器 (WDTA)

看门狗定时器 (WDT) 检测程序的失控。

WDT 内置 14 位递减计数器，如果递减的计数值发生下溢，就能通过复位输出对本 LSI 进行复位，也能选择中断请求的发生。能通过刷新递减计数器的计数值，将计数值返回初始值并且重新开始计数。另外，能设定可刷新期间。如果在可刷新期间进行刷新，就能对计数器进行初始化并且再次进行计数；如果在可刷新期间外进行刷新，就输出复位或者中断请求。因此，能通过调整刷新间隔来检测程序失控。在发生下溢或者不在可刷新期间进行刷新后，WDT 停止计数。在寄存器启动模式中，在刷新后自动重新开始计数；在自动启动模式中，在输出复位或者中断请求后自动重新开始计数。

26.1 概要

WDT 有自动启动模式和寄存器启动模式两种模式，自动启动模式是在解除复位后自动开始计数的模式，寄存器启动模式是在解除复位后通过刷新（写寄存器）开始计数的模式。

在自动启动模式中，在解除复位前给选项功能选择寄存器 0 (OFS0) 设定时钟分频比、刷新的窗口起始 / 结束位置、超时期间、下溢时的复位输出 / 非屏蔽中断请求输出。

在寄存器启动模式中，在解除复位后并且在通过刷新开始计数前给寄存器设定时钟分频比、刷新的窗口起始 / 结束位置、超时期间、下溢时的复位输出 / 非屏蔽中断请求输出。

通过选项功能选择寄存器 0 (OFS0) 的 WDT 启动模式选择位 (OFS0.WDTSTRT) 选择自动启动模式或者寄存器启动模式。

当选择自动启动模式时 (OFS0.WDTSTRT=0)，WDT 控制寄存器 (WDTCR) 和 WDT 复位控制寄存器 (WDTRCR) 的设定无效而选项功能选择寄存器 0 (OFS0) 的设定有效。

当选择寄存器启动模式时 (OFS0.WDTSTRT=1)，选项功能选择寄存器 0 (OFS0) 的设定无效而 WDTCR 寄存器和 WDTRCR 寄存器的设定有效。

WDT 的规格和框图分别如表 26.1 和图 26.1 所示。

表 26.1 WDT 的规格 (1/2)

项目	内容
计数源	外围时钟 (PCLK)
时钟分频比	4 分频、64 分频、128 分频、512 分频、2048 分频、8192 分频
计数	通过 14 位递减计数器进行递减计数。
计数开始条件	<ul style="list-style-type: none"> 复位后自动开始计数 (自动启动模式)。 通过刷新 (在给 WDTRR 寄存器写 "00h" 后写 "FFh") 开始计数 (寄存器启动模式)。
计数停止条件	<ul style="list-style-type: none"> 复位 (递减计数器和寄存器返回初始值) 当发生下溢或者刷新错误时 重新开始计数 (自动启动模式: 在复位或者输出非屏蔽中断请求后自动重新开始计数、寄存器启动模式: 在刷新后重新开始计数)
窗口功能	能设定窗口起始 / 结束位置 (刷新允许 / 禁止期间)。
复位输出源	<ul style="list-style-type: none"> 当递减计数器发生下溢时 在刷新允许期间外进行刷新 (刷新错误) 时
中断请求输出源	<ul style="list-style-type: none"> 在递减计数器发生下溢时产生非屏蔽中断 (WUNI) 时 在刷新允许期间外进行刷新 (刷新错误) 时
读计数器值	能通过读 WDTSR 寄存器，读递减计数器的计数值。
输出信号 (内部信号)	<ul style="list-style-type: none"> 复位输出 中断请求输出

表 26.1 WDT 的规格 (2/2)

项目	内容
自动启动模式 (由选项功能选择寄存器 0 (OFS0) 控制)	<ul style="list-style-type: none"> 选择复位后的时钟分频比 (OFS0.WDTCKS[3:0] 位)。 选择看门狗定时器的超时期限 (OFS0.WDTPPS[1:0] 位)。 选择看门狗定时器的窗口起始位置 (OFS0.WDTRPSS[1:0] 位)。 选择看门狗定时器的窗口结束位置 (OFS0.WDTRPES[1:0] 位)。 选择复位输出或者中断请求输出 (OFS0.WDTRSTIRQS 位)。
寄存器启动模式 (由 WDT 寄存器控制)	<ul style="list-style-type: none"> 选择刷新后的时钟分频比 (WDTCR.CKS[3:0] 位)。 选择看门狗定时器的超时期限 (WDTCR.TOPS[1:0] 位)。 选择看门狗定时器的窗口起始位置 (WDTCR.RPSS[1:0] 位)。 选择看门狗定时器的窗口结束位置 (WDTCR.RPES[1:0] 位)。 选择复位输出或者中断请求输出 (WDTCR.RSTIRQS 位)。

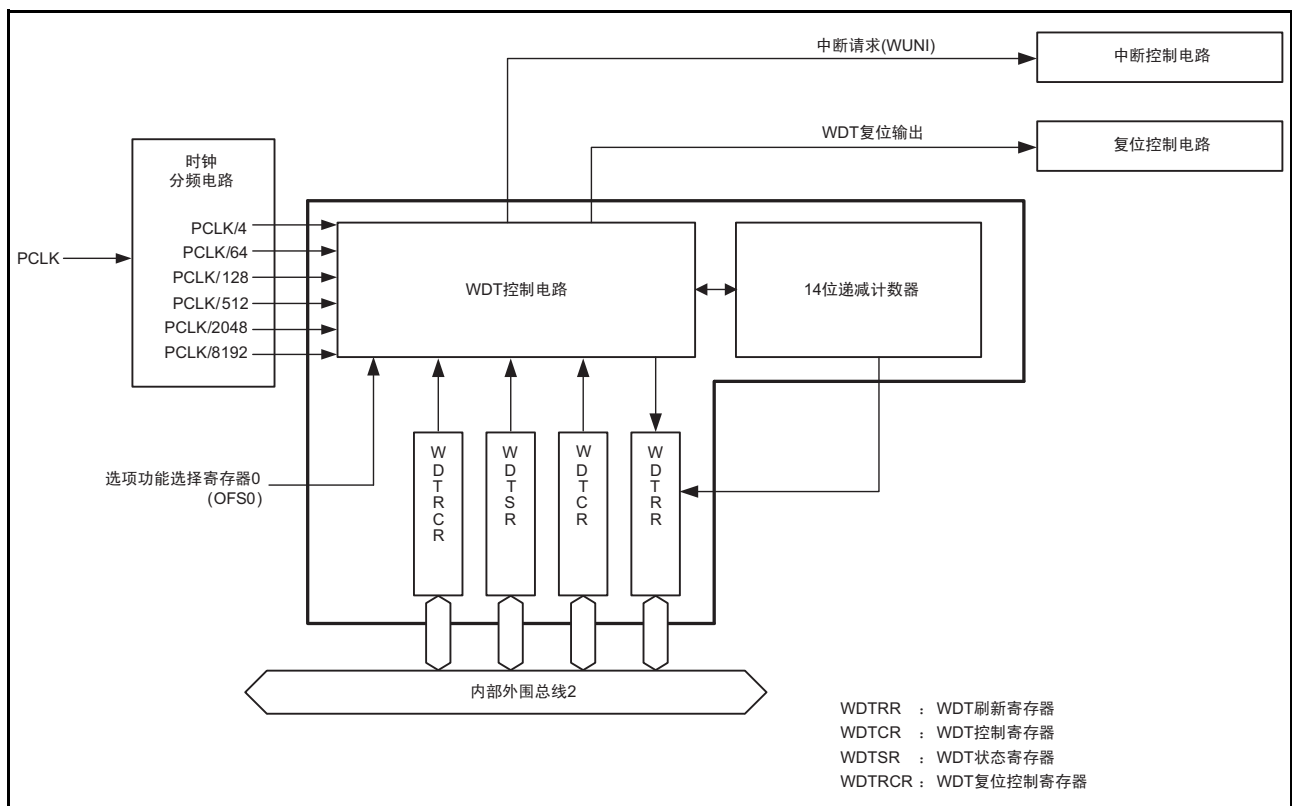
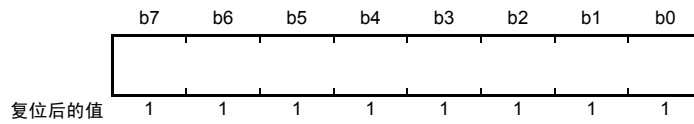


图 26.1 WDT 的框图

26.2 寄存器说明

26.2.1 WDT 刷新寄存器 (WDTRR)

地址 0008 8020h



位	功能	R/W
b7-b0	通过在写“00h”后写“FFh”进行刷新。	R/W

WDTRR 寄存器是刷新 WDT 递减计数器的寄存器。

在刷新允许期间，通过在给 WDTRR 寄存器写“00h”后写“FFh”（刷新），刷新 WDT 递减计数器。

在自动启动模式中，如果递减计数器被刷新，就从选项功能选择寄存器 0 (OFS0) 的 WDT 超时期间选择位 (OFS0.WDTPOPS[1:0]) 设定的值开始递减计数。在寄存器启动模式中，如果递减计数器被刷新，就从 WDT 控制寄存器的超时期间选择位 (WDTCR.TOPS[1:0]) 设定的值开始递减计数，也能通过复位解除后的第 1 次刷新，从 WDTCR.TOPS[1:0] 位设定的值开始递减计数。

如果写“00h”，读取值就为“00h”；如果写“00h”以外的值，读取值就为“FFh”。

有关刷新的详细内容，请参照“26.3.3 刷新”。

26.2.2 WDT 控制寄存器 (WDTCR)

地址 0008 8022h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	RPSS[1:0]	—	—	RPES[1:0]	CKS[3:0]			—	—	TOPS[1:0]				
0	0	1	1	0	0	1	1	1	1	1	1	0	0	1	1

复位后的值

位	符号	位名	功能	R/W
b1-b0	TOPS[1:0]	超时期间选择位	b1 b0 0 0: 1024 个周期 (03FFh) 0 1: 4096 个周期 (0FFFh) 1 0: 8192 个周期 (1FFFh) 1 1: 16384 个周期 (3FFFh)	R/W
b3-b2	—	保留位	读取值为“0”，写操作无效。	R
b7-b4	CKS[3:0]	时钟分频比选择位	b7 b4 0 0 0 1: PCLK/4 0 1 0 0: PCLK/64 1 1 1 1: PCLK/128 0 1 1 0: PCLK/512 0 1 1 1: PCLK/2048 1 0 0 0: PCLK/8192 上述以外: 不能设定	R/W
b9-b8	RPES[1:0]	窗口结束位置选择位	b9 b8 0 0: 75% 0 1: 50% 1 0: 25% 1 1: 0% (没有设定窗口的结束位置)	R/W
b11-b10	—	保留位	读取值为“0”，写操作无效。	R
b13-b12	RPSS[1:0]	窗口起始位置选择位	b13 b12 0 0: 25% 0 1: 50% 1 0: 75% 1 1: 100% (没有设定窗口的开始位置)	R/W
b15-b14	—	保留位	读取值为“0”，写操作无效。	R

WDTCR 寄存器有写操作限制，详细内容请参照“26.3.2 WDTCR 寄存器和 WDTRCR 寄存器的写控制”。

在自动启动模式中，WDTCR 寄存器的设定无效而选项功能选择寄存器 0 (OFS0) 的设定有效。选项功能选择寄存器 0 (OFS0) 的设定和 WDTCR 寄存器各位的设定相同。详细内容请参照“26.3.8 选项功能选择寄存器 0 (OFS0) 和 WDT 寄存器的对应”。

TOPS[1:0] 位 (超时期间选择位)

这些位以 CKS[3:0] 位设定的分频时钟为 1 个周期，从 1024 个周期、4096 个周期、8192 个周期和 16384 个周期中选择递减计数器发生下溢为止的超时期间。

从刷新后到下溢为止的时间 (PCLK 数) 取决于 CKS[3:0] 位和 TOPS[1:0] 位的组合。

CKS[3:0] 位和 TOPS[1:0] 位的设定与超时期间、PCLK 数的关系如表 26.2 所示。

表 26.2 超时期间设定表

CKS[3:0] 位				TOPS[1:0] 位		时钟分频比	超时期间 (周期数)	PCLK 数
b7	b6	b5	b4	b1	b0			
0	0	0	1	0	0	PCLK/4	1024	4096
				0	1		4096	16384
				1	0		8192	32768
				1	1		16384	65536
0	1	0	0	0	0	PCLK/64	1024	65536
				0	1		4096	262144
				1	0		8192	524288
				1	1		16384	1048576
1	1	1	1	0	0	PCLK/128	1024	131072
				0	1		4096	524288
				1	0		8192	1048576
				1	1		16384	2097152
0	1	1	0	0	0	PCLK/512	1024	524288
				0	1		4096	2097152
				1	0		8192	4194304
				1	1		16384	8388608
0	1	1	1	0	0	PCLK/2048	1024	2097152
				0	1		4096	8388608
				1	0		8192	16777216
				1	1		16384	33554432
1	0	0	0	0	0	PCLK/8192	1024	8388608
				0	1		4096	33554432
				1	0		8192	67108864
				1	1		16384	134217728

CKS[3:0] 位 (时钟分频比选择位)

这些位从 4 分频、64 分频、128 分频、512 分频、2048 分频和 8192 分频中选择 WDT 对外围时钟 (PCLK) 进行分频的分频比。能通过和 TOPS[1:0] 位的组合设定, 将 WDT 的计数期间设定在 PCLK 的 4096 ~ 134217728 个时钟之间。

RPES[1:0] 位 (窗口结束位置选择位)

这些位从计数期间的 75%、50%、25% 和 0% 中选择递减计数器的窗口结束位置。窗口结束位置选择小于窗口起始位置的值 (窗口起始位置 > 窗口结束位置)。如果设定窗口结束位置大于窗口起始位置的值, 就只有窗口起始位置的设定有效。

RPSS[1:0] 位和 RPES[1:0] 位设定的窗口起始 / 结束位置的计数器值根据 TOPS[1:0] 位的设定而变。

与 TOPS[1:0] 位的值对应的窗口起始 / 结束位置的计数器值如表 26.3 所示。

表 26.3 超时期间和窗口允许 / 结束位置的计数器值的对应表

TOPS[1:0] 位		超时期间		刷新允许 / 结束计数器值			
b1	b0	周期数	计数器值	100%	75%	50%	25%
0	0	1024	03FFh	03FFh	02FFh	01FFh	00FFh
0	1	4096	0FFFh	0FFFh	0BFFh	07FFh	03FFh
1	0	8192	1FFFh	1FFFh	17FFh	0FFFh	07FFh
1	1	16384	3FFFh	3FFFh	2FFFh	1FFFh	0FFFh

RPSS[1:0] 位 (窗口起始位置选择位)

这些位从计数期间 (开始计数时为 100%、发生下溢时为 0%) 的 100%、75%、50% 和 25% 中选择递减计数器的窗口起始位置。从窗口起始位置到窗口结束位置的期间为刷新允许期间, 其他期间为刷新禁止期间。

RPSS[1:0] 位和 RPES[1:0] 位的设定值与刷新允许 / 禁止期间的关系如图 26.2 所示。

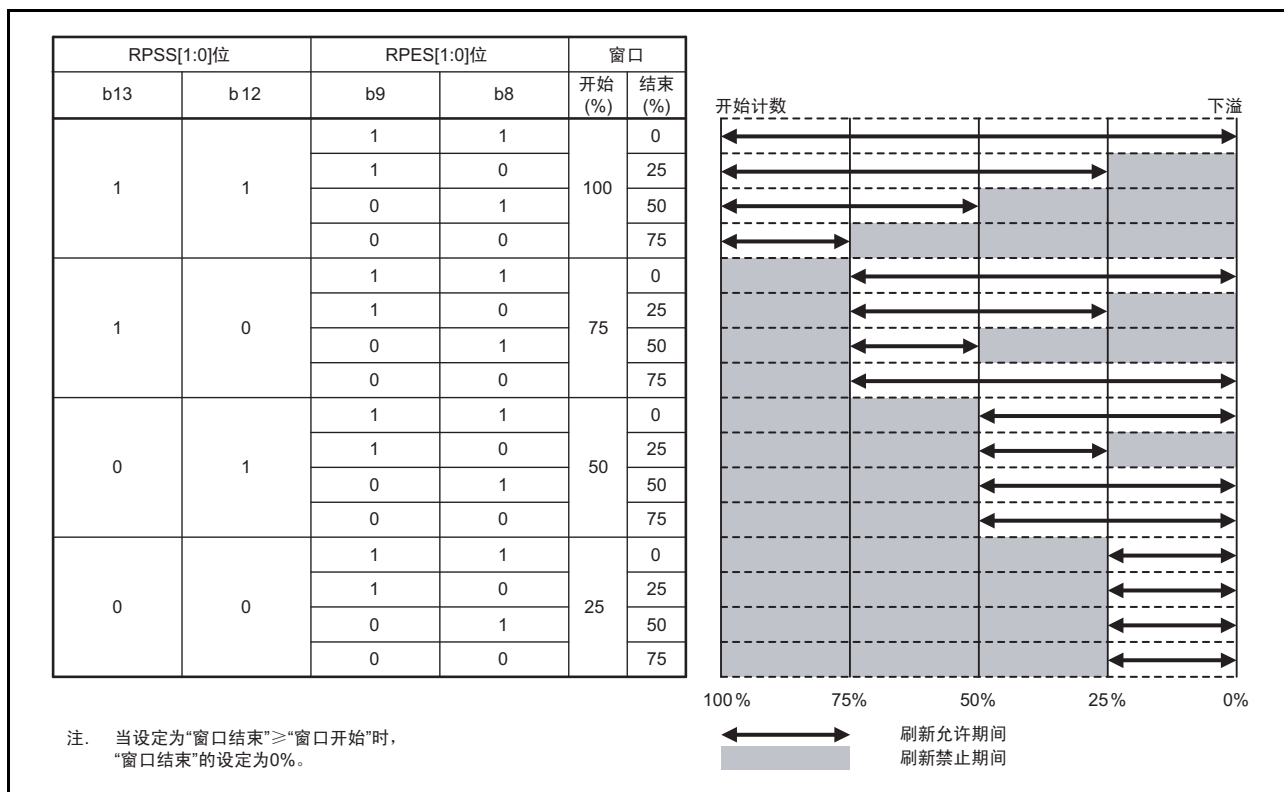
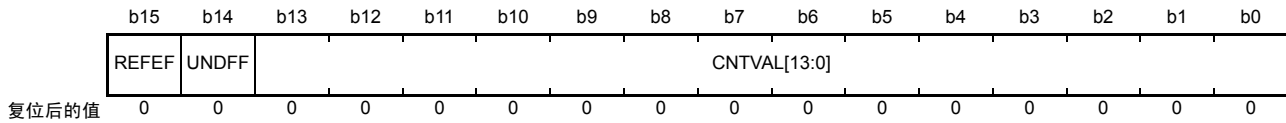


图 26.2 RPSS[1:0] 位、RPES[1:0] 位和刷新允许期间

26.2.3 WDT 状态寄存器 (WDTSR)

地址 0008 8024h



位	符号	位名	功能	R/W
b13-b0	CNTVAL[13:0]	递减计数器值的位	递减计数器的计数值	R
b14	UNDFE	下溢标志	0: 没有发生下溢 1: 发生下溢	R(W) (注1)
b15	REFEF	刷新错误标志	0: 没有发生刷新错误 1: 发生刷新错误	R(W) (注1)

注 1. 为了将标志置“0”，只能写“0”。

通过 WDT 的复位源而不通过其他复位源对 WDTSR 寄存器进行初始化。

CNTVAL[13:0] 位 (递减计数器值的位)

这些位能确认递减计数器的计数值。但是，读到的计数器值有可能和递减计数器的实际值相差 1 个计数时钟。

UNDFE 标志 (下溢标志)

此标志能确认递减计数器的下溢发生状态。

当读取值为“1”时，表示递减计数器发生下溢；当读取值为“0”时，表示递减计数器没有发生下溢。

要将 UNDFE 标志的值置“0”时，必须给此标志写“0”。写“1”无效。

REFEF 标志 (刷新错误标志)

此标志能确认刷新错误 (刷新禁止期间的刷新) 的发生状态。

当读取值为“1”时，表示发生刷新错误；当读取值为“0”时，表示没有发生刷新错误。

要将 REFEF 标志的值置“0”时，必须给此标志写“0”。写“1”无效。

26.2.4 WDT 复位控制寄存器 (WDTRCR)

地址 0008 8026h

	b7	b6	b5	b4	b3	b2	b1	b0
	RSTIR QS	—	—	—	—	—	—	—
复位后的值	1	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b6-b0	—	保留位	读取值为“0”，写操作无效。	R
b7	RSTIRQS	复位中断请求选择位	0: 允许输出非屏蔽中断请求 1: 允许输出复位	R/W

WDTRCR 寄存器有写操作限制，详细内容请参照“26.3.2 WDTCR 寄存器和 WDTRCR 寄存器的写控制”。

在自动启动模式中，WDTRCR 寄存器的设定无效而选项功能选择寄存器 0 (OFS0) 的设定有效。选项功能选择寄存器 0 (OFS0) 的设定和 WDTRCR 寄存器各位的设定相同。详细内容请参照“26.3.8 选项功能选择寄存器 0 (OFS0) 和 WDT 寄存器的对应”。

26.2.5 选项功能选择寄存器 0 (OFS0)

有关选项功能选择寄存器 0 (OFS0)，请参照“26.3.8 选项功能选择寄存器 0 (OFS0) 和 WDT 寄存器的对应”。

26.3 运行说明

26.3.1 各计数开始条件下的运行

通过选项功能选择寄存器 0 (OFS0) 的 WDT 启动模式选择位 (OFS0.WDTSTRT) 选择 WDT 的启动模式。

当 OFS0.WDTSTRT 位为“1” (寄存器启动模式) 时, WDT 控制寄存器 (WDTCR) 和 WDT 复位控制寄存器 (WDTRCR) 的设定有效, 通过刷新 WDT 刷新寄存器 (WDTRR) 开始计数。当 OFS0.WDTSTRT 位为“0” (自动启动模式) 时, OFS0 寄存器有效, 在复位后自动开始计数。

26.3.1.1 寄存器启动模式

当选项功能选择寄存器 0 (OFS0) 的 WDT 启动模式选择位 (OFS0.WDTSTRT) 为“1”时, 为寄存器启动模式, WDT 控制寄存器 (WDTCR) 和 WDT 复位控制寄存器 (WDTRCR) 有效。

在解除复位后, 给 WDTCR 寄存器设定时钟分频比、窗口起始 / 结束位置和超时期间, 并且给 WDTRCR 寄存器设定复位输出或者中断请求输出。然后, 通过刷新将超时期间选择位 (WDTCR.TOPS[1:0]) 选择的值设定到递减计数器并且开始递减计数。

此后, 如果程序正常运行并且在刷新允许期间内进行了刷新, 就在每次刷新时重新设定计数器值并且继续进行递减计数。在此期间, WDT 不输出复位。但是, 如果因程序失控等无法刷新递减计数器而导致递减计数器发生下溢, 或者因在刷新允许期间外进行刷新而发生刷新错误, WDT 就输出复位或者非屏蔽中断请求 (WUNI)。通过设定复位中断请求选择位 (WDTRCR.RSTIRQS), 选择输出复位或者中断请求。

以下条件下的运行例子如图 26.3 所示。

- WDT 启动模式选择位 (OFS0.WDTSTRT): “1” (寄存器启动模式)
- 复位中断请求选择位 (WDTRCR.RSTIRQS): “1” (允许输出复位)
- 窗口起始位置选择位 (WDTCR.RPSS[1:0]): “10b” (75%)
- 窗口结束位置选择位 (WDTCR.RPES[1:0]): “10b” (25%)

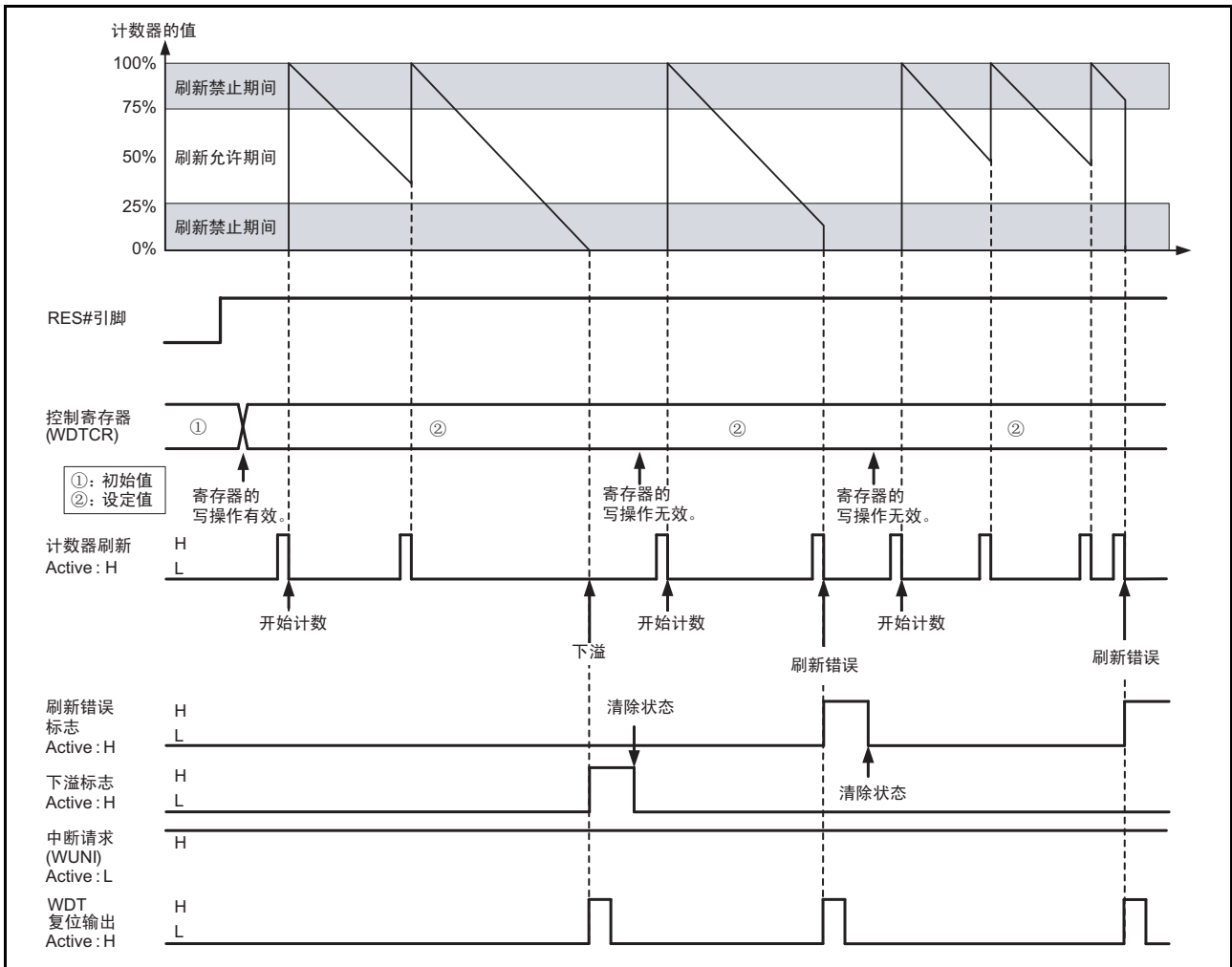


图 26.3 寄存器启动模式的运行例子

26.3.1.2 自动启动模式

当选项功能选择寄存器 0 的 WDT 启动模式选择位 (OFS0.WDTSTRT) 为“0”时，为自动启动模式，WDT 控制寄存器 (WDTCR) 和 WDT 复位控制寄存器 (WDTRCR) 无效。

在复位期间，通过选项功能选择寄存器 0 (OFS0) 设定时钟分频比、窗口起始 / 结束位置、超时期间、复位输出或者中断请求输出。然后通过解除复位将 WDT 超时期间选择位 (OFS0.WDTPS[1:0]) 设定的超时期间值设定到递减计数器并且自动开始递减计数。

此后，如果程序正常运行并且在刷新允许期间内进行了刷新，就在每次刷新时重新设定计数器值并且继续进行递减计数。在此期间，WDT 不输出复位。如果因程序失控等无法刷新递减计数器而导致递减计数器发生下溢，或者因在刷新允许期间外进行刷新而发生刷新错误，WDT 就输出复位或者非屏蔽中断请求 (WUNTI)。在输出 1 个计数周期的复位或者非屏蔽中断请求后，递减计数器重新加载超时期间并且重新开始计数。通过设定 WDT 复位中断请求选择位 (OFS0.WDTRSTIRQS)，选择输出复位或者中断请求。

以下条件下的运行例子如图 26.4 所示。

- WDT 启动模式选择位 (OFS0.WDTSTRT): “0” (自动启动模式)
- 复位中断请求选择位 (OFS0.WDTRSTIRQS): “0” (允许输出非屏蔽中断请求)
- 窗口起始位置选择位 (OFS0.WDTRPSS[1:0]): “10b” (75%)
- 窗口结束位置选择位 (OFS0.WDTRPES[1:0]): “10b” (25%)

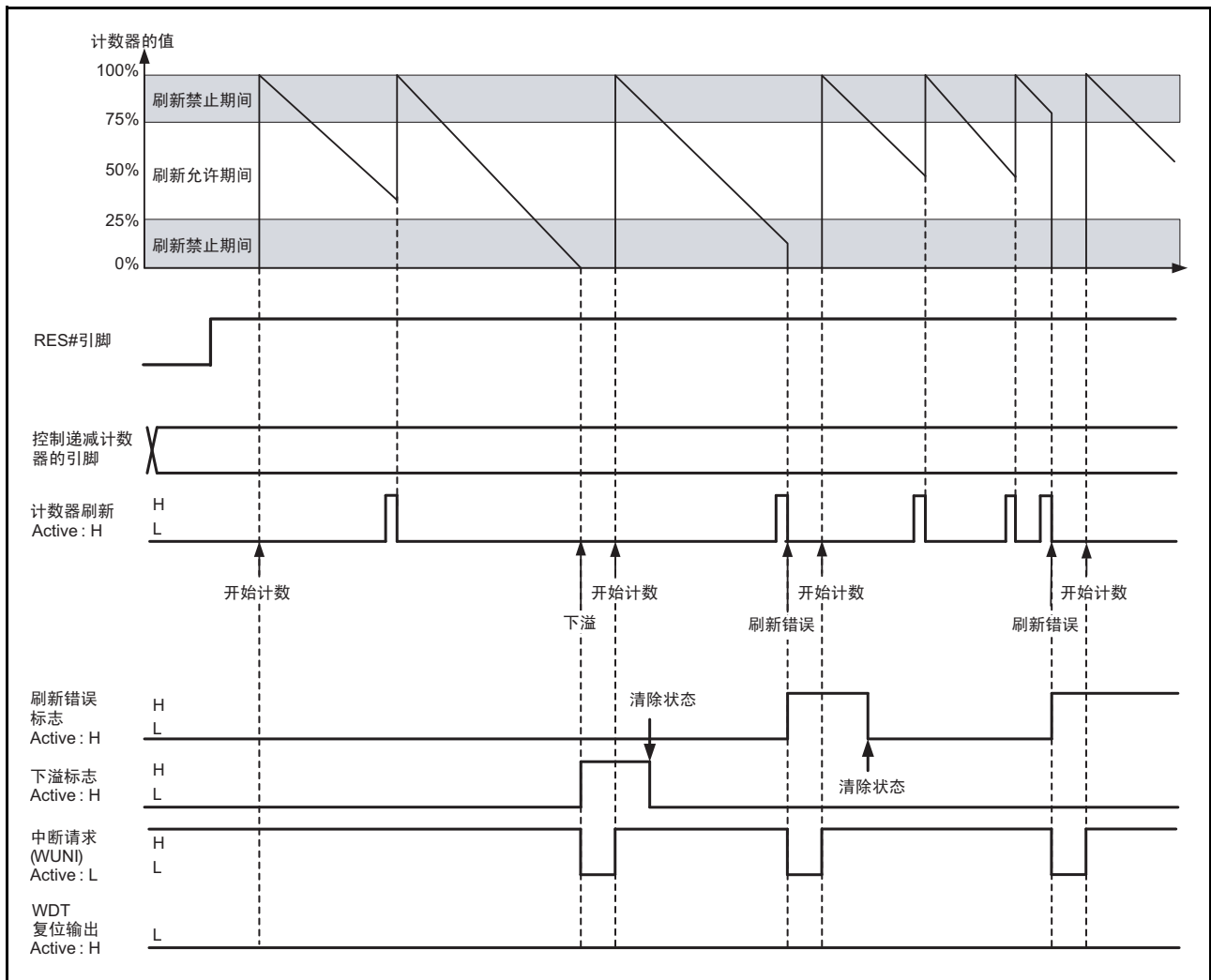


图 26.4 自动启动模式的运行例子

26.3.2 WDTCR 寄存器和 WDTRCR 寄存器的写控制

在从复位解除后到第 1 次刷新的期间，只能写 1 次 WDT 控制寄存器 (WDTCR) 和 WDT 复位控制寄存器 (WDTRCR)。

如果刷新 (开始计数) 或者写 WDTCR 寄存器和 WDTRCR 寄存器，WDT 的内部保护信号就变为“1”，此后保护对 WDTCR 寄存器和 WDTRCR 寄存器的写操作。

通过 WDT 的复位源而不通过其他复位源解除保护。

WDTCR 寄存器的写控制波形如图 26.5 所示。

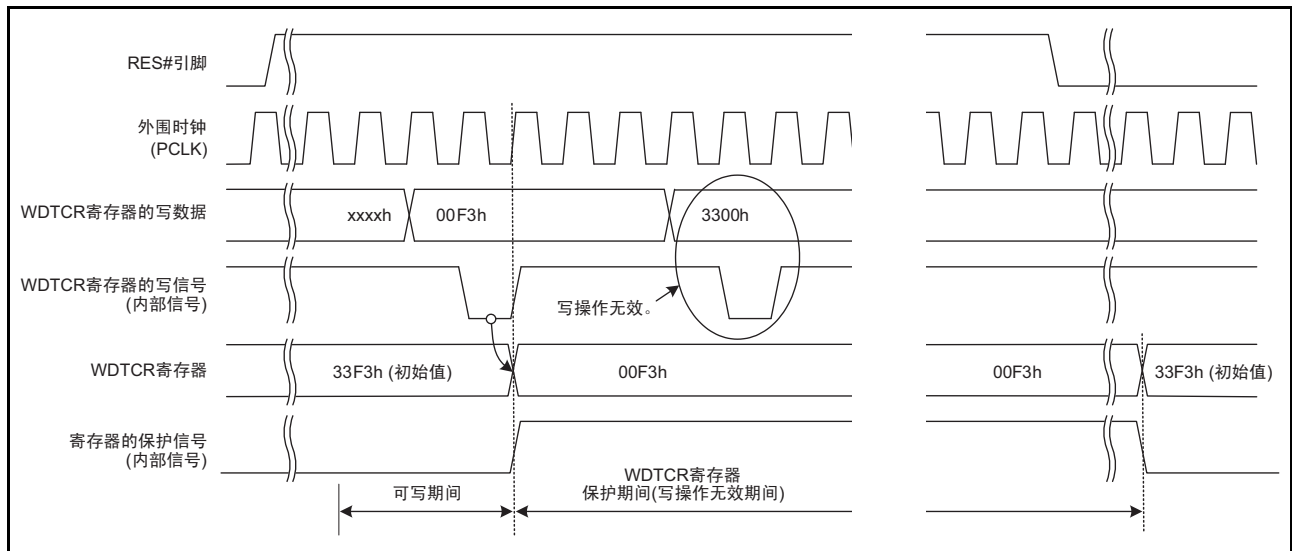


图 26.5 WDTCR 寄存器的写控制波形

26.3.3 刷新

要刷新递减计数器并且使递减计数器开始运行（通过刷新开始计数）时，在给 WDT 的刷新寄存器 (WDTRR) 写 “00h” 后接着写 “FFh”。如果在写 “00h” 后写 “FFh” 以外的值，就不进行刷新。能通过再次按照 “00h”→“FFh” 的顺序写 WDTRR 寄存器，正常地进行刷新。

即使按照 “00h” (第 1 次) → “00h” (第 2 次) 的顺序进行写操作，因为之后通过写 “FFh” 也能使 “00h” → “FFh” 顺序的写操作成立，所以 “00h” (第 n-1 次) → “00h” (第 n 次) → “FFh” 的写操作也有效并且进行刷新。同样，即使在 “00h” 之前写的不是 “00h”，只要 “00h” → “FFh” 顺序的写操作成立，也进行刷新。另外，在给 WDTRR 寄存器写 “00h” 和 “FFh” 的期间，即使存取 WDTRR 以外的寄存器或者读 WDTRR 寄存器，也进行刷新。

【刷新有效的写操作例子】

- “00h” → “FFh”
- “00h” (第 n-1 次) → “00h” (第 n 次) → “FFh”
- “00h” → 存取其他寄存器或者读 WDTRR 寄存器 → “FFh”

【刷新无效的写操作例子】

- “23h” (不是 “00h”) → “FFh”
- “00h” → “54h” (不是 “FFh”)
- “00h” → “AAh” (不是 “00h” 和 “FFh”) → “FFh”

作为刷新运行，即使在刷新允许期间外给 WDTRR 寄存器写 “00h”，只要在刷新允许期间内给 WDTRR 寄存器写 “FFh”，写操作就成立并且进行刷新。

在给 WDTRR 寄存器写 “FFh” 后，递减计数器的刷新时序最多需要 4 个计数周期（1 个周期的外围时钟 (PCLK) 数因时钟分频比选择位 (WDTCR.CKS[3:0]) 的设定值而不同）。因此，在从刷新允许期间结束位置开始到 4 个计数时钟或者递减计数器发生下溢的 4 个计数时钟前为止，必须给 WDTRR 寄存器写完 “FFh”。能通过递减计数器值的位 (WDTSR.CNTVAL[13:0]) 确认递减计数器的值。

【刷新运行的时序例子】

- 当窗口起始位置为 “1FFFh” 时，即使在给 WDTRR 寄存器写 “00h” 后写 “1FFFh” (例如 “2002h”)，只要在 WDTSR.CNTVAL[13:0] 位的值变为 “1FFFh” 后给 WDTRR 寄存器写 “FFh”，也进行刷新。
- 当窗口结束位置为 “1FFFh” 时，如果在按照 “00h” → “FFh” 的顺序写 WDTRR 寄存器后立即读 WDTSR.CNTVAL[13:0] 位的值并且读取值大于等于 “2003h” (“1FFFh” 的 4 个计数时钟前)，就进行刷新。
- 当刷新允许期间持续到 “0000h” 时，能在下溢前进行刷新。此时，如果在按照 “00h” → “FFh” 的顺序写 WDTRR 寄存器后立即读 WDTSR.CNTVAL[13:0] 位的值并且读取值大于等于 “0003h” (下溢的 4 个计数时钟前)，就不产生下溢而进行刷新。

时钟分频比为 PCLK/64 时的 WDT 刷新波形如图 26.6 所示。

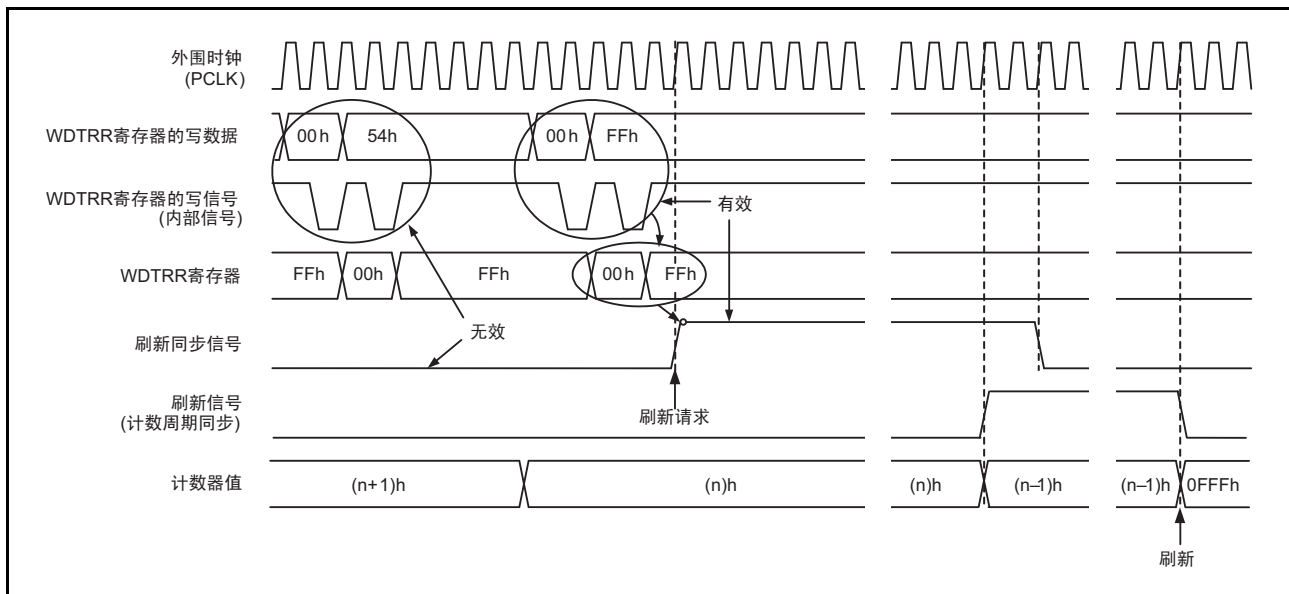


图 26.6 WDT 的刷新波形 (WDTCR.CKS[3:0]=0100b、WDTCR.TOPS[1:0]=01b)

26.3.4 状态标志

刷新错误标志 (WDTSR.REFEF) 和下溢标志 (WDTSR.UNDF) 保持 WDT 输出复位时的复位源或者发生 WDT 中断请求时的中断源。

在解除复位后或者在发生中断请求时，能通过读 WDTSR.REFEF 标志或者 WDTSR.UNDF 标志，确认复位源或者中断源的发生状态。

要将各标志的值置“0”时，必须写“0”。写“1”无效。

即使不将各标志置“0”，也不影响运行。如果不置“0”，就在下次 WDT 输出复位时清除旧的复位源并且写新的复位源，或者在下次 WDT 产生中断请求时清除旧的中断源并且写新的中断源。

另外，要读通过写“0”清除标志后被反映的值时，需要几个 PCLK 时钟（至少 5 个时钟）。

26.3.5 复位输出

如果在寄存器启动模式中将复位中断选择位 (WDTCCR.RSTIRQS) 置“1”或者在自动启动模式中将选项功能选择寄存器 0 (OFS0) 的 WDT 复位中断请求选择位 (OFS0.WDTRSTIRQS) 置“1”，就在发生递减计数器下溢或者刷新错误时输出 1 个计数周期的复位。

在寄存器启动模式中，在输出复位后递减计数器保持初始状态 (ALL“0”)。在解除复位并且重新启动后，通过刷新重新设定计数器值并且开始递减计数。

在自动启动模式中，在输出复位后自动开始递减计数。

26.3.6 中断源

如果在寄存器启动模式中将复位中断选择位 (WDTRCR.RSTIRQS) 置“0”或者在自动启动模式中将选项功能选择寄存器 0 (OFS0) 的 WDT 复位中断请求选择位 (OFS0.WDTRSTIRQS) 置“0”，就在发生递减计数器下溢或者刷新错误时产生非屏蔽中断 (WUNI)。

表 26.4 WDT 的中断源

名称	中断源	DTC 的启动	DMAC 的启动
WUNI	递减计数器的下溢 刷新错误	不能	不能

26.3.7 递减计数器值的读操作

WDT 将计数值保存到 WDT 状态寄存器的递减计数器 (WDTSR.CNTVAL[13:0] 位)。能通过读被保存到 WDTSR.CNTVAL[13:0] 位的值，确认计数器值。

时钟分频比为 PCLK/64 时的递减计数器值的读处理如图 26.7 所示。

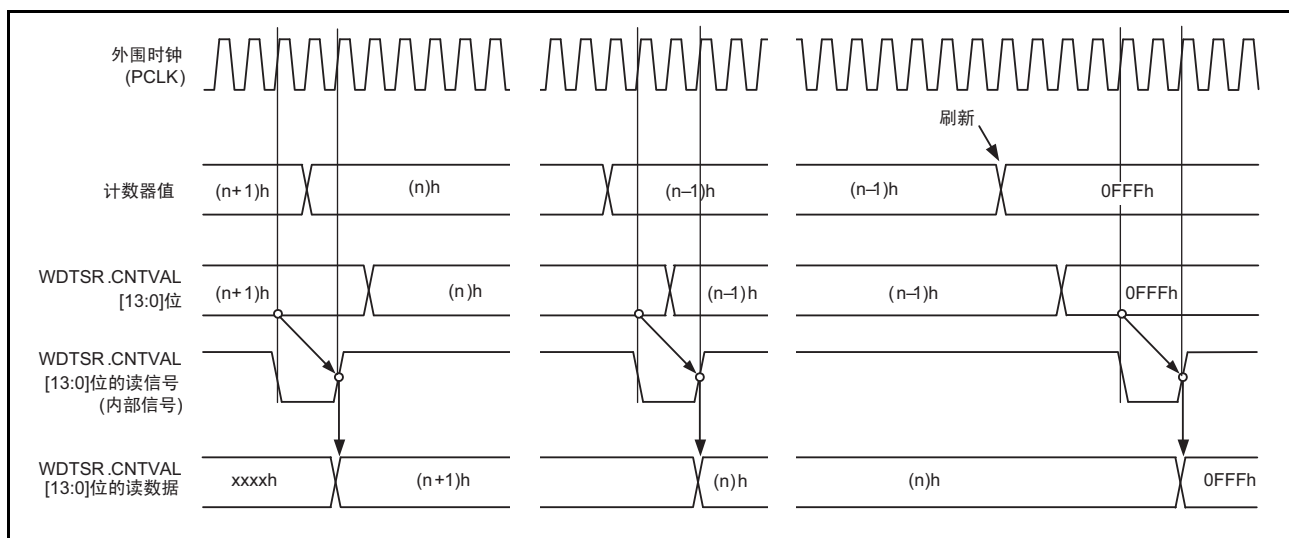


图 26.7 WDT 递减计数器值的读处理 (WDTCR.CKS[3:0]=0100b、WDTCR.TOPS[1:0]=01b)

26.3.8 选项功能选择寄存器 0 (OFS0) 和 WDT 寄存器的对应

通过选项功能选择寄存器 0 (OFS0) 进行的递减计数器控制、复位输出控制或者中断请求输出控制与 WDT 控制寄存器 (WDTCR) 和 WDT 复位控制寄存器 (WDTRCR) 的对应如表 26.5 所示。通过 WDT 启动模式选择位 (OFS0.WDTSTRT) 选择选项功能选择寄存器 0 (OFS0) 和 WDTCR 寄存器、WDTRCR 寄存器控制的有效或者无效。

必须在 WDT 运行过程中固定选项功能选择寄存器 0 (OFS0) 的设定。

有关 OFS0 寄存器，请参照“7.2.1 选项功能选择寄存器 0 (OFS0)”。

表 26.5 选项功能选择寄存器 0 (OFS0) 和 WDT 寄存器的对应

控制	功能	OFS0 寄存器 (在自动启动模式中有效) OFS0.WDTSTRT=0	WDT 寄存器 (在寄存器启动模式中有效) OFS0.WDTSTRT=1
递减计数器	选择超时期间。	OFS0.WDTPSS[1:0]	WDTCR.TOPS[1:0]
	选择时钟分频比。	OFS0.WDTCKS[3:0]	WDTCR.CKS[3:0]
	选择窗口起始位置。	OFS0.WDTRPSS[1:0]	WDTCR.RPSS[1:0]
	选择窗口结束位置。	OFS0.WDTRPES[1:0]	WDTCR.RPES[1:0]
复位输出 / 中断请求输出	选择输出复位或者中断请求。	OFS0.WDTRSTIRQS	WDTRCR.RSTIRQS

26.4 使用时的注意事项

26.4.1 有关刷新运行

在设定刷新时序时，必须考虑到 PCLK 和 IWDTCCLK 的精度，并且设定为在误差范围内即使周期发生变化也能进行刷新的值。

27. 独立看门狗定时器 (IWDTa)

独立看门狗定时器 (IWDT) 是和检测程序失控的传统看门狗定时器独立使用的看门狗定时器。

IWDT 内置 14 位递减计数器，如果递减的计数值发生下溢，就能通过复位输出对本 LSI 进行复位，也能选择中断请求的发生。能通过刷新递减计数器的计数值，将计数值返回初始值并且重新开始计数。另外，能设定可刷新期间。如果在可刷新期间进行刷新，就能对计数器进行初始化并且再次进行计数。如果在可刷新期间外进行刷新，就输出复位或者中断请求。因此，能通过调整刷新间隔检测程序失控。在发生下溢或者在可刷新期间外进行刷新后，IWDT 停止计数。在寄存器启动模式中，在刷新后自动重新开始计数；在自动启动模式中，在输出复位或者中断请求后自动重新开始计数。

27.1 概要

IWDT 有自动启动模式和寄存器启动模式两种模式，自动开始模式是在解除复位后自动开始计数的模式，寄存器启动模式是在解除复位后通过刷新（写寄存器）开始计数的模式。

在自动启动模式中，在解除复位前给选项功能选择寄存器 0 (OFS0) 设定时钟分频比、刷新的窗口起始 / 结束位置、超时期间、下溢时的复位输出 / 非屏蔽中断请求输出以及睡眠模式的计数停止控制。

在寄存器启动模式中，在解除复位后并且在通过刷新开始计数前给寄存器设定时钟分频比、刷新的窗口起始 / 结束位置、超时期间、下溢时的复位输出 / 非屏蔽中断请求输出以及睡眠模式的计数停止控制。

通过选项功能选择寄存器 0 (OFS0) 的 IWDT 启动模式选择位 (OFS0.IWDTSTRT) 选择自动启动模式或者寄存器启动模式。

当选择自动启动模式时 (OFS0.IWDTSTRT=0)，IWDT 控制寄存器 (IWDTCR)、IWDT 复位控制寄存器 (IWDTRCR) 和 IWDT 计数停止控制寄存器 (IWDTCSSTR) 的设定无效而选项功能选择寄存器 0 (OFS0) 的设定有效。

当选择寄存器启动模式时 (OFS0.IWDTSTRT=1)，选项功能选择寄存器 0 (OFS0) 的设定无效而 IWDTCR、IWDTRCR 和 IWDTCSSTR 寄存器的设定有效。

IWDT 的规格如表 27.1 所示。

表 27.1 IWDT 的规格 (1/2)

项目	内容
计数源	IWDT 专用低速时钟 (IWDTCLK)
时钟分频比	1 分频、16 分频、32 分频、64 分频、128 分频、256 分频
计数	通过 14 位递减计数器进行递减计数。
计数开始条件	<ul style="list-style-type: none"> 复位后自动开始计数 (自动启动模式)。 通过刷新 (在给 IWDTRR 寄存器写 "00h" 后写 "FFh") 开始计数 (寄存器启动模式)。
计数停止条件	<ul style="list-style-type: none"> 复位 (递减计数器和寄存器返回初始值) 在发生下溢或者刷新错误时 重新开始计数 (自动启动模式: 在复位或者输出非屏蔽中断请求后自动重新开始计数、寄存器启动模式: 在刷新后重新开始计数)
窗口功能	能设定窗口起始 / 结束位置 (刷新允许 / 禁止期间)。
复位输出源	<ul style="list-style-type: none"> 在递减计数器发生下溢时 在刷新允许期间外进行刷新 (刷新错误) 时
中断请求输出源	<ul style="list-style-type: none"> 在递减计数器发生下溢时产生非屏蔽中断 (WUNI) 时 在刷新允许期间外进行刷新 (刷新错误) 时
读计数器值	能通过读 IWDTSR 寄存器，读递减计数器的计数值。
事件链接功能	<ul style="list-style-type: none"> 在递减计数器发生下溢时 在刷新允许期间外进行刷新 (刷新错误) 时

表 27.1 IWDT 的规格 (2/2)

项目	内容
输出信号 (内部信号)	<ul style="list-style-type: none"> • 复位输出 • 中断请求输出 • 睡眠模式的计数停止控制输出
自动启动模式 (由选项功能选择寄存器 0 (OFS0) 控制)	<ul style="list-style-type: none"> • 选择复位后的时钟分频 (OFS0.IWDTCKS[3:0] 位)。 • 选择看门狗定时器的超时间 (OFS0.IWDTTOPS[1:0] 位)。 • 选择看门狗定时器的窗口起始位置 (OFS0.IWDTRPSS[1:0] 位)。 • 选择看门狗定时器的窗口结束位置 (OFS0.IWDTRPES[1:0] 位)。 • 选择复位输出或者中断请求输出 (OFS0.IWDRSTIRQS 位)。 • 选择在向睡眠模式、软件待机模式、深度软件待机模式或者全模块时钟停止模式转移时停止递减计数 (OFS0.IWDTSLCSTP 位)。
寄存器启动模式 (由 IWDT 寄存器控制)	<ul style="list-style-type: none"> • 选择刷新后的时钟分频比 (IWDTCR.CKS[3:0] 位)。 • 选择看门狗定时器的超时间 (IWDTCR.TOPS[1:0] 位)。 • 选择看门狗定时器的窗口起始位置 (IWDTCR.RPSS[1:0] 位)。 • 选择看门狗定时器的窗口结束位置 (IWDTCR.RPES[1:0] 位)。 • 选择复位输出或者中断请求输出 (IWDTCR.RSTIRQS 位)。 • 选择在向睡眠模式、软件待机模式、深度软件待机模式或者全模块时钟停止模式转移时停止递减计数 (IWDTCSTPR.SLCSTP 位)。

在使用 IWDT 时, 为了使 IWDT 在外围时钟 (PCLK) 停止的情况下运行, 需要外围时钟 (PCLK) 和 IWDT 专用低速时钟 (IWDTCLK)。总线接口部和寄存器部通过 PCLK 运行, 14 位递减计数器和控制电路通过 IWDTCLK 运行。

通过同步电路连接外围时钟运行块和 IWDT 专用低速时钟运行块之间的信号。

IWDT 的框图如图 27.1 所示。

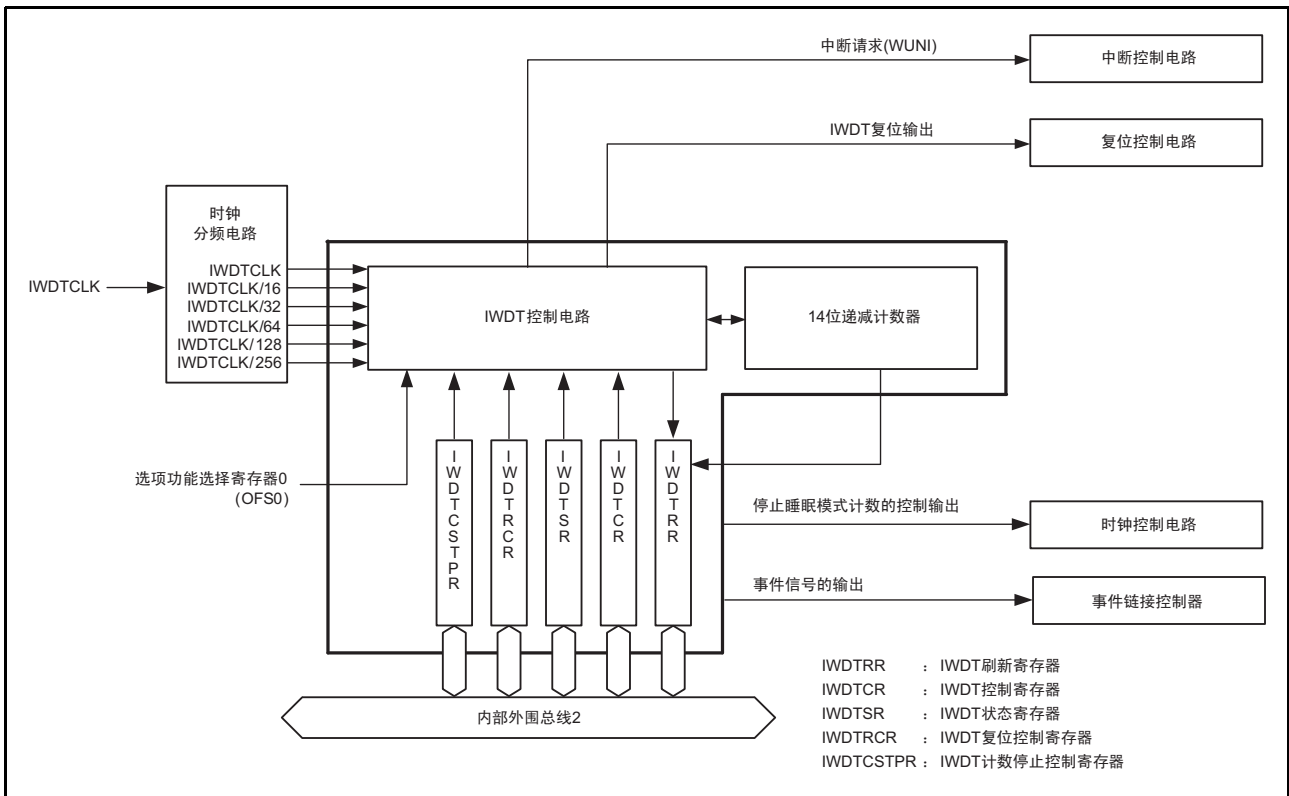
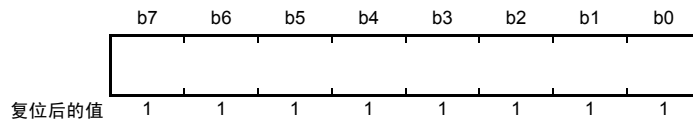


图 27.1 IWDT 的框图

27.2 寄存器说明

27.2.1 IWDT 刷新寄存器 (IWDTRR)

地址 0008 8030h



位	功能	R/W
b7-b0	通过在写“00h”后写“FFh”进行刷新。	R/W

IWDTRR 寄存器是刷新 IWDT 递减计数器的寄存器。

在刷新允许期间，通过在给 IWDTRR 寄存器写“00h”后写“FFh”（刷新），刷新 IWDT 递减计数器。

在自动启动模式中，如果递减计数器被刷新，就从选项功能选择寄存器 0 (OFS0) 的 IWDT 超时期间选择位 (OFS0.IWDTTOPS[1:0]) 设定的值开始递减计数。在寄存器启动模式中，如果递减计数器被刷新，就从 IWDT 控制寄存器的超时期间选择位 (IWDTCR.TOPS[1:0]) 设定的值开始递减计数，也能通过解除复位后的第 1 次刷新，从 IWDTCR.TOPS[1:0] 位设定的值开始递减计数。

如果写“00h”，读取值就为“00h”；如果写“00h”以外的值，读取值就为“FFh”。

有关刷新的详细内容，请参照“27.3.3 刷新”。

27.2.2 IWDT 控制寄存器 (IWDTCR)

地址 0008 8032h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	RPSS[1:0]	—	—	RPES[1:0]	CKS[3:0]			—	—	TOPS[1:0]				
0	0	1	1	0	0	1	1	1	1	1	1	0	0	1	1

复位后的值

位	符号	位名	功能	R/W
b1-b0	TOPS[1:0]	超时期间选择位	b1 b0 0 0: 1024 个周期 (03FFh) 0 1: 4096 个周期 (0FFFh) 1 0: 8192 个周期 (1FFFh) 1 1: 16384 个周期 (3FFFh)	R/W
b3-b2	—	保留位	读取值为“0”，写操作无效。	R
b7-b4	CKS[3:0]	时钟分频比选择位	b7 b4 0 0 0 0: IWDTCLK 0 0 1 0: IWDTCLK/16 0 0 1 1: IWDTCLK/32 0 1 0 0: IWDTCLK/64 1 1 1 1: IWDTCLK/128 0 1 0 1: IWDTCLK/256 上述以外: 不能设定	R/W
b9-b8	RPES[1:0]	窗口结束位置选择位	b9 b8 0 0: 75% 0 1: 50% 1 0: 25% 1 1: 0% (没有设定窗口的结束位置)	R/W
b11-b10	—	保留位	读取值为“0”，写操作无效。	R
b13-b12	RPSS[1:0]	窗口起始位置选择位	b13 b12 0 0: 25% 0 1: 50% 1 0: 75% 1 1: 100% (没有设定窗口的开始位置)	R/W
b15-b14	—	保留位	读取值为“0”，写操作无效。	R

IWDTCR 寄存器有写操作限制。详细内容请参照“27.3.2 IWDTCR 寄存器、IWDTRCR 寄存器和 IWDTCSTPR 寄存器的写控制”。

在自动启动模式中，IWDTCR 寄存器的设定无效而选项功能选择寄存器 0 (OFS0) 的设定有效。选项功能选择寄存器 0 (OFS0) 的设定和 IWDTCR 寄存器各位的设定相同。详细内容请参照“27.3.8 选项功能选择寄存器 0 (OFS0) 和 IWDT 寄存器的对应”。

TOPS[1:0] 位 (超时期间选择位)

这些位以 CKS[3:0] 位设定的分频时钟为 1 个周期，从 1024 个周期、4096 个周期、8192 个周期和 16384 个周期中选择递减计数器发生下溢为止的超时期间。

从刷新后到下溢为止的时间 (IWDTCLK 数) 取决于 CKS[3:0] 位和 TOPS[1:0] 位的组合。

CKS[3:0] 位和 TOPS[1:0] 位的设定与超时期间、IWDTCLK 数的关系如表 27.2 所示。

表 27.2 超时期间设定表

CKS[3:0] 位				TOPS[1:0] 位		时钟分频比	超时期间 (周期数)	IWDTCLK 数
b7	b6	b5	b4	b1	b0			
0	0	0	0	0	0	IWDTCLK	1024	1024
				0	1		4096	4096
				1	0		8192	8192
				1	1		16384	16384
0	0	1	0	0	0	IWDTCLK/16	1024	16384
				0	1		4096	65536
				1	0		8192	131072
				1	1		16384	262144
0	0	1	1	0	0	IWDTCLK/32	1024	32768
				0	1		4096	131072
				1	0		8192	262144
				1	1		16384	524288
0	1	0	0	0	0	IWDTCLK/64	1024	65536
				0	1		4096	262144
				1	0		8192	524288
				1	1		16384	1048576
1	1	1	1	0	0	IWDTCLK/128	1024	131072
				0	1		4096	524288
				1	0		8192	1048576
				1	1		16384	2097152
0	1	0	1	0	0	IWDTCLK/256	1024	262144
				0	1		4096	1048576
				1	0		8192	2097152
				1	1		16384	4194304

CKS[3:0] 位 (时钟分频比选择位)

这些位从 1 分频、16 分频、32 分频、64 分频、128 分频和 256 分频中选择 IWDT 对 IWDTCLK 进行分频的分频比。能通过和 TOPS[1:0] 位的组合，将 IWDT 的计数期间设定在 IWDTCLK 的 1024 ~ 4194304 个时钟之间。

RPES[1:0] 位 (窗口结束位置选择位)

这些位从计数期间的 75%、50%、25% 和 0% 中选择递减计数器的窗口结束位置。窗口结束位置选择小于窗口起始位置的值 (窗口起始位置 > 窗口结束位置)。如果设定窗口结束位置大于窗口起始位置的值，就只有窗口起始位置的设定有效。

RPSS[1:0] 位和 RPES[1:0] 位设定的窗口起始 / 结束位置的计数器值根据 TOPS[1:0] 位的设定而变。

与 TOPS[1:0] 位的值对应的窗口起始 / 结束位置的计数器值如表 27.3 所示。

表 27.3 超时期间和窗口允许 / 结束位置的计数器值的对应表

TOPS[1:0] 位		超时期间		刷新允许 / 结束计数器值			
b1	b0	周期数	计数器值	100%	75%	50%	25%
0	0	1024	03FFh	03FFh	02FFh	01FFh	00FFh
0	1	4096	0FFFh	0FFFh	0BFFh	07FFh	03FFh
1	0	8192	1FFFh	1FFFh	17FFh	0FFFh	07FFh
1	1	16384	3FFFh	3FFFh	2FFFh	1FFFh	0FFFh

RPSS[1:0] 位 (窗口起始位置选择位)

这些位从计数期间 (开始计数时为 100%、发生下溢时为 0%) 的 100%、75%、50% 和 25% 中选择递减计数器的窗口起始位置。从窗口起始位置到窗口结束位置的期间为刷新允许期间, 其他期间为刷新禁止期间。

RPSS[1:0] 位和 RPES[1:0] 位的设定值与刷新允许 / 禁止期间的关系如图 27.2 所示。

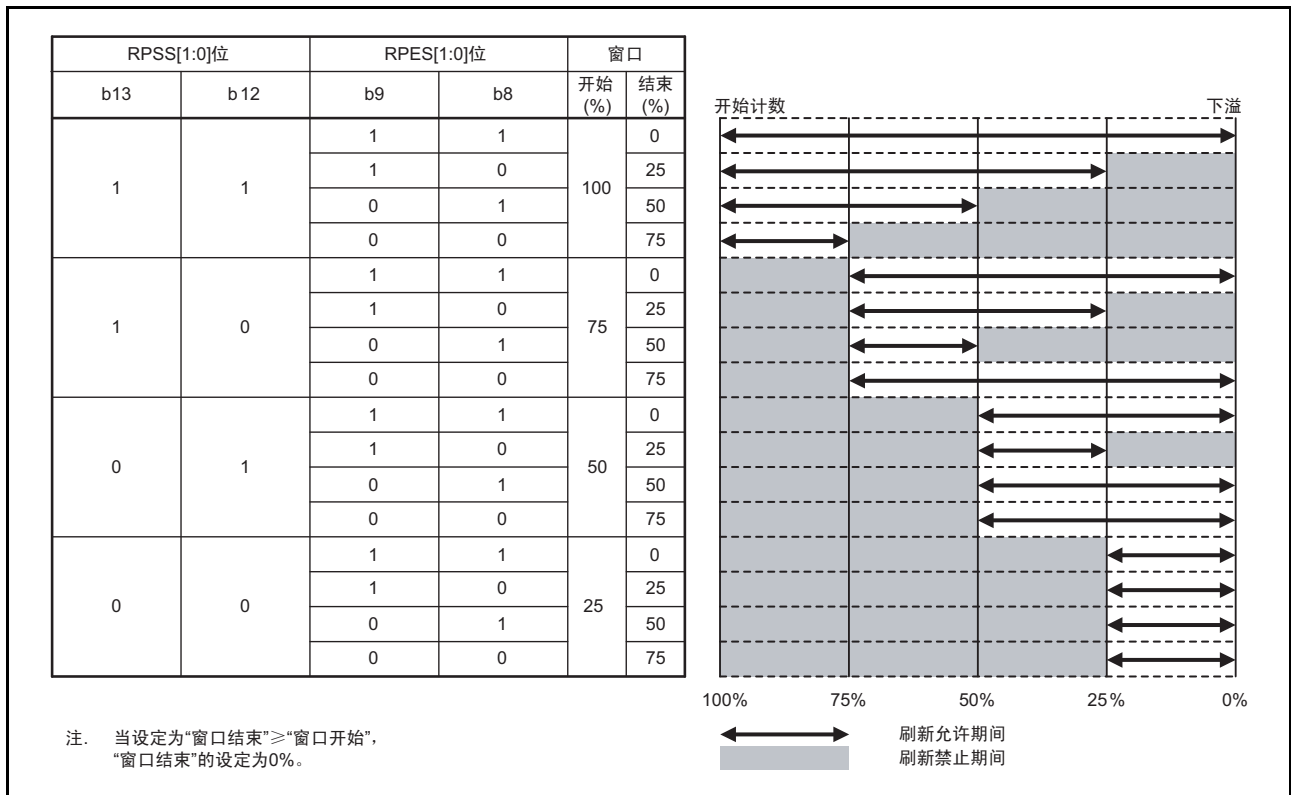
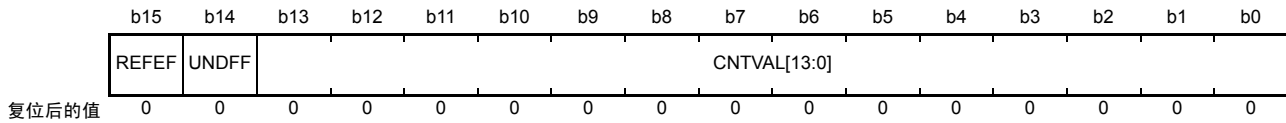


图 27.2 RPSS[1:0] 位、RPES[1:0] 位和刷新允许期间

27.2.3 IWDT 状态寄存器 (IWDTSR)

地址 0008 8034h



位	符号	位名	功能	R/W
b13-b0	CNTVAL[13:0]	递减计数器值的位	递减计数器的计数值	R
b14	UNDFE	下溢标志	0: 没有发生下溢 1: 发生下溢	R(W) (注1)
b15	REFEF	刷新错误标志	0: 没有发生刷新错误 1: 发生刷新错误	R(W) (注1)

注 1. 为了将标志置“0”，只能写“0”。

通过 IWDT 的复位源而不通过其他复位源对 IWDTSR 寄存器进行初始化。

CNTVAL[13:0] 位 (递减计数器值的位)

这些位能确认递减计数器的计数值。但是，读到的计数值有可能和递减计数器的实际值相差 1 个计数时钟。

UNDFE 标志 (下溢标志)

此标志能确认递减计数器的下溢发生状态。

当读取值为“1”时，表示递减计数器发生下溢；当读取值为“0”时，表示递减计数器没有发生下溢。

要将 UNDFE 标志的值置“0”时，必须给此标志写“0”。写“1”无效。

REFEF 标志 (刷新错误标志)

此标志能确认刷新错误 (刷新禁止期间的刷新) 的发生状态。

当读取值为“1”时，表示发生刷新错误；当读取值为“0”时，表示没有发生刷新错误。

要将 REFEF 标志的值置“0”时，必须给此标志写“0”。写“1”无效。

27.2.4 IWDT 复位控制寄存器 (IWDTRCR)

地址 0008 8036h

	b7	b6	b5	b4	b3	b2	b1	b0
	RSTIR QS	—	—	—	—	—	—	—
复位后的值	1	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b6-b0	—	保留位	读取值为“0”，写操作无效。	R
b7	RSTIRQS	复位中断请求选择位	0: 允许输出非屏蔽中断请求 1: 允许输出复位	R/W

IWDTRCR 寄存器有写操作限制，详细内容请参照“27.3.2 IWDTCR 寄存器、IWDTRCR 寄存器和 IWDTCSTPR 寄存器的写控制”。

在自动启动模式中，IWDTRCR 寄存器的设定无效而选项功能选择寄存器 0 (OFS0) 的设定有效。选项功能选择寄存器 0 (OFS0) 的设定和 IWDTRCR 寄存器各位的设定相同。详细内容请参照“27.3.8 选项功能选择寄存器 0 (OFS0) 和 IWDT 寄存器的对应”。

27.2.5 IWDT 计数停止控制寄存器 (IWDTCSTPR)

地址 0008 8038h

	b7	b6	b5	b4	b3	b2	b1	b0
	SLCSTP	—	—	—	—	—	—	—
复位后的值	1	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b6-b0	—	保留位	读取值为“0”，写操作无效。	R
b7	SLCSTP	睡眠模式的计数停止控制位	0: 计数停止无效 1: 在向睡眠模式、软件待机模式、深度软件待机模式和全模块时钟停止模式转移时计数停止有效	R/W

IWDTCSTPR 寄存器对转移到低功耗模式时的 IWDT 的递减计数器的计数停止进行控制。IWDTCSTPR 寄存器有写操作限制，详细内容请参照“27.3.2 IWDTCR 寄存器、IWDTRCR 寄存器和 IWDTCSTPR 寄存器的写控制”。

在自动启动模式中，IWDTCSTPR 寄存器的设定无效而选项功能选择寄存器 0 (OFS0) 的设定有效。选项功能选择寄存器 0 (OFS0) 的设定和 IWDTCSTPR 寄存器各位的设定相同。详细内容请参照“27.3.8 选项功能选择寄存器 0 (OFS0) 和 IWDT 寄存器的对应”。

SLCSTP 位 (睡眠模式的计数停止控制位)

此位选择在向睡眠模式、软件待机模式、深度软件待机模式和全模块时钟停止模式转移时停止计数。

27.2.6 选项功能选择寄存器 0 (OFS0)

有关选项功能选择寄存器 0 (OFS0)，请参照“27.3.8 选项功能选择寄存器 0 (OFS0) 和 IWDT 寄存器的对应”。

27.3 运行说明

27.3.1 各计数开始条件下的运行

通过选项功能选择寄存器 0 (OFS0) 的 IWDT 启动模式选择位 (OFS0.IWDTSTRT) 选择 IWDT 的启动模式。

当 OFS0.IWDTSTRT 位为“1” (寄存器启动模式) 时, IWDT 控制寄存器 (IWDTCR)、IWDT 复位控制寄存器 (IWDTRCR) 和 IWDT 计数停止控制寄存器 (IWDTCSTPR) 的设定有效, 通过刷新 IWDT 刷新寄存器 (IWDTRR) 开始计数。当 OFS0.IWDTSTRT 位为“0” (自动启动模式) 时, 选项功能选择寄存器 0 (OFS0) 有效, 在复位后自动开始计数。

27.3.1.1 寄存器启动模式

当选项功能选择寄存器 0 (OFS0) 的 IWDT 启动模式选择位 (OFS0.IWDTSTRT) 为“1”时, 为寄存器启动模式, IWDT 控制寄存器 (IWDTCR)、IWDT 复位控制寄存器 (IWDTRCR) 和 IWDT 计数停止控制寄存器 (IWDTCSTPR) 有效。

在解除复位后, 给 IWDTCR 寄存器设定时钟分频比、窗口起始 / 结束位置和超时期间, 给 IWDTRCR 寄存器设定复位输出或者中断请求输出, 给 IWDTCSTPR 寄存器设定向低功耗模式转移时的 IWDT 递减计数器的计数停止控制。然后, 通过刷新将超时期间选择位 (IWDTCR.TOPPS[1:0]) 选择的值设定到递减计数器并且开始递减计数。

此后, 如果程序正常运行并且在刷新允许期间内进行了刷新, 就在每次刷新时重新设定计数器值并且继续进行递减计数。在此期间, IWDT 不输出复位。但是, 如果因程序失控等无法刷新递减计数器而导致递减计数器发生下溢, 或者因在刷新允许期间外进行刷新而发生刷新错误, IWDT 就输出复位或者非屏蔽中断请求 (WUNI)。通过设定复位中断请求选择位 (IWDTRCR.RSTIRQS), 选择输出复位或者中断请求。

以下条件下的运行例子如图 27.3 所示。

- IWDT 启动模式选择位 (OFS0.IWDTSTRT): “1” (寄存器启动模式)
- 复位中断请求选择位 (IWDTRCR.RSTIRQS): “1” (允许输出复位)
- 窗口起始位置选择位 (IWDTCR.RPSS[1:0]): “10b” (75%)
- 窗口结束位置选择位 (IWDTCR.RPES[1:0]): “10b” (25%)

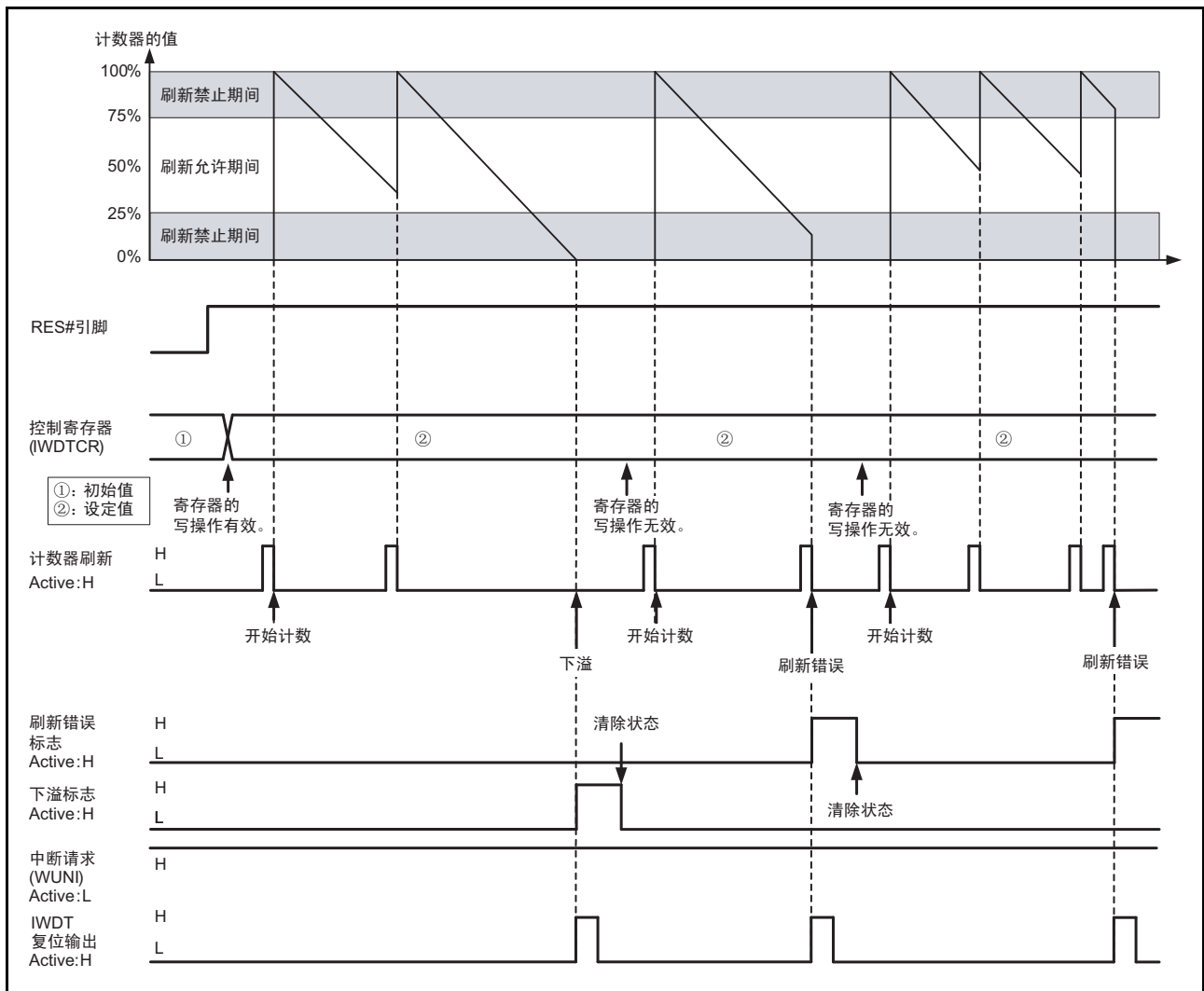


图 27.3 寄存器启动模式的运行例子

27.3.1.2 自动启动模式

当选项功能选择寄存器 0 (OFS0) 的 IWDT 启动模式选择位 (OFS0.IWDTSTRT) 为“0”时，为自动启动模式，IWDT 控制寄存器 0 (IWDTCR)、IWDT 复位控制寄存器 (IWDTRCR) 和 IWDT 计数停止控制寄存器 (IWDTCSTPR) 无效。

在复位期间，通过选项功能选择寄存器 0 (OFS0) 设定时钟分频比、窗口起始 / 结束位置、超时期间、复位输出或者中断请求输出以及向低功耗模式转移时的 IWDT 递减计数器的计数停止控制。然后通过解除复位将 IWDT 超时期间选择位 (OFS0.IWDTTPOPS[1:0]) 设定的超时期间值设定到递减计数器并且自动开始递减计数。

此后，如果程序正常运行并且在刷新允许期间内进行了刷新，就在每次刷新时重新设定计数器值并且继续进行递减计数。在此期间，IWDT 不输出复位。如果因程序失控等无法刷新递减计数器而导致递减计数器发生下溢，或者因在刷新允许期间外进行刷新而发生刷新错误，IWDT 就输出复位或者非屏蔽中断请求 (WUNI)。在输出 1 个计数周期的复位或者非屏蔽中断请求后，递减计数器重新加载超时期间并且重新开始计数。通过设定 IWDT 复位中断请求选择位 (OFS0.IWDRSTIRQS)，选择输出复位或者中断请求。

以下条件下的运行例子如图 27.4 所示。

- IWDT 启动模式选择位 (OFS0.IWDTSTRT)：“0” (自动启动模式)
- 复位中断请求选择位 (OFS0.IWDRSTIRQS)：“0” (允许输出非屏蔽中断请求)
- 窗口起始位置选择位 (OFS0.IWDTRPSS[1:0])：“10b” (75%)
- 窗口结束位置选择位 (OFS0.IWDRPES[1:0])：“10b” (25%)

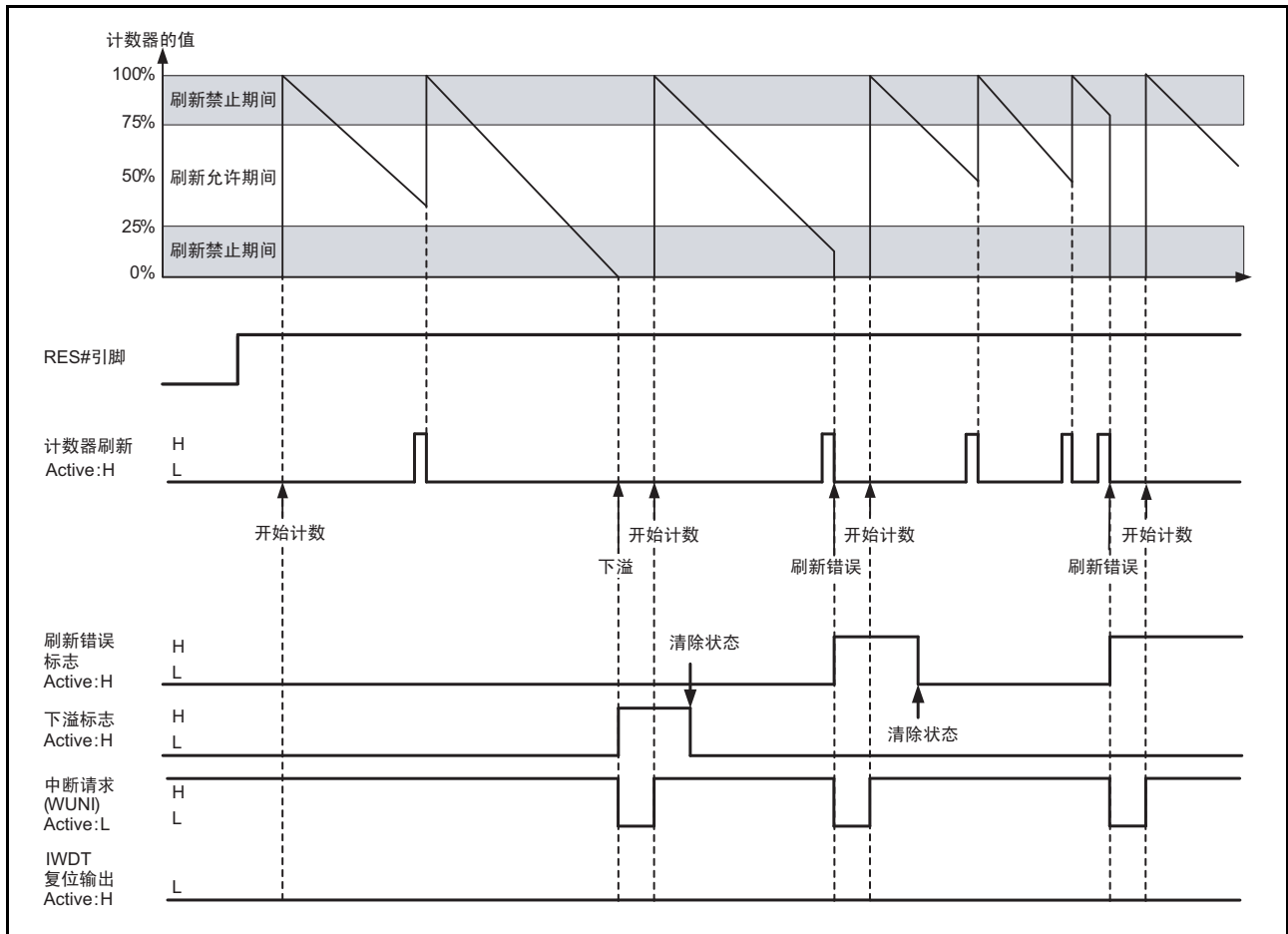


图 27.4 自动启动模式的运行例子

27.3.2 IWDTCR 寄存器、IWDTRCR 寄存器和 IWDTCSSTPR 寄存器的写控制

在从复位解除后到第 1 次刷新的期间，只能写 1 次 IWDT 控制寄存器 (IWDTCR)、IWDT 复位控制寄存器 (IWDTRCR) 和 IWDT 计数停止控制寄存器 (IWDTCSSTPR)。

如果刷新 (开始计数) 或者写 IWDTCR、IWDTRCR、IWDTCSSTPR 寄存器，IWDT 的内部保护信号就变为“1”，此后保护对 IWDTCR、IWDTRCR 和 IWDTCSSTPR 寄存器的写操作。

通过 IWDT 的复位源而不通过其他复位源解除保护。

IWDTCR 寄存器的写控制波形如图 27.5 所示。

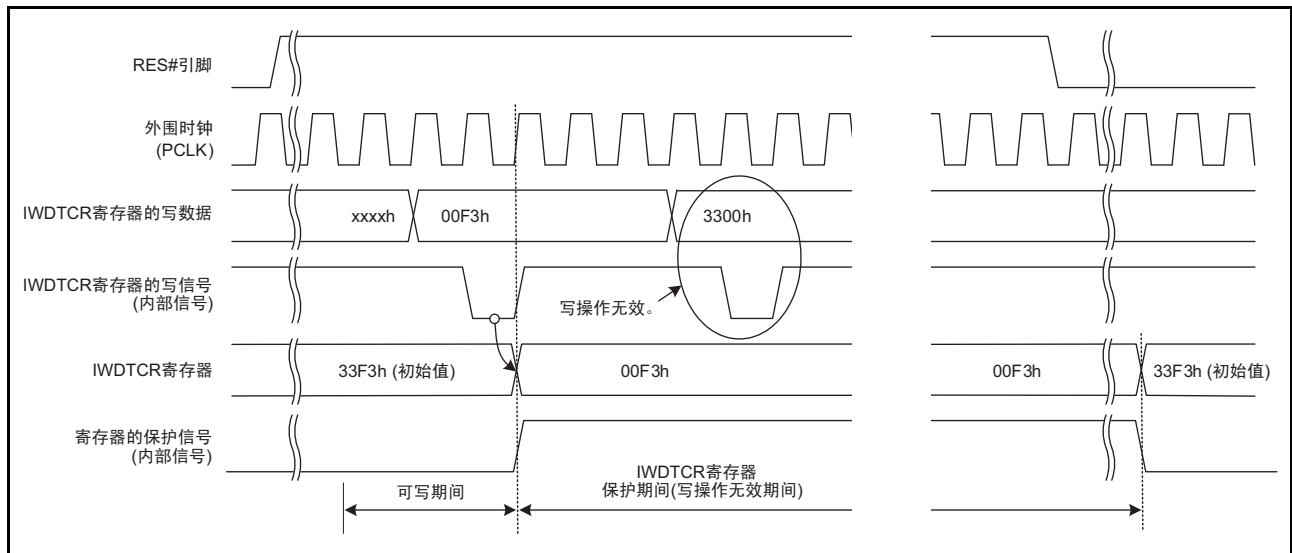


图 27.5 IWDTCR 寄存器的写控制波形

27.3.3 刷新

要刷新递减计数器并且使递减计数器开始运行（通过刷新开始计数）时，在给 IWDTR 刷新寄存器（IWDTRR）写“00h”后接着写“FFh”。如果在写“00h”后写“FFh”以外的值，就不进行刷新。能通过再次按照“00h”→“FFh”的顺序写 IWDTRR 寄存器，正常地进行刷新。

即使按照“00h”（第 1 次）→“00h”（第 2 次）的顺序进行写操作，因为之后通过写“FFh”能使“00h”→“FFh”顺序的写操作成立，所以“00h”（第 n-1 次）→“00h”（第 n 次）→“FFh”的写操作也有效并且进行刷新。同样，即使在“00h”之前写的不是“00h”，只要“00h”→“FFh”顺序的写操作成立，也进行刷新。另外，在给 IWDTRR 寄存器写“00h”和“FFh”的期间，即使存取 IWDTRR 以外的寄存器或者读 IWDTRR 寄存器，也进行刷新。

【刷新有效的写操作例子】

“00h”→“FFh”

“00h”（第 n-1 次）→“00h”（第 n 次）→“FFh”

“00h”→存取其他寄存器或者读 IWDTRR 寄存器→“FFh”

【刷新无效的写操作例子】

“23h”（不是“00h”）→“FFh”

“00h”→“54h”（不是“FFh”）

“00h”→“AAh”（不是“00h”和“FFh”）→“FFh”

作为刷新运行，即使在刷新允许期间外给 IWDTRR 寄存器写“00h”，只要在刷新允许期间内给 IWDTRR 寄存器写“FFh”，写操作就成立并且进行刷新。

在给 IWDTRR 寄存器写“FFh”后，递减计数器的刷新时序最多需要 4 个计数周期（1 个周期期间的 IWDTR 专用低速时钟（IWDTCCLK）数因时钟分频比选择位（IWDTCR.CKS[3:0]）的设定值而不同）。因此，在从刷新允许期间结束位置开始到 4 个计数周期或者递减计数器发生下溢的 4 个计数周期前为止，必须给 IWDTRR 寄存器写完“FFh”。能通过递减计数器值的位（IWDTSR.CNTVAL[13:0]）确认递减计数器的值。

【刷新运行的时序例子】

- 当窗口起始位置为“1FFFh”时，即使在给 IWDTRR 寄存器写“00h”后写“1FFFh”（例如“2002h”），只要在 IWDTSR.CNTVAL[13:0] 位的值变为“1FFFh”后给 IWDTRR 寄存器写“FFh”，也进行刷新。
- 当窗口结束位置为“1FFFh”时，如果在按照“00h”→“FFh”的顺序写 IWDTRR 寄存器后立即读 IWDTSR.CNTVAL[13:0] 位的值并且读取值大于等于“2003h”（“1FFFh”的 4 个计数时钟前），就进行刷新。
- 当刷新允许期间持续到“0000h”时，能在下溢前进行刷新，此时，如果在按照“00h”→“FFh”的顺序写 IWDTRR 寄存器后立即读 IWDTSR.CNTVAL[13:0] 位的值并且读取值大于等于“0003h”（下溢的 4 个计数时钟前），就不产生下溢而进行刷新。

PCLK > IWDTCCLK 并且时钟分频比为 IWDTCCLK 时的 IWDTR 的刷新波形如图 27.6 所示，PCLK < IWDTCCLK 并且时钟分频比为 IWDTCCLK/16 时的 IWDTR 的刷新波形如图 27.7 所示。

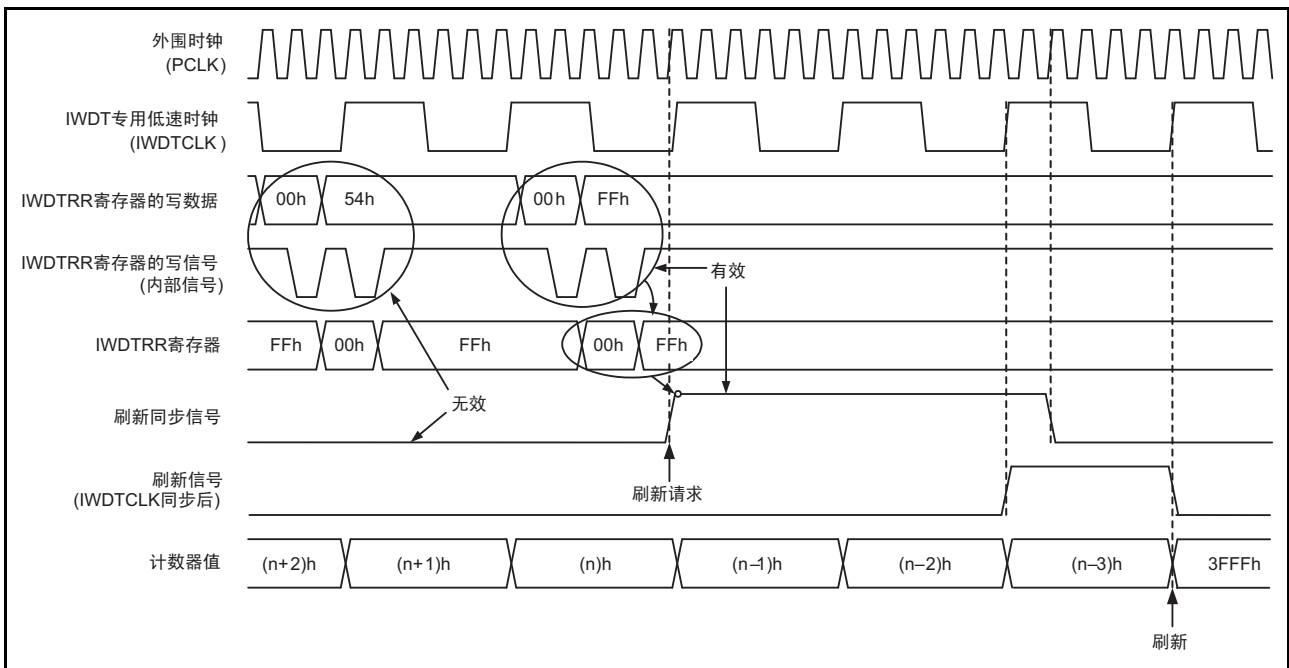


图 27.6 IWDT 的刷新波形 (IWDTCR.CKS[3:0]=0000b、IWDTCR.TOPS[1:0]=11b)

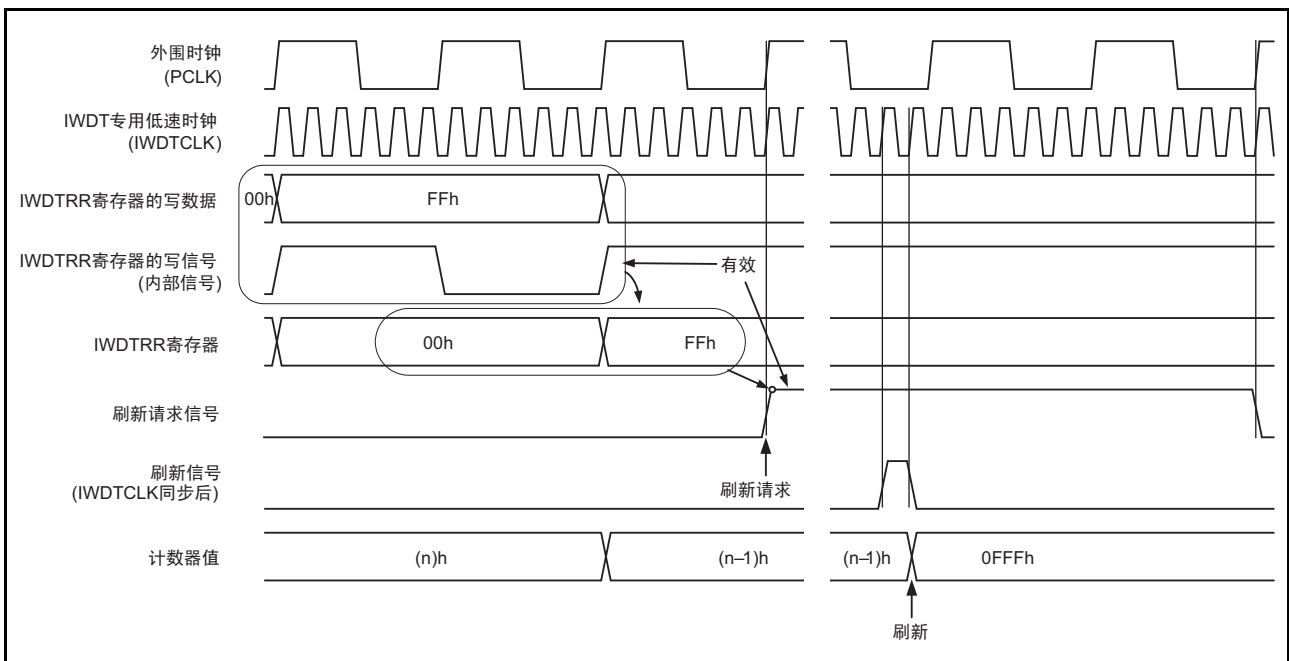


图 27.7 IWDT 的刷新波形 (IWDTCR.CKS[3:0]=0010b、IWDTCR.TOPS[1:0]=01b)

27.3.4 状态标志

刷新错误标志 (IWDTSR.REFEF) 和下溢标志 (IWDTSR.UNDF) 保持 IWDT 输出复位时的复位源或者发生中断请求时的中断源。

在解除复位后或者在发生中断请求时, 能通过读 IWDTSR.REFEF 标志或者 IWDTSR.UNDF 标志, 确认复位源或者中断源的发生状态。

要将各标志的值置“0”时, 必须写“0”。写“1”无效。

即使不将各标志置“0”, 也不影响运行。如果不置“0”, 就在下次 IWDT 输出复位时清除以前的复位源并且写新的复位源, 或者在下次 IWDT 发生中断请求时清除以前的中断源并且写新的中断源。

另外, 要读通过写“0”清除标志后被反映的值时, 需要几个 IWDTCLK 时钟 (至少 3 个时钟) 和几个 PCLK 时钟 (至少 2 个时钟)。

27.3.5 复位输出

如果在寄存器启动模式中将复位中断选择位 (IWDTRCR.RSTIRQS) 置“1”或者在自动启动模式中将选项功能选择寄存器 0 (OFS0) 的 IWDT 复位中断请求选择位 (OFS0.IWDTRSTIRQS) 置“1”, 就在发生递减计数器下溢或者刷新错误时输出 1 个计数周期的复位。

在寄存器启动模式中, 在输出复位后递减计数器保持初始状态 (ALL“0”)。在解除复位并且重新启动后, 通过刷新重新设定计数器值并且开始递减计数。

在自动启动模式中, 在输出复位后自动开始递减计数。

27.3.6 中断源

如果在寄存器启动模式中将复位中断选择位 (IWDTRCR.RSTIRQS) 置“0”或者在自动启动模式中将选项功能选择寄存器 0 (OFS0) 的 IWDT 复位中断请求选择位 (OFS0.IWDTRSTIRQS) 置“0”, 就在发生递减计数器下溢或者刷新错误时产生非屏蔽中断 (WUNI)。

表 27.4 IWDT 的中断源

名称	中断源	DTC 的启动	DMAC 的启动
WUNI	递减计数器的下溢 刷新错误	不能	不能

27.3.7 递减计数器值的读操作

因为 IWDT 的递减计数器通过 IWDT 专用低速时钟 (IWDTCLK) 运行, 所以不能直接读计数器值。因此, IWDT 通过外围时钟 (PCLK) 与计数值同步并且保存到 IWDT 状态寄存器的递减计数器 (IWDTSR.CNTVAL[13:0] 位)。能通过读被保存到 IWDTSR.CNTVAL[13:0] 位的值, 间接确认计数器值。

因为读操作需要几个 PCLK 时钟 (最多 4 个时钟), 所以读到的计数值和递减计数器的实际值有可能相差 1 个计数时钟。

PCLK > IWDTCLK 并且时钟分频比为 IWDTCLK 时的 IWDT 递减计数器值的读处理如图 27.8 所示, PCLK < IWDTCLK 并且时钟分频比为 IWDTCLK/16 时的 IWDT 递减计数器值的读处理如图 27.9 所示。

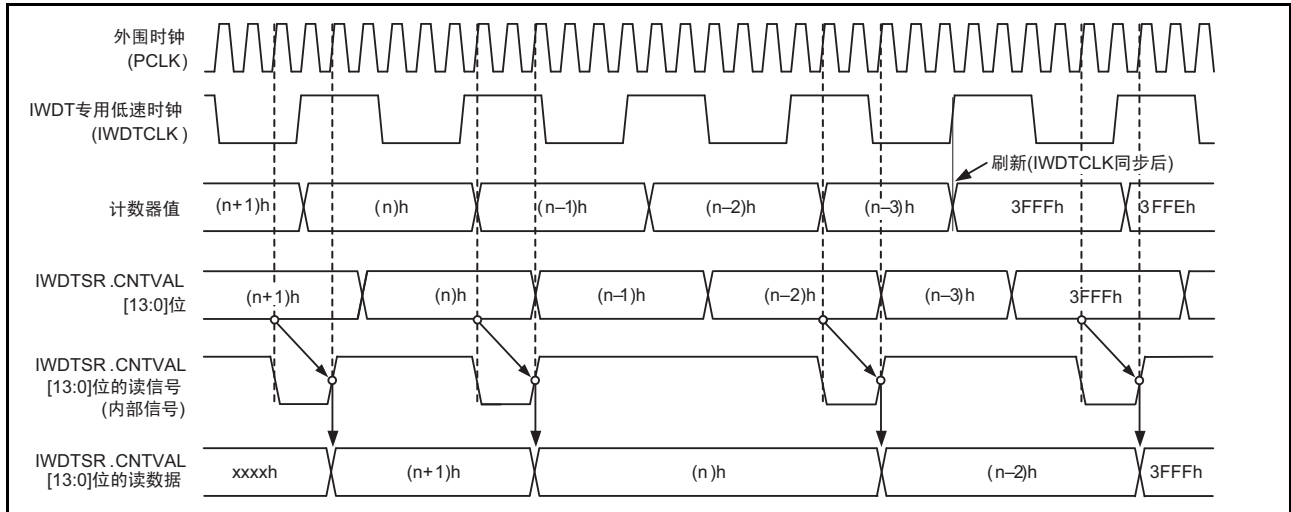


图 27.8 IWDT 递减计数器值的读处理 (IWDTCR.CKS[3:0]=0000b、IWDTCR.TOPS[1:0]=11b)

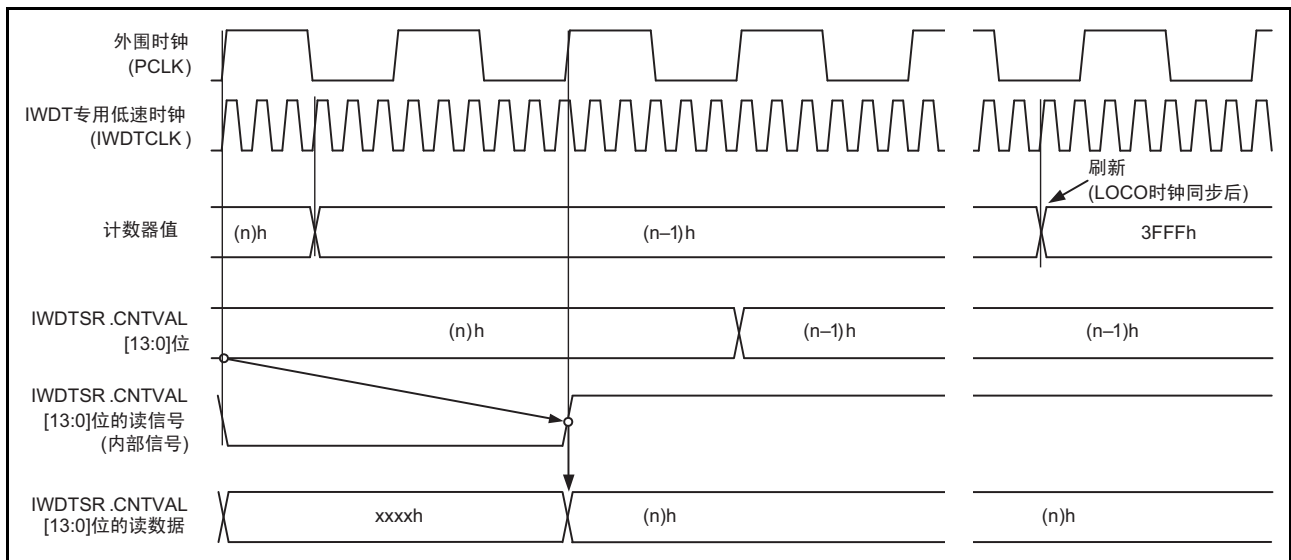


图 27.9 IWDT 递减计数器值的读处理 (IWDTCR.CKS[3:0]=0010b、IWDTCR.TOPS[1:0]=11b)

27.3.8 选项功能选择寄存器 0 (OFS0) 和 IWDT 寄存器的对应

通过选项功能选择寄存器 0 (OFS0) 进行的递减计数器控制、复位输出控制、中断请求输出控制、计数停止控制和 IWDT 控制寄存器 (IWDTCR)、IWDT 复位控制寄存器 (IWDTRCR)、IWDT 计数停止控制寄存器 (IWDTCSSTPR) 的对应如表 27.5 所示。通过 IWDT 启动模式选择位 (OFS0.IWDTSTRT) 选择选项功能选择寄存器 0 (OFS0) 和 IWDTCR、IWDTRCR、IWDTCSSTPR 寄存器控制的有效或者无效。

必须在 IWDT 运行过程中固定选项功能选择寄存器 0 (OFS0) 的设定。

有关选项功能选择寄存器 0 (OFS0)，请参照“7.2.1 选项功能选择寄存器 0 (OFS0)”。

表 27.5 选项功能选择寄存器 0 (OFS0) 和 IWDT 寄存器的对应

控制	功能	OFS0 寄存器 (在自动启动模式中有效) OFS0.IWDTSTRT=0	IWDT 寄存器 (在寄存器启动模式中有效) OFS0.IWDTSTRT=1
递减计数器	选择超时期间。	OFS0.IWDTTOPS[1:0]	IWDTCR.TOPS[1:0]
	选择时钟分频比。	OFS0.IWDTCKS[3:0]	IWDTCR.CKS[3:0]
	选择窗口起始位置。	OFS0.IWDRPSS[1:0]	IWDTCR.RPSS[1:0]
	选择窗口结束位置。	OFS0.IWDRPES[1:0]	IWDTCR.RPES[1:0]
复位输出 / 中断请求输出	选择输出复位或者中断请求。	OFS0.IWDRSTIRQS	IWDRCR.RSTIRQS
计数停止	选择在睡眠模式中停止计数。	OFS0.IWDTSLCSTP	IWDTCSSTPR.SLCSTP

27.4 通过 ELC 进行的链接运行

通过 ELC，IWDT 将中断请求信号用作事件信号，并且对事先设定的模块进行链接运行。在递减计数器发生下溢或者发生刷新错误时输出事件信号。

与寄存器启动模式中的复位中断选择位 (IWDTRCR.RSTIRQS) 或者自动启动模式中的复位中断选择位的设定无关，输出事件信号。另外，在刷新错误标志 (IWDTSR.REFEF) 或者下溢标志 (IWDTSR.UNDF) 为“1”的状态下产生下一个中断源时，也能输出事件信号。

详细内容请参照“18. 事件链接控制器 (ELC)”。

27.5 使用时的注意事项

27.5.1 刷新

在设定刷新时序时，必须考虑 PCLK 和 IWDTCLK 的精度，设定在误差范围内即使周期发生变化也能刷新的值。

28. 串行通信接口 (SCIc、SCIId)

RX210 群具有 7 个独立通道的串行通信接口 (SCI: Serial Communication Interface)。SCI 由 SCIc 模块 (SCI0、SCI1、SCI5、SCI6、SCI8、SCI9) 和 SCIId 模块 (SCI12) 构成。

SCIc (SCI0、SCI1、SCI5、SCI6、SCI8、SCI9) 能进行异步串行通信和时钟同步串行通信。在异步模式中, 能和 Universal Asynchronous Receiver/Transmitter (UART) 或者 Asynchronous Communication Interface Adapter (ACIA) 等标准异步通信 LSI 进行串行通信。另外, 作为异步模式的扩展功能, 支持对应 ISO/IEC 7816-3 (Identification Card) 规格的智能卡 (IC 卡) 接口, 并且支持简易 I²C 总线接口的单主控模式的运行和简易 SPI 接口。

SCIId (SCI12) 除了 SCIc 的功能以外, 还支持由 Start Frame 和 Information Frame 构成的扩展串行通信协议。

28.1 概要

SCIc 的规格、SCIId 的规格和 SCI 各通道功能一览表分别如表 28.1、表 28.2 和表 28.5 所示。

SCI0、SCI1、SCI8 和 SCI9 (SCI5 和 SCI6 除外) 的框图如图 28.1、SCI5 和 SCI6 的框图如图 28.2、SCI12 (SCIId) 的框图如图 28.3 所示。

表 28.1 SCIc 的规格 (1/2)

项目		内容
串行通信方式		<ul style="list-style-type: none"> • 异步 • 时钟同步 • 智能卡接口 • 简易 I²C 总线 • 简易 SPI 接口
传送速度		能通过内部波特率发生器设定任意的位速率。
全双工通信		发送部: 能通过双缓冲结构进行连续的发送。 接收部: 能通过双缓冲结构进行连续的接收。
输入 / 输出引脚		参照表 28.4 ~ 表 28.6。
数据传送		可选择 LSB first 或者 MSB first。
中断源		发送结束、发送数据空、接收数据满、接收错误、开始条件 / 重新开始条件 / 停止条件生成结束 (用于简易 I ² C 模式)
低功耗功能		各通道能设定为模块停止状态。
异步模式	数据长度	7 位或者 8 位
	发送停止位	1 位或者 2 位
	奇偶校验功能	偶校验、奇校验或者无奇偶校验
	接收错误检测功能	奇偶校验错误、溢出错误、帧错误
	硬件流程控制	能控制使用 CTSn 引脚和 RTSn 引脚的发送 / 接收。
	中止的检测	在发生帧错误时, 能通过直接读 RXDn 引脚的电平来检测中止。
	时钟源	可选择内部时钟或者外部时钟。 能输入 TMR 的传送率时钟 (SCI5、SCI6)。
	多处理器通信功能	多个处理器之间的串行通信功能
噪声消除	在 RXDn 引脚的输入线路中内置数字噪声消除器。	
时钟同步模式	数据长度	8 位
	接收错误的检测	溢出错误
	硬件流程控制	能控制使用 CTSn 引脚和 RTSn 引脚的发送 / 接收。

表 28.1 SC1c 的规格 (2/2)

项目		内容
智能卡接口模式	错误处理	如果在接收时检测到奇偶校验错误, 就自动发送错误信号。 如果在发送时接收到错误信号, 就自动重新发送数据。
	数据类型	支持正向协议或者反向协议。
简易 I ² C 模式	通信格式	I ² C 总线格式
	运行模式	主控 (只限于单主控模式的运行)
	传送速度	最大 384kbps
	噪声消除	在 SSCLn 引脚和 SSDAn 引脚的输入线路中内置数字噪声消除器。 能调整噪声消除的宽度。
简易 SPI 模式	数据长度	8 位
	错误的检测	溢出错误
	SS 输入引脚功能	当 SSn# 引脚为 High 电平时, 能将输出引脚置为高阻抗。
	时钟设定	可选择 4 种时钟相位和时钟极性的设定。
事件链接功能		错误 (接收错误和错误信号检测) 事件的输出
		接收数据满事件的输出
		发送数据空事件的输出
		发送结束事件的输出

表 28.2 SC1d 的规格 (1/2)

项目		内容
串行通信方式		<ul style="list-style-type: none"> 异步 时钟同步 智能卡接口 简易 I²C 总线 简易 SPI 接口
传送速度		能通过内部波特率发生器设定任意的位速率。
全双工通信		发送部: 能通过双缓冲结构进行连续的发送。 接收部: 能通过双缓冲结构进行连续的接收。
输入 / 输出引脚		参照表 28.4 ~ 表 28.7。
数据传送		可选择 LSB first 或者 MSB first。
中断源		发送结束、发送数据空、接收数据满、接收错误、开始条件 / 重新开始条件 / 停止条件生成结束 (用于简易 I ² C 模式)
低功耗功能		能设定为模块停止状态。
异步模式	数据长度	7 位或者 8 位
	发送停止位	1 位或者 2 位
	奇偶校验功能	偶校验、奇校验或者无奇偶校验
	接收错误检测功能	奇偶校验错误、溢出错误、帧错误
	硬件流程控制	能控制使用 CTSn 引脚和 RTSn 引脚的发送 / 接收。
	中止的检测	在发生帧错误时, 能通过直接读 RXDn 引脚的电平来检测中止。
	时钟源	可选择内部时钟或者外部时钟。 能输入 TMR 的传送率时钟。
	多处理器通信功能	多个处理器之间的串行通信功能
噪声消除		在 RXDn 引脚的输入线路中内置数字噪声消除器。
时钟同步模式	数据长度	8 位
	接收错误的检测	溢出错误
	硬件流程控制	能控制使用 CTSn 引脚和 RTSn 引脚的发送 / 接收。

表 28.2 SC1d 的规格 (2/2)

项目		内容
智能卡接口模式	错误处理	如果在接收时检测到奇偶校验错误，就自动发送错误信号。 如果在发送时接收到错误信号，就自动重新发送数据。
	数据类型	支持正向协议或者反向协议。
简易 I ² C 模式	通信格式	I ² C 总线格式
	运行模式	主控 (只限于单主控模式的运行)
	传送速度	最大 384kbps
	噪声消除	在 SSCLn 引脚和 SSDAn 引脚的输入线路中内置数字噪声消除器。 能调整噪声消除的宽度。
简易 SPI 模式	数据长度	8 位
	错误的检测	溢出错误
	SS 输入引脚功能	当 SSn# 引脚为 High 电平时，能将输出引脚置为高阻抗。
	时钟设定	可选择 4 种时钟相位和时钟极性的设定。
扩展串行模式	Start Frame 发送	<ul style="list-style-type: none"> 能输出 Break Field Low width，有输出结束中断功能。 有总线冲突检测功能和检测中断功能。
	Start Frame 接收	<ul style="list-style-type: none"> 能检测 Break Field Low width，有检测结束中断功能。 有 Control Field 0 和 Control Field 1 的数据比较 / 相同中断功能。 能在 Control Field 1 中设定主 / 次 2 种比较数据。 能在 Control Field 1 中设定优先级中断位。 也支持没有 Break Field 的 Start Frame。 也支持没有 Control Field 0 的 Start Frame。 有位数率测量功能
	输入 / 输出控制功能	<ul style="list-style-type: none"> 可选择 TXDX12/RXDX12 信号的极性。 能给 RXDX12 信号设定数字滤波器功能。 能进行兼用 RXDX12 引脚和 TXDX12 引脚的半双工通信。 可选择 RXDX12 引脚接收数据的采样时序。 当扩展串行模式控制部为 OFF 时，能将 RXDX12 接收信号直接输出到 SC1c。
	定时器功能	<ul style="list-style-type: none"> 能用作重加载定时器功能。
事件链接功能		错误 (接收错误和错误信号检测) 事件的输出
		接收数据满事件的输出
		发送数据空事件的输出
		发送结束事件的输出

表 28.3 SCI 通道功能一览表

项目	SCI0、SCI1、SCI8、SCI9	SCI5	SCI6	SCI12
异步模式	○	○	○	○
时钟同步模式	○	○	○	○
智能卡接口模式	○	○	○	○
简易 I ² C 模式	○	○	○	○
简易 SPI 模式	○	○	○	○
扩展串行模式	—	—	—	○
TMR 时钟输入	—	○	○	○
事件链接功能	—	○	—	—

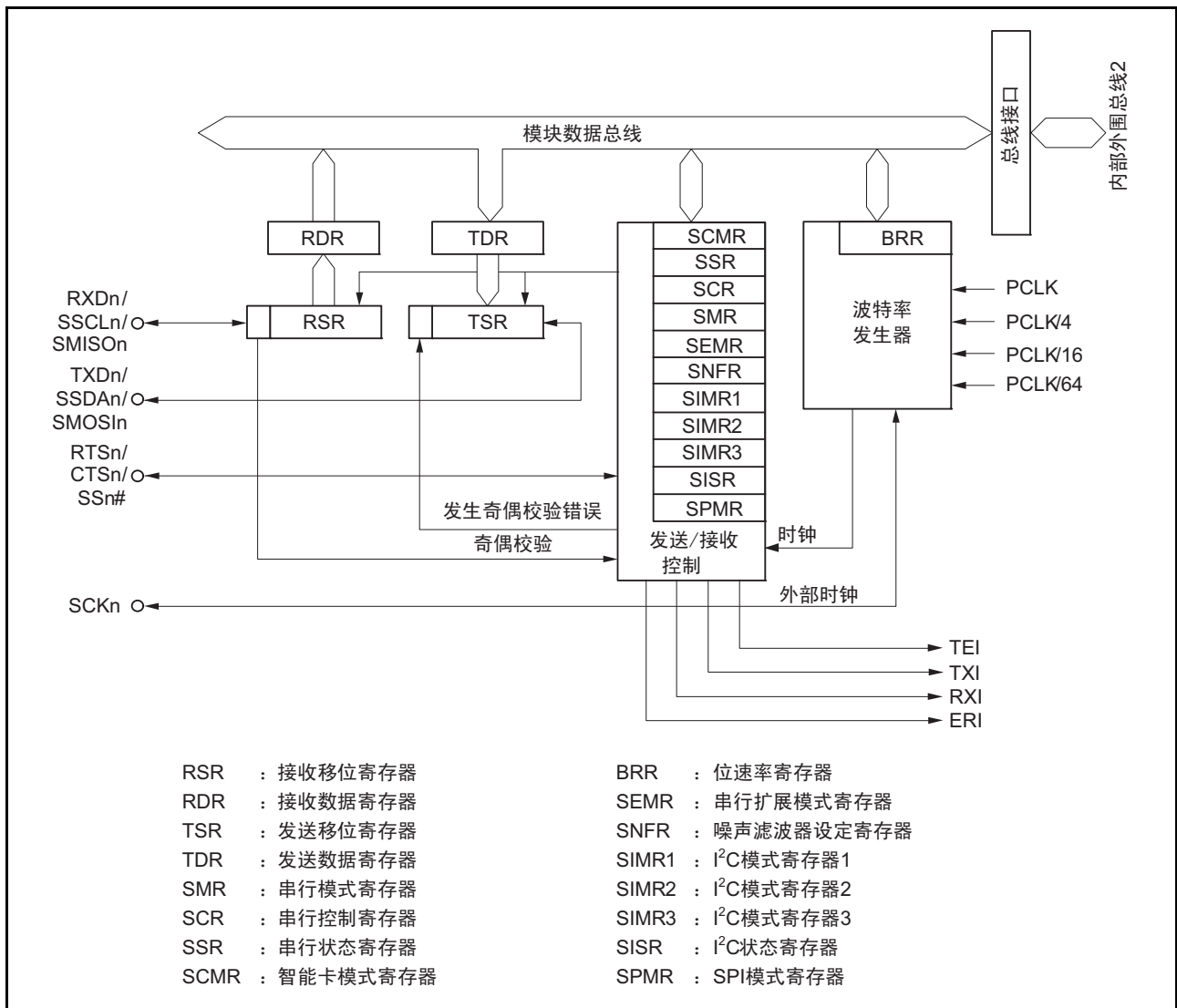


图 28.1 SCI0、SCI1、SCI8 和 SCI9 的框图

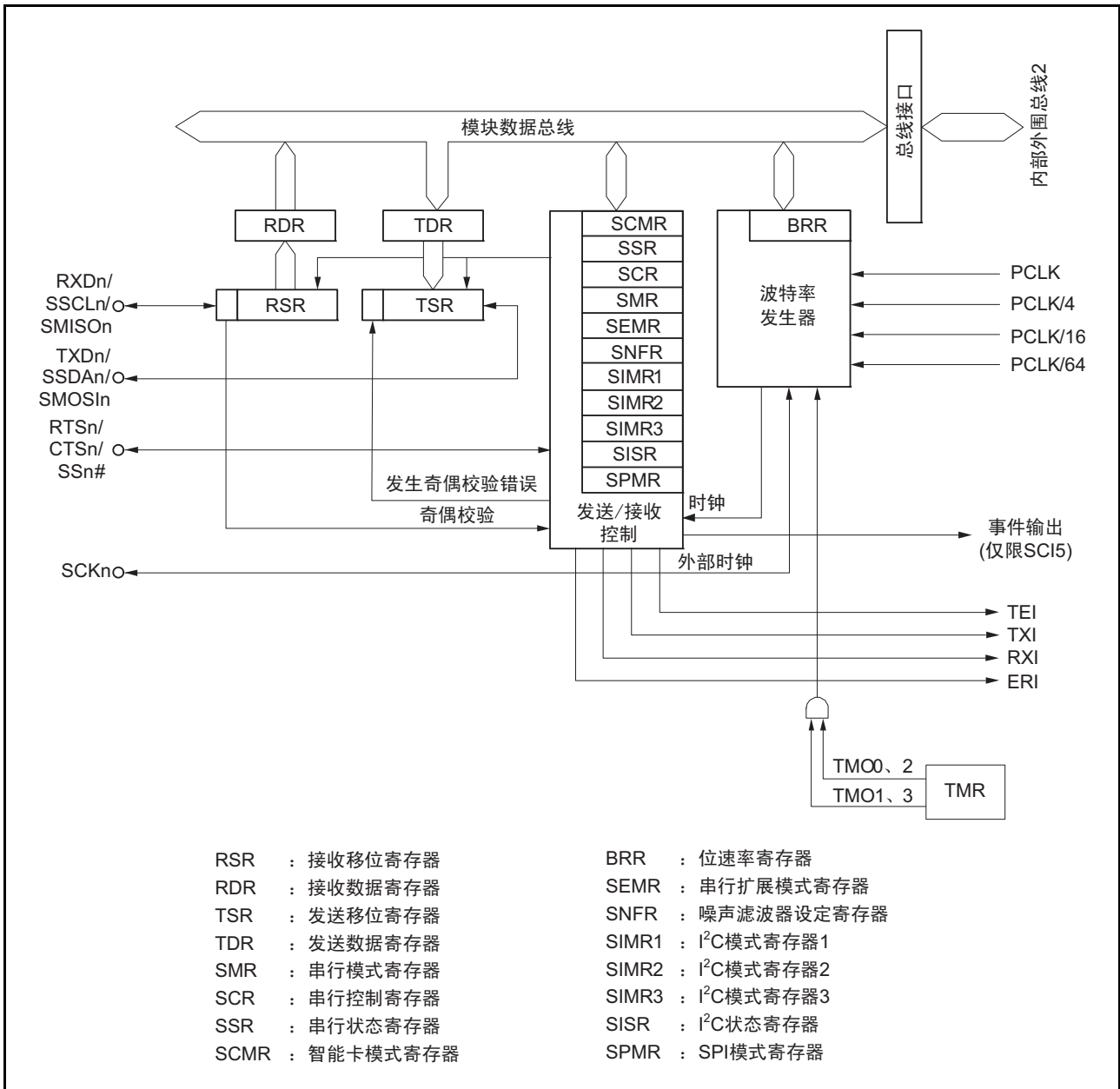


图 28.2 SCI5 和 SCI6 的框图

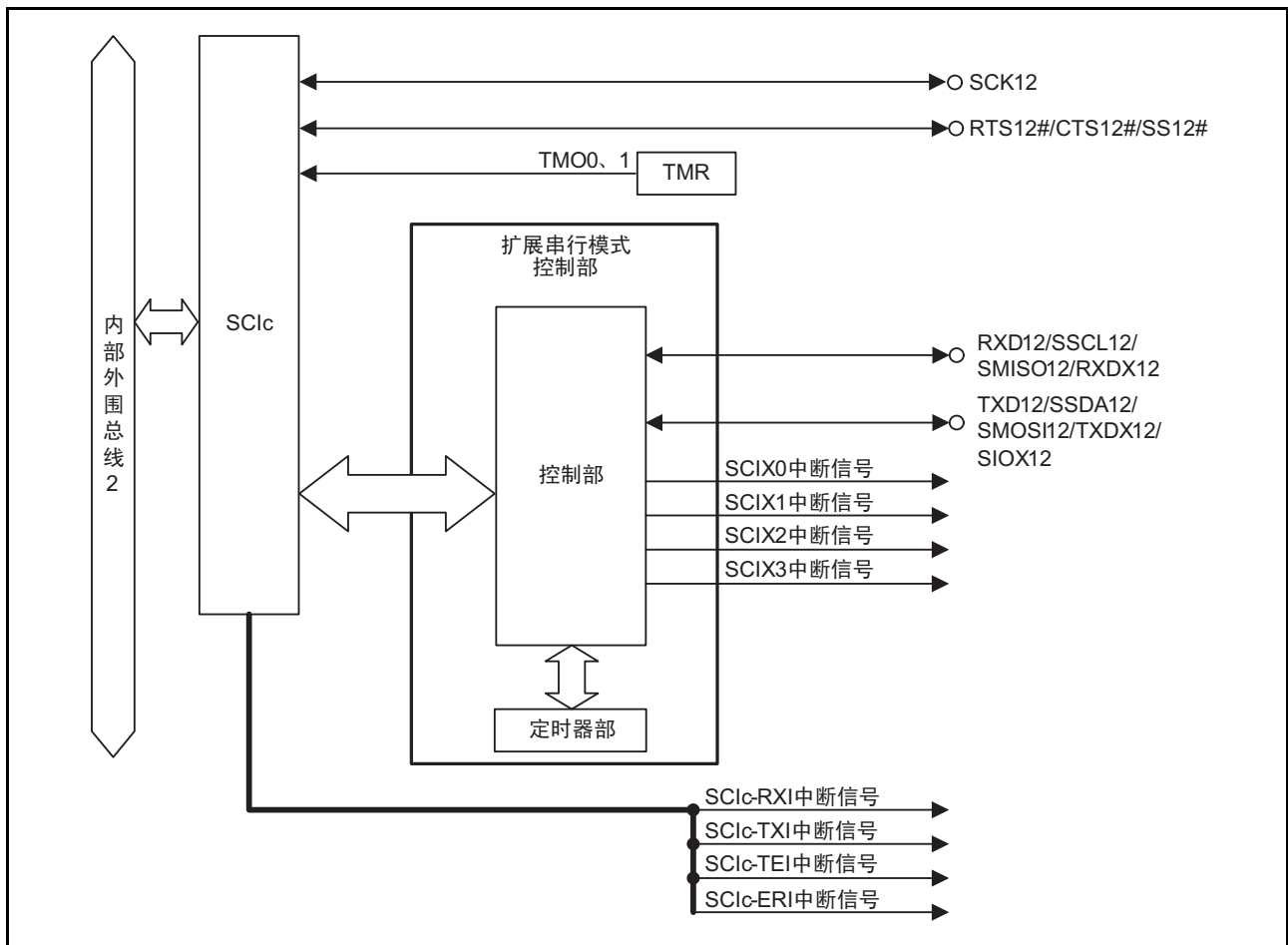


图 28.3 SCI12 (SCId) 的框图

各模式中的 SCI 使用的输入 / 输出引脚如表 28.4 ~ 表 28.7 所示。

表 28.4 SCI 的输入 / 输出引脚 (异步 / 时钟同步模式) (1/2)

通道	引脚名	输入 / 输出	功能
SCI0	SCK0	输入 / 输出	SCI0 的时钟输入 / 输出引脚
	RXD0	输入	SCI0 的接收数据输入引脚
	TXD0	输出	SCI0 的发送数据输出引脚
	CTS0#/RTS0#	输入 / 输出	SCI0 发送 / 接收开始控制的输入 / 输出引脚
SCI1	SCK1	输入 / 输出	SCI1 的时钟输入 / 输出引脚
	RXD1	输入	SCI1 的接收数据输入引脚
	TXD1	输出	SCI1 的发送数据输出引脚
	CTS1#/RTS1#	输入 / 输出	SCI1 发送 / 接收开始控制的输入 / 输出引脚
SCI5	SCK5	输入 / 输出	SCI5 的时钟输入 / 输出引脚
	RXD5	输入	SCI5 的接收数据输入引脚
	TXD5	输出	SCI5 的发送数据输出引脚
	CTS5#/RTS5#	输入 / 输出	SCI5 发送 / 接收开始控制的输入 / 输出引脚

表 28.4 SCI 的输入 / 输出引脚 (异步 / 时钟同步模式) (2/2)

通道	引脚名	输入 / 输出	功能
SCI6	SCK6	输入 / 输出	SCI6 的时钟输入 / 输出引脚
	RXD6	输入	SCI6 的接收数据输入引脚
	TXD6	输出	SCI6 的发送数据输出引脚
	CTS6#/RTS6#	输入 / 输出	SCI6 发送 / 接收开始控制的输入 / 输出引脚
SCI8	SCK8	输入 / 输出	SCI8 的时钟输入 / 输出引脚
	RXD8	输入	SCI8 的接收数据输入引脚
	TXD8	输出	SCI8 的发送数据输出引脚
	CTS8#/RTS8#	输入 / 输出	SCI8 发送 / 接收开始控制的输入 / 输出引脚
SCI9	SCK9	输入 / 输出	SCI9 的时钟输入 / 输出引脚
	RXD9	输入	SCI9 的接收数据输入引脚
	TXD9	输出	SCI9 的发送数据输出引脚
	CTS9#/RTS9#	输入 / 输出	SCI9 发送 / 接收开始控制的输入 / 输出引脚
SCI12	SCK12	输入 / 输出	SCI12 的时钟输入 / 输出引脚
	RXD12	输入	SCI12 的接收数据输入引脚
	TXD12	输出	SCI12 的发送数据输出引脚
	CTS12#/RTS12#	输入 / 输出	SCI12 发送 / 接收开始控制的输入 / 输出引脚

表 28.5 SCI 的输入 / 输出引脚 (简易 I²C 模式)

通道	引脚名	输入 / 输出	功能
SCI0	SSCL0	输入 / 输出	SCI0 的 I ² C 时钟输入 / 输出引脚
	SSDA0	输入 / 输出	SCI0 的 I ² C 数据输入 / 输出引脚
SCI1	SSCL1	输入 / 输出	SCI1 的 I ² C 时钟输入 / 输出引脚
	SSDA1	输入 / 输出	SCI1 的 I ² C 数据输入 / 输出引脚
SCI5	SSCL5	输入 / 输出	SCI5 的 I ² C 时钟输入 / 输出引脚
	SSDA5	输入 / 输出	SCI5 的 I ² C 数据输入 / 输出引脚
SCI6	SSCL6	输入 / 输出	SCI6 的 I ² C 时钟输入 / 输出引脚
	SSDA6	输入 / 输出	SCI6 的 I ² C 数据输入 / 输出引脚
SCI8	SSCL8	输入 / 输出	SCI8 的 I ² C 时钟输入 / 输出引脚
	SSDA8	输入 / 输出	SCI8 的 I ² C 数据输入 / 输出引脚
SCI9	SSCL9	输入 / 输出	SCI9 的 I ² C 时钟输入 / 输出引脚
	SSDA9	输入 / 输出	SCI9 的 I ² C 数据输入 / 输出引脚
SCI12	SSCL12	输入 / 输出	SCI12 的 I ² C 时钟输入 / 输出引脚
	SSDA12	输入 / 输出	SCI12 的 I ² C 数据输入 / 输出引脚

表 28.6 SCI 的输入 / 输出引脚 (简易 SPI 模式)

通道	引脚名	输入 / 输出	功能
SCI0	SCK0	输入 / 输出	SCI0 的时钟输入 / 输出引脚
	SMISO0	输入 / 输出	SCI0 的从属发送数据输入 / 输出引脚
	SMOSI0	输入 / 输出	SCI0 的主控发送数据输入 / 输出引脚
	SS0#	输入	SCI0 的片选输入引脚
SCI1	SCK1	输入 / 输出	SCI1 的时钟输入 / 输出引脚
	SMISO1	输入 / 输出	SCI1 的从属发送数据输入 / 输出引脚
	SMOSI1	输入 / 输出	SCI1 的主控发送数据输入 / 输出引脚
	SS1#	输入	SCI1 的片选输入引脚
SCI5	SCK5	输入 / 输出	SCI5 的时钟输入 / 输出引脚
	SMISO5	输入 / 输出	SCI5 的从属发送数据输入 / 输出引脚
	SMOSI5	输入 / 输出	SCI5 的主控发送数据输入 / 输出引脚
	SS5#	输入	SCI5 的片选输入引脚
SCI6	SCK6	输入 / 输出	SCI6 的时钟输入 / 输出引脚
	SMISO6	输入 / 输出	SCI6 的从属发送数据输入 / 输出引脚
	SMOSI6	输入 / 输出	SCI6 的主控发送数据输入 / 输出引脚
	SS6#	输入	SCI6 的片选输入引脚
SCI8	SCK8	输入 / 输出	SCI8 的时钟输入 / 输出引脚
	SMISO8	输入 / 输出	SCI8 的从属发送数据输入 / 输出引脚
	SMOSI8	输入 / 输出	SCI8 的主控发送数据输入 / 输出引脚
	SS8#	输入	SCI8 的片选输入引脚
SCI9	SCK9	输入 / 输出	SCI9 的时钟输入 / 输出引脚
	SMISO9	输入 / 输出	SCI9 的从属发送数据输入 / 输出引脚
	SMOSI9	输入 / 输出	SCI9 的主控发送数据输入 / 输出引脚
	SS9#	输入	SCI9 的片选输入引脚
SCI12	SCK12	输入 / 输出	SCI12 的时钟输入 / 输出引脚
	SMISO12	输入 / 输出	SCI12 的从属发送数据输入 / 输出引脚
	SMOSI12	输入 / 输出	SCI12 的主控发送数据输入 / 输出引脚
	SS12#	输入	SCI12 的片选输入引脚

表 28.7 SCI 的输入 / 输出引脚 (扩展串行模式)

通道	引脚名	输入 / 输出	功能
SCI12	RDX12	输入	SCI12 的接收数据输入引脚
	TXDX12	输出	SCI12 的发送数据输出引脚
	SIOX12	输入 / 输出	SCI12 发送 / 接收数据的输入 / 输出引脚

28.2 寄存器说明

28.2.1 接收移位寄存器 (RSR)

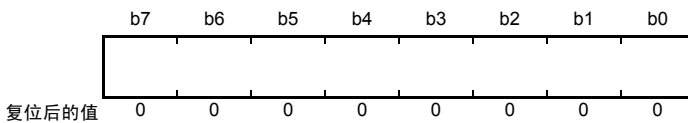
RSR 寄存器是将 RXDn 引脚输入的串行数据转换为并行数据的接收移位寄存器。

如果接收 1 帧的数据，数据就自动传送到 RDR 寄存器。

CPU 不能直接存取 RSR 寄存器。

28.2.2 接收数据寄存器 (RDR)

地址 SCI0.RDR 0008 A005h、SCI1.RDR 0008 A025h、SCI5.RDR 0008 A0A5h、SCI6.RDR 0008 A0C5h、SCI8.RDR 0008 A105h、
SCI9.RDR 0008 A125h、SCI12.RDR 0008 B305h



RDR 寄存器是保存接收数据的 8 位寄存器。

如果接收 1 帧的数据，就将接收数据从 RSR 寄存器传送到此寄存器并且 RSR 寄存器变为能接收下一个数据的状态。

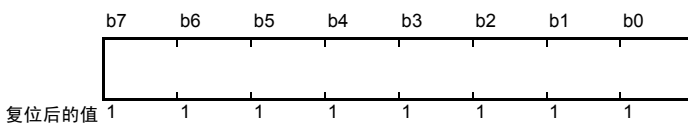
因为 RSR 寄存器和 RDR 寄存器是双缓冲结构，所以能连续接收。

在产生接收数据满中断 (RXI) 请求时，只能读 1 次 RDR 寄存器。必须注意：如果不从 RDR 读接收数据而接收下一帧的数据，就会产生溢出错误。

CPU 不能写 RDR 寄存器。

28.2.3 发送数据寄存器 (TDR)

地址 SCI0.TDR 0008 A003h、SCI1.TDR 0008 A023h、SCI5.TDR 0008 A0A3h、SCI6.TDR 0008 A0C3h、SCI8.TDR 0008 A103h、
SCI9.TDR 0008 A123h、SCI12.TDR 0008 B303h



TDR 寄存器是保存发送数据的 8 位寄存器。

如果检测到 TSR 寄存器为空，就将写在 TDR 寄存器的发送数据传送到 TSR 寄存器，开始发送。

因为 TDR 寄存器和 TSR 寄存器为双缓冲结构，所以能连续发送。如果在发送了 1 帧的数据时将下一个发送数据写到 TDR 寄存器，就将此数据传送到 TSR 寄存器，继续发送。

CPU 能随时读写 TDR 寄存器。在产生发送数据空中断 (TXI) 请求时，只能给 TDR 寄存器写 1 次发送数据。

28.2.4 发送移位寄存器 (TSR)

TSR 寄存器是发送串行数据的移位寄存器。

写在 TDR 寄存器的发送数据自动传送到 TSR 寄存器，通过将数据发送到 TXDn 引脚进行串行数据的发送。

CPU 不能直接存取 TSR 寄存器。

28.2.5 串行模式寄存器 (SMR)

注. 在串行通信接口模式和智能卡接口模式中，SMR 寄存器的部分位的功能不同。

(1) 串行通信接口模式 (SCMR.SMIF 位 =0)

地址 SC10.SMR 0008 A000h、SC11.SMR 0008 A020h、SC15.SMR 0008 A0A0h、SC16.SMR 0008 A0C0h、SC18.SMR 0008 A100h、SC19.SMR 0008 A120h、SC112.SMR 0008 B300h

b7	b6	b5	b4	b3	b2	b1	b0
CM	CHR	PE	PM	STOP	MP	CKS[1:0]	
复位后的值	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b1-b0	CKS[1:0]	时钟选择位	b1 b0 0 0: PCLK 时钟 (n=0) (注1) 0 1: PCLK/4 时钟 (n=1) (注1) 1 0: PCLK/16 时钟 (n=2) (注1) 1 1: PCLK/64 时钟 (n=3) (注1)	R/W (注4)
b2	MP	多处理器模式位	(只在异步模式中有效) 0: 禁止多处理器通信功能 1: 允许多处理器通信功能	R/W (注4)
b3	STOP	停止位长度位	(只在异步模式中有效) 0: 1 个停止位 1: 2 个停止位	R/W (注4)
b4	PM	奇偶校验模式位	(只在异步模式中并且 PE 位为“1”时有效) 0: 偶校验 1: 奇校验	R/W (注4)
b5	PE	奇偶校验允许位	(只在异步模式中有效) • 发送时 0: 无奇偶校验位 1: 附加奇偶校验位 • 接收时 0: 不进行奇偶校验 1: 进行奇偶校验	R/W (注4)
b6	CHR	字符长度位	(只在异步模式中有效) 0: 数据长度为 8 位 (注2) 1: 数据长度为 7 位 (注3)	R/W (注4)
b7	CM	通信模式位	0: 异步通信模式 1: 时钟同步通信模式	R/W (注4)

注 1. n 为设定值的 10 进制数，表示“28.2.9 位速率寄存器 (BRR)”中的 n 的值。

注 2. 在时钟同步模式中，设定无效，数据长度固定为 8 位。

注 3. 固定为 LSB first，在发送时不发送 TDR 寄存器的 MSB (b7)。

注 4. 只能在 SCR.TE 位和 SCR.RE 位都为“0” (禁止串行发送和串行接收) 时写这些位。

CKS[1:0] 位 (时钟选择位)

这些位选择内部波特率发生器的时钟源。

有关 CKS[1:0] 位的设定值和波特率的关系, 请参照“28.2.9 位速率寄存器 (BRR)”。

MP 位 (多处理器模式位)

此位选择禁止或者允许多处理器通信功能。在多处理器模式中, PE 位和 PM 位的设定无效。

STOP 位 (停止位长度位)

此位选择发送数据的停止位长。

在接收时, 与此位的设定无关, 只检查停止位的第 1 位, 当第 2 位为“0”时, 将第 2 位视为下一个发送帧的起始位。

PM 位 (奇偶校验模式位)

此位选择发送和接收时的奇偶校验 (偶校验或者奇校验)。

在多处理器模式中, PM 位的设定无效。

PE 位 (奇偶校验允许位)

当 PE 位为“1”时, 在发送时附加奇偶校验位, 在接收时进行奇偶校验。

与 PE 位的设定无关, 不在多处理器格式中附加奇偶校验位, 也不进行奇偶校验。

CHR 位 (字符长度位)

此位选择发送 / 接收数据的数据长度。

在时钟同步模式中, 数据长度固定为 8 位。

CM 位 (通信模式位)

此位选择异步模式或者时钟同步模式。

(2) 智能卡接口模式 (SCMR.SMIF 位 =1)

地址 SCI0.SMR 0008 A000h、SCI1.SMR 0008 A020h、SCI5.SMR 0008 A0A0h、SCI6.SMR 0008 A0C0h、SCI8.SMR 0008 A100h、SCI9.SMR 0008 A120h、SCI12.SMR 0008 B300h

b7	b6	b5	b4	b3	b2	b1	b0
GM	BLK	PE	PM	BCP[1:0]		CKS[1:0]	
复位后的值	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b1-b0	CKS[1:0]	时钟选择位	b1 b0 0 0: PCLK 时钟 (n=0) (注1) 0 1: PCLK/4 时钟 (n=1) (注1) 1 0: PCLK/16 时钟 (n=2) (注1) 1 1: PCLK/64 时钟 (n=3) (注1)	R/W (注3)
b3-b2	BCP[1:0]	基本时钟脉冲位	通过和 SCMR.BCP2 位组合进行选择。 SCMR.BCP2 位和 SMR.BCP[1:0] 位的设定值 BCP2 b3 b2 0 0 0: 93 个时钟 (S=93) (注2) 0 0 1: 128 个时钟 (S=128) (注2) 0 1 0: 186 个时钟 (S=186) (注2) 0 1 1: 512 个时钟 (S=512) (注2) 1 0 0: 32 个时钟 (S=32) (注2) (初始值) 1 0 1: 64 个时钟 (S=64) (注2) 1 1 0: 372 个时钟 (S=372) (注2) 1 1 1: 256 个时钟 (S=256) (注2)	R/W (注3)
b4	PM	奇偶校验模式位	(只在异步模式中并且 PE 位为“1”时有效) 0: 偶校验 1: 奇校验	R/W (注3)
b5	PE	奇偶校验允许位	当 PE 位为“1”时, 在发送时附加奇偶校验位, 在接收时进行奇偶校验。在智能卡接口模式中, 必须将 PE 位置“1”。	R/W (注3)
b6	BLK	块传送模式位	0: 在正常模式中运行 1: 在块传送模式中运行	R/W (注3)
b7	GM	GSM 模式位	0: 在正常模式中运行 1: 在 GSM 模式中运行	R/W (注3)

注 1. n 为设定值的 10 进制数, 表示“28.2.9 位速率寄存器 (BRR)”中的 n 的值。

注 2. S 表示“28.2.9 位速率寄存器 (BRR)”中的 S 的值。

注 3. 只能在 SCR.TE 位和 SCR.RE 位都为“0” (禁止串行发送和串行接收) 时写这些位。

CKS[1:0] 位 (时钟选择位)

这些位选择内部波特率发生器的时钟源。

有关 CKS[1:0] 位的设定值和波特率的关系, 请参照“28.2.9 位速率寄存器 (BRR)”。

BCP[1:0] 位 (基本时钟脉冲位)

这些位选择智能卡接口模式中 1 位传送时间的基本时钟数。

通过和 SCMR.BCP2 位组合进行选择。

详细内容请参照“28.6.4 接收数据的采样时序和接收容限”。

PM 位 (奇偶校验模式位)

此位选择发送和接收时的奇偶校验 (偶校验或者奇校验)。

有关此位在智能卡接口模式中的使用方法, 请参照“28.6.2 数据格式 (块传送模式除外)”。

PE 位 (奇偶校验允许位)

必须将 PE 位置“1”。

在发送时附加奇偶校验位, 在接收时进行奇偶校验。

BLK (块传送模式位)

如果将 BLK 位置“1”, 就以块传送模式运行。

有关块传送模式, 请参照“28.6.3 块传送模式”。

GM 位 (GSM 模式位)

如果将 GM 位置“1”, 就以 GSM 模式运行。

在 GSM 模式中, SSR.TEND 标志的置位时序为开始发送后的 11.0etu (etu: Elementary Time Unit, 1 位传送时间), 并且追加时钟输出控制功能, 详细内容请参照“28.6.6 串行数据的发送 (块传送模式除外)”和“28.6.8 时钟的输出控制”。

28.2.6 串行控制寄存器 (SCR)

注. 在串行通信接口模式和智能卡接口模式中, SCR 寄存器的部分位的功能不同。

(1) 串行通信接口模式 (SCMR.SMIF 位 = 0)

地址 SCI0.SCR 0008 A002h、SCI1.SCR 0008 A022h、SCI5.SCR 0008 A0A2h、SCI6.SCR 0008 A0C2h、SCI8.SCR 0008 A102h、SCI9.SCR 0008 A122h、SCI12.SCR 0008 B302h

b7	b6	b5	b4	b3	b2	b1	b0
TIE	RIE	TE	RE	MPIE	TEIE	CKE[1:0]	
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b1-b0	CKE[1:0]	时钟允许位	<ul style="list-style-type: none"> • SCI0、SCI1、SCI8 和 SCI9 的情况 (异步模式) b1 b0 0 0: 内部波特率发生器 根据 I/O 端口的设定, SCKn 引脚能用作输入 / 输出端口。 0 1: 内部波特率发生器 从 SCKn 引脚输出频率为位速率的时钟。 1 x: 外部时钟 必须从 SCKn 引脚输入频率为 16 倍位速率的时钟。在 SEMR.ABCS 位为“1”时, 必须输入 8 倍频率的时钟。 (时钟同步模式) b1 b0 0 x: 内部时钟 SCKn 引脚为时钟的输出引脚。 1 x: 外部时钟 SCKn 引脚为时钟的输入引脚。	R/W (注 1)
b1-b0	CKE[1:0]	时钟允许位	<ul style="list-style-type: none"> • SCI5、SCI6 和 SCI12 的情况 (异步模式) b1 b0 0 0: 内部波特率发生器 根据 I/O 端口的设定, SCKn 引脚能用作输入 / 输出端口。 0 1: 内部波特率发生器 从 SCKn 引脚输出频率为位速率的时钟。 1 x: 外部时钟或者 TMR 时钟 <ul style="list-style-type: none"> • 在使用外部时钟的情况下, 必须从 SCKn 引脚输入频率为 16 倍位速率的时钟。在 SEMR.ABCS 位为“1”时, 必须输入 8 倍频率的时钟。 • 能使用 TMR 时钟。 (时钟同步模式) b1 b0 0 x: 内部时钟 SCKn 引脚为时钟的输出引脚。 1 x: 外部时钟 SCKn 引脚为时钟的输入引脚。	R/W (注 1)

位	符号	位名	功能	R/W
b2	TEIE	发送结束中断允许位	0: 禁止 TEI 中断请求 1: 允许 TEI 中断请求	R/W
b3	MPIE	多处理器中断允许位	(在异步模式中 SMR.MP 位为“1”时有效) 0: 通常的接收运行 1: 跳读多处理器位为“0”的接收数据, 禁止 SSR. ORER 和 SSR.FER 各状态标志的置位 (“1”)。如果接收到多处理器位是“1”的数据, 就自动清除 (“0”) MPIE 位, 返回到通常的接收运行。	R/W
b4	RE	接收允许位	0: 禁止串行接收 1: 允许串行接收	R/W (注2)
b5	TE	发送允许位	0: 禁止串行发送 1: 允许串行发送	R/W (注2)
b6	RIE	接收中断允许位	0: 禁止 RXI 中断请求和 ERI 中断请求 1: 允许 RXI 中断请求和 ERI 中断请求	R/W
b7	TIE	发送中断允许位	0: 禁止 TXI 中断请求 1: 允许 TXI 中断请求	R/W

x: Don't care

注 1. 只能在 TE 位和 RE 位都为“0”时写这些位。

注 2. 如果 SMR.CM 位为“1”, 就只能在 TE 位和 RE 位都为“0”时写“1”。一旦将 TE 位或者 RE 位置“1”, 就只能给 TE 位和 RE 位写“0”。如果 SMR.CM 位为“0”, 就能在任意时候写此位。

CKE[1:0] 位 (时钟允许位)

这些位选择时钟源和 SCK_n 引脚的功能。

通过和 SEMR.ACS0 位的组合设定内部 TMR 时钟。

TEIE 位 (发送结束中断允许位)

此位允许或者禁止 TEI 中断请求。

能通过将 TEIE 位置“0”来禁止 TEI 中断请求。

在简易 I²C 模式 (SIMR1.IICM=1) 中, 将开始 / 重新开始 / 停止条件生成结束中断 (STI 中断) 分配给 TEI 中断。此时, 也能通过 TEIE 位允许或者禁止 STI 中断请求。

MPIE 位 (多处理器中断允许位)

如果将 MPIE 位置“1”, 就跳读多处理器位为“0”的接收数据, 并且 SSR.ORER 和 SSR.FER 的各状态标志不变为“1”。如果接收多处理器位为“1”的数据, MPIE 位就自动清除并且返回到通常的接收运行。详细内容请参照“28.4 多处理器通信功能”。

在接收包括 SSR.MPB 位为“0”的接收数据时, 不将接收数据从 RSR 寄存器传送到 RDR 寄存器, 也不检测接收错误并且不将 ORER 和 FER 的各标志置位 (“1”)。

如果接收包括 MPB 位为“1”的接收数据, 就将 MPB 位置“1”, MPIE 位自动变为“0”, 并且允许 RXI 中断请求和 ERI 中断请求 (SCR 的 RIE 位为“1”的情况) 以及允许 ORER 和 FER 各标志的置位 (“1”)。

在不使用多处理器通信功能时, 不能给 MPIE 位写“0”。

RE 位 (接收允许位)

此位允许或者禁止串行接收。

在将 RE 位置“1”后,如果在异步模式中检测到起始位,或者在时钟同步模式中检测到同步时钟输入,就开始串行接收。必须在将 RE 位置“1”前设定 SMR 寄存器,决定接收格式。

即使通过将 RE 位置“0”来停止接收,SSR.ORER 标志、SSR.FER 标志和 SSR.PER 标志也不受影响而保持原来的状态。

TE 位 (发送允许位)

此位允许或者禁止串行发送。

如果将 TE 位置“1”,就通过给 TDR 寄存器写发送数据,开始串行发送。必须在将 TE 位置“1”前设定 SMR 寄存器,决定发送格式。

RIE 位 (接收中断允许位)

此位允许或者禁止 RXI 和 ERI 的中断请求。

能通过将 RIE 位置“0”来禁止 RXI 中断请求。

通过在从 SSR.ORER 标志、SSR.FER 标志和 SSR.PER 标志读“1”后将这些标志位置“0”,或者将 RIE 位置“0”,解除 ERI 中断请求。

TIE 位 (发送中断允许位)

此位允许或者禁止通知 TXI 中断请求。

能通过将 TIE 位置“0”来禁止 TXI 中断请求。

(2) 智能卡接口模式 (SCMR.SMIF 位 =1)

地址 SCI0.SCR 0008 A002h、SCI1.SCR 0008 A022h、SCI5.SCR 0008 A0A2h、SCI6.SCR 0008 A0C2h、SCI8.SCR 0008 A102h、SCI9.SCR 0008 A122h、SCI12.SCR 0008 B302h

b7	b6	b5	b4	b3	b2	b1	b0
TIE	RIE	TE	RE	MPIE	TEIE	CKE[1:0]	
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b1-b0	CKE[1:0]	时钟允许位	<ul style="list-style-type: none"> SMR.GM 位为“0”的情况 b1 b0 0 0: 禁止输出 (根据 I/O 端口的设定, SCKn 引脚能用作输入 / 输出端口) 0 1: 时钟输出 1 x: 不能设定 SMR.GM 位为“1”的情况 b1 b0 0 0: 固定为 Low 电平输出 x 1: 时钟输出 1 0: 固定为 High 电平输出 	R/W (注1)
b2	TEIE	发送结束中断允许位	在智能卡接口模式中, 必须置“0”。	R/W
b3	MPIE	多处理器中断允许位	在智能卡接口模式中, 必须置“0”。	R/W
b4	RE	接收允许位	0: 禁止串行接收 1: 允许串行接收	R/W (注2)
b5	TE	发送允许位	0: 禁止串行发送 1: 允许串行发送	R/W (注2)
b6	RIE	接收中断允许位	0: 禁止 RXI 中断请求和 ERI 中断请求 1: 允许 RXI 中断请求和 ERI 中断请求	R/W
b7	TIE	发送中断允许位	0: 禁止 TXI 中断请求 1: 允许 TXI 中断请求	R/W

x: Don't care

注 1. 只能在 TE 位和 RE 位都为“0”时写这些位。

注 2. 如果 SMR.CM 位为“1”, 就只能在 TE 位和 RE 位都为“0”时写“1”。一旦将 TE 位或者 RE 位置“1”, 就只能给 TE 位和 RE 位写“0”。如果 SMR.CM 位为“0”, 就能在任意时候写此位。

有关各中断源, 请参照“28.11 中断源”。

CKE[1:0] 位 (时钟允许位)

这些位控制 SCKn 引脚的时钟输出。

能在 GSM 模式中动态更改时钟的输出, 详细内容请参照“28.6.8 时钟的输出控制”。

TEIE 位 (发送结束中断允许位)

在智能卡接口模式中, 必须将此位置“0”。

MPIE 位 (多处理器中断允许位)

在智能卡接口模式中, 必须将此位置“0”。

RE 位 (接收允许位)

此位允许或者禁止串行接收。

在将 RE 位置“1”后, 如果检测到起始位, 就开始串行接收。必须在将 RE 位置“1”前设定 SMR 寄存器, 决定接收格式。

即使通过将 RE 位置“0”来停止接收, SSR. ORER 标志、SSR. FER 标志和 SSR. PER 标志也不受影响而保持原来的状态。

TE 位 (发送允许位)

此位允许或者禁止串行发送。

如果将 TE 位置“1”, 就通过给 TDR 寄存器写发送数据, 开始串行发送。必须在将 TE 位置“1”前设定 SMR 寄存器, 决定发送格式。

RIE 位 (接收中断允许位)

此位允许或者禁止 RXI 和 ERI 的中断请求。

能通过将 RIE 位置“0”来禁止 RXI 中断请求。

通过在从 SSR. ORER 标志、SSR. FER 标志和 SSR. PER 标志读“1”后将这些标志位置“0”, 或者将 RIE 位置“0”, 解除 ERI 中断请求。

TIE 位 (发送中断允许位)

此位允许或者禁止通知 TXI 中断请求。

能通过将 TIE 位置“0”来禁止 TXI 中断请求。

28.2.7 串行状态寄存器 (SSR)

注. 在串行通信接口模式和智能卡接口模式中, SSR 寄存器的部分位的功能不同。

(1) 串行通信接口模式 (SCMR.SMIF 位 =0)

地址 SCI0.SSR 0008 A004h、SCI1.SSR 0008 A024h、SCI5.SSR 0008 A0A4h、SCI6.SSR 0008 A0C4h、SCI8.SSR 0008 A104h、SCI9.SSR 0008 A124h、SCI12.SSR 0008 B304h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	ORER	FER	PER	TEND	MPB	MPBT
复位后的值	x	x	0	0	0	1	0	0

x: 不定值

位	符号	位名	功能	R/W
b0	MPBT	多处理器位的传送位	设定给发送帧附加的多处理器位的值。	R/W
b1	MPB	多处理器位	是接收帧中的多处理器位的值。	R
b2	TEND	发送结束标志	0: 正在发送字符 1: 字符发送结束	R
b3	PER	奇偶校验错误标志	0: 未发生奇偶校验错误 1: 发生奇偶校验错误	R(W) (注 1)
b4	FER	帧错误标志	0: 未发生帧错误 1: 发生帧错误	R(W) (注 1)
b5	ORER	溢出错误标志	0: 未发生溢出错误 1: 发生溢出错误	R(W) (注 1)
b7-b6	—	保留位	读取值为不定值, 只能写“1”。	R/W

注 1. 为了清除标志, 只能写“0”。

MPBT 位 (多处理器位的传送位)

此位设定给发送帧附加的多处理器位的值。

MPB 位 (多处理器位)

此位保存接收帧中的多处理器位的值。此位在 SCR.RE 位为“0”时不变。

TEND 标志 (发送结束标志)

此标志表示发送已经结束。

[为“1”的条件]

- 当 SCR.TE 位为“0” (禁止串行发送) 时
在将 SCR.TE 位从“0”置为“1”时, TEND 标志不受影响而保持“1”的状态。
- 在发送字符的最后一位时未更新 TDR 寄存器时

[为“0”的条件]

- 在 SCR.TE 位为“1”的状态下给 TDR 寄存器写发送数据时
在通过给 TDR 寄存器写发送数据来清除 TEND 标志时, 必须读 TEND 标志, 确认 TEND 标志是“0”。

PER 标志 (奇偶校验错误标志)

此标志表示在异步模式中接收数据时发生了奇偶校验错误并且异常结束。

[为“1”的条件]

- 在接收时检测到奇偶校验错误时

将发生奇偶校验错误时的接收数据传送到RDR寄存器，但是不产生RXI中断请求。在PER标志被置“1”的状态下，以后的接收数据不传送到RDR寄存器。

[为“0”的条件]

- 在读到“1”的状态后写“0”（必须在写“0”后确认PER标志已被清除）时
即使将SCR.RE位清“0”（禁止串行接收），PER标志也不受影响而保持原来的状态。

FER 标志 (帧错误标志)

此标志表示在异步模式中接收数据时发生了帧错误并且异常结束。

[为“1”的条件]

- 当停止位为“0”时

在2个停止位模式中，只判断第1个停止位是否为“1”而不检查第2个停止位。将发生帧错误的接收数据传送到RDR寄存器，但是不产生RXI中断请求。在FER标志被置“1”的状态下，以后的接收数据不传送到RDR寄存器。

[为“0”的条件]

- 在读到“1”的状态后写“0”（必须在写“0”后确认FER标志已被清除）时
即使将SCR.RE位置“0”，FER标志也不受影响而保持原来的状态。

ORER 标志 (溢出错误标志)

此标志表示在接收数据时发生了溢出错误并且异常结束。

[为“1”的条件]

- 当不读RDR寄存器的接收数据而接收到下一个数据时

RDR寄存器保持发生溢出错误前的接收数据而放弃后面的接收数据。在ORER标志被置“1”的状态下，不能继续进行以后的串行接收。在时钟同步模式中，也不能继续进行串行发送。

[为“0”的条件]

- 在读到“1”的状态后写“0”（必须在写“0”后确认ORER标志已被清除）时
即使将SCR.RE位置“0”，ORER标志也不受影响而保持原来的状态。

(2) 智能卡接口模式 (SCMR.SMIF 位 = 1)

地址 SCI0.SSR 0008 A004h、SCI1.SSR 0008 A024h、SCI5.SSR 0008 A0A4h、SCI6.SSR 0008 A0C4h、SCI8.SSR 0008 A104h、SCI9.SSR 0008 A124h、SCI12.SSR 0008 B304h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	ORER	ERS	PER	TEND	MPB	MPBT
复位后的值	x	x	0	0	0	1	0	0

x: 不定值

位	符号	位名	功能	R/W
b0	MPBT	多处理器位的传送位	在智能卡接口模式中，必须置“0”。	R/W
b1	MPB	多处理器位	在智能卡接口模式中不使用，必须置“0”。	R
b2	TEND	发送结束标志	0: 正在发送字符 1: 字符发送结束	R
b3	PER	奇偶校验错误标志	0: 未发生奇偶校验错误 1: 发生奇偶校验错误	R/(W) (注 1)
b4	ERS	错误信号状态标志	0: 无错误信号的 Low 电平响应 1: 有错误信号的 Low 电平响应	R/(W) (注 1)
b5	ORER	溢出错误标志	0: 未发生溢出错误 1: 发生溢出错误	R/(W) (注 1)
b7-b6	—	保留位	读取值为不定值，只能写“1”。	R/W

注 1. 为了清除标志，只能写“0”。

MPBT 位 (多处理器位的传送位)

在智能卡接口模式中，必须将此位置“0”。

MPB 位 (多处理器位)

在智能卡接口模式中，不使用此位，必须将此位置“0”。

TEND 标志 (发送结束标志)

在没有接收侧的错误信号响应并且能将下一个发送数据传送到 TDR 寄存器的情况下，此标志变为“1”。

[为“1”的条件]

- 当 SCR.TE 位为“0” (禁止串行发送) 时
在将 SCR.TE 位从“0”置为“1”时，TEND 标志不受影响而保持“1”的状态。
- 在发送 1 字节数据后的一定时间之后 ERS 标志变为“0”并且未更新 TDR 寄存器时
根据寄存器的设定，置位的时序如下：
当 SMR.GM 位和 SMR.BLK 位都为“0”时，在开始发送后的 12.5etu。
当 SMR.GM 位为“0”并且 SMR.BLK 位为“1”时，在开始发送后的 11.5etu。
当 SMR.GM 位为“1”并且 SMR.BLK 位为“0”时，在开始发送后的 11.0etu。
当 SMR.GM 位和 SMR.BLK 位都为“1”时，在开始发送后的 11.0etu。

[为“0”的条件]

- 在 SCR.TE 位为“1”的状态下给 TDR 寄存器写发送数据时

PER 标志 (奇偶校验错误标志)

此标志表示在异步模式中接收数据时发生了奇偶校验错误并且异常结束。

[为“1”的条件]

- 在接收时检测到奇偶校验错误时

将发生奇偶校验错误时的接收数据传送到RDR寄存器，但是不产生RXI中断请求。在PER标志被置“1”的状态下，以后的接收数据不传送到RDR寄存器。

[为“0”的条件]

- 在读到“1”的状态后写“0”（必须在写“0”后确认PER标志已被清除）时
即使将SCR.RE位清“0”（禁止串行接收），PER标志也不受影响而保持原来的状态。

ERS 标志 (错误信号状态标志)

[为“1”的条件]

- 当对错误信号的Low电平进行采样时

[为“0”的条件]

- 在读到“1”的状态后写“0”时

ORER 标志 (溢出错误标志)

此标志表示在接收数据时发生了溢出错误并且异常结束。

[为“1”的条件]

- 当不读RDR寄存器的接收数据而接收到下一个数据时

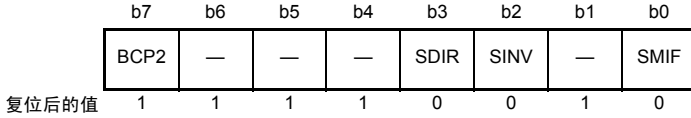
RDR寄存器保持发生溢出错误前的接收数据而放弃后面的接收数据。在ORER标志被置“1”的状态下，不能继续进行以后的串行接收。

[为“0”的条件]

- 在读到“1”的状态后写“0”（必须在写“0”后确认ORER标志已被清除）时
即使将SCR.RE位置“0”，ORER标志也不受影响而保持原来的状态。

28.2.8 智能卡模式寄存器 (SCMR)

地址 SCI0.SCMR 0008 A006h、SCI1.SCMR 0008 A026h、SCI5.SCMR 0008 A0A6h、SCI6.SCMR 0008 A0C6h、SCI8.SCMR 0008 A106h、SCI9.SCMR 0008 A126h、SCI12.SCMR 0008 B306h



位	符号	位名	功能	R/W
b0	SMIF	智能卡接口模式选择位	0: 串行通信接口模式 1: 智能卡接口模式	R/W (注1)
b1	—	保留位	读写值都为“1”。	R/W
b2	SINV	发送 / 接收数据反相位	0: 将 TDR 寄存器的内容直接发送, 并且将接收数据直接保存到 RDR 寄存器。 1: 将 TDR 寄存器的内容取反后发送, 并且将接收数据取反后保存到 RDR 寄存器。	R/W (注1)
b3	SDIR	发送 / 接收数据传送方向位 (注2)	0: 以 LSB first 进行发送和接收 1: 以 MSB first 进行发送和接收	R/W (注1)
b6-b4	—	保留位	读写值都为“1”。	R/W
b7	BCP2	基本时钟脉冲位 2	通过和 SMR.BCP[1:0] 位组合进行选择。 SCMR.BCP2 位和 SMR.BCP[1:0] 位的设定值 BCP2 BCP1 BCP0 0 0 0: 93 个时钟 (S=93) (注3) 0 0 1: 128 个时钟 (S=128) (注3) 0 1 0: 186 个时钟 (S=186) (注3) 0 1 1: 512 个时钟 (S=512) (注3) 1 0 0: 32 个时钟 (S=32) (注3) (初始值) 1 0 1: 64 个时钟 (S=64) (注3) 1 1 0: 372 个时钟 (S=372) (注3) 1 1 1: 256 个时钟 (S=256) (注3)	R/W (注1)

注 1. 只能在 SCR.TE 位和 SCR.RE 位都为“0” (禁止串行发送和串行接收) 时写此位。

注 2. 在选择简易 I²C 模式时, 此位无效。

注 3. S 表示“28.2.9 位速率寄存器 (BRR)”中的 S 的值。

SCMR 寄存器是选择智能卡接口模式以及其格式的寄存器。

SMIF 位 (智能卡接口模式选择位)

在以智能卡接口模式运行时, 将此位置“1”。

在以异步模式或者时钟同步模式运行时, 将此位置“0”。

SINV 位 (发送 / 接收数据反相位)

将发送 / 接收数据的逻辑电平取反。SINV 位不影响奇偶校验位的逻辑电平。如果要将奇偶校验位取反, 就必须将 SMR.PM 位取反。

SDIR 位 (发送 / 接收数据传送方向位)

此位选择串行 / 并行转换的方向，能用于以下模式。

- 异步模式
- 时钟同步模式
- 智能卡接口模式
- 多处理器模式
- 简易 SPI 模式

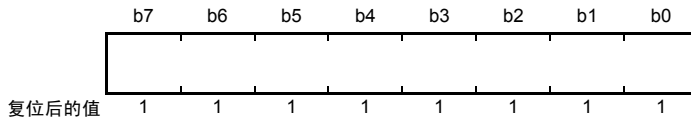
要在简易 I²C 模式中运行时，将此位置“1”。

BCP2 位 (基本时钟脉冲位 2)

在智能卡接口模式中，通过和 SMR.BCP[1:0] 位组合，选择 1 位传送时间的基本时钟数。

28.2.9 位速率寄存器 (BRR)

地址 SCI0.BRR 0008 A001h、SCI1.BRR 0008 A021h、SCI5.BRR 0008 A0A1h、SCI6.BRR 0008 A0C1h、SCI8.BRR 0008 A101h、SCI9.BRR 0008 A121h、SCI12.BRR 0008 B301h



BRR 寄存器是调整位速率的 8 位寄存器。

因为 SCI 各通道的波特率发生器独立运行，所以能设定不同的位速率。在一般的异步模式、多处理器通信、时钟同步模式、智能卡接口模式、简易 SPI 模式和简易 I²C 模式中，BRR 寄存器的设定值 N 和位速率 B 的关系如表 28.8 所示。

BRR 寄存器的初始值为“FFh”。

CPU 能随时读 BRR 寄存器，但是只能在 SCR.TE 位和 SCR.RE 位都为“0”时写此寄存器。

表 28.8 BRR 寄存器的设定值 N 和位速率 B 的关系

模式	SEMR.ABCS位	BRR寄存器的设定值	误差
异步、多处理器通信	0	$N = \frac{PCLK \times 10^6}{64 \times 2^{2n-1} \times B} - 1$	误差(%) = $\left\{ \frac{PCLK \times 10^6}{B \times 64 \times 2^{2n-1} \times (N+1)} - 1 \right\} \times 100$
	1	$N = \frac{PCLK \times 10^6}{32 \times 2^{2n-1} \times B} - 1$	误差(%) = $\left\{ \frac{PCLK \times 10^6}{B \times 32 \times 2^{2n-1} \times (N+1)} - 1 \right\} \times 100$
时钟同步、简易SPI		$N = \frac{PCLK \times 10^6}{8 \times 2^{2n-1} \times B} - 1$	
智能卡接口		$N = \frac{PCLK \times 10^6}{S \times 2^{2n+1} \times B} - 1$	误差(%) = $\left\{ \frac{PCLK \times 10^6}{B \times S \times 2^{2n+1} \times (N+1)} - 1 \right\} \times 100$
简易I ² C ^(注1)		$N = \frac{PCLK \times 10^6}{64 \times 2^{2n-1} \times B} - 1$	

注. B: 位速率 (bps)

N: 波特率发生器的BRR设定值 ($0 \leq N \leq 255$)

PCLK: 工作频率 (MHz)

n和S: 取决于下表中SMR的设定值。

注 1. 必须调整位数率，使简易 I²C 模式中的 SCL 输出的 High/Low 电平宽度满足 I²C 规格。

表 28.9 SCL High/Low 电平宽度的计算式

模式	SCL	计算式 (秒 (s))
I ² C	High 宽度 (min 值)	$(N + 1) \times 4 \times 2^{2n-1} \times 7 \times \frac{1}{PCLK \times 10^6}$
	Low 宽度 (min 值)	$(N + 1) \times 4 \times 2^{2n-1} \times 8 \times \frac{1}{PCLK \times 10^6}$

表 28.10 时钟源的设定

SMR 寄存器的设定值	时钟源	n
CKS[1:0] 位		
0 0	PCLK 时钟	0
0 1	PCLK/4 时钟	1
1 0	PCLK/16 时钟	2
1 1	PCLK/64 时钟	3

表 28.11 智能卡接口模式中的基本时钟设定

SCMR 寄存器的设定值	SMR 寄存器的设定值	基本时钟	S
BCP2 位	BCP[1:0] 位		
0	0 0	93 个时钟	93
0	0 1	128 个时钟	128
0	1 0	186 个时钟	186
0	1 1	512 个时钟	512
1	0 0	32 个时钟	32
1	0 1	64 个时钟	64
1	1 0	372 个时钟	372
1	1 1	256 个时钟	256

一般的异步模式中的 BRR 寄存器值 N 的设定例子如表 28.12 和表 28.14 所示，在各工作频率下能设定的最大位速率如表 28.14 所示；时钟同步模式和简易 SPI 模式中的 BRR 寄存器值 N 的设定例子如表 28.16 所示，智能卡接口模式中的 BRR 寄存器值 N 的设定例子如表 28.16 所示，简易 I²C 模式中的 BRR 寄存器值 N 的设定例子如表 28.16 所示。在智能卡接口模式中，能选择 1 位传送时间的基本时钟数 S。详细内容请参照“28.6.4 接收数据的采样时序和接收容限”。输入外部时钟时的最大位速率如表 28.15 和表 28.17 所示。

在异步模式中，如果将串行扩展模式寄存器 (SEMR) 的异步基本时钟选择位 (ABCS) 置“1”，位速率就变为表 28.12 所示的位速率的 2 倍。

表 28.12 对应位速率的 BRR 设定例子 (异步模式) (1)

位速率 (bps)	工作频率 PCLK (MHz)											
	8			9.8304			10			12		
	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)
110	2	141	0.03	2	174	-0.26	2	177	-0.25	2	212	0.03
150	2	103	0.16	2	127	0.00	2	129	0.16	2	155	0.16
300	1	207	0.16	1	255	0.00	2	64	0.16	2	77	0.16
600	1	103	0.16	1	127	0.00	1	129	0.16	1	155	0.16
1200	0	207	0.16	0	255	0.00	1	64	0.16	1	77	0.16
2400	0	103	0.16	0	127	0.00	0	129	0.16	0	155	0.16
4800	0	51	0.16	0	63	0.00	0	64	0.16	0	77	0.16
9600	0	25	0.16	0	31	0.00	0	32	-1.36	0	38	0.16
19200	0	12	0.16	0	15	0.00	0	15	1.73	0	19	-2.34
31250	0	7	0.00	0	9	-1.70	0	9	0.00	0	11	0.00
38400	—	—	—	0	7	0.00	0	7	1.73	0	9	-2.34

位速率 (bps)	工作频率 PCLK (MHz)								
	12.288			14			16		
	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)
110	2	217	0.08	2	248	-0.17	3	70	0.03
150	2	159	0.00	2	181	0.16	2	207	0.16
300	2	79	0.00	2	90	0.16	2	103	0.16
600	1	159	0.00	1	181	0.16	1	207	0.16
1200	1	79	0.00	1	90	0.16	1	103	0.16
2400	0	159	0.00	0	181	0.16	0	207	0.16
4800	0	79	0.00	0	90	0.16	0	103	0.16
9600	0	39	0.00	0	45	-0.93	0	51	0.16
19200	0	19	0.00	0	22	-0.93	0	25	0.16
31250	0	11	2.40	0	13	0.00	0	15	0.00
38400	0	9	0.00	—	—	—	0	12	0.16

注. 这是 SEMR.ABCS 位为“0”时的例子。
如果将 ABCS 位置“1”，位速率就变为原来的 2 倍。

表 28.13 对应位速率的 BRR 设定例子 (异步模式) (2)

位速率 (bps)	工作频率 PCLK (MHz)											
	17.2032			18			19.6608			20		
	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)
110	3	75	0.48	3	79	-0.12	3	86	0.31	3	88	-0.25
150	2	223	0.00	2	233	0.16	2	255	0.00	3	64	0.16
300	2	111	0.00	2	116	0.16	2	127	0.00	2	129	0.16
600	1	223	0.00	1	233	0.16	1	255	0.00	2	64	0.16
1200	1	111	0.00	1	116	0.16	1	127	0.00	1	129	0.16
2400	0	223	0.00	0	233	0.16	0	255	0.00	1	64	0.16
4800	0	111	0.00	0	116	0.16	0	127	0.00	0	129	0.16
9600	0	55	0.00	0	58	-0.69	0	63	0.00	0	64	0.16
19200	0	27	0.00	0	28	1.02	0	31	0.00	0	32	-1.36
31250	0	16	1.20	0	17	0.00	0	19	-1.70	0	19	0.00
38400	0	13	0.00	0	14	-2.34	0	15	0.00	0	15	1.73

位速率 (bps)	工作频率 PCLK (MHz)					
	25			30		
	n	N	误差 (%)	n	N	误差 (%)
110	3	110	-0.02	3	132	0.13
150	3	80	0.47	3	97	-0.35
300	2	162	-0.15	2	194	0.16
600	2	80	0.47	2	97	-0.35
1200	1	162	-0.15	1	194	0.16
2400	1	80	0.47	1	97	-0.35
4800	0	162	-0.15	0	194	0.16
9600	0	80	0.47	0	97	-0.35
19200	0	40	-0.76	0	48	-0.35
31250	0	24	0.00	0	29	0
38400	0	19	1.73	0	23	1.73

注. 这是 SEMR.ABCS 位为“0”时的例子。
如果将 ABCS 位置“1”，位速率就变为原来的 2 倍。

表 28.14 各工作频率下的最大位速率 (异步模式)

PCLK (MHz)	最大位速率 (bps)	n	N	PCLK (MHz)	最大位速率 (bps)	n	N
8	250000	0	0	18	562500	0	0
9.8304	307200	0	0	19.6608	614400	0	0
10	312500	0	0	20	625000	0	0
12	375000	0	0	25	781250	0	0
12.288	384000	0	0	30	937500	0	0
14	437500	0	0				
16	500000	0	0				
17.2032	537600	0	0				

注. 如果将 SEMR.ABCS 位置“1”，位速率就变为原来的 2 倍。

表 28.15 输入外部时钟时的最大位速率 (异步模式) (1)

PCLK (MHz)	外部输入时钟 (MHz)	最大位速率 (bps)	PCLK (MHz)	外部输入时钟 (MHz)	最大位速率 (bps)
8	2.0000	125000	18	4.5000	281250
9.8304	2.4576	153600	19.6608	4.9152	307200
10	2.5000	156250	20	5.0000	312500
12	3.0000	187500	25	6.2500	390625
12.288	3.0720	192000	30	7.5000	468750
14	3.5000	218750			
16	4.0000	250000			
17.2032	4.3008	268800			

注. 这是 SEMR.ABCS 位为“0”时的例子。

表 28.15 输入外部时钟时的最大位速率 (异步模式) (2)

PCLK (MHz)	外部输入时钟 (MHz)	最大位速率 (bps)	PCLK (MHz)	外部输入时钟 (MHz)	最大位速率 (bps)
8	2.0000	250000	18	4.5000	562500
9.8304	2.4576	307200	19.6608	4.9152	614400
10	2.5000	312500	20	5.0000	625000
12	3.0000	375000	25	6.2500	781250
12.288	3.0720	384000	30	7.5000	937500
14	3.5000	437500			
16	4.0000	500000			
17.2032	4.3008	537600			

注. 这是 SEMR.ABCS 位为“1”时的例子。

表 28.16 对应位速率的 BRR 设定例子 (时钟同步模式、简易 SPI 模式)

位速率 (bps)	工作频率 PCLK (MHz)											
	8		10		16		20		25		30	
	n	N	n	N	n	N	n	N	n	N	n	N
110												
250	3	124	—	—	3	249						
500	2	249	—	—	3	124	—	—			3	233
1k	2	124	—	—	2	249	—	—	3	97	3	116
2.5k	1	199	1	249	2	99	2	124	2	155	2	187
5k	1	99	1	124	1	199	1	249	2	77	2	93
10k	0	199	0	249	1	99	1	124	1	155	1	187
25k	0	79	0	99	0	159	0	199	0	249	1	74
50k	0	39	0	49	0	79	0	99	0	124	0	149
100k	0	19	0	24	0	39	0	49	0	62	0	74
250k	0	7	0	9	0	15	0	19	0	24	0	29
500k	0	3	0	4	0	7	0	9	—	—	0	14
1M	0	1			0	3	0	4	—	—	—	—
2M	0	0 (注 1)	—	—	0	1	—	—	—	—	—	—
2.5M			0	0 (注 1)			0	1	—	—	0	2
4M					0	0 (注 1)	—	—	—	—	—	—
5M							0	0 (注 1)	—	—	—	—
6.25M									0	0 (注 1)	—	—
7.5M											0	0 (注 1)

空栏：不能设定。

—：能设定，但是会出现误差。

注 1. 不能进行连续的发送和接收。

表 28.17 输入外部时钟时的最大位速率 (时钟同步模式、简易 SPI 总线模式)

PCLK (MHz)	外部输入时钟 (MHz)	最大位速率 (bps)	PCLK (MHz)	外部输入时钟 (MHz)	最大位速率 (bps)
8	1.3333	1333333.3	20	3.3333	3333333.3
10	1.6667	1666666.7	25	4.1667	4166666.7
12	2.0000	2000000.0	30	5.0000	5000000.0
14	2.3333	2333333.3			
16	2.6667	2666666.7			
18	3.0000	3000000.0			

表 28.18 对应位速率的 BRR 设定例子 (在智能卡接口模式中, n 为“0”并且 S 为“372”的情况)

位数率 (bps)	PCLK (MHz)	n	N	误差 (%)
9600	7.1424	0	0	0.00
	10.00	0	1	30
	10.7136	0	1	25
	13.00	0	1	8.99
	14.2848	0	1	0.00
	16.00	0	1	12.01
	18.00	0	2	15.99
	20.00	0	2	6.66
	25.00	0	3	12.49
30.00	0	3	5.01	

表 28.19 各工作频率下的最大位速率 (在智能卡接口模式中并且 S 为“372”的情况)

PCLK (MHz)	最大位速率 (bps)	n	N	PCLK (MHz)	最大位速率 (bps)	n	N
10.00	13441	0	0	20.00	26882	0	0
10.7136	14400	0	0	25.00	33602	0	0
13.00	17473	0	0	30.00	40323	0	0
16.00	21505	0	0				
18.00	24194	0	0				

表 28.20 对应位速率的 BRR 设定例子 (简易 I²C 模式)

位速率 (bps)	工作频率 PCLK (MHz)											
	8			10			16			20		
	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)	n	N	误差 (%)
10k	0	24	0.0	0	31	-2.3	1	12	-3.8	1	15	-2.3
25k	0	9	0.0	0	12	-3.8	1	4	0.0	1	6	-10.7
50k	0	4	0.0	0	6	-10.7	1	2	-16.7	1	3	-21.9
100k	0	2	-16.7	0	3	-21.9	0	4	0.0	0	6	-10.7
250k	0	0	0.0	0	1	-37.5	0	1	0.0	0	2	-16.7
350k										0	1	-10.7

位速率 (bps)	工作频率 PCLK (MHz)					
	25			30		
	n	N	误差 (%)	n	N	误差 (%)
10k	1	19	-2.3	1	23	-2.3
25k	1	7	-2.3	1	9	-6.3
50k	1	3	-2.3	1	4	-6.3
100k	1	1	-2.3	1	2	-21.9
250k	0	3	-21.9	0	3	-6.3
350k	0	2	-25.6	0	2	-10.7

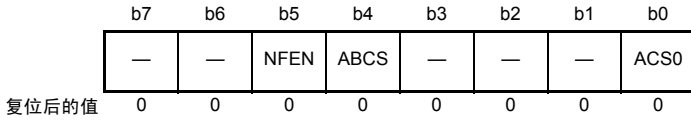
表 28.21 各位速率设定的 SCL High/Low 电平宽度最小值 (简易 I²C 模式)

SCL High/Low 电平宽度 min 值 (μs)	工作频率 PCLK (MHz)											
	8			10			16			20		
	n	N	High/Low 电平宽度	n	N	High/Low 电平宽度	n	N	High/Low 电平宽度	n	N	High/Low 电平宽度
10k	0	24	43.75/50.00	0	31	44.80/51.20	1	12	45.5/52.00	1	15	44.80/51.20
25k	0	9	17.50/20.00	0	12	18.2/20.80	1	4	17.50/20.00	1	6	19.60/22.40
50k	0	4	8.75/10.00	0	6	9.80/11.20	1	2	10.50/12.00	1	3	11.20/12.80
100k	0	2	5.25/6.00	0	3	5.60/6.40	0	4	4.37/5.00	0	6	4.90/5.60
250k	0	0	1.75/2.00	0	1	2.80/3.20	0	1	1.75/2.00	0	2	2.10/2.40
350k										0	1	1.40/1.60

SCL High/Low 电平宽度 min 值 (μs)	工作频率 PCLK (MHz)					
	25			30		
	n	N	High/Low 电平宽度	n	N	High/Low 电平宽度
10k	1	19	44.80/51.20	1	23	44.80/51.20
25k	1	7	17.92/20.48	1	9	18.66/21.33
50k	1	3	8.96/10.24	1	4	9.33/10.66
100k	1	1	4.48/5.12	1	2	5.60/6.40
250k	0	3	2.24/2.56	0	3	1.86/2.13
350k	0	2	1.68/1.92	0	2	1.40/1.60

28.2.10 串行扩展模式寄存器 (SEMR)

地址 SCI0.SEMR 0008 A007h、SCI1.SEMR 0008 A027h、SCI5.SEMR 0008 A0A7h、SCI6.SEMR 0008 A0C7h、SCI8.SEMR 0008 A107h、SCI9.SEMR 0008 A127h、SCI12.SEMR 0008 B307h



位	符号	位名	功能	R/W												
b0	ACS0	异步时钟源选择位	(只在异步模式中有效) 0: 外部时钟的输入 1: TMR 时钟的输入 (只对 SCI5、SCI6 和 SCI12 有效) SCI 通道和比较匹配输出的对应如下所示: <table border="1" style="border-collapse: collapse; margin: 5px 0;"> <thead> <tr> <th>SCI</th> <th>TMR</th> <th>比较匹配输出</th> </tr> </thead> <tbody> <tr> <td>SCI5</td> <td>单元 0</td> <td>TMO0、TMO1</td> </tr> <tr> <td>SCI6</td> <td>单元 1</td> <td>TMO2、TMO3</td> </tr> <tr> <td>SCI12</td> <td>单元 0</td> <td>TMO0、TMO1</td> </tr> </tbody> </table>	SCI	TMR	比较匹配输出	SCI5	单元 0	TMO0、TMO1	SCI6	单元 1	TMO2、TMO3	SCI12	单元 0	TMO0、TMO1	R/W (注 1)
SCI	TMR	比较匹配输出														
SCI5	单元 0	TMO0、TMO1														
SCI6	单元 1	TMO2、TMO3														
SCI12	单元 0	TMO0、TMO1														
b3-b1	—	保留位	读写值都为“0”。	R/W												
b4	ABCS	异步基本时钟选择位	(只在异步模式中有效) 0: 16 个基本时钟周期的时间为 1 位时间的传送率 1: 8 个基本时钟周期的时间为 1 位时间的传送率	R/W (注 1)												
b5	NFEN	数字噪声滤波器功能允许位	(异步模式) 0: RXDn 输入信号的噪声消除功能无效 1: RXDn 输入信号的噪声消除功能有效 (简易 I ² C 模式) 0: SSCLn 和 SSDAn 输入信号的噪声消除功能无效 1: SSCLn 和 SSDAn 输入信号的噪声消除功能有效 在上述以外的模式中, 必须将 NFEN 位置“0”。	R/W (注 1)												
b7-b6	—	保留位	读写值都为“0”。	R/W												

注 1. 只能在 SCR.TE 位和 SCR.RE 位都为“0” (禁止串行发送和串行接收) 时写此位。

SEMR 寄存器选择异步模式中的 1 位时间的时钟。

对于 SCI5、SCI6 和 SCI12, 能将 TMR 单元 0 和单元 1 的 TMO_n (n=0~3) 输出设定为串行传送的基本时钟。

选择 TMR_n (n=0~3) 的 TMO_n 输出时的设定例子如图 28.4 所示。

ACS0 位 (异步时钟源选择位)

此位选择异步模式中的时钟源。

ACS0 位在异步模式 (SMR.CM 位 =0) 中并且输入外部时钟 (SCR.CKE[1:0] 位 =10b、11b) 时有效。能选择外部时钟的输入或者内部 TMR 时钟的输入。

在异步模式以外的模式中, 必须将此位置“0”。

在 SCI5、SCI6 和 SCI12 以外的通道时, 此位为保留位, 只能写“0”。

ABCS 位 (异步基本时钟选择位)

此位选择 1 位时间的基本时钟的脉冲数。

NFEN 位 (数字噪声滤波器功能允许位)

此位选择数字噪声滤波器功能的有效或者无效。

如果置为有效，就在异步模式中消除 $RXDn$ 输入信号的噪声，而在简易 I²C 模式中消除 $SSDAn$ 和 $SSCLn$ 输入信号的噪声。

在上述以外的模式中，必须将 **NFEN** 位置“0”，使数字噪声滤波器功能无效。如果将数字噪声滤波器功能置为无效，输入信号就直接作为内部信号而被传递。

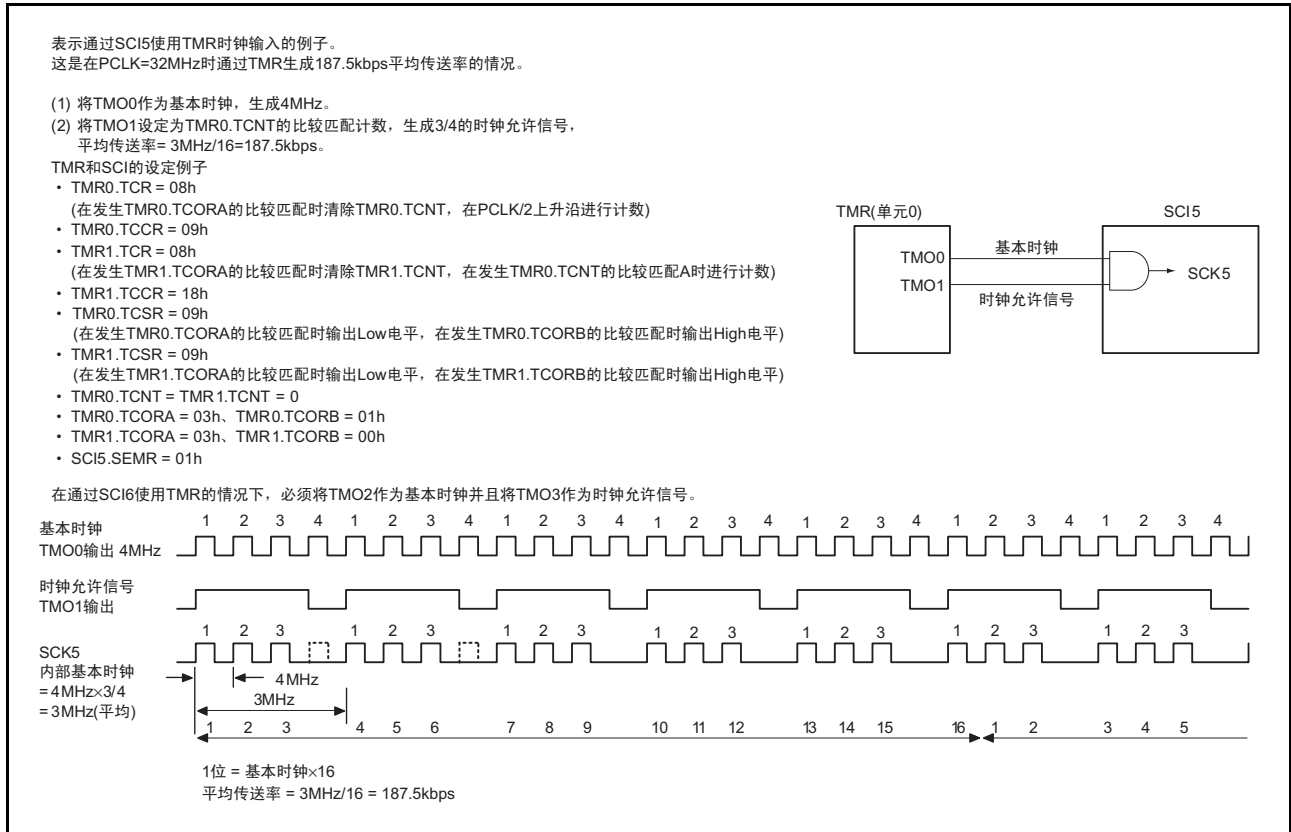
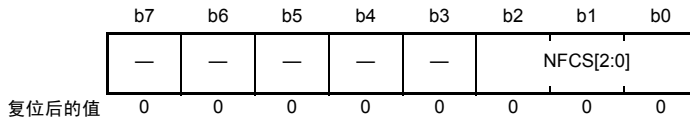


图 28.4 输入 TMR 时钟时的平均传送率的设定例子

28.2.11 噪声滤波器的设定寄存器 (SNFR)

地址 SCI0.SNFR 0008 A008h、SCI1.SNFR 0008 A028h、SCI5.SNFR 0008 A0A8h、SCI6.SNFR 0008 A0C8h、SCI8.SNFR 0008 A108h、SCI9.SNFR 0008 A128h、SCI12.SNFR 0008 B308h



位	符号	位名	功能	R/W
b2-b0	NFCS[2:0]	噪声滤波器的时钟选择位	<p>在异步模式中，以基本时钟为基准。</p> <p>b2 b0 0 0 0: 将 1 分频时钟用于噪声滤波器</p> <p>在简易 I²C 模式中，以 SMR.CKS[1:0] 位所选内部波特率发生器的时钟源为基准。</p> <p>b2 b0 0 0 1: 将 1 分频时钟用于噪声滤波器 0 1 0: 将 2 分频时钟用于噪声滤波器 0 1 1: 将 4 分频时钟用于噪声滤波器 1 0 0: 将 8 分频时钟用于噪声滤波器</p> <p>上述以外：不能设定</p>	R/W (注 1)
b7-b3	—	保留位	读写值都为“0”。	R/W

注 1. 只能在 SCR.TE 位和 SCR.RE 位都为“0”（禁止串行发送和串行接收）时写这些位。

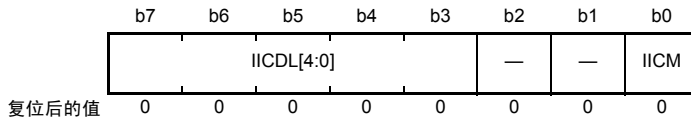
NFCS[2:0] 位（噪声滤波器的时钟选择位）

这些位选择数字噪声滤波器的采样时钟。

在异步模式中使用噪声滤波器的情况下，必须置“000b”；在简易 I²C 模式中使用噪声滤波器的情况下，必须置“001b”～“100b”中的任意一个值。

28.2.12 I²C 模式寄存器 1 (SIMR1)

地址 SCI0.SIMR1 0008 A009h、SCI1.SIMR1 0008 A029h、SCI5.SIMR1 0008 A0A9h、SCI6.SIMR1 0008 A0C9h、SCI8.SIMR1 0008 A109h、SCI9.SIMR1 0008 A129h、SCI12.SIMR1 0008 B309h



位	符号	位名	功能	R/W
b0	IICM	简易 I ² C 模式选择位	SMIF IICM 0 0: 串行接口模式 (异步模式、时钟同步模式或者简易 SPI 模式) 0 1: 简易 I ² C 模式 1 0: 智能卡接口模式 1 1: 不能设定	R/W (注 1)
b2-b1	—	保留位	读写值都为“0”。	R/W
b7-b3	IICDL[4:0]	SSDA 输出延迟选择位	(以内部波特率发生器的时钟源为基准) b7 b3 00000: 没有输出延迟 00001: 0~1 个周期 00010: 1~2 个周期 00011: 2~3 个周期 00100: 3~4 个周期 00101: 4~5 个周期 : 11110: 29~30 个周期 11111: 30~31 个周期	R/W (注 1)

注 1. 只能在 SCR.TE 位和 SCR.RE 位都为“0”（禁止串行发送和串行接收）时写这些位。

SIMR1 寄存器是用于选择简易 I²C 模式和 SSDA 输出延迟段数的寄存器。

IICM 位（简易 I²C 模式选择位）

此位通过和 SCMR.SMIF 位的组合来选择运行模式。

IICDL[4:0] 位（SSDA 输出延迟选择位）

这些位选择相对于 SSCLn 引脚输出的下降沿的 SSDAn 引脚输出的延迟。能将内部波特率发生器的时钟源作为 1 个周期，选择无延迟~ 31 个周期。内部波特率发生器的时钟源是指通过设定 SMR.CKS[1:0] 位将 PCLK 进行分频的时钟。在简易 I²C 模式中必须将这些位置“00001b”~“11111b”中的任意一个值，在其他模式中必须置“00000b”。

28.2.13 I²C 模式寄存器 2 (SIMR2)

地址 SCI0.SIMR2 0008 A00Ah、SCI1.SIMR2 0008 A02Ah、SCI5.SIMR2 0008 A0AAh、SCI6.SIMR2 0008 A0CAh、SCI8.SIMR2 0008 A10Ah、SCI9.SIMR2 0008 A12Ah、SCI12.SIMR2 0008 B30Ah

b7	b6	b5	b4	b3	b2	b1	b0
—	—	IICACKT	—	—	—	IICCSC	IICINTM
复位后的值	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	IICINTM	I ² C 中断模式选择位	0: 使用 ACK/NACK 中断 1: 使用接收中断和发送中断	R/W (注1)
b1	IICCSC	时钟同步位	0: 不进行时钟同步 1: 进行时钟同步	R/W (注1)
b4-b2	—	保留位	读写值都为“0”。	R/W
b5	IICACKT	ACK 发送数据位	0: 发送 ACK 1: 发送 NACK 或者接收 ACK/NACK	R/W
b7-b6	—	保留位	读写值都为“0”。	R/W

注 1. 只能在 SCR.TE 位和 SCR.RE 位都为“0”（禁止串行发送和串行接收）时写这些位。

SIMR2 寄存器是用于选择简易 I²C 模式的发送 / 接收控制的寄存器。

IICINTM 位 (I²C 中断模式选择位)

此位选择简易 I²C 模式中的中断请求源。

IICCSC 位 (时钟同步位)

在为了使其他设备插入等待而将 SSCLn 引脚置为 Low 电平的情况下，如果要与内部生成的 SSCLn 时钟取得同步，就将 IICCSC 位置“1”。

如果将 IICCSC 位置“0”，就不进行 SSCLn 时钟的同步。与 SSCLn 引脚的输入无关，根据 BRR 寄存器设定的位速率生成 SSCLn 时钟。

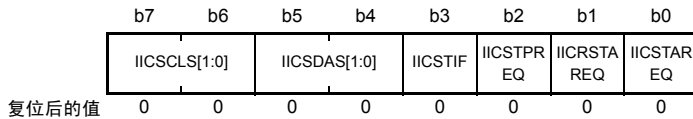
除了调试以外，必须将 IICCSC 位置“1”。

IICACKT 位 (ACK 发送数据位)

此位保存发送数据的 ACK 位。必须在接收 ACK/NACK 位时将此位置“1”。

28.2.14 I²C 模式寄存器 3 (SIMR3)

地址 SCI0.SIMR3 0008 A00Bh、SCI1.SIMR3 0008 A02Bh、SCI5.SIMR3 0008 A0ABh、SCI6.SIMR3 0008 A0CBh、SCI8.SIMR3 0008 A10Bh、SCI9.SIMR3 0008 A12Bh、SCI12.SIMR3 0008 B30Bh



位	符号	位名	功能	R/W
b0	IICSTARREQ	开始条件生成位	0: 不生成开始条件 1: 生成开始条件 (注1、注3、注4)	R/W
b1	IICRSTAREQ	重新开始条件生成位	0: 不生成重新开始条件 1: 生成重新开始条件 (注2、注3、注4)	R/W
b2	IICSTPREQ	停止条件生成位	0: 不生成停止条件 1: 生成停止条件 (注2、注3、注4)	R/W
b3	IICSTIF	开始 / 重新开始 / 停止条件的生成结束标志	0: 没有各条件生成请求的状态或者正在生成的状态 1: 各条件生成结束的状态	R/W
b5-b4	IICSDAS[1:0]	SSDA 输出选择位	b5 b4 0 0: 串行数据输出 0 1: 生成开始条件、重新开始条件和停止条件 1 0: SSDAn 引脚输出 Low 电平 1 1: SSDAn 引脚为高阻抗状态	R/W
b7-b6	IICSCLS[1:0]	SSCL 输出选择位	b7 b6 0 0: 串行时钟输出 0 1: 生成开始条件、重新开始条件和停止条件 1 0: SSCLn 引脚输出 Low 电平 1 1: SSCLn 引脚为高阻抗状态	R/W

注 1. 必须确认总线状态并且在总线空闲的状态下生成开始条件。

注 2. 必须确认总线状态并且在总线忙的状态下生成重新开始条件或者停止条件。

注 3. 不能将 IICSTARREQ 位、IICRSTAREQ 位或者 IICSTPREQ 位中的 2 位或者全部置“1”。

注 4. 必须在将 IICSTIF 标志置“0”后生成各条件。

SIMR3 寄存器用于对简易 I²C 模式的开始条件和停止条件的生成以及对 SSDAn 引脚和 SSCLn 引脚输出值的固定进行控制。

IICSTARREQ 位 (开始条件生成位)

在生成开始条件时, 必须在将 IICSTARREQ 位置“1”的同时将 IICSDAS[1:0] 位和 IICSCLS[1:0] 位分别置“01b”。

[为“1”的条件]

- 当写“1”时

[为“0”的条件]

- 当开始条件生成结束时

IICRSTAREQ 位 (重新开始条件生成位)

在生成重新开始条件时, 必须在将 IICRSTAREQ 位置“1”的同时将 IICSDAS[1:0] 位和 IICSCLS[1:0] 位分别置“01b”。

[为“1”的条件]

- 当写“1”时

[为“0”的条件]

- 当重新开始条件生成结束时

IICSTPREQ 位 (停止条件生成位)

在生成停止条件时, 必须在将 IICSTPREQ 位置“1”的同时将 IICSDAS[1:0] 位和 IICSCLS[1:0] 位分别置“01b”。

[为“1”的条件]

- 当写“1”时

[为“0”的条件]

- 当停止条件生成结束时

IICSTIF 标志 (开始 / 重新开始 / 停止条件的生成结束标志)

此标志表示在生成各条件后生成结束的状态。必须在将 IICSTIF 标志置“0”后, 通过 IICSTAREQ 位、IICRSTAREQ 位和 IICSTPREQ 位生成各条件。

在通过 SCR.TEIE 位允许中断请求的状态下, 在 IICSTIF 标志为“1”时输出开始 / 重新开始 / 停止条件的生成结束中断 (STI) 请求。

[为“1”的条件]

- 当开始、重新开始和停止的各条件生成结束时 (如果和为“0”的条件发生冲突, 就优先为“0”的条件。)

[为“0”的条件]

- 当写“0”时 (必须确认 IICSTIF 标志已变为“0”)
- 当 SIMR1.IICM 位为“0”时 (简易 I²C 模式除外)
- 当 SCR.TE 位为“0”时

IICSDAS 位 (SSDA 输出选择位)

此位控制 SSDAn 引脚的输出。

在正常运行时, 必须给 IICSDAS 位和 IICSCLS 位设定相同的值。

IICSCLS 位 (SSCL 输出选择位)

此位控制 SSCLn 引脚的输出。

在正常运行时, 必须给 IICSCLS 位和 IICSDAS 位设定相同的值。

28.2.15 I²C 状态寄存器 (SISR)

地址 SCI0.SISR 0008 A00Ch、SCI1.SISR 0008 A02Ch、SCI5.SISR 0008 A0ACh、SCI6.SISR 0008 A0CCh、SCI8.SISR 0008 A10Ch、SCI9.SISR 0008 A12Ch、SCI12.SISR 0008 B30Ch

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	IICACKR
复位后的值	0	0	x	x	0	x	0	0

x: 不定值

位	符号	位名	功能	R/W
b0	IICACKR	ACK 接收数据标志	0: 接收 ACK 1: 接收 NACK	R/W (注1)
b1	—	保留位	读写值都为“0”。	R/W
b2	—	保留位	读取值为不定值。	R
b3	—	保留位	读写值都为“0”。	R/W
b5-b4	—	保留位	读取值为不定值。	R
b7-b6	—	保留位	读写值都为“0”。	R/W

注1. 为了清除标志, 只能写“0”。

SISR 寄存器监视简易 I²C 模式的相关状态。

IICACKR 标志 (ACK 接收数据标志)

此标志能读已接收的 ACK/NACK 位。

在接收 ACK/NACK 的位的 SSCLn 时钟上升时更新 IICACK 标志。

28.2.16 SPI 模式寄存器 (SPMR)

地址 SCI0.SPMR 0008 A00Dh、SCI1.SPMR 0008 A02Dh、SCI5.SPMR 0008 A0ADh、SCI6.SPMR 0008 A0CDh、SCI8.SPMR 0008 A10Dh、SCI9.SPMR 0008 A12Dh、SCI12.SPMR 0008 B30Dh

b7	b6	b5	b4	b3	b2	b1	b0
CKPH	CKPOL	—	MFF	—	MSS	CTSE	SSE
复位后的值	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	SSE	SS 引脚功能允许位	0: 禁止 SS 引脚功能 1: 允许 SS 引脚功能	R/W (注 1)
b1	CTSE	CTS 允许位	0: 禁止 CTS 功能 (RTS 输出功能有效) 1: 允许 CTS 功能	R/W (注 1)
b2	MSS	主控从属选择位	0: TXDn 引脚: 发送、RXDn 引脚: 接收 (主控模式) 1: TXDn 引脚: 接收、RXDn 引脚: 发送 (从属模式)	R/W (注 1)
b3	—	保留位	读写值都为“0”。	R/W
b4	MFF	模式故障标志	0: 没有模式故障错误 1: 有模式故障错误	R/W (注 2)
b5	—	保留位	读写值都为“0”。	R/W
b6	CKPOL	时钟极性选择位	0: 没有时钟极性反相 1: 有时钟极性反相	R/W (注 1)
b7	CKPH	时钟相位选择位	0: 没有时钟延迟 1: 有时钟延迟	R/W (注 1)

注 1. 只能在 SCR.TE 位和 SCR.RE 位都为“0” (禁止串行发送和串行接收) 时写这些位。

注 2. 为了清除标志, 只能写“0”。

SPMR 寄存器是用于选择异步模式和时钟同步模式扩展设定的寄存器。

SSE 位 (SS 引脚功能允许位)

在使用 SSn# 引脚进行发送和接收控制时 (简易 SPI 模式), 将此位置“1”, 而在其他通信模式中, 必须将此位置“0”。即使在简易 SPI 模式中, 如果用于主控模式 (SCR.CKE[1:0] 位 =00b 并且 MSS 位 =0) 的单主控, 就不需要使用主控侧的 SSn# 引脚进行发送和接收控制, 并且将 SSE 位置“0”。不能将 SSE 位和 CTSE 位都置为有效 (如果设定, 两位就都变为无效)。

CTSE 位 (CTS 允许位)

在将 SSn# 引脚用作 CTS 控制信号输入引脚进行发送和接收控制时, 将此位置“1”。在将此位置“0”的状态下输出 RTS 信号。在智能卡接口模式、简易 SPI 模式和简易 I²C 模式中, 必须将此位置“0”。不能将 CTSE 位和 SSE 位都置为有效 (即使设定也无效)。

MSS 位 (主控从属选择位)

此位在简易 SPI 模式中选择主控模式和从属模式。如果将 MSS 位置“1”，TXDn 引脚和 RXDn 引脚的功能就相反，从 TXDn 引脚输入接收数据并且从 RXDn 引脚输出发送数据。

在简易 SPI 模式以外的模式中，必须将此位置“0”。

MFF 标志 (模式故障标志)

此标志表示发生了模式故障错误。

在多主控模式中，必须通过读 MFF 标志来判断模式故障错误。

[为“1”的条件]

- 在设定为简易 SPI 模式的主控模式 (SSE 位=1 并且 MSS 位=0) 的情况下 SSn# 引脚的输入变为 Low 电平时。

[为“0”的条件]

- 在读“1”的状态后写“0”时

CKPOL 位 (时钟极性选择位)

此位选择 SCKn 引脚的时钟输出极性，详细内容请参照图 28.52。

在简易 SPI 模式和时钟同步模式以外的模式中，必须将此位置“0”。

CKPH 位 (时钟相位选择位)

此位选择 SCKn 引脚的时钟输出的相位设定，详细内容请参照图 28.52。

在简易 SPI 模式和时钟同步模式以外的模式中，必须将此位置“0”。

28.2.17 扩展串行模式有效寄存器（ESMER）

地址 SCI12.ESMER 0008 B320h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	ESME
0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	ESME	扩展串行模式有效位	0: 扩展串行模式无效 1: 扩展串行模式有效	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

ESME 位（扩展串行模式有效位）

当 ESME 位为“1”时，扩展串行模式控制部有效。

当将 ESME 位置“0”时，为以下状态：

- 扩展串行模式控制部为被初始化的状态。

表 28.22 ESME 位的设定和定时器运行模式的运行保证

ESME 位	定时器模式	Break Field Low width 判定模式	Break Field Low width 输出模式
0	○（注1）	×	×
1	○	○	○

○：需要运行保证。×：不需要运行保证。

注 1. 只在选择 PCLK 时运行。

28.2.18 控制寄存器 0（CR0）

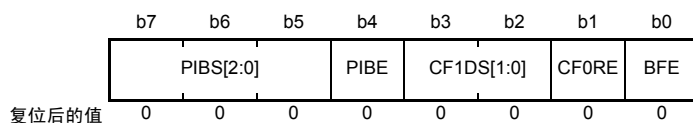
地址 SCI12.CR0 0008 B321h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	BRME	RXDSF	SFSF	—
0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	—	保留位	读写值都为“0”。	R/W
b1	SFSF	Start Frame 状态标志	0: Start Frame 检测功能无效状态 1: Start Frame 检测功能有效状态	R
b2	RXDSF	RXDX12 输入状态标志	0: RXDX12 输入允许状态 1: RXDX12 输入禁止状态	R
b3	BRME	位速率测量允许位	0: 位速率测量无效 1: 位速率测量有效	R/W
b7-b4	—	保留位	读写值都为“0”。	R/W

28.2.19 控制寄存器 1 (CR1)

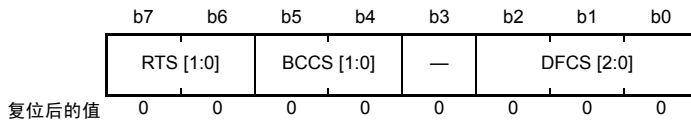
地址 SCI12.CR1 0008 B322h



位	符号	位名	功能	R/W
b0	BFE	Break Field 允许位	0: Break Field 的检测无效 1: Break Field 的检测有效	R/W
b1	CF0RE	Control Field 0 接收允许位	0: Control Field 0 的接收无效 1: Control Field 0 的接收有效	R/W
b3-b2	CF1DS[1:0]	Control Field 1 数据寄存器选择位	b3 b2 0 0: 选择 PCF1DR 作为比较数据 0 1: 选择 SCF1DR 作为比较数据 1 0: 选择 PCF1DR 和 SCF1DR 作为比较数据 1 1: 不能设定	R/W
b4	PIBE	优先级中断位的允许位	0: 优先级中断位无效 1: 优先级中断位有效	R/W
b7-b5	PIBS[2:0]	优先级中断位的选择位	b7 b5 0 0 0: Control Field 1 第 0 位 0 0 1: Control Field 1 第 1 位 0 1 0: Control Field 1 第 2 位 0 1 1: Control Field 1 第 3 位 1 0 0: Control Field 1 第 4 位 1 0 1: Control Field 1 第 5 位 1 1 0: Control Field 1 第 6 位 1 1 1: Control Field 1 第 7 位	R/W

28.2.20 控制寄存器 2 (CR2)

地址 SCI12.CR2 0008 B323h



位	符号	位名	功能	R/W
b2-b0	DFCS[2:0]	RXD _X 信号数字滤波器的时钟选择位	b2 b0 0 0 0: 滤波器无效 0 0 1: 滤波器有效 (SCI 基本时钟) 0 1 0: 滤波器有效 (SCI 基本时钟 /8) 0 1 1: 滤波器有效 (SCI 基本时钟 /16) 1 0 0: 滤波器有效 (SCI 基本时钟 /32) 1 0 1: 滤波器有效 (SCI 基本时钟 /64) 1 1 0: 滤波器有效 (SCI 基本时钟 /128) 1 1 1: 不能设定	R/W
b3	—	保留位	读写值都为“0”。	R/W
b5-b4	BCCS[1:0]	总线冲突检测时钟选择位	b5 b4 0 0: SCI 基本时钟 0 1: SCI 基本时钟的 2 分频 1 0: SCI 基本时钟的 4 分频 1 1: 不能设定	R/W
b7-b6	RTS[1:0]	RXD _X 接收采样时序选择位	<ul style="list-style-type: none"> • SCI12.SEMR.ABCS 位为“0”的情况 <li style="margin-left: 20px;">b7 b6 <li style="margin-left: 20px;">0 0: SCI 基本时钟的第 8 个上升沿 <li style="margin-left: 20px;">0 1: SCI 基本时钟的第 10 个上升沿 <li style="margin-left: 20px;">1 0: SCI 基本时钟的第 12 个上升沿 <li style="margin-left: 20px;">1 1: SCI 基本时钟的第 14 个上升沿 • SCI12.SEMR.ABCS 位为“1”的情况 <li style="margin-left: 20px;">b7 b6 <li style="margin-left: 20px;">0 0: SCI 基本时钟的第 4 个上升沿 <li style="margin-left: 20px;">0 1: SCI 基本时钟的第 5 个上升沿 <li style="margin-left: 20px;">1 0: SCI 基本时钟的第 6 个上升沿 <li style="margin-left: 20px;">1 1: SCI 基本时钟的第 7 个上升沿 	R/W

注. 当 SCI12.SEMR.ABCS 为“0”时, SCI 基本时钟为 1 个数据期间的 1/16 个周期; 当 SCI12.SEMR.ABCS 为“1”时, SCI 基本时钟为 1 个数据期间的 1/8 个周期。在使用 SCI 基本时钟的情况下, 必须将 SCI12.SCR.TE 位置“1”。

28.2.21 控制寄存器 3（CR3）

地址 SCI12.CR3 0008 B324h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	SDST
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	SDST	Start Frame 检测开始位	0: 不进行 Start Frame 的检测 1: 进行 Start Frame 的检测	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

SDST 位（Start Frame 检测开始位）

如果将 SDST 位置“1”，就开始检测 Start Frame。读取值为“0”。

28.2.22 端口控制寄存器（PCR）

地址 SCI12.PCR 0008 B325h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	SHARPS	—	—	RDXPS	TXDXPS
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	TXDXPS	TXDX12 信号极性选择位	0: 不将 TXDX12 信号极性进行反相输出 1: 将 TXDX12 信号极性进行反相输出	R/W
b1	RDXPS	RDX12 信号极性选择位	0: 不将 RDX12 极性进行反相输入 1: 将 RDX12 极性进行反相输入	R/W
b3-b2	—	保留位	读写值都为“0”。	R/W
b4	SHARPS	TXDX12/RDX12 引脚兼用选择位	0: TXDX12 引脚和 RDX12 引脚独立 1: TXDX12 引脚和 RDX12 引脚兼用	R/W
b7-b5	—	保留位	读写值都为“0”。	R/W

SHARPS 位（TXDX12/RDX12 引脚兼用选择位）

当 SHARPS 位为“1”时，能进行 TXDX12 引脚和 RDX12 引脚兼用的半双工通信。

28.2.23 中断控制寄存器 (ICR)

地址 SC12.ICR 0008 B326h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	AEDIE	BCDIE	PIBDIE	CF1MIE	CF0MIE	BFDIE
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	BFDIE	Break Field Low width 检测中断允许位	0: 禁止 Break Field Low width 检测中断 1: 允许 Break Field Low width 检测中断	R/W
b1	CF0MIE	Control Field 0 相同中断允许位	0: 禁止 Control Field 0 相同中断 1: 允许 Control Field 0 相同中断	R/W
b2	CF1MIE	Control Field 1 相同中断允许位	0: 禁止 Control Field 1 相同中断 1: 允许 Control Field 1 相同中断	R/W
b3	PIBDIE	优先级中断位的检测中断允许位	0: 禁止优先级中断位检测中断 1: 允许优先级中断位检测中断	R/W
b4	BCDIE	总线冲突检测中断允许位	0: 禁止总线冲突检测中断 1: 允许总线冲突检测中断	R/W
b5	AEDIE	有效边沿检测中断允许位	0: 禁止有效边沿检测中断 1: 允许有效边沿检测中断	R/W
b7-b6	—	保留位	读写值都为“0”。	R/W

28.2.24 状态寄存器 (STR)

地址 SCI12.STR 0008 B327h

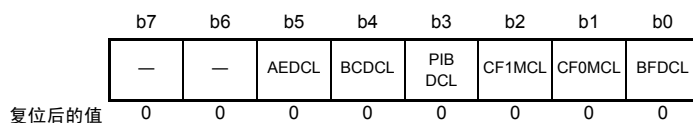
b7	b6	b5	b4	b3	b2	b1	b0
—	—	AEDF	BCDF	PIBDF	CF1MF	CF0MF	BDFD
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	BDFD	Break Field Low width 检测标志	[为“1”的条件] <ul style="list-style-type: none"> 当检测到 Break Field Low width 时 当 Break Field Low width 输出结束时 当定时器发生下溢时 [为“0”的条件] <ul style="list-style-type: none"> 当给 STCR.BFDCL 位写“1”时 	R
b1	CF0MF	Control Field 0 相同标志	[为“1”的条件] <ul style="list-style-type: none"> 当 Control Field 0 的接收数据和设定的数据相同时 [为“0”的条件] <ul style="list-style-type: none"> 当给 STCR.CF0MCL 位写“1”时 	R
b2	CF1MF	Control Field 1 相同标志	[为“1”的条件] <ul style="list-style-type: none"> 当 Control Field 1 的接收数据和设定的数据相同时 [为“0”的条件] <ul style="list-style-type: none"> 当给 STCR.CF1MCL 位写“1”时 	R
b3	PIBDF	优先级中断位检测标志	[为“1”的条件] <ul style="list-style-type: none"> 当检测到优先级中断位时 [为“0”的条件] <ul style="list-style-type: none"> 当给 STCR.PIBDCL 位写“1”时 	R
b4	BCDF	总线冲突检测标志	[为“1”的条件] <ul style="list-style-type: none"> 当检测到总线冲突时 [为“0”的条件] <ul style="list-style-type: none"> 当给 STCR.BCDCL 位写“1”时 	R
b5	AEDF	有效边沿检测标志	[为“1”的条件] <ul style="list-style-type: none"> 当检测到有效边沿时 [为“0”的条件] <ul style="list-style-type: none"> 当给 STCR.AEDCL 位写“1”时 	R
b7-b6	—	保留位	读取值为“0”，写操作无效。	R

28.2.25 状态清除寄存器 (STCR)

地址 SCI12.STCR 0008 B328h



位	符号	位名	功能	R/W
b0	BFDFCL	BFDF 清除位	如果将 BFDFCL 位置“1”，就清除 STR.BFDF 标志。读取值为“0”。	R/W
b1	CF0MCL	CF0MF 清除位	如果将 CF0MCL 位置“1”，就清除 STR.CF0MF 标志。读取值为“0”。	R/W
b2	CF1MCL	CF1MF 清除位	如果将 CF1MCL 位置“1”，就清除 STR.CF1MF 标志。读取值为“0”。	R/W
b3	PIBDCL	PIBDF 清除位	如果将 PIBDFCL 位置“1”，就清除 STR.PIBDF 标志。读取值为“0”。	R/W
b4	BCDCL	BCDF 清除位	如果将 BCDCL 位置“1”，就清除 STR.BCDF 标志。读取值为“0”。	R/W
b5	AEDCL	AEDF 清除位	如果将 AEDCL 位置“1”，就清除 STR.AEDF 标志。读取值为“0”。	R/W
b7-b6	—	保留位	读写值都为“0”。	R/W

28.2.26 Control Field 0 数据寄存器 (CF0DR)

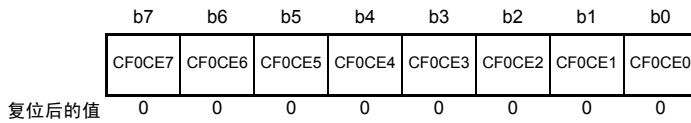
地址 SCI12.CF0DR 0008 B329h



CF0DR 寄存器是保存 Control Field 0 比较数据的 8 位可读写寄存器。

28.2.27 Control Field 0 比较允许寄存器 (CF0CR)

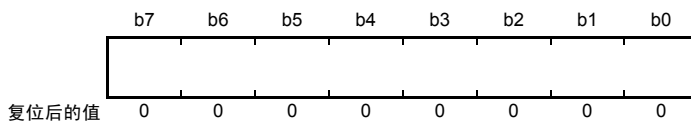
地址 SCI12.CF0CR 0008 B32Ah



位	符号	位名	功能	R/W
b0	CF0CE0	Control Field 0 bit0 的比较允许位	0: Control Field 0 bit0 的比较无效 1: Control Field 0 bit0 的比较有效	R/W
b1	CF0CE1	Control Field 0 bit1 的比较允许位	0: Control Field 0 bit1 的比较无效 1: Control Field 0 bit1 的比较有效	R/W
b2	CF0CE2	Control Field 0 bit2 的比较允许位	0: Control Field 0 bit2 的比较无效 1: Control Field 0 bit2 的比较有效	R/W
b3	CF0CE3	Control Field 0 bit3 的比较允许位	0: Control Field 0 bit3 的比较无效 1: Control Field 0 bit3 的比较有效	R/W
b4	CF0CE4	Control Field 0 bit4 的比较允许位	0: Control Field 0 bit4 的比较无效 1: Control Field 0 bit4 的比较有效	R/W
b5	CF0CE5	Control Field 0 bit5 的比较允许位	0: Control Field 0 bit5 的比较无效 1: Control Field 0 bit5 的比较有效	R/W
b6	CF0CE6	Control Field 0 bit6 的比较允许位	0: Control Field 0 bit6 的比较无效 1: Control Field 0 bit6 的比较有效	R/W
b7	CF0CE7	Control Field 0 bit7 的比较允许位	0: Control Field 0 bit7 的比较无效 1: Control Field 0 bit7 的比较有效	R/W

