

RL78/I1A

用户手册 硬件篇

16 位 单片机

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. 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.
2. 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.
3. 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.
4. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.
"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 etc.
"High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.

Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (nuclear reactor control systems, military equipment etc.). 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 for which it is not intended. 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 which the product is not intended by Renesas Electronics.
6. 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.
7. 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 systems manufactured by you.
8. 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.
9. 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. You should not use Renesas Electronics products or 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. When exporting the Renesas Electronics 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.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
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.

注意事项

1. 本文档中所记载的关于电路、软件和其他相关信息仅用于说明半导体产品的操作和应用实例。用户如在设备设计中应用本文档中的电路、软件和相关信息，请自行负责。对于用户或第三方因使用上述电路、软件或信息而遭受的任何损失，瑞萨电子不承担任何责任。
2. 在准备本文档所记载的信息的过程中，瑞萨电子已尽量做到合理注意，但是，瑞萨电子并不保证这些信息都是准确无误的。用户因本文档中所记载的信息的错误或遗漏而遭受的任何损失，瑞萨电子不承担任何责任。
3. 对于因使用本文档中的瑞萨电子产品或技术信息而造成的侵权行为或因此而侵犯第三方的专利、版权或其他知识产权的行为，瑞萨电子不承担任何责任。本文档所记载的内容不应视为对瑞萨电子或其他人所有的专利、版权或其他知识产权作出任何明示、默示或其它方式的许可及授权。
4. 用户不得更改、修改、复制或或以其他方式部分或全部地非法使用瑞萨电子的任何产品。对于用户或第三方因上述更改、修改、复制或以其他方式非法使用瑞萨电子产品的行为而遭受的任何损失，瑞萨电子不承担任何责任。
5. 瑞萨电子产品根据其质量等级分为两个等级：“标准等级”和“高质量等级”。每种瑞萨电子产品的推荐用途均取决于产品的质量等级，如下所示：
标准等级： 计算机、办公设备、通讯设备、测试和测量设备、视听设备、家用电器、机械工具、个人电子设备以及工业机器人等。
高质量等级： 运输设备（汽车、火车、轮船等）、交通控制系统、防灾系统、预防犯罪系统以及安全设备等。
瑞萨电子产品无意用于且未被授权用于可能对人类生命造成直接威胁的产品或系统及可能造成人身伤害的产品或系统（人工生命维持装置或系统、植埋于体内的装置等）中，或者可能造成重大财产损失的产品或系统（核反应堆控制系统、军用设备等）中。在将每种瑞萨电子产品用于某种特定应用之前，用户应先确认其质量等级。不得将瑞萨电子产品用于超出其设计用途之外的任何应用。对于用户或第三方因将瑞萨电子产品用于其设计用途之外而遭受的任何损害或损失，瑞萨电子不承担任何责任。
6. 使用本文档中记载的瑞萨电子产品时，应在瑞萨电子指定的范围内，特别是在最大额定值、电源工作电压范围、移动电源电压范围、热辐射特性、安装条件以及其他产品特性的范围内使用。对于在上述指定范围之外使用瑞萨电子产品而产生的故障或损失，瑞萨电子不承担任何责任。
7. 虽然瑞萨电子一直致力于提高瑞萨电子产品的质量和可靠性，但是，半导体产品有其自身的具体特性，如一定的故障发生率以及在某些使用条件下会发生故障等。此外，瑞萨电子产品均未进行防辐射设计。所以请采取安全保护措施，以避免当瑞萨电子产品在发生故障而造成火灾时导致人身事故、伤害或损害的事故。例如进行软硬件安全设计（包括但不限于冗余设计、防火控制以及故障预防等）、适当的老化处理或其他适当的措施等。由于难于对微机软件单独进行评估，所以请用户自行对最终产品或系统进行安全评估。
8. 关于环境保护方面的详细内容，例如每种瑞萨电子产品的环境兼容性等，请与瑞萨电子的营业部门联系。使用瑞萨电子产品时，请遵守对管制物质的使用或含量进行管理的所有相应法律法规（包括但不限于《欧盟 RoHS 指令》）。对于因用户未遵守相应法律法规而导致的损害或损失，瑞萨电子不承担任何责任。
9. 不可将瑞萨电子产品和技术用于或者嵌入日本国内或海外相应的法律法规所禁止生产、使用及销售的任何产品或系统中。也不可将本文档中记载的瑞萨电子产品或技术用于与军事应用或者军事用途有关的任何目的（如大规模杀伤性武器的开发等）。在将本文档中记载的瑞萨电子产品或技术进行出口时，应当遵守相应的出口管制法律法规，并按照上述法律法规所规定的程序进行。
10. 向第三方分销或处分产品或以其他方式将产品置于第三方控制之下的瑞萨电子产品买方或分销商，有责任事先向上述第三方通知本文档规定的内容和条件；对于用户或第三方因非法使用瑞萨电子产品而遭受的任何损失，瑞萨电子不承担任何责任。
11. 在事先未得到瑞萨电子书面认可的情况下，不得以任何形式部分或全部转载或复制本文档。
12. 如果对本文档所记载的信息或瑞萨电子产品有任何疑问，或者用户有任何其他疑问，请向瑞萨电子的营业部门咨询。
(注 1) 瑞萨电子：在本文档中指瑞萨电子株式会社及其控股子公司。
(注 2) 瑞萨电子产品：指瑞萨电子开发或生产的任何产品。

关于 CMOS 器件的注意事项

① 输入引脚处的施加电压波形

输入噪声或由反射波引起的波形失真可能导致故障发生。如果由于噪声等影响，使CMOS器件的输入电压范围处于在 V_{IL} (MAX) 和 V_{IH} (MIN) 之间，器件可能发生故障。在输入电平固定时以及输入电平从 V_{IL} (MAX) 到 V_{IH} (MIN) 的过渡期间，要谨防颤振噪声进入器件。

② 未使用的输入引脚的处理

CMOS器件上未连接的输入端可能是故障源。如果一个输入引脚未被连接，则由于噪声等原因可能会产生内部输入电平，从而导致故障。CMOS器件的工作方式与双极性或NMOS器件不同。CMOS器件的输入电平必须借助上拉或下拉电路固定于高电平或低电平。每一个未使用引脚只要有可能成为输出引脚时，都应该通过附加电阻连接到 V_{DD} 或 GND 。对未使用引脚的处理因器件而异，必须遵循与器件相关的规格和说明。

③ ESD防护措施

如果MOS器件周围有强电场，将会击穿氧化栅极，降低器件的工作性能。因此必须采取措施，尽可能防止静电产生。一旦有静电，必须立即释放。环境必须控制适当。如果空气干燥，应当使用加湿器。建议避免使用容易产生静电的绝缘体。半导体器件的存放和运输必须使用抗静电容器、静电屏蔽袋或导电材料包装。所有包括工作台和工作面的测试和测量工具必须良好接地。操作员应当佩戴手腕带以保证良好接地。不能用手直接接触半导体器件。对于装配有半导体器件的PW板也应采取类似的静电防范措施。

④ 初始化之前的状态

上电并不一定定义MOS器件的初始状态。刚接通电源时，具有复位功能的MOS器件并没有被初始化。因此上电不能保证输出引脚的电平、I/O设置和寄存器的内容。器件在收到复位信号后才进行初始化。具有复位功能的器件在上电后必须立即进行复位操作。

⑤ 电源上电/断电序列

器件内部工作和外部接口使用不同电源的情况下，原则上应先接通内部电源之后再接通外部电源。当关闭电源时，原则上先关闭外部电源再关闭内部电源。如果电源开关顺序相反，可能会对器件的内部元件施加电压，从而由于异常电流的流过而造成故障和降低元件的性能。须视具体器件和支配器件的相关规格来单独决定正确的上电/断电序列。

⑥ 断电状态期间的信号输入

不要在器件断电时输入信号或I/O上拉电源。因为输入信号或提供I/O上拉电源将引起电流注入，从而引起器件的误操作，并且此时流过器件的异常电流引起内部元件性能劣化。须视具体器件和支配器件的相关规格来单独决定断电状态期间的信号输入。

本手册的使用方法

读者

本手册适用于那些希望了解 **RL78/I1A** 的功能，并设计开发用于这些设备的应用系统的用户工程师。

目标产品如下：

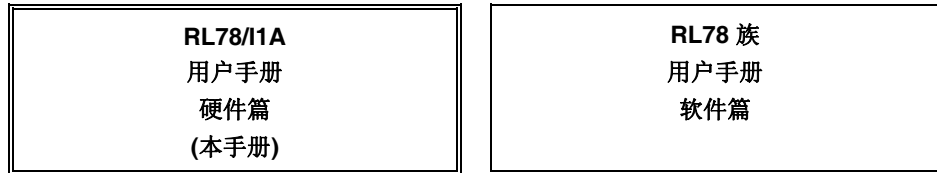
- 20 引脚:R5F1076C
- 30 引脚:R5F107AE, R5F107AC
- 32 引脚:R5F107BC
- 38 引脚:R5F107DE

目的

本手册适用于帮助用户了解下述**结构**中所述的功能。

结构

本手册分为两个部分：本手册和软件编辑 (**RL78** 微控制器共通)。



- | | |
|---|---|
| <ul style="list-style-type: none">• 引脚功能• 内部模块功能• 中断• 其他片上外设功能• 电特性 | <ul style="list-style-type: none">• CPU 功能• 指令组• 各指令的说明 |
|---|---|

如何阅读本手册

假定本手册的读者具备电气工程、逻辑电路以及微控制器领域的常识。

- 寄存器格式的说明：
→ 关于尖括 (<>) 内的位数，其名称在汇编程序中定义为保留字，并在编译器中使用 `#pragma sfr` 指令定义为 `sfr` 变量。
- 要了解 **RL78** 微控制器指令的详细情况：
→ 请参阅单独文件 **RL78 族 用户手册 软件篇(R01US0015E)**。

惯例

数据有效位： 高位在左，低位在右 低电平有效表示
有效低电平表示： $\overline{\text{xxx}}$ (在引脚或信号名称上有上划线)
注： 正文中使用**注**标记的脚注
注意事项： 需特别注意的信息
备注： 补充信息
数字表示法： 二进制 ...xxxx 或 xxxx**B**
 十进制 ...xxxx
 十六进制 ...xxxx**H**

相关文档

本手册中的相关文档可能包括初稿版本。但是，初稿版本并未特别标注。

器件相关文档

文档名称	文档编号
RL78/I1A 用户手册 硬件篇	本手册
RL78 微控制器 用户手册 软件篇	R01US0015E

闪存编程器相关文档

文档名称	文档编号
PG-FP5 闪存编程器 用户手册	R20UT0008E

注意事项 上述列出的文件随时可能被更改，恕不另行通知。设计时，请确保使用各个文件的最新版本。

其他文件

文件名称	文件编号
瑞萨微型计算机整体目录	R01CS0001E
半导体封装安装手册	注
预防静电放电(ESD)损坏半导体器件	C11892E

注 请参阅“半导体封装安装手册”网站
(<http://www.renesas.com/products/package/manual/index.jsp>)

注意事项 上述列出的文件随时可能被更改，恕不另行通知。设计时，请确保使用各个文件的最新版本。

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

第一章 概述	1
1.1 特点	1
1.2 订购信息	2
1.3 引脚配置(顶视图)	4
1.3.1 20 引脚产品.....	4
1.3.2 30 引脚产品.....	5
1.3.3 32 引脚产品.....	6
1.3.4 38 引脚产品.....	7
1.4 引脚名称	8
1.5 框图	9
1.5.1 20 引脚产品.....	9
1.5.2 30 引脚产品.....	10
1.5.3 32 引脚产品.....	11
1.5.4 38 引脚产品.....	12
1.6 功能概述	13
第二章 引脚功能	16
2.1 端口功能	16
2.1.1 20 引脚产品.....	17
2.1.2 30 引脚产品.....	18
2.1.3 32 引脚产品.....	20
2.1.4 38 引脚产品.....	22
2.2 端口以外的引脚功能	24
2.2.1 各种产品的配置功能.....	24
2.3 引脚的输入/输出电路和未使用引脚的推荐连接	27
第三章 CPU 架构	32
3.1 存储空间	32
3.1.1 内部程序存储空间	37
3.1.2 镜像区域.....	40
3.1.3 内部数据存储空间	42
3.1.4 特殊功能寄存器 (SFR) 区域.....	43
3.1.5 扩展特殊功能寄存器 (2nd SFR: 第二特殊功能寄存器) 区域.....	43
3.1.6 数据存储寻址	44
3.2 处理器寄存器	47
3.2.1 控制寄存器.....	47
3.2.2 通用寄存器.....	49

3.2.3 ES 和 CS 寄存器.....	51
3.2.4 特殊功能寄存器(SFR).....	52
3.2.5 扩展特殊功能寄存器 (2nd SFR: 第二特殊功能寄存器).....	57
3.3 指令地址寻址.....	70
3.3.1 相对寻址.....	70
3.3.2 立即寻址.....	70
3.3.3 表间接寻址.....	71
3.3.4 寄存器直接寻址.....	72
3.4 处理数据地址的寻址.....	73
3.4.1 隐含寻址.....	73
3.4.2 寄存器寻址.....	73
3.4.3 直接寻址.....	74
3.4.4 短直接寻址.....	75
3.4.5 SFR 寻址.....	76
3.4.6 寄存器间接寻址.....	77
3.4.7 基址寻址.....	78
3.4.8 基址变址寻址.....	81
3.4.9 堆栈寻址.....	82
第四章 端口功能.....	83
4.1 端口功能.....	83
4.2 端口配置.....	83
4.2.1 端口 0.....	84
4.2.2 端口 1.....	89
4.2.3 端口 2.....	95
4.2.4 端口 3.....	98
4.2.5 端口 4.....	100
4.2.6 端口 7.....	102
4.2.7 端口 12.....	104
4.2.8 端口 13.....	108
4.2.9 端口 14.....	109
4.2.10 端口 20.....	111
4.3 控制端口功能的寄存器.....	119
4.4 端口功能的操作.....	131
4.4.1 输入/输出端口的写入.....	131
4.4.2 输入/输出端口的读取.....	131
4.4.3 操作输入/输出端口.....	131
4.4.4 与不同电位的外部器件(2.5 V、3 V)的连接方法.....	132
4.5 使用复用功能时端口模式寄存器和输出锁存器的设置.....	133
4.6 使用端口功能时的注意事项.....	136
4.6.1 端口寄存器 n (Pn)的位操作指令的相关注意事项.....	136

4.6.2 指定引脚设置时的注意事项	137
第五章 时钟发生电路	138
5.1 时钟发生电路的功能	139
5.2 时钟发生电路的配置	141
5.3 控制时钟发生电路的寄存器	143
5.4 系统时钟振荡电路	158
5.4.1 X1 振荡电路	158
5.4.2 XT1 振荡电路	158
5.4.3 高速片上振荡器	162
5.4.4 低速片上振荡器	162
5.4.5 PLL (锁相环)	162
5.5 时钟发生电路的操作	163
5.6 控制时钟	165
5.6.1 高速片上振荡器的设置示例	165
5.6.2 X1 振荡电路的设置示例	166
5.6.3 XT1 振荡电路的设置示例	167
5.6.4 PLL 电路的设置示例	168
5.6.5 CPU 时钟状态转换图	169
5.6.6 CPU 时钟转换前的条件以及转换后的处理	176
5.6.7 切换 CPU 时钟和切换系统时钟所需要的时间	178
5.6.8 停止时钟振荡前的条件	179
第六章 定时器阵列单元	180
6.1 定时器阵列单元功能	181
6.1.1 单通道操作功能	181
6.1.2 多通道联动操作功能	182
6.1.3 支持 LIN-bus 功能 (仅限通道 7)	184
6.1.4 支持 DMX512 功能 (仅限通道 7)	185
6.2 定时器阵列单元的配置	186
6.3 控制定时器阵列单元的寄存器	192
6.4 定时器阵列单元的基本规则	215
6.4.1 多通道联动操作功能的基本规则	215
6.4.2 8 位定时器操作功能的基本规则 (仅限通道 1 和 3)	217
6.5 计数器的操作	218
6.5.1 计数时钟 (fTCLK)	218
6.5.2 计数器的开始时序	220
6.5.3 计数器的操作	221
6.6 通道输出 (TO0n 引脚) 控制	226
6.6.1 TO0n 引脚输出电路配置	226
6.6.2 TO0n 引脚输出设置	227

6.6.3 通道输出操作的注意事项	228
6.6.4 TO0n 位的集中操作	233
6.6.5 计数操作开始时的定时器中断和 TO0n 引脚输出	234
6.7 定时器阵列单元的单通道操作功能	235
6.7.1 作为间隔定时器/方波输出的操作	235
6.7.2 作为外部事件计数器操作	241
6.7.3 作为输入脉冲间隔测量的操作	246
6.7.4 输入信号高/低电平宽度测量的操作	250
6.7.5 作为延迟计数器的操作	254
6.8 定时器阵列单元的多通道联动操作功能	259
6.8.1 作为单触发脉冲输出功能操作	259
6.8.2 作为 PWM 功能的操作	266
6.8.3 作为多重 PWM 输出功能的操作	273
第七章 16 位定时器 KB0、KB1 和 KB2	281
7.1 16 位定时器 KB0、KB1 和 KB2 的功能	281
7.2 16 位定时器 KB0、KB1 和 KB2 的配置	283
7.3 控制 16 位定时器 KB0、KB1 和 KB2 的寄存器	286
7.4 16 位定时器 KB0、KB1 和 KB2 的操作	303
7.4.1 计数器基本操作	306
7.4.2 默认电平和有效电平	306
7.4.3 停止/重启操作	310
7.4.4 批量覆盖操作	313
7.4.5 单机模式 (通过 TKBCRn0 的周期控制)	314
7.4.6 单机模式 (通过外部触发输入的周期控制)	320
7.4.7 同时开始/停止模式	327
7.4.8 同步开始/清除模式	336
7.4.9 交错式 PFC (Power Factor Correction) 输出模式	344
7.5 16 位定时器 KB0、KB1 和 KB2 的选项功能	357
7.5.1 A/D 转换开始时序信号输出功能	358
7.5.2 PWM 输出抖动功能	360
7.5.3 PWM 输出软启动功能	364
7.5.4 PWM 输出门控功能 (不兼用 PWM 输出软启动功能)	367
7.5.5 PWM 输出门控功能 (可兼用 PWM 输出软启动功能)	369
7.5.6 最大频率限值功能	371
7.6 强制输出停止功能	373
7.6.1 强制输出停止功能 1 和强制输出停止功能 2	374
7.6.2 强制输出停止功能的配置	377
7.6.3 控制强制输出停止功能的寄存器	378
7.7 强制输出停止功能 1 的操作	388
7.7.1 强制输出停止功能 1 的概要	388

7.7.2 强制输出停止功能 1 的软件清除操作.....	389
7.7.3 强制输出停止功能 1 的基本操作.....	390
7.8 强制输出停止功能 2 的操作.....	394
7.8.1 强制输出停止功能 2 的概要.....	394
7.8.2 强制输出停止功能 2 的基本操作.....	395
第八章 16 位定时器 KC0.....	397
8.1 16 位定时器 KC0 的功能.....	397
8.2 16 位定时器 KC0 的配置.....	397
8.3 控制 16 位定时器 KC0 的寄存器.....	400
8.4 16 位定时器 KC0 的操作.....	409
8.4.1 PWM 输出功能.....	411
8.4.2 停止/重启操作.....	414
8.4.3 默认电平和有效电平.....	416
8.4.4 同时开始&停止模式.....	418
第九章 实时时钟.....	419
9.1 实时时钟的功能.....	419
9.2 实时时钟的配置.....	419
9.3 控制实时时钟的寄存器.....	421
9.4 实时时钟的操作.....	436
9.4.1 开始实时时钟的操作.....	436
9.4.2 开始操作后切换至 HALT/STOP 模式.....	437
9.4.3 读/写实时时钟.....	438
9.4.4 设置实时时钟的报警.....	440
9.4.5 实时时钟的 1 Hz 输出.....	441
9.4.6 实时时钟的钟表误差修正示例.....	442
第十章 12 位间隔定时器.....	445
10.1 12 位间隔定时器的功能.....	445
10.2 12 位间隔定时器的配置.....	445
10.3 控制 12 位间隔定时器的寄存器.....	446
10.4 12 位间隔定时器的操作.....	449
第十一章 看门狗定时器.....	450
11.1 看门狗定时器的功能.....	450
11.2 看门狗定时器的配置.....	451
11.3 控制看门狗定时器的寄存器.....	452
11.4 看门狗定时器的操作.....	453
11.4.1 控制看门狗定时器的操作.....	453

11.4.2 设置看门狗定时器的溢出时间	454
11.4.3 设置看门狗定时器的窗口开启期间.....	455
11.4.4 设置看门狗定时器的间隔中断	456
第十二章 A/D 转换器	457
12.1 A/D 转换器的功能.....	457
12.2 A/D 转换器的配置.....	459
12.3 A/D 转换器中使用的寄存器	461
12.4 A/D 转换器的转换操作.....	484
12.5 输入电压和转换结果	486
12.6 A/D 转换器的操作模式.....	487
12.6.1 软件触发模式（选择模式，连续转换模式）	487
12.6.2 软件触发模式（选择模式，单次转换模式）	488
12.6.3 软件触发模式（扫描模式，连续转换模式）	489
12.6.4 软件触发模式（扫描模式，单次转换模式）	490
12.6.5 硬件触发无等待模式（选择模式，连续转换模式）	491
12.6.6 硬件触发无等待模式（选择模式，单次转换模式）	492
12.6.7 硬件触发无等待模式（扫描模式，连续转换模式）	493
12.6.8 硬件触发无等待模式（扫描模式，单次转换模式）	494
12.6.9 硬件触发等待模式（选择模式，连续转换模式）	495
12.6.10 硬件触发等待模式（选择模式，单次转换模式）	496
12.6.11 硬件触发等待模式（扫描模式，连续转换模式）	497
12.6.12 硬件触发等待模式（扫描模式，单次转换模式）	498
12.7 A/D 转换器的设置流程图	499
12.7.1 设置软件触发模式.....	500
12.7.2 设置硬件触发无等待模式	501
12.7.3 设置硬件触发等待模式	502
12.7.4 使用温度传感器时的设置 (软件触发模式和单次转换模式下的示例)	503
12.7.5 设置测试模式.....	504
12.8 SNOOZE 模式功能	505
12.9 如何阅读 A/D 转换器特性表.....	508
12.10 A/D 转换器的注意事项.....	510
第十三章 可编程增益放大器	514
13.1 可编程增益放大器的功能.....	514
13.2 可编程增益放大器的配置.....	514
13.3 可编程增益放大器中使用的寄存器	515
13.4 可编程增益放大器的操作	523
13.5 可编程增益放大器的设置步骤.....	524
第十四章 比较器.....	525

14.1 比较器的功能	525
14.2 比较器的配置	527
14.3 控制比较器的寄存器	527
14.4 比较器的设置步骤	541
第十五章 串行阵列单元 0	544
15.1 串行阵列单元 0 的功能	545
15.1.1 3 线串行输入/输出 (CSI00)	545
15.1.2 UART (UART0, UART1)	546
15.2 串行阵列单元 0 的配置	547
15.3 控制串行阵列单元 0 的寄存器	552
15.4 操作停止模式	578
15.4.1 以单元单位停止操作	578
15.4.2 以通道单位停止操作时	579
15.5 3 线串行输入/输出 (CSI00) 通信的操作	580
15.5.1 主发送	582
15.5.2 主接收	591
15.5.3 主发送/接收	600
15.5.4 从发送	610
15.5.5 从接收	620
15.5.6 从发送/接收	627
15.5.7 SNOOZE 模式功能 (仅限 CSI00)	637
15.5.8 计算传送时钟频率	641
15.5.9 3 线串行输入/输出 (CSI00)通信过程中发生错误时的处理步骤	643
15.6 UART(UART0、UART1)通信的操作	644
15.6.1 UART 发送	646
15.6.2 UART 接收	656
15.6.3 SNOOZE 模式功能 (仅限 UART0 接收)	663
15.6.4 计算波特率	669
15.6.5 UART(UART0, UART1) 通信过程中发生错误时的处理步骤	673
15.7 LIN 通信操作	674
15.7.1 LIN 发送	674
15.7.2 LIN 接收	677
15.8 DMX512 通信操作	682
第十六章 串行阵列单元 4 (DALI/UART4)	683
16.1 串行阵列单元 4 (DALI/UART4)的功能	683
16.2 串行阵列单元 4 (DALI/UART4)的配置	685
16.3 控制串行阵列单元 4 (DALI/UART4)的寄存器	690
16.4 操作停止模式	713
16.4.1 以单元单位停止操作	713

16.4.2 以通道单位停止操作时	714
16.5 UART 通信的操作.....	715
16.5.1 UART 发送	715
16.5.2 UART 接收	725
16.6 DALI 模式	732
16.6.1 DALI 发送	735
16.6.2 DALI 接收	743
16.7 SNOOZE 模式功能 (仅限 DALI/UART4 接收).....	750
16.8 单线数据模式	756
16.9 计算波特率.....	757
16.10 DALI/UART4 通信时的发生错误时的处理步骤	762
16.11 DMX512 通信操作	763
第十七章 串行接口 IICA.....	768
17.1 串行接口 IICA 的功能	768
17.2 串行接口 IICA 的配置	771
17.3 控制串行接口 IICA 的寄存器.....	774
17.4 I²C 总线模式功能	788
17.4.1 引脚配置.....	788
17.4.2 利用 IICWLO 和 IICWHO 寄存器设置传送时钟.....	789
17.5 I²C 总线定义和控制方法	790
17.5.1 开始条件.....	790
17.5.2 地址.....	791
17.5.3 指定传送方向.....	791
17.5.4 应答(ACK)	792
17.5.5 停止条件.....	793
17.5.6 等待.....	794
17.5.7 取消等待.....	796
17.5.8 中断请求(INTIICA0) 的产生时序和等待控制.....	797
17.5.9 地址匹配检测方法	798
17.5.10 错误检测.....	798
17.5.11 扩展代码.....	798
17.5.12 仲裁	798
17.5.13 唤醒功能.....	801
17.5.14 通信预约.....	804
17.5.15 注意事项.....	808
17.5.16 通信操作.....	809
17.5.17 I ² C 中断请求(INTIICA0)的发生时序.....	816
17.6 时序图	837
第十八章 乘除法器 and 乘加器	852

18.1	乘除法器 and 乘加器的功能.....	852
18.2	乘除法器 and 乘加器的配置.....	852
18.3	控制乘除法器 and 乘加器的寄存器.....	858
18.4	乘除法器 and 乘加器的操作.....	860
18.4.1	乘法(无符号)操作.....	860
18.4.2	乘法(有符号)操作.....	861
18.4.3	乘法累加(无符号)操作.....	862
18.4.4	乘法累加(有符号)操作.....	864
18.4.5	除法操作.....	866
第十九章 DMA 控制器.....		868
19.1	DMA 控制器的功能.....	868
19.2	DMA 控制器的配置.....	869
19.3	控制 DMA 控制器的寄存器.....	872
19.4	DMA 控制器的操作.....	876
19.4.1	操作步骤.....	876
19.4.2	传送模式.....	877
19.4.3	结束 DMA 传送.....	877
19.5	DMA 控制器的设置示例.....	878
19.5.1	CSI 连续发送.....	878
19.5.2	A/D 转换结果的连续读入.....	880
19.5.3	UART 连续接收 + ACK 发送.....	882
19.5.4	利用 DWAITn 位挂起 DMA 传送.....	884
19.5.5	利用软件的强制结束.....	885
19.6	使用 DMA 控制器时的注意事项.....	887
第二十章 中断功能.....		889
20.1	中断功能类型.....	889
20.2	中断源和配置.....	889
20.3	控制中断功能的寄存器.....	895
20.4	中断处理的操作.....	909
20.4.1	可屏蔽中断请求受理.....	909
20.4.2	软件中断请求受理.....	912
20.4.3	嵌套中断处理.....	912
20.4.4	保留中断请求.....	916
第二十一章 待机功能.....		917
21.1	待机功能和构成.....	917
21.1.1	待机功能.....	917
21.1.2	控制待机功能的寄存器.....	918

21.2 待机功能操作	921
21.2.1 HALT 模式	921
21.2.2 STOP 模式	928
21.2.3 SNOOZE 模式	934
第二十二章 复位功能	937
22.1 确认复位源的寄存器	948
第二十三章 上电复位电路	950
23.1 上电复位电路的功能	950
23.2 上电复位电路的配置	951
23.3 上电复位电路的操作	951
23.4 上电复位电路的注意事项	954
第二十四章 电压检测电路	956
24.1 电压检测电路的功能	956
24.2 电压检测电路的配置	957
24.3 控制电压检测电路的寄存器	957
24.4 电压检测电路的操作	962
24.4.1 使用复位模式时	962
24.4.2 使用中断模式时	964
24.4.3 使用中断&复位模式时	966
24.5 电压检测电路的注意事项	972
第二十五章 安全功能	974
25.1 安全功能的概述	974
25.2 安全功能使用的寄存器	975
25.3 安全功能的操作	975
25.3.1 闪存 CRC 运算功能(高速 CRC)	975
25.3.2 CRC 运算功能(通用 CRC)	978
25.3.3 RAM 奇偶校验错误检测功能	980
25.3.4 RAM 保护功能	981
25.3.5 SFR 保护功能	982
25.3.6 非法存储器存取检测功能	984
25.3.7 频率检测功能	987
25.3.8 A/D 测试功能	989
第二十六章 稳压器	991
26.1 稳压器的概述	991

第二十七章 选项字节	992
27.1 选项字节的功能	992
27.1.1 用户选项字节 (000C0H 至 000C2H/010C0H 至 010C2H).....	992
27.1.2 片上调试选项字节 (000C3H/ 010C3H).....	993
27.2 用户选项字节的格式	994
27.3 片上调试选项字节的格式	998
27.4 选项字节的设置	999
第二十八章 闪存.....	1000
28.1 使用闪存编程器写入闪存.....	1001
28.1.1 编程环境.....	1003
28.1.2 通信模式.....	1003
28.2 板上引脚连接	1005
28.2.1 P40/TOOL0 引脚	1005
28.2.2 RESET 引脚.....	1005
28.2.3 端口引脚.....	1006
28.2.4 REGC 引脚	1006
28.2.5 X1 和 X2 引脚	1006
28.2.6 电源.....	1006
28.3 数据闪存	1007
28.3.1 数据闪存概述.....	1007
28.3.2 控制数据闪存的寄存器	1008
28.3.3 存取数据闪存的步骤	1009
28.4 编程方法	1010
28.4.1 闪存的控制.....	1010
28.4.2 闪存的编程模式.....	1011
28.4.3 通信模式.....	1012
28.4.4 通信命令.....	1013
28.4.5 签名数据的说明	1014
28.5 安全设置	1015
28.6 通过自编程对闪存编程.....	1017
28.6.1 引导交换功能.....	1019
28.6.2 闪存屏蔽窗口功能.....	1021
第二十九章 On-chip 调试功能.....	1022
29.1 将 E1 On-chip 调试仿真器连接至 RL78/I1A.....	1022
29.2 片上调试安全 ID	1023
29.3 用户资源的预留	1023
第三十章 十进制调整(BCD)电路.....	1025

30.1 十进制调整电路的功能.....	1025
30.2 十进制调整电路使用的寄存器.....	1025
30.3 十进制调整电路的操作.....	1026
第三十一章 指令集.....	1028
31.1 操作列表使用规则.....	1029
31.1.1 操作标识符和标识方法.....	1029
31.1.2 操作栏的说明.....	1030
31.1.3 标志栏的说明.....	1031
31.1.4 PREFIX 指令.....	1031
31.2 操作列表.....	1032
第三十二章 电特性.....	1049
32.1 最大绝对额定值.....	1050
32.2 振荡电路特性.....	1052
32.2.1 X1, XT1 振荡电路特性.....	1052
32.2.2 片上振荡器特性.....	1053
32.2.3 PLL 特性.....	1053
32.3 DC 特性.....	1054
32.3.1 引脚特性.....	1054
32.3.2 电源电流特性.....	1059
32.4 AC 特性.....	1065
32.5 外围功能特性.....	1067
32.5.1 串行阵列单元 0, 4 (UART0, UART1, CSI00, DALI/UART4).....	1067
32.5.2 串行接口 IICA.....	1081
32.5.3 片上调试 (UART).....	1082
32.6 模拟特性.....	1082
32.6.1 A/D 转换器特性.....	1082
32.6.2 温度传感器特性.....	1085
32.6.3 可编程增益放大器.....	1086
32.6.4 比较器.....	1087
32.6.5 POR 电路特性.....	1087
32.6.6 LVD 电路特性.....	1088
32.7 数据存储器 STOP 模式低电源电压数据保持特性.....	1089
32.8 闪存编程特性.....	1089
32.9 切换闪存编程模式的时序规格.....	1090
第三十三章 封装尺寸图.....	1091
33.1 20 引脚产品.....	1091
33.2 30 引脚产品.....	1092

33.3 32 引脚产品	1093
33.4 38 引脚产品	1094
第三十四章 关于等待的注意事项	1095
34.1 关于等待的注意事项	1095
34.2 产生等待的外围硬件	1095
附录 A 修订记录.....	1097

第一章 概述

1.1 特点

- 操作时钟
 - 1 至 32 MHz (当使用高速片上振荡器时钟时)
 - 32.768 kHz (当使用副系统时钟时)
- 通用寄存器: 8 位×32 个寄存器(8 位×8 个寄存器×4 组)
- ROM: 32/64 KB, RAM: 2/4 KB, 数据闪存: 4 KB
- PWM 输出用 16 位定时器 KB0 至 KB2、以及 KC0
 - 16 位定时器 KB0 至 KB2: 最大 6 输出 (3 通道 × 2)
 - 软启动功能、抖动功能、强制输出停止功能 (通过比较器或外部中断)和支持 PFC 控制的交织功能
 - 平均分辨率 1 nsec 输出, 64 MHz (当使用 PLL) + 抖动选项
 - 16 位定时器 KC0 (3 通道)
 - 通过联锁 16 位定时器 KB0、KB1 和 KB2 的 PWM 输出门控功能
- 内置比较器(4 至 6 通道)/可编程增益放大器(1 通道)
- 内置单电源闪存(具有禁止块擦除/写入功能)
- 支持自编程功能 (具有引导交换功能/闪存屏蔽窗口功能)
- On-chip 调试功能
- 上电复位(POR)电路和电压检测电路(LVD)
- 内置乘除法和乘加器
 - 16 位 × 16 位 = 32 位 (无符号或者有符号)
 - 32 位 ÷ 32 位 = 32 位 (无符号)
 - 16 位 × 16 位 + 32 位 = 32 位 (无符号或者有符号)
- 内置十进制调整(BCD)电路
- 输入/输出端口: 13 至 29
- 定时器
 - 定时器阵列单元: 16 位定时器 × 8 通道
 - 看门狗定时器: 1 通道
 - 实时时钟: 1 通道
 - 12 位间隔定时器: 1 通道
- 串行接口
 - CSI, UART, I²C (SM/PM 总线)
 - UART 支持照明通信的 DALI 和 DMX512
- 8/10 位分辨率 A/D 转换器 (V_{DD} = 2.7 至 5.5 V): 6 至 11 通道
- 电源电压: V_{DD} = 2.7 至 5.5 V
- 运行环境温度: T_A = -40 至+105°C, T_A = -40 至+125°C (开发环境下)

- 备注 1.** 安装的功能因产品而异。请参阅 **1.6 功能概述**。
- 2.** 有关扩展温度产品(运行环境温度-40°C 至+125°C), 请联络瑞萨电子公司或授权的瑞萨电子公司代理商。

○ ROM, RAM 容量

Flash ROM	数据闪存	RAM	RL78/I1A			
			20引脚	30引脚	32引脚	38引脚
64 KB	4 KB	4 KB ^注	–	R5F107AE	–	R5F107DE
32 KB	4 KB	2 KB	R5F1076C	R5F107AC	R5F107BC	–

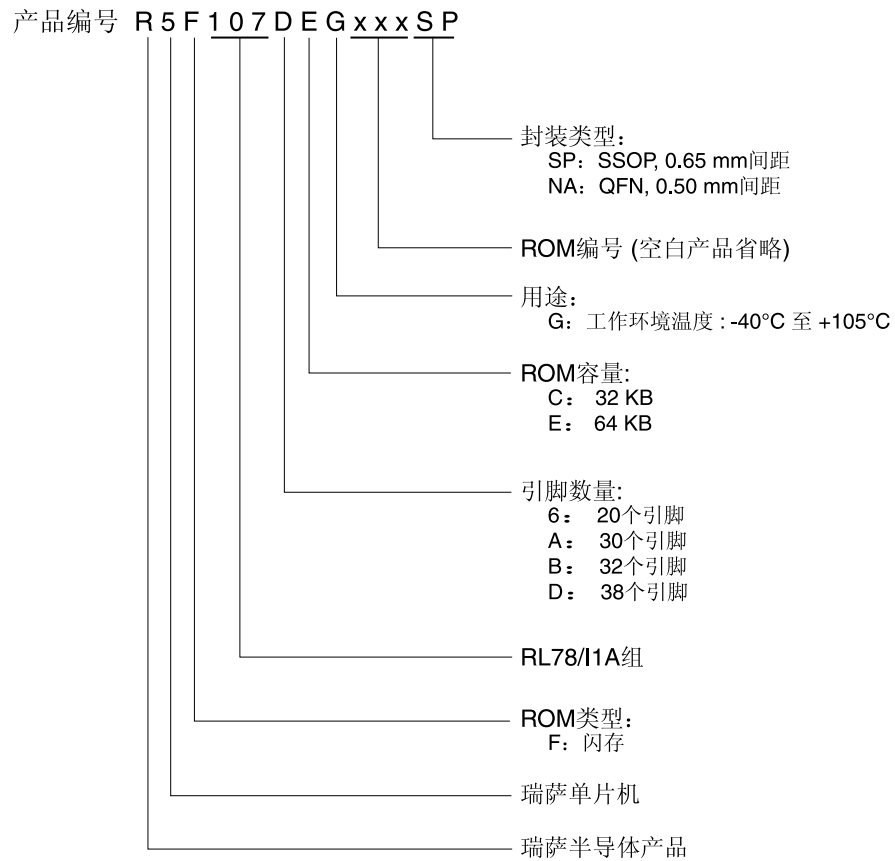
注 使用自编程功能和数据闪存功能时，该值约为 3 KB。(有关详情，请参阅第三章)

1.2 订购信息

引脚数量	封装	运行环境温度	产品编号
20引脚	20引脚塑封TSSOP (5.72 mm (225))	TA = -40至+105°C	R5F1076CGSP
		TA = -40至+125°C	T.B.D.
30引脚	30引脚塑封SSOP (7.62 mm (300))	TA = -40至+105°C	R5F107ACGSP, R5F107AEGSP
		TA = -40至+125°C	T.B.D.
32引脚	32引脚塑封VQFN (5 × 6)	TA = -40至+105°C	R5F107BCGNA
		TA = -40至+125°C	T.B.D.
38引脚	38引脚塑封SSOP (7.62 mm (300))	TA = -40至+105°C	R5F107DEGSP
		TA = -40至+125°C	T.B.D.

注意事项 RL78/I1A 提供用于开发和评估的 **On-chip** 调试功能。对于量产类产品，不得使用 **On-chip** 调试功能，因为在使用该功能时，可能会超过闪存的担保可重写次数，因而会使产品可靠性失去保障。对于使用 **On-chip** 调试功能时发生的问题，瑞萨电子不承担任何责任。

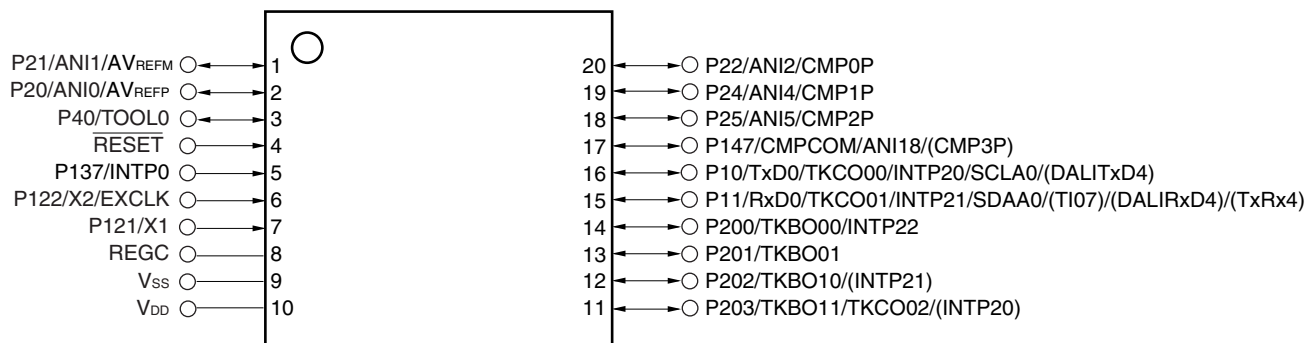
图 1-1. RL78/I1A 的产品编号、存储器大小和封装



1.3 引脚配置(顶视图)

1.3.1 20 引脚产品

- 20 引脚塑封 TSSOP (5.72 mm (225))



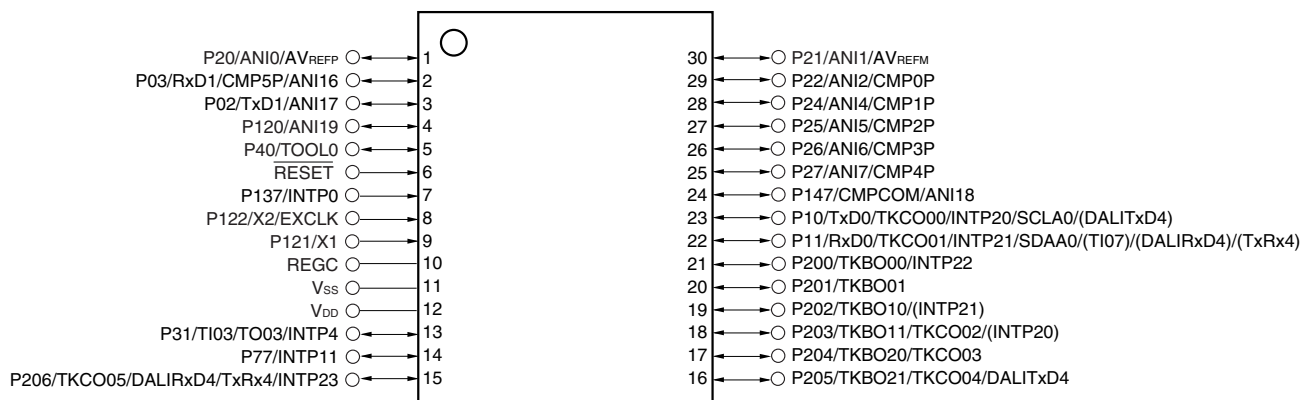
注意事项 通过一个电容 (0.47 至 1 μF) 将 REGC 引脚连接至 VSS。

备注 1. 有关引脚名称, 请参阅 1.4 引脚名称。

2. 可以通过设置外围输入/输出重定向寄存器(PIOR1)或输入切换控制寄存器(ISC)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式和图 15-18. 输入切换控制寄存器(ISC)的格式。
3. 分享功能的 CMP3P 可通过设置比较器输入切换控制寄存器(CMPSEL)的 CMPSEL0 位来分配至 P147。

1.3.2 30 引脚产品

- 30 引脚塑封 SSOP (7.62 mm (300))



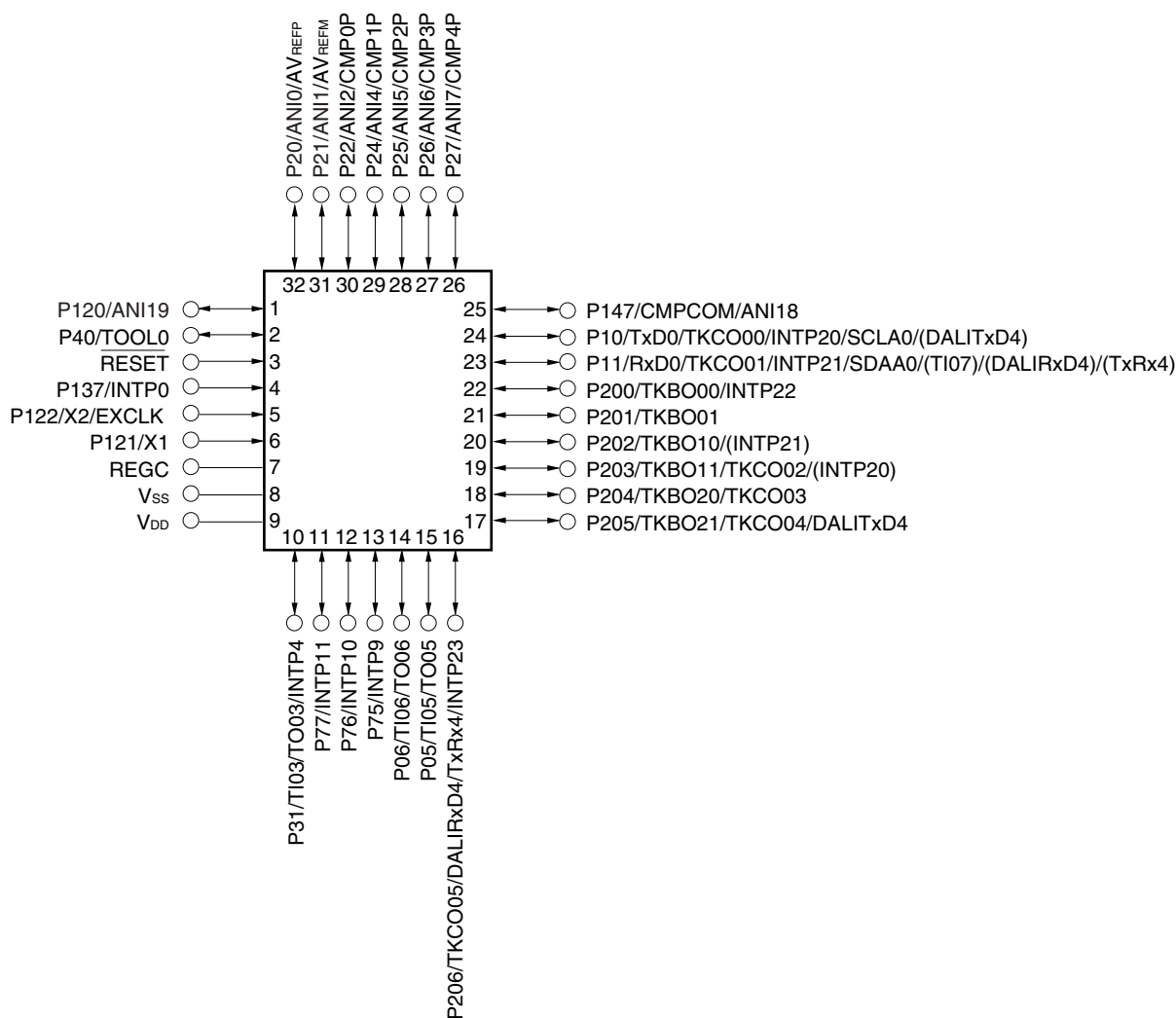
注意事项 通过一个电容 (0.47 至 1 μ F) 将 REGC 引脚连接至 V_{SS}。

备注 1. 有关引脚名称, 请参阅 1.4 引脚名称。

2. 可以通过设置外围输入/输出重定向寄存器(PIOR1)或输入切换控制寄存器(ISC)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式和图 15-18. 输入切换控制寄存器(ISC)的格式。

1.3.3 32 引脚产品

- 32 引脚塑封 VQFN (5 × 6)

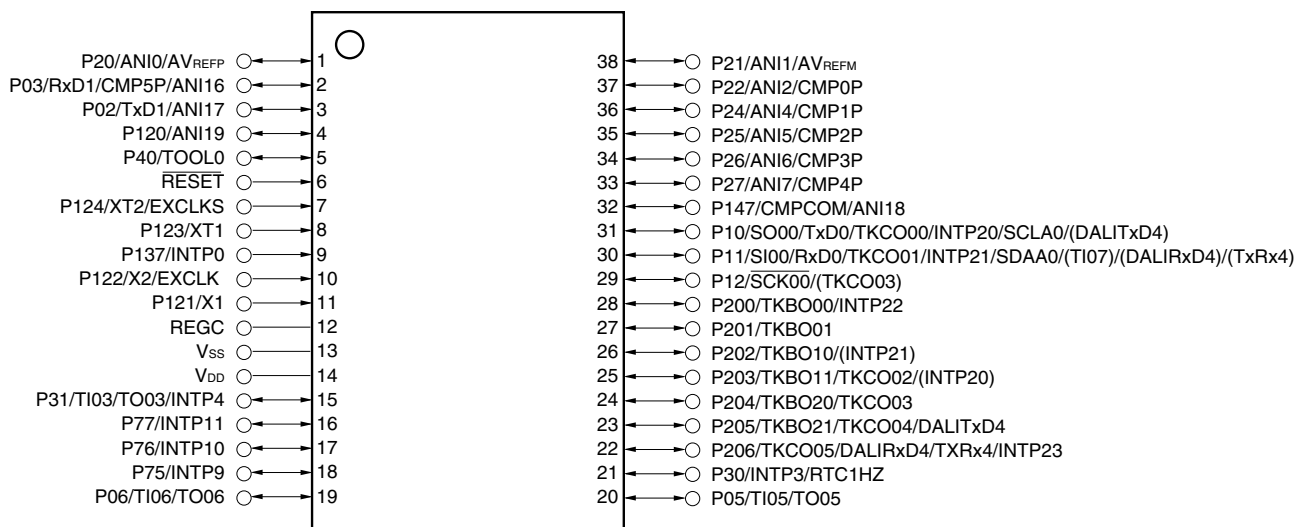


- 注意事项**
1. 通过一个电容 (0.47 至 1 μF) 将 REGC 引脚连接至 VSS。
 2. 32 引脚塑封 VQFN 的规划、规格和引脚的配置为初步内容, 有变更的可能性。

- 备注**
1. 有关引脚名称, 请参阅 1.4 引脚名称。
 2. 可以通过设置外围输入/输出重定向寄存器(PIOR1)或输入切换控制寄存器(ISC)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式和图 15-18. 输入切换控制寄存器(ISC)的格式。

1.3.4 38 引脚产品

- 38 引脚塑封 SSOP (7.62 mm (300))



注意事项 通过一个电容 (0.47 至 1 μF) 将 REGC 引脚连接至 V_{SS}。

备注 1. 有关引脚名称, 请参阅 1.4 引脚名称。

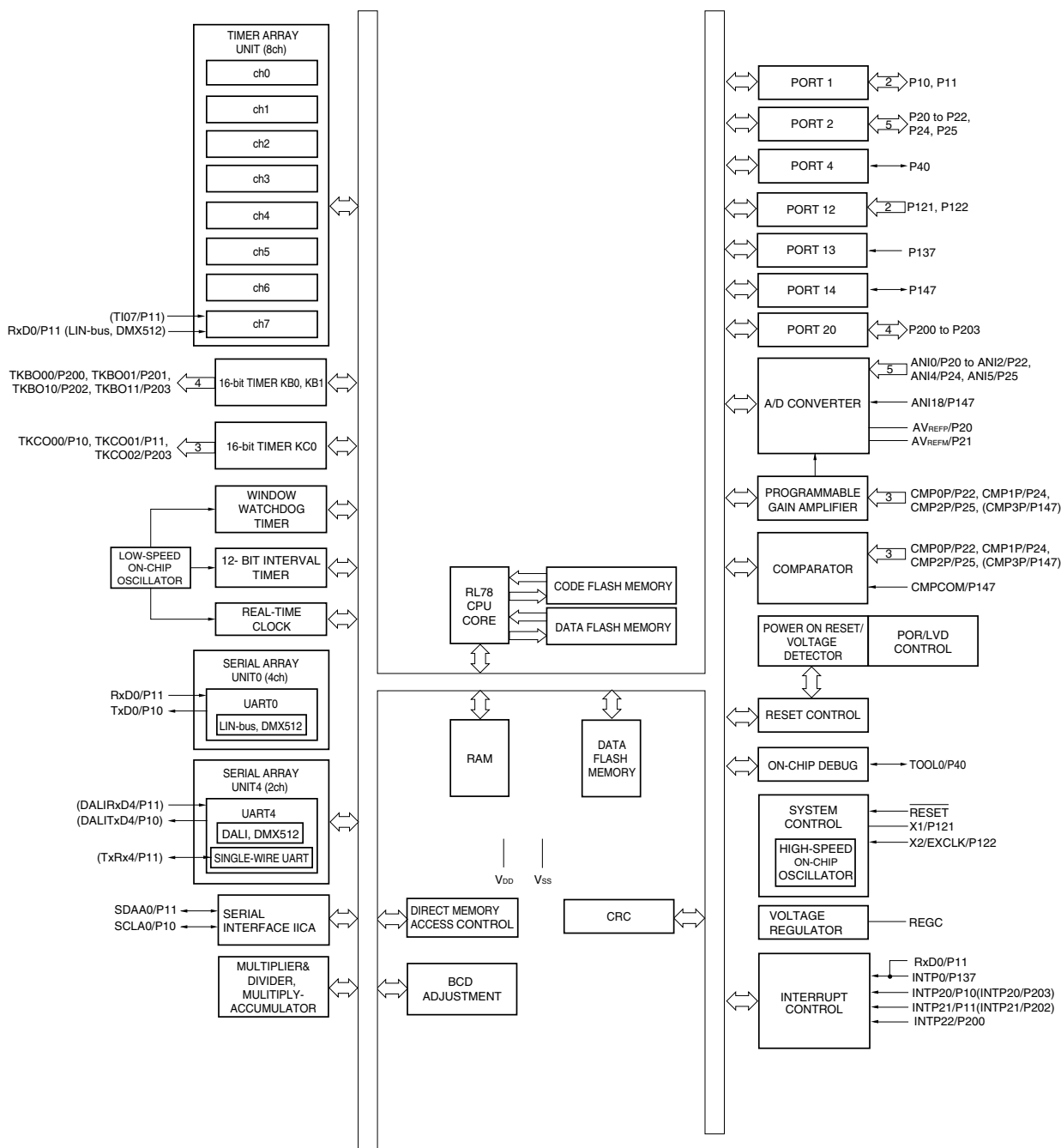
2. 可以通过设置外围输入/输出重定向寄存器(PIOR1)或输入切换控制寄存器(ISC)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式和图 15-18. 输入切换控制寄存器(ISC)的格式。

1.4 引脚名称

ANI0 至 ANI2, ANI4 至 ANI7, ANI16 至 ANI19:	模拟输入	REGC:	稳压器电容
AVREFM:	模拟基准电压 (- 极)	RESET:	复位
AVREFP:	模拟基准电压 (+ 极)	RTC1HZ:	实时时钟校正时钟 (1 Hz)输出
CMP0P 至 CMP5P:	比较器模拟输入	RxD0, RxD1, DALIRxD4:	接收数据
CMPCOM:	比较器外部基准电压	SCK00:	串行时钟输入/输出
EXCLK:	外部时钟输入(主系统时钟)	SCLA0:	串行时钟输入/输出
EXCLKS:	外部时钟输入(副系统时钟)	SDAA0:	串行数据输入/输出
INTP0, INTP3, INTP4, INTP9, INTP10, INTP11, INTP20 至 INTP23:	外部中断输入	SI00:	串行数据输入
P02, P03, P05, P06:	端口 0	SO00:	串行数据输出
P10 至 P12:	端口 1	TI03, TI05, TI06, TI07:	定时器输入
P20 至 P22, P24 至 P27:	端口 2	TO03, TO05, TO06, TKBO00, TKBO01 至 TKBO20, TKBO21,	定时器输出
P30, P31:	端口 3	TKCO00-TKCO05:	定时器输出
P40:	端口 4	TOOL0:	工具用数据输入/输出
P75 至 P77:	端口 7	TxRx4:	单线 UART 的串行数据输入/输出
P120 至 P124:	端口 12	TxD0, TxD1	
P137:	端口 13	DALITxD4:	发送数据
P147:	端口 14	VDD:	电源
P200 至 P206:	端口 20	VSS:	接地
		X1, X2:	晶振 (主系统时钟)
		XT1, XT2:	晶振 (副系统时钟)

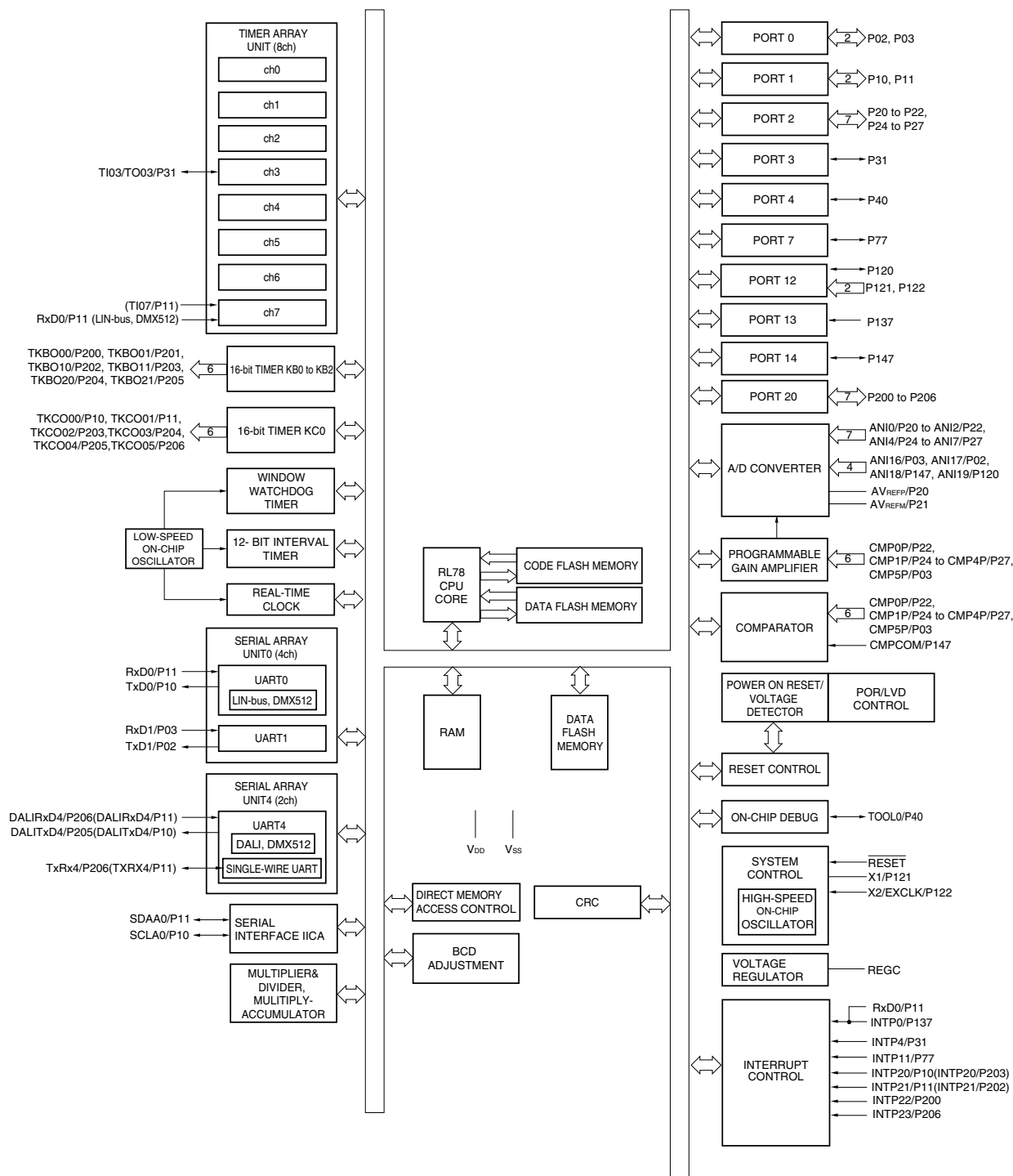
1.5 框图

1.5.1 20 引脚产品



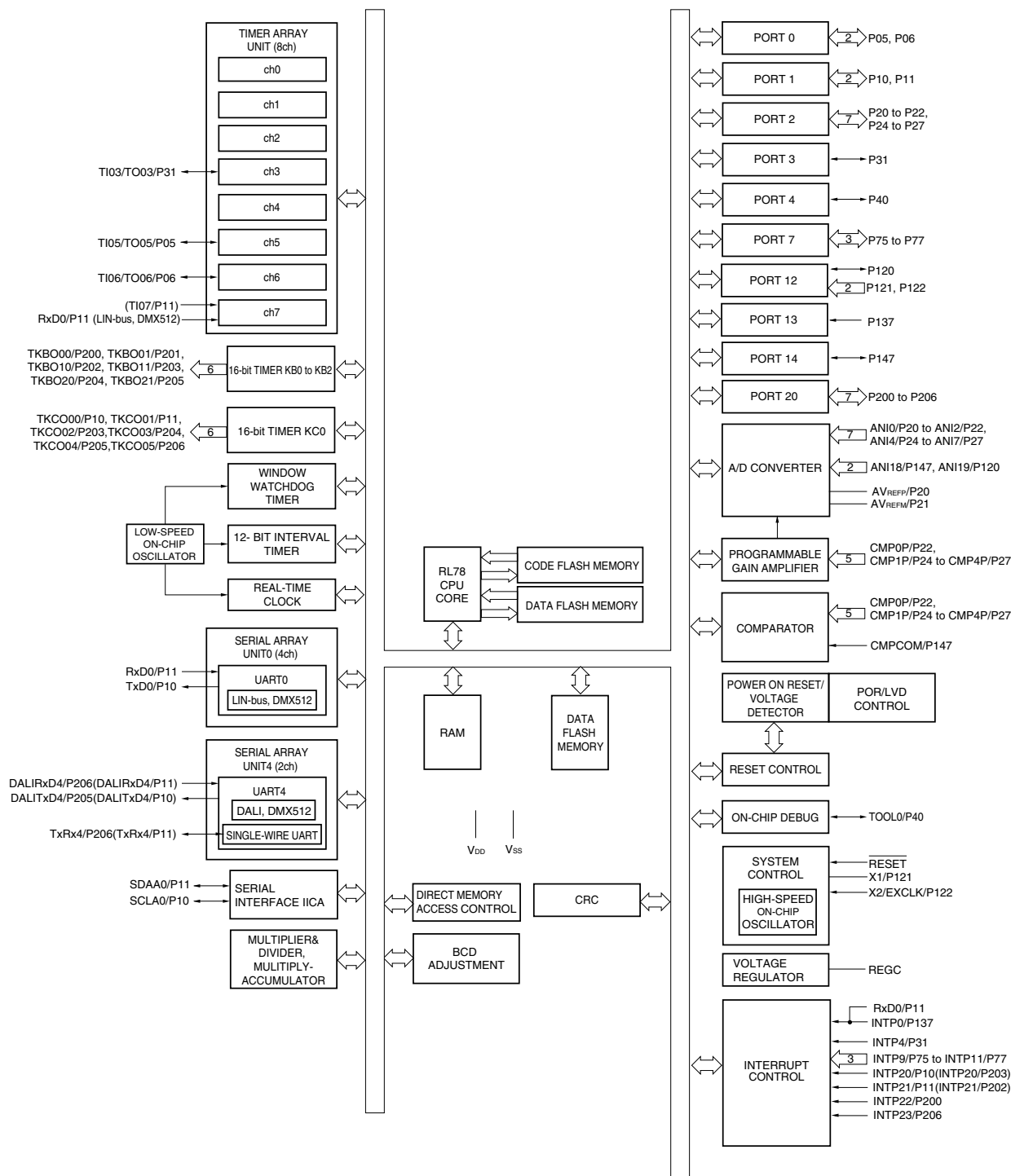
- 备注 1. 可以通过设置外围输入/输出重定向寄存器(PIOR1)或输入切换控制寄存器(ISC)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式和图 15-18. 输入切换控制寄存器(ISC)的格式。
2. 分享功能的 CMP3P 可通过设置比较器输入切换控制寄存器(CMPSEL)的 CMPSEL0 位来分配至 P147。

1.5.2 30 引脚产品



备注 可以通过设置外围输入/输出重定向寄存器(PIOR1)或输入切换控制寄存器(ISC)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式和图 15-18. 输入切换控制寄存器(ISC)的格式。

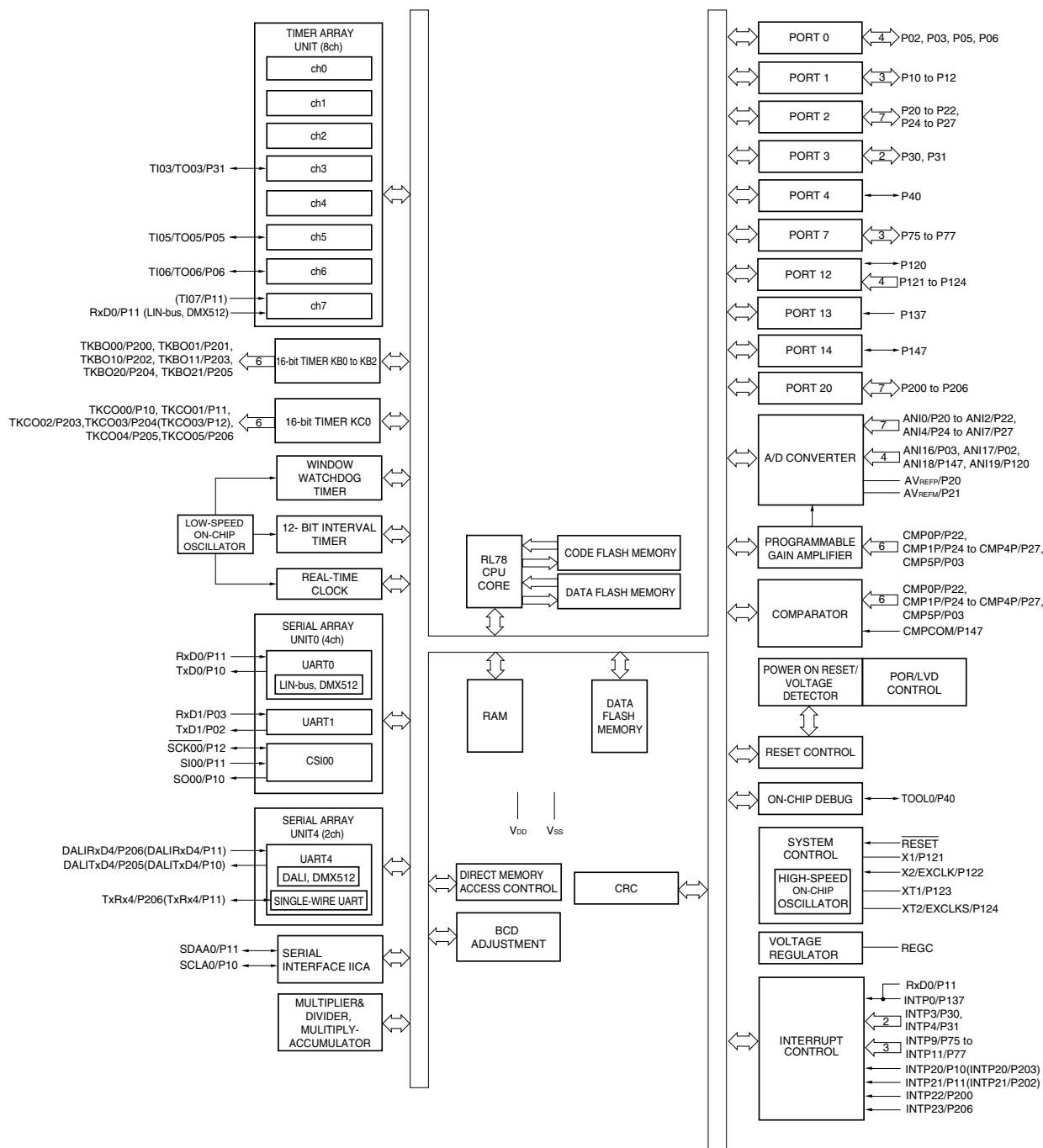
1.5.3 32 引脚产品



注意事项 32 引脚塑封 VQFN 的规划、规格和引脚的配置为初步内容，有变更的可能性。

备注 可以通过设置外围输入/输出重定向寄存器(PIOR1)或输入切换控制寄存器(ISC)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式和图 15-18. 输入切换控制寄存器(ISC)的格式。

1.5.4 38 引脚产品



备注 可以通过设置外围输入/输出重定向寄存器(PIOR1)或输入切换控制寄存器(ISC)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式和图 15-18. 输入切换控制寄存器(ISC)的格式。

1.6 功能概述

- 注意事项 1. 32 引脚塑封 VQFN 的规划、规格和引脚的配置为初步内容，有变更的可能性。
2. 本概述描述的是将外围输入/输出重定向寄存器(PIOR1)设置为 00H 时的功能。

(1/3)

项目		20 引脚	30 引脚		32 引脚	38 引脚
		R5F1076C	R5F107AC	R5F107AE	R5F107BC	R5F107DE
代码闪存(KB)		32	32	64	32	64
数据闪存(KB)		4	4	4	4	4
RAM (KB)		2	2	4 ^{#1}	2	4 ^{#1}
存储空间		1 MB				
主系统时钟	高速系统时钟	X1 (晶体/陶瓷) 振荡，外部主系统时钟输入(EXCLK) 1 至 20 MHz: V _{DD} = 2.7 至 5.5 V				
	高速片上振荡器	HS (高速主) 模式: 1 至 32 MHz (V _{DD} = 2.7 至 5.5 V) LS (低速主) 模式: 1 至 8 MHz (V _{DD} = 2.7 至 5.5 V)				
16 位定时器 KB0 至 KB2、以及 KC0 用的时钟		64 MHz (TYP.): V _{DD} = 2.7 至 5.5 V				
副系统时钟 (仅限 38 引脚产品)		XT1 (晶体) 振荡，外部副系统时钟输入(EXCLKS) 32.768 kHz (TYP.): V _{DD} = 2.7 至 5.5 V				
低速片上振荡器时钟		内部振荡 15 kHz (TYP.): V _{DD} = 2.7 至 5.5 V				
通用寄存器		8 位×32 个寄存器(8 位×8 个寄存器×4 组)				
最短指令执行时间		0.03125 μs (高速片上振荡器: f _{IH} = 32 MHz 操作)				
		0.05 μs (高速系统时钟: f _{MX} = 20 MHz 操作)				
		30.5 μs (副系统时钟: f _{SUB} = 32.768 kHz 操作) (仅限 38 引脚产品)				
指令集		<ul style="list-style-type: none"> • 8 位操作, 16 位操作 • 乘法 (8 位 × 8 位) • 位操作 (置位, 复位, 测试和布尔运算)等 				
输入/输出端口	总数	16	26		28	34
	CMOS 输入/输出	13	23		25	29
	CMOS 输入	3	3		3	5
	CMOS 输出	-	-		-	-
定时器	16 位定时器 TAU	8 通道(无定时器输出)	8 通道(定时器输出: 1, PWM 输出: 1 ^{#2})	8 通道(定时器输出: 3, PWM 输出: 3 ^{#2})	8 通道(定时器输出: 3, PWM 输出: 3 ^{#2})	8 通道(定时器输出: 3, PWM 输出: 3 ^{#2})
	16 位定时器 KB	2 通道 (PWM 输出: 4)	3 通道 (PWM 输出: 6)			
	16 位定时器 KC	1 通道 (PWM 输出: 3)	1 通道 (PWM 输出: 6)			

- 注 1. 使用自编程功能和数据闪存功能时，该值约为 3 KB。(有关详情，请参阅第三章)
2. 输出值因定时器的数目和所用通道的设置而异（请参阅 6.8.3 作为多重 PWM 输出功能的操作）。

(2/3)

项目		20 引脚	30 引脚	32 引脚	38 引脚
		R5F1076C	R5F107AC, R5F107AE	R5F107BC	R5F107DE
定时器	看门狗定时器	1 通道			
	实时时钟(RTC)	1 通道 ^{1,2}			
	12 位间隔定时器(IT)	1 通道			
	RTC 输出	-			1 个 1 Hz (副系统时钟: f _{SUB} = 32.768 kHz)
8/10 位分辨率 A/D 转换器		6 通道	11 通道	9 通道	11 通道
比较器		4 通道	6 通道	5 通道	6 通道
可编程增益放大器		1 通道			
		输入 ³	4 通道	6 通道	5 通道
串行接口		[20、32 引脚产品] <ul style="list-style-type: none"> • UART(对应 LIN-bus 和 DMX512): 1 通道 • UART(对应 DALI 通信): 1 通道 [30 引脚产品] <ul style="list-style-type: none"> • UART(对应 LIN-bus 和 DMX512): 1 通道 • UART: 1 通道 • UART(对应 DALI 通信): 1 通道 [38 引脚产品] <ul style="list-style-type: none"> • CSI: 1 通道/UART (对应 LIN-bus 和 DMX512): 1 通道 • UART: 1 通道 • UART(对应 DALI 通信): 1 通道 			
		I ² C 总线	1 通道	1 通道	1 通道
乘除器和乘加器		<ul style="list-style-type: none"> • 16 位 × 16 位 = 32 位 (无符号或者有符号) • 32 位 ÷ 32 位 = 32 位 (无符号) • 16 位 × 16 位 + 32 位 = 32 位 (无符号或者有符号) 			
DMA 控制器		2 通道			
向量中断源	内部	27	30	28	30
	外部	7	10	10	11
复位		<ul style="list-style-type: none"> • 通过 RESET 引脚复位 • 通过看门狗定时器的内部复位 • 通过上电复位的内部复位 • 通过电压检测器的内部复位 • 通过执行非法指令⁴的内部复位 • 通过 RAM 奇偶检验错误的内部复位 • 通过非法存取内存的内部复位 			

注 1. 仅在 38 引脚产品中，可以选择副系统时钟(f_{SUB})作为操作时钟。

2. 在 20 引脚、30 引脚和 32 引脚产品中仅可使用固定周期中断功能。

3. 比较器输入与模拟输入引脚(ANI 引脚)兼用。

(3/3)

项目	20 引脚	30 引脚	32 引脚	38 引脚
	R5F1076C	R5F107AC, R5F107AE	R5F107BC	R5F107DE
上电复位电路	<ul style="list-style-type: none"> 上电复位: 1.51 ±0.03 V 掉电复位: 1.50 ±0.03 V 			
电压检测电路	<ul style="list-style-type: none"> 上升沿: 2.81 V 至 4.06 V (6 个级别) 下降沿: 2.75 V 至 3.98 V (6 个级别) 			
On-chip 调试功能	有			
电源电压	V _{DD} = 2.7 至 5.5 V			
运行环境温度	T _A = -40 至 +105 °C T _A = -40 至 +125 °C (T.B.D)			

注 执行指令代码 FFH 时产生非法指令。

在电路内置仿真器或片上调试仿真器的仿真中，不会因执行非法指令发生内部复位。

第二章 引脚功能

2.1 端口功能

引脚的输入/输出缓冲器电源由 V_{DD} 提供。

各个端口中设置的输入/输出、缓冲器、上拉电阻，在兼用功能中也有效。

2.1.1 20 引脚产品

功能名称	输入/输出	功能	复位后	复用功能
P10	输入/输出	端口 1。 2 位输入/输出端口。 P10 和 P11 的输入可被设置为 TTL 输入缓冲器。 P10 和 P11 的输出可被设置为 N 沟开漏输出 (V _{DD} 耐压)。 可以以 1 位为单位指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	TxD0/TKCO00/INTP20/SCLA0/(DALITxD4)
P11				RxD0/TKCO01/INTP21/SDAA0/(TI07)/(DALIRxD4)/(TxRx4)
P20	输入/输出	端口 2。 5 位输入/输出端口。 可被设置为模拟输入 ^{注1} 。 可以以 1 位为单位指定为输入/输出。	模拟输入端口	ANI0/AV _{REFP}
P21				ANI1/AV _{REFM}
P22				ANI2/CMP0P
P24				ANI4/CMP1P
P25				ANI5/CMP2P
P40	输入/输出	端口 4。 1 位输入/输出端口。 可以指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	TOOL0
P121	输入	端口 12。 2 位输入专用端口。	输入端口	X1
P122				X2/EXCLK
P137	输入	端口 13。 1 位输入专用端口。	输入端口	INTP0
P147	输入/输出	端口 14。 1 位输入/输出端口。 P147 可被设置为模拟输入 ^{注2} 。 可以指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	ANI18/CMPCOM/(CMP3P)
P200	输入/输出	端口 20。 4 位输入/输出端口。 P200 至 P203 的输出可被设置为 N 沟开漏输出 (V _{DD} 耐压)。 可以以 1 位为单位指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	TKBO00/INTP22
P201				TKBO01
P202				TKBO10/(INTP21)
P203				TKBO11/TKCO02/(INTP20)

注1. 可以通过设置A/D端口配置寄存器(ADPC)来指定各个引脚为数字或模拟。

- 使用各引脚用于输入时，在端口模式控制寄存器x (PMCx)中将它们指定为数字输入或模拟输入（该寄存器以1位为单位指定）。

备注 可以通过设置外围输入/输出重定向寄存器(PIOR1)或输入切换控制寄存器(ISC)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式和图 15-18. 输入切换控制寄存器(ISC)的格式。

2.1.2 30 引脚产品

(1/2)

功能名称	输入/输出	功能	复位后	复用功能
P02	输入/输出	端口 0。 2 位输入/输出端口。 P03 的输入可被设置为 TTL 输入缓冲器。 P02 的输出可被设置为 N 沟开漏输出 (V_{DD} 耐压)。 P02 和 P03 可被设置为模拟输入 ^{注1} 。 可以以 1 位为单位指定为输入/输出。 可通过软件设置使用片上上拉电阻。	模拟输入端口	TxD1/ANI17
P03				RxD1/CMP5P/ANI16
P10	输入/输出	端口 1。 2 位输入/输出端口。 P10 和 P11 的输入可被设置为 TTL 输入缓冲器。 P10 和 P11 的输出可被设置为 N 沟开漏输出 (V_{DD} 耐压)。 可以以 1 位为单位指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	TxD0/TKCO00/INTP2 0/SCLA0/(DALITxD4)
P11				RxD0/TKCO01/INTP2 1/SDAA0/(TI07)/(DALI RxD4)/(TxRx4)
P20	输入/输出	端口 2。 7 位输入/输出端口。 可被设置为模拟输入 ^{注2} 。 可以以 1 位为单位指定为输入/输出。	模拟输入端口	ANI0/AV _{REFP}
P21				ANI1/AV _{REFM}
P22				ANI2/CMP0P
P24				ANI4/CMP1P
P25				ANI5/CMP2P
P26				ANI6/CMP3P
P27				ANI7/CMP4P
P31	输入/输出	端口 3。 1 位输入/输出端口。 可以指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	TI03/TO03/INTP4
P40	输入/输出	端口 4。 1 位输入/输出端口。 可以指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	TOOL0

注1. 使用各引脚用于输入时，在端口模式控制寄存器x (PMCx)中将它们指定为数字输入或模拟输入（该寄存器以1位为单位指定）。

2. 可以通过设置A/D端口配置寄存器(ADPC)来指定各个引脚为数字或模拟。

备注 可以通过设置外围输入/输出重定向寄存器(PIOR1)或输入切换控制寄存器(ISC)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式和图 15-18. 输入切换控制寄存器(ISC)的格式。

(2/2)

功能名称	输入/输出	功能	复位后	复用功能
P77	输入/输出	端口 7。 1 位输入/输出端口。 可以指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	INTP11
P120	输入/输出	端口 12。 1 位输入/输出端口和 2 位输入端口。 P120 可被设置为模拟输入。 可以以 1 位为单位指定为输入/输出。 仅限 P120 可通过软件设置使用片上上拉电阻。	输入端口	ANI19
P121	输入			X1
P122				X2/EXCLK
P137	输入	端口 13。 1 位输入专用端口。	输入端口	INTP0
P147	输入/输出	端口 14。 1 位输入/输出端口。 P147 可被设置为模拟输入。 可以指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	ANI18/CMPCOM
P200	输入/输出	端口 20。 7 位输入/输出端口。 P200 至 206 的输出可被设置为 N 沟开漏输出 (V_{DD} 耐压)。 可以以 1 位为单位指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	TKBO00/INTP22
P201				TKBO01
P202				TKBO10/(INTP21)
P203				TKBO11/TKCO02/(INTP20)
P204				TKBO20/TKCO03
P205				TKBO21/TKCO04/DALITxD4
P206				TKCO05/DALIRxD4/TxRx4/INTP23

注 使用各引脚用于输入时，在端口模式控制寄存器x (PMCx)中将它们指定为数字输入或模拟输入（该寄存器以1位为单位指定）。

备注 可以通过设置外围输入/输出重定向寄存器(PIOR1)或输入切换控制寄存器(ISC)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式和图 15-18. 输入切换控制寄存器(ISC)的格式。

2.1.3 32 引脚产品

(1/2)

功能名称	输入/输出	功能	复位后	复用功能
P05	输入/输出	端口 0。 2 位输入/输出端口。 可以以 1 位为单位指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	TI05/TO05
P06				TI06/TO06
P10	输入/输出	端口 1。 2 位输入/输出端口。 P10 和 P11 的输入可被设置为 TTL 输入缓冲器。 P10 和 P11 的输出可被设置为 N 沟开漏输出 (V_{DD} 耐压)。 可以以 1 位为单位指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	TxD0/TKCO00/INTP2 0/SCLA0/(DALITxD4)
P11				RxD0/TKCO01/INTP2 1/SDAA0/(TI07)/(DALI RxD4)/(TxRx4)
P20	输入/输出	端口 2。 7 位输入/输出端口。 可被设置为模拟输入 ^注 。 可以以 1 位为单位指定为输入/输出。	模拟输入端口	ANI0/AV _{REFP}
P21				ANI1/AV _{REFM}
P22				ANI2/CMP0P
P24				ANI4/CMP1P
P25				ANI5/CMP2P
P26				ANI6/CMP3P
P27				ANI7/CMP4P
P31	输入/输出	端口 3。 1 位输入/输出端口。 可以指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	TI03/TO03/INTP4
P40	输入/输出	端口 4。 1 位输入/输出端口。 可以指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	TOOL0

注 可以通过设置A/D端口配置寄存器(ADPC)来指定各个引脚为数字或模拟。

备注 可以通过设置外围输入/输出重定向寄存器(PIOR1)或输入切换控制寄存器(ISC)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式和图 15-18. 输入切换控制寄存器(ISC)的格式。

(2/2)

功能名称	输入/输出	功能	复位后	复用功能
P75	输入/输出	端口 7。 3 位输入/输出端口。 可以以 1 位为单位指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	INTP9
P76				INTP10
P77				INTP11
P120	输入/输出	端口 12。 1 位输入/输出端口和 2 位输入专用端口。 P120 可被设置为模拟输入。 可以以 1 位为单位指定为输入/输出。 仅限 P120 可通过软件设置使用片上上拉电阻。	输入端口	ANI19
P121	输入			X1
P122				X2/EXCLK
P137	输入	端口 13。 1 位输入专用端口。	输入端口	INTP0
P147	输入/输出	端口 14。 1 位输入/输出端口。 P147 可被设置为模拟输入。 可以指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	ANI18/CMPCOM
P200	输入/输出	端口 20。 7 位输入/输出端口。 P200 至 206 的输出可被设置为 N 沟开漏输出 (V_{DD} 耐压)。 可以以 1 位为单位指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	TKBO00/INTP22
P201				TKBO01
P202				TKBO10/(INTP21)
P203				TKBO11/TKCO02/(INTP20)
P204				TKBO20/TKCO03
P205				TKBO21/TKCO04/DALITxD4
P206				TKCO05/DALIRxD4/TxRx4/INTP23

注 使用各引脚用于输入时，在端口模式控制寄存器x (PMCx)中将它们指定为数字输入或模拟输入（该寄存器以1位为单位指定）。

备注 可以通过设置外围输入/输出重定向寄存器(PIOR1)或输入切换控制寄存器(ISC)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式和图 15-18. 输入切换控制寄存器(ISC)的格式。

2.1.4 38 引脚产品

(1/2)

功能名称	输入/输出	功能	复位后	复用功能
P02	输入/输出	端口 0。 4 位输入/输出端口。 P03 的输入可被设置为 TTL 输入缓冲器。 P02 的输出可被设置为 N 沟开漏输出 (V_{DD} 耐压)。 P02 和 P03 可被设置为模拟输入 ^{注1} 。 可以以 1 位为单位指定为输入/输出。 可通过软件设置使用片上上拉电阻。	模拟输入端口	TxD1/ANI17
P03				RxD1/CMP5P/ANI16
P05				TI05/TO05
P06				TI06/TO06
P10	输入/输出	端口 1。 3 位输入/输出端口。 P10 和 P11 的输入可被设置为 TTL 输入缓冲器。 P10 至 P12 的输出可被设置为 N 沟开漏输出 (V_{DD} 耐压)。 可以以 1 位为单位指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	SO00/TxD0/TKCO00/I NTP20/SCLA0/(DALIT xD4)
P11				SI00/RxD0/TKCO01/I NTP21/SDAA0/(TI07)/ (DALIRxD4)/(TxRx4)
P12				SCK00/(TKCO03)
P20	输入/输出	端口 2。 7 位输入/输出端口。 可被设置为模拟输入 ^{注2} 。 可以以 1 位为单位指定为输入/输出。	模拟输入端口	ANI0/AV _{REFP}
P21				ANI1/AV _{REFM}
P22				ANI2/CMP0P
P24				ANI4/CMP1P
P25				ANI5/CMP2P
P26				ANI6/CMP3P
P27				ANI7/CMP4P
P30	输入/输出	端口 3。 2 位输入/输出端口。 可以以 1 位为单位指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	INTP3/RTC1HZ
P31				TI03/TO03/INTP4
P40	输入/输出	端口 4。 1 位输入/输出端口。 可以指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	TOOL0

注1. 使用各引脚用于输入时，在端口模式控制寄存器x (PMCx)中将它们指定为数字输入或模拟输入（该寄存器以1位为单位指定）。

2. 可以通过设置A/D端口配置寄存器(ADPC)来指定各个引脚为数字或模拟。

备注 可以通过设置外围输入/输出重定向寄存器(PIOR1)或输入切换控制寄存器(ISC)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式和图 15-18. 输入切换控制寄存器(ISC)的格式。

(2/2)

功能名称	输入/输出	功能	复位后	复用功能
P75	输入/输出	端口 7。 3 位输入/输出端口。 可以以 1 位为单位指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	INTP9
P76				INTP10
P77				INTP11
P120	输入/输出	端口 12。 1 位输入/输出端口和 4 位输入专用端口。 P120 可被设置为模拟输入 ^注 。 可以以 1 位为单位指定为输入/输出。 仅限 P120 可通过软件设置使用片上上拉电阻。	输入端口	ANI19
P121				X1
P122				X2/EXCLK
P123				XT1
P124				XT2/EXCLKS
P137	输入	端口 13。 1 位输入专用端口。	输入端口	INTP0
P147	输入/输出	端口 14。 1 位输入/输出端口。 P147 可被设置为模拟输入。 可以指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	ANI18/CMPCOM
P200	输入/输出	端口 20。 7 位输入/输出端口。 P200 至 206 的输出可被设置为 N 沟开漏输出 (V _{DD} 耐压)。 可以以 1 位为单位指定为输入/输出。 可通过软件设置使用片上上拉电阻。	输入端口	TKBO00/INTP22
P201				TKBO01
P202				TKBO10/(INTP21)
P203				TKBO11/TKCO02/(INTP20)
P204				TKBO20/TKCO03
P205				TKBO21/TKCO04/DALITxD4
P206				TKCO05/DALIRxD4/TxRx4/INTP23

注 使用各引脚用于输入时，在端口模式控制寄存器x (PMCx)中将它们指定为数字输入或模拟输入（该寄存器以1位为单位指定）。

备注 可以通过设置外围输入/输出重定向寄存器(PIOR1)或输入切换控制寄存器(ISC)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式和图 15-18. 输入切换控制寄存器(ISC)的格式。

2.2 端口以外的引脚功能

2.2.1 各种产品的配置功能

(1/2)

功能名称	输入/输出	功能	38 引脚	32 引脚	30 引脚	20 引脚
ANI0	输入	A/D 转换器的模拟输入 (参阅 图 12-46. 模拟输入引脚的连接)	√	√	√	√
ANI1			√	√	√	√
ANI2			√	√	√	√
ANI4			√	√	√	√
ANI5			√	√	√	√
ANI6			√	√	√	–
ANI7			√	√	√	–
ANI16			√	–	√	–
ANI17			√	–	√	–
ANI18			√	√	√	√
ANI19			√	√	√	–
CMP0P			输入	比较器 0 至 5 的模拟输入	√	√
CMP1P	√	√			√	√
CMP2P	√	√			√	√
CMP3P	√	√			√	√
CMP4P	√	√			√	–
CMP5P	√	–			√	–
CMPCOM	输入	比较器外部基准电位输入	√	√	√	√
DALIRxD4	输入	串行数据输入至 DALI/UART4	√	√	√	(√)
DALITxD4	输出	DALI/UART4 的串行数据输出	√	√	√	(√)
INTP0	输入	这是一个外部中断请求输入引脚, 可指定有效边沿 (上升沿、下降沿或上升沿与下降沿的双边沿)	√	√	√	√
INTP3			√	–	–	–
INTP4			√	√	√	–
INTP9			√	√	–	–
INTP10			√	√	–	–
INTP11			√	√	√	–
INTP20			√	√	√	√
INTP21			√	√	√	√
INTP22			√	√	√	√
INTP23			√	√	√	–
REGC	–	在内部操作中连接稳压器输出稳压电容。通过电容 (0.47 至 1 μ F) 将该引脚连接至 Vss。 并且, 请使用具有良好特性的电容以稳定内部电压。	√	√	√	√
RTC1HZ	输出	实时时钟校正时钟(1 Hz) 输出	√	–	–	–

备注 可以通过设置外围输入/输出重定向寄存器 (PIOR1)来指定上表括号内的功能。

(2/2)

功能名称	输入/输出	功能	38 引脚	32 引脚	30 引脚	20 引脚
RESET	输入	低电平激活的系统复位输入引脚。 未使用外部复位引脚时，直接或通过电阻将该引脚连接至 V _{DD} 。 使用外部复位引脚时，以 V _{DD} 为基准设计电路。	√	√	√	√
RxD0	输入	串行接口 UART0 和 UART1 的串行数据输入引脚	√	√	√	√
RxD1			√	-	√	-
TxD0	输出	串行接口 UART0 和 UART1 的串行数据输出引脚	√	√	√	√
TxD1			√	-	√	-
SCK00	输入/输出	串行接口 CSI00 的串行时钟输入/输出引脚	√	-	-	-
SI00	输入	串行接口 CSI00 的串行数据输入引脚	√	-	-	-
SO00	输出	串行接口 CSI00 的串行数据输出引脚	√	-	-	-
SCLA0	输入/输出	串行接口 IICA0 的串行时钟输入/输出引脚	√	√	√	√
SDAA0	输入/输出	串行接口 IICA0 的串行数据输入/输出引脚	√	√	√	√
TI03	输入	向 16 位定时器 03, 05 至 07 输入外部计数时钟/捕捉触发的引脚	√	√	√	-
TI05			√	√	-	-
TI06			√	√	-	-
TI07			√	√	√	√
TO03	输出	16 位定时器 03, 05 和 06 的定时器输出引脚	√	√	√	-
TO05			√	√	-	-
TO06			√	√	-	-
TKBO00	输出	16 位定时器 KB0 至 KB2 的输出	√	√	√	√
TKBO01			√	√	√	√
TKBO10			√	√	√	√
TKBO11			√	√	√	√
TKBO20			√	√	√	-
TKBO21			√	√	√	-
TKCO00	输出	16 位定时器 KC0 输出	√	√	√	√
TKCO01			√	√	√	√
TKCO02			√	√	√	√
TKCO03			√	√	√	-
TKCO04			√	√	√	-
TKCO05			√	√	√	-
TxRx4	输入/输出	用于单线 UART 的串行数据输入/输出	√	√	√	√
X1, X2	-	主系统时钟用谐振器连接	√	√	√	√
EXCLK	输入	主系统时钟用外部时钟输入	√	√	√	√
XT1, XT2	-	副系统时钟用谐振器连接	√	-	-	-
EXCLKS	输入	副系统时钟用外部时钟输入	√	-	-	-
V _{DD}	-	所有引脚的正电源	√	√	√	√
AV _{REFP}	输入	A/D 转换器基准电位(+极)输入	√	√	√	√
AV _{REFM}	输入	A/D 转换器基准电位(-极)输入	√	√	√	√
V _{SS}	-	所有引脚的接地电位	√	√	√	√
TOOL0	输入/输出	闪存编程器/调试器用数据输入/输出	√	√	√	√

(注意事项如下页所示。)

注意事项 解除复位后，P40/TOOL0 和操作模式之间的关系如下所示。

表 2-1. 解除复位后 P40/TOOL0 和操作模式之间的关系

P40/TOOL0	操作模式
V _{DD}	正常操作模式
0 V	闪存编程模式

有关详细，请参阅 **28.4 编程方法**。

2.3 引脚的输入/输出电路和未使用引脚的推荐连接

表 2-2 表示引脚的输入/输出电路的类型和未使用引脚的推荐连接。

备注 配置的引脚因产品而异。请参阅 1.3 引脚配置(顶视图)和 2.1 端口功能。

表 2-2. 未使用引脚的连接 (38 引脚产品) (1/2)

引脚名称	输入/输出电路类型	输入/输出	未使用引脚的推荐连接
P02/TxD1/ANI17	11-U	输入/输出	输入： 通过电阻单独连接至 V _{DD} 或 V _{SS} 。 输出： 保持开路。 <N 沟开漏时> 设置端口的输出锁存器为 0，并且输出低电平以保持引脚开路。
P03/RxD1/CMP5P/ANI16	11-W		输入： 通过电阻单独连接至 V _{DD} 或 V _{SS} 。 输出： 保持开路。
P05/TI05/TO05	8-R		
P06/TI06/TO06			
P10/SO00/TxD0/TKCO00/IN TP20/SCLA0/(DALITxD4)	5-AN		输入： 通过电阻单独连接至 V _{DD} 或 V _{SS} 。 输出： 保持开路。 <N 沟开漏时> 设置端口的输出锁存器为 0，并且输出低电平以保持引脚开路。
P11/SI00/RxD0/TKCO01/INT P21/SDAA0/(TI07)/(DALIRxD4)/(TxRx4)			
P12/SCK00/(TKCO03)	8-R		输入： 通过电阻单独连接至 V _{DD} 或 V _{SS} 。 输出： 保持开路。
P20/ANI0/AV _{REFP}	11-T		
P21/ANI1/AV _{REFM}			
P22/ANI2/CMP0P	11-X		
P24/ANI4/CMP1P			
P25/ANI5/CMP2P			
P26/ANI6/CMP3P			
P27/ANI7/CMP4P			
P30/INTP3/RTC1HZ	8-R		
P31/TI03/TO03/INTP4			
P40/TOOL0	8-R		输入： 单独连接至 V _{DD} 或保持开路。 输出： 保持开路。
P75/KR5/INTP9	8-R		输入： 通过电阻单独连接至 V _{DD} 或 V _{SS} 。 输出： 保持开路。
P76/KR6/INTP10			
P77/KR7/INTP11			

备注 可以通过设置外围输入/输出重定向寄存器(PIOR1)或输入切换控制寄存器(ISC)来指定上图括号内的功能。

表 2-2. 未使用引脚的连接 (38 引脚产品) (2/2)

引脚名称	输入/输出电路类型	输入/输出	未使用引脚的推荐连接
P120/ANI19	11-U	输入/输出	输入： 通过电阻单独连接至 V _{DD} 或 V _{SS} 。 输出： 保持开路。
P121/X1	37-C	输入	通过电阻单独连接至 V _{DD} 或 V _{SS} 。
P122/X2/EXCLK			
P123/XT1			
P124/XT2/EXCLKS			
P137/INTP0			
P147/CMPCOM/ANI18	11-Y	输入/输出	输入： 通过电阻单独连接至 V _{DD} 或 V _{SS} 。 输出： 保持开路。
P200/TKBO00/INTP22	8-R	输入/输出	输入： 通过电阻单独连接至 V _{DD} 或 V _{SS} 。 输出： 保持开路。 <N 沟开漏时> 设置端口的输出锁存器为 0，并且输出低电平以保持引脚开路。
P201/TKBO01			
P202/TKBO10/(INTP21)			
P203/TKBO11/TKCO02/(INT P20)			
P204/TKBO20/TKCO03			
P205/TKBO21/TKCO04/DALI TxD4			
P206/TKCO05/DALIRxD4/Tx Rx4/INTP23			
RESET	2	输入	直接或通过电阻连接至 V _{DD} 。
REGC	-	-	通过电容(0.47 至 1 μ F)连接至 V _{SS} 。

备注 可以通过设置外围输入/输出重定向寄存器(PIOR1)或输入切换控制寄存器(ISC)来指定上图括号内的功能。

图 2-1. 各引脚的输入/输出电路列表 (1/3)

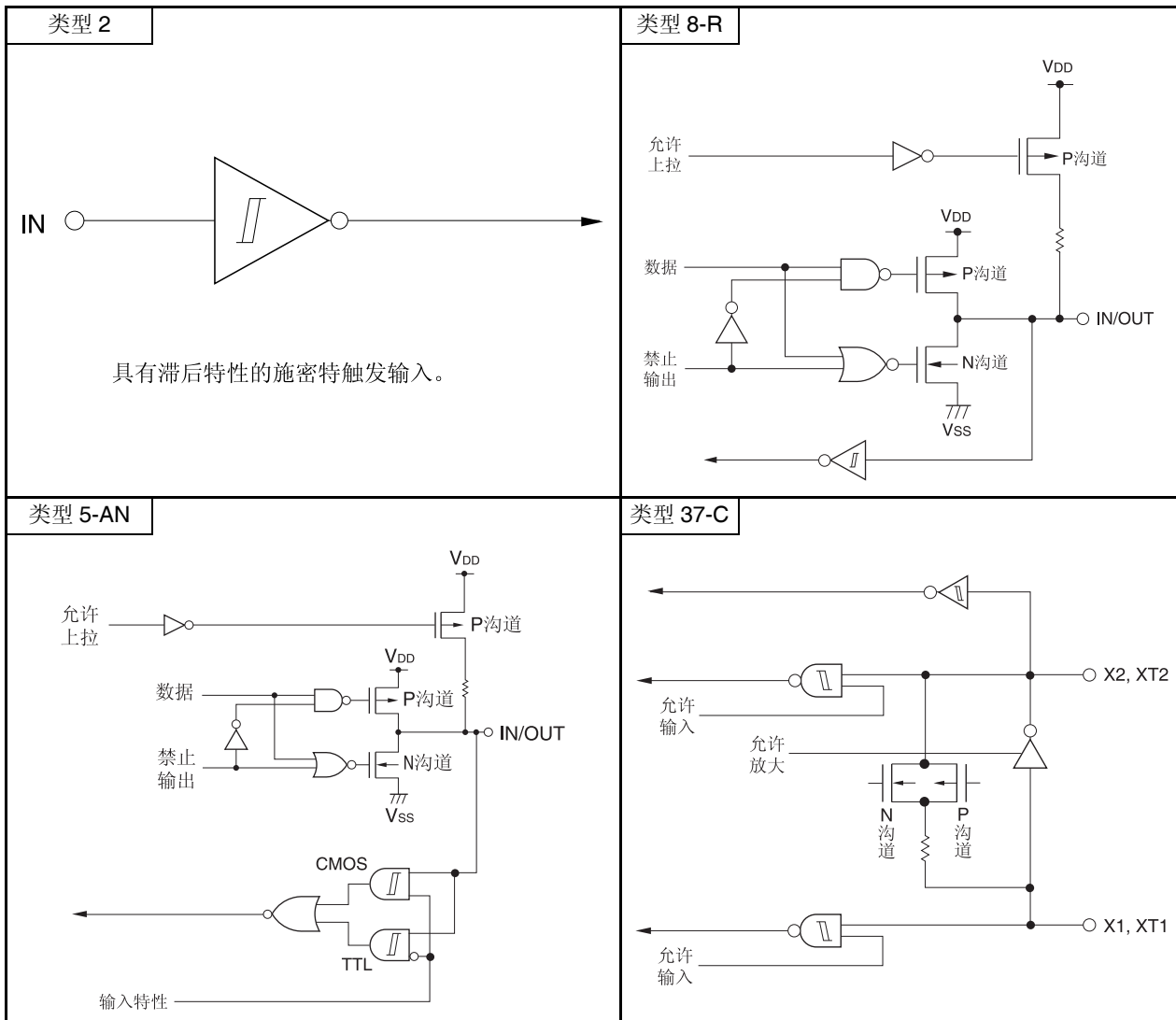


图 2-1. 引脚的输入/输出电路列表 (2/3)

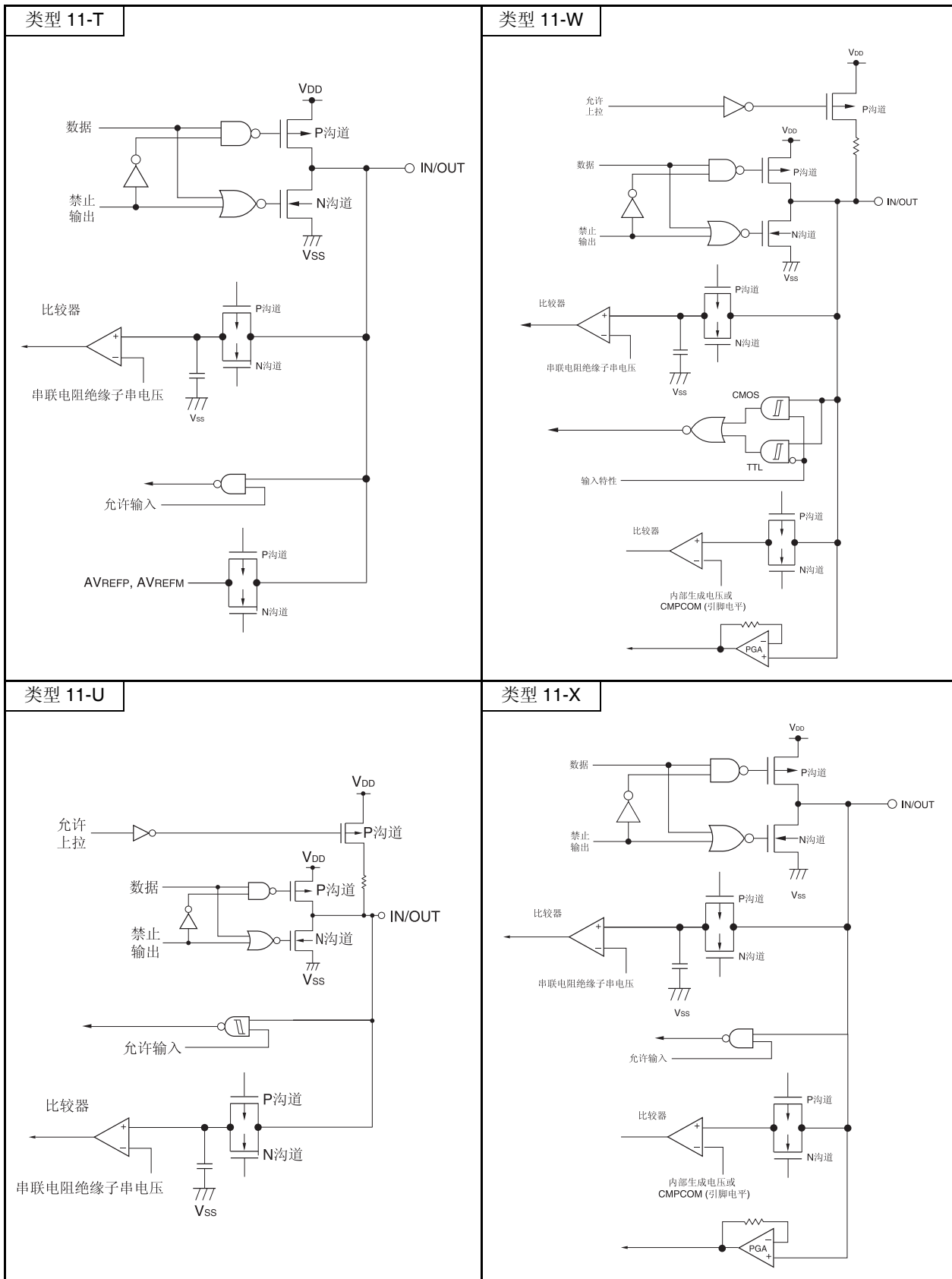
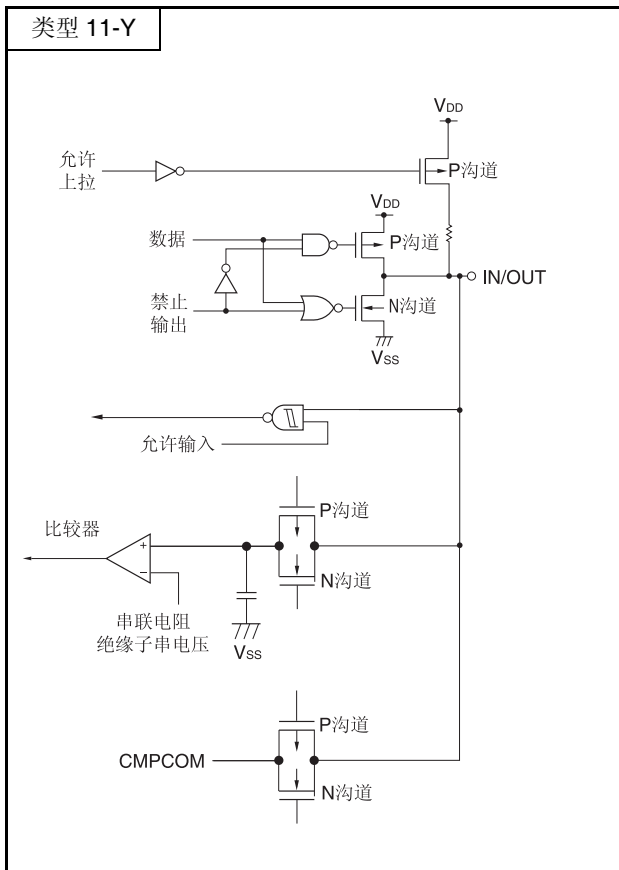


图 2-1. 引脚的输入/输出电路列表 (3/3)

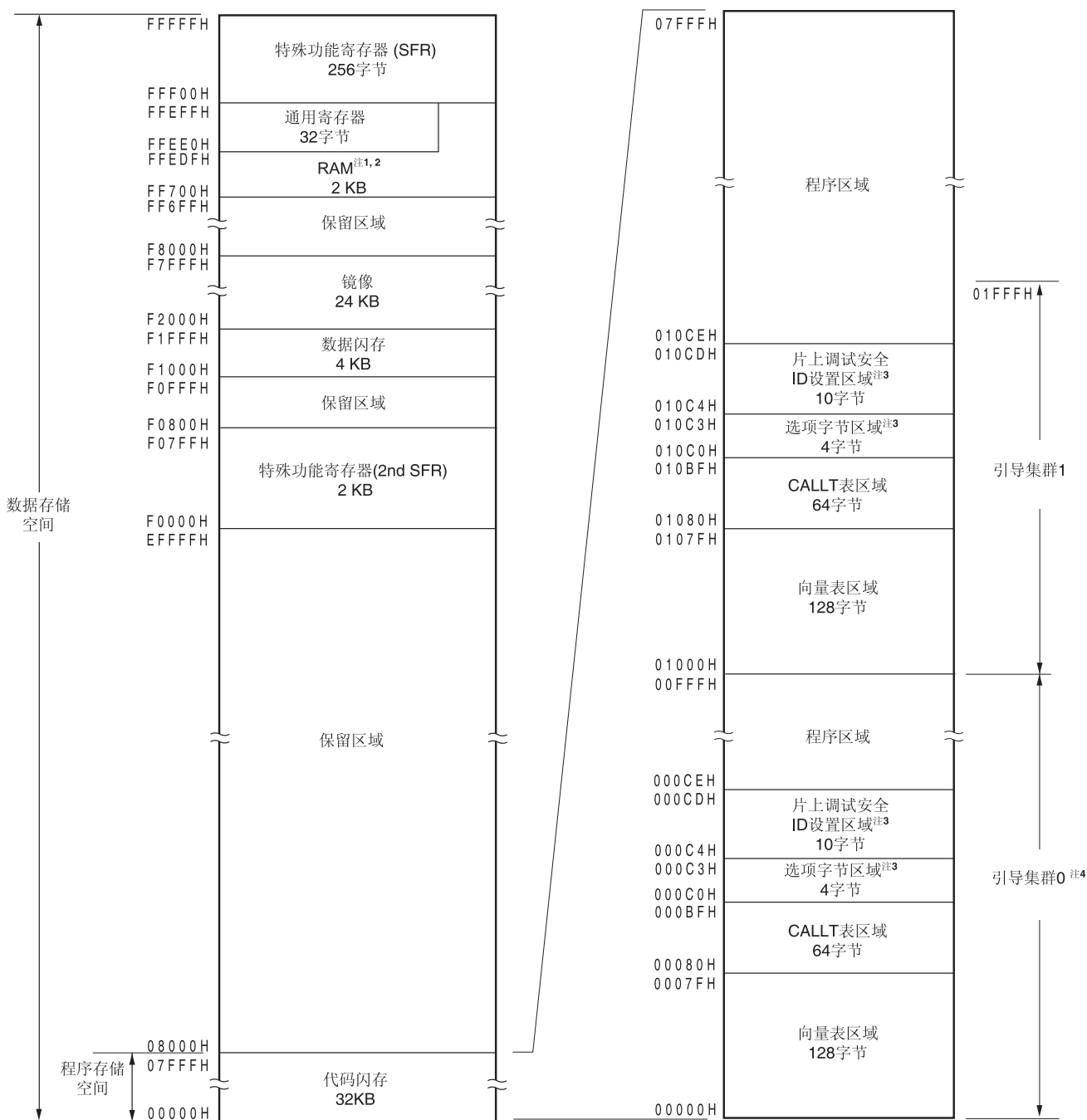


第三章 CPU 架构

3.1 存储空间

RL78/I1A 可以存取 1 MB 的存储空间。存储器映射图如图 3-1 和图 3-2 所示。

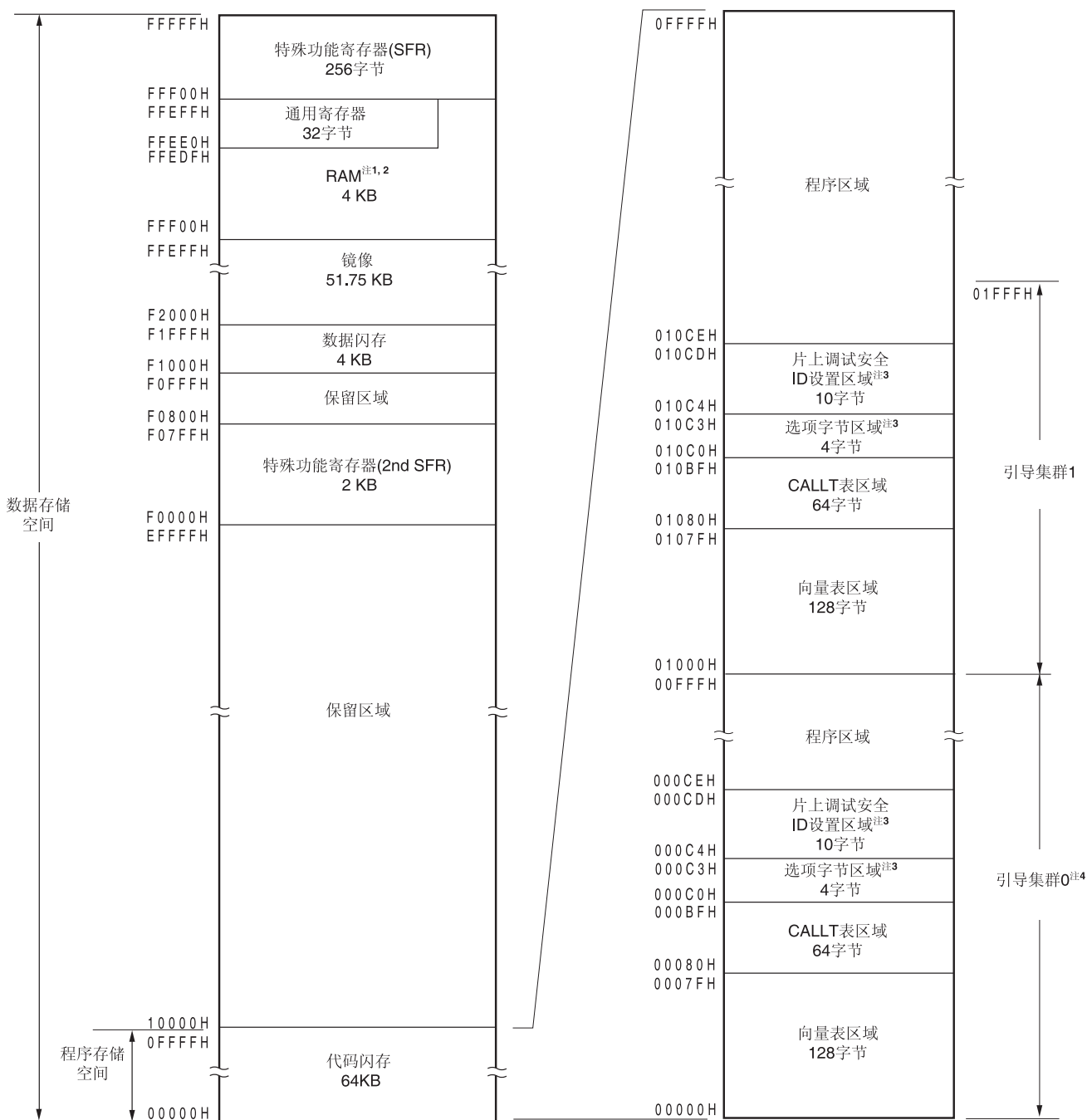
图 3-1. 存储器映射图(R5F1076C, R5F107AC, R5F107BC)



- 注 1. 使用自编程功能和数据闪存功能时，由于 FFE20H 至 FFEDFH 区域供自编程库使用，所以禁止使用该区域。
2. 可以在通用寄存器以外的 RAM 区域中执行指令。
3. 当未使用引导交换功能时： 将选项字节设于 000C0H 至 000C3H，片上调试安全 ID 设于 000C4H 至 000CDH。
当使用引导交换功能时： 将选项字节设于 000C0H 至 000C3H 和 010C0H 至 010C3H，片上调试安全 ID 设于 000C4H 至 000CDH 和 010C4H 至 010CDH。
4. 可以通过安全设置禁止写入引导集群 0 (参阅 28.5 安全设置)。

注意事项 在 RAM 区域中执行指令且允许 RAM 奇偶校验错误产生复位(RPERDIS = 0)的情况下，必须将所使用的 RAM 区域+10 字节的区域初始化。

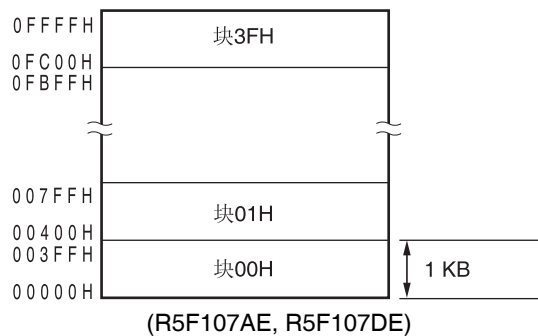
图 3-2. 存储器映射图(R5F107AE, R5F107DE)



- 注 1. 使用自编程序和数据闪存功能时，由于 FFE20H 至 FFEDFH 和 FEF00H 至 FF309H 区域供自编程序使用，所以禁止使用该区域。
- 2. 可以在通用寄存器以外的 RAM 区域中执行指令。
- 3. 当未使用引导交换功能时： 将选项字节设于 000C0H 至 000C3H，片上调试安全 ID 设于 000C4H 至 000CDH。
当使用引导交换功能时： 将选项字节设于 000C0H 至 000C3H 和 010C0H 至 010C3H，片上调试安全 ID 设于 000C4H 至 000CDH 和 010C4H 至 010CDH。
- 4. 可以通过安全设置禁止写入引导集群 0 (参阅 28.5 安全设置)。

注意事项 在 RAM 区域中执行指令且允许 RAM 奇偶校验错误产生复位(RPERDIS = 0)的情况下，必须将所使用的 RAM 区域+10 字节的区域初始化。

备注 闪存分为多个块（1个块= 1 KB）。关于地址值和区域编号，请参阅表 3-1 闪存中地址值和区域编号之间的对应。



闪存中地址值和区域编号之间的对应如下所示。

表 3-1. 闪存中地址值和区域编号之间的对应

地址值	区域编号	地址值	区域编号
00000H至003FFH	00H	08000H至083FFH	20H
00400H至007FFH	01H	08400H至087FFH	21H
00800H至00BFFH	02H	08800H至08BFFH	22H
00C00H至00FFFH	03H	08C00H至08FFFH	23H
01000H至013FFH	04H	09000H至093FFH	24H
01400H至017FFH	05H	09400H至097FFH	25H
01800H至01BFFH	06H	09800H至09BFFH	26H
01C00H至01FFFH	07H	09C00H至09FFFH	27H
02000H至023FFH	08H	0A000H至0A3FFH	28H
02400H至027FFH	09H	0A400H至0A7FFH	29H
02800H至02BFFH	0AH	0A800H至0ABFFH	2AH
02C00H至02FFFH	0BH	0AC00H至0AFFFH	2BH
03000H至033FFH	0CH	0B000H至0B3FFH	2CH
03400H至037FFH	0DH	0B400H至0B7FFH	2DH
03800H至03BFFH	0EH	0B800H至0BBFFH	2EH
03C00H至03FFFH	0FH	0BC00H至0BFFFH	2FH
04000H至043FFH	10H	0C000H至0C3FFH	30H
04400H至047FFH	11H	0C400H至0C7FFH	31H
04800H至04BFFH	12H	0C800H至0CBFFH	32H
04C00H至04FFFH	13H	0CC00H至0CFFFH	33H
05000H至053FFH	14H	0D000H至0D3FFH	34H
05400H至057FFH	15H	0D400H至0D7FFH	35H
05800H至05BFFH	16H	0D800H至0DBFFH	36H
05C00H至05FFFH	17H	0DC00H至0DFFFH	37H
06000H至063FFH	18H	0E000H至0E3FFH	38H
06400H至067FFH	19H	0E400H至0E7FFH	39H
06800H至06BFFH	1AH	0E800H至0EBFFH	3AH
06C00H至06FFFH	1BH	0EC00H至0EFFFH	3BH
07000H至073FFH	1CH	0F000H至0F3FFH	3CH
07400H至077FFH	1DH	0F400H至0F7FFH	3DH
07800H至07BFFH	1EH	0F800H至0FBFFH	3EH
07C00H至07FFFH	1FH	0FC00H至0FFFFH	3FH

备注 R5F1076C, R5F107AC, R5F107BC : 区域编号 00H 至 1FH
R5F107AE, R5F107DE : 区域编号 00H 至 3FH

3.1.1 内部程序存储器空间

内部程序存储器空间中存储程序和表数据。

RL78/I1A 产品中内置的 ROM（闪存）如下所示。

表 3-2. ROM 容量

产品编号	ROM	
	结构	容量
R5F1076C, R5F107AC, R5F107BC	闪存	32768 × 8 位(00000H 至 07FFFH)
R5F107AE, R5F107DE		65536 × 8 位(00000H 至 0FFFFH)

内部程序存储器空间分为以下区域。

(1) 向量表区域

00000H至0007FH的128字节区域作为向量表区域被保留。向量表区存放的是由于复位或各中断请求产生而进行转移时的程序的开始地址。另外，由于向量代码为 2 个字节，因此中断的跳转地址为 00000H 至 0FFFFH 的 64K 地址。在该 16 位地址中，低 8 位存储于偶数地址，高 8 位存储于奇数地址。

若要使用引导交换功能，同时要在 01000H 至 0107FH 区设置一个向量表。

表 3-3. 向量表 (1/2)

向量表地址	中断源	20 引脚	30 引脚	32 引脚	38 引脚
0000H	RESET, POR, LVD, WDT, TRAP, IAW, RPE	√	√	√	√
0004H	INTWDTI	√	√	√	√
0006H	INTLVI	√	√	√	√
0008H	INTP0	√	√	√	√
000EH	INTP3	–	–	–	√
0010H	INTP4	–	√	√	√
001AH	INTDMA0	√	√	√	√
001CH	INTDMA1	√	√	√	√
001EH	INTST0	√	√	√	√
	INTCSI00	–	–	–	√
0020H	INTSR0	√	√	√	√
0022H	INTSRE0	√	√	√	√
	INTTM01H	√	√	√	√
0024H	INTST1	–	√	–	√
0026H	INTSR1	–	√	–	√
0028H	INTSRE1	–	√	–	√
	INTTM03H	√	√	√	√
002AH	INTIICA0	√	√	√	√
002CH	INTTM00	√	√	√	√
002EH	INTTM01	√	√	√	√
0030H	INTTM02	√	√	√	√
0032H	INTTM03	√	√	√	√
0034H	INTAD	√	√	√	√
0036H	INTRTC	√	√	√	√
0038H	INTIT	√	√	√	√
003CH	INTSTDL4	√	√	√	√
003EH	INTSRDL4	√	√	√	√
	INTSREDL4	√	√	√	√
0040H	INTP20	√	√	√	√
	INTP22	√	√	√	√

表 3-3. 向量表 (2/2)

向量表地址	中断源	20 引脚	30 引脚	32 引脚	38 引脚
0042H	INTTM04	√	√	√	√
0044H	INTTM05	√	√	√	√
0046H	INTTM06	√	√	√	√
0048H	INTTM07	√	√	√	√
004AH	INTCMP0	√	√	√	√
004CH	INTCMP1	√	√	√	√
004EH	INTCMP2	√	√	√	√
0050H	INTP9	–	–	√	√
	INTCMP3	√	√	√	√
0052H	INTP10	–	–	√	√
	INTCMP4	–	√	√	√
0054H	INTP11	–	√	√	√
	INTCMP5	–	√	–	√
0056H	INTTMKB0	√	√	√	√
0058H	INTTMKB1	√	√	√	√
005AH	INTTMKB2	–	√	√	√
005CH	INTTMKC0	√	√	√	√
005EH	INTMD	√	√	√	√
0060H	INTP21	√	√	√	√
	INTP23	–	√	√	√
0062H	INTFL	√	√	√	√
007EH	BRK	√	√	√	√

(2) CALLT 指令表区域

00080H 至 000BFH 的 64 字节区域可以存储 2 字节调用指令(CALLT)的子程序入口地址。将子程序入口地址设为 00000H 至 0FFFFH 范围内的一个值（因为地址代码为 2 个字节）。

若要使用引导交换功能，同时要在 01080H 至 010BFH 区设置一个 CALLT 指令表。

(3) 选项字节区域

000C0H 至 000C3H 的 4 字节区域可用作选项字节区域。使用引导交换功能时，在 010C0H 至 010C3H 中也设置选项字节。详情请参阅第二十七章 选项字节。

(4) 片上调试安全 ID 设置区域

000C4H 至 000CDH 和 010C4H 至 010CDH 的 10 字节区域可用作片上调试安全 ID 设置区域。未使用引导交换功能时，将 10 字节的片上调试安全 ID 设于 000C4H 至 000CDH；当使用引导交换功能时，将其设于 000C4H 至 000CDH 以及 010C4H 至 010CDH 区域。详情请参阅第二十九章 On-chip 调试功能。

3.1.2 镜像区域

RL78/I1A 中，00000H 至 0FFFFH 的代码闪存区域被映射到 F0000H 至 FFFFFH 区域（待映射的代码闪存区域通过处理器模式控制寄存器(PMC)设置）。

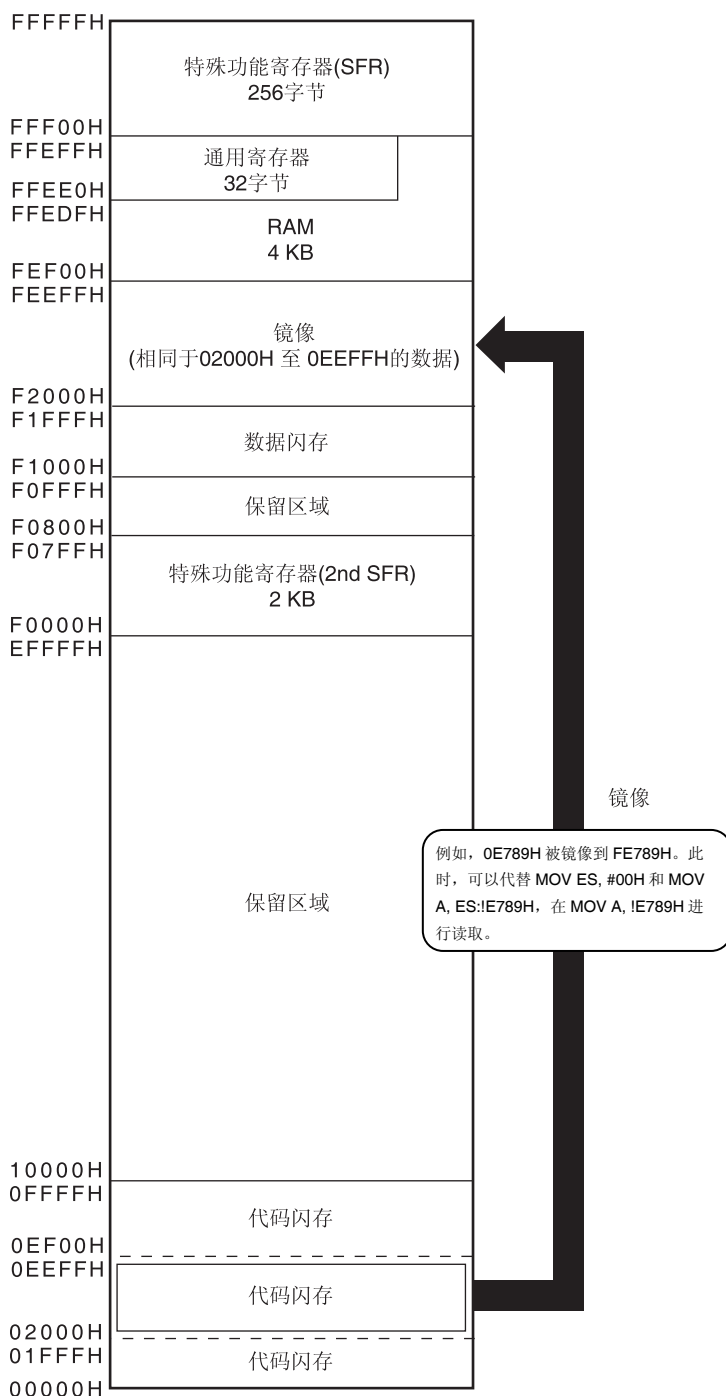
通过从 F0000H 至 FFFFFH 区域读取数据，可以使用不用 ES 寄存器作为操作数的指令，因此，能够用短代码读取代码闪存的内容。然而，不能将代码闪存区域映射到 SFR、扩展 SFR、RAM 和保留区域。

关于每种产品的镜像区域，请参阅 3.1 存储空间。

镜像区域为只读，不得从该区域取指令。

示例如下所示。

示例 R5F107AE, R5F107DE (闪存: 64 KB, RAM: 4 KB)



PMC 寄存器如下页所示。

- **处理器模式控制寄存器(PMC)**

该寄存器设置被镜像到 F0000H 至 FFFFFH 区域的闪存空间。

使用 1 位或 8 位存储器操作指令设置 PMC 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 3-3. 处理器模式控制寄存器(PMC)的格式

地址: FFFFEH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	<0>
PMC	0	0	0	0	0	0	0	MAA

MAA	设置镜像到F0000H至FFFFFH区域的闪存空间
0	00000H至0FFFFH镜像到F0000H至FFFFFH
1	禁止设置

- 注意事项**
1. 必须将该寄存器的位 0(MAA)清除为 0 (初始值)。
 2. 在运行 DMA 控制器之前，仅在初始设定时设置 PMC 寄存器一次。禁止在初始设定以外的时候改写 PMC 寄存器。
 3. 设置 PMC 寄存器后，至少等待一条指令后才可以读取镜像区域。

3.1.3 内部数据存储空间

RL78/I1A 产品中内置了以下 RAM。

表 3-4. 内部 RAM 容量

产品编号	内部 RAM
R5F1076C, R5F107AC, R5F107BC	2048 × 8 位(FE700H 至 FFEFFH)
R5F107AE, R5F107DE	4096 × 8 位(FEF00H 至 FFEFFH)

内部 RAM 可以用作数据区域以及写入并执行指令的程序区域。每组由 8 个 8 位寄存器构成的四组通用寄存器被分配至内部 RAM 区域中的 FFEE0H 至 FFEFFH 的 32 位区域。然而，不能通过通用寄存器执行指令。

内部 RAM 可以用作堆栈存储器。

- 注意事项**
1. 禁止利用通用寄存器（FFEE0H 至 FFEFFH）空间来获取指令，也不得将其用作堆栈区域。
 2. 使用自编程功能和数据闪存功能时，FFE20H 至 FFEFFH 区域不能用作堆栈存储器。此外，FEF00H 至 FF309H 区域也不能与 R5F107AE 和 R5F107DE 一起使用。

3.1.4 特殊功能寄存器 (SFR) 区域

片上外围硬件的特殊功能寄存器(SFR)被分配到 FFF00H 至 FFFFFH 区域(参阅 3.2.4 特殊功能寄存器(SFR) 中的表 3-5)。

注意事项 不得存取未分配 SFR 的地址。

3.1.5 扩展特殊功能寄存器 (2nd SFR: 第二特殊功能寄存器) 区域

片上外围硬件的扩展特殊功能寄存器(2nd SFR)被分配到 F0000H 至 F07FFH 区域(参阅 3.2.5 扩展特殊功能寄存器 (2nd SFR: 第二特殊功能寄存器) 中的表 3-6)。

SFR 区域 (FFF00H 至 FFFFFH) 以外的 SFR 被分配到该区域。存取扩展 SFR 区域的指令比存取 SFR 区域的指令长 1 个字节。

注意事项 1. 不得存取未分配扩展 SFR 的地址。

2. 访问分配到扩展 SFR (2nd SFR)区域中的 F0500H 至 F06FFH 地址的寄存器时, CPU 不开始处理下一个指令, 而是变为等待状态。因此, 当发生这种等待状态时, 执行指令所需的时钟数就会增长等待时钟数。关于详情, 请参阅第三十四章 关于等待的注意事项。

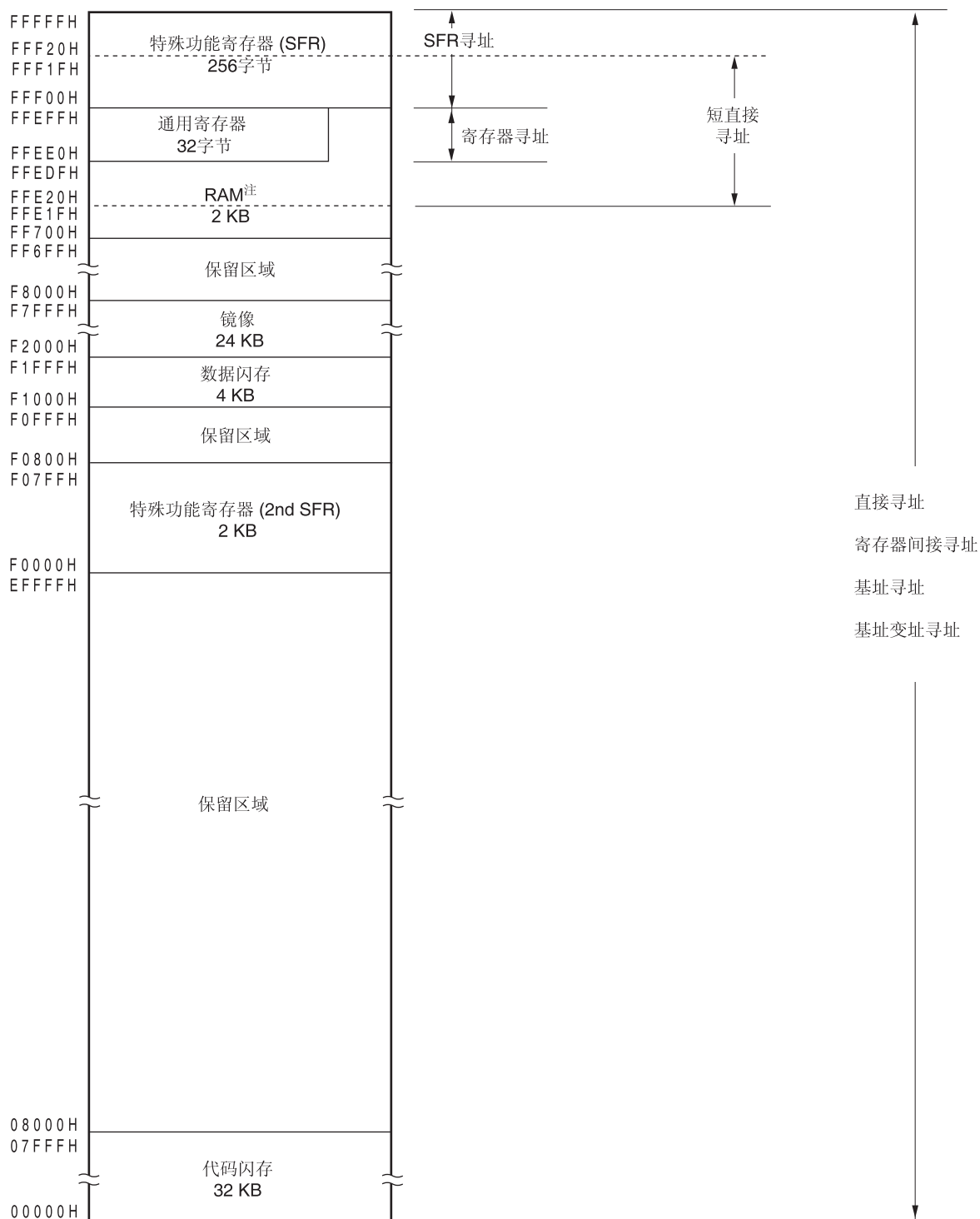
3.1.6 数据存储器寻址

寻址是一种方法，用于指定下一步将执行的指令的地址，或者执行指令时相关的寄存器或存储器等的地址。

考虑到可操作性等，RL78/I1A 中提供了丰富的寻址方式，用于寻址执行指令时作为操作对象的存储器。尤其在具有数据存储器的区域，可以使用配合特殊功能寄存器(SFR)或通用寄存器等的功能而设计的特殊的寻址方法。数据存储器和寻址之间的对应如图 3-4 和图 3-5 所示。

每种寻址的详情请参阅 **3.4 处理数据地址的寻址**。

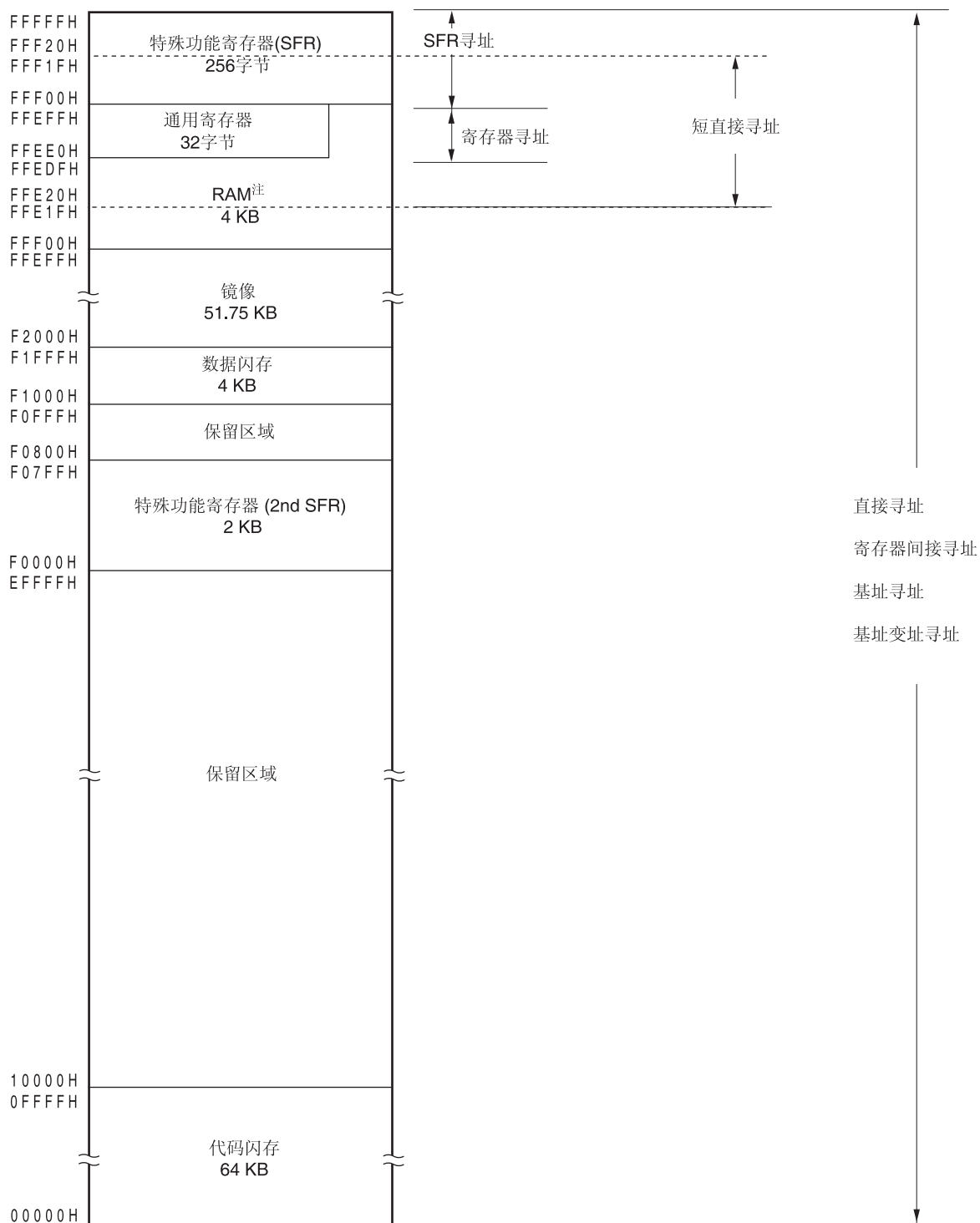
图 3-4. 数据存储器和寻址之间的对应 (R5F1076C, R5F107AC, R5F107BC)



注 在使用自编程功能和数据闪存功能时，由于 FFE20H 至 FFEDFH 区域供自编程库使用，所以禁止使用该区域。

注意事项 在 RAM 区域中执行指令且允许 RAM 奇偶校验错误产生复位 (RPERDIS = 0)的情况下，必须将所使用的 RAM 区域+10 字节的区域初始化。

图 3-5. 数据存储器和寻址之间的对应 (R5F107AE, R5F107DE)



注 在使用自编程功能和数据闪存功能时，由于 FFE20H 至 FFEDFH 和 FEF00H 至 FF309H 区域供自编程库使用，所以禁止使用该区域。

注意事项 在 RAM 区域中执行指令且允许 RAM 奇偶校验错误产生复位(RPERDIS = 0)的情况下，必须将所使用的 RAM 区域+10 字节的区域初始化。

3.2 处理器寄存器

RL78/I1A 产品中内置以下处理器寄存器。

3.2.1 控制寄存器

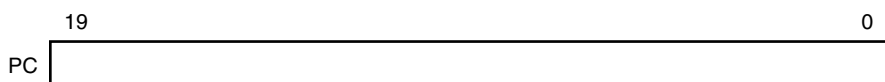
控制寄存器用于控制程序时序、状态和堆栈存储器。控制寄存器由程序计数器(PC)、程序状态字(PSW)和堆栈指针(SP)组成。

(1) 程序计数器 (PC)

程序计数器是一个 20 位寄存器，用于存储即将执行的下个程序的地址信息。

在通常操作中，PC 根据待获取的指令的字节数自动递增。执行分支指令时，设置立即数和寄存器内容。产生复位信号后，地址 0000H 和 0001H 处的复位向量表的值被设置至程序计数器。

图 3-6. 程序计数器的格式

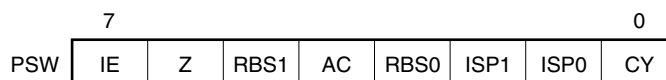


(2) 程序状态字 (PSW)

程序状态字为 8 位寄存器，由通过执行指令来置位/复位的多个标志组成。

程序状态字内容在受理向量中断请求被受理或 PUSH PSW 指令被执行时存储于堆栈区中，并在执行 RETB、RETI 和 POP PSW 指令时恢复。产生复位信号后，PSW 寄存器被设置为 06H。

图 3-7. 程序状态字的格式



(a) 中断允许标志 (IE)

该标志控制 CPU 的中断请求受理操作。

当其为 0 时，IE 标志设置为中断禁止(DI)状态，禁止所有可屏蔽中断请求。

当其为 1 时，IE 标志设置为中断允许(EI)状态，通过受理中断的优先级标志 (ISP1、ISP0)、各中断源的中断屏蔽标志和优先级指定标志来控制中断请求的受理。

IE 标志在执行 DI 指令或中断受理时复位(0)，在执行 EI 指令时被置位(1)。

(b) 零标志 (Z)

当运算结果为零时，该标志被置位(1)。其他情况下被复位(0)。

(c) 寄存器组选择标志 (RBS0, RBS1)

用于从 4 个寄存器组中选出 1 个的 2 位标志。

通过执行 SEL RBn 指令而选择的寄存器组的 2 位信息存储于该标志中。

(d) 辅助进位标志 (AC)

如果运算结果在位 3 进位或在位 3 借位，则该标志被置位(1)。其他情况下被复位(0)。

(e) 受理中断的优先级标志 (ISP1, ISP0)

可以处理的可屏蔽向量中断的优先级由该标志管理。优先级指定标志寄存器 (PRn0L、PRn0H、PRn1L、PRn1H、PRn2L、PRn2H) (参阅 20.3 (3))所指定的低于 ISP0 和 ISP1 标志值的向量中断请求禁止受理。实际上是否受理中断请求由中断允许标志(IE)控制。

备注 n = 0, 1

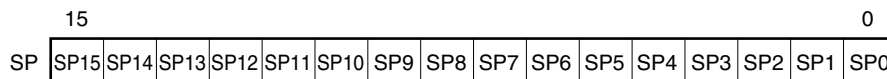
(f) 进位标志 (CY)

该标志存储执行加/减指令时的上溢和下溢。执行循环指令时存储移出值，执行位操作指令时用作位累加器。

(3) 堆栈指针 (SP)

该 16 位寄存器用于保存存储器堆栈区域的起始地址。仅限内部 RAM 区域可设为堆栈区域。

图 3-8. 堆栈指针的格式



SP 在写入（压栈）堆栈存储器之前递减，在从堆栈存储器读取（出栈）之后递增。

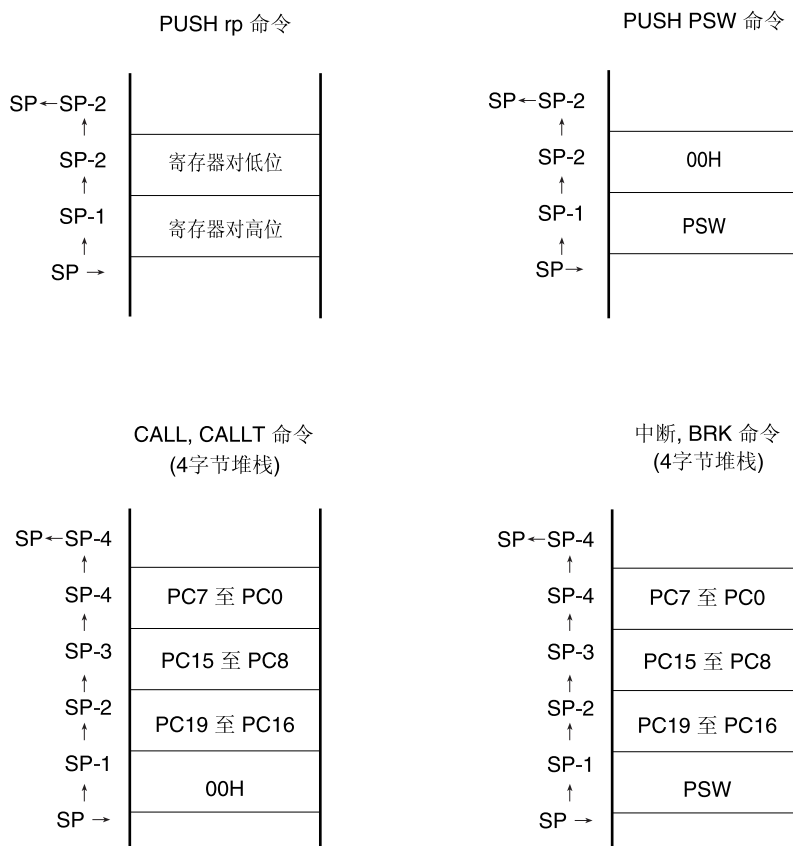
各堆栈操作所保存的数据如图 3-9 所示。

注意事项 1. 由于产生复位信号时会使 SP 的内容不定，因此必须在使用该堆栈之前初始化 SP。

2. 禁止将通用寄存器 (FFEE0H 至 FFEFFH) 空间用作堆栈区。

3. 使用自编程功能和数据闪存功能时，FFE20H 至 FFEFFH 区域不能用作堆栈存储器。此外，FEF00H 至 FF309H 区域也不能与 R5F107AE 和 R5F107DE 一起使用。

图 3-9. 保存到堆栈的数据



3.2.2 通用寄存器

通用寄存器被分配到数据存储器的特定地址 (FFEE0H 至 FFEFFH)。通用寄存器由 4 组寄存器构成, 每组包括 8 个 8 位寄存器 (X、A、C、B、E、D、L 和 H)。

每个寄存器可以作为 1 个 8 位寄存器使用, 也可以将 2 个 8 位寄存器组合起来作为 1 个 16 位寄存器 (AX、BC、DE 和 HL) 使用。

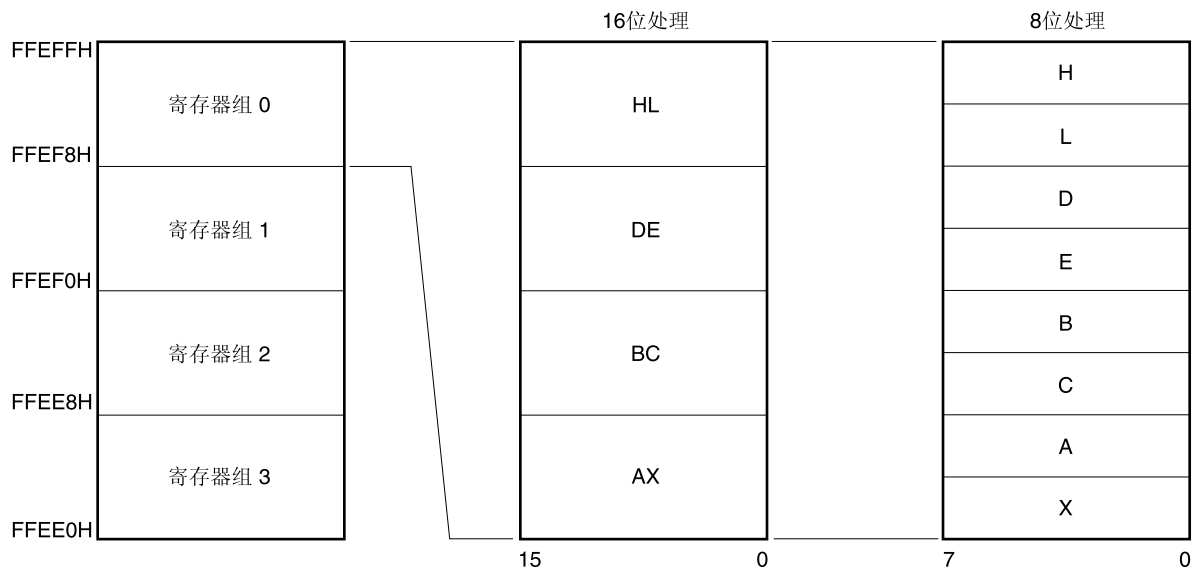
这些寄存器可以用功能名称 (X、A、C、B、E、D、L、H、AX、BC、DE 和 HL) 以及绝对名称 (R0 至 R7 及 RP0 至 RP3) 来描述。

执行指令时所用的寄存器组通过 CPU 控制指令 (SEL RBn) 设置。由于其构成为 4 个寄存器组, 可以将用于正常处理的寄存器和处理中断时所用的寄存器切换, 从而创建高效率的程序。

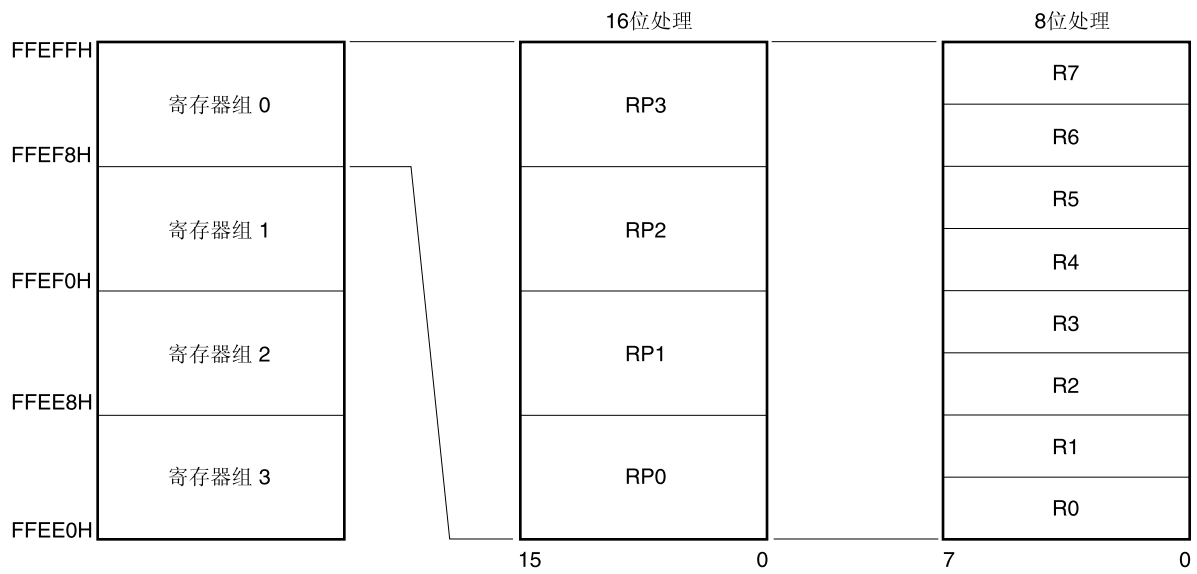
- 注意事项**
1. 禁止利用通用寄存器 (FFEE0H 至 FFEFFH) 空间来获取指令, 也不得将其用作堆栈区域。
 2. 使用自编程功能和数据闪存功能时, FFE20H 至 FFEFFH 区域不能用作堆栈存储器。此外, FEF00H 至 FF309H 区域也不能与 R5F107AE 和 R5F107DE 一起使用。

图 3-10. 通用寄存器的配置

(a) 功能名称



(b) 绝对名称



3.2.3 ES和CS寄存器

ES 寄存器用于数据存取，CS 寄存器用于在执行分支指令时指定高位地址。

ES 寄存器复位后的初始值为 0FH，CS 寄存器复位后的初始值为 00H。

图 3-11. ES 和 CS 寄存器的配置

	7	6	5	4	3	2	1	0
ES	0	0	0	0	ES3	ES2	ES1	ES0
CS	7	6	5	4	3	2	1	0
	0	0	0	0	CS3	CP2	CP1	CP0

3.2.4 特殊功能寄存器(SFR)

与通用寄存器不同，每个 SFR 都具有特殊功能。

SFR 空间被分配到 FFF00H 至 FFFFFH 区域。

通过运算、传送和位操作指令，可以像操作通用寄存器一样操作 SFR。可操作的单位（1、8、16）因 SFR 的不同而异。

操作单位的指定方法如下所示。

- 1 位操作

通过汇编程序的保留符号指定位操作指令的操作数(sfr.bit)。该操作也可以通过地址指定。

- 8 位操作

通过汇编程序的保留符号指定 8 位操作指令的操作数(sfr)。该操作也可以通过地址指定。

- 16 位操作

通过汇编程序的保留符号指定 16 位操作指令的操作数(sfrp)所保留的符号。当指定地址时，请描述为偶数地址。

表 3-5 为 SFR 列表。表中所列项目的含义如下所示。

- 符号

表示特殊功能寄存器地址的符号。在汇编程序中作为保留字，在编译程序中通过 #pragma sfr 指令被定义为 sfr 变量。当使用汇编程序、调试器和仿真器时，可以作为指令的操作数。

- R/W

表示相对应的 SFR 能否读取或写入。

R/W: 可读/写

R: 只读

W: 只写

- 可操作位单元

“√”表示可操作的位单位（1、8、16）。“-”表示不可以操作的位单位。

- 复位后

表示产生复位信号后各个寄存器的状态。

注意事项 不得存取未分配扩展 SFR 的地址。

备注 关于扩展 SFR (2nd SFR)，请参阅 3.2.5 扩展特殊功能寄存器 (2nd SFR: 第二特殊功能寄存器)。

表 3-5. SFR 列表 (1/4)

地址	特殊功能寄存器(SFR)名称	符号		R/W	可操作位范围			复位后
					1 位	8 位	16 位	
FFF00H	端口寄存器 0	P0		R/W	√	√	–	00H
FFF01H	端口寄存器 1	P1		R/W	√	√	–	00H
FFF02H	端口寄存器 2	P2		R/W	√	√	–	00H
FFF03H	端口寄存器 3	P3		R/W	√	√	–	00H
FFF04H	端口寄存器 4	P4		R/W	√	√	–	00H
FFF07H	端口寄存器 7	P7		R/W	√	√	–	00H
FFF0CH	端口寄存器 12	P12		R/W	√	√	–	不定
FFF0DH	端口寄存器 13	P13		R/W	√	√	–	不定
FFF0EH	端口寄存器 14	P14		R/W	√	√	–	00H
FFF10H	串行数据寄存器 00	TXD0/ SIO00	R/W	R/W	–	√	√	0000H
FFF11H		–			–	–		
FFF12H	串行数据寄存器 01	RXD0	R/W	R/W	–	√	√	0000H
FFF13H		–			–	–		
FFF18H	定时器数据寄存器 00	TDR00		R/W	–	–	√	0000H
FFF19H					–	–	–	
FFF1AH	定时器数据寄存器 01	TDR01L	R/W	R/W	–	√	√	00H
FFF1BH		TDR01H			–	√	–	00H
FFF1EH	10 位 A/D 转换结果寄存器	ADCR		R	–	–	√	0000H
FFF1FH	8 位 A/D 转换结果寄存器	ADCRH		R	–	√	–	00H
FFF20H	端口模式寄存器 0	PM0		R/W	√	√	–	FFH
FFF21H	端口模式寄存器 1	PM1		R/W	√	√	–	FFH
FFF22H	端口模式寄存器 2	PM2		R/W	√	√	–	FFH
FFF23H	端口模式寄存器 3	PM3		R/W	√	√	–	FFH
FFF24H	端口模式寄存器 4	PM4		R/W	√	√	–	FFH
FFF27H	端口模式寄存器 7	PM7		R/W	√	√	–	FFH
FFF2CH	端口模式寄存器 12	PM12		R/W	√	√	–	FFH
FFF2EH	端口模式寄存器 14	PM14		R/W	√	√	–	FFH
FFF30H	A/D 转换器模式寄存器 0	ADM0		R/W	√	√	–	00H
FFF31H	模拟输入通道选择寄存器	ADS		R/W	√	√	–	00H
FFF32H	A/D 转换器模式寄存器 1	ADM1		R/W	√	√	–	00H
FFF38H	外部中断上升沿允许寄存器 0	EGP0		R/W	√	√	–	00H
FFF39H	外部中断下降沿允许寄存器 0	EGN0		R/W	√	√	–	00H
FFF3AH	外部中断上升沿允许寄存器 1	EGP1		R/W	√	√	–	00H
FFF3BH	外部中断下降沿允许寄存器 1	EGN1		R/W	√	√	–	00H

表 3-5. SFR 列表 (2/4)

地址	特殊功能寄存器(SFR)名称	符号		R/W	可操作位范围			复位后
					1 位	8 位	16 位	
FFF44H	串行数据寄存器 02	TXD1	R/W	R/W	-	√	√	0000H
FFF45H		-			-	-		
FFF46H	串行数据寄存器 03	RXD1	R/W	R/W	-	√	√	0000H
FFF47H		-			-	-		
FFF50H	IICA 移位寄存器 0	IICA0		R/W	-	√	-	00H
FFF51H	IICA 状态寄存器 0	IICS0		R	√	√	-	00H
FFF52H	IICA 标志寄存器 0	IICF0		R/W	√	√	-	00H
FFF64H	定时器数据寄存器 02	TDR02		R/W	-	-	√	0000H
FFF65H					-	-	-	
FFF66H	定时器数据寄存器 03	TDR03L	R/W	R/W	-	√	√	00H
FFF67H		TDR03H			-	√	00H	
FFF68H	定时器数据寄存器 04	TDR04		R/W	-	-	√	0000H
FFF69H					-	-	-	
FFF6AH	定时器数据寄存器 05	TDR05		R/W	-	-	√	0000H
FFF6BH					-	-	-	
FFF6CH	定时器数据寄存器 06	TDR06		R/W	-	-	√	0000H
FFF6DH					-	-	-	
FFF6EH	定时器数据寄存器 07	TDR07		R/W	-	-	√	0000H
FFF6FH					-	-	-	
FFF90H	间隔定时器控制寄存器	ITMC		R/W	-	-	√	0FFFH
FFF91H					-	-	-	
FFF92H	秒计数寄存器	SEC		R/W	-	√	-	00H
FFF93H	分钟计数寄存器	MIN		R/W	-	√	-	00H
FFF94H	小时计数寄存器	HOUR		R/W	-	√	-	12H [*]
FFF95H	星期计数寄存器	WEEK		R/W	-	√	-	00H
FFF96H	日计数寄存器	DAY		R/W	-	√	-	01H
FFF97H	月计数寄存器	MONTH		R/W	-	√	-	01H
FFF98H	年计数寄存器	YEAR		R/W	-	√	-	00H
FFF99H	时钟误差修正寄存器	SUBCUD		R/W	-	√	-	00H
FFF9AH	报警分钟寄存器	ALARMWMM		R/W	-	√	-	00H
FFF9BH	报警小时寄存器	ALARMWHH		R/W	-	√	-	12H
FFF9CH	报警星期寄存器	ALARMWW		R/W	-	√	-	00H
FFF9DH	实时时钟控制寄存器 0 (RTCC0)	RTCC0		R/W	√	√	-	00H
FFF9EH	实时时钟控制寄存器 1 (RTCC1)	RTCC1		R/W	√	√	-	00H

注 如果复位后 AMPM 位（实时时钟控制寄存器 0(RTCC0)的位 3）设为 1，则该寄存器的值为 00H。

表 3-5. SFR 列表 (3/4)

地址	特殊功能寄存器(SFR)名称	符号	R/W	可操作位范围			复位后
				1 位	8 位	16 位	
FFFA0H	时钟操作模式控制寄存器	CMC	R/W	-	√	-	00H
FFFA1H	时钟操作状态控制寄存器	CSC	R/W	√	√	-	C0H
FFFA2H	振荡稳定时间计数器状态寄存器	OSTC	R	√	√	-	00H
FFFA3H	振荡稳定时间选择寄存器	OSTS	R/W	-	√	-	07H
FFFA4H	系统时钟控制寄存器	CKC	R/W	√	√	-	00H
FFFA8H	复位控制标志寄存器	RESF	R	-	√	-	不定 ^{注1}
FFFA9H	电压检测寄存器	LVIM	R/W	√	√	-	00H ^{注2}
FFFAAH	电压检测电平寄存器	LVIS	R/W	√	√	-	00H/01H/81H ^{注3}
FFFABH	看门狗定时器允许寄存器	WDTE	R/W	-	√	-	1AH/9AH ^{注4}
FFFACH	CRC 输入寄存器	CRCIN	R/W	-	√	-	00H
FFFB0H	DMA SFR 地址寄存器 0	DSA0	R/W	-	√	-	00H
FFFB1H	DMA SFR 地址寄存器 1	DSA1	R/W	-	√	-	00H
FFFB2H	DMA RAM 地址寄存器 0L	DRA0L	R/W	R/W	-	√	√
FFFB3H	DMA RAM 地址寄存器 0H	DRA0H		R/W	-	√	00H
FFFB4H	DMA RAM 地址寄存器 1L	DRA1L	R/W	R/W	-	√	√
FFFB5H	DMA RAM 地址寄存器 1H	DRA1H		R/W	-	√	00H
FFFB6H	DMA 字节计数寄存器 0L	DBC0L	R/W	R/W	-	√	√
FFFB7H	DMA 字节计数寄存器 0H	DBC0H		R/W	-	√	00H
FFFB8H	DMA 字节计数寄存器 1L	DBC1L	DBC1	R/W	-	√	√
FFFB9H	DMA 字节计数寄存器 1H	DBC1H		R/W	-	√	00H
FFFBABH	DMA 模式控制寄存器 0	DMC0	R/W	R/W	√	-	00H
FFFBABH	DMA 模式控制寄存器 1	DMC1	R/W	R/W	√	-	00H
FFFBABH	DMA 操作控制寄存器 0	DRC0	R/W	R/W	√	-	00H
FFFBABH	DMA 操作控制寄存器 1	DRC1	R/W	R/W	√	-	00H
FFFC0H	-	PFCMD ^{注5}	-	-	-	-	不定
FFFC2H	-	PFS ^{注5}	-	-	-	-	不定
FFFC4H	-	FLPMC ^{注5}	-	-	-	-	不定

注1. RESF 寄存器的复位值因复位源而异。

2. LVIM 寄存器的复位值因复位源而异。

3. LVIS 寄存器的复位值因复位源和选项字节的设置而异。

4. WDTE 寄存器的复位值决定于选项字节的设置。

5. 因为 SFR 将被用于自编程库，所以请勿直接对其进行操作。

表 3-5. SFR 列表 (4/4)

地址	特殊功能寄存器(SFR)名称	符号		R/W	可操作位范围			复位后
					1 位	8 位	16 位	
FFFD0H	中断请求标志寄存器 2L	IF2L	IF2	R/W	√	√	√	00H
FFFD1H	中断请求标志寄存器 2H	IF2H		R/W	√	√		00H
FFFD4H	中断屏蔽标志寄存器 2L	MK2L	MK2	R/W	√	√	√	FFH
FFFD5H	中断屏蔽标志寄存器 2H	MK2H		R/W	√	√		FFH
FFFD8H	优先选择标志寄存器 02L	PR02L	PR02	R/W	√	√	√	FFH
FFFD9H	优先选择标志寄存器 00L	PR02H		R/W	√	√		FFH
FFFDCH	优先选择标志寄存器 02L	PR12L	PR12	R/W	√	√	√	FFH
FFDDH	优先选择标志寄存器 02L	PR12H		R/W	√	√		FFH
FFFE0H	中断请求标志寄存器 0L	IF0L	IF0	R/W	√	√	√	00H
FFFE1H	中断请求标志寄存器 0L	IF0H		R/W	√	√		00H
FFFE2H	中断请求标志寄存器 0L	IF1L	IF1	R/W	√	√	√	00H
FFFE3H	中断请求标志寄存器 0L	IF1H		R/W	√	√		00H
FFFE4H	中断屏蔽标志寄存器 0L	MK0L	MK0	R/W	√	√	√	FFH
FFFE5H	中断屏蔽标志寄存器 2L	MK0H		R/W	√	√		FFH
FFFE6H	中断屏蔽标志寄存器 2L	MK1L	MK1	R/W	√	√	√	FFH
FFFE7H	中断屏蔽标志寄存器 2L	MK1H		R/W	√	√		FFH
FFFE8H	优先选择标志寄存器 00L	PR00L	PR00	R/W	√	√	√	FFH
FFFE9H	优先选择标志寄存器 02L	PR00H		R/W	√	√		FFH
FFFEAH	优先选择标志寄存器 02L	PR01L	PR01	R/W	√	√	√	FFH
FFFEBH	优先选择标志寄存器 02L	PR01H		R/W	√	√		FFH
FFFECH	优先选择标志寄存器 02L	PR10L	PR10	R/W	√	√	√	FFH
FF FEDH	优先选择标志寄存器 02L	PR10H		R/W	√	√		FFH
FFFE EH	优先选择标志寄存器 02L	PR11L	PR11	R/W	√	√	√	FFH
FFFE FH	优先选择标志寄存器 02L	PR11H		R/W	√	√		FFH
FFFF0H	乘法/除法数据寄存器 A (L)	MDAL		R/W	-	-	√	0000H
FFFF1H								
FFFF2H	乘法/除法数据寄存器 A (H)	MDAH		R/W	-	-	√	0000H
FFFF3H								
FFFF4H	乘法/除法数据寄存器 B (H)	MDBH		R/W	-	-	√	0000H
FFFF5H								
FFFF6H	乘法/除法数据寄存器 B (L)	MDBL		R/W	-	-	√	0000H
FFFF7H								
FFFFEH	处理器模式控制寄存器	PMC		R/W	√	√	-	00H

备注 关于扩展 SFR (2nd SFR), 请参阅表 3-6 扩展 SFR (2nd SFR) 列表。

3.2.5 扩展特殊功能寄存器 (2nd SFR: 第二特殊功能寄存器)

与通用寄存器不同，每个扩展 SFR (2nd SFR)都具有特殊功能。

扩展 SFR 空间为 F0000H 至 F07FFH 区域。将 SFR 区域 (FFF00H 至 FFFFFH 区) 以外的 SFR 被分配到该区域。存取扩展 SFR 区域的指令比存取 SFR 区域的指令长 1 个字节。

通过运算、传送和位操作指令，可以像操作通用寄存器一样操作扩展 SFR。可操作的位 (1、8、16) 因 SFR 的不同而异。

操作位单位的指定方法如下所示。

- 1 位操作

通过汇编程序的保留符号指定位操作指令的操作数(!addr16.bit)。该操作也可以通过地址指定。

- 8 位操作

通过汇编程序的保留符号指定 8 位操作指令的操作数(!addr16)。该操作也可以通过地址指定。

- 16 位操作

通过汇编程序的保留符号指定 16 位操作指令的操作数(!addr16)。当指定地址时，请描述为偶数地址。

表 3-6 显示了扩展 SFR 的列表。表中所列项目的含义如下所示。

- 符号

表示扩展 SFR 地址的符号。在汇编程序中作为保留字，在编译程序中通过 #pragma sfr 指令被定义为 sfr 变量。使用汇编程序、调试器和仿真器时，可以描述为指令的操作数。

- R/W

表示相对应的扩展 SFR 能否读取或写入。

R/W: 可读/写

R: 只读

W: 只写

- 可操作位单元

“√”表示可操作的单位 (1、8、16)。“-”表示不可以操作的单位。

- 复位后

表示产生复位信号后各个寄存器的状态。

注意事项 1. 不得存取未分配扩展 SFR 的地址。

2. 访问分配到扩展 SFR (2nd SFR)区域中的 F0500H 至 F06FFH 地址的寄存器时，CPU 不开始处理下一个指令，而是变为等待状态。因此，当发生这种等待状态时，执行指令所需的时钟数就会增长等待时钟数。关于详情，请参阅第三十四章 关于等待的注意事项。

备注 关于 SFR 区域中的 SFR，请参阅 3.2.4 特殊功能寄存器 (SFR)。

表 3-6. 扩展 SFR (2nd SFR)列表 (1/12)

地址	特殊功能寄存器(SFR)名称	符号	R/W	可操作位范围			复位后
				1 位	8 位	16 位	
F0010H	A/D 转换器模式寄存器 2	ADM2	R/W	√	√	–	00H
F0011H	转换结果比较上限值设置寄存器	ADUL	R/W	–	√	–	FFH
F0012H	转换结果比较下限值设置寄存器	ADLL	R/W	–	√	–	00H
F0013H	A/D 测试寄存器	ADTES	R/W	–	√	–	00H
F0030H	上拉电阻选择寄存器 0	PU0	R/W	√	√	–	00H
F0031H	上拉电阻选择寄存器 1	PU1	R/W	√	√	–	00H
F0033H	上拉电阻选择寄存器 3	PU3	R/W	√	√	–	00H
F0034H	上拉电阻选择寄存器 4	PU4	R/W	√	√	–	01H
F0037H	上拉电阻选择寄存器 7	PU7	R/W	√	√	–	00H
F003CH	上拉电阻选择寄存器 12	PU12	R/W	√	√	–	00H
F003EH	上拉电阻选择寄存器 14	PU14	R/W	√	√	–	00H
F0040H	端口输入模式寄存器 0	PIM0	R/W	√	√	–	00H
F0041H	端口输入模式寄存器 1	PIM1	R/W	√	√	–	00H
F0050H	端口输出模式寄存器 0	POM0	R/W	√	√	–	00H
F0051H	端口输出模式寄存器 1	POM1	R/W	√	√	–	00H
F0060H	端口模式控制寄存器 0	PMC0	R/W	√	√	–	FFH
F006CH	端口模式控制寄存器 12	PMC12	R/W	√	√	–	FFH
F006EH	端口模式控制寄存器 14	PMC14	R/W	√	√	–	FFH
F0070H	噪声滤波器允许寄存器 0	NFEN0	R/W	√	√	–	00H
F0071H	噪声滤波器允许寄存器 1	NFEN1	R/W	√	√	–	00H
F0073H	输入切换控制寄存器	ISC	R/W	√	√	–	00H
F0074H	定时器输入选择寄存器 0	TIS0	R/W	–	√	–	00H
F0076H	A/D 端口配置寄存器	ADPC	R/W	–	√	–	00H
F0078H	非法存储器存取检测控制寄存器 0	IAWCTL0	R/W	–	√	–	00H
F0090H	数据闪存控制寄存器	DFLCTL	R/W	√	√	–	00H
F00A0H	高速片上振荡器调节寄存器	HIOTRM	R/W	–	√	–	注

注 复位值因芯片而异。

表 3-6. 扩展 SFR (2nd SFR)列表 (2/12)

地址	特殊功能寄存器(SFR)名称	符号		R/W	可操作位范围			复位后
					1 位	8 位	16 位	
F00E0H	乘法/除法数据寄存器 C (L)	MDCL		R/W	-	-	√	0000H
F00E2H	乘法/除法数据寄存器 C (H)	MDCH		R/W	-	-	√	0000H
F00E8H	乘法/除法控制寄存器	MDUC		R/W	√	√	-	00H
F00F0H	外围允许寄存器 0	PER0		R/W	√	√	-	00H
F00F3H	操作速度模式控制寄存器	OSMC		R/W	-	√	-	00H
F00F5H	RAM 奇偶校验错误控制寄存器	RPECTL		R/W	√	√	-	00H
F00FEH	BCD 校正结果寄存器	BCDADJ		R	-	√	-	不定
F0100H	串行状态寄存器 00	SSR00L	SSR00	R	-	√	√	0000H
F0101H		-			-	-		
F0102H	串行状态寄存器 01	SSR01L	SSR01	R	-	√	√	0000H
F0103H		-			-	-		
F0104H	串行状态寄存器 02	SSR02L	SSR02	R	-	√	√	0000H
F0105H		-			-	-		
F0106H	串行状态寄存器 03	SSR03L	SSR03	R	-	√	√	0000H
F0107H		-			-	-		
F0108H	串行标志清除触发寄存器 00	SIR00L	SIR00	R/W	-	√	√	0000H
F0109H		-			-	-		
F010AH	串行标志清除触发寄存器 01	SIR01L	SIR01	R/W	-	√	√	0000H
F010BH		-			-	-		
F010CH	串行标志清除触发寄存器 02	SIR02L	SIR02	R/W	-	√	√	0000H
F010DH		-			-	-		
F010EH	串行标志清除触发寄存器 03	SIR03L	SIR03	R/W	-	√	√	0000H
F010FH		-			-	-		
F0110H	串行模式寄存器 00	SMR00		R/W	R/W	-	√	0020H
F0111H								
F0112H	串行模式寄存器 01	SMR01		R/W	R/W	-	√	0020H
F0113H								
F0114H	串行模式寄存器 02	SMR02		R/W	R/W	-	√	0020H
F0115H								
F0116H	串行模式寄存器 03	SMR03		R/W	R/W	-	√	0020H
F0117H								
F0118H	串行通信操作设置寄存器 00	SCR00		R/W	R/W	-	√	0087H
F0119H								
F011AH	串行通信操作设置寄存器 01	SCR01		R/W	R/W	-	√	0087H
F011BH								
F011CH	串行通信操作设置寄存器 02	SCR02		R/W	R/W	-	√	0087H
F011DH								
F011EH	串行通信操作设置寄存器 03	SCR03		R/W	R/W	-	√	0087H
F011FH								

表 3-6. 扩展 SFR (2nd SFR)列表 (3/12)

地址	特殊功能寄存器(SFR)名称	符号		R/W	可操作位范围			复位后
					1 位	8 位	16 位	
F0120H	串行通道允许状态寄存器 0	SE0L	SE0	R	√	√	√	0000H
F0121H		-			-	-		
F0122H	串行通道开始寄存器 0	SS0L	SS0	R/W	√	√	√	0000H
F0123H		-			-	-		
F0124H	串行通道停止寄存器 0	ST0L	ST0	R/W	√	√	√	0000H
F0125H		-			-	-		
F0126H	串行时钟选择寄存器 0	SPS0L	SPS0	R/W	-	√	√	0000H
F0127H		-			-	-		
F0128H	串行输出寄存器 0	SO0		R/W	R/W	-	√	0F0FH
F0129H					-	-		
F012AH	串行输出允许寄存器 0	SOE0L	SOE0	R/W	√	√	√	0000H
F012BH		-			-	-		
F0134H	串行输出电平寄存器 0	SOL0L	SOL0	R/W	-	√	√	0000H
F0135H		-			-	-		
F0138H	串行待机控制寄存器 0	SSC0L	SSC0	R/W	-	√	√	0000H
		-			-	-		
F0180H	定时器计数器寄存器 00	TCR00		R	-	-	√	FFFFH
F0181H					-	-		
F0182H	定时器计数器寄存器 01	TCR01		R	-	-	√	FFFFH
F0183H					-	-		
F0184H	定时器数据寄存器 02	TCR02		R	-	-	√	FFFFH
F0185H					-	-		
F0186H	定时器计数器寄存器 03	TCR03		R	-	-	√	FFFFH
F0187H					-	-		
F0188H	定时器计数器寄存器 04	TCR04		R	-	-	√	FFFFH
F0189H					-	-		
F018AH	定时器计数器寄存器 05	TCR05		R	-	-	√	FFFFH
F018BH					-	-		
F018CH	定时器计数器寄存器 06	TCR06		R	-	-	√	FFFFH
F018DH					-	-		
F018EH	定时器计数器寄存器 07	TCR07		R	-	-	√	FFFFH
F018FH					-	-		
F0190H	定时器模式寄存器 00	TMR00		R/W	-	-	√	0000H
F0191H					-	-		
F0192H	串行模式寄存器 01	TMR01		R/W	-	-	√	0000H
F0193H					-	-		
F0194H	定时器数据寄存器 02	TMR02		R/W	-	-	√	0000H
F0195H					-	-		

表 3-6. 扩展 SFR (2nd SFR)列表 (4/12)

地址	特殊功能寄存器(SFR)名称	符号		R/W	可操作位范围			复位后
					1 位	8 位	16 位	
F0196H	定时器模式寄存器 03	TMR03		R/W	-	-	√	0000H
F0197H								
F0198H	定时器模式寄存器 04	TMR04		R/W	-	-	√	0000H
F0199H								
F019AH	定时器模式寄存器 05	TMR05		R/W	-	-	√	0000H
F019BH								
F019CH	定时器模式寄存器 06	TMR06		R/W	-	-	√	0000H
F019DH								
F019EH	定时器模式寄存器 07	TMR07		R/W	-	-	√	0000H
F019FH								
F01A0H	定时器状态寄存器 00	TSR00L	TSR00	R	-	√	√	0000H
F01A1H		-			-			
F01A2H	定时器状态寄存器 01	TSR01L	TSR01	R	-	√	√	0000H
F01A3H		-			-			
F01A4H	定时器数据寄存器 02	TSR02L	TSR02	R	-	√	√	0000H
F01A5H		-			-			
F01A6H	定时器状态寄存器 03	TSR03L	TSR03	R	-	√	√	0000H
F01A7H		-			-			
F01A8H	定时器状态寄存器 04	TSR04L	TSR04	R	-	√	√	0000H
F01A9H		-			-			
F01AAH	定时器状态寄存器 05	TSR05L	TSR05	R	-	√	√	0000H
F01ABH		-			-			
F01ACH	定时器状态寄存器 06	TSR06L	TSR06	R	-	√	√	0000H
F01ADH		-			-			
F01AEH	定时器状态寄存器 07	TSR07L	TSR07	R	-	√	√	0000H
F01AFH		-			-			
F01B0H	定时器通道允许状态寄存器 0	TE0L	TE0	R	√	√	√	0000H
F01B1H		-			-			
F01B2H	定时器通道开始寄存器 0	TS0L	TS0	R/W	√	√	√	0000H
F01B3H		-			-			
F01B4H	定时器通道停止寄存器 0	TT0L	TT0	R/W	√	√	√	0000H
F01B5H		-			-			
F01B6H	定时器时钟选择寄存器 0	TPS0		R/W	R/W	-	√	0000H
F01B7H								
F01B8H	定时器输出寄存器 0	TO0L	TO0	R/W	-	√	√	0000H
F01B9H		-			-			
F01BAH	定时器输出允许寄存器 0	TOE0L	TOE0	R/W	√	√	√	0000H
F01BBH		-			-			
F01BCH	定时器输出电平寄存器 0	TOL0L	TOL0	R/W	-	√	√	0000H
F01BDH		-			-			
F01BEH	定时器输出模式寄存器 0	TOM0L	TOM0	R/W	-	√	√	0000H
F01BFH		-			-			

表 3-6. 扩展 SFR (2nd SFR)列表 (5/12)

地址	特殊功能寄存器(SFR)名称	符号	R/W	可操作位范围			复位后
				1 位	8 位	16 位	
F0230H	IICA 控制寄存器 00	IICCTL00	R/W	√	√	–	00H
F0231H	IICA 控制寄存器 10	IICCTL01	R/W	√	√	–	00H
F0232H	IICA 低电平宽度设置寄存器 0	IICWLO	R/W	–	√	–	FFH
F0233H	IICA 高电平宽度设置寄存器 0	IICWH0	R/W	–	√	–	FFH
F0234H	从属地址寄存器 0	SVA0	R/W	–	√	–	00H
F02F0H	闪存 CRC 控制寄存器	CRC0CTL	R/W	√	√	–	00H
F02F2H	闪存 CRC 运算结果寄存器	PGCRCL	R/W	–	–	√	0000H
F02FAH	CRC 数据寄存器	CRCD	R/W	–	–	√	0000H
F0500H	端口寄存器 20	P20	R/W	√	√	–	00H
F0508H	外围允许寄存器 1	PER1	R/W	√	√	–	00H
F0509H	外围允许寄存器 2	PER2	R/W	√	√	–	00H
F050AH	PLL 控制寄存器	PLLCTL	R/W	√	√	–	00H
F0510H	端口模式寄存器 20	PM20	R/W	√	√	–	FFH
F0518H	外部中断上升沿允许寄存器 2	EGP2	R/W	√	√	–	00H
F0519H	外部中断下降沿允许寄存器 2	EGN2	R/W	√	√	–	00H
F0520H	上拉电阻选择寄存器 20	PU20	R/W	√	√	–	00H
F0530H	端口输出模式寄存器 20	POM20	R/W	√	√	–	00H
F0550H	可编程增益放大器控制寄存器	PGACTL	R/W	√	√	–	00H
F0551H	可编程增益放大器输入通道选择寄存器	PGAINS	R/W	–	√	–	00H
F0552H	比较器 0 控制寄存器	C0CTL	R/W	√	√	–	00H
F0553H	比较器 1 控制寄存器	C1CTL	R/W	√	√	–	00H
F0554H	比较器 2 控制寄存器	C2CTL	R/W	√	√	–	00H
F0555H	比较器 3 控制寄存器	C3CTL	R/W	√	√	–	00H
F0556H	比较器 4 控制寄存器	C4CTL	R/W	√	√	–	00H
F0557H	比较器 5 控制寄存器	C5CTL	R/W	√	√	–	00H
F0558H	比较器上升沿允许寄存器	CMPEGP0	R/W	√	√	–	00H
F0559H	比较器下降沿允许寄存器	CMPEGN0	R/W	√	√	–	00H
F055AH	比较器输出监视寄存器	CMPMON	R	√	√	–	00H
F055BH	外部中断控制寄存器	INTPCTL	R/W	√	√	–	00H
F0560H	比较器内部基准电压控制寄存器	CVRCTL	R/W	√	√	–	00H
F0561H	比较器内部基准电压选择寄存器 0	C0RVM	R/W	–	√	–	00H
F0562H	比较器内部基准电压选择寄存器 1	C1RVM	R/W	–	√	–	00H
F0563H	比较器内部基准电压选择寄存器 2	C2RVM	R/W	–	√	–	00H

表 3-6. 扩展 SFR (2nd SFR)列表 (6/12)

地址	特殊功能寄存器(SFR)名称	符号		R/W	可操作位范围			复位后
					1 位	8 位	16 位	
F0564H	窗口比较器功能设置寄存器	CMPWDC		R/W	√	√	–	00H
F0565H	比较器输入切换控制寄存器	CMPSEL		R/W	–	√	–	00H
F0570H	串行数据寄存器 40	TXD4	R/W	R/W	–	√	√	0000H
F0571H		–	R/W	R/W	–	–	–	
F0572H	串行数据寄存器 41	RXD4	R/W	R/W	–	√	√	0000H
F0573H		–	R/W	R/W	–	–	–	
F0578H	DALI 发送数据寄存器 L4	SDTL4		R/W	R/W	–	√	0000H
F0579H					R/W	–	–	
F057AH	DALI 发送数据寄存器 H4	SDTH4		R/W	R/W	–	√	0000H
F057BH						–	–	
F057CH	DALI 接收数据寄存器 L4	SDCL4		R	–	–	√	0000H
F057DH						–	–	
F057EH	DALI 接收数据寄存器 H4	SDCH4		R	–	–	√	0000H
F057FH						–	–	
F0580H	串行状态寄存器 40	SSR40L	SSR40	R	–	√	√	0000H
F0581H		–			–	–	–	
F0582H	串行状态寄存器 41	SSR40L	SSR41	R	–	√	√	0000H
F0583H		–			–	–	–	
F0588H	串行标志清除触发寄存器 40	SIR40L	SIR40	R/W	–	√	√	0000H
F0589H		–		R/W	–	–	–	
F058AH	串行标志清除触发寄存器 41	SIR41L	SIR41	R/W	–	√	√	0000H
F058BH		–		R/W	–	–	–	
F0590H	串行模式寄存器 40	SMR40		R/W	R/W	–	√	0020H
F0591H					R/W	–	–	
F0592H	串行模式寄存器 41	SMR41		R/W	R/W	–	√	0087H
F0593H						–	–	
F0598H	串行通信操作设置寄存器 40	SCR40		R/W	–	–	√	0087H
F0599H						–	–	
F059AH	串行通信操作设置寄存器 41	SCR41		R/W	–	–	√	0020H
F059BH						–	–	
F05A0H	串行通道允许状态寄存器 4	SE4L	SE4	R	√	√	√	0000H
F05A1H		–			–	–	–	
F05A2H	串行通道开始寄存器 4	SS4L	SS4	R/W	√	√	√	0000H
F05A3H		–			–	–	–	
F05A4H	串行通道停止寄存器 4	ST4L	ST4	R/W	√	√	√	0000H
F05A5H		–			–	–	–	
F05A6H	串行时钟选择寄存器 4	SPS4L	SPS4	R/W	–	√	√	0000H
F05A7H		–			–	–	–	
F05A8H	串行输出寄存器 40	SO4		R/W	–	–	√	0F0FH
F05A9H						–	–	

表 3-6. 扩展 SFR (2nd SFR)列表 (7/12)

地址	特殊功能寄存器(SFR)名称	符号		R/W	可操作位范围			复位后
					1 位	8 位	16 位	
F05AAH	串行数据允许寄存器 4	SOE4L	SOE4	R/W	√	√	√	0000H
F05ABH		-			-	-		
F05B4H	串行输出电平寄存器 4	SOL4L	SOL4	R/W	-	√	√	0000H
F05B5H		-			-	-		
F05B8H	串行待机控制寄存器 4	SSC4L	SSC4	R/W	-	√	√	0000H
F05B9H		-			-	-		
F05BAH	串行选项控制寄存器 4	SOC4		R/W	-	-	√	0000H
F05BBH								
F05C0H	外围输入/输出重定向寄存器 1	PIOR1		R/W	-	√	-	00H
F05C1H	噪声滤波器允许寄存器 3	NFEN3		R/W	√	√	-	00H
F05C2H	中断屏蔽标志寄存器 0	INTMK0		R/W	√	√	-	FFH
F05C3H	中断监视标志寄存器 0	INTMF0		R/W	√	√	-	00H
F05C4H	非法存储器存取检测控制寄存器 1	IAWCTL1		R/W	-	√	-	00H
F05C5H	定时器时钟选择寄存器 2	TPS2		R/W	-	√	-	00H
F05C6H	外围功能切换寄存器 0	PFSEL0		R/W	√	√	-	00H
F05C7H	单线 UART 控制寄存器	SUCTL		R/W	√	√	-	00H
F0600H	16 位定时器 KB 比较寄存器 00	TKBCR00		R/W	-	-	√	0000H
F0601H								
F0602H	16 位定时器 KB 比较寄存器 01	TKBCR01		R/W	-	-	√	0000H
F0603H								
F0604H	16 位定时器 KB 比较寄存器 02	TKBCR02		R/W	-	-	√	0000H
F0605H								
F0606H	16 位定时器 KB 比较寄存器 03	TKBCR03		R/W	-	-	√	0000H
F0607H								
F0608H	16 位定时器 KB 触发比较寄存器 0	TKBTGCR0		R/W	-	-	√	0000H
F0609H								
F060AH	16 位定时器 KB 软启动初始占空比寄存器 00	TKBSIR00		R/W	-	-	√	0000H
F060BH								
F060CH	16 位定时器 KB 软启动初始占空比寄存器 01	TKBSIR01		R/W	-	-	√	0000H
F060DH								
F060EH	16 位定时器 KB 抖动计数寄存器 00	TKBDNR00		R/W	-	√	-	00H
F060FH	16 位定时器 KB 软启动步宽寄存器 00	TKBSSR00		R/W	-	√	-	00H

注 使用 8 位存储器操作指令时，仅能读取。

表 3-6. 扩展 SFR (2nd SFR)列表 (8/12)

地址	特殊功能寄存器(SFR)名称	符号	R/W	可操作位范围			复位后
				1 位	8 位	16 位	
F0610H	16 位定时器 KB 抖动计数寄存器 01	TKBDNR01	R/W	-	√	-	00H
F0611H	16 位定时器 KB 软启动步宽寄存器 01	TKBSSR01	R/W	-	√	-	00H
F0612H	16 位定时器 KB 触发寄存器 0	TKBTRG0	W	√	√	-	00H
F0613H	16 位定时器 KB 标志寄存器 0	TKBFLG0	R	√	√	-	00H
F0614H	16 位定时器 KB 比较 1L&抖动计数寄存器 00	TKBCRLD00	R/W	-	-	√	0000H
F0615H							
F0616H	16 位定时器 KB 比较 3L&抖动计数寄存器 01	TKBCRLD01	R/W	-	-	√	0000H
F0617H							
F0622H	16 位定时器 KB 操作控制寄存器 00	TKBCTL00	R/W	-	-	√	0000H
F0623H							
F0624H	16 位定时器 KB 最大频率限制设置寄存器 0	TKBMFR0	R/W	-	-	√	0000H
F0625H							
F0626H	16 位定时器 KB 输出控制寄存器 00	TKBIOC00	R/W	√	√	-	00H
F0627H	16 位定时器 KB 标志清除触发寄存器 0	TKBCLR0	W	√	√	-	00H
F0628H	16 位定时器 KB 输出控制寄存器 01	TKBIOC01	R/W	√	√	-	00H
F0629H	16 位定时器 KB 操作控制寄存器 01	TKBCTL01	R/W	√	√	-	00H
F0630H	强制输出停止功能控制寄存器 00	TKBPACTL00	R/W	-	-	√	0000H
F0631H							
F0632H	强制输出停止功能控制寄存器 01	TKBPACTL01	R/W	-	-	√	0000H
F0633H							
F0634H	强制输出停止功能开始触发寄存器 0	TKBPAHFS0	W	√	√	-	00H
F0635H	强制输出停止功能停止触发寄存器 0	TKBPAHFT0	W	√	√	-	00H
F0636H	强制输出停止功能标志寄存器 0	TKBPAFLG0	R	√	√	-	00H
F0637H	强制输出停止功能控制寄存器 02	TKBPACTL02	R/W	√	√	-	00H

表 3-6. 扩展 SFR (2nd SFR)列表 (9/12)

地址	特殊功能寄存器(SFR)名称	符号	R/W	可操作位范围			复位后
				1 位	8 位	16 位	
F0640H	16 位定时器 KB 比较寄存器 10	TKBCR10	R/W	-	-	√	0000H
F0641H							
F0642H	16 位定时器 KB 比较寄存器 11	TKBCR11	R/W	-	-	√	0000H
F0643H							
F0644H	16 位定时器 KB 比较寄存器 12	TKBCR12	R/W	-	-	√	0000H
F0645H							
F0646H	16 位定时器 KB 比较寄存器 13	TKBCR13	R/W	-	-	√	0000H
F0647H							
F0648H	16 位定时器 KB 触发比较寄存器 1	TKBTGCR1	R/W	-	-	√	0000H
F0649H							
F064AH	16 位定时器 KB 软启动初始占空比寄存器 10	TKBSIR10	R/W	-	-	√	0000H
F064BH							
F064CH	16 位定时器 KB 软启动初始占空比寄存器 11	TKBSIR11	R/W	-	-	√	0000H
F064DH							
F064EH	16 位定时器 KB 抖动计数寄存器 10	TKBDNR10	R/W	-	√	-	00H
F064FH	16 位定时器 KB 软启动步宽寄存器 10	TKBSSR10	R/W	-	√	-	00H
F0650H	16 位定时器 KB 抖动计数寄存器 11	TKBDNR11	R/W	-	√	-	00H
F0651H	16 位定时器 KB 软启动步宽寄存器 11	TKBSSR11	R/W	-	√	-	00H
F0652H	16 位定时器 KB 触发寄存器 1	TKBTRG1	W	√	√	-	00H
F0653H	16 位定时器 KB 标志寄存器 1	TKBFLG1	R	√	√	-	00H
F0654H	16 位定时器 KB 比较 1L&抖动计数寄存器 10	TKBCRLD10	R/W	-	-	√	0000H
F0655H							
F0656H	16 位定时器 KB 比较 1L&抖动计数寄存器 11	TKBCRLD11	R/W	-	-	√	0000H
F0657H							
F0662H	16 位定时器 KB 操作控制寄存器 10	TKBCTL10	R/W	-	-	√	0000H
F0663H							
F0664H	16 位定时器 KB 最大频率限制设置寄存器 1	TKBMFR1	R/W	-	-	√	0000H
F0665H							
F0666H	16 位定时器 KB 输出控制寄存器 10	TKBIOC10	R/W	√	√	-	00H
F0667H	16 位定时器 KB 标志清除触发寄存器 1	TKBCLR1	W	√	√	-	00H

表 3-6. 扩展 SFR (2nd SFR)列表 (10/12)

地址	特殊功能寄存器(SFR)名称	符号	R/W	可操作位范围			复位后
				1 位	8 位	16 位	
F0668H	16 位定时器 KB 输出控制寄存器 11	TKBIOC11	R/W	√	√	–	00H
F0669H	16 位定时器 KB 操作控制寄存器 11	TKBCTL11	R/W	√	√	–	00H
F0670H	强制输出停止功能控制寄存器 10	TKBPACTL10	R/W	–	–	√	0000H
F0671H							
F0672H	强制输出停止功能控制寄存器 11	TKBPACTL11	R/W	–	–	√	0000H
F0673H							
F0674H	强制输出停止功能开始触发寄存器 1	TKBPAHFS1	W	√	√	–	00H
F0675H	强制输出停止功能停止触发寄存器 1	TKBPAHFT1	W	√	√	–	00H
F0676H	强制输出停止功能标志寄存器 1	TKBPAFLG1	R	√	√	–	00H
F0677H	强制输出停止功能控制寄存器 12	TKBPACTL12	R/W	√	√	–	00H
F0680H	16 位定时器 KB 比较寄存器 20	TKBCR20	R/W	–	–	√	0000H
F0681H							
F0682H	16 位定时器 KB 比较寄存器 21	TKBCR21	R/W	–	–	√	0000H
F0683H							
F0684H	16 位定时器 KB 比较寄存器 22	TKBCR22	R/W	–	–	√	0000H
F0685H							
F0686H	16 位定时器 KB 比较寄存器 23	TKBCR23	R/W	–	–	√	0000H
F0687H							
F0688H	16 位定时器 KB 触发比较寄存器 2	TKBTGCR2	R/W	–	–	√	0000H
F0689H							
F068AH	16 位定时器 KB 软启动初始占空比寄存器 20	TKBSIR20	R/W	–	–	√	0000H
F068BH							
F068CH	16 位定时器 KB 软启动初始占空比寄存器 21	TKBSIR21	R/W	–	–	√	0000H
F068DH							
F068EH	16 位定时器 KB 抖动计数寄存器 20	TKBDNR20	R/W	–	√	–	00H
F068FH	16 位定时器 KB 软启动步宽寄存器 20	TKBSSR20	R/W	–	√	–	00H
F0690H	16 位定时器 KB 抖动计数寄存器 21	TKBDNR21	R/W	–	√	–	00H
F0691H	16 位定时器 KB 软启动步宽寄存器 21	TKBSSR21	R/W	–	√	–	00H

表 3-6. 扩展 SFR (2nd SFR)列表 (11/12)

地址	特殊功能寄存器(SFR)名称	符号	R/W	可操作位范围			复位后
				1 位	8 位	16 位	
F0692H	16 位定时器 KB 触发寄存器 2	TKBTRG2	W	√	√	–	00H
F0693H	16 位定时器 KB 标志寄存器 2	TKBFLG2	R	√	√	–	00H
F0694H	16 位定时器 KB 比较 1L&抖动计数寄存器 20	TKBCRLD20	R/W	–	–	√	0000H
F0695H							
F0696H	16 位定时器 KB 比较 1L&抖动计数寄存器 21	TKBCRLD21	R/W	–	–	√	0000H
F0697H							
F06A2H	16 位定时器 KB 操作控制寄存器 20	TKBCTL20	R/W	–	–	√	0000H
F06A3H							
F06A4H	16 位定时器 KB 最大频率限制设置寄存器 2	TKBMFR2	R/W	–	–	√	0000H
F06A5H							
F06A6H	16 位定时器 KB 输出控制寄存器 20	TKBIOC20	R/W	√	√	–	00H
F06A7H	16 位定时器 KB 标志清除触发寄存器 2	TKBCLR2	W	√	√	–	00H
F06A8H	16 位定时器 KB 输出控制寄存器 21	TKBIOC21	R/W	√	√	–	00H
F06A9H	16 位定时器 KB 操作控制寄存器 21	TKBCTL21	R/W	√	√	–	00H
F06B0H	强制输出停止功能控制寄存器 20	TKBPACTL20	R/W	–	–	√	0000H
F06B1H							
F06B2H	强制输出停止功能控制寄存器 21	TKBPACTL21	R/W	–	–	√	0000H
F06B3H							
F06B4H	强制输出停止功能开始触发寄存器 2	TKBPAHFS2	W	√	√	–	00H
F06B5H	强制输出停止功能停止触发寄存器 2	TKBPAHFT2	W	√	√	–	00H
F06B6H	强制输出停止功能标志寄存器 2	TKBPAFLG2	R	√	√	–	00H
F06B7H	强制输出停止功能控制寄存器 22	TKBPACTL22	R/W	√	√	–	00H

表 3-6. 扩展 SFR (2nd SFR)列表 (12/12)

地址	特殊功能寄存器(SFR)名称	符号	R/W	可操作位范围			复位后
				1 位	8 位	16 位	
F06D0H	16 位定时器 KC 比较寄存器 0	TKCCR0	R/W	-	-	√	0000H
F06D2H	16 位定时器 KC 占空比较寄存器 00	TKCDUTY00	R/W	-	-	√	0000H
F06D4H	16 位定时器 KC 占空比较寄存器 01	TKCDUTY01	R/W	-	-	√	0000H
F06D6H	16 位定时器 KC 占空比较寄存器 02	TKCDUTY02	R/W	-	-	√	0000H
F06D8H	16 位定时器 KC 占空比较寄存器 03	TKCDUTY03	R/W	-	-	√	0000H
F06DAH	16 位定时器 KC 占空比较寄存器 04	TKCDUTY04	R/W	-	-	√	0000H
F06DCH	16 位定时器 KC 占空比较寄存器 05	TKCDUTY05	R/W	-	-	√	0000H
F06DEH	16 位定时器 KC 触发寄存器 0	TKCTRG0	W	√	√	-	00H
F06DFH	16 位定时器 KC 标志寄存器 0	TKCFLG0	R	√	√	-	00H
F06E2H	16 位定时器 KC 输出控制寄存器 00	TKCIOC00	R/W	-	-	√	0000H
F06E4H	16 位定时器 KC 输出控制寄存器 01	TKCIOC01	R/W	√	√	-	00H
F06E5H	16 位定时器 KC 操作控制寄存器 0	TKCCTL0	R/W	√	√	-	00H
F06E6H	16 位定时器 KC 输出标志寄存器 0	TKCTOF0	R	√	√	-	00H

备注 关于 SFR 区域中的 SFR，请参阅 3.2.4 特殊功能寄存器 (SFR)的表 3-5。

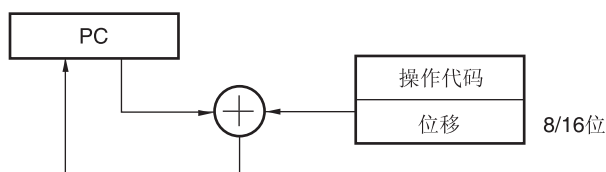
3.3 指令地址寻址

3.3.1 相对寻址

[功能]

相对寻址把指令字中的偏移量（有符号的补码数据：-128 至+127 或-32768 至+32767）中的偏移量与程序计数器(PC)的值（下一条指令的起始地址）之和存储在程序计数器(PC)中，指定转移目的地的程序地址。相对寻址仅用于分支指令。

图 3-12. 相对寻址的图示



3.3.2 立即寻址

[功能]

立即寻址把指令字中的立即数存储于程序计数器中，指定转移目的地的程序地址。

在立即寻址中，利用 `CALL !!addr20` 或 `BR !!addr20` 来指定 20 位地址，利用 `CALL !addr16` 或 `BR !addr16` 来指定 16 位地址。指定 16 位地址时，高 4 位设为 0000。

图 3-13. `CALL !!addr20/BR !!addr20` 的示例

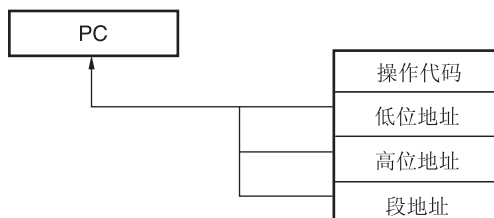
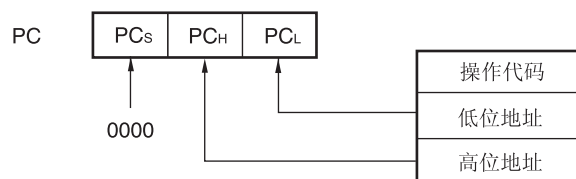


图 3-14. `CALL !addr16/BR !addr16` 的示例



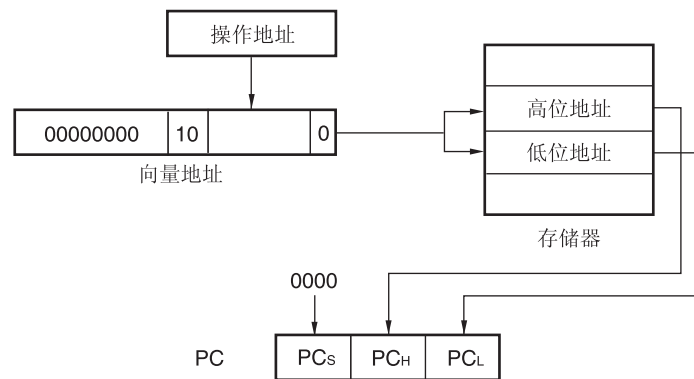
3.3.3 表间接寻址

[功能]

表间接寻址用指令字中的 5 位立即数在 CALLT 表区域 (0080H 至 00BFH) 中指定 1 个表地址, 将表地址和下一个地址的内容以 16 位数据的形式存储在程序计数器(PC)中, 指定程序地址。表间接寻址仅用于 CALLT 指令。

RL78 单片机中, 仅在 00000H 至 0FFFFH 的 64KB 空间中可以转移。

图 3-15. 表间接寻址的图示

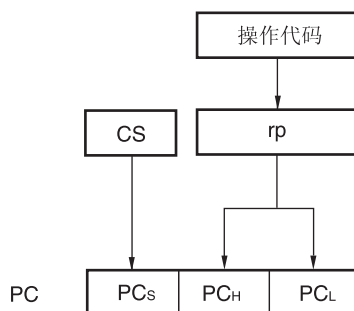


3.3.4 寄存器直接寻址

[功能]

寄存器直接寻址，是将指令字指定的当前寄存器组的通用寄存器对(AX/BC/DE/HL)和 CS 寄存器的内容，以 20 位数据的形式存储在程序计数器(PC)中，指定程序地址。寄存器直接寻址只适用于 CALL AX、BC、DE、HL 和 BR AX 指令。

图 3-16. 寄存器直接寻址的概述



3.4 处理数据地址的寻址

3.4.1 隐含寻址

[功能]

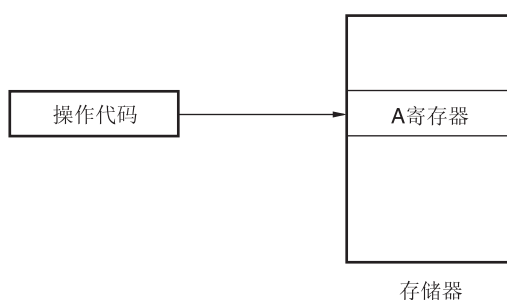
用于存取具有特殊功能的寄存器（例如累加器）的指令，可以通过指令字直接指定，而无需使用指令字中的任何寄存器指定字段。

[操作数格式]

隐含寻址可以通过指令来自动使用，因此不需要特殊的操作数格式。

隐含寻址只适用于 MULU X。

图 3-17. 隐含寻址的图示



3.4.2 寄存器寻址

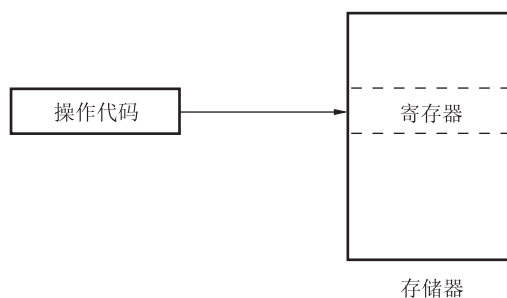
[功能]

寄存器寻址是将通用寄存器作为操作数来进行存取的一种寻址方式。通过 3 位的指令字选择 8 位寄存器，通过 2 位的指令字选择 16 位寄存器。

[操作数格式]

标识符	说明
r	X, A, C, B, E, D, L, H
rp	AX, BC, DE, HL

图 3-18. 寄存器寻址的图示



3.4.3 直接寻址

[功能]

直接寻址是将指令字中的立即数作为操作数地址，从而直接指定目标地址。

[操作数格式]

标识符	说明
ADDR16	标号或16位立即数(仅可以指定F0000H至FFFFFFH的空间)
ES: ADDR16	标号或16位立即数(高4位地址由ES寄存器指定)

图 3-19. ADDR16 的示例

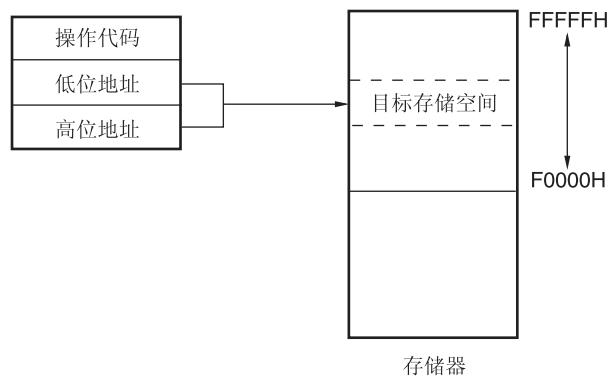
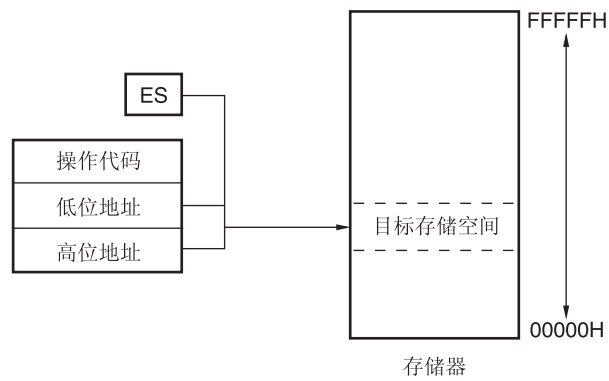


图 3-20. ES:ADDR16 的示例



3.4.4 短直接寻址

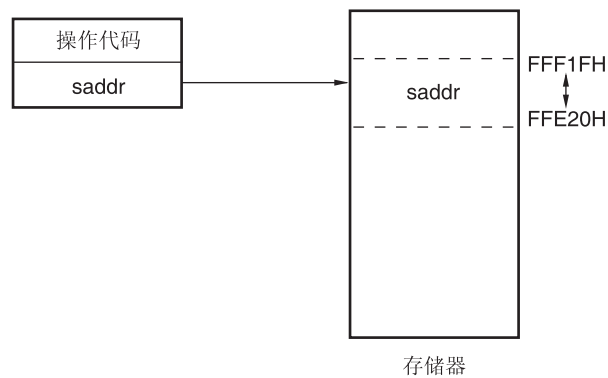
[功能]

短直接寻址利用指令字中的 8 位数据直接指定目标地址。这种寻址方式只适用于从 FFE20H 至 FFF1FH 的空间。

[操作数格式]

标识符	说明
SADDR	标号、FFE20H至FFF1FH区的立即数或0FE20H至0FF1FH区的立即数 (仅可以指定FFE20H至FFF1FH的空间)
SADDRP	标号、FFE20H至FFF1FH区的立即数或0FE20H至0FF1FH区的立即数(仅限偶数地址) (仅可以指定FFE20H至FFF1FH的空间)

图 3-21. 短直接寻址的图示



备注 SADDR 和 SADDRP 用于以 16 位立即数（省略了实际地址的高 4 位）描述 FE20H 至 FF1FH 地址的值，或者以 20 位立即数描述 FFE20H 至 FFF1FH 地址的值。

无论采用 SADDR 还是 SADDRP，均指定存储器中 FFE20H 至 FFF1FH 空间内的地址。

3.4.5 SFR寻址

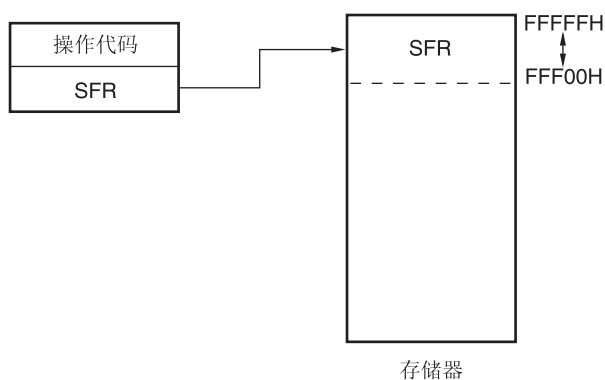
[功能]

SFR 寻址利用指令字中的 8 位数据直接指定目标 SFR 地址。这种寻址方式只适用于从 FFF00H 至 FFFFFH 的空间。

[操作数格式]

标识符	说明
SFR	SFR名称
SFRP	16位可操作SFR名称 (仅限偶数地址)

图 3-22. SFR 寻址的图示



3.4.6 寄存器间接寻址

[功能]

寄存器间接寻址是将指令字所指定的寄存器对的内容作为操作数地址，从而指定所对应的目标地址。

[操作数格式]

标识符	说明
-	[DE], [HL] (仅可以指定F0000H至FFFFFFH的空间)
-	ES:[DE], ES:[HL] (高4位地址由ES寄存器指定)

图 3-23. [DE], [HL]的示例

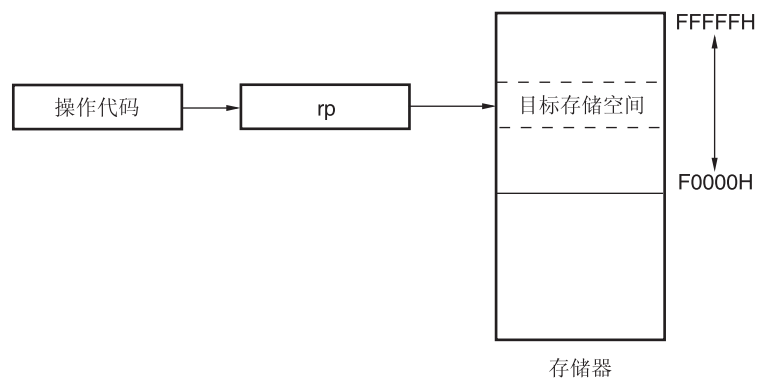
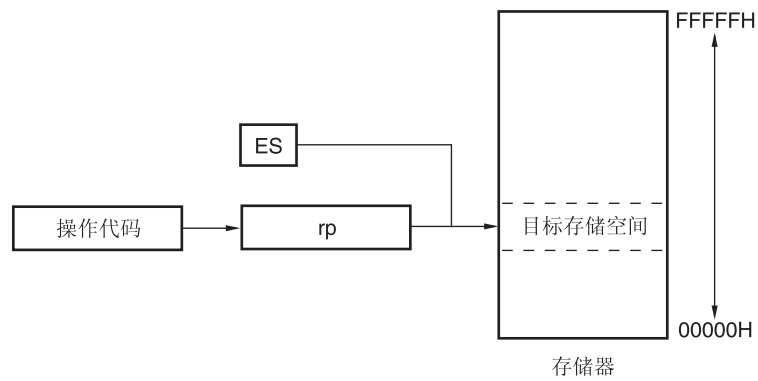


图 3-24. ES:[DE], ES:[HL]的示例



3.4.7 基址寻址

[功能]

基址寻址是将指令字所指定的寄存器对的内容作为基址，将 8 位立即数或 16 位立即数作为偏移量。并利用这些值之和指定目标地址。

[操作数格式]

标识符	说明
-	[HL + byte], [DE + byte], [SP + byte] (仅可以指定F0000H至FFFFFH的空间)
-	字[B],字[C] (仅可以指定F0000H至FFFFFH的空间)
-	字[BC] (仅可以指定F0000H至FFFFFH的空间)
-	ES:[HL + byte], ES:[DE + byte] (高4位地址由ES寄存器指定)
-	ES:字[B], ES:字[C] (高4位地址由ES寄存器指定)
-	ES:字[BC] (高4位地址由ES寄存器指定)

图 3-25. [SP+byte]的示例

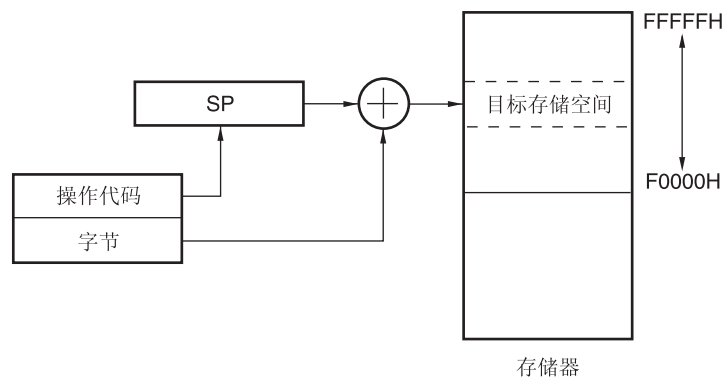


图 3-26. [HL + byte], [DE + byte]的示例

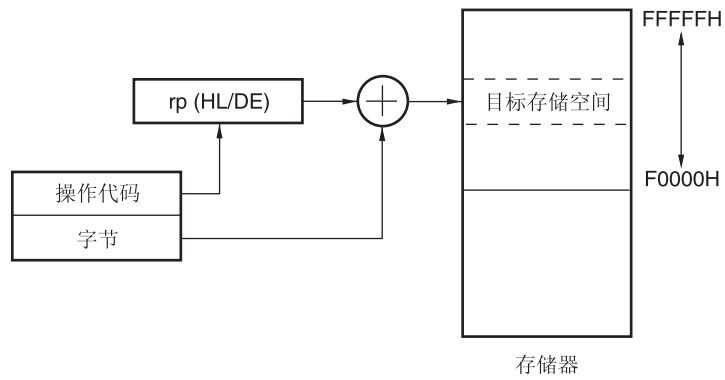


图 3-27. 字[B], 字[C]的示例

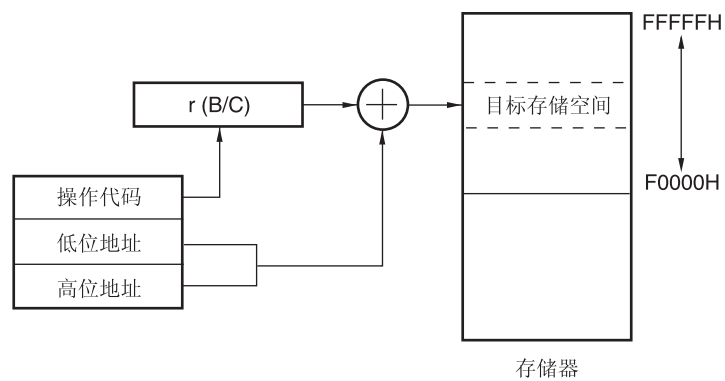


图 3-28. 字[BC]的示例

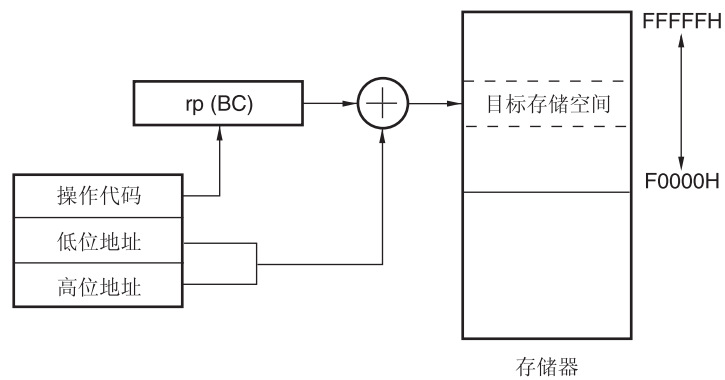


图 3-29. ES:[HL + byte], ES:[DE + byte]的示例

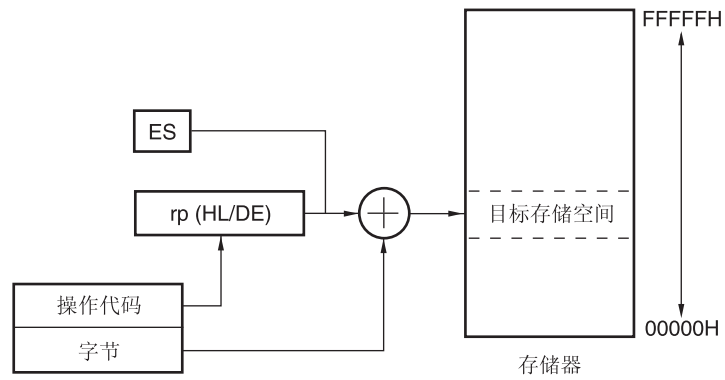


图 3-30. ES:字[B], ES:字[C]的示例

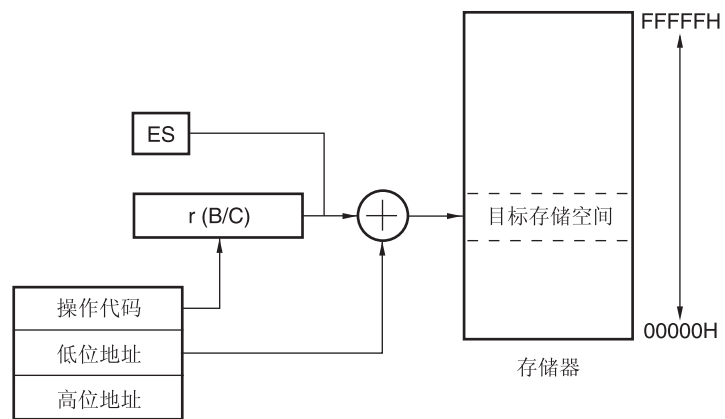
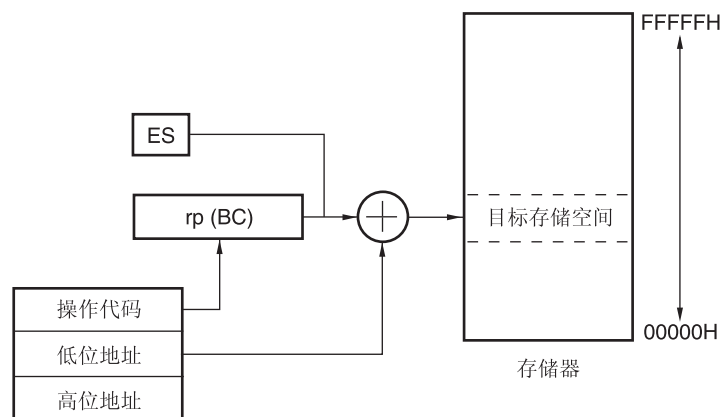


图 3-31. ES:字[BC]的示例



3.4.8 基址变址寻址

[功能]

基址变址寻址是将指令字所指定的寄存器对的内容作为基址，将同样通过指令字所指定的 B 寄存器或 C 寄存器的内容作为偏移地址。并利用这些值之和指定目标地址。

[操作数格式]

标识符	说明
-	[HL+B], [HL+C] (仅可以指定F0000H至FFFFFH的空间)
-	ES:[HL+B], ES:[HL+C] (高4位地址由ES寄存器指定)

图 3-32. [HL+B], [HL+C]的示例

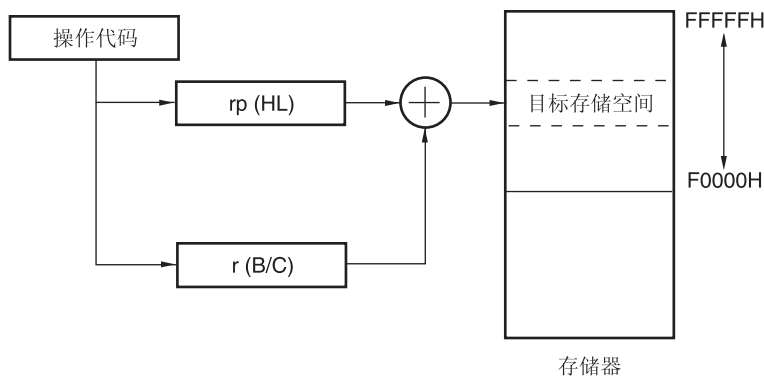
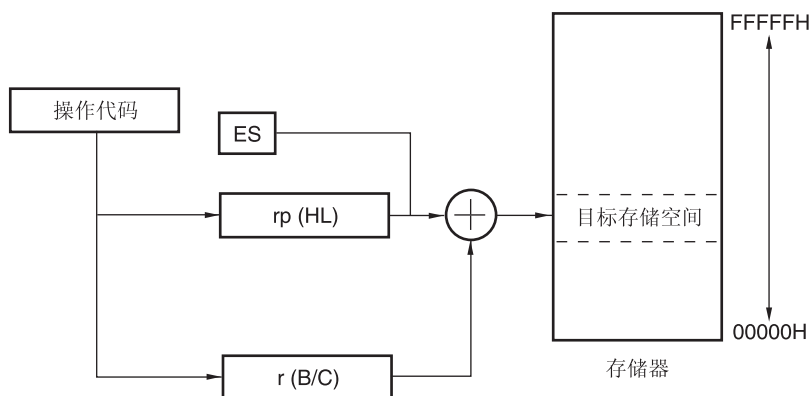


图 3-33. ES:[HL+B], ES:[HL+C]的示例



3.4.9 堆栈寻址

[功能]

通过堆栈指针 (SP)的内容对堆栈区域进行间接寻址。当执行 PUSH、POP、子程序调用和返回指令时，或者由于产生中断请求而保存/恢复寄存器时，将自动使用这种寻址方式。

堆栈寻址仅适用于内部 RAM 区域。

[操作数格式]

标识符	说明
-	PUSH AX/BC/DE/HL POP AX/BC/DE/HL CALL/CALLT RET BRK RETB (产生中断请求) RETI

第四章 端口功能

4.1 端口功能

RL78/I1A 单片机提供数字输入/输出端口，支持多种控制操作。

除了作为数字输入/输出端口的功能以外，这些端口还具有多种复用功能。关于复用功能的详情，请参阅第二章 引脚功能。

4.2 端口配置

端口包括下列硬件。

表 4-1. 端口配置

项目	配置
控制寄存器	端口模式寄存器(PM0 至 PM4, PM7, PM12, PM14, PM20) 端口寄存器(P0 至 P4, P7, P12 至 P14, P20) 上拉电阻选择寄存器(PU0, PU1, PU3, PU4, PU7, PU12, PU14, PU20) 端口输入模式寄存器(PIM0, PIM1) 端口输出模式寄存器(POM0, POM1, POM20) 端口模式控制寄存器 (PMC0, PMC12, PMC14) A/D 端口配置寄存器(ADPC) 外围输入/输出重定向寄存器(PIOR1)
端口	<ul style="list-style-type: none"> • 20 引脚产品 共计：16 (CMOS 输入/输出: 13, CMOS 输入: 3) • 30 引脚产品 共计：26 (CMOS 输入/输出: 23, CMOS 输入: 3) • 32 引脚产品 共计：28 (CMOS 输入/输出: 25, CMOS 输入: 3) • 38 引脚产品 共计：34 (CMOS 输入/输出: 29, CMOS 输入: 5)
上拉电阻	<ul style="list-style-type: none"> • 20 引脚产品 共计：8 • 30 引脚产品 共计：16 • 32 引脚产品 共计：18 • 38 引脚产品 共计：22

注意事项 本章的下述内容基本上是以使用 38 引脚产品并将外围输入/输出重定向寄存器(PIOR1)设置为 00H 为例进行说明的。

4.2.1 端口 0

端口 0 是具有输出锁存器的输入/输出端口。端口 0 可以通过端口模式寄存器 0(PM0)以 1 位为单位设为输入模式或输出模式。当 P02, P03, P05, P06 引脚用作输入端口时, 可通过上拉电阻选择寄存器 0(PU0)以 1 位为单位指定使用片上上拉电阻。

至 P03 引脚的输入, 可利用端口输入模式寄存器 0 (PIM0)通过普通输入缓冲器或 TTL 输入缓冲器以 1 位为单位指定。

P02 引脚的输出可利用端口输出模式寄存器 0 (POM0)以 1 位为单位指定为 N 沟开漏输出 (V_{DD} 耐压)。

P02 和 P03 引脚的输入, 可利用端口模式控制寄存器 0(PMC0)以 1 位为单位指定为模拟输入或数字输入。

端口 0 也可以用作定时器输入/输出、A/D 转换器模拟输入、串行接口数据输入/输出和比较器模拟输入。

产生复位信号时, 将变为以下配置。

- 30 和 38 引脚产品的 P02 和 P03 引脚 … 模拟输入
- 32 和 38 引脚产品的 P05 和 P06 引脚 … 输入模式

表 4-2. 使用端口 0 时寄存器的设置

名称	输入/输出	PM0x	PIM0x	POM0x	PMC0x	复用功能设置	备注
P02	输入	1	-	×	0 ^{注1}	×	
	输出	0		0	0 ^{注1}	TxD1输出 = 1 ^{注2}	CMOS输出
		0		1	0 ^{注1}		N沟开漏输出
P03	输入	1	0	-	0 ^{注1}	×	CMOS输入
		1	1		0 ^{注1}	×	TTL输入
	输出	0	×		0 ^{注1}	×	
P05, P06	输入	1	-	-	-	×	
	输出	0				TO05输出, TO06输出 = 0 ^{注3}	

注 1. 仅限 30、38 引脚产品

- 将 P02/TxD1/ANI17 用作通用端口时，请将串行通道允许状态寄存器 0 (SE0)的位 2 (SE02)、串行输出寄存器 0 (SO0)的位 2 (SO02)和串行输出允许寄存器 0 (SOE0)的位 2 (SOE02)设置为与其初始状态相同的值。
- 将 P05/TI05/TO05、P06/TI06/TO06 用作通用端口时，设置定时器输出寄存器 0 (TO0)的位 5 和 6 (TO05, TO06)、定时器输出允许寄存器 0 (TOE0)的位 5 和 6 (TOE05, TOE06)为 0，即设置为与其初始状态相同的值。

备注 ×: 忽略

PM0x: 端口模式寄存器 0

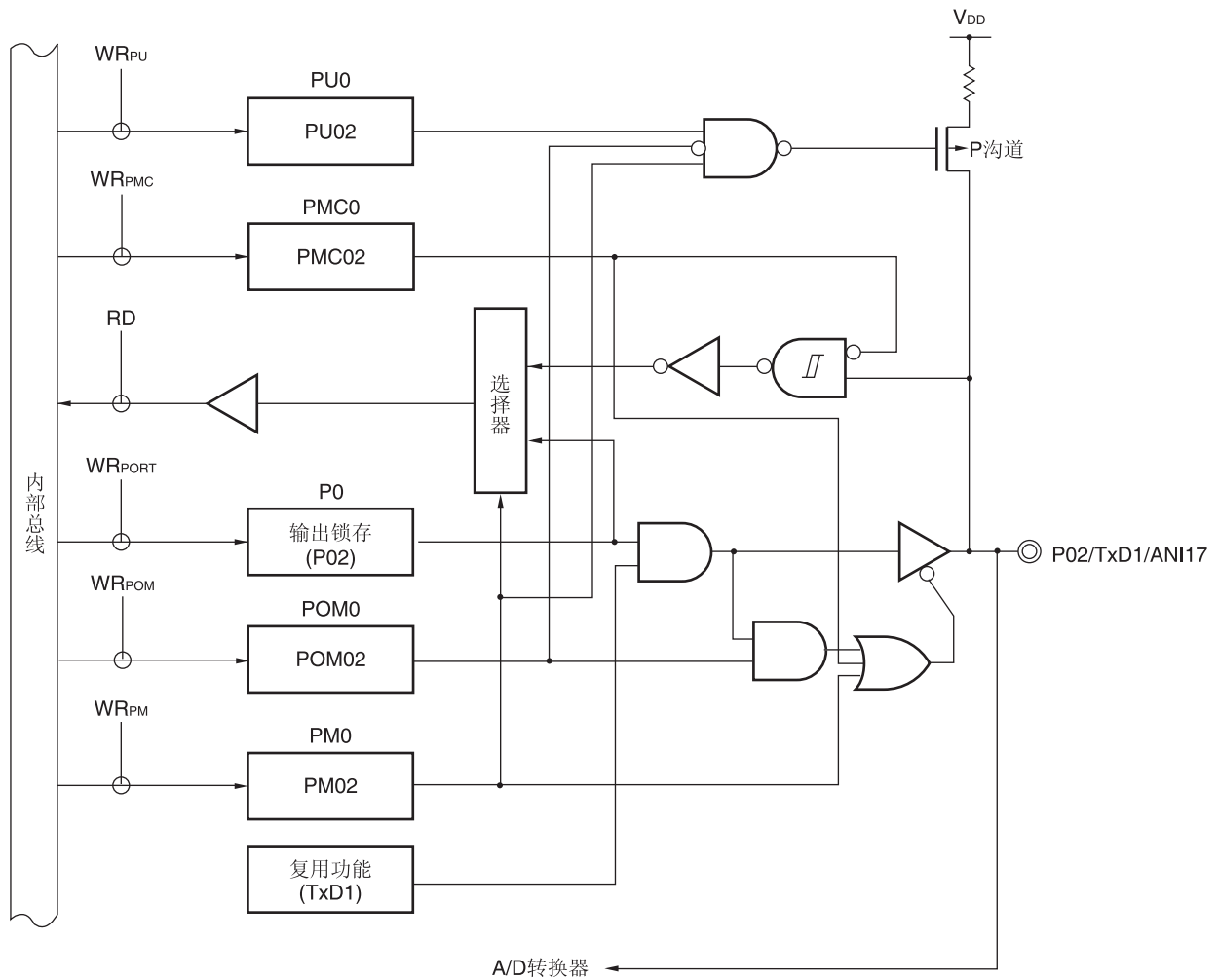
PIM0x: 端口输入模式寄存器 0

POM0x: 端口输出模式寄存器 0

PMC0x: 端口模式控制寄存器 0

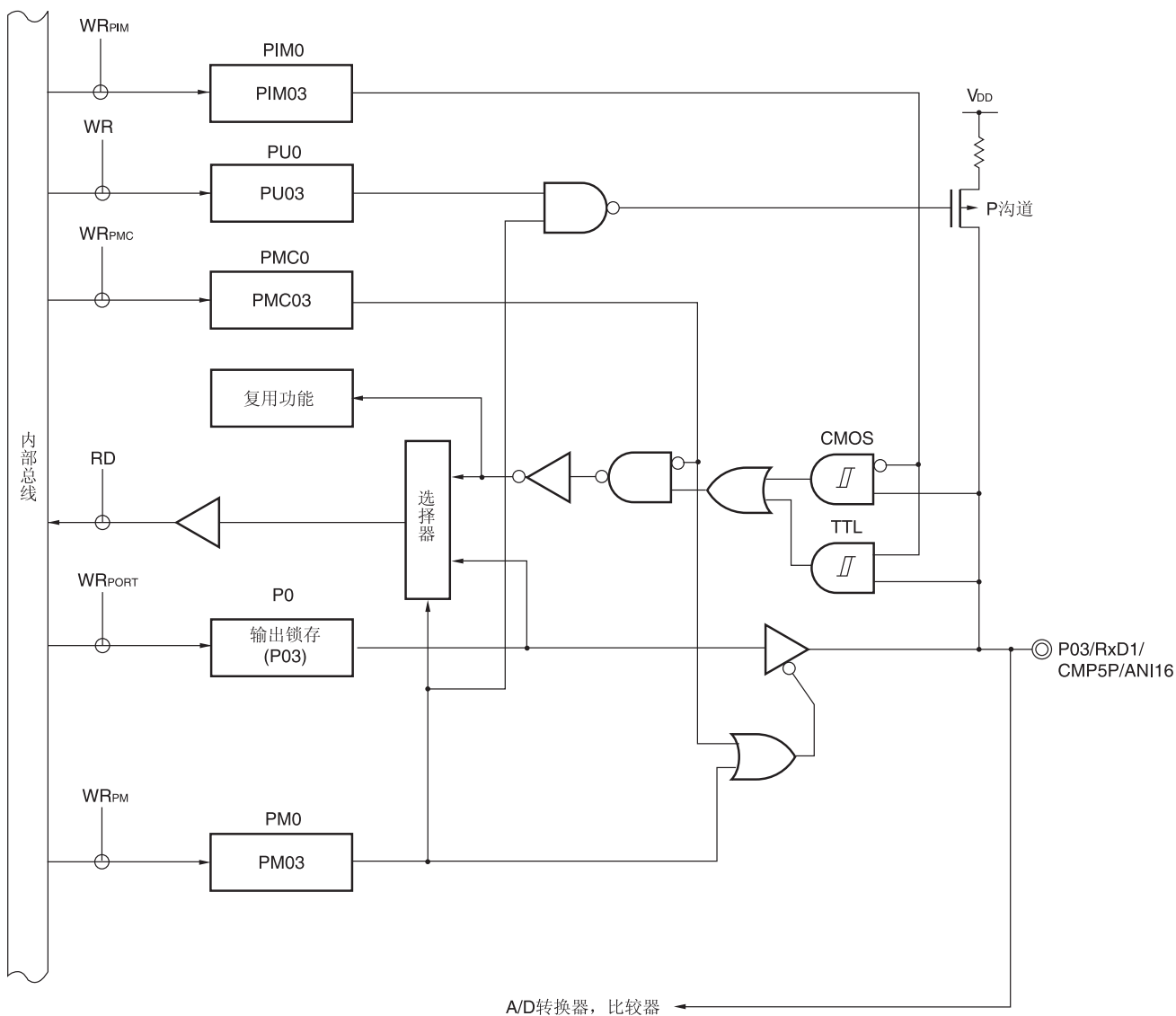
例如，38 引脚产品中端口 0 的框图如图 4-1 至 4-3 所示。

图 4-1. P02 的框图



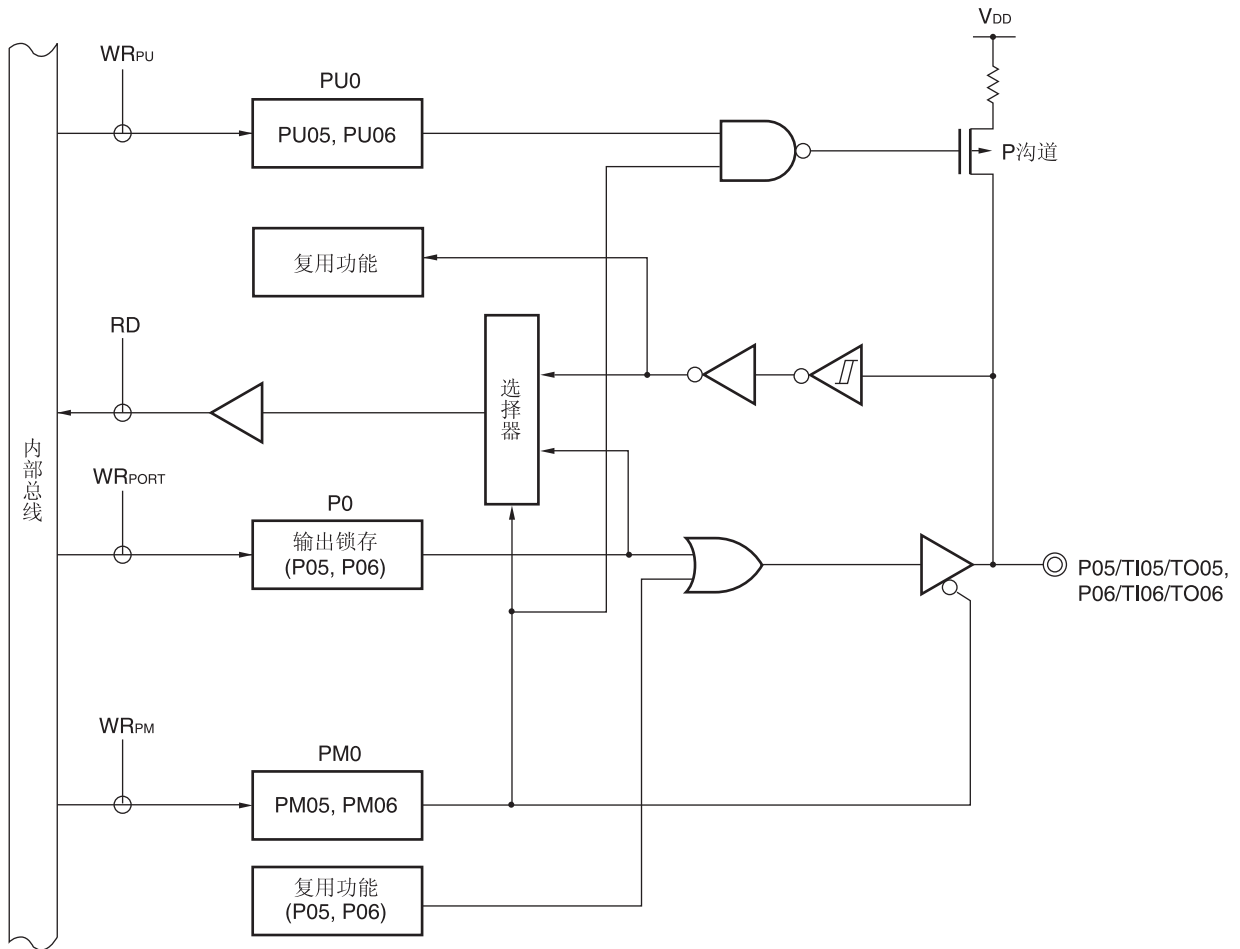
- P0: 端口寄存器 0
- PU0: 上拉电阻选择寄存器 0
- PM0: 端口模式寄存器 0
- POM0: 端口输出模式寄存器 0
- PMC0: 端口模式控制寄存器 0
- RD: 读取信号
- WR_{xx}: 写入信号

图 4-2. P03 的框图



- P0: 端口寄存器 0
- PU0: 上拉电阻选择寄存器 0
- PM0: 端口模式寄存器 0
- PIM0: 端口输入模式寄存器 0
- POM0: 端口输出模式寄存器 0
- PMC0: 端口模式控制寄存器 0
- RD: 读取信号
- WR_{xx}: 写入信号

图 4-3. P05 和 P06 的框图



- P0: 端口寄存器 0
- PU0: 上拉电阻选择寄存器 0
- PM0: 端口模式寄存器 0
- RD: 读取信号
- WR_{xx}: 写入信号

4.2.2 端口 1

端口 1 是具有输出锁存器的输入/输出端口。端口 1 可以通过端口模式寄存器 1(PM1)以 1 位为单位设为输入模式或输出模式。当 P10 至 P12 引脚用作输入端口时，可通过上拉电阻选择寄存器 1(PU1)以 1 位为单位指定使用片上上拉电阻。

P10 和 P11 引脚的输入可利用端口输入模式寄存器 1(PIM1)以 1 位为单位指定为普通输入缓冲器或 TTL 输入缓冲器。

P10 至 P12 引脚的输出可利用端口输出模式寄存器 1(POM1)以 1 位为单位指定为 N 沟开漏输出（V_{DD} 耐压）。

端口 1 也可以兼用作串行接口数据输入/输出、时钟输入/输出、定时器输出和外部中断请求输入。

产生复位信号后，端口 1 为输入模式。

表 4-3. 使用端口 1 寄存器的设置

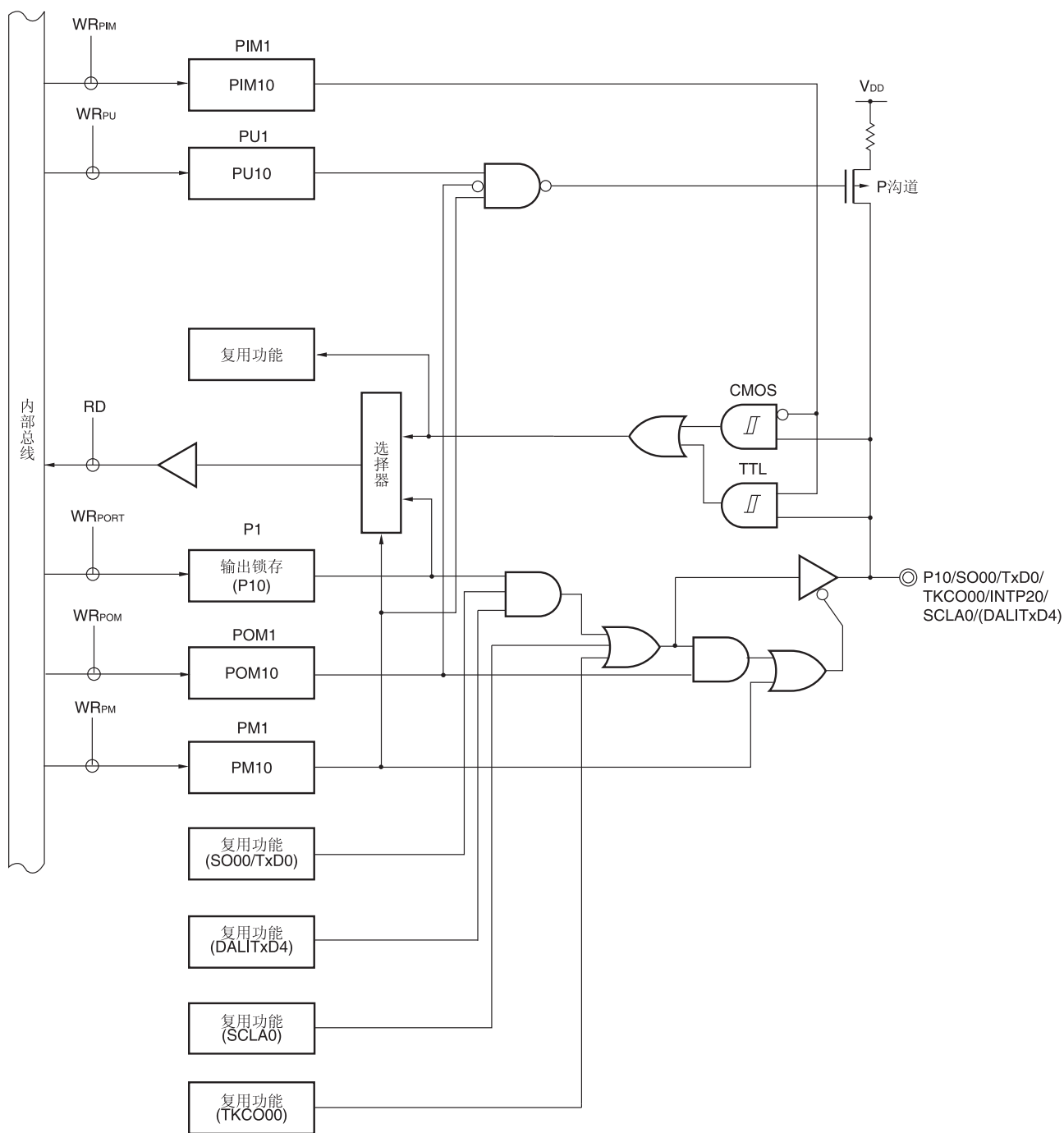
名称	输入/输出	PM1 _x	PIM1 _x	POM1 _x	复用功能设置 ^{注6}	备注
P10	输入	1	0	×	×	CMOS输入
		1	1	×	×	TTL输入
	输出	0	×	0	SO00/TxD0输出 = 1 ^{注1} SCLA0输出 = 0 ^{注2} TKCO00输出 = 0 ^{注3} (DALITxD4输出 = 1 ^{注4})	CMOS 输出
		0	×	1		N沟开漏输出
P11	输入	1	0	×	×	CMOS输入
		1	1	×	×	TTL输入
	输出	0	×	0	TKCO01输出 = 0 ^{注3} SDAA0输出 = 0 ^{注2} (TxRx4输出 = 1 ^{注4})	CMOS 输出
		0	×	1		N沟开漏输出
P12	输入	1	—	×	×	CMOS输入
	输出	0		0	$\overline{\text{SCK00}}$ 输出 = 1 ^{注1} (TKCO03输出 = 0 ^{注5})	CMOS 输出
		0		1		N沟开漏输出

- 注 1. 将 P10/SO00/TxD0/TKCO00/INTP20/SCLA0/(DALITxD4), P12/ $\overline{\text{SCK00}}$ /(TKCO03)用作通用端口时, 请将串行通道允许状态寄存器 0 (SE0)的位 0 (SE00)、串行输出寄存器 0 (SO0)的位 0 (SO00)和串行输出允许寄存器 0 (SOE0)的位 0 (SOE00)设置为与其初始状态相同的值。
2. 将 P10/SO00/TxD0/TKCO00/INTP20/SCLA0/(DALITxD4), P11/SI00/RxD0/TKCO01/INTP21/SDAA0/(TI07)/(DALIRxD4)/(TxRx4)用作通用端口时, 需停止操作串行接口 IICA。
3. 将 P10/SO00/TxD0/TKCO00/INTP20/SCLA0/(DALITxD4), P11/SI00/RxD0/TKCO01/INTP21/SDAA0/(TI07)/(DALIRxD4)/(TxRx4)用作通用端口时, 设置 16 位定时器 KC 输出控制寄存器 01 (TKCIOC01)的位 0 和 1 (TKCTOE00, TKCTOE01)、16 位定时器 KC 输出引脚控制寄存器(TOETKC0)的位 0 和 1 (TOETKC00, TOETKC01)为 0, 即设置为与其初始状态相同的值。
4. 将 P10/SO00/TxD0/TKCO00/INTP20/SCLA0/(DALITxD4), P11/SI00/RxD0/TKCO01/INTP21/SDAA0/(TI07)/(DALIRxD4)/(TxRx4)用作通用端口, 且 PIOR1 被设置为 1 时, 请将串行通道允许状态寄存器 4 (SE4)的位 0 (SE40)、串行输出寄存器 4 (SO4)的位 0 (SO40)和串行输出允许寄存器 4 (SOE4)的位 0 (SOE40)设置为与其初始状态相同的值。
5. 将 P12/ $\overline{\text{SCK00}}$ /(TKCO03)用作通用端口, 且 PIOR1 被设置为 1 时, 设置 16 位定时器 KC 输出控制寄存器 01 (TKCIOC01)的位 3 (TKCTOE03)、16 位定时器 KC 输出引脚控制寄存器(TOETKC0)的位 3 (TOETKC03)为 0, 即设置为与其初始状态相同的值。
6. 括号中的描述表示的是 PIOR1 = 1 的情况。

备注 ×: 忽略
 PM1_x: 端口模式寄存器 1
 PIM1_x: 端口输入模式寄存器 1
 POM1_x: 端口输出模式寄存器 1
 PIOR_x: 外围输入/输出重定向寄存器

例如，38 引脚产品中端口 1 的框图如图 4-4 至 4-6 所示。

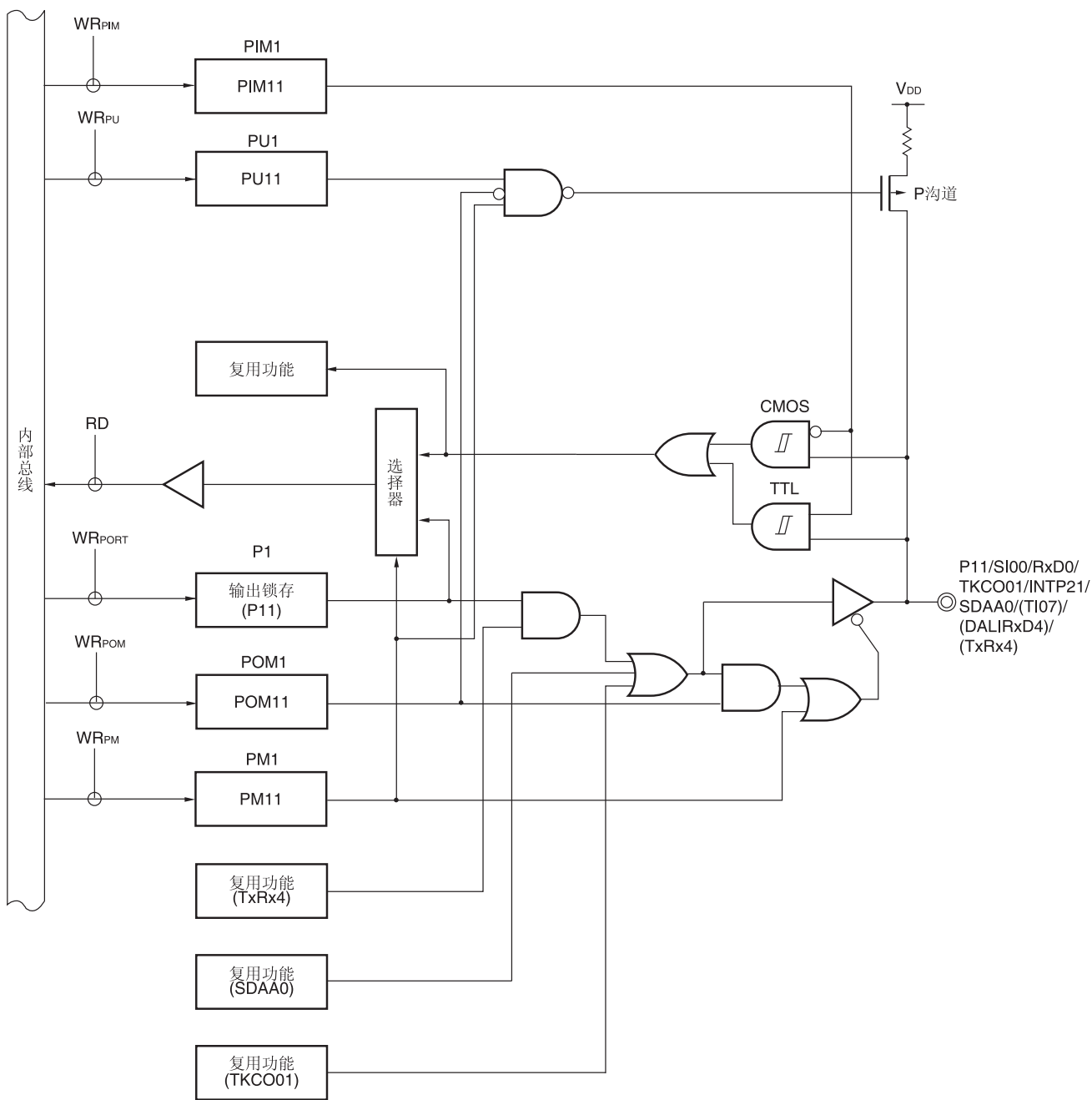
图 4-4. P10 的框图



- P1: 端口寄存器 1
- PU1: 上拉电阻选择寄存器 1
- PM1: 端口模式寄存器 1
- PIM1: 端口输入模式寄存器 1
- POM1: 端口输出模式寄存器 1
- RD: 读取信号
- WR_{xx}: 写入信号

备注 可以通过设置外围输入/输出重定向寄存器 (PIOR1)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式。

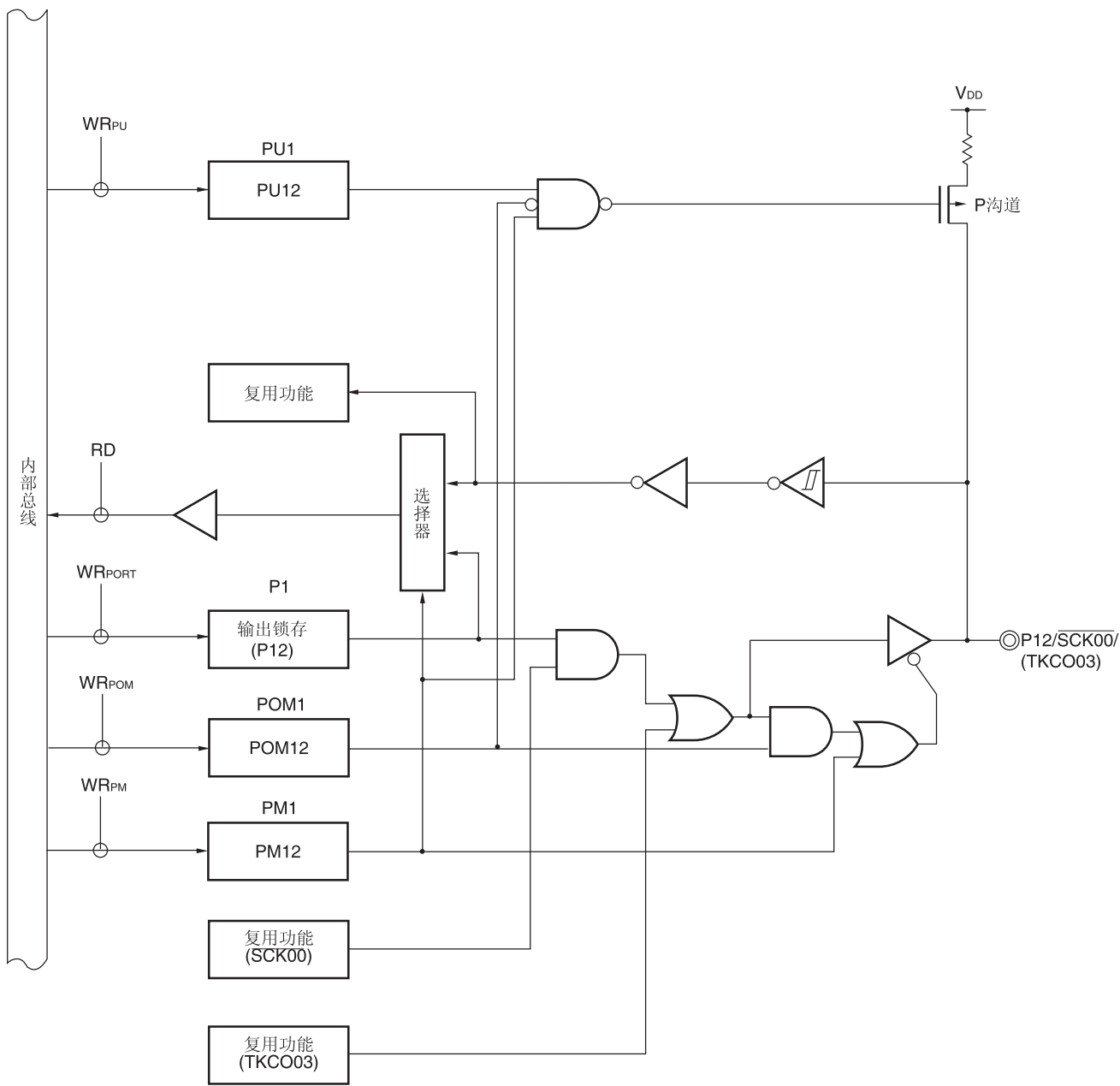
图 4-5. P11 的框图



- P1: 端口寄存器 1
- PU1: 上拉电阻选择寄存器 1
- PM1: 端口模式寄存器 1
- PIM1: 端口输入模式寄存器 1
- POM1: 端口输出模式寄存器 1
- RD: 读取信号
- WR_{xx}: 写入信号

备注 可以通过设置外围输入/输出重定向寄存器(PIOR1)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式。

图 4-6. P12 的框图



- P1: 端口寄存器 1
- PU1: 上拉电阻选择寄存器 1
- PM1: 端口模式寄存器 1
- POM1: 端口输出模式寄存器 1
- RD: 读取信号
- WR_{xx}: 写入信号

备注 可以通过设置外围输入/输出重定向寄存器(PIOR1)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式。

4.2.3 端口 2

端口 2 是具有输出锁存器的输入/输出端口。端口 2 可以通过端口模式寄存器 2(PM2)以 1 位为单位设为输入模式或输出模式。

端口 2 也可以兼用作 A/D 转换器模拟输入、基准电压输入和比较器模拟输入。

将 P20/ANI0 至 P22/ANI2、P24/ANI4 至 P27/ANI7 用作数字输入引脚时，通过 A/D 端口配置寄存器(ADPC)将其设置为数字输入/输出模式，通过 PM2 寄存器将其设置为输入模式。从高位开始使用这些引脚。

将 P20/ANI0 至 P22/ANI2、P24/ANI4 至 P27/ANI7 用作数字输出引脚时，通过 ADPC 寄存器将其设置为数字输入/输出模式，通过 PM2 寄存器将其设置为输出模式。从高位开始使用这些引脚。

将 P20/ANI0 至 P22/ANI2、P24/ANI4 至 P27/ANI7 用作模拟输入引脚时，通过 A/D 端口配置寄存器(ADPC)将其设置为模拟输入模式，通过 PM2 寄存器将其设置为输入模式。从低位开始使用这些引脚。

表 4-4. 使用端口 2 时寄存器的设置

名称	输入/输出	PM2x	ADPC	复用功能设置	备注
P2n	输入	1	01至n+1H	-	将P2n用作端口时，从高位开始使用这些引脚。
	输出	0	01至n+1H		

备注 x: 忽略
 PM2x: 端口模式寄存器 2
 ADPC: A/D 端口配置寄存器

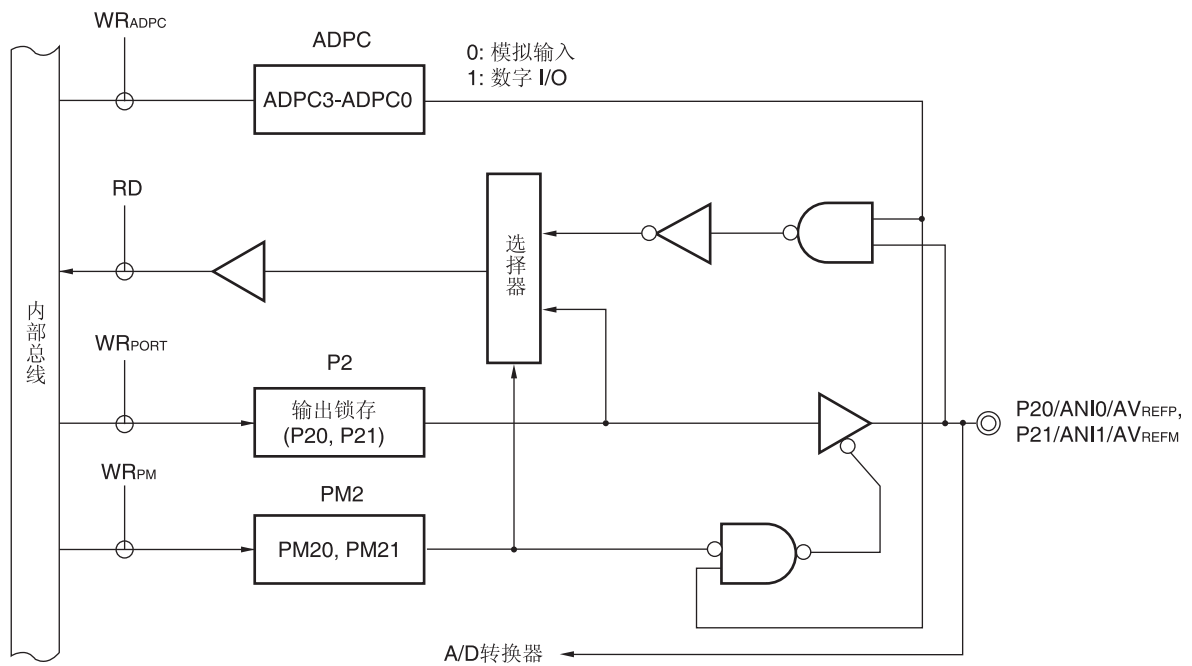
表 4-5. P20/ANI0 至 P22/ANI2、P24/ANI4 至 P27/ANI7 引脚功能的设置

ADPC寄存器	PM2寄存器	ADS寄存器	P20/ANI0至P22/ANI2, P24/ANI4至P27/ANI7引脚
数字输入/输出选择	输入模式	-	数字输入
	输出模式	-	数字输出
模拟输入选择	输入模式	选择ANI。	模拟输入(被转换)
		不选择ANI。	模拟输入(不被转换)
	输出模式	选择ANI。	禁止设置
		不选择ANI。	

产生复位信号后，从 P20/ANI0 至 P22/ANI2、P24/ANI4 至 P27/ANI7 的所有引脚为模拟输入模式。

例如，38 引脚产品中端口 2 的框图如图 4-7、4-8 所示。

图 4-7. P20 和 P21 的框图



ADPC: A/D 端口配置寄存器

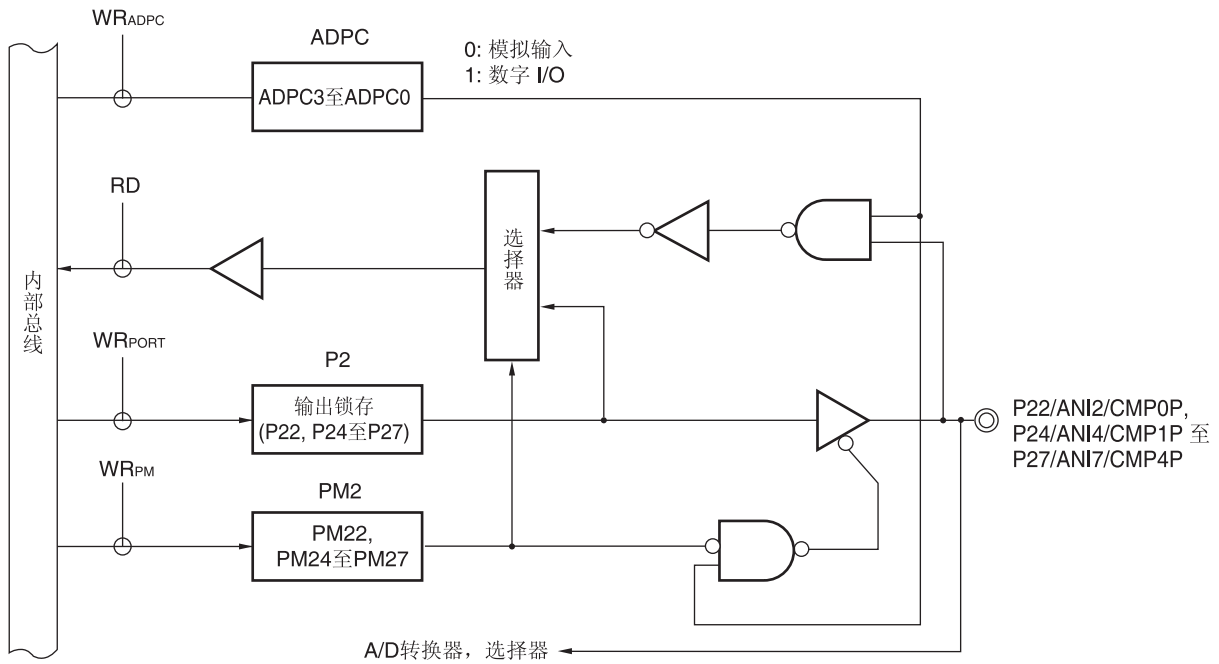
P2: 端口寄存器 2

PM2: 端口模式寄存器 2

RD: 读取信号

WR_{xx}: 写入信号

图 4-8. P22 至 P27 的框图



- ADPC: A/D 端口配置寄存器
- P2: 端口寄存器 2
- PM2: 端口模式寄存器 2
- RD: 读取信号
- WR_{xx} : 写入信号

4.2.4 端口 3

端口 3 是具有输出锁存器的输入/输出端口。端口 3 可以通过端口模式寄存器 3(PM3)以 1 位为单位设为输入模式或输出模式。当 P30 和 P31 引脚用作输入端口时，可通过上拉电阻选择寄存器 3(PU3)以 1 位为单位使用片上上拉电阻。

端口 3 也可以兼用作外部中断请求输入、实时时钟校正时钟输出以及定时器输入/输出。

产生复位信号后，P30 和 P31 为模拟输入。

表 4-6. 使用端口 3 寄存器的设置

名称	输入/输出	PM3 _x	PMC3 _x	复用功能设置	备注
P30	输入	1	-	×	
	输出	0		RTC1HZ输出 = 0 ^{注1}	
P31	输入	1	-	×	
	输出	0		TO03输出 = 0 ^{注2}	

注 1. 将 P30/RTC1HZ/INTP3 用作通用端口时，设置实时时钟控制寄存器 0 (RTCC0)的位 5 (RCLOE1)为 0，即设置为与其初始状态相同的值。

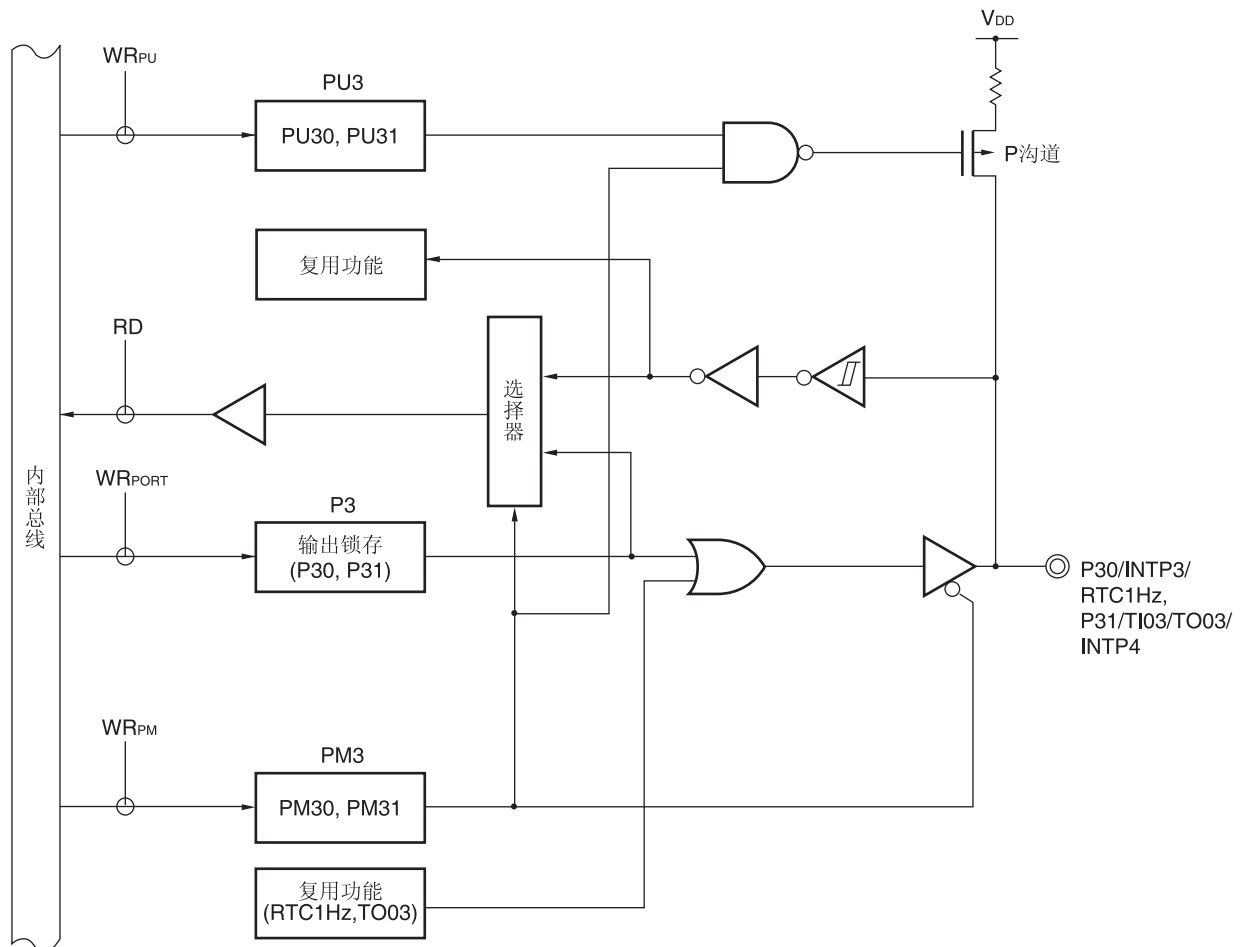
2. 将 P31/TI03/TO03/INTP4 用作通用端口时，设置定时器输出寄存器 0 (TO0)的位 3 (TO03)、定时器输出允许寄存器 0 (TOE0)的位 3 (TOE03) 为 0，即设置为与其初始状态相同的值。

备注 ×: 忽略

PM3_x: 端口模式寄存器 3

例如，38 引脚产品中端口 3 的框图如图 4-9 所示。

图 4-9. P30 和 P31 的框图



- P3: 端口寄存器 3
- PU3: 上拉电阻选择寄存器 3
- PM3: 端口模式寄存器 3
- RD: 读取信号
- WR_{xx} : 写入信号

4.2.5 端口 4

端口 4 是具有输出锁存器的输入/输出端口。端口 4 可以通过端口模式寄存器 4(PM4)以 1 位为单位设为输入模式或输出模式。当 P40 引脚用作输入端口时，可通过上拉电阻选择寄存器 4(PU4)以 1 位为单位指定使用片上上拉电阻。

端口 4 也可以兼用于闪存编程器/调试器的数据输入/输出。

产生复位信号后，端口 4 为输入模式。

表 4-7. 使用端口 4 时寄存器的设置

名称	输入/输出	PM4x	复用功能设置	备注
P40	输入	1	×	
	输出	0	×	

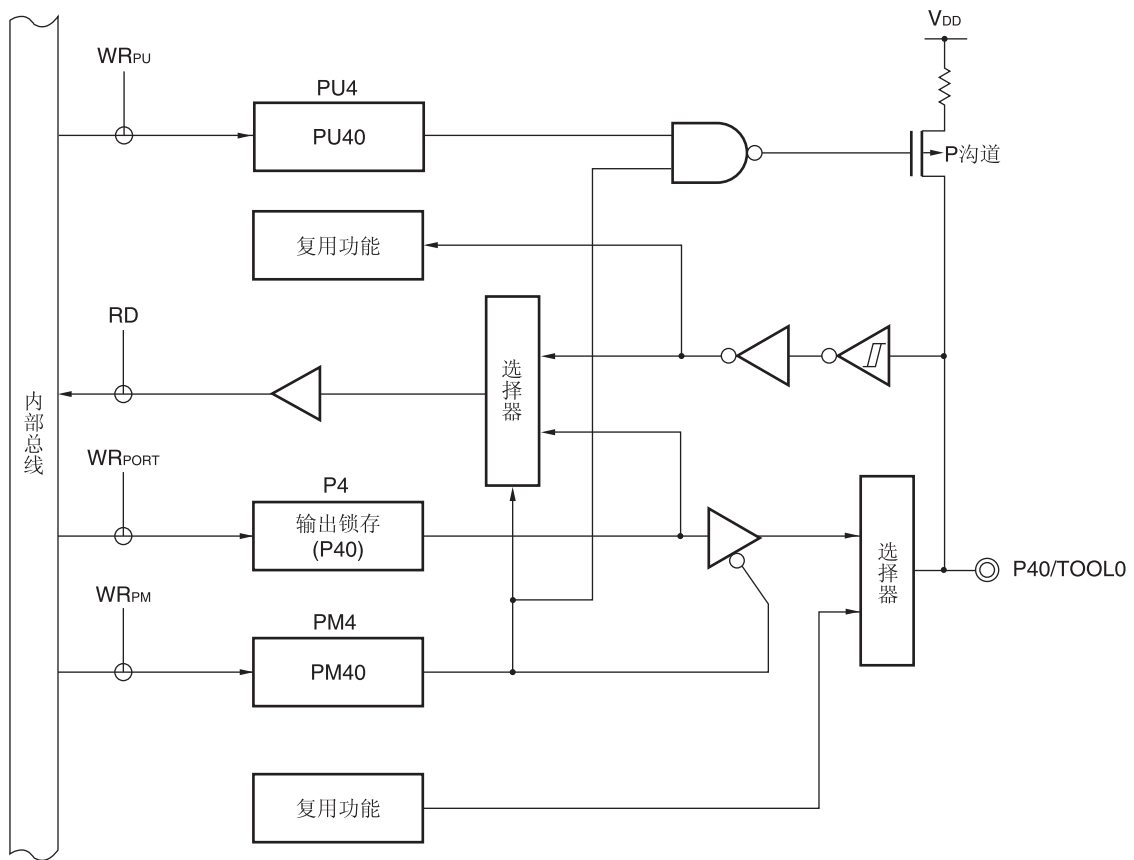
注意事项 连接工具时，P40 引脚不能用作端口引脚。

备注 ×: 忽略

PM4x: 端口模式寄存器 4

例如，38 引脚产品中端口 4 的框图如图 4-10 所示。

图 4-10. P40 的框图



- P4: 端口寄存器 4
 PU4: 上拉电阻选择寄存器 4
 PM4: 端口模式寄存器 4
 RD: 读取信号
 WR_{xx}: 写入信号

4.2.6 端口 7

端口 7 是具有输出锁存器的输入/输出端口。端口 7 可以通过端口模式寄存器 7(PM7)以 1 位为单位设为输入模式或输出模式。当 P75 至 P77 引脚用作输入端口时，可通过上拉电阻选择寄存器 7(PU7)以 1 位为单位使用片上上拉电阻。

端口 7 也可以用作外部中断请求输入。

产生复位信号后，端口 7 为输入模式。

表 4-8. 使用端口 7 时寄存器的设置

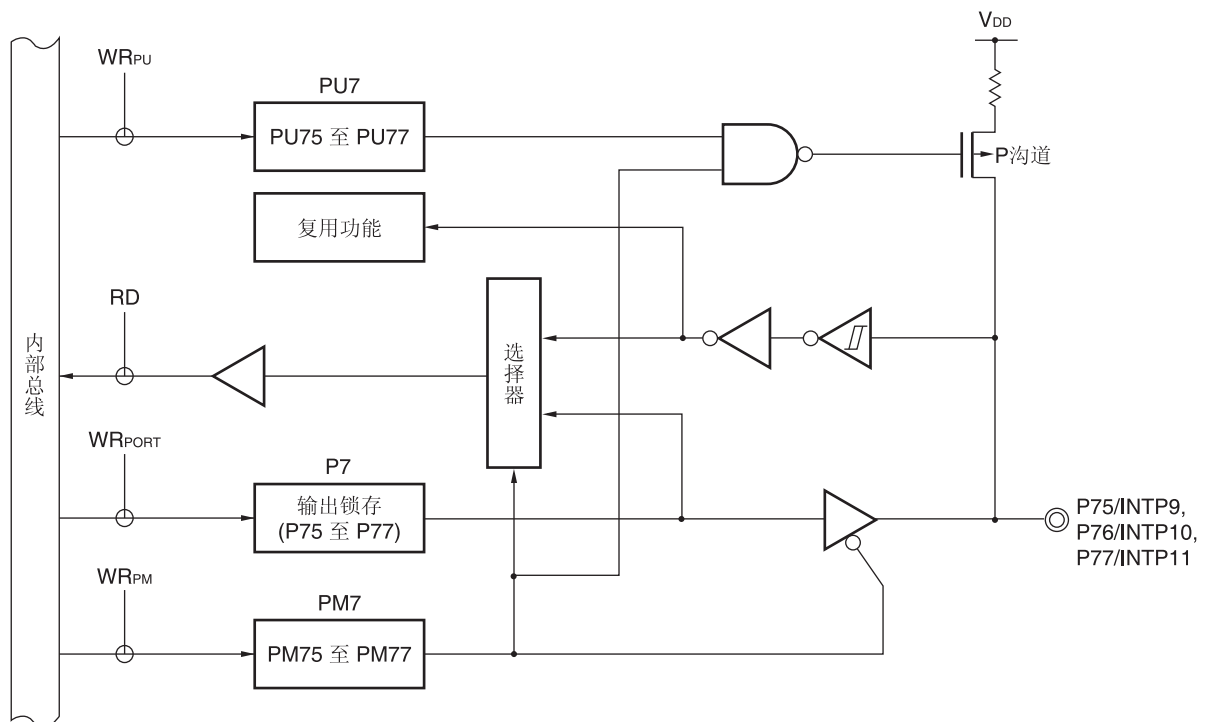
名称	输入/输出	PM7 _x	复用功能设置	备注
P75	输入	1	×	
	输出	0	×	
P76	输入	1	×	
	输出	0	×	
P77	输入	1	×	
	输出	0	×	

备注 ×: 忽略

PM7_x: 端口模式寄存器 7

例如，38 引脚产品中端口 7 的框图如图 4-11 所示。

图 4-11. P70 的框图



- P7: 端口寄存器 7
- PU7: 上拉电阻选择寄存器 7
- PM7: 端口模式寄存器 7
- RD: 读取信号
- WR_{xx}: 写入信号

4.2.7 端口 12

P120 是具有输出锁存器的输入/输出端口。端口 12 可以通过端口模式寄存器 12(PM12)以 1 位为单位设为输入模式或输出模式。当 P120 引脚用作输入端口时，可通过上拉电阻选择寄存器 12 (PU12)使用片上上拉电阻。

P121 至 P124 是 4 位输入专用端口。

P120 引脚的输入可利用端口模式控制寄存器 12(PMC12)以 1 位为单位指定为模拟输入或数字输入。

端口 12 也可以用作 A/D 转换器模拟输入、连接主系统时钟用谐振器、连接副系统时钟用谐振器、主系统时钟用外部时钟输入和副系统时钟用外部时钟输入。

产生复位信号后，P120 为模拟输入，P121 至 P127 为输入模式。

表 4-9. 使用端口 12 时寄存器的设置

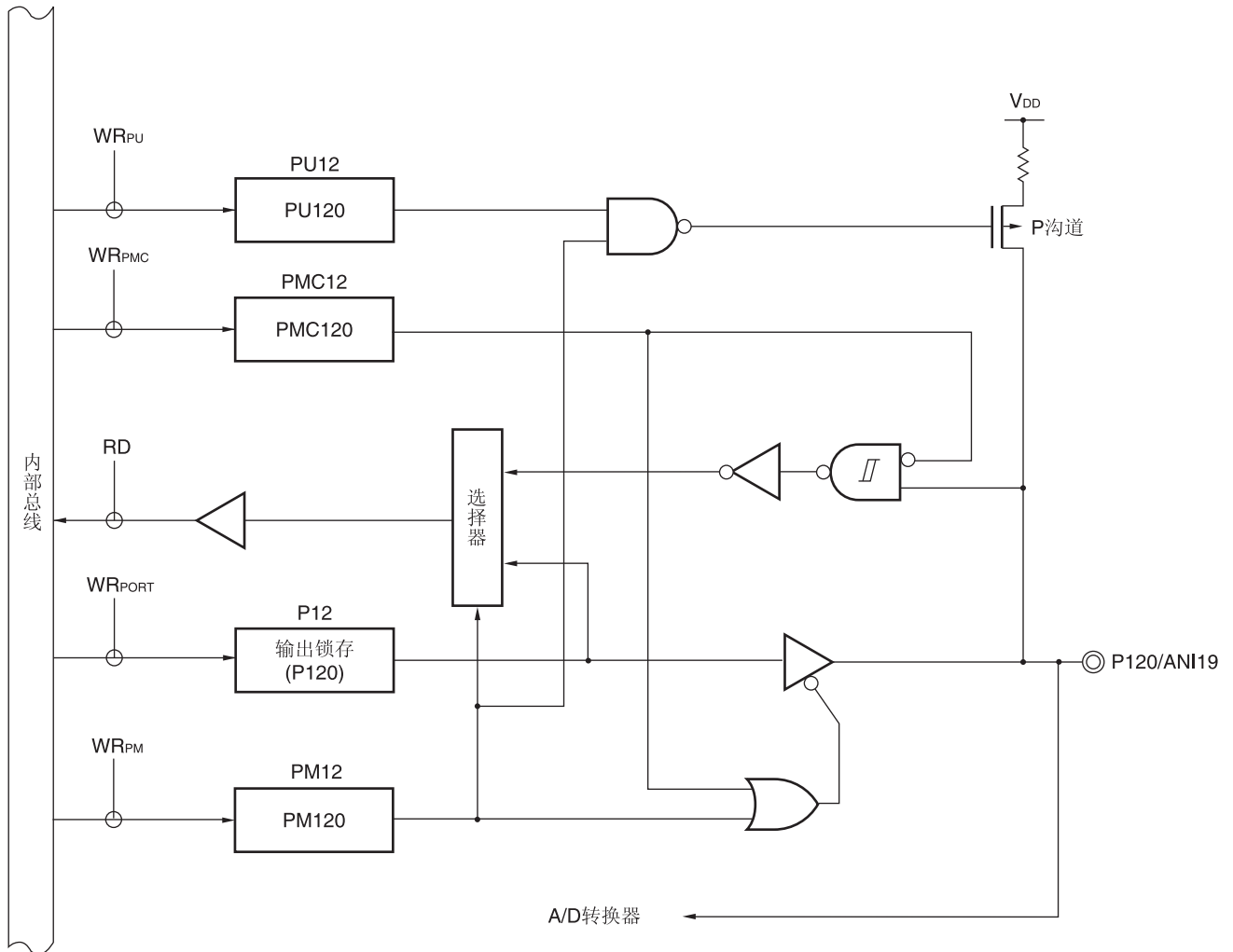
名称	输入/输出	PM12 _x	PMC12 _x	复用功能设置	备注
P120	输入	1	0	×	
	输出	0	0	×	
P121	输入	-	-	CMC寄存器的OSCSEL位 = 0 或EXCLK位 = 1	
P122	输入	-	-	CMC寄存器的OSCSEL位 = 0	
P123	输入	-	-	CMC寄存器的OSCSELS位 = 0 或EXCLKS位 = 1	
P124	输入	-	-	CMC寄存器的OSCSELS位 = 0	

注意事项 P121 至 P124 的功能仅在解除复位后设置一次。设置为连接 X1, XT1 振荡器/外部时钟输入的端口时，在执行复位之前不能用作输入端口。

备注 x: 忽略
 PM12_x: 端口模式寄存器 12
 PMC12_x: 端口模式控制寄存器 12

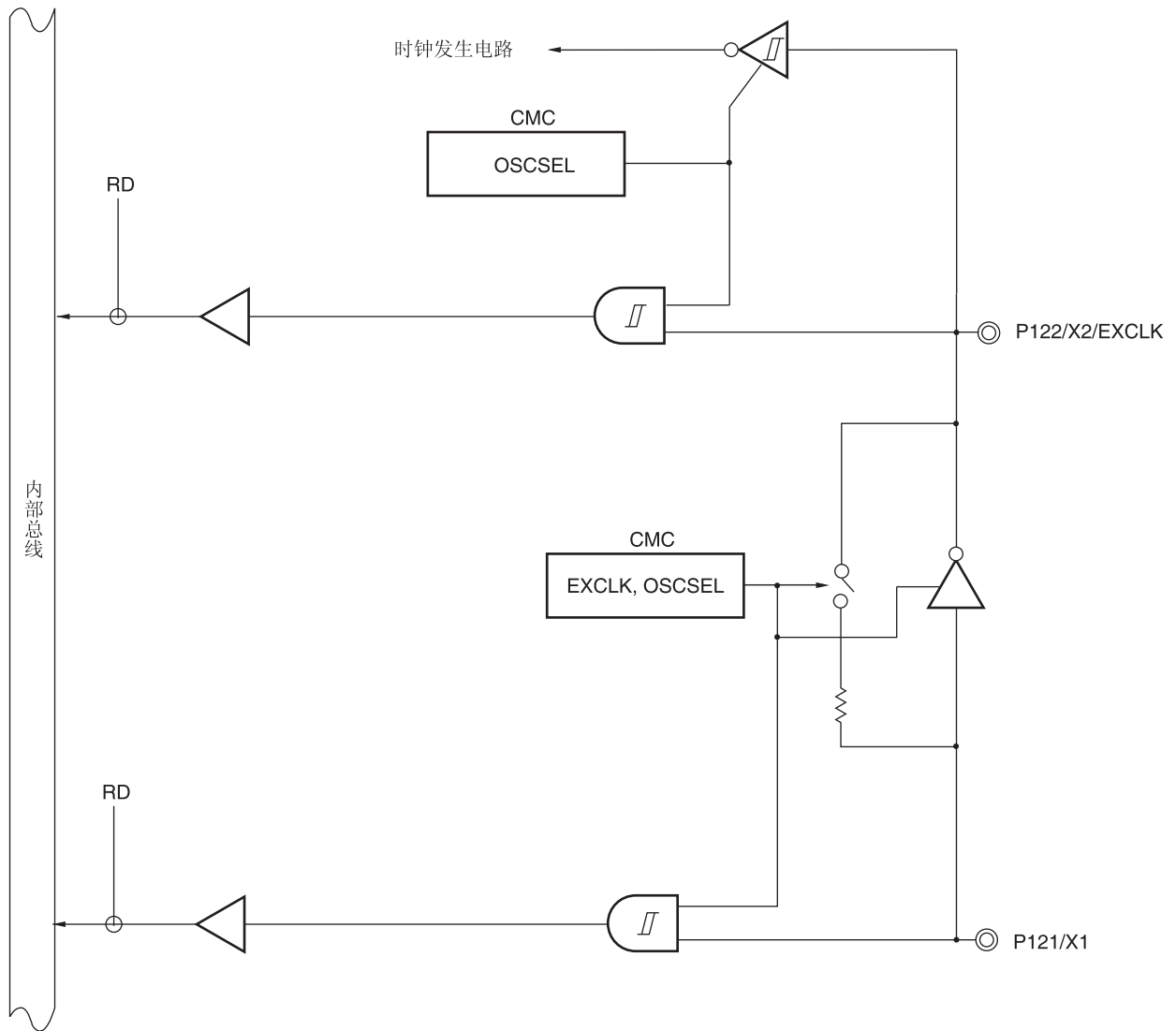
例如，38 引脚产品中端口 12 的框图如图 4-12 至 4-14 所示。

图 4-12. P120 的框图



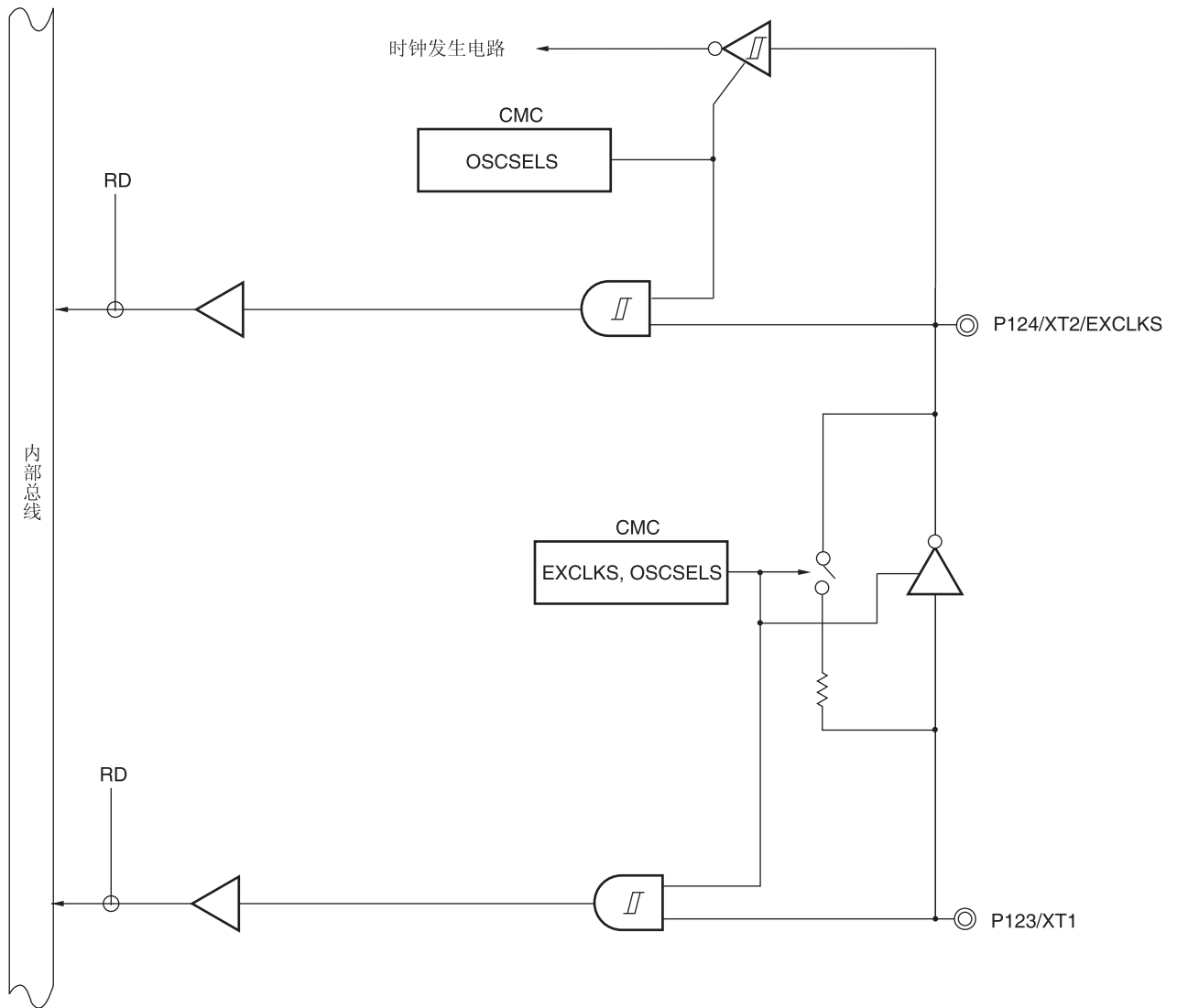
- P12: 端口寄存器 12
- PU12: 上拉电阻选择寄存器 12
- PM12: 端口模式寄存器 12
- PMC12: 端口模式控制寄存器 12
- RD: 读取信号
- WR_{xx}: 写入信号

图 4-13. P121 和 P122 的框图



CMC: 时钟操作模式控制寄存器
RD: 读取信号

图 4-14. P123 和 P124 的框图



CMC: 时钟操作模式控制寄存器
RD: 读取信号

4.2.8 端口 13

P13 是 1 位输入专用端口。

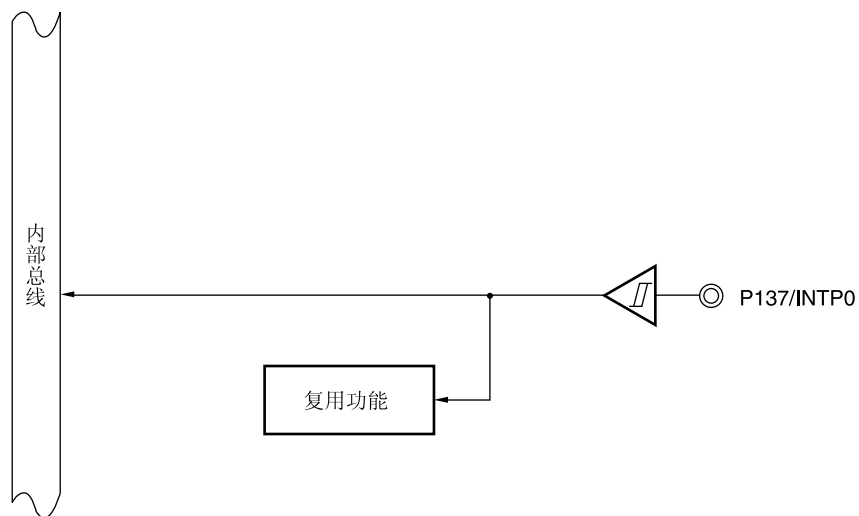
端口 13 也可以兼用作外部中断请求输入。

表 4-10. 使用端口 13 时寄存器的设置

名称	输入/输出	复用功能设置	备注
P137	输入	×	

例如，38 引脚产品中端口 13 的框图如图 4-15 所示。

图 4-15. P137 的框图



4.2.9 端口 14

端口 14 是具有输出锁存器的输入/输出端口。端口 14 可以通过端口模式寄存器 14(PM14)以 1 位为单位设为输入模式或输出模式。当 P147 引脚用作输入端口时，可通过上拉电阻选择寄存器 14(PU14)以 1 位为单位使用片上上拉电阻。

P147 引脚的输入可利用端口模式控制寄存器 14(PMC14)以 1 位为单位指定为模拟输入或数字输入。

端口 14 也可以兼用作 A/D 转换器模拟输入和比较器外部基准电压输入。

产生复位信号后，端口 14 为输入模式。

表 4-11. 使用端口 14 时寄存器的设置

名称	输入/输出	PM14x	PMC14x	复用功能设置	备注
P147	输入	1	0	×	
	输出	0	0	×	

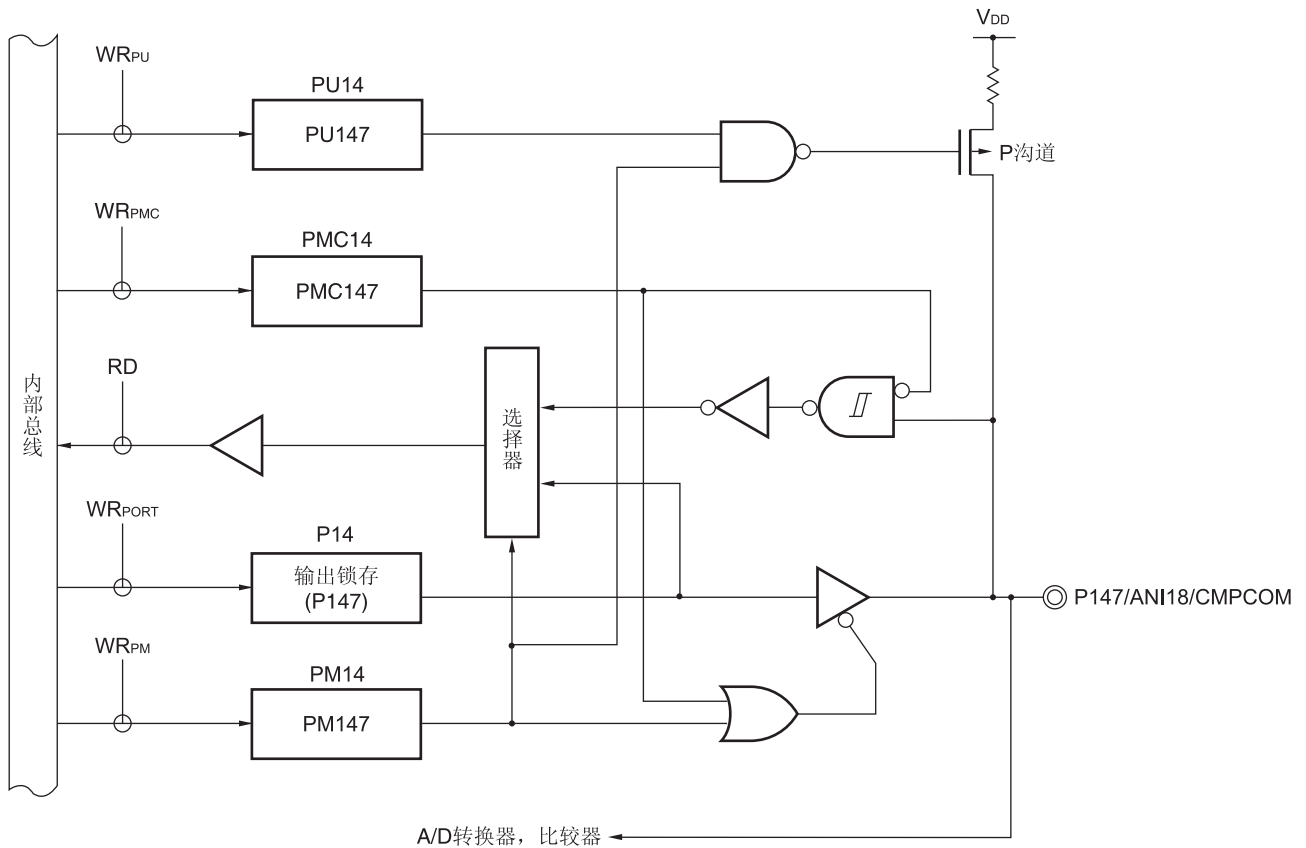
备注 ×: 忽略

PM14x: 端口模式寄存器 14

PMC14x: 端口模式控制寄存器 14

例如，38 引脚产品中端口 14 的框图如图 4-16 所示。

图 4-16. P147 的框图



- P14: 端口寄存器 14
- PU14: 上拉电阻选择寄存器 14
- PM14: 端口模式寄存器 14
- PMC14: 端口模式控制寄存器 14
- RD: 读取信号
- WR_{xx}: 写入信号

4.2.10 端口 20

端口 20 是具有输出锁存器的输入/输出端口。端口 20 可以通过端口模式寄存器 20(PM0)以 1 位为单位设为输入模式或输出模式。当 P200 至 P206 引脚用作输入端口时，可通过上拉电阻选择寄存器 20(PU20)以 1 位为单位使用片上上拉电阻。

P200 至 P206 引脚的输出可利用端口输出模式寄存器 20 (POM20)以 1 位为单位指定为 N 沟开漏输出 (V_{DD} 耐压)。

端口 20 也可以兼用作串行接口数据输入/输出、定时器输入/输出和外部中断请求。

产生复位信号后，端口 20 为输入模式。

表 4-12. 使用端口 20 时寄存器的设置

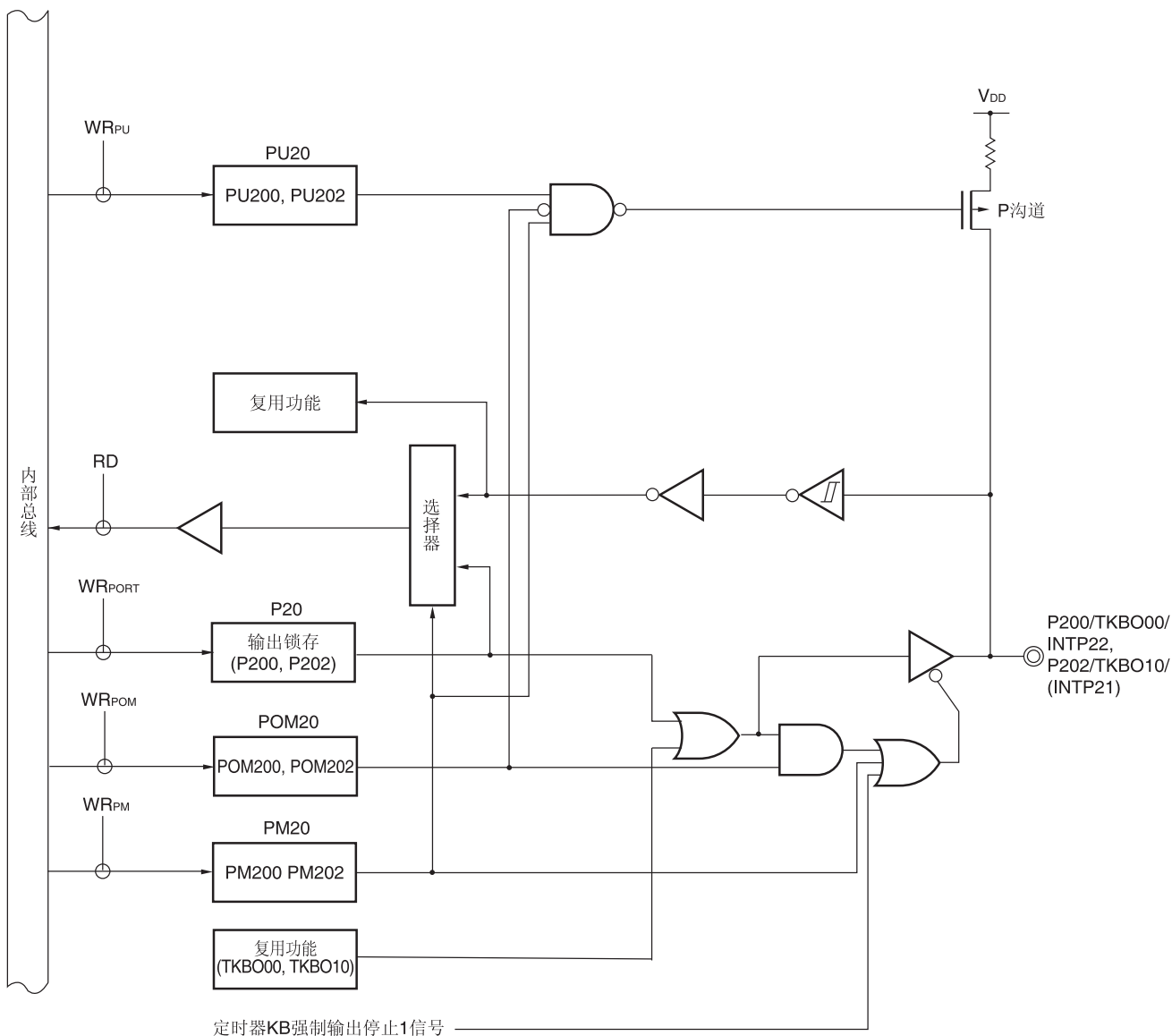
名称	输入/输出	PM1 _x	POM1 _x	复用功能设置	备注
P200	输入	1	×	×	
	输出	0	0	TKBO00输出 = 0 ^{注1}	CMOS输出
		0	1		N沟开漏输出
P201	输入	1	×	×	
	输出	0	0	TKBO01输出 = 0 ^{注1}	CMOS输出
		0	1		N沟开漏输出
P202	输入	1	×	×	
	输出	0	0	TKBO10输出 = 0 ^{注1}	CMOS输出
		0	1		N沟开漏输出
P203	输入	1	×	×	
	输出	0	0	TKBO11输出 = 0 ^{注1}	CMOS输出
		0	1	TKCO02输出 = 0 ^{注2}	N沟开漏输出
P204	输入	1	×	×	
	输出	0	0	TKBO20输出 = 0 ^{注1}	CMOS输出
		0	1	TKCO03输出 = 0 ^{注2}	N沟开漏输出
P205	输入	1	×	×	
	输出	0	0	TKBO21输出 = 0 ^{注1}	CMOS输出
		0	1	TKCO04输出 = 0 ^{注2} DALITx4输出 = 1 ^{注3}	N沟开漏输出
P206	输入	1	×	×	
	输出	0	0	TKCO05输出 = 0 ^{注2}	CMOS输出
		0	1	TxRx4输出 = 1 ^{注3}	N沟开漏输出

- 注 1. 将 P200/TKBO00/INTP22, P201/TKBO01, P202/TKBO10/(INTP21), P203/TKBO11/TKCO02/(INTP20), P204/TKBO20/TKCO03 或 P205/TKBO21/TKCO04/DALITxD4 用作通用端口时, 设置 16 位定时器 KB 输出控制寄存器 n1 (TKBIOCn1)的位 0 和 1 (TKBTOEn0, KTBTOEn1)为 0, 即设置为与其初始状态相同的值。
2. 将 P203/TKBO11/TKCO02/(INTP20), P204/TKBO20/TKCO03, P205/TKBO21/TKCO04/DALITxD4 或 P206/TKCO05/DALIRxD4/TxRx4/INTP23 用作通用端口时, 设置 16 位定时器 KC 输出控制寄存器 01 (TKCIOC01)的位 2 至 5 (TKCTOE02 至 TKCTOE05)、16 位定时器 KC 输出引脚控制寄存器(TOETKC0)的位 2 至 5 (TOETKC02 至 TOETKC05)为 0, 即设置为与其初始状态相同的值。
3. 将 P205/TKBO21/TKCO04/DALITxD4 或 P206/TKCO05/DALIRxD4/TxRx4/INTP23 用作通用端口时, 请将串行通道允许状态寄存器 4 (SE4)的位 0 (SE40)、串行输出寄存器 4 (SO4)的位 0 (SO40)和串行输出允许寄存器 4 (SOE4)的位 0 (SOE40)设置为与其初始状态相同的值。

备注 ×: 忽略
 PM20_x: 端口模式寄存器 20
 POM20_x: 端口输出模式寄存器 20

例如, 38 引脚产品中端口 4 的框图如图 4-17 至 20-22 所示。

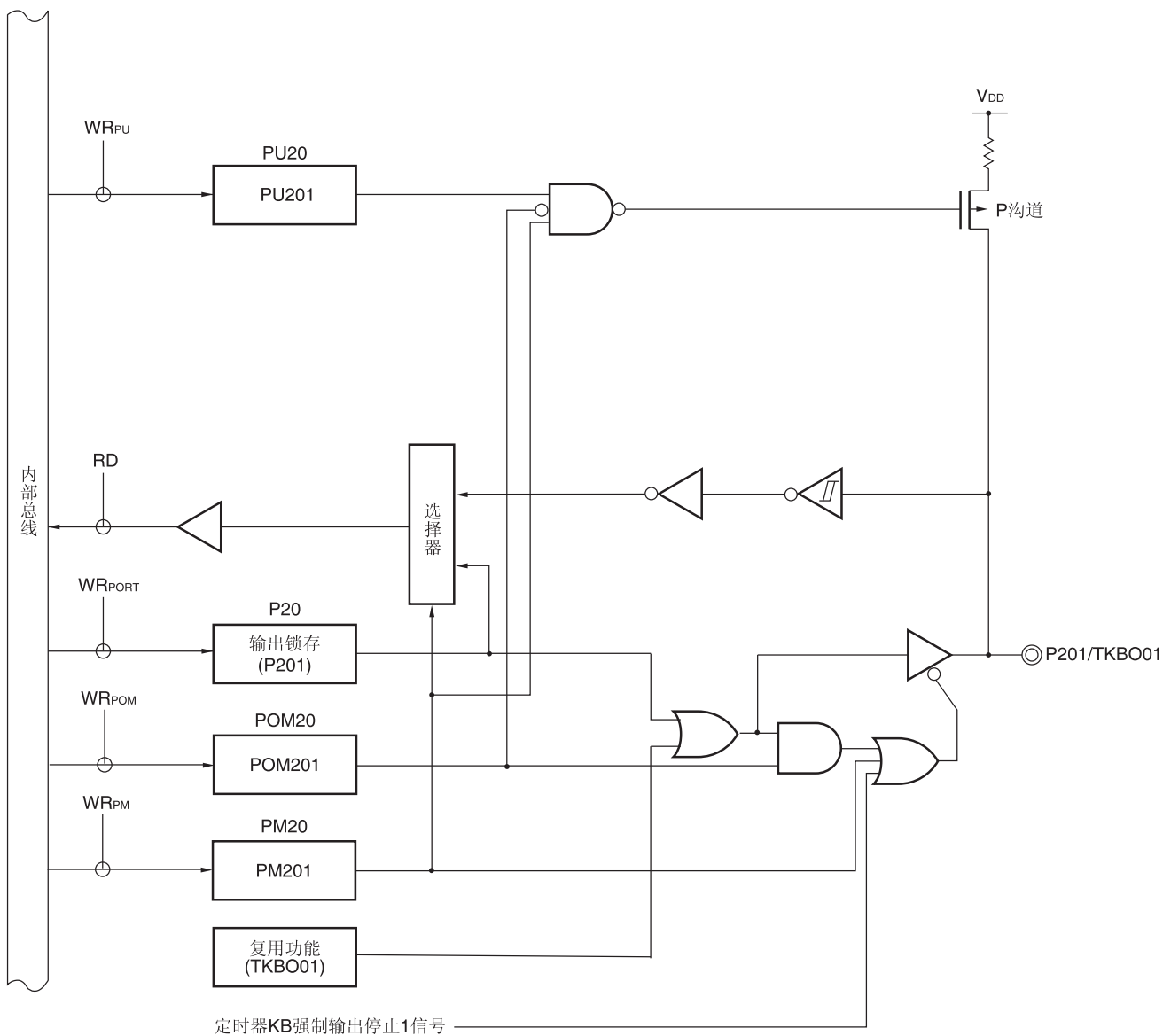
图 4-17. P200 和 P202 的框图



- P20: 端口寄存器 20
- PU20: 上拉电阻选择寄存器 20
- PM20: 端口模式寄存器 20
- POM20: 端口输出模式寄存器 20
- RD: 读取信号
- WR_{xx} : 写入信号

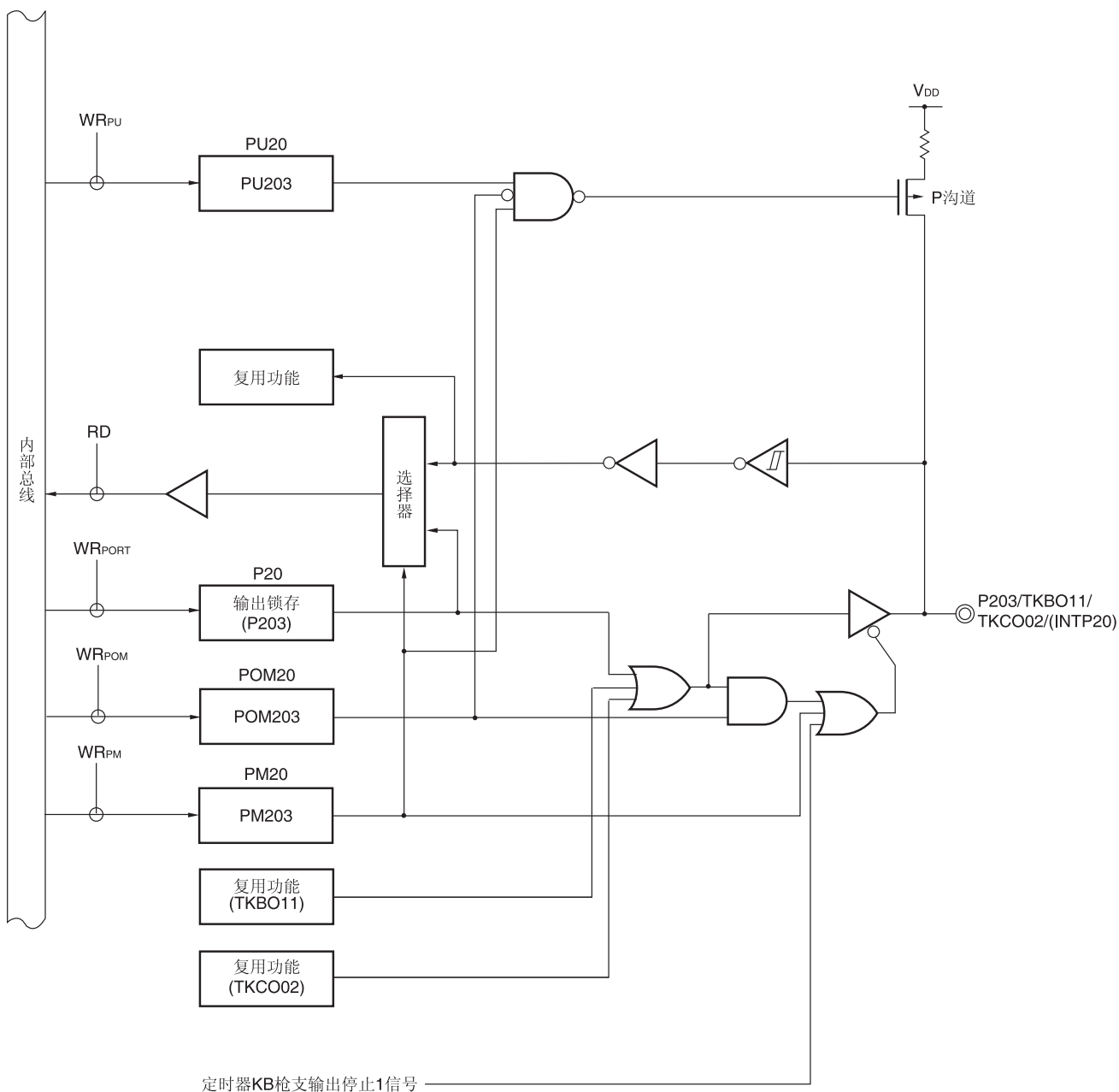
备注 可以通过设置外围输入/输出重定向寄存器(PIOR1)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式。

图 4-18. P201 的框图



- P20: 端口寄存器 20
- PU20: 上拉电阻选择寄存器 20
- PM20: 端口模式寄存器 20
- POM20: 端口输出模式寄存器 20
- RD: 读取信号
- WR_{xx}: 写入信号

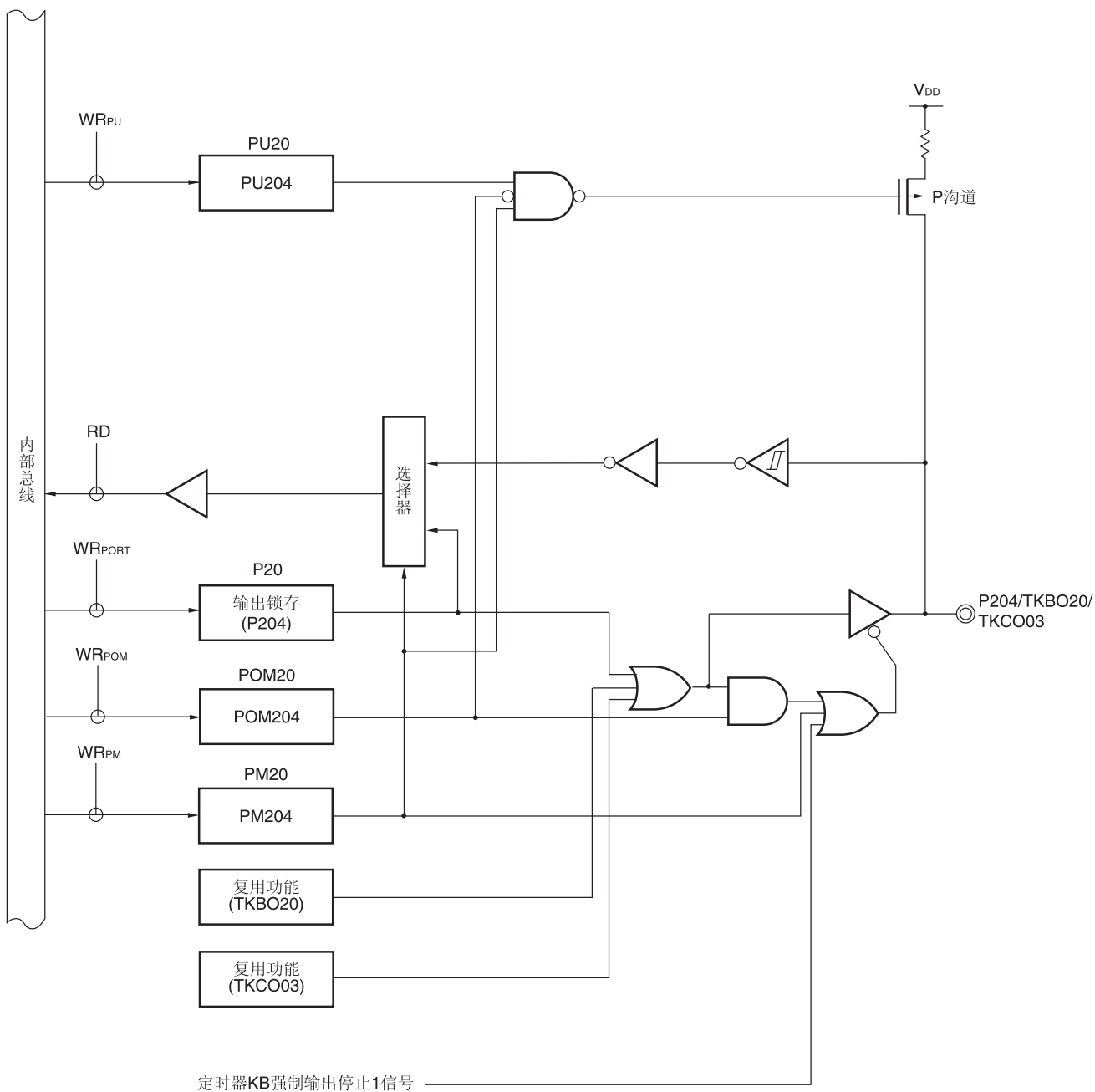
图 4-19. P203 的框图



- P20: 端口寄存器 20
- PU20: 上拉电阻选择寄存器 20
- PM20: 端口模式寄存器 20
- POM20: 端口输出模式寄存器 20
- RD: 读取信号
- WR_{xx}: 写入信号

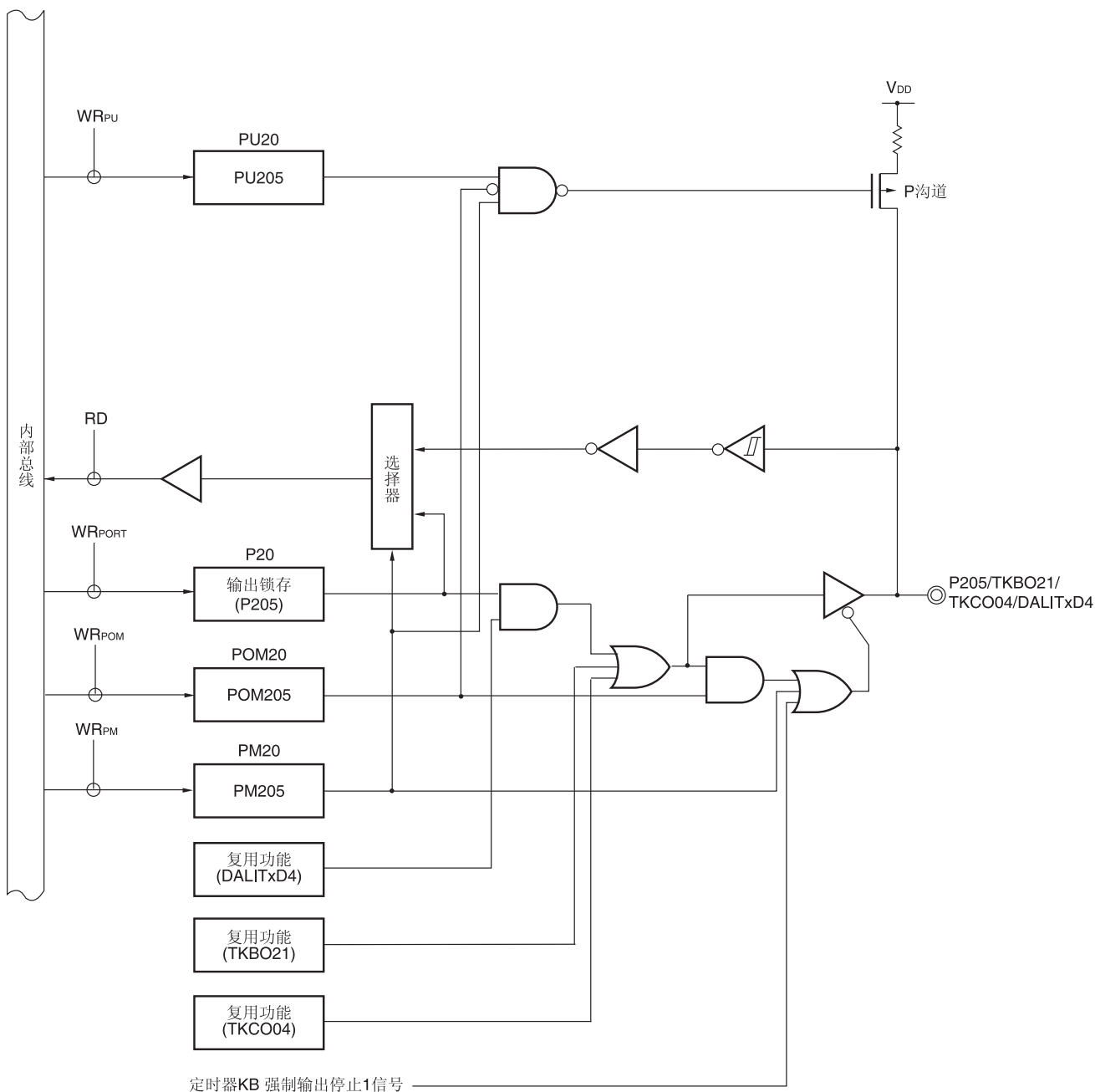
备注 可以通过设置外围输入/输出重定向寄存器(PIOR1)来指定上图括号内的功能。请参阅图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式。

图 4-20. P204 的框图



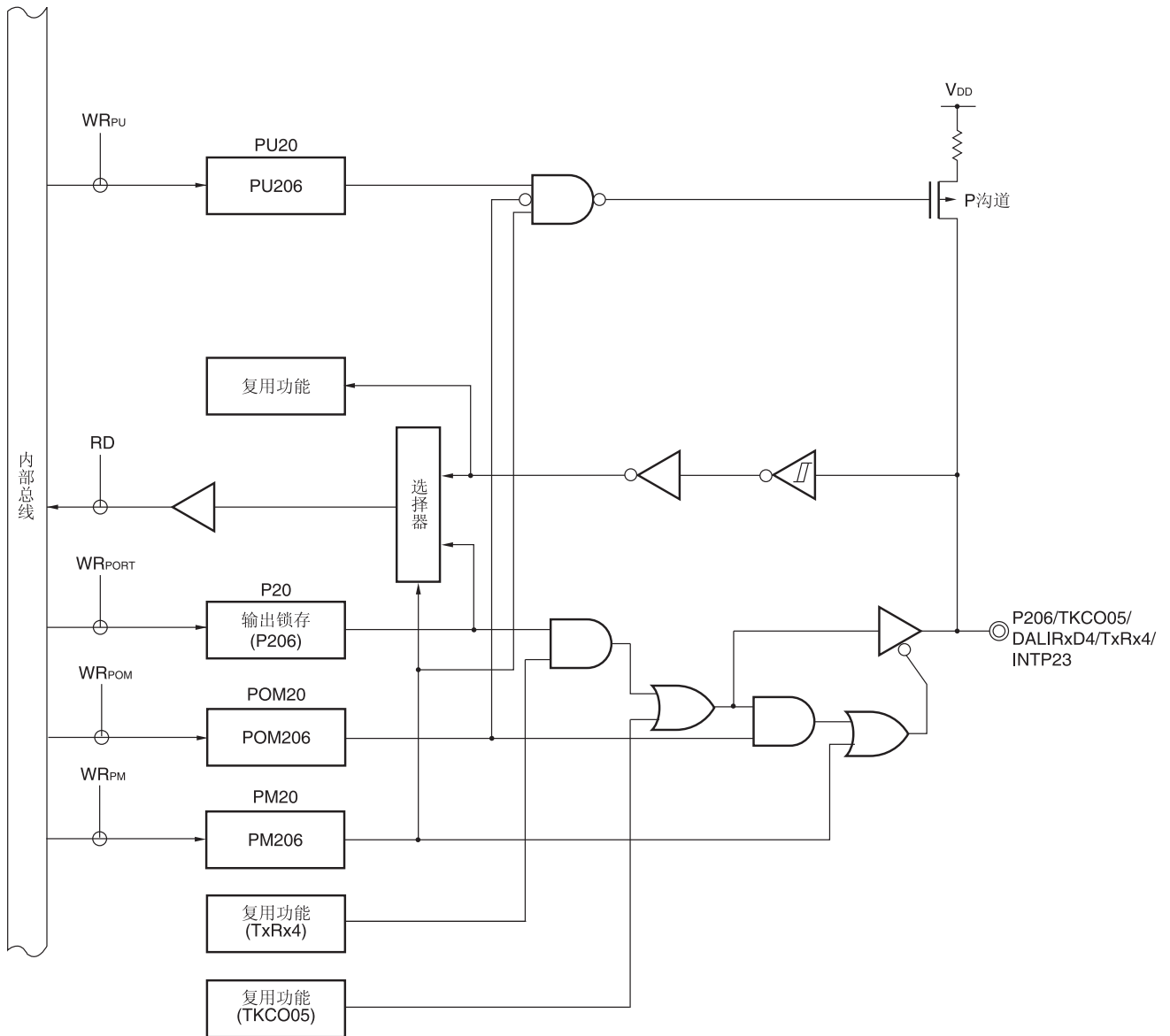
- P20: 端口寄存器 20
- PU20: 上拉电阻选择寄存器 20
- PM20: 端口模式寄存器 20
- POM20: 端口输出模式寄存器 20
- RD: 读取信号
- WR_{xx}: 写入信号

图 4-21. P205 的框图



- P20: 端口寄存器 20
- PU20: 上拉电阻选择寄存器 20
- PM20: 端口模式寄存器 20
- POM20: 端口输出模式寄存器 20
- RD: 读取信号
- WR_{xx}: 写入信号

图 4-22. P206 的框图



- P20: 端口寄存器 20
- PU20: 上拉电阻选择寄存器 20
- PM20: 端口模式寄存器 20
- POM20: 端口输出模式寄存器 20
- RD: 读取信号
- WR_{xx}: 写入信号

4.3 控制端口功能的寄存器

利用以下寄存器控制端口。

- 端口模式寄存器(PMxx)
- 端口寄存器(Pxx)
- 上拉电阻选择寄存器(PUxx)
- 端口输入模式寄存器(PIMxx)
- 端口输出模式寄存器(POMxx)
- 端口模式控制寄存器(PMCxx)
- A/D 端口配置寄存器(ADPC)
- 外围输入/输出重定向寄存器(PIOR1)

注意事项 各寄存器的不定位因产品而异，须在初始值状态下使用。

表 4-13. 配置在各个产品上的 PMxx、Pxx、PUxx、PIMxx、POMxx、PMCxx 寄存器及其对应的位 (1/3)

端口		位名称						20 引脚	30 引脚	32 引脚	38 引脚
		PMxx 寄存器	Pxx 寄存器	PUxx 寄存器	PIMxx 寄存器	POMxx 寄存器	PMCxx 寄存器				
端口 0	0	-	-	-	-	-	-	-	-	-	-
	1	-	-	-	-	-	-	-	-	-	-
	2	PM02	P02	PU02	-	POM02	PMC02	-	√	-	√
	3	PM03	P03	PU03	PIM03	-	PMC03	-	√	-	√
	4	-	-	-	-	-	-	-	-	-	-
	5	PM05	P05	PU05	-	-	-	-	-	√	√
	6	PM06	P06	PU06	-	-	-	-	-	√	√
	7	-	-	-	-	-	-	-	-	-	-
端口 1	0	PM10	P10	PU10	PIM10	POM10	-	√	√	√	√
	1	PM11	P11	PU11	PIM11	POM11	-	√	√	√	√
	2	PM12	P12	PU12	-	POM12	-	-	-	-	√
	3	-	-	-	-	-	-	-	-	-	-
	4	-	-	-	-	-	-	-	-	-	-
	5	-	-	-	-	-	-	-	-	-	-
	6	-	-	-	-	-	-	-	-	-	-
	7	-	-	-	-	-	-	-	-	-	-
端口 2	0	PM20	P20	-	-	-	-	√	√	√	√
	1	PM21	P21	-	-	-	-	√	√	√	√
	2	PM22	P22	-	-	-	-	√	√	√	√
	3	-	-	-	-	-	-	-	-	-	-
	4	PM24	P24	-	-	-	-	√	√	√	√
	5	PM25	P25	-	-	-	-	√	√	√	√
	6	PM26	P26	-	-	-	-	-	√	√	√
	7	PM27	P27	-	-	-	-	-	√	√	√

表 4-13. 配置在各个产品上的 PMxx、Pxx、PUxx、PIMxx、POMxx、PMCx 寄存器及其对应的位 (2/3)

端口		位名称						20 引脚	30 引脚	32 引脚	38 引脚
		PMxx 寄存器	Pxx 寄存器	PUxx 寄存器	PIMxx 寄存器	POMxx 寄存器	PMCxx 寄存器				
端口 3	0	PM30	P30	PU30	-	-	-	-	-	-	√
	1	PM31	P31	PU31	-	-	-	-	√	√	√
	2	-	-	-	-	-	-	-	-	-	-
	3	-	-	-	-	-	-	-	-	-	-
	4	-	-	-	-	-	-	-	-	-	-
	5	-	-	-	-	-	-	-	-	-	-
	6	-	-	-	-	-	-	-	-	-	-
	7	-	-	-	-	-	-	-	-	-	-
端口 4	0	PM40	P40	PU40	-	-	-	√	√	√	√
	1	-	-	-	-	-	-	-	-	-	-
	2	-	-	-	-	-	-	-	-	-	-
	3	-	-	-	-	-	-	-	-	-	-
	4	-	-	-	-	-	-	-	-	-	-
	5	-	-	-	-	-	-	-	-	-	-
	6	-	-	-	-	-	-	-	-	-	-
	7	-	-	-	-	-	-	-	-	-	-
端口 7	0	-	-	-	-	-	-	-	-	-	-
	1	-	-	-	-	-	-	-	-	-	-
	2	-	-	-	-	-	-	-	-	-	-
	3	-	-	-	-	-	-	-	-	-	-
	4	-	-	-	-	-	-	-	-	-	-
	5	PM75	P75	PU75	-	-	-	-	-	√	√
	6	PM76	P76	PU76	-	-	-	-	-	√	√
	7	PM77	P77	PU77	-	-	-	-	√	√	√
端口 12	0	PM120	P120	PU120	-	-	PMC120	-	√	√	√
	1	-	P121	-	-	-	-	√	√	√	√
	2	-	P122	-	-	-	-	√	√	√	√
	3	-	P123	-	-	-	-	-	-	-	√
	4	-	P124	-	-	-	-	-	-	-	√
	5	-	-	-	-	-	-	-	-	-	-
	6	-	-	-	-	-	-	-	-	-	-
	7	-	-	-	-	-	-	-	-	-	-
端口 13	0	-	-	-	-	-	-	-	-	-	-
	1	-	-	-	-	-	-	-	-	-	-
	2	-	-	-	-	-	-	-	-	-	-
	3	-	-	-	-	-	-	-	-	-	-
	4	-	-	-	-	-	-	-	-	-	-
	5	-	-	-	-	-	-	-	-	-	-
	6	-	-	-	-	-	-	-	-	-	-
	7	-	P137	-	-	-	-	√	√	√	√

表 4-13. 配置在各个产品上的 PMxx、Pxx、PUxx、PIMxx、POMxx、PMCx 寄存器及其对应的位 (3/3)

端口		位名称						20 引脚	30 引脚	32 引脚	38 引脚
		PMxx 寄存器	Pxx 寄存器	PUxx 寄存器	PIMxx 寄存器	POMxx 寄存器	PMCxx 寄存器				
端口 14	0	-	-	-	-	-	-	-	-	-	-
	1	-	-	-	-	-	-	-	-	-	-
	2	-	-	-	-	-	-	-	-	-	-
	3	-	-	-	-	-	-	-	-	-	-
	4	-	-	-	-	-	-	-	-	-	-
	5	-	-	-	-	-	-	-	-	-	-
	6	-	-	-	-	-	-	-	-	-	-
	7	PM147	P147	PU147	-	-	PMC147	√	√	√	√
端口 20	0	PM200	P200	PU200	-	POM200	-	√	√	√	√
	1	PM201	P201	PU201	-	POM201	-	√	√	√	√
	2	PM202	P202	PU202	-	POM202	-	√	√	√	√
	3	PM203	P203	PU203	-	POM203	-	√	√	√	√
	4	PM204	P204	PU204	-	POM204	-	-	√	√	√
	5	PM205	P205	PU205	-	POM205	-	-	√	√	√
	6	PM206	P206	PU206	-	POM206	-	-	√	√	√
	7	-	-	-	-	-	-	-	-	-	-

各寄存器的格式如下所述。在此以 38 引脚产品为例进行描述。

关于 38 引脚以外产品上所配置的寄存器，请参阅表 4-13。

(1) 端口模式寄存器(PMxx)

该寄存器以 1 位为单位指定端口的输入模式或输出模式。

使用 1 位或 8 位存储器操作指令来设置该寄存器。

产生复位信号后，该寄存器被设置为 FFH。

当端口引脚用作复用功能的引脚时，端口模式寄存器的设置请参阅 4.5 使用复用功能时端口模式寄存器和输出锁存器的设置。

图 4-23. 端口模式寄存器的格式 (38 引脚产品)

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PM0	1	PM06	PM05	1	PM03	PM02	1	1	FFF20H	FFH	R/W
PM1	1	1	1	1	1	PM12	PM11	PM10	FFF21H	FFH	R/W
PM2	PM27	PM26	PM25	PM24	1	PM22	PM21	PM20	FFF22H	FFH	R/W
PM3	1	1	1	1	1	1	PM31	PM30	FFF23H	FFH	R/W
PM4	1	1	1	1	1	1	1	PM40	FFF24H	FFH	R/W
PM7	PM77	PM76	PM75	1	1	1	1	1	FFF27H	FFH	R/W
PM12	1	1	1	1	1	1	1	PM120	FFF2CH	FFH	R/W
PM14	PM147	1	1	1	1	1	1	1	FFF2EH	FFH	R/W
PM20	1	PM206	PM205	PM204	PM203	PM202	PM201	PM200	F0510H	FFH	R/W
PMmn	Pmn 引脚输入/输出模式选择 (m = 0 至 4, 7, 12, 14, 20; n = 0 至 7)										
0	输出模式 (输出缓冲器启用)										
1	输入模式 (输出缓冲器关闭)										

注意事项 必须将 PM0 寄存器的位 0, 1, 4, 7、PM1 寄存器的位 3 至 7、PM2 寄存器的位 3、PM3 寄存器的位 2 至 7、PM4 寄存器的位 1 至 7、PM7 寄存器的位 0 至 4、PM12 寄存器的位 1 至 7、PM14 寄存器的位 0 至 6、和 PM20 寄存器的位 7 设置为 1。

对于 32, 30 和 20 引脚产品，在解除复位后，必须通过软件将以下的位设置为输出模式 (通过设置端口寄存器和端口模式寄存器为 0 来指定)。

32 引脚产品: PM0 寄存器的位 2 和 3、PM1 寄存器的位 2 和 PM3 寄存器的位 0

30 引脚产品: PM0 寄存器的位 5 和 6、PM1 寄存器的位 2、PM3 寄存器的位 0 和 PM7 寄存器的位 5 和 6

20 引脚产品: PM0 寄存器的位 2, 3, 5 和 6、PM1 寄存器的位 2、PM2 寄存器的位 6 和 7、PM3 寄存器的位 0 和 1、PM7 寄存器的位 5 至 7、PM12 寄存器的位 0 和 PM20 寄存器的位 4 至 6

(2) 端口寄存器 (Pxx)

该寄存器设置端口的输出锁存器的值。

如果输入模式时读取数据，将得到引脚电平。如果输出模式时读取，将会得到输出锁存器的值^注。

使用 1 位或 8 位存储器操作指令来设置该寄存器。

产生复位信号后，该寄存器被清除为 00H。

注 P02, P03, P20 至 P22, P24 至 P27, P120 和 P147 被设为 A/D 转换器的模拟输入时，在输入模式中读取端口时，将始终返回 0 而非引脚电平。

图 4-24. 端口寄存器的格式 (38 引脚产品)

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
P0	0	P06	P05	0	P03	P02	0	0	FFF00H	00H (输出锁存器)	R/W
P1	0	0	0	0	0	P12	P11	P10	FFF01H	00H (输出锁存器)	R/W
P2	P27	P26	P25	P24	0	P22	P21	P20	FFF02H	00H (输出锁存器)	R/W
P3	0	0	0	0	0	0	P31	P30	FFF03H	00H (输出锁存器)	R/W
P4	0	0	0	0	0	0	0	P40	FFF04H	00H (输出锁存器)	R/W
P7	P77	P76	P75	0	0	0	0	0	FFF07H	00H (输出锁存器)	R/W
P12	0	0	0	P124	P123	P122	P121	P120	FFF0CH	不定	R/W ^{※1}
P13	P137	0	0	0	0	0	0	0	FFF0DH	注 2	R
P14	P147	0	0	0	0	0	0	0	FFF0EH	00H (输出锁存器)	R/W
P20	0	P206	P205	P204	P203	P202	P201	P200	F0500H	00H (输出锁存器)	R/W

Pmn	m = 0 至 4, 7, 12 至 14, 20; n = 0 至 7	
	输出数据控制 (输出模式时)	输入数据读取 (输入模式时)
0	输出 0	输入低电平
1	输出 1	输入高电平

注 1. P121 至 P124 和 P137 为只读。

2. P137: 不定

(3) 上拉电阻选择寄存器(PUxx)

该寄存器指定是否使用片上上拉电阻。对于已在上拉电阻选择寄存器中指定使用片上上拉电阻的引脚，只能在设为输入模式的位（ $PMmn = 1$ 且 $POMmn = 0$ ）上以位单位使用片上上拉电阻。无论该寄存器的设置如何，片上上拉电阻不会连接至设为输出模式的位、作为复用功能的输出引脚使用时和模拟设置($PMC = 1, ADPC = 1$)时也是同样如此。

使用 1 位或 8 位存储器操作指令来设置该寄存器。

产生复位信号后，该寄存器被清除为 00H（仅设置 PU4 为 01H）。

注意事项 当带 PIMn 寄存器的端口从不同的电位器件输入到 TTL 缓冲器，通过设置 $PUmn = 0$ 使用外部上拉电阻上拉到不同电位器件的电源处。

图 4-25. 上拉电阻选择寄存器的格式 (38 引脚产品)

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PU0	0	PU06	PU05	0	PU03	PU02	0	0	F0030H	00H	R/W
PU1	0	0	0	0	0	PU12	PU11	PU10	F0031H	00H	R/W
PU3	0	0	0	0	0	0	PU31	PU30	F0033H	00H	R/W
PU4	0	0	0	0	0	0	0	PU40	F0034H	01H	R/W
PU7	PU77	PU76	PU75	0	0	0	0	0	F0037H	00H	R/W
PU12	0	0	0	0	0	0	0	PU120	F003CH	00H	R/W
PU14	PU147	0	0	0	0	0	0	0	F003EH	00H	R/W
PU20	0	PU206	PU205	PU204	PU203	PU202	PU201	PU200	F0520H	00H	R/W

PUmn	选择 Pmn 引脚片上上拉电阻 (m = 0, 1, 3, 4, 7, 12, 14, 20; n = 0 至 7)
0	不连接片上上拉电阻
1	连接片上上拉电阻

(4) 端口输入模式寄存器 (PIMxx)

该寄存器以 1 位为单位设置输入缓冲器。

可以在与电平不同的外部器件进行串行通信期间选择 TTL 输入缓冲器。

使用 1 位或 8 位存储器操作指令来设置该寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 4-26. 端口输入模式寄存器的格式 (38 引脚产品)

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PIM0	0	0	0	0	PIM03	0	0	0	F0040H	00H	R/W
PIM1	0	0	0	0	0	0	PIM11	PIM10	F0041H	00H	R/W
PIMmn	Pmn 引脚输入缓冲器的选择 (m = 0, 1; n = 0, 1, 3)										
0	普通输入缓冲器										
1	TTL 输入缓冲器										

(5) 端口输出模式寄存器 (POMxx)

该寄存器以 1 位为单位设置输出模式。

可以在与电位不同的外部器件进行串行通信期间以及在与电位相同的外部器件进行简易 IICA 通信时，为 SDAA0 引脚选择 N 沟开漏输出 (V_{DD} 耐压) 模式。

另外，通过设置 POMxx 寄存器以及 PUxx 寄存器来选择是否使用片上上拉电阻。

使用 1 位或 8 位存储器操作指令来设置该寄存器。

产生复位信号后，清除该寄存器为 00H。

图 4-27. 端口输出模式寄存器的格式(38 引脚产品)

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
POM0	0	0	0	0	0	POM02	0	0	F0050H	00H	R/W
POM1	0	0	0	0	0	POM12	POM11	POM10	F0051H	00H	R/W
POM20	0	POM206	POM205	POM204	POM203	POM202	POM201	POM200	F0530H	00H	R/W
POMmn	Pmn 引脚输出模式的选择 (m = 0, 1, 20; n = 0 至 6)										
0	普通输出模式 输入模式时，启用 PUm _n 位。										
1	N 沟开漏输出 (V_{DD} 耐压) 模式 输入模式时，禁用 PUm _n 位。										

(6) 端口模式控制寄存器 (PMCxx)

该寄存器以 1 位为单位设置数字输入/输出或模拟输入。

使用 1 位或 8 位存储器操作指令来设置该寄存器。

产生复位信号后，该寄存器被设置为 FFH。

图 4-28. 端口模式控制寄存器的格式(38 引脚产品)

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PMC0	1	1	1	1	PMC03 注1	PMC02 注2	1	1	F0060H	FFH	R/W
PMC12	1	1	1	1	1	1	1	PMC120 注2	F006CH	FFH	R/W
PMC14	PMC147 注3	1	1	1	1	1	1	1	F006EH	FFH	R/W
PMCmn	Pmn 引脚数字输入/输出/模拟输入的选择 (m = 0, 12, 14; n = 0, 2, 3, 7)										
0	数字输入/输出(模拟输入以外的复用功能)										
1	模拟输入										

注 1. 仅限 30、38 引脚产品

2. 仅限 30、32、38 引脚产品

3. 所有产品

注意事项 1. 通过端口模式寄存器 0, 12, 14 (PM0, PM12, PM14)将用于 A/D 转换的通道设置为输入模式。

2. 对于由 PMC 寄存器设置为数字输入/输出的引脚，不要使用模拟输入通道选择寄存器(ADS)对其进行设置。

(7) A/D 端口配置寄存器 (ADPC)

该寄存器将 ANI0/P20, ANI1/P21, ANI2/CMP0P/P22, ANI4/CMP1P/P24 至 ANI7/CMP4P/P27 引脚和 PGAOUT 引脚 (内部引脚) 在端口的数字输入/输出以及 A/D 转换器、可编程增益放大器或比较器的模拟输入之间进行相互切换。

使用 8 位存储器操作指令来设置 ADPC 寄存器。

产生复位信号后, 该寄存器被设置为 00H。

图 4-29. A/D 端口配置寄存器(ADPC)的格式

地址: F0076H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ADPC	0	0	0	0	ADPC3	ADPC2	ADPC1	ADPC0

ADPC3	ADPC2	ADPC1	ADPC0	模拟输入(A)/数字 I/O(D)切换							
				ANI7/ CMP4P/P27	ANI6/ CMP3P/P26	ANI5/ CMP2P/P25	ANI4/ CMP1P/P24	PGAOUT ^注	ANI2/ CMP0P/P22	ANI1/P21	ANI0/P20
0	0	0	0	A	A	A	A	A	A	A	A
0	0	0	1	D	D	D	D	D	D	D	D
0	0	1	0	D	D	D	D	D	D	D	A
0	0	1	1	D	D	D	D	D	D	A	A
0	1	0	0	D	D	D	D	D	A	A	A
0	1	0	1	D	D	D	D	A	A	A	A
0	1	1	0	D	D	D	A	A	A	A	A
0	1	1	1	D	D	A	A	A	A	A	A
1	0	0	0	D	A	A	A	A	A	A	A
1	0	0	1	A	A	A	A	A	A	A	A
1	1	1	1	A	A	A	A	A	A	A	A
其它				禁止设置							

注 这是一个可编程增益放大器的内部输出引脚。将可编程增益放大器的输出信号用作 A/D 转换器的模拟输入通道时, 将 ADPC 设置为等于或大于 0101B。

- 注意事项
1. 通过端口模式寄存器 2 (PM2) 将由 ADPC 寄存器设置为模拟输入的端口设置为输入模式。
 2. 对于由 ADPC 寄存器设置为数字输入/输出的引脚, 不要使用模拟输入通道选择寄存器(ADS)对其进行设置。

(8) 外围输入/输出重定向寄存器 (PIOR1)

该寄存器用于指定是否允许或禁止外围输入/输出重定向功能。

该功能用于切换被分配了复用功能的端口。

使用 PIOR1 寄存器给端口分配重定向功能后，允许操作此功能。

并且，对重定向设置进行改变时要在允许功能操作之前进行。

使用 8 位存储器操作指令来设置 PIOR 寄存器。

产生复位信号后，该寄存器被设置为 00H。

图 4-30. 外围输入/输出重定向寄存器(PIOR1)的格式

地址: F05C0H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
PIOR1	0	0	0	0	PIOR13	PIOR12	PIOR11	PIOR10

位	功能	20引脚		30引脚		32引脚		38引脚	
		设置值		设置值		设置值		设置值	
		0	1	0	1	0	1	0	1
PIOR10	TKCO03	禁止设置		P204	-	P204	-	P204	P12
PIOR11	DALITxD4/ DALIRxD4	-	P10/P11	P205/ P206	P10/P11	P205/ P206	P10/P11	P205/ P206	P10/P11
PIOR12	INTP20	P10	P203	P10	P203	P10	P203	P10	P203
PIOR13	INTP21	P11	P202	P11	P202	P11	P202	P11	P202

4.4 端口功能的操作

如下所示，端口的操作因输入/输出模式设置的不同而异。

4.4.1 输入/输出端口的写入

(1) 输出模式

通过传送指令向输出锁存器写入值，输出锁存器的内容则从引脚输出。

写入输出锁存器的数据被一直保留，直到再次有数据被写入输出锁存器。

产生复位信号后，输出锁存器的数据被清除。

(2) 输入模式

通过传送指令向输出锁存器写入值，但由于输出缓冲器关闭，引脚状态不发生变化。因此，对于既包含输入又包含输出的端口，可以使用字节写入。

写入输出锁存器的数据被一直保留，直到再次有数据被写入输出锁存器。

产生复位信号后，输出锁存器的数据被清除。

4.4.2 输入/输出端口的读取

(1) 输出模式

通过传送指令读取输出锁存器的内容。输出锁存器的内容不发生变化。

(2) 输入模式

通过传送指令读取引脚状态。输出锁存器的内容不发生变化。

4.4.3 操作输入/输出端口

(1) 输出模式

对输出锁存器内容进行运算，结果写入输出锁存器。输出锁存器的内容从引脚输出。

写入输出锁存器的数据被一直保留，直到再次有数据被写入输出锁存器。

产生复位信号后，输出锁存器的数据被清除。

(2) 输入模式

读取引脚电平，并对其内容进行运算。操作结果写入输出锁存器，但由于输出缓冲器关闭，引脚状态不发生变化。因此，对于既包含输入又包含输出的端口，可以使用字节写入。

产生复位信号后，输出锁存器的数据被清除。

4.4.4 与不同电位的外部器件(2.5 V、3 V)的连接方法

根据连接器件的电源更改 V_{DD} ，可与以不同电位（2.5 V 或 3 V）的外部器件连接。在 2.5 V 或 3 V 工作下的外部器件可通过串行接口和通用端口使用端口 0 和 1 进行输入/输出的连接。

外部器件	V_{DD}
3 V	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$
2.5 V	$3.3\text{ V} \leq V_{DD} \leq 4.0\text{ V}$

关于输入，可通过端口输入模式寄存器(PIM0, PIM1)按位进行普通(CMOS)/TTL 输入缓冲器的切换。

另外，关于输出，通过利用端口输出模式寄存器(POM0, POM1)将输出缓冲器切换至 N 沟开漏 (V_{DD} 耐压)，可以实现对不同电位的支持。

下面对串行接口的连接进行说明。

(1) 使用 UART0、UART1 和 CSI00 功能的输入/输出引脚时的设置步骤

(a) 作为 2.5 V、3 V 的输入端口使用时

<1> 如需上拉，从外部将使用的引脚上拉至目标器件的电源处（不能使用片上上拉电阻）。

UART0 的情况下： P11

UART1 的情况下： P03

CSI00 的情况下： P11, P12

<2> 解除复位后，端口模式为输入模式(Hi-Z)。

<3> 将 PIM0 和 PIM1 寄存器的对应位设置为 1，从而切换至 TTL 输入缓冲器。

<4> V_{IH}/V_{IL} 以 2.5 V、3 V 的工作电压工作。

(b) 作为 2.5 V、3 V 的输出端口使用时

<1> 从外部将使用的引脚上拉至目标器件的电源处（不能使用片上上拉电阻）。

UART0 的情况下： P10

UART1 的情况下： P02

CSI00 的情况下： P10, P12

<2> 解除复位后，端口模式改变为输入模式(Hi-Z)。

<3> 将对应端口的输出锁存器设置为 1。

<4> 将 POM0 和 POM1 寄存器的对应位设置为 1，从而设置 N 沟开漏 (V_{DD} 耐压) 模式。

<5> 操作 PM0 和 PM1 寄存器设置输出模式。

此时，输出数据为高电平，因而引脚处于 Hi-Z 状态。

<6> 可通过设置串行阵列单元进行通信。

(2) 使用 IICA0 功能的输入/输出引脚时的设置步骤

<1> 从外部上拉使用的 P10 和 P11 引脚（不能使用片上上拉电阻）。

<2> 解除复位后，端口模式为输入模式(Hi-Z)。

<3> 将对应端口的输出锁存器设置为 1。

<4> 将 POM1 寄存器的对应位设置为 1，从而设置 N 沟开漏 (V_{DD} 耐压) 模式。

<5> 将 PM1 寄存器的对应位设置为输出模式（输出模式下可实现数据输入/输出）。

此时，输出数据为高电平，因而引脚处于 Hi-Z 状态。

<6> 允许操作 IICA0。

4.5 使用复用功能时端口模式寄存器和输出锁存器的设置

使用端口引脚的复用功能时，请参阅表 4-14 所列内容设置端口模式寄存器和输出锁存器。

注意事项 如果复用输出功能指定至一个已用于输出的引脚，需要把不使用的复用功能的输出设置为与其初始状态相同的值。关于可应用的单元和引脚的处理方法，请参阅 4.6.2。

表 4-14. 使用复用功能时端口相关寄存器的设置 (1/2)

引脚名称	复用功能		PIOR _{xx}	POM _{xx}	PMC _{xx}	PM _{xx}	P _{xx}
	功能名称	输入/输出					
P02	ANI17 ^{※1}	输入	×	×	1	1	×
	TxD1	输出	×	0/1	0	0	1
P03	ANI16 ^{※1}	输入	×	–	1	1	×
	CMP5P ^{※1}	输入	×	–	1	1	×
	RxD1	输入	×	–	0	1	×
P05	TI05	输入	×	–	–	1	×
	TO05	输出	×	–	–	0	0
P06	TI06	输入	×	–	–	1	×
	TO06	输出	×	–	–	0	0
P10	SO00	输出	×	0/1	–	0	1
	TxD0	输出	×	0/1	–	0	1
	TKCO00	输出	×	0	–	0	0
	INTP20	输入	0	×	–	1	×
	SCLA0	输入/输出	×	1	–	0	0
	(DALITxD4)	输出	1	0/1	–	0	1
P11	SI00	输入	×	×	–	1	×
	RxD0	输入	×	×	–	1	×
	TKCO01	输出	×	0	–	0	0
	INTP21	输入	0	×	–	1	×
	SDAA0	输入/输出	×	1	–	0	0
	(TI07)	输入	×	×	–	1	×
	(DALIRxD4)	输入	1	×	–	1	×
	(TxRx4)	输入/输出	×	0/1	–	0	1
P12	SCK00	输入	×	×	–	1	×
		输出	×	0/1	–	0	1
	(TKCO03)	输出	1	0	–	0	0

备注 1. ×: 忽略

PIOR_x: 外围输入/输出重定向寄存器

POM_{xx}: 端口输出模式寄存器

PMC_{xx}: 端口模式控制寄存器

PM_{xx}: 端口模式寄存器

P_{xx}: 端口输出锁存器

- 表中所示引脚与其复用功能之间的关系均为使用 38 引脚产品时的情况。对于其他产品，复用功能可能分配至与此不同的引脚，但是 PIOR_x、POM_{xx}、PMC_{xx}、PM_{xx} 和 P_{xx} 的设置相同。
- 可以通过设置外围输入/输出重定向寄存器(PIOR1)或输入切换控制寄存器(ISC)来指定上图括号内的功能。

(注如最末尾的表后所示。)

表 4-14. 使用复用功能时端口相关寄存器的设置 (2/2)

引脚名称	复用功能		PIOR _{xx}	POM _{xx}	PMC _{xx}	PM _{xx}	P _{xx}
	功能名称	输入/输出					
P20	ANI0	输入	×	-	-	1	×
	AV _{REFP}	输入	×	-	-	1	×
P21	ANI1	输入	×	-	-	1	×
	AV _{REFM}	输入	×	-	-	1	×
P22, P24 至 P27 ^{※2}	ANI2, ANI4 至 ANI7 ^{※2}	输入	×	-	-	1	×
	CMP0P 至 CMP4P	输入	×	-	-	1	×
P30	INTP3	输入	×	-	-	1	×
	RTC1HZ	输出	×	-	-	0	0
P31	TI03	输入	×	-	-	1	×
	TO03	输出	×	-	-	0	0
	INTP4	输入	×	-	-	1	×
P40	TOOL0	输入/输出	×	-	-	×	×
P75 至 P77	INTP9 至 INTP11	输入	×	-	-	1	×
P120	ANI19 ^{※1}	输入	×	-	1	1	×
P137	INTP0	输入	×	-	-	1	×
P147	ANI18 ^{※1}	输入	×	-	1	1	×
	CMPCOM	输入	×	-	1	1	×
P200	TKBO00	输出	×	0	-	0	0
	INTP22	输入	×	×	-	1	×
P201	TKBO01	输出	×	0	-	0	0
P202	TKBO10	输出	×	0	-	0	0
	(INTP21)	输入	1	×	-	1	×
P203	TKBO11	输出	×	0	-	0	0
	TKCO02	输出	×	0	-	0	0
	(INTP20)	输入	1	×	-	1	×
P204	TKBO20	输出	×	0	-	0	0
	TKCO03	输出	0	0	-	0	0
P205	TKBO21	输出	×	0	-	0	0
	TKCO04	输出	×	0	-	0	0
	DALITxD4	输出	0	0/1	-	0	1
P206	TKCO05	输出	×	0	-	0	0
	DALIRxD4	输入	0	×	-	1	×
	TxRx4	输入/输出	×	0/1	-	0	1
	INTP23	输入	×	×	-	1	×

备注 1. ×: 忽略

PIOR_x: 外围输入/输出重定向寄存器

POM_{xx}: 端口输出模式寄存器

PMC_{xx}: 端口模式控制寄存器

PM_{xx}: 端口模式寄存器

P_{xx}: 端口输出锁存器

2. 表中所示引脚与其复用功能之间的关系均为使用 38 引脚产品时的情况。对于其他产品，复用功能可能分配至与此不同的引脚，但是 PIOR_x、POM_{xx}、PMC_{xx}、PM_{xx} 和 P_{xx} 的设置相同。

3. 可以通过设置外围输入/输出重定向寄存器(PIOR1)或输入切换控制寄存器(ISC)来指定上图括号内的功能。

(注如下页所示。)

注 1. ANI16/CMP5P/P03, ANI17/P02, ANI18/CMPCOM/P147 和 ANI19/P120 引脚的功能可通过端口模式控制寄存器 0, 12, 14 (PMC0, PMC12, PMC14)、模拟输入通道选择寄存器(ADS)和端口模式寄存器 0, 12, 14 (PM0, PM12, PM14)选择。

表 4-15. ANI16/CMP5P/P03、ANI17/P02、ANI18/CMPCOM/P147 和 ANI19/P120 引脚功能的设置

PMC0, PMC12, PMC14 寄存器	PM0, PM12, PM14 寄存器	ADS寄存器	ANI16/CMP5P/P03, ANI17/P02, ANI18/CMPCOM/P147, and ANI19/P120引脚
数字输入/输出	输入模式	×	数字输入
	输出模式	×	数字输出
模拟输入	输入模式	选择ANI	模拟输入 (被转换)
		不选择ANI	模拟输入 (不被转换)
	输出模式	选择ANI	禁止设置
		不选择ANI	

2. ANI0/P20、ANI1/P21、ANI2/CMP0P/P22 和 ANI4/CMP1P/P24 至 ANI7/CMP4P/P27 引脚的功能可通过 A/D 端口配置寄存器(ADPC)、模拟输入通道选择寄存器(ADS)和端口模式寄存器 2 (PM2)选择。

表 4-16. ANI0/P20、ANI1/P21、ANI2/CMP0P/P22 和 ANI4/CMP1P/P24 至 ANI7/CMP4P/P27 引脚功能的设置

ADPC寄存器	PM2寄存器	ADS寄存器	ANI0/P20, ANI1/P21, ANI2/CMP0P/P22, and ANI4/CMP1P/P24至 ANI7/CMP4P/P27引脚
数字输入/输出	输入模式	×	数字输入
	输出模式	×	数字输出
模拟输入	输入模式	选择ANI	模拟输入 (被转换)
		不选择ANI	模拟输入 (不被转换)
	输出模式	选择ANI	禁止设置
		不选择ANI	

4.6 使用端口功能时的注意事项

4.6.1 端口寄存器n (Pn)的位操作指令的相关注意事项

对于同时具有输入和输出功能的端口执行位操作指令时，除了操作目标位以外，非操作对象的输入端口的输出锁存器值也可能被写入。

因此，将端口由输入模式切换为输出模式前，建议重写输出锁存器。

<示例> 当 P200 为输出端口、P201 至 P206 为输入端口（所有引脚状态均为高电平）且端口 20 的输出锁存器的值为 00H 时，通过位操作指令将输出端口 P200 的输出从低电平改为高电平，则端口 20 的输出锁存器的值变为 7FH。

说明：PMnm 位为 1 的端口，其写入或读取 Pn 寄存器的对象分别是输出锁存器和引脚状态。
在 RL78/I1A 中，位操作指令按以下顺序执行。

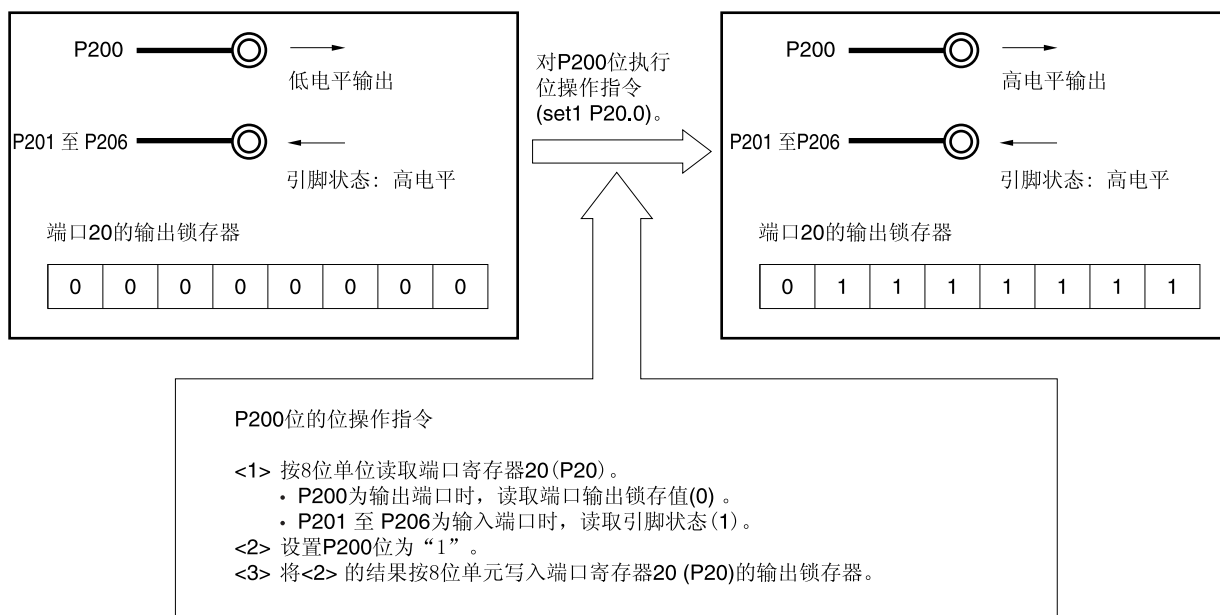
- <1> 以 8 位为单位读取 Pn 寄存器。
- <2> 操作目标位。
- <3> 以 8 位为单位写入 Pn 寄存器。

在步骤<1>中，输出端口 P200 的输出锁存器值 0 被读取，同时，输入端口 P201 至 P206 的引脚状态也被读取。如果 P201 至 P206 的引脚状态此时为高电平，则读取值为 7EH。

步骤<2>的操作使该值变为 7FH。

步骤<3>的操作将 7FH 写入输出锁存器。

图 4-31. 位操作指令(P200)



4.6.2 引脚设置时的注意事项

如果复用输出的功能被设定至一个已用于输出的引脚时，需要把不使用的复用功能的输出设置为与初始状态相同的值以防止输出冲突。该注意事项也适用于通过外围输入/输出重定向寄存器 (PIOR)指定的功能。有关复用输出，请参阅 **4.5 使用复用功能时端口模式寄存器和输出锁存器的设置**。

此外，输入引脚不需要特殊的设置，因为其复用输出功能被禁用(缓冲器输出为 Hi-Z)。

表 4-17. 不使用的复用功能的管理

对象单元	不使用的复用功能的输出或输入/输出引脚	不使用的复用功能的管理
定时器阵列单元	TOmn	将定时器输出寄存器m (TOm)的位m (TOmn)和定时器输出允许寄存器m (TOEm)的位n (TOEmn)设置为初始值0。
16位定时器KB0至KB2	TKBOnp	将16位定时器KB输出控制寄存器n1 (TKBIOCn1)的位0和1 (TKBTOEn0, TKBTOEn1) 设置为初始值0。
16位定时器KC0	TKCO0n	将16位定时器KC输出控制寄存器01 (TKCIOC01) 的位n (TKCTOE0n)和16位定时器KC输出引脚控制寄存器(TOETKC0) 的位n (TOETKC0n) 设置为初始值 (0)。
串行阵列单元	SCKmn, SOmn, SCLmn, SDAmn, TxDn	将串行通道允许状态寄存器m (SEm)的位n (SEmn)、串行输出寄存器 m (SOm)的位n (SOmn)和串行输出允许寄存器m (SOEm)的位n (SOEmn) 设置为初始值(SOmn为1, 其它为0)。
DALI/UART4	DALITxD4, TxRx4	将串行通道允许状态寄存器4 (SE4)的位0 (SE40)、串行输出寄存器4 (SO4)的位n (SO40)和串行输出允许寄存器4 (SOE4)的位0 (SOE40)设置为初始值(SO40为1, 其它为0)。
IICA	SCAA0, SDAA0	将IICCTL00寄存器的位7 (IICE0)设置为0，禁用IICA操作。

示例：20 引脚产品的 P10/TxD0/TKCO00/INTP20/SCLA0 引脚

(1) 当引脚用作 TxD0 输出时

- P10: 将端口模式寄存器 1 的 PM10 设置为 0 指定输出模式。
 TKCO00: 输出引脚，因此将 16 位定时器 KC0 的 TKCTOE0n 和 TOETKC0n 设置为 0。
 INTP20: 输入引脚，因此不适用此注意事项。
 SCLA0: 输出引脚，因此将 IICA 的 IICE0 设置为 0。

(2) 当引脚用作 TO01 输出时

- P10: 将端口模式寄存器 1 的 PM10 设置为 0 指定输出模式。
 TxD0: 输出引脚，因此将串行阵列单元 0 的 SE00、SO00 和 SOE0 分别设置为 0、1 和 0。
 INTP20: 输入引脚，因此不适用于此注意事项。
 SCLA0: 输出引脚，因此将 IICA 的 IICE0 设置为 0。

如同将用于 SO00 输出的 P10/SO00/TxD0/TKCO00/INTP20/SCLA0/(DALITxD4)引脚切换至 TxD0 一样，切换操作模式并不能使指定给同一串行通道引脚的复用功能有效，因此，本注意事项不适用于这类引脚。(如果指定了 CSI 功能 (MD001 = 0)，此引脚并不作为 UART 引脚来运行，因此 TxD0 输出为无效。)

对于那些只有输入或没有输入/输出的模块，建议停止其中未使用的功能，以降低功耗。

第五章 时钟发生电路

主系统时钟谐振器连接引脚、副系统时钟谐振器连接引脚、主系统时钟外部时钟输入引脚以及副系统时钟外部时钟输入引脚的有无因产品而异。

输出引脚	20, 30, 32 引脚	38 引脚
X1, X2 引脚	√	√
EXCLK 引脚	√	√
XT1, XT2 引脚	-	√
EXCLKS 引脚	-	√

注 20、30 和 32 引脚产品不具备副系统时钟。

5.1 时钟发生电路的功能

时钟发生电路产生用于 CPU 和外围硬件的时钟。

可以选择以下三种系统时钟和时钟振荡电路。

(1) 主系统时钟

可以通过设置 MCM0 位（系统时钟控制寄存器(CKC)的位 4）来选择高速系统时钟（X1 时钟或外部主系统时钟）或高速片上振荡器时钟作为主系统时钟。

<1> X1 振荡电路

该电路通过将谐振器连接至 X1 和 X2 来产生 $f_x = 1$ 至 20MHz 的振荡时钟。

可以通过执行 STOP 指令或设置 MSTOP 位(时钟操作状态控制寄存器(CSC)的位 7)来停止振荡。

<2> 高速片上振荡器

通过选项字节(000C2H)可从以下范围内选择振荡频率： $f_{IH} = 32, 24, 16, 12, 8, 4$ 或 1 MHz (typ.)。在解除复位后，CPU 一定利用此高速片上振荡器时钟开始工作。可以通过执行 STOP 指令或设置 HIOSTOP 位(CSC 寄存器的位 0)来停止振荡。

<3> 外部主系统时钟

还可以通过 EXCLK/X2/P122 引脚供应外部主系统时钟($f_{EX} = 1$ 至 20MHz)。可以通过执行 STOP 指令或设置 MSTOP 位来禁止外部主系统时钟输入。

<4> 使用 PLL (锁相环)的倍增功能

如果为高速系统时钟或高速片上振荡器时钟选择 4 MHz，则可以使用 PLL 模式。在此模式中，可以供应高速系统时钟或高速片上振荡器时钟的 16 倍时钟 (64 MHz)作为 16 位定时器 KB0 至 KB2、KC0，以及比较器和可编程增益放大器的计数器时钟，还可以供应高速系统时钟或高速片上振荡器时钟的 16 倍的 1/2 时钟(32 MHz)或者高速系统时钟或高速片上振荡器时钟的 16 倍的 1/4 时钟(16 MHz)至所有其他时钟。可以通过设置 PLLON 位(PLLCTL 寄存器的位 0)来停止振荡。

注意事项 选择 PLL 输出作为主系统时钟时，不能设置为 STOP 模式。请停止 PLL 功能(PLL 控制寄存器(PLLCTL)的 SELPLL = 0 → PLLON = 0)后，选择高速片上振荡时钟(f_{IH})或高速系统时钟(f_{MX})作为主系统时钟，再执行 STOP 指令。

(2) 副系统时钟**<1> XT1 时钟振荡电路**

该电路通过将 32.768 kHz 的谐振器连接至 XT1 和 XT2 来产生 $f_{XT} = 32.768 \text{ kHz}$ 的振荡时钟。可以通过设置 XTSTOP 位(时钟操作状态控制寄存器(CSC)的位 6)来停止振荡。

<2> 外部副系统时钟

可以通过 EXCLKS/XT2/P124 引脚供应外部副系统时钟($f_{ET} = 32.768 \text{ KHz}$)。通过设置 XTSTOP 位可以禁止外部副系统时钟输入。

(3) 低速片上振荡器时钟

该电路产生 $f_{IL} = 15 \text{ kHz}$ (TYP.)的振荡时钟。

低速片上振荡器时钟不能用作 CPU 时钟。

仅限以下外围硬件使用低速片上振荡器时钟工作。

- 看门狗定时器
- 实时时钟
- 12 位间隔定时器

当选项字节(000C0H)的位 4(WDTON)或操作速度模式控制寄存器(OSMC)的位 4(WUTMMCK0)或者两者同时设置为 1 时, 该时钟工作。

然而, 当 $WDTON = 1$ 、 $WUTMMCK0 = 0$ 且选项字节(000C0H)的位 0(WDSTBYON)为 0 时, 执行 HALT 或 STOP 指令后低速片上振荡器停止振荡。

注意事项 仅限在使用固定周期中断功能时, 才可以选择低速片上振荡器时钟(f_{IL})作为实时时钟的工作时钟。

备注	f_X : X1 时钟振荡频率
	f_{IH} : 高速片上振荡器时钟频率
	f_{EX} : 外部主系统时钟频率
	f_{XT} : XT1 时钟振荡频率
	f_{EXT} : 外部副系统时钟频率
	f_{IL} : 低速片上振荡器时钟频率

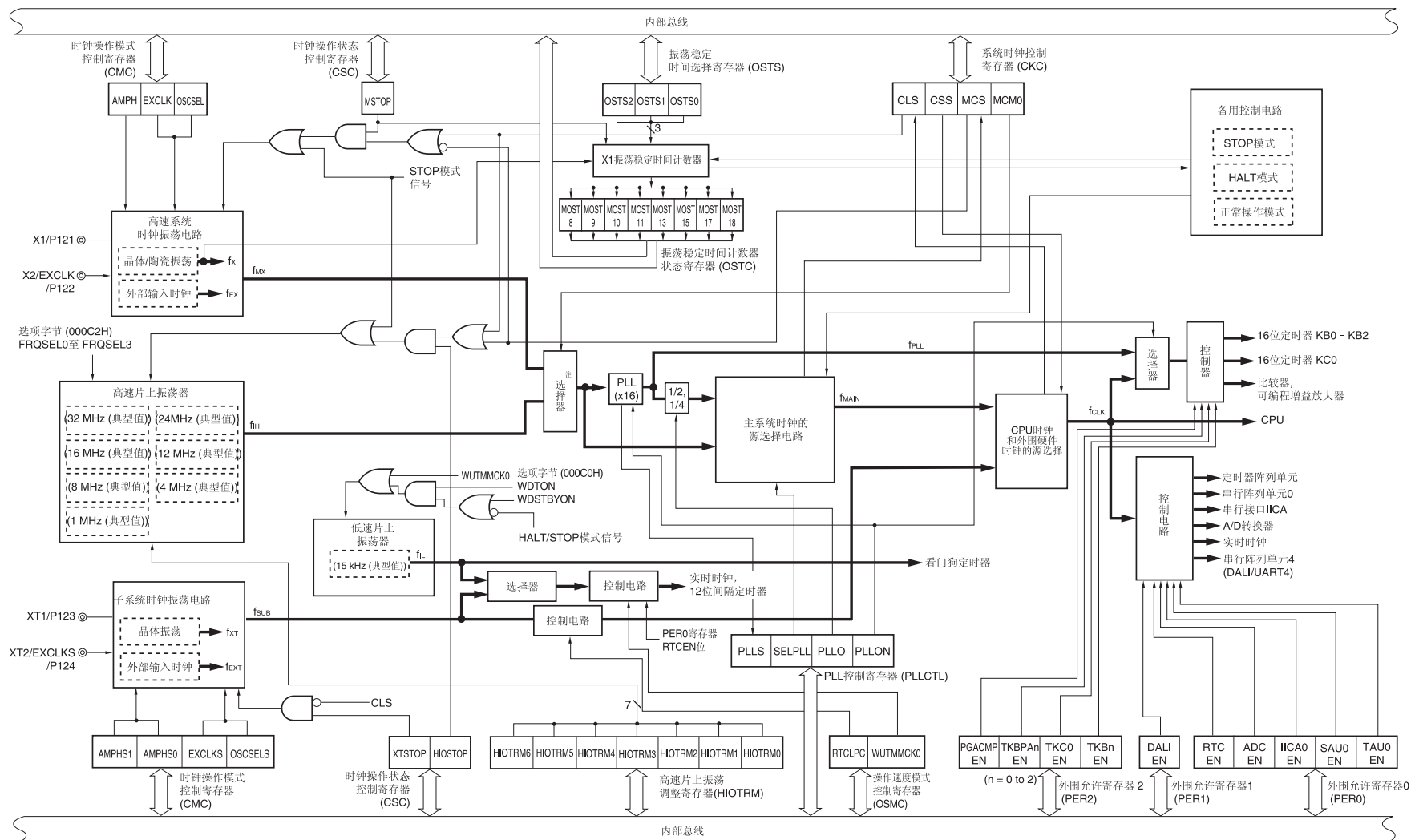
5.2 时钟发生电路的配置

时钟发生电路包括以下硬件。

表 5-1. 时钟发生电路的配置

项目	配置
控制寄存器	时钟操作模式控制寄存器(CMC) 系统时钟控制寄存器(CKC) 时钟操作状态控制寄存器(CSC) 振荡稳定时间计数器状态寄存器(OSTC) 振荡稳定时间选择寄存器(OSTS) PLL 控制寄存器 (PLLCTL) 外围允许寄存器 0 至 2 (PER0 至 PER2) 操作速度模式控制寄存器(OSMC) 高速片上振荡器调节寄存器(HIOTRM)
振荡电路	X1 振荡电路 XT1 振荡电路 高速片上振荡器 低速片上振荡器

图 5-1. 时钟发生电路的框图



(注和备注如下页所示。)

注意事项 当使用 PLL 输出功能时，振荡频率仅能选择 4 MHz。

备注	fx:	X1 时钟振荡频率
	f_{IH}:	高速片上振荡器时钟频率
	f_{EX}:	外部主系统时钟频率
	f_{MX}:	高速系统时钟频率
	f_{MAIN}:	主系统时钟频率
	f_{XT}:	XT1 时钟振荡频率
	f_{EXT}:	外部副系统时钟频率
	f_{SUB}:	副系统时钟频率
	f_{CLK}:	CPU/外围硬件时钟频率
	f_{IL}:	低速片上振荡器时钟频率
	f_{PLL}:	PLL 输出时钟频率

5.3 控制时钟发生电路的寄存器

以下 9 种寄存器可以控制时钟发生电路。

- 时钟操作模式控制寄存器(CMC)
- 系统时钟控制寄存器(CKC)
- 时钟操作状态控制寄存器(CSC)
- 振荡稳定时间计数器状态寄存器(OSTC)
- 振荡稳定时间选择寄存器(OSTS)
- PLL 控制寄存器(PLLCTL)
- 外围允许寄存器 0 至 2 (PER0 至 PER2)
- 操作速度模式控制寄存器(OSMC)
- 高速片上振荡器调节寄存器(HIOTRM)

(1) 时钟操作模式控制寄存器(CMC)

该寄存器用于设置 X1/P121、X2/EXCLK/P122、XT1/P123 和 XT2/EXCLKS/P124 引脚的操作模式，以及选择振荡电路的增益。

解除复位后，通过 8 位存储器操作指令只能写入 CMC 寄存器一次。使用 8 位存储器操作指令可以读取该寄存器。产生复位信号后，该寄存器被清除为 00H。

图 5-2. 时钟操作模式控制寄存器(CMC)的格式

地址: FFFA0H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
CMC	EXCLK	OSCSEL	EXCLKS	OSCSELS	0	AMPHS1	AMPHS0	AMPH
	EXCLK	OSCSEL	高速系统时钟引脚操作模式		X1/P121 引脚		X2/EXCLK/P122 引脚	
	0	0	输入端口模式		输入端口			
	0	1	X1 振荡模式		晶体/陶瓷谐振器连接			
	1	0	输入端口模式		输入端口			
	1	1	外部时钟输入模式		输入端口	外部时钟输入		
	EXCLKS	OSCSELS	副系统时钟引脚操作模式		XT1/P123 引脚		XT2/EXCLKS/P124 引脚	
	0	0	输入端口模式		输入端口			
	0	1	XT1 振荡模式		晶体谐振器连接			
	1	0	输入端口模式		输入端口			
	1	1	外部时钟输入模式		输入端口	外部时钟输入		
	AMPHS1	AMPHS0	XT1 振荡电路振荡模式					
	0	0	低功耗振荡 (默认)					
	0	1	正常振荡					
	1	0	超低功耗振荡					
	1	1	禁止设置					
	AMPH	X1 时钟振荡频率的控制						
	0	$1 \text{ MHz} \leq f_x \leq 10 \text{ MHz}$						
	1	$10 \text{ MHz} \leq f_x \leq 20 \text{ MHz}$						

- 注意事项
- 只能在解除复位后通过 8 位存储器操作指令写入 CMC 寄存器一次。当 CMC 寄存器采用初始值(00H)时, 必须在复位后将其设置为 00H, 以防止程序失控时的误动作。如果误将 00H 以外的值写入, 则不能恢复正常操作。
 - 解除复位后, 通过时钟操作状态控制寄存器(CSC)的设置, 在 X1 或 XT1 振荡开始之前设置 CMC 寄存器。
 - 当 X1 时钟振荡频率超过 10MHz 时, 必须将 AMPH 位设置为 1。
 - 在复位结束后(f_{CLK} 切换为 f_{MX} 之前) f_{IH} 被选择作为 f_{CLK} 时, 指定 AMPH、AMPHS1 和 AMPHS0 位的设置。
 - 使用软件对 f_{XT} 的振荡稳定时间进行计数。
 - 解除复位后, 在 32 引脚、30 引脚和 20 引脚产品中设置 EXCLKS, OSCSELS = 0, 1

(注意事项和备注如下页所示。)

- 注意事项 7. 尽管最大系统时钟频率是 32 MHz，但 X1 振荡器的最大频率是 20 MHz。
8. XT1 振荡电路是为了实现低功耗的低增幅电路。设计电路时请注意以下几点。
- 引脚和电路板中含有寄生电容。因此，要在实际使用的电路板上进行振荡评估，确认不存在问题。
 - 在超低功耗振荡(AMPHS1、AMPHS0 = 1、0)模式下使用 XT1 振荡电路时，请采用第三十二章 电特性 中推荐的谐振器。
 - 尽量缩短 XT1 和 XT2 引脚与谐振器之间的布线，尽量减小寄生电容和布线电阻。选用超低功耗振荡(AMPHS1、AMPHS0 = 1、0)时尤其要注意这一点。
 - 使用寄生电容和布线电阻小的材料构建电路板的电路。
 - 在 XT1 振荡电路的附近，请尽量配置与 Vss 尽可能相近电位的接地布局。
 - 确保 XT1 以及 XT2 引脚与谐振器之间的信号线不和其他信号线交叉。请勿将振荡器的布线靠近通过大波动电流的信号线。
 - 在高湿度环境中，电路板吸湿或结露可能导致 XT1 与 XT2 引脚之间的阻抗下降并使振荡受到影响。在此类环境中使用电路板时，要对电路板采取防潮措施，例如加涂层。
 - 在对电路板进行加涂层处理时，使用不会在 XT1 与 XT2 引脚之间产生电容或漏电的材料。

备注 fx: X1 时钟频率

(2) 系统时钟控制寄存器(CKC)

该寄存器用于选择 CPU/外围硬件时钟和主系统时钟。

使用 1 位或 8 位存储器操作指令来设置 CKC 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 5-3. 系统时钟控制寄存器(CKC) 的格式

地址: FFFA4H 复位后: 00H R/W^{※1}

符号	<7>	<6>	<5>	<4>	3	2	1	0
CKC	CLS	CSS	MCS	MCM0 ^{※2}	0	0	0	0
CLS	CPU/外围硬件时钟(f _{CLK})的状态							
0	主系统时钟(f _{MAIN})							
1	副系统时钟 (f _{SUB})							
CSS	CPU/外围硬件时钟(f _{CLK})的选择							
0	主系统时钟(f _{MAIN})							
1	副系统时钟 (f _{SUB})							
MCS	主系统时钟 (f _{MAIN})的状态							
0	高速片上振荡器时钟(f _{IH})							
1	高速系统时钟(f _{MX})							
MCM0 ^{※2}	PLL 输出时钟 (f _{PLL}) 或主系统时钟(f _{MAIN})的源时钟的选择							
0	选择高速片上振荡器时钟(f _{IH})							
1	选择高速系统时钟(f _{MX})							

注 1. 位 7 和位 5 为只读位。

2. CSS 位设置为 1 时，禁止改变 MCM0 位的值。

备注 1. f_{IH}: 高速片上振荡器时钟频率
 f_{MX}: 高速系统时钟频率
 f_{MAIN}: 主系统时钟频率
 f_{SUB}: 副系统时钟频率

2. x: 忽略

注意事项 1. 必须将位 3 设置为 0。

2. 由 CSS 位设置的时钟供应至 CPU 和外围硬件。因此，如果 CPU 时钟改变，供应至外围硬件的时钟（实时时钟、12 位间隔定时器和看门狗定时器除外）也同时改变。因此，要改变 CPU/外围硬件时钟时必须停止各外围功能。

3. 选用副系统时钟作为外围硬件时钟时，无法保证 A/D 转换器和 IICA 的操作。关于外围硬件的操作特性，请参阅描述各个外围硬件的章节以及第三十二章 电特性。

(3) 时钟操作状态控制寄存器(CSC)

该寄存器用于控制高速系统时钟、高速片上振荡器时钟和副系统时钟（低速片上振荡器时钟除外）的操作。

使用 1 位或 8 位存储器操作指令来设置 CSC 寄存器。

产生复位信号后，该寄存器被设置为 C0H。

图 5-4. 时钟操作状态控制寄存器(CSC) 的格式

地址: FFFA1H 复位后: C0H R/W

符号	<7>	<6>	5	4	3	2	1	<0>
CSC	MSTOP	XTSTOP	0	0	0	0	0	HIOSTOP
MSTOP	高速系统时钟的操作控制							
	X1 振荡模式		外部时钟输入模式			输入端口模式		
	0	X1 振荡电路工作	EXCLK 引脚的外部时钟有效			输入端口		
	1	X1 振荡电路停止	EXCLK 引脚的外部时钟无效					
XTSTOP	副系统时钟的操作控制							
	XT1 振荡模式		外部时钟输入模式			输入端口模式		
	0	XT1 振荡电路工作	EXCLKS 引脚的外部时钟有效			输入端口		
	1	XT1 振荡电路停止	EXCLKS 引脚的外部时钟无效					
HIOSTOP	高速片上振荡器时钟的操作控制							
	0	高速片上振荡器工作						
	1	高速片上振荡器停止						

- 注意事项**
- 解除复位后，设置 CSC 寄存器之前，先设置时钟操作模式控制寄存器(CMC)。
 - 解除复位后，设置 MSTOP 位为 0 之前，先设置振荡稳定时间选择寄存器(OSTS)。注意，使用 OSTS 寄存器的初始值时，无需设置 OSTS 寄存器。
 - 通过设置 MSTOP 位来启动 X1 振荡时，利用振荡稳定时间计数器状态寄存器(OSTC)来确认 X1 时钟的振荡稳定时间。
 - 通过设置 XSTOP 位来启动 XT1 振荡时，请利用软件设置等待副系统时钟所需的振荡稳定时间。
 - CPU/外围硬件时钟(f_{CLK})所选择的时钟，请不要在 OSC 寄存器使其停止。
 - 关于用于停止时钟振荡（使外部时钟输入无效）的寄存器标志的设置以及停止时钟振荡前的条件，请参阅表 5-2。

表 5-2. 停止时钟的方法

时钟	停止时钟前的条件 (使外部时钟输入无效)	CSC 寄存器标志 的设置
X1 时钟	CPU 和外围硬件工作于高速系统时钟以外的时钟。 (CLS = 0 和 MCS = 0, 或 CLS = 1)	MSTOP = 1
外部主系统时钟		
XT1 时钟	CPU 和外围硬件工作于副系统时钟以外的时钟。 (CLS = 0)	XTSTOP = 1
外部副系统时钟		
高速片上振荡器时钟	CPU 和外围硬件工作于高速片上振荡器时钟以外的时钟。 (CLS = 0 和 MCS = 1, 或 CLS = 1)	HIOSTOP = 1

(4) 振荡稳定时间计数器状态寄存器(OSTC)

该寄存器显示 X1 时钟振荡稳定时间计数器的计数状态。

在以下情况下可以确认 X1 时钟振荡稳定时间：

- 高速片上振荡器时钟或副系统时钟用作 CPU 时钟时，X1 时钟开始振荡。
- 高速片上振荡器时钟用作 CPU 时钟且 X1 时钟振荡时，进入 STOP 模式后解除 STOP 模式。

使用 1 位或 8 位存储器操作指令可以读取 OSTC 寄存器。

复位信号的产生，STOP 指令或 MSTOP 位（时钟操作状态控制寄存器(CSC)的位 7) = 1，会将 OSTC 寄存器清除为 00H。

备注 下列情况下，振荡稳定时间计数器开始计数。

- X1 时钟开始振荡(EXCLK、OSCSEL = 0、1 → MSTOP = 0)
- 解除 STOP 模式

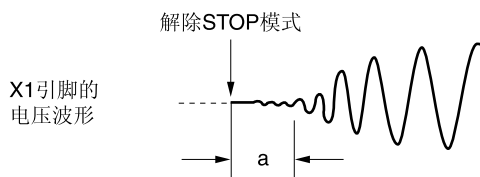
图 5-5. 振荡稳定时间计数器状态寄存器(OSTC) 的格式

地址: FFFA2H 复位后: 00H R

符号	7	6	5	4	3	2	1	0
OSTC	MOST 8	MOST 9	MOST 10	MOST 11	MOST 13	MOST 15	MOST 17	MOST 18

MOST 8	MOST 9	MOST 10	MOST 11	MOST 13	MOST 15	MOST 17	MOST 18	振荡稳定时间状态		
								fx = 10 MHz	fx = 20 MHz	
0	0	0	0	0	0	0	0	2 ⁸ /fx max.	25.6 μs max.	12.8 μs max.
1	0	0	0	0	0	0	0	2 ⁹ /fx min.	25.6 μs min.	12.8 μs min.
1	1	0	0	0	0	0	0	2 ⁹ /fx min.	51.2 μs min.	25.6 μs min.
1	1	1	0	0	0	0	0	2 ¹⁰ /fx min.	102.4 μs min.	51.2 μs min.
1	1	1	1	0	0	0	0	2 ¹¹ /fx min.	204.8 μs min.	102.4 μs min.
1	1	1	1	1	0	0	0	2 ¹³ /fx min.	819.2 μs min.	409.6 μs min.
1	1	1	1	1	1	0	0	2 ¹⁵ /fx min.	3.27 ms min.	1.64 ms min.
1	1	1	1	1	1	1	0	2 ¹⁷ /fx min.	13.11 ms min.	6.55 ms min.
1	1	1	1	1	1	1	1	2 ¹⁸ /fx min.	26.21 ms min.	13.11 ms min.