28.2.28 Control Field 0 接收数据寄存器 (CF0RR)

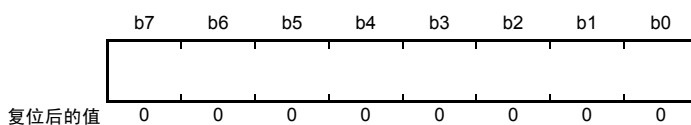
地址 SCI12.CF0RR 0008 B32Bh



CF0RR 寄存器是保存 Control Field 0 接收数据的 8 位可读寄存器。CPU 和 DTC 不能写 CF0RR 寄存器。

28.2.29 主 Control Field 1 数据寄存器 (PCF1DR)

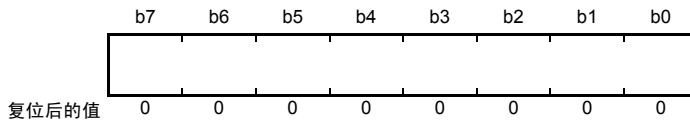
地址 SCI12.PCF1DR 0008 B32Ch



PCF1DR 寄存器是保存 Control Field 1 主比较数据的 8 位可读写寄存器。

28.2.30 次 Control Field 1 数据寄存器（SCF1DR）

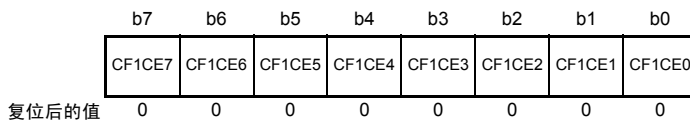
地址 SCI12.SCF1DR 0008 B32Dh



SCF1DR 寄存器是保存 Control Field 1 次比较数据的 8 位可读写寄存器。

28.2.31 Control Field 1 比较允许寄存器（CF1CR）

地址 SCI12.CF1CR 0008 B32Eh



位	符号	位名	功能	R/W
b0	CF1CE0	Control Field 1 bit0 的比较允许位	0: Control Field 1 bit0 的比较无效 1: Control Field 1 bit0 的比较有效	R/W
b1	CF1CE1	Control Field 1 bit1 的比较允许位	0: Control Field 1 bit1 的比较无效 1: Control Field 1 bit1 的比较有效	R/W
b2	CF1CE2	Control Field 1 bit2 的比较允许位	0: Control Field 1 bit2 的比较无效 1: Control Field 1 bit2 的比较有效	R/W
b3	CF1CE3	Control Field 1 bit3 的比较允许位	0: Control Field 1 bit3 的比较无效 1: Control Field 1 bit3 的比较有效	R/W
b4	CF1CE4	Control Field 1 bit4 的比较允许位	0: Control Field 1 bit4 的比较无效 1: Control Field 1 bit4 的比较有效	R/W
b5	CF1CE5	Control Field 1 bit5 的比较允许位	0: Control Field 1 bit5 的比较无效 1: Control Field 1 bit5 的比较有效	R/W
b6	CF1CE6	Control Field 1 bit6 的比较允许位	0: Control Field 1 bit6 的比较无效 1: Control Field 1 bit6 的比较有效	R/W
b7	CF1CE7	Control Field 1 bit7 的比较允许位	0: Control Field 1 bit7 的比较无效 1: Control Field 1 bit7 的比较有效	R/W

28.2.32 Control Field 1 接收数据寄存器（CF1RR）

地址 SCI12.CF1RR 0008 B32Fh



CF1RR 寄存器是保存 Control Field 1 接收数据的 8 位可读寄存器。CPU 和 DTC 不能写 CF1RR 寄存器。

28.2.33 定时器的控制寄存器（TCR）

地址 SCI12.TCR 0008 B330h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	TCST
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	TCST	定时器的计数开始位	0: 定时器停止计数 1: 定时器开始计数	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

28.2.34 定时器的模式寄存器（TMR）

地址 SCI12.TMR 0008 B331h

b7	b6	b5	b4	b3	b2	b1	b0
—	TCSS[2:0]		TWRC	—	TOMS[1:0]		—
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b1-b0	TOMS[1:0]	定时器运行模式选择位（注1）	b1 b0 0 0: 定时器模式 0 1: Break Field Low width 判定模式 1 0: Break Field Low width 输出模式 1 1: 不能设定	R/W
b2	—	保留位	读写值都为“0”。	R/W
b3	TWRC	计数器的写控制位	0: 写重加载寄存器和计数器 1: 只写重加载寄存器	R/W
b6-b4	TCSS[2:0]	定时器的计数时钟源选择位（注1）	b6 b4 0 0 0: PCLK 0 0 1: PCLK/2 0 1 0: PCLK/4 0 1 1: PCLK/8 1 0 0: PCLK/16 1 0 1: PCLK/32 1 1 0: PCLK/64 1 1 1: PCLK/128	R/W
b7	—	保留位	读写值都为“0”。	R/W

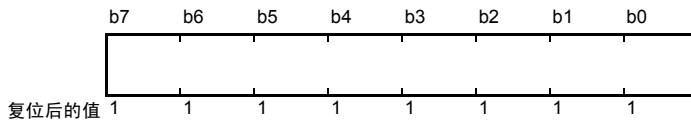
注1. 必须在定时器停止计数时（TCST=0）改写 TOMS[1:0] 位和 TCSS[2:0] 位。

TWRC 位（计数器的写控制位）

在写 TCNT 寄存器和 TPRES 寄存器时，此位选择是只写重加载寄存器还是写重加载寄存器和计数器。

28.2.35 定时器的预分频寄存器 (TPRE)

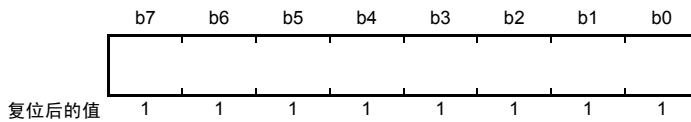
地址 SCI12.TPRE 0008 B332h



TPRE 寄存器由 8 位重加载寄存器、读缓冲器和计数器构成，初始值都为“FFh”。计数器通过 TMR.TCSS[2:0] 位选择的计数时钟源进行递减计数。如果发生下溢，就将重加载寄存器的值装入到计数器，而且下溢为 TCNT 计数器的计数时钟源。重加载寄存器和读缓冲器被分配在同一地址，在写此寄存器时，将值写到重加载寄存器；在读此寄存器时，读取值为传送到读缓冲器的计数值。在将重加载寄存器的值装入到计数器时，需要 1 个系统运行时钟。

28.2.36 定时器的计数器 (TCNT)

地址 SCI12.TCNT 0008 B333h



TCNT 计数器由 8 位重加载寄存器、读缓冲器和计数器构成，初始值都为“FFh”。对 TPRE 寄存器的下溢进行递减计数，如果 TCNT 计数器发生下溢，就将重加载寄存器的值装入到计数器。重加载寄存器和读缓冲器被分配在同一地址，在写此计数器时，将值写到重加载寄存器；在读此计数器时，读取值为传送到读缓冲器的计数值。在将重加载寄存器的值装入到计数器时，需要 1 个系统运行时钟。

28.3 异步模式的运行

异步串行通信的一般数据格式如图 28.5 所示。

按照起始位 (Low 电平)、发送 / 接收数据、奇偶校验位、停止位 (High 电平) 的顺序构成 1 帧。

在异步串行通信模式中, 通信线路通常保持标记状态 (High 电平)。

SCI 监视通信线路, 如果检测到 Low 电平, 就视为起始位并且开始串行通信。

SCI 内部的发送部和接收部各自独立, 因此能进行全双工通信。因为发送部和接收部都为双缓冲结构, 所以能在发送和接收时读写数据, 也能进行连续的发送和接收。

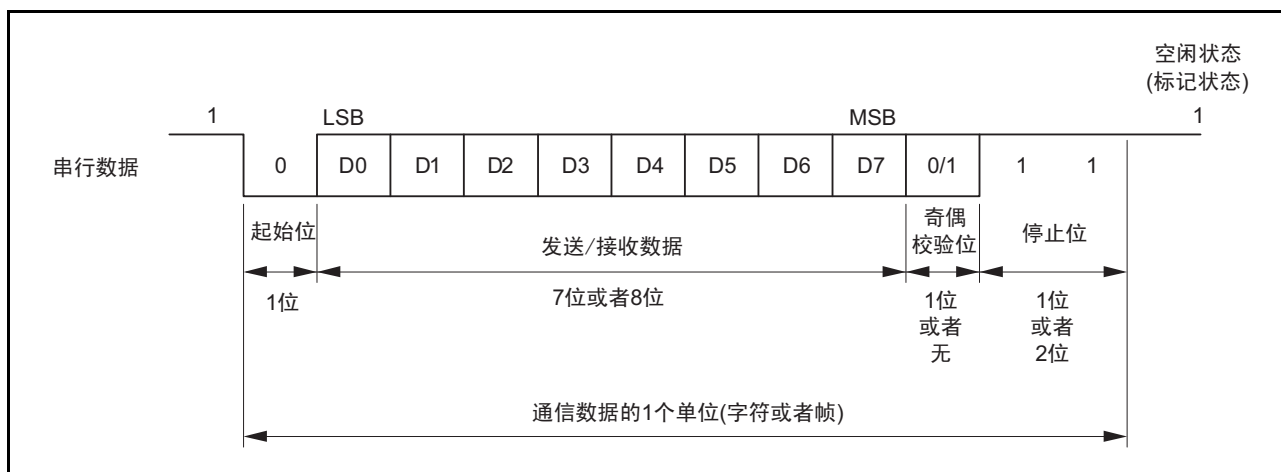


图 28.5 异步串行通信的数据格式 (8 位数据 / 有奇偶校验 / 2 个停止位的例子)

28.3.1 串行发送 / 接收格式

异步模式中能设定的串行发送 / 接收格式如表 28.23 所示。

串行发送 / 接收格式有 12 种，能通过设定 SMR 寄存器进行选择。有关多处理器通信功能的详细内容，请参照“28.4 多处理器通信功能”。

表 28.23 串行发送 / 接收格式 (异步模式)

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

- S : 起始位
- STOP : 停止位
- P : 奇偶校验位
- MPB : 多处理器位

28.3.2 异步模式的接收数据采样时序和接收容限

在异步模式中，SCI 通过频率为 16 倍位速率（注 1）的基本时钟运行。

在接收时，通过基本时钟对起始位的下降沿进行采样，并且与内部取得同步。如图 28.6 所示，通过在基本时钟的第 8 个周期（注 1）的上升沿对接收数据进行采样，在各位的中央取数据。因此，能用表达式（1）表示异步模式的接收容限。

$$M = \left| \left(0.5 - \frac{1}{2N} \right) - (L - 0.5)F - \frac{|D - 0.5|}{N} (1 + F) \right| \times 100 \quad [\%] \quad \dots \text{表达式(1)}$$

- M: 接收容限
- N: 对应时钟的位速率比
(当SEMR.ABCS位为“0”时，N=16；当ABCS位为“1”时，N=8)
- D: 时钟的占空比(D=0.5~1.0)
- L: 帧长(L=9~12)
- F: 时钟频率的偏差绝对值

假设表达式 (1) 中的 F（时钟频率的偏差绝对值）为“0”，D（时钟的占空比）为“0.5”，则

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

但是，此值毕竟是计算值，在进行系统设计时，必须留出 20 ~ 30% 的容限。

注 1. 这是 SEMR.ABCS 位为“0”时的例子。当 ABCS 位为“1”时，基本时钟的频率为位速率的 8 倍，在基本时钟的第 4 个上升沿对接收数据进行采样。

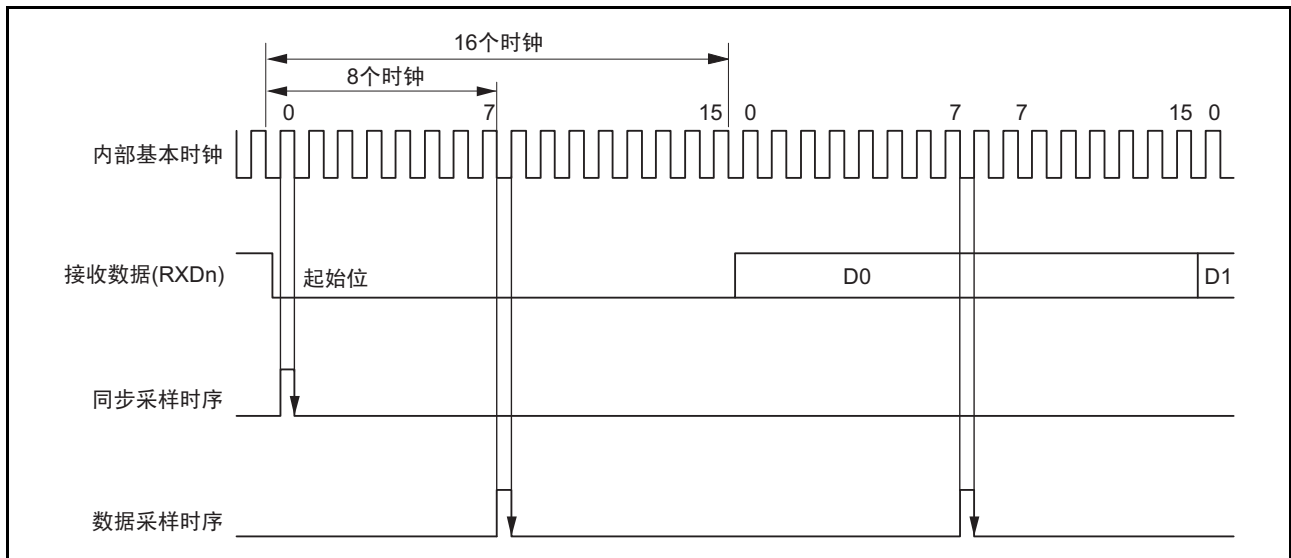


图 28.6 异步模式的接收数据采样时序

28.3.3 时钟

能通过设定 `SMR.CM` 位和 `SCR.CKE[1:0]` 位，选择内部波特率发生器生成的内部时钟或者 `SCKn` 引脚输入的外部时钟作为 `SCI` 的发送 / 接收时钟。

在使用外部时钟的情况下，必须将频率为 16 倍位速率 (`SEMR.ABCS` 位 = 0) 或者频率为 8 倍位速率 (`SEMR.ABCS` 位 = 1) 的时钟输入到 `SCKn` 引脚。如果选择外部时钟，就能通过设定 `SCIn.SEMR.ACS0` 位 ($n=5、6、12$)，选择 `TMR0`、`TMR1` 的基本时钟。

在通过内部时钟运行时，能从 `SCKn` 引脚输出时钟。此时，输出时钟的频率和位速率相等，发送时的相位如图 28.7 所示，时钟在发送数据的中央上升。

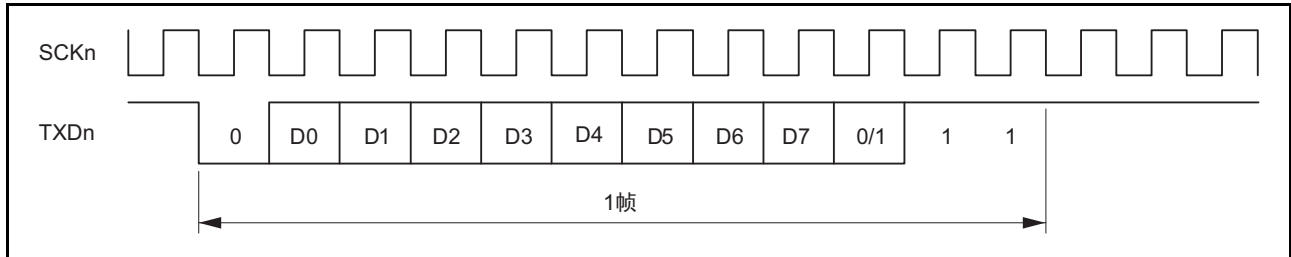


图 28.7 输出时钟和发送数据的相位关系
(异步模式: `SMR.CHR=0`、`SMR.PE=1`、`SMR.MP=0`、`SMR.STOP=1`)

28.3.4 CTS 和 RTS 功能

`CTS` 功能是使用 `CTSn` 引脚的输入进行发送控制的功能。

如果将 `SPMR.CTSE` 位置“1”，`CTS` 功能就有效。在 `CTS` 功能有效时并且只在 `CTSn` 引脚的输入为 `Low` 电平时，开始发送。

如果在发送过程中将 `CTSn` 引脚置为 `Low` 电平，正在发送的帧就不受影响而继续发送。

`RTS` 功能是使用 `RTSn` 引脚的输出进行接收请求的功能。如果处于可接收状态，就输出 `Low` 电平。输出 `Low` 电平或者 `High` 电平的条件如下：

[为 `Low` 电平的条件]

当满足以下全部条件时

- `SCR.RE` 位为“1”
- 不处于接收状态。
- 在读取前没有接收数据。
- `SSR.ORER`、`SSR.FER`、`SSR.PER` 标志全部为“0”。

[为 `High` 电平的条件]

- 当不满足 `Low` 电平的条件时

28.3.5 SCI 的初始化 (异步模式)

必须在发送和接收数据前给 SCR 寄存器写初始值“00h”，并且根据图 28.8 的流程图例子对 SCI 进行初始化。必须在对 SCR 寄存器进行初始化后更改运行模式或者通信格式。

在异步模式中使用外部时钟的情况下，必须提供时钟（包括初始化期间）。

必须注意：即使将 SCR.RE 位置“0”，也不对 SSR.ORER 标志、SSR.FER 标志、SSR.PER 标志和 RDR 寄存器进行初始化。

必须注意：如果将 SCR.TE 位从“1”置为“0”或者从“0”置为“1”，就在 SCR.TIE 位为“1”时产生 TXI 中断请求。

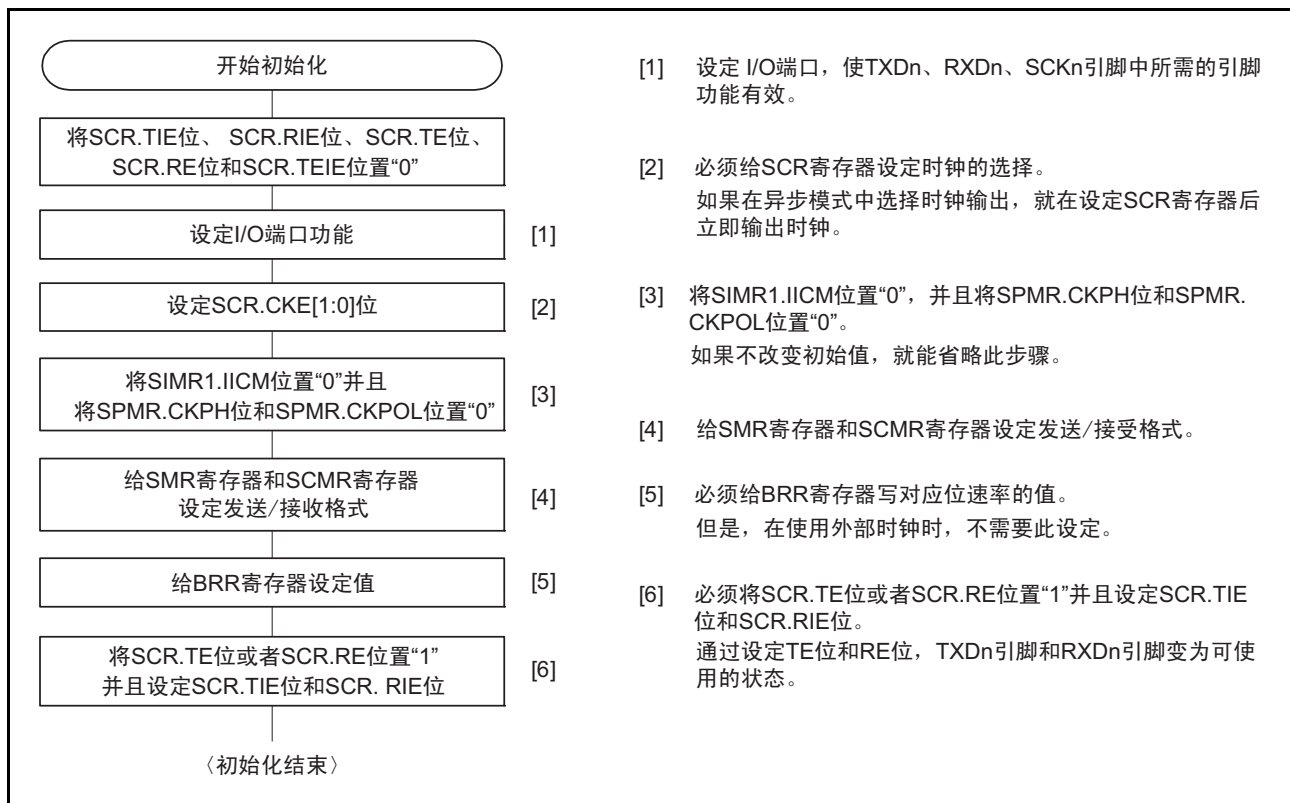


图 28.8 SCI 的初始化流程图例子 (异步模式)

28.3.6 串行数据的发送 (异步模式)

异步模式的串行发送例子如图 28.9 所示。

在发送串行数据时，SCI 的运行如下：

1. 如果通过TXI中断处理程序给TDR寄存器写数据，SCI就将数据从TDR寄存器传送到TSR寄存器。在开始发送时，通过在将SCR.TIE位置“1”后将SCR.TE位置“1”，或者用1条指令同时将这2个位置“1”，产生TXI中断请求。
2. 在SPMR.CTSE位为“0”（禁止CTS功能）或者CTSn#引脚的输入为Low电平的状态下，将数据从TDR寄存器传送到TSR寄存器，开始发送。此时，如果SCR.TIE位为“1”，就产生TXI中断请求。能通过TXI中断处理程序，在上次传送的数据发送结束前给TDR寄存器写下一个发送数据，进行连续的发送。当使用TEI中断请求时，在TXI中断请求处理程序中给TDR寄存器写最后的发送数据，然后将SCR.TIE位置“0”（禁止TXI中断请求）并且将SCR.TEIE位置“1”（允许TEI中断请求）。
3. 从TXDn引脚依次发送起始位、发送数据、奇偶校验位或者多处理器位（根据格式，有时没有此位）和停止位。
4. 在发送停止位时检查TDR寄存器的更新（写）。
5. 如果TDR寄存器已被更新，就在SPMR.CTSE位为“0”（禁止CTS功能）或者CTSn#引脚的输入为Low电平的状态下，将下一个发送数据从TDR寄存器传送到TSR寄存器，并且在发送停止位后开始下一帧的发送。
6. 如果TDR寄存器未被更新，就将SSR.TEND标志置“1”并且发送停止位后，通过输出High电平进入标记状态。此时，如果SCR.TEIE位为“1”，就将SSR.TEND标志置“1”并且产生TEI中断请求。

串行发送的流程图例子如图 28.10 所示。

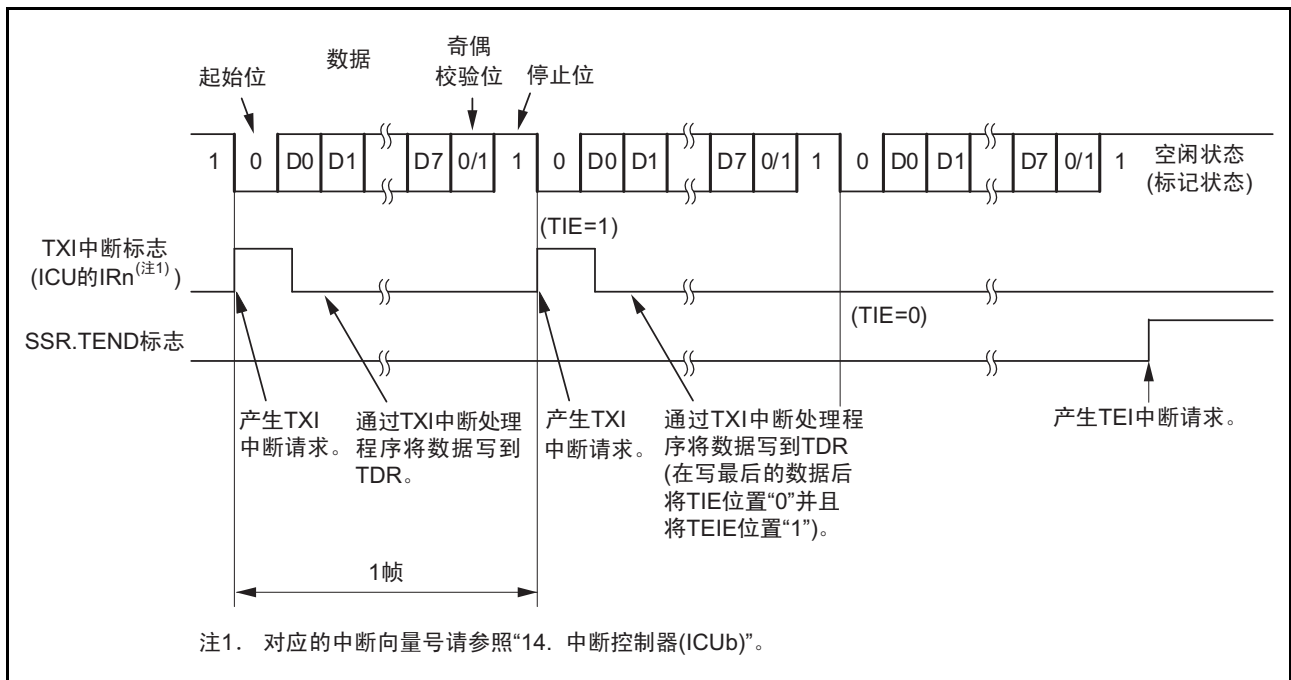


图 28.9 异步模式的串行发送 (正在发送~发送结束的情况) 例子
(8 位数据 / 有奇偶校验 / 1 个停止位的例子)

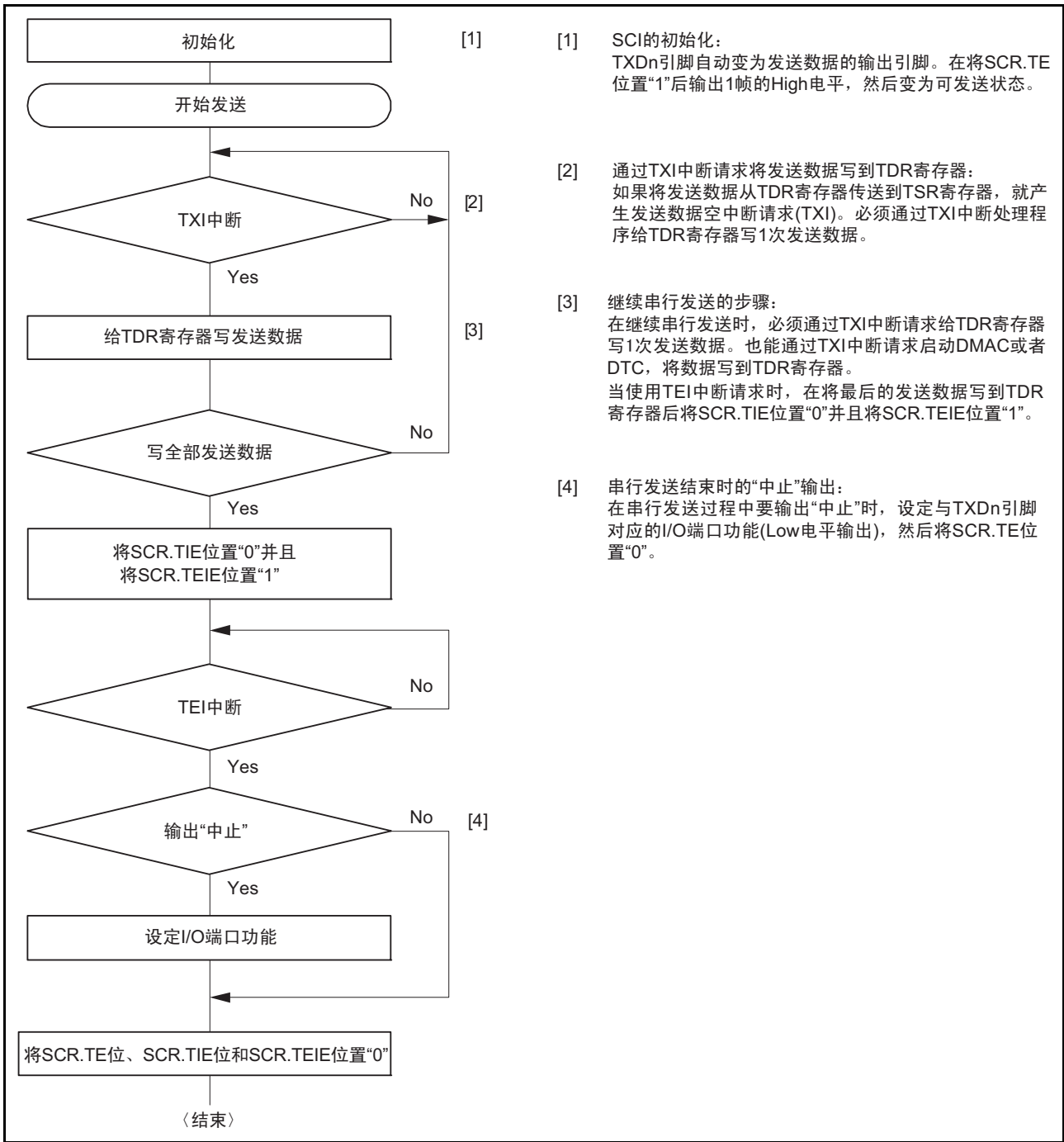


图 28.10 异步模式的串行发送的流程图例子

28.3.7 串行数据的接收 (异步模式)

异步模式的串行接收例子如图 28.11 和图 28.12 所示。

在接收串行数据时，SCI 的运行如下：

1. 如果将SCR.RE位置“1”，就将RTSn#引脚的输出置为Low电平。
2. 监视通信线路，如果检测到起始位，就与内部取得同步将接收数据取到RDR寄存器，并且检查奇偶校验位和停止位。
3. 如果发生溢出错误，就将SSR.ORER标志置位。此时，如果SCR.RIE位为“1”，就产生ERI中断请求。接收数据不传送到RDR寄存器。
4. 如果检测到奇偶校验错误，就将SSR.PER标志置位，并且将接收数据传送到RDR寄存器。此时，如果RIE位为“1”，就产生ERI中断请求。
5. 如果检测到帧错误（停止位为“0”时），就将SSR.FER标志置位，并且将接收数据传送到RDR寄存器。此时，如果RIE位为“1”，就产生ERI中断请求。
6. 当正常接收时，将接收数据传送到RDR寄存器。此时，如果RIE位为“1”，就产生RXI中断请求。能通过此RXI中断处理程序，在下一个数据接收结束前读已被传送到RDR寄存器的接收数据，进行连续的接收。如果读被传送到RDR寄存器的接收数据，就将RTSn#引脚的输出置为Low电平。

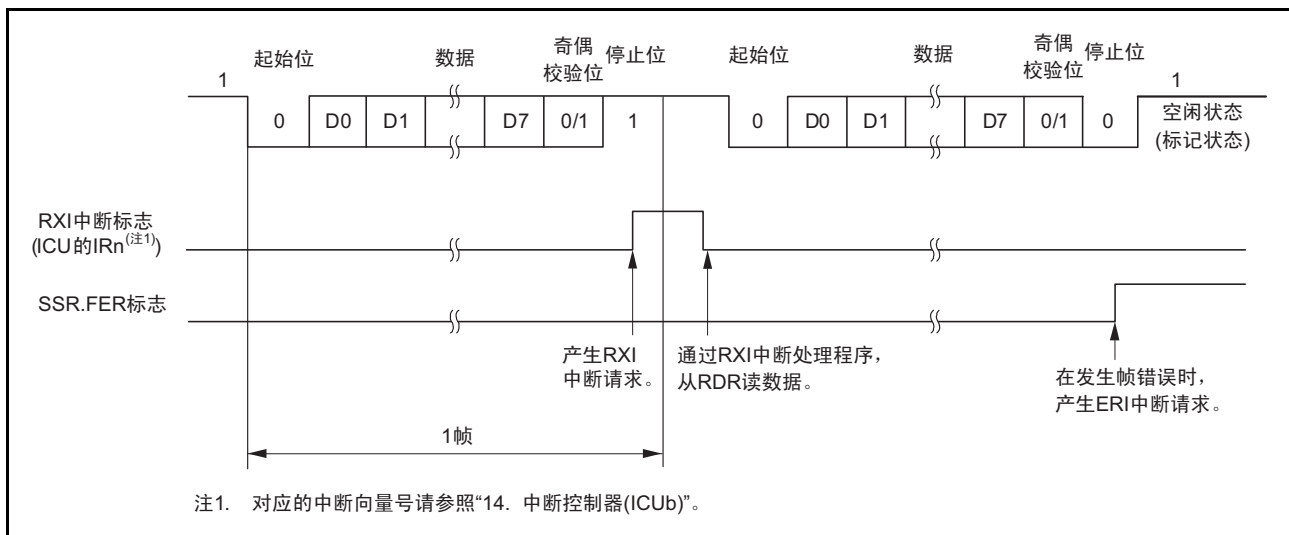


图 28.11 异步模式的串行接收例子 (1) (不使用 RTS 功能的情况)
(8 位数据 / 有奇偶校验 / 1 个停止位的例子)

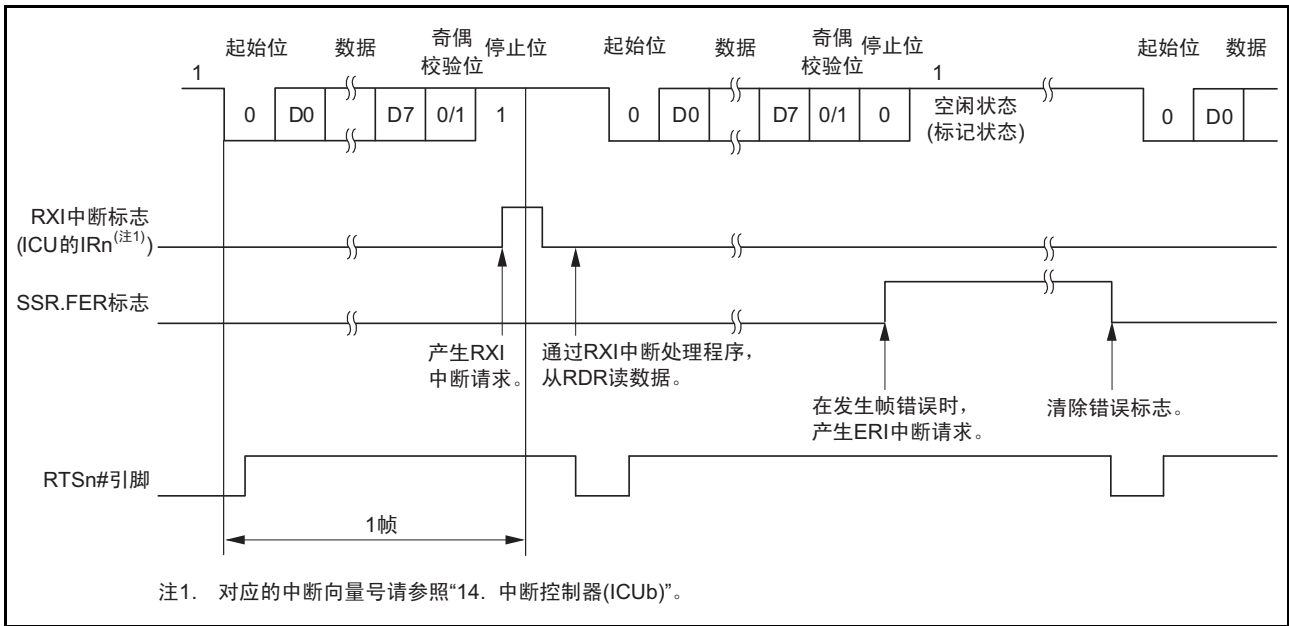


图 28.12 异步模式的串行接收例子 (2) (使用 RTS 功能的情况)
(8 位数据 / 有奇偶校验 / 1 个停止位的例子)

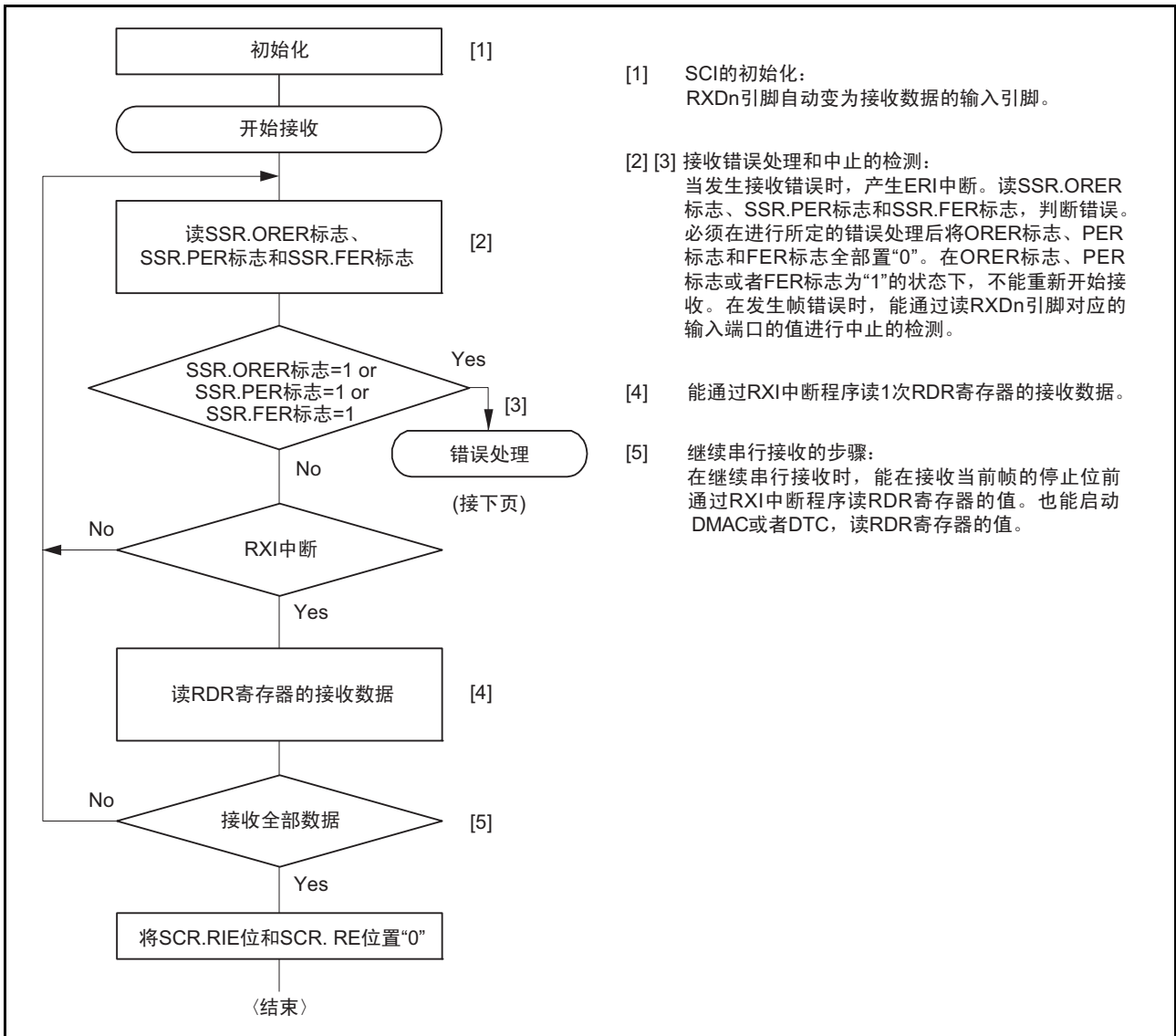
检测到接收错误时的 SSR 寄存器各状态标志的状态和接收数据的处理如表 28.24 所示。

一旦检测到接收错误, 就产生 ERI 中断请求而不产生 RXI 中断请求。在接收错误标志被置位的状态下不能进行以后的接收。因此, 必须在继续接收前将 ORER 标志、FER 标志和 PER 标志置“0”。另外, 在进行溢出错误处理时, 必须读 RDR 寄存器。

串行接收的流程图例子如图 28.13 和图 28.14 所示。

表 28.24 SSR 寄存器状态标志的状态和接收数据的处理

SSR 寄存器的状态标志			接收数据	接收错误的状态
ORER	FER	PER		
1	0	0	消失	溢出错误
0	1	0	传送到 RDR。	帧错误
0	0	1	传送到 RDR。	奇偶校验错误
1	1	0	消失	溢出错误 + 帧错误
1	0	1	消失	溢出错误 + 奇偶校验错误
0	1	1	传送到 RDR。	帧错误 + 奇偶校验错误
1	1	1	消失	溢出错误 + 帧错误 + 奇偶校验错误



- [1] SCI的初始化:
RXDn引脚自动变为接收数据的输入引脚。
- [2] [3] 接收错误处理和中止的检测:
当发生接收错误时,产生ERI中断。读SSR.ORER标志、SSR.PER标志和SSR.FER标志,判断错误。必须在进行所定的错误处理后将ORER标志、PER标志和FER标志全部置“0”。在ORER标志、PER标志或者FER标志为“1”的状态下,不能重新开始接收。在发生帧错误时,能通过读RXDn引脚对应的输入端口的值进行中止的检测。
- [4] 能通过RXI中断程序读1次RDR寄存器的接收数据。
- [5] 继续串行接收的步骤:
在继续串行接收时,能在接收当前帧的停止位前通过RXI中断程序读RDR寄存器的值。也能启动DMAC或者DTC,读RDR寄存器的值。

图 28.13 异步模式的串行接收的流程图例子 (1)

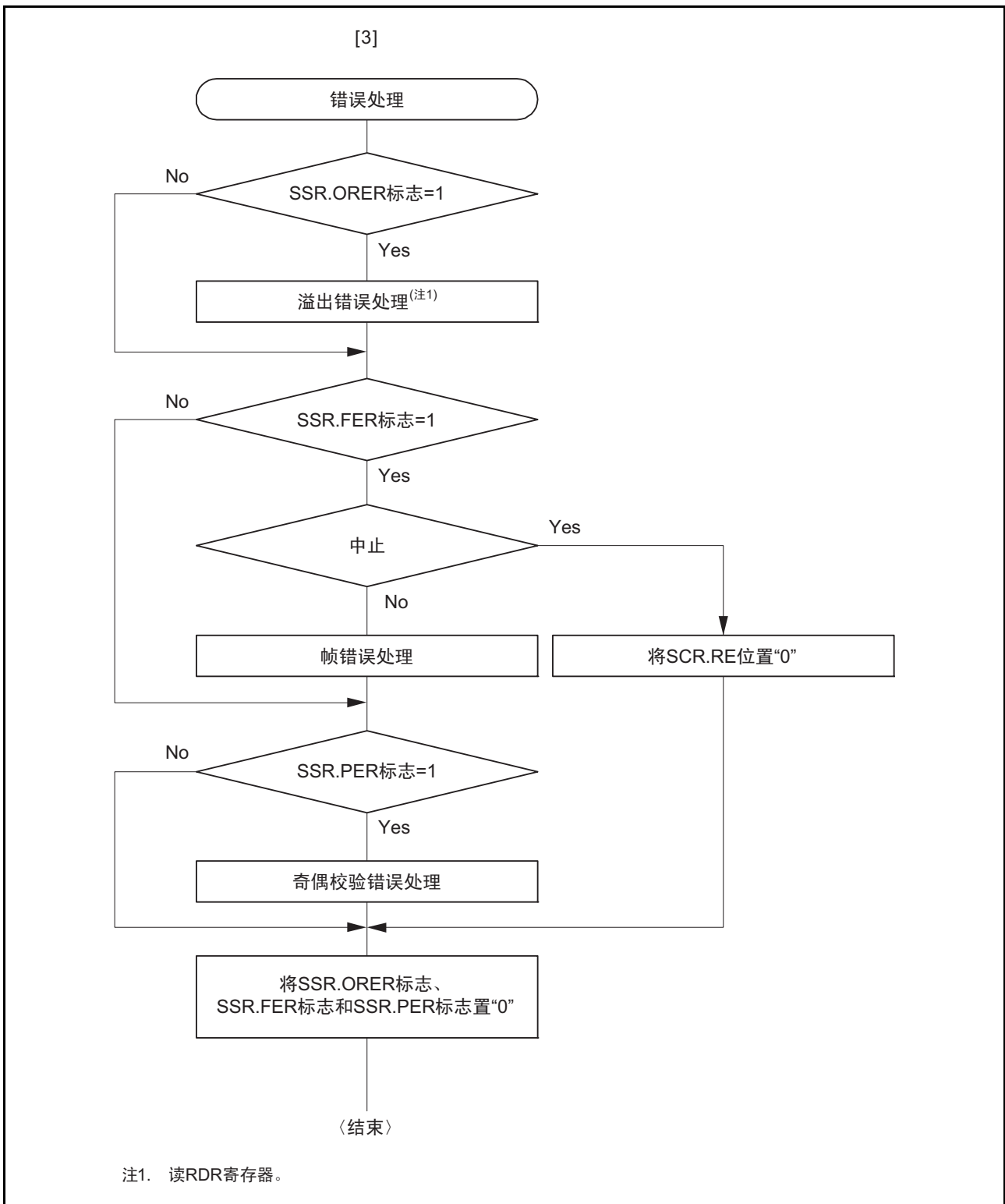


图 28.14 异步模式的串行接收的流程图例子 (2)

28.4 多处理器通信功能

如果使用多处理器通信功能，就能通过附加多处理器位的异步串行通信，在多个处理器之间共享通信线路进行数据的发送和接收。在多处理器通信中，给接收站分配各自特有的 ID 码。串行通信周期由指定接收站的 ID 发送周期和指定接收站的数据发送周期构成，用多处理器位区分 ID 发送周期和数据发送周期。当多处理器位是“1”时，为 ID 发送周期；当多处理器位是“0”时，为数据发送周期。使用多处理器格式的处理器之间的通信例子如图 28.15 所示。发送站首先发送多处理器位为“1”的接收站 ID 码，接着发送多处理器位为“0”的发送数据。如果接收站接收到多处理器位为“1”的通信数据，就将接收数据和本站的 ID 比较，如果相同，就继续接收被发送的通信数据；如果不相同，就在再次接收到下一个多处理器位为“1”的通信数据之前跳读通信数据。

SCI 为了支持此功能，设有 SCR.MPIE 位。如果将 MPIE 置“1”，就在接收到多处理器位为“1”的数据之前，禁止将接收数据从 RSR 寄存器传送到 RDR 寄存器，禁止检测接收错误并且禁止将 SRR. ORER 和 SRR.FER 各状态标志置位。如果接收到多处理器位为“1”的字符，就在 SSR.MPBT 位被置“1”的同时自动清除 SCR.MPIE，然后返回到通常的接收运行状态。此时，如果 SCR.RIE 位为“1”，就产生 RXI 中断。

在指定多处理器格式时，奇偶校验位的指定无效。除此以外与通常的异步模式相同，多处理器通信时的时钟也和通常的异步模式相同。

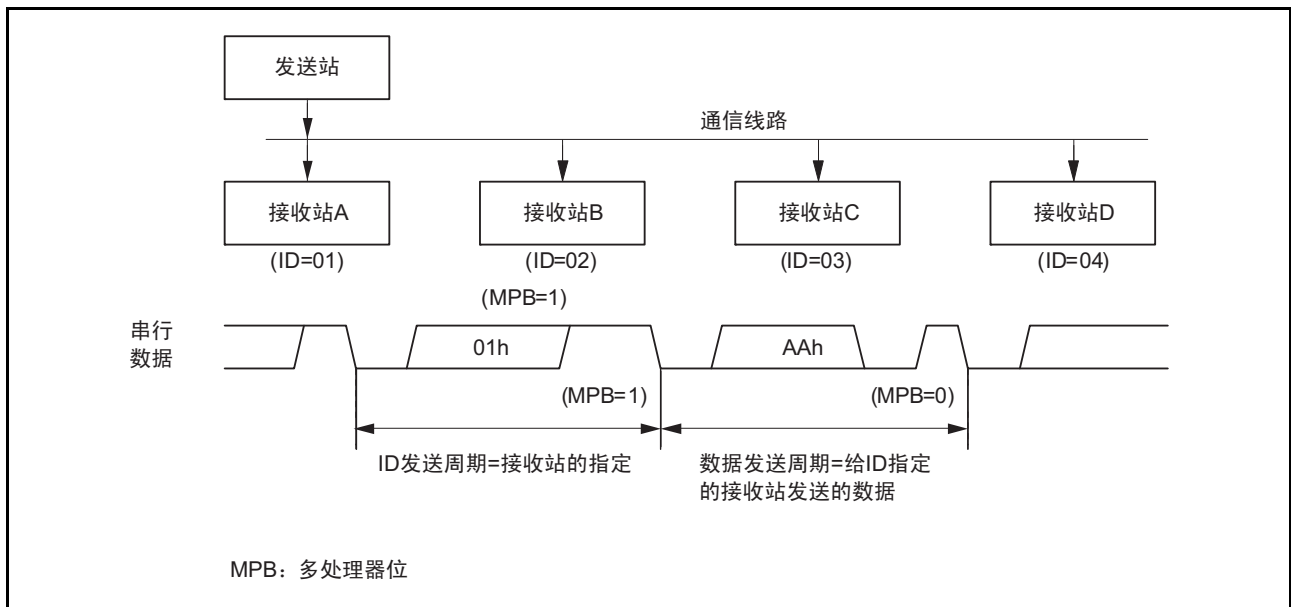


图 28.15 使用多处理器格式的通信的例子（将数据“AAh”发送到接收站 A 的例子）

28.4.1 多处理器串行数据的发送

多处理器数据处理的流程图例子如图 28.16 所示。必须在 ID 发送周期中将 SSR.MPBT 位置“1”后发送 ID 码，在数据发送周期中将 SSR.MPBT 位置“0”后发送数据。其他运行和异步模式的运行相同。

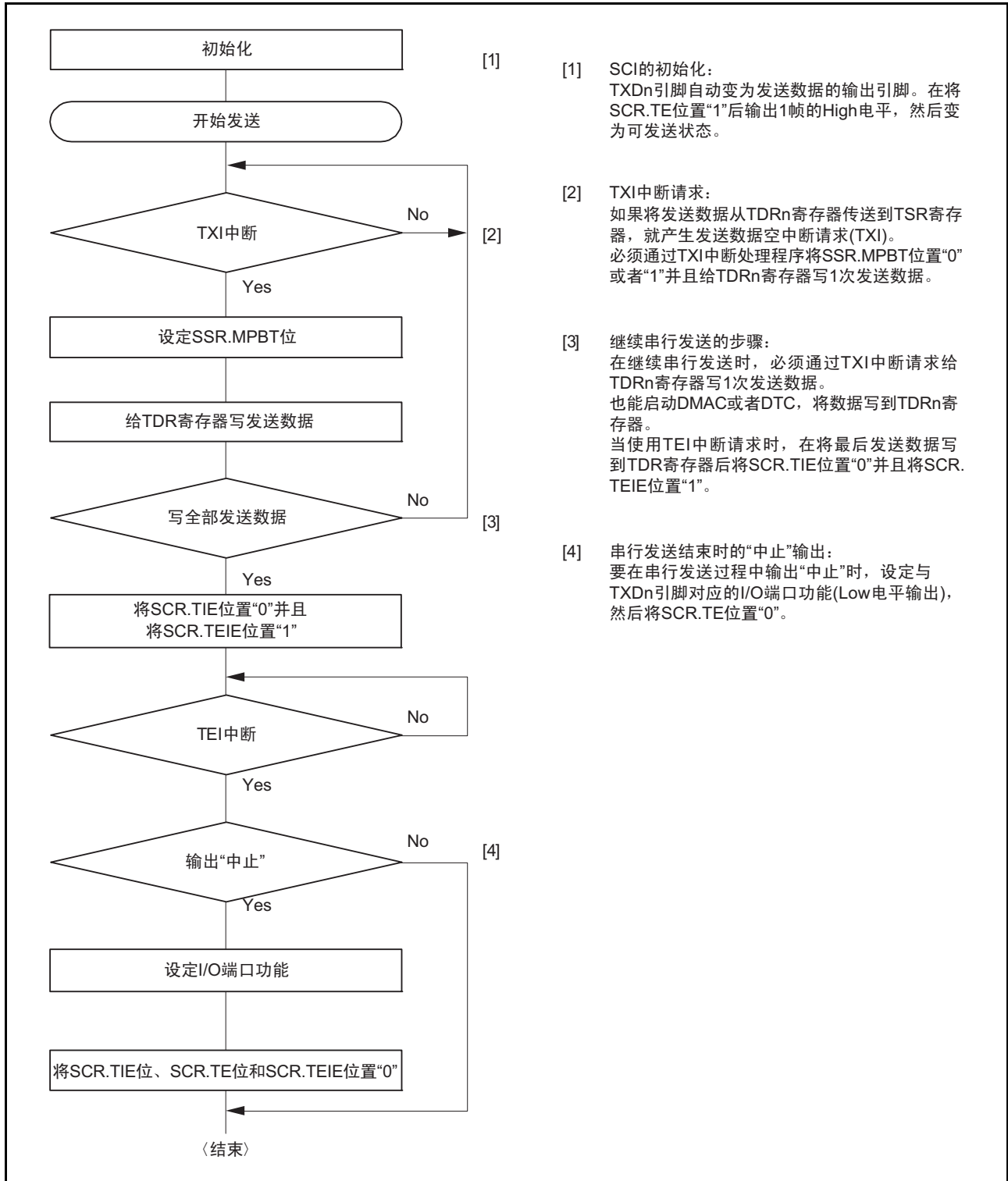


图 28.16 多处理器串行发送的流程图例子

28.4.2 多处理器串行数据的接收

多处理器数据接收的流程图例子如图 28.18 和图 28.19 所示。如果将 SCR.MPIE 位置“1”，就在接收到多处理器位为“1”的通信数据之前跳读通信数据。如果接收到多处理器位为“1”的通信数据，就将接收数据传送到 RDR 寄存器，此时产生 RXI 中断请求。其他运行和异步模式的运行相同。

接收时的运行例子如图 28.17 所示。

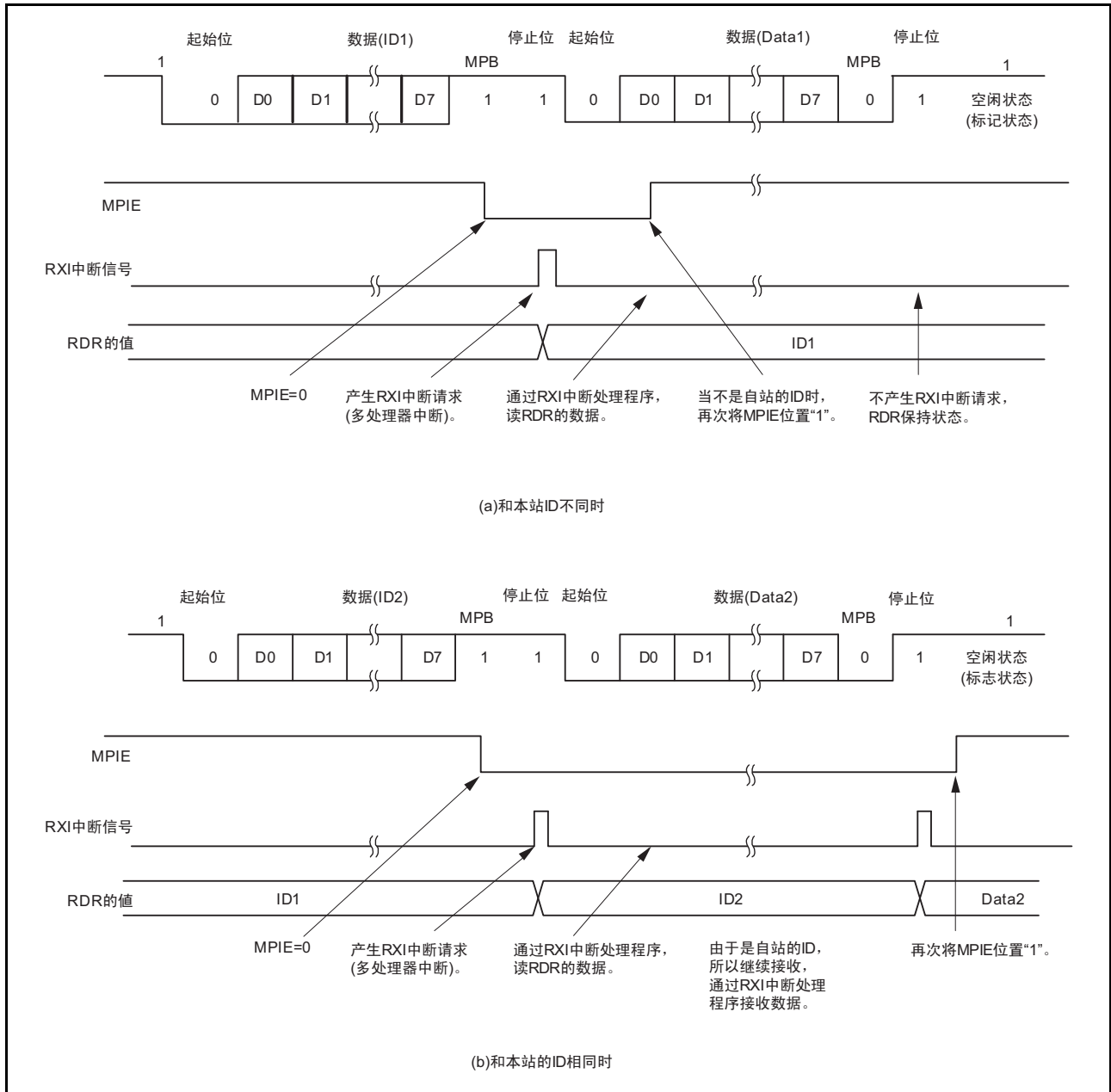


图 28.17 SCI 接收时的运行例子 (8 位数据 / 有多处理器位 / 1 个停止位的例子)

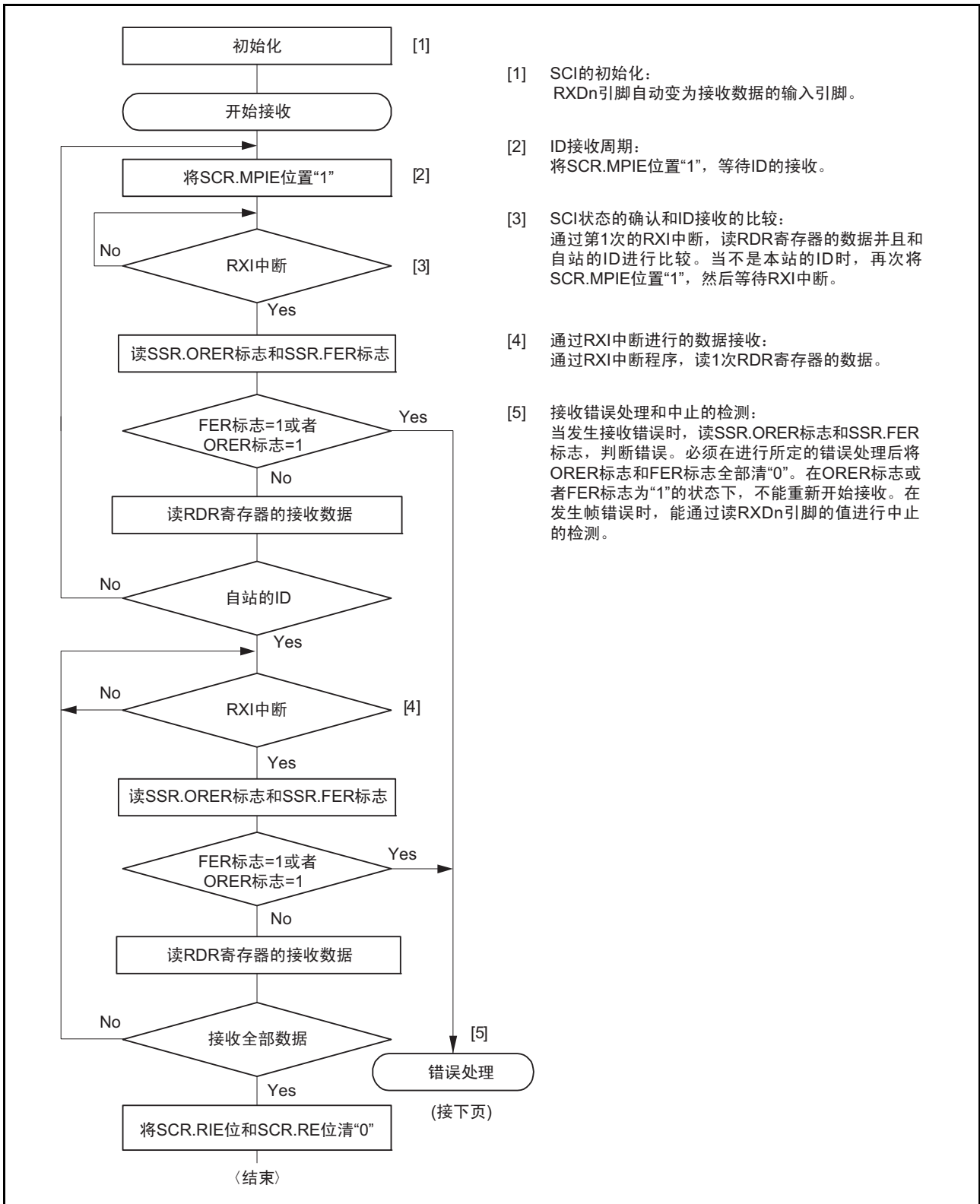


图 28.18 多处理器串行接收的流程图例子 (1)

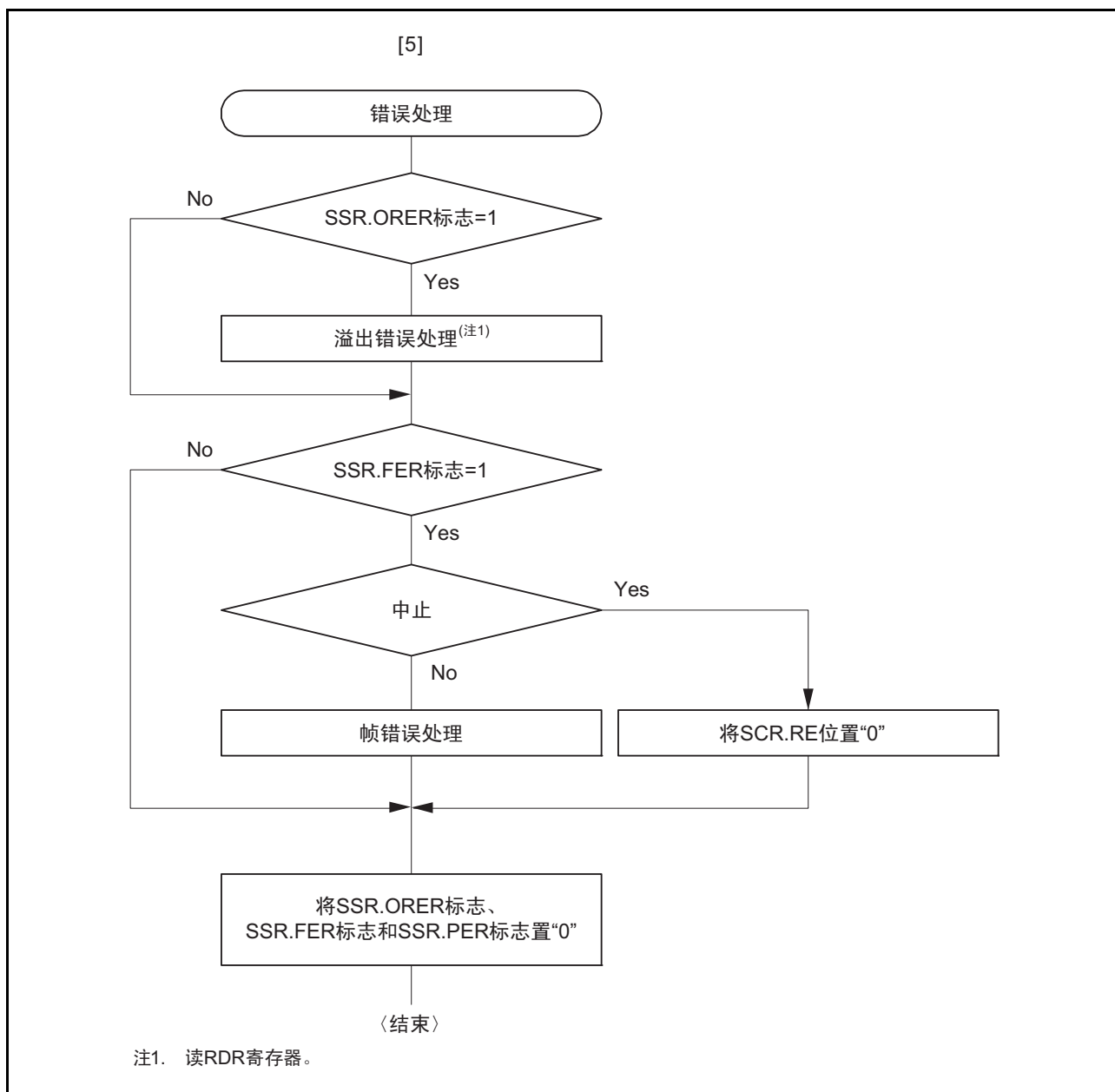


图 28.19 多处理器串行接收的流程图例子 (2)

28.5 时钟同步模式的运行

时钟同步串行通信的数据格式如图 28.20 所示。

在时钟同步模式中，与时钟脉冲同步发送和接收数据。通信数据的 1 个字符由 8 位数据构成，在时钟同步模式中，不能附加奇偶校验位。

SCI 在发送数据时，从同步时钟的下降沿开始到下一个下降沿前输出数据。在接收数据时，与时钟的上升沿同步接收数据。输出 8 位数据后的通信线路保持最后 1 位的输出状态。

SCI 内部的发送部和接收部各自独立，因此能通过共享时钟进行全双工通信。因为发送部和接收部都为双缓冲结构，所以能在发送时写下一个发送数据，通过在接收时读前一个接收数据，进行连续的接收和发送。

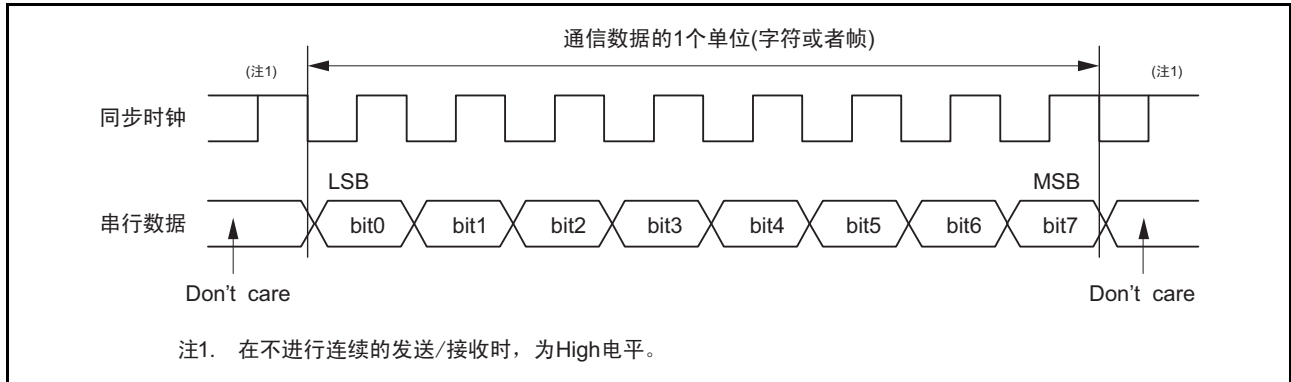


图 28.20 时钟同步串行通信的数据格式 (LSB first)

28.5.1 时钟

能通过设定SCR.CKE[1:0]位，选择内部波特率发生器生成的内部时钟或者SCKn引脚输入的外部同步时钟。

在通过内部时钟运行时，从SCKn引脚输出同步时钟。同步时钟在进行1个字符的发送和接收时输出8个脉冲，而在不进行发送和接收时固定为High电平。但是，当只进行接收时，在CTS功能有效并且CTS#引脚的输入为High电平或者发生溢出错误或者将SCR.RE位置“0”前输出同步时钟。在CTS功能有效的情况下，如果在结束帧的接收时CTS#引脚的输入为High电平，就停止输出同步时钟。

28.5.2 CTS 和 RTS 功能

CTS功能是在选择内部时钟的情况下使用CTS#引脚的输入进行发送和接收开始控制的功能。如果将SPMR.CTSE位置“1”，CTS功能就有效。

在CTS功能有效时并且只在CTS#引脚的输入为Low电平时，开始发送和接收。

RTS功能是在选择外部同步时钟的情况下使用RTS#引脚的输出进行发送和接收开始请求的功能。如果处于能进行串行通信的状态，就输出Low电平。输出Low电平或者High电平的条件如下：

[为 Low 电平的条件]

当满足以下全部条件时

- SCR.RE位或者SCR.TE位为“1”。
- 不处于发送和接收状态。
- 在读取前没有接收数据（SCR.RE位为“1”的情况）。
- 写完发送数据（SCR.TE位为“1”的情况）。
- SSR.ORER标志为“0”。

[为 High 电平的条件]

当不满足 Low 电平的条件时

28.5.3 SCI 的初始化 (时钟同步模式)

必须在发送和接收数据前给 SCR 寄存器写初始值“00h”，并且根据图 28.21 的流程图例子进行初始化。必须在对 SCR 寄存器进行初始化后更改运行模式或者通信格式。

必须注意: 即使将 SCR.RE 位置“0”，也不对 SSR.ORER 标志、SSR.FER 标志、SSR.PER 标志或者 RDR 寄存器进行初始化。

必须注意: 如果将 SCR.TE 位从“1”置为“0”或者从“0”置为“1”，就在 SCR.TIE 位为“1”时产生 TXI 中断。

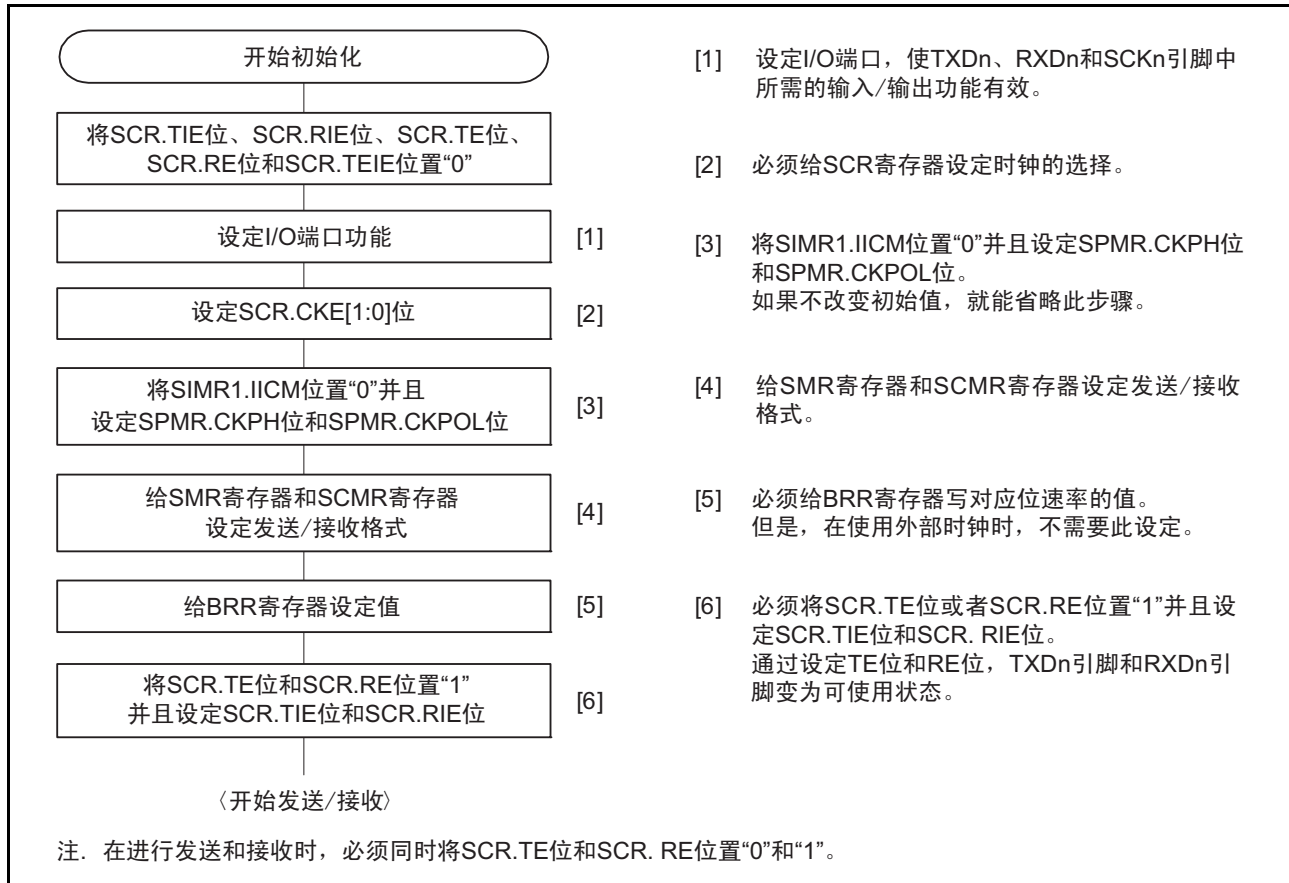


图 28.21 SCI 的初始化流程图例子 (时钟同步模式)

28.5.4 串行数据的发送 (时钟同步模式)

时钟同步模式的串行发送例子如图 28.22 所示。

在发送串行数据时，SCI 的运行如下：

1. 如果通过 TXI 中断处理程序给 TDR 寄存器写数据，SCI 就将数据从 TDR 寄存器传送到 TSR 寄存器。在开始发送时，通过将 SCR.TIE 位置“1”后将 SCR.TE 位置“1”，或者用 1 条指令同时将这 2 个位置“1”，产生 TXI 中断请求。
2. 通过将数据从 TDR 寄存器传送到 TSR 寄存器，开始发送。此时，如果 SCR.TIE 位为“1”，就产生 TXI 中断请求。能通过此 TXI 中断处理程序，在上次传送的数据发送结束前给 TDR 寄存器写下一个发送数据，进行连续的发送。当使用 TEI 中断请求时，在 TXI 中断请求处理程序中给 TDR 寄存器写最后的发送数据，然后将 SCR.TIE 位置“0”（禁止 TXI 中断请求）并且将 SCR.TEIE 位置“1”（允许 TEI 中断请求）。
3. 如果设定为时钟输出模式，就与输出时钟同步从 TXDn 引脚输出 8 位数据。如果设定为外部时钟，就与输入时钟同步从 TXDn 引脚输出 8 位数据。当 SPMR.CTSE 位为“1”（允许 CTS 功能）时，在等到 CTS 输入信号变为 Low 电平后开始输出时钟。
4. 在发送最后 1 位数据时检查 TDR 寄存器的更新（写）。
5. 如果 TDR 寄存器已被更新，就将数据从 TDR 寄存器传送到 TSR 寄存器，开始下一帧的发送。
6. 如果 TDR 寄存器未被更新，就将 SSR.TEND 标志置“1”，并且保持最后 1 位的输出状态。此时，如果 SCR.TEIE 位为“1”，就产生 TEI 中断请求。SCKn 引脚被固定为 High 电平。

串行发送的流程图例子如图 28.23 所示。

不能在接收错误标志（SSR.ORER、SSR.FER、SSR.PER）被置“1”的状态下开始发送，必须在开始发送前将接收错误标志置“0”。必须注意：即使将 SCR.RE 位置“0”，也不清除接收错误标志。

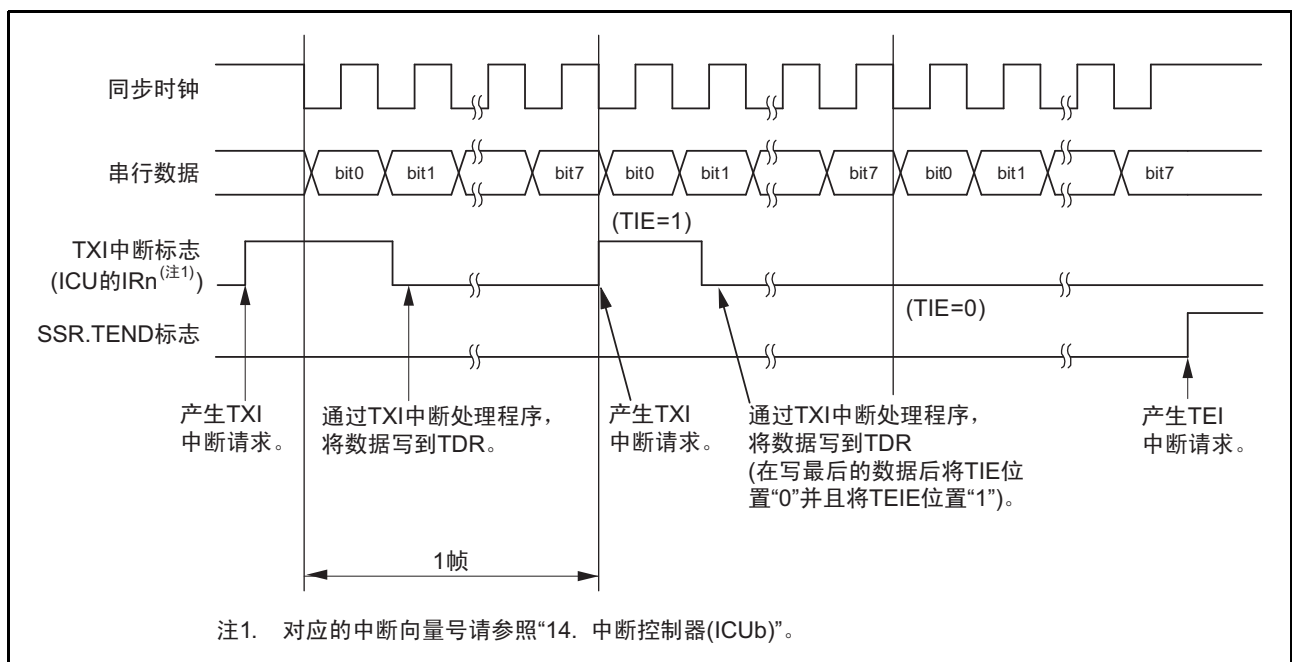


图 28.22 时钟同步模式的串行发送（正在发送～发送结束）例子

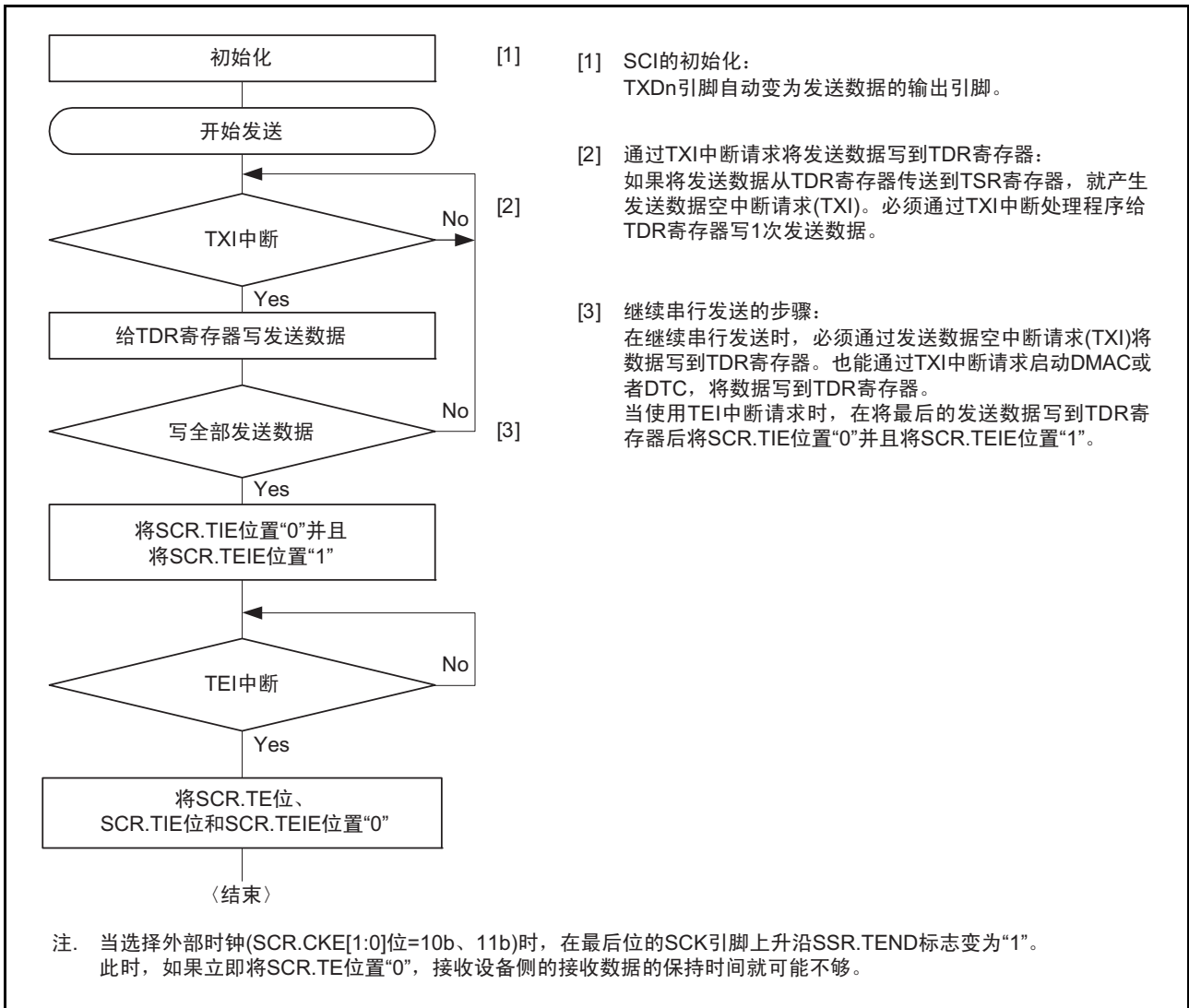


图 28.23 时钟同步模式的串行发送的流程图例子

28.5.5 串行数据的接收 (时钟同步模式)

时钟同步模式的串行接收例子如图 28.24 和图 28.25 所示。

在接收串行数据时, SCI 的运行如下:

1. 如果将SCR.RE位置“1”,就将RTS信号输出置为Low电平(使用RTS功能的情况)。
2. SCI与同步时钟的输入同步或者与输出同步对内部进行初始化后开始接收,并且将接收数据取到RSR寄存器。
3. 当发生溢出错误时,将SSR.ORER标志置位。此时,如果SCR.RIE位为“1”,就产生ERI中断请求。接收数据不传送到RDR寄存器。
4. 当正常接收时,将接收数据传送到RDR寄存器。此时,如果RIE位为“1”,就产生RXI中断请求。能通过此RXI中断处理程序,在下一个数据接收结束前读已被传送到RDR寄存器的接收数据,进行连续的接收。如果读被传送到RDR寄存器的接收数据,就将RTS信号输出置为Low电平(使用RTS功能的情况)。

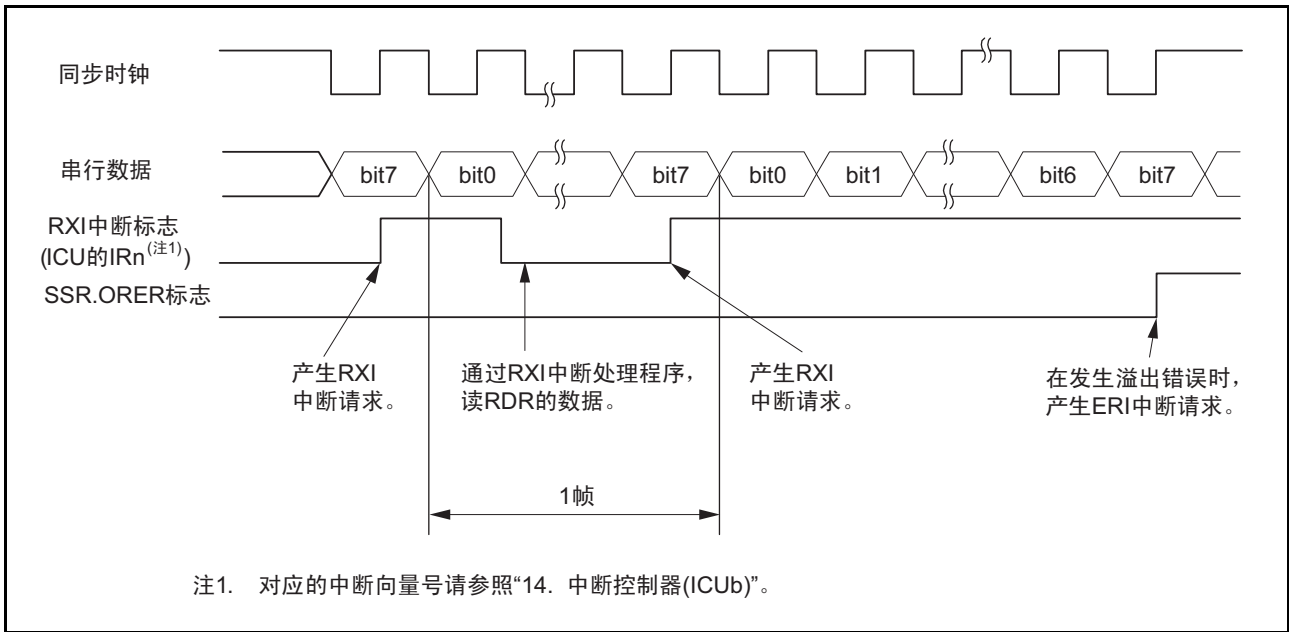


图 28.24 时钟同步模式的串行接收例子 (1) (不使用 RTS 功能的情况)

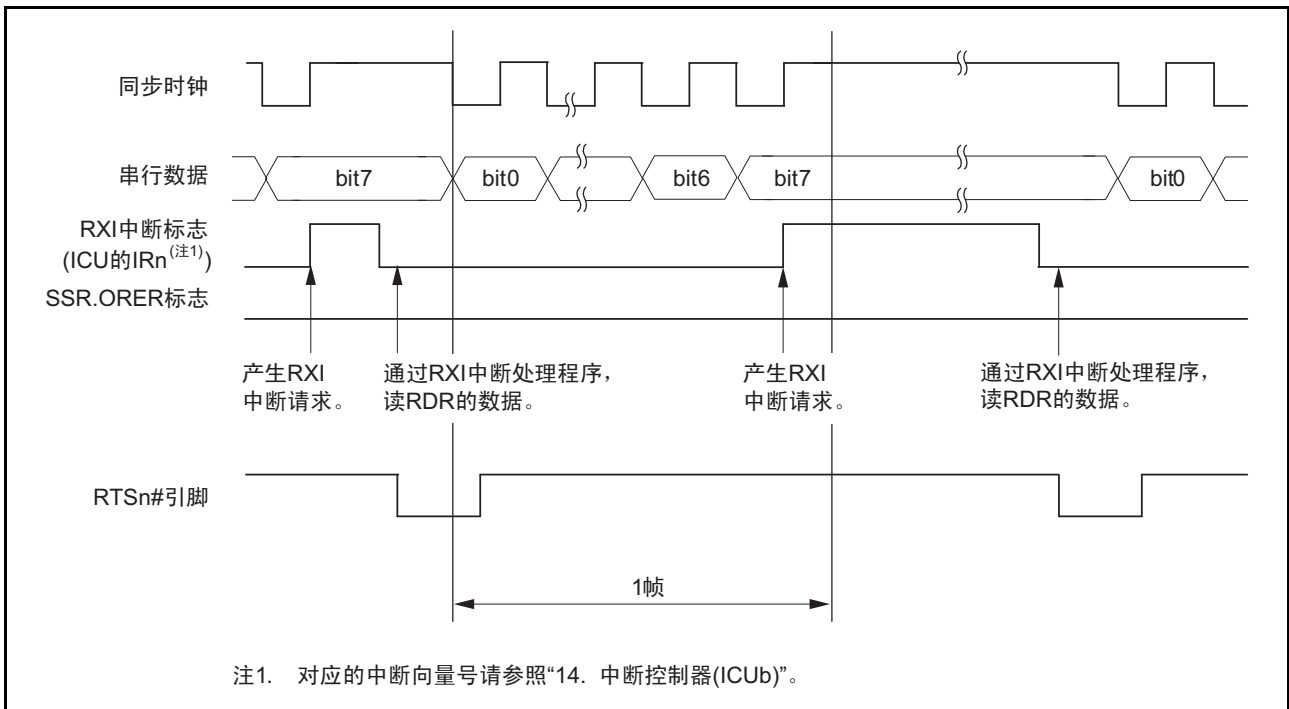


图 28.25 时钟同步模式的串行接收例子 (2) (使用 RTS 功能的情况)

在接收错误标志被置位的状态下，不能进行以后的发送和接收。因此，必须在继续接收前将 SSR.ORER 标志、SSR.FER 标志和 SSR.PER 标志置“0”。另外，必须在进行溢出错误处理时读 RDR 寄存器。

串行接收的流程图例子如图 28.26 所示。

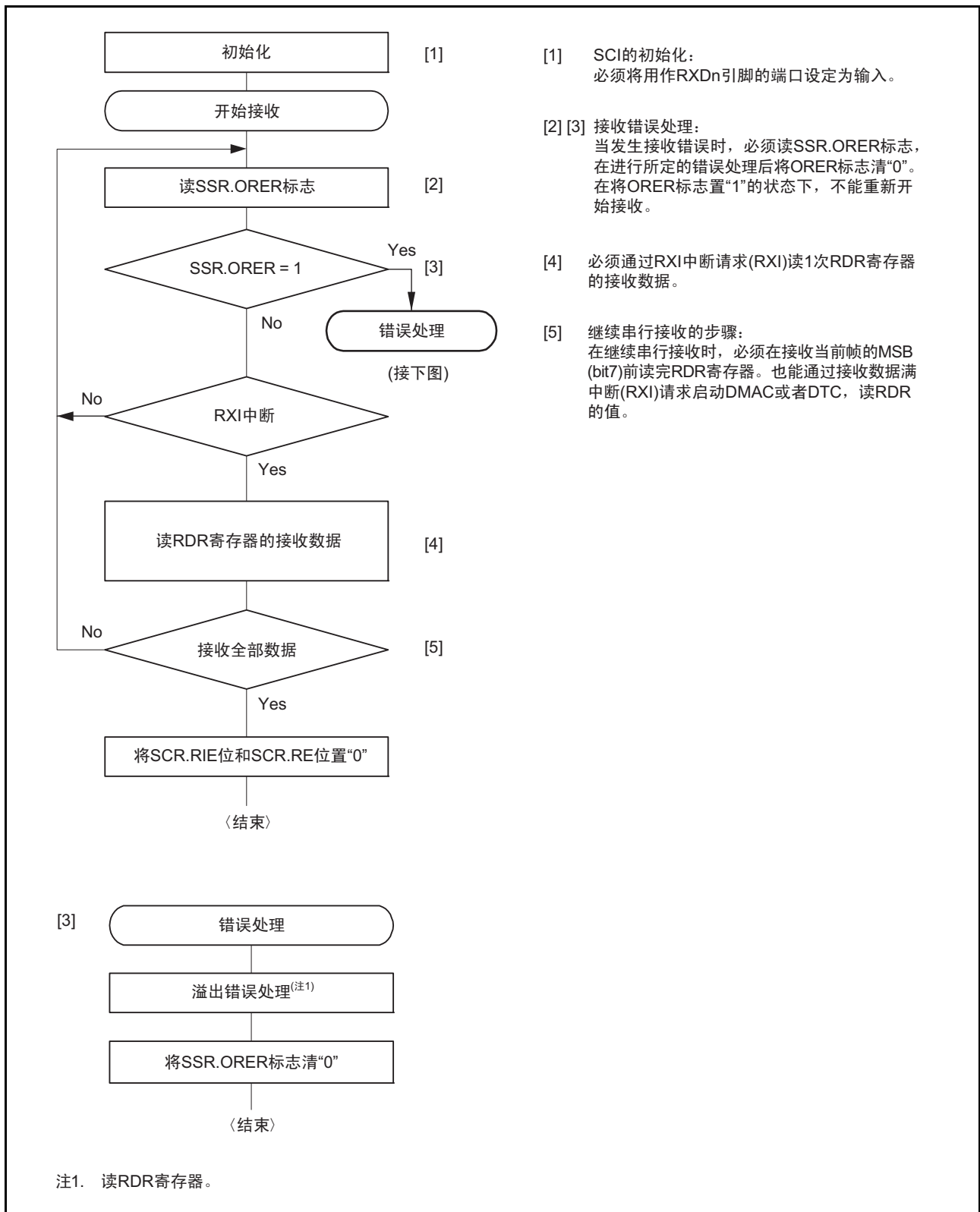


图 28.26 时钟同步模式的串行接收的流程图例子

28.5.6 串行数据的同时发送和接收 (时钟同步模式)

时钟同步模式的同时串行发送和接收的流程图例子如图 28.27 所示。

在对 SCI 进行初始化后, 必须按照以下步骤进行同时串行发送和接收。

要从发送转换为同时发送和接收时, 必须通过为“1”的 SSR.TEND 标志, 确认 SCI 处于发送结束状态。然后, 必须在对 SCR 寄存器进行初始化后, 用 1 条指令将 SCR.TIE 位、SCR.RIE 位、SCR.TE 位和 SCR.RE 位同时置“1”。

要从接收转换为同时发送和接收时, 必须在确认 SCI 处于接收结束状态后, 先将 SCR.RIE 位和 SCR.RE 位置“0”, 然后确认错误标志 (SSR.ORER、SSR.FER、SSR.PER) 已被置“0”, 最后用 1 条指令将 SCR.TIE 位、SCR.RIE 位、SCR.TE 位和 SCR.RE 位同时置“1”。

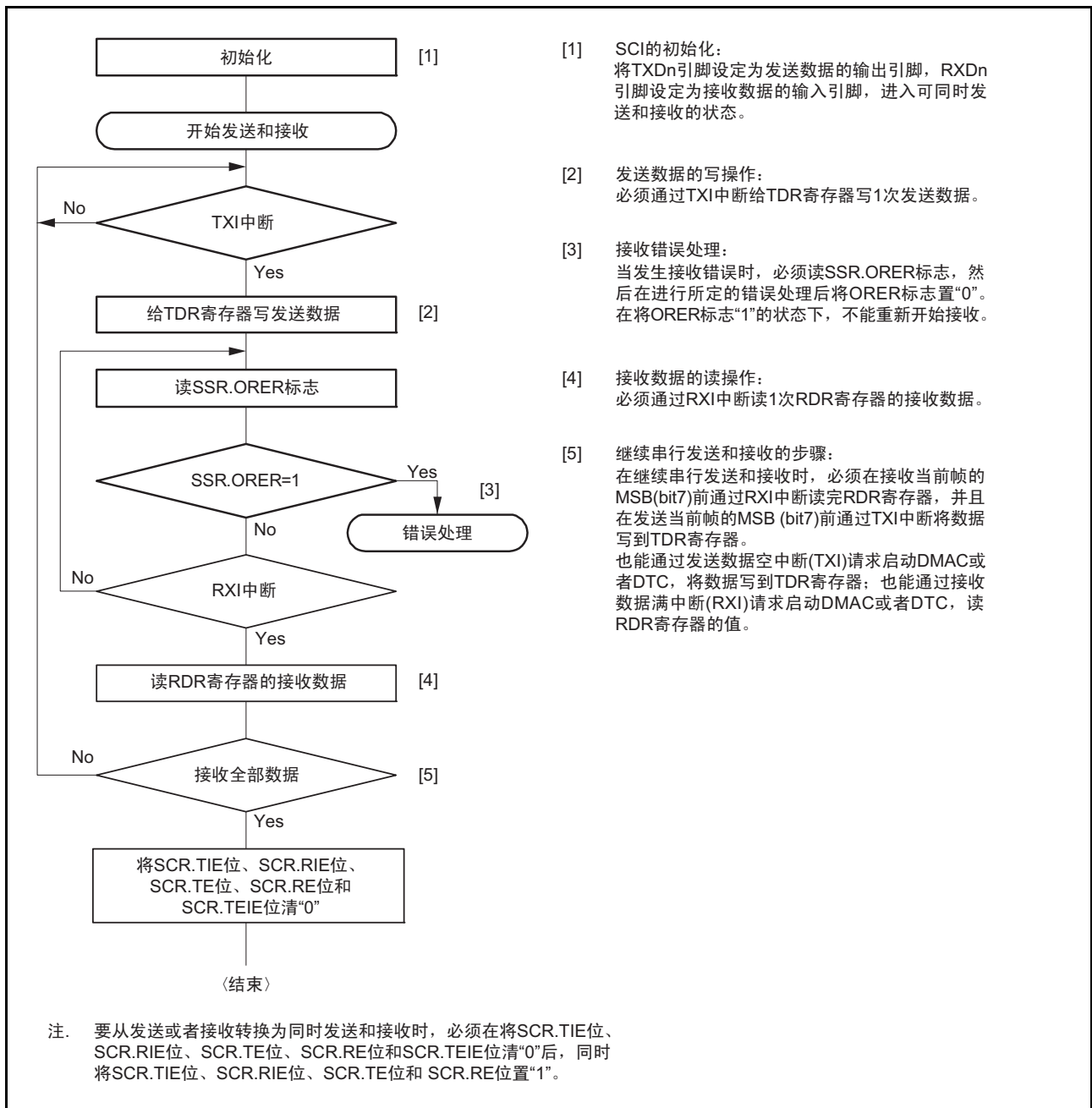


图 28.27 时钟同步模式的同时串行发送和接收的流程图例子

28.6 智能卡接口模式的运行

作为 SCI 的扩展功能，支持符合 ISO/IEC 7816-3 (Identification Card) 规格的智能卡 (IC 卡) 接口。
通过寄存器转换为智能卡接口模式。

28.6.1 连接例子

智能卡 (IC 卡) 的连接例子如图 28.28 所示。

IC 卡是指通过 1 条数据传送线进行发送和接收，因此必须连接 TXDn 引脚和 RXDn 引脚并且通过电阻将数据传送线上拉到电源 VCC 侧。

如果在不连接 IC 卡的状态下将 SCR.TE 位和 SCR.RE 位都置“1”，就能将发送和接收闭合连接进行自诊断。

如果给 IC 卡提供由 SCI 生成的时钟，就必须将 SCKn 引脚的输出信号输入到 IC 卡的 CLK 引脚。

能将 RX210 群的输出端口用于复位信号的输出。

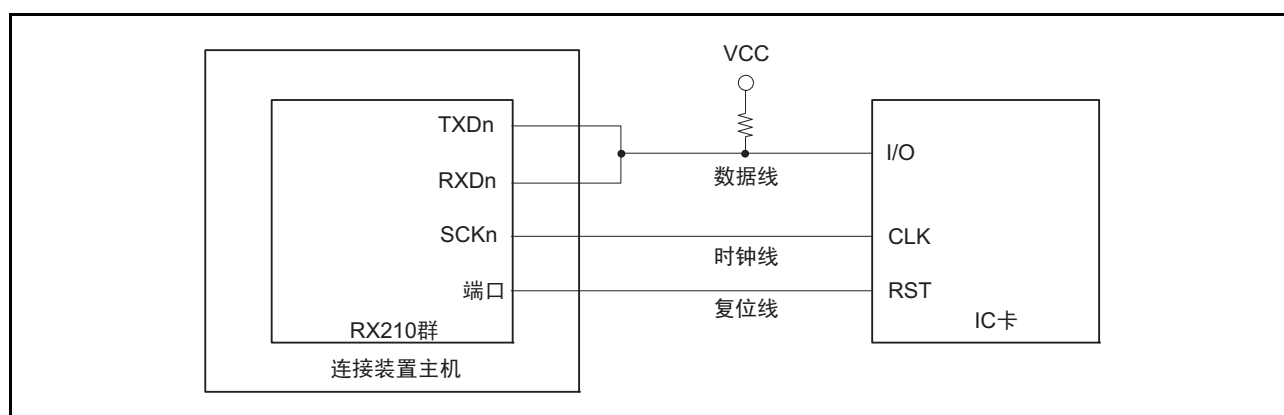


图 28.28 智能卡 (IC 卡) 的连接例子

28.6.2 数据格式 (块传送模式除外)

智能卡接口模式的发送 / 接收格式如图 28.29 所示。

- 在异步模式中，1 帧由 8 位数据和奇偶校验位构成。
- 在发送时，从奇偶校验位结束到下一帧的开始至少需要 2etu (Elementary Time Unit: 1 位传送时间) 的保护时间。
- 如果在接收时检测到奇偶校验错误，就在经过 10.5etu 后，从起始位输出 1etu 期间的错误信号 (Low 电平)。
- 如果在发送时对错误信号进行采样，就在至少经过 2etu 后，自动重新发送相同的数据。

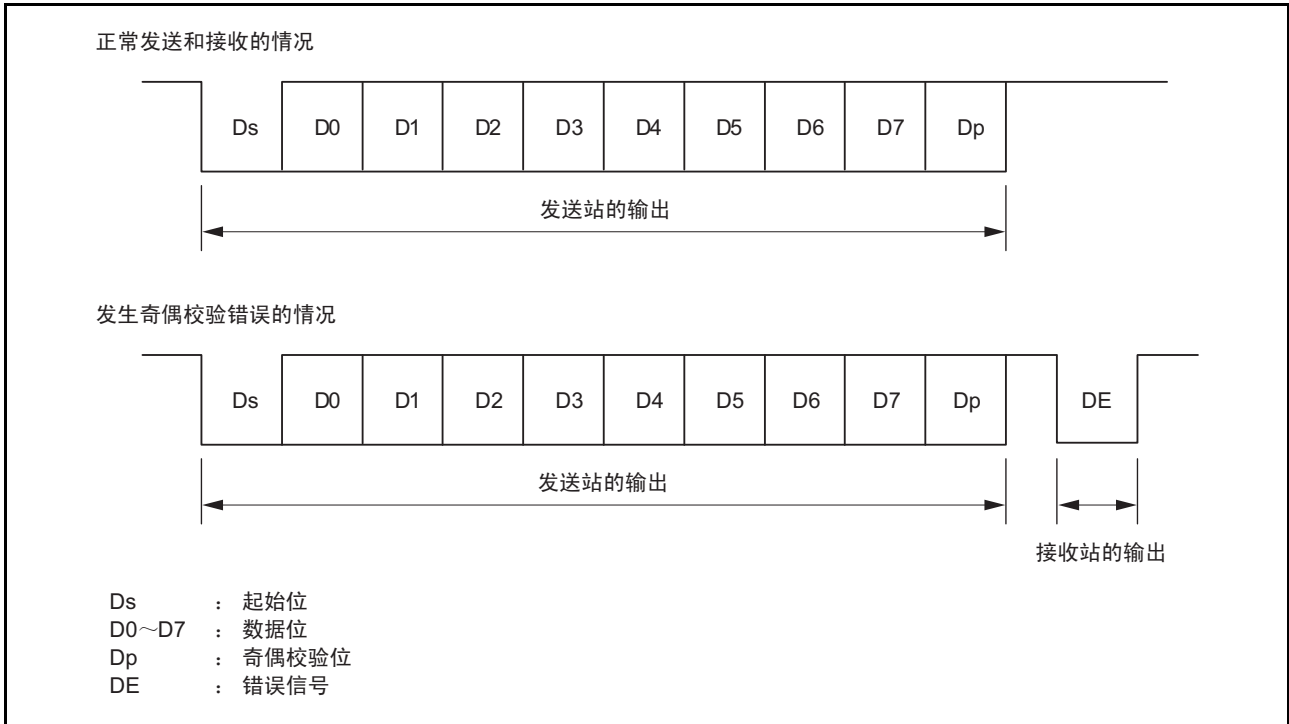


图 28.29 智能卡接口模式的数据格式

正向协议型和反向协议型这 2 种 IC 卡的发送和接收如下：

(1) 正向协议型

正向协议型如图 28.30 所示的开始字符例子，使逻辑 1 电平对应状态 Z，使逻辑 0 电平对应状态 A，以 LSB first 进行发送和接收。图 28.30 中的开始字符数据为“3Bh”。

对于正向协议型，必须将 SCMR.SDIR 位和 SCMR.SINV 位都置“0”。为了根据智能卡的规格进行偶校验，必须将 SMR.PM 位置“0”。

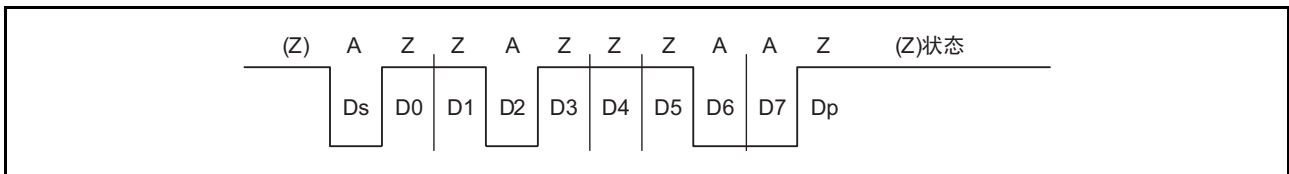


图 28.30 正向协议 (SCMR.SDIR 位 = 0、SCMR.SINV 位 = 0、SMR.PM 位 = 0)

(2) 反向协议型

反向协议型使逻辑 1 电平对应状态 A，使逻辑 0 电平对应状态 Z，以 MSB first 进行发送和接收。图 28.31 中的开始字符数据为“3Fh”。

对于反向协议型，必须将 SCMR.SDIR 位和 SCMR.SINV 位都置“1”。在根据智能卡的规格进行偶校验时，奇偶校验位为逻辑 0，对应状态 Z。在 RX210 群中，SINV 位只将数据位 D7 ~ D0 取反。因此，在发送和接收时，必须将 SMR.PM 位置“1”，将奇偶校验位取反。

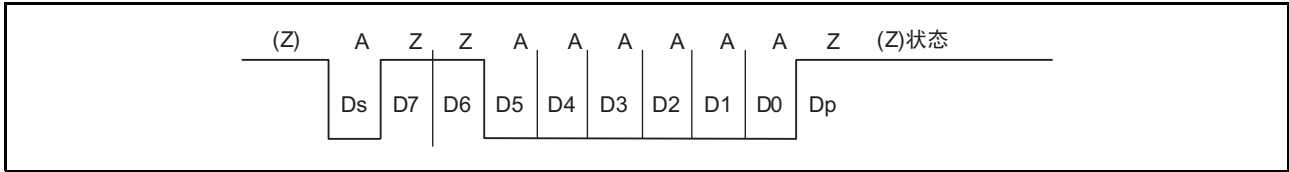


图 28.31 反向协议 (SCMR.SDIR 位 =1、SCMR.SINV 位 =1、SMR.PM 位 =1)

28.6.3 块传送模式

块传送模式和一般的智能卡接口模式比较，有以下不同点：

- 在接收时进行奇偶校验，即使检测到错误，也不输出错误信号。因为 SSR.PER 标志被置位，所以必须在接收下一帧的奇偶校验位前清除此位。
- 在发送时，从奇偶校验位结束到下一帧的开始至少需要 1etu 的保护时间。
- 因为不重新进行发送，所以在开始发送后的 11.5etu 后，将 SSR.TEND 标志置位。
- 和一般的智能卡接口模式一样，SSR.ERS 标志表示错误信号的状态。因为不发送和接收错误信号，所以 SSR.ERS 标志总是为“0”。

28.6.4 接收数据的采样时序和接收容限

能用于智能卡接口模式的发送 / 接收时钟只有内部波特率发生器生成的内部时钟。

在智能卡接口模式中，根据 SCMR.BCP2 位和 SMR.BCP[1:0] 位的设定，SCI 通过频率为 32 倍、64 倍、372 倍、256 倍、93 倍、128 倍、186 倍或者 512 倍（在一般的异步模式中固定为 16 倍）位速率的基本时钟运行。

在接收时，通过基本时钟对起始位的下降沿进行采样，并且与内部取得同步。如图 28.32 所示，通过在基本时钟的 16 个、32 个、186 个、128 个、46 个、64 个、93 个、256 个周期的上升沿分别对接收数据进行采样，在各位的中央取数据。能用以下表达式表示此时的接收容限：

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

- M：接收容限(%)
- N：对应时钟的位速率比(N=32、64、372、256)
- D：时钟的占空比(D=0~1.0)
- L：帧长(L=10)
- F：时钟频率的偏差绝对值

假设在上述的表达式中，F=0，D=0.5，N=372，则接收容限如下所示：

$$M = \{0.5 - 1 / (2 \times 372)\} \times 100 [\%] = 49.866\%$$

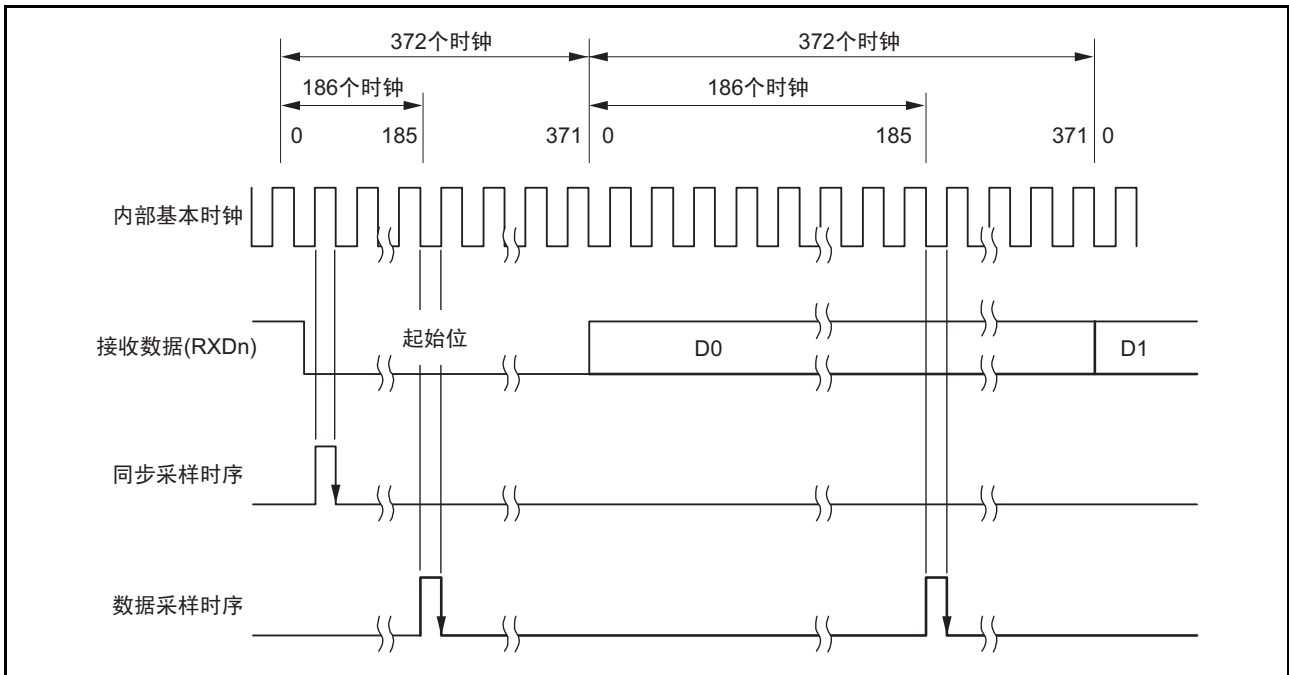


图 28.32 智能卡接口模式的接收数据的采样时序（使用 372 倍时钟的情况）

28.6.5 SCI 的初始化（智能卡接口模式）

在发送和接收数据前，必须按照以下步骤对 SCI 进行初始化。在从发送模式转换为接收模式或者从接收模式转换为发送模式时，都需要进行初始化。

1. 给 SCR 寄存器写初始值“00h”。
2. 必须设定 I/O 端口，使 TXDn 引脚、RXDn 引脚和 SCKn 引脚中所需的输入/输出功能有效。
3. 必须将 SSR 寄存器的错误标志（ORER、ERS、PER）置“0”。
4. 必须将 SIMR1.IICM 位置“0”并且将 SPMR.CKPH 位和 SPMR.CKPOL 位置“0”。
（如果不改变初始值，就能省略此步骤）
5. 必须设定 SMR.GM 位、SMR.BLK 位、SMR.PM 位、SMR.BCP[1:0] 位、SMR.CKS[1:0] 位和 SCMR.BCP2 位。此时，必须将 SMR.PE 位置“1”。
6. 必须设定 SCMR.SDIR 位、SCMR.SINV 位和 SCMR.SMIF 位。TXDn 引脚和 RXDn 引脚为高阻抗状态。
7. 给 BRR 寄存器设定与位速率对应的值。
8. 必须设定 SCR.CKE[1:0] 位。此时，必须将 SCR.TIE 位、SCR.RIE 位、SCR.TE 位、SCR.RE 位和 SCR.TEIE 位置“0”。
如果将 CKE[0] 位置“1”，就从 SCKn 引脚输出时钟。
9. 必须设定 SCR.TIE 位、SCR.RIE 位、SCR.TE 位和 SCR.RE 位。除了进行自诊断以外，必须将 TE 位和 RE 位同时置位。

要从接收模式转换为发送模式时，必须在确认接收已结束后从初始化开始进行，并且将 TE 位置“1”以及将 RE 位置“0”。能通过 RXI 中断请求、SSR.ORER 标志或者 SSR.PER 标志确认接收的结束。

要从发送模式转换为接收模式时，必须在确认发送已结束后从初始化开始进行，并且将 TE 位置“0”以及将 RE 位置“1”。能通过 SSR.TEND 标志确认发送的结束。

28.6.6 串行数据的发送 (块传送模式除外)

智能卡接口模式的串行发送包括错误信号的采样和重新发送处理, 因此和一般的串行通信接口模式的运行不同 (块传送模式除外)。发送时的重新传送如图 28.33 所示。

1. 如果在结束1帧的发送后对接收侧的错误信号进行采样, 就将SSR.ERS标志置“1”。此时, 如果SCR.RIE位为“1”, 就产生ERI中断请求。必须在对下一个奇偶校验位进行采样前清除ERS标志。
2. 在接收到错误信号的帧中, 不将SSR.TEND标志置位。再次将数据从TDR寄存器传送到TSR寄存器, 自动进行重新发送。
3. 如果接收侧没有返回错误信号, 就不将ERS标志置位。
4. 在判断出包括重新传送的1帧的发送已结束后, 将SSR.TEND标志置位。此时, 如果SCR.TIE位为“1”, 就产生TXI中断请求。通过给TDR寄存器写发送数据, 开始下一个数据的发送。

串行发送的流程图例子如图 28.35 所示。能通过 TXI 中断源启动 DTC 或者 DMAC, 自动进行上述的一系列处理。

在发送时, 如果将 SCR.TIE 位置“1”, 就在 SSR.TEND 标志被置“1”时产生 TXI 中断请求。如果预先将 TXI 中断请求设定为 DTC 或者 DMAC 的启动源, 就在通过 TXI 中断请求启动 DTC 或者 DMAC 后, 进行发送数据的传送。在通过 DTC 或者 DMAC 进行数据传送时, TEND 标志自动变为“0”。

如果发生错误, SCI 就自动重新发送相同的数据。在此期间, TEND 标志保持“0”的状态并且不启动 DTC 或者 DMAC。因此, 自动发送 SCI 以及 DTC 或者 DMAC 指定的字节数, 包括发生错误时的重新发送。但是, 在发生错误时不自动清除 ERS 标志, 因此必须预先将 RIE 位置“1”, 使在发生错误时产生 ERI 中断请求, 并且清除 ERS 标志。

在使用 DTC 或者 DMAC 进行发送和接收时, 必须先设定 DTC 或者 DMAC, 在设定为允许状态后进行 SCI 的设定。

有关 DTC 或者 DMAC 的设定方法, 请参照“16. DMA 控制器 (DMACA)”和“17. 数据传送控制器 (DTCa)”。

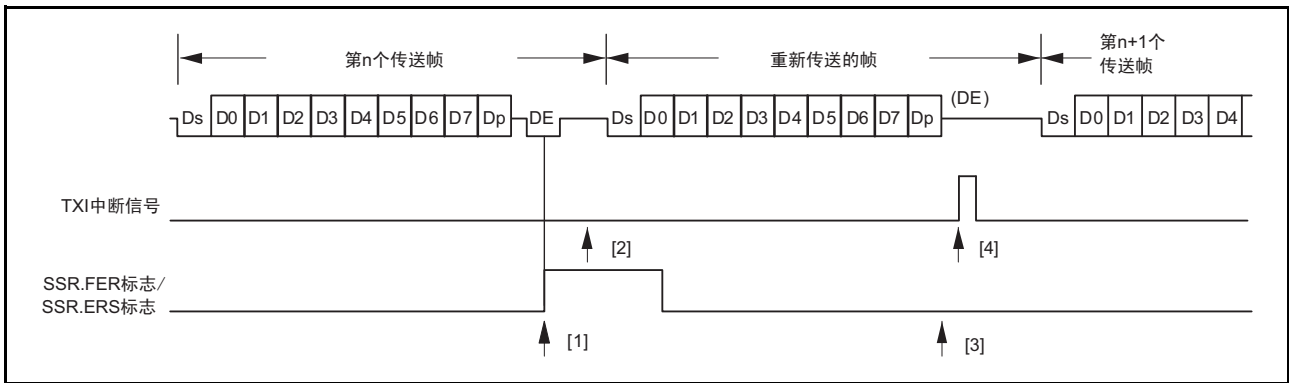


图 28.33 SCI 发送模式的重新传送 (发送时的重新传送)

SSR.TEND 标志的置位时序取决于 SMR.GM 位的设定，TEND 标志的产生时序如图 28.34 所示。

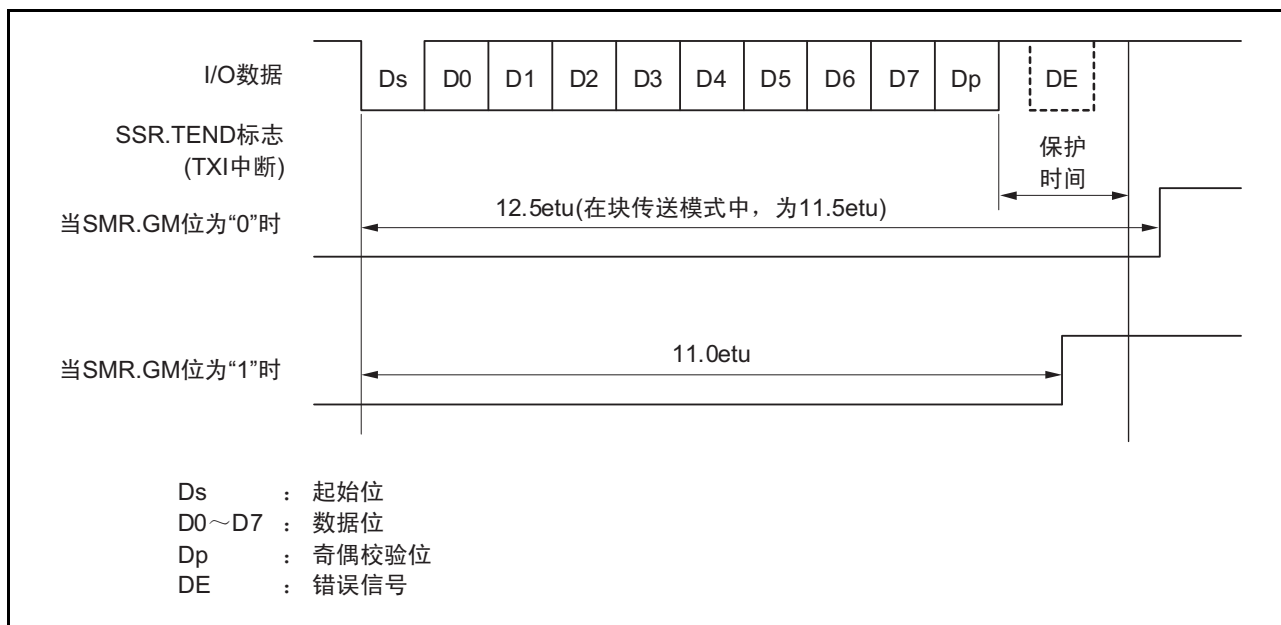


图 28.34 发送时的 SSR.TEND 标志的产生时序

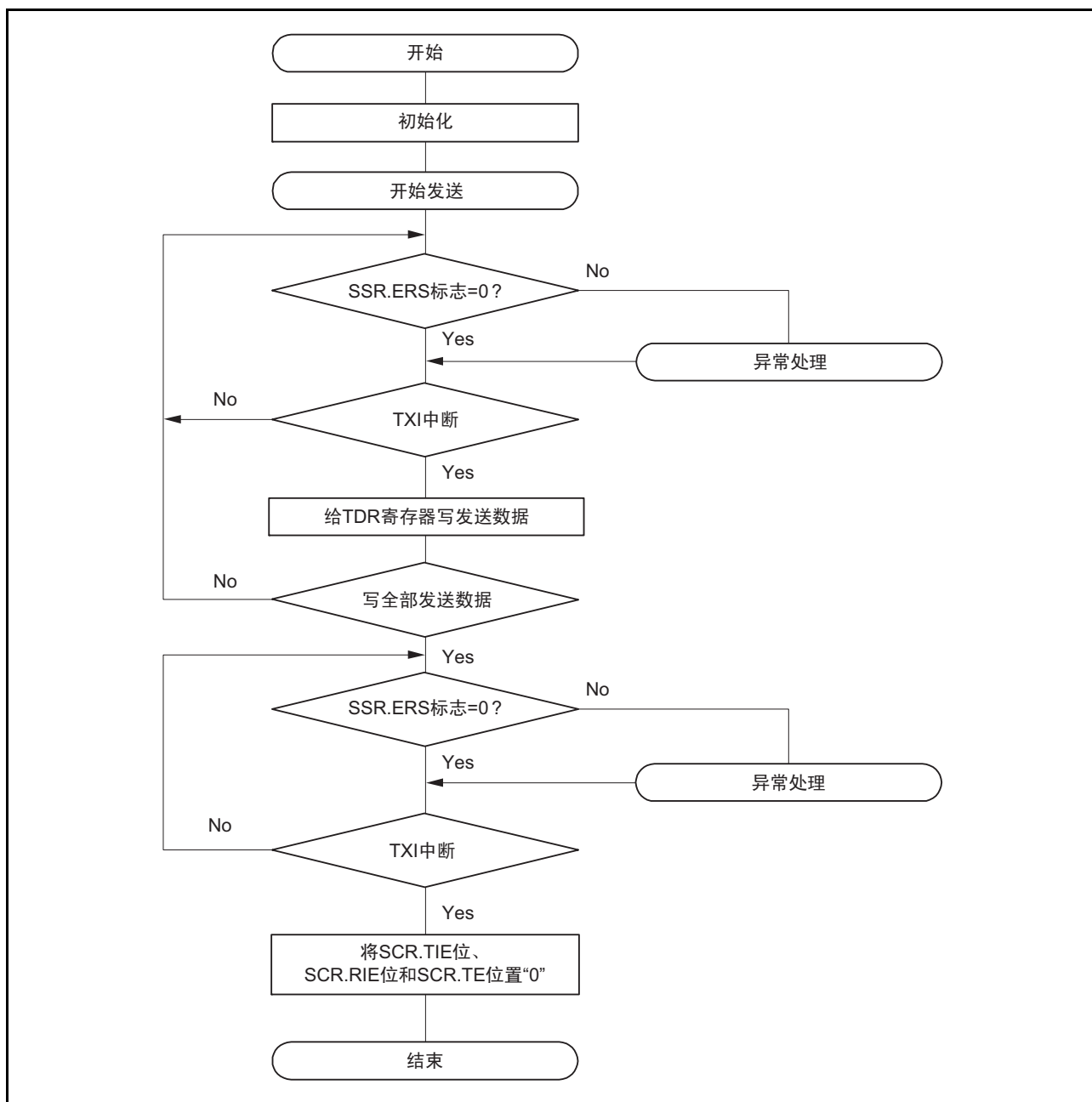


图 28.35 智能卡接口发送的流程图例子

28.6.7 串行接收（块传送模式除外）

智能卡接口模式的串行接收和串行通信接口模式的处理步骤相同。接收模式的重新传送如图 28.36 所示。

1. 如果在接收数据时检测到奇偶校验错误，就将 **SSR.PER** 标志置“1”。此时，如果 **SCR.RIE** 位为“1”，就产生 **ERI** 中断请求。必须在下一个奇偶校验位的采样时序前清除 **PER** 标志。
2. 对于检测到奇偶校验错误的帧，不发生 **RXI** 中断。
3. 如果检测不到奇偶校验错误，就不将 **SSR.PER** 标志置位。
4. 如果在判断出接收正常结束后 **RIE** 位为“1”，就产生 **RXI** 中断请求。

串行接收的流程图例子如图 28.37 所示。能通过 **RXI** 中断请求启动 **DTC** 或者 **DMAC**，自动进行上述的一系列处理。

如果在接收时预先将 **RIE** 位置“1”，就产生 **RXI** 中断请求。如果预先将 **RXI** 中断设定为 **DTC** 或者 **DMAC** 的启动源，就在通过 **RXI** 中断请求启动 **DTC** 或者 **DMAC** 后，进行接收数据的传送。

如果在接收时发生错误并且 **SSR.ORER** 标志或者 **SSR.PER** 标志被置“1”，就产生接收错误中断（**ERI**）请求，因此必须清除错误标志。因为在发生错误时不启动 **DTC** 或者 **DMAC** 而省略接收数据，所以只传送由 **DTC** 或者 **DMAC** 设定的字节数的接收数据。

即使在接收时发生奇偶校验错误并且 **PER** 标志被置“1”，也将接收数据传送到 **RDR** 寄存器，因此能读此数据。

注：有关块传送模式，请参照“28.3 异步模式的运行”。

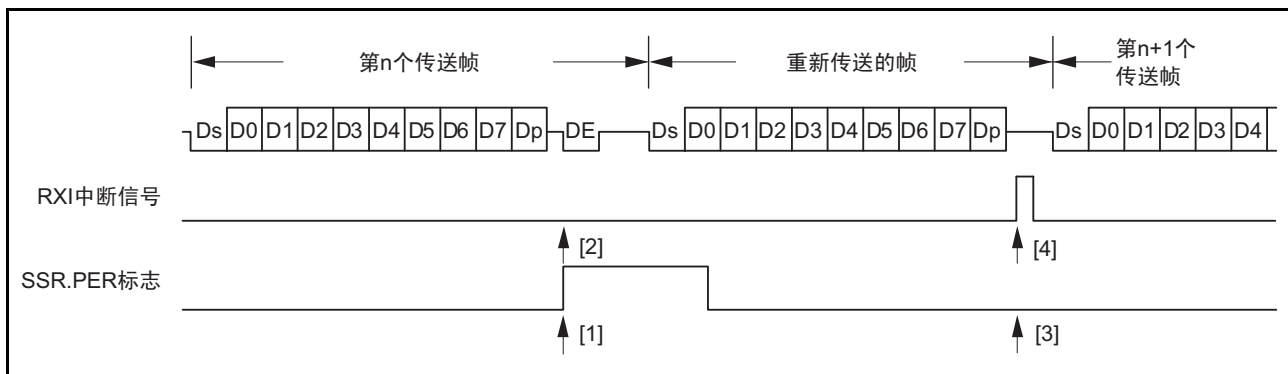


图 28.36 SCI 接收模式的重新传送（接收时的重新传送）

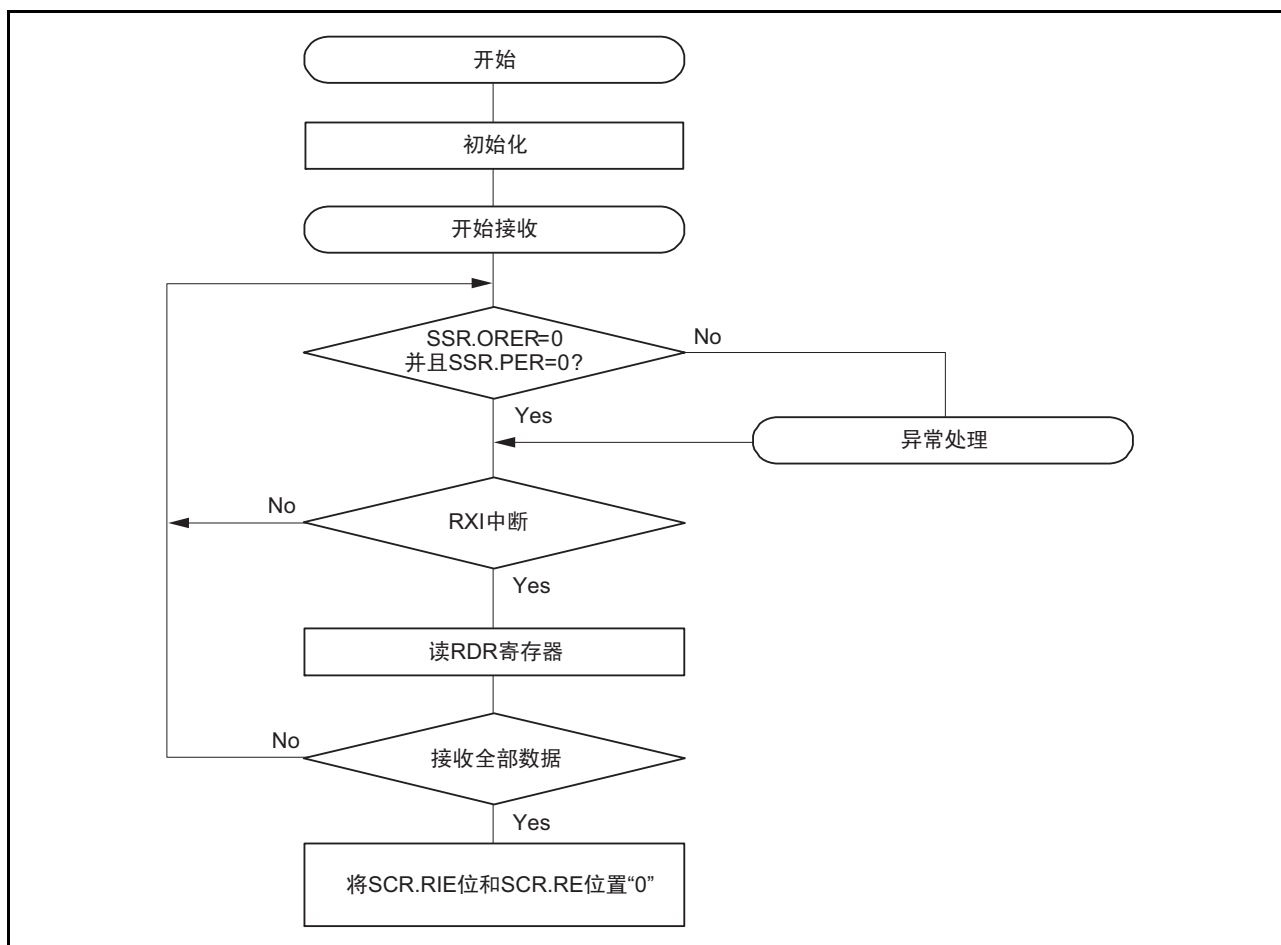


图 28.37 智能卡接口接收的流程图例子

28.6.8 时钟的输出控制

当 SMR.GM 位为“1”时，能通过 SCR.CKE[1:0] 位固定时钟输出。此时，能将时钟脉冲的最小宽度设定为指定的宽度。

时钟输出固定时序如图 28.38 所示，这是在 GM 位为“1”并且 CKE1 位为“0”的状态下控制 CKE0 位的例子。

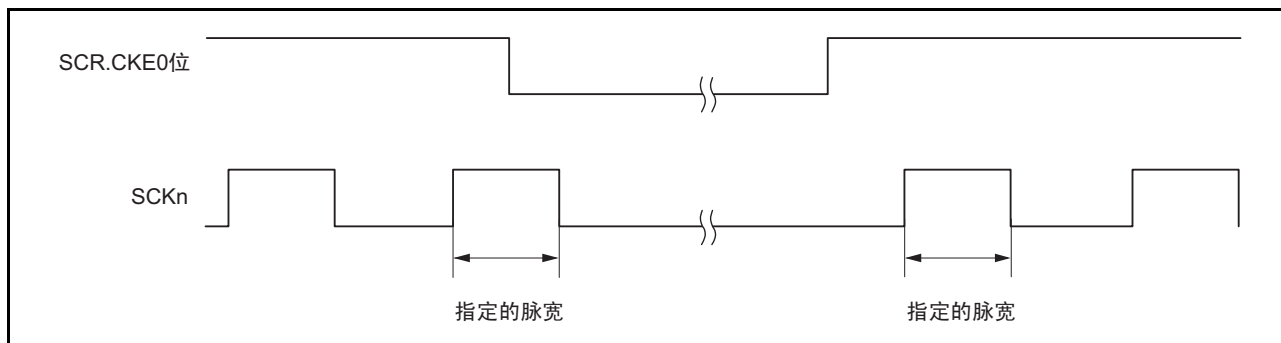


图 28.38 时钟输出固定时序

在接通电源、向软件待机模式转移或者从软件待机模式返回时，为了确保时钟的占空比，必须按照以下步骤进行处理：

(1) 电源接通时

为了从接通电源开始确保时钟的占空比，必须按照以下步骤进行转换处理：

1. 初始状态为端口输入的高阻抗状态。要固定电位时，必须使用上拉电阻或者下拉电阻。
2. 必须设定 SCR.CKE[1] 位和 I/O 端口功能，将 SCKn 引脚固定为指定的输出。
3. 必须设定 SMR 寄存器和 SCMR 寄存器，转换为智能卡接口模式的运行。
4. 必须将 SCR.CKE[0] 位置“1”，开始时钟的输出。

(2) 转换模式时

(a) 从智能卡接口模式转移到软件待机模式的情况

1. 必须设定 I/O 端口，使 SCKn 引脚变为软件待机模式中所需的输出固定状态的值。
2. 必须给 SCR.TE 位和 SCR.RE 位写“0”，停止发送和接收。
同时，必须给 SCR.CKE[1] 位设定软件待机时的输出固定状态的值。
3. 必须给 SCR.CKE[0] 位写“0”，停止时钟。
4. 必须等待 1 个串行时钟周期。在此期间，保持占空比并且以指定的电平固定时钟的输出。
5. 必须转移到软件待机状态。

(b) 从软件待机模式返回到智能卡接口模式的情况

6. 必须解除软件待机状态。
7. 必须将 SCR.CKE[0] 位置“1”，输出时钟。开始以正常的占空比生成信号。

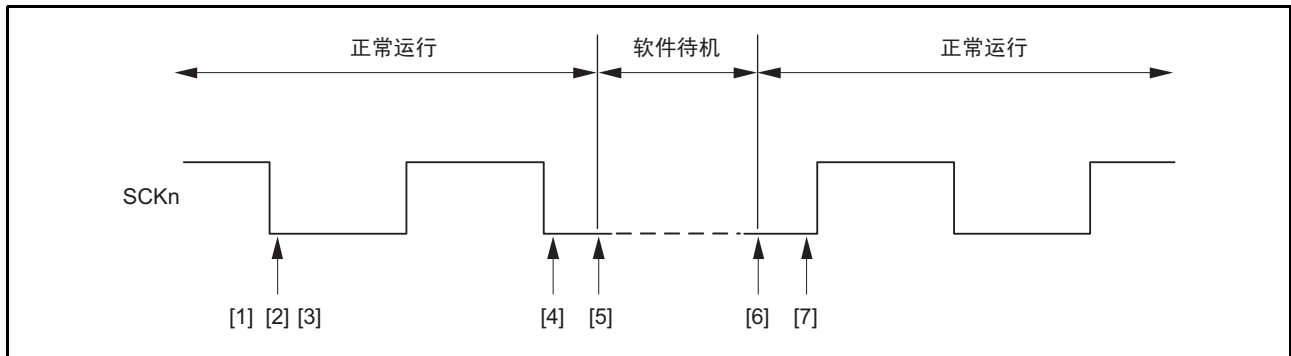


图 28.39 时钟的停止和重新启动的步骤

28.7 简易 I²C 模式的运行

简易 I²C 总线格式由 8 位数据和 1 个应答位构成。接在开始条件和重新开始条件之后的帧为从属地址帧，用于主控设备指定通信对象的从属设备。指定的从属设备有效到指定新的从属设备或者满足停止条件为止。从 MSB 开始依次发送各帧中的 8 位数据。

I²C 总线格式和总线时序分别如图 28.40 和图 28.41 所示。

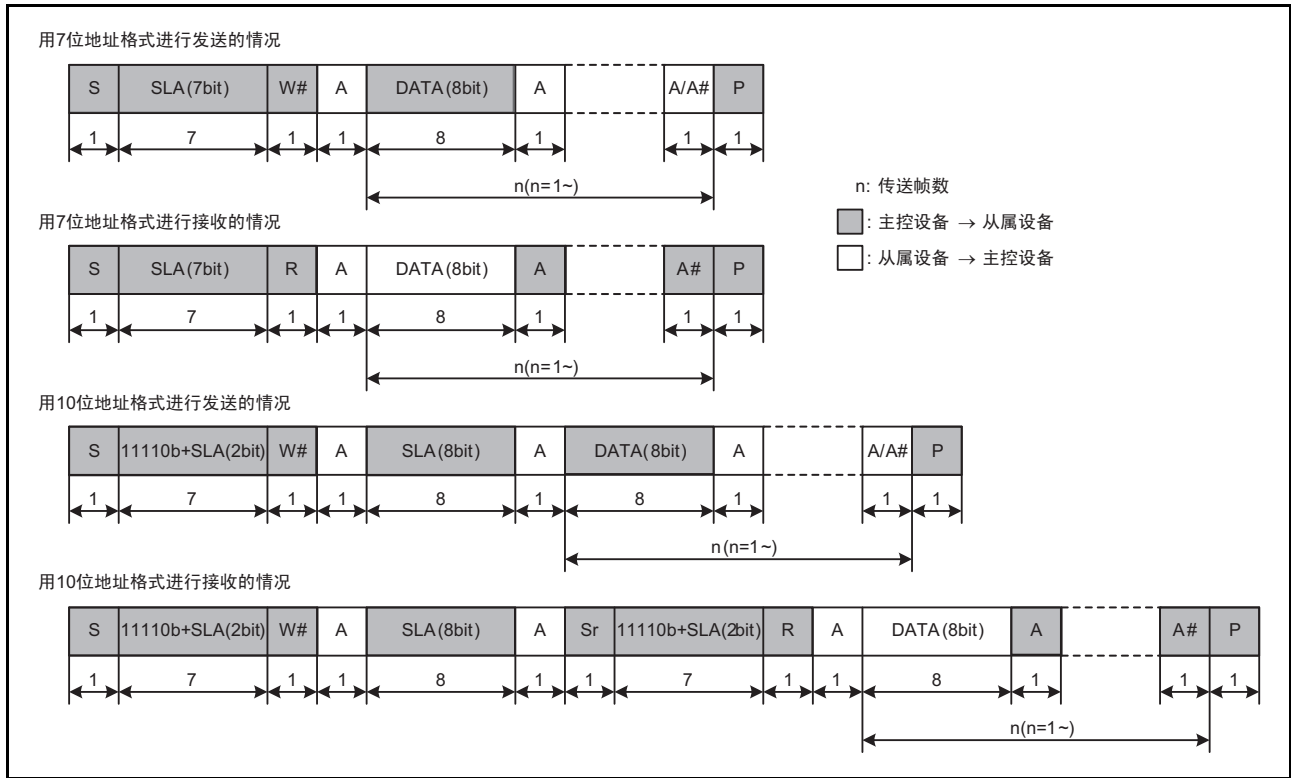


图 28.40 I²C 总线格式

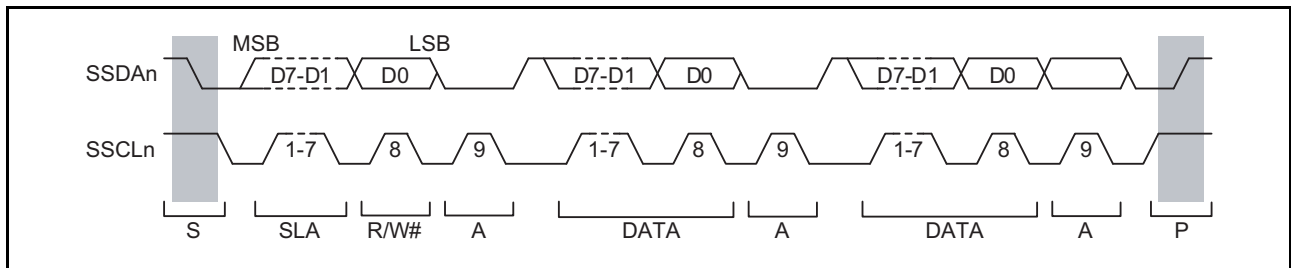


图 28.41 I²C 总线时序 (SLA 为 7 位的情况)

- S : 表示开始条件。主控设备在 SSCLn 线为 High 电平的状态下将 SSDAn 线从 High 电平变为 Low 电平。
- SLA : 表示从属地址。主控设备选择从属设备。
- R/W# : 表示发送和接收的方向。当 R/W# 为“1”时，将数据从从属设备发送到主控设备；当 R/W# 为“0”时，将数据从主控设备发送到从属设备。
- A/A# : 表示应答 (在主控发送模式中，从属设备返回应答；在主控接收模式中，主控设备返回应答)。ACK 返回 Low 电平，NACK 返回 High 电平。
- Sr : 表示重新开始条件。主控设备在 SSCLn 线为 High 电平的状态下经过准备时间后将 SSDAn 线从 High 电平变为 Low 电平。
- DATA : 表示发送和接收数据。
- P : 表示停止条件。主控设备在 SSCLn 线为 High 电平的状态下将 SSDAn 线从 Low 电平变为 High 电平。

28.7.1 开始条件、重新开始条件和停止条件的生成

通过给 SIMR3.IICSTAREQ 位写“1”生成开始条件。开始条件的生成如下所示：

- 降低SSDAn线的电平（从High电平变为Low电平），SSCLn线保持释放状态。
- 确保BRR寄存器所设位速率的一半时间以及开始条件的保持时间。
- 降低SSCLn线的电平（从High电平变为Low电平），使SIMR3.IICSTAREQ位变为“0”并且输出开始条件的生成中断请求。

通过给 SIMR3.IICRSTAREQ 位写“1”生成重新开始条件。重新开始条件的生成如下所示：

- 释放SSDAn线，SSCLn线保持Low电平。
- 确保BRR寄存器所设位速率的一半时间以及SSCLn线的Low电平时间。
- 释放SSCLn线（从Low电平变为High电平）。
- 在检测到SSCLn线的High电平后，确保BRR寄存器所设位速率的一半时间以及重新开始条件的准备时间。
- 降低SSDAn线的电平（从High电平变为Low电平）。
- 确保BRR寄存器所设位速率的一半时间以及重新开始条件的保持时间。
- 降低SSCLn线的电平（从High电平转移到Low电平），使SIMR3.IICRSTAREQ位变为“0”并且输出重新开始条件的生成中断请求。

通过给 SIMR3.IICSTPREQ 位写“1”生成停止条件。停止条件的生成如下所示：

- 降低SSDAn线的电平（从High电平变为Low电平），SSCLn线保持Low电平。
- 确保BRR寄存器所设位速率的一半时间以及SSCLn线的Low电平时间。
- 释放SSCLn线（从Low电平变为High电平）。
- 在检测到SSCLn线的High电平后，确保BRR寄存器所设位速率的一半时间以及停止条件的准备时间。
- 释放SSDAn线的电平（从Low电平变为High电平），使SIMR3.IICSTPREQ位变为“0”并且输出停止条件的生成中断请求。

开始条件、重新开始条件和停止条件生成的运行时序如图 28.42 所示。

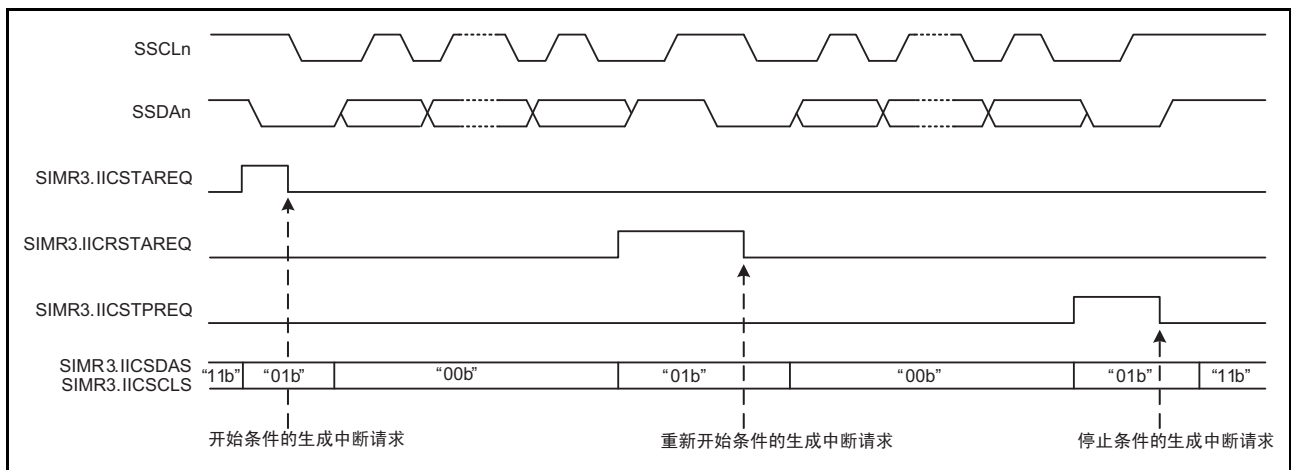


图 28.42 开始条件、重新开始条件和停止条件生成的运行时序

28.7.2 时钟同步

有时为了使通信对象的从属设备插入等待而将 SSCLn 线置为 Low 电平。如果将 SIMR2.IICCSC 位置“1”，就在内部 SSCLn 时钟和 SSCLn 引脚的输入电平不同时对同步的取得进行控制。

当 SIMR2.IICCSC 位为“1”时，如果内部 SSCLn 时钟从 Low 电平变为 High 电平，就在 SSCLn 引脚的输入为 Low 电平期间停止 High 电平期间的计数，一旦 SSCLn 引脚的输入变为 High 电平，就开始 High 电平期间的计数。此时，从 SSCLn 引脚变为 High 电平到开始 High 电平期间的计数为止，需要 SSCLn 引脚输入延迟、SSCLn 引脚输入的噪声滤波器延迟（2 ~ 3 个的噪声滤波器的采样时钟周期）和内部处理延迟（1 ~ 2 个 PCLK 周期）的时间。在此期间，即使其他设备不将 SSCLn 线置为 Low 电平，也会延长内部 SSCLn 时钟的 High 电平时间。

当 SIMR2.IICCSC 位为“1”时，与 SSCLn 引脚输入和内部 SSCLn 时钟的逻辑与同步进行数据的发送和接收；当 SIMR2.IICCSC 位为“0”时，与内部 SSCLn 时钟同步进行数据的接收和发送。

在从发行开始条件、重新开始条件和停止条件的生成请求后到内部 SSCLn 时钟从 Low 电平变为 High 电平为止的期间，如果从属设备插入等待，生成结束就延长此期间的的时间。

如果在内部 SSCLn 时钟变为 High 电平后从属设备插入等待，即使在此等待期间也不停止，而发行生成结束的中断请求，但是不保证条件的生成。

时钟同步的运行例子如图 28.43 所示。

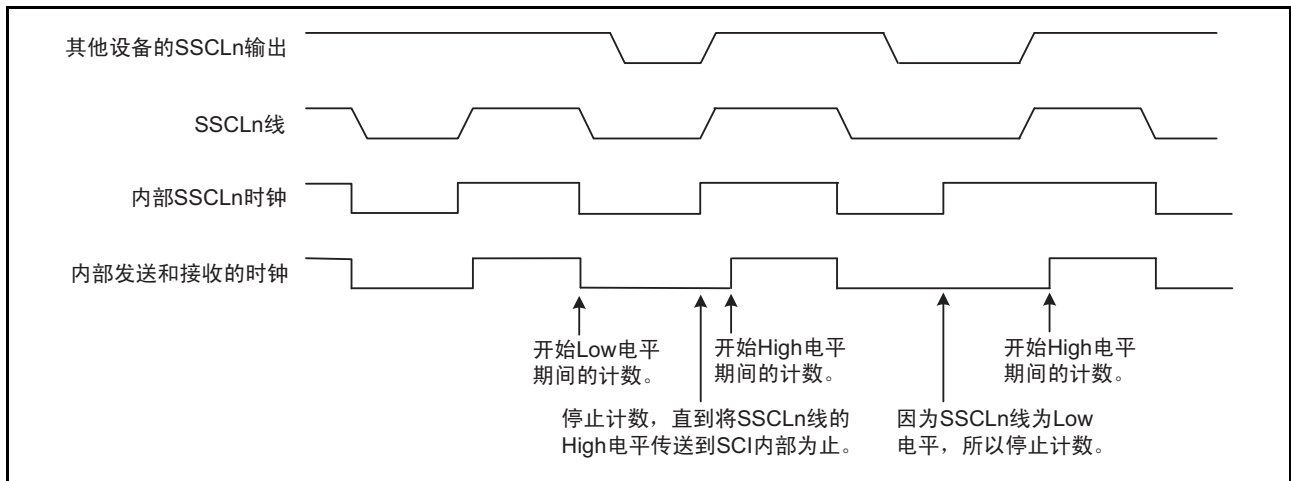


图 28.43 时钟同步的运行例子

28.7.3 SSDA 输出延迟

对于 SSCLn 引脚输出的下降沿，能通过 SIMR1.IICDL[4:0] 位延迟 SSDAn 引脚的输出。延迟时间能以内部波特率发生器的时钟源为基准（以 PCLK 为基准通过 SMR.CKS[1:0] 选择的分频时钟），选择 0 ~ 31 个周期的延迟时间。延迟 SSDAn 引脚输出的对象为开始条件信号、重新开始条件信号、停止条件信号、8 位发送数据和应答。

如果 SSDA 输出延迟时间比 SSCLn 引脚输出的下降时间短，SSDA 引脚的输出就可能在 SSCLn 引脚输出的下降过程中开始变化，使从属设备发生误动作。因此必须将 SSDA 输出延迟设定为大于 SSCLn 引脚输出的下降时间的最大值（在 I²C 的标准模式和快速模式中为 300ns）。

SSDA 输出延迟的时序如图 28.44 所示。

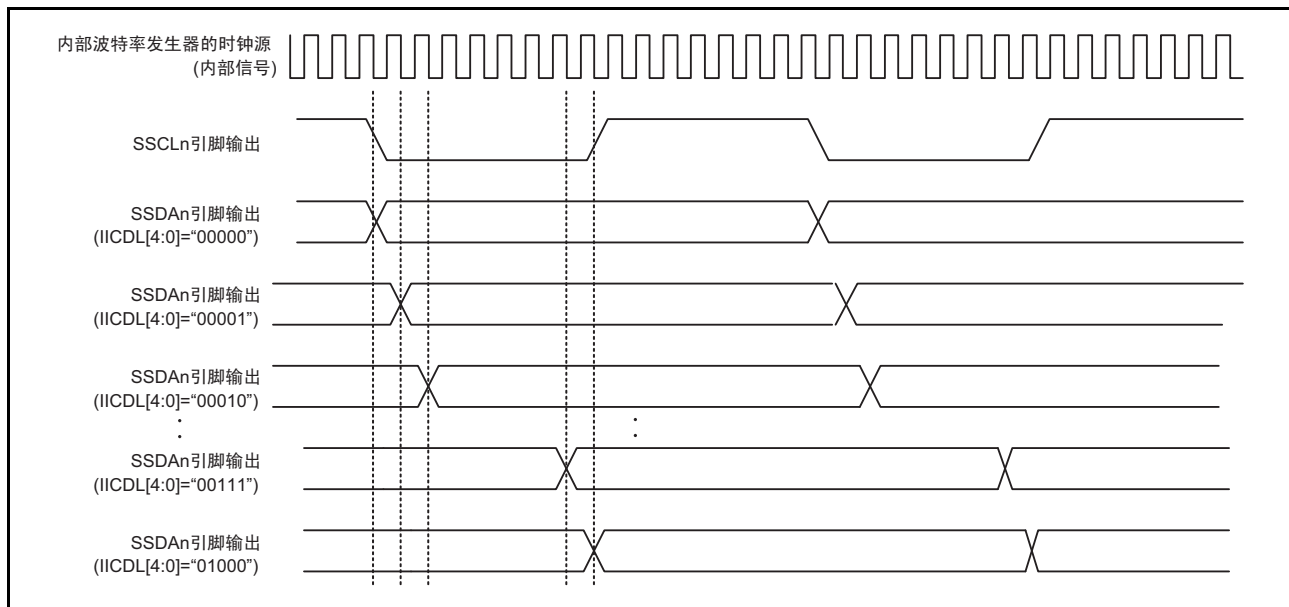


图 28.44 SSDA 输出延迟的时序

28.7.4 SCI 的初始化 (简易 I²C 模式)

必须在发送和接收数据前给 SCR 寄存器写初始值“00h”，并且按照图 28.45 的流程图例子进行初始化。

必须在将 SCR 寄存器置为初始值后更改运行模式和通信格式。另外，在简易 I²C 模式中，必须在端口侧设定通信端口的漏极开路。

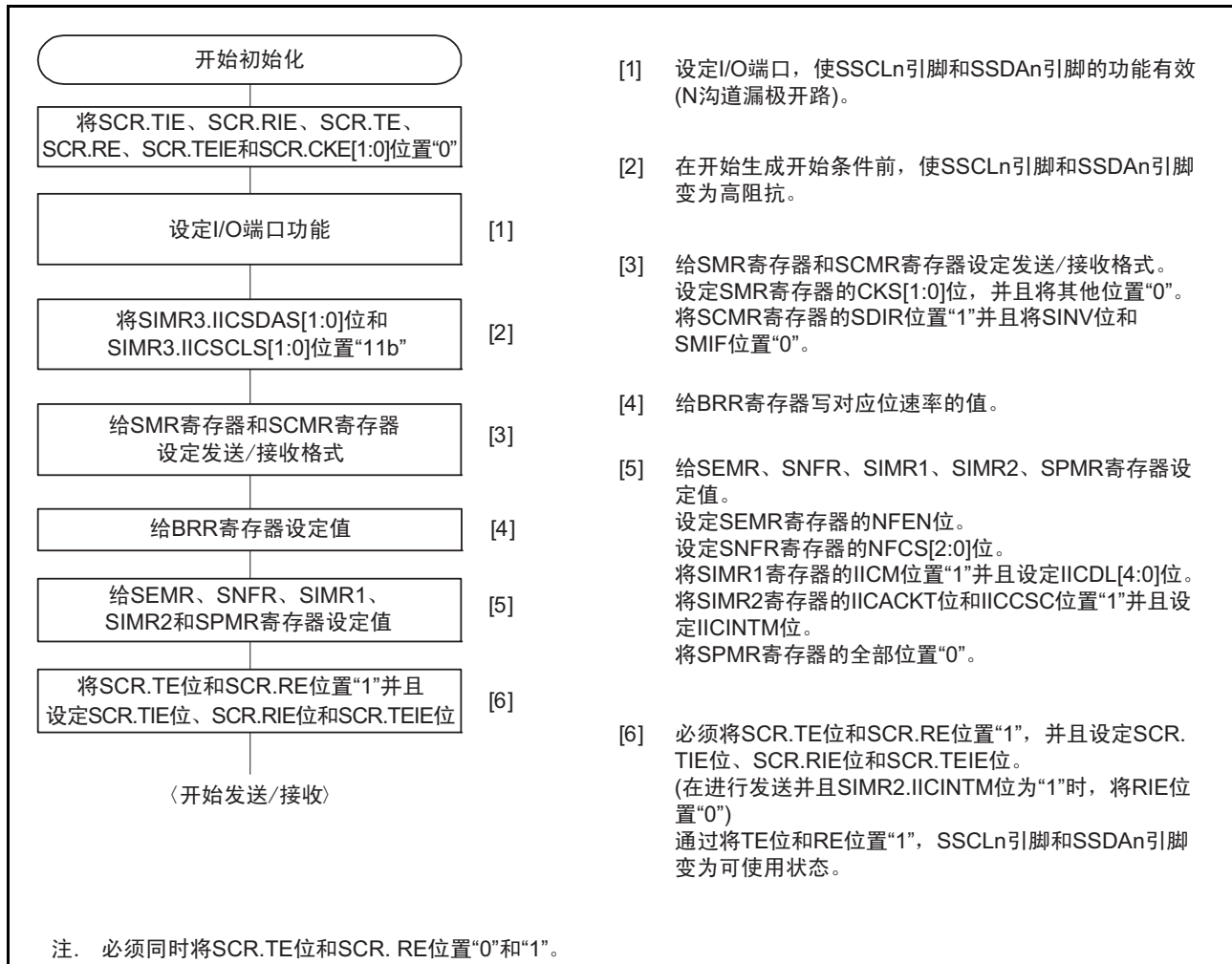


图 28.45 SCI 的初始化流程图例子 (简易 I²C 模式)

28.7.5 主控发送 (简易 I²C 模式)

简易 I²C 模式的主控发送运行例子如图 28.46 和图 28.47 所示，数据发送的流程图例子如图 28.48 所示。这些例子都假设将 SIMR2.IICINTM 位置“1”（使用接收中断和发送中断）并且将 SCR.RIE 位置“0”（禁止接收中断请求）。有关 STI 中断，请参照表 28.29。

在 10 位从属地址模式中，重复 2 次图 28.48 中的步骤 [3] ~ [4]。

简易 I²C 模式中的发送结束中断 (TXI) 发生在 1 帧的通信结束时，和时钟同步发送时的 TXI 中断请求发生时序不同。

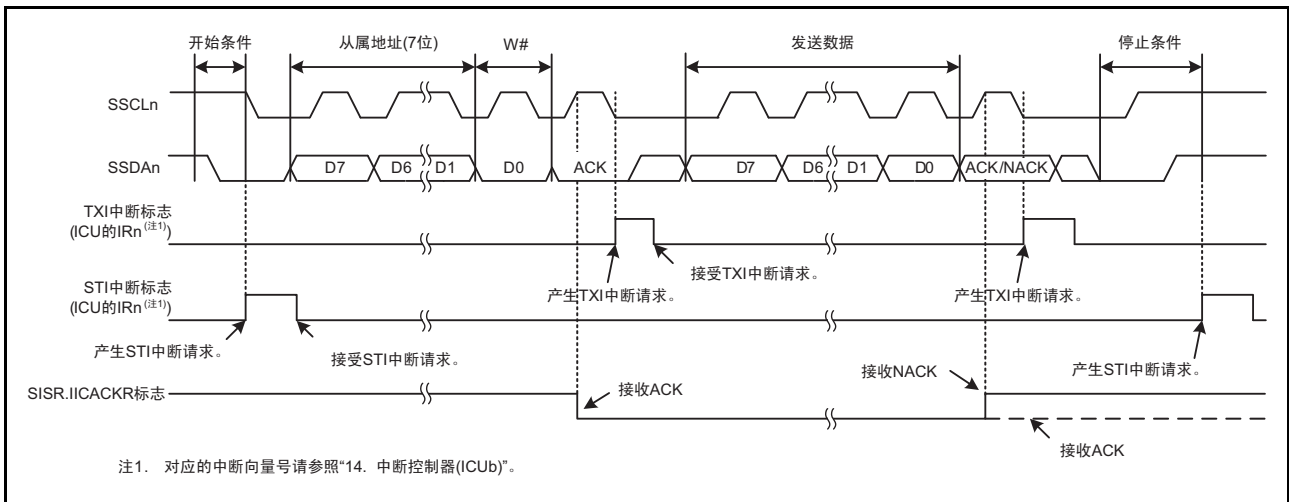


图 28.46 简易 I²C 总线模式的主控发送运行例子 1 (使用 7 位从属地址、发送中断和接收中断的情况)

如果在主控发送时将 SIMR2.IICINTM 位置“0”（使用 ACK 中断和 NACK 中断），就以 ACK 中断为触发，启动 DTC 或者 DMAC 并且发送所需字节数的数据。如果接收到 NACK，就以 NACK 中断为触发，中止发送并且进行重新发送等错误处理。

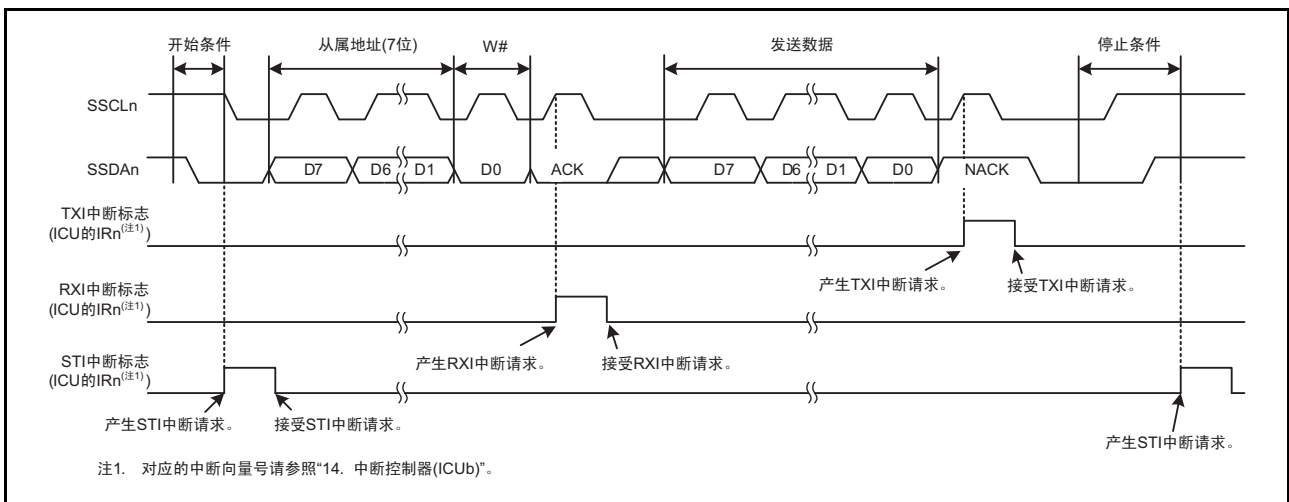


图 28.47 简易 I²C 总线模式的主控发送运行例子 2 (使用 7 位从属地址、ACK 中断和 NACK 中断的情况)

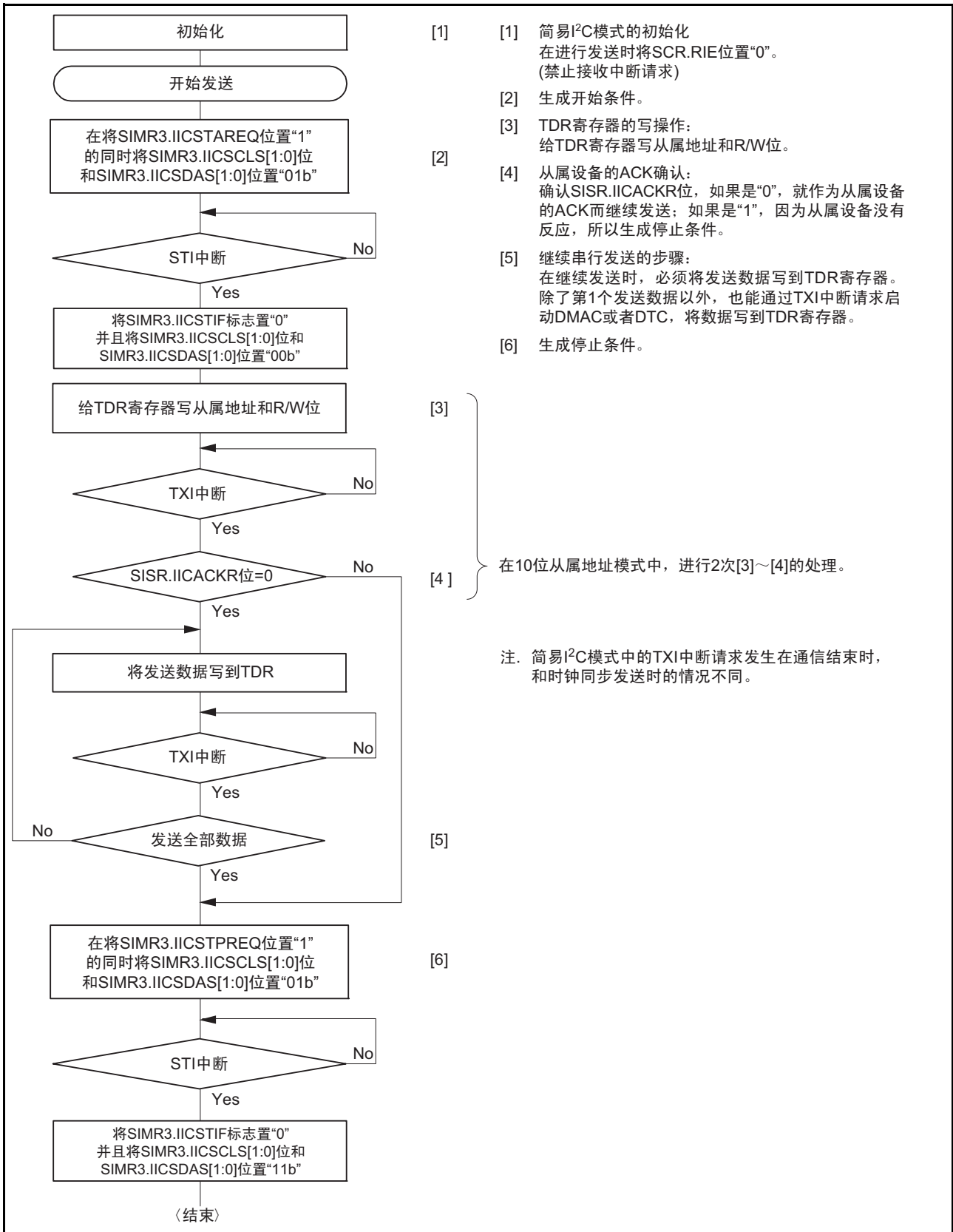


图 28.48 简易 I²C 模式的主控发送运行流程图例子 (使用发送中断和接收中断的情况)

28.7.6 主控接收 (简易 I²C 模式)

简易 I²C 模式的主控接收运行例子和主控接收的流程图例子分别如图 28.49 和图 28.50 所示。这些例子都假设将 SIMR2.IICINTM 位置“1” (使用接收中断和发送中断)。

简易 I²C 模式的发送结束中断 (TXI) 发生在 1 帧的通信结束时, 和时钟同步发送时的 TXI 中断请求发生时序不同。

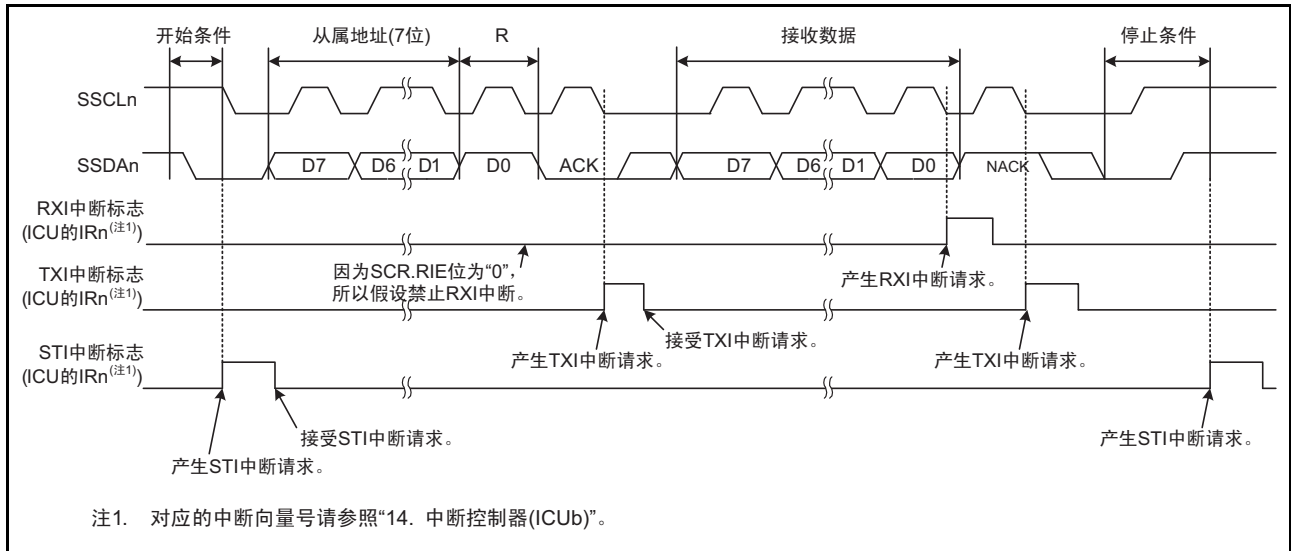


图 28.49 简易 I²C 总线模式的主控接收运行例子 (使用 7 位从属地址、发送中断和接收中断的情况)

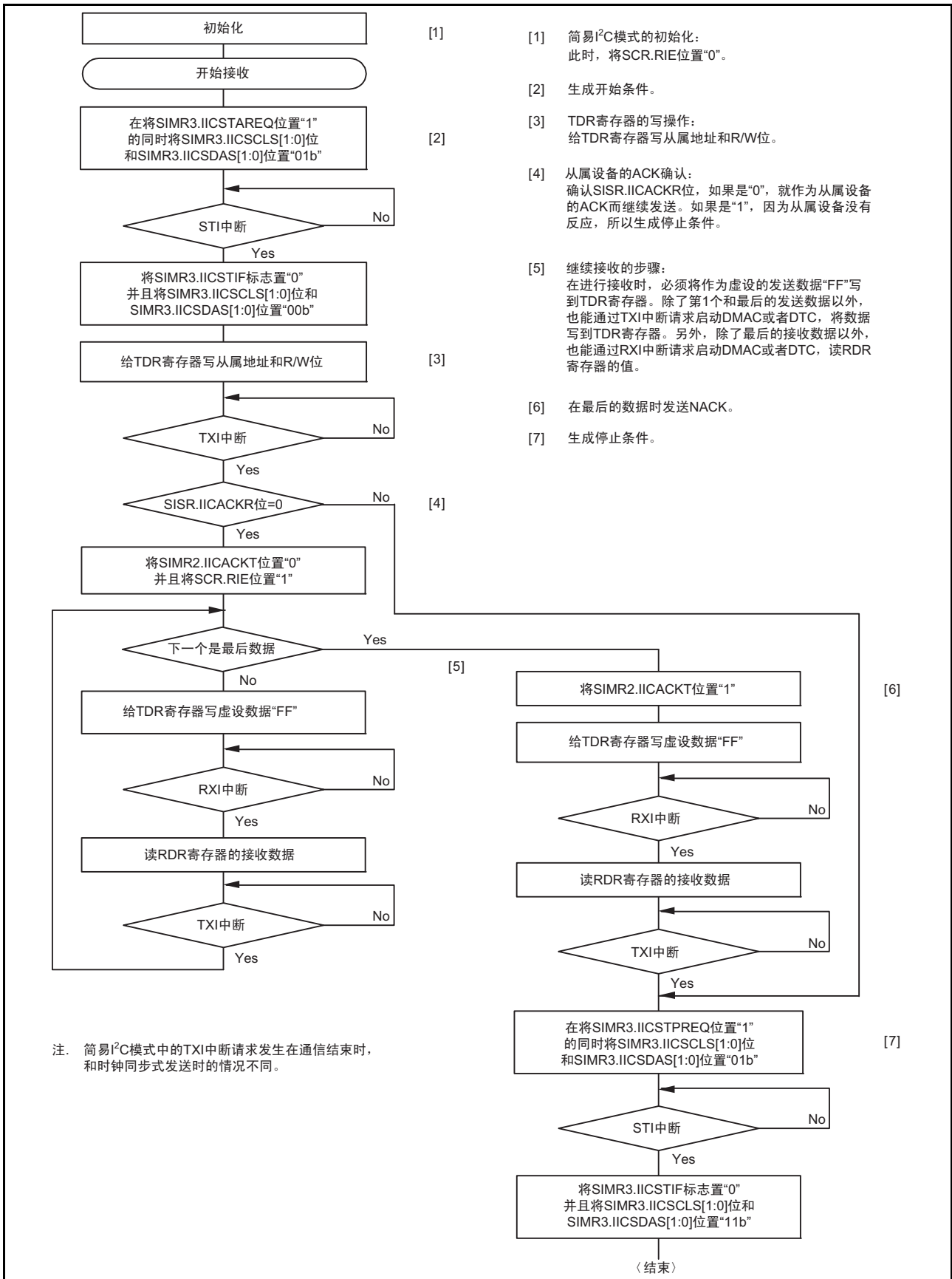


图 28.50 简易 I²C 模式的主控接收运行流程图例子 (使用发送中断和接收中断的情况)

28.8 简易 SPI 模式的运行

作为 SCI 的扩展功能，支持 1 个或者多个主控设备能对多个从属设备进行通信的简易 SPI 模式。

通过设定时钟同步模式 (SCMR.SMIF 位 =0、SIMR1.IICM 位 =0、SMR.CM 位 =1) 并且将 SPMR.SSE 位置“1”，进入简易 SPI 模式。在简易 SPI 模式用途中用于主控模式的单主控时，不需要主控侧的 SS 引脚功能，将 SPMR.SSE 位置“0”。

简易 SPI 模式的连接例子如图 28.51 所示。必须通过通用端口控制主控设备的 SS 信号的输出。

简易 SPI 模式和时钟同步模式一样，与时钟脉冲同步进行数据的发送和接收。通信数据的 1 个字符由 8 位数据构成，不能附加奇偶校验位。能通过将 SCMR.SINV 位置“1”，将发送 / 接收数据取反。

SCI 内部的发送部和接收部独立，因此能通过共享时钟进行全双工通信。因为发送部和接收部都为双缓冲结构，所以能通过发送过程中写下一个发送数据并且在接收过程中读前一个接收数据，进行连续的发送和接收。

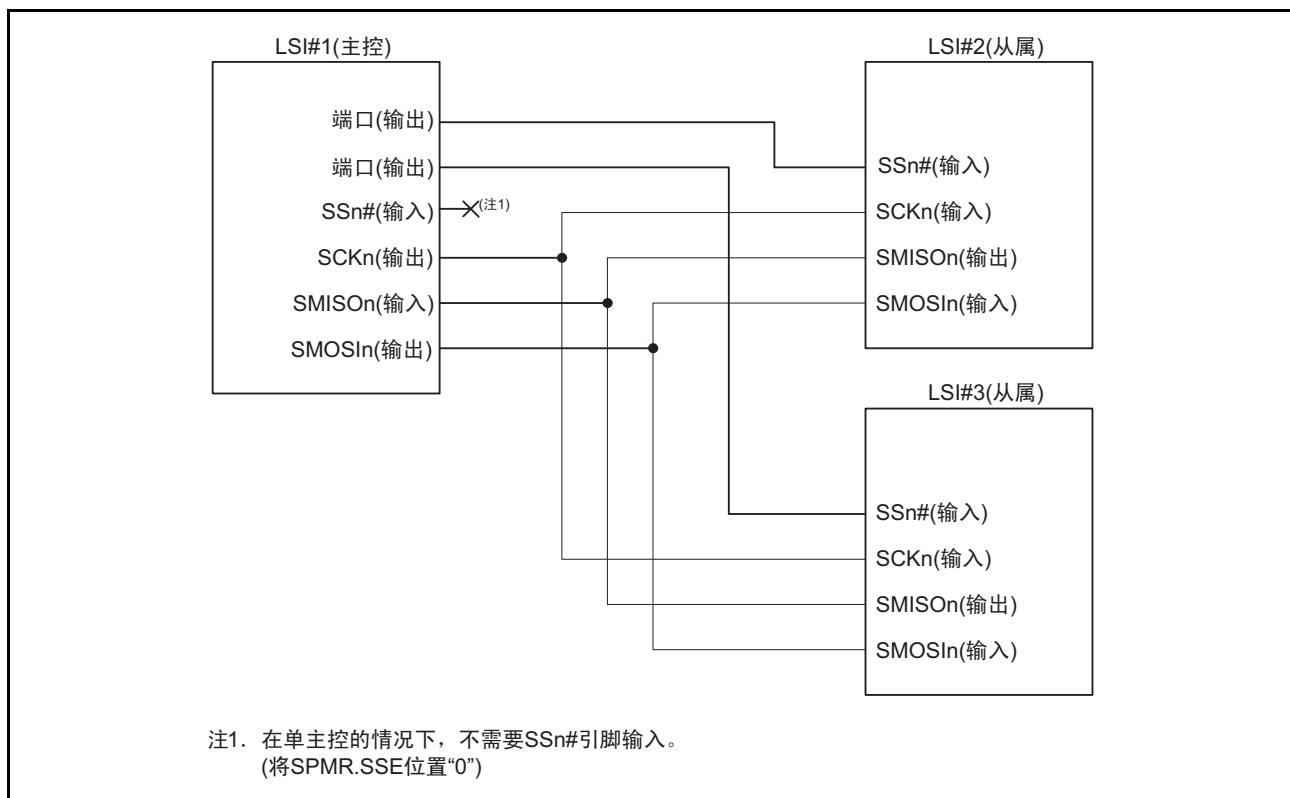


图 28.51 简易 SPI 模式的连接例子 (单主控 (SPMR.SSE 位 =0))

28.8.1 主控模式、从属模式和各引脚的状态

在简易 SPI 模式中，主控模式 (SCR.CKE[1:0] 位 =00 或者 01 并且 SPMR.MSS 位 =0) 和从属模式 (SCR.CKE[1:0] 位 =10 或者 11 并且 SPMR.MSS 位 =1) 中的各引脚的输入 / 输出方向不同。

模式、SSn# 引脚输入和各引脚状态的关系如表 28.25 所示。

表 28.25 模式、SSn# 引脚输入和各引脚状态的关系

模式	SSn# 引脚输入	SMOSIn 引脚状态	SMISOIn 引脚状态	SCKn 引脚状态
主控模式 (注 1)	High 电平 (能通信)	发送数据输出 (注 2)	接收数据输入	时钟输出 (注 3)
	Low 电平 (不能通信)	高阻抗	接收数据输入 (无效)	高阻抗
从属模式	High 电平 (不能通信)	接收数据输入 (无效)	高阻抗	时钟输入 (无效)
	Low 电平 (能通信)	接收数据输入	发送数据输出	时钟输入

注 1. 在单主控时 (SPMR.SSE 位 =0) 为可通信状态，与 SSn# 引脚的输入电平无关 (和 SSn# 引脚的输入为 High 电平时等效)。不使用 SSn# 引脚，而此引脚能用作其他用途。

注 2. 在禁止发送时 (SCR.TE 位 =0) 为高阻抗状态。

注 3. 在多主控 (SPMR.SSE 位 =1) 并且禁止发送和接收时 (SCR.TE 位 =00 并且 SCR.RE 位 =00) 为高阻抗状态。

28.8.2 主控模式中的 SS 功能

通过将 SCR.CKE[1:0] 位置 “00” 并且将 SPMR.MSS 位置 “0”，进入主控模式。

在单主控时 (SPMR.SSE 位 =0) 不使用 SSn# 引脚，与 SSn# 引脚输入的电平无关，能进行发送和接收。

在多主控 (SPMR.SSE 位 =1) 并且 SSn# 引脚的输入为 High 电平时，作为表示没有其他主控或者其他主控没有进行发送和接收的主控，从 SCKn 引脚输出时钟并且进行发送和接收。在多主控 (SPMR.SSE 位 =1) 并且 SSn# 引脚的输入为 Low 电平时，表示有其他主控并且正在进行发送和接收。此时，SCI 将 SMOSIn 引脚和 SCKn 引脚的输出置为高阻抗，不开始发送和接收。在发生模式故障错误时，SPMR.MFF 标志为 “1”。在多主控时，必须通过读 SPMR.MFF 标志来进行错误处理。即使在发送和接收过程中发生模式故障，也不停止发送和接收，而在发送和接收结束后，将 SMOSIn 引脚和 SCKn 引脚的输出置为高阻抗。

必须通过通用端口控制主控的 SS 信号输出。

28.8.3 从属模式中的 SS 功能

通过将 SCR.CKE[1:0] 位置 “10” 并且将 SPMR.MSS 位置 “1”，进入从属模式。

当 SSn# 引脚的输入为 High 电平时，SMISOIn 引脚的输出为高阻抗，忽视 SCKn 引脚的时钟输入；当 SSn# 引脚的输入为 Low 电平时，SCKn 引脚的时钟输入有效并且能进行发送和接收。如果在发送和接收过程中 SSn# 引脚的输入从 Low 电平变为 High 电平，就根据 SCKn 引脚的时钟输入，在发送和接收结束前不停止运行。此后，在 SSn# 引脚的输入从 High 电平变为 Low 电平前，忽视 SCKn 引脚的时钟输入。

28.8.4 时钟和发送 / 接收数据的关系

能通过 SPMR.CKPOL 位和 SPMR.CKPH 位从 4 种时钟中选择用于发送和接收的时钟。时钟和发送 / 接收数据的关系如图 28.52 所示。在从属模式和从属模式中，时钟和发送 / 接收数据的关系都相同 (和 SSn# 引脚的输入为 High 电平时等效)。SSn# 引脚能用于其他用途。详细内容请参照 “28.8.2 主控模式中的 SS 功能”。

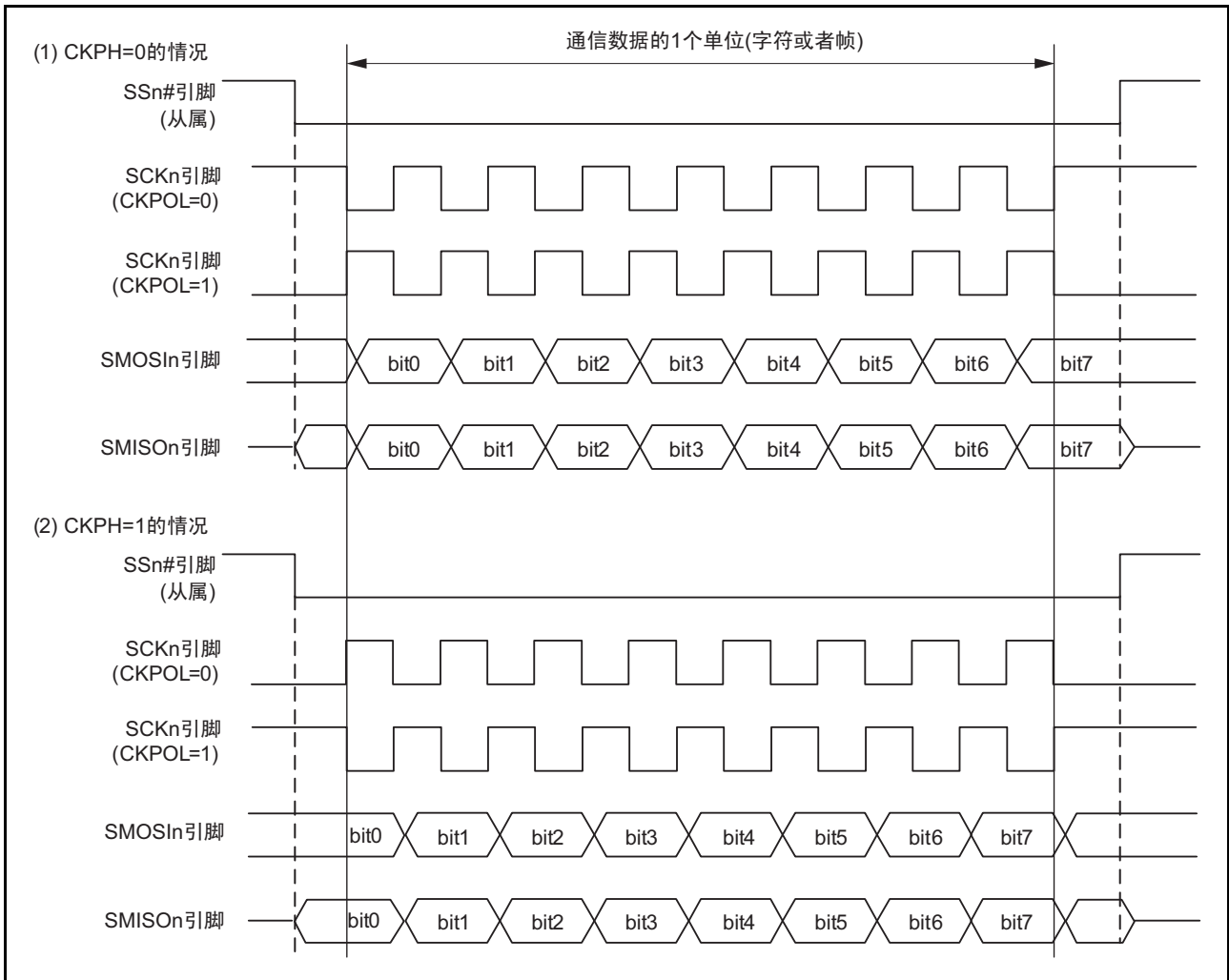


图 28.52 简易 SPI 模式的时钟和发送 / 接收数据的关系

28.8.5 SCI 的初始化 (简易 SPI 模式)

和时钟同步模式的初始化步骤 (图 28.21 的 SCI 的初始化流程图例子) 相同。SPMR.CKPOL 位和 SPMR.CKPH 位选择的时钟种类适合于主控设备和从属设备。

必须在将 SCR 寄存器置为初始值后进行初始化以及更改运行模式和通信格式。

必须注意: 即使将 RE 位置 “0”, 也不对 SSR.ORER 标志、SSR.FER 标志、SSR.PER 标志和 RDR 寄存器进行初始化。

必须注意: 如果将 TE 位从 “1” 置为 “0” 或者从 “0” 置为 “1”, 就在 SCR.TIE 位为 “1” 时产生 TXI 中断。

28.8.6 串行数据的发送和接收 (简易 SPI 模式)

在主控模式中, 如果在开始发送和接收前将发送和接收目标的从属设备的 SSn# 引脚置为 Low 电平并且结束发送和接收, 就将发送和接收目标的从属设备的 SSn# 引脚置为 High 电平。其他步骤和时钟同步模式相同。

28.9 扩展串行模式控制部的运行说明

28.9.1 串行通信协议

如图 28.53 所示，SCId 的扩展串行模式控制部和 SCIc 一起实现由 Start Frame 和 Information Frame 构成的串行通信协议。

Start Frame 由 Break Field、Control Field 0 和 Control Field 1 构成。Information Frame 能由几个 Data Field、CRC16 Upper Field 和 CRC16 Lower Field 构成。

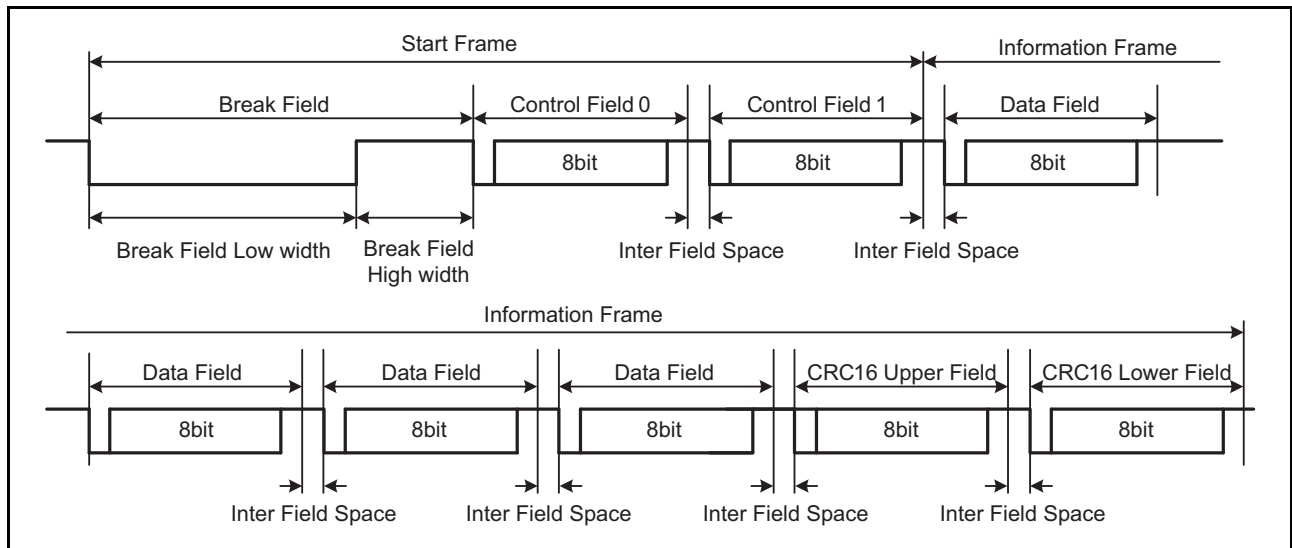


图 28.53 扩展串行模式控制部的串行通信协议

28.9.2 Start Frame 发送

发送由 Break Field Low width、Control Field 0 和 Control Filed 1 构成的 Start Frame 时的运行例子如图 28.54 所示，发送 Start Frame 的流程如图 28.55 和图 28.56 所示。

扩展串行模式控制部在 Start Frame 发送时的运行如下所示。SCI12 用于异步模式。

- (1) 在将定时器的运行模式设定为 Break Field Low width 输出模式的状态下，如果给 TCR 的 TCST 位写“1”，定时器就开始计数，在 TCNT 和 TPRE 所设期间从 TXDX12 引脚输出 Low 电平。
- (2) 如果定时器发生下溢，TXDX12 引脚就进行反相输出并且 STR 的 BFDL 位变为“1”。此时，如果 ICR 的 BFDIE 位为“1”，就产生 SCIX0 中断。
- (3) 给 TCR 的 TCST 位写“0”，停止定时器的计数，使用 SCI12 发送 Control Field 0 的数据。必须在从输出 Break Field Low width 后到发生下一次下溢前停止计数。
- (4) 在发送完 Control Field 0 的数据后，使用 SCI12 发送 Control Field 1 的数据。
- (5) 在发送完 Control Field 1 的数据后，使用 SCI12 进行 Information Frame 的通信。

必须根据 Start Frame 的结构省略 Break Field 和 Control Field 0。

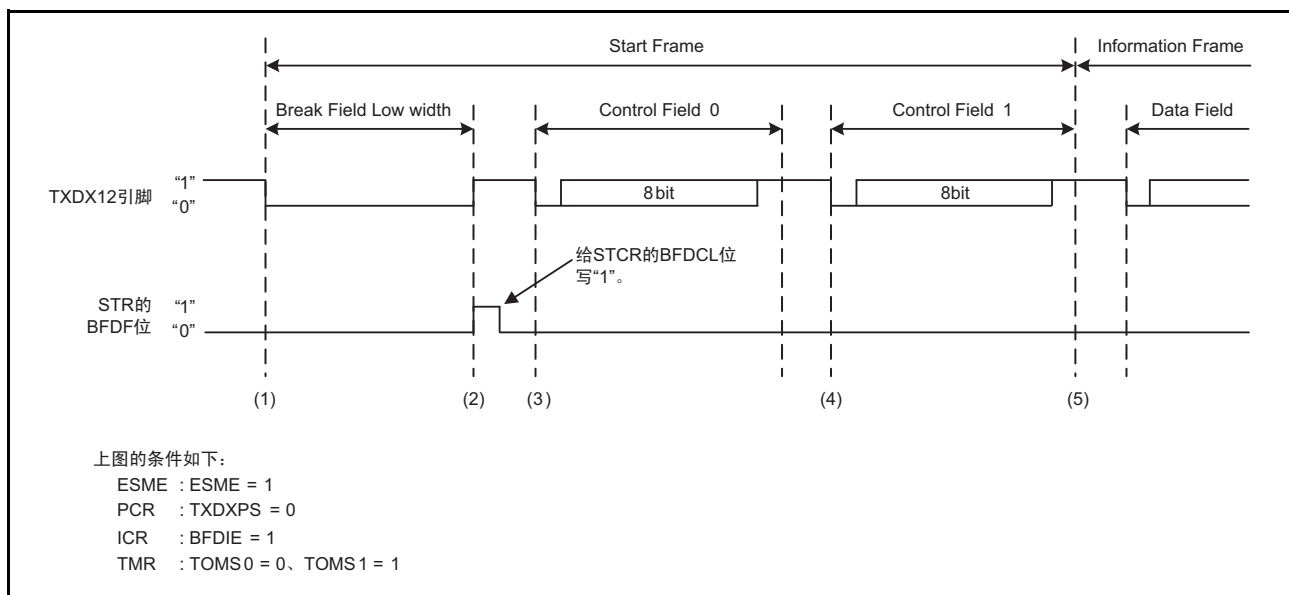


图 28.54 发送 Start Frame 时的运行例子

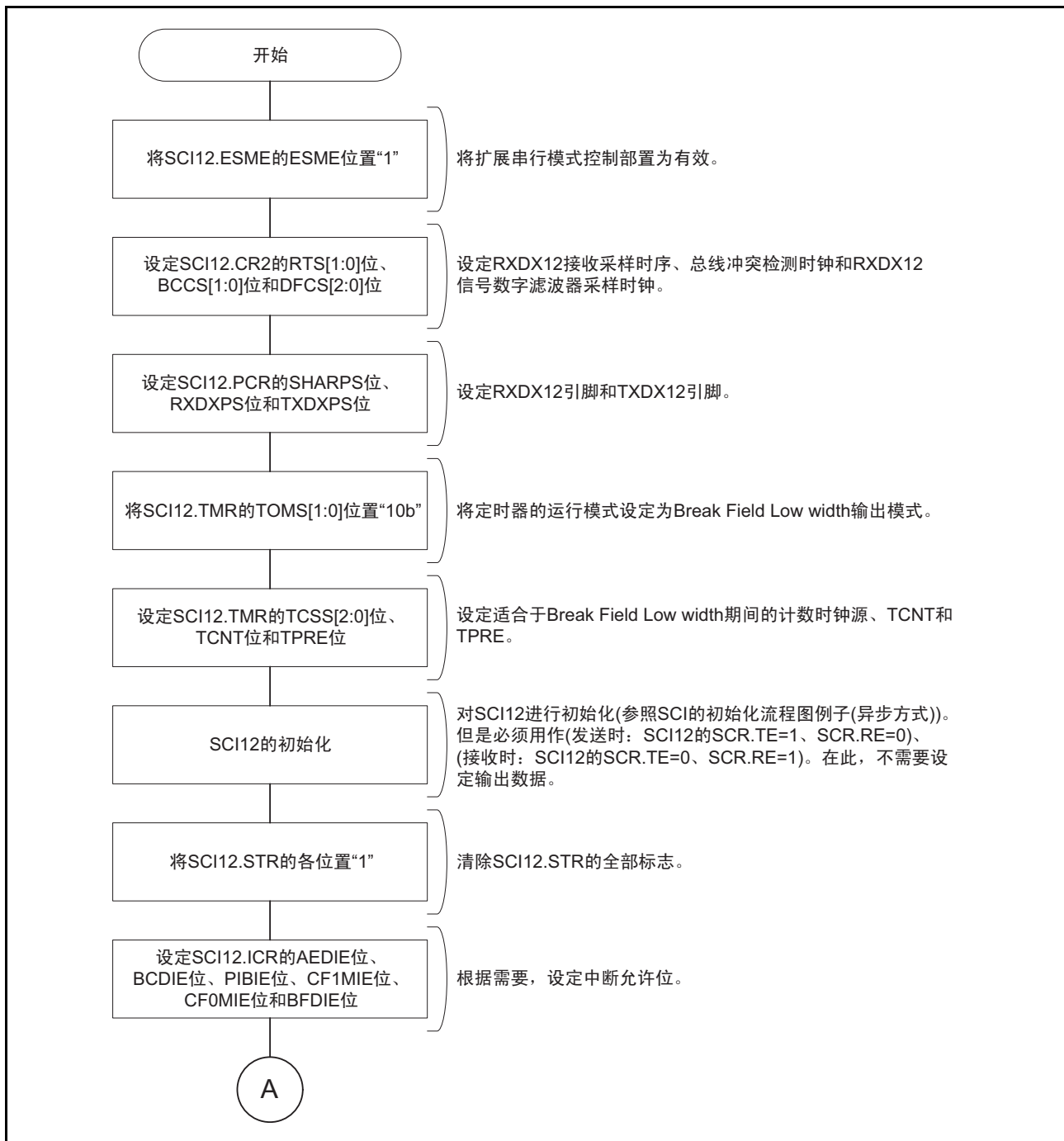


图 28.55 Start Frame 的发送流程图例子 (1)

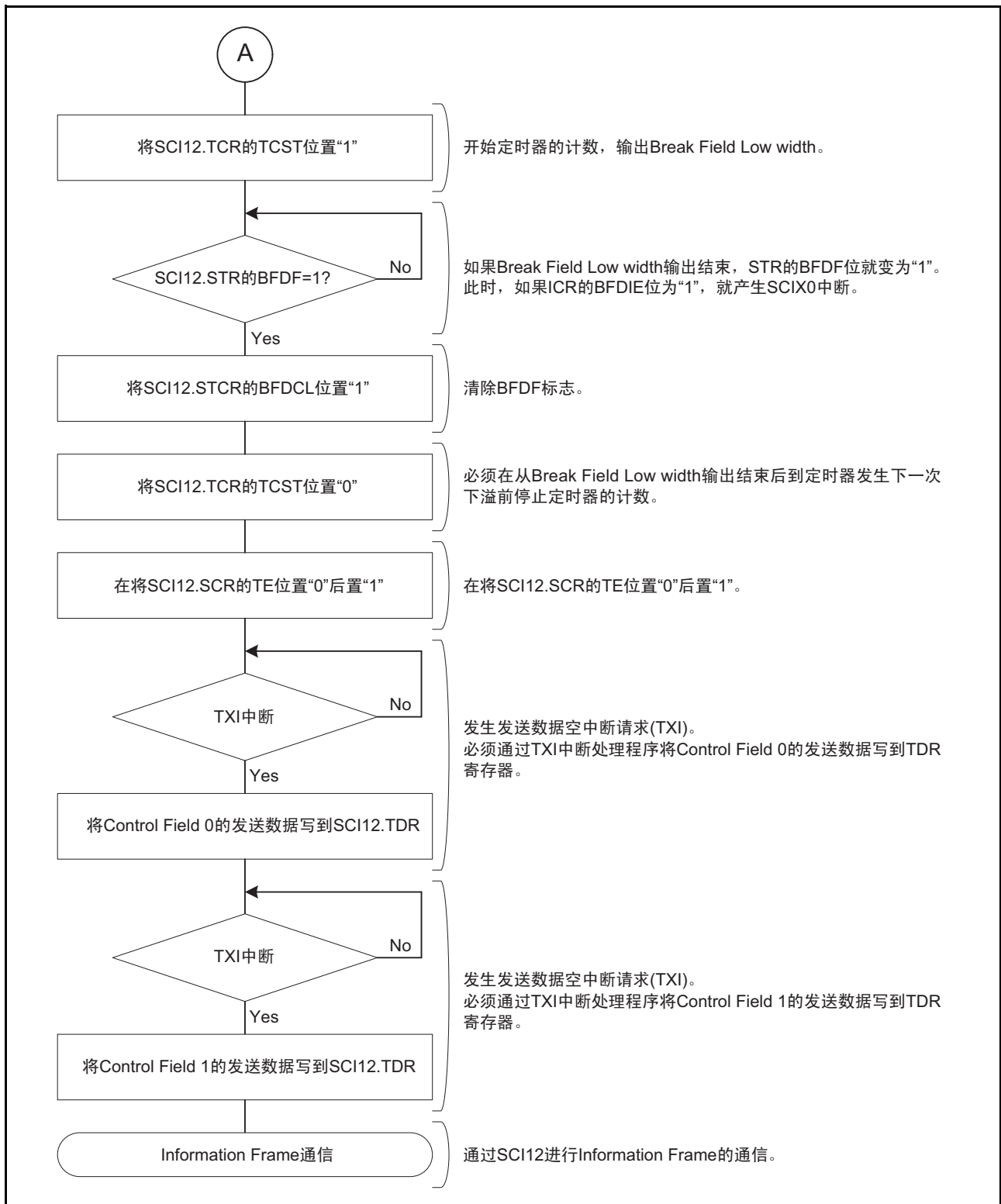


图 28.56 Start Frame 的发送流程图例子 (2)

28.9.3 Start Frame 接收

扩展串行模式控制部能检测到如表 28.26 所示结构的 Start Frame。

表 28.26 Start Frame 的结构

位的设定		Start Frame 的结构
BFE	CF0RE	
0	0	
0	1	
1	0	
1	1	

接收由 Break Field Low width、Control Field 0 和 Control Field 1 构成的 Start Frame 时的运行例子如图 28.57 所示。接收 Start Frame 的流程图如图 28.58 和图 28.59 所示，扩展串行模式控制部的状态转移图如图 28.60 所示。扩展串行模式控制部在接收 Start Frame 时的运行如下所示。SCI12 用于异步模式。

- (1) 如果在将定时器的运行模式设定为 Break Field Low width 检测模式后给 CR3 的 SDST 位写“1”，就能检测 Break Field Low width。此时，禁止将 RXDX12 输入到 SCI12。
- (2) 如果从 RXDX12 引脚至少输入定时器的 TCNT 和 TPRES 设定期间的 Low 电平，就检测为 Break Field Low width。此时，STR 的 BFDF 位变为“1”，如果 ICR 的 BFDIE 位为“1”，就产生 SCIX0 中断。
- (3) 如果在检测到 Break Field Low width 后 RXDX12 引脚的输入为 High 电平，CR0 的 RXDSF 位就变为“0”，通过 SCI12 开始接收 Control Field 0。
- (4) 如果 Control Field 0 接收到的数据和 CF0DR 设定的数据相同，STR 的 CF0MF 位就变为“1”。此时，如果 ICR 的 CF0MIE 位为“1”，就产生 SCIX1 中断，而且通过 SCI12 开始接收 Control Field 1。如果 Control Field 0 接收到的数据和 CF0DR 设定的数据不同，就转移到检测 Break Field Low width 前的状态。
- (5) 如果 Control Field 1 接收到的数据和 PCF1DR 或者 SCF1DR 设定的数据相同，STR 的 CF1MF 位就变为“1”。此时，如果 ICR 的 CF1MIE 位为“1”，就产生 SCIX1 中断，而且使用 SCI12 进行 Information Frame 的通信。如果 Control Field 1 接收到的数据和 PCF1DR 或者 SCF1DR 设定的数据都不同，就转移到检测 Break Field Low width 前的状态。

必须根据 Start Frame 的结构省略 Break Field 和 Control Field 0 的处理。

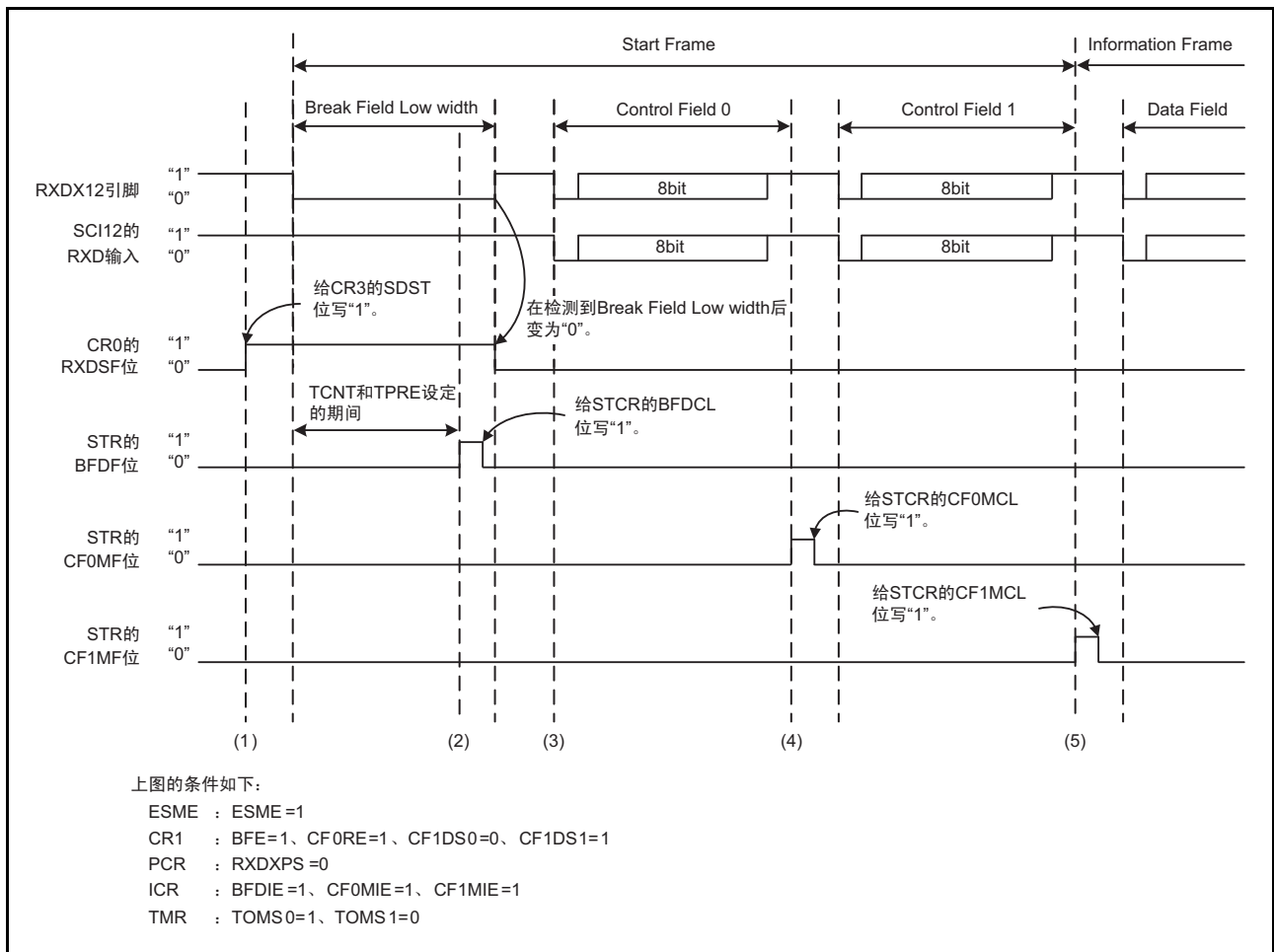


图 28.57 接收 Start Frame 时的运行例子



图 28.58 Start Frame 的接收流程图例子 (1)

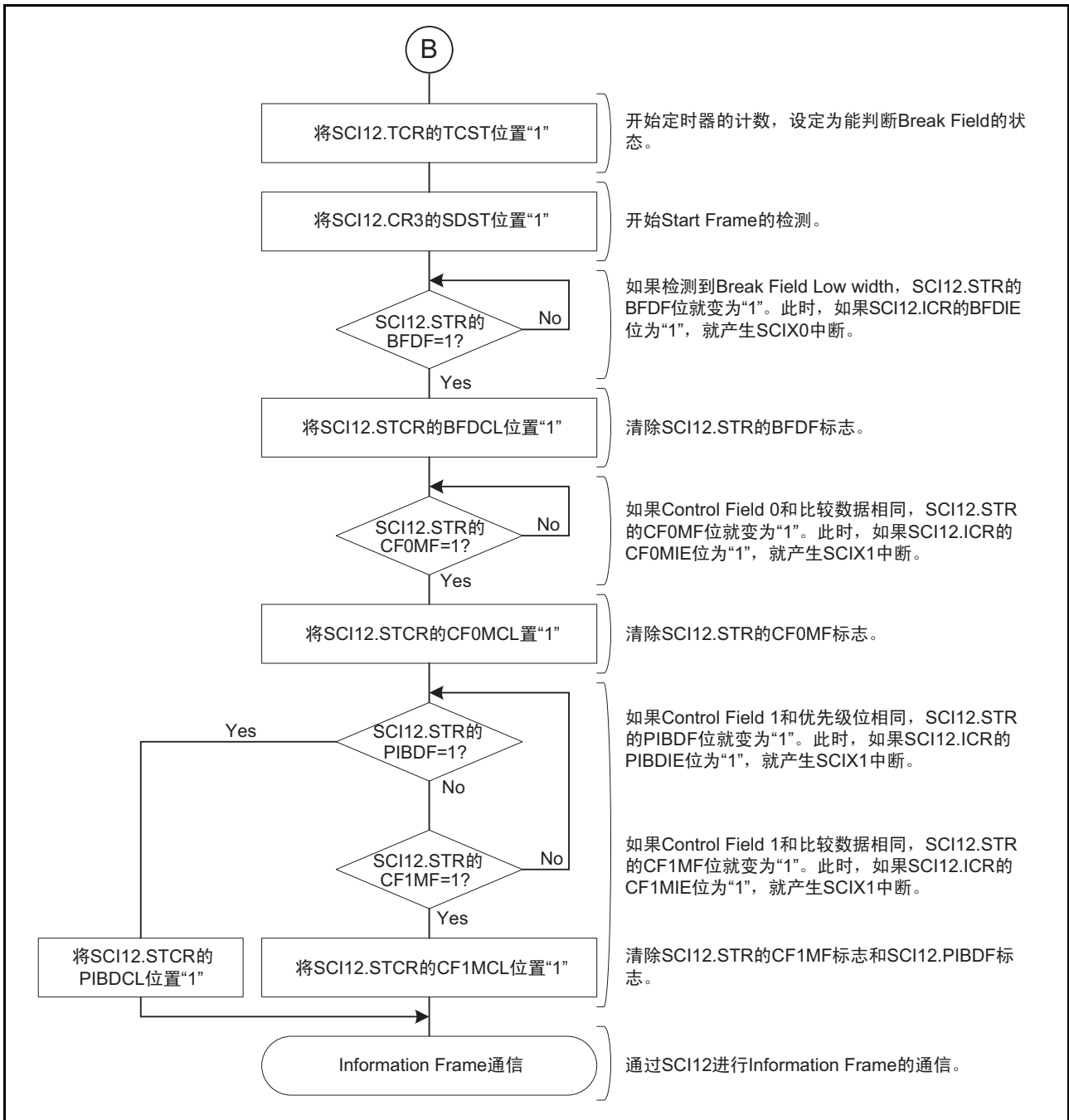


图 28.59 Start Frame 的接收流程图例子 (2)

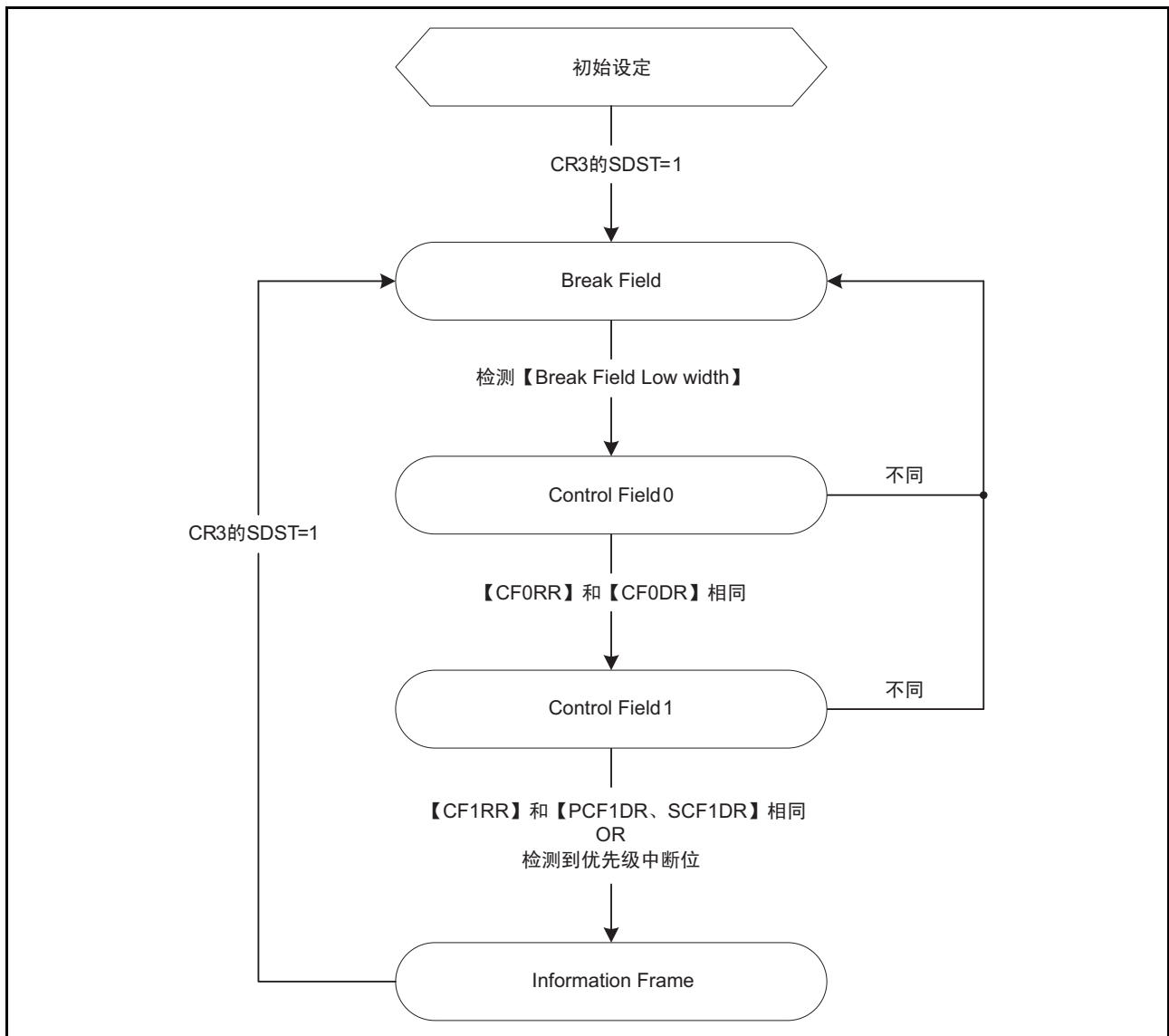


图 28.60 接收 Start Frame 时的状态转移图

28.9.3.1 优先级中断位

接收使用了优先级中断位的 Start Frame 时的运行例子如图 28.61 所示。通过将 CR1 的 PIBE 位置“1”，优先级中断位变为有效。

扩展串行模式控制部在接收使用了优先级中断位的 Start Frame 时的运行如下所示：

(1) ~ (4) 和图 28.57 中的接收 Start Frame 时的运行例子 (1) ~ (4) 相同。

(5) 如果 CR1 的 PIBS[2:0] 位指定的位数据和 PCF1DR 设定的数据相同，STR 的 PIBDF 位就变为“1”。此时，如果 ICR 的 PIBDIE 位为“1”，就产生 SCIX1 中断，而且通过 SCI12 进行 Information Frame 的通信。如果 Control Field 1 接收到的数据和 PCF1DR 或者 SCF1DR 设定的数据都不同并且也检测不到优先级中断位，就转移到检测 Break Field Low width 前的状态。

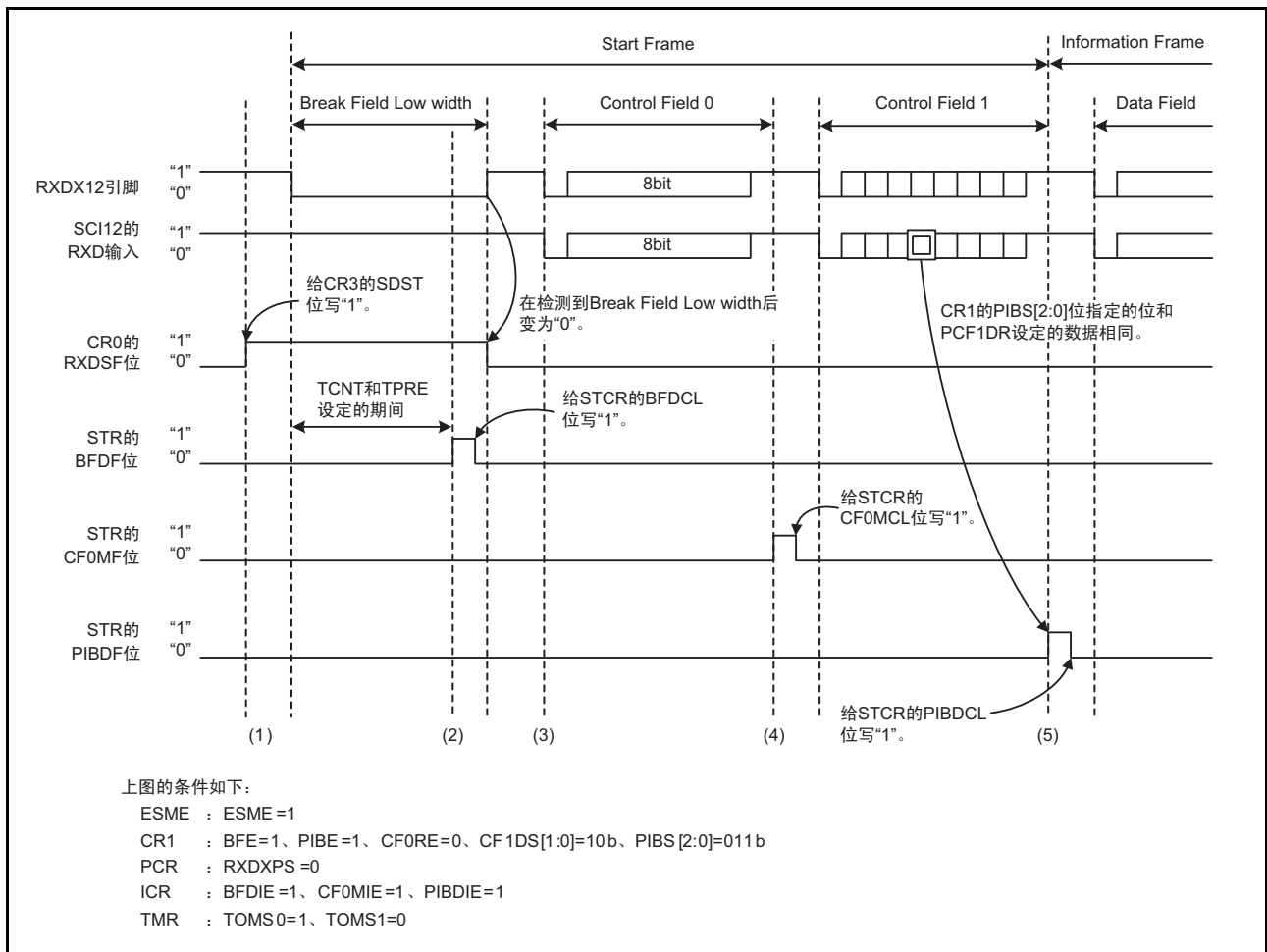


图 28.61 接收 Start Frame 时的运行例子 (使用优先级中断位的情况)