- 注意事项
1. 经过上述时间后，从 **MOST8** 位开始逐位依次变为 1 并保持 1 不变。
 2. 振荡稳定时间计数器仅在振荡稳定时间选择寄存器(OSTS)所设置的振荡稳定时间内计数。
 下列情况下，将 **OSTS** 寄存器的振荡稳定时间设置为大于利用 **OSTC** 寄存器进行确认的计数值。
 - 高速片上振荡器时钟或副系统时钟用作CPU时钟时，X1时钟开始振荡。
 - 高速片上振荡器时钟用作CPU时钟且X1时钟振荡时，进入STOP模式然后解除该模式。
 (注意，解除STOP模式后，只有OSTS寄存器所设置的振荡稳定时间内的状态被设置至OSTC寄存器。)
 3. X1 时钟振荡稳定等待时间不包括时钟振荡开始前的时间（下图“a”）。



备注 fx: X1 时钟振荡频率

(5) 振荡稳定时间选择寄存器(OSTS)

该寄存器用于选择解除 STOP 模式时的 X1 时钟振荡稳定等待时间。

当选用 X1 时钟作为 CPU 时钟时，解除 STOP 模式后，会自动等待 OSTS 寄存器设置的时间。

当选用高速片上振荡器时钟作为 CPU 时钟时，解除 STOP 模式后，请通过振荡稳定时间计数器状态寄存器(OSTC)确认所需振荡稳定时间已过。在 OSTS 寄存器预先设置的时间范围内都可以通过 OSTC 寄存器进行确认。

使用 8 位存储器操作指令来设置 OSTS 寄存器。

产生复位信号后，OSTS 寄存器被设置为 07H。

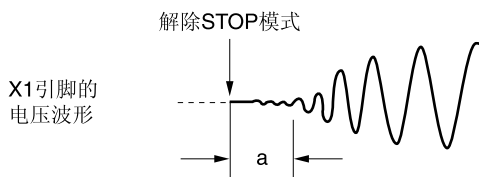
图 5-6. 振荡稳定时间选择寄存器(OSTS) 的格式

地址: FFFA3H 复位后: 07H R/W

符号	7	6	5	4	3	2	1	0
OSTS	0	0	0	0	0	OSTS2	OSTS1	OSTS0

OSTS2	OSTS1	OSTS0		选择振荡稳定时间	
				fx = 10 MHz	fx = 20 MHz
0	0	0	$2^8/f_x$	25.6 μ s	12.8 μ s
0	0	1	$2^9/f_x$	51.2 μ s	25.6 μ s
0	1	0	$2^{10}/f_x$	102.4 μ s	51.2 μ s
0	1	1	$2^{11}/f_x$	204.8 μ s	102.4 μ s
1	0	0	$2^{13}/f_x$	819.2 μ s	409.6 μ s
1	0	1	$2^{15}/f_x$	3.27 ms	1.64 ms
1	1	0	$2^{17}/f_x$	13.11 ms	6.55 ms
1	1	1	$2^{18}/f_x$	26.21 ms	13.11 ms

- 注意事项
1. 用 X1 时钟作为 CPU 时钟并进入 STOP 模式时，执行 STOP 指令之前须设置好 OSTS 寄存器。
 2. 若需更改 OSTS 寄存器的值，请在时钟工作状态控制寄存器(CSC)的 MSTOP 位设置为 0 之前设置。
 3. X1 时钟振荡稳定时间期间不要更改 OSTS 寄存器的值。
 4. 振荡稳定时间计数器仅在 OSTS 寄存器设置的振荡稳定时间内计数。
 下列情况下，将 OSTS 寄存器的振荡稳定时间设定为大于振荡开始后将通过 OSTC 寄存器确认的值。
 - 高速片上振荡器时钟或副系统时钟用作CPU时钟时，X1时钟开始振荡。
 - 高速片上振荡器时钟用作CPU时钟且X1时钟振荡时，进入STOP模式然后解除该模式。
 (注意，解除STOP模式后，只有OSTS寄存器所设置的振荡稳定时间内的状态被设置至OSTC寄存器。)
 5. X1 时钟振荡稳定等待时间不包括时钟振荡开始前的时间（下图“a”）。



备注 fx: X1 时钟振荡频率

(6) PLL 控制寄存器(PLLCTL)

该寄存器控制 PLL 功能。

使用 PLL 功能时，仅选择 4 MHz 用于高速系统时钟或高速片上振荡器时钟。

当稳压器处于低耗电模式时停止 PLL 功能。

使用 1 位或 8 位存储器操作指令设置 PLLCTL 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 5-7. PLL 控制寄存器(PLLCTL)的格式

地址: F050AH 复位后: 00H R/W

符号	7	6	5	4	<3>	<2>	<1>	<0>
PLLCTL	0	0	0	0	PLLS	SELPLL	PLLO	PLLON

PLLS	PLL 输出时钟供应状态标志
0	不供应
1	供应 (CPU/外围硬件时钟 (f _{CLK}) 工作于 16 MHz (PLLO = 1) 或 32 MHz (PLLO = 0))

SELPLL	用作 CPU/外围硬件时钟(f _{CLK})的 PLL 输出时钟的选择
0	不选择 PLL 输出时钟(由系统时钟控制寄存器(CKC) 选择的时钟被供应至 f _{CLK})
1	选择 PLL 输出时钟(16 MHz (PLLO = 1) 或 32 MHz (PLLO = 0) 被供应至 f _{CLK})

PLLO	PLL 输出分频位
0	2 分频时钟(32 MHz)
1	4 分频时钟(16 MHz)

PLLON	操作或停止 PLL 功能
0	停止
1	操作

注 位 3 为只读位。

注意事项 当 PLLON = 1 时，必须使主系统时钟处于运行状态(MSTOP = 0 或 HIOSTOP = 0)。

(7) 外围允许寄存器 0 至 2 (PER0 至 PER2)

该寄存器用于允许或禁止向外围硬件供应时钟。通过停止向未使用的硬件供应时钟，可以降低功耗和噪声。使用由该寄存器控制的下列外围功能时，进行外围功能初始设定之前，要将各功能对应的位置位 1。

- 实时时钟、12 位间隔定时器
- A/D 转换器
- 串行接口 IICA
- 串行阵列单元 0
- 串行阵列单元 4 (DALI/UART4)
- 串行阵列单元 0
- 16 位定时器 KB0 至 KB2
- 16 位定时器 KB0 至 KB2 的强制输出停止功能
- 16 位定时器 KC0
- 比较器/可编程增益放大器

使用 1 位或 8 位存储器操作指令来设置 PER0 至 PER2 寄存器。
产生复位信号后，该寄存器被清除为 00H。

图 5-8. 外围允许寄存器 0 (PER0) 的格式 (1/2)

地址: F00F0H 复位后: 00H R/W

符号	<7>	6	<5>	<4>	3	<2>	1	<0>
PER0	RTCEN	0	ADCEN	IICA0EN	0	SAU0EN	0	TAU0EN

RTCEN	控制实时时钟 (RTC) 和 12 位间隔定时器的输入时钟供应
0	停止输入时钟供应。 • 不可写入用于实时时钟(RTC)和 12 位间隔定时器的 SFR。 • 实时时钟(RTC)和 12 位间隔定时器处于复位状态。
1	允许输入时钟供应。 • 可以读取和写入用于实时时钟(RTC)和 12 位间隔定时器的 SFR。

ADCEN	A/D 转换器输入时钟供应的控制
0	停止输入时钟供应。 • 不可写入用于 A/D 转换器的 SFR。 • A/D 转换器处于复位状态。
1	允许输入时钟供应。 • 可以读取和写入用于 A/D 转换器的 SFR。

注意事项 必须将位 1、3 和 6 清除为“0”。

图 5-8. 外围允许寄存器 0 (PER0) 的格式 (2/2)

地址: F00F0H 复位后: 00H R/W

符号	<7>	6	<5>	<4>	3	<2>	1	<0>
PER0	RTCEN	0	ADCEN	IICA0EN	0	SAU0EN	0	TAU0EN

IICA0EN	串行接口 IICA 输入时钟供应的控制
0	停止输入时钟供应。 • 不可写入用于串行接口 IICA 的 SFR。 • 串行接口 IICA 处于复位状态。
1	允许输入时钟供应。 • 可以读取和写入用于串行接口 IICA 的 SFR。

SAU0EN	串行阵列单元 0 输入时钟供应的控制
0	停止输入时钟供应。 • 不可写入用于串行阵列单元 0 的 SFR。 • 串行阵列单元 0 处于复位状态。
1	允许输入时钟供应。 • 可以读取和写入用于串行阵列单元 0 的 SFR。

TAU0EN	定时器阵列单元输入时钟供应的控制
0	停止输入时钟供应。 • 不可写入用于定时器阵列单元的 SFR。 • 定时器阵列单元处于复位状态。
1	允许输入时钟供应。 • 可以读取和写入用于定时器阵列单元的 SFR。

注意事项 必须将位 1、3 和 6 清除为“0”。

图 5-9. 外围允许寄存器 1 (PER1) 的格式

地址: F0508H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	<0>
PER1	0	0	0	0	0	0	0	DALIEN

DALIEN	串行阵列单元 4 (DALI/UART4) 输入时钟供应的控制
0	停止输入时钟供应。 • 不可写入用于串行阵列单元 4 (DALI/UART4) 的 SFR。 • 串行阵列单元 4 (DALI/UART4) 处于复位状态。
1	允许输入时钟供应。 • 可以读取和写入用于串行阵列单元 4 (DALI/UART4) 的 SFR。

注意事项 必须将位 1 至 7 清除为“0”。

图 5-10. 外围允许寄存器 2 (PER2)的格式

地址: F0509H 复位后: 00H R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PER2	PGACMPEN	TKBPA2EN	TKBPA1EN	TKBPA0EN	TKC0EN	TKB2EN [※]	TKB1EN	TKB0EN

PGACMPEN	比较器/可编程增益放大器的输入时钟供应的控制
0	停止输入时钟供应。 <ul style="list-style-type: none"> 不可写入用于比较器/可编程增益放大器的 SFR。 比较器/可编程增益放大器处于复位状态。
1	允许输入时钟供应。 <ul style="list-style-type: none"> 可以读取和写入用于比较器/可编程增益放大器的 SFR。

TKBPAnEN	16 位定时器 K _{Bn} (n = 0 至 2)输入时钟供应的强制输出停止功能的控制
0	停止输入时钟供应。 <ul style="list-style-type: none"> 不可写入用于 16 位定时器 K_{Bn} 的强制输出停止功能的 SFR。 16 位定时器 K_{Bn} 的强制输出停止功能处于复位状态。
1	允许输入时钟供应。 <ul style="list-style-type: none"> 可以读取和写入用于 16 位定时器 K_{Bn} 的强制输出停止功能的 SFR。

TKC0EN	定时器KC0输入时钟的控制
0	停止输入时钟的供应。 <ul style="list-style-type: none"> 不可写入用于定时器KC0的SFR。 定时器KC0 处于复位状态。
1	供应输入时钟。 <ul style="list-style-type: none"> 可以读取/写入用于定时器KC0的SFR。

TKBnEN	16 位定时器 K _{Bn} (n = 0 至 2)输入时钟供应的控制
0	停止输入时钟供应。 <ul style="list-style-type: none"> 不可以写入用于 16 位定时器 K_{Bn} 的 SFR。 16 位定时器 K_{Bn} 处于复位状态。
1	允许输入时钟供应。 <ul style="list-style-type: none"> 可以读取和写入用于 16 位定时器 K_{Bn} 的 SFR。

注 仅限 30 引脚、32 引脚和 38 引脚产品。

(8) 操作速度模式控制寄存器(OSMC)

该寄存器是通过停止不必要的时钟功能来降低功耗的目的。

将 RTCLPC 位设为 1 时，在 STOP 模式下以及 CPU 工作于副系统时钟时的 HALT 模式下，停止向除实时时钟和 12 位间隔定时器以外的外围功能供应时钟，可以降低功耗。进行此项设置之前，先将外围允许寄存器 0(PER0)的位 7(RTCEN)设置为 1。

另外，OSMC 寄存器可用于选择实时时钟和 12 位间隔定时器的工作时钟。

使用 8 位存储器操作指令来设置 OSMC 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 5-11. 操作速度模式控制寄存器(OSMC) 的格式

地址: F00F3H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
OSMC	RTCLPC	0	0	WUTMMCK0	0	0	0	0

RTCLPC	STOP 模式下以及 CPU 工作于副系统时钟时的 HALT 模式下的设置
0	允许向外围功能供应副系统时钟 (关于允许操作的外围功能请参阅表 21-1。)
1	停止向外围功能供应除了实时时钟和 12 位间隔定时器以外的副系统时钟。

WUTMMCK0	选择实时时钟和 12 位间隔定时器的工作时钟
0	副系统时钟 (f _{SUB})
1	低速片上振荡器时钟(f _{IL})

注意事项 仅限 38 引脚产品中可使用 WUTMMCK0 位选择副系统时钟(f_{SUB})作为实时时钟和 12 位间隔定时器的操作时钟。

(9) 高速片上振荡器调节寄存器(HIOTRM)

该寄存器用于调节高速片上振荡器的精度。

通过采用高精度外部时钟输入（定时器阵列单元）的定时器等来对高速片上振荡器的频率进行自我测量，从而调节精度。

使用 8 位存储器操作指令来设置 HIOTRM 寄存器。

注意事项 调节精度后，如果温度和 V_{DD} 引脚电压发生改变，则频率也会有所变化。在温度和 V_{DD} 电压发生变化的情况下，必须定期或在需要高精度频率之前执行精度调节。

图 5-12. 高速片上振荡器调节寄存器(HIOTRM)的格式

地址: F00A0H 复位后: T.B.D R/W

符号	7	6	5	4	3	2	1	0
HIOTRM	0	HIOTRM6	HIOTRM5	HIOTRM4	HIOTRM3	HIOTRM2	HIOTRM1	HIOTRM0

HIOTRM	HIOTRM	HIOTRM	HIOTRM	HIOTRM	HIOTRM	HIOTRM	高速片上振荡器
6	5	4	3	2	1	0	
0	0	0	0	0	0	0	最低速度
0	0	0	0	0	0	1	↑ ↓
0	0	0	0	0	1	0	
0	0	0	0	0	1	1	
0	0	0	0	1	0	0	
			• • •				
1	1	1	1	1	1	0	
1	1	1	1	1	1	1	最高速度

注 复位值因芯片而异。

5.4 系统时钟振荡电路

5.4.1 X1 振荡电路

X1 振荡电路是通过连接至 X1 和 X2 引脚的晶体谐振器或陶瓷谐振器（1 至 20MHz）产生振荡的。

也可输入外部时钟。此时，将时钟信号输入至 EXCLK 引脚。

使用 X1 振荡电路时，须将时钟操作模式控制寄存器(CMC)的位 7 和位 6(EXCLK、OSCSEL)进行如下设置。

- 晶体或者陶瓷振荡： EXCLK, OSCSEL = 0, 1
- 外部时钟输入： EXCLK, OSCSEL = 1, 1

不使用 X1 振荡电路时，设置为输入端口模式(EXCLK、OSCSEL = 0、0)。

此外，也不用作输入端口引脚使用时，请参阅表 2-2. 未使用引脚的连接。

X1 振荡电路的外接电路示例如图 5-13 所示。

图 5-13. X1 振荡电路的外接电路示例



注意事项如下页所示。

5.4.2 XT1 振荡电路

XT1 振荡电路是通过连接至 XT1 和 XT2 引脚的晶体谐振器（标准：32.768 kHz）产生振荡的。

使用 XT1 振荡电路时，须将时钟操作模式控制寄存器(CMC)的位 4(OSCSELS)设置为 1。

也可输入外部时钟。此时，将时钟信号输入至 EXCLKS 引脚。

使用 XT1 振荡电路时，须将时钟操作模式控制寄存器(CMC)的位 5 和位 4(EXCLKS、OSCSELS)进行如下设置。

- 晶体振荡： EXCLKS, OSCSELS = 0, 1
- 外部时钟输入： EXCLKS, OSCSELS = 1, 1

不使用 XT1 振荡电路时，设置为输入端口模式(EXCLKS、OSCSELS = 0、0)。

此外，也不用作输入端口引脚时，请参阅表 2-2. 未使用引脚的连接。

XT1 振荡电路的外接电路示例如图 5-14 所示。

图 5-14. XT1 振荡电路的外接电路示例



注意事项 1. 使用 X1 振荡电路和 XT1 振荡电路时，按照以下要求对图 5-13 和 5-14 中虚线框起来的部分进行布线，从而避免布线电容的不良影响。

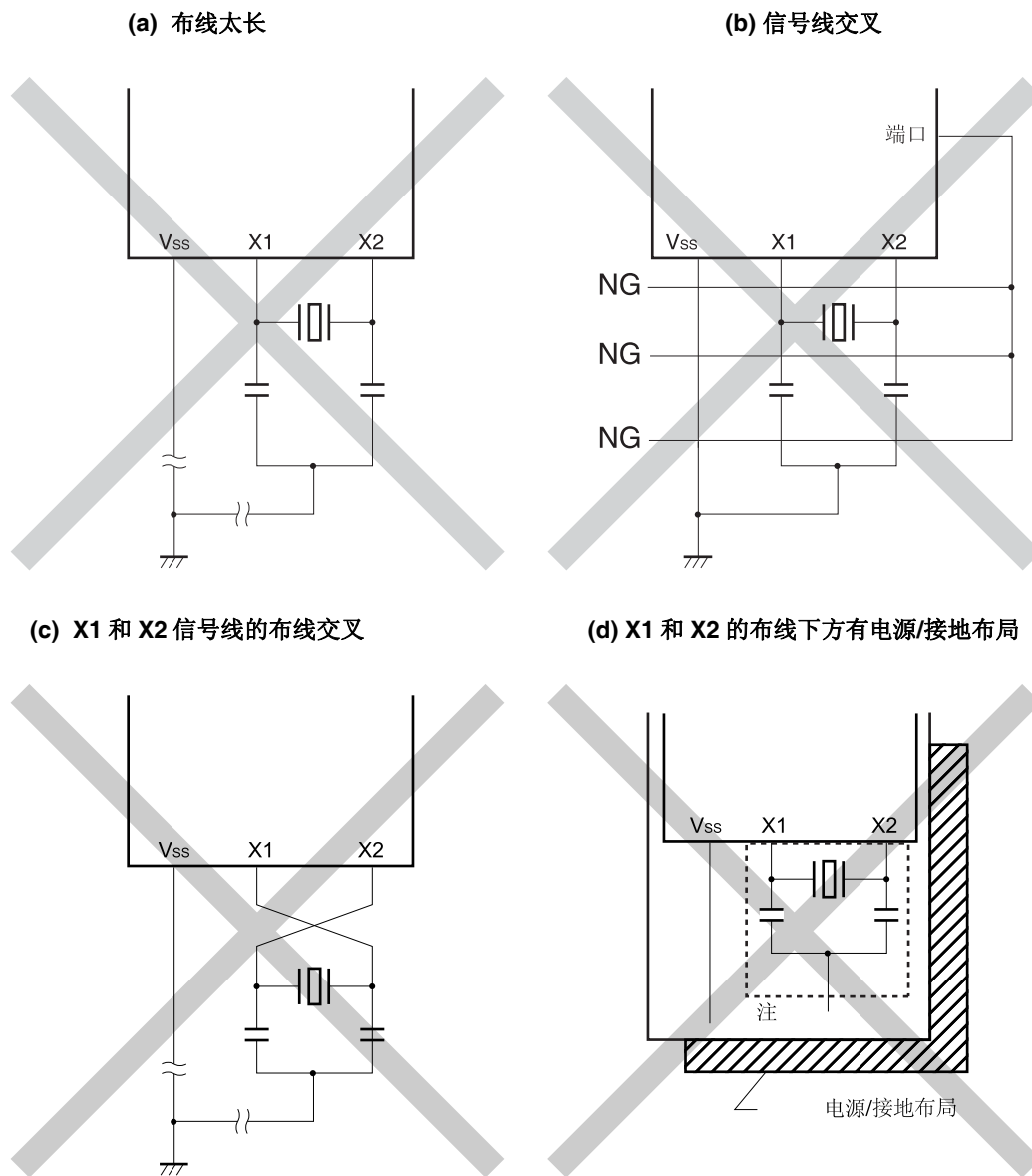
- 布线要保证尽可能地短。
- 振荡器布线不要和其他的信号线互相交叉。请勿将振荡器的布线靠近通过大波动电流的信号线。
- 始终保持振荡电路电容的接地点与 Vss 具有同电位。请勿将电容的接地点连接到通过大电流的接地布局。
- 请勿从振荡电路提取信号。

XT1 振荡电路是用于实现低功耗的低增幅电路。设计电路时请注意以下几点。

- 引脚和电路板间含有寄生电容。因此，要在实际使用的电路板上进行振荡评估，确认不存在问题。
- 在超低功耗振荡(AMPHS1、AMPHS0 = 1、0)模式下使用 XT1 振荡电路时，请使用第三十二章 电特性中推荐的谐振器。
- 尽量缩短 XT1 和 XT2 引脚与谐振器之间的布线，尽量减小寄生电容和布线电阻。选用超低功耗振荡(AMPHS1、AMPHS0 = 1、0)时尤其要注意这一点。
- 使用寄生电容和布线电阻小的材料构建电路板的电路。
- 在 XT1 振荡电路的附近，请尽量配置与 Vss 尽可能相近电位的接地布局。
- 确保 XT1 以及 XT2 引脚与谐振器之间的信号线不和其他信号线交叉。请勿将振荡器的布线靠近通过大波动电流的信号线。
- 在高湿度环境中，电路板吸湿或结露可能导致 XT1 与 XT2 引脚之间的阻抗下降并使振荡受到影响。在此类环境中使用电路板时，要对电路板采取防潮措施，例如加涂层。
- 在对电路板进行加涂层处理时，使用在 XT1 与 XT2 引脚之间不会产生电容或漏电的材料。

图 5-15 为不正确的谐振器连接示例。

图 5-15. 不正确的谐振器连接示例(1/2)

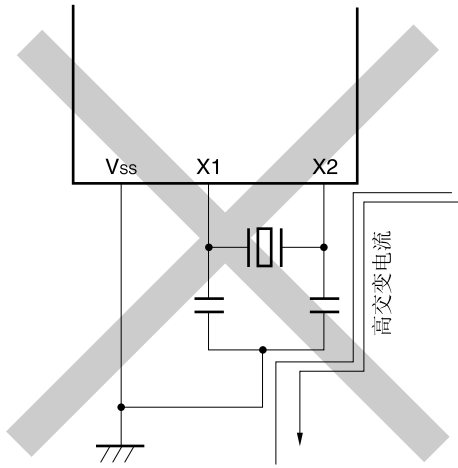


注 在多层板或双面板中，不要在 X1 和 X2 引脚以及谐振器的布线区（图中虚线围住的区域）设置电源/接地布局。不要配置会形成电容效应并影响振荡特性的布局。

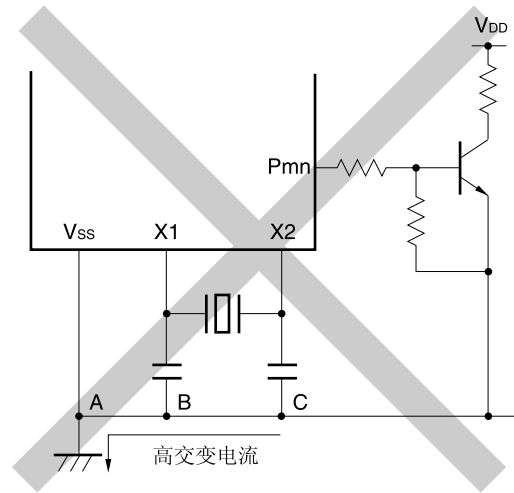
备注 使用副系统时钟时，分别用 XT1 和 XT2 代替 X1 和 X2。同时，在 XT2 侧串入电阻。

图 5-15. 不正确的谐振器连接示例(2/2)

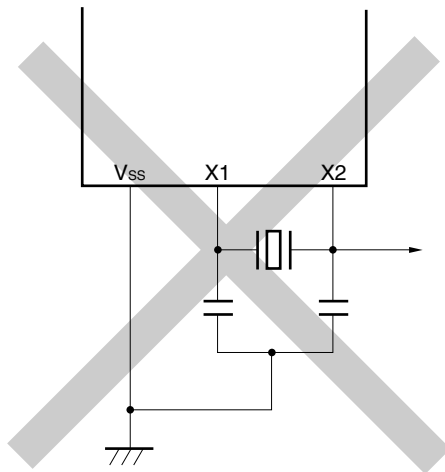
(e) 在高交变电流的附近接线



(f) 电流通过振荡电路的接地线 (A、B、C 点的电位会出现波动)



(g) 信号提取



注意事项 当 X2 和 XT1 并行布线时，X2 的串扰噪声会叠加于 XT1 上，从而导致错误动作。

备注 使用副系统时钟时，分别用 XT1 和 XT2 代替 X1 和 X2。同时，在 XT2 侧串入电阻。

5.4.3 高速片上振荡器

RL78/I1A 内置有高速片上振荡器。通过选项字节(000C2H)可从以下范围中选择频率：32、24、16、12、8、4 或 1 MHz。可以通过时钟操作状态控制寄存器(CSC)的位 0(HIOSTOP)来控制振荡。解除复位后高速片上振荡器自动开始振荡。

5.4.4 低速片上振荡器

RL78/I1A 内置有低速片上振荡器。

低速片上振荡器时钟只能用作看门狗定时器、实时时钟和 12 位间隔定时器时钟的时钟。不能用作 CPU 时钟。

当选项字节(000C0H)的位 4(WDTON)或操作速度模式控制寄存器(OSMC)的位 4(WUTMMCK0)或者两者同时设置为 1 时, 该时钟工作。

除非看门狗定时器停止工作且 WUTMMCK0 的值为零, 否则低速片上振荡器将持续振荡。当看门狗定时器运行时, 即使程序失控, 低速片上振荡器时钟也不会停止。

5.4.5 PLL (锁相环)

RL78/I1A 内置有 PLL 电路。

使用 PLL 可以倍增高速片上振荡器时钟或高速系统时钟。

可以通过 PLL 控制寄存器(PLLCTL)的位 0 (PLLON)来控制 PLL 电路的操作。

使用 PLL 功能时, 仅选择 4 MHz 用于高速系统时钟或高速片上振荡器时钟。

- 注意事项**
1. 从 PLL 模式切换至高速片上振荡器时钟和高速系统时钟时, 停止提供 PLL 输出时钟(f_{PLL})的功能(定时器 KB0 至 KB2、KC0 和比较器/可编程增益放大器)。
 2. 当工作于副系统时钟时, 不能执行 PLL 操作。
 3. 切换至 STOP 模式时, 须在停止 PLL 功能(PLL 控制寄存器(PLLCTL)中 SELPLL = 0 → PLLON = 0)后, 选择高速片上振荡器时钟(f_{IH})或高速系统时钟(f_{MX})作为主系统时钟, 然后执行 STOP 指令。

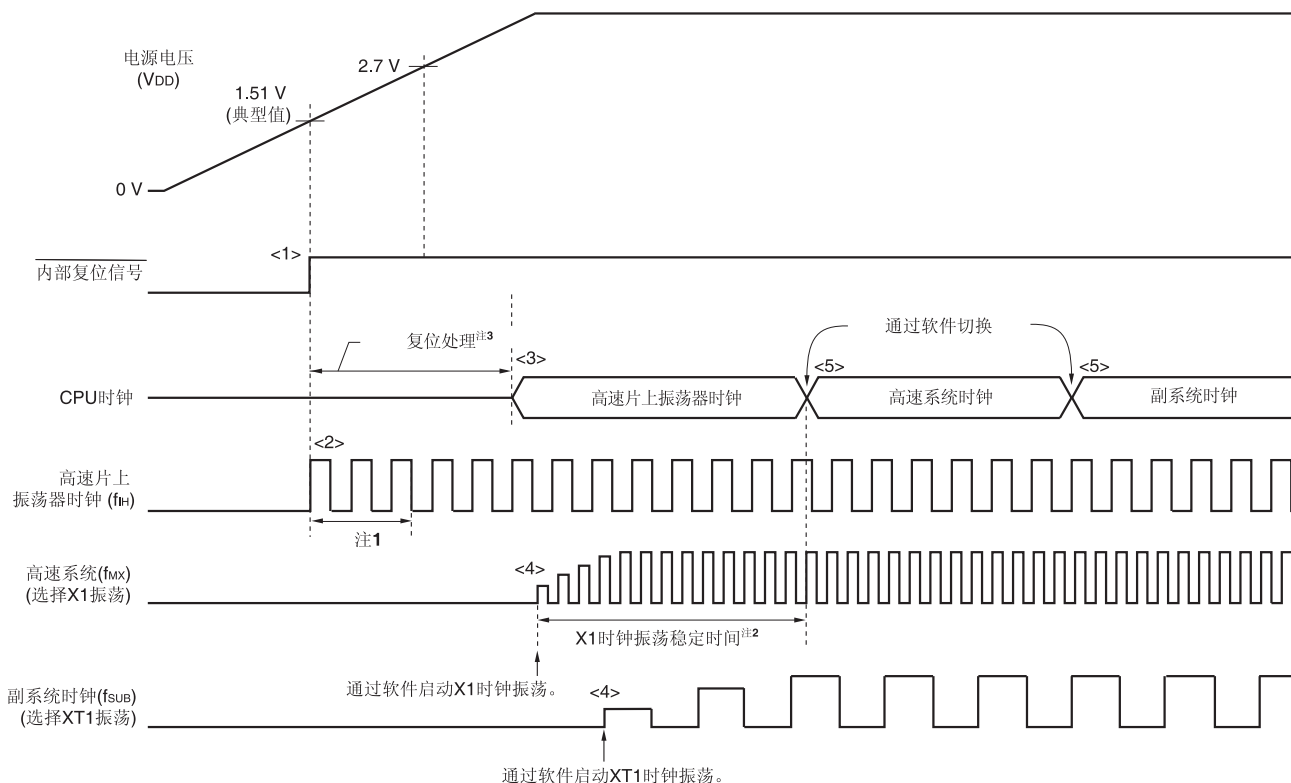
5.5 时钟发生电路的操作

时钟发生电路产生如下所示时钟，并控制如待机模式等 CPU 的操作模式(参阅图 5-1)。

- 主系统时钟 f_{MAIN}
 - 高速系统时钟 f_{MX}
 - X1 时钟 f_x
 - 外部主系统时钟 f_{EX}
 - 高速片上振荡器时钟 f_{IH}
- 副系统时钟 f_{SUB}
 - XT1 时钟 f_{XT}
 - 外部副系统时钟 f_{EXT}
- 低速片上振荡器时钟 f_{IL}
- CPU/外围硬件时钟 f_{CLK}
- PLL 输出时钟 f_{PLL}

在 RL78/I1A 中，解除复位后，CPU 将在高速片上振荡器开始输出后开始工作。电源电压投入时，时钟发生电路的操作如图 5-16 所示。

图 5-16. 电源投入时时钟发生电路的操作



- <1> 投入电源后，上电复位(POR)电路产生内部复位信号。
- <2> 当电源电压超过 1.51V(TYP.)时，复位被解除，高速片上振荡器自动开始振荡。
- <3> 解除复位后，执行例如等待电源电压或稳压器电压稳定等的复位处理后，CPU 以高速片上振荡器时钟开始运行。
- <4> 通过软件启动 X1 或 XT1 时钟的振荡（参阅 5.6.2 X1 振荡电路的设置示例和 5.6.3 XT1 振荡电路的设置示例）。
- <5> 将 CPU 时钟切换为 X1 或 XT1 时钟时，等时钟振荡稳定后，通过软件切换（参阅 5.6.2 X1 振荡电路的设置示例和 5.6.3 XT1 振荡电路的设置示例）。

注 1. 内部复位处理时间包括高速片上振荡器时钟的振荡精度稳定等待时间。

2. 解除复位时，通过振荡稳定时间计数器状态寄存器(OSTC)确认 X1 时钟的振荡稳定时间。

3. 复位处理时间： 497 至 720 μ s (LVD 启用时)

265 至 407 μ s (LVD 关闭时)

注意事项 使用来自 EXCLK 引脚的外部时钟输入时，不必等待振荡稳定时间。

备注 使用 PLL 时，请参阅 5.6.4 PLL 电路的设置示例。

5.6 控制时钟

5.6.1 高速片上振荡器的设置示例

解除复位后，CPU/外围硬件时钟 (f_{CLK}) 总是开始工作于高速片上振荡器时钟。通过选项字节(000C2H)的 FRQSEL0 至 FRQSEL3，可从以下范围内选择高速片上振荡器的频率：32、24、16、12、8、4 和 1MHz。

[选项字节设置]

地址：000C2H

	7	6	5	4	3	2	1	0
选项字节 (000C2H)	CMODE1	CMODE0			FRQSEL3	FRQSEL2	FRQSEL1	FRQSEL0
	1	0/1	1	0	0/1	0/1	0/1	0/1

CMODE1	CMODE0	Flash 操作模式的设置	
1	0	LS (低速主)模式	V _{DD} = 2.7 V 至 5.5 V @ 1 MHz 至 8 MHz
1	1	HS (高速主)模式	V _{DD} = 2.7 V 至 5.5 V @ 1 MHz 至 32 MHz

FRQSEL3	FRQSEL2	FRQSEL1	FRQSEL0	高速片上振荡器的频率
1	0	0	0	32 MHz
0	0	0	0	24 MHz
1	0	0	1	16 MHz
0	0	0	1	12 MHz
1	0	1	0	8 MHz
1	0	1	1	4 MHz
1	1	0	1	1 MHz
其他				禁止设置

5.6.2 X1 振荡电路的设置示例

解除复位后，CPU/外围硬件时钟 (f_{CLK}) 总是开始工作于高速片上振荡器时钟。之后，将时钟切换为 X1 振荡时钟时，须利用振荡稳定时间选择寄存器(OSTS)、时钟操作模式控制寄存器(CMC)和时钟操作状态控制寄存器(CSC)设置振荡器以及控制振荡的开始，并通过振荡稳定时间计数器状态寄存器(OSTC)等待振荡稳定。振荡稳定后，使用系统时钟控制寄存器(CKC)将 X1 振荡时钟设置为 f_{CLK}。

[寄存器设置] 按照如下<1>至<5>的顺序设置寄存器。

<1> 将 CMC 寄存器的 OSCSEL 位置位(1)，在 f_x > 10 MHz 的情况时，将 AMPH 位置位(1)，使 X1 振荡电路开始工作。

	7	6	5	4	3	2	1	0
CMC	EXCLK	OSCSEL	EXCLKS	OSCSELS		AMPHS1	AMPHS0	AMPH
	0	1	0	0	0	0	0	1

AMPH 位：当 X1 振荡时钟等于或小于 10MHz 时，将该位设置为 0。

<2> 解除 STOP 模式时，用 OSTS 寄存器选择 X1 振荡电路的振荡稳定时间。

示例：10MHz 谐振器需要等待至少 102.4 μs 时，设置为以下值。

	7	6	5	4	3	2	1	0
OSTS						OSTS2	OSTS1	OSTS0
	0	0	0	0	0	0	1	0

<3> 清除 CSC 寄存器的 MSTOP 位为 0，使 X1 振荡电路开始振荡。

	7	6	5	4	3	2	1	0
CSC	MSTOP	XTSTOP						HIOSTOP
	0	1	0	0	0	0	0	0

<4> 通过 OSTC 寄存器等待 X1 振荡电路的振荡稳定。

示例：10MHz 谐振器需要等待至少 102.4 μs 时，须等到各个位成为以下的值为止。

	7	6	5	4	3	2	1	0
OSTC	MOST8	MOST9	MOST10	MOST11	MOST13	MOST15	MOST17	MOST18
	1	1	1	0	0	0	0	0

<5> 用 CKC 寄存器的 MCM0 位将 X1 振荡时钟设置为 CPU/外围硬件时钟。

	7	6	5	4	3	2	1	0
CKC	CLS	CSS	MCS	MCM0				
	0	0	0	1	0	0	0	0

5.6.3 XT1 振荡电路的设置示例

解除复位后，CPU/外围硬件时钟 (fCLK) 总是开始工作于高速片上振荡器时钟。之后，将时钟切换为 XT1 振荡时钟时，须利用操作速度模式控制寄存器(OSMC)、时钟操作模式控制寄存器(CMC)和时钟操作状态控制寄存器(CSC)设置振荡器以及控制振荡的开始，并通过系统时钟控制寄存器(CKC)将 XT1 振荡时钟设置为 fCLK。

[寄存器设置] 按照如下<1>至<5>的顺序设置寄存器。

<1> 在 STOP 模式及 CPU 工作于副系统时钟的 HALT 模式下，如果仅使工作于副系统时钟的实时时钟和间隔定时器工作（超低电流消耗）时，将 RTCLPC 位设置为 1。

	7	6	5	4	3	2	1	0
OSMC	RTCLPC			WUTMMCK0				
	0/1	0	0	0	0	0	0	0

<2> 将 CMC 寄存器的 OSCSELS 位置位(1)，使 XT1 振荡电路开始工作。

	7	6	5	4	3	2	1	0
CMC	EXCLK	OSCSEL	EXCLKS	OSCSELS		AMPHS1	AMPHS0	AMPH
	0	0	0	1	0	0/1	0/1	0

AMPHS0 和 AMPHS1 位：设置 XT1 振荡电路的振荡模式。

<3> 将 CSC 寄存器的 XTSTOP 位清除为 0，使 XT1 振荡电路开始工作。

	7	6	5	4	3	2	1	0
CSC	MSTOP	XTSTOP						HIOSTOP
	1	0	0	0	0	0	0	0

<4> 借助软件，使用定时器功能或其他功能等待副系统时钟振荡稳定。

<5> 使用 CKC 寄存器的 CSS 位设置 XT1 振荡时钟为 CPU/外围硬件时钟。

	7	6	5	4	3	2	1	0
CKC	CLS	CSS	MCS	MCM0				
	0	1	0	0	0	0	0	0

5.6.4 PLL电路的设置示例

设置高速系统时钟和高速片上振荡器时钟 (参阅 5.6.1 高速片上振荡器的设置示例和 5.6.2 X1 振荡电路的设置示例)后, 使用 PLL 控制寄存器(PLLCTL)来控制 PLL 电路。

使用 PLL 功能时, 仅选择 4 MHz 用于高速系统时钟或高速片上振荡器时钟。

[寄存器设置] 按照如下<1>至<5>的顺序设置寄存器。

<1> 设置 PLLCTL 寄存器的 PLLO 位, 使 $f_{PLL} = 64 \text{ MHz}$ 被设置用于供应 16 位定时器 KB0 至 KB2 和 KC0、比较器以及可编程增益放大器, 使 16 MHz (PLLO = 1)或 32 MHz (PLLO = 0) 被设置用于供应其他硬件(CPU 和外围硬件)。

	7	6	5	4	3	2	1	0
PLLCTL					PLLS	SELPLL	PLLO	PLLON
	0	0	0	0	0	0	0/1	0

<2> 将 PLLCTL 寄存器的 PLLON 位设置为(1), 使 PLL 电路开始工作。

	7	6	5	4	3	2	1	0
PLLCTL					PLLS	SELPLL	PLLO	PLLON
	0	0	0	0	0	0	0/1	1

<3> 使用软件等待 40 μs 。

<4> 设置 PLLCTL 寄存器的 SELPLL 位为 1, 来选择用于 CPU 和外围硬件时钟(f_{CLK})的 PLL 输出。

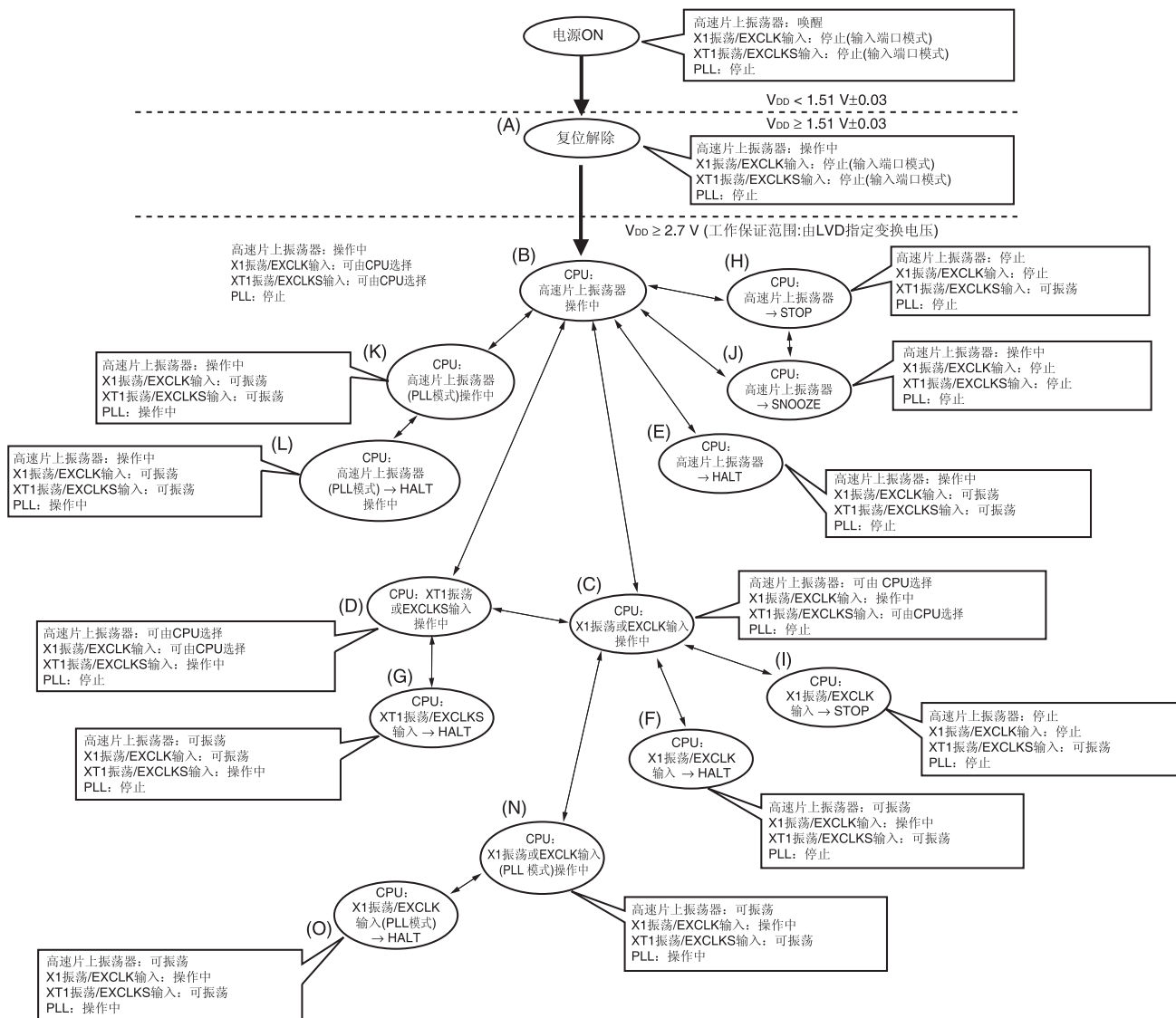
	7	6	5	4	3	2	1	0
PLLCTL					PLLS	SELPLL	PLLO	PLLON
	0	0	0	0	0	1	0/1	1

<5> 等待直到 PLLS 位变为 1。

5.6.5 CPU时钟状态转换图

本产品的CPU时钟状态转换图如图5-17所示。

图 5-17. CPU 时钟状态转换图



CPU 时钟转换和 SFR 寄存器的设置示例如表 5-3 所示。

表 5-3. CPU 时钟转换和 SFR 寄存器的设置示例(1/6)

(1) 解除复位(A)后, CPU 工作于高速片上振荡器时钟(B)

状态转换	SFR 寄存器设置
(A) → (B)	无需设置 SFR 寄存器 (解除复位后的初始状态)

(2) 解除复位(A)后, CPU 工作于高速系统时钟(C)

(CPU 在解除复位(B)后立即工作于高速片上振荡器时钟。)

(SFR 寄存器的设置序列) →

SFR 寄存器的设置标志 状态转换	CMC 寄存器 ^注			OSTS 寄存器	CSC 寄存器 MSTOP	OSTC 寄存器	CKC 寄存器 MCM0
	EXCLK	OSCSEL	AMPH				
(A) → (B) → (C) (X1 时钟: $1 \text{ MHz} \leq f_x \leq 10 \text{ MHz}$)	0	1	0	注 2	0	需要确认	1
(A) → (B) → (C) (X1 时钟: $10 \text{ MHz} \leq f_x \leq 20 \text{ MHz}$)	0	1	1	注 2	0	需要确认	1
(A) → (B) → (C) (外部主时钟)	1	1	×	注 2	0	无需确认	1

注 1. 只能在解除复位后通过 8 位存储器操作指令写入时钟操作模式控制寄存器(CMC)一次。

2. 设置振荡稳定时间如下。

- 期待振荡稳定时间计数器状态寄存器(OSTC)的振荡稳定时间 ≤ 振荡稳定时间选择寄存器(OSTS)设置的振荡稳定时间

注意事项 当电源电压达到待设置时钟的可操作电压之后, 再设置该时钟(参阅第三十二章 电特性)。

(3) 解除复位(A)后, CPU 工作于副系统时钟(D)

(CPU 在解除复位(B)后立即工作于高速片上振荡器时钟。)

(SFR 寄存器的设置序列) →

SFR 寄存器的设置标志 状态转换	CMC 寄存器 ^注				CSC 寄存器 XTSTOP	等待振荡 稳定	CKC 寄存器 CSS
	EXCLKS	OSCSELS	AMPHS1	AMPHS0			
(A) → (B) → (D) (XT1 时钟)	0	1	0/1	0/1	0	需要等待	1
(A) → (B) → (D) (外部副时钟)	1	1	×	×	0	需要等待	1

注 解除复位后只能通过 8 位存储器操作指令写入时钟操作模式控制寄存器(CMC)一次。

备注 1. ×: 忽略

2. 表 5-3 的(A)至(O)对应于图 5-17 的(A)至(O)。

表 5-3. CPU 时钟转换和 SFR 寄存器的设置示例(2/6)

(4) CPU 时钟从高速片上振荡器时钟(B)切换至高速系统时钟(C)

(SFR 寄存器的设置序列) →

SFR 寄存器的设置标志 状态转换	CMC 寄存器 ^{注1}			OSTS 寄存器	CSC 寄存器	OSTC 寄存器	CKC 寄存器
	EXCLK	OSCSEL	AMPH		MSTOP		
(B) → (C) (X1 时钟: 1 MHz ≤ fX ≤ 10 MHz)	0	1	0	注 2	0	需要确认	1
(B) → (C) (X1 时钟: 10 MHz < fX ≤ 20 MHz)	0	1	1	注 2	0	需要确认	1
(B) → (C) (外部主时钟)	1	1	×	注 2	0	无需确认	1

曾经设置完毕则无必要
 高速系统时钟振荡中则无必要

注 1. 解除复位后，只能更改时钟操作模式控制寄存器(CMC)一次。已经设置完毕的情况下不需要再次设置。

2. 按下述方式设置振荡稳定时间。

- 振荡稳定时间计数器状态寄存器(OSTC)所需的振荡稳定时间 ≤ 振荡稳定时间选择寄存器(OSTS)设置的振荡稳定时间

注意事项 当电源电压达到待设置时钟的可操作电压之后，再设置该时钟(参阅第三十二章 电特性)。

(5) CPU 时钟从高速片上振荡器时钟(B)切换至副系统时钟(D)

(SFR 寄存器的设置序列) →

SFR 寄存器的设置标志 状态转换	CMC 寄存器 ^注		CSC 寄存器	等待振荡稳定	CKC 寄存器
	EXCLKS	OSCSELS	XTSTOP		CSS
(B) → (D) (XT1 时钟)	0	1	0	需要等待	1
(B) → (D) (外部副时钟)	1	1	0	需要等待	1

副系统时钟振荡中则无必要

注 解除复位后只能通过 8 位存储器操作指令写入时钟操作模式控制寄存器(CMC)一次。

备注 1. ×: 忽略

2. 表 5-3 的(A)至(O)对应于图 5-17 的(A)至(O)。

表 5-3. CPU 时钟转换和 SFR 寄存器的设置示例(3/6)

(6) CPU 时钟从高速系统时钟(C)切换至高速片上振荡器时钟(B)

(SFR 寄存器的设置序列) →

SFR 寄存器的设置标志	CSC 寄存器	振荡精确度稳定时间	CKC 寄存器
	HIOSTOP		MCM0
状态转换			
(C) → (B)	0	10 μs	0

高速片上振荡器时钟振荡中则无必要

(7) CPU 时钟从高速系统时钟(C)切换至副系统时钟(D)

(SFR 寄存器的设置序列) →

SFR 寄存器的设置标志	CSC 寄存器	等待振荡稳定	CKC 寄存器
	XTSTOP		CSS
状态转换			
(C) → (D)	0	需要等待	1

副系统时钟振荡中则无必要

(8) CPU 时钟从副系统时钟(D)切换至高速片上振荡器时钟(B)

(SFR 寄存器的设置序列) →

SFR 寄存器的设置标志	CSC 寄存器	CKC 寄存器	
	HIOSTOP	CSS	MCM0
状态转换			
(D) → (B)	0	0	0

高速片上振荡器时钟振荡中则无必要

曾经设置完毕则无必要

备注 表 5-3 的(A)至(O)对应于图 5-17 的(A)至(O)。

表 5-3. CPU 时钟转换和 SFR 寄存器的设置示例(4/6)

(9) CPU 时钟从副系统时钟(D)切换至高速系统时钟(C)

(SFR 寄存器的设置序列) →

SFR 寄存器的设置标志 状态转换	OSTS 寄存器	CSC 寄存器	OSTC 寄存器	CKC 寄存器	
		MSTOP		CSS	MCM0
(D) → (C) (X1 时钟: $1 \text{ MHz} \leq f_x \leq 10 \text{ MHz}$)	注	0	须要确认	0	1
(D) → (C) (X1 时钟: $10 \text{ MHz} < f_x \leq 20 \text{ MHz}$)	注	0	须要确认	0	1
(D) → (C) (外部主时钟)	注	0	无须确认	0	1

高速系统时钟振荡中则不必要
曾经设置完毕则不必要

注 按下述方式设置振荡稳定时间。

- 振荡稳定时间计数器状态寄存器(OSTC)所需的振荡稳定时间 ≤ 振荡稳定时间选择寄存器(OSTS)设置的振荡稳定时间

注意事项 等电源电压达到待设置时钟的可操作电压之后,再设置该时钟(参阅第三十二章 电特性)。

备注 表 5-3 的(A)至(O)对应于图 5-17 的(A)至(O)。

表 5-3. CPU 时钟转换和 SFR 寄存器的设置示例(5/6)

- (10) • CPU 时钟从高速片上振荡器时钟(B)切换为高速片上振荡器时钟(PLL 模式) (K)
- CPU 时钟从高速系统时钟(C)切换为高速系统时钟(PLL 模式) (N)

(SFR 寄存器的设置序列) →

SFR 寄存器的设置标志	PLLCTL 寄存器			
	PLLO	PLLON	等待振荡稳定	SELPLL
状态转换				
(B) → (K)	0/1	1	40 μs	1
(C) → (N)				

- (11) • CPU 时钟从高速片上振荡器时钟(PLL 模式) (K)切换为高速片上振荡器时钟(B)
- CPU 时钟从高速系统时钟(PLL 模式) (N)切换为高速系统时钟(N)

(SFR 寄存器的设置序列) →

SFR 寄存器的设置标志	PLLCTL 寄存器		
	SELPLL	PLLS	PLLON
状态转换			
(K) → (B)	0	0	0
(N) → (C)			

注意事项 从 PLL 模式切换至高速片上振荡器时钟和高速系统时钟时，停止提供 PLL 输出时钟(f_{PLL})的功能(定时器 KB0 至 KB2、KC0 和比较器/可编程增益放大器)。

- (12) • 当 CPU 工作于高速片上振荡器时钟(B)时切换为 HALT 模式(E)
- 当 CPU 工作于高速系统时钟(C)时切换为 HALT 模式(F)
- 当 CPU 工作于副系统时钟(D)时切换为 HALT 模式(G)
- 当 CPU 工作于高速片上振荡器时钟(PLL 模式) (K)时切换为 HALT 模式(L)
- 当 CPU 工作于高速系统时钟(PLL 模式) (N)时切换为 HALT 模式(O)


状态转换	设置
(B) → (E)	执行 HALT 指令
(C) → (F)	
(D) → (G)	
(K) → (L)	
(N) → (O)	

备注 表 5-3 的(A)至(O)对应于图 5-17 的(A)至(O)。

表 5-3. CPU 时钟转换和 SFR 寄存器的设置示例(6/6)

(13) • 当 CPU 工作于高速片上振荡器时钟(B)时切换为 STOP 模式(H)

- 当 CPU 工作于高速系统时钟(C)时切换为 STOP 模式(I)

(设置序列) 

状态转换		设置	
(B) → (H)		停止无法工作于 STOP 模式的外围功能	–
(C) → (I)	在 X1 振荡时		设置 OSTS 寄存器
	外部主系统时钟		–

(14) CPU 从 STOP 模式(H)切换至 SNOOZE 模式(J)

关于从 STOP 模式切换至 SNOOZE 模式时，支持 SNOOZE 模式的各功能的设置，请参阅以下章节。

- A/D 转换器 → 12.8 SNOOZE 模式功能
- 串行阵列单元 0 的 CSI00 → 15.5.7 SNOOZE 模式功能 (仅限 CSI00)
- 串行阵列单元 0 的 UART0 → 15.6.3 SNOOZE 模式功能 (仅限 UART0 接收)
- 串行阵列单元 4 的 DALI/UART4 → 16.7 SNOOZE 模式功能 (仅限 DALI/UART4 接收)

(15) • 当 CPU 工作于高速片上振荡器时钟(PLL 模式) (K)时切换为 STOP 模式(H)

- 当 CPU 工作于高速系统时钟(PLL 模式) (N)时切换为 STOP 模式(I)

从 PLL 模式操作切换至高速片上振荡器时钟和高速系统时钟操作 (参阅 5.6.5 节的(11))，然后执行 STOP 指令。

注意事项 从 PLL 模式切换至高速片上振荡器时钟和高速系统时钟时，停止提供 PLL 输出时钟(f_{PLL})的功能(定时器 KB0 至 KB2、KC0 和比较器/可编程增益放大器)。

备注 表 5-3 的(A)至(O)对应于图 5-17 的(A)至(O)。

5.6.6 CPU时钟转换前的条件以及转换后的处理

CPU 时钟转换之前的条件以及 CPU 时钟转换之后的处理如下所示。

表 5-4. CPU 时钟的转换(1/2)

CPU 时钟		转换前的条件	转换后的处理
转换前	转换后		
高速片上振荡器时钟	X1 时钟	X1 振荡稳定 • OSCSEL = 1, EXCLK = 0, MSTOP = 0 • 经过了振荡稳定时间	通过停止高速片上振荡器(HIOSTOP = 1)可以减小操作电流。
	外部主系统时钟	使 EXCLK 引脚的外部时钟输入有效 • OSCSEL = 1, EXCLK = 1, MSTOP = 0	
	XT1 时钟	XT1 振荡稳定 • OSCSELS = 1, EXCLKS = 0, XTSTOP = 0 • 经过了振荡稳定时间	
	外部副系统时钟	使 EXCLKS 引脚的外部时钟输入有效 • OSCSELS = 1, EXCLKS = 1, XTSTOP = 0	
X1 时钟	高速片上振荡器时钟	高速片上振荡器振荡中 • HIOSTOP = 0 • 经过了振荡稳定时间	可以停止 X1 振荡 (MSTOP = 1)。
	外部主系统时钟	不可以转换 (如要转换时钟, 执行复位然后再次设置。)	-
	XT1 时钟	XT1 振荡稳定 • OSCSELS = 1, EXCLKS = 0, XTSTOP = 0 • 经过了振荡稳定时间	可以停止 X1 振荡 (MSTOP = 1)。
	外部副系统时钟	使 EXCLKS 引脚的外部时钟输入有效 • OSCSELS = 1, EXCLKS = 1, XTSTOP = 0	可以停止 X1 振荡 (MSTOP = 1)。
外部主系统时钟	高速片上振荡器时钟	高速片上振荡器振荡中 • HIOSTOP = 0 • 经过了振荡精度稳定时间	可以禁止外部主系统时钟输入(MSTOP = 1)。
	X1 时钟	不可以转换	-
	XT1 时钟	XT1 振荡稳定 • OSCSELS = 1, EXCLKS = 0, XTSTOP = 0 • 经过了振荡稳定时间	可以禁止外部主系统时钟输入(MSTOP = 1)。
	外部副系统时钟	使 EXCLKS 引脚的外部时钟输入有效 • OSCSELS = 1, EXCLKS = 1, XTSTOP = 0	可以禁止外部主系统时钟输入(MSTOP = 1)。

表 5-5. CPU 时钟的转换(2/2)

CPU 时钟		转换前的条件	转换后的处理
转换前	转换后		
XT1 时钟	高速片上振荡器时钟	高速片上振荡器振荡中并选择高速片上振荡器时钟作为主系统时钟 • HIOSTOP = 0, MCS = 0	可以停止 XT1 振荡 (XTSTOP = 1)。
	X1 时钟	X1 振荡稳定并选择高速系统时钟作为主系统时钟 • OSCSEL = 1, EXCLK = 0, MSTOP = 0 • 经过了振荡稳定时间 • MCS = 1	
	外部主系统时钟	使 EXCLK 引脚的外部时钟输入有效并选择高速系统时钟作为主系统时钟 • OSCSEL = 1, EXCLK = 1, MSTOP = 0 • MCS = 1	
	外部副系统时钟	不可以转换	-
外部副系统时钟	高速片上振荡器时钟	高速片上振荡器振荡中并选择高速片上振荡器时钟作为主系统时钟 • HIOSTOP = 0, MCS = 0	可以禁止外部副系统时钟输入(XTSTOP = 1)。
	X1 时钟	X1 振荡稳定并选择高速系统时钟作为主系统时钟 • OSCSEL = 1, EXCLK = 0, MSTOP = 0 • 经过了振荡稳定时间 • MCS = 1	
	外部主系统时钟	使 EXCLK 引脚的外部时钟输入有效并选择高速系统时钟作为主系统时钟 • OSCSEL = 1, EXCLK = 1, MSTOP = 0 • MCS = 1	
	XT1 时钟	不可以转换 (如要转换时钟, 执行复位然后再次设置。)	-

5.6.7 切换CPU时钟和切换系统时钟所需要的时间

通过设置系统时钟控制寄存器(CKC)的位 4 和位 6(MCM0、CSS)，可以切换 CPU 时钟（在主系统时钟与副系统时钟之间）以及切换主系统时钟（在高速片上振荡器时钟与高速系统时钟之间）。

但改写 CKC 寄存器之后，并不立即进行切换，将以切换前的时钟继续操作数个时钟周期 (参阅表 5-5 至表 5-7)。

通过 CKC 寄存器的位 7(CLS)来确认 CPU 是利用主系统时钟还是副系统时钟工作。通过 CKC 寄存器的位 5(MCS)来确认主系统时钟是工作于高速系统时钟还是高速片上振荡器时钟。

切换 CPU 时钟时，外围硬件时钟也同时被切换。

表 5-5. 切换系统时钟需要的最长时间

时钟 A	切换方向	时钟 B	备注
f _{IH}	↔	f _{MX}	参阅表 5-6
f _{MAIN}	↔	f _{SUB}	参阅表 5-7

表 5-6. f_{IH} ↔ f_{MX} 需要的最大时钟数

切换前的设置值		切换后的设置值	
MCM0		MCM0	
		0 (f _{MAIN} = f _{IH})	1 (f _{MAIN} = f _{MX})
0 (f _{MAIN} = f _{IH})	f _{MX} ≥ f _{IH}		2 个时钟
	f _{MX} < f _{IH}		2f _{IH} /f _{MX} 时钟
1 (f _{MAIN} = f _{MX})	f _{MX} ≥ f _{IH}	2f _{MX} /f _{IH} 时钟	
	f _{MX} < f _{IH}	2 个时钟	

表 5-7. f_{MAIN} ↔ f_{SUB} 需要的最大时钟数

切换前的设置值		切换后的设置值	
CSS		CSS	
		0 (f _{CLK} = f _{MAIN})	1 (f _{CLK} = f _{SUB})
0 (f _{CLK} = f _{MAIN})			1 + 2f _{MAIN} /f _{SUB} 时钟
1 (f _{CLK} = f _{SUB})		3 个时钟	

- 备注 1. 表 5-6 和表 5-7 中列出的时钟数是切换前的 CPU 时钟数。
- 2. 计算表 5-6 和表 5-7 中列出的时钟数时要去掉小数部分。

示例 当主系统时钟从高速系统时钟切换至高速片上振荡器时钟（@ 振荡频率 f_{IH} = 8 MHz, f_{MX} = 10 MHz）

$$2f_{MX}/f_{IH} = 2(10/8) = 2.5 \rightarrow 3 \text{ 个时钟}$$

5.6.8 停止时钟振荡前的条件

用于停止时钟振荡（禁止外部时钟输入）的寄存器标志设置以及停止时钟振荡之前的条件如下所示。

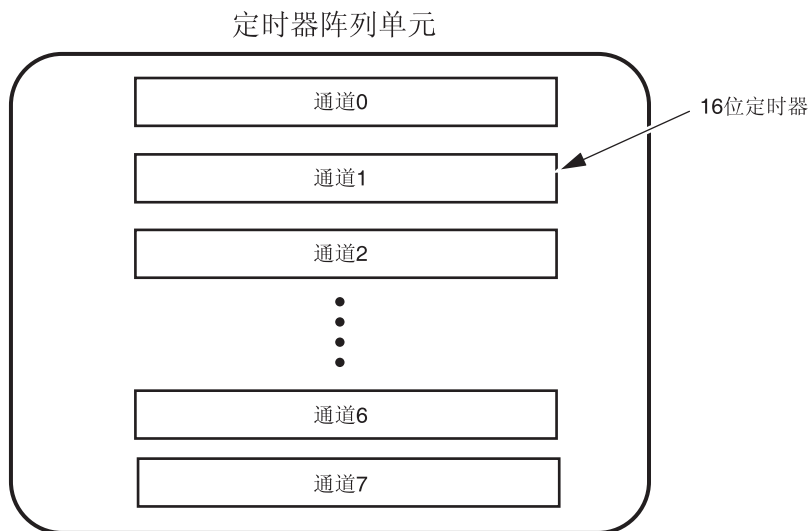
表 5-7. 停止时钟振荡前的条件和标志设置

时钟	停止时钟振荡前的条件 (禁止外部时钟输入)	SFR 寄存器的标志设置
高速片上振荡器时钟	MCS = 1 或 CLS = 1 (CPU 工作于高速片上振荡器时钟以外的时钟。)	HIOSTOP = 1
X1 时钟	MCS = 0 或 CLS = 1 (CPU 工作于高速系统时钟以外的时钟。)	MSTOP = 1
外部主系统时钟		
XT1 时钟	CLS = 0 (CPU 工作于副系统时钟以外的时钟。)	XTSTOP = 1
外部副系统时钟		

第六章 定时器阵列单元

定时器阵列单元具有 8 个 16 位定时器。

每个 16 位定时器被称为一个通道，可用作独立的定时器。另外，两个或更多“通道”可用来构建一个高精度的定时器。



关于每个功能的详情，请参阅下表。

单通道操作功能	多通道联动操作功能
<ul style="list-style-type: none"> • 间隔定时器 (→ 参阅 6.7.1) • 方波输出 (→ 参阅 6.7.1) • 外部事件计数器 (→ 参阅 6.7.2) • 输入脉冲间隔测量 (→ 参阅 6.7.3) • 输入信号的高/低电平宽度的测量 (→ 参阅 6.7.4) • 延迟计数器 (→ 参阅 6.7.5) 	<ul style="list-style-type: none"> • 单触发脉冲输出(→ 参阅 6.8.1) • PWM 输出(→ 参阅 6.8.2) • 多重 PWM 输出(→ 参阅 6.8.3)

可以将通道 1 和通道 3 的 16 位定时器用作两个 8 位定时器（高位和低位）。将通道 1 和通道 3 用作 8 位定时器时可用的功能如下：

- 间隔定时器/方波输出
- 外部事件计数器(仅限低 8 位定时器)
- 延迟计数器(仅限低 8 位定时器)

将通道 7 与串行阵列单元 0 的 UART0 联合使用的话，可以实现 LIN-bus 和 DMX512 接收处理。

6.1 定时器阵列单元功能

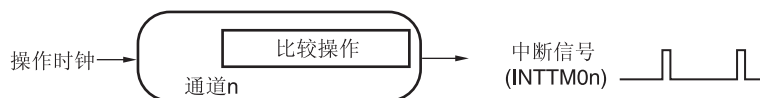
定时器阵列单元具有以下功能：

6.1.1 单通道操作功能

通过单独操作一个通道，从而不受其他通道操作模式影响地达到以下目的。

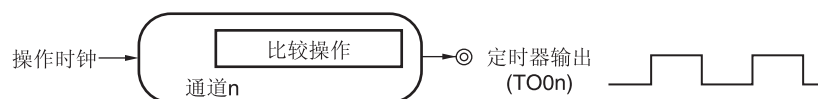
(1) 间隔定时器

单元中的各个定时器都可用作按照固定间隔产生中断(INTTM0n)的基准定时器。



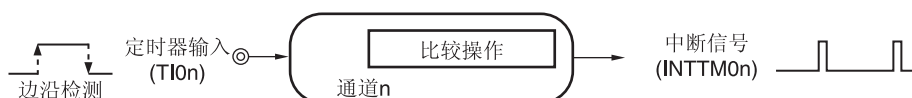
(2) 方波输出

每当产生 INTTM0n 中断时执行交替操作，并从定时器输出引脚(TO0n)输出占空比 50%的方波。



(3) 外部事件计数器

单元中的各个定时器可以用作事件计数器，当输入至定时器输入引脚(TI0n)的信号的有效边沿数达到指定值时，产生一个中断。



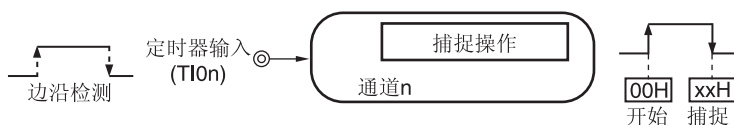
(4) 输入脉冲间隔测量

从输入至定时器引脚(TI0n)的脉冲信号的有效边沿开始计数。并且在下一个脉冲的有效边沿捕捉定时器的计数值。从而测量输入脉冲的间隔。



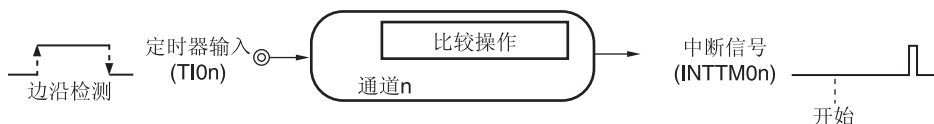
(5) 输入信号的高/低电平宽度的测量

从输入至定时器输入引脚(TI0n)的信号的一个边沿开始计数，在另一个边沿捕捉计数值。从而测量输入信号的高电平或低电平宽度。



(6) 延迟计数器

从输入至定时器输入引脚(TI0n)的信号的有效边沿开始计数，并在经过任意延迟期间后产生中断。



备注 1 n: 通道编号 (n = 0 至 7)

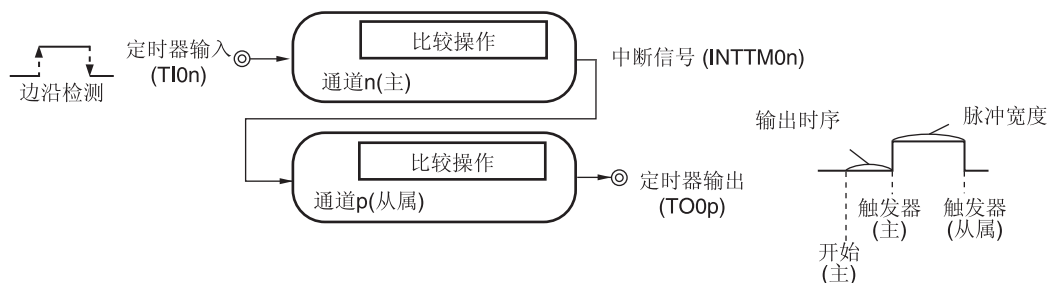
2. 通道 0 至通道 7 的定时器输入/输出引脚的存在与否因产品而异。有关详情，请参阅表 6-2 每种产品中配置的定时器输入/输出引脚。

6.1.2 多通道联动操作功能

通过联合使用主通道（主要用于控制周期的基准定时器）和从属通道（以主通道为基准而工作的定时器），来实现以下功能。

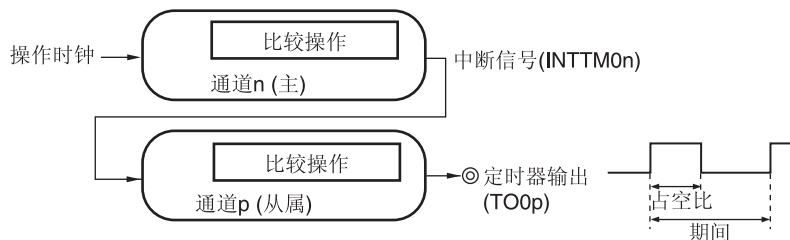
(1) 单触发脉冲输出

将两个通道成组使用，生成可任意设置输出时序和脉冲宽度的单触发脉冲。



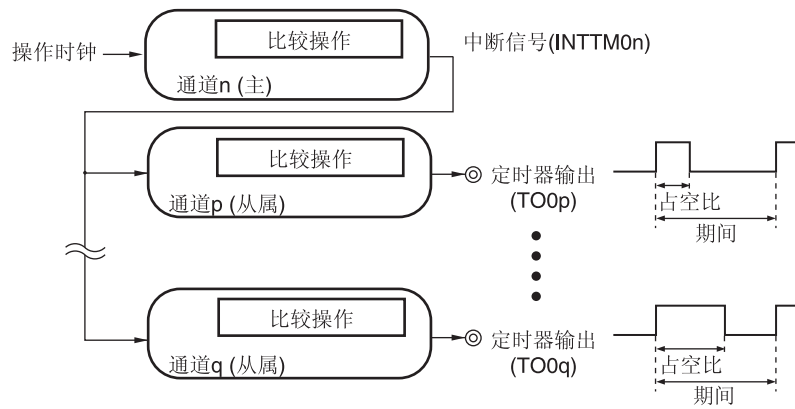
(2) PWM（脉冲宽度调制）输出

将两个通道成组使用，生成可任意设置周期和占空比的脉冲。



(3) 多重 PWM (脉冲宽度调制) 输出

通过扩展 PWM 功能，并利用一个主通道和两个或更多的从属通道，从而生成最多七种具有指定周期且指定占空比的 PWM 信号。



注意事项 关于多通道联动操作功能规则的详情，请参阅 6.4.1 多通道联动操作功能的基本规则。

6.1.3 支持LIN-bus功能（仅限通道 7）

定时器阵列单元用于检查 LIN-bus 通信中接收的信号与 LIN-bus 通信格式是否匹配。

(1) 唤醒信号的检测

从 UART0 的串行数据输入引脚(RxD0)输入的信号的下降沿定时器开始计数，并且在上升沿捕捉定时器的计数值。从而测量输入脉冲的间隔。如果低电平宽度大于或等于指定值，则被识别为唤醒信号。

(2) 间隔段的检测

检测到唤醒信号之后，从 UART0 的串行数据输入引脚(RxD0)输入的信号的下降沿定时器开始计数，并且在上升沿捕捉定时器的计数值。从而测量低电平宽度。如果低电平宽度大于或等于指定值，则被识别为间隔段。

(3) 同步段脉冲宽度的测量

检测到同步间隔段之后，测量从 UART0 的串行数据输入引脚(RxD0)输入的信号的低电平宽度和高电平宽度。根据以这种方式测得的同步段的位间隔，计算出波特率。

备注 关于支持 LIN-bus 通信功能的操作设置的详情，请参阅 **6.3 (13) 输入切换控制寄存器(ISC)**和 **6.7.4 输入信号高/低电平宽度测量的操作**。

6.1.4 支持DMX512 功能（仅限通道 7）

定时器阵列单元用于检查 DMX512 通信中接收的信号与 DMX512 通信格式是否匹配。

(1) BREAK 信号的检测

从 UART0 的串行数据输入引脚(RxD0)输入的信号的下降沿定时器开始计数，并且在上升沿捕捉定时器的计数值。从而测量输入脉冲的间隔。如果低电平宽度大于或等于指定值，则被识别为 BREAK 信号。

(2) 信号宽度的检测

测量输入至 UART0 的串行数据输入引脚(RxD0)的信号的低电平宽度和高电平宽度。

备注 关于兼容 DMX512 通信功能的操作设置的详情，请参阅 **6.3 (13) 输入切换控制寄存器(ISC)**和 **6.7.4 输入信号高/低电平宽度测量的操作**。

6.2 定时器阵列单元的配置

定时器阵列单元包含以下硬件。

表 6-1. 定时器阵列单元的配置

项目	配置
定时器/计数器	定时器/计数器寄存器 0n (TCR0n)
寄存器	定时器数据寄存器 0n (TDR0n)
定时器输入	TI03, TI05, TI06 引脚 ^{※1} , RxD0 引脚(用于 LIN-bus, DMX512)
定时器输出	TO03, TO05, TO06 引脚 ^{※1} , 输出控制电路
控制寄存器	<单元设置块的寄存器> <ul style="list-style-type: none"> • 外围允许寄存器 0 (PER0) • 定时器时钟选择寄存器 0 (TPS0) • 定时器通道允许状态寄存器 0 (TE0) • 定时器通道开始寄存器 0 (TS0) • 定时器通道停止寄存器 0 (TT0) • 定时器输入选择寄存器 0 (TIS0) • 定时器输出允许寄存器 0 (TOE0) • 定时器输出寄存器 0 (TO0) • 定时器输出电平寄存器 0 (TOL0) • 定时器输出模式寄存器 0 (TOM0)
	<每个通道的寄存器> <ul style="list-style-type: none"> • 定时器模式寄存器 0n (TMR0n) • 定时器状态寄存器 0n (TSR0n) • 输入切换控制寄存器 (ISC) • 噪声滤波器允许寄存器 1 (NFEN1) • 端口模式寄存器 (PM0, PM3)^{※2} • 端口寄存器 (P0, P3)^{※2}

注 1. 通道 0 至通道 7 的定时器输入/输出引脚的存在与否因产品而异。有关详情，请参阅表 6-2 每种产品中配置的定时器输入/输出引脚。

2. 端口模式寄存器 (PM0, PM3) 和端口寄存器 (P0, P3) 的设置因产品而异。详情请参阅 6.3 (15) 端口模式寄存器 0, 3 (PM0, PM3)。

备注 n: 通道编号 (n = 0 至 7)

各定时器阵列单元通道的定时器输入/输出引脚的存在与否因产品而异。

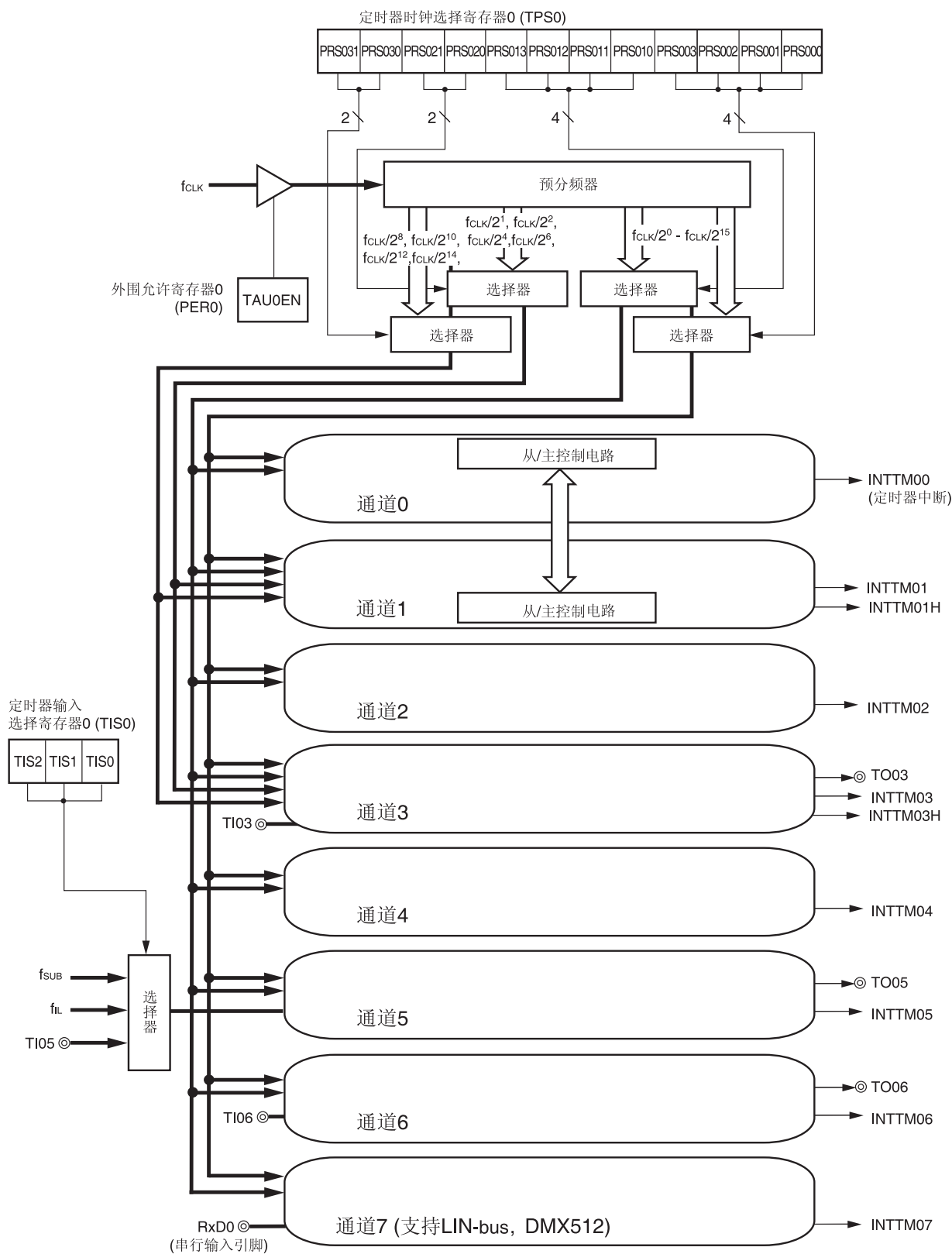
表 6-2. 每种产品中配置的定时器输入/输出引脚

定时器阵列单元通道		每种产品的输入/输出引脚			
		38引脚	32引脚	30引脚	20引脚
单元0	通道0	-	-	-	-
	通道1	-	-	-	-
	通道2	-	-	-	-
	通道3	P31/TI03/TO03		P31/TI03/TO03	-
	通道4	-	-	-	-
	通道5	P05/TI05/TO05		-	-
	通道6	P06/TI06/TO06		-	-
	通道7	-			

- 备注 1.** 由于定时器输入和定时器输出共用同一引脚，仅可以使用定时器输入或定时器输出。
- 2.** -: 没有定时器输入/输出引脚，但通道有效。(然而，通道仅能用作间隔定时器。)

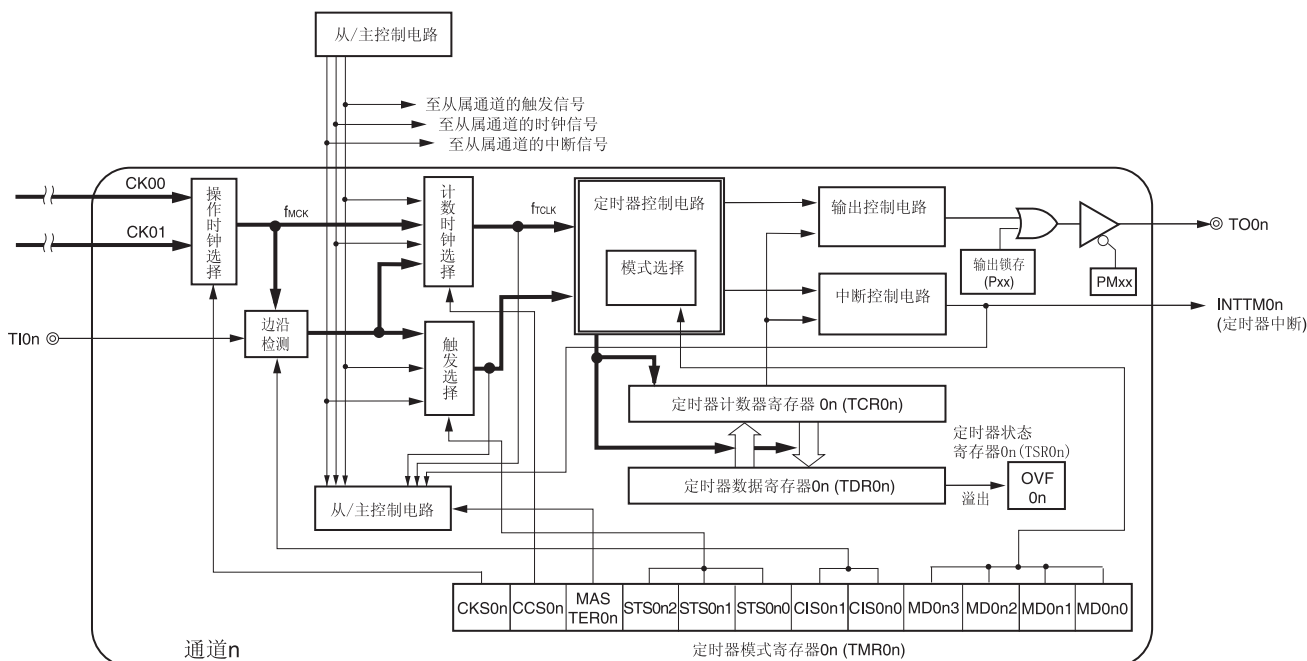
定时器阵列单元的框图如图 6-1 和图 6-2 所示。

图 6-1. 定时器阵列单元的整体框图 (示例: 38 引脚产品)



备注 f_{SUB}: 副系统时钟频率
 f_{IL}: 内部低速片上振荡器时钟频率

图 6-2. 定时器阵列单元的通道内部框图



备注 n = 3, 6

(1) 定时器计数寄存器 0n (TCR0n)

TCR0n 寄存器是 16 位只读寄存器，用于对计数时钟进行计数。

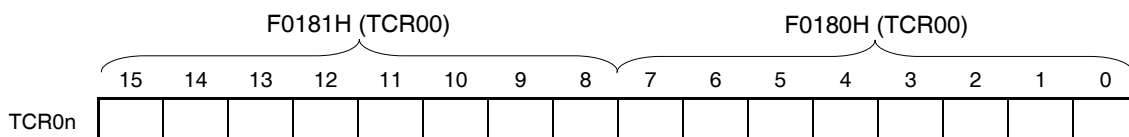
与计数时钟的上升沿同步，进行递增或递减计数。

通过定时器模式寄存器 0n (TMR0n) 的 MD0n3 位至 MD0n0 位来选择操作模式，从而切换至递增或递减计数（参阅

6.3 (3) 定时器模式寄存器 0n (TMR0n)）。

图 6-3. 定时器计数寄存器 0n (TCR0n) 的格式

地址: F0180H, F0181H (TCR00)至 F018EH, F018FH (TCR07) 复位后: FFFFH R



备注 n: 通道编号 (n = 0 至 7)

可以通过读取定时器计数寄存器 0n (TCR0n)来读取计数值。

下列情况中计数值被设置为 FFFFH。

- 产生复位信号后
- 外围允许寄存器 0(PER0)的 TAU0EN 位被清除时
- 在 PWM 输出模式下完成从属通道计数时
- 在延迟计数模式下完成从属通道计数时
- 在单触发脉冲输出模式下完成主/从属通道计数时
- 在多重 PWM 输出模式下完成从属通道计数时

下列情况中计数值被清除为 0000H。

- 在捕捉模式下输入开始触发时
- 在捕捉模式下完成捕捉时

注意事项 即使在 TCR0n 寄存器被读取时，也不会将计数值捕捉至定时器数据寄存器 0n (TDR0n)。

如下所示，TCR0n 寄存器读取值因操作模式和操作状态而异。

表 6-3. 各种操作模式下定时器/计数器寄存器 0n (TCR0n)的读取值

操作模式	计数模式	定时器/计数器寄存器0n (TCR0n)的读取值 ^注			
		解除复位后操作模式发生变化时的值	暂停计数操作(TT0n = 1)时的值	暂停计数操作后(TT0n = 1)操作模式发生变化时的值	一次计数后等待开始触发时的值
间隔定时器模式	递减计数	FFFFH	停止值	不定	-
捕捉模式	递增计数	0000H	停止值	不定	-
事件计数器模式	递减计数	FFFFH	停止值	不定	-
单计数模式	递减计数	FFFFH	停止值	不定	FFFFH
捕捉&单计数模式	递增计数	0000H	停止值	不定	TDR0n寄存器的捕捉值 + 1

注 表示当通道 n 为定时器操作停止状态(TE0n = 0)且计数操作刚被允许(TS0n = 1)时 TCR0n 寄存器的读取值。直到开始计数操作为止，该值被保持在 TCR0n 寄存器中。

备注 n: 通道编号 (n = 0 至 7)

(2) 定时器数据寄存器 0n (TDR0n)

可以在捕捉功能和比较功能中使用的 16 位寄存器。

利用定时器模式寄存器 0n (TMR0n) 的 MD0n3 位至 MD0n0 位来选择操作模式，从而切换至捕捉功能或比较功能。

TDR0n 寄存器的值可以随时更改。

可以以 16 位为单位读取或写入。

另外，对于 TDR01 和 TDR03 寄存器，当处于 8 位定时器模式时（定时器模式寄存器 01 和 03 (TMR01、TMR03) 的 SPLIT 位为 1 时），可以以按 8 位为单位改写数据，其中，TDR01H 和 TDR03H 用作高 8 位，TDR01L 和 TDR03L 用作低 8 位。然而，只能以 16 位为单位读取。

产生复位信号后，该寄存器被清除为 0000H。

图 6-4. 定时器数据寄存器 0n (TDR0n)的格式 (n = 0, 2, 4 至 7)

地址: FFF18H, FFF19H (TDR00), FFF64H, FFF65H (TDR02), 复位后: 0000H R/W
FFF68H, FFF69H (TDR04)至 FFF6EH, FFF6FH (TDR07)

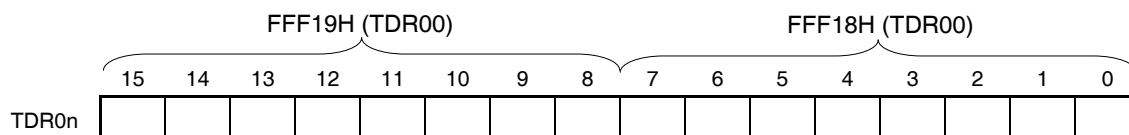
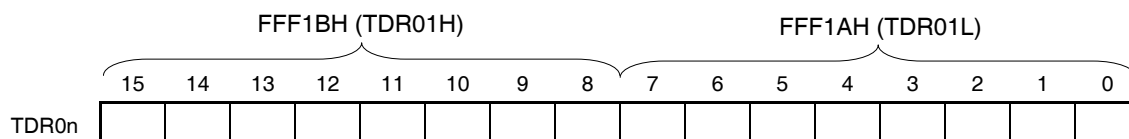


图 6-5. 定时器数据寄存器 0n (TDR0n)的格式 (n = 1, 3)

地址: FFF1AH, FFF1BH (TDR01), FFF66H, FFF67H (TDR03) 复位后: 0000H R/W

**(i) 使用定时器数据寄存器 0n (TDR0n)作为比较寄存器时**

从设置到 TDR0n 寄存器的值开始递减计数。当计数值达到 0000H 时，产生中断信号(INTTM0n)。直到改写为止 TDR0n 寄存器将保持其值。

注意事项 当设置为比较功能时，即使输入捕捉触发，TDR0n 寄存器也不会执行捕捉操作。

(ii) 使用定时器数据寄存器 0n (TDR0n)作为捕捉寄存器时

输入捕捉触发时，定时器/计数寄存器 0n (TCR0n)的计数值将被捕捉至 TDR0n 寄存器。
可以通过定时器模式寄存器 0n (TMR0n)来选择 TIO0n 引脚的有效边沿作为捕捉触发信号。

备注 n: 通道编号 (n = 0 至 7)

6.3 控制定时器阵列单元的寄存器

下列寄存器控制着定时器阵列单元。

- 外围允许寄存器 0 (PER0)
- 定时器时钟选择寄存器 0 (TPS0)
- 定时器模式寄存器 0n (TMR0n)
- 定时器状态寄存器 0n (TSR0n)
- 定时器通道允许状态寄存器 0 (TE0)
- 定时器通道开始寄存器 0 (TS0)
- 定时器通道停止寄存器 0 (TT0)
- 定时器输入选择寄存器 0 (TIS0)
- 定时器输出允许寄存器 0 (TOE0)
- 定时器输出寄存器 0 (TO0)
- 定时器输出电平寄存器 0 (TOL0)
- 定时器输出模式寄存器 0 (TOM0)
- 输入切换控制寄存器 (ISC)
- 噪声滤波器允许寄存器 1 (NFEN1)
- 端口模式寄存器 (PM0, PM03)^注
- 端口寄存器 (P0, P3)^注

注 端口模式寄存器 (PM0, PM3) 和端口寄存器 (P0, P3) 的设置因产品而异。详情请参阅 **6.3 (15) 端口模式寄存器 0, 3 (PM0, PM3)**。

备注 n: 通道编号 (n = 0 至 7)

(1) 外围允许寄存器 0 (PER0)

该寄存器用于允许或禁止向外围硬件供应时钟。通过停止向未使用的硬件供应时钟，可以降低功耗和噪声。

使用定时器阵列单元时，必须将该寄存器的位 0(TAU0EN)设置为 1。

使用 1 位或 8 位存储器操作指令设置 PER0 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 6-6. 外围允许寄存器 0 (PER0)的格式

地址: F00F0H 复位后: 00H R/W

符号	<7>	6	<5>	<4>	3	<2>	1	<0>
PER0	RTCEN	0	ADCEN	IICA0EN	0	SAU0EN	0	TAU0EN

TAU0EN	定时器阵列单元输入时钟的控制
0	停止输入时钟的供应。 <ul style="list-style-type: none"> • 不可以写入用于定时器阵列单元的SFR。 • 定时器阵列单元处于复位状态。
1	允许输入时钟的供应。 <ul style="list-style-type: none"> • 可以读取/写入用于定时器阵列单元的SFR。

注意事项 1. 设置定时器阵列单元时，必须先将 TAU0EN 位设置为 1。如果 TAU0EN = 0，向定时器阵列单元的控制寄存器进行的写入操作将被忽略，而且，所有读取值均为初始值（定时器输入选择寄存器 0(TIS0)、输入切换控制寄存器(ISC)、噪声滤波器允许寄存器 1 (NFEN1)、端口模式控制寄存器 0, 3 (PMC0, PMC3)、端口模式寄存器 0, 3 (PM0, PM3)和端口寄存器 0, 3 (P0, P3)除外）。

2. 必须将位 1、3 和 6 清除为“0”。

(2) 定时器时钟选择寄存器 0 (TPS0)

TPS0 寄存器为 16 位寄存器，用于选择由外部预分频器供应至各通道共通的两种或四种操作时钟(CK00、CK01)。通过 TPS0 寄存器的位 7 至位 4 选择 CK01，通过位 3 至位 0 选择 CK00。另外，通道 1 和通道 3 中，通过 TPS0 寄存器的位 9 和位 8 选择 CK02，通过位 13 和位 12 选择 CK03。

只有在以下情况下才可以在定时器操作期间改写 TPS0 寄存器。

如果需要改写 PRS000 至 PRS003 位($n = 0$ 至 7):

选择 CK00 作为操作时钟($CKS0n1, CKS0n0 = 0, 0$)的所有通道停止($TE0n = 0$)。

如果需要改写 PRS010 至 PRS013 位($n = 0$ 至 7):

选择 CK01 作为操作时钟($CKS0n1, CKS0n0 = 0, 1$)的所有通道停止($TE0n = 0$)。

如果需要改写 PRS020 和 PRS021 位($n = 1、3$):

选择 CK02 作为操作时钟($CKS0n1, CKS0n0 = 1, 0$)的所有通道停止($TE0n = 0$)。

如果需要改写 PRS030 和 PRS031 位($n = 1、3$):

选择 CK03 作为操作时钟($CKS0n1, CKS0n0 = 1, 1$)的所有通道停止($TE0n = 0$)。

使用 16 位存储器操作指令设置 TPS0 寄存器。

产生复位信号后，该寄存器被清除为 0000H。

图 6-7. 定时器时钟选择寄存器 0 (TPS0)的格式 (1/2)

地址: F01B6H, F01B7H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TPS0	0	0	PRS 031	PRS 030	0	0	PRS 021	PRS 020	PRS 013	PRS 012	PRS 011	PRS 010	PRS 003	PRS 002	PRS 001	PRS 000

PRS 0k3	PRS 0k2	PRS 0k1	PRS 0k0		操作时钟(CK0k)的选择 [※] (k = 0, 1)				
					f _{CLK} = 2 MHz	f _{CLK} = 5 MHz	f _{CLK} = 10 MHz	f _{CLK} = 20 MHz	f _{CLK} = 32 MHz
0	0	0	0	f _{CLK}	2 MHz	5 MHz	10 MHz	20 MHz	32 MHz
0	0	0	1	f _{CLK} /2	1 MHz	2.5 MHz	5 MHz	10 MHz	16 MHz
0	0	1	0	f _{CLK} /2 ²	500 kHz	1.25 MHz	2.5 MHz	5 MHz	8 MHz
0	0	1	1	f _{CLK} /2 ³	250 kHz	625 kHz	1.25 MHz	2.5 MHz	4 MHz
0	1	0	0	f _{CLK} /2 ⁴	125 kHz	312.5 kHz	625 kHz	1.25 MHz	2 MHz
0	1	0	1	f _{CLK} /2 ⁵	62.5 kHz	156.2 kHz	312.5 kHz	625 kHz	1 MHz
0	1	1	0	f _{CLK} /2 ⁶	31.25 kHz	78.1 kHz	156.2 kHz	312.5 kHz	500 kHz
0	1	1	1	f _{CLK} /2 ⁷	15.62 kHz	39.1 kHz	78.1 kHz	156.2 kHz	250 kHz
1	0	0	0	f _{CLK} /2 ⁸	7.81 kHz	19.5 kHz	39.1 kHz	78.1 kHz	125 kHz
1	0	0	1	f _{CLK} /2 ⁹	3.91 kHz	9.76 kHz	19.5 kHz	39.1 kHz	62.5 kHz
1	0	1	0	f _{CLK} /2 ¹⁰	1.95 kHz	4.88 kHz	9.76 kHz	19.5 kHz	31.25 kHz
1	0	1	1	f _{CLK} /2 ¹¹	976 Hz	2.44 kHz	4.88 kHz	9.76 kHz	15.63 kHz
1	1	0	0	f _{CLK} /2 ¹²	488 Hz	1.22 kHz	2.44 kHz	4.88 kHz	7.81 kHz
1	1	0	1	f _{CLK} /2 ¹³	244 Hz	610 Hz	1.22 kHz	2.44 kHz	3.91 kHz
1	1	1	0	f _{CLK} /2 ¹⁴	122 Hz	305 Hz	610 Hz	1.22 kHz	1.95 kHz
1	1	1	1	f _{CLK} /2 ¹⁵	61 Hz	153 Hz	305 Hz	610 Hz	976 Hz

注 要更改被选作 f_{CLK} 的时钟（通过更改系统时钟控制寄存器(CKC)的值）时，须停止定时器阵列单元 (TT0 = 00FFH)。

- 注意事项 1. 必须将位 15、14、11、10 清除为 0。
2. 如果 f_{CLK} (没有分频)被选择作为操作时钟 (CKmk)，且 TDRmn 被设置为 0000H (m = 0 或 1, n = 0 至 7)，则不能检测出来自定时器阵列单元的中断请求输出。

- 备注 1. f_{CLK}: CPU/外围硬件时钟频率
2. 通过 TPSm 寄存器选择的时钟波形从上升沿开始的 f_{CLK} 的仅 1 个周期为高电平(m = 1 至 15)。详情请参阅 6.5.1 计数时钟(f_{TCLK})。

图 6-7. 定时器时钟选择寄存器 0 (TPS0)的格式 (2/2)

地址: F01B6H, F01B7H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TPS0	0	0	PRS 031	PRS 030	0	0	PRS 021	PRS 020	PRS 013	PRS 012	PRS 011	PRS 010	PRS 003	PRS 002	PRS 001	PRS 000

PRS 021	PRS 020	操作时钟(CK02)的选择 ^注					
			f _{CLK} = 2 MHz	f _{CLK} = 5 MHz	f _{CLK} = 10 MHz	f _{CLK} = 20 MHz	f _{CLK} = 32 MHz
0	0	f _{CLK} /2	1 MHz	2.5 MHz	5 MHz	10 MHz	16 MHz
0	1	f _{CLK} /2 ²	500 kHz	1.25 MHz	2.5 MHz	5 MHz	8 MHz
1	0	f _{CLK} /2 ⁴	125 kHz	312.5 kHz	625 MHz	1.25 MHz	2 MHz
1	1	f _{CLK} /2 ⁶	31.25 kHz	78.1 kHz	156.2 kHz	312.5 kHz	500 kHz

PRS 031	PRS 030	操作时钟(CK03)的选择 ^注					
			f _{CLK} = 2 MHz	f _{CLK} = 5 MHz	f _{CLK} = 10 MHz	f _{CLK} = 20 MHz	f _{CLK} = 32 MHz
0	0	f _{CLK} /2 ⁸	7.81 kHz	19.5 kHz	39.1 kHz	78.1 kHz	125 kHz
0	1	f _{CLK} /2 ¹⁰	1.95 kHz	4.88 kHz	9.76 kHz	19.5 kHz	31.25 kHz
1	0	f _{CLK} /2 ¹²	488 Hz	1.22 kHz	2.44 kHz	4.88 kHz	7.81 kHz
1	1	f _{CLK} /2 ¹⁴	122 Hz	305 Hz	610 Hz	1.22 kHz	1.95 kHz

注 要更改被选作 f_{CLK} 的时钟（通过更改系统时钟控制寄存器(CKC)的值）时，须停止定时器阵列单元 (TT0 = 00FFH)。

在通过 CKS0n0 和 CKS0n1 位指定的操作时钟(f_{mck})和 TIO_n 引脚输入信号的有效边沿的两者中选择一个作为计数时钟(f_{rclk})时，也须停止定时器阵列单元。

注意事项 必须将位 15、14、11、10 清除为 0。

备注 f_{CLK}: CPU/外围硬件时钟频率

通过将通道 1 和通道 3 用作 8 位定时器模式，并将 CK02 或 CK03 指定为操作时钟，可以利用间隔定时器功能实现如表 6-4 所示间隔时间。

表 6-4. 对操作时钟 CKS02 或者 CKS03 可用的间隔时间

时钟		间隔时间(f _{CLK} = 32 MHz)			
		10 μs ^注	100 μs ^注	1 ms ^注	10 ms ^注
CK02	f _{CLK} /2	√	–	–	–
	f _{CLK} /2 ²	√	–	–	–
	f _{CLK} /2 ⁴	√	√	–	–
	f _{CLK} /2 ⁶	√	√	–	–
CK03	f _{CLK} /2 ⁸	–	√	√	–
	f _{CLK} /2 ¹⁰	–	√	√	–
	f _{CLK} /2 ¹²	–	–	√	√
	f _{CLK} /2 ¹⁴	–	–	√	√

注 包含 5% 以内的误差。

备注 1. f_{CLK}: CPU/外围硬件时钟频率

2. 关于通过 TPS0 寄存器选择的信号 f_{CLK}/2^l 的详情，请参阅 6.5.1 计数时钟(f_{rclk})。

(3) 定时器模式寄存器 0n (TMR0n)

TMR0n 寄存器用于设置通道 n 的操作模式。该寄存器用于选择操作时钟(f_{MCK})，选择计数时钟，选择主/从属，选择 16 位或 8 位定时器（仅限通道 1 和通道 3），设置开始触发和捕捉触发，选择计数器输入的有效边沿，以及设置操作模式（间隔、捕捉、事件计数器、单计数、或者捕捉&单计数）。

当定时器通道运行（当 TE0n = 1）时，禁止改写 TMR0n 寄存器。然而，在定时器通道的某些功能运行(TE0n = 1) 时，位 7 和位 6(CIS0n1, CIS0n0)是可以改写的（详情请参阅 6.7 定时器阵列单元的单通道操作功能和 6.8 定时器阵列单元的多通道联动操作功能）。

使用 16 位存储器操作指令设置 TMR0n 寄存器。

产生复位信号后，该寄存器被清除为 0000H。

注意事项 TMR0n 寄存器的位 11 因通道而异。

TMR02, TMR04, TMR06: MASTER0n 位(n = 2, 4, 6)

TMR01, TMR03: SPLIT0n 位(n = 1, 3)

TMR00, TMR05, TMR07: 固定为 0

图 6-8. 定时器模式寄存器 0n (TMR0n)的格式 (1/4)

地址: F0190H, F0191H (TMR00) 至 F019EH, F019FH (TMR07) 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMR0n (n = 2, 4, 6)	CKS 0n1	CKS 0n0	0	CCS 0n	MAST ER0n	STS 0n2	STS 0n1	STS 0n0	CIS 0n1	CIS 0n0	0	0	MD 0n3	MD 0n2	MD 0n1	MD 0n0

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMR0n (n = 1, 3)	CKS 0n1	CKS 0n0	0	CCS 0n	SPLIT 0n	STS 0n2	STS 0n1	STS 0n0	CIS 0n1	CIS 0n0	0	0	MD 0n3	MD 0n2	MD 0n1	MD 0n0

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMR0n (n = 0, 5, 7)	CKS 0n1	CKS 0n0	0	CCS 0n	0 [#]	STS 0n2	STS 0n1	STS 0n0	CIS 0n1	CIS 0n0	0	0	MD 0n3	MD 0n2	MD 0n1	MD 0n0

CKS 0n1	CKS 0n0	通道n操作时钟(f _{mck})的选择
0	0	定时器时钟选择寄存器m (TP0m)设置的操作时钟CK00
0	1	定时器时钟选择寄存器0 (TPS0)设置的操作时钟CK02
1	0	定时器时钟选择寄存器0 (TPS0)设置的操作时钟CK01
1	1	定时器时钟选择寄存器0 (TPS0)设置的操作时钟CK03
边沿检测电路所使用的操作时钟(f _{mck})。通过设置CCS0n位来产生计数时钟(f _{clk})和采样时钟。 仅限通道1和通道3可以选择操作时钟CK02和CK03。		

CCS 0n	通道n计数时钟(f _{clk})的选择
0	由CKS0n0和CKS0n1位指定的操作时钟(f _{mck})
1	TIO _n 引脚的输入信号的有效边沿 通道5中, 由TIS0选择的输入信号的有效边沿
计数时钟(f _{clk})用于定时器/计数器, 输出控制电路和中断控制电路。	

注 位 11 为只读位, 固定为 0, 写入无效。

- 注意事项 1. 必须将位 13、5 和 4 清除为“0”。
2. 要更改被选作 f_{clk} 的时钟 (通过更改系统时钟控制寄存器(CKC)的值) 时, 无论是在通过 CKS0n0 和 CKS0n1 位指定的操作时钟(f_{mck})和 TIO_n 引脚输入信号的有效边沿的两者中选择哪个作为计数时钟 (f_{clk})时, 都须停止定时器阵列单元(TT0 = 00FFH)。

备注 n: 通道编号 (n = 0 至 7)

图 6-8. 定时器模式寄存器 0n (TMR0n)的格式 (2/4)

地址: F0190H, F0191H (TMR00) 至 F019EH, F019FH (TMR07) 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMR0n (n = 2, 4, 6)	CKS 0n1	CKS 0n0	0	CCS 0n	MAST ER0n	STS 0n2	STS 0n1	STS 0n0	CIS 0n1	CIS 0n0	0	0	MD 0n3	MD 0n2	MD 0n1	MD 0n0
符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMR0n (n = 1, 3)	CKS 0n1	CKS 0n0	0	CCS 0n	SPLIT 0n	STS 0n2	STS 0n1	STS 0n0	CIS 0n1	CIS 0n0	0	0	MD 0n3	MD 0n2	MD 0n1	MD 0n0
符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMR0n (n = 0, 5, 7)	CKS 0n1	CKS 0n0	0	CCS 0n	0 [#]	STS 0n2	STS 0n1	STS 0n0	CIS 0n1	CIS 0n0	0	0	MD 0n3	MD 0n2	MD 0n1	MD 0n0

(TMR0n 的位 11 (n = 2, 4, 6))

MAS TER 0n	选择对通道n进行单独操作或 与另一个通道(作为从属或主)一起进行联动操作
0	单通道操作功能, 或者作为从属通道的联动通道操作功能。
1	作为主通道的联动通道操作功能。
仅可以设置通道2, 4, 6为主通道(MASTER0n = 1)。 使用通道0, 5, 7时必须将其固定为0 (由于通道0为最高位通道, 因此不论该位设置如何, 其始终作为主通道操作)。 通道用于单通道操作功能时, 将MASTER0n位清除为0。	

(TMR0n 的位 11 (n = 1, 3))

SPLI T0n	用于通道1和3的8位或者16位定时器操作的选择
0	作为16位定时器动作。 (单通道操作功能, 或者作为从属通道的联动通道操作功能。)
1	作为8位定时器动作。

STS 0n2	STS 0n1	STS 0n0	通道n的开始触发或者捕捉触发的设置
0	0	0	仅限软件触发开始有效 (其他触发源不可选)。
0	0	1	TI0n引脚输入的有效边沿被同时用作开始触发和捕捉触发。
0	1	0	TI0n引脚的两个边沿分别被用作开始触发和捕捉触发。
1	0	0	使用主通道的中断信号 (当该通道用作联动通道操作功能时的从属通道)。
其他			禁止设置

注 位 11 为只读位, 固定为 0, 写入无效。

备注 n: 通道编号 (n = 0 至 7)

图 6-8. 定时器模式寄存器 0n (TMR0n)的格式 (3/4)

地址: F0190H, F0191H (TMR00) 至 F019EH, F019FH (TMR07) 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMR0n (n = 2, 4, 6)	CKS 0n1	CKS 0n0	0	CCS 0n	MAST ER0n	STS 0n2	STS 0n1	STS 0n0	CIS 0n1	CIS 0n0	0	0	MD 0n3	MD 0n2	MD 0n1	MD 0n0

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMR0n (n = 1, 3)	CKS 0n1	CKS 0n0	0	CCS 0n	SPLIT 0n	STS 0n2	STS 0n1	STS 0n0	CIS 0n1	CIS 0n0	0	0	MD 0n3	MD 0n2	MD 0n1	MD 0n0

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMR0n (n = 0, 5, 7)	CKS 0n1	CKS 0n0	0	CCS 0n	0	STS 0n2	STS 0n1	STS 0n0	CIS 0n1	CIS 0n0	0	0	MD 0n3	MD 0n2	MD 0n1	MD 0n0

CIS 0n1	CIS 0n0	TIO _n 引脚输入有效边沿的选择													
0	0	下降沿													
0	1	上升沿													
1	0	双边沿(测量低电平宽度时) 开始触发: 下降沿, 捕捉触发: 上升沿													
1	1	双边沿(测量高电平宽度时) 开始触发: 上升沿, 捕捉触发: 下降沿													
当STS _{0n2} 至STS _{0n0} 位的值不为010B且指定双边沿时, 请将 CIS _{0n1} 至CIS _{0n0} 位设置为10B。															

MD 0n3	MD 0n2	MD 0n1	通道n的操作模式	对应功能	TCR的计数操作
0	0	0	间隔定时器模式	间隔定时器/ 方波输出/ 分频器功能 / PWM输出(主)	递减计数
0	1	0	捕捉模式	输入脉冲间隔测量	递增计数
0	1	1	事件计数器模式	外部事件计数器	递减计数
1	0	0	单计数模式	延迟计数器/ 单触发脉冲输出/ PWM输出(从属)	递减计数
1	1	0	捕捉&单计数模式	输入信号的高/低电平宽度的测量	递增计数
其他			禁止设置		
各模式的操作根据MD _{0n0} 位的不同而有所差异 (参阅下表)。					

注 位 11 为只读位, 固定为 0, 写入无效。

备注 n: 通道编号 (n = 0 至 7)

图 6-8. 定时器模式寄存器 0n (TMR0n)的格式 (4/4)

地址: F0190H, F0191H (TMR00) 至 F019EH, F019FH (TMR07) 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMR0n (n = 2, 4, 6)	CKS 0n1	CKS 0n0	0	CCS 0n	MAST ER0n	STS 0n2	STS 0n1	STS 0n0	CIS 0n1	CIS 0n0	0	0	MD 0n3	MD 0n2	MD 0n1	MD 0n0
符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMR0n (n = 1, 3)	CKS 0n1	CKS 0n0	0	CCS 0n	SPLIT 0n	STS 0n2	STS 0n1	STS 0n0	CIS 0n1	CIS 0n0	0	0	MD 0n3	MD 0n2	MD 0n1	MD 0n0
符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMR0n (n = 0, 5, 7)	CKS 0n1	CKS 0n0	0	CCS 0n	0 ^{#1}	STS 0n2	STS 0n1	STS 0n0	CIS 0n1	CIS 0n0	0	0	MD 0n3	MD 0n2	MD 0n1	MD 0n0

操作模式 (由MD0n3至MD0n1位设置值 (参阅上表))	MD 0n0	开始计数和中断的设置
• 间隔定时器模式 (0, 0, 0)	0	开始计数时不发生定时器中断 (定时器输出也不发生变化)。
• 捕捉模式 (0, 1, 0)	1	开始计数时发生定时器中断 (定时器输出也会发生变化)。
• 事件计数器模式 (0, 1, 1)	0	开始计数时不发生定时器中断 (定时器输出也不发生变化)。
• 单计数模式 ^{#2} (1, 0, 0)	0	计数操作中的开始触发为无效。 但是不产生中断。
	1	计数操作中的开始触发为有效 ^{#3} 。 但是不产生中断。
• 捕捉&单计数模式 (1, 1, 0)	0	开始计数时不发生定时器中断 (定时器输出也不发生变化)。 计数操作中的开始触发为无效。 但是不产生中断。
其他		禁止设置

- 注 1. 位 11 为只读位，固定为 0，写入无效。
 2. 单计数模式下，不控制开始计数操作时的中断输出(INTTM0n)和 TO0n 输出。
 3. 操作过程中执行开始触发(TS0n = 1)后，计数器将被初始化，同时重新开始计数（不发生中断请求）。

备注 n: 通道编号 (n = 0 至 7)

(4) 定时器状态寄存器 0n (TSR0n)

TSR0n 寄存器表示通道 n 的计数器的溢出状态。

TSR0n 寄存器仅在捕捉模式(MD0n3 至 MD0n1 = 010B)和捕捉&单计数模式(MD0n3 至 MD0n1 = 110B)下有效。关于 OVF 位在各种操作模式下的操作以及设置/清除条件，请参阅表 6-5。

使用 16 位存储器操作指令读取 TSR0n 寄存器。

可以通过 TSR0nL 用 8 位存储器操作指令读取 TSR0n 寄存器的低 8 位。

产生复位信号后，该寄存器被清除为 0000H。

图 6-9. 定时器状态寄存器 0n (TSR0n)的格式

地址: F01A0H, F01A1H (TSR00)至F01AEH, F01AFH (TSR07) 复位后: 0000H R

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSR0n	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	OVF

OVF	通道n的计数器溢出状态
0	没有发生溢出。
1	发生溢出。
当OVF = 1, 捕捉到下一个值而不产生溢出时, 该标志被清除(OVF = 0)。	

备注 n: 通道编号 (n = 0 至 7)

表 6-5. 每种操作模式下的 OVF 位操作和设置/清除条件

定时器操作模式	OVF位	设置/清除条件
• 捕捉模式	清除	捕捉时没有发生溢出的情况下
• 捕捉&单计数模式	设置	捕捉时发生溢出的情况下
• 间隔定时器模式	清除	— (禁止使用)
• 事件计数器模式	设置	
• 单计数模式		

备注 计数器溢出后 OVF 位不会立即改变，而是在随后的捕捉后改变。

(5) 定时器通道允许状态寄存器 0 (TE0)

TE0 寄存器用于表示各通道的允许或停止定时器操作的状态。

TE0 寄存器的每个位对应于定时器通道开始寄存器 0(TS0)以及定时器通道停止寄存器 0(TT0)的每个位。当设置 TS0 寄存器的某一位为 1 时，该寄存器的对应位也被设置为 1。当设置 TT0 寄存器的某一位为 1 时，该寄存器的对应位被清除为 0。

使用 16 位存储器操作指令读取 TE0 寄存器。

可以通过 TE0L 用 1 位或 8 位存储器操作指令读取 TE0 寄存器的低 8 位。

产生复位信号后，该寄存器被清除为 0000H。

图 6-10. 定时器通道允许状态寄存器 0 (TE0)的格式

地址: F01B0H, F01B1H 复位后: 0000H R

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TE0	0	0	0	0	TEH03	0	TEH01	0	TE07	TE06	TE05	TE04	TE03	TE02	TE01	TE00

TEH03	当通道3处于8位定时器模式时，表示高8位定时器的操作允许或停止状态
0	停止操作状态。
1	允许操作。

TEH01	当通道1处于8位定时器模式时，表示高8位定时器的操作允许或停止状态
0	停止操作状态。
1	允许操作。

TE0n	表示通道n的操作允许或停止状态
0	停止操作状态。
1	允许操作。
当通道1或3处于8位定时器模式时，该位表示TE01和TE03的低8位定时器的操作允许或停止状态。	

备注 n: 通道编号 (n = 0 至 7)

(6) 定时器通道开始寄存器 0 (TS0)

TS0 寄存器是触发寄存器，用于清除定时器/计数器寄存器 0n(TCR0n)并开始各通道的计数操作。

当设置该寄存器的某一位为 1，定时器通道允许状态寄存器 0 (TE0)的对应位也被设置为 1。因为 TS0n、TSH01、TSH03 位是触发位，允许操作(TE0n, TEH01, TEH03 = 1)后立即被清除为 0。

使用 16 位存储器操作指令设置 TS0 寄存器。

可以通过 TS0L 用 1 位或 8 位存储器操作指令设置 TS0 寄存器的低 8 位。

产生复位信号后，该寄存器被清除为 0000H。

图 6-11. 定时器通道开始寄存器 0 (TS0)的格式

地址: F01B2H, F01B3H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TS0	0	0	0	0	TSH03	0	TSH01	0	TS07	TS06	TS05	TS04	TS03	TS02	TS01	TS00

TSH03	当通道3处于8位定时器模式时，允许（开始）高8位定时器操作的触发
0	无触发操作。
1	将TEH03位设置为1并允许计数操作。 计数操作允许状态下，在间隔定时器模式时TCR03寄存器开始计数操作(参阅表6-6)。

TSH01	当通道1处于8位定时器模式时，允许（开始）高8位定时器操作的触发
0	无触发操作。
1	将TEH01位设置为1并允许计数操作。 计数操作允许状态下，在间隔定时器模式时TCR01寄存器开始计数操作(参阅表6-6)。

TS0n	允许（开始）通道n操作的触发
0	无触发操作。
1	将TE0n位设置为1并允许计数操作。 计数操作允许状态下，TCR0n寄存器的计数操作的开始因操作模式而异（参阅表6-6）。 当通道1或3处于8位定时器模式时，TS01和TS03为允许（开始）低8位定时器操作的触发。

注意事项 1. 必须将位 15 至 12、10、8 清除为 0。

2. 从不使用 TI0n 引脚输入的功能切换到使用 TI0n 引脚输入的功能的情况下，从设置定时器模式寄存器 0n(TMR0n)后，到 TS0n(TSH01, TSH03)位被设置为 1 为止，需要等待以下期间。

当允许 TI0n 引脚噪声滤波器(TNFEN0n = 1)时：操作时钟(f_{MCK})的四个周期

当禁止 TI0n 引脚噪声滤波器(TNFEN0n = 0)时：操作时钟(f_{MCK})的两个周期

备注 1. TS0 寄存器的读取值总是为 0。

2. n: 通道编号 (n = 0 至 7)

(7) 定时器通道停止寄存器 0 (TT0)

TT0 寄存器为用于停止各通道的计数操作的触发寄存器。

当设置该寄存器的某一位为 1 时，定时器通道允许状态寄存器 0(TE0)的对应位被清除为 0。因为 TT0n、TTH01、TTH03 位是触发位，停止操作(TE0n, TEH01, TEH03 = 0)后立即被清除。

使用 16 位存储器操作指令设置 TT0 寄存器。

可以通过 TT0L 用 1 位或 8 位存储器操作指令设置 TT0 寄存器的低 8 位。

产生复位信号后，该寄存器被清除为 0000H。

图 6-12. 定时器通道结束寄存器 0 (TT0)的格式

地址: F01B4H, F01B5H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TT0	0	0	0	0	TTH03	0	TTH01	0	TT07	TT06	TT05	TT04	TT03	TT02	TT01	TT00

TTH03	当通道3处于8位定时器模式时，停止高8位定时器操作的触发
0	无触发操作。
1	操作停止（产生停止触发）。

TTH01	当通道1处于8位定时器模式时，停止高8位定时器操作的触发
0	无触发操作。
1	操作停止（产生停止触发）。

TT0n	通道n的操作停止触发
0	无触发操作。
1	TE0n位清除为0，变为计数操作停止状态。 当通道1或3处于8位定时器模式时，TT01和TT03为停止低8位定时器操作的触发。

注意事项 必须将 TT0 寄存器的位 15 至 12、10、8 清除为 0。

备注 1. TT0 寄存器的读取值总是为 0。

2. n: 通道编号 (n = 0 至 7)

(8) 定时器输入选择寄存器 0 (TIS0)

TIS0 寄存器用于选择通道 5 的定时器输入。

使用 8 位存储器操作指令设置 TIS0 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 6-13. 定时器输入选择寄存器 0 (TIS0)的格式

地址: F0074H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
TIS0	0	0	0	0	0	TIS02	TIS01	TIS00

TIS02	TIS01	TIS00	选择通道5使用的定时器输入
0	0	0	定时器输入引脚 (TI05)的输入信号
0	0	1	
0	1	0	
0	1	1	
1	0	0	低速片上振荡器时钟(f_{IL})
1	0	1	副系统时钟 (f_{SUB})
其他			禁止设置

注意事项 选择定时器输入的高电平宽度、低电平宽度要求大于 $1/f_{MCK} + 10 \text{ ns}$ 。

因此，选择 f_{SUB} 作为 f_{CLK} (CK5 寄存器的 CSS 位 = 1)时，不能将 TIS02 位设置为 1。

(9) 定时器输出允许寄存器 0 (TOE0)

TOE0 寄存器用于允许或禁止各通道的定时器输出。

对于已经允许定时器输出的通道 n ，不能通过软件改写下文所述的定时器输出寄存器 0(TO0)的 TO0n 位的值，同时，通过计数操作从定时器输出引脚(TO0n)输出定时器输出功能反映的值。

使用 16 位存储器操作指令设置 TOE0 寄存器。

可以通过 TOE0L 用 1 位或 8 位存储器操作指令设置 TOE0 寄存器的低 8 位。

产生复位信号后，该寄存器被清除为 0000H。

图 6-14. 定时器输出允许寄存器 0 (TOE0)的格式

地址: F01BAH, F01BBH 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TOE0	0	0	0	0	0	0	0	0	0	TOE 06	TOE 05	0	TOE 03	0	0	0

TOE 0n	允许/禁止通道n的定时器输出
0	禁止定时器的输出。 定时器操作没有反映到TO0n位，固定输出。 允许写入TO0n位。
1	允许定时器的输出。 定时器操作反映到TO0n位，产生输出波形。 禁止写入TO0n位（写入被忽略）。

注意事项 必须将位 15 至 7、4 和 2 至 0 清除为 0。

备注 n: 通道编号 (n = 3, 5, 6)

(10) 定时器输出寄存器 0 (TO0)

TO0 寄存器是各通道的定时器输出的缓冲器寄存器。

该寄存器的各个位的值从各通道的定时器输出引脚(TO0n)输出。

仅限禁止定时器输出(TOE0n = 0)时可以用软件改写该寄存器的 TO0n 位。当允许定时器输出(TOE0n = 1)时，用软件改写该寄存器无效，且仅能通过定时器操作改变其值。

要把 P31/TI03/TO03、P05/TI05/TO05 或 P06/TI06/TO06 引脚用作端口功能使用时，须将对应的 TO0n 位设置为 0。

使用 16 位存储器操作指令设置 TO0 寄存器。

可以通过 TO0L 用 8 位存储器操作指令设置 TO0 寄存器的低 8 位。

产生复位信号后，该寄存器被清除为 0000H。

图 6-15. 定时器输出寄存器 0 (TO0)的格式

地址: F01B8H, F01B9H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TO0	0	0	0	0	0	0	0	0	0	TO06	TO05	0	TO03	0	0	0

TO0 n	通道n的定时器输出
0	定时器输出值为0。
1	定时器输出值为1。

注意事项 必须将位 15 至 7、4 和 2 至 0 清除为 0。

备注 n: 通道编号 (n = 3, 5, 6)

(11) 定时器输出电平寄存器 0 (TOL0)

TOL0 寄存器是用于控制各通道定时器输出电平的寄存器。

当在从属通道输出模式(TOM0n = 1)下允许定时器输出(TOE0n = 1)时, 该寄存器对各通道 n 的反转设定, 可以通过定时器输出信号置位复位的时序来反映。主通道输出模式(TOM0n = 0)时, 对该寄存器的设置无效。

使用 16 位存储器操作指令设置 TOL0 寄存器。

可以通过 TOL0L 用 8 位存储器操作指令设置 TOL0 寄存器的低 8 位。

产生复位信号后, 该寄存器被清除为 0000H。

图 6-16. 定时器输出电平寄存器 0 (TOL0)的格式

地址: F01BCH, F01BDH 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TOL0	0	0	0	0	0	0	0	0	0	TOL 06	TOL 05	0	TOL 03	0	0	0

TOL 0n	通道n定时器输出电平的控制	
0	正逻辑输出(高电平有效)	
1	反相输出(低电平有效)	

注意事项 必须将位 15 至 7、4 和 2 至 0 清除为 0。

备注 1. 如果在定时器操作过程中改写该寄存器的值, 定时器输出反相发生在下一个定时器输出信号改变时, 而不是在改写之后立即反相。

2. n: 通道编号 (n = 3, 5, 6)

(12) 定时器输出模式寄存器 0 (TOM0)

TOM0 寄存器用于控制各通道的定时器输出模式。

某个通道作为单通道操作功能使用时，将使用通道的对应位设置为 0。

某个通道作为多通道联动操作功能（PWM 输出、单触发脉冲输出或多重 PWM 输出）使用时，将主通道的对应位设置为 0，将从属通道的对应位设置为 1。

当允许定时器输出($TOE0n = 1$)时，该寄存器对各通道 n 的设定可以通过定时器输出信号置位复位的时序来反映。

使用 16 位存储器操作指令设置 TOM0 寄存器。

可以通过 TOM0L 用 8 位存储器操作指令设置 TOM0 寄存器的低 8 位。

产生复位信号后，该寄存器被清除为 0000H。

图 6-17. 定时器输出模式寄存器 0 (TOM0)的格式

地址: F01BEH, F01BFH 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TOM0	0	0	0	0	0	0	0	0	0	TOM06	TOM05	0	TOM03	0	0	0

TOM0n	通道n定时器输出模式的控制
0	主通道输出模式 (通过定时器中断请求信号(INTTM0n)产生交替输出)
1	从属通道输出模式 (输出由主通道的定时器中断请求信号(INTTM0n)置位，由从属通道的定时器中断请求信号(INTTM0p)复位)

注意事项 必须将位 15 至 7、4 和 2 至 0 清除为 0。

备注 n: 通道编号

n = 0 至 7 (主通道时 n = 2, 4)

p: 从属通道编号

n = 2 时, p = 3

n = 4 时, p = 5, 6

(关于主通道和从属通道之间关系的详情，请参阅 6.4.1 多通道联动操作功能的基本规则。)

(13) 输入切换控制寄存器(ISC)

ISC 寄存器的 ISC1 和 ISC0 位通过联动使用通道 7 与串行阵列单元 0，以实现 LIN-bus 和 DMX512 通信操作。当 ISC1 位被设置为 1 时，串行数据输入引脚(RxD0)的输入信号被选作定时器输入信号。

使用 1 位或 8 位存储器操作指令设置 ISC 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 6-18. 输入切换控制寄存器(ISC)的格式

地址: F0073H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ISC	0	0	0	0	0	0	ISC1	ISC0

ISC1	切换定时器阵列单元通道7的输入
0	不要在通道7使用定时器输入信号。
1	RxD0引脚的输入信号用作定时器输入 (LIN-bus中，用于检测唤醒信号、测量间隔段的低电平宽度和同步段的脉冲宽度；DMX512种，用于检测各信号宽度)。

ISC0	切换外部中断(INTP0)输入
0	使用INTP0引脚的输入信号作为外部中断(普通操作)。
1	使用RxD0引脚的输入信号作为外部中断。 (LIN-bus中，用于检测唤醒信号；DMX512中，用于检测BREAK信号)。

注意事项 必须将位 7 至 2 清除为 0。

备注 使用 LIN-bus 和 DMX512 通信功能时，通过设置 ISC1 为 1 来选择 RxD0 引脚作为输入信号。

(14) 噪声滤波器允许寄存器 1 (NFEN1)

NFEN1 寄存器用于设置是否将噪声滤波器用于输入各通道的定时器输入信号。

将需要消除噪声的引脚的对应位设置为 1，允许噪声滤波器。

允许噪声滤波器时，对 CPU/外围硬件时钟(fmck)执行 2 个时钟的一致检测和同步。关闭噪声滤波器时，只对 CPU/外围硬件时钟(fclk)执行同步。

使用 1 位或 8 位存储器操作指令设置 NFEN1 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 6-19. 噪声滤波器允许寄存器 1 (NFEN1)的格式

地址: F0071H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
NFEN1	TNFEN07	TNFEN06	TNFEN05	0	TNFEN03	0	0	0
TNFEN07	RxD0/P11引脚输入信号的噪声滤波器的允许/禁止使用 ^注							
0	噪声滤波器关闭							
1	噪声滤波器启用							
TNFEN06	TI06/TO06/P06引脚输入信号的噪声滤波器的允许/禁止使用							
0	噪声滤波器关闭							
1	噪声滤波器启用							
TNFEN05	TI05/TO05/P05引脚输入信号的噪声滤波器的允许/禁止使用							
0	噪声滤波器关闭							
1	噪声滤波器启用							
TNFEN03	TI03/TO03/P31引脚输入信号的噪声滤波器的允许/禁止使用							
0	噪声滤波器关闭							
1	噪声滤波器启用							

注 如要使 RxD0/P11 引脚的选择设置有效，请将输入切换控制寄存器(ISC)的 ISC1 位设置为 1。

备注 通道 0 至通道 7 的定时器输入/输出引脚的存在与否因产品而异。有关详情，请参阅表 6-2 每种产品中配置的定时器输入/输出引脚。

(15) 端口模式寄存器 0, 3 (PM0, PM3)

这些寄存器可以以 1 位为单位设置端口 0 和 3 的输入/输出。

定时器输入/输出引脚的有无因产品而异。使用定时器阵列单元时，各个产品中设置的端口模式寄存器如下所示。

30 引脚产品：PM3

32 和 38 引脚产品：PM0, PM3

将用于定时器输出引脚的复用端口(P31/TI03/TO03、P05/TI05/TO05 和 P06/TI06/TO06)用作定时器输出时，须将对应于各端口的端口模式寄存器(PMxx)的位和端口寄存器(Pxx)的位设置为 0。

示例： 将 P31/TI03/TO03 用作定时器输出时
将端口模式寄存器 3 的 PM31 位设置为 0。
将端口寄存器 3 的 P31 位设为 0。

将用于定时器输入引脚的复用端口 (P31/TI03/TO03、P05/TI05/TO05 和 P06/TI06/TO06) 用作定时器输入时，须将对应于各端口的端口模式寄存器(PMxx)位设为 1。此时，端口寄存器(Pxx)位可以是 0 或 1。

示例： 将 P31/TI03/TO03 用作定时器输出时
将端口模式寄存器 3 的 PM31 位设置为 1。
将端口寄存器 3 的 P31 位设为 1。

使用 1 位或 8 位存储器操作指令来设置 PM0 和 PM3 寄存器。

产生复位信号后，该寄存器被设置为 FFH。

图 6-20. 端口模式寄存器 0、3 的格式 (PM0, PM3) (38 引脚产品)

地址：FFF20H 复位后：FFH R/W

符号	7	6	5	4	3	2	1	0
PM0	1	PM06	PM05	1	PM03	PM02	1	1

地址：FFF23H 复位后：FFH R/W

符号	7	6	5	4	3	2	1	0
PM3	1	1	1	1	1	1	PM31	PM30

PMmn	Pmn引脚输入/输出模式选择(m = 0, 3; n = 0至3, 5, 6)
0	输出模式 (输出缓冲器on)
1	输入模式 (输出缓冲器off)

注意事项 必须将 PM0 寄存器的位 0、1、4、7 以及 PM3 寄存器的位 2 至 7 清除为“1”。

对于 32, 30 和 20 引脚产品，在解除复位后，必须通过软件将以下的位设置为输出模式 (通过设置端口寄存器和端口模式寄存器为 0 来指定)。

32 引脚产品： PM0 寄存器的位 2 和 3 以及 PM3 寄存器的位 0

30 引脚产品： PM0 寄存器的位 5 和 6 以及 PM3 寄存器的位 0

20 引脚产品： PM0 寄存器的位 2、3、5 和 6 以及 PM3 寄存器的位 0 和 1

备注 38 引脚产品的端口模式寄存器 0 和 3 的格式如上图所示。关于其他产品的端口模式寄存器的格式，请参阅表 4-13. 配置在各个产品上的 PMxx、Pxx、PUxx、PIMxx、POMxx、PMCx 寄存器及其对应的位。

6.4 定时器阵列单元的基本规则

6.4.1 多通道联动操作功能的基本规则

使用多通道联动操作功能，即联合使用主通道（主要用于对周期进行计数的基准定时器）和从属通道（以主通道为基准而工作的定时器）时，须遵守以下规则。

- (1) 只能将偶数通道（通道 2、4）设置为主通道。
- (2) 通道 3、5 和 6 可以设置为从属通道。
- (3) 从属通道的编号必须高于主通道。

示例：设置通道 2 为主通道时，可以将通道 3 设置为从属通道。设置通道 4 为主通道时，可以将通道 5 和 6 设置为从属通道。

- (4) 对一个主通道可以设置两个或更多从属通道。
- (5) 使用两个或更多主通道时，从属于一个主通道的从属通道不能跨越其他主通道进行设置。

示例：如果将通道 2 和通道 4 设置为主通道，则可以将通道 3 设置为主通道 2 的从属通道。而不能将通道 5 和 6 设置为主通道 2 的从属通道。

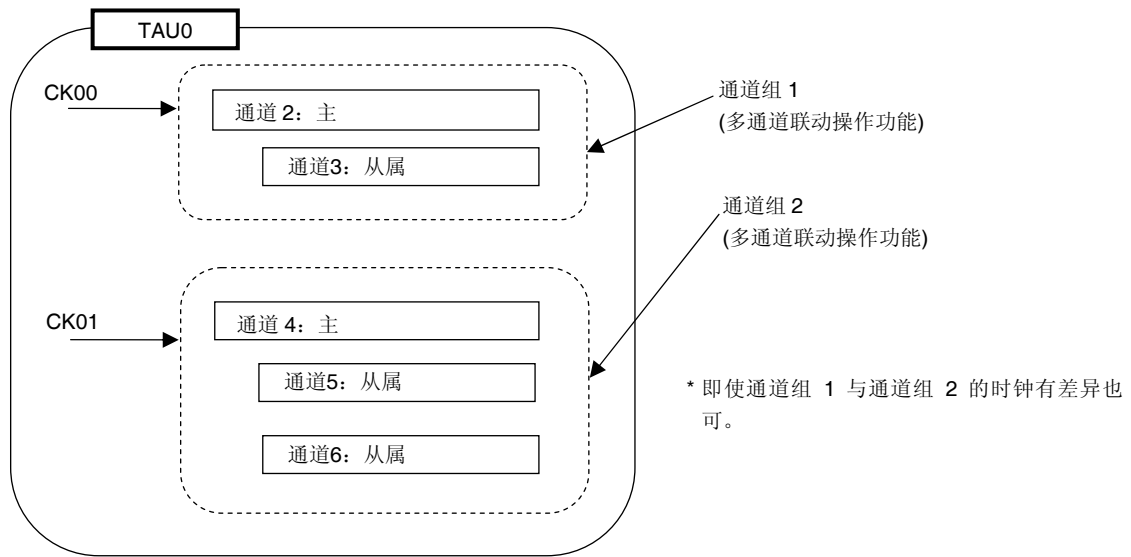
- (6) 与某个主通道联动的从属通道的操作时钟必须与该主通道时钟相同。与主通道联动操作的从属通道的 CKS0n0、CKS0n1 位（定时器模式寄存器 0n (TMR0n) 的位 15、14）的值必须与主通道的相同。
- (7) 主通道可以向更高编号的通道发送 INTTM0n（中断）、开始软件触发和计数时钟。
- (8) 从属通道可以将主通道的 INTTM0n（中断）、开始软件触发或计数时钟用作源时钟，但不能将自己的 INTTM0n（中断）、开始软件触发或计数时钟发送至更高编号的通道。
- (9) 主通道不能将更低编号的主通道的 INTTM0n（中断）、开始软件触发或计数时钟用作源时钟。
- (10) 要同时启用联动操作的通道，因此必须同时设置联动通道的通道开始触发位(TS0n)。
- (11) 在计数操作时，可以仅设置主通道的 TS0n 位或者联动操作的所有通道的 TS0n 位。不能仅设置从属通道的 TS0n 位。
- (12) 要同时停止联动操作的通道，必须同时设置联动通道的通道停止触发位(TT0n)。
- (13) 操作联动通道时，须保持主通道和从属通道的工作时钟同步，因此不能选择 CK02/CK03。
- (14) 定时器模式寄存器 00 (TMR00)中没有主从设置位（固定为 0）。然而，由于通道 0 是最低通道，因此联动操作时可以将通道 0 用作主通道。

多通道联动操作功能的基本规则在通道组（一个实现多通道联动功能的主通道和从属通道的集合）内适用。

如果指定两个或更多的彼此间不联动操作的通道组，则通道组之间不适用本章中所述的规则。

备注 n: 通道编号 (n = 0 至 7)

示例



6.4.2 8 位定时器操作功能的基本规则（仅限通道 1 和 3）

通过 8 位定时器操作功能，可以将 16 位定时器通道作为两个 8 位定时器通道来使用。

该功能仅用于通道 1 和通道 3，并须遵守一些规则。

该功能的基本规则如下所示：

- (1) 8 位定时器操作功能仅适用于通道 1 和通道 3。
- (2) 作为 8 位定时器使用时，将定时器模式寄存器 0n (TMR0n) 的 SPLIT 位设置为 1。
- (3) 高 8 位定时器可以用作间隔定时器功能。
- (4) 开始操作时，高 8 位输出 INTTM01H/INTTM03H（中断）（执行与设置 MD0n0 为 1 时相同的操作）。
- (5) 高 8 位定时器的操作时钟的选择须根据低位 TMR0n 寄存器的 CKS0n1 和 CKS0n0 位进行操作。
- (6) 对高 8 位定时器，通过操作 TSH1/TSH3 位来开始通道操作，通过操作 TTH1/TTH3 位来停止通道操作。可以利用 TEH1/TEH3 位来确认通道状态。
- (7) 低 8 位定时器依据 TMR0n 寄存器的设置工作。以下三种功能支持低 8 位定时器操作：
 - 间隔定时器功能
 - 外部事件计数器(仅限通道 3)
 - 延迟计数功能(仅限通道 3)
- (8) 对于低 8 位定时器，通过操作 TS1/TS3 位来开始通道操作，通过操作 TT1/TT3 位来停止通道操作。可以利用 TE1/TE3 位来确认通道状态。
- (9) 在 16 位定时器操作中，对 TSH1、TSH3、TTH1 和 TTH3 位进行的操作无效。通过操作 TS1、TS3、TT1 和 TT3 位来操作通道 1 和通道 3。TEH3 位和 TEH1 位不变。
- (10) 在使用 8 位定时器功能时，不能使用联动操作功能（单触发脉冲、PWM 和多重 PWM）。

备注 n: 通道编号 (n = 1, 3)

6.5 计数器的操作

6.5.1 计数时钟(f_{CLK})

定时器阵列单元的计数时钟(f_{CLK})可以通过定时器模式寄存器 0n (TMR0n)的 CCS0n 位从以下两者之中选择。

- 由 CKS0n0 和 CKS0n1 位指定的操作时钟(f_{MCK})
- TIO0n 引脚的输入信号的有效边沿

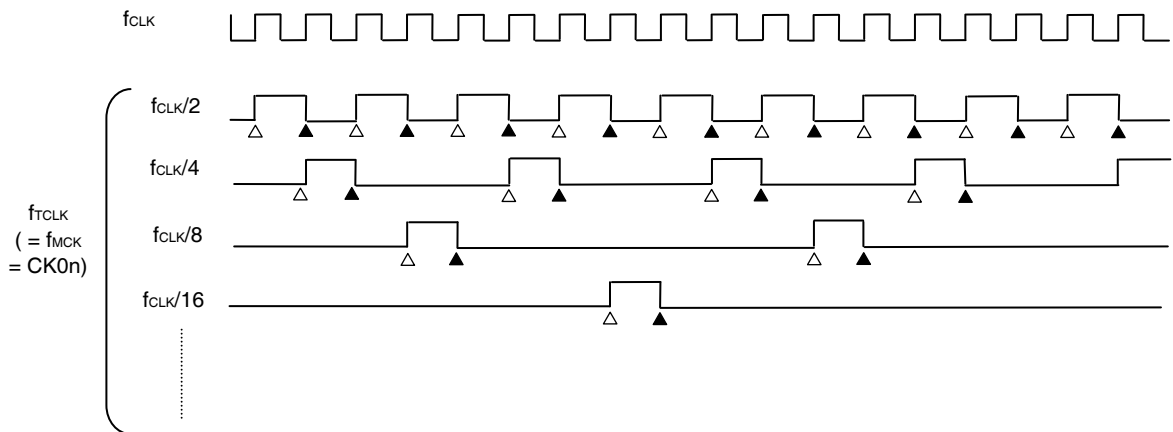
由于定时器阵列单元被设计为与 f_{CLK} 同步操作，计数时钟(f_{CLK})的时序如下所示。

(1) 选择由 CKS0n0 和 CKS0n1 位指定的操作时钟(f_{MCK})时 (CCS0n = 0)

根据定时器时钟选择寄存器 m(TPSm)的设置，计数时钟(f_{CLK}) 在 f_{CLK} 与 $f_{CLK} / 2^{15}$ 之间进行选择。然而，当选择为 f_{CLK} 的分频时，由 TPS0n 寄存器选择的时钟变成了每个上升沿后只保持 1 个 f_{CLK} 周期的高电平的信号。选择为 f_{CLK} 时，固定在高电平。

为了与 f_{CLK} 保持同步，定时器/计数器寄存器 0n (TCR0n)是在计数时钟的上升沿等待 1 个 f_{CLK} 时钟周期后开始计数。出于方便，将其称为“在计数时钟的上升沿计数”。

图 6-21. f_{CLK} 和计数时钟(f_{CLK})的时序(CCS0n = 0 时)



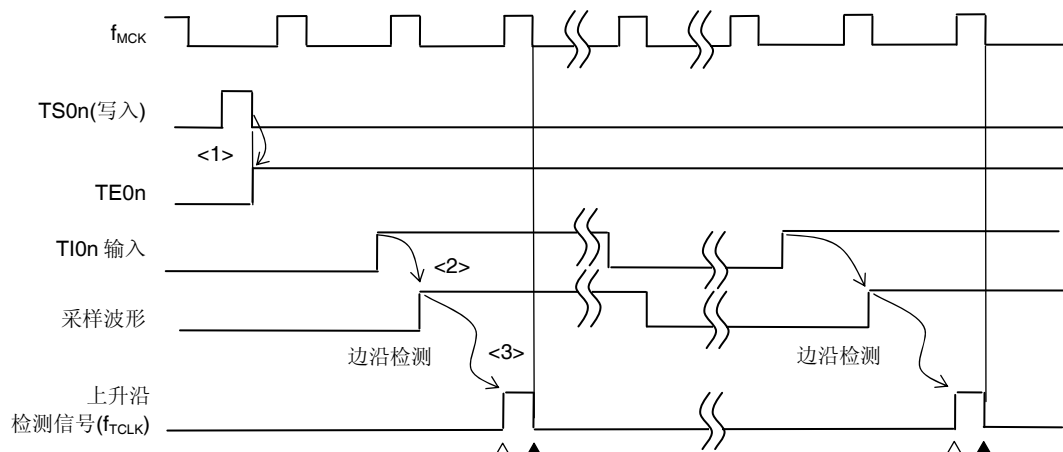
- 备注 1. Δ : 计数时钟的上升沿
 \blacktriangle : 同步、计数器的递增/递减
2. f_{CLK} : CPU/外围硬件时钟

(2) 选择 TI0n 引脚输入信号的有效边沿时(CCS0n = 1)

计数时钟(f_{TCLK})为检测 TI0n 引脚输入信号的有效边沿并同步至下一个 f_{MCK} 上升沿的信号。计数时钟(f_{TCLK})比实际上的 TI0n 引脚的输入信号延迟了 1 至 2 个 f_{MCK} 时钟周期(使用噪声滤波器时, 延迟 3 至 4 个时钟周期)。

为了与 f_{CLK} 保持同步, 定时器/计数器寄存器 0n (TCR0n)是在计数时钟的上升沿等待 1 个 f_{CLK} 时钟周期后开始计数。出于方便, 将其称为“在 TI0 引脚输入信号的有效边沿计数”。

图 6-22. f_{CLK} 和计数时钟(f_{TCLK})的时序(CCS0n = 1, 未使用噪声滤波器时)



<1> 设置 TS0n 位为 1 以启动定时器, 并成为等待 TI0n 引脚输入信号的有效边沿的状态。

<2> f_{MCK} 对 TI0n 引脚输入信号的上升沿进行采样。

<3> 在采样信号的上升沿检测边沿, 并输出检测信号(计数时钟)。

备注 1. Δ: 计数时钟的上升沿

▲: 同步、计数器的递增/递减

2. f_{CLK} : CPU/外围硬件时钟

f_{MCK} : 通道 n 的操作时钟

3. 输入脉冲间隔测量、输入信号高/低电平的测量、延迟计数器以及单触发脉冲输出功能的 TI0n 引脚输入信号具有相同的波形, 如图 6-22 所示。

6.5.2 计数器的开始时序

通过设置定时器通道开始寄存器 0 (TS0n)的 TS0n 位，可以允许操作定时器/计数器寄存器 0n(TCR0n)。

从计数操作允许状态到定时器/计数器寄存器 0n (TCR0n)计数开始为止的操作如表 6-6 所示。

表 6-6. 从计数操作允许状态到定时器/计数器寄存器 0n (TCR0n)计数开始为止的操作

定时器操作模式	设置为TS0n = 1时的操作
• 间隔定时器模式	从检测到开始触发(TS0n=1)直到产生计数时钟为止不执行任何操作。 在第一个计数时钟将TDR0n寄存器的值载入TCR0n寄存器，在之后的计数时钟执行递减计数操作（参阅 6.5.3 (1) 间隔定时器模式下的操作）。
• 事件计数器模式	向TS0n位写入1，把TDR0n寄存器的值载入TCR0n寄存器。 如果输入TI0n的检测沿。后续计数时钟执行递减计数操作（参阅6.5.3 (2) 事件计数器模式下的操作）。
• 捕捉模式	从检测到开始触发(TS0n = 1)到产生计数时钟为止不执行任何操作。 在第一个计数时钟将0000H载入TCR0n寄存器，在后续计数时钟执行递增计数操作（参阅 6.5.3 (3) 捕捉模式下的操作 (输入脉冲的间隔测量)）。
• 单计数模式	在定时器停止(TE0n = 0)状态下，向TS0n位写入1进入等待开始触发状态。 从检测到开始触发直到产生计数时钟为止不执行任何操作。 在第一个计数时钟将TDR0n寄存器的值载入TCR0n寄存器，在之后的计数时钟执行递减计数操作（参阅 6.5.3 (4) 单计数模式下的操作）。
• 捕捉&单计数模式	在定时器停止(TE0n = 0)状态下，向TS0n位写入1进入等待开始触发状态。 从检测到开始触发直到产生计数时钟为止不执行任何操作。 在第一个计数时钟将0000H载入TCR0n寄存器，在后续计数时钟执行递增计数操作(参阅 6.5.3 (5) 捕捉&单计数模式下的操作 (高电平宽度的测量))。

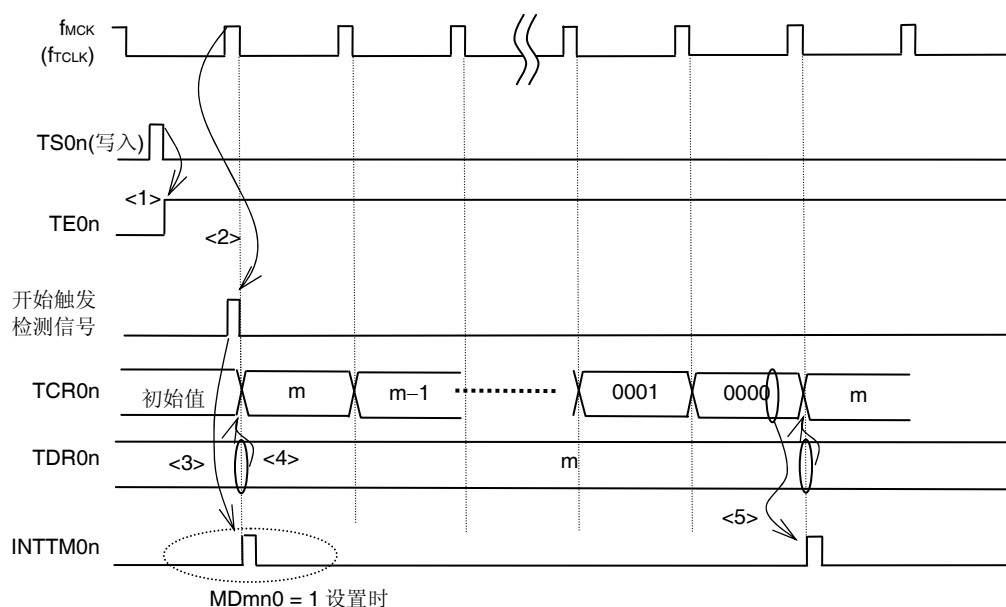
6.5.3 计数器的操作

以下对各模式下的计数器操作进行说明。

(1) 间隔定时器模式下的操作

- <1> 通过向 TS0n 位写入 1 允许定时器操作 (TE0n = 1)。直到产生计数时钟为止定时器/计数器寄存器 0n (TCR0n) 将保持初始值。
- <2> 允许操作时, 在第一个计数时钟产生开始触发信号。
- <3> 设置 MD0n0 位为 1 时, 开始触发信号将产生 INTTM0n。
- <4> 在允许操作后的第一个计数时钟, 将定时器数据寄存器 0n (TDR0n) 的值被载入 TCR0n 寄存器, 并在间隔定时器模式下开始计数。
- <5> 当 TCR0n 寄存器递减计数至其计数值为 0000H 时, 产生 INTTM0n, 并将定时器数据寄存器 0n (TDR0n) 的值载入 TCR0n 寄存器, 继续计数。

图 6-23. 操作时序(间隔定时器模式下)



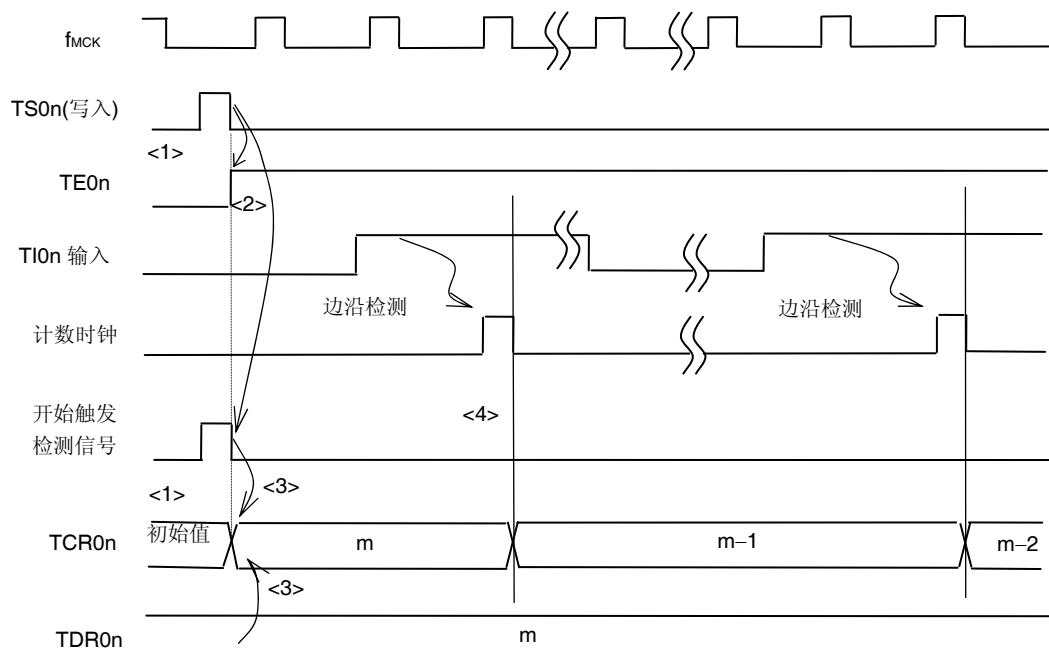
注意事项 在计数时钟的第一个周期操作中, 写 TS0n 位之后到产生计数时钟为止会发生开始计数的延迟, 将产生最大为一个时钟周期的误差。需要获得开始计数时序信息时, 可以通过设置 MD0n0 = 1 在开始计数时产生中断。

备注 f_{MCK} 、开始触发检测信号和 INTTM0n 与 f_{CLK} 同步后, 在一个时钟周期之内被激活。

(2) 事件计数器模式下的操作

- <1> 停止操作($TE0n = 0$)时, 定时器/计数器寄存器 $0n$ ($TCR0n$)保持初始值。
- <2> 通过向 $TS0n$ 位写入 1 允许定时器操作($TE0n = 1$)。
- <3> 向 $TS0n$ 位写入 1 并且 $TE0n$ 位被设置为 1 后, 立即将定时器数据寄存器 $0n$ ($TDR0n$)的值载入到 $TCR0n$ 寄存器并开始计数。
- <4> 此后, 依据 $Ti0n$ 输入的有效边沿的计数时钟对 $TCR0n$ 寄存器值进行递减计数。

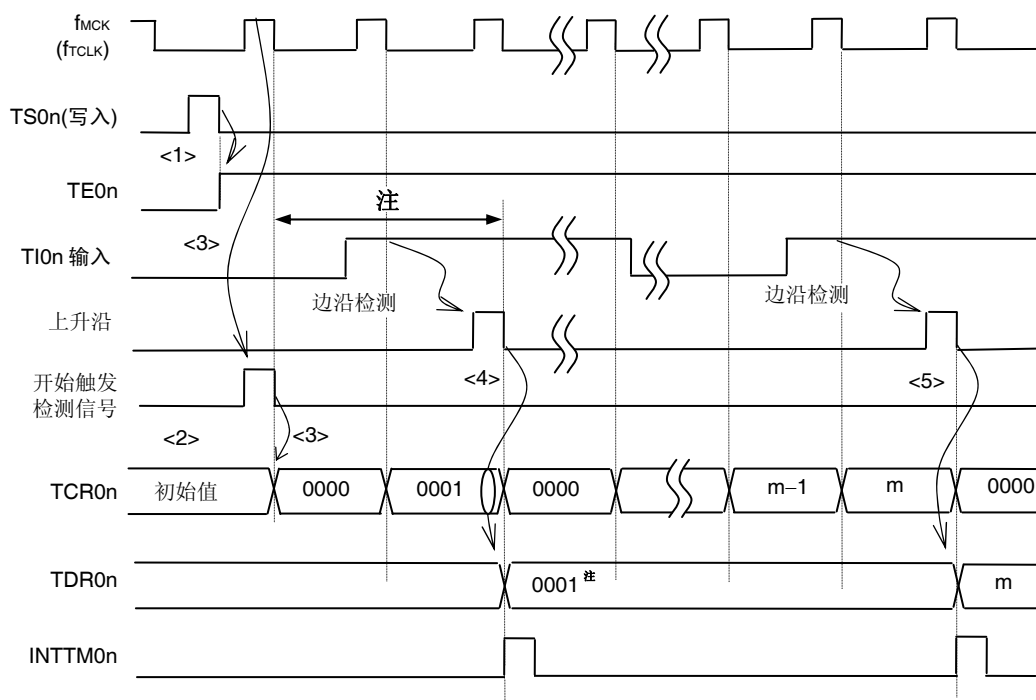
图 6-24. 操作时序(事件计数器模式下)



备注 不使用噪声滤波器时的时序如图 6-24 所示。启用噪声滤波器时, 边沿检测将比 $Ti0n$ 输入再晚 2 个 $fMCK$ 周期 (合计 3 至 4 个周期)。

(3) 捕捉模式下的操作 (输入脉冲的间隔测量)

- <1> 通过向 $TS0n$ 位写入 1 允许定时器操作($TE0n = 1$)。
- <2> 直到产生计数时钟为止定时器/计数器寄存器 $0n$ ($TCR0n$)将保持初始值。
- <3> 允许操作时, 在第一个计数时钟产生开始触发信号。将 $0000H$ 载入 $TCR0n$ 寄存器并以捕捉模式开始计数。(设置 $MD0n0$ 位为 1 时, 开始触发将产生 $INTTM0n$ 。)
- <4> 检测到 $Ti0n$ 输入的有效边沿时, 将 $TCR0n$ 寄存器的值捕捉至定时器数据寄存器 $0n$ ($TDR0n$), 并产生 $INTTM0n$ 。此时捕捉值没有意义。 $TCR0n$ 寄存器从 $0000H$ 开始继续计数。
- <5> 检测到下一个 $Ti0n$ 输入的有效边沿时, 将 $TCR0n$ 寄存器的值捕捉至定时器数据寄存器 $0n$ ($TDR0n$), 并产生 $INTTM0n$ 。

图 6-25. 操作时序(捕捉模式下: 输入脉冲的间隔测量)

注 如果捕捉启动时时钟被一直输入到 $Ti0n$ (存在触发), 即便没有检测到沿, 当检测到触发时就会开始计数。因此, 第一个捕捉值(<4>) 并不能决定脉冲间隔(上图中 0001 仅表示 2 个时钟周期, 并不能决定脉冲间隔), 因此用户可忽略。

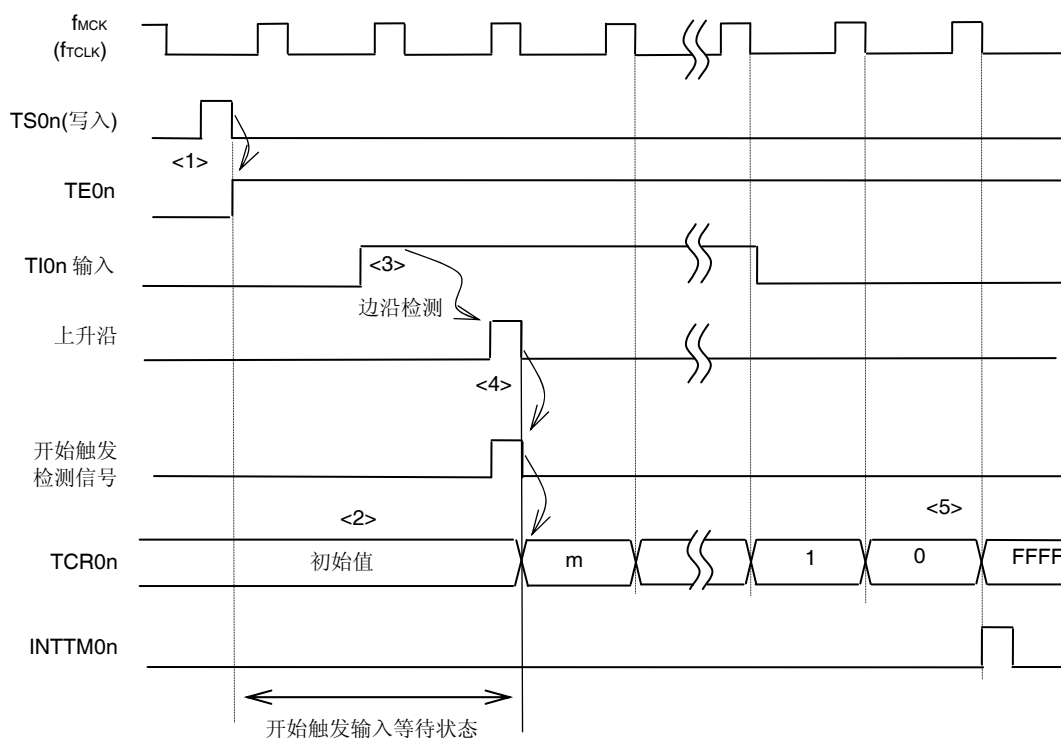
注意事项 在计数时钟的第一个周期操作中, 写入 $TS0n$ 位之后到产生计数时钟为止发生了开始计数的延迟, 将产生最大为一个时钟周期的误差。需要获得开始计数时序信息时, 可以通过设置 $MD0n0 = 1$ 在开始计数时产生中断。

备注 不使用噪声滤波器时的时序如图 6-25 所示。启用噪声滤波器时, 边沿检测将比 $Ti0n$ 输入再晚 2 个 f_{MCK} 周期 (合计 3 至 4 个周期)。

(4) 单计数模式下的操作

- <1> 通过向 TS0n 位写入 1 允许定时器操作 (TE0n = 1)。
- <2> 定时器/计数器寄存器 0n (TCR0n) 保持初始值直到产生开始触发信号为止。
- <3> 检测到 TI0n 输入的上升沿。
- <4> 检测到开始触发信号时，将定时器数据寄存器 0n (TDR0n) 的值载入 TCR0n 寄存器并开始计数。
- <5> 当 TCR0n 寄存器递减计数至其计数值为 0000H 时，产生 INTTM0n，TCR0n 寄存器的值变成 FFFFH 并停止计数。

图 6-25. 操作时序(单计数模式下)

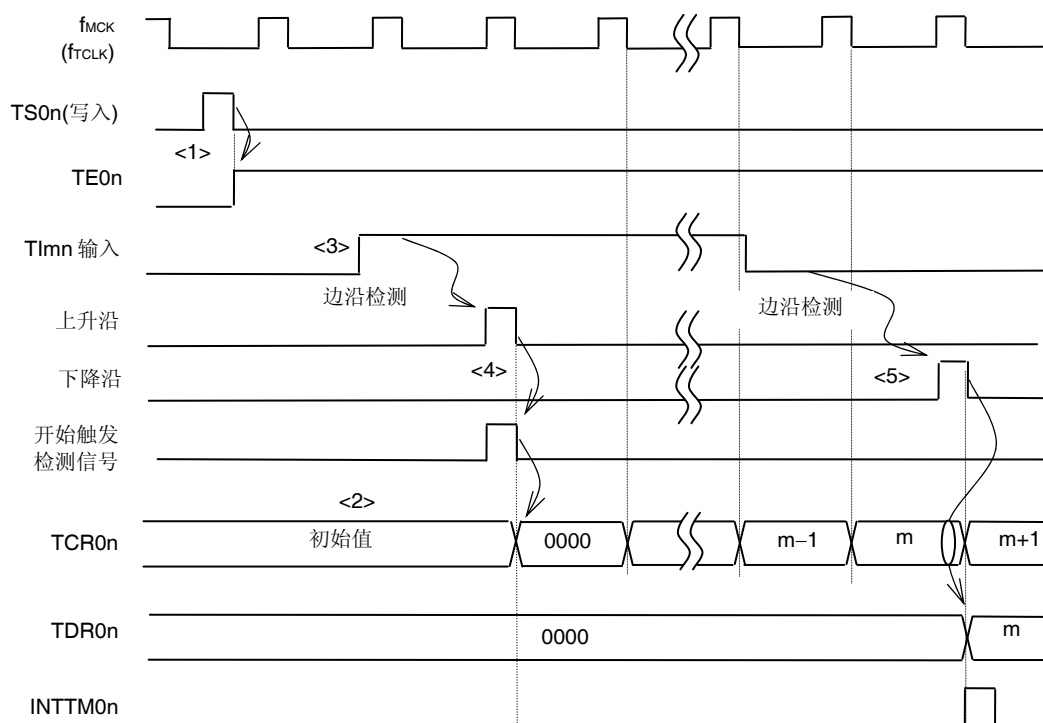


备注 不使用噪声滤波器时的时序如图 6-26 所示。启用噪声滤波器时，边沿检测将比 TI0n 输入再晚 2 个 f_{MCK} 周期（合计 3 至 4 个周期）。1 个周期的误差来源于 TI0n 输入与计数时钟(f_{MCK})的周期之间的不同步。

(5) 捕捉&单计数模式下的操作（高电平宽度的测量）

- <1> 通过向定时器通道开始寄存器 0(TS0)的 TS0n 位写入 1 允许定时器操作(TE0n = 1)。
- <2> 定时器/计数器寄存器 0n (TCR0n)保持初始值直到产生开始触发信号为止。
- <3> 检测到 TI0n 输入的上升沿。
- <4> 检测到开始触发信号时，将 0000H 值载入 TCR0n 寄存器并开始计数。
- <5> 检测到 TI0n 输入的下落沿时，将 TCR0n 寄存器的值捕捉至定时器数据寄存器 0n (TDR0n)，并产生 INTTM0n。

图 6-27. 操作时序(捕捉&单计数模式下：高电平宽度的测量)

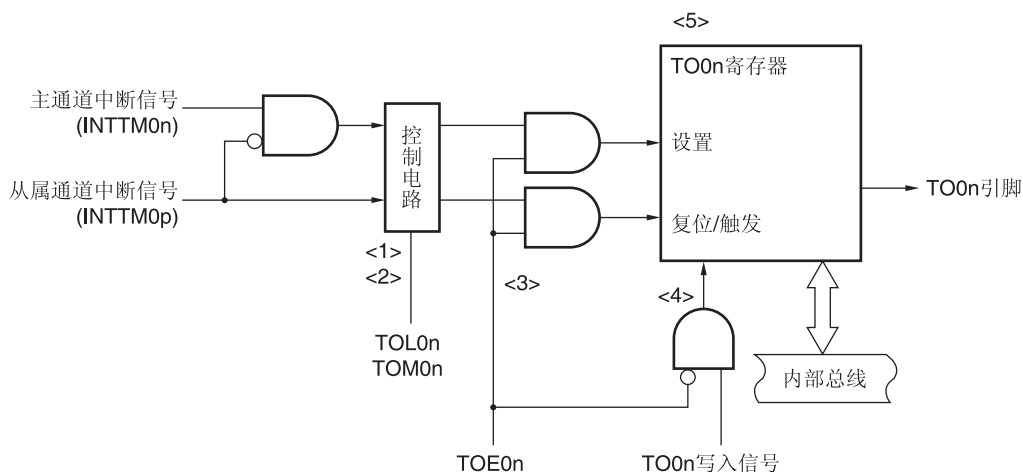


备注 不使用噪声滤波器时的时序如图 6-27 所示。启用噪声滤波器时，边沿检测将比 TI0n 输入再晚 2 个 fMCK 周期（合计 3 至 4 个周期）。1 个周期的误差来源于 TI0n 输入与计数时钟(fMCK)的周期之间的不同步。

6.6 通道输出（TO0n引脚）控制

6.6.1 TO0n引脚输出电路配置

图 6-28. 输出电路配置



下面介绍 TO0n 引脚输出电路。

- <1> 当 $TOM0n = 0$ （主通道输出模式）时，定时器输出电平寄存器 0 (TOL0) 的设定值被忽略，仅将 INTTM0p（从属通道定时器中断）发送至定时器输出寄存器 0 (TO0)。
- <2> 当 $TOM0n = 1$ （从属通道输出模式）时，将 INTTM0n（主通道定时器中断）和 INTTM0p（从属通道定时器中断）的双方都发送至 TO0 寄存器。
此时，TOL0 寄存器变为有效，并按以下方式控制信号：

当 $TOL0n = 0$ ：正逻辑输出（INTTM0n → 置位，INTTM0p → 复位）
 当 $TOL0n = 1$ ：负逻辑输出（INTTM0n → 复位，INTTM0p → 置位）