28.9.4 总线冲突检测功能

在ESMER的ESME位和SCI12.SCR的TE位都为“1”的状态下输出Break Field Low width以及通过SCI12发送数据的过程中，总线冲突检测功能有效。

总线冲突检测功能的运行例子如图28.62所示。通过CR2的BCCS[1:0]所设总线冲突检测时钟对TXDX12引脚的输出和RXDX12引脚的输入进行采样，如果连续3次不同，STR的BCDF位就变为“1”。此时，如果ICR的BCDIE位为“1”，就产生SCIX2中断。

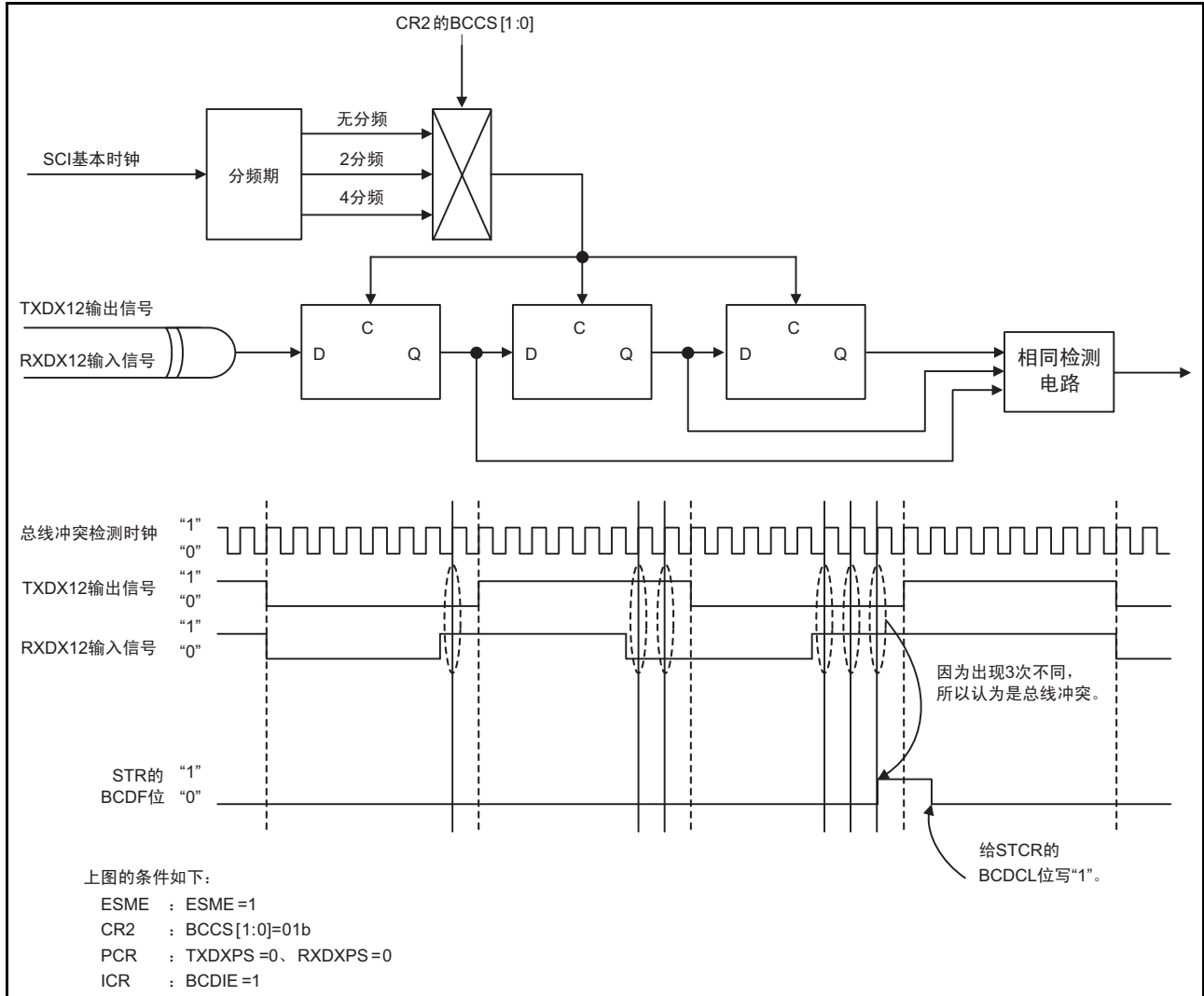


图 28.62 总线冲突检测功能的运行例子

28.9.5 RXDX12 引脚输入的数字滤波器功能

能通过数字滤波器电路将 RXDX12 引脚的输入信号取到内部。数字滤波器电路由 3 段串联的触发电路和相同检测电路构成。通过 CR2 的 DFCS[2:0] 位所选时钟对 RXDX12 引脚的输入信号进行采样，如果 3 个锁存器的输出信号相同，就将此电平传送到后段；如果不相同，就保持以前的值。即，如果至少保持 3 个采样时钟相同的电平就识别为信号，但是如果低于 3 个采样时钟就将信号的变化判断为噪声，而不识别为信号变化。数字滤波器功能的运行例子如图 28.63 所示。

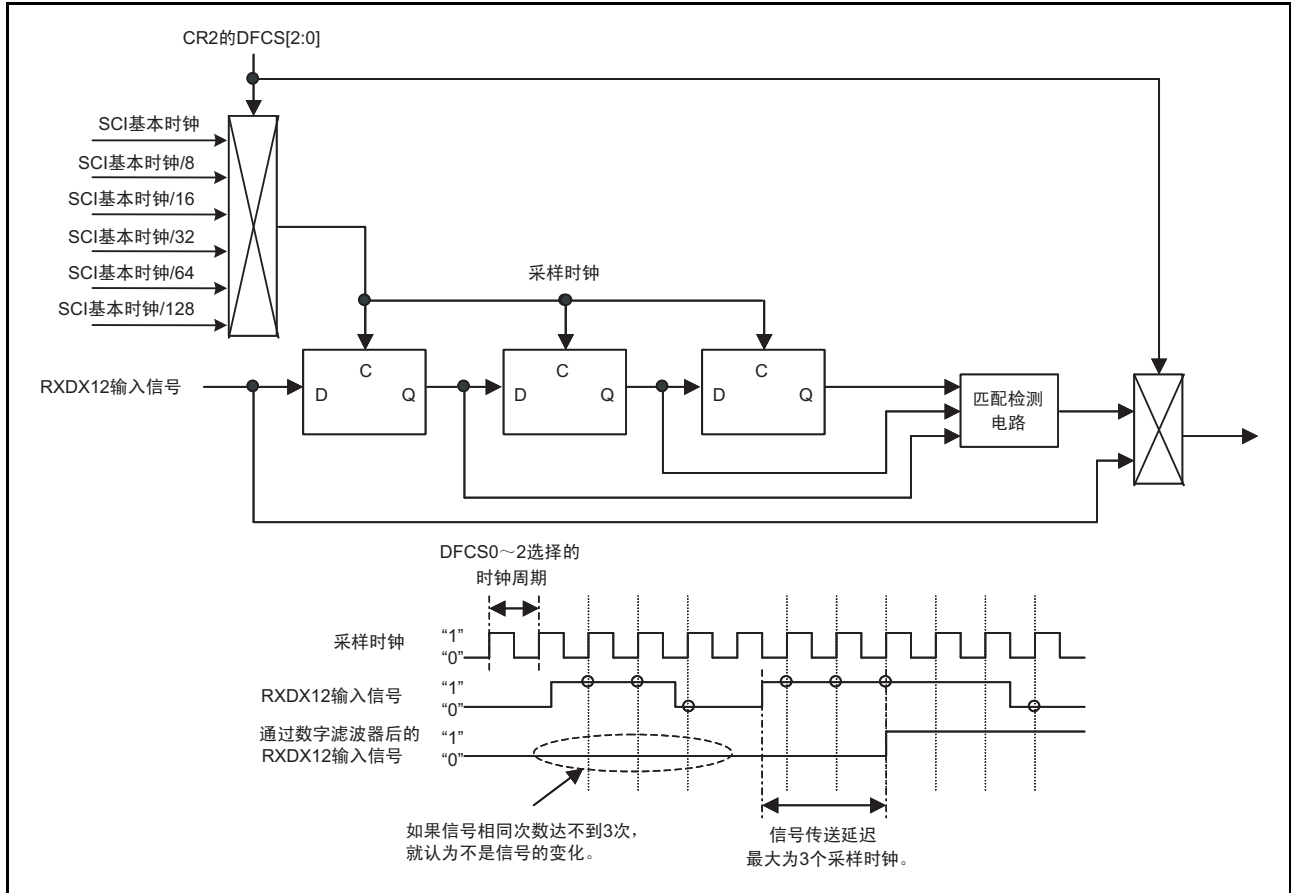


图 28.63 数字滤波器功能的运行例子

28.9.6 位速率测量功能

此功能是测量 RXDX12 引脚输入信号的上升沿和下降沿期间或者下降沿和上升沿期间的功能。位速率测量功能的运行例子如图 28.64 所示。

1. 如果给 CR0 的 BRME 位写“1”，位速率测量就有效。只有在要进行测量时才能将 BRME 位置“1”。即使将 BRME 位置“1”，也不在 Break Field 中测量位速率。
2. 在检测到 Break Field Low width 后，如果 RXDX12 引脚的输入变为 High 电平，就开始测量位速率。
3. 在开始测量位速率后，如果从 RXDX12 引脚输入有效边沿（上升沿和下降沿），定时器就将当时的计数值保持到读缓冲器并且重新加载计数器。此时，如果 ICR 的 AEDIE 位为“1”，就产生 SCIX3 中断。通过读 TCNT 和 TPRES 来解除保持状态。
4. 能从有效边沿期间的计数值计算出位速率，并且通过更改 SCI12 的设定来调整位速率。要在 Control Field 1 相同后将位速率测量功能置为无效时，必须给 CR0 的 BRME 位写“0”。

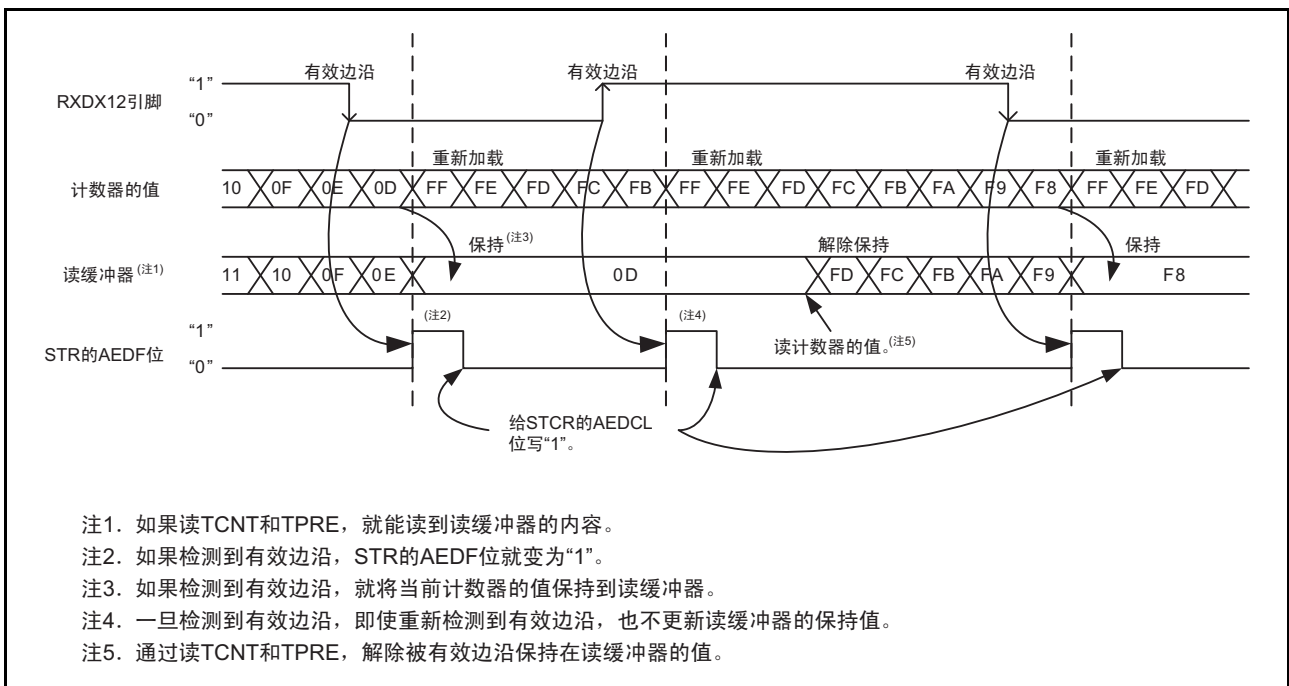


图 28.64 位速率测量功能的运行例子

28.9.7 RXDX12 接收数据的采样时序选择功能

扩展串行模式控制部能通过 CR2 的 RTS0 和 RTS1 从 SCI 基本时钟的 8 个时钟的上升沿、10 个时钟的上升沿、12 个时钟的上升沿或者 14 个时钟的上升沿中选择 SCI12 的 RXDX12 接收数据的采样时序。当 SCI12 的 SEMR 的 ABCS 位为“1”时，能从 PCLK 的 4 个时钟的上升沿、5 个时钟的上升沿、6 个时钟的上升沿和 7 个时钟的上升沿中选择接收数据的采样时序。RXDX12 接收数据的采样时序如图 28.65 所示。

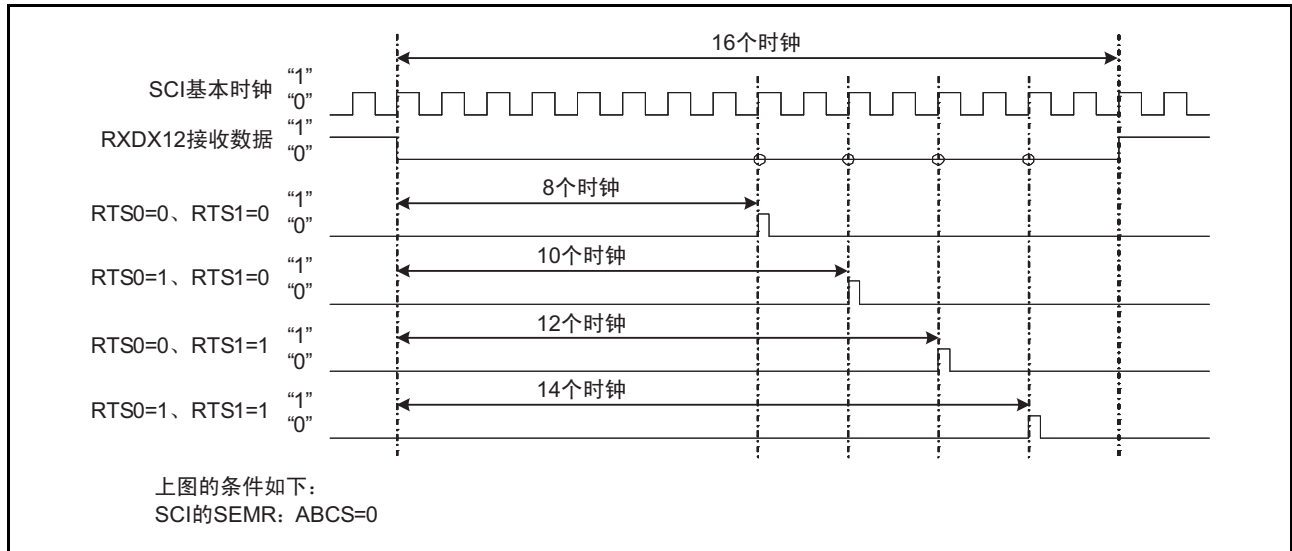


图 28.65 RXDX12 接收数据的采样时序

28.9.8 定时器

定时器有以下运行模式。

(1) Break Field Low width 输出模式

此模式在发送 Start Frame 时从 TXDX12 引脚输出 Break Field Low width 的 Low 电平。如果将 TMR 的 TOMS0 位置“0”并且将 TMR 的 TOMS1 位置“1”，就为 Break Field Low width 输出模式的运行。通过 TMR 的 TCSS[2:0] 位选择计数时钟源。如果给 TCR 的 TCST 位写“1”，就将 TXDX12 引脚的输出置为 Low 电平并且开始计数。如果定时器发生下溢，就将 TXDX12 引脚的输出置为 High 电平并且 STR 的 BFDL 位变为“1”。此时，如果 ICR 的 BFDIE 位为“1”，就产生 SCIX0 中断。如果给 TCR 的 TCST 位写“0”，就在重新加载 TPRES 和 TCNT 后停止计数。必须在 Break Field Low width 输出结束后到定时器再次发生下溢前停止计数。Break Field Low width 输出模式的运行例子如图 28.66 所示。

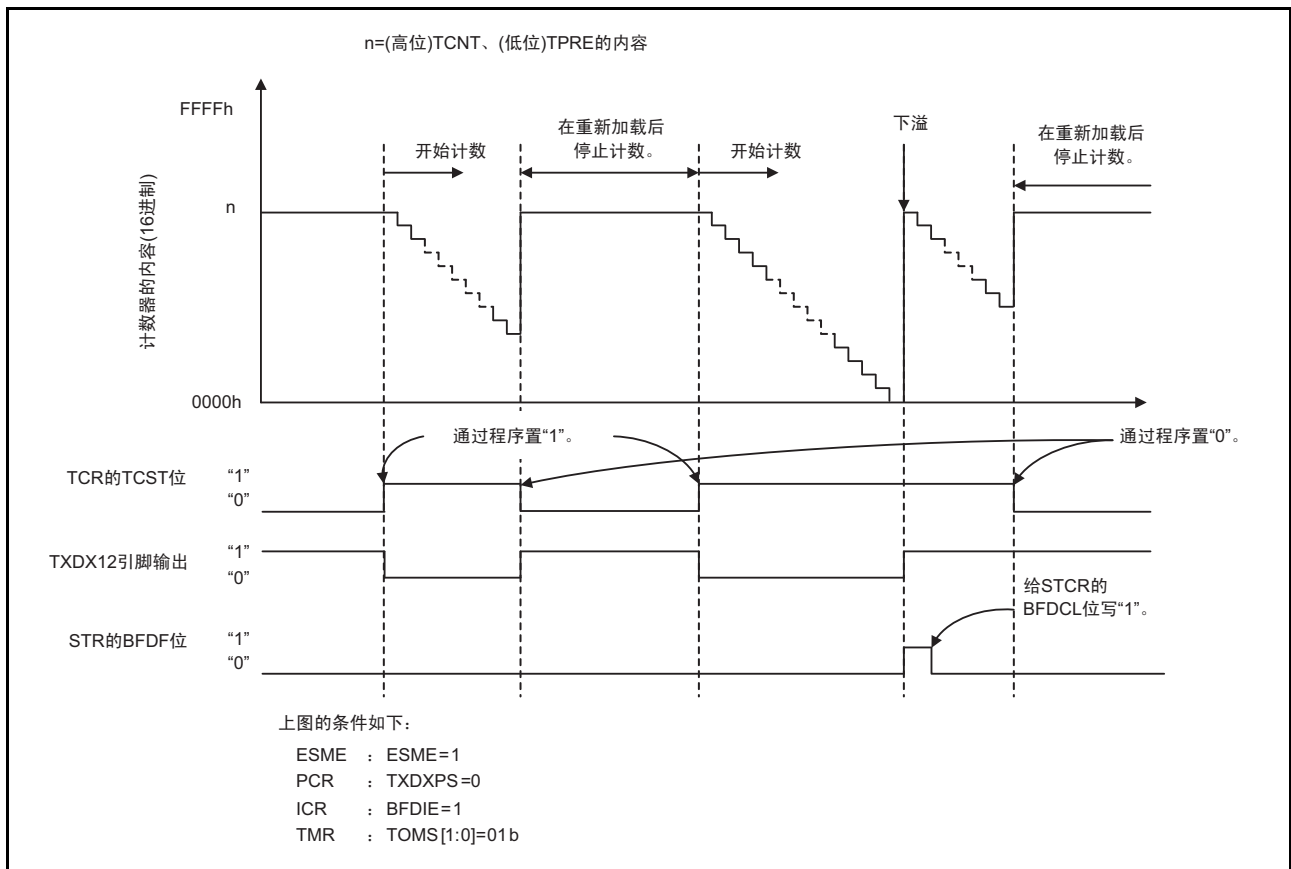


图 28.66 Break Field Low width 输出模式的运行例子

(2) Break Field Low width 判定模式

此模式在接收 Start Frame 时判定从 RXDX12 引脚输入的 Break Field Low width。如果将 TMR 的 TOMS0 位置“1”并且将 TMR 的 TOMS1 位置“0”，就为 Break Field Low width 判定模式的运行。通过 TMR 的 TCSS[2:0] 位选择计数时钟源。如果给 TCR 的 TCST 位写“1”，就进入能判定 Break Field Low width 的状态。如果从 RXDX12 引脚输入 Low 电平，就开始判定；如果从 RXDX12 引脚输入 High 电平，就重新加载 TPRES 和 TCNT 并且进入能判定 Break Field Low width 的状态。如果在 Break Field Low width 的判定过程中定时器发生下溢，STR 的 BFDF 位就变为“1”。此时，如果 ICR 的 BFDIE 位为“1”，就产生 SCIX0 中断。如果在数据通信过程中不能让定时器发生下溢或者发生中断，就必须在判定 Break Field Low width 后停止定时器。Break Field Low width 判定模式的运行例子如图 28.67 所示。

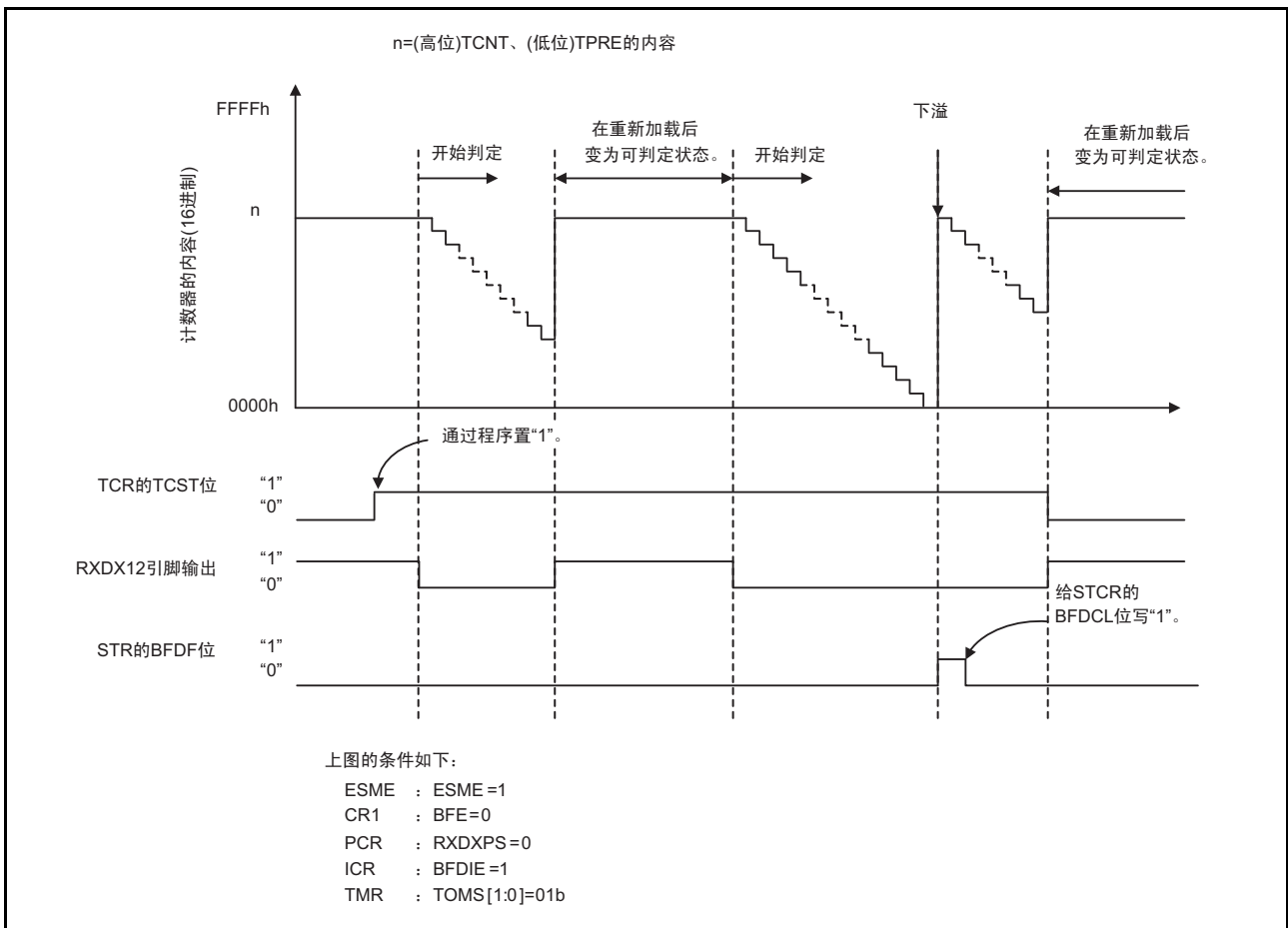


图 28.67 Break Field Low width 判定模式的运行例子

(3) 定时器模式

在此模式中将内部时钟作为计数时钟源。如果将 TMR 的 TOMS0 位和 TMR 的 TOMS1 位都置“0”，就为定时器模式运行。通过 TMR 的 TCSS[2:0] 位选择计数时钟源。如果给 TCR 的 TCST 位写“1”，就开始计数；如果给 TCST 位写“0”，就停止计数。通过输入到 TPRES 的计数时钟源的周期，TPRES 进行递减计数。将 TPRES 的下溢置为计数时钟源，TCNT 进行递减计数。如果定时器发生下溢，STR 的 BFDF 位就变为“1”。此时，如果 ICR 的 BFDIE 位为“1”，就产生 SCIX0 中断。

28.10 噪声消除功能

用于噪声消除功能的噪声滤波器的结构如图 28.68 所示。噪声滤波器由 2 段触发电路和相同检测电路构成。根据设定的采样周期，如果引脚的采样电平 3 次相同，就将相同的电平传送到内部，并且相同电平被持续传送到内部，直到引脚的采样电平再次 3 次相同为止。

在异步模式中，能将噪声消除功能用于 RXDn 的输入信号。采样周期为基本时钟的周期（在 SEMR.ABCS 位为“0”时，为 1 位时间的 1/16 周期；在 SEMR.ABCS 位为“1”时，为 1 位时间的 1/8 周期）。

在简易 I²C 模式中，能将噪声消除功能用于 SSDAn 和 SSCLn 的输入信号。通过设定 SNFR.NFCS[2:0] 位，从内部波特率发生器的时钟源的 1/2/4/8 分频时钟中选择采样时钟。

如果在噪声滤波器有效的状态下停止基本时钟，就在基本时钟重新开始输入时从停止时的噪声滤波器状态开始运行。如果在输入基本时钟期间将 SCR.TE 位和 SCR.RE 位都置“0”，噪声滤波器的触发器就全部被初始化为“1”，当重新开始接收时的输入数据为“1”时，作为检测到相同信号而被传送到内部；当输入数据为“0”时，噪声滤波器的输出保持初始值，直到引脚的采样电平 3 次相同为止。

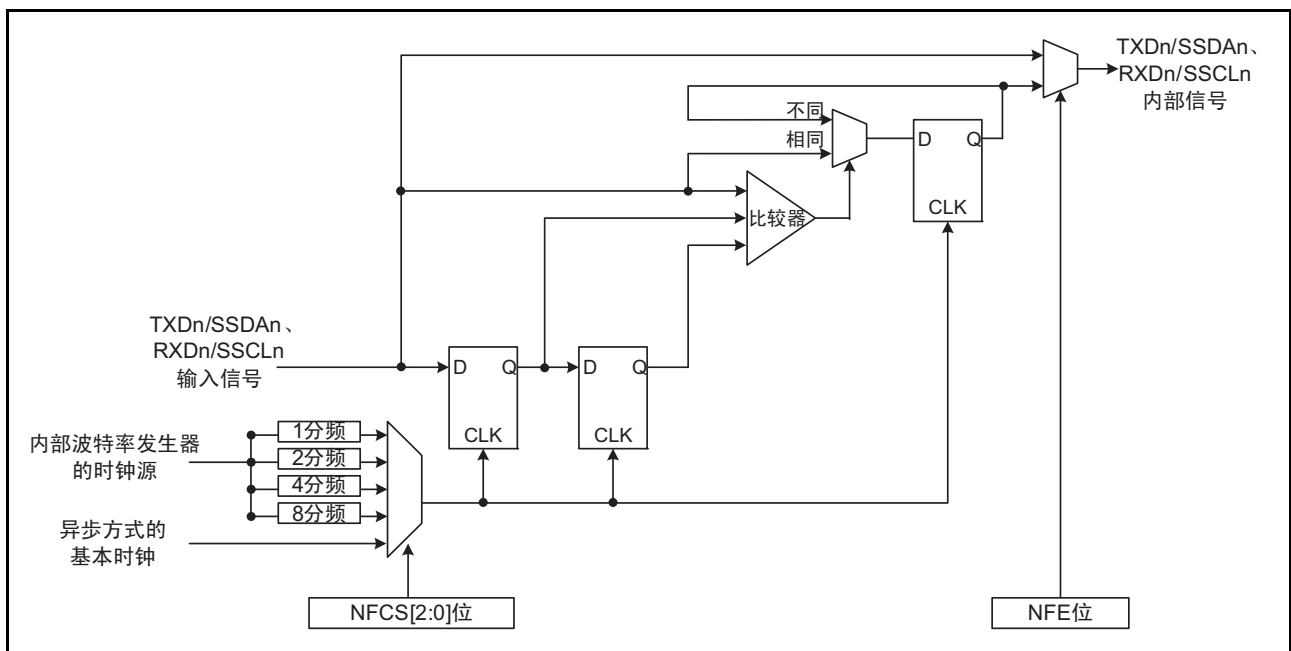


图 28.68 数字噪声滤波器电路的框图

28.11 中断源

28.11.1 TXI 中断和 RXI 中断的缓冲运行

对于 TXI 中断和 RXI 中断，即使在中断控制器的中断状态标志为“1”时满足中断产生条件，也不将中断请求输出到中断控制器而保持在内部（在内部能保持的容量是每个中断源 1 个请求）。

如果中断控制器的中断状态标志为“0”，就将保持的中断请求输出到中断控制器。如果输出被保持的中断请求，就自动清除此中断的内部保持。也能通过将对应的中断允许位（SCR.TIE 位或者 SCR.RIE 位）置“0”来清除保持在内部的中断请求。

28.11.2 串行通信接口模式和简易 SPI 模式的中断

串行通信接口模式和简易 SPI 模式的中断源如表 28.27 所示。各中断源分配有不同的中断向量，能通过 SCR 寄存器的允许位独立允许各中断源。

如果在 SCR.TIE 位为“1”时将发送数据从 TDR 寄存器传送到 TSR 寄存器，就产生 TXI 中断请求。通过在将 SCR.TIE 位置“1”后将 SCR.TE 位置“1”，或者用 1 条指令同时将 SCR.TIE 位和 SCR.TE 位置“1”，也产生 TXI 中断请求。能在通过 TXI 中断请求启动 DTC 或者 DMAC 后进行数据传送。

如果在 SCR.TIE 位为“0”的状态下将 SCR.TE 位置“1”以及在 SCR.TE 位为“1”的状态下将 SCR.TIE 位置“1”，就不产生 TXI 中断请求。（注 1）

在 SCR.TEIE 位为“1”时，如果在发送数据的最后位之前不给 TDR 寄存器写下一个数据，SSR.TEND 标志就变为“1”并且产生 TEI 中断请求。另外，在将 SCR.TE 位置“1”后到给 TDR 寄存器写发送数据为止的期间，SSR.TEND 标志保持“1”的状态，如果将 SCR.TEIE 位置“1”，就产生 TEI 中断请求。

如果给 TDR 寄存器写数据，就清除 SSR.TEND 标志并且取消 TEI 中断请求，但是需要取消时间。

如果在 SCR.RIE 位为“1”时将接受数据保存到 RDR 寄存器，就产生 RXI 中断请求。能在通过 RXI 中断请求启动 DTC 或者 DMAC 后进行数据传送。

如果在 SCR.RIE 位为“1”时将 SSR.ORER 标志、SSR.FER 标志或者 SSR.PER 标志置“1”，就产生 ERI 中断请求。此时，不产生 RXI 中断请求。能通过清除 SSR.ORER、SSR.FER 和 SSR.PER 的全部位，取消 ERI 中断请求。

注 1. 在发送最后的数据时，必须暂时禁止 TXI 中断；在进行发送结束中断处理后要重新开始发送数据时，必须通过 TXI 中断对应的中断控制器的中断请求允许位而不是 SCR.TIE 位，控制中断的禁止和允许。从而能防止 TXI 中断请求（用于发送新数据）的抑止。

表 28.27 SCI 中断源

名称	中断源	中断标志	DTC 的启动	DMAC 的启动	优先级
ERI	接收错误	ORER、FER、PER	不能	不能	高 ↑ 低
RXI	接收数据满	—	能	能	
TXI	发送数据空	—	能	能	
TEI	发送结束	TEND	不能	不能	

28.11.3 智能卡接口模式的中断

智能卡接口模式的中断源如表 28.28 所示，不能使用发送结束中断 (TEI) 请求。

表 28.28 SCI 中断源

名称	中断源	中断标志	DTC 的启动	DMAC 的启动	优先级
ERI	接收错误、错误信号检测	ORER、PER、ERS	不能	不能	高
RXI	接收数据满	—	能	能	↑
TXI	发送数据空	TEND	能	能	低

智能卡接口模式和一般的 SCI 相同，能通过 DTC 或者 DMAC 进行发送和接收。如果在发送时将 SSR.TEND 标志置“1”，就产生 TXI 中断请求。如果预先将 TXI 中断请求设定为 DTC 或者 DMAC 的启动源，就在通过 TXI 中断请求启动 DTC 或者 DMAC 后传送发送数据。在通过 DTC 或者 DMAC 进行数据传送时，TEND 标志就自动变为“0”。

如果发生错误，SCI 就自动重新发送相同的数据。在此期间，TEND 标志保持“0”，不启动 DTC 或者 DMAC。因此，SCI 和 DTC、DMAC 自动发送指定字节数的数据，包括发生错误时的重新发送。但是，在发生错误时，不自动清除 SSR.ERS 标志，因此必须先将 SCR.RIE 位置“1”，然后在发生错误时产生 ERI 中断请求，清除 ERS 标志。

在通过 DTC 或者 DMAC 进行发送和接收时，必须先设定 DTC 或者 DMAC，在设定为允许状态后进行 SCI 的设定。有关 DTC 或者 DMAC 的设定方法，请参照“16. DMA 控制器 (DMACA)”和“17. 数据传送控制器 (DTCa)”。

如果在接收时将接收数据设定到 RDR 寄存器，就产生 RXI 中断请求。如果预先将 RXI 中断请求设定为 DTC 或者 DMAC 的启动源，就在通过 RXI 中断请求启动 DTC 或者 DMAC 后传送接收数据。如果发生错误，就将错误标志置位。因此，不启动 DTC 或者 DMAC 而向 CPU 请求 ERI 中断，所以必须清除错误标志。

28.11.4 简易 I²C 模式的中断

在简易 I²C 模式中，有表 28.29 所示的中断源。STI 中断分配给发送结束中断 (TEI) 请求，不使用接收错误中断 (ERI) 请求。

简易 I²C 模式也能使用 DTC 或者 DMAC 进行发送和接收。

当 SIMR2.IICINTM 位为“1”时，在 SSCLn 引脚的第 8 个下降沿产生 RXI 中断请求。如果预先将 RXI 中断请求设定为 DTC 或者 DMAC 的启动源，就在通过 RXI 中断请求启动 DTC 或者 DMAC 后进行接收数据的传送。另外，在 SSCLn 引脚的第 9 个下降沿 (应答位) 产生 TXI 中断请求。如果预先将 TXI 中断请求设定为 DTC 或者 DMAC 的启动源，就在通过 TXI 中断请求启动 DTC 或者 DMAC 后进行发送数据的传送。

当 SIMR2.IICINTM 位为“0”时，在 SSCLn 引脚的第 9 个下降沿 (应答位) SSDAn 引脚的输入为 Low 电平时，产生 RXI 中断请求 (检测到 ACK)；在 SSDAn 引脚的输入为 High 电平时，产生 TXI 中断请求 (检测到 NACK)。如果预先将 RXI 中断请求设定为 DTC 或者 DMAC 的启动源，就能在通过 RXI 中断请求启动 DTC 或者 DMAC 后进行接收数据或者发送数据的传送。

要使用 DTC 或者 DMAC 进行发送和接收时，必须先设定 DTC 或者 DMAC，在设定为允许状态后设定 SCI。

如果使用 SIMR3.IICSTAREQ、SIMR3.IICRSTAREQ 和 SIMR3.IICSTPREQ 的各位生成开始条件、重新开始条件和停止条件，就在生成结束时产生 STI 中断请求。

表 28.29 SCI 中断源

名称	中断源	中断标志	DTC 的启动	DMAC 的启动	优先级
RXI	接收、ACK 检测	—	能	能	高
TXI	发送、NACK 检测	—	能 (注 1)	能 (注 1)	↑
STI	开始条件、重新开始条件、 停止条件生成结束	IICSTIF	不能	不能	低

注 1. 只有在 SIMR2.IICINTM 位为“1” (选择接收中断和发送中断) 时才能启动 DTC 和 DMAC。

28.11.5 扩展串行模式控制部的中断请求

SCIId 的扩展串行模式控制部生成的中断请求有 SCIX0 中断 (检测到 Break Field Low width)、SCIX1 中断 (Control Field 0 相同, Control Field 1 相同, 检测到优先级中断位)、SCIX2 中断 (检测到总线冲突) 和 SCIX3 中断 (检测到有效边沿) 共 6 种。如果产生各中断源, 状态标志置就变为“1”。各中断请求的内容如表 28.30 所示。

表 28.30 扩展串行模式控制部的中断请求

中断请求	状态标志	中断源
SCIX0 中断 (检测到 Break Field Low width)	BFDF	<ul style="list-style-type: none"> 当检测到长于定时器所设期间的 Break Field Low width 时 在定时器所设期间 Break Field Low width 输出结束时 当定时器发生下溢时
SCIX1 中断 (Control Field 0 相同)	CF0MF	当 Control Field 0 的接收数据和 CF0DR 中设定的数据相同时
SCIX1 中断 (Control Field 1 相同)	CF1MF	当 Control Field 1 的接收数据和 PCF1DR 或者 SCF1DR 中设定的数据相同时
SCIX1 中断 (检测到优先级中断位)	PIBDF	当优先级中断位中指定的位数据和 PCF1DR 中设定的数据相同时
SCIX2 中断 (检测到总线冲突)	BCDF	通过总线冲突检测时钟对 TXDX12 引脚的输出和 RXDX12 引脚的输入进行采样, 连续 3 次不同时
SCIX3 中断 (检测到有效边沿)	AEDF	在测量位数率的过程中检测到有效边沿时

28.12 事件链接功能

通过事件链接控制器 (ELC)，SCI5 能将中断请求信号用作事件信号，并且对事先设定的模块进行链接运行。

与对应的中断请求位的设定无关，输出事件信号。

1. 错误（接收错误和错误信号检测）事件的输出
 - 表示在接收时发生奇偶校验错误并且异常结束。
 - 表示在接收时发生帧错误并且异常结束。
 - 表示在接收时发生溢出错误并且异常结束。
 - 表示在智能卡接口模式的发送时检测到错误信号。

2. 接收数据满事件的输出

- 表示接收数据被设定到接收数据寄存器（RDR 寄存器）。
- 表示在简易 I²C 模式中的 SIMR2.IICINTM 位为“0”时，检测到 ACK。
- 表示在简易 I²C 模式中的 SIMR2.IICINTM 位为“1”时，检测到第 8 位的 SSCL5 引脚的下降沿。

在简易 I²C 模式的主控发送并且 SIMR2.IICINTM 位为“1”时，必须通过设定事件链接控制器来禁止使用接收数据满事件。

3. 发送数据空事件的输出

- 表示 SCR.TE 位从“0”变为“1”。
- 表示发送数据从发送数据寄存器（TDR 寄存器）传送到发送移位寄存器（TSR 寄存器）。
- 表示智能卡接口模式中的发送结束。
- 表示在简易 I²C 模式中的 SIMR2.IICINTM 位为“0”时，检测到 NACK。
- 表示在简易 I²C 模式中的 SIMR2.IICINTM 位为“1”时，检测到第 9 位的 SSCL5 引脚的下降沿。

4. 发送结束事件的输出

- 表示发送结束。
- 表示在简易 I²C 模式中结束开始条件、重新开始条件和停止条件的生成。

28.13 使用时的注意事项

28.13.1 模块停止功能的设定

能通过模块停止控制寄存器 B (MSTPCRB) 和模块停止控制寄存器 C (MSTPCRC) 禁止或者允许 SCI 的运行, 复位后的值为停止 SCI 的运行。通过解除模块停止状态, 使寄存器变为可存取的状态。详细内容请参照“11. 低功耗功能”。

28.13.2 有关中止的检测和处理

在检测到帧错误时, 能通过直接读 RXDn 引脚的值检测中止。因为在中止时 RXDn 引脚的输入全部为“0”, 所以 SSR.FER 标志被置“1” (发生帧错误), 并且 SSR.PER 标志也有可能被置“1” (发生奇偶校验错误)。SCI 在接收到中止后继续进行接收。因此必须注意: 即使将 FER 标志置“0” (未发生帧错误), FER 标志也会再次变为“1”。

28.13.3 标记状态和中止的发送

当 SCR.TE 位为“0” (禁止串行发送) 时, 能通过设定 I/O 端口功能, 将 TXDn 引脚用作能选择输入 / 输出方向和电平的 I/O 端口。能通过此变化将 TXDn 引脚置为标记状态或者在发送数据时发送中止。

为了在将 SCR.TE 位置“1” (允许串行发送) 前将通信线路置为标记状态 (“1” 的状态), 通过设定 I/O 端口功能将 TXDn 引脚设定为输出 High 电平。另外, 要在发送数据时发送中止的情况下, 必须在通过设定 I/O 端口功能将 TXDn 引脚设定为输出 Low 电平后将 SCR.TE 位置“0”。如果将 SCR.TE 位置“0”, 就对发送部进行初始化, 与当前的发送状态无关, TXDn 引脚变为 I/O 端口, 并且根据 I/O 端口功能的设定, 从 TXDn 引脚输出 Low 电平或者 High 电平。

28.13.4 有关接收错误标志和发送 (只限于时钟同步模式)

在接收错误标志 (SSR.ORER) 为“1” 的状态下, 即使给 TDR 寄存器写数据, 也不能开始发送。在开始发送时, 必须将接收错误标志置“0”。必须注意: 即使将 SCR.RE 位置“0” (禁止串行接收), 也不能将接收错误标志置“0”。

28.13.5 有关 TDR 寄存器的写操作

能随时给 TDR 寄存器写数据。但是, 如果在 TDR 寄存器留有发送数据的状态下给 TDR 寄存器写新数据, 保存在 TDR 寄存器的数据就可能没有被传送到 TSR 寄存器而丢失。因此, 必须通过 TXI 中断请求给 TDR 寄存器写发送数据。

28.13.6 时钟同步发送时的限制事项

如果将外部时钟源用于同步时钟, 就必须在通过 DMAC 或者 DTC 更新 TDR 寄存器并且通过 PCLK 时钟至少经过5个时钟后, 输入发送时钟。如果在更新 TDR 寄存器后的4个时钟以内输入发送时钟, 就可能发生误动作。

28.13.7 使用 DMAC 或者 DTC 时的限制事项

在通过 DMAC 或者 DTC 读 RDR 寄存器时, 必须将该 SCI 的接收结束中断 (RXI) 设定为启动源。

28.13.8 有关开始通信的注意事项

如果在开始通信时中断控制器的中断状态标志为“1”，就必须在允许运行（将 SCR.TE 位或者 SCR.RE 位置“1”）前按以下步骤清除中断请求：

- 确认通信处于停止状态（SCR.TE 位或者 SCR.RE 位为“0”）。
- 将对应的中断允许位（SCR.TIE 位或者 SCR.RIE 位）置“0”。
- 读对应的中断允许位（SCR.TIE 位或者 SCR.RIE 位），确认是“0”。
- 将中断控制器的中断状态标志置“0”。

28.13.9 有关低功耗状态时的运行

(1) 发送

必须在停止运行（SCR.TIE 位 =0、SCR.TE 位 =0、SCR.TEIE 位 =0）的状态下设定为模块停止状态或者向软件待机模式转移。通过清除 TE 位，对 TSR 寄存器、TDR 寄存器和 SSR 寄存器进行复位。模块停止状态以及软件待机模式时的输出引脚的状态取决于端口的设定，在解除模块停止状态或者软件待机模式后，为 High 电平输出。如果在发送过程中进行转移，发送中的数据就不确定。

在解除低功耗状态后不更改发送模式而进行发送时，就必须按照将 TE 位置“1”以及读 SSR 寄存器 → 写 TDR 寄存器的顺序开始发送。如果在更改发送模式后进行发送，就必须在进行初始设定后开始发送。

发送时的软件待机模式转移的流程图例子如图 28.69 所示，软件待机模式转移时的端口的引脚状态如图 28.70 和图 28.72 所示。

必须在停止运行（TE 位 =0）的状态下，从通过 DTC 传送进行的发送模式设定为模块停止状态或者转移到软件待机模式。在解除这些模式后通过 DTC 进行发送时，如果将 TE 位置“1”，就产生 TXI 中断并且开始通过 DTC 进行的发送。

(2) 接收

必须在停止接收（SCR.RE 位 =0）的状态下设定为模块停止状态或者向软件待机模式转移。如果在接收时进行转移，正在接收的数据就无效。

在解除低功耗状态后不更改接收模式而进行接收时，必须通过将 RE 位置“1”，开始接收。如果在更改接收模式后进行接收，就必须在进行初始设定后开始接收。

接收时的软件待机模式转移的流程图例子如图 28.72 所示。

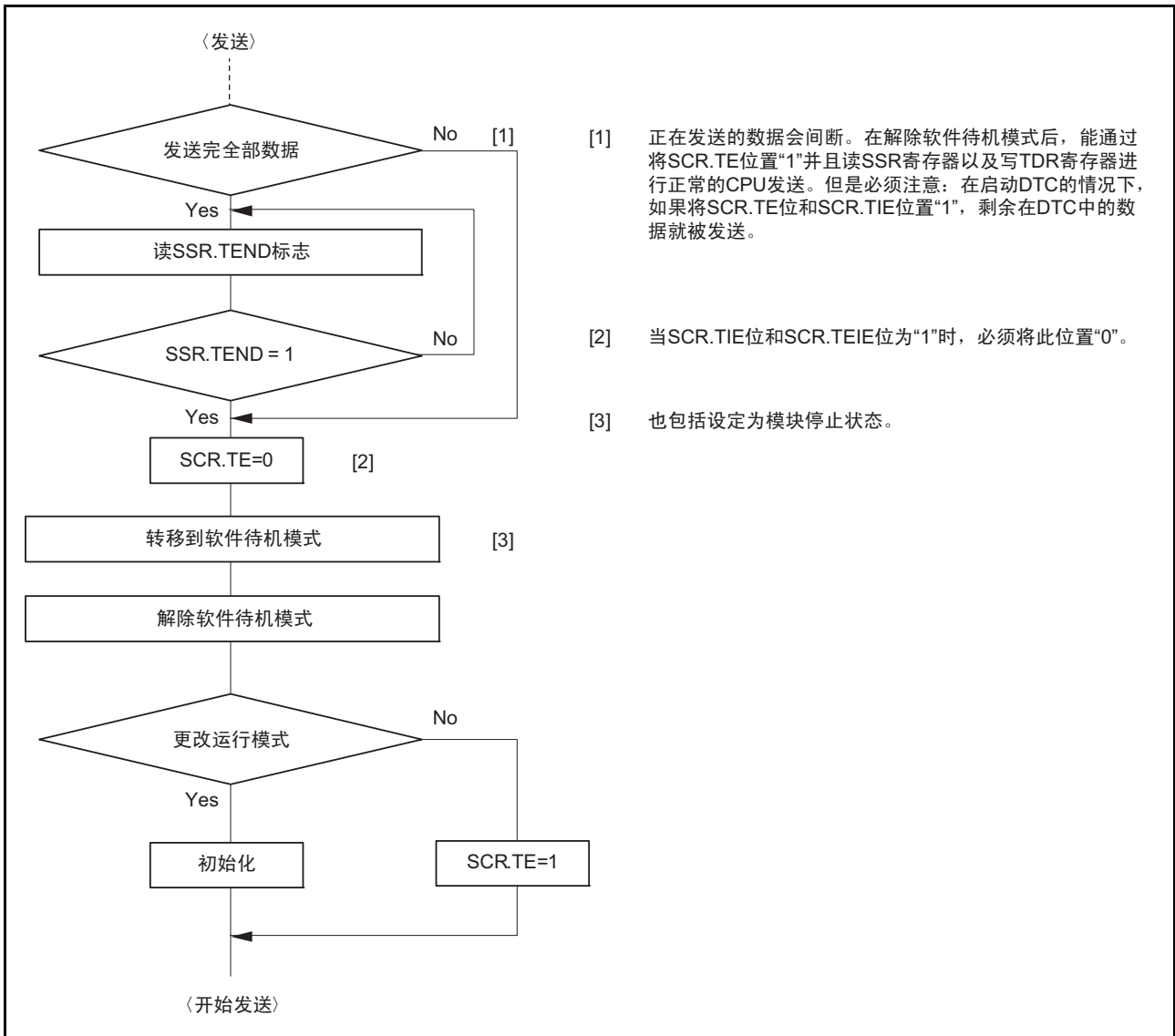


图 28.69 发送时向软件待机模式转移的流程图例子

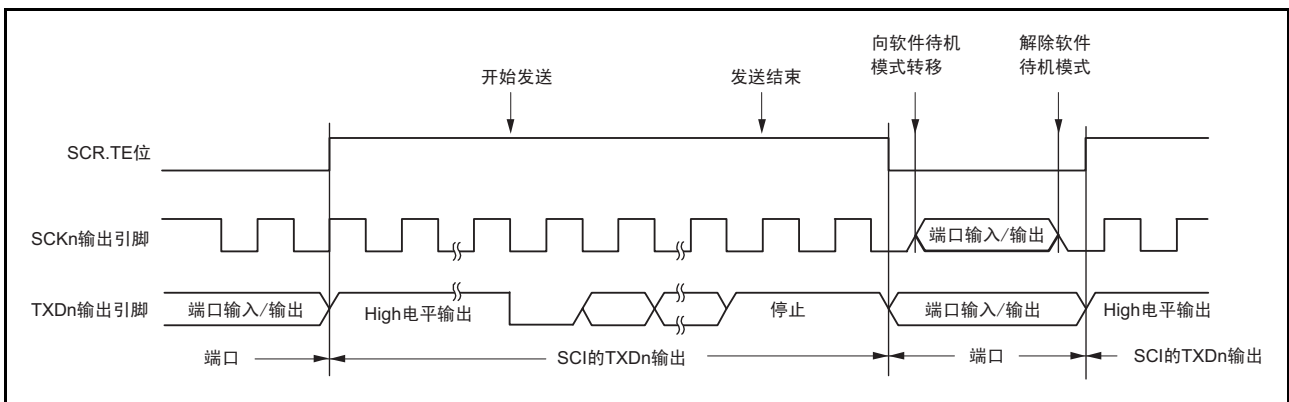


图 28.70 向软件待机模式转移时的端口引脚状态 (内部时钟、异步发送)

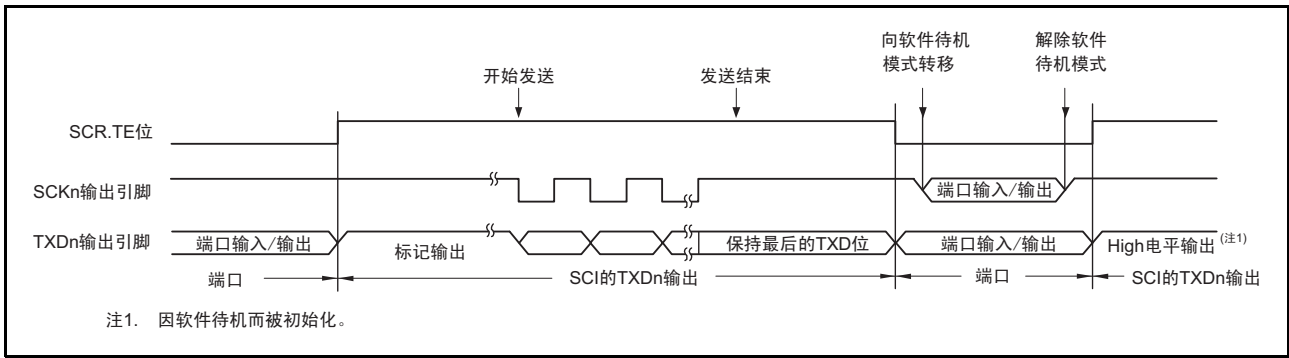


图 28.71 向软件待机模式转移时的端口引脚状态 (内部时钟、时钟同步发送)

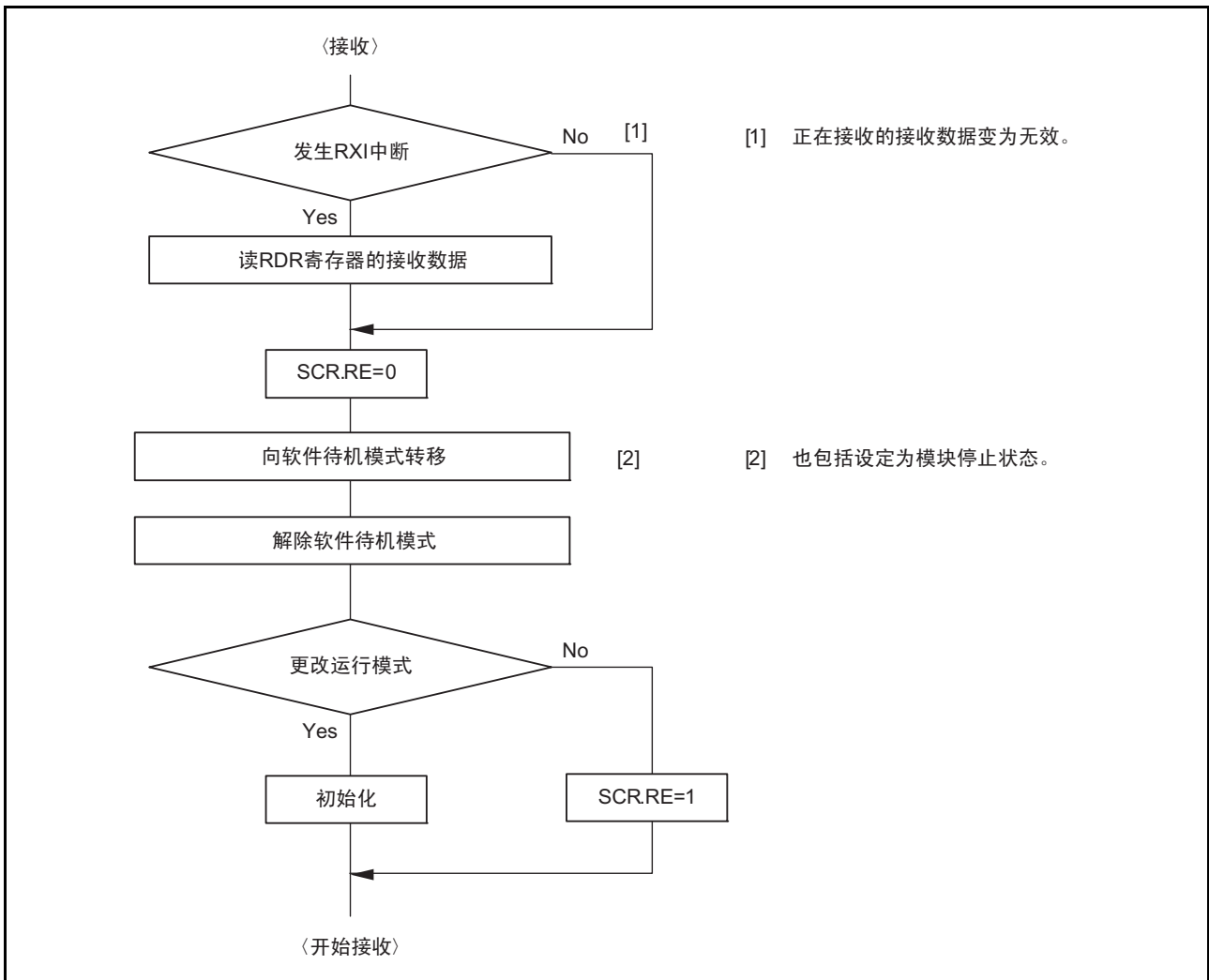


图 28.72 接收时向软件待机模式转移的流程图例子

28.13.10 时钟同步模式的外部时钟输入

在时钟同步模式中，外部时钟的 SCKn 输入必须满足：High 电平脉冲期间和 Low 电平脉冲期间必须至少为 2 个 PCLK，周期至少为 6 个 PCLK。

28.13.11 简易 SPI 模式的限制事项

(1) 主控模式

- 必须配合由 SPMR.CKPH 位和 SPMR.CKPOL 位设定的发送和接收时钟的初始值，用电阻上拉（下拉）时钟线。
- 当设定为有时钟延迟（SPMR.CKPH 位=1）时，如图 28.73 所示，在 SCKn 引脚的最后时钟的前一个边沿产生接收数据满中断（RXI 中断）。此时，必须注意：如果立即将 SCR.TE 位和 SCR.RE 位置“0”，SCKn 引脚的输出就变为高阻抗，最后发送和接收时钟的时钟脉宽变短。另外，还必须注意：如果在发生 RXI 中断后立即将连接从属设备的 SSn# 引脚输入信号置为 High 电平，从属设备就可能发生误动作。
- 在多主控模式中，如果在发送和接收字符的中途发生模式故障错误，就在 SS# 引脚的输入为 Low 电平期间 SCKn 引脚的输出为高阻抗，并且停止给连接的从属设备提供发送和接收时钟。为了避免在重新开始发送和接收时发生位的错位，必须重新设定连接的从属设备。

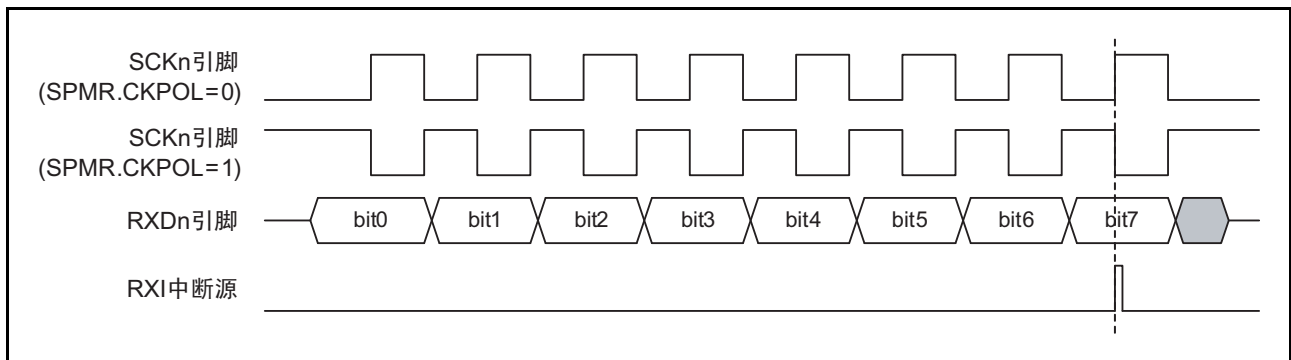


图 28.73 简易 SPI 模式（有时钟延迟）的 RXI 中断产生时序

(2) 从属模式

- 从给 TDR 寄存器写发送数据并且给 SSn# 引脚输入 Low 电平到开始输入外部时钟为止，必须至少确保 5 个 PCLK 的时间。
- 由主控提供的外部时钟必须和传送数据的长度相同。
- 必须在数据传送的开始前和结束后控制 SSn# 引脚的输入。
- 如果在发送和接收字符的中途将 SSn# 引脚的输入从 Low 电平变为 High 电平，就必须将 SCR.TE 位和 SCR.RE 位置“0”，并且在设定后从第 1 字节开始重新传送。

28.13.12 扩展串行模式控制部的使用限制事项 1

如果将 PCR 的 SHARPS 位置“1”，就在以下情况下 TXDX12/RXDX12 引脚变为输出引脚：

- 当 SCId 定时器运行在 Break Field Low width 输出模式中并且将 TCR 的 TCST 位置“1”时（从将 TCR 的 TCST 位置“1”到输出 Low 电平为止，最多输出 1 个定时器计数时钟源周期的 High 电平）
- 当 SCI12.SCR 的 TE 位为“1”时

28.13.13 扩展串行模式控制部的使用限制事项 2

即使将扩展串行模式置为有效，也生成 SCIc 的中断请求。因为 SCIId 在接收起始帧的过程中使用 SCIc 的中断请求事件，所以不能使用 SCIc 的中断请求。有以下 2 种对应方法。而且，在检测到接收错误时，必须按照图 28.74 的流程图的例子，清除 SCIc 的错误标志并且对 SCIId 控制部进行初始化。

1. 必须将 SCIc 的 SCR.RIE 位置“0”，禁止输出中断请求。此时，因为在发生接收错误时不产生 ERI 中断，所以必须在接收完起始帧时检查 SCIc 的 SSR 寄存器的错误标志。在从接收完起始帧到接收完信息帧的第 1 字节为止的期间，必须将 SCIc 的 SCR.RIE 位置“1”。
2. 必须将 SCIc 的 SCR.RIE 位置“1”，禁止 ICU 的 RXI 中断而允许 ICU 的 ERI 中断。在从接收完起始帧到接收完信息帧的第 1 字节为止的期间，必须清除 ICU 的 RXI 中断对应的 IRn.IR 标志，允许 ICU 的 RXI 中断。

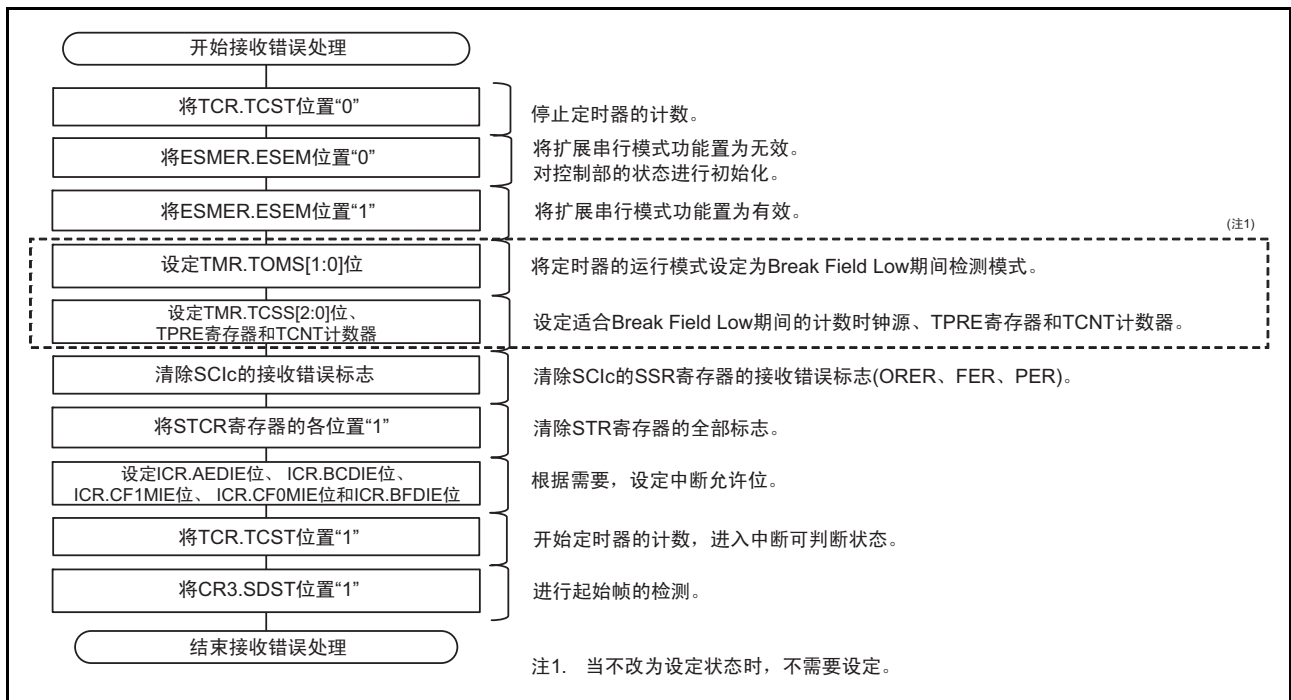


图 28.74 接收错误处理的流程图例子（接收起始帧的过程中）

29. I²C 总线接口 (RIIC)

RX210 群内置 1 个通道的 I²C 总线接口 (RIIC)。

RIIC 以 NXP 公司提倡的 I²C 总线 (Inter-IC-Bus) 接口方式为基准, 内置了子集功能。

29.1 概要

RIIC 的规格和框图分别如表 29.1 和图 29.1 所示, 输入 / 输出引脚的外部电路连接例子 (I²C 总线的结构例子) 如图 29.2 所示, RIIC 使用的输入 / 输出引脚如表 29.2 所示。

表 29.1 RIIC 的规格

项目	内容
通信格式	<ul style="list-style-type: none"> I²C 总线格式或者 SMBus 格式 能选择主控模式或者从属模式。 自动确保与所设传送速度对应的各种准备时间、保持时间和总线空闲时间。
传送速度	~ 400kbps
SCL 时钟	在 主控模式中, 能将 SCL 时钟的占空比设定在 4% ~ 96% 的范围内。
条件发行和条件检测	自动生成开始条件、重新开始条件和停止条件, 并且能检测到开始条件 (包括重新开始条件) 和停止条件。
从属地址	<ul style="list-style-type: none"> 能设定 3 组从属地址。 对应 7 位或者 10 位的地址格式 (能同时存在)。 能检测到全呼地址、设备 ID 地址和 SMBus 的主机地址。
应答	<ul style="list-style-type: none"> 在发送时自动装入应答位。 能在接收 NACK 时自动中止下一个发送数据的传送。 在接收时自动发送应答位。 如果选择在第 8 个时钟和第 9 个时钟之间有等待, 就能通过软件控制与接收数据内容对应的应答位。
等待功能	<ul style="list-style-type: none"> 在接收时, 能通过保持 SCL 时钟的 Low 电平进行等待。 在第 8 个时钟和第 9 个时钟之间等待。 在第 9 个时钟和下次传送的第 1 个时钟之间等待 (WAIT 功能)。
SDA 输出延迟功能	能延迟数据发送 (包括发送 ACK) 的输出时序。
仲裁	<ul style="list-style-type: none"> 对应多主控 在和其他主控发生 SCL 时钟冲突时, 能与 SCL 时钟同步运行。 在发生开始条件的发行竞争时, 如果 SDA 线上的信号状态不同, 就能检测到仲裁失败。 在 主控模式中, 能在发送数据不同时检测到仲裁失败。 如果在总线忙时发行开始条件, 就能检测到仲裁失败 (防止双重发行开始条件)。 在发送 NACK 时, 如果 SDA 线上的信号状态不同, 就能检测到仲裁失败。 在从属发送模式中, 能在数据不同时检测到仲裁失败。
超时检测功能	能通过内部超时检测功能检测 SCL 时钟的长时间停止。
噪声消除	SCL 输入和 SDA 输入内置数字噪声滤波器, 噪声消除宽度为可编程调整。
中断源	<ul style="list-style-type: none"> 4 种 通信错误 / 事件的发生 (AL 检测、NACK 检测、超时检测、开始条件检测 (包括重新开始条件)、停止条件检测) 接收数据满 (包括从属地址匹配的情况) 发送数据空 (包括从属地址匹配的情况) 发送结束
低功耗功能	能设定为模块停止状态。
事件链接功能 (仅支持 RIIC0)	<ul style="list-style-type: none"> 4 种 发生通信错误 / 事件 (AL 检测、NACK 检测、超时检测、开始条件检测 (包括重新开始条件)、停止条件检测) 接收数据满 (包括从属地址匹配的情况) 发送数据空 (包括从属地址匹配的情况) 发送结束

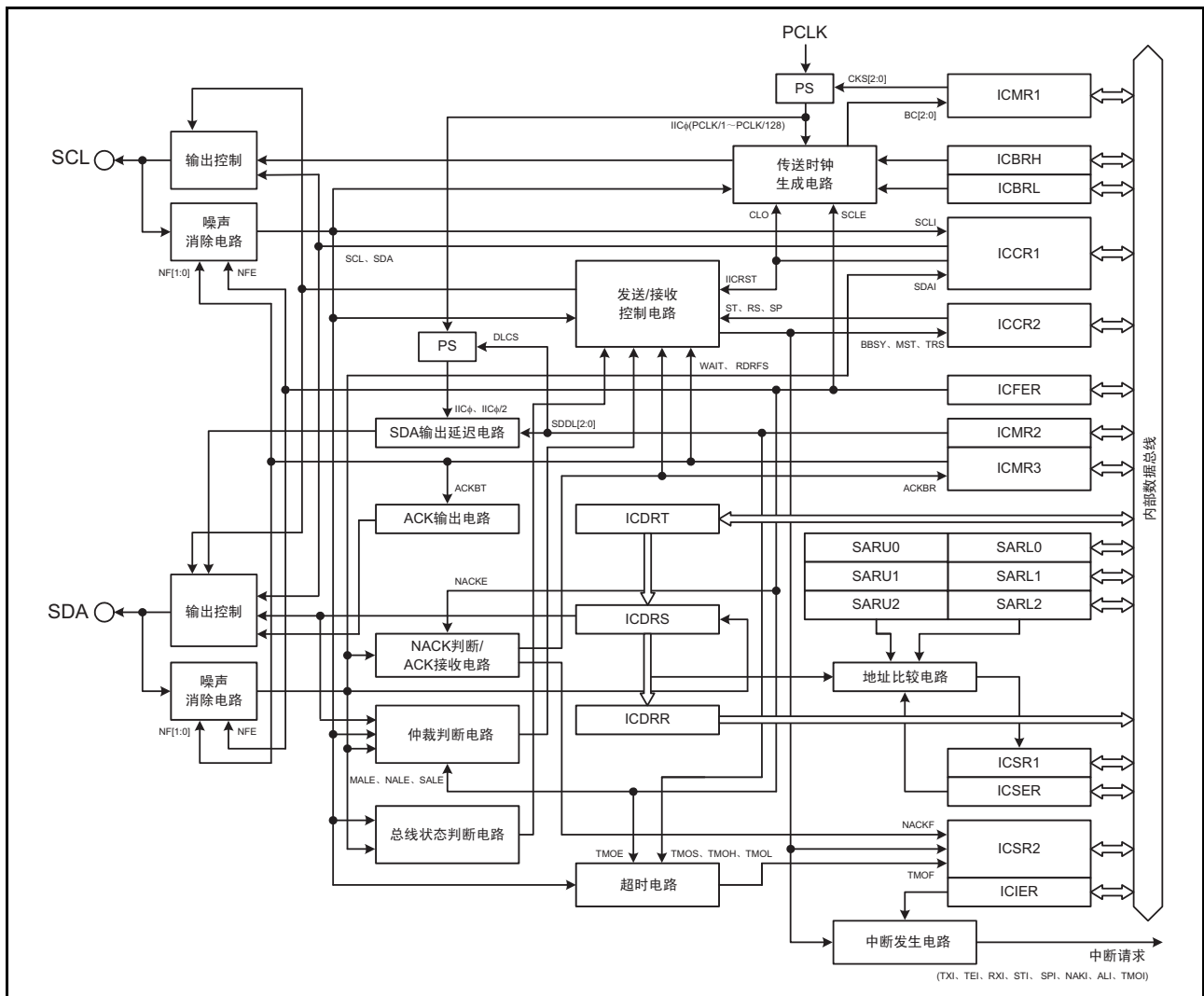


图 29.1 RIIC 的框图

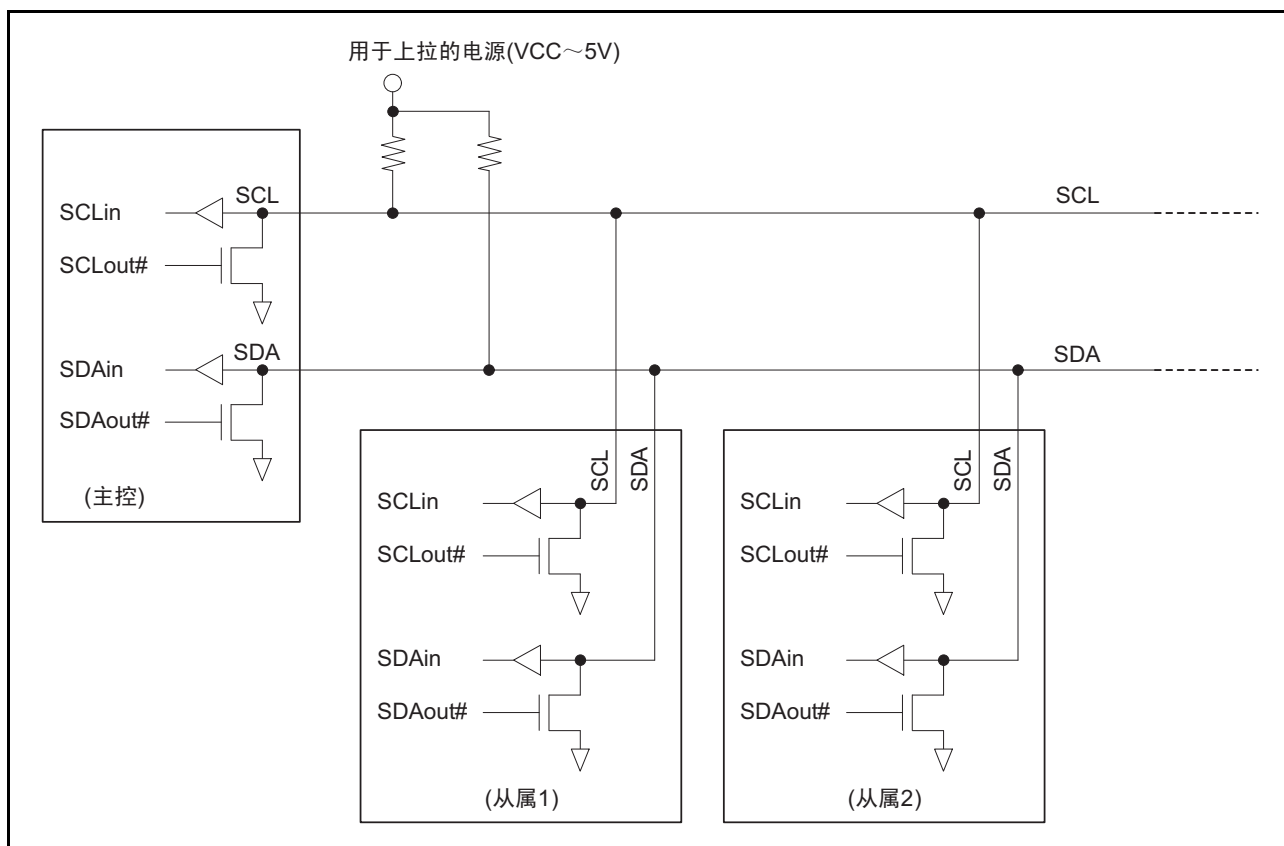


图 29.2 输入 / 输出引脚的外部电路连接例子 (I²C 总线的结构例子)

当选择 I²C 总线时 (SMBS 位 =0)，RIIC 各信号的输入电平为 CMOS 电平；当选择 SMBus 时 (SMBS 位 =1)，RIIC 各信号的输入电平为 TTL 电平。

表 29.2 RIIC 的输入 / 输出引脚

通道	引脚名	输入 / 输出	功能
RIIC0	SCL	输入 / 输出	RIIC0 串行时钟的输入 / 输出引脚
	SDA	输入 / 输出	RIIC0 串行数据的输入 / 输出引脚

29.2 寄存器说明

29.2.1 I²C 总线控制寄存器 1 (ICCR1)

地址 RIIC0.ICCR1 0008 8300h

	b7	b6	b5	b4	b3	b2	b1	b0
	ICE	IICRST	CLO	SOWP	SCLO	SDAO	SCLI	SDAI
复位后的值	0	0	0	1	1	1	1	1

位	符号	位名	功能	R/W
b0	SDAI	SDA 总线输入监视位	0: SDA 引脚输入为 Low 电平 1: SDA 引脚输入为 High 电平	R
b1	SCLI	SCL 总线输入监视位	0: SCL 引脚输入为 Low 电平 1: SCL 引脚输入为 High 电平	R
b2	SDAO	SDA 输出控制位	<ul style="list-style-type: none"> 读时 0: SDA 引脚为 Low 电平输出 1: SDA 引脚为高阻抗 写时 0: 将 SDA 引脚改为 Low 电平 1: 将 SDA 引脚改为高阻抗 (通过外部上拉电阻输出 High 电平) 	R/W (注1、注2)
b3	SCLO	SCL 输出控制位	<ul style="list-style-type: none"> 读时 0: SCL 引脚为 Low 电平输出 1: SCL 引脚为高阻抗 写时 0: 将 SCL 引脚改为 Low 电平输出 1: 将 SCL 引脚改为高阻抗 (通过外部上拉电阻输出 High 电平) 	R/W (注1、注2)
b4	SOWP	SCLO/SDAO 写保护位	0: 设定 SCLO 位和 SDAO 位的值 (读取值为“1”。)	R/W (注2)
b5	CLO	SCL 时钟的追加输出位	0: 不追加输出 SCL 时钟 (正常状态) 1: 追加输出 SCL 时钟 (在输出 1 个时钟后, 此位自动变为“0”。)	R/W
b6	IICRST	I ² C 总线接口内部复位的位	0: 解除 RIIC 复位或者内部复位 1: RIIC 复位或者内部复位状态 (清除位计数器并且解除 SCL/SDA 的输出锁存)	R/W
b7	ICE	I ² C 总线接口允许位	0: 禁止 (SCL 引脚、SDA 引脚为非驱动状态) 1: 允许 (SCL 引脚、SDA 引脚为驱动状态) • 通过与 IICRST 位的组合选择 RIIC 复位或者内部复位	R/W

注 1. 不能在通信过程中写此位。如果在通信过程中更改此位的值, 就可能发生发送 / 接收异常或者 AL 错误。

注 2. 必须在将 SOWP 位置“0”的同时改写 SDAO 位和 SCLO 位。

CLO 位 (SCL 时钟的追加输出位)

此位具有以 1 个时钟为单位追加输出 SCL 时钟的功能，用于调试或者异常处理。

在一般情况下，必须将此位置“0”。如果在正常的通信过程中使用此位，就可能引起通信错误。

有关此功能的详细内容，请参照“29.11.2 SCL 时钟追加输出功能”。

IICRST 位 (I²C 总线内部复位的位)

此位对 RIIC 的内部状态进行复位。

如果将 IICRST 位置“1”，就能进行 RIIC 复位或者内部复位。

RIIC 复位和内部复位取决于 IICRST 位和 ICE 位的组合。RIIC 复位的种类如表 29.3 所示。

RIIC 复位是指对包括 ICCR2.BBSY 标志在内的全部寄存器和内部状态进行复位；内部复位是指对位计数器 (ICMR1.BC[2:0] 位)、I²C 总线移位寄存器 (ICDRS)、I²C 总线状态寄存器 (ICSR1 和 ICSR2) 和内部状态进行复位。有关各寄存器的复位状况，请参照“29.14 复位状况”。

如果在运行时 (ICE 位为“1”的状态) 因通信故障等引起总线和 RIIC 发生意外停机的情况下将 IICRST 位置“1”，就能在不对端口的设定、RIIC 的各控制寄存器和设定寄存器进行初始化的情况下对 RIIC 的内部状态进行复位。

如果在 RIIC 输出 Low 电平的状态下发生意外停机，就能通过对内部状态进行复位，将 SCL 引脚 /SDA 引脚置为高阻抗，然后释放总线。

注 1. 在从属模式中和主控设备进行通信时，如果因总线发生意外停机而通过 IICRST 位进行内部复位，RIIC 就可能变为和主控设备的状态不同的状态 (主要是双方的位计数器信息产生差异)，因此原则上不在从属模式中进行内部复位而从主控设备进行恢复处理。在从属模式中将 SCL 线置为 Low 电平输出的状态下，如果 RIIC 发生意外停机而需要内部复位，就必须在进行内部复位后从主控设备发行重新开始条件，或者在发行停止条件后发行开始条件，重新开始通信。如果只单独对从属设备进行复位，并且在没有从主控设备发行开始条件或者重新开始条件的情况下重新开始通信，就可能因双方运行状态的差异而导致不同步。

表 29.3 RIIC 复位的种类

IICRST	ICE	状态	内容
1	0	RIIC 复位	对 RIIC 的全部寄存器和内部状态进行复位。
	1	内部复位	对 ICMR1.BC[2:0] 位、ICSR1、ICSR2、ICDRS 寄存器和内部状态进行复位。

ICE 位 (I²C 总线接口允许位)

此位选择 SCL 引脚和 SDA 引脚为驱动状态或者非驱动状态。另外能通过与 IICRST 位的组合，进行 2 种复位。有关 RIIC 复位的种类，请参照“表 29.3 RIIC 复位的种类”。

在使用 RIIC 时，必须将 ICE 位置“1”。当 ICE 位为“1”时，SCL 引脚和 SDA 引脚为驱动状态。

在不使用 RIIC 时，必须将 ICE 位置“0”。当 ICE 位为“0”时，SCL 引脚和 SDA 引脚为非驱动状态。另外，在设定多功能引脚控制器 (MPC) 时不能将 SCL 引脚和 SDA 引脚分配到 RIIC。必须注意：如果被分配到 RIIC，就进行从属地址比较运行。

29.2.2 I²C 总线控制寄存器 2 (ICCR2)

地址 RIIC0.ICCR2 0008 8301h

b7	b6	b5	b4	b3	b2	b1	b0
BBSY	MST	TRS	—	SP	RS	ST	—
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	—	保留位	读写值都为“0”。	R/W
b1	ST	开始条件发行请求位	0: 不请求发行开始条件 1: 请求发行开始条件	R/W
b2	RS	重新开始条件发行请求位	0: 不请求发行重新开始条件 1: 请求发行重新开始条件	R/W
b3	SP	停止条件发行请求位	0: 不请求发行停止条件 1: 请求发行停止条件	R/W
b4	—	保留位	读写值都为“0”。	R/W
b5	TRS	发送 / 接收模式位	0: 接收模式 1: 发送模式	R/W (注 1)
b6	MST	主控 / 从属模式位	0: 从属模式 1: 主控模式	R/W (注 1)
b7	BBSY	总线忙检测标志	0: I ² C 总线为释放状态 (总线空闲状态) 1: I ² C 总线为占有状态 (总线忙状态或者总线处于空闲期间)	R

注 1. 当 ICMR1.MTWP 位为“1”时, 能写 MST 位和 TRS 位。

ST 位 (开始条件发行请求位)

此位请求向主控模式的转移以及开始条件的发行。

如果 ST 位为“1”, 就请求发行开始条件, 并且在 BBSY 标志为“0” (总线空闲) 时发行开始条件。有关发行开始条件的详细内容, 请参照“29.10 开始条件、重新开始条件和停止条件的发行功能”。

[为“1”的条件]

- 当写“1”时

[为“0”的条件]

- 当写“0”时
- 当结束开始条件的发行时
- 当 ICSR2.AL 标志变为“1” (仲裁失败) 时
- 当给 ICCR1.IICRST 位写“1”并且进行 RIIC 复位或者内部复位时

注 1. 必须在 BBSY 标志为“0” (总线空闲) 时将 ST 位置“1” (请求发行开始条件)。

必须注意: 如果在 BBSY 标志为“1” (总线忙) 时将 ST 位置“1” (请求发行开始条件), 就作为开始条件的发行错误而产生仲裁失败。

RS 位 (重新开始条件发行请求位)

此位在主控模式中请求发行重新开始条件。

如果 RS 位为“1”，就请求发行重新开始条件，在 BBSY 标志为“1”（总线忙）并且 MST 位为“1”（主控模式）时，发行重新开始条件。

有关发行重新开始条件的详细内容，请参照“29.10 开始条件、重新开始条件和停止条件的发行功能”。

[为“1”的条件]

- 在 ICCR2.BBSY 标志为“1”的状态下写“1”时

[为“0”的条件]

- 当写“0”时
- 当结束重新开始条件的发行或者检测到开始条件时
- 当 ICSR2.AL 标志变为“1”（仲裁失败）时
- 当给 ICCR1.IICRST 位写“1”并且进行 RIIC 复位或者内部复位时

注. 不能在发行停止条件的过程中将 RS 位置“1”。

必须注意：如果在主控模式以外的模式中给 RS 位写“1”（请求发行重新开始条件），就不能在该运行模式中发行重新开始条件而 RS 位保持“1”。如果在此状态下将运行模式转移到主控模式，就可能发行重新开始条件。

SP 位 (停止条件发行请求位)

此位在主控模式中请求发行停止条件。

如果 SP 位为“1”，就请求发行停止条件，在 BBSY 标志为“1”（总线忙）并且 MST 位为“1”（主控模式）时，发行停止条件。

有关发行停止条件的详细内容，请参照“29.10 开始条件、重新开始条件和停止条件的发行功能”。

[为“1”的条件]

- 在 ICCR2.BBSY 标志为“1”并且 ICCR2.MST 位为“1”的状态下写“1”时

[为“0”的条件]

- 当写“0”时
- 当结束停止条件的发行或者检测到停止条件时
- 当 ICSR2.AL 标志变为“1”（仲裁失败）时
- 当检测到开始条件和重新开始条件时
- 当给 ICCR1.IICRST 位写“1”并且进行 RIIC 复位或者内部复位时

注. 不能在 BBSY 标志为“0”（总线空闲）时写此位。

不能在发行重新开始条件的过程中将 SP 位置“1”。

TRS 位 (发送 / 接收模式位)

此位是表示发送模式或者接收模式的位。

当 TRS 位为“0”时，为接收模式；当 TRS 位为“1”时，为发送模式。通过和 MST 位的组合，表示 RIIC 的运行模式。

TRS 位通过开始条件的发行和检测以及 R/W# 位等变为“1”或者“0”，并且运行模式自动转移到发送模式或者接收模式。当 ICMR1.MTWP 位为“1”时，能写 TRS 位，但是一般不需要写。

[为“1”的条件]

- 根据开始条件的发行请求，正常地发行了开始条件（在 ST 位为“1”的状态下检测到开始条件）时
- 在 主控模式中，附加到从属地址的 R/W# 位为“0”时
- 在 从属模式中，接收的从属地址与 ICSER 寄存器的有效地址匹配，并且 R/W# 位接收到“1”时
- 在 ICMR1.MTWP 位为“1”的状态下写“1”时

[为“0”的条件]

- 当检测到停止条件时
- 当 ICSR2.AL 标志变为“1”（仲裁失败）时
- 在 主控模式中，附加到从属地址的 R/W# 位为“1”时
- 在 从属模式中，接收的从属地址与 ICSE 寄存器的有效地址匹配，并且 R/W# 位接收到“0”（包括全呼地址）时
- 在 从属模式中，检测到重新开始条件（在 ICCR2.BBSY 位“1”并且 ICCR2.MST 为“0”的状态下检测到开始条件）时
- 在 ICMR1.MTWP 位为“1”的状态下写“0”时
- 当给 ICCR1.IICRST 位写“1”并且进行 RIIC 复位或者内部复位时

MST 位（主控 / 从属模式位）

此位是表示主控模式或者从属模式的位。

当 MST 位为“0”时，为从属模式；当 MST 位为“1”时，为主控模式。通过和 TRS 位的组合，表示 RIIC 的运行模式。

通过开始条件的发行以及停止条件的发行和检测，MST 位变为“1”或者“0”，并且运行模式自动转移到主控模式或者从属模式。当 ICMR1.MTWP 位为“1”时，能写 MST 位，但是一般不需要写。

[为“1”的条件]

- 根据开始条件的发行请求，正常地发行了开始条件（在 ST 位为“1”的状态下检测到开始条件）时
- 在 ICMR1.MTWP 位为“1”的状态下写“1”时

[为“0”的条件]

- 当检测到停止条件时
- 当 ICSR2.AL 标志变为“1”（仲裁失败）时
- 在 ICMR1.MTWP 位为“1”的状态下写“0”时
- 当给 ICCR1.IICRST 位写“1”并且进行 RIIC 复位或者内部复位时

BBSY 标志（总线忙检测标志）

此标志表示 I²C 总线的占有（总线忙）状态或者释放状态（总线空闲）。

如果在 SCL 线为 High 电平的状态下 SDA 线从 High 电平变为 Low 电平，就认为发行了开始条件，此标志变为“1”。

如果在 SCL 线为 High 电平的状态下 SDA 线从 Low 电平变为 High 电平，就认为发行了停止条件，在没有检测到总线空闲时间（ICBRL 寄存器的设定时间）的开始条件时，此标志变为“0”。

[为“1”的条件]

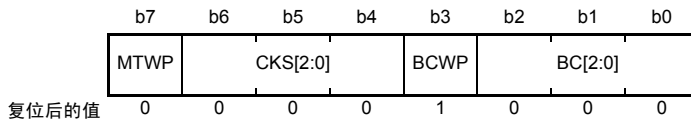
- 当检测到开始条件时

[为“0”的条件]

- 在检测到停止条件后没有检测到总线空闲时间（ICBRL 寄存器的设定时间）的开始条件时
- 在 ICCR1.IICE 位为“0”的状态下给 ICCR1.IICRST 位写“1”（RIIC 复位）时

29.2.3 I²C 总线模式寄存器 1 (ICMR1)

地址 RIIC0.ICMR1 0008 8302h



位	符号	位名	功能	R/W
b2-b0	BC[2:0]	位计数器	b2 b0 0 0 0: 9 位 0 0 1: 2 位 0 1 0: 3 位 0 1 1: 4 位 1 0 0: 5 位 1 0 1: 6 位 1 1 0: 7 位 1 1 1: 8 位	R/W (注 1)
b3	BCWP	BC 写保护位	0: 允许设定 BC[2:0] 的值 (读取值为“1”)	R/W (注 1)
b6-b4	CKS[2:0]	内部基准时钟选择位	b6 b4 0 0 0: PCLK/1 时钟 0 0 1: PCLK/2 时钟 0 1 0: PCLK/4 时钟 0 1 1: PCLK/8 时钟 1 0 0: PCLK/16 时钟 1 0 1: PCLK/32 时钟 1 1 0: PCLK/64 时钟 1 1 1: PCLK/128 时钟	R/W
b7	MTWP	MST/TRS 写保护位	0: 禁止写 ICCR2.MST 位和 ICCR2.TRS 位 1: 允许写 ICCR2.MST 位和 ICCR2.TRS 位	R/W

注 1. 必须在将 BCWP 位置“0”后, 使用 MOV 指令改写 BC[2:0] 位。

BC[2:0] 位 (位计数器)

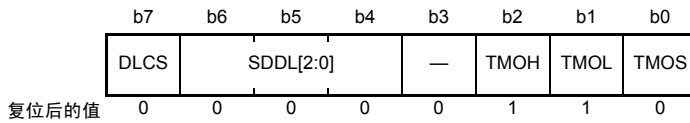
这些位是在 SCL 线的上升沿进行递减计数的计数器。如果读这些位, 就能得知剩下的传送位数。能读写这些位, 但是一般不需要存取。

写这些位时, 必须指定要传送的数据位数 +1 (数据附加 1 位应答位后传送), 并且在传送帧期间以及 SCL 线为 Low 电平的状态下进行。

在结束包含应答的数据传送或者检测到开始条件 (包括重新开始条件) 时, BC[2:0] 位自动返回“000b”。

29.2.4 I²C 总线模式寄存器 2 (ICMR2)

地址 RIIC0.ICMR2 0008 8303h



位	符号	位名	功能	R/W
b0	TMOS	超时检测时间选择位	0: 选择长模式 1: 选择短模式	R/W
b1	TMOL	超时 L 电平计数控制位	0: 在 SCL 线为 Low 电平时, 禁止计数。 1: 在 SCL 线为 Low 电平时, 允许计数。	R/W
b2	TMOH	超时 H 电平计数控制位	0: 在 SCL 线为 High 电平时, 禁止计数。 1: 在 SCL 线为 High 电平时, 允许计数。	R/W
b3	—	保留位	读写值都为“0”。	R/W
b6-b4	SDDL[2:0]	SDA 输出延迟计数器	<ul style="list-style-type: none"> • 当 ICMR2.DLCS 位为“0” (IICϕ) 时 <li style="padding-left: 20px;">b6 b4 0 0 0: 没有输出延迟 0 0 1: 1 个 IICϕ 周期 0 1 0: 2 个 IICϕ 周期 0 1 1: 3 个 IICϕ 周期 1 0 0: 4 个 IICϕ 周期 1 0 1: 5 个 IICϕ 周期 1 1 0: 6 个 IICϕ 周期 1 1 1: 7 个 IICϕ 周期 • 当 ICMR2.DLCS 位为“1” (IICϕ/2) 时 <li style="padding-left: 20px;">b6 b4 0 0 0: 没有输出延迟 0 0 1: 1 ~ 2 个 IICϕ 周期 0 1 0: 3 ~ 4 个 IICϕ 周期 0 1 1: 5 ~ 6 个 IICϕ 周期 1 0 0: 7 ~ 8 个 IICϕ 周期 1 0 1: 9 ~ 10 个 IICϕ 周期 1 1 0: 11 ~ 12 个 IICϕ 周期 1 1 1: 13 ~ 14 个 IICϕ 周期 	R/W
b7	DLCS	SDA 输出延迟时钟源选择位	0: 选择内部基准时钟 (IIC ϕ) 为 SDA 输出延迟计数器的时钟源 1: 选择内部基准时钟的 2 分频时钟 (IIC ϕ /2) 为 SDA 输出延迟计数器的时钟源	R/W

TMOS 位 (超时检测时间选择位)

此位是在超时检测功能有效时 (ICFER.TMOE 位 =1) 选择超时检测时间的位。如果将此位置“0”, 就为长模式; 如果置“1”, 就为短模式。在长模式中, 用于检测超时的内部计数器用作 16 位计数器; 在短模式中, 此内部计数器用作 14 位计数器。在 SCL 线为 TMOH 位或者 TMOL 位选择的状态时, 将内部基准时钟 (IIC ϕ) 作为计数源进行递增计数。

有关超时检测功能的详细内容, 请参照“29.11.1 超时检测功能”。

TMOL 位 (超时 L 电平计数控制位)

在超时检测功能有效时 (ICFER.TMOE 位 =1)，此位选择在 SCL 线为 Low 电平期间允许或者禁止超时检测功能的内部计数器的递增计数。

TMOH 位 (超时 H 电平计数控制位)

在超时检测功能有效时 (ICFER.TMOE 位 =1)，此位选择在 SCL 线为 High 电平期间允许或者禁止超时检测功能的内部计数器的递增计数。

SDDL[2:0] 位 (SDA 输出延迟计数器)

能通过 SDDL[2:0] 位的设定值使 SDA 延迟输出。SDA 输出延迟计数器通过由 DLCS 位选择的时钟源进行计数。另外，此功能的设定还适用于包括应答位发送在内的全部 SDA 输出。

有关此功能的详细内容，请参照“29.5 SDA 输出延迟功能”。

注 . 必须注意：SDA 输出延迟的设定必须符合 I²C 总线规格 (数据有效时间 / 应答有效时间 (注 1) 内) 或者 SMBus 规格 (数据保持时间: 至少为 300ns, SCL 时钟的 Low 电平宽度的数据准备时间: 250ns 的范围内)，否则就可能引起通信设备的通信故障，或者根据总线状态视为开始条件或者停止条件。

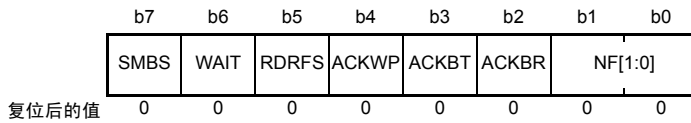
注 1. 数据有效时间 / 应答有效时间

3450ns (~ 100kbps: 标准模式 [Sm])

900ns (~ 400kbp: 快速模式 [fm])

29.2.5 I²C 总线模式寄存器 3 (ICMR3)

地址 RIIC0.ICMR3 0008 8304h



位	符号	位名	功能	R/W
b1-b0	NF[1:0]	噪声滤波器的段数选择位	b1 b0 0 0: 消除不超过 1 个 IIC ϕ 的噪声 (滤波器为 1 段) 0 1: 消除不超过 2 个 IIC ϕ 的噪声 (滤波器为 2 段) 1 0: 消除不超过 3 个 IIC ϕ 的噪声 (滤波器为 3 段) 1 1: 消除不超过 4 个 IIC ϕ 的噪声 (滤波器为 4 段)	R/W
b2	ACKBR	接收应答位	0: 应答位接收“0” (接收 ACK) 1: 应答位接收“1” (接收 NACK)	R
b3	ACKBT	发送应答位	0: 应答位发送“0” (发送 ACK) 1: 应答位发送“1” (发送 NACK)	R/W (注 1)
b4	ACKWP	ACKBT 写保护位	0: 禁止写 ACKBT 位 1: 允许写 ACKBT 位	W (注 1)
b5	RDRFS	RDRF 标志的置位时序选择位	0: 在 SCL 时钟的第 9 个时钟上升时, 此位为“1”。 (在第 8 个时钟的下降沿, SCL 线不保持 Low 电平。) 1: 在 SCL 时钟的第 8 个时钟上升时, 此位为“1”。 (在第 8 个时钟的下降沿, SCL 线保持 Low 电平。) 通过写 ACKBT 位来解除保持的 Low 电平。	R/W (注 2)
b6	WAIT	WAIT 位	0: 无 WAIT (在第 9 个时钟和第 1 个时钟之间不保持 Low 电平。) 1: 有 WAIT (在第 9 个时钟和第 1 个时钟之间保持 Low 电平。) 通过读 ICDRR 寄存器来解除保持的 Low 电平。	R/W (注 2)
b7	SMBS	SMBus/I ² C 总线选择位	0: 选择 I ² C 总线 1: 选择 SMBus	R/W

注 1. 即使在写 ACKBT 位的同时将 ACKWP 位置“1”, 也不能写 ACKBT 位。

注 2. WAIT 位和 RDRFS 位只在接收模式中有效, 在发送模式中无效。

ICMR3 寄存器控制 ACK 发送接收功能或者 RIIC 接收运行中的 RDRF 标志和 WAIT 位的运行。

注. 必须注意: 用噪声滤波器消除的噪声宽度必须小于 SCL 线的 High/Low 电平宽度。

如果大于等于 (SCL 时钟的宽度: High 电平宽度和 Low 电平宽度中较短的一方) - {1.5 个内部基准时钟同步 (IIC ϕ) + 模拟噪声滤波器: 120ns (参考值)} 的值, RIIC 的噪声滤波器功能就可能将 SCL 时钟视为噪声而无法正常运行。

ACKBR 位 (接收应答位)

在发送模式中, 此位保存从接收设备收到的应答位的内容。

[为“1”的条件]

- 在 ICCR2.TRS 位为“1”的状态下应答位接收到“1”时

[为“0”的条件]

- 在 ICCR2.TRS 位为“1”的状态下应答位接收到“0”时
- 在 ICCR1.ICE 位为“0”的状态下给 ICCR1.IICRST 位写“1” (RIIC 复位) 时

ACKBT 位 (发送应答位)

在接收模式中, 此位设定应答时要发送的位。

[为“1”的条件]

- 在 ACKWP 位为“1”的状态下写“1”时

[为“0”的条件]

- 在 ACKWP 位为“1”的状态下写“0”时
- 当检测到停止条件的发行 (在 ICCR2.SP 位为“1”的状态下检测到停止条件) 时
- 在 ICCR1.ICE 位为“0”的状态下给 ICCR1.IICRST 位写“1” (RIIC 复位) 时

注. 必须在 ACKWP 位为“1”的状态下写 ACKBT 位。如果在 ACKWP 位为“0”的状态下写 ACKBT 位, 写操作就无效。

ACKWP 位 (ACKBT 写保护位)

此位控制 ACKBT 位的写操作。

RDRFS 位 (RDRF 标志的置位时序选择位)

此位选择接收模式中的 RDRF 标志的置位时序以及在 SCL 时钟的第 8 个时钟的下降沿是否保持 SCL 线的 Low 电平。

当 RDRFS 位为“0”时, 在第 8 个时钟的下降沿 SCL 线不保持 Low 电平, 在第 9 个时钟的上升沿将 RDRF 标志置“1”。

当 RDRFS 位为“1”时, 在第 8 个时钟的上升沿将 RDRF 标志置“1”, 在第 8 个时钟的下降沿 SCL 线保持 Low 电平。通过写 ACKBT 位来解除此 SCL 线保持的 Low 电平。

因为在进行此设定时, 在接收数据后到发送应答位前 SCL 线自动保持 Low 电平, 所以能根据接收数据的内容发送 ACK (ACKBT 位为“0”) 或者 NACK (ACKBT 位为“1”)。

WAIT 位 (WAIT 位)

在接收模式中, 如果每接收 1 字节数据, 就在读完接收数据缓冲器 (ICDRR 寄存器) 前, WAIT 位控制在 SCL 时钟的第 9 个时钟和下次传送的第 1 个时钟之间是否保持 Low 电平。

当 WAIT 位为“0”时, 在 SCL 时钟的第 9 个时钟和下次传送的第 1 个时钟之间不保持 Low 电平, 继续接收数据。在 RDRFS 位和 WAIT 位都为“0”时, 也能通过双缓冲器进行连续接收。

当 WAIT 位为“1”时, 如果每接收 1 字节数据, 就从第 9 个时钟下降后到读 ICDRR 寄存器的值前, SCL 线保持 Low 电平。因此能按字节接收数据。

注. 必须先读 ICDRR, 然后将 WAIT 位置“0”。

SMBS 位 (SMBus/I²C 总线选择位)

如果将 SMBS 位置“1”, 就选择 SMBus 并且 IC SER.HOAE 位有效。

29.2.6 I²C 总线功能允许寄存器 (ICFER)

地址 RIIC0.ICFER 0008 8305h

b7	b6	b5	b4	b3	b2	b1	b0
—	SCLE	NFE	NACKE	SALE	NALE	MALE	TMOE
0	1	1	1	0	0	1	0

复位后的值

位	符号	位名	功能	R/W
b0	TMOE	超时检测功能有效位	0: 超时检测功能无效 1: 超时检测功能有效	R/W
b1	MALE	主控仲裁失败检测允许位	0: 禁止主控仲裁失败的检测 (将仲裁失败检测功能设定为无效, 在发生仲裁失败时不自动清除 ICCR2.MST 位和 ICCR2.TRS 位。) 1: 允许主控仲裁失败的检测 (将仲裁失败检测功能设定为有效, 在发生仲裁失败时自动清除 ICCR2.MST 位和 ICCR2.TRS 位。)	R/W
b2	NALE	NACK 发送仲裁失败检测允许位	0: 禁止 NACK 发送仲裁失败的检测 1: 允许 NACK 发送仲裁失败的检测	R/W
b3	SALE	从属仲裁失败检测允许位	0: 禁止从属仲裁失败的检测 1: 允许从属仲裁失败的检测	R/W
b4	NACKE	NACK 接收传送中止允许位	0: 在接收 NACK 时, 不中止传送 (禁止中止传送)。 1: 在接收 NACK 时, 中止传送 (允许中止传送)。	R/W
b5	NFE	数字噪声滤波器电路有效位	0: 不使用数字噪声滤波器电路 1: 使用数字噪声滤波器电路	R/W
b6	SCLE	SCL 同步电路有效位	0: SCL 同步电路无效 1: SCL 同步电路有效	R/W
b7	—	保留位	读写值都为“0”。	R/W

TMOE 位 (超时检测功能有效位)

此位选择超时检测功能的有效或者无效。

有关超时检测功能的详细内容, 请参照“29.11.1 超时检测功能”。

MALE 位 (主控仲裁失败检测允许位)

此位决定主控模式中仲裁失败检测功能的有效或者无效。一般必须将此位置“1”。

NALE 位 (NACK 发送仲裁失败检测允许位)

在接收模式中, 此位选择在发送 NACK 时检测到 ACK 的情况下 (总线上有相同地址的从属设备, 或者 2 个以上 (包括 2 个) 的主控设备同时选择了相同的从属设备并且各自接收的字节数不同等情况) 是否产生仲裁失败。

SALE 位 (从属仲裁失败检测允许位)

在从属发送模式中，此位选择在总线上检测到的值和正在发送的值不同的情况下（总线上有相同地址的从属设备，或者因噪声的影响而产生和发送数据不同的数据等情况）是否产生仲裁失败。

NACKE 位 (NACK 接收传送中止允许位)

在发送模式中，此位选择在从从属设备接收到 NACK 时是继续传送还是中止传送。一般必须将此位置“1”。

如果在 NACKE 位为“1”时接收到 NACK，就中止下一个传送。

当 NACKE 位为“0”时，与接收应答的内容无关，继续进行下一个传送。

有关 NACK 接收传送中止功能的详细内容，请参照“29.8.2 NACK 接收传送中止功能”。

SCLE 位 (SCL 同步电路允许位)

对于 SCL 输入时钟，此位选择是否与 SCL 时钟同步。一般必须将此位置“1”。

如果将 SCLE 位置“0” (SCL 同步电路无效)，就不与时钟同步。在此设定下，与 SCL 线的状态无关，RIIC 输出 ICBRH 寄存器和 ICBRL 寄存器所设传送速度的 SCL 时钟。因此必须注意：在 I²C 总线的总线负载远远大于规格值时或者多主控模式中的 SCL 时钟输出发生重叠时，有可能变为规格外的短时钟。在 SCL 同步电路无效的情况下，也影响开始条件、重新开始条件、停止条件的发行以及 SCL 时钟追加输出的连续输出。

只在确认是否输出了所设传送速度时才能将 SCLE 位置“0”。

29.2.7 I²C 总线状态允许寄存器 (ICSER)

地址 RIIC0.ICSER 0008 8306h

	b7	b6	b5	b4	b3	b2	b1	b0
	HOAE	—	DIDE	—	GCAE	SAR2E	SAR1E	SAR0E
复位后的值	0	0	0	0	1	0	0	1

位	符号	位名	功能	R/W
b0	SAR0E	从属地址寄存器 0 允许位	0: SARL0 和 SARU0 的设定值无效 1: SARL0 和 SARU0 的设定值有效	R/W
b1	SAR1E	从属地址寄存器 1 允许位	0: SARL1 和 SARU1 的设定值无效 1: SARL1 和 SARU1 的设定值有效	R/W
b2	SAR2E	从属地址寄存器 2 允许位	0: SARL2 和 SARU2 的设定值无效 1: SARL2 和 SARU2 的设定值有效	R/W
b3	GCAE	全呼地址允许位	0: 全呼地址的检测无效 1: 全呼地址的检测有效	R/W
b4	—	保留位	读写值都为“0”。	R/W
b5	DIDE	设备 ID 地址检测允许位	0: 设备 ID 地址的检测无效 1: 设备 ID 地址的检测有效	R/W
b6	—	保留位	读写值都为“0”。	R/W
b7	HOAE	主机地址允许位	0: 主机地址的检测无效 1: 主机地址的检测有效	R/W

SARyE 位 (从属地址寄存器 y 允许位) (y=0 ~ 2)

此位选择 SARLy 寄存器和 SARUy 寄存器设定的从属地址是否有效。

如果将 SARyE 位置“1”，SARLy 寄存器和 SARUy 寄存器的设定值就有效，和接收的从属地址进行比较。

如果将 SARyE 位置“0”，SARLy 寄存器和 SARUy 寄存器的设定值就无效，即使与接收的从属地址匹配，也忽视此设定值。

GCAE 位 (全呼地址允许位)

在接收到全呼地址 (0000 000b+0[W]: All“0”) 时，此位选择是否忽视此地址。

在 GCAE 位为“1”时，如果接收的从属地址与全呼地址匹配，RIIC 就将此从属地址视为全呼地址进行接收，与 SARLy 寄存器和 SARUy 寄存器 (y=0 ~ 2) 设定的从属地址无关。

在 GCAE 位为“0”时，即使接收的从属地址与全呼地址匹配，也忽视此地址。

DIDE 位 (设备 ID 地址检测允许位)

在检测到开始条件或者重新开始条件后的第 1 帧接收到设备 ID 地址 (1111 100b) 时，此位选择是否将此地址视为设备 ID 地址。

在 DIDE 位为“1”时，如果接收的第 1 帧与设备 ID 地址匹配，RIIC 就认为已接收到设备 ID 地址，当后续的 R/W# 位为“0”[W] 时，将第 2 帧以后的帧视为从属地址，继续接收。

在 DIDE 位为“0”时，即使接收的第 1 帧与设备 ID 地址匹配，也忽视此帧，而将第 1 帧视为一般的从属地址。

有关设备 ID 地址检测的详细内容，请参照“29.7.3 设备 ID 地址检测功能”。

HOAE 位 (主机地址允许位)

在 ICMR3.SMBS 位为“1”时接收到主机地址 (0001 000b) 的情况下, 此位选择是否忽视此地址。

在 ICMR3.SMBS 位为“1”并且 HOAE 位为“1”时, 如果接收的从属地址与主机地址匹配, RIIC 就将此从属地址视为主机地址进行接收, 与 SARLy 寄存器和 SARUy 寄存器 (y=0 ~ 2) 设定的从属地址无关。

在 ICMR3.SMBS 位或者 HOAE 位为“0”时, 即使接收的从属地址与主机地址匹配, 也忽视此地址。

29.2.8 I²C 总线中断允许寄存器 (ICIER)

地址 RIIC0.ICIER 0008 8307h

b7	b6	b5	b4	b3	b2	b1	b0
TIE	TEIE	RIE	NAKIE	SPIE	STIE	ALIE	TMOIE
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	TMOIE	超时中断允许位	0: 禁止超时中断 (TMOI) 1: 允许超时中断 (TMOI)	R/W
b1	ALIE	仲裁失败中断允许位	0: 禁止仲裁失败中断 (ALI) 1: 允许仲裁失败中断 (ALI)	R/W
b2	STIE	开始条件检测中断允许位	0: 禁止开始条件检测中断 (STI) 1: 允许开始条件检测中断 (STI)	R/W
b3	SPIE	停止条件检测中断允许位	0: 禁止停止条件检测中断 (SPI) 1: 允许停止条件检测中断 (SPI)	R/W
b4	NAKIE	NACK 接收中断允许位	0: 禁止 NACK 接收中断 (NAKI) 1: 允许 NACK 接收中断 (NAKI)	R/W
b5	RIE	接收数据满中断允许位	0: 禁止接收数据满中断 (ICRXI) 1: 允许接收数据满中断 (ICRXI)	R/W
b6	TEIE	发送结束中断允许位	0: 禁止发送结束中断 (ICTEI) 1: 允许发送结束中断 (ICTEI)	R/W
b7	TIE	发送数据空中断允许位	0: 禁止发送数据空中断 (ICTXI) 1: 允许发送数据空中断 (ICTXI)	R/W

TMOIE 位 (超时中断允许位)

当 ICSR2.TMOF 标志为“1”时，此位选择允许或者禁止超时中断 (TMOI)。能通过将 TMOF 标志或者 TMOIE 位置“0”来解除 TMOI 中断。

ALIE 位 (仲裁失败中断允许位)

当 ICSR2.AL 标志为“1”时，此位选择允许或者禁止仲裁失败中断 (ALI)。能通过将 AL 标志或者 ALIE 位置“0”来解除 ALI 中断。

STIE 位 (开始条件检测中断允许位)

当 ICSR2.START 标志为“1”时，此位选择允许或者禁止开始条件的检测中断 (STI)。能通过将 START 标志或者 STIE 位置“0”来解除 STI 中断。

SPIE 位 (停止条件检测中断允许位)

当 ICSR2.STOP 标志为“1”时，此位选择允许或者禁止停止条件的检测中断 (SPI)。能通过将 STOP 标志或者 SPIE 位置“0”来解除 SPI 中断。

NAKIE 位 (NACK 接收中断允许位)

当 ICSR2.NACKF 标志为“1”时，此位选择允许或者禁止 NACK 接收中断 (NAKI)。通过将 NACKF 标志或者 NAKIE 位置“0”来解除 NAKI 中断。

RIE 位 (接收数据满中断允许位)

当 ICSR2.RDRF 标志为“1”时，此位选择允许或者禁止接收数据满中断 (ICRXI)。

TEIE 位 (发送结束中断允许位)

当 ICSR2.TEND 标志为“1”时，此位选择允许或者禁止发送结束中断 (ICTEI)。能通过将 TEND 标志或者 TEIE 位置“0”来解除 ICTEI 中断。

TIE 位 (发送数据空中断允许位)

当 ICSR2.TDRE 标志为“1”时，此位选择允许或者禁止发送数据空中断 (ICTXI)。

29.2.9 I²C 总线状态寄存器 1 (ICSR1)

地址 RIIC0.ICSR1 0008 8308h

b7	b6	b5	b4	b3	b2	b1	b0
HOA	—	DID	—	GCA	AAS2	AAS1	AAS0
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	AAS0	从属地址 0 的检测标志	0: 未检测到从属地址 0 1: 检测到从属地址 0 • 在 SARU0.FS 位为“0” (选择 7 位地址格式) 的情况下, 接收的从属地址与 SARL0.SVA[6:0] 匹配时 • 在 SARU0.FS 位为“1” (选择 10 位地址格式) 的情况下, 接收的从属地址与 1111 0b+SARU0.SVA[1:0] 匹配, 并且后续的地址与 SARL0 寄存器匹配时 (在 SARL0 匹配判断帧的 SCL 时钟的第 9 个时钟上升时, 此标志变为“1”。)	R(W) (注 1)
b1	AAS1	从属地址 1 的检测标志	0: 未检测到从属地址 1 1: 检测到从属地址 1 • 在 SARU1.FS 位为“0” (选择 7 位地址格式) 的情况下, 接收的从属地址与 SARL1.SVA[6:0] 匹配时 • 在 SARU1.FS 位为“1” (选择 10 位地址格式) 的情况下, 接收的从属地址与 1111 0b+SARU1.SVA[1:0] 匹配, 并且后续的地址与 SARL1 寄存器匹配时 (在 SARL1 匹配判断帧的 SCL 时钟的第 9 个时钟上升时, 此标志变为“1”。)	R(W) (注 1)
b2	AAS2	从属地址 2 的检测标志	0: 未检测到从属地址 2 1: 检测到从属地址 2 • 在 SARU2.FS 位为“0” (选择 7 位地址格式) 的情况下, 接收的从属地址与 SARL2.SVA[6:0] 匹配时 • 在 SARU2.FS 位为“1” (选择 10 位地址格式) 的情况下, 接收的从属地址与 1111 0b+SARU2.SVA[1:0] 匹配, 并且后续的地址与 SARL2 寄存器匹配时 (在 SARL2 匹配判断帧的 SCL 时钟的第 9 个时钟上升时, 此标志变为“1”。)	R(W) (注 1)
b3	GCA	全呼地址的检测标志	0: 未检测到全呼地址 1: 检测到全呼地址 • 当接收的从属地址与全呼地址 (All“0”) 匹配时	R(W) (注 1)
b4	—	保留位	读写值都为“0”。	R/W
b5	DID	设备 ID 地址的检测标志	0: 未检测到设备 ID 地址 1: 检测到设备 ID 地址 • 当紧接在开始条件后的第 1 帧与设备 ID 地址 (1111 100b) +0[W] 匹配时	R(W) (注 1)
b6	—	保留位	读写值都为“0”。	R/W
b7	HOA	主机地址的检测标志	0: 未检测到主机地址 1: 检测到主机地址 • 当接收的从属地址与主机地址 (0001 000b) 匹配时	R(W) (注 1)

注 1. 只能写“0”。

AASy 标志 (从属地址 y 的检测标志) (y=0 ~ 2)

[为“1”的条件]

【选择 7 位地址格式时: SARUy.FS 位 =0】

- 在 IC SER.SARyE 位为“1” (从属地址 y 的检测有效) 的状态下接收的从属地址与 SARLy.SVA[6:0] 匹配时, 该帧的 SCL 时钟的第 9 个时钟上升。

【选择 10 位地址格式时: SARUy.FS 位 =1】

- 在 IC SER.SARyE 位为“1” (从属地址 y 的检测有效) 的状态下接收的从属地址与 1111 0b+SARUy.SVA[1:0] 匹配, 并且后续的地址与 SARLy 寄存器匹配时, 该帧的 SCL 时钟的第 9 个时钟上升。

[为“0”的条件]

- 在读“1”后写“0”时
- 当检测到停止条件时
- 当给 ICCR1.IICRST 位写“1”并且进行 RIIC 复位或者内部复位时

【选择 7 位地址格式时: SARUy.FS 位 =0】

- 在 IC SER.SARyE 位为“1” (从属地址 y 的检测有效) 的状态下接收的从属地址与 SARLy.SVA[6:0] 不匹配时, 该帧的 SCL 时钟的第 9 个时钟上升。

【选择 10 位地址格式时: SARUy.FS 位 =1】

- 在 IC SER.SARyE 位为“1” (从属地址 y 的检测有效) 的状态下接收的从属地址与 1111 0b+SARUy.SVA[1:0] 不匹配时, 该帧的 SCL 时钟的第 9 个时钟上升。
- 在 IC SER.SARyE 位为“1” (从属地址 y 的检测有效) 的状态下接收的从属地址与 1111 0b+SARUy.SVA[1:0] 匹配, 而后续的地址与 SARLy 寄存器不匹配时, 该帧的 SCL 时钟的第 9 个时钟上升。

GCA 标志 (全呼地址的检测标志)

[为“1”的条件]

- 在 IC SER.GCAE 位为“1” (全呼地址的检测有效) 的状态下接收的从属地址与全呼地址 (0000 000b +0[W]) 匹配时, 该帧的 SCL 时钟的第 9 个时钟上升。

[为“0”的条件]

- 在读“1”后写“0”时
- 当检测到停止条件时
- 在 IC SER.GCAE 位为“1” (全呼地址的检测有效) 的状态下接收的从属地址与全呼地址 (0000 000b +0[W]) 不匹配时, 该帧的 SCL 时钟的第 9 个时钟上升。
- 当给 ICCR1.IICRST 位写“1”并且进行 RIIC 复位或者内部复位时

DID 标志 (设备 ID 地址的检测标志)

[为“1”的条件]

- 在 IC SER.DIDE 位为“1” (设备 ID 地址的检测有效) 的状态下检测到开始条件或者重新开始条件后的第 1 帧与设备 ID 地址 (1111 100b) +0[W] 匹配时, 该帧的 SCL 时钟的第 9 个时钟上升。

[为“0”的条件]

- 在读“1”后写“0”时
- 当检测到停止条件时
- 在 IC SER.DIDE 位为“1” (设备 ID 地址的检测有效) 的状态下检测到开始条件或者重新开始条件检测后的第 1 帧与设备 ID 地址 (1111 100b) 不匹配时, 该帧的 SCL 时钟的第 9 个时钟上升。
- 在 IC SER.DIDE 位为“1” (设备 ID 地址的检测有效) 的状态下检测到开始条件或者重新开始条件检测后的第 1 帧与设备的 ID 地址 (1111 100b) +0[W] 匹配, 而后续的第 2 帧与从属地址 0~2 全部不匹配时, 该帧的 SCL 时钟的第 9 个时钟上升。
- 当给 ICCR1.IICRST 位写“1”并且进行 RIIC 复位或者内部复位时

HOA 标志 (主机地址的检测标志)

[为“1”的条件]

- 在 IC SER.HOAE 位为“1” (主机地址的检测有效) 的状态下接收的从属地址与主机地址 (0001 000b) 匹配时, 该帧的 SCL 时钟的第 9 个时钟上升。

[为“0”的条件]

- 在读“1”后写“0”时
- 当检测到停止条件时
- 当给 ICMR3.SMBS 位或者 IC SER.HOAE 位写“0”时
- 在 IC SER.HOAE 位为“1” (主机地址的检测有效) 的状态下接收的从属地址与主机地址 (0001 000b) 不匹配时, 该帧的 SCL 时钟的第 9 个时钟上升。
- 当给 ICCR1.IICRST 位写“1”并且进行 RIIC 复位或者内部复位时

29.2.10 I²C 总线状态寄存器 2 (ICSR2)

地址 RIIC0.ICSR2 0008 8309h

b7	b6	b5	b4	b3	b2	b1	b0
TDRE	TEND	RDRF	NACKF	STOP	START	AL	TMOF
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	TMOF	超时检测标志	0: 未检测到超时 1: 检测到超时	R(W) (注1)
b1	AL	仲裁失败标志	0: 未发生仲裁失败 1: 发生仲裁失败	R(W) (注1)
b2	START	开始条件检测标志	0: 未检测到开始条件 1: 检测到开始条件	R(W) (注1)
b3	STOP	停止条件检测标志	0: 未检测到停止条件 1: 检测到停止条件	R(W) (注1)
b4	NACKF	NACK 检测标志	0: 未检测到 NACK 1: 检测到 NACK	R(W) (注1)
b5	RDRF	接收数据满标志	0: ICDRR 寄存器无接收数据 1: ICDRR 寄存器有接收数据	R(W) (注1)
b6	TEND	发送结束标志	0: 正在发送数据 1: 数据发送结束	R(W) (注1)
b7	TDRE	发送数据空标志	0: ICDRT 寄存器有发送数据 1: ICDRT 寄存器无发送数据	R

注 1. 只能写“0”。

TMOF 标志 (超时检测标志)

如果 SCL 线的状态在一定的期间内不发生变化, 就视为超时, 此标志变为“1”。

[为“1”的条件]

- 当 ICFER.TMOE 位为“1” (超时检测功能有效) 并且在指定为主控模式或者从属模式的状态下 ICMR2.TMOH 位、ICMR2.TMOL 位、ICMR2.TMOS 位所选条件的期间内, SCL 线的状态没有变化时

[为“0”的条件]

- 在读“1”后写“0”时
- 当给 ICCR1.IICRST 位写“1”并且进行 RIIC 复位或者内部复位时

AL 标志 (仲裁失败标志)

在发行开始条件或者发送地址和数据时, 此标志表示因总线竞争等而失去了总线占有权 (仲裁失败)。RIIC 在发送过程中监视 SDA 线的电平, 如果输出数据和 SDA 线的电平不同, 就将 AL 标志置“1”, 表示总线被其他设备占有。

另外, 通过设定, RIIC 也能在可控模式中检测发送 NACK 时的仲裁失败, 在从属模式中检测发送数据时的仲裁失败。

[为“1”的条件]

【主控仲裁失败的检测有效时：ICFER.MALE 位为“1”】

- 在发送模式的数据发送（包括从属地址的发送）过程中，在ACK期间以外的SCL时钟的上升沿，RIIC自身发送的SDA信号和SDA线上的信号状态不同（内部SDA输出为High电平输出（=SDA引脚为高阻抗）而检测到SDA线为Low电平）时
- 在ICCR2.ST位为“1”（请求发行开始条件）的状态下检测到开始条件时，RIIC自身发送的SDA信号和SDA线上的信号状态不同时
- 在ICCR2.BBSY标志为“1”的状态下将ICCR2.ST位置“1”（请求发行开始条件）时

【NACK 仲裁失败的检测有效时：ICFER.NALE 位为“1”】

- 在接收模式的NACK发送过程中，在ACK期间的SCL时钟的上升沿，RIIC自身发送的SDA信号和SDA线上的信号状态不同时

【从属仲裁失败的检测有效时：ICFER.SALE 位为“1”】

- 在从属发送模式的数据发送过程中，在ACK期间以外的SCL时钟上升沿，RIIC自身发送的SDA信号和SDA线上的信号状态不同时

[为“0”的条件]

- 在读“1”后写“0”时
- 当给ICCR1.IICRST位写“1”并且进行RIIC复位或者内部复位时

表 29.4 仲裁失败发生源和各仲裁失败允许功能的关系

ICFER			ICSR2	错误内容	仲裁失败发生源
MALE	NALE	SALE	AL		
1	x	x	1	开始条件发行错误	在 ICCR2.ST 位为“1”的状态下检测到开始条件时，RIIC 自身发送的 SDA 信号和 SDA 线上的信号状态不同。
			1	发送数据不同	在 ICCR2.BBSY 位为“1”的状态下将 ICCR2.ST 位置“1”。
x	1	x	1	发送的 NACK 不同	在主控接收模式或者从属接收模式中，在发送 NACK 时检测到 ACK。
x	x	1	1	发送数据不同	在从属发送模式中，发送数据和总线状态不同。

x: Don't care

START 标志（开始条件检测标志）

[为“1”的条件]

- 当检测到开始条件（包括重新开始条件）时

[为“0”的条件]

- 在读“1”后写“0”时
- 当检测到停止条件时
- 当给ICCR1.IICRST位写“1”并且进行RIIC复位或者内部复位时

STOP 标志（停止条件检测标志）

[为“1”的条件]

- 当检测到停止条件时

[为“0”的条件]

- 在读“1”后写“0”时
- 当给ICCR1.IICRST位写“1”并且进行RIIC复位或者内部复位时

NACKF 标志 (NACK 检测标志)

[为“1”的条件]

- 在 ICFER.NACKF 位为“1” (允许中止传送) 的状态下, 发送模式中没有来自接收设备的应答 (接收到 NACK) 时

[为“0”的条件]

- 在读“1”后写“0”时
- 当给 ICCR1.IICRST 位写“1”并且进行 RIIC 复位或者内部复位时

注. 如果 NACKF 标志变为“1”, RIIC 就中止通信。即使在 NACKF 标志为“1”的状态下写 ICDRT 寄存器 (在发送模式中) 或者读 ICDRR 寄存器 (在接收模式中), 也不进行发送和接收。要重新开始通信时, 必须将 NACKF 标志置“0”。

RDRF 标志 (接收数据满标志)

[为“1”的条件]

- 在将接收数据从 ICDRS 寄存器传送到 ICDRR 寄存器时, 通过设定 ICMR3.RDRFS 位, 在 SCL 时钟的第 8 个时钟或者第 9 个时钟的上升沿 RDRF 标志变为“1”。
- 在检测到开始条件 (包括重新开始条件) 后接收的从属地址匹配并且 ICCR2.TRS 位为“0”时

[为“0”的条件]

- 在读“1”后写“0”时
- 当读 ICDRR 寄存器时
- 当给 ICCR1.IICRST 位写“1”并且进行 RIIC 复位或者内部复位时

TEND 标志 (发送结束标志)

[为“1”的条件]

- 在 TDRE 标志为“1”的状态下 SCL 时钟的第 9 个时钟上升。

[为“0”的条件]

- 在读“1”后写“0”时
- 当给 ICDRT 寄存器写数据时
- 当检测到停止条件时
- 当给 ICCR1.IICRST 位写“1”并且进行 RIIC 复位或者内部复位时

TDRE 标志 (发送数据空标志)

[为“1”的条件]

- 当将数据从 ICDRT 寄存器传送到 ICDRS 寄存器并且 ICDRT 寄存器为空时
- 当 ICCR2.TRS 位变为“1”时
 - a. 在检测到开始条件 (包括重新开始条件) 后 ICCR2.MST 位为“1”时
 - b. 当从接收模式变为发送模式时
 - c. 在 ICMR1.MTWP 位为“1”的状态下写“1”时
- 当接收的从属地址匹配并且 TRS 位为“1”时

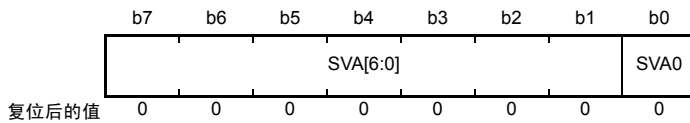
[为“0”的条件]

- 当给 ICDRT 寄存器写数据时
- 当 ICCR2.TRS 位变为“0”时
 - a. 检测到停止条件时
 - b. 从发送模式变为接收模式时
 - c. 在 ICMR1.MTWP 位为“1”的状态下写“0”时
- 当给 ICCR1.IICRST 位写“1”并且进行 RIIC 复位或者内部复位时

注. 如果在 ICFER.NACKF 位为“1”的状态下 NACKF 标志变为“1”, RIIC 就中止通信。此时, 如果 TDRE 标志为“0” (已写下一个发送数据的状态), 就在第 9 个时钟的上升沿将数据传送到 ICDRS 寄存器并且 ICDRT 寄存器变为空状态, 但是 TDRE 标志不变为“1”。

29.2.11 从属地址寄存器 Ly (SARLy) (y=0 ~ 2)

地址 RIIC0.SARL0 0008 830Ah、RIIC0.SARL1 0008 830Ch、RIIC0.SARL2 0008 830Eh



位	符号	位名	功能	R/W
b0	SVA0	10 位地址的最低位	必须设定从属地址。 • 当 SARUy.FS 位为“0”（选择 7 位地址格式）时，SVA0 位无效。 • 当 SARUy.FS 位为“1”（选择 10 位地址格式）时，SVA0 位有效。SVA0 位和 SVA[6:0] 位合并为 10 位从属地址的低 8 位地址。	R/W
b7-b1	SVA[6:0]	7 位地址 /10 位地址的低位	必须设定从属地址。 • 当 SARUy.FS 位为“0”（选择 7 位地址格式）时，SVA[6:0] 位为 7 位从属地址。 • 当 SARUy.FS 位“1”（选择 10 位地址格式）时，SVA[6:0] 位和 SVA0 位合并为 10 位从属地址的低 8 位地址。	R/W

SVA0 位（10 位地址的最低位）

在选择 10 位地址格式时（SARUy.FS 位 =1），用作 10 位地址的最低位，此位和 SVA[6:0] 位一起设定 10 位地址的低 8 位。

当 ICSEr.SARyE 位为“1”（SARLy 寄存器和 SARUy 寄存器有效）并且 SARUy.FS 位为“1”时，设定值有效；当 SARUy.FS 位或者 SARyE 位为“0”时，忽视设定值。

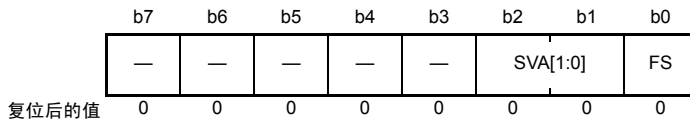
SVA[6:0] 位（7 位地址 /10 位地址的低位）

在选择 7 位地址格式时（SARUy.FS 位 =0），用作 7 位地址；在选择 10 位地址格式时（SARUy.FS 位 =1），此位和 SVA0 位一起用作 10 位地址的低 8 位。

当 ICSEr.SARyE 位为“0”时，忽视设定值。

29.2.12 从属地址寄存器 Uy (SARUy) (y=0 ~ 2)

地址 RIIC0.SARU0 0008 830Bh、RIIC0.SARU1 0008 830Dh、RIIC0.SARU2 0008 830Fh



位	符号	位名	功能	R/W
b0	FS	7 位 /10 位地址格式选择位	0: 选择 7 位地址格式 1: 选择 10 位地址格式	R/W
b2-b1	SVA[1:0]	10 位地址的高位	必须设定从属地址。 • 当 SARUy.FS 位为“0”（选择 7 位地址格式）时，SVA[1:0] 位无效。 • 当 SARUy.FS 位为“1”（选择 10 位地址格式）时，SVA[1:0] 位有效，作为 10 位从属地址的高 2 位地址。	R/W
b7-b3	—	保留位	读写值都为“0”。	R/W

FS 位（7 位 /10 位地址格式选择位）

此位选择从属地址 y (SARLy 寄存器、SARUy 寄存器) 为 7 位地址或者 10 位地址。

当 ICSEr.SARyE 位为“1” (SARLy 寄存器和 SARUy 寄存器有效) 并且 SARUy.FS 位为“0”时，选择从属地址 y 为 7 位地址格式，SARLy.SVA[6:0] 位的设定值有效，忽视 SVA[1:0] 位和 SARLy.SVA0 位的设定值。

当 ICSEr.SARyE 位为“1” (SARLy 寄存器和 SARUy 寄存器有效) 并且 SARUy.FS 位为“1”时，选择从属地址 y 为 10 位地址格式，SVA[1:0] 位和 SARLy 寄存器的设定值有效。

当 ICSEr.SARyE 位为“0” (SARLy 寄存器和 SARUy 寄存器无效) 时，SARUy.FS 位的设定值无效。

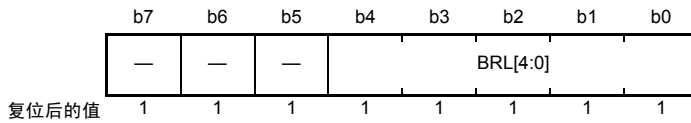
SVA[1:0] 位（10 位地址的高位）

在选择 10 位地址格式时 (FS 位 =1)，用作 10 位地址的高 2 位地址。

当 ICSEr.SARyE 位为“1” (SARLy 寄存器和 SARUy 寄存器有效) 并且 SARUy.FS 位为“1”时，设定值有效；当 SARUy.FS 位或者 SARyE 位为“0”时，忽视设定值。

29.2.13 I²C 总线位速率低电平寄存器 (ICBRL)

地址 RIIC0.ICBRL 0008 8310h



位	符号	位名	功能	R/W
b4-b0	BRL[4:0]	位速率 Low 电平宽度设定位	设定 SCL 时钟的 Low 电平宽度的值。	R/W
b7-b5	—	保留位	读写值都为“1”。	R/W

ICBRL 寄存器是设定 SCL 时钟的 Low 电平宽度的 5 位寄存器。

在 SCL 自动保持 Low 电平时 (参照“29.8 SCL 的 Low 电平自动保持功能”), ICBRL 寄存器用作数据准备时间的确保寄存器。因此, 在 RIIC 总是用于从属模式的情况下, 设定值不能小于数据准备时间 (注 1) 的值。

通过 ICMR1.CKS[2:0] 位选择的内部基准时钟源 (IIC ϕ), ICBRL 寄存器对 Low 电平宽度进行计数。

在允许使用数字噪声滤波器电路 (ICFER.NFE 位 =1) 时, 必须给 ICBRL 寄存器设定大于等于“噪声滤波器的段数 +1”的值。有关噪声滤波器的段数, 请参照 ICMR3.NF[1:0] 位。

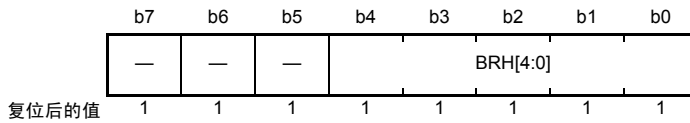
注 1. 数据准备时间 (t_{SU:DAT})

250ns (~100kbps: 标准模式 [Sm])

100ns (~400kbps: 快速模式 [fm])

29.2.14 I²C 总线位速率高电平寄存器 (ICBRH)

地址 RIIC0.ICBRH 0008 8311h



位	符号	位名	功能	R/W
b4-b0	BRH[4:0]	位速率 High 电平宽度设定位	设定 SCL 时钟的 High 电平宽度的值。	R/W
b7-b5	—	保留位	读写值都为“1”。	R/W

ICBRH 寄存器是用于设定 SCL 时钟的 High 电平宽度的 5 位寄存器，在主控模式中有效。在 RIIC 总是用于从属模式的情况下，不需要设定 High 电平宽度。

通过 ICMR1.CKS[2:0] 位选择的内部基准时钟源 (IIC ϕ)，ICBRH 寄存器对 High 电平宽度进行计数。

在允许使用数字噪声滤波器电路 (ICFER.NFE 位 =1) 时，必须给 ICBRH 寄存器设定大于等于“噪声滤波器的段数 +1”的值。有关噪声滤波器的段数，请参照 ICMR3.NF[1:0] 位。

用以下表达式计算 I²C 传送速度和 SCL 时钟的占空比。

传送速度 = $1 / \{ [(ICBRH+1) + (ICBRL+1)] / IIC\phi \text{ (注1)} + \text{SCL 线的上升时间 [tr]} + \text{SCL 的下降时间 [tf]} \}$

占空比 = $\{ \text{SCL 线的上升时间 [tr]} \text{ (注2)} + (ICBRH+1) / IIC\phi \} / \{ \text{SCL 线的下降时间 [tf]} \text{ (注2)} + (ICBRL+1) / IIC\phi \}$

注 1. $IIC\phi = PCLK \times 10^6 \times \text{分频比}$

注 2. SCL 线的上升时间 [tr] 和下降时间 [tf] 取决于总线的总电容量 [Cb] 和上拉电阻 [Rp]，详细内容请参照 NXP 公司的 I²C 总线规格书。

ICBRH 寄存器和 ICBRL 寄存器值的设定例子如表 29.5 所示。

表 29.5 对应传送速度的 ICBRH 寄存器和 ICBRL 寄存器的设定例子

传送速度 (kbps)	工作频率 PCLK (MHz)								
	8			10			12.5		
	CKS[2:0]	ICBRH	ICBRL	CKS[2:0]	ICBRH	ICBRL	CKS[2:0]	ICBRH	ICBRL
10	100b	22 (F6h)	25 (F9h)	101b	13 (EDh)	15 (EFh)	101b	16 (F0h)	20 (F4h)
50	010b	16 (F0h)	19 (F3h)	010b	21 (F5h)	24 (F8h)	011b	12 (ECh)	15 (EFh)
100	001b	15 (EFh)	18 (F2h)	001b	19 (F3h)	23 (F7h)	001b	24 (F8h)	29 (FDh)
400	000b	4 (E4h)	10 (EAh)	000b	5 (E5h)	12 (ECh)	000b	7 (E7h)	16 (F0h)

传送速度 (kbps)	工作频率 PCLK (MHz)								
	16			20			25		
	CKS[2:0]	ICBRH	ICBRL	CKS[2:0]	ICBRH	ICBRL	CKS[2:0]	ICBRH	ICBRL
10	101b	22 (F6h)	25 (F9h)	110b	13 (EDh)	15 (EFh)	110b	16 (F0h)	20 (F4h)
50	011b	16 (F0h)	19 (F3h)	011b	21 (F5h)	24 (F8h)	100b	12 (ECh)	15 (EFh)
100	010b	15 (EFh)	18 (F2h)	010b	19 (F3h)	23 (F7h)	010b	24 (F8h)	29 (FDh)
400	000b	9 (E9h)	20 (F4h)	000b	11 (EBh)	25 (F9h)	001b	7 (E7h)	16 (F0h)

传送速度 (kbps)	工作频率 PCLK (MHz)		
	30		
	CKS[2:0]	ICBRH	ICBRL
10	110b	20 (F4h)	24 (F8h)
50	100b	15 (EFh)	18 (F2h)
100	010b	2 (E2h)	3 (E3h)
400	001b	8 (E8h)	19 (F3h)

注. 计算的设定例子是假设:

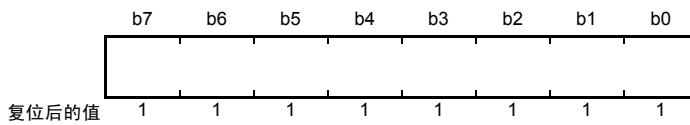
SCL 线的上升时间 (tr) 不超过 ~100kbps, [Sm] 为 1000ns 或者 ~400kbps, [Fm] 为 300ns。

SCL 线的下降时间 (tf) 不超过 ~400kbps, [Sm/Fm] 为 300ns。

有关 SCL 线的上升时间 (tr) 和下降时间 (tf) 的规格值, 请参照 NXP 公司的 I²C 总线规格书。

29.2.15 I²C 总线发送数据寄存器 (ICDRT)

地址 RIIC0.ICDRT 0008 8312h



如果检测到 I²C 总线移位寄存器 (ICDRS) 为空, 就将写在 ICDRT 寄存器的发送数据传送到 ICDRS 寄存器, 在发送模式中开始数据发送。

ICDRT 寄存器和 ICDRS 寄存器为双缓冲结构, 如果在 ICDRS 寄存器的数据发送过程中将下一个要发送的数据写到 ICDRT 寄存器, 就能进行连续的发送。

能随时读写 ICDRT 寄存器。只能在发生发送数据空中断 (ICTXI) 请求时给 ICDRT 寄存器写 1 次发送数据。

29.2.16 I²C 总线接收数据寄存器 (ICDRR)

地址 RIIC0.ICDRR 0008 8313h



如果 1 字节数据接收结束, 就能将接收数据从 I²C 总线移位寄存器 (ICDRS) 传送到 ICDRR 寄存器, 进入能接收下一个数据的状态。

ICDRS 寄存器和 ICDRR 寄存器为双缓冲结构, 如果在 ICDRS 寄存器的数据接收过程中从 ICDRR 寄存器读接收数据, 就能进行连续的接收。

不能写 ICDRR 寄存器。只能在发生接收数据满中断 (ICRXI) 请求时读 1 次 ICDRR 寄存器。

如果不从 ICDRR 寄存器读接收数据 (ICSR2.RDRF 标志为“1”的状态) 而立即接收下一个数据, RIIC 就在 RDRF 标志变为“1”时的前一个 SCL 时钟自动保持 Low 电平。

29.2.17 I²C 总线移位寄存器 (ICDRS)



ICDRS 寄存器是用于发送和接收数据的移位寄存器。

在发送时, 将发送数据从 ICDRT 寄存器传送到 ICDRS 寄存器, 从 SDA 引脚发送数据。在接收时, 一旦 1 字节数据接收结束, 就将数据从 ICDRS 寄存器传送到 ICDRR 寄存器。

不能直接存取 ICDRS 寄存器。

29.3 运行说明

29.3.1 通信数据的格式

I²C 总线格式由 8 位数据和 1 个应答位构成。接在开始条件或者重新开始条件后面的帧是地址帧，用于指定主控设备通信对象的从属设备。在指定新的从属设备或者发行停止条件前，指定的从属设备有效。

I²C 总线的格式及其总线时序分别如图 29.3 和图 29.4 所示。

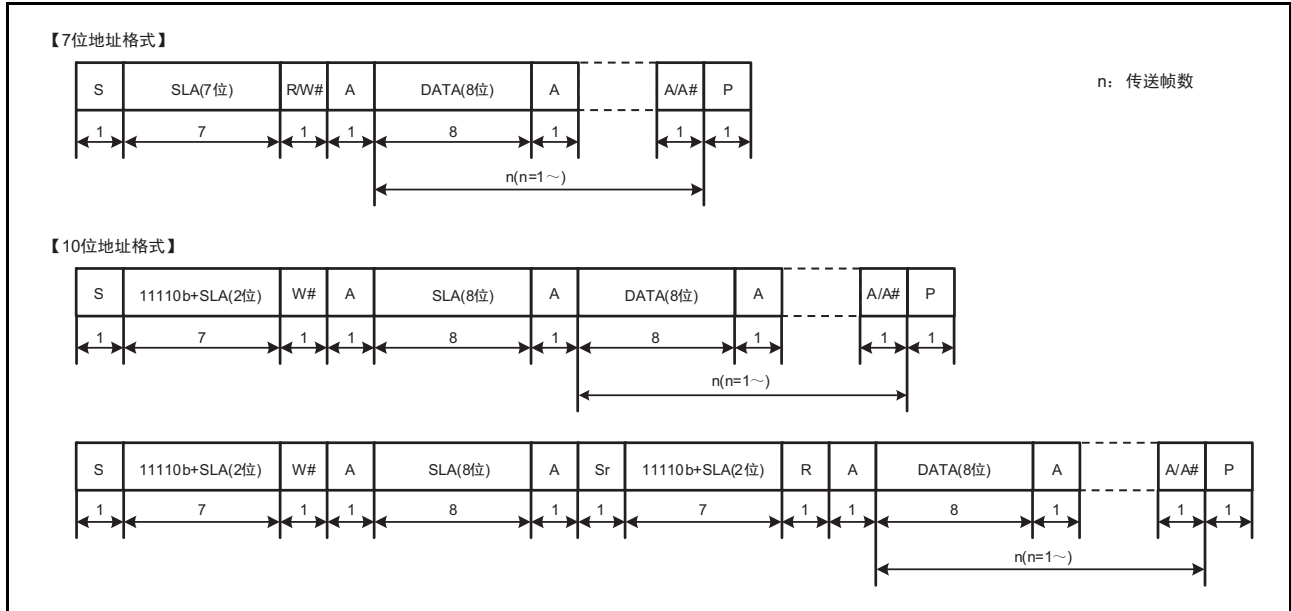


图 29.3 I²C 总线格式

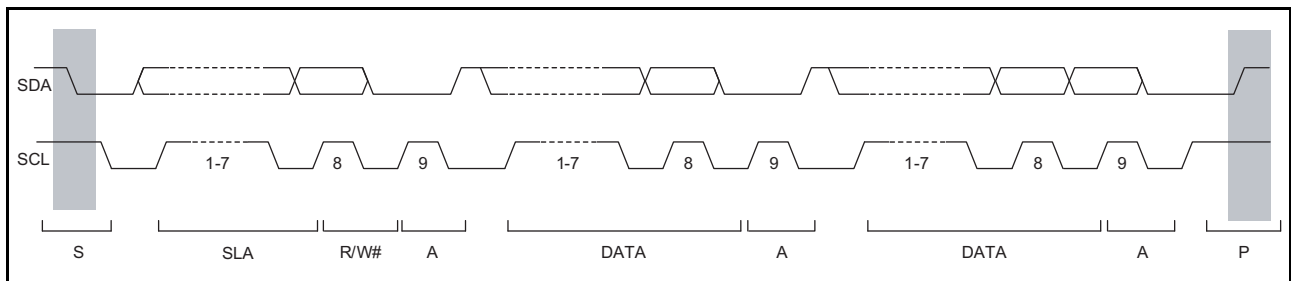


图 29.4 I²C 总线时序 (SLA 为 7 位)

- S : 表示开始条件。主控设备在 SCL 线为 High 电平的状态下将 SDA 线从 High 电平变为 Low 电平。
- SLA : 表示从属地址。主控设备选择从属设备。
- R/W# : 表示发送和接收的方向。当 R/W# 为 “1” 时，将数据从从属设备发送到主控设备；当 R/W# 为 “0” 时，将数据从主控设备发送到从属设备。
- A : 表示应答。接收设备将 SDA 线置为 Low 电平（在发送模式中，从属设备返回应答；在接收模式中，主控设备返回应答）。
- Sr : 表示重新开始条件。主控设备在 SCL 线为 High 电平的状态下经过准备时间后将 SDA 线从 High 电平变为 Low 电平。
- DATA : 表示发送和接收的数据。
- P : 表示停止条件。主控设备在 SCL 线为 High 电平的状态下将 SDA 线从 Low 电平变为 High 电平。

29.3.2 初始设定

在开始发送或者接收数据时，必须按照图 29.5 所示的步骤对 RIIC 进行初始化。

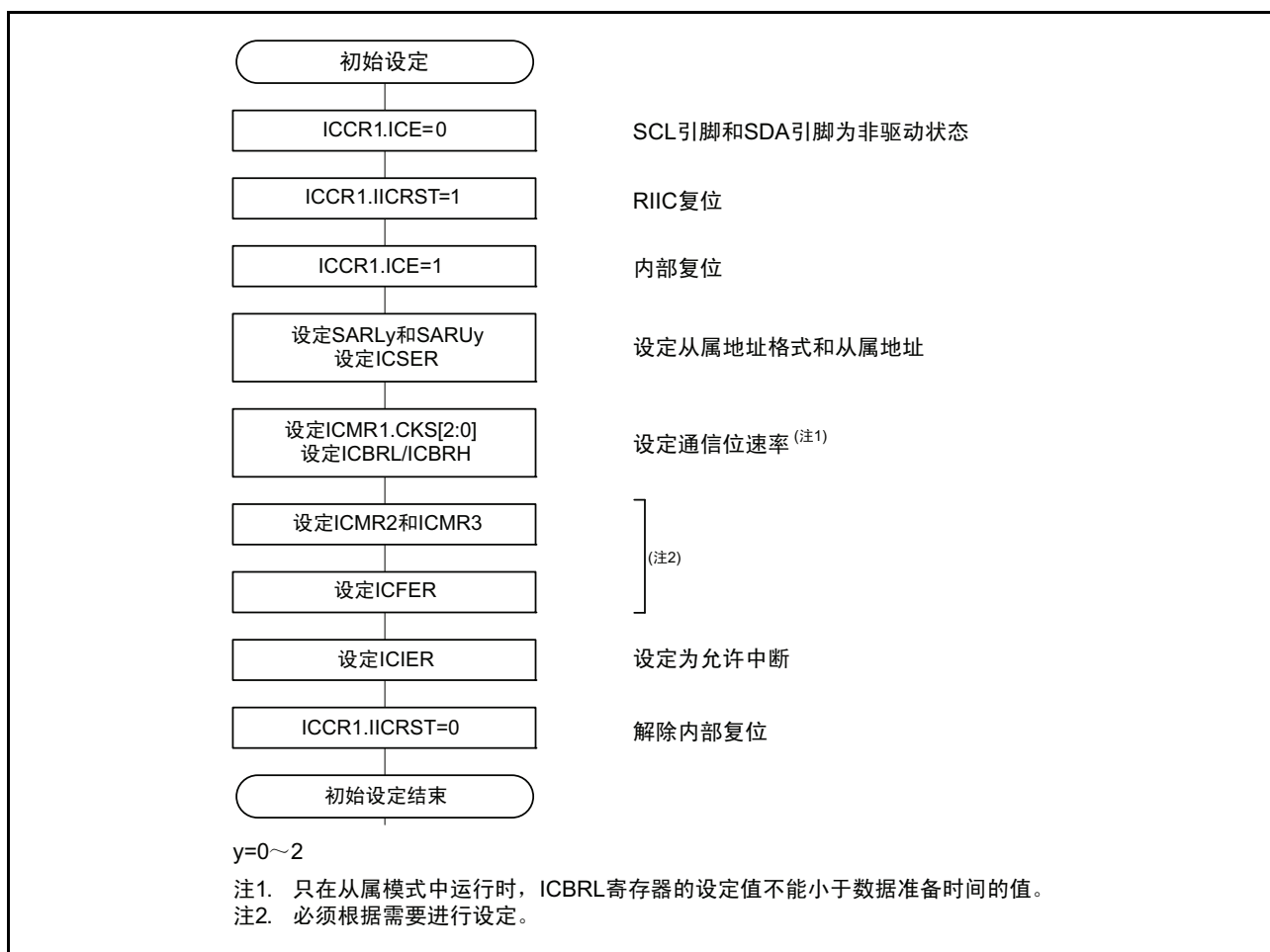


图 29.5 RIIC 的初始化流程图例子

29.3.3 主控发送

在主控发送模式中，主控设备的 RIIC 输出 SCL 时钟和发送数据，从属设备返回应答。主控发送模式的使用例子和运行时序分别如图 29.6 和图 29.7 ~ 图 29.9 所示。

主控发送模式的发送步骤和运行如下所示：

1. 在将 ICCR1.ICE 位置“0”（SCL 引脚和 SDA 引脚为非驱动状态）的状态下，通过将 ICCR1.IICRST 位置“1”（RIIC 复位）后将 ICCR1.ICE 位置“1”（内部复位），对 ICSR1 寄存器的各标志和内部状态进行初始化。然后，设定 SARLy、SARUy、ICSER、ICMR1、ICBRH、ICBRL 寄存器（y=0~2）并且必须根据需要设定其他寄存器（有关 RIIC 的初始设定请参照图 29.5）。一旦设定完所需的寄存器，就必须将 ICCR1.IICRST 位置“0”（解除复位）。在 RIIC 已初始化的情况下，不需要此步骤。
2. 读 ICCR2.BBSY 标志，在确认总线为释放状态后将 ICCR2.ST 位置“1”（请求发行开始条件）。如果 RIIC 接受开始条件的发行请求，就发行开始条件。如果 RIIC 检测到开始条件，就自动将 BBSY 标志和 ICSR2.START 标志置“1”，并且自动将 ST 位置“0”。此时，如果在 ST 位为“1”的状态下 RIIC 自身发送的 SDA 信号和 SDA 线的信号状态相同，并且检测到开始条件，RIIC 就视为通过 ST 位正确地发行了开始条件，在将 ICCR2.MST 位和 ICCR2.TRS 位自动置“1”后变为主控发送模式。另外，ICSR2.TDRE 因 TRS 位为“1”而自动变为“1”。
3. 必须在确认 ICSR2.TDRE 标志是“1”后将发送数据（从属地址和 R/W# 位）写到 ICDRT 寄存器。一旦将发送数据写到 ICDRT 寄存器，TDRE 标志就自动变为“0”，在将数据从 ICDRT 寄存器传送到 ICDRS 寄存器后，TDRE 标志再次变为“1”。一旦结束包含 R/W# 位的从属地址的发送，就根据被发送的 R/W# 位自动更改 TRS 位，并且选择发送模式或者接收模式。如果接收到为“0”的 R/W# 位，RIIC 就继续保持主控发送模式的状态。

此时，如果 ICSR2.NACKF 标志为“1”，因为没有识别到从属设备或者发生了通信故障，所以必须发行停止条件。通过给 ICCR2.SP 位写“1”来发行停止条件。

在用 10 位地址格式进行发送时，必须先在第 1 次地址发送处理中给 ICDRT 寄存器写 1111 0b+ 从属地址的高 2 位+W，然后在第 2 次地址发送处理中给 ICDRT 寄存器写从属地址的低 8 位。

4. 必须在确认 ICSR2.TDRE 标志是“1”后将发送数据写到 ICDRT 寄存器。在准备好发送数据之前或者发行停止条件前，RIIC 自动将 SCL 线保持为 Low 电平。
5. 在将要发送的全部字节写到 ICDRT 寄存器后，必须在等待 ICSR2.TEND 标志变为“1”后再给 ICCR2.SP 位写“1”（请求发行停止条件）。如果 RIIC 接受停止条件的发行请求，就发行停止条件。
6. 如果 RIIC 检测到停止条件，ICCR2.MST 位和 ICCR2.TRS 位就自动变为“00b”并且转移到从属接收模式，而且 ICSR2.TDRE 标志和 ICSR2.TEND 标志因检测到停止条件而自动变为“0”，ICSR2.STOP 标志变为“1”。
7. 在确认 ICSR2.STOP 标志是“1”后，必须将 ICSR2.NACKF 标志和 ICSR2.STOP 标志置“0”，以便进行下一次的通信。

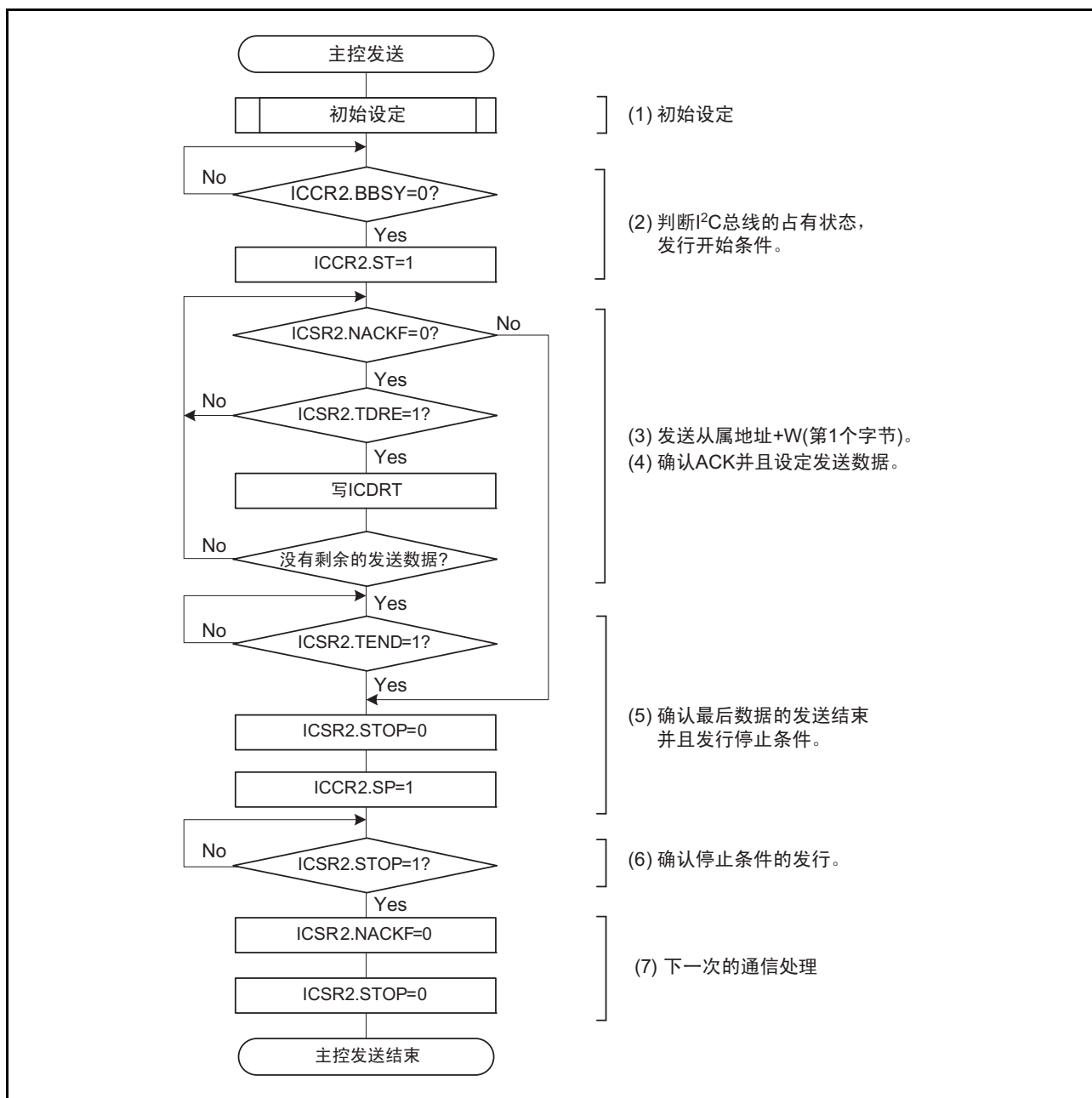


图 29.6 主控发送模式的流程图例子

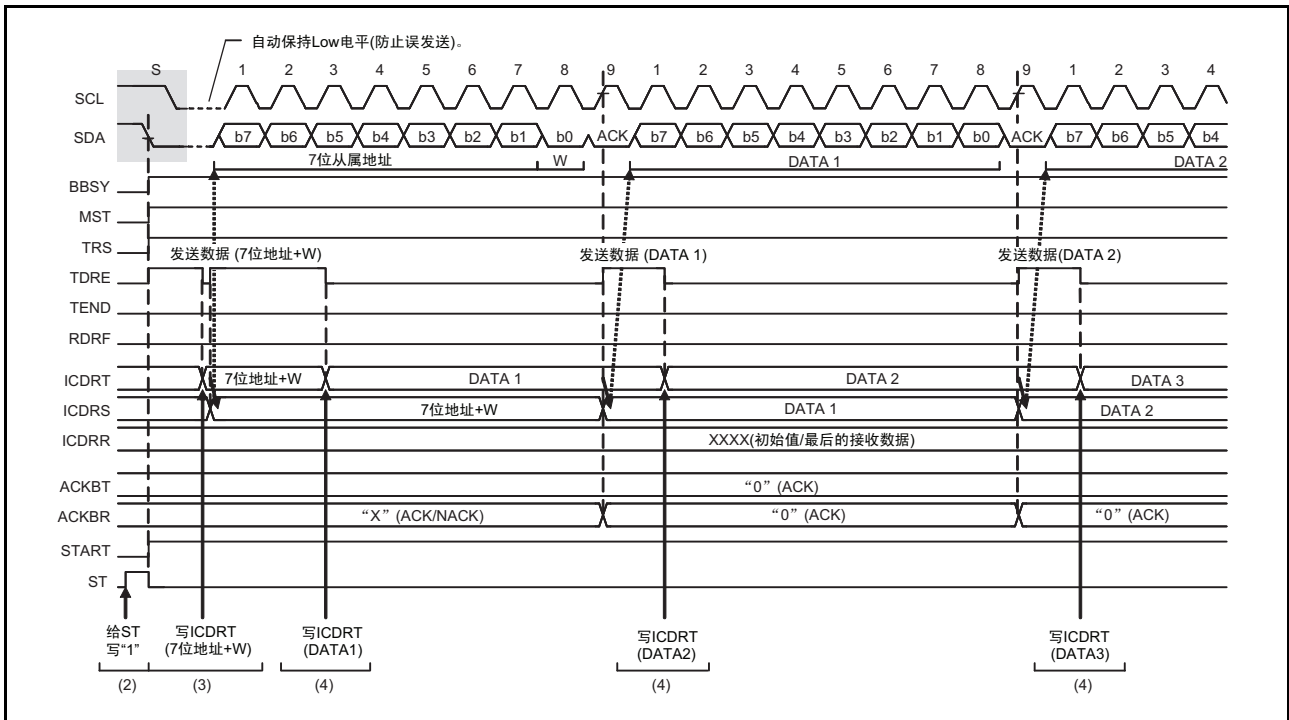


图 29.7 主控发送模式的运行时序 (1) (7 位地址格式的情况)

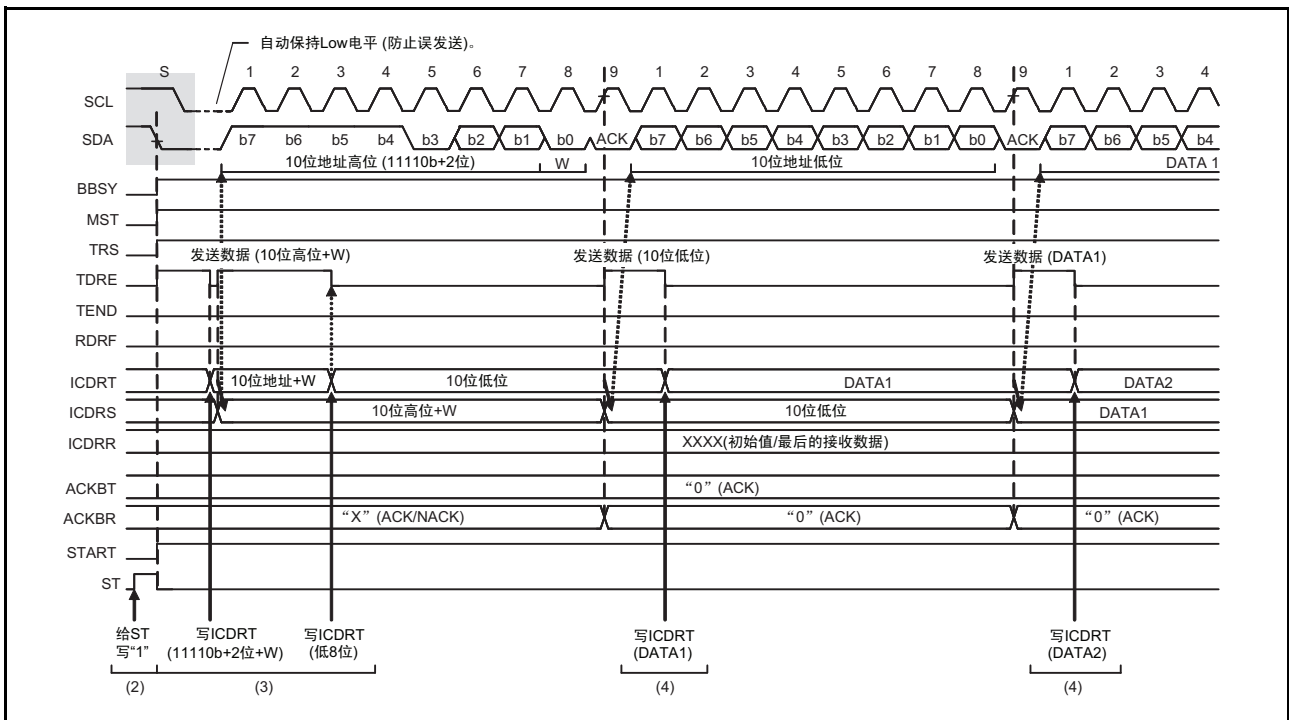


图 29.8 主控发送模式的运行时序 (2) (10 位地址格式的情况)

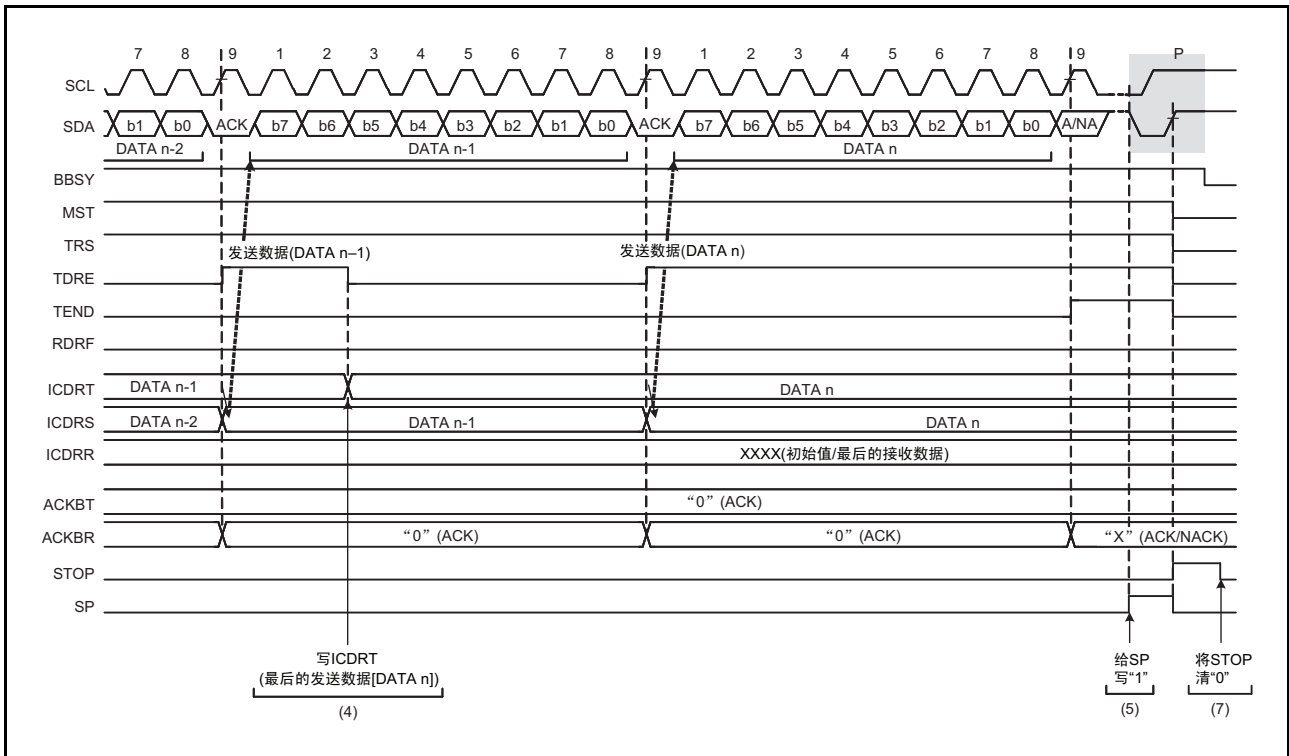


图 29.9 主控发送模式的运行时序 (3)

29.3.4 主控接收

在**主控接收模式**中，主控设备的**RIIC**输出**SCL**时钟，从属设备接收数据后返回应答。因为首先需要将从属地址发送到从属设备，所以必须先**在**主控发送模式中发送从属地址，然后**在**主控接收模式中接收数据。

主控接收模式的使用例子（7位地址格式）和运行时序分别如图29.10和图29.11～图29.13所示。

主控接收模式的接收步骤和运行如下所示：

1. 在将**ICCR1.ICE**位置“0”（**SCL**引脚和**SDA**引脚为非驱动状态）的状态下，通过将**ICCR1.IICRST**位置“1”（**RIIC**复位）后将**ICCR1.ICE**位置“1”（内部复位），对**ICSR1**寄存器的各标志和内部状态进行初始化。然后，设定**SARLy**、**SARUy**、**ICSER**、**ICMR1**、**ICBRH**、**ICBRL**寄存器（ $y=0\sim 2$ ）并且必须根据需要设定其他的寄存器（有关**RIIC**的初始设定请参照图29.5）。必须在设定完所需的寄存器后**ICCR1.IICRST**位置“0”（解除复位）。在**RIIC**已初始化的情况下，不需要此步骤。
2. 读**ICCR2.BBSY**标志，在确认总线为释放状态后将**ICCR2.ST**位置“1”（请求发行开始条件）。如果**RIIC**接受开始条件的发行请求，就发行开始条件。如果**RIIC**检测到开始条件，就自动将**BBSY**标志和**ICSR2.START**标志置“1”，并且自动将**ST**位置“0”。此时，如果在**ST**位为“1”的状态下**RIIC**自身发送的**SDA**信号和**SDA**线的状态相同，并且检测到开始条件，**RIIC**就视为通过**ST**位正常地发行了开始条件，在将**ICCR2.MST**位和**ICCR2.TRS**位自动置“1”后变为**主控发送模式**。**ICSR2.TDRE**标志因**TRIS**位为“1”而自动变为“1”。
3. 必须在确认**ICSR2.TDRE**标志是“1”后将发送数据（从属地址和**R/W#**位）写到**ICDRT**寄存器。一旦将发送数据写到**ICDRT**寄存器，**TDRE**标志就自动变为“0”，在将数据从**ICDRT**寄存器传送到**ICDRS**寄存器后，**TDRE**标志再次变为“1”。一旦结束包含**R/W#**位的从属地址的发送，就根据被发送的**R/W#**位自动更改**ICCR2.TRS**位，并且选择发送模式或者接收模式。如果**RIIC**接收到为“1”的**R/W#**位，就在第9个时钟的上升沿将**TRIS**位置“0”后转移到**主控接收模式**，此时**TDRE**标志变为“0”，**ICSR2.RDRF**标志自动变为“1”。

此时，如果**ICSR2.NACKF**标志为“1”，因为没有识别到从属设备或者发生了通信故障，所以必须发行停止条件。通过将**ICCR2.SP**位置“1”来发行停止条件。

在用10位地址格式进行**主控接收**时，先在**主控发送模式**中发送10位地址，然后发行重新开始条件。接着，通过发送1111 0b+从属地址的高2位+R，转移到**主控接收模式**。

4. 如果在确认**ICSR2.RDRF**标志是“1”后虚读**ICDRR**寄存器，**RIIC**就在输出**SCL**时钟后开始接收。
5. 结束1字节数据的接收，在**ICMR3.RDRFS**位设定的**SCL**时钟的第8个时钟或者第9个时钟的上升沿**ICSR2.RDRF**标志变为“1”。此时，如果读**ICDRR**寄存器，就能读到接收数据，同时**RDRF**标志自动变为“0”，并且将**ICMR3.ACKBT**位的设定值返回给**SCL**时钟的第9个时钟的应答位。另外，当下一个接收字节为最后字节-1时，必须在读**ICDRR**寄存器（第(最后字节-2)个字节）前将**ICMR3.WAIT**位置“1”（有**WAIT**）。因此，即使在将**ICMR3.ACKBT**位置“1”（**NACK**）的后续6.处理迟于其他中断等的情况下，也能在最后字节时输出**NACK**，并且同时能在接收最后字节时的第9个时钟的下降沿将**SCLn**线固定为**Low**电平，进入能发行停止条件的状态。
6. 在**ICMR3.RDRFS**位为“0”并且需要将“下一个数据接收后通信结束”通知从属设备时，必须将**ICMR3.ACKBT**位置“1”（**NACK**）。
7. 在读**ICDRR**寄存器（第(最后字节-1)个字节）后，必须先确认**ICSR2.RDRF**标志是“1”，然后给**ICCR2.SP**位写“1”（请求发行停止条件）并且读**ICDRR**寄存器（最后字节）。**RIIC**通过读**ICDRR**寄存器来解除**WAIT**状态，在结束第9个时钟的**Low**电平输出或者解除**SCL**线保持的**Low**电平后发行停止条件。
8. 如果**RIIC**检测到停止条件，**ICCR2.MST**位和**ICCR2.TRS**位就自动变为“00b”并且转移到从属接收模式，而且**ICSR2.STOP**标志因检测到停止条件而变为“1”。
9. 在确认**ICSR2.STOP**标志是“1”后，必须将**ICSR2.NACKF**和**ICSR2.STOP**标志置“0”，以便进行下一次的通信。

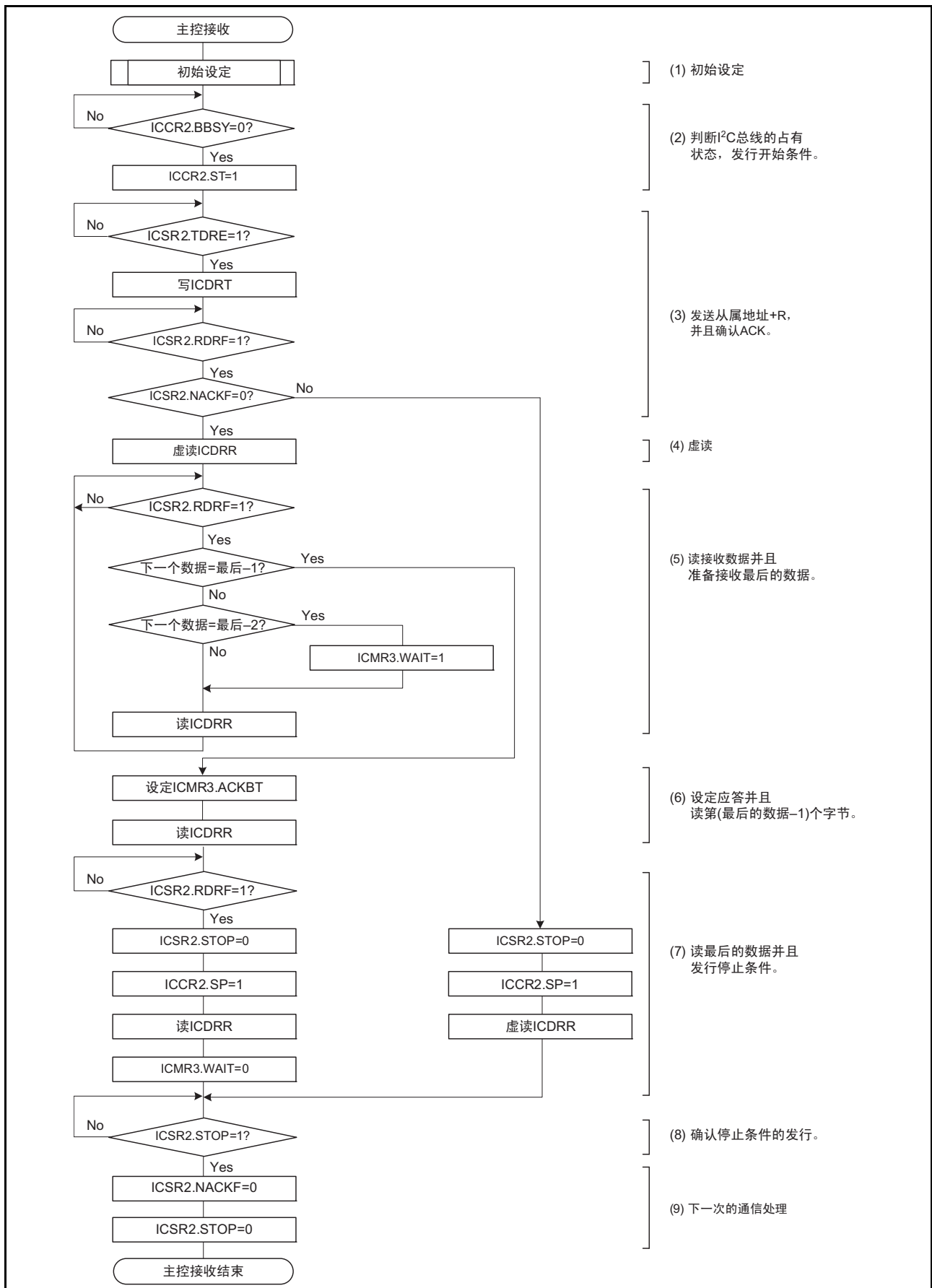


图 29.10 主控接收模式的流程图例子 (7 位地址格式的情况)

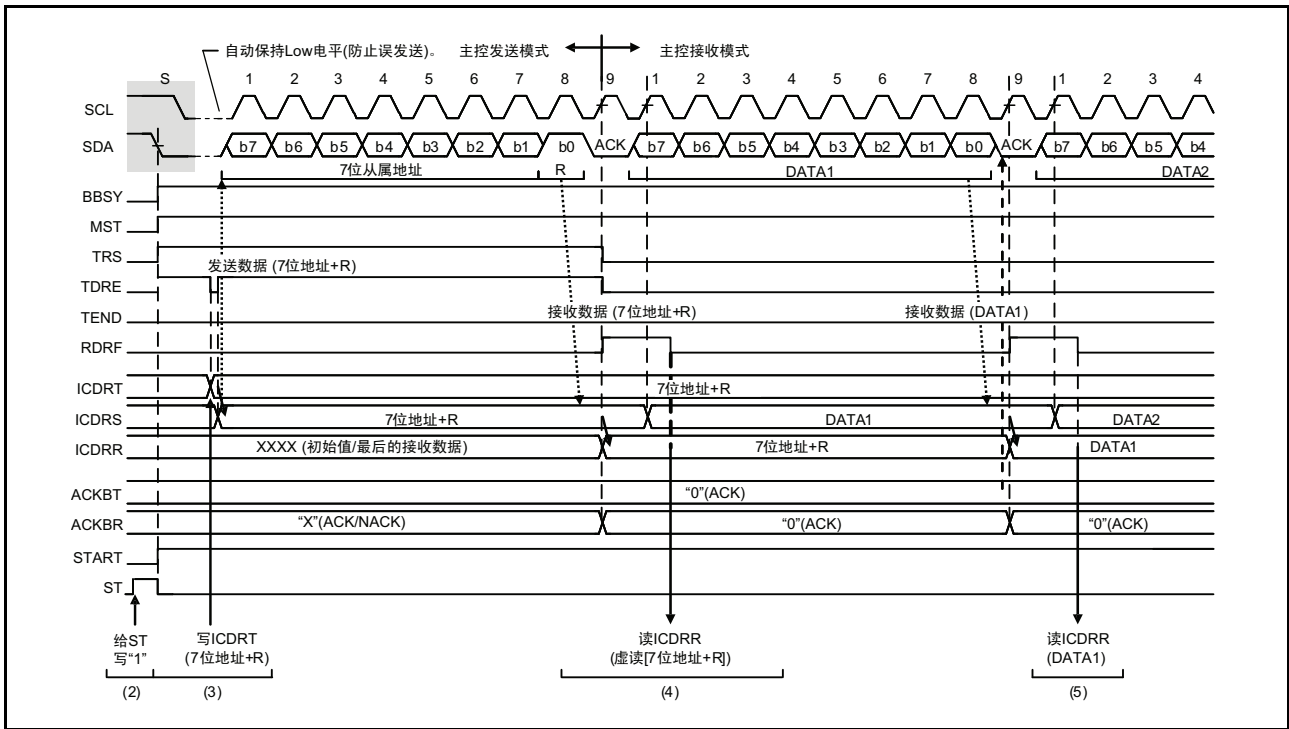


图 29.11 主控接收模式的运行时序 (1) (7 位地址格式并且 RDRFS 位为“0”的情况)

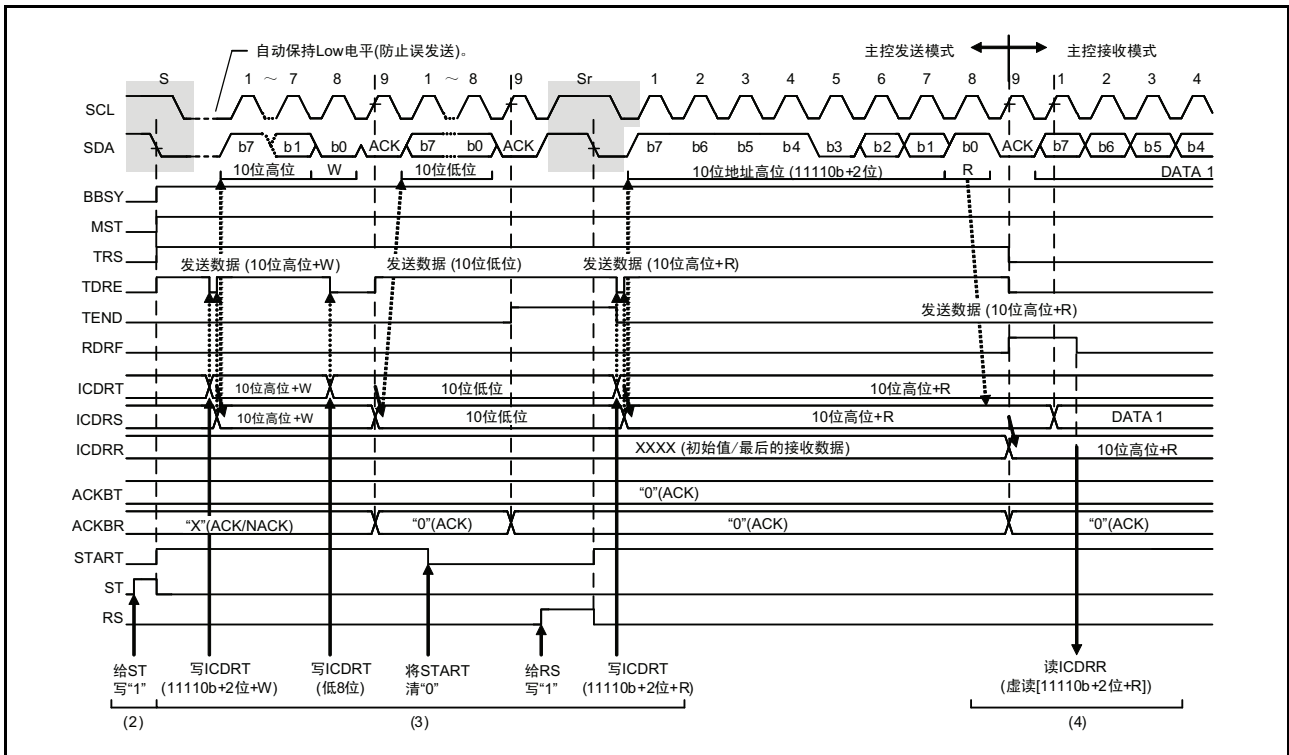


图 29.12 主控接收模式的运行时序 (2) (10 位地址格式并且 RDRFS 位为“0”的情况)

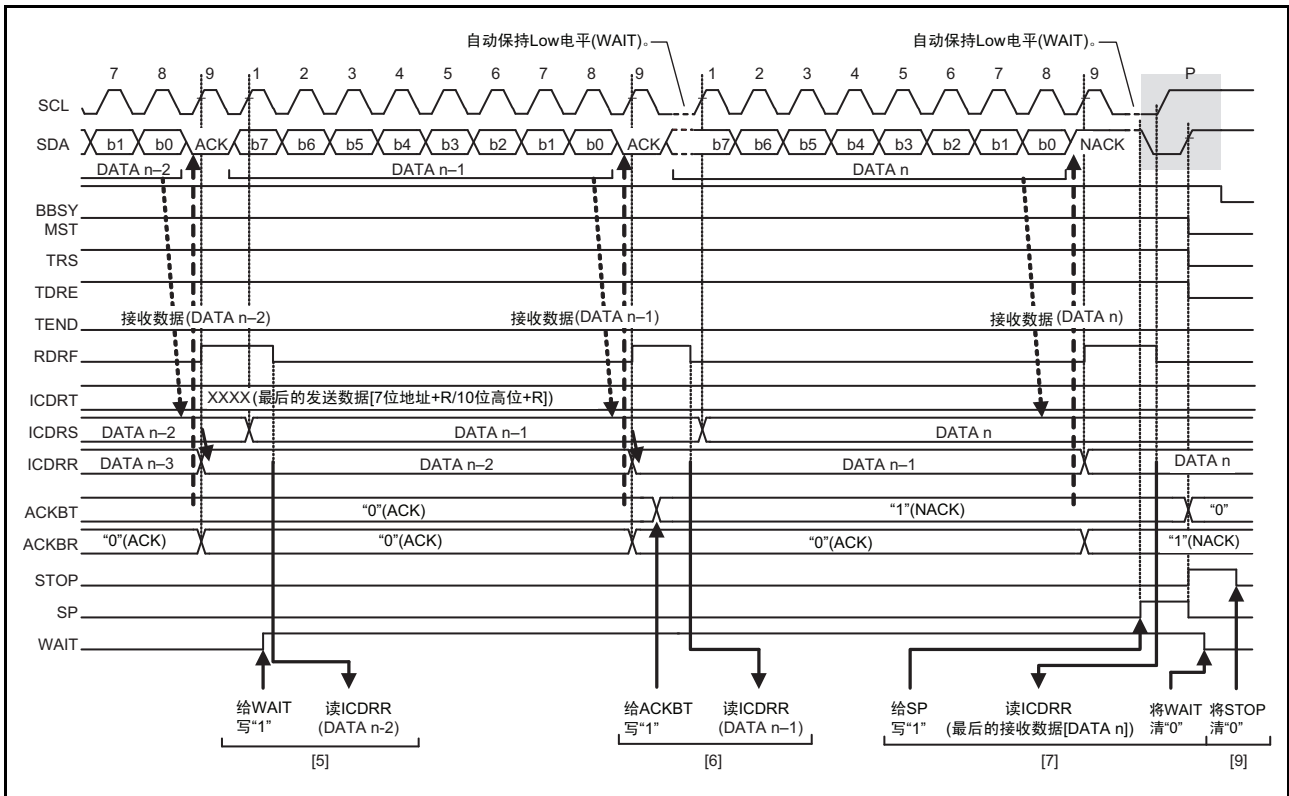


图 29.13 主控接收模式的运行时序 (3) (RDRFS 位为“0”的情况)

29.3.5 从属发送

在从属发送模式中，主控设备输出 SCL 时钟，从属设备的 RIIC 发送数据，并且主控设备返回应答。

从属发送模式的使用例子和运行时序分别如图 29.14 和图 29.15 ~ 图 29.16 所示。

从属发送模式的发送步骤及其运行如下所示：

1. 必须按照图 29.5 所示的步骤对 RIIC 进行初始设定。在 RIIC 已初始化的情况下，不需要此步骤。从初始设定结束后到从属地址匹配前，RIIC 处于待机状态。
2. 如果从属地址匹配，RIIC 就在 SCL 时钟的第 9 个时钟上升沿将对应的 ICSR1.HOA 位、ICSR1.GCA 位、或者 ICSR1.AASy 位 (y=0~2) 置“1”，并且将 ICMR3.ACKBT 位的设定值返回给 SCL 时钟的第 9 个时钟的应答位。此时，如果接收的 R/W# 位为“1”，就将 ICCR2.TRS 位和 ICSR2.TDRE 标志置“1”并且自动转换为从属发送模式。
3. 必须在确认 ICSR2.TDRE 标志是“1”后将发送数据写到 ICDRT 寄存器。此时，如果在 ICFER.NACKF 位为“1”的状态下主控设备没有应答（接收到 NACK），RIIC 就中止下一次的通信。
4. 在 ICSR2.NACKF 标志变为“1”或者将最后的发送数据写到 ICDRT 寄存器后，必须在 ICSR2.TDRE 标志为“1”的状态下等待 ICSR2.TEND 标志变为“1”。当 ICSR2.NACKF 标志或者 TEND 标志为“1”时，RIIC 在第 9 个时钟下降后将 SCL 线保持为 Low 电平。
5. 当 ICSR2.NACKF 标志或者 ICSR2.TEND 标志为“1”时，必须通过虚读 ICDRR 寄存器来结束处理，从而释放 SCL 线。
6. 如果 RIIC 检测到停止条件，就自动将 ICSR1.HOA 位、ICSR1.GCA 位、ICSR1.AASy 位 (y=0~2)、ICSR2.TDRE 位、ICSR2.TEND 标志和 ICCR2.TRS 位置“0”并且转移到从属接收模式。
7. 在确认 ICSR2.STOP 标志是“1”后，必须将 ICSR2.NACKF 标志和 ICSR2.STOP 标志置“0”，以便进行下一次的通信。

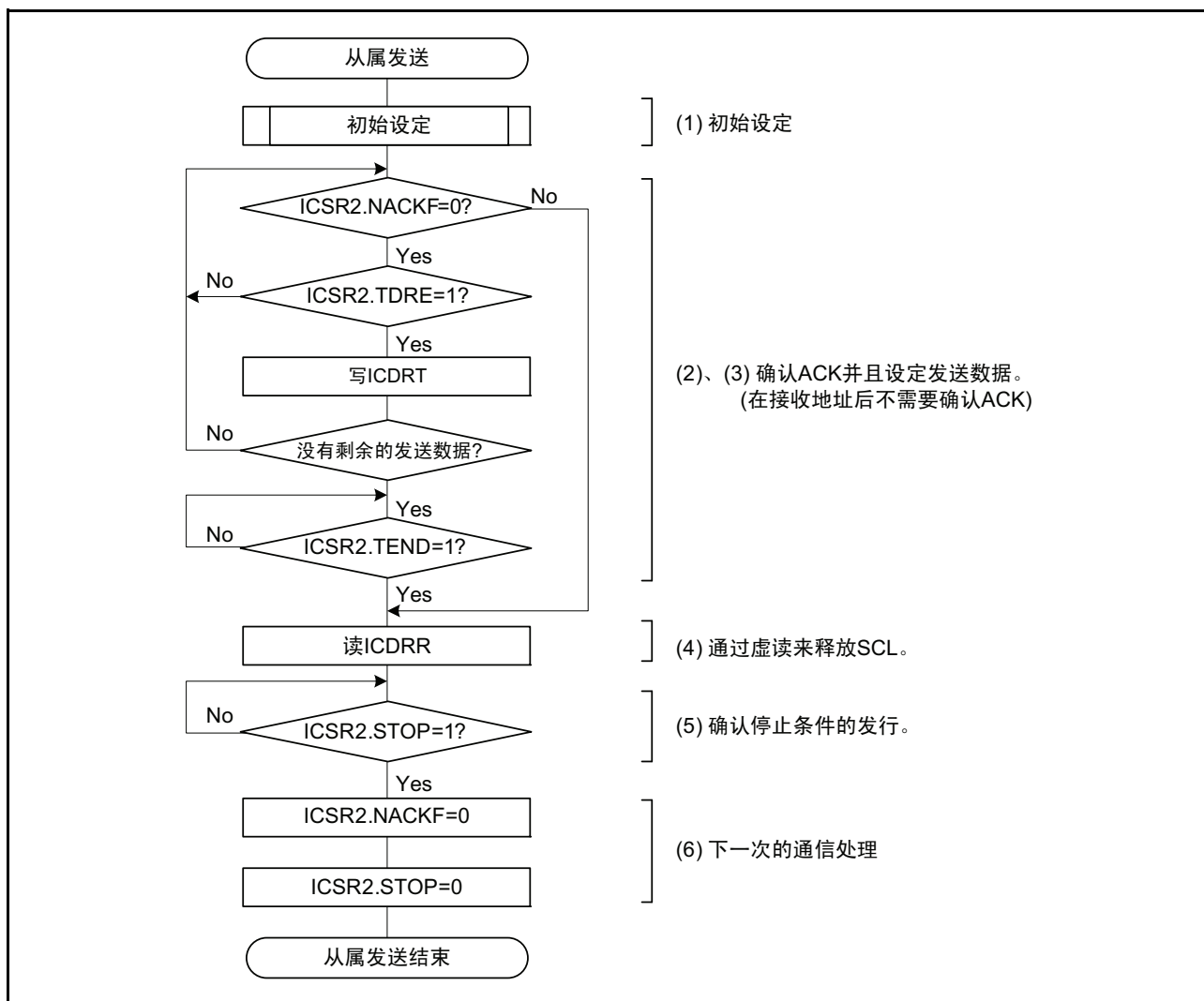


图 29.14 从属发送模式的流程图例子

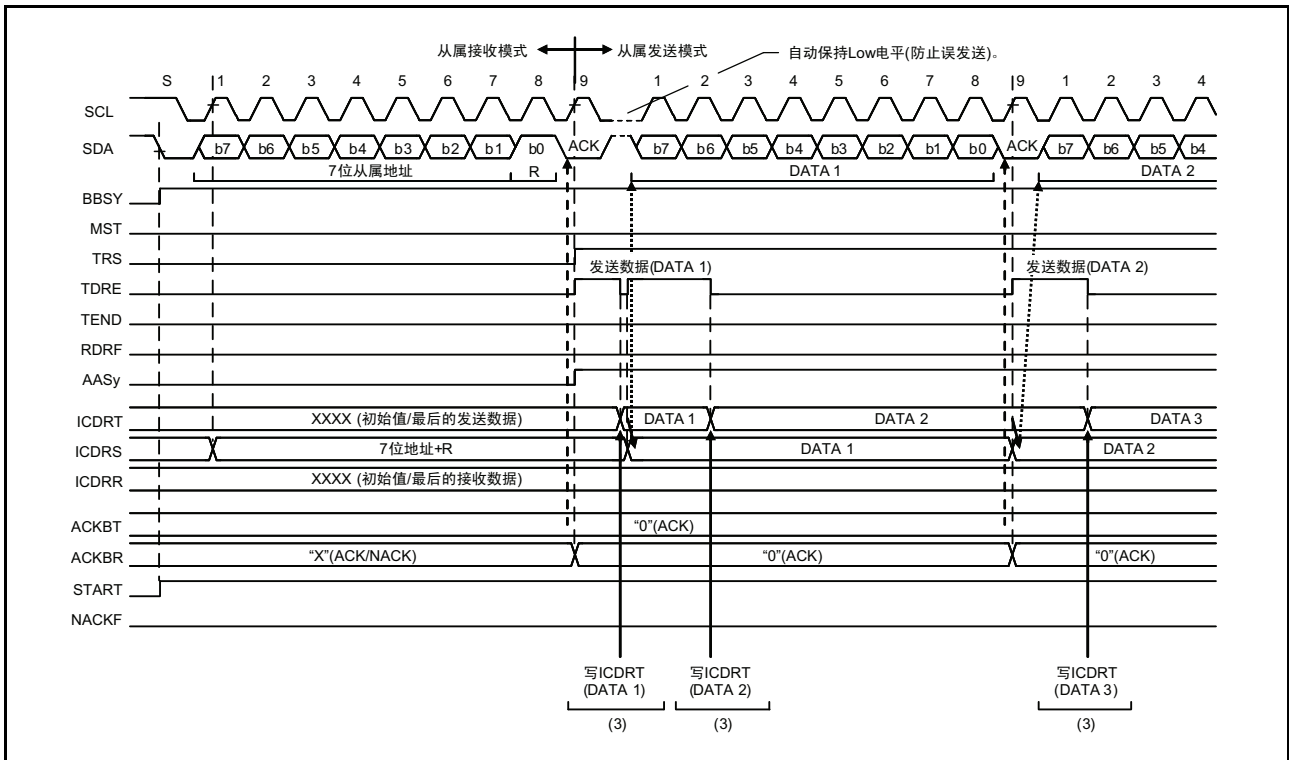


图 29.15 从属发送模式的运行时序 (1) (7 位地址格式的情况)

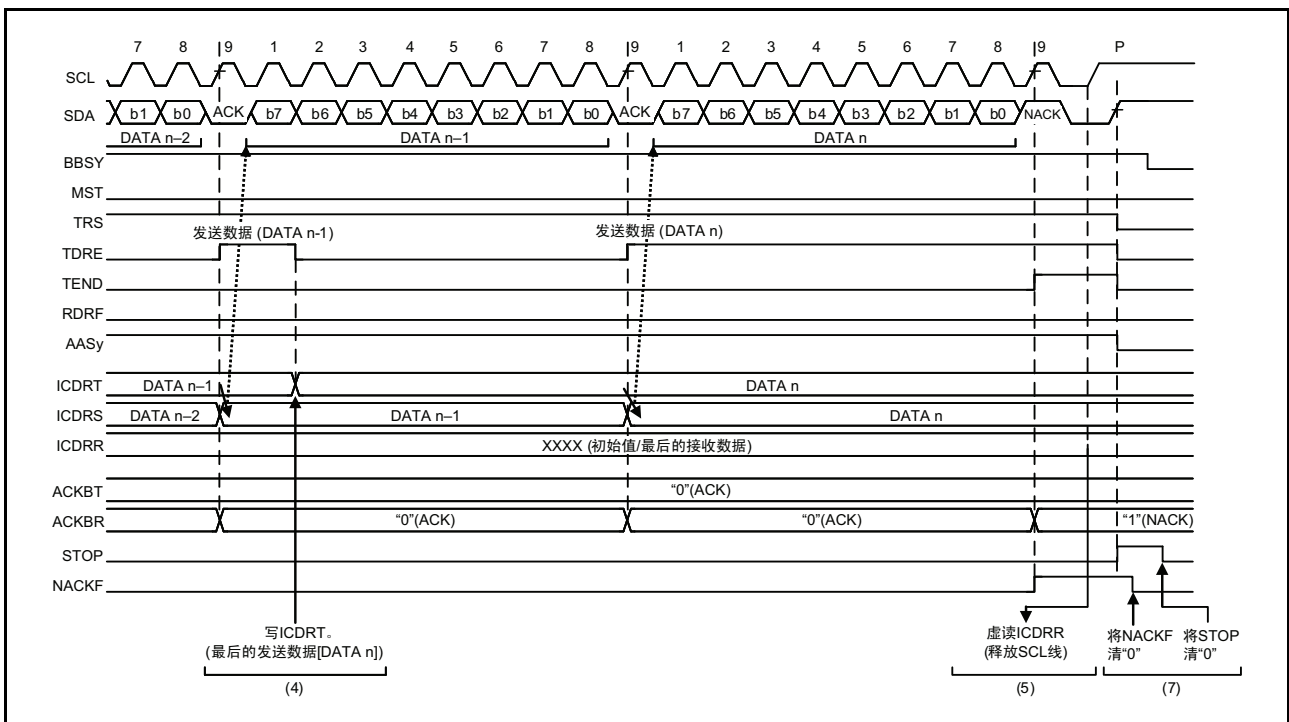


图 29.16 从属发送模式的运行时序 (2)

29.3.6 从属接收

在从属接收模式中，主控设备输出 SCL 时钟和发送数据，从属设备的 RIIC 返回应答。

从属接收模式的使用例子和运行时序分别如图 29.17、图 29.18 和图 29.19 所示。

从属接收模式的接收步骤和运行如下所示：

1. 必须按照图 29.5 所示的步骤对 RIIC 进行初始设定。在 RIIC 已初始化的情况下，不需要此步骤。在初始设定结束后到从属地址匹配前，RIIC 处于待机状态。
2. 如果从属地址匹配，RIIC 就在 SCL 时钟的第 9 个时钟的上升沿将对应的 ICSR1.HOA 位、ICSR1.GCA 位或者 ICSR1.AASy 位 (y=0~2) 置“1”并且将 ICMR3.ACKBT 位的设定值返回给 SCL 时钟的第 9 个时钟的应答位。此时，如果接收的 R/W# 位为“0”，就继续保持从属接收模式的状态并且将 ICSR2.RDRF 标志置“1”。
3. 在确认 ICSR2.STOP 标志是“0”并且 ICSR2.RDRF 标志是“1”后，第 1 次必须虚读 ICDRR 寄存器（虚读接收数据在 7 位地址格式时为从属地址 +R/W# 位，在 10 位地址格式时为低 8 位地址）。
4. 如果读 ICDRR 寄存器，RIIC 就自动将 ICSR2.RDRF 标志置“0”。如果在延迟了 ICDRR 寄存器的读操作并且 RDRF 标志为“1”的状态下接收下一个数据，RIIC 就在 RDRF 标志变为“1”时的前一个 SCL 时钟下降沿将 SCL 线保持为 Low 电平。通过读 ICDRR 寄存器来解除被保持的 Low 电平，RIIC 释放 SCL 线。必须在 ICSR2.STOP 标志和 ICSR2.RDRF 标志都为“1”或者接收完全部数据时读 ICDRR 寄存器。
5. 如果 RIIC 检测到停止条件，就自动将 ICSR1.HOA 位、ICSR1.GCA 位和 ICSR1.AASy 位 (y=0~2) 置“0”。
6. 在确认 ICSR2.STOP 标志是“1”后，必须将 ICSR2.STOP 标志置“0”，以便进行下一次的通信。

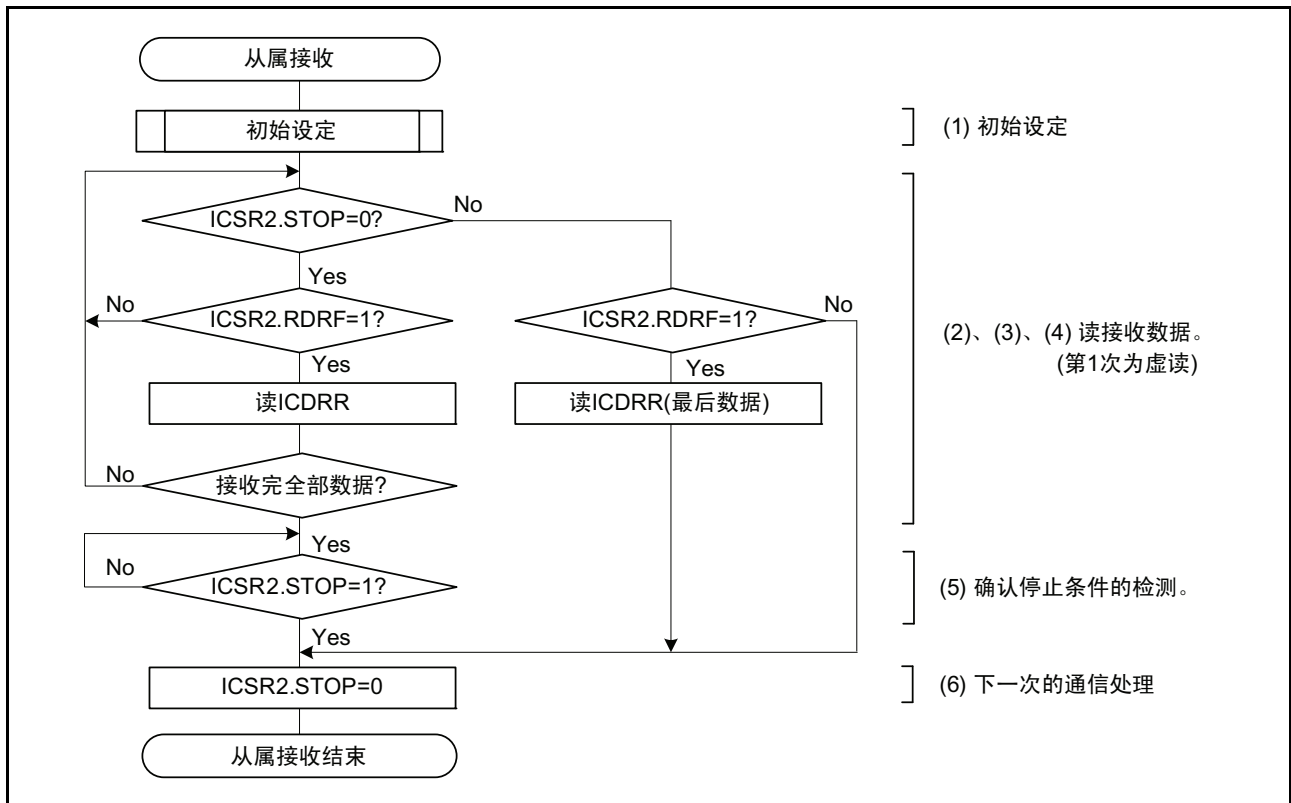


图 29.17 从属接收模式的流程图例子

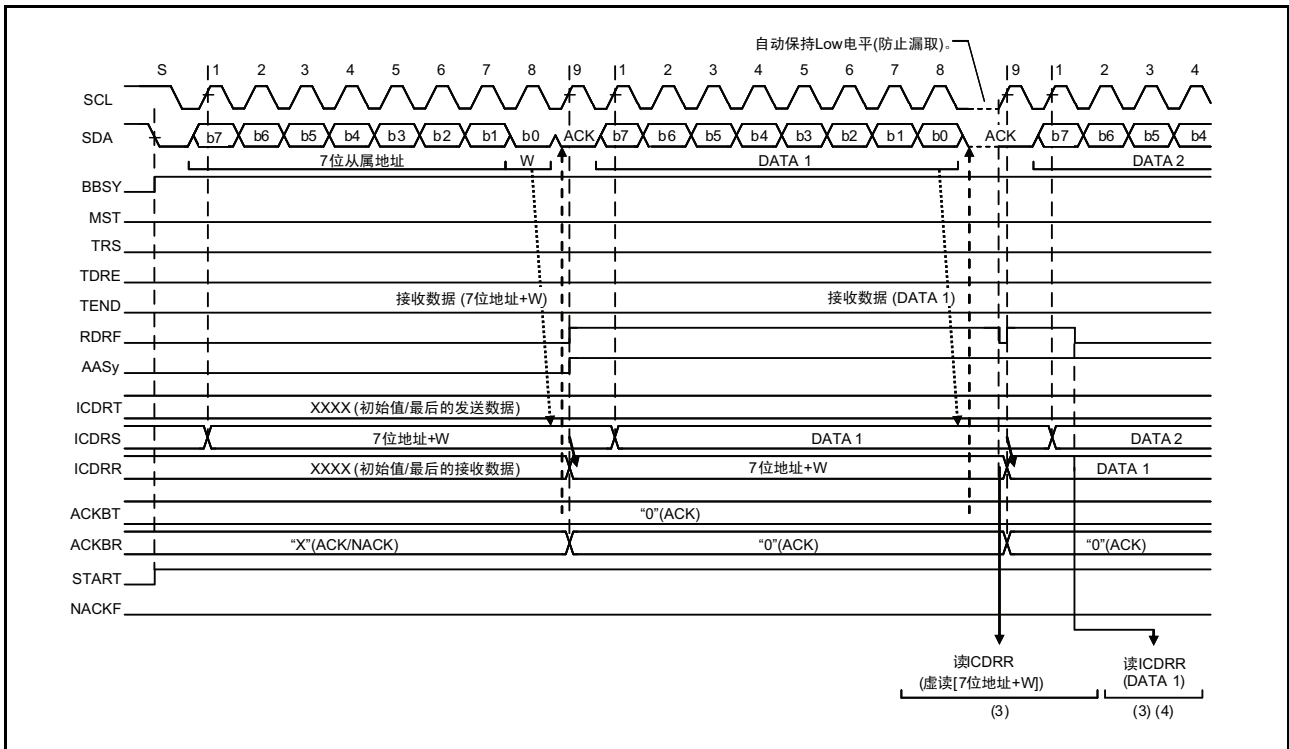


图 29.18 从属接收模式的运行时序 (1) (7 位地址格式并且 RDRFS 位为“0”的情况)

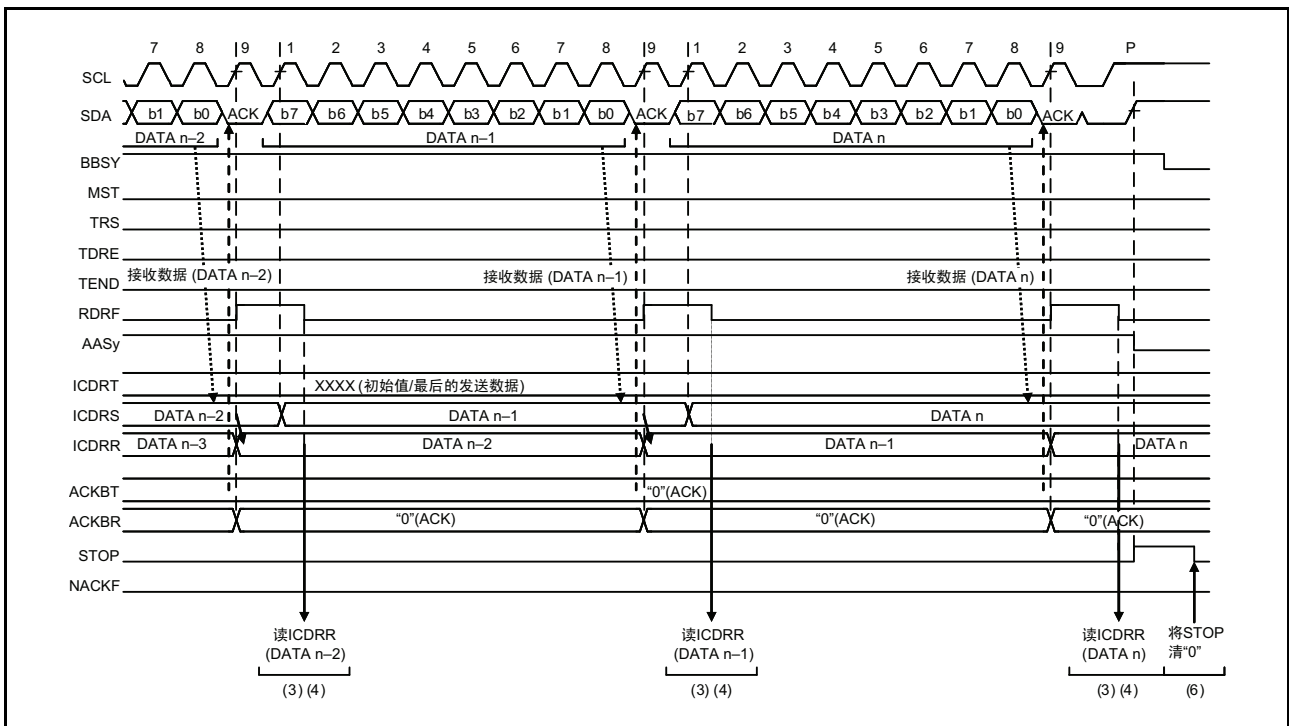


图 29.19 从属接收模式的运行时序 (2) (RDRFS 位为“0”的情况)

29.4 SCL 同步电路

如果 RIIC 的 SCL 时钟生成电路检测到 SCL 线的上升沿, 就开始对 ICBRH 寄存器设定的 High 电平宽度进行计数, 在结束 High 电平宽度的计数时将 SCL 线驱动为 Low 电平。如果检测到 SCL 线的下降沿, 就开始对 ICBRL 寄存器设定的 Low 电平宽度进行计数, 在结束 Low 电平宽度的计数时结束 SCL 线的 Low 电平驱动, 然后释放 SCL 线。通过此方法生成 SCL 时钟。

在多主控模式中使用 I²C 总线时, 有可能因和其他主控设备的竞争而使 SCL 时钟和 SCL 时钟发生冲突。如果 SCL 时钟发生冲突, 主控设备就需要与 SCL 时钟同步, 而且需要逐位进行 SCL 时钟的同步。RIIC 具有 SCL 同步电路功能, 在主控模式中监视 SCL 线, 并且一边逐位取得同步, 一边生成 SCL 时钟。

RIIC 检测到 SCL 线的上升沿并且在对 ICBRH 寄存器设定的 High 电平进行计数的过程中, 如果因其他主控设备的 SCL 时钟输出而使 SCL 线下降, RIIC 就在检测到 SCL 线的下降沿时中止 High 电平宽度的递增计数, 并且在 SCL 线被驱动为 Low 电平的同时开始对 ICBRL 寄存器设定的 Low 电平宽度进行递增计数, 在结束 Low 电平宽度的计数时结束 SCL 线的 Low 电平驱动, 然后释放 SCL 线。此时, 如果其他主控设备的 SCL 时钟的 Low 电平宽度大于 RIIC 设定的 Low 电平宽度, 就延长 SCL 时钟的 Low 电平宽度。当其他主控设备结束 Low 电平输出时, 释放 SCL 线并且 SCL 时钟上升。因此, 在发生 SCL 时钟输出冲突时, SCL 时钟的 High 电平宽度与短时钟同步, Low 电平宽度与长时钟同步。此 SCL 同步在 ICFER.SCLE 位为“1”时有效。

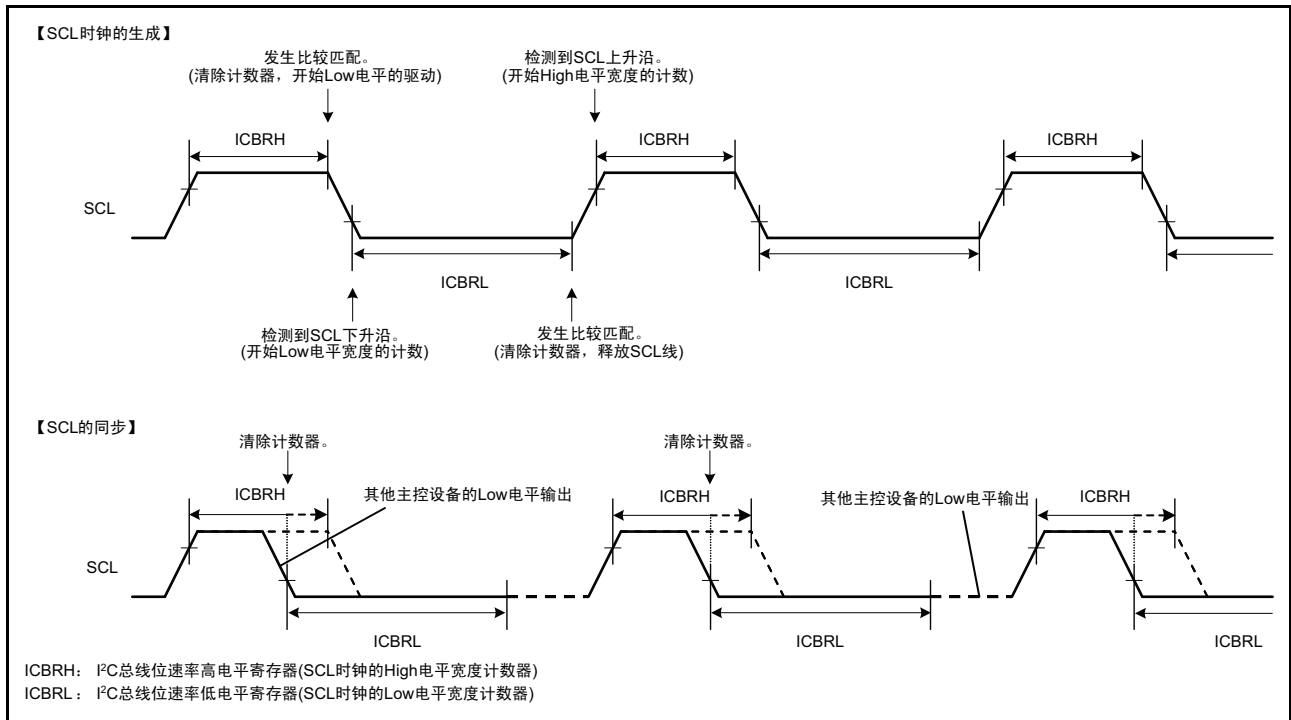


图 29.20 RIIC 的 SCL 时钟生成和 SCL 同步

29.5 SDA 输出延迟功能

RIIC 有 SDA 输出延迟功能。能通过 SDA 输出延迟功能，使全部的 SDA 输出时序（发行（开始 / 重新开始 / 停止）、数据输出和 ACK/NACK 输出）延迟。

SDA 输出延迟功能是从检测到 SCL 时钟的下降沿开始延迟 SDA 的输出，并且通过在 SCL 时钟的 Low 电平期间确保 SDA 的输出，防止通信设备的误动作，也能用于满足 SMBus 数据保持时间（300ns (min)）的规格。

SDA 输出延迟功能在 ICMR2.SDDL[2:0] 位不为“000b”时有效，在 SDDL[2:0] 位为“000b”时无效。

在 SDA 输出延迟功能有效（SDDL[2:0] 位不为“000b”）时，SDA 输出延迟计数器将 ICMR2.DLCS 位选择的内部基准时钟（IIC ϕ ）或者内部基准时钟的 2 分频时钟（IIC ϕ /2）作为计数源进行 SDDL[2:0] 位设定周期的计数，RIIC 在结束延迟周期的计数时进行 SDA 输出（发行（开始 / 重新开始 / 停止）、数据输出和 ACK/NACK 输出）。

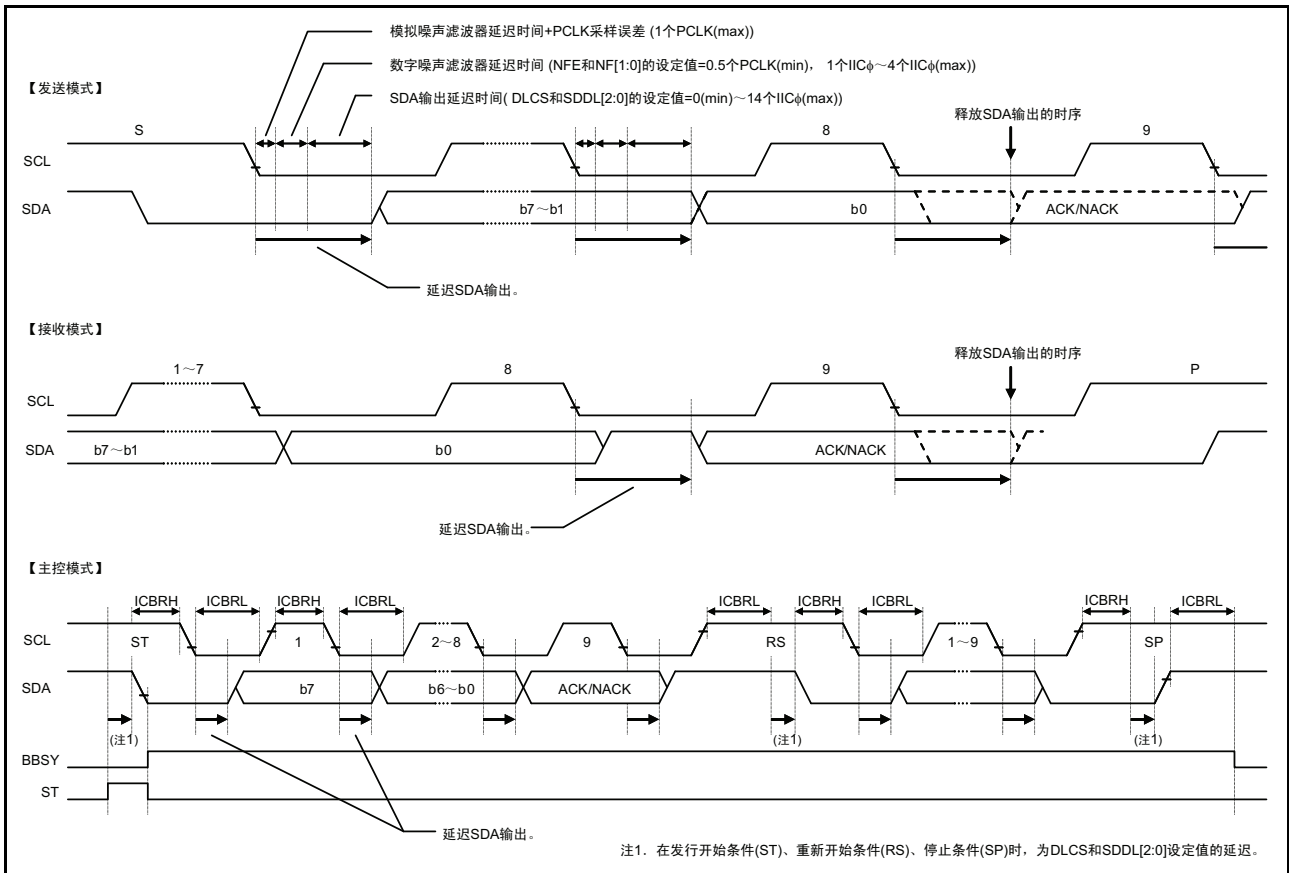


图 29.21 SDA 输出延迟时序

29.6 数字噪声滤波器电路

SCL 引脚和 SDA 引脚的状态经由模拟噪声滤波器电路和数字噪声滤波器电路进入内部。数字噪声滤波器电路的框图如图 29.22 所示。

RIIC 的内部数字噪声滤波器电路由 4 段串联的触发电路和相同检测电路构成。

通过 ICMR3.NF[1:0] 位选择数字噪声滤波器的有效段数，根据选择的有效段数，噪声消除能力为 1 ~ 4 个 IIC ϕ 周期。

在 IIC ϕ 的下降沿对 SCL 引脚的输入信号（或者 SDA 引脚的输入信号）进行采样，如果 ICMR3.NF[1:0] 位设定的有效段数的触发电路输出全部匹配，就将该电平作为内部信号进行传输，否则就保持原来的值。

在内部运行时钟（PCLK）与通信速度的比小的情况下（如：PCLK 为 4MHz 时的 400kbps 的通信），根据数字噪声滤波器的特性，有可能在产生噪声时需要的信号也被消除。在此情况下，能禁止（ICFER.NFE 位 =0）使用数字噪声滤波器电路而只使用模拟噪声滤波器电路。

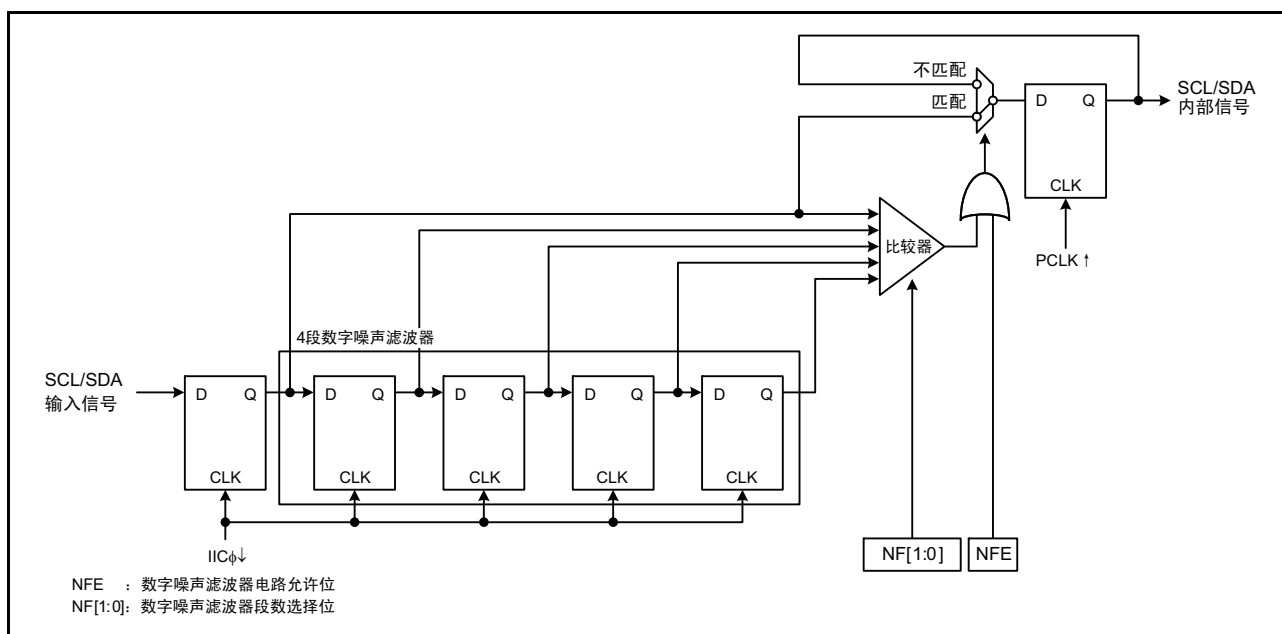


图 29.22 数字噪声滤波器电路的框图

29.7 地址匹配检测功能

RIIC 能设定全呼地址和主机地址以外的 3 种从属地址，从属地址能设定 7 位地址或者 10 位地址。

29.7.1 从属地址匹配检测功能

RIIC 能设定 3 种从属地址，有分别对应的从属地址检测功能。当 IC_{SER}.SAR_yE 位 (y=0 ~ 2) 为“1”时，能检测到 SAR_{Uy}/SAR_{Ly} 寄存器 (y=0 ~ 2) 设定的从属地址。

如果 RIIC 检测到设定的从属地址匹配，就在 SCL 时钟的第 9 个时钟的下降沿将对应的 IC_{SR1}.AAS_y 标志 (y=0 ~ 2) 置“1”，然后根据后续的 R/W# 位将 IC_{SR2}.RDRF 标志或者 IC_{SR2}.TDRE 标志置“1”。因此，能产生接收数据满中断 (IC_{RXI}) 或者发送数据空中断 (IC_{TXI})，并且能通过确认 AAS_y 标志判断指定了哪个从属地址。

AAS_y 标志变为“1”的时序如图 29.23 ~ 图 29.25 所示。

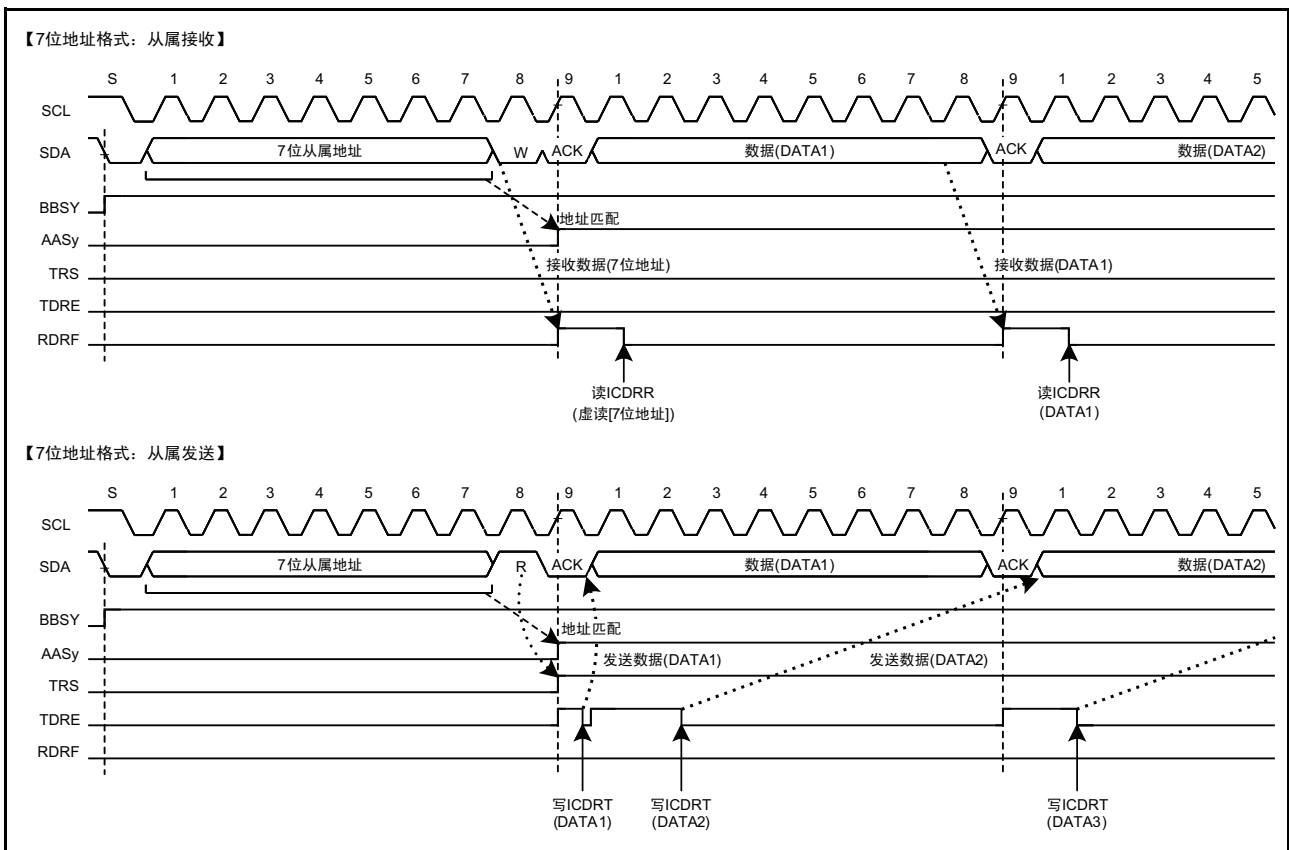


图 29.23 选择 7 位地址格式时的 AAS_y 标志变为“1”的时序

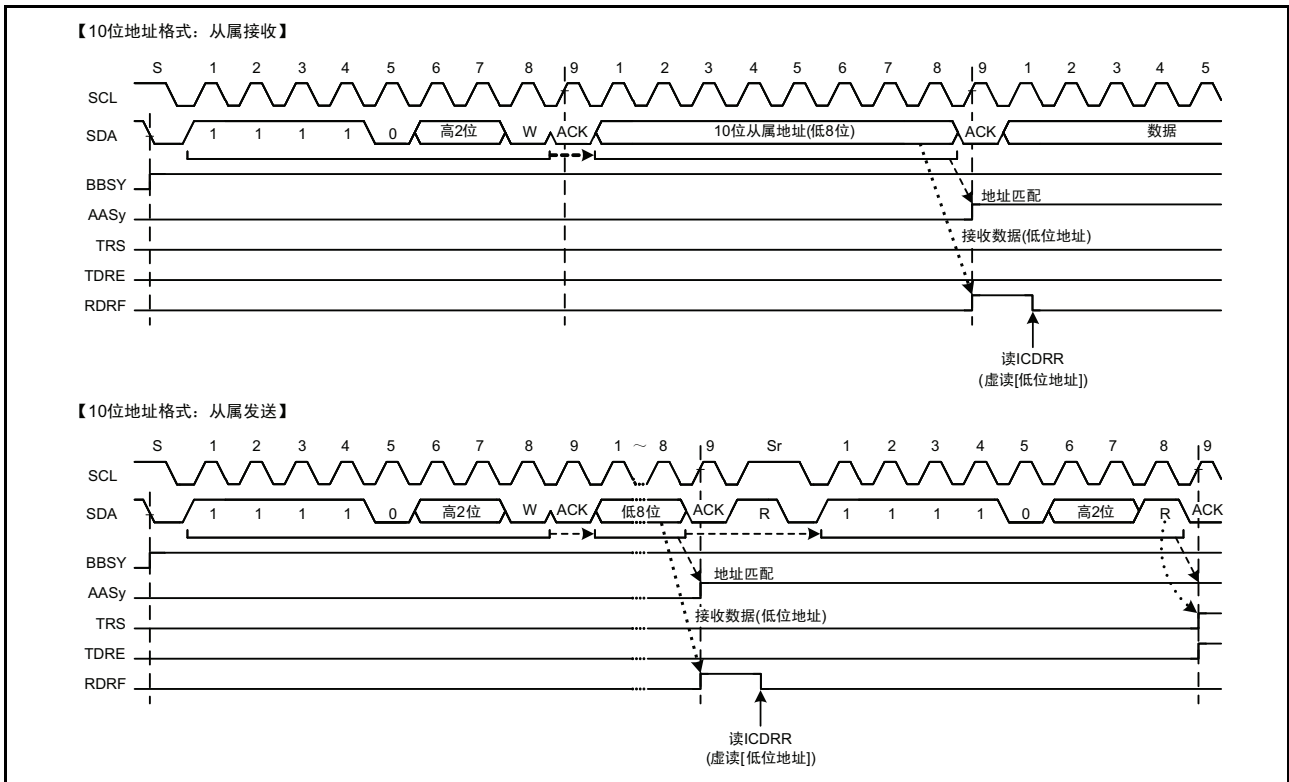


图 29.24 选择 10 位地址格式时的 AASy 标志变为“1”的时序

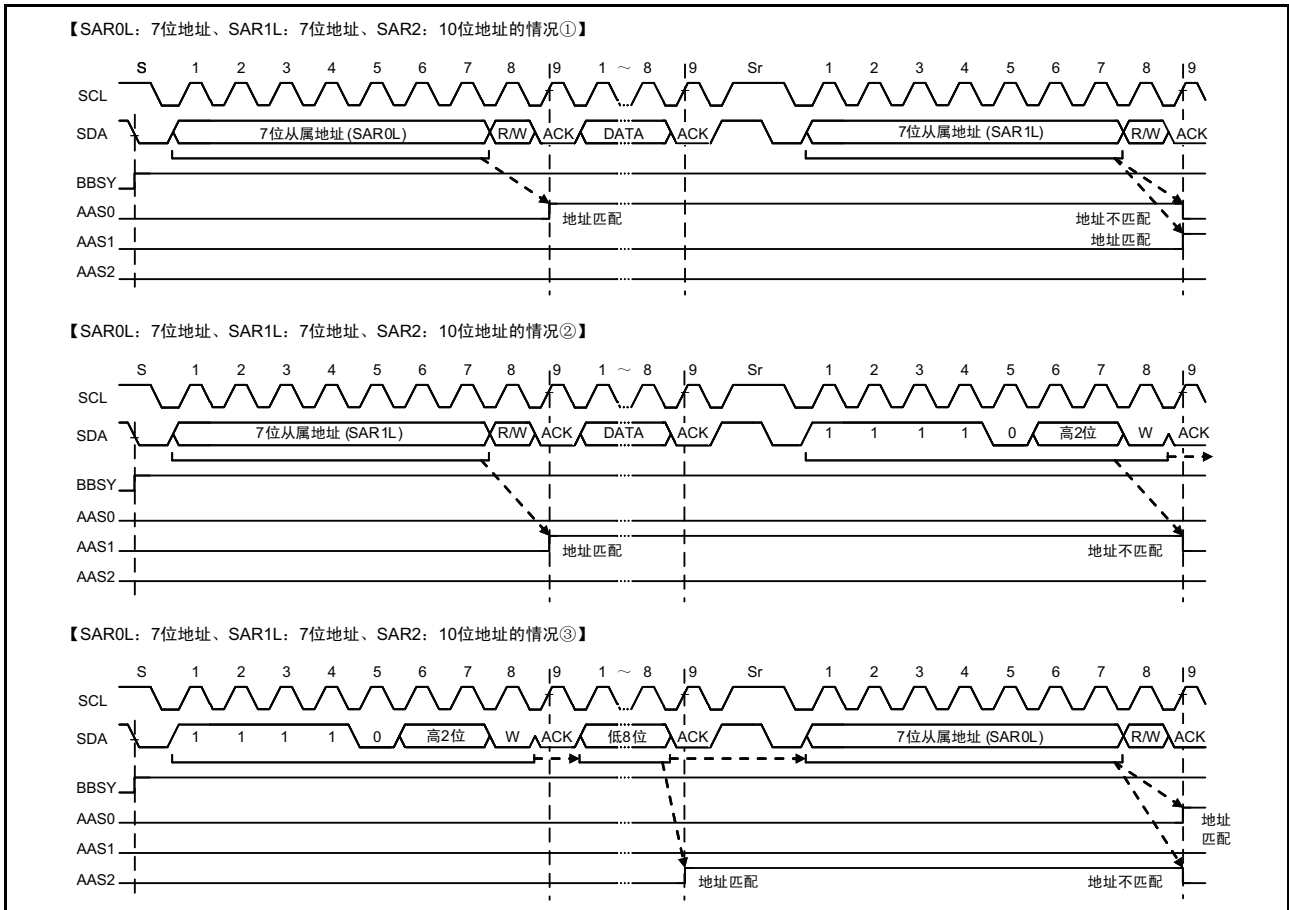


图 29.25 7 位 /10 位地址格式同时存在时的 AASy 标志变为“1”/“0”的时序

29.7.2 全呼地址检测功能

RIIC 具有全呼地址 (0000 000b+0[W]) 检测功能。当 ICSER.GCAE 位为 “1” 时, 能检测全呼地址。

如果开始条件或者重新开始条件后的地址为 0000 000b+1[R] (开始字节), RIIC 就将此地址视为 All“0” 的从属地址而不视为全呼地址。

如果 RIIC 检测到全呼地址, 就在 SCL 时钟的第 9 个时钟的下降沿将 ICSR1.GCA 标志置 “1”, 同时将 ICSR2.RDRF 标志置 “1”。因此, 能产生接收数据满中断 (ICRXI), 并且能通过确认 GCA 标志判断是否发送了全呼地址。

全呼地址检测后的运行和普通的从属接收运行相同。

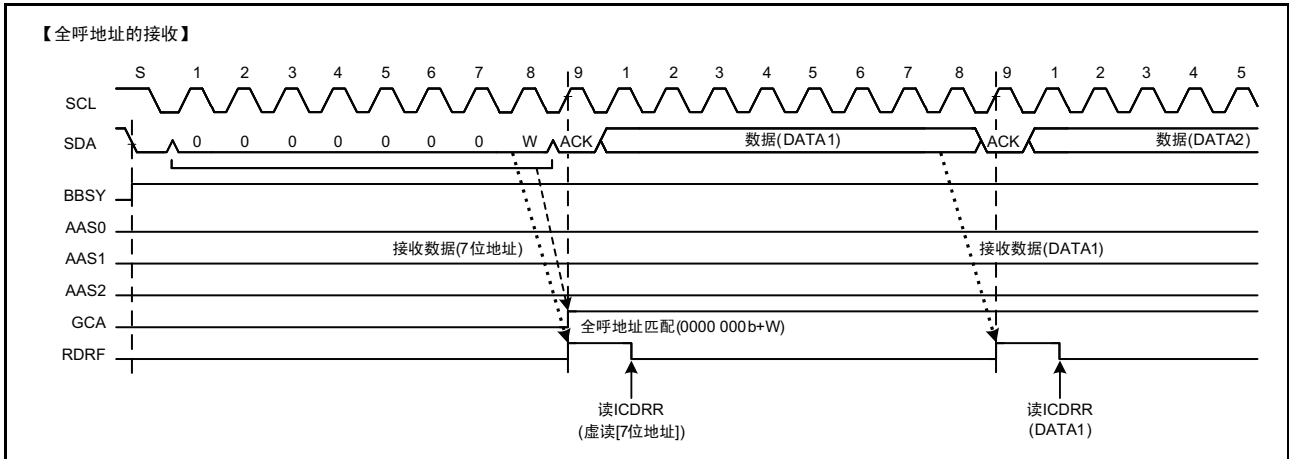


图 29.26 接收全呼地址时的 GCA 标志变为 “1” 的时序

29.7.3 设备 ID 地址检测功能

RIIC 具有以 I²C 总线 (Rev.03) 为基准的设备 ID 地址检测功能。在将 ICSER.DIDE 位置 “1” 的状态下, 如果开始条件或者重新开始条件之后的第 1 个字节接收到 “1111 100b”, RIIC 就将此地址视为设备 ID 地址, 当后续的 R/W# 位为 “0” 时, 就在 SCL 时钟的第 9 个时钟的上升沿将 ICSR1.DID 标志置 “1”, 然后将第 2 个字节以后的地址和自己的从属地址进行比较。如果第 2 个字节以后的地址与从属地址寄存器的值匹配, 对应的 ICSR1.AASy 标志 (y=0 ~ 2) 就变为 “1”。

此后, 如果开始条件或者重新开始条件之后的第 1 个字节再次与设备 ID 地址 (1111 100b) 匹配并且后续的 R/W# 位为 “1”, RIIC 就不比较第 2 个字节以后的地址, 而将 ICSR2.TDRE 标志置 “1”。

设备 ID 地址检测功能, 在与自己的从属地址不匹配时或者自己的从属地址匹配后的重新开始条件后面的地址与设备 ID 地址不匹配时, 将 DID 标志置 “0”。在开始条件或者重新开始条件之后的第 1 个字节与设备 ID 地址 (1111 100b) 匹配并且 R/W# 位为 “0” 时, 将 DID 标志置 “1”, 并且将第 2 个字节以后的地址和从属地址进行比较。当 R/W# 位为 “1” 时, DID 标志继续保持原来的值, 并且不比较第 2 个字节以后的从属地址。因此, 通过在确认 TDRE 是 “1” 后检查 DID 标志, 能确认已接收到设备 ID。

在接收一连串的设备 ID 后, 要发送给主机的设备 ID 字段所需的信息 (3 个字节: 厂商 [12 位]+ 部件识别 [9 位]+ 版本 [3 位]) 和正常的发送数据一样, 必须预先准备。有关设备 ID 字段所需信息的详细内容, 请向 NXP 公司询问。

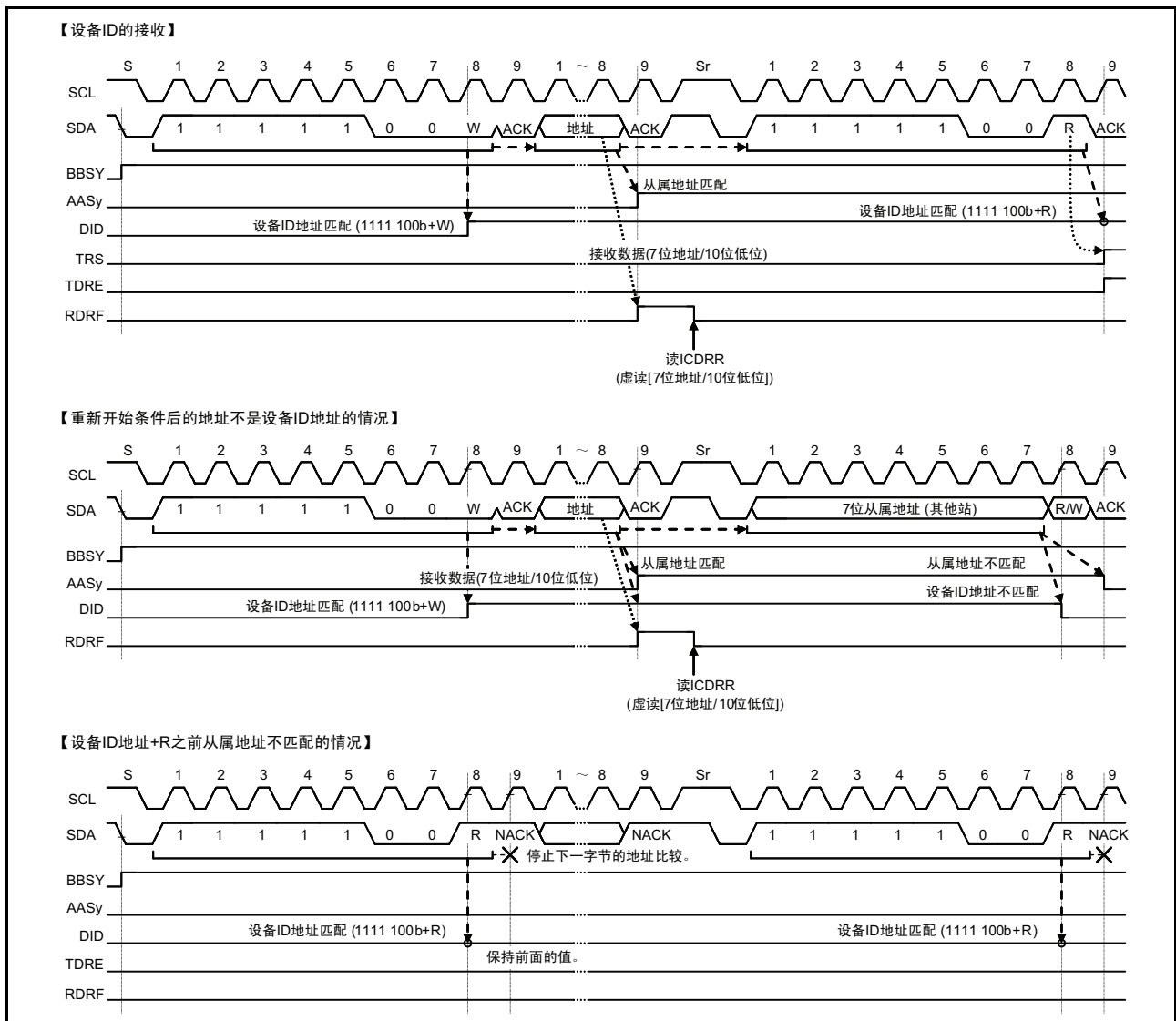


图 29.27 接收设备 ID 地址时的 AASy 标志和 DID 标志的置位 / 清除时序

29.7.4 主机地址检测功能

RIIC 具有 SMBus 运行时的主机地址检测功能。如果在 ICMR3.SMBS 位为“1”时将 IC SER.HOAE 位置“1”，就能在从属接收模式 (ICCR2.MST 位和 ICCR2.TRS 位为“00b”) 中检测主机地址 (0001 000b)。

如果 RIIC 检测到主机地址，就在 SCL 时钟的第 9 个时钟的下降沿将 IC SR1.HOA 标志置“1”，并且在 Wr 位的情况下 (R/W# 位接收到“0”)，将 IC SR2.TDRE 标志置“1”。因此，能产生发送数据空中断 (ICTXI)，并且能通过确认 HOA 标志，视为从智能电池等发送了主机地址。

即使接在主机地址 (0001 000b) 后面的位是 Rd 位 (R/W# 位接收到“1”)，也能检测主机地址。主机地址检测后的运行和普通的从属模式运行相同。

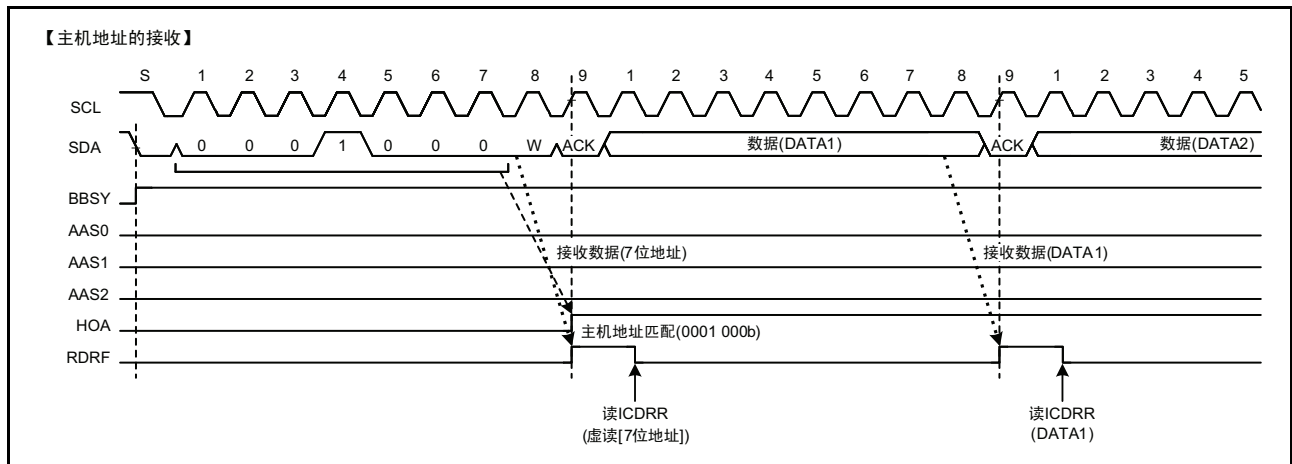


图 29.28 接收主机地址时的 HOA 标志变为“1”的时序

29.8 SCL 的 Low 电平自动保持功能

29.8.1 发送数据的误发送防止功能

在发送模式中 (ICCR2.TRS 位 =1)，如果移位寄存器 (ICDRS 寄存器) 为空状态并且未写发送数据 (ICDRT 寄存器)，RIIC 就在以下所示的区间自动保持 SCL 线的 Low 电平。Low 电平的保持延长了写发送数据前的 Low 电平区间，防止意想不到的数据误发送。

【主控发送模式】

- 发行开始条件或者重新开始条件后的 Low 电平区间
- 第 9 个时钟和下次传送的第 1 个时钟的 Low 电平区间

【从属发送模式】

- 第 9 个时钟和下次传送的第 1 个时钟的 Low 电平区间

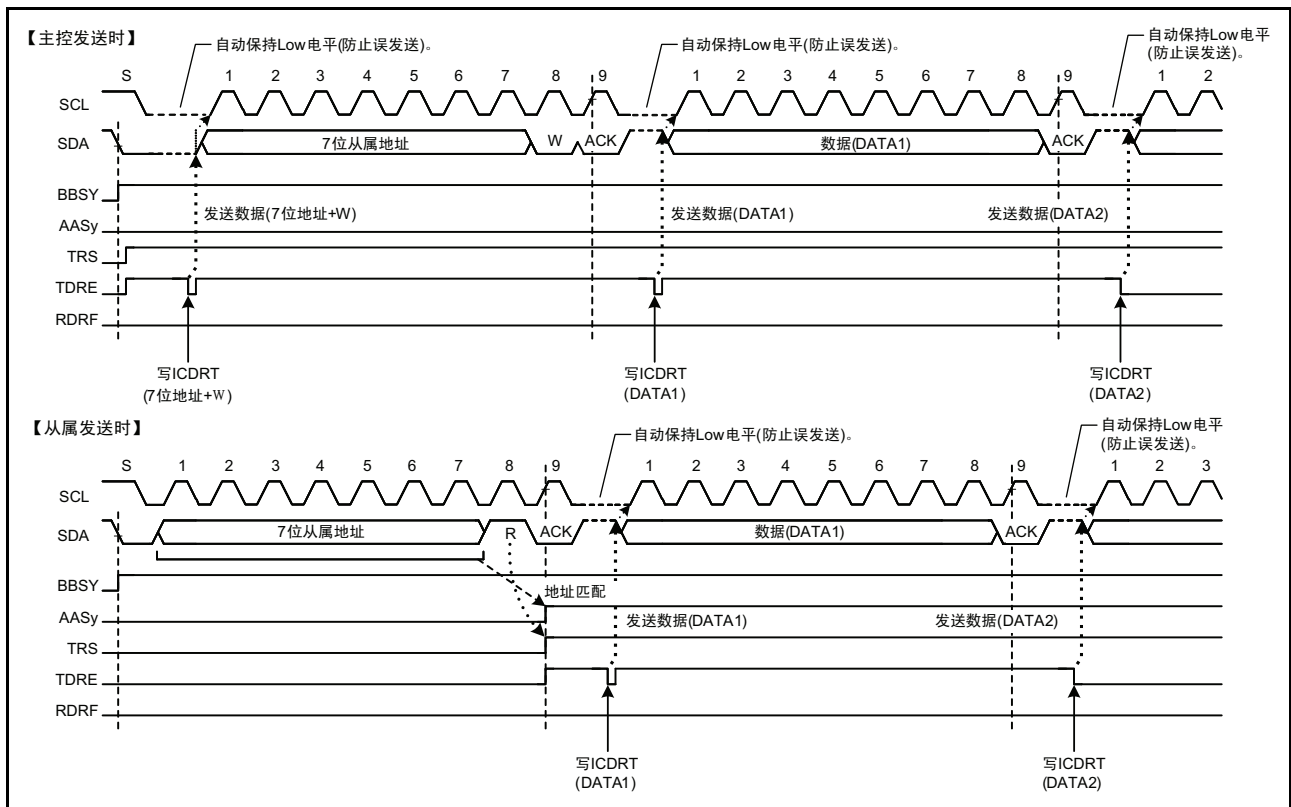


图 29.29 发送模式的 Low 电平自动保持运行

29.8.2 NACK 接收传送中止功能

RIIC 具有在发送模式中 (ICCR2.TRS 位 =1) 接收到 NACK 时中止传送的功能。此功能在 ICFER.NACKE 位为 “1” (允许中止传送) 时有效, 如果在接收 NACK 时已经写了下一个发送数据 (ICSR2.TDRE 标志为 “0” 的状态), 就在 SCL 时钟的第 9 个时钟下降沿时自动中止下一个数据的发送。因此, 能在下一个发送数据的 MSB 为 “0” 时防止 SDA 线被固定为 Low 电平输出。

如果通过 NACK 接收传送中止功能来中止传送 (ICSR2.NACKF 标志 =1), 就不进行以后的发送和接收。要重新开始传送时, 必须将 NACKF 标志置 “0”。在主机发送模式中, 必须将 NACKF 标志置 “0”, 然后通过发行重新开始条件或者停止条件后发行开始条件, 重新开始传送。

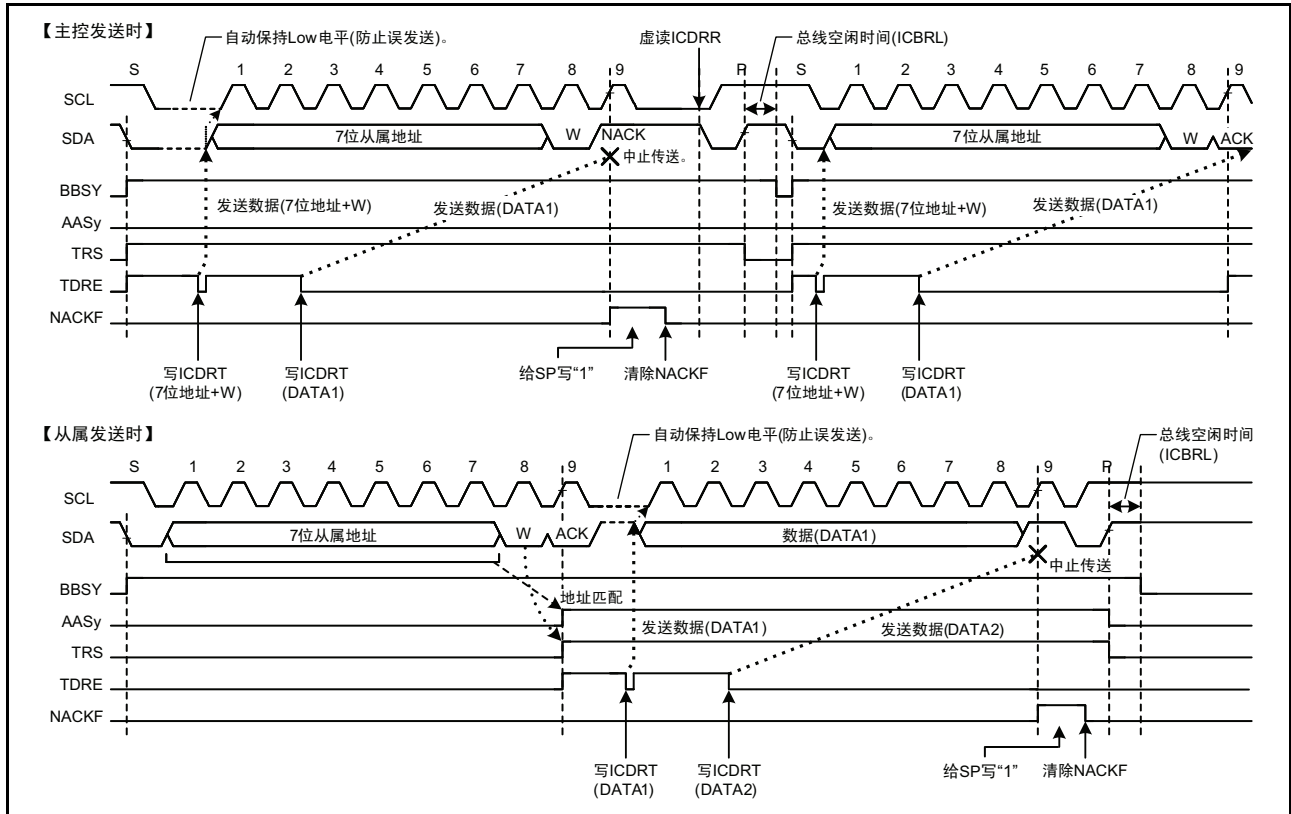


图 29.30 接收 NACK 时的传送中止 (NACKE 位为 “1”)

29.8.3 接收数据漏取防止功能

如果在接收模式中 (ICCR2.TRS 位 =0) 并且接收数据满 (ICSR2.RDRF 标志 =1) 的状态下因至少推迟 1 个传送帧的读接收数据 (ICDRR 寄存器) 等而发生响应处理延迟, RIIC 就在开始下一个数据接收前自动保持 SCL 线的 Low 电平, 防止漏取接收数据。

在最后接收数据的读处理延迟期间, 即使在停止条件后指定了自己的从属地址, 通过自动保持 Low 电平防止漏取接收数据的功能仍然有效。因为在停止条件后自己的从属地址不匹配时不保持 Low 电平, 所以不妨碍其他通信。

RIIC 能通过 ICMR3.WAIT 位和 ICMR3.RDRFS 位的组合选择保持 Low 电平的区间。

(1) 通过 WAIT 位进行的 1 字节接收以及 Low 电平的自动保持功能

如果将 ICMR3.WAIT 位置“1”，RIIC 就通过 WAIT 位的功能进行 1 字节的接收。当 ICMR3.RDRFS 位为“0”时，RIIC 就自动给从 SCL 时钟的第 8 个时钟的下降沿到第 9 个时钟的下降沿期间的应答位发送 ICMR3.ACKBT 位的内容，如果检测到第 9 个时钟的下降沿，就通过 WAIT 位的功能自动保持 SCL 线的 Low 电平。通过读 ICDRR 寄存器来解除保持的 Low 电平。因此，能逐字节地进行接收。

在主控接收模式或者从属接收模式中，WAIT 位的功能从与自己的从属地址（包括全呼地址和主机地址）匹配后的接收帧开始有效。

(2) 通过 RDRFS 位进行的 1 字节接收（控制 ACK/NACK 发送）以及 Low 电平的自动保持功能

如果将 ICMR3.RDRFS 位置“1”，RIIC 就通过 RDRFS 位的功能进行 1 字节的接收。如果将 RDRFS 位置“1”，就将接收数据满标志（ICSR2.RDRF 标志）变为“1”的时序更改为 SCL 时钟的第 8 个时钟的上升沿。如果检测到第 8 个时钟的下降沿，就自动保持 SCL 线的 Low 电平。通过写 ICMR3.ACKBT 位来解除保持的 Low 电平，而在读 ICDRR 寄存器时无法解除。因此，能逐字节地根据接收数据的内容进行 ACK/NACK 发送的接收运行。

在主控接收模式或者从属接收模式中，RDRFS 位的功能从与自己的从属地址（包括全呼地址和主机地址）匹配后的接收帧开始有效。

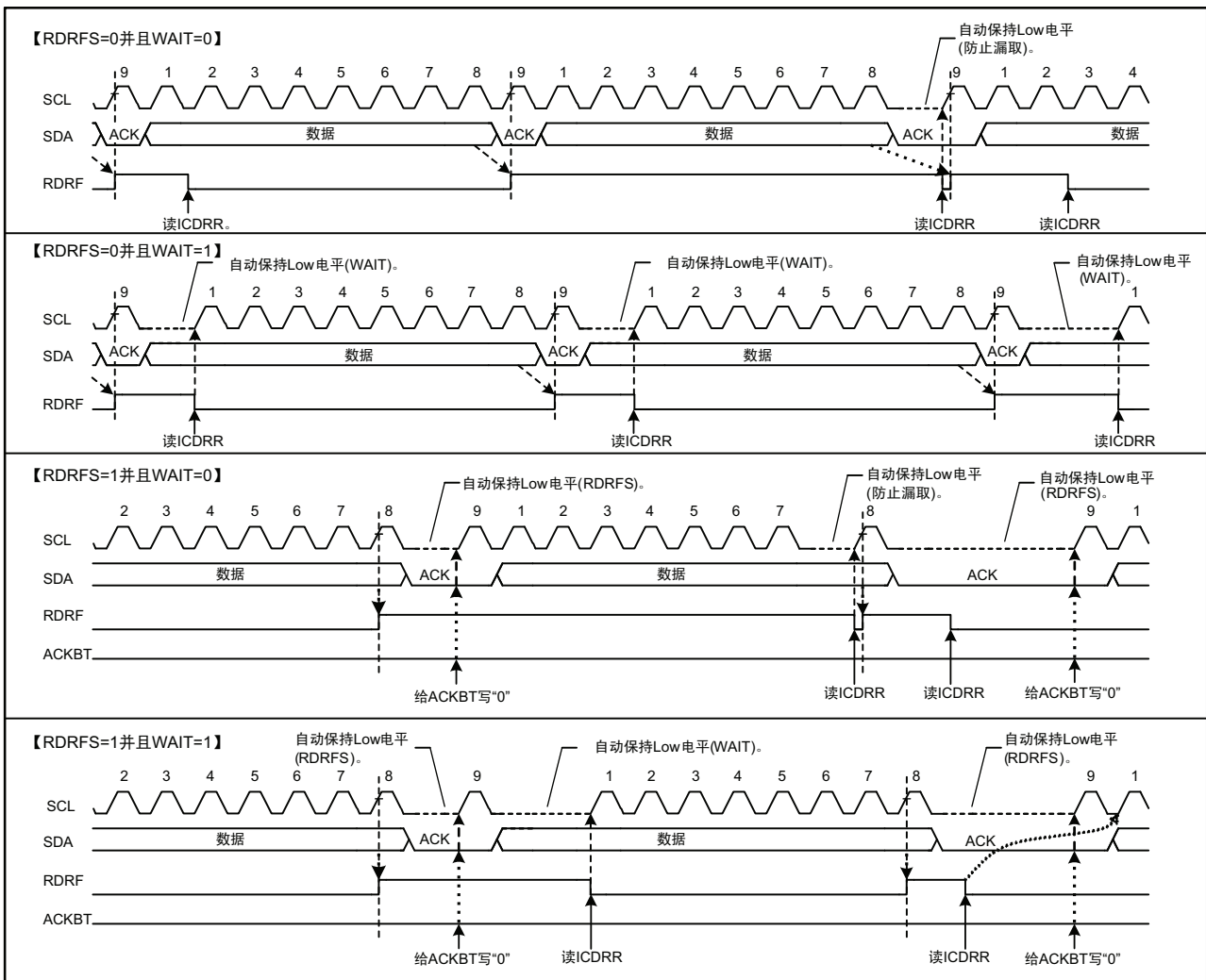


图 29.31 接收模式的 Low 电平自动保持（RDRFS 位和 WAIT 位）

29.9 仲裁失败检测功能

RIIC 除了 I²C 总线规格规定的普通仲裁失败检测功能以外，还具有防止双重发行开始条件功能、NACK 发送时的仲裁失败检测功能以及从属发送时的仲裁失败检测功能。

29.9.1 主控仲裁失败检测功能 (MALE 位)

在发行开始条件时，RIIC 将 SDA 线置为 Low 电平。但是，如果其他主控设备提前发行开始条件后将 SDA 线置为 Low 电平，就产生仲裁失败并且优先其他主控设备的通信。同样地，如果在 ICCR2.BBSY 标志为“1”（总线忙）时将 ICCR2.ST 位置“1”，就产生仲裁失败并且优先其他主控设备的通信。此时，不生成开始条件。

在正常地发行开始条件时，如果包括发送地址在内的发送数据（SDA 信号）和 SDA 线上的信号状态不同（自身发送的 SDA 输出为 High 电平输出（SDA 引脚为高阻抗状态）而检测到 SDA 线为 Low 电平时），就产生仲裁失败。

如果发生主控仲裁失败，RIIC 就立即转移到从属接收模式。此时，如果包括全呼地址在内的从属地址匹配，就继续从属模式的运行。

对于主控仲裁失败检测，在 ICFER.MALE 位为“1”（允许主控仲裁失败检测）的状态下以下条件成立时，检测到仲裁失败。

[主控仲裁失败条件]

- 在 ICCR2.BBSY 标志为“0”的状态下通过将 ICCR2.ST 位置“1”来发行开始条件时，SDA 信号和 SDA 线上的信号状态不同（开始条件发行错误）时
- 在 ICCR2.BBSY 标志为“1”的状态下将 ICCR2.ST 位置“1”（开始条件双重发行错误）时
- 在主控发送模式中（ICCR2.MST 位和 ICCR2.TRS 位为“11b”），应答除外的发送数据（SDA 信号）和 SDA 线上的信号状态不同时

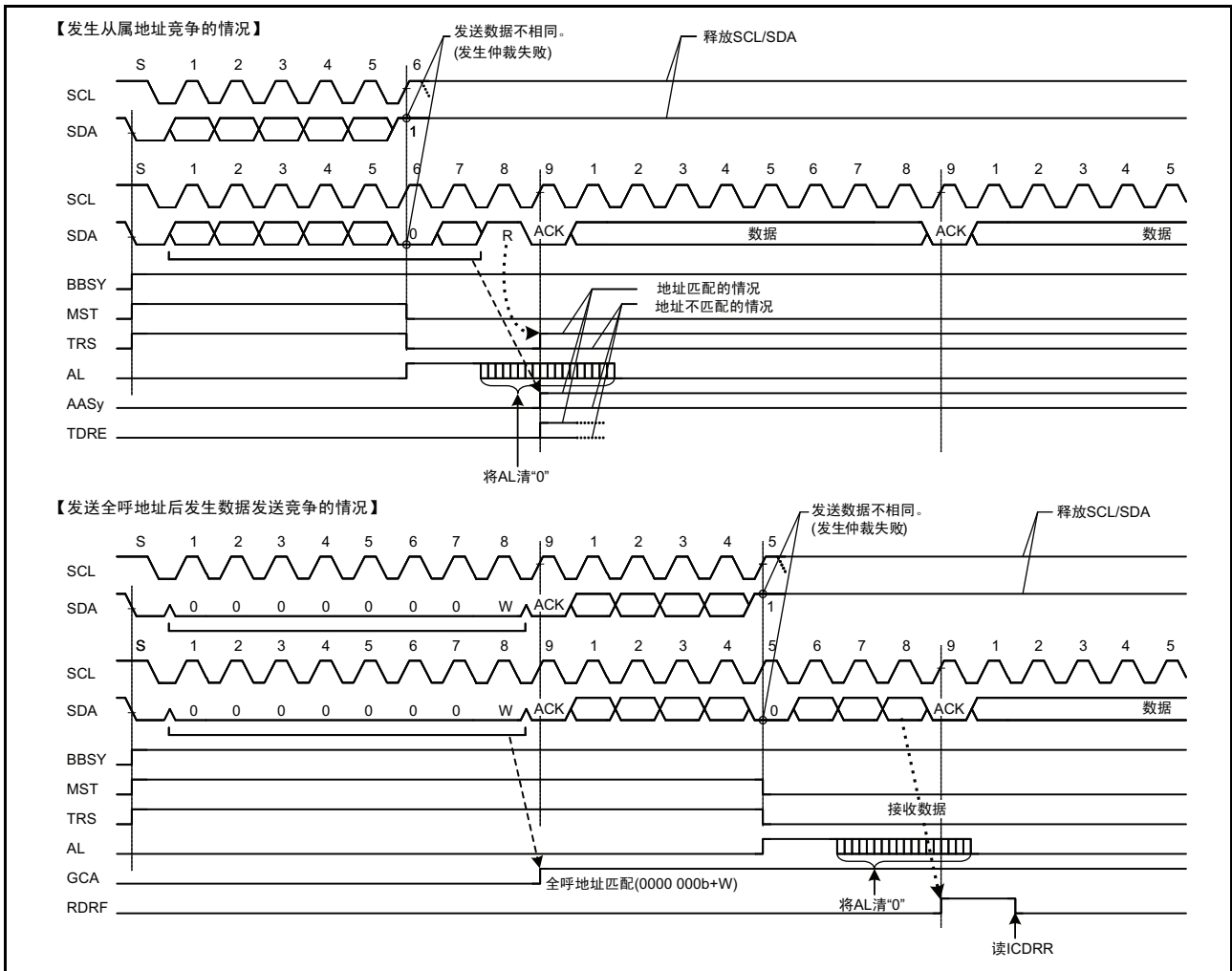


图 29.32 主控仲裁失败检测的运行例子 (MALE 位为“1”的情况)

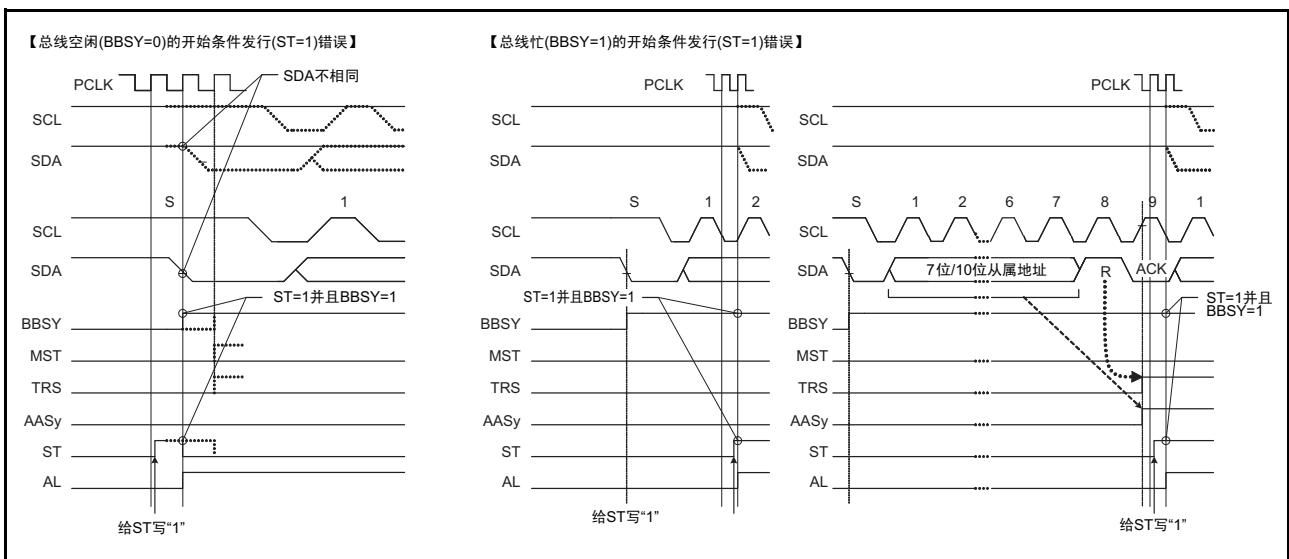


图 29.33 发行开始条件时的仲裁失败 (MALE 位为“1”的情况)

29.9.2 NACK 发送仲裁失败检测功能 (NALE 位)

在接收模式中，如果在发送 NACK 时 RIIC 自身发送的 SDA 信号和 SDA 线上的信号状态不同（自身发送的 SDA 输出为 High 电平输出（SDA 引脚为高阻抗状态）而检测到 SDA 线为 Low 电平时），就产生仲裁失败。此仲裁失败功能主要在多主控系统中 2 个或者 2 个以上的主控设备同时从同一个从属设备接收数据时发生 NACK 发送和 ACK 发送的冲突。这种情况发生在 2 个或者 2 个以上的主控设备通过 1 个从属设备交换共同信息时。NACK 发送仲裁失败检测的运行例子如图 29.34 所示。

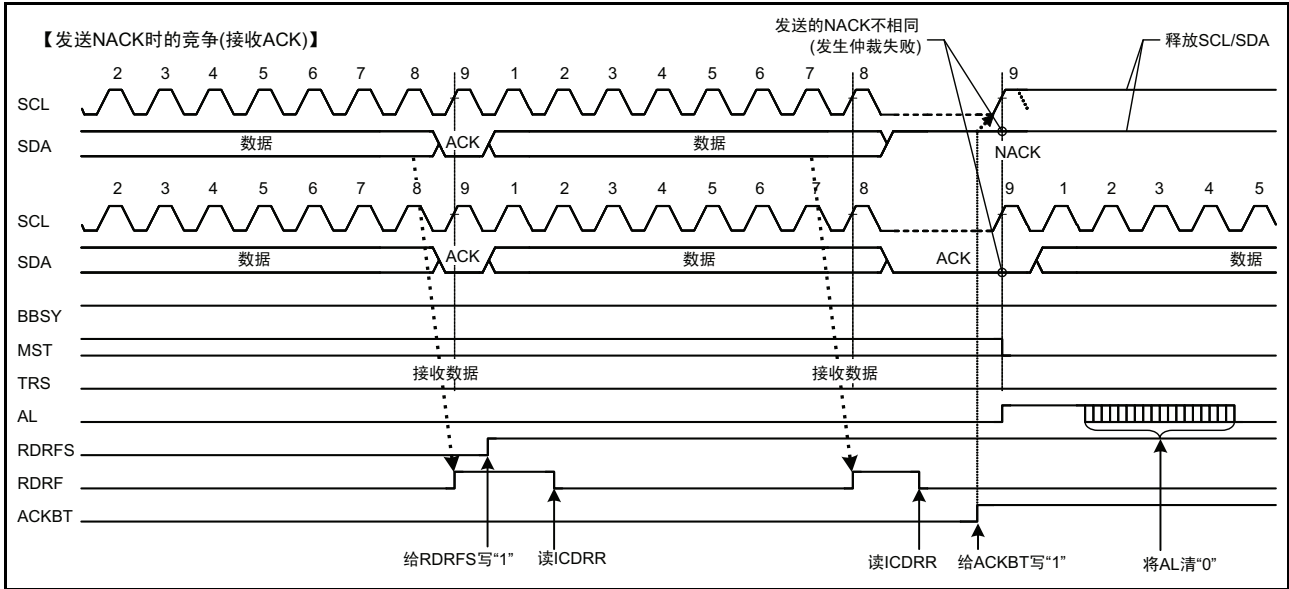


图 29.34 NACK 发送仲裁失败检测的运行例子 (NALE 位为“1”的情况)

以 2 个主控设备（主控 A、主控 B）和 1 个从属设备连接总线的情况为例进行说明。假设主控 A 从从属设备接收 2 字节的数据，主控 B 从从属设备接收 4 字节的数据。

在主控 A 和主控 B 同时存取从属设备的情况下，因为从属地址相同，所以主控 A 和主控 B 在存取从属设备时不发生仲裁失败，因此主控 A 和主控 B 都取得了总线权。然后，主控 A 在从从属设备接收完最后 2 字节数据时发送 NACK，主控 B 因为没有从从属设备接收到需要的 4 字节而发送 ACK，因此主控 A 发送的 NACK 和主控 B 发送的 ACK 发生了冲突。一般情况下，如果发生这种状况，因为主控 A 在无法检测到主控 B 发送的 ACK 的状态下发行停止条件，所以与主控 B 的 SCL 时钟输出发生竞争并且阻碍通信。

RIIC 对于此类发送 NACK 时接收 ACK 的情况，能检测到和其他主控设备的竞争失败，产生仲裁失败。

如果发生 NACK 发送仲裁失败，RIIC 就立即解除从属匹配状态并且转移到从属接收模式。因此，能将停止条件的发行防范于未然并且防止总线的通信故障。

在 SMBus 的 ARP 命令处理中，能省去在与指定地址的 UDID（唯一设备标识）不匹配时的 NACK 发送后以及确定指定地址后的 Get UDID（通用）的 NACK 发送后的剩余处理（FFh 发送处理）。

对于 NACK 发送仲裁失败检测，在 ICFER.NALE 位为“1”（允许 NACK 发送仲裁失败检测）的状态下以下条件成立时，检测到仲裁失败。

[NACK 发送仲裁失败条件]

- 在发送 NACK 时 (ICMR3.ACKBT 位=1)，自身发送的 SDA 信号和 SDA 线上的信号状态不同（接收到 ACK）。

29.9.3 从属仲裁失败检测功能 (SALE 位)

如果在从属发送时发送数据 (自身发送的 SDA 信号) 和 SDA 线上的信号状态不同 (自身发送的 SDA 输出为 High 电平输出 (SDA 引脚为高阻抗状态) 而检测到 SDA 线为 Low 电平), RIIC 就产生仲裁失败。此仲裁失败功能主要用于 SMBus 的 UDID (唯一设备标识符) 的发送。

如果发生从属仲裁失败, RIIC 就解除从属匹配状态并且转移到从属接收模式。

通过使用此功能, 能省去发送 SMBus 的 UDID 时的数据冲突检测和冲突后的剩余处理 (FFh 发送处理)。

对于从属仲裁失败检测, 在 ICFER.SALE 位为 “1” (允许从属仲裁失败检测) 的状态下以下条件成立时, 检测到仲裁失败。

[从属仲裁失败条件]

- 在从属发送模式中 (ICCR2.MST 位和 ICCR2.TRS 位为 “01b”), 应答除外的发送数据 (自身发送的 SDA 信号) 和 SDA 线上的信号状态不同。

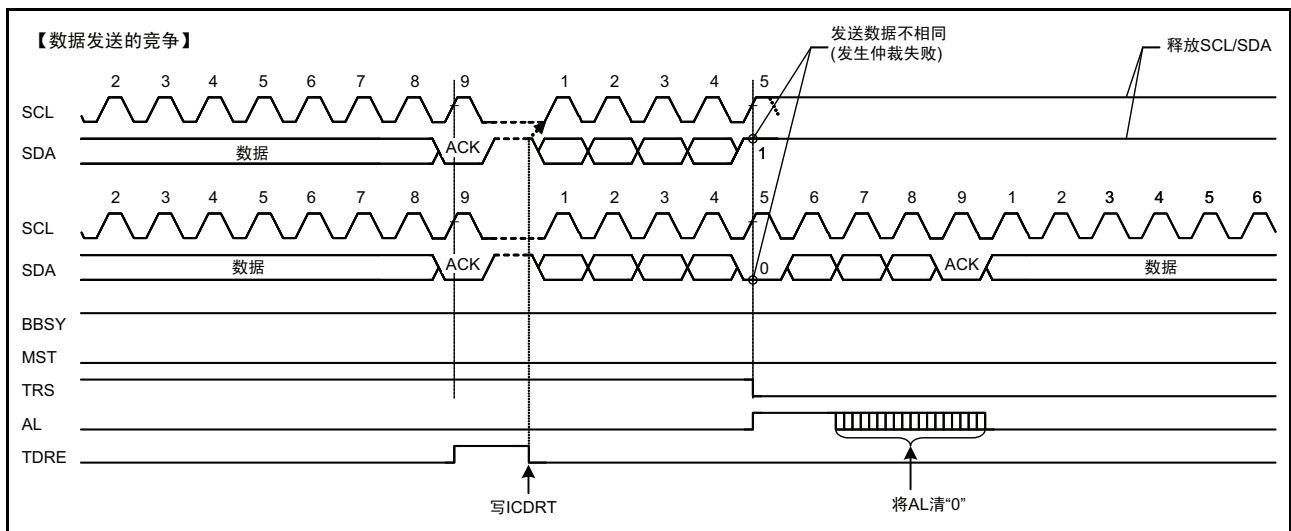


图 29.35 从属仲裁失败检测的运行例子 (SALE 位为 “1” 的情况)

29.10 开始条件、重新开始条件和停止条件的发行功能

29.10.1 开始条件的发行

RIIC 通过 ICCR2.ST 位发行开始条件。

如果将 ST 位置“1”，就请求发行开始条件，在 ICCR2.BBSY 标志为“0”（总线空闲）的状态下发行开始条件。如果正常地发行开始条件，RIIC 就自动转移到主控发送模式。

开始条件的发行步骤如下：

[开始条件的发行]

1. 使 SDA 下降（从 High 电平变为 Low 电平）。
2. 确保 ICBRH 寄存器所设时间的开始条件的保持时间。
3. 使 SCL 线下降（从 High 电平变为 Low 电平）。
4. 在检测到 SCL 线的 Low 电平后，确保 ICBRL 寄存器所设时间的 SCL 线的 Low 电平宽度。

29.10.2 重新开始条件的发行

RIIC 通过 ICCR2.RS 位发行重新开始条件。

如果将 RS 位置“1”，就请求发行重新开始条件，RIIC 在 ICCR2.BBSY 标志为“1”（总线忙）并且 ICCR2.MST 位为“1”（主控模式）的状态下发行重新开始条件。

重新开始条件的发行步骤如下：

[重新开始条件的发行]

1. 释放 SDA 线。
2. 确保 ICBRL 寄存器所设时间的 SCL 线的 Low 电平宽度。
3. 释放 SCL 线（从 Low 电平变为 High 电平）。
4. 在检测到 SCL 线的高电平后，确保 ICBRL 寄存器所设时间的重新开始条件的准备时间。
5. 使 SDA 线下降（从 High 电平变为 Low 电平）。
6. 确保 ICBRH 寄存器所设时间的重新开始条件的保持时间。
7. 使 SCL 线下降（从 High 电平变为 Low 电平）。
8. 在检测到 SCL 线的 Low 电平后，确保 ICBRL 寄存器所设时间的 SCL 线的 Low 电平宽度。

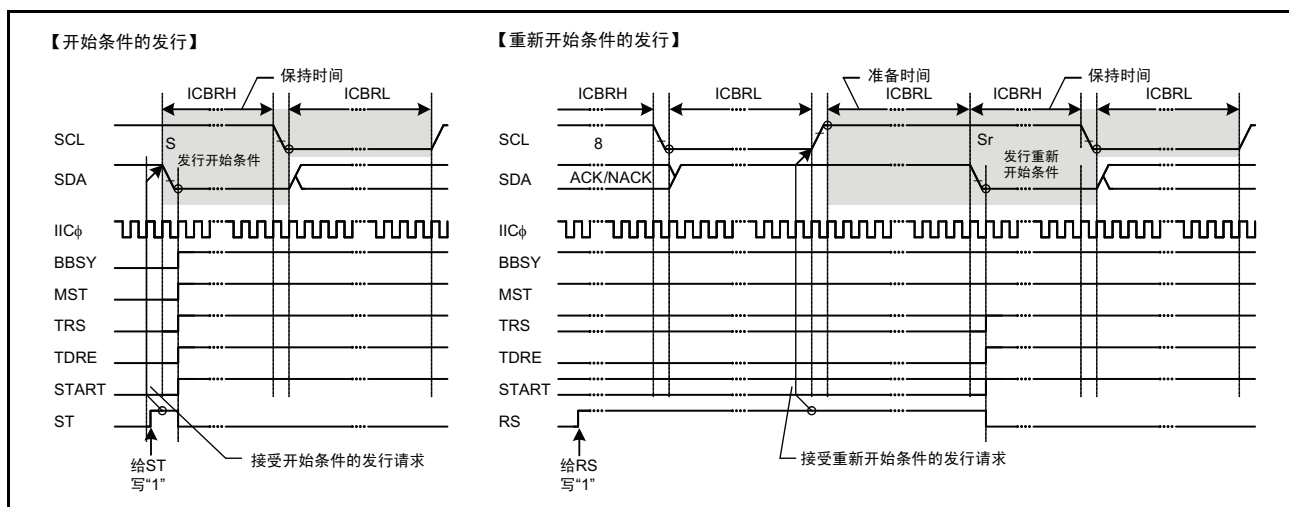


图 29.36 开始条件 / 重新开始条件的发行时序 (ST 位、RS 位)

29.10.3 停止条件的发行

RIIC 通过 ICCR2.SP 位发行停止条件。

如果将 SP 位置“1”，就请求发行停止条件，RIIC 在 ICCR2.BBSY 标志为“1”（总线忙）并且 ICCR2.MST 位为“1”（主控模式）的状态下发行停止条件。

停止条件的发行步骤如下：

[停止条件的发行]

- 使 SDA 线下降（从 High 电平变为 Low 电平）
- 确保 ICBRL 寄存器所设时间的 SCL 线的 Low 电平宽度
- 释放 SCL 线（从 Low 电平变为 High 电平）
- 在检测到 SCL 线的高电平后，确保 ICBRH 寄存器所设时间的停止条件的准备时间
- 释放 SDA 线（从 Low 电平变为 High 电平）
- 确保 ICBRL 寄存器所设时间的总线空闲时间
- 清除 BBSY 标志（释放总线权）

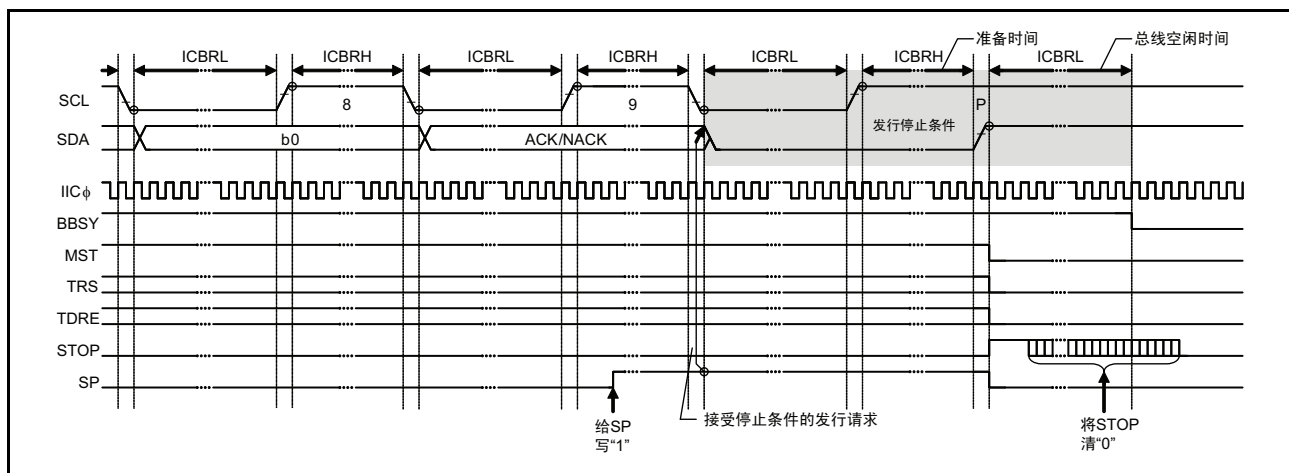


图 29.37 停止条件的发行时序 (SP 位)

29.11 总线意外停机

如果 I²C 总线主要因噪声等的影响而在主控设备和从属设备之间变为不同步，就可能在 SCL 线和 SDA 线被固定的状态下发生总线意外停机。

对于总线意外停机的状态，RIIC 具有超时检测功能（能通过监视 SCL 线来检测总线意外停机状态）、SCL 时钟的追加输出功能（用于解除因不同步引起的总线意外停机状态）和 RIIC/ 内部复位功能。

能通过 ICCR1.SCLO 位、ICCR1.SDAO 位、ICCR1.SCLI 位和 ICCR1.SDAI 位，确认是 RIIC 自身给 SCL 线 /SDA 线输出了 Low 电平还是通信设备输出了 Low 电平。

29.11.1 超时检测功能

RIIC 具有超时检测功能，检测 SCL 线在一定时间后没有变化的状态。在总线忙的状态下，RIIC 能通过检测到 SCL 线的 Low 电平或者 High 电平被固定了一定的时间以上，并且检测到总线的异常状态。

超时检测功能监视 SCL 线的状态，通过内部计数器对 Low 电平或者 High 电平的时间进行计数。如果 SCL 线有变化（上升 / 下降），就对内部计数器进行复位，否则就继续进行计数。如果在 SCL 线没有变化的状态下内部计数器发生上溢，RIIC 就能检测到超时并且通知总线的异常状态。

超时检测功能在 ICFER.TMOE 位为“1”时有效，在主控模式中总线忙时（ICCR2.BBSY 标志为“1”的状态），或者在从属模式中 BBSY 标志为“1”并且自己的从属地址匹配时（ICSR1 寄存器 ≠00h），能检测到 SCL 线被固定为 Low 电平或者 High 电平的总线异常状态。

超时检测功能的内部计数器将 ICMR1.CKS[2:0] 位设定的内部基准时钟（IICφ）作为计数源，在选择长模式时（ICMR2.TMOS 位 =0），用作 16 位计数器；在选择短模式时（TMOS 位 =1），用作 14 位计数器。

对于内部计数器的计数，能通过设定 ICMR2.TMOH 位和 ICMR2.TMOL 位选择是在 SCL 线的 Low 电平还是在 High 电平的状态下进行计数，或者在 Low 电平和 High 电平的状态下都进行计数。如果将 ICMR2.TMOH 位和 ICMR2.TMOL 位都置“0”，就不进行内部计数。

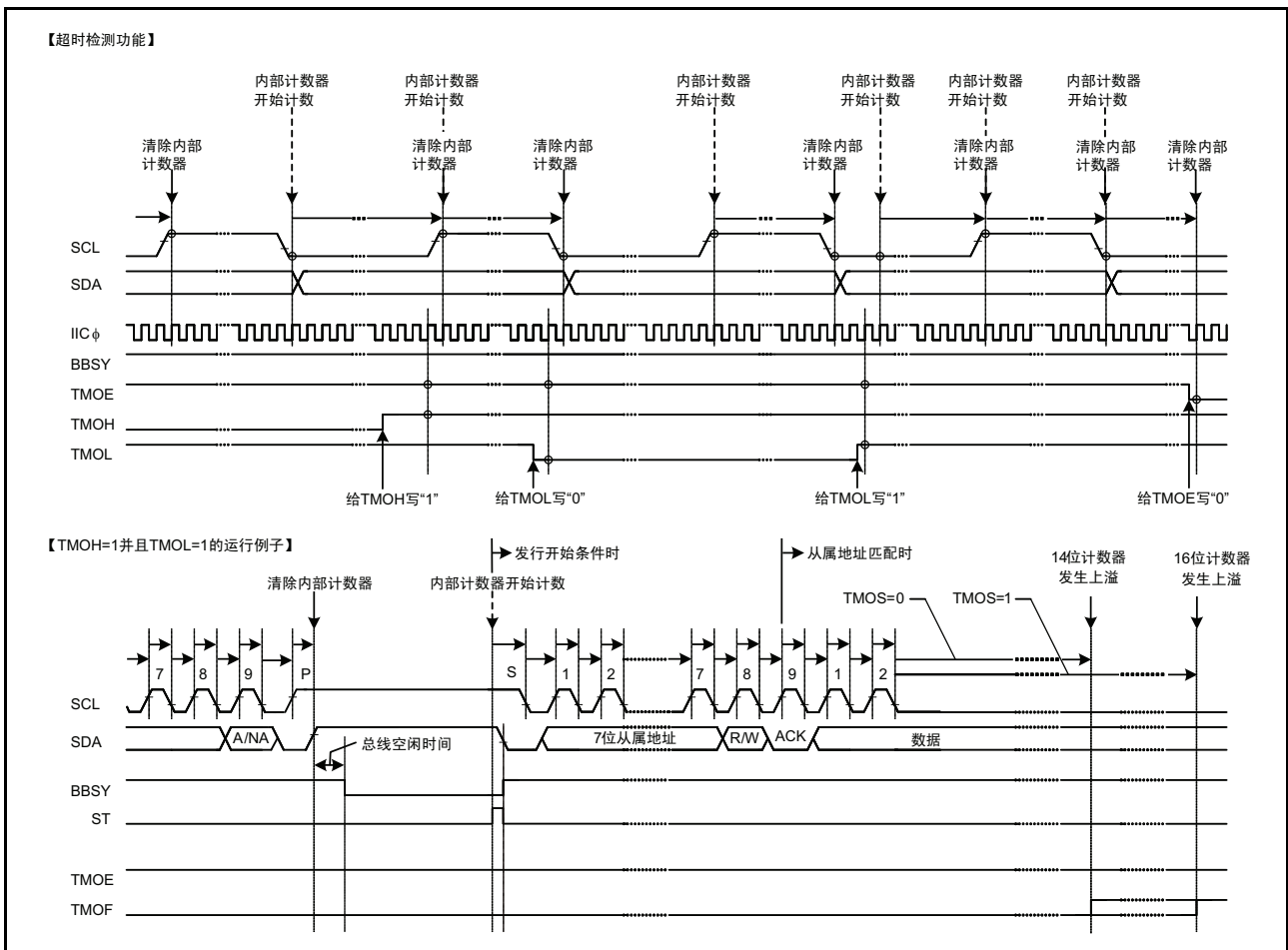


图 29.38 超时检测功能（TMOE 位、TMOS 位、TMOH 位、TMOL 位）

29.11.2 SCL 时钟追加输出功能

RIIC 具有 SCL 时钟的追加输出功能，在主控模式中用于释放因与从属设备不同步而使从属设备的 SDA 线被固定为 Low 电平的状态。

SCL 时钟追加输出功能是以 1 个时钟为单位追加输出 SCL 时钟的功能，主要用于在主控模式中从属设备在 SDA 线固定为 Low 电平的状态而无法发行停止条件时释放从属设备的 SDA 线的固定状态。一般情况下，不使用此功能。如果在正常的通信过程中使用此功能，就可能引起通信异常。

对于 SCL 时钟的追加输出，如果将 ICCR1.CLO 位置“1”，就将 ICMR1.CKS[2:0] 位、ICBRH 寄存器和 ICBRL 寄存器所设传送速度的 SCL 时钟作为 1 个时钟的追加时钟进行输出。一旦输出完 1 个时钟周期的追加时钟，CLO 位就自动变为“0”。因此，能通过软件在确认 CLO 位是“0”后写“1”，连续输出追加时钟。

在可控模式中，因噪声等的影响引起与从属设备不同步，导致从属设备将 SDA 线固定为 Low 电平的状态而无法发行停止条件的总线异常。在这种情况下，RIIC 能通过使用 SCL 时钟追加输出功能逐个输出追加时钟，释放从属设备的 SDA 线 Low 电平的固定状态，恢复总线状态。能通过检查 ICCR1.SDAI 位，确认此从属设备的 SDA 线的释放。为了在确认从属设备的 SDA 线被释放后结束通信，必须重新发行停止条件。

在使用此功能时，必须将 ICFER.MALE 位置“0”（禁止主控仲裁失败检测）。必须注意：当 MALE 位为“1”（允许主控仲裁失败检测）时，在 ICCR1.SDAO 位的值和 SDA 线上的信号状态不同时会产生仲裁失败。

[ICCR1.CLO 位的输出条件]

- 在总线空闲状态（ICCR2.BBSY 标志=0）下或者在可控模式（ICCR2.MST 位为“1”并且 BBSY 标志为“1”的状态）中
- 当通信设备没有保持 SCL 线的 Low 电平状态时

SCL 时钟追加输出功能（CLO 位）如图 29.39 所示。

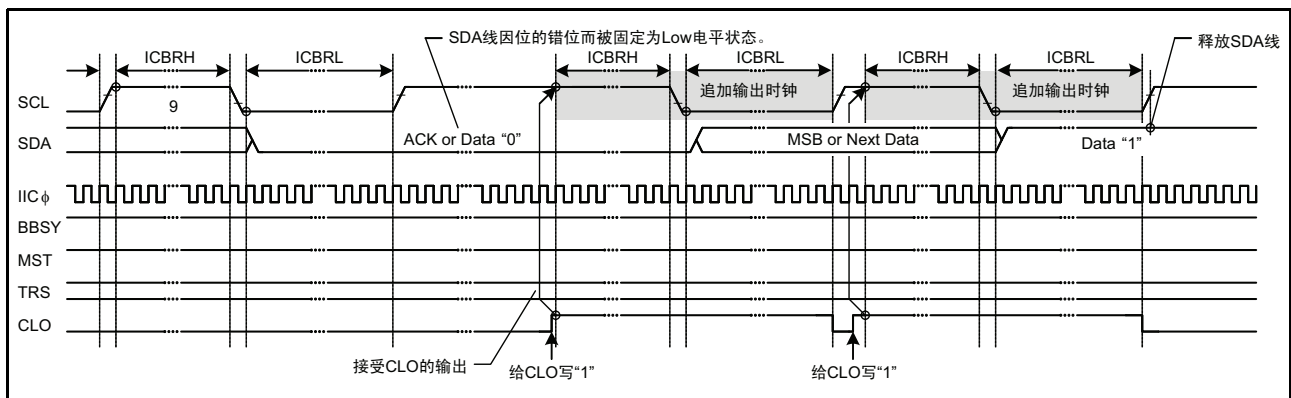


图 29.39 SCL 时钟追加输出功能（CLO 位）

29.11.3 RIIC 复位 / 内部复位

RIIC 具有对 RIIC 模块进行复位的功能。有 2 种复位，1 种是对包括 ICCR2.BBSY 标志在内的全部寄存器进行初始化的 RIIC 复位，另一种是在保持各种设定值的状态下解除从属地址匹配状态以及对内部计数器进行初始化等的内部复位。

复位后，必须将 ICCR1.IICRST 位置“0”。

因为无论进行哪种复位都要解除 SCL 引脚 / SDA 引脚的输出状态而变为高阻抗状态，所以也能用于解除总线意外停机状态。

从属模式中的复位会引起与主控设备的不同步，因此尽量避免使用。必须注意：在 RIIC 复位（ICCR1.ICE 位和 ICCR1.IICRST 位为“01b”）过程中不能监视开始条件等的总线状态。

有关 RIIC 复位 / 内部复位的详细内容，请参照“29.14 复位状况”。

29.12 SMBus 运行

RIIC 能进行以 SMBus (Ver.2.0) 为基准的通信。要进行 SMBus 通信时, 必须将 ICMR3.SMBS 位置“1”。必须通过设定 ICMR1.CKS[2:0] 位、ICBRH 寄存器和 ICBRL 寄存器, 将传送速度设定在 SMBus 规格的 10kbps ~ 100kbps 范围内, 通过设定 ICMR2.DLCS 位和 ICMR2.SDDL[2:0] 位的值, 使数据的保持时间符合 300ns (min) 的规格。在只将 RIIC 用作从属设备时, 不需要设定传送速度, 但是 ICBRL 的设定值至少为数据准备时间 (250ns) 的值。

对于 SMBus 设备的默认地址 (1100 001b), 必须使用从属地址寄存器 L0 ~ L2 (SARL0、SARL1、SARL2) 中的任意 1 个, 并且必须将对应的 SARU_y.FS 位 (y=0 ~ 2) (7 位/10 位地址格式选择位) 置“0” (7 位地址格式)。

在发送 UDID (唯一设备标识符) 时, 必须将 ICFER.SALE 位置“1”, 使从属仲裁失败检测功能有效。

29.12.1 SMBus 超时测量

(1) 从属设备的超时测量

SMBus 通信的从属设备需要测量以下所示的区间 (超时间隔: $T_{\text{LOW:SEXT}}$)。

- 开始条件到停止条件的区间

在通过从属设备进行超时测量时, 使用 RIIC 的开始条件检测中断 (STI) 和停止条件检测中断 (SPI) 并且通过 MTU 定时器或者 TMR 定时器, 测量从检测到开始条件到检测到停止条件的的时间。此超时的测量时间必须在 SMBus 规格的时钟 Low 电平的累积时间 [从属设备] $T_{\text{LOW:SEXT}}$: 25ms (max) 以内。

如果通过 MTU 或者 TMR 测量的时间超过 SMBus 规格的时钟 Low 电平检测的超时 T_{TIMEOUT} : 25ms (min), 从属设备就需要释放总线。从属设备释放总线时, 必须给 ICCR1.IICRST 位写“1”, 进行 RIIC 的内部复位。一旦进行内部复位, RIIC 就能中止 SCL 引脚/SDA 引脚的总线驱动, 并且将引脚置为高阻抗状态, 因此能释放总线。

(2) 主控设备的超时测量

SMBus 通信的主控设备需要测量以下所示的区间 (超时间隔: $T_{\text{LOW:MEXT}}$)。

- 开始条件到应答位的区间
- 应答位到下一个应答位的区间
- 应答位到停止条件的区间

在通过主控设备进行超时测量时, 使用 RIIC 的开始条件检测中断 (STI)、停止条件检测中断 (SPI) 以及发送结束中断 (ICTEI) 或者接收数据满中断 (ICRXI), 通过 MTU 定时器或者 TMR 定时器测量各区间的的时间。此超时的测量时间必须在 SMBus 规格的时钟 Low 电平的累积时间 [主控设备] $T_{\text{LOW:MEXT}}$: 10ms (max) 以内, 开始条件到停止条件的全部 $T_{\text{LOW:MEXT}}$ 的累加结果必须在 $T_{\text{LOW:SEXT}}$: 25ms (max) 以内。

在主控发送模式中 (主控发送器), 需要通过 ICSR2.TEND 标志监视 ACK 接收时序 (SMBCLK 的第 9 个时钟上升沿); 在主控接收模式中 (主控接收器), 需要通过 ICSR2.RDRF 标志监视 ACK 接收时序。因此, 在主控发送时, 进行 1 字节的发送; 在主控接收时, 必须在接收最后字节前将 ICMR3.RDRFS 位置“0”。当 RDRFS 位为“0”时, RDRF 标志在 SMBCLK 的第 9 个时钟的上升沿变为“1”。

如果通过 MTU 或者 TMR 测量的时间超过 SMBus 规格的时钟 Low 电平的累积时间 [主控设备] $T_{\text{LOW:MEXT}}$: 10ms (max), 或者各测量时间的累加结果超过 SMBus 规格的时钟 Low 电平检测的超时 T_{TIMEOUT} : 25ms (min), 主控设备就需要中止处理。在主控发送时, 必须立即中止发送 (写 ICDRT 寄存器)。通过发行停止条件中止主控设备的处理。

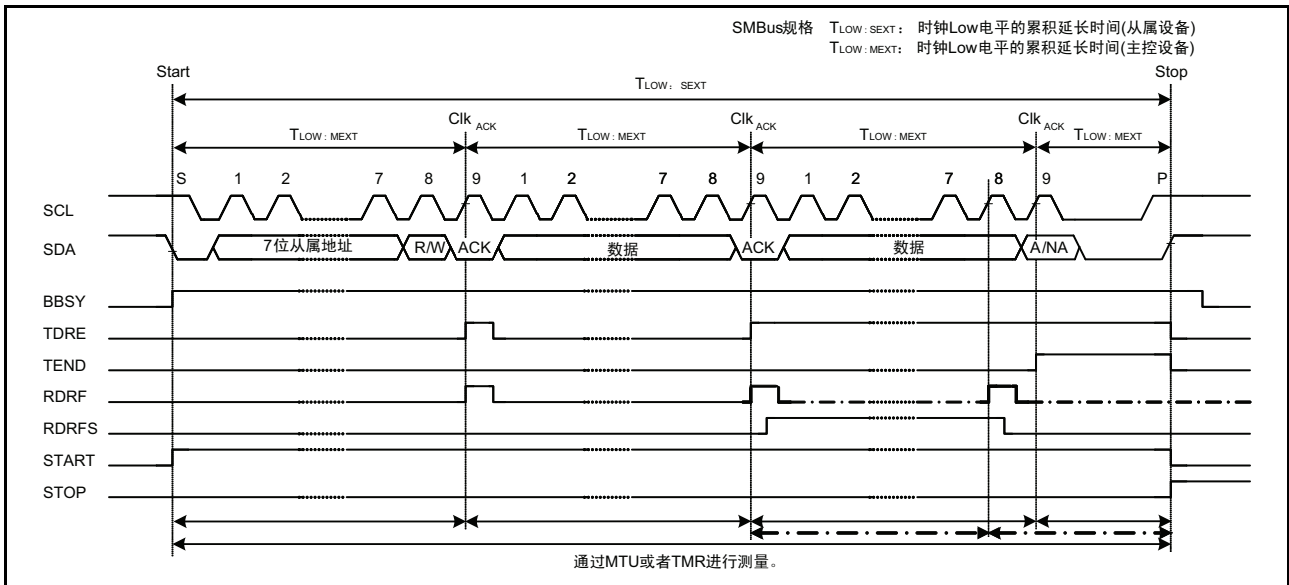


图 29.40 SMBus 超时测量

29.12.2 数据包错误码 (PEC)

RX210 群内置 CRC 运算器。在 RIIC 通信中，能利用 CRC 运算器，发送 SMBus 的数据包错误码 (PEC) 或者检查接收数据。有关 CRC 运算器的多项式，请参照“31. CRC 运算器 (CRC)”。

能通过给 CRC 运算器的 CRC 数据输入寄存器 (CRCDIR) 写全部的发送数据，生成主控发送 (主控发送器) 的 PEC 数据。

通过给 CRC 运算器的 CRCDIR 寄存器写全部的接收数据，得到 CRC 数据输出寄存器 (CRCDOR) 的值，然后将此值和接收的 PEC 数据进行比较，检查主控接收 (主控接收器) 的 PEC 数据。

在检查 PEC 码的过程中接收最后的字节接收时，根据是否匹配发送 ACK 或者 NACK，就必须在接收最后字节的 SMBCLK 的第 8 个时钟上升前将 ICMR3.RDRFS 位置“1”，并且在第 8 个时钟的下降沿将 SCL 线保持为 Low 电平。

29.12.3 SMBus 主机通知协议 /Notify ARP master

在 SMBus 中，从属设备能临时变为主控设备，将自己的从属地址通知给 SMBus 主机 (或者 ARP 主控)，或者向 SMBus 主机 (或者 ARP 主控) 请求自己的从属地址。

将 RX210 群用作 SMBus 主机 (或者 ARP 主控) 时，需要将主从设备发送的主机地址 (0001 000b) 作为从属地址进行检测，RIIC 具有此主机地址的检测功能。如果将主机地址作为从属地址进行检测，就必须将 ICMR3.SMBS 位和 ICSER.HOAE 位置“1”。主机地址检测后的运行和普通的从属模式运行相同。

29.13 中断源

RIIC 的中断源有通信错误 / 事件的发生 (仲裁失败检测、NACK 检测、超时检测、开始条件检测、停止条件检测)、接收数据满、发送数据空、发送结束共 4 种。

中断一览表如表 29.6 所示。能根据接收数据满和发送数据空的中断请求, 在启动 DTC 或者 DMAC 后进行数据传送。

表 29.6 中断源

名称	中断源	中断标志	DTC 的启动	DMAC 的启动	优先级	中断条件
ICEEI	通信错误 / 事件的发生	AL	不能	不能	高 ↑	AL=1 并且 ALIE=1
		NACKF				NACKF=1 并且 NAKIE=1
		TMOF				TMOF=1 并且 TMOIE=1
		START				START=1 并且 STIE=1
		STOP				STOP=1 并且 SPIE=1
ICRXI	接收数据满	—	能	能	RDRF=1 并且 RIE=1	
ICTXI	发送数据空	—	能	能	TDRE=1 并且 TIE=1	
ICTEI	发送结束	TEND	不能	不能	低	TEND=1 并且 TEIE=1

必须在中断处理中清除或者屏蔽各自的标志。

【中断处理时的注意事项】

1. CPU 执行写外围模块的指令和实际写模块的时序有延迟。如果清除或者屏蔽中断标志, 就必须重新读中断标志, 并且在确认清除或者写屏蔽位结束后从中断处理返回。如果不确认已写完模块而从中断处理返回, 就可能再次产生相同的中断。
2. 因为 ICTXI 中断为边沿中断, 所以不需要清除。通过给 ICDRT 寄存器写发送数据或者检测到停止条件 (ICSR2.STOP 标志 =1), 作为 ICTXI 中断条件的 ICSR2.TDRE 标志自动变为“0”。
3. 因为 ICRXI 中断为边沿中断, 所以不需要清除。通过读 ICDRR 寄存器, 作为 ICRXI 中断条件的 ICSR2.RDRF 标志自动变为“0”。
4. 在使用 ICTEI 中断时, 必须在 ICTEI 中断处理中清除 ICSR2.TEND 标志。通过给 ICDRT 寄存器写发送数据或者检测到停止条件 (ICSR2.STOP 标志 =1), ICSR2.TEND 标志自动变为“0”。

29.13.1 ICTXI 中断和 ICRXI 中断的缓冲运行

对于 ICTXI 中断和 ICRXI 中断, 即使在 ICUIRn.IR 标志为“1”时满足中断产生条件, 也不将中断请求输出到 ICU 而保持在内部 (在内部能保持的容量是每个中断源 1 个请求)。

如果 ICUIRn.IR 标志变为“0”, 就将保持的中断请求输出到 ICU。在一般使用的状态下, 自动清除保持在内部的中断请求。

也能通过将对应的外围中断允许位置“0”来清除保持在内部的中断请求。

29.14 复位状况

RIIC 具有芯片复位、RIIC 复位和内部复位的复位功能，各种复位的范围和状况如表 29.7 所示。

表 29.7 复位状况

		芯片复位	RIIC 复位 (ICE 位 =0 并且 IICRST 位 =1)	内部复位 (ICE 位 =1 并且 IICRST 位 =1)	开始条件 / 重新开始 条件的检测	停止条件的检测	
ICCR1	ICE、 IICRST	复位	保持	保持	运行 (保持)	运行 (保持)	
	SCLO、 SDAO		复位	复位			
	其他			保持			
ICCR2	BBSY	复位	复位	运行	运行	运行	
	ST			复位	复位	运行 (保持)	
	其他					复位	
ICMR1	BC[2:0]	复位	复位	复位	复位	运行 (保持)	
	其他			保持	运行 (保持)		
ICMR2		复位	复位	保持	运行 (保持)	运行 (保持)	
ICMR3		复位	复位	保持	运行 (保持)	运行 (保持)	
ICFER		复位	复位	保持	运行 (保持)	运行 (保持)	
ICSER		复位	复位	保持	运行 (保持)	运行 (保持)	
ICIER		复位	复位	保持	运行 (保持)	运行 (保持)	
ICSR1		复位	复位	复位	运行 (保持)	复位	
ICSR2	TDRE、 TEND	复位	复位	复位	运行 (保持)	复位	
	START				运行		
	STOP				运行 (保持)		运行
	其他						运行 (保持)
SARL0、SARL1、 SARL2 SARU0、SARU1、 SARU2		复位	复位	保持	运行 (保持)	运行 (保持)	
ICBRH、ICBRL		复位	复位	保持	运行 (保持)	运行 (保持)	
ICDRT		复位	复位	保持	运行 (保持)	运行 (保持)	
ICDRR		复位	复位	保持	运行 (保持)	运行 (保持)	
ICDRS		复位	复位	复位	运行 (保持)	运行 (保持)	
超时检测功能		复位	复位	运行	运行	运行	
总线空闲时间的测量		复位	复位	运行	运行	运行	

29.15 事件链接输出功能

RIIC 具有对事件链接控制器 (ELC) 输出以下事件的功能。

1. 发生通信错误/事件的输出
如果发生通信错误/事件, 就作为事件信号经由 ELC 输出到其他模块。
2. 接收数据满的输出
如果接收数据满, 就作为事件信号经由 ELC 输出到其他模块。
3. 发送数据空的输出
如果发送数据空, 就作为事件信号经由 ELC 输出到其他模块。
4. 发送结束的输出
如果发送结束, 就作为事件信号经由 ELC 输出到其他模块。

29.15.1 中断处理和事件链接的关系

RIIC 的中断有 4 种, 分别是发生通信错误 / 事件 (仲裁失败检测、NACK 检测、超时检测、开始条件检测、停止条件检测)、接收数据满、发送数据空和发送结束。这些中断分别具有控制允许或者禁止中断的中断允许位。如果产生中断源, 就在中断允许位为允许的情况下对 CPU 输出中断请求信号。

与此相对, 事件链接输出信号与中断允许位的设定无关, 一旦产生中断源, 就作为事件信号经由 ELC 输出到其他模块。

有关中断源, 请参照“表 29.6”。

29.16 使用时的注意事项

29.16.1 模块停止功能的设定

能通过模块停止控制寄存器 B (MSTPCRB) 转移到模块停止状态或者解除模块停止状态, 初始值为 RIIC 处于模块停止状态。通过解除模块停止状态, 使 RIIC 的寄存器变为可存取的状态。

有关模块停止控制寄存器 B 的详细内容, 请参照“11. 低功耗功能”。

29.16.2 有关开始通信的注意事项

如果在开始通信 (ICCR1.ICE 位 =1) 时 ICU.IRn.IR 标志为“1”, 就必须在允许运行前按以下步骤清除中断请求。如果在 ICU.IRn.IR 标志为“1”的状态下开始通信 (ICCR1.ICE 位 =1), 开始通信后的中断请求就被保持在内部, 因此 ICU.IRn.IR 标志可能会发生意想不到的变化。

1. 确认 ICCR1.ICE 位是“0”。
2. 将对应的外围中断允许位 (ICIER.TIE 等) 置“0”。
3. 读对应的外围中断允许位 (ICIER.TIE 等), 确认是“0”。
4. 将 ICU.IRn.IR 标志置“0”。

30. 串行外围接口 (RSPI)

30.1 概要

RX210 群内置独立的 1 个通道的串行外围接口 (RSPI)。

RSPI 能进行全双工同步串行通信，还内置和多个处理器、外围设备进行高速通信的功能。

RSPI 的规格和框图分别如表 30.1 和图 30.1 所示。

在本章中，RSPI 命令寄存器 m (SPCMDm) 使用的 m 表示 0 ~ 7。

表 30.1 RSPI 的规格

项目	内容
通道数	1 个通道
RSPI 传送功能	<ul style="list-style-type: none"> 能使用 MOSI (Master Out Slave In)、MISO (Master In Slave Out)、SSL (Slave Select) 和 RSPCK (RSPI Clock) 信号，通过 SPI 运行 (4 线式) 或者时钟同步运行 (3 线式) 进行串行通信。 只能进行发送。 能在主控模式或者从属模式中进行串行通信。 能更改串行传送时钟的极性。 能更改串行传送时钟的相位。
数据格式	<ul style="list-style-type: none"> 能选择 MSB first 或者 LSB first。 能将传送位长改为 8 位、9 位、10 位、11 位、12 位、13 位、14 位、15 位、16 位、20 位、24 位或者 32 位。 发送 / 接收缓冲器为 128 位。 1 次发送 / 接收最多传送 4 帧 (1 帧最多 32 位)。
缓冲器结构	发送 / 接收缓冲器的结构为双缓冲结构。
错误检测	<ul style="list-style-type: none"> 模式故障错误检测 溢出错误检测 奇偶校验错误检测
SSL 控制功能	<ul style="list-style-type: none"> 每个通道有 4 个 SSL 信号 (SSLA0 ~ SSLA3)。 在设定为单主控模式时，输出 SSLA0 ~ SSLA3 信号。 在设定为多主控模式时，SSLA0 信号为输入信号，SSLA1 ~ SSLA3 信号为输出信号或者不使用。 在设定为从属模式时，SSLA0 信号为输入信号，不使用 SSLA1 ~ SSLA3 信号。 能设定从 SSL 输出有效开始到 RSPCK 运行为止的延迟 (RSPCK 延迟)。 设定范围: 1 ~ 8 个 RSPCK 设定单位: 1 个 RSPCK 能设定从 RSPCK 停止开始到 SSL 输出无效为止的延迟 (SSL 无效延迟)。 设定范围: 1 ~ 8 个 RSPCK 设定单位: 1 个 RSPCK 能设定下次存取的 SSL 输出有效的等待 (下次存取延迟)。 设定范围: 1 ~ 8 个 RSPCK 设定单位: 1 个 RSPCK SSL 极性变更功能
主控传送时的控制方式	<ul style="list-style-type: none"> 能按顺序循环执行由最多 8 个命令构成的传送。 能给各命令设定以下项目: SSL 信号值、位速率、RSPCK 极性 / 相位、传送数据长度、LSB/MSB first、突发、RSPCK 延迟、SSL 无效延迟、下次存取延迟 能通过写发送缓冲器来启动传送。 能设定 SSL 无效时的 MOSI 信号值。
中断源	<ul style="list-style-type: none"> 可屏蔽的中断源 RSPI 接收中断 (接收缓冲器满) RSPI 发送中断 (发送缓冲器空) RSPI 错误中断 (模式故障错误、溢出错误和奇偶校验错误) RSPI 空闲中断 (RSPI 空闲)
事件链接功能 (仅支持 RSPIO)	<p>能将 5 种事件输出到事件链接控制器</p> <ul style="list-style-type: none"> 接收缓冲器满事件的输出 发送缓冲器空事件的输出 模式故障 / 溢出 / 奇偶校验错误事件的输出 RSPI 空闲事件的输出 发送结束事件的输出
其他功能	<ul style="list-style-type: none"> CMOS / 漏极开路输出的转换功能 RSPI 初始化功能 环回模式功能
低功耗功能	能设定为模块停止状态。

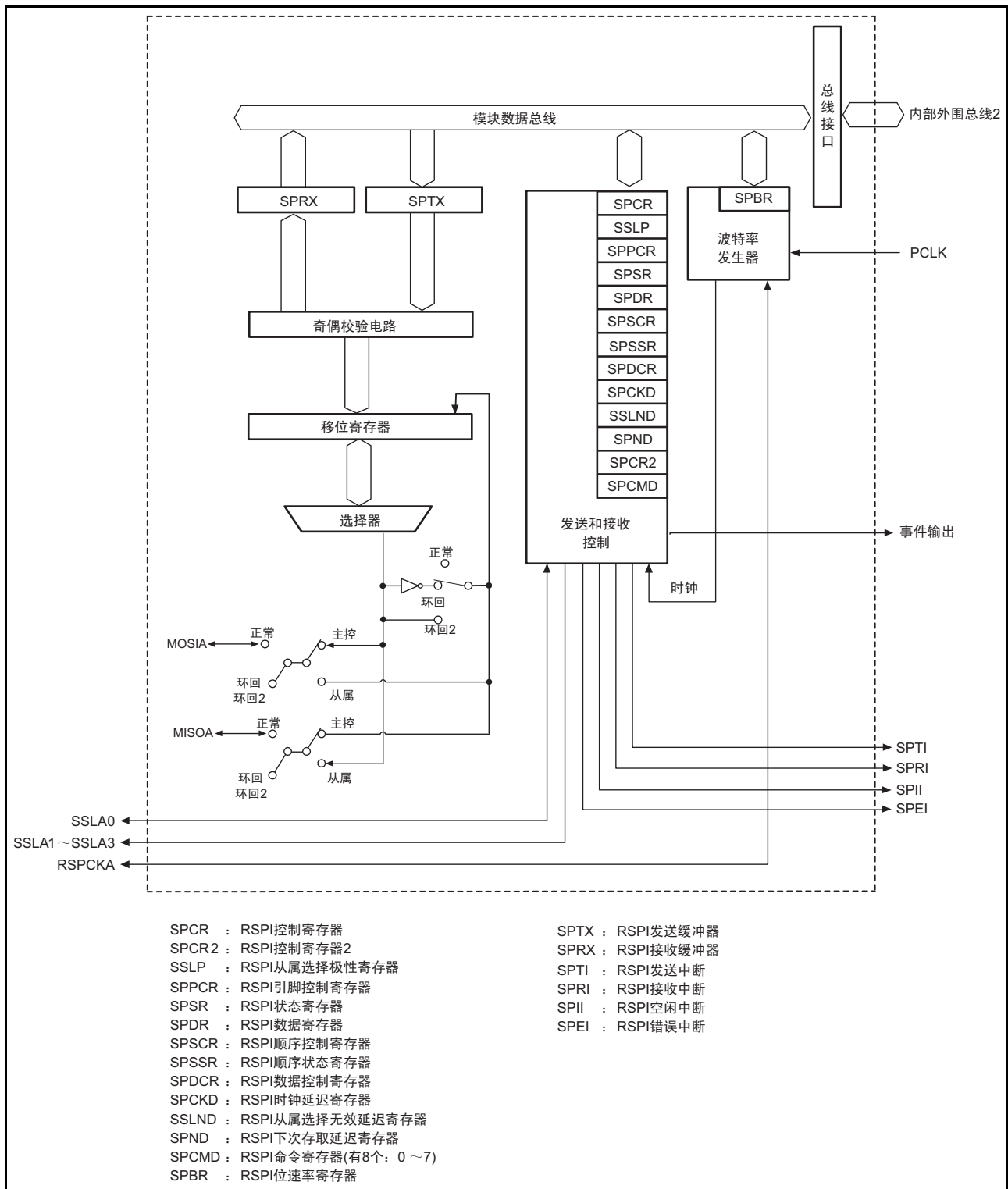


图 30.1 RSPI 的框图

RSPI 使用的输入 / 输出引脚如表 30.2 所示。

在设定为单主控模式时，RSPI 自动将 SSLA0 引脚的输入 / 输出方向更改为输出；在设定为多主控模式和从属模式时，RSPI 自动将 SSLA0 引脚的输入 / 输出方向更改为输入。根据主控模式 / 从属模式的设定和 SSLA0 引脚的输入电平，RSPI 自动更改 RSPCKA、MOSIA 和 MISOA 引脚的输入 / 输出方向。

详细内容请参照“30.3.2 RSPI 引脚的控制”。

表 30.2 RSPI 的输入 / 输出引脚

通道	引脚名	输入 / 输出	功能
RSPI0	RSPCKA	输入 / 输出	时钟输入 / 输出引脚
	MOSIA	输入 / 输出	主控发送数据输入 / 输出引脚
	MISOA	输入 / 输出	从属发送数据输入 / 输出引脚
	SSLA0	输入 / 输出	从属选择输入 / 输出引脚
	SSLA1	输出	从属选择输出引脚
	SSLA2	输出	从属选择输出引脚
	SSLA3	输出	从属选择输出引脚

30.2 寄存器说明

30.2.1 RSPI 控制寄存器 (SPCR)

地址 RSPi0.SPCR 0008 8380h

	b7	b6	b5	b4	b3	b2	b1	b0
	SPRIE	SPE	SPTIE	SPEIE	MSTR	MODFEN	TXMD	SPMS
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	SPMS	RSPI 模式选择位	0: SPI 运行 (4 线式) 1: 时钟同步运行 (3 线式)	R/W
b1	TXMD	通信运行模式选择位	0: 全双工同步串行通信 1: 只进行发送运行的串行通信	R/W
b2	MODFEN	模式故障错误检测允许位	0: 禁止模式故障错误的检测 1: 允许模式故障错误的检测	R/W
b3	MSTR	RSPI 主控 / 从属模式选择位	0: 从属模式 1: 主控模式	R/W
b4	SPEIE	RSPI 错误中断允许位	0: 禁止 RSPI 错误中断请求的产生 1: 允许 RSPI 错误中断请求的产生	R/W
b5	SPTIE	RSPI 发送中断允许位	0: 禁止 RSPI 发送中断请求的产生 1: 允许 RSPI 发送中断请求的产生	R/W
b6	SPE	RSPI 功能允许位	0: RSPI 功能无效 1: RSPI 功能有效	R/W
b7	SPRIE	RSPI 接收中断允许位	0: 禁止 RSPI 接收中断请求的产生 1: 允许 RSPI 接收中断请求的产生	R/W

如果在 SPCR.SPE 位为“1”的状态下更改 SPCR.MSTR 位、SPCR.MODFEN 位和 SPCR.TXMD 位的设定值，就不保证以后的运行。

SPMS 位 (RSPI 模式选择位)

此位选择 SPI 运行 (4 线式) 或者时钟同步运行 (3 线式)。

在进行时钟同步运行时，不使用 SSLA0 ~ 3 引脚而使用 RSPCKA 引脚、MOSIA 引脚和 MISOA 引脚这 3 个引脚进行通信。在 主控模式中 (SPCR.MSTR=1) 进行时钟同步运行时，能将 SPCMDm.CPHA 位置“0”或者“1”；在从属模式中 (SPCR.MSTR=0) 进行时钟同步运行时，就不保证将 CPHA 位置“0”时的运行。

TXMD 位 (发送运行模式选择位)

此位选择全双工同步串行通信或者只进行发送运行的串行通信。

在将 TXMD 位置“1”后进行通信时，只进行发送运行而不进行接收运行 (参照“30.3.6 通信运行模式”)。

如果将 TXMD 置“1”，就不能使用接收缓冲器满的中断请求。

MODFEN 位 (模式故障错误检测允许位)

此位允许或者禁止模式故障错误检测 (参照“30.3.8 错误检测”)。RSPI 根据 MODFEN 位和 MSTR 位的组合，决定 SSLA0 ~ 3 引脚的输入 / 输出方向 (参照“30.3.2 RSPI 引脚的控制”)。

MSTR 位 (RSPI 主控 / 从属模式选择位)

此位选择 RSPI 的主控模式或者从属模式。RSPI 根据 MSTR 位的设定, 决定 RSPCKA、MOSIA、MISOA 和 SSLA0 ~ 3 引脚的方向。

SPEIE 位 (RSPI 错误中断允许位)

如果在 RSPI 检测到模式故障错误后将 SPSR.MODF 标志置“1”, 或者在 RSPI 检测到溢出错误后将 SPSR.OVRF 标志置“1”, 或者在 RSPI 检测到奇偶校验错误后将 SPSR.PERF 标志置“1”, 此位就允许或者禁止 RSPI 错误中断请求的产生。详细内容请参照“30.3.8 错误检测”。

SPTIE 位 (RSPI 发送中断允许位)

此位允许或者禁止产生 RSPI 的发送中断请求。

SPE 位 (RSPI 功能允许位)

此位选择 RSPI 功能的有效或者无效。

当 SPSR.MODF 位为“1”时, 不能将 SPE 位置“1”。详细内容请参照“30.3.8 错误检测”。

如果将 SPE 位置“0”, 就将 RSPI 功能置为无效并且对一部分模块功能进行初始化。详细内容请参照“30.3.9 RSPI 的初始化”。通过将 SPE 位从“0”的状态变为“1”的状态或者从“1”的状态变为“0”的状态, 产生 RSPI 发送中断请求。

SPRIE 位 (RSPI 接收中断允许位)

此位允许或者禁止产生 RSPI 的接收中断请求。

30.2.2 RSPI 从属选择极性寄存器 (SSLP)

地址 RSPi0.SSLP 0008 8381h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	SSL3P	SSL2P	SSL1P	SSL0P
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	SSL0P	SSL0 信号极性设定位	0: SSL0 信号的低电平有效 1: SSL0 信号的高电平有效	R/W
b1	SSL1P	SSL1 信号极性设定位	0: SSL1 信号的低电平有效 1: SSL1 信号的高电平有效	R/W
b2	SSL2P	SSL2 信号极性设定位	0: SSL2 信号的低电平有效 1: SSL2 信号的高电平有效	R/W
b3	SSL3P	SSL3 信号极性设定位	0: SSL3 信号的低电平有效 1: SSL3 信号的高电平有效	R/W
b7-b4	—	保留位	读写值都为“0”。	R/W

如果在 SPCR.SPE 位为“1”的状态下改写 SSLP 寄存器, 就不保证以后的运行。

30.2.3 RSPI 引脚控制寄存器 (SPPCR)

地址 RSPI0.SPPCR 0008 8382h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	MOIFE	MOIFV	—	—	SPLP2	SPLP
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	SPLP	RSPI 环回位	0: 正常模式 1: 环回模式 (发送数据的反相 = 接收数据)	R/W
b1	SPLP2	RSPI 环回 2 位	0: 正常模式 1: 环回模式 (发送数据 = 接收数据)	R/W
b3-b2	—	保留位	读写值都为“0”。	R/W
b4	MOIFV	MOSI 空闲固定值位	0: MOSI 空闲固定值为“0” 1: MOSI 空闲固定值为“1”	R/W
b5	MOIFE	MOSI 空闲值固定允许位	0: MOSI 输出值为上次传送的最后数据 1: MOSI 输出值为 MOIFV 位的设定值	R/W
b7-b6	—	保留位	读写值都为“0”。	R/W

如果在 SPCR.SPE 位为“1”的状态下改写 SPPCR 寄存器，就不保证以后的运行。

SPLP 位 (RSPI 环回位)

此位选择 RSPI 的引脚模式。

当 SPLP 位被置“1”时，如果 SPCR.MSTR 位为“1”，RSPI 就切断 MISOA 引脚和移位寄存器之间的线路，而连接移位寄存器的输入线路和输出线路（取反）；如果 SPCR.MSTR 位为“0”，RSPI 就在从属运行时切断 MOSIA 引脚和移位寄存器之间的线路，而连接移位寄存器的输入线路和输出线路（取反）（环回模式）。

SPLP2 位 (RSPI 环回 2 位)

此位选择 RSPI 的引脚模式。

当 SPLP2 位被置“1”时，如果 SPCR.MSTR 位为“1”，RSPI 就切断 MISOA 引脚和移位寄存器之间的线路，而连接移位寄存器的输入线路和输出线路；如果 SPCR.MSTR 位为“0”，RSPI 就在从属运行时切断 MOSIA 引脚和移位寄存器之间的线路，而连接移位寄存器的输入线路和输出线路（环回模式）。

MOIFV 位 (MOSI 空闲固定值位)

在主机模式中，当 MOIFE 位为“1”时，此位选择 SSL 无效期间（包括突发传送中的 SSL 保持期间）的 MOSIA 引脚输出值。

MOIFE 位 (MOSI 空闲值固定允许位)

主机模式的 RSPI 在 SSL 无效期间（包含突发传送中的 SSL 保持期间）使用此位固定 MOSIA 的输出值。当 MOIFE 位为“0”时，RSPI 在 SSL 无效期间内将上次串行传送的最后数据输出到 MOSIA；当 MOIFE 位为“1”时，RSPI 将 MOIFV 位设定的固定值输出到 MOSIA。

30.2.4 RSPI 状态寄存器 (SPSR)

地址 RSPI0.SPSR 0008 8383h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	PERF	MODF	DLNF	OVRF
复位后的值	x	0	x	0	0	0	0	0

x: 不定值

位	符号	位名	功能	R/W
b0	OVRF	溢出错误标志	0: 未发生溢出错误 1: 发生溢出错误	R/(W) (注1)
b1	IDLNF	RSPI 空闲标志	0: RSPI 为空闲状态 1: RSPI 为传送状态	R
b2	MODF	模式故障错误标志	0: 未发生模式故障错误 1: 发生模式故障错误	R/(W) (注1)
b3	PERF	奇偶校验错误标志	0: 未发生奇偶校验错误 1: 发生奇偶校验错误	R/(W) (注1)
b4	—	保留位	读写值都为“0”。	R/W
b5	—	保留位	读取值为不定值, 只能写“1”。	R/W
b6	—	保留位	读写值都为“0”。	R/W
b7	—	保留位	读取值为不定值, 只能写“1”。	R/W

注1. 为了清除标志, 只能在读“1”后写“0”。

SPSR 寄存器是保存表示 RSPI 运行状态的标志的寄存器。只在一定条件下才能写 SPSR 寄存器。

OVRF 标志 (溢出错误标志)

此标志表示溢出错误的发生状况。

[为“1”的条件]

- 在 SPCR.TXMD 位为“0”并且接收缓冲器有未读数据的状态下串行传送结束时

[为“0”的条件]

- 在 OVRF 标志为“1”的状态下读 SPSR 寄存器后给 OVRF 标志写“0”时

IDLNF 标志 (RSPI 空闲标志)

此标志表示 RSPI 的传送状况。

[为“1”的条件]

【主控模式】

- 当下述[为“0”的条件]的【主控模式】中的条件一个也不满足时

【从属模式】

- 当 SPCR.SPE 位为“1” (RSPI 功能有效) 时

[为“0”的条件]

【主控模式】

- 当 SPCR.SPE 位为“0” (RSPI 的初始化) 时
- 当发送缓冲器 (SPTX) 为空 (未设定下一个传送数据) 时
- 当 SPSR.SPCP[2:0] 位为“000b” (顺序控制的开头) 时
- 当 RSPI 内部定序器转移到空闲状态时 (在下次存取延迟前结束运行的状态)

当满足上述 1. 条件或者满足上述 2. ~ 4. 的全部条件时

【从属模式】

- 当 SPCR.SPE 位为“0” (RSPI 的初始化) 时

MODF 标志 (模式故障错误标志)

此标志表示发生了模式故障错误。

[为“1”的条件]

【多主控模式】

- 在 SPCR.MSTR 位为“1” (主控模式) 并且 SPCR.MODFEN 位为“1” (允许模式故障错误的检测) 的状态下, SSLAi 引脚的输入电平为有效电平并且 RSPI 检测到模式故障错误时

【从属模式】

- 在 SPCR.MSTR 位为“0” (从属模式) 并且 SPCR.MODFEN 位为“1” (允许模式故障错误的检测) 的状态下, 在数据传送所需的 RSPCK 周期结束前 SSLAi 引脚无效并且 RSPI 检测到模式故障错误时

SSLAi 信号的有效电平取决于 SSLP.SSLiP 位 (SSL 信号极性设定位)。

[为“0”的条件]

- 在 MODF 标志为“1”的状态下读 SPSR 寄存器后给 MODF 标志写“0”时

PERF 标志 (奇偶校验错误标志)

此标志表示发生了奇偶校验错误。

[为“1”的条件]

- 在 SPCR.TXMD 位为“0”并且 SPCR2.SPPE 位为“1”的状态下串行传送结束, 并且检测到奇偶校验错误时

[为“0”的条件]

- 在 PERF 标志为“1”的状态下读 SPSR 寄存器后给 PERF 标志写“0”时

30.2.5 RSPI 数据寄存器 (SPDR)

地址 RSPI0.SPDR 0008 8384h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24	SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16
复位后的值	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
复位后的值	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SPDR 寄存器是保存 RSPI 发送 / 接收数据的缓冲器。发送缓冲器和接收缓冲器是独立的缓冲器。SPDR 寄存器的结构图如图 30.2 所示。

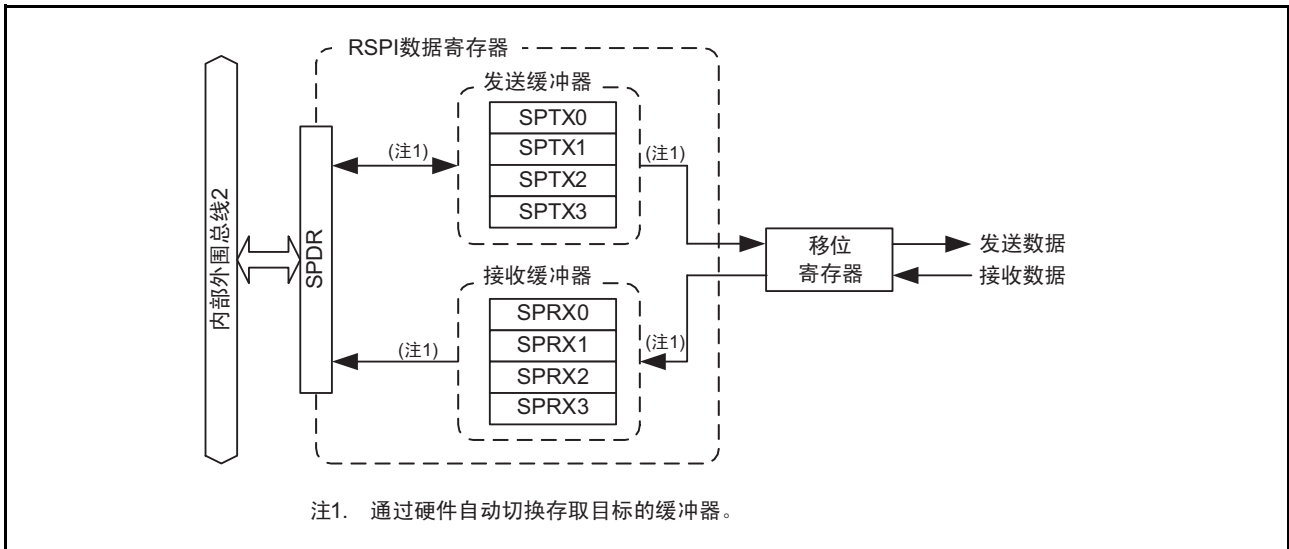


图 30.2 SPDR 寄存器的结构图

发送缓冲器和接收缓冲器分别有 4 个缓冲器，能通过 RSPI 数据控制寄存器的帧数设定位 (SPDCR.SPFC[1:0]) 设定要使用的缓冲器个数。这 8 个缓冲器被映像到 SPDR 寄存器的 1 个地址。

能通过写 SPDR 寄存器，将值写到发送缓冲器 (SPTX0 ~ 3) 并且发送写的的数据。

如果接收完数据，接收缓冲器就保存接收数据。当发生溢出错误时，不更新接收缓冲器的值。

当数据长度不是 32 位时，将 SPTX_n (n=0 ~ 3) 的非参照位保存到 SPRX_n (n=0 ~ 3) 的非参照位。例如，当数据长度是 9 位时，将 SPTX_n[31:9] 保存到 SPRX_n[31:9] (SPTX_n[8:0] 保存接收数据)。

(1) 总线接口

SPDR 寄存器有 4 个 32 位发送缓冲器和 4 个 32 位接收缓冲器，合计 32 字节。此 32 字节被映像到 SPDR 寄存器的 4 字节空间。通过 RSPI 数据控制寄存器的 RSPI 长字存取 / 字存取设定位 (SPDCR.SPLW) 设定的存取长度，进行 SPDR 寄存器的存取。

必须以往 LSB 靠紧的格式写发送数据。以往 LSB 靠紧的格式保存接收数据。

SPDR 寄存器的读写操作如下所示。

(a) 写操作

发送缓冲器有发送缓冲器写指针，通过写 SPDR 寄存器，指针自动切换到下一个缓冲器。

发送缓冲器的总线接口 (写时) 的结构图如图 30.3 所示。

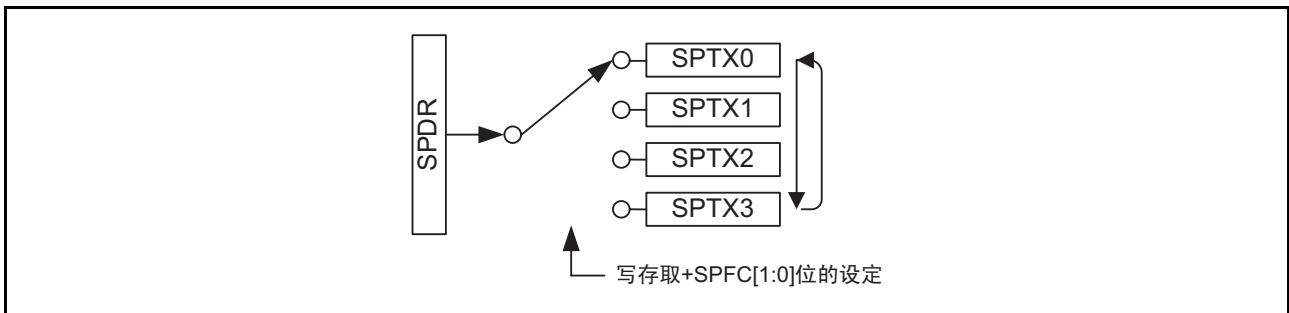


图 30.3 SPDR 寄存器的结构图 (写时)

发送缓冲器写指针的切换顺序因 RSPI 数据控制寄存器的帧数设定位 (SPDCR.SPFC[1:0]) 的设定而不同。

- SPFC[1:0] 位的设定和 SPTX0~3 的切换顺序
 - 当 SPFC[1:0] 位为“00b”时, SPTX0→SPTX0→SPTX0→...
 - 当 SPFC[1:0] 位为“01b”时, SPTX0→SPTX1→SPTX0→SPTX1→...
 - 当 SPFC[1:0] 位为“10b”时, SPTX0→SPTX1→SPTX2→SPTX0→SPTX1→...
 - 当 SPFC[1:0] 位为“11b”时, SPTX0→SPTX1→SPTX2→SPTX3→SPTX0→SPTX1→...

如果在 RSPI 控制寄存器的 RSPI 功能允许位 (SPCR.SPE) 为“0”的状态下写“1”, 下一个写目标就为 SPTX0。

必须在发生 RSPI 发送中断后, 将 RSPI 数据控制寄存器 (SPDCR) 的帧数设定位 (SPFC[1:0]) 所设帧数的发送数据写到发送缓冲器。在发生下一个 RSPI 发送中断前, 即使写发送缓冲器也放弃 SPDR 寄存器的写存取。

(b) 读操作

能通过读 SPDR 寄存器, 读到接收缓冲器 (SPRX0 ~ 3) 或者发送缓冲器 (SPTX0 ~ 3) 的值。能通过 RSPI 数据控制寄存器的 RSPI 接收/发送数据选择位 (SPDCR.SPRDTR) 选择是读接收缓冲器还是读发送缓冲器。

在 SPDR 寄存器的读结构中, 有 2 个独立的指针 (接收缓冲器读指针和发送缓冲器读指针)。通过读 SPDR 寄存器, 指针自动切换到下一个缓冲器。

接收缓冲器和发送缓冲器的总线接口 (读时) 的结构图如图 30.4 所示。

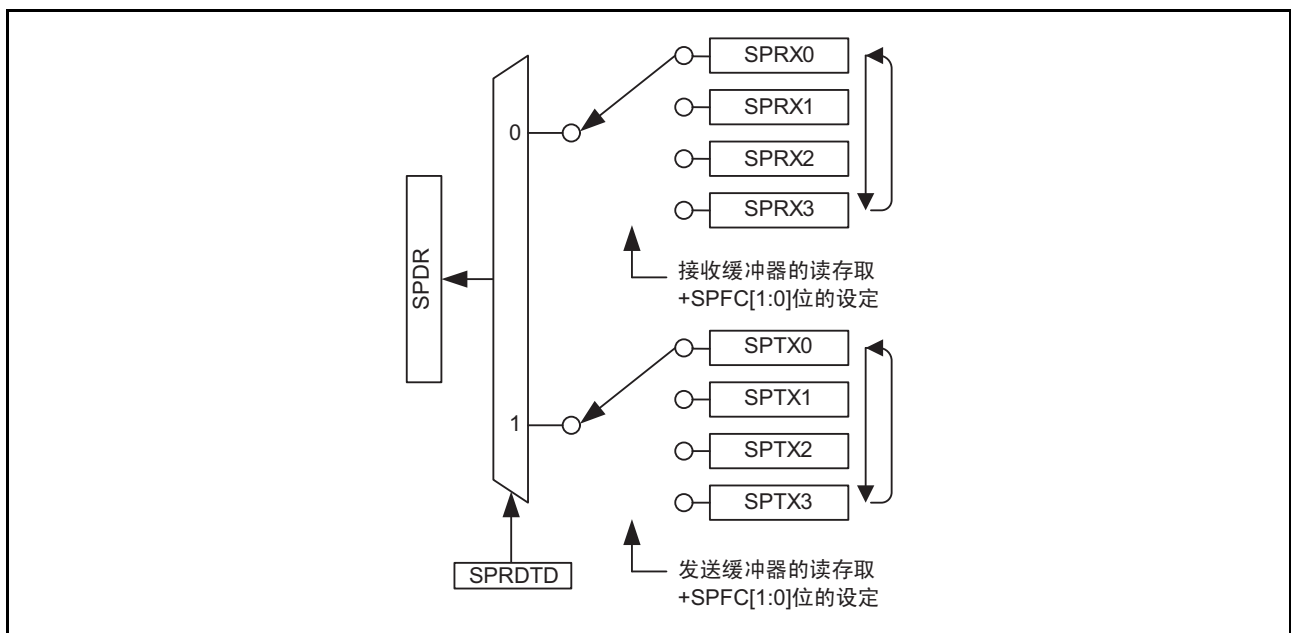


图 30.4 SPDR 寄存器的结构图 (读时)

接收缓冲器读指针的切换顺序和发送缓冲器读指针相同。如果读发送缓冲器, 就能读到刚写入发送缓冲器的值。读时, 只更新由 RSPI 数据控制寄存器的 RSPI 接收/发送数据选择位 (SPDCR.SPRDTR) 选择的读缓冲器的缓冲器读指针, 而其他缓冲器读指针保持不变。

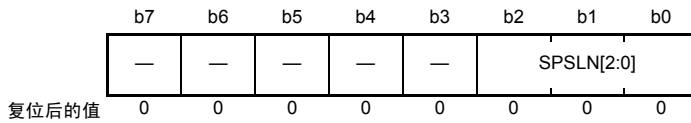
如果在 RSPI 控制寄存器的 RSPI 功能允许位 (SPCR.SPE) 为“0”的状态下写“1”, 下一个读目标就由 SPRXn 的缓冲器读指针指向 SPTX0。

在发生 RSPI 发送中断以及写 RSPI 数据控制寄存器的帧数设定位 (SPDCR.SPFC[1:0]) 所设帧数的发送数据之后, 如果在发生下一个 RSPI 发送中断之前读发送缓冲器, 读取值就全部为“0”。

在发生 RSPI 发送中断后, 如果写 RSPI 数据控制寄存器的帧数设定位 (SPDCR.SPFC[1:0]) 所设帧数的发送数据, 就清除 SPTXn 的缓冲器读指针。

30.2.6 RSPI 顺序控制寄存器 (SPSCR)

地址 RSPI0.SPSCR 0008 8388h



位	符号	位名	功能	R/W
b2-b0	SPSLN[2:0]	RSPI 顺序长度设定位	b2 b0 顺序长度 参照的 SPCMD0 ~ 7 寄存器 (序号) 0 0 0: 1 0→0→... 0 0 1: 2 0→1→0→... 0 1 0: 3 0→1→2→0→... 0 1 1: 4 0→1→2→3→0→... 1 0 0: 5 0→1→2→3→4→0→... 1 0 1: 6 0→1→2→3→4→5→0→... 1 1 0: 7 0→1→2→3→4→5→6→0→... 1 1 1: 8 0→1→2→3→4→5→6→7→0→... 根据所设定的顺序长度, 更改要参照的 SPCMD0 ~ 7 寄存器和参照顺序。SPSLN[2:0] 位的设定值和顺序长度、RSPI 要参照的 SPCMD0 ~ 7 寄存器的关系如上所述。另外, 从属模式的 RSPI 能随时参照 SPCMD0 寄存器。	R/W
b7-b3	—	保留位	读写值都为“0”。	R/W

SPSCR 寄存器是在 RSPI 主控运行时设定顺序长度的寄存器。如果在 SPCR.MSTR 位和 SPCR.SPE 位都为“1”的状态下改写 SPSCR.SPSSLN[2:0] 位, 就必须在 SPSR.IDLNF 标志为“0”时进行改写。

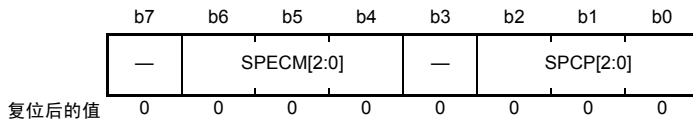
SPSLN[2:0] 位 (RSPI 顺序长度设定位)

这些位设定主控模式的 RSPI 进行顺序运行时的顺序长度。主控模式的 RSPI 根据 SPSLN[2:0] 位所设定的顺序长度, 更改要参照的 SPCMD0 ~ 7 寄存器和参照顺序。

从属模式的 RSPI 能随时参照 SPCMD0 寄存器。

30.2.7 RSPI 顺序状态寄存器 (SPSSR)

地址 RSPI0.SPSSR 0008 8389h



位	符号	位名	功能	R/W
b2-b0	SPCP[2:0]	RSPI 命令指针位	b2 b0 0 0 0: SPCMD0 0 0 1: SPCMD1 0 1 0: SPCMD2 0 1 1: SPCMD3 1 0 0: SPCMD4 1 0 1: SPCMD5 1 1 0: SPCMD6 1 1 1: SPCMD7	R
b3	—	保留位	读取值为“0”。	R
b6-b4	SPECM[2:0]	RSPI 错误命令位	b6 b4 0 0 0: SPCMD0 0 0 1: SPCMD1 0 1 0: SPCMD2 0 1 1: SPCMD3 1 0 0: SPCMD4 1 0 1: SPCMD5 1 1 0: SPCMD6 1 1 1: SPCMD7	R
b7	—	保留位	读取值为“0”。	R

SPSSR 寄存器表示 RSPI 主控运行时的顺序控制状态。

SPSSR 寄存器的写操作无效。

SPCP[2:0] 位 (RSPI 命令指针位)

这些位通过 RSPI 的顺序控制，表示当前指针所指向的 SPCMD_m 寄存器。

有关 RSPI 的顺序控制，请参照“30.3.10.1 主控模式的运行”。

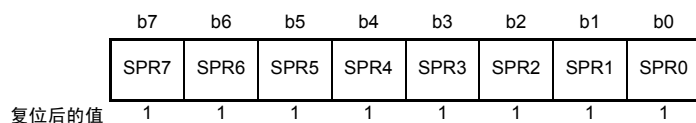
SPECM[2:0] 位 (RSPI 错误命令位)

这些位通过 RSPI 顺序控制，表示在检测到错误时 SPCP[2:0] 位指定的 SPCMD_m 寄存器。只有在检测到错误时，RSPI 才能更新 SPECM[2:0] 位。如果 SPSR.OVRF 位和 MODF 标志都为“0”并且没有发生错误，SPECM[2:0] 位的值就没有含义。

有关 RSPI 的错误检测功能，请参照“30.3.8 错误检测”；有关 RSPI 的顺序控制，请参照“30.3.10.1 主控模式的运行”。

30.2.8 RSPI 位速率寄存器 (SPBR)

地址 RSPi0.SPBR 0008 838Ah



SPBR 寄存器用于设定主控模式中的位速率。如果在 SPCR.MSTR 和 SPCR.SPE 位都为“1”的状态下改写 SPBR 寄存器，就不保证以后的运行。

在从属模式中使用 RSPI 时，与 SPBR 寄存器和 SPCMDm.BRDV[1:0] 位（位速率分频设定位）无关，取决于输入时钟的位速率（必须使用满足电特性的位速率）。

位速率取决于 SPBR 寄存器的设定值和 SPCMDm.BRDV[1:0] 位设定值的组合，位速率的计算式如下。在计算式中，n 为 SPBR 寄存器的设定值（0、1、2、……、255），N 为 BRDV[1:0] 位的设定值（0、1、2、3）。

$$\text{位速率} = \frac{f(\text{PCLK})}{2 \times (n+1) 2^N}$$

SPBR 寄存器、BRDV[1:0] 位的设定值和位速率的关系例子如表 30.3 所示。

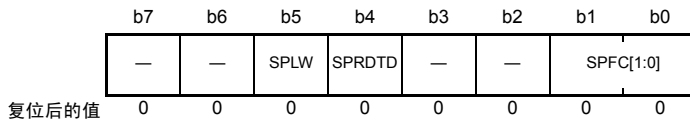
表 30.3 SPBR 寄存器、BRDV[1:0] 位的设定值和位速率（注 1）

SPBR 寄存器的设定值 (n)	BRDV[1:0] 位的设定值 (N)	分频比	位速率
			PCLK=32MHz
1	0	4	8.00Mbps
2	0	6	5.33Mbps
3	0	8	4.00Mbps
4	0	10	3.20Mbps
5	0	12	2.67Mbps
5	1	24	1.33Mbps
5	2	48	667kbps
5	3	96	333kbps
255	3	4096	7.81kbps

注 1. 禁止设定 n=0、N=0 的组合。

30.2.9 RSPI 数据控制寄存器 (SPDCR)

地址 RSPi0.SPDCR 0008 838Bh



位	符号	位名	功能	R/W
b1-b0	SPFC[1:0]	帧数设定位	b1 b0 0 0: 1 帧 0 1: 2 帧 1 0: 3 帧 1 1: 4 帧	R/W
b3-b2	—	保留位	读写值都为“0”。	R/W
b4	SPRDTD	RSPI 接收 / 发送数据选择位	0: SPDR 读接收缓冲器 1: SPDR 读发送缓冲器 (但是, 在发送缓冲器为空时)	R/W
b5	SPLW	RSPI 长字存取 / 字存取设定位	0: 对 SPDR 寄存器进行字存取 1: 对 SPDR 寄存器进行长字存取	R/W
b7-b6	—	保留位	读写值都为“0”。	R/W

通过 SPCMDm.SPB[3:0] 位、SPSCR.SPSSLN[2:0] 位和 SPDCR.SPFC[1:0] 位的组合, 启动 1 次发送和接收能进行最多 4 帧的发送和接收。

如果在 SPCR.SPE 位为“1”的状态下改写 SPDCR.SPFC[1:0] 位, 就必须在 SPSR.IDLNF 标志为“0”时进行改写。

SPFC[1:0] 位 (帧数设定位)

这些位设定 SPDR 寄存器能保存 (1 次传送启动) 的帧数。通过设定 SPSCR.SPSSLN[2:0] 位和 SPDCR.SPFC[1:0] 位, 启动 1 次发送和接收能进行最多 4 帧的发送和接收。另外, 通过设定 SPFC[1:0] 位, 更改用于发生 RSPI 接收中断和 RSPI 发送中断或者开始发送的帧数。SPDR 寄存器能保存的帧结构和发送 / 接收设定的组合例子如表 30.4 所示。如果设定组合例子中没有的组合, 就不保证以后的运行。

表 30.4 SPSSLN[2:0] 位和 SPFC[1:0] 位能设定的组合

设定	SPSSLN[2:0]	SPFC[1:0]	1 个顺序 传送的帧数	发生接收缓冲器满中断或者发送缓冲器 有数据时的帧数
1-1	000	00	1	1
1-2	000	01	2	2
1-3	000	10	3	3
1-4	000	11	4	4
2-1	001	01	2	2
2-2	001	11	4	4
3	010	10	3	3
4	011	11	4	4
5	100	00	5	1
6	101	00	6	1
7	110	00	7	1
8	111	00	8	1

SPRDTD 位 (RSPI 接收 / 发送数据选择位)

此位选择 SPDR 寄存器的读取值是从接收缓冲器读取还是从发送缓冲器读取。

如果读发送缓冲器，就能读到刚写入 SPDR 寄存器的值。

必须在发生 RSPI 发送中断后并且在写完 SPFC[1:0] 位所设帧数前，读发送缓冲器。

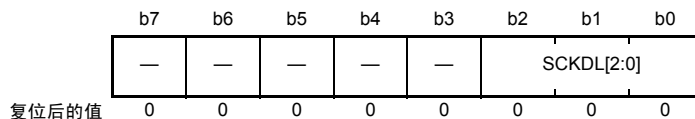
SPLW 位 (RSPI 长字存取 / 字存取设定位)

此位设定 SPDR 寄存器的存取宽度。当 SPLW 位为“0”时，对 SPDR 寄存器进行字存取；当 SPLW 位为“1”时，对 SPDR 寄存器进行长字存取。

当 SPLW 位为“0”时，必须给 SPCMDm.SPB[3:0] 位 (RSPI 数据长度设定位) 设定 8 ~ 16 位。如果设定 20 位、24 位或者 32 位，就不保证运行。

30.2.10 RSPI 时钟延迟寄存器 (SPCKD)

地址 RSPI0.SPCKD 0008 838Ch



位	符号	位名	说明	R/W
b2-b0	SCKDL[2:0]	RSPCK 延迟设定位	b2 b0 0 0 0: 1 个 RSPCK 0 0 1: 2 个 RSPCK 0 1 0: 3 个 RSPCK 0 1 1: 4 个 RSPCK 1 0 0: 5 个 RSPCK 1 0 1: 6 个 RSPCK 1 1 0: 7 个 RSPCK 1 1 1: 8 个 RSPCK	R/W
b7-b3	—	保留位	读写值都为“0”。	R/W

在 SPCMDm.SCKDEN 位为“1”的状态下，SPCKD 寄存器设定从 SSLAi 信号有效开始到 RSPCK 振荡的期间 (RSPCK 延迟)。如果在 SPCR.MSTR 位和 SPCR.SPE 位都为“1”的状态下改写 SPCKD 寄存器，就不保证以后的运行。

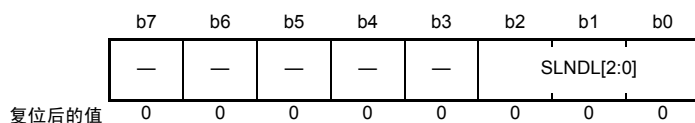
SCKDL[2:0] 位 (RSPCK 延迟设定位)

这些位设定 SPCMDm.SCKDEN 位为“1”时的 RSPCK 延迟值。

在从属模式中使用 RSPI 时，必须将 SCKDL[2:0] 位置“000b”。

30.2.11 RSPI 从属选择无效延迟寄存器 (SSLND)

地址 RSPi0.SSLND 0008 838Dh



位	符号	位名	功能	R/W
b2-b0	SLNDL[2:0]	SSL 无效延迟设定位	b2 b0 0 0 0: 1 个 RSPCK 0 0 1: 2 个 RSPCK 0 1 0: 3 个 RSPCK 0 1 1: 4 个 RSPCK 1 0 0: 5 个 RSPCK 1 0 1: 6 个 RSPCK 1 1 0: 7 个 RSPCK 1 1 1: 8 个 RSPCK	R/W
b7-b3	—	保留位	读写值都为“0”。	R/W

SSLND 寄存器设定主控模式的 RSPI 从发送串行传送最后的 RSPCK 边沿到将 SSLAi 信号置为无效的期间 (SSL 无效延迟)。如果在 SPCR.MSTR 位和 SPCR.SPE 位都为“1”的状态下改写 SSLND 寄存器, 就不保证以后的运行。

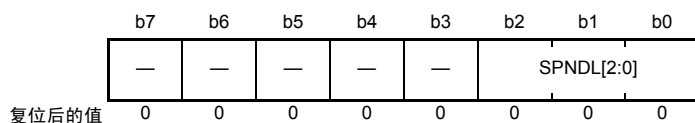
SLNDL[2:0] 位 (SSL 无效延迟设定位)

这些位设定主控模式的 RSPI 的 SSL 无效延迟值。

在从属模式中使用 RSPI 时, 必须将 SLNDL[2:0] 位置“000b”。

30.2.12 RSPI 下次存取延迟寄存器 (SPND)

地址 RSPi0.SPND 0008 838Eh



位	符号	位名	功能	R/W
b2-b0	SPNDL[2:0]	RSPI 下次存取延迟设定位	b2 b0 0 0 0: 1 个 RSPCK+2 个 PCLK 0 0 1: 2 个 RSPCK+2 个 PCLK 0 1 0: 3 个 RSPCK+2 个 PCLK 0 1 1: 4 个 RSPCK+2 个 PCLK 1 0 0: 5 个 RSPCK+2 个 PCLK 1 0 1: 6 个 RSPCK+2 个 PCLK 1 1 0: 7 个 RSPCK+2 个 PCLK 1 1 1: 8 个 RSPCK+2 个 PCLK	R/W
b7-b3	—	保留位	读写值都为“0”。	R/W

在 SPCMDm.SPNDEN 位为“1”的状态下，SPND 寄存器设定串行传送结束后的 SSLAi 信号的无效期间（下次存取延迟）。如果在 SPCR.MSTR 位和 SPCR.SPE 位都为“1”的状态下改写 SPND 寄存器，就不保证以后的运行。

SPNDL[2:0] 位 (RSPI 下次存取延迟设定位)

这些位设定 SPCMDm.SPNDEN 位为“1”时的下次存取延迟。

在从属模式中使用 RSPI 时，必须将 SPNDL[2:0] 位置“000b”。

30.2.13 RSPI 控制寄存器 2 (SPCR2)

地址 RSPI0.SPCR2 0008 838Fh

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	PTE	SPIIE	SPOE	SPPE

复位后的值

0 0 0 0 0 0 0 0

位	符号	位名	功能	R/W
b0	SPPE	奇偶校验允许位	0: 不给发送数据附加奇偶校验位, 也不进行接收数据的奇偶校验。 1: 给发送数据附加奇偶校验位并且进行接收数据的奇偶校验 (当 SPCR.TXMD=0 时)。 给发送数据附加奇偶校验位, 但是不进行接收数据的奇偶校验 (当 SPCR.TXMD=1 时)。	R/W
b1	SPOE	奇偶校验模式位	0: 通过偶校验进行发送和接收 1: 通过奇校验进行发送和接收	R/W
b2	SPIIE	RSPI 空闲中断允许位	0: 禁止空闲中断请求的产生 1: 允许空闲中断请求的产生	R/W
b3	PTE	奇偶校验自诊断位	0: 奇偶校验电路自诊断功能无效 1: 奇偶校验电路自诊断功能有效	R/W
b7-b4	—	保留位	读写值都为“0”。	R/W

如果在 SPCR.SPE 位为“1”的状态下改写 SPCR2 寄存器的 SPPE 位和 SPOE 位的设定值, 就不保证以后的运行。

SPPE 位 (奇偶校验允许位)

此位选择奇偶校验功能的有效或者无效。

当 SPCR.TXMD 位为“0”并且 SPCR2.SPPE 位为“1”时, 给发送数据附加奇偶校验位并且进行接收数据的奇偶校验。

当 SPCR.TXMD 位为“1”并且 SPCR2.SPPE 位为“1”时, 给发送数据附加奇偶校验位, 但是不进行接收数据的奇偶校验。

SPOE 位 (奇偶校验模式位)

此位设定偶校验或者奇校验。

在决定偶校验的奇偶校验位时, 需要使奇偶校验位和发送 / 接收字符中“1”的个数的合计为偶数。同样, 在决定奇校验的奇偶校验位时, 需要使奇偶校验位和发送 / 接收字符中“1”的个数的合计为奇数。

SPOE 位只在 SPPE 位为“1”时有效。

SPIIE 位 (RSPI 空闲中断允许位)

当检测到 RSPI 为空闲状态并且 SPSR.IDLNF 标志为“0”时, 此位允许或者禁止 RSPI 空闲中断请求的产生。

PTE 位 (奇偶校验自诊断位)

这是为了确认奇偶校验功能处于正常状态而将奇偶校验电路的自诊断置为有效的位。

30.2.14 RSPI 命令寄存器 0 ~ 7 (SPCMD0 ~ SPCMD7)

地址 RSPi0.SPCMD0 0008 8390h、RSPi0.SPCMD1 0008 8392h、RSPi0.SPCMD2 0008 8394h、RSPi0.SPCMD3 0008 8396h、
RSPi0.SPCMD4 0008 8398h、RSPi0.SPCMD5 0008 839Ah、RSPi0.SPCMD6 0008 839Ch、RSPi0.SPCMD7 0008 839Eh

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
SCKDEN	SLNDEN	SPNDEN	LSBF	SPB[3:0]			SSLKP	SSLA[2:0]		BRDV[1:0]		CPOL	CPHA		
0	0	0	0	0	1	1	1	0	0	0	0	1	1	0	1

复位后的值

位	符号	位名	功能	R/W
b0	CPHA	RSPCK 相位设定位	0: 在奇数边沿进行数据采样, 在偶数边沿数据发生变化。 1: 在奇数边沿数据发生变化, 在偶数边沿进行数据采样。	R/W
b1	CPOL	RSPCK 极性设定位	0: 空闲时的 RSPCK 为 Low 电平 1: 空闲时的 RSPCK 为 High 电平	R/W
b3-b2	BRDV[1:0]	位速率分频设定位	b3 b2 0 0: 选择基本位速率 0 1: 选择基本位速率的 2 分频 1 0: 选择基本位速率的 4 分频 1 1: 选择基本位速率的 8 分频	R/W
b6-b4	SSLA[2:0]	SSL 信号有效设定位	b6 b4 0 0 0: SSL0 0 0 1: SSL1 0 1 0: SSL2 0 1 1: SSL3 1 x x: 不能设定 x: Don't care	R/W
b7	SSLKP	SSL 信号电平保持位	0: 在传送结束时将全部 SSL 信号置为无效 1: 在传送结束后到下次存取开始前保持 SSL 信号的电平	R/W
b11-b8	SPB[3:0]	RSPI 数据长度设定位	b11 b8 0100 ~ 0111: 8 位 1 0 0 0: 9 位 1 0 0 1: 10 位 1 0 1 0: 11 位 1 0 1 1: 12 位 1 1 0 0: 13 位 1 1 0 1: 14 位 1 1 1 0: 15 位 1 1 1 1: 16 位 0 0 0 0: 20 位 0 0 0 1: 24 位 0010、0011: 32 位	R/W
b12	LSBF	RSPI LSB first 位	0: MSB first 1: LSB first	R/W
b13	SPNDEN	RSPI 下次存取延迟设定允许位	0: 下次存取延迟为 1 个 RSPCK+2 个 PCLK 1: 下次存取延迟为 RSPI 下次存取延迟寄存器 (SPND) 的设定值	R/W
b14	SLNDEN	SSL 无效延迟设定允许位	0: SSL 无效延迟为 1 个 RSPCK 1: SSL 无效延迟为 RSPI 从属选择无效延迟寄存器 (SSLND) 的设定值	R/W
b15	SCKDEN	RSPCK 延迟设定允许位	0: RSPCK 延迟为 1 个 RSPCK 1: RSPCK 延迟为 RSPI 时钟延迟寄存器 (SPCKD) 的设定值	R/W

SPCMDm 寄存器设定主控模式的 RSPI 传送格式。1 个通道的 RSPI 有 8 个 (SPCMD0 ~ SPCMD7) RSPI 命令寄存器。SPCMD0 寄存器的部分位用于设定从属模式的 RSPI 传送格式。主控模式的 RSPI 根据 SPSCR.SPSSLN[2:0] 位的设定, 按顺序参照 SPCMDm 寄存器并且执行由所参照的 SPCMDm 寄存器设定的串行传送。

必须在发送缓冲器为空 (未设定下次传送的数据) 的状态下, 在参照该 SPCMDm 寄存器来设定要发送的数据前设定 SPCMDm 寄存器。

能通过 SPSSR.SPCP[2:0] 位, 确认主控模式的 RSPI 正在参照的 SPCMDm 寄存器。如果在 SPCR.MSTR 位为“0”并且 SPCR.SPE 位为“1”的状态下改写 SPCMDm 寄存器, 就不保证以后的运行。

CPHA 位 (RSPCK 相位设定位)

此位设定主控模式 / 从属模式的 RSPI 的 RSPCK 相位。在进行 RSPI 模块之间的数据通信时, 需要在模块之间设定相同的 RSPCK 相位。

CPOL 位 (RSPCK 极性设定位)

此位设定主控模式 / 从属模式的 RSPI 的 RSPCK 极性。在进行 RSPI 模块之间的数据通信时, 需要在模块之间设定相同的 RSPCK 极性。

BRDV[1:0] 位 (位速率分频设定位)

这些位设定位速率。位速率取决于 BRDV[1:0] 位和 SPBR 寄存器设定值的组合 (参照“30.2.8 RSPI 位速率寄存器 (SPBR)”)。SPBR 寄存器的设定值决定基本位速率。BRDV[1:0] 位的设定值选择基本位速率的无分频、2 分频、4 分频或者 8 分频。能分别对 SPCMDm 寄存器设定不同的 BRDV[1:0] 位。因此, 各命令能以不同的位速率进行串行传送。

SSLA[2:0] 位 (SSL 信号有效设定位)

这些位控制主控模式的 RSPI 进行串行传送时的 SSLAi 信号的有效。SSLA[2:0] 位的设定值控制 SSLAi 信号的有效。SSLAi 信号有效时的信号极性取决于 SSLP 寄存器的设定值。如果在多主控模式中将 SSLA[2:0] 位置“000b”, 就在全部 SSL 信号无效的状态下进行串行传送 (SSLA0 引脚为输入引脚)。

在从属模式中使用 RSPI 时, 必须将 SSLA[2:0] 位置“000b”。

SSLKP 位 (SSL 信号电平保持位)

在主控模式的 RSPI 进行串行传送时, 此位设定在从当前命令对应的 SSL 无效到下一个命令对应的 SSL 有效的期间是保持当前命令的 SSLAi 信号电平还是将此电平置为无效。

在从属模式中使用 RSPI 时, 必须将 SSLKP 位置“0”。

SPB[3:0] 位 (RSPI 数据长度设定位)

这些位设定主控模式 / 从属模式的 RSPI 传送数据长度。

LSBF 位 (RSPI LSB first 位)

此位设定是将主控模式 / 从属模式的 RSPI 数据格式设定为 MSB first 还是 LSB first。

SPNDEN 位 (RSPI 下次存取延迟允许位)

此位设定在从属模式的 RSPI 结束串行传送后从 SSLAi 信号无效到下次存取的 SSLAi 信号有效的期间 (下次存取延迟)。当 SPNDEN 位为“0”时, RSPI 将下次存取延迟设定为 1 个 RSPCK+2 个 PCLK; 当 SPNDEN 位为“1”时, RSPI 根据 SPND 寄存器的设定, 插入下次存取延迟。

在从属模式中使用 RSPI 时, 必须将 SPNDEN 位置“0”。

SLNDEN 位 (SSL 无效延迟设定允许位)

此位设定主控模式的 RSPI 从 RSPCK 振荡停止到 SSLAi 信号无效的期间 (SSL 无效延迟)。当 SLNDEN 位为“0”时, RSPI 将 SSL 无效延迟设定为 1 个 RSPCK; 当 SLNDEN 位为“1”时, RSPI 在经过 SSLND 寄存器设定的 RSPCK 延迟后将 SSL 置为无效。

在从属模式中使用 RSPI 时, 必须将 SLNDEN 位置“0”。

SCKDEN 位 (RSPCK 延迟设定允许位)

此位设定主控模式的 RSPI 从 SSLAi 信号有效到 RSPCK 振荡为止的期间 (RSPCK 延迟)。当 SCKDEN 位为“0”时, RSPI 将 RSPCK 延迟设定为 1 个 RSPCK; 当 SCKDEN 位为“1”时, RSPI 在经过 SPCKD 寄存器设定的 RSPCK 延迟后开始 RSPCK 的振荡。

在从属模式中使用 RSPI 时, 必须将 SCKDEN 位置“0”。

30.3 运行说明

在本节中，串行传送期间是指从开始驱动有效数据到取最后有效数据的期间。

30.3.1 RSPI 运行概要

RSPI 能在从属模式 (SPI 运行)、单主控模式 (SPI 运行)、多主控模式 (SPI 运行)、从属模式 (时钟同步运行) 和主控模式 (时钟同步运行) 中进行同步串行传送。能通过 SPCR.MSTR 位和 SPCR.MODFEN 位和 SPCR.SPMS 位设定 RSPI 的模式。RSPI 的模式和 SPCR 寄存器设定的关系以及各模式概要如表 30.5 所示。

表 30.5 RSPI 的模式和 SPCR 寄存器设定的关系以及各模式概要

模式	从属 (SPI 运行)	单主控 (SPI 运行)	多主控 (SPI 运行)	从属 (时钟同步运行)	主控 (时钟同步运行)
MSTR 位的设定	0	1	1	0	1
MODFEN 位的设定	0 or 1	0	1	0	0
SPMS 位的设定	0	0	0	1	1
RSPCKA 信号	输入	输出	输出 /Hi-Z	输入	输出
MOSIA 信号	输入	输出	输出 /Hi-Z	输入	输出
MISOA 信号	输出 /Hi-Z	输入	输入	输出	输入
SSLA0 信号	输入	输出	输入	Hi-Z (注1)	Hi-Z (注1)
SSLA1 ~ SSLA3 信号	Hi-Z (注1)	输出	输出 /Hi-Z	Hi-Z (注1)	Hi-Z (注1)
SSL 极性变更功能	有	有	有	—	—
传送率	~ PCLK/8	~ PCLK/2	~ PCLK/2	~ PCLK/8	~ PCLK/2
时钟源	RSPCK 输入	内部波特率发生器	内部波特率发生器	RSPCK 输入	内部波特率发生器
时钟极性	2 种	2 种	2 种	2 种	2 种
时钟相位	2 种	2 种	2 种	1 种 (CPHA=1)	2 种
开始传送的位	MSB/LSB	MSB/LSB	MSB/LSB	MSB/LSB	MSB/LSB
传送数据长度	8 ~ 32 位	8 ~ 32 位	8 ~ 32 位	8 ~ 32 位	8 ~ 32 位
突发传送	能 (CPHA=1)	能 (CPHA=0、1)	能 (CPHA=0、1)	—	—
RSPCK 延迟控制	无	有	有	无	有
SSL 无效延迟控制	无	有	有	无	有
下次存取延迟控制	无	有	有	无	有
传送启动方法	SSL 输入有效或者 RSPCK 振荡	通过发送缓冲器空中断请求，写发送缓冲器	通过发送缓冲器空中断请求，写发送缓冲器	RSPCK 振荡	通过发送缓冲器空中断请求，写发送缓冲器
顺序控制	无	有	有	无	有
发送缓冲器空检测	有	有	有	有	有
接收缓冲器满检测	有 (注2)	有 (注2)	有 (注2)	有 (注2)	有 (注2)
溢出错误检测	有 (注2)	有 (注2)	有 (注2)	有 (注2)	有 (注2)
奇偶校验错误检测	有 (注2) (注3)	有 (注2) (注3)	有 (注2) (注3)	有 (注2) (注3)	有 (注2) (注3)
模式故障错误检测	有 (MODFEN=1)	无	有	无	无

注 1. 在此模式中不使用。

注 2. 当 SPCR.TXMD 位为“1”时，不进行接收缓冲器满的检测、溢出错误的检测和奇偶校验错误的检测。

注 3. 当 SPCR2.SPPE 位为“0”时，不进行奇偶校验错误的检测。

30.3.2 RSPI 引脚的控制

RSPI 通过设定 SPCR.MSTR 位、SPCR.MODFEN 位、SPCR.SPMS 位和 I/O 端口的 ODRn.Bi 位来转换引脚状态。引脚状态和各位设定值的关系如表 30.6 所示。I/O 端口的设定也必须相同。

表 30.6 RSPI 引脚的状态和控制位设定值的关系

模式	引脚	引脚状态 (注 2)	
		I/O 端口的 ODRn.Bi 位 =0	I/O 端口的 ODRn.Bi 位 =1
单主控 (SPI 运行) (MSTR=1、MODFEN=0、SPMS=0)	RSPCKA	CMOS 输出	漏极开路输出
	SSLA0 ~ 3	CMOS 输出	漏极开路输出
	MOSIA	CMOS 输出	漏极开路输出
	MISOA	输入	输入
多主控 (SPI 运行) (MSTR=1、MODFEN=1、SPMS=0)	RSPCKA (注 3)	CMOS 输出 /Hi-Z	漏极开路输出 /Hi-Z
	SSLA0	输入	输入
	SSLA1 ~ 3 (注 3)	CMOS 输出 /Hi-Z	漏极开路输出 /Hi-Z
	MOSIA (注 3)	CMOS 输出 /Hi-Z	漏极开路输出 /Hi-Z
	MISOA	输入	输入
从属 (SPI 运行) (MSTR=0、SPMS=0)	RSPCKA	输入	输入
	SSLA0	输入	输入
	SSLA1 ~ 3 (注 5)	Hi-Z (注 1)	Hi-Z (注 1)
	MOSIA	输入	输入
	MISOA (注 4)	CMOS 输出 /Hi-Z	漏极开路输出 /Hi-Z
主控 (时钟同步运行) (MSTR=1、MODFEN=0、SPMS=1)	RSPCKA	CMOS 输出	漏极开路输出
	SSLA0 ~ 3 (注 5)	Hi-Z (注 1)	Hi-Z (注 1)
	MOSIA	CMOS 输出	漏极开路输出
	MISOA	输入	输入
从属 (时钟同步运行) (MSTR=0、SPMS=1)	RSPCKA	输入	输入
	SSLA0 ~ 3 (注 5)	Hi-Z (注 1)	Hi-Z (注 1)
	MOSIA	输入	输入
	MISOA	CMOS 输出	漏极开路输出

注 1. 在此模式中不使用。

注 2. 未选择 RSPI 功能的多功能引脚不反映 RSPI 的设定值。

注 3. 当 SSLA0 为有效电平时，引脚状态为 Hi-Z。

注 4. 当 SSLA0 为无效电平或者 SPCR.SPE 位为“0”时，引脚状态为 Hi-Z。

注 5. 能用作 I/O 端口。

如表 30.7 所示，单主控模式 (SPI 运行) 和多主控模式 (SPI 运行) 的 RSPI 根据 SPPCR.MOIFE 位和 SPPCR.MOIFV 位的设定，决定 SSL 无效期间 (包括突发传送中的 SSL 保持期间) 的 MOSI 信号值。

表 30.7 SSL 无效期间的 MOSI 信号值的决定方法

MOIFE 位	MOIFV 位	SSL 无效期间的 MOSIA 信号值
0	0、1	上次传送的最后数据
1	0	总是为 Low 电平。
1	1	总是为 High 电平。

30.3.3 RSPI 系统结构例子

30.3.3.1 单主控模式 / 单从属模式 (本 LSI= 主控设备)

本 LSI 用作主控设备时的单主控模式 / 单从属模式的 RSPI 系统结构例子如图 30.5 所示。在单主控模式 / 单从属模式的系统结构例子中，不使用本 LSI (主控设备) 的 SSLA0 ~ SSLA3 输出。通过将 RSPI 从属设备的 SSL 输入固定为 Low 电平，设定为能随时选择 RSPI 从属模式的状态 (注 1)。

本 LSI (主控设备) 随时驱动 RSPCKA 和 MOSIA，RSPI 从属设备随时驱动 MISO。

注 1. 在类似 SPCMDm.CPHA 位为“0”时的传送格式中，也有些从属设备不能将 SSL 信号固定为有效电平。在不能固定 SSL 信号时，必须将本 LSI 的 SSL 输出连接到从属设备的 SSLAi 输入。

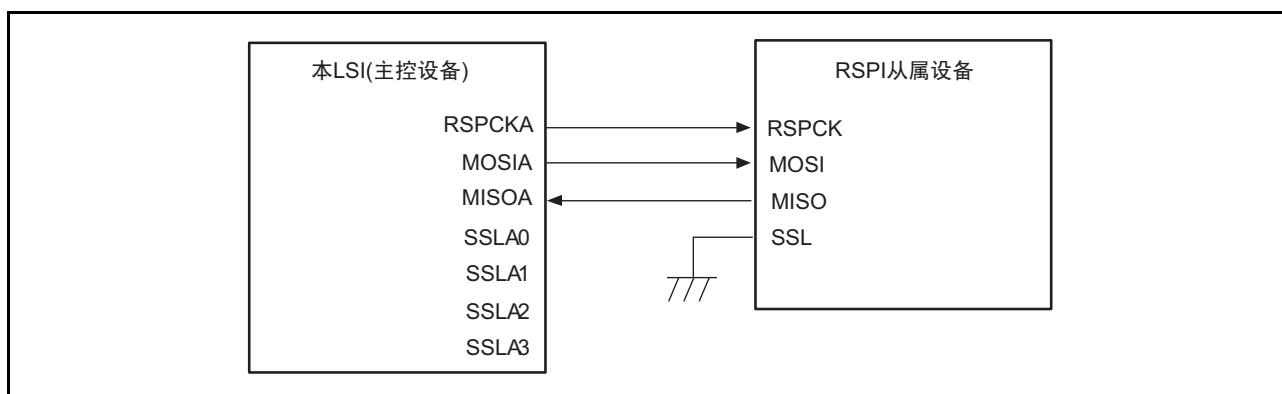


图 30.5 单主控模式 / 单从属模式的系统结构例子 (本 LSI= 主控设备)

30.3.3.2 单主控模式 / 单从属模式 (本 LSI= 从属设备)

将本 LSI 用作从属设备时的单主控模式 / 单从属模式的 RSPI 系统结构例子如图 30.6 所示。在将本 LSI 用作从属设备时，将 SSLA0 引脚用作 SSL 输入。RSPI 主控设备随时驱动 RSPCK 和 MOSI，本 LSI (从属设备) 随时驱动 MISOA (注 1)。

在 SPCMDm.CPHA 位为“1”的单从属模式的系统结构中，通过将本 LSI (从属设备) 的 SSLA0 输入引脚固定为 Low 电平，设定为能随时选择本 LSI (从属设备) 的状态并且也能进行串行传送 (图 30.7)。

注 1. 在 SSLA0 为无效电平时，引脚状态为 Hi-Z。

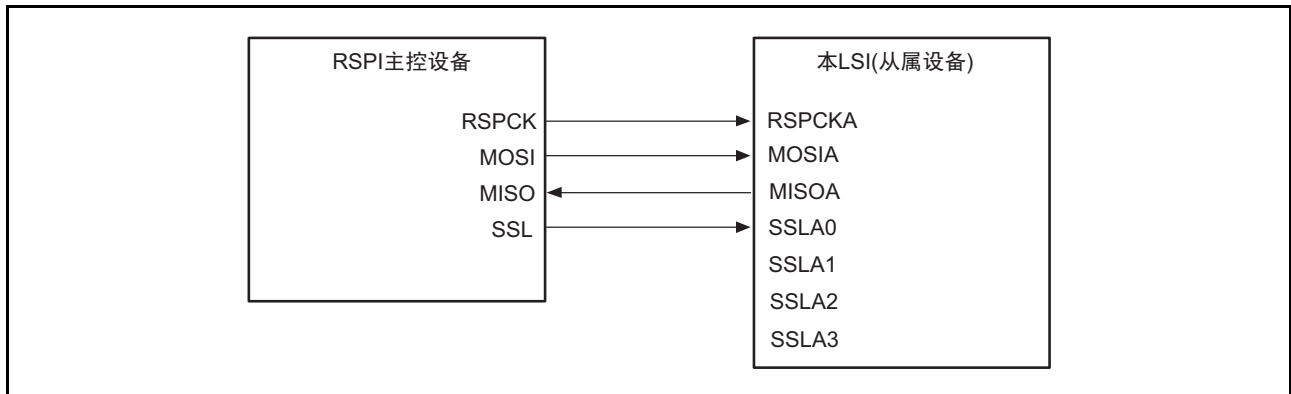


图 30.6 单主控模式 / 单从属模式的系统结构例子 (本 LSI= 从属设备、CPHA=0)

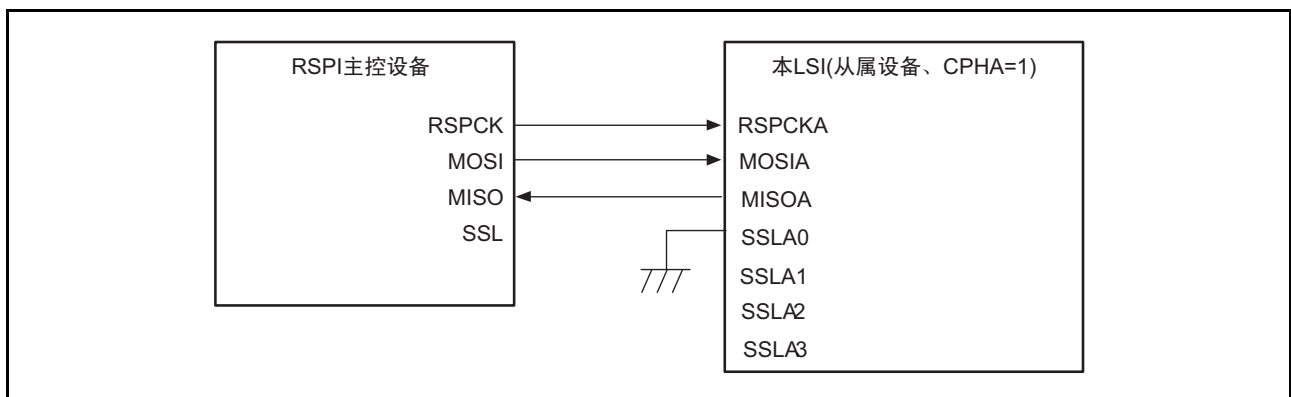


图 30.7 单主控模式 / 单从属模式的系统结构例子 (本 LSI= 从属设备、CPHA=1)

30.3.3.3 单主控模式 / 多从属模式 (本 LSI= 主控设备)

本 LSI 用作主控设备时的单主控模式 / 多从属模式的 RSPI 系统结构例子如图 30.8 所示。在图 30.8 的例子中，RSPI 系统由本 LSI (主控设备) 和 4 个从属设备 (RSPI 从属设备 0 ~ RSPI 从属设备 3) 构成。

将本 LSI (主控设备) 的 RSPCKA 输出和 MOSIA 输出连接 RSPI 从属设备 0 ~ RSPI 从属设备 3 的 RSPCK 输入和 MOSI 输入，RSPI 从属设备 0 ~ RSPI 从属设备 3 的 MISO 输出全部连接本 LSI (主控设备) 的 MISOA 输入，本 LSI (主控设备) 的 SSLA0 ~ SSLA3 输出分别连接 RSPI 从属设备 0 ~ RSPI 从属设备 3 的 SSL 输入。

本 LSI (主控设备) 随时驱动 RSPCK、MOSI 和 SSLA0 ~ SSLA3。在 RSPI 从属设备 0 ~ 从属设备 3 中，SSL 输入为 Low 电平的从属设备驱动 MISO。

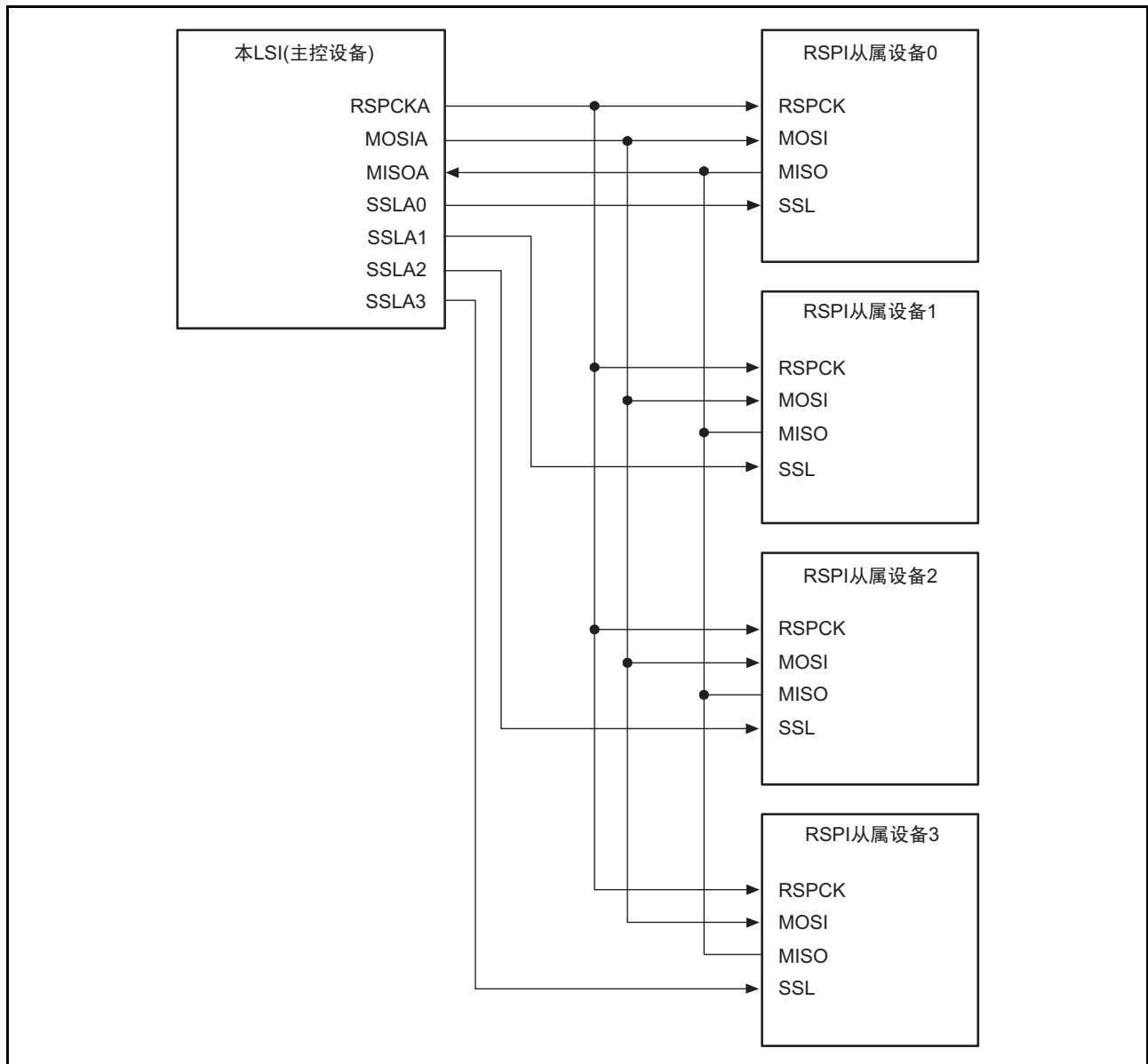


图 30.8 单主控模式 / 多从属模式的系统结构例子 (本 LSI= 主控设备)

30.3.3.4 单主控模式 / 多从属模式 (本 LSI= 从属设备)

本 LSI 用作从属设备时的单主控模式 / 多从属模式的 RSPI 系统结构例子如图 30.9 所示。在图 30.9 的例子中，RSPI 系统由 RSPI 主控设备和 2 个本 LSI (从属设备 X、从属设备 Y) 构成。

RSPI 主控设备的 RSPCK 输出和 MOSI 输出连接本 LSI (从属设备 X、从属设备 Y) 的 RSPCKA 输入和 MOSIA 输入，本 LSI (从属设备 X、从属设备 Y) 的 MISOA 输出连接 RSPI 主控设备的 MISO 输入，RSPI 主控设备的 SSLX 输出和 SSLY 输出连接本 LSI (从属设备 X、从属设备 Y) 的 SSLA0 输入。

RSPI 主控设备随时驱动 RSPCK、MOSI、SSLX 和 SSLY。在本 LSI (从属设备 X、从属设备 Y) 中，SSLA0 输入为 Low 电平的从属设备驱动 MISOA。

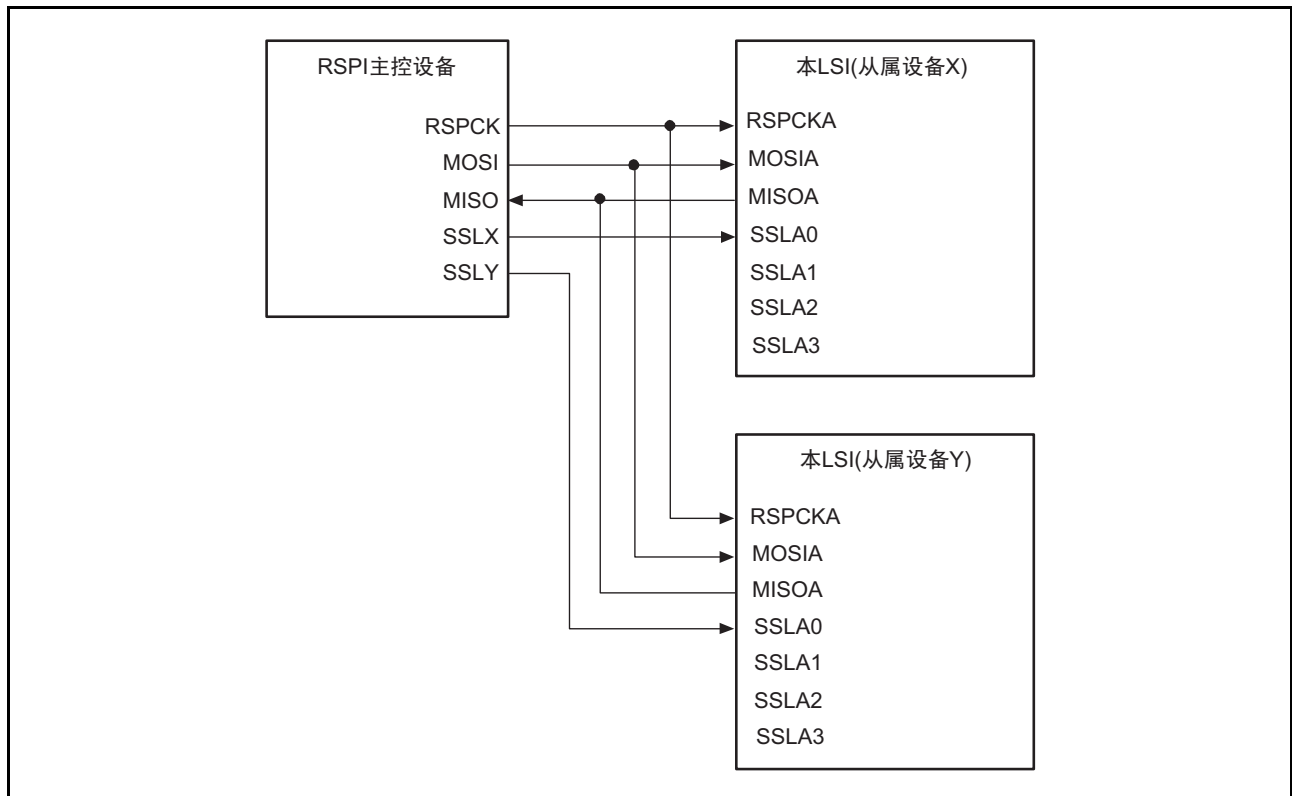


图 30.9 单主控模式 / 多从属模式的系统结构例子 (本 LSI= 从属设备)

30.3.3.5 多主控模式 / 多从属模式 (本 LSI= 主控设备)

本 LSI 用作主控设备时的多主控模式 / 多从属模式的 RSPI 系统结构例子如图 30.10 所示。在图 30.10 的例子中，RSPI 系统由 2 个本 LSI (主控设备 X、主控设备 Y) 和 2 个 RSPI 从属设备 (RSPI 从属设备 1、RSPI 从属设备 2) 构成。

本 LSI (主控设备 X、主控设备 Y) 的 RSPCKA 输出和 MOSIA 输出连接 RSPI 从属设备 1 和 RSPI 从属设备 2 的 RSPCK 输入和 MOSI 输入，RSPI 从属设备 1 和 RSPI 从属设备 2 的 MISO 输出连接本 LSI (主控设备 X、主控设备 Y) 的 MISOA 输入，本 LSI (主控设备 X) 任意的通用端口 Y 输出引脚连接本 LSI (主控设备 Y) 的 SSLA0 输入，本 LSI (主控设备 Y) 任意的通用端口 X 输出连接本 LSI (主控设备 X) 的 SSLA0 输入，本 LSI (主控设备 X、主控设备 Y) 的 SSLA1 输出和 SSLA2 输出连接 RSPI 从属设备 1 和 RSPI 从属设备 2 的 SSL 输入。在此结构例子中，因为系统只由 SSLA0 输入、连接从属设备的 SSLA1 输出和 SSLA2 输出构成，所以不使用本 LSI 的 SSLA3 输出。

当 SSLA0 输入电平为 High 电平时，本 LSI 驱动 RSPCKA、MOSIA、SSLA1 和 SSLA2；当 SSLA0 输入电平为 Low 电平时，检测模式故障错误，在将 RSPCKA、MOSIA、SSLA1 和 SSLA2 置为 Hi-Z 后，将 RSPI 总线权释放给其他主控设备。在 RSPI 从属设备 1 和 RSPI 从属设备 2 中，SSL 输入为 Low 电平的从属设备驱动 MISO。

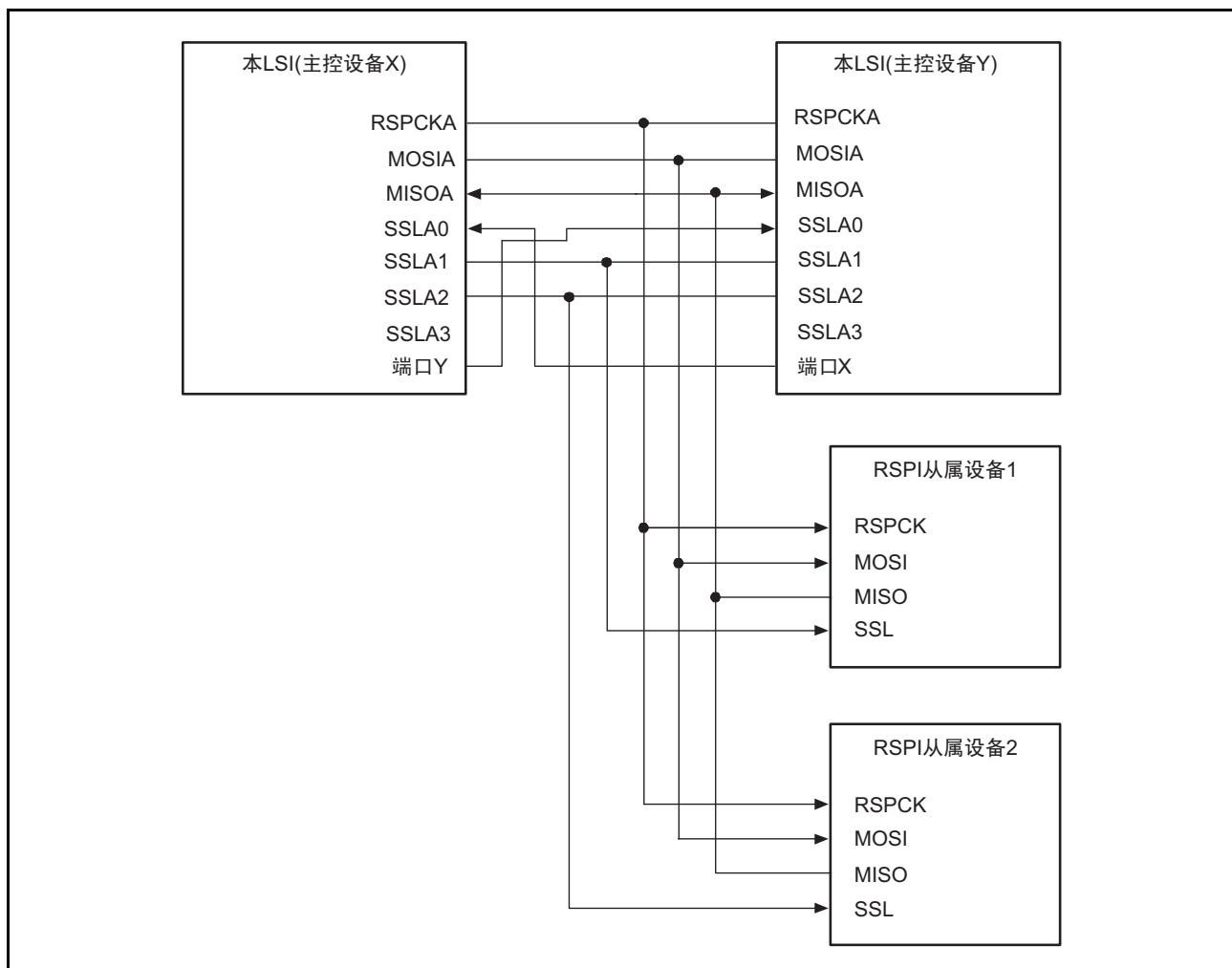


图 30.10 多主控模式 / 多从属模式的系统结构例子 (本 LSI= 主控设备)

30.3.3.6 主控模式 (时钟同步运行) / 从属模式 (时钟同步运行) (本 LSI= 主控设备)

本 LSI 用作主控设备时的主控模式 (时钟同步运行) / 从属模式 (时钟同步运行) 的 RSPI 系统结构例子如图 30.11 所示。在 主控模式 (时钟同步运行) / 从属模式 (时钟同步运行) 的结构中, 不使用本 LSI (主控设备) 的 SSLA0 ~ SSLA3。

本 LSI (主控设备) 随时启动 RSPCKA 和 MOSIA, RSPI 从属设备随时驱动 MISO。

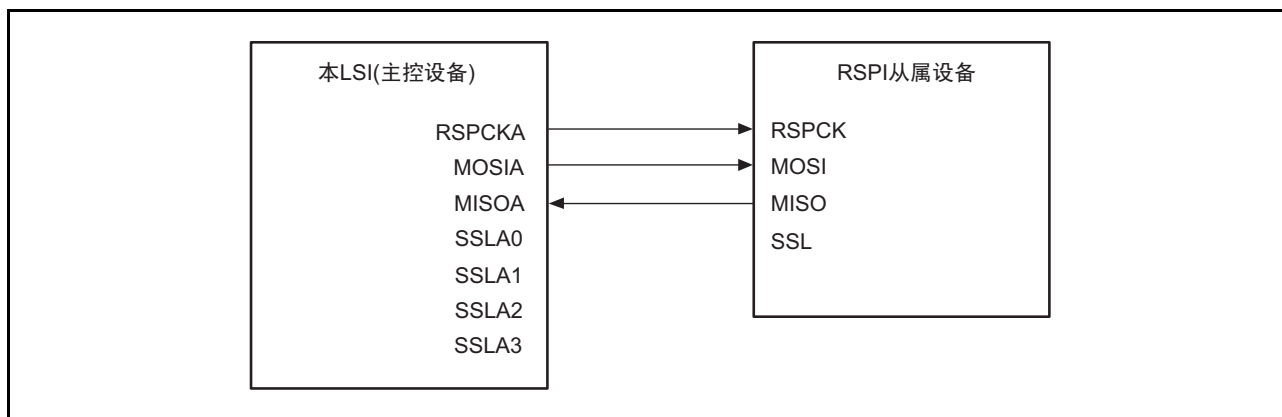


图 30.11 主控模式 (时钟同步运行) / 从属模式 (时钟同步运行) 的系统结构例子 (本 LSI= 主控设备)

30.3.3.7 主控模式 (时钟同步运行) / 从属模式 (时钟同步运行) (本 LSI= 从属设备)

本 LSI 用作从属设备时的主控模式 (时钟同步运行) / 从属模式 (时钟同步运行) 的 RSPI 系统结构例子如图 30.12 所示。在将本 LSI 用作从属设备 (时钟同步运行) 时, 本 LSI (从属设备) 随时驱动 MISOA, RSPI 主控设备随时驱动 RSPCK 和 MOSI。不使用本 LSI (从属设备) 的 SSLA0 ~ SSLA3。

只有在将 SPCMDm.CPHA 位置“1”的单从属模式的系统结构中, 本 LSI (从属设备) 才能进行串行传送。

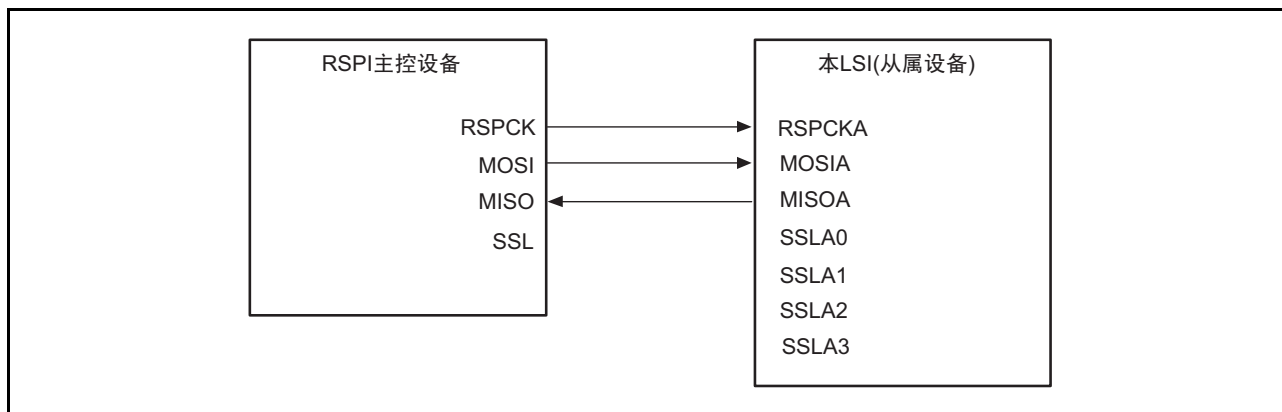


图 30.12 主控模式 (时钟同步运行) / 从属模式 (时钟同步运行) 的系统结构例子 (本 LSI= 从属设备, CPHA=1)

30.3.4 数据格式

RSPI 的数据格式取决于 RSPI 命令寄存器 m (SPCMD m) ($m=0 \sim 7$) 和 RSPI 控制寄存器 2 的奇偶校验允许位 (SPCR2.SPPE) 设定值。与 MSB/LSB first 无关, RSPI 将 RSPI 数据寄存器 (SPDR) 的 LSB 位开始的设定数据长度的范围作为传送数据进行处理。

发送和接收时的 1 帧数据格式如下所示。

(a) 奇偶校验功能无效的情况

在奇偶校验功能无效时, 进行 RSPI 命令寄存器 m 的 RSPI 数据长度设定位 (SPCMD m .SPB[3:0]) 所设位长的数据发送和接收。

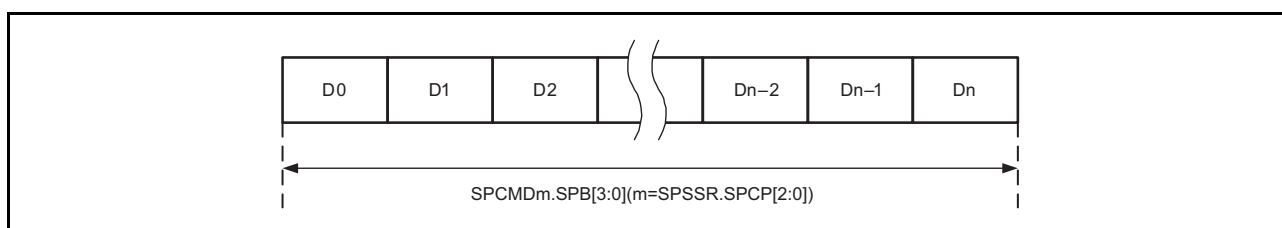


图 30.13 数据格式概要 (奇偶校验功能无效的情况)

(b) 奇偶校验功能有效的情况

在奇偶校验功能有效时, 进行 SPCMD m .SPB[3:0] 所设位长的数据发送和接收。但是, 最后位为奇偶校验位。

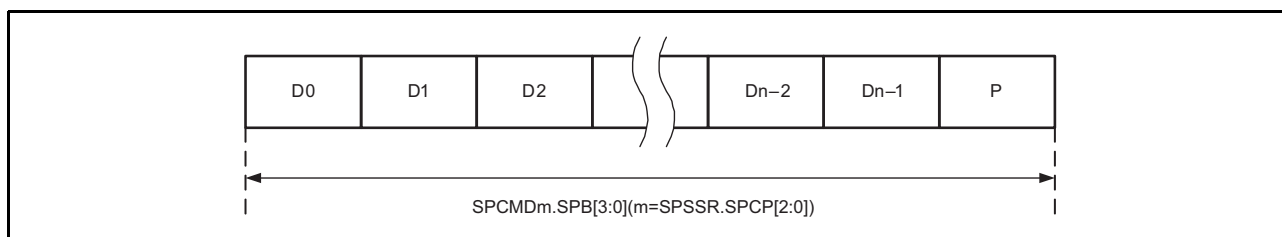


图 30.14 数据格式概要 (奇偶校验功能有效的情况)

30.3.4.1 奇偶校验功能无效的情况 (SPCR2.SPPE=0)

在奇偶校验功能无效时，将发送缓冲器的数据复制到移位寄存器而不进行加工。根据 MSB/LSB first 和位长的组合，对 RSPI 数据寄存器 (SPDR) 和移位寄存器的关系进行以下的说明。

(1) MSB first 传送 (32 位数据)

在奇偶校验功能无效的情况下，RSPI 进行 32 位数据的 MSB first 传送时的 SPDR 寄存器和移位寄存器的运行内容如图 30.15 所示。

在发送时，将发送缓冲器的 T31 ~ T00 复制到移位寄存器。按 T31→T30→...→T00 的顺序将移位寄存器的值进行移位，并且发送数据。

在接收时，将接收数据保存到移位寄存器的 bit0，并且按数据进行接收数据的移位。输入需要的 RSPCK，如果收集到 R31 ~ R00 的数据，就将移位寄存器的值复制到接收缓冲器。

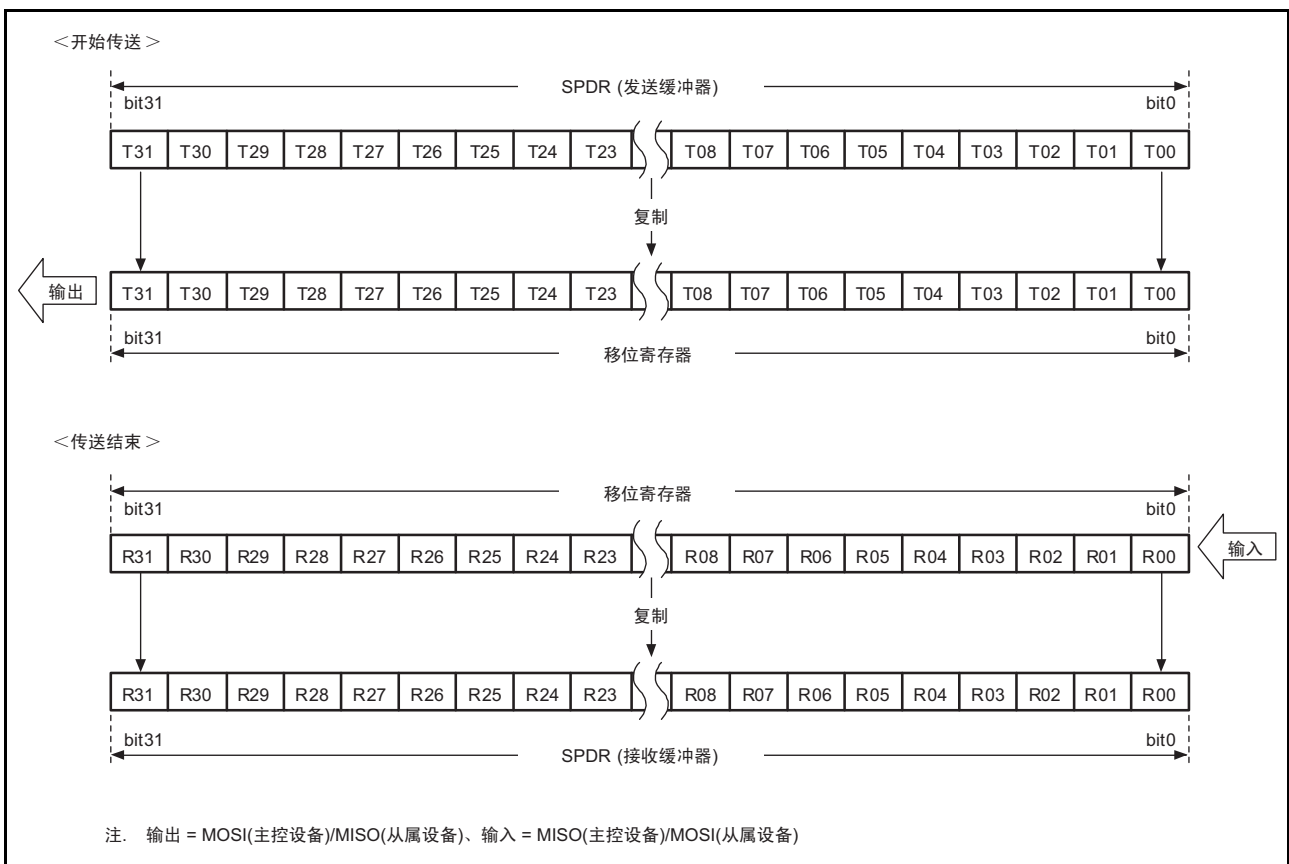


图 30.15 MSB first 传送 (32 位数据 / 奇偶校验功能无效)

(2) MSB first 传送 (24 位数据)

在奇偶校验功能无效的情况下，以对 32 位以外的数据进行 MSB first 传送为例，RSPI 进行 24 位数据传送时的 RSPI 数据寄存器 (SPDR) 和移位寄存器的运行内容如图 30.16 所示。

在发送时，将发送缓冲器的低 24 位 (T23 ~ T00) 复制到移位寄存器。按 T23→T22→...→T00 的顺序将移位寄存器的值进行移位，并且发送数据。

在接收时，将接收数据保存到移位寄存器的 bit0，并且按数据进行接收数据的移位。输入需要的 RSPCK，如果收集到 R23 ~ R00 的数据，就将移位寄存器的值复制到接收缓冲器。此时，将发送缓冲器的高 8 位保存到接收缓冲器的高 8 位。能在发送时预先给 T31 ~ T24 写“0”，使接收缓冲器的高 8 位变为“0”。

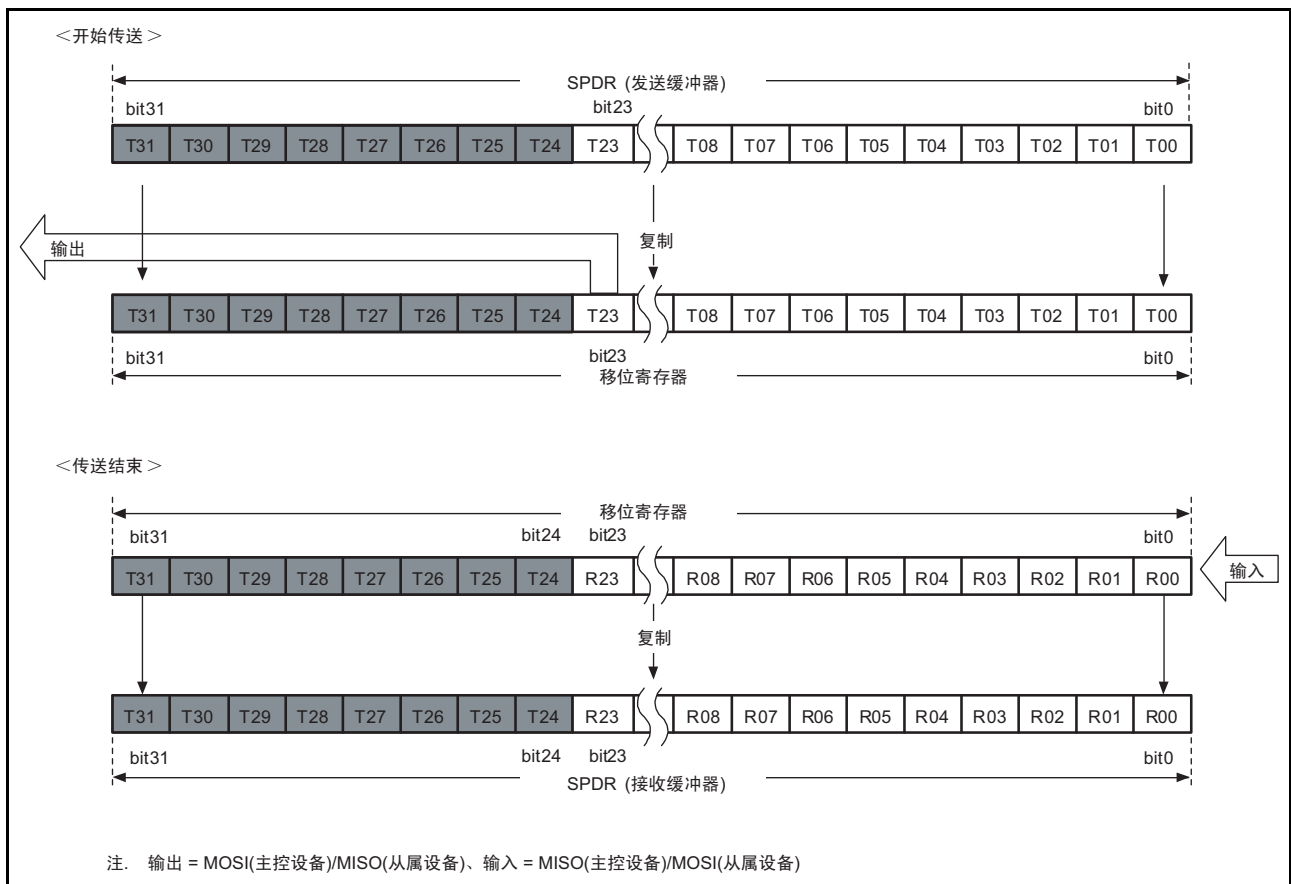


图 30.16 MSB first 传送 (24 位数据 / 奇偶校验功能无效)

(3) LSB first 传送 (32 位数据)

在奇偶校验功能无效的情况下，RSPI 进行 32 位数据长度的 LSB first 传送时的 RSPI 数据寄存器 (SPDR) 和移位寄存器的运行内容如图 30.17 所示。

在发送时，以位为单位将发送缓冲器的数据 (T31 ~ T00) 调换为 T00 ~ T31 的顺序，并且复制到移位寄存器。按 T00→T01→...→T31 的顺序将移位寄存器的值进行移位，并且发送数据。

在接收时，将最初的数据保存到移位寄存器的 bit0，并且按数据进行接收数据的移位。输入需要的 RSPCK，如果收集到 R00 ~ R31 的数据，就将移位寄存器的值复制到接收缓冲器。

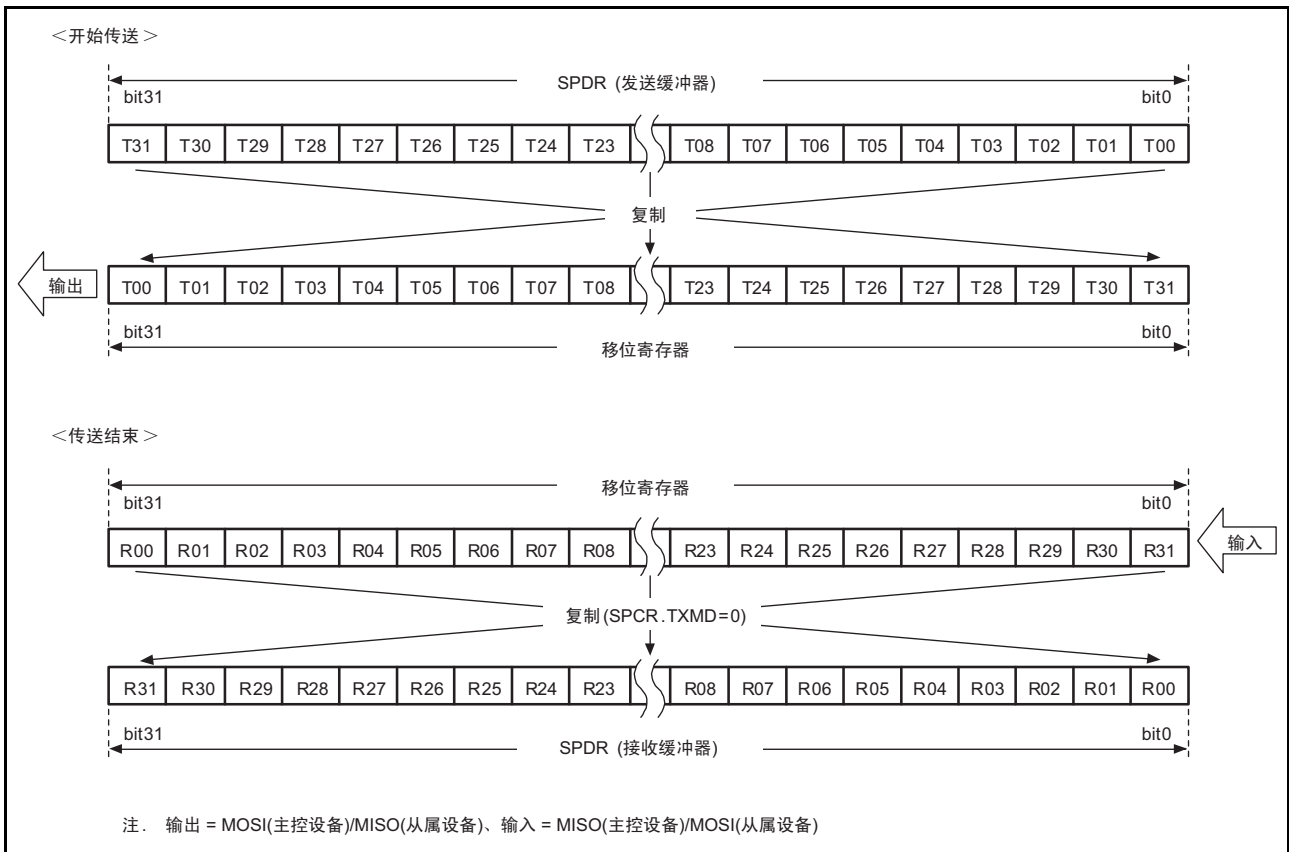


图 30.17 LSB first 传送 (32 位数据 / 奇偶校验功能无效)

(4) LSB first 传送 (24 位数据)

在奇偶校验功能无效的情况下, 以对 32 位以外的数据进行 LSB first 传送为例, RSPI 进行 24 位数据传送时的 RSPI 数据寄存器 (SPDR) 和移位寄存器的运行内容如图 30.18 所示。

在发送时, 以位为单位将发送缓冲器的低 24 位 (T23 ~ T00) 调换为 T00 ~ T23 的顺序, 并且复制到移位寄存器。按 T00→T01→…→T23 的顺序将移位寄存器的值进行移位, 并且发送数据。

在接收时, 将接收数据保存到移位寄存器的 bit8, 并且按数据进行接收数据的移位。输入需要的 RSPCK, 如果收集到 R00 ~ R23 的数据, 就将移位寄存器的值复制到接收缓冲器。

此时, 将发送缓冲器的高 8 位保存到接收缓冲器的高 8 位。能在发送时预先给 T31 ~ T24 写“0”, 使接收缓冲器的高 8 位变为“0”。

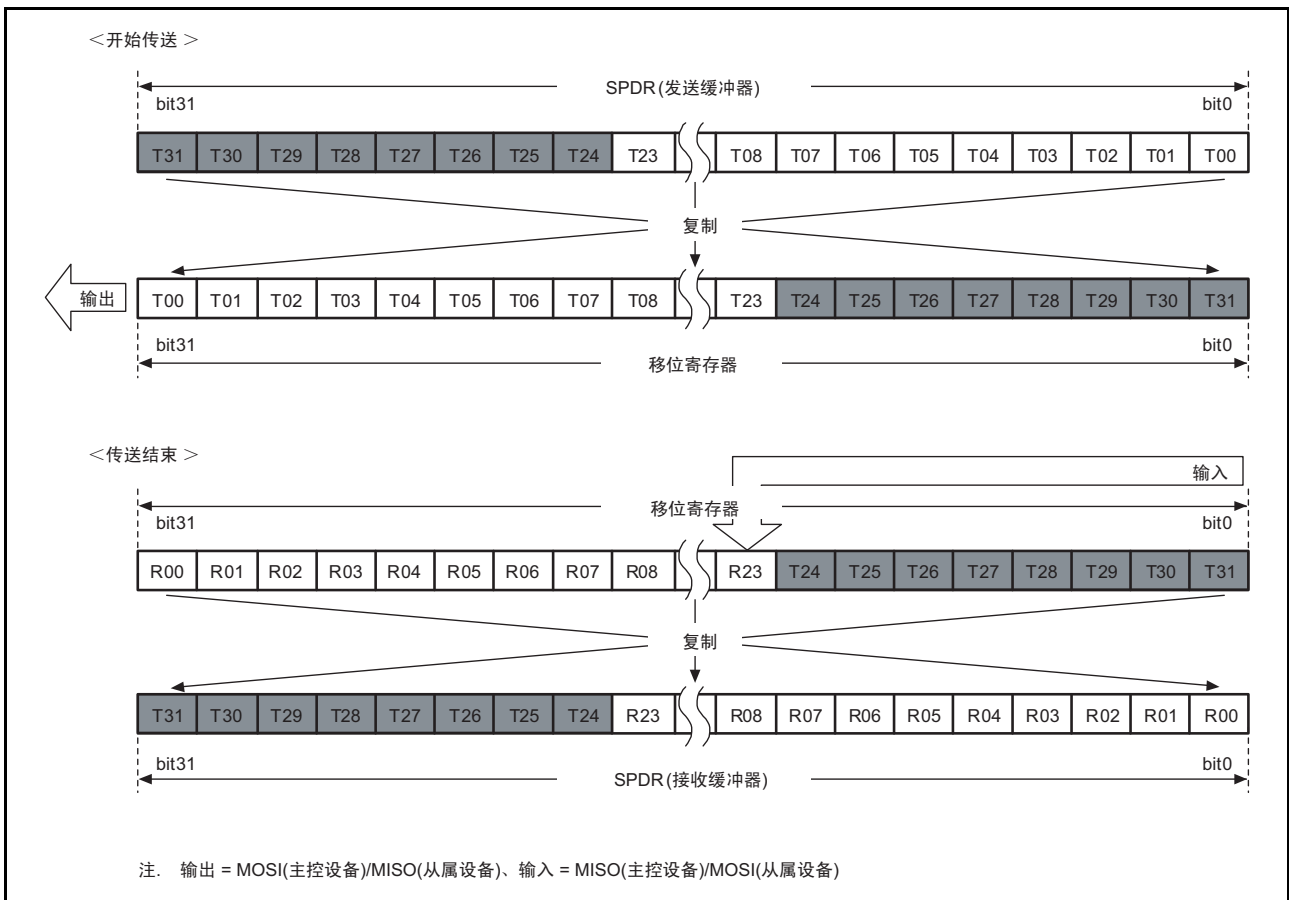


图 30.18 LSB first 传送 (24 位数据 / 奇偶校验功能无效)

30.3.4.2 奇偶校验功能有效的情况 (SPCR2.SPPE=1)

在奇偶校验功能有效时，将发送和接收数据的最低位转换为奇偶校验位。奇偶校验位的值由硬件进行计算和转换。

(1) MSB first 传送 (32 位数据)

在奇偶校验功能有效的情况下，RSPI 进行 32 位数据长度的 MSB first 传送时的 RSPI 数据寄存器 (SPDR) 和移位寄存器的运行内容如图 30.19 所示。

在发送时，先根据 T31 ~ T01 的数据进行奇偶校验位 (P) 的运算，替换最后位的 T00，然后复制到移位寄存器。按 T31→T30→...T01→P 的顺序发送数据。

在接收时，将接收数据保存到移位寄存器的 bit0，并且按数据进行接收数据的移位。输入需要的 RSPCK，如果收集到 R31 ~ P 的数据，就将移位寄存器的值复制到接收缓冲器。如果将数据复制到移位寄存器，就检查 R31 ~ P 的数据，判断奇偶校验错误。

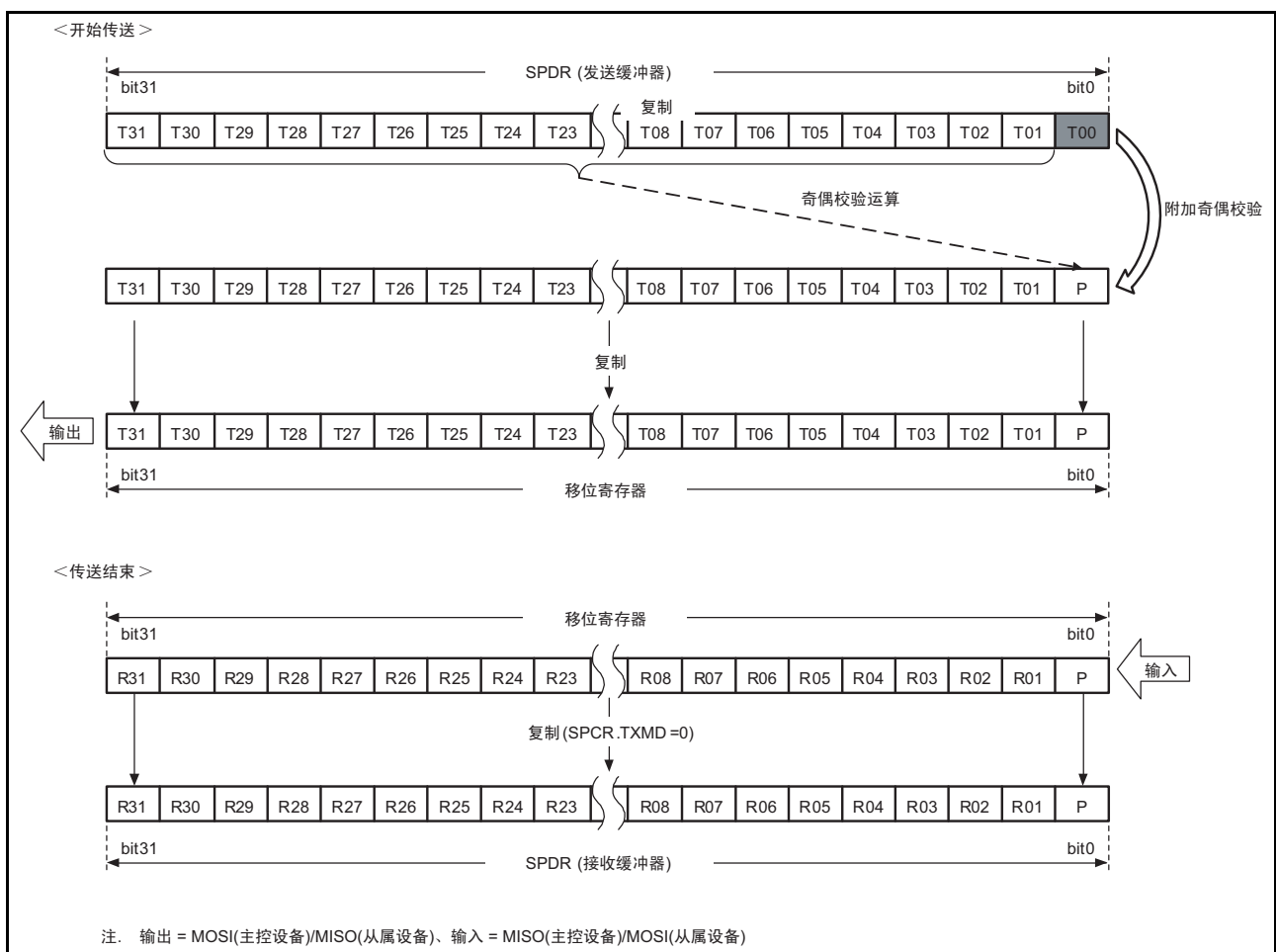


图 30.19 MSB first 传送 (32 位数据 / 奇偶校验功能有效)

(2) MSB first 传送 (24 位数据)

在奇偶校验功能有效的情况下，以对 32 位以外的数据进行 MSB first 传送为例，RSPI 进行 24 位数据传送时的 RSPI 数据寄存器 (SPDR) 和移位寄存器的运行内容如图 30.20 所示。

在发送时，先根据 T23 ~ T01 的数据进行奇偶校验位 (P) 的运算，替换最后位的 T00，然后复制到移位寄存器。按 T23→T22→...T01→P 的顺序发送数据。

在接收时，将接收数据保存到移位寄存器的 bit0，并且按数据进行接收数据的移位。输入需要的 RSPCK，如果收集到 R23 ~ P 的数据，就将移位寄存器的值复制到接收缓冲器。如果将数据复制到移位寄存器，就检查 R23 ~ P 的数据，判断奇偶校验错误。此时，将发送缓冲器的高 8 位保存到接收缓冲器的高 8 位。能在发送时预先给 T31 ~ T24 写“0”，使接收缓冲器的高 8 位变为“0”。

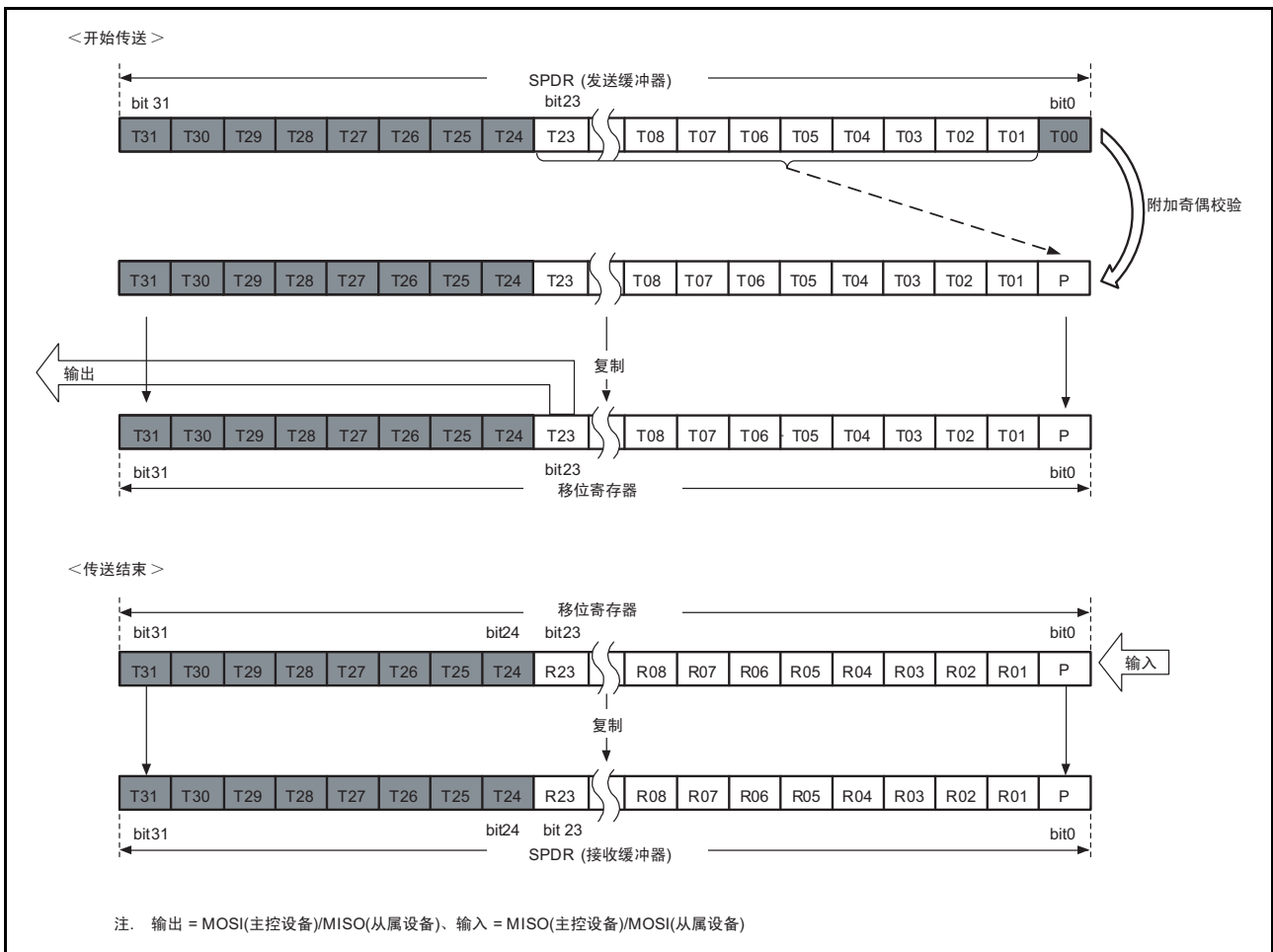


图 30.20 MSB first 传送 (24 位数据 / 奇偶校验功能有效)

(3) LSB first 传送 (32 位数据)

在奇偶校验功能有效的情况下, RSPI 进行 32 位数据长度的 LSB first 传送时的 RSPI 数据寄存器 (SPDR) 和移位寄存器的运行内容如图 30.21 所示。

在发送时, 先根据 T30 ~ T00 的数据进行奇偶校验位 (P) 的运算, 替换最后位的 T31, 然后复制到移位寄存器。按 T00→T01→…T30→P 的顺序发送数据。

在接收时, 将接收数据保存到移位寄存器的 bit0, 并且按数据进行接收数据的移位。输入需要的 RSPCK, 如果收集到 R00 ~ P 的数据, 就将移位寄存器的值复制到接收缓冲器。如果将数据复制到移位寄存器, 就检查 R00 ~ P 的数据, 判断奇偶校验错误。

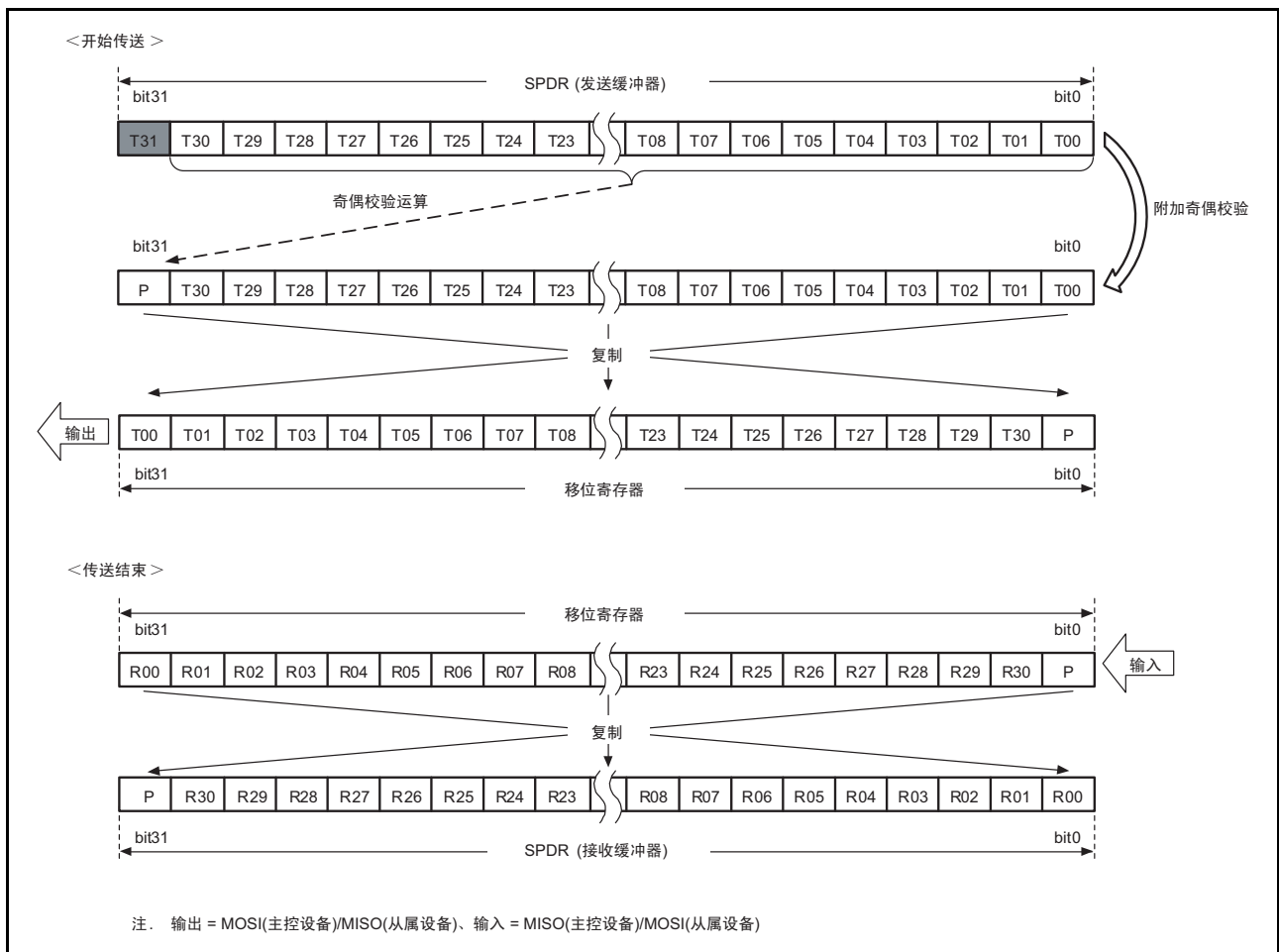


图 30.21 LSB first 传送 (32 位数据 / 奇偶校验功能有效)

(4) LSB first 传送 (24 位数据)

在奇偶校验功能有效的情况下, 以对 32 位以外的数据进行 LSB first 传送为例, RSPI 进行 24 位数据传送时的 RSPI 数据寄存器 (SPDR) 和移位寄存器的运行内容如图 30.22 所示。

在发送时, 先根据 T22 ~ T00 的数据进行奇偶校验位 (P) 的运算, 替换最后位的 T23, 然后复制到移位寄存器。按 T00→T01→…T22→P 的顺序发送数据。

在接收时, 将接收数据保存到移位寄存器的 bit8, 并且按数据进行接收数据的移位。输入需要的 RSPCK, 如果收集到 R23 ~ P 的数据, 就将移位寄存器的值复制到接收缓冲器。如果将数据复制到移位寄存器, 就检查 R00 ~ P 的数据, 判断奇偶校验错误。此时, 将发送缓冲器的高 8 位保存到接收缓冲器的高 8 位。能在发送时预先给 T31 ~ T24 写“0”, 使接收缓冲器的高 8 位变为“0”。