当同时产生 INTTM0n 和 INTTM0p（占空比为 0% 的 PWM 波形被输出）时，INTTM0p（复位信号）优先，INTTM0n（设置信号）被屏蔽。

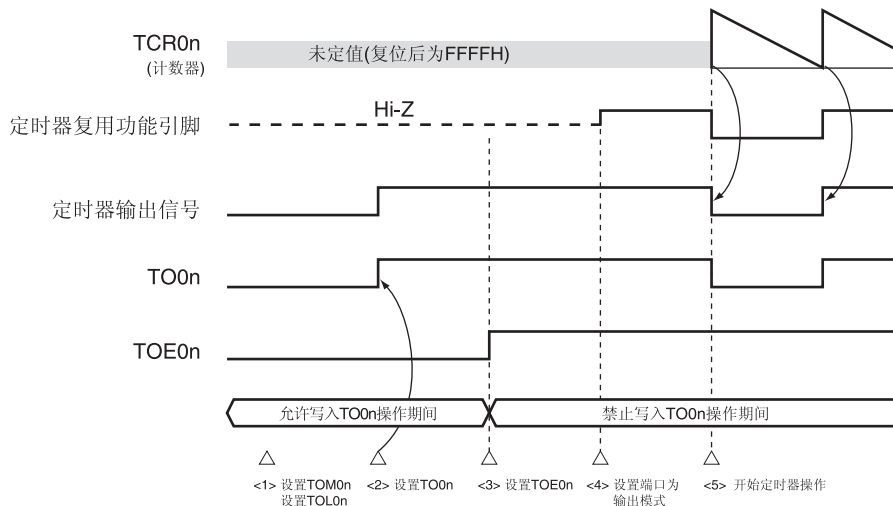
- <3> 当允许定时器输出 ($TOE0n = 1$) 时，将 INTTM0n（主通道定时器中断）和 INTTM0p（从属通道定时器中断）的双方都发送至 TO0 寄存器。向 TO0 寄存器的写入 (TO0n 写信号) 无效。
当 $TOE0n = 1$ 时，中断信号以外的信号不会改变 TO0n 引脚输出。
初始化 TO0n 引脚输出电平时，需要设置为禁止定时器输出 ($TOE0n = 0$) 并向 TO0 寄存器写入值。
- <4> 当禁止定时器输出 ($TOE0n = 0$) 时，对目标通道的 TO0n 位的写入操作 (TO0n 写信号) 有效。当禁止定时器输出 ($TOE0n = 0$) 时，不将 INTTM0n（主通道定时器中断）和 INTTM0p（从属通道定时器中断）的任意一方发送至 TO0 寄存器。
- <5> 任何时候均可读取 TO0 寄存器，确认 TO0n 引脚输出电平。

备注 n: 通道编号
 n = 0 至 7 (n = 2, 4 用于主通道)
 p: 从属通道编号
 n = 2 时, p = 3
 n = 4 时, p = 5, 6

6.6.2 TO0n引脚输出设置

从 TO0n 输出引脚的初始设置到开始定时器操作为止的步骤和状态变化如下图所示。

图 6-29. 从定时器输出设置到操作开始的状态转换



<1> 设置定时器输出的操作模式。

- TOM0n 位 (0: 主通道输出模式, 1: 从属通道输出模式)
- TOL0n 位 (0: 正相输出, 1: 反相输出)

<2> 通过设置定时器输出寄存器 0 (TO0)将定时器输出信号设置为初始状态。

<3> 向 TOE0n 位写入 1 以允许定时器输出操作 (禁止写入 TO0 寄存器)。

<4> 将端口输入/输出设置设为输出 (参阅 6.3 (15) 端口模式寄存器 0, 3 (PM0, PM3))。

<5> 允许定时器操作 (TS0n = 1)。

备注 n: 通道编号 (n = 3, 5, 6)

6.6.3 通道输出操作的注意事项

(1) 在定时器操作时 TO0、TOE0 和 TOL0 寄存器的设置值的变化

由于定时器操作（定时器/计数器寄存器 0n(TCR0n)和定时器数据寄存器 0n(TDR0n)的操作）独立于 TO0n 输出电路，更改在定时器输出寄存器 0(TO0)、定时器输出允许寄存器 0(TOE0)和定时器输出电平寄存器 0(TOL0)的设置值不会影响定时器操作，在定时器操作期间可以更改这些值。然而，各个定时器操作时若要从 TO0n 引脚输出预期波形的话，须将 TO0、TOE0、TOL0 和 TOM0 寄存器设置为如 6.7 和 6.8 节所示各个操作时寄存器设置内容示例的值。

如果在接近各通道定时器中断(INTTM0n)发生时更改 TOE0 和 TOL0 寄存器（TO0 寄存器除外）的设置值，根据更改的执行是在定时器中断(INTTM0n)发生前还是发生后，输出至 TO0n 引脚的波形可能不同。

备注 n: 通道编号 (n = 3, 5, 6)

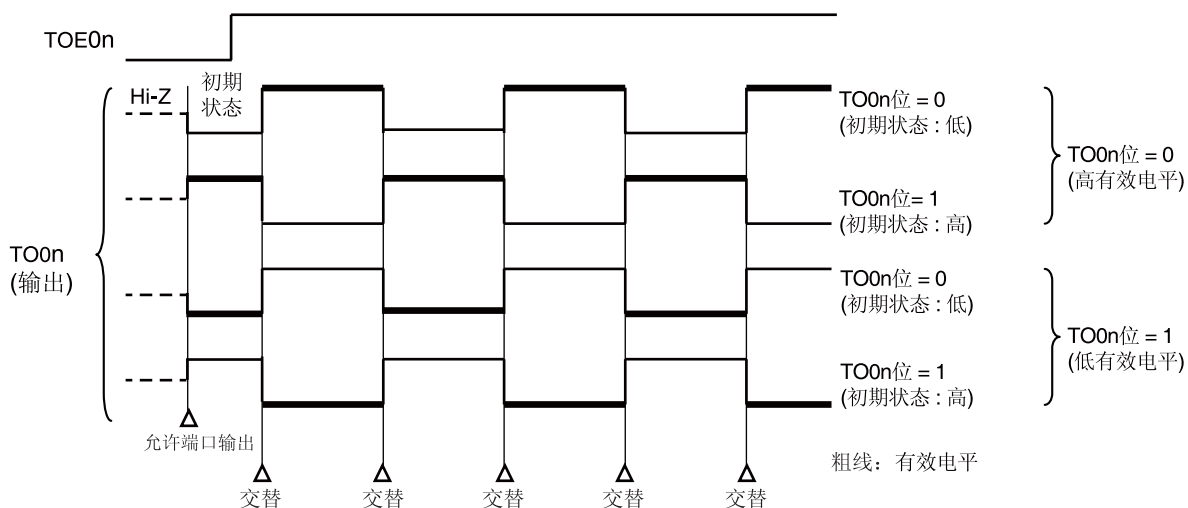
(2) TO0n 引脚的初始电平和开始定时器操作后的输出电平

在允许端口输出之前，禁止定时器输出(TOE0n = 0)的状态下，写入定时器输出寄存器 0(TO0)更改初始电平后，设置为允许定时器输出(TOE0n = 1)时，TO0n 引脚输出电平的变化如下所示。

(a) 在主通道输出模式(TOM0n = 0)设置下开始操作时

主通道输出模式(TOM0n = 0)时，定时器输出电平寄存器 0(TOL0)的设置无效。设置初始电平后开始定时器操作时，产生交替信号使 TO0n 引脚的输出电平反相。

图 6-30. 交替输出(TOM0n = 0)时的 TO0n 引脚输出状态

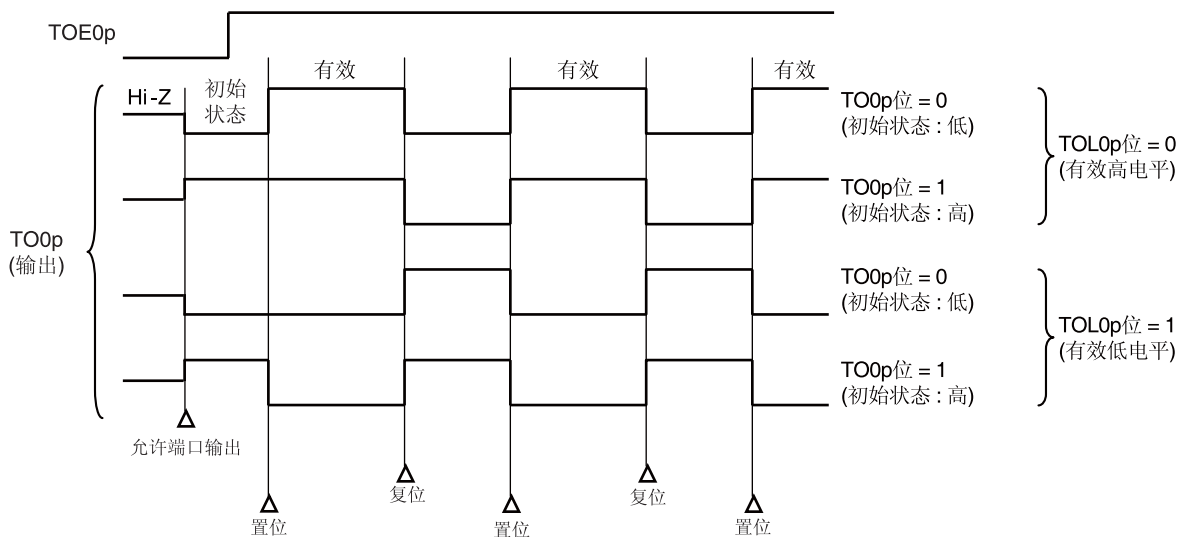


- 备注 1. 交替: TO0n 引脚输出状态反相
- 2. n: 通道编号 (n = 3, 5, 6)

(b) 从属通道输出模式(TOM0p = 1)设置下开始操作时 (PWM 输出)

从属通道输出模式(TOM0p = 1)时, 有效电平取决于定时器输出电平寄存器 0(TOL0)的设置。

图 6-31. PWM 输出时 (TOM0p = 1)的 TO0p 引脚输出状态



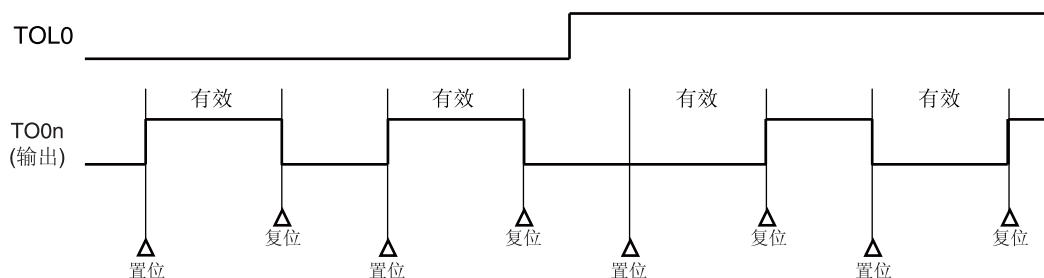
- 备注 1. 置位: TO0p 引脚的输出信号从无效电平变为有效电平。
 复位: TO0p 引脚的输出信号从有效电平变为无效电平。
2. p: 通道编号 (n = 3, 5, 6)

(3) 在从属通道输出模式(TOM0n = 1)下 TO0n 引脚的操作**(a) 在定时器操作期间改变定时器输出电平寄存器 0 (TOL0)设置时**

定时器操作期间更改 TOL0 寄存器的设置时，在 TO0n 引脚变更条件的产生时序设置生效。改写 TOL0 寄存器不会立刻改变 TO0n 引脚的输出电平。

当 TOM0n 被设置为 1，在操作定时器(TE0n = 1)期间更改 TOL0 寄存器的值时的操作如下所示。

图 6-32. 在定时器操作期间改变 TOL0 寄存器内容时的操作



备注 1. 置位：TO0n 引脚的输出信号从无效电平变为有效电平。

复位：TO0n 引脚的输出信号从有效电平变为无效电平。

2. n: 通道编号(n = 3, 5, 6)

(b) 置位/复位时序

为了在 PWM 输出时实现 0%/100%输出，使用从属通道，将产生主通道定时器中断(INTTM0n)时 TO0n 引脚/TO0n 位的设置时序延迟 1 个计数时钟周期。

同时产生设置条件和复位条件时，后者优先。

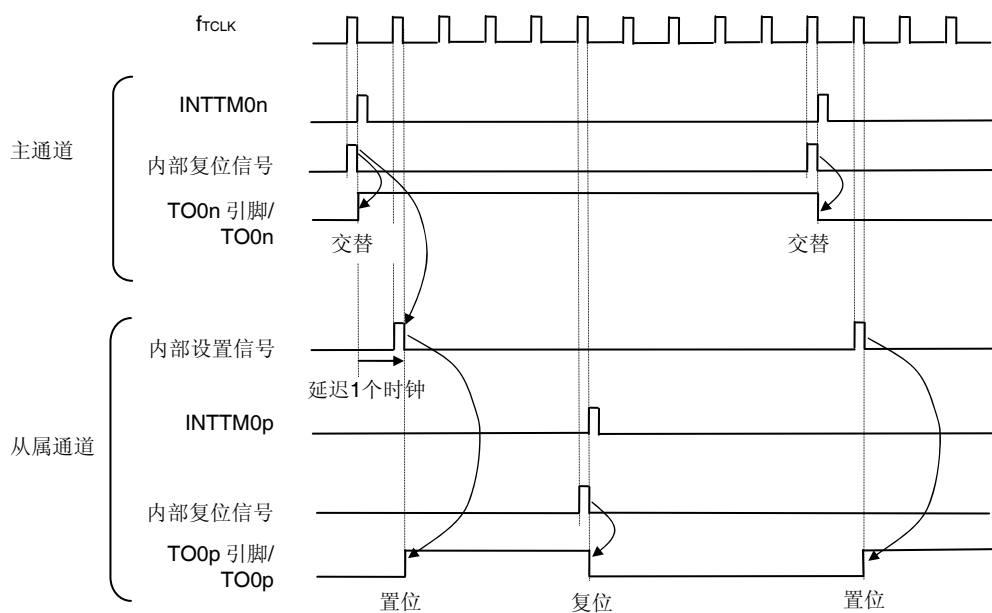
按以下方式设置主/从属通道时，置位/复位操作状态如图 6-31 所示。

主通道： TOE0n = 1, TOM0n = 0, TOL0n = 0

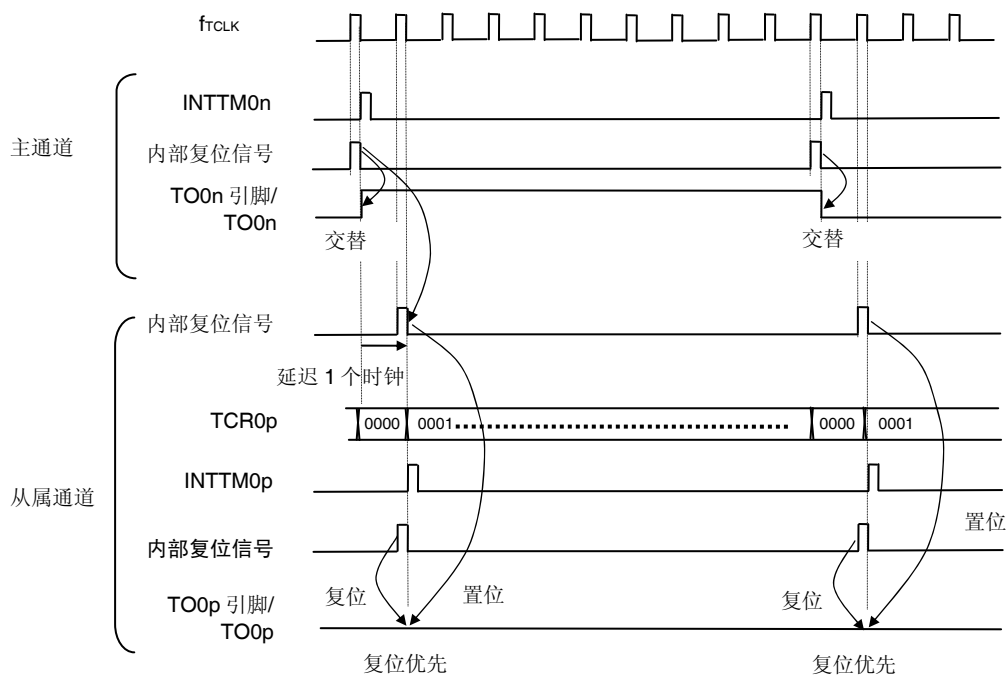
从属通道： TOE0p = 1, TOM0p = 1, TOL0p = 0

图 6-33. 设置/复位时序操作状态

(1) 基本操作时序



(2) 0%占空时的操作时序



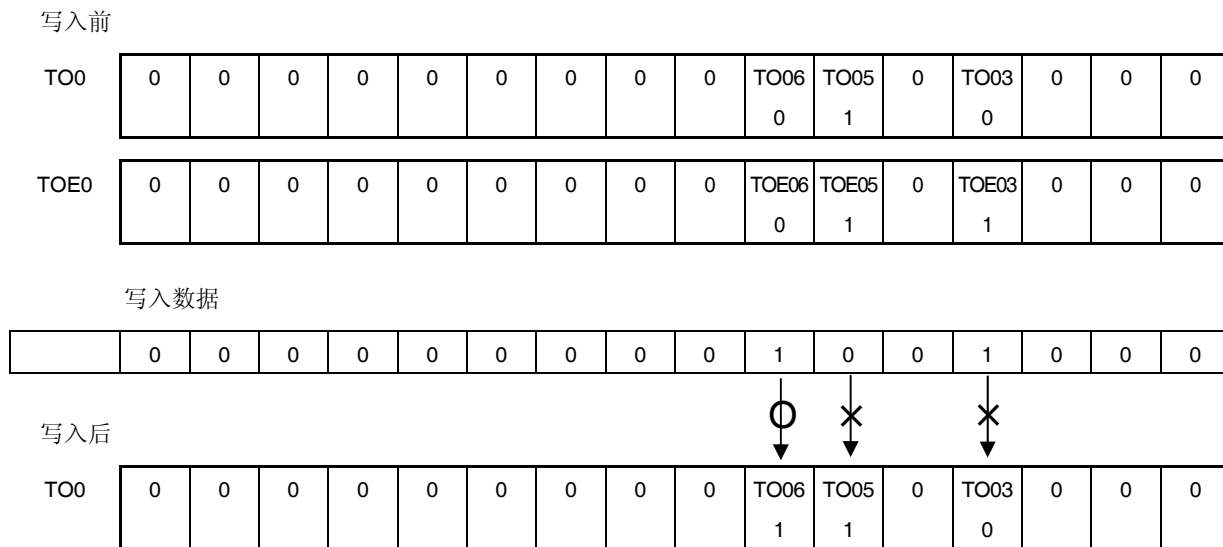
- 备注 1. 内部复位信号: TO0n 引脚复位/交替信号
内部置位信号: TO0n 引脚置位信号
2. n: 通道编号 (n = 0 至 7)
n = 0 至 7 (n = 2, 4 用于主通道)
p: 从属通道编号
n = 2 时, p = 3
n = 4 时, p = 5, 6

6.6.4 TO0n位的集中操作

在定时器输出寄存器 0(TO0)中，采用与定时器通道开始寄存器 0(TS0)相同的形式，在一个寄存器中配置有所有通道的设置位。因此，可以集中操作所有通道的 TO0n 位。

而且因为可以对只想要作为操作对象的通道输出 (TOmn) 的 TOmn 位进行写入，所以对该寄存器的任意位都可以进行独立操作。

图 6-34. 集中操作 TO0n 位的示例



仅可写入 TOE0n = 0 的 TO0n 位，向 TOE0n = 1 的 TO0n 位的写入无效。

设置为 TOE0n = 1 的 TO0n (通道输出) 不受写操作影响。即使对 TO0n 位执行写操作也是无效的，定时器操作导致的输出改变可以被正常执行。

注意事项 允许定时器输出(TOE0n = 1)时，即使各个通道的定时器中断(INTTM0n)所导致的输出与向 TO0n 位的写入相冲突，也正常执行 TO0n 引脚的输出。

备注 n: 通道编号(n = 3, 5, 6)

6.6.5 计数操作开始时的定时器中断和TO0n引脚输出

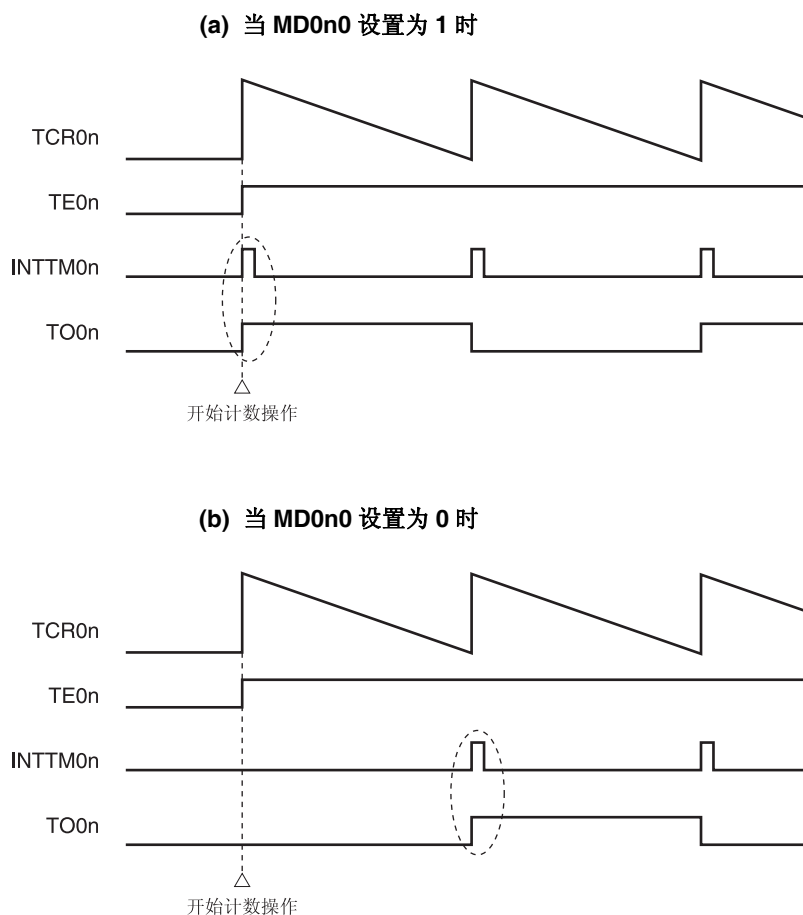
在间隔定时器模式或捕捉模式下，定时器模式寄存器 0n(TOR0n)的 MD0n0 位用于设置是否在计数开始时产生定时器中断。

当 MD0n0 被设置为 1 时，可以通过定时器中断(INTTM0n)的产生来获知计数操作开始时序。

在其他模式下，不控制计数操作开始时的定时器中断和 TO0n 输出。

设置为间隔定时器模式(TOE0n = 1, TOM0n = 0)时的操作示例如图 6-35 所示。

图 6-35. 计数操作开始时的定时器中断和 TO0n 输出的操作示例



当 MD0n0 被设置为 1 时，将在计数操作开始时输出定时器中断(INTTM0n)，TO0n 执行一次交替操作。

当 MD0n0 被设置为 0 时，不在计数操作开始时输出定时器中断(INTTM0n)，TO0n 也不改变。在计数一个周期之后，输出 INTTM0n，TO0n 执行一次交替操作。

备注 n: 通道编号(n = 3, 5, 6)

6.7 定时器阵列单元的单通道操作功能

6.7.1 作为间隔定时器/方波输出的操作

(1) 间隔定时器

定时器阵列单元可用作以固定间隔产生 INTTM0n（定时器中断）的基准定时器。
中断产生周期可以用下述表达式计算。

$$\text{INTTM0n (定时器中断) 的产生周期} = \text{计数时钟的周期} \times (\text{TDR0n 的设置值} + 1)$$

(2) 作为方波输出的操作

TO0n 将在产生 INTTM0n 时执行一次交替操作，并输出占空比为 50% 的方波。
从 TO0n 输出方波的周期和频率可以用下述表达式计算。

$$\bullet \text{ 从 TO0n 输出的方波周期} = \text{计数时钟的周期} \times (\text{TDR0n 的设置值} + 1) \times 2$$

$$\bullet \text{ 从 TO0n 输出的方波频率} = \text{计数时钟的频率} / \{(\text{TDR0n 的设置值} + 1) \times 2\}$$

定时器/计数器寄存器 0n (TCR0n) 在间隔定时器模式时充当递减计数器。

在定时器通道开始寄存器 0 (TS0) 的通道开始触发位 (TS0n, TSH01, TSH03) 被设置为 1 之后的第一个计数时钟，定时器数据寄存器 0n (TDR0n) 的值被载入 TCR0n 寄存器。如果定时器模式寄存器 0n (TMR0n) 的 MD0n0 位在此时被设置为 0，则不输出 INTTM0n，TO0n 输出也不会交替。如果 TMR0n 寄存器的 MD0n0 位被设置为 1，则输出 INTTM0n，并交替 TO0n 输出。

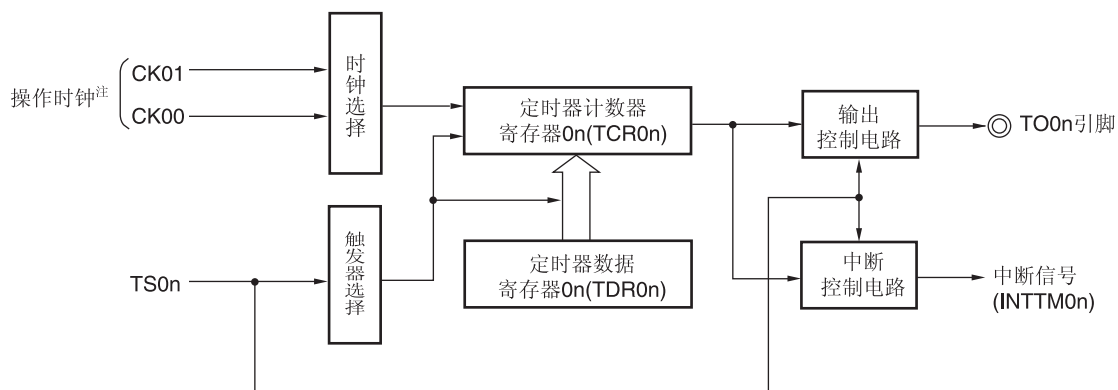
此后，TCR0n 寄存器与计数时钟同步并进行递减计数。

当 TCR0n = 0000H 时，在下一个计数时钟输出 INTTM0n，并交替 TO0n 输出。同时，TDR0n 寄存器的值再次被载入 TCR0n 寄存器。之后重复同样的操作。

可以随时改写 TDR0n 寄存器。从下一个周期开始 TDR0n 寄存器的新值有效。

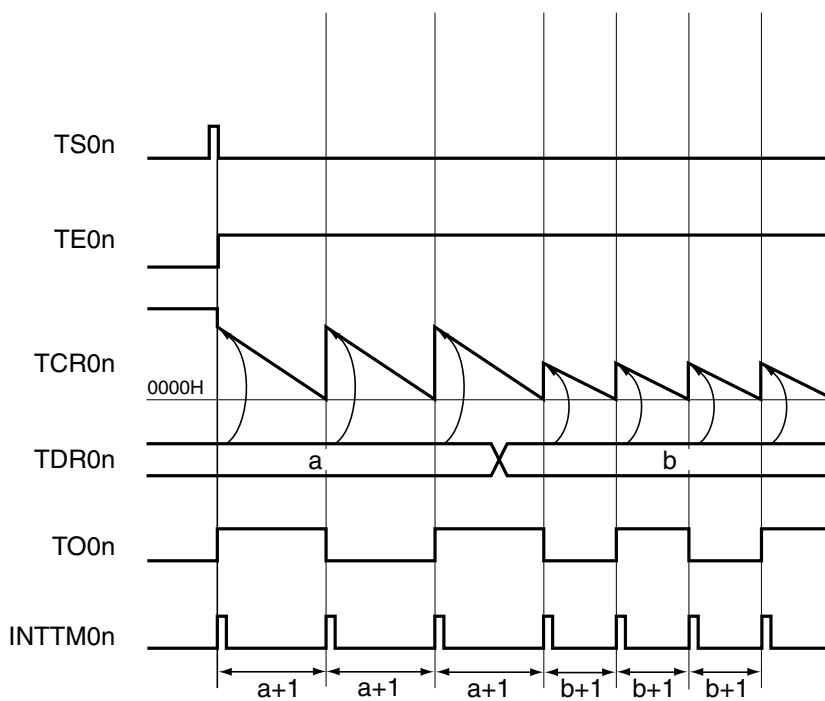
备注 n: 通道编号 (n = 0 至 7)

图 6-36. 作为间隔定时器/方波输出的操作框图



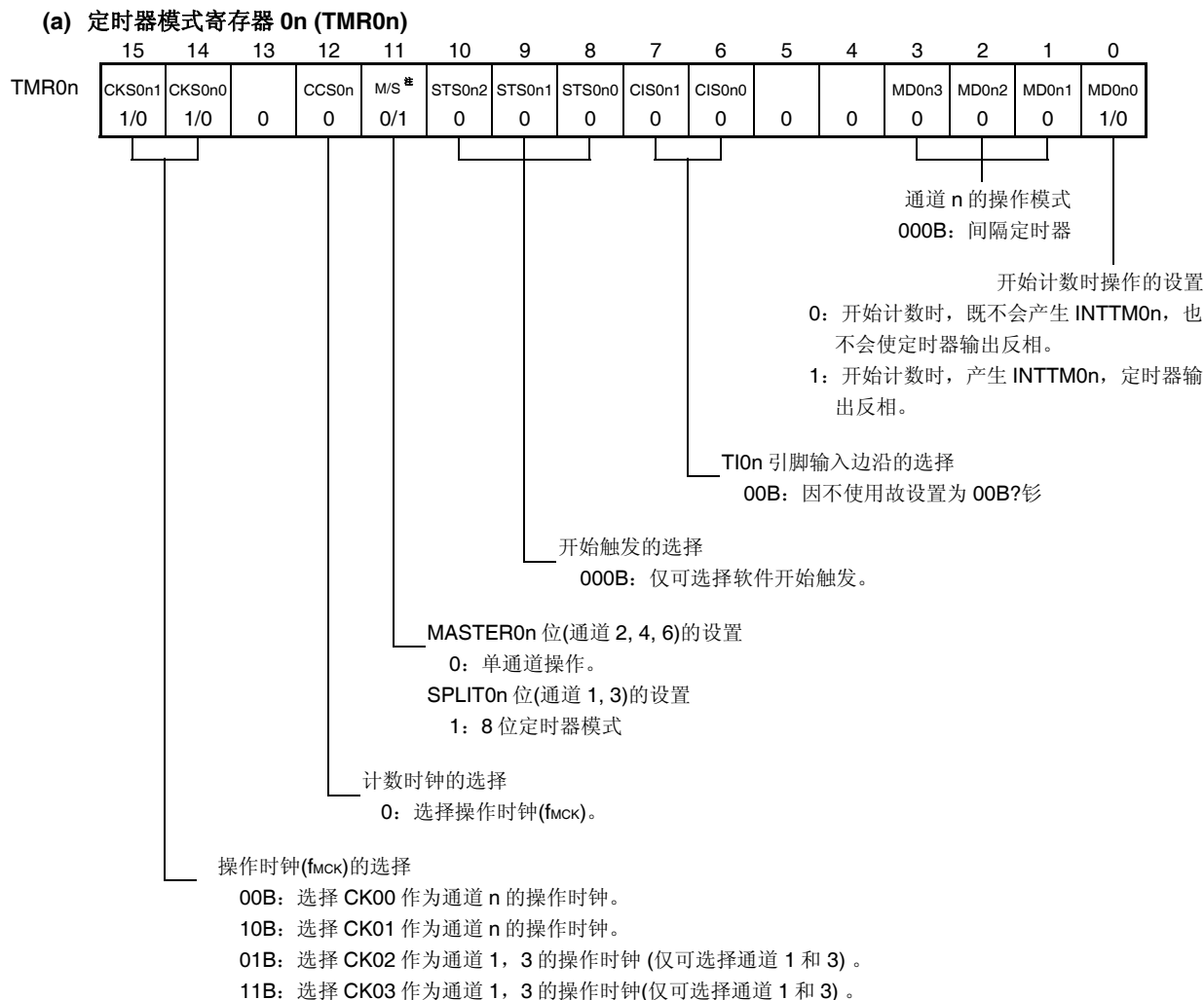
注 对于通道 1 和 3, 可从 CK00、CK01、CK02 和 CK03 中选择时钟。

图 6-37. 作为间隔定时器/方波输出的操作基本时序示例 (MD0n0 = 1)



- 备注1. n: 通道编号 (n = 0 至 7)
2. TS0n: 定时器通道开始寄存器 0 (TS0)的位 n
 TE0n: 定时器通道允许状态寄存器 0 (TE0)的位 n
 TCR0n: 定时器/计数器寄存器 0n (TCR0n)
 TDR0n: 定时器数据寄存器 0n (TDR0n)
 TO0n: TO0n 引脚输出信号

图 6-38. 间隔定时器/方波输出时的寄存器设置内容示例 (1/2)



(b) 定时器输出寄存器 0 (TO0)

	位 n	
TO0	TO0n	0: 由 TO0n 输出 0。
	1/0	1: 由 TO0n 输出 1。

(c) 定时器输出允许寄存器 0 (TOE0)

	位 n	
TOE0	TOE0n	0: 停止通过计数动作来进行 TO0n 输出的操作。
	1/0	1: 允许通过计数动作来进行 TO0n 输出的操作。

注 TMR02, TMR04, TMR06: MASTER0n 位
TMR01, TMR03: SPLIT0n 位
TMR00, TMR05, TMR07: 固定为 0。

备注 n: 通道编号 (n = 0 至 7)

图 6-38. 间隔定时器/方波输出时的寄存器设置内容示例 (2/2)

(d) 定时器输出电平寄存器 0 (TOL0)

TOL0

位 n
TOL0n
0

 0: 当 TOM0n = 0(主通道输出模式)时设置为 0。

(e) 定时器输出模式寄存器 0 (TOM0)

TOM0

位 n
TOM0n
0

 0: 设置主通道输出模式。

备注 n: 通道编号 (n = 0 至 7)

图 6-39. 间隔定时器/方波输出功能时的操作步骤 (1/2)

	软件操作	硬件状态
TAU初始设定		断电状态 (时钟供应被停止, 向各寄存器的写操作被禁止。)
	将外围有效寄存器0(PER0)的TAU0EN位设置为1。	上电状态。每个通道处于停止操作状态。 (开始时钟供应, 允许写入各寄存器。)
	设置定时器时钟选择寄存器0(TPS0)。决定CK00至CK03的时钟频率。	
通道初始设定	设置定时器模式寄存器0n (TMR0n) (决定通道的操作模式)。 将间隔 (周期) 值设置到定时器数据寄存器0n (TDR0n)。	通道处于停止操作状态。 (供应时钟, 并产生一定的功耗。)
	使用TO0n输出时 将定时器输出模式寄存器0(TOM0)的TOM0n位设置为0 (主通道输出模式)。 将TOL0n位设置为0。 设置TO0n位, 并决定TO0n输出的初始电平。	TO0n引脚处于Hi-Z输出状态。 当端口模式寄存器处于输出模式且端口寄存器为0时, 输出TO0n的初始设置电平。
	将TOE0n位设置为1, 并允许操作TO0n。 将端口寄存器和端口模式寄存器设置为0。	因为操作通道处于停止状态, 所以TO0n不会变化。 TO0n引脚输出TO0n 设置电平。
开始操作	(仅当使用TO0n 输出并恢复定时器操作时, 将 TOE0n位设置为1)。 将TS0n(TSH01,TSH03)位设置为1。 TS0n (TSH01, TSH03)位是是触发位, 将自动返回为0。	TE0n (TEH01, TEH03) = 1, 并且开始计数操作。 通过计数时钟输入将TDR0n寄存器的值被载入定时器/计数器寄存器0n (TCR0n)。如果 TMR0n寄存器的MD0n0位被设置为1, 将产生 INTTM0n, 同时, TO0n执行交替操作。
操作期间	不能更改 TMR0n寄存器、TOM0n和TOL0n位的设置值。 可以任意更改 TDR0n寄存器的设置值。 可以任意读取TCR0n寄存器。 不使用TSR0n寄存器。 可以更改TO0和TOE0寄存器的设置值。	计数器(TCR0n)递减计数。当计数值达到0000H时, TDR0n寄存器的值再次被载入TCR0n寄存器, 继续计数操作。检测到TCR0n = 0000H时, 产生 INTTM0n, TO0n执行交替操作。之后重复以上操作。
停止操作	将TT0n(TTH01,TTH03)位设置为1。 TT0n (TTH01, TTH03)位是触发位, 将自动返回为0。	TE0n (TEH01, TEH03) = 0, 并且停止计数操作。 TCR0n寄存器保持计数值并停止计数操作。 TO0n输出不被初始化, 而是保持当前状态。
	将TOE0n位清除为0, 并对TO0n位设置。	TO0n引脚输出TO0n位的设置电平。

恢复操作。

(备注如下页所示。)

图 6-39. 间隔定时器/方波输出功能时的操作步骤 (2/2)

	软件操作	硬件状态
TAU 停止	保持TO0n引脚输出电平时 将希望保持的值设置到端口寄存器后，将TO0n位清除为0。	TO0n引脚输出电平由端口功能保持。
	不需要保持 TO0n引脚输出电平时 将端口模式寄存器切换至输入模式。	TO0n引脚输出电平处于Hi-Z输出状态。
	将PER0寄存器的TAU0EN位设置为0。	断电状态 初始化所有电路，同时初始化各通道的SFR。 (TO0n位变为0，并且TO0n引脚变为端口模式。)

备注 n: 通道编号 (n = 0 至 7)

6.7.2 作为外部事件计数器操作

定时器阵列单元作为外部事件计数器使用时，用于对检测到的 $Tl0n$ 引脚输入有效边沿（外部事件）的次数进行计数。达到指定计数次数时，事件计数器产生中断。指定的计数次数可以用下述表达式计算。

$$\text{指定计数次数} = \text{TDR0n 的设置值} + 1$$

定时器/计数器寄存器 $0n$ (TCR0n)在事件计数器模式时充当递减计数器。

通过设置定时器通道开始寄存器 0 (TS0)的任意通道开始触发位(TS0n, TSH03)为 1 ，可以把定时器数据寄存器 $0n$ (TDR0n)的值载入 TCR0n 寄存器。

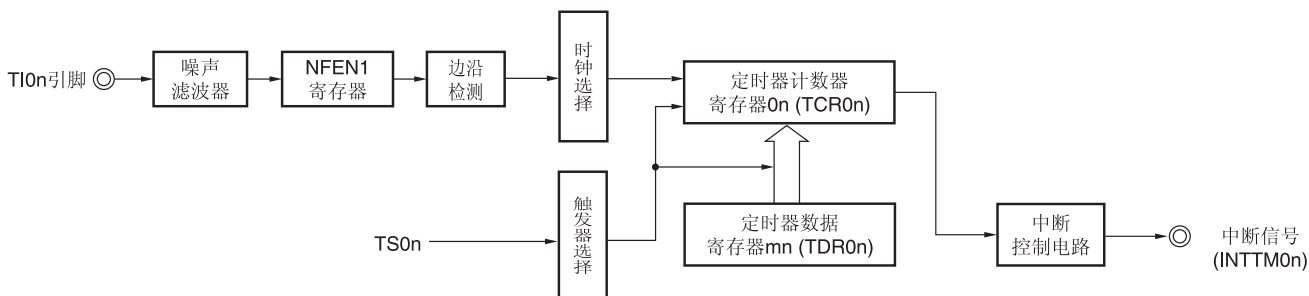
每当检测到 $Tl0n$ 引脚输入有效边沿时，TCR0n 寄存器进行递减计数。当 $TCR0n = 0000H$ 时，TCR0n 寄存器将再次载入 TDR0n 寄存器的值，并输出 INTTM0n。

之后重复以上操作。

TO0n 引脚会根据外部事件而输出不规则的波形。所以请将定时器输出允许寄存器 0 (TOE0)的 TOE0n 位设置为 0 以停止输出。

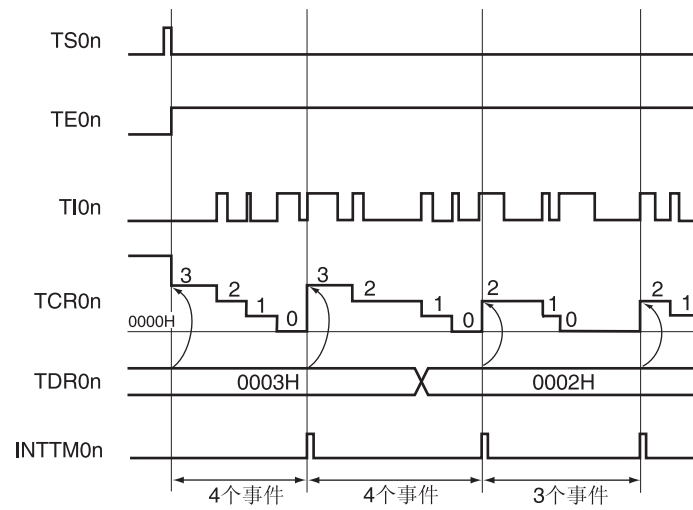
可以随时改写 TDR0n 寄存器。从下一个计数周期开始 TDR0n 寄存器的新值有效。

图 6-40. 作为外部事件计数器的操作框图



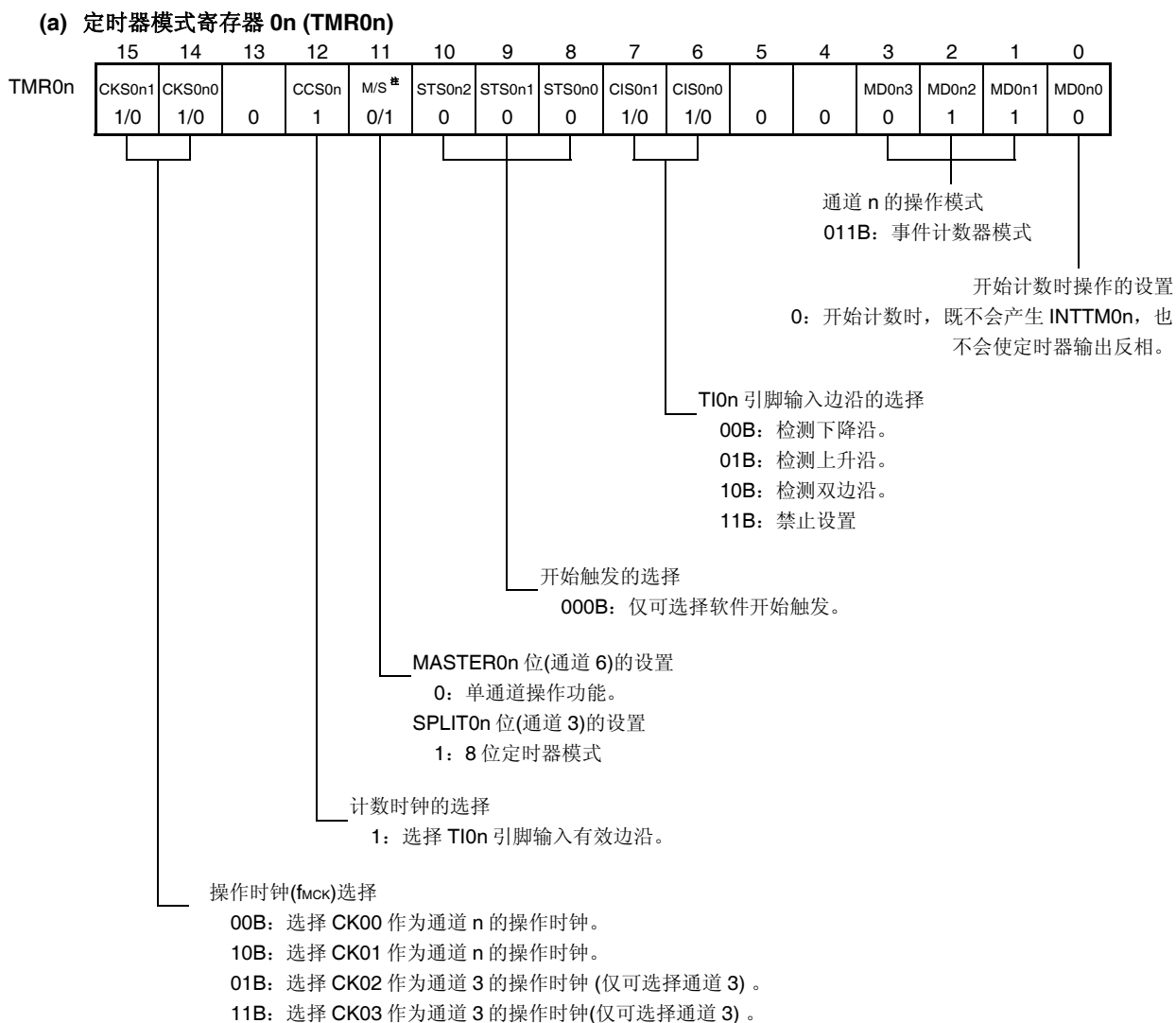
备注 n: 通道编号(n = 3, 5, 6)

图 6-41. 作为外部事件计数器的操作基本时序示例

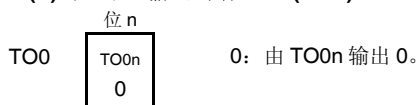


- 备注 1. n: 通道编号(n = 3, 5, 6)
2. TS0n: 定时器通道开始寄存器 0 (TS0)的位 n
- TE0n: 定时器通道允许状态寄存器 0 (TE0)的位 n
- TI0n: TI0n 引脚输入信号
- TCR0n: 定时器/计数器寄存器 0n (TCR0n)
- TDR0n: 定时器数据寄存器 0n (TDR0n)

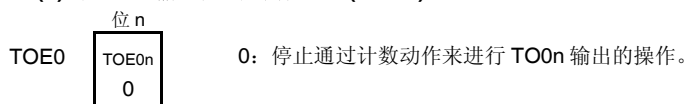
图 6-42. 外部事件计数器模式时的寄存器设置内容示例 (1/2)



(b) 定时器输出寄存器 0 (TO0)



(c) 定时器输出允许寄存器 0 (TOE0)



注 TMR06: MASTER06 位
TMR03: SPLIT03 位
TMR00, TMR05, TMR07: 固定为 0。

备注 n: 通道编号(n = 3, 5, 6)

图 6-42. 外部事件计数器模式时的寄存器设置内容示例 (2/2)

(d) 定时器输出电平寄存器 0 (TOL0)

TOL0

位 n
TOL0n
0

 0: 当 TOM0n = 0(主通道输出模式)时清除为 0。

(e) 定时器输出模式寄存器 0 (TOM0)

TOM0

位 n
TOM0n
0

 0: 设置主通道输出模式。

备注 n: 通道编号(n = 3, 5, 6)

图 6-43. 使用外部事件计数器功能时的操作步骤

	软件操作	硬件状态
TAU初始 设定		断电状态 (时钟供应被停止, 向各寄存器的写操作被禁止。)
	将外围有效寄存器0(PER0)的TAU0EN位设置为1。	上电状态。每个通道处于停止操作状态。 (开始时钟供应, 允许写入各寄存器。)
	设置定时器时钟选择寄存器0(TPS0)。 决定CK00至CK03的时钟频率。	
通道初始 设定	设置定时器模式寄存器0n (TMR0n) (决定通道的操作模式)。 将计数次数设置到定时器数据寄存器0n (TDR0n)。 将定时器输出允许寄存器0 (TOE0)的TOE0n位清除为0。	通道处于停止操作状态。 (供应时钟, 并产生一定的功耗。)
开始操作	将TS0n (TSH03)位设置为1。 TS0n (TSH03)位是触发位, 将自动返回为 0。	TE0n (TEH03) = 1, 并且开始计数操作。 TDR0n寄存器的值被载入定时器/计数器寄存器0n (TCR0n), 进入TI0n引脚输入边沿的等待检测状态。
操作期间	可以任意更改TDR0n寄存器的设置值。 将噪声滤波器允许寄存器1 (NFEN1)的对应位设置为1。 可以任意读取TCR0n寄存器。 不使用TSR0n寄存器。 不可更改TMR0n寄存器、TOM0n、TOL0n、TO0n和TOE0n位的设置值。	每当检测到TI0n引脚的输入边沿时, 计数器(TCR0n)都会递减计数。当计数值达到0000H时, TDR0n寄存器的值再次被载入TCR0n寄存器, 继续计数操作。检测到TCR0n = 0000H时输出INTTM0n。 之后重复以上操作。
停止操作	将TT0n (TTH03)位设置为1。 TT0n (TTH03)位是触发位, 将自动返回为0。	TE0n (TEH03) = 0, 并且停止计数操作。 TCR0n寄存器保持计数值并停止计数操作。
TAU 停止	将PER0寄存器的TAU0EN位设置为0。	断电状态 初始化所有电路, 同时初始化各通道的SFR。

恢复操作。

备注 n: 通道编号(n = 3, 5, 6)

6.7.3 作为输入脉冲间隔测量的操作

可以在 TI0n 有效边沿捕捉计数值，并可测量输入 TI0n 的脉冲的间隔。

脉冲间隔可以用下述表达式计算。

$$\text{TI0n 输入脉冲间隔} = \text{计数时钟的周期} \times ((10000\text{H} \times \text{TSR0n: OVF}) + (\text{TDR0n 的捕捉值} + 1))$$

注意事项 使用定时器模式寄存器 0n (TMR0n)的 CKS0n 位选择的操作时钟对 TI0n 引脚输入进行采样，因此会发生最大为一个操作时钟周期的误差。

定时器/计数器寄存器 0n (TCR0n)在捕捉模式时充当递增计数器。

在定时器通道开始寄存器 0 (TS0)的通道开始触发位(TS0n)被设置为 1 时，TCR0n 寄存器将与计数时钟同步从 0000H 开始递增计数。

检测到 TI0n 引脚输入有效边沿时，TCR0n 寄存器的计数值被传送（捕捉）至定时器数据寄存器 0n (TDR0n)，同时，TCR0n 寄存器被清除为 0000H，并输出 INTTM0n。此时如果计数器溢出，则定时器状态寄存器 0n (TSR0n)的 OVF 位被设置为 1。如果计数器不溢出，则 OVF 位被清除。之后重复以上操作。

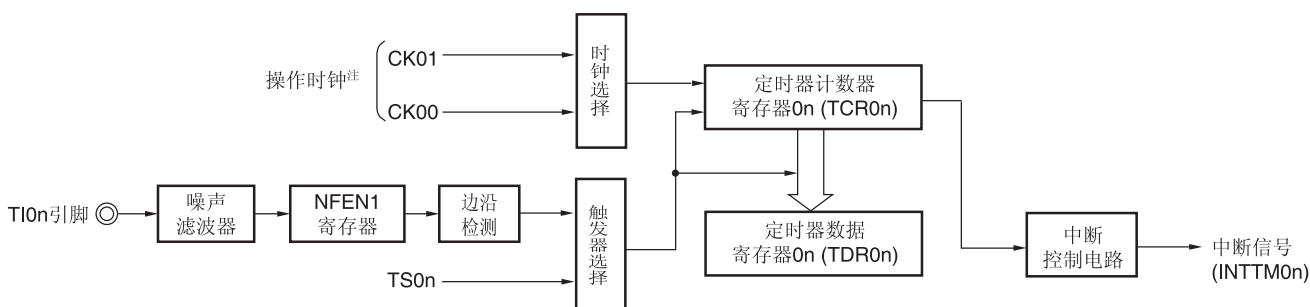
计数值被捕捉至 TDR0n 寄存器时，依据测量期间计数器是否发生溢出，TSR0n 寄存器的 OVF 位将被更新。从而可以确认捕捉值的溢出状态。

如果计数器进行了两个以上周期的完整计数，也能发生溢出，TSR0n 寄存器的 OVF 位可被设置为 1。然而，在发生两次以上的溢出时，不能通过 OVF 位测量出正常间隔值。

通过将 TMR0n 寄存器的 STS0n2 至 STS0n0 位设置为 001B，TI0n 的有效边沿可以用作开始触发和捕捉触发。

当 TE0n = 1 时，可以使用软件操作 (TS0n = 1) 来代替 TI0n 引脚输入作为捕捉触发。

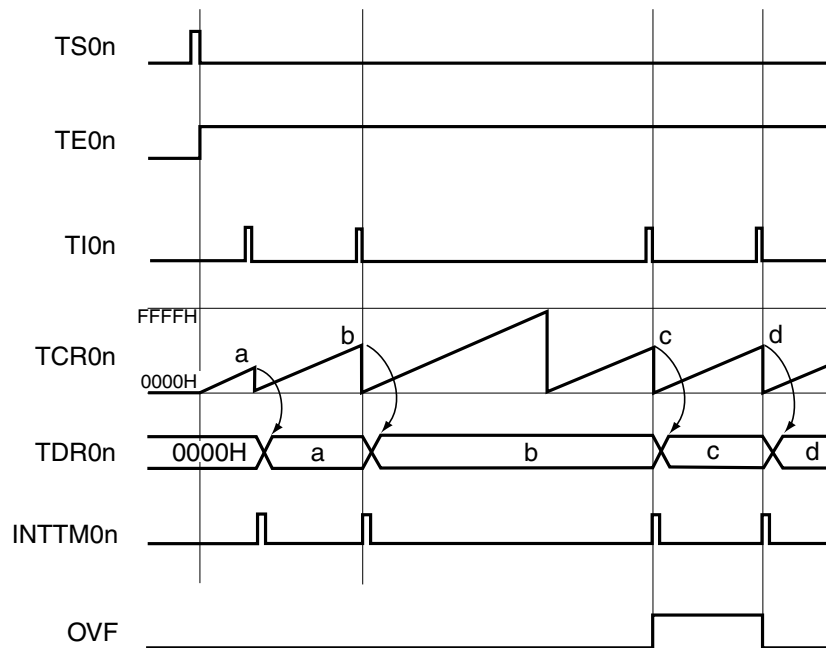
图 6-44. 作为输入脉冲间隔测量的操作框图



注 对于通道 3，可从 CK00、CK01、CK02 和 CK03 中选择时钟。

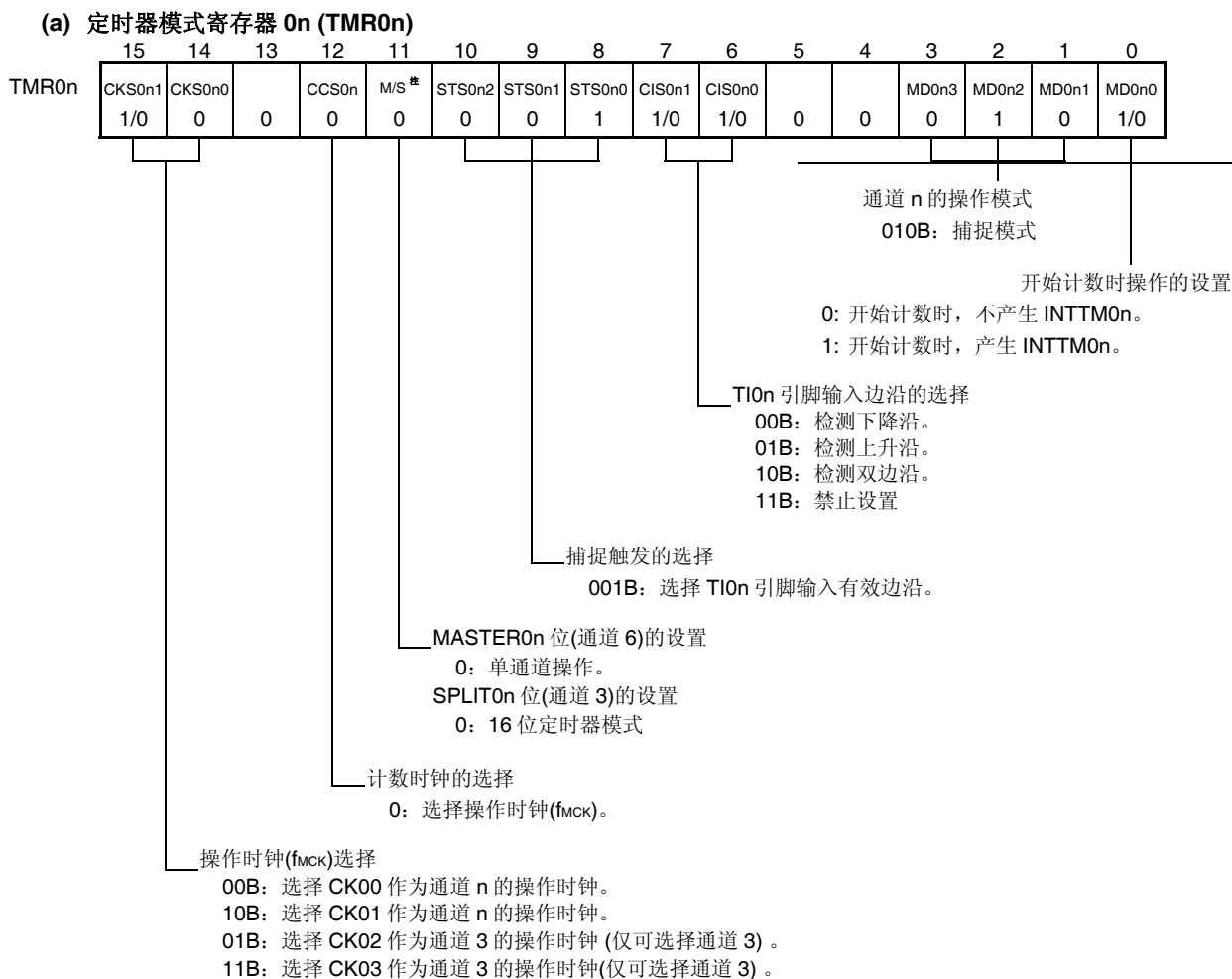
备注 n: 通道编号(n = 3, 5, 6)

图 6-45. 作为输入脉冲间隔测量的操作基本时序示例 (MD0n0 = 0)

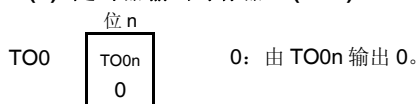


- 备注 1. n: 通道编号(n = 3, 5, 6)
2. TS0n: 定时器通道开始寄存器 0 (TS0)的位 n
 TE0n: 定时器通道允许状态寄存器 0 (TE0)的位 n
 TI0n: TI0n 引脚输入信号
 TCR0n: 定时器/计数器寄存器 0n (TCR0n)
 TDR0n: 定时器数据寄存器 0n (TDR0n)
 OVF: 定时器状态寄存器 0n (TSR0n)的位 0

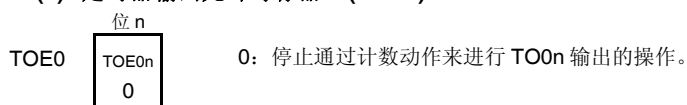
图 6-46. 输入脉冲间隔测量时的寄存器设置内容示例



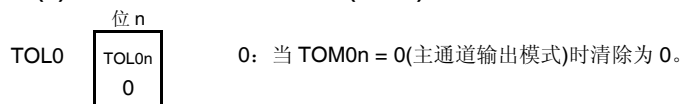
(b) 定时器输出寄存器 0 (TO0)



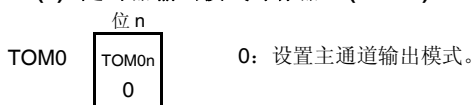
(c) 定时器输出允许寄存器 0 (TOE0)



(d) 定时器输出电平寄存器 0 (TOL0)



(e) 定时器输出模式寄存器 0 (TOM0)



注 TMR06: MASTER0n 位

TMR03: SPLIT0n 位

TMR05: 固定为 0。

备注 n: 通道编号 (n = 3, 5, 6)

图 6-47. 使用输入脉冲间隔测量功能时的操作步骤

	软件操作	硬件状态
TAU初始 设定		断电状态 (时钟供应被停止, 向各寄存器的写操作被禁止。)
	将外围有效寄存器0(PER0)的TAU0EN位设置为1。	上电状态。每个通道处于停止操作状态。 (开始时钟供应, 允许写入各寄存器。)
	设置定时器时钟选择寄存器0(TPS0)。 决定CK00至CK03的时钟频率。	
通道初始 设定	设置定时器模式寄存器0n (TMR0n) (决定通道的操作模式)。	通道处于停止操作状态。 (供应时钟, 并产生一定的功耗。)
开始操作	将TS0n位设置为1。 TS0n位是触发位, 将自动返回为0。	TE0n = 1, 并且开始计数操作。 输入计数时钟时, 定时器/计数器寄存器0n (TCR0n)被清除为0000H。 当TMR0n寄存器的MD0n0位为1时, 产生INTTM0n中断。
操作期间	仅可以任意更改TMR0n寄存器的CIS0n1和CIS0n0位的设置值。 将噪声滤波器允许寄存器1 (NFEN1)的对应位设置为1。 可以任意读取TDR0n寄存器。 可以任意读取TCR0n寄存器。 可以任意读取TSR0n寄存器。 不能更改TOM0n、TOL0n、TO0n和TOE0n位的设置值。	计数器(TCR0n)从0000H开始递增计数。当检测到TI0n引脚输入有效边沿时, 计数值被传送(捕捉)至定时器数据寄存器0n (TDR0n)。同时, TCR0n寄存器被清除为0000H, 并产生INTTM0n信号。 此时如果发生溢出, 则定时器状态寄存器0n (TSR0n)的OVF位被设置; 如果不发生溢出, 则OVF位被清除。 之后重复以上操作。
停止操作	将TT0n位设置为1。 TT0n位是触发位, 将自动返回为0。	TE0n = 0, 并且停止计数操作。 TCR0n寄存器保持计数值并停止计数操作。 同时保持TSR0n寄存器的OVF位。
TAU 停止	将PER0寄存器的TAU0EN位设置为0。	断电状态 初始化所有电路, 同时初始化各通道的SFR。

恢复操作。

备注 n: 通道编号(n = 3, 5, 6)

6.7.4 输入信号高/低电平宽度测量的操作

注意事项 当使用通道执行支持 LIN-bus 和 DMX512 的功能时，请自行将以下说明中的 TI0n 替换为 RxD0。

通过在 TI0n 引脚输入的一个边沿开始计数，在另一个边沿捕捉计数次数，可以测量 TI0n 的信号宽度（高电平宽度/低电平宽度）。TI0n 的信号宽度可用下述表达式计算。

$$\text{TI0n 输入的信号宽度} = \text{计数时钟的周期} \times ((10000\text{H} \times \text{TSR0n: OVF}) + (\text{TDR0n 的捕捉值} + 1))$$

注意事项 使用定时器模式寄存器 0n (TMR0n)的 CKS0n 位选择的操作时钟对 TI0n 引脚输入进行采样，因此会发生一个操作时钟周期的误差。

定时器/计数器寄存器 0n (TCR0n)在捕捉模式和单计数模式时充当递增计数器。

当定时器通道开始寄存器 0 (TS0)的通道开始触发位(TS0n)被设置为 1 时，TE0n 位将被设置为 1，同时 TI0n 引脚被设置为开始边沿检测等待状态。

当检测到 TI0n 引脚输入开始边沿（测量高电平宽度时为 TI0n 引脚输入的上升沿）时，计数器与计数时钟同步并从 0000H 开始递增计数。之后检测到有效捕捉边沿（测量高电平宽度时为 TI0n 引脚输入的下降沿）时，计数值被传送至定时器数据寄存器 0n (TDR0n)，同时输出 INTTM0n。此时如果计数器溢出，则定时器状态寄存器 0n (TSR0n)的 OVF 位被设置为 1。如果计数器不溢出，则 OVF 位被清除。当值达到“传送至 TDR0n 寄存器的值 + 1”时，停止 TCR0n 寄存器，同时设置为 TI0n 引脚进入开始边沿检测等待状态。之后重复以上操作。

计数值被捕捉至 TDR0n 寄存器时，依据测量期间计数器是否发生溢出，TSR0n 寄存器的 OVF 位将被更新。从而可以确认捕捉值的溢出状态。

如果计数器进行了两个以上周期的完整计数，也能发生溢出，TSR0n 寄存器的 OVF 位被设置为 1。然而，在发生两次以上的溢出时，不能通过 OVF 位测量出正常间隔值。

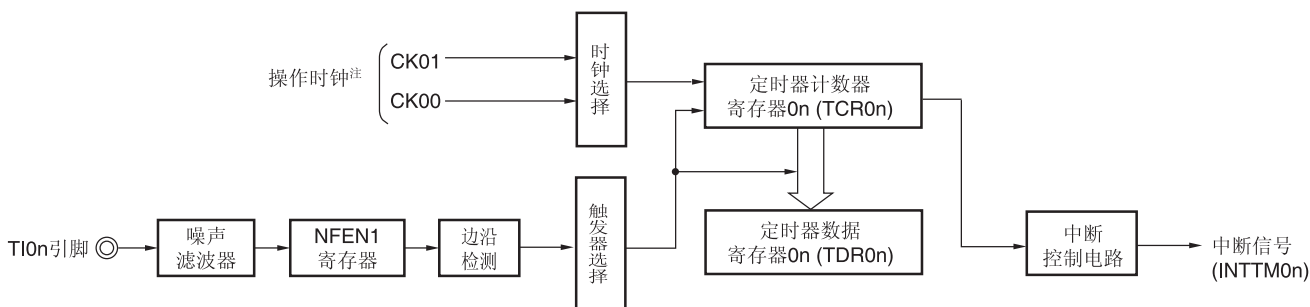
可以通过 TMR0n 寄存器的 CIS0n1 和 CIS0n0 位来选择测量 TI0n 引脚的高电平宽度或低电平宽度。

由于该功能用于测量 TI0n 引脚输入的信号宽度，因此，在 TE0n 位为 1 时，不能将 TS0n 位设置为 1。

TMR0n 寄存器的 CIS0n1, CIS0n0 = 10B: 测量低电平宽度。

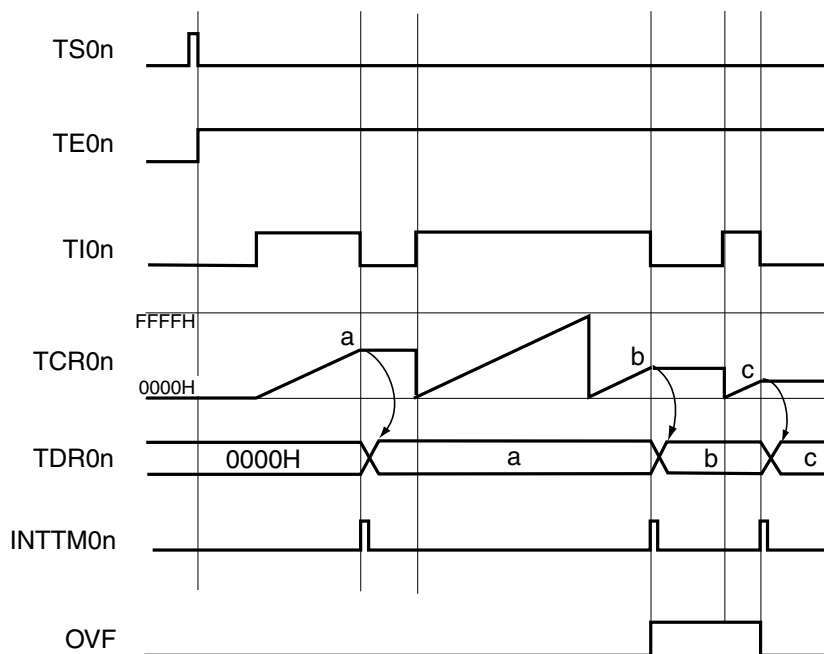
TMR0n 寄存器的 CIS0n1, CIS0n0 = 11B: 测量高电平宽度。

图 6-48. 作为输入信号的高/低电平宽度测量的操作框图



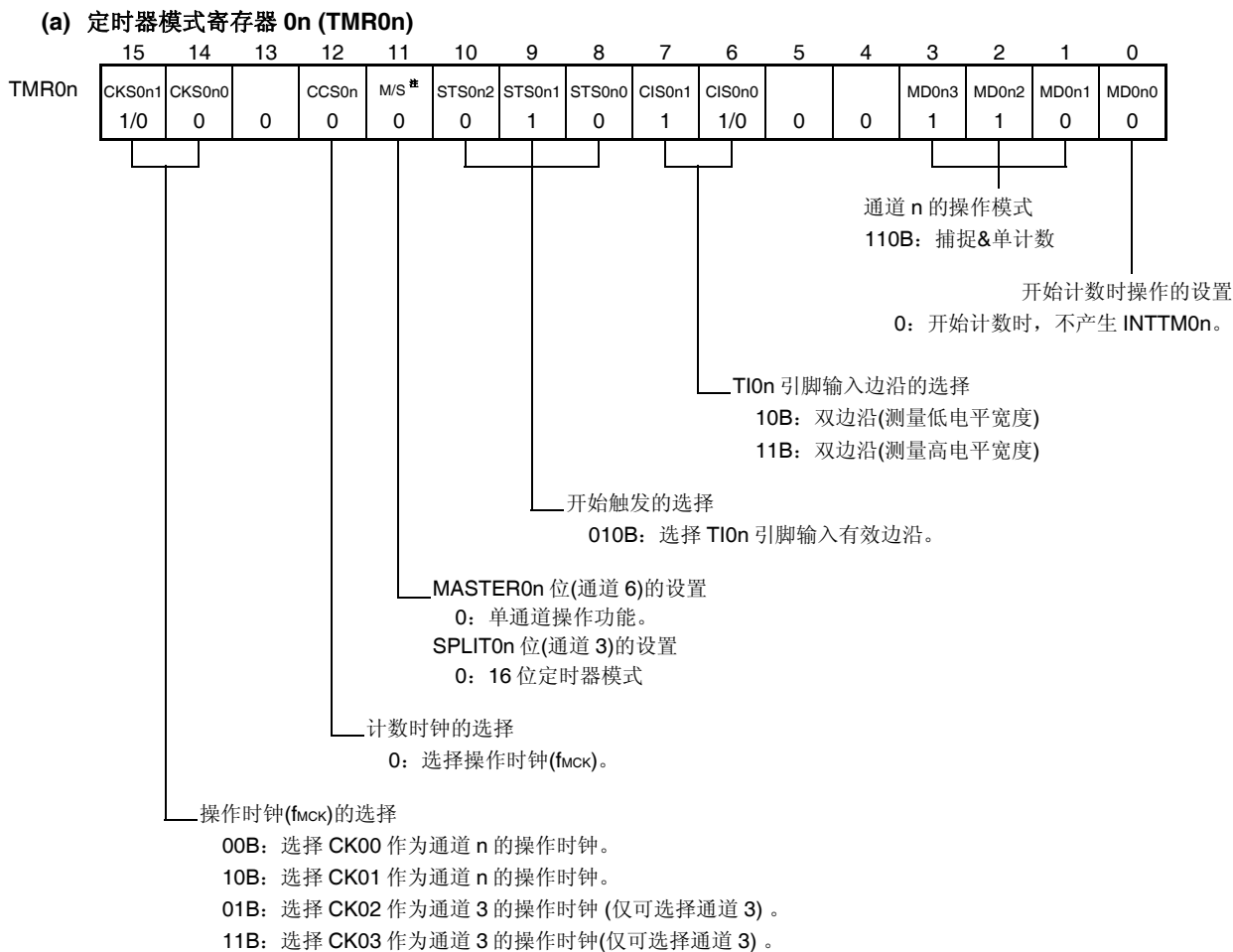
注 对于通道 3，可从 CK00、CK01、CK02 和 CK03 中选择时钟。

图 6-49. 作为输入信号的高/低电平宽度测量的操作基本时序示例

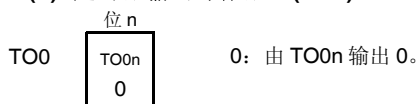


- 备注 1. n: 通道编号(n = 3, 5, 6)
2. TS0n: 定时器通道开始寄存器 0 (TS0)的位 n
 TE0n: 定时器通道允许状态寄存器 0 (TE0)的位 n
 TI0n: TI0n 引脚输入信号
 TCR0n: 定时器/计数器寄存器 0n (TCR0n)
 TDR0n: 定时器数据寄存器 0n (TDR0n)
 OVF: 定时器状态寄存器 0n (TSR0n)的位 0

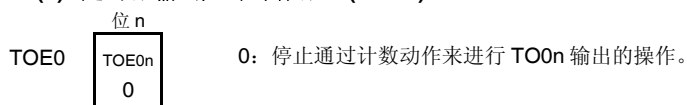
图 6-50. 输入信号的高低电平宽度测量时的寄存器设置内容示例



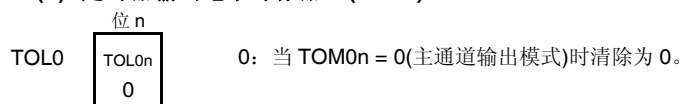
(b) 定时器输出寄存器 0 (TO0)



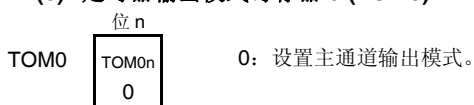
(c) 定时器输出允许寄存器 0 (TOE0)



(d) 定时器输出电平寄存器 0 (TOL0)



(e) 定时器输出模式寄存器 0 (TOM0)



注 TMR06: MASTER06 位
 TMR03: SPLIT03 位
 TMR05: 固定为 0。

备注 n: 通道编号(n = 3, 5, 6)

图 6-51. 使用输入信号高/低电平宽度测量功能时的操作步骤

	软件操作	硬件状态
TAU初始 设定		断电状态 (时钟供应被停止, 向各寄存器的写操作被禁止。)
	将外围有效寄存器0(PER0)的TAU0EN位设置为1。	上电状态。每个通道处于停止操作状态。 (开始时钟供应, 允许写入各寄存器。)
	设置定时器时钟选择寄存器0(TPS0)。 决定CK00至CK03的时钟频率。	
通道初始 设定	设置定时器模式寄存器0n (TMR0n) (决定通道的操作模式)。 将TOE0n位设置为0, 并且停止操作TO0n。	通道处于停止操作状态。 (供应时钟, 并产生一定的功耗。)
开始操作	将TS0n位设置为1。 TS0n位是触发位, 将自动返回为0。	TE0n = 1, 变为TI0n引脚开始边沿检测等待状态。
	检测TI0n引脚输入的计数开始有效边沿。	将定时器/计数器寄存器0n (TCR0n) 清除为0000H并开始递增计数。
操作期间	可以任意读取 TDR0n寄存器的设置值。 可以任意读取TCR0n寄存器。 将噪声滤波器允许寄存器1 (NFEN1)的对应位设置为1。不使用TSR0n寄存器。 不可更改TMR0n寄存器、TOM0n、TOL0n、TO0n和TOE0n位的设置值。	当检测到TI0n引脚开始边沿时, 计数器(TCR0n)从0000H开始递增计数。如果检测到TI0n引脚的捕捉边沿, 则计数值将被传送到定时器数据寄存器0n (TDR0n), 并产生INTTM0n。 此时如果发生溢出, 则定时器状态寄存器0n (TSR0n)的OVF位被设置; 如果不发生溢出, 则OVF位被清除。停止TCR0n寄存器计数操作, 直至检测到下一个TI0n引脚开始边沿为止。
停止操作	将TT0n位设置为1。 TT0n位是触发位, 将自动返回为0。	TE0n = 0, 并且停止计数操作。 TCR0n寄存器保持计数值并停止计数操作。 同时保持TSR0n寄存器的OVF位。
TAU 停止	将PER0寄存器的TAU0EN位设置为0。	断电状态 初始化所有电路, 同时初始化各通道的SFR。

恢复操作。

备注 n: 通道编号(n = 3, 5, 6)

6.7.5 作为延迟计数器的操作

当检测到 TI0n 引脚输入的有效边沿（外部事件）时，可以开始递减计数，并在任意指定间隔后产生 INTTM0n（定时器中断）。

也可通过软件设置 TS0n = 1，以任意间隔产生 INTTM0n（定时器中断），并在 TE0n 期间开始递减计数。中断产生周期可以用下述表达式计算。

$$\text{INTTM0n (定时器中断) 的生成周期} = \text{计数时钟的周期} \times (\text{TDR0n 的设置值} + 1)$$

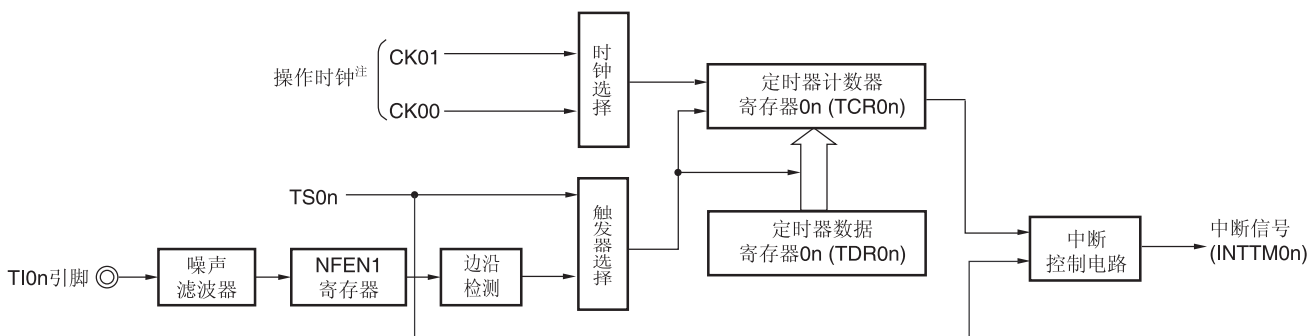
定时器/计数器寄存器 0n (TCR0n) 在单计数模式时充当递减计数器。

当定时器通道开始寄存器 0 (TS0) 的通道开始触发位 (TS0n, TSH03) 被设置为 1 时，TE0n、TEH03 位将被设置为 1，同时 TI0n 引脚变为输入有效边沿检测等待状态。

检测到 TI0n 引脚输入有效边沿时定时器/计数器寄存器 0n (TCR0n) 开始动作，并载入定时器数据寄存器 0n (TDR0n) 的值。TCR0n 寄存器与计数时钟同步，从载入的 TDR0n 寄存器的值开始递减计数。当 TCR0n = 0000H 时，输出 INTTM0n 并停止计数，直至检测到下一个 TI0n 引脚输入有效边沿为止。

可以随时改写 TDR0n 寄存器。从下一个周期开始 TDR0n 寄存器的新值有效。

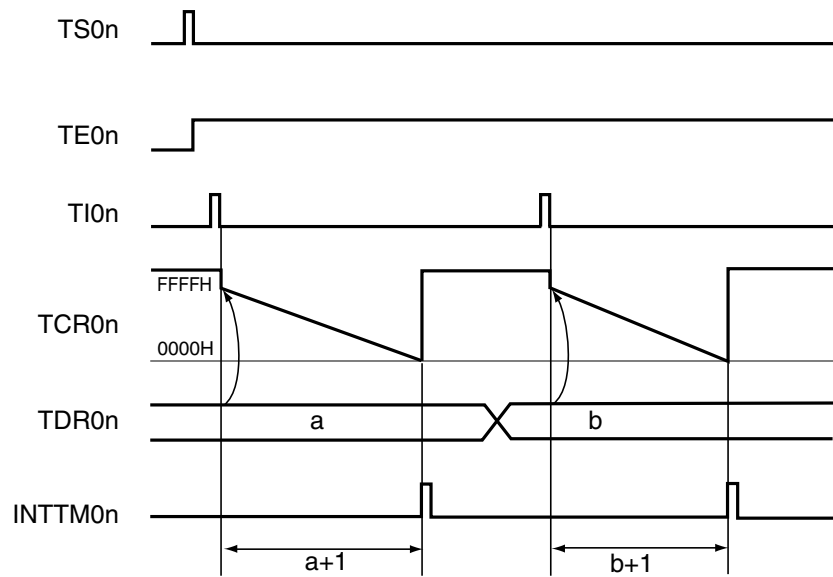
图 6-52. 作为延迟计数器的操作框图



注 使用通道 3 时，可从 CK00、CK01、CK02 和 CK03 中选择时钟。

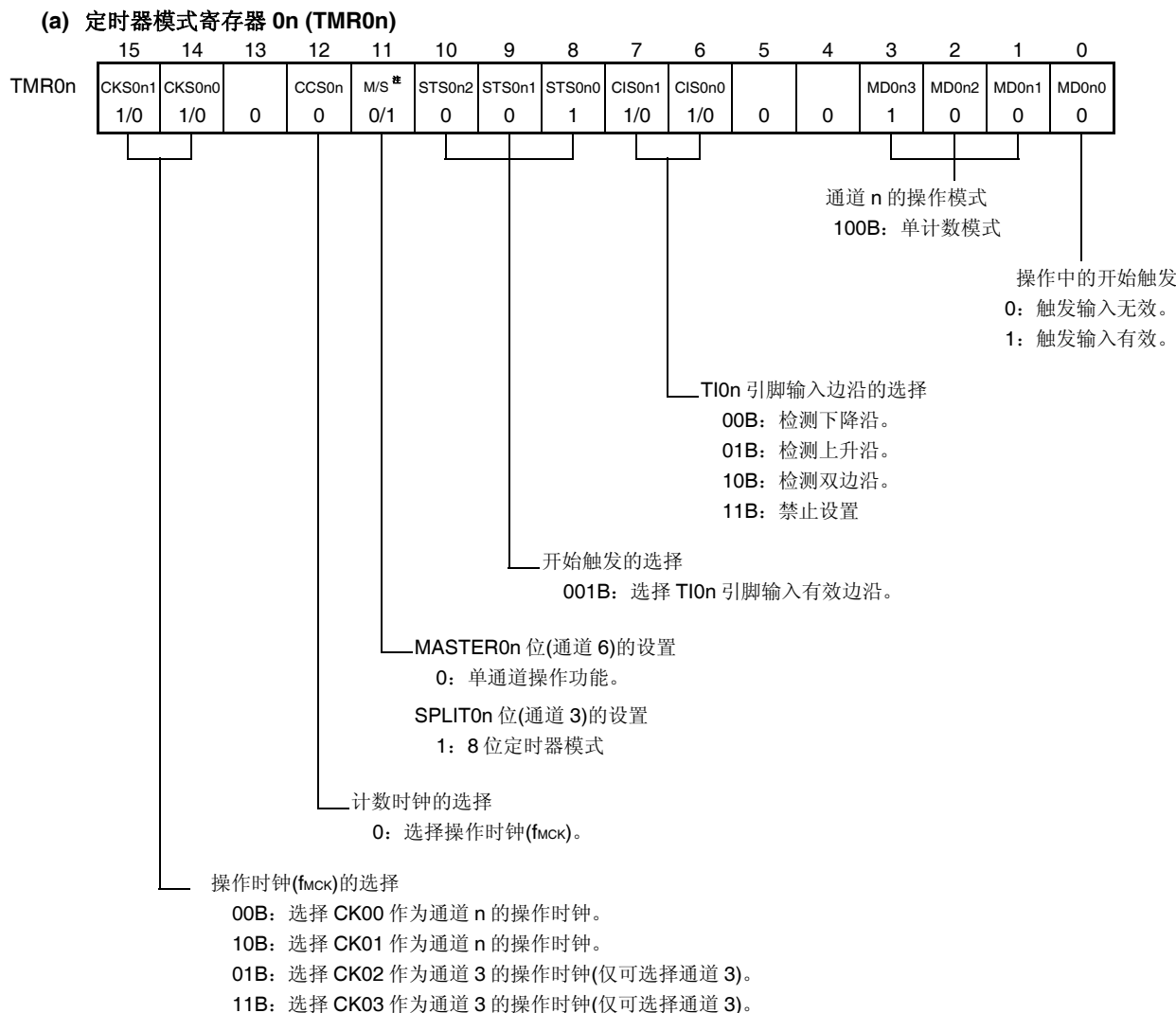
备注 n: 通道编号 (n = 3, 5, 6)

图 6-53. 作为延迟计数器的操作基本时序示例

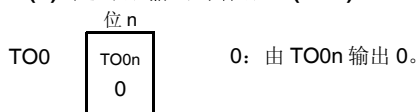


- 备注 1. n: 通道编号(n = 3, 5, 6)
2. TS0n: 定时器通道开始寄存器 0 (TS0)的位 n
 TE0n: 定时器通道允许状态寄存器 0 (TE0)的位 n
 TI0n: TI0n 引脚输入信号
 TCR0n: 定时器/计数器寄存器 0n (TCR0n)
 TDR0n: 定时器数据寄存器 0n (TDR0n)

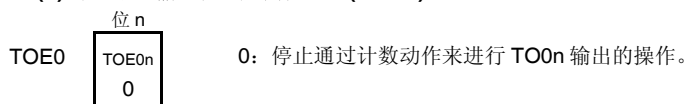
图 6-54. 使用延迟计数器功能时的寄存器设置内容示例 (1/2)



(b) 定时器输出寄存器 0 (TO0)



(c) 定时器输出允许寄存器 0 (TOE0)



注 TMR06: MASTER06 位

TMR03: SPLIT03 位

TMR05: 固定为 0。

备注 n: 通道编号(n = 3, 5, 6)

图 6-54. 使用延迟计数器功能时的寄存器设置内容示例 (2/2)

(d) 定时器输出电平寄存器 0 (TOL0)

TOL0

位 n
TOL0n
0

 0: 当 TOM0n = 0(主通道输出模式)时清除为 0。

(e) 定时器输出模式寄存器 0 (TOM0)

TOM0

位 n
TOM0n
0

 0: 设置主通道输出模式。

备注 n: 通道编号(n = 3, 5, 6)

图 6-55. 使用延迟计数器功能时的操作步骤

	软件操作	硬件状态
TAU初始 设定		断电状态 (时钟供应被停止, 向各寄存器的写操作被禁止。)
	将外围有效寄存器0(PER0)的TAU0EN位设置为1。	上电状态。每个通道处于停止操作状态。 (开始时钟供应, 允许写入各寄存器。)
	设置定时器时钟选择寄存器0(TPS0)。 决定CK00至CK03的时钟频率。	
通道初始 设定	设置定时器模式寄存器0n (TMR0n) (决定通道的操作模式)。 将 INTTM0n输出延迟设置到定时器数据寄存器0n (TDR0n)。 将TOE0n位设置为0, 并且停止操作TO0n。	通道处于停止操作状态。 (供应时钟, 并产生一定的功耗。)
开始操作	将TS0n (TSH03)位设置为1。 TS0n (TSH03)位是触发位, 将自动返回为0。	TE0n (TEH03) = 1, TI0n引脚变为输入有效边沿检测等待状态。
	检测TI0n引脚输入的有效边沿。	将TDR0n寄存器的值载入定时器/计数器寄存器0n (TCR0n)。
操作期间	可以任意更改TDR0n寄存器的设置值。 将噪声滤波器允许寄存器1 (NFEN1)的对应位设置为1。 可以任意读取TCR0n寄存器。 不使用TSR0n寄存器。	计数器(TCR0n)递减计数。当TCR0n递减计数至0000H时, 输出INTTM0n并停止计数 (使TCR0n保持0000H) 直到下一个TI0n引脚输入为止。
停止操作	将TT0n (TTH03)位设置为1。 TT0n (TTH03)位是触发位, 将自动返回为0。	TE0n (TEH03) = 0, 并且停止计数操作。 TCR0n寄存器保持计数值并停止计数操作。
TAU 停止	将PER0寄存器的TAU0EN位设置为0。	断电状态 初始化所有电路, 同时初始化各通道的SFR。

恢复操作。

备注 n: 通道编号(n = 3, 5, 6)

6.8 定时器阵列单元的多通道联动操作功能

6.8.1 作为单触发脉冲输出功能操作

通过联合使用两个通道，可以从输入 TI0n 引脚的信号产生一个任意延迟脉冲宽度的单触发脉冲。延迟时间和脉冲宽度可用下述表达式计算。

$\begin{aligned} \text{延迟时间} &= \{\text{TDR0n (主) 的设置值} + 2\} \times \text{计数时钟周期} \\ \text{脉冲宽度} &= \{\text{TDR0p (从属) 的设置值}\} \times \text{计数时钟周期} \end{aligned}$

主通道工作于单计数模式，并对延迟计数。主通道的定时器/计数器寄存器 0n (TCR0n) 在检测到开始触发时开始工作，并载入定时器数据寄存器 0n (TDR0n) 的值。

TCR0n 寄存器与计数时钟同步，从载入的 TDR0n 寄存器的值开始递减计数。当 TCR0n = 0000H 时，输出 INTTM0n 并停止计数，直至检测到下一个开始触发为止。

从属通道工作于单计数模式，并对脉冲宽度计数。将主通道的 INTTM0n 作为开始触发，从属通道的 TCR0p 寄存器开始工作，并载入 TDR0p 寄存器的值。TCR0p 寄存器与计数时钟同步，从载入的 TDR0p 寄存器值开始递减计数。当计数值 = 0000H 时，将输出 INTTM0p 并停止计数，直至检测到下一个开始触发（主通道的 INTTM0n）为止。从主通道产生 INTTM0n 并经过一个计数时钟后 TO0p 的输出电平成为有效电平，当 TCR0p = 0000H 时成为无效电平。

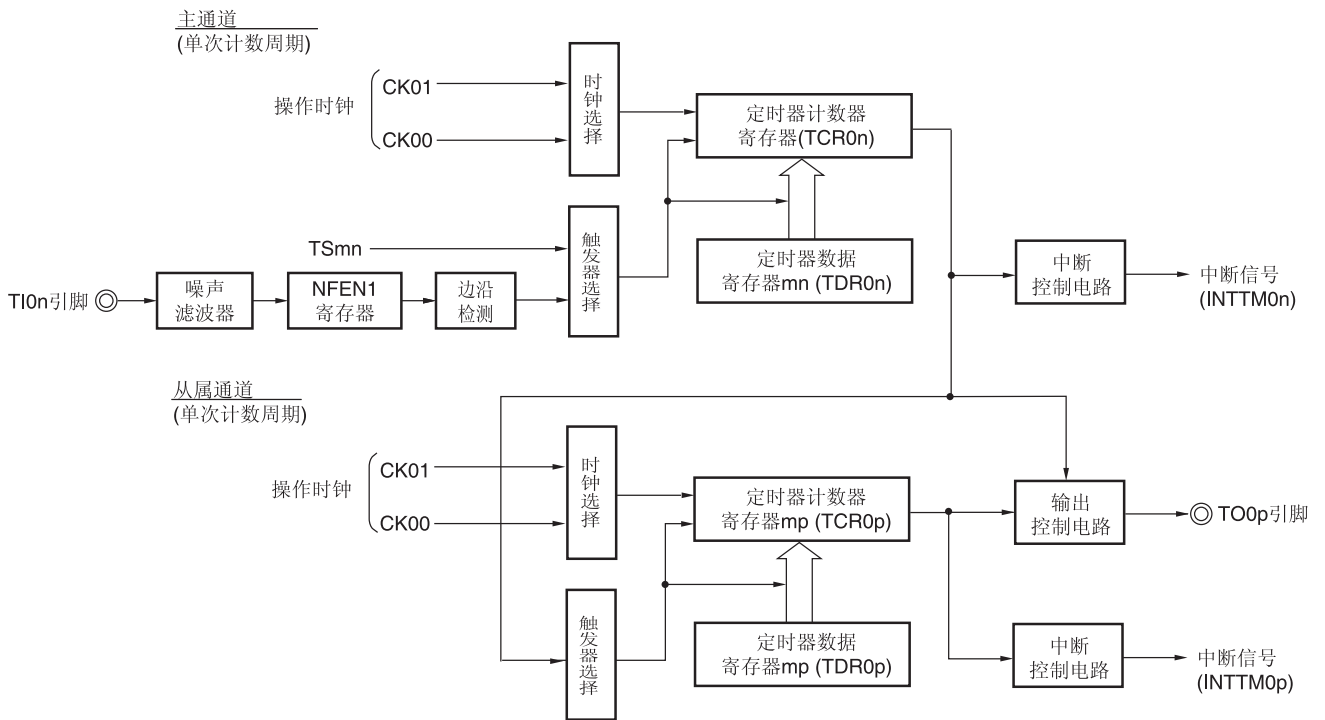
也可以不使用 TI0n 引脚输入，用软件操作 (TS0n = 1) 作为开始触发来输出单触发脉冲。

注意事项 主通道定时器数据寄存器 0n (TDR0n) 与从属通道 TDR0p 寄存器的载入时序不同。如果在操作过程中改写 TDR0n 和 TDR0p 寄存器，则会输出非法波形。请在产生 INTTM0n 后改写 TDR0n 寄存器，在产生 INTTM0p 后改写 TDR0p 寄存器。

备注 n: 通道编号 (n = 2, 4)

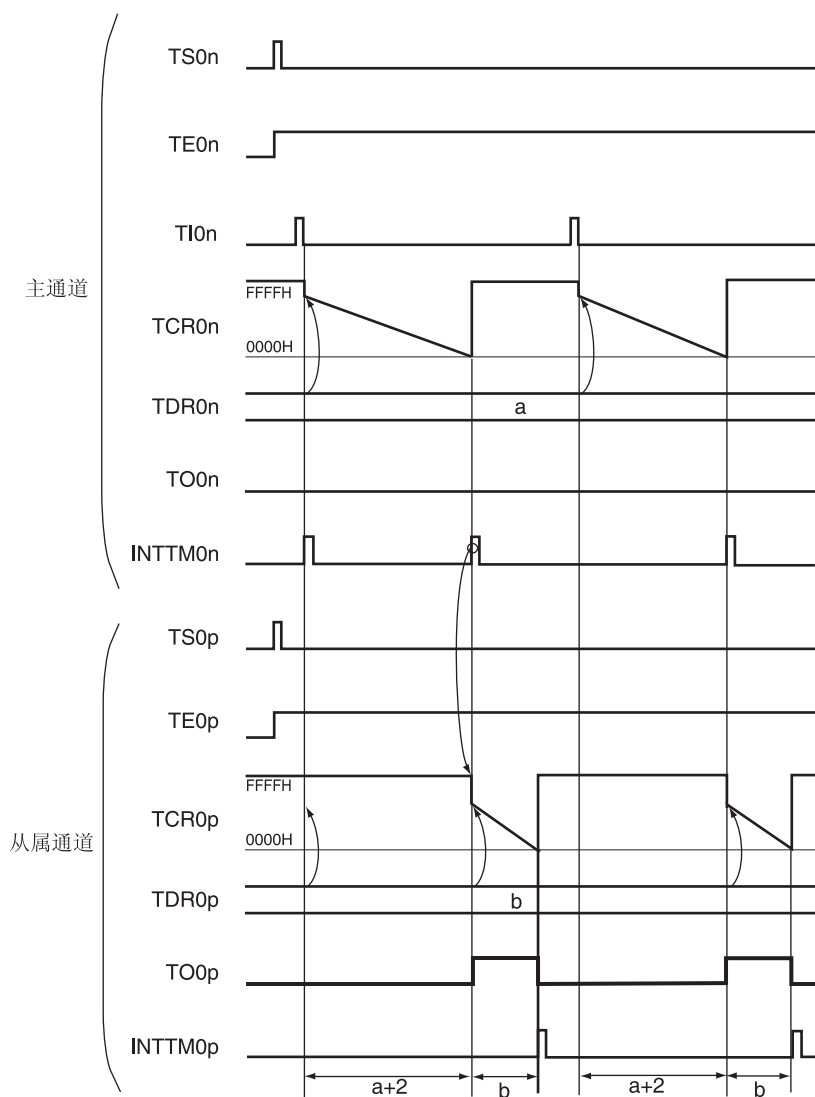
p: 从属通道编号 (n = 2 时, p = 3; n = 4 时, p = 5)

图 6-56. 作为单触发脉冲输出功能的操作框图



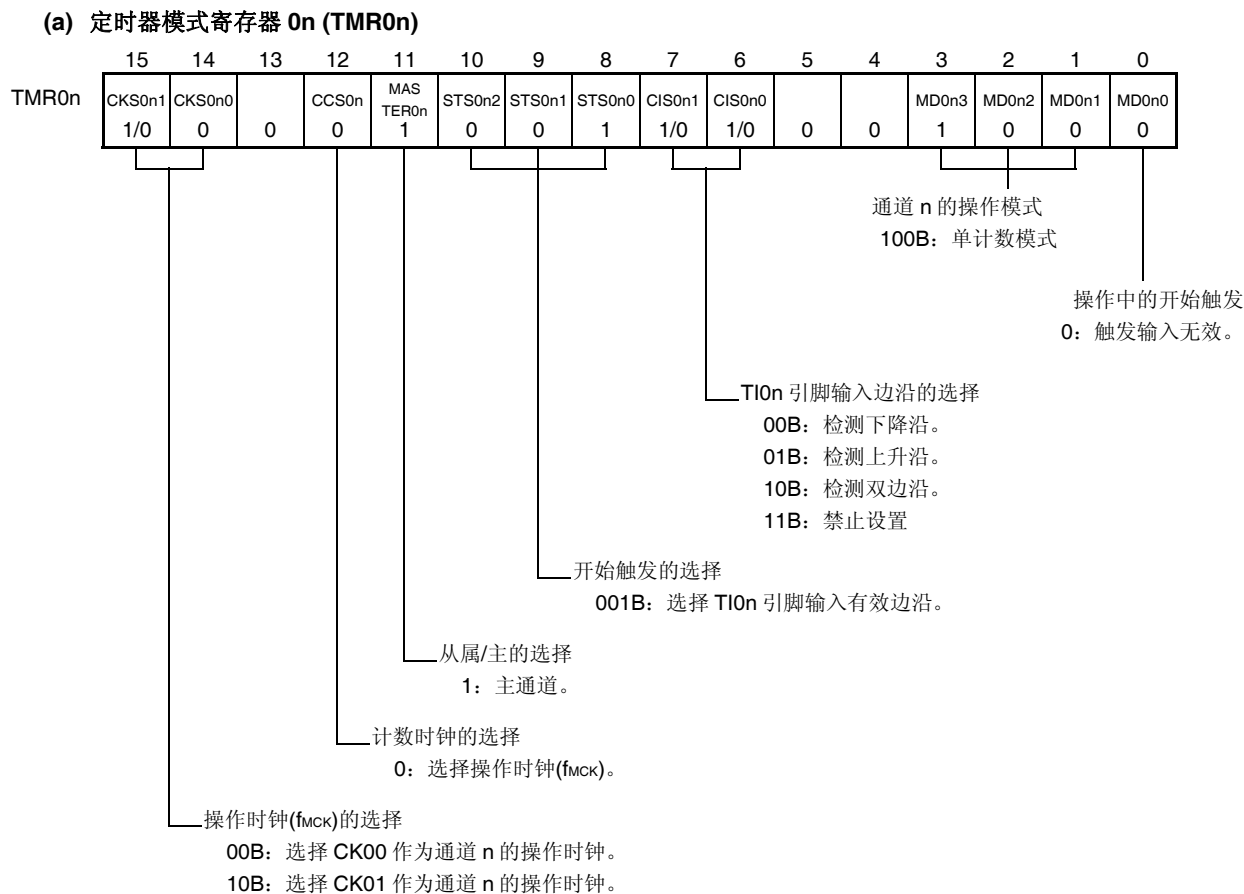
备注 n: 通道编号(n = 2, 4)
 p: 从属通道编号(n = 2 时, p = 3; n = 4 时, p = 5)

图 6-57. 作为单触发脉冲输出功能的操作基本时序示例

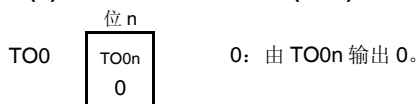


- 备注 1.** n: 通道编号 (n = 2, 4)
 p: 从属通道编号 (n = 2 时, p = 3; n = 4 时, p = 5)
- 2.** TS0n, TS0p: 定时器通道开始寄存器 0 (TS0) 的位 n、p
 TE0n, TE0p: 定时器通道允许状态寄存器 0 (TE0) 的位 n、p
 TI0n, TI0p: TI0n 和 TI0p 引脚输入信号
 TCR0n, TCR0p: 定时器/计数器寄存器 0n、0p (TCR0n、TCR0p)
 TDR0n, TDR0p: 定时器数据寄存器 0n、0p (TDR0n、TDR0p)
 TO0n, TO0p: TO0n 和 TO0p 引脚输出信号

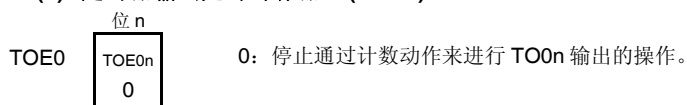
图 6-58. 使用单触发脉冲输出功能(主通道)时的寄存器设置内容示例



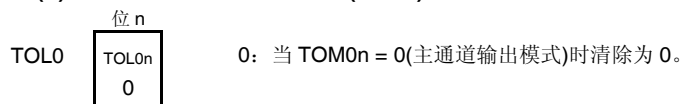
(b) 定时器输出寄存器 0 (TO0)



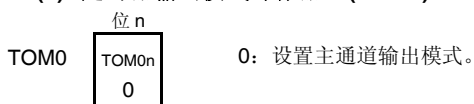
(c) 定时器输出允许寄存器 0 (TOE0)



(d) 定时器输出电平寄存器 0 (TOL0)

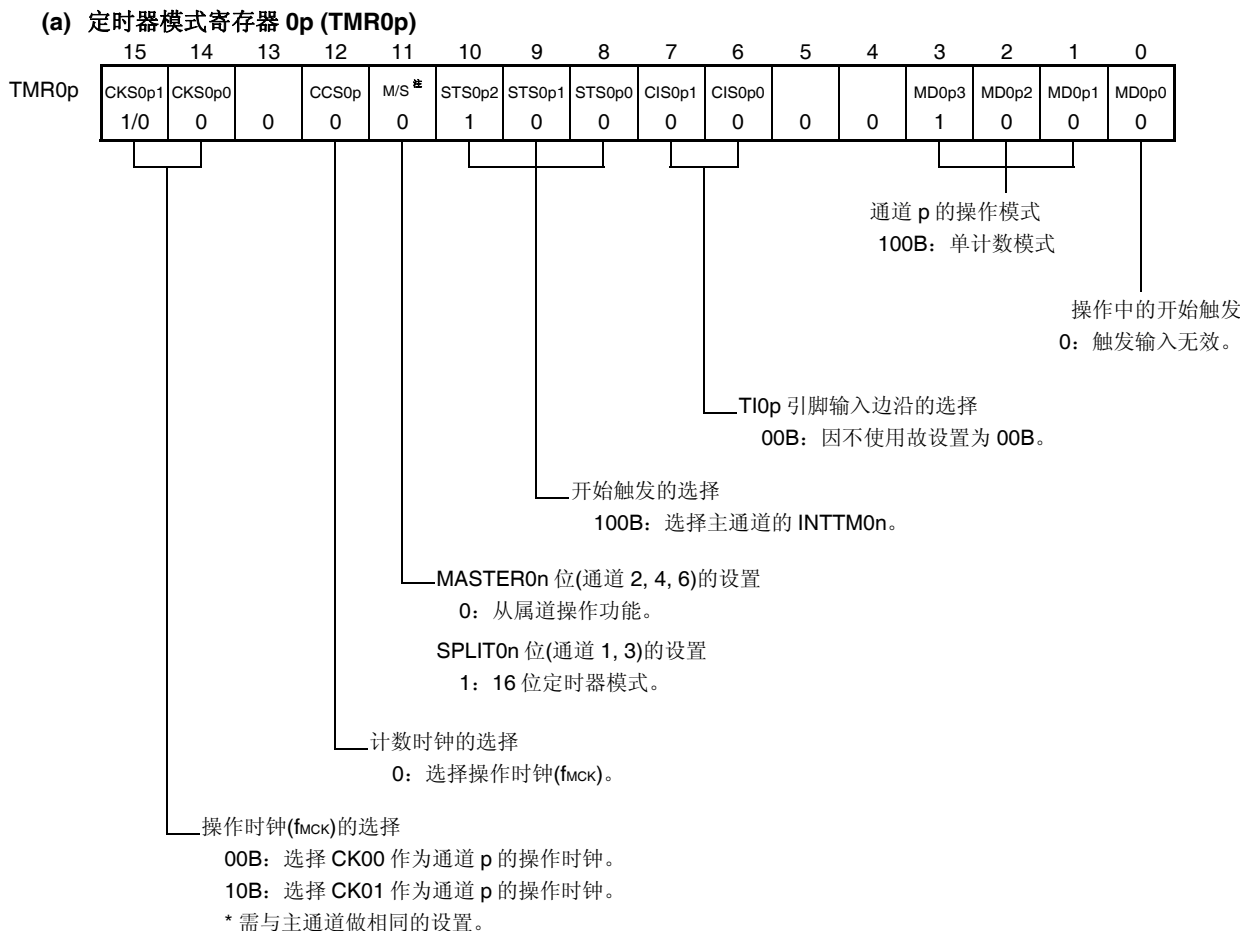


(e) 定时器输出模式寄存器 0 (TOM0)



备注 n: 通道编号 (n = 2, 4)

图 6-59. 使用单触发脉冲输出功能(从属通道)时的寄存器设置内容示例



(b) 定时器输出寄存器 0 (TO0)

	位 p	
TO0	TO0p 1/0	0: 由 TO0p 输出 0。 1: 由 TO0p 输出 1。

(c) 定时器输出允许寄存器 0 (TOE0)

	位 p	
TOE0	TOE0p 1/0	0: 停止通过计数动作来进行 TO0p 输出的操作。 1: 允许通过计数动作来进行 TO0p 输出的操作。

(d) 定时器输出电平寄存器 0 (TOL0)

	位 p	
TOL0	TOL0p 1/0	0: 正逻辑输出 (高电平有效) 1: 反相输出 (低电平有效)

(e) 定时器输出模式寄存器 0 (TOM0)

	位 p	
TOM0	TOM0p 1	1: 设置从属通道输出模式。

注 TMRm2, TMRm4, TMRm6: MASTER0n 位

TMRm1, TMRm3: SPLITmp 位

TMRm5, TMRm7: 固定为 0。

备注 n: 通道编号(n = 2, 4)

p: 从属通道编号(n = 2 时, p = 3; n = 4 时, p = 5)

图 6-60. 单触发脉冲输出功能的操作步骤 (1/2)

	软件操作	硬件状态
TAU初始 设定		断电状态 (时钟供应被停止, 向各寄存器的写操作被禁止。)
	将外围有效寄存器0(PER0)的TAU0EN位设置为1。 设置定时器时钟选择寄存器0(TPS0)。 决定CK00和CK01的时钟频率。	上电状态。每个通道处于停止操作状态。 (开始时钟供应, 允许写入各寄存器。)
通道初始 设定	设置即将使用的两个通道的定时器模式寄存器0n、0p (TMR0n、TMR0p) (决定通道的操作模式)。 向主通道定时器数据寄存器0n (TDR0n)设置输出延迟, 并 向从属通道TDR0p寄存器设置脉冲宽度。	通道处于停止操作状态。 (供应时钟, 并产生一定的功耗。)
	设置从属通道。 设置定时器输出模式寄存器0(TOM0)的TOM0p位设置 为1(从属通道输出模式)。 设置TOL0p位。 设置 TO0p位并决定 TO0p 输出的初始电平。 将TOE0p位设置为1, 并允许操作TO0p。 将端口寄存器和端口模式寄存器设置为0。	TO0p引脚处于Hi-Z输出状态。 当端口模式寄存器处于输出模式且端口寄存器为0时, 输出 TO0p的初始设置电平。 因为操作通道处于停止状态, 所以TO0p不会变化。 TO0p引脚输出TO0p的设置电平。

备注 n: 通道编号(n = 2, 4)

p: 从属通道编号(n = 2 时, p = 3; n = 4 时, p = 5)

图 6-60. 单触发脉冲输出功能的操作步骤 (2/2)

	软件操作	硬件状态
开始操作	将TOE0p位（从属）设置为1(仅限恢复操作时)。同时将定时器通道开始寄存器0 (TS0)的TS0n（主）和TS0p（从属）位设置为1。 TS0n和TS0p位是触发位，将自动返回为0。	TE0n和TE0p位被设置为1，主通道进入TI0n 输入边沿检测等待状态。 计数器停止操作。
	检测主通道的TI0n引脚输入有效边沿。	主通道开始计数。
操作期间	可以任意更改TMR0n寄存器的CIS0n1和CIS0n0位的设置值。 将噪声滤波器允许寄存器1 (NFEN1)的对应值设置为1。 不能更改TMR0p、TDR0n、TDR0p寄存器、TOM0n、TOM0p、TOL0n和TOL0p位的设置值。 可以任意读取TCR0n和TCR0p寄存器。 不使用 TSR0n和TSR0p寄存器。 可以更改从属通道的TO0和TOE0寄存器的设置值。	主通道在检测到TI0n引脚有效输入边沿时，将 TDR0n寄存器的值载入定时器/计数器寄存器0n (TCR0n)，计数器开始递减计数。当计数值达到TCRmn = 0000H时，将产生INTTM0n中断输出，停止计数器直到下一个有效边沿输入TI0n引脚为止。 从属通道（由主通道的INTTM0n 触发）将 TDR0p寄存器的值载入TCR0p寄存器，计数器开始递减计数。TO0p的输出电平在从主通道产生INTTM0n并经过一个计数时钟之后成为有效电平。当TCR0p = 0000H时成为无效电平，停止计数操作。 之后重复以上操作。
停止操作	同时将TT0n（主）和TT0p（从属）位设置为1。 TT0n和TT0p位是触发位，将自动返回为0。	TE0n, TE0p = 0，并且停止计数操作。 TCR0n和TCR0p寄存器保持计数值并停止计数操作。 TO0p输出不被初始化，而是保持当前状态。
	将从属通道的TOE0p位清除为0，并给TO0p位设置值。	TO0p引脚输出TO0p的设置电平。
TAU 停止	保持TO0p引脚输出电平时 将希望保持的值设置到端口寄存器后，将TO0p位清除为0。	TO0p引脚输出电平由端口功能保持。
	不需要保持TO0p引脚输出电平时 切换端口模式寄存器至输入模式。	TO0p引脚输出电平处于Hi-Z输出状态。
	将PER0寄存器的TAU0EN位设置为0。	断电状态 初始化所有电路，同时初始化各通道的SFR。 (TO0p位清除为0，TO0p引脚被设置为端口模式。)

恢复操作。

备注 n: 通道编号(n = 2, 4)
p: 从属通道编号(n = 2 时, p = 3; n = 4 时, p = 5)

6.8.2 作为PWM功能的操作

可以利用两个通道作为一组，来产生任意周期和占空比的脉冲。

输出脉冲的周期和占空比可用下述表达式计算。

$$\begin{aligned} \text{脉冲周期} &= \{\text{TDR0n (主) 的设置值} + 1\} \times \text{计数时钟周期} \\ \text{占空比 [\%]} &= \{\text{TDR0p (从属) 的设置值}\} / \{\text{TDR0n (主) 的设置值} + 1\} \times 100 \\ \text{0\% 输出:} & \quad \text{TDR0p (从属) 的设置值} = 0000\text{H} \\ \text{100\% 输出:} & \quad \text{TDR0p (从属) 的设置值} \geq \{\text{TDR0n (主) 的设置值} + 1\} \end{aligned}$$

备注 如果 TDR0p (从属) 的设置值 > (TDR0n (主) 的设置值 + 1)，则占空比将超过 100%，而合为 100% 输出。

主通道工作于间隔定时器模式。如果定时器通道开始寄存器 0 (TS0) 的通道开始触发位 (TS0n) 被设置为 1，则输出中断 (INTTM0n)，定时器数据寄存器 0n (TDR0n) 的设置值被载入定时器/计数器寄存器 0n (TCR0n)，计数器与计数时钟同步并递减计数。当计数器达到 0000H 时，输出 INTTM0n 中断，再次将 TDR0n 寄存器的值载入 TCR0n 寄存器，计数器递减计数。重复该操作直到定时器通道停止寄存器 0 (TT0) 的通道停止触发位 (TT0n) 被设置为 1 为止。

如果用两个通道来输出一个 PWM 波形，直到主通道递减计数至 0000H 为止的周期为 PWM 输出 (TO0p) 周期。

从属通道工作于单计数模式。通过将主通道的 INTTM0n 中断用作开始触发，TCR0p 寄存器载入 TDR0p 寄存器的值，计数器递减计数至 0000H。当计数器达到 0000H 时，输出 INTTM0p 中断并等待直到产生下一个开始触发 (主通道的 INTTM0n) 为止。

如果用两个通道来输出一个 PWM 波形，直到从属通道递减计数至 0000H 为止的期间为 PWM 输出 (TO0p) 占空。

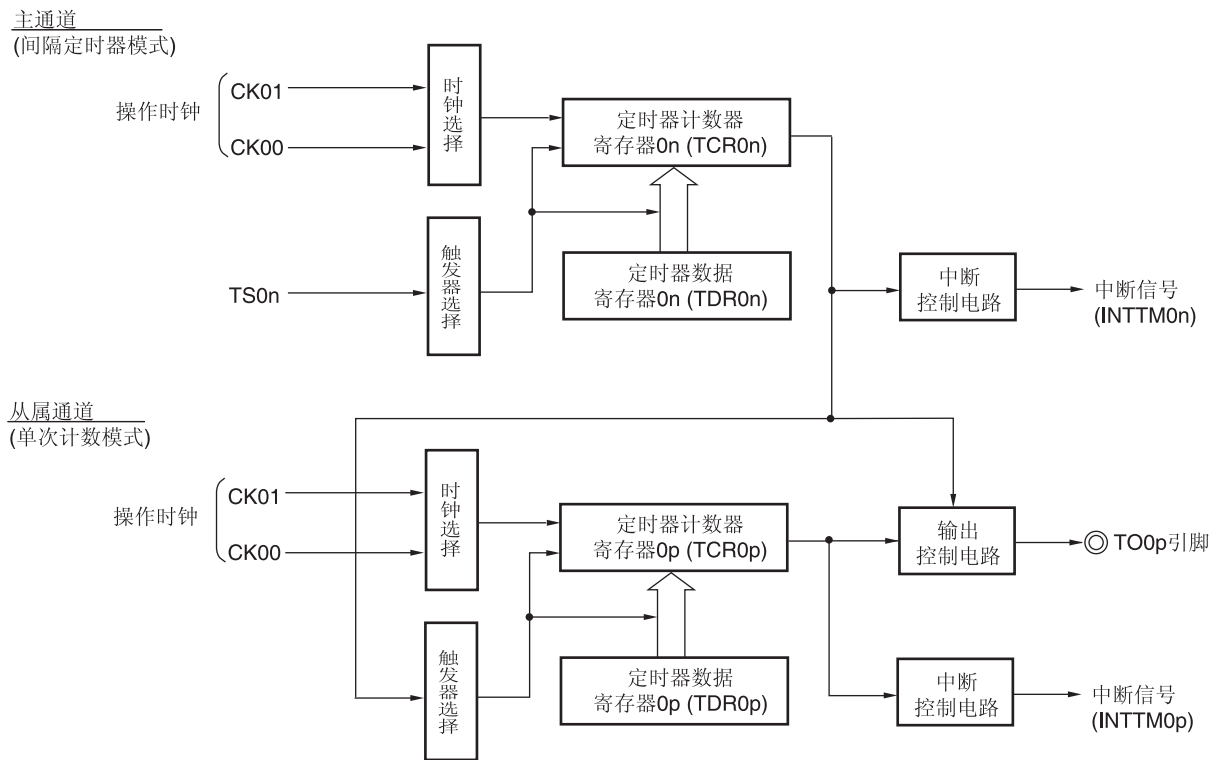
PWM 输出 (TO0p) 在主通道产生 INTTM0n 中断并经过一个时钟之后成为有效电平，当从属通道的 TCR0p 寄存器变成 0000H 时成为无效电平。

注意事项 要同时改写主通道定时器数据寄存器 0n (TDR0n) 和从属通道的 TDR0p 寄存器，则需执行两次写存取。将 TDR0n 和 TDR0p 寄存器的值载入至 TCR0n 和 TCR0p 寄存器的时序为主通道 INTTM0n 发生的时候。因此，当在主通道 INTTM0n 发生前和发生后分开执行改写时，TO0p 引脚将不能输出预期波形。将主的 TDR0n 寄存器和从属的 TDR0p 寄存器的双方都改写时，必须要在主通道产生 INTTM0n 之后立即改写这两个寄存器。

备注 n: 通道编号 (n = 2, 4)

p: 从属通道编号 (n = 2 时, p = 3; n = 4 时, p = 5)

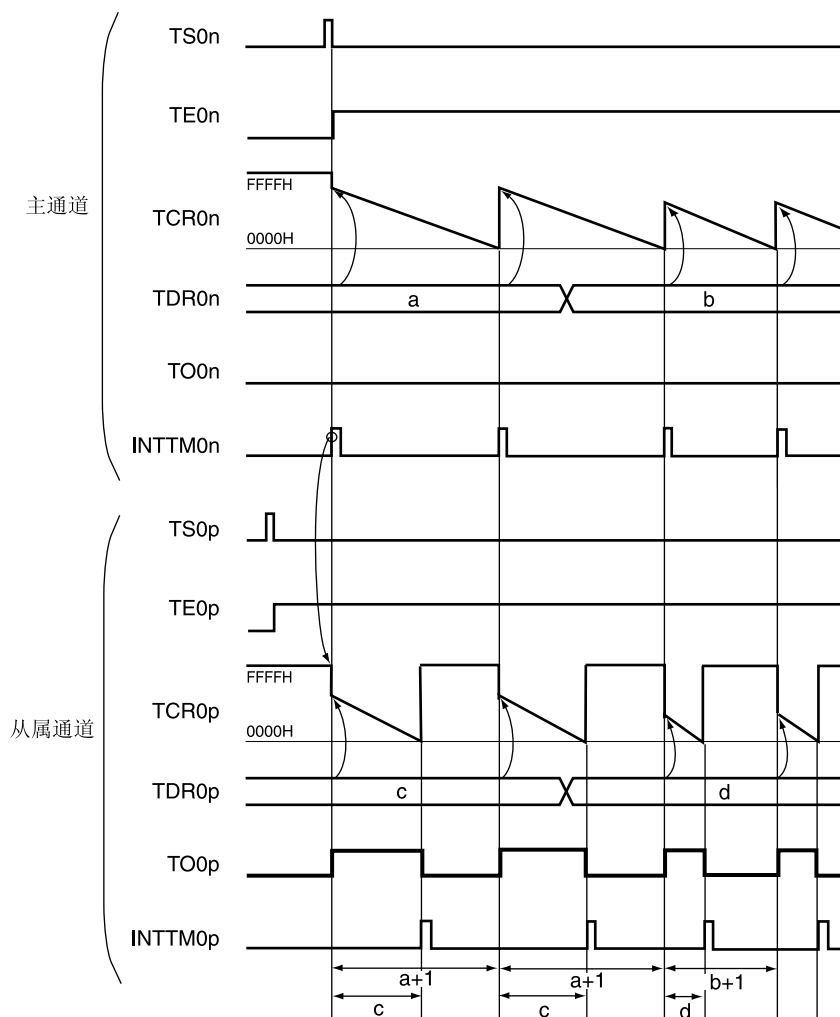
图 6-61. 作为 PWM 功能的操作框图



备注 n: 通道编号(n = 2, 4)

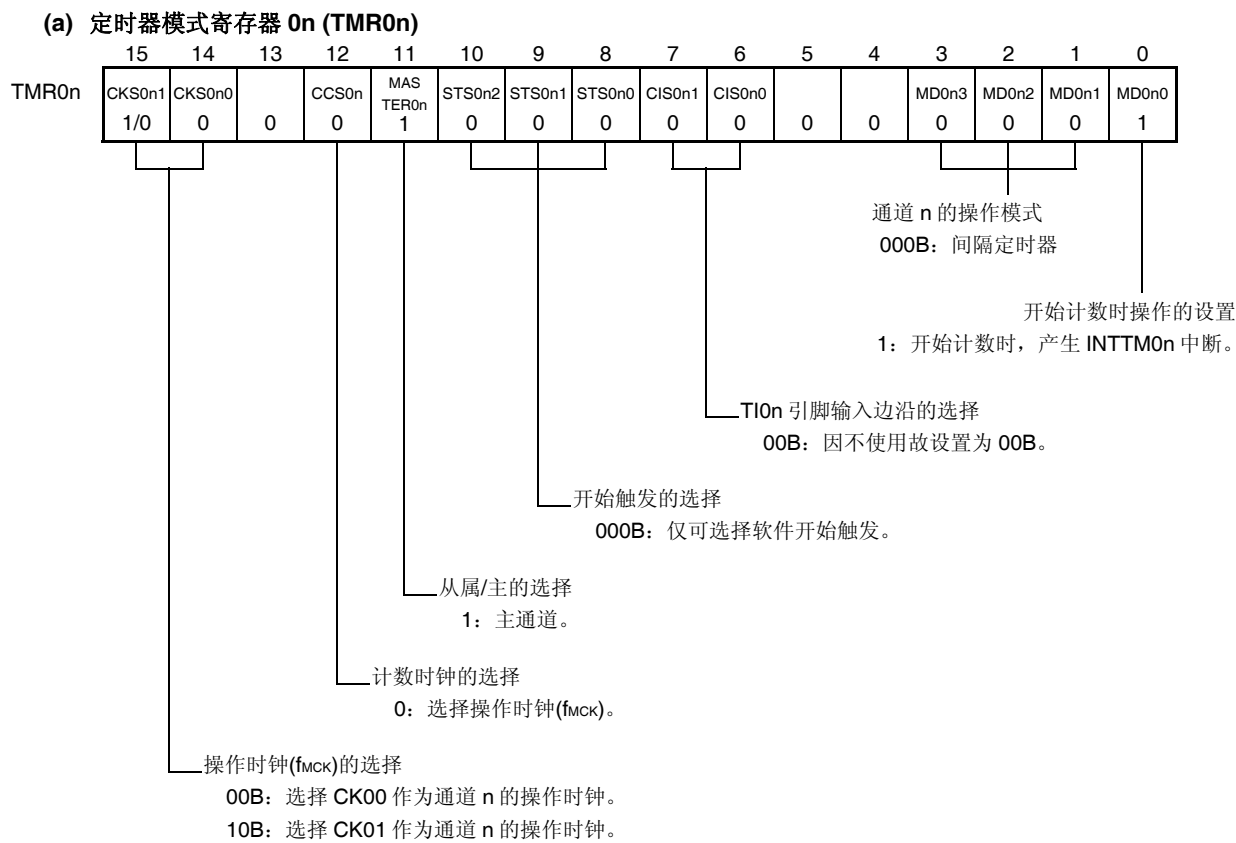
p: 从属通道编号(n = 2 时, p = 3; n = 4 时, p = 5)

图 6-62. 作为 PWM 功能的操作基本时序示例



- 备注 1.** n: 通道编号(n = 2, 4)
p: 从属通道编号(n = 2 时, p = 3; n = 4 时, p = 5)
- 2.** TS0n, TS0p: 定时器通道开始寄存器 0 (TS0)的位 n、p
TE0n, TE0p: 定时器通道允许状态寄存器 0 (TE0)的位 n、p
TCR0n, TCR0p: 定时器/计数器寄存器 0n、0p (TCR0n、TCR0p)
TDR0n, TDR0p: 定时器数据寄存器 0n、0p (TDR0n、TDR0p)
TO0n, TO0p: TO0n 和 TO0p 引脚输出信号

图 6-63. 使用 PWM 功能（主通道）时的寄存器设置内容示例



(b) 定时器输出寄存器 0 (TO0)

TO0

位 n
TO0n
0

 0: 由 TO0n 输出 0。

(c) 定时器输出允许寄存器 0 (TOE0)

TOE0

位 n
TOE0n
0

 0: 停止通过计数动作来进行 TO0n 输出的操作。

(d) 定时器输出电平寄存器 0 (TOL0)

TOL0

位 n
TOL0n
0

 0: 当 TOM0n = 0(主通道输出模式)时清除为 0。

(e) 定时器输出模式寄存器 0 (TOM0)

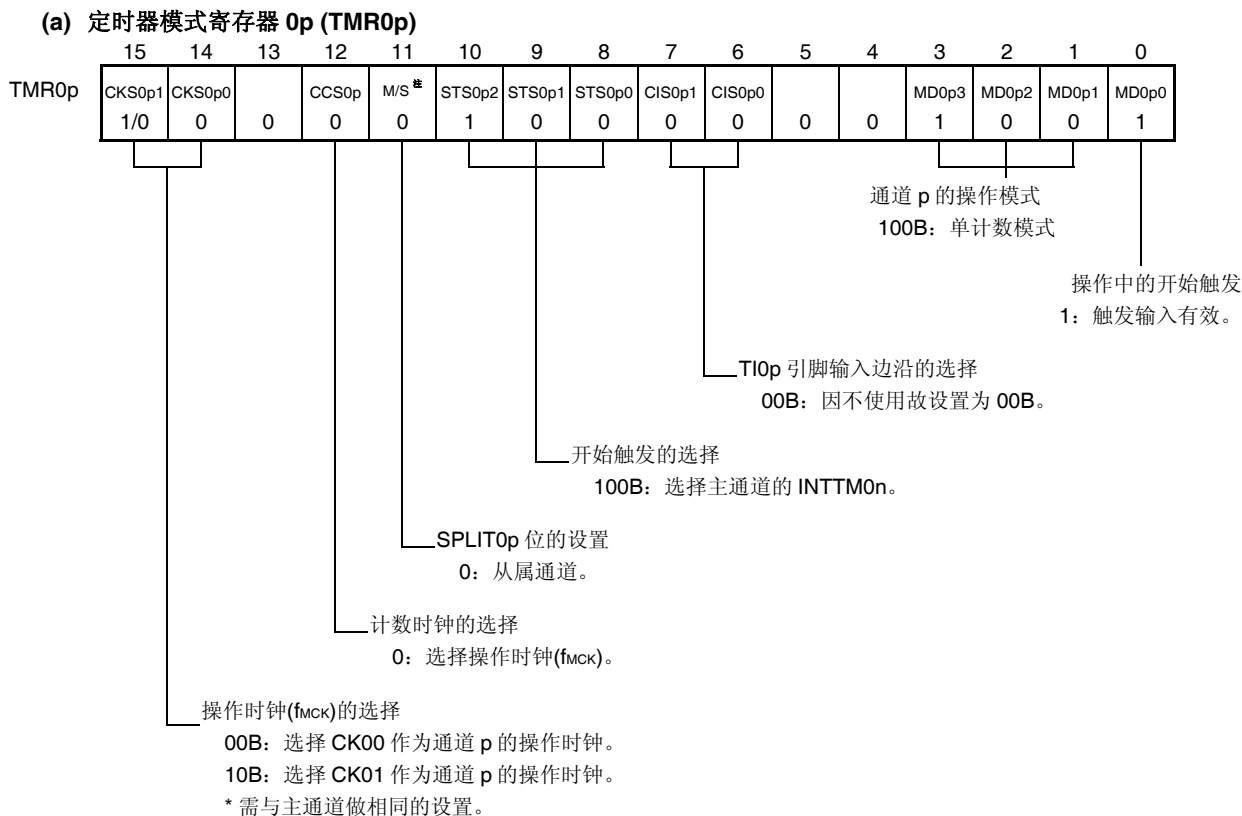
TOM0

位 n
TOM0n
0

 0: 设置主通道输出模式。

备注 n: 通道编号(n = 2, 4)

图 6-64. 使用 PWM 功能（从属通道）时的寄存器设置内容示例



(b) 定时器输出寄存器 0 (TO0)

	位 p	
TO0	TO0p	0: 由 TO0p 输出 0。 1: 由 TO0p 输出 1。
	1/0	

(c) 定时器输出允许寄存器 0 (TOE0)

	位 p	
TOE0	TOE0p	0: 停止通过计数动作来进行 TO0p 输出的操作。 1: 允许通过计数动作来进行 TO0p 输出的操作。
	1/0	

(d) 定时器输出电平寄存器 0 (TOL0)

	位 p	
TOL0	TOL0p	0: 正逻辑输出（高电平有效） 1: 反相输出（低电平有效）
	1/0	

(e) 定时器输出模式寄存器 0 (TOM0)

	位 p	
TOM0	TOM0p	1: 设置从属通道输出模式。
	1	

注 TMR05, TMR07: 固定为 0。

TMR01, TMR03: SPLIT0p 位

备注 n: 通道编号(n = 2, 4)

p: 从属通道编号(n = 2 时, p = 3; n = 4 时, p = 5)

图 6-65. 使用 PWM 功能时的操作步骤 (1/2)

	软件操作	硬件状态
TAU初始 设定		断电状态 (时钟供应被停止, 向各寄存器的写操作被禁止。)
	将外围有效寄存器0(PER0)的TAU0EN位设置为1。	上电状态。每个通道处于停止操作状态。 (开始时钟供应, 允许写入各寄存器。)
	设置定时器时钟选择寄存器0(TPS0)。 决定CK00和CK01的时钟频率。	
通道初始 设定	设置即将使用的两个通道的定时器模式寄存器0n, 0p (TMR0n, TMR0p) (决定通道的操作模式)。 向主通道的定时器数据寄存器0n (TDR0n)设置间隔(周期)值, 并向从属通道的TDR0p寄存器设置占空比。	通道处于停止操作状态。 (供应时钟, 并产生一定的功耗。)
	设置从属通道。 设置定时器输出模式寄存器0(TOM0)的TOM0p位设置为1(从属通道输出模式)。 设置TOL0p位。 设置TO0p位并决定TO0p输出的初始电平。	TO0p引脚处于Hi-Z输出状态。
	将TOE0p位设置为1, 并允许操作TO0p。	当端口模式寄存器处于输出模式且端口寄存器为0时, 输出TO0p的初始设置电平。
	将端口寄存器和端口模式寄存器设置为0。	因为操作通道处于停止状态, 所以TO0p不会变化。 TO0p引脚输出TO0p设置电平。

备注 n: 通道编号(n = 2, 4)

p: 从属通道编号(n = 2 时, p = 3; n = 4 时, p = 5)

图 6-65. 使用 PWM 功能时的操作步骤 (2/2)

	软件操作	硬件状态	
恢复操作。	开始操作	<p>将TOE0p位（从属）设置为1(仅限恢复操作时)。同时将定时器通道开始寄存器0 (TS0)的TS0n（主）和TS0p（从属）位设置为1。TS0n和TS0p位是触发位，将自动返回为0。</p>	<p>TE0n = 1, TE0p = 1 主通道开始计数时，产生INTTM0n。通过该中断触发，从属通道也开始计数。</p>
	操作期间	<p>不能更改TMR0n和TMR0p寄存器、TOM0n、TOM0p、TOL0n和TOL0p位的设置值。 产生主通道INTTM0n之后，可以更改TDR0n和TDR0p寄存器的设置值。 可以任意读取TCR0n和TCR0p寄存器。 不使用 TSR0n和TSR0p寄存器。 可以更改TO0和TOE0寄存器的设置值。</p>	<p>主通道的计数器将TDR0n寄存器值载入定时器/计数器寄存器0n (TCR0n)，并递减计数。当计数值达到TCR0n = 0000H时，输出INTTM0n。同时，TDR0n寄存器的值将被载入TCR0n寄存器，计数器再次开始递减计数。 在从属通道，TDR0p寄存器的值载入TCR0p寄存器（由主通道的INTTM0n触发），计数器开始递减计数。TO0p的输出电平在主通道输出INTTM0n并经过一个计数时钟之后成为有效电平。当TCR0p = 0000H时成为无效电平，停止计数操作。 之后重复以上操作。</p>
	停止操作	<p>同时将TT0n（主）和TT0p（从属）位设置为1。TT0n和TT0p位是触发位，将自动返回为0。</p>	<p>TE0n, TE0p = 0，并且停止计数操作。 TCR0n和TCR0p寄存器保持计数值并停止计数操作。 TO0p输出不被初始化，而是保持当前状态。</p>
	TAU 停止	<p>保持TO0p引脚输出电平时 将希望保持的值设置到端口寄存器后，将TO0p位清除为0。 不需要保持TO0p引脚输出电平时 切换端口模式寄存器至输入模式。 将PER0寄存器的TAU0EN位设置为0。</p>	<p>TO0p引脚输出电平由端口功能保持。 TO0p引脚输出电平处于Hi-Z输出状态。 断电状态 初始化所有电路，同时初始化各通道的SFR。 (TO0p位清除为0，TO0p引脚被设置为端口模式。)</p>

备注 n: 通道编号(n = 2, 4)

p: 从属通道编号(n = 2 时, p = 3; n = 4 时, p = 5)

6.8.3 作为多重PWM输出功能的操作

通过扩展 PWM 功能并利用多个从属通道，可以输出具有不同占空比的多个 PWM 波形。

例如，使用两个从属通道时，输出脉冲的周期和占空比可用下述表达式计算。

$$\begin{aligned} \text{脉冲周期} &= \{\text{TDR0n (主) 的设置值} + 1\} \times \text{计数时钟周期} \\ \text{占空比 1 [\%]} &= \{\text{TDR0p (从属 1) 的设置值}\} / \{\text{TDR0n (主) 的设置值} + 1\} \times 100 \\ \text{占空比 2 [\%]} &= \{\text{TDR0q (从属 2) 的设置值}\} / \{\text{TDR0n (主) 的设置值} + 1\} \times 100 \end{aligned}$$

备注 如果 TDR0p (从属 1) 的设置值 > {TDR0n (主) 的设置值+ 1}，或者 {TDR0q (从属 2) 的设置值} > {TDR0n (主) 的设置值+ 1}，则占空比将超过 100%，而合为 100%输出。

主通道的定时器/计数器寄存器 0n (TCR0n) 工作于间隔定时器模式，对周期进行计数。

从属通道 1 的 TCR0p 寄存器工作于单计数模式，对占空比进行计数，并从 TO0p 引脚输出 PWM 波形。将主通道的 INTTM0n 用作开始触发，TCR0p 寄存器载入定时器数据寄存器 0p (TDR0p) 的值，并开始递减计数。当 TCR0p = 0000H 时，TCR0p 将输出 INTTM0n 并停止计数，直到输入下一个开始触发 (主通道的 INTTM0n) 为止。从主通道产生 INTTM0n 并经过一个计数时钟后 TO0p 的输出电平成为有效电平，当 TCR0p = 0000H 时成为无效电平。

采用与从属通道 1 的 TCR0p 寄存器一样的方式，从属通道 2 的 TCR0q 寄存器工作于单计数模式，对占空比进行计数，并从 TO0q 引脚输出 PWM 波形。将主通道的 INTTM0n 用作开始触发，TO0q 寄存器载入 TO0q 寄存器的值，并开始递减计数。当 TCR0q = 0000H 时，TCR0q 寄存器输出 INTTM0q 并停止计数，直到输入下一个开始触发 (主通道的 INTTM0n) 为止。从主通道产生 INTTM0n 并经过一个计数时钟后 TO0q 的输出电平成为有效电平，当 TCR0q = 0000H 时成为无效电平。

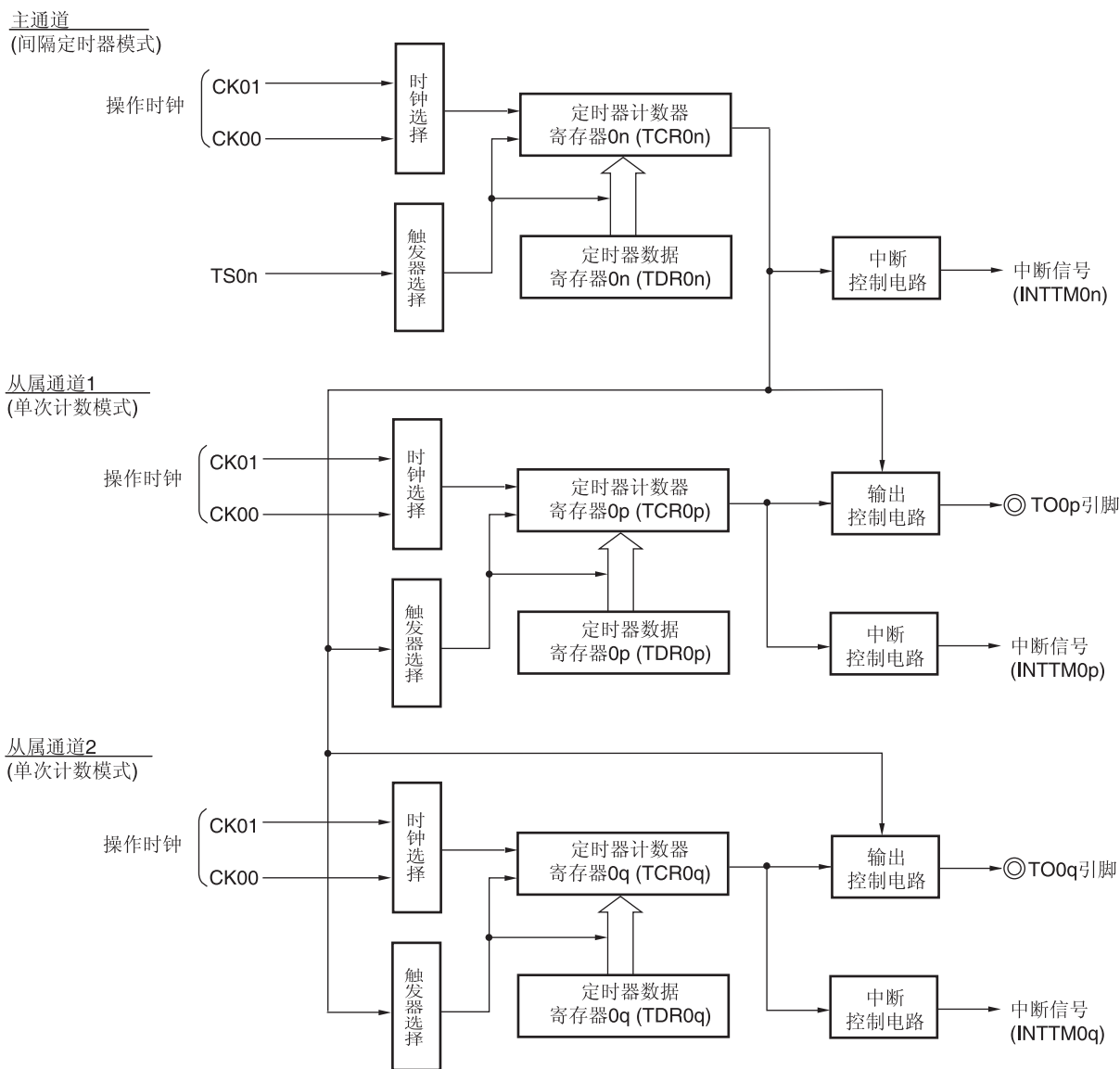
当通道 0 被用作上述主通道时，最多可以同时输出七种 PWM 信号。

注意事项 要同时改写主通道定时器数据寄存器 0n (TDR0n) 和从属通道 1 的 TDR0p 寄存器，至少需要执行两次写操作。由于 TDR0n 和 TDR0p 寄存器的值是在主通道产生 INTTM0n 之后被载入 TCR0n 和 TCR0p 寄存器的，因此，如果在主通道产生 INTTM0n 之前和之后分别执行改写，则 TO0p 引脚不能输出预期波形。改写主的 TDR0n 寄存器和从属的 TDR0p 寄存器的双方时，必须要在主通道产生 INTTM0n 之后立即改写这两个寄存器 (同样适用于从属通道 2 的 TDR0q 寄存器)。

备注 n: 通道编号 (n = 4)

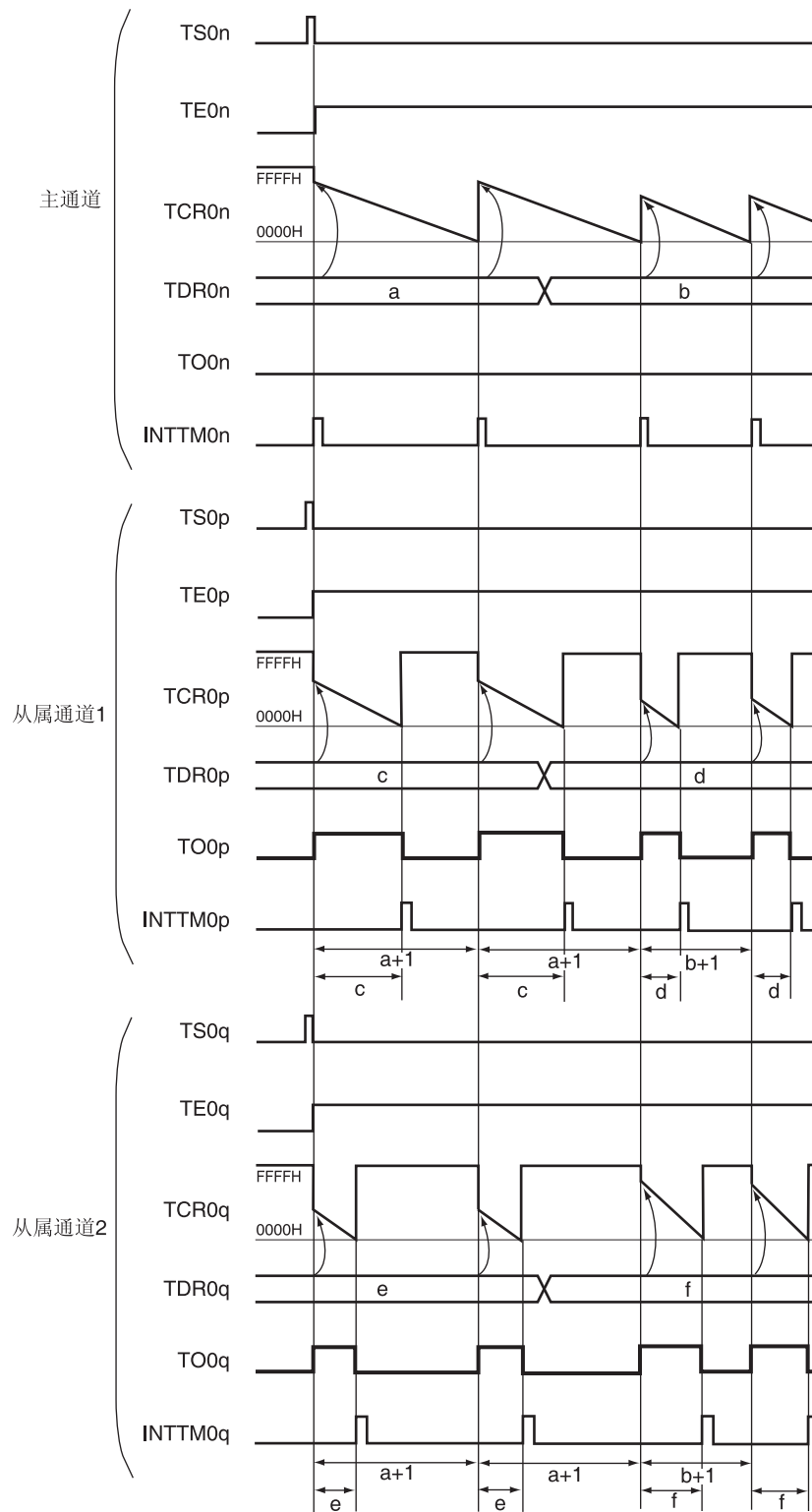
p: 从属通道编号 1 (p = 5), q: 从属通道编号 2 (q = 6)

图 6-66. 多重 PWM 输出功能的操作框图 (输出两种 PWM)



备注 n: 通道编号(n = 4)
 p: 从属通道编号 1 (p = 5), q: 从属通道编号 2 (q = 6)

图 6-67. 作为多重 PWM 输出功能的操作基本时序示例
(输出两种 PWM) (1/2)

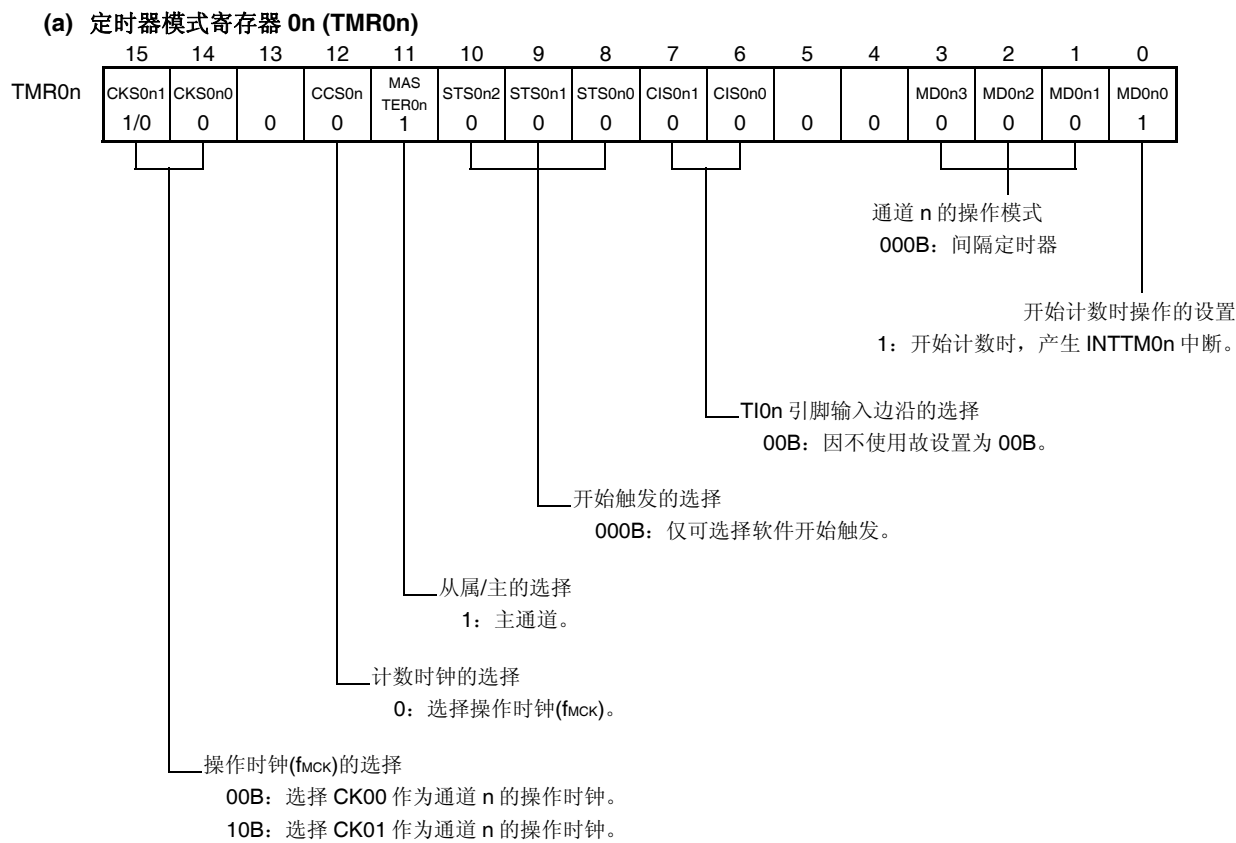


(备注如下页所示。)

图 6-67. 作为多重 PWM 输出功能的操作基本时序示例
(输出两种 PWM) (2/2)

- 备注 1. n: 通道编号(n = 4)
p: 从属通道编号 1 (p = 5), q: 从属通道编号 2 (q = 6)
2. TS0n, TS0p, TS0q: 定时器通道开始寄存器 0 (TS0)的位 n、p、q
TE0n, TE0p, TE0q: 定时器通道允许状态寄存器 0 (TE0)的位 n、p、q
TCR0n, TCR0p, TCR0q: 定时器/计数器寄存器 0n、0p、0q (TCR0n、TCR0p、TCR0q)
TDR0n, TDR0p, TDR0q: 定时器数据寄存器 0n、0p、0q (TDR0n、TDR0p、TDR0q)
TO0n, TO0p, TO0q: TO0n、TO0p 和 TO0q 引脚输出信号

图 6-68. 使用多重 PWM 输出功能(主通道)时的寄存器设置内容示例



(b) 定时器输出寄存器 0 (TO0)

TO0

位 n
TO0n
0

 0: 由 TO0n 输出 0。

(c) 定时器输出允许寄存器 0 (TOE0)

TOE0

位 n
TOE0n
0

 0: 停止通过计数动作来进行 TO0n 输出的操作。

(d) 定时器输出电平寄存器 0 (TOL0)

TOL0

位 n
TOL0n
0

 0: 当 TOM0n = 0(主通道输出模式)时清除为 0。

(e) 定时器输出模式寄存器 0 (TOM0)

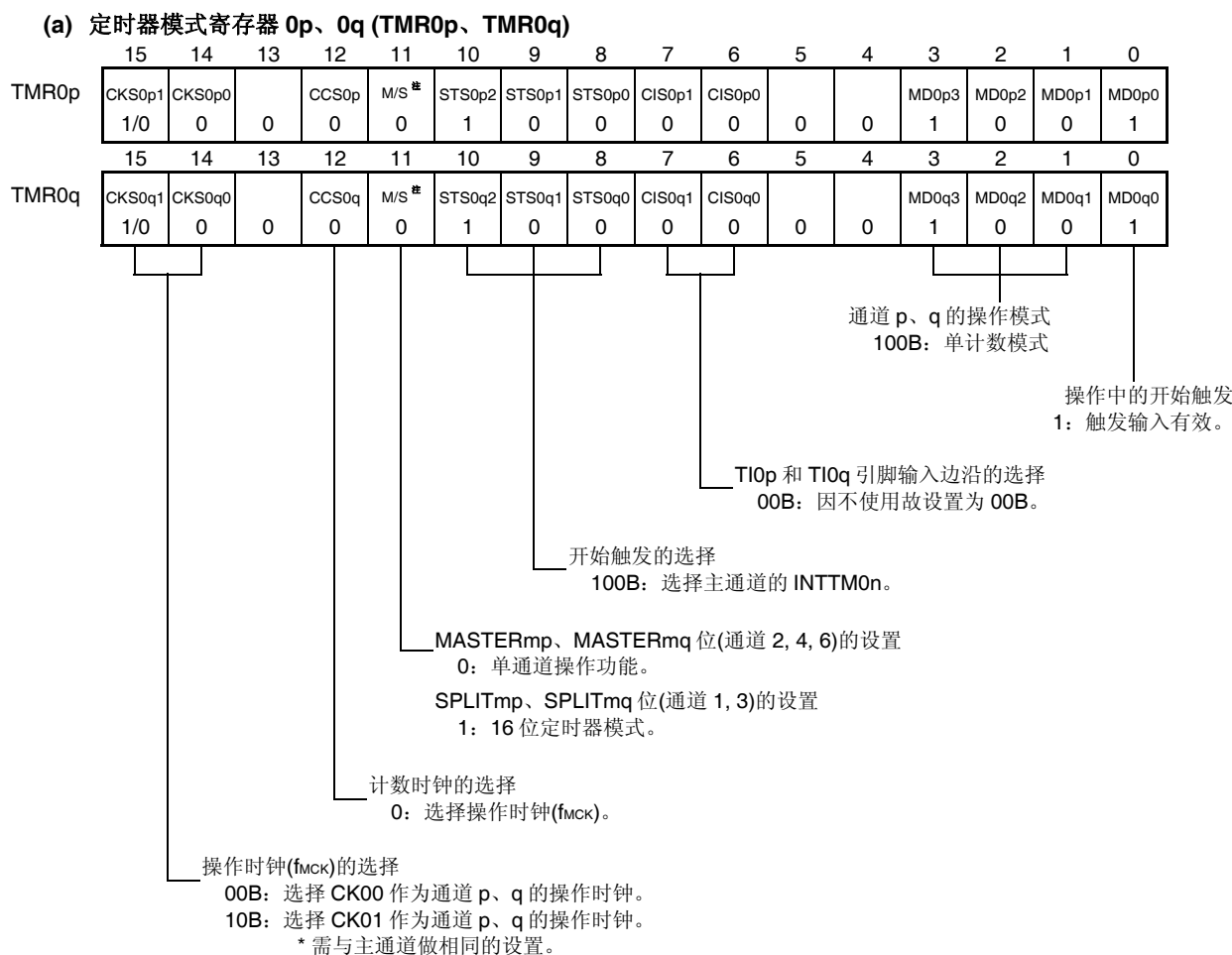
TOM0

位 n
TOM0n
0

 0: 设置主通道输出模式。

备注 n: 通道编号(n = 4)

图 6-69. 使用多重 PWM 输出功能(从属通道)时的寄存器设置内容示例(输出两种 PWM)



(b) 定时器输出寄存器 0 (TO0)

	位 q	位 p	
TO0	TO0q	TO0p	0: 由 TO0p 或者 TO0q 输出 0。 1: 由 TO0p 或者 TO0q 输出 1。
	1/0	1/0	

(c) 定时器输出允许寄存器 0 (TOE0)

	位 q	位 p	
TOE0	TOE0q	TOE0p	0: 停止通过计数动作来进行 TO0p 或 TO0q 输出的操作。 1: 允许通过计数动作来进行 TO0p 或 TO0q 输出的操作。
	1/0	1/0	

(d) 定时器输出电平寄存器 0 (TOL0)

	位 q	位 p	
TOL0	TOL0q	TOL0p	0: 正逻辑输出 (高电平有效) 1: 反相输出 (低电平有效)
	1/0	1/0	

(e) 定时器输出模式寄存器 0 (TOM0)

	位 q	位 p	
TOM0	TOM0q	TOM0p	1: 设置从属通道输出模式。
	1	1	

注 TMRm2, TMRm4, TMRm6: MASTERmp, MASTERmq 位
TMRm1, TMRm3: SPLITmp, SPLITmq 位
TMRm5, TMRm7: 固定为 0。

备注 n: 通道编号(n = 4)
p: 从属通道编号 1 (p = 5), q: 从属通道编号 2 (q = 6)

图 6-70. 使用多重 PWM 输出功能时的操作步骤 (1/2)

	软件操作	硬件状态
TAU初始设定	将外围有效寄存器0(PER0)的TAU0EN位设置为1。 设置定时器时钟选择寄存器0(TPS0)。决定CK00和CK01的时钟频率。	断电状态 (时钟供应被停止, 向各寄存器的写操作被禁止。) 上电状态。每个通道处于停止操作状态。 (开始时钟供应, 允许写入各寄存器。)
通道初始设定	设置即将使用的各个通道的定时器模式寄存器0n、0p、0q (TMR0n, TMR0p, TMR0q) (决定通道的操作模式)。向主通道定时器数据寄存器0n (TDR0n)设置间隔 (周期)值, 并向从属通道的TDR0p和TDR0q寄存器设置占空比。 设置从属通道。 将定时器输出模式寄存器0(TOM0)的TOM0p和TOM0q位设置为1(从属通道输出模式)。 设置TOL0p和TOL0q位。 设置TO0p和TO0q位, 并决定TO0p和TO0q输出的初始电平。 将TOE0p和TOE0q位设置为1, 并允许操作TO0p和TO0q。 将端口寄存器和端口模式寄存器设置为0。	通道处于停止操作状态。 (供应时钟, 并产生一定的功耗。) TO0p和TO0q引脚进入 Hi-Z 输出状态。 当端口模式寄存器处于输出模式且端口寄存器为0时, 输出TO0p和TO0q 初始设置电平。 因为操作通道处于停止状态, 所以TO0p和TO0q不会变化。 TO0p和TO0q引脚输出TO0p和TO0q的设置电平。

备注 n: 通道编号(n = 4)
 p: 从属通道编号 1 (p = 5), q: 从属通道编号 2 (q = 6)

图 6-70. 使用多重 PWM 输出功能时的操作步骤 (2/2)

	软件操作	硬件状态
恢复操作。	开始操作 (仅限恢复操作时将 TOE0p和TOE0q (从属) 位设置为 1。) 同时将定时器通道开始寄存器0 (TS0)的TS0n位 (主)、TS0p和TS0q (从属) 位设置为1。 TS0n、TS0p和TS0q 位是触发位, 将自动返回0。	TE0n = 1, TE0p, TE0q = 1 主通道开始计数时, 产生INTTM0n中断。通过该中断触发, 从属通道也开始计数。
	操作期间 不能更改 TMR0n、TMR0p、TMR0q寄存器、TOM0n、TOM0p、TOM0q、TOL0n、TOL0p和TOL0q位的设置值。 产生主通道INTTM0n之后, 可以更改TDR0n、TDR0p和TDR0q寄存器的设置值。 可以任意读取TCR0n、TCR0p和TCR0q寄存器。 不使用TSR0n、TSR0p和TSR0q寄存器。 可以更改TO0和TOE0寄存器的设置值。	主通道的计数器将 TDR0n寄存器值载入定时器/计数器寄存器0n (TCR0n), 并递减计数。当计数值达到TCR0n = 0000H时, 输出INTTM0n。同时, TDR0n寄存器的值将被载入 TCR0n寄存器, 计数器再次开始递减计数。 在从属通道1, TDR0p寄存器的值被传送到TCR0p寄存器 (由主通道的INTTM0n触发), 计数器开始递减计数。 TO0p的输出电平在主通道输出INTTM0n并经过一个计数时钟之后成为有效电平。当TCR0p = 0000H时成为无效电平, 停止计数操作。 在从属通道2, TDR0q寄存器的值被传送到TCR0q寄存器 (由主通道的INTTM0n触发), 计数器开始递减计数。 TO0q的输出电平在主通道输出INTTM0n且经过一个计数时钟之后成为有效电平。当TCR0q = 0000H时成为无效电平, 停止计数操作。 之后重复以上操作。
	停止操作 同时将TT0n位 (主)、TT0p和TT0q (从属) 位设置为 1。 TT0n、TT0p和TT0q位是触发位, 将自动返回为0。	TE0n, TE0p, TE0q = 0, 并且停止计数操作。 TCR0n、TCR0p和TCR0q寄存器保持计数值并停止计数操作。 TO0p和TO0q输出不被初始化, 而是保持当前状态。
	将从属通道的TOE0p和TOE0q位清除为0, 并设置TO0p和TO0q位的值。	由TO0p和TO0q引脚输出TO0p和TO0q设置电平。
TAU 停止	保持TO0p和TO0q引脚输出电平时 将希望保持的值设置到端口寄存器之后, 将 TO0p和TO0q位清除为0。 不需要保持TO0p和TO0q引脚输出电平时 切换端口模式寄存器至输入模式。 将PER0寄存器的TAU0EN位设置为0。	TO0p和TO0q引脚输出电平由端口功能保持。 TO0p和TO0q引脚输出电平处于Hi-Z输出状态。 断电状态 初始化所有电路, 同时初始化各通道的SFR。 (TO0p和TO0q位被清除为0, TO0p和TO0q引脚被设置为端口模式。)

备注 n: 通道编号(n = 4)

p: 从属通道编号 1 (p = 5), q: 从属通道编号 2 (q = 6)

第七章 16 位定时器 KB0、KB1 和 KB2

16 位定时器 KB0、KB1 和 KB2 可生成适合电源和照明控制的 PWM 输出。

16 位定时器的通道数量因产品而异。

	20引脚	30引脚	32引脚	38引脚
16位定时器KB0	√	√	√	√
16位定时器KB1	√	√	√	√
16位定时器KB2	–	√	√	√

注意事项 1. 本章的下述内容大多是以 38 引脚产品为例。

2. 20 引脚产品不向外部输出 16 位定时器 KB2。

7.1 16 位定时器KB0、KB1 和KB2 的功能

16 位定时器 KB0、KB1 和 KB2 是专用 PWM 输出定时器，分别具有 2 个输出，最多可生成 6 个 PWM 输出。此外，还可以生成互补 PWM 输出，用来控制半桥电路（2 个输出）、全桥电路（4 个输出）或 3 相反相器电路（6 个输出）。同时，通过连接比较器 INTP20 或 INTP21，可以紧急停止 PWM 输出。

16 位定时器 KB0、KB1 和 KB2 具备以下功能。

(1) PWM 输出

- 定时器运行期间，可以更改 PWM 输出的占空比和周期。
- 可以分别将定时器停止工作时的默认电平和定时器运行时的有效电平设置为高电平和低电平。

(2) A/D 转换开始时序信号输出

可以使用比较寄存器 TKBTGCRn 来输出 A/D 转换开始时序信号。通过该功能，可以实现 16 位定时器 KBn 和 A/D 转换开始时序的同步。

(3) 同时开始/停止模式

通过将 16 位定时器 KB0 设置为主定时器，将 16 位定时器 KB1 和 KB2 设置为从属定时器，可以与定时器 KB0 的计数开始/停止时序保持同步地同时启动/停止 16 位从属定时器 KB1 和 KB2。

(4) 同步开始/清除模式

通过将 16 位定时器 KB0 设置为主定时器，将 16 位定时器 KB1 和 KB2 设置为从属定时器，可以实现主定时器和从属定时器的计数周期的同步。在该模式下，可以生成例如互补 PWM 输出。

(5) PWM 输出门控功能（通过联锁 16 位定时器 KC0）

利用 16 位定时器 KC0 的输出，可以对最多 6 个定时器 KBn 输出（TKCO00 至 TKCO05 输出）进行门控。

(6) 定时器重启功能（通过与比较器和 INTP 联锁）

当出现触发源（比较器 1 至 3 输出、INTP20、INTP21）时，可以直接重启定时器输出（不通过 CPU）。利用此功能，可以实现例如临界传导模式的 PFC 控制。

(7) 强制输出停止功能 1（通过与比较器联锁）

当出现触发源（比较器 0 至 5 输出）时，可以与 16 位定时器 KBn 和 KC0 电路的操作时钟 f_{KBKC} 异步地直接将定时器输出固定为 Hi-Z 电平、高电平或低电平（不通过 CPU）。通过设置强制输出停止功能 1 的停止触发器，可以与 16 位定时器 KBn 和 KC0 电路的操作时钟 f_{KBKC} 同步地取消强制输出停止状态。

(8) 强制输出停止功能 2（通过与比较器和 INTP 联锁）

当出现触发源（比较器 0 至 5 输出、INTP20）时，可以与 16 位定时器 KBn 和 KC0 电路的操作时钟 f_{KBKC} 异步地直接将定时器输出固定为高电平或低电平（不通过 CPU）。出现触发源或者触发源信号变为无效电平后，系统将在下一个计数器周期开始时取消强制输出停止状态。

(9) PWM 输出抖动功能

每 16 个周期中，可以输出 0 至 15 次“设定占空比 + 1”的波形。利用该功能，可以输出计数时钟 16 倍的 PWM 来作为定时器 KBn 的 16 个周期的平均分辨率。

(10) PWM 输出软启动功能

可以执行软启动，以在 PWM 输出开始后自动增加占空比，直至达到所配置的占空比。
可以配置初始占空比和“占空比 + 1”增量周期。

(11) 最大频率限值功能

使用定时器重启功能时，如果在设定的最大频率之前出现触发，则可以在达到设定的最大频率为止保留重启。

(12) 交错式 PFC 输出模式

借助定时器重启功能，可以通过外部因素自动地对两个输出交替进行重启输出。利用临界传导模式，可以进行交错式 PFC 控制。

备注 临界传导模式是一种 PFC 控制方法，通过检测电感电流的零电平来激活可切换的 FET。

7.2 16 位定时器KB0、KB1 和KB2 的配置

16 位定时器 KB0、KB1 和 KB2 由以下硬件构成。

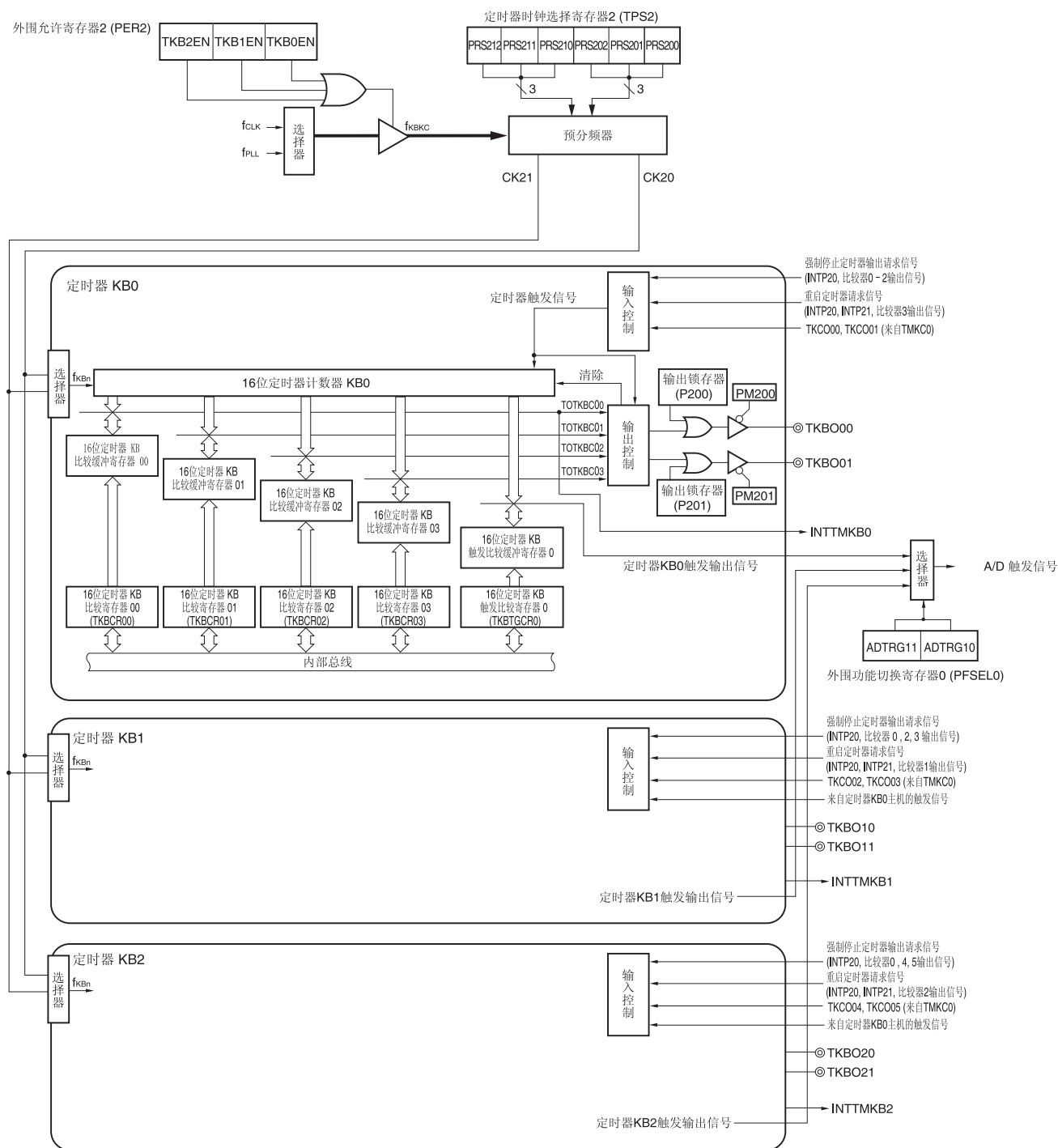
表 7-1. 16 位定时器 KB0、KB1 和 KB2 的配置

项目	配置
定时器/计数器	16 位定时器计数器 KBn (TKBCNTn)
寄存器	16 位定时器 KB 比较寄存器 n0 至 n3 (TKBCRn0 至 TKBCRn3) 16 位定时器 KB 触发比较寄存器 n (TKBTGCRn)
定时器输出	TKBOn0, TKBOn1
控制寄存器	外围允许寄存器 2 (PER2) 定时器时钟选择寄存器 2 (TPS2) 16 位定时器 KB 操作控制寄存器 n0 (TKBCTLn0) 16 位定时器 KB 操作控制寄存器 n1 (TKBCTLn1) 16 位定时器 KB 输出控制寄存器 n0 (TKBIOCn0) 16 位定时器 KB 输出控制寄存器 n1 (TKBIOCn1) 16 位定时器 KB 标志寄存器 n (TKBFLGn) 16 位定时器 KB 触发寄存器 n (TKBTRGn) 16 位定时器 KB 标志清除触发寄存器 n (TKBCLRn) 16 位定时器 KB 抖动计数寄存器 n0, n1 (TKBDNRn0, TKBDNRn1) 16 位定时器 KB 比较 1L&抖动计数寄存器 n0 (TKBCRLDn0) 16 位定时器 KB 比较 3L&抖动计数寄存器 n1 (TKBCRLDn1) 16 位定时器 KB 软启动初始占空比寄存器 n0, n1 (TKBSIRn0, TKBSIRn1) 16 位定时器 KB 软启动步宽寄存器 n0, n1 (TKBSSRn0, TKBSSRn1) 16 位定时器 KB 最大频率限值设置寄存器 n (TKBMFRn) 外围功能切换寄存器 0 (PFSEL0) 端口模式寄存器 20 (PM20) 端口寄存器 20 (P20)

备注 n = 0 至 2

框图如图 7-1 所示。

图 7-1. 16 位定时器 KBn 的框图



- 备注 1. f_{KBKC}: 16 位定时器 KBn 和 KC0 电路整体的操作时钟
 f_{KBn}: 16 位定时器 KBn 的计数时钟
 2. n = 0 至 2

(1) 16 位定时器 KB 比较寄存器 n0 至 n3 (TKBCRn0 至 TKBCRn3)

当定时器计数(TKBCEn = 1)时, 可以刷新 TKBCRnm (写入相同的值) 或者重写它的值。在定时器工作时如果重写 TKBCRnm 的值, 该值将被锁存, 并在下述时序被传送至 TKBCRnm, 从而改变 TKBCRnm 的值。

- 当计数器开始计数操作(TKBCEn = 0)时
- 当出现批量覆盖触发(TKBRDTn = 1)或外部触发(TKBTSEn = 1)时

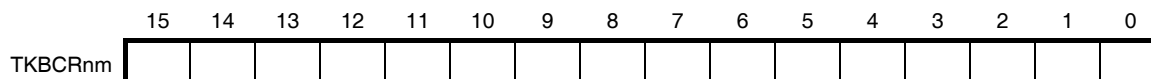
可以按 16 位单位读取或写入。

产生复位信号后, 清除该寄存器为 0000H。

图 7-2. 16 位定时器 KB 比较寄存器 nm (TKBCRnm)的格式

地址: F0600H (TKBCR00), F0602H (TKBCR01), F0604H (TKBCR02), F0606H (TKBCR03),
F0640H (TKBCR10), F0642H (TKBCR11), F0644H (TKBCR12), F0646H (TKBCR13),
F0680H (TKBCR20), F0682H (TKBCR21), F0684H (TKBCR22), F0686H (TKBCR23)

复位后: 0000H R/W



备注 n = 0 至 2, m = 0 至 3

(2) 16 位定时器 KB 触发比较寄存器 n (TKBTGCRn)

当定时器计数(TKBCEn = 1)时, 可以刷新 TKBTGCRn (写入相同的值) 或者重写它的值。在定时器工作时如果重写 TKBCRnm 的值, 该值将被锁存, 并在下述时序被传送至 TKBCRnm, 从而改变 TKBCRnm 的值。

- 当计数器开始计数操作(TKBCEn = 0)时
- 当出现批量覆盖触发(TKBRDTn = 1)或外部触发(TKBTSEn = 1)时

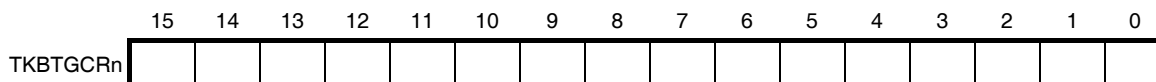
来自该寄存器的周期信号可用作 A/D 转换的硬件触发。

可以按 16 位单位读取或写入。

产生复位信号后, 清除该寄存器为 0000H。

图 7-3. 16 位定时器 KB 触发比较寄存器 n (TKBTGCRn)的格式

地址: F0608H (TKBTGCR0), F0648H (TKBTGCR1), F0688H (TKBTGCR2) 复位后: 0000H R/W



备注 n = 0 至 2

7.3 控制 16 位定时器KB0、KB1 和KB2 的寄存器

控制 16 位定时器 KB0、KB1 和 KB2 的寄存器如下所示。

- 外围允许寄存器 2 (PER2)
- 定时器时钟选择寄存器 2 (TPS2)
- 16 位定时器 KB 操作控制寄存器 n0, n1 (TKBCTLn0, TKBCTLn1)
- 16 位定时器 KB 输出控制寄存器 n0, n1 (TKBIOCn0, TKBIOCn1)
- 16 位定时器 KB 标志寄存器 n (TKBFLGn)
- 16 位定时器 KB 触发寄存器 n (TKBTRGn)
- 16 位定时器 KB 标志清除触发寄存器 n (TKBCLRn)
- 16 位定时器 KB 抖动计数寄存器 n0, n1 (TKBDNRn0, TKBDNRn1)
- 16 位定时器 KB 比较 1L&抖动计数寄存器 n0 (TKBCRLDn0)
- 16 位定时器 KB 比较 3L&抖动计数寄存器 n1 (TKBCRLDn1)
- 16 位定时器 KB 软启动初始占空比寄存器 n0, n1 (TKBSIRn0, TKBSIRn1)
- 16 位定时器 KB 软启动步宽寄存器 n0, n1 (TKBSSRn0, TKBSSRn1)
- 16 位定时器 KB 最大频率限值设置寄存器 n (TKBMFRn)
- 外围功能切换寄存器 0 (PFSEL0)
- 端口模式寄存器 20 (PM20)
- 端口寄存器 20 (P20)

(1) 外围允许寄存器 2 (PER2)

该寄存器用于允许或禁止供应时钟至外围硬件。并且中止向未使用的硬件供应时钟，以减少电力消耗和噪声。

当使用定时器 KB0、KB1 和 KB2 时，务必将该寄存器的位 2 至 0 (TKB2EN 至 TKB0EN) 设置为 1。

使用 8 位存储器操作指令设置 PER2 寄存器。

产生复位信号后，清除该寄存器为 00H。

图 7-4. 外围允许寄存器 2 (PER2)的格式

地址: F0509H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
PER2	PGACMPEN	TKBPA2EN	TKBPA1EN	TKBPA0EN	TKC0EN	TKB2EN	TKB1EN	TKB0EN

TKBnEN	定时器KBn输入时钟的控制
0	停止输入时钟的供应。 <ul style="list-style-type: none"> 不可写入用于定时器KBn的SFR。 定时器KBn处于复位状态。
1	供应输入时钟。 <ul style="list-style-type: none"> 可以读取/写入用于定时器KBn的SFR。

注意事项 要设置定时器 KBn，务必首先将 TKBnEN 位设置为“1”。如果 TKBnEN = 0，则对定时器 KBn 的控制寄存器进行的写操作将被忽略，所有读取值均为初始值（定时器时钟选择寄存器 2 (TPS2)、外围功能切换寄存器 0 (PFSEL0)、端口模式寄存器 20 (PM20)和端口寄存器 20 (P20)除外）。

(2) 定时器时钟选择寄存器 2 (TPS2)

TPS2 寄存器为 16 位寄存器，用于选择通常由外部预分频器供应至定时器 KB0、KB1、KB2 和 KC0 的两种操作时钟 (CK20、CK21)。使用 TPS2 寄存器的位 6 至位 4 选择 CK21，使用位 2 至位 0 选择 CK20。

只有在以下情况下才可以可定时器操作期间重写 TPS2 寄存器。

如果可以重写 PRS200 至 PRS202 位($n = 0$ 至 2):

选择 CK20 作为操作时钟($TKBCKSn = 0$, $TKCCKS0 = 0$)的所有通道处于停止状态($TKBCEn = 0$, $TKCCE0 = 0$)。

如果可以重写 PRS210 至 PRS212 位($n = 0$ 至 2):

选择 CK21 作为操作时钟($TKBCKSn = 1$, $TKCCKS0 = 1$)的所有通道处于停止状态($TKBCEn = 0$, $TKCCE0 = 0$)。

使用 8 位存储器操作指令来设置 TPS2 寄存器。

产生复位信号后，清除该寄存器为 00H。

图 7-5. 定时器时钟选择寄存器 2 (TPS2)的格式

地址: F05C5H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
TPS2	0	TPS 212	TPS 211	TPS 210	0	TPS 202	TPS 201	TPS 200

TPS 2k2	TPS 2k1	TPS 2k0	操作时钟(CK2k)的选择 ^{#1, 2} ($k = 0, 1$)					
			$f_{CLK} =$ 2 MHz	$f_{CLK} =$ 5 MHz	$f_{CLK} =$ 20 MHz	$f_{CLK} =$ 32 MHz	$f_{PLL} =$ 64 MHz	
0	0	0	f_{CLK}, f_{PLL}	2 MHz	5 MHz	20 MHz	32 MHz	64 MHz
0	0	1	$f_{CLK}/2, f_{PLL}/2$	1 MHz	2.5 MHz	10 MHz	16 MHz	32 MHz
0	1	0	$f_{CLK}/2^2, f_{PLL}/2^2$	500 kHz	1.25 MHz	5 MHz	8 MHz	16 MHz
0	1	1	$f_{CLK}/2^3, f_{PLL}/2^3$	250 kHz	625 kHz	2.5 MHz	4 MHz	8 MHz
1	0	0	$f_{CLK}/2^4, f_{PLL}/2^4$	125 kHz	312.5 kHz	1.25 MHz	2 MHz	4 MHz
1	0	1	$f_{CLK}/2^5, f_{PLL}/2^5$	62.5 kHz	156.2 kHz	625 kHz	1 MHz	2 MHz
1	1	0	禁止设置	-	-	-	-	-
1	1	1	禁止设置	-	-	-	-	-

注 1. 要更改被选作 f_{CLK} 的时钟（通过更改系统时钟控制寄存器(CKC)的值）时，须停止 KB0、KB1、KB2 和 KC0($TKBCEn = 0$, $TKCCE0 = 0$)。

2. 当 PLL 控制寄存器(PLLCTL)中 $PLLON = 1$ 时，供应 f_{PLL} 。

注意事项 务必将位 7 和 3 清除为“0”。

备注 f_{CLK} : CPU/外围硬件时钟频率

f_{PLL} : PLL 输出时钟

(3) 16 位定时器 KB 操作控制寄存器 n0 (TKBCTLn0)

TKBCTLn0 寄存器用于控制计数操作以及设置 16 位定时器的计数时钟。

使用 16 位存储器操作指令设置 TKBCTLn0 寄存器。

产生复位信号后，清除该寄存器为 0000H。

图 7-6. 16 位定时器 KB 操作控制寄存器 n0 (TKBCTLn0)的格式 (1/2)

地址: F0622H (TKBCTL00), F0662H (TKBCTL10), F06A2H (TKBCTL20) 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8
TKBCTLn0	0	TKBGTEn1	TKBSSEn1	TKBDIEn1	0	TKBGTEn0	TKBSSEn0	TKBDIEn0
	7	6	5	4	3	2	1	0
	TKBMFEn	0	TKBIRSn1	TKBIRSn0	0	TKBTSEn	TKBSTSn1	TKBSTSn0
TKBGTEnp	通过定时器KC0输出来控制TKBOnp的PWM输出门控功能							
0	不使用PWM输出门控功能。							
1	使用PWM输出门控功能。							
TKBSSEnp	TKBOnp的PWM输出软启动功能的控制							
0	不使用PWM输出软启动功能。							
1	使用PWM输出软启动功能。							
TKBDIEnp	TKBOnp的PWM输出抖动功能的控制							
0	不使用PWM输出抖动功能。							
1	使用PWM输出抖动功能。							
TKBMFEn	TKBOn0和TKBOn1最大频率限值功能的控制							
0	不使用最大频率限值功能。							
1	使用最大频率限值功能。							
TKBIRn1	TKBIRn0	交错式PFC输出模式时立即输出TKBOn1的INTP21输入的允许范围的配置						
0	0	T/2至T/2+T/64						
0	1	T/2至T/2+T/32						
1	0	T/2至T/2+T/16						
1	1	T/2至T/2+ T/8						
TKBTSEn	由外部触发设置的比较寄存器批量覆盖功能的控制							
0	不使用由外部触发设置的比较寄存器批量覆盖功能。							
1	使用由外部触发设置的比较寄存器批量覆盖功能。							

备注 1. n = 0 至 2, p = 0, 1

2. T 是上一次的重启周期。

图 7-6. 16 位定时器 KB 操作控制寄存器 n0 (TKBCTLn0)的格式 (2/2)

TKBSTSn1	TKBSTSn0	定时器KBn计数开始触发的选择
0	0	不使用触发输入。
0	1	外部中断信号(INTP20)
1	0	外部中断信号(INTP21)
1	1	当n = 0时: 比较器3检测信号 当n = 1时: 比较器1检测信号 当n = 2时: 比较器2检测信号

- 注意事项**
1. 在定时器操作期间，禁止设置 TKBCTLn0 寄存器的其它位。不过，可以刷新 TKBCTLn0 寄存器（写入相同的值）。
 2. 务必将位 15、11、6 和 3 清除为“0”。
 3. 使用 TKBOnp 的 PWM 输出门控功能时，将 TKCIOC01 寄存器的对应位 5 至 0(TKCTOE05 至 TKCTOE00)设置为 1（允许 TKCO05 至 TKCO00 输出）。

备注 n = 0 至 2, p = 0, 1

(4) 16 位定时器 KB 操作控制寄存器 n1 (TKBCTLn1)

TKBCTLn1 寄存器用于控制计数操作以及设置 16 位定时器的计数时钟。

使用 1 位或 8 位存储器操作指令来设置 TKBCTLn1 寄存器。

产生复位信号后，清除该寄存器为 00H。

图 7-7. 16 位定时器 KB 操作控制寄存器 n1 (TKBCTLn1)的格式

地址: F0629H (TKBCTL01), F0669H (TKBCTL11), F06A9H (TKBCTL21) 复位后: 00H R/W

符号	<7>	6	5	4	3	2	1	0
TKBCTL01	TKBCE0	0	0	TKBCKS0	TKBSCM0	0	TKBMD01	TKBMD00

符号	<7>	6	5	4	3	2	1	0
TKBCTLm1	TKBCEm	0	0	TKBCKSm	0	0	TKBMDm1	TKBMDm0

(m = 1, 2)

TKBCEn	定时器KBn操作的控制
0	停止定时器操作（计数器被设置为FFFF）。
1	允许定时器计数操作。

TKBCKS0	定时器KB0时钟的选择
0	通过TPS202至TPS200位选择CK20时钟
1	通过TPS212至TPS210位选择CK21时钟

TKBSCMn	定时器KBn开始操作的控制
0	工作于通过TKBCKSn位选择的时钟
1	计数开始时序是在CK20和CK21时钟匹配时。操作开始后，工作于通过TKBCKSn位选择的时钟。
注意事项 利用TKBSCMn位将从属设置为同时开始模式的话，可以使从属和主的开始时序一致。	

TKBMDn1	TKBMDn0	定时器KBn操作模式的选择
0	0	独立模式（用于主）
0	1	同步开始/停止模式（用于从属）
1	0	同步开始/清除模式（用于从属）
1	1	交错式PFC输出模式

- 注意事项 1. 在定时器操作期间，禁止设置 TKBCTLn1 寄存器的其它位。不过，可以刷新 TKBCTLn1 寄存器（写入相同的值）。
2. 务必将 TKBCTL01 的位 6、5 和 2 清除为“0”。
3. 务必将 TKBCTLm1 的位 6、5、3 和 2 清除为“0”。

备注 n = 0 至 2, m = 1, 2

(5) 16 位定时器 KB 输出控制寄存器 n0 (TKBIOCn0)

TKBIOCn0 寄存器用于设置 16 位定时器 KBn 输出(TKBOnp)的默认电平/有效电平。

使用 1 位或 8 位存储器操作指令来设置 TKBIOCn0 寄存器。

产生复位信号后，清除该寄存器为 00H。

图 7-8. 16 位定时器 KB 输出控制寄存器 n0 (TKBIOCn0)的格式

地址: F0626H (TKBIOC00), F0666H (TKBIOC10), F06A6H (TKBIOC20) 复位后: 00H R/W

符号	7	6	5	4	<3>	<2>	<1>	<0>
TKBIOCn0	0	0	0	0	TKBTOLn1	TKBTOLn0	TKBTODn1	TKBTODn0

TKBTOLnp	定时器输出TKBOnp的有效电平设置	
0	高电平	
1	低电平	

TKBTODnp	定时器输出TKBOnp的默认电平设置	
0	低电平	
1	高电平	

- 注意事项**
1. 在定时器操作期间，禁止设置 TKBIOCn0 寄存器的其它位。不过，可以刷新 TKBIOCn0 寄存器（写入相同的值）。
 2. 务必将位 7 至 4 清除为“0”。
 3. TKBOnp 输出以及兼用端口的端口模式寄存器(PMxx)和端口寄存器(Pxx)共同决定着实际的 TKBOnp 引脚输出。

备注 n = 0 至 2, p = 0, 1

(6) 16 位定时器 KB 输出控制寄存器 n1 (TKBIOCn1)

TKBIOCn1 寄存器用于控制 16 位定时器 KBn 输出(TKBOnp)的输出禁止/允许。

使用 1 位或 8 位存储器操作指令来设置 TKBIOCn1 寄存器。

产生复位信号后，清除该寄存器为 00H。

图 7-9. 16 位定时器 KB 输出控制寄存器 n1 (TKBIOCn1)的格式

地址: F0628H (TKBIOC01), F0668H (TKBIOC11), F06A8H (TKBIOC21) 复位后: 00H R/W

符号	7	6	5	4	3	2	<1>	<0>
TKBIOCn1	0	0	0	0	0	0	TKBTOEn1	TKBTOEn0

TKBTOEnp	定时器输出TKBOnp的输出允许/禁止
0	禁止定时器输出(当TKBTODnp = 0时低电平输出, 当TKBTODnp = 1时高电平输出。)
1	允许定时器输出

- 注意事项
1. 定时器操作期间，可以覆盖 TKBIOCn1 寄存器。
 2. 务必将位 7 至 2 清除为“0”。
 3. TKBOnp 输出以及兼用端口的端口模式寄存器(PMxx)和端口寄存器(Pxx)共同决定着实际的 TKBOnp 引脚输出。

备注 n = 0 至 2, p = 0, 1

(7) 16 位定时器 KB 标志寄存器 n (TKBFLGn)

TKBFLGn 寄存器用于显示 16 位定时器 KBn 的状态标志。

使用 1 位或 8 位存储器操作指令来读取 TKBFLGn 寄存器。

产生复位信号后，清除该寄存器为 00H。

图 7-10. 16 位定时器 KB 标志寄存器 n (TKBFLGn)的格式

地址: F0613H (TKBFLG0), F0653H (TKBFLG1), F0693H (TKBFLG2) 复位后: 00H R

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
TKBFLGn	TKBSSF _{n1}	TKBSSF _{n0}	TKBSEF _{n1}	TKBSEF _{n0}	TKBIRF _n	TKBIEF _n	TKBMFF _n	TKBRSF _n
TKBSSF _{n_p}	TKBOn _p 引脚的PWM输出软启动功能的状态标志							
0	PWM输出软启动功能停止期间							
1	正在执行PWM输出软启动功能							
TKBSEF _{n_p}	TKBOn _p 引脚的PWM输出软启动功能的错误标志							
0	无错误或由TKBCLIR _n 清除完成							
1	错误（执行PWM输出软启动(TKBSSF _{n_p} = 0)时发生了TKBRDT _{n_p} = 1）							
TKBIRF _n	交错式PFC模式时INTP20触发未检测错误标志							
0	无错误或由TKBCLIR _n 清除完成							
1	错误（在0至T/2以及TKBIRS _{n1} 和TKBIRS _{n0} 设置的判断范围内未检测到INTP20触发）							
TKBIEF _n	交错式PFC模式时INTP21触发多重检测错误标志							
0	无错误或由TKBCLIE _n 清除完成							
1	错误（TKBOn1有效输出期间检测到另一个INTP21触发）							
TKBMFF _n	最大频率限值功能的状态标志							
0	未发生最大频率限值功能或由TKBCLMF _n 清除完成							
1	发生最大频率限值功能							
TKBRSF _n	批量覆盖触发的保留状态标志							
0	批量覆盖允许状态，或由批量覆盖触发的批量覆盖完成							
1	由写入批量覆盖触发位TKBRDT _n 所引发的批量覆盖保留（等待完成）状态							

- 备注 1. n = 0 至 2, p = 0, 1
2. T 是上一次的重启周期。

(8) 16 位定时器 KB 触发寄存器 n (TKBTRGn)

TKBTRGn 是触发寄存器，用于批量覆盖 16 位定时器 KBn 的比较寄存器。

使用 1 位或 8 位存储器操作指令来写入 TKBTRGn 寄存器。

产生复位信号后，清除该寄存器为 00H。

图 7-11. 16 位定时器 KB 触发寄存器 n (TKBTRGn)的格式

地址: F0612H (TKBTRG0), F0652H (TKBTRG1), F0692H (TKBTRG2) 复位后: 00H W

符号	7	6	5	4	3	2	1	<0>
TKBTRGn	0	0	0	0	0	0	0	TKBRDTn

TKBRDTn	比较寄存器的批量覆盖请求触发
0	设置无效
1	比较寄存器的批量覆盖请求

备注 n = 0 至 2

(9) 16 位定时器 KB 标志清除触发寄存器 n (TKBCLRn)

TKBCLRn 寄存器用于清除 16 位定时器 KB 标志寄存器 n (TKBFLGn)的标志。

使用 1 位或 8 位存储器操作指令来写入 TKBCLRn 寄存器。

产生复位信号后，清除该寄存器为 00H。

图 7-12. 16 位定时器 KB 标志清除触发寄存器 n (TKBCLRn)的格式

地址: F0627H (TKBCLR0), F0667H (TKBCLR1), F06A7H (TKBCLR2) 复位后: 00H W

符号	7	6	<5>	<4>	<3>	<2>	<1>	0
TKBCLRn	0	0	TKBCLSEn1	TKBCLSEn0	TKBCLIRn	TKBCLIEn	TKBCLMFn	0

TKBCLSEnp	用于清除TKBOnp引脚的PWM输出软启动功能错误标志的触发
0	设置无效
1	将TKBSEFnp标志清除为“0”。

TKBCLIRn	用于清除交错式PFC模式时INTP20触发未检测错误标志的触发
0	设置无效
1	将TKBIRFn标志清除为“0”。

TKBCLIEn	用于清除交错式PFC模式时INTP21触发多重检测错误标志的触发
0	设置无效
1	将TKBIEFn标志清除为“0”。

TKBCLMFn	用于清除最大频率限值功能的状态标志的触发
0	设置无效
1	将TKBMFFn标志清除为“0”。

备注 n = 0 至 2, p = 0, 1

(10) 16 位定时器 KB 抖动计数寄存器 n0, n1 (TKBDNRn0, TKBDNRn1)

TKBDNRnp 寄存器用于 TKBOnp 输出的 PWM 抖动功能。

该寄存器的高 4 位的值为 N (N = 0H 至 FH)时, 在每 16 个 PWM 输出周期中, 输出 N 次有效期间 (有效期间延长 1 个计数时钟)。

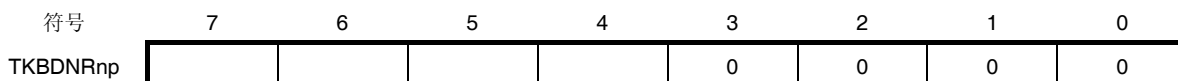
表 7-2 显示了 TKBDNRnp 设置与延长了 1 个计数时钟的有效期间的重复次数 N 之间的关系。

使用 8 位存储器操作指令设置 TKBDNRnp 寄存器。

产生复位信号后, 清除该寄存器为 00H。

图 7-13. 16 位定时器 KB 抖动计数寄存器 np (TKBDNRnp)的格式

地址: F060EH (TKBDNR00), F064EH (TKBDNR10), F068EH (TKBDNR20) 复位后: 00H R/W
 F0610H (TKBDNR01), F0650H (TKBDNR11), F0690H (TKBDNR21)



注意事项 务必将位 3 至 0 清除为“0”。

备注 n = 0 至 2, p = 0, 1

表 7-2. 16 位定时器 KB 抖动计数寄存器 np (TKBDNRnp)的设置

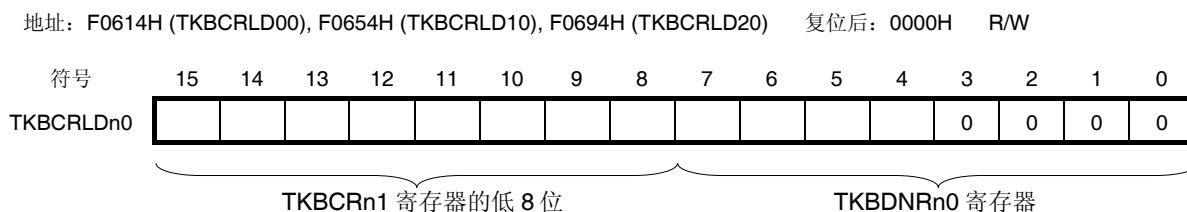
周期 次数(N)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0																
1	■															
2	■							■								
3	■				■			■								
4	■			■	■			■				■				
5	■		■	■	■			■				■				
6	■		■	■	■			■			■					
7	■		■	■	■		■				■					
8	■		■	■	■		■				■				■	
9	■	■	■	■	■		■				■				■	
10	■	■	■	■	■		■			■					■	
11	■	■	■	■	■		■			■					■	
12	■	■	■	■	■		■			■			■		■	
13	■	■	■	■	■		■			■			■		■	
14	■	■	■	■	■		■			■		■		■	■	
15	■	■	■	■	■		■	■		■		■		■	■	

- 备注 1. 单元格的周期: 通过 TKB0CR1 和 TKB0CR3 寄存器的设置值使输出无效
- 单元格的周期: 通过“TKB0CR1 和 TKB0CR3 寄存器的设置值 + 1”使输出无效
- 2. n = 0 至 2, p = 0, 1

(11) 16 位定时器 KB 比较 1L&抖动计数寄存器 n0 (TKBCRLDn0)

在 TKBCRLDn0 寄存器的高 8 位存储“TKBCRn1 寄存器的低 8 位”值，在低 8 位存储“TKBDNRn0 寄存器”值。
 使用 16 位存储器操作指令设置 TKBCRLDn0 寄存器。
 产生复位信号后，清除该寄存器为 0000H。

图 7-14. 16 位定时器 KB 比较 1L&抖动计数寄存器 n0 (TKBCRLDn0)的格式



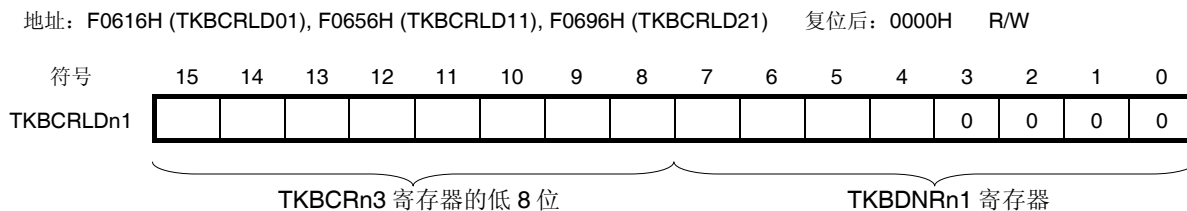
注意事项 务必将位 3 至 0 清除为“0”。

备注 n = 0 至 2

(12) 16 位定时器 KB 比较 3L&抖动计数寄存器 n1 (TKBCRLDn1)

在 TKBCRLDn1 寄存器的高 8 位存储“TKBCRn3 寄存器的低 8 位”值，在低 8 位存储“TKBDNRn1 寄存器”值。
 使用 16 位存储器操作指令设置 TKBCRLDn1 寄存器。
 产生复位信号后，清除该寄存器为 0000H。

图 7-15. 16 位定时器 KB 比较 3L&抖动计数寄存器 n1 (TKBCRLDn1)的格式



注意事项 务必将位 3 至 0 清除为“0”。

备注 n = 0 至 2

(13) 16 位定时器 KB 软启动初始占空比寄存器 n0, n1 (TKBSIRn0, TKBSIRn1)

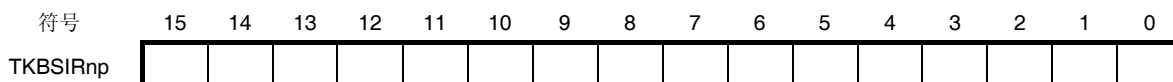
TKBSIRnp 寄存器用于设置 TKBOnp 输出的 PWM 输出软启动功能的初始占空比。

使用 16 位存储器操作指令设置 TKBSIRnp 寄存器。

产生复位信号后，清除该寄存器为 0000H。

图 7-16. 16 位定时器 KB 软启动初始占空比寄存器 np (TKBSIRnp)的格式

地址: F060AH (TKBSIR00), F064AH (TKBSIR10), F068AH (TKBSIR20) 复位后: 0000H R/W
F060CH (TKBSIR01), F064CH (TKBSIR11), F068CH (TKBSIR21)



备注 n = 0 至 2, p = 0, 1

(14) 16 位定时器 KB 软启动步宽寄存器 n0, n1 (TKBSSRn0, TKBSSRn1)

TKBSIRnp 寄存器用于 TKBOnp 输出的 PWM 输出软启动功能。

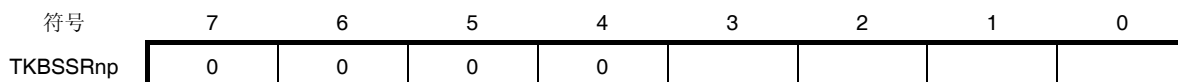
当该寄存器的值为 N (N = 0000B 至 1111B)时，由 TKBSIRnp 所设置的有效输出期间的 PWM 将被输出 N + 1 次。之后，继续输出 N + 1 次的(有效期间 + 1 个时钟)波形，然后是 N + 1 次的(有效期间 + 2 个时钟)波形，以此类推。最终，当具有与 TKBCRn1 或 TKBCRn3 相同的占空比时，解除 PWM 输出软启动功能，并设置为正常的 PWM 输出。

使用 8 位存储器操作指令设置 TKBSSRnp 寄存器。

产生复位信号后，清除该寄存器为 00H。

图 7-17. 16 位定时器 KB 软启动步宽寄存器 np (TKBSSRnp)的格式

地址: F060FH (TKBSSR00), F064FH (TKBSSR10), F068FH (TKBSSR20) 复位后: 00H R/W
F0611H (TKBSSR01), F0651H (TKBSSR11), F0691H (TKBSSR21)



注意事项 务必将位 7 至 4 清除为“0”。

备注 n = 0 至 2, p = 0, 1

(15) 16 位定时器 KB 最大频率限值设置寄存器 n (TKBMFRn)

TKBMFRn 寄存器用于设置外部触发的定时器重启最小周期。

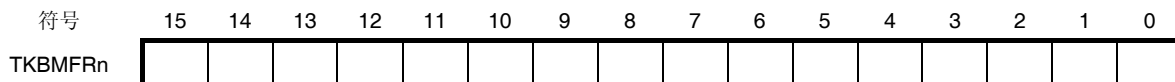
当计数器(TKBCNTn)的值小于 TKBMFRn 的值时，如果检测到触发输入，触发将被保留，在计数达到 TKBMFRn 的设置值后，将清除（重启）计数器(TKBCNTn)。

使用 16 位存储器操作指令设置 TKBMFRn 寄存器。

产生复位信号后，清除该寄存器为 0000H。

图 7-18. 16 位定时器 KB 最大频率限值设置寄存器 n (TKBMFRn)的格式

地址: F0624H (TKBMFR0), F0664H (TKBMFR1), F06A4H (TKBMFR2) 复位后: 0000H R/W



备注 n = 0 至 2

(16) 外围功能切换寄存器 0 (PFSEL0)

PFSEL0 寄存器用于设置 16 位定时器 KB0、KB1 和 KB2 与外围功能的输入/输出。

位 0 和位 1 通过外部中断 INTP20 和 INTP21 来实现 16 位定时器 KB0、KB1 和 KB2 的 PWM 控制或清除停止模式。

位 2 和位 3 选择用于 A/D 转换的定时器触发。

位 4 选择是否要使用 INTP20 噪声滤波器。在强制输出停止功能 2 中使用 INTP20 时，选择不使用噪声滤波器，以加快触发输入和停止输出之间的反应时间。

使用 1 位或 8 位存储器操作指令来写入 PFSEL0 寄存器。

产生复位信号后，清除该寄存器为 00H。

备注 在强制输出停止功能 2 中使用 INTP20 时，选择不使用噪声滤波器，以缩短从触发输入到输出终止为止的响应间隔。

图 7-19. 外围功能切换寄存器 0 (PFSEL0)的格式

地址: F05C6H 复位后: 00H R/W

符号	7	<6>	<5>	<4>	3	2	<1>	<0>
PFSEL0	0	CMP2STEN	CMP0STEN	PNFEN	ADTRG11	ADTRG10	TMRSTEN1	TMRSTEN0

CMP2STEN	CMP0STEN	比较器中断的切换
参阅第十四章 比较器。		

PNFEN	是否使用外部中断INTP20的噪声滤波器
0	使用噪声滤波器
1	不使用噪声滤波器

ADTRG11	ADTRG10	A/D转换的定时器触发的选择
0	0	定时器KB0的触发源
0	1	定时器KB1的触发源
1	0	定时器KB2的触发源
1	1	禁止设置

TMRSTEN1	外部中断INTP21的用途的切换
0	可用于清除STOP模式 (不可用于定时器重启功能)
1	可用于定时器重启功能(不可用于清除STOP模式)

TMRSTEN0	外部中断INTP20的用途的切换
0	可用于清除STOP模式 (不可用于定时器重启功能)
1	可用于定时器重启功能(不可用于清除STOP模式)

(17) 端口模式寄存器 20 (PM20)

该寄存器按 1 位单位指定端口 20 的输入模式或输出模式。

将 P200/TKBO00/INTP22，P201/TKBO01，P202/TKBO10/(INTP21)，P203/TKBO11/TKCO02/(INTP20)，P204/TKBO20/TKCO03 和 P205/TKBO21/TKCO04/DALITxD4 引脚作为定时器输出使用时，请将 PM200 至 PM205 以及 P200 至 P205 的输出锁存器设置为 0。

使用 1 位或 8 位存储器操作指令来设置 PM20 寄存器。

产生复位信号后，设置该寄存器为 FFH。

图 7-20. 端口模式寄存器 20 (PM20)的格式 (38 引脚产品)

地址: F0510H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM20	1	PM206	PM205	PM204	PM203	PM202	PM201	PM200
s	选择P20n引脚的输入/输出模式 (n = 0至6)							
0	输出模式 (输出缓冲器on)							
1	输入模式 (输出缓冲器off)							

注意事项 务必将 PM20 寄存器的位 7 设置为“1”。

对于 20 引脚产品，在解除复位后，必须通过软件将 PM20 寄存器的位 4 至 6 设置为输出模式 (通过设置端口寄存器和端口模式寄存器为 0 来指定)。

备注 上图展示的是 38 引脚产品的端口模式寄存器 20 的格式。关于其它产品的端口模式寄存器的格式，请参阅表 4-13. 配置在各个产品上的 PMxx、Pxx、PUxx、PIMxx、POMxx、PMCx 寄存器及其对应的位。

7.4 16 位定时器KB0、KB1 和KB2 的操作

以下说明 16 位定时器 KB0、KB1 和 KB2 的操作规格。

- 计数器基本操作 (参阅 7.4.1)
- 默认电平和有效电平 (参阅 7.4.2)
- 停止/重启操作 (参阅 7.4.3)
- 批量覆盖 (参阅 7.4.4)

16 位定时器 KB0、KB1 和 KB2 备有如下的 6 种操作模式。

- 单机模式 (通过 TKBCRn0 的周期控制) (参阅 7.4.5)
- 单机模式 (通过外部触发输入的周期控制) (参阅 7.4.6)
- 同时开始/停止模式 (通过 TKBCRn0 的周期控制) (参阅 7.4.7)
- 同时开始/停止模式 (通过外部触发输入的周期控制) (参阅 7.4.7)
- 同步开始/清除模式 (通过主的周期控制) (参阅 7.4.8)
- 交错式 PFC 输出模式 (参阅 7.4.9)

图 7-21. 定时器 KB 操作设置示例 (开始操作流程图)

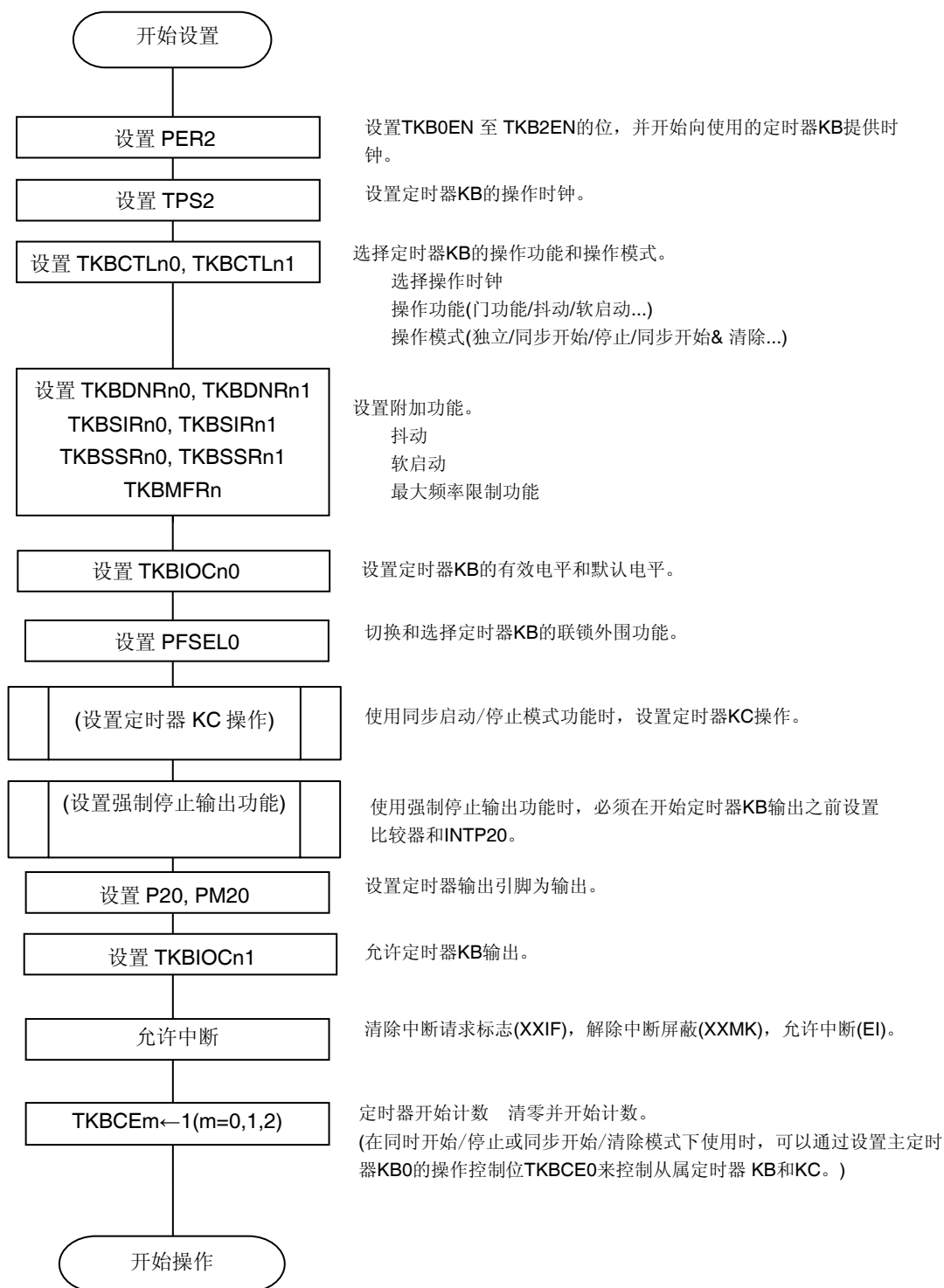


图 7-22. 定时器 KB 操作设置示例 (停止操作流程图)

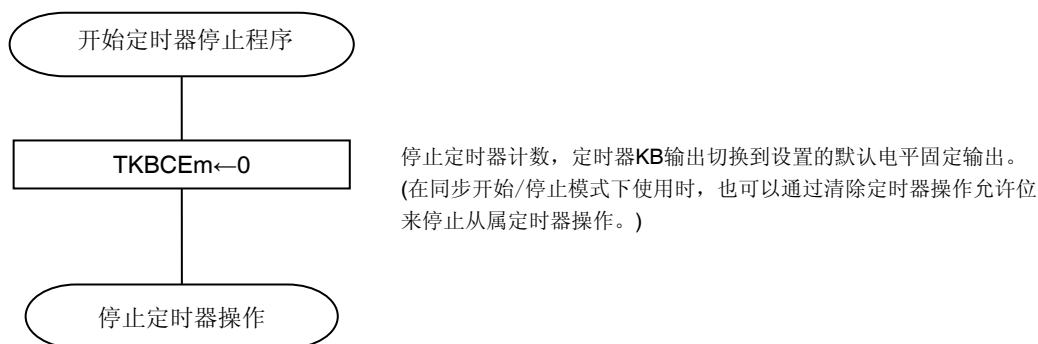


图 7-23. 定时器 KB 操作设置示例 (比较寄存器批量覆盖流程图)



备注 在定时器 KB 操作中更改定时器计数操作定时，使用批量处理功能。设定值将在下一次重启后生效。

7.4.1 计数器基本操作

(1) 计数开始操作

在任何模式下，KB 定时器的 16 位计数器都从初始值 FFFFH 开始计数。计数器的累加方式为 FFFFH 至 0000H、0001H、0002H、0003H 等等。

(2) 清除操作

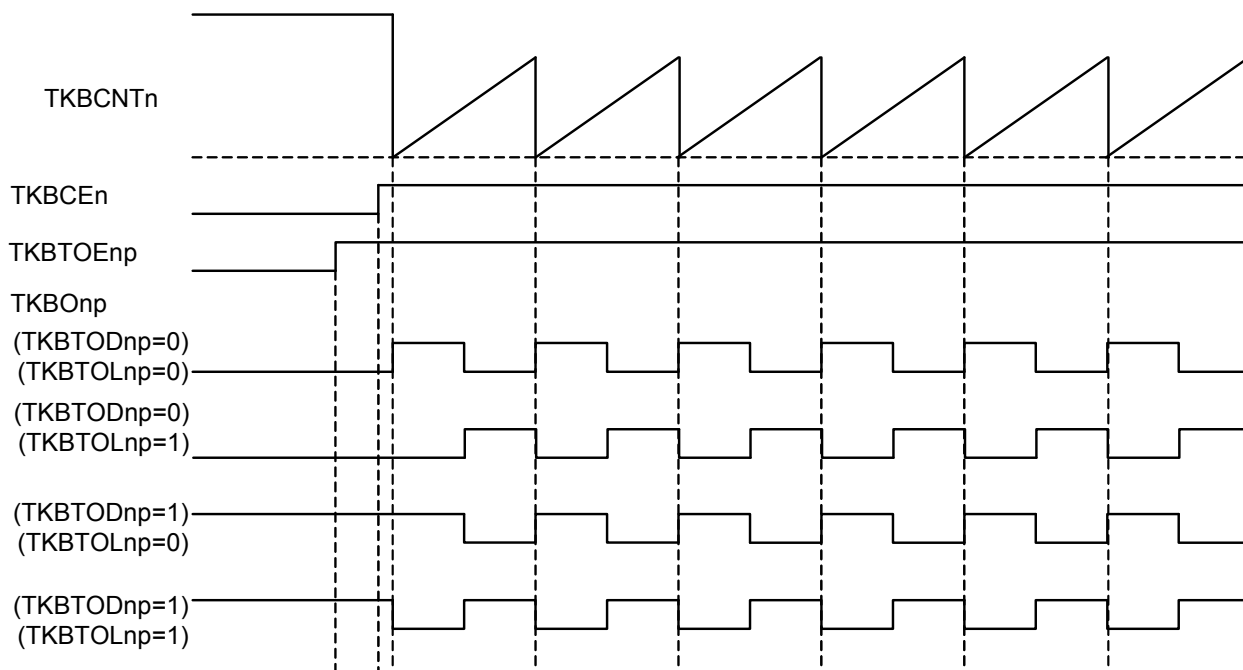
当 16 位计数器值与 TKBCRn0 中定义的值匹配时，或者当周期由外部触发决定的情况下发生外部触发时，16 位计数器将复位为 0000H。如果在计数器与 TKBCRn0 中定义的值匹配时清除计数器，则会产生 INTTKBn 中断，但由于外部触发而清除计数器时则不出现该中断。

7.4.2 默认电平和有效电平

(1) 基本操作

通过 16 位定时器 KB 输出控制寄存器 n0 (TKBIOCn0)，可以设置定时器 KB 输出的默认电平和有效电平。

图 7-24. 默认电平和有效电平的时序图
(基本操作)



当 TKBTOEn 从“0”改变为“1”时，允许 TKBOnp 输出，根据 TKBTOLnp 的设置输出 PWM 波形。
当 TKBTOEn 从“1”改变为“0”时，禁止 TKBOnp 输出，根据 TKBTODnp 的设置输出默认电平。

(2) **TKBTOEnp** 由“0”改变为“1”

定时器计数器操作期间，如果在计数器 **TKBCNTn** 的值与比较寄存器 **TKBCRnp** 的值匹配之前将 **TKBTOEnp** 从 0 更改为 1，则定时器输出将依据 **TKBTOLnp** 的设置匹配时序输出 PWM 波形。

如果在计数器 **TKBCNTn** 的值与比较寄存器 **TKBCRnp** 的值匹配之后将 **TKBTOEnp** 从 0 更改为 1，则直到下一个匹配时序为止定时器输出将保持默认电平。

图 7-25. 默认电平和有效电平的时序图
(在计数器和比较寄存器(TKBCRn1 至 TKBCRn3)匹配之前将 **TKBTOEnp** 由“0”更改为“1”时)

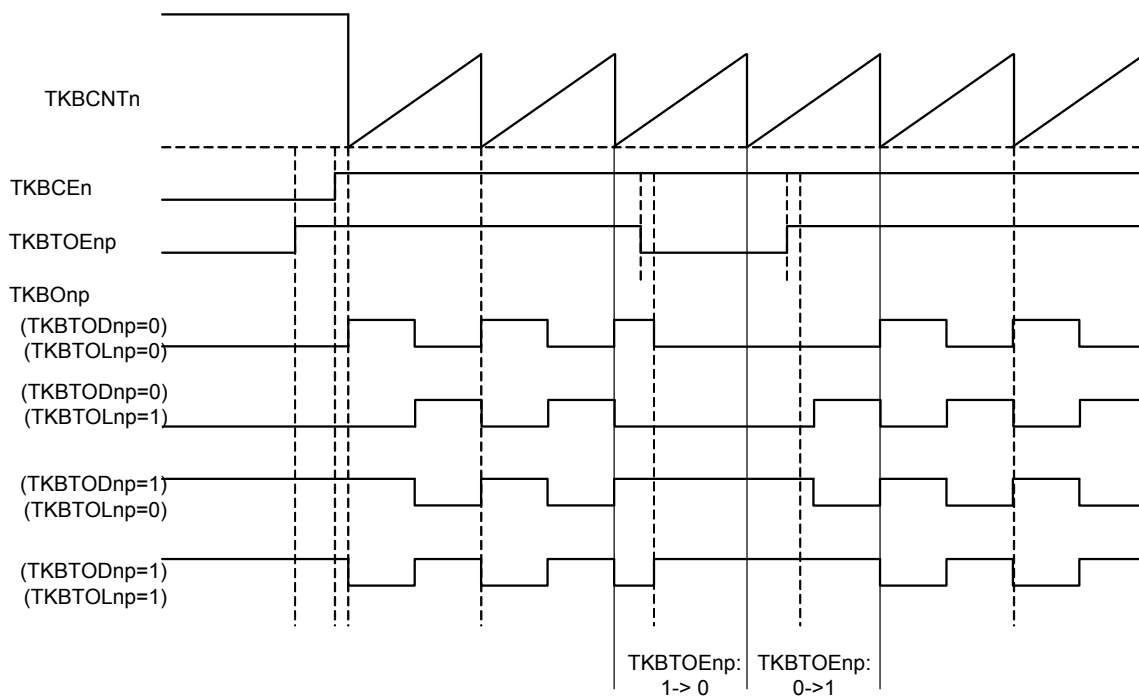
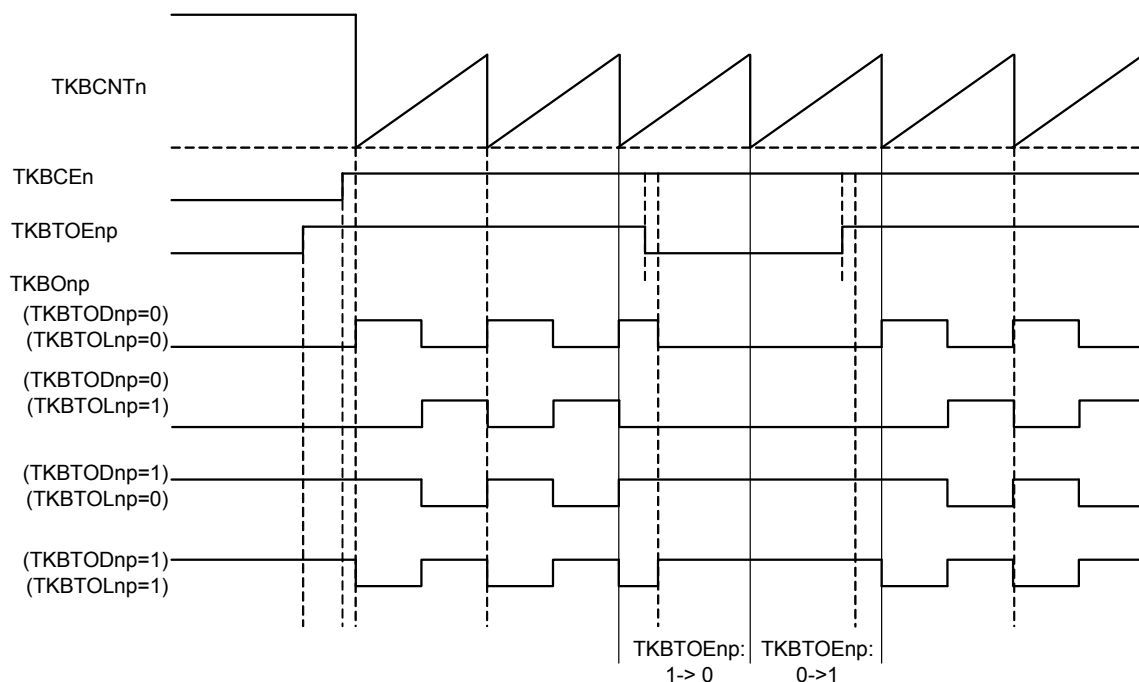


图 7-26. 默认电平和有效电平的时序图
(在计数器和比较寄存器(TKBCRn1 至 TKBCRn3)匹配后将 **TKBTOEnp** 由“0”更改为“1”时)

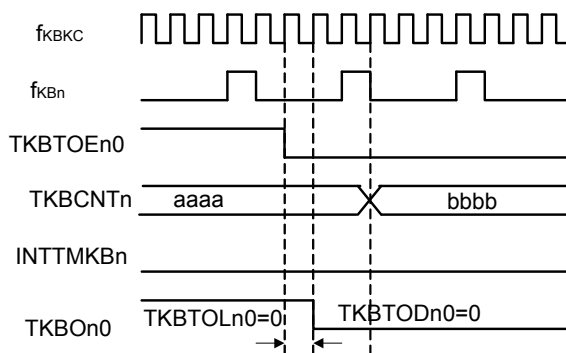


(3) **TKBTOEnp** 由“1”改变为“0”

(a) 基本时序

当 **TKBTOEnp** 由“1”改变为“0”时，在 1 个 f_{KBKC} 之后，由 **TKBTODn** 设置的默认电平为 **TKBOnp**。

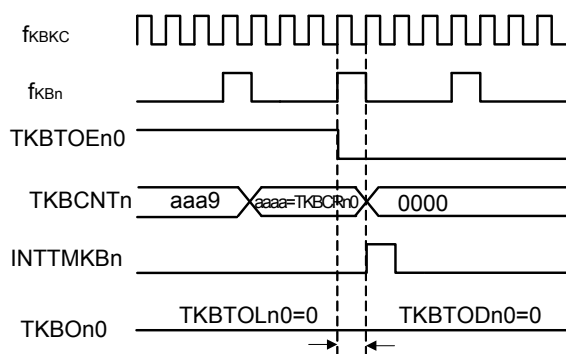
图 7-27. 默认电平和有效电平的时序图
(**TKBTOEn0** 由“1”改变为“0”)



(b) 当 **TKBCRn0** 值匹配引起的设置与 **TKBTOEnp** 的清除同时发生时:

当 **TKBTOEnp** 的设置时序(由“1”改变为“0”)与 **TKBCNTn** 和 **TKBCRnm** 的匹配时序相同时，优先 **TKBTOEnp** 的变更，成为由 **TKBTODnp** 设置的默认电平。

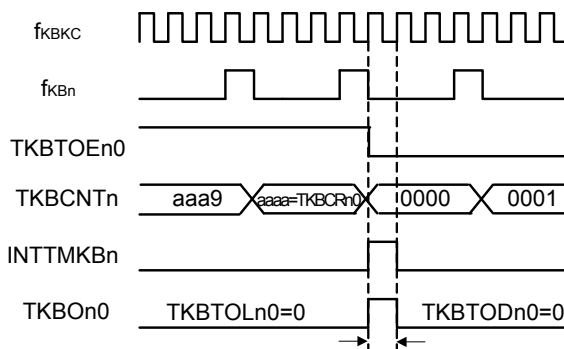
图 7-28. 默认电平和有效电平的时序图
(当 **TKBTOEnp** 的设置时序(由“1”改变为“0”)与 **TKBCNTn** 和 **TKBCRnm** 的匹配时序相同)



(c) 当 **TKBTOEnp** 操作与定时器计数器时钟生成同时发生时

当 **TKBTOEnp** 操作与定时器计数器时钟生成同时发生时， $TKBCNTn = TKBCRnm$ 匹配将使 **TKBOnp** 被设置。
1 个 f_{KBn} 后，**TKBOnp** 成为由 **TKBTODnp** 设置的默认电平。

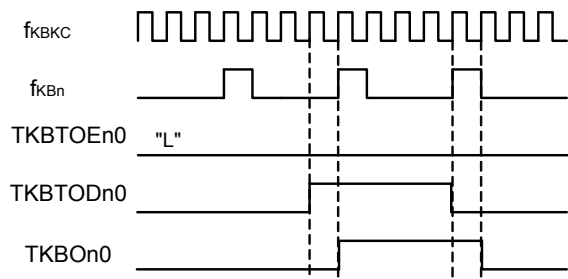
图 7-29. 默认电平和有效电平的时序图
(当 **TKBTOEn0** 操作与定时器计数器时钟生成同时发生时)



(4) 当 **TKBTOEnp = 0** 时更改 **TKBTODnp**

如果在 **TKBTOEnp = 0** 时更改 **TKBTODnp**，则在 1 个 f_{KBKC} 后，**TKBOnp** 成为由 **TKBTODnp** 设置的默认电平。

图 7-30. 默认电平和有效电平的时序图
(当 **TKBTOEnp = 0** 时更改 **TKBTODnp**)



7.4.3 停止/重启操作

可以通过控制 TKBCEn 来进行 16 位定时器 KB 的停止和开始操作。

通过将 TKBCEn 从“1”改为“0”，可以使 16 位定时器 KB 复位并停止操作。

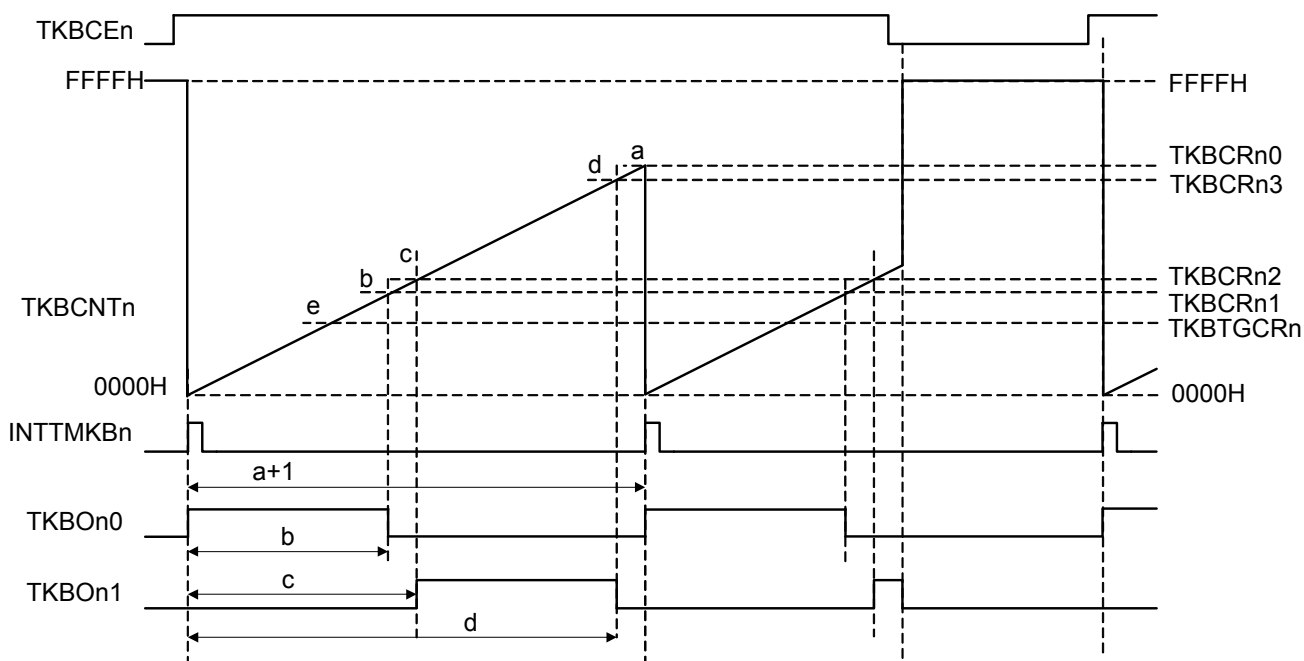
计数器 TKBCNTn 被复位为 FFFFH，并停止操作。

TKBOnp 输出将输出由 TKBTODn 设置的默认电平。

通过将 TKBCEn 从“0”改为“1”，可以使 16 位定时器 KB 开始操作。

当 TKBCEn = 0 时，计数器 TKBCNTn 保持为 FFFFH；将 TKBCEn 从“0”改为“1”时，开始计数操作。

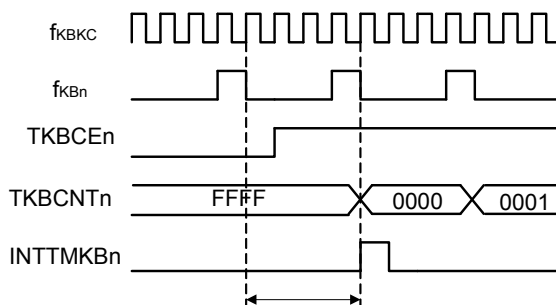
图 7-31. 停止操作的时序图 (TKBTOLnp = 0, TKBTODnp = 0)



(1) 计数操作开始时序

当 TKBCEn 从“0”更改为“1”时，在经过最小 1 个 f_{KBK} 到最大 1 个 f_{KBn} 之后开始计数操作。
 在计数操作开始时序输出 INTTMKBn。

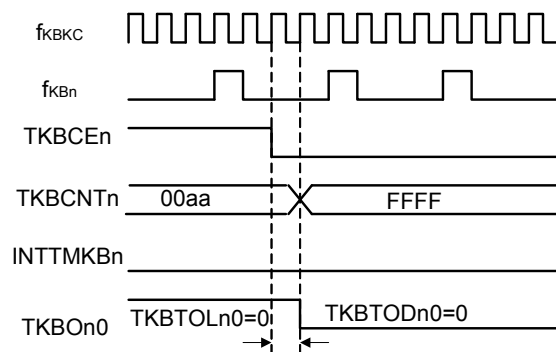
图 7-32. 开始操作的时序图 (TKBCEn 从“0”更改为“1”时)



(2) 计数操作停止时序

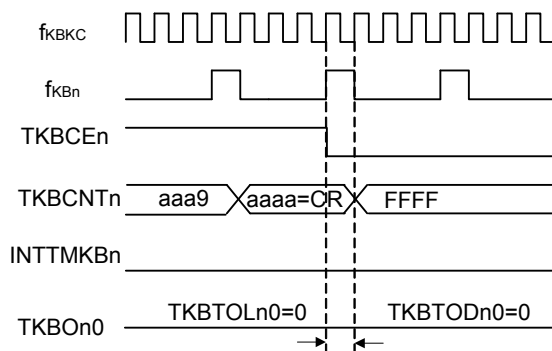
当 TKBCEn 从“1”更改为“0”时，在经过最小 1 个 f_{KBK} 之后停止计数操作。
 TKBCNTn 被复位为 FFFFH，TKBOnp 为通过 TKBTODn 设置的默认电平。

图 7-33. 停止操作的时序图 (TKBCEn 从“1”更改为“0”时)



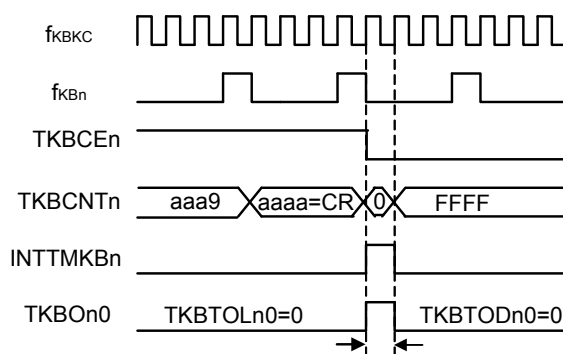
当 TKBCEn 从“1”更改为“0”时，在经过最小 1 个 f_{KBK} 之后停止计数操作。
 在发生 f_{KBn} 之前，即使发生 TKBCn = TKBCRn0 匹配也不输出 INTTMKBn。

图 7-34. 停止操作的时序图
 (发生定时器计数时钟之前操作 TKBCEn 时)



当 TKBCEn 操作与 f_{KBK} 生成同时发生时，TKBCn = TKBCRnm 匹配将使 TKBOnp 被设置且输出 INTTMKBn。
 经过 1 个 f_{KBK} 之后 TKBCn 被复位为 FFFFH，TKBOnp 为通过 TKBTODnp 设置的默认电平。

图 7-35. 停止操作的时序图
 (TKBCEn 操作与定时器计数时钟生成同时发生时)

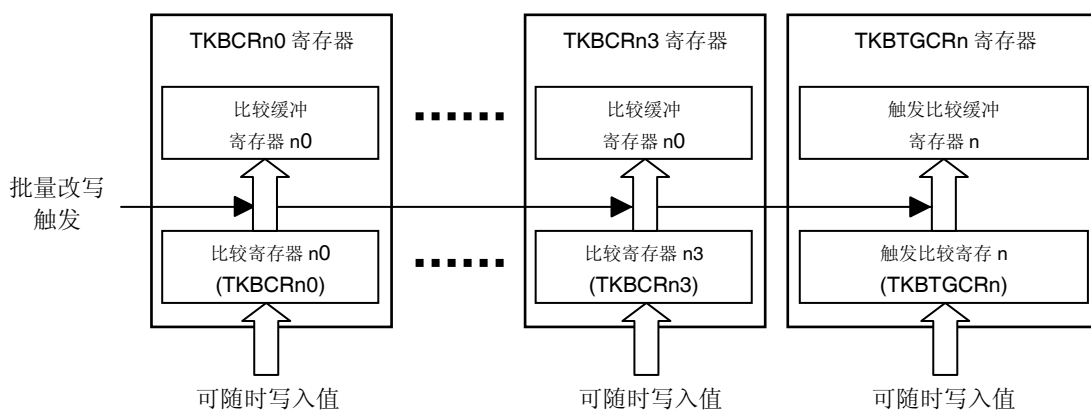


7.4.4 批量覆盖操作

如图 7-36 所示，定时器 KB 的 TMKB 比较寄存器 np (TKBCR np)为两层结构。

因此，即使程序为 TKBCR np 设置任意值，该值也不会立即生效。在任意时序为 TKBCR np 设置的值将在计数器开始运行或出现传送触发信号后一起被传送到缓冲寄存器，实际上用于比较操作。因此，可以在不同的时序为多个比较寄存器设置值。

图 7-36. 比较寄存器批量覆盖功能



备注 如上所示，虽然 TMKB 比较寄存器 np (TKBCR np)为两层结构，除了在向其写入值时，系统均会将其视为单一的寄存器。

(1) 批量覆盖的时序

在以下三种情况时，批量写入多个比较寄存器。这三种情况中，(c)可以通过配置寄存器来进行控制。

- (a) 定时器 KB 开始计数操作时
- (b) 16 位计数器的计数值与 TMKB 比较寄存器 $n0$ (TKBCR $n0$)的设置值匹配。
- (c) 由于外部触发而允许批量覆盖时，再次发生外部触发。

7.4.5 单机模式 (通过TKBCRn0 的周期控制)

(1) 功能概要

在单机模式时，由 TKBCRn0 的设置值定义周期，由 TKBCRn0 和 TKBCRn1 生成 TKBO_{n0}，由 TKBCRn2 和 TKBCRn3 生成 TKBO_{n1}。

可以在 0%至 100%的范围内设置占空比，并且可以使用以下公式计算周期和占空比。

[TKBO_{n0} 输出的计算公式]

脉冲周期 = (TKBCRn0 的设置值 + 1) × 计数器时钟周期

占空比[%] = (TKBCRn1 的设置值 / (TKBCRn0 的设置值 + 1)) × 100

0%输出: TKBCRn1 的设置值 = 0000H

100%输出: TKBCRn1 的设置值 ≥ TKBCRn0 的设置值 + 1

[TKBO_{n1} 输出的计算公式]

占空比[%] = ((TKBCRn3 的设置值 - TKBCRn2 的设置值) / (TKBCRn0 的设置值 + 1)) × 100

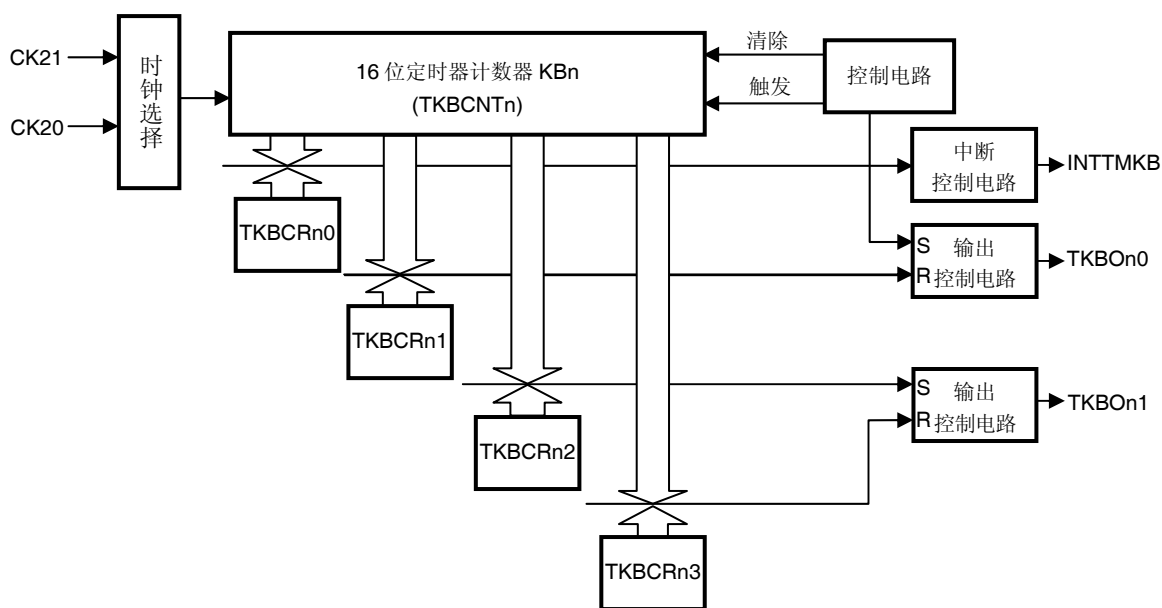
0%输出: TKBCRn3 的设置值 = TKBCRn2 的设置值

100%输出: TKBCRn2 的设置值 = 0000H, TKBCRn3 的设置值 ≥ TKBCRn0 的设置值 + 1

注意事项 务必保障 TKBCRn2 的设置值 ≤ TKBCRn3 的设置值。

单机模式下的配置如图 7-37 所示。

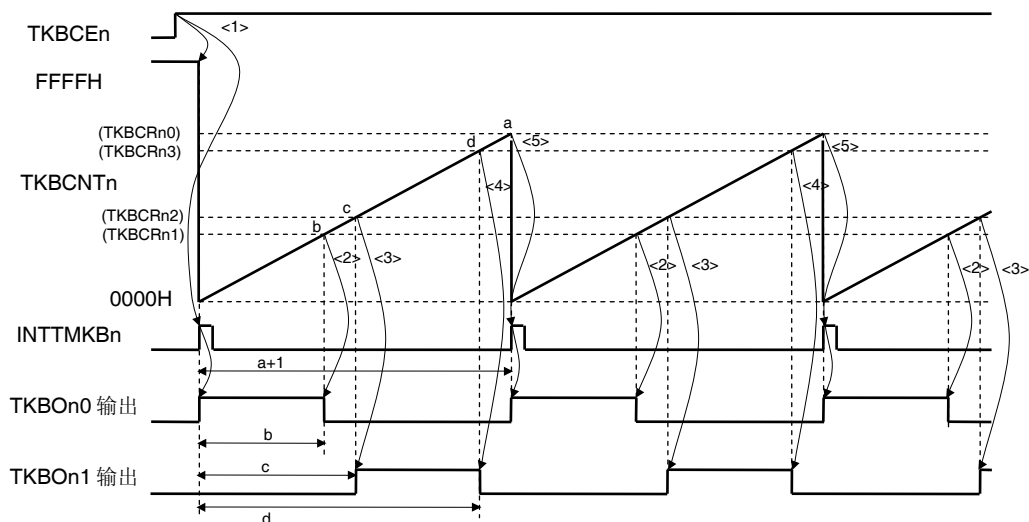
图 7-37. 单机模式的配置 (通过 TKBCRn0 的周期控制)



(2) 操作概要

单机模式下的操作时序例如图 7-38 所示。

图 7-38. 单机模式的操作时序例 (通过 TKBCRn0 的周期控制)
(输出的默认值为低电平(TKBTODnp = 0), 有效电平为高电平(TKBTOLnp = 0))



本节介绍关于单机操作（TKBCR0 进行周期控制）的示例。下列描述与图 7-38 的<1>至<5>相关联。

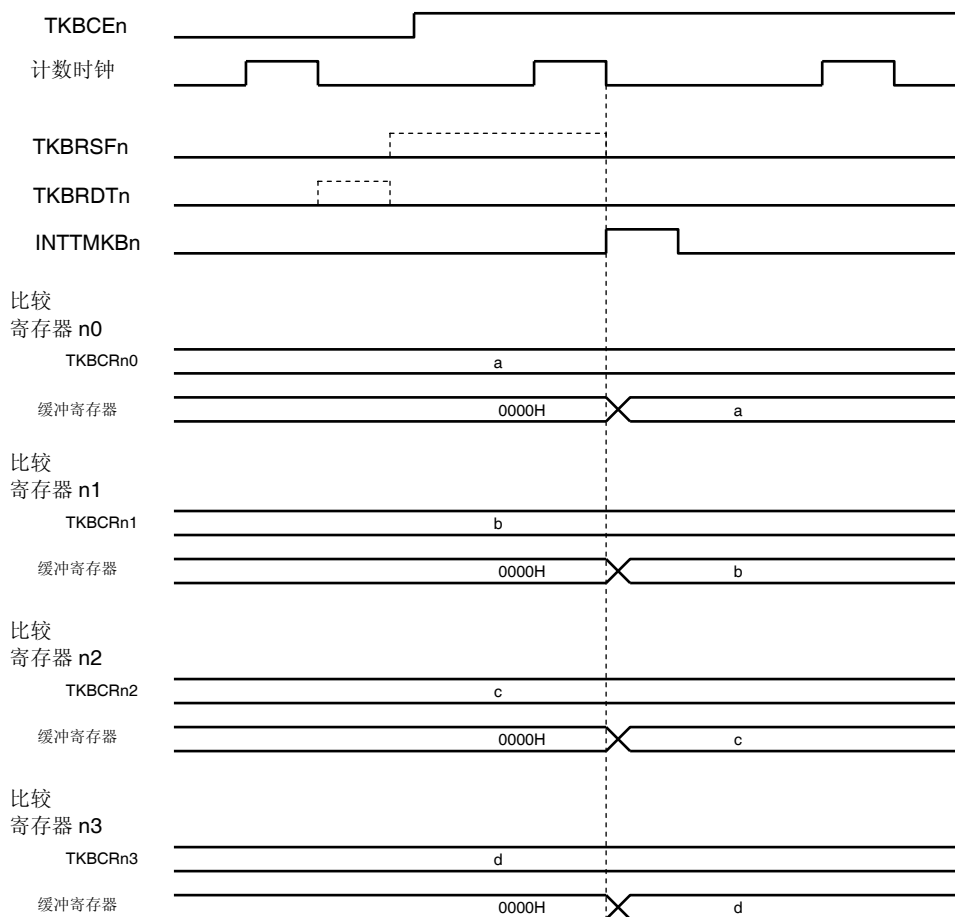
- <1> 当 TKBCEn 设置为值 1 时，16 位定时器计数器 KBn (TKBCNTn) 与计数时钟同步且从 FFFFH 改变为 0000H，然后开始累加计数。与此同时，输出 INTTMKBn, TKBOn0 输出则从默认值（由 TKB0IOC0 寄存器的 TKB0TOD0 位指定）改变为有效值（在此例中为高电平，由 TKB0TOL0 位指定）（TKBOn1 输出保留由 TKB0TOD1 位指定的默认值）。
- <2> 当 TKBCNTn 累加计数且其值与 TMKB 比较寄存器 n1 (TKBCRn1) 指定的值匹配时，TKBOn0 输出将成为无效电平。
- <3> 当 TKBCNTn 累加计数且其值与 TMKB 比较寄存器 n2 (TKBCRn2) 指定的值匹配时，TKBOn1 输出将成为有效电平。
- <4> 当 TKBCNTn 累加计数且其值与 TMKB 比较寄存器 n3 (TKBCRn3) 指定的值匹配时，TKBOn1 输出将成为无效电平。
- <5> 当 TKBCNTn 累加计数且其值与 TMKB 比较寄存器 n0 (TKBCRn0) 指定的值匹配时，在下一个计数时钟处输出 INTTMKBn, 而且 TKBOn0 输出将成为有效电平。TKBCNTn 从 0000H 开始累加计数。
- <6> 重复<2>至<5>。

(3) 批量覆盖操作（开始计数操作时）

定时器 KB0、KB1 和 KB2 的比较寄存器具有在计数器操作（用“1”覆盖 TKBCEn 位之后生成计数时钟而引起）开始的同时批量更新内部缓冲寄存器功能。

仅在计数操作开始时序处不需要用“1”覆盖 TKBRDTn 位即可发生批量覆盖（参阅图 7-39）。

图 7-39. 批量覆盖功能：计数操作开始时缓冲器更新的时序图



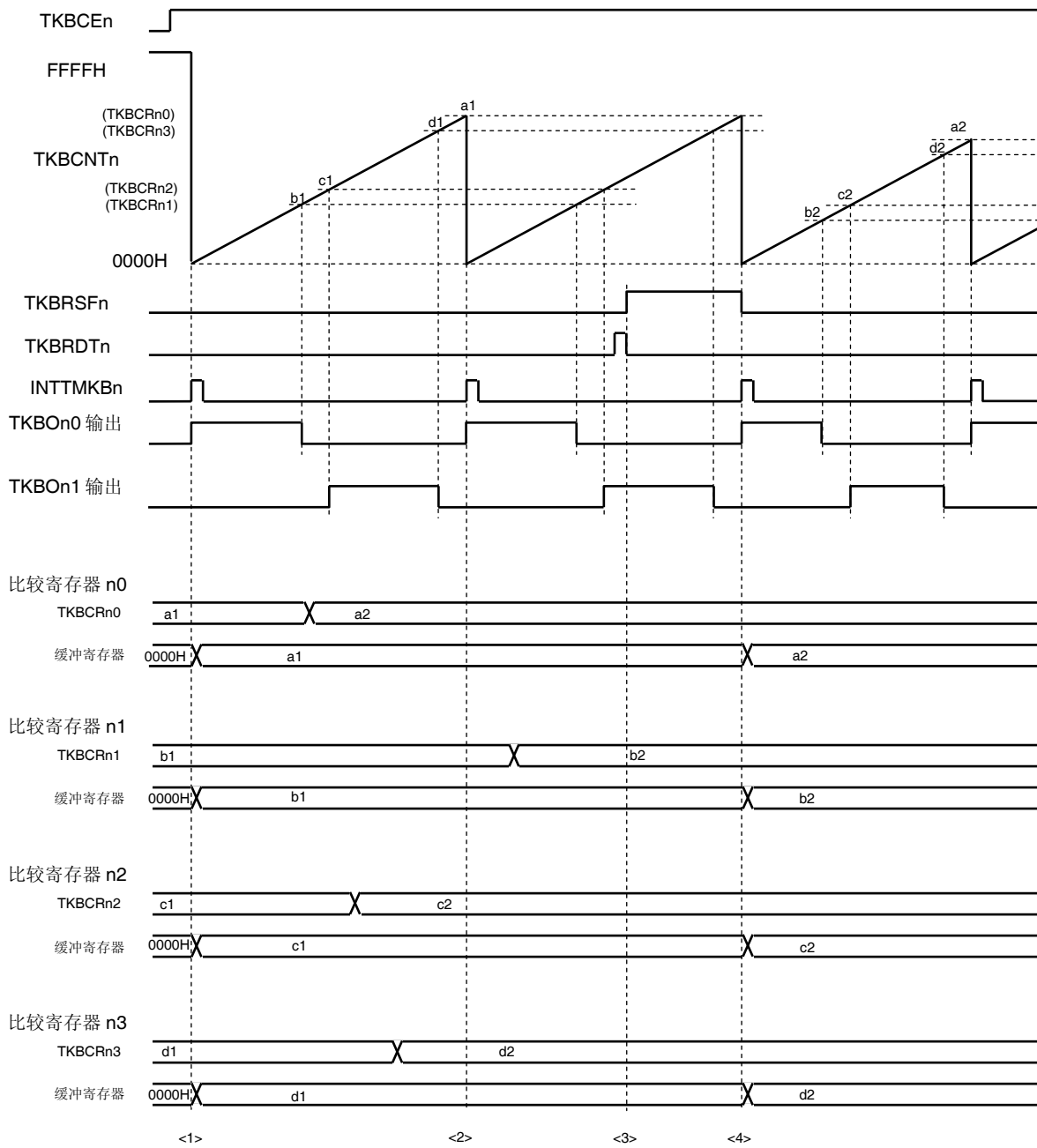
备注 当 TKBCEn = 0 时，若向 TKBRDTn 写入“1”，则 TKBRSF_n 被设置为“1”。
在计数操作开始时序（产生了计数器开始触发）时，TKBRSF_n 被清除为“0”。

(4) 批量覆盖功能：计数操作期间更新缓冲器

定时器 KB0、KB1 和 KB2 的比较寄存器把用“1”覆盖 TKBRDTn 位的操作识别为批量覆盖触发，在下次发生计数器清除（TKBCNTn 和 TKBCRn0 批配）的同时批量更新内部缓冲器寄存器。TKBRSF_n 作为标志，用于显示将“1”写入 TKBRDTn 位直到完成批量覆盖（参阅图 7-40）。

- <1> 在 TKBCE_n 由“0”改变为“1”且 TKBCNT_n 开始计数操作时，比较寄存器的设置被传送至缓冲器寄存器。
- <2> 如果未执行向 TKBRDT_n 写入“1”的操作，即使在 TKBCR_{n0-3} 和 TKBTGCR_n 寄存器被覆盖后执行了清除计数器，也不会产生覆盖。
- <3> 通过向 TKBRDT_n 写入“1”，可以将批量覆盖保留标志 (TKBRSF_n) 设置为“1”。
- <4> 在 TKBRSF_n = 1 且清除计数器时，比较寄存器的设置被传送至缓冲器寄存器。同时，TKBRSF_n 成为“0”。

图 7-40. 批量覆盖功能：计数操作期间缓冲器更新的时序图



(5) 单机模式 (通过 TKBCRn0 的周期控制)的寄存器设置内容示例

位编号	15	14	13	12	11	10	9	8
TKBCTLn0	-	TKBGTEn1	TKBSSEn1	TKBDIEn1	-	TKBGTEn0	TKBSSEn0	TKBDIEn0
设置	0	1/0	1/0	1/0	0	1/0	1/0	1/0

位编号	7	6	5	4	3	2	1	0
TKBCTLn0	TKBMFEn	-	TKBIRSn1	TKBIRSn0	-	TKBTSEn	TKBSTSn1	TKBSTSn0
设置	0	0	0	0	0	0	0	0

位编号	7	6	5	4	3	2	1	0
TKBCTLn1	TKBCEn	-	-	TKBCKSn	TKBSCMn	-	TKBMDn1	TKBMDn0
设置	1	0	0	1/0	0	0	0	0

位编号	7	6	5	4	3	2	1	0
TKBIOCn0	-	-	-	-	TKBTOLn1	TKBTOLn0	TKBTODn1	TKBTODn0
设置	0	0	0	0	1/0	1/0	1/0	1/0

位编号	7	6	5	4	3	2	1	0
TKBIOCn1	-	-	-	-	-	-	TKBTOEn1	TKBTOEn0
设置	0	0	0	0	0	0	1/0	1/0

TKBCRn0	0000H至FFFFH
TKBCRn1	0000H至FFFFH
TKBCRn2	0000H至FFFFH
TKBCRn3	0000H至FFFFH
TKBTGCRn	0000H至FFFFH
TKBSIRn0	0000H至FFFFH
TKBSIRn1	0000H至FFFFH
TKBSSRn0	00H至0FH
TKBSSRn1	00H至0FH
TKBDNRn0	00H至F0H
TKBDNRn1	00H至F0H
TKBMFRn	0000H

□：在此模式下设置被固定 ■：不需要设置（默认设置）

7.4.6 单机模式 (通过外部触发输入的周期控制)

(1) 功能概要

在单机模式时，不仅可通过 TKBCRn0 控制周期，而且可以利用外部触发输入来控制周期。

根据 16 位定时器 KB 操作控制寄存器的 TKBSTSn1 和 TKBSTSn0 位所选择的输入信号用于检测外部触发输入（定时器重启功能）。利用此功能，可以实现例如临界传导模式的 PFC 控制。

检测到外部触发输入后，计数器 TKBCNTn 被清除为 0000H，而 TKBOn0/TKBOn1 输出将分别被设置为有效电平和无效电平。如果在检测到外部触发输入之前发生了 TKBCRn0 的设置值和计数器(TKBCNTn)匹配，则计数器被清除为 0000H，并且继续执行操作。

关于未检测到外部触发输入且通过 TKBCRn0 控制周期的情况下 TKBOn0/TKBOn1 输出的计算公式，请参阅 7.4.5 单机模式 (通过 TKBCRn0 的周期控制)。

在通过外部触发输入检测来控制周期的情况下，TKBOn0/TKBOn1 输出的计算公式如下所示：

[TKBOn0 输出的计算公式]

脉冲周期 = (外部触发输入检测时的计数器值 + 1) × 计数时钟周期

占空比[%] = (TKBCRn1 的设置值 / (外部触发输入检测时的计数器值 + 1)) × 100

0%输出：TKBCRn1 的设置值 = 0000H

100%输出：TKBCRn1 的设置值 ≥ 外部触发输入检测时的计数器值 + 1

[TKBOn1 输出的计算公式]

脉冲周期 = (外部触发输入检测时的计数器值 + 1) × 计数时钟周期

占空比[%] = ((TKBCRn3 的设置值 - TKBCRn2 的设置值) / (外部触发输入检测时的计数器值 + 1)) × 100

0%输出：TKBCRn3 的设置值 = TKBCRn2 的设置值

100%输出：TKBCRn2 的设置值 = 0000H，TKBCRn3 的设置值 ≥ 外部触发输入检测时的计数器值 + 1

注意事项 务必保障 TKBCRn2 的设置值 ≤ TKBCRn3 的设置值。

单机模式(通过外部触发输入的周期控制)的配置如图 7-41 所示。

图 7-41. 单机模式(通过外部触发输入的周期控制)的配置

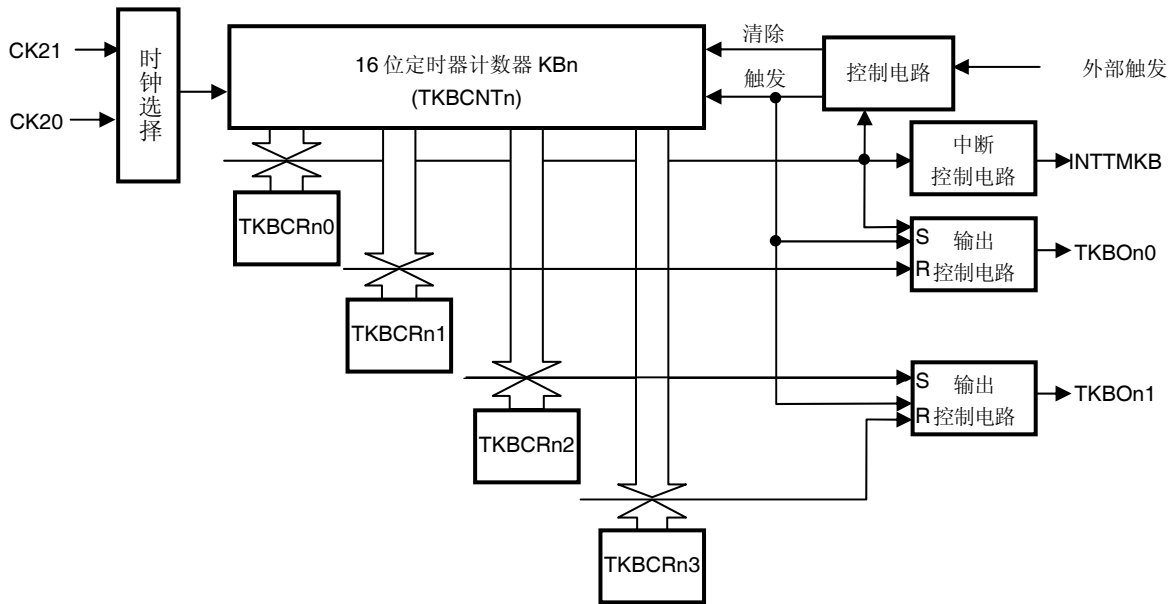


表 7-3. 单机操作 (通过外部触发输入的周期控制)的外部触发分配列表

	定时器KB0	定时器KB1	定时器KB2
比较器0	-	-	-
比较器1	-	√	-
比较器2	-	-	√
比较器3	√	-	-
比较器4	-	-	-
比较器5	-	-	-
INTP20	√	√	√
INTP21	√	√	√

(2) 批量覆盖功能(通过外部触发输入的周期控制, 计数操作中的缓冲器更新 (将TKBTSEn位设置为1))

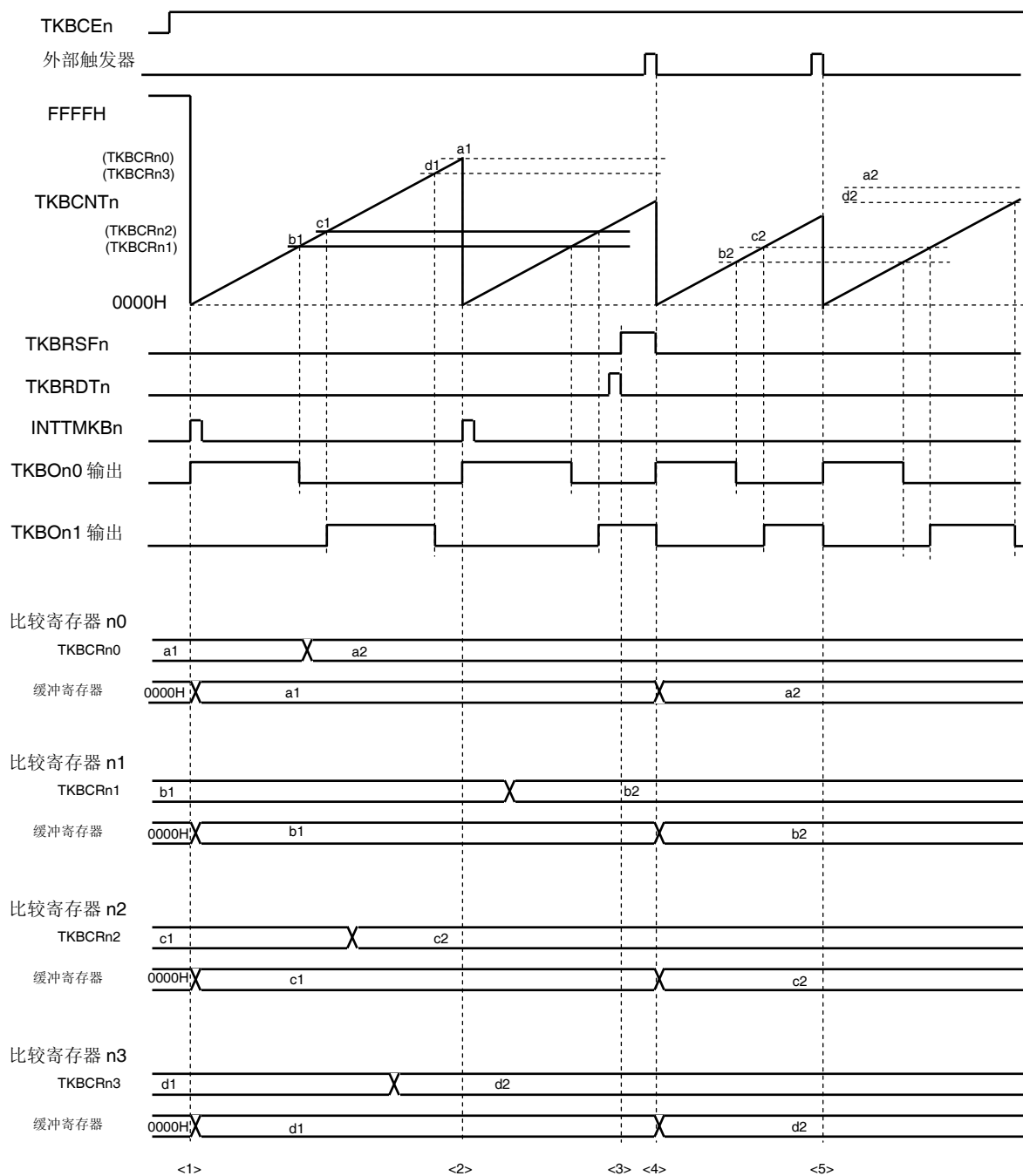
在通过外部触发输入来控制周期的单机模式下, 通过将TKBCTLn0寄存器的TKBTSEn位设置为“1”, 在向TKBRDTn位写入“1”后, 检测到外部触发输入时, 执行计数器清除和比较寄存器批量覆盖操作。

同计数器清除一样, 向TKBRDTn位写入“1”后检测到外部触发输入之前, 在TKBCRn0和计数器(TKBCNTn)匹配时执行批量覆盖。

通过TKBCTLn0寄存器的TKBSTSn1和TKBSTSn0位选择外部触发输入因素。将TKBTSEn位设置为“1”时批量覆盖操作的时序示例如图7-42所示。

- <1> 在TKBCEn由“0”改变为“1”且TKBCNTn开始计数操作时, 比较寄存器的设置被传送至缓冲器寄存器。
- <2> 如果未执行向TKBRDTn写入“1”的操作, 即使在TKBCRn0-3和TKBTGCRn寄存器被覆盖后执行了清除计数器, 也不会产生覆盖。
- <3> 通过向TKBRDTn写入“1”, 可以将批量覆盖保留标志(TKBRSFn)设置为“1”。
- <4> 如果在TKBTSEn位被设置为“1”且TKBRSFn位也为“1”的情况下, 由于外部触发输入而产生了计数器清除时, 比较寄存器的设置值将被传送至缓冲器寄存器。同时, TKBRSFn位成为“0”。
- <5> 如果未向TKBRDTn位写入“1”, 即使由于外部触发输入而产生了计数器清除也不会执行批量覆盖。

图 7-42. 批量覆盖功能：通过外部触发输入的周期控制，计数操作中的缓冲器更新 (将 $TKBTSEn$ 位设置为 1)



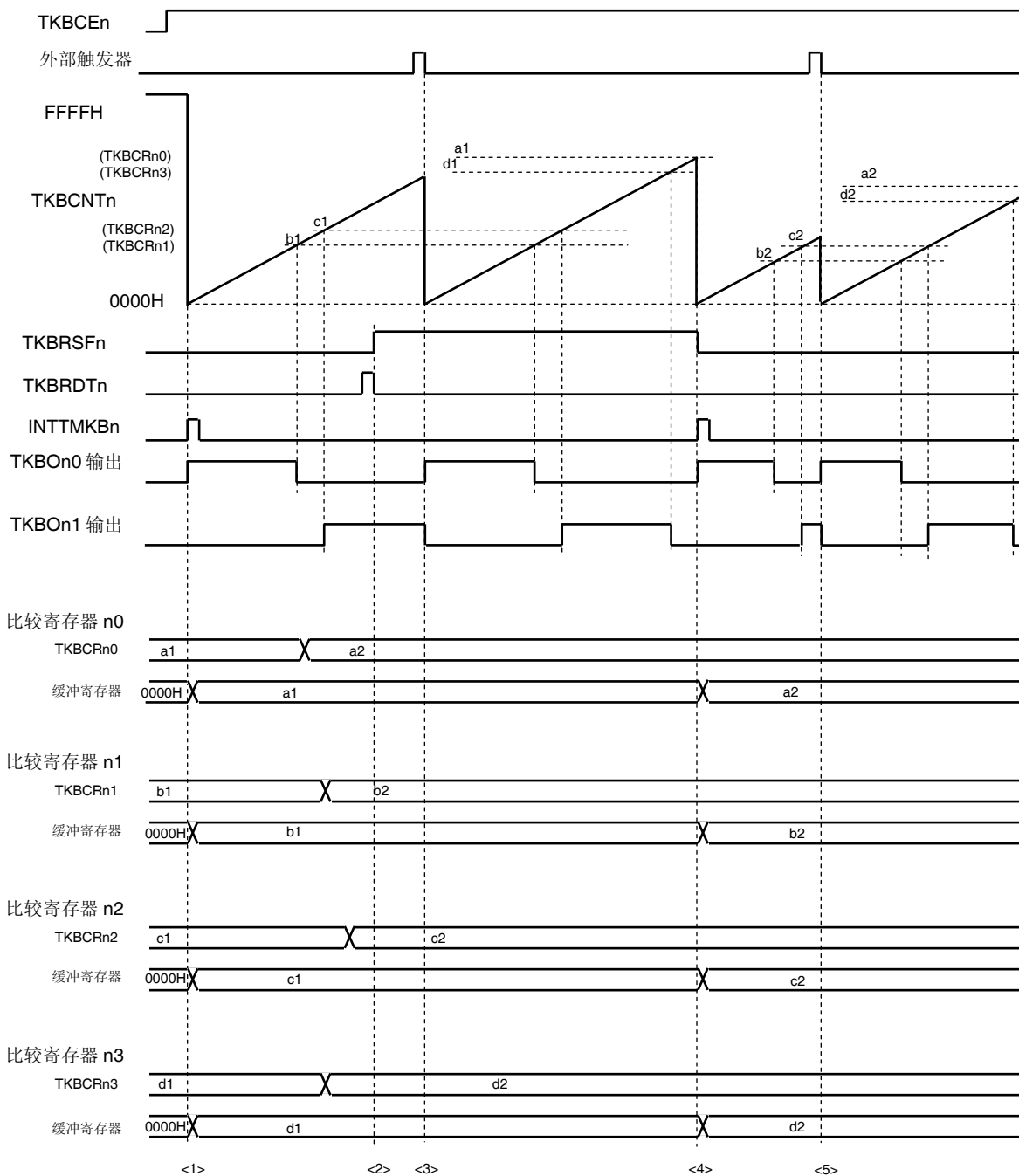
(3) 批量覆盖功能(通过外部触发输入的周期控制, 计数操作中的缓冲器更新 (将TKBTSEn位清除为0))

这是一个通过外部触发输入控制周期的单机模式时, 将TKBCTLn0寄存器的TKBTSEn位设置为“0”的情况下的示例。在这种情况下, 当值“1”被写入TKBRDTn位而且在覆盖保留标志(TKBRSFn位)为“1”的状态下检测到外部触发输入时, 执行计数器清除, 但不执行比较寄存器的批量覆盖。

通过TKBCTLn0寄存器的TKBSTSn1和TKBSTSn0位选择外部触发输入因素。将TKBTSEn位设置为“0”时批量覆盖操作的时序示例如图7-43所示。

- <1> 在TKBCEn由“0”改变为“1”且TKBCNTn开始计数操作时, 比较寄存器的设置被传送至缓冲器寄存器。
- <2> 在重写TKBCRn0至TKBCRn3和TKBTGCRn寄存器之后, 通过向TKBRDTn写入“1”, 可以将批量覆盖保留标志(TKBRSFn)设置为“1”。
- <3> 如果TKBTSEn位不为“1”, 即使由于外部触发输入而产生了计数器清除也不会执行批量覆盖。
- <4> 如果在TKBRSFn位为“1”的状态下发生计数器清除(TKBCNTn与TKBCRn0匹配)时, 比较寄存器的设置值将被传送至缓冲寄存器。同时, TKBRSFn位成为“0”。
- <5> 除非TKBTSEn和TKBRSFn位均为“1”, 否则即使由于外部触发输入而产生了计数器清除也不执行批量覆盖。

图 7-43. 批量覆盖功能：通过外部触发输入的周期控制，计数操作中的缓冲器更新 (将 TKBTSEn 位清除为 0)



(4) 单机模式 (通过外部触发输入的周期控制)的寄存器设置内容示例

	15	14	13	12	11	10	9	8
TKBCTLn0	- 0	TKBGTEn1 0	TKBSSEn1 0	TKBDIEn1 0	- 0	TKBGTEn0 0	TKBSSEn0 0	TKBDIEn0 0
	7	6	5	4	3	2	1	0
	TKBMFEn 1/0	- 0	TKBIRSn1 0	TKBIRSn0 0	- 0	TKBTSEn 1/0	TKBSTSn1 1/0	TKBSTSn0 1/0
	7	6	5	4	3	2	1	0
TKBCTLn1	TKBCEn 1/0	- 0	- 0	TKBCKSn 1/0	TKBSCMn 0	- 0	TKBMDn1 0	TKBMDn0 0
	7	6	5	4	3	2	1	0
TKBIOCn0	- 0	- 0	- 0	- 0	TKBTOLn1 1/0	TKBTOLn0 1/0	TKBTODn1 1/0	TKBTODn0 1/0
	7	6	5	4	3	2	1	0
TKBIOCn1	- 0	- 0	- 0	- 0	- 0	- 0	TKBTOEn1 1/0	TKBTOEn0 1/0
TKBCRn0	0000H至FFFFH							
TKBCRn1	0000H至FFFFH							
TKBCRn2	0000H至FFFFH							
TKBCRn3	0000H至FFFFH							
TKBTGCRn	0000H至FFFFH							
TKBSIRn0	0000H							
TKBSIRn1	0000H							
TKBSSRn0	00H							
TKBSSRn1	00H							
TKBDNRn0	00H							
TKBDNRn1	00H							
TKBMFRn	0000H至FFFFH							

：在此模式下设置被固定 ：不需要设置（默认设置）

7.4.7 同时开始/停止模式

(1) 功能概要

使用多个 KBn 定时器配置主/从属结构，从而通过与主定时器 KB 的计数开始/停止操作同步，可以同时开始/停止从属定时器 KBm。

在这种情况下，为主选择“单机模式(TKBMD01, TKBMD00 = 0, 0)”，为从属选择“同时开始/停止模式(TKBMDm1, TKBMDm0 = 0, 1)”。

在同时开始/停止模式时，仅有主和从属的开始/停止时序为同步。

如果主和从属选择的计数时钟(CK0/CK1)不同，可通过将主的 TKBSCM0 位设置为“1”来使主和从属的计数操作开始时序保持一致。

计数操作开始时序之后，每个定时器独立运行。

仅对主的 TKBSCM0 位进行设置。

- 注意事项**
1. 仅限定时器 KB 可作为主定时器。
 2. 主的选择时钟必须比从属的选择时钟快，或与其相当。

主和从属的选择时钟的关系	CK0和CK1的关系	主的TKBSCM0位	可对应
主和从属选择相同时钟	-	0	○
主和从属选择不同时钟	当主的选择时钟比从属的选择时钟快时	1	○
主和从属选择不同时钟	当主的选择时钟比从属的选择时钟慢时	-	×

有关同时开始/停止模式的 TKBOn0/TKBOn1 输出的计算公式，请参阅 7.4.5 单机模式 (通过 TKBCRn0 的周期控制) 和 7.4.6 单机模式 (通过外部触发输入的周期控制)。

(2) 可在同时开始/停止模式下使用的主和从属的操作模式的组合

可在同时开始/停止模式下使用的主和从属的操作模式如下所示。

主:

操作模式	TKBMDn1, TKBMDn0	TKBSTSn1, TKBSTSn0	可设置
单机模式 (通过TKBCRn0的周期控制)	00B	00B	○
单机模式 (通过外部触发输入的周期控制)	00B	01B/10B/11B	○
同时开始/停止模式 (通过TKBCRn0的周期控制)	01B	00B	×
同时开始/停止模式 (通过外部触发输入的周期控制)	01B	01B/10B/11B	×
同步开始/清除模式 (通过主的周期控制)	10B	—	×
交错式PFC输出模式	11B	—	×

从属:

操作模式	TKBMDn1, TKBMDn0	TKBSTSn1, TKBSTSn0	可设置
单机模式 (通过TKBCRn0的周期控制)	00B	00B	×
单机模式 (通过外部触发输入的周期控制)	00B	01B/10B/11B	×
同时开始/停止模式 (通过TKBCRn0的周期控制)	01B	00B	○
同时开始/停止模式 (通过外部触发输入的周期控制)	01B	01B/10B/11B	○
同步开始/清除模式 (通过主的周期控制)	10B	—	×
交错式PFC输出模式	11B	—	×

(3) 同时开始/停止模式

主：单机模式 (通过 TKBCRn0 的周期控制)的寄存器设置内容示例

	15	14	13	12	11	10	9	8
TKBCTLn0	- 0	TKBGTEn1 1/0	TKBSSEn1 1/0	TKBDIEn1 1/0	- 0	TKBGTEn0 1/0	TKBSSEn0 1/0	TKBDIEn0 1/0
	7	6	5	4	3	2	1	0
	TKBMFEEn 0	- 0	TKBIRSn1 0	TKBIRSn0 0	- 0	TKBTSEn 0	TKBSTSn1 0	TKBSTSn0 0
	7	6	5	4	3	2	1	0
TKBCTLn1	TKBCEEn 1	- 0	- 0	TKBCKSn 1/0	TKBSCMn 1/0	- 0	TKBMDn1 0	TKBMDn0 0
	7	6	5	4	3	2	1	0
TKBIOCn0	- 0	- 0	- 0	- 0	TKBTOLn1 1/0	TKBTOLn0 1/0	TKBTODn1 1/0	TKBTODn0 1/0
	7	6	5	4	3	2	1	0
TKBIOCn1	- 0	- 0	- 0	- 0	- 0	- 0	TKBTOEn1 1/0	TKBTOEn0 1/0
TKBCRn0	0000H至FFFFH							
TKBCRn1	0000H至FFFFH							
TKBCRn2	0000H至FFFFH							
TKBCRn3	0000H至FFFFH							
TKBTGCRn	0000H至FFFFH							
TKBSIRn0	0000H至FFFFH							
TKBSIRn1	0000H至FFFFH							
TKBSSRn0	00H至0FH							
TKBSSRn1	00H至0FH							
TKBDNRn0	00H至F0H							
TKBDNRn1	00H至F0H							
TKBMFRn	0000H							

□：在此模式下设置被固定 ■：不需要设置（默认设置）

(4) 同时开始/停止模式

从属：单机模式 (通过 TKBCRn0 的周期控制)的寄存器设置内容示例

	15	14	13	12	11	10	9	8
TKBCTLn0	- 0	TKBGTEn1 1/0	TKBSSEn1 1/0	TKBDIEn1 1/0	- 0	TKBGTEn0 1/0	TKBSSEn0 1/0	TKBDIEn0 1/0
	7	6	5	4	3	2	1	0
	TKBMFEEn 0	- 0	TKBIRSn1 0	TKBIRSn0 0	- 0	TKBTSEn 0	TKBSTSn1 0	TKBSTSn0 0
	7	6	5	4	3	2	1	0
TKBCTLn1	TKBCEEn 1	- 0	- 0	TKBCKSn 1/0	TKBSCMn 0	- 0	TKBMDn1 0	TKBMDn0 1
	7	6	5	4	3	2	1	0
TKBIOcn0	- 0	- 0	- 0	- 0	TKBTOLn1 1/0	TKBTOLn0 1/0	TKBTODn1 1/0	TKBTODn0 1/0
	7	6	5	4	3	2	1	0
TKBIOcn1	- 0	- 0	- 0	- 0	- 0	- 0	TKBTOEn1 1/0	TKBTOEn0 1/0
TKBCRn0	0000H至FFFFH							
TKBCRn1	0000H至FFFFH							
TKBCRn2	0000H至FFFFH							
TKBCRn3	0000H至FFFFH							
TKBTGCRn	0000H至FFFFH							
TKBSIRn0	0000H至FFFFH							
TKBSIRn1	0000H至FFFFH							
TKBSSRn0	00H至0FH							
TKBSSRn1	00H至0FH							
TKBDNRn0	00H至F0H							
TKBDNRn1	00H至F0H							
TKBMFRn	0000H							

□：在此模式下设置被固定 ■：不需要设置（默认设置）

(5) 同时开始/停止模式

主：单机模式 (通过外部触发输入的周期控制)的寄存器设置内容示例

	15	14	13	12	11	10	9	8
TKBCTLn0	- 0	TKBGTEn1 0	TKBSSEn1 0	TKBDIEn1 0	- 0	TKBGTEn0 0	TKBSSEn0 0	TKBDIEn0 0
	7	6	5	4	3	2	1	0
	TKBMFEn 1/0	- 0	TKBIRSn1 0	TKBIRSn0 0	- 0	TKBTSEn 1/0	TKBSTSn1 1/0	TKBSTSn0 1/0
	7	6	5	4	3	2	1	0
TKBCTLn1	TKBCEn 1	- 0	- 0	TKBCKSn 1/0	TKBSCMn 1/0	- 0	TKBMDn1 0	TKBMDn0 0
	7	6	5	4	3	2	1	0
TKBIOCn0	- 0	- 0	- 0	- 0	TKBTOLn1 1/0	TKBTOLn0 1/0	TKBTODn1 1/0	TKBTODn0 1/0
	7	6	5	4	3	2	1	0
TKBIOCn1	- 0	- 0	- 0	- 0	- 0	- 0	TKBTOEn1 1/0	TKBTOEn0 1/0
TKBCRn0	0000H至FFFFH							
TKBCRn1	0000H至FFFFH							
TKBCRn2	0000H至FFFFH							
TKBCRn3	0000H至FFFFH							
TKBTGCRn	0000H至FFFFH							
TKBSIRn0	0000H							
TKBSIRn1	0000H							
TKBSSRn0	00H							
TKBSSRn1	00H							
TKBDNRn0	00H							
TKBDNRn1	00H							
TKBMFRn	0000H至FFFFH							

☐：在此模式下设置被固定 □：不需要设置（默认设置）

(6) 同时开始/停止模式

从属：单机模式 (通过外部触发输入的周期控制)的寄存器设置内容示例

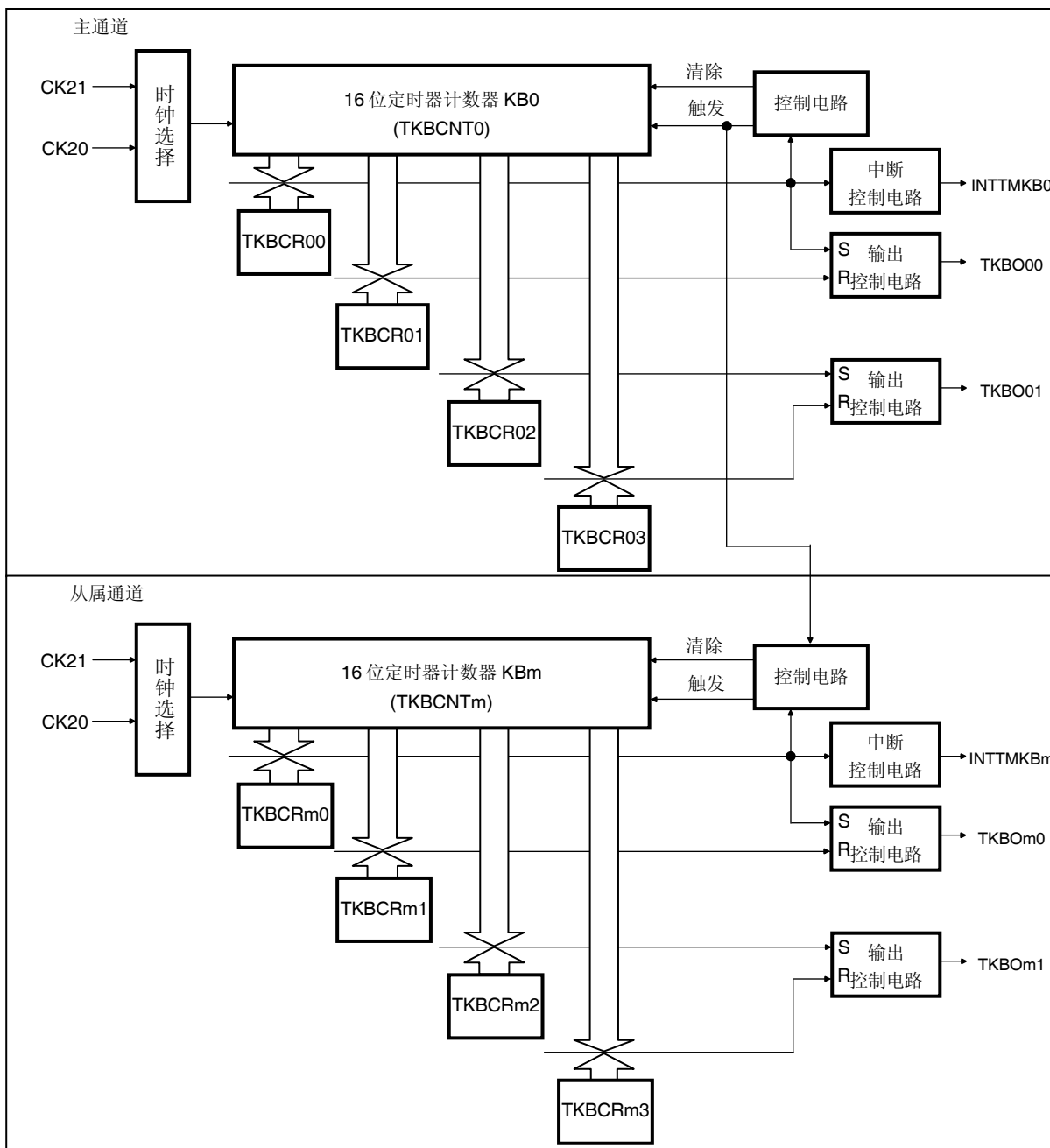
	15	14	13	12	11	10	9	8
TKBCTLn0	- 0	TKBGTEn1 0	TKBSSEn1 0	TKBDIE n1 0	- 0	TKBGTE n0 0	TKBSSE n0 0	TKBDIE n0 0
	7	6	5	4	3	2	1	0
	TKBMFEn 1/0	- 0	TKBIRSn1 0	TKBIRSn0 0	- 0	TKBTSEn 1/0	TKBSTSn1 1/0	TKBSTSn0 1/0
TKBCTLn1	7	6	5	4	3	2	1	0
	TKBCE n 1	- 0	- 0	TKBCKSn 1/0	TKBSCMn 0	- 0	TKBMDn1 0	TKBMDn0 1
TKBIOCn0	7	6	5	4	3	2	1	0
	- 0	- 0	- 0	- 0	TKBTOLn1 1/0	TKBTOLn0 1/0	TKBTODn1 1/0	TKBTODn0 1/0
TKBIOCn1	7	6	5	4	3	2	1	0
	- 0	- 0	- 0	- 0	- 0	- 0	TKBTOEn1 1/0	TKBTOEn0 1/0
TKBCRn0	0000H至FFFFH							
TKBCRn1	0000H至FFFFH							
TKBCRn2	0000H至FFFFH							
TKBCRn3	0000H至FFFFH							
TKBTGCRn	0000H至FFFFH							
TKBSIRn0	0000H							
TKBSIRn1	0000H							
TKBSSRn0	00H							
TKBSSRn1	00H							
TKBDNRn0	00H							
TKBDNRn1	00H							
TKBMFRn	0000H至FFFFH							

：在此模式下设置被固定 ：不需要设置（默认设置）

(7) 同时开始/停止模式 (通过 TKBCRn0 的周期控制)的配置

同时开始/停止模式下的配置如图 7-44 所示。

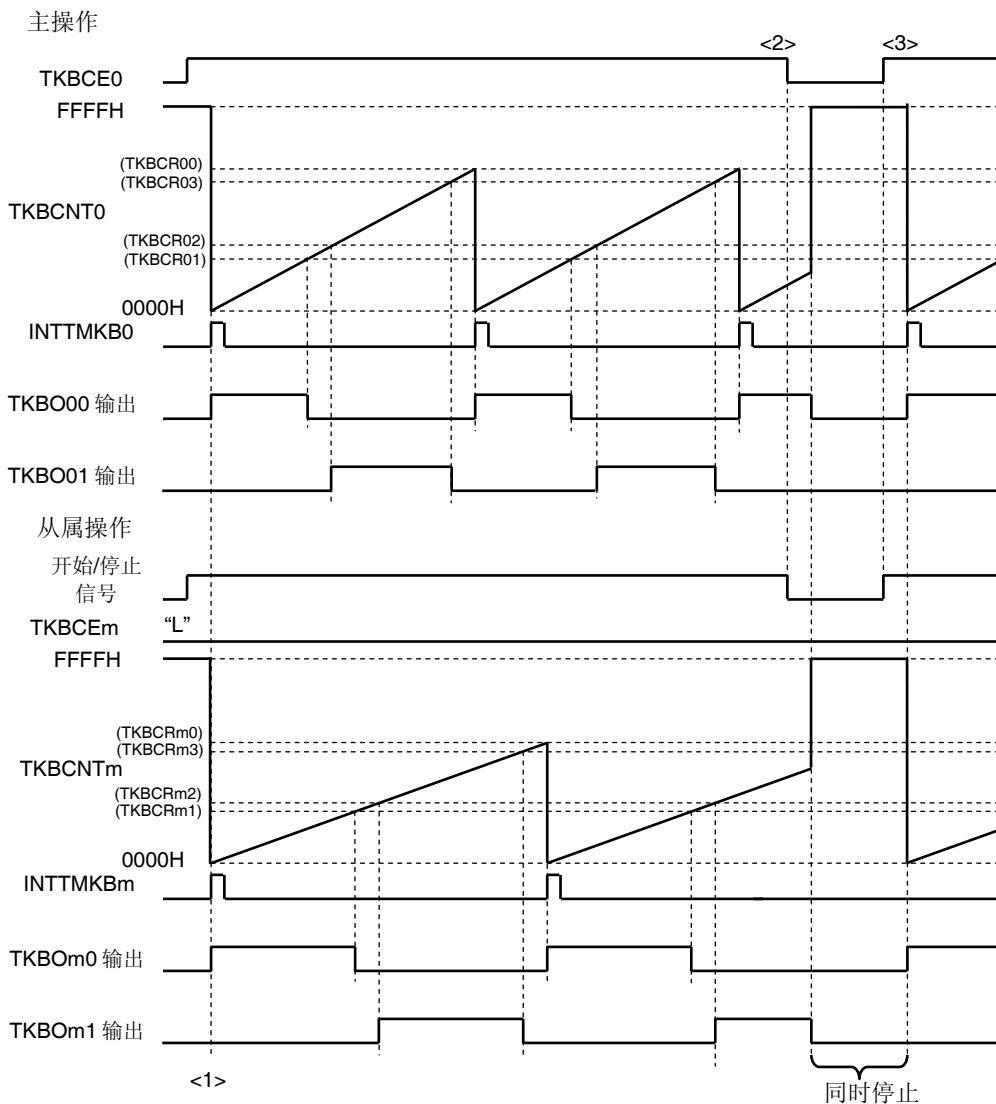
图 7-44. 同时开始/停止模式 (通过 TKBCRn0 的周期控制)的配置



(8) 操作概要

同时开始/停止模式下的操作时序例如图 7-45 所示。

图 7-45. 同时开始/停止模式的操作时序例 (通过 TKBCRn0 的周期控制)
 (输出的默认值为低电平(TKBTODnp = 0), 有效电平为高电平(TKBTOLnp = 0))



同时开始/停止模式的操作示例如下所示。下列描述与图 7-45 的<1>至<3>相关联。

- <1> 将作为主的 $TKBCE0$ 设置为 1 时，主 16 位定时器计数器 $KB0$ ($TKBCNT0$)和从属 16 位定时器计数器 KBm ($TKBCNTm$)与计数时钟同步地从 $FFFFH$ 改变为 $0000H$ ，并开始累加计数。同时，主和从属分别产生 $INTTMKB0$ 和 $INTTMKBm$ ，并且 $TKBO00$ 和 $TKB0m0$ 输出从默认值改变为有效值（在此例中为高电平）。关于操作的详情，请参阅图 7-38. 单机模式的操作时序例 (通过 $TKBCRn0$ 的周期控制) (输出的默认值为低电平($TKBTODnp = 0$)，有效电平为高电平($TKBTOLnp = 0$))。
- <2> 如果将 $TKBCE0$ 设置为 0，则作为主的 $TKBCNT0$ 和作为从属的 $TKBCNTm$ 将与定时器 KB 的输入时钟同步，且停止累加计数，成为 $FFFFH$ 。同时，主和从属的输出都成为各自的默认电平。该状态持续到作为主的 $TKBCE0$ 被设置为 1 为止。
- <3> 如果将作为主的 $TKBCE0$ 设置为 1，则从<1>开始重复同样的操作。

7.4.8 同步开始/清除模式

使用多个 KBn 定时器配置主/从属结构，从而通过与主定时器 $KB0$ 的计数开始/停止操作时序、计数器清除时序或批量覆盖时序同步，可以同时开始或清除从属定时器 KBm 。

在这种情况下，使主工作于“单机模式($TKBMD01, TKBMD00 = 00$)”，使从属工作于“同步开始/清除模式($TKBMDm1, TKBMDm0 = 10$)”。

在 $TKBCKS0$ 和 $TKBCKSm$ 位，给主和从属选择相同的分频时钟。

有关主的 $TKBO00/TKBO01$ 输出的计算公式，请参阅 7.4.5 单机模式(通过 $TKBCRn0$ 的周期控制)。

通过向主的 $TKBRDT0$ 写入“1”来控制批量覆盖。

读取 $TKBRSFn$ 标志时，需要验证主的 $TKBRSF0$ 。

清除主的 $TKBCNT0$ 时从属的 $TKBCNTm$ 也被清除。

对主执行批量覆盖时，也对从属比较寄存器执行批量覆盖。

从属工作依据主的 $TKBCR00$ 生成的周期，因此从属的 $TKBCRm0$ 成为了用于设置 $TKB0m0$ 的有效时序的寄存器。当检测到 $TKBCNTm$ 和 $TKBCRm0$ 匹配时，生成 $INTTMKBm$ 。然而，不输出开始计数操作时序的 $INTTMKBm$ 。

根据以下公式计算从属占空比（设置范围是 0%至 100%）。

[从属 $TKB0m0$ 输出的计算公式]

脉冲周期 = (主 $TKBCR00$ 的设置值 + 1) × 计数器时钟周期

占空比[%] = (($TKBCRm1$ 的设置值 - $TKBCRm0$ 的设置值) / (主 $TKBCR00$ 的设置值 + 1)) × 100

0%输出: $TKBCRm1$ 的设置值 = $TKBCRm0$ 的设置值

100%输出: $TKBCRm0$ 的设置值 = 0000H, $TKBCRm1$ 的设置值 ≥ 主 $TKBCR00$ 的设置值 + 1

注意事项 务必设置为 $TKBCRm0$ 的设置值 ≤ $TKBCRm1$ 的设置值。

[从属 $TKB0m1$ 输出的计算公式]

脉冲周期 = (主 $TKBCR00$ 的设置值 + 1) × 计数器时钟周期

占空比[%] = (($TKBCRm3$ 的设置值 - $TKBCRm2$ 的设置值) / (主 $TKBCR00$ 的设置值 + 1)) × 100

0%输出: $TKBCRm3$ 的设置值 = $TKBCRm2$ 的设置值

100%输出: $TKBCRm2$ 的设置值 = 0000H, $TKBCRm3$ 的设置值 ≥ 主 $TKBCR00$ 的设置值 + 1

注意事项 务必设置为 $TKBCRm2$ 的设置值 ≤ $TKBCRm3$ 的设置值。

备注 $m = 1, 2$

(1) 可在同步开始/清除模式下使用的主和从属的操作模式的组合

可在同步开始/清除模式下使用的主和从属的操作模式如下所示。

主:

操作模式	TKBMD01, TKBMD00	TKBSTS01, TKBSTS00	可设置
单机模式(通过TKBCR00的周期控制)	00B	00B	○
单机模式(通过外部触发输入的周期控制)	00B	01B/10B/11B	×
同时开始/停止模式(通过TKBCR00的周期控制)	01B	00B	×
同时开始/停止模式(通过外部触发输入的周期控制)	01B	01B/10B/11B	×
同步开始/清除模式(通过主的周期控制)	10B	—	×
交错式PFC输出模式	11B	—	×

从属:

操作模式	TKBMDm1, TKBMDm0	TKBSTSm1, TKBSTSm0	可设置
单机模式(通过TKBCRm0的周期控制)	00B	00B	×
单机模式(通过外部触发输入的周期控制)	00B	01B/10B/11B	×
同时开始/停止模式(通过TKBCRm0的周期控制)	01B	00B	×
同时开始/停止模式(通过外部触发输入的周期控制)	01B	01B/10B/11B	×
同步开始/清除模式(通过主的周期控制)	10B	—	○
交错式PFC输出模式	11B	—	×

(2) 同步开始/清除模式：主的寄存器设置列表

位编号	15	14	13	12	11	10	9	8
TKBCTL00	-	TKBGTE01	TKBSSE01	TKBDIE01	-	TKBGTE00	TKBSSE00	TKBDIE00
设置	0	1/0	1/0	1/0	0	1/0	1/0	1/0

位编号	7	6	5	4	3	2	1	0
TKBCTL00	TKBMFE0	-	TKBIRS01	TKBIRS00	-	TKBTSE0	TKBSTS01	TKBSTS00
设置	0	0	0	0	0	0	0	0

位编号	7	6	5	4	3	2	1	0
TKBCTL01	TKBCE0	-	-	TKBCKS0	TKBSCM0	-	TKBMD01	TKBMD00
设置	1	0	0	1/0	0	0	0	0

位编号	7	6	5	4	3	2	1	0
TKBIOC00	-	-	-	-	TKBTOL01	TKBTOL00	TKBTOD01	TKBTOD00
设置	0	0	0	0	1/0	1/0	1/0	1/0

位编号	7	6	5	4	3	2	1	0
TKBIOC01	-	-	-	-	-	-	TKBTOE01	TKBTOE00
设置	0	0	0	0	0	0	1/0	1/0

TKBCR00	0000H-FFFFH
TKBCR01	0000H-FFFFH
TKBCR02	0000H-FFFFH
TKBCR03	0000H-FFFFH
TKBTGCR0	0000H-FFFFH
TKBSIR00	0000H-FFFFH
TKBSIR01	0000H-FFFFH
TKBSSR00	00H-0FH
TKBSSR01	00H-0FH
TKBDNR00	00H-F0H
TKBDNR01	00H-F0H
TKBMFR0	0000H

：在此模式下设置被固定 ：不需要设置（默认设置）

(3) 同时开始/停止模式：从属的寄存器设置列表

位编号	15	14	13	12	11	10	9	8
TKBCTLm0	-	TKBGTEm1	TKBSSEm1	TKBDIEm1	-	TKBGTEm0	TKBSSEm0	TKBDIEm0
设置	0	1/0	1/0	1/0	0	1/0	1/0	1/0

位编号	7	6	5	4	3	2	1	0
TKBCTLm0	TKBMFEm	-	TKBIRSm1	TKBIRSm0	-	TKBTSEm	TKBSTSm1	TKBSTSm0
设置	0	0	0	0	0	0	0	0

位编号	7	6	5	4	3	2	1	0
TKBCTLm1	TKBCEm	-	-	TKBCKSm	TKBSCMm	-	TKBMDm1	TKBMDm0
设置	1	0	0	1/0	0	0	1	0

位编号	7	6	5	4	3	2	1	0
TKBIOCM0	-	-	-	-	TKBTOLm1	TKBTOLm0	TKBTODm1	TKBTODm0
设置	0	0	0	0	1/0	1/0	1/0	1/0

位编号	7	6	5	4	3	2	1	0
TKBIOCM1	-	-	-	-	-	-	TKBTOEm1	TKBTOEm0
设置	0	0	0	0	0	0	1/0	1/0

TKBCRm0	0000H-FFFFH
TKBCRm1	0000H-FFFFH
TKBCRm2	0000H-FFFFH
TKBCRm3	0000H-FFFFH
TKBTGCRm	0000H-FFFFH
TKBSIRm0	0000H-FFFFH
TKBSIRm1	0000H-FFFFH
TKBSSRm0	00H-0FH
TKBSSRm1	00H-0FH
TKBDNRm0	00H-F0H
TKBDNRm1	00H-F0H
TKBMFRm	0000H

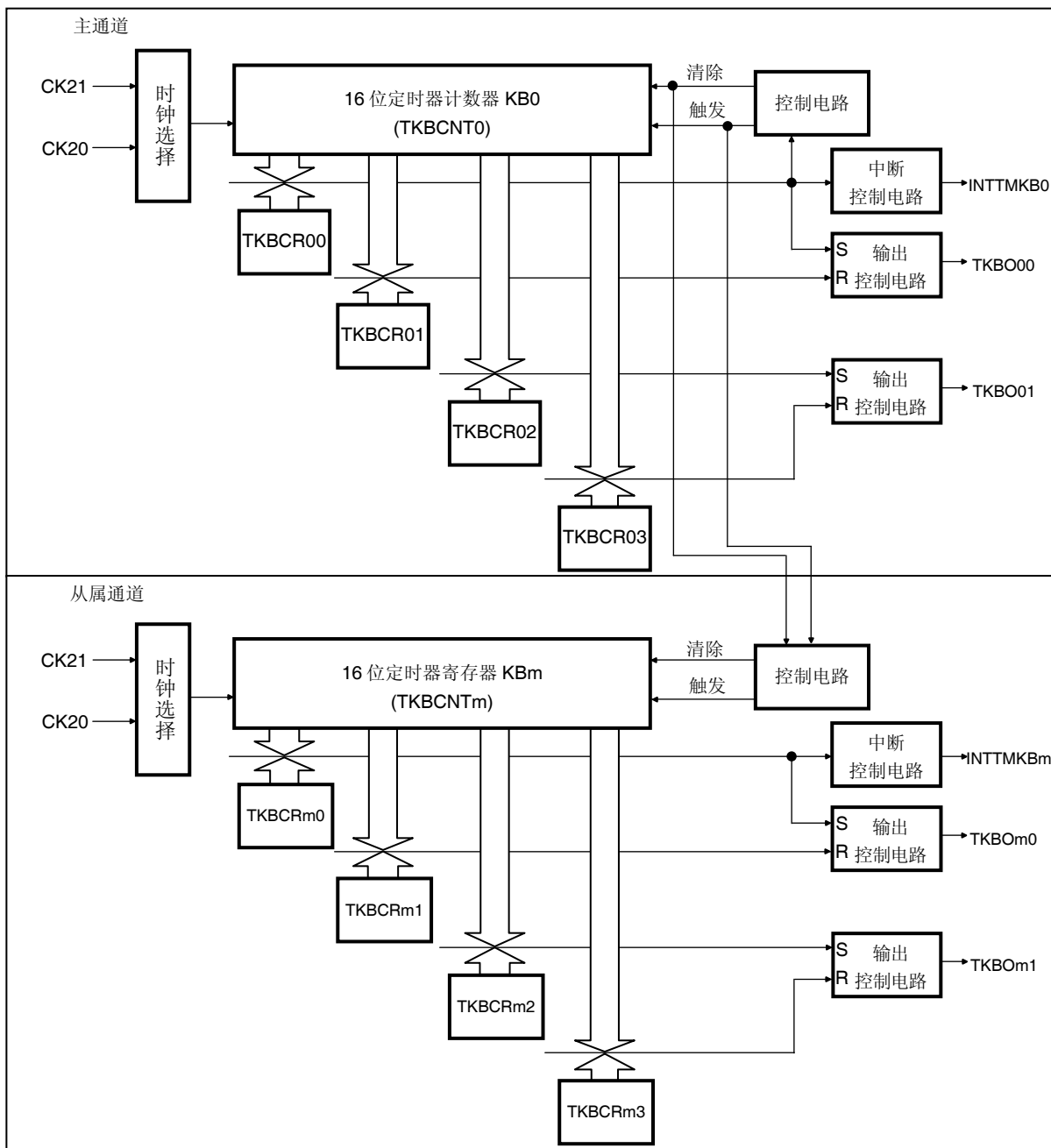
：在此模式下设置被固定 ：不需要设置（默认设置）

备注 m = 1, 2

(4) 同步开始/清除模式(通过主的周期控制)的配置

同步开始/清除模式下的配置如图 7-46 所示。

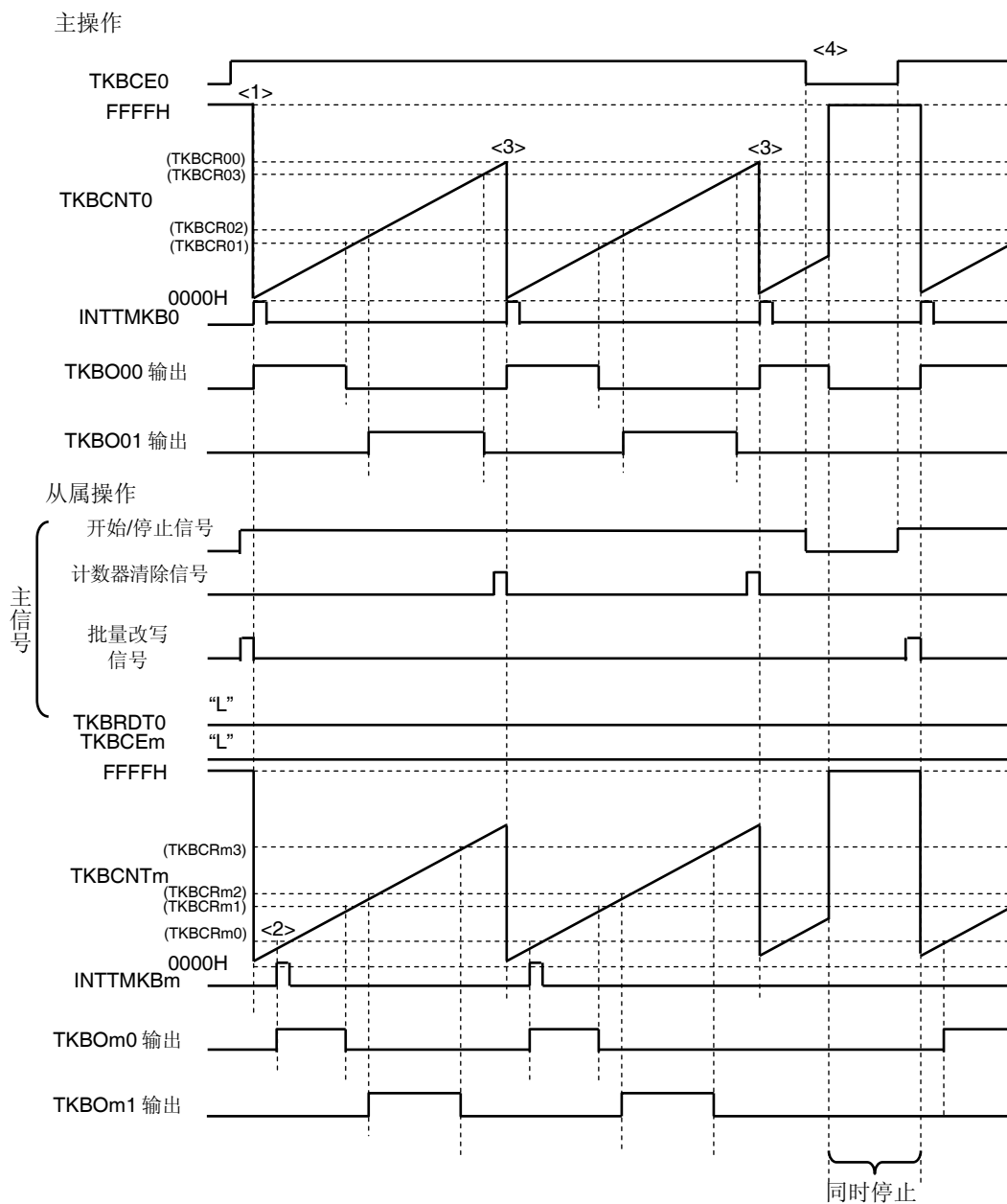
图 7-46. 同步开始/清除模式(通过主的周期控制)的配置



(5) 操作概要

同步开始/清除模式下的操作时序例如图 7-47 所示。

图 7-47. 同步开始/清除模式的操作时序例 (通过主的周期控制)
 (输出的默认值为低电平(TKBTODnp = 0), 有效电平为高电平(TKBTOLnp = 0))



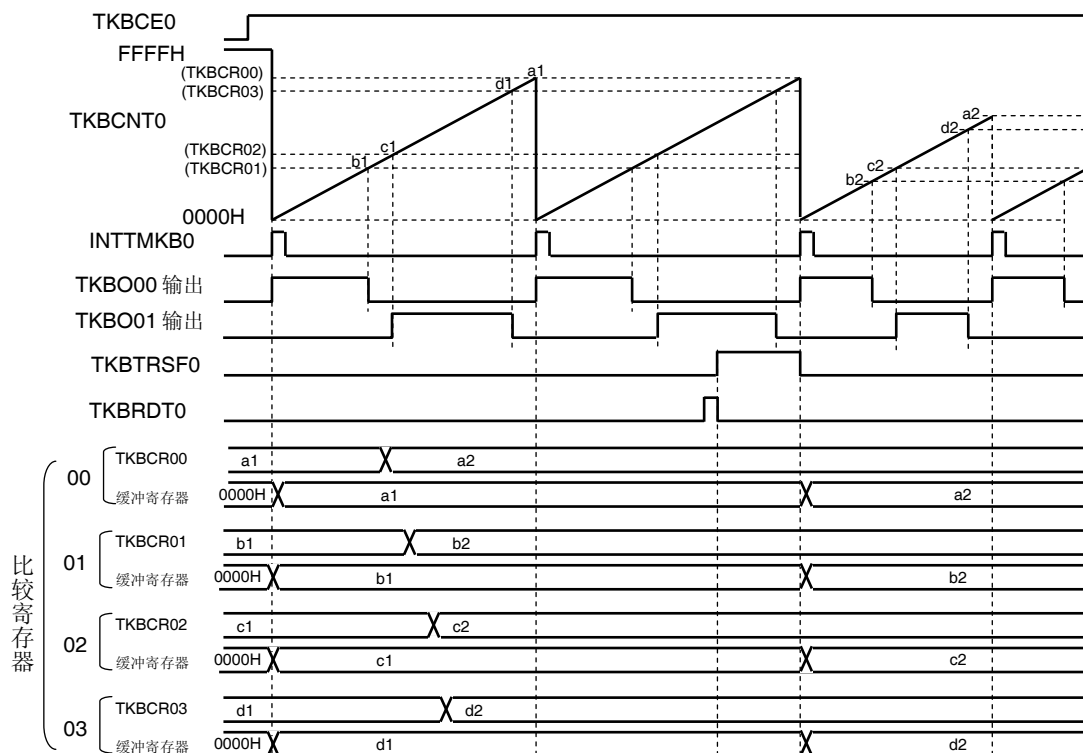
同步开始/清除模式的操作示例如下所示。下列描述与图 7-47 的<1>至<4>相关联。

- <1> 将作为主的 TKBCE0 设置为 1 时，主 16 位定时器计数器 KB0 (TKBCNT0)和从属 16 位定时器计数器 KBm (TKBCNTm)与计数时钟同步地从 FFFFH 改变为 0000H，并开始累加计数。同时，主产生 INTTMKB0，并且 TKBO00 输出从默认值改变为有效值（在此例中为高电平）。
- <2> 当 TKBCNTm 的计数值与 TMKB 比较寄存器 n2 (TKBCRn2)指定的值匹配时，从属的 TKBOn1 输出将成为有效电平。关于操作的详情，请参阅图 7-38. **单机模式的操作时序例 (通过 TKBCRn0 的周期控制) (输出的默认值为低电平(TKBTODnp = 0)，有效电平为高电平(TKBTOLnp = 0))。**
- <3> 当 TKBCNT0 的计数值与 TMKB 比较寄存器 00 (TKBCR00)指定的值匹配时，输出主的清除信号。同时，主和从属的 16 位定时器计数器(TKBCCNT0, TKBCNTm)被清除。
- <4> 如果将 TKBCE0 设置为 0，则作为主的 TKBCNT0 和作为从属的 TKBCNTm 将与定时器 KB 的输入时钟同步，且停止累加计数，成为 FFFFH。同时，主和从属的输出都成为各自的默认电平。该状态持续到作为主的 TKBCE0 被设置为 1 为止。

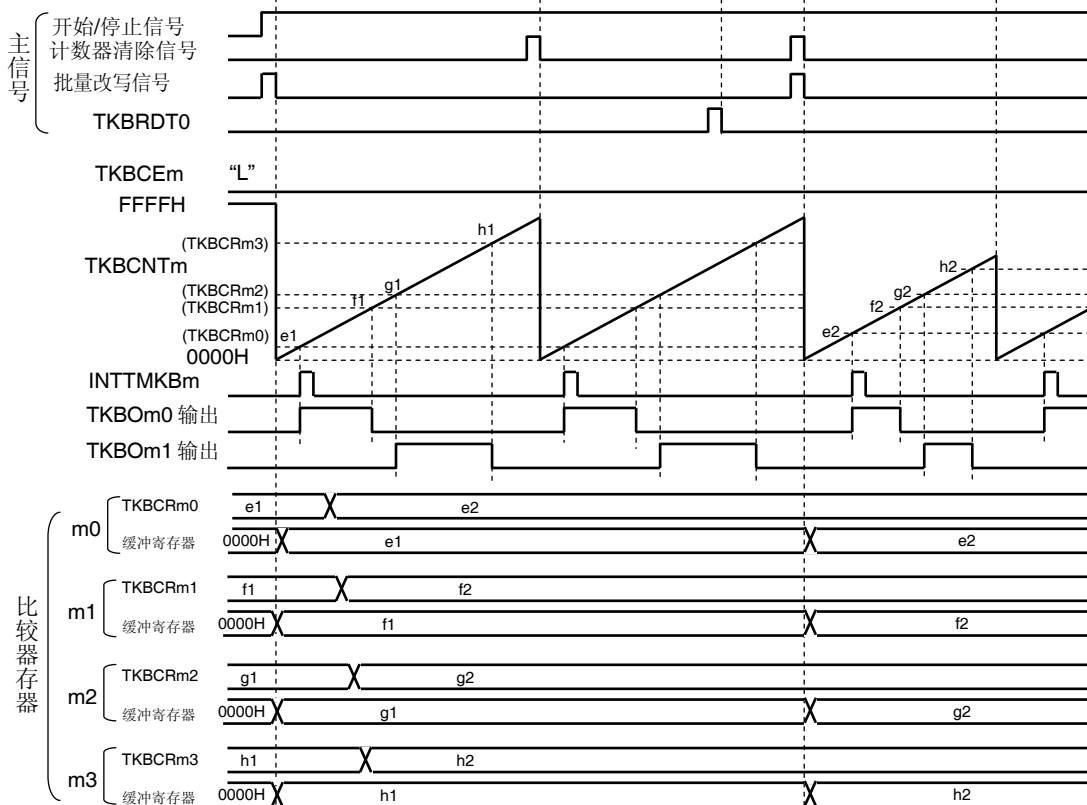
同步开始/清除模式下批量覆盖的操作时序示例如图 7-48 所示 这种情况下，将主的 TKBRDT0 位设置为 1，则在下一个清除时序处将对从属执行批量覆盖。

图 7-48. 同步开始/清除模式的操作时序例 (通过主的周期控制)
(批量覆盖时)

主操作



从属操作



7.4.9 交错式PFC (Power Factor Correction)输出模式

交错式 PF 输出模式可以输出用于控制 PFC 电路的信号，PFC 电路可以调节电源的谐波电流。

与单一式 PFC 电路相比，交错式 PFC 电路能够抑制峰值输入电流，由此可以实现部件的小型化以及电源的大功率化。

交错式 PFC 控制需要两个用来检测零电流的输入，以及用于切换的两个 PWM 输出。TMKB 是利用外部中断输入 INTP20 和定时器输出 TKBOn0、以及中断输入 INTP21 和定时器输出 TKBOn1 的组合，来执行交错式 PFC 控制。

对应于 INTP20 的信号输入，TKBOn0 输出脉冲；对应于 INTP21 的信号输入，TKBOn1 输出脉冲。

这种情况下，依据 TKBOn0 的输出时序，控制 TKBOn1 输出为相位偏移 180 度的状态。

备注 可以在单机模式(通过外部触发输入的周期控制)下执行单一 PFC 控制。

详情请参阅 7.4.6 单机模式(通过外部触发输入的周期控制)。

没有检测到外部输入 INTP20 的情况下，定时器 KB 重启周期由 TKBCRn0 设置。

TKBOn0 输出的有效宽度由 TKBCRn1 设置。

TKBOn1 输出的有效宽度由 TKBCRn3 设置。

因此，该功能不使用 TKBCRn2。

备注 在交错式 PFC (Power Factor Correction)输出模式下不使用 TKBCRn2。

TKBTOLn0 位和 TKBTODn0 位的设置值以及 TKBTOLn1 位和 TKBTODn1 位的设置值必须相同。因此，当默认电平为低（高）电平时，有效电平将为高（低）电平。

[TKBOn0 输出&TKBOn1 输出的计算公式]

脉冲周期 (最大值)^注 = (TKBCRn0 的设置值 + 1) × 计数时钟周期

TKBOn0 输出的有效宽度 = TKBCRn1 的设置值

TKBOn1 输出的有效宽度 = TKBCRn3 的设置值

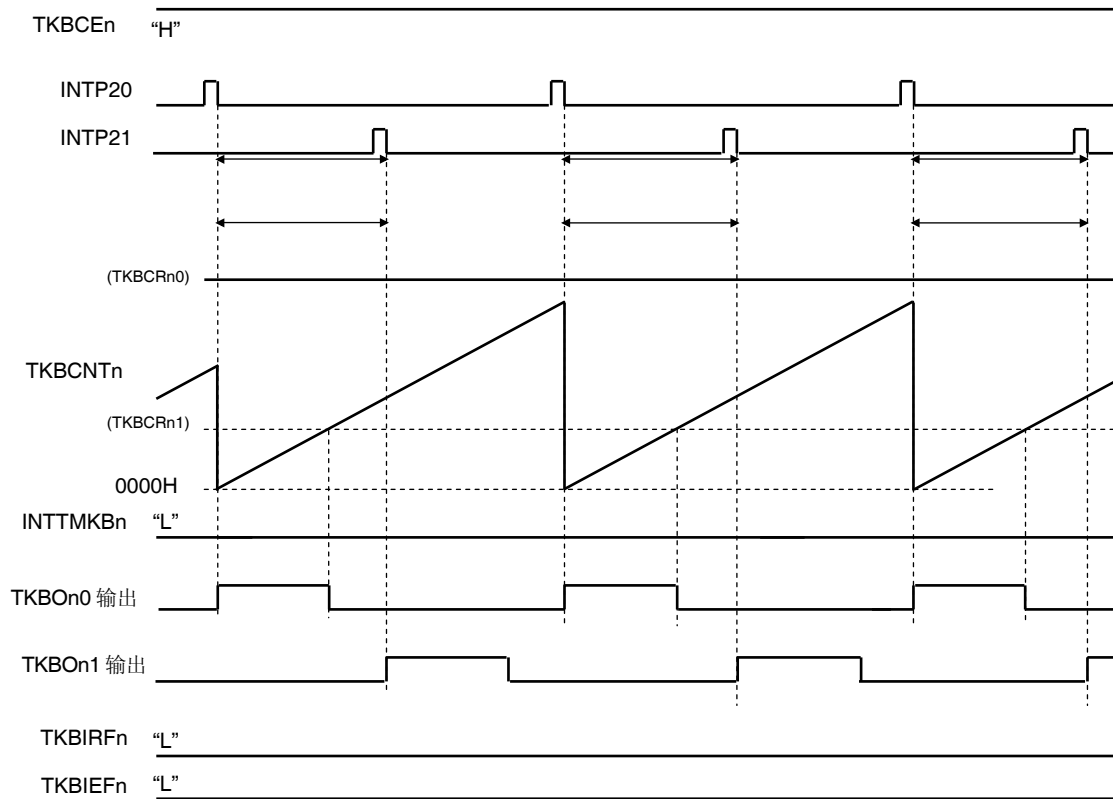
注 没有检测到外部输入 INTP20 时的定时器 KBn 重启周期。

图 7-49 显示了交错式 PFC 模式的基本操作概况。在交错式 PFC 模式的基本操作中，以 INTP20 为触发，TKBCNTn 被清除并且从 0000H 开始累加计数。在此情况下，TKBOn0 为有效电平，并在与 TKBCRn1 寄存器的设置值匹配时变为无效电平。

TKBOn1 经 INTP21（具有相对于 INTP20 的相位发生了偏移的相位）触发后成为有效电平，并在与 TKBCRn3 寄存器的设置值匹配时改变为无效电平。

在 TKBCNTn 与 TKBCRn0 寄存器的设置值匹配前，输入另一个 INTP20，然后重复上述操作。

图 7-49. 交错式 PFC 模式基本操作的操作概要
 (输出的默认值为低电平(TKBTODnp = 0), 有效电平为高电平(TKBTOLnp = 0))



(1) 在交错式 PFC 下的 TKBOn1 的输出条件

按照下表控制 TKBOn1 输出须具备的输出条件。

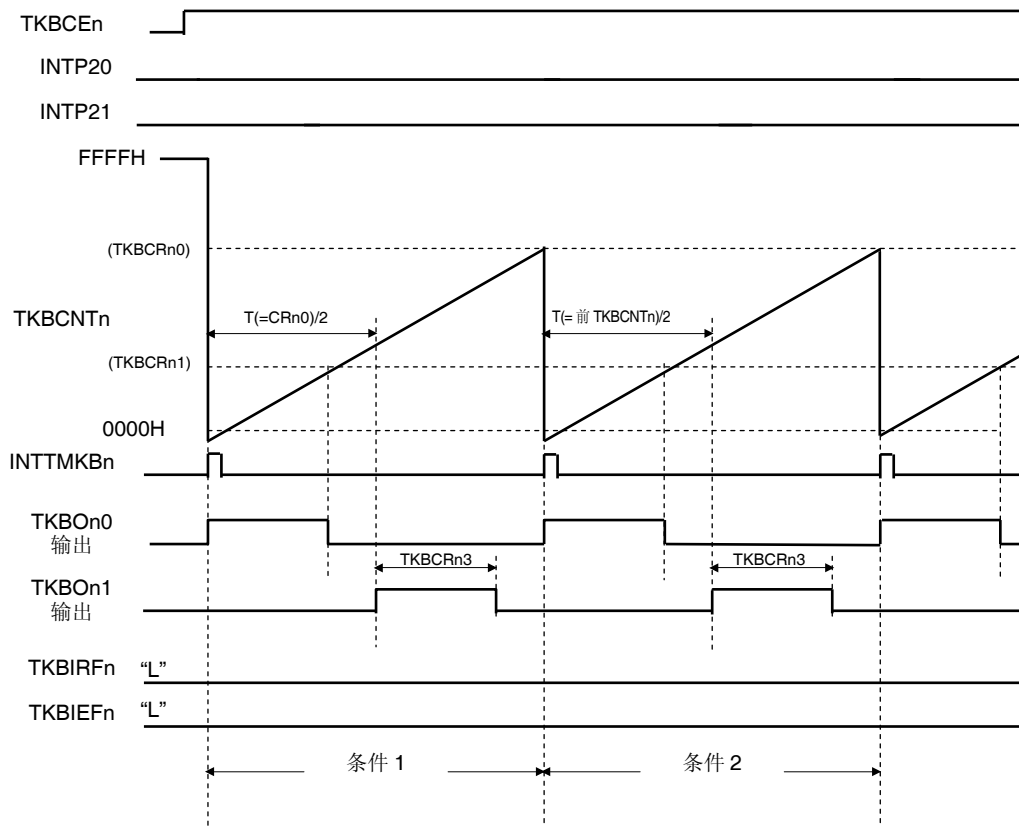
条件编号	判定状态1 (INTP20输入)	判定状态2 (与CR0匹配/INTP21输入)	判定状态3 (周期宽幅)	输出状态
1	第1个周期 (将CR0设置为T并输出波形)	-	-	在T/2输出
2	未检测到INTP20输入	CNTn和CRn0匹配 (忽略INTP21输入检测)	下一个周期超过T/2时	在T/2输出
3	↑	↑	下一个周期低于T/2时	保持状态
4	No.3的下一个周期	-	-	在T/2输出
5	检测到INTP20输入(第一次)*1	-	-	在T/2输出
6	检测到INTP20输入(第二次之后)*2	检测到INTP21(在上一个TKBOn1下降沿到T/2的范围内)	-	在T/2输出
7	检测到INTP20输入(第二次之后)*2	检测到INTP21 (T/2到T/2+T/(TKBIRSn1-TKBIRSn0的设置值)的范围内)	-	在触发输入时输出
8	检测到INTP20输入(第二次之后)*2	检测到INTP21 (T/2到T/2+T/(TKBIRSn1-TKBIRSn0的设置值)的范围之后)	-	保持状态
9	No.8的下一个周期	-	-	在T/2输出
10	检测到INTP20输入	-	下一个周期低于T/2时	保持状态
11	No.10的下一个周期	-	-	在T/2输出

*1 (首次) 检测到 INTP20 输入表示上一周期没有因为检测到 INTP20 而被清除。

*2 (从第二次开始) 检测到 INTP20 输入表示上一周期因为检测到 INTP20 被清除。

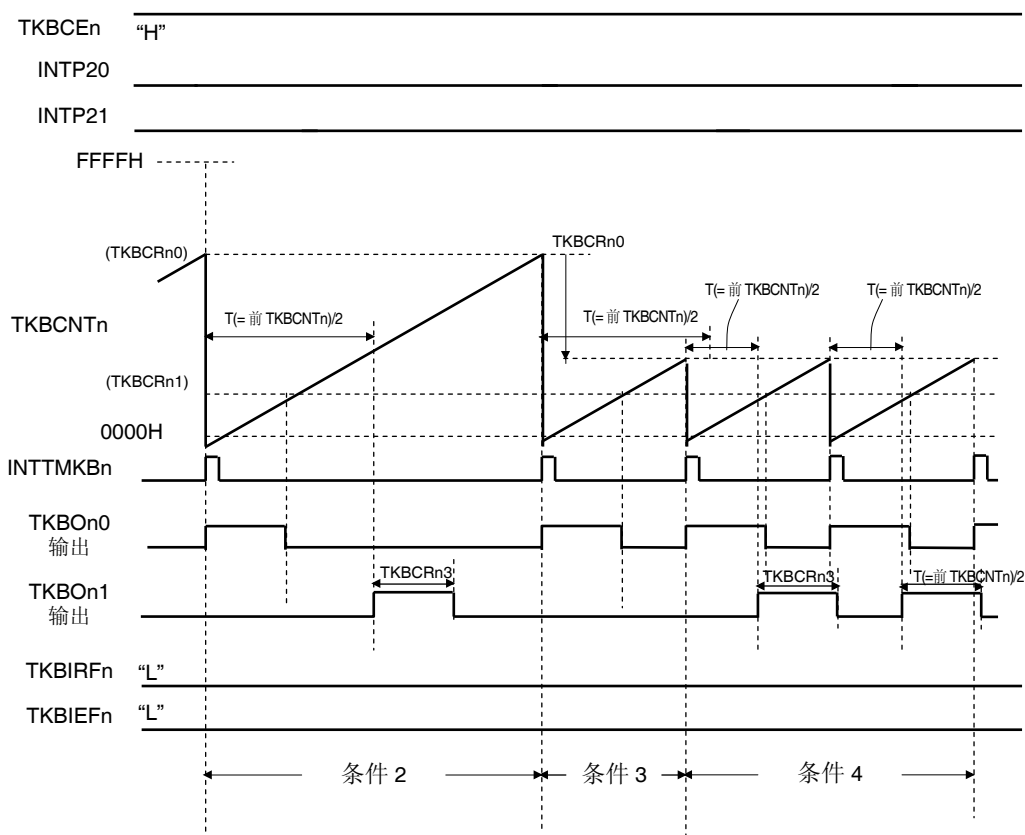
请参见下面与每个“条件编号”相对应的波形图。

图 7-50. 交错式 PFC 模式的时序图(在条件 No. 1 至 2 的操作)



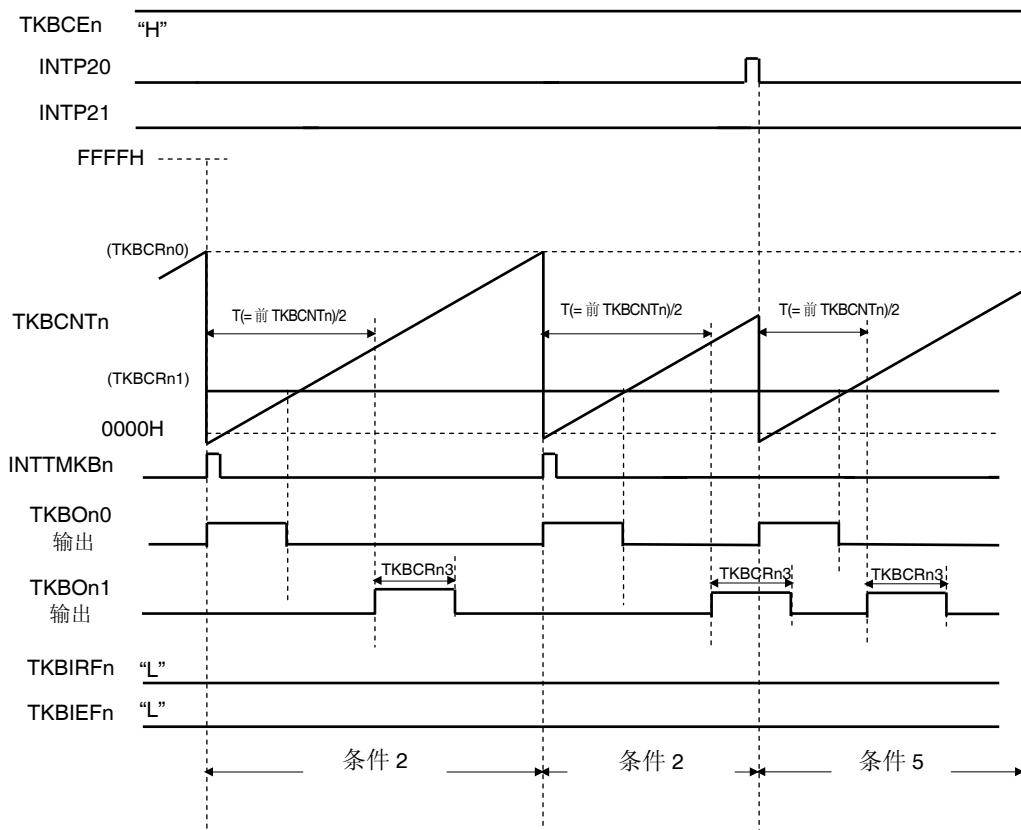
- 条件 No.1 仅在设置 TKBCEn = 1 后的第一个周期，将 TKBCRn0 设置为“T”时在 T/2 处输出采用 TKBCRn3 的设置宽度的 TKBOn1。
- 条件 No.2 在第二个周期，将在上一周期的 T/2 处输出采用 TKBCRn3 的设置宽度的 TKBOn1。

图 7-51. 交错式 PFC 模式的时序图(在 No. 3 至 4 的操作)
 (输出的默认值为低电平(TKBTODnp = 0), 有效电平为高电平(TKBTOLnp = 0))



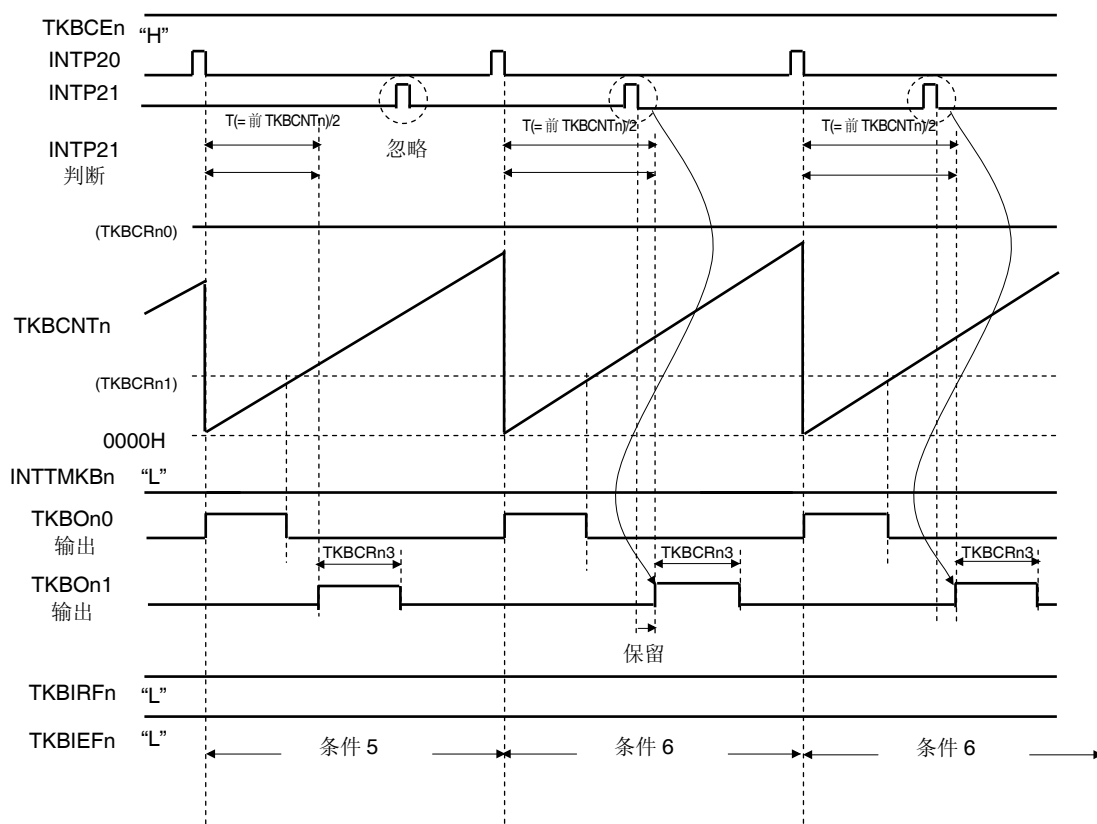
- 条件 No.3 不能确保上一周期的 $T/2$ 时, **TKBOn1** 保持状态不变。
- 条件 No.4 在上一周期的 $T/2$ 处输出采用 **TKBCRn3** 设置宽度的 **TKBOn1**。

图 7-52. 交错式 PFC 模式的时序图
(在条件 No. 5 的操作：尚未达到 INTP21)



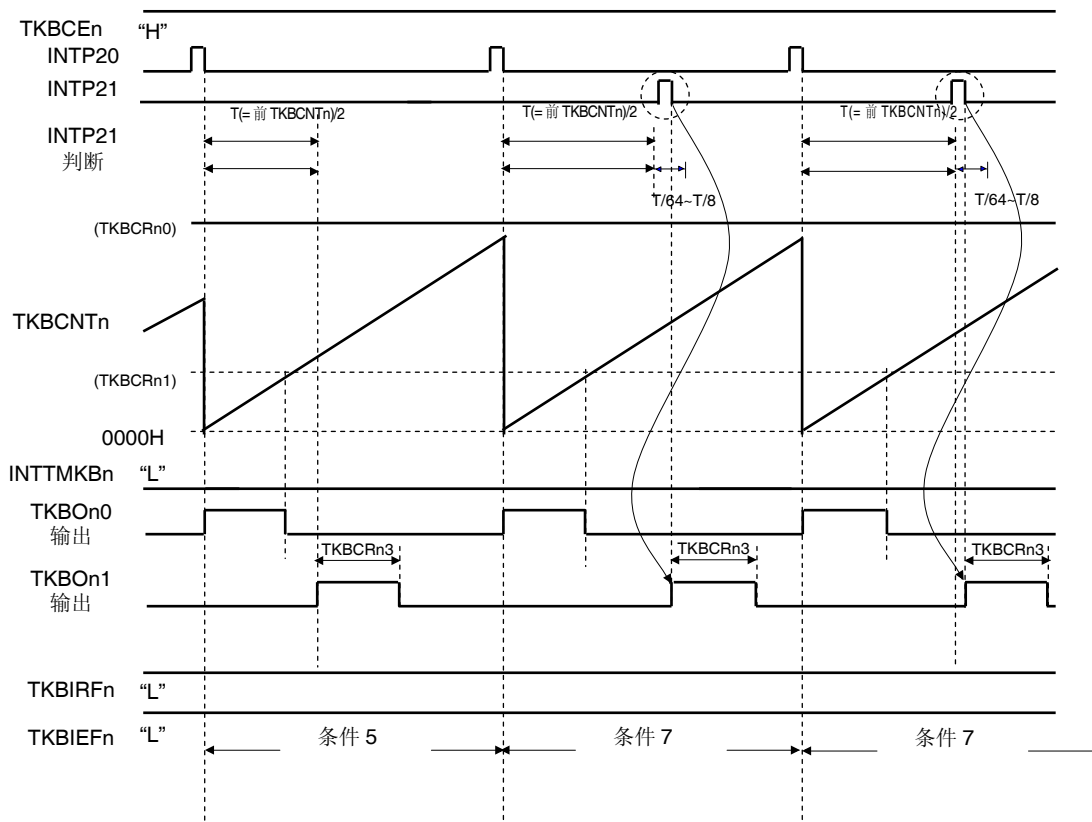
条件 No.5 设置 TKBCEn = 1 后，首次检测到的 INTP20 输出采用 TKBCRn3 设置宽度的 TKBOn1。

图 7-53. 交错式 PFC 模式的时序图(在条件 No. 6 的操作)
 (输出的默认值为低电平(TKBTODnp = 0), 有效电平为高电平(TKBTOLnp = 0))



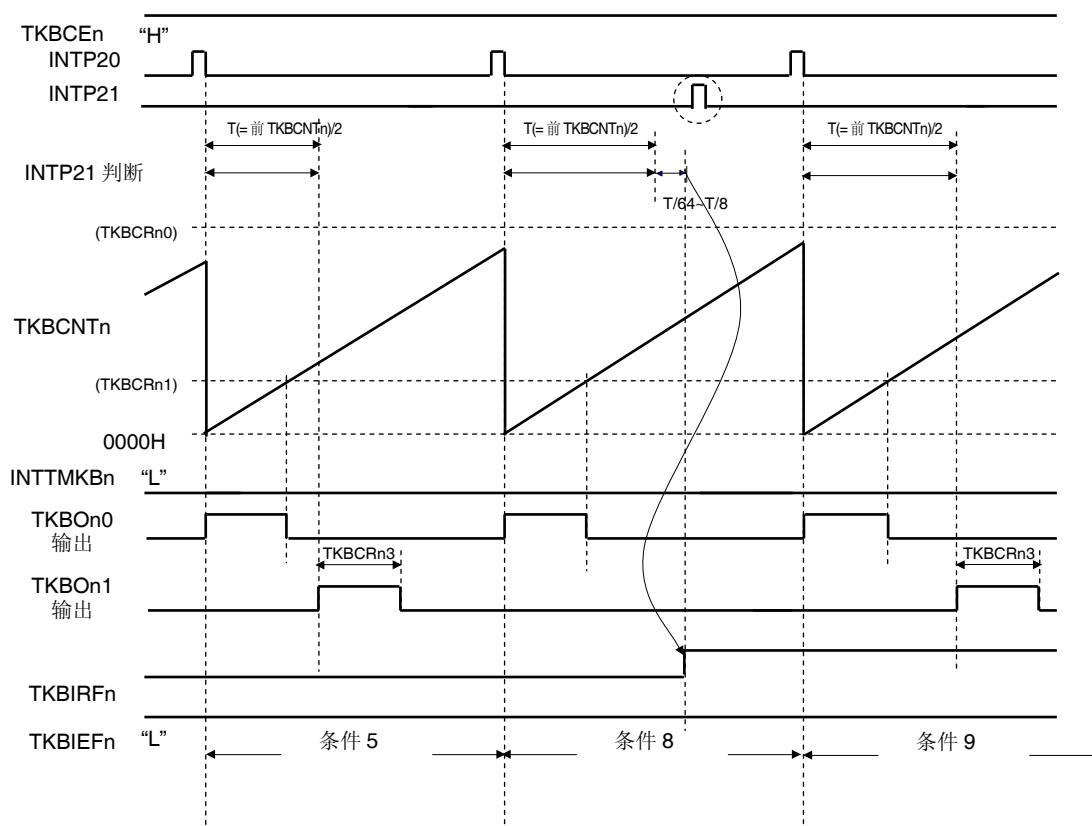
条件 No.6 由于 INTP21 输入在上一周期的 T/2 以内, 因此将在上一周期的 T/2 处输出采用 TKBCRn3 设置宽度的 TKBOn1。

图 7-54. 交错式 PFC 输出模式的时序图(在条件 No. 7 的操作)
 (输出的默认值为低电平(TKBTODnp = 0), 有效电平为高电平(TKBTOLnp = 0))



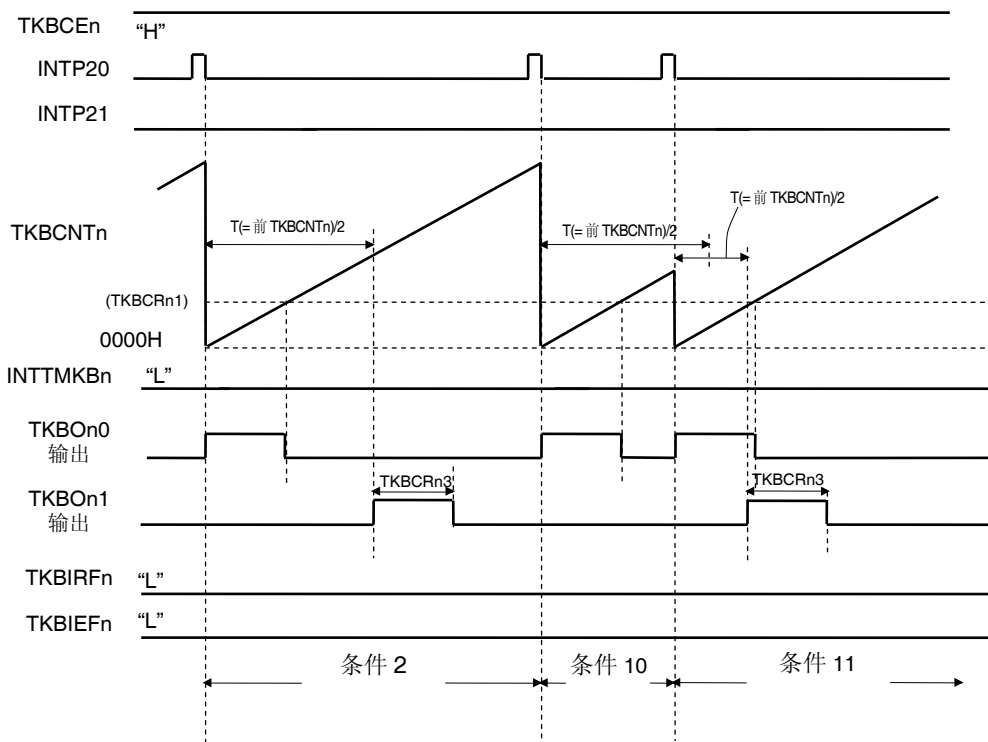
条件 No.7 检测到 INTP20 后, 在上一周期的 $T/2$ 到 $T/2 + T/m$ (m 表示 8/16/32/64, 由 TKBIRSn1、TKBIRSn0 设置) 的范围内检测到 INTP21 时, 输出采用 TKBCRn3 的设置宽度的 TKBOn1。

图 7-55. 交错式 PFC 输出模式的时序图(在条件 No. 8 至 9 的操作)
 (输出的默认值为低电平(TKBTODnp = 0), 有效电平为高电平(TKBTOLnp = 0))



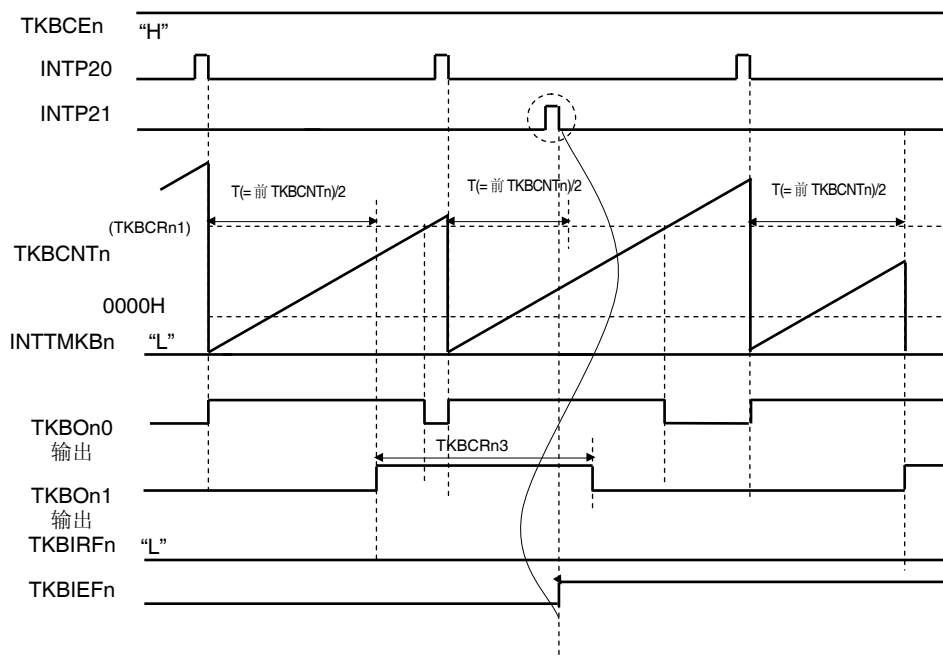
- 条件 No.8 在上一周期的 $T/2 + T/m$ (m 表示 8/16/32/64, 由 $TKBIRSn1$ 和 $TKBIRSn0$ 设置) 范围内未检测到 $INTP21$ 时, $TKBOn1$ 保持状态不变。此时 $TKBIRFn$ 被设置为“1”。
- 条件 No.9 在上一周期的 $T/2$ 处输出采用 $TKBCRn3$ 设置宽度的 $TKBOn1$ 。

图 7-56. 交错式 PFC 输出模式的时序图(在条件 No. 10 至 11 的操作)
 (输出的默认值为低电平(TKBTODnp = 0), 有效电平为高电平(TKBTOLnp = 0))



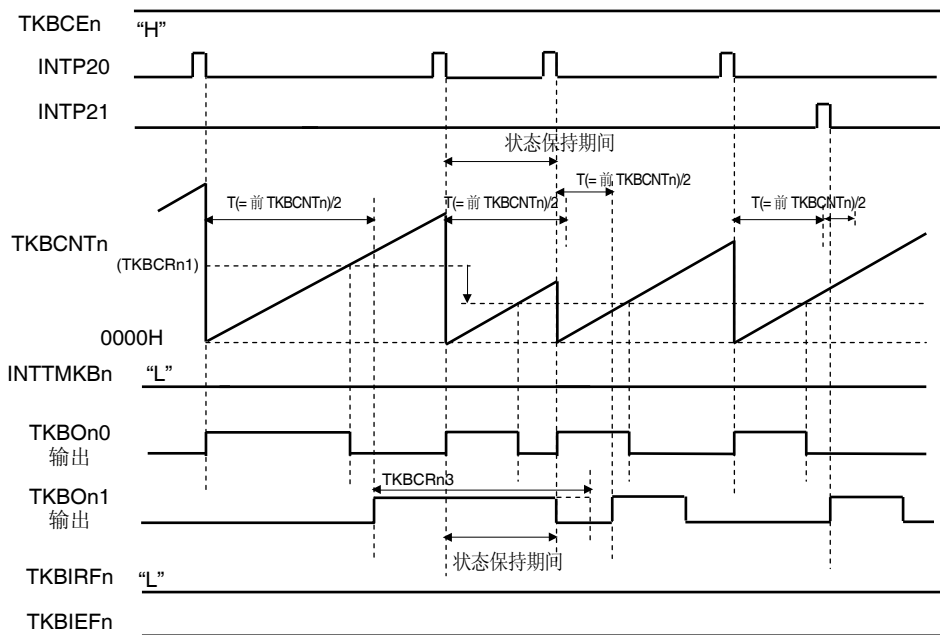
- 条件 No.10 不能确保上一周期的 T/2 时, TKBOn1 保持状态不变。
- 条件 No.11 在上一周期的 T/2 处输出采用 TKBCRn3 设置宽度的 TKBOn1。

图 7-57. 交错式 PFC 输出模式的时序图
(在 TKBOn1 期间再次生成触发的情况下)



在上一个周期的 TKBOn1 输出期间生成了之后的 TKBOn1 输出触发时，忽略该触发。此时 TKBIEFn 被设置为“1”。

图 7-58. 交错式 PFC 输出模式的时序图
(TKBOn1 输出的宽度为上次输出的宽度，并超出状态维持期间)



如果 TKBOn1 输出的上次输出宽度超过了状态维持期间，则在状态维持期间结束后的下一个周期的开始时序强制设置为默认输出。

(2) 在交错式 PFC 输出模式下的寄存器设置列表

位编号	15	14	13	12	11	10	9	8
TKBCTLn0	-	TKBGTEn1	TKBSSEn1	TKBDIE1	-	TKBGTEn0	TKBSSEn0	TKBDIE0
设置	0	0	0	0	0	0	0	0

位编号	7	6	5	4	3	2	1	0
TKBCTLn0	TKBMFEn	-	TKBIRSn1	TKBIRSn0	-	TKBTSEn	TKBSTSn1	TKBSTSn0
设置	1/0	0	0/1	0/1	0	1	0	0

位编号	7	6	5	4	3	2	1	0
TKBCTLn1	TKBCEn	-	-	TKBCKSn	TKBSCMn	-	TKBMDn1	TKBMDn0
设置	1	0	0	1/0	0	0	1	1

位编号	7	6	5	4	3	2	1	0
TKBIOcn0	-	-	-	-	TKBTOLn1	TKBTOLn0	TKBTODn1	TKBTODn0
设置	0	0	0	0	1/0	1/0	1/0	1/0

位编号	7	6	5	4	3	2	1	0
TKBIOcn1	-	-	-	-	-	-	TKBTOEn1	TKBTOEn0
设置	0	0	0	0	0	0	1/0	1/0

TKBCRn0	0000H-FFFFH
TKBCRn1	0000H-FFFFH
TKBCRn2	0000H
TKBCRn3	0000H-FFFFH
TKBTGCRn	0000H-FFFFH
TKBSIRn0	0000H
TKBSIRn1	0000H
TKBSSRn0	00H
TKBSSRn1	00H
TKBDNRn0	00H
TKBDNRn1	00H
TKBMFRn	0000H-FFFFH

：在此模式下设置被固定 ：不需要设置（默认设置）

7.5 16位定时器KB0、KB1和KB2的选项功能

可以给定时器KB0、KB1和KB2添加选项功能。

下表显示了定时器KB0、KB1和KB2的各个操作模式的可用选项。

操作模式		单机模式		同时开始/停止模式		同步开始/清除模式	交错式PFC输出模式
在各操作模式下的周期控制方法		通过CR0的周期控制	通过触发的周期控制	通过CR0的周期控制	通过触发的周期控制	通过主的周期控制	通过INTP20/CR0的周期控制
选项功能	A/D转换开始时序信号输出功能	○	○	○	○	○	○
	PWM输出抖动功能	○	×	○	×	○	×
	PWM输出软启动功能	○	×	○	×	○	×
	PWM输出门控功能	○	×	○	×	○	×
	最大频率限值功能	×	○	×	○	×	○

备注 关于操作规范的详情，请参阅7.4.2 默认电平和有效电平以及7.4.3 停止/重启操作。

7.5.1 A/D转换开始时序信号输出功能

通过设置 16 位定时器 KB 触发比较寄存器 n (TKBTGCRn)，可生成 A/D 转换开始时序信号输出。由此，可以实现 16 位定时器 KBn 和 A/D 转换开始时序的同步。

在检测到 TKBCNTn 和 TKBTGCRn 匹配时，输出定时器 KBn 触发输出信号，从而在对应于 TKBCRnm 设置周期的任何时序都可以输出触发定时器 KBn 触发输出信号的输出宽度为定时器时钟的 1 个时钟的宽度。根据以下公式可计算从 PWM 输出周期开始的触发输出时序：

触发输出时序 = TKBTGCRn 的设置值 × 计数时钟周期

注意事项 当 $TKBCRn0 < TKBTGCRn$ 时，不能输出定时器 KBn 触发输出信号。

图 7-59. 在单机模式(通过 TKB0CR0 的周期控制)下的 A/D 转换开始时序信号输出功能

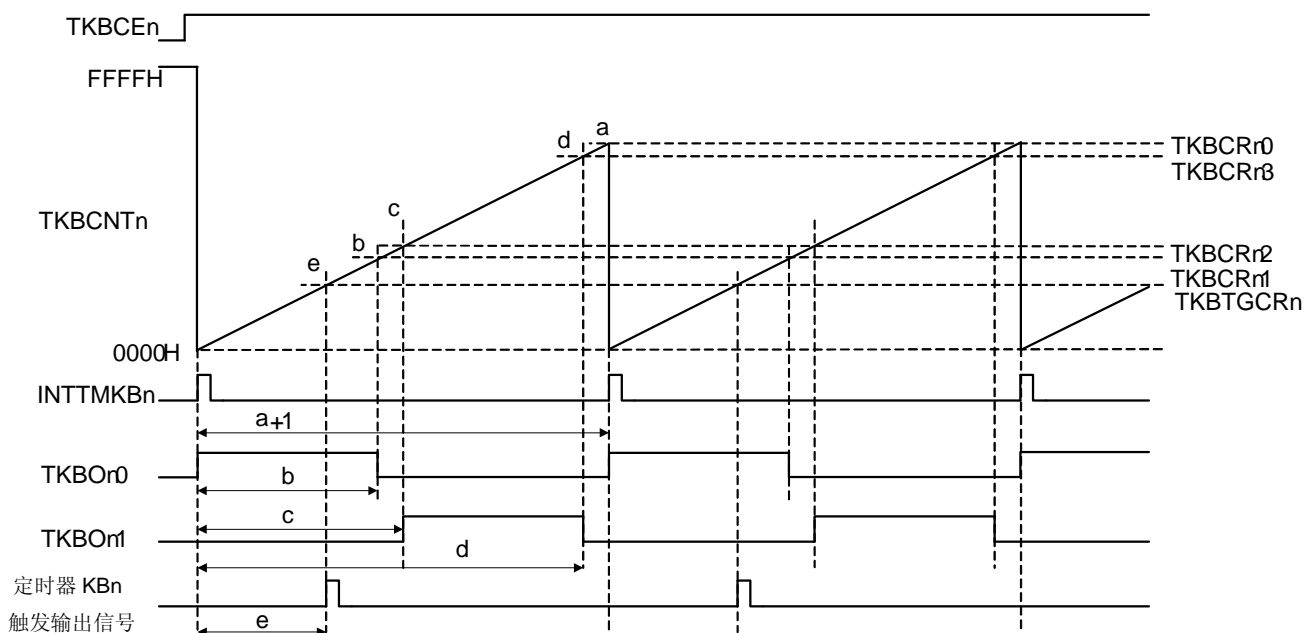
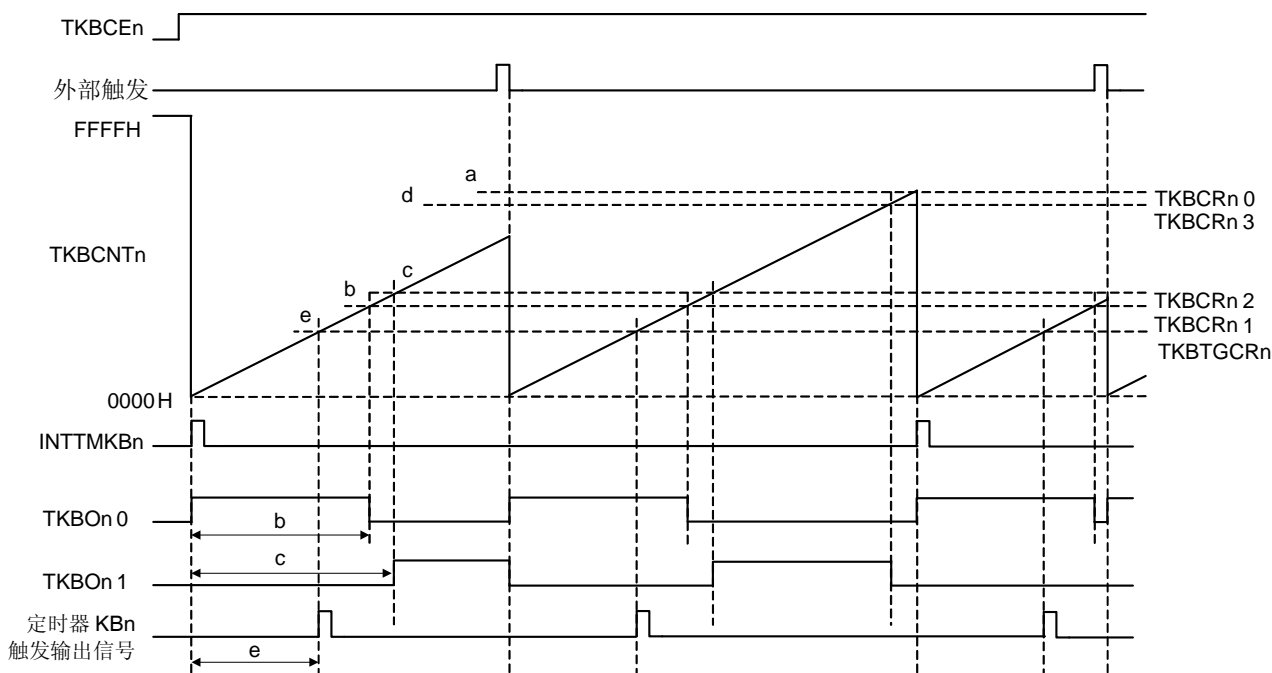


图 7-60. 在单机模式(通过外部触发输入的周期控制)下的 A/D 转换开始时序信号输出功能



7.5.2 PWM输出抖动功能

利用 PWM 输出抖动功能，16 位定时器 KB 可进行高分辨率 PWM 输出。

以 PWM 周期的 16 个周期为基准，在 16 个周期中的 n 个周期($n = 0$ 至 15)处将有效期间延长 1 个计数时钟，从而可以实现提高至 16 倍的 PWM 输出作为平均分辨率。

通过 TKBDNRnp 来定义 16 个周期中将有效期间延长 1 个计数时钟的周期。

TKBDNRnp 和将有效期间延长 1 个计数时钟的周期之间的关系如下所示：

图 7-61. TKBDNRnp 和将有效期间延长 1 个计数时钟的周期之间的关系图

周期 次数(N)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0																
1	■															
2	■								■							
3	■				■				■							
4	■				■				■				■			
5	■		■		■				■				■			
6	■		■		■				■			■				
7	■		■		■		■		■			■				
8	■		■		■		■		■			■			■	
9	■	■			■		■		■			■			■	
10	■	■			■		■		■	■		■			■	
11	■	■	■		■		■		■	■		■			■	
12	■	■	■		■		■		■	■		■			■	
13	■	■	■	■		■		■	■		■				■	
14	■	■	■	■	■		■		■	■		■	■		■	
15	■	■	■	■	■	■		■	■	■		■	■	■	■	

- 备注 1. 单元格的周期：通过 TKBnCR1 和 TKBnCR3 寄存器的设置值使输出波形复位
 单元格的周期：通过“TKBnCR1 和 TKBnCR3 寄存器的设置值 + 1”使输出波形复位
2. $n = 0$ 至 2, $p = 0, 1$

图 7-62. 抖动操作的波形图

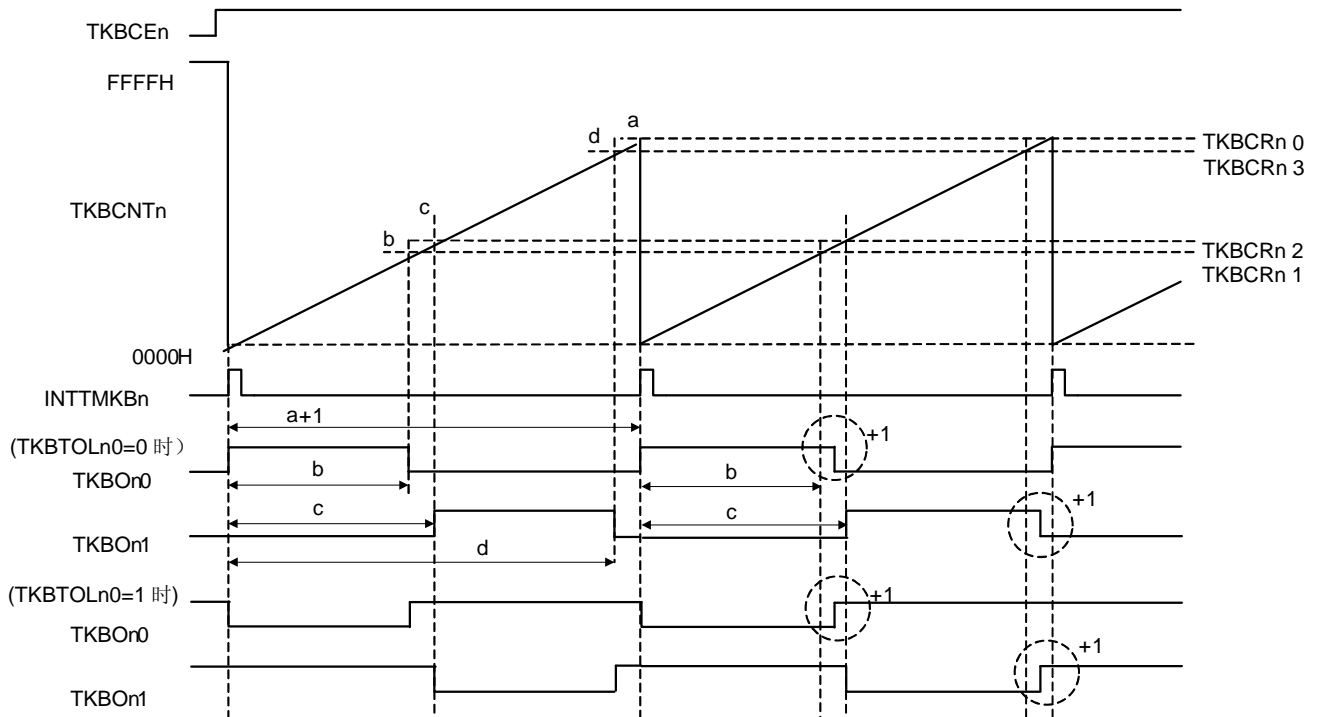


图 7-63. 抖动操作的波形图
(当 TKBCRn1 = TKBCRn0 (接近 100%), TKBCRn2 = TKBCRn3(接近 0%)时)

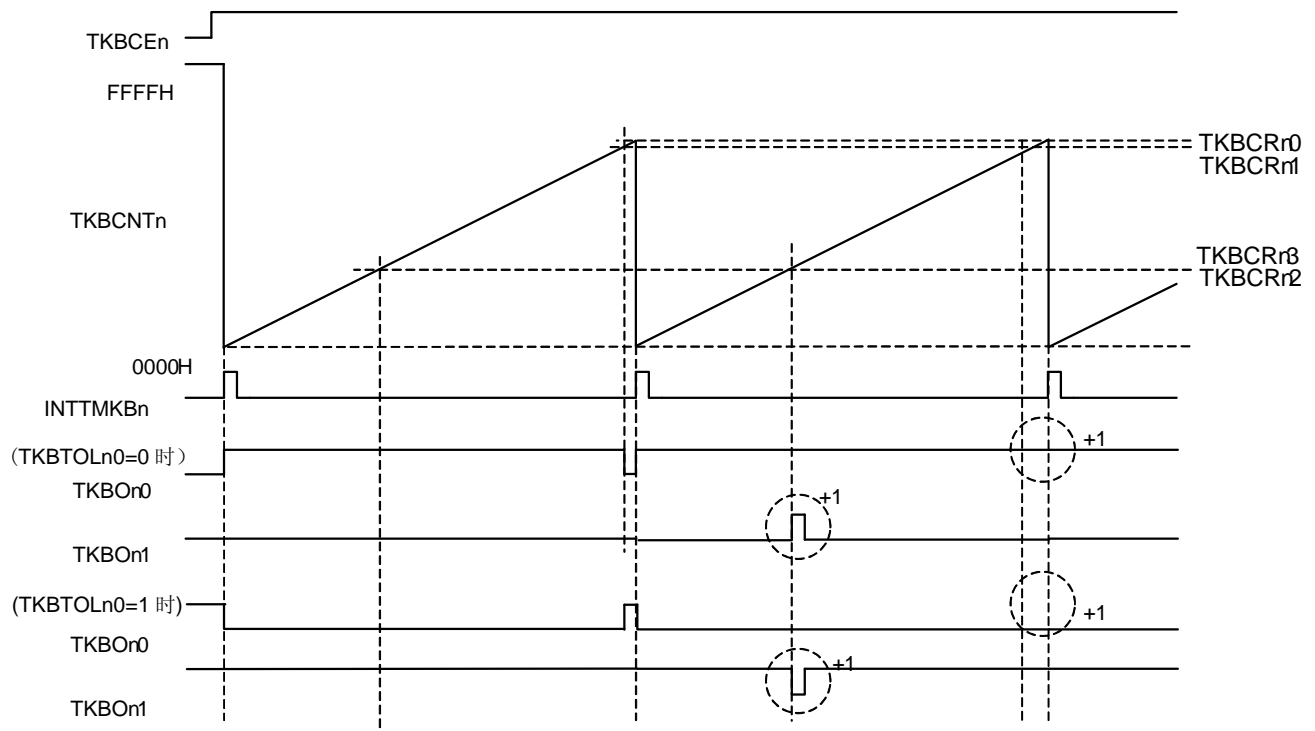
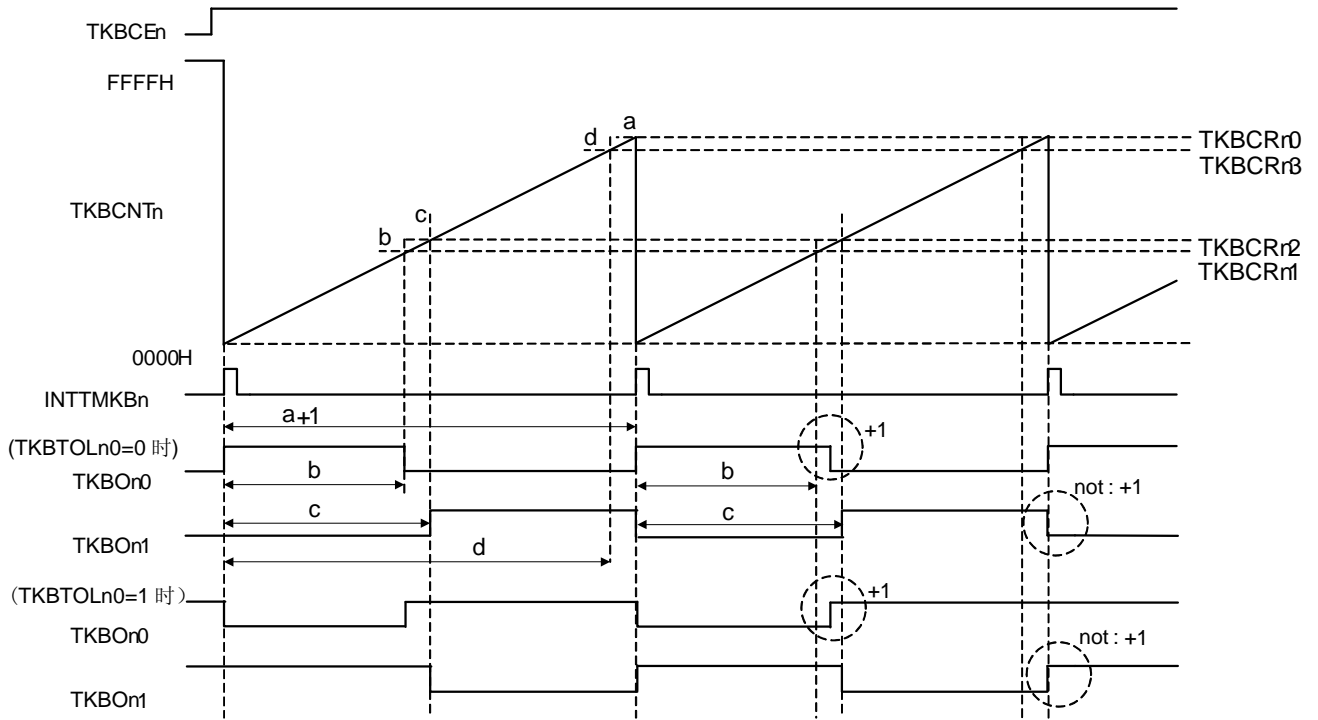


图 7-64. 抖动操作的波形图
(当 $TKBCRn3 = TKBCRn0+1$ 时)



(1) 可使用的操作模式

在 TKBCTLn0 寄存器(TKBSTSn1 和 TKBSTSn0 位)和 TKBCTLn1 寄存器(TKBMDn1 和 TKBMDn0 位)所指定的各种模式下的可否操作的详情如下所示。

操作模式	TKBMDn1, TKBMDn0	TKBSTSn1, TKBSTSn0	可否设定
单机模式(通过TKBCRn0的周期控制)	00	00	○
单机模式(通过外部触发输入的周期控制)	00	01/10/11	×
同时开始/停止模式(通过TKBCRn0的周期控制)	01	00	○
同时开始/停止模式(通过外部触发输入的周期控制)	01	01/10/11	×
同步开始/清除模式(通过主的周期控制)	10	-	○
交错式PFC输出模式	11	-	×

在不使用外部触发输入并且通过 TKBCRn0 控制周期时，可以使用 PWM 输出抖动功能。
TKBDNRn0/TKBDNRn1 分别控制 TKBOn0/TKBOn1 的 PWM 输出抖动功能。

注意事项 1. [TKBDNRn0/TKBDNRn1 寄存器操作(TKBCEn = 1)时进行覆盖]

TKBDNRn0/TKBDNRn1 具有缓冲器，因此可以在操作(TKBCEn = 1)时执行覆盖。

此时，通过向 TKBRDTn 位写入“1”可以执行批量覆盖。

2. [通过 TKBCRLDn0/TKBCRLDn1 寄存器进行存取]

TKBCRLDn0 是映射 TKBCRn1 的低 8 位和 TKBDNRn0 所得的 16 位寄存器。TKBCRLDn1 是映射 TKBCRn3 的低 8 位和 TKBDNRn1 所得的 16 位寄存器。

即使只对 TKBCRLDn0/TKBCRLDn1 寄存器进行存取也能导致 TKBDNRn0/TKBDNRn1 的值改变。

即使只对 TKBCRLDn0/TKBCRLDn1 寄存器进行存取也能导致 TKBCRn1/TKBCRn3 的值改变。

请注意对 TKBCRLDn0/TKBCRLDn1 寄存器进行存取仅能改变 TKBCRn1/TKBCRn3 的低 8 位。

3. [兼用 PWM 输出软启动功能和 PWM 输出抖动功能时]

当执行 PWM 输出软启动功能(TKBSSFnp = 1)时 PWM 输出抖动功能无效。

当 PWM 输出软启动功能停止(TKBSSFnp = 0)时 PWM 输出抖动功能有效。

7.5.3 PWM输出软启动功能

定时器 KB0、KB1 和 KB2 具有 PWM 输出软启动功能，可以控制冲击电流并预防过电压。PWM 输出软启动功能在定时器开始时序处开始工作。用户过去用软件执行的处理可以通过硬件的选项功能来轻松完成。将 16 位定时器 KB 软启动默认占空比寄存器(TKBSIRnp)的设置值作为 1 个周期的有效期间，并产生 PWM 波形。在“16 位定时器 KB 软启动步宽寄存器 (TKBSSRnp)的指定值 + 1 周期”输出同一有效期间的 PWM 波形，然后执行“有效期间 + 1”，并且在“TKBSSRnp + 1 周期”再次输出同一波形。