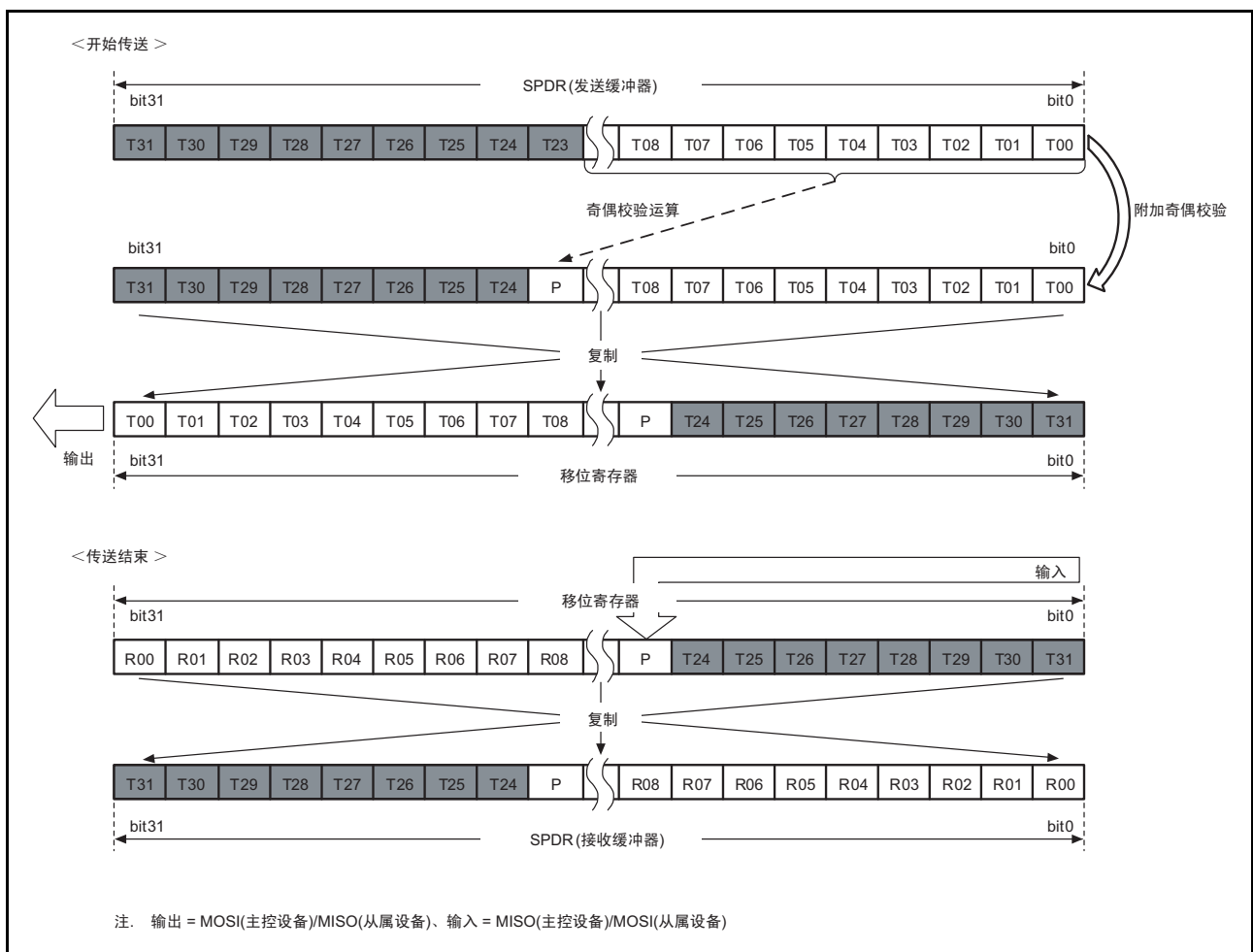


图 30.22 LSB first 传送 (24 位数据 / 奇偶校验功能有效)

30.3.5 传送格式

30.3.5.1 CPHA 位为“0”的情况

在 SPCMDm.CPHA 位为“0”的情况下，对 8 位数据进行串行传送时的传送格式例子如图 30.23 所示。但是，不保证 RSPI 为从属模式 (SPCR.MSTR=0) 并且 CPHA 位为“0”时的时钟同步运行 (SPCR.SPMS 位为“1”)。在图 30.23 中，RSPCKA (CPOL=0) 是 SPCMDm.CPOL 位为“0”时的 RSPCKA 信号波形，RSPCKA (CPOL=1) 是 SPCMDm.CPOL 位为“1”时的 RSPCKA 信号波形。采样时序表示 RSPI 将串行传送数据取到移位寄存器的时序。各信号的输入/输出方向取决于 RSPI 的设定，详细内容请参照“30.3.2 RSPI 引脚的控制”。

当 SPCMDm.CPHA 位为“0”时，在 SSLAi 信号的有效时序开始驱动 MOSIA 信号和 MISOA 信号的有效数据。在 SSLAi 信号有效后发生的第 1 个 RSPCKA 信号变化时序为第 1 个传送数据的取时序，在此时序之后，按每个 RSPCK 周期对数据进行采样。MOSIA 信号和 MISOA 信号的变化时序总是在传送数据取时序的 1/2 个 RSPCK 周期之后。CPOL 位的设定值不影响 RSPCK 信号的运行时序而只影响信号的极性。

t1 表示从 SSLAi 信号有效到 RSPCKA 振荡的期间 (RSPCK 延迟)，t2 表示从 RSPCKA 振荡停止到 SSLAi 信号无效的期间 (SSL 无效延迟)，t3 表示在串行传送结束后抑制下次传送的 SSLAi 信号有效的期间 (下次存取延迟)。t1、t2 和 t3 由 RSPI 系统上的主控设备进行控制。有关本 LSI 的 RSPI 为主控模式时的 t1、t2 和 t3，请参照“30.3.10.1 主控模式的运行”。

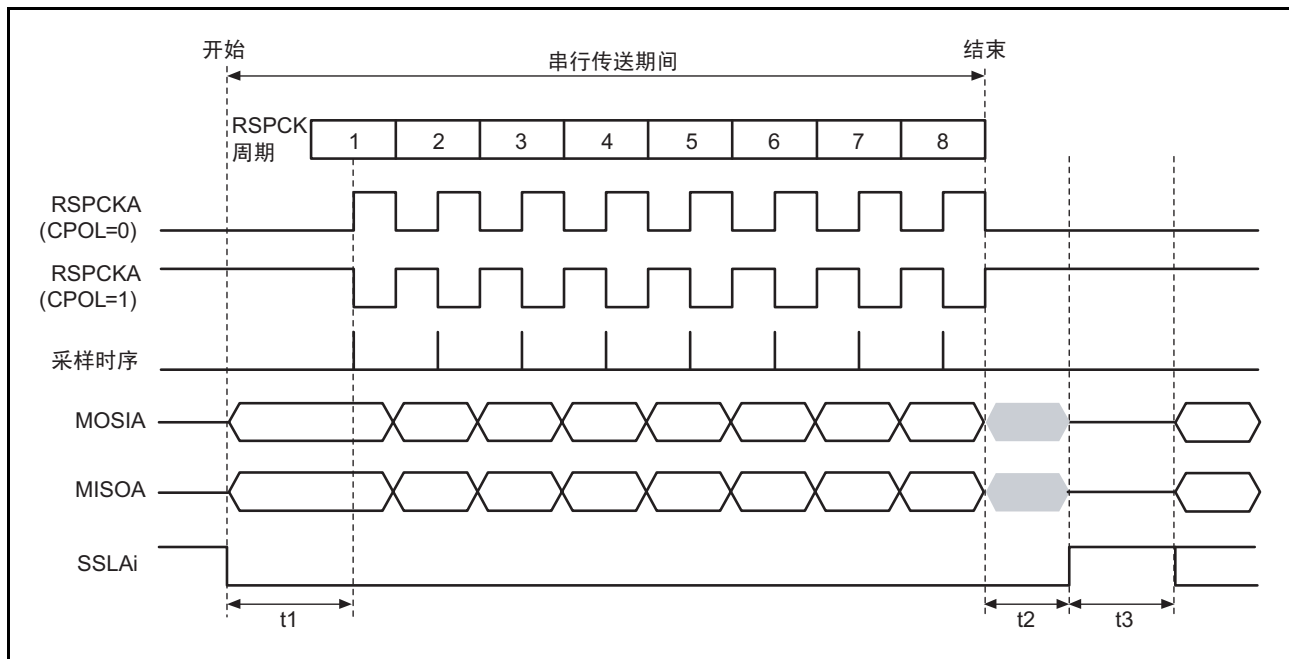


图 30.23 RSPI 传送格式 (CPHA 位 =0)

30.3.5.2 CPHA 位为“1”的情况

在 SPCMDm.CPHA 位为“1”的情况下，对 8 位数据进行串行传送时的传送格式例子如图 30.24 所示。但是，当 SPCR.SPMS 位为“1”时，不使用 SSLAi 信号而只使用 RSPCKA 信号、MOSIA 信号和 MISOA 信号进行通信。在图 30.24 中，RSPCK (CPOL=0) 是 SPCMDm.CPOL 位为“0”时的 RSPCKA 信号波形，RSPCK (CPOL=1) 是 SPCMDm.CPOL 位为“1”的 RSPCKA 信号波形。采样时序表示 RSPI 将串行传送数据取到移位寄存器的时序。各信号的输入/输出方向取决于 RSPI 的模式（主控/从属模式），详细内容请参照“30.3.2 RSPI 引脚的控制”。

当 SPCMDm.CPHA 位为“1”时，在 SSLAi 信号的有效时序开始驱动 MISOA 信号的无效数据。在 SSLAi 信号有效后发生的第 1 个 RSPCKA 信号变化时序，开始输出 MOSIA 信号和 MISOA 信号的有效数据。在此时序后，按每个 RSPCK 周期更新数据。传送数据的取时序总是在此时序的 1/2 个 RSPCK 周期之后。SPCMDm.CPOL 位的设定值不影响 RSPCKA 信号的运行时序而只影响信号的极性。

t1、t2、t3 的内容和 CPHA 位为“0”的情况相同。有关本 LSI 的 RSPI 为主控模式时的 t1、t2 和 t3，请参照“30.3.10.1 主控模式的运行”。

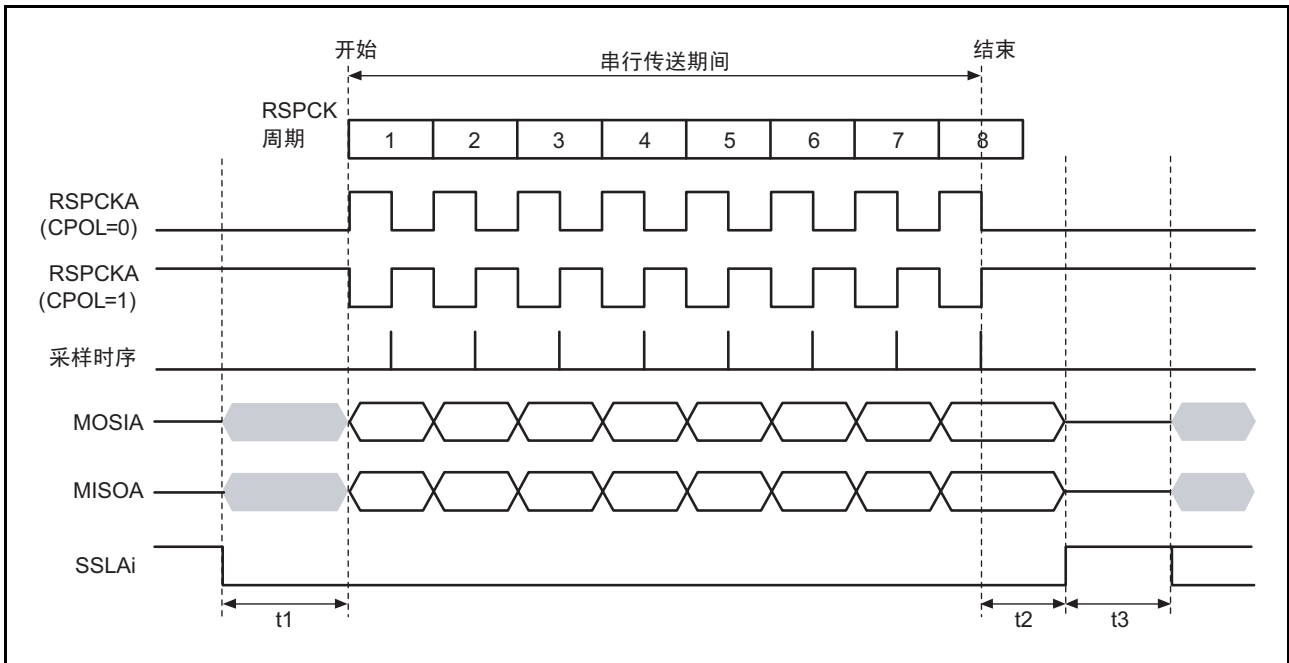


图 30.24 RSPI 传送格式 (CPHA 位 =1)

30.3.6 通信运行模式

通过设定 SPCR.TXMD 位，选择是进行全双工同步串行通信还是只进行发送的串行通信。图 30.25 和图 30.26 记载的“SPDR 存取”表示 SPDR 寄存器的存取状况，“W”表示写周期。

30.3.6.1 全双工同步串行通信 (SPCR.TXMD=0)

SPCR.TXMD 位为“0”时的运行例子如图 30.25 所示。在图 30.25 的例子中，SPDCR.SPFC[1:0] 位为“00b”，SPCMDm.CPHA 位为“1”并且 SPCMDm.CPOL 位为“0”，RSPI 进行 8 位串行传送。RSPCKA 波形下面记载的数字表示 RSPCK 周期数 (= 传送位数)。

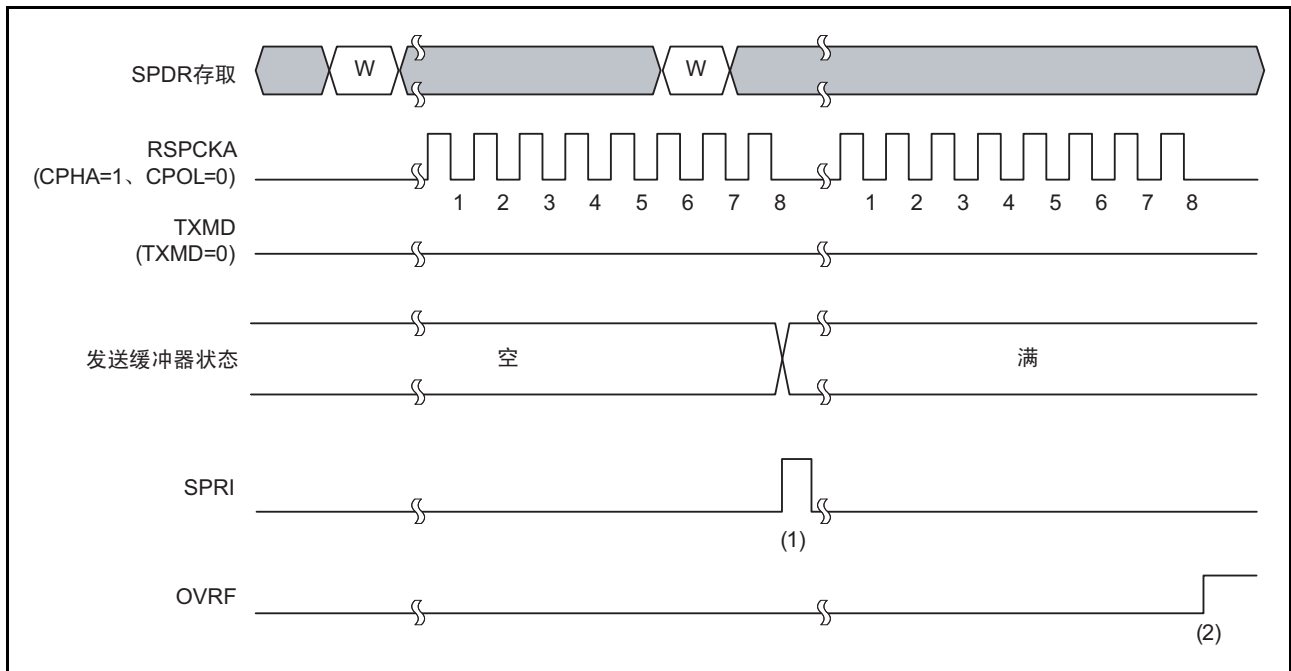


图 30.25 SPCR.TXMD 为“0”时的运行例子

以下说明在图中的 (1) 和 (2) 所示的时序中标志的运行内容：

- (1) 如果在 SPDR 寄存器的接收缓冲器为空的状态下结束串行传送，RSPI 就在产生接收缓冲器满中断请求 (SPRI) 后，将移位寄存器的接收数据复制到接收缓冲器。
- (2) 如果在 SPDR 寄存器的接收缓冲器有以前的接收数据的状态下结束串行传送，RSPI 就在将 SPSR.OVRF 标志置“1”后，放弃移位寄存器的接收数据。

在进行全双工同步串行通信时 (SPCR.TXMD=0)，发送数据以及接收数据。因此，在 (1) 和 (2) 各自的时序中，SPSR.OVRF 标志变为“1”。

30.3.6.2 只进行发送的运行 (SPCR.TXMD=1)

SPCR.TXMD 位为“1”时的运行例子如图 30.26 所示。在图 30.26 的例子中，SPDCR.SPFC[1:0] 位为“00b”，SPCMDm.CPHA 位为“1”并且 SPCMDm.CPOL 位为“0”，RSPI 进行 8 位串行传送。RSPCKA 波形下面记载的数字表示 RSPCK 周期数 (= 传送位数)。

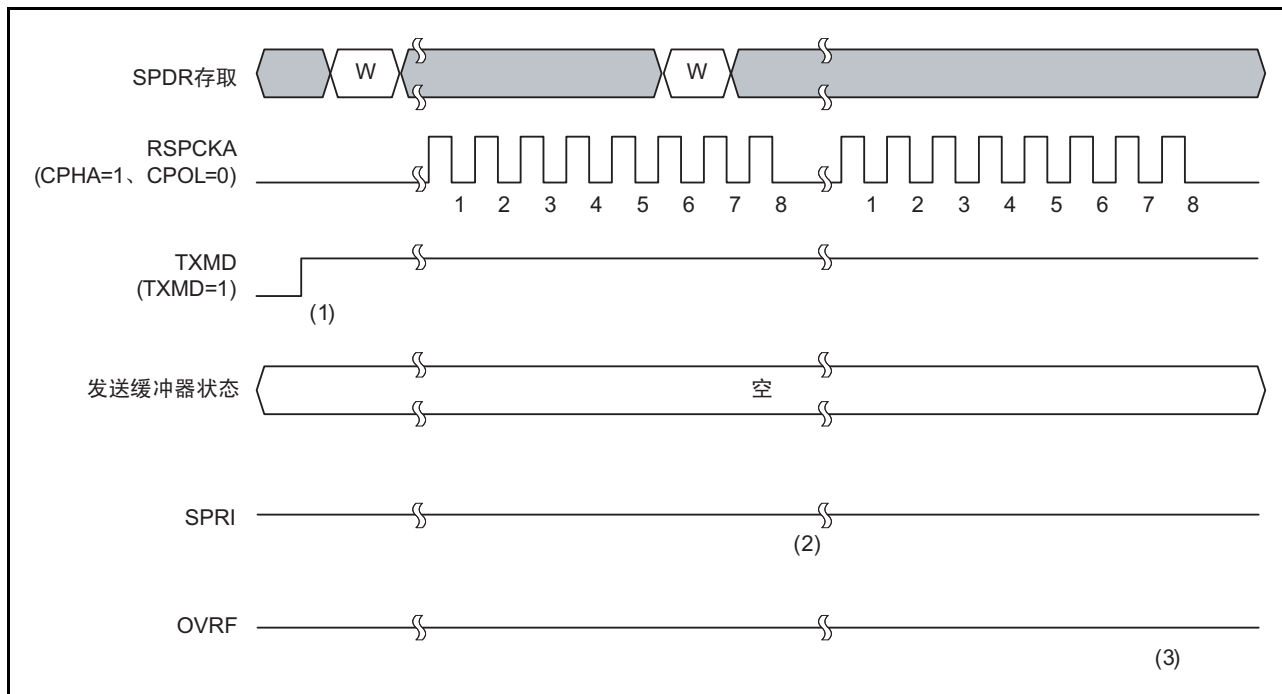


图 30.26 SPCR.TXMD 为“1”时的运行例子

以下说明在图中的 (1)、(2)、(3) 所示的时序中标志的运行内容:

- (1) 必须在确认接收缓冲器没有数据并且 SPSR.OVRF 标志是“0”后转移到只进行发送的运行 (SPCR.TXMD=1)。
- (2) 如果在 SPDR 寄存器的接收缓冲器为空的状态下结束串行传送，就在只进行发送的运行 (SPCR.TXMD=1) 时不将移位寄存器的数据复制到接收缓冲器。
- (3) 因为 SPDR 寄存器的接收缓冲器没有以前的接收数据，所以即使结束串行传送 SPSR.OVRF 标志也保持“0”，并且不将移位寄存器的数据复制到接收缓冲器。

在只进行发送的运行 (SPCR.TXMD=1)，发送数据而不接收数据。因此，在 (1)、(2)、(3) 各自的时序中，SPSR.OVRF 标志保持“0”。

30.3.7 发送缓冲器空中断 / 接收缓冲器满中断

RSPI 发送缓冲器空中断 (SPTI) 和 RSPI 接收缓冲器满中断 (SPRI) 的运行例子如图 30.27 所示。图 30.27 中记载的“SPDR 存取”表示 SPDR 寄存器的存取状况，“W”表示写周期，“R”表示读周期。在图 30.27 的例子中，SPCR.TXMD 位为“0”，SPDCR.SPFC[1:0] 位为“00b”，SPCMDm.CPHA 位为“1”并且 SPCMDm.CPOL 为“0”，RSPI 进行 8 位串行传送。RSPCKA 波形下面记载的数字表示 RSPCK 周期数 (= 传送位数)。

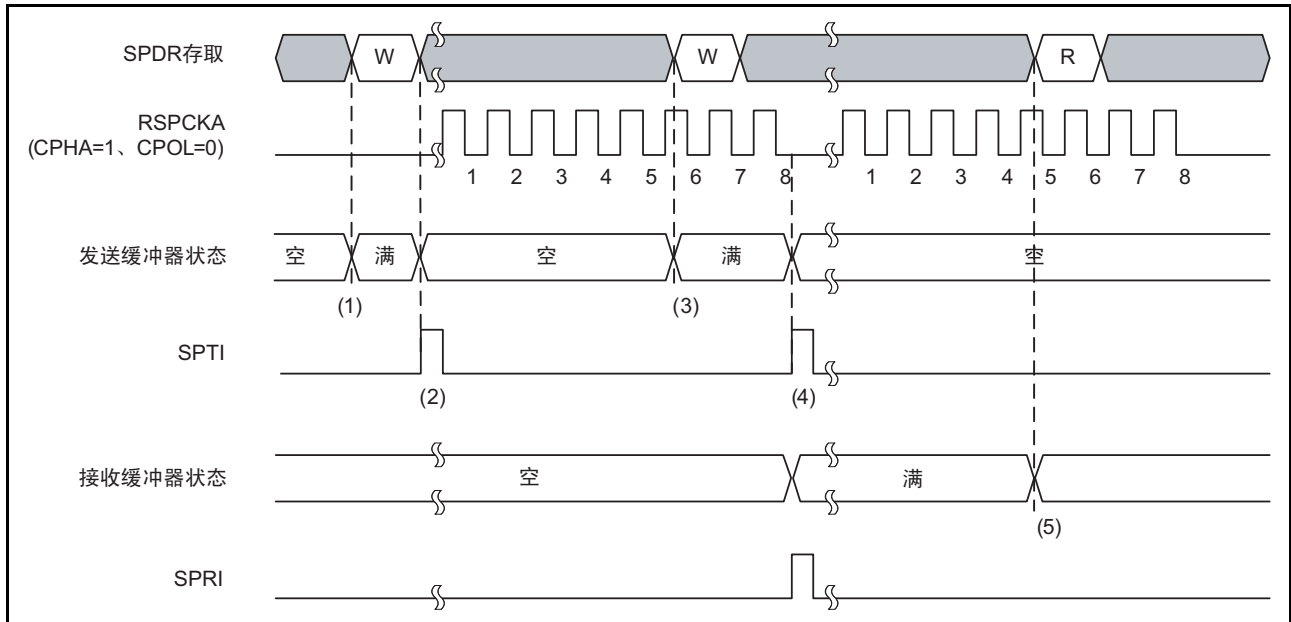


图 30.27 SPTI 中断和 SPRI 中断的运行例子

以下说明在图中的 (1) ~ (5) 所示的时序中中断的运行内容：

- (1) 在 SPDR 寄存器的发送缓冲器为空的（未设定下次传送的数据）状态下，如果将发送数据写到 SPDR 寄存器，RSPI 就将数据写到发送缓冲器。
- (2) 如果移位寄存器为空，就在 RSPI 将发送缓冲器的数据复制到移位寄存器后产生发送缓冲器空中断（SPTI）。串行传送的开始方法取决于 RSPI 的模式（参照“30.3.10 SPI 运行”和“30.3.11 时钟同步运行”）。
- (3) 如果通过发送缓冲器空中断程序将发送数据写到 SPDR 寄存器，就将数据传送到发送缓冲器。因为串行传送过程中的数据被保存在移位寄存器，所以 RSPI 不将发送缓冲器的数据复制到移位寄存器。
- (4) 如果在 SPDR 寄存器的接收缓冲器为空的状态下结束串行传送，RSPI 就将移位寄存器的接收数据复制到接收缓冲器并且产生接收缓冲器满中断请求（SPRI）。一旦串行传送结束，移位寄存器就变空。因此，如果在串行传送结束前发送缓冲器满，RSPI 就将发送缓冲器的数据复制到移位寄存器。即使在发生溢出错误的状态下不将接收数据从移位寄存器复制到接收缓冲器，只要串行传送结束，RSPI 也判断为移位寄存器处于空的状态，变为可将数据从发送缓冲器传送到移位寄存器的状态。
- (5) 如果通过接收缓冲器满中断程序读 SPDR 寄存器，就能读到接收数据。

如果在发送缓冲器有未发送数据的状态下写 SPDR 寄存器，RSPI 就不更新发送缓冲器的数据。必须通过发送缓冲器空中断程序来写 SPDR 寄存器。在使用 RSPI 发送中断时，必须将 SPCR 的 SPTIE 位置“1”。

在 RSPI 功能无效（SPCR 的 SPE 位为“0”）状态下，必须将 SPTIE 位置“0”。

如果在接收缓冲器满的状态下结束串行传送，RSPI 就不将数据从移位寄存器复制到接收缓冲器，而检测到溢出错误（参照“30.3.8 错误检测”）。为了防止接收数据的溢出，必须在下次串行传送结束前，通过接收缓冲器满中断请求读接收数据。在使用 RSPI 接收中断时，必须将 SPCR.SPRIE 位置“1”。

能通过发送 / 接收中断或者对应 ICU 的 IRn.IR 标志（n = 中断向量号）确认发送 / 接收缓冲器的状态。有关中断向量号，请参照“14. 中断控制器（ICUb）”。

30.3.8 错误检测

在正常的 RSPI 串行传送中，能对写在 SPDR 寄存器的发送缓冲器中的数据进行串行发送，并且能从 SPDR 寄存器的接收缓冲器读串行接收的数据。根据存取 SPDR 寄存器时的发送 / 接收缓冲器的状态以及串行传送开始 / 结束时的 RSPI 的状态，有可能发生异常传送。

如果发生部分异常传送，RSPI 就检测为溢出错误、奇偶校验错误或者模式故障错误。异常传送和 RSPI 错误检测功能的关系如表 30.8 所示。

表 30.8 异常传送的产生条件和 RSPI 的错误检测功能

	发生条件	RSPI 运行	错误检测
A	在发送缓冲器满的状态下写 SPDR 寄存器。	<ul style="list-style-type: none"> 保持发送缓冲器内容。 写数据丢失。 	无
B	在从属模式中未将发送数据设定到移位寄存器的状态下开始串行传送。	对上次串行传送时的接收数据进行串行发送。	无
C	在接收缓冲器为空的状态下读 SPDR 寄存器。	输出上次的串行接收数据。	无
D	在接收缓冲器满的状态下结束串行传送。	保持接收缓冲器内容并且串行接收数据丢失。	溢出错误检测
E	在进行全双工同步串行通信并且奇偶校验功能有效的状态下接收到错误的奇偶校验位。	奇偶校验错误标志有效。	奇偶校验错误检测
F	在多主控模式的串行传送空闲时 SSLA0 输入信号有效。	<ul style="list-style-type: none"> 停止驱动 RSPCKA、MOSIA、SSLA1 ~ 3 输出信号。 RSPI 功能无效。 	模式故障错误检测
G	在多主控模式的串行传送过程中 SSLA0 输入信号有效。	<ul style="list-style-type: none"> 中止串行传送。 发送 / 接收数据丢失。 停止驱动 RSPCKA、MOSIA、SSLA1 ~ 3 输出信号。 RSPI 功能无效。 	模式故障错误检测
H	在从属模式的串行传送过程中 SSLA0 输入信号无效。	<ul style="list-style-type: none"> 中止串行传送。 发送 / 接收数据丢失。 停止驱动 MISOA 输出信号。 RSPI 功能无效。 	模式故障错误检测

对于表 30.8 的 A 所示的运行，RSPI 不检测错误。为了在写 SPDR 寄存器时不丢失数据，必须通过发送中断请求来写 SPDR 寄存器。

即使对于 B 所示的运行，RSPI 也不检测错误。在更新移位寄存器前启动的串行传送中，RSPI 发送上次串行传送时的接收数据，而不将 B 所示的运行作为错误进行处理。因为上次串行传送时的接收数据保存在 SPDR 寄存器的接收缓冲器，所以能正常读取（如果不在串行传送结束前读 SPDR 寄存器，就发生溢出错误）。

即使对于 C 所示的运行，RSPI 也不检测错误。为了只读需要的数据，必须通过接收中断来读 SPDR 寄存器。

有关 D 所示的溢出错误，请参照“30.3.8.1 溢出错误”；有关 E 所示的奇偶校验错误，请参照“30.3.8.2 奇偶校验错误”；有关 F ~ H 所示的模式故障错误，请参照“30.3.8.3 模式故障错误”。

有关发送中断和接收中断，请参照“30.3.7 发送缓冲器空中断 / 接收缓冲器满中断”。

30.3.8.1 溢出错误

如果在 SPDR 寄存器的接收缓冲器满的状态下结束串行传送，RSPI 就检测到溢出错误，然后将 SPSR.OVRF 标志置“1”。因为在 OVRF 标志为“1”的状态下 RSPI 不将移位寄存器的数据复制到接收缓冲器，所以发生错误前的数据保存在接收缓冲器。要将 OVRF 标志置“0”时，需要在 OVRF 标志为“1”的状态下读 SPSR 寄存器后给 OVRF 标志写“0”。

SPSR.OVRF 标志的运行如图 30.28 所示。图 30.28 中记载的 SPSR 存取和 SPDR 存取分别表示 SPSR 寄存器和 SPDR 寄存器的存取状况，“W”表示写周期，“R”表示读周期。在图 30.28 的例子中，SPCMDm.CPHA 位为“1”并且 SPCMDm.CPOL 位为“0”，RSPI 进行 8 位串行传送。RSPCKA 波形下面记载的数字表示 RSPCK 周期数 (= 传送位数)。

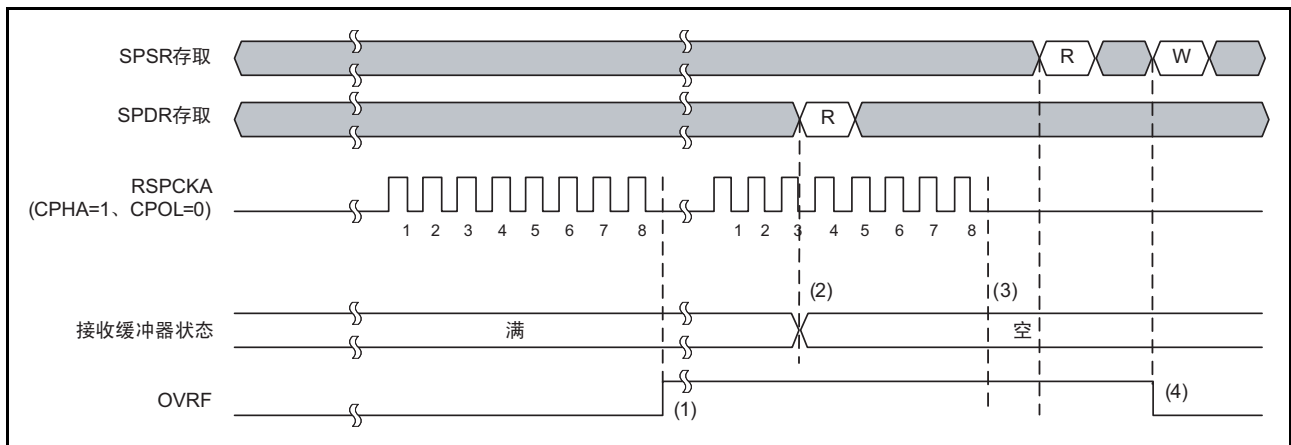


图 30.28 OVRF 标志的运行例子

以下说明在图中的 (1) ~ (4) 所示的时序中标志的运行内容：

- (1) 如果在接收缓冲器满的状态下结束串行传送，RSPI 就检测到溢出错误并且将 OVRF 标志置“1”。RSPI 不将移位寄存器的数据复制到接收缓冲器。即使 SPPE 位为“1”时，也不检测奇偶校验错误。在主控模式中，将指向 SPCMDm 寄存器的指针值复制到 SPSR.SPECM[2:0] 位。
- (2) 如果读 SPDR，RSPI 就能读到接收缓冲器的数据。即使接收缓冲器为空，OVRF 标志也不变为“0”。
- (3) 如果在 OVRF 标志为“1”的状态（溢出错误）下结束串行传送，RSPI 就不将移位寄存器的数据复制到接收缓冲器，也不产生接收缓冲器满中断。即使 SPPE 位为“1”，也不检测奇偶校验错误。在主控模式中，RSPI 不更新 SPECM[2:0] 位。在发生溢出错误的状态下，如果在没有将接收数据从移位寄存器复制到接收缓冲器的情况下结束串行传送，RSPI 就判断为移位寄存器处于空的状态，变为可将数据从发送缓冲器传送到移位寄存器的状态。
- (4) 如果在 OVRF 标志为“1”的状态下读 SPSR 寄存器后给 OVRF 标志写“0”，RSPI 就将 OVRF 标志置“0”。

能通过读 SPSR 寄存器或者通过 RSPI 错误中断并且读 SPSR 寄存器，确认溢出的发生。在进行串行传送时，必须通过在读 SPDR 寄存器后立即读 SPSR 寄存器等方法，尽早检测到溢出错误的发生。在主控模式中使用 RSPI 时，能通过读 SPSR.SPECM[2:0] 位，确认发生错误时的指向 SPCMDm 寄存器的指针值。

如果在发生溢出错误后 OVRF 标志变为“1”，就在 OVRF 标志变为“0”前无法进行正常的接收运行。OVRF 标志为“0”的条件如下：

- 在 OVRF 标志为“1”的状态下读 SPSR 寄存器后给 OVRF 标志写“0”。

30.3.8.2 奇偶校验错误

在 SPCR.TXMD 位为“0”并且 SPCR2.SPPE 位为“1”的状态下，如果进行全双工同步串行通信并且结束传送，就判断为奇偶校验错误。当 RSPI 检测到接收数据有奇偶校验错误时，将 SPSR.PERF 标志置“1”。在 SPSR.OVRF 位为“1”的状态下，因为 RSPI 不将移位寄存器的数据复制到接收缓冲器，所以不对接收数据进行奇偶校验错误的检测。要将 PERF 标志置“0”时，需要在 PERF 标志为“1”的状态下读 SPSR 寄存器后给 PERF 标志写“0”。

OVRF 标志和 PERF 标志的运行如图 30.29 所示。图 30.29 中记载的“SPSR 存取”表示 SPSR 寄存器的存取状况，“W”表示写周期，“R”表示读周期。在图 30.29 的例子中，在 SPCR.TXMD 位为“0”并且 SPCR2.SPPE 位为“1”的状态下进行全双工同步串行通信。通过 SPCMDm.CPHA 位为“1”并且 SPCMDm.CPOL 位为“0”的设定，RSPI 进行 8 位串行传送。RSPCKA 波形下面记载的数字表示 RSPCK 周期数 (= 传送位数)。

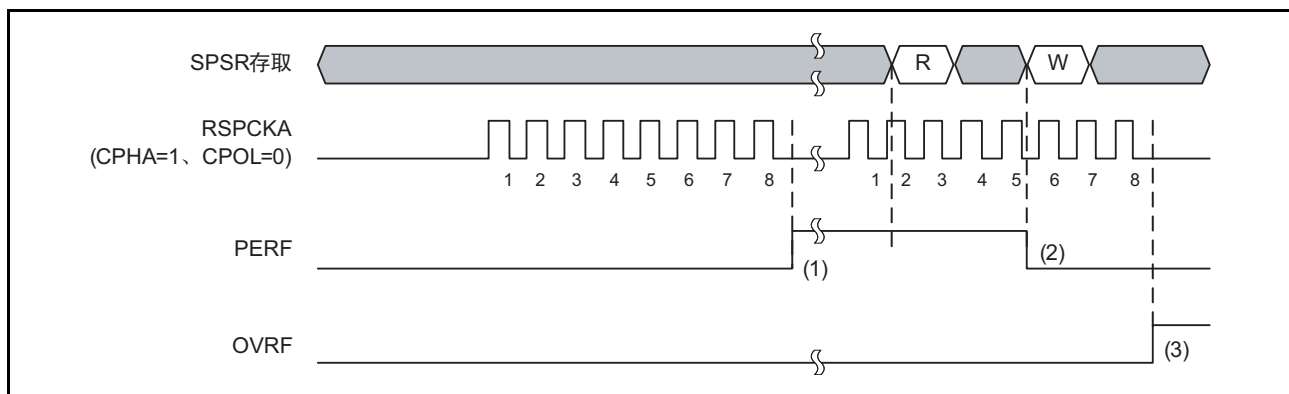


图 30.29 PERF 标志的运行例子

以下说明在图中 (1) ~ (3) 所示的时序中标志的运行内容：

- (1) 如果 RSPI 没有检测到溢出错误而结束串行传送，就将移位寄存器的数据复制到接收缓冲器。此时，RSPI 判断接收数据，如果检测到奇偶校验错误，就将 PERF 标志置“1”。在主控模式中，将指向 SPCMDm 寄存器的指针值复制到 SPSSR.SPECM[2:0] 位。
- (2) 如果在 PERF 标志为“1”的状态下读 SPSR 寄存器后给 PERF 标志写“0”，就将 OVRF 标志置“0”。
- (3) 如果 RSPI 检测到溢出错误并且结束串行传送，就将移位寄存器的数据复制到接收缓冲器。此时，RSPI 不检测奇偶校验错误。

能通过读 SPSR 寄存器或者通过 RSPI 错误中断并且读 SPSR 寄存器，确认奇偶校验错误的发生。在进行串行传送时，必须通过读 SPSR 等方法，尽早检测到奇偶校验错误的发生。在主控模式中使用 RSPI 时，能通过读 SPSSR.SPECM[2:0] 位，确认发生错误时的指向 SPCMDm 寄存器的指针值。

PERF 标志为“0”的条件如下：

- 在 PERF 标志为“1”的状态下读 SPSR 寄存器后给 PERF 标志写“0”时

30.3.8.3 模式故障错误

当 SPCR.MSTR 位为“1”、SPCR.SPMS 位为“0”并且 SPCR.MODFEN 位为“1”时，RSPI 以多主控模式运行。如果给多主控模式的 RSPI 的 SSLA0 输入信号输入有效电平，RSPI 就检测到模式故障错误，然后将 SPSR.MODF 标志置“1”，与串行传送状态无关。如果检测到模式故障错误，RSPI 就将指向 SPCMDm 寄存器的指针值复制到 SPSSR.SPECM[2:0] 位。SSLA0 信号的有效电平取决于 SSLP.SSL0P。

当 MSTR 位为“0”时，RSPI 以从属模式运行。当从属模式的 RSPI 的 MODFEN 位为“1”并且 SPMS 位为“0”时，如果在串行传送期间（从开始驱动有效数据到取最后的有效数据前）使 SSLA0 输入信号无效，RSPI 就检测到模式故障错误。

如果 RSPI 检测到模式故障错误，就停止驱动输出信号并且清除 SPCR.SPE 位（参照“30.3.9 RSPI 的初始化”）。在多主控结构中，如果使用模式故障错误来驱动输出信号，就能停止 RSPI 功能并且释放主控权。

能通过读 SPSR 寄存器或者通过 RSPI 错误中断并且读 SPSR 寄存器，确认模式错误的发生。为了检测模式错误，需要轮询 SPSR 寄存器。在主控模式中使用 RSPI 时，能通过读 SPSSR.SPECM[2:0] 位，确认发生错误时的指向 SPCMDm 寄存器的指针值。

在 MODF 标志为“1”的状态下，忽视给 SPE 位的写“1”。为了使 RSPI 功能在检测到模式故障错误后有效，必须将 MODF 标志置“0”。将 MODF 标志清“0”的条件如下：

- 在 MODF 标志为“1”的状态下读 SPSR 寄存器后给 MODF 标志写“0”。

30.3.9 RSPI 的初始化

如果给 SPCR.SPE 位写“0”或者 RSPI 通过模式故障错误检测将 SPE 位置“0”，RSPI 就使 RSPI 功能无效并且对部分模块功能进行初始化。如果发生系统复位，RSPI 就对全部模块功能进行初始化。以下说明通过将 SPCR.SPE 位置“0”进行的初始化以及通过系统复位进行的初始化。

30.3.9.1 通过清除 SPE 位进行的初始化

当 SPCR.SPE 位为“0”时，RSPI 进行以下所示的初始化：

- 中止正在进行的串行传送。
- 在从属模式中停止驱动输出信号 (Hi-Z)。
- 对 RSPI 内部状态进行初始化。
- 将 RSPI 发送缓冲器清空。

在将 SPE 位置“0”进行初始化时，不对 RSPI 的控制位进行初始化。因此，只要重新将 SPE 位置“1”，就能在和初始化前相同的传送模式中启动 RSPI。

不对 SPSR.OVRF 和 SPSR.MODF 标志的值进行初始化，也不对 SPSSR 寄存器的值进行初始化。因此，在 RSPI 的初始化后，也能通过读接收缓冲器的数据，确认 RSPI 传送时的错误发生状况。

将发送缓冲器初始化为空状态。因此，如果在 RSPI 的初始化后将 SPCR.SPTIE 位置“1”，就产生 RSPI 发送中断。在通过 CPU 对 RSPI 进行初始化的情况下，为了禁止 RSPI 发送中断，必须在给 SPE 位写“0”的同时给 SPTIE 位写“0”。要在检测到模式故障错误后禁止 RSPI 发送中断时，必须通过错误处理程序给 SPTIE 位写“0”。

30.3.9.2 系统复位

在通过系统复位进行初始化时，除了“30.3.9.1 通过清除 SPE 位进行的初始化”记载的事项以外，还对用于 RSPI 控制的全部位、状态位和数据寄存器进行初始化，RSPI 完全被初始化。

30.3.10 SPI 运行

30.3.10.1 主控模式的运行

单主控模式运行和多主控模式运行的不同点在于模式故障错误检测（参照“30.3.8 错误检测”）。单主控模式的 RSPI 不检测模式故障错误，多主控模式的 RSPI 检测模式故障错误。本节说明单主控模式 / 多主控模式的通用运行。

(1) 串行传送的开始

在 RSPI 发送缓冲器为空（未设定下次传送的数据）的状态下，如果将数据写到 SPDR 寄存器，RSPI 就更新 SPDR 寄存器的发送缓冲器（SPTX）的数据。在将 SPDCR.SPFC[1:0] 位所设帧的数据写到 SPDR 寄存器后移位寄存器变空时，RSPI 在将发送缓冲器的数据复制到移位寄存器后开始串行传送。如果 RSPI 将发送数据复制到移位寄存器，就将移位寄存器状态改为满状态；如果串行传送结束，就将移位寄存器改为空状态。不能参照移位寄存器的状态。

有关 RSPI 传送格式的详细内容，请参照“30.3.4 数据格式”。SSLAi 输出引脚的极性取决于 SSLP 寄存器的设定值。

(2) 串行传送的结束

与 SPCMDm.CPHA 位无关，如果 RSPI 发送最后采样时序所对应的 RSPCKA 边沿，就结束串行传送。在接收缓冲器（SPRX）为空的情况下，在串行传送结束后将数据从移位寄存器复制到 SPDR 寄存器的接收缓冲器。

最后的采样时序取决于传送数据的位长，主控模式的 RSPI 数据长度取决于 SPCMDm.SPB[3:0] 位的设定值，SSLAi 输出引脚的极性取决于 SSLP 寄存器的设定值。有关 RSPI 传送格式的详细内容，请参照“30.3.5 传送格式”。

(3) 顺序控制

主控模式时的传送格式取决于 SPSCR 寄存器、SPCMDm 寄存器、SPBR 寄存器、SPCKD 寄存器、SSLND 寄存器和 SPND 寄存器。

SPSCR 寄存器决定主控模式的 RSPI 要执行的串行传送的顺序结构。给 SPCMDm 寄存器设定 SSLAi 引脚的输出信号值、MSB/LSB first、数据长度、部分位速率、RSPCK 极性 / 相位以及设定是否要参照 SPCKD 寄存器、SSLND 寄存器和 SPND 寄存器；给 SPBR 寄存器设定部分位速率，给 SPCKD 设定 RSPI 时钟延迟值，给 SSLND 寄存器设定 SSL 无效延迟并且给 SPND 寄存器设定 RSPI 的下次存取延迟值。

RSPI 根据 SPSCR 寄存器设定的顺序长度，由部分或者全部 SPCMDm 寄存器构成顺序。RSPI 有构成顺序的 SPCMDm 寄存器的对应指针。能通过读 SPSSR.SPCP[2:0] 位来确认此指针的值。如果通过将 SPCR.SPE 位置“1”来允许 RSPI 功能，RSPI 就将指向命令的指针设定到 SPCMD0 寄存器，并且在开始串行传送时将 SPCMD0 寄存器的设定内容反映到传送格式。每当各数据传送的下次存取延迟期间结束时，RSPI 将指针进行递增。当构成顺序的最后命令对应的串行传送结束时，RSPI 将指针设定到 SPCMD0 寄存器，重复执行顺序。

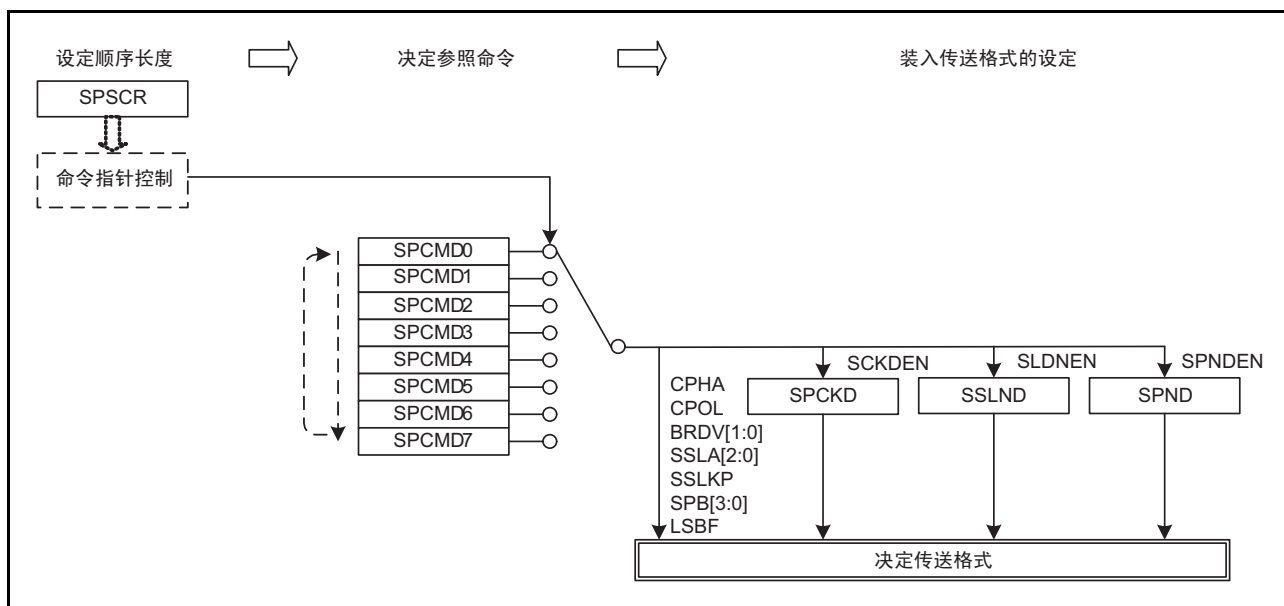


图 30.30 主控模式中的串行传送方式的决定方法 (SPI 运行)

在本章节中，将数据 (SPDR) 和设定 (SPCMDm) 合成 1 帧。

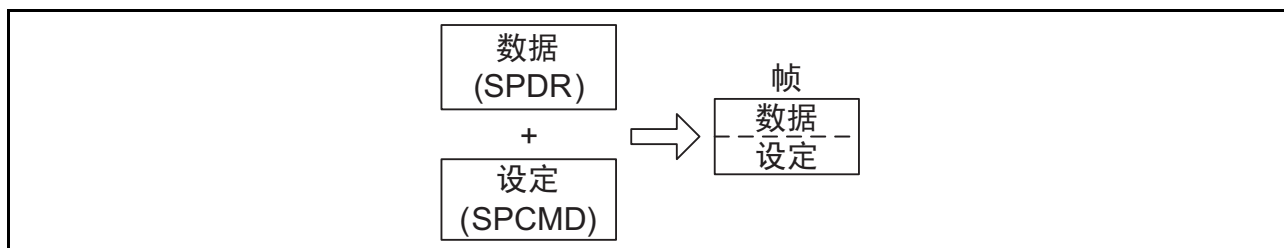


图 30.31 帧的概念图

通过表 30.4 的设定进行顺序运行时的命令和发送缓冲器 / 接收缓冲器的关系如图 30.32 所示。

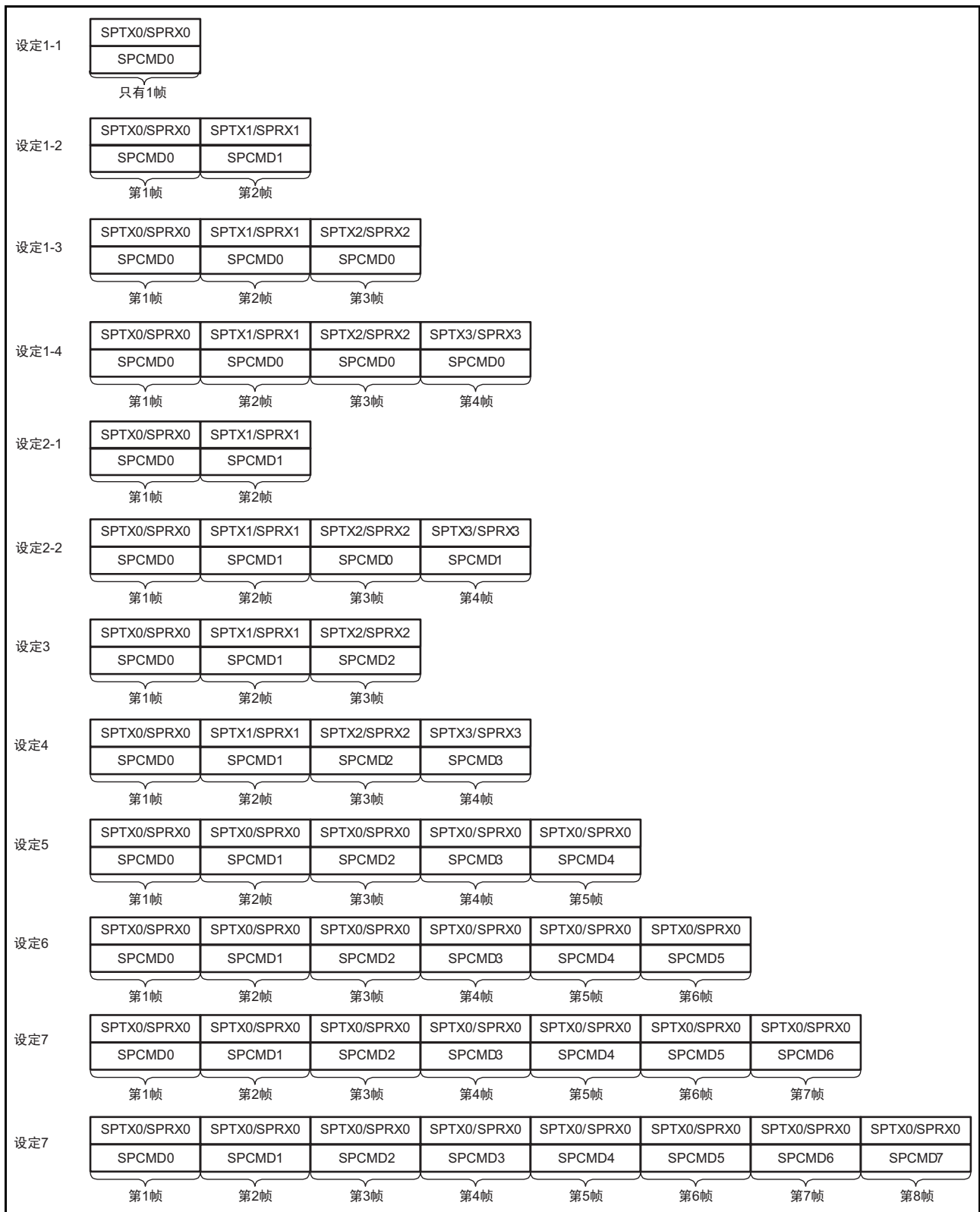


图 30.32 顺序运行时的 RSPI 命令寄存器和发送 / 接收缓冲器

(4) 突发传送

当 RSPI 正在进行的串行传送所参照的 SPCMDm.SSLKP 位为“1”时，RSPI 将串行传送过程中的 SSLAi 信号电平保持到下次串行传送的 SSLAi 信号开始有效为止。如果下次串行传送的 SSLAi 信号电平和正在进行的串行传送的 SSLAi 信号电平相同，RSPI 就能在保持 SSLAi 信号有效的状态下，连续进行串行传送（突发传送）。

通过设定 SPCMD0 寄存器和 SPCMD1 寄存器来实现突发传送时的 SSLAi 信号运行例子如图 30.33 所示。以下说明图 30.33 中记载的 (1) ~ (7) 的 RSPI 运行内容。SSLAi 输出信号的极性取决于 SSLP 寄存器的设定值。

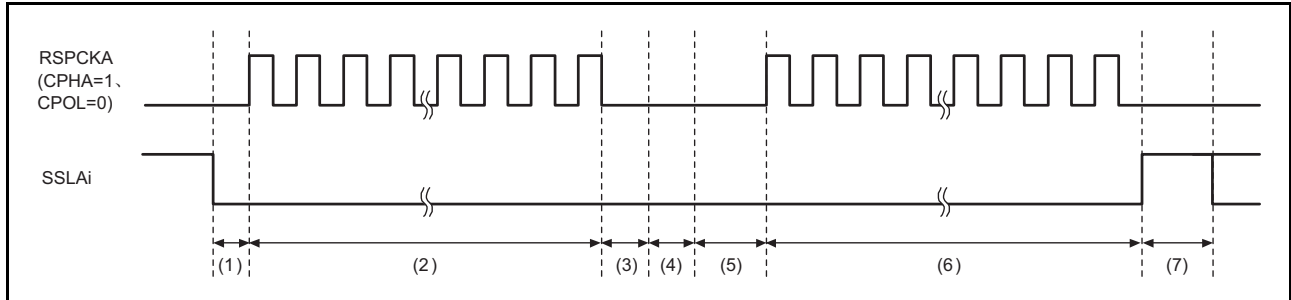


图 30.33 使用 SSLKP 位的突发传送运行例子

- (1) 根据 SPCMD0 寄存器，使 SSLAi 信号有效并且插入 RSPCK 延迟。
- (2) 根据 SPCMD0 寄存器进行串行传送。
- (3) 插入 SSL 无效延迟。
- (4) 因为 SPCMD0.SSLKP 位为“1”，所以保持 SPCMD0 寄存器中的 SSLAi 信号值。在此期间，保持的时间最短也要和下次存取 SPCMD0 寄存器的延迟相同。如果在经过最短期间后移位寄存器变空，就在将下次要传送的发送数据保存到移位寄存器前继续此期间。
- (5) 根据 SPCMD1 寄存器，使 SSLAi 信号有效并且插入 RSPCK 延迟。
- (6) 根据 SPCMD1 寄存器进行串行传送。
- (7) 因为 SPCMD1.SSLKP 位为“0”，所以 SSLAi 信号无效。根据 SPCMD1 寄存器插入下次的存取延迟。

在 SSLKP 位为“1”的 SPCMDm 寄存器所设 SSLAi 信号输出和下次传送使用的 SPCMDm 寄存器所设 SSLAi 信号输出不同的情况下，RSPI 在下次传送命令所对应的 SSLAi 信号有效时（图 30.33 的 (5)）切换 SSLAi 信号状态。必须注意：如果切换 SSLAi 信号，就可能因驱动 MISOA 的从属设备发生竞争而引起信号电平的冲突。

主控模式的 RSPI 在模块内部参照不使用 SSLKP 位时的 SSLAi 信号运行。即使 SPCMDm.CPHA 位为“0”，RSPI 也能使用在内部检测到的下次传送的有效 SSLAi 信号，正确地开始串行传送。因此，与 CPHA 位的设定值无关，能进行主控模式的突发传送（参照“30.3.10 SPI 运行”）。

(5) RSPCK 延迟 (t1)

主控模式的 RSPI 的 RSPCK 延迟值取决于 SPCMDm.SCKDEN 位的设定和 SPCKD 寄存器的设定。RSPI 通过指针控制来决定串行传送要参照的 SPCMDm 寄存器，并且使用所选的 SPCMDm.SCKDEN 位和 SPCKD 寄存器来决定如表 30.9 所示的串行传送时的 RSPCK 延迟值。有关 RSPCK 延迟的定义，请参照“30.3.5 传送格式”。

表 30.9 SCKDEN 位、SPCKD 寄存器和 RSPCK 延迟值的关系

SPCMDm.SCKDEN 位	SPCKD.SCKDL[2:0] 位	RSPCK 延迟值
0	000 ~ 111	1 个 RSPCK
1	000	1 个 RSPCK
	001	2 个 RSPCK
	010	3 个 RSPCK
	011	4 个 RSPCK
	100	5 个 RSPCK
	101	6 个 RSPCK
	110	7 个 RSPCK
	111	8 个 RSPCK

(6) SSL 无效延迟 (t2)

主控模式的 RSPI 的 SSL 无效延迟值取决于 SPCMDm.SLNDEN 位的设定和 SSLND 寄存器的设定。RSPI 通过指针控制来决定串行传送要参照的 SPCMDm 寄存器，并且使用所选的 SPCMDm.SLNDEN 位和 SSLND 寄存器来决定表 30.10 所示的串行传送时的 SSL 无效延迟值。有关 SSL 无效延迟的定义，请参照“30.3.5 传送格式”。

表 30.10 SSLND 寄存器和 SSL 无效延迟值的关系

SPCMDm.SLNDEN 位	SSLND.SLNDL[2:0] 位	SSL 无效延迟值
0	000 ~ 111	1 个 RSPCK
1	000	1 个 RSPCK
	001	2 个 RSPCK
	010	3 个 RSPCK
	011	4 个 RSPCK
	100	5 个 RSPCK
	101	6 个 RSPCK
	110	7 个 RSPCK
	111	8 个 RSPCK

(7) 下次存取延迟 (t3)

主控模式的 RSPI 的下次存取延迟取决于 SPCMDm.SPNDEN 位的设定和 SPND 寄存器的设定。RSPI 通过指针控制来决定串行传送要参照的 SPCMDm 寄存器，并且使用所选的 SPCMDm.SPNDEN 位和 SPND 寄存器来决定如表 30.11 所示的串行传送时的 RSPCK 延迟。有关下次存取延迟的定义，请参照“30.3.5 传送格式”。

表 30.11 SPNDEN 位、SPND 寄存器和下次存取延迟值的关系

SPCMDm.SPNDEN 位	SPND.SPNDL[2:0] 位	下次存取延迟值
0	000 ~ 111	1 个 RSPCK+2 个 PCLK
1	000	1 个 RSPCK+2 个 PCLK
	001	2 个 RSPCK+2 个 PCLK
	010	3 个 RSPCK+2 个 PCLK
	011	4 个 RSPCK+2 个 PCLK
	100	5 个 RSPCK+2 个 PCLK
	101	6 个 RSPCK+2 个 PCLK
	110	7 个 RSPCK+2 个 PCLK
	111	8 个 RSPCK+2 个 PCLK

(8) 初始化流程

在 SPI 运行的情况下，在 主控模式中使用 RSPI 时的初始化流程例子如图 30.34 所示。有关中断控制器、DMAC 和输入 / 输出端口的设定方法，请参照各模块的说明。

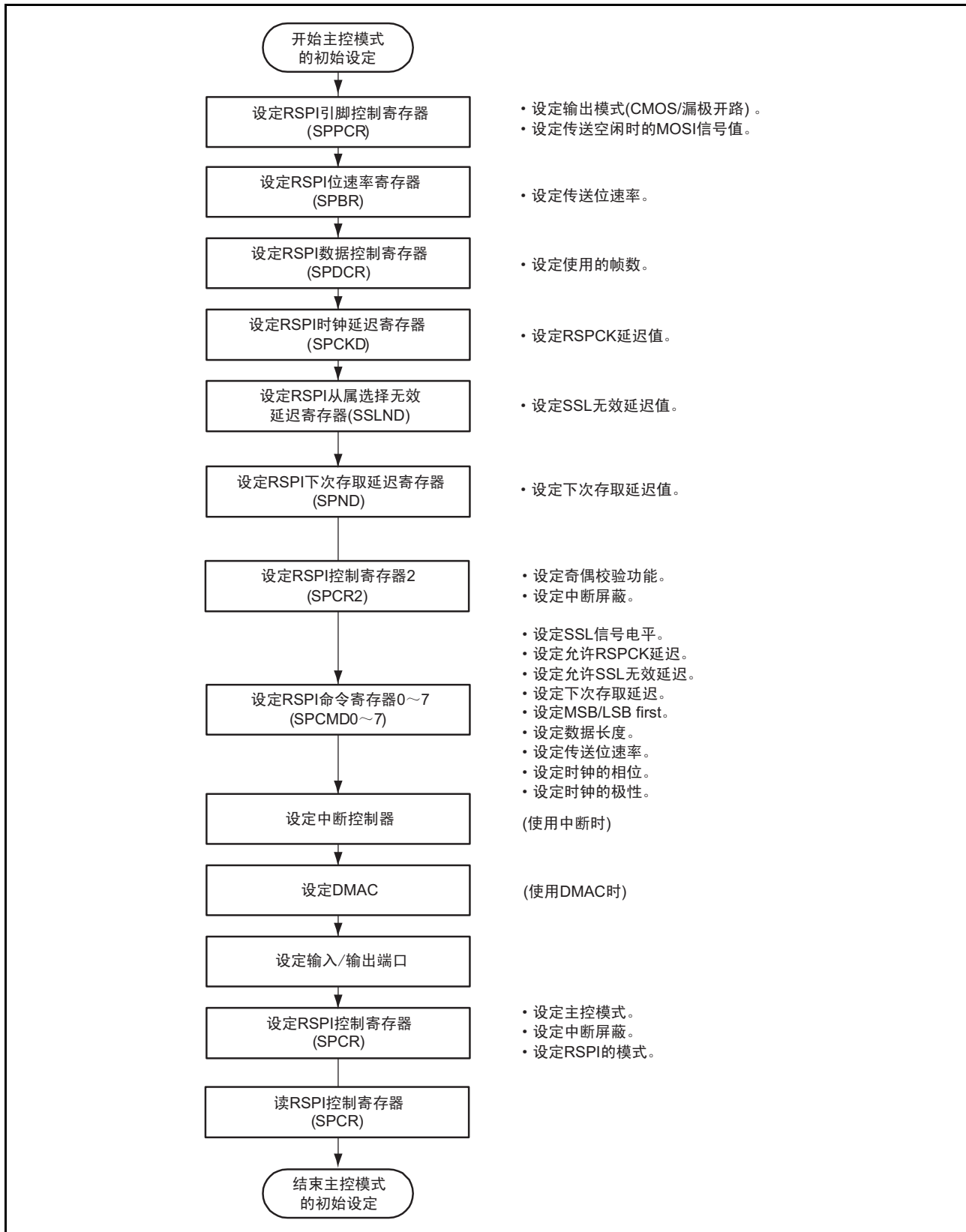


图 30.34 主控模式时的初始化流程例子 (SPI 运行)

(9) 运行流程

在 SPI 运行时，主控模式的运行流程如图 30.35 所示。

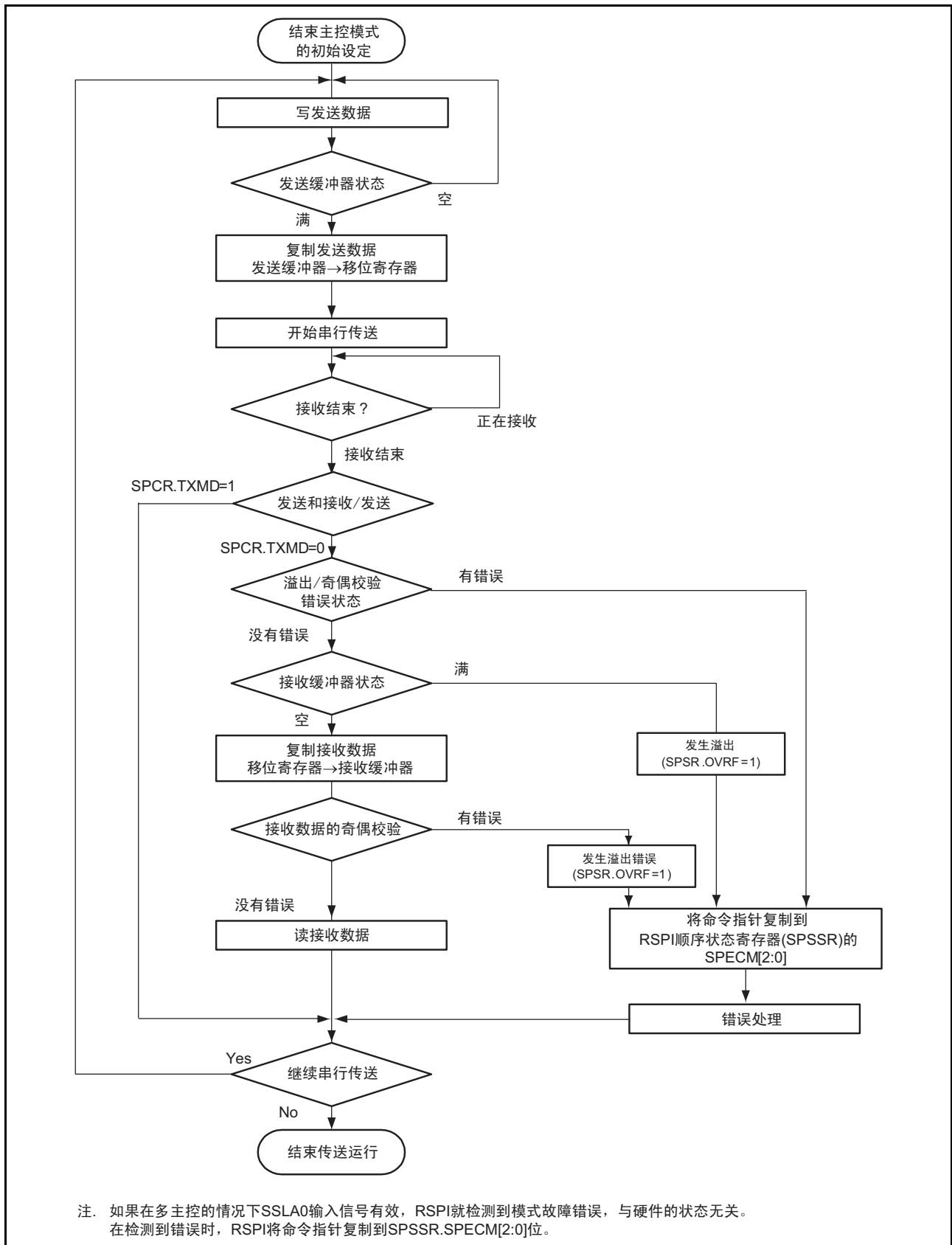


图 30.35 主控模式时的传送运行流程 (SPI 运行)

30.3.10.2 从属模式的运行

(1) 串行传送的开始

当 SPCMD0.CPHA 位为“0”时，如果 RSPI 检测到 SSLA0 输入信号有效，就需要开始向 MISOA 输出信号驱动有效数据。因此，在 CPHA 位为“0”时，SSLA0 输入信号的有效为串行传送的开始触发。

当 CPHA 位为“1”时，如果在 SSLA0 输入信号有效的状态下 RSPI 检测到最初的 RSPCKA 边沿，就需要开始向 MISOA 输出信号驱动有效数据。因此，在 CPHA 位为“1”时，SSLA0 信号有效状态下的最初的 RSPCKA 边沿为串行传送的开始触发。

如果 RSPI 在移位寄存器为空的状态下检测到串行传送的开始，就将移位寄存器改为满状态，并且不能在串行传送过程中将数据从发送缓冲器传送到移位寄存器。如果在开始串行传送前移位寄存器已满，RSPI 就保持移位寄存器的满状态。

与 CPHA 位的设定无关，RSPI 开始驱动 MISOA 输出信号的时序为 SSLA0 信号的有效时序。RSPI 的输出数据的有效或者无效因 CPHA 位的设定而发生变化。

有关 RSPI 传送格式的详细内容，请参照“30.3.5 传送格式”。SSLA0 输入信号的极性取决于 SSLP.SSL0P 位的设定值。

(2) 串行传送的结束

与 SPCMD0.CPHA 位无关，如果 RSPI 检测到相当于最后采样时序的 RSPCKA 边沿，就结束串行传送。在接收缓冲器为空的情况下，RSPI 在串行传送结束后将接收数据从移位寄存器复制到 SPDR 寄存器的接收缓冲器。与接收缓冲器的状态无关，RSPI 在串行传送结束后将移位寄存器改为空状态。在串行传送开始到串行传送结束的期间，如果 RSPI 检测到 SSLA0 输入信号无效，就发生模式故障错误（参照“30.3.8 错误检测”）。

最后的采样时序取决于传送数据的位长，从属模式的 RSPI 的数据长度取决于 SPCMD0.SPB[3:0] 位的设定值，SSLA0 输入信号的极性取决于 SSLP.SSL0P 位的设定值。有关 RSPI 传送格式的详细内容，请参照“30.3.5 传送格式”。

(3) 单从属模式中的注意事项

当 SPCMD0.CPHA 位为“0”时，如果 RSPI 检测到 SSLA0 输入信号的有效边沿，就开始串行传送。在图 30.7 的例子所示的结构中，在单从属模式中使用 RSPI 时，因为 SSLA0 输入信号总是被固定为有效状态，所以 CPHA 位为“0”的 RSPI 无法正常开始串行传送。在 SSLA0 输入信号被固定为有效状态的结构中，要使从属模式的 RSPI 正常进行发送和接收，必须将 CPHA 位置“1”。如果需要将 CPHA 位置“0”，就不能固定 SSLA0 输入信号。

(4) 突发传送

当 SPCMD0.CPHA 位为“1”时，能在保持 SSLA0 输入信号有效的状态下进行连续的串行传送（突发传送）。当 CPHA 位为“1”时，从 SSLA0 输入信号有效状态的最初 RSPCKA 边沿到最后位的接收采样时序的期间相当于串行传送期间。即使 SSLA0 输入信号保持有效电平，因为能检测到存取的开始，所以仍然能应对突发传送。

当 CPHA 位为“0”时，不能正确进行突发传送第 2 次以后的串行传送。

(5) 初始化流程

在SPI运行的情况下，在从属模式中使用RSPI时的初始化流程例子如图30.36所示。有关中断控制器、DMAC和输入/输出端口的设定方法，请参照各块的说明。

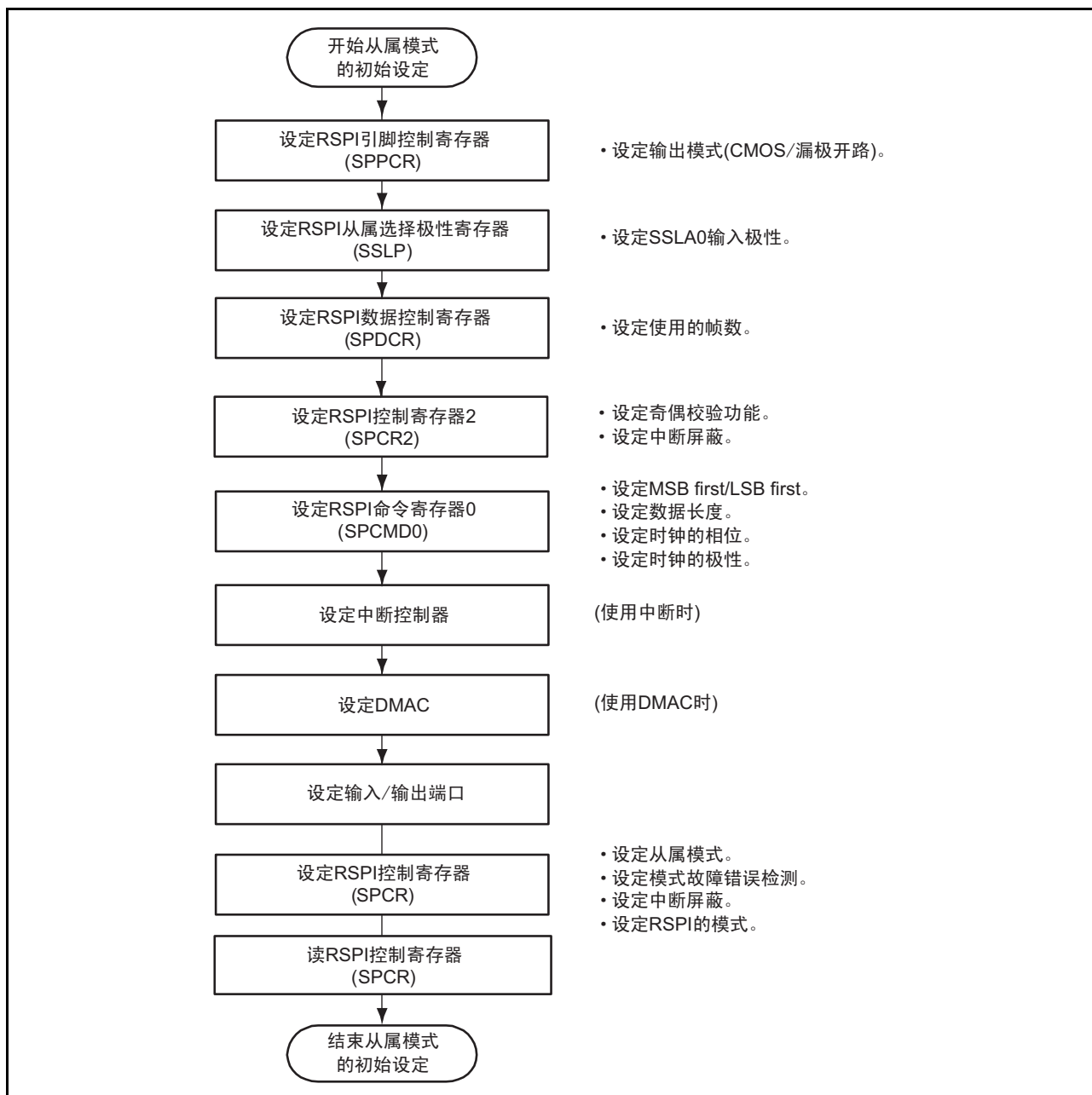


图 30.36 从属模式时的初始化流程例子 (SPI 运行)

(6) 运行流程

在 SPI 运行时，从属模式的运行流程如图 30.37 所示。

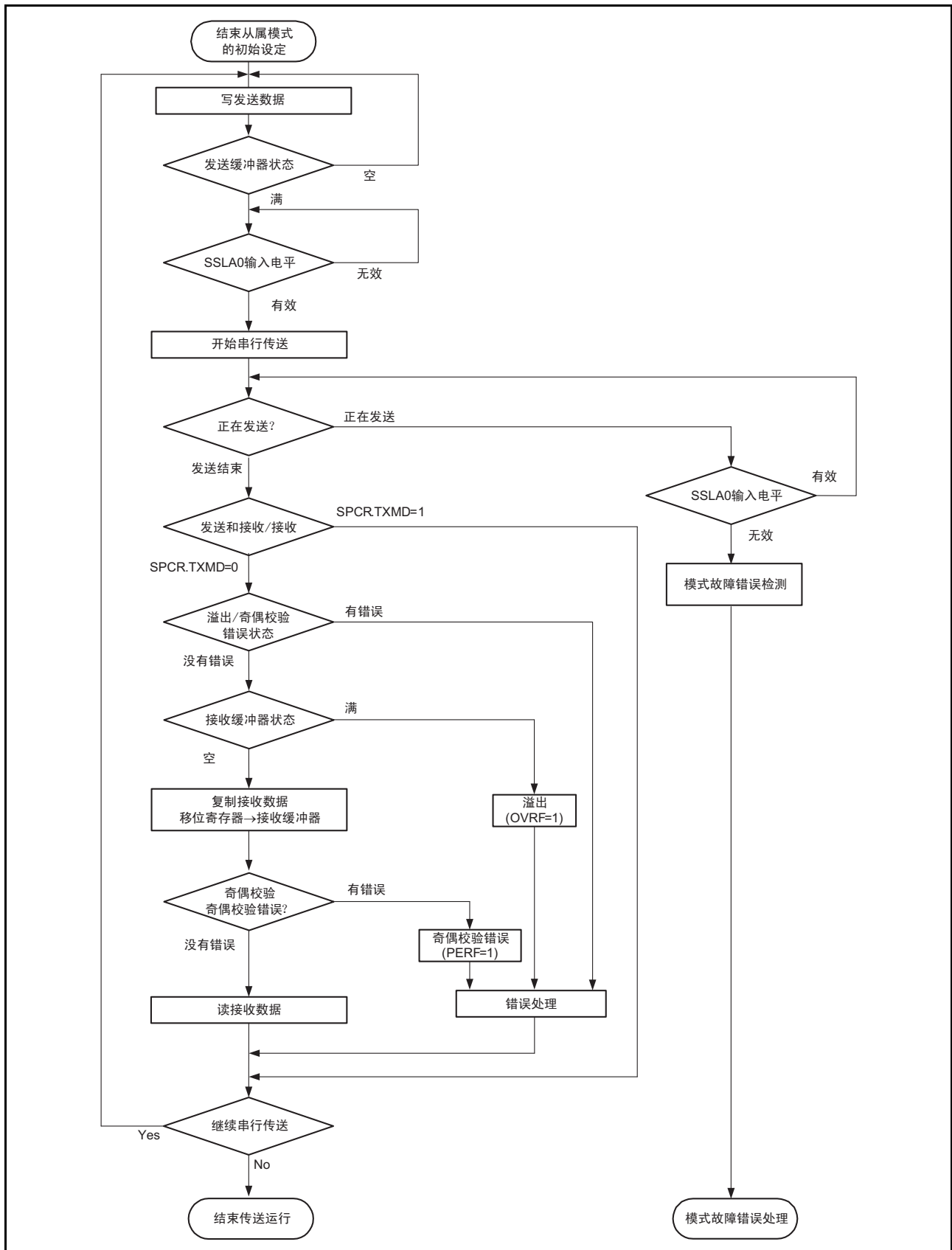


图 30.37 从属模式时的传送运行流程例子 (SPI 运行)

30.3.11 时钟同步运行

在 SPCR.SPMS 位为“1”时，RSPI 进行时钟同步运行。在进行时钟同步运行时，不使用 SSLAi 引脚而使用 RSPCKA、MOSIA 和 MISOA 这 3 个引脚进行通信，SSLAi 引脚能用作 I/O 端口。

在进行时钟同步运行时，不使用 SSLAi 引脚进行通信，但是模块内部的运行和 SPI 运行模式相同。在主控运行或者从属运行中，能按照和 SPI 运行时相同的流程进行通信。因为不使用 SSLAi 引脚，所以检测不到模式故障错误。

在进行时钟同步运行时，无法保证从属模式中 (SPCR.MSTR=0) SPCMDm.CPHA 位为“0”时的运行。

30.3.12 主控模式的运行

(1) 串行传送的开始

在发送缓冲器为空的（未设定下次传送的数据）状态下，如果将数据写到 SPDR 寄存器，RSPI 就更新 SPDR 寄存器的发送缓冲器 (SPTX) 的数据。在将 SPCR.SPFC[1:0] 位所设数据写到 SPDR 寄存器后移位寄存器变空时，RSPI 在将发送缓冲器的数据复制到移位寄存器后开始串行传送。如果 RSPI 将发送数据复制到移位寄存器，就将移位寄存器改为满状态；如果串行传送结束，就将移位寄存器改为空状态。不能参照移位寄存器的状态。

有关 RSPI 传送格式的详细内容，请参照“30.3.5 传送格式”。但是，在进行时钟同步通信时，不使用 SSLA0 输出信号。

(2) 串行传送的结束

如果 RSPI 发送与最后采样时序对应的 RSPCKA 边沿，就结束串行传送。在接收缓冲器 (SPRX) 为空的情况下，在串行传送结束后将数据从移位寄存器复制到 SPDR 寄存器的接收缓冲器。

最后的采样时序取决于传送数据的位长，主控模式的 RSPI 的数据长度取决于 SPCMDm.SPB[3:0] 位的设定值。有关 RSPI 传送格式的详细内容，请参照“30.3.5 传送格式”。但是，在进行时钟同步通信时，不使用 SSLA0 输出信号。

(3) 顺序控制

主控模式中的传送格式取决于 SPSCR 寄存器、SPCMDm 寄存器、SPBR 寄存器、SPCKD 寄存器、SSLND 寄存器或者 SPND 寄存器。在进行时钟同步运行时，不输出 SSLAi 信号，但是这些设定有效。

SPSCR 寄存器决定主控模式的 RSPI 要执行的串行传送的顺序结构。给 SPCMDm 寄存器设定 SSLAi 输出信号值、MSB/LSB first、数据长度、部分位速率、RSPCKA 极性 / 相位以及设定是否要参照 SPCKD 寄存器、SSLND 寄存器和 SPND 寄存器；给 SPBR 寄存器设定部分位速率，给 SPCKD 寄存器设定 RSPI 时钟延迟值，给 SSLND 寄存器设定 SSL 无效延迟并且给 SPND 寄存器设定下次存取延迟值。

RSPI 根据 SPSCR 寄存器设定的顺序长度，由部分或者全部 SPCMDm 寄存器构成顺序。RSPI 有构成顺序的 SPCMDm 寄存器的对应指针。能通过读 SPSSR.SPCP[2:0] 位来确认此指针的值。如果通过将 SPCR.SPE 位置“1”来允许 RSPI 运行，RSPI 就将指向命令的指针设定到 SPCMD0 寄存器，并且在开始串行传送时将 SPCMD0 寄存器的设定内容反映到传送格式。每当各数据传送的下次存取延迟期间结束时，RSPI 将指针进行递增。当构成顺序的最后命令对应的串行传送结束时，RSPI 将指针设定到 SPCMD0 寄存器，重复执行顺序。

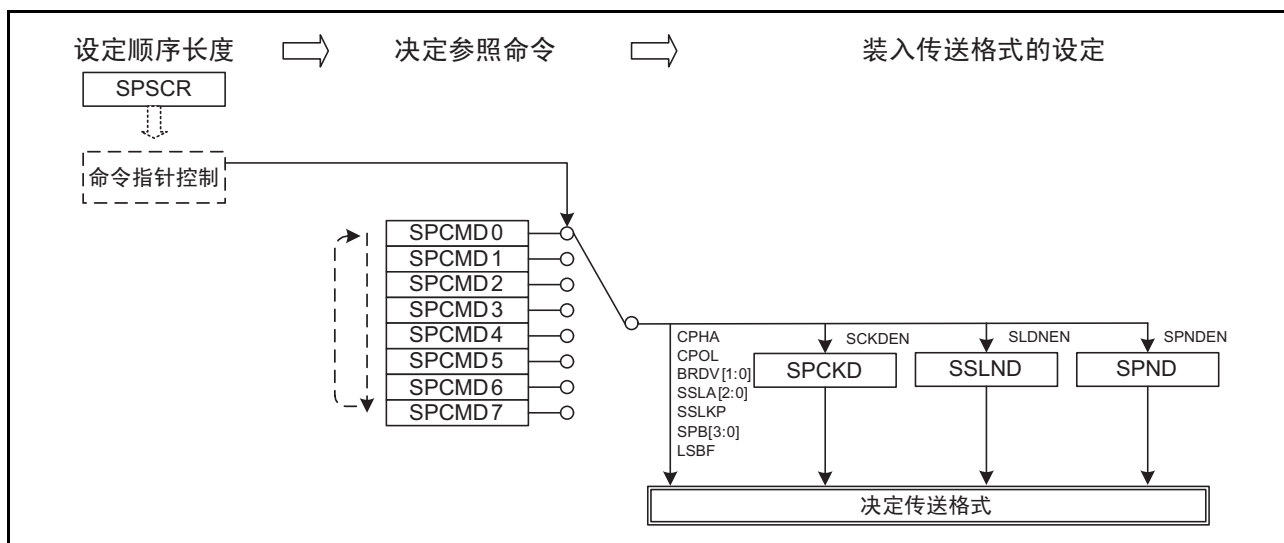


图 30.38 主控模式中的串行传送方式的决定方法 (时钟同步运行)

在本章节中，将数据 (SPDR) 和设定 (SPCMDm) 合成 1 帧。

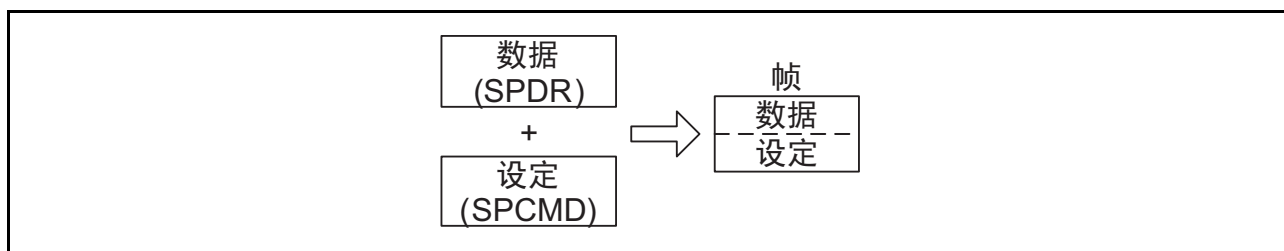


图 30.39 帧的概念图

通过表 30.4 的设定进行顺序运行时的命令和发送缓冲器 / 接收缓冲器的关系如图 30.40 所示。

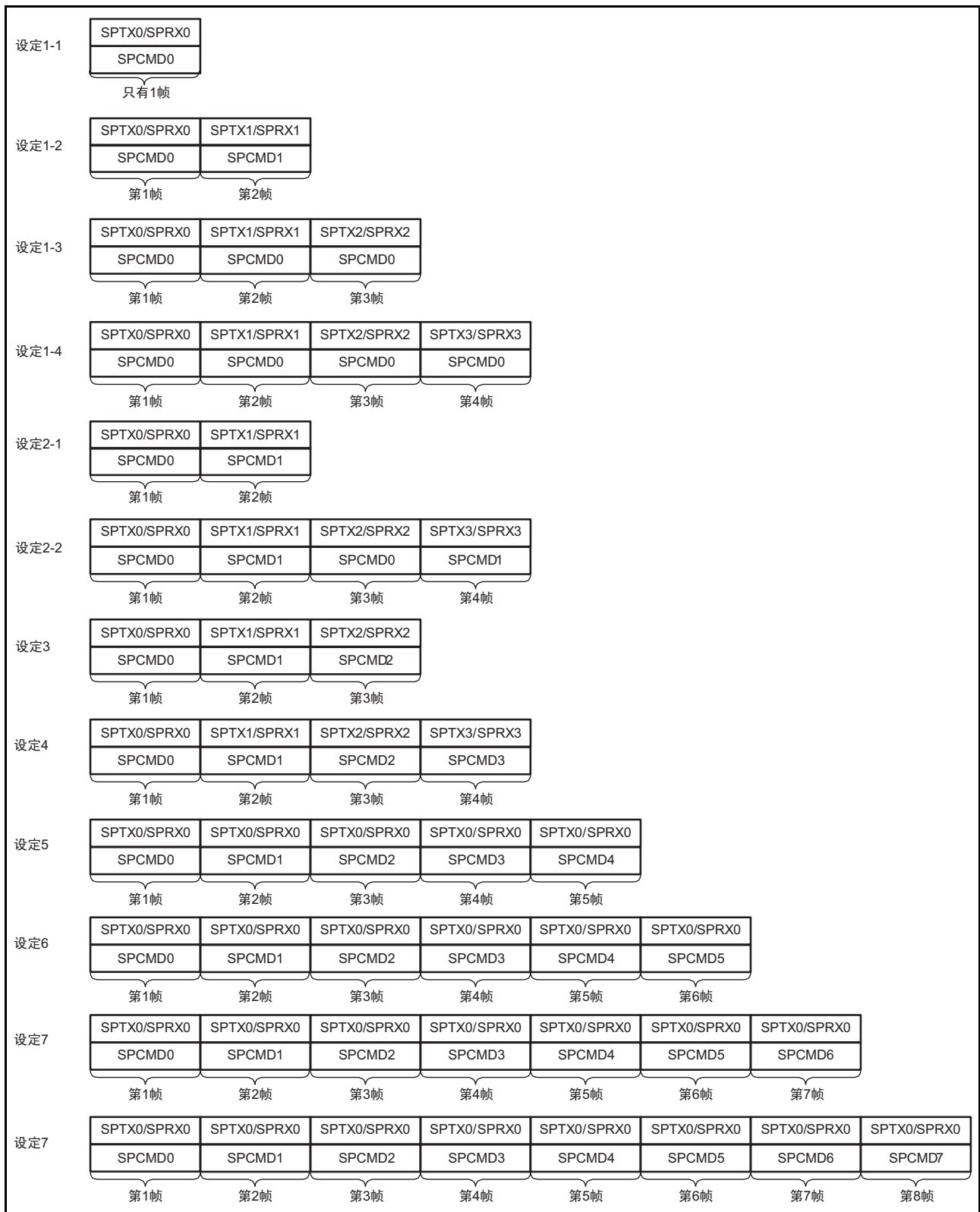


图 30.40 顺序运行时的 RSPI 命令寄存器和发送 / 接收缓冲器的对应

(4) 初始化流程

在时钟同步运行的情况下，在 主控模式中使用 RSPI 时的初始化流程例子如图 30.41 所示。有关中断控制器、DMAC 和输入 / 输出端口的设定方法，请参照各块的说明。

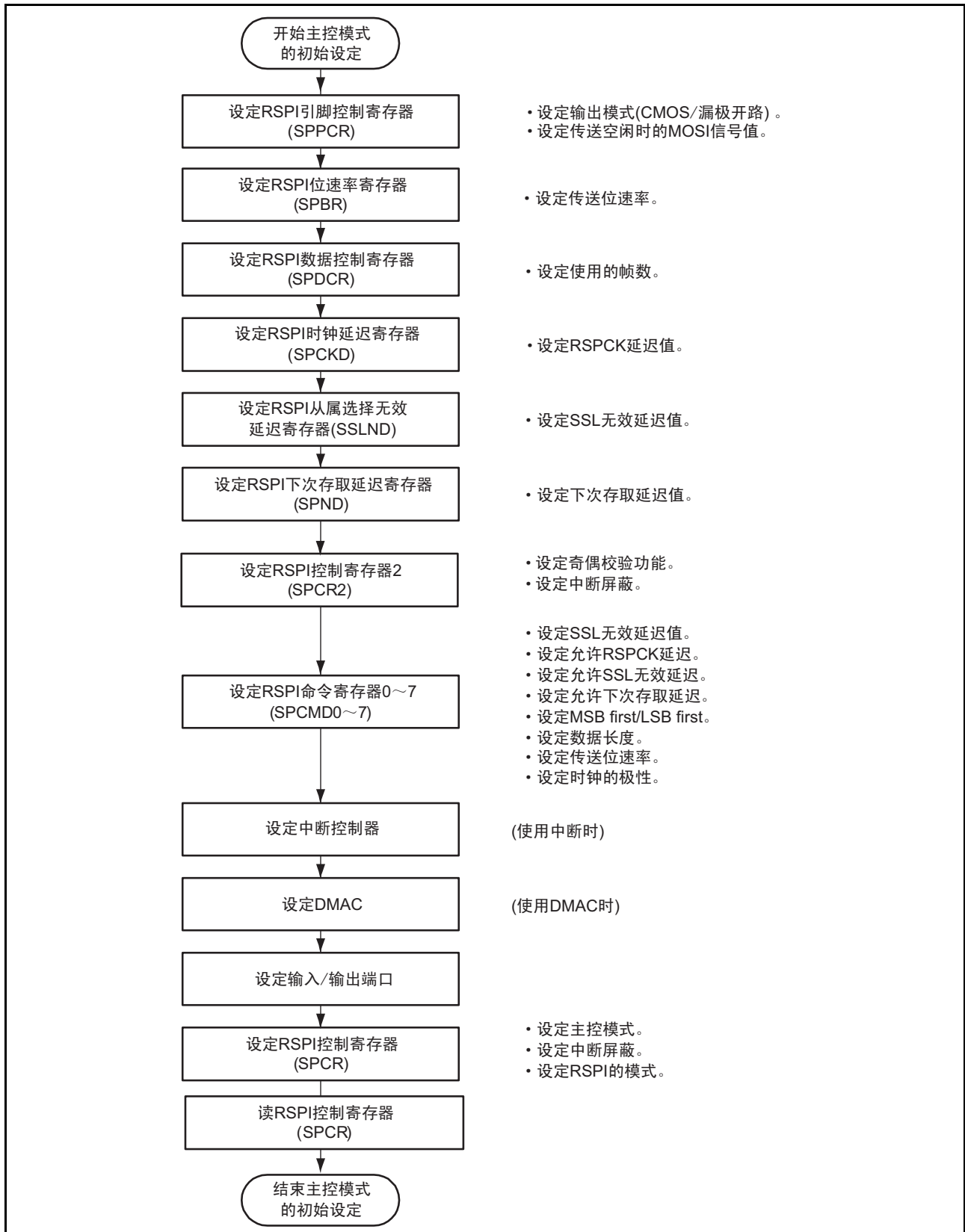


图 30.41 主控模式时的初始化流程例子 (时钟同步运行)

(5) 运行流程

在时钟同步运行时，主控模式的运行流程如图 30.42 所示。

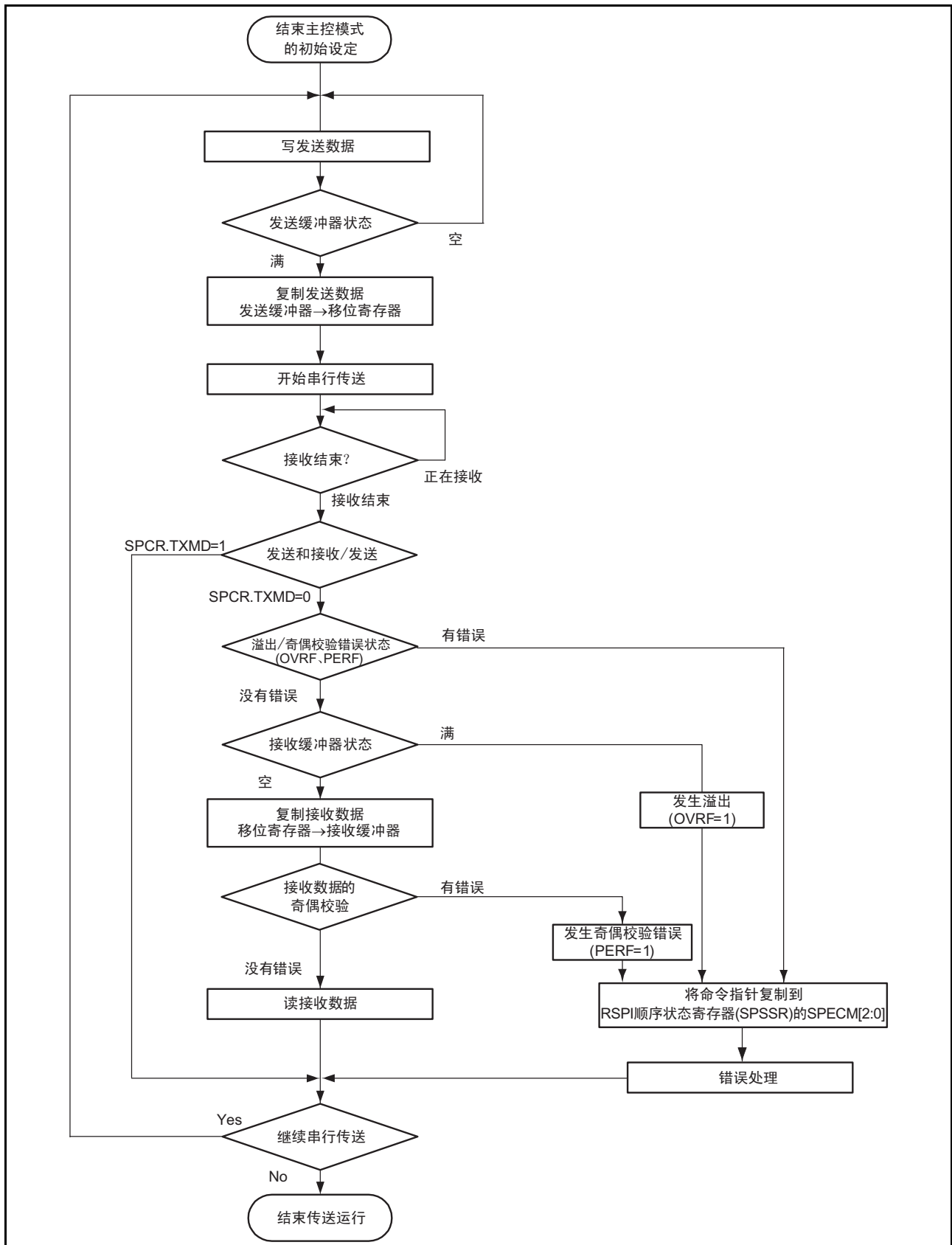


图 30.42 主控模式时的传送运行流程例子 (时钟同步运行)

30.3.13 从属模式的运行

(1) 串行传送的开始

在 SPCR.SPMS 位为“1”时，RSPI 的最初的 RSPCKA 边沿为串行传送的开始触发。

如果 RSPI 在移位寄存器为空的状态下检测到串行传送的开始，就将移位寄存器改为满状态，并且不能在串行传送过程中将数据从发送缓冲器复制到移位寄存器。如果在开始串行传送前移位寄存器已满，RSPI 就保持移位寄存器的满状态。

在 SPMS 位为“1”时，RSPI 随时驱动 MISOA 输出信号。

有关 RSPI 传送格式的详细内容，请参照“30.3.5 传送格式”。但是，在时钟同步运行时，不使用 SSL0 输入信号。

(2) 串行传送的结束

如果 RSPI 检测到相当于最后采样时序的 RSPCKA 边沿，就结束串行传送。在接收缓冲器为空的情况下，RSPI 在串行传送结束后将接收数据从移位寄存器复制到 SPDR 寄存器的接收缓冲器。在串行传送结束后，RSPI 将移位寄存器改为空状态。最后的采样时序取决于传送数据的位长，从属模式的 RSPI 的数据长度取决于 SPCMD0.SPB[3:0] 位的设定值。有关 RSPI 传送格式的详细内容，请参照“30.3.5 传送格式”。

(3) 初始化流程

在时钟同步运行的情况下，在从属模式中使用 RSPI 时的初始化流程例子如图 30.43 所示。有关中断控制器、DMAC 和输入 / 输出端口的设定方法，请参照各块的说明。

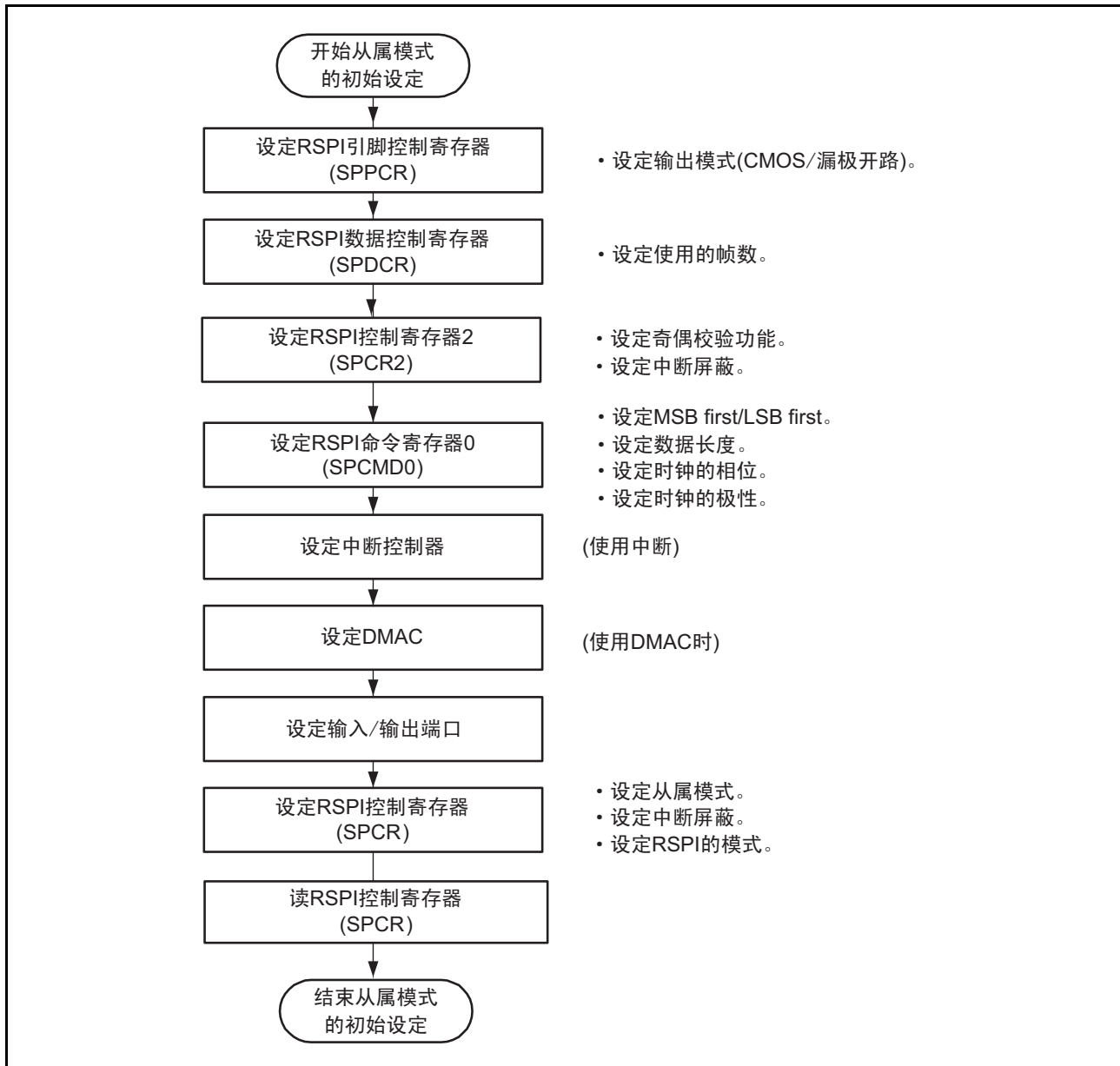


图 30.43 从属模式时的初始化流程例子 (时钟同步运行)

(4) 运行流程

在时钟同步运行时，从属模式的运行流程如图 30.44 所示。

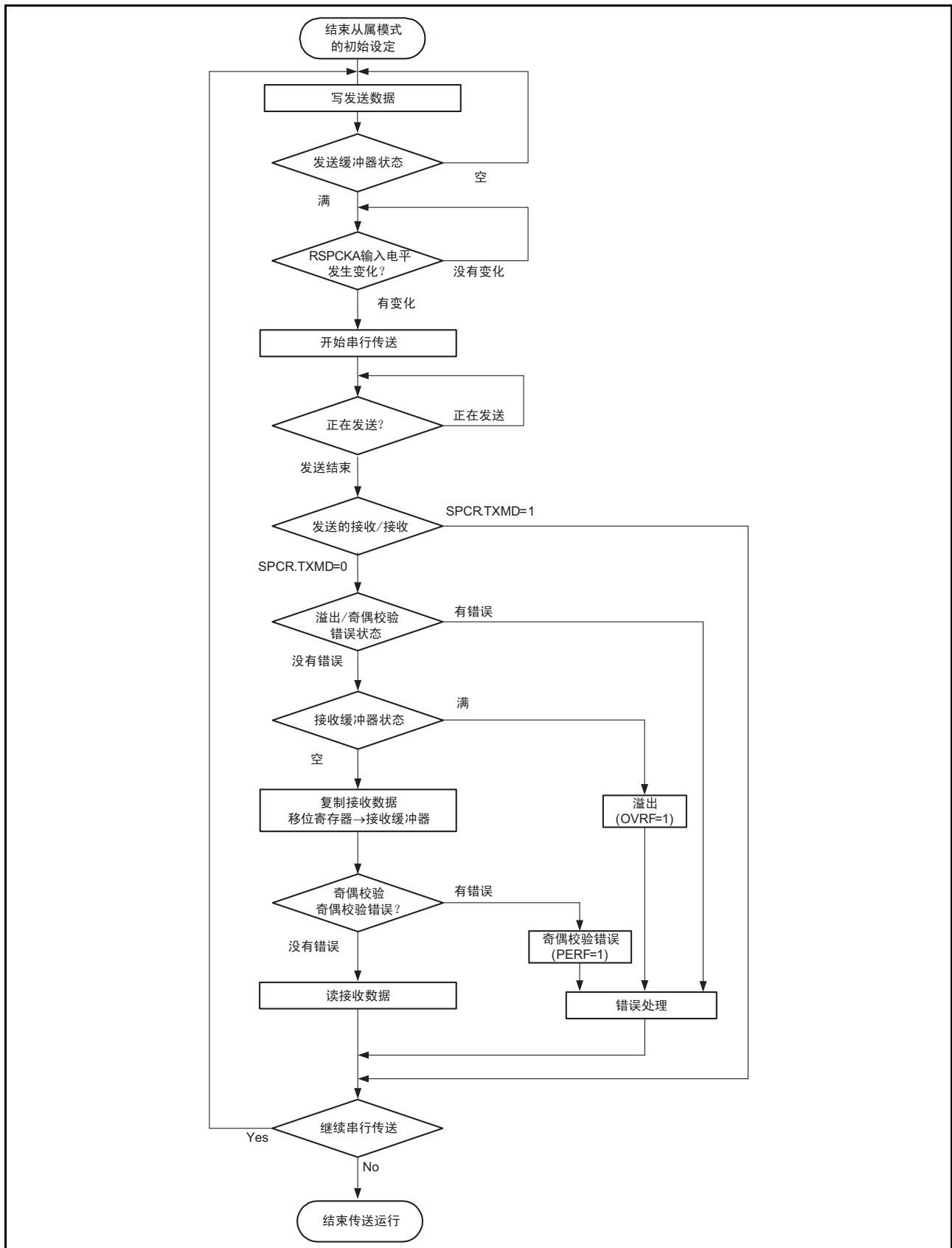


图 30.44 从属模式时的传送运行流程例子 (CPHA=1) (时钟同步运行)

30.3.14 错误处理

RSPI 的错误处理如图 30.45 ~ 图 30.47 所示。对于在主控模式和从属模式中发生的错误，能通过以下的错误处理从错误状态返回。

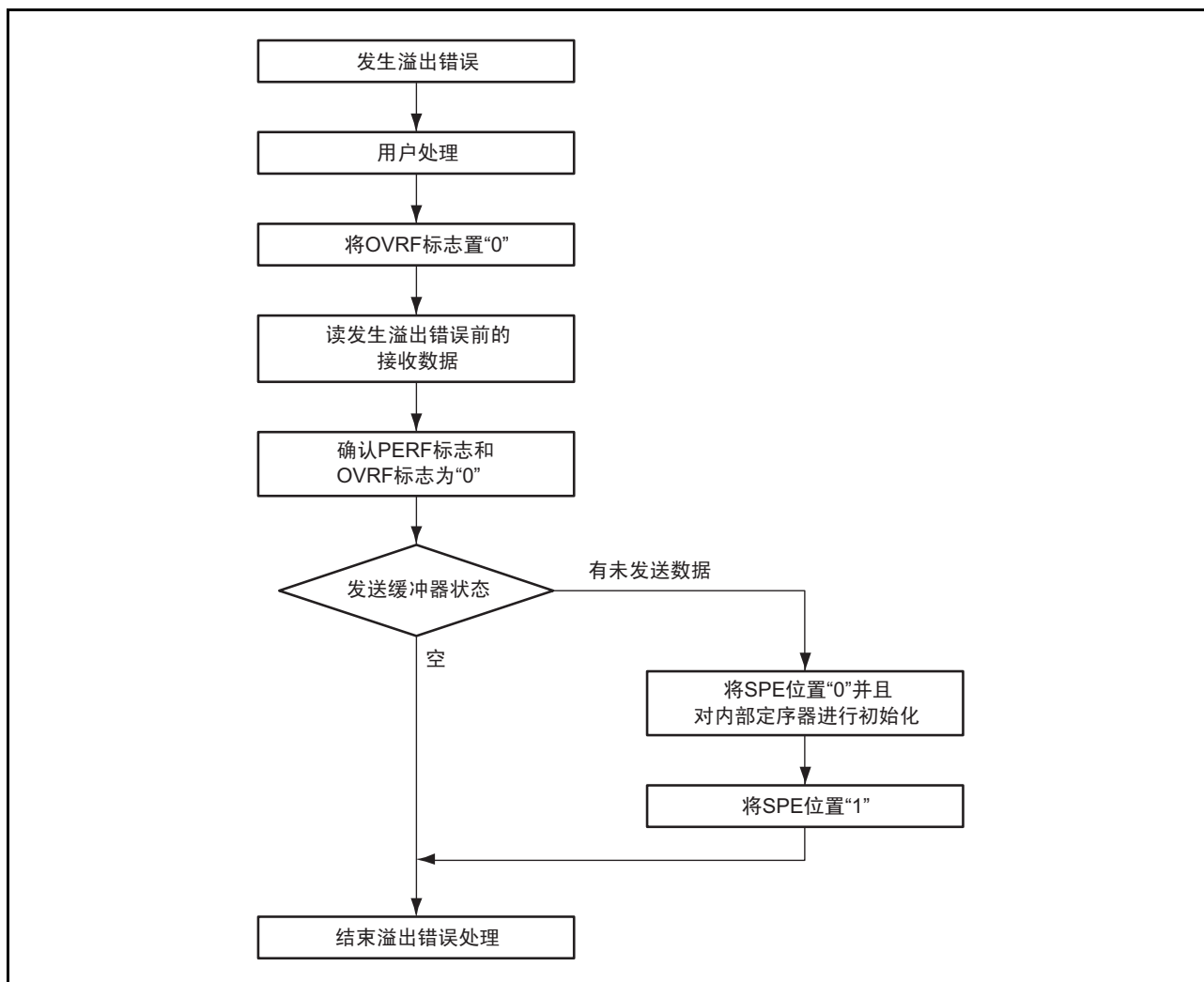


图 30.45 错误处理 (溢出错误)

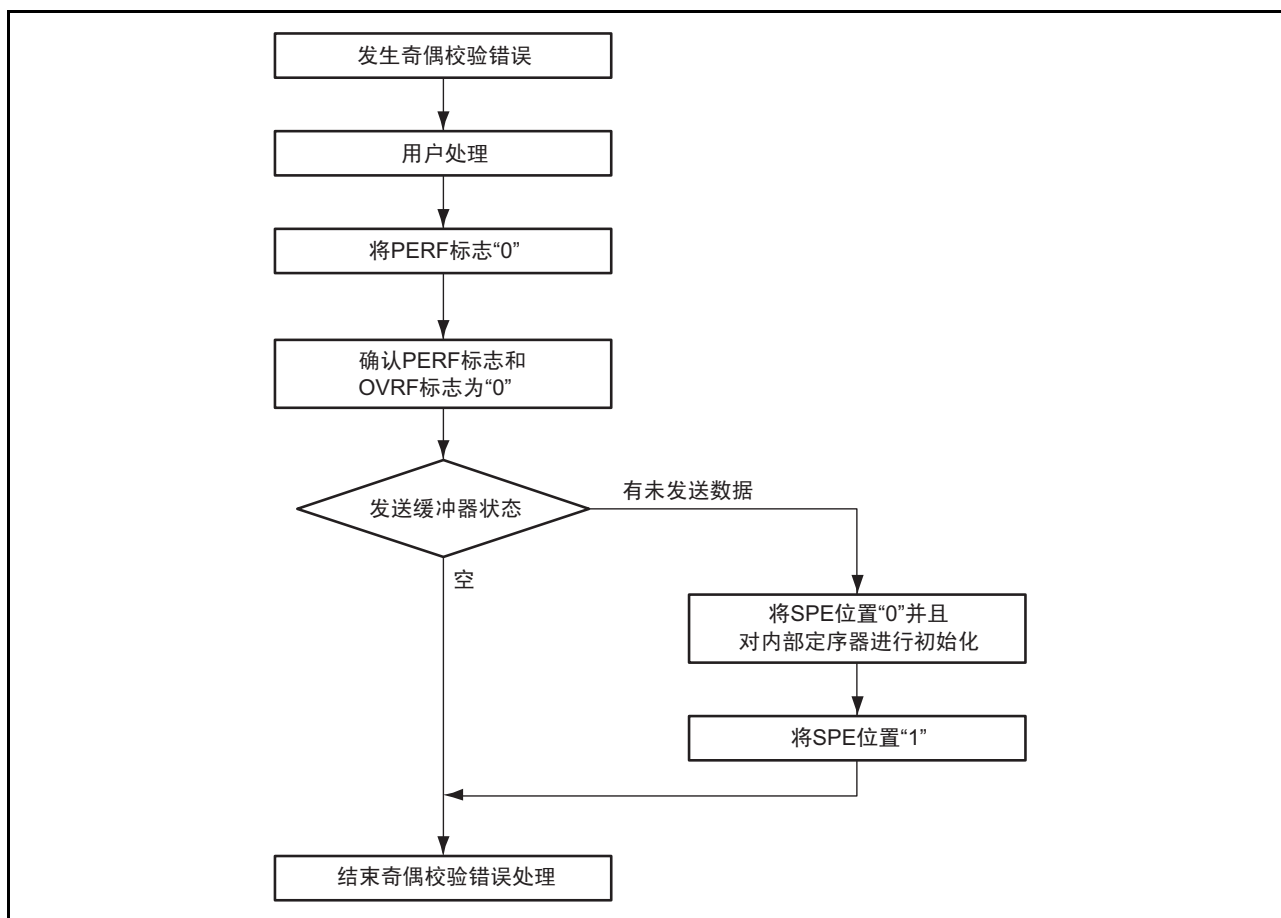


图 30.46 错误处理 (奇偶校验错误)

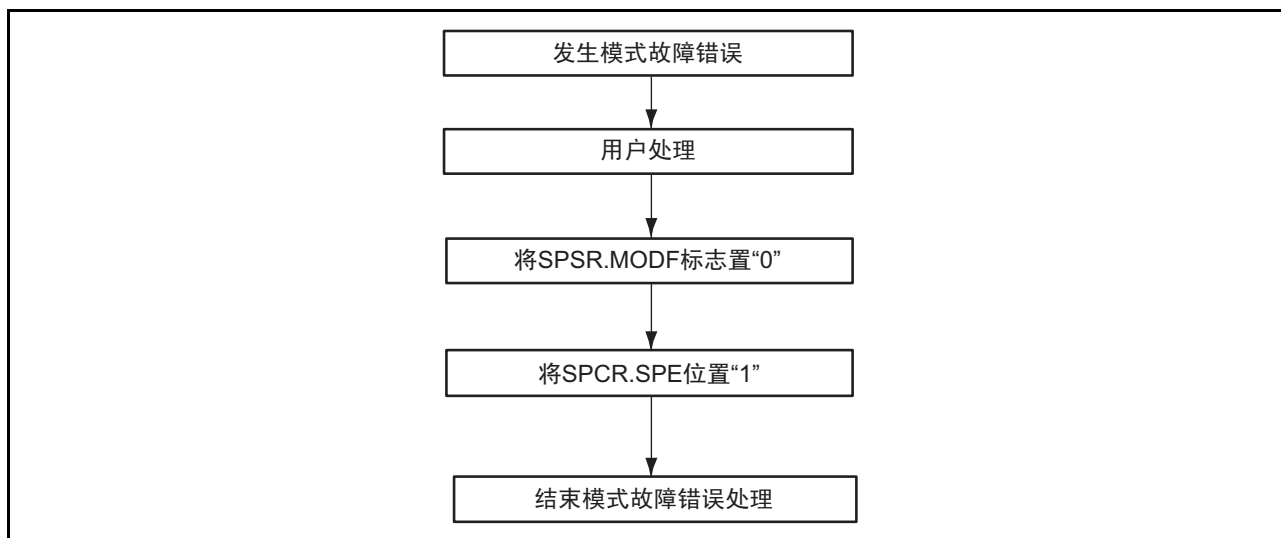


图 30.47 错误处理 (模式故障错误)

30.3.15 环回模式

在给 SPPCR.SPLP2 位或者 SPPCR.SPLP 位写“1”时，如果 SPCR.MSTR 位为“1”，RSPI 就切断 MISOA 引脚和移位寄存器之间的线路，而连接移位寄存器的输入线路和输出线路；如果 SPCR.MSTR 位为“0”，就切断 MOSIA 引脚和移位寄存器之间的线路，而连接移位寄存器的输入线路和输出线路。另外，如果 SPCR.MSTR 位为“1”，就切断 MOSIA 引脚和移位寄存器之间的线路；如果 SPCR.MSTR 位为“0”，就切断 MISOA 引脚和移位寄存器之间的线路。称此为环回模式。如果通过环回模式进行串行传送，RSPI 的发送数据或者发送数据取反后的数据就为 RSPI 的接收数据。

SPLP2 位和 SPLP 位的设定和接收数据的关系如表 30.12 所示，将主控模式的 RSPI 设定为环回模式 (SPPCR.SPLP2=0、SPPCR.SPLP=1) 时的移位寄存器输入 / 输出线路的结构如图 30.48 所示。

表 30.12 SPLP2 位、SPLP 位的设定和接收数据

SPPCR.SPLP2 位	SPPCR.SPLP 位	接收数据
0	0	MOSIA 引脚或者 MISOA 引脚的输入数据
0	1	发送数据取反后的数据
1	0	发送数据
1	1	发送数据

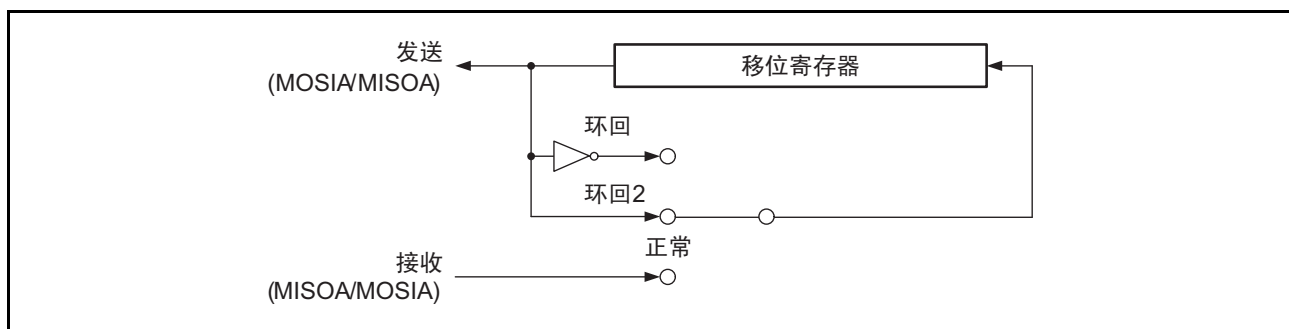


图 30.48 环回模式时的移位寄存器输入 / 输出结构 (主控模式)

30.3.16 奇偶校验位功能的自诊断

奇偶校验电路由发送数据的奇偶校验附加部分和接收数据的错误检测部分构成。要检测奇偶校验电路的奇偶校验附加部分和错误检测部分的故障时，必须按照如图 30.49 所示的流程，进行奇偶校验电路的自诊断。

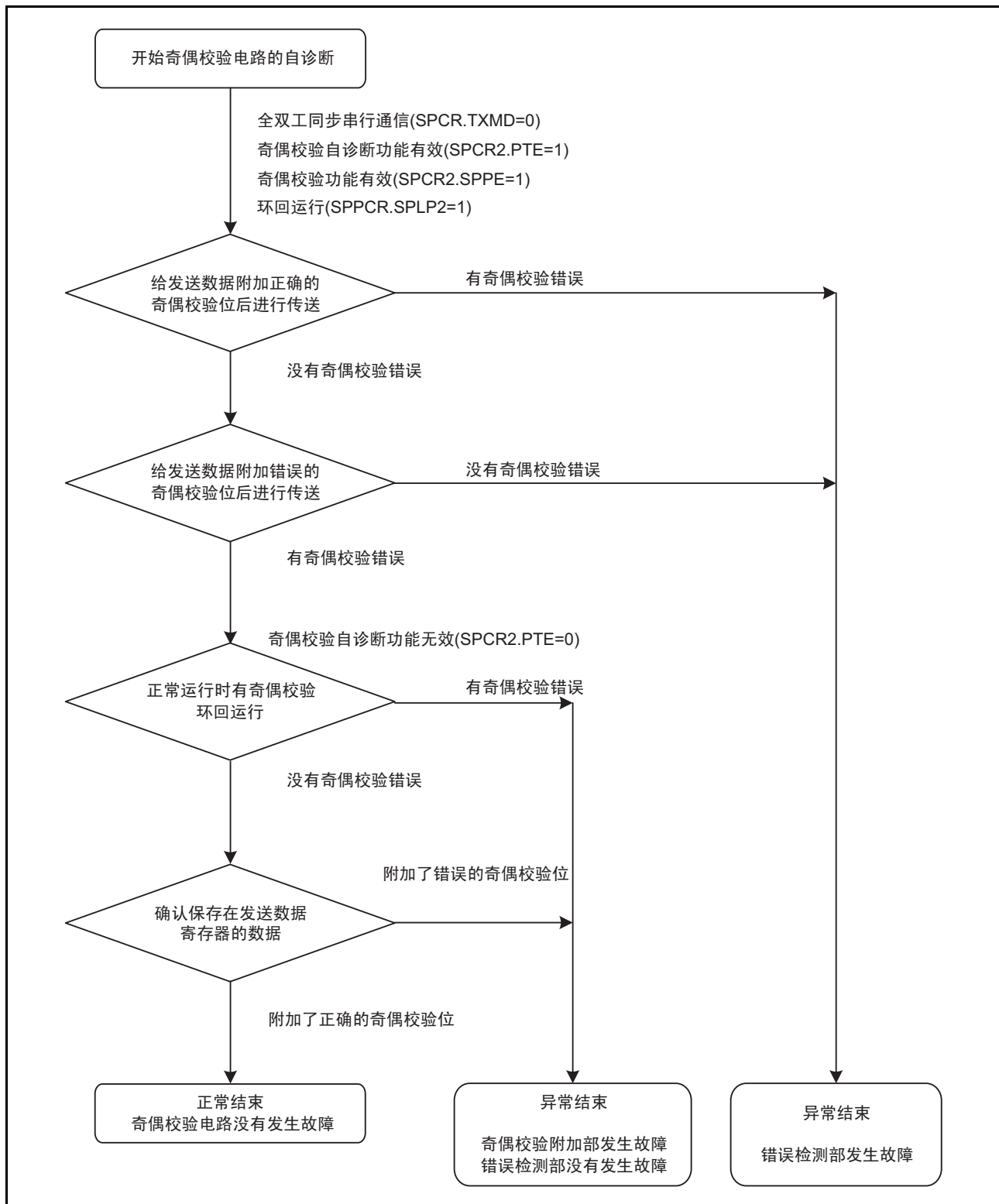


图 30.49 奇偶校验电路的自诊断流程

30.3.17 中断源

RSPI 的中断源有接收缓冲器满、发送缓冲器空、模式故障、溢出、奇偶校验错误和 RSPI 空闲。能通过接收缓冲器满或者发送缓冲器空的中断请求来启动 DTC 或者 DMAC，进行数据传送。

因为模式故障、溢出和奇偶校验错误的中断请求分配到 SPEI 的向量地址，所以需要通过标志判断中断源。RSPI 的中断源如表 30.13 所示。一旦表 30.13 所示的中断条件成立，就产生中断。对于接收缓冲器满和发送缓冲器空的中断源，必须通过数据传送来清除中断源。

在使用 DTC 或者 DMAC 进行发送或者接收时，必须先设定 DTC 或者 DMAC，然后在设定为允许状态后设定 RSPI。有关 DTC 或者 DMAC 的设定方法，请参照“16. DMA 控制器 (DMACA)”和“17. 数据传送控制器 (DTCa)”。

对于发送缓冲器空中断和接收缓冲器满中断，即使在 ICU.IRn.IR 标志为“1”时满足中断产生条件，也不将中断请求输出到 ICU 而保持在内部（在内部能保持的容量是每个中断源 1 个请求）。如果 ICU.IRn.IR 标志变为“0”，就将保持的中断请求输出到 ICU。如果输出被保持的中断请求，就自动清除保持的中断请求。也能通过将对应的中断允许位 (SPCR.SPTIE 位或者 SPCR.SPRIE 位) 置“0”来清除保持在内部的中断请求。

表 30.13 RSPI 的中断源

中断源	略称	中断条件	DMAC/DTC 的启动
接收缓冲器满	SPRI	在 SPCR.SPRIE 位为“1”的状态下接收缓冲器变满时	能
发送缓冲器空	SPTI	在 SPCR.SPTIE 位为“1”的状态下发送缓冲器变空时	能
RSPI 错误（模式故障、溢出、奇偶校验错误）	SPEI	在 SPCR.SPEIE 位为“1”的状态下 SPSR.MODF、SPSR.OVRF 和 SPSR.PERF 标志变为“1”时	不能
RSPI 空闲	SPII	在 SPCR.SPIIE 位为“1”的状态下 IDLNF 标志变为“0”时	不能

30.4 事件链接输出功能

RSPI 具有对事件链接控制器 (ELC) 输出以下事件的功能。与中断允许位的设定无关，能输出事件链接输出信号。

30.4.1 接收缓冲器满事件的输出

在串行传送结束并且将接收数据从移位寄存器传送到 SPDR 寄存器时，输出事件。

30.4.2 发送缓冲器空事件的输出

将发送数据从发送缓冲器传送到移位寄存器并且在 SPE 位从“0”变为“1”时，输出信号。

30.4.3 模式故障 / 溢出 / 奇偶校验错误事件的输出

(1) 模式故障

模式故障事件的输出条件如表 30.14 所示。

表 30.14 模式故障事件的输出条件

	SPCR.MODFEN 位	SSLA0 引脚	备注
主控 (MSTR 位为“1”)	1	有效	与发送运行的状态无关，只在有效时输出。
从属 (MSTR 位为“0”)	1	无效	只在发送过程中变为无效时输出。

作为模式故障事件的输出条件，在多主控模式时，如果在 SPCR.MSTR 位为“1”（主控模式）、SPCR.MODFEN 位为“1”（允许检测模式故障错误）的状态下，SSL 引脚的输入电平变为有效电平，RSPI 就在检测到模式故障错误时输出事件；在从属模式时，如果在 SPCR.MSTR 位为“0”（从属模式）、SPCR.MODFEN 位为“1”（允许检测模式故障错误）的状态下，SSL 引脚在结束数据传送所需的 RSPCK 周期前被置为无效，RSPI 就在检测到模式故障错误时输出事件。

(2) 溢出

作为溢出事件的输出条件，在 SPCR.TXMD 位为“0”并且接收缓冲器中有未读数据的状态下结束串行传送时，OVRF 标志就变为“1”并且输出事件。

(3) 奇偶校验错误

作为奇偶校验错误事件的输出条件，在 SPCR.TXMD 位为“0”、SPCR2.SPPE 位为“1”的状态下结束串行传送，并且在检测到奇偶校验错误时，输出事件。

30.4.4 RSPI 空闲事件的输出

(1) 主控模式的情况

如果为主控模式，就在 IDLNF 标志 (RSPI 空闲标志) 从“1”变为“0”时输出事件。当满足以下 (a) 或者 (b) 的条件时就为从“1”变为“0”的时序。

(a) 在发送过程中 SPCR.SPE 位变为“0” (RSPI 初始化) 时

(b) 在以下 3 个条件全部匹配时

- 发送缓冲器 (SPTX) 为空 (还未设定下一个传送数据) 时
- SPSSR.SPCP[2:0] 位为“000b” (顺序控制的起始位置) 时
- RSPI 内部定序器转移到空闲状态时 (在下一个存取延迟前结束运行的状态)

(2) 从属模式的情况

如果为从属模式，就在 SPCR.SPE 位为“0” (RSPI 初始化) 时输出事件。

30.4.5 发送结束事件的输出

如果为主控模式，就在 IDLNF 标志 (RSPI 空闲标志) 从“1”变为“0”时输出事件。如果为从属模式，就在表 30.15 所示的发送结束事件的输出条件 (从属) 下输出事件。

表 30.15 发送结束事件的输出条件 (从属)

	发送缓冲器的状态	移位寄存器的状态	其他
SPI 运行 (SPMS 位为“0”)	空	空	SSLA0 输入无效
时钟同步运行 (SPMS 位为“1”)	空	空	检测到最后 RSPCLK 的边沿

与主控模式 / 从属模式无关，在发送过程中给 SPCR.SPE 位写“0”，或者在通过发生模式故障清除 SPCR.SPE 位时，都不输出事件。

30.4.6 中断处理和事件链接的关系

RSPI 有接收缓冲器满、发送缓冲器空、模式故障、溢出、奇偶校验错误、RSPI 空闲等中断。这些中断分别有控制允许或者禁止中断的中断允许位。如果产生中断源，就在中断允许位为允许的情况下，对 CPU 输出中断请求信号。

与此相对，事件链接输出信号与中断允许位的设定无关，一旦产生中断源，就作为事件信号经由 ELC 输出到其他模块。

30.5 使用时的注意事项

30.5.1 模块停止功能的设定

能通过模块停止控制寄存器 B (MSTPCRB) 禁止或者允许 RSPI 的运行, 复位后的值为停止 RSPI 的运行。通过解除模块停止状态, 使寄存器变为可存取的状态。详细内容请参照“11. 低功耗功能”。

30.5.2 低功耗功能的注意事项

当使用低功耗功能来降低 RSPI 功耗时, 必须在将 SPCR.SPE 位置“0”并且结束通信后使用低功耗功能。

30.5.3 有关开始通信的注意事项

如果在 ICU.IRn.IR 标志为“1”的状态下开始通信, 通信开始后的中断请求就被保持在内部, 所以 ICU.IRn.IR 标志可能会导致意想不到的动作。

如果在开始通信时 ICU.IRn.IR 标志为“1”, 就必须在允许运行 (将 SPCR.SPE 位置“1”) 前按以下步骤清除中断请求:

1. 确认通信处于停止状态 (SPCR.SPE 位为“0”)。
2. 将对应的中断允许位 (SPCR.SPTIE 位或者 SPCR.SPRIE 位) 置“0”。
3. 读对应的中断允许位 (SPCR.SPTIE 位或者 SPCR.SPRIE 位), 确认是“0”。
4. 将 ICU.IRn.IR 标志置“0”。

31. CRC 运算器 (CRC)

CRC (Cyclic Redundancy Check) 运算器生成数据块的 CRC 码。

31.1 概要

CRC 运算器的规格和框图分别如表 31.1 和图 31.1 所示。

表 31.1 CRC 运算器的规格

项目	内容
CRC 运算的对象数据 (注 1)	对 8n 位的数据生成 CRC 码 (n 为自然数)。
数据块的大小	8 位
CRC 运算的处理方式	并行进行 8 位运算。
CRC 生成多项式	可从 3 个多项式中选择: <ul style="list-style-type: none"> • 8 位 CRC X^8+X^2+X+1 • 16 位 CRC $X^{16}+X^{15}+X^2+1$ $X^{16}+X^{12}+X^5+1$
CRC 运算的转换	能选择生成 LSB first 通信的 CRC 码或者生成 MSB first 通信的 CRC 码。
低功耗功能	能设定为模块停止状态。

注 1. 没有将运算对象数据分割成数据块的功能。必须以 8 位为单位进行写操作。

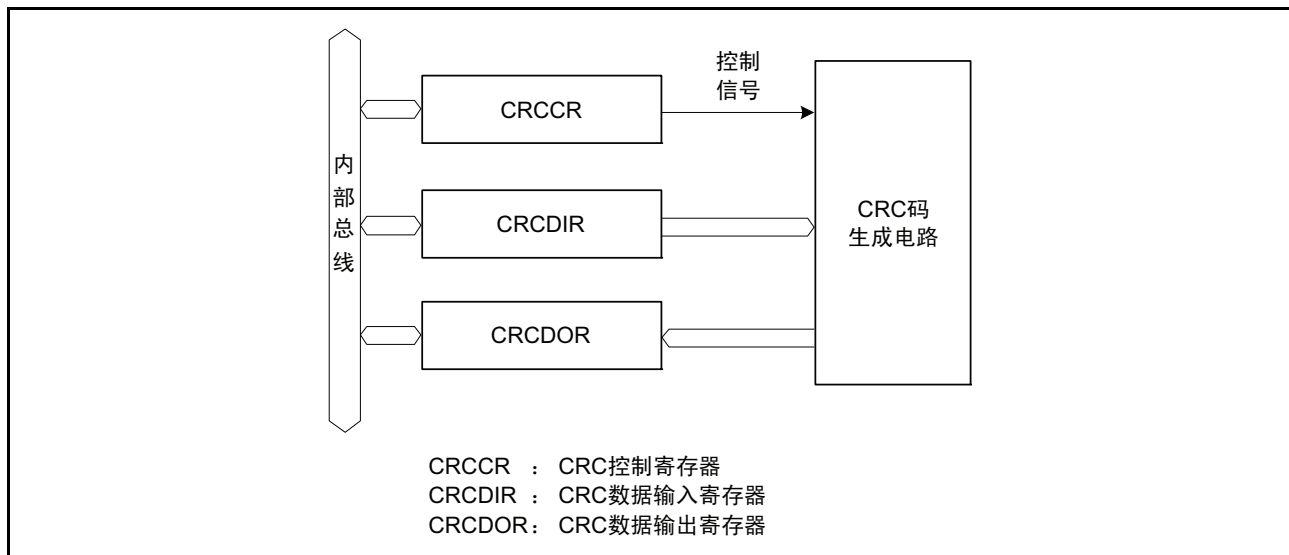
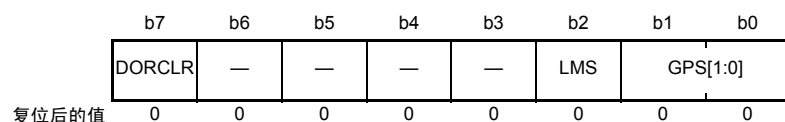


图 31.1 CRC 运算器的框图

31.2 寄存器说明

31.2.1 CRC 控制寄存器 (CRCCR)

地址 0008 8280h



位	符号	位名	功能	R/W
b1-b0	GPS[1:0]	CRC 生成多项式转换位	b1 b0 0 0: 不运算 0 1: X^8+X^2+X+1 1 0: $X^{16}+X^{15}+X^2+1$ 1 1: $X^{16}+X^{12}+X^5+1$	R/W
b2	LMS	CRC 运算转换位	0: 将进行 LSB first 通信时的 CRC 运算的 CRCDOR 寄存器值 (CRC 码) 以字节为单外分开发送时, 必须先发送低位字节 (b7 ~ b0)。 1: 将进行 MSB first 通信时的 CRC 运算的 CRCDOR 寄存器值 (CRC 码) 以字节为单外分开发送时, 必须先发送高位字节 (b15 ~ b8)。	R/W
b6-b3	—	保留位	读写值都为“0”。	R/W
b7	DORCLR	CRCDOR 寄存器清除位	1: 清除 CRCDOR 寄存器 (注 1) 读取值为“0”。	W

注 1. 只能写“1”。

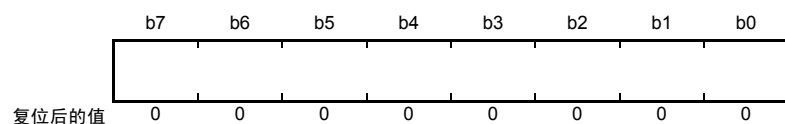
DORCLR 位 (CRCDOR 寄存器清除位)

如果将 DORCLR 位置“1”，CRCDOR 寄存器就变为“0000h”。

只能写“1”，读取值为“0”。

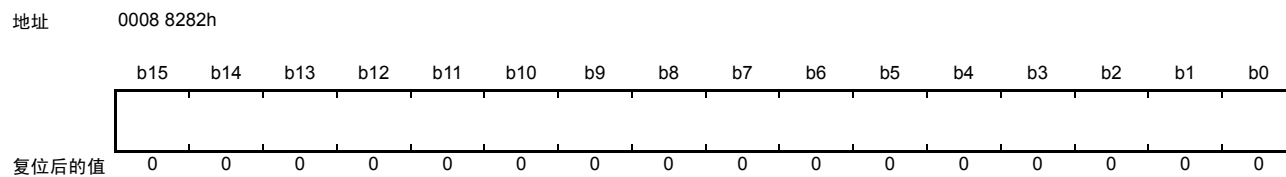
31.2.2 CRC 数据输入寄存器 (CRCDIR)

地址 0008 8281h



CRCDIR 寄存器是设定 CRC 运算对象数据块的 8 位可读写寄存器。

31.2.3 CRC 数据输出寄存器 (CRCDOR)



CRCDOR 寄存器是保存运算结果的 16 位可读写寄存器。

在一般情况下，如果在为了检查通信数据而在数据通信之后进行 CRC 码的运算时没有发生错误，CRCDOR 寄存器的值就为“0”。

如果使用 8 位 CRC (X^8+X^2+X+1 的多项式)，低位字节 (b7 ~ b0) 就能得到有效的 CRC 码。高位字节 (b15 ~ b8) 就被更新。

31.3 CRC 运算器的运行说明

CRC 运算器生成用于 LSB first/MSB first 通信的 CRC 码。

假设 CRCCR.GPS[1:0] 位为 “11b”，使用 16 位 CRC ($X^{16}+X^{12}+X^5+1$ 的多项式)，生成数据 “F0h” 的 CRC 码的例子如下所示。

如果使用 8 位 CRC (X^8+X^2+X+1 的多项式)，CRCDOR 寄存器的低位字节就能得到有效的 CRC 码。

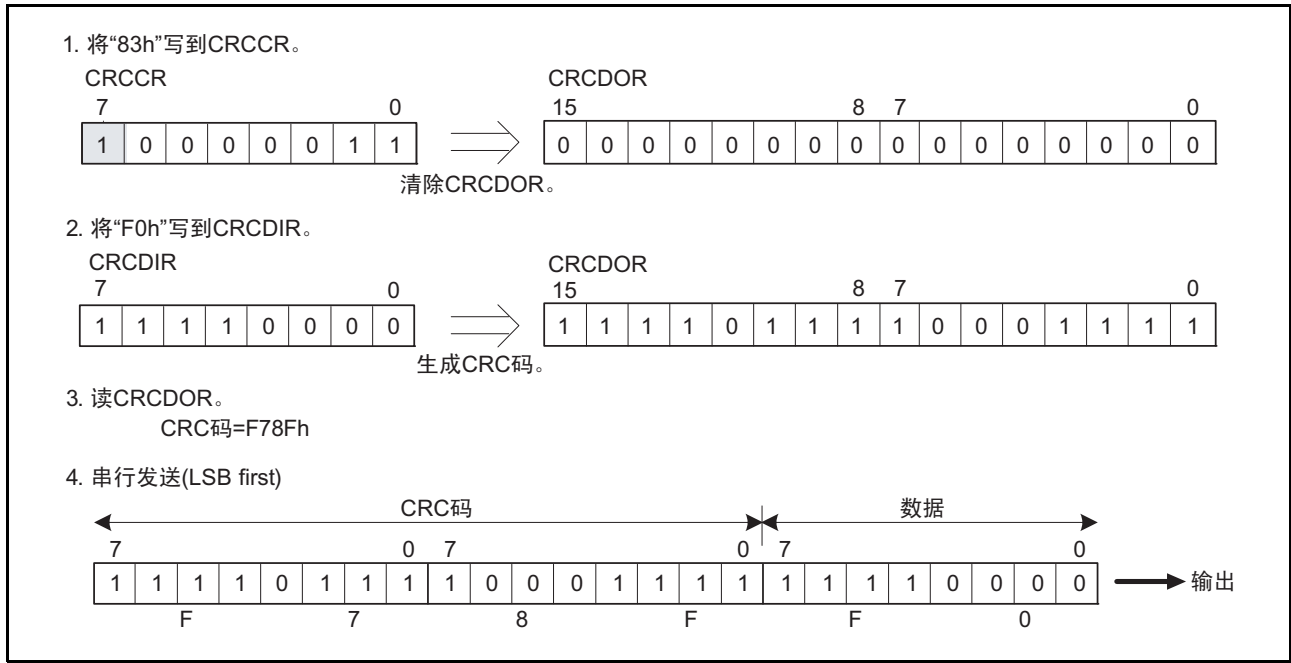


图 31.2 LSB first 的数据发送

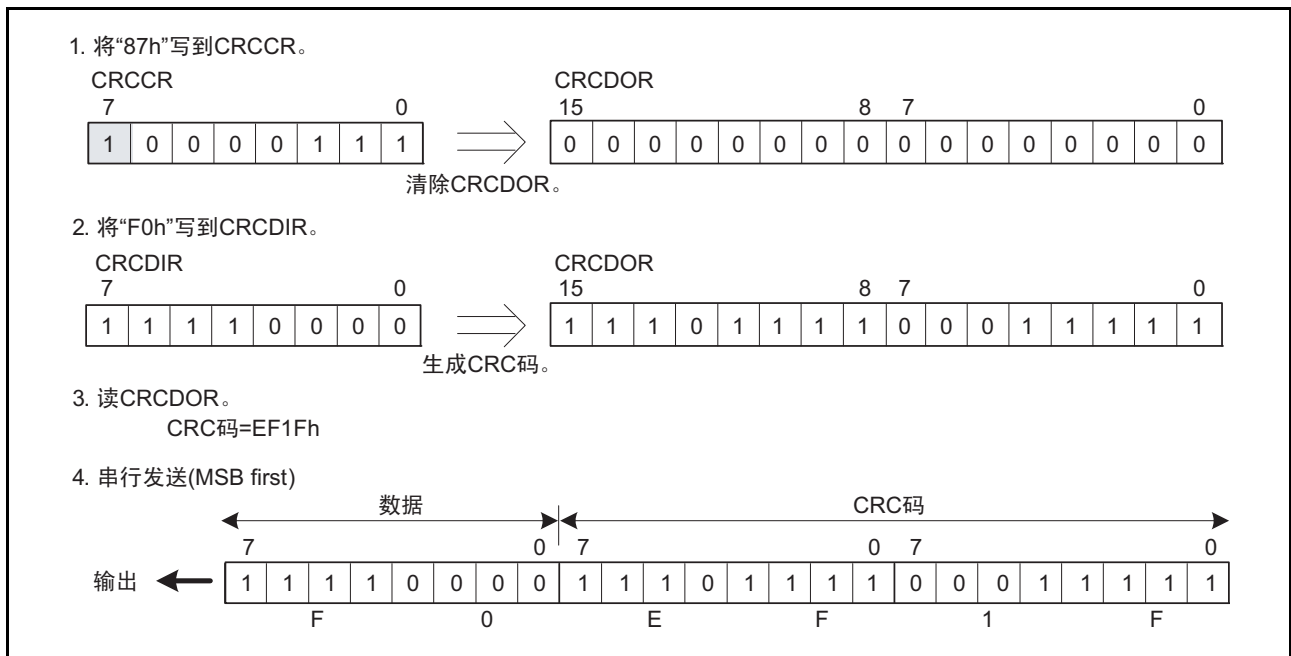


图 31.3 MSB first 的数据发送

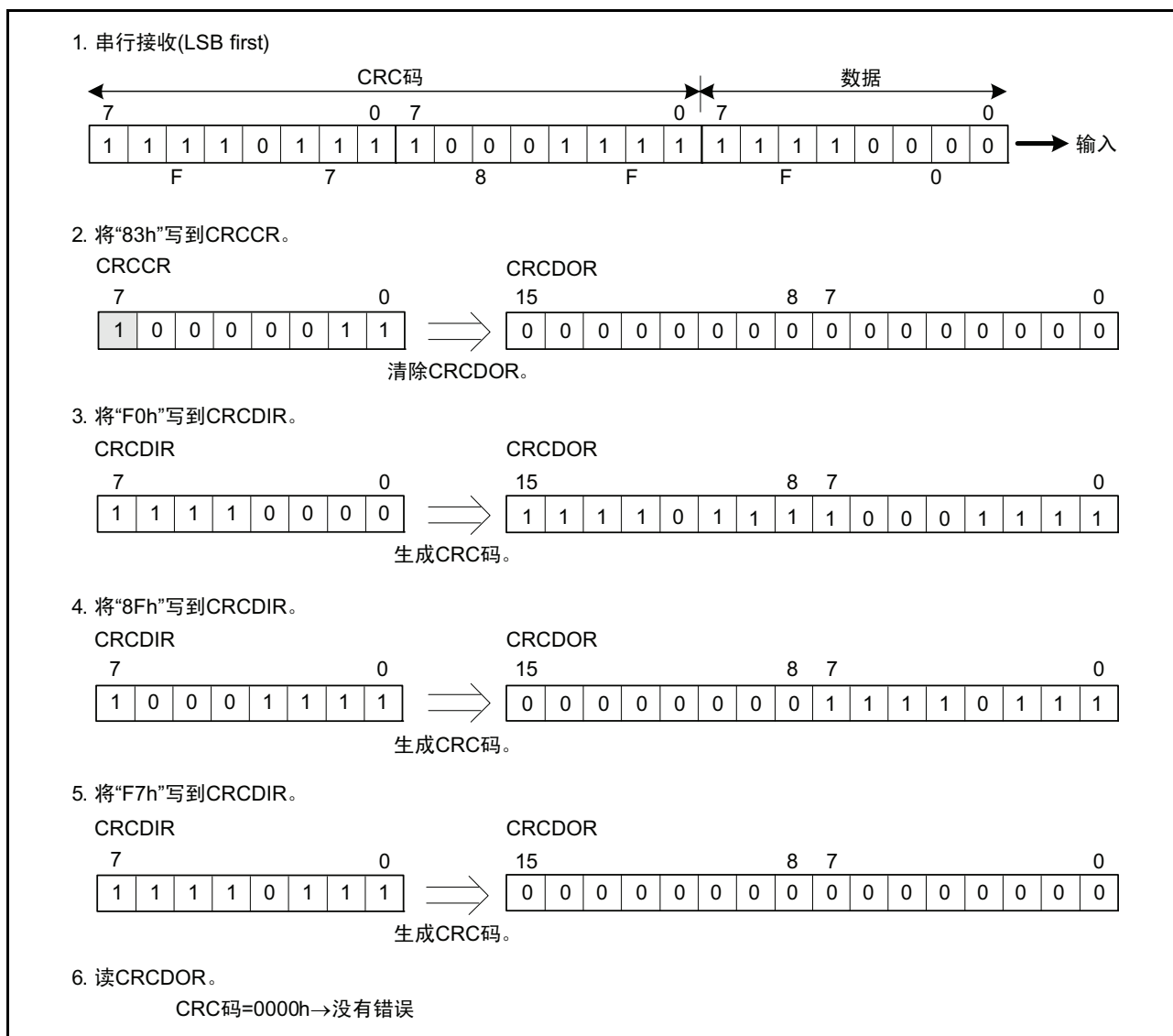


图 31.4 LSB first 的数据接收

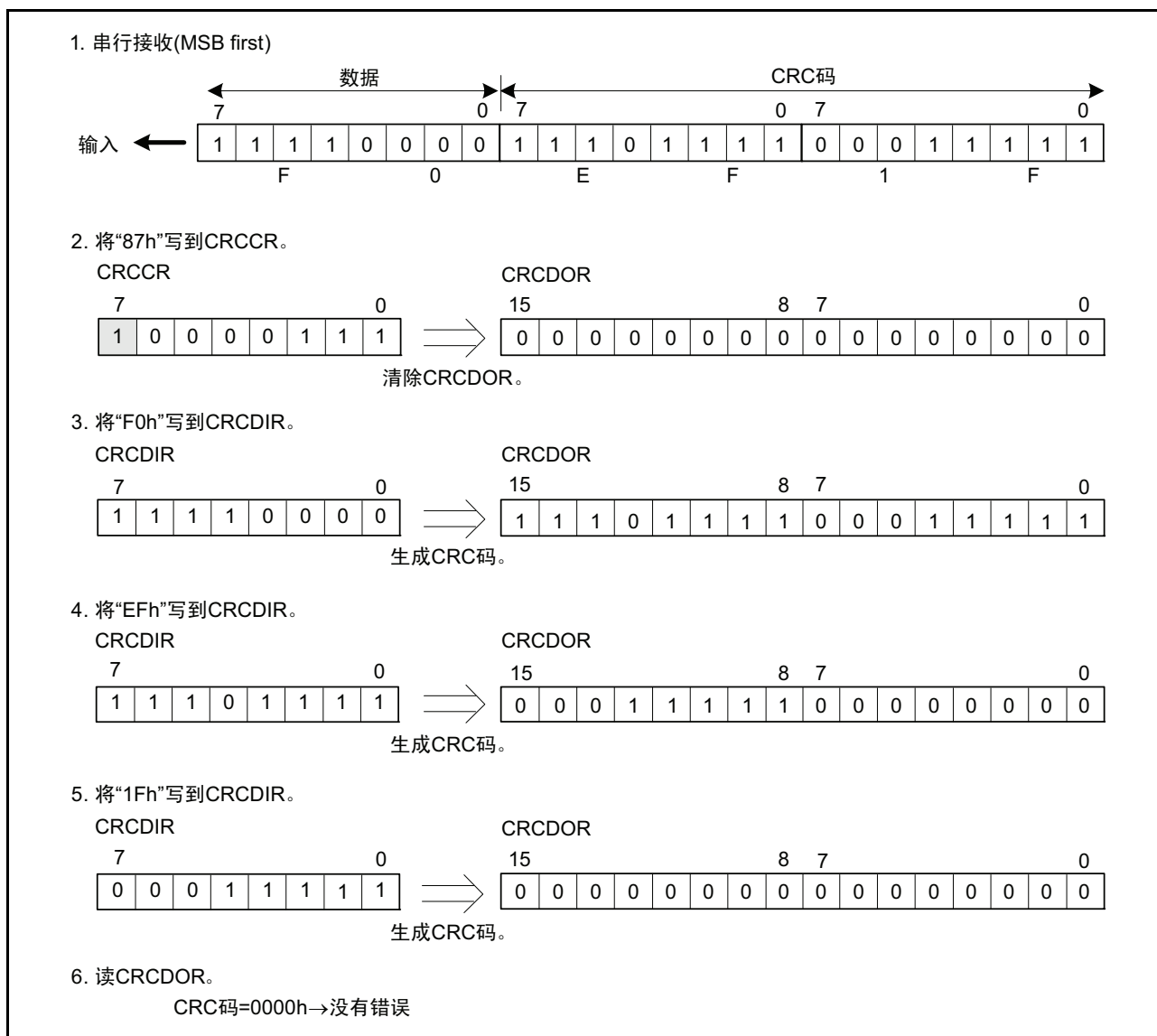


图 31.5 MSB first 的数据接收

31.4 使用时的注意事项

31.4.1 模块停止功能的设定

能通过模块停止控制寄存器 B (MSTPCRB) 禁止或者允许 CRC 运算器的运行, 初始值为停止 CRC 运算器的运行。能通过解除模块停止状态, 使寄存器变为可存取的状态。详细内容请参照“11. 低功耗功能”。

31.5 传送时的注意事项

必须注意: LSB first 发送和 MSB first 发送时的 CRC 码的发送顺序不同。

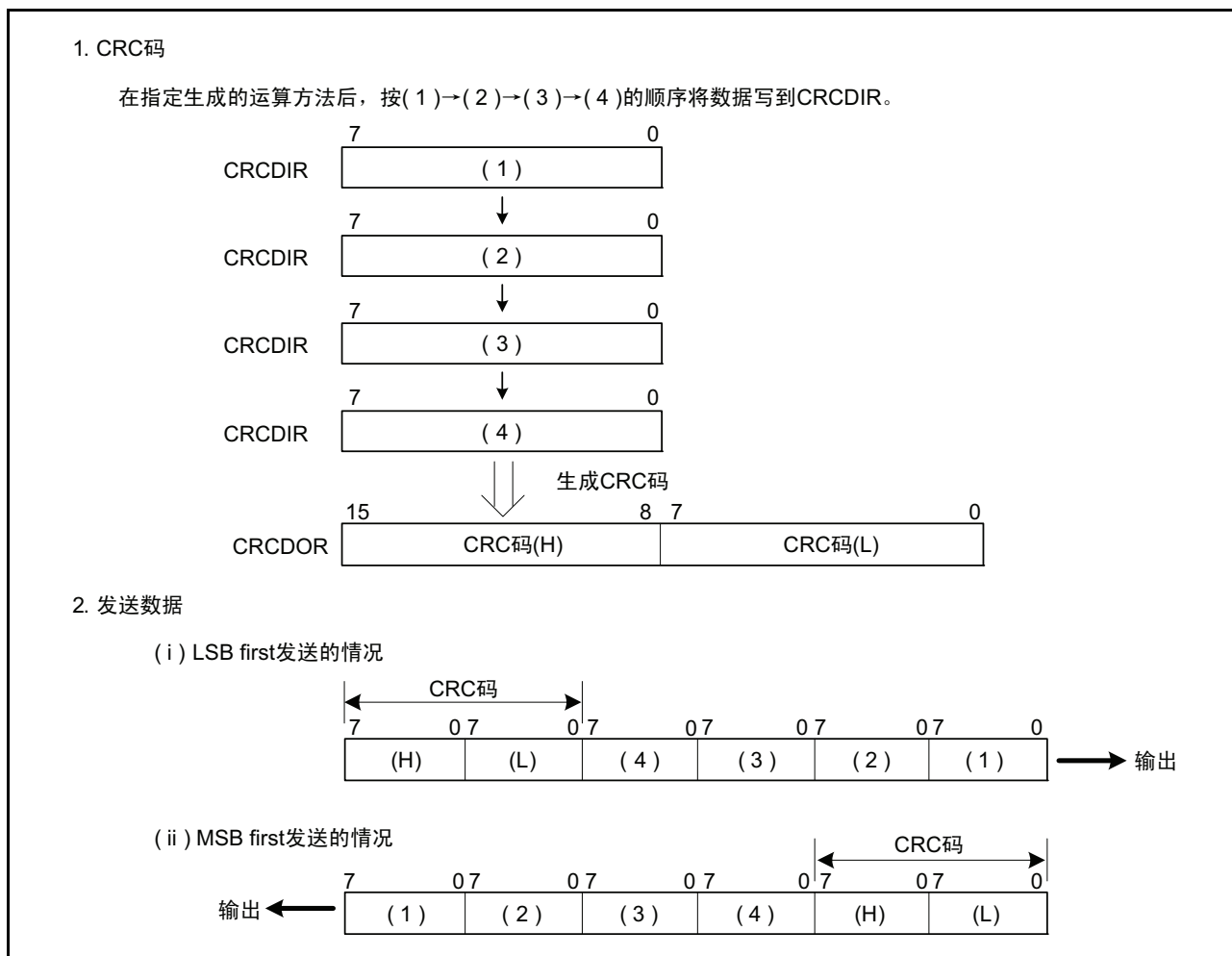


图 31.6 LSB first 和 MSB first 的发送数据

32. 12 位 A/D 转换器 (S12ADb)

32.1 概要

RX210 群内置 1 个单元的逐次逼近方式的 12 位 A/D 转换器，能选择最多 16 个通道的模拟输入、温度传感器输出或者内部基准电压。

12 位 A/D 转换器通过逐次逼近方式将选择的最多 16 个通道的模拟输入、温度传感器输出或者内部基准电压转换为 12 位数字值。A/D 转换器的运行模式有单次扫描模式（按照从小到大的通道号顺序对任意选择的最多 16 个通道的模拟输入只进行 1 次转换）、连续扫描模式（按照从小到大的通道号顺序对任意选择的最多 16 个通道的模拟输入进行连续转换）和分组扫描模式（将任意选择的最多 16 个通道的模拟输入分为 2 组（组 A 和组 B），然后以组为单位按照从小到大的通道号顺序对所选通道的模拟输入进行转换）。

在分组扫描模式中，能通过分别选择组 A 和组 B 的扫描开始条件，在不同时序开始组 A 和组 B 的 A/D 转换。

在双触发模式中，通过单次扫描模式或者分组扫描模式（组 A）对任意选择的 1 个通道的模拟输入进行转换，并且将通过第 1 次 A/D 转换开始触发转换的数据和通过第 2 次 A/D 转换开始触发转换的数据分别保存到不同的寄存器（A/D 转换数据的双重化）。

在每次扫描的最开始执行 1 次自诊断，对在 12 位 A/D 转换器内部生成的 3 个电压值中的 1 个进行 A/D 转换。

温度传感器输出和内部基准电压分别单独进行 A/D 转换。

12 位 A/D 转换器的规格、功能概要和框图分别如表 32.1、表 32.2 和图 32.1 所示。

表 32.1 12 位 A/D 转换器的规格 (1/2)

项目	内容
单元数	1 个单元
输入通道	16 个通道
扩展模拟输入	温度传感器输出、内部基准电压
A/D 转换方式	逐次逼近方式
分辨率	12 位
转换时间	每个通道 1.0 μ s（A/D 转换时钟，以 ADCLK=50MHz 运行时）
A/D 转换时钟	能用以下的分频比设定外围模块时钟 PCLK（注 1）和 A/D 转换时钟 ADCLK（注 1） PCLK:ADCLK 分频比 = 1:1、1:2、1:4、1:8、2:1、4:1 在时钟发生电路中设定 ADCLK。
数据寄存器	用于模拟输入：16 个 用于双触发模式中的 A/D 转换数据双重化：1 个 用于温度传感器输出：1 个 用于内部基准电压：1 个 将 A/D 转换结果保存到 12 位 A/D 数据寄存器。 在加法运算模式中，将 14 位 A/D 转换结果的加法运算值保存到 A/D 数据寄存器。 A/D 转换数据的双重化 • 将所选 1 个通道的模拟输入的第 1 次 A/D 转换数据保存到 A/D 数据寄存器 y，第 2 次 A/D 转换数据保存到双重化寄存器。 • 在单次扫描模式和分组扫描模式中，只有在选择双触发模式时才能执行双重化。

表 32.1 12 位 A/D 转换器的规格 (2/2)

项目	内容
运行模式	<ul style="list-style-type: none"> • 单次扫描模式： 对任意选择的最多 16 个通道的模拟输入只进行 1 次 A/D 转换。 对温度传感器输出只进行 1 次 A/D 转换。 对内部基准电压只进行 1 次 A/D 转换。 • 连续扫描模式：对任意选择的最多 16 个通道的模拟输入重复进行 A/D 转换（注 2）。 • 分组扫描模式 将最多 16 个通道的模拟输入分为组 A 和组 B，并且以组为单位对所选全部通道的模拟输入只进行 1 次 A/D 转换。 能通过分别选择组 A 和组 B 的转换开始条件，在不同的时序开始组 A 和组 B 的 A/D 转换。
A/D 转换开始条件	<ul style="list-style-type: none"> • 软件触发 • 同步触发 MTU、ELC 或者温度传感器的触发 • 异步触发 能通过 ADTRG0# 引脚开始 A/D 转换。
功能	<ul style="list-style-type: none"> • 采样 & 保持功能 • 通道专用采样 & 保持功能 • 采样状态数可变功能 • 12 位 A/D 转换器的自诊断功能 • A/D 转换值加法运算模式 • 模拟输入断线检测辅助功能 • 双触发模式（A/D 转换数据双重化功能）
中断源	<ul style="list-style-type: none"> • 除双触发模式和分组扫描模式外，在 1 次扫描结束时产生扫描结束中断请求（S12ADI0）。 • 在双触发模式中，在 2 次扫描结束时产生扫描结束中断请求（S12ADI0）。 • 在分组扫描模式中，在组 A 的扫描结束时产生扫描结束中断请求（S12ADI0）；在组 B 的扫描结束时产生组 B 专用的扫描结束中断请求（GBADI）。 • 在分组扫描模式的双触发模式中，在组 A 的 2 次扫描结束时产生扫描结束中断请求（S12ADI0）；在组 B 的扫描结束时产生组 B 专用的扫描结束中断请求（GBADI）。 • 能通过 S12ADI0 中断或者 GBADI 中断来启动 DMA 控制器（DMAC）或者数据传送控制器（DTC）。
事件链接功能	<ul style="list-style-type: none"> • 除分组扫描模式中的组 B 的扫描外，能在扫描结束时发生 ELC 事件。 • 能通过 ELC 的触发开始 A/D 转换。
低功耗功能	<ul style="list-style-type: none"> • 能设定为模块停止状态（注 3）。

注 1. 外围模块时钟 PCLK 变为 SCKCR.PCKB[3:0] 位设定的频率，A/D 转换时钟 ADCLK 变为 SCKCR.PCKD[3:0] 位设定的频率。

注 2. 如果选择温度传感器输出或者内部基准电压，就不能使用连续扫描模式和分组扫描模式。

注 3. 在解除模块停止状态后，必须在等待 1 μ s 后开始 A/D 转换。

表 32.2 12 位 A/D 转换器的功能概要

项目			功能	
模拟输入通道			AN000 ~ AN015、温度传感器输出、内部基准电压	
A/D 转换 开始条件	软件	软件触发	能	
	异步触发	ADTRG0#	能	
	同步触发	MTU0 的 TRGA 的比较匹配 / 输入捕捉		TRG0AN
				TRG0BN
		MTU0 ~ MTU4 的 TRGA 的比较匹配 / 输入捕捉或者互补 PWM 模式中的 MTU4.TCNT 的下溢 (波谷)		TRGAN
		MTU0 的 TRGE 的比较匹配		TRG0EN
		MTU0 的 TRGF 的比较匹配		TRG0FN
		MTU4.TADCORA 和 MTU4.TCNT 的比较匹配 (中断减少功能 1)		TRG4AN
		MTU4.TADCORB 和 MTU4.TCNT 的比较匹配 (中断减少功能 1)		TRG4BN
		MTU4.TADCORA 和 MTU4.TCNT、MTU4.TADCORB 和 MTU4.TCNT 的比较匹配 (中断减少功能 1)		TRG4ABN
		ELC 的触发		能
		温度传感器的触发		能
中断			S12ADI0 中断、GBADI 中断	
模块停止功能的设定 (注 1)			MSTPCRA.MSTPA17 位	

注 1. 详细内容请参照“11. 低功耗功能”。

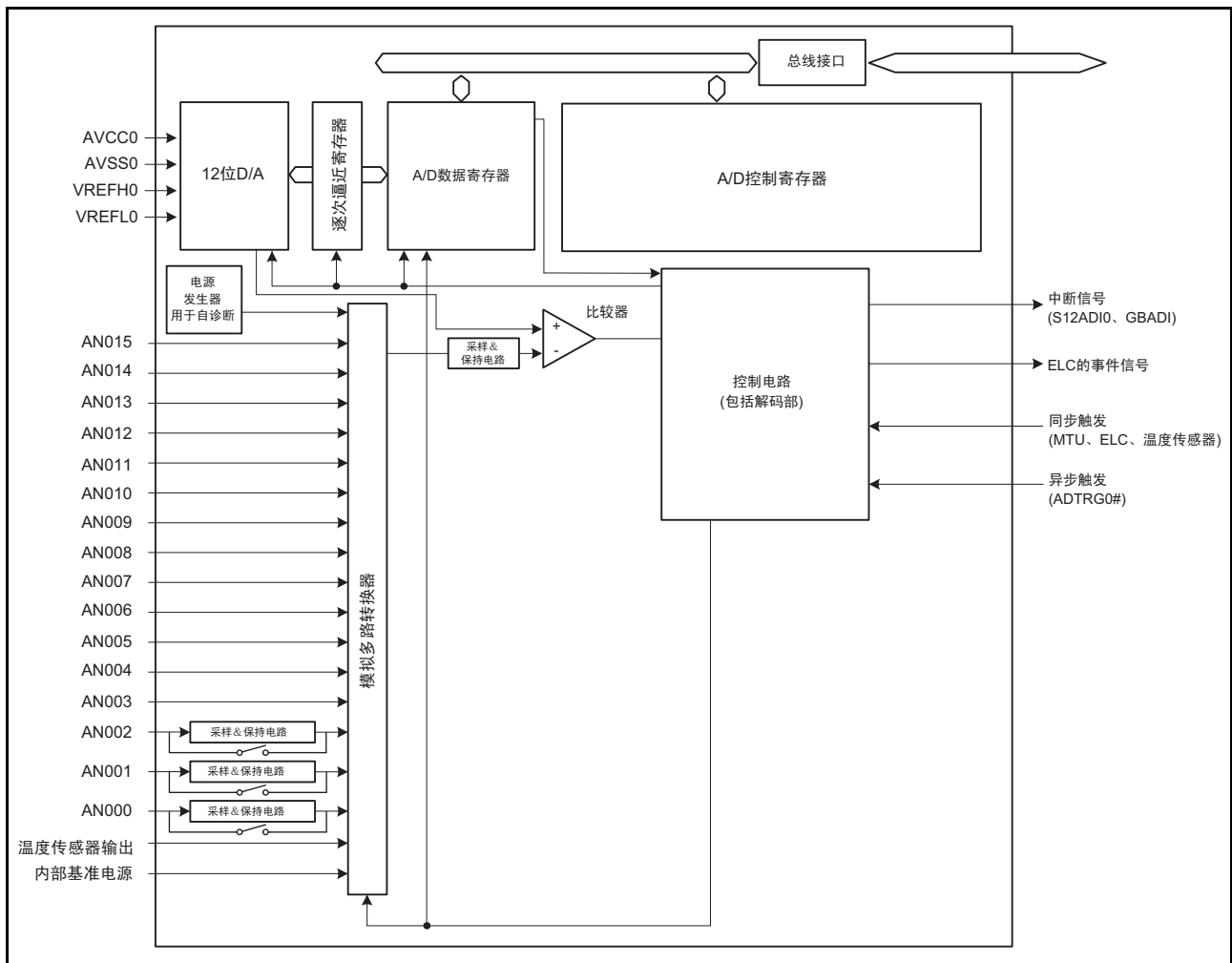


图 32.1 12 位 A/D 转换器的框图

12 位 A/D 转换器使用的输入引脚如表 32.3 所示。

表 32.3 12 位 A/D 转换器的输入引脚

引脚名	输入	功能
AVCC0	输入	模拟部的电源引脚
AVSS0	输入	模拟部的接地引脚
VREFH0	输入	基准电源引脚
VREFL0	输入	基准电源接地引脚
AN000 ~ AN015	输入	模拟输入引脚 (AN000、AN001、AN002 有通道专用采样 & 保持功能)
ADTRG0#	输入	用于开始 A/D 转换的外部触发输入引脚

32.2 寄存器说明

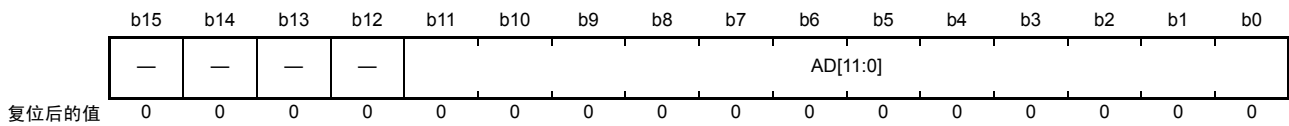
32.2.1 A/D 数据寄存器 y (ADDRy) (y=0 ~ 15)

ADDRy 寄存器是保存通道的模拟输入 (AN000 ~ AN015) 的 A/D 转换结果的 16 位只读寄存器。

根据 A/D 数据寄存器格式选择位 (ADCER.ADRFMT) 和 A/D 转换值加法运算模式的设定, ADDRy 寄存器的格式如下所示。

- ADCER.ADRFMT=0 (设定为往右靠紧的格式)

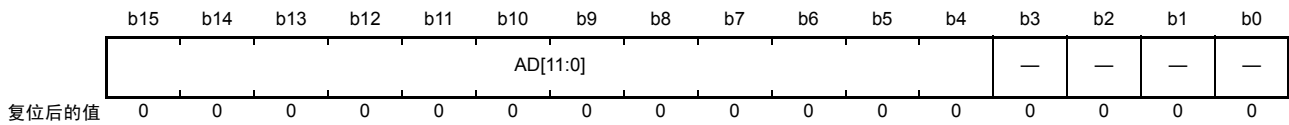
地址 ADDR0: 0008 9020h ~ ADDR15: 0008 903Eh



位	符号	位名	功能	R/W
b11-b0	AD[11:0]	—	12 位 A/D 转换值	R
b15-b12	—	保留位	读写值都为“0”。	R/W

- ADCER.ADRFMT=1 (设定为往左靠紧的格式)

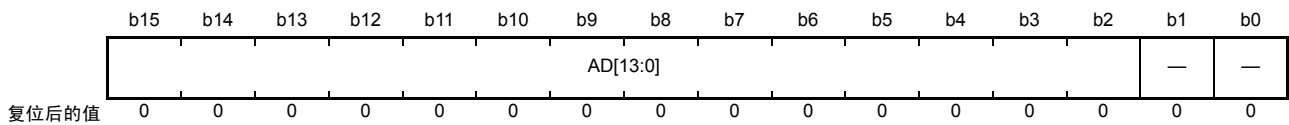
地址 ADDR0: 0008 9020h ~ ADDR15: 0008 903Eh



位	符号	位名	功能	R/W
b3-b0	—	保留位	读写值都为“0”。	R/W
b15-b4	AD[11:0]	—	12 位 A/D 转换值	R

- 选择 A/D 转换值加法运算模式的情况

地址 ADDR0: 0008 9020h ~ ADDR15: 0008 903Eh



位	符号	位名	功能	R/W
b1-b0	—	保留位	读写值都为“0”。	R/W
b15-b2	AD[13:0]	—	14 位 A/D 转换值的加法运算结果	R

如果设定为 A/D 转换值加法运算模式，ADDRy.AD[13:0] 位就表示对相同通道的 A/D 转换值进行加法运算后的值。而且 ADCER.ADRFMT 位的设定无效，为往左靠紧的格式。

选择 A/D 转换值加法运算模式的通道的最小值和最大值如下所示：

- 1 次转换时：0000h \leq ADDRy (y=0 ~ 15) \leq 3FFCh
ADDRy (y=0 ~ 15)：bit15、bit14=00b、bit13 ~ bit2=AD[11:0]、bit1、bit0=00b
- 2 次转换时：0000h \leq ADDRy (y=0 ~ 15) \leq 7FF8h
ADDRy (y=0 ~ 15)：bit15=0、bit14 ~ bit2=AD[12:0]、bit1、bit0=00b
- 3 次转换时：0000h \leq ADDRy (y=0 ~ 15) \leq BFF4h
ADDRy (y=0 ~ 15)：bit15 ~ bit2=AD[13:0]、bit1、bit0=00b
- 4 次转换时：0000h \leq ADDRy (y=0 ~ 15) \leq FFF0h
ADDRy (y=0 ~ 15)：bit15 ~ bit2=AD[13:0]、bit1、bit0=00b

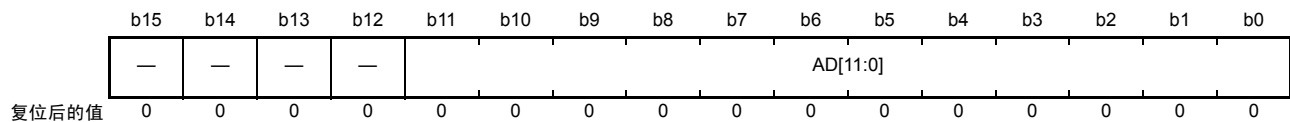
32.2.2 A/D 数据双重化寄存器 (ADDBLDR)

ADDBLDR 寄存器是 16 位只读寄存器，用于保存通过双触发模式中的第 2 次 A/D 转换开始触发对选择双重化的通道的模拟输入进行 A/D 转换后的结果。

根据 A/D 数据寄存器格式选择位 (ADCER.ADRFMT) 和 A/D 转换值加法运算模式的设定，ADDBLDR 寄存器的格式如下所示。

- ADCER.ADRFMT=0 (设定为往右靠紧的格式)

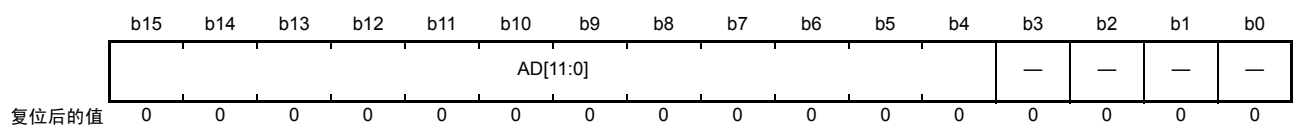
地址 0008 9018h



位	符号	位名	功能	R/W
b11-b0	AD[11:0]	—	12 位 A/D 转换值	R
b15-b12	—	保留位	读写值都为“0”。	R/W

- ADCER.ADRFMT=1 (设定为往左靠紧的格式)

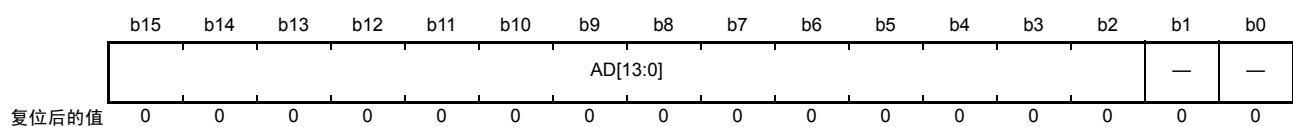
地址 0008 9018h



位	符号	位名	功能	R/W
b3-b0	—	保留位	读写值都为“0”。	R/W
b15-b4	AD[11:0]	—	12 位 A/D 转换值	R

- 选择 A/D 转换值加法运算模式的情况

地址 0008 9018h



位	符号	位名	功能	R/W
b1-b0	—	保留位	读写值都为“0”。	R/W
b15-b2	AD[13:0]	—	14 位 A/D 转换值的加法运算结果	R

如果设定为 A/D 转换值加法运算模式，ADDBLDR.AD[13:0] 位就表示对相同通道的 A/D 转换值进行加法运算后的值。而且 ADCER.ADRFMT 位的设定无效，为往左靠紧的格式。

32.2.3 A/D 温度传感器的数据寄存器 (ADTSDR)

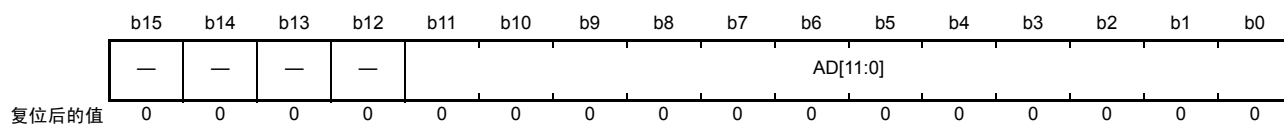
ADTSDR 寄存器是保存温度传感器输出的 A/D 转换结果的 16 位只读寄存器。

根据 A/D 数据寄存器格式选择位 (ADCER.ADRFMT) 的设置, ADTSDR 寄存器的格式如下所示。

ADTSDR 寄存器不能设定为 A/D 转换加法运算模式。

- ADCER.ADRFMT=0 (设定为往右靠紧的格式)

地址 0008 901Ah



位	符号	位名	功能	R/W
b11-b0	AD[11:0]	—	12 位 A/D 转换值	R
b15-b12	—	保留位	读写值都为“0”。	R/W

- ADCER.ADRFMT=1 (设定为往左靠紧的格式)

地址 0008 901Ah



位	符号	位名	功能	R/W
b3-b0	—	保留位	读写值都为“0”。	R/W
b15-b4	AD[11:0]	—	12 位 A/D 转换值	R

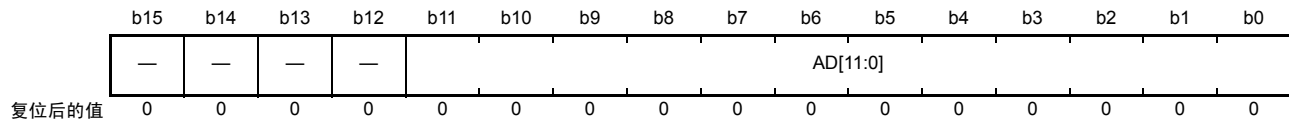
32.2.4 A/D 内部基准电压的数据寄存器 (ADOCDR)

ADOCDR 寄存器是保存内部基准电压的 A/D 转换结果的 16 位只读寄存器。

根据 A/D 数据寄存器格式选择位 (ADCER.ADRFMT) 和 A/D 转换值加法运算模式的设定, ADOCDR 寄存器的格式如下所示。

- ADCER.ADRFMT=0 (设定为往右靠紧的格式)

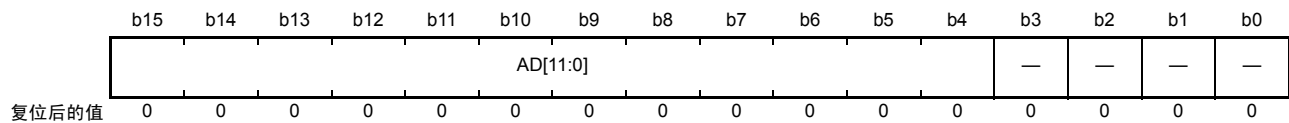
地址 0008 901Ch



位	符号	位名	功能	R/W
b11-b0	AD[11:0]	—	12 位 A/D 转换值	R
b15-b12	—	保留位	读写值都为“0”。	R/W

- ADCER.ADRFMT=1 (设定为往左靠紧的格式)

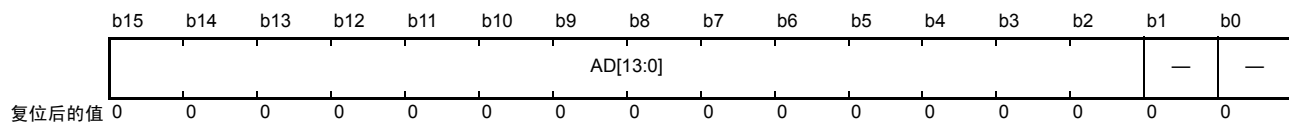
地址 0008 901Ch



位	符号	位名	功能	R/W
b3-b0	—	保留位	读写值都为“0”。	R/W
b15-b4	AD[11:0]	—	12 位 A/D 转换值	R

- 选择 A/D 转换值加法运算模式的情况

地址 0008 901Ch



位	符号	位名	功能	R/W
b1-b0	—	保留位	读写值都为“0”。	R/W
b15-b2	AD[13:0]	—	14 位 A/D 转换值的加法运算结果	R

如果设定为 A/D 转换值加法运算模式, ADOCDR.AD[13:0] 位就表示对内部基准电压的 A/D 转换值进行加法运算后的值。而且 ADCER.ADRFMT 位的设定无效, 为往左靠紧的格式。

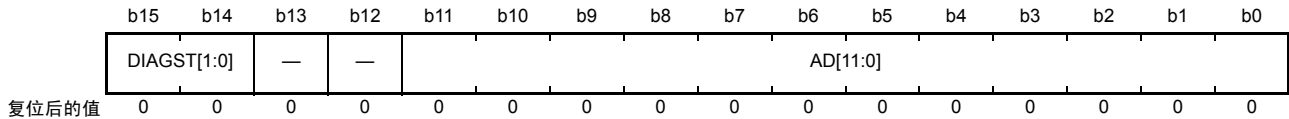
32.2.5 A/D 自诊断数据寄存器 (ADRD)

ADRD 寄存器是 16 位只读寄存器，用于保存通过 12 位 A/D 转换器的自诊断进行 A/D 转换后的结果。

根据 A/D 数据寄存器格式选择位 (ADCER.ADRFMT) 的设定，ADRD 寄存器的格式如下所示。ADRD 寄存器不能设定为 A/D 转换加法运算模式。

- ADCER.ADRFMT=0 (设定为往右靠紧的格式)

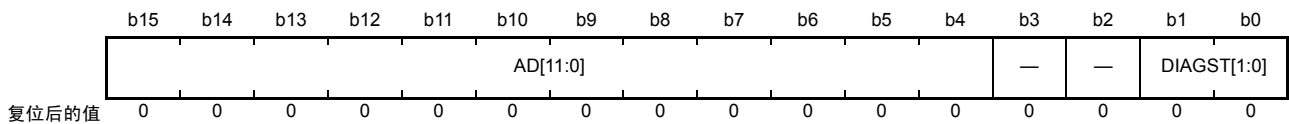
地址 0008 901Eh



位	符号	位名	功能	R/W
b11-b0	AD[11:0]	—	12 位 A/D 转换值	R
b13-b12	—	保留位	读写值都为“0”。	R/W
b15-b14	DIAGST[1:0]	自诊断状态位	b ¹⁵ b ¹⁴ 0 0: 表示自通电后还未执行 1 次自诊断 0 1: 表示执行了电压值为 0V 的自诊断 1 0: 表示执行了电压值为 VREFH0×1/2 的自诊断 1 1: 表示执行了电压值为 VREFH0 的自诊断 有关自诊断的详细内容，请参照“32.2.11 A/D 控制扩展寄存器 (ADCER)”。	R

- ADCER.ADRFMT=1 (设定为往左靠紧的格式)

地址 0008 901Eh



位	符号	位名	功能	R/W
b1-b0	DIAGST[1:0]	自诊断状态位	b ¹ b ⁰ 0 0: 表示自通电后还未执行 1 次自诊断 0 1: 表示执行了电压值为 0V 的自诊断 1 0: 表示执行了电压值为 VREFH0×1/2 的自诊断 1 1: 表示执行了电压值为 VREFH0 的自诊断 有关自诊断的详细内容，请参照“32.2.11 A/D 控制扩展寄存器 (ADCER)”。	R
b3-b2	—	保留位	读写值都为“0”。	R/W
b15-b4	AD[11:0]	—	12 位 A/D 转换值	R

32.2.6 A/D 控制寄存器 (ADCSR)

地址 0008 9000h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
ADST	ADCS[1:0]	ADIE	—	—	TRGE	EXTRG	DBLE	GBADIE	—	DBLANS[4:0]					
复位后的值	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b4-b0	DBLANS [4:0]	A/D 转换数据双重化的通道选择位	从 16 个通道中选择 1 个通道的模拟输入进行 A/D 转换数据的双重化。只在选择双触发模式时有效。	R/W
b5	—	保留位	读写值都为“0”。	R/W
b6	GBADIE	组 B 扫描结束中断允许位	0: 禁止在组 B 的扫描结束后发生 GBADI 中断 1: 允许在组 B 的扫描结束后发生 GBADI 中断	R/W
b7	DBLE	双触发模式选择位	0: 不选择双触发模式 1: 选择双触发模式	R/W
b8	EXTRG	触发选择位 (注 1)	0: 选择通过同步触发 (MTU、ELC、温度传感器) 开始 A/D 转换 1: 选择通过异步触发 (ADTRG0#) 开始 A/D 转换	R/W
b9	TRGE	触发启动允许位	0: 禁止通过同步触发、异步触发开始 A/D 转换 1: 允许通过同步触发、异步触发开始 A/D 转换	R/W
b10-b11	—	保留位	读写值都为“0”。	R/W
b12	ADIE	扫描结束中断允许位	0: 禁止在扫描结束后发生 S12ADI0 中断 1: 允许在扫描结束后发生 S12ADI0 中断	R/W
b14-b13	ADCS[1:0]	扫描模式选择位	b14 b13 0 0: 单次扫描模式 0 1: 分组扫描模式 1 0: 连续扫描模式 1 1: 禁止设定	R/W
b15	ADST	A/D 转换开始位	0: 停止 A/D 转换 1: 开始 A/D 转换	R/W

注 1. 通过外部引脚 (异步触发) 开始 A/D 转换的方法:

在给外部引脚 (ADTRG0#) 输入 High 电平的状态下, 将 ADCSR.TRGE 位置“1”并且将 ADCSR.EXTRG 位置“1”。此后, 如果使 ADTRG0# 的信号变为 Low 电平, 就检测到 ADTRG0# 的下降沿并且开始扫描转换。此时, Low 电平输入的脉宽至少需要 1.5 个 PCLK 时钟。

DBLANS[4:0] 位 (A/D 转换数据双重化的通道选择位)

这些位选择在双触发模式中进行 A/D 转换数据双重化的 1 个通道。将通过双触发模式的第 1 次 A/D 转换开始触发对 DBLANS[4:0] 位选择的通道的模拟输入进行 A/D 转换后的结果保存到 A/D 数据寄存器 y, 将通过第 2 次 A/D 转换开始触发进行的 A/D 转换的结果保存到 A/D 数据双重化寄存器。A/D 转换数据双重化的通道选择表如表 32.4 所示。在双触发模式的加法运算模式中, 能通过 ADADS 寄存器选择 DBLANS[4:0] 位选择的通道。如果选择双触发模式, ADANSA 寄存器选择的通道就变为无效, 而 DBLANS[4:0] 位选择的 1 个通道变为进行 A/D 转换的通道。对通道的模拟输入进行 A/D 转换时, 不能选择温度传感器输出和内部基准电压的 A/D 转换。必须在 ADST 位为“0”时设定 DBLANS[4:0] 位 (不能在给 ADST 位写“1”的同时进行设定)。

表 32.4 DBLANS[4:0] 位的设定值与双触发有效通道的关系

DBLANS[4:0]	双重化通道	DBLANS[4:0]	双重化通道
00000	AN000	01000	AN008
00001	AN001	01001	AN009
00010	AN002	01010	AN010
00011	AN003	01011	AN011
00100	AN004	01100	AN012
00101	AN005	01101	AN013
00110	AN006	01110	AN014
00111	AN007	01111	AN015

GBADIE 位 (组 B 扫描结束中断允许位)

此位允许或者禁止发生分组扫描模式中的组 B 扫描结束中断 (GBADI)。

DBLE 位 (双触发模式选择位)

双触发模式以 ADSTRGR.TRSA[3:0] 位选择的 MTU 触发或者 ELC 触发开始扫描为条件进行以下两个运行。

1. 如果将 ADIE 位置“1”，在第 1 次扫描结束时就不输出扫描结束中断，而在第 2 次扫描结束时输出扫描结束中断。
2. 将 DBLANS[4:0] 位选择的通道的模拟输入的第 1 次 A/D 转换数据保存到 A/D 数据寄存器 y，第 2 次 A/D 转换数据保存到 A/D 数据双重化寄存器。

如果将 DBLE 位置“1”，ADANSA 寄存器选择的通道就变为无效。在连续扫描模式中不能选择双触发模式，而且也不能选择温度传感器输出和内部基准电压的 A/D 转换。如果选择双触发模式，就不能使用软件触发。必须在 ADST 位为“0”时设定 DBLE 位（不能在给 ADST 位写“1”的同时进行设定）。

TRGE 位 (触发启动允许位)

此位允许或者禁止通过同步触发或者异步触发开始 A/D 转换。
在分组扫描模式中，必须将此位置“1”。

ADIE 位 (扫描结束中断允许位)

除分组扫描模式中的组 B 外，此位允许或者禁止发生 A/D 扫描转换结束中断 (S12ADIO)。

在设定为不选择双触发模式的情况下，如果在 1 次扫描结束时 ADIE 位为“1”，就发生 S12ADIO 中断。

在选择温度传感器输出或者内部基准电压的情况下，如果在 A/D 转换结束时 ADIE 位为“1”，也发生 S12ADIO 中断。

在选择双触发模式的情况下，只在通过 ADSTRGR.TRSA[3:0] 位选择的 MTU 触发或者 ELC 触发开始扫描时，如果在第 2 次扫描结束时 ADIE 位为“1”，就发生 S12ADIO 中断。

ADCS[1:0] 位 (扫描模式选择位)

这些位选择扫描转换模式。

在单次扫描模式中，按照从小到大的通道号顺序对 ADANSA 寄存器选择的最多 16 个通道的模拟输入进行 A/D 转换，如果所选通道的转换全部结束，就停止扫描转换。

在连续扫描模式中，在 ADCSR.ADST 位为“1”的期间，按照从小到大的通道号顺序对 ADANSA 寄存器选择的最多 16 个通道的模拟输入进行 A/D 转换，如果所选通道的转换全部结束，就返回到最初的通道继续进行 A/D 转换。如果将 ADCSR.ADST 位置“0”，就停止 A/D 转换。

在分组扫描模式中，将 ADSTRGR.TRSA[3:0] 位选择的 MUT 触发或者 ELC 触发作为 A/D 转换开始条件，按照从小到大的通道号顺序对 ADANSA 寄存器选择的最多 16 个通道的模拟输入（组 A）进行 A/D 转换，如果所选通道的 A/D 转换全部结束，就停止 A/D 转换；同样，将 ADSTRGR.TRSA[3:0] 位选择的 MUT 触发或者 ELC 触发作为 A/D 转换开始条件，按照从小到大的通道号顺序对 ADANSB 寄存器选择的最多 16 个通道的模拟输入（组 B）进行 A/D 转换，如果所选通道的 A/D 转换全部结束，就停止 A/D 转换。在选择分组扫描模式时，必须给组 A 和组 B 选择不同的通道和不同的触发。

在单次扫描模式中选择温度传感器输出或者内部基准电压时，必须在将 ADANSA 寄存器选择的通道全部设定为不选择后进行 A/D 转换。如果选择的温度传感器输出或者内部基准电压的 A/D 转换结束，就停止 A/D 转换。

必须在 ADST 位为“0”时设定 ADCS 位（不能在给 ADST 位写“1”的同时进行设定）。

ADST 位 (A/D 转换开始位)

此位控制 A/D 转换的开始或者停止。

必须在将 ADST 位置“1”前设定 A/D 转换时钟、转换模式和转换对象模拟输入。

[为“1”的条件]

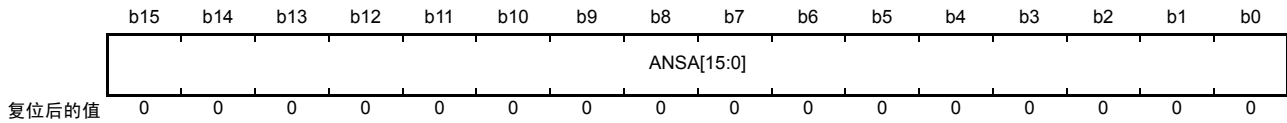
- 当通过软件写“1”时
- 将 ADCSR.EXTRG 位置“0”，将 ADCSR.TRGE 位置“1”并且检测到 ADSTRGR.TRSA[3:0] 位选择的同步触发（MTU、ELC、温度传感器）时
- 在分组扫描模式中将 ADCSR.TRGE 位置“1”并且检测到 ADSTRGR.TRSA[3:0] 位选择的同步触发（MTU、ELC）时
- 将 ADCSR.TRGE 位和 ADCSR.EXTRG 位都置“1”，将 ADSTRGR.TRSA[3:0] 位置“0000b”并且检测到异步触发时

[为“0”的条件]

- 当通过软件写“0”时
- 在单次扫描模式中所选通道的 A/D 转换全部结束时
- 在单次扫描模式中温度传感器输出或者内部基准电压的 A/D 转换结束时
- 在分组扫描模式中组 A 的扫描结束时
- 在分组扫描模式中组 B 的扫描结束时

32.2.7 A/D 通道选择寄存器 A (ADANSA)

地址 0008 9004h



位	符号	位名	功能	R/W
b15-b0	ANSA[15:0]	A/D 转换通道选择位	0: 不将 AN000 ~ AN015 作为转换对象 1: 将 AN000 ~ AN015 作为转换对象	R/W

ADANSA 寄存器是选择 A/D 转换通道的模拟输入 AN000 ~ AN015 的寄存器。在分组扫描模式中，选择组 A 的通道。

ANSA[15:0] 位 (A/D 转换通道选择位)

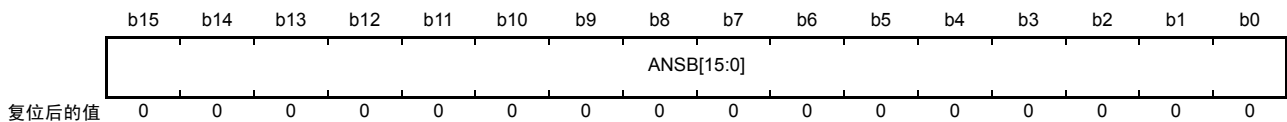
这些位选择 A/D 转换通道的模拟输入 AN000 ~ AN015，能任意设定要选择的通道和通道数。ANSA[0] 位对应 AN000，ANSA[15] 位对应 AN015。如果对通道的模拟输入进行 A/D 转换，就不能对温度传感器输出和内部基准电压进行 A/D 转换。

在选择双触发模式时，ADCSR.DBLANS[4:0] 位选择的 1 个通道为组 A 的通道，ANSA[15:0] 位的设定无效。

必须在 ADCSR.ADST 位为“0”时设定 ANSA[15:0] 位。

32.2.8 A/D 通道选择寄存器 B (ADANSB)

地址 0008 9014h



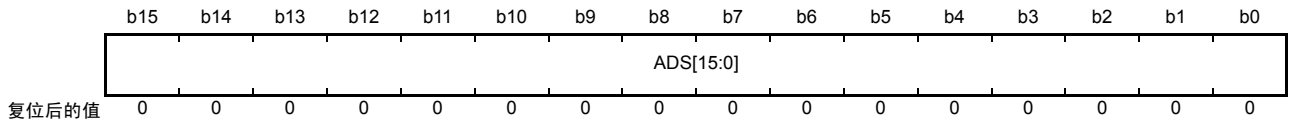
位	符号	位名	功能	R/W
b15-b0	ANSB[15:0]	A/D 转换通道选择位	0: 不将 AN000 ~ AN015 作为转换对象 1: 将 AN000 ~ AN015 作为转换对象	R/W

ADANSB 寄存器选择分组模式中进行组 B 的 A/D 转换的通道 AN000 ~ AN015。ADANSB 寄存器不能在其他模式中使用。从 ADANSA 寄存器或者双触发模式中的 ADCSR.DBLANS[4:0] 位选择的组 A 的通道以外的通道中，设定要选择的通道和通道数。ANSB[0] 位对应 AN000，ANSB[15] 位对应 AN015。如果对通道的模拟输入进行 A/D 转换，就不能对温度传感器输出和内部基准电压进行 A/D 转换。

必须在 ADST 位为“0”时设定 ANSB[15:0] 位。

32.2.9 A/D 转换值加法运算模式选择寄存器 (ADADS)

地址 0008 9008h



位	符号	位名	功能	R/W
b15-b0	ADS[15:0]	A/D 转换值加法运算通道选择位	0: 不选择 AN000 ~ AN015 的 A/D 转换值加法运算模式 1: 选择 AN000 ~ AN015 的 A/D 转换值加法运算模式	R/W

ADADS 寄存器选择连续进行 2 ~ 4 次 A/D 转换后进行加法运算 (累加运算) 的 A/D 转换通道 0 ~ 15。

ADS[15:0] 位 (A/D 转换值加法运算通道选择位)

如果将通道号与 ADANSA.ANSA[n] (n=0 ~ 15) 或者 ADCSR.DBLANS[4:0] 位和 ADANSB.ANSB[n] 位 (n=0 ~ 15) 选择的 A/D 转换通道相同的 ADS[n] 位置“1”，就对所选通道的模拟输入连续进行 ADADC.ADC[1:0] 位所设次数 (2 ~ 4 次) 的 A/D 转换，并且将加法运算 (累加运算) 值返回到 A/D 数据寄存器。通常，对不选择加法运算模式的 A/D 转换通道进行 1 次转换，然后将值返回到 A/D 数据寄存器。

必须在 ADCSR.ADST 位为“0”时设定 ADS[15:0] 位。

将 ADS[2] 位和 ADS[6] 位置“1”后的扫描运行顺序如图 32.2 所示。

在连续扫描模式 (ADCSR.ADCS[1:0]=10b) 中，设定 4 次 (ADADC.ADC[1:0]=11b) 加法运算，并且将 AN000 ~ AN007 作为被选择的对象 (ADANS0.ANSA[15:0]=00FFh)。从 AN000 开始转换，对 AN002 连续进行 4 次转换，然后将加法运算 (累加运算) 值返回到 A/D 数据寄存器 2。此后，开始 AN003 的转换，对 AN006 连续进行 4 次转换，然后将加法运算 (累加运算) 值返回到 A/D 数据寄存器 6。在转换 AN007 后，按照同样的顺序重新从 AN000 开始转换。

不选择加法运算模式的通道的 A/D 数据寄存器格式取决于 ADCER.ADRFMT 位 (往右靠紧 / 往左靠紧) 的设定。

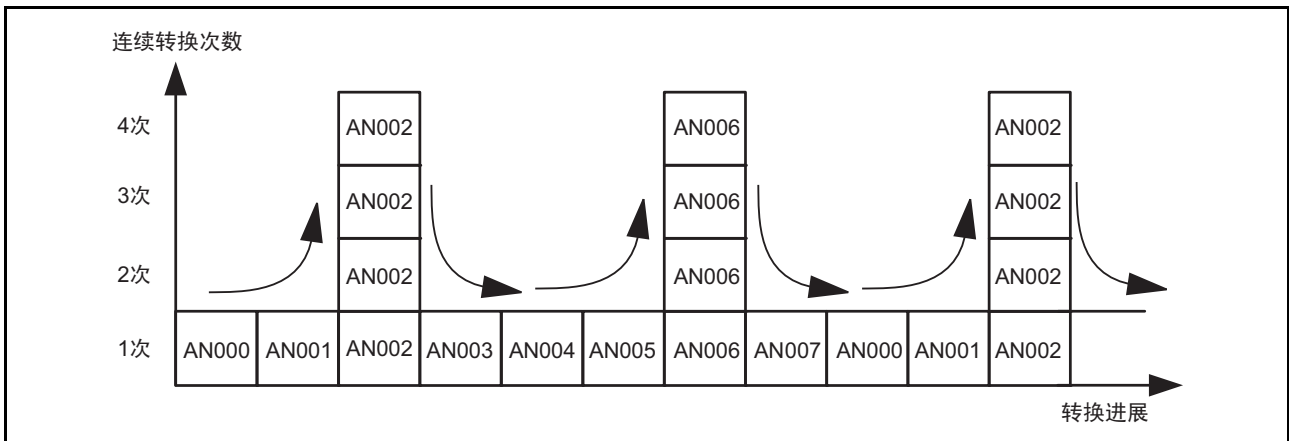
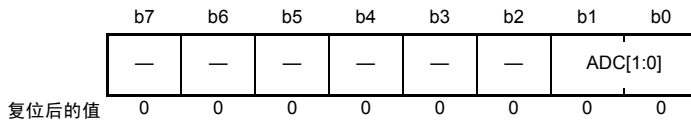


图 32.2 选择 ADADC.ADC[1:0] 为“11b”并且 ADS[2] 和 ADS[6] 都为“1”时的扫描转换顺序

32.2.10 A/D 转换值加法运算次数选择寄存器 (ADADC)

地址 0008 900Ch



位	符号	位名	功能	R/W
b1-b0	ADC[1:0]	加法运算次数选择位	b1 b0 0 0: 转换 1 次 (没有加法运算, 和一般转换一样) 0 1: 转换 2 次 (进行 1 次加法运算) 1 0: 转换 3 次 (进行 2 次加法运算) 1 1: 转换 4 次 (进行 3 次加法运算)	R/W
b7-b2	—	保留位	读写值都为“0”。	R/W

ADADC 寄存器给选择 A/D 转换值加法运算模式的通道和内部基准电压的 A/D 转换设定加法运算次数。

ADC[1:0] 位 (加法运算次数选择位)

给选择 A/D 转换 (包括在双触发模式中, 通过 ADCSR.DBLANS[4:0] 位选择的通道的 A/D 转换) 和加法运算模式的通道以及内部基准电压的 A/D 转换设定共用的加法运算次数。

必须在 ADCSR.ADST 位为“0”时设定 ADC[1:0] 位。

32.2.11 A/D 控制扩展寄存器 (ADCER)

地址 0008 900Eh

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
ADRFMT	—	—	—	DIAGM	DIAGLD	DIAGVAL[1:0]	—	—	ACE	—	—	—	—	—	—
复位后的值	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b4-b0	—	保留位	读写值都为“0”。	R/W
b5	ACE	自动清除允许位	0: 禁止自动清除 1: 允许自动清除	R/W
b7-b6	—	保留位	读写值都为“0”。	R/W
b9-b8	DIAGVAL [1:0]	自诊断转换电压选择位	b9 b8 0 0: 在自诊断有效时禁止设定 0 1: 使用 0V 电压进行自诊断 1 0: 使用 VREFH0×1/2 的电压进行自诊断 1 1: 使用 VREFH0 电压进行自诊断	R/W
b10	DIAGLD	自诊断模式选择位	0: 自诊断电压轮流模式 1: 自诊断电压固定模式	R/W
b11	DIAGM	自诊断运行位	0: 不执行 12 位 A/D 转换器的自诊断 1: 执行 12 位 A/D 转换器的自诊断	R/W
b14-b12	—	保留位	读写值都为“0”。	R/W
b15	ADRFMT	A/D 数据寄存器格式选择位	0: 将 A/D 数据寄存器的格式设定为往右靠紧 1: 将 A/D 数据寄存器的格式设定为往左靠紧	R/W

ACE 位 (自动清除允许位)

此位选择是否通过 CPU、DTC 或者 DMAC 读 ADDRy、ADRD、ADTSDR、ADOCADR、ADDBLDR 寄存器后自动清除对应的寄存器 (All“0”)。能通过自动清除检测到 ADDRy、ADRD、ADTSDR、ADOCADR、ADDBLDR 寄存器未更新的故障。

DIAGVAL[1:0] 位 (自诊断转换电压选择位)

详细内容请参照 ADCER.DIAGLD 位的说明。

在 ADCER.DIAGVAL[1:0] 位为“00b”的状态下将 ADCER.DIAGLD 位置“1”，不执行自诊断。

DIAGLD 位 (自诊断模式选择位)

此位选择是轮流转换通过自诊断转换的 3 个电压值还是固定电压值。

如果将 ADCER.DIAGLD 位置“0”，就按 0V → VREFH0×1/2 → VREFH0 的顺序轮流转换。复位后，如果选择自诊断电压轮流模式，就从 0V 开始自诊断；如果选择自诊断电压固定模式，就固定为 ADCER.DIAGVAL[1:0] 位选择的电压进行转换。在自诊断电压轮流模式中，即使扫描转换结束，也不返回 0V。如果再次进行扫描转换，就接着上一次进行轮流转换。从自诊断电压固定模式转换到自诊断电压轮流模式时，从固定的电压值开始轮流转换。

必须在 ADST 位为“0”时设定 DIAGLD 位。

DIAGM 位 (自诊断允许位)

此位选择是否执行自诊断。

自诊断是为了检测 12 位 A/D 转换器故障的功能。转换在内部生成的 0V、VREFH0×1/2 和 VREFH0 这 3 个电压值中的任意一个电压值。如果转换结束，就将转换的电压信息和转换值保存到自诊断数据寄存器 (ADRD)。之后，通过软件读 ADRD 寄存器，由此判断转换值是 (正常) 否 (异常) 在正常的范围内。在每次扫描的最开始执行自诊断，对 3 个电压值中的 1 个电压值进行 A/D 转换。自诊断的执行时间和 1 个通道的 A/D 转换时间相同。在执行自诊断时，不能选择温度传感器输出和内部基准电压的 A/D 转换。否则，就不执行自诊断。如果在分组扫描模式中选择自诊断，组 A 和组 B 分别执行自诊断。

必须在 ADST 位为“0”时设定 DIAGM 位。

ADRFMT 位 (A/D 数据寄存器格式选择位)

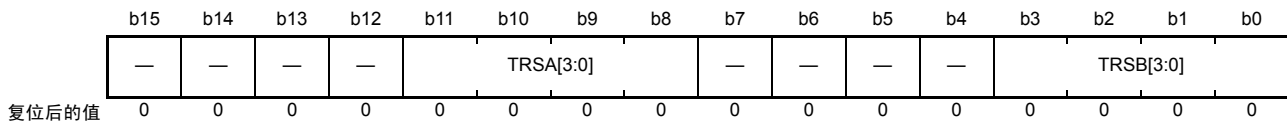
此位选择 ADDRy、ADRD、ADTSDR、ADOCDR、ADDBLDR 寄存器中保存的数据是往右靠紧还是往左靠紧。

在选择了 A/D 转换值加法运算模式时，各数据寄存器的格式与 ADCER.ADRFMT 位的设定无关，固定为往左靠紧。

有关数据寄存器格式的详细内容，请参照“32.2.1 A/D 数据寄存器 y (ADDRy) (y=0 ~ 15)”、“32.2.2 A/D 数据双重化寄存器 (ADDBLDR)”、“32.2.3 A/D 温度传感器的数据寄存器 (ADTSDR)”、“32.2.4 A/D 内部基准电压的数据寄存器 (ADOCDR)”和“32.2.5 A/D 自诊断数据寄存器 (ADRD)”。

32.2.12 A/D 开始触发选择寄存器 (ADSTRGR)

地址 0008 9010h



位	符号	位名	功能	R/W
b3-b0	TRSB[3:0]	组 B 专用 A/D 转换开始触发选择位	在分组扫描模式中，选择组 B 的 A/D 转换开始触发。	R/W
b7-b4	—	保留位	读写值都为“0”。	R/W
b11-b8	TRSA[3:0]	A/D 转换开始触发选择位	在单次扫描模式和连续扫描模式中，选择 A/D 转换开始触发。在分组扫描模式中，选择组 A 的 A/D 转换开始触发。	R/W
b15-b12	—	保留位	读写值都为“0”。	R/W

TRSB[3:0] 位 (组 B 专用 A/D 转换开始触发选择位)

此位选择组 B 所选的模拟输入的扫描的开始触发。TRSB[3:0] 位只在分组扫描模式中需要设定，在其他模式中不使用。组 B 的扫描开始触发禁止设定为软件触发和异步触发。因此，在分组扫描模式中，必须将 TRSB[3:0] 位设定为“0000”以外的值，必须将 ADCSR.TRGE 位设定为“1”。

通过 TRSB[3:0] 位选择的 A/D 启动源一览表如表 32.5 所示。

TRSA[3:0] 位 (A/D 转换开始触发选择位)

这些位选择单次扫描模式和连续扫描模式中的 A/D 转换开始触发，还能选择分组扫描模式中组 A 所选的模拟输入的扫描的开始触发。在分组扫描模式或者双触发模式中进行扫描时，不能使用软件触发和异步触发。

- 在使用同步触发 (MTU、ELC、温度传感器) 的 A/D 转换启动源时，必须将 ADCSR.TRGE 位置“1”并且将 ADCSR.EXTRG 位置“0”。
- 在使用异步触发 (ADTRG0#) 时，必须将 ADCSR.TRGE 位置“1”并且将 ADCSR.EXTRG 位置“1”。
- 软件触发 (ADCSR.ADST) 与 ADCSR.TRGE 位、ADCSR.EXTRG 位和 TRSA[3:0] 位的设定值无关，总是有效。

通过 TRSA[3:0] 位选择的 A/D 启动源一览表如表 32.6 所示。

表 32.5 通过 TRSB[3:0] 位选择的 A/D 启动源一览表

模块	启动源	备注	TRSB[3]	TRSB[2]	TRSB[1]	TRSB[0]
MTU	TRG0AN	MTU0 的 TRGA 的输入捕捉 / 比较匹配	0	0	0	1
	TRG0BN	MTU0 的 TRGB 的输入捕捉 / 比较匹配 B	0	0	1	0
	TRGAN	MTU0 ~ MTU4 的 TRGA 的输入捕捉 / 比较匹配或者互补 PWM 模式中的 MTU4.TCNT 的下溢 (波谷)	0	0	1	1
	TRG0EN	MTU0 的 TRGE 的比较匹配	0	1	0	0
	TRG0FN	MTU0 的 TRGF 的比较匹配	0	1	0	1
	TRG4AN	MTU4.TADCORA 和 MTU4.TCNT 的比较匹配 (中断减少功能 1)	0	1	1	0
	TRG4BN	MTU4.TADCORB 和 MTU4.TCNT 的比较匹配 (中断减少功能 1)	0	1	1	1
	TRG4ABN	MTU4.TADCORA 和 MTU4.TCNT、MTU4.TADCORB 和 MTU4.TCNT 的比较匹配 (中断减少功能 1)	1	0	0	0
ELC	ELC	ELC 的触发	1	0	0	1

表 32.6 通过 TRSA[3:0] 位选择的 A/D 启动源一览表

模块	启动源	备注	TRSA[3]	TRSA[2]	TRSA[1]	TRSA[0]
ADC	ADST	软件触发	—	—	—	—
外部输入	ADTRG0#	A/D 转换启动触发引脚	0	0	0	0
MTU	TRG0AN	MTU0 的 TRGA 的输入捕捉 / 比较匹配	0	0	0	1
	TRG0BN	MTU0 的 TRGB 的输入捕捉 / 比较匹配 B	0	0	1	0
	TRGAN	MTU0 ~ MTU4 的 TRGA 的输入捕捉 / 比较匹配或者互补 PWM 模式时的 MTU4.TCNT 的下溢 (波谷)	0	0	1	1
	TRG0EN	MTU0 的 TRGE 的比较匹配	0	1	0	0
	TRG0FN	MTU0 的 TRGF 的比较匹配	0	1	0	1
	TRG4AN	MTU4.TADCORA 和 MTU4.TCNT 的比较匹配 (中断减少功能 1)	0	1	1	0
	TRG4BN	MTU4.TADCORB 和 MTU4.TCNT 的比较匹配 (中断减少功能 1)	0	1	1	1
	TRG4ABN	MTU4.TADCORA 和 MTU4.TCNT、MTU4.TADCORB 和 MTU4.TCNT 的比较匹配 (中断减少功能 1)	1	0	0	0
ELC	ELC	ELC 的触发	1	0	0	1
温度传感器	温度传感器	温度传感器的触发	1	0	1	0

32.2.13 A/D 转换扩展输入控制寄存器 (ADEXICR)

地址 0008 9012h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	OCS	TSS	—	—	—	—	—	—	OCSAD	—
复位后的值	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	—	保留位	读写值都为“0”。	R/W
b1	OCSAD	内部基准电压 A/D 转换值加法运算模式选择位	0: 不选择内部基准电压 A/D 转换值加法运算模式 1: 选择内部基准电压 A/D 转换值加法运算模式	R/W
b7-b2	—	保留位	读写值都为“0”。	R/W
b8	TSS	温度传感器输出 A/D 转换选择位	0: 不对温度传感器输出进行 A/D 转换 1: 对温度传感器输出进行 A/D 转换	R/W
b9	OCS	内部基准电压 A/D 转换选择位	0: 不对内部基准电压进行 A/D 转换 1: 对内部基准电压进行 A/D 转换	R/W
b15-b10	—	保留位	读写值都为“0”。	R/W

OCSAD 位 (内部基准电压 A/D 转换值加法运算模式选择位)

如果选择内部基准电压的 A/D 转换, 并且将 OCSAD 位置“1”, 就对内部基准电压连续进行 ADADC.ADC[1:0] 位所设次数 (2 ~ 4 次) 的 A/D 转换, 然后将累加运算值返回到 A/D 内部基准电压数据寄存器 (ADOCADR)。必须在 ADCSR.ADST 位为“0”时设定 OCSAD 位。

TSS 位 (温度传感器输出 A/D 转换选择位)

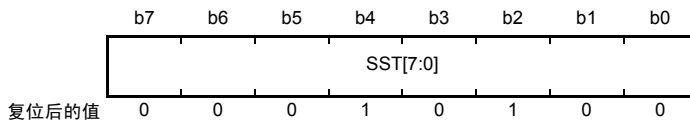
此位选择温度传感器输出的 A/D 转换。在进行温度传感器输出的 A/D 转换时, 必须选择单次扫描模式, 并且将 ADANSA 寄存器的全部位以及 OCS 位置“0”。必须在 ADCSR.ADST 位为“0”时设定 TSS 位。

OCS 位 (内部基准电压 A/D 转换选择位)

此位选择内部基准电压的 A/D 转换。在进行内部基准电压的 A/D 转换时, 必须选择单次扫描模式, 并且将 ADANSA 寄存器的全部位以及 TSS 位置“0”。必须在 ADST 位为“0”时设定 OCS 位。

32.2.14 A/D 采样状态寄存器 n (ADSSTRn) (n=0 ~ 7、L、T、O)

地址 ADSSTR0: 0008 9060h、ADSSTR1: 0008 9073h、ADSSTR2: 0008 9074h、ADSSTR3: 0008 9065h、ADSSTR4: 0008 9076h、
ADSSTR5: 0008 9077h、ADSSTR6: 0008 9078h、ADSSTR7: 0008 9079h、ADSSTRL: 0008 9061h、ADSSTRT: 0008 9070h、
ADSSTRO: 0008 9071h



位	符号	位名	功能	R/W
b7-b0	SST[7:0]	采样时间设定位	以 12 ~ 255 个状态为单位，设定采样时间。	R/W

ADSSTRn 寄存器是设定模拟输入的采样时间的定时器。

如果 1 个状态为 1 个 ADCLK (A/D 转换时钟) 的宽度并且 ADCLK 时钟为 50MHz，1 个状态就为 20ns，初始值为 20 个状态。能在模拟输入信号源阻抗高而采样时间不够或者 ADCLK 时钟为低速的情况下调整采样时间。必须在 ADCSR.ADST 位为“0”时设定 SST[7:0] 位。

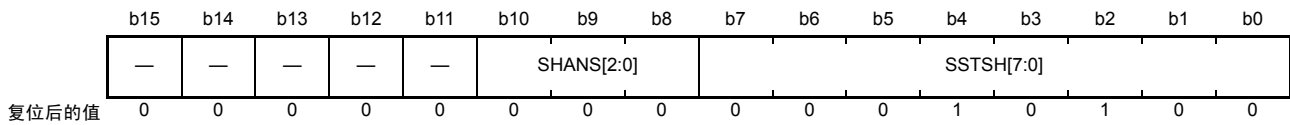
必须将采样时间的设定值设定为大于等于 12 个状态并且小于等于 255 个状态的值。另外，必须将采样时间设定为大于等于 0.4μs。A/D 采样状态寄存器和对象通道的关系如表 32.7 所示。

表 32.7 A/D 采样状态寄存器和对象通道的关系

位名	对象通道
ADSSTR0.SST[7:0] 位	AN000/ 自诊断
ADSSTR1.SST[7:0] 位	AN001
ADSSTR2.SST[7:0] 位	AN002
ADSSTR3.SST[7:0] 位	AN003
ADSSTR4.SST[7:0] 位	AN004
ADSSTR5.SST[7:0] 位	AN005
ADSSTR6.SST[7:0] 位	AN006
ADSSTR7.SST[7:0] 位	AN007
ADSSTRL.SST[7:0] 位	AN008 ~ AN015
ADSSTRT.SST[7:0] 位	温度传感器输出
ADSSTRO.SST[7:0] 位	内部基准电压

32.2.15 采样 & 保持电路控制寄存器 (ADSHCR)

地址 0008 9066h



位	符号	位名	功能	R/W
b7-b0	SSTSH[7:0]	采样时间采样 & 保持电路设定位	在 4 ~ 255 个状态间设定采样时间。	R/W
b10-b8	SHANS[2:0]	通道专用采样 & 保持电路的旁路选择位	选择使用 AN000 ~ AN002 的通道专用采样 & 保持电路或者不使用 AN000 ~ AN002 的通道专用采样 & 保持电路 (旁路) 0: 不使用通道专用采样 & 保持电路 (旁路) 1: 使用通道专用采样 & 保持电路	R/W
b15-b11	—	保留位	读写值都为“0”。	R/W

ADSHCR 寄存器是设定通道专用采样 & 保持电路的寄存器。

SSTSH[7:0] 位 (采样时间采样 & 保持电路设定位)

这些位设定通道专用采样 & 保持电路的采样时间。如果 1 个状态为 1 个 ADCLK (A/D 转换时钟) 的宽度并且 ADCLK 时钟为 50MHz, 1 个状态就为 20ns, 初始值为 20 个状态。能在模拟输入信号源阻抗高而采样时间不够或者 ADCLK 时钟为低速的情况下调整采样时间。必须在 ADCSR.ADST 位为“0”时设定 SSTSH[7:0] 位。必须将采样时间的设定值设定为大于等于 4 个状态并且小于等于 255 个状态的值。另外, 必须将采样时间设定为大于等于 0.4μs。例如, ADCLK 时钟为 25MHz 时, 采样状态设定值的下限为 10 个状态。

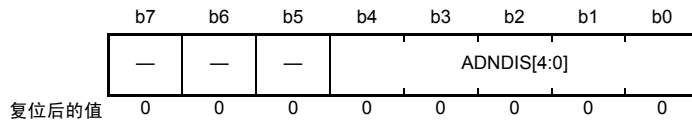
SHANS[2:0] 位 (通道专用采样 & 保持电路的旁路选择位)

这些位选择使用模拟输入 AN000 ~ AN002 的通道专用采样 & 保持电路或者不使用模拟输入 AN000 ~ AN002 的通道专用采样 & 保持电路 (旁路)。SHANS[0] 位为 AN000 的通道专用采样 & 保持电路的选择位, SHANS[1] 位为 AN001 的通道专用采样 & 保持电路的选择位, SHANS[2] 位为 AN002 的通道专用采样 & 保持电路的选择位。必须在 ADCSR.ADST 位为“0”时设定 SHANS[2:0] 位。

当 AVCC0 小于 2.7V 时, 通道专用采样 & 保持电路不运行。必须将 SHANS[2:0] 位置“000b”。

32.2.16 AD 断线检测控制寄存器 (ADDISCR)

地址 0008 907Ah



位	符号	位名	功能	R/W
b4-b0	ADNDIS[4:0]	断线检测辅助设定位	设定断线检测辅助功能。	R/W
b7-b5	—	保留位	读写值都为“0”。	R/W

ADDISCR 寄存器是设定断线检测辅助功能的寄存器。

ADNDIS[4:0] 位 (A/D 断线检测辅助设定位)