重复上述操作，直到与 TKBCRn1 和 TKBCRn3 定义的有效周期相同时，解除 PWM 输出软启动功能。

应根据以下条件设置 16 位定时器 KB 软启动初始占空比寄存器：

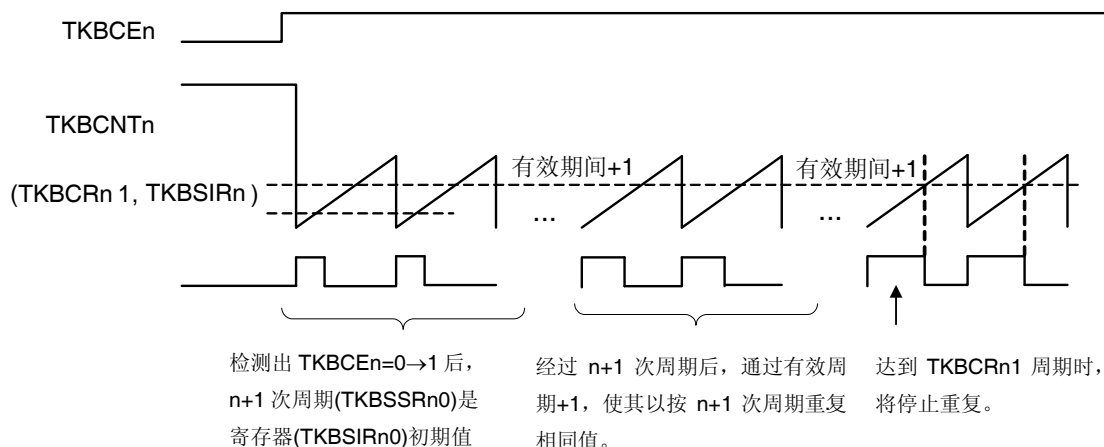
$$0000H \leq TKBSIRn0 < TKBCRn1 \leq TKBCRn0 + 1$$

$$TKBCRn2 \leq TKBSIRn1 < TKBCRn3 \leq TKBCRn0 + 1$$

在使用同步开始/清除模式时，应根据以下条件设置：

$$TKBCRn0 \leq TKBSIRn0 < TKBCRn1 \leq \text{主的 TKBCR00} + 1$$

图 7-65. PWM 输出软启动功能



(1) 可以使用 PWM 输出软启动功能的操作模式

操作模式	TKBMDn1, TKBMDn0	TKBSTSn1, TKBSTSn0	可设置
单机模式(通过TKBCRn0的周期控制)	00B	00B	○
单机模式(通过外部触发输入的周期控制)	00B	01B / 10B / 11B	×
同时开始/停止模式(通过TKBCRn0的周期控制)	01B	00B	○
同时开始/停止模式(通过外部触发输入的周期控制)	01B	01B / 10B / 11B	×
同步开始/清除模式(通过主的周期控制)	10B	-	○
交错式PFC输出模式	11B	-	×

(2) TKBSIRn0/TKBSIRn1/TKBSSRn0/TKBSSRn1 寄存器操作(TKBCEn = 1)时执行覆盖

TKBSIRn0/TKBSIRn1/TKBSSRn0/TKBSSRn1 寄存器操作(TKBCEn = 1)时可以执行覆盖。

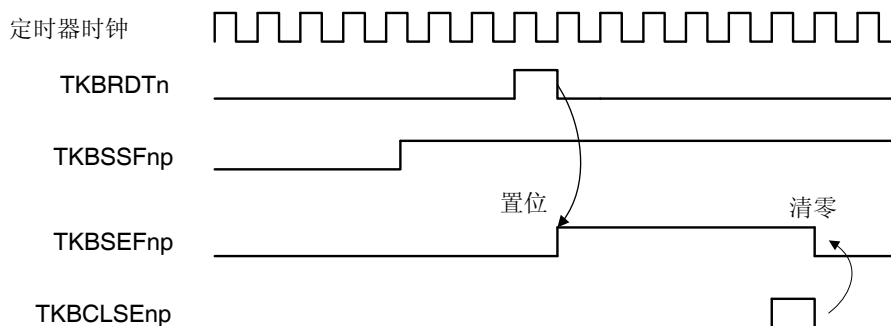
TKBSIRn0/TKBSIRn1/TKBSSRn0/TKBSSRn1 具有缓冲器，因此可以通过向 TKBRDTn 位写入“1”来执行批量覆盖。对于 TKBSIRn0/TKBSIRn1，PWM 输出软启动功能开始时序处的缓冲器值成为占空比的初始值，而 TKBSSRn0/TKBSSRn1 则为内部 4 位计数器的比较值。

内部 4 位计数器将 TKBCNTn 的周期作为计数时钟进行累加，并在与 TKBSSRn0/TKBSSRn1 匹配时成为 0H，然后继续执行计数操作。

(3) TKBCRn0/TKBCRn1/TKBCRn2/TKBCRn3/TKBSIRn0/TKBSIRn1/TKBSSRn0/TKBSSRn1 寄存器操作(TKBCEn = 1)时执行覆盖

如果在 PWM 输出软启动(TKBSSFnp = 1 且 TKBSSFnp = 1)期间将 TKBRDTn 设置为“1”，则批量覆盖被屏蔽而且 TKBSEFnp 标志被设置。要执行批量覆盖，须清除 TKBSEFnp 并确保 TKBSSFnp 成为“0”，将 TKBRDTn 设置为“1”。

图 7-66. TKBCRn0/TKBCRn1/TKBCRn2/TKBCRn3/TKBSIRn0/TKBSIRn1/TKBSSRn0/TKBSSRn1 寄存器的软启动功能操作(TKBSSFnp = 1)时执行覆盖



(4) 兼用 PWM 输出软启动功能和 PWM 输出抖动功能时

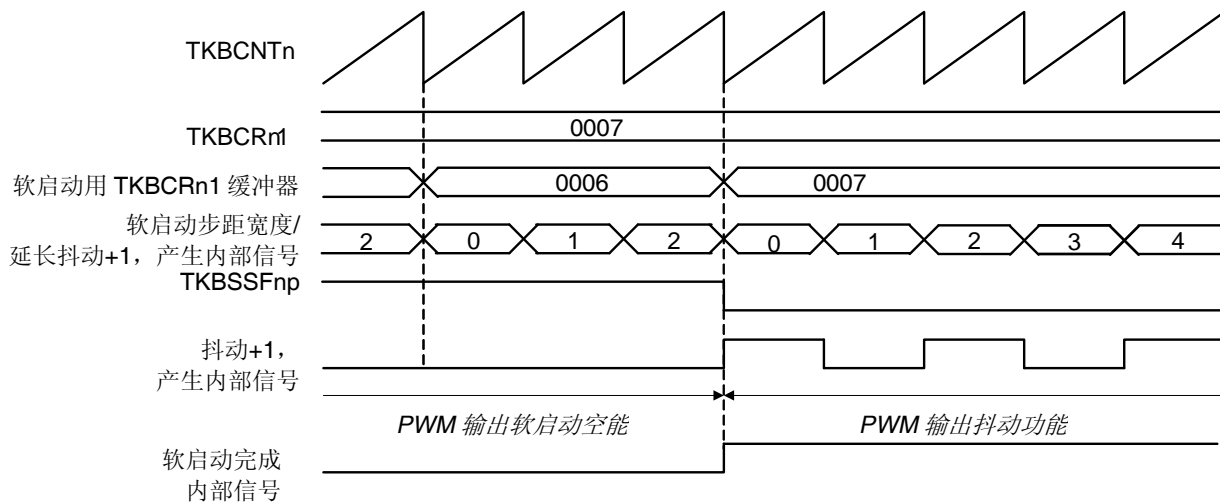
当执行 PWM 输出软启动功能(TKBSSFnp = 1)时 PWM 输出抖动功能无效。

当 PWM 输出软启动功能停止(TKBSSFnp = 0)时 PWM 输出抖动功能有效。

(5) PWM 输出软启动功能的完成和 TKBSSFnp 的操作

图 7-67 显示了当 TKBCRn1 为 0007H、TKBDNRnp 为 70H 和 TKBSSRnp 为 02H 时的图示。在 TKBCRn1 = 0007H 和内部软启动所用 TKBCRn1 缓冲器值匹配的时序，清除 TKBSSFnp，并开始执行抖动功能。

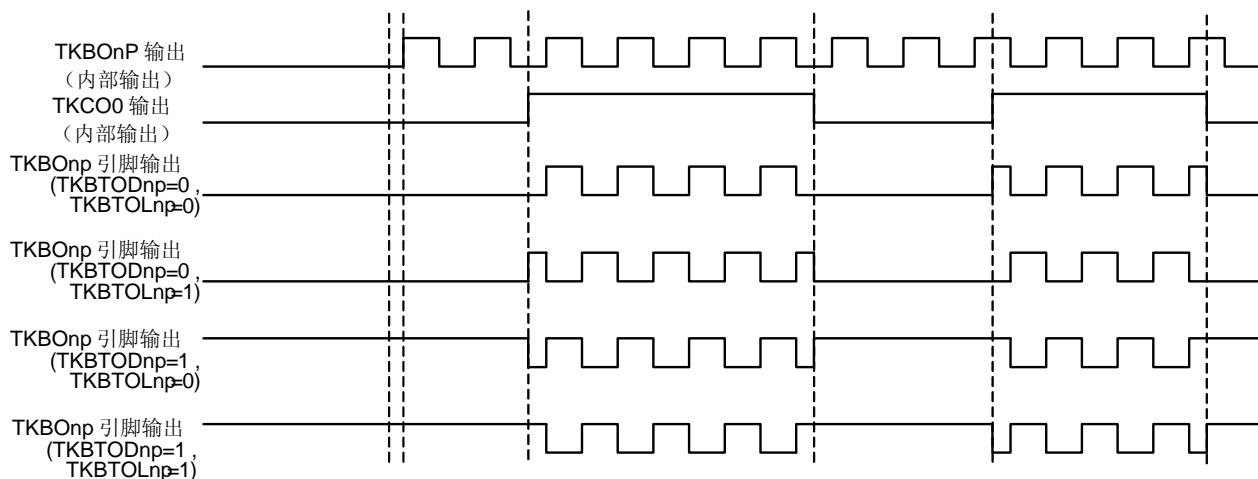
图 7-67. PWM 输出软启动功能的完成和 TKBSSFnp 的操作



7.5.4 PWM输出门控功能 (不兼用PWM输出软启动功能)

利用此功能，在 16 位定时器 KC0 输出(TKCO00 至 TKCO05)为高电平时，从 TKBOnp 的 16 位定时器 KB0 至 KB2 的输出引脚输出 PWM 脉冲。在 16 位定时器 KC0 输出(TKCO00 至 TKCO05)为低电平时，从 TKBOnp 的 16 位定时器 KB0 至 KB2 的输出引脚输出默认电平。

图 7-68. PWM 输出门控功能



如下所示，TKCO0q 门控输出与要进行门控的 TKBOnp 输出之间是一一对应的关系：

TKCO00 : TKBO00
 TKCO01 : TKBO01
 TKCO02 : TKBO10
 TKCO03 : TKBO11
 TKCO04 : TKBO20
 TKCO05 : TKBO21

(1) 可以使用 PMW 输出门控功能的操作模式

在以下操作模式时可以使用输出门控功能。

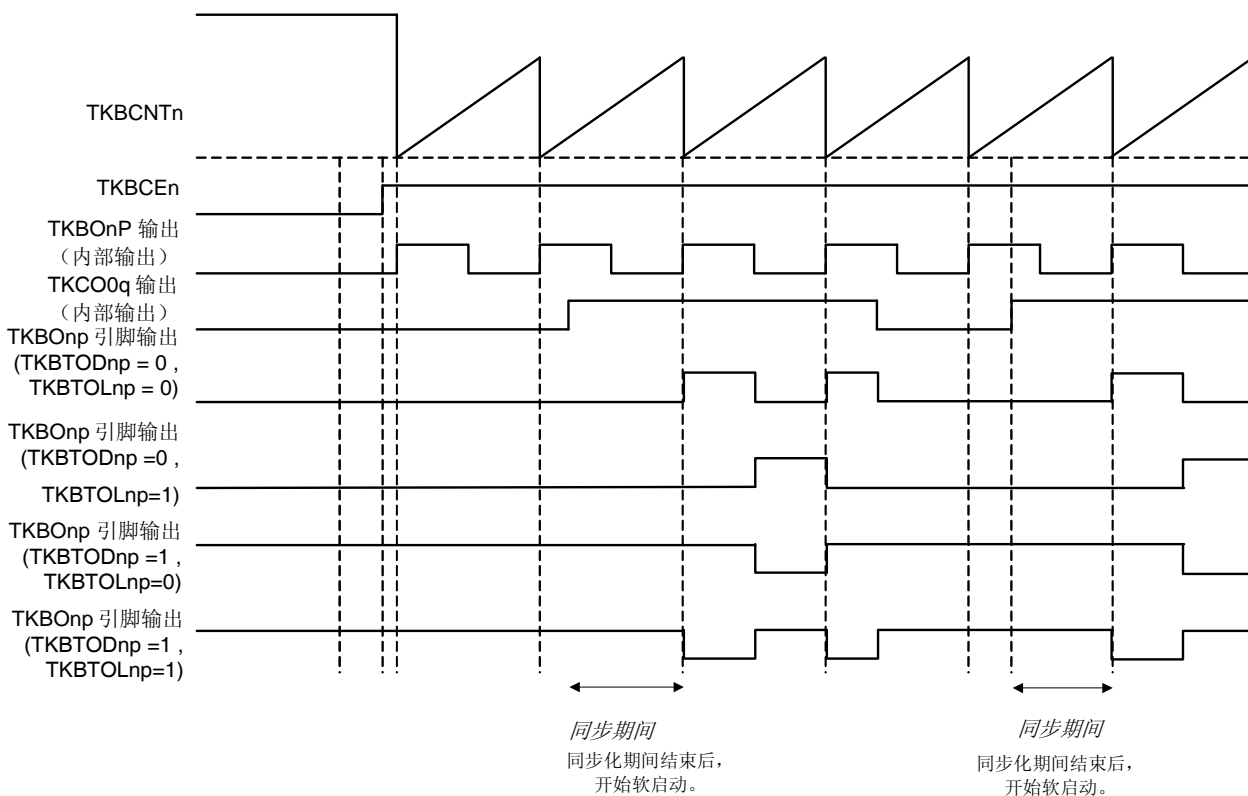
操作模式	TKBMDn1, TKBMDn0	TKBSTSn1, TKBSTSn0	可设置
单机模式(通过TKBCRn0的周期控制)	00B	00B	○
单机模式(通过外部触发输入的周期控制)	00B	01B / 10B / 11B	×
同时开始/停止模式(通过TKBCRn0的周期控制)	01B	00B	○
同时开始/停止模式(通过外部触发输入的周期控制)	01B	01B / 10B / 11B	×
同步开始/清除模式(通过主的周期控制)	10B	—	○
交错式PFC输出模式	11B	—	×

7.5.5 PWM输出门控功能(可兼用PWM输出软启动功能)

可以兼用 PWM 输出门控功能和 PWM 输出软启动功能。

同时使用软启动时，在检测到 16 位定时器 KC 的定时器 KC0 输出(TKCO00 至 TKCO05)的上升沿后，与 16 位定时器 KB0 至 KB2 的周期同步地从 16 位定时器 KB0 至 KB2 的 TKBOnp 输出引脚输出 PWM 脉冲。在检测到 16 位定时器 KC 的定时器 KC0 输出(TKCO00 至 TKCO05)的下降沿后，从 TKBOnp 的 16 位定时器 KB0 至 KB2 的输出引脚输出默认电平 (TKBTODnp)。

图 7-69. PWM 输出门控功能(兼用 PWM 输出软启动功能)的 TKCO0q 输出开始时的 TKBOnp 输出同步波形示例



(1) 可以使用 PMW 输出门控功能的操作模式

在以下操作模式时可以使用输出门控功能。

操作模式	TKBMDn1, TKBMDn0	TKBSTSn1, TKBSTSn0	可设置
单机模式(通过TKBCRn0的周期控制)	00B	00B	○
单机模式(通过外部触发输入的周期控制)	00B	01B/10B/11B	×
同时开始/停止模式(通过TKBCRn0的周期控制)	01B	00B	○
同时开始/停止模式(通过外部触发输入的周期控制)	01B	01B/10B/11B	×
同步开始/清除模式(通过主的周期控制)	10B	–	○
交错式PFC输出模式	11B	–	×

关于 PWM 输出软启动功能的详情，请参阅 7.5.5 PWM 输出门控功能(可兼用 PWM 输出软启动功能)。

7.5.6 最大频率限值功能

在通过外部触发控制周期或交错式 PFC 输出模式时，定时器 KB0、KB1 和 KB2 具有限制计数器清除的最小周期（最大频率）的功能。

使用该功能时，如果在计数器值小于最大频率限值寄存器(TKBMFRn)的设置值时产生了用于清除计数器的外部触发输入，则保留该输入且继续计数，直至达到 TKBMFRn 的设置值时执行计数器清除。

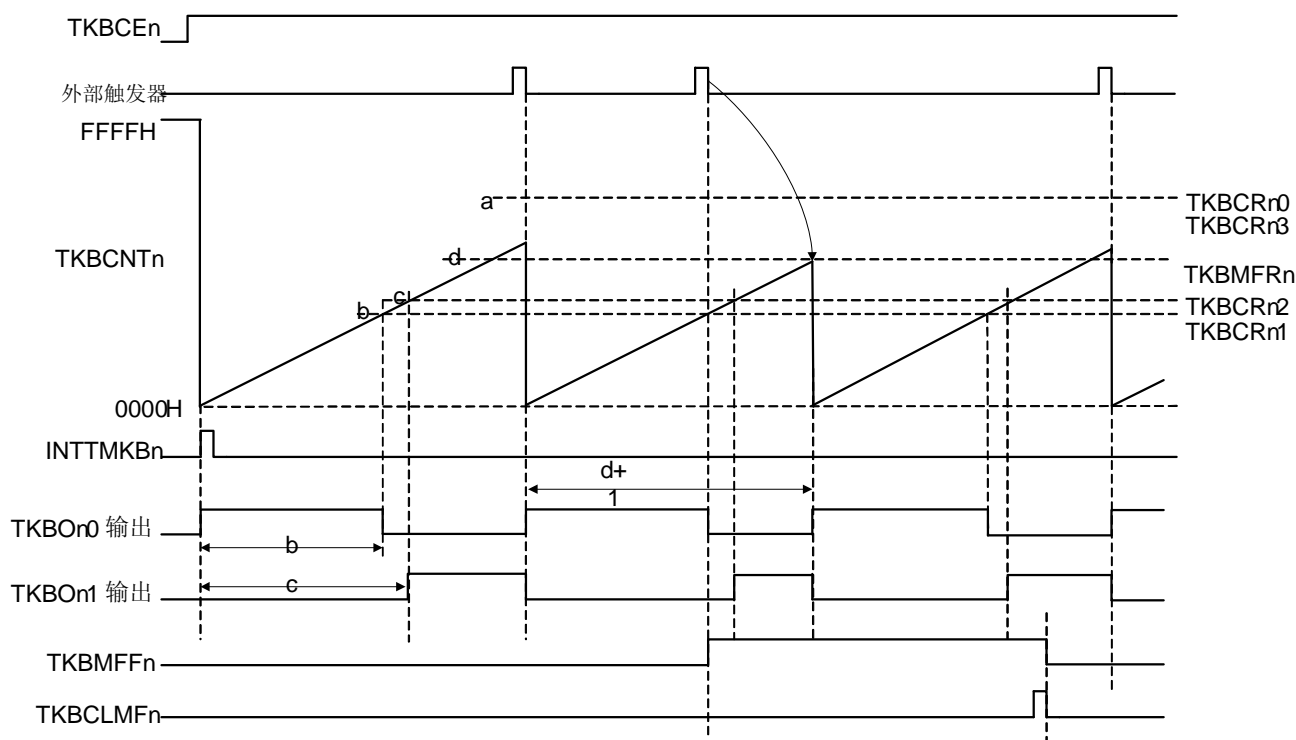
(1) 最大频率限值 (= 1/最小周期)的计算公式

最小周期 (= 1/最大频率限值) = (TKBMFRn 的设置值 + 1) × 计数时钟周期

注意事项 须满足以下条件：TKBMFRn 的设置值 ≤ TKBCRn0 的设置值

在检测到外部触发输入的时序处，如果计数器值小于 TKBMFRn，则将 TKBMFFn 标志设置为“1”。通过向 TKBCLMFn 位写入“1”，使 TKBMFFn 标志被清除为“0”。

图 7-70. 最大频率限值功能



备注 通过外部触发输入的周期控制时。

(2) 可以使用最大频率限值功能的操作模式

操作模式	TKBMDn1, TKBMDn0	TKBSTSn1, TKBSTSn0	可设置
单机模式(通过TKBCRn0的周期控制)	00B	00B	×
单机模式(通过外部触发输入的周期控制)	00B	01B/10B/11B	○
同时开始/停止模式(通过TKBCRn0的周期控制)	01B	00B	×
同时开始/停止模式(通过外部触发输入的周期控制)	01B	01B/10B/11B	○
同步开始/清除模式(通过主的周期控制)	10B	—	×
交错式PFC输出模式	11B	—	○

备注 通过外部触发输入的周期控制时可以使用。

7.6 强制输出停止功能

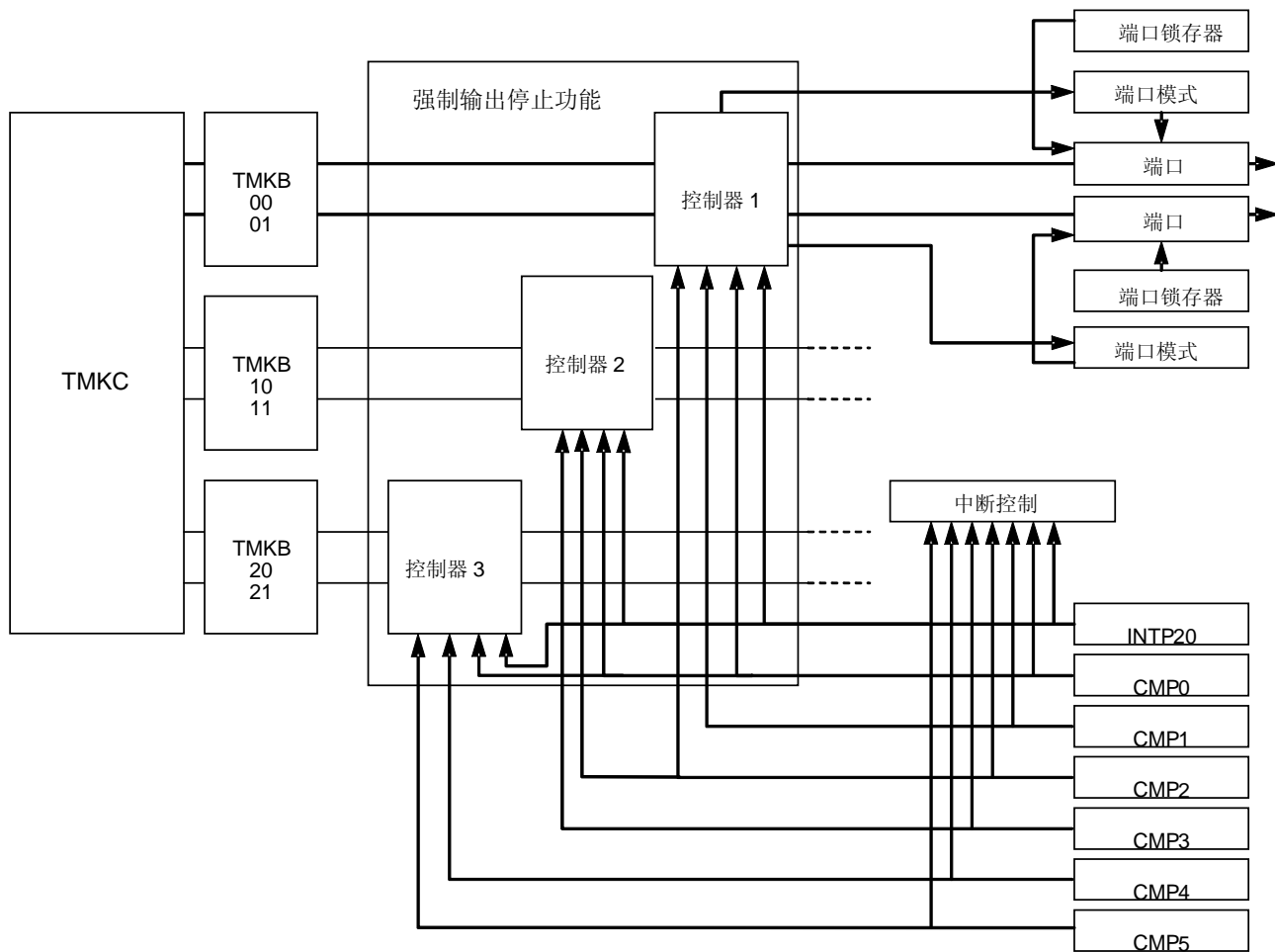
强制输出停止功能是为了保护电源电路等的功能。

如果微型计算机外面配置的电路中出现任何异常情况，导致过电压或过电流，则将电压或电流感测信号导入 INTP20/比较器可对电路进行保护，其原理是保持定时器输出高阻抗或固定为输出状态，而不用 CPU 的程序控制充当媒介。

利用此功能，只有在检测到输入信号边沿时才会识别异常。而没有边沿的固定电平不会被识别为异常。

强制输出停止功能的系统结构如下图所示。

图 7-71. 强制输出停止功能的系统结构



7.6.1 强制输出停止功能 1 和强制输出停止功能 2

强制输出停止功能具有两种控制方法。强制输出停止功能 1 可以选择固定电平输出或高阻抗输出，强制输出停止功能 2 仅能设置固定电平输出。控制方法的区别如下所示。

(1) 强制输出停止功能 1/强制输出停止功能 2 的可选择的输出电平

可选择的输出电平	强制输出停止	
	功能1	功能2
高阻抗输出	○	×
低电平固定输出	○	○
高电平固定输出	○	○

(2) 强制输出停止功能 1/强制输出停止功能 2 的开始/清除条件

功能/操作说明 (开始强制输出停止)	强制输出停止	
	功能1	功能2
检测到比较器输出的上升沿后，开始强制输出停止功能。	○	○
检测到外部中断输入(INTP20)的上升沿后，开始强制输出停止功能。	×	○
通过软件位(TKBAHTSn _p)设置，开始强制输出停止功能。	○	×

功能/操作说明 (清除强制输出停止)	强制输出停止	
	功能1	功能2
通过软件位(TKBAHTTn _{pp})设置，清除强制输出停止功能。	○	×
通过设置软件位(TKBAHTTn _p)后与TMKB周期同步，清除强制输出停止功能。	○	×
开始强制输出停止功能后，在下一计数器周期清除强制输出停止功能。	×	○
检测到强制输出停止功能触发信号的下降沿后，在下一计数器周期清除强制输出停止功能。	×	○

备注 n = 0 至 2, p = 0, 1

(3) 强制输出停止功能 1/强制输出停止功能 2 的可选择的触发信号和可使用的触发位的条件

可选择的触发信号	强制输出停止	
	功能1	功能2
比较器0至5	○	○
外部中断输入(INTP20)	×	○

可使用的触发位	强制输出停止	
	功能1	功能2
TKBPAHTSn _p (启动TKBOnp输出的强制输出停止功能的触发位)	○	×
TKBPAHTTn _p (清除TKBOnp输出的强制输出停止功能的触发位)	○	×

备注 n = 0 至 2, p = 0, 1

表 7-4. 强制输出停止功能 1 的外部触发分配列表

	TKBO00	TKBO01	TKBO10	TKBO11	TKBO20	TKBO21
比较器0	√	√	√	√	√	√
比较器1	√	√	-		-	-
比较器2	√	√	√	√	-	-
比较器3	-	-	√	√	-	-
比较器4	-	-	-	-	√	√
比较器5	-	-	-	-	√	√
INTP20	-	-	-	-	-	-
INTP21	-	-	-	-	-	-

表 7-5. 强制输出停止功能 2 的外部触发分配列表

	TKBO00	TKBO01	TKBO10	TKBO11	TKBO20	TKBO21
比较器0	√	√	√	√	√	√
比较器1	√	√	-	-	-	-
比较器2	√	√	√	√	-	-
比较器3	-	-	√	√	-	-
比较器4	-	-	-	-	√	√
比较器5	-	-	-	-	√	√
INTP20	√	√	√	√	√	√
INTP21	-	-	-	-	-	-

7.6.2 强制输出停止功能的配置

强制输出停止功能由如下的硬件构成。

表 7-6. 强制输出停止功能的配置

项目	配置
控制寄存器	外围允许寄存器 2 (PER2) 强制输出停止功能控制寄存器 n0 (TKBPACTLn0) 强制输出停止功能控制寄存器 n1 (TKBPACTLn1) 强制输出停止功能控制寄存器 n2 (TKBPACTLn2) 强制输出停止功能标志寄存器 n (TKBPAFLGn) 强制输出停止功能 1 开始触发寄存器 n (TKBPAHFSn) 强制输出停止功能清除触发寄存器 n (TKBPAHFTn)

7.6.3 控制强制输出停止功能的寄存器

控制强制输出停止功能的寄存器如下所示。

- 外围允许寄存器 2 (PER2)
- 强制输出停止功能控制寄存器 n0 (TKBPACTLn0)
- 强制输出停止功能控制寄存器 n1 (TKBPACTLn1)
- 强制输出停止功能控制寄存器 n2 (TKBPACTLn2)
- 强制输出停止功能标志寄存器 n (TKBPAFLGn)
- 强制输出停止功能 1 开始触发寄存器 n (TKBPAHFSn)
- 强制输出停止功能清除触发寄存器 n (TKBPAHFTn)

(1) 外围允许寄存器 2 (PER2)

该寄存器用于允许或禁止供应时钟至外围硬件。并且中止向未使用的硬件供应时钟，以减少电力消耗和噪声。当使用强制输出停止功能时，请务必将位 6 至 4 (TKBPA2EN 至 TKBPA0EN) 设置为 1。

使用 8 位存储器操作指令来设置 PER2 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 7-72. 外围允许寄存器 2 (PER2)的格式

地址: F0509H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
PER2	PGACMPEN	TKBPA2EN	TKBPA1EN	TKBPA0EN	TKC0EN	TKB2EN	TKB1EN	TKB0EN

TKBPAnEN	控制向定时器KBn的强制输出停止功能的输入时钟
0	停止输入时钟的供应。 <ul style="list-style-type: none"> • 不可写入用于定时器KBn的强制输出停止功能的SFR。 • 定时器KBn的强制输出停止功能处于初始状态。
1	供应输入时钟。 <ul style="list-style-type: none"> • 可以读取/写入写入用于定时器KBn的强制输出停止功能的SFR。

注意事项 要设置定时器 KBn 的强制输出停止功能，务必首先将 TKBPAnEN 位设置为“1”。如果 TKBPAnEN = 0，对于定时器 KBn 强制输出停止功能的控制寄存器的写入被忽略，而且读取值总为初始值。

备注 n = 0 至 2

(2) 强制输出停止功能控制寄存器 n0, n1 (TKBPACTLn0, TKBPACTLn1)

TKBPACTLn_p 寄存器选择用于控制强制输出停止功能的触发信号，并选择用于设置强制输出停止模式的引脚。

使用 16 位存储器操作指令设置 TKBPACTLn_p 寄存器。

产生复位信号后，该寄存器被清除为 0000H。

图 7-73. 强制输出停止功能控制寄存器 0p (TKBPACTL0p)的格式 (1/2)

地址: F0630H (TKBPACTL00), F0632H (TKBPACTL01) 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8
TKBPACTL0p	TKBPAFXS0p3	TKBPAFXS0p2	TKBPAFXS0p1	TKBPAFXS0p0	0	0	0	TKBPAFCM0p
	7	6	5	4	3	2	1	0
	0	TKBPAHZS0p2	TKBPAHZS0p1	TKBPAHZS0p0	TKBPAHCM0p1	TKBPAHCM0p0	TKBPAMD0p1	TKBPAMD0p0

TKBPAFXS0p3	选择强制输出停止功能2的外部中断触发
0	不能将INTP20作为触发。
1	可将INTP20作为触发。

TKBPAFXS0p2	选择强制输出停止功能2的比较器触发
0	不能将比较器2作为触发。
1	可将比较器2作为触发。

TKBPAFXS0p1	选择强制输出停止功能2的比较器触发
0	不能将比较器1作为触发。
1	可将比较器1作为触发。

TKBPAFXS0p0	选择强制输出停止功能2的比较器触发
0	不能将比较器0作为触发。
1	可将比较器0作为触发。

TKBPAFCM0p	选择强制输出停止功能2的操作模式
0	输入触发时启动强制输出停止功能 2，并在下一个计数器周期清除该功能。
1	输入触发时启动强制输出停止功能 2，并在检测到触发反相沿的下一个计数器周期清除该功能。

图 7-73. 强制输出停止功能控制寄存器 0p (TKBPACTL0p)的格式 (2/2)

TKBPAHVS0p2	选择强制输出停止功能1的比较器触发
0	不能将比较器2作为触发。
1	可将比较器2作为触发。

TKBPAHVS0p1	选择强制输出停止功能1的比较器触发
0	不能将比较器1作为触发。
1	可将比较器1作为触发。

TKBPAHVS0p0	选择强制输出停止功能1的比较器触发
0	不能将比较器0作为触发。
1	可将比较器0作为触发。

TKBPAHCM0p1	TKBPAHCM0p0	选择强制输出停止功能1的清除条件
0	0	输入触发时启动强制输出停止功能1，并在写入Hi-Z 停止触发(TKBPAHTT0) = 1时清除该功能（无论触发信号电平如何）。
0	1	输入触发时启动强制输出停止功能1，当触发信号为有效时，写入“Hi-Z 停止触发(TKBPAHTT0) = 1”的操作无效。当触发信号为无效时，写入“Hi-Z 停止触发(TKBPAHTT0) = 1”将清除强制输出停止功能1。
1	0	输入触发时启动强制输出停止功能1，并在写入Hi-Z 停止触发(TKBPAHTT0) = 1后的下一计数器周期清除该功能（无论触发信号电平如何）。
1	1	输入触发时启动强制输出停止功能1，当触发信号为有效时，写入“Hi-Z 停止触发(TKBPAHTT0) = 1”的操作无效。当触发信号为无效时，写入“Hi-Z 停止触发(TKBPAHTT0) = 1”后的下一计数器周期将清除强制输出停止功能1。

TKBPAMD0p1	TKBPAMD0p0	选择执行强制输出停止功能时的输出状态	
		强制输出停止功能1	强制输出停止功能2
0	0	Hi-Z输出	低电平固定输出
0	1	Hi-Z输出	高电平固定输出
1	0	低电平固定输出	低电平固定输出
1	1	高电平固定输出	高电平固定输出

- 注意事项 1. 在定时器操作期间，禁止设置 TKBPACTL0p 寄存器的其它位。不过，可以刷新 TKBPACTL0p 寄存器（写入相同的值）。
2. 务必将位 11 至 9 以及 7 清除为“0”。

备注 n = 0 至 2, p = 0, 1

图 7-74. 强制输出停止功能控制寄存器 1p (TKBPACTL1p)的格式 (1/2)

地址: F0670H (TKBPACTL10), F0672H (TKBPACTL11) 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8
TKBPACTL1p	TKBPAFXS1p3	TKBPAFXS1p2	TKBPAFXS1p1	TKBPAFXS1p0	0	0	0	TKBPAFCM1p
	7	6	5	4	3	2	1	0
	0	TKBPAHZS1p2	TKBPAHZS1p1	TKBPAHZS1p0	TKBPAHCM1p1	TKBPAHCM1p0	TKBPAMD1p1	TKBPAMD1p0

TKBPAFXS1p3	选择强制输出停止功能2的外部中断触发
0	不能将INTP20作为触发。
1	可将INTP20作为触发。

TKBPAFXS1p2	选择强制输出停止功能2的比较器触发
0	不能将比较器3作为触发。
1	可将比较器3作为触发。

TKBPAFXS1p1	选择强制输出停止功能2的比较器触发
0	不能将比较器2作为触发。
1	可将比较器2作为触发。

TKBPAFXS1p0	选择强制输出停止功能2的比较器触发
0	不能将比较器0作为触发。
1	可将比较器0作为触发。

TKBPAFCM1p	选择强制输出停止功能2的操作模式
0	输入触发时启动强制输出停止功能 2，并在下一个计数器周期清除该功能。
1	输入触发时启动强制输出停止功能 2，并在检测到触发反相沿的下一个计数器周期清除该功能。

图 7-74. 强制输出停止功能控制寄存器 2p (TKBPACTL1p)的格式 (2/2)

TKBPAHVS1p2	选择强制输出停止功能1的比较器触发
0	不能将比较器3作为触发。
1	可将比较器3作为触发。

TKBPAHVS1p1	选择强制输出停止功能1的比较器触发
0	不能将比较器2作为触发。
1	可将比较器2作为触发。

TKBPAHVS1p0	选择强制输出停止功能1的比较器触发
0	不能将比较器0作为触发。
1	可将比较器0作为触发。

TKBPAHCM1p1	TKBPAHCM1p0	选择强制输出停止功能1的清除条件
0	0	输入触发时启动强制输出停止功能1，并在写入Hi-Z 停止触发(TKBPAHTT1) = 1时清除该功能（无论触发信号电平如何）。
0	1	输入触发时启动强制输出停止功能1，当触发信号为有效时，写入“Hi-Z 停止触发(TKBPAHTT1) = 1”的操作无效。当触发信号为无效时，写入“Hi-Z 停止触发(TKBPAHTT1) = 1”将清除强制输出停止功能1。
1	0	输入触发时启动强制输出停止功能1，并在写入Hi-Z 停止触发(TKBPAHTT1) = 1后的下一计数器周期清除该功能（无论触发信号电平如何）。
1	1	输入触发时启动强制输出停止功能1，当触发信号为有效时，写入“Hi-Z 停止触发(TKBPAHTT1) = 1”的操作无效。当触发信号为无效时，写入“Hi-Z 停止触发(TKBPAHTT1) = 1”后的下一计数器周期将清除强制输出停止功能1。

TKBPAMD1p1	TKBPAMD1p0	选择执行强制输出停止功能时的输出状态	
		强制输出停止功能1	强制输出停止功能2
0	0	Hi-Z输出	低电平固定输出
0	1	Hi-Z输出	高电平固定输出
1	0	低电平固定输出	低电平固定输出
1	1	高电平固定输出	高电平固定输出

- 注意事项 1. 在定时器操作期间，禁止设置 TKBPACTL1p 寄存器的其它位。不过，可以刷新 TKBPACTL1p 寄存器（写入相同的值）。
2. 务必将位 11 至 9 以及 7 清除为“0”。

备注 n = 0 至 2, p = 0, 1

图 7-75. 强制输出停止功能控制寄存器 2p (TKBPACTL2p)的格式 (1/2)

地址: F06B0H (TKBPACTL20), F06B2H (TKBPACTL21) 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8
TKBPACTL2p	TKBPAFXS2p3	TKBPAFXS2p2	TKBPAFXS2p1	TKBPAFXS2p0	0	0	0	TKBPAFCM2p
	7	6	5	4	3	2	1	0
	0	TKBPAHZS2p2	TKBPAHZS2p1	TKBPAHZS2p0	TKBPAHCM2p1	TKBPAHCM2p0	TKBPAMD2p1	TKBPAMD2p0

TKBPAFXS2p3	选择强制输出停止功能2的外部中断触发
0	不能将INTP20作为触发。
1	可将INTP20作为触发。

TKBPAFXS2p2	选择强制输出停止功能2的比较器触发
0	不能将比较器5作为触发。
1	可将比较器5作为触发。

TKBPAFXS2p1	选择强制输出停止功能2的比较器触发
0	不能将比较器3作为触发。
1	可将比较器3作为触发。

TKBPAFXS2p0	选择强制输出停止功能2的比较器触发
0	不能将比较器0作为触发。
1	可将比较器0作为触发。

TKBPAFCM2p	选择强制输出停止功能2的操作模式
0	输入触发时启动强制输出停止功能 2, 并在下一个计数器周期清除该功能。
1	输入触发时启动强制输出停止功能 2, 并在检测到触发反相沿的下一个计数器周期清除该功能。

图 7-75. 强制输出停止功能控制寄存器 2p (TKBPACTL2p)的格式 (2/2)

TKBPAHVS2p2	选择强制输出停止功能1的比较器触发
0	不能将比较器5作为触发。
1	可将比较器5作为触发。

TKBPAHVS2p1	选择强制输出停止功能1的比较器触发
0	不能将比较器4作为触发。
1	可将比较器4作为触发。

TKBPAHVS2p0	选择强制输出停止功能1的比较器触发
0	不能将比较器0作为触发。
1	可将比较器0作为触发。

TKBPAHCM2p1	TKBPAHCM2p0	选择强制输出停止功能1的清除条件
0	0	输入触发时启动强制输出停止功能1，并在写入Hi-Z 停止触发(TKBPAHTT2) = 1时清除该功能（无论触发信号电平如何）。
0	1	输入触发时启动强制输出停止功能1，当触发信号为有效时，写入“Hi-Z 停止触发(TKBPAHTT2) = 1”的操作无效。当触发信号为无效时，写入“Hi-Z 停止触发(TKBPAHTT2) = 1”将清除强制输出停止功能1。
1	0	输入触发时启动强制输出停止功能1，并在写入Hi-Z 停止触发(TKBPAHTT2) = 1后的下一计数器周期清除该功能（无论触发信号电平如何）。
1	1	输入触发时启动强制输出停止功能1，当触发信号为有效时，写入“Hi-Z 停止触发(TKBPAHTT2) = 1”的操作无效。当触发信号为无效时，写入“Hi-Z 停止触发(TKBPAHTT2) = 1”后的下一计数器周期将清除强制输出停止功能1。

TKBPAMD2p1	TKBPAMD2p0	选择执行强制输出停止功能时的输出状态	
		强制输出停止功能1	强制输出停止功能2
0	0	Hi-Z输出	低电平固定输出
0	1	Hi-Z输出	高电平固定输出
1	0	低电平固定输出	低电平固定输出
1	1	高电平固定输出	高电平固定输出

- 注意事项 1. 在定时器操作期间，禁止设置 TKBPACTL2p 寄存器的其它位。不过，可以刷新 TKBPACTL2p 寄存器（写入相同的值）。
2. 务必将位 11 至 9 以及 7 清除为“0”。

备注 n = 0 至 2, p = 0, 1

(3) 强制输出停止功能控制寄存器 n2 (TKBPACTLn2)

TKBPACTLn2 是用来允许/禁止强制输出停止功能的寄存器。

使用 1 位或 8 位存储器操作指令来设置 TKBPACTLn2 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 7-76. 强制输出停止功能控制寄存器 n2 (TKBPACTLn2)的格式

地址: F0637H (TKBPACTL02), F0677H (TKBPACTL02), F06B7H (TKBPACTL02)

复位后: 00H R/W

符号

	7	6	5	4	3	2	<1>	<0>
TKBPACTLn2	0	0	0	0	0	0	TKBPACEn1	TKBPACEn0

TKBPACEnp	控制强制输出停止功能所使用的触发信号的输入
0	禁止强制输出停止功能操作
1	允许强制输出停止功能操作

- 注意事项**
1. 定时器操作期间，可以覆盖 TKBPACTLn2 寄存器。
 2. 务必将位 7 至 2 清除为“0”。

备注 n = 0 至 2, p = 0, 1

(4) 强制输出停止功能标志寄存器(TKBPAFLGn)

TKBPAFLGn 是用来显示强制输出停止功能的状态标志的寄存器。

使用 1 位或 8 位存储器操作指令来读取 TKBPAFLGn 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 7-77. 强制输出停止功能标志寄存器(TKBPAFLGn)的格式

地址: F0636H (TKBPAFLG0), F0676H (TKBPAFLG1), F06B6H (TKBPAFLG2) 复位后: 00H R

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
TKBPAFLGn	TKBPAFSFn1	TKBPAHSFn1	TKBPAFSFn0	TKBPAHSFn0	TKBPAFIFn1	TKBPAHIFn1	TKBPAFIFn0	TKBPAHIFn0

TKBPAFSFn _p	针对TKBOnp引脚的强制输出停止功能2的状态标志
0	强制输出停止清除状态
1	强制输出停止状态

TKBPAHSFn _p	针对TKBOnp引脚的强制输出停止功能1的状态标志
0	强制输出停止清除状态
1	强制输出停止状态

TKBPAFIFn _p	针对TKBOnp引脚的强制输出停止功能2的输入监视器位
0	强制输出停止功能2触发信号为低电平 (无效)
1	强制输出停止功能2触发信号为高电平 (有效)

TKBPAHIFn _p	针对TKBOnp引脚的强制输出停止功能1的输入监视器位
0	强制输出停止功能1触发信号为低电平 (无效)
1	强制输出停止功能1触发信号为高电平 (有效)

备注 n = 0 至 2, p = 0, 1

(5) 强制输出停止功能 1 开始触发寄存器 n (TKBPAHFSn)

TKBPAHFSn 是用于强制输出停止功能 1 的开始触发寄存器。
使用 1 位或 8 位存储器操作指令来写入 TKBPAHFSn 寄存器。
产生复位信号后，该寄存器被清除为 00H。

图 7-78. 强制输出停止功能 1 开始触发寄存器 n (TKBPAHFSn)的格式

地址: F0634H (TKBPAHFS0), F0674H (TKBPAHFS1), F06B4H (TKBPAHFS2)
复位后: 00H R/W

符号	7	6	5	4	3	2	<1>	<0>
TKBPAHFSn	0	0	0	0	0	0	TKBPAHTSn1	TKBPAHTSn0

TKBPAHTSn _p	针对TKBOnp输出的强制输出停止功能1的开始触发
0	设置无效
1	开始针对TKBOnp输出的强制输出停止功能1

- 注意事项**
1. 定时器操作期间，可以覆盖 TKBPAHFSn 寄存器。
 2. 务必将位 7 至 2 清除为“0”。
 3. TKBPAHFSn 寄存器的读取值为 0。

(6) 强制输出停止功能清除触发寄存器 n (TKBPAHFTn)

TKBPAHFTn 是用于强制输出停止功能 1 的清除触发寄存器。
使用 1 位或 8 位存储器操作指令来写入 TKBPAHFTn 寄存器。
产生复位信号后，该寄存器被清除为 00H。

图 7-79. 强制输出停止功能清除触发寄存器 n (TKBPAHFTn)的格式

地址: F0635H (TKBPAHFT0), F0675H (TKBPAHFT1), F06B5H (TKBPAHFT2)
复位后: 00H R/W

符号	7	6	5	4	3	2	<1>	<0>
TKBPAHFTn	0	0	0	0	0	0	TKBPAHTTn1	TKBPAHTTn0

TKBPAHTTn _p	针对 TKBOnp 输出的强制输出停止功能的清除触发
0	设置无效
1	清除针对 TKBOnp 输出的强制输出停止功能 1

- 注意事项**
1. 定时器操作期间，可以覆盖 TKBPAHFTn 寄存器。
 2. 务必将位 7 至 2 清除为“0”。
 3. TKBPAHFSn 寄存器的读取值为 0。

备注 n = 0 至 2, p = 0, 1

7.7 强制输出停止功能 1 的操作

当出现触发源（比较器 0 至 5 输出）时，可以与 16 位定时器 KB_n 和 KC_0 电路的操作时钟 f_{KBKC} 异步地直接将定时器输出固定为 Hi-Z 电平、高电平或低电平（不通过 CPU）。通过设置强制输出停止功能 1 的停止触发器，可以与 16 位定时器 KB_n 和 KC_0 电路的操作时钟 f_{KBKC} 同步地取消强制输出停止状态。

7.7.1 强制输出停止功能 1 的概要

在该功能中，使用比较器输出信号和软件触发作为强制输出停止功能 1 的触发信号。

强制输出停止功能可以选择的输出电平由 $TKBPACTL_{np}$ 寄存器的 $TKBPAMD_{np0}$ 和 $TKBPAMD_{np1}$ 位控制。

下表显示了其与定时器 KB_n 的输出 p 引脚($TKBOn_p$)的强制输出停止功能 1 之间的关系。

$TKBPAMD_{np1}$	$TKBPAMD_{np0}$	选择执行强制输出停止功能 1 时的输出电平
0	0	Hi-Z 输出
0	1	Hi-Z 输出
1	0	低电平固定输出
1	1	高电平固定输出

所用比较器输出的选择由强制输出停止功能控制寄存器 np ($TKBPACTL_{np}$)的 $TKBPAHZS_{np2}$ 至 $TKBPAHZS_{np0}$ 位控制。

下表显示了定时器 KB_n 的输出 p 引脚($TKBOn_p$)的强制输出停止功能 1 的触发选择。

位	可选择的触发信号		
	定时器KB0	定时器KB1	定时器KB2
$TKBPAHZS_{np0}$	比较器0		
$TKBPAHZS_{np1}$	比较器1	比较器2	比较器4
$TKBPAHZS_{np2}$	比较器2	比较器3	比较器5

备注 $n = 0$ 至 2 , $p = 0, 1$

7.7.2 强制输出停止功能 1 的软件清除操作

开始强制输出停止功能 1 的开始触发(TKBAHFSn 寄存器的 TKBAHTSnp 位)的设置如下表所示。

表 7-7. 强制输出停止功能 1 的开始触发(TKBAHTSnp 位)的操作

TKBAHTSnp	通过软件开始强制输出停止功能
0	设置无效
1	写入“1”将启动对TKBOnp输出的高阻抗/低电平/高电平执行固定输出控制（与强制输出停止功能1中检测到触发信号上升沿时的功能相同）。

用于清除强制输出停止功能 1 的清除触发(TKBAHFTn 寄存器的 TKBAHTTnp 位)的设置如下表所示。

表 7-8. 强制输出停止功能 1 的清除触发(TKBAHTTnp 位)的操作

TKBAHTTnp 寄存器		通过软件清除强制输出停止功能1
TKBAHTTnp1	TKBAHTTnp0	
0	0	写入清除触发(TKBAHTTnp) = 1时清除强制输出停止功能（无论触发信号电平如何）。
0	1	当触发信号为有效电平时，写入清除触发(TKBAHTTnp) = 1的操作无效。当触发信号为无效电平时，通过写入清除触发(TKBAHTTnp) = 1来清除强制输出停止功能。
1	0	在写入清除触发(TKBAHTTnp) = 1后的下一个计数器周期清除强制输出停止功能（无论触发信号电平如何）。
1	1	当触发信号为有效电平时，写入清除触发(TKBAHTTnp) = 1的操作无效。当触发信号为无效电平时，在写入清除触发(TKBAHTTnp) = 1后的下一个计数器周期清除强制输出停止功能。

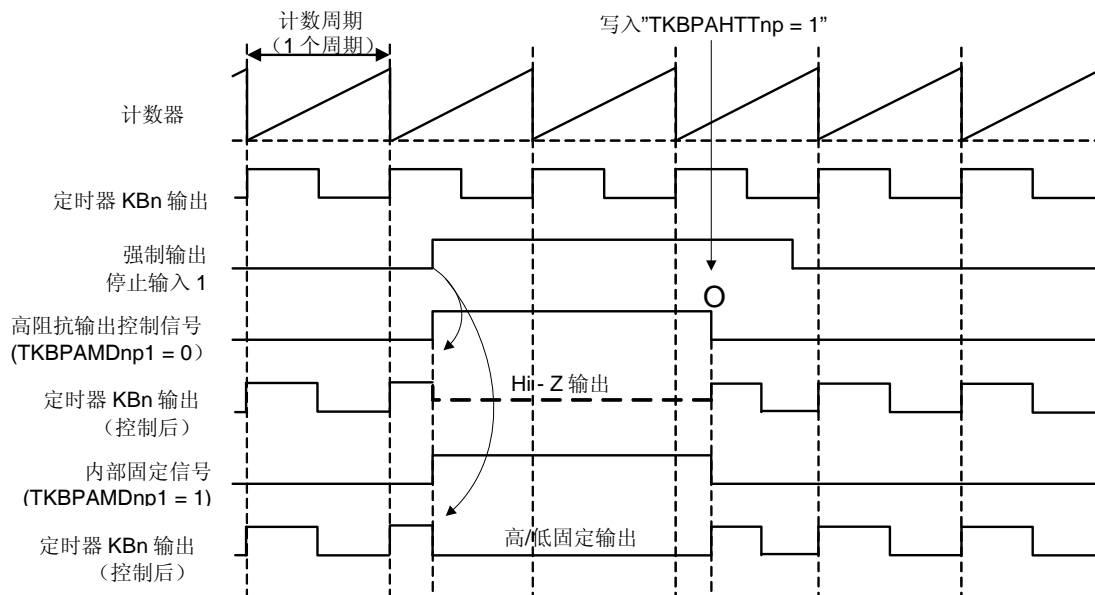
备注 n = 0 至 2, p = 0, 1

7.7.3 强制输出停止功能 1 的基本操作

下面介绍在 TKBPAHCMnp1 和 TKBPAHCMnp0 寄存器的各种设置时强制输出功能 1 的操作。

启动强制输出停止功能 1 的触发信号（强制输出停止输入 1）是，由强制输出停止功能控制寄存器 np (TKBPACTLnp) 的 TKBPAHZSn0 至 TKBPAHZSn2 位选择的触发信号，以及强制输出停止功能 1 开始触发寄存器 n (TKBPAHFSn) 的 TKBPAHTSn 位的 OR 输出。

(1) TKBPAHCMnp1, TKBPAHCMnp0 = 0, 0 时的强制输出停止功能 1



(a) TKBPAMDnp1 = 0 (Hi-z 输出)

检测到强制输出停止输入 1 的上升沿时成为高阻抗输出。

无论强制输出停止输入 1 的输入电平如何，向清除触发（TKBPAHTTnp 位）写入“1”时将返回至定时器输出。

高阻抗输出控制信号的高电平期间即为强制输出停止 1（高阻抗输出）期间。

(b) TKBPAMDnp1 = 1 (固定输出)

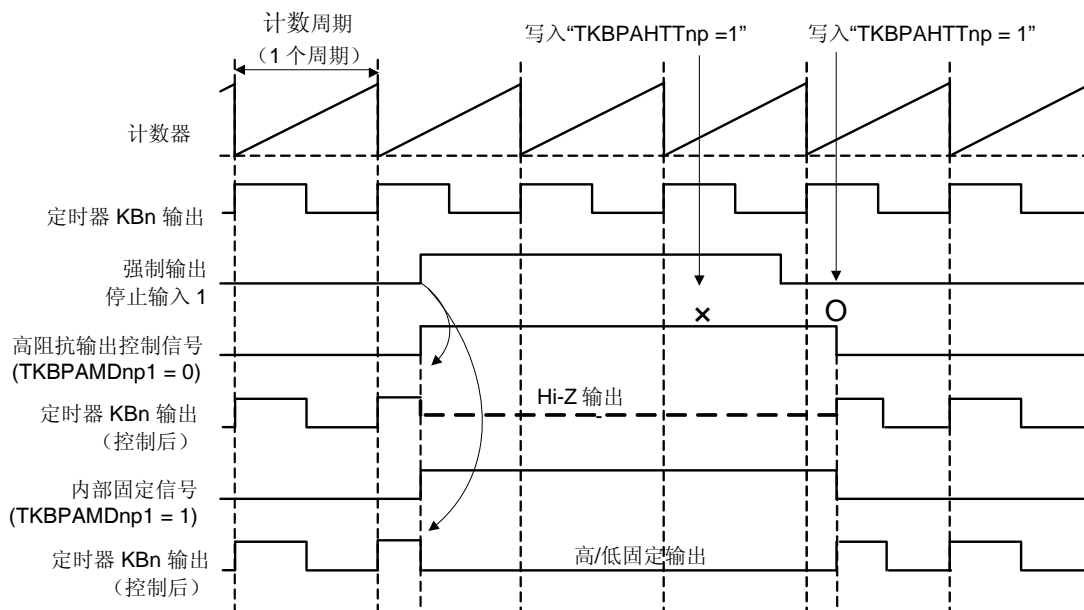
检测到强制输出停止输入 1 的上升沿时，根据 TKBPAMDnp0 设置执行低电平/高电平的固定输出。

无论强制输出停止输入 1 的输入电平如何，向清除触发（TKBPAHTTnp 位）写入“1”时将清除固定输出电平并返回至定时器输出。

内部固定信号的高电平期间即为强制输出停止 1（低电平/高电平固定输出）期间。

备注 n = 0 至 2, p = 0, 1

(2) TKBPAHCMnp1, TKBPAHCMnp0 = 0, 1 时的强制输出停止功能 1



(a) TKBPAMDnp1 = 0 (Hi-z 输出)

检测到强制输出停止输入 1 的上升沿时成为高阻抗输出。

在强制输出停止输入 1 的有效电平（高电平）期间，向清除触发（TKBPAHTTnp 位）写入“1”无效。

在强制输出停止输入 1 改变为无效电平（低电平）的期间，向清除触发（TKBPAHTTnp 位）写入“1”时将返回至定时器输出。

高阻抗输出控制信号的高电平期间即为强制输出停止 1（高阻抗输出）期间。

(b) TKBPAMDnp1 = 1 (固定输出)

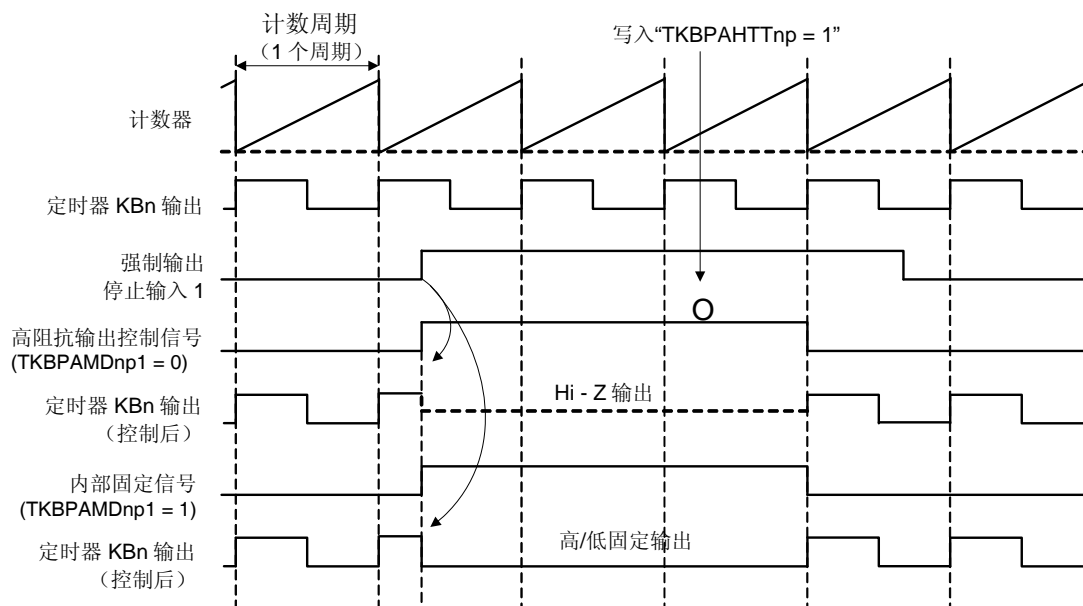
检测到强制输出停止输入 1 的上升沿时，根据 TKBPAMDnp0 设置执行低电平/高电平的固定输出。

在强制输出停止输入 1 的有效电平（高电平）期间，向清除触发（TKBPAHTTnp 位）写入“1”无效。

在强制输出停止输入 1 改变为无效电平（低电平）的期间，向清除触发（TKBPAHTTnp 位）写入“1”时将清除输出电平固定并返回至定时器输出。

内部固定信号的高电平期间即为强制输出停止 1（低电平/高电平固定输出）期间。

备注 n = 0 至 2, p = 0, 1

(3) TKBPAHCMnp1, TKBPAHCMnp0 = 1, 0 时的强制输出停止功能 1**(a) TKBPAMDnp1 = 0 (Hi-z 输出)**

检测到强制输出停止输入 1 的上升沿时成为高阻抗输出。

无论强制输出停止输入 1 的输入电平如何，向清除触发 (TKBPAHTTnp 位) 写入“1”后的下一个计数器周期将返回至定时器输出。

高阻抗输出控制信号的高电平期间即为强制输出停止 1 (高阻抗输出) 期间。

(b) TKBPAMDnp1 = 1 (固定输出)

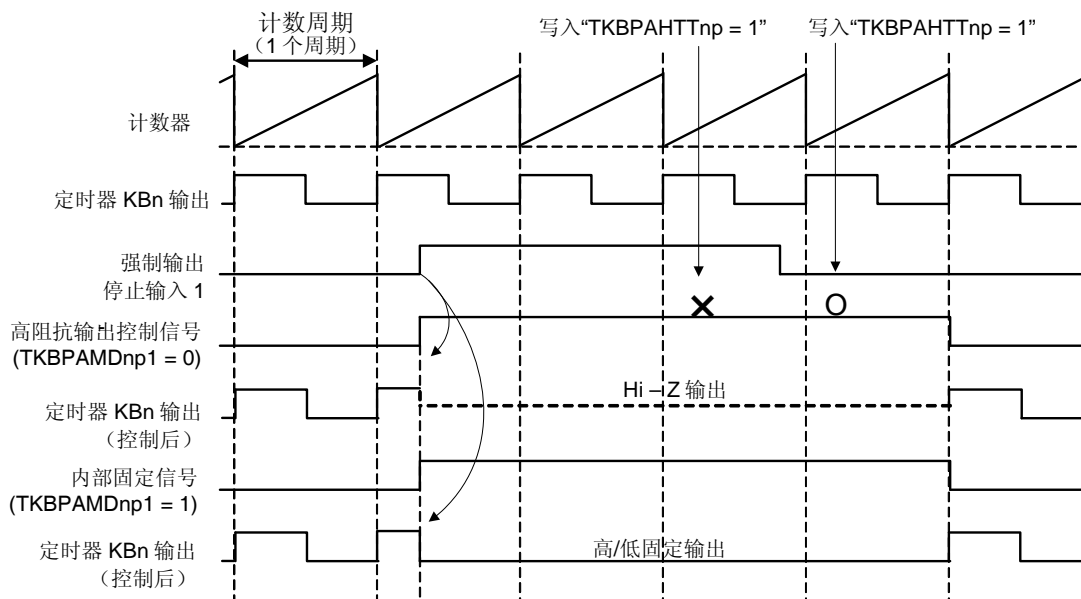
检测到强制输出停止输入 1 的上升沿时，根据 TKBPAMDnp0 设置执行低电平/高电平的固定输出。

无论强制输出停止输入 1 的输入电平如何，向清除触发 (TKBPAHTTnp 位) 写入“1”后的下一个计数器周期将清除固定输出电平并返回至定时器输出。

内部固定信号的高电平期间即为强制输出停止 1 (低电平/高电平固定输出) 期间。

备注 n = 0 至 2, p = 0, 1

(4) TKBPAHCMnp1, TKBPAHCMnp0 = 1, 1 时的强制输出停止功能 1



(a) TKBPAMDnp1 = 0 (Hi-z 输出)

检测到强制输出停止输入 1 的上升沿时成为高阻抗输出。

在强制输出停止输入 1 的有效电平（高电平）期间，向清除触发（TKBPAHTTnp 位）写入“1”无效。

在强制输出停止输入 1 的无效电平（低电平）期间，向清除触发（TKBPAHTTnp 位）写入“1”后的下一个计数器周期将返回至定时器输出。

高阻抗输出控制信号的高电平期间即为强制输出停止 1（高阻抗输出）期间。

(b) TKBPAMDnp1 = 1 (固定输出)

检测到强制输出停止输入 1 的上升沿时，根据 TKBPAMDnp0 设置执行低电平/高电平的固定输出。

在强制输出停止输入 1 的有效电平（高电平）期间，向清除触发（TKBPAHTTnp 位）写入“1”无效。

在强制输出停止输入 1 成为无效电平（低电平）的期间，向清除触发（TKBPAHTTnp 位）写入“1”后的下一个计数器周期将清除输出电平固定并返回至定时器输出。

内部固定信号的高电平期间即为强制输出停止 1（低电平/高电平固定输出）期间。

备注 n = 0 至 2, p = 0, 1

7.8 强制输出停止功能 2 的操作

当出现触发源（比较器 0 至 5 输出、INTP20）时，可以与 16 位定时器 KBn 和 KC0 电路的操作时钟 f_{KBKC} 异步地直接将定时器输出固定为高电平或低电平（不通过 CPU）。出现触发源或者触发源信号变为无效电平后，系统将在下一个计数器周期开始时取消强制输出停止状态。

7.8.1 强制输出停止功能 2 的概要

在该功能中，使用比较器输出信号和软件触发作为强制输出停止功能 2 的触发信号。

强制输出停止功能可以选择的输出电平由 TKBPACTLnp 寄存器的 TKBPAMDnp0 和 TKBPAMDnp1 位控制。

下表显示了其与定时器 KBn 的输出 p 引脚(TKBOnp)的强制输出停止功能 2 之间的关系。

TKBPAMDnp1	TKBPAMDnp0	选择执行强制输出停止功能 2 时的输出电平
0	0	低电平固定输出
0	1	高电平固定输出
1	0	低电平固定输出
1	1	高电平固定输出

所用比较器输出的选择由强制输出停止功能控制寄存器 np (TKBPACTLnp)的 TKBPAFXSn3 至 TKBPAFXSn0 位控制。

下表显示了定时器 KBn 的输出 p 引脚(TKBOnp)的强制输出停止功能 2 的触发选择。

位	可选择的触发信号		
	定时器KB0	定时器KB1	定时器KB2
TKBPAFXSn0	比较器0		
TKBPAFXSn1	比较器1	比较器2	比较器4
TKBPAFXSn2	比较器2	比较器3	比较器5
TKBPAFXSn3	INTP20	INTP20	INTP20

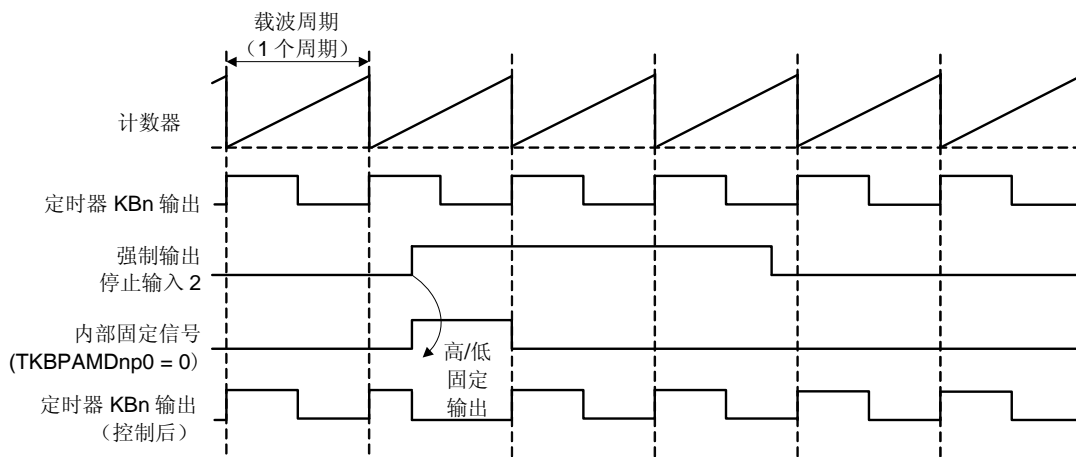
备注 n = 0至2, p = 0, 1

7.8.2 强制输出停止功能 2 的基本操作

下面介绍在 TKBPAFCMnp 位的各种设置时强制输出功能 2 的操作。

启动强制输出停止功能 2 的触发信号（强制输出停止输入 2）是，由强制输出停止功能控制寄存器 np (TKBPACTLnp) 的 TKBPAFXSnp0 至 TKBPAFXSnp3 位选择的触发信号。

(1) TKBPAFCMnp = 0 时的强制输出停止功能 2



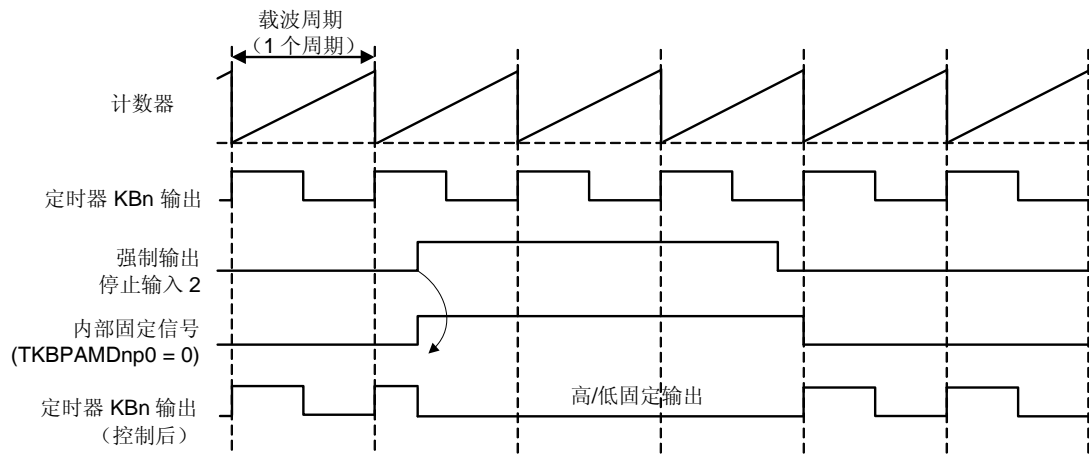
检测到强制输出停止输入 2 的上升沿时，根据 TKBPAMDnp0 设置执行低电平/高电平的固定输出。

无论强制输出停止输入 2 的输入电平如何，在下一个计数器周期将清除固定输出电平并返回至定时器输出。

内部固定信号的高电平期间即为强制输出停止 2（低电平/高电平固定输出）期间。

备注 n = 0 至 2, p = 0, 1

(2) TKBPFCMnp0 = 1 时的强制输出停止功能 2



检测到强制输出停止输入 2 的上升沿时，根据 TKBPAMDnp0 设置执行低电平/高电平的固定输出。
 在强制输出停止输入 1 的有效电平（高电平）期间，向清除触发（TKBPAHTTnp 位）写入“1”无效。
 强制输出停止输入 2 成为反相边沿后，在下一个计数器周期将清除固定输出电平并返回至定时器输出。
 内部固定信号的高电平期间即为强制输出停止 2（低电平/高电平固定输出）期间。

备注 n = 0 至 2, p = 0, 1

第八章 16 位定时器 KC0

8.1 16 位定时器KC0 的功能

所有 RL78/I1A 微控制器产品中均安装有 16 位定时器 KC0。

16 位定时器 KC0 具备 6 个输出，最多可生成 6 种 PWM 输出。

此外，通过与定时器 KB0、KB1 和 KB2 联锁，最多可对 6 种 PWM 输出执行门控。

16 位定时器 KC0 具备以下功能。

(1) PWM 输出

- 在定时器工作时，对于一个周期可为任意占空比的可变脉冲被输出。
- 可以设置默认的定时器输出电平（高电平或低电平）。

(2) PWM 输出门控功能（通过与 16 位定时器 KB0、KB1 和 KB2 的联锁）

通过定时器 KC 输出，最多可以分别以任意占空比对最多 6 组的 16 位定时器 KB0、KB1、KB2 输出（TKBO00、TKBO01、TKBO10、TKBO11、TKBO20、TKBO21 输出）执行门控。

(3) 同时开始&停止模式

通过将 16 位定时器 KB 设置为主定时器，将 16 位定时器 KC 设置为从属定时器，可以与定时器 KB0 的计数开始/停止时序保持同步地启动/停止 16 位从属定时器 KC。

8.2 16 位定时器KC0 的配置

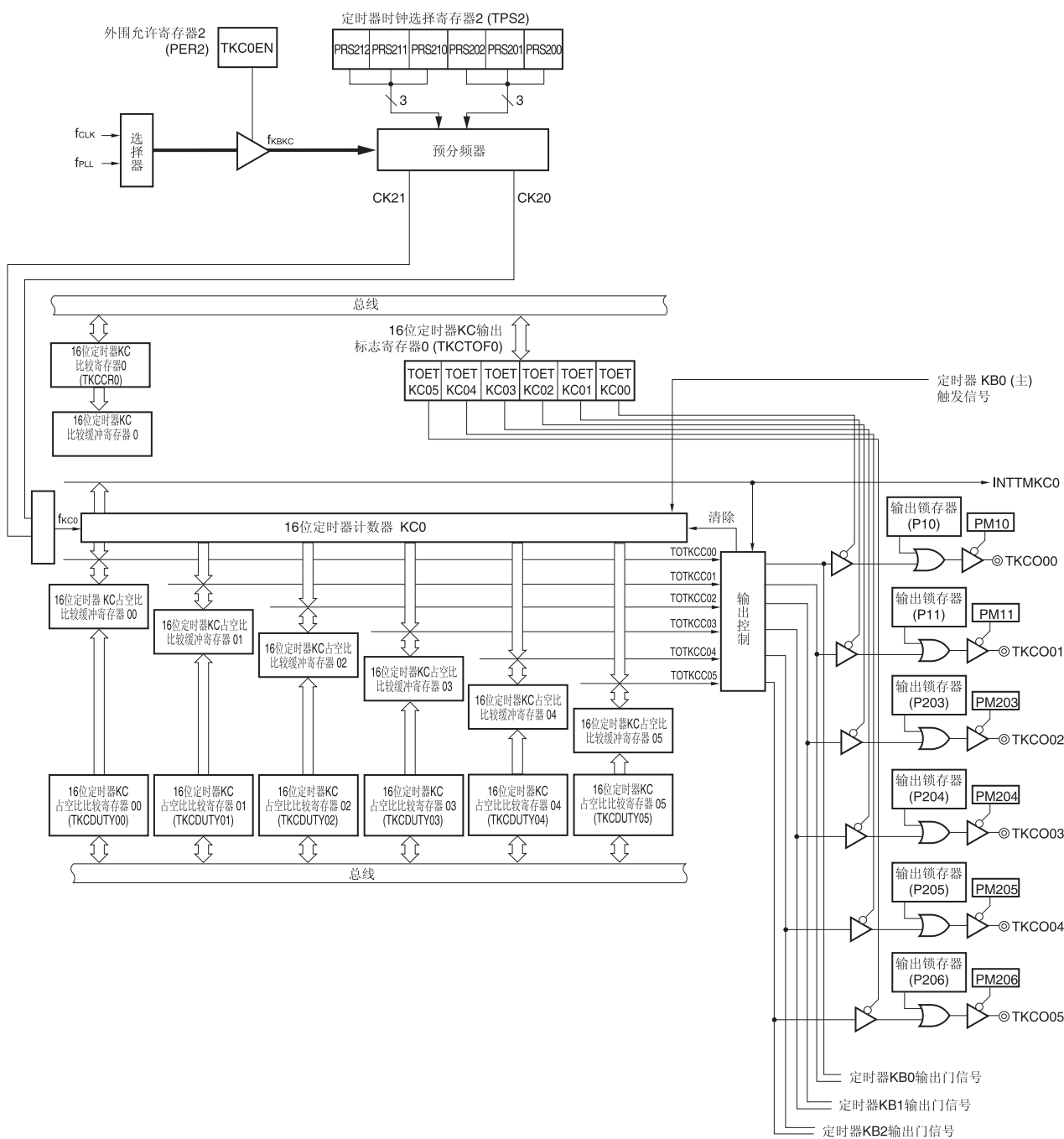
16 位定时器 KC0 包括以下硬件。

表 8-1. 16 位定时器 KC0 的配置

项目	配置
定时器/计数器	16 位定时器计数器 KC0 (TKCNT0)
寄存器	16 位定时器 KC 比较寄存器 0 (TKCCR0) 16 位定时器 KC 占空比较寄存器 0 至 5 (TKCDUTY00 至 TKCDUTY05)
定时器输出	TKCO00 至 TKCO05
控制寄存器	16 位定时器 KC 操作控制寄存器 0 (TKCCTL0) 16 位定时器 KC 输出控制寄存器 00 (TKCIOC00) 16 位定时器 KC 输出控制寄存器 01 (TKCIOC01) 16 位定时器 KC 输出引脚控制寄存器(TOETKC0) 16 位定时器 KC 输出标志寄存器 0 (TKCTOF0) 16 位定时器 KC 标志寄存器 0 (TKCFLG0) 16 位定时器 KC 触发寄存器 0 (TKCTRG0) 端口模式寄存器 1, 20 (PM1, PM20) 端口寄存器 1, 20 (P1, P20)

框图如图 8-1 所示。

图 8-1. 16 位定时器 KC0 的框图



- 备注 1. f_{KBK} : 16 位定时器 KB_n 和 KC_0 电路整体的操作时钟
 f_{KC_0} : 16 位定时器 KC_0 的计数时钟
 2. $n = 0$ 至 2

(1) 16 位定时器 KC 比较寄存器 0 (TKCCR0)

TKCCR0 寄存器用于设置定时器 KC 的周期。

当定时器计数(TKCCE0 = 1)时，可以刷新 TKCCR0（写入相同的值）或者重写它的值。在定时器工作时如果重写 TKCCR0 的值，该值将被锁存，并在下述时序被传送至 TKCCR0，从而改变 TKCCR0 的值。

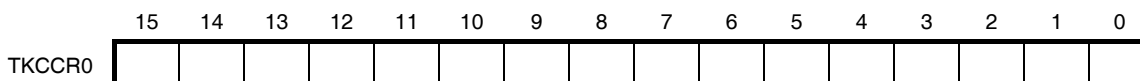
- 当计数器开始计数操作(TKCCE0 = 0)时
- 当出现批量覆盖触发(TKCRDT0 = 1)或外部触发(TKCTSE0 = 1)时

可以按 16 位单位读取或写入。

产生复位信号后，该寄存器被清除为 0000H。

图 8-2. 16 位定时器 KC 比较寄存器 0 (TKCCR0)的格式

地址: F06D0H 复位后: 0000H R/W

**(2) 16 位定时器 KC 占空比较寄存器 00 至 05 (TKCDUTY00 至 TKCDUTY05)**

TKCDUTY0m 寄存器用于设置定时器 KC 输出的有效期间。

当定时器计数(TKCCE0 = 1)时，可以刷新 TKCDUTY0m（写入相同的值）或者重写它的值。在定时器工作时如果重写 TKCDUTY0m 的值，该值将被锁存，并在下述时序被传送至 TKCDUTY0m，从而改变 TKCDUTY0m 的值。

- 当计数器开始计数操作(TKCCE0 = 0)时
- 当出现批量覆盖触发(TKCRDT0 = 1)或外部触发(TKCTSE0 = 1)时

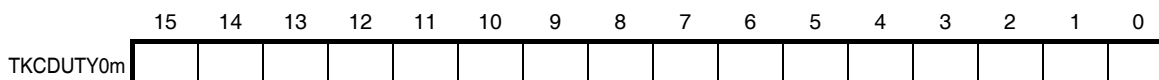
可以按 16 位单位读取或写入。

产生复位信号后，该寄存器被清除为 0000H。

图 8-3. 16 位定时器 KC 占空比较寄存器 0m (TKCDUTY0m)的格式

地址: F06D2H (TKCDUTY00), F06D4H (TKCDUTY01), F06D6H (TKCDUTY02), 复位后: 0000H R/W
F06D8H (TKCDUTY03), F06DAH (TKCDUTY04), F06DCH (TKCDUTY05)

复位后: 0000H R/W



备注 m = 0 至 5

8.3 控制 16 位定时器KC0 的寄存器

以下寄存器用于控制 16 位定时器 KC0。

- 外围允许寄存器 2 (PER2)
- 定时器时钟选择寄存器 2 (TPS2)
- 16 位定时器 KC 操作控制寄存器 0 (TKCCTL0)
- 16 位定时器 KC 输出控制寄存器 00 (TKCIOC00)
- 16 位定时器 KC 输出控制寄存器 01 (TKCIOC01)
- 16 位定时器 KC 输出引脚控制寄存器 (TOETKC0)
- 16 位定时器 KC 输出标志寄存器 0 (TKCTOF0)
- 16 位定时器 KC 标志寄存器 0 (TKCFLG0)
- 16 位定时器 KC 触发寄存器 0 (TKCTRG0)
- 端口模式寄存器 1, 20 (PM1, PM20)
- 端口寄存器 1, 20 (P1, P20)

(1) 外围允许寄存器 2 (PER2)

该寄存器用于允许或禁止供应时钟至外围硬件。并且中止向未使用的硬件供应时钟，以减少电力消耗和噪声。

当使用比较器 KC0 时，务必确实设置该寄存器的位 3 (TKC0EN) 为 1。

使用 8 位存储器操作指令来设置 PER2 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 8-4. 外围允许寄存器 2 (PER2) 的格式

地址: F0509H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
PER2	PGACMPEN	TKBPA2EN	TKBPA1EN	TKBPA0EN	TKC0EN	TKB2EN	TKB1EN	TKB0EN

TKC0EN	定时器KC0输入时钟的控制
0	停止输入时钟的供应。 <ul style="list-style-type: none"> 不可写入用于定时器KC0的SFR。 定时器KC0 处于初始状态。
1	供应输入时钟。 <ul style="list-style-type: none"> 可以读取/写入用于定时器KC0的SFR。

注意事项 要设置定时器 KC0，务必首先将 TKC0EN 位设置为“1”。若 TKC0EN = 0 时，写入到定时器 KC0 的控制寄存器会被忽略，并且所有的读取值均变为初始值(除定时器时钟选择寄存器 2 (TPS2)，端口模式寄存器 1、20 (PM0, PM20)和端口寄存器 1、20 (P1, P20)以外)。

(2) 定时器时钟选择寄存器 2 (TPS2)

TPS2 寄存器为 16 位寄存器，用于选择通常由外部预分频器供应至定时器 KB0、KB1、KB2 和 KC0 的两种操作时钟 (CK20、CK21)。使用 TPS2 寄存器的位 6 至位 4 选择 CK21，使用位 2 至位 0 选择 CK20。

只有在以下情况下才可以可定时器操作期间重写 TPS2 寄存器。

如果可以重写 PRS200 至 PRS202 位(n = 0 至 2):

选择 CK20 作为操作时钟(TKBCKSn = 0, TKCKS0 = 0)的所有通道处于停止状态(TKBCEn = 0, TKCCE0 = 0)。

如果可以重写 PRS210 至 PRS212 位(n = 0 至 2):

选择 CK21 作为操作时钟(TKBCKSn = 1, TKCKS0 = 1)的所有通道处于停止状态(TKBCEn = 0, TKCCE0 = 0)。

使用 8 位存储器操作指令来设置 TPS2 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 8-5. 定时器时钟选择寄存器 2 (TPS2)的格式

地址: F05C5H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
TPS2	0	TPS 212	TPS 211	TPS 210	0	TPS 202	TPS 201	TPS 200

TPS 2k2	TPS 2k1	TPS 2k0	操作时钟(CK2k)的选择 ^{#1, 2} (k = 0, 1)					
				f _{CLK} = 2 MHz	f _{CLK} = 5 MHz	f _{CLK} = 20 MHz	f _{CLK} = 32 MHz	f _{PLL} = 64 MHz
0	0	0	f _{CLK} , f _{PLL}	2 MHz	5 MHz	20 MHz	32 MHz	64 MHz
0	0	1	f _{CLK} /2, f _{PLL} /2	1 MHz	2.5 MHz	10 MHz	16 MHz	32 MHz
0	1	0	f _{CLK} /2 ² , f _{PLL} /2 ²	500 kHz	1.25 MHz	5 MHz	8 MHz	16 MHz
0	1	1	f _{CLK} /2 ³ , f _{PLL} /2 ³	250 kHz	625 kHz	2.5 MHz	4 MHz	8 MHz
1	0	0	f _{CLK} /2 ⁴ , f _{PLL} /2 ⁴	125 kHz	312.5 kHz	1.25 MHz	2 MHz	4 MHz
1	0	1	f _{CLK} /2 ⁵ , f _{PLL} /2 ⁵	62.5 kHz	156.2 kHz	625 kHz	1 MHz	2 MHz
1	1	0	禁止设置	-	-	-	-	-
1	1	1	禁止设置	-	-	-	-	-

注 1. 要更改被选作 f_{CLK} 的时钟（通过更改系统时钟控制寄存器(CKC)的值）时，须停止 KB0、KB1、KB2 和 KC0(TKBCEn = 0, TKCCE0 = 0)。

2. 当 PLL 控制寄存器(PLLCTL)中 PLLON = 1 时，供应 f_{PLL}。

注意事项 务必将位 7 和 3 清除为“0”。

备注 f_{CLK}: CPU/外围硬件时钟频率

f_{PLL}: PLL 输出时钟

(3) 16 位定时器 KC 操作控制寄存器 0 (TKCCTL0)

TKCCTL0 寄存器用于控制计数操作以及设置 16 位定时器的计数时钟。

使用 1 位或 8 位存储器操作指令来设置 TKCCTL0 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 8-6. 16 位定时器 KC 操作控制寄存器 0 (TKCCTL0)的格式

地址: 06E5H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
TKCCTL0	TKCCE0	0	0	TKCCKS0	0	0	0	TKCMD0

TKCCE0	定时器KC0操作的控制
0	停止定时器操作（计数器被清除为 FFFFH）。
1	允许定时器计数操作。

TKCCKS0	定时器KC0时钟的选择
0	通过 TPS202 至 TPS200 位选择 CK20 时钟
1	通过 TPS212 至 TPS210 位选择 CK21 时钟

TKCMD0	定时器KC0操作模式的选择
0	单机模式
1	同步开始/停止模式（用于从属，定时器 KB0 作为主）

- 注意事项**
1. 在定时器操作期间，禁止设置 TKCCTL0 寄存器的其它位。不过，可以刷新 TKCCTL0 寄存器（写入相同的值）。
 2. 务必将位 6、5、3 至 1 清除为“0”。

(4) 16 位定时器 KC 输出控制寄存器 00 (TKCIOC00)

TKCIOC00 寄存器用于设置 16 位定时器 KBn 输出 TKCO0m 的默认电平/有效电平。

使用 16 位存储器操作指令设置 TKCIOC00 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 8-7. 16 位定时器 KC 输出控制寄存器 00 (TKCIOC00)的格式

地址: F06E2H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8
TKCIOC00	0	0	TKCTOL05	TKCTOL04	TKCTOL03	TKCTOL02	TKCTOL01	TKCTOL00
	7	6	5	4	3	2	1	0
	0	0	TKCTOD05	TKCTOD04	TKCTOD03	TKCTOD02	TKCTOD01	TKCTOD00

TKCTOL0m	定时器输出TKCO0m的有效电平设置
0	有效高电平
1	有效低电平

TKCTOD0m	定时器输出TKCO0m的默认电平设置
0	低电平（通常输出）
1	高电平（反相输出）

- 注意事项
1. 在定时器操作期间，禁止设置 TKCIOC00 寄存器的其它位。不过，可以刷新 TKCIOC00 寄存器（写入相同的值）。
 2. 务必将位 15、14、7 和 6 清除为“0”。
 3. TKCO0m 输出以及兼用端口的端口模式寄存器(PMxx)和端口寄存器(Pxx)共同决定着实际的 TKCO0m 引脚输出。

备注 m = 0 至 5

(5) 16 位定时器 KC 输出控制寄存器 01 (TKCIOC01)

TKCIOC01 寄存器用于设置将 16 位定时器 KC0 输出用作 16 位定时器 KB 的 PWM 输出门控功能或者充当 TKCO0p 输出。

如果用作 PWM 输出门控功能，需要对 16 位定时器 KB 进行设置。

同样，如果用该输出充当 TKCO0p 输出，则除了端口寄存器和端口模式寄存器以外，还需要对 TOETKC0 寄存器进行设置。

使用 1 位或 8 位存储器操作指令来设置 TKCIOC01 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 8-8. 16 位定时器 KC 输出控制寄存器 01 (TKCIOC01)的格式

地址: F06E4H 复位后: 00H R/W

符号	7	6	<5>	<4>	<3>	<2>	<1>	<0>
TKCIOC01	0	0	TKCTOE05	TKCTOE04	TKCTOE03	TKCTOE02	TKCTOE01	TKCTOE00
TKCTOE0m	定时器KB的PWM输出门控功能的使用以及定时器输出TKCO0m的输出允许/禁止							
0	使用定时器KB的PWM输出门控功能且禁止定时器输出（当TKCTOD0m = 0时固定为低电平输出，当TKCTOD0m = 1时固定为高电平输出。）							
1	使用定时器KB的PWM输出门控功能且允许定时器输出（PWM输出）							

- 注意事项**
1. 定时器操作期间，可以覆盖 TKCIOC01 寄存器。
 2. 务必将位 7 和 6 清除为“0”。
 3. TKCO0m 输出以及兼用端口的端口模式寄存器(PMxx)和端口寄存器(Pxx)共同决定着实际的 TKCO0m 引脚输出。

备注 m = 0 至 5

(6) 16 位定时器 KC 输出引脚控制寄存器(TOETKC0)

该寄存器用于允许/禁止将来自 16 位定时器 KC 的定时器输出传送至引脚。

无论定时器输出门控功能（与 16 位定时器 KB 联动）的设置如何，均可控制 TKCOm 的输出。

使用 1 位或 8 位存储器操作指令来读取 TOETKC0 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 8-9. 16 位定时器 KC 输出引脚控制寄存器(TOETKC0)的格式

地址: F05C8H 复位后: 00H R

符号	7	6	<5>	<4>	<3>	<2>	<1>	<0>
TOETKC0	0	0	TOETKC05	TOETKC04	TOETKC03	TOETKC02	TOETKC01	TOETKC00
TOETKC0m	定时器输出TKCO0m的引脚输出允许/禁止							
0	禁止TKCO0m的引脚输出。							
1	允许TKCO0m的引脚输出。							

备注 m = 0 至 5

(7) 16 位定时器 KC 输出标志寄存器 0 (TKCTOF0)

TKCTOF0 是 16 位定时器 KC0 输出的 TKCO0m 的监视器标志寄存器。

使用 1 位或 8 位存储器操作指令来读取 TKCTOF0 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 8-10. 16 位定时器 KC 输出标志寄存器 0 (TKCTOF0)的格式

地址: F06E6H 复位后: 00H R

符号	7	6	<5>	<4>	<3>	<2>	<1>	<0>
TKCTOF0	0	0	TKCTOF05	TKCTOF04	TKCTOF03	TKCTOF02	TKCTOF01	TKCTOF00

TKCTOF0m	定时器输出TKCO0m的监视器标志
0	低电平状态
1	高电平状态

注意事项 TKCO0m 输出以及兼用端口的端口模式寄存器(PMxx)和端口寄存器(Pxx)共同决定着实际的 TKCO0m 引脚输出。

备注 m = 0 至 5

(8) 16 位定时器 KC 标志寄存器 0 (TKCFLG0)

TKCFLG0 寄存器用于显示 16 位定时器 KC0 的状态标志。

使用 1 位或 8 位存储器操作指令来读取 TKCFLG0 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 8-11. 16 位定时器 KC 标志寄存器 0 (TKCFLG0)的格式

地址: F06DFH 复位后: 00H R

符号	7	6	5	4	3	2	1	<0>
TKCFLG0	0	0	0	0	0	0	0	TKCRSF0

TKCRSF0	批量覆盖触发的保留状态标志
0	批量覆盖允许状态，或由批量覆盖触发引起的批量覆盖完成
1	批量覆盖保留（等待完成）状态 或将“01H”写入批量覆盖触发寄存器(TKCTRG0)

(9) 16 位定时器 KC 触发寄存器 0 (TKCTRG0)

TKCTRG0 是触发寄存器，用于批量覆盖 16 位定时器 KC0 的比较寄存器。

使用 1 位或 8 位存储器操作指令来写入 TKCTRG0 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 8-12. 16 位定时器 KC 触发寄存器 0 (TKCTRG0)的格式

地址: F06DEH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
TKCTRG0	0	0	0	0	0	0	0	TKCRDT0

TKCRDT0	比较寄存器的批量覆盖请求触发
0	设置无效
1	比较寄存器的批量覆盖请求 (将TKCRSF0设置为“1”)

备注 TKCTRG0 寄存器的读取值为 0。

(10) 端口模式寄存器 1, 20 (PM1, PM20)

该寄存器按 1 位单位指定端口 1 或 20 的输入模式或输出模式。

将 P10/TKCO00/INTP20/SO00/TxD0/SCLA0/(DALITxD4),

P11/TKCO01/INTP21/SI00/RxD0/SDAA0/(DALIRxD4)/(TI07)/(TxRx4), P12/(TKCO03)/SCK00,

P203/TKCO02/TKBO11/(INTP20), P204/TKCO03/TKBO20, P205/TKCO04/TKBO21/DALITxD4 和

P206/TKCO05/DALIRxD4/TxRx4/INTP23 引脚作为定时器输出使用时, 请将 PM10 至 PM12 和 PM203 至 PM206 以及 P10 至 P12 和 P203 至 P206 的输出锁存器设置为 0。

使用 1 位或 8 位存储器操作指令来设置 PM1 和 PM20 寄存器。

产生复位信号后, 设置该寄存器为 FFH。

图 8-13. 端口模式寄存器 1、20 的格式 (PM1, PM20) (38 引脚产品)

地址: FFF21H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM1	1	1	1	1	1	PM12	PM11	PM10

PM1y	选择P1x引脚的输入/输出模式 (y = 0至2)
0	输出模式 (输出缓冲器on)
1	输入模式 (输出缓冲器off)

地址: F0510H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM20	1	PM206	PM205	PM204	PM203	PM202	PM201	PM200

PM20x	选择P20x引脚的输入/输出模式 (x = 0至6)
0	输出模式 (输出缓冲器on)
1	输入模式 (输出缓冲器off)

注意事项 务必将 PM1 寄存器的位 3 至 7 以及 PM20 寄存器的位 7 设置为“1”。

对于 32, 30 和 20 引脚产品, 在解除复位后, 必须通过软件将以下的位设置为输出模式 (通过设置端口寄存器和端口模式寄存器为 0 来指定)。

32 引脚产品: PM1 寄存器的位 2

30 引脚产品: PM1 寄存器的位 2

20 引脚产品: PM1 寄存器的位 2 和 PM20 寄存器的位 4 至 6

备注 38 引脚产品的端口模式寄存器 1 和 20 的格式如上图所示。关于其它产品的端口模式寄存器的格式, 请参阅表 4-13. 配置在各个产品上的 PMxx、Pxx、PUxx、PIMxx、POMxx、PMCx 寄存器及其对应的位。

8.4 16 位定时器KC0 的操作

图 8-14. 定时器 KC 操作设置示例 (操作开始流程图)

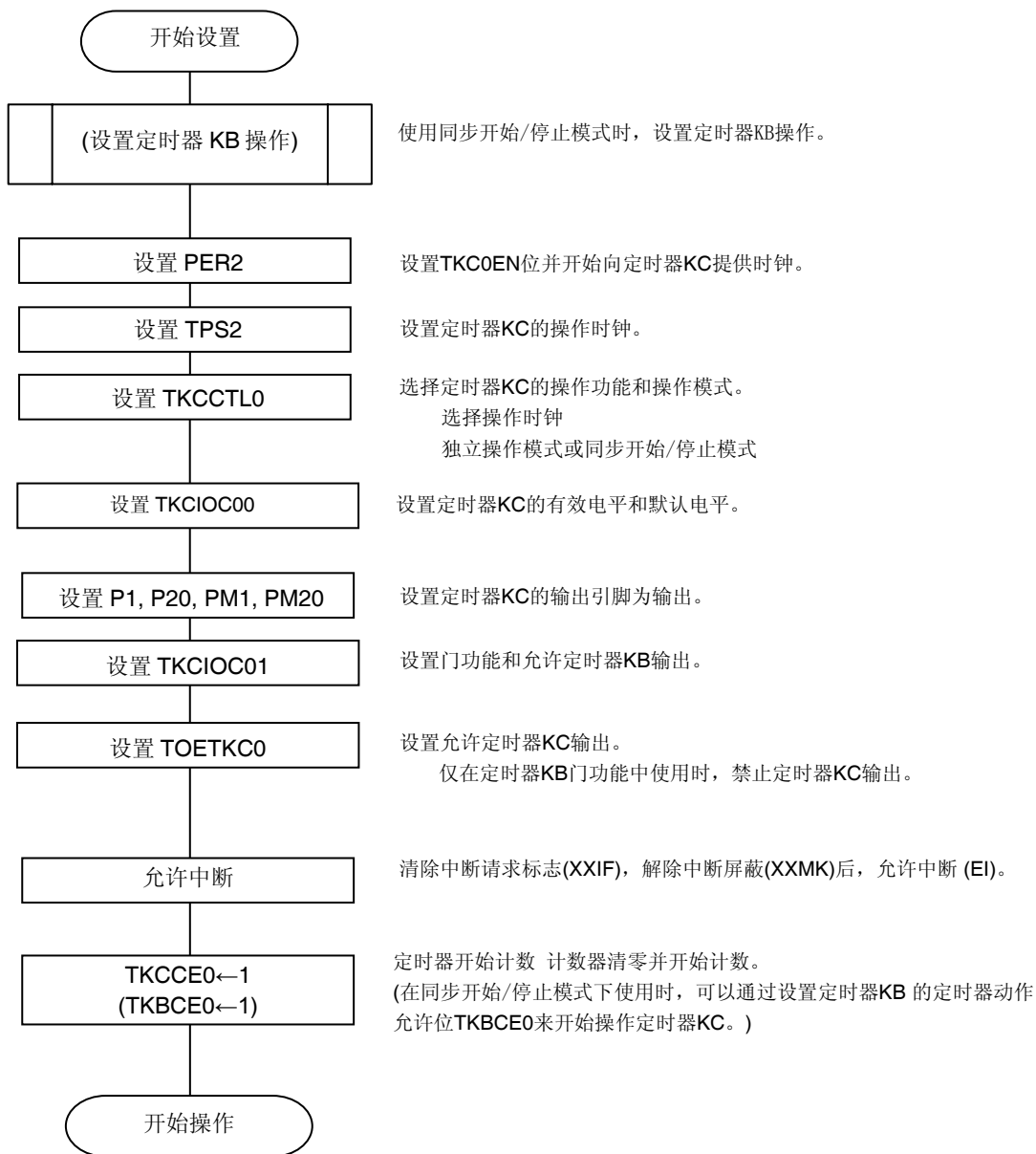
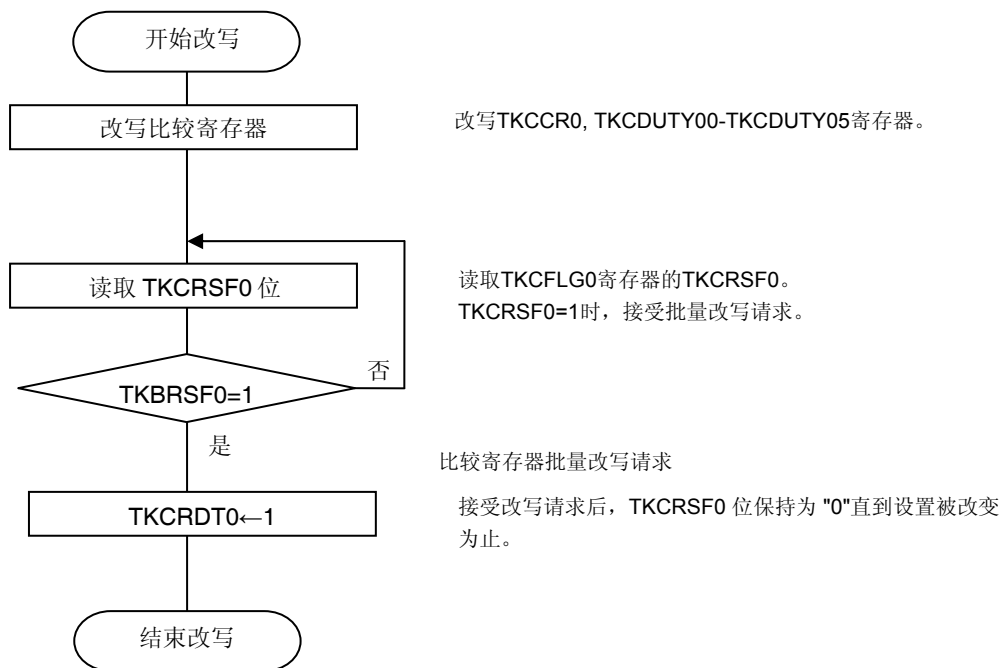


图 8-15. 定时器 KC 操作设置示例 (操作停止流程图)



图 8-16. 定时器 KC 操作设置示例 (比较寄存器批量覆盖流程图)



备注 在定时器 KB 操作中更改定时器计数操作定时，使用批量处理功能。设定值将在下一次重启后生效。

8.4.1 PWM输出功能

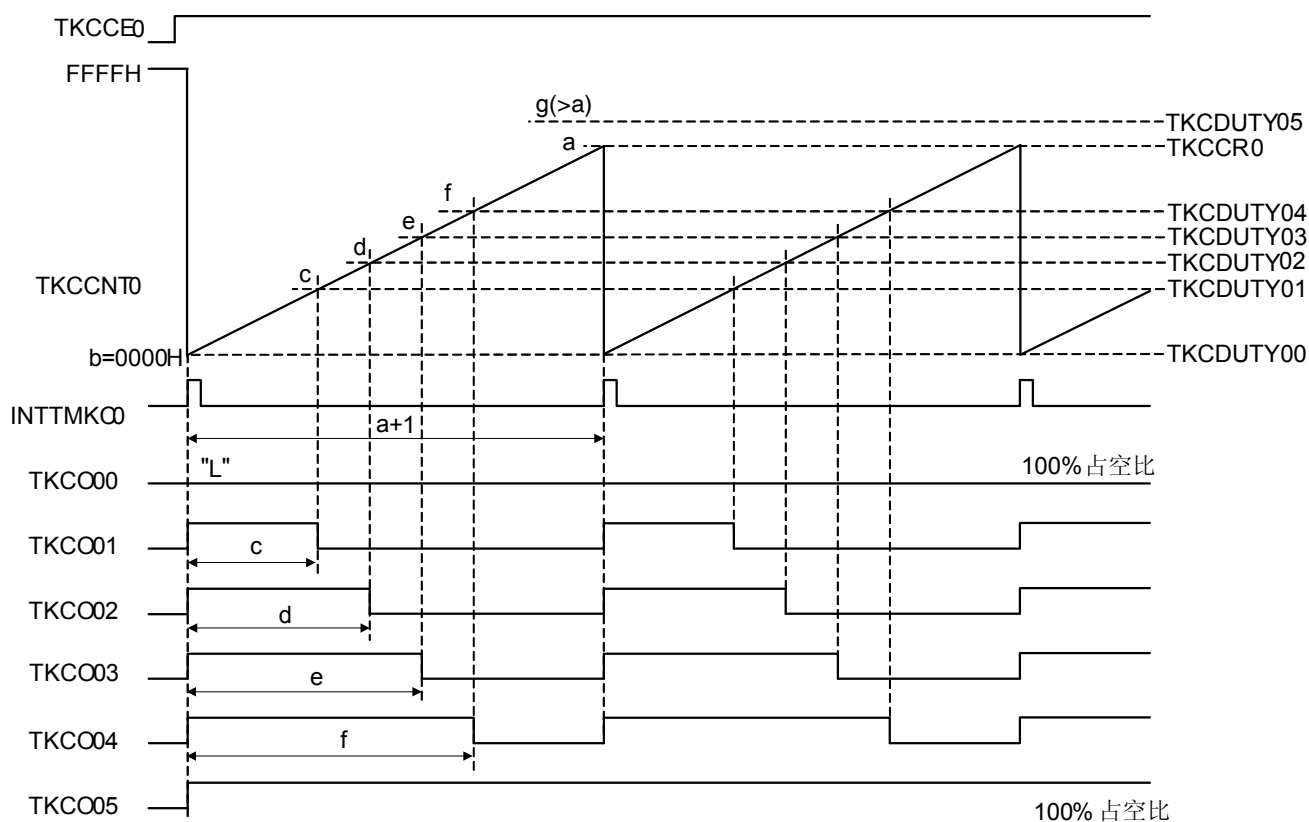
16 位定时器 KC0 可以输出 6 种具有相同周期但可以分别设定不同占空比的 PWM 波形。
可以在 0%至 100%的范围内独立地设置占空比，并且可以使用以下公式计算 TKCO0m 周期和占空比。

脉冲周期 = (TKCCR0 的设置值 + 1) × 计数器时钟周期
占空比[%] = (TKCDUTY0m 的设置值 / (TKCCR0 的设置值 + 1)) × 100

- 0%输出: TKCDUTY0m 的设置值 = 0000H
- 100%输出: TKCDUTY0m 的设置值 ≤ TKCCR0 的设置值 + 1

备注 m = 0 至 5

图 8-17. PWM 输出功能的基本时序例 (TKCTOL0m = 0, TKCTOD0m = 0)

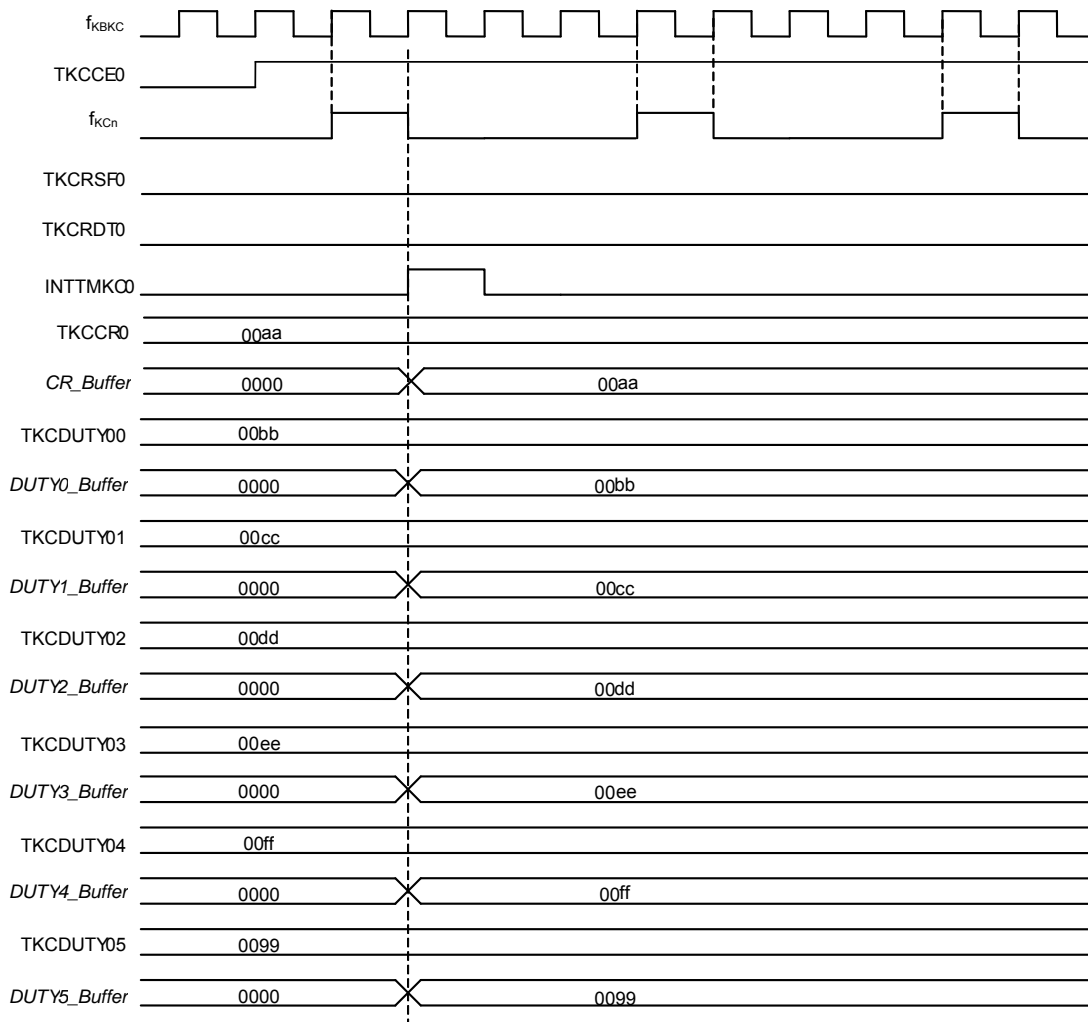


备注 上述图示为 TKCO00 (0%)至 TKCO05 (100%)时

(1) 批量覆盖功能：开始计数操作时更新缓冲器

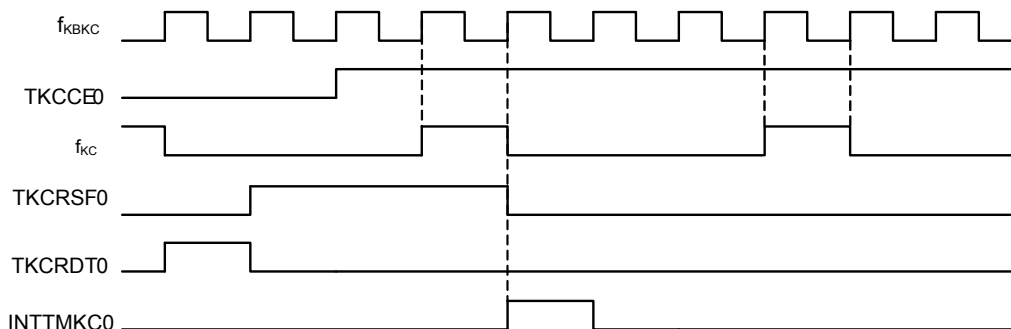
定时器 KC0 的比较寄存器具有在计数器操作（用“1”覆盖 TKCCE0 位之后生成计数时钟而引起）开始的同时批量更新内部缓冲器寄存器的功能。仅在计数操作开始时序处不需要用“1”覆盖 TKCRDT0 位即可发生批量覆盖。

图 8-18. 批量覆盖功能：计数操作开始时缓冲器更新的时序图
(TKCTOL0m = 0, TKCTOD0m = 0)



当 TKCCE0 = 0 时，若向 TKCRDT0 写入“1”，则 TKCRSF0 被设置为“1”。
在计数操作开始时序（产生了计数器开始触发）时，TKCRSF0 被清除为“0”。

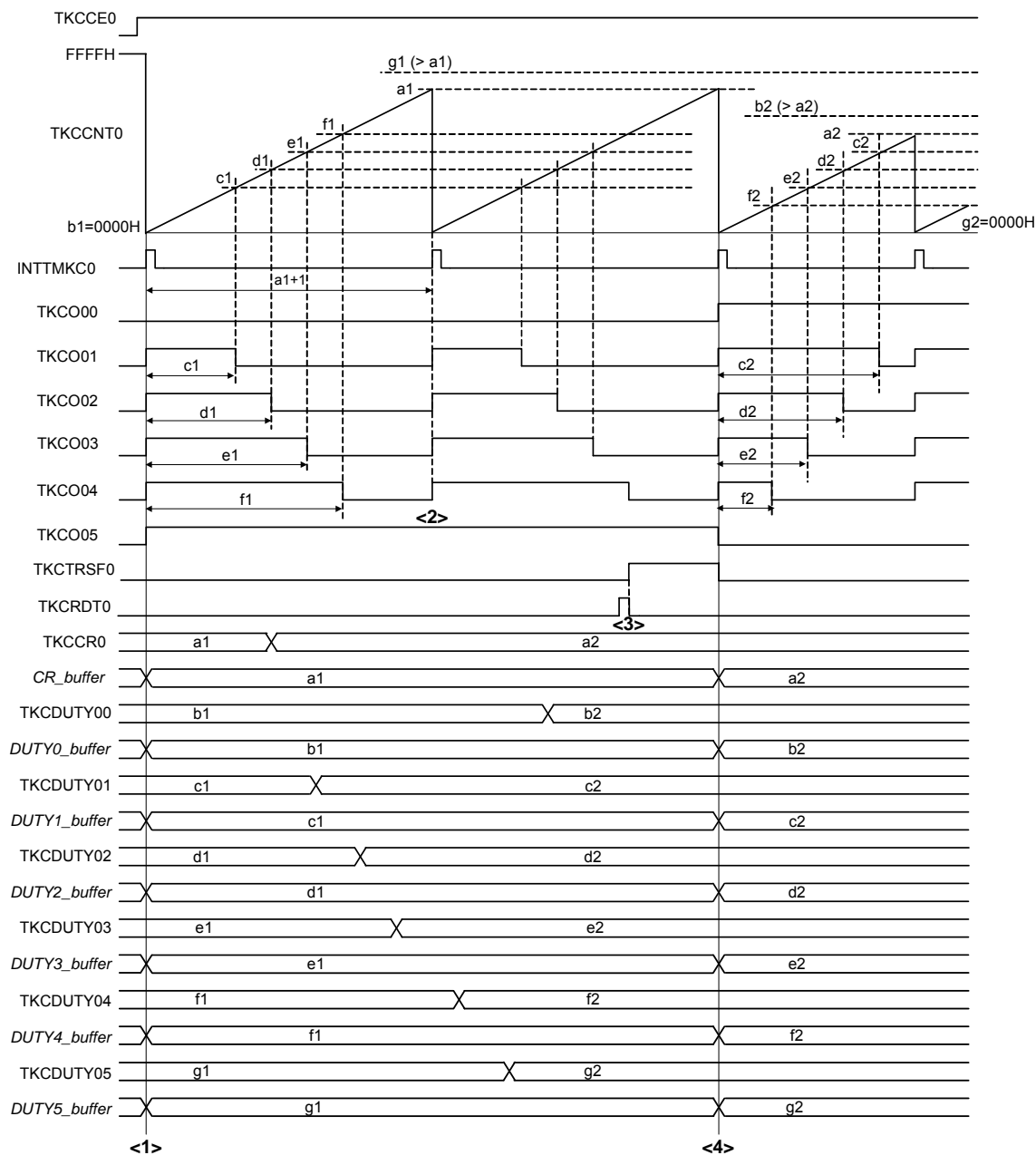
图 8-19. 批量覆盖功能：计数操作开始时缓冲器更新的时序图



(2) 批量覆盖功能：计数操作期间更新缓冲器

定时器 KC0 的比较寄存器把用“1”覆盖 TKCRDT0 位的操作识别为批量覆盖触发，在下次发生计数器清除（TKCNT0 和 TKCCR0 匹配）的同时批量更新内部缓冲器寄存器。TKCRSF0 作为标志，用于显示将“1”写入 TKCRDT0 位直到完成批量覆盖。有关详情，请参阅图 8-20。

图 8-20. 批量覆盖功能：计数操作期间缓冲器更新的时序图(TKCTOL0m = 0, TKCTOD0m = 0)



<1>: 在 TKBCEn 由“0”改变为“1”且 TKCNT0 开始计数操作时，比较寄存器的设置被传送至缓冲器寄存器。

<2>: 如果未执行向 TKCRDT0 写入“1”的操作，即使在 TKCCR0 和 TKCDUTY05 至 TKCDUTY00 寄存器被覆盖后执行了清除计数器，也不会产生覆盖。

<3>: 通过向 TKCRDT0 写入“1”，可以将批量覆盖保留标志 (TKCRSF0) 设置为“1”。

<4>: 在 TKCRSF0 = 1 且清除计数器时，比较寄存器的设置被传送至缓冲器寄存器。同时，TKCRSF0 成为“0”。

8.4.2 停止/重启操作

可以通过控制 TKCCE0 来进行 16 位定时器 KC0 的停止和重启操作。

- (1) 通过将 TKCCE0 从“1”改为“0”，可以使 16 位定时器 KC0 复位并停止操作。

计数器 TKCCNT0 被复位为 FFFFH，并停止操作。

TKCO0m 输出将输出由 TKCTOD0m 设置的默认电平。

- (2) 通过将 TKCCE0 从“0”改为“1”，可以使 16 位定时器 KC0 开始操作。

当 TKCCE0 = 0 时，计数器 TKCCNT0 保持为 FFFFH；将 TKCCE0 从“0”改为“1”时，开始计数操作。

TKCO0m 输出是根据 TKCTOE0m/TKCTOL0m/TKCTOD0m 的设置来输出波形。

图 8-21. 停止/重启操作的时序图 (TKCTOL0m = 0, TKBTOD0m = 0)

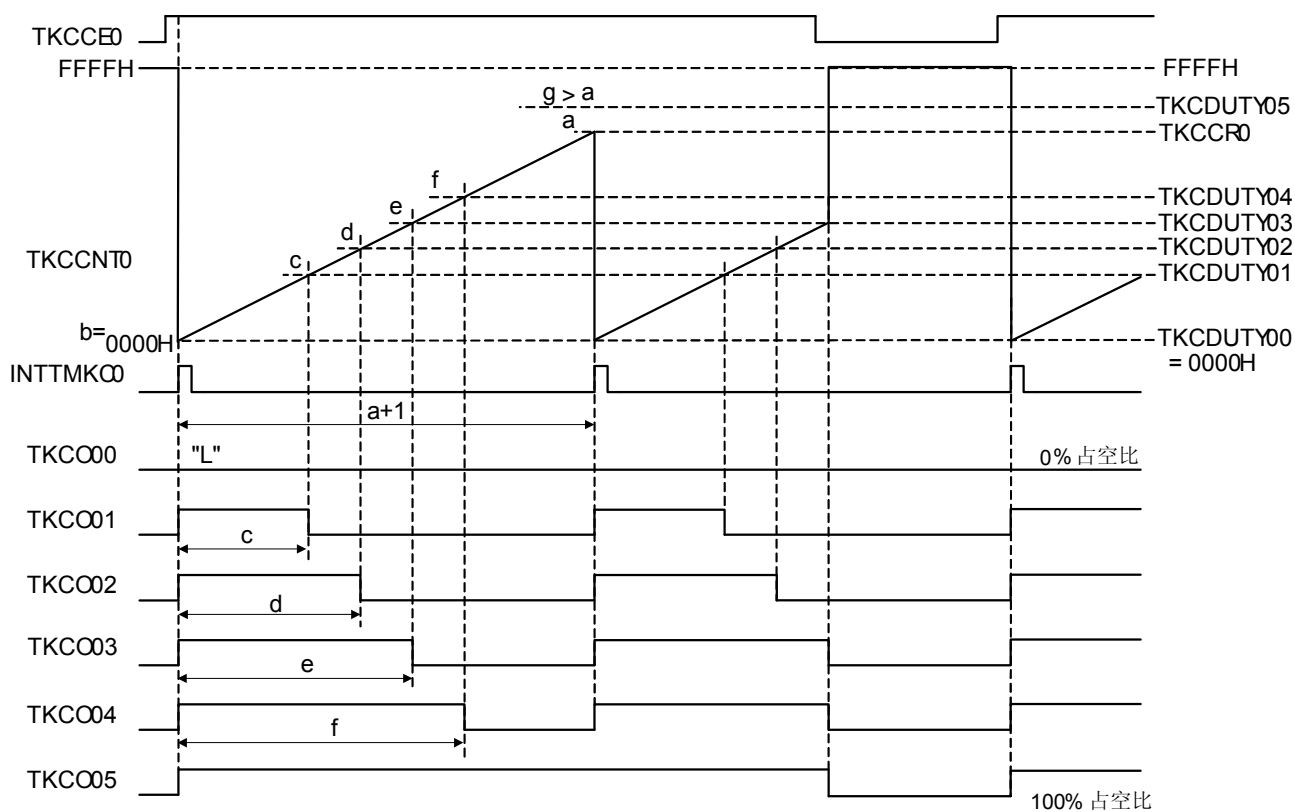
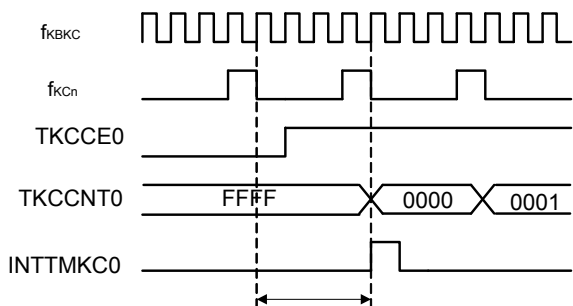
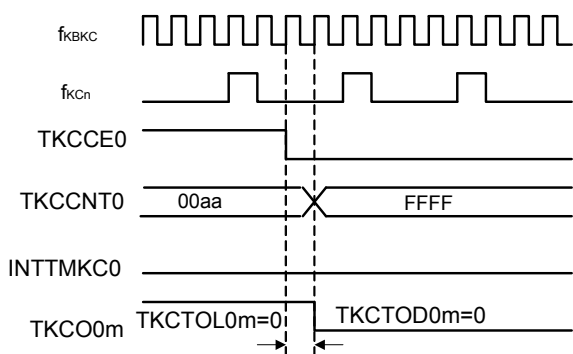


图 8-22. 计数操作开始时序

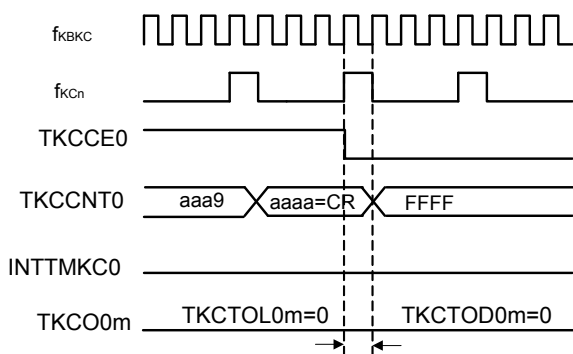


当 $TKBCE_n$ 从“0”更改为“1”时，在经过最小 1 个 f_{kBKC} 到最大 1 个 f_{kCn} 之后开始计数操作。
在计数操作开始时序输出 $INTTMK_{Bn}$ 。

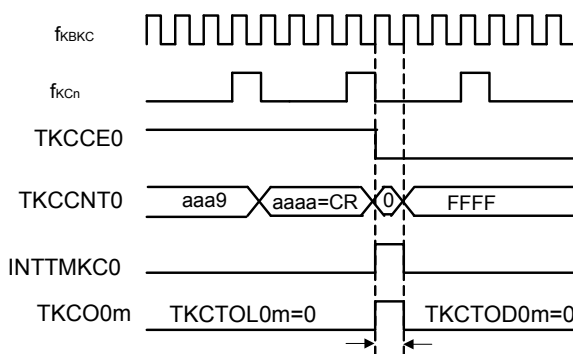
图 8-23. 计数操作停止时序



$TKCCE0$ 从“1”切换到“0”时，至少经过 1 f_{kBKC} 后，将停止计数操作。
 $TKCCNT0$ 将被复位到 FFFFH， $TKCO0m$ 将变为由 $TKCTOD0m$ 设置的默认电平。



$TKCCE0$ 从“1”切换到“0”时，至少经过 1 f_{kBKC} 后，将停止计数操作。
产生 1 f_{kCn} 前，即使产生 $TKCCNT0 = TKCCR0$ 的匹配，也不输出 $INTTMK_{C0}$ 信号。



$TKCCE0$ 操作与 1 f_{kCn} 同时产生时，通过产生 $TKCCNT0 = TKCCR0$ 的匹配而输出 $INTTMK_{C0}$ ，并且 $TKCO0m$ 将被设置。
经过 1 f_{kBKC} 后， $TKCCNT0$ 将被复位到 FFFFH， $TKCO0m$ 将变为由 $TKCTOD0m$ 设置的默认电平。

8.4.3 默认电平和有效电平

通过 16 位定时器 KC 输出控制寄存器 00 (TKCIOC00)，可以设置定时器 KC 输出的默认电平和有效电平。

当 TKCTOE0m 从“1”改变为“0”时，TKCO0m 将根据 TKCTOD0m 的设置输出默认电平。

当 TKCTOE0m 从“0”改变为“1”时，根据 TKCO0m 的设置条件/复位条件的产生以及 TKCTOL0m 的设置来输出 PWM 波形。

图 8-24. 在 PWM 复位时序(占空比匹配)之前 TKCTOE0m 由“0”改变为“1”时

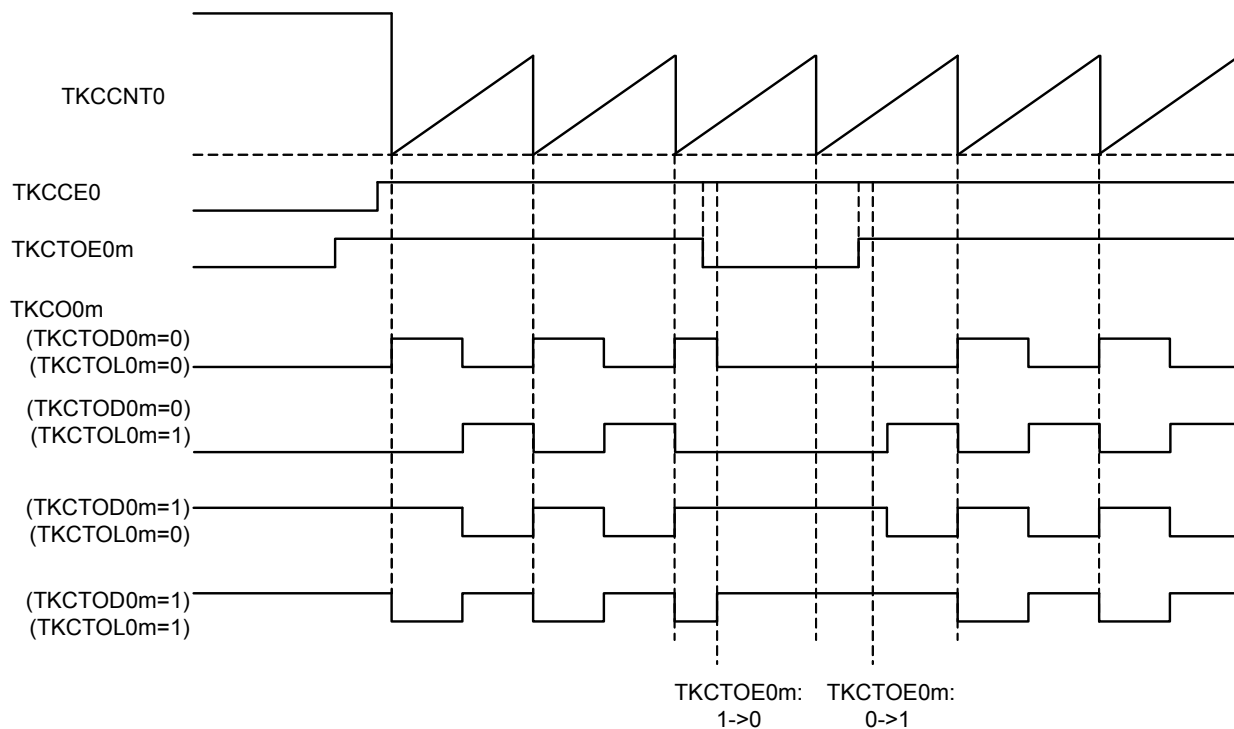


图 8-25. 在 PWM 复位时序(占空比匹配)之后 TKCTOE0m 由“0”改变为“1”时

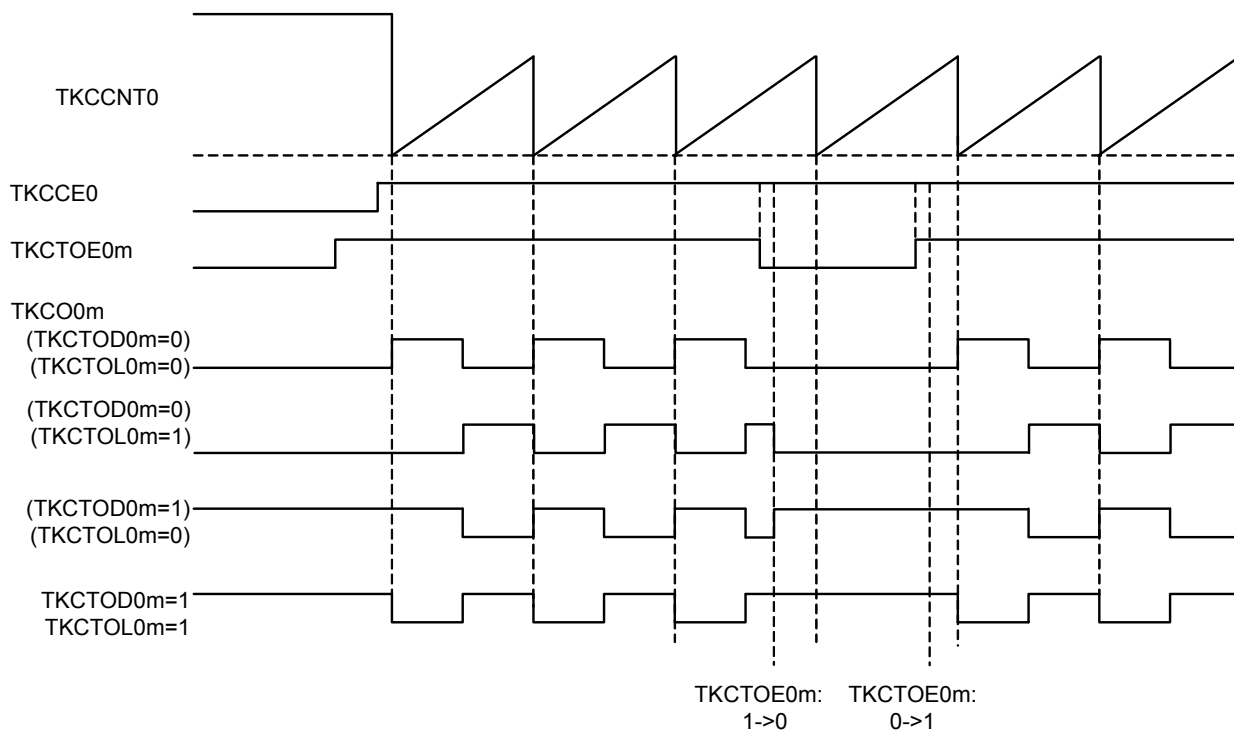
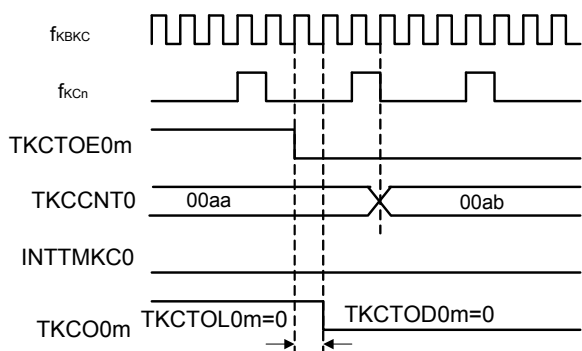
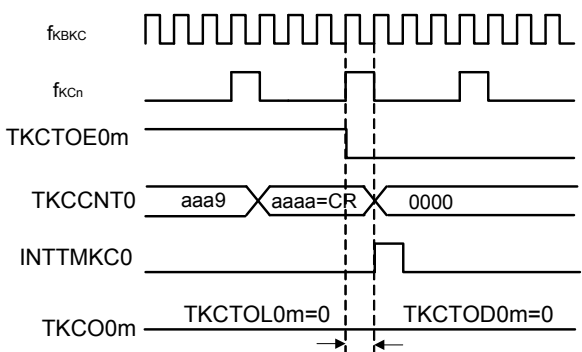


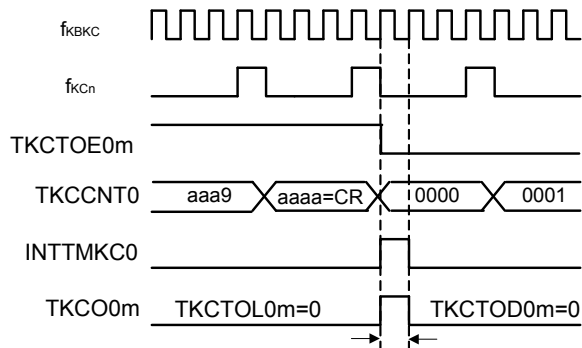
图 8-26. TKCTOE0m 由“1”改变为“0”时



TKCTOE0m 从 "1" 切换到 "0" 时, 经过 1 f_{KBKC} 后, TKCO0m 将变为由 TKCTOD0m 设置的默认电平。



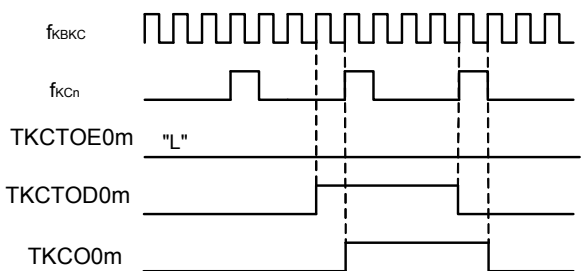
在 TKCCNT0 与 TKCCR0 相匹配而设置 TKCO0m (低电平至高电平) 的时序相同时, TKCTOE0m 将被优先变为由 TKCTOD0m 设置的默认电平。



由于 TKCTOE0m 操作与 1 f_{KCn} 的产生同时进行, 将产生 TKCCNT0 = TKCCR0 的匹配, 使 TKCO0m 被设置。

经过 1 f_{KBKC} 后, TKCO0m 将变为由 TKCTOD0m 设置的默认电平。

图 8-27. TKCTOE0m = 0 时更改 TKCTOD0m



若在 TKCTOE0m = 0 时更改 TKCTOD0m, 则经过 1 f_{KBKC} 后, TKCO0m 将变为由 TKCTOD0m 设置的默认电平。

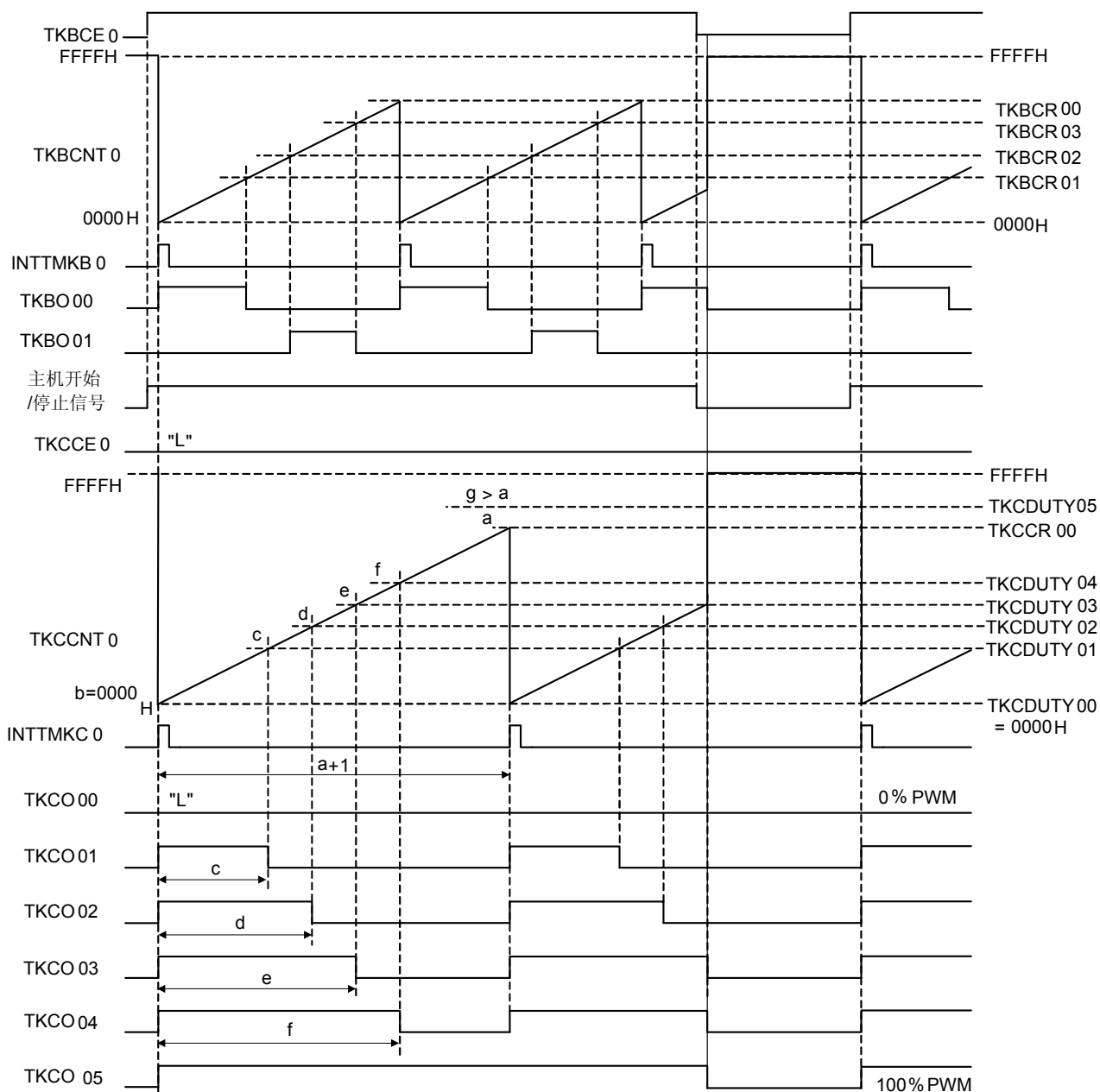
8.4.4 同时开始&停止模式

使用 16 位定时器 KB0/KC0 配置主/从属结构，从而通过与主定时器 KB0 的计数开始/停止操作同步，可以同时开始/停止从属定时器 KC0。

在这种情况下，为主选择“单机模式(TKBMDn1, TKBMDn0 = 0, 0)”，为从属选择“同时开始&停止模式(TKCMD0 = 1)”。

关于 TKCO05 至 TKCO00 输出的计算公式，请参阅 8.4.1 PWM 输出功能。

图 8-28. 同时开始&停止模式的基本时序例 (TKCTOL0m = 0, TKCTOD0m = 0)



第九章 实时时钟

9.1 实时时钟的功能

实时时钟具有如下功能。

- 具有年、月、星期、日、小时、分和秒计数器，并且可以计数到 99 年。
- 固定周期中断功能（周期：0.5 秒、1 秒、1 分、1 小时、1 日、1 个月）
- 报警中断功能（报警：星期、小时、分）（仅限 38 引脚产品）。
- 1 Hz 的引脚输出功能（仅限 38 引脚产品）。

注意事项 1. 仅当选择副系统时钟 ($f_{SUB} = 32.768 \text{ kHz}$) 作为实时时钟的操作时钟时，才可对年、月、星期、日、时、分和秒进行计数。当选择低速片上振荡器时钟 ($f_{IL} = 15 \text{ kHz}$) 时，仅能使用固定周期中断功能。20 至 32 引脚产品中，因为没有配置副系统时钟，所以只能使用固定周期中断功能。但是，选择 f_{IL} 时的固定周期中断间隔按以下方法计算：固定周期（用 RTCC0 寄存器选择的值） $\times f_{SUB}/f_{IL}$ 。

2. 本章的下述内容主要是以 38 引脚产品为例进行说明。

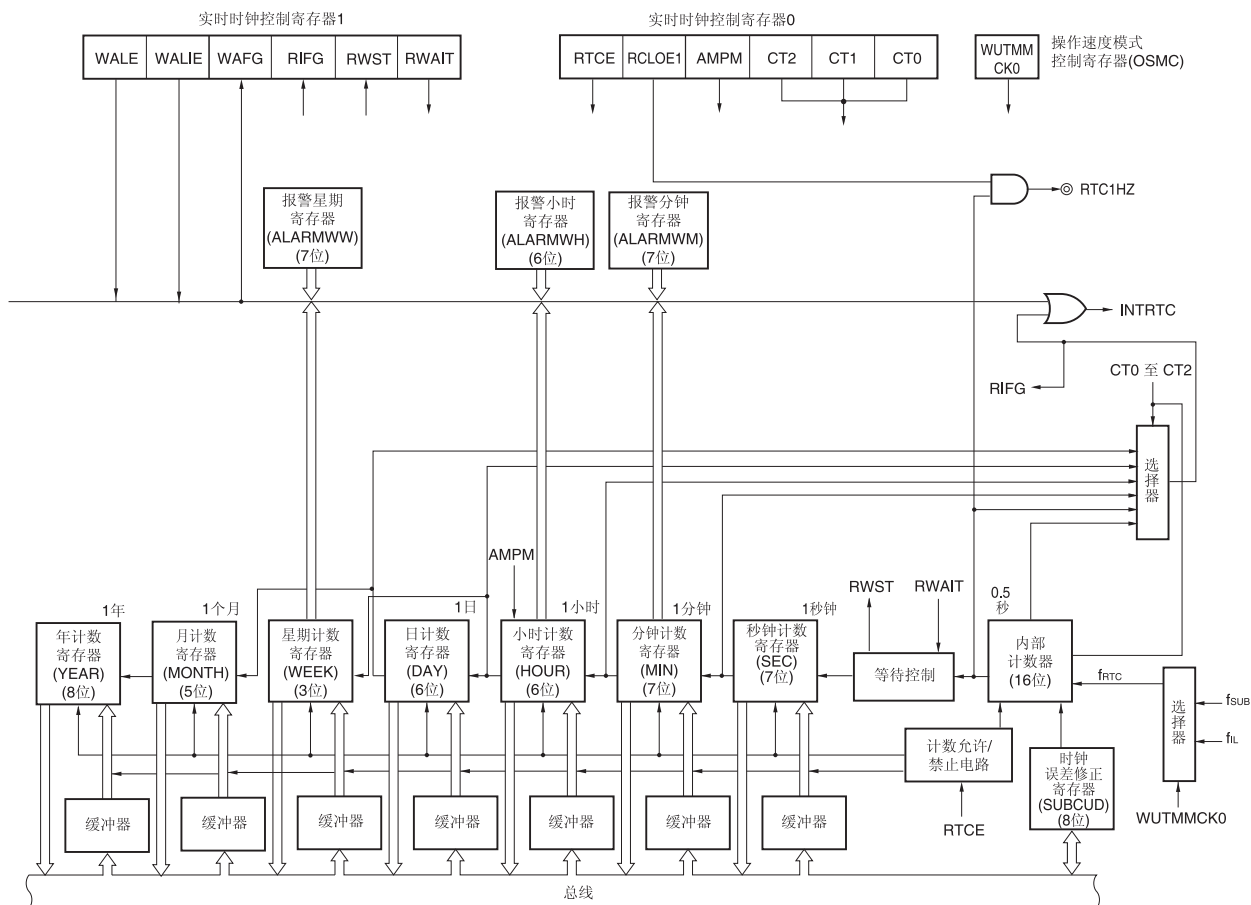
9.2 实时时钟的配置

实时时钟包括下列硬件。

表 9-1. 实时时钟的配置

项目	配置
计数器	计数器（16 位）
控制寄存器	外围允许寄存器 0 (PER0)
	操作速度模式控制寄存器(OSMC)
	实时时钟控制寄存器 0 (RTCC0)
	实时时钟控制寄存器 1 (RTCC1)
	秒计数寄存器(SEC)
	分钟计数寄存器(MIN)
	小时计数寄存器(HOUR)
	日计数寄存器(DAY)
	星期计数寄存器(WEEK)
	月计数寄存器(MONTH)
	年计数寄存器(YEAR)
	时钟误差修正寄存器(SUBCUD)
	报警分钟寄存器(ALARMWM)
报警小时寄存器(ALARMWH)	
报警星期寄存器(ALARMWW)	

图 9-1. 实时时钟的框图



注意事项 仅当选择副系统时钟 ($f_{SUB} = 32.768 \text{ kHz}$) 作为实时时钟的操作时钟时,才可对年、月、星期、日、时、分和秒进行计数。当选择低速片上振荡器时钟 ($f_{IL} = 15 \text{ kHz}$)时,仅能使用固定周期中断功能。20 至 32 引脚产品中,因为没有配置副系统时钟,所以只能使用固定周期中断功能。但是,选择 f_{IL} 时的固定周期中断间隔按以下方法计算: 固定周期 (用 $RTCC0$ 寄存器选择的值) $\times f_{SUB}/f_{IL}$ 。

9.3 控制实时时钟的寄存器

以下寄存器用于控制实时时钟。

- 外围允许寄存器 0 (PER0)
- 操作速度模式控制寄存器(OSMC)
- 实时时钟控制寄存器 0 (RTCC0)
- 实时时钟控制寄存器 1 (RTCC1)
- 秒计数寄存器(SEC)
- 分钟计数寄存器(MIN)
- 小时计数寄存器(HOUR)
- 日计数寄存器(DAY)
- 星期计数寄存器(WEEK)
- 月计数寄存器(MONTH)
- 年计数寄存器(YEAR)
- 时钟误差修正寄存器(SUBCUD)
- 报警分钟寄存器(ALARMWM)
- 报警小时寄存器(ALARMWH)
- 报警星期寄存器(ALARMWW)

(1) 外围允许寄存器 0 (PER0)

该寄存器用于允许或禁止向外围硬件供应时钟。通过停止向未使用的硬件供应时钟，可以降低功耗和噪声。

使用实时时钟时，必须将该寄存器的位 7(RTCEN)设置为 1。

使用 1 位或 8 位存储器操作指令设置 PER0 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 9-2. 外围允许寄存器 0 (PER0)的格式

地址: F00F0H 复位后: 00H R/W

符号	<7>	6	<5>	<4>	3	<2>	1	<0>
PER0	RTCEN	0	ADCEN	IICA0EN	0	SAU0EN	0	TAU0EN

RTCEN	控制实时时钟 (RTC) 和 12 位间隔定时器的输入时钟供应
0	停止输入时钟供应。 <ul style="list-style-type: none"> 不可写入用于实时时钟(RTC)和 12 位间隔定时器的 SFR。 实时时钟(RTC)和 12 位间隔定时器处于复位状态。
1	允许输入时钟供应。 <ul style="list-style-type: none"> 可以读取和写入用于实时时钟(RTC)和 12 位间隔定时器的 SFR。

- 注意事项**
- 使用实时时钟时，在输入时钟(f_{RTC})处于振荡稳定状态后，首先要将 RTCEN 位设置为 1。当 RTCEN = 0 时，对实时时钟或 12 位间隔定时器的控制寄存器进行的写入操作被忽略，而且读出的值总为初始值。
 - 通过将操作速度模式控制寄存器(OSMC)的 RTCLPC 位设置为 1，在 STOP 模式以及副系统时钟时的 HALT 模式下，可以停止除了实时时钟和 12 位间隔定时器以外的外围功能的副系统时钟的供应。此时，请将 PER0 寄存器的 RTCEN 位设置为 1，将其他位（位 0 至位 6）设置为 0。
 - 必须将位 1、3 和 6 清除为 0。

(2) 操作速度模式控制寄存器(OSMC)

可通过 WUTMMCK0 位选择实时时钟的操作时钟 (f_{RTC})。

另外，通过停止不必要的时钟功能，RTCLPC 位可以用于降低功耗。有关 RTCLPC 位的设置详情，请参阅第五章时钟发生电路。

使用 8 位存储器操作指令来设置 OSMC 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 9-3. 操作速度模式控制寄存器(OSMC)的格式

地址: F00F3H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
OSMC	RTCLPC	0	0	WUTMMCK0	0	0	0	0

WUTMMCK0	选择实时时钟和 12 位间隔定时器的操作时钟(f _{RTC})
0	副系统时钟 (f _{SUB})
1	低速片上振荡器时钟(f _{IL})

- 注意事项 1.** 仅当选择副系统时钟 (f_{SUB} = 32.768 kHz) 作为实时时钟的操作时钟时，才可对年、月、星期、日、时、分和秒进行计数。当选择低速片上振荡器时钟(f_{IL} = 15 kHz)时，仅能使用固定周期中断功能。20 至 32 引脚产品中，因为没有配置副系统时钟，所以只能使用固定周期中断功能。但是，选择 f_{IL} 时的固定周期中断间隔按以下方法计算：固定周期（用 RTCC0 寄存器选择的值）× f_{SUB}/f_{IL}。
- 2.** 本章的下述内容主要是以 38 引脚产品为例进行说明。

(3) 实时时钟控制寄存器 0 (RTCC0)

RTCC0 寄存器是 8 位寄存器，用于开始或停止实时时钟操作，控制 RTC1HZ 引脚，设置 12 小时或 24 小时系统，以及设置固定周期中断功能。

使用 1 位或 8 位存储器操作指令设置 RTCC0 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 9-4. 实时时钟控制寄存器 0 (RTCC0)的格式

地址: FFF9DH 复位后: 00H R/W

符号	<7>	6	<5>	4	3	2	1	0
RTCC0	RTCE	0	RCLOE1	0	AMPM	CT2	CT1	CT0

RTCE	实时时钟的操作控制
0	停止计数器操作。
1	开始计数操作。

RCLOE1	RTC1HZ引脚的输出控制
0	禁止 RTC1HZ 引脚的输出 (1 Hz)。
1	允许 RTC1HZ 引脚的输出 (1 Hz)。

AMPM	12/24小时系统的选择
0	12 小时系统 (显示 a.m. 和 p.m.。)
1	24 小时系统
<ul style="list-style-type: none"> 在将 RWAIT 位(实时时钟控制寄存器 1(RTCC1)的位 0)设置为 1 之后改写 AMPM 位的值。如果改变 AMPM 位的值，则小时计数寄存器(HOUR)的值将会随着所设置的时间系统的值而改变。 所显示的时间数字如表 9-2 所示。 	

CT2	CT1	CT0	固定周期中断(INTRTC)的选择	
			选择副系统时钟(f_{sub})	选择低速片上振荡器时钟(f_{IL})
0	0	0	不使用固定周期中断功能。	
0	0	1	0.5 秒一次 (与秒累加同步)	1.1 秒一次
0	1	0	1 秒一次 (与秒累加同步)	2.2 秒一次
0	1	1	1 分钟一次 (每分钟的 00 秒)	2.2 分钟一次
1	0	0	1 小时一次 (每小时的 00 分钟 00 秒)	2.2 小时一次
1	0	1	1 日一次 (每日的 00 点 00 分 00 秒)	2.2 天一次
1	1	×	1 个月一次 (每月的 1 日上午 00 点 00 分 00 秒)	2.2 个月一次

在计数器计数期间(RTCE = 1)要改变 CT2 至 CT0 位的值时，使用中断屏蔽标志寄存器将 INTRTC 设置为禁止中断处理之后，再改写 CT2 至 CT0 位的值。另外，改写 CT2 至 CT0 位的值之后，先清除 RIFG 和 RTCIF 标志再允许中断处理。

注意事项 当 RTCE = 1 时不要改变 RTCLOE1 位的值。

备注 ×: 忽略

(4) 实时时钟控制寄存器 1 (RTCC1)

RTCC1 寄存器是 8 位寄存器，用于控制报警中断功能和计数器等待。

使用 1 位或 8 位存储器操作指令设置 RTCC1 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 9-5. 实时时钟控制寄存器 1 (RTCC1)的格式(1/2)

地址: FFF9EH 复位后: 00H R/W

符号	<7>	<6>	5	<4>	<3>	2	<1>	<0>
RTCC1	WALE	WALIE	0	WAFG	RIFG	0	RWST	RWAIT

WALE	报警的操作控制
0	一致操作无效。
1	一致操作有效。
在计数器计数期间(RTCE = 1)且 WALIE = 1 的情况下要设置 WALE 位的值时，使用中断屏蔽标志寄存器将 INTRTC 设置为禁止中断处理之后，再改写 WALE 位。另外，改写 WALE 位之后，须清除 WAFG 和 RTCIF 标志。在设置各个报警寄存器（实时时钟控制寄存器 1(RTCC1)的 WALIE 标志、报警分寄存器(ALARMWM)、报警小时寄存器 (ALARMWH) 和报警星期寄存器(ALARMWW)的 WALIE 标志) 时，将 WALE 位设置为一致操作无效(0)。	

WALIE	报警中断(INTRTC) 功能的操作控制
0	报警一致时不产生中断。
1	报警一致时产生中断。

WAFG	报警检测状态标志
0	未检测到报警一致
1	检测到报警一致
用于表示检测到报警一致的状态标志。仅当 WALE = 1 时有效，并在检测到报警一致并经过一个时钟(32.768 kHz)时被设置为 1。当写入 0 时，该标志被清除。写入 1 无效。	

RIFG	固定周期中断状态标志
0	未产生固定周期中断。
1	产生固定周期中断。
该标志表示产生固定周期中断的状态。产生固定周期中断时，该标志被设置为 1。当写入 0 时，该标志被清除。写入 1 无效。	

图 9-5. 实时时钟控制寄存器 1 (RTCC1)的格式(2/2)

RWST	实时时钟的等待状态标志
0	计数器操作中。
1	计数器值的读取或写入模式中。
该状态标志表示 RWAIT 位的设置是否有效。 对计数器值进行读取或写入之前，请先确认该标志的值为 1。	
RWAIT	实时时钟的等待控制
0	设置计数器操作。
1	停止 SEC 至 YEAR 计数器。计数器值的读取或写入模式中。
该位控制计数器的操作。 读取或写入计数器值时必须向其写入 1。 因为计数器（16 位）计数在继续，请在 1 秒内完成读取或写入，然后返回该位的设置为 0。 设置 RWAIT = 1 后，最多经过 1 个时钟(f _{RTC})，可以读取或写入计数器值(RWST = 1)。 在 RWAIT = 1 的情况下计数器（16 位）发生溢出时，溢出事件将保持，直到 RWAIT = 0 后递增计数。 但是，对秒计数寄存器进行写入操作的情况下，溢出事件不保持。	

注意事项 如果用 1 位操作指令对 RTCC1 寄存器进行写操作，则可能清除掉 RIFG 和 WAFG 标志。因此，若要对 RTCC1 寄存器进行写操作，必须使用 8 位操作指令。为了避免在写操作期间清除掉 RIFG 和 WAFG 标志，请将相应位设置为 1 (写入无效)。另外，如果不使用 RIFG 和 WAFG 标志且值可以改写的情况下，也可以用位操作指令对 RTCC1 寄存器进行写操作。

备注 固定周期中断和报警一致中断使用相同的中断源(INTRTC)。当同时使用这两种中断时，可在发生 INTRTC 时通过确认固定周期中断状态标志(RIFG)和报警检测状态标志(WAFG)来判断发生的是哪种中断。

(5) 秒计数寄存器(SEC)

SEC 寄存器是 8 位寄存器，它的值为 0 至 59（十进制），表示秒的计数值。

当计数器（16 位）溢出时，该计数器递增计数。

当数据写入该寄存器时，数据被写入缓冲器，并在最多 2 个时钟(f_{RTC})后被写入计数器。以 BCD 码格式设置 00 至 59 的十进制数到该寄存器。

使用 8 位存储器操作指令来设置 SEC 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 9-6. 秒计数寄存器(SEC)的格式

地址: FFF92H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
SEC	0	SEC40	SEC20	SEC10	SEC8	SEC4	SEC2	SEC1

注意事项 计数器操作期间(RTCE = 1)读取或写入寄存器时，请依据 9.4.3 读/写实时时钟中所示步骤进行。

(6) 分钟计数寄存器(MIN)

MIN 寄存器是 8 位寄存器，它的值为 0 至 59（十进制），表示分钟的计数值。

当秒计数器溢出时，该计数器递增计数。

当数据写入该寄存器时，数据被写入缓冲器，并在最多 2 个时钟(f_{RTC})后被写入计数器。在对该寄存器进行写操作时，即使秒计数寄存器发生溢出也将其忽略，并设置为写入值。以 BCD 码格式设置 00 至 59 的十进制数到该寄存器。

使用 8 位存储器操作指令来设置 MIN 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 9-7. 分钟计数寄存器(MIN)的格式

地址: FFF93H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
最小值	0	MIN40	MIN20	MIN10	MIN8	MIN4	MIN2	MIN1

注意事项 计数器操作期间(RTCE = 1)读取或写入寄存器时，请依据 9.4.3 读/写实时时钟中所示步骤进行。

(7) 小时计数寄存器(HOUR)

HOUR 寄存器是 8 位寄存器，它的值为 00 至 23 或 01 至 12 和 21 至 32（十进制），表示小时的计数值。

当分钟计数器溢出时，该计数器递增计数。

当数据写入该寄存器时，数据被写入缓冲器，并在最多 2 个时钟(f_{RTC})后被写入计数器。在对该寄存器进行写入操作时，即使分钟计数寄存器发生溢出也将其忽略，并设置为写入值。对应于用实时时钟控制寄存器 0(RTCC0)的位 3(AMPM)指定的时间系统，以 BCD 码格式设置一个以下范围内的十进制值：00 至 23、01 至 12 和 21 至 32。

如果改变 AMPM 位的值，则 HOUR 寄存器的值将会随着所设置的时间系统的值而改变。

使用 8 位存储器操作指令来设置 HOUR 寄存器。

产生复位信号后，该寄存器被设置为 12H。

但是，如果复位后设置 AMPM 位 (RTCC0 寄存器的位 3) 为 1，则该寄存器的值为 00H。

图 9-8. 小时计数寄存器(HOUR)的格式

地址: FFF94H 复位后: 12H R/W

符号	7	6	5	4	3	2	1	0
HOUR	0	0	HOUR20	HOUR10	HOUR8	HOUR4	HOUR2	HOUR1

- 注意事项**
1. 当选择 AMPM = 0 (如果选择 12 小时系统)时，HOUR 寄存器的第 5 位表示 AM(0)/PM(1)。
 2. 计数器操作期间(RTCE = 1)读取或写入寄存器时，请依据 9.4.3 读/写实时时钟中所示步骤进行。

表 9-2 列出了 AMPM 位设置值、小时计数寄存器(HOUR)寄存器值和时间之间的关系。

表 9-2. 时间数字显示

24 小时显示(AMPM = 1)		12 小时显示(AMPM = 0)	
时间	HOUR 寄存器	时间	HOUR 寄存器
0	00H	12 a.m.	12H
1	01H	1 a.m.	01H
2	02H	2 a.m.	02H
3	03H	3 a.m.	03H
4	04H	4 a.m.	04H
5	05H	5 a.m.	05H
6	06H	6 a.m.	06H
7	07H	7 a.m.	07H
8	08H	8 a.m.	08H
9	09H	9 a.m.	09H
10	10H	10 a.m.	10H
11	11H	11 a.m.	11H
12	12H	12 p.m.	32H
13	13H	1 p.m.	21H
14	14H	2 p.m.	22H
15	15H	3 p.m.	23H
16	16H	4 p.m.	24H
17	17H	5 p.m.	25H
18	18H	6 p.m.	26H
19	19H	7 p.m.	27H
20	20H	8 p.m.	28H
21	21H	9 p.m.	29H
22	22H	10 p.m.	30H
23	23H	11 p.m.	31H

当 AMPM 位为 0 时，HOUR 寄存器的值以 12 小时显示，当 AMPM 位为“1”时，以 24 小时显示。在 12 小时显示中，HOUR 寄存器的第 5 位为 0 表示上午（AM），为 1 表示下午（PM）。

(8) 日计数寄存器(DAY)

DAY 寄存器是 8 位寄存器，它的值为 1 至 31（十进制），表示日的计数值。

当小时计数器溢出时，该计数器递增计数。

计数器计数如下。

- 01 至 31 (一月、三月、五月、七月、八月、十月、十二月)
- 01 至 30 (四月、六月、九月、十一月)
- 01 至 29 (闰年的二月)
- 01 至 28 (平年的二月)

当数据写入该寄存器时，数据被写入缓冲器，并在最多 2 个时钟(f_{RTC})后被写入计数器。在对该寄存器进行写入操作时，即使小时计数寄存器发生溢出也将其忽略，并设置为写入值。以 BCD 码格式设置 01 至 31 的十进制数到该寄存器。

使用 8 位存储器操作指令来设置 DAY 寄存器。

产生复位信号后，该寄存器被设置为 01H。

图 9-9. 日计数寄存器(DAY)的格式

地址: FFF96H 复位后: 01H R/W

符号	7	6	5	4	3	2	1	0
DAY	0	0	DAY20	DAY10	DAY8	DAY4	DAY2	DAY1

注意事项 计数器操作期间(RTCE = 1)读取或写入寄存器时，请依据 9.4.3 读/写实时时钟中所示步骤进行。

(9) 星期计数寄存器(WEEK)

WEEK 寄存器是 8 位寄存器，它的值为 0 至 6（十进制），表示星期的计数值。

当日计数器溢出时，该计数器递增计数。

当数据写入该寄存器时，数据被写入缓冲器，并在最多 2 个时钟(f_{RTC})后被写入计数器。以 BCD 码格式设置 00 至 06 的十进制数到该寄存器。

使用 8 位存储器操作指令来设置 WEEK 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 9-10. 星期计数寄存器(WEEK)的格式

地址: FFF95H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
WEEK	0	0	0	0	0	WEEK4	WEEK2	WEEK1

注意事项 1. 月计数寄存器(MONTH)及日计数寄存器(DAY)的对应值不会自动存储到星期寄存器(WEEK)。解除复位后，将星期计数寄存器设置为如下所示。

星期	WEEK
星期日	00H
星期一	01H
星期二	02H
星期三	03H
星期四	04H
星期五	05H
星期六	06H

2. 计数器操作期间($RTCE = 1$)读取或写入寄存器时，请依据 9.4.3 读/写实时时钟中所示步骤进行。

(10) 月计数寄存器(MONTH)

MONTH 寄存器是 8 位寄存器，它的值为 1 至 12（十进制），表示月的计数值。

当日计数器溢出时，该计数器递增计数。

当数据写入该寄存器时，数据被写入缓冲器，并在最多 2 个时钟(f_{RTC})后被写入计数器。在对该寄存器进行写入操作时，即使日计数寄存器发生溢出也将其忽略，并设置为写入值。以 BCD 码格式设置 01 至 12 的十进制数到该寄存器。

使用 8 位存储器操作指令来设置 MONTH 寄存器。

产生复位信号后，该寄存器被设置为 01H。

图 9-11. 月计数寄存器(MONTH)的格式

地址: FFF97H 复位后: 01H R/W

符号	7	6	5	4	3	2	1	0
MONTH	0	0	0	MONTH10	MONTH8	MONTH4	MONTH2	MONTH1

注意事项 计数器操作期间(RTCE = 1)读取或写入寄存器时，请依据 9.4.3 读/写实时时钟中所示步骤进行。

(11) 年计数寄存器(YEAR)

YEAR 寄存器是 8 位寄存器，它的值为 0 至 99（十进制），表示年的计数值。

当月计数器溢出时，该计数器递增计数。

00、04、08、...、92 和 96 表示闰年。

当数据写入该寄存器时，数据被写入缓冲器，并在最多 2 个时钟(f_{RTC})后被写入计数器。在对该寄存器进行写入操作时，即使 MONTH 寄存器发生溢出也将其忽略，并设置为写入值。以 BCD 码格式设置 00 至 99 的十进制数到该寄存器。

使用 8 位存储器操作指令来设置 YEAR 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 9-12. 年计数寄存器(YEAR)的格式

地址: FFF98H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
YEAR	YEAR80	YEAR40	YEAR20	YEAR10	YEAR8	YEAR4	YEAR2	YEAR1

注意事项 计数器操作期间(RTCE = 1)读取或写入寄存器时，请依据 9.4.3 读/写实时时钟中所示步骤进行。

(12) 时钟误差修正寄存器(SUBCUD)

该寄存器用于高精度修正时钟，当时钟慢或快时，通过改变从计数器（16 位）到秒计数器(SEC)的溢出值（参考值：7FFFH），来修正时钟。

使用 8 位存储器操作指令来设置 SUBCUD 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 9-13. 时钟误差修正寄存器(SUBCUD)的格式

地址: FFF99H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
SUBCUD	DEV	F6	F5	F4	F3	F2	F1	F0

DEV	时钟误差修正时序的设置
0	在秒位为 00、20 或 40 秒时(每 20 秒)修正时钟误差。
1	仅在秒位为 00 秒时(每 60 秒)修正时钟误差。
禁止在以下时序写入 SUBCUD 寄存器。	
<ul style="list-style-type: none"> • 设置 DEV = 0 时: SEC = 00H、20H、40H 的期间 • 设置 DEV = 1 时: SEC = 00H 的期间 	

F6	时钟误差修正值的设置
0	仅增加 $\{(F5, F4, F3, F2, F1, F0) - 1\} \times 2$ 。
1	仅减少 $\{/(F5, /F4, /F3, /F2, /F1, /F0) + 1\} \times 2$ 。
当(F6, F5, F4, F3, F2, F1, F0) = (*, 0, 0, 0, 0, 0, *)时，不进行时钟误差的修正。*是 0 或 1。 /F5 至/F0 为对应位的反转值(当 111100 时为 000011)。	
修正值的范围: (当 F6 = 0 时) 2, 4, 6, 8, ..., 120, 122, 124 (当 F6 = 1 时) -2, -4, -6, -8, ..., -120, -122, -124	

能够通过时钟误差修正寄存器(SUBCUD)来进行修正的范围如下所示。

	DEV = 0 (每20秒修正)	DEV = 1 (每60秒修正)
可修正范围	-189.2 ppm 至 189.2 ppm	-63.1 ppm 至 63.1 ppm
最大量化误差	±1.53 ppm	±0.51 ppm
最小分辨率	±3.05 ppm	±1.02 ppm

备注 如果可修正范围小于或等于 -63.1 ppm，大于或等于 63.1 ppm 时，请设置 DEV= 0。

(13) 报警分钟寄存器(ALARMWM)

该寄存器用于设置报警的分钟。

使用 8 位存储器操作指令来设置 ALARMWM 寄存器。

产生复位信号后，该寄存器被清除为 00H。

注意事项 以 BCD 码格式设置 00 至 59 的十进制数到该寄存器。如果设置的值超出范围，则不检测该报警。

图 9-14. 报警分钟寄存器(ALARMWM)的格式

地址: FFF9AH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ALARMWM	0	WM40	WM20	WM10	WM8	WM4	WM2	WM1

(14) 报警小时寄存器(ALARMWH)

该寄存器用于设置报警的小时。

使用 8 位存储器操作指令来设置 ALARMWH 寄存器。

产生复位信号后，该寄存器被设置为 12H。

但是，如果复位后设置 AMPM 位（RTCC0 寄存器的位 3）为 1，则该寄存器的值为 00H。

注意事项 以 BCD 码设置 00 至 23 或 01 至 12 和 21 至 32 到该寄存器。如果设置的值超出范围，则不检测该报警。

图 9-15. 报警小时寄存器(ALARMWH)的格式

地址: FFF9BH 复位后: 12H R/W

符号	7	6	5	4	3	2	1	0
ALARMWH	0	0	WH20	WH10	WH8	WH4	WH2	WH1

注意事项 当选择 AMPM = 0 (如果选择 12 小时系统)时，ALARMWH 寄存器的第 5 位(WH20)表示 AM(0)/PM(1)。

(15) 报警星期寄存器(ALARMWW)

该寄存器用于设置报警的星期。

使用 8 位存储器操作指令来设置 ALARMWW 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 9-16. 报警星期寄存器(ALARMWW)的格式

地址: FFF9CH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ALARMWW	0	WW6	WW5	WW4	WW3	WW2	WW1	WW0

报警时间的设置示例如下所示。

设置的报警时间	星期							12 小时显示				24 小时显示			
	星期日	星期一	星期二	星期三	星期四	星期五	星期六	小时 10	小时 1	分钟 10	分钟 1	小时 10	小时 1	分钟 10	分钟 1
	W W 0	W W 1	W W 2	W W 3	W W 4	W W 5	W W 6								
每天, 00:00:00 a.m.	1	1	1	1	1	1	1	1	2	0	0	0	0	0	0
每天, 01:30:00 a.m.	1	1	1	1	1	1	1	0	1	3	0	0	1	3	0
每天, 11:59 a.m.	1	1	1	1	1	1	1	1	1	5	9	1	1	5	9
星期一至星期五, 0:00 p.m.	0	1	1	1	1	1	0	3	2	0	0	1	2	0	0
星期日, 1:30 p.m.	1	0	0	0	0	0	0	2	1	3	0	1	3	3	0
星期一、星期三、星期五, 11:59 p.m.	0	1	0	1	0	1	0	3	1	5	9	2	3	5	9

9.4 实时时钟的操作

9.4.1 开始实时时钟的操作

图 9-17. 开始实时时钟的操作步骤



- 注 1. 当输入时钟(f_{RTC})处于振荡稳定状态后，首先要将 RTCEN 位设置为 1。
2. 仅在必须修正时钟误差时设置 SUBCUD 寄存器。关于修正值的计算方法，请参阅 9.4.6 实时时钟的时钟误差修正示例。
3. 在 RTCE = 1 之后，不等到 INTRTC = 1 就切换至 HALT/STOP 模式时，请参阅 9.4.2 开始操作后切换至 HALT/STOP 模式中所示的步骤。

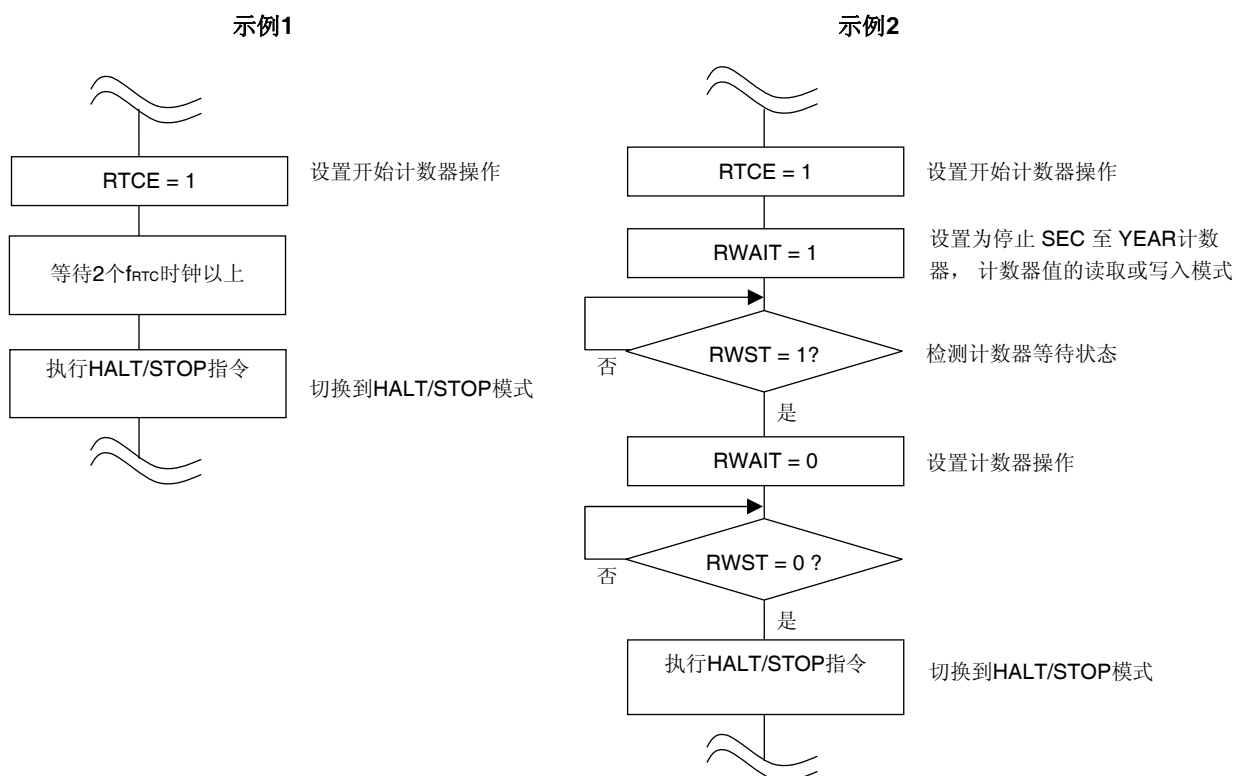
9.4.2 开始操作后切换至HALT/STOP模式

将 RTCE 位设置为 1 之后立即切换至 HALT/STOP 模式时，须执行以下处理中的一个。

但是，将 RTCE 位设置为 1 之后，如果在 INTRTC 中断发生后切换至 HALT/STOP 模式时，无需执行该处理。

- 将 RTCE 位设置为 1 并且至少经过两个输入时钟(f_{RTC})之后切换至 HALT/STOP 模式 (参阅图 9-18 示例 1)。
- 在将 RTCE 位设置为 1 之后，将 RWAIT 位设置为 1 并通过轮询确认 RWST 位被置为 1。之后，将 RWAIT 位设置为 0 并再次通过轮询确认 RWST 位被置为 0，然后切换至 HALT/STOP 模式(参阅图 9-18 示例 2)。

图 9-18. 将 RTCE 位设置为 1 之后切换至 HALT/STOP 模式时的步骤



9.4.3 读/写实时时钟

读取或写入计数器之前，先将 **RWAIT** 设置为 1。

读取或写入计数器之后，将 **RWAIT** 设置为 0。

图 9-19. 读取实时时钟的步骤



注 设置 STOP 模式之前，请确保 **RWST = 0**。

注意事项 请在 1 秒内完成置 **RWAIT** 位为 1 到清 **RWAIT** 位为 0 的系列处理。

备注 可以以任意顺序读取秒计数寄存器(**SEC**)、分钟计数寄存器(**MIN**)、小时计数寄存器(**HOUR**)、星期计数寄存器(**WEEK**)、日计数寄存器(**DAY**)、月计数寄存器(**MONTH**)和年计数寄存器(**YEAR**)。
无需读取所有寄存器，可以仅读取一部分寄存器。

图 9-20. 写入实时时钟的步骤



注 设置 STOP 模式之前，请确保 RWST = 0。

注意事项 1. 请在 1 秒内完成置 RWAIT 位为 1 到清 RWAIT 位为 0 的系列处理。

2. 在计数器计数期间(RTCE = 1)要改写 SEC、MIN、HOUR、WEEK、DAY、MONTH 和 YEAR 寄存器的值时，使用中断屏蔽标志寄存器将 INTRTC 设置为禁止中断处理之后，再进行改写。此外，进行改写后，清除 WAFG、RIFG 和 RTCIF 标志。

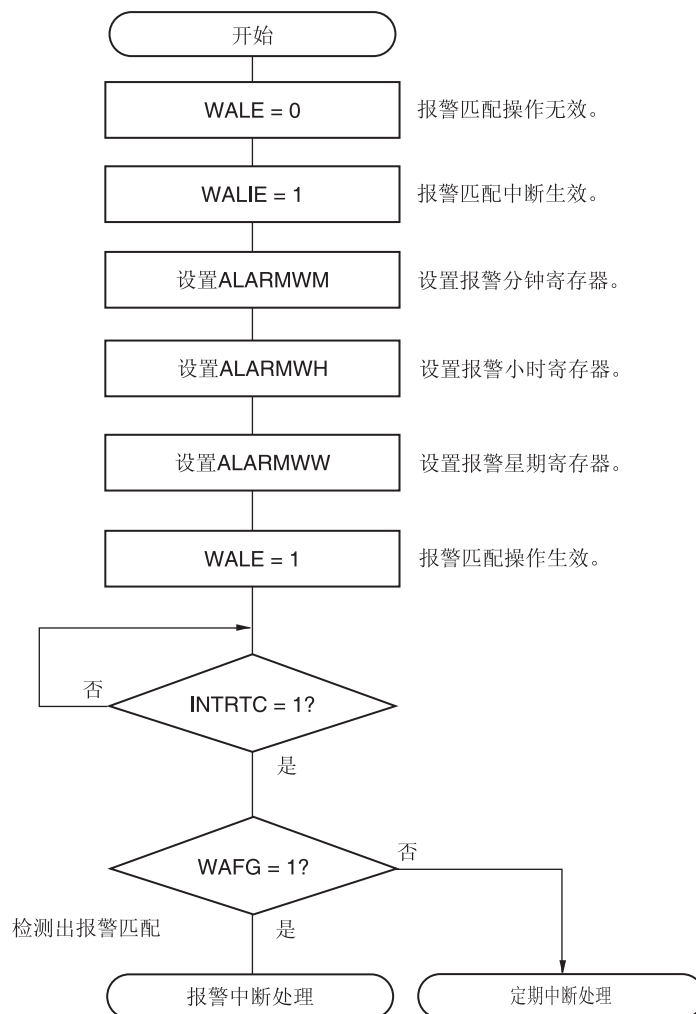
备注 可以以任意顺序写入秒计数寄存器(SEC)、分钟计数寄存器(MIN)、小时计数寄存器(HOUR)、星期计数寄存器(WEEK)、日计数寄存器(DAY)、月计数寄存器(MONTH)和年计数寄存器(YEAR)。

无需设置所有寄存器，可以仅读取一部分寄存器。

9.4.4 设置实时时钟的报警

先将 WALE 设置为 0（报警操作无效）之后再行报警时间的设置。

图 9-21. 报警处理步骤

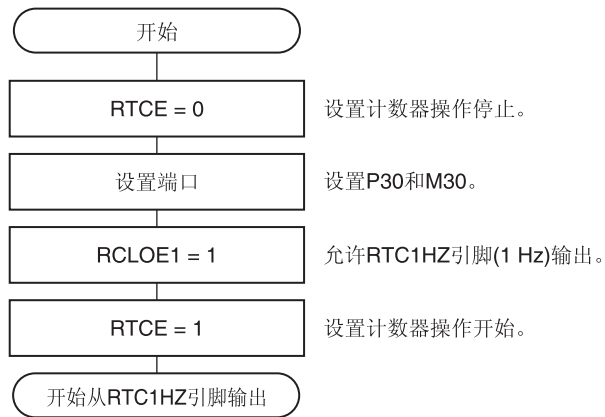


备注 1. 可以任意顺序写入报警分钟寄存器 (ALARMWMM)、报警小时寄存器 (ALARMWH) 和报警星期寄存器 (ALARMWW)。

2. 固定周期中断和报警一致中断使用相同的中断源 (INTRTC)。当同时使用这两种中断时，可在发生 INTRTC 时通过确认固定周期中断状态标志 (RIFG) 和报警检测状态标志 (WAFG) 来判断发生的是哪种中断。

9.4.5 实时时钟的 1 Hz 输出

图 9-22. 1 Hz 输出的设置步骤



注意事项 当输入时钟(f_{SUB})处于振荡稳定状态后，首先要将 **RTCEN** 位设置为 1。

9.4.6 实时时钟的时钟误差修正示例

当时钟慢或快时，可以通过对时钟误差修正寄存器设置值来实现对时钟的高精度修正。

修正值的计算示例

修正计数器(16位)的计数值时所使用的修正值通过以下表达式计算。

当修正范围小于或等于 -63.1 ppm ，大于或等于 63.1 ppm 时，将 DEV 位设置为 0。

(当 DEV = 0 时)

$$\text{修正值}^{\#} = 1 \text{ 分钟内的修正计数量} \div 3 = (\text{振荡频率} \div \text{目标频率} - 1) \times 32768 \times 60 \div 3$$

(当 DEV = 1 时)

$$\text{修正值}^{\#} = 1 \text{ 分钟内的修正计数量} = (\text{振荡频率} \div \text{目标频率} - 1) \times 32768 \times 60$$

注 修正值为通过时钟误差修正寄存器(SUBCUD)的位 6 至位 0 的值计算得出的时钟误差修正值。

(当 F6 = 0 时) 修正值 = $\{(F5, F4, F3, F2, F1, F0) - 1\} \times 2$

(当 F6 = 1 时) 修正值 = $-\{(/F5, /F4, /F3, /F2, /F1, /F0) + 1\} \times 2$

当(F6, F5, F4, F3, F2, F1, F0)为(*, 0, 0, 0, 0, 0, *)时，不执行时钟误差修正。“*”为 0 或 1。

/F5 至 /F0 为位反转值(当 111100 时为 000011)。

备注 1. 修正值是 2, 4, 6, 8, 到 120, 122, 124 或 -2, -4, -6, -8, 到 -120, -122, -124。

2. 振荡频率为输入时钟(f_{RTC})的值。

可以在时钟误差修正寄存器被设置为初始值(00H)时通过 RTC1HZ 引脚的输出频率 $\times 32768$ 来计算得到。

3. 目标频率为使用时钟误差修正寄存器进行修正后得到的频率。

修正示例

从 32767.4 Hz 修正至 32768 Hz (32767.4 Hz + 18.3 ppm)的示例

[测量振荡频率]

当时钟误差修正寄存器(SUBCUD)被设置为初始值(00H)时, 通过从 RTC1HZ 引脚输出大约 1Hz 的信号, 即可测得各产品的振荡频率[※]。

注 从 RTC1HZ 引脚输出大约 1Hz 的信号的设置步骤请参阅 9.4.5 实时时钟的 1 Hz 输出。

[计算修正值]

(当 RTC1HZ 引脚的输出频率为 0.9999817 Hz 时)

振荡频率= 32768 × 0.9999817 ≈ 32767.4 Hz

假设目标频率为 32768 Hz (32767.4 Hz + 18.3 ppm)且 DEV 为 1。

当 DEV 为 1 时用于计算修正值的表达式是适用的。

$$\begin{aligned} \text{修正值} &= 1 \text{ 分钟内的修正计数量} \\ &= (\text{振荡频率} \div \text{目标频率} - 1) \times 32768 \times 60 \\ &= (32767.4 \div 32768 - 1) \times 32768 \times 60 \\ &= -36 \end{aligned}$$

[计算(F6 至 F0)的设置值]

(当修正值为-36 时)

当修正值小于等于 0 (加快)时, 假设 F6 为 1。

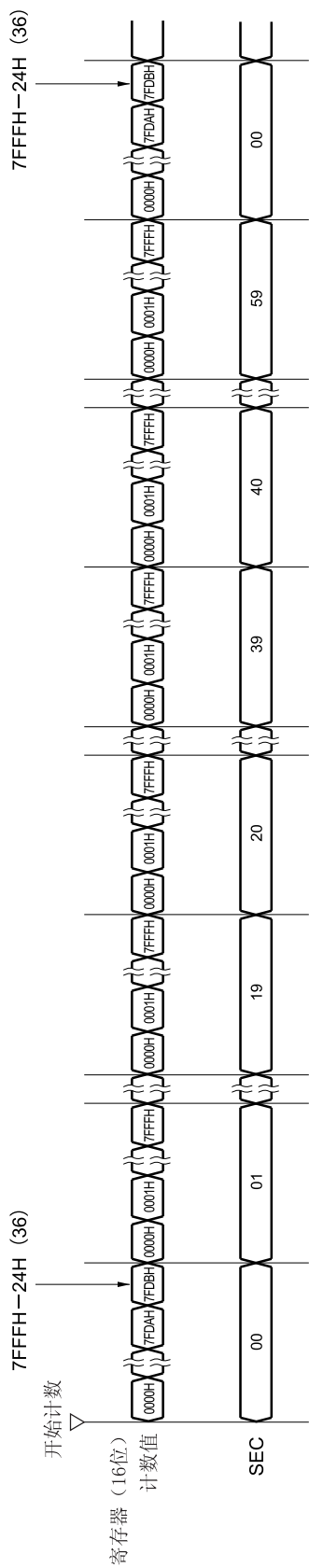
根据修正值计算(F5, F4, F3, F2, F1, F0)。

$$\begin{aligned} -\{(/F5, /F4, /F3, /F2, /F1, /F0) + 1\} \times 2 &= -36 \\ (/F5, /F4, /F3, /F2, /F1, /F0) &= 17 \\ (/F5, /F4, /F3, /F2, /F1, /F0) &= (0, 1, 0, 0, 0, 1) \\ (F5, F4, F3, F2, F1, F0) &= (1, 0, 1, 1, 1, 0) \end{aligned}$$

因此, 将 32767.4 Hz 修正 32768 Hz (32767.4 Hz + 18.3 ppm)时, 将修正寄存器设置为 DEV 为 1 且修正值为-36 (SUBCUD 寄存器的位 6 至位 0: 1101110), 则结果为 32768 Hz (0 ppm)。

当(DEV, F6, F5, F4, F3, F2, F1, F0)为(1, 1, 1, 0, 1, 1, 1, 0)时的修正操作如图 9-23 所示。

图 9-23. 当(DEV, F6, F5, F4, F3, F2, F1, F0) = (1, 1, 1, 0, 1, 1, 1, 0)时的操作



第十章 12位间隔定时器

10.1 12位间隔定时器的功能

可按照预先设置的任意的时间间隔产生中断(INTIT)。可用于从 STOP 模式的唤醒，以及 A/D 转换器的 SNOOZE 模式的触发。

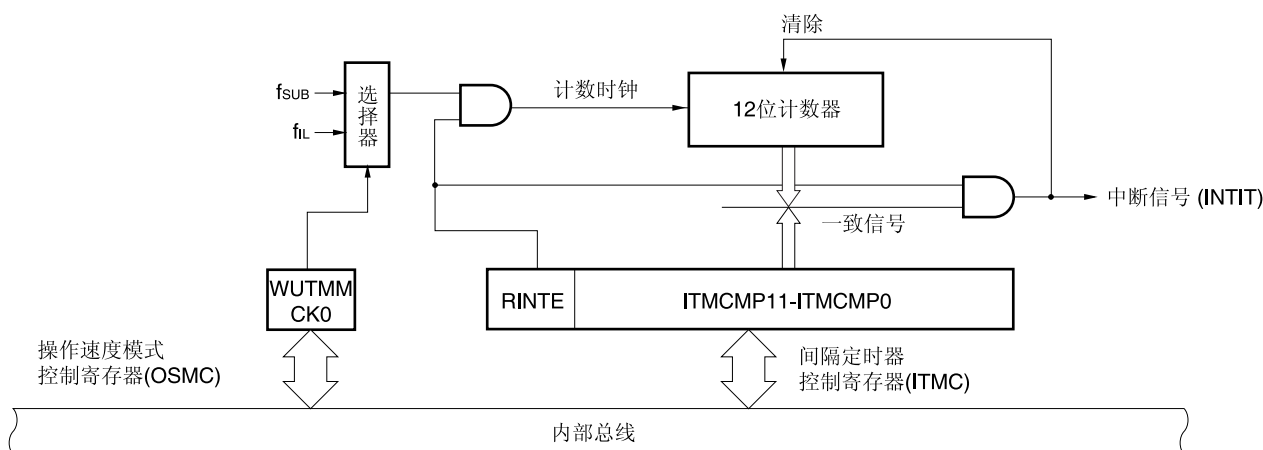
10.2 12位间隔定时器的配置

12位间隔定时器包含以下硬件。

表 10-1. 12位间隔定时器的配置

项目	配置
计数器	12位计数器
控制寄存器	外围允许寄存器 0 (PER0)
	操作速度模式控制寄存器(OSMC)
	间隔定时器控制寄存器(ITMC)

图 10-1. 12位间隔定时器的框图



注意事项 仅在 38 引脚产品中，可以选择副系统时钟(f_{SUB})作为操作时钟。

10.3 控制 12 位间隔定时器的寄存器

12 位间隔定时器由下列寄存器控制。

- 外围允许寄存器 0 (PER0)
- 操作速度模式控制寄存器(OSMC)
- 间隔定时器控制寄存器(ITMC)

(1) 外围允许寄存器 0 (PER0)

该寄存器用于允许或禁止向外围硬件供应时钟。通过停止向未使用的硬件供应时钟，可以降低功耗和噪声。

使用 12 位间隔定时器时，务必将该寄存器的位 7(RTCEN)设置为 1。

使用 1 位或 8 位存储器操作指令设置 PER0 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 10-2. 外围允许寄存器 0 (PER0)的格式

地址: F00F0H 复位: 00H R/W

符号	<7>	6	<5>	<4>	3	<2>	1	<0>
PER0	RTCEN	0	ADCEN	IICA0EN	0	SAU0EN	0	TAU0EN

RTCEN	控制实时时钟 (RTC) 和 12 位间隔定时器的输入时钟供应
0	停止输入时钟供应。 <ul style="list-style-type: none"> • 不可写入用于实时时钟(RTC)和 12 位间隔定时器的 SFR。 • 实时时钟(RTC)和 12 位间隔定时器处于复位状态。
1	允许输入时钟供应。 <ul style="list-style-type: none"> • 可以读取和写入用于实时时钟(RTC)和 12 位间隔定时器的 SFR。

- 注意事项**
1. 使用 12 位间隔定时器时，在输入时钟(f_{RTC})处于振荡稳定状态下，首先要将 RTCEN 位设置为 1。当 RTCEN = 0 时，对实时时钟或 12 位间隔定时器的控制寄存器进行的写入操作无效，而且读取值总为初始值。
 2. 在 STOP 模式以及副系统时钟时的 HALT 模式下，通过将操作速度模式控制寄存器(OSMC)的 RTCLPC 位设置为 1，停止向除实时时钟和 12 位间隔定时器以外的外围功能供应时钟。此时，请将 PER0 寄存器的 RTCEN 位设置为 1，将其他位（位 0 至位 6）设置为 0。
 3. 必须将位 1、3 和 6 清除为 0。

(2) 操作速度模式控制寄存器(OSMC)

WUTMMCK0 位可以用于选择 12 位间隔定时器操作时钟。

另外，通过停止不必要的时钟功能，RTCLPC 位可以用于降低功耗。有关 RTCLPC 位的设置详情，请参阅第五章时钟发生电路。

使用 8 位存储器操作指令来设置 OSMC 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 10-3. 操作速度模式控制寄存器(OSMC) 的格式

地址: F00F3H 复位: 00H R/W

符号	7	6	5	4	3	2	1	0
OSMC	RTCLPC	0	0	WUTMMCK0	0	0	0	0

WUTMMCK0	选择实时时钟和 12 位间隔定时器的操作时钟
0	副系统时钟 (f _{SUB})
1	低速片上振荡器时钟 (f _{IL})

(3) 间隔定时器控制寄存器(ITMC)

该寄存器用于设置 12 位间隔定时器的开始和停止，以及指定定时器比较值。

使用 16 位存储器操作指令设置 ITMC 寄存器。

产生复位信号后，该寄存器被设置为 0FFFH。

图 10-4. 间隔定时器控制寄存器 (ITMC)的格式

地址: FFF90H 复位: 0FFFH R/W

符号	15	14	13	12	11 至 0
ITMC	RINTE	0	0	0	ITMCMP11 至 ITMCMP0

RINTE	12位间隔定时器的操作控制
0	停止计数操作 (计数清除)
1	开始计数操作

ITMCMP11 to ITMCMP0	12位间隔定时器的比较值设置
001H	这些位以固定周期 (计数时钟周期 x (ITCMP 设置+ 1)) 产生中断。
•	
•	
FFFH	禁止设置
000H	

当 ITMCMP11 至 ITMCMP0 被设置为 001H 或 FFFH 时的中断周期示例

- ITMCMP11 至 ITMCMP0 = 001H、计数时钟: $f_{SUB} = 32.768 \text{ kHz}$ 时
 $1/32.768 \text{ [kHz]} \times (1 + 1) = 0.06103515625 \text{ [ms]} \cong 61.03 \text{ [}\mu\text{s]}$
- ITMCMP11 至 ITMCMP0 = FFFH、计数时钟: when $f_{SUB} = 32.768 \text{ kHz}$ 时
 $1/32.768 \text{ [kHz]} \times (4095 + 1) = 125 \text{ [ms]}$

- 注意事项**
1. 将 RINTE 位从 1 更改为 0 之前，先用中断屏蔽标志寄存器来禁止处理 INTIT 中断。重新开始操作 (从 0 更改为 1) 时，清除 ITIF 标志，然后允许处理中断。
 2. 设置 RINTE 位并经过一个计数时钟周期后，才能从 RINTE 位读取到值。
 3. 在从待机模式返回并再次进入待机模式之后，在设置 RINTE 位时，请确认 RINTE 位的写入值可以被读取，或者从待机模式恢复后等待计数器时钟的 1 个时钟以后。再进入待机模式。
 4. 仅当 RINTE = 0 时，才能更改 ITMCMP11 至 ITMCMP0 位的设置。
然而，可以在将 RINTE 从 0 更改为 1 或者从 1 更改为 0 的同时，更改 ITMCMP11 至 ITMCMP0 位的设置。

10.4 12位间隔定时器的操作

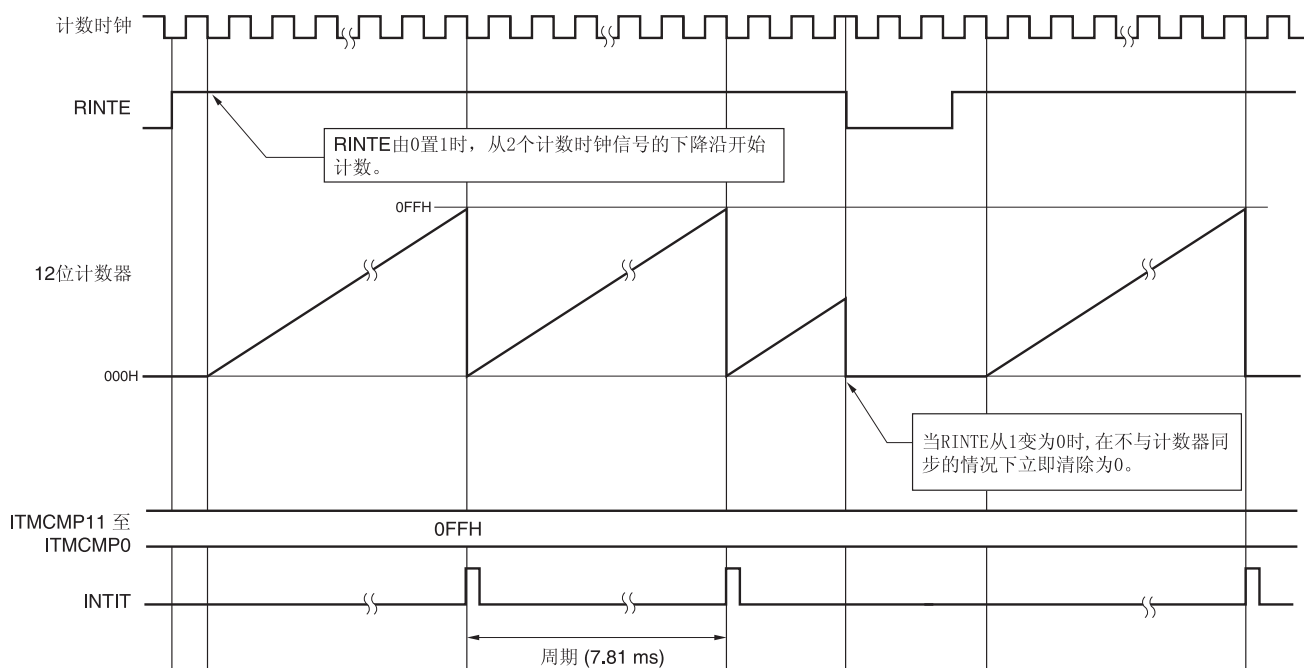
作为间隔定时器操作，将 ITMCMP11 至 ITMCMP0 位的指定计数值作为间隔，操作 12 位以重复产生中断请求 (INTIT)。

当 RINTE 位被设置为 1 时，12 位计数器开始计数。

当 12 位计数器的值和 ITMCMP11 至 ITMCMP0 位的指定值相一致时，清除 12 位计数器为 0，继续计数，同时产生中断请求信号 (INTIT)。

12 位间隔定时器的基本操作如下所示。

图 10-5. 12 位间隔定时器操作时序(ITMCMP11 至 ITMCMP0 = 0FFH, 计数时钟: $f_{SUB} = 32.768 \text{ kHz}$)



第十一章 看门狗定时器

11.1 看门狗定时器的功能

看门狗定时器工作于低速片上振荡器时钟。

看门狗定时器用于检测程序失控。检测到程序失控时产生内部复位信号。

下述情况时检测到程序失控。

- 当看门狗定时器发生溢出时
- 对看门狗定时器允许寄存器 (WDTE) 执行位操作指令时
- 将“ACH”以外的数据写入 WDTE 寄存器时
- 在窗口关闭期间将数据写入 WDTE 寄存器时

由于看门狗定时器而发生复位时，复位控制标志寄存器 (RESF) 的位 4 (WDTRF) 被设置为 1。关于 RESF 寄存器的详情，请参阅 **第二十二章 复位功能**。

达到 $75\% + 1/2/f_{IL}$ 的溢出时间时，可以产生间隔中断。

11.2 看门狗定时器的配置

看门狗定时器包括以下硬件。

表 11-1. 看门狗定时器的配置

项目	配置
控制寄存器	看门狗定时器允许寄存器(WDTE)

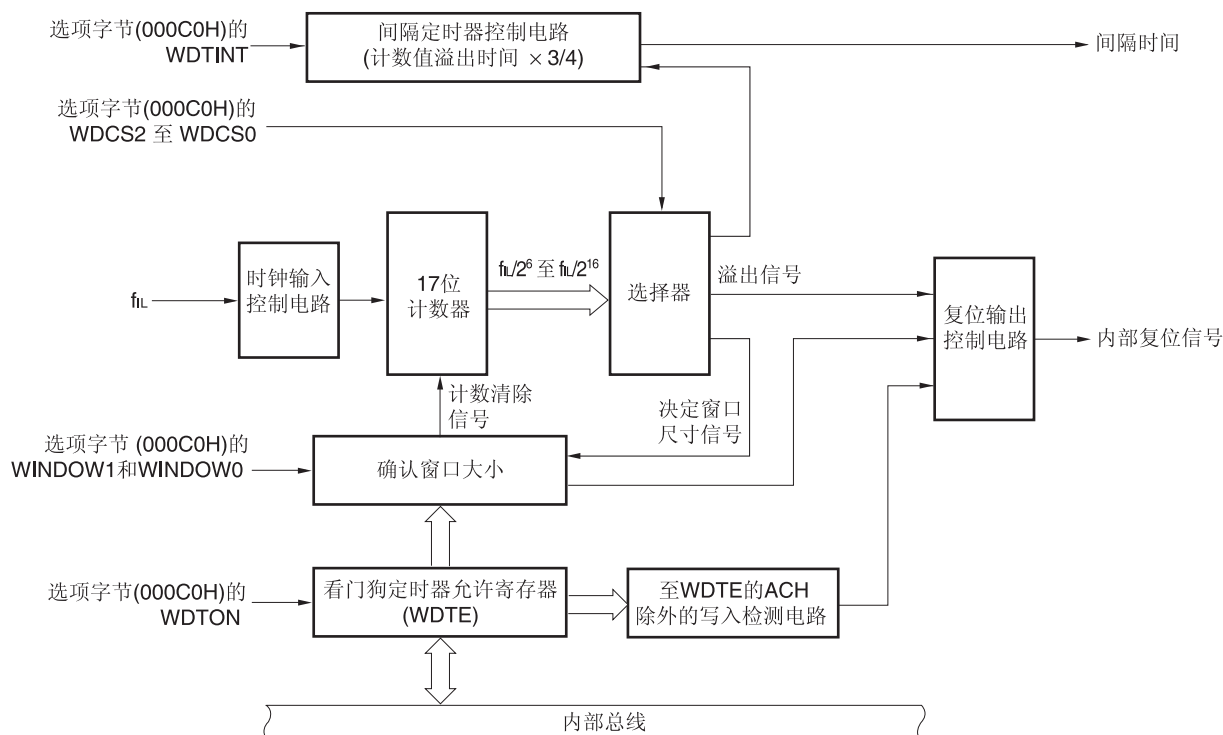
使用选项字节设置计数器操作工作方式、看门狗定时器的溢出时间、窗口开启期间以及间隔中断。

表 11-2. 选项字节和看门狗定时器的设置

看门狗定时器的设置	选项字节(000C0H)
看门狗定时器间隔中断	位7 (WDTINT)
窗口开启期间	位6和5 (WINDOW1, WINDOW0)
控制看门狗定时器的计数器操作	位4 (WDTON)
看门狗定时器的溢出时间	位3至1 (WDCS2至WDCS0)
控制看门狗定时器的计数器操作 (HALT/STOP模式下)	位0 (WDSTBYON)

备注 有关选项字节，请参阅第二十七章 选项字节。

图 11-1. 看门狗定时器的框图



11.3 控制看门狗定时器的寄存器

可利用看门狗定时器允许寄存器 (WDTE)控制看门狗定时器。

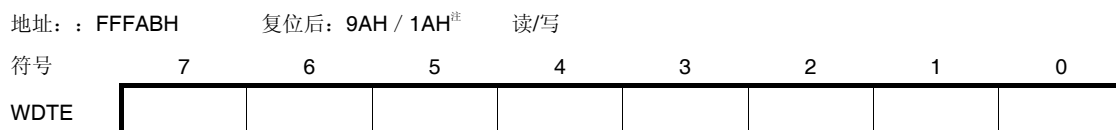
(1) 看门狗定时器允许寄存器(WDTE)

向 WDTE 寄存器写入“ACH”可以清除看门狗定时器计数器，并且再次开始计数。

使用 8 位存储器操作指令设置该寄存器。

产生复位信号后，该寄存器被设置为 9AH 或 1AH^注。

图 11-2. 看门狗定时器允许寄存器 (WDTE)的格式



注 WDTE 寄存器复位值因选项字节(000C0H)的 WDTON 位的设置值而异。将 WDTON 位设置为 1 以启动看门狗定时器。

WDTON位设置值	WDTE寄存器复位值
0 (禁止看门狗定时器的计数操作)	1AH
1 (允许看门狗定时器的计数操作)	9AH

- 注意事项**
1. 将“ACH”以外的值写入 WDTE 寄存器会产生内部复位信号。
 2. 对 WDTE 寄存器执行位操作指令会产生内部复位信号。
 3. WDTE 寄存器的读取值为 9AH/1AH (与写入值(ACH)不同)。

11.4 看门狗定时器的操作

11.4.1 控制看门狗定时器的操作

1. 使用看门狗定时器时，下列操作由选项字节(000C0H)指定。

- 将选项字节(000C0H)的位 4(WDTON)设置为 1 以允许看门狗定时器的计数操作（解除复位后计数器开始操作）(详情请参阅第二十七章)。

WDTON	看门狗定时器计数器
0	禁止计数器操作(复位后停止计数)
1	允许计数器操作(复位后开始计数)

- 用选项字节 (000C0H)的位 3 至 1 (WDCS2 至 WDCS0) 设置溢出时间 (详情请参阅 11.4.2 和第二十七章)。
- 使用选项字节 (000C0H)的位 6 和 5 (WINDOW1 和 WINDOW0) 设置窗口开启期间 (详情请参阅 11.4.3 和第二十七章)。

2. 解除复位后，看门狗定时器开始计数。

3. 在看门狗定时器开始计数之后到选项字节设置的溢出时间之前，向看门狗定时器允许寄存器(WDTE)写入“ACH”，则看门狗定时器被清除并且再次开始计数。

4. 此后，解除复位后对 WDTE 寄存器的第二次及之后的写入操作须在窗口开启期间执行。如果在窗口关闭期间写入 WDTE 寄存器，将产生内部复位信号。

5. 如果超过溢出时间前未将“ACH”写入 WDTE 寄存器，将产生内部复位信号。

下列情况下也会产生内部复位信号：

- 对 WDTE 寄存器执行位操作指令时
- 将“ACH”以外的数据写入 WDTE 寄存器时

注意事项 1. 解除复位后首次向看门狗定时器允许寄存器(WDTE)写入数据时，与窗口开启时间无关，在溢出时间之前任何时序写入 WDTE，看门狗定时器都被清除，且看门狗定时器再次开始计数。

2. 通过向 WDTE 寄存器写入“ACH”来清除看门狗定时器时，实际溢出时间与选项字节设置的溢出时间可能具有最大 $2/f_{IL}$ 秒的差异。

3. 直到计数值溢出之前，都可以清除看门狗定时器。

注意事项 4. 如下所示，看门狗定时器在 HALT、STOP 和 SNOOZE 模式下的操作因选项字节(000C0H)的位 0(WDSTBYON) 的设置值而异。

	WDSTBYON = 0	WDSTBYON = 1
HALT 模式时	停止看门狗定时器操作。	继续看门狗定时器操作。
STOP 模式时		
SNOOZE 模式时		

如果 WDSTBYON = 0，解除 HALT 或 STOP 模式后看门狗定时器恢复计数。此时，计数器被清除为 0 并开始计数。

解除 STOP 模式后工作于 X1 振荡时钟时，经过振荡稳定时间后 CPU 开始工作。

因此，如果从解除 STOP 模式到看门狗定时器溢出为止的时间较短，则会在振荡稳定时间内发生看门狗溢出而导致复位。

所以，通过间隔中断解除 STOP 模式后 CPU 工作在 X1 振荡时钟时，必须经过振荡稳定时间才能清除看门狗定时器，设置看门狗溢出时间时要考虑到这一点。

11.4.2 设置看门狗定时器的溢出时间

用选项字节(000C0H)的位 3 至位 1 (WDCS2 至 WDCS0) 设置看门狗定时器的溢出时间。

如果发生溢出，则产生内部复位信号。在溢出时间之前的窗口开启期间，通过向看门狗定时器允许寄存器(WDTE)写入“ACH”，清除当前计数，看门狗定时器重新开始计数。

可设置的溢出时间如下所示。

表 11-3. 设置看门狗定时器的溢出时间

WDCS2	WDCS1	WDCS0	看门狗定时器的溢出时间 (f _{IL} = 17.25 kHz (MAX.))
0	0	0	2 ⁶ /f _{IL} (3.71 ms)
0	0	1	2 ⁷ /f _{IL} (7.42 ms)
0	1	0	2 ⁸ /f _{IL} (14.84 ms)
0	1	1	2 ⁹ /f _{IL} (29.68 ms)
1	0	0	2 ¹¹ /f _{IL} (118.72 ms)
1	0	1	2 ¹³ /f _{IL} (474.90 ms)
1	1	0	2 ¹⁴ /f _{IL} (949.80 ms)
1	1	1	2 ¹⁶ /f _{IL} (3799.19 ms)

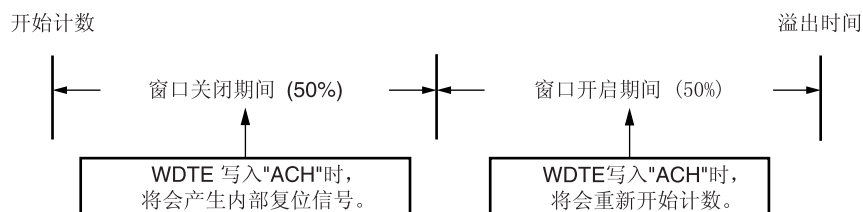
备注 f_{IL}: 低速片上振荡器时钟频率

11.4.3 设置看门狗定时器的窗口开启期间

用选项字节(000C0H)的位 6 和位 5(WINDOW1、WINDOW0)设置看门狗定时器的窗口开启期间。窗口概要如下所示。

- 如果在窗口开启期间将“ACH”写入看门狗定时器允许寄存器(WDTE)，则看门狗定时器被清除，并再次开始计数。
- 在窗口关闭期间即使将“ACH”写入 WDTE 寄存器，也会检测到异常并产生内部复位信号。

示例：窗口开启期间为 50%时



注意事项 解除复位后首次向 WDTE 寄存器写入数据时，与窗口开启时间无关，在溢出时间之前任何时序写入 WDTE，看门狗定时器都被清除，且看门狗定时器再次开始计数。

可设置的窗口开启期间如下所示。

表 11-4. 设置看门狗定时器的窗口开启期间

WINDOW1	WINDOW0	看门狗定时器的窗口开启期间
0	0	禁止设置
0	1	50%
1	0	75%
1	1	100%

注意事项 当选项字节(000C0H)的位 0(WDSTBYON) = 0 时，不论 WINDOW1 和 WINDOW0 位的值为多少，窗口开启期间都为 100%。

备注 溢出时间设置为 $2^9/f_{IL}$ 时的窗口关闭时间和开启时间如下所示。

	窗口开启期间的设置		
	50%	75% + 1/2f _{IL}	100%
窗口关闭时间	0至20.08 ms	0至10.04 ms	无
窗口开启期间	20.08至29.68 ms	10.04至29.68 ms	0至29.68 ms

<当窗口开启期间为 50%时>

- 溢出时间:
 $2^9/f_{IL} (\text{MAX.}) = 2^9/17.25 \text{ kHz} = 29.68 \text{ ms}$
- 窗口关闭时间:
 $0 \text{ 至 } 2^9/f_{IL} (\text{MIN.}) \times (1 - 0.5) = 0 \text{ 至 } 2^9/12.75 \text{ kHz} \times 0.5 = 0 \text{ 至 } 20.08 \text{ ms}$
- 窗口开启期间:
 $2^9/f_{IL} (\text{MIN.}) \times (1 - 0.5) \text{ 至 } 2^9/f_{IL} (\text{MAX.}) = 2^9/12.75 \text{ kHz} \times 0.5 \text{ 至 } 2^9/17.25 \text{ kHz}$
 $= 20.08 \text{ 至 } 29.68 \text{ ms}$

11.4.4 设置看门狗定时器的间隔中断

根据选项字节(000C0H)的位 7(WDTINT)的设置，当达到溢出时间的 75%时，可以产生间隔中断(INTWDTI)。

表 11-5. 看门狗定时器的间隔中断的设置

WDTINT	看门狗定时器间隔中断的使用
0	使用间隔中断。
1	当达到溢出时间的 75% + 1/2f _{IL} 时产生间隔中断。

注意事项 解除 STOP 模式后操作于 X1 振荡时钟时，经过振荡稳定时间后 CPU 开始操作。因此，如果从解除 STOP 模式到看门狗定时器溢出为止的时间较短，则在振荡稳定时间内发生溢出时导致复位。所以，通过间隔中断解除 STOP 模式后 CPU 工作在 X1 振荡时钟时，必须经过振荡稳定时间才能清除看门狗定时器，设置看门狗溢出时间时要考虑到这一点。

备注 即使在产生 INTWDTI 之后，看门狗定时器也会继续计数（直到将 ACH 写入看门狗定时器允许寄存器(WDTE)为止）。如果未在溢出时间之前将“ACH”写入 WDTE 寄存器，则会产生内部复位信号。

第十二章 A/D 转换器

A/D 转换器的模拟输入通道数因产品而异。

	20引脚	32引脚	30引脚, 38引脚
模拟输入通道	6通道 (ANI0至ANI2, ANI4, ANI5, ANI8)	9通道 (ANI0至ANI2, ANI4至ANI7, ANI18, ANI19)	11通道 (ANI0至ANI2, ANI4至ANI7, ANI16至ANI19)

注意事项 本章的下述内容主要是以 38 引脚产品为例进行说明。

12.1 A/D转换器的功能

A/D 转换器是 10 位分辨率^注转换器，用于将模拟输入信号转换为数字值，配置了可控制的共计 12 个通道的模拟输入，包括最多 11 个通道的 A/D 转换器模拟输入（ANI0 至 ANI2、ANI4 至 ANI7 以及 ANI16 至 ANI19）和 1 个内部可编程增益放大器输出信号(PGAOUT)。

A/D 转换器具有以下功能。

- 10 位分辨率 A/D 转换^注

对选自 ANI0 至 ANI12、ANI4 至 ANI7 以及 ANI16 至 ANI19 的其中一个模拟输入通道重复执行 10 位分辨率 A/D 转换。每结束一次 A/D 转换时，产生一个中断请求(INTAD)（处于选择模式时）。

注 可以使用 A/D 转换器模式寄存器 2 (ADM2)的 ADTYP 位来选择 8 位分辨率。

通过下述的模式组合，可以设置各种各样的 A/D 转换器模式。

触发模式	通道选择模式	转换操作模式
<ul style="list-style-type: none"> • 软件触发 通过软件设置触发来开始转换。 • 硬件触发无等待模式 通过检测硬件触发来开始转换。 • 硬件触发等待模式 电源关闭且处于转换待机状态时，通过检测硬件触发来打开电源，然后，当经过稳定等待时间后自动开始转换。 	<ul style="list-style-type: none"> • 选择模式 对一个通道的模拟输入执行 A/D 转换。 • 扫描模式 对四个通道的模拟输入依次执行 A/D 转换。 	<ul style="list-style-type: none"> • 单次转换模式 对选择的通道执行一次 A/D 转换。 • 连续转换模式 对选择的通道连续执行 A/D 转换，直到通过软件停止为止。

图12-1. A/D转换器的框图

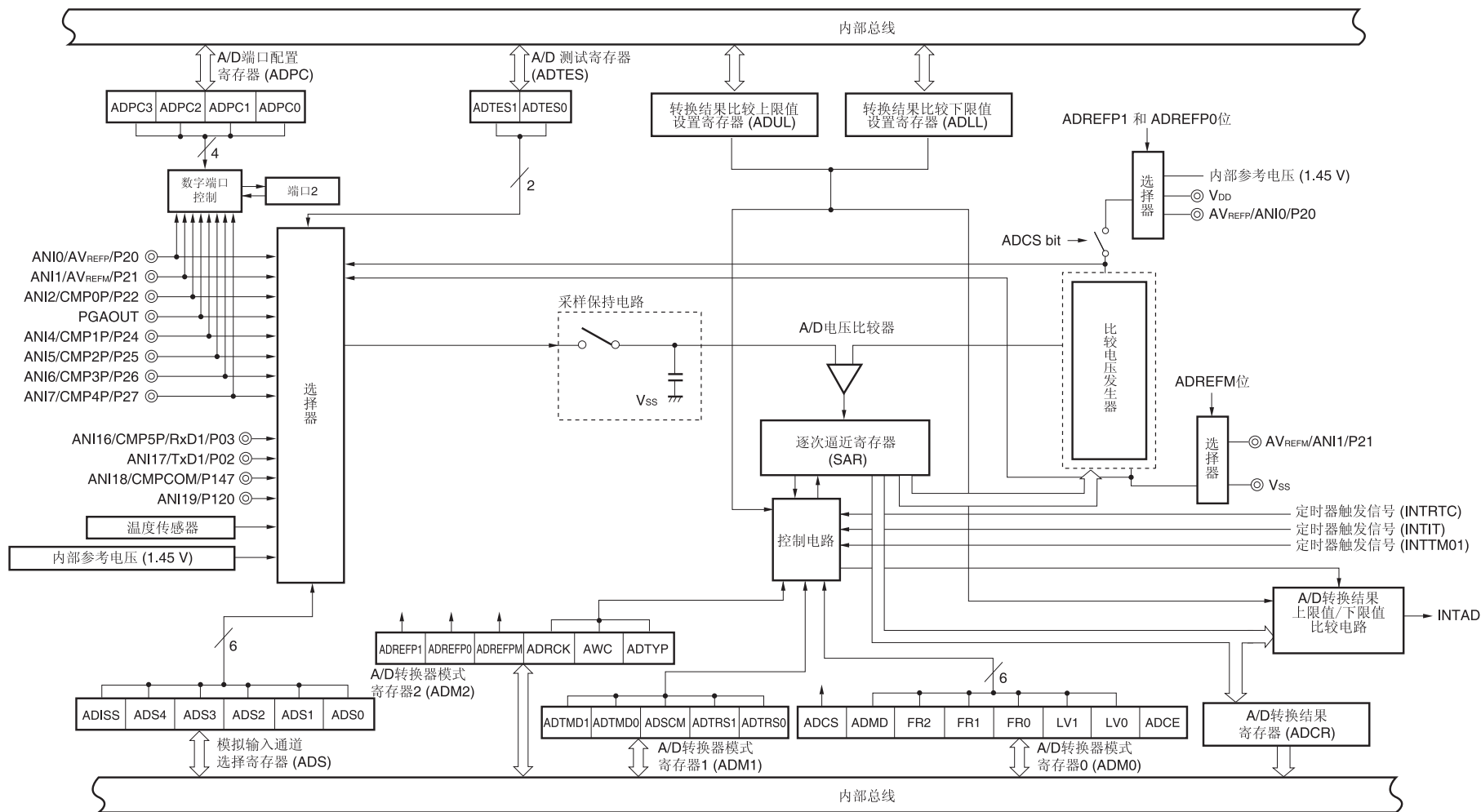


图12-1中的模拟输入引脚为使用38引脚产品时。

12.2 A/D转换器的配置

A/D 转换器包括以下硬件。

(1) ANI0 至 ANI2, ANI4 至 ANI7 和 ANI16 至 ANI19 引脚

这些是 A/D 转换器的 11 个通道的模拟输入引脚。它们用于输入模拟信号以转换成数字信号。被选为模拟输入引脚以外的其他引脚可以用作输入/输出端口引脚。

(2) PGAOUT

这是一个可编程增益放大器的内部输出引脚。A/D 转换器可通过选择可编程增益放大器的输出信号作为模拟输入来执行 A/D 转换。

(3) 采样&保持电路

采样&保持电路对来自输入电路的各个模拟输入电压依次进行采样，并将其发送至 A/D 电压比较器。在 A/D 转换期间，该电路保持采样得到的模拟输入电压。

(4) A/D 电压比较器

A/D 电压比较器将产生于比较电压生成电路的分接电压与模拟输入电压进行比较。对于比较结果，如果模拟输入电压大于基准电压($1/2 AV_{REF}$)，则置位逐次逼近寄存器(SAR)的最高有效位(MSB)。如果模拟输入电压小于基准电压($1/2 AV_{REF}$)，则将 SAR 寄存器的 MSB 位复位。

然后，SAR 寄存器的位 8 被自动置 1，并进行下一次比较。根据位 9 的设置结果值选择比较电压生成电路的分接电压。

位 9 = 0: ($1/4 AV_{REF}$)

位 9 = 1: ($3/4 AV_{REF}$)

将比较电压生成电路的分接电压与模拟输入电压进行比较，并根据比较结果，对 SAR 寄存器的位 8 进行操作。

模拟输入电压 \geq 比较电压生成电路的分接电压：位 8 = 1

模拟输入电压 \leq 比较电压生成电路的分接电压：位 8 = 0

继续进行比较，直到 SAR 寄存器的位 0 为止。

当以 8 位分辨率执行 A/D 转换时，继续进行比较直到 SAR 寄存器的位 2 为止。

备注 AV_{REF} : A/D 转换器的+侧基准电压。可以选自 AV_{REFP} 、内部基准电压(1.45 V)以及 V_{DD} 。

(5) 比较电压生成电路

比较电压生成电路生成从模拟输入引脚输入的电压的比较电压。

(6) 逐次逼近寄存器(SAR)

SAR 寄存器从最高有效位(MSB)开始按顺序在每位设置分接电压数据, 该数据为与模拟输入引脚的电压值匹配的来自比较电压生成电路的分接电压值。

当设置到了 SAR 寄存器的最低有效位(LSB) (A/D 转换结束), 则 SAR 寄存器中的内容 (转换结果) 将保持在 A/D 转换结果寄存器(ADCR)中。当所有指定的 A/D 转换都结束时, 产生 A/D 转换结束中断请求信号 (INTAD)。

(7) 10 位 A/D 转换结果寄存器(ADCR)

每次 A/D 转换完成时, A/D 转换结果将从逐次逼近寄存器被载入至该寄存器, ADCR 寄存器将 A/D 转换结果保持在其高 10 位 (低 6 位固定为 0)。

(8) 8 位 A/D 转换结果寄存器(ADCRH)

每次 A/D 转换完成时, A/D 转换结果将从逐次逼近寄存器被载入至该寄存器, ADCRH 寄存器存储 A/D 转换结果的高 8 位。

(9) 控制电路

该电路用于控制将被转换为数字信号的模拟输入的转换时间, 并用于开始或停止转换操作。当 A/D 转换完成时, 该控制电路产生 INTAD 信号。

(10) AV_{REFP} 引脚

该引脚输入外部基准电压(AV_{REFP})。

如果将 AV_{REFP} 用作 A/D 转换器的+侧基准电压, 须将 A/D 转换器模式寄存器 2(ADM2)的 ADREFP1 和 ADREFP0 位设置为 1。

根据施加在 AV_{REFP} 和-侧基准电压(AV_{REFM}/V_{SS})之间的电压, 将输入到 ANI0 至 ANI2、ANI4 至 ANI7 以及 ANI16 至 ANI20 的模拟信号转化为数字信号。

除了 AV_{REFP} 以外, 可以选择 V_{DD}或内部基准电压 (1.45 V) 作为 A/D 转换器的+侧基准电压。

(11) AV_{REFM} 引脚

该引脚输入外部基准电压(AV_{REFM})。如果将 AV_{REFM} 用作 A/D 转换器的-侧基准电压, 须将 ADM2 寄存器的 ADREFM 位设置为 1。

除了 AV_{REFM} 以外, 可以选择 V_{SS} 作为 A/D 转换器的-侧基准电压。

12.3 A/D转换器中使用的寄存器

A/D 转换器使用以下寄存器。

- 外围允许寄存器 0 (PER0)
- A/D 转换器模式寄存器 0 (ADM0)
- A/D 转换器模式寄存器 1 (ADM1)
- A/D 转换器模式寄存器 2 (ADM2)
- 10 位 A/D 转换结果寄存器(ADCR)
- 8 位 A/D 转换结果寄存器(ADCRH)
- 模拟输入通道选择寄存器(ADS)
- 转换结果比较上限值设置寄存器(ADUL)
- 转换结果比较下限值设置寄存器(ADLL)
- A/D 测试寄存器(ADTES)
- A/D 端口配置寄存器(ADPC)
- 端口模式控制寄存器 0, 12, 14 (PMC0, PMC12, PMC14)
- 端口模式寄存器 0, 2, 12 和 14 (PM0, PM2, PM12, PM14)

(1) 外围允许寄存器 0 (PER0)

该寄存器用于允许或禁止向外围硬件供应时钟。通过停止向未使用的硬件供应时钟，可以降低功耗和噪声。

使用 A/D 转换器时，必须将该寄存器的位 5(ADCEN)设置为 1。

使用 1 位或 8 位存储器操作指令设置 PER0 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 12-2. 外围允许寄存器 0 (PER0)的格式

地址: F00F0H 复位后: 00H R/W

符号	<7>	6	<5>	<4>	3	<2>	1	<0>
PER0	RTCEN	0	ADCEN	IICA0EN	0	SAU0EN	0	TAU0EN

ADCEN	A/D转换器输入时钟供应的控制
0	停止输入时钟供应。 <ul style="list-style-type: none"> • 不可以向A/D转换器使用的SFR写入。 • A/D转换器处于复位状态。
1	允许输入时钟供应。 <ul style="list-style-type: none"> • 可以对A/D转换器使用的SFR读出/写入。

注意事项 1. 设置 A/D 转换器时，必须先将 ADCEN 位设置为 1。如果 ADCEN = 0，对于 A/D 转换器的控制寄存器进行的写操作将被忽略，而且该寄存器的读取值总为初始值（端口模式寄存器 0、2、12 和 14（PM0、PM2、PM12、PM14）、端口模式控制寄存器 0、12 和 14（PMC0、PMC12、PMC14）以及 A/D 端口配置寄存器(ADPC)除外）。

2. 必须将位 1、3 和 6 清除为 0。

(2) A/D 转换器模式寄存器 0 (ADM0)

该寄存器用于设置进行 A/D 转换的模拟输入的转换时间，并开始/停止转换。

使用 1 位或 8 位存储器操作指令来设置 ADM0 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 12-3. A/D 转换器模式寄存器 0 (ADM0)的格式

地址: FFF30H 复位后: 00H R/W

符号	<7>	6	5	4	3	2	1	<0>
ADM0	ADCS	ADMD	FR2 ^{注1}	FR1 ^{注1}	FR0 ^{注1}	LV1 ^{注1}	LV0 ^{注1}	ADCE

ADCS	A/D转换操作控制
0	停止转换操作 [读取时] 转换停驶/待机状态
1	允许转换操作 [读取时] 软件触发模式下: 转换操作状态 硬件触发等待模式下: 稳定等待状态 + 转换操作状态

ADMD	设置A/D转换通道选择模式
0	选择模式
1	扫描模式

ADCE	A/D电压比较器操作控制 ^{注2}
0	停止A/D电压比较器操作
1	允许A/D电压比较器操作

注 1. 关于 FR2 至 FR0、LV1、LV0 位和 A/D 转换的详情，请参阅表 12-3. A/D 转换时间的选择。

2. 当处于软件触发模式或硬件触发无等待模式时，A/D 电压比较器的操作由 ADCS 和 ADCE 位控制，从操作开始到操作稳定需要 1 μ s。因此，在 ADCE 位被设置为 1 且经过 1 μ s 或更长时间之后，将 ADCS 位设置为 1，则此时的转换结果相对于初次转换结果是有效的。若未等到 1 μ s 就将 ADCS 位设置为 1，则忽略初次转换结果。

注意事项 1. 请在转换处于停止或者待机 (ADCS = 0) 状态时更改 ADMD、FR2 至 FR0、LV1、LV0 和 ADCE 位。

2. 禁止使用 8 位操作指令同时将 ADCE = 0、ADCS = 0 更改为 ADCE = 1、ADCS = 1。请务必按照 12.7 A/D 转换器的设置流程图中描述的顺序设置这些位。

表 12-1. ADCS 和 ADCE 位的设置

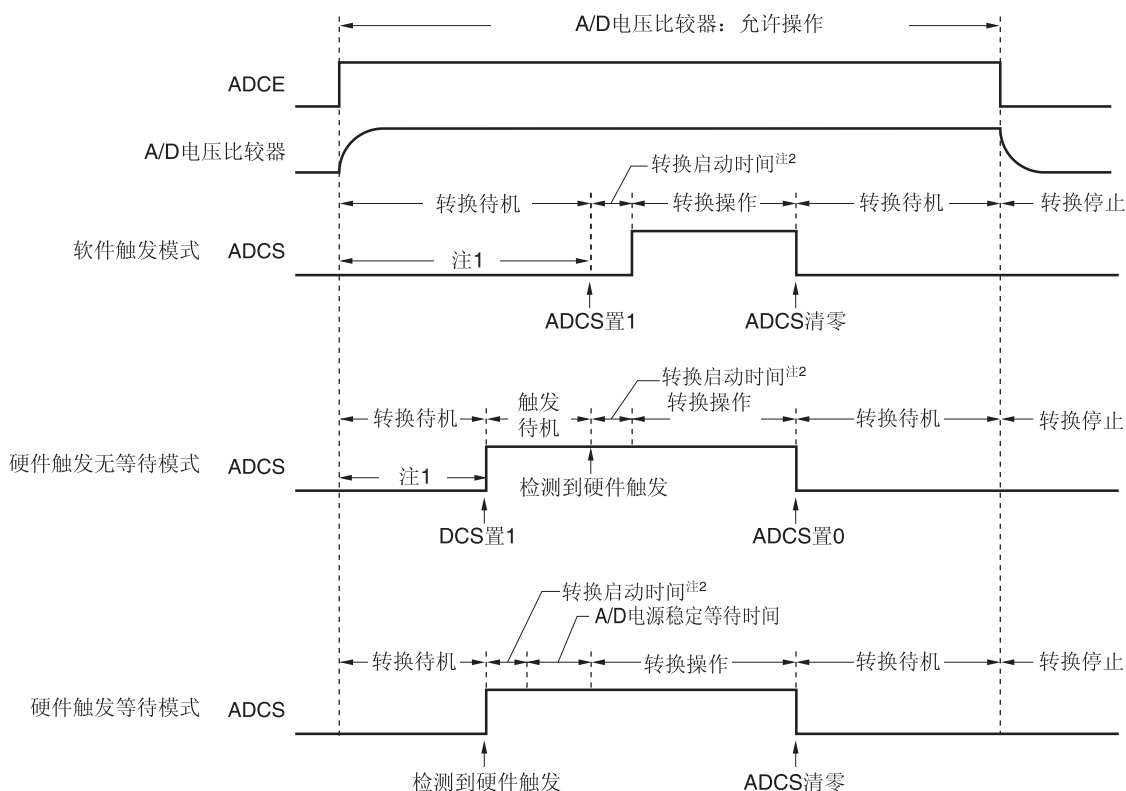
ADCS	ADCE	A/D 转换操作
0	0	停止状态 (不存在 DC 功耗路径)
0	1	转换待机模式 (仅 A/D 电压比较器消耗功率 ^注)
1	0	禁止设置
1	1	转换模式 (A/D 电压比较器: 允许操作)

注 硬件触发等待模式时, 即使在转换待机模式下也没有 DC 功耗路径。

表 12-2. ADCS 位的置位和清除条件

A/D 转换模式			置位条件	清除条件
软件触发	选择模式	连续转换模式	当写入 ADCS = 1 时	当写入 ADCS = 0 时
		单次转换模式		<ul style="list-style-type: none"> 当写入 ADCS = 0 时 A/D 转换结束时, 该位自动清除至 0。
	扫描模式	连续转换模式		当写入 ADCS = 0 时
		单次转换模式		<ul style="list-style-type: none"> 当写入 ADCS = 0 时 所设定的四个通道的转换结束时, 该位自动清除至 0。
硬件触发无等待模式	选择模式	连续转换模式		当写入 ADCS = 0 时
		单次转换模式		当写入 ADCS = 0 时
	扫描模式	连续转换模式		当写入 ADCS = 0 时
		单次转换模式		当写入 ADCS = 0 时
硬件触发等待模式	选择模式	连续转换模式	当硬件触发被输入	当写入 ADCS = 0 时
		单次转换模式		<ul style="list-style-type: none"> 当写入 ADCS = 0 时 A/D 转换结束时, 该位自动清除至 0。
	扫描模式	连续转换模式		当写入 ADCS = 0 时
		单次转换模式		<ul style="list-style-type: none"> 当写入 ADCS = 0 时 所设定的四个通道的转换结束时, 该位自动清除至 0。

图 12-4. 使用 A/D 电压比较器时的时序图



- 注 1. 当处于软件触发模式或硬件触发无等待模式时，从 ADCE 位上升到 ADCS 位上升的时间必须为 1 μs 或更长，以使内部电路稳定。
- 2. 启动转换最长需要的时间如下表所示。

ADM0			转换时钟 (f _{AD})	转换开始时间(f _{CLK} 时钟数)	
FR2	FR1	FR0		软件触发模式/ 硬件触发无等待模式	硬件触发等待模式
0	0	0	f _{CLK} /64	63	1
0	0	1	f _{CLK} /32	31	
0	1	0	f _{CLK} /16	15	
0	1	1	f _{CLK} /8	7	
1	0	0	f _{CLK} /6	5	
1	0	1	f _{CLK} /5	4	
1	1	0	f _{CLK} /4	3	
1	1	1	f _{CLK} /2	1	

但是，在连续转换模式的第二次以后的连续转换中和在扫描模式由扫描 1, 2, 3 指定的通道转换中，检测到硬件触发后，不发生转换启动时间和 A/D 电源稳定等待时间。

- 注意事项 1. 如果使用硬件触发等待模式，禁止将 ADCS 位设置为 1（但当检测到硬件触发信号时，该位自动切换至 1）。然而，可以将 ADCS 位清除至 0，以设置 A/D 转换待机模式。
- 2. 当处于硬件触发无等待模式的单次转换模式时，ADCS 标志不会在 A/D 转换结束时自动清除至 0。相反，将继续保持为 1。

- 注意事项 3. 仅在 $ADCS = 0$ （当处于转换停止/转换待机状态时）对 $ADCE$ 位进行改写。
4. 要完成 A/D 转换，设置如下所示的时间或更长时间为硬件触发间隔时间：
硬件触发无等待模式： $2 f_{CLK}$ 时钟 + A/D 转换时间
硬件触发等待模式： $2 f_{CLK}$ 时钟 + 稳定等待时间 + A/D 转换时间

备注 f_{CLK} ：CPU/外围硬件时钟频率

表 12-3. A/D 转换时间的选择 (1/4)

(1) 无稳定等待时间

正常模式 1, 2 (软件触发模式/硬件触发无等待模式)

A/D 转换器模式寄存器 0 (ADM0)					模式	转换时钟 (f _{AD})	转换时钟数	转换时间	转换时间的选择					
FR2	FR1	FR0	LV1	LV0					2.7 V ≤ V _{DD} ≤ 5.5 V					
									f _{CLK} = 1 MHz	f _{CLK} = 4 MHz	f _{CLK} = 8 MHz	f _{CLK} = 16 MHz	f _{CLK} = 32 MHz	
0	0	0	0	0	正常1	f _{CLK} /64	19 f _{AD} (采样时钟数: 7 f _{AD})	1216/f _{CLK}	禁止设置	禁止设置	禁止设置	76 μs	38 μs	
0	0	1	f _{CLK} /32	608/f _{CLK}		76 μs		38 μs				19 μs		
0	1	0	f _{CLK} /16	304/f _{CLK}		76 μs		38 μs				19 μs	9.5 μs	
0	1	1	f _{CLK} /8	152/f _{CLK}		38 μs		19 μs				9.5 μs	4.75 μs	
1	0	0	f _{CLK} /6	114/f _{CLK}		28.5 μs		14.25 μs				7.125 μs	3.5625 μs	
1	0	1	f _{CLK} /5	95/f _{CLK}		95 μs		23.75 μs				11.875 μs	5.938 μs	2.9688 μs ^{注1}
1	1	0	f _{CLK} /4	76/f _{CLK}		76 μs		19 μs				9.5 μs	4.75 μs	2.375 μs ^{注1}
1	1	1	f _{CLK} /2	38/f _{CLK}		38 μs		9.5 μs				4.75 μs	2.375 μs ^{注1,2}	禁止设置
0	0	0	0	1	正常2	f _{CLK} /64	17 f _{AD} (采样时钟数: 5 f _{AD})	1088/f _{CLK}	禁止设置	禁止设置	禁止设置	68 μs	34 μs	
0	0	1	f _{CLK} /32	544/f _{CLK}		68 μs		34 μs				17 μs		
0	1	0	f _{CLK} /16	272/f _{CLK}		68 μs		34 μs				17 μs	8.5 μs	
0	1	1	f _{CLK} /8	136/f _{CLK}		34 μs		17 μs				8.5 μs	4.25 μs	
1	0	0	f _{CLK} /6	102/f _{CLK}		25.5 μs		12.75 μs				6.375 μs	3.1875 μs ^{注2}	
1	0	1	f _{CLK} /5	85/f _{CLK}		85 μs		21.25 μs				10.625 μs	5.3125 μs	2.6563 μs ^{注1,2}
1	1	0	f _{CLK} /4	68/f _{CLK}		68 μs		17 μs				8.5 μs	4.25 μs	2.125 μs ^{注1,2}
1	1	1	f _{CLK} /2	34/f _{CLK}		34 μs		8.5 μs				4.25 μs	2.125 μs ^{注1,2}	禁止设置

注 1. V_{DD} < 3.6 V 时禁止设置

2. 使用温度传感器时禁止使用。

注意事项 1. 当处于转换停止/转换待机状态(ADCS = 0)下, 将 FR2 至 FR0、LV1 以及 LV0 位改写为不同数据。

2. 上述转换时间不包括转换启动时间。第一次转换中要加上转换启动时间。选择转换时间时要考虑到时钟频率误差。

备注 f_{CLK}: CPU/外围硬件时钟频率

表 12-3. A/D 转换时间的选择 (2/4)

(2) 无稳定等待时间

低电压模式 1, 2^{注1} (软件触发模式/硬件触发无等待模式)

A/D转换器模式寄存器0 (ADM0)					模式	转换时钟 (f _{AD})	转换时钟数	转换时间	转换时间的选择										
FR2	FR1	FR0	LV1	LV0					2.7 V ≤ V _{DD} ≤ 5.5 V										
									f _{CLK} = 1 MHz	f _{CLK} = 4 MHz	f _{CLK} = 8 MHz	f _{CLK} = 16 MHz	f _{CLK} = 32 MHz						
0	0	0	1	0	低电压1	f _{CLK} /64	19 f _{AD} (采样时钟数: 7 f _{AD})	1216/f _{CLK}	禁止设置	禁止设置	禁止设置	76 μs	38 μs						
						f _{CLK} /32								608/f _{CLK}	76 μs	38 μs	19 μs		
						f _{CLK} /16								304/f _{CLK}	76 μs	38 μs	19 μs	9.5 μs	
						f _{CLK} /8								152/f _{CLK}	38 μs	19 μs	9.5 μs	4.75 μs	
						f _{CLK} /6								114/f _{CLK}	28.5 μs	14.25 μs	7.125 μs	3.5625 μs	
						f _{CLK} /5								95/f _{CLK}	95 μs	23.75 μs	11.875 μs	5.938 μs ^{注2}	2.9688 μs ^{注2}
						f _{CLK} /4								76/f _{CLK}	76 μs	19 μs	9.5 μs	4.75 μs	2.375 μs ^{注2}
						f _{CLK} /2								38/f _{CLK}	38 μs	9.5 μs	4.75 μs	2.375 μs ^{注2}	禁止设置 ^{注2}
0	0	0	1	1	低电压2	f _{CLK} /64	17 f _{AD} (采样时钟数: 5 f _{AD})	1088/f _{CLK}	禁止设置	禁止设置	禁止设置	68 μs	34 μs						
						f _{CLK} /32								544/f _{CLK}	68 μs	34 μs	17 μs		
						f _{CLK} /16								272/f _{CLK}	68 μs	34 μs	17 μs	8.5 μs	
						f _{CLK} /8								136/f _{CLK}	34 μs	17 μs	8.5 μs	4.25 μs	
						f _{CLK} /6								102/f _{CLK}	25.5 μs	12.75 μs	6.375 μs	3.1875 μs	
						f _{CLK} /5								85/f _{CLK}	85 μs	21.25 μs	10.625 μs	5.3125 μs ^{注2}	2.6563 μs ^{注2}
						f _{CLK} /4								68/f _{CLK}	68 μs	17 μs	8.5 μs	4.25 μs	2.125 μs ^{注2}
						f _{CLK} /2								34/f _{CLK}	34 μs	8.5 μs	4.25 μs	2.125 μs ^{注2}	禁止设置 ^{注2}

注 1. 使用温度传感器时禁止使用此模式。
 2. V_{DD} < 3.6 V 时禁止设置

注意事项 1. 当处于转换停止/转换待机状态(ADCS = 0)下, 将 FR2 至 FR0、LV1 以及 LV0 位改写为不同数据。
 2. 上述转换时间不包括转换启动时间。第一次转换中要加上转换启动时间。选择转换时间时要考虑到时钟频率误差。

备注 f_{CLK}: CPU/外围硬件时钟频率

表 12-3. A/D 转换时间的选择 (3/4)

(3) 有稳定等待时间

正常模式 1, 2 (硬件触发等待模式^{注1)})

A/D转换器模式寄存器0 (ADM0)					模式	转换时钟 (f _{AD})	稳定等待 时钟数	转换时钟数	稳定等待 时钟 + 转 换时间	稳定等待时间 + 转换时间的选择					
FR 2	FR 1	FR 0	LV 1	LV 0						2.7 V ≤ V _{DD} ≤ 5.5 V					
										f _{CLK} = 1 MHz	f _{CLK} = 4 MHz	f _{CLK} = 8 MHz	f _{CLK} = 16 MHz	f _{CLK} = 32 MHz	
0	0	0	0	0	正常1	f _{CLK} /64	8 f _{AD}	19 f _{AD} (采样 时钟数: 7 f _{AD})	1728/f _{CLK}	禁止 设置	禁止设置	禁止设置	108 μs	54 μs	
0	0	1	f _{CLK} /32	864/f _{CLK}		108 μs			54 μs				27 μs		
0	1	0	f _{CLK} /16	432/f _{CLK}		108 μs			54 μs				27 μs	13.5 μs	
0	1	1	f _{CLK} /8	216/f _{CLK}		54 μs			27 μs				13.5 μs	6.75 μs	
1	0	0	f _{CLK} /6	162/f _{CLK}		40.5 μs			20.25 μs				10.125 μs	5.0625 μs	
1	0	1	f _{CLK} /5	135/f _{CLK}		135 μs			33.75 μs				16.875 μs	8.4375 μs	4.21875 μs ^{注2}
1	1	0	f _{CLK} /4	108/f _{CLK}		108 μs			27 μs				13.5 μs	6.75 μs	3.375 μs ^{注2}
1	1	1	f _{CLK} /2	54/f _{CLK}		54 μs			13.5 μs				6.75 μs	3.375 μs ^{注2, 3}	禁止设置
0	0	0	0	1	正常2	f _{CLK} /64	8 f _{AD}	17 f _{AD} (采样 时钟数: 5 f _{AD})	1600/f _{CLK}	禁止 设置	禁止设置	禁止设置	100 μs	50 μs	
0	0	1	f _{CLK} /32	800/f _{CLK}		100 μs			50 μs				25 μs		
0	1	0	f _{CLK} /16	400/f _{CLK}		100 μs			50 μs				25 μs	12.5 μs	
0	1	1	f _{CLK} /8	200/f _{CLK}		50 μs			25 μs				12.5 μs	6.25 μs	
1	0	0	f _{CLK} /6	150/f _{CLK}		37.5 μs			18.75 μs				9.375 μs	4.6875 μs ^{注3}	
1	0	1	f _{CLK} /5	125/f _{CLK}		125 μs			31.25 μs				15.625 μs	7.8125 μs	3.90625 μs ^{注2, 3}
1	1	0	f _{CLK} /4	100/f _{CLK}		100 μs			25 μs				12.5 μs	6.25 μs	3.125 μs ^{注2, 3}
1	1	1	f _{CLK} /2	50/f _{CLK}		50 μs			12.5 μs				6.25 μs	3.125 μs ^{注2, 3}	禁止设置

注 1. 在连续转换模式的第二次以后的连续转换中和在扫描模式由扫描 1, 2, 3 指定的通道转换中, 检测到硬件触发(参阅表 12-3 (1/4))后, 不发生转换启动时间和 A/D 电源稳定等待时间。

2. V_{DD} < 3.6 V 时禁止设置
3. 使用温度传感器时禁止使用。

注意事项 1. 当处于转换停止/转换待机状态(ADCS = 0)下, 将 FR2 至 FR0、LV1 以及 LV0 位改写为不同数据。

2. 上述转换时间不包括转换启动时间。第一次转换中要加上转换启动时间。选择转换时间时要考虑到时钟频率误差。
3. 当处于硬件触发等待模式时, 转换时间包括在检测到硬件触发之后等待稳定所用的时间。

备注 f_{CLK}: CPU/外围硬件时钟频率

表 12-3. A/D 转换时间的选择 (4/4)

(4) 无稳定等待时间
低电压模式 1, 2^{注1} (硬件触发等待模式^{注2})

A/D转换器模式寄存器0 (ADM0)					模式	转换时钟 (f _{AD})	稳定等待时钟数	转换时钟数	稳定等待时钟 + 转换时间	稳定等待时间 + 转换时间的选择				
FR2	FR1	FR0	LV1	LV0						2.7 V ≤ V _{DD} ≤ 5.5 V				
										f _{CLK} = 1 MHz	f _{CLK} = 4 MHz	f _{CLK} = 8 MHz	f _{CLK} = 16 MHz	f _{CLK} = 32 MHz
0	0	0	0	0	低电压1	f _{CLK} /64	2 f _{AD}	19 f _{AD} (采样时钟数: 7 f _{AD})	1344/f _{CLK} 禁止设置	禁止设置	禁止设置	禁止设置	84 μs	42 μs
0	0	1	f _{CLK} /32	设置		84 μs				42 μs	21 μs	21 μs		
0	1	0	f _{CLK} /16	84 μs		42 μs				21 μs	10.5 μs			
0	1	1	f _{CLK} /8	42 μs		21 μs				10.5 μs	5.25 μs			
1	0	0	f _{CLK} /6	31.25 μs		15.75 μs				7.875 μs	3.9375 μs			
1	0	1	f _{CLK} /5	105/f _{CLK}		105 μs				26.25 μs	13.125 μs	6.5625 μs ^{注3}	3.28125 μs ^{注3}	
1	1	0	f _{CLK} /4	84/f _{CLK}		84 μs				21 μs	10.5 μs	5.25 μs	2.625 μs ^{注3}	
1	1	1	f _{CLK} /2	42/f _{CLK}		42 μs				10.5 μs	5.25 μs	2.625 μs	禁止设置	
0	0	0	0	1	低电压2	f _{CLK} /64	2 f _{AD}	17 f _{AD} (采样时钟数: 5 f _{AD})	1216/f _{CLK} 禁止设置	禁止设置	禁止设置	禁止设置	76 μs	38 μs
0	0	1	f _{CLK} /32	设置		76 μs				38 μs	19 μs	19 μs		
0	1	0	f _{CLK} /16	76 μs		38 μs				19 μs	9.5 μs			
0	1	1	f _{CLK} /8	38 μs		19 μs				9.5 μs	4.75 μs			
1	0	0	f _{CLK} /6	28.5 μs		14.25 μs				7.125 μs	3.5625 μs			
1	0	1	f _{CLK} /5	96/f _{CLK}		96 μs				23.75 μs	12 μs	5.938 μs	2.9688 μs ^{注3}	
1	1	0	f _{CLK} /4	76/f _{CLK}		76 μs				19 μs	9.5 μs	4.75 μs	2.375 μs ^{注3}	
1	1	1	f _{CLK} /2	38/f _{CLK}		38 μs				9.5 μs	4.75 μs	2.375 μs ^{注3}	禁止设置 ^{注3}	

注 1. 使用温度传感器时禁止使用此模式。

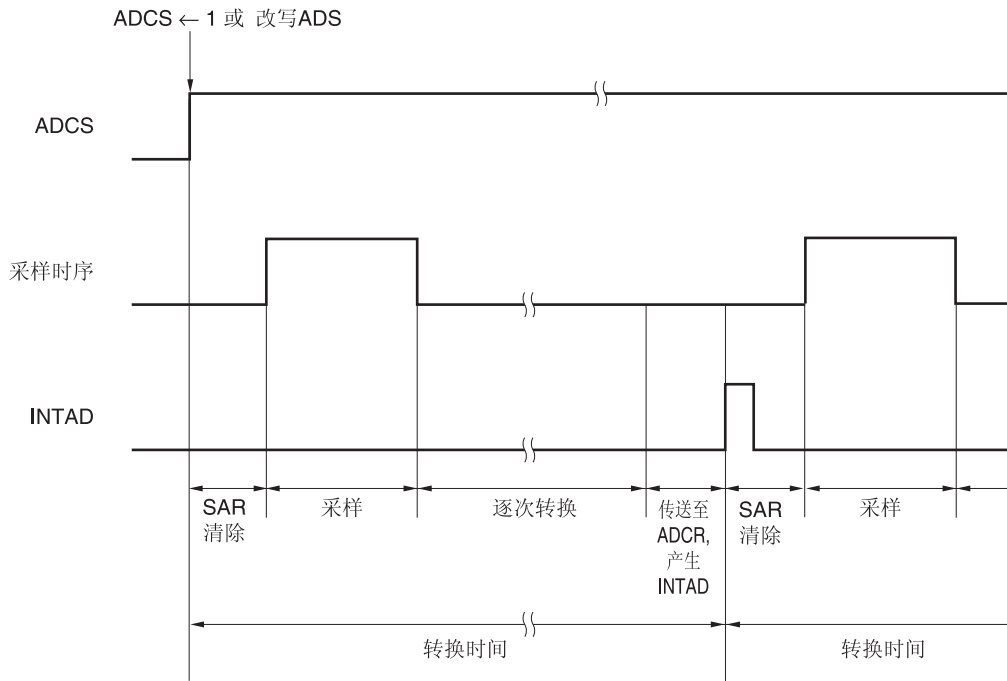
- 在连续转换模式的第二次以后的连续转换中和在扫描模式由扫描 1, 2, 3 指定的通道转换中, 检测到硬件触发(参阅表 12-3 (2/4))后, 不发生转换启动时间和 A/D 电源稳定等待时间。
- V_{DD} < 3.6 V 时禁止设置

注意事项 1. 当处于转换停止/转换待机状态(ADCS = 0)下, 将 FR2 至 FR0、LV1 以及 LV0 位改写为不同数据。

- 上述转换时间不包括转换启动时间。第一次转换中要加上转换启动时间。选择转换时间时要考虑到时钟频率误差。
- 当处于硬件触发等待模式时, 转换时间包括在检测到硬件触发之后等待稳定所用的时间。

备注 f_{CLK}: CPU/外围硬件时钟频率

图 12-5. A/D 转换器采样和 A/D 转换的时序 (以软件触发模式为例)



(3) A/D 转换器模式寄存器 1 (ADM1)

该寄存器用于设置 A/D 转换触发、转换模式和硬件触发信号。

使用 1 位或 8 位存储器操作指令来设置 ADM1 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 12-6. A/D 转换器模式寄存器 1 (ADM1)的格式

地址: FFF32H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ADM1	ADTMD1	ADTMD0	ADSCM	0	0	0	ADTRS1	ADTRS0

ADTMD1	ADTMD0	A/D 转换触发模式的选择
0	x	软件触发模式
1	0	硬件触发无等待模式
1	1	硬件触发等待模式

ADSCM	A/D 转换模式的设置
0	连续转换模式
1	单次转换模式

ADTRS1	ADTRS0	硬件触发信号的选择
0	0	定时器通道 1 的计数结束或捕捉结束中断信号(INTTM01)
0	1	定时器 KB0 至 KB2 的 A/D 触发信号 ^注
1	0	实时时钟中断信号(INTRTC)
1	1	12 位间隔定时器中断信号(INTIT)

注 使用外围功能切换寄存器 0 (PFSEL0)的位 2 和 3 (ADTRG10 和 ADTRG11) 从定时器 KB0 至 KB2 的 A/D 触发信号中选择一个信号。关于 PFSEL0 寄存器的详情，请参阅 7.3 (16) 外围功能切换寄存器 0 (PFSEL0)。

- 注意事项 1. 仅在停止转换操作时改写 ADM1 寄存器的值（即 A/D 转换器模式寄存器 0(ADM0)的 ADCE 位为 0 时）。
2. 要完成 A/D 转换，设置如下所示的时间或更长时间为硬件触发间隔时间：
 硬件触发无等待模式： 2 f_{CLK} 时钟 + A/D 转换时间
 硬件触发等待模式： 2 f_{CLK} 时钟 + 稳定等待时间 + A/D 转换时间
3. SNOOZE 以外的模式中，在 INTRTC 或 INTIT 输入后的最大 4 个 f_{CLK} 周期内，下一个 INTRTC 或 INTIT 的输入不被识别为有效的硬件触发。

备注 1. x: 忽略

2. f_{CLK}: CPU/外围硬件时钟频率

(4) A/D 转换器模式寄存器 2 (ADM2)

该寄存器用于选择 A/D 转换器基准电压，检验 A/D 转换结果的上限值和下限值，选择分辨率，并设置是否使用唤醒功能（SNOOZE 模式）。

使用 1 位或 8 位存储器操作指令来设置 ADM2 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 12-7. A/D 转换器模式寄存器 2 (ADM2)的格式 (1/2)

地址: F0010H 复位后: 00H R/W

符号	7	6	5	4	<3>	<2>	1	<0>
ADM2	ADREFP1	ADREFP0	ADREFM	0	ADRCK	AWC	0	ADTYP

ADREFP1	ADREFP0	A/D 转换器的+ 侧的基准电压源的选择
0	0	由 V _{DD} 提供
0	1	由 P20/AV _{REFP} /ANI0 提供
1	0	由内部基准电压(1.45 V)提供 ^注
1	1	禁止设置

- 当 ADREFP1 或 ADREFP0 位被改写时，必须按照以下步骤进行设置：
 - 设置 ADCE = 0
 - 更改 ADREFP1 和 ADREFP0 的值
 - 稳定等待时间 (A)
 - 设置 ADCE = 1
 - 稳定等待时间 (B)
 当 ADREFP1 和 ADREFP0 被设置为 1 和 0 时，更改设置为 A = 5 μs, B = 1 μs。
 当 ADREFP1 和 ADREFP0 被设置为 0 和 0 或者 0 和 1 时，A 不需等待，而且 B = 1 μs。
- 当 ADREFP1 和 ADREFP0 分别被设置为 1 和 0 时，不能对温度传感器输出和内部基准电压输出执行 A/D 转换。
 当 ADISS = 0 时，必须执行 A/D 转换。

ADREFM	A/D 转换器的 - 侧的基准电压源的选择
0	由 V _{SS} 提供
1	由 P21/AV _{REFM} /ANI1 提供

注 此设置仅能在 HS (高速主) 模式下使用。

- 注意事项 1. 仅在停止转换操作时（即 A/D 转换器模式寄存器 0(ADM0)的 ADCS 位为 0 时）改写 ADM2 寄存器的值。
2. 如果切换至 STOP 模式或者当 CPU 为副系统时钟操作时切换至 HALT 模式，不要将 ADREFP1 位设置为 1。如果把 ADREFP1 位设置为 1，当 CPU 在主系统时钟操作时切换至 HALT 模式，温度传感器操作电流(32.4.2 电源电流特性 (I_{TMPs})) 就会被加到消耗电流上。
3. 使用 AV_{REFP} 和 AV_{REFM} 时，设置 ANI0 和 ANI1 作为模拟输入通道，并且通过端口模式寄存器设置它们为输入模式。

图 12-7. A/D 转换器模式寄存器 2 (ADM2)的格式 (2/2)

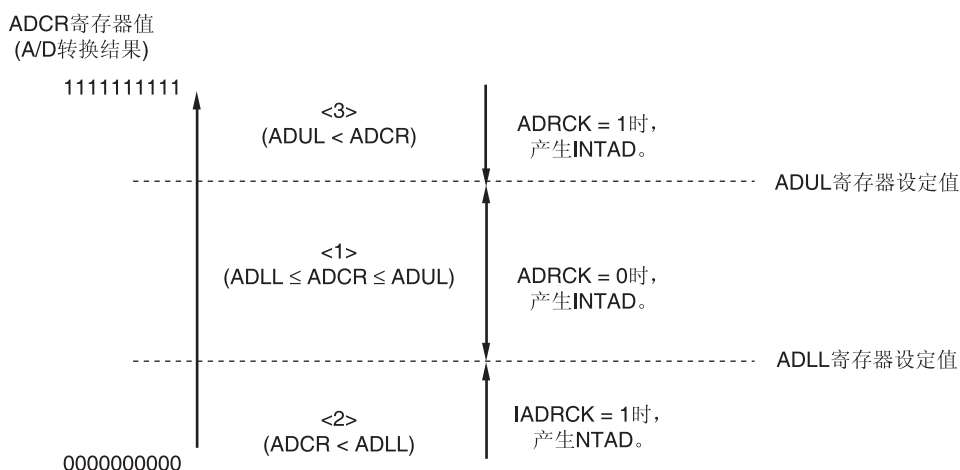
地址: F0010H 复位后: 00H R/W

符号	7	6	5	4	<3>	<2>	1	<0>
ADM2	ADREFP1	ADREFP0	ADREFM	0	ADRCK	AWC	0	ADTYP
ADRCK	检验转换结果的上限值和下限值							
0	当 ADLL 寄存器 \leq ADCR 寄存器 \leq ADUL 寄存器(<1>)时, 输出中断信号(INTAD)。							
1	当 ADCR 寄存器<ADLL 寄存器(<2>) 或 ADUL 寄存器< ADCR 寄存器(<3>)时, 输出中断信号(INTAD)。							
<1> 至<3>的中断信号(INTAD)产生范围如图 12-8 所示。								
AWC	唤醒功能(SNOOZE 模式)的设置							
0	不使用 SNOOZE 模式功能。							
1	使用 SNOOZE 模式功能。							
通过 STOP 模式下的硬件触发信号, 退出 STOP 模式, 并且在不操作 CPU 的情况下执行 A/D 转换 (SNOOZE 模式)。								
<ul style="list-style-type: none"> • 仅在选择高速片上振荡器时钟作为 CPU/外围硬件时钟(f_{CLK})时可以选择 SNOOZE 模式功能。如果选用任何其他时钟, 则禁止选择该模式。 • 禁止在软件触发模式或硬件触发无等待模式时使用 SNOOZE 模式功能。 • 使用 SNOOZE 模式功能时, 须设置硬件触发间隔为“切换到 SNOOZE 模式的时间^注 + A/D 电源稳定等待时间 + A/D 转换时间 + 2 f_{CLK} 时钟”或更长时间。 • 即便使用 SNOOZE 模式, 请在正常操作模式中设置 AWC 位为 0, 并且在切换至 STOP 模式之前将 AWC 位更改为 1。此外, 从 STOP 模式恢复到正常操作模式后, 必须将 AWC 位更改为 0。如果保持 AWC 位为 1, 则其后不论是 SNOOZE 模式还是正常操作模式, A/D 转换都不会正常启动。 								
ADTYP	A/D 转换分辨率的选择							
0	10 位分辨率							
1	8 位分辨率							

注 参阅 21.2.3 SNOOZE 模式中的“从 STOP 至 SNOOZE”。

注意事项 仅在停止转换操作时 (即 A/D 转换器模式寄存器 0(ADM0)的 ADCS 位为 0 时) 改写 ADM2 寄存器的值。

图 12-8. ADRCK 位中断信号产生范围



备注 如果不产生 INTAD, 则 A/D 转换结果不会存储在 ADCR 或 ADCRH 寄存器中。

(7) 模拟输入通道选择寄存器(ADS)

该寄存器用于指定执行 A/D 转换的模拟电压的输入通道。

使用 1 位或 8 位存储器操作指令设置 ADS 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 12-11. 模拟输入通道选择寄存器(ADS)的格式

地址: FFF31H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ADS	ADISS	0	0	ADS4	ADS3	ADS2	ADS1	ADS0

○ 选择模式 (ADMD = 0)

ADISS	ADS4	ADS3	ADS2	ADS1	ADS0	模拟输入通道	输入源
0	0	0	0	0	0	ANI0	P20/ANI0/AV _{REFP} 引脚
0	0	0	0	0	1	ANI1	P21/ANI1/AV _{REFM} 引脚
0	0	0	0	1	0	ANI2	P22/ANI2 引脚
0	0	0	0	1	1	PGAOUT	PGA 输出信号引脚
0	0	0	1	0	0	ANI4	P24/ANI4 引脚
0	0	0	1	0	1	ANI5	P25/ANI5 引脚
0	0	0	1	1	0	ANI6	P26/ANI6 引脚
0	0	0	1	1	1	ANI7	P27/ANI7 引脚
0	1	0	0	0	0	ANI16	P03/ANI16 引脚
0	1	0	0	0	1	ANI17	P02/ANI17 引脚
0	1	0	0	1	0	ANI18	P147/ANI18 引脚
0	1	0	0	1	1	ANI19	P120/ANI19 引脚
1	0	0	0	0	0	-	温度传感器 0 输出 ^注
1	0	0	0	0	1	-	内部基准电压输出(1.45 V) ^注
其他						禁止设置	

○ 扫描模式 (ADMD = 1)

ADS4	ADS3	ADS2	ADS1	ADS0	模拟输入通道			
					扫描 0	扫描 1	扫描 2	扫描 3
0	0	0	0	0	ANI0	ANI1	ANI2	PGAOUT
0	0	0	0	1	ANI1	ANI2	PGAOUT	ANI4
0	0	0	1	0	ANI2	PGAOUT	ANI4	ANI5
0	0	0	1	1	PGAOUT	ANI4	ANI5	ANI6
0	0	1	0	0	ANI4	ANI5	ANI6	ANI7
其他					禁止设置			

注 此设置仅能在 HS (高速主) 模式下使用。

(注意事项和备注如下页所示。)

- 注意事项
1. 必须将位 5 和位 6 清除为 0。
 2. 对于通过 ADPC 和 PMC 寄存器设置为模拟输入的端口，使用端口模式寄存器 0、2、12 和 14 (PM0, PM2, PM12, PM14) 设置其为输入模式。
 3. 不要用 ADS 寄存器设置已经由 A/D 端口配置寄存器(ADPC)设置为数字输入/输出的引脚。
 4. 不要用 ADS 寄存器设置已经由端口模式控制寄存器 0、12 或 14 (PMC0, PMC12, PMC14)设置为数字输入/输出的引脚。
 5. 仅在停止转换操作时（即 A/D 转换器模式寄存器 0(ADM0)的 ADCE 位为 0 时）改写 ADISS 位的值。
 6. 如果使用 AV_{REFP} 作为 A/D 转换器的+侧基准电压源，则不得选择 ANI0 作为 A/D 转换通道。
 7. 如果使用 AV_{REFM} 作为 A/D 转换器的-侧基准电压源，则不得选择 ANI1 作为 A/D 转换通道。
 8. 如果将 ADISS 设置为 1，则不得将内部基准电压 (1.45 V)用作+侧基准电压源。
 9. 当 PGAOUT 被选择作为模拟输入时，设置 PGA 操作后再设置 ADS 寄存器 (参阅第十三章 可编程增益放大器)。
 10. 如果切换至 STOP 模式或者当 CPU 在副系统时钟中操作时切换至 HALT 模式，不要将 ADISS 位设置为 1。如果把 ADISS 位设置为 1，当 CPU 在主系统时钟操作时切换至 HALT 模式，32.3.2 电源电流特性中所示的 A/D 转换基准电压电流 (I_{ADREF})就会被加到消耗电流上。
 11. 如果使用的产品中不存在对应的 ANI 引脚，请忽略转换结果。

(8) 转换结果比较上限值设置寄存器(ADUL)

该寄存器用于设置检验 A/D 转换结果的上限值。

将 A/D 转换结果与 ADUL 寄存器值进行比较，在 A/D 转换器模式寄存器 2(ADM2)的 ADRCK 位设置的范围内（如图 12-8 所示），控制中断信号(INTAD)的产生。

使用 8 位存储器操作指令来设置 ADUL 寄存器。

产生复位信号后，该寄存器被设置为 FFH。

注意事项 选择 10 位分辨率 A/D 转换时，将 10 位 A/D 转换结果寄存器(ADCR)的高 8 位与 ADUL 寄存器进行比较。

图 12-12. 转换结果比较上限值设置寄存器(ADUL)的格式

地址: F0011H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
ADUL	ADUL7	ADUL6	ADUL5	ADUL4	ADUL3	ADUL2	ADUL1	ADUL0

(9) 转换结果比较下限值设置寄存器(ADLL)

该寄存器用于设置检验 A/D 转换结果的下限值。

将 A/D 转换结果与 ADLL 寄存器值进行比较，在 A/D 转换器模式寄存器 2(ADM2)的 ADRCK 位设置的范围内（如图 12-8 所示），控制中断信号(INTAD)的产生。

使用 8 位存储器操作指令来设置 ADLL 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 12-13. 转换结果比较上限值设置寄存器(ADUL)的格式

地址: F0012H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ADLL	ADLL7	ADLL6	ADLL5	ADLL4	ADLL3	ADLL2	ADLL1	ADLL0

注意事项 选择 10 位分辨率 A/D 转换时，将 10 位 A/D 转换结果寄存器(ADCR)的高 8 位与 ADLL 寄存器进行比较。

(10) A/D 测试寄存器(ADTES)

在 A/D 测试功能中，该寄存器用于选择 A/D 转换器的+侧基准电压(AV_{REFP})、-侧基准电压(AV_{REFM})、模拟输入通道(AN_{Ixx})、或者 PGAOUT 作为 A/D 转换目标。

使用 8 位存储器操作指令来设置 ADTES 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 12-14. A/D 测试寄存器(ADTES)的格式

地址: F0013H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ADTES	0	0	0	0	0	0	ADTES1	ADTES0

ADTES1	ADTES0	A/D 转换目标
0	0	AN _{Ixx} 、PGAOUT (由模拟输入通道选择寄存器(ADS)设置。)
1	0	AV _{REFM}
1	1	AV _{REFP}
其他		禁止设置

注意事项 关于 A/D 测试功能的详细信息，请参阅第二十五章 安全功能。

(11) A/D 端口配置寄存器(ADPC)

该寄存器将 ANI0/P20, ANI1/P21, ANI2/CMP0P/P22, ANI4/CMP1P/P24 至 ANI7/CMP4P/P27 引脚和 PGAOUT 引脚 (内部引脚)切换为 A/D 转换器的模拟输入或端口的数字输入/输出。

使用 8 位存储器操作指令来设置 ADPC 寄存器。

产生复位信号后, 该寄存器被清除为 00H。

图 12-15. A/D 端口配置寄存器(ADPC)的格式

地址: F0076H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ADPC	0	0	0	0	ADPC3	ADPC2	ADPC1	ADPC0

ADPC3	ADPC2	ADPC1	ADPC0	模拟输入(A)/数字输入/输出(D)的切换							
				ANI7/ CMP4P/P27	ANI6/ CMP3P/P26	ANI5/ CMP2P/P25	ANI4/ CMP1P/P24	PGAOUT ^注	ANI2/ CMP0P/P22	ANI1/P21	ANI0/P20
0	0	0	0	A	A	A	A	A	A	A	A
0	0	0	1	D	D	D	D	D	D	D	D
0	0	1	0	D	D	D	D	D	D	D	A
0	0	1	1	D	D	D	D	D	D	A	A
0	1	0	0	D	D	D	D	D	A	A	A
0	1	0	1	D	D	D	D	A	A	A	A
0	1	1	0	D	D	D	A	A	A	A	A
0	1	1	1	D	D	A	A	A	A	A	A
1	0	0	0	D	A	A	A	A	A	A	A
1	0	0	1	A	A	A	A	A	A	A	A
1	1	1	1	A	A	A	A	A	A	A	A
其他				禁止设置							

注 这是一个可编程增益放大器的内部输出引脚。将可编程增益放大器的输出信号用作 A/D 转换器的模拟输入通道时, 设置 ADPC 等于或大于 0101B。

- 注意事项
1. 通过端口模式寄存器 2 (PM2)将用于 A/D 转换的端口设置为输入模式。
 2. 对于由 ADPC 寄存器设置为数字输入/输出的引脚, 不要用模拟输入通道选择寄存器(ADS)进行设置。
 3. 使用 AVREFP 和 AVREFM 时, 指定 ANI0 和 ANI1 作为模拟输入通道, 并且使用端口模式寄存器设置它们为输入模式。

(12) 端口模式控制寄存器 0, 12, 14 (PMC0, PMC12, PMC14)

使用该寄存器以 1 位为单位设置端口 0、12 或 14 的数字输入/输出或模拟输入。

当使用 ANI16/CMP5P/RxD1/P03、ANI17/TxD1/P02、ANI18/CMPCOM/P147 或 ANI19/P120 引脚作为模拟输入端口时，将 PMC03、PMC02、PMC147 或 PMC120 位设置为 1。

使用 1 位或 8 位存储器操作指令设置该寄存器。

产生复位信号后，该寄存器被设置为 FFH。

图 12-16. 端口模式控制寄存器 0、12 和 14 (PMC0, PMC12, PMC14)的格式

地址: F0060H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PMC0	1	1	1	1	PMC03	PMC02	1	1

地址: F006CH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PMC12	1	1	1	1	1	1	1	PMC120

地址: F006EH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PMC14	PMC147	1	1	1	1	1	1	1

PMCmn	Pmn 引脚数字输入/输出/模拟输入的选择 (m = 0, 12, 14; n = 0, 2, 3, 7)
0	数字输入/输出 (模拟输入以外的复用功能)
1	模拟输入

注意事项 通过端口模式寄存器 x (PMx)将由 PMC 寄存器设置为模拟输入的端口设置为输入模式。

(13) 端口模式寄存器 0、2、12 和 14 (PM0, PM2, PM12, PM14)

当使用 ANI0/AV_{REFP}/P20、ANI1/AV_{REFM}/P21、ANI2/CMP0P/P22、ANI4/CMP1P/P24 至 ANI7/CMP4P/P27、ANI16/CMP5P/RxD1/P03、ANI17/TxD1/P02、ANI18/CMPCOM/P147 或 ANI19/P120 引脚作为模拟输入端口时，将 PM20 至 PM22、PM24 至 PM27、PM03、PM02、PM147 或 PM120 位设置为 1。此时，P20 至 P22、P24 至 P27、P03、P02、P147 或 P120 的输出锁存器可为 0 或 1。

如果 PM20 至 PM22、PM24 至 PM27、PM03、PM02、PM147 和 PM120 位被设置为 0，则其不能用作模拟输入端口。

使用 1 位或 8 位存储器操作指令设置 PM0、PM2、PM12 和 PM14 寄存器。

产生复位信号后，该寄存器被设置为 FFH。

注意事项 读取被设置为模拟输入端口的引脚时，读取值不是引脚电平而是一直为 0。

图 12-17. 端口模式寄存器 0、2、12 和 14 (PM0, PM2, PM12, PM14)的格式

地址: FFF20H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM0	1	PM06	PM05	1	PM03	PM02	1	1

地址: FFF22H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM2	PM27	PM26	PM25	PM24	1	PM22	PM21	PM20

地址: FFF2CH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM12	1	1	1	1	1	1	1	PM120

地址: FFF2EH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM14	PM147	1	1	1	1	1	1	1

PMmn	Pmn 引脚输入/输出模式选择(m = 0, 2, 12, 14; n = 0 至 7)
0	输出模式 (输出缓冲器 on)
1	输入模式 (输出缓冲器 off)

注意事项 1. 必须将 PM0 寄存器的位 0, 1, 4, 7、PM2 寄存器的位 3、PM12 寄存器的位 1 至 7 以及 PM14 寄存器的位 0 至 6 设置为“1”。

对于 32, 30 和 20 引脚产品，在解除复位后，必须通过软件将以下的位设置为输出模式 (通过设置端口寄存器和端口模式寄存器为 0 来指定)。

32 引脚产品: PM0 寄存器的位 2 和位 3

30 引脚产品: PM0 寄存器的位 5 和位 6

20 引脚产品: PM0 寄存器的位 2, 3, 5 和 6, PM2 寄存器的位 6 和 7, 以及 PM12 寄存器的位 0

2. 使用 AV_{REFP} 和 AV_{REFM} 时，设置 ANI0 和 ANI1 为模拟输入，并且使用端口模式寄存器设置它们为输入模式。

如下所示，ANI0/P20 至 ANI2/P22, ANI4/P24 至 ANI7/P27 引脚以及 PGAOUT/P23 引脚(内部引脚)的功能，因 A/D 端口配置寄存器(ADPC)、模拟输入通道选择寄存器(ADS)和 PM2 寄存器的设置而异。

表 12-4. ANI0/P20 至 ANI2/P22、ANI4/P24 至 ANI7/P27 引脚以及 PGAOUT/P23 引脚(内部引脚)功能的设置

ADPC	PM2	ADS	ANI0/P20至ANI2/P22, ANI4/P24至 ANI7/P27引脚, 和PGAOUT/P23引脚(内部引脚)
数字输入/输出	输入模式	–	数字输入
	输出模式	–	数字输出
模拟输入	输入模式	选择ANI。	模拟输入(将被转换)
		不选择ANI。	模拟输入(不被转换)
	输出模式	选择ANI。	禁止设置
		不选择ANI。	

如下所示，ANI16/P03、ANI17/P02、ANI18/P147 和 ANI19/P120 引脚的功能，因端口模式控制寄存 0、12 和 14 (PMC0、PMC12、PMC14)、模拟输入通道选择寄存器(ADS)、PM0、PM12 和 PM14 寄存器的设置而异。

表 12-5. ANI16/P03、ANI17/P02、ANI18/P147 和 ANI19/P120 引脚功能的设置

PMC0, PMC12和 PMC14	PM0, PM12和PM14	ADS	ANI16/P03, ANI17/P02, ANI18/P147 和ANI19/P120引脚
数字输入/输出	输入模式	–	数字输入
	输出模式	–	数字输出
模拟输入	输入模式	选择ANI。	模拟输入(将被转换)
		不选择ANI。	模拟输入(不被转换)
	输出模式	选择ANI。	禁止设置
		不选择ANI。	

12.4 A/D转换器的转换操作

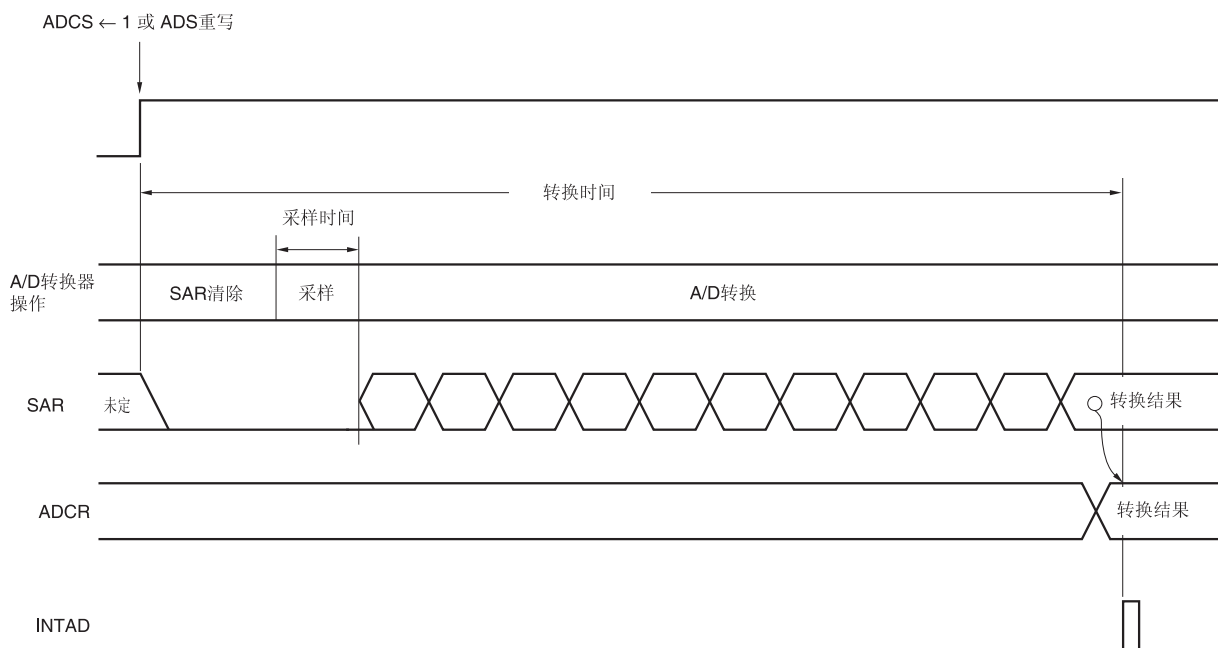
A/D 转换器的转换操作如下所示。

- <1> 通过采样&保持电路对选择的模拟输入通道的输入电压进行采样。
- <2> 当采样进行一定时间时，采样&保持电路被置于保持状态且保持采样电压，直到 A/D 转换操作结束为止。
- <3> 置位逐次逼近寄存器(SAR)的位 9。通过分接选择器将串联电阻串的分接电压设置为 $(1/2) AV_{REF}$ 。
- <4> 使用电压比较器对串联电阻串的分接电压与采样电压之间的电压差进行比较。如果模拟输入大于 $(1/2) AV_{REF}$ ，SAR 寄存器的 MSB 位保持为 1。如果模拟输入小于 $(1/2) AV_{REF}$ ，MSB 位复位为 0。
- <5> 接着，自动将 SAR 寄存器的位 8 设置为 1，并进行下一次比较。根据下述位 9 的当前设置值选择串联电阻串的分接电压。
 - 位 9 = 1: $(3/4) AV_{REF}$
 - 位 9 = 0: $(1/4) AV_{REF}$
 对此分接电压与采样电压进行比较，并对 SAR 寄存器的位 8 进行如下操作：
 - 采样电压 \geq 分接电压：位 8 = 1
 - 采样电压 $<$ 分接电压：位 8 = 0
- <6> 照此继续进行比较，直到 SAR 寄存器的位 0 为止。
- <7> 完成 10 位的比较时，一个有效数字结果值将保持于 SAR 寄存器之中，该结果值被传送至 A/D 转换结果寄存器(ADCR、ADCRH)，并被锁存^{注1}。同时，产生 A/D 转换结束中断请求(INTAD)^注。
- <8> 重复步骤<1>至<7>，直到 ADCS 位被清除至 0^{注2}。将 ADCS 位清除至 0 以停止 A/D 转换器。

- 注 1.** 如果 A/D 转换结果在 ADRCK 位和 ADUL、ADLL 寄存器(参阅图 12-8)指定的范围以外时，则不产生 A/D 转换结果中断请求信号，A/D 转换结果不存储在 ADCR 和 ADCRH 寄存器。
- 2.** 连续转换模式时，不会自动将 ADCS 标志清除至 0。硬件触发无等待模式的单次转换模式时，也不会自动将该标志清除至 0。相反，将继续保持为 1。

- 备注 1.** 有两种 A/D 转换结果寄存器。
- ADCR 寄存器(16 位)： 存储 10 位的 A/D 转换值
 - ADCRH 寄存器(8 位)： 存储 8 位的 A/D 转换值
- 2.** AV_{REF} ： A/D 转换器的+侧基准电压。可以选自 AV_{REFP} 、内部基准电压(1.45 V)以及 V_{DD} 。

图 12-18. A/D 转换器的转换操作 (软件触发模式时)



A/D 转换操作持续进行，直到 A/D 转换器模式寄存器(ADM)的位 7(ADCS)被软件复位(0)为止。

如果在 A/D 转换操作过程中对模拟输入通道选择寄存器(ADS)执行写操作，则会初始化转换操作。如果 ADCS 位被设置为 1，则会再次从头开始转换。

产生复位信号后，A/D 转换结果寄存器(ADCR、ADCRH)被清除为 0000H 或 00H。

12.5 输入电压和转换结果

输入至模拟输入引脚(ANI0 至 ANI2、ANI4 至 ANI7、ANI16 至 ANI19、PGAOUT)的模拟输入电压与理论 A/D 转换结果(存储于 10 位 A/D 转换结果寄存器(ADCR)中)之间的关系可用以下表达式表示。

$$\text{SAR} = \text{INT} \left(\frac{V_{\text{AIN}}}{V_{\text{REF}}} \times 1024 + 0.5 \right)$$

$$\text{ADCR} = \text{SAR} \times 64$$

或

$$\left(\frac{\text{ADCR}}{64} - 0.5 \right) \times \frac{V_{\text{REF}}}{1024} \leq V_{\text{AIN}} < \left(\frac{\text{ADCR}}{64} + 0.5 \right) \times \frac{V_{\text{REF}}}{1024}$$

此时, INT(): 返回括号中数值的整数部分的函数

V_{AIN} : 模拟输入电压

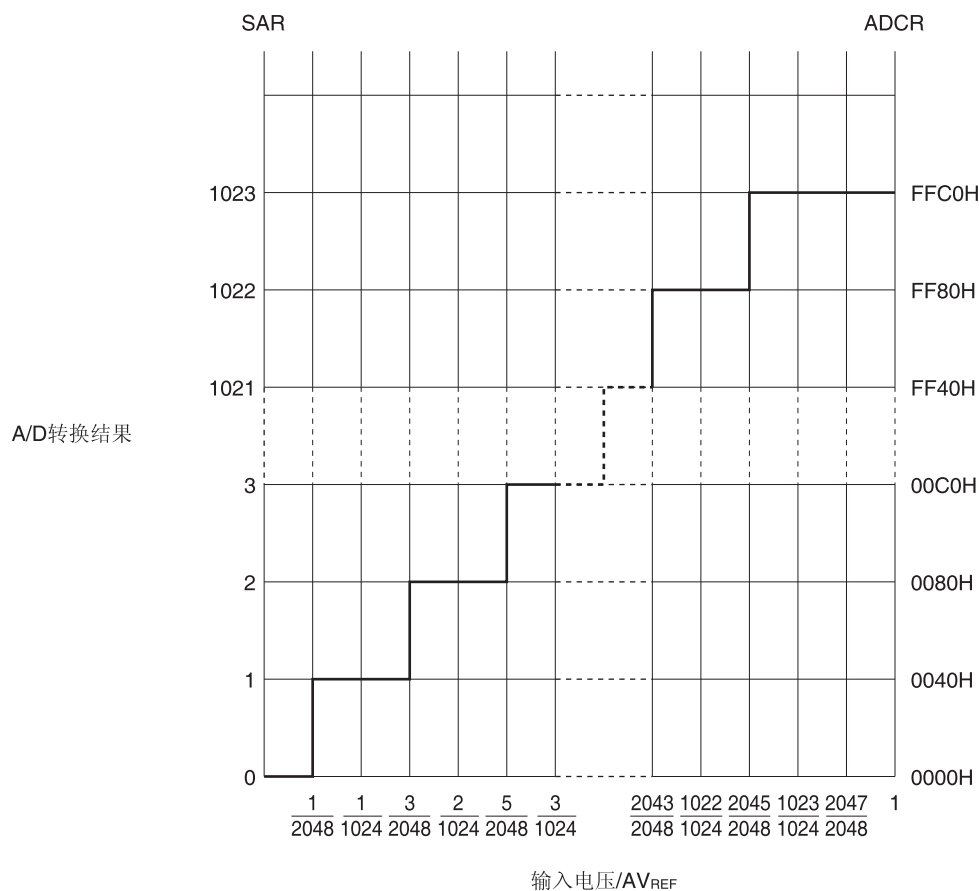
V_{REF} : V_{REF} 引脚电压

ADCR: A/D 转换结果寄存器(ADCR)的值

SAR: 逐次逼近寄存器

图 12-19 表示模拟输入电压与 A/D 转换结果之间的关系。

图 12-19. 模拟输入电压与 A/D 转换结果之间的关系



备注 V_{REF} : A/D 转换器的+侧基准电压。可以选自 V_{REFP} 、内部基准电压(1.45 V)以及 V_{DD} 。

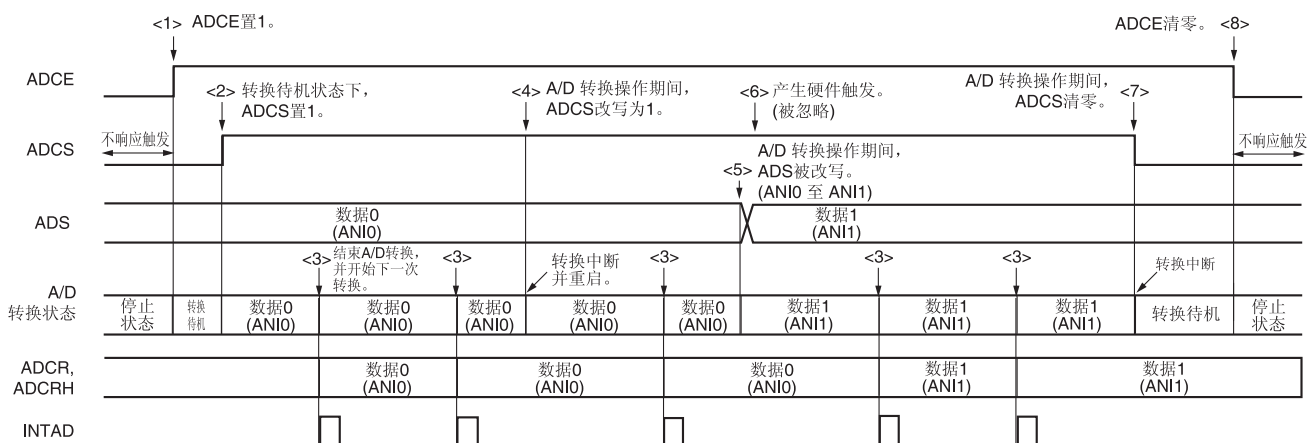
12.6 A/D转换器的操作模式

A/D 转换器的各模式的操作如下所示。另外，各模式的设置步骤请参阅 12.7 A/D 转换器的设置流程图。

12.6.1 软件触发模式（选择模式，连续转换模式）

- <1> 在停止状态下，A/D 转换器模式寄存器 0(ADM0)的 ADCE 位被设置为 1，系统进入 A/D 转换待机状态。
- <2> 软件计数至稳定等待时间(1 μs)之后，ADM0 寄存器的 ADCS 位被设置为 1，以执行由模拟输入通道选择寄存器(ADS)指定的模拟输入的 A/D 转换。
- <3> A/D 转换结束时，转换结果存储于 A/D 转换结果寄存器(ADCR、ADCRH)，并产生 A/D 转换结束中断请求信号(INTAD)。A/D 转换结束后，立即开始下一个 A/D 转换。
- <4> 转换操作过程中用 1 覆盖 ADCS 时，当前 A/D 转换中断，转换重新开始。转换至中途的数据被丢弃。
- <5> 转换操作过程中改写或覆盖 ADS 寄存器的值时，当前 A/D 转换中断，并对 ADS 寄存器重新指定的模拟输入执行 A/D 转换。转换至中途的数据被丢弃。
- <6> 转换操作过程中，即使输入硬件触发也不开始 A/D 转换。
- <7> 转换操作过程中将 ADCS 清除至 0 时，当前 A/D 转换中断，系统进入 A/D 转换待机状态。
- <8> 在 A/D 转换待机状态下，当 ADCE 被清除至 0 时，A/D 转换器进入停止状态。当 ADCE= 0 时，将 ADCS 设置为 1 的操作被忽略，不开始 A/D 转换。

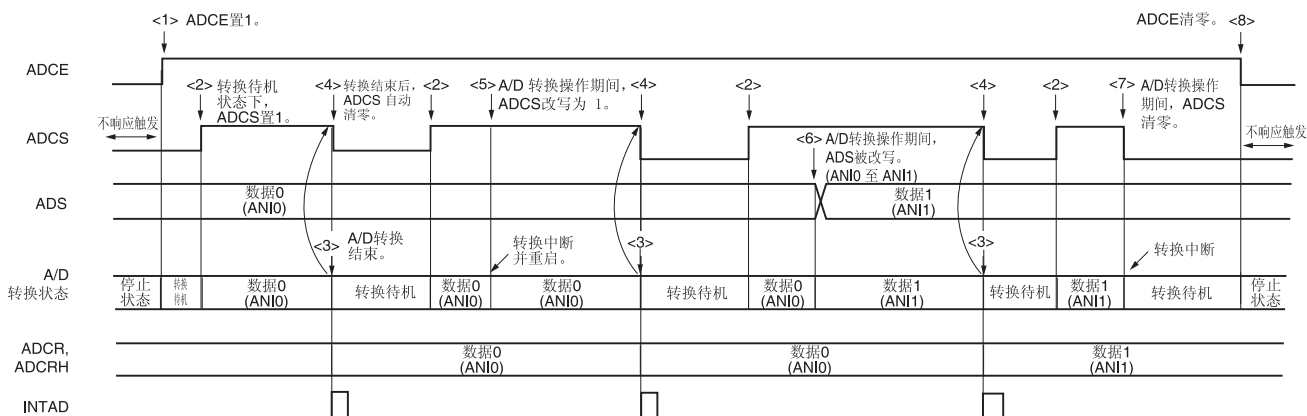
图 12-20. 软件触发模式（选择模式，连续转换模式）操作时序的示例



12.6.2 软件触发模式（选择模式，单次转换模式）

- <1> 在停止状态下，A/D 转换器模式寄存器 0(ADM0)的 ADCE 位被设置为 1，系统进入 A/D 转换待机状态。
- <2> 软件计数至稳定等待时间(1 μs)之后，ADM0 寄存器的 ADCS 位被设置为 1，以执行由模拟输入通道选择寄存器 (ADS)指定的模拟输入的 A/D 转换。
- <3> A/D 转换结束时，转换结果存储于 A/D 转换结果寄存器(ADCR、ADCRH)，并产生 A/D 转换结束中断请求信号 (INTAD)。
- <4> A/D 转换结束后，ADCS 位将被自动清除至 0，系统进入 A/D 转换待机状态。
- <5> 转换操作过程中用 1 覆盖 ADCS 时，当前 A/D 转换中断，转换重新开始。转换至中途的数据被丢弃。
- <6> 转换操作过程中改写或覆盖 ADS 寄存器的值时，当前 A/D 转换中断，并对 ADS 寄存器重新指定的模拟输入执行 A/D 转换。转换至中途的数据被丢弃。
- <7> 转换操作过程中将 ADCS 清除至 0 时，当前 A/D 转换中断，系统进入 A/D 转换待机状态。
- <8> 在 A/D 转换待机状态下，当 ADCE 被清除至 0 时，A/D 转换器进入停止状态。当 ADCE= 0 时，将 ADCS 设置为 1 的操作被忽略，不开始 A/D 转换。另外，即使在 A/D 转换待机状态时输入硬件触发，也不开始 A/D 转换。

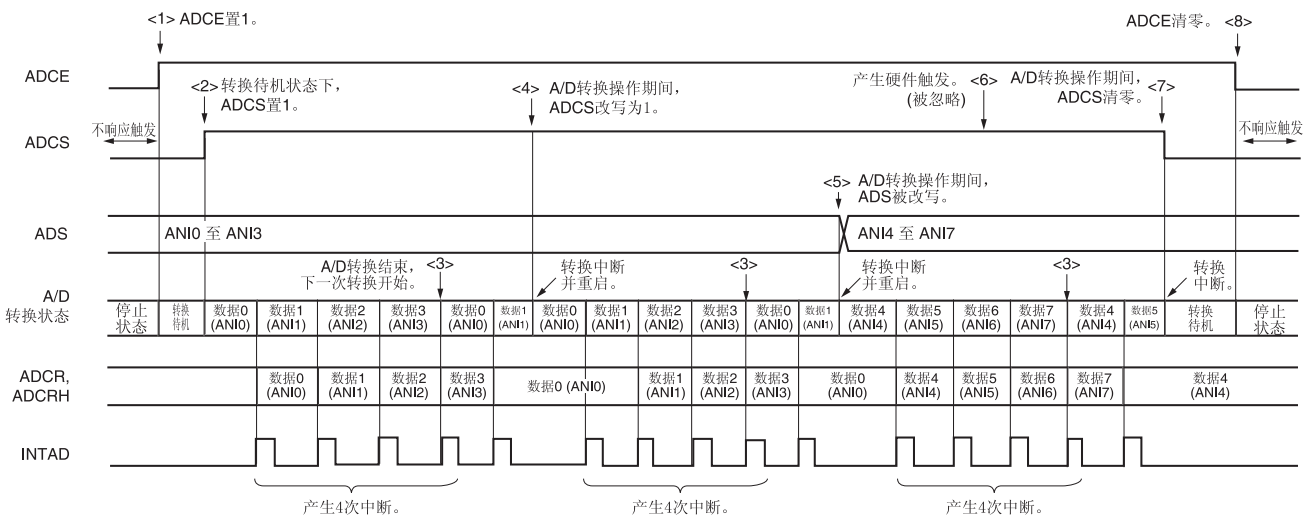
图 12-21. 软件触发模式（选择模式，单次转换模式）操作时序的示例



12.6.3 软件触发模式（扫描模式，连续转换模式）

- <1> 在停止状态下，A/D 转换器模式寄存器 0(ADM0)的 ADCE 位被设置为 1，系统进入 A/D 转换待机状态。
- <2> 软件计数至稳定等待时间(1 μs)之后，ADM0 寄存器的 ADCS 位被设置为 1，对由模拟输入通道选择寄存器道(ADS)指定的从扫描 0 至扫描 3 的四个模拟输入通道执行 A/D 转换。从扫描 0 指定的通道开始，依次对模拟输入通道执行 A/D 转换。
- <3> 依次对四个模拟输入通道执行 A/D 转换，每次转换结束时，将转换结果存储于 A/D 转换结果寄存器(ADCR、ADCRH)，并产生 A/D 转换结束中断请求信号(INTAD)。当四个通道的 A/D 转换结束时，立即自动开始指定通道的下一次 A/D 转换（直到四个通道全部完成为止）。
- <4> 在转换操作过程中用 1 覆盖 ADCS 时，当前 A/D 转换中断，从第一个通道重新开始转换。转换至中途的数据被丢弃。
- <5> 当在转换操作过程中改写或覆盖 ADS 寄存器的值时，当前 A/D 转换中断，并从 ADS 寄存器重新指定的第一个通道开始执行 A/D 转换。转换至中途的数据被丢弃。
- <6> 转换操作过程中，即使输入硬件触发也不开始 A/D 转换。
- <7> 转换操作过程中将 ADCS 清除至 0 时，当前 A/D 转换中断，系统进入 A/D 转换待机状态。
- <8> 在 A/D 转换待机状态下，当 ADCE 被清除至 0 时，A/D 转换器进入停止状态。当 ADCE= 0 时，将 ADCS 设置为 1 的操作被忽略，不开始 A/D 转换。

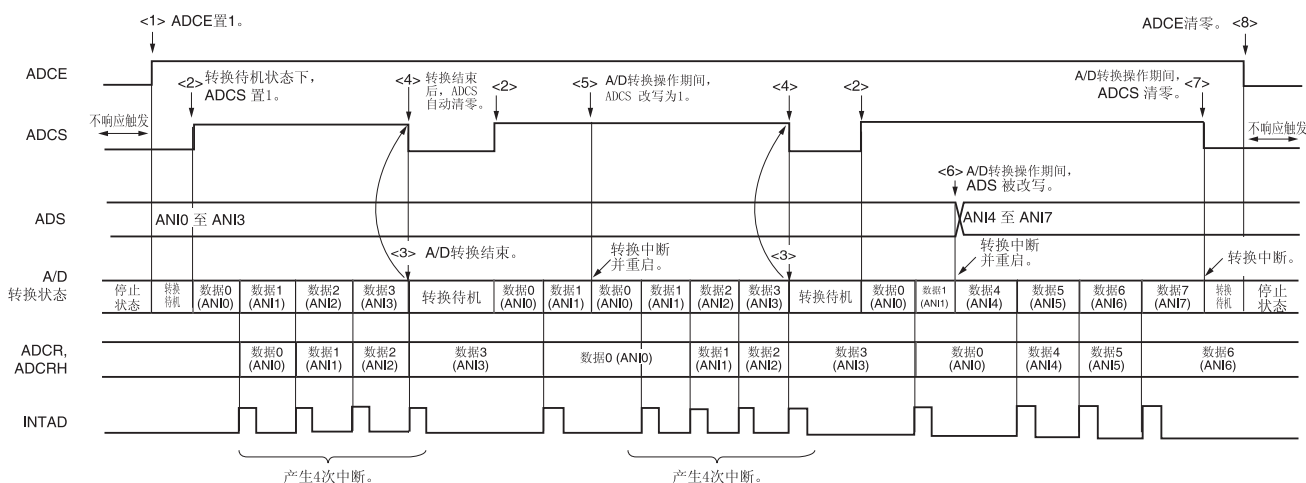
图 12-22. 软件触发模式（扫描模式，连续转换模式）操作时序的示例



12.6.4 软件触发模式（扫描模式，单次转换模式）

- <1> 在停止状态下，A/D 转换器模式寄存器 0(ADM0)的 ADCE 位被设置为 1，系统进入 A/D 转换待机状态。
- <2> 软件计数至稳定等待时间(1 μs)之后，ADM0 寄存器的 ADCS 位被设置为 1，对由模拟输入通道选择寄存器(ADS)指定的从扫描 0 至扫描 3 的四个模拟输入通道执行 A/D 转换。从扫描 0 指定的通道开始，依次对模拟输入通道执行 A/D 转换。
- <3> 依次对四个模拟输入通道执行 A/D 转换，每次转换结束时，将转换结果存储于 A/D 转换结果寄存器(ADCR、ADCRH)，并产生 A/D 转换结束中断请求信号(INTAD)。
- <4> 四个通道的 A/D 转换结束后，ADCS 位被自动清除至 0，系统进入 A/D 转换待机状态。
- <5> 在转换操作过程中用 1 覆盖 ADCS 时，当前 A/D 转换中断，从第一个通道重新开始转换。转换至中途的数据被丢弃。
- <6> 当在转换操作过程中改写或覆盖 ADS 寄存器的值时，当前 A/D 转换中断，并从 ADS 寄存器重新指定的第一个通道开始执行 A/D 转换。转换至中途的数据被丢弃。
- <7> 转换操作过程中将 ADCS 清除至 0 时，当前 A/D 转换中断，系统进入 A/D 转换待机状态。
- <8> 在 A/D 转换待机状态下，当 ADCE 被清除至 0 时，A/D 转换器进入停止状态。当 ADCE= 0 时，将 ADCS 设置为 1 的操作被忽略，不开始 A/D 转换。另外，即使在 A/D 转换待机状态时输入硬件触发，也不开始 A/D 转换。

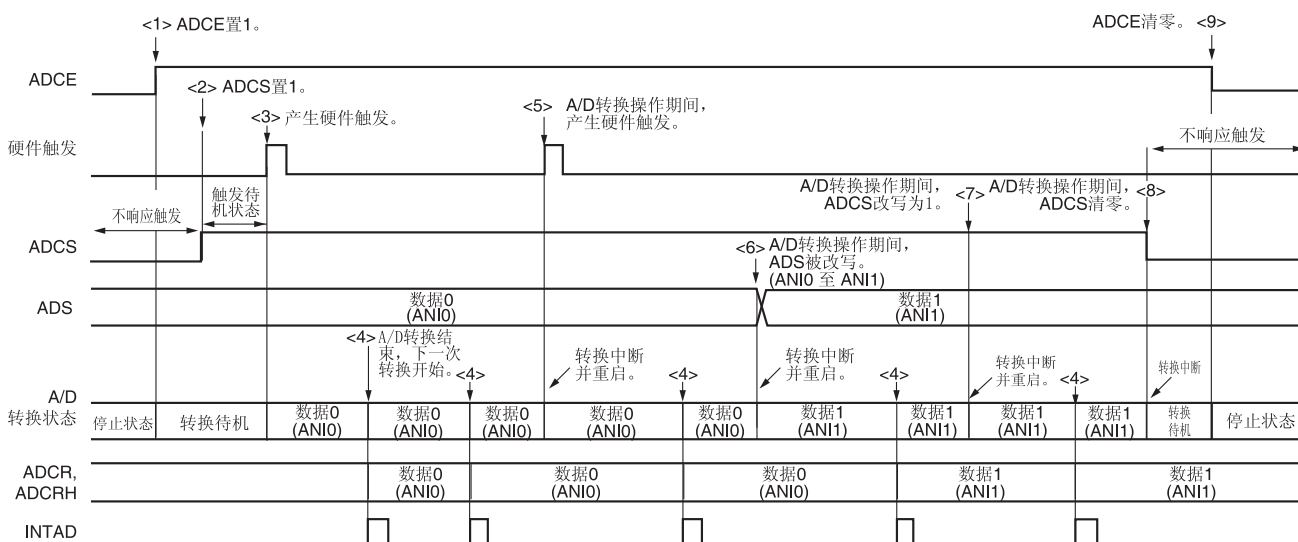
图 12-23. 软件触发模式（扫描模式，单次转换模式）操作时序的示例



12.6.5 硬件触发无等待模式（选择模式，连续转换模式）

- <1> 在停止状态下，A/D 转换器模式寄存器 0(ADM0)的 ADCE 位被设置为 1，系统进入 A/D 转换待机状态。
- <2> 软件计数至稳定等待时间(1 μs)之后，ADM0 寄存器的 ADCS 位被设置为 1，以使系统处于硬件触发待机状态（此时不开始转换）。请注意，处于该状态时，即使将 ADCS 设置为 1，也不开始 A/D 转换。
- <3> 如果 ADCS = 1 时输入硬件触发，则对模拟输入通道选择寄存器(ADS)指定的模拟输入执行 A/D 转换。
- <4> A/D 转换结束时，转换结果存储于 A/D 转换结果寄存器(ADCR、ADCRH)，并产生 A/D 转换结束中断请求信号(INTAD)。A/D 转换结束后，立即开始下一个 A/D 转换。
- <5> 在转换操作过程中输入硬件触发时，当前 A/D 转换中断，重新开始转换。转换至中途的数据被丢弃。
- <6> 转换操作过程中改写或覆盖 ADS 寄存器的值时，当前 A/D 转换中断，并对 ADS 寄存器重新指定的模拟输入执行 A/D 转换。转换至中途的数据被丢弃。
- <7> 转换操作过程中用 1 覆盖 ADCS 时，当前 A/D 转换中断，转换重新开始。转换至中途的数据被丢弃。
- <8> 转换操作过程中将 ADCS 清除至 0 时，当前 A/D 转换中断，系统进入 A/D 转换待机状态。然而，该状态下 A/D 转换器不进入停止状态。
- <9> 在 A/D 转换待机状态下，当 ADCE 被清除至 0 时，A/D 转换器进入停止状态。当 ADCS = 0 时，输入硬件触发的操作将被忽略，不开始 A/D 转换。

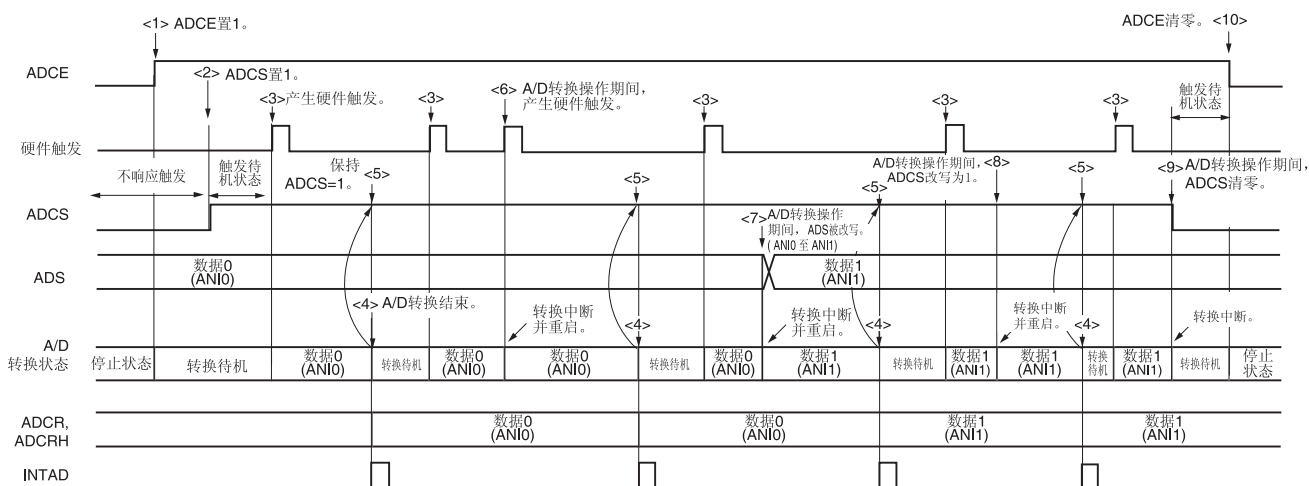
图 12-24. 硬件触发无等待模式（选择模式，连续转换模式）操作时序的示例



12.6.6 硬件触发无等待模式（选择模式，单次转换模式）

- <1> 在停止状态下，A/D 转换器模式寄存器 0(ADM0)的 ADCE 位被设置为 1，系统进入 A/D 转换待机状态。
- <2> 软件计数至稳定等待时间(1 μs)之后，ADM0 寄存器的 ADCS 位被设置为 1，以使系统处于硬件触发待机状态（此时不开始转换）。请注意，处于该状态时，即使将 ADCS 设置为 1，也不开始 A/D 转换。
- <3> 如果 ADCS = 1 时输入硬件触发，则对模拟输入通道选择寄存器(ADS)指定的模拟输入执行 A/D 转换。
- <4> A/D 转换结束时，转换结果存储于 A/D 转换结果寄存器(ADCR、ADCRH)，并产生 A/D 转换结束中断请求信号(INTAD)。
- <5> A/D 转换结束后，ADCS 位保持 1，系统进入 A/D 转换待机状态。
- <6> 在转换操作过程中输入硬件触发时，当前 A/D 转换中断，重新开始转换。转换至中途的数据被丢弃。
- <7> 转换操作过程中改写或覆盖 ADS 寄存器的值时，当前 A/D 转换中断，并对 ADS 寄存器重新指定的模拟输入执行 A/D 转换。转换至中途的数据被丢弃。
- <8> 转换操作过程中用 1 覆盖 ADCS 时，当前 A/D 转换中断，转换重新开始。转换至中途的数据被丢弃。
- <9> 转换操作过程中将 ADCS 清除至 0 时，当前 A/D 转换中断，系统进入 A/D 转换待机状态。然而，该状态下 A/D 转换器不进入停止状态。
- <10> 在 A/D 转换待机状态下，当 ADCE 被清除至 0 时，A/D 转换器进入停止状态。当 ADCS = 0 时，输入硬件触发的操作将被忽略，不开始 A/D 转换。

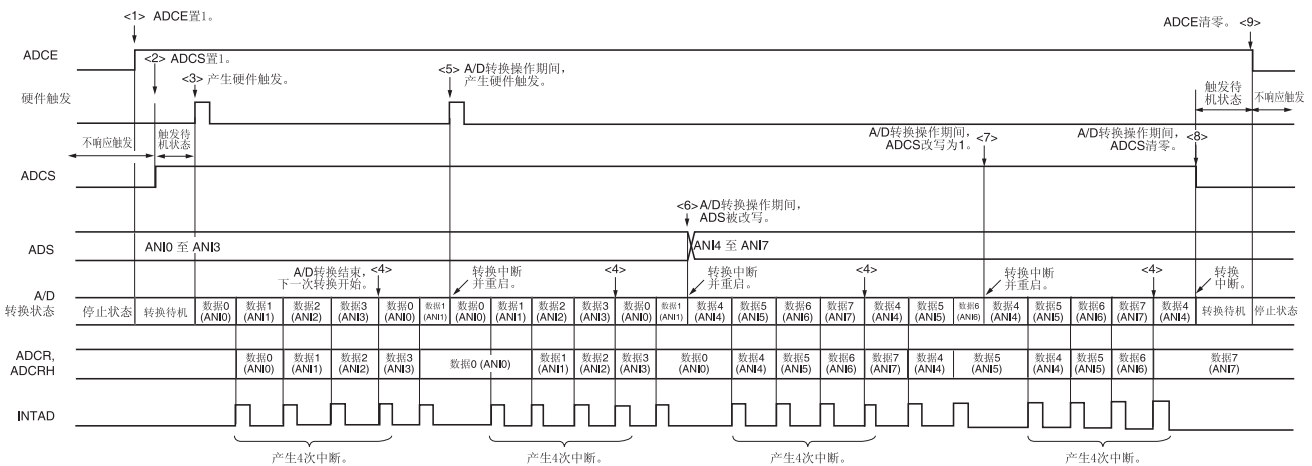
图 12-25. 硬件触发无等待模式（选择模式，单次转换模式）操作时序的示例



12.6.7 硬件触发无等待模式（扫描模式，连续转换模式）

- <1> 在停止状态下，A/D 转换器模式寄存器 0(ADM0)的 ADCE 位被设置为 1，系统进入 A/D 转换待机状态。
- <2> 软件计数至稳定等待时间(1 μs)之后，ADM0 寄存器的 ADCS 位被设置为 1，以使系统处于硬件触发待机状态（此时不开始转换）。请注意，处于该状态时，即使将 ADCS 设置为 1，也不开始 A/D 转换。
- <3> 如果 ADCS = 1 时输入硬件触发，则对由模拟输入通道选择寄存器(ADS)指定的从扫描 0 至扫描 3 的四个模拟输入通道执行 A/D 转换。从扫描 0 指定的通道开始，依次对模拟输入通道执行 A/D 转换。
- <4> 依次对四个模拟输入通道执行 A/D 转换，每次转换结束时，将转换结果存储于 A/D 转换结果寄存器(ADCR、ADCRH)，并产生 A/D 转换结束中断请求信号(INTAD)。四个通道的 A/D 转换结束时，立即自动开始指定通道的下一次 A/D 转换。
- <5> 在转换操作过程中输入硬件触发时，当前 A/D 转换中断，从第一个通道重新开始转换。转换至中途的数据被丢弃。
- <6> 当在转换操作过程中改写或覆盖 ADS 寄存器的值时，当前 A/D 转换中断，并从 ADS 寄存器重新指定的第一个通道开始执行 A/D 转换。转换至中途的数据被丢弃。
- <7> 转换操作过程中用 1 覆盖 ADCS 时，当前 A/D 转换中断，转换重新开始。转换至中途的数据被丢弃。
- <8> 转换操作过程中将 ADCS 清除至 0 时，当前 A/D 转换中断，系统进入 A/D 转换待机状态。然而，该状态下 A/D 转换器不进入停止状态。
- <9> 在 A/D 转换待机状态下，当 ADCE 被清除至 0 时，A/D 转换器进入停止状态。当 ADCE= 0 时，将 ADCS 设置为 1 的操作被忽略，不开始 A/D 转换。

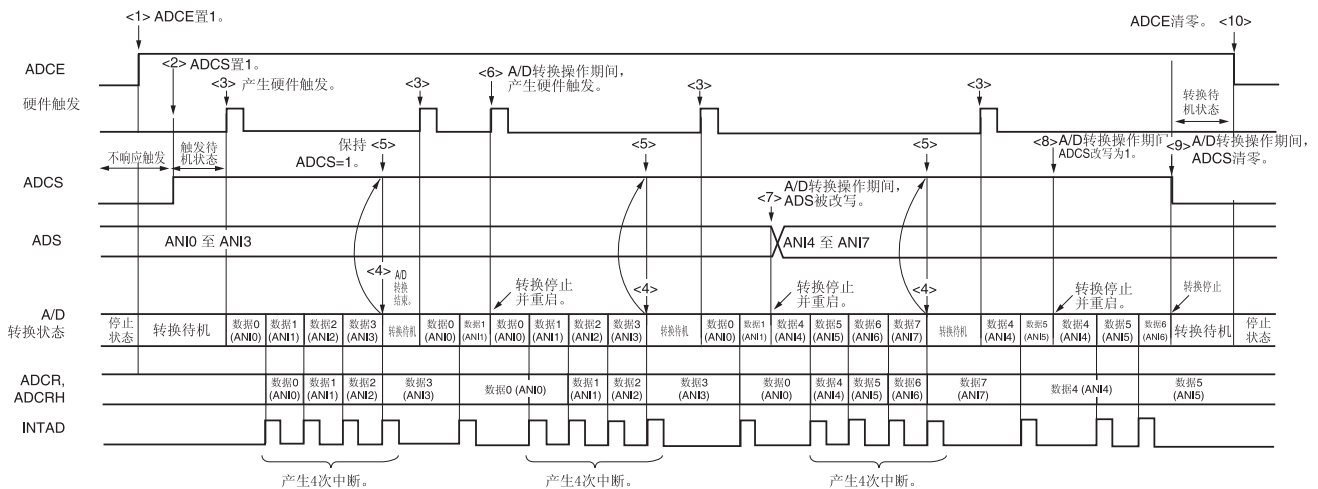
图 12-26. 硬件触发无等待模式（扫描模式，连续转换模式）操作时序的示例



12.6.8 硬件触发无等待模式（扫描模式，单次转换模式）

- <1> 在停止状态下，A/D 转换器模式寄存器 0(ADM0)的 ADCE 位被设置为 1，系统进入 A/D 转换待机状态。
- <2> 软件计数至稳定等待时间(1 μs)之后，ADM0 寄存器的 ADCS 位被设置为 1，以使系统处于硬件触发待机状态（此时不开始转换）。请注意，处于该状态时，即使将 ADCS 设置为 1，也不开始 A/D 转换。
- <3> 如果 ADCS = 1 时输入硬件触发，则对由模拟输入通道选择寄存器(ADS)指定的从扫描 0 至扫描 3 的四个模拟输入通道执行 A/D 转换。从扫描 0 指定的通道开始，依次对模拟输入通道执行 A/D 转换。
- <4> 依次对四个模拟输入通道执行 A/D 转换，每次转换结束时，将转换结果存储于 A/D 转换结果寄存器(ADCR、ADCRH)，并产生 A/D 转换结束中断请求信号(INTAD)。
- <5> 当四个通道的 A/D 转换结束后，ADCS 位保持 1，系统进入 A/D 转换待机状态。
- <6> 在转换操作过程中输入硬件触发时，当前 A/D 转换中断，从第一个通道重新开始转换。转换至中途的数据被丢弃。
- <7> 当在转换操作过程中改写或覆盖 ADS 寄存器的值时，当前 A/D 转换中断，并从 ADS 寄存器重新指定的第一个通道开始执行 A/D 转换。转换至中途的数据被丢弃。
- <8> 在转换操作过程中用 1 覆盖 ADCS 时，当前 A/D 转换中断，从第一个通道重新开始转换。转换至中途的数据被丢弃。
- <9> 转换操作过程中将 ADCS 清除至 0 时，当前 A/D 转换中断，系统进入 A/D 转换待机状态。然而，该状态下 A/D 转换器不进入停止状态。
- <10> 在 A/D 转换待机状态下，当 ADCE 被清除至 0 时，A/D 转换器进入停止状态。当 ADCS = 0 时，输入硬件触发的操作将被忽略，不开始 A/D 转换。

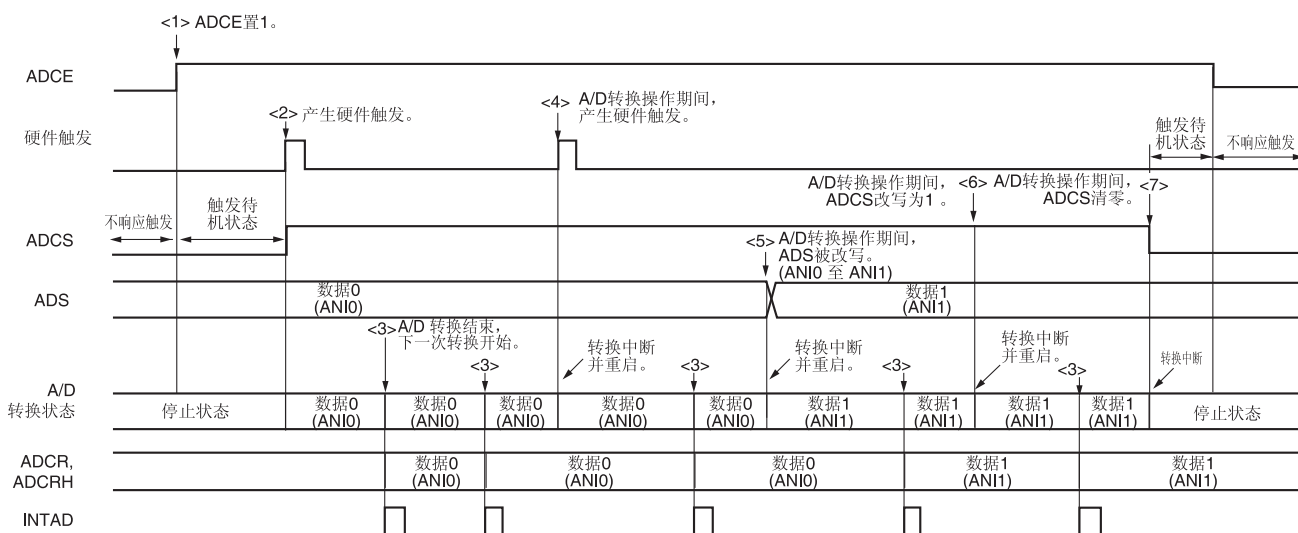
图 12-27. 硬件触发无等待模式（扫描模式，单次转换模式）操作时序的示例



12.6.9 硬件触发等待模式（选择模式，连续转换模式）

- <1> 在停止状态下，A/D 转换器模式寄存器 0(ADM0)的 ADCE 位被设置为 1，系统进入硬件触发待机状态。
- <2> 如果在硬件触发待机状态时输入硬件触发，则对模拟输入通道选择寄存器(ADS)指定的模拟输入执行 A/D 转换。随着硬件触发输入，ADM0 寄存器的 ADCS 位自动被设置为 1。
- <3> A/D 转换结束时，转换结果存储于 A/D 转换结果寄存器(ADCR、ADCRH)，并产生 A/D 转换结束中断请求信号(INTAD)。A/D 转换结束后，立即开始下一个 A/D 转换。(此时不需要硬件触发。)
- <4> 在转换操作过程中输入硬件触发时，当前 A/D 转换中断，重新开始转换。转换至中途的数据被丢弃。
- <5> 转换操作过程中改写或覆盖 ADS 寄存器的值时，当前 A/D 转换中断，并对 ADS 寄存器重新指定的模拟输入执行 A/D 转换。转换至中途的数据被丢弃。
- <6> 转换操作过程中用 1 覆盖 ADCS 时，当前 A/D 转换中断，转换重新开始。转换至中途的数据被丢弃。
- <7> 转换操作过程中 ADCS 被清除至 0 时，当前 A/D 转换中断，系统进入硬件触发待机状态，而且 A/D 转换器进入停止状态。当 ADCE= 0 时，输入硬件触发的操作被忽略，不开始 A/D 转换。

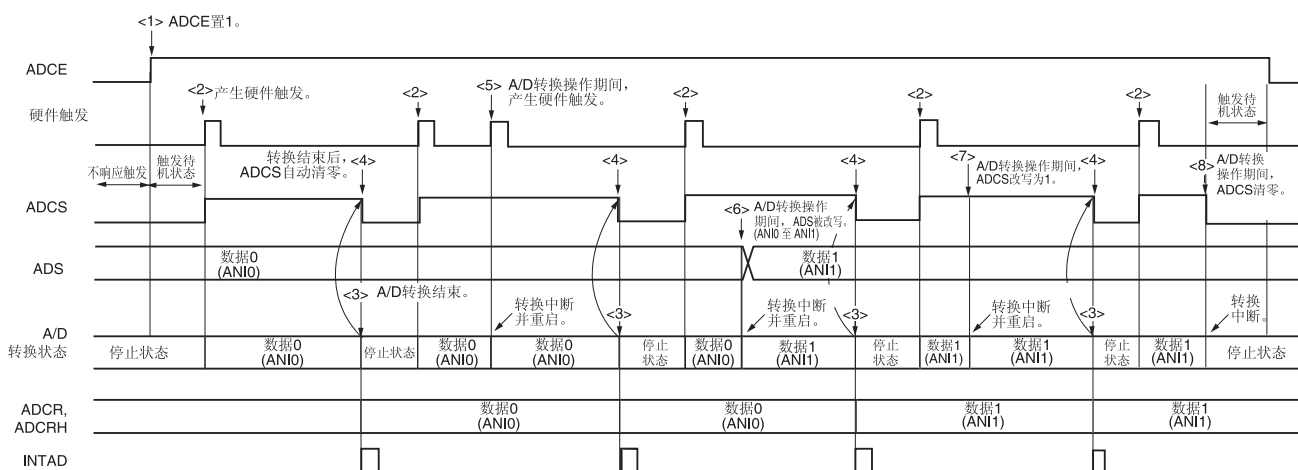
图 12-28. 硬件触发等待模式（选择模式，连续转换模式）操作时序的示例



12.6.10 硬件触发等待模式（选择模式，单次转换模式）

- <1> 在停止状态下，A/D 转换器模式寄存器 0(ADM0)的 ADCE 位被设置为 1，系统进入硬件触发待机状态。
- <2> 如果在硬件触发待机状态时输入硬件触发，则对于模拟输入通道选择寄存器(ADS)指定的模拟输入执行 A/D 转换。随着硬件触发输入，ADM0 寄存器的 ADCS 位自动被设置为 1。
- <3> A/D 转换结束时，转换结果存储于 A/D 转换结果寄存器(ADCR、ADCRH)，并产生 A/D 转换结束中断请求信号(INTAD)。
- <4> A/D 转换结束后，ADCS 位被自动清除至 0，A/D 转换器进入停止状态。
- <5> 在转换操作过程中输入硬件触发时，当前 A/D 转换中断，重新开始转换。转换至中途的数据被丢弃。
- <6> 转换操作过程中改写或覆盖 ADS 寄存器的值时，当前 A/D 转换中断，并对 ADS 寄存器重新指定的模拟输入执行 A/D 转换。转换至中途的数据被丢弃。
- <7> 转换操作过程中用 1 覆盖 ADCS 时，当前 A/D 转换中断，转换重新开始。转换至中途的数据被初始化。
- <8> 转换操作过程中 ADCS 被清除至 0 时，当前 A/D 转换中断，系统进入硬件触发待机状态，而且 A/D 转换器进入停止状态。当 ADCE= 0 时，输入硬件触发的操作被忽略，不开始 A/D 转换。

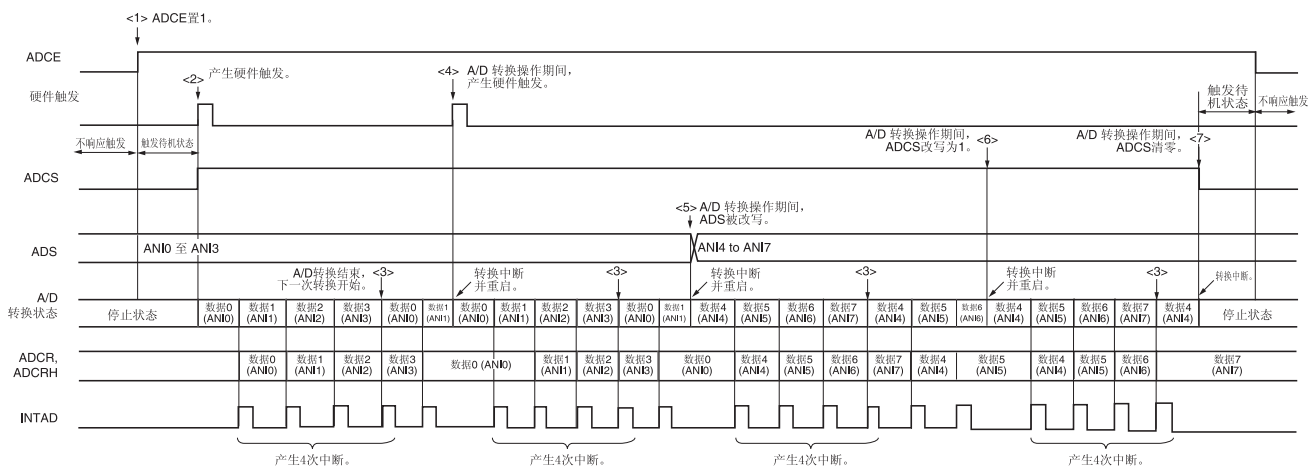
图 12-29. 硬件触发等待模式（选择模式，单次转换模式）操作时序的示例



12.6.11 硬件触发等待模式（扫描模式，连续转换模式）

- <1> 在停止状态下，A/D 转换器模式寄存器 0(ADM0)的 ADCE 位被设置为 1，系统进入 A/D 转换待机状态。
- <2> 如果在硬件触发待机状态下输入硬件触发，则由由模拟输入通道选择寄存器(ADS)指定的从扫描 0 至扫描 3 的 4 个模拟输入通道执行 A/D 转换。随着硬件触发输入，ADM0 寄存器的 ADCS 位自动被设置为 1。从扫描 0 指定的通道开始，依次对模拟输入通道执行 A/D 转换。
- <3> 依次对四个模拟输入通道执行 A/D 转换，每次转换结束时，将转换结果存储于 A/D 转换结果寄存器(ADCR、ADCRH)，并产生 A/D 转换结束中断请求信号(INTAD)。四个通道的 A/D 转换结束时，立即自动开始指定通道的下一次 A/D 转换。
- <4> 在转换操作过程中输入硬件触发时，当前 A/D 转换中断，从第一个通道重新开始转换。转换至中途的数据被丢弃。
- <5> 当在转换操作过程中改写或覆盖 ADS 寄存器的值时，当前 A/D 转换中断，并从 ADS 寄存器重新指定的第一个通道开始执行 A/D 转换。转换至中途的数据被丢弃。
- <6> 转换操作过程中用 1 覆盖 ADCS 时，当前 A/D 转换中断，转换重新开始。转换至中途的数据被丢弃。
- <7> 转换操作过程中 ADCS 被清除至 0 时，当前 A/D 转换中断，系统进入硬件触发待机状态，而且 A/D 转换器进入停止状态。当 ADCE= 0 时，输入硬件触发的操作被忽略，不开始 A/D 转换。

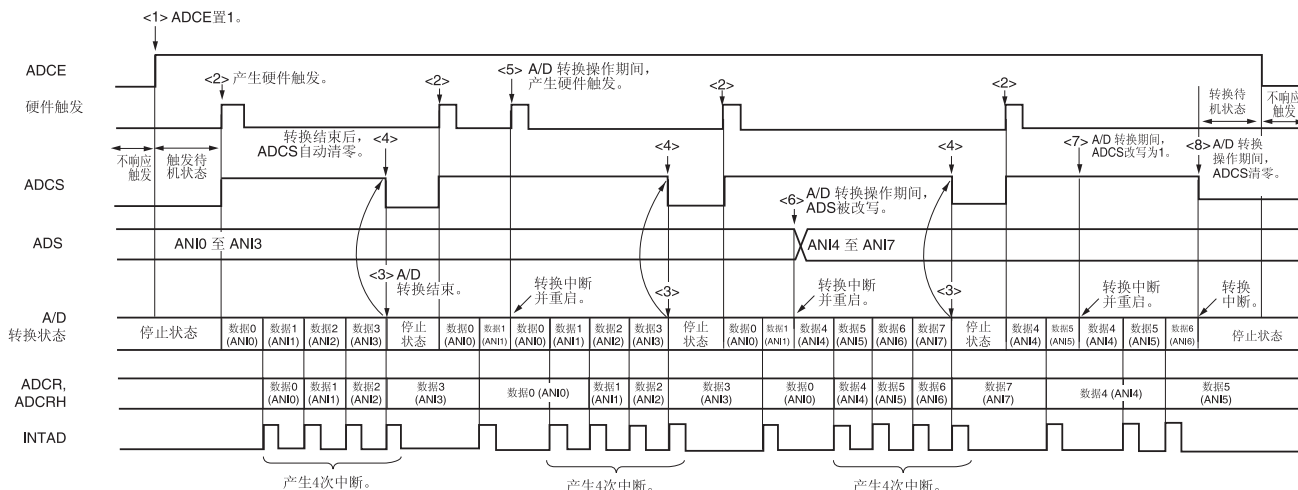
图 12-30. 硬件触发等待模式（扫描模式，连续转换模式）操作时序的示例



12.6.12 硬件触发等待模式（扫描模式，单次转换模式）

- <1> 在停止状态下，A/D 转换器模式寄存器 0(ADM0)的 ADCE 位被设置为 1，系统进入 A/D 转换待机状态。
- <2> 如果在硬件触发待机状态下输入硬件触发，则由模拟输入通道选择寄存器(ADS)指定的从扫描 0 至扫描 3 的 4 个模拟输入通道执行 A/D 转换。随着硬件触发输入，ADM0 寄存器的 ADCS 位自动被设置为 1。从扫描 0 指定的通道开始，依次对模拟输入通道执行 A/D 转换。
- <3> 依次对四个模拟输入通道执行 A/D 转换，每次转换结束时，将转换结果存储于 A/D 转换结果寄存器(ADCR、ADCRH)，并产生 A/D 转换结束中断请求信号(INTAD)。
- <4> A/D 转换结束后，ADCS 位被自动清除至 0，A/D 转换器进入停止状态。
- <5> 在转换操作过程中输入硬件触发时，当前 A/D 转换中断，从第一个通道重新开始转换。转换至中途的数据被丢弃。
- <6> 当在转换操作过程中改写或覆盖 ADS 寄存器的值时，当前 A/D 转换中断，并从 ADS 寄存器重新指定的第一个通道开始执行 A/D 转换。转换至中途的数据被丢弃。
- <7> 转换操作过程中用 1 覆盖 ADCS 时，当前 A/D 转换中断，转换重新开始。转换至中途的数据被丢弃。
- <8> 转换操作过程中 ADCS 被清除至 0 时，当前 A/D 转换中断，系统进入硬件触发待机状态，而且 A/D 转换器进入停止状态。当 ADCE= 0 时，输入硬件触发的操作被忽略，不开始 A/D 转换。

图 12-31. 硬件触发等待模式（扫描模式，单次转换模式）操作时序的示例

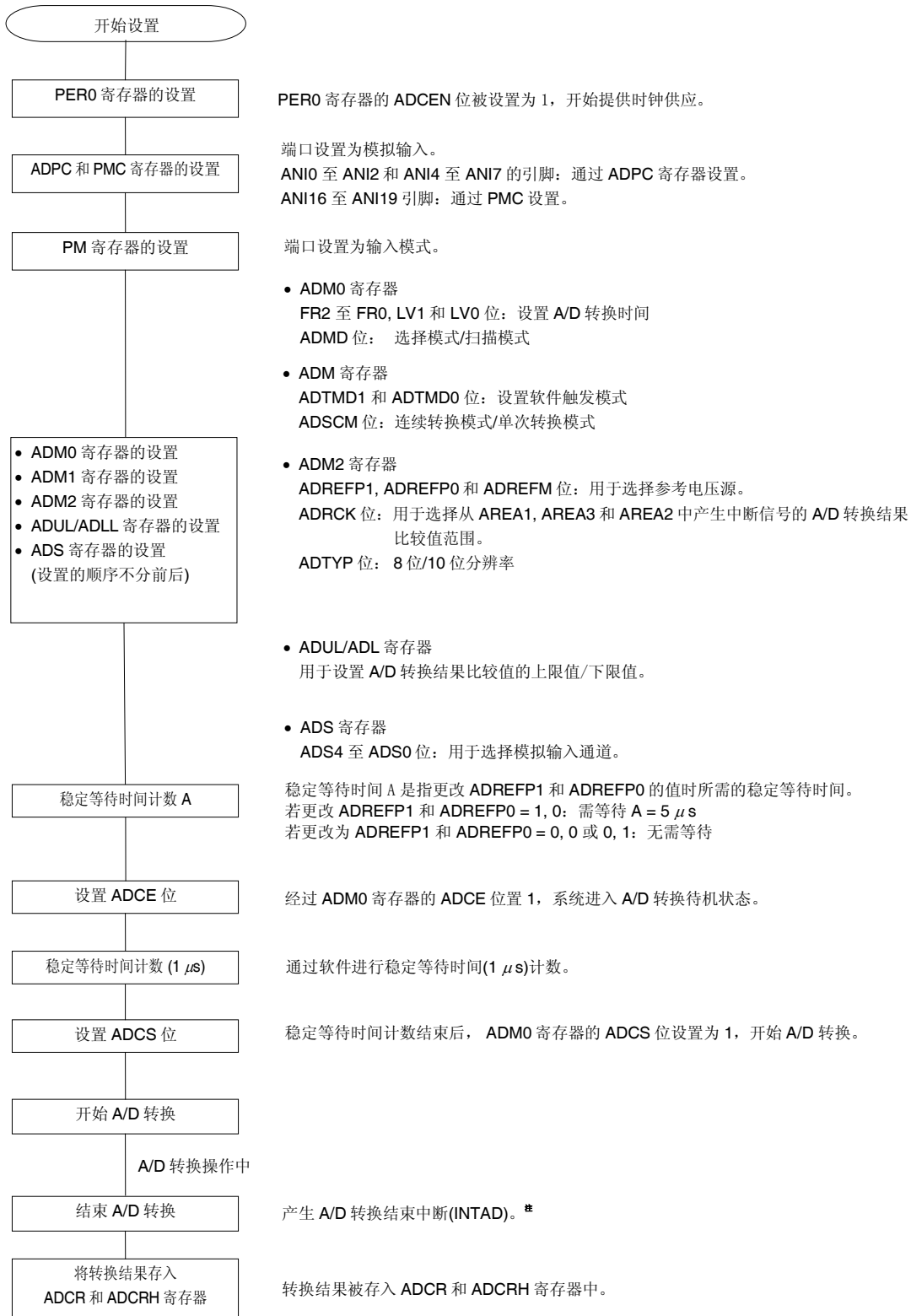


12.7 A/D转换器的设置流程图

各操作模式时的 A/D 转换器的设置流程图如下所示。

12.7.1 设置软件触发模式

图 12-32. 设置软件触发模式

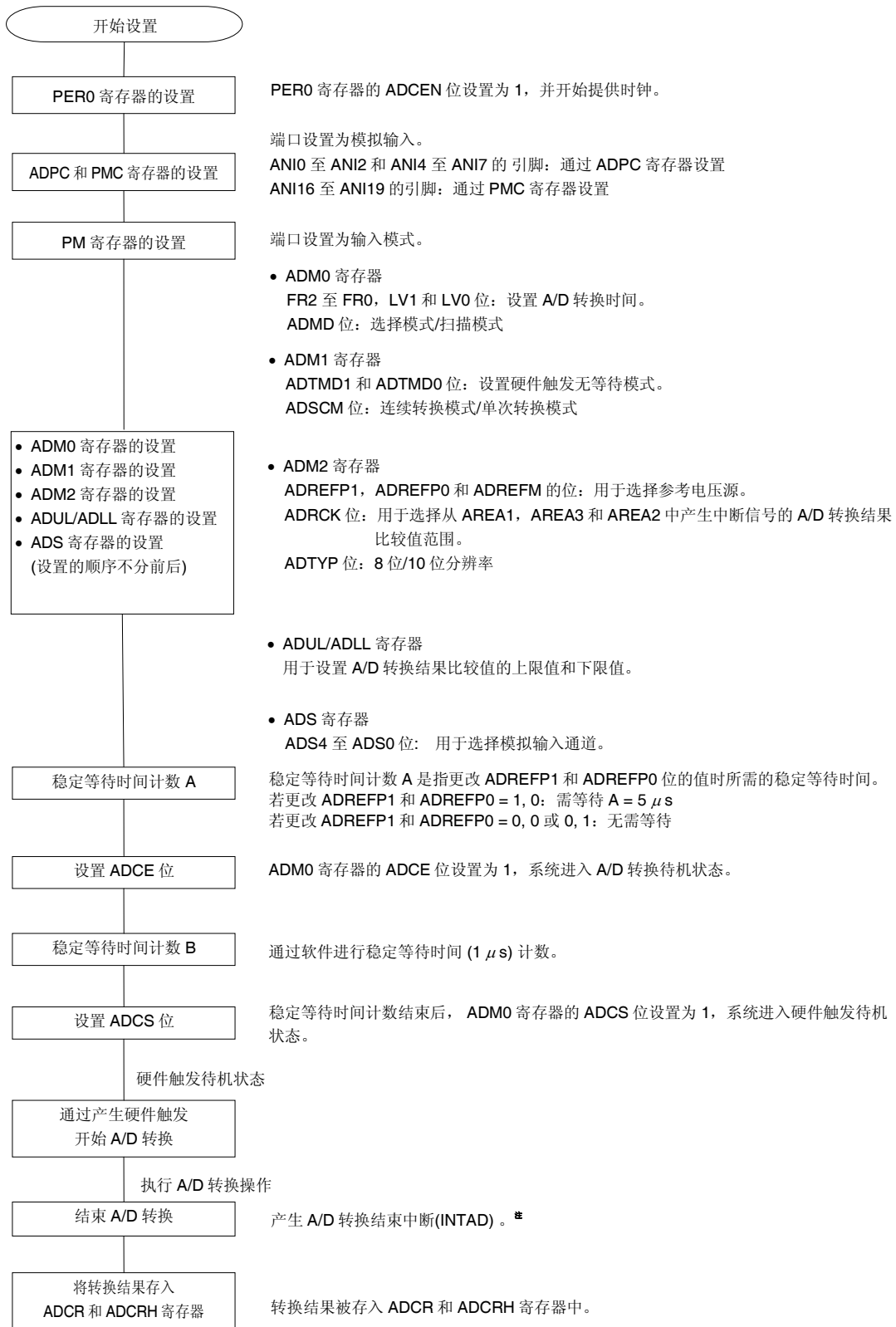


注 根据 ADRCK 位和 ADUL/ADLL 寄存器的设置，有可能不产生中断信号。这种情况下，结果不存储于 ADCR 寄存器。

注意事项 当 PGAOUT 被选择作为模拟输入时，设置 PGA 操作后再设置 ADS 寄存器 (参阅第十三章 可编程增益放大器)。

12.7.2 设置硬件触发无等待模式

图 12-33. 设置硬件触发无等待模式

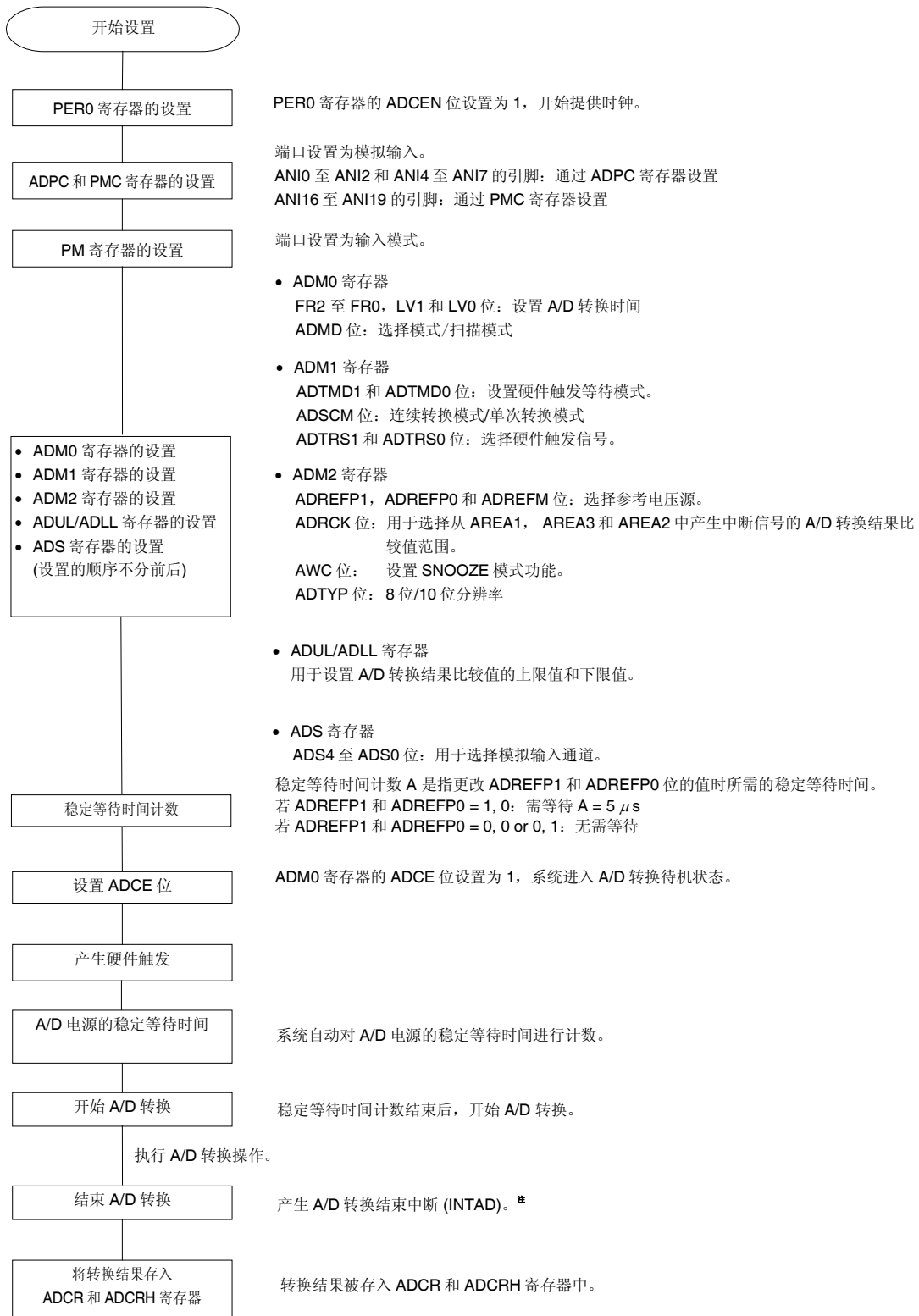


注 根据 ADRCK 位和 ADUL/ADLL 寄存器的设置，有可能不产生中断信号。这种情况下，结果不存储于 ADCR 寄存器。

注意事项 当 PGAOUT 被选择作为模拟输入时，设置 PGA 操作后再设置 ADS 寄存器 (参阅第十三章 可编程增益放大器)。

12.7.3 设置硬件触发等待模式

图 12-34. 设置硬件触发等待模式



注 根据 ADRCK 位和 ADUL/ADLL 寄存器的设置，有可能不产生中断信号。这种情况下，结果不存储于 ADCR 寄存器。

注意事项 当 PGAOUT 被选择作为模拟输入时，设置 PGA 操作后再设置 ADS 寄存器 (参阅第十三章 可编程增益放大器)。

12.7.4 使用温度传感器时的设置 (软件触发模式和单次转换模式下的示例)

图 12-35. 使用温度传感器时的设置

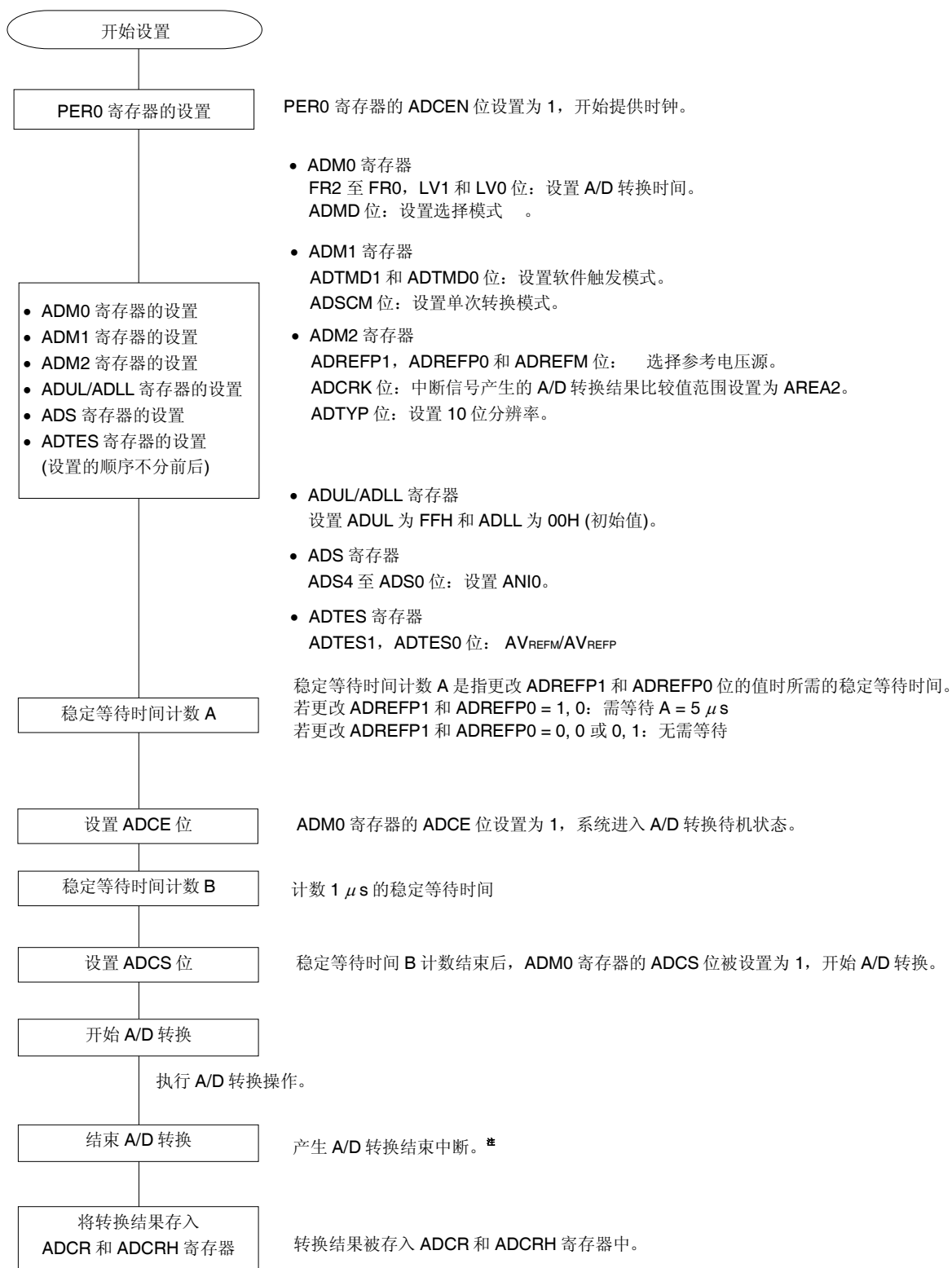


注 根据 ADRCK 位和 ADUL/ADLL 寄存器的设置，有可能不产生中断信号。这种情况下，结果不存储于 ADCR 寄存器。

注意事项 此设置仅能在 HS (高速主) 模式下使用。

12.7.5 设置测试模式

图 12-36. 设置测试模式



注 根据 ADCRK 位和 ADUL/ADLL 寄存器的设置，有可能不产生中断信号。这种情况下，结果不存储于 ADCR 寄存器。

12.8 SNOOZE模式功能

SNOOZE 模式时，通过在 STOP 模式下输入硬件触发来触发 A/D 转换。正常情况下，在 STOP 模式时停止 A/D 转换，但是，在 SNOOZE 模式时，可以在不操作 CPU 的情况下通过输入硬件触发执行 A/D 转换。这样可以有效减少工作电流。

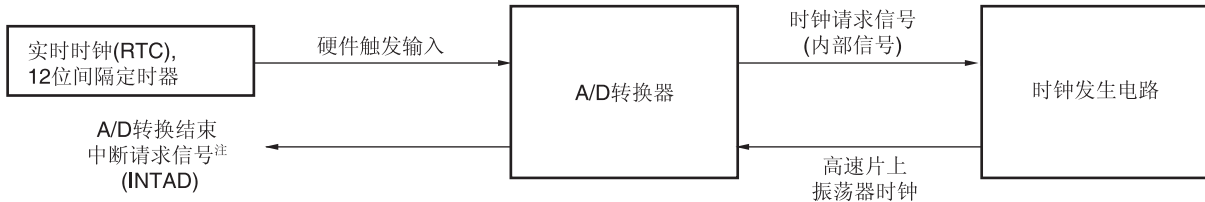
如果在 SNOOZE 模式中使用 ADUL 和 ADLL 寄存器指定 A/D 转换结果范围，每过一段时间就可判断出 A/D 的转换结果。使用此功能，可启动电源电压监视和根据 A/D 输入进行输入按键判断。

SNOOZE 模式时，仅能使用以下两种转换模式：

- 硬件触发等待模式 (选择模式，单次转换模式)
- 硬件触发等待模式 (扫描模式，单次转换模式)

注意事项 仅在选用高速片上振荡器时钟作为 f_{CLK} 时可以设置 SNOOZE 模式。

图 12-37. 使用 SNOOZE 模式功能时的框图



使用 SNOOZE 模式功能时，在切换至 STOP 模式之前进行各寄存器的初始设置。(有关这些设置的详情，请参阅 12.7.3 设置硬件触发等待模式^{注2}。)在切换至 STOP 模式之前，设置 A/D 转换器模式寄存器 2(ADM2)的位 2(AWC)为 1。初始设置完成后，设置 A/D 转换器模式寄存器 0(ADM0)的位 0(ADCE)为 1。

如果在切换至 STOP 模式之后输入硬件触发，则向 A/D 转换器供应高速片上振荡器时钟。供应该时钟之后，系统将自动递增计数至稳定等待时间，然后开始 A/D 转换。

A/D 转换结束后的 SNOOZE 模式操作因是否产生中断信号而有所不同^{注1}。

注 1. 根据 A/D 转换结果比较功能的设置 (ADRCCK 位、ADUL/ADLL 寄存器)，有可能不产生中断请求信号。

2. 必须将 ADM1 寄存器设置为 E2H。

备注 硬件触发包括 INTTM01、INTRTC、INTIT 以及定时器 KB0 至 KB2 的 A/D 触发信号。

通过 A/D 转换器模式寄存器 1(ADM1)设置硬件触发。

(1) A/D 转换结束后产生中断时

如果 A/D 转换结果值处于 A/D 转换结果比较功能设置的值的范围之内（通过 ADRCK 位和 ADUL/ADLL 寄存器设置），则产生 A/D 转换结束中断请求信号(INTAD)。

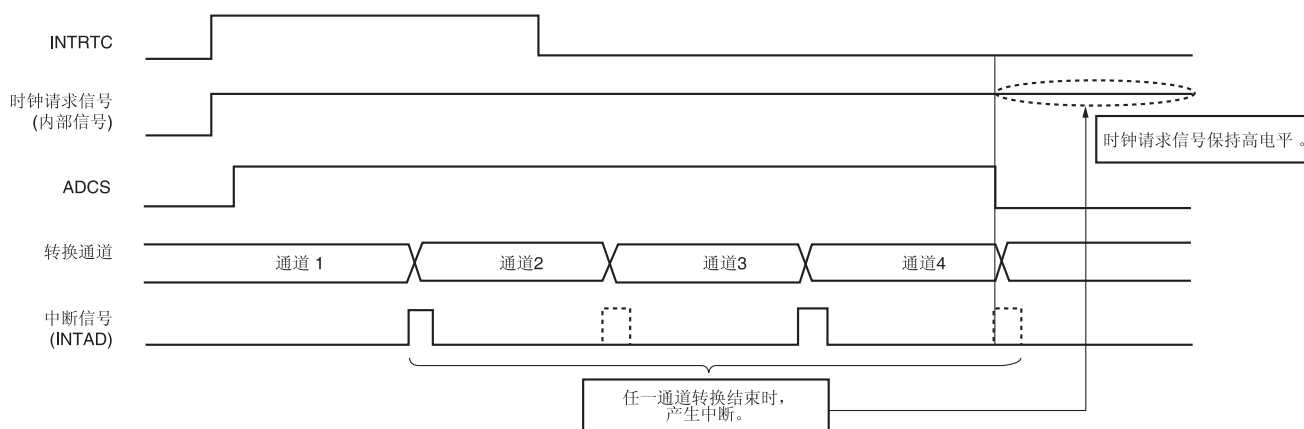
- 当处于选择模式时

在 A/D 转换结束且产生 A/D 转换结束中断请求信号(INTAD)之后，A/D 转换器从 SNOOZE 模式切换至正常操作模式。此时，必须将 A/D 转换器模式寄存器 2(ADM2)的位 2(AWC = 0: SNOOZE 模式解除)清除。如果 AWC 位保持 1，不管在之后的 SNOOZE 模式下或者在正常操作模式下 A/D 转换都不会正常启动。

- 当处于扫描模式时

在四个通道的 A/D 转换过程中即使产生一个 A/D 转换结束中断请求信号(INTAD)，时钟请求信号也保持高电平，A/D 转换器将从 SNOOZE 模式切换至正常操作模式。此时，必须将 A/D 转换器模式寄存器 2(ADM2)的位 2(AWC = 0: SNOOZE 模式解除)清除至 0。如果 AWC 位保持 1，不管在之后的 SNOOZE 模式下或者在正常操作模式下 A/D 转换都不会正常启动。

图 12-38. A/D 转换结束后产生中断时的操作示例(扫描模式下)



(2) 如果在 A/D 转换结束后不产生中断

如果 A/D 转换结果值处于 A/D 转换结果比较功能设置的值的范围之外（通过 ADRCK 位和 ADUL/ADLL 寄存器设置），则不产生 A/D 转换结束中断请求信号(INTAD)。

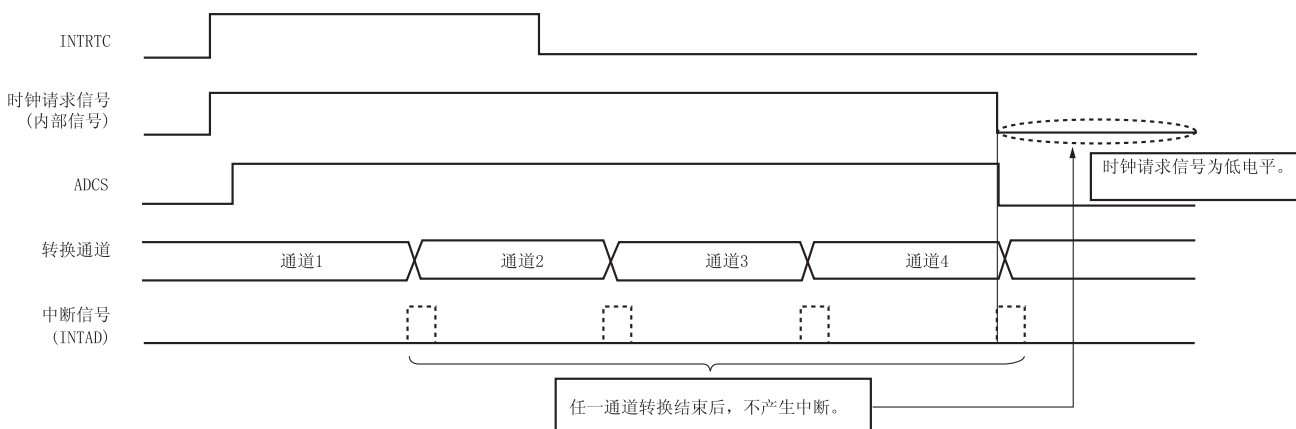
- 当处于选择模式时

如果 A/D 转换结束后未产生 A/D 转换结束中断请求信号(INTAD)，时钟请求信号（内部信号）将被自动设置为低电平，并停止供应高速片上振荡器时钟。之后，如果输入硬件触发，则会再次在 SNOOZE 模式下执行 A/D 转换。

- 当处于扫描模式时

如果未在四个通道的 A/D 转换过程中产生 A/D 转换结束中断请求信号(INTAD)，则时钟请求信号（内部信号）将在四个通道的 A/D 转换结束后被自动设置为低电平，并停止供应高速片上振荡器时钟。之后，如果输入硬件触发，则会再次在 SNOOZE 模式下执行 A/D 转换。

图 12-39. A/D 转换结束后不产生中断时的操作示例(扫描模式下)



12.9 如何阅读A/D转换器特性表

以下将介绍 A/D 转换器的专用术语。

(1) 分辨率

可以分辨的最低模拟输入电压。即是说，每位数字输出的模拟输入电压的百分比被称为 1LSB（最低有效位）。1LSB 对满刻度的比率被表示为%FSR（满刻度范围）。

当分辨率为 10 位时，1LSB 表示如下：

$$\begin{aligned} 1\text{LSB} &= 1/2^{10} = 1/1024 \\ &= 0.098\%\text{FSR} \end{aligned}$$

精度与分辨率无关，而是取决于总误差。

(2) 总误差

总误差是实际测量值和理论值之间的最大误差值。

将零刻度误差、满刻度误差、积分线性误差和微分线性误差以及这些组合所产生的误差综合起来，即为总误差。

请注意，总误差特性表中不包括量化误差。

(3) 量化误差

当模拟值被转换为数字值时，必然会出现转 $\pm 1/2\text{LSB}$ 的误差。在 A/D 转换器中， $\pm 1/2\text{LSB}$ 范围内的模拟输入电压被转换成同样的数字代码，因此量化误差不可避免。

请注意，量化误差不包括在特性表中的总误差、零刻度误差、满刻度误差、积分线性误差和微分线性误差之中。

图 12-40. 总误差

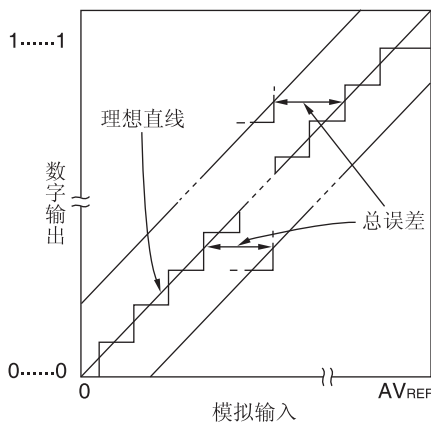
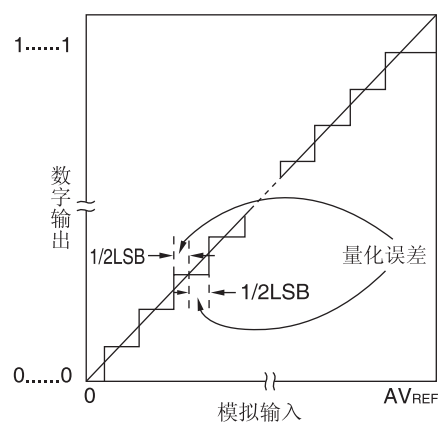


图 12-41. 量化误差



(4) 零刻度误差

零刻度误差是指当数字输出从 0.....000 变为 0.....001 时，模拟输入电压的实际测量值和理论值 ($1/2\text{LSB}$) 之间的差值。

如果实际测量值大于理论值，则表示当数字输出从 0.....001 变为 0.....010 时，模拟输入电压的实际测量值和理论值 ($3/2\text{LSB}$) 之间的差值。

(5) 满刻度误差

满刻度误差是当数字输出从 1.....110 变为 1.....111 时，模拟输入电压的实际测量值和理论值（满刻度 - 3/2LSB）之间的差值。

(6) 积分线性误差

积分线性误差说明了转换特性偏离理想线性关系的程度。它表示当零刻度误差和满刻度误差均为 0 时，实际测量值和理想直线之间的最大误差。

(7) 微分线性误差

微分线性误差是指，输出代码的理想宽度为 1LSB 时，某个代码输出宽度的实际测量值与理想值之间的差值。

图 12-42. 零刻度误差

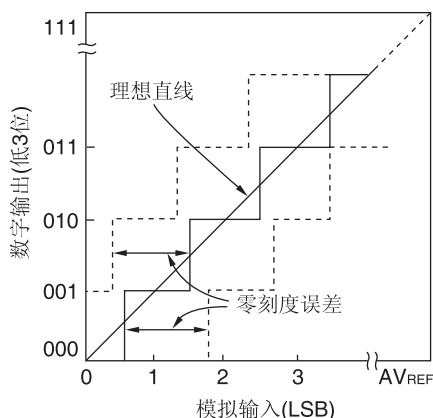


图 12-43. 满刻度误差

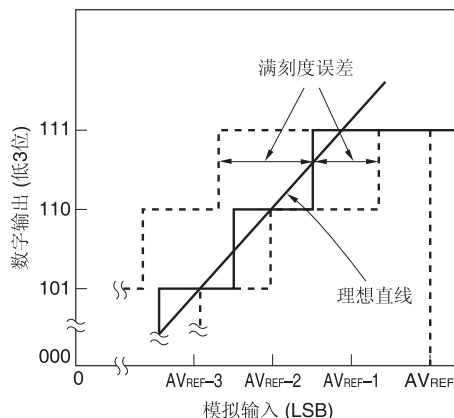


图 12-44. 积分线性误差

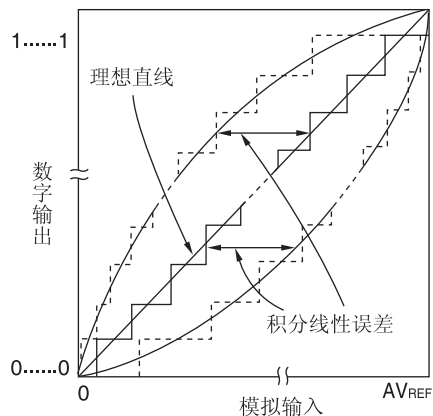
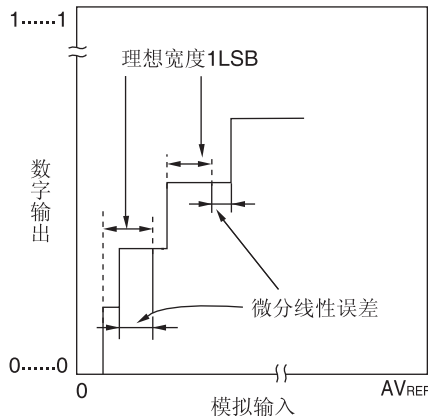


图 12-45. 微分线性误差

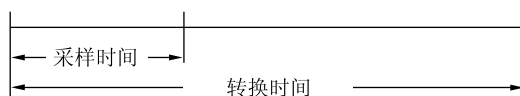


(8) 转换时间

转换时间是指从开始采样到获得数字输出的时间。特性表中的转换时间包括采样时间。

(9) 采样时间

采样时间是指为了将模拟电压载入采样&保持电路而打开模拟开关的时间。



12.10 A/D转换器的注意事项

(1) STOP 模式下的操作电流

停止 A/D 转换器（将 A/D 转换器模式寄存器 0(ADM0)的位 7(ADCS)设置为 0）之后切换至 STOP 模式。同时，可通过将 ADM0 寄存器的位 0(ADCE)设置为 0 来减少操作电流。

从待机状态重启时，将中断请求标志寄存器 1H(IF1H)的位 0(ADIF)清除为 0 之后再开始操作。

(2) ANI0 至 ANI2, ANI4 至 ANI7, ANI16 至 ANI19 和 PGAOUT 引脚的输入范围

请遵守 ANI0 至 ANI2、ANI4 至 ANI7、ANI16 至 ANI19 和 PGAOUT 引脚输入电压的额定范围。如果向模拟输入通道输入大于或等于 V_{DD} 和 AV_{REFP} 或者小于或等于 V_{SS} 和 AV_{REFM} 的电压（即使在绝对最大额定范围内），则该通道的转换值将不确定。另外，也可能影响到其他通道的转换值。

注意事项 仅在 HS (高速主)模式中可使用内部基准电压 (1.45 V)。

(3) 冲突操作

<1> 转换结束时，对 A/D 转换结果寄存器(ADCR、ADCRH)的写入与通过指令对于 ADCR 或 ADCRH 寄存器的读取之间的冲突

对 ADCR 或 ADCRH 寄存器的读取优先。读取操作之后，新转换结果被写入 ADCR 或 ADCRH 寄存器。

<2> 转换结束时，对于 ADCR 或 ADCRH 寄存器的写入、对于 A/D 转换器模式寄存器 0(ADM0)的写入、对模拟输入通道选择寄存器(ADS)或 A/D 端口配置寄存器(ADPC)的写入之间的冲突

对 ADM0、ADS 或 ADPC 寄存器的写入优先。不执行对 ADCR 或 ADCRH 寄存器的写入，也不产生转换结束中断信号(INTAD)。

(4) 噪声对策

为了保持 10 位分辨率，必须注意输入至 AV_{REFP} 、 V_{DD} 、ANI0 至 ANI2、ANI4 至 ANI7 和 ANI16 至 ANI19 引脚的噪声。

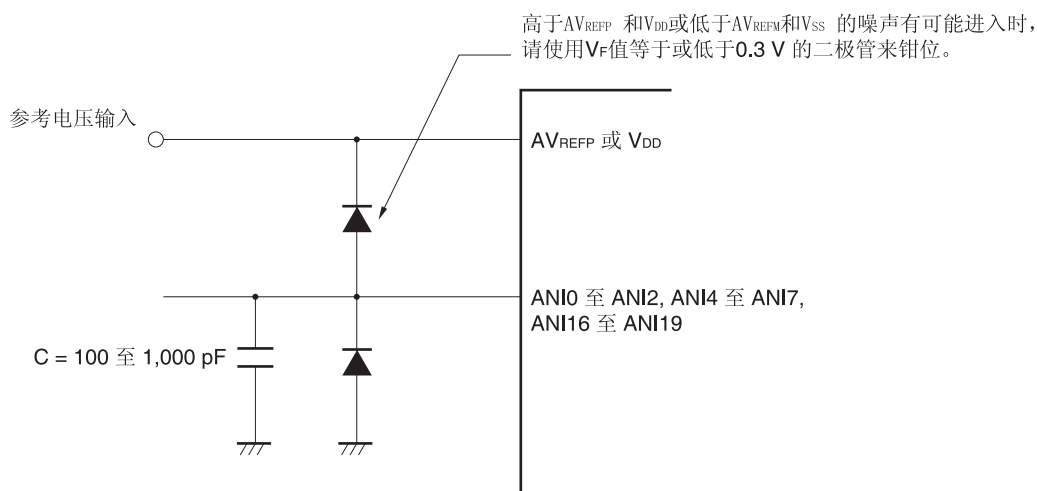
<1> 连接到电源的电容器请使用具备等效电阻低且频率响应良好的电容器。

<2> 模拟输入源的输出阻抗越高，影响越大。如图 12-46 所示，建议外接 C 以减少噪声。

<3> 转换过程中，不要切换引脚。

<4> 转换开始后立即设置为 HALT 模式可以提高精度。

图 12-46. 模拟输入引脚的连接



(5) 模拟输入 (ANIn) 引脚

<1> 模拟输入引脚 (AN10 至 AN12、AN14 至 AN17、AN16 至 AN19) 也用作输入端口引脚 (P20 至 P22、P24 至 P27、P03、P02、P147、P120)。

选择 AN10 至 AN12、AN14 至 AN17 和 AN16 至 AN19 引脚中的任意一个引脚并对其执行 A/D 转换时，不要在转换过程中更改引脚 P20 至 P22、P24 至 P27、P03、P02、P147 和 P120 的输出值。否则，可能导致转换分辨率下降。

<2> 如果将正在执行 A/D 转换的引脚相邻的引脚用作数字输入/输出端口的话，由于耦合噪声的影响，可能会取得预期不同的 A/D 转换值。因此，请不要输入或输出这样的脉冲。

(6) 模拟输入(ANIn)引脚的输入阻抗

A/D 转换器在采样时间对内部的采样电容充电，并进行采样。

因此，不进行采样时仅通过漏电流，采样时还通过对电容充电的电流。所以输入阻抗将依据是否进行采样而产生波动。然而，为了保证采样有效，建议将模拟输入源的输出阻抗保持在 $1k\Omega$ 以内，并且在高输出阻抗时将大约 $100pF$ 的电容连接至 AN10 至 AN12、AN14 至 AN17 以及 AN16 至 AN19 引脚(参阅图 12-46)。

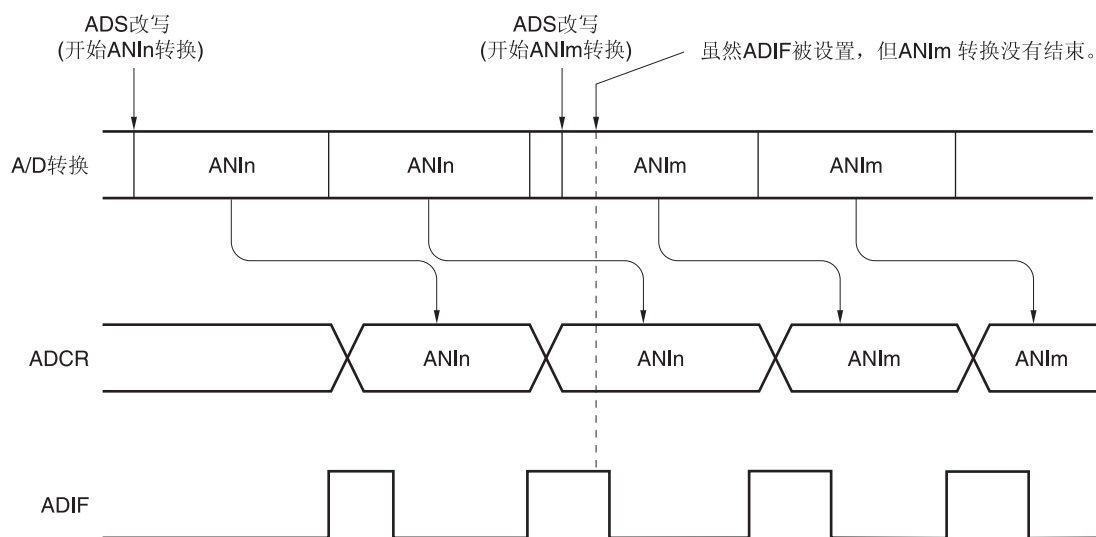
(7) 中断请求标志(ADIF)

即使更改模拟输入通道选择寄存器(ADS)，中断请求标志(ADIF)也不会被清除。

因此，A/D 转换过程中更改了模拟输入引脚时，改写 ADS 寄存器之前，变更前的模拟输入的 A/D 转换结果和 ADIF 标志有被设置的可能性。因此，改写 ADS 寄存器之后立即读取 ADIF 标志时，要注意即使变更后的模拟输入的 A/D 转换尚未结束 ADIF 标志也会被置位。

另外，暂停 A/D 转换后又重新开始时，须在重新开始 A/D 转换之前清除 ADIF 标志。

图 12-47. 产生 A/D 转换结束中断请求的时序

**(8) A/D 转换开始后的初次转换结果**

当在软件触发模式或硬件触发无等待模式时，如果在 ADCE 位设置为 1 之后的 $1\ \mu\text{s}$ 内将 ADCS 位设置为 1，则开始 A/D 转换操作后初次产生的 A/D 转换值可能不在额定范围内。须采取例如轮询 A/D 转换结束中断请求(INTAD)以舍弃初次的转换结果等措施。

(9) A/D 转换结果寄存器(ADCR、ADCRH)的读取操作

对 A/D 转换器模式寄存器 0(ADM0)、模拟输入通道选择寄存器(ADS)、A/D 端口配置寄存器(ADPC)以及端口模式控制寄存器(PMC) 执行写操作时，ADCR 和 ADCRH 寄存器的内容可能不确定。转换操作结束后，对 ADM0、ADS、ADPC 或 PMC 寄存器执行写操作之前读取转换结果。如果不在上述时序执行，则可能读取错误的转换结果。

(10) 内部等效电路

模拟输入模块的等效电路如下所示。

图 12-48. ANIn 引脚的内部等效电路

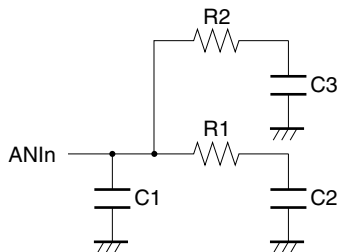


表 12-6. 等效电路的电阻和电容值 (参考值)