这些位设定 A/D 断线检测辅助功能的预充电或者放电，以及设定预充电 / 放电的期间。在 ADNDIS[4] 位为“1”时选择预充电，在 ADNDIS[4] 位为“0”时选择放电。通过 ADNDIS[3:0] 位设定预充电 / 放电的期间。当 ADNDIS[3:0] 位为“0000b”时，断线检测辅助功能无效。当 ADNDIS[3:0] 位不为“0000b”时，设定的值就为预充电 / 放电期间的状态数。

对温度传感器输出或者内部基准电压进行转换时，不能使用断线检测辅助功能。必须将 ADNDIS[4:0] 位设定为“00000b”。

32.3 运行说明

32.3.1 扫描的运行说明

扫描是指依次对选择的通道的模拟输入进行 A/D 转换。

扫描转换的运行模式有 3 种，分别是单次扫描模式、连续扫描模式和分组扫描模式。

单次扫描模式是指在对选择的至少 1 个通道进行 1 次扫描后结束的模式。连续扫描模式是指对选择的至少 1 个通道无限制地重复进行扫描，直到通过软件将 ADCSR.ADST 位清“0”（从“1”变为“0”）为止。分组扫描模式是指通过分别选择的触发开始组 A 和组 B 的扫描，并且在分别对组 A 和组 B 所选通道进行 1 次扫描后结束。

如果在单次扫描模式或者连续扫描模式中开始扫描转换，就从 ADANSA 寄存器选择的 ANn 中按照 n 从小到大的顺序开始 A/D 转换。在分组扫描模式中，组 A 从 ADANSA 寄存器选择的 ANn 中按照 n 从小到大的顺序开始 A/D 转换，组 B 从 ADANSB 寄存器选择的 ANn 中按照 n 从小到大的顺序开始 A/D 转换。

当选择自诊断时，在每次扫描的最开始执行 1 次自诊断，对 12 位 A/D 转换器内部生成的 3 个电压值中的 1 个电压值进行 A/D 转换。

当选择温度传感器输出或者内部基准电压时，在单次扫描模式中进行 A/D 转换。

双触发模式用于单次扫描模式或者分组扫描模式。如果允许双触发模式，只在通过 ADSTRGR.TRSA[3:0] 位选择的 MTU 触发或者 ELC 触发启动扫描时，对 ADCSR.DBLANS[4:0] 位选择的 1 个通道的 A/D 转换数据进行双重化。

如果通过 ADSHCR.SHANS[2:0] 位设定 AN000 ~ AN002 中的任意 1 个使用通道专用采样 & 保持电路，就在每次扫描的第 1 个 A/D 转换开始前执行对象模拟输入的采样 & 保持。

32.3.2 单次扫描模式

32.3.2.1 基本运行（无通道专用采样 & 保持电路）

单次扫描模式的基本运行对指定通道的模拟输入只进行 1 个周期的 A/D 转换，如下所示。在对所选通道进行扫描的情况下，温度传感器输出 A/D 转换选择位 (ADEXICR.TSS) 和内部基准电压 A/D 转换选择位 (ADEXICR.OCs) 都被置“0”（不选择）。

1. 如果通过软件触发、同步触发 (MTU、ELC) 或者异步触发的输入将 ADCSR.ADST 位置“1”（开始 A/D 转换），就从 ADANSA 寄存器所选 ANn 中按照 n 从小到大的顺序开始 A/D 转换。
2. 如果 1 个通道的 A/D 转换结束，就将 A/D 转换结果保存到对应的 A/D 数据寄存器 (ADDRy)。
3. 在所选通道的 A/D 转换全部结束后，如果 ADCSR.ADIE 位已被置“1”（允许由扫描结束引起的 S12ADI0 中断），就产生 S12ADI0 中断请求。
4. ADCSR.ADST 位在 A/D 转换过程中保持“1”（开始 A/D 转换），一旦所选通道的 A/D 转换全部结束就自动清除，并且 12 位 A/D 转换器进入待机状态。

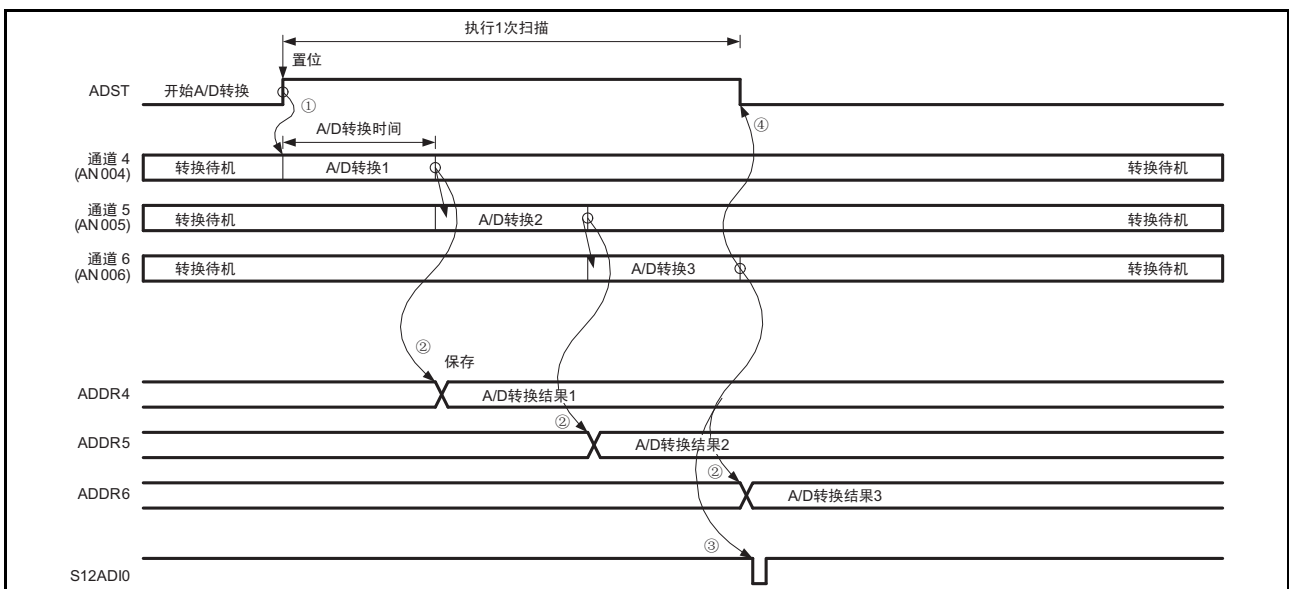


图 32.3 单次扫描模式的运行例子（基本运行：选择 AN004 ~ AN006）

32.3.2.2 基本运行 (有通道专用采样 & 扫描电路)

如果使用通道专用采样 & 扫描电路,就在执行采样 & 保持后对所选的全部通道的模拟输入只进行 1 次 A/D 转换,如下所示。通过 ADShCR.SHANS[2:0] 位选择使用通道专用采样 & 保持电路的通道。

在对所选通道进行扫描的情况下,温度传感器输出 A/D 转换选择位 (ADEXICR.TSS) 和内部基准电压 A/D 转换选择位 (ADEXICR.OCS) 都被置“0”(不选择)。

1. 如果通过软件触发、同步触发 (MTU、ELC) 或者异步触发的输入将 ADCSR.ADST 位置“1”(开始 A/D 转换), 就开始对使用通道专用采样 & 保持电路的全部通道的模拟输入进行采样。
2. 在执行采样 & 保持后, 从 ADANSA 寄存器选择的通道 ANn 中按照 n 从小到大的顺序开始 A/D 转换。
3. 如果通道的 A/D 转换结束, 就将 A/D 转换结果保存到对应的 A/D 数据寄存器 (ADDRy)。
4. 在所选通道的 A/D 转换全部结束后, 如果 ADCSR.ADIE 位已被置“1”(允许由扫描结束引起的 S12ADI0 中断), 就产生 S12ADI0 中断请求。
5. ADCSR.ADST 位在 A/D 转换过程中保持“1”(开始 A/D 转换), 一旦所选通道的 A/D 转换全部结束就自动清除, 并且 12 位 A/D 转换器进入待机状态。

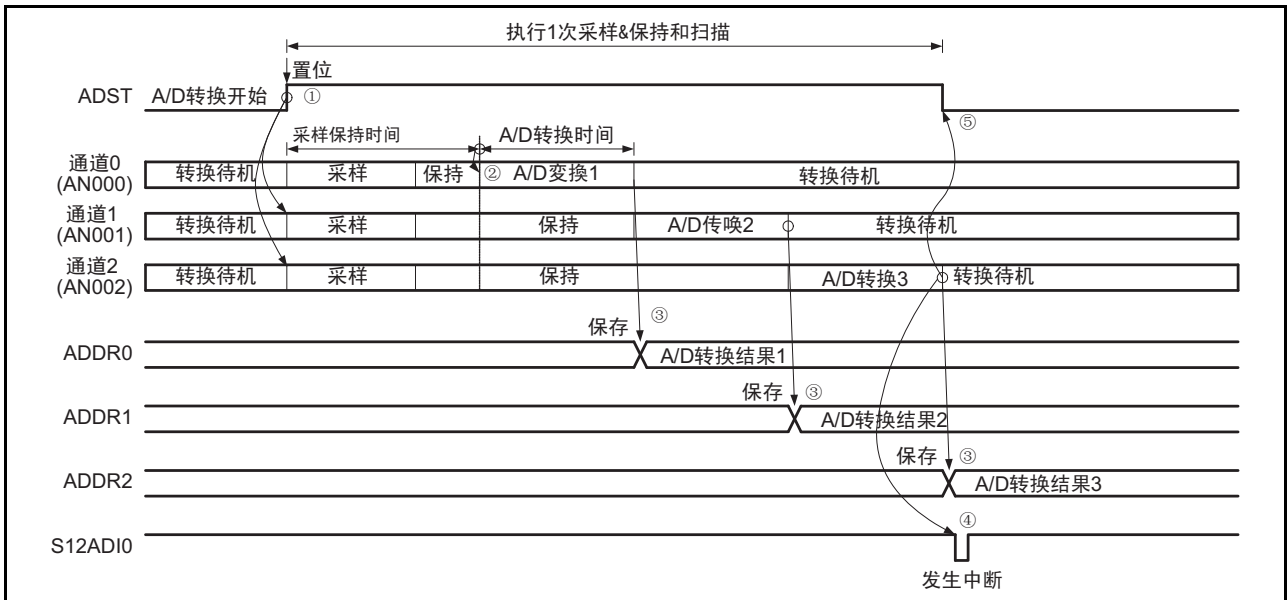


图 32.4 单次扫描模式的运行例子 (有通道专用采样 & 保持电路)

32.3.2.3 通道选择和自诊断 (无通道专用采样 & 保持电路)

如果在选择通道的同时选择自诊断, 就对提供给 12 位 A/D 转换器的基准电压 VREFH0 (×0、×1/2、×1 中的任意 1 个) 进行 A/D 转换, 之后对所选通道的模拟输入只进行 1 次 A/D 转换, 如下所示。

在对所选通道进行扫描的情况下, 温度传感器输出 A/D 转换选择位 (ADEXICR.TSS) 和内部基准电压 A/D 转换选择位 (ADEXICR.OCS) 都被置“0” (不选择)。

1. 如果通过软件触发、同步触发 (MTU、ELC) 或者异步触发的输入将 ADCSR.ADST 位置“1” (开始 A/D 转换), 就最先开始自诊断的 A/D 转换。
2. 在自诊断的 A/D 转换结束后, 将 A/D 转换结果保存到 A/D 自诊断数据寄存器 (ADRD), 然后从 ADANSA 寄存器选择的通道 ANn 中按照 n 从小到大的顺序开始 A/D 转换。
3. 如果通道的 A/D 转换结束, 就将 A/D 转换结果保存到对应的 A/D 数据寄存器 (ADDRy)。
4. 在所选通道的 A/D 转换全部结束后, 如果 ADCSR.ADIE 位已被置“1” (允许由扫描结束引起的 S12ADI0 中断), 就产生 S12ADI0 中断请求。
5. ADCSR.ADST 位在 A/D 转换过程中保持“1” (开始 A/D 转换), 一旦所选通道的 A/D 转换全部结束就自动清除, 并且 12 位 A/D 转换器进入待机状态。

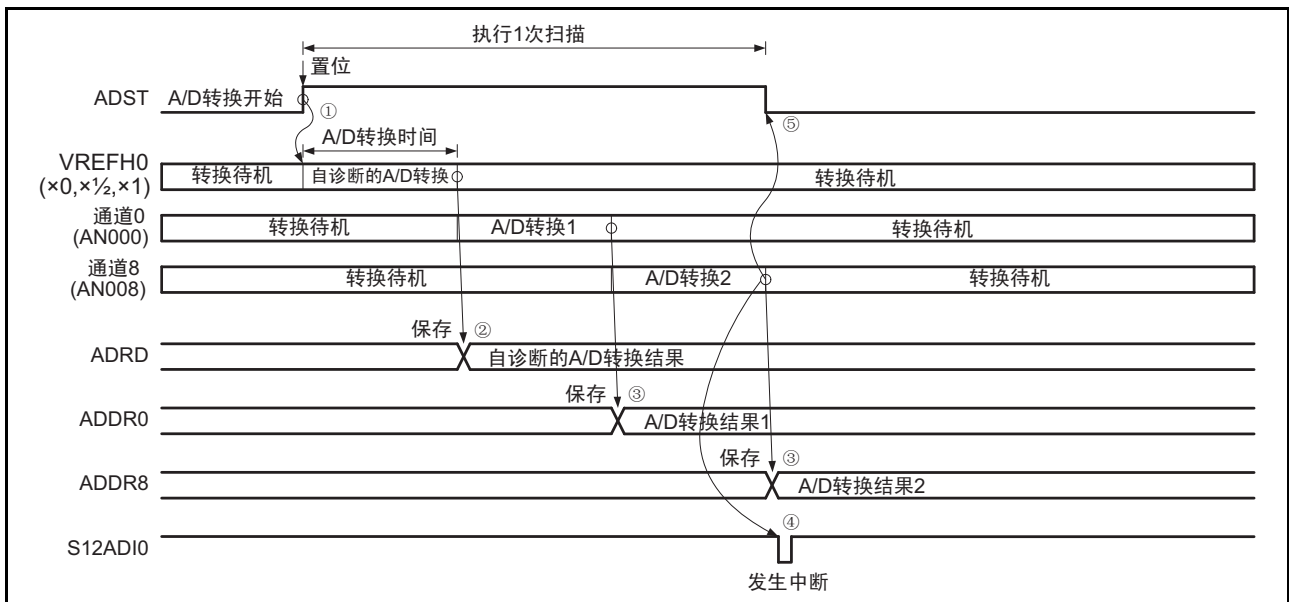


图 32.5 单次扫描模式的运行例子 (基本运行和自诊断)

32.3.2.4 通道选择和自诊断 (有通道专用采样 & 保持电路)

如果在选择通道的同时选择自诊断，并且使用通道专用采样 & 保持电路，就在执行采样 & 保持后对提供给 12 位 A/D 转换器的基准电压 VREFH0 (x0、x1/2、x1 中的任意 1 个) 进行 A/D 转换，之后对所选通道的模拟输入只进行 1 次 A/D 转换，如下所示。

在对所选通道进行扫描的情况下，温度传感器输出 A/D 转换选择位 (ADEXICR.TSS) 和内部基准电压 A/D 转换选择位 (ADEXICR.OCS) 都被置“0” (不选择)。

1. 如果通过软件触发、同步触发 (MTU、ELC) 或者异步触发的输入将 ADCSR.ADST 位置“1” (开始 A/D 转换)，就开始对使用通道专用采样 & 保持电路的全部通道的模拟输入进行采样。
2. 在执行采样 & 保持后，开始自诊断的 A/D 转换。
3. 在自诊断的 A/D 转换结束后，将 A/D 转换结果保存到 A/D 自诊断数据寄存器 (ADRD)，然后从 ADANSA 寄存器选择的通道 ANn 中按照 n 从小到大的顺序开始 A/D 转换。
4. 如果通道的 A/D 转换结束，就将 A/D 转换结果保存到对应的 A/D 数据寄存器 (ADDRy)。
5. 在所选通道的 A/D 转换全部结束后，如果 ADCSR.ADIE 位已被置“1” (允许由扫描结束引起的 S12ADI0 中断)，就产生 S12ADI0 中断请求。
6. ADCSR.ADST 位在 A/D 转换过程中保持“1” (开始 A/D 转换)，一旦所选通道的 A/D 转换全部结束就自动清除，并且 12 位 A/D 转换器进入待机状态。



图 32.6 单次扫描模式的运行例子 (有通道专用采样 & 保持电路和自诊断)

32.3.2.5 选择温度传感器输出时的 A/D 转换运行

在单次扫描模式中进行温度传感器输出的 A/D 转换，如下所示。

将所选通道全部设定为不选择 (ADANSA.ANSA[15:0] 位为 “0000h” 并且 ADCSR.DBLE 位为 “0”)，并且自诊断也设定为不选择。另外，必须将内部基准电压 A/D 转换选择位 (ADEXICR.OCS) 置 “0” (不选择)。

1. 如果通过温度传感器触发的输入将 ADCSR.ADST 位置 “1” (开始 A/D 转换)，就开始温度传感器输出的 A/D 转换。
2. 如果 A/D 转换结束，就将 A/D 转换结果保存到对应的 A/D 温度传感器的数据寄存器 (ADTSDR)。如果 ADCSR.ADIE 位已被置 “1” (允许由扫描结束引起的 S12ADI0 中断)，就产生 S12ADI0 中断请求。
3. ADCSR.ADST 位在 A/D 转换过程中保持 “1” (开始 A/D 转换)，一旦 A/D 转换结束就自动清除，并且 12 位 A/D 转换器进入待机状态。

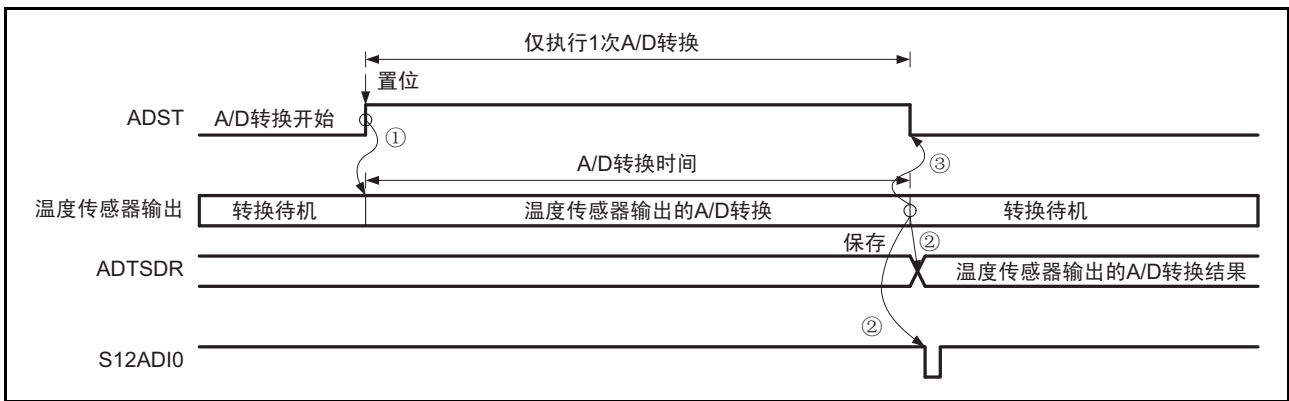


图 32.7 单次扫描模式的运行例子 (选择温度传感器输出)

32.3.2.6 选择内部基准电压时的 A/D 转换运行

在单次扫描模式中进行内部基准电压的 A/D 转换，如下所示。

将所选通道全部设定为不选择 (ADANSA.ANSA[15:0] 位为 “0000h” 并且 ADCSR.DBLE 位为 “0”)，并且自诊断也设定为不选择。另外，必须将温度传感器输出 A/D 转换选择位 (ADEXICR.TSS) 置 “0” (不选择)。

1. 如果通过软件触发、同步触发 (MTU、ELC) 或者异步触发的输入将 ADCSR.ADST 位置 “1” (开始 A/D 转换)，就开始内部基准电压的 A/D 转换。
2. 如果 A/D 转换结束，就将 A/D 转换结果保存到对应的 A/D 内部基准电压的数据寄存器 (ADOCADR)。如果 ADCSR.ADIE 位已被置 “1” (允许由扫描结束引起的 S12ADI0 中断)，就产生 S12ADI0 中断请求。
3. ADCSR.ADST 位在 A/D 转换过程中保持 “1” (开始 A/D 转换)，一旦 A/D 转换结束就自动清除，并且 12 位 A/D 转换器进入待机状态。



图 32.8 单次扫描模式的运行例子 (选择内部基准电压)

32.3.2.7 选择双触发模式时的运行

如果在单次扫描模式中选择双触发模式，就通过 MTU 触发或者 ELC 触发执行 2 次单次扫描模式，如下所示。

将自诊断设定为不选择，并且将温度传感器输出 A/D 转换选择位 (ADEXICR.TSS) 和内部基准电压 A/D 转换选择位 (ADEXICR.OCS) 都置“0” (不选择)。

如果将双重化的通道号设定到 ADCSR.DBLANS[4:0] 位，并且将 ADCSR.DBLE 位置“1”，A/D 转换数据的双重化就有效。将 ADCSR.DBLE 位置“1”时，ADANSA 寄存器的通道选择就无效。另外，在选择双触发模式时，必须通过 ADSTRGR.TRSA[3:0] 位选择 MTU 触发或者 ELC 触发，并且将 ADCSR.EXTRG 位置“0”、ADCSR.TRGE 位置“1”。而且不能使用软件触发。

1. 如果通过 MTU 触发或者 ELC 触发的输入将 ADCSR.ADST 位置“1” (开始 A/D 转换)，就开始对 ADCSR.DBLANS[4:0] 位选择的 1 个通道进行 A/D 转换。
2. 如果 A/D 转换结束，就将 A/D 转换结果保存到对应的 A/D 数据寄存器 (ADDRy)。
3. 自动清除 ADST 位，并且 12 位 A/D 转换器进入待机状态。此时，与 ADCSR.ADIE 位 (允许由扫描结束引起的 S12ADI0 中断) 的设定无关，不发生 S12ADI0 中断。
4. 如果通过第 2 次触发输入将 ADCSR.ADST 位置“1” (开始 A/D 转换)，就开始对 ADCSR.DBLANS[4:0] 位选择的 1 个通道进行 A/D 转换。
5. 如果 A/D 转换结束，就将 A/D 转换结果保存到双触发模式专用的 A/D 数据双重化寄存器 (ADDBLDR)。
6. 如果 ADCSR.ADIE 位已被置“1” (允许由扫描结束引起的 S12ADI0 中断)，就产生 S12ADI0 中断请求。
7. ADCSR.ADST 位在 A/D 转换过程中保持“1” (开始 A/D 转换)，一旦 A/D 转换结束就自动清除，并且 12 位 A/D 转换器进入待机状态。

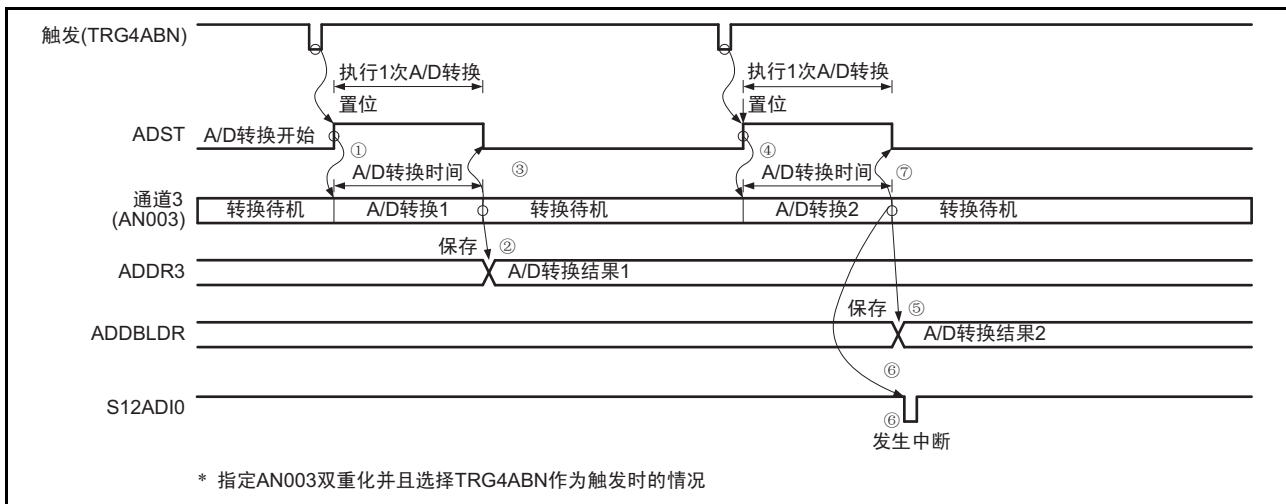


图 32.9 单次扫描模式的运行例子 (选择双触发模式、AN003 的双重化)

32.3.3 连续扫描模式

32.3.3.1 基本运行 (无通道专用采样 & 保持电路)

连续扫描模式的基本运行对所选通道的模拟输入重复进行 A/D 转换，如下所示。

在连续扫描模式中，温度传感器输出 A/D 转换选择位 (ADEXICR.TSS) 和内部基准电压 A/D 转换选择位 (ADEXICR.OCs) 都被置“0” (不选择)。

1. 如果通过软件触发、同步触发 (MTU、ELC) 或者异步触发的输入将 ADCSR.ADST 位置“1” (开始 A/D 转换)，就从 ADANSA 寄存器选择的 ANn 中按照 n 从小到大的顺序开始 A/D 转换。
2. 如果 1 个通道的 A/D 转换结束，就将 A/D 转换结果保存到对应的 A/D 数据寄存器 (ADDRy)。
3. 在所选通道的 A/D 转换全部结束后，如果 ADCSR.ADIE 位已被置“1” (允许由扫描结束引起的 S12ADI0 中断)，就产生 S12ADI0 中断请求。另外，12 位 A/D 转换器继续从 ADANSA 寄存器选择的 ANn 中按照 n 从小到大的顺序开始 A/D 转换。
4. 不自动清除 ADCSR.ADST 位，而在此位为“1” (开始 A/D 转换) 的期间重复步骤 2~3。如果将 ADCSR.ADST 位置“0” (停止 A/D 转换)，就中止 A/D 转换并且 12 位 A/D 转换器进入待机状态。
5. 此后，如果将 ADCSR.ADST 位置“1” (开始 A/D 转换)，就重新开始从 ADANSA 寄存器选择的 ANn 中按照 n 从小到大的顺序进行 A/D 转换。

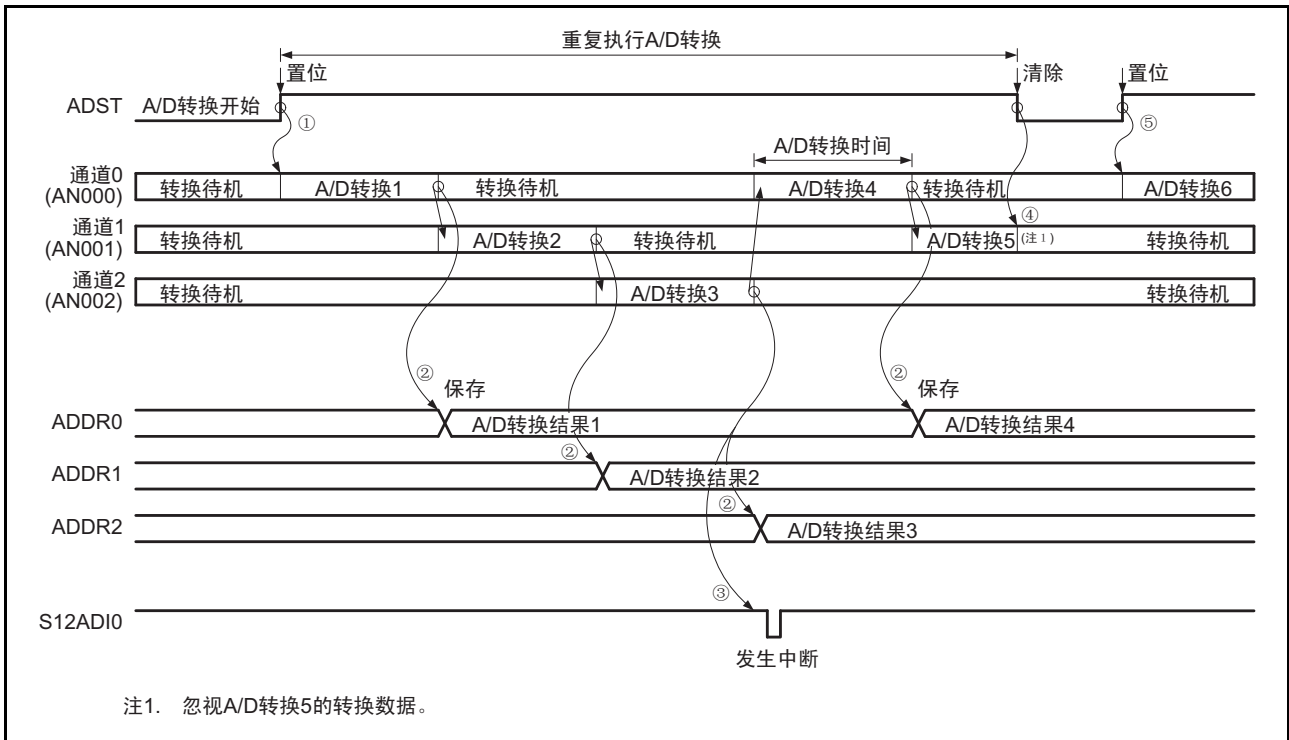


图 32.10 连续扫描模式的运行例子 (基本运行: 选择 AN000 ~ AN002)

32.3.3.2 基本运行 (有通道专用采样 & 保持电路)

如果使用通道专用采样 & 保持电路,就在执行采样 & 保持后,对选择的全部通道的模拟输入进行连续的 A/D 转换,如下所示。通过 ADSHCR.SHANS[2:0] 位选择使用通道专用采样 & 保持电路的通道。

在连续扫描模式中,温度传感器输出 A/D 转换选择位 (ADEXICR.TSS) 和内部基准电压 A/D 转换选择位 (ADEXICR.OCS) 都被置“0”(不选择)。

1. 如果通过软件触发或者同步触发 (MTU、ELC) 的输入将 ADCSR.ADST 位置“1”(开始 A/D 转换),就开始对使用通道专用采样 & 保持电路的全部通道的模拟输入进行采样。
2. 在执行采样 & 保持后,从 ADANSA 寄存器选择的 ANn 中按照 n 从小到大的顺序开始 A/D 转换。
3. 如果通道的 A/D 转换结束,就将 A/D 转换结果保存到对应的 A/D 数据寄存器 (ADDRy)。
4. 在所选通道的 A/D 转换全部结束后,如果 ADCSR.ADIE 位已被置“1”(允许由扫描结束引起的 S12ADI0 中断),就产生 S12ADI0 中断请求。另外,开始对使用通道专用采样 & 保持电路的通道的模拟输入全部进行采样。
5. 不自动清除 ADCSR.ADST 位,而在此位为“1”(开始 A/D 转换)的期间重复步骤 2~4。如果将 ADCSR.ADST 位置“0”(停止 A/D 转换),就中止 A/D 转换并且 12 位 A/D 转换器进入待机状态。
6. 此后,如果将 ADCSR.ADST 位置“1”(开始 A/D 转换),就重新开始对使用通道专用采样 & 保持电路的全部通道的模拟输入进行采样。

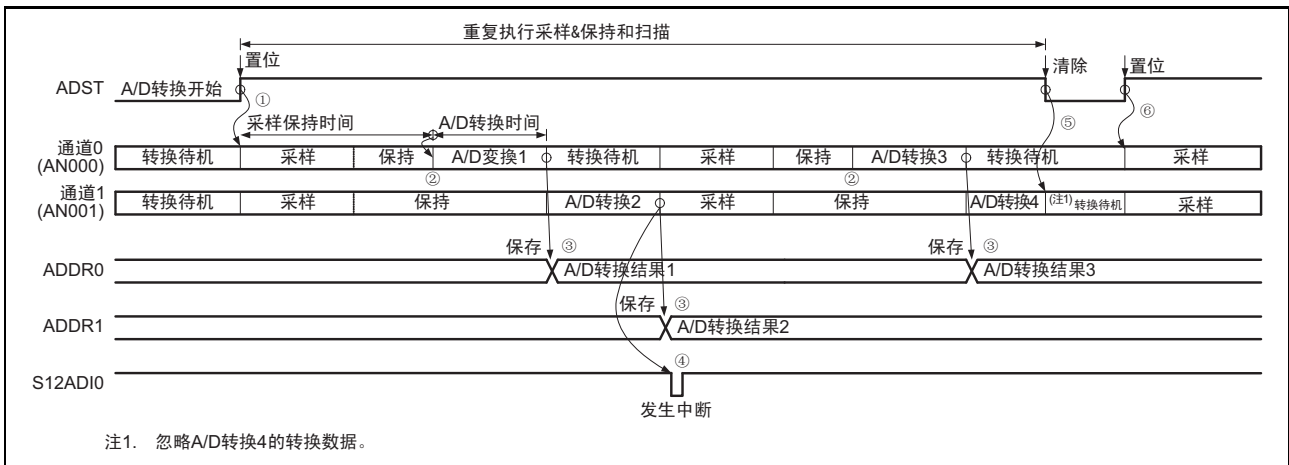


图 32.11 连续扫描模式的运行例子 (有通道专用采样 & 保持电路)

32.3.3.3 通道选择和自诊断 (无通道专用采样 & 保持电路)

如果在选择通道的同时选择自诊断, 就对提供给 12 位 A/D 转换器的基准电压 VREFH0 (×0、×1/2、×1 中的任意 1 个) 进行 A/D 转换, 之后对所选通道的模拟输入重复进行 A/D 转换, 如下所示。

在连续扫描模式中, 温度传感器输出 A/D 转换选择位 (ADEXICR.TSS) 和内部基准电压 A/D 转换选择位 (ADEXICR.OCS) 都被置“0” (不选择)。

1. 如果通过软件触发、同步触发 (MTU、ELC) 或者异步触发的输入将 ADCSR.ADST 位置“1” (开始 A/D 转换), 就最先开始自诊断的 A/D 转换。
2. 如果自诊断的 A/D 转换结束, 就将 A/D 转换结果保存到 A/D 自诊断数据寄存器 (ADRD), 然后从 ADANSA 寄存器选择的 ANn 中按照 n 从小到大的顺序开始 A/D 转换。
3. 如果通道的 A/D 转换结束, 就将 A/D 转换结果保存到对应的 A/D 数据寄存器 (ADDRy)。
4. 在所选通道的 A/D 转换全部结束后, 如果 ADCSR.ADIE 位已被置“1” (允许由扫描结束引起的 S12ADI0 中断), 就产生 S12ADI0 中断请求。另外, 12 位 A/D 转换器继续自诊断的 A/D 转换, 并且在转换结束后从 ADANSA 寄存器选择的 ANn 中按照 n 从小到大的顺序开始 A/D 转换。
5. 不自动清除 ADST 位, 而在此位为“1” (开始 A/D 转换) 的期间重复步骤 2~4。如果将 ADST 位置“0” (停止 A/D 转换), 就中止 A/D 转换并且 12 位 A/D 转换器进入待机状态。
6. 此后, 如果将 ADST 位置“1” (开始 A/D 转换), 就重新开始自诊断的 A/D 转换。

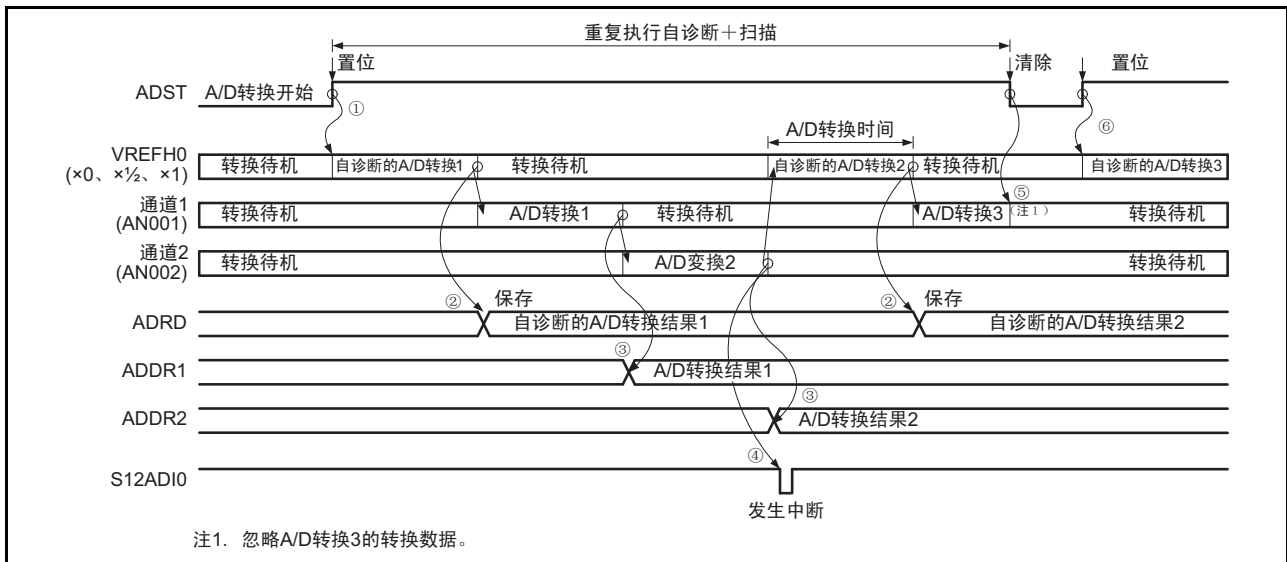


图 32.12 连续扫描模式的运行例子 (基本运行和自诊断)

32.3.3.4 通道选择和自诊断 (有通道专用采样 & 保持电路)

如果在选择通道的同时选择自诊断, 并且使用通道专用采样 & 保持电路, 就在执行采样 & 保持后, 对提供 12 位 A/D 转换器的基准电压 VREFH0 (×0、×1/2、×1 中的任意 1 个) 进行 A/D 转换, 之后对所选通道的模拟输入重复进行 A/D 转换, 如下所示。

在连续扫描模式中, 温度传感器输出 A/D 转换选择位 (ADEXICR.TSS) 和内部基准电压 A/D 转换选择位 (ADEXICR.OCs) 都被置“0” (不选择)。

1. 如果通过软件触发、同步触发 (MTU、ELC) 或者异步触发的输入将 ADCSR.ADST 位置“1” (开始 A/D 转换), 就开始对使用通道专用采样 & 保持电路的全部通道的模拟输入进行采样。
2. 在执行采样 & 保持后, 开始自诊断的 A/D 转换。
3. 如果自诊断的 A/D 转换结束, 就将 A/D 转换结果保存到 A/D 自诊断数据寄存器 (ADRD), 然后从 ADANSA 寄存器选择的 ANn 中按照 n 从小到大的顺序开始 A/D 转换。
4. 如果通道的 A/D 转换结束, 就将 A/D 转换结果保存到对应的 A/D 数据寄存器 (ADDRy)。
5. 在所选通道的 A/D 转换全部结束后, 如果 ADCSR.ADIE 位已被置“1” (允许由扫描结束引起的 S12ADI0 中断), 就产生 S12ADI0 中断请求。另外, 开始对使用通道专用采样 & 保持电路的全部通道的模拟输入进行采样。
6. 不自动清除 ADST 位, 而在此位为“1” (开始 A/D 转换) 的期间重复步骤 2~5。如果将 ADST 位置“0” (停止 A/D 转换), 就中止 A/D 转换并且 12 位 A/D 转换器进入待机状态。
7. 此后, 如果将 ADST 位置“1” (开始 A/D 转换), 就重新开始对使用通道专用采样 & 保持电路的全部通道的模拟输入进行采样。

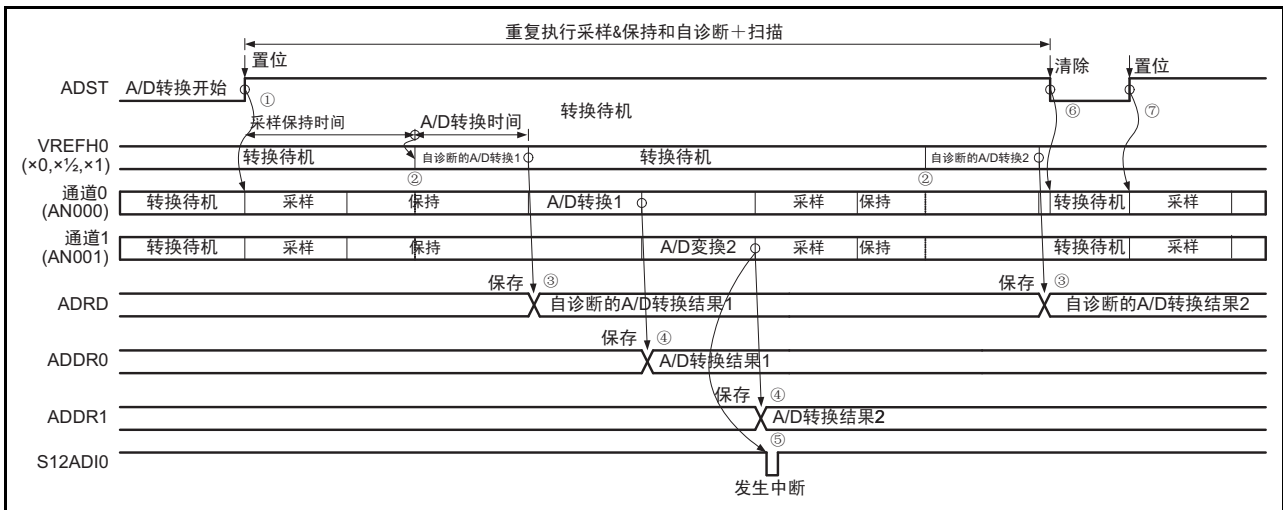


图 32.13 连续扫描模式的运行例子 (通道专用采样 & 保持电路运行和自诊断)

32.3.4 分组扫描模式

32.3.4.1 基本运行

分组扫描模式的基本运行以 MTU 触发或者 ELC 触发作为扫描开始条件，并且对组 A 和组 B 分别选择的全部通道的模拟输入只进行 1 次 A/D 转换，如下所示。组 A 和组 B 的扫描运行与单次扫描模式中的运行相同。

对于分组扫描模式中的触发设定，通过 ADSTRGR.TRSA[3:0] 位选择组 A 的触发，通过 ADSTRGR.TRSB[3:0] 位选择组 B 的触发。必须给组 A 和组 B 选择不同的触发，以避免同时发生组 A 和组 B 的 A/D 转换。另外，不能使用软件触发。

对于 A/D 转换的对象通道，通过 ADANSA 寄存器选择组 A 的通道，通过 ADANSB 寄存器选择组 B 的通道。组 A 和组 B 不能选择相同的通道。

在分组扫描模式中，温度传感器输出 A/D 转换选择位 (ADEXICR.TSS) 和内部基准电压 A/D 转换选择位 (ADEXICR.OCS) 都被置“0” (不选择)。

在分组扫描模式中选择自诊断时，组 A 和组 B 分别执行自诊断。

通过 MTU 触发进行分组扫描模式的运行例子如下所示。设定为组 A 通过 MTU 的 TRG4AN 触发开始转换，组 B 通过 MTU 的 TRG4BN 触发开始转换。

1. 通过 MTU 的 TRG4AN 触发开始组 A 的扫描。
2. 在组 A 的扫描结束时，如果 ADCSR.ADIE 位已被置“1” (允许 S12ADI0 中断)，就输出 S12ADI0 中断。
3. 通过 MTU 的 TRG4BN 触发开始组 B 的扫描。
4. 在组 B 的扫描结束时，如果 ADCSR.GBADIE 位已被置“1” (允许 GBADI 中断)，就输出 GBADI 中断。

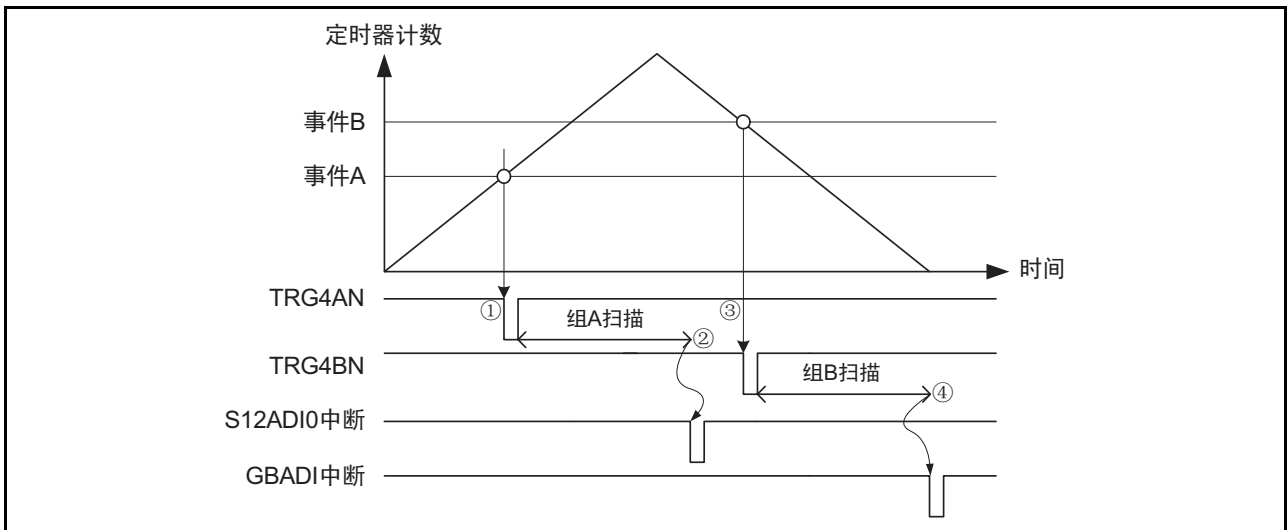


图 32.14 分组扫描模式的运行例子 (通过 MTU 触发进行的基本运行)

32.3.4.2 选择双触发模式的运行

在分组扫描模式中选择双触发模式时，组 A 执行 2 次通过 MTU 触发或者 ELC 触发开始的单次扫描模式，组 B 的运行与通过 MTU 触发或者 ELC 触发开始的单次扫描模式的运行相同。

对于分组扫描模式中的触发设定，通过 ADSTRGR.TRSA[3:0] 位选择组 A 的触发，通过 ADSTRGR.TRSB[3:0] 位选择组 B 的触发。必须给组 A 和组 B 选择不同的触发，以避免同时发生组 A 和组 B 的 A/D 转换。另外，不能使用软件触发、同步触发（温度传感器）以及异步触发（ADTRG0#）。

对于 A/D 转换的对象通道，通过 ADCSR.DBLANS[4:0] 位选择组 A 的通道，通过 ADANSB 寄存器选择组 B 的通道。组 A 和组 B 不能选择相同的通道。

在分组扫描模式中，温度传感器输出 A/D 转换选择位（ADEXICR.TSS）和内部基准电压 A/D 转换选择位（ADEXICR.OCS）都被置“0”（不选择）。

在分组扫描模式中选择双触发模式时，不能选择自诊断。

如果将要双重化的通道号设定到 ADCSR.DBLANS[4:0] 位，并且将 ADCSR.DBLE 位置“1”，A/D 转换数据的双重化就有效。

在 MTU 触发开始的分组扫描模式中选择双触发模式的运行例子如下所示。设定为组 A 通过 MTU 的 TRG4ABN 触发开始转换，组 B 通过 MTU 的 TRG0AN 触发开始转换。

1. 通过 MTU 的 TRG0AN 触发开始组 B 的扫描。
2. 在组 B 的扫描结束时，如果 ADCSR.GBADIE 位已被置“1”（允许 GBADI 中断），就输出 GBADI 中断。
3. 通过 MTU 的第 1 次 TRG4ABN 触发开始组 A 的第 1 次扫描。
4. 在组 A 的第 1 次扫描结束时，将转换数据保存到 ADDRy 寄存器，并且与 ADCSR.ADIE 位的设定无关，不产生 S12ADI0 中断请求。
5. 通过 MTU 的第 2 次 TRG4ABN 触发开始组 A 的第 2 次扫描。
6. 在组 A 的第 2 次扫描结束时，将转换数据保存到 ADDBLDR 寄存器，并且如果 ADCSR.ADIE 位已被置“1”（允许 S12ADI0 中断），就输出 S12ADI0 中断。

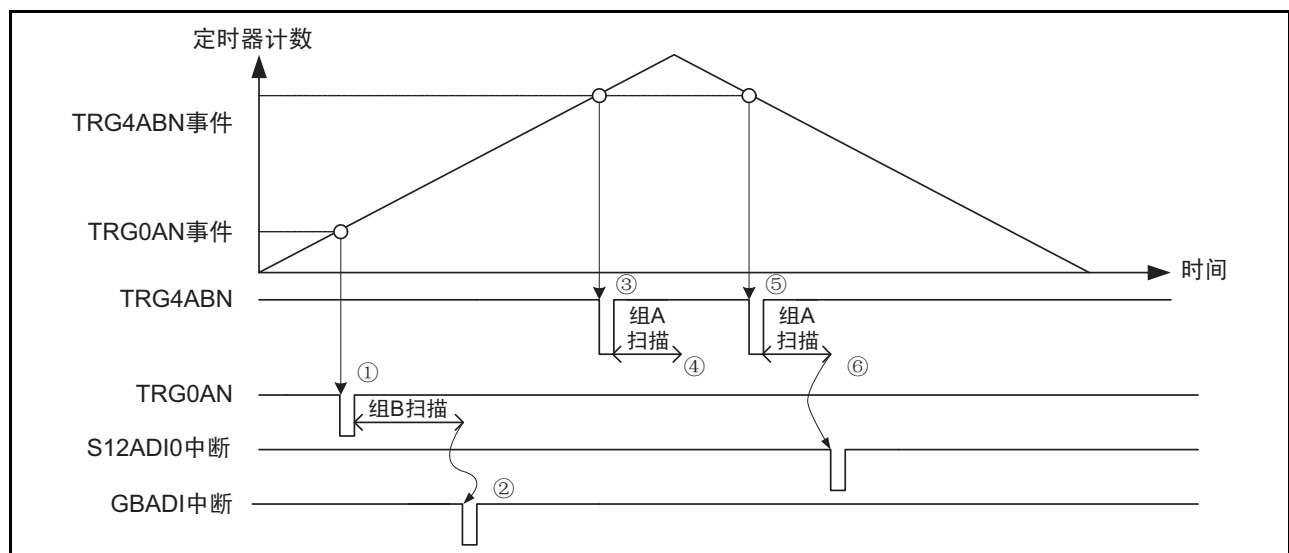


图 32.15 在分组扫描模式中选择双触发模式时的运行例子（通过 MTU 触发进行的基本运行）

32.3.5 模拟输入的采样和扫描转换时间

能选择通过软件触发、MTU 触发、ELC 触发、温度传感器触发或者 ADTRG0# (外部触发) 启动扫描转换。在扫描转换开始延迟时间 (t_D) 后, 执行通道专用采样 & 保持电路的采样、断线检测辅助处理、自诊断转换处理, 然后开始执行 A/D 转换处理。

在单次扫描模式中, 通过软件触发启动和通过 MTU 触发、ELC 触发、或者温度传感器触发启动扫描转换的时序如图 32.16 所示。在单次扫描模式中, 通过 ADTRG0# (外部触发) 启动扫描转换的时序如图 32.17 所示。扫描转换时间 (t_{SCAN}) 包括扫描转换开始延迟时间 (t_D)、通道专用采样 & 保持电路的采样时间 (t_{SH}) (注 1)、断线检测辅助处理时间 (t_{DIS}) (注 2)、自诊断转换时间 (t_{DIAG}) (注 3)、A/D 转换处理时间 (t_{CONV})、通道专用采样 & 保持电路结束处理时间 (t_{SHED}) (注 4) 以及扫描转换结束延迟时间 (t_{ED})。扫描转换时间如表 32.8 所示。

选择的通道数为 n 的单次扫描的扫描转换时间 (t_{SCAN}) 如下所示:

$$t_{SCAN} = t_D + t_{SH} + (t_{DIS} \times n) + t_{DIAG} + (t_{CONV} \times n) + t_{ED}$$

连续扫描的第 1 个周期为: 从单次扫描的 t_{SCAN} 中去掉 t_{ED} 后加上 t_{SHED} 的时间

从连续扫描的第 2 个周期开始固定为: $t_{SH} + (t_{DIS} \times n) + t_{DIAG} + (t_{CONV} \times n) + t_{SHED}$

断线检测辅助处理时间 (t_{DIS}) 为: ADNDIS[3:0] 位的设定值

通道专用采样 & 保持电路的采样时间 (t_{SH}) 为: 10 个状态 (固定) + ADSHCR.SSTSH[7:0] 位的设定值

自诊断转换处理时间 (t_{DIAG}) 和 A/D 转换处理时间 (t_{CONV}) 如下所示:

自诊断转换处理时间 (t_{DIAG}) 为: 30 个状态 (固定) + ADSSTR0.SST[7:0] 位的设定值

A/D 转换处理时间 (t_{CONV}) 为: 30 个状态 (固定) + ADSSTn.SST[7:0] 位 (注 5) 的设定值

注 1. 在不使用通道专用采样 & 保持电路时, t_{SH} 为“0”。

注 2. 在没有设定断线检测辅助功能时, t_{DIS} 为“0”。

注 3. 在没有设定自诊断时, t_{DIAG} 为“0”。

注 4. 在不使用通道专用采样 & 保持电路时, t_{SHED} 为“0”。

这是执行连续扫描模式时所需要的时间。在执行单次扫描模式和分组扫描模式时, 包括在扫描结束处理时间 (t_{ED}) 中。

注 5. 表示表 32.7 中的各寄存器。

表 32.8 扫描转换时间 (通过 PCLK 和 ADCLK 的周期数表示)

项目	符号	种类 / 条件	周期
扫描转换开始延迟时间 (注 1)	t_D	MTU 触发、ELC 触发、温度传感器触发、软件触发	2 个 PCLK+4 个 ADCLK
		外部触发	4 个 PCLK+4 个 ADCLK
通道专用采样 & 保持电路的采样时间	t_{SH}	ADSHCR.SSTSH[7:0] 位 (初始设定值: 14h)	30 个 ADCLK
断线检测辅助处理时间	t_{DIS}	ADNDIS[3:0] 位 (初始设定值: 00h)	0 个 ADCLK
自诊断转换处理时间 (注 1)	t_{DIAG}	ADSSTR0.SST[7:0] 位 (初始设定值: 14h)	50 个 ADCLK
A/D 转换处理时间 (注 1)	t_{CONV}	ADSSTn.SST[7:0] 位 (初始设定值: 14h)	50 个 ADCLK
通道专用采样 & 保持电路结束处理时间	t_{SHED}	—	2 个 ADCLK
扫描转换结束延迟时间 (注 1)	t_{ED}	—	1 个 PCLK+3 个 ADCLK
扫描转换时间 (注 2)	t_{SCAN}	—	5 个 PCLK+(50n+87) 个 ADCLK

注 1. 有关 t_D 、 t_{DIAG} 、 t_{CONV} 和 t_{ED} 的各时序, 请参照图 32.16 和图 32.17。

注 2. 这是通过异步触发启动, 并且选择通道专用采样 & 保持电路、不选择断线检测辅助功能、选择自诊断转换以及选择单次扫描模式时的情况。n 为通道数。

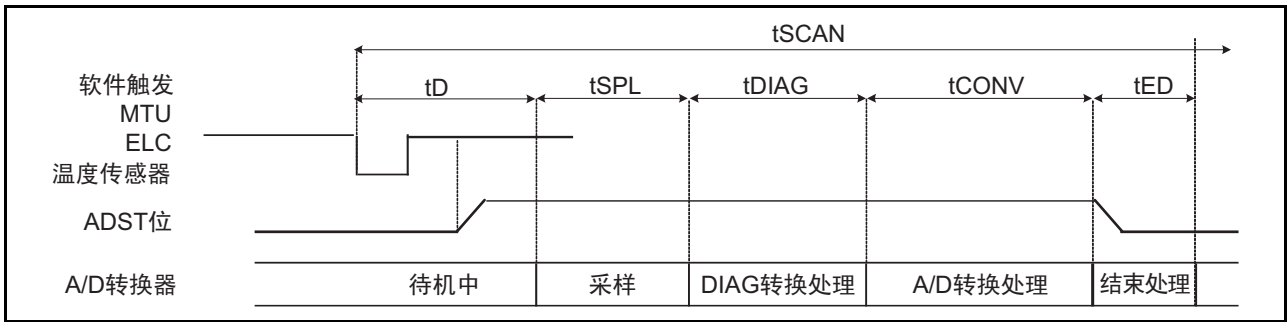


图 32.16 扫描转换的时序 (通过软件触发启动、MTU 触发、ELC 触发或者温度传感器触发启动的情况)

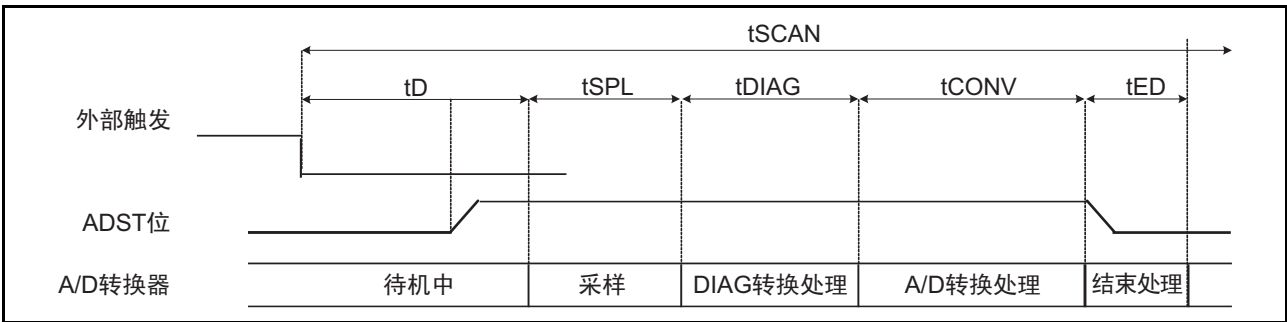


图 32.17 扫描转换的时序 (通过 ADTRG# 启动的情况)

32.3.6 寄存器的自动清除功能的使用例子

能通过将 ADCER.ACE 位置“1”，在 CPU、DTC 或者 DMAC 读 A/D 数据寄存器 (ADDRy、ADRD、ADTSDR、ADOCDR) 时将 ADDRy、ADRD、ADTSDR、ADOCDR 寄存器自动清“0000h”。

能通过使用自动清除功能检测到 ADDRy、ADRD、ADTSDR、ADOCDR 寄存器未更新的故障。ADDRy 寄存器的自动清除功能无效 / 有效时的例子如下所示。

当 ADCER.ACE 位为“0” (禁止自动清除) 时，如果因某种原因而无法将 A/D 转换结果 (0222h) 写到 ADDRy 寄存器，以前的数据 (0111h) 就为 ADDRy 寄存器的值。如果使用 A/D 转换结束中断将此 ADDRy 寄存器的值读到通用寄存器，就能将以前的数据 (0111h) 保存到通用寄存器等。但是，在进行未更新的检查时，需要边将以前的数据逐个保存到 RAM 或者通用寄存器边进行检查。

当 ADCER.ACE 位为“1” (允许自动清除) 并且通过 CPU、DTC 或者 DMAC 读到 ADDRy 寄存器为“0111h”时，ADDRy 寄存器自动清“0000h”。此后，如果因某种原因而无法将 A/D 转换结果“0222h”传送到 ADDRy 寄存器，被清除的数据 (0000h) 就留作 ADDRy 寄存器的值。此时，如果使用 A/D 转换结束中断将 ADDRy 寄存器的值读到通用寄存器等，就将“0000h”保存到通用寄存器等。只要检查到读取的数据值为“0000h”，就能判断 ADDRy 寄存器发生未更新的故障。

32.3.7 A/D 转换值加法运算功能

对相同的通道连续进行 2 ~ 4 次 A/D 转换后，将此转换值的合计保存到数据寄存器。使用此结果的平均值，根据噪声成分能提高 A/D 转换精度，但是不保证一定能提高 A/D 转换精度。

能在选择所选通道的模拟输入 A/D 转换或者内部基准电压 A/D 转换时使用 A/D 转换值加法运算功能。

32.3.8 断线检测辅助功能

12 位 A/D 转换器内置断线检测辅助功能。该功能在 A/D 转换开始前，将采样电容的电荷固定到规定的状态 (VREFH0 或者 VREFL0)。能通过该功能检测连接模拟输入的布线是否断线。

使用断线检测辅助功能时的 A/D 转换运行例子如图 32.18 所示。另外，在 VREFH0 侧的断线检测例子 (选择预充电) 如图 32.19，在 VREFL0 侧的断线检测例子 (选择放电) 如图 32.20 所示。

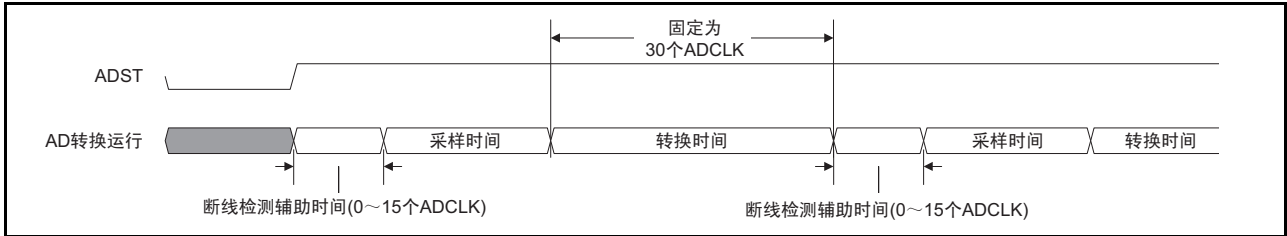


图 32.18 使用断线检测辅助功能时的 A/D 转换运行例子

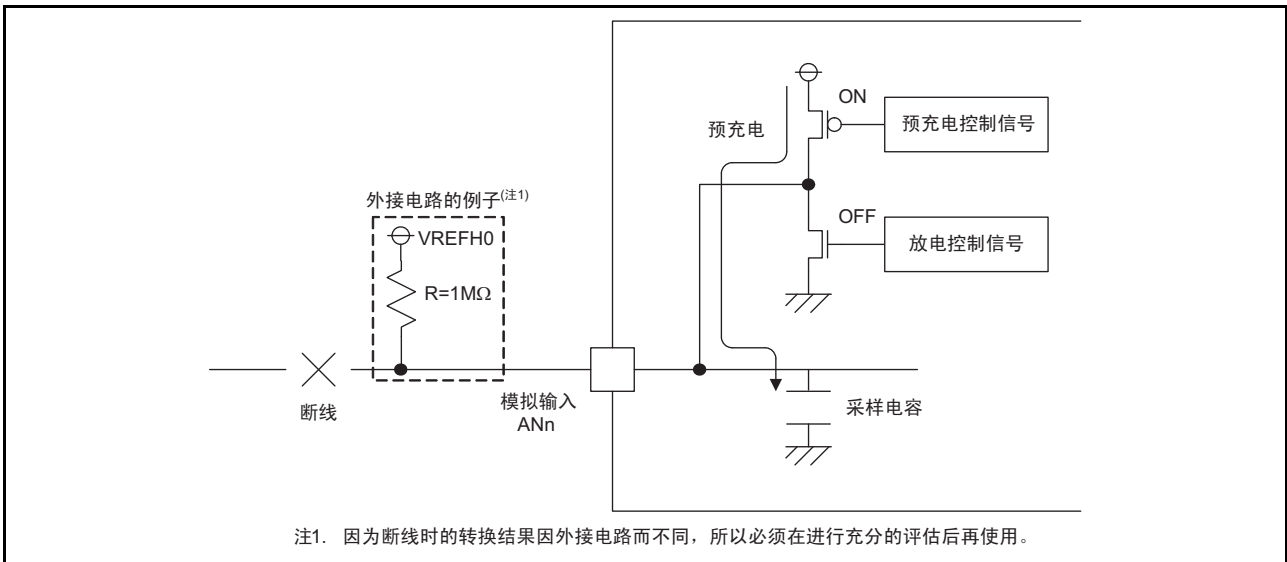


图 32.19 在 VREFH0 侧的断线检测例子 (选择预充电)

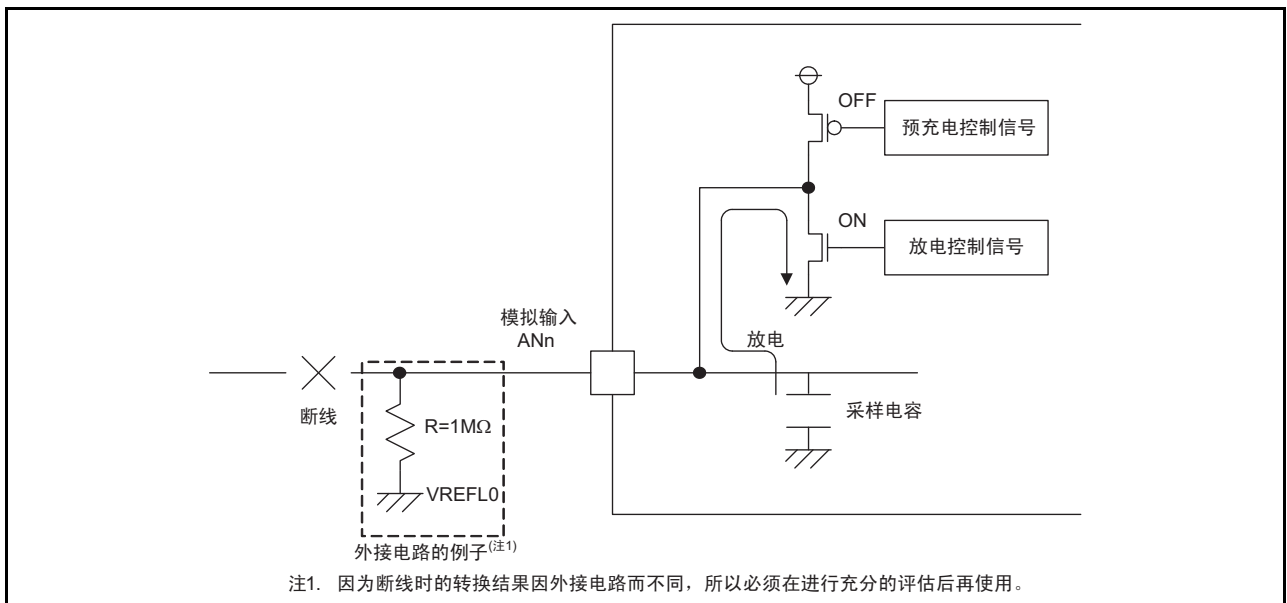


图 32.20 在 VREFL0 侧的断线检测例子 (选择放电)

32.3.9 通过异步触发开始的 A/D 转换

能通过输入异步触发来开始 AD 转换。使用异步触发开始 A/D 转换时，先将 A/D 转换开始触发选择位 (ADSTRGR.TRGA[3:0]) 置“0000b”，然后在给异步触发 (ADTRG0# 引脚) 输入 High 电平后，将 ADCSR.TRGE 位和 ADCSR.EXTRG 位都置“1”。异步触发的输入时序如图 32.21 所示。

有关从 ADST 位被置位到开始 A/D 转换为止的时间，请参照“32.7.3 开始和强制停止 A/D 转换时的运行时序”。

在分组扫描模式中使用的组 B 专用 A/D 转换开始触发选择位 (ADSTRGR.TRSB[3:0] 位) 不能选择异步触发。

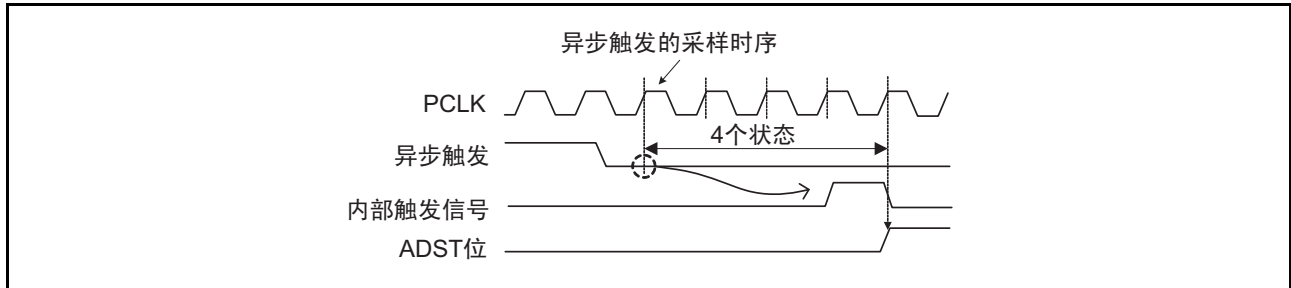


图 32.21 异步触发的输入时序

32.3.10 通过外围模块的同步触发开始的 A/D 转换

能通过 MTU 触发、ELC 触发或者温度传感器触发等同步触发来开始 A/D 转换。在通过同步触发开始 A/D 转换时，将 ADCSR.TRGE 位置“1”，将 ADCSR.EXTRG 位置“0”并且通过 ADSTRGR.TRSA[3:0] 位和 ADSTRGR.TRSB[3:0] 位设定为对应的启动源。

在对温度传感器输出进行转换时，必须通过温度传感器的触发开始 A/D 转换。在对模拟输入通道或者内部基准电压进行转换时，不能使用温度传感器的触发。

32.4 中断源和 DMA 传送请求

32.4.1 扫描结束时的中断请求

12 位 A/D 转换器能向 CPU 请求扫描结束中断 (S12ADIO 中断和 GBADI 中断)。

如果将 ADCSR.ADIE 位置“1”，就允许 S12ADIO 中断；置“0”，就禁止 S12ADIO 中断。如果将 ADCSR.GBADIE 位置“1”，就允许 GBADI 中断；置“0”，就禁止 GBADI 中断。

另外，在发生 S12ADIO 中断和 GBADI 中断时能启动 DTC 或者 DMAC。如果通过 DTC 或者 DMAC 读由 S12ADIO 中断或者 GBADI 中断转换的数据，就能无软件负担的实现连续转换。DTC 的设定请参照“17. 数据传送控制器 (DTCa)”、DMAC 的设定请参照“16. DMA 控制器 (DMACA)”。

32.5 事件链接功能

32.5.1 向 ELC 输出事件的运行

通过事件链接控制器 (ECL)，12 位 A/D 转换器将 S12ADIO 中断请求信号用作事件信号，并且对事先设定的模块进行链接运行。不能将 GBADI 中断请求信号用作事件信号。与对应的中断请求允许位的设定无关，能输出事件信号。12 位 A/D 转换器输出 A/D 转换结束事件。

32.5.2 通过 ELC 的事件进行的 12 位 A/D 转换器运行

12 位 A/D 转换器能通过事先设定的事件 (根据 ELC 的 ELSRn 的设定) 开始 A/D 转换的运行。

32.5.3 通过 ELC 的事件进行的 12 位 A/D 转换器运行的注意事项

在 A/D 转换过程中发生事件时，事件无效。

32.6 A/D 转换精度的定义

A/D 转换精度的定义如下：

- 分辨率
这是指 12 位 A/D 转换器的数字输出码数。
- 偏移误差
这是在数字输出从最小电压值“000000000000”变为“000000000001”时与模拟输入电压值的理想 A/D 转换特性的偏差。但是不包含量化误差。
- 满刻度误差
这是在数字输出从“11111111110”变为“11111111111”时与模拟输入电压值的理想 A/D 转换特性的偏差。但是不包含量化误差。
- 量化误差
这是 12 位 A/D 转换器固有的偏差，为 1/2LSB。
- 非线性误差
这是在零电压到满刻度电压之间的理想 A/D 转换特性的误差，但是不包含偏移误差、满刻度误差和量化误差。
- 绝对精度
这是数字值和模拟输入值的偏差，包含偏移误差、满刻度误差、量化误差和非线性误差。

32.7 使用时的注意事项

32.7.1 读数据寄存器的注意事项

必须以字为单位读 A/D 数据寄存器、A/D 数据双重化寄存器、A/D 温度传感器的数据寄存器、A/D 内部基准电压的数据寄存器以及 A/D 自诊断数据寄存器。为了避免因以字节为单位分 2 次读高位字节和低位字节而使第 1 次读到的 A/D 转换值和第 2 次读到的 A/D 转换值发生变化，不能以字节为单位读数据。

32.7.2 停止 A/D 转换时的注意事项

在选择了异步触发或者同步触发为 A/D 转换开始条件时，为了停止 A/D 转换，必须在将 ADCSR.TRGE 位置“0”并且将 A/D 转换开始条件设定为软件触发后，将 ADCSR.ADST 位置“0”（停止 A/D 转换）。

32.7.3 开始和强制停止 A/D 转换时的运行时序

在 12 位 A/D 转换器的模拟部停止的状态下，将 ADCSR.ADST 位置“1”，需要最多 4 个 ADCLK 时钟的时间来开始 12 位 A/D 转换器模拟部的运行。如果通过将 ADCSR.ADST 位置“0”来强制停止 A/D 转换，就需要最多 2 个 ADCLK 时钟的时间来停止 12 位 A/D 转换器模拟部的运行。

32.7.4 扫描结束中断处理的注意事项

在通过触发启动对相同模拟输入进行 2 次扫描时，从发生第 1 次扫描结束中断到通过第 2 次扫描进行第 1 个模拟输入的 A/D 转换结束为止，如果 CPU 未读完 A/D 转换数据，第 2 次的 A/D 转换数据就盖写第 1 次的 A/D 转换数据。

32.7.5 模块停止功能的设定

能通过模块停止控制寄存器设定禁止或者允许 12 位 A/D 转换器的运行，初始值为停止 12 位 A/D 转换器的运行。通过解除模块停止状态，使寄存器变为可存取的状态。在解除模块停止状态后，必须在等待 1 μ s 后开始 A/D 转换。详细内容请参照“11. 低功耗功能”。

32.7.6 向低功耗状态转移时的注意事项

要向模块停止模式或者软件待机模式转移时，必须停止 A/D 转换。在停止 A/D 转换时，需要确保从将 ADCSR.ADST 位置“0”到 12 位 A/D 转换器的模拟部停止为止的时间。为了确保此时间，必须按照以下步骤进行设定。

必须在将 ADCSR.TRGE 位置“0”（软件触发）后将 ADCSR.ADST 位置“0”。此后，必须在确认 A/D 转换已经停止后转移到模块停止状态或者软件待机模式。

32.7.7 有关容许信号源阻抗

为了实现 $1.0\mu\text{s}$ 的高速转换，对于信号源阻抗不超过 $0.5\text{k}\Omega$ 的输入信号，本 LSI 的模拟输入能保证转换精度。在单次扫描模式中只进行 1 个引脚的转换并且外接大电容的情况下，因为输入负载实际上只有 $3.0\text{k}\Omega$ 的内部输入电阻，所以信号源阻抗可忽略不计。但是，由于形成低通滤波器，所以有可能无法跟上微分系数大的模拟信号（例如至少 $5\text{mV}/\mu\text{s}$ ）（图 32.22）。在转换高速模拟信号或者在扫描模式中进行多个引脚的转换时，必须插入阻抗低的缓冲器。

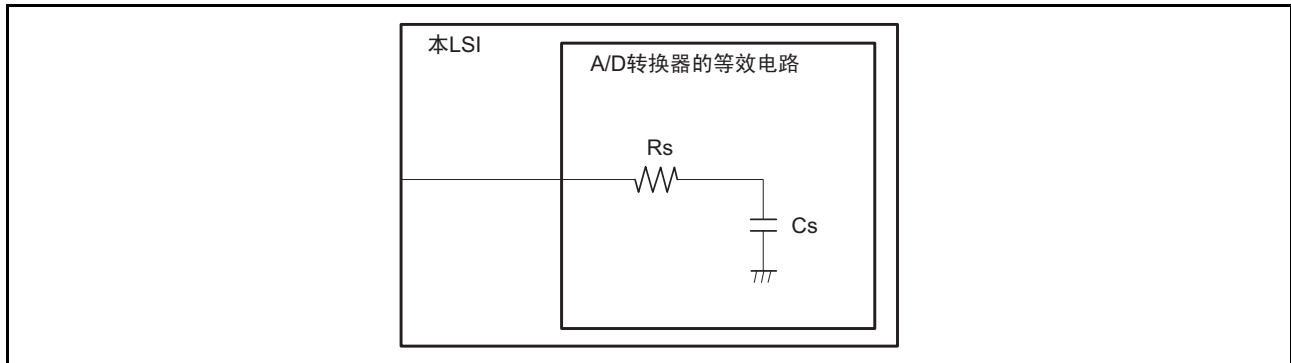


图 32.22 模拟输入引脚的内部等效电路

表 32.9 模拟引脚的规格

项目	min	max	单位
允许信号源阻抗 (注1)	—	0.5	$\text{k}\Omega$
引脚的内部等效电路 (注2)	Rs	3.0	$\text{k}\Omega$
	Cs	20	pF

注 1. 因模拟电源电压和模拟输入引脚而不同。详细内容请参照“41. 电特性”。

注 2. 这是对应引脚为 AN003、AN004 和 AN006，电压条件为 $\text{AVCC} \geq 2.7\text{V}$ 时的值。

32.7.8 对绝对精度的影响

由于附加电容会导致与 GND 的耦合，如果 GND 有噪声就可能降低绝对精度，因此必须连接 AVSS0 等电特性稳定的 GND。

必须注意：滤波器电路不要干扰安装电路板上的数字信号，也不要充当天线。

32.7.9 模拟电源引脚以外的设定范围

如果超过以下所示电压的设定范围使用 LSI，就可能给 LSI 的可靠性带来不良影响。

- 模拟输入电压的设定范围
给模拟输入引脚 ANn 外加的电压必须在 $VREFL0 \leq VAN \leq VREFH0$ 的范围内。
- 各电源引脚 (AVCC0-AVSS0、VREFH0-VREFL0、VCC-VSS) 的关系
AVCC0、AVSS0 与 VCC、VSS 的关系必须是 AVCC0=VCC 并且 AVSS0=VSS。为了能形成最短的闭环，必须在各电源之间连接 0.1 μ F 的电容器，如图 32.23 所示。必须在供电处连接为 AVCC0=VCC、VREFL0=AVSS0=VSS。在不使用 12 位 A/D 转换器时，必须使 VREFH0=AVCC0=VCC、VREFL0=AVSS0=VSS。
- VREFH0 的设定范围
VREFH0 引脚的基准电压设定范围必须为 $VREFH0 \leq AVCC0$ 。

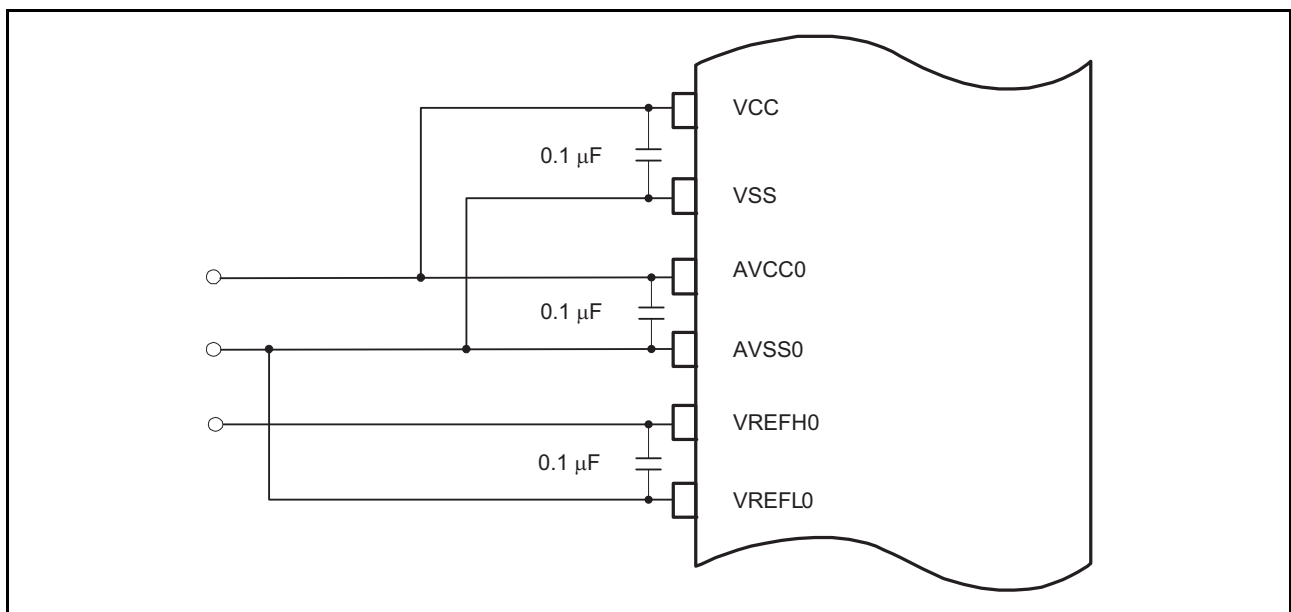


图 32.23 各电源引脚的连接例子

32.7.10 电路板设计时的注意事项

在设计电路板时，必须尽量将数字电路和模拟电路分开，而且不能使数字电路的信号线和模拟电路的信号线交叉或者靠近，否则会给模拟信号带来噪声并且给 A/D 转换值的精度带来不良影响。必须通过模拟接地 (AVSS0) 将模拟输入引脚 (AN000 ~ AN015)、模拟基准电压 (VREFH0、VREFL0)、模拟电源 (AVCC0) 和数字电路分开，而且模拟接地 (AVSS0) 必须与电路板上稳定的数字接地 (VSS) 进行单点连接。

32.7.11 噪声对策的注意事项

如图 32.24 所示，为了防止因过大的电涌等异常电压给模拟输入引脚（AN000 ~ AN015）造成破坏，必须在 AVCC0 和 AVSS0 之间、VREFH0 和 VREFL0 之间连接电容，并且以模拟输入引脚（AN000 ~ AN015）为基准连接保护电路。

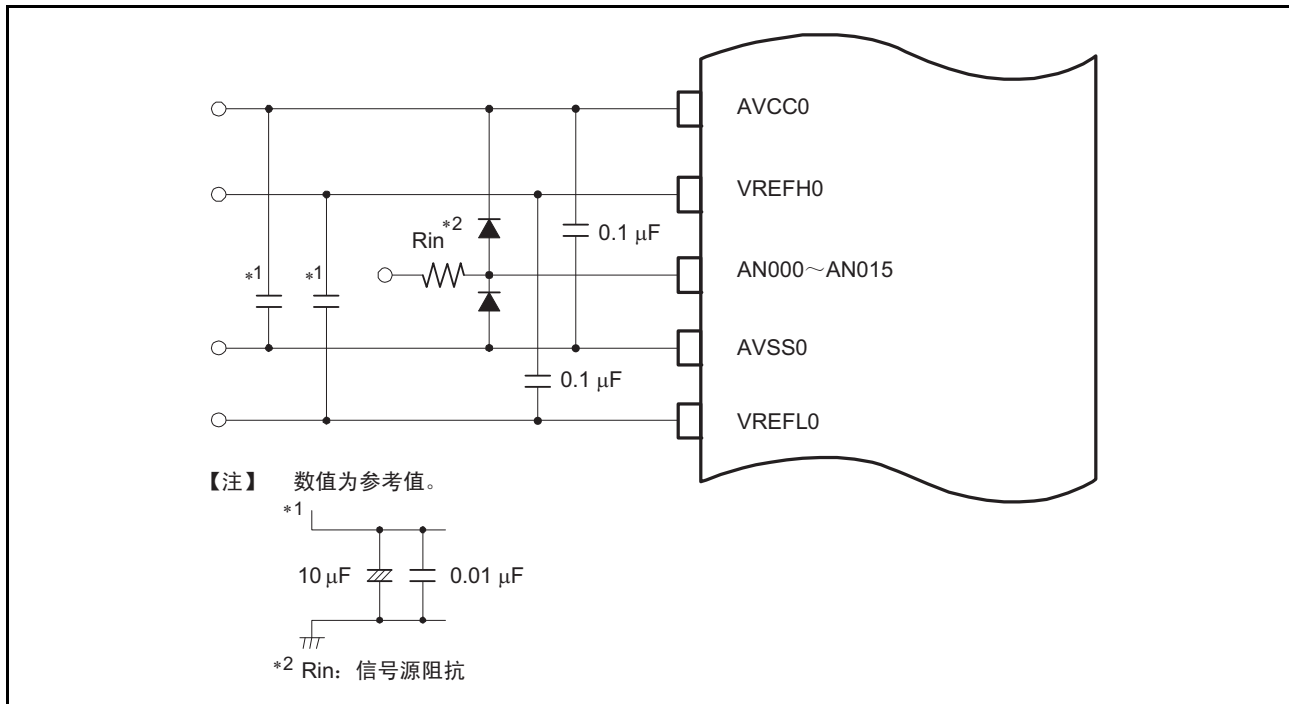


图 32.24 模拟输入保护电路的例子

32.7.12 使用 12 位 A/D 转换器输入时的端口设定

在端口 4 和端口 E 的引脚中，只要有 1 个引脚用作 12 位 A/D 转换器的模拟输入引脚，就不能使用端口 0 和端口 4 的端口输出。这是因为在端口 0 和端口 4 的部分电路中使用模拟电源。

32.7.13 使用断线检测辅助功能时的绝对精度误差

在使用断线检测辅助功能时，如果在模拟输入引脚中输入上拉/下拉电阻（ R_p ）和信号源电阻（ R_s ）的电阻分压的误差电压，就产生 A/D 转换器的绝对精度误差。用以下表达式表示绝对精度的误差。必须在充分评估断线检测辅助功能后再使用。

$$\text{最大绝对精度误差 (LSB)} = 4095 \times R_s / R_p$$

33. D/A 转换器 (DA)

33.1 概要

RX210 群内置 2 个通道的 10 位 D/A 转换器。

D/A 转换器的规格和框图分别如表 33.1 和图 33.1 所示。

表 33.1 D/A 转换器的规格

项目	内容
分辨率	10 位
输出通道	2 个通道
低功耗功能	能设定为模块停止状态。
事件链接功能 (输入)	能通过输入事件信号开始 D/A0 转换。

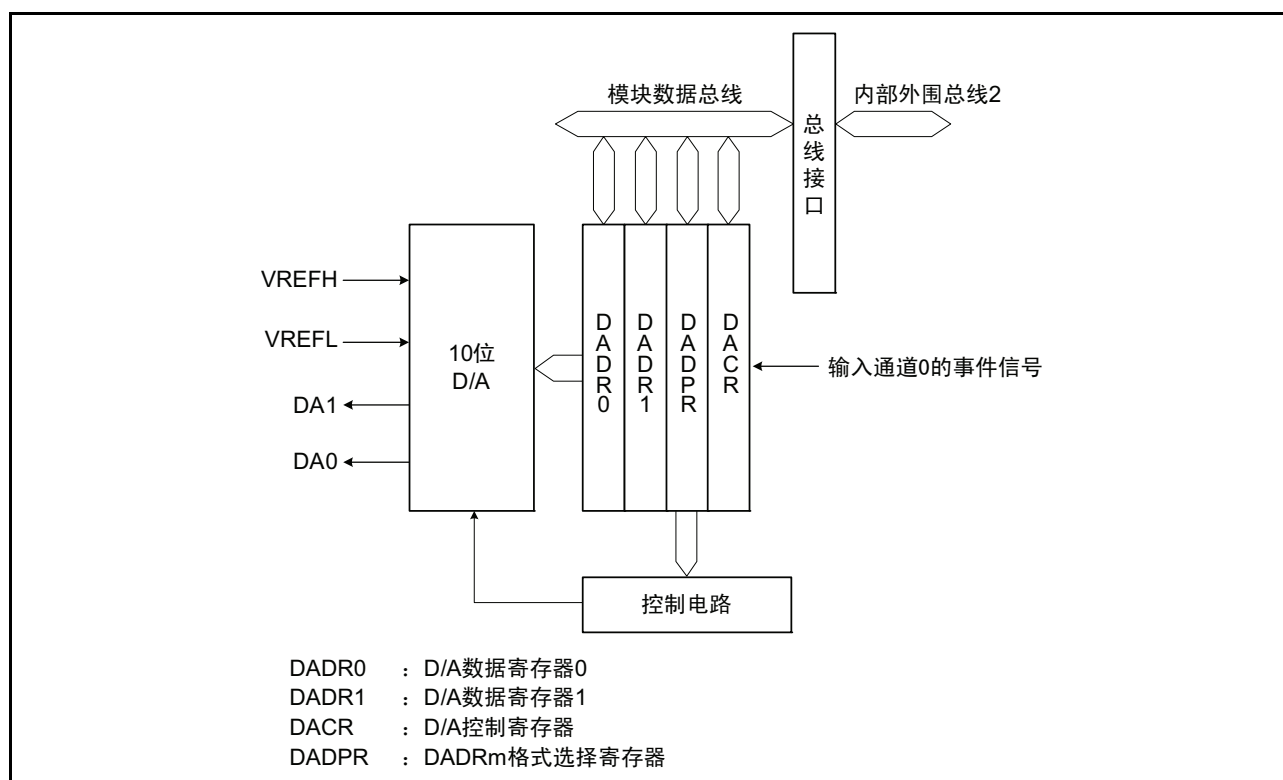


图 33.1 D/A 转换器的框图

D/A 转换器使用的输入 / 输出引脚如表 33.2 所示。

表 33.2 D/A 转换器的输入 / 输出引脚

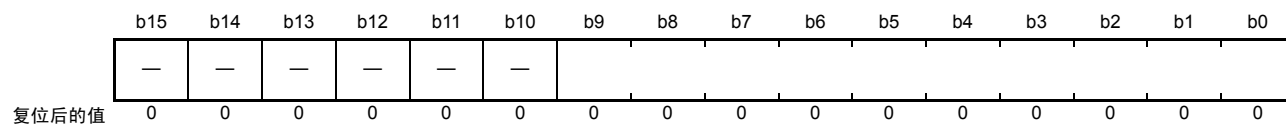
引脚名	输入 / 输出	功能
V_{REFH}	输入	D/A 转换器的基准电压输入引脚 也能用作模拟电源引脚。在不使用 D/A 转换器时，必须连接 VCC。
V_{REFL}	输入	D/A 转换器的基准电压输入引脚 也能用作模拟接地引脚。必须设定与 VSS 引脚相同的电位。
$DA0$	输出	通道 0 的模拟输出
$DA1$	输出	通道 1 的模拟输出

33.2 寄存器说明

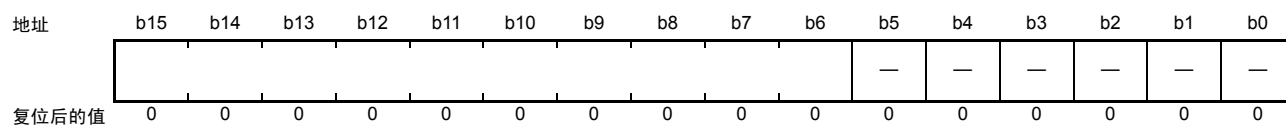
33.2.1 D/A 数据寄存器 m (DADRm) (m=0、1)

地址 DADR0 0008 80C0h、DADR1 0008 80C2h

- DADPR.DPSEL 位 =0 (数据往右靠紧)



- DADPR.DPSEL 位 =1 (数据往左靠紧)



DADRm 寄存器是保存 D/A 转换数据的 16 位可读写寄存器。如果允许模拟输出，就转换 DADRm 寄存器的值并且输出到模拟输出引脚。

能通过设定 DADPR.DPSEL 位，更改 10 位数据的排列。“—”的位的读写值都为“0”。

33.2.2 D/A 控制寄存器 (DACR)

地址 0008 80C4h

b7	b6	b5	b4	b3	b2	b1	b0
DAOE1	DAOE0	DAE	—	—	—	—	—
0	0	0	1	1	1	1	1

复位后的值

位	符号	位名	功能	R/W
b4-b0	—	保留位	读写值都为“1”。	R/W
b5	DAE (注1)	D/A 允许位	0: 分别控制通道 0 和通道 1 的 D/A 转换 1: 同时允许通道 0 和通道 1 的 D/A 转换	R/W
b6	DAOE0	D/A 输出允许 0 位	0: 禁止通道 0 的模拟输出 (DA0) 1: 允许通道 0 的 D/A 转换 允许通道 0 的模拟输出 (DA0) (注2)	R/W
b7	DAOE1	D/A 输出允许 1 位	0: 禁止通道 1 的模拟输出 (DA1) 1: 允许通道 1 的 D/A 转换 允许通道 1 的模拟输出 (DA1) (注2)	R/W

注 1. 通过和 DAOEi 位 (i=0、1) 的组合对 D/A 转换进行控制, 由 DAOEi 位控制转换结果的输出, 详细内容请参照表 33.3。

注 2. 必须将用作模拟输出引脚的 PORT0.PDR.Bm 位 (m=3、5) 和 PORT0.PMR.Bm 位 (m=3、5) 都置“0”。另外, 必须通过 P03PFS 寄存器和 P05PFS 寄存器设定为模拟引脚。详细内容请参照“19. I/O 端口”和“20. 多功能引脚控制器 (MPC)”。

表 33.3 D/A 转换的控制

b5	b7	b6	说明
DAE	DAOE1	DAOE0	
0	0	0	禁止 D/A 转换和模拟输出 (DA0、DA1) (注1)。
		1	允许通道 0 的 D/A 转换, 禁止通道 1 的 D/A 转换。 允许通道 0 的模拟输出 (DA0), 禁止通道 1 的模拟输出 (DA1) (注1)。
	1	0	禁止通道 0 的 D/A 转换, 允许通道 1 的 D/A 转换。 禁止通道 0 的模拟输出 (DA0) (注1), 允许通道 1 的模拟输出 (DA1)。
		1	允许通道 0 和通道 1 的 D/A 转换。 允许通道 0 和通道 1 的模拟输出 (DA0 和 DA1)。
1	0	0	允许通道 0 和通道 1 的 D/A 转换。 禁止通道 0 和通道 1 的模拟输出 (DA0 和 DA1) (注1)。
		1	允许通道 0 和通道 1 的 D/A 转换。 允许通道 0 的模拟输出 (DA0), 禁止通道 1 的模拟输出 (DA1) (注1)。
	1	0	允许通道 0 和通道 1 的 D/A 转换。 禁止通道 0 的模拟输出 (DA0) (注1), 允许通道 1 的模拟输出 (DA1)。
		1	允许通道 0 和通道 1 的 D/A 转换。 允许通道 0 和通道 1 的模拟输出 (DA0 和 DA1)。

注 1. 禁止模拟输出时, 模拟输出变为 Hi-Z。

DAE 位 (D/A 允许位)

此位通过和 DAOE_i 位 (i=0、1) 的组合对 D/A 转换进行控制。

当 DAE 位为“0”时，分别控制通道 0 和通道 1 的 D/A 转换；当 DAE 位为“1”时，同时控制通道 0 和通道 1 的 D/A 转换。由 DAOE_i 位控制转换结果的输出。

DAOE0 位 (D/A 输出允许 0 位)

此位控制 D/A 转换和模拟输出。

能通过事件链接功能将 DAOE0 位置“1”。如果发生 ELC 的 ELSR16 寄存器设定的事件，DAOE0 位就变为“1”，并且开始 D/A 转换输出。

DAOE1 位 (D/A 输出允许 1 位)

此位控制 D/A 转换和模拟输出。

33.2.3 DADR_m 格式选择寄存器 (DADPR)

地址 0008 80C5h

	b7	b6	b5	b4	b3	b2	b1	b0
	DPSEL	—	—	—	—	—	—	—
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b6-b0	—	保留位	读写值都为“0”。	R/W
b7	DPSEL	DADR _m 格式选择位	0: D/A 数据寄存器往右靠紧 1: D/A 数据寄存器往左靠紧	R/W

33.3 运行说明

2 个通道的 D/A 转换器能分别独立进行转换。如果将 DACR.DA0Ei 位 (i=0、1) 置“1”，就允许 D/A 转换并且输出转换结果。

进行通道 0 的 D/A 转换时的运行例子如下所示，此时的运行时序如图 33.2 所示。

1. 给 DADR0 寄存器写转换数据。
2. 如果将 DACR.DA0E0 位置“1”，就开始进行 D/A 转换。在经过 tD CONV 时间后，从模拟输出引脚 DA0 输出转换结果。在改写 DADR0 寄存器或者将 DA0E0 位置“0”前，连续输出该转换结果。用以下表达式计算输出值：

$$\frac{\text{DADR0寄存器的值}}{1024} \times \text{VREFH}$$

3. 如果改写 DADR0 寄存器，就开始转换。在经过 tD CONV 时间后，输出转换结果。
4. 如果将 DA0E0 位置“0”，就禁止模拟输出。

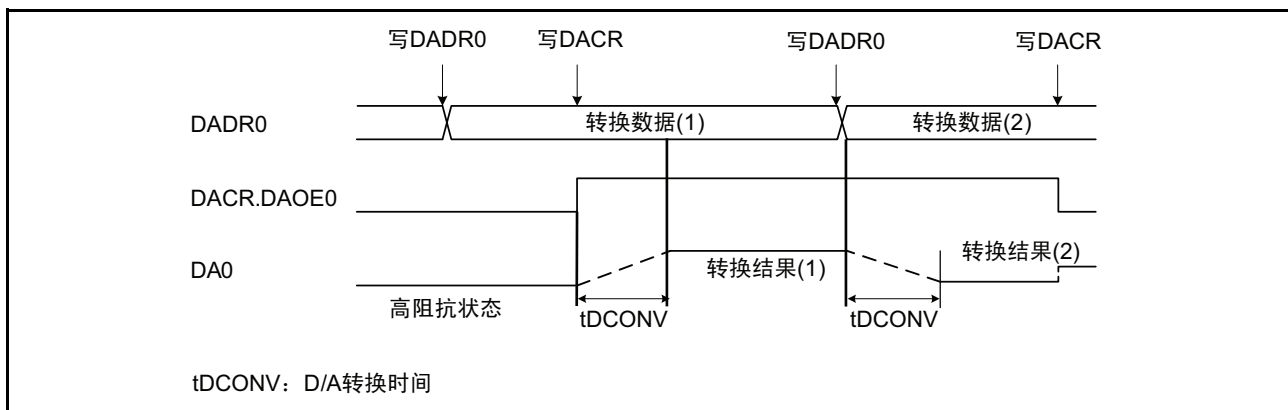


图 33.2 D/A 转换器的运行例子

33.4 事件链接运行的设定步骤

事件链接运行的设定步骤如下所示。

1. 设定 DADPR.DPSEL 位，并且给 DADR0 寄存器设定用于 D/A 转换的数据。
2. 设定与 ELC 的 ELSR16 寄存器链接的 ELS16 设定事件信号位的值。
3. 通过将 ELCR.ELCON 位置“1”，使设定了事件链接的模块的事件链接运行全部变为有效。
4. 设定事件输出源模块，并且启动。通过该模块输出的事件，使 DACR.DA0E0 位变为“1”，并且开始通道 0 的 D/A 转换。
5. 如果要停止 D/A 转换器通道 0 的事件链接运行，就必须将 ELSR16.ELS[7:0] 位设定为“0000 0000b”。另外，能通过将 ELCR.ELCON 位置“0”，停止全部模块的事件链接运行。

33.5 事件链接运行时的注意事项

如果在写 DACR.DA0E0 位的过程中发生 ELSR16 寄存器指定的事件，就不写 DACR.DA0E0 位，而优先通过发生的事件将 DACR.DA0E0 位置“1”。

33.6 使用时的注意事项

33.6.1 模块停止功能的设定

能通过模块停止控制寄存器设定禁止或者允许 D/A 转换器的运行，初始值为停止 D/A 转换器的运行。能通过解除模块停止状态，使寄存器变为可存取的状态。详细内容请参照“11. 低功耗功能”。

33.6.2 模块停止时的 D/A 转换器的运行

如果在允许 D/A 转换的状态下变为模块停止状态，就保持 D/A 输出，模拟电源电流和 D/A 转换过程中的电流相同。如果需要在模块停止时减少模拟电源电流，就必须在将 DACR.DAOE1 位、DACR.DAOE0 位和 DACR.DAE 位全部置“0”后禁止 D/A 输出。

33.6.3 软件待机模式时的 D/A 转换器的运行

如果在允许 D/A 转换的状态下 RX210 群转移到软件待机模式，就保持 D/A 输出，模拟电源电流和 D/A 转换过程中的电流相同。如果需要在软件待机模式中减少模拟电源电流，就必须在将 DACR.DAOE1 位、DACR.DAOE0 位和 DACR.DAE 位全部置“0”后禁止 D/A 输出。

33.6.4 深度软件待机模式时的注意事项

如果在允许 D/A 转换的状态下 RX210 群转移到深度软件待机模式，D/A 输出就变为高阻抗状态。

33.6.5 使用 D/A 转换器输出时的端口设定

在端口 05 和端口 03 的引脚中，只要有 1 个引脚用作 D/A 转换器的输出引脚，就不能使用端口 0 和端口 4 的端口输出。这是因为在端口 0 和端口 4 的部分电路中使用模拟电源。

34. 温度传感器 (TEMPSa)

34.1 概要

RX210 群内置温度传感器，温度传感器输出随温度而变化的电压。因为温度传感器的输出连接 PGA (Programmable Gain Amp)，因此能放大温度传感器的输出电压。能通过 12 位 A/D 转换器将被放大的温度传感器的输出电压转换为数字值，并且通过将此数字值换算为温度来求 LSI 的外围温度。

温度传感器的规格和框图分别如表 34.1 和图 34.1 所示。

表 34.1 温度传感器的规格

项目	内容
温度传感器电压输出	经由 PGA (Programmable Gain Amp) 输出到 12 位 A/D 转换器。
低功耗功能	能设定为模块停止状态。

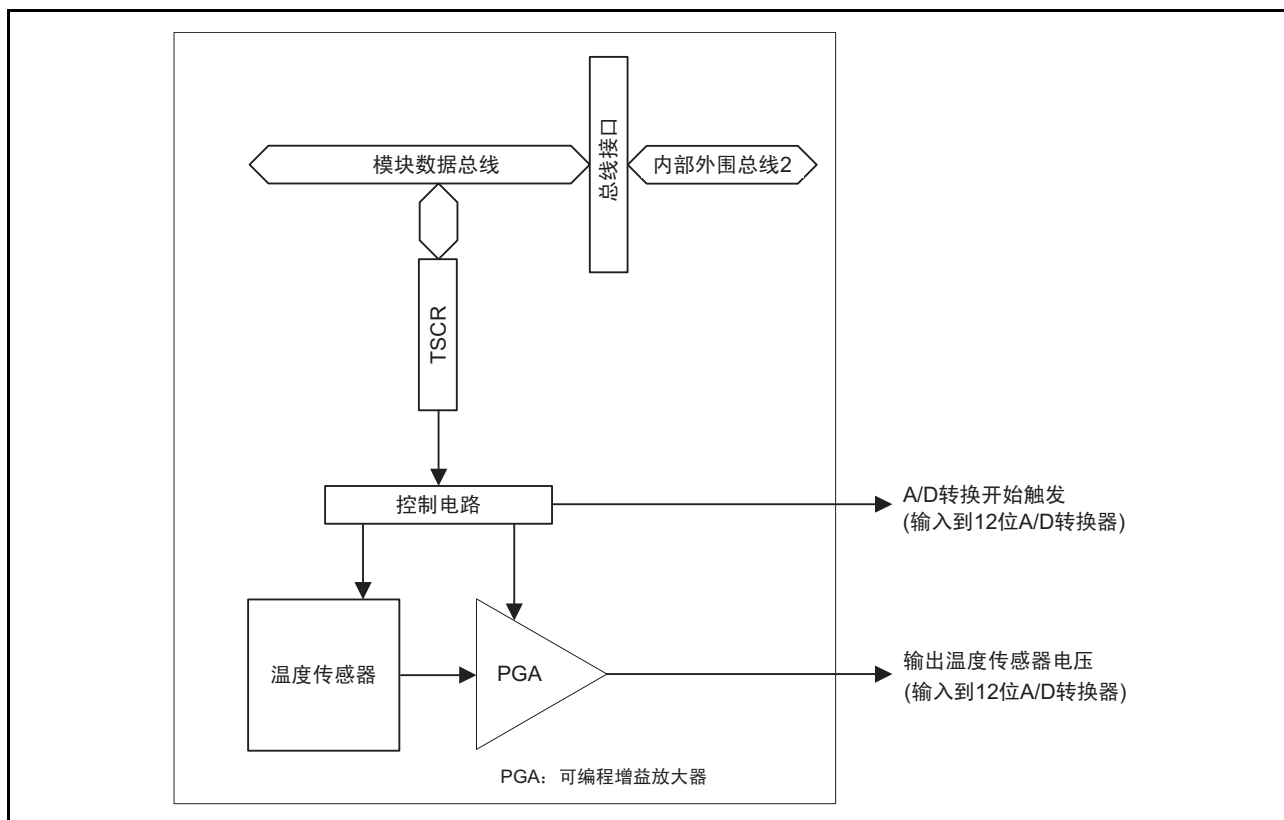


图 34.1 温度传感器的框图

34.2 寄存器说明

34.2.1 温度传感器的控制寄存器 (TSCR)

地址 0008 C500h

	b7	b6	b5	b4	b3	b2	b1	b0
	TSEN	PGAEN	—	—	—	—	PGAGAIN[1:0]	
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b1-b0	PGAGAIN [1:0]	PGA 增益选择位	b1 b0 0 0: $1.8V \leq AVCC < 2.7V$ 0 1: $2.7V \leq AVCC < 3.6V$ 1 0: $3.6V \leq AVCC < 4.5V$ 1 1: $5.5V \leq AVCC \leq 5.5V$	R/W
b5-b2	—	保留位	读写值都为“0”。	R/W
b6	PGAEN	PGA 允许位 (注1)	0: PGA 停止 1: PGA 运行	R/W
b7	TSEN	温度传感器允许位 (注2)	0: 温度传感器停止 1: 温度传感器运行	R/W

注1. 在 TSEN 位为“1” (温度传感器运行) 的状态下写 PGAEN 位。

注2. 在 PGAEN 位为“0” (PGA 停止) 的状态下写 TSEN 位。

PGAGAIN[1:0] 位 (PGA 增益选择位)

这些位根据 AVCC0 设定 PGA 的增益, 以放大温度传感器的输出电压从而在 A/D 转换器取得充分的分辨率。功能栏中的 AVCC0 电压值为基准值。当 AVCC0 的外加电压大于设定电压时, 对应 A/D 转换器的分辨率 1LSB 的温度就变高。当 AVCC0 的外加电压小于设定电压时, 对应 A/D 转换器的分辨率 1LSB 的温度就变低。但是必须注意: 有可能超出范围。

PGAEN 位 (PGA 允许位)

此位选择温度传感器的 PGA 运行或者停止。必须在 TSEN 位为“1” (温度传感器运行) 的状态下写此位。如果在此位为“0”时写“1”, PGA 就运行, 同时产生 12 位 A/D 转换器的启动触发。如果 A/D 转换结束, 此位就自动变为“0”。有关与 A/D 转换器的联动, 请参照“图 34.3 温度传感器输出的 A/D 转换时序例子 (进行 2 次转换的情况)”。

TSEN 位 (温度传感器允许位)

此位选择温度传感器的运行或者停止。

必须在 PGAEN 位为“0” (PGA 停止) 的状态下写此位。如果让温度传感器运行, 就流入稳定电流。在不使用温度传感器时, 一旦停止温度传感器, 就能降低消耗电流。

34.3 温度传感器的使用方法

温度传感器输出随温度而变化的电压。通过 PGA 对此电压进行放大后使用 12 位 A/D 转换器进行数字转换，并且通过将转换值换算为温度来求 LSI 的外围温度。12 位 A/D 转换器通过让处于停止状态的 PGA 运行，产生启动触发，并且开始 A/D 转换。

34.3.1 使用前的准备

以下表示温度传感器的温度特性。温度传感器的输出电压和温度变化成正比关系，用以下表达式表示。

温度特性的表达式：

$$T=(V_s-V_1)/\text{Slope}+T_1$$

T: 测量温度 (°C)

V_s: 测量温度时的温度传感器的输出电压 (V)

T₁: 第 1 点的试验测量时的温度 (°C)

V₁: 第 1 点的试验测量时的温度传感器的输出电压 (V)

T₂: 第 2 点的试验测量时的温度 (°C)

V₂: 第 2 点的试验测量时的温度传感器的输出电压 (V)

Slope: 温度传感器的温度梯度 (V/°C) Slope=(V₂-V₁)/(T₂-T₁)

因为温度传感器自身有偏差，所以推荐对以下 2 点不同的温度进行试验测量。

首先，通过 12 位 A/D 转换器进行的试验测量求温度 T₁ 时的温度传感器的输出电压 V₁。

其次，通过 12 位 A/D 转换器进行的试验测量求和温度 T₁ 不同的温度 T₂ 时的温度传感器的输出电压 V₂。

从两者的测量结果求温度梯度 (Slope=(V₂-V₁)/(T₂-T₁))。

将此 Slope 代入温度特性的表达式，求温度特性 T=(V_s-V₁)/Slope+T₁。

通过使用“41.7 温度传感器特性”中记载的温度梯度以及通过 12 位 A/D 转换器进行的试验测量求温度 T₁ 时的温度传感器的输出电压 V₁，并且用以下表达式计算出测量温度。此测量温度精度低于 2 点测量方法。

$$T=(V_s-V_1)/\text{Slope}+T_1$$

34.3.2 12 位 A/D 转换器的设定

通过对温度传感器的输出进行 A/D 转换，温度传感器能检测到温度。为了对温度传感器的输出进行 A/D 转换，需要将 12 位 A/D 转换器的寄存器进行如下设定：

- **温度传感器输出的 A/D 转换对象选择**
必须将 ADEXICR.TSS 位置“1”，将温度传感器输出选择为 A/D 转换对象。此时，必须将 ADANSA 位、ADANSB 位和 ADEXICR.OCS 位都置“0”，取消转换对象。
- **单次扫描模式的设定**
将 ADCSR.ADCS[1:0] 位置“00”，选择单次扫描模式。不能设定单次扫描模式以外的模式。
- **12 位 A/D 转换器的采样状态数设定**
在对温度传感器的输出进行 A/D 转换时，需要“41.7 温度传感器特性”中所记载采样时间。必须根据 A/D 转换器的转换时钟 (PCLKD) 频率来设定采样状态数。采样时间为 70 μ s (标准值) 时的设定例子如表 34.2 所示。

表 34.2 A/D 转换器的转换时钟 (PCLKD) 的频率和 ADSSTRT 寄存器的设定例子

A/D 转换器的转换时钟 (PCLKD) 的频率 (MHz)	ADSSTRT 寄存器的设定值
1	70
2	140
3	210

- **A/D 转换开始触发的设定**
为了对温度传感器的输出进行 A/D 转换，必须先将 ADCSR.TRGE 位置“1”，将 EXTRG 位置“0”，再将温度传感器的触发设定为有效，然后将 ADSTRGR.TRSA[3:0] 位置“0101”，最后选择温度传感器作为 A/D 转换开始触发。不能选择温度传感器以外的 A/D 转换开始触发。软件触发也不能使用。

34.3.3 温度传感器的 A/D 转换结果

如果温度传感器输出的 A/D 转换结束，就将准换结果保存到 ADTSDR 寄存器。

34.3.4 温度传感器的使用步骤

温度传感器的使用步骤流程如图 34.2 所示。

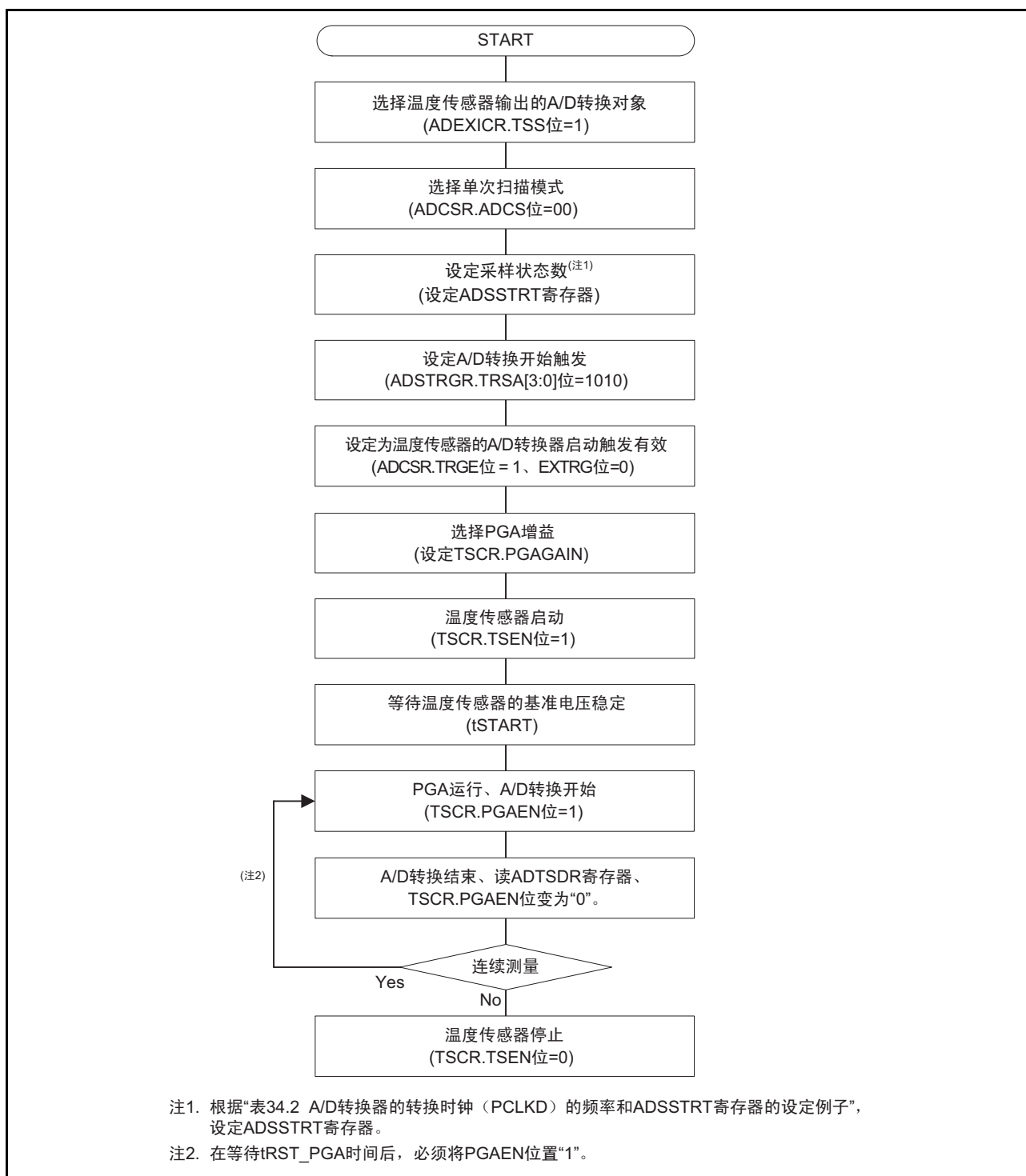


图 34.2 温度传感器的使用步骤流程

34.3.5 温度传感器输出的 A/D 转换时序

如果在选择温度传感器输出作为 12 位 A/D 转换器的启动触发的状态下将 TSCR.PGAEN 位从“0”置“1”，并且使温度传感器运行，就产生 12 位 A/D 传感器的启动触发，而且开始 A/D 转换。

温度传感器输出的 A/D 转换时序如图 34.3 所示。

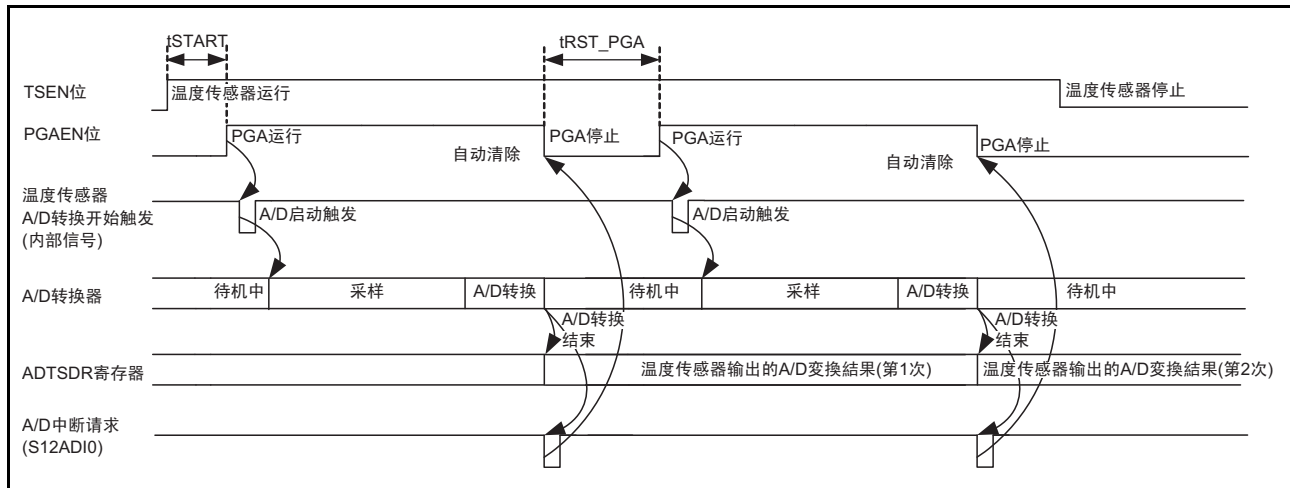


图 34.3 温度传感器输出的 A/D 转换时序例子 (进行 2 次转换的情况)

34.4 使用时的注意事项

34.4.1 模块停止功能的设定

能通过模块停止控制寄存器 B (MSTPCRB) 设定禁止或者允许温度传感器的运行，初始值为停止温度传感器的运行。通过解除模块停止状态，使寄存器变为可存取的状态。详细内容请参照“11. 低功耗功能”。

35. 比较器 A (CMPA)

比较器 A 比较基准输入电压和模拟输入电压，并且具有比较器 A1 和比较器 A2 两个独立的比较器。但是，比较器 A1、比较器 A2 与电压监视 1、电压监视 2 共用电压检测电路。能选择比较器 A1、比较器 A2 或者电压监视 1、电压监视 2 使用电压检测电路。

35.1 概要

能通过软件读基准输入电压和模拟输入电压的比较结果。还能选择 CVREFA 引脚的输入电压作为基准输入电压。另外，能使用比较器 A1 中断和比较器 A2 中断。

比较器 A 的规格、框图和引脚结构分别如表 35.1、图 35.1 和表 35.2 所示。

表 35.1 比较器 A 的规格

项目		比较器 A1	比较器 A2
模拟输入电压		CMPA1 引脚的输入电压	CMPA2 引脚的输入电压
基准输入电压		CVREFA 引脚的输入电压	
比较对象		比较模拟输入电压上升或者下降过程中是否经过基准输入电压。	
比较结果的监视		LVD1SR.LVD1MON 位	LVD2SR.LVD2MON 位
		表示模拟输入电压高于或者低于基准输入电压。	
中断请求		比较器 A1 中断 (能选择非屏蔽中断或者可屏蔽中断)	比较器 A2 中断 (能选择非屏蔽中断或者可屏蔽中断)
中断请求的发生时序		CMPA1 引脚的输入电压 <ul style="list-style-type: none"> 在上升过程中经过 CVREFA 引脚的基准输入电压时 在下降过程中经过 CVREFA 引脚的基准输入电压时 在上升和下降过程中都经过 CVREFA 引脚的基准输入电压时 	CMPA2 引脚的输入电压 <ul style="list-style-type: none"> 在上升过程中经过 CVREFA 引脚的基准输入电压时 在下降过程中经过 CVREFA 引脚的基准输入电压时 在上升和下降过程中都经过 CVREFA 引脚的基准输入电压时
ELC 的事件发生时序		CMPA1 引脚的输入电压 <ul style="list-style-type: none"> 在上升过程中经过 CVREFA 引脚的基准输入电压时 在下降过程中经过 CVREFA 引脚的基准输入电压时 在上升和下降过程中都经过 CVREFA 引脚的基准输入电压时 	CMPA2 引脚的输入电压 <ul style="list-style-type: none"> 在上升过程中经过 CVREFA 引脚的基准输入电压时 在下降过程中经过 CVREFA 引脚的基准输入电压时 在上升和下降过程中都经过 CVREFA 引脚的基准输入电压时
数字滤波器	有效 / 无效的转换	有	
	采样时间	(LOCO 的 n 分频) × 2 n: 1、2、4、8	
比较结果的输出		能通过事件链接控制器 (ELC) 从端口输出比较结果。	能通过事件链接控制器 (ELC) 从端口输出比较结果。

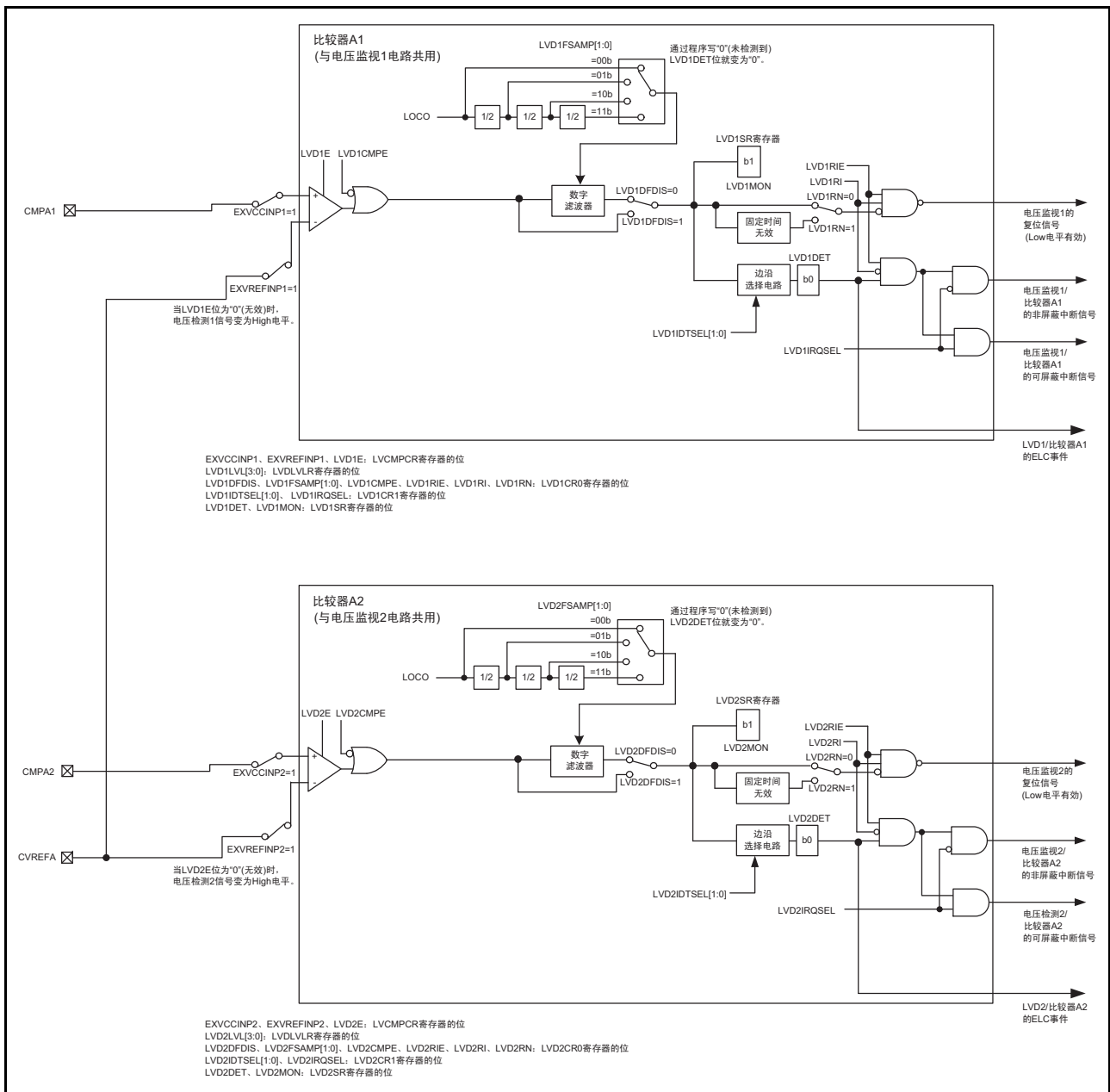


图 35.1 比较器 A 的框图

表 35.2 比较器 A 的引脚结构

引脚名	输入 / 输出	功能
CMPA1	输入	用于比较器 A1 的模拟引脚
CMPA2	输入	用于比较器 A2 的模拟引脚
CVREFA	输入	用于比较器的基准电压引脚

35.2 寄存器说明

35.2.1 电压监视 1 电路 / 比较器 A1 的控制寄存器 1 (LVD1CR1)

地址 0008 00E0h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	LVD1IRQSEL	LVD1IDTSEL[1:0]	
0	0	0	0	0	0	0	1

复位后的值

位	符号	位名	功能	R/W
b1-b0	LVD1IDTSEL [1:0]	电压监视 1/ 比较器 A1 的中断和 ELC 事件的发生条件选择位	b1 b0 0 0: CMPA1 \geq CVREFA 0 1: CMPA1 < CVREFA 1 0: 检测到下降和上升时 1 1: 不能设定	R/W
b2	LVD1IRQSEL	电压监视 1/ 比较器 A1 的中断种类选择位	0: 非屏蔽中断 1: 可屏蔽中断	R/W
b7-b3	—	保留位	读写值都为“0”。	R/W

35.2.2 电压监视 1 电路 / 比较器 A1 的状态寄存器 (LVD1SR)

地址 0008 00E1h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	LVD1MON	LVD1DET
0	0	0	0	0	0	1	0

复位后的值

位	符号	位名	功能	R/W
b0	LVD1DET	电压监视 1/ 比较器 A1 的电压变化检测标志	0: 未检测到 1: 检测到比较器 A1 的电压发生变化	R/W (注 1)
b1	LVD1MON	电压监视 1/ 比较器 A1 的信号监视标志	0: CMPA1 < CVREFA 1: CMPA1 \geq CVREFA	R
b7-b2	—	保留位	读写值都为“0”。	R/W

注 1. 只能写“0”。从写“0”后到被反映为 LVD1DET 位的读取值为止，需要 2 个系统时钟周期。

LVD1DET 标志 (电压监视 1/ 比较器 A1 的电压变化检测标志)

LVD1DET 标志在 LVCMPCR.LVD1E 位为“1” (电压检测 1 电路有效) 并且 LVD1CR0.LVD1CMPE 位为“1” (允许输出电压监视 1 电路的比较结果) 时有效。

如果要将 LVD1DET 标志置“0”，必须先将 LVD1CR0.LVD1RIE 位置“0” (禁止)。如果要重新将 LVD1CR0.LVD1RIE 位置“1” (允许)，必须等待 2 个 PCLKB 周期。

能通过读 PCLKB 定义的 I/O 寄存器，确保存取周期数至少为 2 个 PCLKB 周期的等待时间。

LVD1MON 标志 (电压监视 1/ 比较器 A1 的信号监视标志)

LVD1MON 标志在 LVCMPCR.LVD1E 位为“1” (电压检测 1 电路有效) 并且 LVD1CR0.LVD1CMPE 位为“1” (允许输出电压监视 1 电路的比较结果) 时有效。

35.2.3 电压监视 2 电路 / 比较器 A2 的控制寄存器 1 (LVD2CR1)

地址 0008 00E2h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	LVD2IRQSEL	LVD2IDTSEL[1:0]	
0	0	0	0	0	0	0	1

复位后的值

位	符号	位名	功能	R/W
b1-b0	LVD2IDTSEL [1:0]	电压监视 2/ 比较器 A2 的中断和 ELC 事件的发生条件选择位	b1 b0 0 0: CMPA2 ≥ CVREFA 0 1: CMPA2 < CVREFA 1 0: 检测到下降和上升时 1 1: 不能设定	R/W
b2	LVD2IRQSEL	电压监视 2/ 比较器 A2 的中断种类选择位	0: 非屏蔽中断 1: 可屏蔽中断	R/W
b7-b3	—	保留位	读写值都为“0”。	R/W

35.2.4 电压监视 2 电路 / 比较器 A2 的状态寄存器 (LVD2SR)

地址 0008 00E3h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	LVD2MON	LVD2DET
0	0	0	0	0	0	1	0

复位后的值

位	符号	位名	功能	R/W
b0	LVD2DET	电压监视 2/ 比较器 A2 的电压变化检测标志	0: 未检测到 1: 检测到比较器 A2 的电压发生变化	R/W (注 1)
b1	LVD2MON	电压监视 2/ 比较器 A2 的信号监视标志	0: CMPA2 < CVREFA 1: CMPA2 ≥ CVREFA	R
b7-b2	—	保留位	读写值都为“0”。	R/W

注 1. 只能写“0”。从写“0”后到被反映为 LVD2DET 位的读取值为止，需要 2 个系统时钟周期。

LVD2DET 标志 (电压监视 2/ 比较器 A2 的电压变化检测标志)

LVD2DET 标志在 LVCMPCR.LVD2E 位为“1” (电压检测 2 电路有效) 并且 LVD2CR0.LVD2CMPE 位为“1” (允许输出电压监视 2 电路的比较结果) 时有效。

如果要将 LVD2DET 标志置“0”，必须先将 LVD2CR0.LVD2RIE 位置“0” (禁止)。如果要重新将 LVD2CR0.LVD2RIE 位置“1” (允许)，必须等待 2 个 PCLKB 周期。

能通过读 PCLKB 定义的 I/O 寄存器，确保存取周期数至少为 2 个 PCLKB 周期的等待时间。

LVD2MON 标志 (电压监视 2/ 比较器 A2 的信号监视标志)

LVD2MON 标志在 LVCMPCR.LVD2E 位为“1” (电压检测 2 电路有效) 并且 LVD2CR0.LVD2CMPE 位为“1” (允许输出电压监视 2 电路的比较结果) 时有效。

35.2.5 电压监视电路 / 比较器 A 的控制寄存器 (LVCMPCR)

地址 0008 C297h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	LVD2E	LVD1E	—	EXVCCIN P2	EXVREFI NP2	EXVCCIN P1	EXVREFI NP1
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	EXVREFINP1	比较器 A1 的基准电压外部输入选择位	1: CVREFA 引脚的输入电压 在比较器 A1 中使用时, 必须将此位置“1”。	R/W
b1	EXVCCINP1	比较器 A1 的比较电压外部输入选择位	1: CMPA1 引脚的输入电压 在比较器 A1 中使用时, 必须将此位置“1”。	R/W
b2	EXVREFINP2	比较器 A2 的基准电压外部输入选择位	1: CVREFA 引脚的输入电压 在比较器 A2 中使用时, 必须将此位置“1”。	R/W
b3	EXVCCINP2	比较器 A2 的比较电压外部输入选择位	1: CMPA2 引脚的输入电压 在比较器 A2 中使用时, 必须将此位置“1”。	R/W
b4	—	保留位	读写值都为“0”。	R/W
b5	LVD1E	电压检测 1/ 比较器 A1 的允许位	0: 电压检测 1/ 比较器 A1 电路无效 1: 电压检测 1/ 比较器 A1 电路有效	R/W
b6	LVD2E	电压检测 2/ 比较器 A2 的允许位	0: 电压检测 2/ 比较器 A2 电路无效 1: 电压检测 2/ 比较器 A2 电路有效	R/W
b7	—	保留位	读写值都为“0”。	R/W

只有在 LVD1E 位和 LVD2E 位都为“0”（电压检测电路无效）时才能更改 EXVREFINP1 位、EXVCCINP1 位、EXVREFINP2 位和 EXVCCINP2 位。

LVD1E 位（电压检测 1/ 比较器 A1 的允许位）

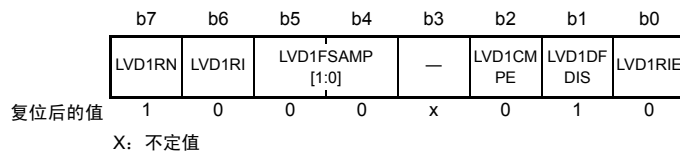
在使用电压检测 1/ 比较器 A1 的中断 / 复位或者使用 LVD1SR.LVD1MON 位时, 必须将 LVD1E 位置“1”。如果将 LVD1E 位从“0”置“1”, 在经过 $t_d(E-A)$ 时间后, 电压检测 1/ 比较器 A1 电路就运行。另外, 如果在深度软件待机模式中使用电压检测 1/ 比较器 A1 电路, 就不能将 DPSBYCR.DEEPCUT1 位置“1”。

LVD2E 位（电压检测 2/ 比较器 A2 的允许位）

在使用电压检测 2/ 比较器 A2 的中断 / 复位或者使用 LVD2SR.LVD2MON 位时, 必须将 LVD2E 位置“1”。如果将 LVD2E 位从“0”置“1”, 在经过 $t_d(E-A)$ 时间后, 电压检测 2/ 比较器 A2 电路就运行。另外, 如果在深度软件待机模式中使用电压检测 2/ 比较器 A2 电路, 就不能将 DPSBYCR.DEEPCUT1 位置“1”。

35.2.6 电压监视 1 电路 / 比较器 A1 的控制寄存器 0 (LVD1CR0)

地址 0008 C29Ah



位	符号	位名	功能	R/W
b0	LVD1RIE	电压监视 1/ 比较器 A1 的中断 / 复位允许位	0: 禁止 1: 允许	R/W
b1	LVD1DFDIS	电压监视 1/ 比较器 A1 的数字滤波器无效模式选择位	0: 数字滤波器有效 1: 数字滤波器无效	R/W
b2	LVD1CMPE	电压监视 1 电路 / 比较器 A1 的比较结果输出允许位	0: 禁止输出比较器 A1 电路的比较结果 1: 允许输出比较器 A1 电路的比较结果	R/W
b3	—	保留位	读取值为不定值, 只能写“0”。	R/W
b5-b4	LVD1FSAMP [1:0]	采样时钟选择位	b5 b4 0 0: LOCO 的 1 分频 0 1: LOCO 的 2 分频 1 0: LOCO 的 4 分频 1 1: LOCO 的 8 分频	R/W
b6	LVD1RI	电压监视 1 电路 / 比较器 A1 的模式选择位	0: 在 CMPA1 经过 CVREFA 时发生比较器 A1 中断 1: 在 CMPA1 经过 CVREFA 时发生比较器 A1 复位	R/W
b7	LVD1RN	电压监视 1/ 比较器 A1 的复位无效选择位	0: 从检测到 CMPA1 > CVREFA 开始经过一定时间 (tLVD1) 后无效 1: 从比较器 A1 复位有效开始经过一定时间 (tLVD1) 后无效	R/W

LVD1RIE 位 (电压监视 1/ 比较器 A1 的中断 / 复位允许位)

在闪存的编程 / 擦除过程中, 不能发生 LVD1 复位和 LVD1 非屏蔽中断。

LVD1DFDIS (电压监视 1/ 比较器 A1 的数字滤波器无效模式选择位)

如果要将 LVD1DFDIS 位置“0” (数字滤波器电路有效), 必须将 LOCOCR.LCSTP 位置“0” (LOCO 运行)。

如果要在软件待机模式或者深度软件待机模式中使用电压监视 1 电路, 就必须将 LVD1DFDIS 位置“1” (数字滤波器电路无效)。

LVD1FSAMP[1:0] 位 (采样时钟选择位)

能在 LVD1DFDIS 位为“1” (数字滤波器电路无效) 时改写 LVD1FSAMP[1:0] 位。在 LVD1DFDIS 位为“0” (数字滤波器电路有效) 时不能改写 LVD1FSAMP[1:0] 位。

LVD1RI 位 (电压监视 1 电路 / 比较器 A1 的模式选择位)

在 LVD1RI 位为“1” (选择电压监视 1 复位) 或者 LVD2CR0.LVD2RI 位为“1” (选择电压监视 2 复位) 时, 不能转移到深度软件待机模式, 但是能转移到软件待机模式。如果要转移到深度软件待机模式, 就必须将 LVD1RI 位置“0” (选择电压监视 1 中断), 并且将 LVD2CR0.LVD2RI 位置“0” (选择电压监视 2 中断)。

LVD1RN 位 (电压监视 1/ 比较器 A1 的复位无效选择位)

如果要将 LVD1RN 位置“1” (从 LVD1 复位有效开始经过一定时间后无效), 就必须将 LOCOCR.LCSTP 位置“0” (LOCO 运行)。另外, 如果要转移到软件待机模式或者深度软件待机模式, 就只能将 LVD1RN 位置“0” (从检测到 CMPA1 > CVREFA 开始经过一定时间后无效), 不能将 LVD1RN 位置“1” (从 LVD1 复位有效开始经过一定时间后无效)。

35.2.7 电压监视 2 电路 / 比较器 A2 的控制寄存器 0 (LVD2CR0)

地址 0008 C29Bh

b7	b6	b5	b4	b3	b2	b1	b0
LVD2RN	LVD2RI	LVD2FSAMP [1:0]	—	LVD2CM PE	LVD2DF DIS	LVD2RIE	
1	0	0 0	x	0	1	0	

复位后的值
X: 不定值

位	符号	位名	功能	R/W
b0	LVD2RIE	电压监视 2/ 比较器 A2 的中断 / 复位允许位	0: 禁止 1: 允许	R/W
b1	LVD2DFDIS	电压监视 2/ 比较器 A2 的数字滤波器无效模式选择位	0: 数字滤波器有效 1: 数字滤波器无效	R/W
b2	LVD2CMPE	电压监视 2 电路 / 比较器 A2 的比较结果输出允许位	0: 禁止输出比较器 A2 电路的比较结果 1: 允许输出比较器 A2 电路的比较结果	R/W
b3	—	保留位	读取值为不定值, 只能写“0”。	R/W
b5-b4	LVD2FSAMP [1:0]	采样时钟选择位	b5 b4 0 0: LOCO 的 1 分频 0 1: LOCO 的 2 分频 1 0: LOCO 的 4 分频 1 1: LOCO 的 8 分频	R/W
b6	LVD2RI	电压监视 2 电路 / 比较器 A2 的模式选择位	0: 在 CMPA2 经过 CVREFA 时发生比较器 A2 中断 1: 在 CMPA2 经过 CVREFA 时发生比较器 A2 复位	R/W
b7	LVD2RN	电压监视 2/ 比较器 A2 的复位无效选择位	0: 从检测到 CMPA2 > CVREFA 开始经过一定时间 (tLVD2) 后无效 1: 从比较器 A2 复位有效开始经过一定时间 (tLVD2) 后无效	R/W

LVD2RIE 位 (电压监视 2/ 比较器 A2 的中断 / 复位允许位)

在闪存的编程 / 擦除过程中, 不能发生 LVD2 复位和 LVD2 非屏蔽中断。

LVD2DFDIS (电压监视 2/ 比较器 A2 的数字滤波器无效模式选择位)

如果要将 LVD2DFDIS 位置“0” (数字滤波器电路有效), 必须将 LOCOCR.LCSTP 位置“0” (LOCO 运行)。

如果要在软件待机模式或者深度软件待机模式中使用电压监视 2 电路, 就必须将 LVD2DFDIS 位置“1” (数字滤波器电路无效)。

LVD2FSAMP[1:0] 位 (采样时钟选择位)

能在 LVD2DFDIS 位为“1” (数字滤波器电路无效) 时改写 LVD2FSAMP[1:0] 位。在 LVD2DFDIS 位为“0” (数字滤波器电路有效) 时不能改写 LVD2FSAMP[1:0] 位。

LVD2RI 位 (电压监视 2 电路 / 比较器 A2 的模式选择位)

在 LVD2RI 位为“1” (选择电压监视 2 复位) 或者 LVD1CR0.LVD1RI 位为“1” (选择电压监视 1 复位) 时, 不能转移到深度软件待机模式, 但是能转移到软件待机模式。如果要转移到深度软件待机模式, 就必须将 LVD2RI 位置“0” (选择电压监视 2 中断), 并且将 LVD1CR0.LVD1RI 位置“0” (选择电压监视 1 中断)。

LVD2RN 位 (电压监视 2/ 比较器 A2 的复位无效选择位)

如果要将 LVD2RN 位置“1” (从 LVD2 复位有效开始经过一定时间后无效), 就必须将 LOCOCR.LCSTP 位置“0” (LOCO 运行)。另外, 如果要转移到软件待机模式或者深度软件待机模式, 就只能将 LVD2RN 位置“0” (从检测到 CMPA2 > CVREFA 开始经过一定时间后无效), 不能将 LVD2RN 位置“1” (从 LVD2 复位有效开始经过一定时间后无效)。

35.3 比较结果的监视

35.3.1 比较器 A1 的监视

在进行以下设定后，能通过 LVD1SR.LVD1MON 位监视比较器 A1 的比较结果。

1. 将 LVCMPCR.EXVREFINP1 位置“1” (CVREFA 引脚的输入电压)，
将 LVCMPCR.EXVCCINP1 位置“1” (CMPA1 引脚的输入电压)。
2. 通过 LVD1CR0.LVD1FSAMP[1:0] 位选择数字滤波器的采样时钟。
3. 将 LVCMPCR.LVD1E 位置“1” (比较器 A1 电路有效)。
4. 在等待 $t_d(E-A)$ 时间后，将 LVD1CR0.LVD1CMPE 位置“1” (允许输出比较器 A1 电路的比较结果)。
5. 等待至少 1 个 LOCO 周期。
6. 将 LVD1CR0.LVD1DFDIS 位置“0” (数字滤波器有效)。
7. 等待至少 $2n+3$ 个 LOCO 周期 ($n=1、2、4、8$: 数字滤波器的采样时钟=LOCO 的 n 分频)。

注. 在数字滤波器无效的情况下使用时，不需要步骤 2、步骤 5 ~ 步骤 7。

35.3.2 比较器 A2 的监视

在进行以下设定后，能通过 LVD2SR.LVD2MON 位监视比较器 A2 的比较结果。

1. 将 LVCMPCR.EXVREFINP2 位置“1” (CVREFA 引脚的输入电压)，
将 LVCMPCR.EXVCCINP2 位置“1” (CMPA2 引脚的输入电压)。
2. 通过 LVD2CR0.LVD2FSAMP[1:0] 位选择数字滤波器的采样时钟。
3. 将 LVCMPCR.LVD2E 位置“1” (比较器 A2 电路有效)。
4. 在等待 $t_d(E-A)$ 时间后，将 LVD2CR0.LVD2CMPE 位置“1” (允许输出比较器 A2 电路的比较结果)。
5. 等待至少 1 个 LOCO 周期。
6. 将 LVD2CR0.LVD2DFDIS 位置“0” (数字滤波器有效)。
7. 等待至少 $2n+3$ 个 LOCO 周期 ($n=1、2、4、8$: 数字滤波器的采样时钟=LOCO 的 n 分频)。

注. 在数字滤波器无效的情况下使用时，不需要步骤 2、步骤 5 ~ 步骤 7。

35.4 运行说明

比较器 A1 和比较器 A2 能各自独立运行。

比较器 A1 和比较器 A2 能通过软件读基准输入电压和模拟输入电压的比较结果，还能将 CVREFA 引脚的输入电压用作基准输入电压，以及能使用比较器 A1 中断和比较器 A2 中断，并且能将比较器 A1 中断和比较器 A2 中断分别选择为非屏蔽中断或者可屏蔽中断。

35.4.1 比较器 A1

比较器 A1 中断 /ELC 的相关位的运行设定步骤如表 35.3 所示，比较器 A1 中断 /ELC 的相关位的停止设定步骤如表 35.4 所示，比较器 A1 的运行例子如图 35.2 所示。

表 35.3 比较器 A1 中断 /ELC 的相关位的运行设定步骤

步骤	使用数字滤波器的情况	不使用数字滤波器的情况
1 (注2)	将 LVCMPCR.EXVREFINP1 位置“1” (CVREFA 引脚的输入电压)、EXVCCINP1 位置“1” (CMPA1 引脚的输入电压)。	
2 (注1)	通过 LVD1CR0.LVD1FSAMP[1:0] 位选择数字滤波器的采样时钟。	—
3 (注1、注2)	将 LVD1CR0.LVD1RI 位置“0” (比较器 A1 中断)。	
4	通过 LVD1CR1.LVD1IDTSEL[1:0] 位选择中断请求的时序，通过 LVD1CR1.LVD1IRQSEL 位选择中断的种类。	
5 (注2)	将 LVCMPCR.LVD1E 位置“1” (比较器 A1 电路有效)。	
6 (注2)	等待 td(E-A)。	
7	将 LVD1CR0.LVD1CMPE 位置“1” (允许输出比较器 A1 电路的比较结果)。	
8	等待至少 1 个 LOCO 周期。	—
9	将 LVD1CR0.LVD1DFDIS 位置“0” (数字滤波器有效)。	—
10	等待至少 $2n+3$ 个 LOCO 周期 ($n=1、2、4、8$: 数字滤波器的采样时钟 = LOCO 的 n 分频)。	—
11	将 LVD1SR.LVD1DET 位置“0”	
12	将 LVD1CR0.LVD1RIE 位置“1” (允许比较器 A1 中断)。与此位的设定无关，向 ELC 输出事件。	

注 1. 能用 1 条指令同时执行步骤 2 和步骤 3。

注 2. 在以下的任何一种情况下，不需要步骤 1、3、5、6。

在通过设定为比较器 A1 中断 (LVD1CR0.LVD1RI=0) 而运行时，如果

- 在停止后只更改 LVD1CR0.LVD1DFDIS 位、LVD1CR0.LVD1FSAMP 位或者 LVD1CR1.LVD1IRQSEL 位、LVD1CR1.LVD1IDTSEL 位的设定而重新运行时
- 在停止后不更改比较器 A1 电路的相关设定而重新运行时

表 35.4 比较器 A1 中断 /ELC 的相关位的停止设定步骤

步骤	
1	将 LVD1CR0.LVD1RIE 位置“0” (禁止比较器 A1 中断)。
2	将 LVD1CR0.LVD1CMPE 位置“0” (禁止输出比较器 A1 电路的比较结果)。
3 (注1)	将 LVCMPCR.LVD1E 置“0” (比较器 A1 电路无效)。
4	更改除 LVCMPCR.LVD1E 位、LVD1CR0.LVD1RIE 位、LVD1CR0.LVD1CMPE 位以外的电压检测电路相关寄存器的设定。

注 1. 在以下的任何一种情况下，不需要步骤 3。

在通过设定为比较器 A1 中断 (LVD1CR0.LVD1RI=0) 而运行时，如果

- 在停止后只更改 LVD1CR0.LVD1DFDIS 位、LVD1CR0.LVD1FSAMP 位或者 LVD1CR1.LVD1IRQSEL 位、LVD1CR1.LVD1IDTSEL 位的设定而重新运行时
- 在停止后不更改比较器 A1 电路的相关设定而重新运行时

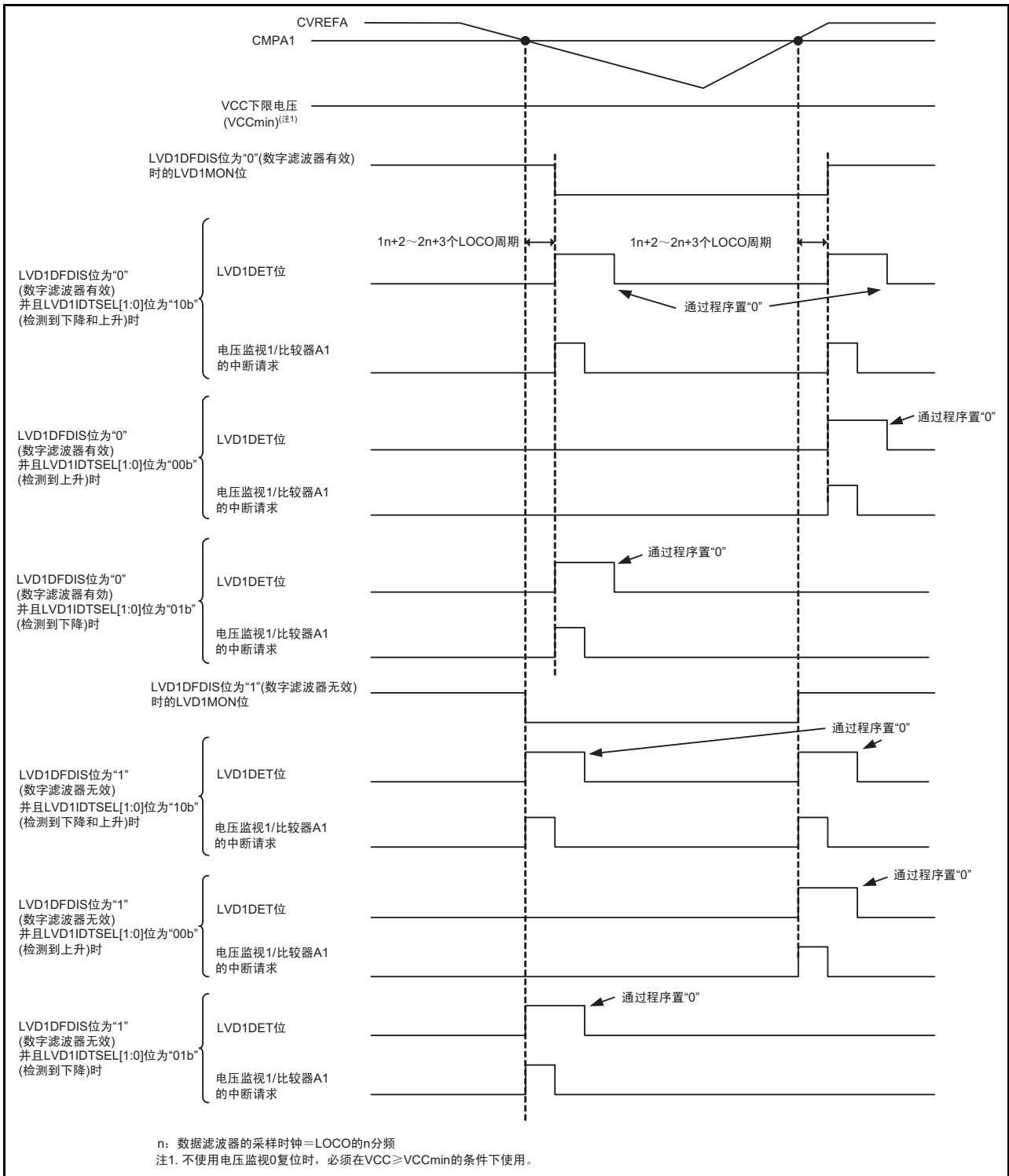


图 35.2 比较器 A1 的运行例子

35.4.2 比较器 A2

比较器 A2 中断 /ELC 的相关位的运行设定步骤如表 35.5 所示，比较器 A2 中断 /ELC 的相关位的停止设定步骤如表 35.6 所示，比较器 A2 的运行例子如图 35.3 所示。

表 35.5 比较器 A2 中断 /ELC 的相关位的运行设定步骤

步骤	使用数字滤波器的情况	不使用数字滤波器的情况
1 (注2)	将 LVCMPCR.EXVREFINP2 位置“1” (CVREFA 引脚的输入电压)、EXVCCINP2 位置“1” (CMPA2 引脚的输入电压)。	
2 (注1)	通过 LVD2CR0.LVD2FSAMP[1:0] 位选择数字滤波器的采样时钟。	—
3 (注1、注2)	将 LVD2CR0.LVD2RI 位置“0” (比较器 A2 中断)。	
4	通过 LVD2CR1.LVD2IDTSEL[1:0] 位选择中断请求的时序，通过 LVD2CR1.LVD2IRQSEL 位选择中断的种类。	
5 (注2)	将 LVCMPCR.LVD2E 位置“1” (比较器 A2 电路有效)。	
6 (注2)	等待 td(E-A)。	
7	将 LVD2CR0.LVD2CMPE 位置“1” (允许输出比较器 A2 电路的比较结果)。	
8	等待至少 1 个 LOCO 周期。	—
9	将 LVD2CR0.LVD2DFDIS 位置“0” (数字滤波器有效)。	—
10	等待至少 $2n+3$ 个 LOCO 周期 ($n=1、2、4、8$: 数字滤波器的采样时钟 =LOCO 的 n 分频)。	—
11	将 LVD2SR.LVD2DET 位置“0”。	
12	将 LVD2CR0.LVD2RIE 位置“1” (允许比较器 A2 中断)，与此位的设定无关，向 ELC 输出事件。	

注 1. 能用 1 条指令同时执行步骤 2 和步骤 3。

注 2. 在以下的任何一种情况下，不需要步骤 1、3、5、6。

在通过设定为比较器 A2 中断 (LVD2CR0.LVD2RI=0) 而运行时，如果

- 在停止后只更改 LVD2CR0.LVD2DFDIS 位、LVD2CR0.LVD2FSAMP 位或者 LVD2CR1.LVD2IRQSEL 位、LVD2CR1.LVD2IDTSEL 位的设定而重新运行时
- 在停止后不更改比较器 A2 电路的相关设定而重新运行时

表 35.6 比较器 A2 中断 /ELC 的相关位的停止设定步骤

步骤	
1	将 LVD2CR0.LVD2RIE 位置“0” (禁止比较器 A2 中断)。
2	将 LVD2CR0.LVD2CMPE 位置“0” (禁止输出比较器 A2 电路的比较结果)。
3 (注1)	将 LVCMPCR.LVD2E 置“0” (比较器 A2 电路无效)。
4	更改除 LVCMPCR.LVD2E 位、LVD2CR0.LVD2RIE 位、LVD2CR0.LVD2CMPE 位以外的电压检测电路相关寄存器的设定。

注 1. 在以下的任何一种情况下，不需要步骤 3。

在通过设定为比较器 A2 中断 (LVD2CR0.LVD2RI=0) 而运行时，如果

- 在停止后只更改 LVD2CR0.LVD2DFDIS 位、LVD2CR0.LVD2FSAMP 位或者 LVD2CR1.LVD2IRQSEL 位、LVD2CR1.LVD2IDTSEL 位的设定而重新运行时
- 在停止后不更改比较器 A1 电路的相关设定而重新运行时

35.5 比较器 A1 中断和比较器 A2 中断

在以下时序产生比较器 A1 和比较器 A2 的中断请求。

1. 当 CMPA1 引脚的输入电压经过 CVREFA 引脚的基准输入电压时
2. 当 CMPA2 引脚的输入电压经过 CVREFA 引脚的基准输入电压时

能通过 LVDiCR1.LVDiDTSEL[1:0] 位从“在上升过程中经过时”、“在下降过程中经过时”、和“在上升和下降过程中都经过时”中选择中断请求的产生时序。

从非屏蔽中断和可屏蔽中断中选择各中断的种类。中断的详细内容请参照“14. 中断控制器 (ICUb) ”。

35.5.1 非屏蔽中断

如果将 LVDiCR1.LVDiIRQSEL 位置“0”，比较器 Ai 中断就用作非屏蔽中断。当产生所选中断请求的时序时，LVDiSR.LVDiMON 位变为“1”，此时，如果中断控制器 (ICU) 的 NMIER.LVDiEN 位为“1”，就产生比较器 Ai 的非屏蔽中断请求。

35.5.2 可屏蔽中断

如果将 LVDiCR1.LVDiIRQSEL 位置“1”，比较器 Ai 中断就用作可屏蔽中断。

如果在 LVDiCR0.LVDiRIE 位为“1”、LVDiCR0.LVDiDFDIS 位为“0”时产生所选中断请求的时序，LVDiSR.LVDiMON 位就变为“1”，此时，如果中断控制器 (ICU) 的 IER0B.IEN0 位和 IER0B.IEN1 位为“1”（允许中断），并且 IPR088.IPR[3:0] 位和 IPR089.IPR[3:0] 位的设定电平高于 CPU 的 PSW.IPL[3:0] 位的电平，IR088.IR 位和 IR089.IR 位就变为“1”（有中断请求），并且产生比较器 Ai 的可屏蔽中断请求。

有关 IEN0B、IR088、IR089、IPR088 和 IPR089 寄存器以及中断向量，请参照“14. 中断控制器 (ICUb) ”。

35.6 事件链接输出功能

比较器 A1 和比较器 A2 在以下时序时向事件链接控制器 (ELC) 输出事件，并且能使事先设定的模块运行。

1. 当 CMPA1 引脚的输入电压经过 CVREFA 引脚的基准输入电压时
2. 当 CMPA2 引脚的输入电压经过 CVREFA 引脚的基准输入电压时

和产生中断请求的条件相同，能通过 LVDiCR1.LVDiDTSEL[1:0] 位从“在上升过程中经过时”、“在下降过程中经过时”、和“在上升和下降过程中都经过时”中选择 ELC 的事件发生条件。

如果要将比较器 A 的事件链接输出功能设定为有效，就必须在将比较器 A 设定为有效后，将 ELC 侧的比较器 A 的事件链接功能设定为有效。另外，如果要停止比较器 A 的事件链接输出功能时，就必须在设定停止比较器 A 前，将 ELC 侧的比较器 A 的事件链接功能设定为无效。

35.7 中断处理和事件链接的关系

向事件链接控制器 (ELC) 输出事件和向中断控制器请求中断是相互独立的关系。因此，与 LVDiCR0.LVDiRIE 位、LVDiCR0.LVDiRI 位和 LVDiCR1.LVDiIRQSEL 位的设定无关，向 ELC 输出事件。

36. 比较器 B (CMPB)

比较器 B 比较基准输入电压和模拟输入电压，并且具有比较器 B0 和比较器 B1 两个独立的比较器。

36.1 概要

能通过软件读基准输入电压和模拟输入电压的比较结果。能将 CVREFBn (n=0、1) 引脚的输入电压用作基准输入电压。

比较器 B 的规格和框图分别如表 36.1 和图 36.1 所示，输入 / 输出引脚如表 36.2 所示。

表 36.1 比较器 B 的规格

项目	规格
模拟输入电压	CMPBn 引脚的输入电压 (n=0、1)
基准输入电压	CVREFBn 引脚的输入电压 (n=0、1)
比较结果	读 CPBFLG.CPBiOUT 位 (n=0、1)
中断请求的发生时序	当比较器 B0 的比较结果发生变化时 当比较器 B1 的比较结果发生变化时
ELC 的事件发生时序	当比较器 B0 的比较结果发生变化时 当比较器 B0 或者 B1 的比较结果发生变化时
选择功能	数字滤波器功能 能选择数字滤波器的有无和采样频率
低功耗功能	能设定为功能停止状态

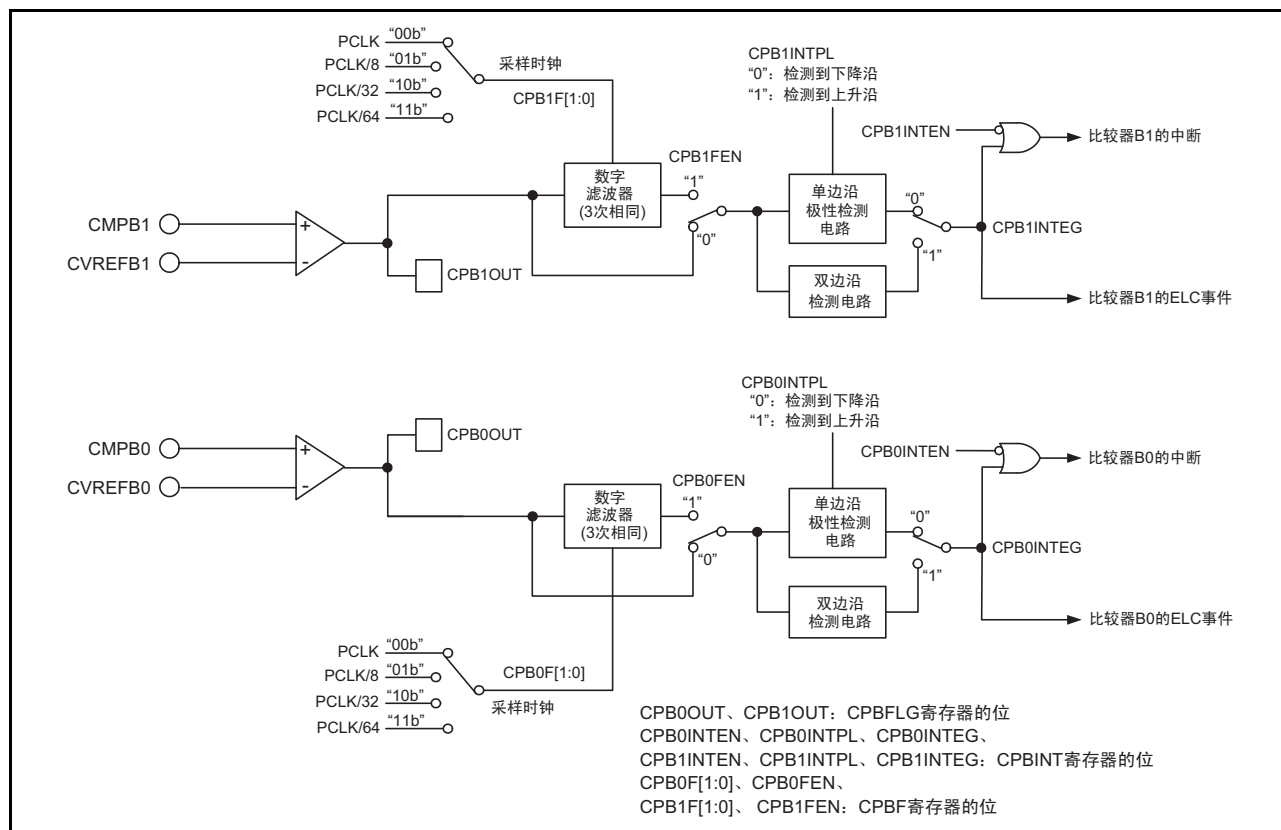


图 36.1 比较器 B 的框图

表 36.2 输入 / 输出引脚

引脚名	输入 / 输出	功能
CMPB0	输入	用于比较器 B0 的模拟引脚
CVREFB0	输入	用于比较器 B0 的基准电压引脚
CMPB1	输入	用于比较器 B1 的模拟引脚
CVREFB1	输入	用于比较器 B1 的基准电压引脚

36.2 寄存器说明

36.2.1 比较器 B 控制寄存器 1 (CPBCNT1)

地址 0008 C580h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	CPB1INI	—	—	—	CPB0INI
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	CPB0INI	比较器 B0 允许位	0: 禁止 1: 允许	R/W
b3-b1	—	保留位	读写值都为“0”。	R/W
b4	CPB1INI	比较器 B1 允许位	0: 禁止 1: 允许	R/W
b7-b5	—	保留位	读写值都为“0”。	R/W

36.2.2 比较器 B 标志寄存器 (CPBFLG)

地址 0008 C582h

b7	b6	b5	b4	b3	b2	b1	b0
CPB1OUT	—	—	—	CPB0OUT	—	—	—
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b2-b0	—	保留位	读写值都为“0”。	R/W
b3	CPB0OUT	比较器 B0 监视标志	0: CMPB0 < CVREFB0 1: CMPB0 > CVREFB0	R
b6-b4	—	保留位	读写值都为“0”。	R/W
b7	CPB1OUT	比较器 B1 监视标志	0: CMPB1 < CVREFB1 1: CMPB1 > CVREFB1	R

36.2.3 比较器 B 中断控制寄存器 (CPBINT)

地址 0008 C583h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	CPB1INT PL	CPB1INT EG	CPB1INT EN	—	CPB0INT PL	CPB0INT EG	CPB0INT EN
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	CPB0INTEN	比较器 B0 中断允许位	0: 禁止 1: 允许	R/W
b1	CPB0INTEG	比较器 B0 的中断 /ELC 边沿选择位 (注 1)	0: 单边沿 1: 双边沿	R/W
b2	CPB0INTPL	比较器 B0 的中断 /ELC 边沿极性选择位 (注 2)	0: 下降沿 1: 上升沿	R/W
b3	—	保留位	读写值都为“0”。	R/W
b4	CPB1INTEN	比较器 B1 中断允许位	0: 禁止 1: 允许	R/W
b5	CPB1INTEG	比较器 B1 的中断 /ELC 边沿选择位 (注 1)	0: 单边沿 1: 双边沿	R/W
b6	CPB1INTPL	比较器 B1 的中断 /ELC 边沿极性选择位 (注 2)	0: 下降沿 1: 上升沿	R/W
b7	—	保留位	读写值都为“0”。	R/W

注 1. 如果更改 CPB0INTPL 位, IR058.IR 位就有可能变为“1”(有中断请求), 如果更改 CPB1INTPL 位, IR059.IR 位就有可能变为“1”(有中断请求)。请参照“14. 中断控制器 (ICUb)”。

注 2. 只有在 CPBiINTEG 位为“0”(给比较器中断边沿选择单边沿)时 CPBiINTPL 位有效。

36.2.4 比较器 B 的滤波器选择寄存器 (CPBF)

地址 0008 C584h

b7	b6	b5	b4	b3	b2	b1	b0
CPB1F[1:0]	—	CPB1F EN	CPB0F[1:0]	—	CPB0F EN	—	CPB0F EN
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	CPB0FEN	比较器 B0 的滤波器有效 / 无效选择位	0: 滤波器无效 1: 滤波器有效	R/W
b1	—	保留位	读写值都为“0”。	R/W
b3-b2	CPB0F[1:0]	比较器 B0 的滤波器选择位	b3 b2 0 0: 以 PCLK 采样 0 1: 以 PCLK/8 采样 1 0: 以 PCLK/32 采样 1 1: 以 PCLK/64 采样	R/W
b4	CPB1FEN	比较器 B1 的滤波器有效 / 无效选择位	0: 滤波器无效 1: 滤波器有效	R/W
b5	—	保留位	读写值都为“0”。	R/W
b7-b6	CPB1F[1:0]	比较器 B1 的滤波器选择位	b7 b6 0 0: 以 PCLK 采样 0 1: 以 PCLK/8 采样 1 0: 以 PCLK/32 采样 1 1: 以 PCLK/64 采样	R/W

36.3 运行说明

比较器 B0 和比较器 B1 能分别独立进行相同运行。比较器 B 的相关寄存器的设定步骤如表 36.3 所示。

表 36.3 比较器 B 的相关寄存器的设定步骤

顺序	寄存器	位名	设定值
1	CPBCNT1	CPBiINI (i=0、1)	接通电源 (ON): 1
2	CPBF	选择有无滤波器、采样时钟	
3	CPBINT	CPBiINTEN (i=0、1)	使用中断时: 1 (允许中断)
		CPBiINTEG (i=0、1)	使用中断/ELC 时: 选择输入边沿 (双边沿 / 单边沿)
		CPBiINTPL (i=0、1)	使用中断/ELC 时: 如果 CPBiINTEG 为“0” (选择单边沿), 选择输入极性 (上升沿 / 下降沿)
4	IPR058 (比较器 B0) IPR059 (比较器 B1)	IPR[3:0]	使用中断时: 选择中断优先级
	IR058 (比较器 B0) IR059 (比较器 B1)	IR	使用中断时: 0 (无中断请求: 初始化)
	IER07	IEN2 (比较器 B0) IEN3 (比较器 B1)	使用中断时: 1 (允许中断控制器 (ICU) 侧的中断)
5	等待比较器稳定的时间 (最大 100μs)		

比较器 Bi (i=0、1) 的运行例子如图 36.2 所示。

当模拟输入的电压高于基准输入电压时, CPBFLG.CPBiOUT 位变为“1”; 当模拟输入电压低于基准输入电压时, CPBFLG.CPBiOUT 位变为“0”。

在使用比较器 Bi 中断时, 必须将 CPBINT.CPBiINTEN 位置“1” (允许中断)。此时, 如果比较结果发生变化, 就产生比较器 Bi 中断请求。有关中断, 请参照“36.4 比较器 B0 中断和比较器 B1 中断”。

比较器 Bi 能向 ELC 输出事件, 使其他模块运行。有关 ELC, 请参照“36.5 事件链接输出功能”。

在进行比较的过程中, 不能更改各寄存器的值。

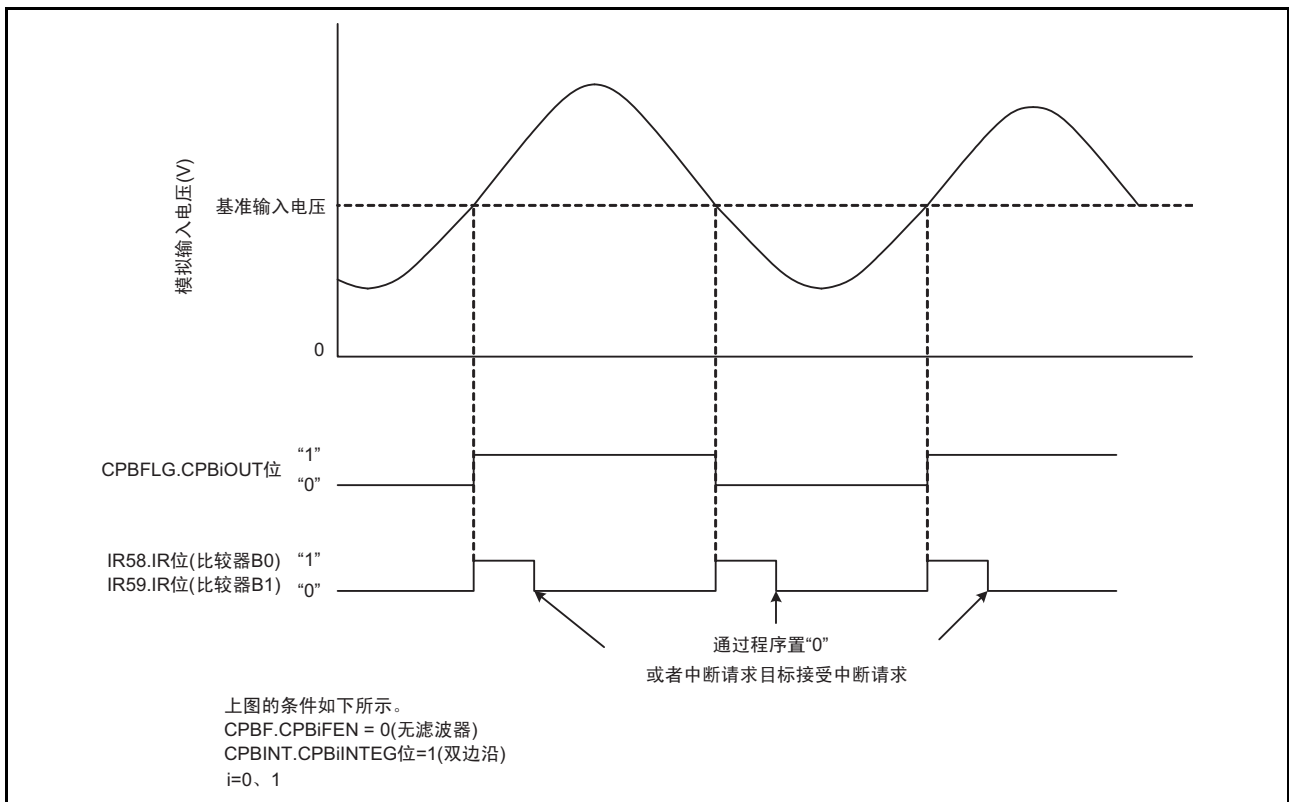


图 36.2 比较器 Bi (i=0、1) 的运行例子

36.3.1 比较器 Bi 的数字滤波器 (i=0、1)

能通过 CPBF.CPBIF[1:0] 位选择采样时钟。在每个采样时钟对比较器 Bi 的 CPBiOUT 输出信号（内部信号）进行采样，当电平 3 次相同时，IR058.IR 位（选择比较器 B0 时）、IR059.IR 位（选择比较器 B1 时）就变为“1”（有中断请求），并且输出 ELC 事件。

比较器 Bi 的数字滤波器结构如图 36.3 所示，比较器 Bi 的数字滤波器运行例子如图 36.4 所示。

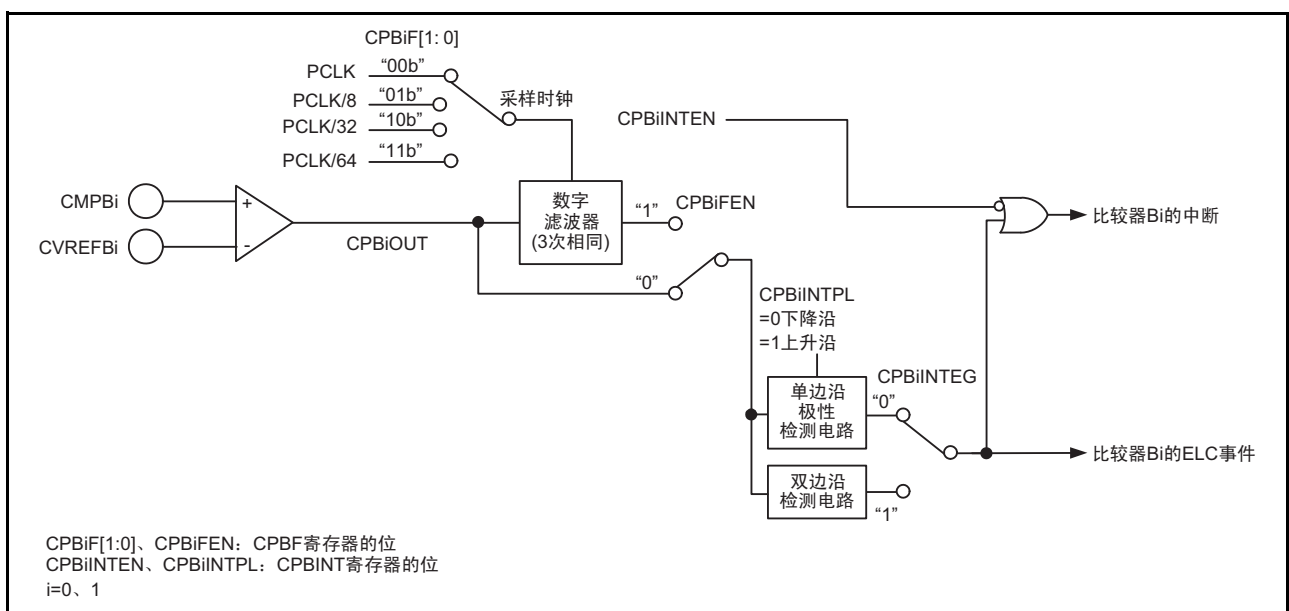


图 36.3 比较器 Bi 的数字滤波器结构

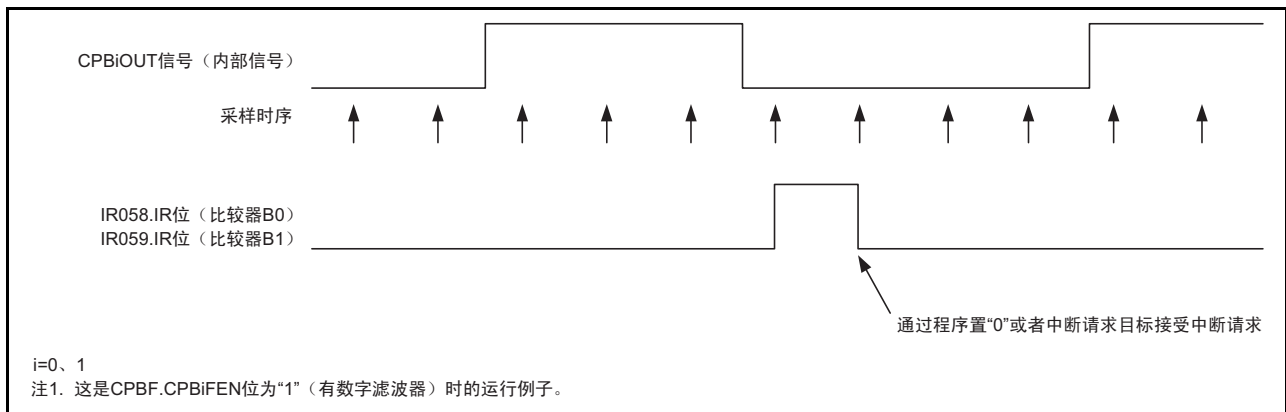


图 36.4 比较器 Bi 的数字滤波器运行例子

36.4 比较器 B0 中断和比较器 B1 中断

比较器 B 产生比较器 B0 和比较器 B1 的两个中断请求。比较器 Bi ($i=0, 1$) 中断分别具有 1 个中断向量以及 IR058.IR 位、IR059.IR 位、IPR058.IPR[3:0] 位和 IPR059.IPR[3:0] 位。

在使用比较器 Bi 中断时，必须将 CPBINT.CPBiINTEN 位置“1” (允许中断)。能通过 CPBINT.CPBiINTEG 位选择单边沿检测或者双边沿检测。在选择单边沿检测时，能通过 CPBINT.CPBiINTPL 位选择极性。

另外，比较器 B 能通过具有 3 种采样时钟的数字滤波器进行输入。

36.5 事件链接输出功能

CMPB 具有在以下时序向事件链接控制器 (ELC) 输出事件的功能。

1. 当比较器 B0 的比较结果发生变化时
2. 当比较器 B0 或者 B1 的比较结果发生变化时

当比较器 B0、比较器 B1 的比较结果同时或者连续输出时，作为 1 个事件输出。

36.5.1 中断处理和事件链接的关系

比较器 Bi 向 ELC (Even Link Controller) 输出事件，并且能使事先设定的模块运行。与 CPBiINTEN 位的值无关，向 ELC 输出事件。

与中断请求信号相同，从比较器 Bi 向 ELC 输出的事件输出信号能通过 CPBINT.CPBiINTEG 位选择单边沿检测或者双边沿检测。在选择单边沿检测时，能通过 CPBINT.CPBiINTPL 位选择极性。

36.6 使用时的注意事项

36.6.1 模块停止功能的设定

能通过模块停止控制寄存器 B (MSTPCRB) 设定禁止或者允许比较器 B 的运行。初始值为停止比较器 B 的运行。能通过解除模块停止状态存取寄存器。详细内容请参照“11. 低功耗功能”。

37. 数据运算电路 (DOC)

37.1 概要

数据运算电路 (DOC) 对 16 位数据进行比较、加法运算或者减法运算。
数据运算电路 (DOC) 的规格和框图分别如表 37.1 和图 37.1 所示。

- 能比较 16 位数据, 并且在选择的条件下发生中断。
- 能对 16 位数据进行加法运算。
- 能对 16 位数据进行减法运算。

表 37.1 数据运行电路 (DOC) 的规格

项目	内容
数据运算功能	对 16 位数据进行比较、加法运算或者减法运算
低功耗功能	能设定为模块停止状态

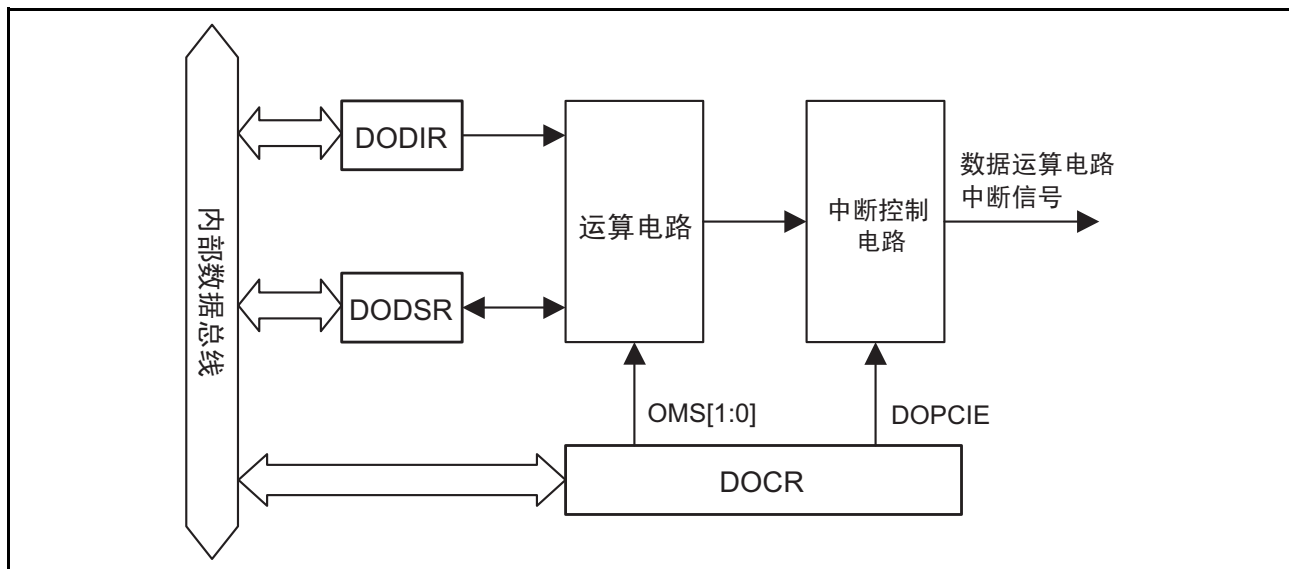


图 37.1 数据运算电路的框图

37.2 寄存器说明

37.2.1 DOC 控制寄存器 (DOCR)

地址 0008 B080h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	DOPCFCL	DOPCF	DOPCIE	—	DCSEL	OMS [1:0]	
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b1-b0	OMS[1:0]	运行模式选择位	b1 b0 0 0: 数据比较模式 0 1: 数据加法运算模式 1 0: 数据减法运算模式 1 1: 不能设定	R/W
b2	DCSEL (注 1)	检测条件选择位	0: 检测到数据比较的结果不相同 1: 检测到数据比较的结果相同	R/W
b3	—	保留位	读写值都为“0”。	R/W
b4	DOPCIE	数据运算电路中断允许位	0: 数据运算电路中断无效 1: 数据运算电路中断有效	R/W
b5	DOPCF	数据运算电路标志	显示运算结果	R
b6	DOPCFCL	DOPCF 清除位	0: 不进行任何运行 1: 清除 DOPCF 标志	R/W
b7	—	保留位	读写值都为“0”。	R/W

注 1. 只在数据比较模式中有效。

OMS[1:0] 位 (运行模式选择位)

通过设定此位选择数据运算电路的运行模式。

DCSEL 位 (检测条件选择位)

只在数据比较模式中有效。

通过设定此位选择数据比较模式中的结果检测条件。

DOPCIE 位 (数据运算电路中断允许位)

当此位为“1”时，允许数据运算电路中断。

DOPCF 标志 (数据运算电路标志)

[为“1”的条件]

- 当变为 DCSEL 位选择的条件时
- 当数据加法运算的结果大于“FFFFh”时
- 当数据减法运算的结果小于“0000h”时

[为“0”的条件]

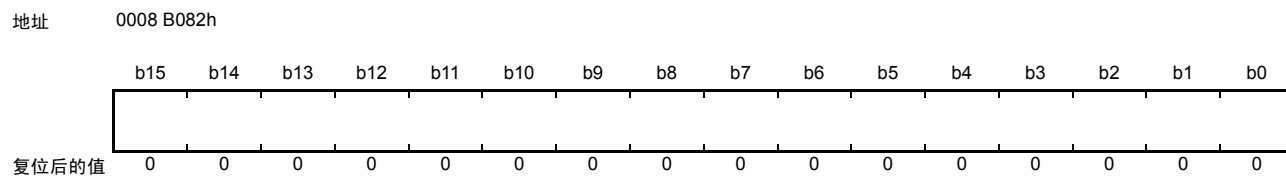
- 给 DOPCFCL 位写“1”时

DOPCFCL 位 (DOPCF 清除位)

如果将此位置“1”，就清除 DOPCF 标志。

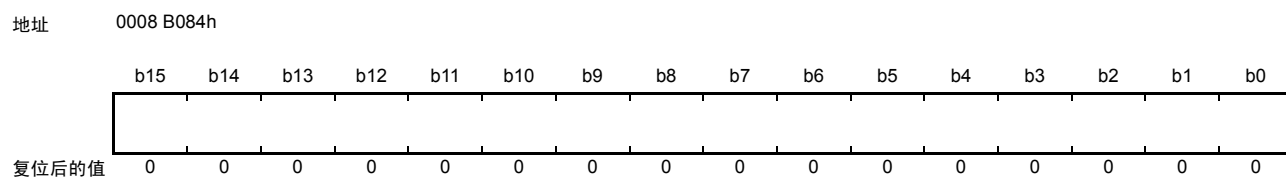
读取值为“0”。

37.2.2 DOC 数据输入寄存器 (DODIR)



DODIR 寄存器是保存运算对象的 16 位数据的 16 位可读写寄存器。

37.2.3 DOC 数据设定寄存器 (DODSR)



DODSR 寄存器在数据比较模式中是保存基准 16 位数据的 16 位可读写寄存器。另外，在数据加法运算模式和数据减法运算模式中是保存运算结果的寄存器。

37.3 运算说明

37.3.1 数据比较模式

数据比较模式的运行例子如图 37.2 所示。

在数据比较模式中，数据运算电路如下运行。

1. 如果给 DOCR.OMS[1:0] 位写“00b”，就为数据比较模式。
2. 给 DODSR 寄存器设定作为基准的 16 位数据。
3. 给 DODIR 寄存器写要进行比较的 16 位数据。
4. 给 DODIR 寄存器写要进行比较的 16 位数据，直到写完全部要进行比较的数据。
5. 当写到 DODIR 寄存器的数据与设定到 DODSR 寄存器的数据不相同，DOCR.DOPCF 标志就被置“1”。另外，如果给 DOCR.DOPCIE 位设定“1”，就发生数据运算电路中断。

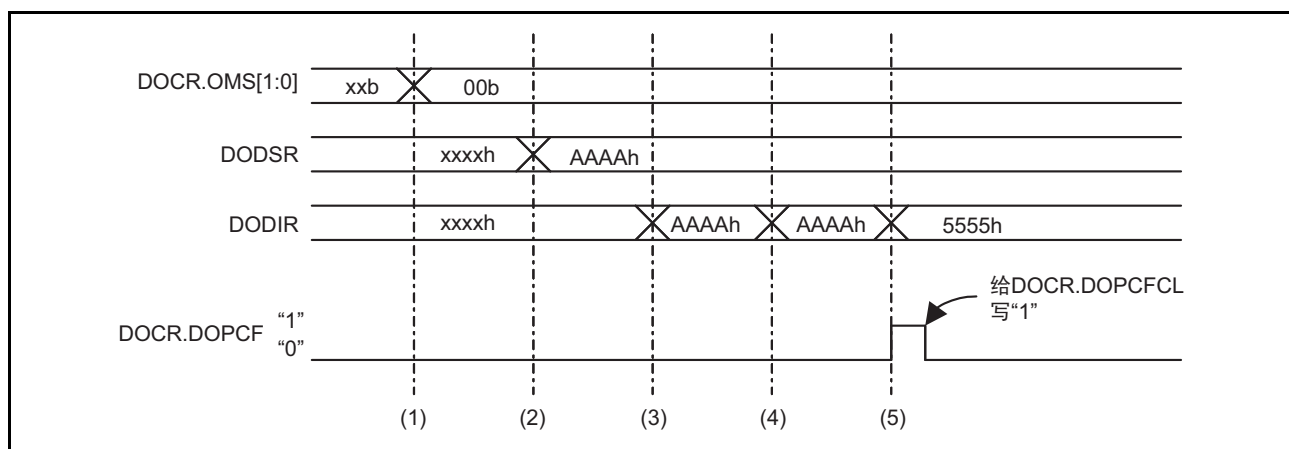


图 37.2 数据比较模式的运行例子

37.3.2 数据加法运算模式

数据加法运算模式的运行例子如图 37.3 所示。

在数据加法运算模式中，数据运算电路如下运行。

1. 如果给 DOCR.OMS[1:0] 位写“01b”，就为数据加法运算模式。
2. 给 DODSR 寄存器设定作为初始值的 16 位数据。
3. 给 DODIR 寄存器写要进行加法运算的 16 位数据。运算结果保存到 DODSR 寄存器。
4. 给 DODIR 寄存器写要进行加法运算的 16 位数据，直到写完全部要进行加法运算的数据。
5. 当运算结果大于“FFFFh”时，DOCR.DOPCF 标志就被置“1”。另外，如果给 DOCR.DOPCIE 位设定“1”，就发生数据运算电路中断。

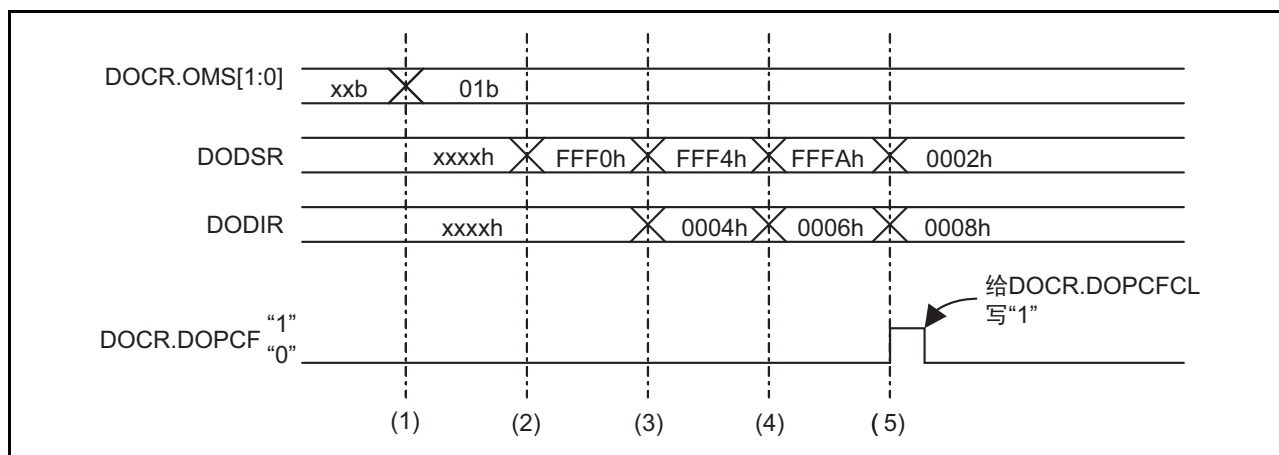


图 37.3 数据加法运算模式的运行例子

37.3.3 数据减法运算模式

数据减法运算模式的运行例子如图 37.4 所示。
在数据减法运算模式中，数据运算电路如下运行。

1. 如果给 DOCR.OMS[1:0] 位写“10b”，就为数据减法运算模式。
2. 给 DODSR 寄存器设定作为初始值的 16 位数据。
3. 给 DODIR 寄存器写要进行减法运算的 16 位数据。运算结果保存到 DODSR 寄存器。
4. 给 DODIR 寄存器写要进行减法运算的 16 位数据，直到写全部要进行减法运算的数据。
5. 当运算结果小于“0000h”时，DOCR.DOPCF 标志就被置“1”。另外，如果给 DOCR.DOPCIE 位设定“1”，就发生数据运算电路中断。

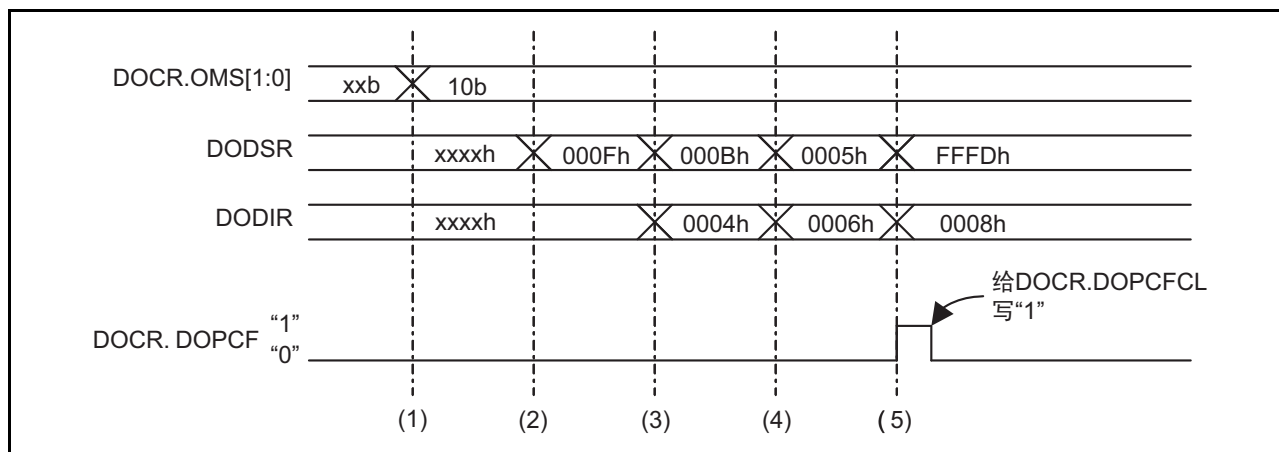


图 37.4 数据减法运算模式的运行例子

37.4 中断请求

数据运算电路生成的中断请求只有数据运算电路中断请求。如果产生中断源，数据运算电路标志就被置“1”。中断请求的内容如表 37.2 所示。

表 37.2 数据运算电路中断请求

中断请求	数据运算电路标志	中断源
数据运算电路中断	DOPCF	<ul style="list-style-type: none"> • 当变为 DOCR.DCSEL 位选择的条件时 • 当数据加法运算的结果大于“FFFFh”时 • 当数据减法运算的结果小于“0000h”时

37.5 使用时的注意事项

37.5.1 模块停止功能的设定

能通过模块停止控制寄存器 B (MSTPCRB) 设定禁止或者允许数据运算电路的运算。初始值时为停止数据运算电路的运行。能通过解除模块停止状态存取寄存器。详细内容请参照“11. 低功耗功能”。

38. RAM

RX210 群内置高速静态 RAM。

38.1 概要

RAM 的规格如表 38.1 所示。

表 38.1 RAM 的规格

项目	内容
RAM 容量	最大 64K 字节（RAM0: 64K 字节）（注 2）
存取	<ul style="list-style-type: none"> 以 1 个周期进行读写。 能选择内部 RAM 的有效或者无效（注 1）。
数据保持功能	无（不能在深度软件待机模式中保持 RAM 数据。）
低功耗功能	能设定为模块停止状态。

注 1. 能通过 SYSCR1.RAME 位进行选择。有关 SYSCR1 寄存器，请参照“3.2.4 系统控制寄存器 1（SYSCR1）”。

注 2. RAM 容量因产品而不同。

RAM 容量	RAM 地址
64K 字节	RAM0: 0000 0000h ~ 0000 FFFFh
32K 字节	RAM0: 0000 0000h ~ 0000 7FFFh
20K 字节	RAM0: 0000 0000h ~ 0000 4FFFh

38.2 运行说明

38.2.1 数据的保持

因为内部 RAM 在深度软件待机模式中停止给 RAM 提供内部电源，所以无法保持 RAM 数据。

38.2.2 低功耗功能

通过设定模块停止控制寄存器 C（MSTPCRC）停止给 RAM 提供时钟，能降低功耗。

如果将 MSTPCRC.MSTPC0 位置“1”，提供给 RAM 的时钟就停止。

通过停止提供时钟，使 RAM 变为模块停止状态。但是在复位后的初始值时，RAM 为运行状态。

一旦进入模块停止状态，就无法存取 RAM。不能在存取 RAM 的过程中转移到模块停止状态。

有关 MSTPCRC 寄存器的详细内容，请参照“11. 低功耗功能”。

39. ROM（保存代码的闪存）

RX210 群内置用于保存最多 512K 字节代码的闪存（ROM）。

本章说明用于保存代码的闪存，有关 E2 数据闪存，请参照“40. E2 数据闪存（保存数据的闪存）”。

39.1 概要

ROM 的规格如表 39.1 所示，ROM 容量和 ROM 地址的对应表如表 39.2 所示，ROM 和 E2 数据闪存的框图如图 39.1 所示。

表 39.1 ROM 的规格

项目	内容	
存储空间	<ul style="list-style-type: none"> • 用户区：最多 512K 字节 • 用户引导区：16K 字节 	
高速读取能力	能进行 1 个 ICLK 周期的高速读取。	
编程 / 擦除方式	<ul style="list-style-type: none"> • 内置对 ROM 进行改写的专用定序器（FCU）。 • 能通过给 FCU 发行命令，进行 ROM 的编程或者擦除。 • 如果读擦除状态的 ROM，就能以 32 位为单位读到“FFFF FFFFh”。 	
BGO（后台操作）功能	<ul style="list-style-type: none"> • 在对 E2 数据闪存进行编程或者擦除的期间，能执行分配在 ROM 区的程序。 • 在对 ROM 进行编程或者擦除的期间，CPU 能执行分配在 ROM/E2 数据闪存以外区域的程序 	
挂起 / 恢复功能	<ul style="list-style-type: none"> • 中止对 ROM 进行的编程或者擦除，CPU 能执行 ROM 区的程序（挂起）。 • 在中止后，能重新开始对 ROM 进行编程或者擦除（恢复）。 	
编程 / 擦除单位	<ul style="list-style-type: none"> • 用户区和用户引导区的编程单位：2/8/128 字节 • 用户区的擦除单位：以块为单位 • 用户引导区的擦除单位：16K 字节 	
板上编程 (3 种)	通过引导模式改写 <ul style="list-style-type: none"> • 使用异步串行接口（SCI1）。 • 自动调整通信速度。 • 能改写用户引导区。 	
	通过用户引导模式改写 <ul style="list-style-type: none"> • 能建立用户独自の引导程序。 	
	通过用户程序中的 ROM 改写程序改写 <ul style="list-style-type: none"> • 不通过系统复位就能进行 ROM 的改写。 	
板外编程	能使用 PROM 编程器改写用户区和用户引导区。	
保护功能	软件保护功能	能通过 FENTRYR.FENTRY0 位、FWEPROR.FLWE[1:0] 位和锁定位防止意料不到的改写。
	错误保护功能	如果在编程或者擦除过程中检测到异常，就禁止以后的编程或者擦除处理。
编程时间 / 擦除时间 / 改写次数	请参照“41. 电特性”。	

表 39.2 ROM 容量和 ROM 地址的对应表

ROM 容量	ROM 地址
128K 字节	FFFE 0000h ~ FFFF FFFFh
256K 字节	FFFC 0000h ~ FFFF FFFFh
384K 字节	FFFA 0000h ~ FFFF FFFFh
512K 字节	FFF8 0000h ~ FFFF FFFFh

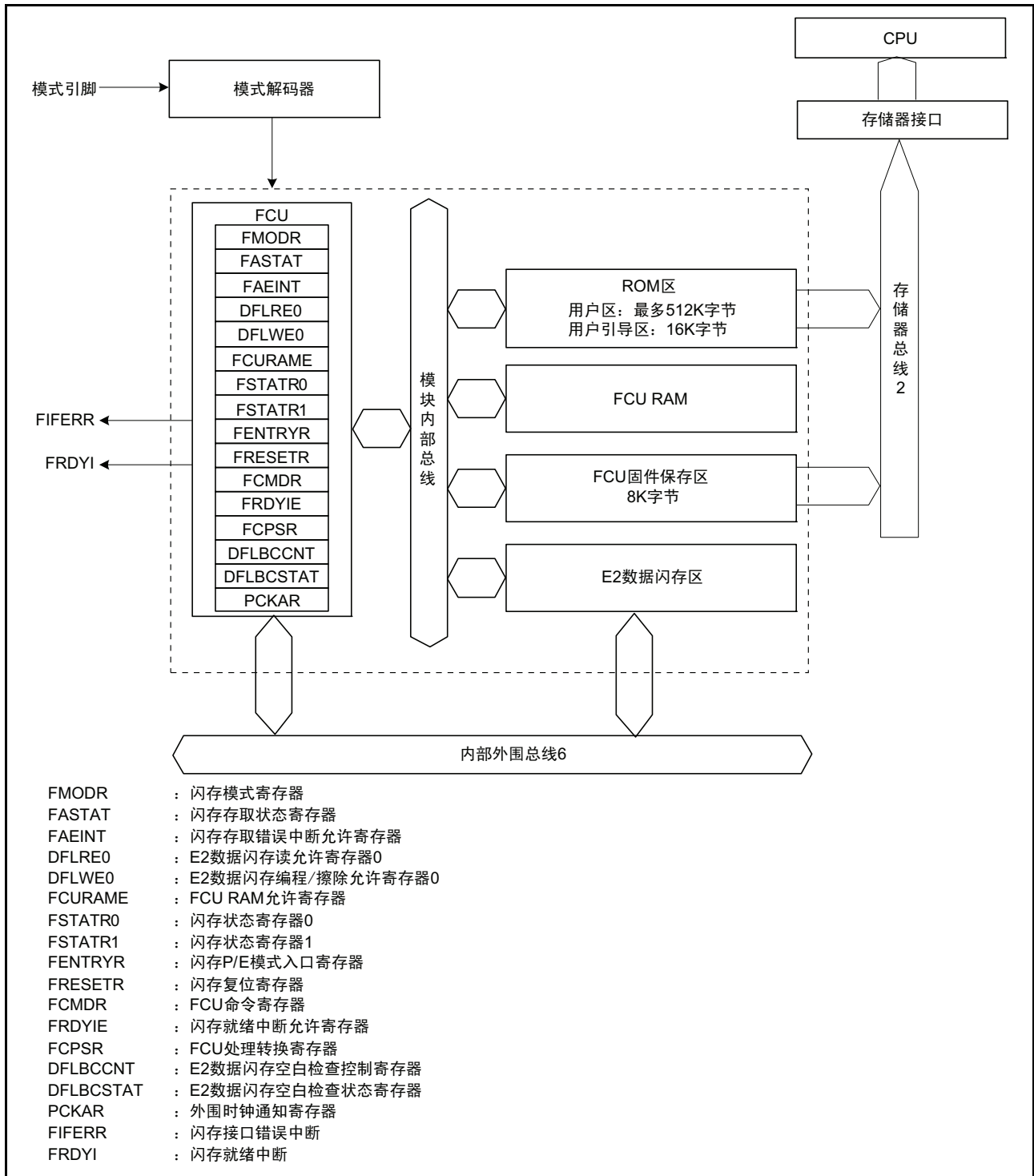


图 39.1 ROM 的框图

ROM 的相关输入 / 输出引脚如表 39.3 所示。

表 39.3 ROM 的相关输入 / 输出引脚

引脚名	输入 / 输出	功能
MD	输入	设定运行模式。
PC7	输入	设定引导模式、用户引导模式。
P30/RXD1	输入	用于引导模式。SCI1 的接收数据 (用于主机通信)。
P26/TXD1	输出	用于引导模式。SCI1 的接收数据 (用于主机通信)。

39.2 寄存器说明

部分寄存器也有 E2 数据闪存的相关位，本章只说明 ROM 的相关位的功能。有关 E2 数据闪存相关位功能的详细内容，请参照“40. E2 数据闪存（保存数据的闪存）”的“40.2 寄存器说明”。

P/E 表示编程 / 擦除。

39.2.1 闪存编程 / 擦除保护寄存器（FWEPROR）

地址 0008 C296h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	FLWE[1:0]	
0	0	0	0	0	0	1	0

复位后的值

位	符号	位名	功能	R/W
b1-b0	FLWE[1:0]	闪存编程 / 擦除位	b1 b0 0 0: 不能进行编程和擦除 0 1: 能进行编程和擦除 1 0: 不能进行编程和擦除 1 1: 不能进行编程和擦除	R/W
b7-b2	—	保留位	读写值都为“0”。	R/W

在软件待机模式和深度软件待机模式中，对 FWEPROR 寄存器进行初始化。

FLWE[1:0] 位（闪存编程 / 擦除位）

这些位通过软件对闪存编程和擦除的执行进行保护。

39.2.2 闪存模式寄存器（FMODR）

地址 007F C402h

b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	FRDMD	—	—	—	—
0	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b3-b0	—	保留位	读写值都为“0”。	R/W
b4	FRDMD	FCU 读模式选择位	0: 存储区读方式 在 ROM 锁定位读模式中读 ROM 的锁定位时进行设定。 1: 寄存器读方式 在使用锁定位读 2 命令读 ROM 的锁定位时进行设定。	R/W
b7-b5	—	保留位	读写值都为“0”。	R/W

在内部 ROM 无效时，FMODR 寄存器的读数据为“00h”，不能写。

FRDMD 位（FCU 读模式选择位）

此位指定锁定位的读法。

在使用 E2 数据闪存的空白检查命令时，需要设定为寄存器读方式。

详细内容请参照“40. E2 数据闪存（保存数据的闪存）”。

39.2.3 闪存存取状态寄存器（FASTAT）

地址 007F C410h

	b7	b6	b5	b4	b3	b2	b1	b0
	ROMAE	—	—	CMDLK	DFLAE	—	DFLRPE	DFLWPE
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	DFLWPE	E2 数据闪存的编程 / 擦除保护违反位	请参照“40. E2 数据闪存（保存数据的闪存）”。	R/(W) (注1)
b1	DFLRPE	E2 数据闪存的读保护违反位	请参照“40. E2 数据闪存（保存数据的闪存）”。	R/(W) (注1)
b2	—	保留位	读写值都为“0”。	R/W
b3	DFLAE	E2 数据闪存的存取违反位	请参照“40. E2 数据闪存（保存数据的闪存）”。	R/(W) (注1)
b4	CMDLK	FCU 命令锁定位	0: FCU 不处于命令锁定状态 1: FCU 处于命令锁定状态	R
b6-b5	—	保留位	读写值都为“0”。	R/W
b7	ROMAE	ROM 存取违反位	0: 没有 ROM 存取错误 1: 有 ROM 存取错误	R/(W) (注1)

注. 为了将标志置“0”，只能在读“1”后写“0”。

在内部 ROM 无效时，FASTAT 寄存器的读数据为“00h”，不能写。如果 FASTAT 寄存器的某位变为“1”，FCU 就进入命令锁定状态（参照“39.8.2 错误保护”）。为了解除命令锁定状态，需要在将 FASTAT 寄存器置“10h”后，给 FCU 发行状态寄存器清除命令。

CMDLK 位（FCU 命令锁定位）

此位表示 FCU 处于命令锁定状态（参照“39.8.2 错误保护”）。

[为“1”的条件]

- FCU 检测到错误并且转移到命令锁定状态后

[为“0”的条件]

- 在 FASTAT 寄存器为“10h”的状态下 FCU 发行状态寄存器清除命令后

ROMAE 位（ROM 存取违反位）

此位表示有没有违反 ROM 的存取。

如果 ROMAE 位变为“1”，FSTATR0.ILGLERR 位就变为“1”，FCU 进入命令锁定状态。

[为“1”的条件]

- 在 ROM P/E 正常模式的状态下对 ROM 编程/擦除地址发行读存取命令时

ROM 容量	ROM 编程 / 擦除地址
	FENTRY0 位为“1”
128K 字节	00FE 0000h ~ 00FF FFFFh
256K 字节	00FC 0000h ~ 00FF FFFFh
384K 字节	00FA 0000h ~ 00FF FFFFh
512K 字节	00F8 0000h ~ 00FF FFFFh

- 当对ROM编程/擦除地址发行存取命令时

ROM 容量	ROM 编程 / 擦除地址
	FENTRY0 位为“0”
128K 字节	00FE 0000h ~ 00FF FFFFh
256K 字节	00FC 0000h ~ 00FF FFFFh
384K 字节	00FA 0000h ~ 00FF FFFFh
512K 字节	00F8 0000h ~ 00FF FFFFh

- 在设定FENTRYR寄存器后转移到ROM P/E模式的状态下，对ROM读地址发行读存取命令时

ROM 容量	ROM 编程 / 擦除地址
128K 字节	FFFE 0000h ~ FFFF FFFFh
256K 字节	FFFC 0000h ~ FFFF FFFFh
384K 字节	FFFA 0000h ~ FFFF FFFFh
512K 字节	FFF8 0000h ~ FFFF FFFFh

[为“0”的条件]

- 在读“1”后写“0”时

39.2.4 闪存存取错误中断允许寄存器（FAEINT）

地址 007F C411h

	b7	b6	b5	b4	b3	b2	b1	b0
	ROMAE IE	—	—	CMDLK IE	DFLAEIE	—	DFLRPE IE	DFLWPE IE
复位后的值	1	0	0	1	1	0	1	1

位	符号	位名	功能	R/W
b0	DFLWPEIE	E2 数据闪存的编程 / 擦除保护违反中断允许位	请参照“40. E2 数据闪存（保存数据的闪存）”。	R/W
b1	DFLRPEIE	E2 数据闪存的读保护违反中断允许位	请参照“40. E2 数据闪存（保存数据的闪存）”。	R/W
b2	—	保留位	读写值都为“0”。	R/W
b3	DFLAEIE	E2 数据闪存的存取违反中断允许位	请参照“40. E2 数据闪存（保存数据的闪存）”。	R/W
b4	CMDLKIE	FCU 命令锁定中断允许位	0: 在 FASTAT.CMDLK 位为“1”时，不产生 FIFERR 中断请求。 1: 在 FASTAT.CMDLK 位为“1”时，产生 FIFERR 中断请求。	R/W
b6-b5	—	保留位	读写值都为“0”。	R/W
b7	ROMAEIE	ROM 存取违反中断允许位	0: 在 FASTAT.ROMAE 位为“1”时，不产生 FIFERR 中断请求。 1: 在 FASTAT.ROMAE 位为“1”时，产生 FIFERR 中断请求。	R/W

在内部 ROM 无效时，FAEINT 寄存器的读数据为“00h”，不能写。

CMDLKIE 位（FCU 命令锁定中断允许位）

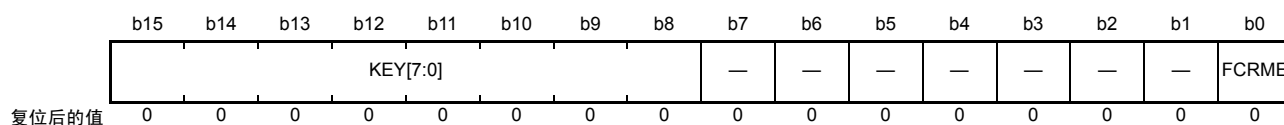
在发生 FCU 命令锁定并且 FASTAT.CMDLK 位变为“1”时，此位允许或者禁止 FIFERR 中断请求的产生。

ROMAEIE 位（ROM 存取违反中断允许位）

在发生 ROM 存取违反并且 FASTAT.ROMAE 位变为“1”时，此位允许或者禁止 FIFERR 中断请求的产生。

39.2.5 FCU RAM 允许寄存器（FCURAME）

地址 007F C454h



位	符号	位名	功能	R/W
b0	FCRME	FCU RAM 允许位	0: 禁止存取 FCU RAM 1: 允许存取 FCU RAM	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W
b15-b8	KEY[7:0]	键码	控制能否改写 FCRME 位。	R/(W) (注1)

注 1. 不保持写数据。

只在给高位字节写特定值时才能通过字存取写此寄存器，但是不保持高位字节的写数据。

在内部 ROM 无效时，FCURAME 寄存器的读数据为“00h”，不能写。

FCRME 位（FCU RAM 允许位）

此位允许或者禁止存取 FCU RAM。

只在 KEY[7:0] 位为“C4h”时才能通过字存取写 FCRME 位。要写 FCU RAM 时，必须将 FENTRYR 寄存器置“0000h”，使 FCU 停止。

与允许或者禁止存取 FCU RAM 无关，不能读 FCU RAM。如果读 FCU RAM，读取值就为不定值。

39.2.6 闪存状态寄存器 0（FSTATR0）

地址 007F FFB0h

b7	b6	b5	b4	b3	b2	b1	b0
FRDY	ILGLERR	ERSERR	PRGERR	SUSRDY	—	ERSSPD	PRGSPD
1	0	0	0	0	0	0	0

复位后的值

位	符号	位名	功能	R/W
b0	PRGSPD	编程挂起状态位	0: 下述以外的状态 1: 正在进行编程的中止处理或者正在编程挂起	R
b1	ERSSPD	擦除挂起状态位	0: 下述以外的状态 1: 正在进行擦除的中止处理或者正在擦除挂起	R
b2	—	保留位	读取值为“0”，写操作无效。	R
b3	SUSRDY	挂起就绪位	0: 不能接受 P/E 挂起命令 1: 能接受 P/E 挂起命令	R
b4	PRGERR	编程错误位	0: 编程处理正常结束 1: 在编程处理过程中发生错误	R
b5	ERSERR	擦除错误位	0: 擦除处理正常结束 1: 在擦除处理过程中发生错误	R
b6	ILGLERR	非法命令错误位	0: FCU 没有检测到非法命令和 ROM/E2 数据闪存的非法存取 1: FCU 检测到非法命令或者 ROM/E2 数据闪存的非法存取	R
b7	FRDY	闪存就绪位	0: 正在处理编程或者擦除，正在处理编程或者擦除的中止，正在处理锁定位读 2 命令，正在处理外围时钟通知命令，或者正在处理 E2 数据闪存的空白检查（参照“40. E2 数据闪存（保存数据的闪存）”）。 1: 没有执行上述处理	R

在内部 ROM 无效时，FSTATR0 寄存器的读数据为“00h”。

通过复位或者将 FRESETR.FRESET 位置“1”，对 FSTATR0 寄存器进行初始化。

PRGSPD 位（编程挂起状态位）

此位表示 FCU 正在进行编程的中止处理或者已经转移到编程挂起状态，详细内容请参照“39.7 挂起”。

[为“1”的条件]

- 当开始进行编程的中止处理时

[为“0”的条件]

- 当接受恢复命令时

ERSSPD 位（擦除挂起状态位）

此位表示 FCU 正在进行擦除的中止处理或者已经转移到擦除挂起状态，详细内容请参照“39.7 挂起”。

[为“1”的条件]

- 当开始进行擦除的中止处理时

[为“0”的条件]

- 当接受恢复命令时

SUSRDY 位（挂起就绪位）

此位表示 FCU 能否接受 P/E 挂起命令。

[为“1”的条件]

- 在开始进行编程或者擦除处理后，转移到能接受P/E挂起命令的状态时

[为“0”的条件]

- 当接受P/E挂起命令时
- 在编程或者擦除处理过程中转移到命令锁定状态时

PRGERR 位（编程错误位）

此位表示通过 FCU 对 ROM/E2 数据闪存进行编程处理的结果。

当 PRGERR 位为“1”时，FCU 进入命令锁定状态，详细内容请参照“39.8.2 错误保护”。

[为“1”的条件]

- 在编程过程中发生错误时
- 当对由锁定位保护的区域发行编程命令时

[为“0”的条件]

- 在FCU发行状态寄存器清除命令后

ERSERR 位（擦除错误位）

此位表示通过 FCU 对 ROM/E2 数据闪存进行擦除处理的结果。

当 ERSERR 位为“1”时，FCU 进入命令锁定状态，详细内容请参照“39.8.2 错误保护”。

[为“1”的条件]

- 在擦除过程中发生错误时
- 当对由锁定位保护的区域发行擦除命令时

[为“0”的条件]

- 在FCU发行状态寄存器清除命令后

ILGLERR 位（非法命令错误位）

此位表示 FCU 检测到非法命令或者 ROM/E2 数据闪存的非法存取。

当 ILGLERR 位为“1”时，FCU 进入命令锁定状态，详细内容请参照“39.8.2 错误保护”。

[为“1”的条件]

- 当FCU检测到非法命令时
- 当FCU检测到ROM/E2数据闪存的非法存取时
(FASTAT.ROMAE位、FASTAT.DFLAE位、FASTAT.DFLRPE位或者FASTAT.DFLWPE位为“1”)
- 当FENTRYR寄存器的设定不正确时

[为“0”的条件]

- 在FASTAT寄存器为“10h”的状态下，FCU发行状态寄存器清除命令后

FRDY 位（闪存就绪位）

此位确认 FCU 的处理状态。

39.2.7 闪存状态寄存器 1（FSTATR1）

地址 007F FFB1h

	b7	b6	b5	b4	b3	b2	b1	b0
	FCUERR	—	—	FLOCKST	—	—	—	—
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b3-b0	—	保留位	读取值为“0”，写操作无效。	R
b4	FLOCKST	锁定位的状态位	0: 保护状态 1: 非保护状态	R
b6-b5	—	保留位	读取值为“0”，写操作无效。	R
b7	FCUERR	FCU 错误位	0: 在 FCU 处理中未发生错误 1: 在 FCU 处理中发生错误	R

在内部 ROM 无效时，FSTATR1 寄存器的读数据为“00h”。

通过复位或者将 FRESETR.FRESET 位置“1”，对 FSTATR1 寄存器进行初始化。

FLOCKST 位（锁定位的状态位）

当使用锁定位读 2 命令时，此位反映锁定位的读数据。

在发行锁定位读 2 命令后 FSTATR0.FRDY 位变为“1”时，将有效数据保存到 FLOCKST 位。FLOCKST 位的值保持到下一个锁定位读 2 命令结束为止。

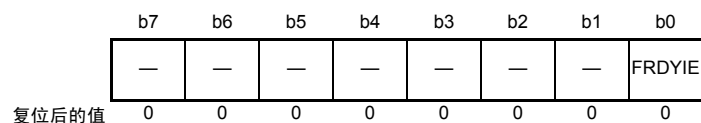
FCUERR 位（FCU 错误位）

此位表示在 FCU 内部处理中发生错误。

当 FCUERR 位为“1”时，必须将 FRESETR.FRESET 位置“1”，对 FCU 进行初始化。必须将 FCU 固件从 FCU 固件区重新复制到 FCU RAM 区。

39.2.8 闪存就绪中断允许寄存器（FRDYIE）

地址 007F C412h



位	符号	位名	功能	R/W
b0	FRDYIE	闪存就绪中断允许位	0: 禁止 FRDYI 中断请求的产生 1: 允许 FRDYI 中断请求的产生	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W

在内部 ROM 无效时，FRDYIE 寄存器的读数据为“00h”，不能写。

FRDYIE 位（闪存就绪中断允许位）

此位允许或者禁止在结束编程或者擦除处理时产生 FRDYI 中断请求。

如果在将 FRDYIE 位置“1”的状态下结束 FCU 命令的执行（FSTATR0.FRDY 位从“0”变为“1”），就产生闪存就绪中断请求（FRDYI）。

39.2.9 闪存 P/E 模式入口寄存器（FENTRYR）

地址 007F FFB2h



位	符号	位名	功能	R/W
b0	FENTRY0	ROM P/E 模式入口位 0	0: ROM 的 128K 字节、256K 字节、384K 字节或者 512K 字节为 ROM 读模式 1: ROM 的 128K 字节、256K 字节、384K 字节或者 512K 字节为 ROM P/E 模式	R/W
b6-b1	—	保留位	读写值都为“0”。	R/W
b7	FENTRYD	E2 数据闪存 P/E 模式入口位	请参照“40. E2 数据闪存（保存数据的闪存）”。	R/W
b15-b8	FEKEY[7:0]	键码	控制能否改写 FENTRYD 位和 FENTRY0 位。	R/(W) (注 1)

注 1. 不保持写数据。

为了将 ROM/E2 数据闪存设定为 P/E 模式，使 FCU 能接受命令，需要将 FENTRYD 位或者 FENTRY0 位置“1”。但是，如果将这些位中的多个位置“1”，FSTATR0.ILGLERR 位就变为“1”，FCU 进入命令锁定状态。

在存取 FENTRYR 寄存器后转移到 ROM 读模式时，必须先确认写 FENTRYR 寄存器后的此寄存器的读取值是否为设定值，然后再进行 ROM 读运行。

只在给高位字节写特定值时才能通过字存取写此寄存器，而在其他情况下写此寄存器时，此寄存器就被初始化，但是不保持高位字节的写数据。

在内部 ROM 无效模式中，FENTRYR 寄存器的读数据为“0000h”，不能写。

通过复位或者将 FRESETR.FRESET 位置“1”，对 FENTRYR 寄存器进行初始化。

FENTRY0 位（ROM P/E 模式入口位 0）

此位将 ROM 128K 字节（读地址：FFFA 0000h ~ FFFF FFFFh，编程 / 擦除地址：00FE 0000h ~ 00FF FFFFh）、ROM 256K 字节（读地址：FFFC 0000h ~ FFFF FFFFh，编程 / 擦除地址：00FC 0000h ~ 00FF FFFFh）、ROM 384K 字节（读地址：FFFA 0000h ~ FFFF FFFFh，编程 / 擦除地址：00FA 0000h ~ 00FF FFFFh）或者 ROM 512K 字节（读地址：FFF8 0000h ~ FFFF FFFFh，编程 / 擦除地址：00F8 0000h ~ 00FF FFFFh）、设定为 P/E 模式。

[写操作的有效条件（满足以下全部条件时）]

- 内部 ROM 为有效模式。
- FSTATR0.FR DY 位为“1”。
- 通过字存取给 FEKEY[7:0] 位写“AAh”。

[为“1”的条件]

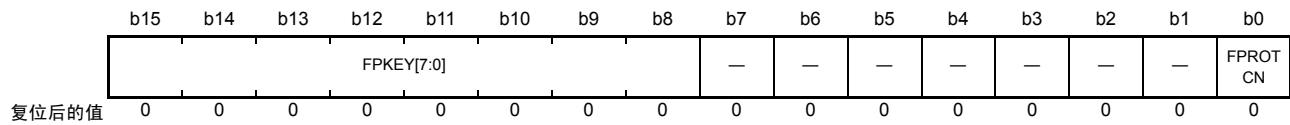
- 在满足写操作的有效条件并且 FENTRYR 寄存器为“0000h”的状态下，给 FENTRY0 位写“1”时

[为“0”的条件]

- 当通过字节存取进行写操作时
- 在 FEKEY[7:0] 位不为“AAh”的状态下，通过字存取进行写操作时
- 在满足写操作的有效条件的状态下给 FENTRY0 位写“0”时
- 在满足写操作的有效条件并且 FENTRYR 寄存器不为“0000h”的状态下，写 FENTRYR 寄存器时

39.2.10 闪存保护寄存器（FPROTR）

地址 007F FFB4h



位	符号	位名	功能	R/W
b0	FPROTCN	锁定位保护取消位	0: 通过锁定位进行的保护有效 1: 通过锁定位进行的保护无效	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W
b15-b8	FPKEY[7:0]	键码	控制能否改写 FPROTCN 位。	R/(W) (注1)

注 1. 不保持写数据。

只在给高位字节写特定值时才能通过字存取写此寄存器，而在其他情况下写此寄存器时，此寄存器被初始化，但是不保持高位字节的写数据。

在内部 ROM 无效模式中，FPROTR 寄存器的读数据为“0000h”，不能写。

通过复位或者将 FRESETR.FRESET 位置“1”，对 FPROTR 寄存器进行初始化。

FPROTCN 位（锁定位保护取消位）

此位设定通过锁定位进行的编程 / 擦除保护的有效或者无效。

[为“1”的条件]

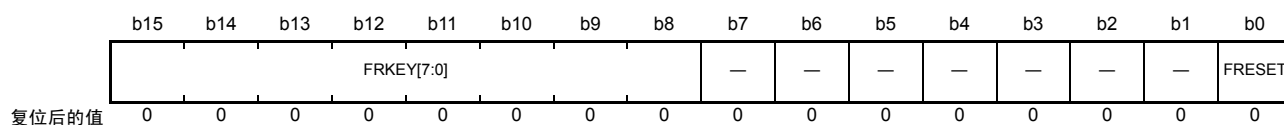
- 在 FENTRYR 寄存器的值不为“0000h”的状态下，通过字存取给 FPKEY[7:0] 位写“55h”并且给 FPROTCN 位写“1”时

[为“0”的条件]

- 当通过字节存取进行写操作时
- 在 FPKEY[7:0] 位不为“55h”的状态下，通过字存取进行写操作时
- 通过字存取给 FPKEY[7:0] 位写“55h”并且给 FPROTCN 位写“0”时
- 当 FENTRYR 寄存器的值为“0000h”时

39.2.11 闪存复位寄存器（FRESETR）

地址 007F FFB6h



位	符号	位名	功能	R/W
b0	FRESET	闪存复位的位	0: 不对 FCU 进行复位 1: 对 FCU 进行复位	R/W
b7-b1	—	保留位	读写值都为“0”。	R/W
b15-b8	FRKEY[7:0]	键码	控制能否改写 FRESET 位。	R/(W) (注1)

注 1. 不保持写数据。

只在给高位字节写特定值时才能通过字存取写此寄存器，但是不保持高位字节的写数据。

在内部 ROM 无效模式中，FRESETR 寄存器的读数据为“0000h”，不能写。

FRESET 位（闪存复位的位）

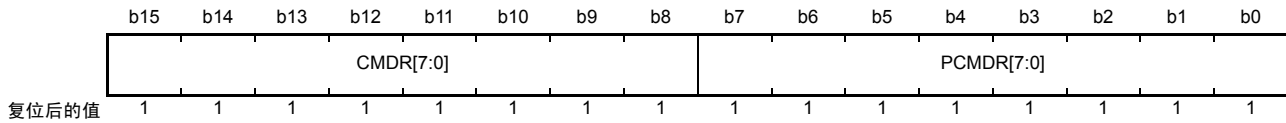
如果将 FRESET 位置“1”，ROM/E2 数据闪存的编程或者擦除就被强制结束并且 FCU 被初始化。

给编程或者擦除过程中的 ROM/E2 数据闪存的存储器外加高电压。为了确保存储器外加电压的下降所需的时间，在对 FCU 进行初始化时，FRESET 位为“1”的状态必须保持 tRESW2（参照“41. 电特性”）的时间。在 FRESET 位保持“1”的期间，必须禁止读 ROM/E2 数据闪存。因为在 FRESET 位为“1”的状态下对 FENTRYR 寄存器进行初始化，所以不能使用 FCU 命令。

只在 FRKEY[7:0] 位为“CCh”时才能通过字存取写 FRESET 位。

39.2.12 FCU 命令寄存器（FCMDR）

地址 007F FFBAh



位	符号	位名	功能	R/W
b7-b0	PCMDR[7:0]	预置命令	保存 FCU 接受的前一个命令。	R
b15-b8	CMDR[7:0]	命令	保存 FCU 接受的最新命令。	R

在内部 ROM 无效时，FCMDR 寄存器的读数据为“0000h”，写操作无效。

通过复位或者将 FRESETR.FRESET 位置“1”，对 FCMDR 进行初始化。

接受各命令后的 FCMDR 寄存器的状态如表 39.4 所示，空白检查的内容请参照 E2 数据闪存的“40.6 E2 数据闪存的编程 / 擦除”。

表 39.4 接受各命令后的 FCMDR 寄存器状态

命令	CMDR	PCMDR
正常模式转移	FFh	前一个命令
状态读模式转移	70h	前一个命令
锁定位读模式转移（锁定位读 1）	71h	前一个命令
外围时钟通知命令	E9h	前一个命令
编程	E8h	前一个命令
块擦除	D0h	20h
P/E 挂起	B0h	前一个命令
P/E 恢复	D0h	前一个命令
状态寄存器清除	50h	前一个命令
锁定位读 2 的空白检查	D0h	71h
锁定位编程	D0h	77h

39.2.13 FCU 处理转换寄存器（FCPSR）

地址 007F FFC8h

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ESUSPMD
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	ESUSPMD	编程 / 擦除挂起模式位	0: 挂起优先模式 1: 擦除优先模式	R/W
b15-b1	—	保留位	读写值都为“0”。	R/W

在内部 ROM 无效时，FCPSR 寄存器的读数据为“0000h”，不能写。

通过复位或者将 FRESETR.FRESET 位置“1”，对 FCPSR 寄存器进行初始化。

ESUSPMD 位（编程 / 擦除挂起模式位）

在 FCU 进行 ROM/E2 数据闪存的擦除处理过程中发行 P/E 挂起命令时，此位选择擦除中止处理模式，详细内容请参照“39.7 挂起”。

39.2.14 闪存 P/E 状态寄存器（FPESTAT）

地址 007F FFCCh

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0		
—	—	—	—	—	—	—	—	PEERRST[7:0]								—	—
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		

位	符号	位名	功能	R/W
b7-b0	PEERRST [7:0]	P/E 错误状态位	00h: 没有错误 01h: 锁定位保护区的编程错误 02h: 由锁定位保护以外的原因引起的编程错误 11h: 锁定位保护区的擦除错误 12h: 由锁定位保护以外的原因引起的擦除错误 (上述以外: 保留)	R
b15-b8	—	保留位	读取值为“0”，写操作无效。	R

在内部 ROM 无效时，FPESTAT 寄存器的读数据为“0000h”，不能写。

通过复位或者将 FRESETR.FRESET 位置“1”，对 FPESTAT 寄存器进行初始化。

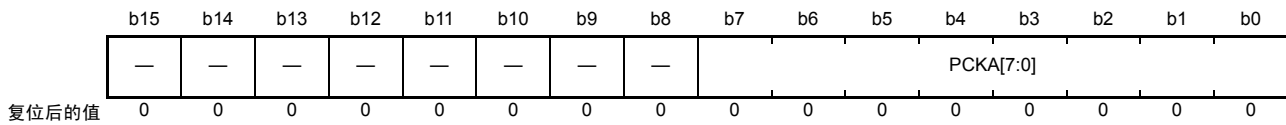
PEERRST[7:0] 位（P/E 错误状态位）

这些位表示在 ROM/E2 数据闪存的编程或者擦除处理过程中发生错误时的错误原因。

只在 FSTATR0.ERSERR 位或者 FSTATR0.PRGERR 位为“1”的状态下并且 FSTATR0.FRDIY 位变为“1”时，PEERRST[7:0] 位的值才有效。当 ERSERR 位和 PRGERR 位为“0”时，PEERRST[7:0] 位保持过去发生的错误原因的值。

39.2.15 外围时钟通知寄存器（PCKAR）

地址 007F FFE8h



位	符号	位名	功能	R/W
b7-b0	PCKA[7:0]	外围时钟通知位	在对 ROM/E2 数据闪存进行编程或者擦除时，这些位设定 FlashIF 时钟（FCLK）。	R/W
b15-b8	—	保留位	读写值都为“0”。	R/W

在内部 ROM 无效时，PCKAR 寄存器的读数据为“0000h”，不能写。
通过复位或者将 FRESETR.FRESET 位置“1”，对 PCKAR 寄存器进行初始化。

PCKA[7:0] 位（外围时钟通知位）

在对 ROM/E2 数据闪存进行编程或者擦除时，这些位设定 FlashIF 时钟（FCLK）。

必须在编程或者擦除前给 PCKA[7:0] 位设定 FCLK 的频率，然后发行外围时钟通知命令。不能在 ROM/E2 数据闪存进行编程或者擦除过程中更改频率。

如下进行设定值的计算：

- 将以 MHz 为单位表现的工作频率转换为 2 进制数并且写到 PCKA[7:0] 位。
以 FlashIF 时钟的工作频率为 29.9MHz 时的情况为具体例子，说明如下：
- 将 29.9 的第 1 位小数进行四舍五入。
- 将 30 转换为 2 进制数，给 PCKA[7:0] 位的高位设定“00h”，给低位设定“1Eh”（“0001 1110b”）。

注 . 如果将 PCKA[7:0] 位设定在 4MHz ~ 32MHz 的范围外，就不能对 ROM/E2 数据闪存发行改写命令。
如果给 PCKA[7:0] 位设定和 FCLK 不同的频率，就可能破坏 ROM/E2 数据闪存的数据。
即使有效地利用 PCKA[7:0] 位，在一定程度上改写时间也取决于频率。

39.3 ROM 的存储区结构

RX210 群的 ROM 由 512K 字节的用户区和 16K 字节的用户引导区构成，ROM 的存储区结构如图 39.2 所示。

必须注意：在进行读、编程或者擦除时，用户区的地址不同。

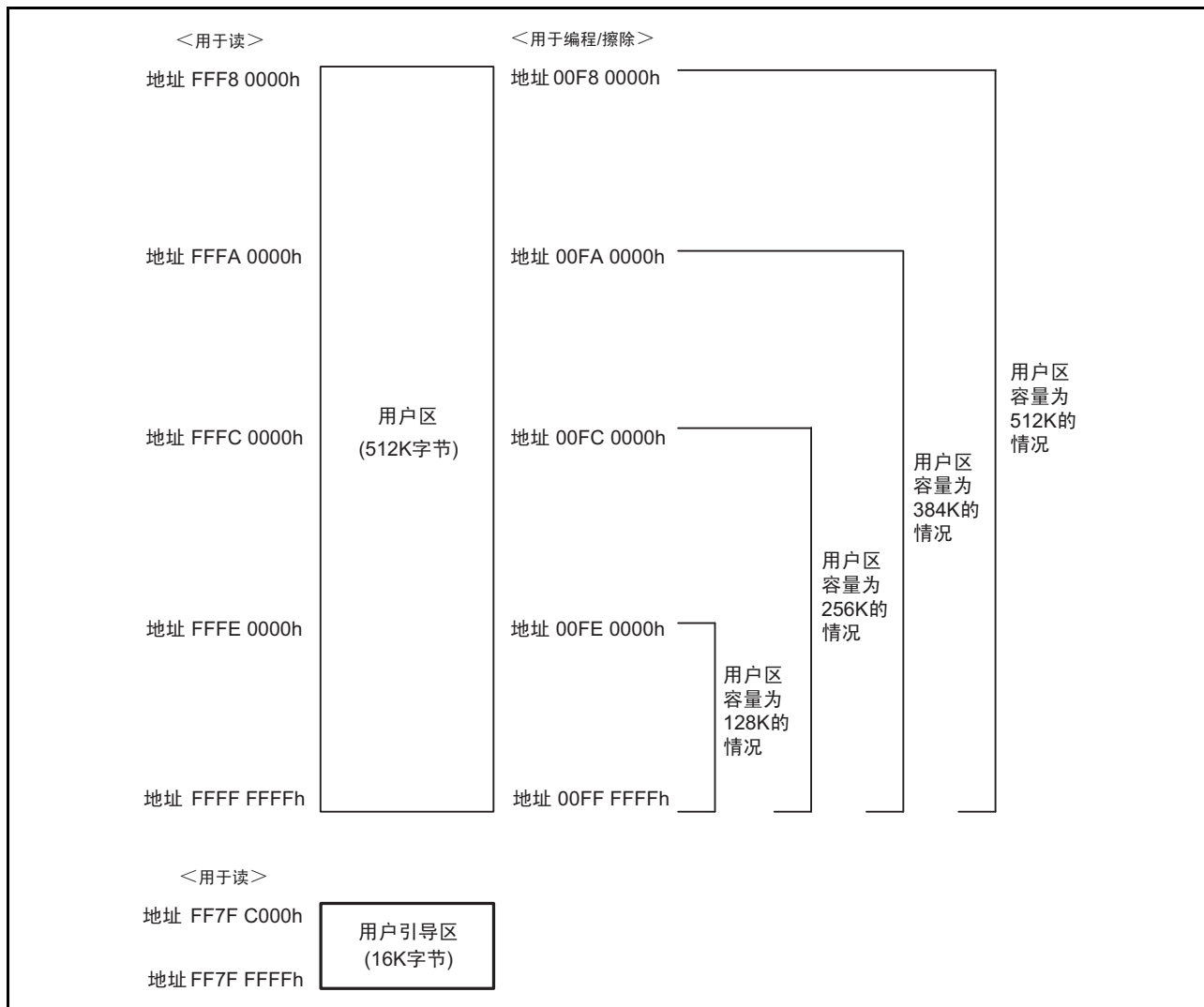


图 39.2 ROM 的存储区结构

39.4 块结构

用户区的擦除块结构如图 39.3 所示。

用户区根据 ROM 容量被分割为以下的块，以块为单位进行擦除。以低位地址为“00h”开始的 2/8/128 字节为单位进行编程。

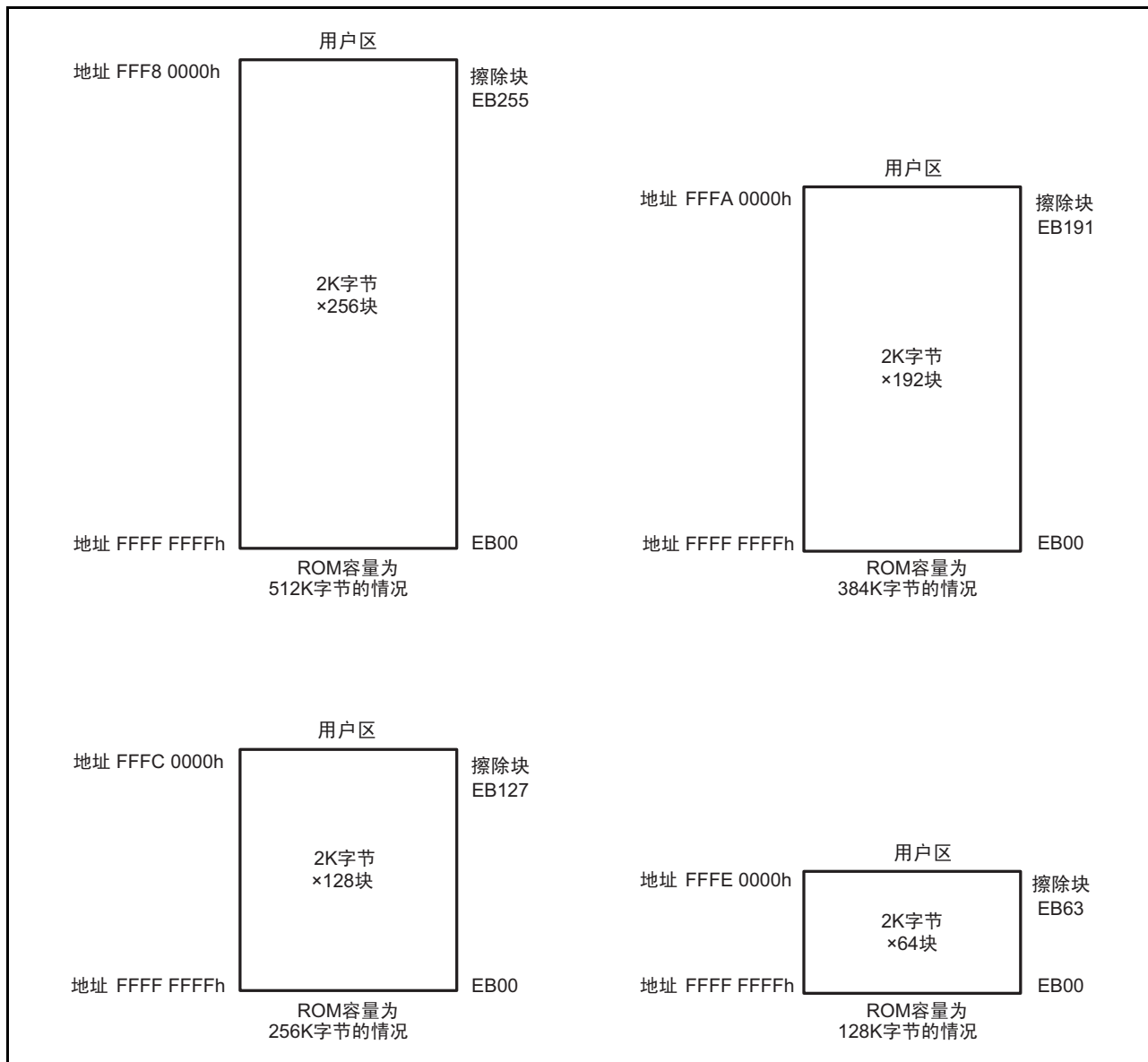


图 39.3 用户区的擦除块结构

39.5 ROM 相关的运行模式

有关 MD 引脚的设定值与 RX210 群的运行模式的关系，请参照“3. 运行模式”。

在引导模式、单芯片模式（内部 ROM 有效）或者内部 ROM 有效扩展模式中，能进行板上 ROM 的读、编程或者擦除。

在各模式中，能编程和擦除的区域、复位时的启动区域等不同，各模式的不同点如表 39.5 所示。

表 39.5 各模式的不同点

项目	引导模式	用户引导模式	单芯片模式（内部 ROM 有效）/ 内部 ROM 有效扩展模式
编程 / 擦除环境	板上编程		
能编程 / 擦除的区域	用户区 / 用户引导区	用户区	用户区
块分割擦除	○（注 1）	○	○
复位时的启动程序	引导程序	用户引导程序	用户程序

注 1. 有可能在启动时被全部擦除。此后能擦除特定块，详细内容请参照“39.10.4 ID 码保护（引导模式）”。

- 只能在引导模式中对用户引导区进行编程和擦除。
- 在引导模式中，主机能经由 SCI 对用户区、用户引导区或者数据区进行编程、擦除和读。
- 在引导模式中，通过程序使用内部 RAM。因此，不保持内部 RAM 的数据。
- 在用户引导模式中，能（从引导区启动后）从用户引导区启动，并且能使用任意接口对用户区或者数据区进行编程和读。

39.6 ROM 的编程 / 擦除

通过给编程 / 擦除的专用定序器（FCU）发行命令（FCU 命令），对 ROM 进行编程和擦除。FCU 有 5 种模式，通过在模式转移后发行编程和擦除命令，进行编程和擦除。

以下说明 ROM 的编程和擦除所需的模式转移和命令体系。这些内容在引导模式、用户引导模式、单芯片模式（内部 ROM 有效）和内部 ROM 有效扩展模式中通用。

39.6.1 FCU 的模式

FCU 有 5 种模式，通过写 FENTRYR 寄存器以及 FCU 命令进行模式的转移，FCU 的模式转移图如图 39.4 所示。

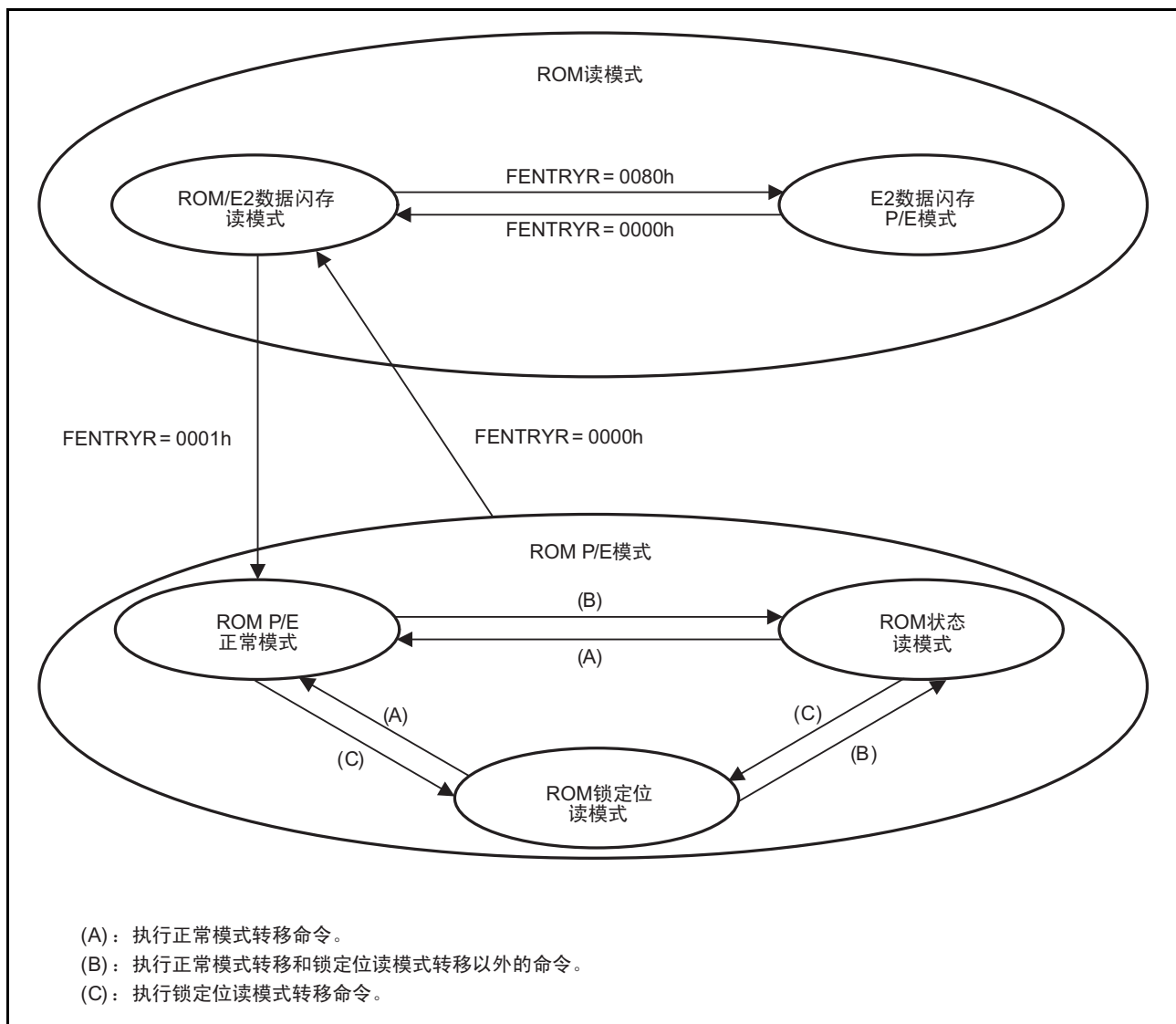


图 39.4 FCU 的模式转移图（ROM 相关）

39.6.1.1 ROM 读模式

ROM 读模式是高速读 ROM 的模式。如果对读地址进行读存取，就能进行 1 个 ICLK 周期的高速读取。
ROM 读模式有 ROM/E2 数据闪存读模式和 E2 数据闪存 P/E 模式。

(1) ROM/E2 数据闪存读模式

ROM/E2 数据闪存读模式是能读 ROM 和 E2 数据闪存的模式，不接受 FCU 命令。如果将 FENTRYR.FENTRY0 位和 FENTRYR.FENTRYD 位都置“0”，就转移到此模式。

(2) E2 数据闪存 P/E 模式

E2 数据闪存 P/E 模式是对 E2 数据闪存进行编程和擦除的模式，能高速读 ROM。此模式接受 E2 数据闪存的 FCU 命令，但是不接受 ROM 的 FCU 命令。如果将 FENTRYR.FENTRY0 位置“0”并且将 FENTRYR.FENTRYD 位置“1”，就转移到此模式。

有关 E2 数据闪存 P/E 模式的详细内容，请参照“40. E2 数据闪存（保存数据的闪存）”的“40.6.1 FCU 的模式”。

39.6.1.2 ROM P/E 模式

ROM P/E 模式是对 ROM 进行编程和擦除的模式，不能高速读 ROM。如果对读地址进行读存取，就发生 ROM 存取违反，FCU 进入命令锁定状态（参照“39.8.2 错误保护”）。

ROM P/E 模式有 ROM P/E 正常模式、ROM 状态读模式和 ROM 锁定位读模式共 3 种。

(1) ROM P/E 正常模式

ROM P/E 正常模式是在对 ROM 进行编程或者擦除时最先转移的模式。如果在 ROM 读模式中将 FENTRYR.FENTRYD 位置“0”并且将 FENTRYR.FENTRY0 位置“1”，或者在 ROM P/E 模式中接受正常模式转移命令，就转移到此模式。能接受的命令如表 39.8 所示。

如果在 FENTRYR.FENTRY0 位为“1”的状态下对编程 / 擦除地址进行读存取，就产生 ROM 的存取违反，FCU 进入命令锁定状态（参照“39.8.2 错误保护”）。

(2) ROM 状态读模式

ROM 状态读模式是能读 ROM 状态的模式。如果在 ROM P/E 模式中接受正常模式转移和锁定位读模式转移以外的命令，就转移到此模式。

FSTATR0.FRDY 位为“0”的状态以及发生错误后的命令锁定状态也是 ROM 状态读模式中的状态。能接受的命令如表 39.8 所示。

如果在 FENTRYR.FENTRY0 位为“1”的状态下对相应的编程 / 擦除地址进行读存取，就能读 FSTATR0 寄存器的值。

(3) ROM 锁定位读模式

ROM 锁定位读模式是通过读 ROM 来读锁定位的模式。如果在 ROM P/E 模式中接受锁定位读模式转移命令，就转移到此模式。能接受的命令如表 39.8 所示。

如果在 FENTRYR.FENTRY0 位为“1”的状态下对相应的编程 / 擦除地址进行读存取，读数据的全部位就为存取对象的擦除块的锁定位值。

39.6.2 FCU 命令一览表

FCU 命令有 FCU 模式转移命令和编程 / 擦除命令。ROM 能使用的 FCU 命令一览表如表 39.6 所示。

表 39.6 FCU 命令一览表（ROM 相关）

命令	功能
P/E 正常模式转移	转移到正常模式（参照“39.6.3 FCU 的模式和命令的关系”）。
状态读模式转移	转移到状态读模式（参照“39.6.3 FCU 的模式和命令的关系”）。
锁定位读模式转移（锁定位读 1）	转移到锁定位读模式（参照“39.6.3 FCU 的模式和命令的关系”）。
外围时钟通知	设定外围时钟的频率。
编程	对 ROM 进行编程（以 2/8/128 字节为单位）。
块擦除	擦除 ROM（以块为单位，同时擦除锁定位）。
P/E 挂起	中止编程和擦除。
P/E 恢复	重新开始编程和擦除。
状态寄存器清除	清除 FSTATR0.ILGLERR 位、FSTATR0.ERSERR 位和 FSTATR0.PRGERR 位以及解除命令锁定状态。
锁定位读 2/ 空白检查	读指定的擦除块的锁定位（将锁定位反映到 FSTATR1.FLOCKST 位）/E2 数据闪存的空白检查。
锁定位编程	对指定的擦除块的锁定位进行编程。

锁定位读 2 命令兼用 E2 数据闪存的空白检查命令。如果对 E2 数据闪存发行锁定位读 2 命令，就进行 E2 数据闪存的空白检查（参照“40. E2 数据闪存（保存数据的闪存）”）。

通过对 ROM 的编程 / 擦除地址进行写存取来给 FCU 发行命令。FCU 命令的格式如表 39.7 所示。如果在 FCU 的特定条件下进行表 39.7 所示的写存取，FCU 就进行各命令对应的处理。

有关 FCU 的特定条件和命令的使用方法，请分别参照“39.6.3 FCU 的模式和命令的关系”和“39.6.4 FCU 命令的使用方法”。

表 39.7 FCU 命令的格式

命令	总线周期数	第 1 周期		第 2 周期		第 3 周期		第 4 ~ 第 5 周期		第 6 周期		第 7 ~ N+2 周期		第 n+3 周期	
		地址	数据	地址	数据	地址	数据	地址	数据	地址	数据	地址	数据	地址	数据
P/E 正常模式转移	1	RA	FFh	—	—	—	—	—	—	—	—	—	—	—	—
状态读模式转移	1	RA	70h	—	—	—	—	—	—	—	—	—	—	—	—
锁定位读模式转移 (锁定位读 1)	1	RA	71h	—	—	—	—	—	—	—	—	—	—	—	—
外围时钟通知	6	RA	E9h	RA	03h	WA	0F0Fh	RA	0F0Fh	RA	D0h	—	—	—	—
编程 (2 字节编程: N=1)	4	RA	E8h	RA	01h	WA	WDn	—	—	—	—	—	—	RA	D0h
编程 (8 字节编程: N=4)	7	RA	E8h	RA	04h	WA	WDn	RA	WDn	RA	WDn	—	—	RA	D0h
编程 (128 字节编程: N=64)	67	RA	E8h	RA	40h	WA	WDn	RA	WDn	RA	WDn	RA	WDn	RA	D0h
块擦除	2	RA	20h	BA	D0h	—	—	—	—	—	—	—	—	—	—
P/E 挂起	1	RA	B0h	—	—	—	—	—	—	—	—	—	—	—	—
P/E 恢复	1	RA	D0h	—	—	—	—	—	—	—	—	—	—	—	—
状态寄存器清除	1	RA	50h	—	—	—	—	—	—	—	—	—	—	—	—
锁定位读 2	2	RA	71h	BA	D0h	—	—	—	—	—	—	—	—	—	—
锁定位编程	2	RA	77h	BA	D0h	—	—	—	—	—	—	—	—	—	—

地址列 RA: ROM 的编程 / 擦除地址

当ROM容量为128K字节并且FENTRYR.FENTRY0位为“1”时,为00FE 0000h~00FF FFFFh的任意地址。

当ROM容量为256K字节并且FENTRYR.FENTRY0位为“1”时,为00FC 0000h~00FF FFFFh的任意地址。

当ROM容量为384K字节并且FENTRYR.FENTRY0位为“1”时,为00FA 0000h~007F FFFFh的任意地址。

当ROM容量为512K字节并且FENTRYR.FENTRY0位为“1”时,为00F8 0000h~00F7 FFFFh的任意地址。

WA: ROM 的编程目标地址

编程数据 128 字节的起始地址

BA: ROM 的擦除块地址

擦除对象块内的任意地址 (由编程 / 擦除地址指定)

数据列 WDn: 编程数据的第 n 个字 (n=1 ~ 64)

39.6.3 FCU 的模式和命令的关系

FCU 的各模式能接受的 FCU 命令是固定的，而且能接受的命令也因这些模式的 FCU 状态而不同。

需要在进行 FCU 的模式转移后确认 FCU 的状态，然后发行 FCU 命令。

FCU 的模式 / 状态和能接受的命令如表 39.8 所示。如果发行不能接受的命令，FCU 就进入命令锁定状态（参照“39.8.2 错误保护”）。

必须在转移到能接受的模式后，确认 FSTATR0.FRDY 位、FSTATR0.ILGLERR 位、FSTATR0.ERSERR 位、FSTATR0.PRGERR 位和 FSTATR1.FCUEERR 位的值，然后发行 FCU 命令。能根据 FASTAT.CMDLK 位的值确认有没有发生错误。FASTAT.CMDLK 位的值是 FSTATR0.ILGLERR 位、FSTATR0.ERSERR 位、FSTATR0.PRGERR 位和 FSTATR1.FCUEERR 位的值的逻辑或。

表 39.8 FCU 的模式 / 状态和能接受的命令的关系（ROM P/E 模式）

	P/E 正常模式			状态读模式									锁定位读模式		
	编程挂起中	擦除挂起中	其他状态	编程 / 擦除处理中	擦除挂起中的编程处理中	编程 / 擦除中止处理中	锁定位读 2 处理中	编程挂起中	擦除挂起中	命令锁定状态 (FRDY=0)	命令锁定状态 (FRDY=1)	其他状态	编程挂起中	擦除挂起中	其他状态
FSTATR0.FRDY 位	1	1	1	0	0	0	0	1	1	0	1	1	1	1	1
FSTATR0.SUSRDY 位	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
FSTATR0.ERSSPD 位	0	1	0	0	1	0/1	0	0	1	0/1	0/1	0	0	1	0
FSTATR0.PRGSPD 位	1	0	0	0	0	0/1	0	1	0	0/1	0/1	0	1	0	0
FASTAT.CMDLK 位	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
正常模式转移	○	○	○	×	×	×	×	○	○	×	×	○	○	○	○
状态读模式转移	○	○	○	×	×	×	×	○	○	×	×	○	○	○	○
锁定位读模式转移（锁定位读 1）	○	○	○	×	×	×	×	○	○	×	×	○	○	○	○
外围时钟通知	×	×	○	×	×	×	×	×	×	×	×	○	×	×	○
编程	×	Δ	○	×	×	×	×	×	Δ	×	×	○	×	Δ	○
块擦除	×	×	○	×	×	×	×	×	×	×	×	○	×	×	○
P/E 挂起	×	×	×	○	×	×	×	×	×	×	×	×	×	×	×
P/E 恢复	○	○	×	×	×	×	×	○	○	×	×	×	○	○	×
状态寄存器清除	○	○	○	×	×	×	×	○	○	×	○	○	○	○	○
锁定位读 2	○	○	○	×	×	×	×	○	○	×	×	○	○	○	○
锁定位编程	×	Δ	○	×	×	×	×	×	Δ	×	×	○	×	Δ	○

○：能接受、Δ：只能接受对非擦除中止块进行的编程、×：不能接受

39.6.4 FCU 命令的使用方法

FCU 命令有转移 FCU 模式的命令、实际对 ROM 进行编程和擦除的命令、错误处理命令和挂起 / 恢复命令。以下说明各命令，有关能接受各命令的模式和状态，请参照“39.6.3 FCU 的模式和命令的关系”。

39.6.4.1 模式的转移

在此说明模式转移的相关命令，各模式转移的关系请参照图 39.4。

(1) ROM P/E 模式的转移方法

要执行 ROM 相关的 FCU 命令时，需要转移到 ROM P/E 模式。

为了转移到 ROM P/E 模式，将要进行编程或者擦除的 ROM 地址对应的 FENTRYR.FENTRY0 位置“1”。

在进行编程或者擦除时，必须以字节为单位给 FWEPROR 寄存器写“01h”，进入能进行编程和擦除的状态（参照“39.2.1 闪存编程 / 擦除保护寄存器（FWEPROR）”）。

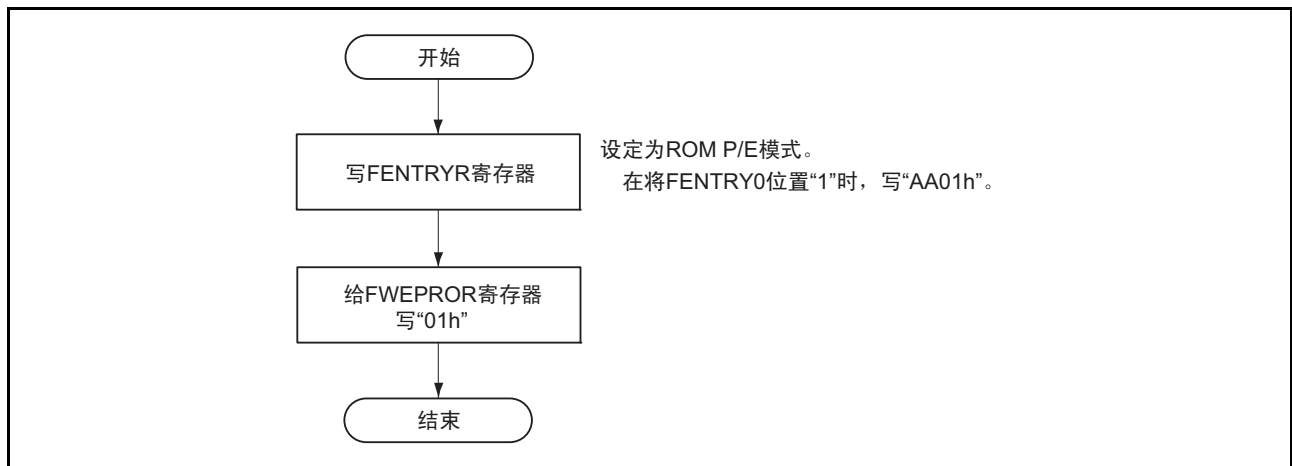


图 39.5 ROM P/E 模式的转移流程

(2) ROM 读模式的转移方法

要高速读 ROM 时，需要将 FENTRYR.FENTRY0 位置“0”，将 FCU 设定为 ROM 读模式。

还需要通过字节存取给 FWEPROR 寄存器写“02h”，进入不能编程和擦除的状态（参照“39.2.1 闪存编程 / 擦除保护寄存器（FWEPROR）”）。

必须在结束 FCU 的命令处理并且 FCU 未检测到错误的状态下从 ROM P/E 模式转移到 ROM 读模式。

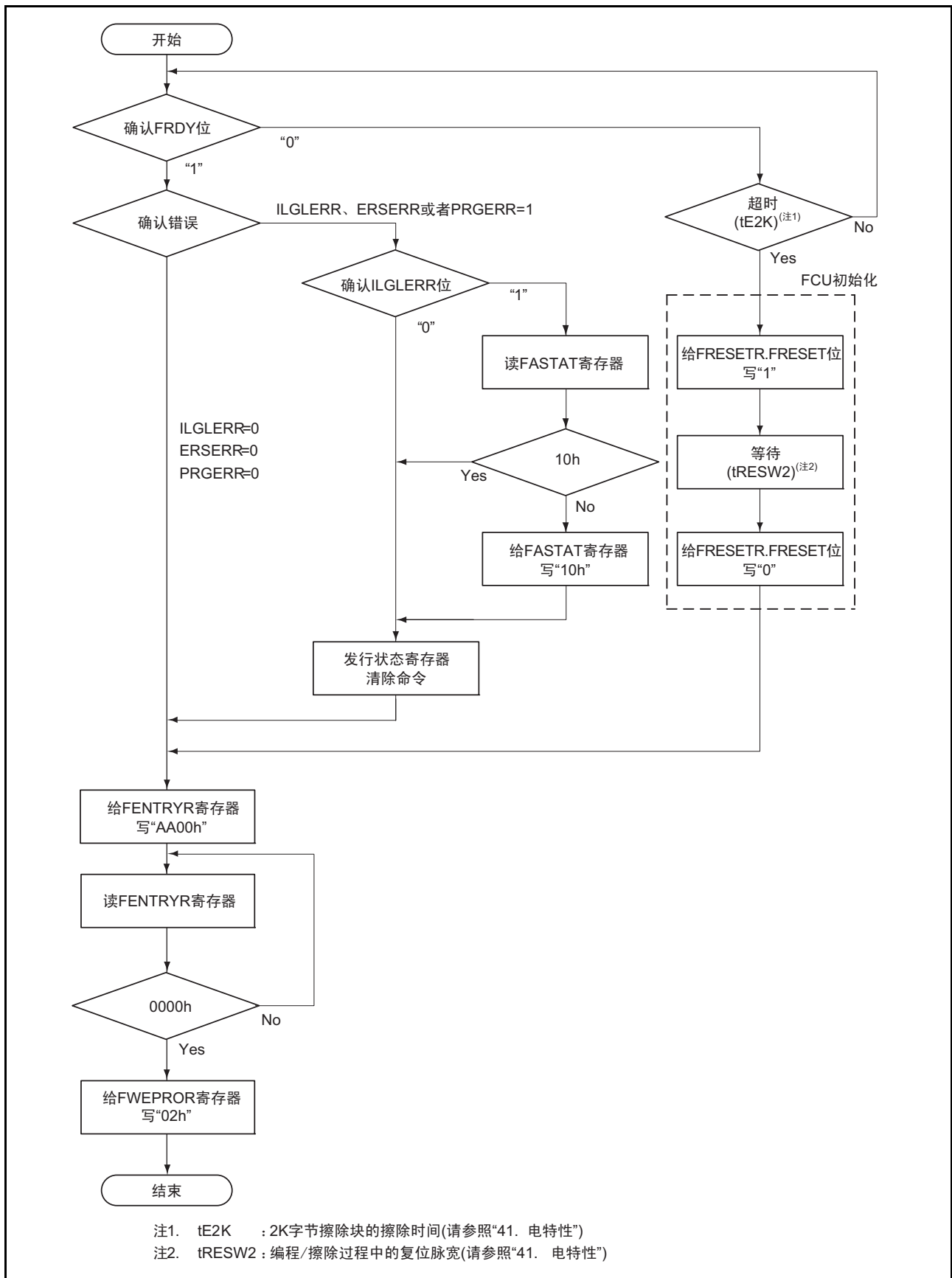


图 39.6 ROM 读模式的转移流程

(3) ROM P/E 正常模式的转移方法

向 ROM P/E 正常模式的转移方法有：在 ROM 读模式中设定 FENTRYR 寄存器的方法（参照“39.6.1 FCU 的模式”）以及在 ROM P/E 模式中发行正常模式转移命令（图 39.7）的方法。通过字节存取给 ROM 的编程 / 擦除地址写“FFh”，执行正常模式转移命令。

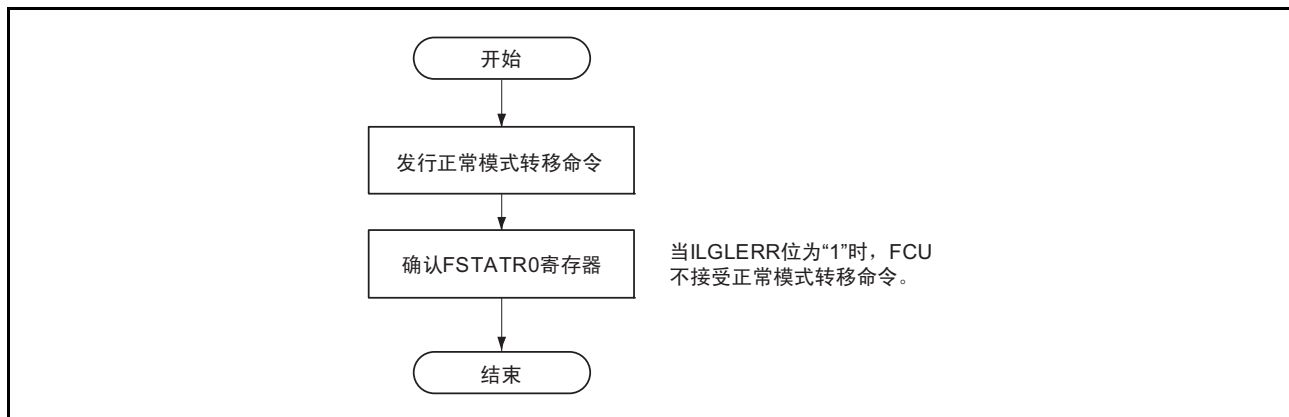


图 39.7 ROM P/E 正常模式的转移流程

(4) ROM 状态读模式的转移方法

如果发行正常模式转移和锁定位读模式转移以外的 FCU 命令，FCU 就转移到 ROM 状态读模式。也能通过发行状态读模式转移命令，转移到 ROM 状态读模式。FSTATR0 寄存器的确认例子如图 39.8 所示。在此例中，通过发行状态读模式转移命令，转移到 ROM 状态读模式，然后对 ROM 的编程 / 擦除地址进行读存取，确认 FSTATR0 的内容。

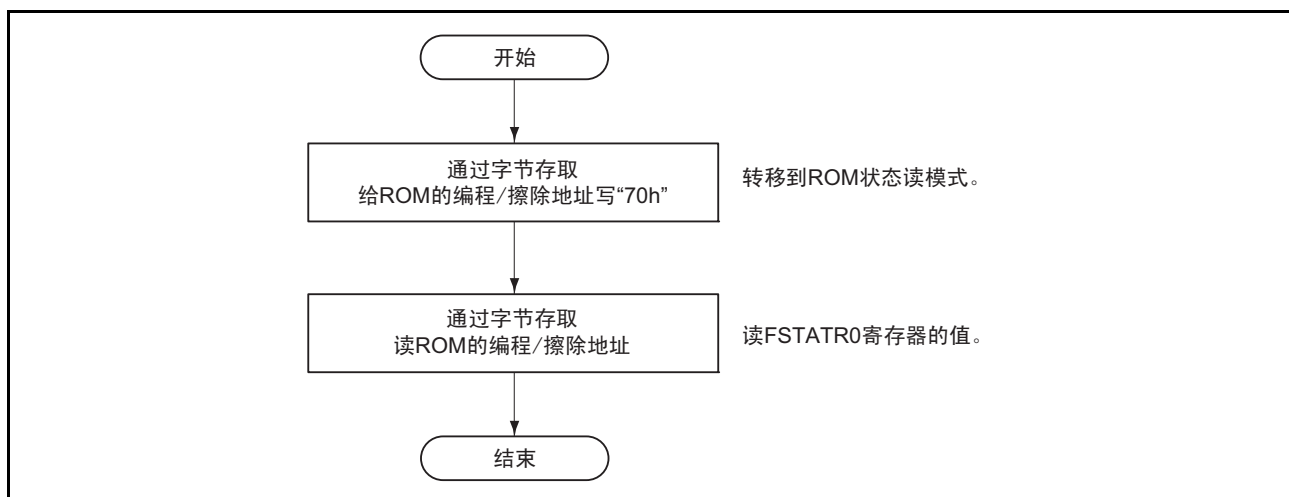


图 39.8 ROM 状态读模式的转移流程以及状态的确认方法

(5) ROM 锁定位读模式的转移方法

在 FMODR.FRDM 位为“0”（存储区读方式）时，通过发行锁定位读模式转移命令（锁定位读 1），转移到 ROM 锁定位读模式。如果在转移到 ROM 锁定位读模式后对 ROM 的编程 / 擦除地址进行读存取，就读与存取对象对应的擦除块的锁定位并且复制到读数据的全部位（图 39.9）。

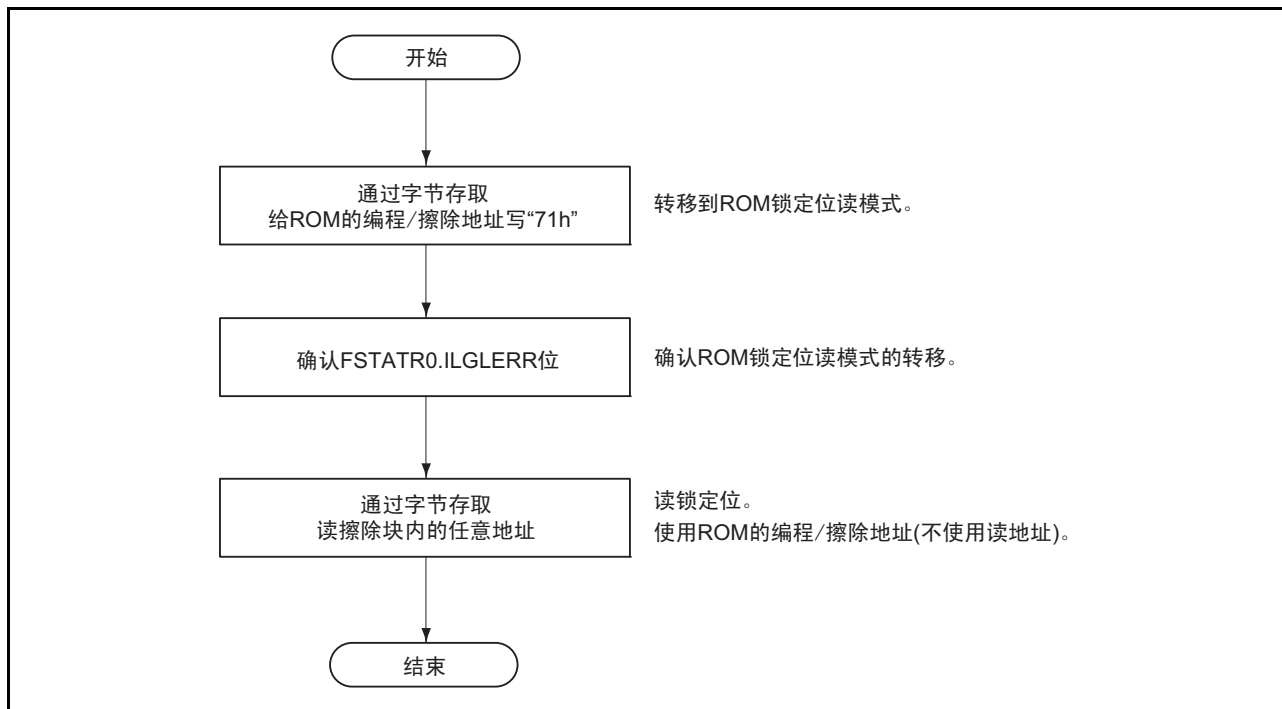


图 39.9 ROM 锁定位读模式的转移流程以及锁定位的读法

39.6.4.2 编程 / 擦除的方法步骤

在此说明 ROM 的编程 / 擦除流程。有关 FCU 命令的接受条件，请参照“39.6.3 FCU 的模式和命令的关系”。FCU 命令的概略流程如图 39.10 所示。

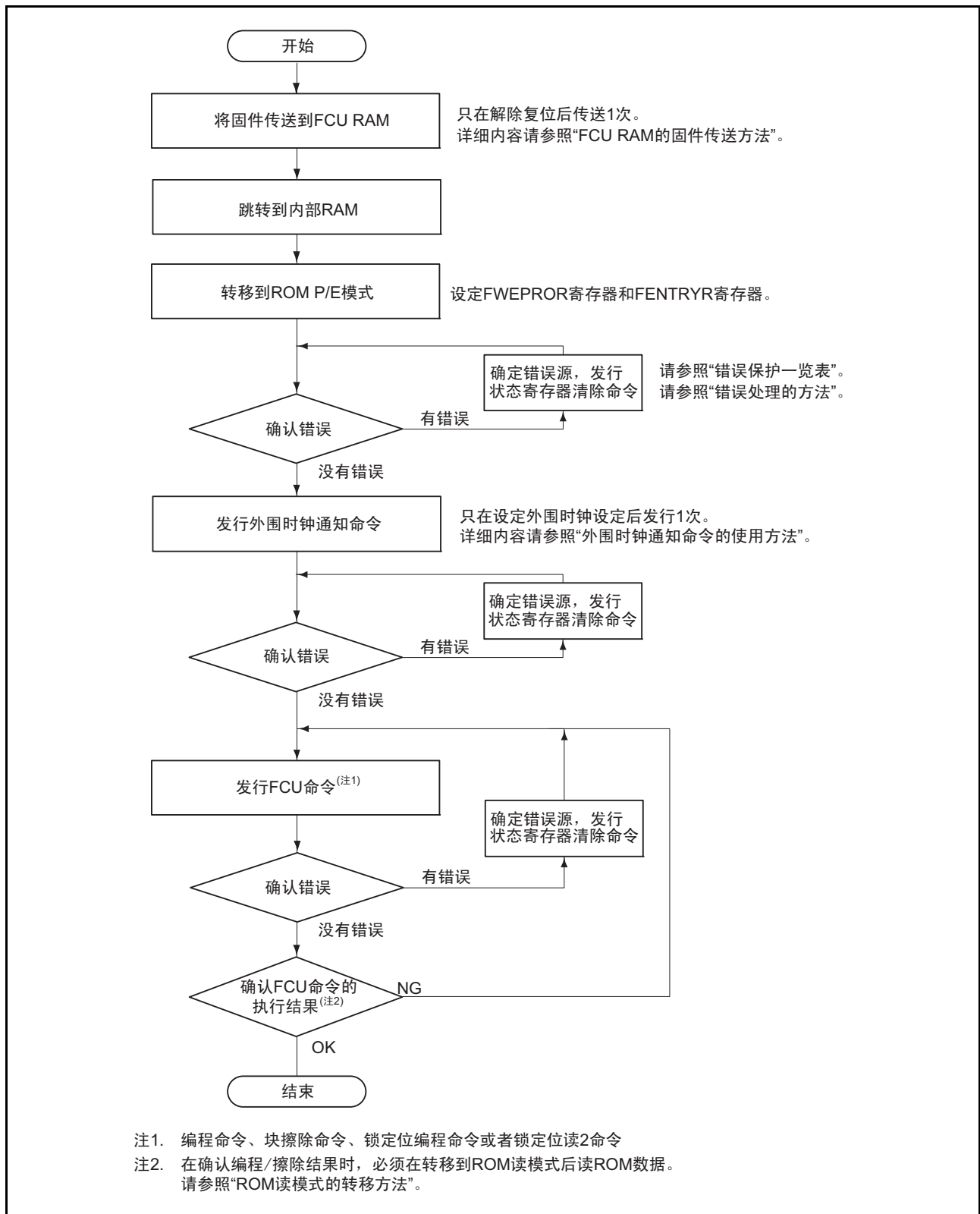


图 39.10 编程 / 擦除处理的概略流程

(1) FCU RAM 的固件传送方法

要使用 FCU 命令时，需要将 FCU 的固件保存到 FCU RAM。因为在启动芯片时没有将 FCU 的固件保存到 FCU RAM，所以需要将保存在 FCU 固件区的 FCU 固件复制到 FCU RAM。当 FSTATR1.FCUERR 位为“1”时，保存在 FCU RAM 的固件就可能被破坏，因此需要对 FCU 进行复位并且重新复制 FCU 固件。

FCU RAM 的固件传送流程如图 39.11 所示。在给 FCU RAM 写数据时，必须将 FENTRYR 寄存器置“0000h”，使 FCU 停止。有关 DMAC 设定方法的详细内容，请参照“16. DMA 控制器 (DMACA) ”。

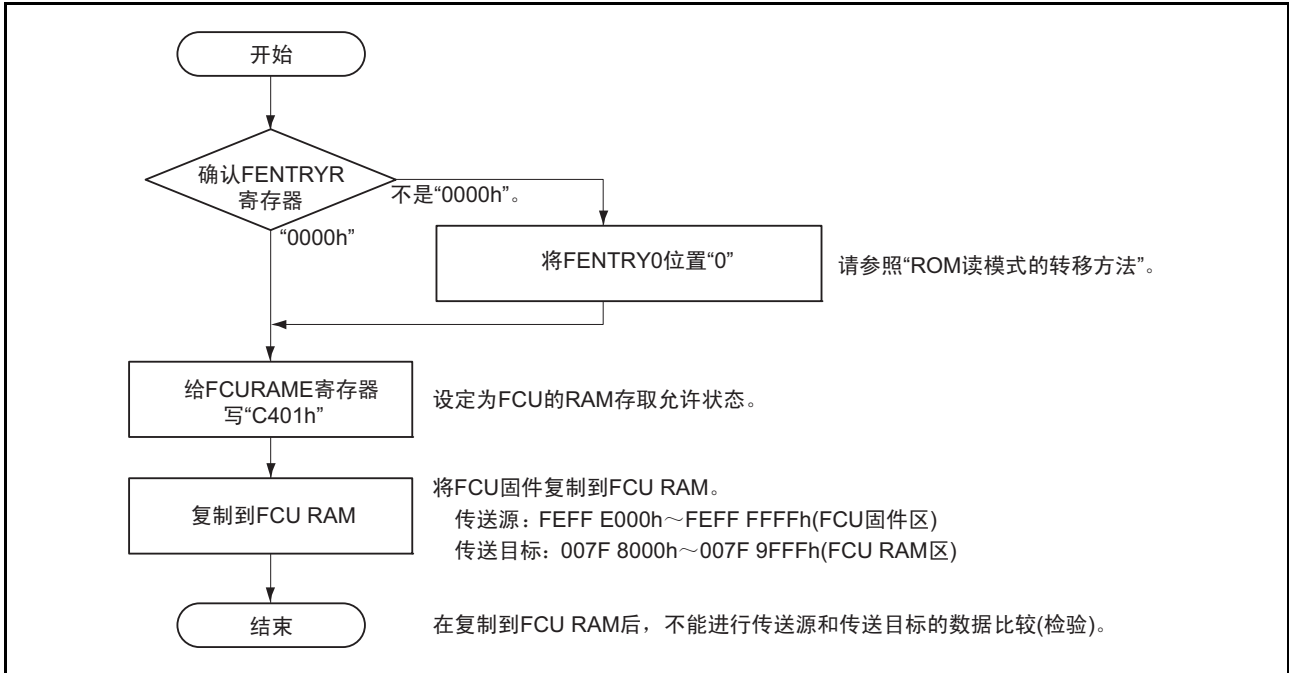


图 39.11 FCU RAM 的固件传送流程

(2) 向内部 RAM 的跳转

在对 ROM 进行编程或者擦除时，因为不对 ROM 取指令，所以需要转移到 ROM 以外的区域。必须在将所需的指令码复制到内部 RAM 后跳转到内部 RAM。

(3) ROM P/E 模式的转移

需要设定 FENTRYR.FENTRY0 位和 FWEPROR 寄存器，并且将 FCU 设定为 ROM P/E 模式。详细内容请参照“39.6.4.1 (1) ROM P/E 模式的转移方法”。

(4) 外围时钟通知命令的使用方法

需要给 PCKAR 寄存器设定在对 ROM 进行编程或者擦除前使用的 FlashIF 时钟（FCLK）的频率。能设定的频率范围是 4MHz ~ 32MHz。如果频率不在此范围内，FCU 就检测到错误，进入命令锁定状态（参照“39.8.2 错误保护”）。另外，如果将 PCKAR 寄存器的 PCKA[7:0] 位设定在 4MHz ~ 32MHz 范围外，就不能对 ROM/E2 数据闪存发行改写命令。

在设定 PCKAR 寄存器后使用外围时钟通知命令。在外围时钟通知命令的第 1 周期，通过字节存取给 ROM 的编程 / 擦除地址写“E9h”；在第 2 周期，通过字节存取给 ROM 的编程 / 擦除地址写“03h”。在命令的第 3 周期 ~ 第 5 周期，通过字存取进行写操作。此时，起始地址必须使用调整为以 4 字节为边界的地址。在通过 3 次字存取给 ROM 的编程 / 擦除地址写数据“0F0Fh”后，如果在第 6 周期通过字节存取给 ROM 的编程 / 擦除地址写“D0h”，FCU 就开始进行外围时钟的频率设定处理。能通过 FSTATR0.FRDY 位确认设定的结束。

第 1 周期 ~ 第 6 周期中能指定的地址因 FENTRYR.FENTRY0 位的设定而不同，必须指定与 FENTRYR.FENTRY0 位对应的地址。如果因 FENTRYR.FENTRY0 位和指定地址的错误组合而发行命令，FCU 就检测到错误，进入命令锁定状态（参照“39.8.2 错误保护”）。

如果在复位后不更改正在使用的外围时钟的设定，此设定就只需执行 1 次而且对后续的 FCU 命令有效。

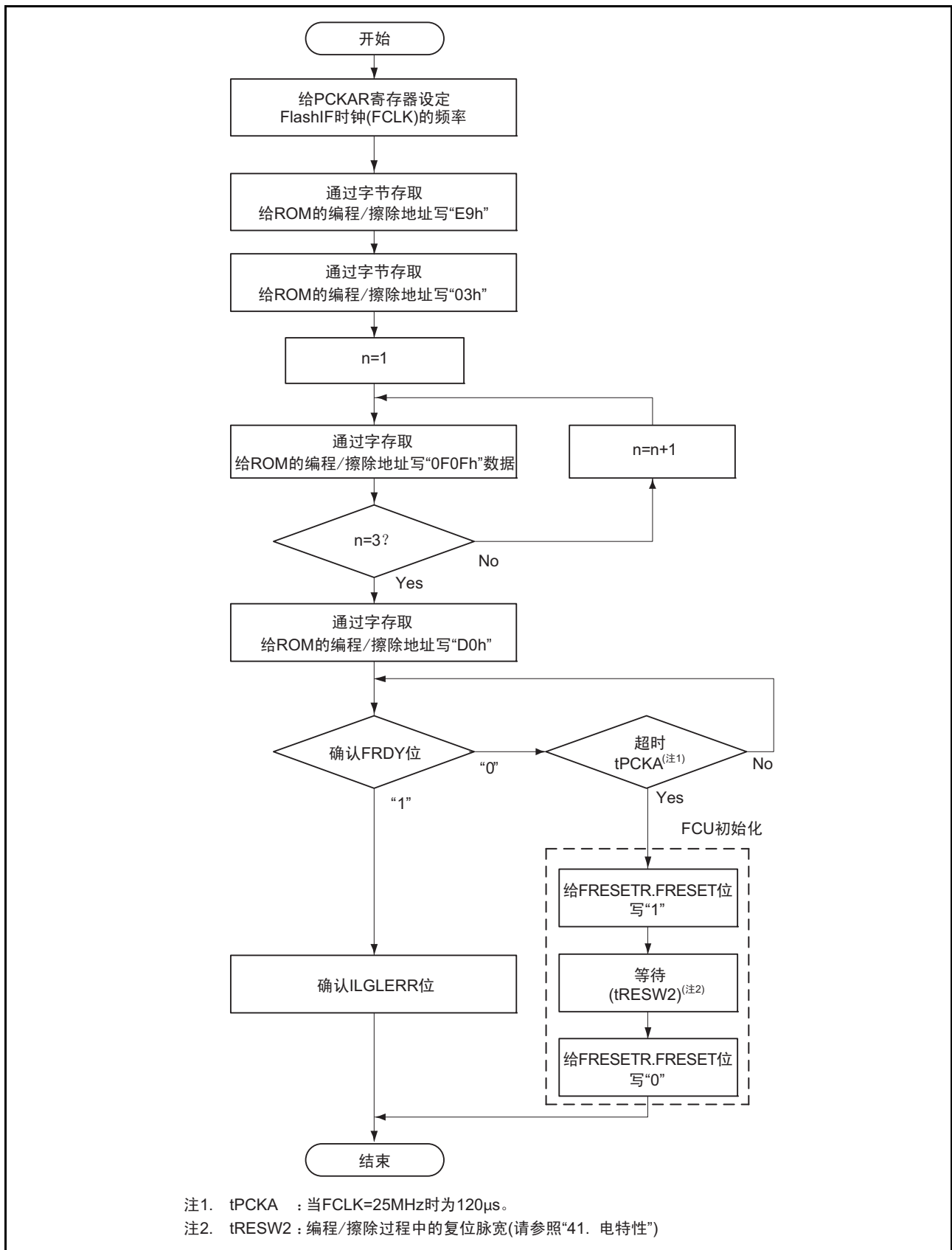


图 39.12 外围时钟通知命令的使用方法

(5) 编程方法

使用编程命令对 ROM 进行数据编程。

在编程命令的第 1 周期，通过字节存取给 ROM 的编程 / 擦除地址写“E8h”；在第 2 周期，如果为 2 字节编程，就通过字节存取给 ROM 的编程 / 擦除地址写“01h”，如果为 8 字节编程，就通过字节存取给 ROM 的编程 / 擦除地址写“04h”，如果为 128 字节编程，就通过字节存取给 ROM 的编程 / 擦除地址写“40h”。在第 3 周期的存取中，必须通过字存取给编程对象区的起始地址写编程数据。此时，起始地址必须使用调整为以 128 字节为边界的地址。在第 4 周期～第 66（128 字节编程的情况）周期，必须通过 63 次字存取给 ROM 的编程 / 擦除地址写编程数据。如果在第 67 周期通过字节存取给 ROM 的编程 / 擦除地址写“D0h”，FCU 就开始对 ROM 进行编程处理。能通过 FSTATR0.FRDY 位确认编程的结束。

第 1 周期～第 67 周期中能指定的地址因 FENTRYR.FENTRY0 位的设定而不同，必须指定与 FENTRYR.FENTRY0 位对应的地址。如果因 FFENTRYR.FENTRY0 位和指定地址的错误组合而发行命令，FCU 就检测到错误，进入命令锁定状态（参照“39.8.2 错误保护”）。

如果在第 3 周期～第 66 周期中存取的区域包括不需要编程的地址，就必须将该地址的编程数据置“FFFFh”。如果要在将通过锁定位进行的保护设定为无效后进行编程，就必须将 FPROTR.FPROTCN 位置“1”。

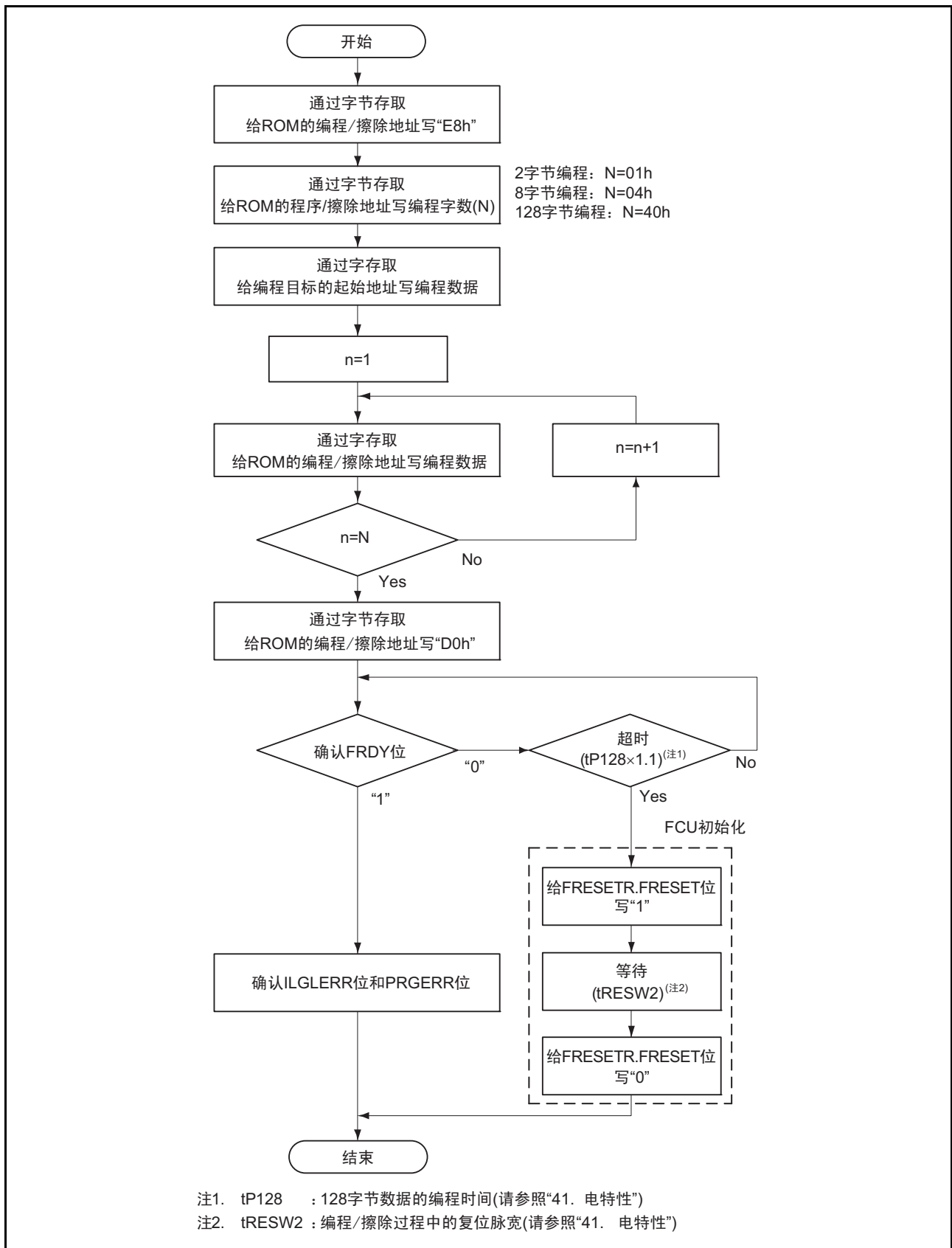


图 39.13 ROM 的编程方法

(6) 擦除方法

使用块擦除命令来擦除 ROM。

在块擦除命令的第 1 周期，通过字节存取给 ROM 的编程 / 擦除地址写“20h”。如果在第 2 周期通过字节存取给擦除对象块内的任意地址写“D0h”，FCU 就开始对 ROM 进行擦除处理。能通过 FSTATR0.FRDY 位确认擦除的结束。如果通过 CPU 读擦除状态的 ROM，就以 32 位为单位读到“FFFF FFFFh”。

如果要在将通过锁定位进行的保护设定为无效后进行擦除，就必须在将 FPROTR.FPROTCN 位置位后进行擦除。

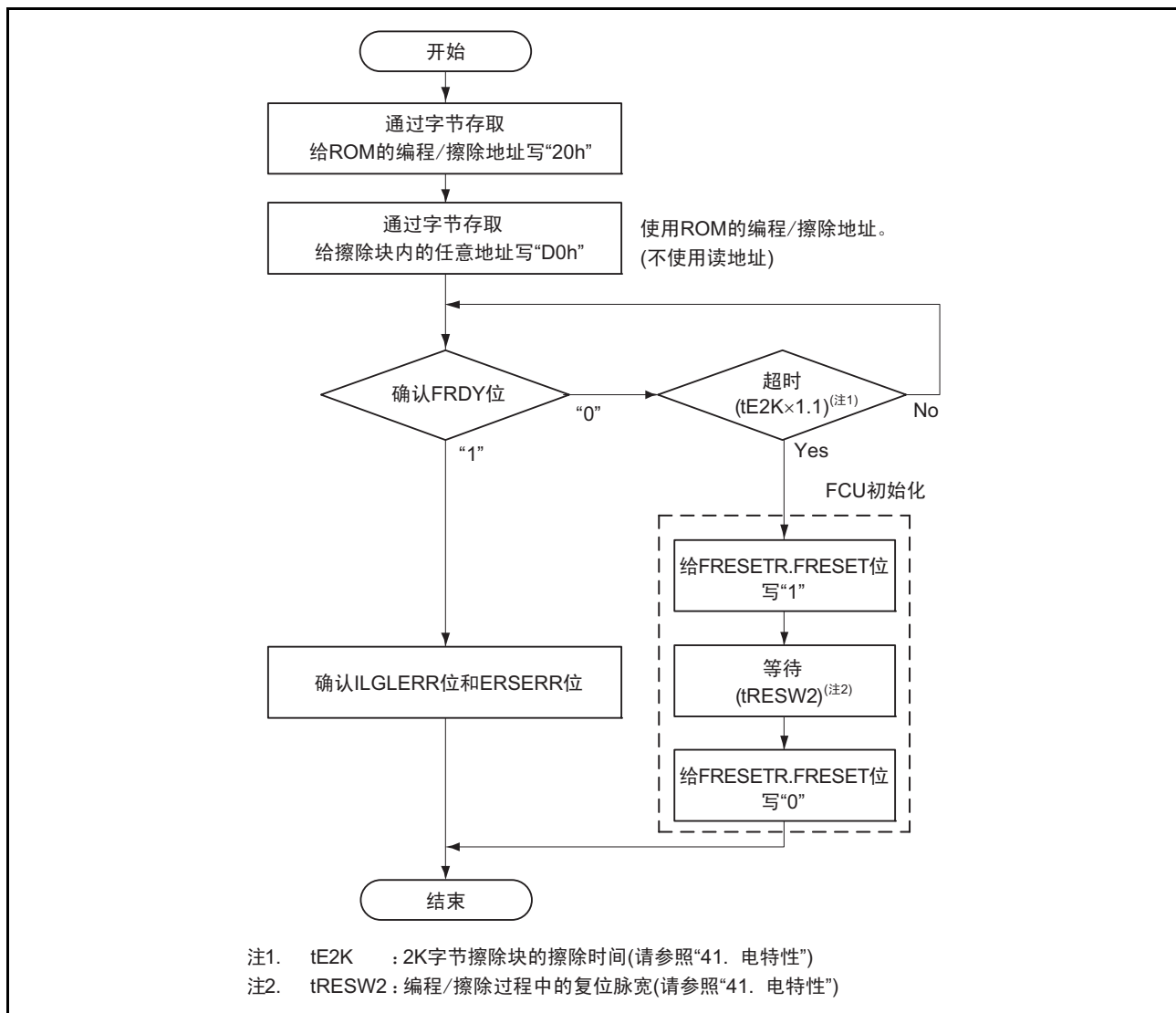


图 39.14 ROM 的擦除方法

(7) 锁定位的编程 / 擦除方法

用户区的各擦除块内置锁定位。使用锁定位编程命令对锁定位进行编程。在锁定位编程命令的第 1 周期，通过字节存取给 ROM 的编程 / 擦除地址写“77h”。如果在第 2 周期通过字节存取给要进行锁定位编程的擦除块内的任意地址写“D0h”，FCU 就开始进行锁定位的编程处理。能通过 FSTATR0.FRDY 位确认编程的结束。

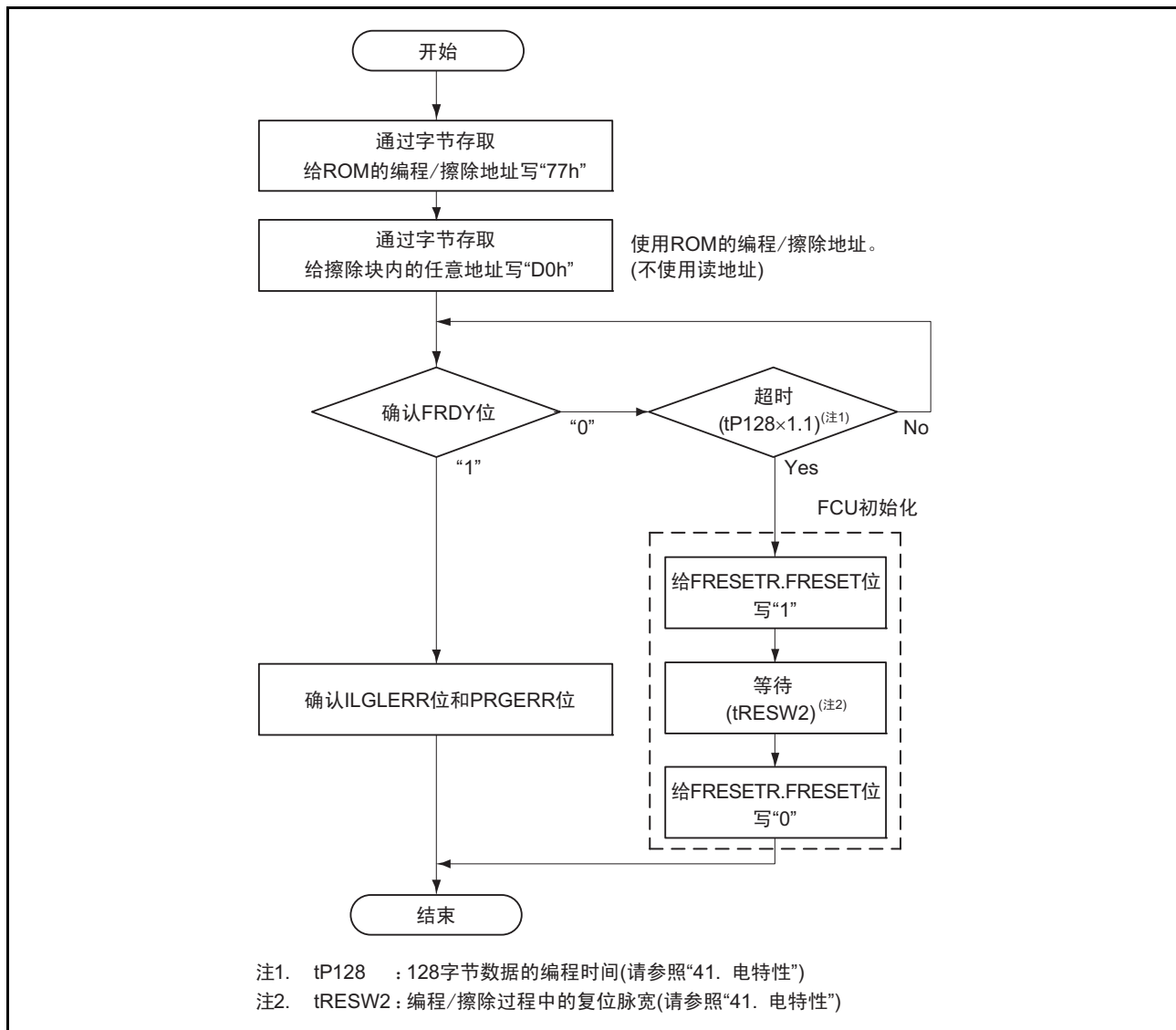


图 39.15 锁定位的编程设定方法

使用块擦除命令来擦除锁定位。

在 FPROTR.FPROTCN 位为“0”的状态下，不能擦除锁定位为“0”的擦除块。要擦除锁定位时，必须在将 FPROTCN 位置“1”的状态下发行块擦除命令。如果使用块擦除命令，擦除块内的全部数据就被擦除，而不能只擦除锁定位。

(8) 锁定位的读法

锁定位的读法有存储区读方式和寄存器读方式。

寄存器读方式（FMODR.FRDM位为“1”）使用锁定读2命令，对要读锁定位的擦除块的编程/擦除地址发行锁定读2命令。如果在锁定读2命令的第1周期通过字节存取写“71h”，在第2周期通过字节存取写“D0h”，就将对应的擦除块的锁定位复制到FSTATR1.FLOCKST位。

通过转移到锁定位读模式并且读ROM的编程/擦除地址来执行存储区读方式（FMODR.FRDM位为“0”）。有关详细内容，请参照“39.6.4.1 (5) ROM 锁定位读模式的转移方法”。

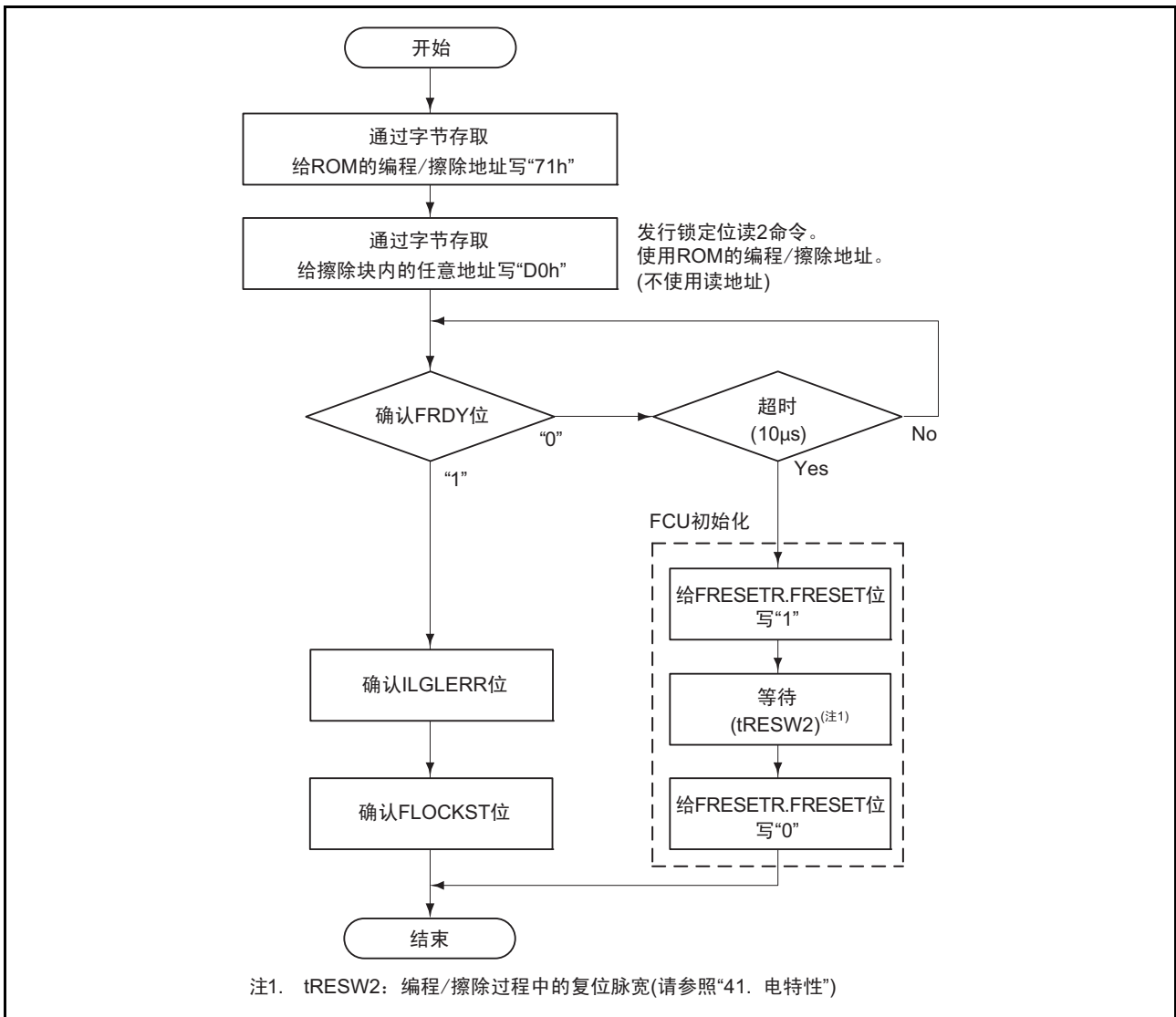


图 39.16 通过寄存器读方式读锁定位的方法

39.6.4.3 错误的处理方法

以下说明发生错误时的处理方法，各种错误的内容请参照“39.8 保护”。

(1) 闪存状态寄存器 0（FSTATR0）的确认方法

FSTATR0 寄存器的确认方法有直接读 FSTATR0 寄存器的方法和在 ROM 状态读模式中读 ROM 的编程 / 擦除地址的方法。

在 ROM 状态读模式中的读法请参照“39.6.4.1 (4) ROM 状态读模式的转移方法”。

(2) 闪存状态寄存器 0（FSTATR0）的清除方法

使用状态寄存器清除命令将 FSTATR0.ILGLERR 位、FSTATR0.ERSERR 位和 FSTATR0.PRGERR 位置“0”。

当 FSTATR0.ILGLERR 位、FSTATR0.ERSERR 位或者 FSTATR0.PRGERR 位为“1”时，FCU 进入命令锁定状态，不接受状态寄存器清除命令以外的 FCU 命令。当 ILGLERR 位为“1”时，也必须确认 FASTAT.ROMAE 位、FASTAT.DFLAE 位、FASTAT.DFLRPE 位和 FASTAT.DFLWPE 位的值。即使不清除这些位而发行状态寄存器清除命令，ILGLERR 位也不变为“0”。

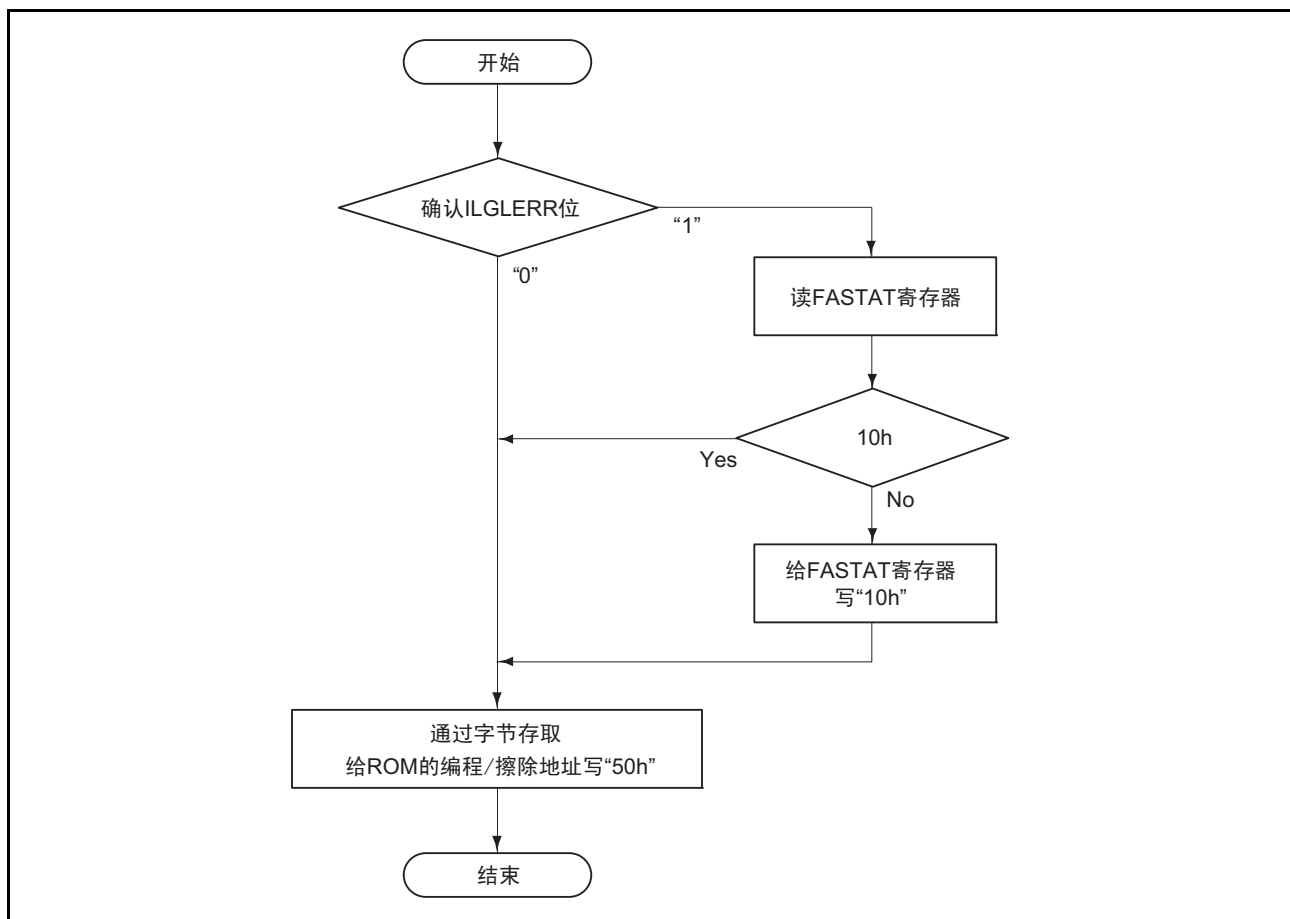


图 39.17 FSTATR0 寄存器的清除方法

(3) FCU 的初始化方法

在发行 FCU 命令后，如果因超时而使 FSTATR0.FR DY 位不变为“1”，就需要通过 FRESETR 寄存器对 FCU 进行初始化。在 FSTATR1.FCUERR 位为“1”时，也需要通过 FRESETR 寄存器对 FCU 进行初始化。无论在何种情况下，FRESETR.FRESETR 位为“1”的状态都必须保持 tRESW2 的时间（参照“41. 电特性”）。在 FRESETR 位保持“1”的期间，禁止读 ROM/E2 数据闪存。因为在 FRESETR 位为“1”的状态下 FENTRYR 寄存器被初始化，所以不能使用 FCU 命令，必须重新进行图 39.10 的处理。

39.6.4.4 挂起 / 恢复

(1) 编程 / 擦除的挂起方法

使用 P/E 挂起命令来中止 ROM 的编程或者擦除。

要发行 P/E 挂起命令时，必须预先通过检查 FSTATR0.ILGLERR 位、FSTATR0.ERSERR 位、FSTATR0.PRGERR 位和 FSTATR1.FCUERR 位为“0”，确认正常进行了编程或者擦除处理。要确认能接受挂起命令时，也必须确认 FSTATR0.SUSRDY 位是“1”。在发行 P/E 挂起命令后，必须读 FSTATR0 寄存器和 FSTATR1 寄存器，确认没有发生错误。

如果在编程或者擦除处理过程中发生异常，ILGLERR 位、PRGERR 位、ERSERR 位和 FCUERR 位中至少有 1 位变为“1”。在确认 SUSRDY 位是“1”后到接受 P/E 挂起命令的期间，如果编程或者擦除处理已经结束，发行的 P/E 挂起命令就被检测为非法命令，因此 ILGLERR 位变为“1”。

如果在接受 P/E 挂起命令的同时编程 / 擦除处理结束，就不产生错误，也不转移到挂起状态（FSTATR0.FR DY 位为“1”并且 FSTATR0.ERSSPD 位和 FSTATR0.PRGSPPD 位为“0”）。如果在接受 P/E 挂起命令并且编程 / 擦除的中止处理正常结束，FCU 就转移到挂起状态，FRDY 位变为“1”并且 ERSSPD 位或者 PRGSPPD 位变为“1”。在发行 P/E 挂起命令后，必须通过检查 ERSSPD 位或者 PRGSPPD 位为“1”，确认转移到挂起状态，然后决定后续的流程。即使未转移到挂起状态，只要在后续的流程中发行 P/E 恢复命令，也发生非法命令错误，FCU 转移到命令锁定状态（参照“39.8.2 错误保护”）。

如果转移到擦除挂起状态，就能对擦除对象以外的块进行编程。在编程或者擦除挂起状态下，能通过清除 FENTRYR 寄存器转移到 ROM 读模式。

有关接受 P/E 挂起命令时的 FCU 运行内容，请参照“39.7 挂起”。

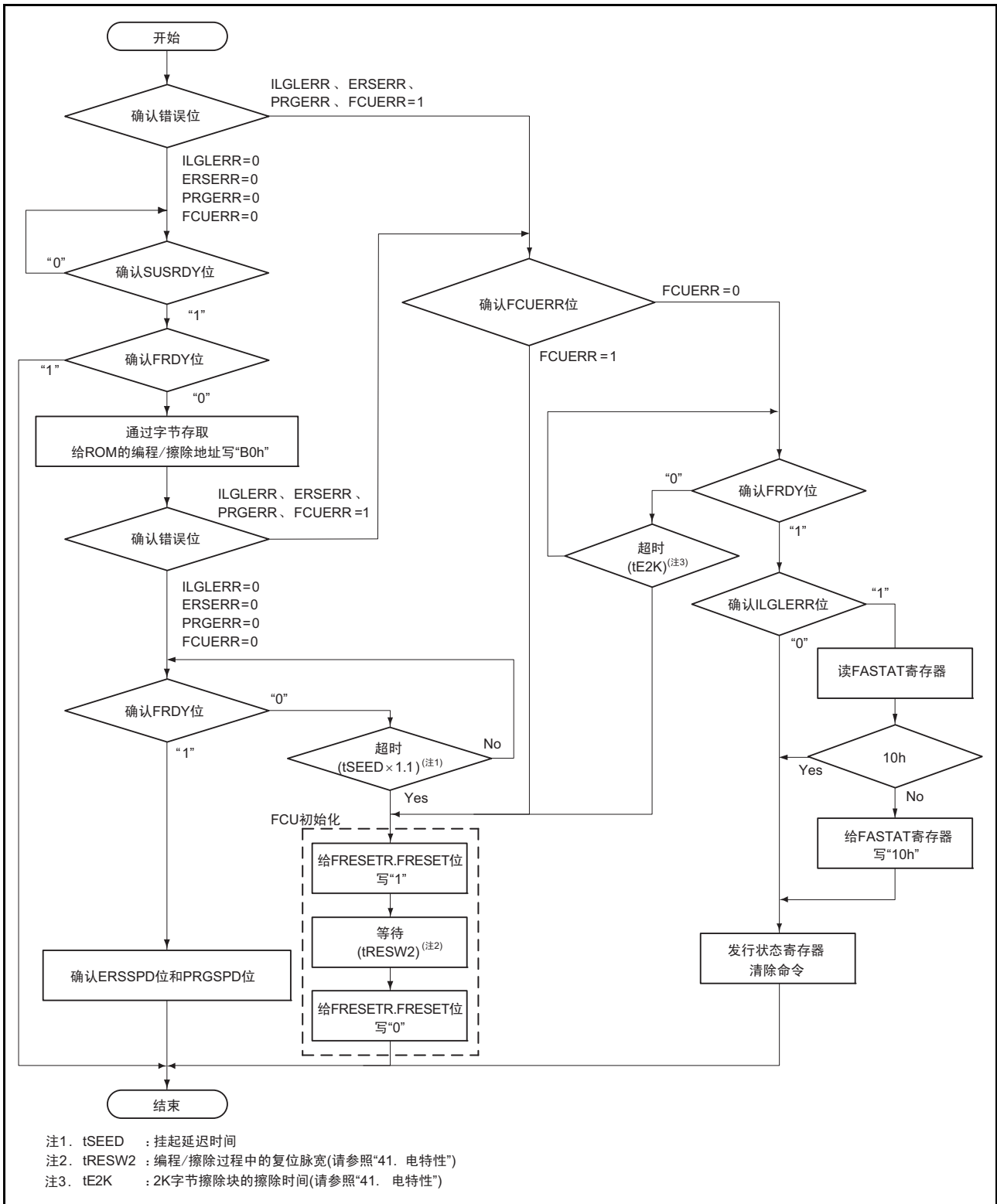


图 39.18 编程 / 擦除的挂起方法

(2) 编程 / 擦除的恢复方法

使用 P/E 恢复命令重新开始挂起的编程或者擦除处理。如果在挂起过程中更改了 FENTRYR 寄存器的设定，就必须在发行 P/E 恢复命令前将发行 P/E 挂起命令前的值重新设定到 FENTRYR 寄存器。

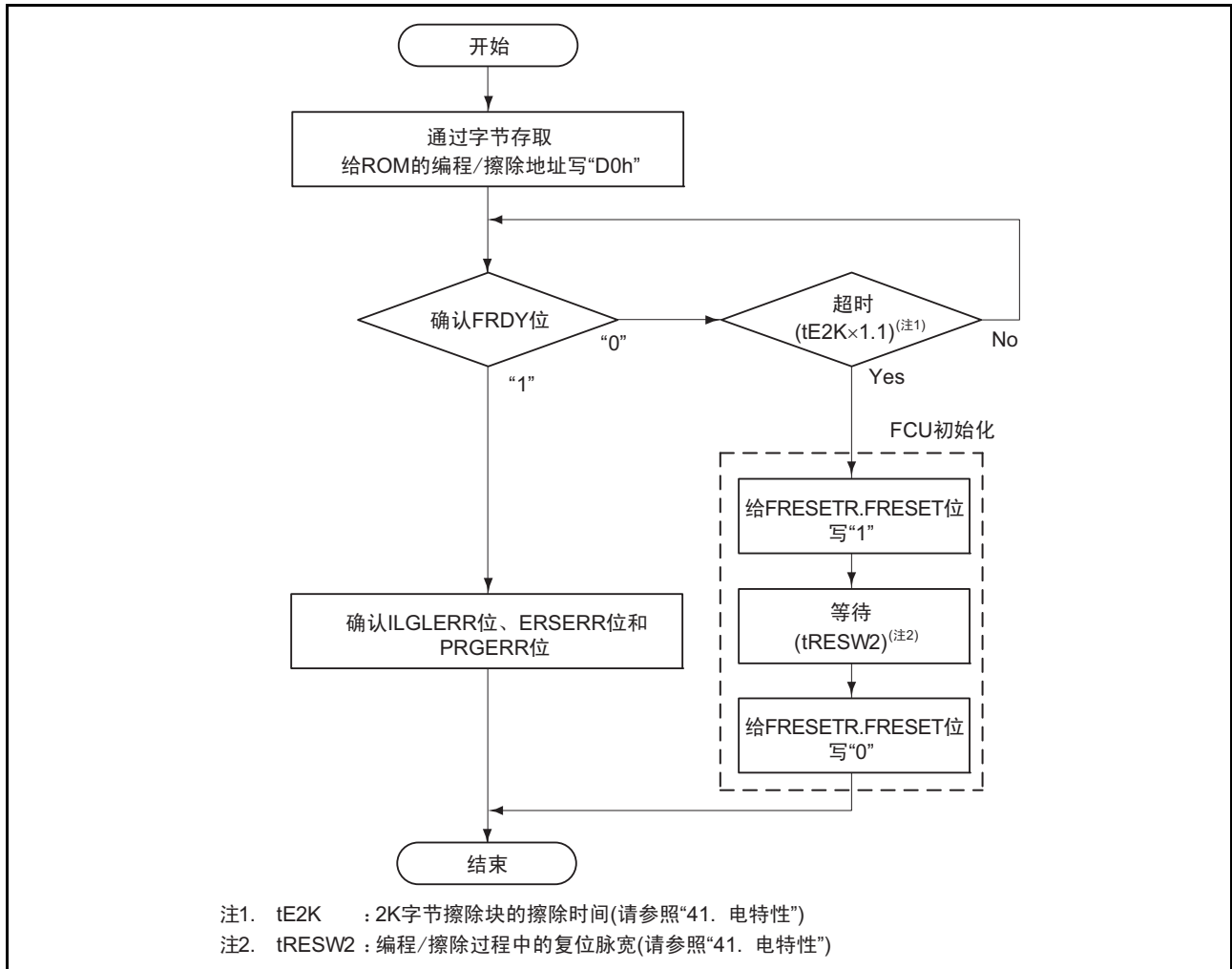


图 39.19 编程 / 擦除的恢复方法

39.7 挂起

不能在编程或者擦除处理过程中读 ROM。通过发行 P/E 挂起命令来中止 ROM 的编程或者擦除处理，变为能读 ROM 的状态。P/E 挂起命令有 2 种编程模式（挂起优先模式、编程 / 擦除优先模式）和 2 种擦除模式（挂起优先模式、擦除优先模式），还有用于重新开始被中止的编程或者擦除处理的 P/E 恢复命令。

(1) 擦除时的挂起

- 挂起优先模式：能以 1 个脉冲（注1）1 次的频率中止擦除，执行挂起。
- 编程 / 擦除优先模式：在 1 个脉冲（注1）结束后，执行挂起（参照图 39.20）。

(2) 编程时的挂起

- 挂起优先模式：能以 1 个脉冲（注1）1 次的频率中止编程，执行挂起。
- 编程 / 擦除优先模式：在 1 个脉冲（注1）结束后，执行挂起。

注 1. 对 1 次编程 / 擦除命令产生多个脉冲，并且执行编程 / 擦除。

39.7.1 编程和擦除时的挂起（挂起优先模式）

擦除挂起模式为挂起优先模式（FCPSR.ESUSPMD 位为“0”）时的擦除处理的中止如图 39.20 所示。

如果 FCU 接受擦除命令，就将 FSTATR0.FRDY 位清“0”，开始进行擦除处理。如果在开始擦除处理后 FCU 转移到能接受 P/E 挂起命令的状态，FSTATR0.SUSRDY 位就变为“1”。如果发行 P/E 挂起命令，FCU 就在接受挂起命令后将 SUSRDY 位置“0”。如果在擦除处理过程中接受挂起命令，即使在外加擦除脉冲过程中，FCU 也开始中止处理，然后将 FSTATR0.ERSSPD 位置“1”。一旦中止处理结束，FCU 就将 FRDY 位置“1”，转移到擦除挂起状态。如果 FCU 在擦除挂起状态下接受 P/E 恢复命令，就将 FRDY 位和 ERSSPD 位置“0”，重新开始擦除处理。中止或者重新开始擦除处理时的 FRDY 位、SUSRDY 位和 ERSSPD 位的操作相同，而不取决于擦除挂起模式。

擦除挂起模式的设定影响擦除脉冲的控制方式。在挂起优先模式中，如果 FCU 在外加过去没有被中止的擦除脉冲 A 过程中接受 P/E 挂起命令，就在中止外加擦除脉冲 A 后转移到擦除挂起状态。通过 P/E 恢复命令重新开始擦除并且重新外加擦除脉冲 A，如果在此期间 FCU 接受 P/E 挂起命令，FCU 就继续外加擦除脉冲 A。如果超过规定的脉冲外加时间，FCU 就在结束擦除脉冲的外加后转移到擦除挂起状态。接着，如果在 FCU 接受 P/E 恢复命令并且开始外加新的擦除脉冲 B 后，FCU 再次接受 P/E 挂起命令，就中止擦除脉冲 B 的外加。在挂起优先模式中，因为以 1 个脉冲中止 1 次擦除脉冲的外加并且优先挂起处理，所以能缩短挂起的延迟时间。

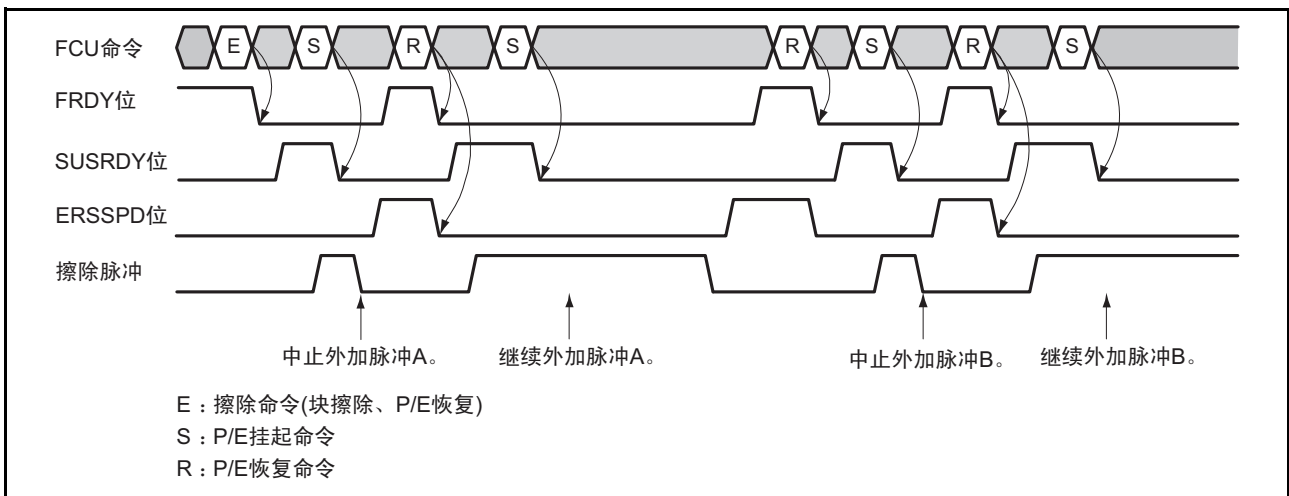


图 39.20 擦除处理的中止（挂起优先模式）

39.7.2 编程和擦除时的挂起 (编程和擦除优先模式)

擦除优先模式 (FCPSR.ESUSPMD 位为“1”)时的擦除处理的中止如图 39.21 所示。擦除优先模式的擦除脉冲控制方式和编程中止处理的编程脉冲控制方式相同。

如果 FCU 在外加擦除脉冲过程中接受 P/E 挂起命令, 就继续外加擦除脉冲。在此模式中, 因为在发行 P/E 恢复命令时不会再次外加擦除脉冲, 所以与挂起优先模式相比, 能缩短整个擦除处理所需的时间。

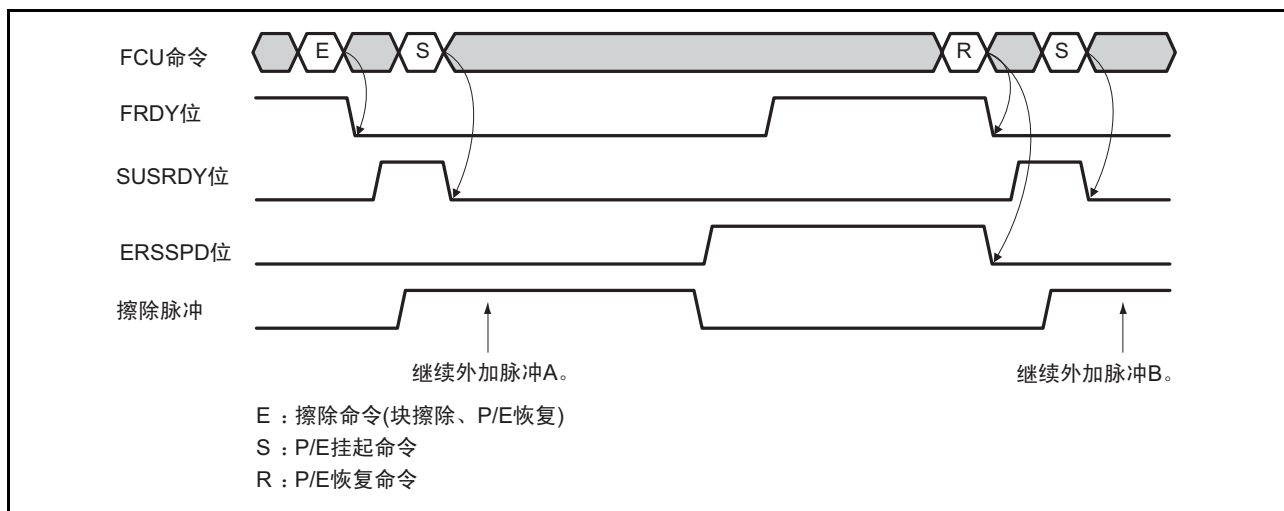


图 39.21 擦除处理的中止 (擦除优先模式)

39.8 保护

ROM 的编程 / 擦除的保护有软件保护和错误保护。

39.8.1 软件保护

软件保护是通过设定控制寄存器和用户区的锁定位来禁止对 ROM 进行编程 / 擦除的状态。如果违反软件保护而发行 ROM 的编程 / 擦除命令，FCU 就检测到错误，进入命令锁定状态。

(1) 通过 FWEPROR 寄存器进行的保护

如果不将 FWEPROR.FLWE[1:0] 位置“01b”，在任何模式中都不能进行改写。

(2) 通过 FENTRYR 寄存器进行的保护

当 FENTRYR.FENTRY0 位都为“0”时，FCU 为 ROM 读模式。因为在 ROM 读模式中不接受 FCU 命令，所以 ROM 为禁止编程和擦除的状态。如果在 ROM 读模式中发行 FCU 命令，FCU 就检测到非法命令错误，进入命令锁定状态（参照“39.8.2 错误保护”）。

(3) 通过锁定位进行的保护

用户区的各擦除块内置锁定位。当 FPROTR.FPROTCN 位为“0”时，锁定位为“0”的擦除块为禁止编程和擦除的状态。要对锁定位为“0”的擦除块进行编程或者擦除时，必须将 FPROTCN 位置“1”。如果违反通过锁定位进行的保护而发行 ROM 的编程 / 擦除命令，FCU 就检测到编程 / 擦除错误，进入命令锁定状态（参照“39.8.2 错误保护”）。

39.8.2 错误保护

错误保护是因误发行 FCU 命令以及发生禁止的存取而导致 FCU 在检测到误动作后禁止接受 FCU 命令的状态（命令锁定状态）。

一旦 FCU 进入命令锁定状态（FASTAT.CMDLK 位为“1”），就将状态位（FSTATR0.ILGLERR 位、FSTATR0.ERSERR 位、FSTATR0.PRGERR 位、FSTATR1.FCUERR 位、FASTAT.ROMAE 位）中的某位或者多个位置“1”，禁止对 ROM 进行编程和擦除。要解除命令锁定状态时，需要在 FASTAT 寄存器为“10h”的状态下发行状态寄存器清除命令。

当 FAEINT.CMDLKIE 位为“1”时，如果 FCU 进入命令锁定状态（FASTAT.CMDLK 位为“1”），就产生闪存接口错误（FIFERR）中断；当 FAEINT.ROMAEIE 位为“1”时，即使 FASTAT.ROMAE 位为“1”，也产生 FIFERR 中断。

ROM 相关的错误保护内容和错误检测时的状态位的值（FSTATR0.ILGLERR 位、FSTATR0.ERSERR 位、FSTATR0.PRGERR 位、FSTATR1.FCUERR 位、FASTAT.ROMAE 位）的关系如表 39.9 所示。如果在编程或者擦除处理过程中发行挂起以外的命令，就转移到命令锁定状态，但是 FCU 继续进行编程或者擦除处理。在此状态下无法通过发行 P/E 挂起命令来中止编程或者擦除。如果在命令锁定状态下发行命令，ILGLERR 位就变为“1”。

表 39.9 错误保护一览表（ROM 专用 +ROM/E2 数据闪存共用）

分类	内容	ILGLERR	ERSERR	PRGERR	FCUERR	ROMAE
FENTRYR 的设置错误	将 FENTRYR.FENTRYD 和 FENTRYR.FENTRY0 位置“1”。	1	0	0	0	0
	挂起和恢复时的 FENTRYR 寄存器的设定不同。	1	0	0	0	0
非法命令错误	在 FCU 命令的第 1 周期指定未定义代码。	1	0	0	0	0
	在多个周期的 FCU 命令的最后周期指定“D0h”以外的数据。	1	0	0	0	0
	给 PCKAR 寄存器设定 1 ~ 100MHz 以外的外围时钟。 (在设定 1 ~ 4MHz 和 50 ~ 100MHz 时, 不检测错误)	1	0	0	0	0
	在编程或者擦除处理过程中发行挂起以外的命令。	1	0	0	0	0
	在编程和擦除以外的处理过程中发行挂起命令。	1	0	0	0	0
	在挂起状态下发行挂起命令。	1	0	0	0	0
	在挂起以外的状态下发行恢复命令。	1	0	0	0	0
	在编程挂起状态下发行编程 / 擦除 (编程、锁定位编程、块擦除) 命令。	1	0	0	0	0
	在擦除挂起状态下发行块擦除命令。	1	0	0	0	0
	在擦除挂起状态下发行擦除挂起对象区的编程命令/锁定位编程命令。	1	0	0	0	0
	在编程命令的第 2 个周期指定“80h”以外的数据。	1	0	0	0	0
在命令锁定状态下发行命令。	1	0/1	0/1	0/1	0/1	
擦除错误	在擦除处理过程中发生错误。	0	1	0	0	0
	在 FPROTR.FPROTCN 位为“0”时, 对锁定位为“0”的擦除块发行块擦除命令。	0	1	0	0	0
编程错误	在编程处理过程中发生错误。	0	0	1	0	0
	在 FPROTR.FPROTCN 位为“0”时, 对锁定位为“0”的擦除块发行编程命令或者锁定位编程命令。	0	0	1	0	0
FCU 错误	在 FCU 内部处理过程中发生错误。	0	0	0	1	0
ROM 存取违反	在 FENTRYR.FENTRY0 位为“1”并且为 ROM P/E 正常模式时, 当用户区的容量为 128 字节时, 对 00FE 0000h ~ 00FF FFFFh 发行读存取命令; 当用户区的容量为 256K 字节时, 对 00FC 0000h ~ 00FF FFFFh 发行读存取命令; 当用户区的容量为 384K 字节时, 对 00FA 0000h ~ 00FF FFFFh 发行读存取命令; 当用户区的容量为 512K 字节时, 对 00F8 0000h ~ 00FF FFFFh 发行读存取命令。	1	0	0	0	1
	在 FENTRYR.FENTRY0 位为“0”的情况下, 当用户区的容量为 128 字节时, 对 00FE 0000h ~ 00FF FFFFh 发行读存取命令; 当用户区的容量为 256K 字节时, 对 00FC 0000h ~ 00FF FFFFh 发行读存取命令; 当用户区的容量为 384K 字节时, 对 00FA 0000h ~ 00FF FFFFh 发行读存取命令; 当用户区的容量为 512K 字节时, 对 00F8 0000h ~ 00FF FFFFh 发行存取命令。	1	0	0	0	1
	在设定 FENTRYR 寄存器后转移到 ROM P/E 模式的状态下, 当用户区的容量为 128 字节时, 对 FFFE 0000h ~ FFFF FFFFh 发行读存取命令; 当用户区的容量为 256K 字节时, 对 FFFC 0000h ~ FFFF FFFFh 发行读存取命令; 当用户区的容量为 384K 字节时, 对 FFFA 0000h ~ FFFF FFFFh 发行读存取命令; 当用户区的容量为 512K 字节时, 对 FFF8 0000h ~ FFFF FFFFh 发行存取命令。	1	0	0	0	1

39.9 用户引导模式

如果通过 MD 引脚设定为用户引导模式并且解除复位，就转移到用户引导模式。此时的复位向量为用户引导区的地址 FF7F FFFCh。其他向量表请参照一般的向量表（参照“14. 中断控制器（ICUb）”）。

在用户引导模式中，能建立使用任意接口的编程程序，并且能通过发行 FCU 命令进行用户区或者数据区的编程和擦除。另外，必须在引导模式中对用户引导区进行编程。

经由引导模式转移到用户引导模式。通过引导程序读取写到用户引导区的信息，判断是否转移到用户引导模式。

39.10 引导模式

39.10.1 系统结构

在引导模式中，能从主机发送控制命令和编程数据，对用户区或者数据区进行编程和擦除。为了进行主机和 RX210 之间的通信，将内部 SCI 用于异步模式。需要准备用于将控制命令发送到主机的工具和编程数据。

如果通过引导模式启动 RX210，就执行引导区中的程序，此引导区中的程序在自动调整 SCI 的位速率以及接受主机的控制命令后，控制编程和擦除。

引导模式的系统结构如图 39.22 所示。

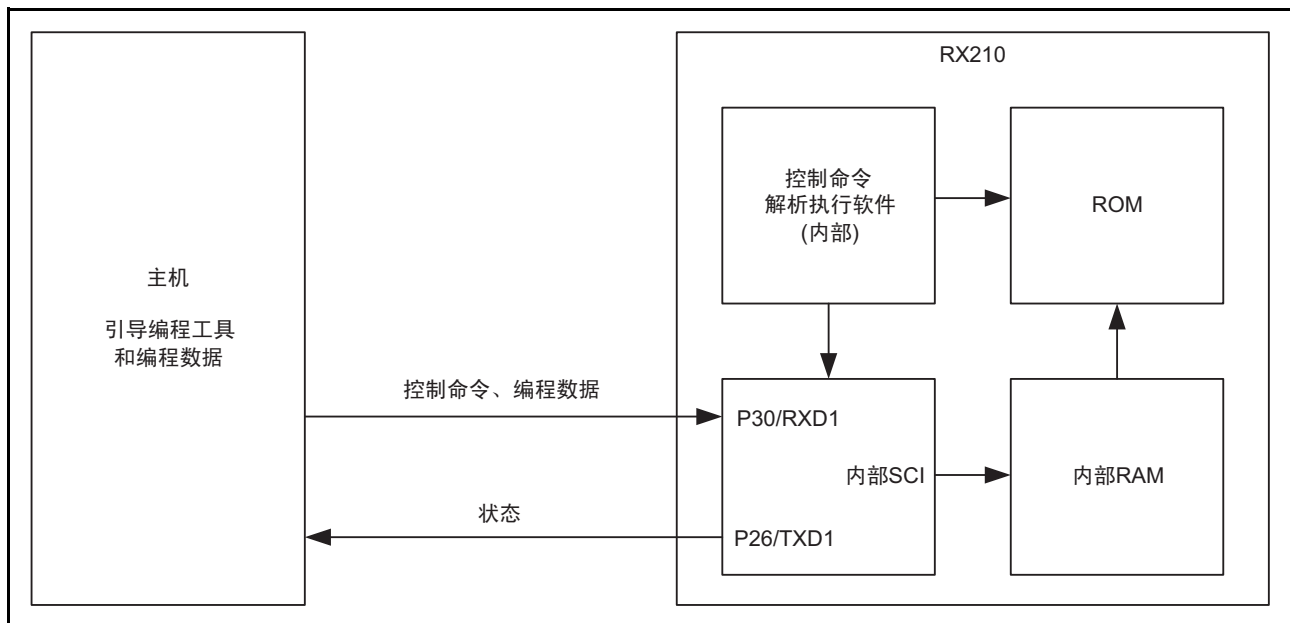


图 39.22 引导模式的系统结构

39.10.2 引导模式的状态转移

引导模式的状态转移图如图 39.23 所示。

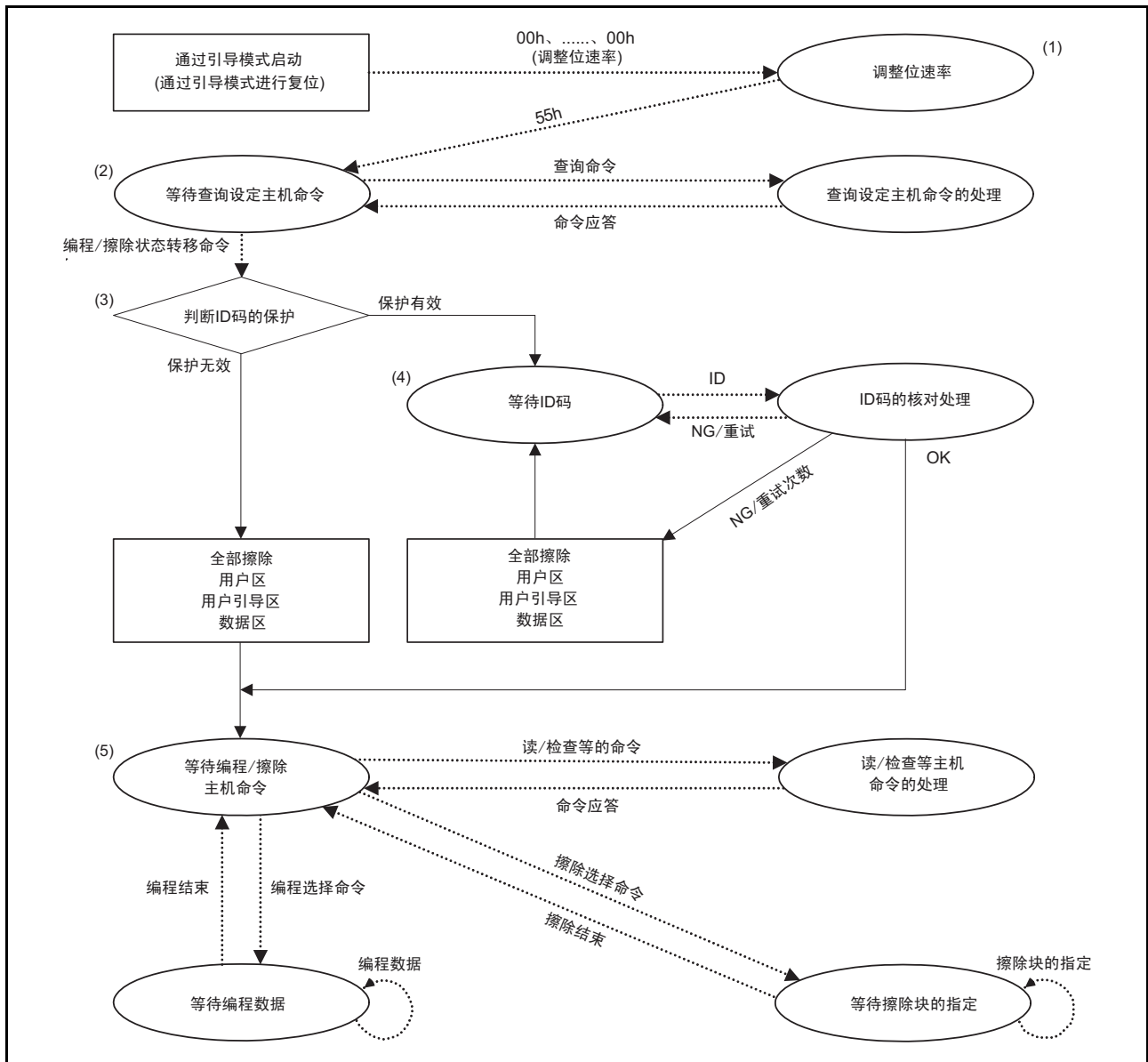


图 39.23 引导模式的状态转移图

(1) 位速率的调整

如果通过引导模式启动 RX210，就自动调整主机和 SCI 的位速率。如果位速率的自动调整结束，就将“00h”从 RX210 发送到主机。此后，如果 RX210 正常接收从主机送来的“55h”，就转移到查询设定主机命令等待状态。有关位速率调整的详细内容，请参照“39.10.3 位速率的自动调整”。

(2) 查询设定主机命令的等待

这是查询区域大小、区域结构、区域起始地址和支持状况等以及选择器件、时钟模式和位速率的状态。如果从主机发行编程 / 擦除状态转移命令，就转移到判断 ID 码保护的有效或者无效的状态。有关查询设定主机命令的详细内容，请参照“39.10.6 查询设定主机命令的等待状态”。

(3) ID 码保护的判断

判断 ID 码保护的有效或者无效。通过写在 ROM 中的控制码和 ID 码，判断 ID 码保护的有效或者无效。当 ID 码保护有效时，转移到 ID 码等待状态；当 ID 码保护无效时，全部擦除用户区和数据区，并且转移到编程 / 擦除主机命令等待状态。有关控制码和 ID 码的详细内容，请参照“39.10.4 ID 码保护（引导模式）”。

(4) ID 码的等待

等待从主机送来的控制码和 ID 码。将主机送来的控制码、ID 码和 ROM 中的代码进行比较，如果相同，就转移到编程 / 擦除主机命令等待状态；如果不同，就返回到 ID 码等待状态。如果连续 3 次不同并且保护状态为验证方法 1，就进行全部擦除并且再次返回到 ID 码等待状态。要解除此不同的状态时，需要进行复位。有关控制码和 ID 码的详细内容，请参照“39.10.4 ID 码保护（引导模式）”。

(5) 编程 / 擦除主机命令的等待

这是根据主机的命令进行编程或者擦除的状态。根据 RX210 接收的命令，转移到编程数据等待状态、擦除块指定等待状态、读 / 检查等命令处理执行状态。

如果 RX210 接收编程选择命令，就转移到编程数据等待状态。主机在发送编程选择命令后，必须接着发送编程起始地址和编程数据。如果将编程起始地址设定为“FFFF FFFFh”，就在编程结束后，从编程数据等待状态转移到编程 / 擦除命令等待状态。

如果 RX210 接收擦除选择命令，就转移到擦除块指定等待状态。主机在发送擦除选择命令后，必须接着发送擦除块号。如果将擦除块号设定为“FFh”，就在擦除结束后，从擦除块指定等待状态转移到编程 / 擦除命令等待状态。因为在通过引导模式启动后到转移到编程 / 擦除主机命令状态的期间，用户区、用户引导区和数据区被全部擦除，所以在引导模式中除了不进行复位而要擦除新编程的数据的情况以外，不需要进行擦除。

除了编程 / 擦除命令以外，还有用户区和用户引导区的校验和命令、空白检查（擦除检查）命令、存储器读命令、状态信息取命令等主机命令。

39.10.3 位速率的自动调整

如果通过引导模式启动 RX210，就测量从主机连续发送的异步 SCI 通信数据“00h”的 Low 电平期间。在测量 Low 电平期间时，主机的 SCI 发送 / 接收格式必须为 8 位数据、1 个停止位、没有奇偶校验，位速率必须设定为 9600bps 或者 19200bps。从测量的 Low 电平期间计算主机的 SCI 的位速率，如果位速率调整结束，RX210 就将“00h”发送到主机。如果主机正常接收到“00h”，就必须将“55h”从主机发送到 RX210。否则，就必须通过引导模式重新启动 RX210，并且重新自动调整位速率。如果 RX210 正常接收“55h”，就发送“E6h”。否则，就发送“FFh”。

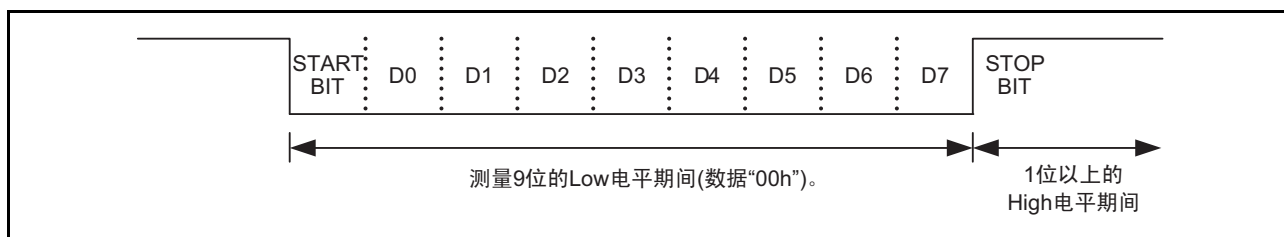


图 39.24 自动调整位速率时的 SCI 发送 / 接收格式

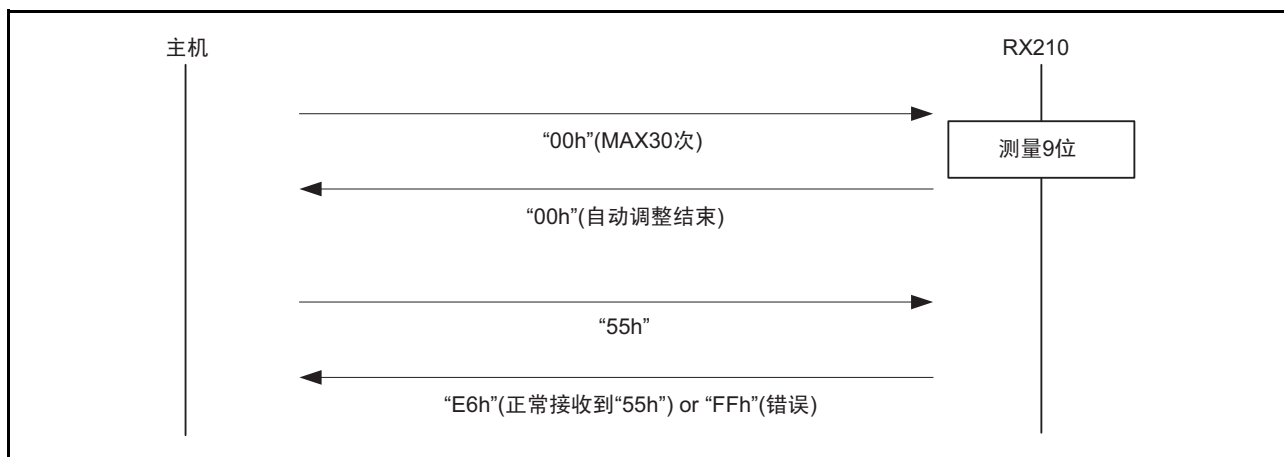


图 39.25 主机和 RX210 之间的通信顺序

位速率的自动调整取决于主机的 SCI 的位速率和 RX210 的外围时钟的频率，因为有可能无法进行正常的调整，所以必须在表 39.10 所示的条件下进行 SCI 的通信。

表 39.10 可自动调整位速率的条件

主机的 SCI 的位速率	HOCO 的频率
9600bps	2MHz
19200bps	2MHz

39.10.4 ID 码保护（引导模式）

这是禁止从 PC 等主机读、编程和擦除的功能。

在通过引导模式启动并且自动调整位速率后，使用写在 ROM 中的控制码和 ID 码，判断 ID 码保护的有效或者无效以及 ID 码保护。在 ID 码保护有效的情况下，只在从主机送来的代码和 ROM 中的控制码及 ID 码相同时，允许读、编程和擦除。

ROM 中的控制码和 ID 码是 32 位长的 4 字数据，控制码和 ID 码的结构如图 39.26 所示。必须以 32 位为单位设定 ID 码。

	31	24	23	16	15	8	7	0
FFFF FFA0h	控制码		ID码1		ID码2		ID码3	
FFFF FFA4h	ID码4		ID码5		ID码6		ID码7	
FFFF FFA8h	ID码8		ID码9		ID码10		ID码11	
FFFF FFACH	ID码12		ID码13		ID码14		ID码15	

图 39.26 ROM 中的控制码和 ID 码的结构

(1) 控制码

控制码决定 ID 码保护的有效或者无效以及主机的验证方法，控制码和验证方法如表 39.11 所示。

表 39.11 ID 码保护的规格

控制码	ID 码	保护状态	连接 SCI 时的运行
45h	任意	保护有效 (验证方法 1)	ID 码相同：结束 ID 码保护并且转移到主机命令等待状态。 ID 码不同：再次转移到 ID 码等待状态。但是，如果连续 3 次 ID 码不同，就进行全部擦除。
52h	50h、72h、6Fh、74h、65h、63h、74h、FFh、.....、FFh 除外	保护有效 (验证方法 2)	ID 码相同：结束 ID 码保护并且转移到主机命令等待状态。 ID 码不同：再次转移到 ID 码等待状态。
	50h、72h、6Fh、74h、65h、63h、74h、FFh、.....、FFh	保护有效 (验证方法 3)	总是判断为 ID 码不同。
上述以外	—	保护无效	擦除全部的块。

(2) ID 码

ID 码能设定为任意的值。如果控制码为“52h”并且 ID 码从 ID 码 1 开始依次设定“50h”、“72h”、“6Fh”、“74h”、“65h”、“63h”、“74h”、“FFh”、.....、“FFh”，就不判断 ID 码是否相同而总是视为不同，禁止从主机读、编程和擦除。

(3) 设定 ID 码的编程例子

控制码为“45h”，ID 码从 ID 码 1 开始按依次设定“01h”、“02h”、“03h”、“04h”、“05h”、“06h”、“07h”、“08h”、“0Ah”、“0Bh”、“0Ch”、“0Dh”、“0Eh”、“0Fh”时的编程例子：

```
.SECTION ID_CODE, CODE
.ORG 0FFFFFFA0h
.LWORD 45010203h
.LWORD 04050607h
.LWORD 08090A0Bh
.LWORD 0C0D0E0Fh
```

39.10.5 UB 码

有关 UB 码，请参照“7.3 UB 码”。

39.10.6 查询设定主机命令的等待状态

在查询设定主机命令等待状态下能使用的主机命令一览表如表 39.12 所示。在编程 / 擦除主机命令等待状态下，也能使用引导程序状态查询命令。其他命令只能在查询设定主机命令等待状态下使用。

表 39.12 查询设定主机命令

主机命令名	功能
支持器件查询	查询器件码和引导程序的产品型号。
器件选择	选择器件码。
时钟模式查询	查询时钟模式数和各自的值。
时钟模式选择	通知已选择的时钟模式。
倍增比查询	查询时钟的种类、倍增比 / 分频比的种类以及倍增比 / 分频比。
工作频率查询	查询时钟的种类、最大工作频率和最小工作频率。
用户引导区信息查询	查询用户引导区的个数以及起始地址和结束地址。
用户区信息查询	查询用户区的个数以及起始地址和结束地址。
擦除块信息查询	查询块数以及起始地址和结束地址。
编程长度查询	查询编程时的数据长度。
新位速率选择	更改主机和 RX210 之间的 SCI 通信的位速率。
编程 / 擦除状态转移	转移到 ID 码保护判断状态。
引导程序状态查询	查询处理状态。

如果主机发送未定义的命令，RX210 就发送命令错误应答。命令错误应答的内容如下所示。在命令中保存主机发送的命令的起始字节。

错误应答	80h	命令
------	-----	----

在查询设定主机命令等待状态下，必须参考查询命令的应答，按照器件的选择 → 时钟模式的选择 → 新位速率的选择的顺序从主机发送选择命令，设定 RX210。在发行时钟模式选择命令前，不能使用支持器件的查询和时钟模式的查询以外的查询命令。如果搞错命令的发送顺序，RX210 就发送命令错误应答。查询设定主机命令等待状态下的主机命令使用例子如图 39.27 所示。

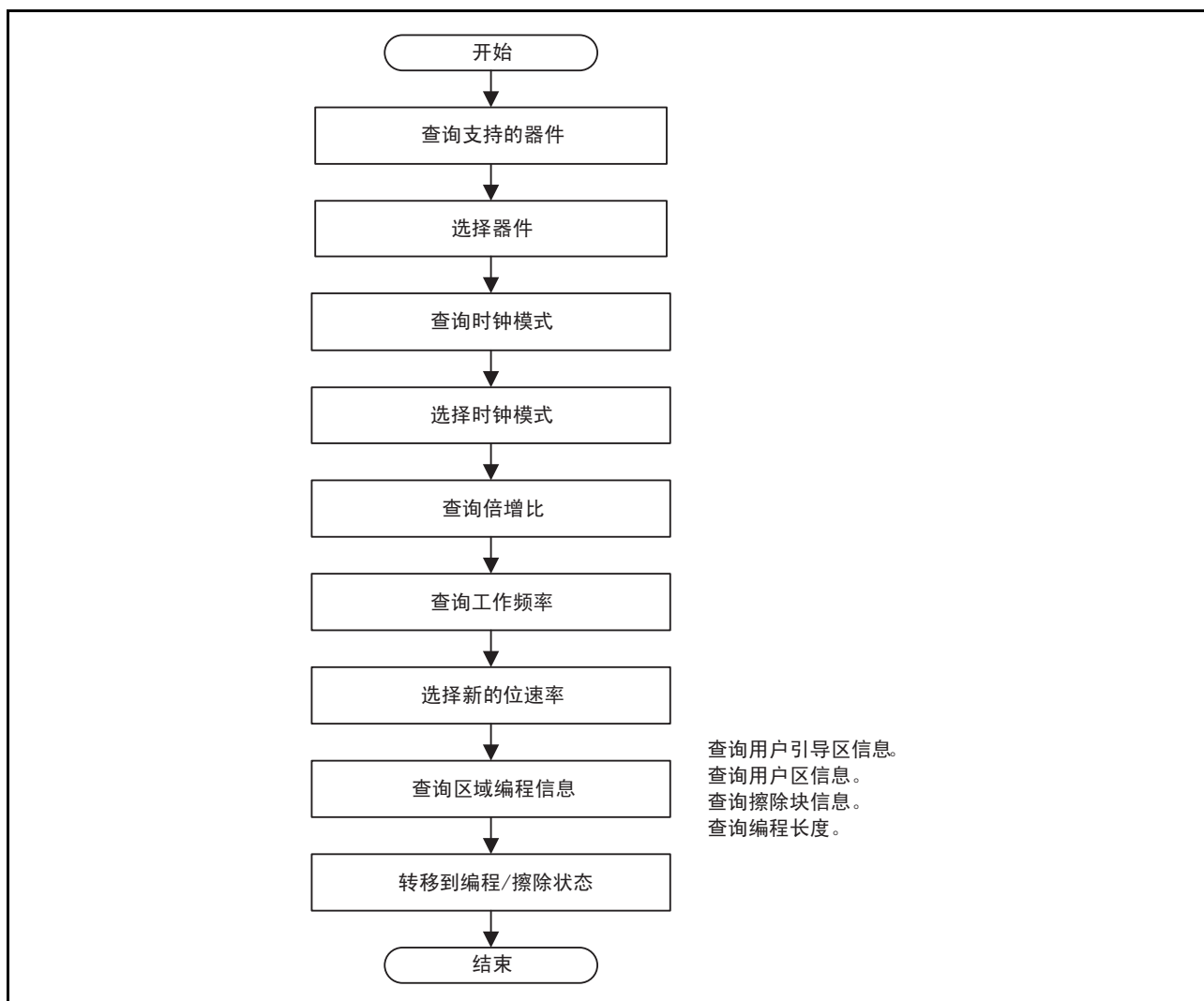


图 39.27 用户区 / 用户引导区的查询设定主机命令的使用例子

以下说明各主机命令的详细内容。说明内容中的“命令”是从主机发送到 RX210 的命令，“应答”是从 RX210 发送到主机的应答。“校验和”是指在合计 RX210 发送的各字节时，为了得到“00h”而算出的字节数据。

(1) 支持器件查询

如果主机发送支持器件查询命令，RX210 就通过引导程序发送能支持的器件信息。如果在主机选择器件后发送支持器件查询命令，RX210 就只发送所选的器件信息。作为支持器件查询命令的应答，RX210 按各自的顺序发送小端法指定和大端法指定的 2 个器件信息。

命令	20h		
应答	30h	长度	器件数
	字符数	器件码（小端法指定）	
	字符数	器件码（大端法指定）	
	SUM	产品名称	

- 长度（1 字节）：器件数、字符数、器件码和名称的数据总字节数
 器件数（1 字节）：引导程序支持的品种数
 字符数（1 字节）：器件码和名称的字符数
 器件码（4 字节）：芯片名称的 ASCII 码
 产品名称（n 字节）：支持器件名称的 ASCII 码
 SUM（1 字节）：校验和

(2) 器件选择

如果主机发送器件选择命令，RX210 就检查所指定的器件是否为能支持的器件。如果是能支持的器件，RX210 就将支持器件更改为指定的器件并且发送应答（06h）。如果不是能支持的器件或者送来的命令为非法命令，RX210 就发送错误应答（90h）。

必须根据要编程的数据，从支持器件查询命令发送的 2 个器件信息中选择某个字节序指定的器件码。

命令	10h	长度	器件码	SUM
应答	06h			
错误应答	90h	错误		

- 长度（1 字节）：器件码的字符数（是固定值，为“4”。）
 器件码（4 字节）：芯片名称的 ASCII 码（和支持器件查询命令的应答相同的代码）
 SUM（1 字节）：校验和
 错误（1 字节）：错误码
 11h: 校验和错误（命令为非法命令）
 21h: 器件码不同

(3) 时钟模式查询

如果主机发送时钟模式查询命令，RX210 就发送可选择的时钟模式。如果在主机选择时钟模式后发送时钟模式查询命令，RX210 就只发送所选时钟模式的信息。

命令	21h			
应答	31h	长度		
	模式	模式	...	模式
	SUM			

长度（1 字节）：模式数、模式的数据总字节数

模式（1 字节）：可选择的时钟模式（例：01h 时钟模式 1）

SUM（1 字节）：校验和

(4) 时钟模式选择

如果主机发送时钟模式选择命令，RX210 就检查所指定的时钟模式是否为能支持的模式。如果是能支持的模式，RX210 就将时钟模式更改为指定的模式并且发送应答（06h）。如果不是能支持的模式或者送来的命令为非法命令，RX210 就发送错误应答（91h）。

必须在发送器件选择命令后发送时钟模式选择命令。即使在时钟模式查询结果的时钟模式数为“00h”或者“01h”时，也必须通过时钟模式选择命令设定查询结果的模式的值。

命令	11h	长度	模式	SUM
应答	06h			
错误应答	91h	错误		

长度（1 字节）：模式的字符数（是固定值，为“1”。）

模式（1 字节）：时钟模式（和时钟模式查询命令的应答相同的模式）

SUM（1 字节）：校验和

错误（1 字节）：错误码

11h: 校验和错误（命令为非法命令）

22h: 时钟模式不同

(5) 倍增比查询

如果主机发送倍增比查询命令，RX210 就发送时钟的种类、倍增比 / 分频比的种类以及倍增比 / 分频比的信息。

命令

22h

32h	长度	时钟数		
倍增比的种类	倍增比	倍增比	...	倍增比
倍增比的种类	倍增比	倍增比	...	倍增比
...
倍增比的种类	倍增比	倍增比	...	倍增比
SUM				

- 长度（1 字节）：时钟数、倍增比的种类、倍增比的数据总字节数
 时钟数（1 字节）：时钟的种类（例：02h 系统时钟和外围时钟）
 倍增比的种类（1 字节）：可选择的倍增比 / 分频比的种类
 （例：04h 系统时钟有 1 倍增、2 倍增、4 倍增、8 倍增共 4 种）
 倍增比（1 字节）：倍增比（例：04h=4 4 倍增）← 用正数指定
 分频比（例：FEh=-2 2 分频）← 用负数指定
 SUM（1 字节）：校验和

(6) 工作频率查询

如果主机发送工作频率查询命令，RX210 就发送各时钟的工作频率的最小值和最大值的消息。

命令

23h

33h	长度	时钟数
最小频率	最大频率	
最小频率	最大频率	
...	...	
最小频率	最大频率	
SUM		

- 长度（1 字节）：时钟数、最小频率、最大频率的数据总字节数
 时钟数（1 字节）：时钟的种类（例：02h 系统时钟和外围时钟）
 最小频率（2 字节）：工作频率的最小值（例：07D0h 20.00MHz）
 这是将频率（MHz）的小数点后的 2 位的值扩大 100 倍的值。
 最大频率（2 字节）：工作频率的最大值
 格式和最小频率相同。
 SUM（1 字节）：校验和

(7) 用户引导区信息查询

如果主机查询用户引导区信息，RX210 就发送用户引导区的区域数和地址的信息。

命令	24h		
应答	34h	长度	区域数
	区域的起始地址		
	区域的结束地址		
	区域的起始地址		
	区域的结束地址		
	...		
	区域的起始地址		
	区域的结束地址		
	SUM		

长度（1 字节）：区域数、区域的起始地址、区域的结束地址的数据总字节数

区域数（1 字节）：用户区的区域数（连续区域为 1 个区域）

区域的起始地址（4 字节）：用户区的起始地址

区域的结束地址（4 字节）：用户区的结束地址

SUM（1 字节）：校验和

(8) 用户区信息查询

如果主机查询用户区信息，RX210 就发送用户区的区域数和地址的信息。

命令	25h		
应答	35h	长度	区域数
	区域的起始地址		
	区域的结束地址		
	区域的起始地址		
	区域的结束地址		
	...		
	区域的起始地址		
	区域的结束地址		
	SUM		

长度（1 字节）：区域数、区域的起始地址、区域的结束地址的数据总字节数

区域数（1 字节）：用户区的区域数（连续区域为 1 个区域）

区域的起始地址（4 字节）：用户区的起始地址

区域的结束地址（4 字节）：用户区的结束地址

SUM（1 字节）：校验和

(9) 擦除块信息查询

如果主机查询擦除块信息，RX210 就发送用户区和数据区的合计擦除块数和地址的信息。

命令	26h		
应答	36h	长度	块数
	块的起始地址		
	块的结束地址		
	块的起始地址		
	块的结束地址		
	...		
	块的起始地址		
	块的结束地址		
	SUM		

长度（2 字节）：块数、块的起始地址、块的结束地址的数据总字节数

块数（1 字节）：用户区的擦除块数

区域的起始地址（4 字节）：擦除块的起始地址

区域的结束地址（4 字节）：擦除块的结束地址

SUM（1 字节）：校验和

(10) 编程长度查询

如果主机查询编程长度，RX210 就发送编程长度的信息。

命令	27h			
应答	37h	长度	编程长度	SUM

长度（1 字节）：编程长度的字符数（是固定值，为“2”。）

编程长度（2 字节）：编程单位（以字节数为单位）

SUM（1 字节）：校验和

(11) 新位速率选择

如果主机发送新位速率选择命令，RX210 就检查能否将内部 SCI 设定为指定的新位速率。如果能设定为新位速率，RX210 就发送应答（06h）并且将 SCI 设定为新位速率。如果不能设定为新位速率或者送来的命令为非法命令，RX210 就发送错误应答（BFh）。如果主机接收应答（06h），就用发送新位速率选择命令时的位速率进行 1 位时间的等待，并且将主机的位速率更改为新位速率。然后，主机通过新位速率发送确认数据（06h），RX210 发送确认数据的应答（06h）。

必须在发送时钟模式选择命令后发送新位速率选择命令。



图 39.28 新位速率的选择顺序

命令	3Fh	长度	位速率	输入频率
	时钟数	倍增比 1	倍增比 2	
	SUM			
应答	06h			
错误应答	BFh	错误		
确认	06h			
应答	06h			

- 长度（1 字节）：位速率、输入频率、时钟数、倍增比的数据总字节数
- 位速率（2 字节）：新位速率（例：00C0h 19200bps）
将位速率值设定为 1/100 的值。
- 输入频率（2 字节）：RX210 的输入频率（例：04E2h 12.50MHz）
这是将输入频率的小数点后的 2 位的值扩大 100 倍的值。
- 时钟数（1 字节）：时钟的种类（例：02h 系统时钟和外围时钟）
- 倍增比 1（1 字节）：输入频率对应的系统时钟（ICLK）的倍增比 / 分频比
倍增比（例：04h=4 4 倍增）← 用正数指定
分频比（例：FEh=-2 2 分频）← 用负数指定
- 倍增比 2（1 字节）：输入频率对应的外围时钟（PCLK）的倍增比 / 分频比
和倍增比 1 相同的格式
- SUM（1 字节）：校验和
- 错误：错误码
- 11h: 校验和错误
 - 24h: 位速率不能选择错误
 - 25h: 输入频率错误
 - 26h: 倍增比错误
 - 27h: 工作频率错误

- 位速率不能选择错误
 如果因RX210的SCI的误差不足4%而无法通过新位速率选择命令设定所指定的位速率，就产生位速率不能选择错误。假设通过新位速率选择命令指定的位速率为B、输入频率为 f_{EX} 、倍增比2为 $M_{P\phi}$ 、SCI的位速率寄存器（BRR）的设定值为N以及串行模式寄存器（SMR）的CKS[1:0]位的设定值为n，则用以下表达式求位速率的误差。

$$\text{误差 (\%)} = \frac{f_{EX} \times M_{P\phi} \times 10^6}{(N + 1) \times B \times 64 \times 2^{2n-1}} - 1$$
- 输入频率错误
 如果通过新位速率选择命令指定的输入频率不在与通过时钟模式选择命令指定的时钟模式对应的输入频率的最小值和最大值的范围内，就产生输入频率错误。
- 倍增比错误
 如果通过新位速率选择命令指定的倍增比不是与通过时钟模式选择命令指定的时钟模式对应的倍增比，就产生倍增比错误。必须使用倍增比查询命令确认能选择的倍增比。
- 工作频率错误
 如果在通过新位速率选择命令指定的工作频率下RX210无法运行，就产生工作频率错误。RX210通过新位速率选择命令指定的输入频率和倍增比进行工作频率的计算，并且检查计算结果是否在各时钟工作频率的最小值和最大值的范围内。必须使用工作频率查询命令确认各时钟工作频率的最小值和最大值。

(12) 编程 / 擦除状态转移

如果主机发送编程 / 擦除状态转移命令，RX210 就通过写在 ROM 中的控制码和 ID 码判断 ID 码保护的有效或者无效。当 ID 码保护有效时，发送应答（16h）并且转移到 ID 码等待状态；当 ID 码保护无效时，全部擦除用户区、用户引导区和数据区。在全部擦除后，RX210 发送应答（26h）并且转移到编程 / 擦除主机命令等待状态。如果发生错误而没有全部擦除，RX210 就发送错误应答（C0h→51h）。

不能在选择器件、时钟模式和新位速率前发行编程 / 擦除状态转移命令。

命令	40h	
应答	ACK	
错误应答	C0h	51h

ACK（1 字节） : ACK 码
 26h: ID 码保护无效
 16h: ID 码保护有效

(13) 引导程序状态查询

如果主机发送引导程序状态查询命令，RX210 就发送当前的状态。能在查询设定主机命令等待状态和编程 / 擦除主机命令等待状态下使用引导程序状态查询命令。

命令	4Fh			
应答	5Fh	长度	状态	错误

长度（1 字节）：状态、错误的总数据字节数（是固定值，为“2”。）

状态（1 字节）：RX210 的状态（参照表 39.13）

错误（1 字节）：RX210 的错误发生状况（参照表 39.14）

表 39.13 状态的内容

代码	内容
11h	等待器件的选择。
12h	等待时钟模式的选择。
13h	等待位速率的选择。
1Fh	等待向编程 / 擦除主机命令等待状态的转移（位速率选择结束）。
31h	正在擦除用户区和用户引导区。
3Fh	等待编程 / 擦除主机命令。
4Fh	等待编程数据的接收。
5Fh	等待擦除块的指定。

表 39.14 错误的内容

代码	内容
00h	没有错误
11h	校验和错误
21h	器件码不同错误
22h	时钟模式不同错误
24h	位速率不能选择错误
25h	输入频率错误
26h	倍增比错误
27h	工作频率错误
29h	块号错误
2Ah	地址错误
2Bh	数据长度错误
51h	擦除错误
52h	未擦除错误
53h	编程错误
54h	选择处理错误
80h	命令错误
FFh	位速率调整确认错误

39.10.7 ID 码等待状态

在 ID 码等待状态下能使用的主机命令一览表如表 39.15 所示。

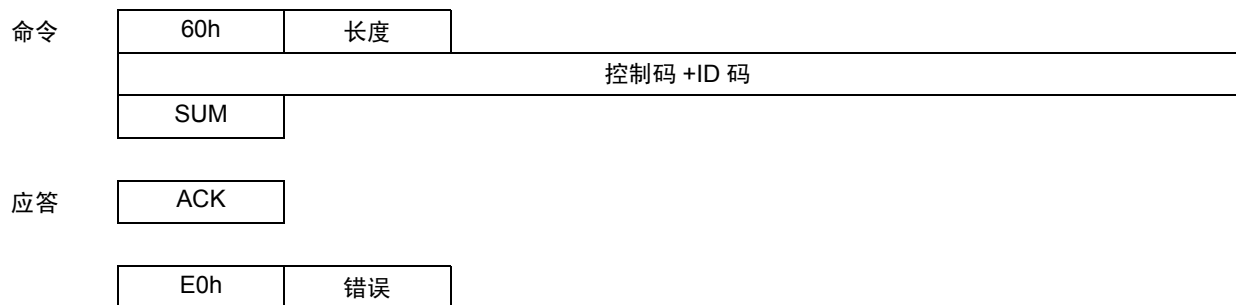
表 39.15 ID 码检查主机命令

主机命令名	功能
ID 码检查	进行 ID 码的检查。

如果主机发送未定义命令，RX210 就发送命令错误应答。有关命令错误的内容，请参照“39.10.6 查询设定主机命令的等待状态”。

(1) ID 码检查

如果主机发送 ID 码检查命令，RX210 就将 ROM 中的控制码、ID 码和主机送来的代码进行比较，并且送回比较结果。



长度（1 字节）：ID 码的字节数（是固定值，为“16”。）

ID 码（16 字节）：控制码（1 字节）+ ID 码（15 字节）

SUM（1 字节）：校验和

ACK（1 字节）：ACK 码

26h：对编程 / 擦除状态转移的应答

错误（1 字节）：错误码

11h：校验和错误

61h：ID 码不同

63h：ID 码不同 [擦除错误]

因 ID 码不同而发生擦除错误。

39.10.8 编程 / 擦除主机命令等待状态

在编程 / 擦除主机命令等待状态下能使用的主机命令一览表如表 39.16 所示。

表 39.16 编程 / 擦除主机命令

主机命令名	功能
用户引导区编程选择	选择用户引导区的编程。
用户区编程选择	选择用户区的编程。
256 字节编程	256 字节的编程
擦除选择	选择擦除。
块擦除	擦除块数据。
存储器读	读存储器。
用户引导区校验和	用户引导区的校验和
用户区校验和	用户区的校验和
用户引导区空白检查	用户引导区的空白检查
用户区空白检查	用户区的空白检查
读锁定位状态	读锁定位。
锁定位编程	对锁定位进行编程。
锁定位有效	设定为锁定位保护有效。
锁定位无效	设定为锁定位保护无效。
引导程序状态查询	查询 RX210 的状态。

如果主机发送未定义命令，RX210 就发送命令错误应答。有关命令错误的内容，请参照“39.10.6 查询设定主机命令的等待状态”

在进行 ROM 的编程时，主机在发送编程选择命令（选择用户区编程 / 用户引导区编程）后发送 256 字节编程命令。如果主机发送编程选择命令，RX210 就进入编程数据等待状态（参照“39.10.2 引导模式的状态转移”）。如果在编程数据等待状态下主机发送 256 字节编程命令，RX210 就对 ROM 进行数据编程。如果主机在将编程目标地址设定为“FFFF FFFFh”后发送 256 字节编程命令，RX210 就判断为编程结束，转移到编程 / 擦除主机命令等待状态。

在擦除 ROM 时，主机在发送擦除选择命令后发送块擦除命令。如果主机发送擦除选择命令，RX210 就进入擦除块指定等待状态（参照“39.10.2 引导模式的状态转移”）。如果在擦除块指定等待状态下主机发送块擦除命令，RX210 就对 ROM 进行块擦除。如果主机在将块号设定为“FFh”后发送块擦除命令，RX210 就判断为擦除结束，转移到编程 / 擦除主机命令等待状态。

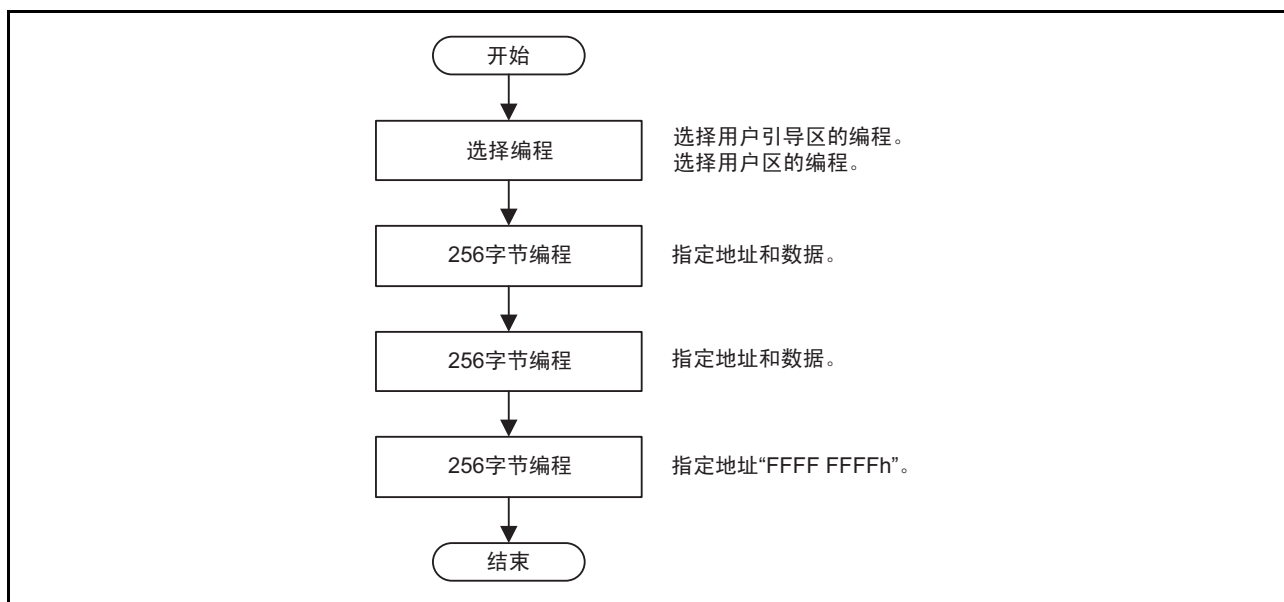


图 39.29 引导模式的 ROM 编程方法

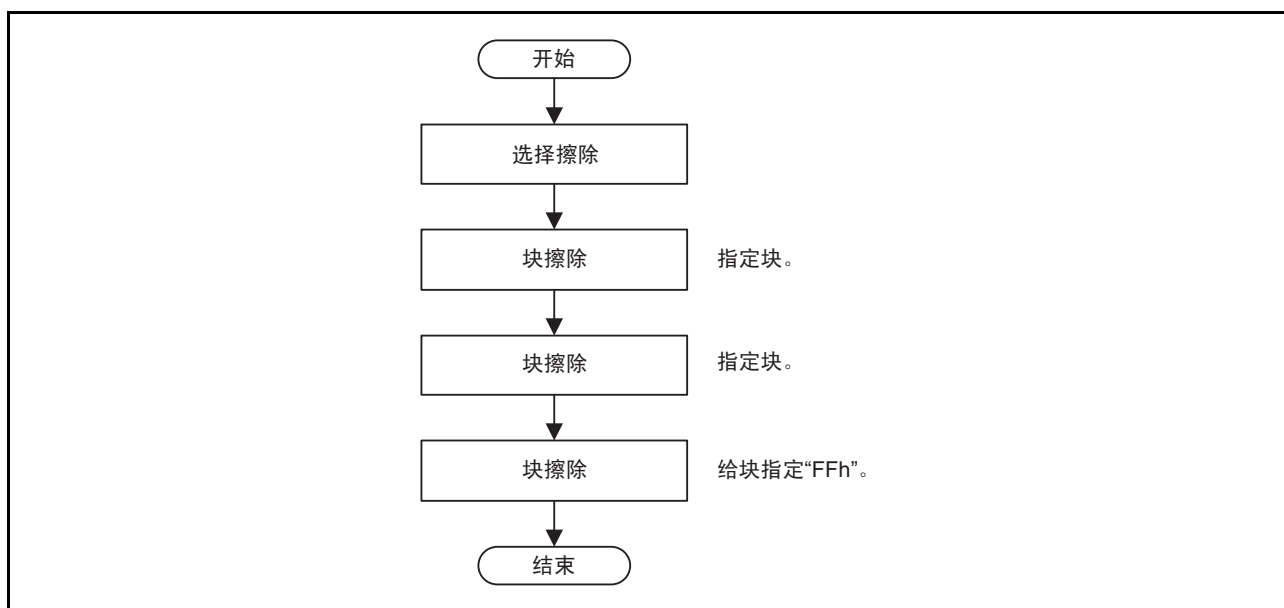


图 39.30 引导模式的 ROM 擦除方法

以下说明各主机命令的详细内容。说明内容中的“命令”是从主机发送到 RX210 的命令，“应答”是从 RX210 发送到主机的应答。“校验和”是指在合计发送的各字节时，为了得到“00h”而算出的字节数据。

(1) 用户引导区编程选择

如果主机发送用户引导区编程选择命令，RX210 就选择用户引导区编程程序，进入编程数据等待状态。

命令	42h
应答	06h

(2) 用户区编程选择

如果主机发送用户区编程选择命令，RX210 就选择用户区编程程序，进入编程数据等待状态。

命令	43h
应答	06h

(3) 256 字节编程

如果主机发送 256 字节编程命令，RX210 就进行 ROM 的编程。如果 ROM 的编程正常结束，RX210 就发送应答（06h）。如果在编程处理过程中发生错误，RX210 就发送错误应答（D0h）。

命令	50h	编程地址		
	数据	数据	...	数据
	SUM			
应答	06h			
错误应答	D0h	错误		

编程地址（4 字节）： 编程目标地址
 在进行编程时调整为以 256 字节为边界的地址。
 在指定编程结束时发送“FFFF FFFFh”。

数据（256 字节）： 编程数据
 给不需要编程的字节指定“FFh”。
 在指定编程结束时不需要发送数据。
 （按照编程地址 →SUM 的顺序进行发送）

SUM（1 字节）： 校验和

错误（1 字节）： 错误码
 11h: 校验和错误
 2Ah: 地址错误（地址不在指定的区域内）
 53h: 发生编程错误，不能进行编程。

(4) 擦除选择

如果主机发送擦除选择命令，RX210 就选择擦除程序，进入擦除块指定等待状态。

命令	48h
----	-----

应答	06h
----	-----

(5) 块擦除

如果主机发送块擦除命令，RX210 就擦除 ROM。如果 ROM 擦除正常结束，RX210 就发送应答（06h）。如果在擦除处理过程中发生错误，RX210 就发送错误应答（D8h）。

命令	58h	长度	块	SUM
----	-----	----	---	-----

应答	06h
----	-----

错误应答	D8h	错误
------	-----	----

长度（1 字节）：块的数据字节数（是固定值，为“1”。）

块（1 字节）：要擦除的擦除块号
在指定擦除的结束时发送“FFh”。

SUM（1 字节）：校验和

错误（1 字节）：错误码
11h: 校验和错误
29h: 块号错误（块号不正确）
51h: 发生擦除错误，不能进行擦除。

(6) 存储器读

如果主机发送存储器读命令，RX210 就读 ROM。如果读操作正常结束，RX210 就发送由存储器读命令指定地址的数据。如果没有执行读操作，RX210 就发送错误应答（D2h）。

命令	52h	长度	区域	读起始地址
	读长度			SUM

应答	52h	读长度		
	数据	数据	...	数据
	SUM			

错误应答	D2h	错误
------	-----	----

长度（1 字节）：区域、读地址、读长度的数据总字节数

区域（1 字节）：读对象的区域
00h: 用户引导区
01h: 用户区

读起始地址（4 字节）：读对象区的起始地址

读长度（4 字节）：读取数据的长度（以字节为单位）

SUM（1 字节）：校验和

数据（1 字节）：从 ROM 读取的数据

错误（1 字节）：错误码
11h: 校验和错误
2Ah: 地址错误
 • 在区域的选择中指定“00h”和“01h”以外的值。
 • 读起始地址不在指定的区域内。
2Bh: 长度错误
 • 在读长度的选择中指定“00h”。
 • 读长度超过区域的长度。
 • 由读起始地址和读长度计算出的地址不在区域内。

(7) 用户区校验和

如果主机发送用户区校验和命令，RX210 就发送以字节为单位将用户区的数据相加后的结果（校验和）。

命令	4Ah
----	-----

应答	5Ah	长度	区域的校验和	SUM
----	-----	----	--------	-----

长度（1 字节）：区域的校验和的字节数（是固定值，为“4”。）

MAT 的校验和（4 字节）：用户引导区的校验和结果

SUM（1 字节）：校验和（应答数据的校验和）

(8) 用户区校验和

如果主机发送用户区校验和命令，RX210 就发送以字节为单位将用户区的数据相加后的结果（校验和）。

命令	4Bh			
应答	5Bh	长度	区域的校验和	SUM

长度（1 字节）：区域的校验和的字节数（是固定值，为“4”。）

MAT 的校验和（4 字节）：用户区的校验和结果
 用户区也包含用于验证调试功能的键码。
 必须注意：相加的结果包含键码值。

SUM（1 字节）：校验和（应答数据的校验和）

(9) 用户引导区空白检查

如果主机发送用户引导区空白检查命令，RX210 就检查用户引导区是否全部为擦除状态。如果用户引导区全部为擦除状态，RX210 就发送应答（06h）。如果用户引导区有未擦除区域，RX210 就发送错误应答（CCh →52h）。

命令	4Ch	
应答	06h	
错误应答	CCh	52h

(10) 用户区空白检查

如果主机发送用户区空白检查命令，RX210 就检查用户区是否全部为擦除状态。如果用户区全部为擦除状态，RX210 就发送应答（06h）。如果用户区有未擦除区域，RX210 就发送错误应答（CDh →52h）。

命令	4Dh	
应答	06h	
错误应答	CDh	52h

(11) 读锁定位状态

如果主机发送读锁定位状态命令，RX210 就读锁定位。如果读操作正常结束，RX210 就发送由读锁定位状态命令指定的地址的数据。如果没有执行读操作，RX210 就发送错误应答（F1h）。

命令	71h	长度	区域	中位地址	高位地址	最高位地址	SUM
应答	状态						
错误应答	F1h	错误					

长度（1 字节）：区域、中位地址、高位地址、最高位地址的数据总字节数（RX210 为固定值“4”。）

区域（1 字节）：读对象的区域
01h: 用户区

中位地址（1 字节）：指定块的结束地址的中位地址（8～15 位）

高位地址（1 字节）：指定块的结束地址的高位地址（16～23 位）

最高位地址（1 字节）：指定块的结束地址的最高位地址（24～31 位）

SUM（1 字节）：校验和

状态（1 字节）：当 bit6 为“0”时，为锁定状态。
当 bit6 为“1”时，为非锁定状态。

错误（1 字节）：错误码
11h: 校验和错误
2Ah: 地址错误（地址不在指定的区域内）

(12) 锁定位编程

如果主机发送锁定位编程命令，RX210 就对锁定位进行编程，将指定块设定为锁定状态。如果被正常锁定，RX210 就发送应答（06h）。否则，RX210 就发送错误应答（F7h）。

命令	77h	长度	区域	中位地址	高位地址	最高位地址	SUM
应答	06h						
错误应答	F7h	错误					

长度（1 字节）：区域、中位地址、高位地址、最高位地址的数据总字节数（RX210 为固定值“4”。）

区域（1 字节）：锁定对象的区域
01h: 用户区

中位地址（1 字节）：指定块的结束地址的中位地址（8～15 位）

高位地址（1 字节）：指定块的结束地址的高位地址（16～23 位）

最高有效位地址（1 字节）：指定块的结束地址的最高位地址（24～31 位）

SUM（1 字节）：校验和

错误（1 字节）：错误码
11h: 校验和错误
2Ah: 地址错误（地址不在指定的区域内）
53h: 发生编程错误，不能设定为锁定状态。

(13) 锁定位有效

如果主机发送锁定位有效命令，RX210 就将锁定位位置为有效。

命令

7Ah

应答

06h

(14) 锁定位无效

如果主机发送锁定位无效命令，RX210 就将锁定位位置为无效。

命令

75h

应答

06h

(15) 引导程序状态查询

请参照“39.10.6 查询设定主机命令的等待状态”。

39.11 On-chip 调试器的 ID 码保护

这是禁止连接 on-chip 调试器的功能。在连接 on-chip 调试器时，使用写在 ROM 中的控制码和 ID 码，决定 on-chip 调试器 ID 码保护的有效或者无效以及判断 on-chip 调试器的 ID 码保护。当 ID 码保护有效时，将从 on-chip 调试器送来的代码和 ROM 中的控制码、ID 码进行比较。如果相同，就允许连接 on-chip 调试器；否则就不能连接 on-chip 调试器。但是，如果控制码为“52h”并且 ID 码从 ID 码 1 到 ID 码 7 开始依次设定“50h”、“72h”、“6Fh”、“74h”、“65h”、“63h”、“74h”，就不判断 ID 码而总是视为不同，禁止连接 on-chip 调试器。当控制码和 ID 码都为“FFh”时，就不判断 ID 码而视为相同，允许连接 on-chip 调试器。闪存的 ID 码的结构和图 39.26 相同。

表 39.17 on-chip 调试器的 ID 码保护的规格

控制码	ID 码	保护状态	连接 on-chip 调试器时的运行
FFh	FFh、.....、FFh（全部为“FFh”）	保护无效	不判断控制码和 ID 码，总是视为和 ID 码相同，允许连接 on-chip 调试器。
52h	50h、72h、6Fh、74h、65h、63h、74h	保护有效	不判断控制码和 ID 码，总是视为和 ID 码不同，禁止连接 on-chip 调试器。
上述以外	上述以外	保护有效	ID 码相同：结束 on-chip 调试器的验证，允许连接 on-chip 调试器。 ID 码不同：再次转移到 ID 码等待状态。

39.12 ROM 码保护

ROM 码保护是在使用 PROM 编程器时禁止读和改写闪存的功能。闪存的 ROM 码是 32 位数据。ROM 码的结构如图 39.31 所示，必须以 32 位为单位设定 ROM 码。

要解除 ROM 码保护时，必须在引导模式或者用户编程模式中擦除包括 ROM 码的用户区的 EB00 块。

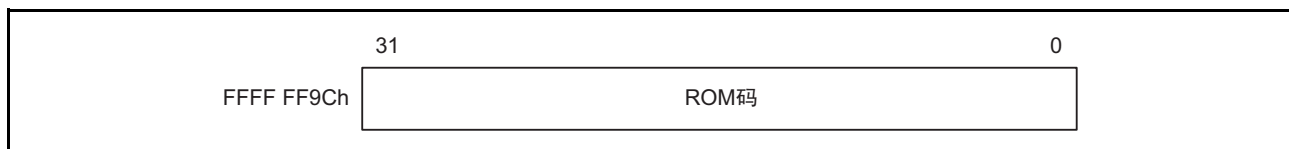


图 39.31 ROM 码的结构

表 39.18 ROM 码保护的规格

ROM 码	保护状态	连接 PROM 编程器时的运行
0000 0000h	ROM 码保护有效（ROM 码保护 1）	禁止读和改写用户区和用户引导区。
0000 0001h	ROM 码保护有效（ROM 码保护 2）	禁止读用户区和用户引导区。
上述以外	ROM 码保护无效	允许读或者改写用户区和用户引导区。

39.13 使用时的注意事项

(1) 编程 / 擦除挂起的对象区

编程 / 擦除挂起中的区域的保存数据为不定值。必须注意：不能执行编程 / 擦除挂起对象区的指令以及读数据，以免因读到不确定的数据而发生误动作。

(2) 编程 / 擦除挂起引起的中止

如果通过编程 / 擦除挂起命令中止编程 / 擦除处理，就必须通过恢复命令结束处理。在发行恢复命令后的 20 μ s 以内（FCLK=25MHz 时），不能重新发行编程 / 擦除命令。

(3) 追加编程的禁止

不能对同一区域进行 2 次或者 2 次以上的编程。要改写已编程的区域时，必须擦除该区域。

(4) 编程 / 擦除过程中的复位

不能在编程和擦除过程中发生复位（RES# 引脚复位、LVD0 ~ LVD2、独立看门狗定时器复位、看门狗定时器复位、软件复位），否则有可能损坏闪存。如果不得已而要从 RES# 输入复位，就必须在至少经过 tRESWF（“41. 电特性”）的复位输入期间后解除复位。

要在编程和擦除过程中通过 FRESETR.FRESET 位对 FCU 进行复位或者通过 WDT 的上溢进行内部复位时，必须保持 tRESW2（参照“41. 电特性”）时间的复位状态。不能在对 FCU 进行复位期间读 ROM。

(5) 编程 / 擦除过程中的非屏蔽中断的禁止

如果在编程 / 擦除过程中发生非屏蔽中断（NMI 引脚中断、LVD1、LVD2、振荡停止检测、独立看门狗定时器、看门狗定时器），就从 ROM 取向量，读到不确定的数据。因此，不能在对 ROM 进行编程或者擦除过程中发生非屏蔽中断。

(6) 编程 / 擦除过程中的中断向量的分配

如果在编程 / 擦除过程中发生中断，就可能从 ROM 取向量。为了避免从 ROM 取向量，可以通过 CPU 的中断表寄存器（INTB）将取中断向量目标表设定在 ROM 以外的区域。

(7) 低速运行模式 1 和低速运行模式 2 中的编程和擦除

如果通过运行功率控制寄存器（OPCCR）选择低速运行模式 1 或者低速运行模式 2，就不能对闪存进行编程和擦除。

(8) 编程 / 擦除的异常结束

如果因编程 / 擦除过程中的复位、FRESETR.FRESET 位引起的复位以及错误检测引起的命令锁定状态，使编程 / 擦除无法正常结束，锁定位就可能变为“0”（保护状态）。此时，必须在将 FPROTR.FPROTCN 位置“1”的状态下发行块擦除命令，擦除锁定位。

(9) 编程 / 擦除过程中的禁止事项

在编程 / 擦除过程中给闪存内部加有高电压，为了防止损坏闪存，不能进行以下的操作：

- 使 RX210 的电源电压低于工作电压。
- 更新 FWEPROR.FLWE[1:0] 位的值。
- 通过设定 SYSCR0.ROME 位来更改运行模式。
- 更新 OPCCR.OPCM[2:0] 位的值。
- 更改 SCKCR3 寄存器的时钟源选择位。
- 通过设定 RSTCKCR.RSTCKEN 位，将从睡眠模式返回时的时钟源转换置为有效。
- 更改 FlashIF 时钟（FCLK）的分频比。
- 转移到全模块时钟停止模式、软件待机模式或者深度软件待机模式。

40. E2 数据闪存（保存数据的闪存）

RX210 群内置用于保存 8K 字节数据的闪存（E2 数据闪存）。

本章说明 E2 数据闪存，有关 ROM 请参照“39. ROM（保存代码的闪存）”。

40.1 概要

E2 数据闪存的规格如表 40.1 所示，ROM 和 E2 数据闪存外围的框图如图 40.1 所示。

表 40.1 E2 数据闪存的规格

项目	内容
存储空间	数据区：8K 字节
经由外围总线的读操作	在进行字存取或者字节存取时，以 4 个 FCLK 周期进行读操作。
编程 / 擦除方式	<ul style="list-style-type: none"> • 内置对 ROM/E2 数据闪存进行改写的专用定序器（FCU）。 • 能通过给 FCU 发行命令，对 ROM/E2 数据闪存进行编程或者擦除。
BGO（后台操作）功能	<ul style="list-style-type: none"> • 在对 E2 数据闪存进行编程或者擦除的期间，能执行分配在 ROM 区的程序。 • 在对 ROM 进行编程或者擦除的期间，CPU 能执行分配在 ROM/E2 数据闪存以外区域的程序。
挂起 / 恢复功能	<ul style="list-style-type: none"> • 中止对 E2 数据闪存进行的编程或者擦除，CPU 能读 E2 数据闪存区（挂起）。 • 在中止后，能重新开始对 E2 数据闪存进行编程或者擦除（恢复）。
编程 / 擦除单位	<ul style="list-style-type: none"> • 数据区的编程单位：2 字节或者 8 字节 • 数据区的擦除单位：128 字节
空白检查功能	<ul style="list-style-type: none"> • 能执行用于确认 E2 数据闪存擦除状态的空白检查命令。 • 能进行空白检查的区域为 2 字节或者 2K 字节。
板上编程 (3 种)	通过引导模式改写
	<ul style="list-style-type: none"> • 使用异步串行接口（SCI1）。 • 自动调整通信速度。 • 能改写用户引导区。
	通过用户引导模式改写
保护功能	软件保护功能
	错误保护功能
编程时间 / 擦除时间 / 改写次数	请参照“41. 电特性”。

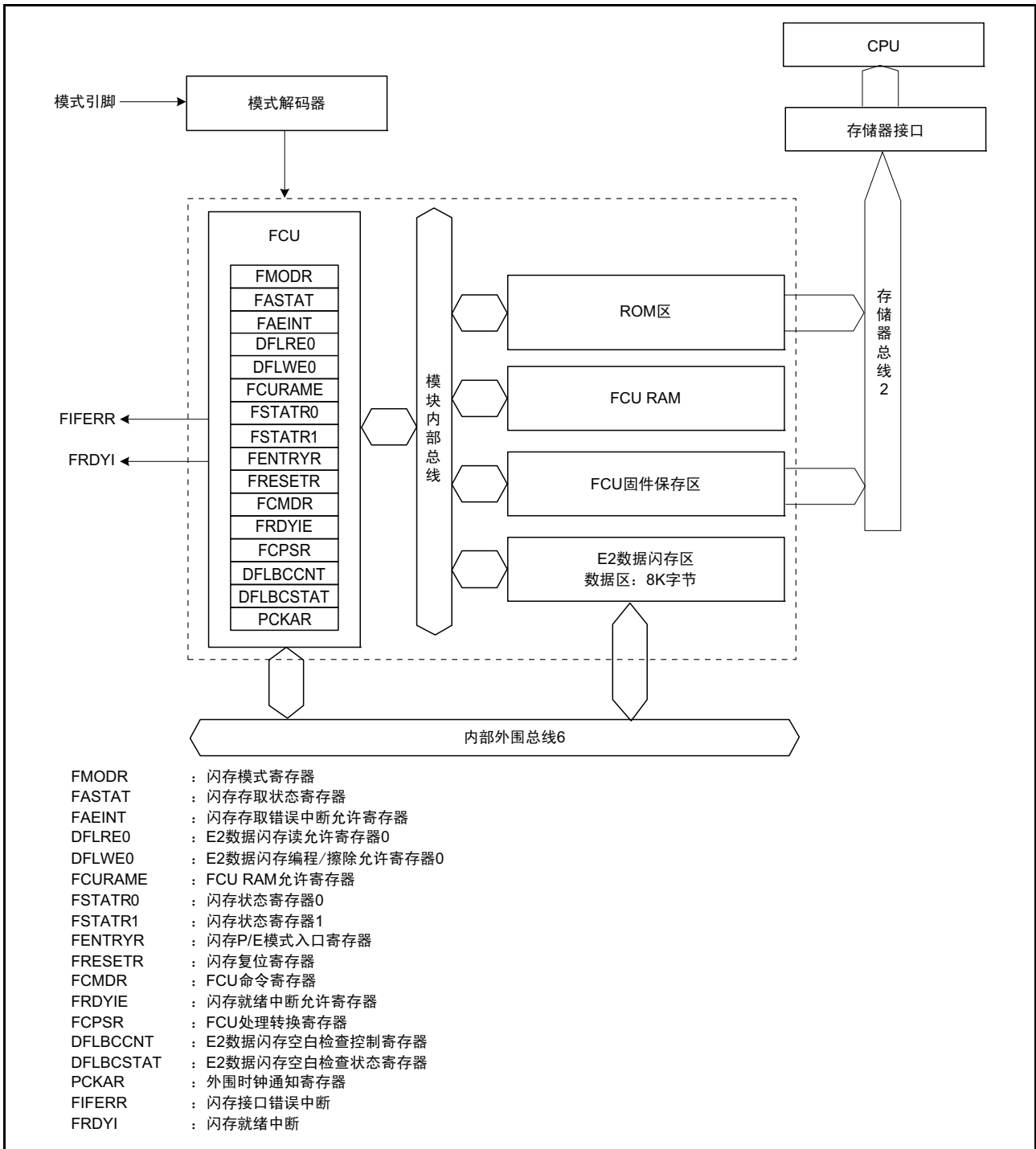


图 40.1 E2 数据闪存的框图

E2 数据闪存的相关输入 / 输出引脚如表 40.2 所示。

表 40.2 E2 数据闪存的相关输入 / 输出引脚

引脚名	输入 / 输出	功能
MD	输入	设定运行模式。
PC7	输入	设定引导模式、用户引导模式。
P30/RXD1	输入	用于引导模式。SCI1 的接收数据（用于主机通信）。
P26/TXD1	输出	用于引导模式。SCI1 的接收数据（用于主机通信）。

40.2 寄存器说明

部分寄存器也有 ROM 的相关位，本章只说明 E2 数据闪存相关位的功能。有关由 ROM/E2 数据闪存的公共位构成的寄存器（FRDYIE、FCURAME、FSTATR0、FSTATR1、FRESETR、FCMDR、FCPSR、PCKAR、FWEPROR）以及 ROM 专用位功能的详细内容，请参照“39. ROM（保存代码的闪存）”的“39.2 寄存器说明”。

P/E 表示编程 / 擦除。

40.2.1 闪存模式寄存器（FMODER）

地址 007F C402h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	FRDMD	—	—	—	—
复位后的值	0	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b3-b0	—	保留位	读写值都为“0”。	R/W
b4	FRDMD	FCU 读模式选择位	0: 存储区读方式 在要转移到 E2 数据闪存锁定位读模式时进行设定。E2 数据闪存没有锁定位，如果在转移到锁定位读模式后读 E2 数据闪存区，就会读到不确定的数据。 1: 寄存器读方式 在使用空白检查命令时进行设定。	R/W
b7-b5	—	保留位	读写值都为“0”。	R/W

在使用空白检查命令时，必须将 FRDMD 位置“1”。

在内部 ROM 无效模式中，FMODER 寄存器的读数据为“00h”，不能写。

FRDMD 位（FCU 读模式选择位）

此位选择是进行 E2 数据闪存锁定位读模式的转移处理还是进行空白检查处理。

在读 ROM 的锁定位时，使用 FRDMD 位选择锁定位的读法（参照“39. ROM（保存代码的闪存）”）。

40.2.2 闪存存取状态寄存器（FASTAT）

地址 007F C410h

b7	b6	b5	b4	b3	b2	b1	b0
ROMAE	—	—	CMDLK	DFLAE	—	DFLRPE	DFLWPE
复位后的值	0	0	0	0	0	0	0

位	符号	位名	功能	R/W
b0	DFLWPE	E2 数据闪存的编程 / 擦除保护违反位	0: 不发行违反 DFLWE0 寄存器设定的 E2 数据闪存的编程 / 擦除命令 1: 发行违反 DFLWE0 寄存器设定的 E2 数据闪存的编程 / 擦除命令	R/(W) (注 1)
b1	DFLRPE	E2 数据闪存的读保护违反位	0: 不读违反 DFLRE0 寄存器设定的 E2 数据闪存 1: 读违反 DFLRE0 寄存器设定的 E2 数据闪存	R/(W) (注 1)
b2	—	保留位	读写值都为“0”。	R/W
b3	DFLAE	E2 数据闪存的存取违反位	0: 不违反 E2 数据闪存的存取 1: 违反 E2 数据闪存的存取	R/(W) (注 1)
b4	CMDLK	FCU 命令锁定位	0: FCU 不处于命令锁定状态 1: FCU 处于命令锁定状态	R
b6-b5	—	保留位	读写值都为“0”。	R/W
b7	ROMAE	ROM 存取违反位	请参照“39. ROM（保存代码的闪存）”。	R/(W) (注 1)

注 1. 为了将标志置“0”，只能在读“1”后写“0”。

在内部 ROM 无效时，FASTAT 寄存器的读数据为“00h”，不能写。如果 FASTAT 寄存器的任意位变为“1”，FCU 就进入命令锁定状态（参照“40.7.2 错误保护”）。为了解除命令锁定状态，需要在将 FASTAT 寄存器置“10h”后，给 FCU 发行状态寄存器清除命令。

DFLWPE 位（E2 数据闪存的编程 / 擦除保护违反位）

此位表示有没有违反由 DFLWE0 寄存器设定的编程 / 擦除保护。

如果 DFLWPE 位变为“1”，FSTATR0.ILGLERR 位就变为“1”并且 FCU 进入命令锁存状态。

有关 FSTATR0 寄存器，请参照“39.2.6 闪存状态寄存器 0（FSTATR0）”。

[为“1”的条件]

- 对通过 DFLWE0 寄存器设定为禁止编程或者擦除的 E2 数据闪存区发行编程 / 擦除命令。

[为“0”的条件]

- 在读“1”后写“0”时

DFLRPE 位（E2 数据闪存的读保护违反位）

此位表示有没有违反由 DFLRE0 寄存器设定的读保护。

如果 DFLRPE 位变为“1”，FSTATR0.ILGLERR 位就变为“1”并且 FCU 进入命令锁存状态。

有关 FSTATR0 寄存器，请参照“39.2.6 闪存状态寄存器 0（FSTATR0）”。

[为“1”的条件]

- 对通过 DFLRE0 寄存器设定为禁止读的 E2 数据闪存区发行读存取命令。

[为“0”的条件]

- 在读“1”后写“0”时

DFLAE 位（E2 数据闪存的存取违反位）

此位表示有没有违反 E2 数据闪存的存取。

如果 DFLAE 位变为“1”，FSTATR0.ILGLERR 位就变为“1”，并且 FCU 进入命令锁定状态。

有关 FSTATR0 寄存器，请参照“39.2.6 闪存状态寄存器 0（FSTATR0）”。

[为“1”的条件]

- 在 FENTRYR.FENTRYD 位为“1”时并且在 E2 数据闪存 P/E 正常模式中，对 E2 数据闪存区发行读存取命令。
- 在 FENTRYD 位为“0”的状态下，对 E2 数据闪存区发行写存取命令。
- 在 FENTRYR.FENTRY0 位为“1”的状态下，对 E2 数据闪存区发行存取命令。

[为“0”的条件]

- 在读“1”后写“0”时

CMDLK 位（FCU 命令锁定位）

此位表示 FCU 处于命令锁定状态（参照“40.7.2 错误保护”）。

[为“1”的条件]

- 在 FCU 检测到错误并且转移到命令锁定状态后

[为“0”的条件]

- 在 FCU 发行状态寄存器清除命令后

40.2.3 闪存存取错误中断允许寄存器（FAEINT）

地址 007F C411h

	b7	b6	b5	b4	b3	b2	b1	b0
	ROMAE IE	—	—	CMDLK IE	DFLAEIE	—	DFLRPE IE	DFLWPE IE
复位后的值	1	0	0	1	1	0	1	1

位	符号	位名	功能	R/W
b0	DFLWPEIE	E2 数据闪存的编程 / 擦除保护违反中断允许位	0: 在 FASTAT.DFLWPE 位为“1”时, 不发生 FIFERR 中断请求。 1: 在 FASTAT.DFLWPE 位为“1”时, 发生 FIFERR 中断请求。	R/W
b1	DFLRPEIE	E2 数据闪存的读保护违反中断允许位	0: 在 FASTAT.DFLRPE 位为“1”时, 不发生 FIFERR 中断请求。 1: 在 FASTAT.DFLRPE 位为“1”时, 发生 FIFERR 中断请求。	R/W
b2	—	保留位	读写值都为“0”。	R/W
b3	DFLAEIE	E2 数据闪存的存取违反中断允许位	0: 在 FASTAT.DFLAE 位为“1”时, 不发生 FIFERR 中断请求。 1: 在 FASTAT.DFLAE 位为“1”时, 发生 FIFERR 中断请求。	R/W
b4	CMDLKIE	FCU 命令锁定中断允许位	0: 在 FASTAT.CMDLK 位为“1”时, 不发生 FIFERR 中断请求。 1: 在 FASTAT.CMDLK 位为“1”时, 发生 FIFERR 中断请求。	R/W
b6-b5	—	保留位	读写值都为“0”。	R/W
b7	ROMAEIE	ROM 存取违反中断允许位	请参照“39. ROM（保存代码的闪存）”。	R/W

在内部 ROM 无效时, FAEINT 寄存器的读数据为“00h”, 不能写。

DFLWPEIE 位（E2 数据闪存的编程 / 擦除保护违反中断允许位）

在发生 E2 数据闪存的编程 / 擦除保护违反并且 FASTAT.DFLWPE 位变为“1”时, 此位允许或者禁止 FIFERR 中断请求的发生。

DFLRPEIE 位（E2 数据闪存的读保护违反中断允许位）

在发生 E2 数据闪存的读保护违反并且 FASTAT.DFLRPE 位变为“1”时, 此位允许或者禁止 FIFERR 中断请求的发生。

DFLAEIE 位（E2 数据闪存的存取违反中断允许位）

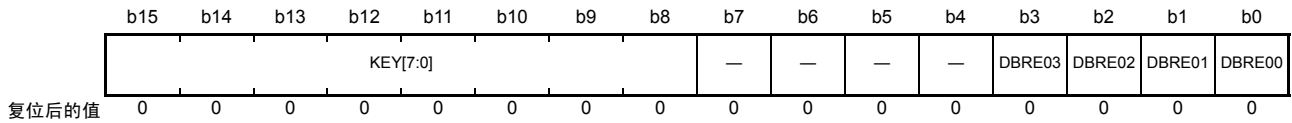
在发生 E2 数据闪存的存取违反并且 FASTAT.DFLAE 位变为“1”时, 此位允许或者禁止 FIFERR 中断请求的发生。

CMDLKIE 位（FCU 命令锁定中断允许位）

在发生 FCU 命令锁定并且 FASTAT.CMDLK 位变为“1”时, 此位允许或者禁止 FIFERR 中断请求的发生。

40.2.4 E2 数据闪存读允许寄存器 0（DFLRE0）

地址 007F C440h



位	符号	位名	功能	R/W
b0	DBRE00	DB00 ~ DB15 块的读允许位	0: 禁止读 1: 允许读	R/W
b1	DBRE01	DB16 ~ DB31 块的读允许位		R/W
b2	DBRE02	DB32 ~ DB47 块的读允许位		R/W
b3	DBRE03	DB48 ~ DB63 块的读允许位		R/W
b7-b4	—	保留位	读写值都为“0”。	R/W
b15-b8	KEY[7:0]	键码	控制能否改写 DBREj 位 (j=00 ~ 03)。	R/(W) (注 1)

注 1. 不保持写数据。

DFLRE0 寄存器允许或者禁止读数据区的 DB00 ~ DB63 块（参照图 40.3）。

只在通过字存取给高位字节写特定值时才能写此寄存器，但是不保持高位字节的写数据。

在内部 ROM 无效模式中，DFLRE0 寄存器的读数据为“0000h”，不能写。

DBREj 位（DBj 块的读允许位）（j=00 ~ 03）

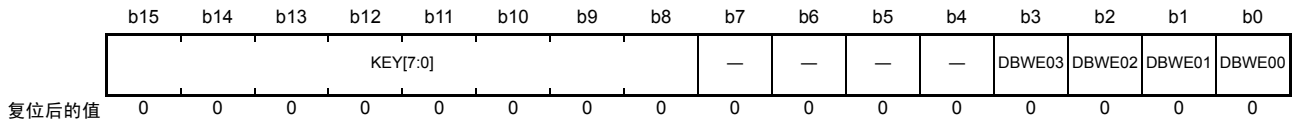
此位设定允许或者禁止读数据区的 DB00 ~ DB63 块。

DBREj 位用于控制 DB00 ~ DB63 块的读操作。

只在 KEY[7:0] 位为“2Dh”时才能通过字存取写 DBREj 位。

40.2.5 E2 数据闪存编程 / 擦除允许寄存器 0（DFLWE0）

地址 007F C450h



位	符号	位名	功能	R/W
b0	DBWE00	DB00 ~ DB15 块的编程 / 擦除允许位	0: 禁止编程和擦除 1: 允许编程和擦除	R/W
b1	DBWE01	DB16 ~ DB31 块的编程 / 擦除允许位		R/W
b2	DBWE02	DB32 ~ DB47 块的编程 / 擦除允许位		R/W
b3	DBWE03	DB48 ~ DB63 块的编程 / 擦除允许位		R/W
b7-b4	—	保留位	读写值都为“0”。	R/W
b15-b8	KEY[7:0]	键码	控制能否改写 DBWE _j 位 (j=00 ~ 03)。	R/(W) (注 1)

注 1. 不保持写数据。

DFLWE0 寄存器允许或者禁止对数据区的 DB00 ~ DB63 块（参照图 40.3）进行编程或者擦除。

只在通过字存取给高位字节写特定值时才能写此寄存器，但是不保持高位字节的写数据。

在内部 ROM 无效模式中，DFLWE0 寄存器的读数据为“0000h”，不能写。

DBWE_j 位（DB_j 块的编程 / 擦除允许位）（j=00 ~ 03）

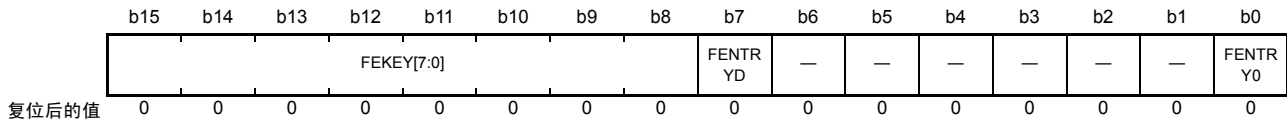
此位设定允许或者禁止对数据区的 DB00 ~ DB63 块进行编程或者擦除。

DBWE_j 位用于控制 DB00 ~ DB63 块的编程和擦除。

只在 KEY[7:0] 位为“1Eh”时才能通过字存取写 DBWE_j 位。

40.2.6 闪存 P/E 模式入口寄存器（FENTRYR）

地址 007F FFB2h



位	符号	位名	功能	R/W
b0	FENTRY0	ROM P/E 模式入口位 0	请参照“39. ROM（保存代码的闪存）”。	R/W
b6-b1	—	保留位	读写值都为“0”。	R/W
b7	FENTRYD	E2 数据闪存的 P/E 模式入口位	0: E2 数据闪存为读模式 1: E2 数据闪存为 P/E 模式	R/W
b15-b8	FEKEY[7:0]	键码	控制能否改写 FENTRYD 位和 FENTRY0 位。	R/(W) (注 1)

注 1. 不保持写数据。

为了将 ROM/E2 数据闪存设定为 P/E 模式，使 FCU 能接收命令，需要将 FENTRYD 位或者 FENTRY0 置“1”。但是，如果给多个位设定多个“1”，FSTAT0.ILGLERR 位就变为“1”，并且 FCU 进入命令锁定状态。

只在通过字存取给高位字节写特定值时才能写此寄存器，而在其他情况下写此寄存器时，此寄存器被初始化。不保持高位字节的写数据。

在内部 ROM 无效模式中，FENTRYR 寄存器的读数据为“0000h”，不能写。

通过复位或者将 FRESETR.FRESETR 位置“1”，对 FENTRYR 寄存器进行初始化。

有关 FSTAT0 寄存器，请参照“39.2.6 闪存状态寄存器 0（FSTAT0）”。

有关 FRESETR 寄存器，请参照“39.2.11 闪存复位寄存器（FRESETR）”。

FENTRYD 位（E2 数据闪存 P/E 模式入口位）

FENTRYD 位将 E2 数据闪存设定为 P/E 模式。

[写操作的有效条件（满足以下全部条件时）]

- 内部 ROM 为有效模式。
- FSTAT0.FRDI 位为“1”。
- 通过字存取给 FEKEY[7:0] 位写“AAh”。

[为“1”的条件]

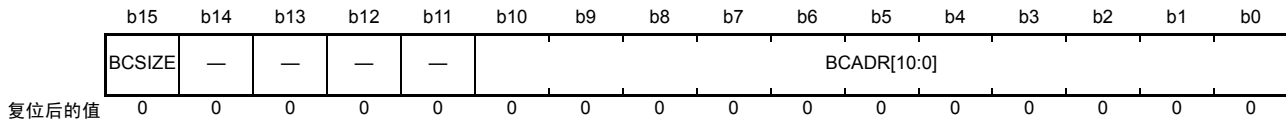
- 在满足写操作的有效条件并且 FENTRYR 寄存器为“0000h”的状态下，给 FENTRYD 位写“1”时

[为“0”的条件]

- 通过字节存取进行写操作时
- 在 FEKEY[7:0] 位不为“AAh”的状态下，通过字存取进行写操作时
- 在满足写操作的有效条件的状态下，给 FENTRYD 位写“0”时
- 在满足写操作的有效条件并且 FENTRYR 寄存器不为“0000h”的状态下，写 FENTRYR 寄存器时

40.2.7 E2 数据闪存空白检查控制寄存器（DFLBCCNT）

地址 007F FFCAh



位	符号	位名	功能	R/W
b10-b0	BCADR[10:0]	空白检查地址设定位	设定检查对象区的地址。	R/W
b14-b11	—	保留位	读写值都为“0”。	R/W
b15	BCSIZE	空白检查大小设定位	0: 空白检查对象区为 2 字节 1: 空白检查对象区为 2K 字节	R/W

在内部 ROM 无效时，DFLBCCNT 寄存器的读数据为“0000h”，不能写。

通过复位或者将 FRESETR.FRESET 位置“1”，对 DFLBCCNT 寄存器进行初始化。

有关 FRESETR 寄存器，请参照“39.2.11 闪存复位寄存器（FRESETR）”。

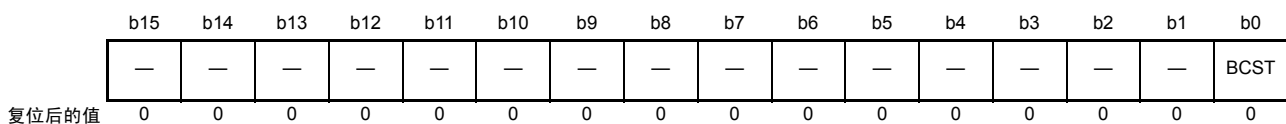
BCADR[10:0] 位（空白检查地址设定位）

在空白检查命令的检查对象区大小为 2 字节（BCSIZE 位为“0”）时，此位设定检查对象区的地址。BCADR[0] 位必须指定“0”。

当 BCSIZE 位为“0”时，检查对象区的起始地址是 DFLBCCNT 寄存器的设定值加上发行空白检查命令时指定的读 / 编程 / 擦除的允许 / 禁止块起始地址后的值。

40.2.8 E2 数据闪存空白检查状态寄存器（DFLBCSTAT）

地址 007F FFCEh



位	符号	位名	功能	R/W
b0	BCST	空白检查状态位	0: 空白检查对象区为擦除状态（空白） 1: 空白检查对象区为写“0”或者“1”的状态	R
b15-b1	—	保留位	读写值都为“0”。	R/W

在内部 ROM 无效时，DFLBCSTAT 寄存器的读数据为“0000h”，不能写。

通过复位或者将 FRESETR.FRESET 位置“1”，对 DFLBCSTAT 寄存器进行初始化。

有关 FRESETR 寄存器，请参照“39.2.11 闪存复位寄存器（FRESETR）”。

40.3 E2 数据闪存的存储区结构

RX210 群的 E2 数据闪存由 8K 字节的数据区构成，此区域的地址如图 40.2 所示。
在读、编程和擦除时，数据区的地址相同。

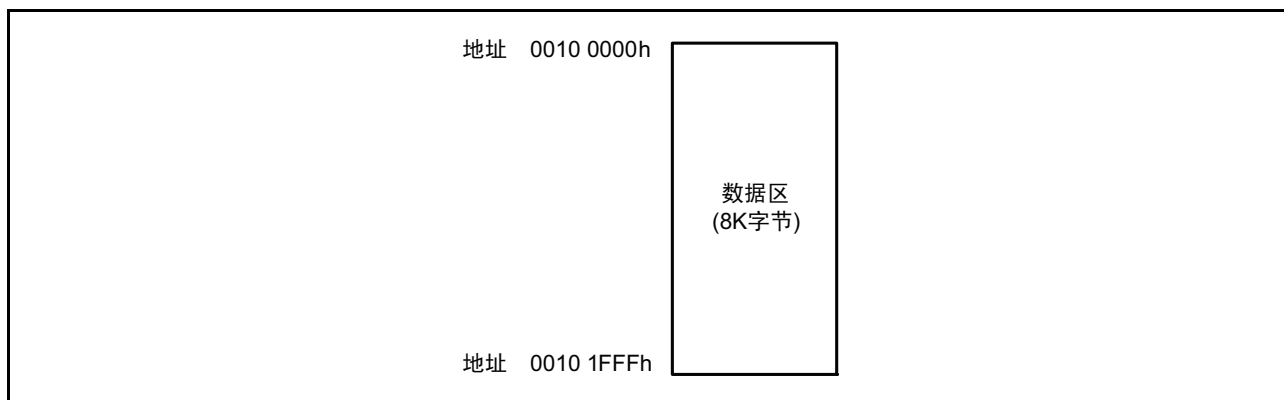


图 40.2 E2 数据闪存的数据区结构

40.4 块结构

数据区的擦除块结构如图 40.3 所示。数据区被分割为 128 字节（64 块），以块为单位进行擦除，以 2 字节或者 8 字节为单位进行编程。在 2 位编程时，以低位地址的 2 位为“0h”或者“2h”开始的 2 位字节为单位进行编程。在 8 位编程时，以低位地址的 3 位全部为“0”的 8 字节为单位进行编程。

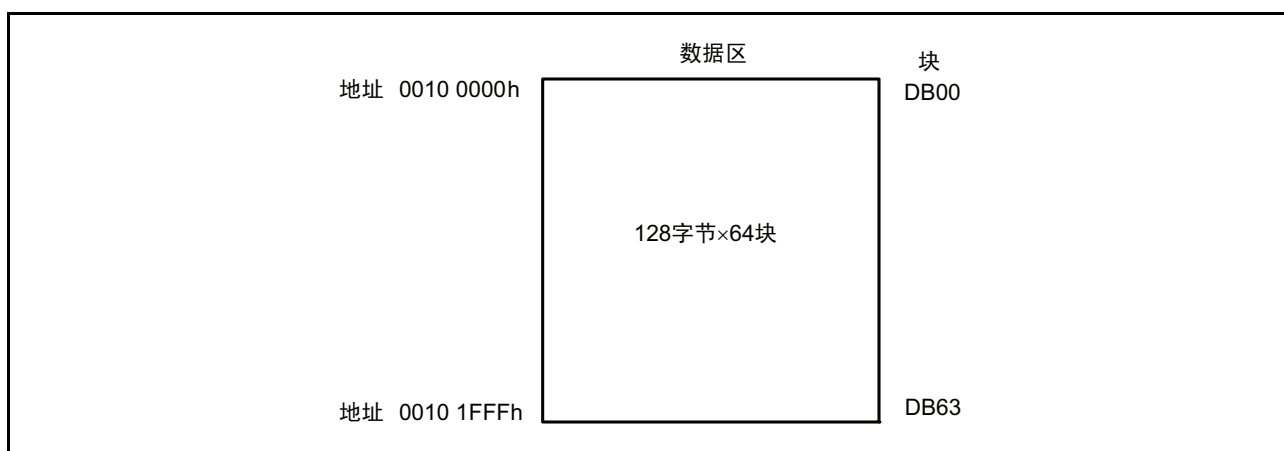


图 40.3 数据区的块分割

40.5 E2 数据闪存相关的运行模式

有关运行模式的转移，请参照“39.5 ROM 相关的运行模式”。

在引导模式、用户引导模式、单芯片模式（内部 ROM 有效）和内部 ROM 有效扩展模式中，能进行板上数据区的读、编程或者擦除。

各模式的不同点如表 40.3 所示。

表 40.3 各模式的不同点

项目	引导模式	用户引导模式	单芯片模式（内部 ROM 有效）/ 内部 ROM 有效扩展模式
编程 / 擦除环境	板上编程		
能编程和擦除的区域	数据区	数据区	数据区
块分割擦除	○（注 1）	○（注 1）	○
复位时的启动程序	引导程序	用户引导程序	用户程序

注 1. 有可能在启动时被全部擦除。此后能擦除特定块，详细内容请参照“39.10.4 ID 码保护（引导模式）”和“39.10.2 引导模式的状态转移”。

- 在引导模式中，主机能经由 SCI 对数据区进行编程、擦除和读。
- 在引导模式中，通过引导程序使用内部 RAM。因此，不保持内部 RAM 的数据。
- 从用户引导区域启动用户引导模式，并且通过任意接口对用户区或者数据区进行读写。

40.6 E2 数据闪存的编程 / 擦除

通过给编程 / 擦除的专用定序器（FCU）发行命令（FCU 命令），对 E2 数据闪存进行编程和擦除。FCU 有 5 种模式，通过在模式转移后发行编程和擦除命令，进行编程和擦除。

以下说明 E2 数据闪存的编程和擦除所需的模式转移和命令体系。这些内容在引导模式、用户引导模式、单芯片模式（内部 ROM 有效）和内部 ROM 有效扩展模式中通用。

40.6.1 FCU 的模式

FCU 有 5 种模式，通过写 FENTRYR 寄存器以及 FCU 命令进行模式的转移，FCU 的模式转移图如图 40.4 所示。

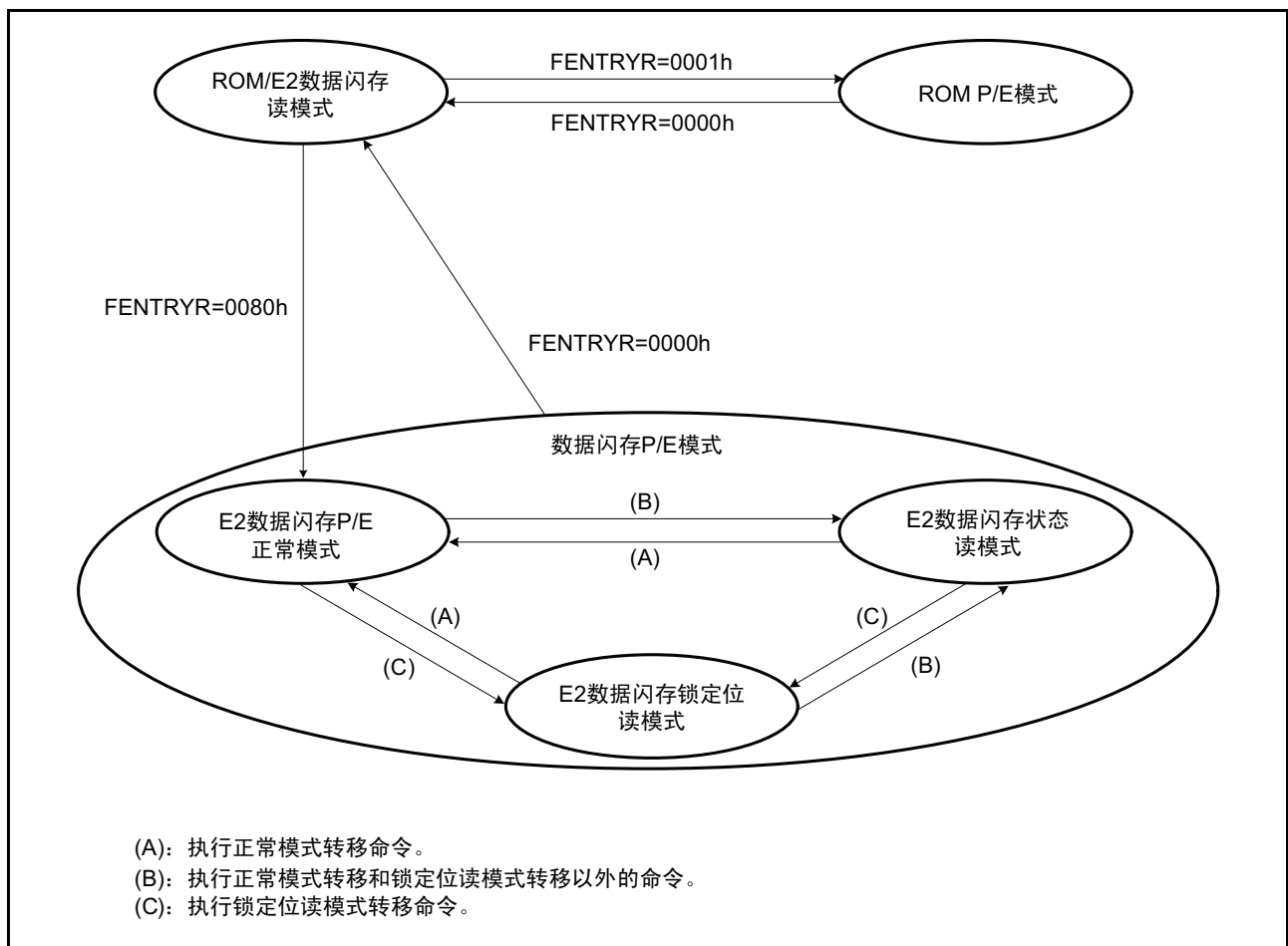


图 40.4 FCU 的模式转移图（E2 数据闪存相关）

40.6.1.1 ROM P/E 模式

ROM P/E 模式是对 ROM 进行编程和擦除的模式。

有关 ROM P/E 模式的详细内容，请参照“39.6.1.2 ROM P/E 模式”。

40.6.1.2 ROM/E2 数据闪存读模式

ROM/E2数据闪存读模式是能读ROM和E2数据闪存的模式，不接受FCU命令。如果将FENTRYR.FENTRYD位置“0”并且将FENTRYR.FENTRY0位置“0”，就转移到此模式。

40.6.1.3 E2 数据闪存 P/E 模式

E2 数据闪存 P/E 模式是对 E2 数据闪存进行编程和擦除的模式，不能读 E2 数据闪存。

E2数据闪存P/E模式有E2数据闪存P/E正常模式、E2数据闪存状态读模式、E2数据闪存锁定位读模式共3种。

(1) E2 数据闪存 P/E 正常模式

E2 数据闪存 P/E 正常模式是在对 E2 数据闪存进行编程或者擦除时最先转移的模式。如果在 ROM/E2 数据闪存读模式中将 FENTRYR.FENTRYD 位置“1”并且将 FENTRYR.FENTRY0 位置“0”，或者在 E2 数据闪存 P/E 模式中接受正常模式转移命令，就转移到此模式。能接受的命令如表 40.8 所示。

(2) E2 数据闪存状态读模式

E2 数据闪存状态读模式是能读 E2 数据闪存状态的模式。

如果在 E2 数据闪存 P/E 模式中接受正常模式转移和锁定位读模式转移以外的命令，就转移到此模式。FSTATR0.FRDY 位为“0”的状态以及发生错误后的命令锁定状态也是 E2 数据闪存状态读模式中的状态。能接受的命令如表 40.8 所示。

如果对 E2 数据闪存区发行读存取，就读 FSTATR0 寄存器的值。能高速读 ROM。

(3) E2 数据闪存锁定位读模式

E2 数据闪存锁定位读模式是能读 E2 数据闪存锁定位的模式。但是，因为 E2 数据闪存没有锁定位，所以不能读锁定位。

如果在 E2 数据闪存 P/E 模式中接受锁定位读模式转移命令，就转移到 E2 数据闪存锁定位读模式。能接受的命令如表 40.8 所示。

E2 数据闪存没有锁定位，如果对 E2 数据闪存区发行读存取，读数据就为不定值，但是不发生 E2 数据闪存的存取违反。能高速读 ROM。

40.6.2 FCU 命令一览表

FCU 命令有 FCU 模式转移命令和编程 / 擦除命令。对 E2 数据闪存进行编程或者擦除时能使用的 FCU 命令一览表如表 40.4 所示。

表 40.4 FCU 命令一览表（E2 数据闪存相关）

命令	功能
P/E 正常模式转移	转移到正常模式（参照“40.6.3 FCU 的模式和命令的关系”）。
状态读模式转移	转移到状态读模式（参照“40.6.3 FCU 的模式和命令的关系”）。
锁定位读模式转移（锁定位读 1）	转移到锁定位读模式（参照“40.6.3 FCU 的模式和命令的关系”）。
外围时钟通知	设定 FlashIF 时钟（FCLK）的频率。
编程	对 E2 数据闪存进行编程（2 字节或者 8 字节）。
块擦除	擦除 E2 数据闪存（以 128 字节为单位）。
P/E 挂起	中止编程和擦除。
P/E 恢复	重新开始编程和擦除。
状态寄存器清除	清除 FSTATR0.ILGLERR 位、FSTATR0.ERSERR 位和 FSTATR0.PRGERR 位以及解除命令锁定状态。

ROM 也使用空白检查命令以外的 FCU 命令。锁定位读 2 命令兼用 E2 数据闪存的空白检查命令。如果对 ROM 发行空白检查命令，就读 ROM 的锁定位。

通过写 E2 数据闪存区给 FCU 发行命令。

编程命令和空白检查命令的格式如表 40.5 所示。有关编程命令和空白检查命令以外的 FCU 命令格式，请参照“39. ROM（保存代码的闪存）”的“39.6.2 FCU 命令一览表”。

如果在特定条件下进行如表 40.5 所示的写存取，FCU 就进行各命令对应的处理。有关 FCU 的特定条件和命令的使用方法，请分别参照“40.6.3 FCU 的模式和命令的关系”和“40.6.4 FCU 命令的使用方法”。

表 40.5 FCU 命令的格式（E2 数据闪存专用命令）

命令	总线周期数	第 1 周期		第 2 周期		第 3 周期		第 4 ~ N+2 周期		第 N+3 周期	
		地址	数据	地址	数据	地址	数据	地址	数据	地址	数据
编程 (2 字节编程: N=1)	4	EA	E8h	EA	01h	WA	WDn	EA	WDn	EA	D0h
编程 (8 字节编程: N=4)	7	EA	E8h	EA	04h	WA	WDn	EA	WDn	EA	D0h
空白检查	2	EA	71h	BA	D0h	—	—	—	—	—	—

地址列 EA: E2 数据闪存区的地址
0010 0000h ~ 0010 1FFFh 的任意地址

WA: 编程数据的调整后的起始地址

BA: E2 数据闪存的读 / 编程 / 擦除的允许 / 禁止块地址
读 / 编程 / 擦除的允许 / 禁止对象块内的任意地址

数据列 WDn: 编程数据的第 n 个字 (n=1 ~ N)

40.6.3 FCU 的模式和命令的关系

FCU 的各模式能接受的 FCU 命令是固定的，而且能接受的命令也因这些模式的 FCU 状态而不同。

需要在进行 FCU 的模式转移后确认 FCU 的状态，然后发行 FCU 命令。

FCU 的模式 / 状态和能接受的命令如表 40.6 所示。如果发行不能接受的命令，FCU 就进入命令锁定状态（参照“40.7.2 错误保护”）。必须在转移到能接受的模式后，确认 FSTATR0.FRDY 位、FSTATR0.ILGLERR 位、FSTATR0.ERSERR 位、FSTATR0.PRGERR 位和 FSTATR1.FCUERR 位的值，然后发行 FCU 命令。能根据 FASTAT.CMDLK 位的值确认有没有发生错误，FASTAT.CMDLK 位的值是 FSTATR0.ILGLERR 位、FSTATR0.ERSERR 位、FSTATR0.PRGERR 位和 FSTATR1.FCUERR 位的值的逻辑或。

表 40.6 FCU 的模式 / 状态和能接受的命令的关系（E2 数据闪存 P/E 模式）

	P/E 正常模式			状态读模式									锁定位读模式		
	编程挂起中	擦除挂起中	其他状态	编程 / 擦除处理中	擦除挂起中的编程处理中	编程 / 擦除中止处理中	空白检查处理中	编程挂起中	擦除挂起中	命令锁定状态 (FRDY=0)	命令锁定状态 (FRDY=1)	其他状态	编程挂起中	擦除挂起中	其他状态
FSTATR0.FRDY 位	1	1	1	0	0	0	0	1	1	0	1	1	1	1	1
FSTATR0.SUSRDY 位	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
FSTATR0.ERSSPD 位	0	1	0	0	1	0/1	0	0	1	0/1	0/1	0	0	1	0
FSTATR0.PRGSPD 位	1	0	0	0	0	0/1	0	1	0	0/1	0/1	0	1	0	0
FASTAT.CMDLK 位	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
P/E 正常模式转移	○	○	○	×	×	×	×	○	○	×	×	○	○	○	○
状态读模式转移	○	○	○	×	×	×	×	○	○	×	×	○	○	○	○
锁定位读模式转移（锁定位读 1）	○	○	○	×	×	×	×	○	○	×	×	○	○	○	○
外围时钟通知	×	×	○	×	×	×	×	×	×	×	×	○	×	×	○
编程	×	△	○	×	×	×	×	×	△	×	×	○	×	△	○
块擦除	×	×	○	×	×	×	×	×	×	×	×	○	×	×	○
P/E 挂起	×	×	×	○	×	×	×	×	×	×	×	×	×	×	×
P/E 恢复	○	○	×	×	×	×	×	○	○	×	×	×	○	○	×
状态寄存器清除	○	○	○	×	×	×	×	○	○	×	○	○	○	○	○
空白检查	○	○	○	×	×	×	×	○	○	×	×	○	○	○	○

○：能接受、△：只能接受对非擦除中止块进行的编程、×：不能接受

40.6.4 FCU 命令的使用方法

FCU 命令的使用方法有：使用编程命令和块擦除命令对 E2 数据闪存进行编程和擦除的方法以及使用空白检查命令确认 E2 数据闪存的擦除状态的方法。有关 FCU RAM 的固件传送方法以及其他 FCU 命令的使用方法，请参照“39. ROM（保存代码的闪存）”的“39.6.4 FCU 命令的使用方法”。

(1) 外围时钟通知命令的使用方法

通知外围时钟的频率，详细内容请参照“39. ROM（保存代码的闪存）”的“39.6.4 FCU 命令的使用方法”。必须在将 FENTRYR.FENTRYD 位置“1”后设定 E2 数据闪存区内的地址。

(2) 编程方法

使用编程命令对 E2 数据闪存进行数据编程。

在编程命令的第 1 周期，通过字节存取给 E2 数据闪存区的地址写“E8h”，在第 2 周期，通过字节存取给 E2 数据闪存区的地址写编程字数（N）（注 1）。在命令的第 3～第 N+2 周期，必须通过字存取写编程数据。在第 3 周期的存取中，必须给编程对象区的起始地址写编程数据。在进行 2 字节编程时，必须将起始地址调整为以 2 字节为边界的地址。如果在通过 N 次字存取给 E2 数据闪存区的地址写编程数据后，在第 N+3 周期通过字节存取给 E2 数据闪存区的地址写“D0h”，FCU 就开始对 E2 数据闪存进行编程处理。能通过 FSTATR0.FRDY 位确认编程的结束。

如果在第 3 周期～第 N+2 周期中存取的区域包括不需要编程的地址，就必须将该地址的编程数据置“FFFFh”。如果要在将通过 DFLWE0 寄存器控制的编程 / 擦除保护功能设定为无效后进行编程，就必须将编程对象块的编程 / 擦除允许位置“1”。

E2 数据闪存的编程方法如图 40.5 所示。

注 1. 在进行 2 字节编程时，N 为“01h”，在进行 8 字节编程时，N 为“04h”。

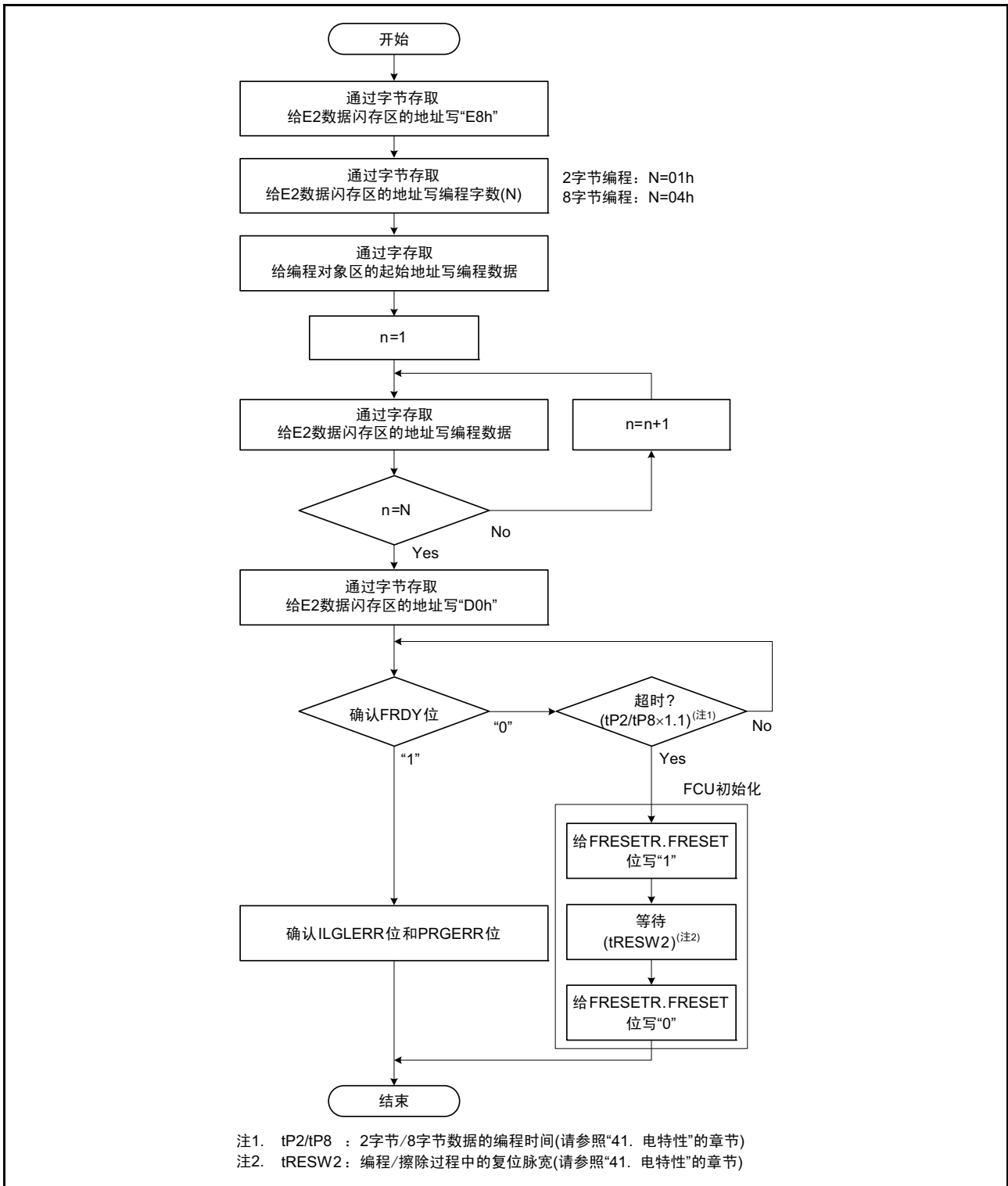


图 40.5 E2 数据闪存的编程

(3) 擦除方法

使用块擦除命令来擦除 E2 数据闪存。使用块擦除命令的擦除方法和 ROM 的擦除方法相同（参照“39. ROM（保存代码的闪存）”）。

必须注意：E2 数据闪存有 DFLWE0 寄存器控制的编程 / 擦除保护功能。如果要在将通过 DFLWE0 寄存器控制的保护功能设定为无效后进行擦除，就必须将擦除对象块的编程 / 擦除允许位置“1”。

(4) 空白检查

即使通过 CPU 读擦除状态的 E2 数据闪存，读取值也为不定值，因此需要使用空白检查命令确认擦除状态。在使用空白检查命令时，必须预先将 FMODR.FRDMMD 位置“1”，使空白检查命令处于可使用的状态，然后给 DFLBCCNT 寄存器设定检查对象区的大小和地址。当 DFLBCCNT.BCSIZE 位为“1”时，能在空白检查命令的第 2 周期对指定的整个擦除块（2K 字节）进行空白检查；当 BCSIZE 位为“0”时，能在空白检查命令的第 8 周期对指定的擦除块的起始地址和 DFLBCCNT 寄存器值相加后的地址开始的 2 字节区域进行空白检查。在空白检查命令的第 1 周期，通过字节存取给 E2 数据闪存区的地址写“71h”。在命令的第 2 周期，如果通过字节存取给包括空白检查对象区的擦除块内的任意地址写“D0h”，FCU 就开始进行 E2 数据闪存空白检查处理。能通过 FSTATR0.FRDY 位确认空白检查的结束。如果在结束空白检查后确认 DFLBCSTAT.BCST 位的值，就能确认检查对象区是为擦除状态还是为写“0”或者“1”的状态。

E2 数据闪存空白检查方法如图 40.6 所示。

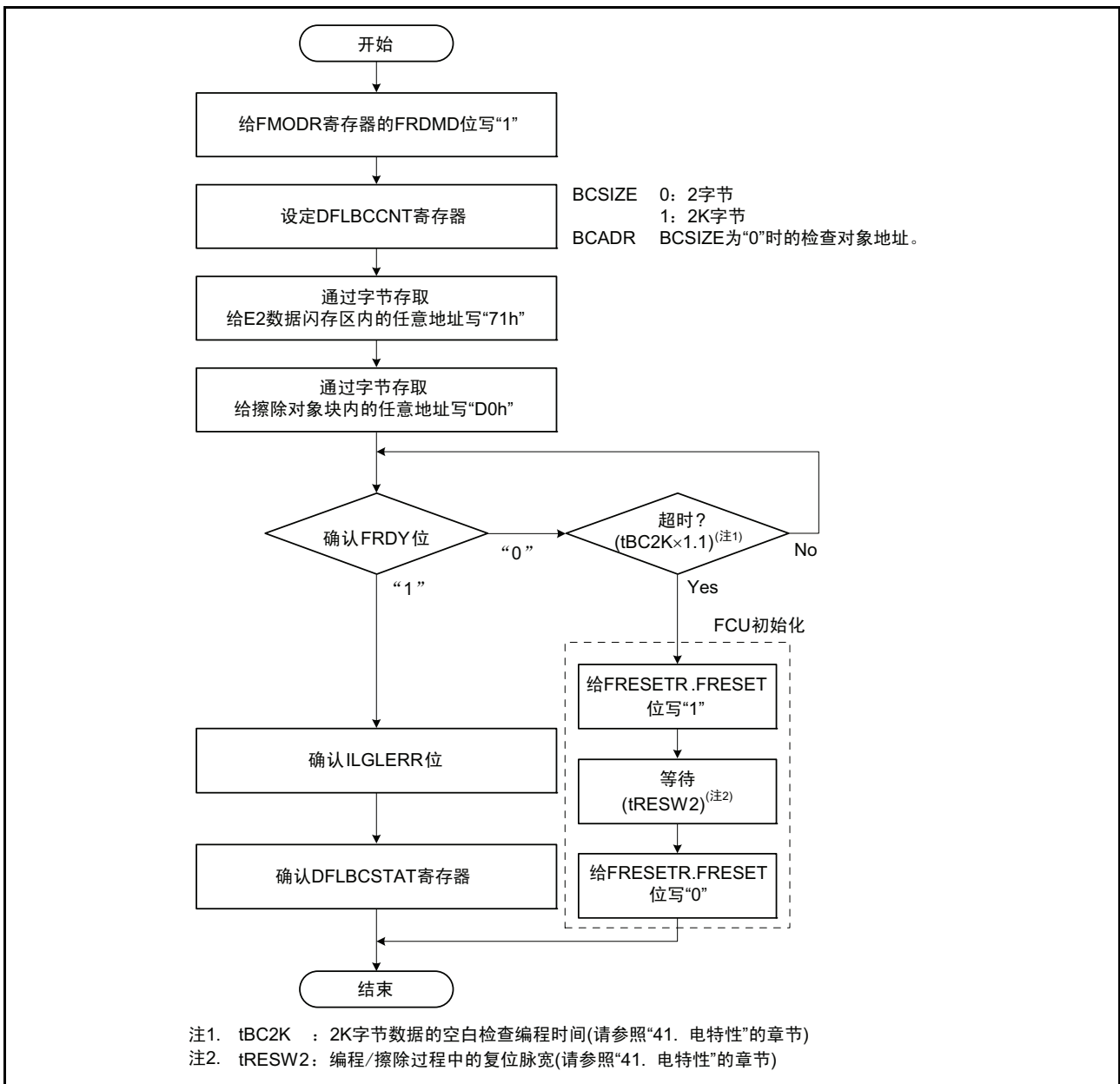


图 40.6 E2 数据闪存的空白检查

40.7 保护

E2 数据闪存的编程 / 擦除的保护有软件保护和错误保护。

40.7.1 软件保护

软件保护是通过设定控制寄存器禁止对 E2 数据闪存进行编程、擦除和读的状态。如果违反软件保护而对 E2 数据闪存发行编程 / 擦除命令或者读 E2 数据闪存，FCU 就检测到错误，进入命令锁定状态。

(1) 通过 FWEPROR 寄存器进行的保护

如果不将 FWEPROR.FLWE[1:0] 位置“01b”，就不能在任何模式中进行改写。

(2) 通过 FENTRYR 寄存器进行的保护

当 FENTRYR.FENTRYD 位为“0”时，FCU 进入 ROM/E2 数据闪存读模式。因为在 ROM/E2 数据闪存读模式中不接受 FCU 命令，所以 E2 数据闪存为禁止编程和擦除的状态。如果在 ROM/E2 数据闪存读模式中对 E2 数据闪存发行 FCU 命令，FCU 就检测到非法命令错误，进入命令锁定状态（参照“40.7.2 错误保护”）。

(3) 通过 DFLWE0 寄存器进行的保护

当 DFLWE0.DBWE_j (j=00 ~ 03) 位为“0”时，数据区的 DB_j 块为禁止编程和擦除的状态。如果在 DBWE_j 位为“0”的状态下对 DB_j 块进行编程或者擦除，FCU 就检测到写保护违反，进入命令锁定状态（参照“40.7.2 错误保护”）。

(4) 通过 DFLRE0 寄存器进行的保护

当 DFLRE0.DBRE_j (j=00 ~ 03) 位为“0”时，数据区的 DB_j 块为禁止读的状态。如果在 DBRE_j 位为“0”的状态下读 DB_j 块，FCU 就检测到读保护违反，进入命令锁定状态（参照“40.7.2 错误保护”）。

40.7.2 错误保护

错误保护是因误发行 FCU 命令以及发生禁止的存取而导致 FCU 在检测到误动作后禁止接受 FCU 命令的状态（命令锁定状态）。如果 FCU 进入命令锁定状态（FASTAT.CMDLK 位为“1”），状态位（FSTATR0.ILGLERR 位、FSTATR0.ERSERR 位、FSTATR0.PRGERR 位、FSTATR1.FCUERR 位、FASTAT.DFLAE 位、FASTAT.DFLRPE 位、FASTAT.DFLWPE 位）中的任意位或者多个位就为“1”，禁止对 E2 数据闪存进行编程和擦除。要解除命令锁定状态时，需要在 FASTAT 寄存器为“10h”的状态下发行状态寄存器清除命令。

当 FAEINT.CMDLKIE 位为“1”时，如果 FCU 进入命令锁定状态（FASTAT.CMDLK 位为“1”），就产生闪存接口错误（FIFERR）中断。当 FAEINT 寄存器的 E2 数据闪存相关的中断允许位（DFLAEIE 位、DFLRPEIE 位、DFLWPEIE 位）为“1”时，即使 FASTAT 寄存器的对应位（DFLAE 位、DFLRPE 位、DFLWPE 位）变为“1”，也产生 FIFERR 中断。

E2 数据闪存相关的错误保护内容和错误检测后的状态位的值（FSTATR0.ILGLERR 位、FSTATR0.ERSERR 位、FSTATR0.PRGERR 位、FASTAT.DFLAE 位、FASTAT.DFLRPE 位、FASTAT.DFLWPE 位）的关系如表 40.7 所示。有关 ROM/E2 数据闪存通用的错误保护内容（FENTRYR 设定错误、大部分非法命令、擦除错误、编程错误、FCU 错误），请参照“39. ROM（保存代码的闪存）”的“39.8.2 错误保护”。

如果在编程或者擦除处理过程中发行挂起以外的命令，就转移到命令锁定状态，但是 FCU 继续进行编程或者擦除处理。在此状态下无法通过发行 P/E 挂起命令来中止编程或者擦除。如果在命令锁定状态下发行命令，ILGLERR 位就变为“1”。

表 40.7 错误保护一览表（E2 数据闪存专用）

分类	内容	ILGERR	ERSERR	PRGERR	DFLAE	DFLRPE	DFLWPE
非法命令	在编程命令的第 2 周期指定“04h”、“40h”以外的数据。	1	0	0	0	0	0
	在 FENTRYR.FENTRYD 位为“1”的状态下，对 E2 数据闪存区发行锁定位编程命令。	1	0	0	0	0	0
E2 数据闪存的存取违反	在 FENTRYD 位为“1”时并且在 E2 数据闪存 P/E 正常模式中，对 E2 数据闪存区发行读存取命令。	1	0	0	1	0	0
	在 FENTRYD 位为“0”的状态下，对 E2 数据闪存区发行写存取命令。	1	0	0	1	0	0
	在 FENTRYR.FENTRY0 位为“1”的状态下，对 E2 数据闪存区发行写存取命令。	1	0	0	1	0	0
E2 数据闪存的读保护违反	对通过 DFLRE0 寄存器设定为禁止读的 E2 数据闪存区发行读存取命令。	1	0	0	0	1	0
E2 数据闪存的写保护违反	对通过 DFLWE0 寄存器设定为禁止编程和擦除的 E2 数据闪存区发行编程 / 块擦除命令。	1	0	0	0	0	1

40.8 引导模式

在引导模式中，能从主机发送控制命令和编程数据，并且对数据区进行编程或者擦除。有关引导模式的系统结构和使用方法的详细内容，请参照“40.8 引导模式”。以下说明 E2 数据闪存固有的命令。

40.8.1 查询设定主机命令

E2 数据闪存固有的查询设定主机命令一览表如表 40.8 所示。在“39.10.6 查询设定主机命令的等待状态”和“图 39.27 用户区 / 用户引导区的查询设定主机命令的使用例子”所示的流程中，“查询区域编程信息”使用数据区有无查询命令和数据区信息查询命令。

表 40.8 查询设定主机命令（E2 数据闪存固有）

主机命令名	功能
数据区有无查询	查询是否有数据区。
数据区信息查询	查询数据区的个数、起始地址和结束地址。

以下说明各命令的详细内容。说明内容中的“命令”是从主机发送到 RX210 的命令，“应答”是从 RX210 发送到主机的应答。“校验和”是指在合计 RX210 发送的各字节时，为了得到“00h”而算出的字节数据。

(1) 数据区有无查询

如果主机发送数据区有无查询命令，RX210 就发送表示有数据区的的信息。

命令	2Ah			
应答	3Ah	长度	有无区域	SUM

长度（1 字节）：有无区域的字符数（是固定值，为“1”。）

有无区域（1 字节）：有无数据区（是固定值，为“21h”。）

21h: 有数据区

SUM（1 字节）：校验和

(2) 数据区信息查询

如果主机发送数据区信息查询命令，RX210 就发送数据区的区域数和地址的信息。

命令	2Bh		
应答	3Bh	长度	区域数
	区域的起始地址		
	区域的结束地址		
	区域的起始地址		
	区域的结束地址		
	...		
	区域的起始地址		
	区域的结束地址		
	SUM		

长度（1 字节）：区域数、区域的起始地址、区域的结束地址的数据总字节数

区域数（1 字节）：数据区的区域数（连续区域为 1 个区域）

区域的起始地址（4 字节）：数据区的起始地址

区域的最后地址（4 字节）：数据区的结束地址

SUM（1 字节）：校验和

数据区的块结构信息包括在擦除块信息查询命令（参照“39.10.6 查询设定主机命令的等待状态”）的应答中。

40.8.2 编程 / 擦除主机命令

E2 数据闪存固有的编程 / 擦除主机命令一览表如表 40.9 所示。E2 数据闪存固有的主机命令只有数据区校验和命令以及数据区空白检查命令，而编程命令、擦除命令、读命令和 ROM 通用。

在对数据区进行编程时，在发行用户区编程选择命令后，用 256 字节编程命令给编程地址指定数据区的地址。在擦除数据区时，在发行擦除选择命令后，用块擦除命令指定数据区的擦除块。数据区的擦除块信息包括在擦除块查询命令的应答中。在读数据区时，用存储器读命令选择用户区并且给读对象地址指定数据区的地址。

有关用户区编程选择命令、用户引导区编程选择命令、256 字节编程命令、擦除选择命令、块擦除命令和存储器读命令的详细内容，请参照“39.10.8 编程 / 擦除主机命令等待状态”。有关擦除块信息查询命令的详细内容，请参照“39.10.6 查询设定主机命令的等待状态”。

表 40.9 编程 / 擦除主机命令（E2 数据闪存固有）

主机命令名	功能
数据区校验和	数据区的校验和
数据区空白检查	数据区的空白检查

以下说明各命令的详细内容。说明内容中的“命令”是从主机发送到 RX210 的命令，“应答”是从 RX210 发送到主机的应答。“校验和”是指在合计 RX210 发送的各字节时，为了得到“00h”而算出的字节数据。

(1) 数据区校验和

如果主机发送数据区校验和命令，RX210 就发送以字节为单位将数据区的数据相加后的结果（校验和）。

命令	61h			
应答	71h	长度	区域校验和	SUM

长度（1 字节）：区域校验和的字节数（是固定值，为“4”。）

区域校验和（4 字节）：数据区的校验和结果

SUM（4 字节）：校验和（应答数据的校验和）

(2) 数据区空白检查

如果主机发送数据区空白检查命令，RX210 就检查数据区是否为全部擦除状态。如果数据区为全部擦除状态，RX210 就发送应答（06h）。如果数据区有未擦除的区域，RX210 就发送错误应答（E2h → 52h）。

命令	62h	
应答	06h	
错误应答	E2h	52h

40.9 使用时的注意事项

(1) 启动复位后的数据区保护状态

因为 DFLRE0 寄存器和 DFLWE0 寄存器的初始值为“0000h”，所以启动复位后的数据区为禁止读、编程和擦除的状态。如果需要读数据区，就必须在设定 DFLRE0 寄存器后存取数据区。如果需要对数据区进行编程或者擦除，就必须在设定 DFLWE0 寄存器后发行编程 / 擦除的 FCU 命令。如果不设定寄存器而进行读、编程或者擦除，FCU 就检测到错误，进入命令锁定状态。

(2) 其他注意事项

其他注意事项和 ROM 相同，请参照“39. ROM（保存代码的闪存）”的“39.13 使用时的注意事项”。但是，在 E2 数据闪存的情况下，编程 / 擦除的注意事项里有空白检查。

41. 电特性

41.1 绝对最大额定值

表 41.1 绝对最大额定值

条件: VSS=AVSS0=VREFL=VREFL0=0V

项目	符号	额定值	单位
电源电压	VCC	-0.3 ~ +6.5	V
输入电压 (5V 容许支持端口 (注1) 以外)	V _{in}	-0.3 ~ VCC+0.3	V
输入电压 (5V 容许支持端口 (注1))	V _{in}	-0.3 ~ +6.5	V
基准电源电压	VREFH、VREFH0	-0.3 ~ VCC+0.3	V
模拟电源电压	AVCC0 (注2)	-0.3 ~ +6.5	V
模拟输入电压	V _{AN}	-0.3 ~ VCC+0.3	V
工作温度	T _{opr}	-40 ~ +105	°C
保存温度	T _{stg}	-55 ~ +125	°C

【使用时的注意事项】如果在使用 LSI 时超过绝对最大额定值，就可能造成 LSI 的永久性破坏。

为了防止因噪声引起的误动作，必须在各 VCC 引脚和 VSS 引脚、AVCC0 引脚和 AVSS0 引脚以及 VREFH0 引脚和 VREFL0 引脚之间插入频率特性良好的电容器。尽量使用最短距离和粗布线，将 0.1μF 的电容器连接在电源引脚的附近，VCL 引脚通过 0.1μF 的电容器连接在 VSS。电容器必须连接在引脚的附近。

注 1. 端口 12、13、16、17 支持 5V 容许。

注 2. AVCC0 必须连接 VCC。在不使用 A/D 转换器和 D/A 转换器时，不能将 AVCC0、VREFH、VREFH0、AVSS0、VREFL、VREFL0 引脚置为开路。AVCC0、VREFH 和 VREFH0 引脚必须连接 VCC，AVSS0、VREFL 和 VREFL0 引脚必须连接 VSS。

41.2 DC 特性

表 41.2 DC 特性 (1)

条件: $V_{CC}=AV_{CC0}=2.7 \sim 5.5V$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、 $T_a=-40 \sim +105^\circ C$

项目		符号	min	typ	max	单位	测量条件
施密特触发 输入电压	RIIC 输入引脚 (SMBus 除外、5V 容许)	V_{IH}	$V_{CC} \times 0.7$	—	5.8	V	
	端口 12、13、16、17 (5V 容许)		$V_{CC} \times 0.8$	—	5.8		
	其他引脚		$V_{CC} \times 0.8$	—	$V_{CC} + 0.3$		
	RIIC 输入引脚 (SMBus 除外)	V_{IL}	-0.3	—	$V_{CC} \times 0.3$		
	RIIC 输入引脚除外		-0.3	—	$V_{CC} \times 0.2$		
	RIIC 输入引脚 (SMBus 除外)	ΔV_T	$V_{CC} \times 0.05$	—	—		
	RIIC 输入引脚除外		$V_{CC} \times 0.1$	—	—		
输入电平电压 (施密特触发输 入引脚除外)	MD 引脚	V_{IH}	$V_{CC} \times 0.9$	—	$V_{CC} + 0.3$	V	
	EXTAL、WAIT#、XCIN、 RSPI 输入引脚		$V_{CC} \times 0.8$	—	$V_{CC} + 0.3$		
	D0 ~ D15		$V_{CC} \times 0.7$	—	$V_{CC} + 0.3$		
	RIIC 输入引脚 (SMBus)		2.1	—	$V_{CC} + 0.3$		
	MD 引脚	V_{IL}	-0.3	—	$V_{CC} \times 0.1$		
	EXTAL、WAIT#、XCIN、 RSPI 输入引脚		-0.3	—	$V_{CC} \times 0.2$		
	D0 ~ D15		-0.3	—	$V_{CC} \times 0.3$		
	RIIC 输入引脚 (SMBus)		-0.3	—	0.8		

表 41.3 DC 特性 (2)

条件: $V_{CC}=AV_{CC0}=1.62 \sim 2.7V$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、 $T_a=-40 \sim +105^\circ C$

项目		符号	min	typ	max	单位	测量条件
施密特触发 输入电压	端口 12、13、16、17 (5V 容许)	V_{IH}	$V_{CC} \times 0.8$	—	5.8	V	
	其他引脚		$V_{CC} \times 0.8$	—	$V_{CC} + 0.3$		
	全部引脚	V_{IL}	-0.3	—	$V_{CC} \times 0.2$		
	全部引脚		ΔV_T	$V_{CC} \times 0.05$	—		
输入电平电压 (施密特触发输 入引脚除外)	MD 引脚	V_{IH}	$V_{CC} \times 0.9$	—	$V_{CC} + 0.3$	V	
	EXTAL、WAIT#、XCIN、 RSPI 输入引脚		$V_{CC} \times 0.8$	—	$V_{CC} + 0.3$		
	D0 ~ D15		$V_{CC} \times 0.7$	—	$V_{CC} + 0.3$		
	MD 引脚	V_{IL}	-0.3	—	$V_{CC} \times 0.1$		
	EXTAL、WAIT#、XCIN、 RSPI 输入引脚		-0.3	—	$V_{CC} \times 0.2$		
	D0 ~ D15		-0.3	—	$V_{CC} \times 0.3$		

表 41.4 DC 特性 (3)

条件: $V_{CC}=AVCC0=1.62 \sim 5.5V$ 、 $V_{SS}=AVSS0=VREFL=VREFL0=0V$ 、 $T_a=-40 \sim +105^\circ C$

项目		符号	min	typ	max	单位	测量条件
输入漏泄电流	RES#、MD 引脚、NMI	I_{in}	—	—	1.0	μA	$V_{in}=0V$ 、 V_{CC}
三态漏泄电流 (断态)	端口 4	I_{Tsil}	—	—	1.0	μA	$V_{in}=0V$ 、 V_{CC}
	5V 容许支持端口、端口 4 以外		—	—	0.2		
	5V 容许支持端口		—	—	1.0		$V_{in}=0V$ 、 $5.8V$
输入电容	全部输入引脚 (端口 12、13、16、17、端口 4、 端口 A1、A3、A4、端口 E 以外)	C_{in}	—	—	15	pF	$V_{in}=0V$ 、 $f=1MHz$ 、 $T_a=25^\circ C$
	端口 12、13、16、17、端口 4、 端口 A1、A3、A4、端口 E		—	—	30		

表 41.5 DC 特性 (4)

条件: $V_{CC}=AVCC0=1.62 \sim 5.5V$ 、 $V_{SS}=AVSS0=VREFL=VREFL0=0V$ 、 $T_a=-40 \sim +105^\circ C$

项目	符号	VCC						单位	测量条件	
		1.62 ~ 2.7V		2.7 ~ 4.0V		4.0 ~ 5.5V				
		min	max	min	max	min	max			
输入上拉 MOS 电流	全部端口 (端口 35 除外)	I_p	-5	-150	-10	-200	-50	-400	μA	$V_{in}=0V$

表 41.6 DC 特性 (5)

条件: $V_{CC}=AVCC0=2.7 \sim 5.5V$ 、 $V_{SS}=AVSS0=VREFL=VREFL0=0V$ 、 $T_a=-40 \sim +105^\circ C$

项目			符号	min	typ (注 5)	max	单位	测量条件
消耗电流 (注 1)	高速运行模式	最大运行时 (注 2)	I_{CC}	—	—	55	mA	ICLK=50MHz PCLKB=25MHz FCLK=25MHz BCLK=25MHz
		正常运行时		—	14	—		
		正常运行时 (注 3)		10	—	—		
		睡眠模式		—	10.5	35		ICLK=50MHz PCLKB=25MHz FCLK=25MHz BCLK=25MHz
		全模块时钟停止模式		—	7.5	20		
		BGO 运行时的增加量 (注 4)		—	25	—		FCLK=32MHz

注 1. 消耗电流值是指全部的输出引脚为无负载状态并且内部上拉 MOS 为 OFF 状态时的值。

注 2. 外围功能为提供时钟的状态，BGO 运行除外。

注 3. 外围功能为时钟停止的状态，BGO 运行除外。

注 4. 这是在程序执行过程中对 ROM 或者用于保存数据的闪存进行数据编程 / 擦除时的增加量。

注 5. 这是 $V_{CC}=3.3V$ 时的值。

表 41.7 DC 特性 (6)

条件: $V_{CC}=AV_{CC0}=1.62 \sim 5.5V$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、 $T_a=-40 \sim +105^{\circ}C$

项目		符号	typ (注5)	max	单位	测量条件	
消耗电流 (注1)	中速运行模式 A、B	最大运行时 (注2)	I_{CC}	—	40	mA	ICLK=32MHz PCLKB=32MHz FCLK=32MHz BCLK=16MHz
		正常运行时 (注3)		10	—		
		睡眠模式		8.5	30		
		全模块时钟停止模式		6.5	15		
		BGO 运行时 的增加量 (注4)		25	—		
	低速运行模式 1	中速运行模式 A		20	—		
		中速运行模式 B		—	—		
	低速运行模式 1	最大运行时 (注2)		—	7		ICLK=1MHz
		正常运行时 (注3)		0.8	—		
	低速运行模式 2	最大运行时 (注2)		—	3		ICLK=32kHz
正常运行时 (注3)		0.03	—				

注 1. 消耗电流值是指全部的输出引脚为无负载状态并且内部上拉 MOS 为 OFF 状态时的值。

注 2. 外围功能为提供时钟的状态, BGO 运行除外。

注 3. 外围功能为时钟停止的状态, BGO 运行除外。

注 4. 这是在程序执行过程中对 ROM 或者用于保存数据的闪存进行数据编程 / 擦除时的增加量。

注 5. 这是 $V_{CC}=3.3V$ 时的值。

表 41.8 DC 特性 (7)

条件: $V_{CC}=AV_{CC0}=1.62 \sim 5.5V$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、 $T_a=-40 \sim +105^{\circ}C$

项目	符号	typ	max	单位	测量条件
容许总功耗 (注1)	I_{CC}	—	350	mW	$T_a < 85^{\circ}C$
		—	150		$T_a = 85 \sim 105^{\circ}C$

注 1. 这是整个芯片 (包括输出电流) 的总功耗。

表 41.9 DC 特性 (8)

条件: $V_{CC}=AV_{CC0}=1.62 \sim 5.5V$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、 $T_a=-40 \sim +105^\circ C$

项目			符号	typ (注3)	max		单位	测量条件
					$T_a=85^\circ C$	$T_a=105^\circ C$		
消耗电流 (注1)	软件待机 模式 (注2)	提供闪存电源、提供 HOCO 电 源、POR 的低功耗功能无效 (SOFTCUT=000b)	I_{CC}	175	480	550	μA	
		提供闪存电源、不提供 HOCO 电源、POR 的低功耗功能有效 (SOFTCUT=110b)		3.0	130	150		
		不提供闪存电源、不提供 HOCO 电源、POR 的低功耗功能有效 (SOFTCUT=111b)		2.0	120	140		
	深度软件 待机模式	不提供闪存电源、不提供 HOCO 电源、POR 的低功耗功能有效		0.45	20	25		
	电压检测电路运行, 因 POR 的低功耗功能无效 引起增加。			1.4	—	—		
	RTC 运行的增加量 (低 CL 时)			0.8	—	—		
	RTC 运行的增加量 (标准 CL 时)			2.0	—	—		

注 1. 消耗电流值是指全部的输出引脚为无负载状态并且内部上拉 MOS 为 OFF 状态时的值。

注 2. 这是 IWDT 和 LVD 停止运行, POR 为低功耗功能有效的情况。

注 3. 这是 $V_{CC}=3.3V$ 时的值。

表 41.10 DC 特性 (9)

条件: $V_{CC}=AV_{CC0}=1.62 \sim 5.5V$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、 $T_a=-40 \sim +105^\circ C$

项目		符号	min	typ	max	单位	测量条件
模拟电源电流	正在 A/D 转换	$I_{A_{CC}}$	—	1.6	3.0	mA	
	正在 D/A 转换 (每个通道) (注1)		—	0.25	0.50		
	温度传感器		—	60	200	μA	
	A/D、D/A 转换待机时 (全部单元)		—	0.2	0.4		
基准电源电流	正在 A/D 转换	I_{REFH}	—	0.01	0.03	mA	
	A/D、D/A 转换待机时 (全部单元)	I_{REFH0}	—	0.2	0.4		

注 1. D/A 转换器的电源电流包括基准电流。

表 41.11 DC 特性 (10)

条件: $V_{CC}=AV_{CC0}=0 \sim 5.5V$ 、 $V_{REFH}=V_{REFH0}=0 \sim AV_{CC0}$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、 $T_a=-40 \sim +105^\circ C$

项目	符号	min	typ	max	单位
VCC 上升斜率	Sr_{VCC}	0.02	—	20	ms/V

表 41.12 输出容许电流值 (1)

条件: $V_{CC}=AV_{CC0}=1.62 \sim 5.5V$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、总功率 (mW) $< 1000-10 \times T_a$ 时

项目		符号	max	单位
输出 Low 电平的容许电流 (每个引脚的平均值)	正常输出时	I_{OL}	4.0	mA
	高驱动输出时		8.0	
输出 Low 电平的容许电流 (每个引脚的最大值)	正常输出时		4.0	mA
	高驱动输出时		8.0	
输出 Low 电平的容许电流 (总和)	输出引脚的总和	ΣI_{OL}	80	mA
输出 High 电平的容许电流 (每个引脚的平均值)	正常输出时	I_{OH}	-4.0	mA
	高驱动输出时		-8.0	
输出 High 电平的容许电流 (每个引脚的最大值)	正常输出时		-4.0	mA
	高驱动输出时		-8.0	
输出 High 电平的容许电流 (总和)	全部输出引脚的总和	ΣI_{OH}	-80	mA

表 41.13 输出容许电流值 (2)

条件: $V_{CC}=AV_{CC0}=1.62 \sim 5.5V$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、总功率 (mW) $\geq 1000-10 \times T_a$ 时

项目		符号	max	单位
输出 Low 电平的容许电流 (每个引脚的平均值)	正常输出时	I_{OL}	2.0	mA
	高驱动输出时		4.0	
输出 Low 电平的容许电流 (每个引脚的最大值)	正常输出时		2.0	mA
	高驱动输出时		4.0	
输出 Low 电平的容许电流 (总和)	输出引脚的总和	ΣI_{OL}	40	mA
输出 High 电平的容许电流 (每个引脚的平均值)	正常输出时	I_{OH}	-2.0	mA
	高驱动输出时		-4.0	
输出 High 电平的容许电流 (每个引脚的最大值)	正常输出时		-2.0	mA
	高驱动输出时		-4.0	
输出 High 电平的容许电流 (总和)	全部输出引脚的总和	ΣI_{OH}	-40	mA

表 41.14 输出电流值 (1)

条件: $V_{CC}=AV_{CC0}=1.62 \sim 2.7V$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、 $T_a=-40 \sim +105^\circ C$

项目		符号	max	单位	测量条件
输出 Low 电平	全部输出引脚 (RIIC 以外)	普通输出时	0.5	mA	$V_{OL}=0.4V$
		高驱动输出时			
输出 High 电平	全部输出引脚	普通输出时	-0.5	mA	$V_{OH}=V_{CC}-0.4V$
		高驱动输出时			

表 41.15 输出电流值 (2)

条件: $V_{CC}=AV_{CC0}=2.7 \sim 5.5V$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、 $T_a=-40 \sim +105^\circ C$

项目			符号	VCC		单位	测量条件
				2.7 ~ 4.0V	4.0 ~ 5.5V		
				max			
输出 Low 电平	全部输出引脚 (RIIC 以外)	普通输出时	I_{OL}	3.0	4.0	mA	$V_{OL}=1.0V$
		高驱动输出时		5.0	8.0		
	RIIC 引脚	标准模式		3.0	3.0		$V_{OL}=0.4V$
		快速模式		6.0	6.0		$V_{OL}=0.6V$
输出 High 电平	全部输出引脚	普通输出时	I_{OH}	-3.0	-4.0	mA	$V_{OH}=V_{CC}-1.0V$
		高驱动输出时		-5.0	-8.0		

41.3 AC 特性

表 41.16 工作频率（高速运行模式）

条件：VCC=AVCC0=2.7 ~ 5.5V、VSS=AVSS0=VREFL=VREFL0=0V、T_a=-40 ~ +105°C

项目		符号	VCC		单位
			2.7 ~ 5.5V		
最大工作频率	系统时钟（ICLK）	f _{max}	50		MHz
	FlashIF 时钟（FCLK）（注1）		32		
	外围模块时钟（PCLKB）		32		
	外围模块时钟（PCLKD）（注2）		50		
	外部总线时钟（BCLK）		25		
	BCLK 引脚输出		12.5		

注 1. 闪存 P/E 时的 FCLK 的频率下限值为 4MHz。

注 2. 使用 A/D 转换器时的 PCLKD 的频率下限值为 1MHz。

表 41.17 工作频率（中速运行模式 A）

条件：VCC=AVCC0=1.62 ~ 5.5V、VSS=AVSS0=VREFL=VREFL0=0V、T_a=-40 ~ +105°C

项目		符号	VCC			单位
			1.62 ~ 1.8V	1.8 ~ 2.7V	2.7 ~ 5.5V	
最大工作频率	系统时钟（ICLK）	f _{max}	20	32	32	MHz
	FlashIF 时钟（FCLK）（注1）		20	32	32	
	外围模块时钟（PCLKB）		20	32	32	
	外围模块时钟（PCLKD）（注2）		20	32	32	
	外部总线时钟（BCLK）		12	16	25	
	BCLK 引脚输出		6	8	12.5	

注 1. 闪存 P/E 时的 VCC 为 2.7 ~ 5.5V，FCLK 的频率下限值为 4MHz。

注 2. 使用 A/D 转换器时的 PCLKD 的频率下限值为 1MHz。

表 41.18 工作频率（中速运行模式 B）

条件：VCC=AVCC0=1.62 ~ 5.5V、VSS=AVSS0=VREFL=VREFL0=0V、T_a=-40 ~ +105°C

项目		符号	VCC			单位
			1.62 ~ 1.8V	1.8 ~ 2.7V	2.7 ~ 5.5V	
最大工作频率	系统时钟（ICLK）	f _{max}	20	32	32	MHz
	FlashIF 时钟（FCLK）（注1）		20	32	32	
	外围模块时钟（PCLKB）		20	32	32	
	外围模块时钟（PCLKD）（注2）		20	32	32	
	外部总线时钟（BCLK）		12	16	25	
	BCLK 引脚输出		6	8	12.5	

注 1. 闪存 P/E 时的 VCC 为 1.62 ~ 3.6V，FCLK 的频率下限值为 4MHz。

注 2. 使用 A/D 转换器时的 PCLKD 的频率下限值为 1MHz。

表 41.19 工作频率（低速运行模式 1）

条件：VCC=AVCC0=1.62 ~ 5.5V、VSS=AVSS0=VREFL=VREFL0=0V、 $T_a=-40 \sim +105^{\circ}\text{C}$

项目		符号	VCC			单位
			1.62 ~ 1.8V	1.8 ~ 2.7V	2.7 ~ 5.5V	
最大工作频率	系统时钟 (ICLK)	f_{\max}	1	1	1	MHz
	FlashIF 时钟 (FCLK) (注1)		1	1	1	
	外围模块时钟 (PCLKB)		1	1	1	
	外围模块时钟 (PCLKD) (注2)		1	1	1	
	外部总线时钟 (BCLK)		1	1	1	
	BCLK 引脚输出		1	1	1	

注 1. 不能进行闪存的 P/E。

注 2. 使用 A/D 转换器时的 PCLKD 的频率下限值为 1MHz。

表 41.20 工作频率（低速运行模式 2）

条件：VCC=AVCC0=1.62 ~ 5.5V、VSS=AVSS0=VREFL=VREFL0=0V、 $T_a=-40 \sim +105^{\circ}\text{C}$

项目		符号	VCC			单位
			1.62 ~ 1.8V	1.8 ~ 2.7V	2.7 ~ 5.5V	
最大工作频率	系统时钟 (ICLK)	f_{\max}	32.768	32.768	32.768	kHz
	FlashIF 时钟 (FCLK) (注1)		32.768	32.768	32.768	
	外围模块时钟 (PCLKB)		32.768	32.768	32.768	
	外围模块时钟 (PCLKD) (注2)		32.768	32.768	32.768	
	外部总线时钟 (BCLK)		32.768	32.768	32.768	
	BCLK 引脚输出		32.768	32.768	32.768	

注 1. 不支持闪存 P/E。

注 2. 不能使用 A/D 转换器。

41.4 时钟时序

表 41.21 BCLK 时序 (1)

条件: $V_{CC}=AV_{CC0}=2.7 \sim 5.5V$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、
 $f_{BCLK} \leq 25MHz$ (BCLK 引脚输出频率 $\leq 12.5MHz$)、 $T_a=-40 \sim +105^\circ C$

项目	符号	min	typ	max	单位	测量条件
BCLK 引脚输出的周期时间	t_{Bcyc}	80	—	—	ns	图 41.1
BCLK 引脚输出的 High 电平脉宽	t_{CH}	20	—	—	ns	
BCLK 引脚输出的 Low 电平脉宽	t_{CL}	20	—	—	ns	
BCLK 引脚输出的上升时间	t_{Cr}	—	—	15	ns	
BCLK 引脚输出的下降时间	t_{Cf}	—	—	15	ns	

表 41.22 BCLK 时序 (2)

条件: $V_{CC}=AV_{CC0}=1.8 \sim 2.7V$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、
 $f_{BCLK} \leq 16MHz$ (BCLK 引脚输出频率 $\leq 8MHz$)、 $T_a=-40 \sim +105^\circ C$

项目	符号	min	typ	max	单位	测量条件
BCLK 引脚输出的周期时间	t_{Bcyc}	125	—	—	ns	图 41.1
BCLK 引脚输出的 High 电平脉宽	t_{CH}	30	—	—	ns	
BCLK 引脚输出的 Low 电平脉宽	t_{CL}	30	—	—	ns	
BCLK 引脚输出的上升时间	t_{Cr}	—	—	25	ns	
BCLK 引脚输出的下降时间	t_{Cf}	—	—	25	ns	

表 41.23 BCLK 时序 (3)

条件: $V_{CC}=AV_{CC0}=1.62 \sim 1.8V$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、
 $f_{BCLK} \leq 12MHz$ (BCLK 引脚输出频率 $\leq 6MHz$)、 $T_a=-40 \sim +105^\circ C$

项目	符号	min	typ	max	单位	测量条件
BCLK 引脚输出的周期时间	t_{Bcyc}	166.6	—	—	ns	图 41.1
BCLK 引脚输出的 High 电平脉宽	t_{CH}	42	—	—	ns	
BCLK 引脚输出的 Low 电平脉宽	t_{CL}	42	—	—	ns	
BCLK 引脚输出的上升时间	t_{Cr}	—	—	35	ns	
BCLK 引脚输出的下降时间	t_{Cf}	—	—	35	ns	

注. 必须将 BCLK 引脚输出端口的驱动能力置为高驱动输出。

表 41.24 时钟时序

条件: VCC=AVCC0=1.62 ~ 5.5V、VSS=AVSS0=VREFL=VREFL0=0V、T_a=-40 ~ +105°C

项目	符号	min	typ	max	单位	测量条件	
EXTAL 外部时钟输入的周期时间	t _{EXcyc}	50	—	—	ns	图 41.2	
EXTAL 外部时钟输入的 Low 电平脉宽	t _{EXH}	20	—	—	ns		
EXTAL 外部时钟输入的 High 电平脉宽	t _{EXL}	20	—	—	ns		
EXTAL 外部时钟的上升时间	t _{EXr}	—	—	5	ns		
EXTAL 外部时钟的下降时间	t _{EXf}	—	—	5	ns		
EXTAL 外部时钟输入的待机时间 (注1)	t _{EXWT}	1	—	—	ms	图 41.3	
主时钟振荡器振荡频率 (注2)	f _{MAIN}	1	—	20	MHz		
主时钟振荡稳定时间 (晶体) (注2)	t _{MAINOSC}	—	3	—	ms		
主时钟振荡稳定时间 (陶瓷) (注2)	t _{MAINOSC}	—	50	—	μs		
主时钟振荡稳定待机时间 (晶体) (注2)	f _{MAINOSCW}	—	6	—	ms		
主时钟振荡稳定待机时间 (陶瓷) (注2)	t _{MAINOSCW}	—	100	—	μs	图 41.4	
LOCO 时钟周期时间	t _{cyc}	7.27	8	8.89	μs		
LOCO 时钟振荡频率	f _{LOCO}	112.5	125	137.5	kHz		
LOCO 时钟振荡稳定待机时间	t _{LOCOWT}	—	—	20	μs		
HOCO 时钟振荡频率	f _{HOCO}	31.680	32	32.320	MHz		T _a =0 ~ 50°C
		36.495	36.864	37.233			
		39.600	40	40.400			
		49.500	50	50.500			
		31.520	32	32.480		T _a =-40 ~ 105°C	
		36.311	36.864	37.417			
		39.400	40	40.600			
		49.250	50	50.750			
HOCO 时钟振荡稳定待机时间 1	t _{HOCO1}	—	—	300	μs	图 41.5	
HOCO 时钟振荡稳定待机时间 2	t _{HOCO2}	—	—	175	μs	图 41.6	
HOCO 时钟振荡稳定待机时间	t _{HOCOWT}	—	—	350	μs	图 41.6	
HOCO 时钟电源稳定时间	f _{HOCOP}	—	—	350	μs	图 41.7	
PLL 输入频率	f _{PLLIN}	4	—	12.5	MHz	图 41.8	
PLL 电路振荡频率	f _{PLL}	50	—	100	MHz		
PLL 时钟振荡稳定时间	主时钟振荡稳定后开始 PLL 运行	t _{PLL1}	—	—	500	μs	图 41.9
PLL 时钟振荡稳定待机时间		t _{PLLWT1}	1.5	—	—	ms	
PLL 时钟振荡稳定时间 (注4)	主时钟振荡稳定前开始 PLL 运行	t _{PLL2}	—	3.5 (注3)	—	ms	图 41.9
PLL 时钟振荡稳定待机时间 (注4)		t _{PLLWT2}	—	7	—	ms	
副时钟振荡器振荡频率	f _{SUB}	—	32.768	—	kHz	图 41.10	
副时钟振荡稳定时间 (注5)	t _{SUBOSC}	2	—	—	s		

注 1. 这是将 P36、P37 设定为输入，并且将主时钟振荡器停止位 (MOSCCR.MOSTP) 置“0”后到能使用前的时间。

注 2. 给 MOSCWTCR 设定的主时钟振荡稳定时间必须不小于振荡器厂家推荐的稳定时间。主时钟振荡稳定待机时间必须考虑到主时钟振荡稳定时间的容限 (推荐值的 2 倍)。如果通过 MOSTP 位或者 MOFCR.MOFXIN 位更改主时钟振荡器的运行设定，就在经过主时钟振荡稳定待机时间 (t_{MAINOSCW}) 后开始使用主时钟。
这是使用 8MHz 振荡器时的参考值。

注 3. 这是主时钟振荡稳定时间和 PLL 振荡稳定时间相加后的值。

注 4. 这是使用 8MHz 振荡器时的参考值。

注 5. 给 SOSCWTCR 设定的副时钟振荡稳定时间是振荡器厂家推荐的稳定时间减去 2s 后的时间。副时钟振荡稳定待机时间必须考虑到副时钟振荡稳定时间的容限 (推荐值的 2 倍)。如果通过 SOSTP 位或者 RCR3.RTCEN 位更改主时钟振荡器的运行设定，就在经过副时钟振荡稳定待机时间 (t_{SUBOSCW}) 后开始使用副时钟。

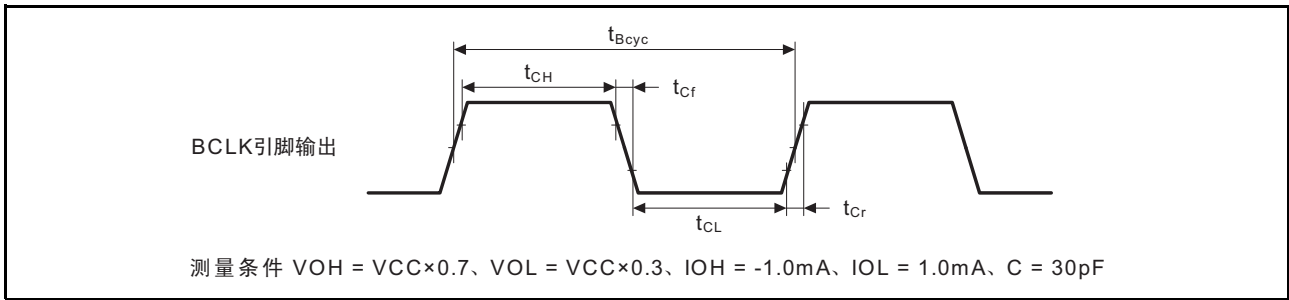


图 41.1 BCLK 引脚输出时序

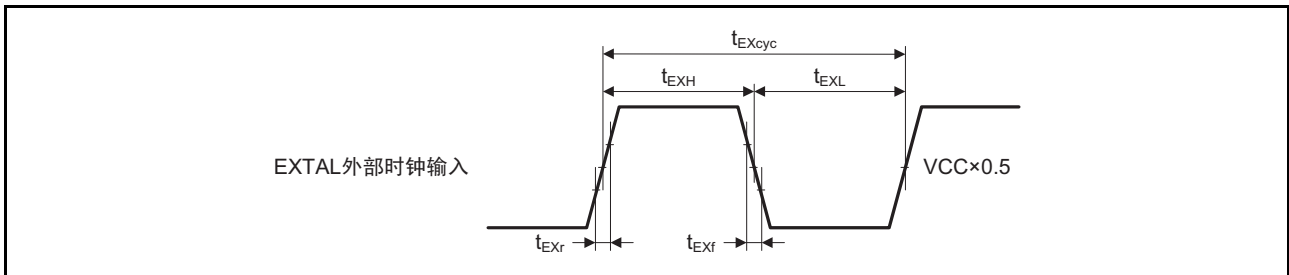


图 41.2 EXTAL 外部时钟输入时序

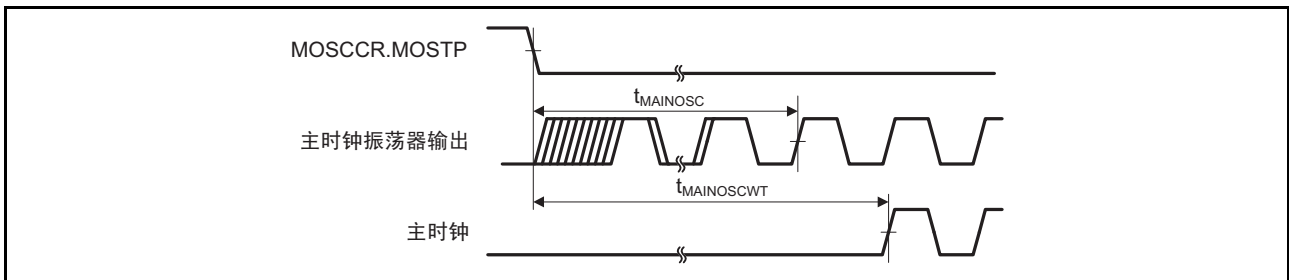


图 41.3 主时钟振荡的开始时序

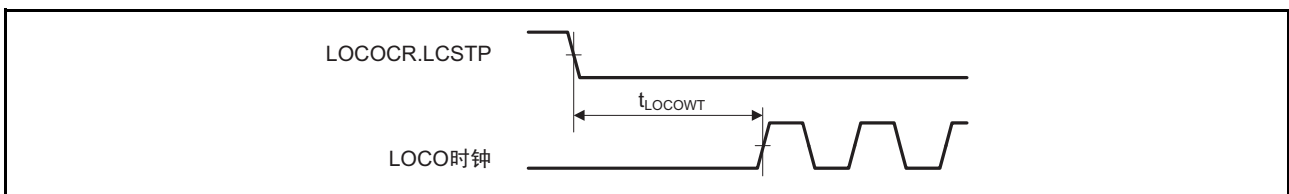


图 41.4 LOCO 时钟振荡的开始时序

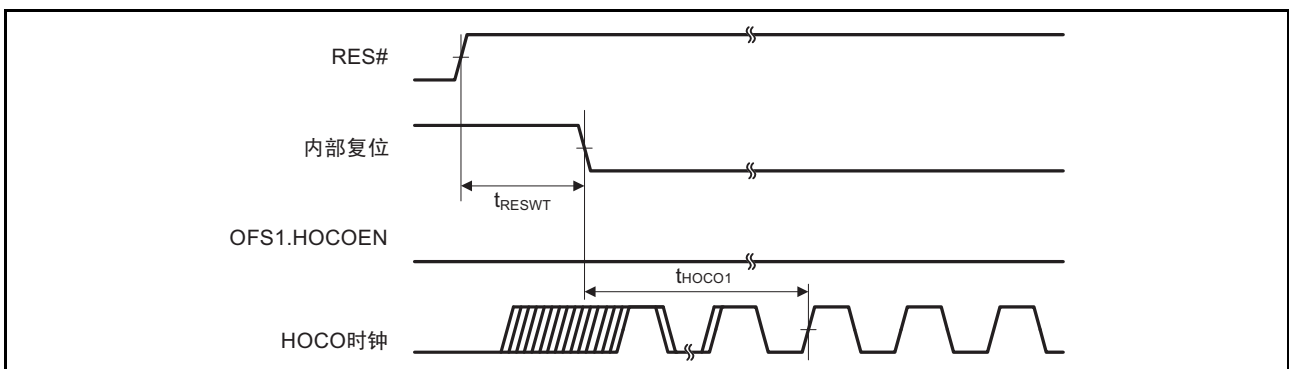


图 41.5 HOCO 时钟振荡的开始时序（在解除 OFS1.HOCOEN 位为“0”时的复位后）

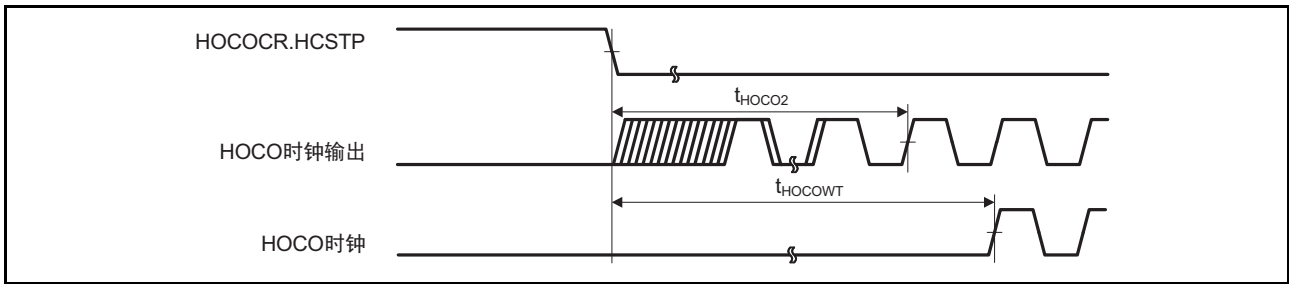


图 41.6 HOCO 时钟振荡的开始时序（通过设定 HOCOCR.HCSTP 位开始振荡）

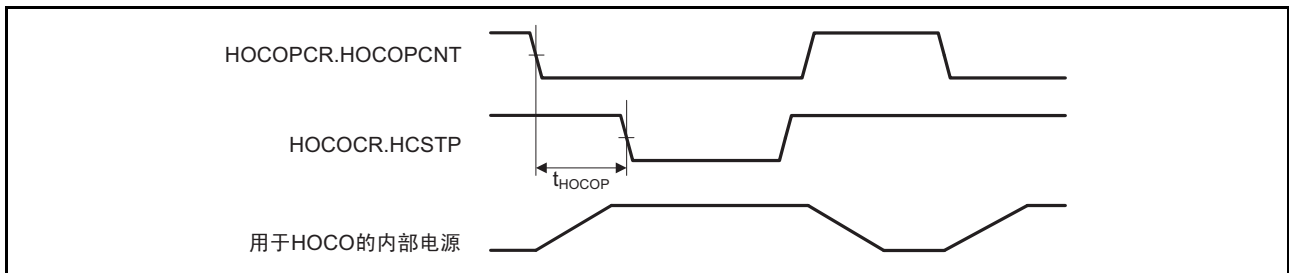


图 41.7 HOCO 电源控制时序

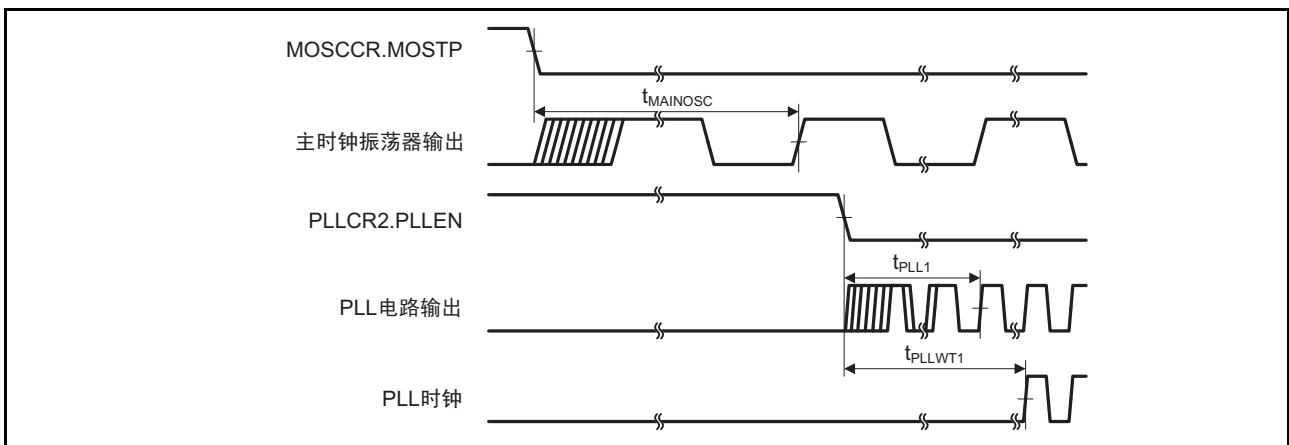


图 41.8 PLL 时钟振荡的开始时序（等到主时钟振荡稳定后开始 PLL 运行的情况）

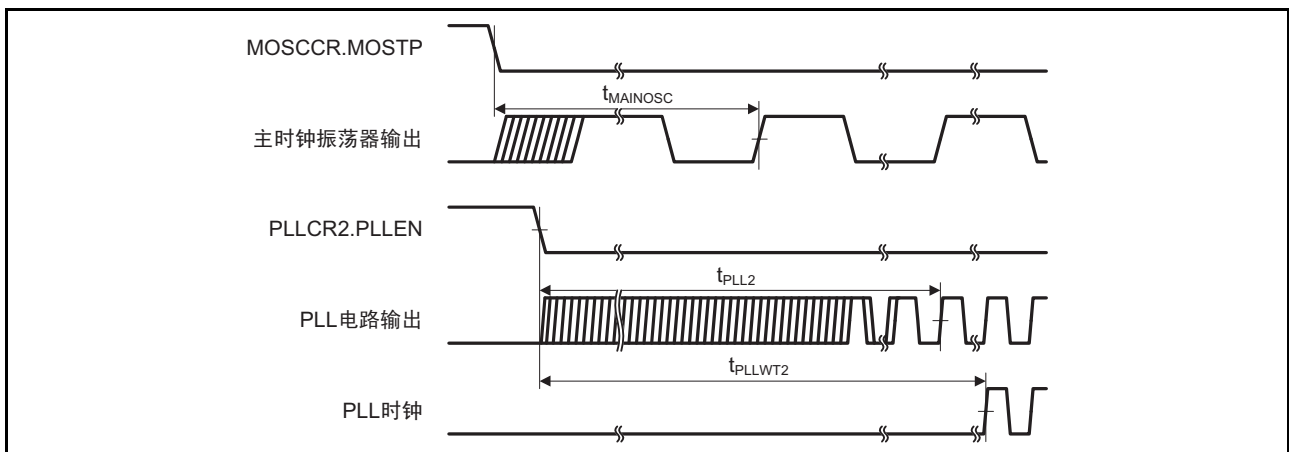


图 41.9 PLL 时钟振荡的开始时序（不等主时钟振荡稳定就开始 PLL 运行的情况）

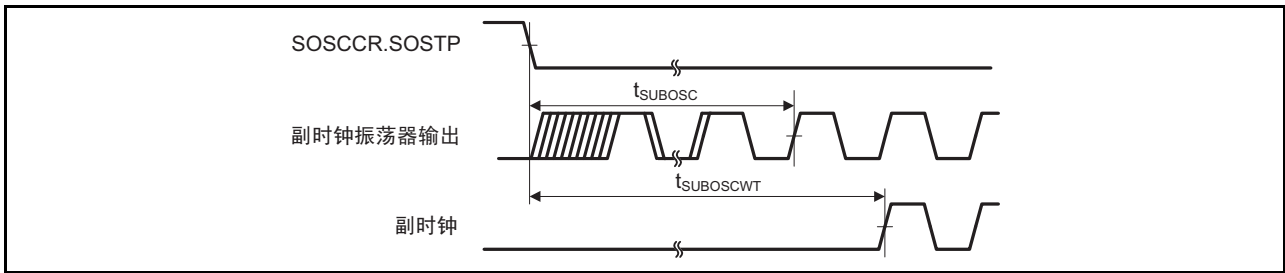


图 41.10 副时钟振荡的开始时序

41.4.1 复位时序

表 41.25 复位时序

条件: VCC=AVCC0=1.62 ~ 5.5V、VSS=AVSS0=VREFL=VREFL0=0V、Ta=-40 ~ +105°C

项目		符号	min	typ	max	单位	测量条件
RES# 脉宽	接通电源时	t _{RESWP}	8	—	—	ms	图 41.11
	深度软件待机模式	t _{RESWD}	8	—	—	ms	图 41.12
	软件待机模式、低速运行模式 1、2	t _{RESWS}	1	—	—	ms	
	上述以外 (除了正在进行 ROM 的编程 / 擦除、E2 数据闪存的编程 / 擦除 / 空白检查)	t _{RESW}	200	—	—	μs	
RES# 解除后的待机时间	t _{RESWT}	—	—	912	μs	图 41.11	
内部复位时间 (独立看门狗定时器复位、看门狗定时器复位、软件复位)	t _{RESW2}	—	—	1.4	ms		

注. 在进行 ROM 的编程 / 擦除、E2 数据闪存的编程 / 擦除和 E2 数据闪存的空白检查过程中，不能通过 RES# 引脚进行复位。详细内容请参照“39. ROM（保存代码的闪存）”的“39.13 使用时的注意事项”。

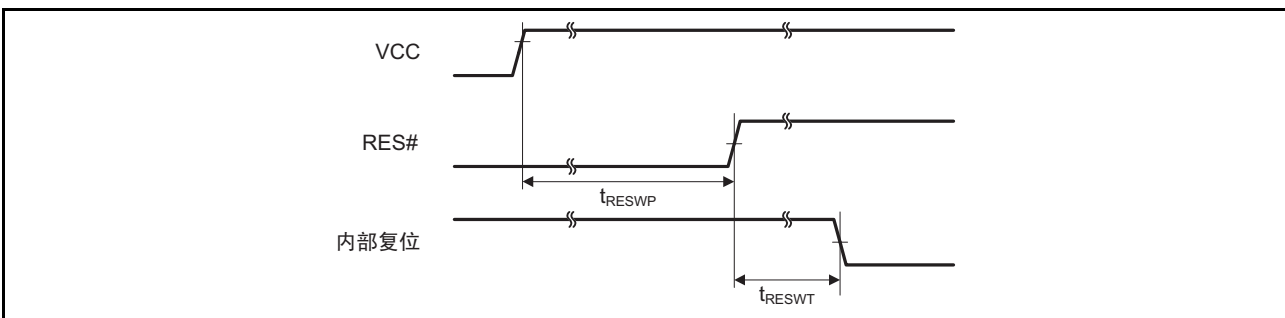


图 41.11 接通电源时的复位输入时序

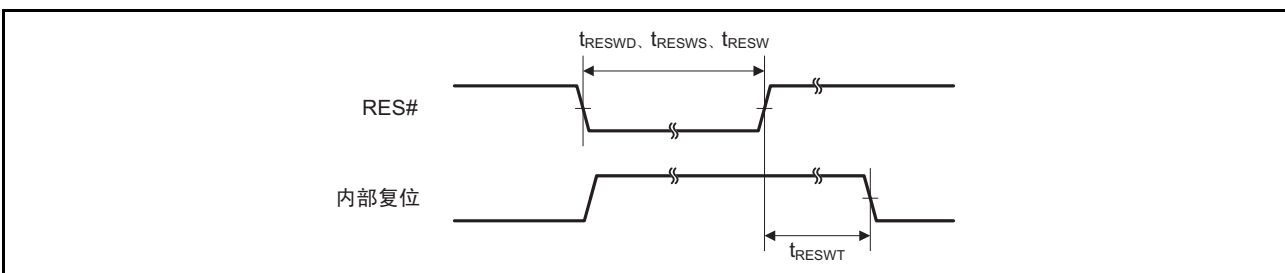


图 41.12 复位输入时序

41.4.2 从低功耗状态恢复的时序

表 41.26 从低功耗状态恢复的时序

条件: $V_{CC}=AV_{CC0}=1.62 \sim 5.5V$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、 $T_a=-40 \sim +105^{\circ}C$

项目		符号	min	typ	max	单位	测量条件	
软件待机模式解除后的恢复时间 (提供闪存、HOCO 电源) (SOFTCUT=000b)	给主时钟振荡器	主时钟振荡器运行	t_{SBYMC}	—	3	—	ms	图 41.13
	连接晶体谐振器 (注 1)	主时钟振荡器和 PLL 电路运行	t_{SBYPC}	—	3.5	—	ms	
	向主时钟振荡器输入外部时钟	主时钟振荡器运行	t_{SBYEX}	10	—	—	μs	
		主时钟振荡器和 PLL 电路运行	t_{SBYPE}	0.5	—	—	ms	
	副时钟振荡器运行		t_{SBYSC}	2	—	—	s	
	HOCO 时钟运行		t_{SBYHO}	—	—	500	μs	
	LOCO 时钟运行		t_{SBYLO}	—	—	90	μs	
软件待机模式解除后的恢复时间 (提供闪存电源、不提供 HOCO 电源) (SOFTCUT=110b)	给主时钟振荡器	主时钟振荡器运行	t_{SBYMC}	—	3	—	ms	图 41.13
	连接晶体谐振器 (注 1)	主时钟振荡器和 PLL 电路运行	t_{SBYPC}	—	3.5	—	ms	
	向主时钟振荡器输入外部时钟	主时钟振荡器运行	t_{SBYEX}	40	—	—	μs	
		主时钟振荡器和 PLL 电路运行	t_{SBYPE}	0.5	—	—	ms	
	副时钟振荡器运行		t_{SBYSC}	2	—	—	s	
	HOCO 时钟运行		t_{SBYHO}	—	—	1.2	ms	
	LOCO 时钟运行		t_{SBYLO}	—	—	90	μs	
软件待机模式解除后的恢复时间 (不提供闪存电源、HOCO 电源) (SOFTCUT=111b)	给主时钟振荡器	主时钟振荡器运行	t_{SBYMC}	—	3	—	ms	图 41.13
	连接晶体谐振器 (注 1)	主时钟振荡器和 PLL 电路运行	t_{SBYPC}	—	3.5	—	ms	
	向主时钟振荡器输入外部时钟	主时钟振荡器运行	t_{SBYEX}	100	—	—	μs	
		主时钟振荡器和 PLL 电路运行	t_{SBYPE}	0.5	—	—	ms	
	副时钟振荡器运行		t_{SBYSC}	2	—	—	s	
	HOCO 时钟运行		t_{SBYHO}	—	—	1.2	ms	
	LOCO 时钟运行		t_{SBYLO}	—	—	10	ms	
深度软件待机模式解除后的恢复时间		t_{DSBY}	—	—	8	ms	图 41.14	
深度软件待机模式解除后的待机时间		t_{DSBYWT}	—	—	0.8	ms		

注 . 等待时间因执行 WAIT 指令时的各振荡器状态而不同。多个振荡器振荡时的恢复时间和正在振荡的振荡器中恢复最费时间的振荡器单独振荡时的恢复时间相同。

注 1. 这是晶体谐振器为 8MHz 时的情况。

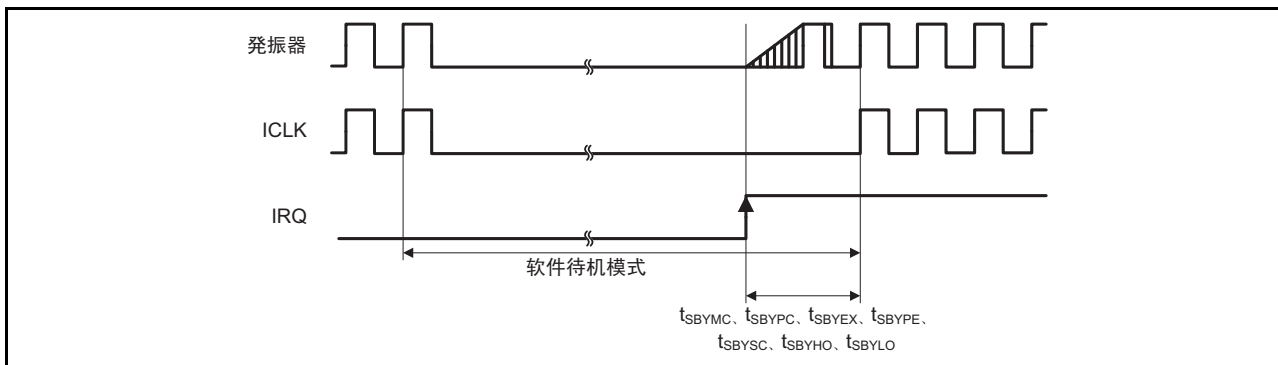


图 41.13 软件待机模式的解除时序

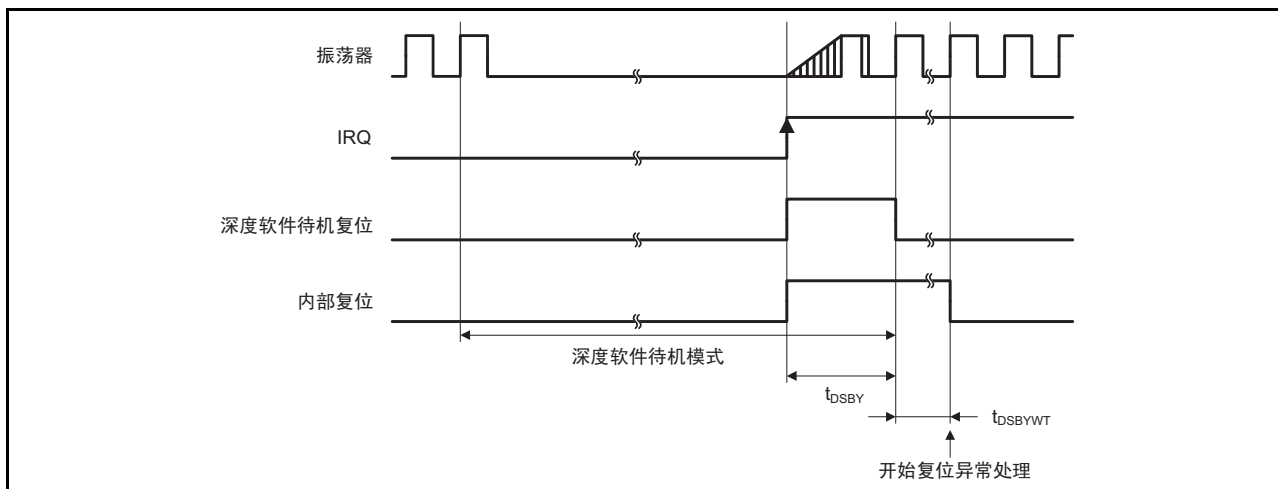


图 41.14 深度软件待机模式的解除时序

41.4.3 控制信号的时序

表 41.27 控制信号的时序

条件: VCC=AVCC0=1.62 ~ 5.5V、VSS=AVSS0=VREFL=VREFL0=0V、Ta=-40 ~ +105°C

项目	符号	min	typ	max	单位	测量条件
NMI 脉宽	t _{NMIW}	200	—	—	ns	t _c (PCLKB)×2 ≤ 200ns、图 41.15
		t _c (PCLKB)×2	—	—	ns	t _c (PCLKB)×2 > 200ns、图 41.15
IRQ 脉宽	t _{IRQW}	200	—	—	ns	t _c (PCLKB)×2 ≤ 200ns、图 41.16
		t _c (PCLKB)×2	—	—	ns	t _c (PCLKB)×2 > 200ns、图 41.16

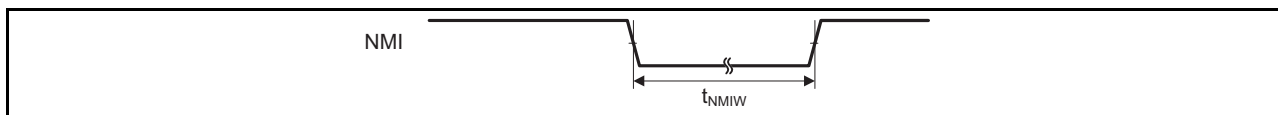


图 41.15 NMI 中断的输入时序

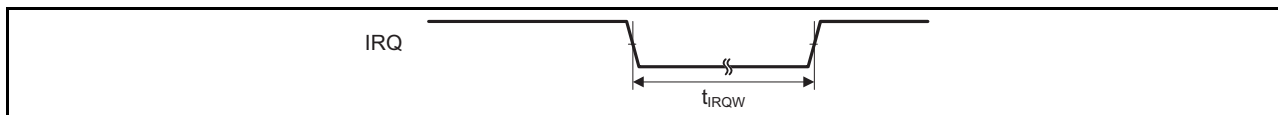


图 41.16 IRQ 中断的输入时序

41.4.4 总线时序

表 41.28 总线时序 (1)

条件: $V_{CC}=AV_{CC0}=2.7 \sim 5.5V$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、(fBCLK 输出 $\leq 12.5MHz$)、
 $T_a=-40 \sim +105^{\circ}C$ 、 $V_{OH}=V_{CC} \times 0.5$ 、 $V_{OL}=V_{CC} \times 0.5$ 、 $I_{OH}=-1.0mA$ 、 $I_{OL}=1.0mA$ 、 $C_L=30pF$

项目	符号	min	max	单位	测量条件
地址的延迟时间	t_{AD}	—	60	ns	图 41.17 ~ 图 41.20
字节控制的延迟时间	t_{BCD}	—	60	ns	
CS# 延迟时间	t_{CSD}	—	60	ns	
RD# 延迟时间	t_{RSD}	—	60	ns	
读数据的准备时间	t_{RDS}	40	—	ns	
读数据的保持时间	t_{RDH}	0	—	ns	
WR# 延迟时间	t_{WRD}	—	60	ns	
写数据的延迟时间	t_{WDD}	—	60	ns	
写数据的保持时间	t_{WDH}	0	—	ns	
WAIT# 准备时间	t_{WTS}	40	—	ns	图 41.21
WAIT# 保持时间	t_{WTH}	0	—	ns	

表 41.29 总线时序 (2)

条件: $V_{CC}=AV_{CC0}=1.8 \sim 2.7V$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、(fBCLK 输出 $\leq 8MHz$)、
 $T_a=-40 \sim +105^{\circ}C$ 、 $V_{OH}=V_{CC} \times 0.5$ 、 $V_{OL}=V_{CC} \times 0.5$ 、 $I_{OH}=-1.0mA$ 、 $I_{OL}=1.0mA$ 、 $C_L=30pF$

项目	符号	min	max	单位	测量条件
地址的延迟时间	t_{AD}	—	90	ns	图 41.17 ~ 图 41.20
字节控制的延迟时间	t_{BCD}	—	90	ns	
CS# 延迟时间	t_{CSD}	—	90	ns	
RD# 延迟时间	t_{RSD}	—	90	ns	
读数据的准备时间	t_{RDS}	60	—	ns	
读数据的保持时间	t_{RDH}	0	—	ns	
WR# 延迟时间	t_{WRD}	—	90	ns	
写数据的延迟时间	t_{WDD}	—	90	ns	
写数据的保持时间	t_{WDH}	0	—	ns	
WAIT# 准备时间	t_{WTS}	60	—	ns	图 41.21
WAIT# 保持时间	t_{WTH}	0	—	ns	

表 41.30 总线时序 (3)

条件: $V_{CC}=AV_{CC0}=1.62 \sim 1.8V$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、(fBCLK 输出 $\leq 6MHz$)、
 $T_a=-40 \sim +105^{\circ}C$ 、 $V_{OH}=V_{CC} \times 0.5$ 、 $V_{OL}=V_{CC} \times 0.5$ 、 $I_{OH}=-0.5mA$ 、 $I_{OL}=0.5mA$ 、 $C_L=30pF$

项目	符号	min	max	单位	测量条件
地址的延迟时间	t_{AD}	—	125	ns	图 41.17 ~ 图 41.20
字节控制的延迟时间	t_{BCD}	—	125	ns	
CS# 延迟时间	t_{CSD}	—	125	ns	
RD# 延迟时间	t_{RSD}	—	125	ns	
读数据的准备时间	t_{RDS}	85	—	ns	
读数据的保持时间	t_{RDH}	0	—	ns	
WR# 延迟时间	t_{WRD}	—	125	ns	
写数据的延迟时间	t_{WDD}	—	125	ns	
写数据的保持时间	t_{WDH}	0	—	ns	
WAIT# 准备时间	t_{WTS}	85	—	ns	图 41.21
WAIT# 保持时间	t_{WTH}	0	—	ns	

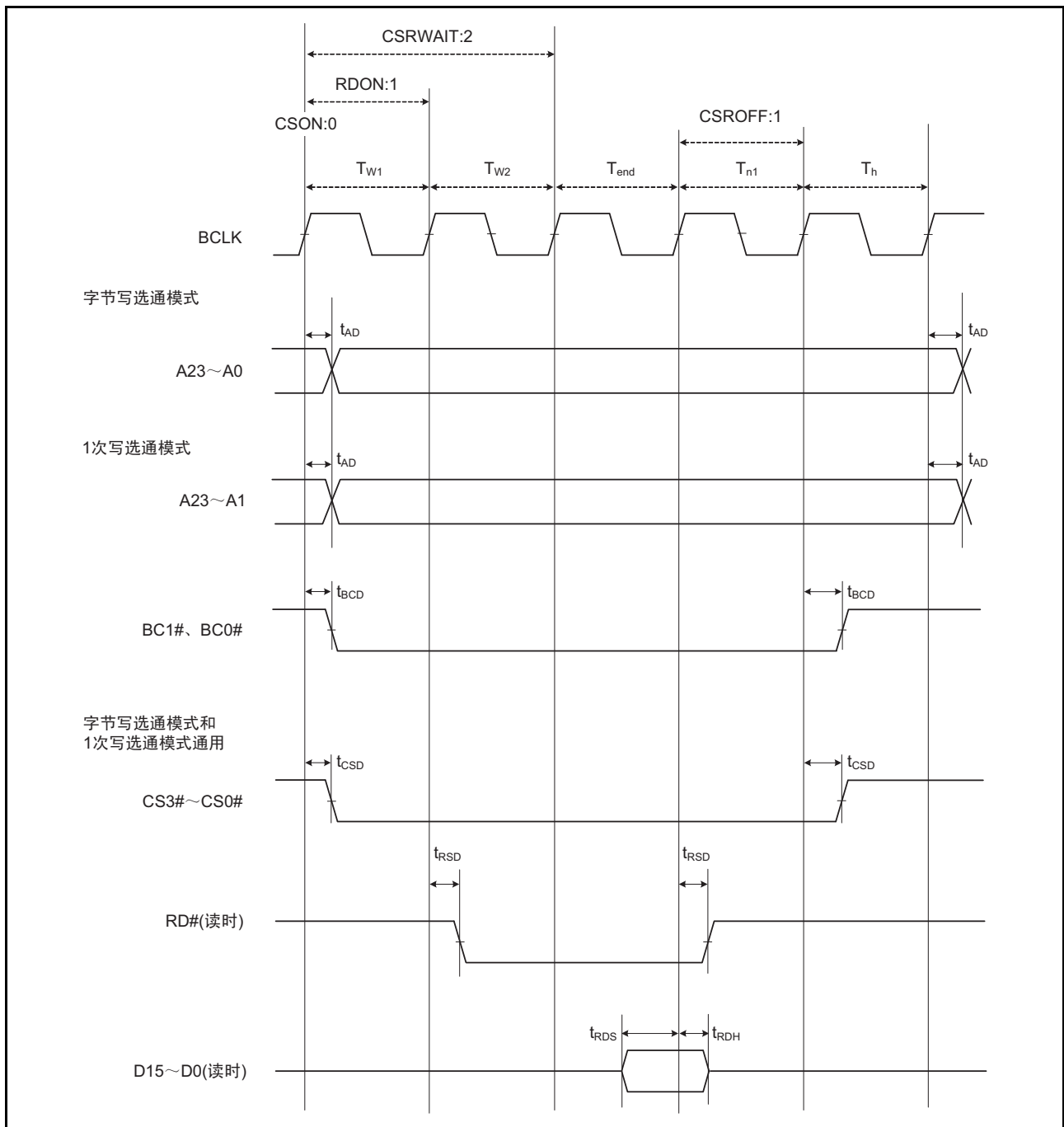


图 41.17 外部总线时序 / 正常读周期 (总线时钟同步)

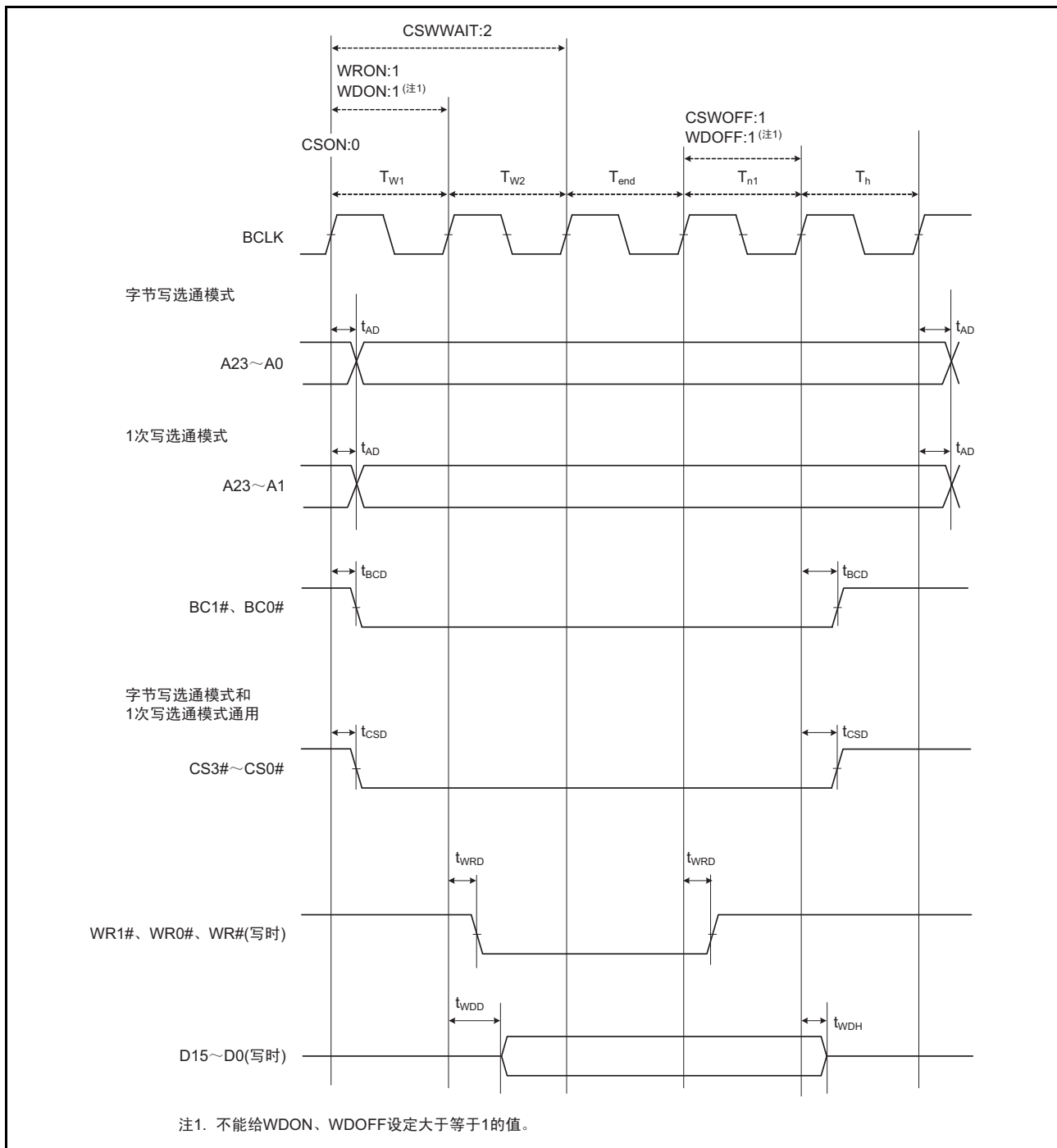


图 41.18 外部总线时序 / 正常写周期 (总线时钟同步)

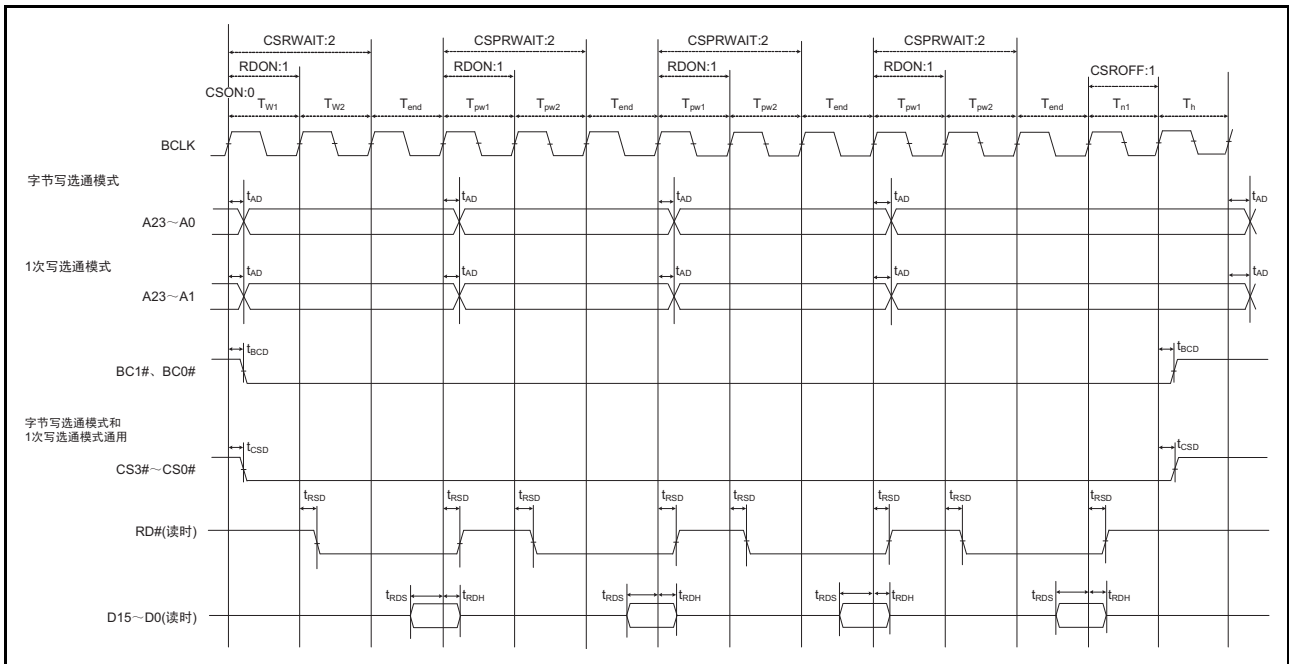


图 41.19 外部总线时序 / 页面读周期 (总线时钟同步)

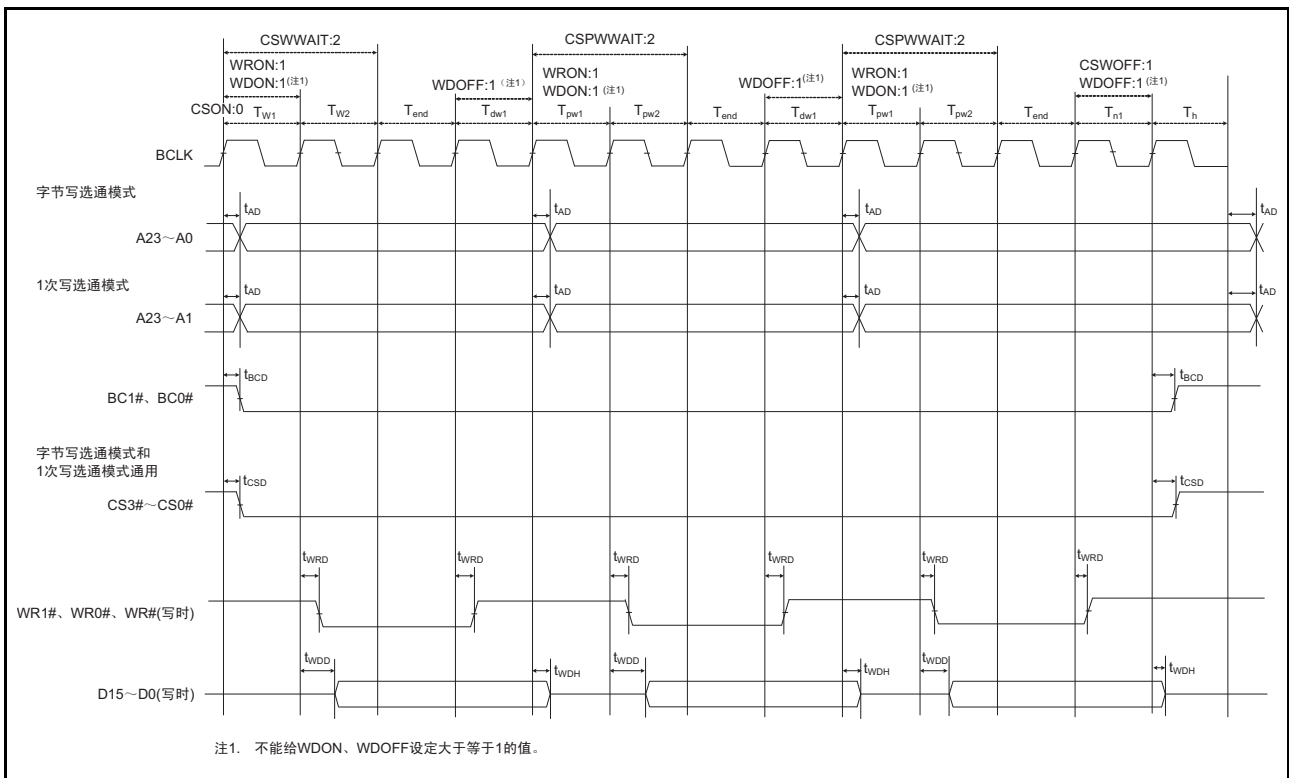


图 41.20 外部总线时序 / 页面写周期 (总线时钟同步)

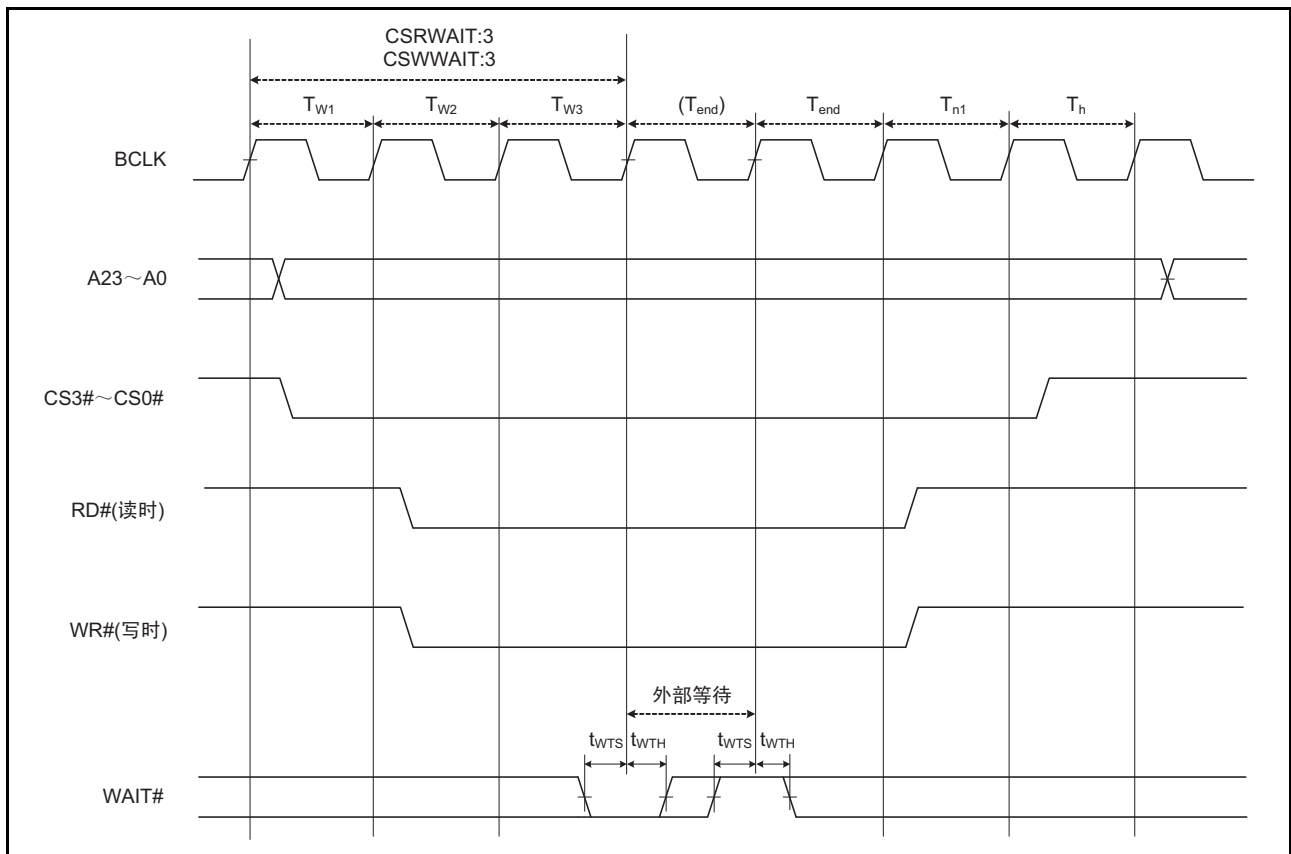


图 41.21 外部总线时序 / 外部等待控制

表 41.31 总线时序（多路复用总线）（1）

条件: $V_{CC}=AV_{CC0}=2.7 \sim 5.5V$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、(fBCLK 输出 $\leq 12.5MHz$)、 $T_a=-40 \sim +105^\circ C$ 、 $V_{OH}=V_{CC} \times 0.5$ 、 $V_{OL}=V_{CC} \times 0.5$ 、 $I_{OH}=-1.0mA$ 、 $I_{OL}=1.0mA$ 、 $C_L=30pF$

项目	符号	min	max	单位	测量条件
地址的延迟时间	t_{AD}	—	60	ns	图 40.17 ~ 图 40.20
字节控制的延迟时间	t_{BCD}	—	60	ns	
CS# 延迟时间	t_{CSD}	—	60	ns	
RD# 延迟时间	t_{RSD}	—	60	ns	
读数据的准备时间	t_{RDS}	40	—	ns	
读数据的保持时间	t_{RDH}	0	—	ns	
WR# 延迟时间	t_{WRD}	—	60	ns	
写数据的延迟时间	t_{WDD}	—	60	ns	
写数据的保持时间	t_{WDH}	0	—	ns	
WAIT# 准备时间	t_{WTS}	40	—	ns	图 40.21
WAIT# 保持时间	t_{WTH}	0	—	ns	
ALE 输出的延迟时间 (BCLK 基准)	$t_{d(BCLK-ALE)}$ 、 t_{ALEd}	—	60	ns	图 41.22、 图 41.23
ALE 输出的保持时间 (BCLK 基准)	$t_{h(BCLK-ALE)}$ 、 t_{ALEh}	60	—	ns	

表 41.32 总线时序（多路复用总线）（2）

条件：VCC=AVCC0=1.8 ~ 2.7V、VSS=AVSS0=VREFL=VREFL0=0V、(fBCLK 输出 ≤ 8MHz)、
 $T_a = -40 \sim +105^\circ\text{C}$ 、 $V_{OH} = VCC \times 0.5$ 、 $V_{OL} = VCC \times 0.5$ 、 $I_{OH} = -1.0\text{mA}$ 、 $I_{OL} = 1.0\text{mA}$ 、 $C_L = 30\text{pF}$

项目	符号	min	max	单位	测量条件
地址的延迟时间	t_{AD}	—	90	ns	图 41.17 ~ 图 41.20
字节控制的延迟时间	t_{BCD}	—	90	ns	
CS# 延迟时间	t_{CSD}	—	90	ns	
RD# 延迟时间	t_{RSD}	—	90	ns	
读数据的准备时间	t_{RDS}	60	—	ns	
读数据的保持时间	t_{RDH}	0	—	ns	
WR# 延迟时间	t_{WRD}	—	90	ns	
写数据的延迟时间	t_{WDD}	—	90	ns	
写数据的保持时间	t_{WDH}	0	—	ns	
WAIT# 准备时间	t_{WTS}	60	—	ns	
WAIT# 保持时间	t_{WTH}	0	—	ns	
ALE 输出的延迟时间（BCLK 基准）	$t_{d(BCLK-ALE)}$ 、 t_{ALEd}	—	90	ns	图 41.22、 图 41.23
ALE 输出的保持时间（BCLK 基准）	$t_{h(BCLK-ALE)}$ 、 t_{ALEd}	90	—	ns	

表 41.33 总线时序（多路复用总线）（3）

条件：VCC=AVCC0=1.62 ~ 1.8V、VSS=AVSS0=VREFL=VREFL0=0V、(fBCLK 输出 ≤ 6MHz)、
 $T_a = -40 \sim +105^\circ\text{C}$ 、 $V_{OH} = VCC \times 0.5$ 、 $V_{OL} = VCC \times 0.5$ 、 $I_{OH} = -0.5\text{mA}$ 、 $I_{OL} = 0.5\text{mA}$ 、 $C_L = 30\text{pF}$

项目	符号	min	max	单位	测量条件
地址的延迟时间	t_{AD}	—	125	ns	图 41.17 ~ 图 41.20
字节控制的延迟时间	t_{BCD}	—	125	ns	
CS# 延迟时间	t_{CSD}	—	125	ns	
RD# 延迟时间	t_{RSD}	—	125	ns	
读数据的准备时间	t_{RDS}	85	—	ns	
读数据的保持时间	t_{RDH}	0	—	ns	
WR# 延迟时间	t_{WRD}	—	125	ns	
写数据的延迟时间	t_{WDD}	—	125	ns	
写数据的保持时间	t_{WDH}	0	—	ns	
WAIT# 准备时间	t_{WTS}	85	—	ns	
WAIT# 保持时间	t_{WTH}	0	—	ns	
ALE 输出的延迟时间（BCLK 基准）	$t_{d(BCLK-ALE)}$ 、 t_{ALEd}	—	125	ns	图 41.22、 图 41.23
ALE 输出的保持时间（BCLK 基准）	$t_{h(BCLK-ALE)}$ 、 t_{ALEd}	125	—	ns	

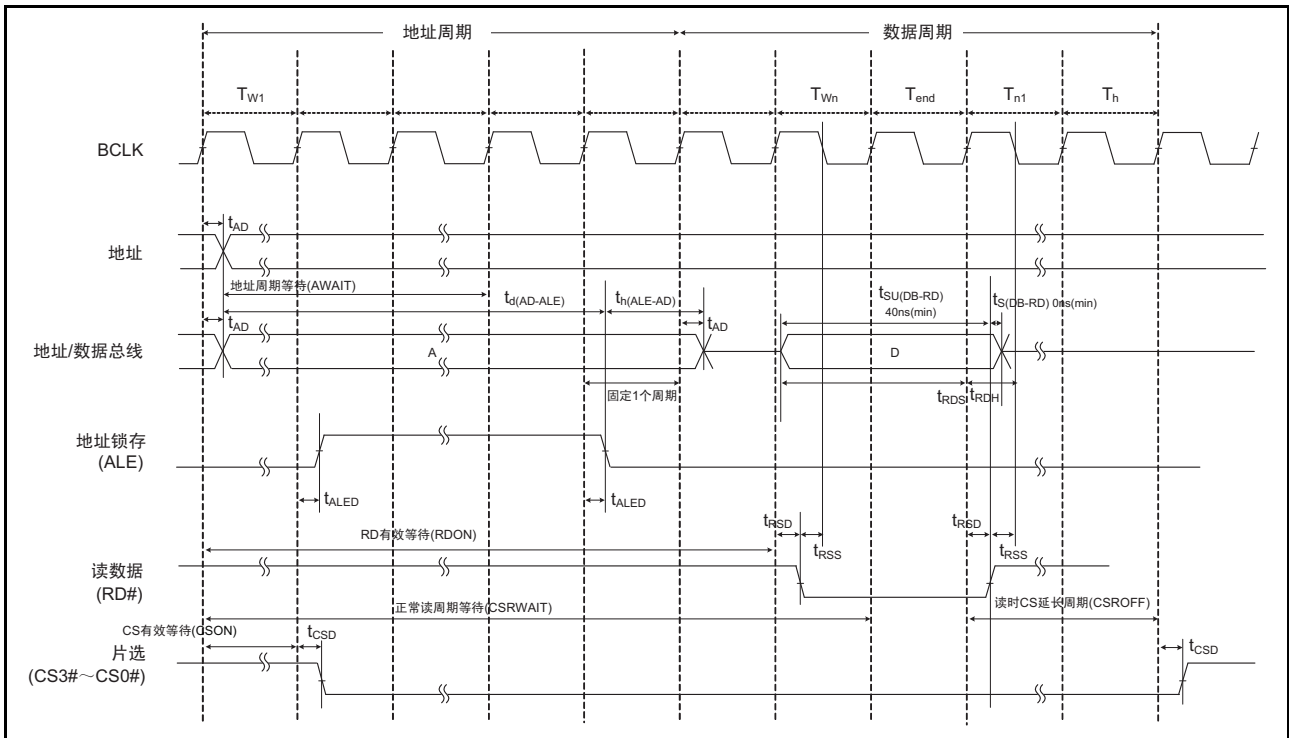


图 41.22 外部总线时序 / 读存取的运行例子 (多路复用)

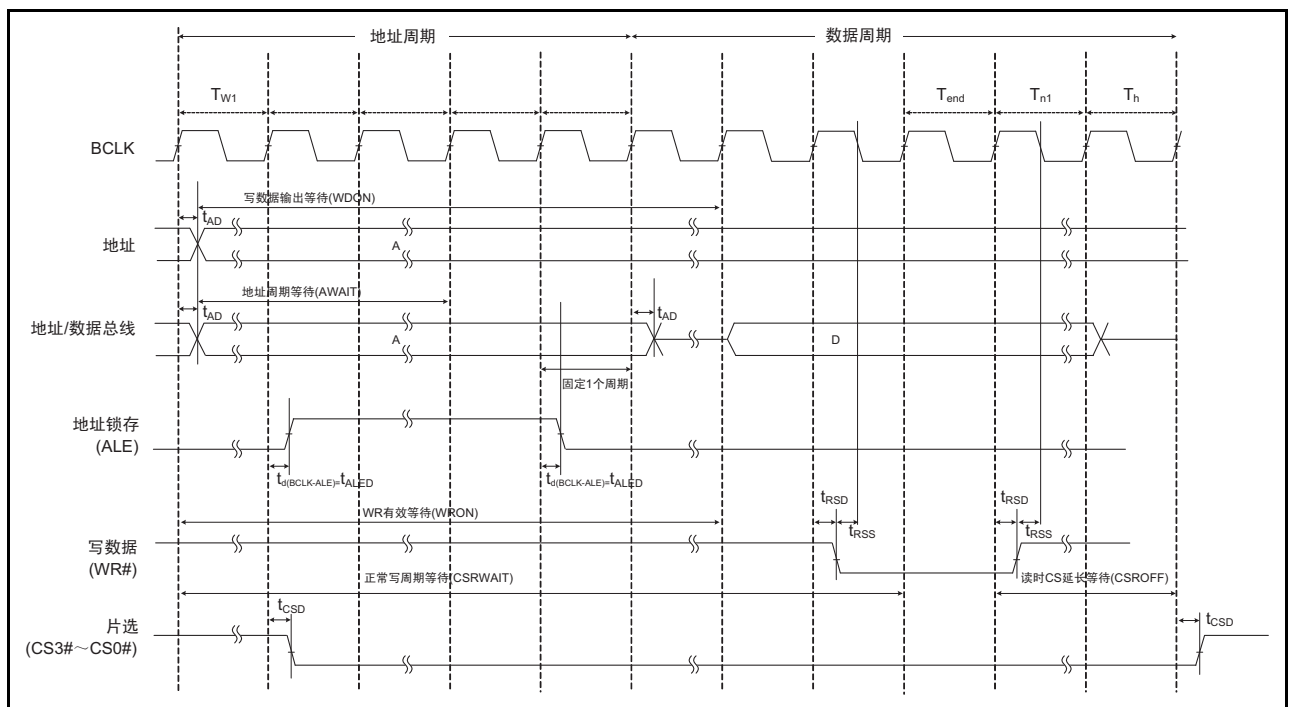


图 41.23 外部总线时序 / 写存取的运行例子 (多路复用)

41.4.5 内部外围模块的时序

表 41.34 内部外围模块的时序 (1)

条件: VCC=AVCC0=1.62 ~ 5.5V、VSS=AVSS0=VREFL=VREFL0=0V、T_a=-40 ~ +105°C

项目		符号	min	max	单位 (注1)	测量条件	
I/O 端口	输入数据脉宽	t _{PRW}	1.5	—	t _{Pcyc}	图 41.24	
MTU	输入捕捉的输入脉宽	指定单边沿	t _{TICW}	1.5	—	t _{Pcyc}	图 41.25
		指定双边沿		2.5	—		
	定时器时钟脉宽	指定单边沿	t _{TCKWH} 、	1.5	—	t _{Pcyc}	图 41.26
		指定双边沿	t _{TCKWL}	2.5	—		
相位系数模式			2.5	—			
POE	POE# 输入脉宽	t _{POEW}	1.5	—	t _{Pcyc}	图 41.27	
8 位定时器	定时器时钟脉宽	指定单边沿	t _{TMCWH} 、	1.5	—	t _{Pcyc}	图 41.28
		指定双边沿	t _{TMCWL}	2.5	—		
SCI	输入时钟周期	异步	t _{Scyc}	4	—	t _{Pcyc}	图 41.29
		时钟同步		6	—		
	输入时钟脉宽		t _{SCKW}	0.4	0.6	t _{Scyc}	
	输入时钟的上升时间		t _{SCKr}	—	20	ns	
	输入时钟的下降时间		t _{SCKf}	—	20	ns	
	输出时钟周期 (注2)	异步	t _{Scyc}	16	—	t _{Pcyc}	
		时钟同步		4	—		
	输出时钟脉宽 (注2)		t _{SCKW}	0.4	0.6	t _{Scyc}	
	输出时钟的上升时间 (注2)		t _{SCKr}	—	20	ns	
	输出时钟的下降时间 (注2)		t _{SCKf}	—	20	ns	
	发送数据的延迟时间 (注3)	时钟同步	t _{TXD}	—	40	ns	
	接收数据的准备时间	时钟同步	t _{RXS}	40	—	ns	
接收数据的保持时间	时钟同步	t _{RXH}	40	—	ns		
A/D 转换器	触发输入的脉宽	t _{TRGW}	1.5	—	t _{Pcyc}	图 41.31	

注 1. t_{Pcyc}: PCLK 的周期

注 2. 必须将时钟输出端口的驱动能力设定为高驱动输出。

注 3. 必须将数据输出端口的驱动能力设定为高驱动输出。

表 41.35 内部外围模块的时序 (2)

条件: VCC=AVCC0=1.62 ~ 5.5V、VSS=AVSS0=VREFL=VREFL0=0V、T_a=-40 ~ +105°C

项目		符号	min	max	单位 (注 1)	测量条件		
RSPI	RSPCK 时钟周期 (注 2)	主控	t _{SPCyc}	4	4096	t _{Pcyc}	图 41.32	
		从属		8	4096			
	RSPCK 时钟的 High 电平脉宽 (注 2)	主控	t _{SPCKWH}	$(t_{SPCyc} - t_{SPCKR} - t_{SPCKF})/2 - 3$	—	ns		图 41.33 ~ 图 41.36
		从属			$(t_{SPCyc} - t_{SPCKR} - t_{SPCKF})/2$			
	RSPCK 时钟的 Low 电平脉宽 (注 2)	主控	t _{SPCKWL}	$(t_{SPCyc} - t_{SPCKR} - t_{SPCKF})/2 - 3$	—	ns		
		从属			$(t_{SPCyc} - t_{SPCKR} - t_{SPCKF})/2$			
	RSPCK 时钟的 上升 / 下降时间 (注 2)	输出	t _{SPCKr}	—	10	ns		
		输入	t _{SPCKf}	—	1	μs		
	数据输入的准备时间	主控	t _{SU}	VCC ≥ 3.0V	15	—	ns	
				VCC < 3.0V	20	—		
	数据输入的保持时间	从属		2 × t _{Pcyc} - 20	—			
		主控	t _H	0	—	ns		
	SSL 准备时间	从属		20 + 2 × t _{Pcyc}	—			
		主控	t _{LEAD}	1	8	t _{SPcyc}		
	SSL 保持时间	从属		4	—	t _{Pcyc}		
		主控	t _{LAG}	1	8	t _{SPcyc}		
数据输出的延迟时间	从属		4	—	t _{Pcyc}			
	主控	t _{OD}	—	86	ns			
数据输出的保持时间	从属		—	3 × t _{Pcyc} + 40				
	主控	t _{OH}	0	—	ns			
连续发送的延迟时间	从属		0	—				
	主控	t _{TD}	t _{SPcyc} + 2 × t _{Pcyc}	8 × t _{SPcyc} + 2 × t _{Pcyc}	ns			
MOSI、MISO 的 上升 / 下降时间	从属		4 × t _{Pcyc}	—				
	输出	t _{Dr} 、t _{Df}	—	20	ns			
输入	—		1	μs				
SSL 的上升 / 下降时间	输出	t _{SSLr}	—	20	ns			
	输入	t _{SSLf}	—	1	μs			
从属存取时间		t _{SA}	—	4	t _{Pcyc}	图 41.35、 图 41.36		
从属输出释放时间		t _{REL}	—	3	t _{Pcyc}			

注 1. t_{Pcyc}: PCLK 的周期

注 2. 必须将时钟输出端口的驱动能力设定为高驱动输出。

表 41.36 内部外围模块的时序 (3)

条件: $VCC=AVCC0=1.62 \sim 5.5V$ 、 $VSS=AVSS0=VREFL=VREFL0=0V$ 、 $T_a=-40 \sim +105^{\circ}C$

项目		符号	min	max	单位 (注1)	测量条件	
简易 SPI	SCK 时钟周期输出 (主控) (注2)	t_{SPcyc}	4	65536	t_{Pcyc}	图 41.32	
	SCK 时钟周期输入 (从属)		8	65536			
	SCK 时钟的 High 电平脉宽 (注2)	t_{SPCKWH}	0.4	0.6	t_{SPcyc}		
	SCK 时钟的 Low 电平脉宽 (注2)	t_{SPCKWL}	0.4	0.6	t_{SPcyc}		
	SCK 时钟的上升 / 下降时间 (注2)	t_{SPCKr} 、 t_{SPCKf}	—	20	ns		
	数据输入的准备时间	t_{SU}	40	—	ns	图 41.33、 图 41.34	
	数据输入的保持时间	t_H	40	—	ns		
	SS 输入的准备时间	t_{LEAD}	1	—	t_{SPcyc}		
	SS 输入的保持时间	t_{LAG}	1	—	t_{SPcyc}		
	数据输出的延迟时间	t_{OD}	—	40	ns		
	数据输出的保持时间	t_{OH}	0	—	ns		
	数据的上升 / 下降时间	t_{Dr} 、 t_{Df}	—	20	ns		
	SS 输入的上升 / 下降时间	t_{SSLr} 、 t_{SSLf}	—	20	ns		
	从属存取时间	t_{SA}	—	5	t_{Pcyc}		图 41.35
	从属输出释放时间	t_{REL}	—	5	t_{Pcyc}		图 41.36

注 1. t_{Pcyc} : PCLK 的周期

注 2. 必须将时钟输出端口的驱动能力设定为高驱动输出。

表 41.37 内部外围模块的时序 (4)

条件: $VCC=AVCC0=2.7 \sim 5.5V$ 、 $VSS=AVSS0=VREFL=VREFL0=0V$ 、 $fPCLKB \leq 32MHz$ 、 $T_a=-40 \sim +105^\circ C$

项目		符号	min (注1、注2)	max	单位	测量条件
RIIC (标准模式、SMBus)	SCL 输入的周期时间	t_{SCL}	$8(10) \times (1/PCLK) + 1300$	—	ns	图 41.37
	SCL 输入的 High 电平脉宽	t_{SCLH}	$3(5) \times (1/PCLK) + 300$	—	ns	
	SCL 输入的 Low 电平脉宽	t_{SCLL}	$5 \times (1/PCLK) + 1000$	—	ns	
	SCL 输入和 SDA 输入的上升时间	t_{Sr}	—	1000	ns	
	SCL 输入和 SDA 输入的下降时间	t_{Sf}	—	300	ns	
	SCL 输入和 SDA 输入尖峰脉冲的消除时间	t_{SP}	0	$4 \times (1/PCLK)$	ns	
	SDA 输入的总线空闲时间	t_{BUF}	$5 \times (1/PCLK) + 1000$	—	ns	
	开始条件输入的保持时间	t_{STAH}	$3(5) \times (1/PCLK) + 300$	—	ns	
	重新开始条件输入的准备时间	t_{STAS}	$5 \times (1/PCLK) + 1000$	—	ns	
	停止条件输入的准备时间	t_{STOS}	$3(5) \times (1/PCLK) + 300$	—	ns	
	数据输入的准备时间	t_{SDAS}	250	—	ns	
	数据输入的保持时间	t_{SDAH}	0	—	ns	
	SCL 和 SDA 的电容性负载	C_b	—	400	pF	
RIIC (快速模式)	SCL 输入的周期时间	t_{SCL}	$8(10) \times (1/PCLK) + 600$	—	ns	图 41.37
	SCL 输入的 High 电平脉宽	t_{SCLH}	$3(5) \times (1/PCLK) + 300$	—	ns	
	SCL 输入的 Low 电平脉宽	t_{SCLL}	$5 \times (1/PCLK) + 300$	—	ns	
	SCL 输入和 SDA 输入的上升时间	t_{Sr}	$20 + 0.1C_b$	300	ns	
	SCL 输入和 SDA 输入的下降时间	t_{Sf}	$20 + 0.1C_b$	300	ns	
	SCL 输入和 SDA 输入尖峰脉冲的消除时间	t_{SP}	0	$4 \times (1/PCLK)$	ns	
	SDA 输入的总线空闲时间	t_{BUF}	$5 \times (1/PCLK) + 300$	—	ns	
	开始条件输入的保持时间	t_{STAH}	$3(5) \times (1/PCLK) + 300$	—	ns	
	重新开始条件输入的准备时间	t_{STAS}	$5 \times (1/PCLK) + 300$	—	ns	
	停止条件输入的准备时间	t_{STOS}	$3(5) \times (1/PCLK) + 300$	—	ns	
	数据输入的准备时间	t_{SDAS}	100	—	ns	
	数据输入的保持时间	t_{SDAH}	0	—	ns	
	SCL 和 SDA 的电容性负载	C_b	—	400	pF	

注 1. () 内的数值表示在 ICFER.NFE 为“1”并且将数字滤波器置为有效的状态下 ICMR3.NF[1:0] 为“11b”的情况。

注 2. C_b 是总线的电容总和。

表 41.38 内部外围模块的时序 (5)

条件: $VCC=AVCC0=2.7 \sim 5.5V$ 、 $VSS=AVSS0=VREFL=VREFL0=0V$ 、 $fPCLKB \leq 32MHz$ 、 $T_a=-40 \sim +105^\circ C$

项目		符号	min (注1)	max	单位	测量条件
简易 IIC (标准模式)	SDA 输入的上升时间	t_{Sr}	—	1000	ns	图 41.37
	SDA 输入的下降时间	t_{Sf}	—	300	ns	
	SDA 输入尖峰脉冲的消除时间	t_{SP}	0	$4 \times (1/PCLK)$	ns	
	数据输入的准备时间	t_{SDAS}	250	—	ns	
	数据输入的保持时间	t_{SDAH}	0	—	ns	
	SCL 和 SDA 的电容性负载	C_b	—	400	pF	
简易 IIC (快速模式)	SCL 输入和 SDA 输入的上升时间	t_{Sr}	$20 + 0.1C_b$	300	ns	图 41.37
	SCL 输入和 SDA 输入的下降时间	t_{Sf}	$20 + 0.1C_b$	300	ns	
	SCL 输入和 SDA 输入尖峰脉冲的消除时间	t_{SP}	0	$4 \times (1/PCLK)$	ns	
	数据输入的准备时间	t_{SDAS}	100	—	ns	
	数据输入的保持时间	t_{SDAH}	0	—	ns	
	SCL 和 SDA 的电容性负载	C_b	—	400	pF	

注 1. C_b 是总线的电容总和。

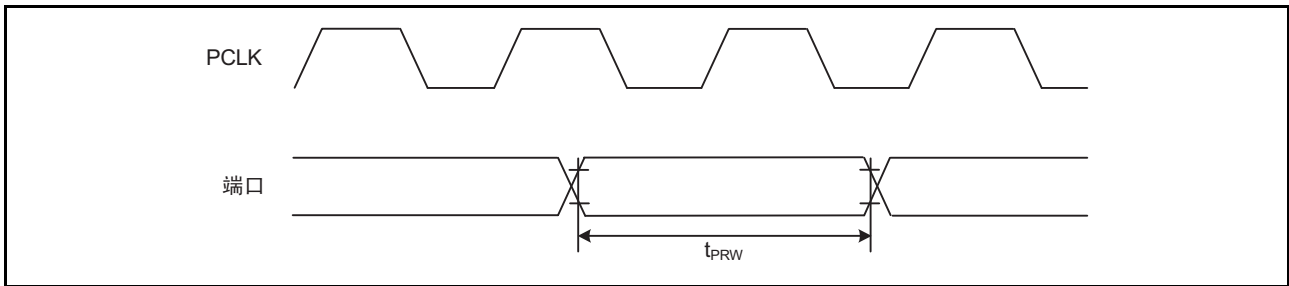


图 41.24 I/O 端口输入时序

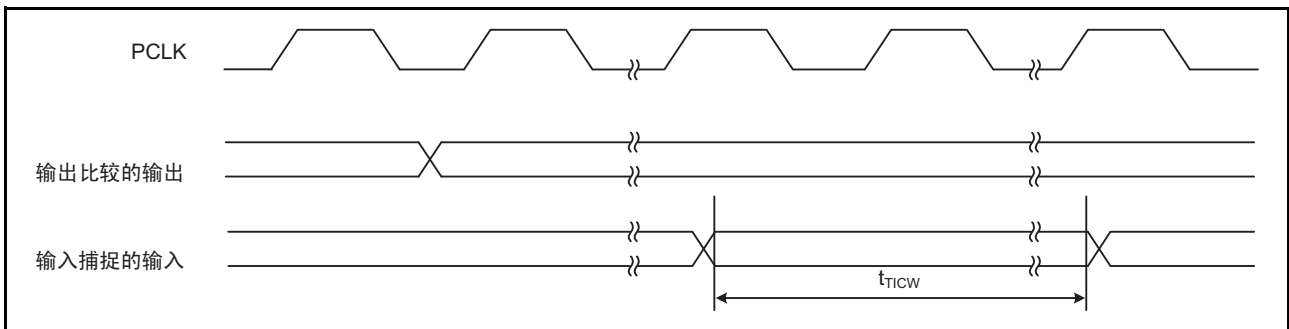


图 41.25 MTU 输入 / 输出时序

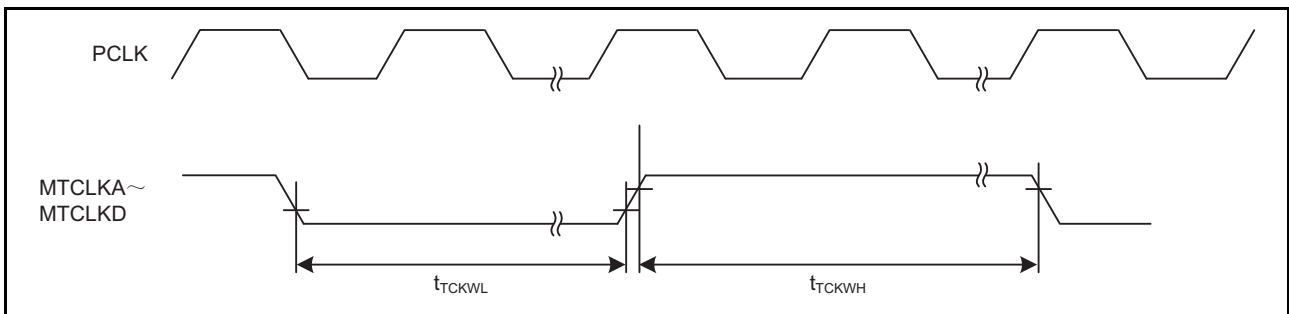


图 41.26 MTU 时钟输入时序

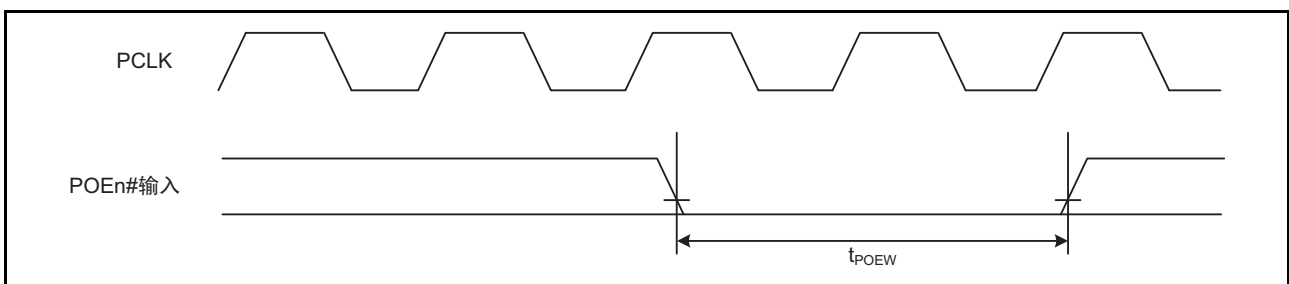


图 41.27 POE# 输入时序

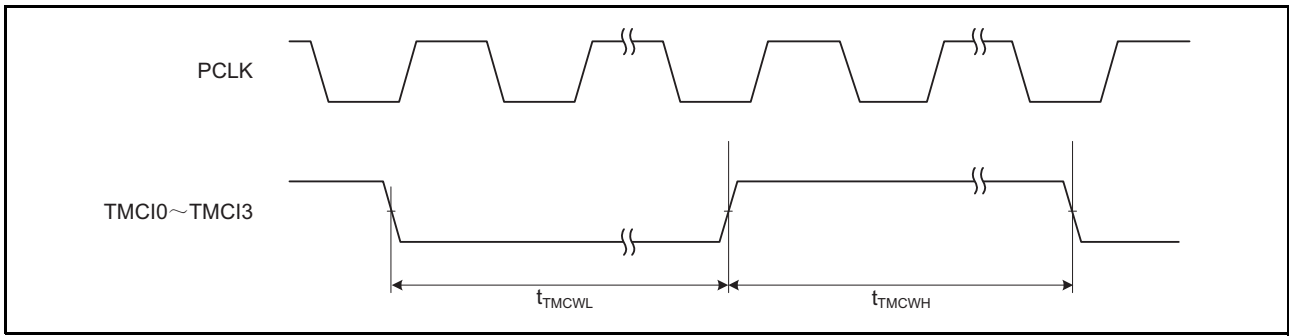


图 41.28 8 位定时器时钟输入时序

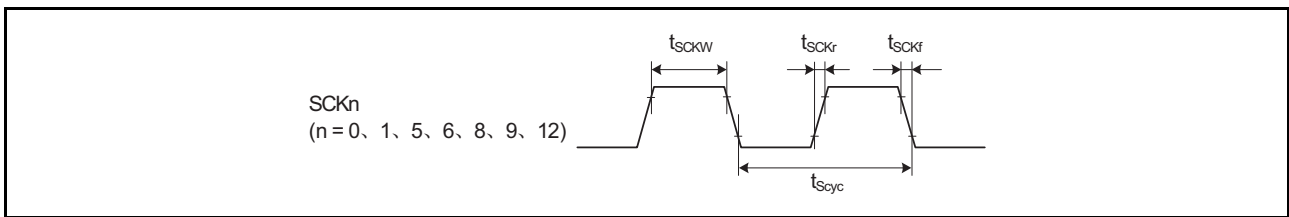


图 41.29 SCK 时钟输入时序

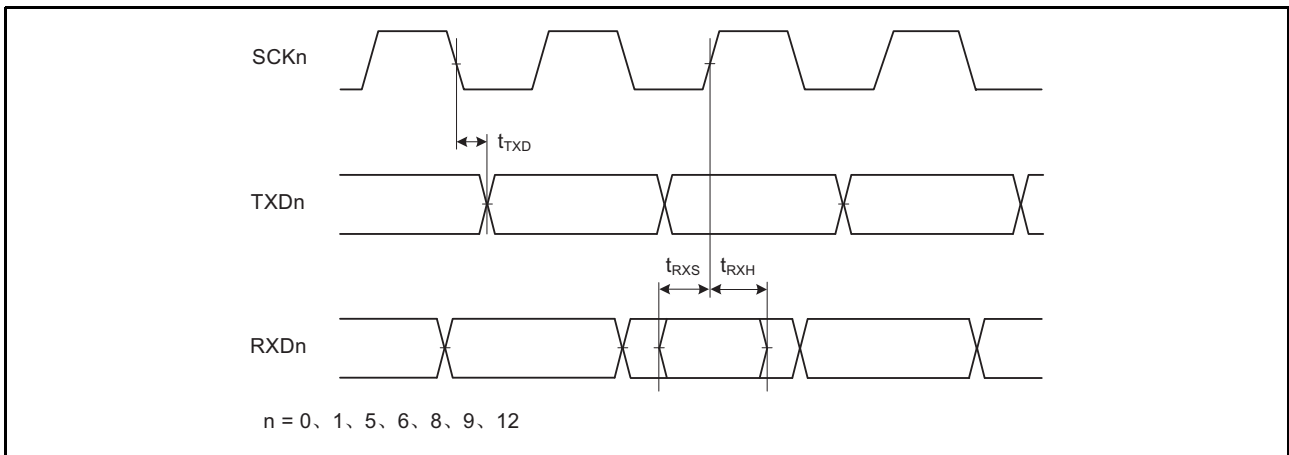


图 41.30 SCI 输入 / 输出时序 / 时钟同步模式

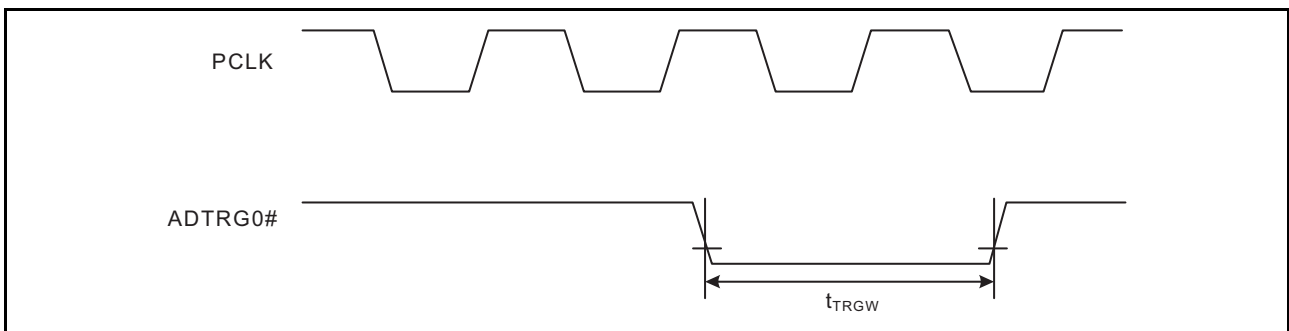


图 41.31 A/D 转换器的外部触发输入时序

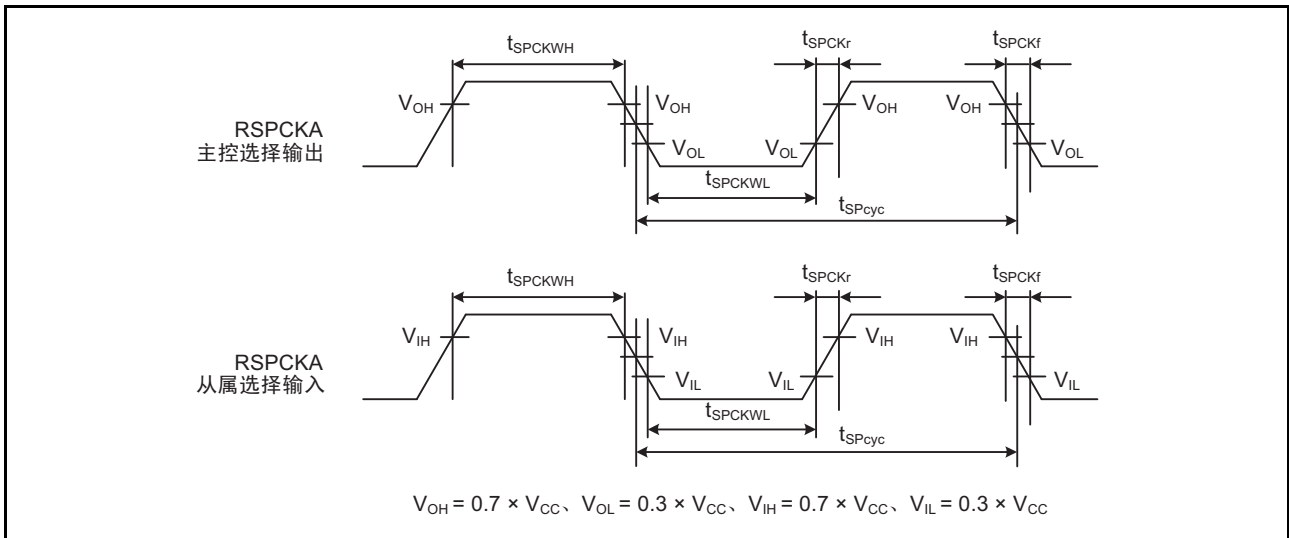


图 41.32 RSPCKA 时钟时序 / 简易 SPI 时钟时序

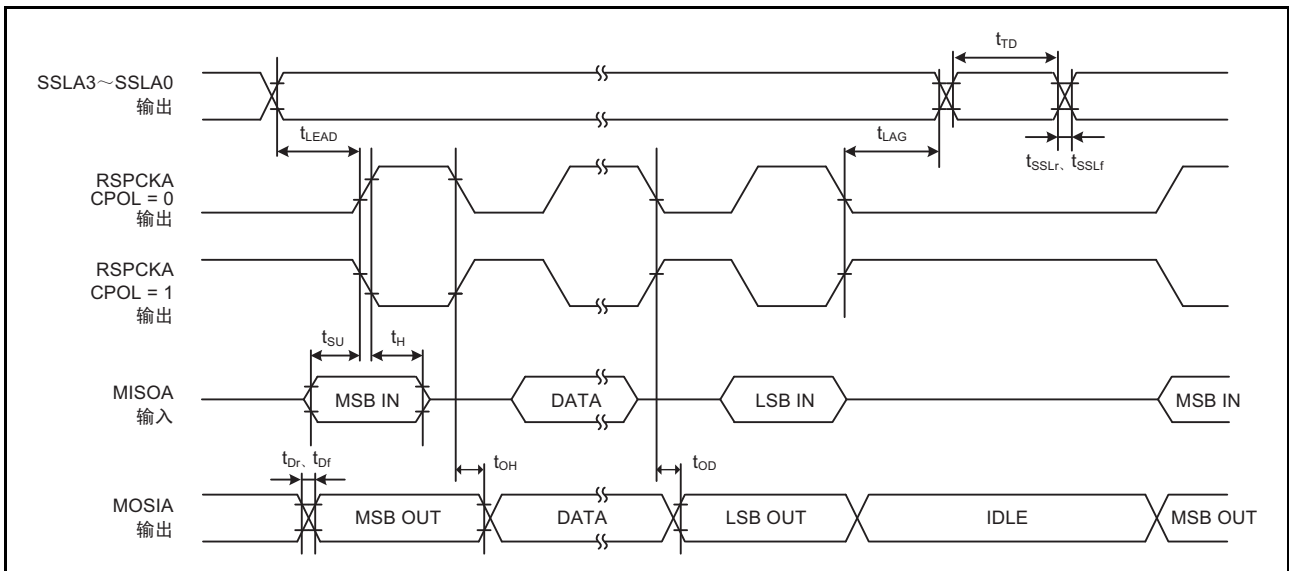


图 41.33 RSPCKA 时序 (主控, CPHA=0) / 简易 SPI 时序 (主控, CPHA=0)

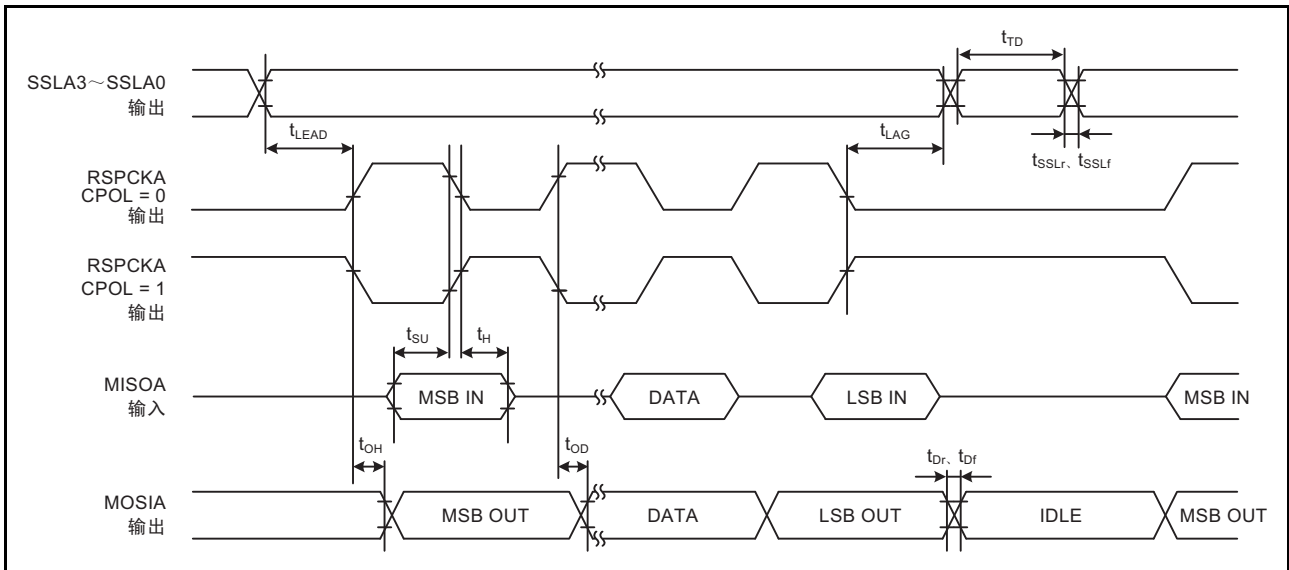


图 41.34 RSPCI 时序 (主控, CPHA=1) / 简易 SPI 时序 (主控, CPHA=1)

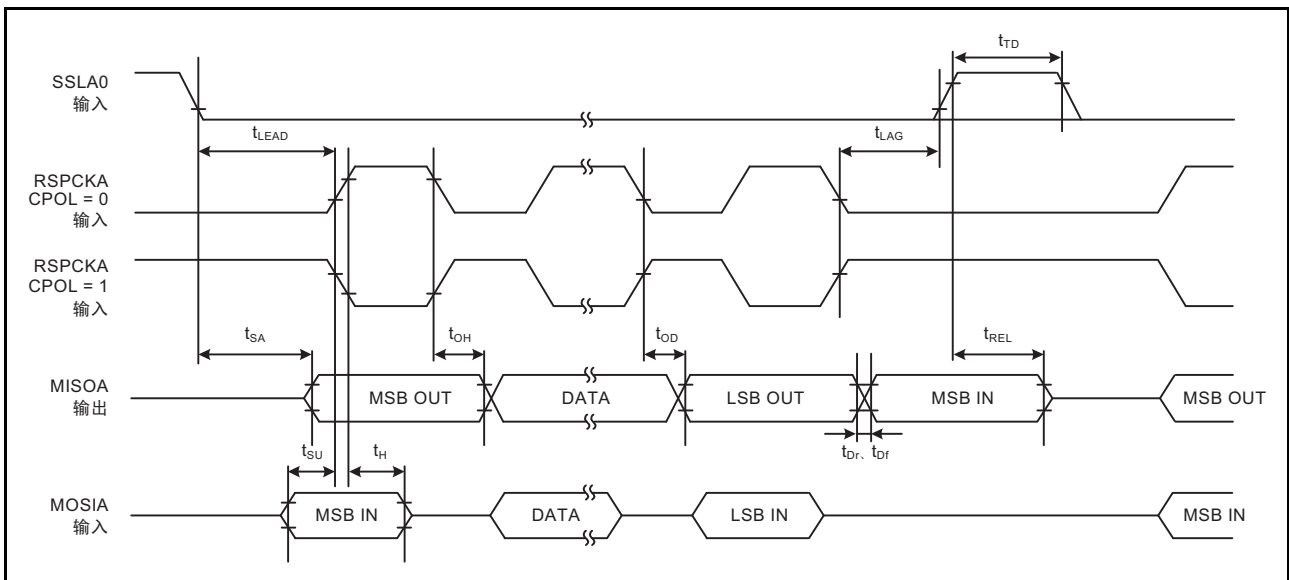


图 41.35 RSPCI 时序 (从属, CPHA=0) / 简易 SPI 时序 (从属, CPHA=0)

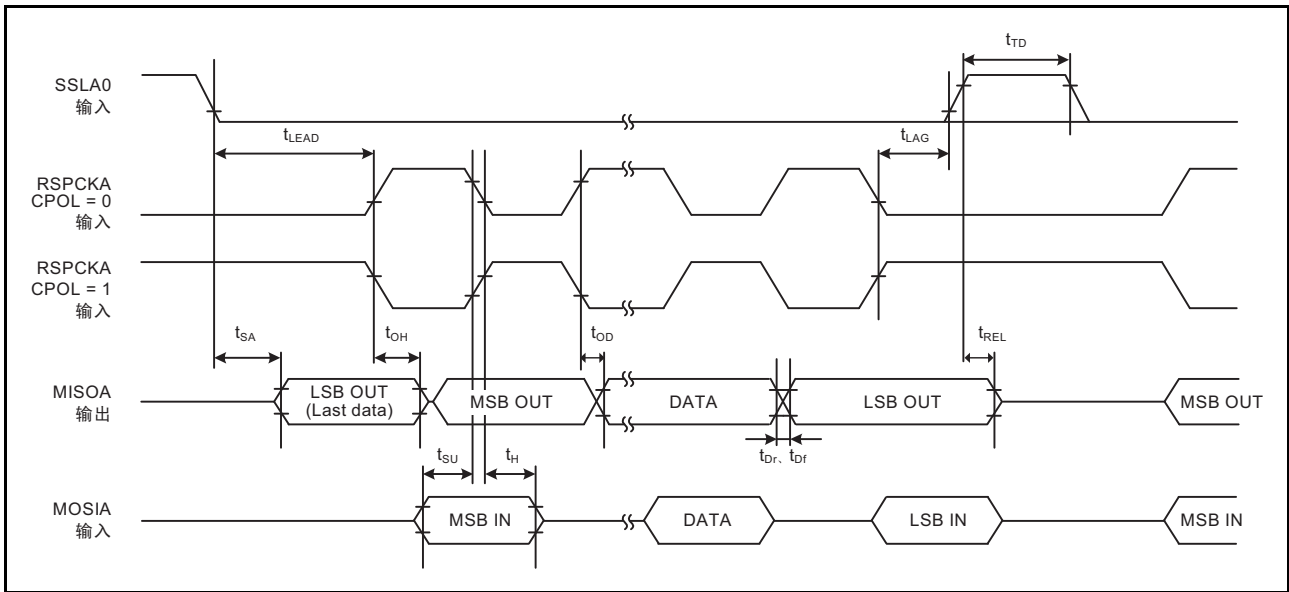


图 41.36 RSPI 时序 (从属, CPHA=1) / 简易 SPI 时序 (从属, CPHA=1)

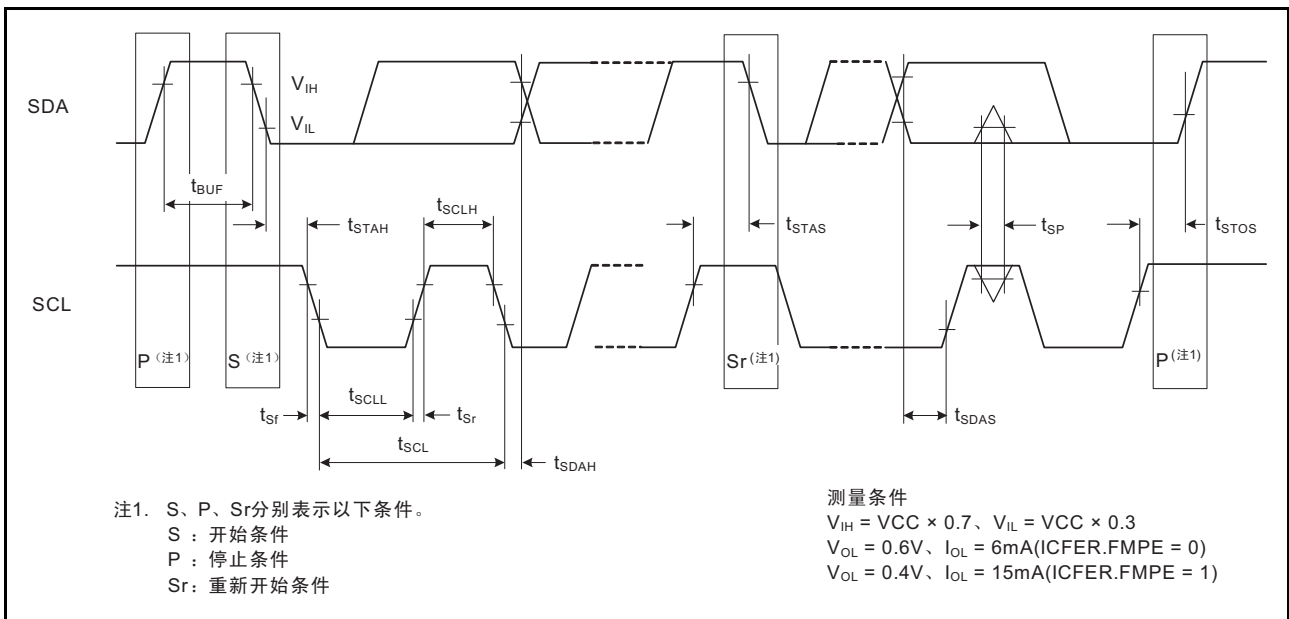


图 41.37 RIIC 总线接口的输入 / 输出时序 / 简易 IIC 总线接口的输入 / 输出时序

41.5 A/D 转换特性

表 41.39 A/D 转换特性 (1)

条件: $V_{CC}=AV_{CC0}=2.7 \sim 5.5V$ 、 $V_{REFH}=V_{REFH0}=(AV_{CC0}-0.9V) \sim AV_{CC0}$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、 $f_{PCLKD}=1 \sim 50MHz$ 、 $T_a=-40 \sim +105^\circ C$

项目		min	typ	max	单位	测量条件
分辨率		—	—	12	位	
转换时间 (注1) ($f_{PCLKD}=50MHz$ 时) (注3)	容许信号源阻抗 max=0.5k Ω	1.0 (0.4) (注2)	—	—	μs	采样的 20 个状态
	容许信号源阻抗 max=1k Ω	1.1 (0.5) (注2)	—	—		采样的 25 个状态
	容许信号源阻抗 max=5k Ω	1.5 (0.9) (注2)	—	—		采样的 45 个状态
模拟输入的电容		—	—	30	pF	
偏移误差		—	± 2.0	± 7.5	LSB	
满刻度误差		—	± 3.0	± 7.5	LSB	
量化误差		—	± 0.5	—	LSB	
绝对精度		—	± 3.0	± 5.0	LSB	高精度通道
		—	± 3.0	± 8.0	LSB	普通精度通道
DNL 微分非线性误差		—	± 2.0	—	LSB	
INL 积分非线性误差		—	± 2.0	± 3.0	LSB	

注. 在选择 HOCO 作为 A/D 转换时钟时, 必须将 PCLKD 设定为 $\leq 40MHz$ 。这是使用 A/D 转换器输入引脚功能时的特性。

注 1. 转换时间是采样时间和比较时间的总和, 各项的测量条件表示采样的状态数。

注 2. () 表示采样时间。

注 3. PCLKD 的频率下限值为 1MHz。

表 41.40 A/D 转换器通道的分类表

分类	对象通道	条件	
高精度通道	AN003 ~ AN007	$AV_{CC0} = 1.62 \sim 5.5V$	在使用 A/D 转换器时, 禁止将 AN000 ~ AN007 引脚用作数字输出。
	AN000、AN001、AN002	$AV_{CC0} = 2.7 \sim 5.5V$ 使用采样 & 保持电路时	
		$AV_{CC0} = 1.62 \sim 5.5V$ 未使用采样 & 保持电路时	
普通精度通道	AN008 ~ AN015	$AV_{CC0} = 1.62 \sim 5.5V$	

表 41.41 A/D 内部基准电压特性

条件: $V_{CC}=AV_{CC0}=1.62 \sim 5.5V$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、 $T_a=-40 \sim +105^\circ C$

项目	min	typ	max	单位	测量条件
A/D 内部基准电压	1.35	1.50	1.65	V	

表 41.42 A/D 转换特性 (2)

条件: $V_{CC}=AV_{CC0}=1.8 \sim 2.7V$ 、 $V_{REFH}=V_{REFH0}=(AV_{CC0}-0.9V) \sim AV_{CC0}$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、 $f_{PCLKD}=1 \sim 32MHz$ 、 $T_a=-40 \sim +105^{\circ}C$

项目		min	typ	max	单位	测量条件
分辨率		—	—	12	位	
转换时间 (注1) ($f_{PCLKD}=25MHz$ 时) (注3)	容许信号源阻抗 max=1k Ω	2.0 (0.8) (注2)	—	—	μs	采样的 20 个状态
	容许信号源阻抗 max=5k Ω	2.2 (1.0) (注2)	—	—		采样的 25 个状态
模拟输入的电容		—	—	30	pF	
偏移误差		—	± 1.5	± 7.5	LSB	
满刻度误差		—	± 2.5	± 7.5	LSB	
量化误差		—	± 0.5	—	LSB	
绝对精度		—	± 3.5	± 8.0	LSB	
DNL 微分非线性误差		—	± 2.0	—	LSB	
INL 积分非线性误差		—	± 2.5	± 3.0	LSB	

注. 这是不使用非 A/D 转换器输入引脚时的特性。

注 1. 转换时间是采样时间和比较时间的总和, 各项的测量条件表示采样的状态数。

注 2. () 表示采样时间。

注 3. PCLKD 的频率下限值为 1MHz。

表 41.43 A/D 转换特性 (3)

条件: $V_{CC}=AV_{CC0}=1.62 \sim 1.8V$ 、 $V_{REFH}=V_{REFH0}=(AV_{CC0}-0.9V) \sim AV_{CC0}$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、 $f_{PCLKD}=1 \sim 16MHz$ 、 $T_a=-40 \sim +105^{\circ}C$

项目		min	typ	max	单位	测量条件
分辨率		—	—	12	位	
转换时间 (注1) ($f_{PCLKD}=12.5MHz$ 时) (注3)	容许信号源阻抗 max=1k Ω	3.36 (0.96) (注2)	—	—	μs	采样的 12 个状态
	容许信号源阻抗 max=5k Ω	3.6 (1.2) (注2)	—	—		采样的 15 个状态
模拟输入的电容		—	—	30	pF	
偏移误差		—	± 1.5	± 7.5	LSB	
满刻度误差		—	± 3.0	± 7.5	LSB	
量化误差		—	± 0.5	—	LSB	
绝对精度		—	± 4.0	± 8.0	LSB	
DNL 微分非线性误差		—	± 2.0	—	LSB	
INL 积分非线性误差		—	± 2.5	± 3.0	LSB	

注. 这是不使用非 A/D 转换器输入引脚时的特性。

注 1. 转换时间是采样时间和比较时间的总和, 各项的测量条件表示采样的状态数。

注 2. () 表示采样时间。

注 3. PCLKD 的频率下限值为 1MHz。

41.6 D/A 转换特性

表 41.44 D/A 转换特性 (1)

条件: $VCC=AVCC0=2.7 \sim 5.5V$ 、 $VREFH=VREFH0=2.7V \sim AVCC0$ 、 $VSS=AVSS0=VREFL=VREFL0=0V$ 、 $fPCLKB \leq 32MHz$ 、 $T_a=-40 \sim +105^\circ C$

项目	min	typ	max	单位	测量条件
分辨率	—	—	10	位	
转换时间	—	—	3.0	μs	负载电容 20pF
绝对精度	—	± 3.0	± 5.0	LSB	负载电阻 4M Ω
	—	—	± 4.0	LSB	负载电阻 8M Ω
RO 输出电阻	—	4.1	—	k Ω	

表 41.45 D/A 转换特性 (2)

条件: $VCC=AVCC0=2.7 \sim 5.5V$ 、 $VREFH=VREFH0=1.8V \sim AVCC0$ 、 $VSS=AVSS0=VREFL=VREFL0=0V$ 、 $fPCLKB \leq 32MHz$ 、 $T_a=-40 \sim +105^\circ C$

项目	min	typ	max	单位	测量条件
分辨率	—	—	10	位	
转换时间	—	—	10.0	μs	负载电容 20pF
绝对精度	—	± 5.0	± 6.0	LSB	负载电阻 4M Ω
	—	—	± 5.0	LSB	负载电阻 8M Ω
RO 输出电阻	—	4.1	—	k Ω	

41.7 温度传感器特性

表 41.46 温度传感器特性

条件: $VCC=AVCC0=1.8 \sim 5.5V$ 、 $VREFH=VREFH0=(AVCC0-0.9V) \sim AVCC0$ 、 $VSS=AVSS0=VREFL=VREFL0=0V$ 、 $T_a=-40 \sim +105^\circ C$

项目	符号	min	typ	max	单位	测量条件
相对精度	—	—	± 1.0	—	$^\circ C$	
温度梯度	$1.8 \leq AVCC0 < 2.7$	—	7.27	—	$mV/^\circ C$	PGAGAIN=00b
	$2.7 \leq AVCC0 < 3.6$	—	10.48	—		PGAGAIN=01b
	$3.6 \leq AVCC0 < 4.5$	—	13.98	—		PGAGAIN=10b
	$4.5 \leq AVCC0 \leq 5.5$	—	21.65	—		PGAGAIN=11b
输出电位 (@ $25^\circ C$)	—	—	1.375	—	V	$VCC=3.6V$
温度传感器的启动时间	t_{START}	—	—	80	μs	图 41.38
采样时间	—	30	72	300	μs	
PGA 重新启动时间	t_{RST_PGA}	—	—	40	μs	

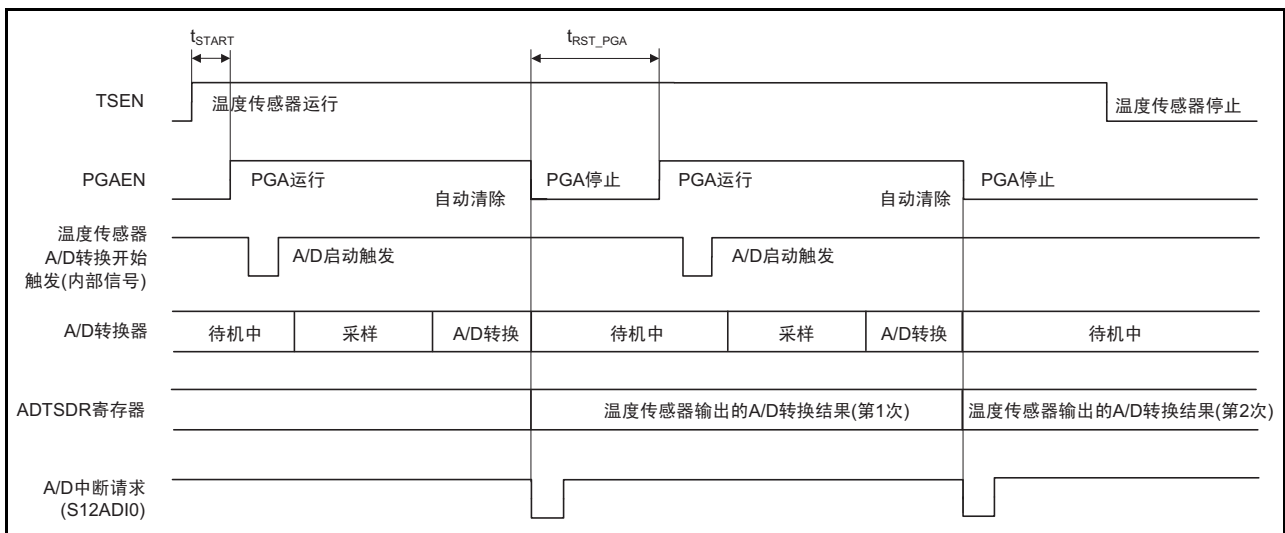


图 41.38 温度传感器的 A/D 转换时序例子 (进行 2 次转换的情况)

41.8 比较器特性

表 41.47 转换器特性

条件: $VCC=AVCC0=1.62 \sim 5.5V$ 、 $VSS=AVSS0=VREFL=VREFL0=0V$ 、 $T_a=-40 \sim +105^\circ C$

项目		符号	min	typ	max	单位	测量条件	
比较器 A	外部基准电压的输入范围	LVREF	1.4	—	VCC	V		
	外部比较电压 (CMPA1、CMPA2) 的输入范围	VI	-0.3	—	VCC+0.3	V		
	偏移量	—	—	±50	150	mV		
	比较器输出的延迟时间 (注 1)		—	—	3	—	μs	下降时 VI=LVREF-110mV
					2	—	μs	下降时 VI < LVREF-1V
					3	—	μs	上升时 VI=LVREF+160mV
1.5					—	μs	上升时 VI > LVREF+1V	
比较器的工作电流	ICMPA	—	0.5	—	μA	VCC=5.0V		
比较器 B	CVREFB0、CVREFB1 的输入基准电压	VREF	0	—	VCC-1.4	V		
	CMPB0、CMPB1 的输入电压	VI	-0.3	—	VCC+0.3	V		
	偏移量	—	—	10	100	mV		
	比较器输出的延迟时间	t _d	—	—	1	μs	VI=VREF+100mV	
	比较器的工作电流	ICMPB	—	75	150	μA	VCC=5.0V、2 个通道的总和	

注 1. 这是数字滤波器无效时的情况。

41.9 上电复位电路和电压检测电路的特性

表 41.48 上电复位电路和电压检测电路的特性 (1)

条件: VCC=AVCC0、VSS=AVSS0=VREFL=VREFL0=0V、T_a=-40 ~ +105°C

项目		符号	min	typ	max	单位	测量条件	
电压检测电平	上电复位 (POR)	低功耗功能无效 (注1)	V _{POR}	1.30	1.40	1.55	V	图 41.39、 图 41.40
		低功耗功能有效 (注2)		1.00	1.20	1.45		
	电压检测电路 (LVDO) (注3)		V _{det0_0}	3.65	3.80	3.95	V	图 41.41
			V _{det0_1}	2.70	2.85	2.90		
			V _{det0_2}	1.80	1.90	2.00		
			V _{det0_3}	1.62	1.72	1.82		
	电压检测电路 (LVD1) (注4)		V _{det1_0}	4.00	4.15	4.30	V	图 41.42 VCC 下降时
			V _{det1_1}	3.85	4.00	4.15		
			V _{det1_2}	3.70	3.85	4.00		
			V _{det1_3}	3.55	3.70	3.85		
			V _{det1_4}	3.40	3.55	3.70		
			V _{det1_5}	3.25	3.40	3.55		
			V _{det1_6}	3.10	3.25	3.40		
			V _{det1_7}	2.95	3.10	3.25		
			V _{det1_8}	2.85	2.95	3.05		
			V _{det1_9}	2.70	2.80	2.90		
			V _{det1_A}	2.55	2.65	2.75		
			V _{det1_B}	2.40	2.50	2.60		
			V _{det1_C}	2.25	2.35	2.45		
V _{det1_D}	2.10	2.20	2.30					
V _{det1_E}	1.95	2.05	2.15					
V _{det1_F}	1.80	1.90	2.00					

注. 这是在电源上没有重叠噪声状态下的特性。

注 1. 这是非软件待机模式和非深度软件待机模式的情况，或者是 FHSSBYCR.SOFTCUT2 位为“0”并且转移到软件待机模式或者 DPSBYCR.DEEPCUT1 位为“0”并且转移到深度软件待机的情况。

注 2. 这是 FHSSBYCR.SOFTCUT2 位为“1”并且转移到软件待机模式或者 DPSBYCR.DEEPCUT1 位为“1”并且转移到深度软件待机模式的情况。

注 3. 符号 V_{det0_#} 中的 # 表示 LDSEL[1:0] 位的值。

注 4. 符号 V_{det1_#} 中的 # 表示 LVDLVL.R.LVD1LVL[3:0] 位的值。

表 41.49 上电复位电路和电压检测电路的特性 (2)

条件: VCC=AVCC0、VSS=AVSS0=VREFL=VREFL0=0V、 $T_a=-40 \sim +105^{\circ}\text{C}$

项目	符号	min	typ	max	单位	测量条件	
电压检测电平	电压检测电路 (LVD2) (注 1)	V_{det2_0}	4.00	4.15	4.30	V	图 41.43 VCC 下降时
		V_{det2_1}	3.85	4.00	4.15		
		V_{det2_2}	3.70	3.85	4.00		
		V_{det2_3}	3.55	3.70	3.85		
		V_{det2_4}	3.40	3.55	3.70		
		V_{det2_5}	3.25	3.40	3.55		
		V_{det2_6}	3.10	3.25	3.40		
		V_{det2_7}	2.95	3.10	3.25		
		V_{det2_8}	2.85	2.95	3.05		
		V_{det2_9}	2.70	2.80	2.90		
		V_{det2_A}	2.55	2.65	2.75		
		V_{det2_B}	2.40	2.50	2.60		
		V_{det2_C}	2.25	2.35	2.45		
		V_{det2_D}	2.10	2.20	2.30		
		V_{det2_E}	1.95	2.05	2.15		
V_{det2_F}	1.80	1.90	2.00				
	$V_{\text{det2_EXT}}$	1.18	1.33	1.48		EXVCCINP2=1	
内部复位时间	上电复位时间	t_{POR}	—	9	—	ms	图 41.40
	电压监视 0 复位时间	t_{LVD0}	—	9	—		图 41.41
	电压监视 1 复位时间	t_{LVD1}	—	1.4	—		图 41.42
	电压监视 2 复位时间	t_{LVD2}	—	1.4	—		图 41.43
最短 VCC 降低时间 (注 2)	t_{VOFF}	200	—	—	μs	图 41.40	
应答延迟时间	t_{det}	—	—	200	μs	图 41.40	
LVD 运行稳定时间 (选择 LVD 有效时)	$T_{\text{d(E-A)}}$	—	—	15	μs	图 41.42、图 41.43	
上电复位有效时间	$t_{\text{W(POR)}}$	1	—	—	ms	图 41.40 VCC $\leq 0.9\text{V}$	
滞后宽度 (电压检测电路 (LVD1、LVD2))	V_{LVH}	—	100	—	mV	选择 VdetX_0 ~ 7 时	
		—	50	—		选择 VdetX_8 ~ F 时	

注. 这是在电源上没有重叠噪声状态下的特性。

注 1. 符号 Vdet2_# 中 # 表示 LVDLVLRLVD2LVL[3:0] 位的值。

注 2. 最短 VCC 下降时间是指 VCC 的值小于 POR/LVD 的电压检测电平 V_{POR} 、 V_{det0} 、 V_{det1} 、 V_{det2} 的 min 值时的时间。

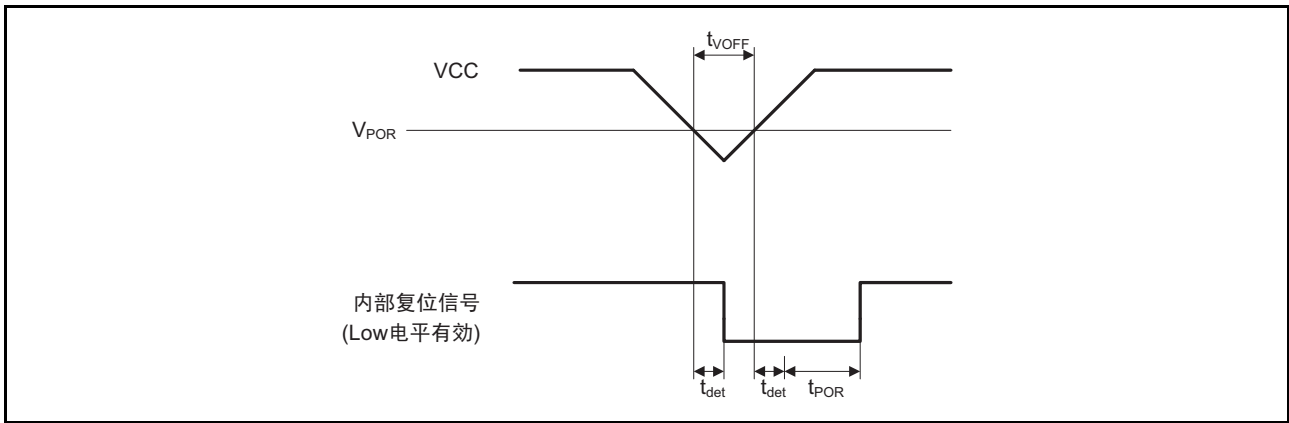


图 41.39 电压检测复位时序

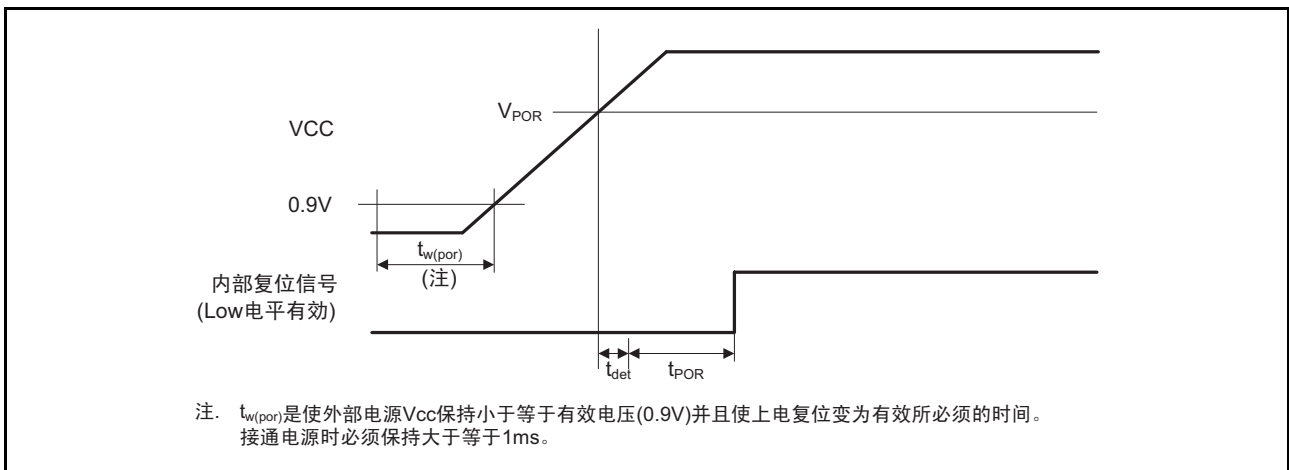


图 41.40 上电复位时序

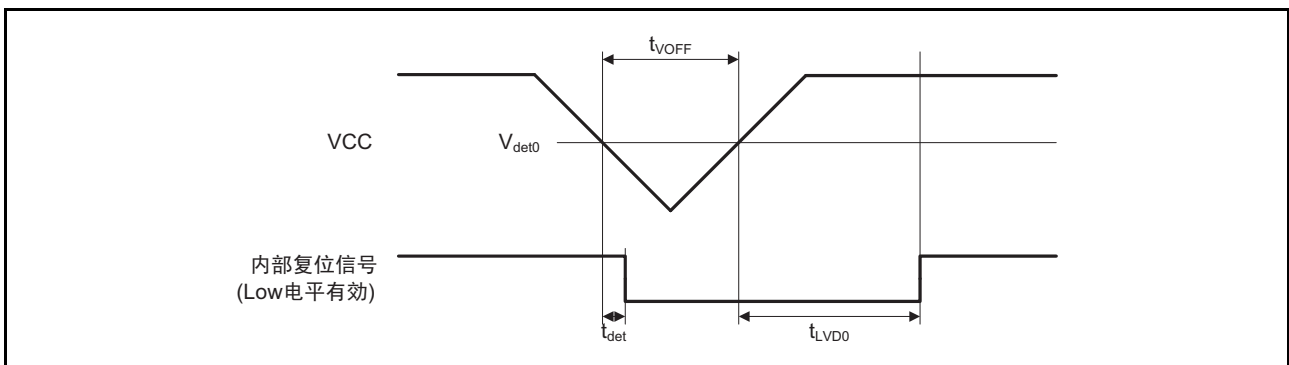


图 41.41 电压检测电路时序 (V_{det0})

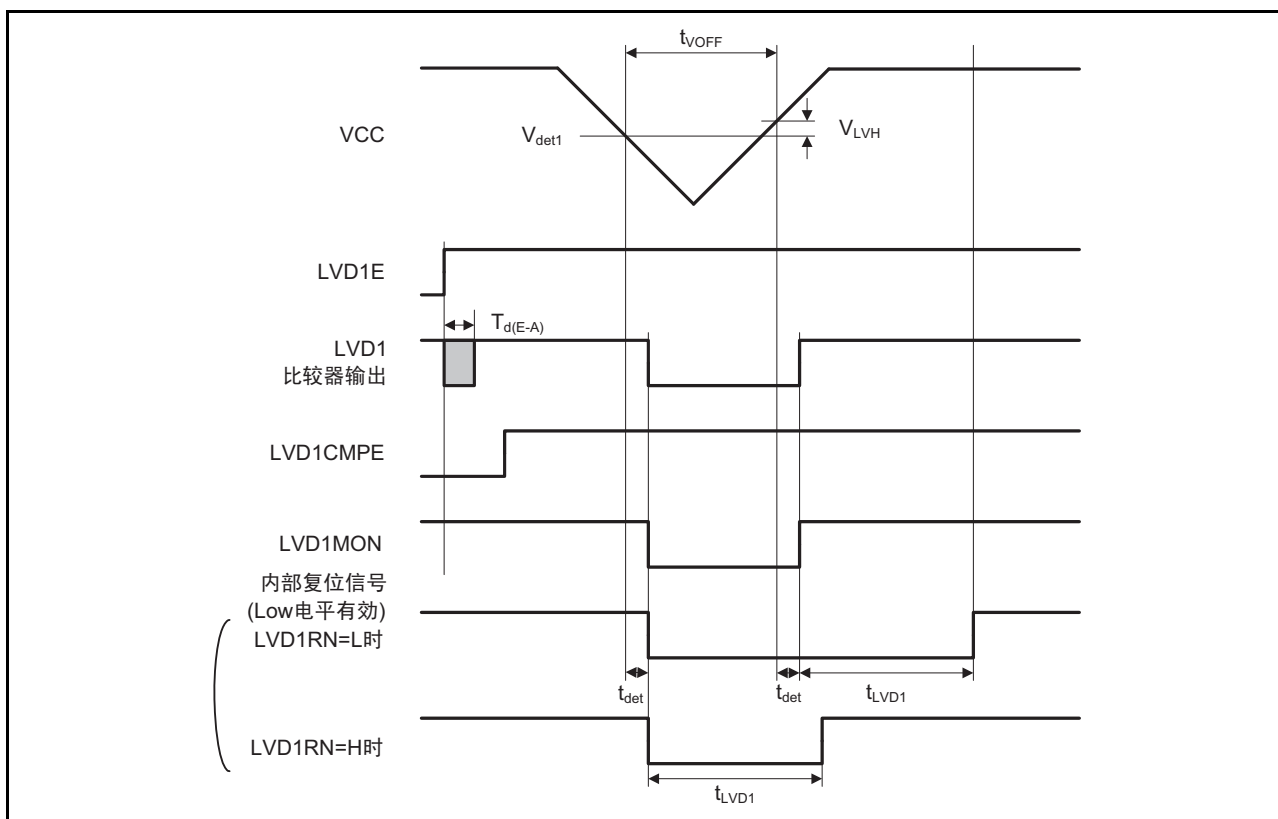


图 41.42 电压检测电路时序 (V_{det1})

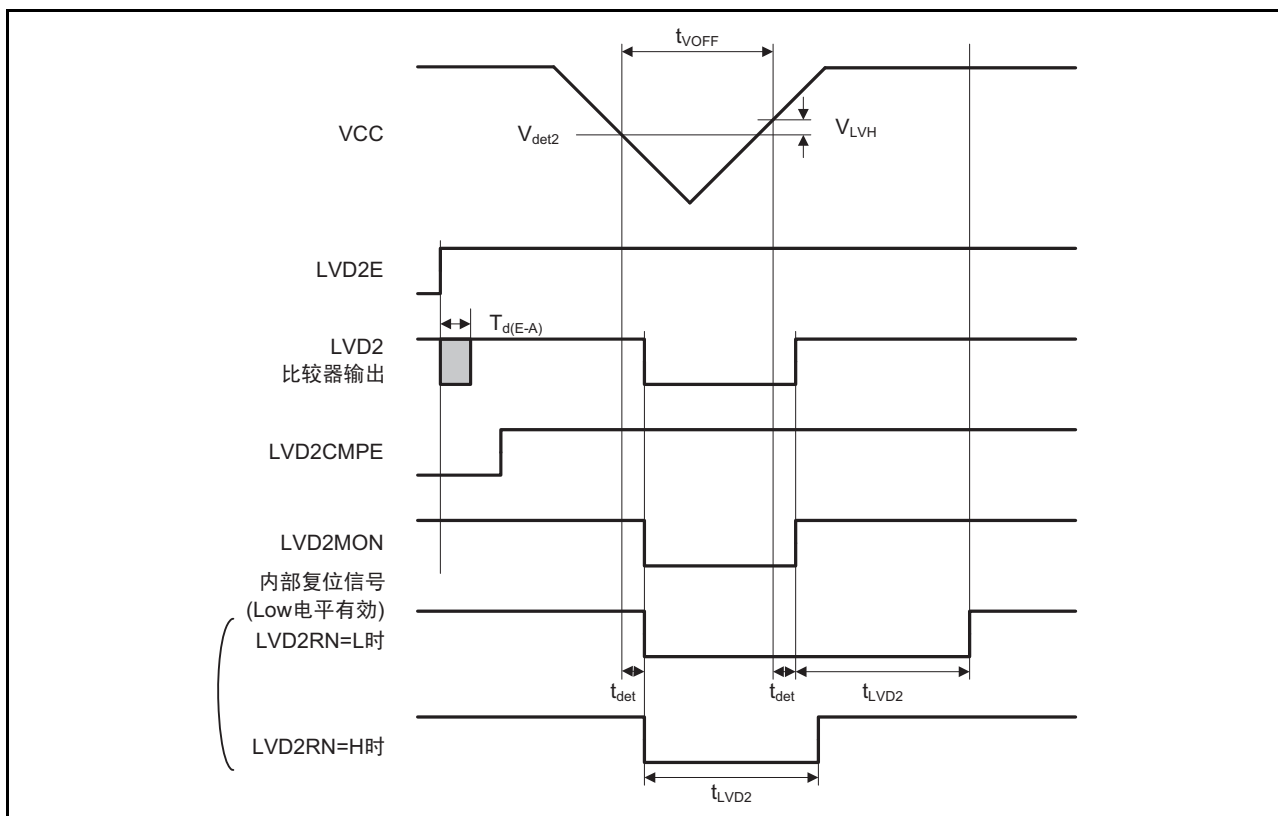


图 41.43 电压检测电路时序 (V_{det2})

41.10 振荡停止检测时序

表 41.50 振荡停止检测电路特性

条件: $V_{CC}=AV_{CC0}$ 、 $V_{SS}=AV_{SS0}=V_{REFL}=V_{REFL0}=0V$ 、 $T_a=-40 \sim +105^{\circ}C$

项目	符号	min	typ	max	单位	测量条件
检测时间	t_{dr}	—	—	1	ms	图 41.44

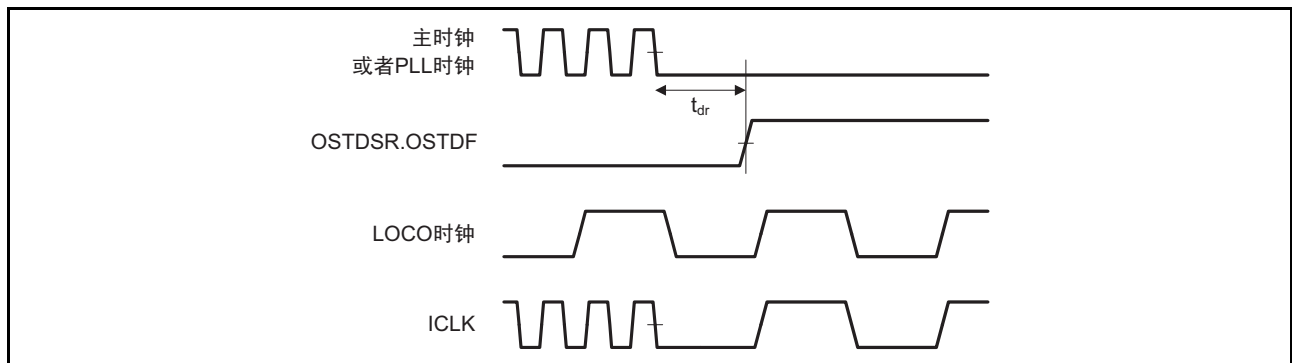


图 41.44 振荡停止检测时序

41.11 ROM（保存代码的闪存）特性

表 41.51 ROM（保存代码的闪存）特性（1）

项目	符号	min	typ	max	单位	测量条件
再编程 / 擦除周期（注1）	N_{PEC}	1000	—	—	次	
数据保持时间（注2）	t_{DRP}	10	—	—	年	
FCU 复位时间	t_{FCUR}	$\geq 20\mu s$ 并且 $\geq FCLK \times 6$	—	—	μs	

注 1. 再编程 / 擦除周期的定义:

再编程 / 擦除周期是指每块的擦除次数。如果再编程 / 擦除周期为 n ($n=1000$)，就能逐块擦除 n 次。例如，对于 2K 字节的块，如果将 128 字节的数据分 16 次写到不同的地址后擦除该块，再编程 / 擦除周期次数就计为 1。但是，对于 1 次的擦除，不能对相同的地址进行多次编程（禁止重写）。

注 2. 这是在包括 min 值的规格范围内进行改写时的特性。

表 41.52 ROM（保存代码的闪存）特性（2）（高速运行模式、中速运行模式 A）

条件：VCC=AVCC0=2.7 ~ 5.5V、VREFH=VREFH0=AVCC0、VSS=AVSS0=VREFL=VREFL0=0V、
编程 / 擦除时的工作温度范围：T_a=-40 ~ +105°C

项目		符号	min	typ	max	单位	测量条件
编程时间 N _{PEC} ≤ 100 次时	2 字节	t _{P2}	—	0.5	2.5	ms	FCLK=32MHz
	8 字节	t _{P8}	—	0.5	2.5		
	128 字节	t _{P128}	—	1.0	4.8		
编程时间 N _{PEC} > 100 次时	2 字节	t _{P2}	—	—	3.0	ms	FCLK=32MHz
	8 字节	t _{P8}	—	—	3.2		
	128 字节	t _{P128}	—	—	6.0		
擦除时间 N _{PEC} ≤ 100 次时	2K 字节	t _{E2K}	—	16	29	ms	FCLK=32MHz
擦除时间 N _{PEC} > 100 次时	2K 字节	t _{E2K}	—	—	60	ms	FCLK=32MHz
编程时的挂起延迟时间（在编程 / 擦除优先模式中）		t _{SPD}	—	—	0.8	ms	图 41.45 FCLK=32MHz
编程时的第 1 次挂起延迟时间（在挂起优先模式中）		t _{SPSD1}	—	—	120	μs	
编程时的第 2 次挂起延迟时间（在挂起优先模式中）		t _{SPSD2}	—	—	0.8	ms	
擦除时的挂起延迟时间（在编程 / 擦除优先模式中）		t _{SED}	—	—	0.8	ms	
擦除时的第 1 次挂起延迟时间（在挂起优先模式中）		t _{SESD1}	—	—	120	μs	
擦除时的第 2 次挂起延迟时间（在挂起优先模式中）		t _{SESD2}	—	—	0.8	ms	

表 41.53 ROM（保存代码的闪存）特性（3）（中速运行模式 B）

条件：VCC=AVCC0=1.62 ~ 3.6V、VREFH=VREFH0=AVCC0、VSS=AVSS0=VREFL=VREFL0=0V、
编程 / 擦除时的工作温度范围：T_a=-40 ~ +105°C

项目		符号	min	typ	max	单位	测量条件
编程时间（注 1） N _{PEC} ≤ 100 次时	2 字节	t _{P2}	—	0.8	3.5	ms	FCLK=32MHz
	8 字节	t _{P8}	—	0.8	3.5		
	128 字节	t _{P128}	—	1.6	8.3		
编程时间 N _{PEC} > 100 次时	2 字节	t _{P2}	—	—	4.2	ms	FCLK=32MHz
	8 字节	t _{P8}	—	—	4.5		
	128 字节	t _{P128}	—	—	10		
擦除时间 N _{PEC} ≤ 100 次时	2K 字节	t _{E2K}	—	27	46	ms	FCLK=32MHz
擦除时间 N _{PEC} > 100 次时	2K 字节	t _{E2K}	—	—	90	ms	FCLK=32MHz
编程时的挂起延迟时间（在编程 / 擦除优先模式中）		t _{SPD}	—	—	1.6	ms	图 41.45 FCLK=32MHz （注 1）
编程时的第 1 次挂起延迟时间（在挂起优先模式中）		t _{SPSD1}	—	—	120	μs	
编程时的第 2 次挂起延迟时间（在挂起优先模式中）		t _{SPSD2}	—	—	1.6	ms	
擦除时的挂起延迟时间（在编程 / 擦除优先模式中）		t _{SED}	—	—	1.6	ms	
擦除时的第 1 次挂起延迟时间（在挂起优先模式中）		t _{SESD1}	—	—	120	μs	
擦除时的第 2 次挂起延迟时间（在挂起优先模式中）		t _{SESD2}	—	—	1.6	ms	

注 1. 当 1.62 ≤ 电压范围 < 1.8V 时，工作频率为 20MHzmax。

41.12 E2 数据闪存（保存数据的闪存）特性

表 41.54 E2 数据闪存（保存数据的闪存）特性（1）

项目	符号	min	typ	max	单位	测量条件
在编程 / 擦除周期（注1）	N_{DPEC}	100000	—	—	次	
数据保持时间（注2）	t_{DRP}	10	—	—	年	

注 1. 再编程 / 擦除周期的定义:

再编程 / 擦除周期是指每块的擦除次数。如果再编程 / 擦除周期为 n ($n=100000$)，就能逐块擦除 n 次。例如，对于 128 字节的块，如果将 8 字节的数据分 16 次写到不同的地址后擦除该块，再编程 / 擦除周期次数就计为 1。但是，对于 1 次的擦除，不能对相同的地址进行多次编程（禁止重写）。

注 2. 这是在包括 min 值的规格范围内进行改写时的特性。

表 41.55 E2 数据闪存（保存数据的闪存）特性（2）（高速运行模式、中速运行模式 A）

条件: $VCC=AVCC0=2.7 \sim 5.5V$ 、 $VREFH=VREFH0=AVCC0$ 、 $VSS=AVSS0=VREFL=VREFL0=0V$ 、
编程 / 擦除时的工作温度范围: $T_a=-40 \sim +105^\circ C$

项目		符号	min	typ	max	单位	测量条件
编程时间 $N_{DPEC} \leq 100$ 次时	2 字节	t_{DP2}	—	0.3	2.0	ms	FCLK=32MHz
	8 字节	t_{DP8}	—	0.4	2.2		
编程时间 $N_{DPEC} > 100$ 次时	2 字节	t_{DP2}	—	—	3.0	ms	FCLK=32MHz
	8 字节	t_{DP8}	—	—	3.2		
擦除时间 $N_{DPEC} \leq 100$ 次时	128 字节	t_{DE128}	—	4.5	8	ms	FCLK=32MHz
	128 字节	t_{DE128}	—	—	12		
空白检查时间	2 字节	t_{DBC2}	—	—	35	μs	FCLK=32MHz
	2K 字节	t_{DBC2K}	—	—	2.5		
编程时的挂起延迟时间（在编程 / 擦除优先模式中）		t_{DPSD}	—	—	0.8	ms	图 41.45 FCLK=32MHz
编程时的第 1 次挂起延迟时间（在挂起优先模式中）		t_{DPSD1}	—	—	120	μs	
编程时的第 2 次挂起延迟时间（在挂起优先模式中）		t_{DPSD2}	—	—	0.8	ms	
擦除时的挂起延迟时间（在编程 / 擦除优先模式中）		t_{DSED}	—	—	0.8	ms	
擦除时的第 1 次挂起延迟时间（在挂起优先模式中）		t_{DSED1}	—	—	120	μs	
擦除时的第 2 次挂起延迟时间（在挂起优先模式中）		t_{DSED2}	—	—	0.8	ms	

表 41.56 E2 数据闪存（保存数据的闪存）特性（3）（中速运行模式 B）

条件：VCC=AVCC0=1.62 ~ 3.6V、VREFH=VREFH0=AVCC0、VSS=AVSS0=VREFL=VREFL0=0V、
编程 / 擦除时的工作温度范围：T_a=-40 ~ +105°C

项目		符号	min	typ	max	单位	测量条件
编程时间（注1）	2 字节	t _{DP2}	—	0.6	2.8	ms	FCLK=32MHz
	8 字节	t _{DP8}	—	0.6	3.2		
	2 字节	t _{DP2}	—	—	4.2	ms	FCLK=32MHz
	8 字节	t _{DP8}	—	—	4.5		
擦除时间	128 字节	t _{DE128}	—	7	12	ms	FCLK=32MHz
	128 字节	t _{DE128}	—	—	17	ms	FCLK=32MHz
空白检查时间	2 字节	t _{DBC2}	—	—	40	μs	FCLK=32MHz (注1)
	2K 字节	t _{DBC2K}	—	—	2.6	ms	
编程时的挂起延迟时间（在编程 / 擦除优先模式中）		t _{DSPD}	—	—	1.6	ms	图 41.45 FCLK=32MHz (注1)
编程时的第 1 次挂起延迟时间（在挂起优先模式中）		t _{DSPSD1}	—	—	120	μs	
编程时的第 2 次挂起延迟时间（在挂起优先模式中）		t _{DSPSD2}	—	—	1.6	ms	
擦除时的挂起延迟时间（在编程 / 擦除优先模式中）		t _{DSED}	—	—	1.6	ms	
擦除时的第 1 次挂起延迟时间（在挂起优先模式中）		t _{DSESD1}	—	—	12	μs	
擦除时的第 2 次挂起延迟时间（在挂起优先模式中）		t _{DSESD2}	—	—	1.6	ms	

注 1. 当 1.62 ≤ 电压范围 < 1.8V 时，工作频率为 20MHzmax。

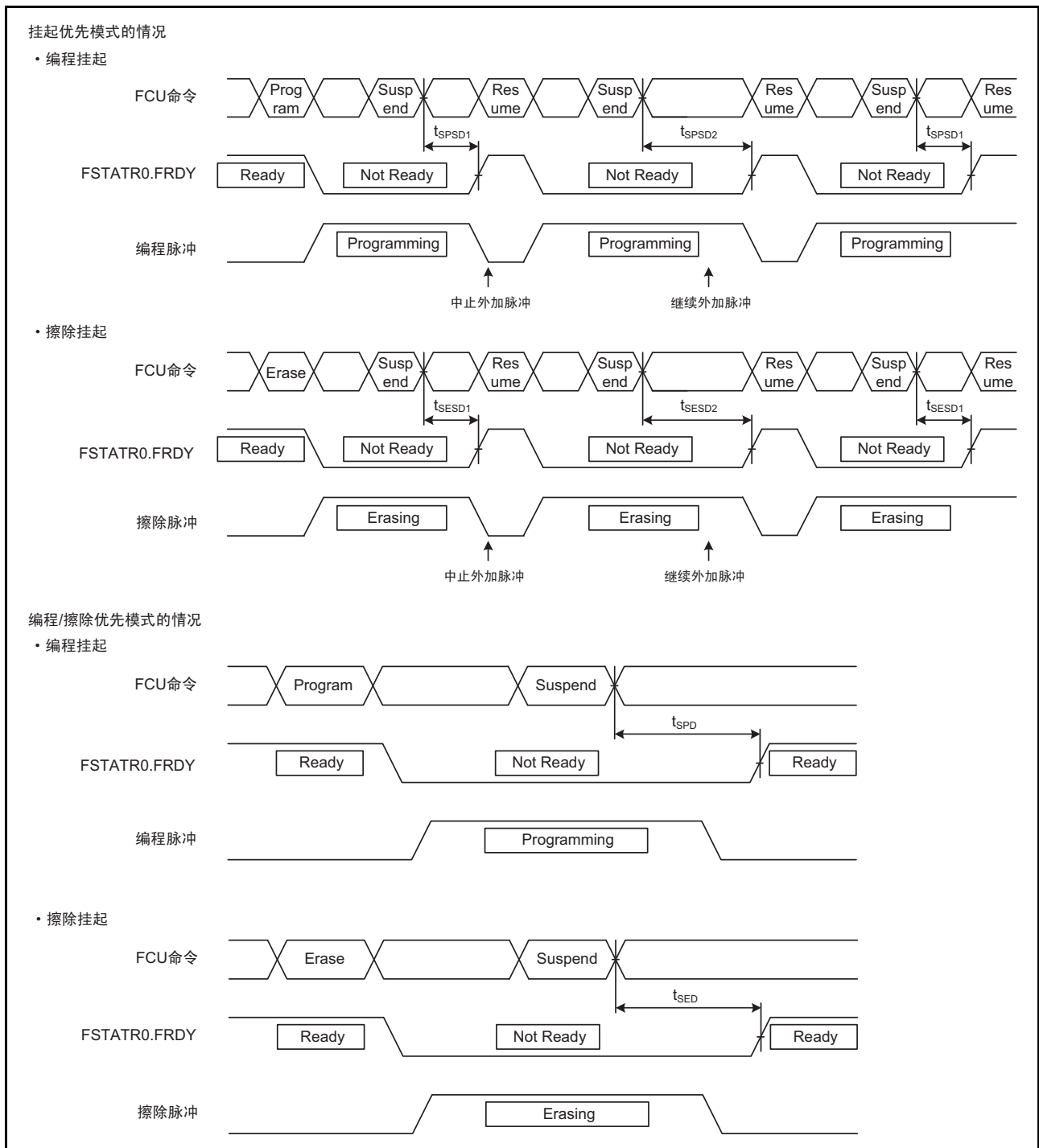


图 41.45 闪存的编程 / 擦除挂起时序

附录

附录 1. 各运行模式中的端口状态

附表 1.1 各运行模式中的端口状态 (1/3)

端口名 引脚名	寄存器设定的运行模式		复位	软件待机模式		深度软件待机模 式 IOKEEP=1/0	解除深度软件待机模式后 (返回到启动模式)	
				OPE=1	OPE=0		IOKEEP=1 (注1)	IOKEEP=0
P03 (DA0)	全部模式	输出 DA0 时 (DAOE0=1)	Hi-Z	保持 DA 输出		Hi-Z	Hi-Z	Hi-Z
		上述以外 (DAOE0=0)	Hi-Z	Keep-O		Keep	Keep	
P05 (DA1)	全部模式	输出 DA1 时 (DAOE1=1)	Hi-Z	保持 DA 输出		Hi-Z	Hi-Z	Hi-Z
		上述以外 (DAOE1=0)	Hi-Z	Keep-O		Keep	Keep	
P07	全部模式		Hi-Z	Keep-O		Keep	Keep	Hi-Z
P12 ~ P15 (IRQ2 ~ IRQ5)	全部模式		Hi-Z	Keep-O (注2)		Keep	Keep	Hi-Z
P16、P17 (IRQ6/SCL-DS、 IRQ7/SDA-DS)	全部模式		Hi-Z	Keep-O (注2)		Keep-O (注3)	Keep	Hi-Z
P20 ~ P23	全部模式		Hi-Z	Keep-O		Keep	Keep	Hi-Z
P24 ~ P27 (CS0#、CS1#、 CS2#、CS3#)	单芯片模式 (EXBE=0)		Hi-Z	Keep-O		Keep	Keep	Hi-Z
	内部 ROM 有效 / 无效扩展 模式 (EXBE=1)			[输出 CS 时] H [上述以外] Keep-O	[输出 CS 时] Hi-Z [上述以外] Keep-O			
P30 ~ P33 (IRQ0-DS、 IRQ1-DS、 IRQ2-DS、 IRQ3-DS)	全部模式		Hi-Z	Keep-O (注2)		Keep-O (注3)	Keep	Hi-Z
P34 (IRQ4)	全部模式		Hi-Z	Keep-O (注2)		Keep	Keep	Hi-Z
P35 (NMI)	全部模式		Hi-Z	Keep-O (注2)		Keep-O (注3)	Keep	Hi-Z
P36、P37	全部模式		Hi-Z	Keep-O		Keep	Keep	Hi-Z
P40 ~ P47	全部模式		Hi-Z	Keep-O		Keep	Keep	Hi-Z
P50 (WR0#/WR#)	单芯片模式 (EXBE=0)		Hi-Z	Keep-O		Keep	Keep	Hi-Z
	内部 ROM 有效 / 无效扩展 模式 (EXBE=1)			[输出 WR0#/ WR# 时] H	[输出 WR0#/ WR# 时] Hi-Z			
P51 (WR1#/BC1#)	单芯片模式 (EXBE=0)		Hi-Z	Keep-O		Keep	Keep	Hi-Z
	内部 ROM 有效 / 无效扩展 模式 (EXBE=1)			[输出 WR1#/ BC1# 时] H [上述以外] Keep-O	[输出 WR1#/ BC1# 时] Hi-Z [上述以外] Keep-O			
P52 (RD#)	单芯片模式 (EXBE=0)		Hi-Z	Keep-O		Keep	Keep	Hi-Z
	内部 ROM 有效 / 无效扩展 模式 (EXBE=1)			[输出 RD# 时] H	[输出 RD# 时] Hi-Z			

附表 1.1 各运行模式中的端口状态 (2/3)

端口名 引脚名	寄存器设定的运行模式	复位	软件待机模式		深度软件待机模 式 IOKEEP=1/0	解除深度软件待机模式后 (返回到启动模式)	
			OPE=1	OPE=0		IOKEEP=1 (注1)	IOKEEP=0
P53 (BCLK)	全部模式	Hi-Z	[输出时钟时] H [上述以外] Keep-O		Keep	Keep	Hi-Z
P54/ALE	单芯片模式 (EXBE=0)	Hi-Z	Keep-O		Keep	Keep	Hi-Z
	内部 ROM 有效 / 无效扩展 模式 (EXBE=1)		[输出 ALE 时] L [上述以外] Keep-O	[输出 ALE 时] Hi-Z [上述以外] Keep-O			
P55	全部模式	Hi-Z	Keep-O		Keep	Keep	Hi-Z
PA0 (A0/BC0#)	单芯片模式 (EXBE=0)	Hi-Z	Keep-O		Keep	Keep	Hi-Z
	内部 ROM 有效 / 无效扩展 模式 (EXBE=1)		[输出地址时] 保持地址输出 [输出 BC0# 时] H [上述以外] Keep-O	[输出地址时] Hi-Z [输出 BC0# 时] Hi-Z [上述以外] Keep-O			
PA1、PA2 (A1、A2)	单芯片模式 (EXBE=0)	Hi-Z	Keep-O		Keep	Keep	Hi-Z
	内部 ROM 有效 / 无效扩展 模式 (EXBE=1)		[输出地址时] 保持地址输出 [上述以外] Keep-O	[输出地址时] Hi-Z [上述以外] Keep-O			
PA3、PA4 (IRQ6-DS/A3、 IRQ5-DS/A4)	单芯片模式 (EXBE=0)	Hi-Z	Keep-O (注2)		Keep-O (注3)	Keep	Hi-Z
	内部 ROM 有效 / 无效扩展 模式 (EXBE=1)		[输出地址时] 保持地址输出 [上述以外] Keep-O	[输出地址时] Hi-Z [上述以外] Keep-O			
PA5 ~ PA7 (A5 ~ A7)	单芯片模式 (EXBE=0)	Hi-Z	Keep-O		Keep	Keep	Hi-Z
	内部 ROM 有效 / 无效扩展 模式 (EXBE=1)		[输出地址时] 保持地址输出 [上述以外] Keep-O	[输出地址时] Hi-Z [上述以外] Keep-O			
PB0 (A8)	单芯片模式 (EXBE=0)	Hi-Z	Keep-O		Keep	Keep	Hi-Z
	内部 ROM 有效 / 无效扩展 模式 (EXBE=1)		[输出地址时] 保持地址输出 [上述以外] Keep-O	[输出地址时] Hi-Z [上述以外] Keep-O			
PB1 (IRQ4-DS/A9)	单芯片模式 (EXBE=0)	Hi-Z	Keep-O (注2)		Keep-O (注3)	Keep	Hi-Z
	内部 ROM 有效 / 无效扩展 模式 (EXBE=1)		[输出地址时] 保持地址输出 [上述以外] Keep-O	[输出地址时] Hi-Z [上述以外] Keep-O			
PB2 ~ PB7 (A10 ~ A15)	单芯片模式 (EXBE=0)	Hi-Z	Keep-O		Keep	Keep	Hi-Z
	内部 ROM 有效 / 无效扩展 模式 (EXBE=1)		[输出地址时] 保持地址输出 [上述以外] Keep-O	[输出地址时] Hi-Z [上述以外] Keep-O			
PC0 ~ PC3 (A16 ~ 19)	单芯片模式 (EXBE=0)	Hi-Z	Keep-O		Keep	Keep	Hi-Z
	内部 ROM 有效 / 无效扩展 模式 (EXBE=1)		[输出地址时] 保持地址输出 [上述以外] Keep-O	[输出地址时] Hi-Z [上述以外] Keep-O			

附表 1.1 各运行模式中的端口状态 (3/3)

端口名 引脚名	寄存器设定的运行模式		复位	软件待机模式		深度软件待机模 式 IOKEEP=1/0	解除深度软件待机模式后 (返回到启动模式)	
				OPE=1	OPE=0		IOKEEP=1 (注1)	IOKEEP=0
PC4 ~ PC7 (A20/CS3# ~ A23/CS0#)	单芯片模式 (EXBE=0)		Hi-Z	Keep-O		Keep	Keep	Hi-Z
	内部 ROM 有效 / 无效扩展 模式 (EXBE=1)			[输出地址时] 保持地址输出 [输出 CS 时] H [上述以外] Keep-O	[输出地址时] Hi-Z [输出 CS 时] Hi-Z [上述以外] Keep-O			
PD0 ~ PD7 (D0/IRQ0 ~ D7/IRQ7)	单芯片模式 (EXBE=0)		Hi-Z	Keep-O (注2)		Keep	Keep	Hi-Z
	内部 ROM 有效 / 无效扩展 模式 (EXBE=1)			Hi-Z				
PE0、PE1 (D8、D9)	单芯片模式 (EXBE=0)		Hi-Z	Keep-O		Keep	Keep	Hi-Z
	内部 ROM 有效 / 无效 扩展模式 (EXBE=1)	8 位总线 宽度		Keep-O				
		16 位总线 宽度		Hi-Z				
PE2 (D10/IRQ7-DS)	单芯片模式 (EXBE=0)		Hi-Z	Keep-O (注2)		Keep-O (注3)	Keep	Hi-Z
	内部 ROM 有效 / 无效 扩展模式 (EXBE=1)	8 位总线 宽度		Keep-O (注2)				
		16 位总线 宽度		Hi-Z				
PE3、PE4 (D11、D12)	单芯片模式 (EXBE=0)		Hi-Z	Keep-O		Keep	Keep	Hi-Z
	内部 ROM 有效 / 无效 扩展模式 (EXBE=1)	8 位总线 宽度		Keep-O				
		16 位总线 宽度		Hi-Z				
PE5 ~ PE7 (D13/IRQ5 ~ D15/IRQ7)	单芯片模式 (EXBE = 0)		Hi-Z	Keep-O (注2)		Keep	Keep	Hi-Z
	内部 ROM 有效 / 无效 扩展模式 (EXBE=1)	8 位总线 宽度		Keep-O (注2)				
		16 位总线 宽度		Hi-Z				
PH0	全部模式		Hi-Z	Keep-O		Keep	Keep	Hi-Z
PH1、PH2 (IRQ0、IRQ1)	全部模式		Hi-Z	Keep-O (注2)		Keep	Keep	Hi-Z
PH3	全部模式		Hi-Z	Keep-O		Keep	Keep	Hi-Z
PJ1、PJ3	全部模式		Hi-Z	Keep-O		Keep	Keep	Hi-Z

H : High 电平

L : Low 电平

Keep-O : 当用作输出引脚时, 保持之前的值; 当用作输入引脚时, 为高阻抗。

Keep : 保持软件待机中的引脚状态。

Hi-Z : 高阻抗

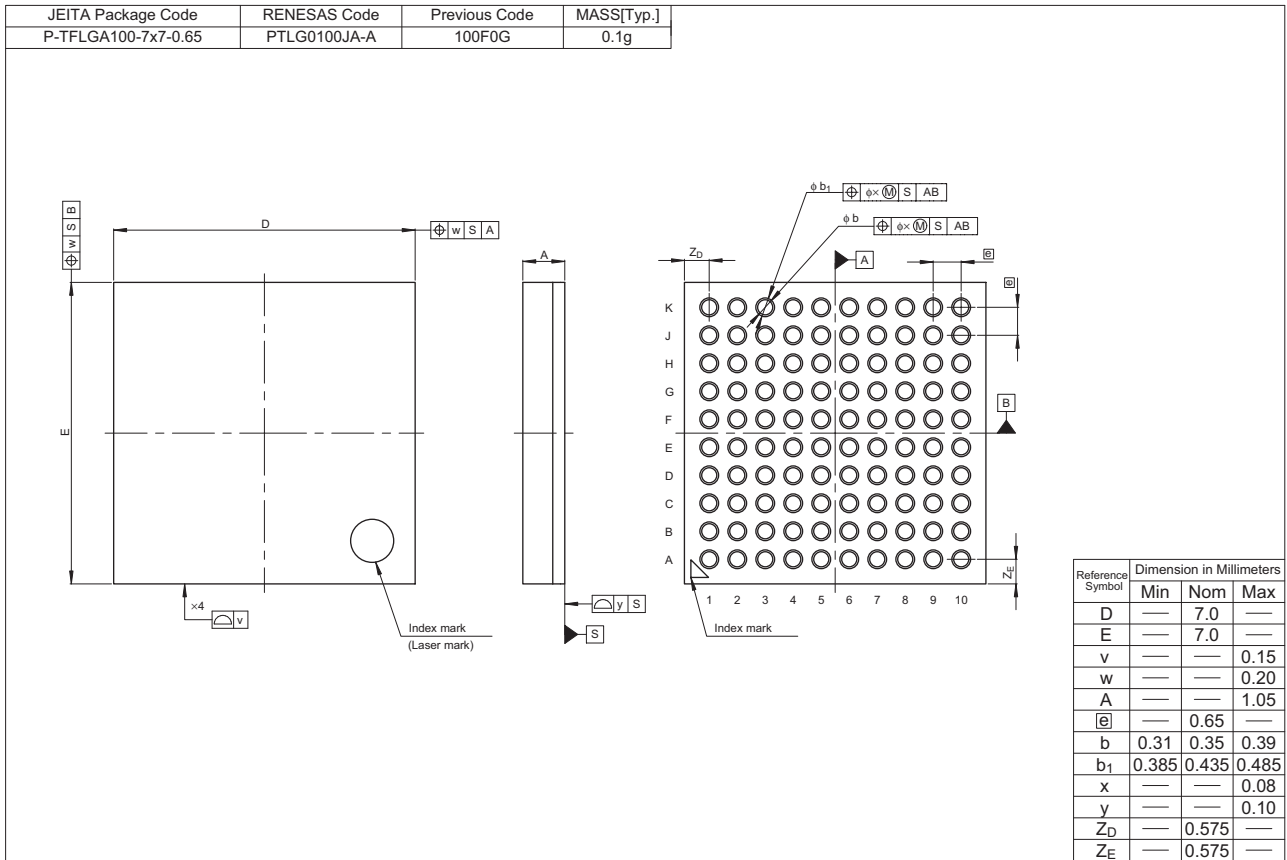
注 1. 在将 DPSBYCR.IOKEEP 位置“0”前, 保持 I/O 端口的状态。

注 2. 当用作外部中断引脚时, 能在设定为软件待机解除请求的情况下进行输入。

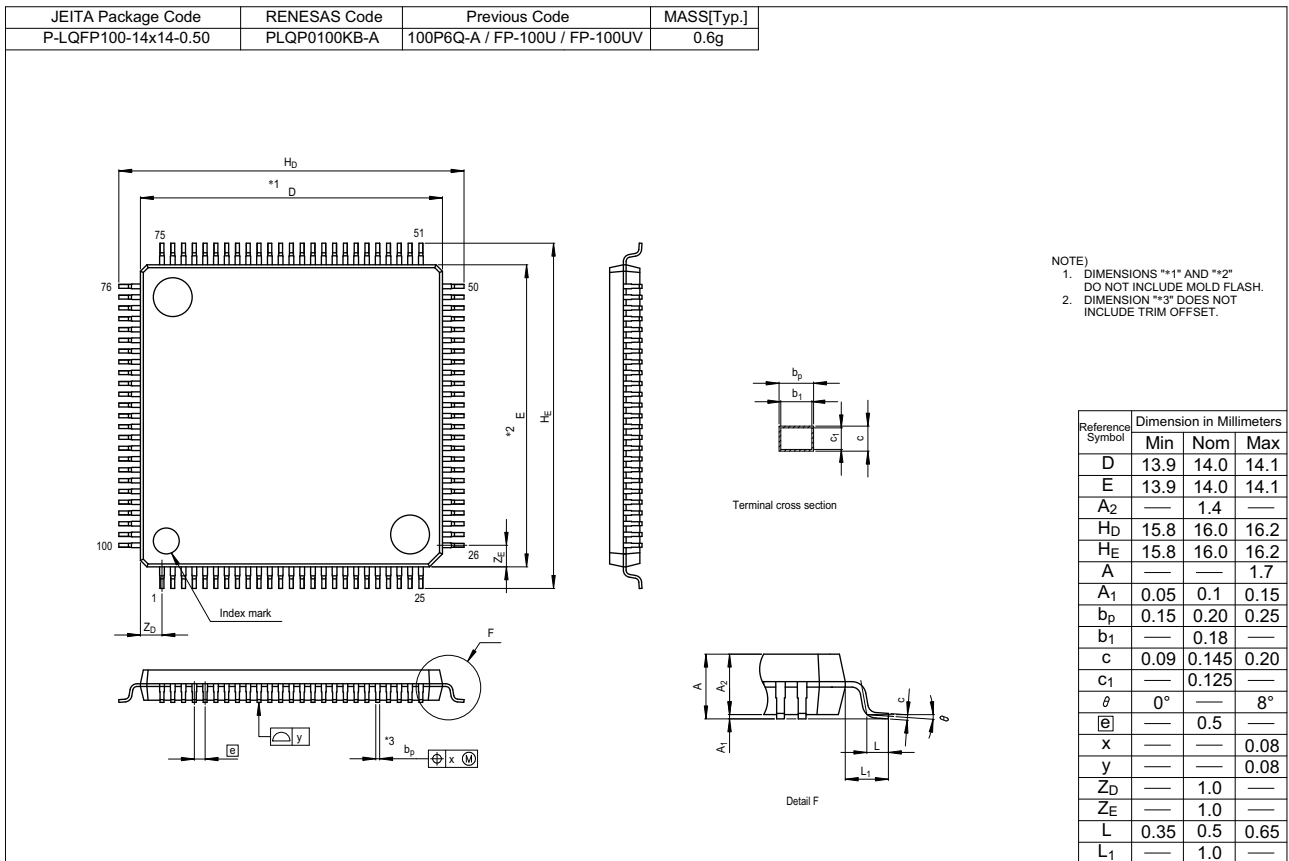
注 3. 能在设定为深度软件待机解除请求的情况下进行输入。

附录 2. 封装尺寸图

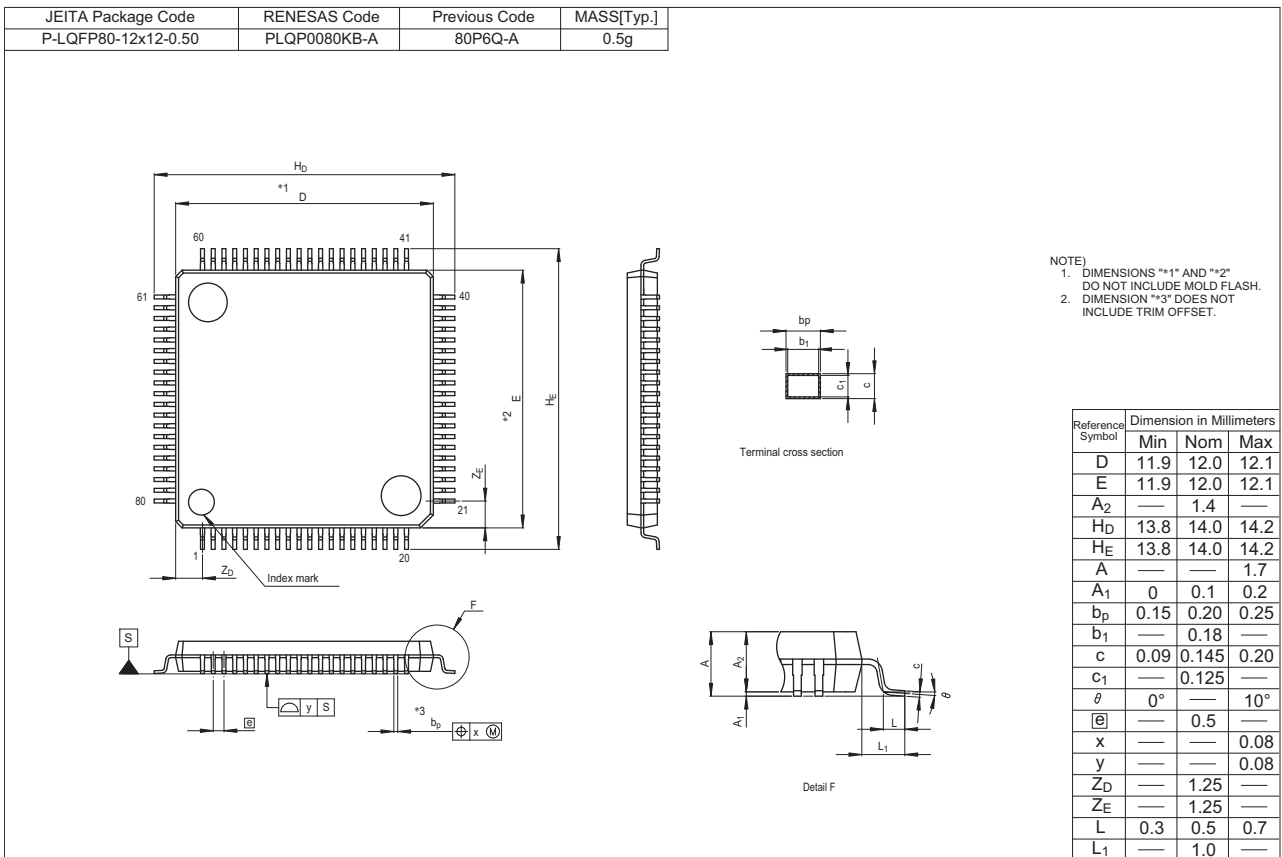
有关封装尺寸图的最新版和安装的信息，刊登在瑞萨电子的主页的“封装”中。



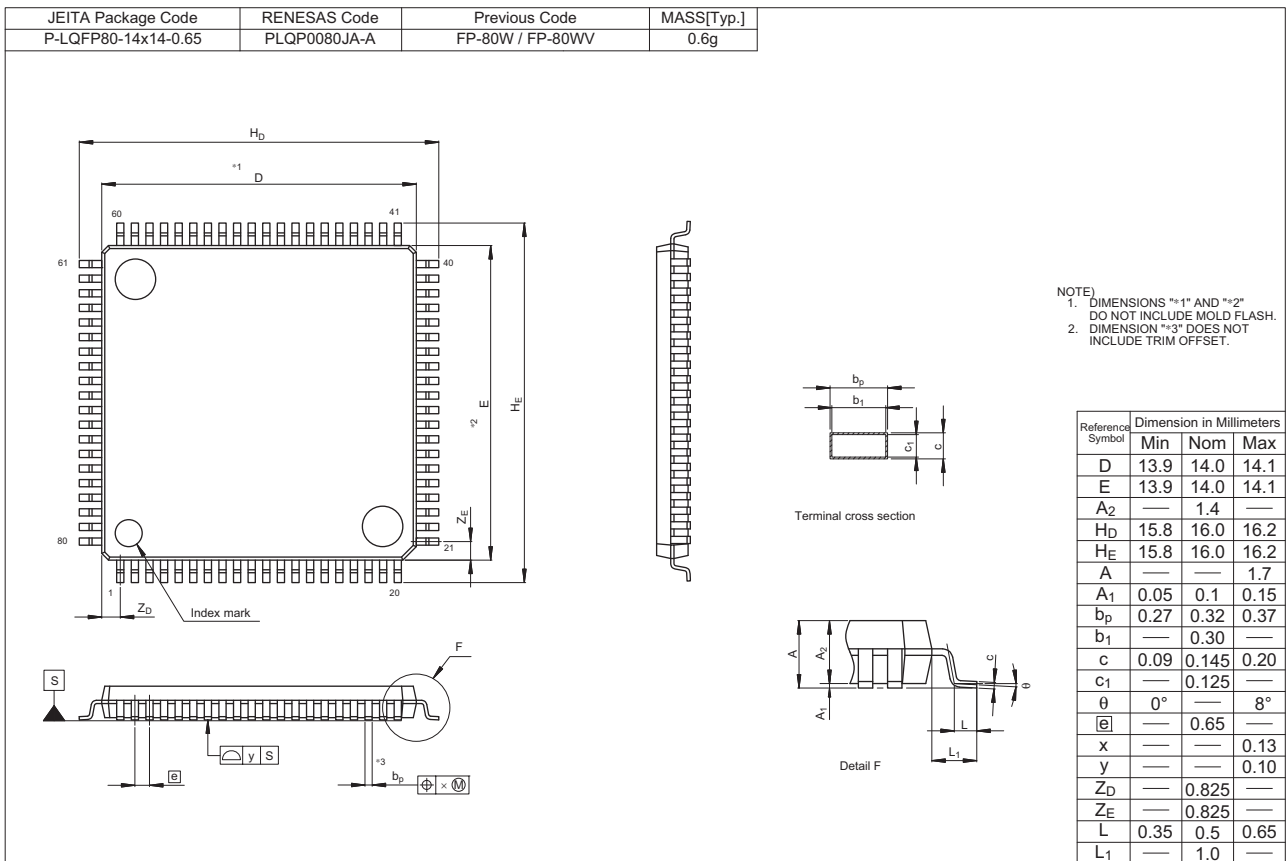
附图 2.1 100 引脚 TFLGA (PTLG0100JA-A)



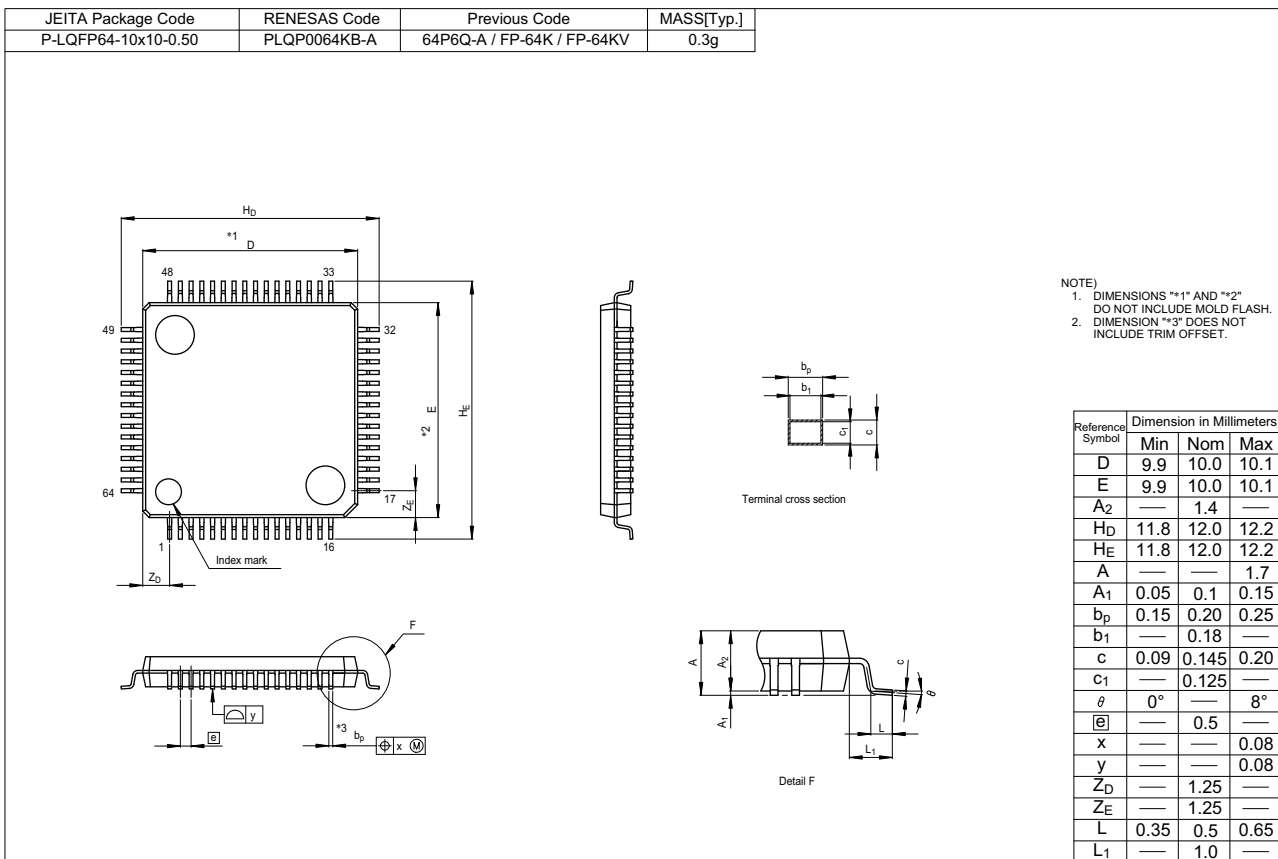
附图 2.2 100 引脚 LQFP (PLQP0100KB-A)



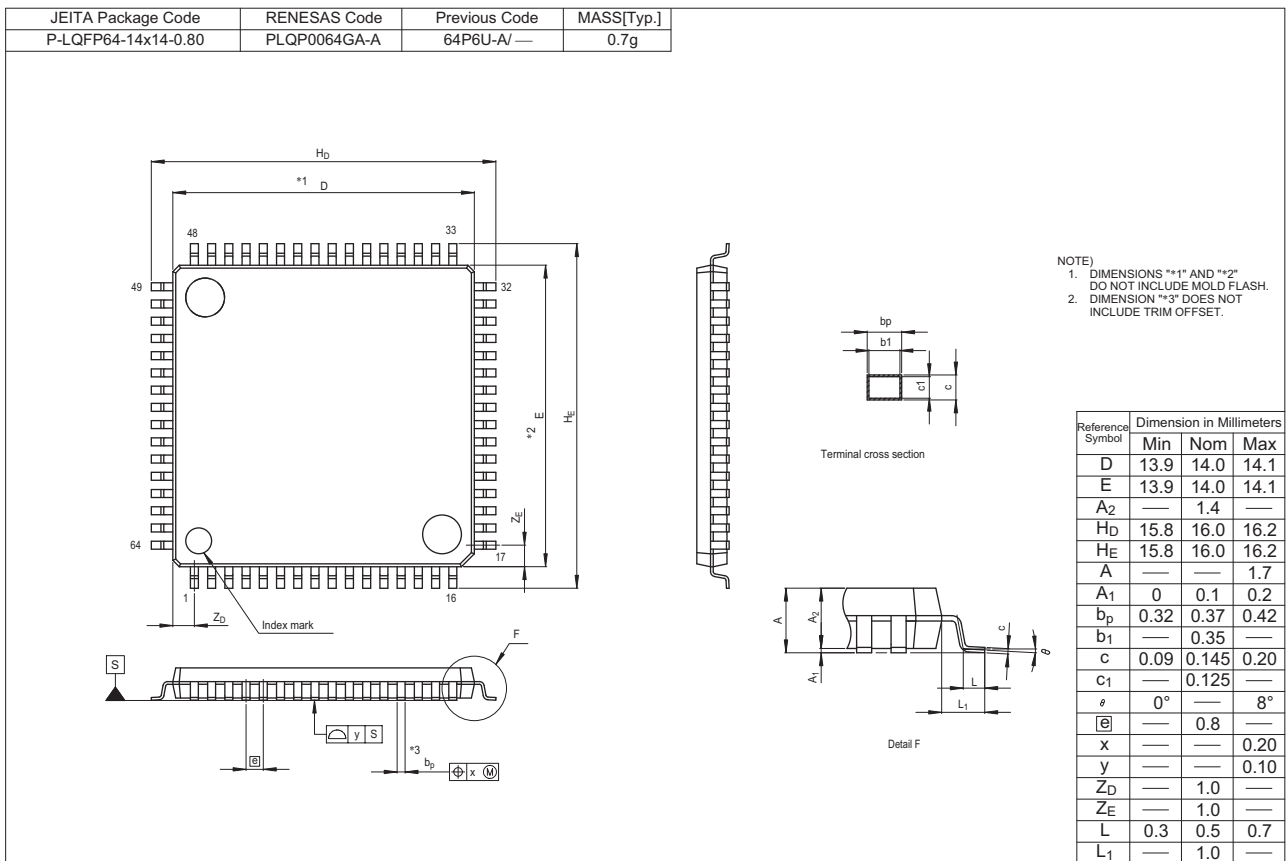
附图 2.3 80 引脚 LQFP (PLQP0080KB-A)



附图 2.4 80 引脚 LQFP (PLQP0080JA-A)



附图 2.5 64 引脚 LQFP (PLQP0064KB-A)



附图 2.6 64 引脚 LQFP (PLQP0064GA-A)

修订记录	RX210 群 用户手册 硬件篇
------	------------------

Rev.	发行日	修订内容	
		页	修订处
1.00	2012.03.09	—	初版发行

RX210 群
用户手册 硬件篇

Publication Date: Rev1.00 Mar 09, 2012

Published by: Renesas Electronics Corporation

**SALES OFFICES****Renesas Electronics Corporation**<http://www.renesas.com>Refer to "<http://www.renesas.com/>" for the latest and detailed information.**Renesas Electronics America Inc.**2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130**Renesas Electronics Canada Limited**1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220**Renesas Electronics Europe Limited**Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900**Renesas Electronics Europe GmbH**Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327**Renesas Electronics (China) Co., Ltd.**7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679**Renesas Electronics (Shanghai) Co., Ltd.**Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898**Renesas Electronics Hong Kong Limited**Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044**Renesas Electronics Taiwan Co., Ltd.**13F, No. 363, Fu Shing North Road, Taipei, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670**Renesas Electronics Singapore Pte. Ltd.**1 harbourFront Avenue, #06-10, Keppel Bay Tower, Singapore 098632
Tel: +65-6213-0200, Fax: +65-6278-8001**Renesas Electronics Malaysia Sdn.Bhd.**Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510**Renesas Electronics Korea Co., Ltd.**11F., Samik Lavied' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

RX210群



瑞萨电子株式会社

R01UH0037CJ0100