AV_{REFP}, V_{DD}	ANIn 引脚	R1 [k Ω]	C1 [pF]	C2 [pF]
$3.6\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	ANI0至ANI2, ANI4至ANI7	14	8	2.5
	ANI16至ANI19	18	8	7.0
$2.7\text{ V} \leq V_{DD} < 3.6\text{ V}$	ANI0至ANI2, ANI4至ANI7	39	8	2.5
	ANI16至ANI19	53	8	7.0

AV_{REFP}, V_{DD}	R2/C3	最小值	典型值	最大值
$2.7\text{ V} \leq V_{DD} < 3.6\text{ V}$	R2 [k]	2		3200
	C3 [pF]			2

备注 表 12-6 所示电阻和电容值均非保证值。

(11) 开始 A/D 转换器的操作

在 AV_{REFP} 和 V_{DD} 的电压稳定后，开始进行 A/D 转换器的操作。

第十三章 可编程增益放大器

RL78/I1A 中内置可编程增益放大器的电路。

与可编程增益放大器对应的模拟输入通道数因产品而异。

	20引脚	32引脚	30引脚, 38引脚
模拟输入通道	4通道 (ANI2/CMP0P, ANI4/CMP1P ANI5/CMP2P, ANI18/(CMP3P)/(CMPCOM))	5通道 (ANI2/CMP0P, ANI4/CMP1P至 ANI7/CMP4P)	6通道 (ANI2/CMP0P, ANI4/CMP1P至 ANI7/CMP4P, ANI16/CMP5P)

注意事项 本章的下述内容大多是以 38 引脚产品为例。

13.1 可编程增益放大器的功能

可编程增益放大器具备以下的功能。

- 可编程增益放大器的输入可从 CMP0P 至 CMP5P 六个引脚中选择。
- 有四种增益率可供选择。
- 可编程增益放大器的输出信号可设置为 A/D 转换器的模拟输入。

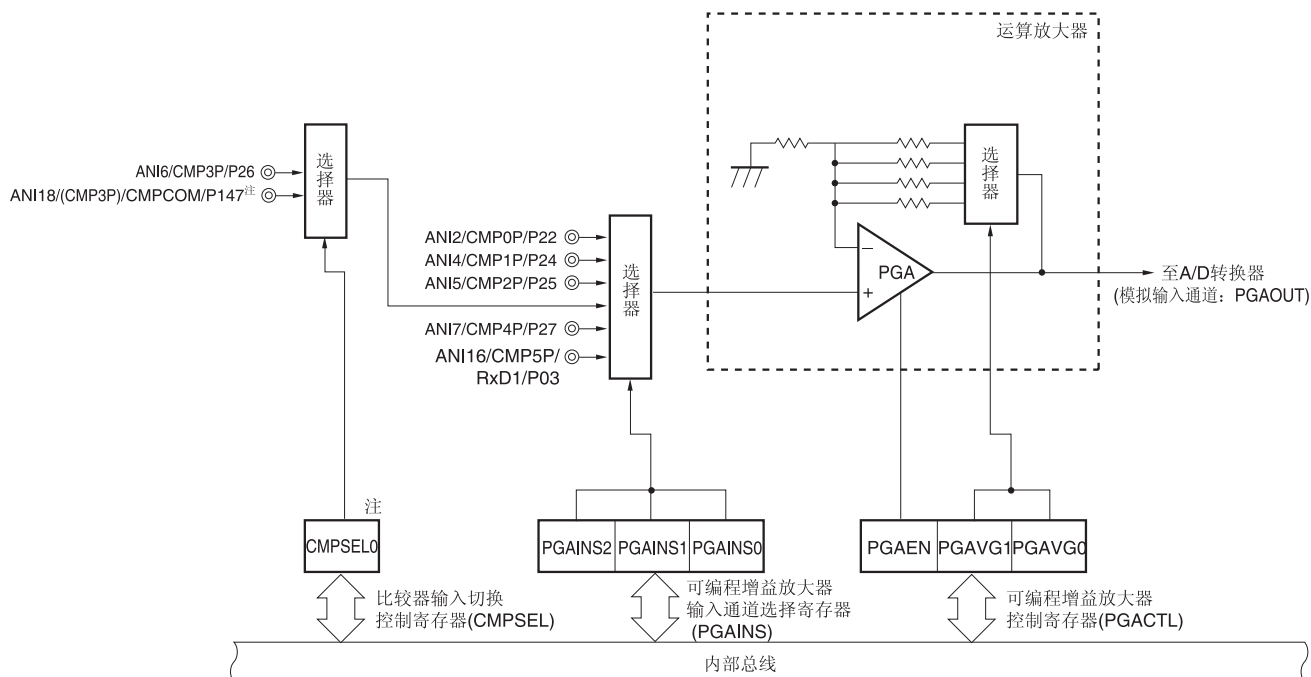
13.2 可编程增益放大器的配置

可编程增益放大器由以下硬件构成。

表 13-1. 可编程增益放大器的配置

项目	配置
可编程增益放大器本体	模拟输入通道 最大 6 通道
可编程增益放大器的输入	CMP0P 至 CMP5P 引脚
控制寄存器	外围允许寄存器 2 (PER2) 可编程增益放大器控制寄存器(PGACTL) 可编程增益放大器输入通道选择寄存器(PGAINS) 比较器输入切换控制寄存器(CMPSEL) A/D 端口配置寄存器(ADPC) 端口模式控制寄存器 0, 14 (PMC0, PMC14) 端口模式寄存器 0, 2, 14 (PM0, PM2, PM14) 比较器内部基准电压控制寄存器(CVRCTL)

图 13-1. 可编程增益放大器的框图



注 仅限 20 引脚产品。30 引脚、32 引脚和 38 引脚产品的默认选择为 ANI6/CMP3P/P26。

13.3 可编程增益放大器中使用的寄存器

可编程增益放大器使用以下八种寄存器。

- 外围允许寄存器 2 (PER2)
- 可编程增益放大器控制寄存器 (PGACTL)
- 可编程增益放大器输入通道选择寄存器 (PGAINS)
- 比较器输入切换控制寄存器 (CMPSEL) (仅限 20 引脚产品)
- A/D 端口配置寄存器 (ADPC)
- 端口模式控制寄存器 0, 14 (PMC0, PMC14)
- 端口模式寄存器 0, 2, 14 (PM0, PM2, PM14)
- 比较器内部基准电压控制寄存器 (CVRCTL)

(1) 外围允许寄存器 2 (PER2)

该寄存器用于允许或禁止供应时钟至外围硬件。并且中止向未使用的硬件供应时钟，以减少电力消耗和噪声。

当使用可编程增益放大器时，务必将该寄存器的位 7 (PGACMPEN) 设置为 1。

使用 1 位或 8 位存储器操作指令设置 PER2 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 13-2. 外围允许寄存器 2 (PER2) 的格式

地址: F0509H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
PER2	PGACMPEN	TKBPA2EN	TKBPA1EN	TKBPA0EN	TKC0EN	TKB2EN ^注	TKB1EN	TKB0EN

PGACMPEN	比较器/可编程增益放大器输入时钟控制
0	停止输入时钟的供应。 <ul style="list-style-type: none"> 不可写入用于比较器或可编程增益放大器的SFR。 比较器或可编程增益放大器处于复位状态。
1	供应输入时钟。 <ul style="list-style-type: none"> 可以读取和写入用于比较器或可编程增益放大器的SFR。

注 仅限 20 引脚产品、32 引脚产品和 38 引脚产品。

(2) 可编程增益放大器控制寄存器(PGACTL)

PGACTL 控制可编程增益放大器的操作。

使用 1 位或 8 位存储器操作指令设置 PGACTL 寄存器。

产生复位信号后，该寄存器被清除为 00H。

表 13-3. 可编程增益放大器控制寄存器(PGACTL)的配置

地址: F0550H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
PGACTL	PGAEN	0	0	0	0	0	PGAVG1	PGAVG0

PGAEN	可编程增益放大器操作的控制
0	停止可编程增益放大器的操作。
1	允许可编程增益放大器的操作。

PGAVG1	PGAVG0	可编程增益放大器增益率的选择
0	0	×4
0	1	×8
1	0	×16
1	1	×32

- 注意事项**
1. 当使用可编程增益放大器，使用 ADPC 寄存器来选择 CMP0P/ANI2/P22、CMP1P/ANI4/P24、CMP2P/ANI5/P25、CMP3P/ANI6/P26、CMP4P/ANI7/P27 引脚和 PGAOUT 引脚(内部引脚)作为模拟输入。使用 PMC0 和 PMC14 寄存器来选择 CMP5P/ANI16/RxD1/P03 和 CMPCOM/ANI18/P147 引脚作为模拟输入。
 2. 使用可编程增益放大器时，将不用于可编程增益放大器的端口 0 和 2 用作数字输入时，务必确认数字输入端口的输入电平被固定，以防止 A/D 转换精度降低。
 3. 请设置了增益率后，再允许可编程增益放大器的操作(PGAEN = 1)。操作允许状态(PGAEN = 1)时禁止更改增益率的设置。
 4. 对于可编程增益放大器，在设置 PGAEN = 1 后需要有操作稳定等待时间(T.B.D.)。

(3) 可编程增益放大器输入通道选择寄存器(PGAINS)

该寄存器选择可编程增益放大器的输入通道。

使用 8 位存储器操作指令来设置 PGAINS 寄存器。

产生复位信号后，该寄存器被清除为 00H。

表 13-4. 可编程增益放大器输入通道选择寄存器(PGAINS)的格式

地址: F0551H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
PGAINS	0	0	0	0	0	PGAINS2	PGAINS1	PGAINS0

PGAINS2	PGAINS1	PGAINS0	输入到可编程增益放大器的模拟输入通道
0	0	0	ANI2/CMP0P
0	0	1	ANI4/CMP1P
0	1	0	ANI5/CMP2P
0	1	1	ANI6/CMP3P或ANI18/(CMP3P) ^注
1	0	0	ANI7/CMP4P
1	0	1	ANI16/CMP5P
其它			禁止设置

注 通过比较器输入切换控制寄存器(CMPSEL)来进行选择(仅限 20 引脚产品)

注意事项 请在可编程增益放大器(PGAEN = 0)停止操作时设置 PGAINS 寄存器。

(4) 比较器输入切换控制寄存器(CMPSEL) (仅限 20 引脚产品)

这个寄存器用来选择比较器 3 的输入信号。

使用 8 位存储器操作指令来设置 CMPSEL 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 13-5. 比较器输入切换控制寄存器(CMPSEL)的格式

地址: F0565H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
CMPSEL	0	0	0	0	0	0	0	CMPSEL0

CMPSEL0	比较器3的+ 侧输入控制
0	20引脚产品中不使用比较器3。
1	(CMP3P)/CMPCOM/ANI18/P147引脚

- 注意事项 1. 比较器停止操作(C3CTL 寄存器中 CMP3EN = 0)时重写 CMPSEL 寄存器。
2. 当可编程增益放大器用 ANI18 输入进行操作时，禁止重写 CMPSEL 寄存器。

(5) A/D 端口配置寄存器(ADPC)

该寄存器将 ANI0/P20、ANI1/P21、ANI2/CMP0P/P22、ANI4/CMP1P/P24 至 ANI7/CMP4P/P27 引脚和 PGAOUT 引脚 (内部引脚)在端口的数字输入/输出以及 A/D 转换器、可编程增益放大器或比较器的模拟输入之间进行切换。

当使用可编程增益放大器或比较器时，使用 ADPC 寄存器来选择 ANI2/CMP0P/P22、ANI4/CMP1P/P24 至 ANI7/CMP4P/P27 引脚和 PGAOUT 引脚(内部引脚)作为模拟输入。

使用 8 位存储器操作指令来设置 ADPC 寄存器。

产生复位信号后，设置该寄存器为 00H。

图 13-6. A/D 端口配置寄存器(ADPC)的格式

地址: F0076H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ADPC	0	0	0	0	ADPC3	ADPC2	ADPC1	ADPC0

ADPC3	ADPC2	ADPC1	ADPC0	模拟输入(A)/数字 I/O(D)切换							
				ANI7/ CMP4P/P27	ANI6/ CMP3P/P26	ANI5/ CMP2P/P25	ANI4/ CMP1P/P24	PGAOUT ^注	ANI2/ CMP0P/P22	ANI1/P21/ AVREFM	ANI0/P20/ AVREFP
0	0	0	0	A	A	A	A	A	A	A	A
0	0	0	1	D	D	D	D	D	D	D	D
0	0	1	0	D	D	D	D	D	D	D	A
0	0	1	1	D	D	D	D	D	D	A	A
0	1	0	0	D	D	D	D	D	A	A	A
0	1	0	1	D	D	D	D	A	A	A	A
0	1	1	0	D	D	D	A	A	A	A	A
0	1	1	1	D	D	A	A	A	A	A	A
1	0	0	0	D	A	A	A	A	A	A	A
1	0	0	1	A	A	A	A	A	A	A	A
1	1	1	1	A	A	A	A	A	A	A	A
其它				禁止设置							

注 这是一个可编程增益放大器的内部输出引脚。将可编程增益放大器的输出信号用作 A/D 转换器的模拟输入通道时，设置 ADPC 等于 0000B 或大于等于 0101B。

- 注意事项
1. 通过端口模式寄存器 2 (PM2)将用于 A/D 转换的通道设置为输入模式。
 2. 对于由 ADPC 寄存器设置为数字输入/输出的引脚，不要用模拟输入通道指定寄存器(ADS)进行设置。

(6) 端口模式控制寄存器 0, 14 (PMC0, PMC14)

使用该寄存器按 1 位单位设置端口 0、14 的数字 I/O 或模拟输入。

当使用 ANI16/CMP5P/RxD1/P03 或 ANI18/CMPCOM/P147 引脚作为模拟输入端口时，将 PMC03 或 PMC147 位设置为 1。

用 1 位或 8 位存储器操作指令来设置 PMC0 和 PMC14 寄存器。

产生复位信号后，设置该寄存器为 FFH。

图 13-7. 端口模式控制寄存器 0、14 (PMC0, PMC14)的格式

地址: F0060H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PMC0	1	1	1	1	PMC03	PMC02	1	1

地址: F006EH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PMC14	PMC147	1	1	1	1	1	1	1

PMCmn	选择Pmn引脚数字输入/输出和模拟输入(mn = 02, 03, 147)
0	数字输入/输出(模拟输入以外的复用功能)
1	模拟输入

(7) 端口模式寄存器 0, 2, 14 (PM0, PM2, PM14)

当使用 ANI2/CMP0P/P22、ANI4/CMP1P/P24 至 ANI7/CMP4P/P27、ANI16/CMP5P/RxD1/P03 或 ANI18/CMPCOM/P147 引脚作为模拟输入端口时，将 PM22、PM24 至 PM27、PM03 或 PM147 位设置为 1。此时，P22、P24 至 P27、P03 和 P147 的输出锁存器可为 0 或 1。

如果 PM22、PM24 至 PM27、PM03 和 PM147 位被设置为 0，则不能用作模拟输入端口引脚。

用 1 位或 8 位存储器操作指令设置 PM0、PM2 和 PM14 寄存器。

产生复位信号后，设置该寄存器为 FFH。

注意事项 读取被设置为模拟输入端口的引脚时，即使 $PM_{xx} = 1$ ，读取值也不是引脚电平而是一直为“0”。

图 13-8. 端口模式寄存器 0, 2, 14 (PM0, PM2, PM14)的格式(38 引脚产品)

地址: FFF20H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM0	1	PM06	PM05	1	PM03	PM02	1	1

地址: FFF22H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM2	PM27	PM26	PM25	PM24	1	PM22	PM21	PM20

地址: FFF2EH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM14	PM147	1	1	1	1	1	1	1

PMmn	Pmn引脚输入/输出模式选择(m = 0, 2, 14; n = 0至7)
0	输出模式 (输出缓冲器on)
1	输入模式 (输出缓冲器off)

注意事项 务必将 PM0 寄存器的位 0, 1, 4, 7、PM2 寄存器的位 3、PM14 寄存器的位 0 至 6 以及 PM20 寄存器的位 7 设置为“1”。

对于 32, 30 和 20 引脚产品，在解除复位后，必须通过软件将以下的位设置为输出模式 (通过设置端口寄存器和端口模式寄存器为 0 来指定)。

32 引脚产品: PM0 寄存器的位 2 和位 3

30 引脚产品: PM0 寄存器的位 5 和位 6

20 引脚产品: PM0 寄存器的位 2、3、5 和 6 以及 PM2 寄存器的位 6 和 7

备注 38 引脚产品的端口模式寄存器 0、2 和 14 的格式如上图所示。关于其它产品的端口模式寄存器的格式，请参阅表 4-13. 配置在各个产品上的 PMxx、Pxx、PUxx、PIMxx、POMxx、PMCx 寄存器及其对应的位。

可以通过 A/D 端口配置寄存器 (ADPC) 和 PM2 寄存器来选择 ANI2/CMP0P/P22、ANI4/CMP1P/P24 至 ANI7/CMP4P/P27 引脚和 PGAOUT/P23 引脚 (内部引脚)的功能。

表 13-2. ANI2/CMP0P/P22、ANI4/CMP1P/P24 至 ANI7/CMP4P/P27 引脚以及 PGAOUT/P23 引脚(内部引脚)功能的设置

ADPC	PM2	ANI2/CMP0P/P22, ANI4/CMP1P/P24至 ANI7/CMP4P/P27引脚, 以及 PGAOUT/P23引脚 (内部引脚)
数字输入/输出选择	输入模式	数字输入
	输出模式	数字输出
模拟输入选择	输入模式	模拟输入
	输出模式	禁止设置

可以通过端口模式控制寄存器 0, 14 (PMC0, PMC14)、PM0 和 PM14 寄存器来选择 ANI16/CMP5P/RxD1/P03 和 ANI18/CMPCOM/P147 引脚的功能。

表 13-3. ANI16/CMP5P/RxD1/P03 和 ANI18/CMPCOM/P147 引脚功能的设置

PMC0, PMC14	PM0, PM14	ANI16/CMP5P/RxD1/P03 and ANI18/CMPCOM/P147引脚
数字输入/输出选择	输入模式	数字输入
	输出模式	数字输出
模拟输入选择	输入模式	模拟输入
	输出模式	禁止设置

13.4 可编程增益放大器的操作

从 CMP0P 至 CMP5P 和 CMPCOM 引脚输入的模拟电压在微控制器中进行增益。增益率可从 4 种类型中选择 (×4, ×8, ×16 和×32)。

增益电压可用作 A/D 转换器的模拟输入。

可编程增益放大器的开始操作步骤如下所述。

- <1> 使用 ADPC、PMC0 和 PMC14 寄存器将可编程增益放大器中使用的引脚(CMP0P 至 CMP5P 和 CMPCOM)设置为模拟输入。
- <2> 使用 PM0、PM2 和 PM14 寄存器将可编程增益放大器中使用的引脚(CMP0P 至 CMP5P 和 CMPCOM)设置为输入模式。
- <3> 用 PGAVG0 和 PGAVG1 位来选择增益率(×4, ×8, ×16 和×32)。
- <4> 使用 PGAINS0 至 PGAINS2 位来选择充当可编程增益放大器输入的引脚。
- <5> 使用 ADS 寄存器来选择 PGAOUT 作为模拟输入通道。
- <6> 设置 PGAEN 位为(1)并允许可编程增益放大器的操作。

13.5 可编程增益放大器的设置步骤

图 13-9. 可编程增益放大器(PGA)的操作设置流程图
(使用 PGA 输出作为 A/D 转换器的模拟输入)

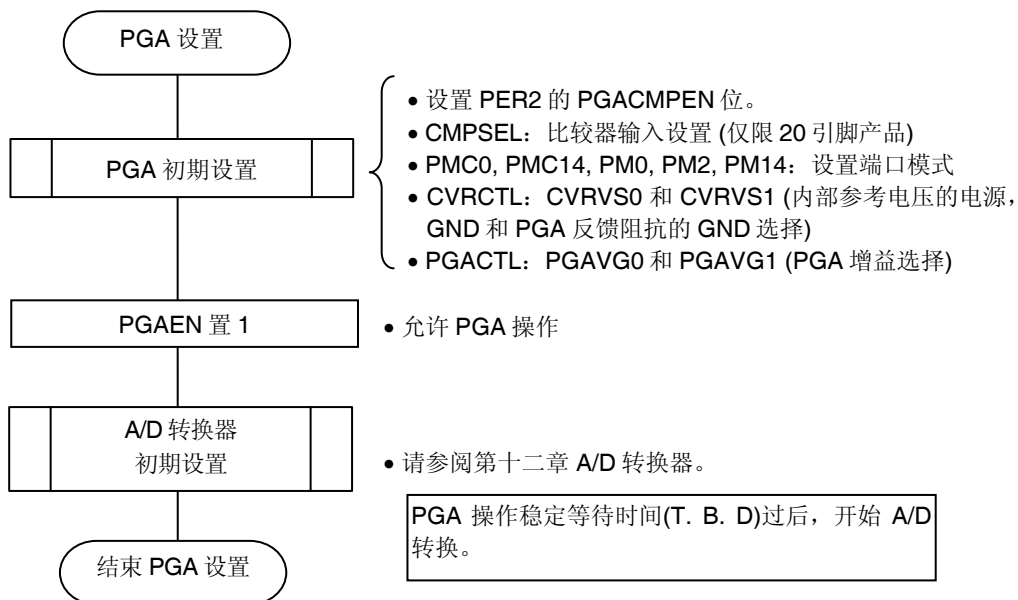
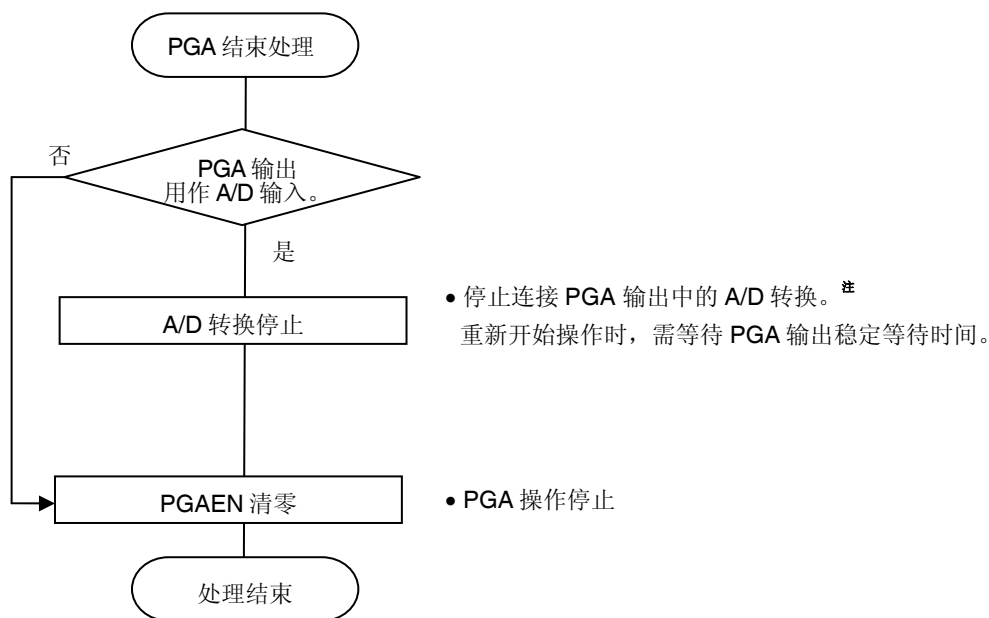


图 13-10. 可编程增益放大器的操作结束流程图



• 切换连接 PGA 输出中的 A/D 或者重新开始操作时, 应执行 PGA 结束流程后, 进行操作设置流程。

注 可以进行未连接PGA输出的A/D转换。

第十四章 比较器

比较器的通道数量因产品而异。

	20引脚	32引脚	30引脚, 38引脚
通道 (模拟输入通道)	4通道 (ANI2/CMP0P, ANI4/CMP1P ANI5/CMP2P, ANI18/(CMP3P)/(CMPCOM))	5通道 (ANI2/CMP0P, ANI4/CMP1P至 ANI7/CMP4P, P147/CMPCOM/ANI18)	6通道 (ANI2/CMP0P, ANI4/CMP1P至 ANI7/CMP4P, ANI16/CMP5P, P147/CMPCOM/ANI18))

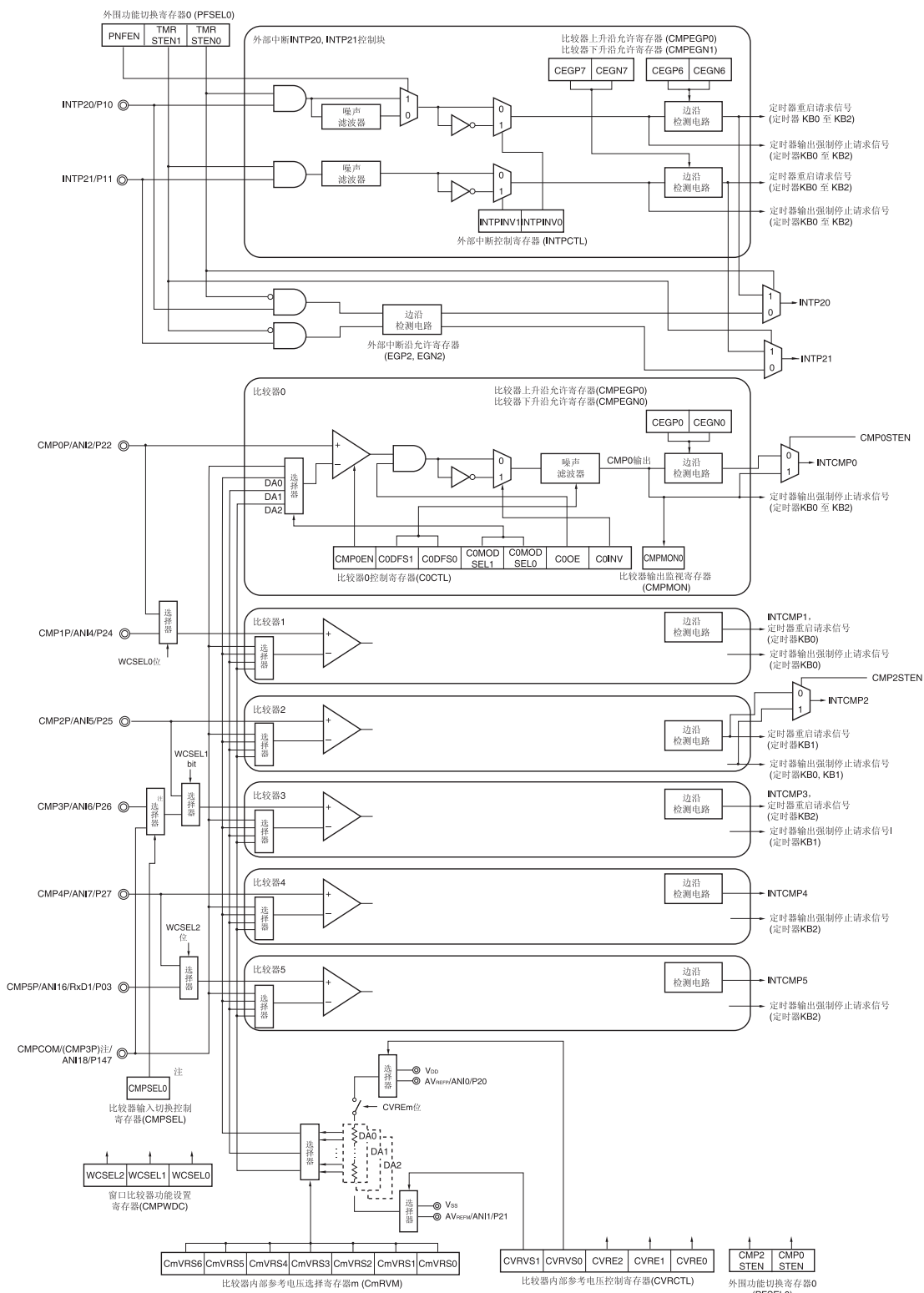
注意事项 本章的下述内容大多是以 **38** 引脚产品为例。

14.1 比较器的功能

比较器具备以下的功能。

- 比较器配置了 6 个通道(比较器 0 至 5)。
- 可选择以下的基准电压。
 - <1> 内部基准电压: 3 (基于 V_{DD}/AV_{REFP} 和 V_{SS}/AV_{REFM} 的 256 分辨率)
 - <2> 来自外部基准电压输入引脚 (CMPCOM)的输入电压
- 通过检测比较器输入的有效沿可产生中断信号。利用 CEGPn 和 CEGNn 位 (n = 0 至 5)可设置有效沿。
- 比较器的输出可用作 16 位定时器 KB0、KB1 和 KB2 的 PWM 输出和定时器计数的复位(参阅第七章 16 位定时器 KB0、KB1 和 KB2)。
- 可选择噪声消除数字滤波器的消除宽度。
- 窗口比较器功能 (比较器 0 和 1、比较器 2 和 3、比较器 4 和 5)
为两个比较器选择同一个输入电压时, 使用该功能可以将一个输入电压与两个基准电压进行比较。

图 14-1. 比较器的框图



注 仅限 20 引脚产品。30 引脚、32 引脚和 38 引脚产品的默认选择为 ANI16/CMP3P/P26。

备注 m = 0 至 2

14.2 比较器的配置

比较器包括以下硬件。

表 14-1. 比较器的配置

项目	配置
比较器本体	比较器 最大 6 通道
基准电压生成电路	内部基准电压 3 和外部 CAPCOM 输入
噪声滤波器	噪声消除数字滤波器
控制寄存器	外围允许寄存器 2 (PER2) 比较器 n 控制寄存器(CnCTL) 比较器内部基准电压控制寄存器(CVRCTL) 比较器内部基准电压选择寄存器 m (CmRVM) 比较器上升沿允许寄存器(CMPEGP0) 比较器下降沿允许寄存器(CMPEGN0) 比较器输出监视寄存器(CMPMON) 窗口比较器功能设置寄存器(CMPWDC) 比较器输入切换控制寄存器(CMPSEL) (仅限 20 引脚产品) 外部中断控制寄存器(INTPCTL) 外围功能切换寄存器 0 (PFSEL0) A/D 端口配置寄存器(ADPC) 端口模式控制寄存器 0, 14 (PMC0, PMC14) 端口模式寄存器 0, 2, 14 (PM0, PM2, PM14)

14.3 控制比较器的寄存器

比较器使用以下 14 种寄存器。

- 外围允许寄存器 2 (PER2)
- 比较器 n 控制寄存器(CnCTL)
- 比较器内部基准电压控制寄存器(CVRCTL)
- 比较器内部基准电压选择寄存器 m (CmRVM)
- 比较器上升沿允许寄存器(CMPEGP0)
- 比较器下降沿允许寄存器(CMPEGN0)
- 比较器输出监视寄存器(CMPMON)
- 窗口比较器功能设置寄存器(CMPWDC)
- 比较器输入切换控制寄存器(CMPSEL) (仅限 20 引脚产品)
- 外部中断控制寄存器(INTPCTL)
- A/D 端口配置寄存器(ADPC)
- 外围功能切换寄存器 0 (PFSEL0)
- 端口模式控制寄存器 0, 14 (PMC0, PMC14)
- 端口模式寄存器 0, 2, 14 (PM0, PM2, PM14)

备注 n = 0 至 5, m = 0 至 2

(1) 外围允许寄存器 2 (PER2)

该寄存器用于允许或禁止供应时钟至外围硬件。并且中止向未使用的硬件供应时钟，以减少电力消耗和噪声。

当使用比较器时，务必确实设置该寄存器的位 7 (PGACMPEN) 为 1。

使用 1 位或 8 位存储器操作指令设置 PER2 寄存器。

产生复位信号后，清除该寄存器为 00H。

图 14-2. 外围允许寄存器 2 (PER2)的格式

地址: F0509H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
PER2	PGACMPEN	TKBPA2EN	TKBPA1EN	TKBPA0EN	TKC0EN	TKB2EN ^注	TKB1EN	TKB0EN

PGACMPEN	比较器/可编程增益放大器输入时钟控制
0	停止输入时钟的供应。 <ul style="list-style-type: none"> 不可写入用于比较器或可编程增益放大器的SFR。 比较器或可编程增益放大器处于复位状态。
1	供应输入时钟。 <ul style="list-style-type: none"> 可以读取和写入用于比较器或可编程增益放大器的SFR。

注 仅限 20 引脚产品、32 引脚产品和 38 引脚产品。

(2) 比较器 n 控制寄存器(CnCTL)

该寄存器用来控制比较器 n 的操作、允许或禁止比较器输出、反相输出、设置噪声消除宽度和基准电压。

使用 1 位或 8 位存储器操作指令设置 CnCTL 寄存器。

产生复位信号后，清除该寄存器为 00H。

备注 n = 0 至 5

图 14-3. 比较器 n 控制寄存器 n (CnCTL)的格式 (1/2)

地址: F0552H (C0CTL), F0553H (C1CTL), F0554H (C2CTL) 复位后: 00H R/W
 F0555H (C3CTL), F0556H (C4CTL), F0557H (C5CTL)

符号	7	6	5	4	3	2	1	0
CnCTL	CMPnEN	CnDFS1	CnDFS0	CnMODSEL1	CnMODSEL0	0	CnOE	CnINV

CMPnEN	比较器n操作的控制
0	停止操作。 比较器n的输出信号为低电平。
1	允许操作。 允许输入至比较器n。

图 14-3. 比较器 n 控制寄存器 n (CnCTL)的格式 (2/2)

地址: F0552H (C0CTL), F0553H (C1CTL), F0554H (C2CTL) 复位后: 00H R/W
F0555H (C3CTL), F0556H (C4CTL), F0557H (C5CTL)

符号	7	6	5	4	3	2	1	0
CnCTL	CMPnEN	CnDFS1	CnDFS0	CnMODSEL1	CnMODSEL0	0	CnOE	CnINV
CnDFS1		CnDFS0		噪声消除宽度的设置				
0		0		不使用噪声滤波器				
0		1		$2^3/f_{CLK}, 2^3/f_{PLL}$ ‡ (当 $f_{PLL} = 64$ MHz, 125至187.5 ns)				
1		0		$2^4/f_{CLK}, 2^4/f_{PLL}$ ‡ (当 $f_{PLL} = 64$ MHz, 250至375 ns)				
1		1		$2^5/f_{CLK}, 2^5/f_{PLL}$ ‡ (当 $f_{PLL} = 64$ MHz, 500至750 ns)				
CnMODSEL1		CnMODSEL0		基准电压的选择				
0		0		内部基准电压: DA0				
0		1		内部基准电压: DA1				
1		0		内部基准电压: DA2				
1		1		外部基准电压: CMPCOM				
CnOE		允许或禁止比较器输出						
0		禁止比较器输出 (禁止输出定时器强制输出停止请求信号(输出信号 = 固定在低电平))						
1		允许比较器输出 (允许输出定时器强制输出停止请求信号)						
CnINV		输出反相的设置						
0		正相						
1		反相						

注 当 PLL 控制寄存器(PLLCTL)中 PLLON = 1 时, 供应 f_{PLL} 。

- 注意事项 1. 请将比较器 n 输出设置为禁止状态(CnOE = 0)后, 重写 CnDFS1、CnDFS0、CnMODSEL1、CnMODSEL0 和 CnINV 位。请在比较器停止操作(CMPnEN = 0)时重写 CnMODSEL1 和 CnMODSEL0 位。
- 关于噪声消除宽度, 可能比设定值多消除 1 个 CPU/外围硬件时钟频率(f_{CLK})或 PLL 输出时钟(f_{PLL})。
 - 如果比较器输出的噪声间隔在“设定的噪声消除宽度 + 1 时钟”范围内, 则可能输出非法波形。
 - 要使用内部基准电压, 允许(CVREn = 1)内部基准电压的操作后再允许(CMPnEN = 1)比较器的操作。
 - 在允许(CMPnEN = 1)比较器操作后需要有操作稳定等待时间(T.B.D.)。

- 备注 1. f_{CLK} : CPU/外围硬件时钟频率
2. n = 0 至 5

(3) 比较器内部基准电压控制寄存器(CVRCTL)

该寄存器用于控制比较器的内部基准电压，或者选择内部基准电压源、比较器或可编程增益放大器的 GND。

使用 CVREm 位可允许或停止内部基准电压。

使用 1 位或 8 位存储器操作指令设置 CVRCTL 寄存器。

产生复位信号后，清除该寄存器为 00H。

图 14-4. 比较器内部基准电压控制寄存器(CVRCTL)的格式

地址: F0560 H 复位后: 00H R/W

符号	7	6	5	4	3	<2>	<1>	<0>
CVRCTL	0	0	CVRVS1	CVRVS0	0	CVRE2	CVRE1	CVRE0
CVRVS1	内部基准电压、可编程增益放大器的GND选择							
0	V _{SS}							
1	AV _{REFM}							
CVRVS0	内部基准电压的电源选择							
0	V _{DD}							
1	AV _{REFP}							
CVREm	产生内部基准电压(DAm)的操作控制							
0	停止操作							
1	允许操作							

注意事项 1. 在设置 CVREm 位后需要有稳定等待时间(20 μs)。

2. 内部基准电压 (CVREm = 0)停止操作时重写 CVRVS1 和 CVRVS0 寄存器。请在可编程增益放大器的操作停止 (PGACTL 寄存器的 PGAEN = 0) 时重写 CVRVS1 位。

备注 m = 0 至 2

(4) 比较器内部基准电压选择寄存器 m (CmRVM)

该寄存器用于设置比较器的内部基准电压电平。

使用 8 位存储器操作指令设置 CmRVM 寄存器。

产生复位信号后，清除该寄存器为 00H。

图 14-5. 比较器内部基准电压选择寄存器(CmRVM)的格式

地址: 0561H (C0RVM), F0562H (C1RVM), F0563H (C2RVM) 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
CmRVM	CmVRS7	CmVRS6	CmVRS5	CmVRS4	CmVRS3	CmVRS2	CmVRS1	CmVRS0

CmVRS7至CmVRS0	内部基准电压(DAm)的电平设置
00000000	$((AV_{REFP} \text{或} V_{DD}) / 256) \times 0$
00000001	$((AV_{REFP} \text{或} V_{DD}) / 256) \times 1$
00000010	$((AV_{REFP} \text{或} V_{DD}) / 256) \times 2$
•	•
•	•
•	•
11111101	$((AV_{REFP} \text{或} V_{DD}) / 256) \times 253$
11111110	$((AV_{REFP} \text{或} V_{DD}) / 256) \times 254$
11111111	$((AV_{REFP} \text{或} V_{DD}) / 256) \times 255$

注意事项 内部基准电压的操作停止(CVREm = 0)时重写 CmRVM 寄存器。

备注 m = 0 至 2

(5) 比较器上升沿允许寄存器 0 (CMPEGP0)、比较器下降沿允许寄存器 0 (CMPEGN0)

这些寄存器用来设置比较器 n 检测中断信号(INTCMPn)或者用于定时器输出强制停止的外部中断(INTP20, INTP21)的有效沿。

使用 1 位或 8 位存储器操作指令来设置 CMPEGP0 和 CMPEGN0 寄存器。

产生复位信号后，清除该寄存器为 00H。

图 14-6. 比较器上升沿允许寄存器 0 (CMPEGP0)、比较器下降沿允许寄存器 0 (CMPEGN0)的格式

地址: F0558 H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
CMP EGP0	CEGP7	CEGP6	CEGP5	CEGP4	CEGP3	CEGP2	CEGP1	CEGP0

地址: F0559 H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
CMP EGN0	CEGN7	CEGN6	CEGN5	CEGN4	CEGN3	CEGN2	CEGN1	CEGN0

CEGP7	CEGN7	INTP21引脚有效沿的选择
0	0	禁止边沿检测 (禁止定时器强制输出停止信号的输出(输出信号 = 固定在低电平))
0	1	下降沿 (允许定时器强制输出停止信号的输出)
1	0	上升沿 (允许定时器强制输出停止信号的输出)
1	1	上升和下降双沿 (允许定时器强制输出停止信号的输出)

CEGP6	CEGN6	INTP20引脚有效沿的选择
0	0	禁止边沿检测 (禁止定时器强制输出停止信号的输出(输出信号 = 固定在低电平))
0	1	下降沿 (允许定时器强制输出停止信号的输出)
1	0	上升沿 (允许定时器强制输出停止信号的输出)
1	1	上升和下降双沿 (允许定时器强制输出停止信号的输出)

CEGPn	CEGNn	INTCMPn有效沿的选择
0	0	禁止检测边沿
0	1	下降沿
1	0	上升沿
1	1	上升和下降沿

注意事项 通过 CnCTL 寄存器的 CnINV 位，对于将比较器检测信号不反相或反相以后得到的信号来设置有效沿。

备注 n = 0 至 5

(6) 比较器输出监视寄存器(CMPMON)

该寄存器显示定时器强制输出停止请求信号(比较器的输出信号)的电平。

仅可使用 1 位或 8 位存储器操作指令来读取 CMPMON 寄存器。

产生复位信号后, 清除该寄存器为 00H。

图 14-7. 比较器输出监视寄存器(CMPMON)的格式

地址: F055AH 复位后: 00H R

符号	7	6	5	4	3	2	1	0
CMPMON	0	0	CMPMON5	CMPMON4	CMPMON3	CMPMON2	CMPMON1	CMPMON0

CMPMONn	比较器n的输出电平 (n = 0至5)
0	定时器强制输出停止请求信号为低电平
1	定时器强制输出停止请求信号为高电平

(7) 窗口比较器功能设置寄存器(CMPWDC)

窗口比较器功能选择一个输入电压用于两个比较器，因此一个输入电压可以和两个基准电压进行比较。使用该窗口比较器功能时，CMPWDC 寄存器可用来控制对输入信号的选择。

使用 1 位或 8 位存储器操作指令设置 CMPWDC 寄存器。

产生复位信号后，清除该寄存器为 00H。

图 14-8. 窗口比较器功能设置寄存器(CMPWDC)的格式

地址: F0564H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
CMPWDC	0	0	0	0	0	WCSEL2	WCSEL1	WCSEL0

WCSEL2	比较器5的+ 侧输入的选择
0	CMP5P
1	CMP4P
设置为WCSEL2 = 1时: 利用比较器4和比较器5对CMP4的输入电压和基准电压进行比较	

WCSEL1	比较器3的+ 侧输入的选择
0	CMP3P
1	CMP2P
设置为WCSEL1 = 1时: 利用比较器2和比较器3对CMP2的输入电压和基准电压进行比较	

WCSEL0	比较器1的+ 侧输入的选择
0	CMP1P
1	CMP0P
设置为WCSEL0 = 1时: 利用比较器1和比较器0对CMP0的输入电压和基准电压进行比较	

注意事项 请在比较器停止操作(CMPnEN = 0)时重写 CMPWDC 寄存器。

(8) 比较器输入切换控制寄存器(CMPSEL) (仅限 20 引脚产品)

该寄存器用于设置比较器 3 的输入信号。

使用 8 位存储器操作指令来设置 CMPSEL 寄存器。

产生复位信号后，清除该寄存器为 00H。

图 14-9. 比较器输入切换控制寄存器(CMPSEL)的格式

地址: F0565H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
CMPSEL	0	0	0	0	0	0	0	CMPSEL0

CMPSEL0	比较器3的+ 侧输入控制
0	20引脚产品中不使用比较器3。
1	(CMP3P)/CMPCOM/ANI18/P147引脚

注意事项 请在比较器停止操作(CMP3EN = 0)时重写 CMPSEL 寄存器。

(9) 外部中断控制寄存器(INTPCTL)

该寄存器是对用于定时器输出强制停止的外部中断 INTP20 和 INTP21 的反相输出进行设置。

使用 1 位或 8 位存储器操作指令设置 INTPCTL 寄存器。

产生复位信号后，清除该寄存器为 00H。

图 14-10. 外部中断控制寄存器(INTPCTL)的格式

地址: F055BH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
INTPCTL	0	0	0	0	0	0	INTPINV1	INTPINV0

INTPINV1	INTP21信号的输出反相设置
0	INTP21信号正相输出
1	INTP21信号反相输出

INTPINV0	INTP20信号的输出反相设置
0	INTP20信号正相输出
1	INTP20信号反相输出

注意事项 请在 INTP20 和 INTP21 边沿检测禁止(CEGPn 和 CEGNn = 0, 0) (n = 6, 7)时重写 INTPCTL 寄存器。

(10) A/D 端口配置寄存器(ADPC)

该寄存器将 ANI0/P20、ANI1/P21、ANI2/CMP0P/P22、ANI4/CMP1P/P24 至 ANI7/CMP4P/P27 引脚在端口的数字输入/输出以及 A/D 转换器、可编程增益放大器或比较器的模拟输入之间进行切换。

当使用可编程增益放大器或比较器时，使用 ADPC 寄存器来选择 CMP0P/ANI2/P22 和 CMP1P/ANI4/P24 至 CMP4P/ANI7/P27 引脚作为模拟输入。

使用 8 位存储器操作指令来设置 ADPC 寄存器。

产生复位信号后，设置该寄存器为 00H。

图 14-11. A/D 端口配置寄存器(ADPC)的格式

地址: F0076H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ADPC	0	0	0	0	ADPC3	ADPC2	ADPC1	ADPC0

ADPC3	ADPC2	ADPC1	ADPC0	模拟输入(A)/数字 I/O(D)切换							
				ANI7/ CMP4P/P27	ANI6/ CMP3P/P26	ANI5/ CMP2P/P25	ANI4/ CMP1P/P24	PGAOUT ^注	ANI2/ CMP0P/P22	ANI1/P21/ AVREFM	ANI0/P20/ AVREFP
0	0	0	0	A	A	A	A	A	A	A	A
0	0	0	1	D	D	D	D	D	D	D	D
0	0	1	0	D	D	D	D	D	D	D	A
0	0	1	1	D	D	D	D	D	D	A	A
0	1	0	0	D	D	D	D	D	A	A	A
0	1	0	1	D	D	D	D	A	A	A	A
0	1	1	0	D	D	D	A	A	A	A	A
0	1	1	1	D	D	A	A	A	A	A	A
1	0	0	0	D	A	A	A	A	A	A	A
1	0	0	1	A	A	A	A	A	A	A	A
1	1	1	1	A	A	A	A	A	A	A	A
其它				禁止设置							

注 这是一个可编程增益放大器的内部输出引脚。将可编程增益放大器的输出信号用作 A/D 转换器的模拟输入通道时，设置 ADPC 等于 0000B 或大于等于 0101B。

注意事项 通过端口模式寄存器 2 (PM2)将用于 A/D 转换的通道设置为输入模式。

(11) 外围功能切换寄存器 0 (PFSELO)

PFSELO 寄存器的位 0 和位 1 通过外部中断 INTP20 和 INTP21 来实现 16 位定时器 KB0、KB1 和 KB2 的 PWM 控制或清除停止模式。

位 4 选择是否使用 INTP20 噪声滤波器。在强制输出停止功能 2 中使用 INTP20 时，选择不使用噪声滤波器，以加快触发输入和停止输出之间的反应时间。

位 5 和位 6 用于选择 CMP0 和 CMP2 的检测中断功能。关于其它位的详情，请参阅 7.3 (16) 外围功能切换寄存器 0 (PFSELO)。

使用 1 位或 8 位存储器操作指令来写入 PFSELO 寄存器。

产生复位信号后，清除该寄存器为 00H。

图 14-12. 外围功能切换寄存器 0 (PFSELO)的格式

地址: F05C6H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
PFSELO	0	CMP2STEN	CMP0STEN	PNFEN	ADTRG11	ADTRG10	TMRSTEN1	TMRSTEN0

CMP2STEN	比较器2检测中断(INTCMP2)的切换
0	禁止解除STOP模式
1	允许解除STOP模式，但仅限于不使用噪声滤波器时 (可在低消耗RTC模式 (OSMC寄存器中的RTCLPC = 1)操作时设置)

CMP0STEN	比较器0检测中断(INTCMP0)的切换
0	禁止解除STOP模式
1	允许解除STOP模式，但仅限于不使用噪声滤波器时 (可在低消耗RTC模式 (OSMC寄存器中的RTCLPC = 1)操作时设置)

PNFEN	使用/不使用外部中断INTP20的噪声滤波器
0	使用噪声滤波器
1	不使用噪声滤波器

TMRSTEN1	外部中断INTP21的用途选择
0	可用于解除STOP模式 (不可用于定时器重启功能)
1	可用于定时器重启功能(不可用于解除STOP模式)

TMRSTEN0	外部中断INTP20的用途选择
0	可用于解除STOP模式 (不可用于定时器重启功能)
1	可用于定时器重启功能(不可用于解除STOP模式)

(12) 端口模式控制寄存器 0, 14 (PMC0, PMC14)

使用该寄存器按 1 位单位设置端口 0、14 的数字 I/O 或模拟输入。

当使用 CMP5P/ANI16/RxD1/P03 或 CMPCOM/ANI18/P147 引脚作为模拟输入端口时，将 PMC03 或 PMC147 位设置为 1。

用 1 位或 8 位存储器操作指令来设置 PMC0 和 PMC14 寄存器。

产生复位信号后，设置该寄存器为 FFH。

图 14-13. 端口模式控制寄存器 0、14 (PMC0, PMC14)的格式

地址: F0060H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PMC0	1	1	1	1	PMC03	PMC02	1	1

地址: F006EH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PMC14	PMC147	1	1	1	1	1	1	1

PMCmn	选择Pmn引脚数字输入/输出和模拟输入(m = 02, 03, 147)
0	数字输入/输出(模拟输入以外的复用功能)
1	模拟输入

(13) 端口模式寄存器 0, 2, 14 (PM0, PM2, PM14)

当使用 CMP0P/ANI2/P22、CMP1P/ANI4/P24 至 CMP4P/ANI7/P27、CMP5P/ANI16/RxD1/P03 或 CMPCOM/ANI18/P147 引脚作为模拟输入端口时，将 PM22、PM24 至 PM27、PM03 或 PM147 位设置为 1。此时，P22、P24 至 P27、P03 和 P147 的输出锁存器可为 0 或 1。

如果 PM22、PM24 至 PM27、PM03 和 PM147 位被设置为 0，则不能用作模拟输入端口引脚。

使用 1 位或 8 位存储器操作指令来设置 PM0、PM2 和 PM14 寄存器。

产生复位信号后，设置该寄存器为 FFH。

注意事项 读取被设置为模拟输入端口的引脚时，即使 $PM_{xx} = 1$ ，读取值也不是引脚电平而是一直为“0”。

图 14-14. 端口模式寄存器 0, 2, 14 (PM0, PM2, PM14)的格式

地址: FFF20H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM0	1	PM06	PM05	1	PM03	PM02	1	1

地址: FFF22H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM2	PM27	PM26	PM25	PM24	1	PM22	PM21	PM20

地址: FFF2EH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM14	PM147	1	1	1	1	1	1	1

PMmn	Pmn引脚输入/输出模式选择(m = 0, 2, 14; n = 0至7)
0	输出模式 (输出缓冲器on)
1	输入模式 (输出缓冲器off)

注意事项 务必将 PM0 寄存器的位 0, 1, 4, 7、PM2 寄存器的位 3 和 PM14 寄存器的位 0 至 6 设置为“1”。对于 32, 30 和 20 引脚产品，在解除复位后，必须通过软件将以下的位设置为输出模式 (通过设置端口寄存器和端口模式寄存器为 0 来指定)。

32 引脚产品: PM0 寄存器的位 2 和位 3

30 引脚产品: PM0 寄存器的位 5 和位 6

20 引脚产品: PM0 寄存器的位 2、3、5 和 6 以及 PM2 寄存器的位 6 和 7

可以通过 A/D 端口配置寄存器(ADPC)和 PM2 寄存器来选择 CMP0P/ANI2/P22 和 CMP1P/ANI4/P24 至 CMP4P/ANI7/P27 引脚的功能。

表 14-2. CMP0P/ANI2/P22 和 CMP1P/ANI4/P24 至 CMP4P/ANI7/P27 引脚功能的设置

ADPC	PM2	CMP0P/ANI2/P22和CMP1P/ANI4/P24至 CMP4P/ANI7/P27引脚
数字输入/输出选择	输入模式	数字输入
	输出模式	数字输出
模拟输入选择	输入模式	模拟输入
	输出模式	禁止设置

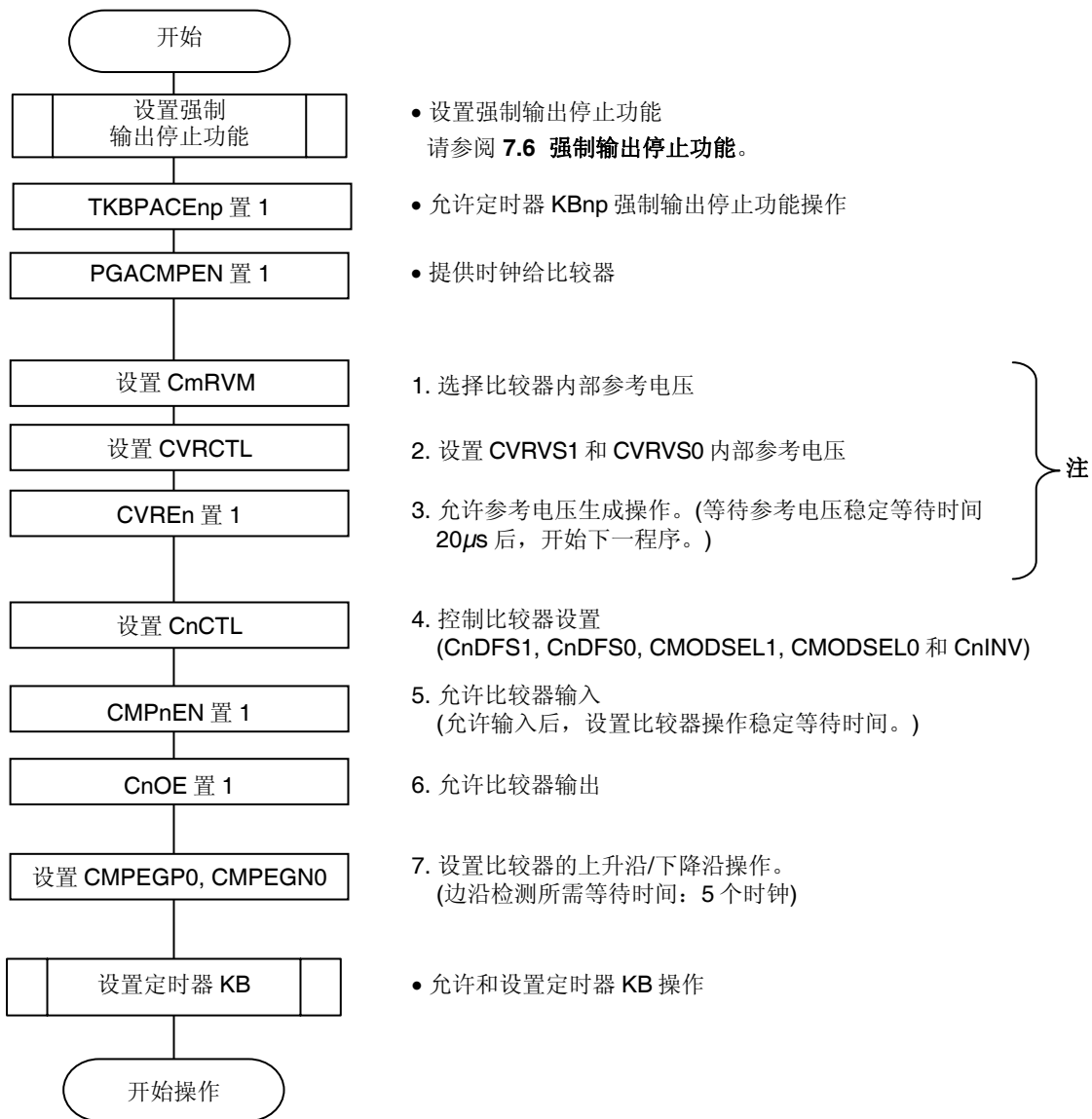
可以通过端口模式控制寄存器 0, 14 (PMC0, PMC14)、PM0 和 PM14 寄存器来选择 CMP5P/ANI16/RxD1/P03 和 CMPCOM/ANI18/P147 引脚的功能。

表 14-3. CMP5P/ANI16/RxD1/P03 和 CMPCOM/ANI18/P147 引脚功能的设置

PMC0, PMC14	PM0, PM14	CMP5P/ANI16/RxD1/P03和CMPCOM/ANI18/P147 引脚
数字输入/输出选择	输入模式	数字输入
	输出模式	数字输出
模拟输入选择	输入模式	模拟输入
	输出模式	禁止设置

14.4 比较器的设置步骤

图 14-15. 比较器(CMP)的操作设置流程 1
(使用 INTCMPn、CMPnHZO 输出时)



注 当外部引脚 CMPCOM 用作基准电压时, 不要求此项。

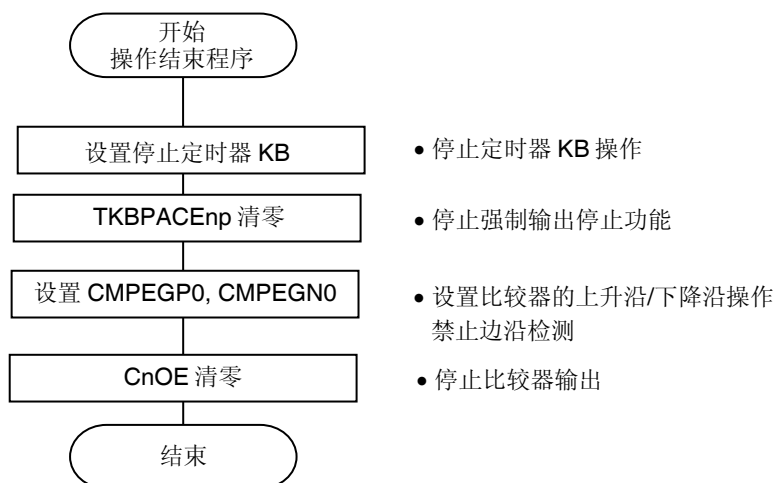
注意事项 应在 INTCMP 处理禁止状态下设置上述 1.至 7.项。

图 14-16. 比较器(CMP)的操作设置流程 2
(使用 INTPm、INTPmHZO 输出时(仅使用边沿电路))



注意事项 应在 INTCMP 处理禁止状态下设置上述 1.至 2.项。

图 14-17. 比较器的操作结束流程图



第十五章 串行阵列单元 0

串行阵列单元 0 有四个串行通道。各通道可以实现 3 线串行(CSI)和 UART 通信。
RL78/I1A 中支持的各通道的功能分配如下所示。

• 20、32 引脚产品

通道	用作 CSI	用作 UART	用作简易 I ² C
0	-	UART0 (支持 LIN-bus、DMX512)	-
1	-		-
2	-	-	-
3	-		-

• 30 引脚产品

通道	用作 CSI	用作 UART	用作简易 I ² C
0	-	UART0 (支持 LIN-bus、DMX512)	-
1	-		-
2	-	UART1	-
3	-		-

• 38 引脚产品

通道	用作 CSI	用作 UART	用作简易 I ² C
0	CSI00	UART0 (支持 LIN-bus、DMX512)	-
1	-		-
2	-	UART1	-
3	-		-

38 引脚产品中，在通道 0 和 1 使用“UART0”时，不能使用 CSI00，但可以使用通道 2、3 的“UART1”。

注意事项 本章的下述内容大多是以 38 引脚产品的单元和通道为例。

15.1 串行阵列单元 0 的功能

RL78/I1A 支持的各个串行接口均具有如下特点。

15.1.1 3 线串行输入/输出 (CSI00)

与主通道输出的串行时钟($\overline{\text{SCK}}$)同步后执行数据的发送或接收。

3 线串行通信为时钟同步式通信功能，通过三条通信线路实现：一条用于串行时钟($\overline{\text{SCK}}$)、一条用于发送串行数据(SO)、一条用于接收串行数据(SI)。

有关具体的设置示例，请参阅 **15.5 3 线串行输入/输出(CSI00)通信的操作**。

[数据的发送/接收]

- 数据长度为 7 或 8 位
- 发送/接收数据的相控制
- MSB/LSB 选择
- 发送/接收数据的电平设置

[时钟控制]

- 主/从属的选择
- 输入/输出时钟的相控制
- 通过预分频器和通道内部计数器设置传送周期
- 最大传送速率

主通信时(CSI00): $\text{Max. } f_{\text{MCK}}/2$ ^{註1,2}

主通信时(CSI00 以外): $\text{Max. } f_{\text{MCK}}/4$ ^{註2}

从通信时: $\text{Max. } f_{\text{MCK}}/6$ ^{註2}

[中断功能]

- 传送结束中断/缓冲器空中断

[错误检测标志]

- 溢出错误

另外，CSI00 (通道 0)支持 SNOOZE 模式。借助 SNOOZE 模式，在 STOP 模式下检测到 $\overline{\text{SCK00}}$ 引脚输入时不需要 CPU 就执行数据接收。

注 1. 主通信(CSI00)模式下，满足以下条件时最大传送速率为 $f_{\text{MCK}}/2$:

- $2.7 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$
- $f_{\text{MCK}} \leq 24 \text{ MHz}$
- $\text{PIOR1} = 0$

其他情况下的最大传送速率为 $f_{\text{MCK}}/4$ 。

2. 在满足 $\overline{\text{SCK}}$ 周期时间(t_{KCY})特性的范围内使用时钟(参阅**第三十二章 电特性**)。

15.1.2 UART (UART0, UART1)

这是一种使用 2 线：串行数据发送线(TxD)和串行数据接收线(RxD)的异步通信功能。利用这两条通信线路，各数据帧（由一个起始位、数据、奇偶校验位和停止位构成）在单片机与其他通信方之间（以内部波特率）异步传送。全双工 UART 通信可以利用一个发送专用通道（偶数通道）和一个接收专用通道（奇数通道）来实现。可以结合使用定时器阵列单元和外部中断(INTP0)来实现支持 LIN-bus 和 DMX512。

有关具体的设置例，请参阅 15.6 UART(UART0、UART1)通信的操作。

[数据的发送/接收]

- 数据长度为 7、8 或 9 位^注
- MSB/LSB 选择
- 发送/接收数据的电平设置及反相选择
- 附加奇偶校验位和奇偶校验功能
- 附加停止位

[中断功能]

- 传送结束中断/缓冲器空中断
- 出现帧错误、奇偶检验错误或溢出错误时的错误中断

[错误检测标志]

- 帧错误、奇偶校验错误、溢出错误

另外，UART0 接收(通道 1)支持 SNOOZE 模式。借助 SNOOZE 模式，在 STOP 模式下检测到 RxD 输入时不需要 CPU 动作就可以接收数据。

UART0 (通道 0 和 1)支持 LIN-bus 和 DMX512。

[LIN-bus 功能]

- 唤醒信号检测
- 间隔段(BF)检测
- 同步段测量、波特率计算

使用外部中断(INTP0)和
定时器阵列单元 0

[DMX512 功能]

- BREAK 信号检测
- 脉冲宽度检测

使用外部中断(INTP0)和
定时器阵列单元 0

注 仅限以下 UART 可以指定 9 位的数据长度。

15.2 串行阵列单元 0 的配置

定时器阵列单元 0 包含以下硬件。

表 15-1. 串行阵列单元 0 的配置

项目	配置
移位寄存器	8 位或 9 位 ^{※1}
缓冲寄存器	串行数据寄存器 mn (SDRmn) ^{※1,2} 的低 8 位或 9 位
串行时钟输入/输出	$\overline{\text{SCK00}}$ 引脚 (用作 3 线串行输入/输出)
串行数据输入	SI00 引脚 (用作 3 线串行输入/输出), RxD0 引脚 (用作支持 LIN-bus 和 DMX512 的 UART), RxD1 引脚 (用作 UART)
串行数据输出	SO00 引脚 (用作 3 线串行输入/输出), TxD0 引脚 (用作支持 LIN-bus 和 DMX512 的 UART), TxD1 引脚 (用作 UART), 输出控制电路
控制寄存器	<p><单元设置块的寄存器></p> <ul style="list-style-type: none"> • 外围允许寄存器 0 (PER0) • 串行时钟选择寄存器 m (SPSm) • 串行通道允许状态寄存器 m (SEm) • 串行通道开始寄存器 m (SSm) • 串行通道停止寄存器 m (STm) • 串行输出允许寄存器 m (SOEm) • 串行输出寄存器 m (SOM) • 串行输出电平寄存器 m (SOLm) • 串行待机控制寄存器 0 (SSC0) • 输入切换控制寄存器 (ISC) • 噪声滤波器允许寄存器 0 (NFEN0)
	<p><每个通道的寄存器></p> <ul style="list-style-type: none"> • 串行数据寄存器 mn (SDRmn) • 串行模式寄存器 mn (SMRmn) • 串行通信操作设置寄存器 mn (SCRmn) • 串行状态寄存器 mn (SSRmn) • 串行标志清除触发寄存器 mn (SIRmn)
	<ul style="list-style-type: none"> • 端口输入模式寄存器 0, 1 (PIM0, PIM1) • 端口输出模式寄存器 0, 1 (POM0, POM1) • 端口模式寄存器 0, 1 (PM0, PM1) • 端口寄存器 0, 1 (P0, P1)

(注和备注如下页所示。)

注 1. 用于移位寄存器和缓冲寄存器的位数因单元和通道而异。

mn = 00, 01: 低 9 位, mn = 02, 03: 低 8 位

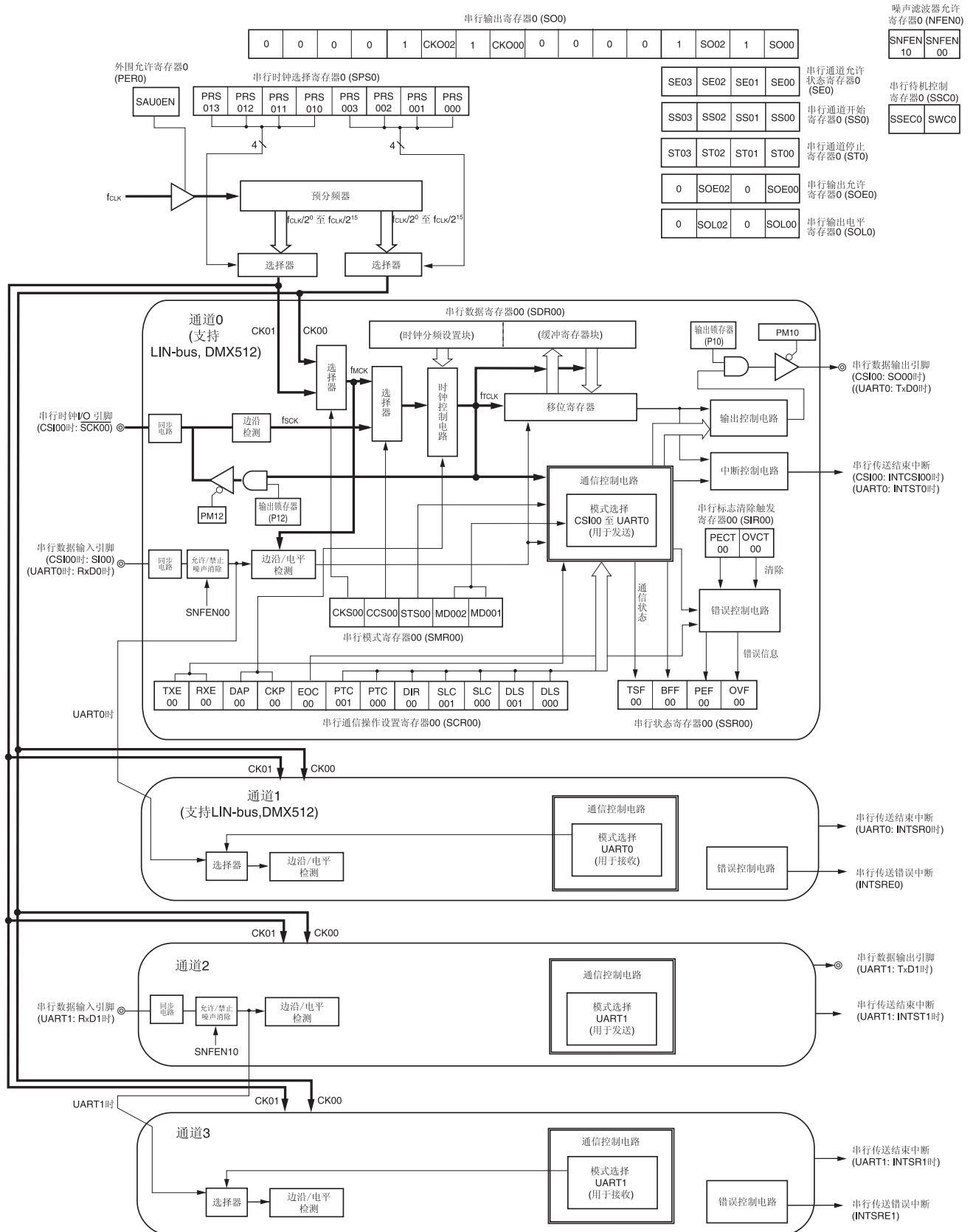
2. 依据通信模式, 串行数据寄存器 mn (SDRmn)的低 8 位可以按以下 SFR 名称读取或写入。

- CSIp 通信 ... SIOp (CSIp 数据寄存器)
- UARTq 接收 ... RxDq (UARTq 接收数据寄存器)
- UARTq 发送 ... TxDq (UARTq 发送数据寄存器)

备注 m: 单元编号(m = 0), n: 通道编号(n = 0 至 3), p: CSI 编号(p = 00), q: UART 编号(q = 0, 1)

串行阵列单元 0 的框图如图 15-1 所示。

图 15-1. 串行阵列单元 0 的框图



(1) 移位寄存器

这是一个 9 位寄存器，可将并行数据转换为串行数据，反之亦然。

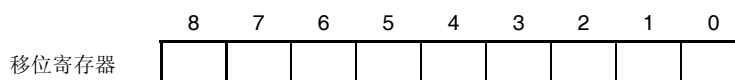
在 9 位数据 UART 通信的情况下，将使用 9 位（位 0 至位 8）^{注1}。

接收期间，将输入至串行输入引脚的数据转换为并行数据。

发送数据时，将设置到该寄存器的值转换为串行数据从串行输出引脚输出。

移位寄存器不能直接通过程序操作。

使用串行数据寄存器 mn (SDRmn) 的低 8/9 位读取或写入移位寄存器。

**(2) 串行数据寄存器 mn (SDRmn) 的低 8/9 位**

SDRmn 寄存器是通道 n 的发送/接收数据寄存器（16 位）。位 8 至位 0（低 9 位）^{注1} 或位 7 至位 0（低 8 位）作为发送/接收缓冲寄存器，位 15 至位 9 作为设置操作时钟 (f_{MCK}, f_{SCK}) 分频比的寄存器。

接收数据时，由移位寄存器转换的并行数据被存储于低 8/9 位。发送数据时，把传送到移位寄存器的发送值设置到低 8/9 位。

依据串行通信操作设置寄存器 mn (SCRmn) 位 0 和位 1 (DLSmn0、DLSmn1) 的设置，存储于该寄存器低 8/9 位的数据的情况如下所示（与数据输出顺序无关）：

- 7 位数据长度 (存储于 SDRmn 寄存器的位 0 至位 6)
- 8 位数据长度 (存储于 SDRmn 寄存器的位 0 至位 7)
- 9 位数据长度 (存储于 SDRmn 寄存器的位 0 至位 8)^{注1}

SDRmn 寄存器可以按照 16 位单位进行读取或写入。

依据通信模式，SDRmn 寄存器的低 8/9 位可以按以下 SFR 名称以 8 位单位读取或写入^{注2}。

- CSIp 通信 ... SIOp (CSIp 数据寄存器)
- UARTq 接收 ... RxDq (UARTq 接收数据寄存器)
- UARTq 发送 ... TxDq (UARTq 发送数据寄存器)

注 停止操作 (SEmn = 0) 时，禁止按 8 位单位写入。

产生复位信号后，SDRmn 寄存器被清除为 0000H。

注 1. 仅限以下 UART 可以设定 9 位的数据长度。

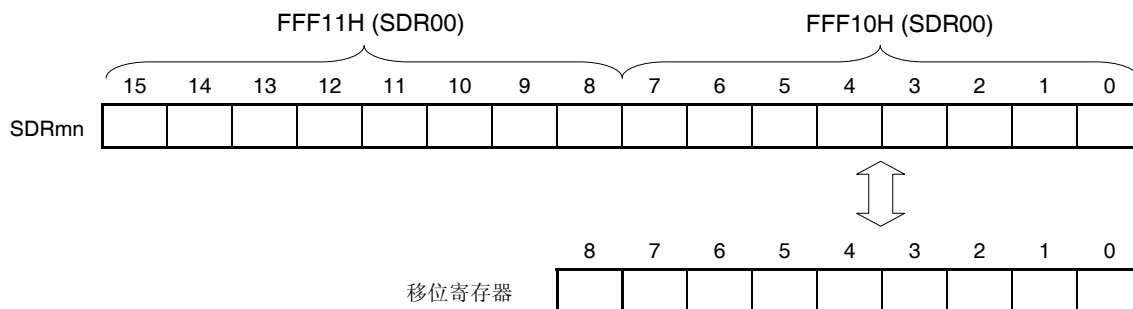
2. 停止操作 (SEmn = 0) 时，禁止以 8 位为单位进行写入。

备注 1. 数据接收完成后，0 将存储于位 0 至位 8 中超过数据长度部分的位。

2. m: 单元编号 (m = 0), n: 通道编号 (n = 0 至 3), p: CSI 编号 (p = 00), q: UART 编号 (q = 0, 1)

图 15-2. 串行数据寄存器 mn (SDRmn)的格式 (mn = 00, 01)

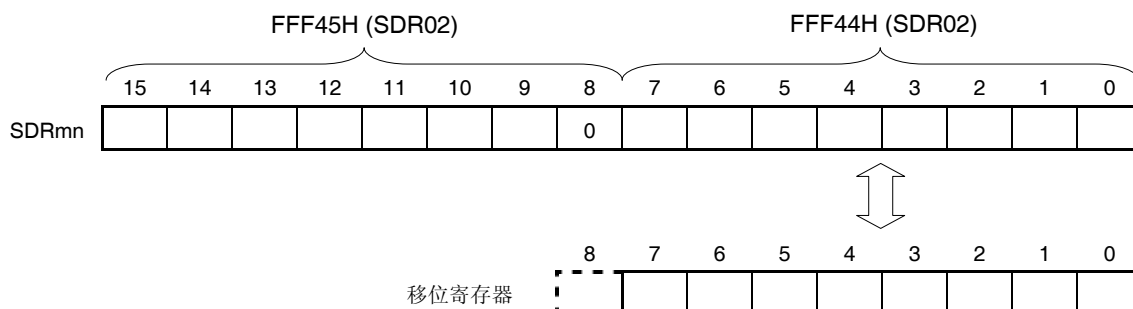
地址: FFF10H, FFF11H (SDR00), FFF12H, FFF13H (SDR01) 复位后: 0000H R/W



备注 关于 SDRmn 寄存器的高 7 位功能, 请参阅 15.3 控制串行阵列单元 0 的寄存器。

图 15-3. 串行数据寄存器 mn (SDRmn)的格式 (mn = 02, 03)

地址: FFF44H, FFF45H (SDR02), FFF46H, FFF47H (SDR03), 复位后: 0000H R/W



注意事项 必须将位 8 清除为 0。

备注 关于 SDRmn 寄存器的高 7 位功能, 请参阅 15.3 控制串行阵列单元 0 的寄存器。

15.3 控制串行阵列单元 0 的寄存器

串行阵列单元 0 由下列寄存器控制。

- 外围允许寄存器 0 (PER0)
- 串行时钟选择寄存器 m (SPSm)
- 串行模式寄存器 mn (SMRmn)
- 串行通信操作设置寄存器 mn (SCRmn)
- 串行数据寄存器 mn (SDRmn)
- 串行标志清除触发寄存器 mn (SIRmn)
- 串行状态寄存器 mn (SSRmn)
- 串行通道开始寄存器 m (SSm)
- 串行通道停止寄存器 m (STm)
- 串行通道允许状态寄存器 m (SEm)
- 串行输出允许寄存器 m (SOEm)
- 串行输出电平寄存器 m (SOLm)
- 串行输出寄存器 m (SOM)
- 串行待机控制寄存器 0 (SSC0)
- 输入切换控制寄存器 (ISC)
- 噪声滤波器允许寄存器 0 (NFEN0)
- 端口输入模式寄存器 0, 1 (PIM0, PIM1)
- 端口输出模式寄存器 0, 1 (POM0, POM1)
- 端口模式寄存器 0, 1 (PM0, PM1)
- 端口寄存器 0, 1 (P0, P1)

备注 m: 单元编号(m = 0), n: 通道编号(n = 0 至 3)

(1) 外围允许寄存器 0 (PER0)

PER0 用于允许或禁止向外围硬件供应时钟。通过停止向未使用的硬件供应时钟，可以降低功耗和噪声。

当使用串行阵列单元 0 时，必须将该寄存器的位 2 (SAU0EN) 设置为 1。

使用 1 位或 8 位存储器操作指令设置 PER0 寄存器。

产生复位信号后，PER0 寄存器被清除为 00H。

图 15-4. 外围允许寄存器 0 (PER0) 的格式

地址: F00F0H 复位后: 00H R/W

符号	<7>	6	<5>	<4>	<3>	<2>	1	<0>
PER0	RTCEN	0	ADCEN	IICA0EN	0	SAU0EN	0	TAU0EN

SAU0EN	串行阵列单元0输入时钟供应的控制
0	停止输入时钟的供应。 <ul style="list-style-type: none"> 不可写入用于串行阵列单元0的SFR。 串行阵列单元0处于复位状态。
1	允许输入时钟供应。 <ul style="list-style-type: none"> 可以读取和写入用于串行阵列单元0的SFR。

- 注意事项 1.** 在设置串行阵列单元 0 时，必须先将 SAUmEN 位设置为 1。如果 SAUmEN = 0，对串行阵列单元 0 的控制寄存器进行的写入操作被忽略，而且该寄存器的读取值总为初始值（输入切换控制寄存器 (ISC)、噪声滤波器允许寄存器 0 (NFEN0)、端口输入模式寄存器 0、1 (PIM0、PIM1)、端口输出模式寄存器 0、1 (POM0、POM1)、端口模式寄存器 0、1 (PM0、PM1) 以及端口寄存器 0、1 (P0、P1) 除外）。
- 2.** 在将 SAU0EN 位设置为 1 后，需经过 fCLK 的 4 个或更多时钟后才可以设置串行时钟选择寄存器 m (SPSm)。
- 3.** 必须将位 1、位 3 和位 6 清除为 0。

(2) 串行时钟选择寄存器 m (SPSm)

SPSm 寄存器为 16 位寄存器，用于选择供应给各个通道共通的两种操作时钟(CKm0、CKm1)。通过 SPSm 寄存器的位 7 至位 4 选择 CKm1，位 3 至位 0 选择 CKm0。

当寄存器在运行 ($SEmn = 1$) 时，禁止改写 SPSm 寄存器。

用 16 位存储器操作指令设置 SPSm 寄存器。

通过 SPSmL 使用 8 位存储器操作指令设置 SPSm 寄存器的低 8 位。

产生复位信号后，SPSm 寄存器被清除为 0000H。

图 15-5. 串行时钟选择寄存器 m (SPSm)的格式

地址: F0126H, F0127H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPSm	0	0	0	0	0	0	0	0	PRS m13	PRS m12	PRS m11	PRS m10	PRS m03	PRS m02	PRS m01	PRS m00

PRS mk3	PRS mk2	PRS mk1	PRS mk0		操作时钟(CKmk)的选择 ^{注1}				
					f _{CLK} = 2 MHz	f _{CLK} = 5 MHz	f _{CLK} = 10 MHz	f _{CLK} = 20 MHz	f _{CLK} = 32 MHz
0	0	0	0	f _{CLK}	2 MHz	5 MHz	10 MHz	20 MHz	32 MHz
0	0	0	1	f _{CLK} /2	1 MHz	2.5 MHz	5 MHz	10 MHz	16 MHz
0	0	1	0	f _{CLK} /2 ²	500 kHz	1.25 MHz	2.5 MHz	5 MHz	8 MHz
0	0	1	1	f _{CLK} /2 ³	250 kHz	625 kHz	1.25 MHz	2.5 MHz	4 MHz
0	1	0	0	f _{CLK} /2 ⁴	125 kHz	313 kHz	625 kHz	1.25 MHz	2 MHz
0	1	0	1	f _{CLK} /2 ⁵	62.5 kHz	156 kHz	313 kHz	625 kHz	1 MHz
0	1	1	0	f _{CLK} /2 ⁶	31.3 kHz	78.1 kHz	156 kHz	313 kHz	500 kHz
0	1	1	1	f _{CLK} /2 ⁷	15.6 kHz	39.1 kHz	78.1 kHz	156 kHz	250 kHz
1	0	0	0	f _{CLK} /2 ⁸	7.81 kHz	19.5 kHz	39.1 kHz	78.1 kHz	125 kHz
1	0	0	1	f _{CLK} /2 ⁹	3.91 kHz	9.77 kHz	19.5 kHz	39.1 kHz	62.5 kHz
1	0	1	0	f _{CLK} /2 ¹⁰	1.95 kHz	4.88 kHz	9.77 kHz	19.5 kHz	31.3 kHz
1	0	1	1	f _{CLK} /2 ¹¹	977 Hz	2.44 kHz	4.88 kHz	9.77 kHz	15.6 kHz
1	1	0	0	f _{CLK} /2 ¹²	488 Hz	1.22 kHz	2.44 kHz	4.88 kHz	7.81 kHz
1	1	0	1	f _{CLK} /2 ¹³	244 Hz	610 Hz	1.22 kHz	2.44 kHz	3.91 kHz
1	1	1	0	f _{CLK} /2 ¹⁴	122 Hz	305 Hz	610 Hz	1.22 kHz	1.95 kHz
1	1	1	1	f _{CLK} /2 ¹⁵	61 Hz	153 kHz	305 Hz	610 Hz	977 Hz

注 要在串行阵列单元(SAU)操作中改变 f_{CLK} 的时钟选择(通过更改系统时钟控制寄存器(CKC)的值), 必须在停止串行阵列单元 0 (SAU0)操作(串行通道停止寄存器 m (STm) = 000FH)之后才可以执行。

注意事项 必须将位 15 至 8 清除为 0。

- 备注 1. f_{CLK}: CPU/外围硬件时钟频率
f_{SUB}: 副系统时钟频率
2. m: 单元编号(m = 0)
3. k = 0, 1

(3) 串行模式寄存器 mn (SMRmn)

SMRmn 寄存器用于设置通道 n 的操作模式。也用于选择操作时钟(f_{mck})，指定是否可以输入串行时钟(f_{sck})，设置开始触发、操作模式 (CSI 或 UART) 和中断源。该寄存器还用于仅在 UART 模式下反相接收数据的电平。禁止向操作中 (SE_{mn} = 1 时) 的 SMRmn 寄存器进行改写。但是，可以在操作过程中改写 MD_{mn0} 位。用 16 位存储器操作指令设置 SMRmn 寄存器。产生复位信号后，SMRmn 寄存器被设置为 0020H。

图 15-6. 串行模式寄存器 mn (SMRmn)的格式(1/2)

地址: F0110H, F0111H (SMR00)至F0116H, F0117H (SMR03), 复位后: 0020H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMRmn	CKS mn	CCS mn	0	0	0	0	0	STS mn [#]	0	SIS mn0 [#]	1	0	0	0	MD mn1	MD mn0

CKS mn	通道n操作时钟(f _{mck})的选择
0	由SPSm寄存器设置的操作时钟CKm0
1	由SPSm寄存器设置的操作时钟CKm1
边沿检测电路所使用的操作时钟(f _{mck})。外，根据SDRmn寄存器的CCSmn位和高7位的设置，产生传送时钟(f _{trclk})。	

CCS mn	通道n的传送时钟(f _{trclk})的选择
0	由CKSmn位指定的分频操作时钟f _{mck}
1	来自SCKp引脚的输入时钟 f _{sck} (CSI模式的从传送)
传送时钟f _{trclk} 用于移位寄存器、通信控制电路、输出控制器、中断控制电路和错误控制电路。当CCSmn = 0时，使用SDRmn寄存器的高7位设置操作时钟(f _{mck})的分频比。	

STS mn [#]	开始触发源的选择
0	仅限软件触发有效 (CSI和UART发送时选择)。
1	RxDq引脚的有效沿 (UART接收时选择)。
将SSm寄存器设置为1并且满足以上条件时，开始传送。	

注 仅限 SMR01 和 SMR03 寄存器。

注意事项 必须将位 13 至 9、7、4 和 3 (或者 SMR00 或 SMR02 寄存器的位 13 至 6、4 和 3)清除为 0。必须将位 5 设置为 1。

备注 m: 单元编号(m = 0), n: 通道编号(n = 0 至 3), p: CSI 编号(p = 00), q: UART 编号(q = 0, 1)

图 15-6. 串行模式寄存器 mn (SMRmn)的格式(2/2)

地址: F0110H, F0111H (SMR00)至F0116H, F0117H (SMR03), 复位后: 0020H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMRmn	CKS mn	CCS mn	0	0	0	0	0	STS mn [‡]	0	SIS mn0 [‡]	1	0	0	0	MD mn1	MD mn0

SIS mn0	UART模式下, 通道n接收数据的电平反相的控制
0	下降沿被检测为起始位。 按原样捕捉被输入的通信数据。
1	上升沿被检测为起始位。 将被输入的通信数据反相并捕捉。

MD mn1	通道n操作模式的设置
0	CSI模式
1	UART模式

MD mn0	通道n中断源的选择
0	传送结束中断
1	缓冲器空中断 (当数据从SDRmn寄存器传送至移位寄存器时。)
连续发送时, 将MDmn0位设置为1, 并且在SDRmn的数据为空时, 写入下一个发送数据。	

注 仅限 SMR01 和 SMR03 寄存器。

注意事项 必须将位 13 至 9、7、4 和 3 (或者 SMR00 或 SMR02 寄存器的位 13 至 6、4 和 3)清除为 0。必须将位 5 设置为 1。

备注 m: 单元编号(m = 0), n: 通道编号(n = 0 至 3), p: CSI 编号(p = 00), q: UART 编号(q = 0, 1)

(4) 串行通信操作设置寄存器 mn (SCRmn)

SCRmn 寄存器是通道 n 的通信操作设置寄存器。该寄存器用于设置数据发送/接收模式、数据和时钟相位、是否屏蔽错误信号、奇偶检验位、起始位、停止位和数据长度。

当寄存器在运行时(SEmn = 1 时), 禁止改写 SCRmn 寄存器。

用 16 位存储器操作指令设置 SCRmn 寄存器。

产生复位信号后, SCRmn 寄存器被设置为 0087H。

图 15-7. 串行通信操作设置寄存器 mn (SCRmn)的格式 (1/2)

地址: F0118H, F0119H (SCR00)至F011EH, F011FH (SCR03), 复位后: 0087H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCRmn	TxE mn	RxE mn	DAP mn	CKP mn	0	EOC mn	PTC mn1	PTC mn0	DIR mn	0	SLCm n1 ^{#1}	SLC mn0	0	1	DLSm n1 ^{#2}	DLS mn0

TxE mn	RxE mn	通道n操作模式的设置
0	0	禁止通信
0	1	仅接收
1	0	仅发送
1	1	发送/接收

DAP mn	CKP mn	CSI模式下数据和时钟相位的选择	类型
0	0		1
0	1		2
1	0		3
1	1		4

UART模式下必须设置为DAPmn、CKPmn = 0、0。

EOC mn	选择是否屏蔽错误中断信号(INTSREx (x = 0至3))
0	屏蔽错误中断INTSREx (不屏蔽INTSRx)。
1	允许产生错误中断INTSREx (如果出现错误, INTSRx将被屏蔽)。

CSI模式、UART发送期间设置为EOCmn = 0^{#3}。

- 注 1. 仅限 SCR00 和 SCR02 寄存器。
- 2. 仅限 SCR00 和 SCR01 寄存器。
- 3. EOCmn 不为 0 时使用 CSImn 的情况下, 有可能产生错误中断 INTSREn。

注意事项 必须将位 3、6 和 11 清除为 0。(将 SCR01 或 SCR03 寄存器的位 5 清除为 0。)必须将位 2 设置为 1。(将 SCR02 或 SCR03 寄存器的位 1 设置为 0。)

备注 m: 单元编号(m = 0), n: 通道编号(n = 0 至 3), p: CSI 编号(p = 00)

图 15-7. 串行通信操作设置寄存器 mn (SCRmn)的格式 (2/2)

地址: F0118H, F0119H (SCR00)至F011EH, F011FH (SCR03), 复位后: 0087H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCRmn	TxE mn	RxE mn	DAP mn	CKP mn	0	EOC mn	PTC mn1	PTC mn0	DIR mn	0	SLCm n1 ^{#1}	SLC mn0	0	1	DLSm n1 ^{#2}	DLS mn0

PTC mn1	PTC mn0	UART模式下奇偶校验位的设置	
		发送	接收
0	0	不输出奇偶校验位	无奇偶校验的接收
0	1	输出零奇偶校验 ^{#3}	无奇偶校验判断
1	0	输出偶数校验	偶数校验判断
1	1	输出奇数校验	奇数校验判断

CSI模式下必须设置为PTCmn1、PTCmn0 = 0、0。

DIR mn	CSI和UART模式下数据传送序列的选择	
0	MSB优先的输入/输出数据	
1	LSB优先的输入/输出数据	

SLCm n1 ^{#1}	SLC mn0	UART模式下停止位的设置	
0	0	无停止位	
0	1	停止位长度 = 1位	
1	0	停止位长度 = 2位 (仅限mn = 00, 02)	
1	1	禁止设置	

选择传送结束中断时, 传送完所有停止位时产生中断。
 UART接收时设置为1位(SLCmn1、SLCmn0 = 0、1)。
 CSI模式时设置为无停止位(SLCmn1、SLCmn0 = 0、0)。

DLSm n1 ^{#2}	DLS mn0	CSI和UART模式下数据长度的设置	
0	1	9位数据长度 (存储于SDRmn寄存器的位0至位8) (仅限UART模式下可设置)	
1	0	7位数据长度 (存储于SDRmn寄存器的位0至位6)	
1	1	8位数据长度 (存储于SDRmn寄存器的位0至位7)	
其他		禁止设置	

- 注 1. 仅限 SCR00 和 SCR02 寄存器。
 2. 仅限 SCR00 和 SCR01 寄存器。
 3. 不论数据内容如何, 始终附加 0。

注意事项 必须将位 3、6 和 11 清除为 0。(将 SCR01 和 SCR03 寄存器的位 5 也清除为 0。)必须将位 2 设置为 1。
 (将 SCR02 和 SCR03 寄存器的位 1 也设置为 0。)

备注 m: 单元编号(m = 0), n: 通道编号(n = 0 至 3), p: CSI 编号(p = 00)

(5) 串行数据寄存器 mn (SDRmn)的高 7 位

SDRmn 寄存器为通道 n 的发送/接收数据寄存器（16 位）。SDR00 和 SDR01 的位 8 至位 0（低 9 位）或 SDR02 和 SDR03 的位 7 至位 0（低 8 位）作为发送/接收缓冲寄存器，位 15 至位 9 作为设置操作时钟(fmck, fscck)分频比的寄存器。

将串行模式寄存器 mn (SMRmn)的 CCSmn 位清除为 0 时，通过 SDRmn 寄存器的高 7 位设置工作时钟的分频率作为传送时钟。

SDRmn 寄存器的低 8/9 位用作发送/接收缓冲寄存器。接收过程中，由移位寄存器转换的并行数据将存储于低 8/9 位中，在发送过程中，发送至移位寄存器的数据将被设置到低 8/9 位。

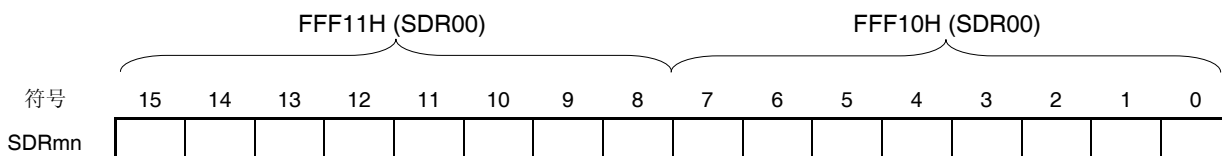
使用 16 位存储器操作指令设置 SDRmn 寄存器。

然而，仅当停止操作(SEmn = 0)时，才可写入或读取高 7 位。操作过程中 (SEmn = 1)，仅向 SDRmn 寄存器的低 8/9 位写入值。操作过程中 SDRmn 寄存器的读取值始终为 0。

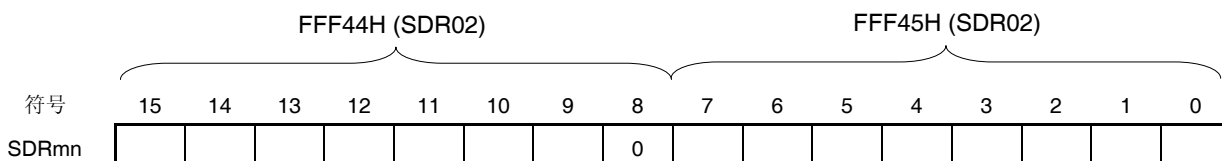
产生复位信号后，SDRmn 寄存器被清除为 0000H。

图 15-8. 串行数据寄存器 mn (SDRmn)的格式

地址： FFF10H, FFF11H (SDR00), FFF12H, FFF13H (SDR01) 复位后： 0000H R/W



地址： FFF44H, FFF45H (SDR02), FFF46H, FFF47H (SDR03), 复位后： 0000H R/W



SDRmn[15:9]							操作时钟(fmck)的分频率的设置
0	0	0	0	0	0	0	f _{mck} /2, f _{scck} /2 (在CSI从传送模式时)
0	0	0	0	0	0	1	f _{mck} /4
0	0	0	0	0	1	0	f _{mck} /6
0	0	0	0	0	1	1	f _{mck} /8
•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•
1	1	1	1	1	1	0	f _{mck} /254
1	1	1	1	1	1	1	f _{mck} /256

(注意事项和备注如下页所示。)

- 注意事项**
1. 必须将 **SDR02** 和 **SDR03** 的位 8 清除为 0。
 2. 使用 **UART** 时，禁止设置为 **SDRmn[15:9] = (0000000B, 0000001B, 0000010B)**。
 3. 停止操作 (**SEmn = 0**)时，不要向低 8 位执行 8 位写入。（如果写入这些位，高 7 位将被清除为 0。）
- 备注**
1. 关于 **SDRmn** 寄存器的低 8/9 位功能，请参阅 **15.2 串行阵列单元 0 的配置**。
 2. m: 单元编号(m = 0), n: 通道编号(n = 0 至 3)

(6) 串行标志清除触发寄存器 mn (SIRmn)

SIRmn 寄存器是用于清除通道 n 各个错误标志的触发寄存器。

当该寄存器的各位 (FECTmn、PECTmn、OVCTmn) 被设置为 1 时, 串行状态寄存器 mn 的相应位 (FEFmn、PEFmn、OVFmn) 将被清除为 0。由于 SIRmn 寄存器是一种触发寄存器, 当 SSRmn 寄存器的相应位被清零时, 它们会被立即清零。

用 16 位存储器操作指令设置 SIRmn 寄存器。

通过 SIRmnL 使用 8 位存储器操作指令设置 SIRmn 寄存器的低 8 位。

产生复位信号后, SIRmn 寄存器被清除为 0000H。

图 15-9. 串行标志清除触发寄存器 mn (SIRmn)的格式

地址: F0108H, F0109H (SIR00)至F010EH, F010FH (SIR03), 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SIRmn	0	0	0	0	0	0	0	0	0	0	0	0	0	FECT mn ^注	PEC Tmn	OVC Tmn

FEC Tmn	通道n的帧错误标志的清除触发														
0	不清除														
1	将SSRmn寄存器的FEFmn位清除为0。														

PEC Tmn	通道n的奇偶校验错误标志的清除触发														
0	不清除														
1	将SSRmn寄存器的PEFmn位清除为0。														

OVC Tmn	通道n的溢出错误标志的清除触发														
0	不清除														
1	将SSRmn寄存器的OVFmn位清除为0。														

注 仅限 SIR01 和 SIR03 寄存器。

注意事项 必须将位 15 至 3 (或者 SIR00 或 SIR02 寄存器的位 15 至 2)清除为 0。

- 备注 1. m: 单元编号(m = 0), n: 通道编号(n = 0 至 3)
 2. SIRmn 寄存器的读取值总为 0000H。

(7) 串行状态寄存器 mn (SSRmn)

SSRmn 寄存器用于指示通道 n 的通信状态和错误发生状态。该寄存器指示的错误为帧错误、奇偶校验错误和溢出错误。

使用 16 位存储器操作指令读取 SSRmn 寄存器。

通过 SSRmnL 使用 8 位存储器操作指令设置 SSRmn 寄存器的低 8 位。

产生复位信号后，SSRmn 寄存器被清除为 0000H。

图 15-10. 串行状态寄存器 mn (SSRmn)的格式(1/2)

地址: F0100H, F0101H (SSR00)至F0106H, F0107H (SSR03), 复位后: 0000H R

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSRmn	0	0	0	0	0	0	0	0	0	TSF mn	BFF mn	0	0	FEFm n [#]	PEF mn	OVF mn

TSF mn	通道n的通信状态显示标志
0	通信停止或暂停。
1	通信正在进行。
<清除条件>	
<ul style="list-style-type: none"> • STm寄存器的STmn位被设置为1 (通信停止)或者SSm寄存器的SSmn位被设置为1(通信暂停)。 • 通信结束。 	
<设置条件>	
<ul style="list-style-type: none"> • 通信开始。 	

BFF mn	通道n的缓冲寄存器状态显示标志
0	SDRmn寄存器中不存储有效数据。
1	SDRmn寄存器中存储有效数据。
<清除条件>	
<ul style="list-style-type: none"> • 发送时，从SDRmn寄存器至移位寄存器的发送数据的传送结束。 • 接收时，来自SDRmn寄存器的接收数据的读取结束。 • STm寄存器的STmn位被设置为1 (通信停止)或者SSm寄存器的SSmn位被设置为1 (允许通信)。 	
<设置条件>	
<ul style="list-style-type: none"> • 当SCRmn寄存器的TxEmn位被设置为1 (各种通信模式下的发送或者发送和接收模式) 时，发送数据被写入 SDRmn寄存器。 • 当SCRmn寄存器的RxEmn位被设置为1 (各种通信模式下的发送或者发送和接收模式) 时，接收数据被存储于 SDRmn寄存器中。 • 发生接收错误。 	

注 仅限 SSR01 和 SSR03 寄存器。

注意事项 如果在 BFFmn = 1 时将数据写入 SDRmn 寄存器，存储于该寄存器中的发送/接收数据将被丢弃，并检测到溢出错误(OVEmn = 1)。

备注 m: 单元编号(m = 0), n: 通道编号(n = 0 至 3)

图 15-10. 串行状态寄存器 mn (SSRmn)的格式(2/2)

地址: F0100H, F0101H (SSR00)至F0106H, F0107H (SSR03), 复位后: 0000H R

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSRmn	0	0	0	0	0	0	0	0	0	TSMn	BFFmn	0	0	FEFmn ^注	PEFmn	OVFmn

FEFmn ^注	通道n的帧错误检测标志
0	无错误发生。
1	错误发生 (UART接收期间)。
<清除条件> <ul style="list-style-type: none"> • 将1写入SIRmn寄存器的FECTmn位。 <设置条件> <ul style="list-style-type: none"> • 当UART接收结束时未检测到停止位。 	

PEFmn	通道n的奇偶校验错误检测标志
0	无错误发生。
1	奇偶校验错误发生 (UART接收期间)。
<清除条件> <ul style="list-style-type: none"> • 将1写入SIRmn寄存器的PECTmn位。 <设置条件> <ul style="list-style-type: none"> • 当UART接收结束时, 发送数据的奇偶校验和奇偶校验位不匹配(奇偶校验错误)。 	

OVFmn	通道n的溢出错误检测标志
0	无错误发生。
1	错误发生。
<清除条件> <ul style="list-style-type: none"> • 将1写入SIRmn寄存器的OVCTmn位。 <设置条件> <ul style="list-style-type: none"> • 当SCRmn寄存器的RxEmn位被设置为1 (各种通信模式下的接收或者发送和接收模式) 时, 即使接收数据存储于SDRmn寄存器中, 也不读取这些数据, 而是写入发送数据或写入下一接收数据。 • CSI模式的从发送或者发送和接收时, 尚未准备好发送数据。 	

注 仅限 SSR01 和 SSR03 寄存器。

备注 m: 单元编号(m = 0), n: 通道编号(n = 0 至 3)

(8) 串行通道开始寄存器 m (SSm)

SSm 寄存器是对各个通道进行设置以允许开始通信/计数的触发寄存器。

当将 1 写入该寄存器(SSmn)的一位时，串行通道允许状态寄存器 m (SEmn) 的对应位(SEmn) 将被设置为 1（允许操作）。由于 SSmn 位为触发位，所以当 SEmn = 1 时，该位将立即被清除。

用 16 位存储器操作指令设置 SSm 寄存器。

通过 SSmL 使用 1 位或 8 位存储器操作指令设置 SSm 寄存器的低 8 位。

产生复位信号后，SSm 寄存器被清除为 0000H。

图 15-11. 串行通道开始寄存器 m (SSm)的格式

地址: F0122H, F0123H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSm	0	0	0	0	0	0	0	0	0	0	0	0	SSm	SSm	SSm	SSm
													3	2	1	0

SSmn	通道n的操作开始触发
0	无触发操作。
1	将SEmn位设置为1，进入通信等待状态 [※] 。

注 如果在通信操作中设置 SSmn = 1，则会停止通信进入等待状态。
 此时保持控制寄存器和移位寄存器、SCKmn 和 SOmn 引脚、以及 FEFmn、PEFmn、OVFmn 标志的状态值。

- 注意事项**
1. 必须将位 15 至 4 清除为 0。
 2. 对于 UART 接收，将 SCRmn 寄存器的 RXEmn 位设置为 1，然后确保在经过 4 个或更多 fmck 时钟之后将 SSmn 设置为 1。

- 备注**
1. m: 单元编号(m = 0), n: 通道编号(n = 0 至 3)
 2. SSm 寄存器的读取值总为 0000H。

(9) 串行通道停止寄存器 m (STm)

STm 寄存器是对各个通道进行设置以允许停止通信/计数的触发寄存器。

当将 1 写入该寄存器(SSmn)的一位时，串行通道允许状态寄存器 m (SEm)的对应位(SEmn) 将被清除为 0（停止操作）。由于 STmn 位为触发位，所以当 SEmn = 0 时，该位将立即被清除为 0。

用 16 位存储器操作指令写入 STm 寄存器。

通过 STmL 使用 1 位或 8 位存储器操作指令设置 STm 寄存器的低 8 位。

产生复位信号后，STm 寄存器被清除为 0000H。

图 15-12. 串行通道停止寄存器 m (STm)的格式

地址: F0124H, F0125H 复位后: 0000H W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
STm	0	0	0	0	0	0	0	0	0	0	0	0	STm 3	STm 2	STm 1	STm 0

STm n	通道n的操作停止触发
0	无触发操作
1	将SEmn位清除为0，停止通信操作 [※]

注 保持控制寄存器和移位寄存器，SCKmn 和 SOMn 引脚，以及 FEFmn、PEFmn、OVFmn 标志的状态值。

注意事项 必须将位 15 至 4 清除为 0。

备注 1. m: 单元编号(m = 0), n: 通道编号(n = 0 至 3)

2. STm 寄存器的读取值总为 0000H。

(10) 串行通道允许状态寄存器 m (SEm)

SEm 寄存器用于确认各个通道的串行发送/接收操作处于允许或停止状态。

当将 1 写入串行通道开始寄存器 m (SSm) 的一个位时，该寄存器的对应位将被设置为 1。当将 1 写入串行通道停止寄存器 m (STm) 的一个位时，对应位将被清除为 0。

对于允许操作的通道 n，不能通过软件改写串行输出寄存器 m (SOM)（描述见后文）的 CKOmn 位的值（通道 n 的串行时钟输出），通信操作反映的值由串行时钟引脚输出。

对于停止操作的通道 n，可以通过软件设置 SOM 寄存器的 CKOmn 位的值，并从串行时钟引脚输出该值。这样，任何波形（如开始条件/停止条件的波形）均可由软件创建。

使用 16 位存储器操作指令读取 SEm 寄存器。

通过 SEmL 使用 1 位或 8 位存储器操作指令设置 SEm 寄存器的低 8 位。

产生复位信号后，SEm 寄存器被清除为 0000H。

图 15-13. 串行通道允许状态寄存器 m (SEm) 的格式

地址: F0120H, F0121H 复位后: 0000H R

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SEm	0	0	0	0	0	0	0	0	0	0	0	0	SEm 3	SEm 2	SEm 1	SEm 0
SEmn	表示通道n的操作允许或停止状态															
0	停止操作。															
1	允许操作。															

备注 m: 单元编号(m = 0), n: 通道编号(n = 0 至 3)

(11) 串行输出允许寄存器 m (SOEm)

SOEm 寄存器用于设置是否允许或停止各个通道的串行通信操作的输出。

对于允许输出的通道 n，将不能通过软件改写串行输出寄存器 m (SOm)（描述见后文）的 SOmn 位的值，通信操作反映的值由串行时钟引脚输出。

对于停止输出的通道 n，可以通过软件设置 SOm 寄存器的 SOmn 位的值，并从串行数据输出引脚输出该值。这样，任何波形（如开始条件/停止条件的波形）均可由软件创建。

用 16 位存储器操作指令设置 SOEm 寄存器。

通过 SOEmL 使用 1 位或 8 位存储器操作指令设置 SOEm 寄存器的低 8 位。

产生复位信号后，SOEm 寄存器被清除为 0000H。

图 15-14. 串行输出允许寄存器 m (SOEm)的格式

地址: F012AH, F012BH 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOEm	0	0	0	0	0	0	0	0	0	0	0	0	0	SOE m2	0	SOE m0

SOE mn	通道n的串行输出允许/停止
0	停止串行通信操作输出。
1	允许串行通信操作输出。

注意事项 必须将位 15 至 3 以及 1 清除为 0。

备注 m: 单元编号(m = 0), n: 通道编号(n = 0, 2)

(12) 串行输出寄存器 m (SOM)

SOM 寄存器是用于各个通道串行输出的缓冲寄存器。

该寄存器的 SOMn 位的值从通道 n 的串行数据输出引脚输出。

该寄存器的 CKOm_n 位的值从通道 n 的串行时钟输出引脚输出。

该寄存器的 SOMn 位只有在串行输出被禁止时 (SOEm_n = 0) 才可用软件改写。当允许串行输出(SOEm_n = 1)时，用软件改写将被忽略，且该寄存器的值只能通过串行通信操作更改。

该寄存器的 CKOm_n 位只有在停止通道操作时 (SEm_n = 0) 才可用软件改写。当允许通道操作(SEm_n = 1)时，用软件改写将被忽略，且该 CKOm_n 位的值只能通过串行通信操作更改。

要将串行接口引脚用作端口功能引脚，须将相应的 CKOm_n 和 SOMn 位设置为 1。

用 16 位存储器操作指令设置 SOM 寄存器。

产生复位信号后，SOM 寄存器被清除为 0F0FH。

图 15-15. 串行输出寄存器 m (SOM)的格式

地址: F0128H, F0129H 复位后: 0F0FH R/W

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
符号																
SOM	0	0	0	0	1	CKO m ₂	1	CKO m ₀	0	0	0	0	1	SO m ₂	1	SO m ₀

CKO m _n	通道n的串行时钟输出
0	串行时钟输出值为0
1	串行时钟输出值为1

SO m _n	通道n的串行数据输出
0	串行数据输出值为0
1	串行数据输出值为1

注意事项 必须将位 11、9、3 和 1 设置为 1。必须将位 15 至 12 和 7 至 4 清除为 0。

备注 m: 单元编号(m = 0), n: 通道编号(n = 0, 2)

(13) 串行输出电平寄存器 m (SOLm)

SOLm 寄存器是用于设置各个通道数据输出电平反相的寄存器。

仅限 UART 模式时可以设置该寄存器。CSI 模式时，必须将对应位设置为 0。

仅当允许串行输出时 (SOEmn = 1)，通过该寄存器对通道 n 所作的反相设置才会体现在引脚输出上。当串行输出被禁止时 (SOEmn = 0)，SOMn 位的值将按原样输出。

当寄存器在运行 (SEmn = 1) 时，禁止改写 SOLm 寄存器。

用 16 位存储器操作指令设置 SOLm 寄存器。

通过 SOLmL 使用 8 位存储器操作指令设置 SOLm 寄存器的低 8 位。

产生复位信号后，SOLm 寄存器被清除为 0000H。

图 15-16. 串行输出电平寄存器 m (SOLm)的格式

地址: F0134H, F0135H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOLm	0	0	0	0	0	0	0	0	0	0	0	0	0	SOL m2	0	SOL m0

SOL mn	UART模式时，通道n的发送数据电平反相的选择
0	通信数据按原样输出。
1	通信数据反相并输出。

注意事项 必须将位 15 至 3 以及 1 清除为 0。

备注 m: 单元编号(m = 0), n: 通道编号(n = 0, 2)

(14) 串行待机控制寄存器 0 (SSC0)

SSC0 寄存器可通过接收 CSI00 或 UART0 的串行数据，来控制在 STOP 模式下启动接收（SNOOZE 模式）。

使用 16 位存储器操作指令设置 SSC0 寄存器。

通过 SSC0L 使用 8 位存储器操作指令设置 SSC0 寄存器的低 8 位。

产生复位信号后，SSC0 寄存器被清除为 0000H。

注意事项 SNOOZE 模式下的最大传送速率如下所示。

- 使用 CSI00 时 : 1 Mbps
- 使用 UART0 时 : 9600 bps

图 15-17. 串行待机控制寄存器 0 (SSC0)的格式

地址: F0138H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSC0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SS ECO	SW CO

SS ECO	允许或停止产生传送结束中断的选择
-----------	------------------

0	允许产生错误中断(INTSRE0)。以下情况下，时钟发生器的时钟请求信号（内部信号）也将被清除： <ul style="list-style-type: none"> • SWC0位被清除为0 • 当误测到UART接收起始位时
---	--

1	停止产生错误中断(INTSRE0)。以下情况下，时钟发生器的时钟请求信号（内部信号）也将被清除： <ul style="list-style-type: none"> • SWC0位被清除为0 • 当误测到UART接收起始位时 • 由于奇偶检验错误或帧错误而产生传送结束中断的时序
---	--

SW CO	SNOOZE模式的设置
----------	-------------

0	不使用SNOOZE模式功能。
---	----------------

1	使用SNOOZE模式功能。
---	---------------

<ul style="list-style-type: none"> • 由于STOP模式下的硬件触发信号，退出STOP模式，并且在不操作CPU的情况下执行CSI/UART接收（SNOOZE模式）。 • 仅在选择高速片上振荡器时钟作为CPU/外围硬件时钟(f_{CLK})时可以使用SNOOZE模式功能。如果选用任何其他时钟，则禁止使用该模式。 • 使用SNOOZE模式时，请在正常操作模式中设置SWC0位为0，并且在切换至STOP模式之前设置AWC位为1。此外，从STOP模式恢复到正常操作模式前，请清除SWC0位为0。 	
--	--

注意事项 禁止设置 SSEC0, SWC0 = 1, 0。

(15) 输入切换控制寄存器(ISC)

通过将 ISC 寄存器的 ISC1 和 ISC0 位与外部中断和定时器阵列单元一起使用，从而实现 UART0 时的 LIN-bus 和 DMX512 通信操作。

当位 0 被设置为 1 时，串行数据输入(RxD0)引脚的输入信号将被选为外部中断输入(INTP0)，可用于检测唤醒信号和 BREAK 信号。

当位 1 被设置为 1 时，串行数据输入(RxD0)引脚的输入信号将被选为定时器输入，以便能检测到唤醒信号，并且定时器能够测量间隔段的低宽度、同步段的脉冲宽度和 DMX512 通信的输入信号宽度。

使用 1 位或 8 位存储器操作指令设置 ISC 寄存器。

产生复位信号后，ISC 寄存器被清除为 00H。

图 15-18. 输入切换控制寄存器(ISC)的格式

地址: F0073H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ISC	0	0	0	0	0	0	ISC1	ISC0

ISC1	切换定时器阵列单元通道7的输入
0	不要在通道7使用定时器输入信号。
1	RxD0引脚的输入信号用作定时器输入(LIN-bus: 检测唤醒信号，测量间隔段的低电平宽度和同步段的脉冲宽度；DMX512: 测量输入信号宽度)。

ISC0	切换外部中断(INTP0)输入
0	将INTP0引脚的输入信号用作外部中断的输入(正常操作)。
1	将RxD0引脚的输入信号用作外部中断的输入 (LIN-bus: 唤醒信号检测；DMX512: BREAK信号检测)。

注意事项 必须将位 7 至 2 清除为 0。

(16) 噪声滤波器允许寄存器 0 (NFEN0)

NFEN0 寄存器用来设置噪声滤波器是否可用于各通道串行数据输入引脚的输入信号。

用于 CSI 通信的引脚中，将该寄存器的对应位清除为 0，禁止使用噪声滤波器。

用于 UART 通信的引脚中，将该寄存器的对应位设置为 1，允许使用噪声滤波器。

噪声滤波器启用时，对 CPU/外围硬件时钟(f_{CLK})执行两个时钟匹配检测以及同步。关闭噪声滤波器时，只与 CPU/外围硬件时钟(f_{MCK})执行同步。

使用 1 位或 8 位存储器操作指令设置 NFEN0 寄存器。

产生复位信号后，NFEN0 寄存器被清除为 00H。

图 15-19. 噪声滤波器允许寄存器 0 (NFEN0)的格式

地址: F0070H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
NFEN0	0	0	0	0	0	SNFEN10	0	SNFEN00

SNFEN10	RxD1 引脚的噪声滤波器的使用 (RxD1/CMP5P/ANI16/P03)
0	噪声滤波器关闭
1	噪声滤波器启用
作为 RxD1 引脚使用时，将 SNFEN10 设为 1。 作为 RxD1 引脚以外使用时，将 SNFEN10 清除为 0。	

SNFEN00	RxD0 引脚的滤波器的使用 (SI00/RxD0/TKCO01/INTP21/SDAA0/(TI07)/(DALIRxD4)/(TxRx4)/P11)
0	噪声滤波器关闭
1	噪声滤波器启用
作为 RxD0 引脚使用时，将 SNFEN00 设为 1。 作为 RxD0 引脚以外使用时，将 SNFEN00 清除为 0。	

注意事项 必须将位 7 至 3 和 1 清除为 0。

(17) 端口输入模式寄存器 0, 1 (PIM0, PIM1)

该寄存器以 1 位为单位设置端口 0 和 1 的输入缓冲。

用 1 位或 8 位存储器操作指令来设置 PIM0 和 PIM1 寄存器。

产生复位信号后，PIM0 和 PIM1 寄存器被清除为 00H。

图 15-20. 端口输入模式寄存器 0, 1 (PIM0, PIM1) 的格式

地址: F0040H	复位后: 00H	R/W								
符号	7	6	5	4	3	2	1	0		
PIM0	0	0	0	0	PIM03	0	0	0		

地址: F0041H	复位后: 00H	R/W								
符号	7	6	5	4	3	2	1	0		
PIM1	0	0	0	0	0	0	PIM11	PIM10		

PIMmn	Pmn引脚输入缓冲器的选择 (m = 0, 1; n = 0, 1, 3)
0	普通输入缓冲器
1	TTL输入缓冲器

(18) 端口输出模式寄存器 0, 1 (PIM0, PIM1)

该寄存器以 1 位为单位设置端口 0 和 1 的输出模式。

使用 1 位或 8 位存储器操作指令来设置 POM0 和 POM1 寄存器。

另外，通过设置 POM0 和 POM1 寄存器以及 PUxx 寄存器来选择是否使用片上上拉电阻。

产生复位信号后，POM0 和 POM1 寄存器被清除为 00H。

图 15-21. 端口输出模式寄存器 0, 1 (POM0, POM1) 的格式

地址: F0050H	复位后: 00H	R/W								
符号	7	6	5	4	3	2	1	0		
POM0	0	0	0	0	0	POM02	0	0		

地址: F0051H	复位后: 00H	R/W								
符号	7	6	5	4	3	2	1	0		
POM1	0	0	0	0	0	POM12	POM11	POM10		

POMmn	Pmn引脚的输出缓冲器的选择(m = 0, 1; n = 0至2)
0	普通输出模式 输入时，允许PUmn位。
1	N沟开漏输出 (V _{DD} 耐压)模式 输入时，PUmn位无效。

(19) 端口模式寄存器 0, 1 (PM0, PM3)

该寄存器以 1 位为单位设置端口 0 和 1 的输入/输出。

将复用于串行数据输出引脚或串行时钟输出引脚的端口（例如 P02/TxD1/ANI17、P12/SCK00/(TKCO03)），用作串行数据输出或串行时钟输出时，将与各端口对应的端口模式寄存器(PMxx)位和端口模式控制寄存器(PMCxx)位设置为 0。此时将与各端口对应的端口寄存器(Pxx)位设置为 1。

示例： P02/TxD1/ANI17 用于串行数据输出或串行时钟输出时
端口模式控制寄存器 0 的 PMC02 位设为 0。
端口模式寄存器 0 的 PM02 位设为 0。
端口寄存器 0 的 P02 位设为 1。

将复用于串行数据输入引脚或串行时钟输入引脚的端口（例如 P03/RxD1/CMP5P/ANI16、P12/SCK00/(TKCO03)），用作串行数据输入或串行时钟输入时，将与各端口对应的端口模式寄存器(PMxx)位设置为 1。并将端口模式控制寄存器(PMCxx)位设置为 0。这时可以将端口寄存器(Pxx)位设置为 0 或 1。

示例： P12/SCK00/(TKCO03)用于串行数据输入或串行时钟输入时
端口模式控制寄存器 1 的 PMC12 位设置为 0。
端口模式寄存器 1 的 PM12 位设置为 1。
端口寄存器 1 的 P12 位设置为 0 或 1。

用 1 位或 8 位存储器操作指令来设置 PM0 和 PM1 寄存器。
产生复位信号后，PM0 和 PM1 寄存器被设置为 FFH。

图 15-22. 端口模式寄存器 0 和 1 (PM0 和 PM1)的格式

地址: FFF20H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM0	1	PM06	PM05	1	PM03	PM02	1	1

地址: FFF21H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM1	1	1	1	1	1	PM12	PM11	PM10

PMmn	Pmn引脚输入/输出模式选择(m = 0, 1; n = 0至3, 5, 6)
0	输出模式 (输出缓冲器启用)
1	输入模式 (输出缓冲器关闭)

注意事项 必须将 PM0 寄存器的位 0、1、4、7 以及 PM1 寄存器的位 3 至 7 清除为 1。

对于 32, 30 和 20 引脚产品, 在解除复位后, 必须通过软件将以下的位设置为输出模式 (通过设置端口寄存器和端口模式寄存器为 0 来指定)。

32 引脚产品: PM0 寄存器的位 2 和 3 以及 PM1 寄存器的位 2

30 引脚产品: PM0 寄存器的位 5 和 6 以及 PM1 寄存器的位 2

20 引脚产品: PM0 寄存器的位 2、3、5 和 6 以及 PM1 寄存器的位 2

15.4 操作停止模式

串行阵列单元的各串行接口均有操作停止模式。

在此模式下，不能执行串行通信，因而可降低功耗。

另外，在该模式下，用于串行接口的引脚可用作端口功能引脚。

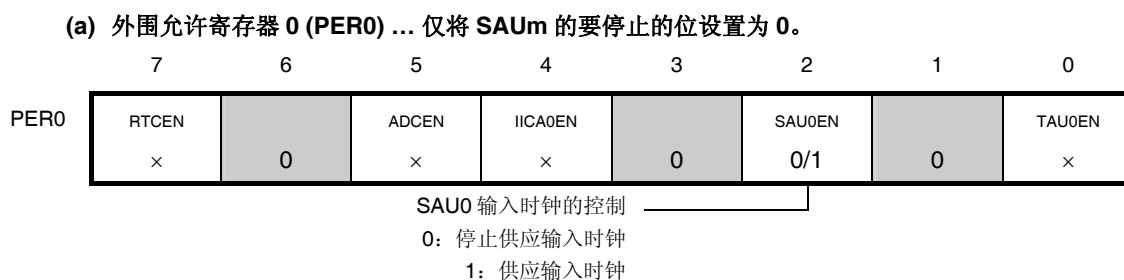
15.4.1 以单元单位停止操作

使用外围允许寄存器 0 (PER0)按照单元单位设置停止操作。

PER0 寄存器用于允许或禁止向外围硬件供应时钟。通过停止向未使用的硬件供应时钟，可以降低功耗和噪声。

将位 2 (SAU0EN)设置为 0 以停止操作串行阵列单元 0。

图 15-23. 以单元单位停止操作时外围允许寄存器 0 (PER0) 的设置



注意事项 1. 如果 SAU0EN = 0，对于串行阵列单元 0 的控制寄存器进行的写入操作被忽略，而且该寄存器的读取值总为初始值。

但如下的寄存器除外：

- 输入切换控制寄存器(ISC)
- 噪声滤波器允许寄存器 0 (NFEN0)
- 串行待机控制寄存器 0 (SSC0)
- 端口输入模式寄存器 0, 1 (PIM0, PIM1)
- 端口输出模式寄存器 0, 1 (POM0, POM1)
- 端口模式控制寄存器 0 (PMC0)
- 端口模式寄存器 0, 1 (PM0, PM1)
- 端口寄存器 0, 1 (P0, P1)

2. 必须将位 1、3 和 6 清除为 0。

备注 ：禁止设置 (设置为初始值)

x：串行阵列单元未使用的位 (取决于其他外围功能的设置)

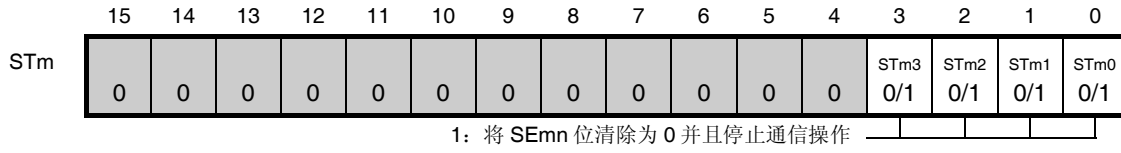
0/1：可根据用户的用途设为 0 或 1

15.4.2 以通道单位停止操作

使用以下各种寄存器设置通道单位停止操作。

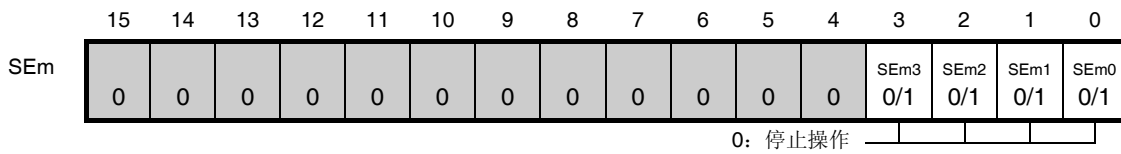
图 15-24. 以通道单位停止操作时每个寄存器的设置

(a) 串行通道停止寄存器 m (STm) ... 该寄存器用于设置是否允许各个通道的通信/计数的触发寄存器。



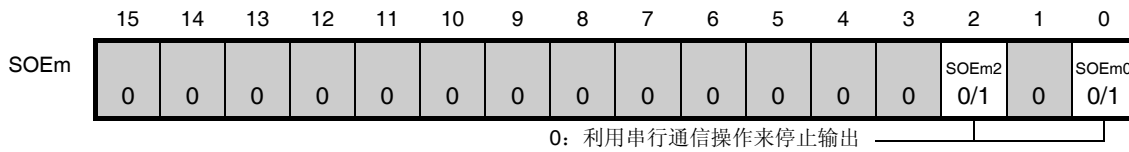
* 由于 STmn 位为触发位，所以当 SEmn = 0 时，该位将立即被清除。

(b) 串行通道允许状态寄存器 m (SEm) ... 该寄存器指示是否允许或停止各个通道的数据发送/接收操作。



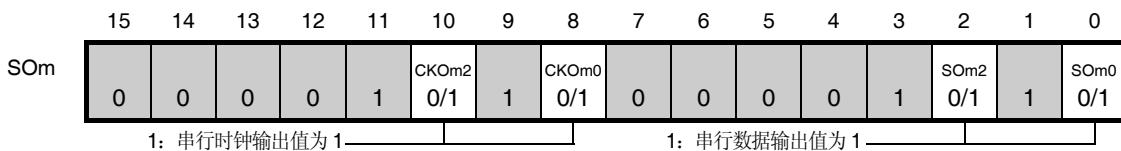
* SEm 寄存器为只读状态寄存器。通过设置 STm 寄存器来停止操作。
对于已经停止操作的通道，可以通过软件设置 SOm 寄存器的 CKOm 位的值。

(c) 串行输出允许寄存器 m (SOEm) ... 该寄存器用于设置允许或停止各个通道的串行通信操作的输出。



* 对于通道 n (其串行输出已停止)，可以用软件设置 SOm 寄存器的 SOMn 位的值。

(d) 串行输出寄存器 m (SOm) ... 该寄存器是各个通道的串行输出的缓冲寄存器。



* 在将对应于各个通道的引脚用作端口功能引脚时，将相应的 CKOm、SOMn 位设置为 1。

备注 1. m: 单元编号(m = 0), n: 通道编号(n = 0 至 3)

2. ■: 禁止设置 (设为初始值), 0/1: 可根据客户的用途设为 0 或 1

15.5 3 线串行输入/输出 (CSI00) 通信的操作

这是一种采用三条线路的时钟同步式通信功能：串行时钟($\overline{\text{SCK}}$) 和串行数据 (SI 和 SO)线路。

[数据的发送/接收]

- 数据长度为 7 或 8 位
- 发送/接收数据的相控制
- MSB/LSB 选择
- 发送/接收数据的电平设置

[时钟控制]

- 主/从的选择
- 输入/输出时钟的相位控制
- 通过预分频器和通道内部计数器设置传送周期
- 最大传送速率

主通信时：Max. $f_{\text{MCK}}/2$ ^{註1, 2}

从通信时：Max. $f_{\text{MCK}}/6$ ^{註2}

[中断功能]

- 传送结束中断/缓冲器空中断

[错误检测标志]

- 溢出错误

另外，以下通道的 CSI 支持 SNOOZE 模式。当在 STOP 模式下检测到 $\overline{\text{SCK}}$ 输入时，借助 SNOOZE 模式，不需要 CPU 操作就可以接收数据。仅可以设置以下 CSI。

注 1. 主通信模式下，满足以下条件时最大传送速率为 $f_{\text{MCK}}/2$ ：

- $2.7 \text{ V} \leq V_{\text{DD}} \leq 5.5 \text{ V}$
- $f_{\text{MCK}} \leq 24 \text{ MHz}$
- $\text{PIOR1} = 0$

其他情况下的最大传送速率为 $f_{\text{MCK}}/4$ 。

2. 在满足 $\overline{\text{SCK}}$ 周期时间(t_{KCY})特性的范围内使用时钟(参阅第三十二章 电特性)。

支持 3 线串行输入/输出 (CSI00)的通道为通道 0。

• 20 和 32 引脚产品

通道	用作 CSI	用作 UART	用作简易 I ² C
0	-	UART0 (支持 LIN-bus、DMX512)	-
1	-		-
2	-	-	-
3	-		-

• 30 引脚产品

通道	用作 CSI	用作 UART	用作简易 I ² C
0	-	UART0 (支持 LIN-bus、DMX512)	-
1	-		-
2	-	UART1	-
3	-		-

• 38 引脚产品

通道	用作 CSI	用作 UART	用作简易 I ² C
0	CSI00	UART0 (支持 LIN-bus、DMX512)	-
1	-		-
2	-	UART1	-
3	-		-

3 线串行输入/输出 (CSI00)具有如下七种类型的通信操作。

- 主发送 (参阅 15.5.1)
- 主接收 (参阅 15.5.2)
- 主发送/接收 (参阅 15.5.3)
- 从发送 (参阅 15.5.4)
- 从接收 (参阅 15.5.5)
- 从发送/接收 (参阅 15.5.6)
- SNOOZE 模式功能 (参阅 15.5.7)

15.5.1 主发送

主发送指 RL78/I1A 输出传送时钟，并将数据发送至另一个器件。

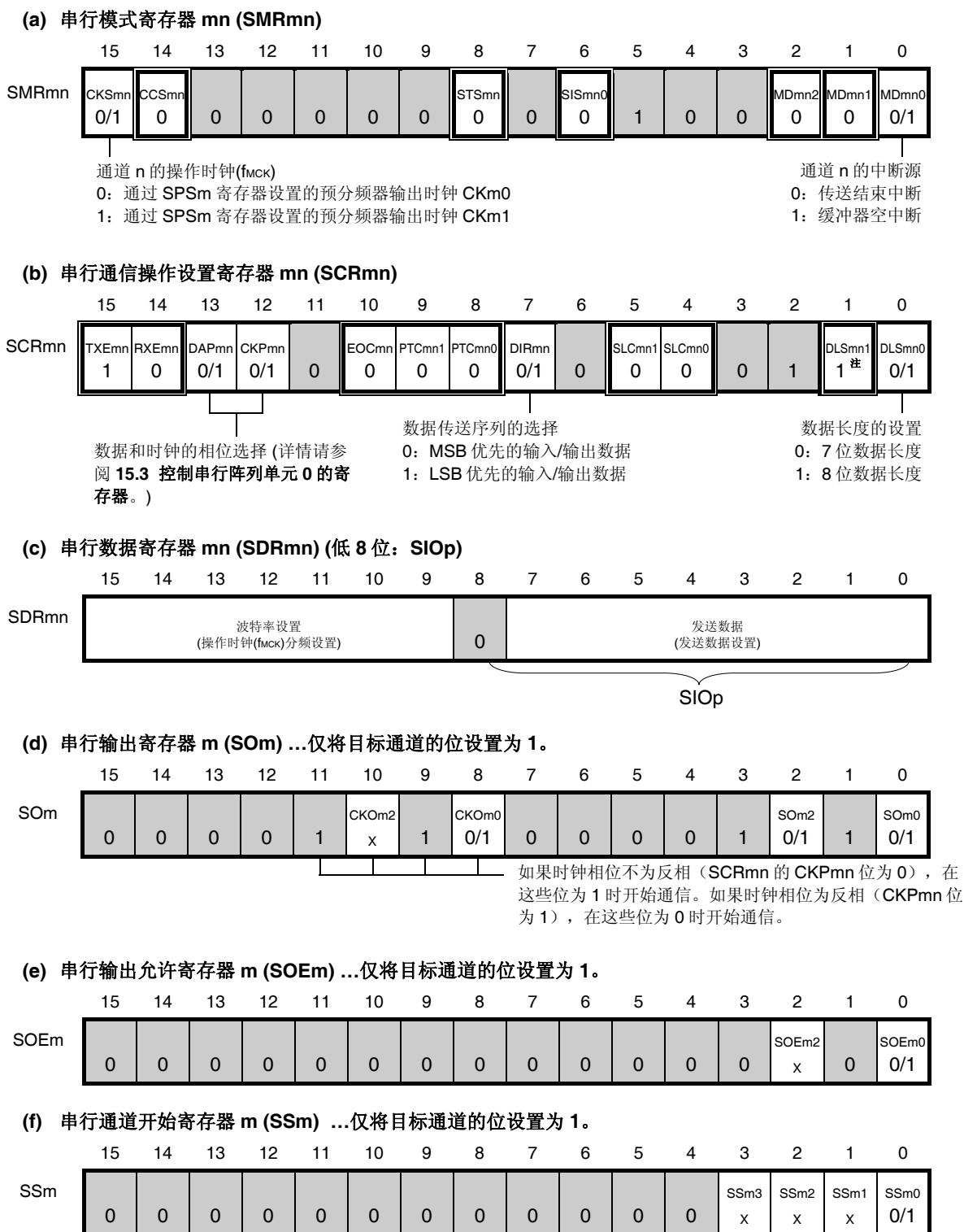
3线串行输入/输出	CSI00
目标通道	SAU0的通道0
使用引脚	$\overline{\text{SCK00}}$, SO00
中断	INTCSI00
	可选择传送结束中断(单传送模式时)或缓冲器空中断(连续传送模式时)
错误检测标志	无
传送数据长度	7或8位
传送速率	Max. $f_{\text{MCK}}/2$ [Hz] (CSI00) Min. $f_{\text{CLK}}/(2 \times 2^{15} \times 128)$ [Hz] [※] f_{CLK} : 系统时钟频率
数据相位	可通过SCRmn寄存器的DAPmn位选择 <ul style="list-style-type: none"> • DAPmn = 0: 从开始串行时钟操作时开始数据输出。 • DAPmn = 1: 从开始串行时钟操作的半个时钟前开始数据输出。
时钟相位	可通过SCRmn寄存器的CKPmn位选择 <ul style="list-style-type: none"> • CKPmn = 0: 不反相 (在$\overline{\text{SCK}}$的下降沿输出数据，在其上升沿输入数据) • CKPmn = 1: 反相 (在$\overline{\text{SCK}}$的上升沿输出数据，在其下降沿输入数据)
数据方向	MSB或LSB优先

注 在满足以上条件及电特性中的外围功能特性(参阅第三十二章 电特性)的范围内使用此操作。

备注 m: 单元编号(m = 0), n: 通道编号(n = 0)

(1) 寄存器设置

图 15-25. 3 线串行输入/输出(CSI00)的主发送时的寄存器设置内容示例



备注 1. m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)
 2. □: 设置固定于 CSI 主传送模式, □ 禁止设置 (设为初始值)
 0/1: 可根据客户的用途设为 0 或 1

(2) 操作步骤

图 15-26. 主发送的初始设定步骤

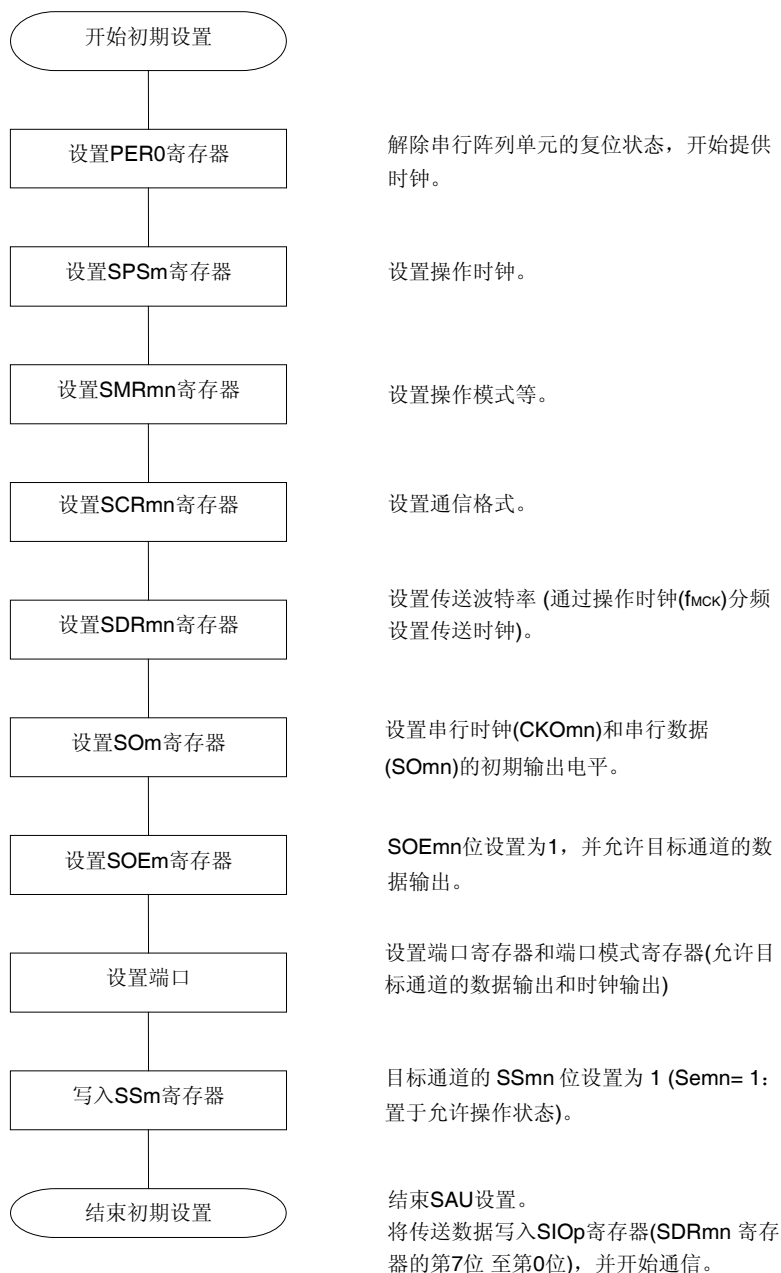


图 15-27. 停止主发送的步骤

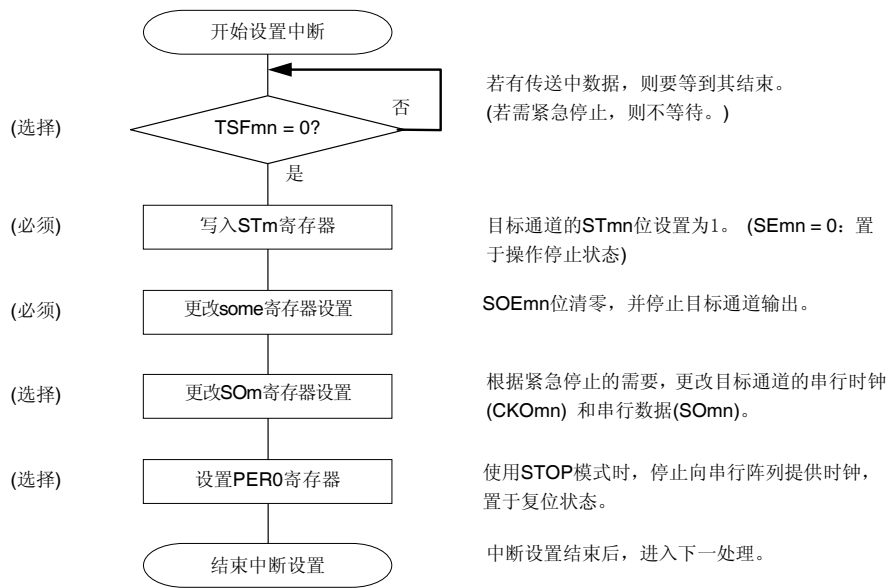
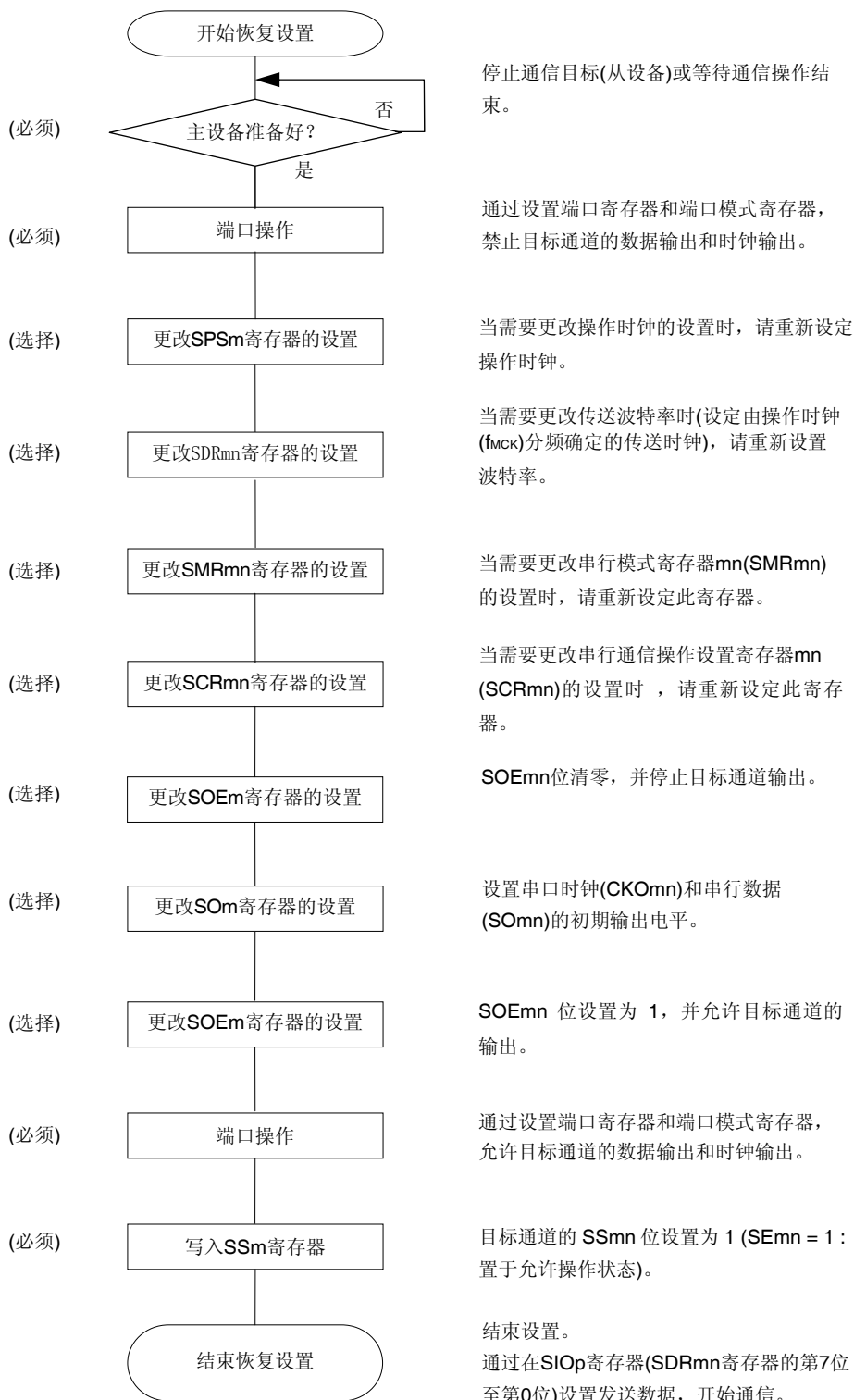


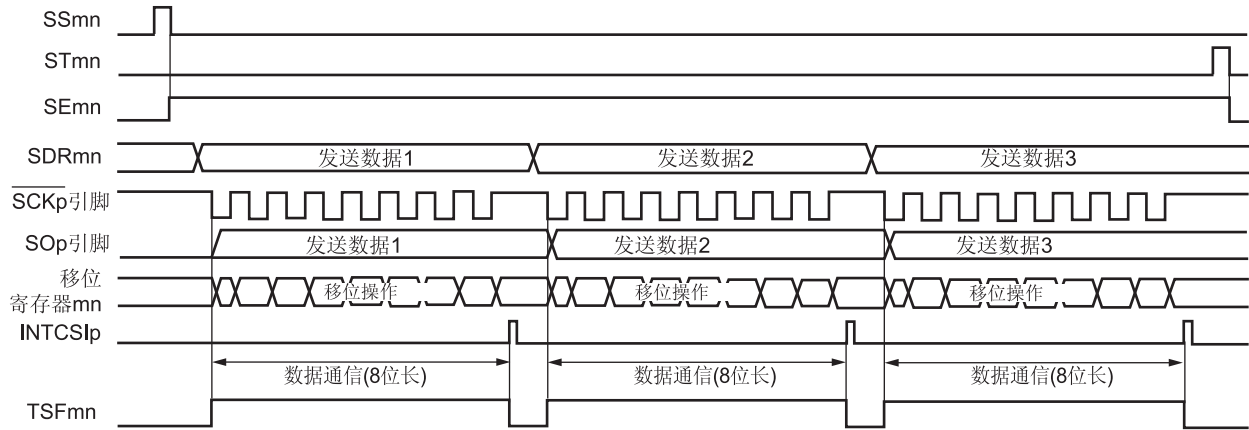
图 15-28. 恢复主发送的步骤



备注 停止主发送时如果 PER0 被改写，且停止了时钟供给时，请等待通信目标（从设备）停止或通信结束，然后执行初始设置，而不是重传设置。

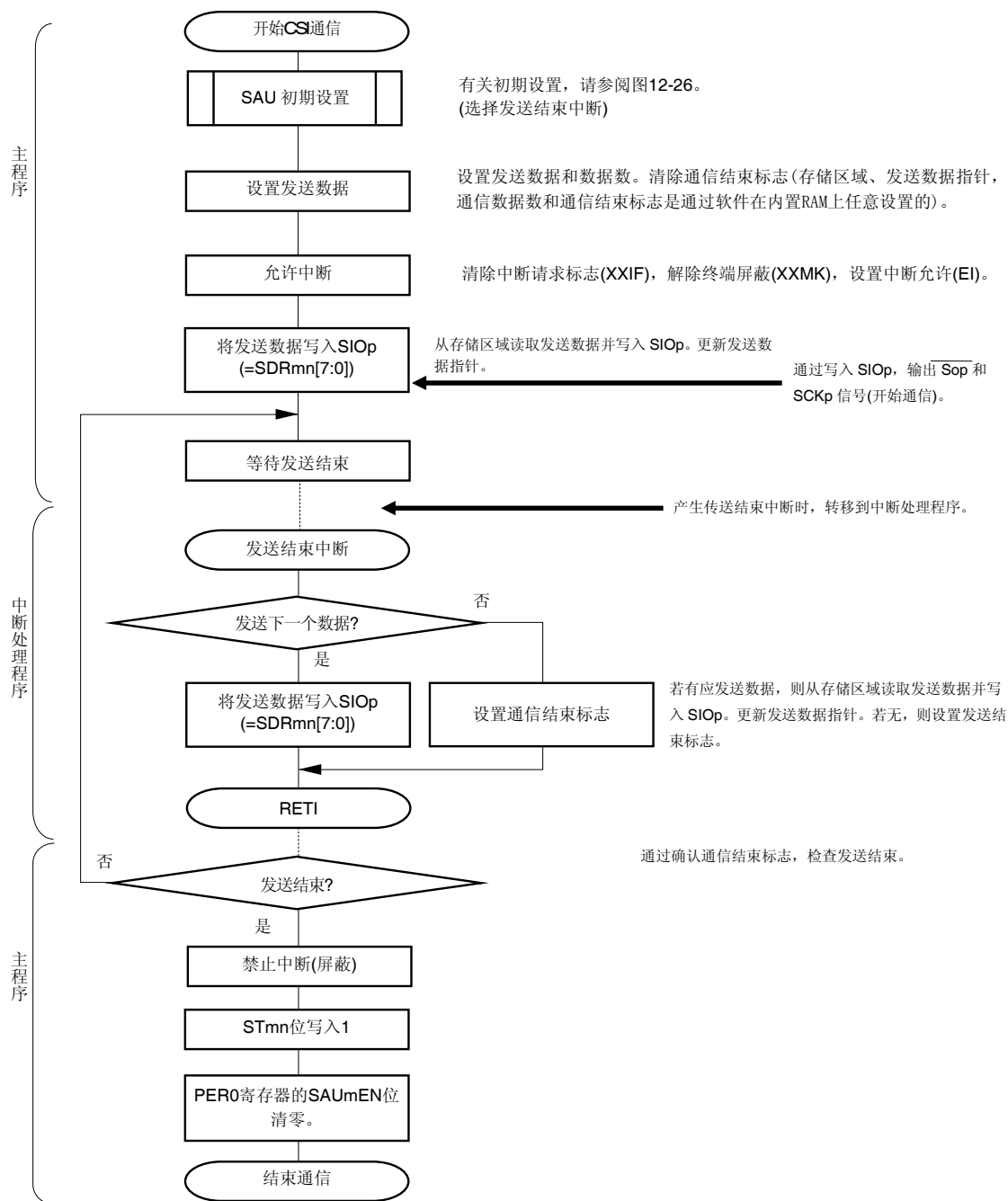
(3) 处理流程 (单发送模式时)

图 15-29. 主发送(单发送模式时)的时序图
(类型 1: DAPmn = 0、CKPmn = 0)



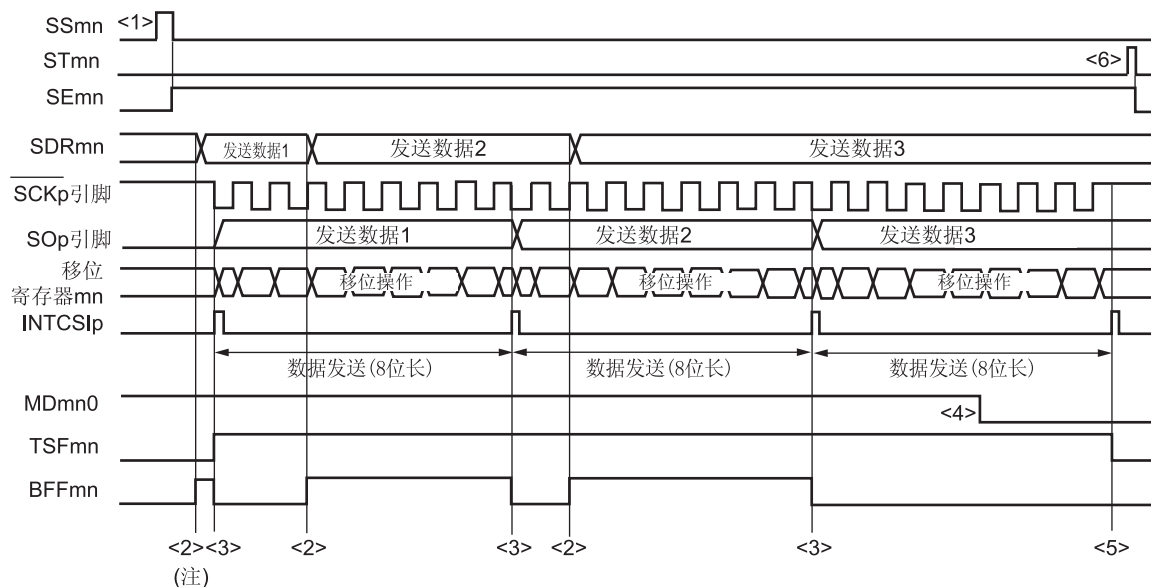
备注 m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)

图 15-30. 主发送 (单发送模式时)的流程图



(4) 处理流程 (连续传送模式时)

图 15-31. 主发送 (连续传送模式时)的时序图
(类型 1: DAPmn = 0、CKPmn = 0)

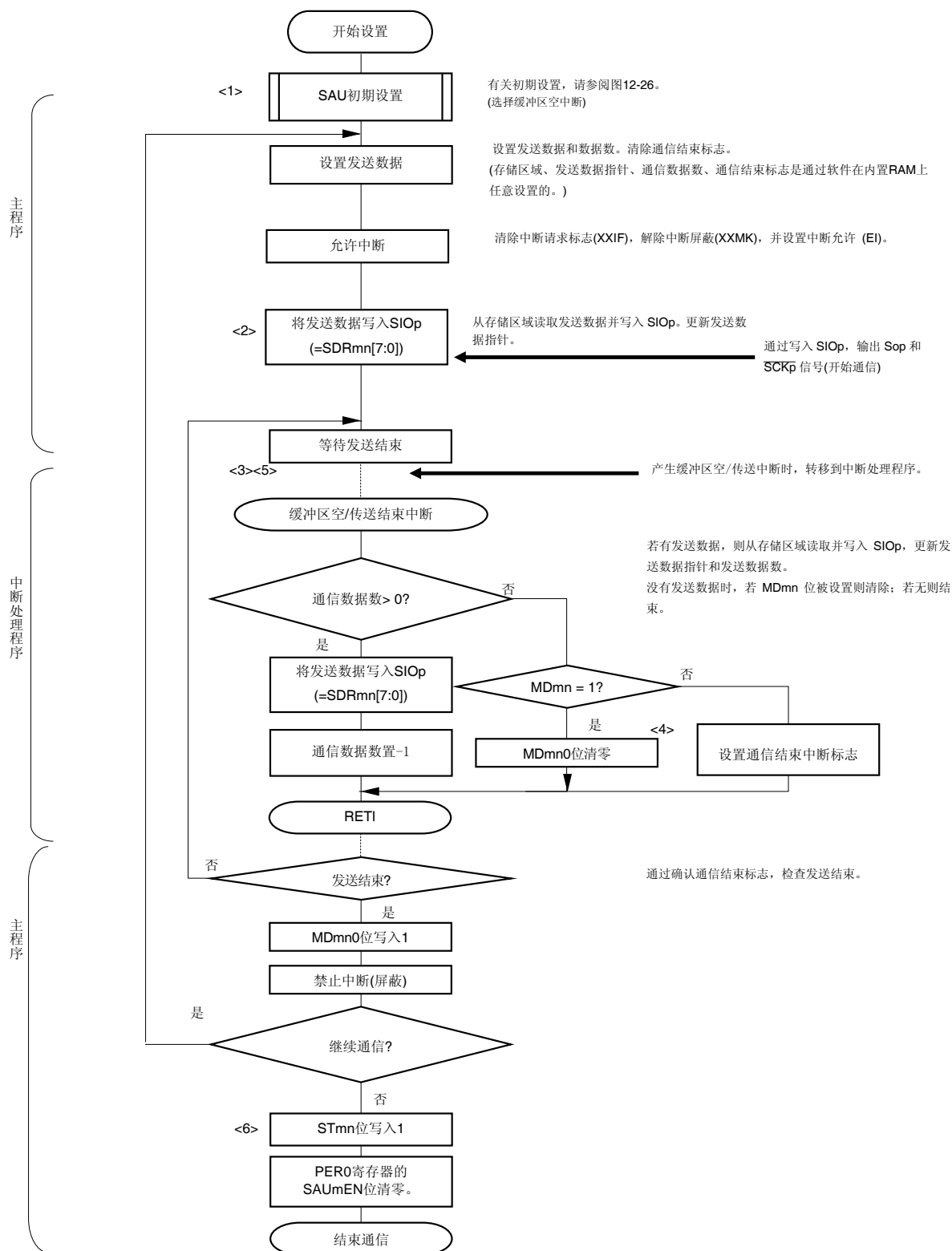


注 如果在串行状态寄存器 mn (SSRmn) 的 BFFmn 位为 1 时将发送数据写入 SDRmn 寄存器 (将有效数据存储于串行数据寄存器 mn (SDRmn) 中)，则发送数据将被覆盖。

注意事项 即使在操作过程中，也可改写串行模式寄存器 mn (SMRmn) 的 MDmn0 位。然而必须在开始传送最后一位之前改写，以便在最后发送数据的传送结束中断前完成改写。

备注 m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)

图 15-32. 主发送 (连续传送模式时)的流程图



备注 此图中的<1>至<6>对应于图 15-31 主发送 (连续传送模式时)的时序图中的<1>至<6>。

15.5.2 主接收

主接收指 RL78/I1A 输出传送时钟，并从其他器件接收数据。

3线串行输入/输出	CSI00	CSI01	CSI10	CSI11	CSI20	CSI21	CSI30	CSI31
目标通道	SAU0的 通道0	SAU0的 通道1	SAU0的 通道2	SAU0的 通道3	SAU1的 通道0	SAU1的 通道1	SAU1的 通道2	SAU1的 通道3
使用引脚	$\overline{\text{SCK00}}$, SI00	$\overline{\text{SCK01}}$, SI01	$\overline{\text{SCK10}}$, SI10	$\overline{\text{SCK11}}$, SI11	$\overline{\text{SCK20}}$, SI20	$\overline{\text{SCK21}}$, SI21	$\overline{\text{SCK30}}$, SI30	$\overline{\text{SCK31}}$, SI31
中断	INTCSI00	INTCSI01	INTCSI10	INTCSI11	INTCSI20	INTCSI21	INTCSI30	INTCSI31
	可选择传送结束中断(单传送模式时)或缓冲器空中断(连续传送模式时)							
错误检测标志	仅限溢出错误检测标志 (OVFmn)							
传送数据长度	7或8位							
传送速率	Max. $f_{\text{MCK}}/2$ [Hz] (CSI00), $f_{\text{MCK}}/4$ [Hz] (other than CSI00) Min. $f_{\text{CLK}}/(2 \times 2^{15} \times 128)$ [Hz] [#] f_{CLK} : 系统时钟频率							
数据相位	可通过SCRmn寄存器的DAPmn位选择 <ul style="list-style-type: none"> • DAPmn = 0: 从开始串行时钟操作时开始数据输入。 • DAPmn = 1: 从开始串行时钟操作的半个时钟前开始数据输入。 							
时钟相位	可通过SCRmn寄存器的CKPmn位选择 <ul style="list-style-type: none"> • CKPmn = 0: 正相 • CKPmn = 1: 反相 							
数据方向	MSB或LSB优先							

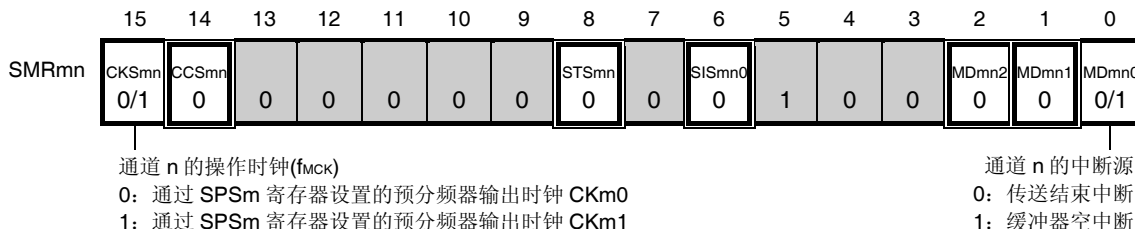
注 在满足以上条件及电特性中的外围功能特性(参阅第三十二章 电特性)的范围内使用此操作。

备注 m: 单元编号(m = 0), n: 通道编号(n = 0)

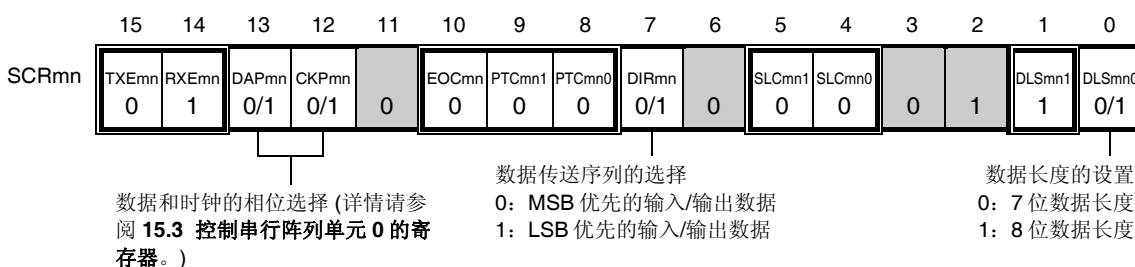
(1) 寄存器设置

图 15-33. 3 线串行输入/输出 (CSI00)的主接收时的寄存器设置内容示例(1/2)

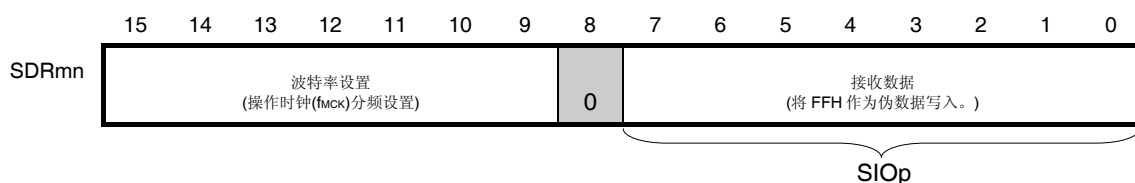
(a) 串行模式寄存器 mn (SMRmn)



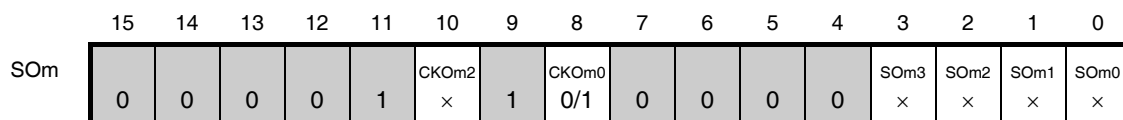
(b) 串行通信操作设置寄存器 mn (SCRmn)



(c) 串行数据寄存器 mn (SDRmn) (低 8 位: SIOp)



(d) 串行输出寄存器 m (SOM) ... 仅将目标通道的位设置为 1。



如果时钟相位不为反相 (SCRmn 的 CKPmn 位为 0)，在这些位为 1 时开始通信。如果时钟相位为反相 (CKPmn 位为 1)，在这些位为 0 时开始通信。

- 备注 1. m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)
2. : 设置固定于 CSI 主接收模式, : 禁止设置 (设为初始值)
 x: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)
 0/1: 可根据客户的用途设为 0 或 1

图 15-33. 3 线串行输入/输出 (CSI00)的主接收时的寄存器设置内容示例(2/2)

(e) 串行输出允许寄存器 m (SOEm) ...此模式下不使用的寄存器。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOEm	0	0	0	0	0	0	0	0	0	0	0	0	0	SOEm2 ×	0	SOEm0 ×

(f) 串行通道开始寄存器 m (SSm) ...仅将目标通道的位设置为 1。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSm	0	0	0	0	0	0	0	0	0	0	0	0	SSm3 ×	SSm2 ×	SSm1 ×	SSm0 0/1

- 备注 1. m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)
2. : 禁止设置(设置为初始值)
 x: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)
 0/1: 可根据客户的用途设为 0 或 1

(2) 操作步骤

图 15-34. 主接收的初始设定步骤

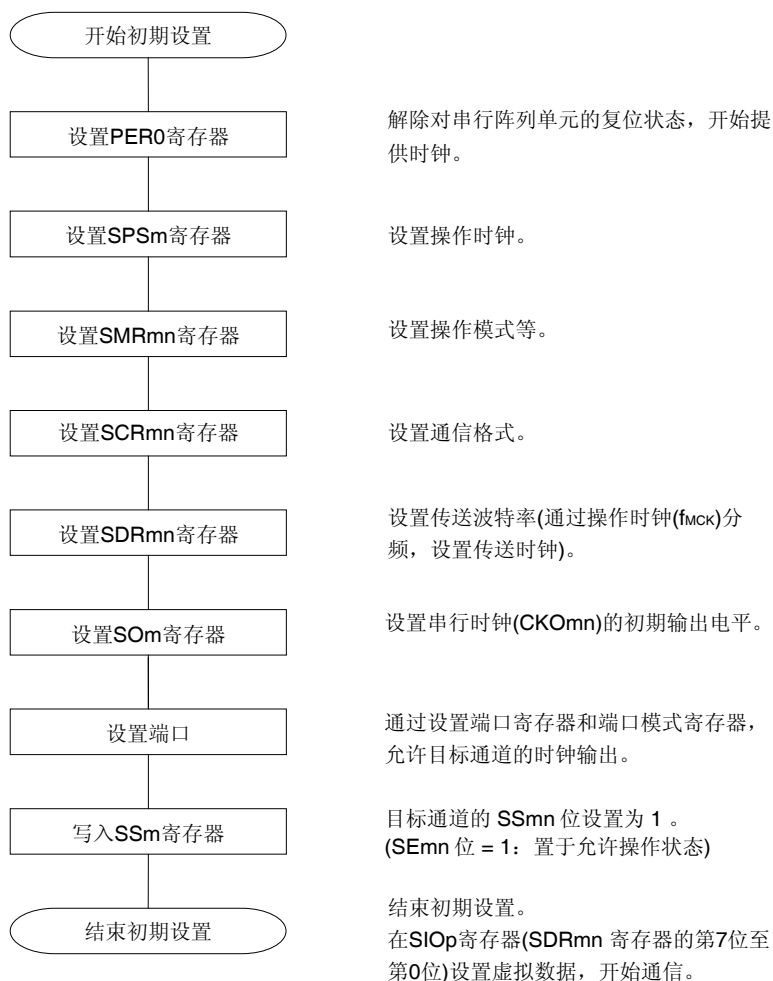


图 15-35. 停止主接收的步骤

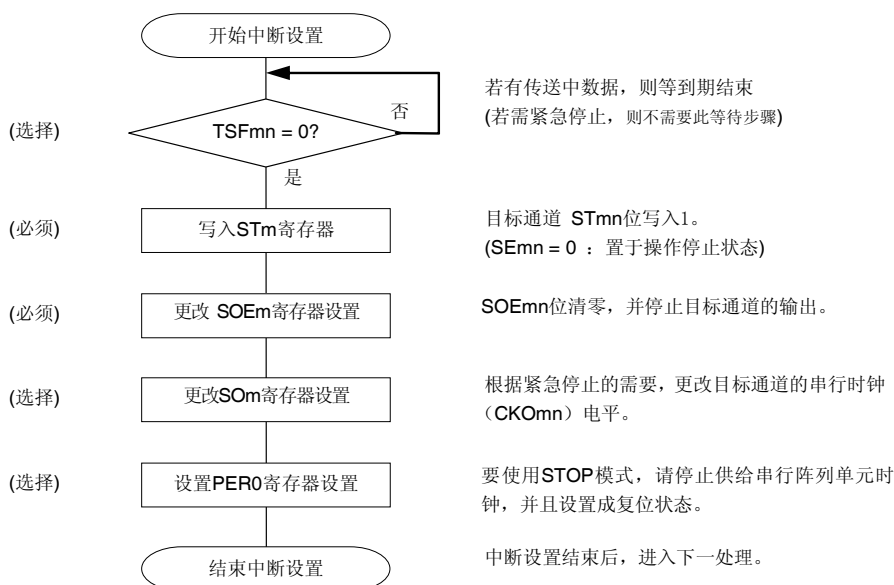


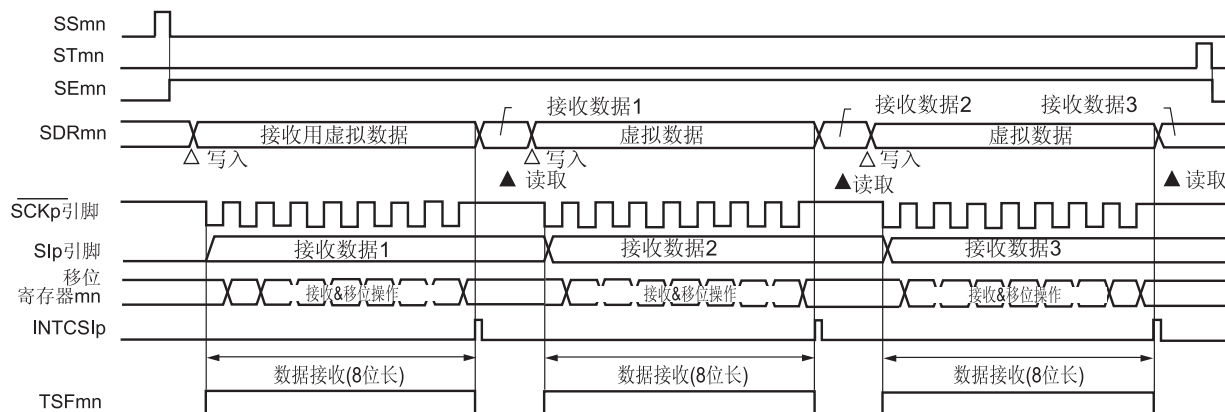
图 15-36. 恢复主接收的步骤



备注 中断主发送时如果 PER0 被改写, 且停止了时钟供给时, 请等待至发送目标 (从) 停止或发送结束, 然后执行初始设置, 而不是重传设置。

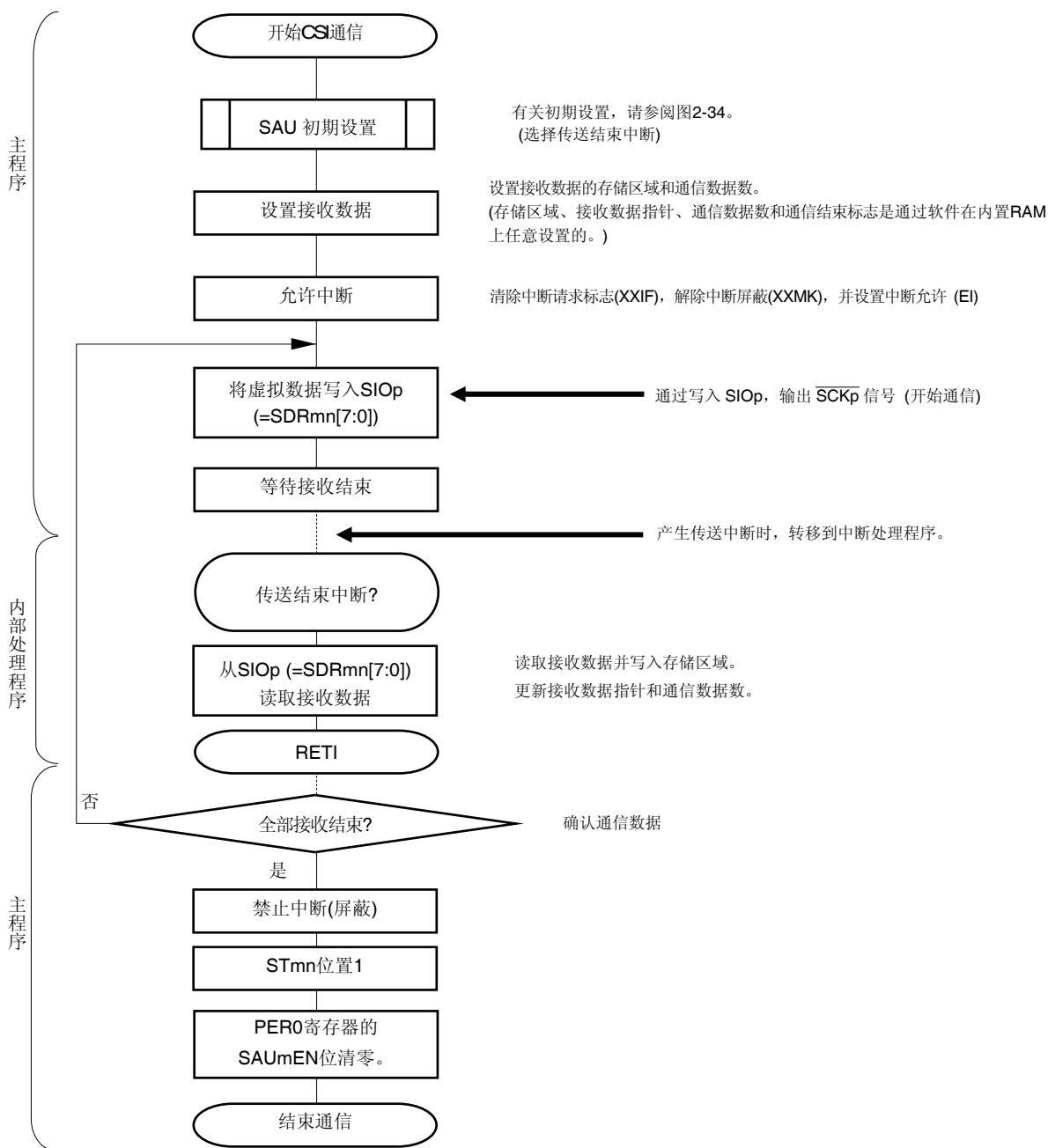
(3) 处理流程 (单接收模式时)

图 15-37. 主接收(单接收模式时)的时序图
(类型 1: DAPmn = 0、CKPmn = 0)



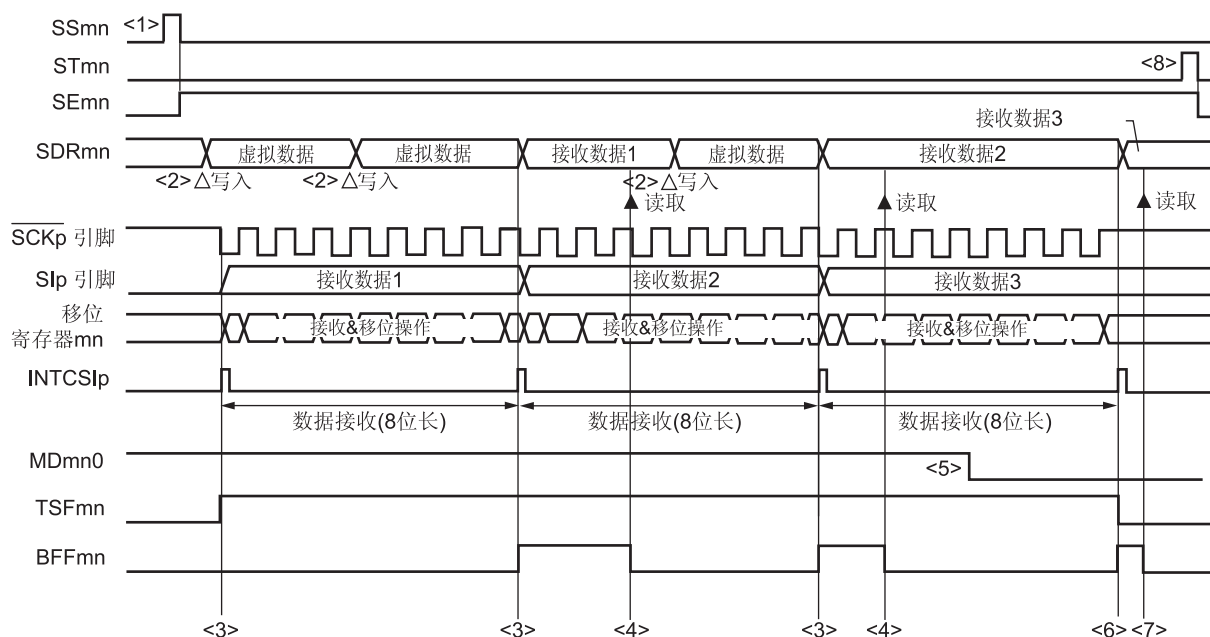
备注 m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)

图 15-38. 主接收 (单接收模式时)的流程图



(4) 处理流程 (连续接收模式时)

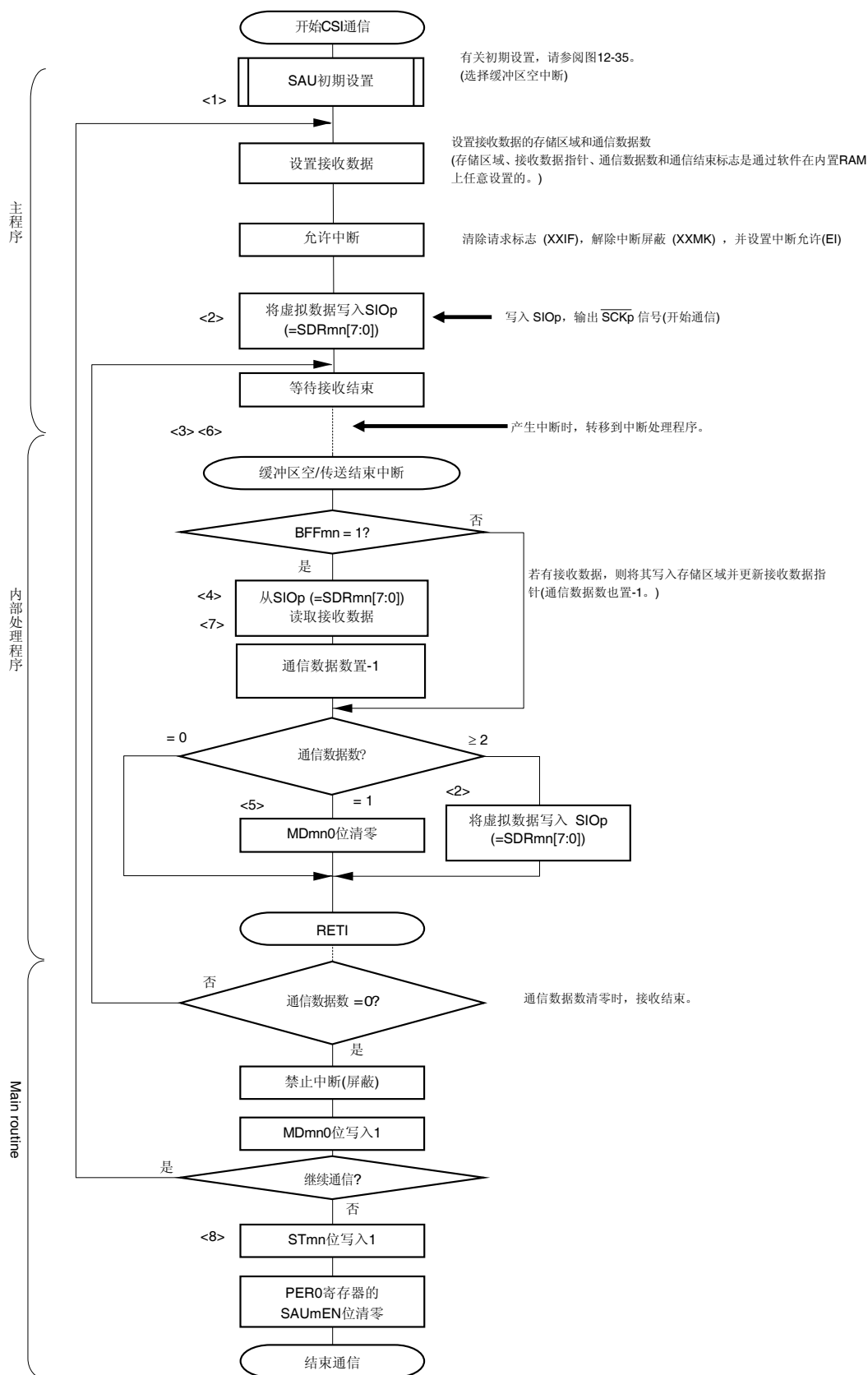
图 15-39. 主接收 (连续接收模式时)的时序图(类型 1: DAPmn = 0、CKPmn = 0)



注意事项 即使在操作过程中也可改写 MDmn0 位。
然而必须在开始接收最后一位之前改写，以便在最后接收数据的传送结束中断之前完成改写。

- 备注**
1. 此图中的<1>至<8>对应于图 15-40 主接收 (连续接收模式时)的流程图中的<1>至<8>。
 2. m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)

图 15-40. 主接收 (连续接收模式时)的流程图



备注 此图中的<1>至<8>对应于图 15-39 主接收 (连续接收模式时)的时序图中的<1>至<8>。

15.5.3 主发送/接收

主发送/接收指 RL78/I1A 输出传送时钟，并将数据发送至其他器件或从其他器件接收数据。

3线串行输入/输出	CSI00
目标通道	SAU0的通道0
使用引脚	SCK00, SI00, SO00
中断	INTCSI00
	可选择传送结束中断(单传送模式时)或缓冲器空中断(连续传送模式时)
错误检测标志	仅限溢出错误检测标志 (OVFmn)
传送数据长度	7或8位
传送速率	Max. $f_{MCK}/2$ [Hz] Min. $f_{CLK}/(2 \times 2^{15} \times 128)$ [Hz] [#] f_{CLK} : 系统时钟频率
数据相位	可通过SCRmn寄存器的DAPmn位选择 <ul style="list-style-type: none"> • DAPmn = 0: 从开始串行时钟操作时开始数据输入/输出 • DAPmn = 1: 从开始串行时钟操作的半个时钟前开始数据输入/输出。
时钟相位	可通过SCRmn寄存器的CKPmn位选择 <ul style="list-style-type: none"> • CKPmn = 0: 正相 • CKPmn = 1: 反相
数据方向	MSB或LSB优先

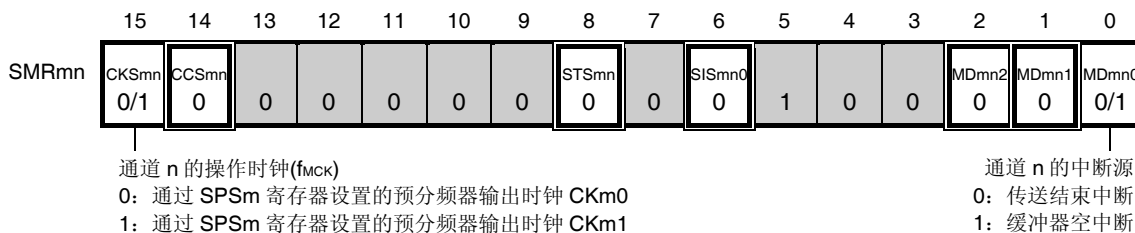
注 在满足以上条件及电特性中的外围功能特性(参阅第三十二章 电特性)的范围内使用此操作。

备注 m: 单元编号(m = 0), n: 通道编号(n = 0)

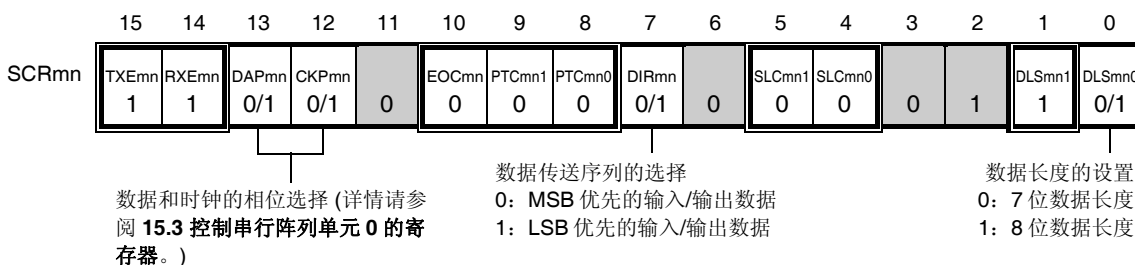
(1) 寄存器设置

图 15-41. 3 线串行输入/输出 (CSI00)的主发送/接收时的寄存器设置内容示例(1/2)

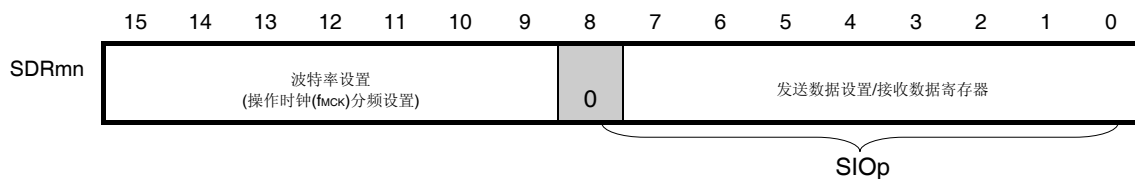
(a) 串行模式寄存器 mn (SMRmn)



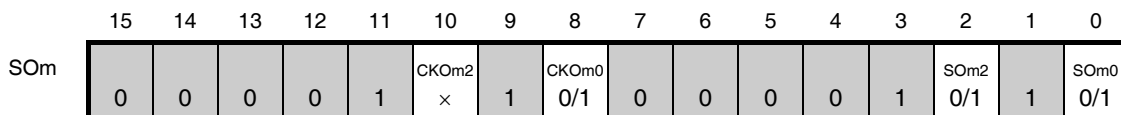
(b) 串行通信操作设置寄存器 mn (SCRmn)



(c) 串行数据寄存器 mn (SDRmn) (低 8 位: SIOp)



(d) 串行输出寄存器 m (SOM) ...仅将目标通道的位设置为 1。



如果时钟相位不为反相 (SCRmn 的 CKPmn 位为 0)，在这些位为 1 时开始通信。如果时钟相位为反相 (CKPmn 位为 1)，在这些位为 0 时开始通信。

- 备注 1. m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)
2. : 设置固定于 CSI 主发送/接收模式
 : 禁止设置 (设为初始值)
 x: 此模式下不能使用的位 (任意模式下均不使用设为初始值)
 0/1: 可根据客户的用途设为 0 或 1

图 15-41. 3 线串行输入/输出 (CSI00)的主发送/接收时的寄存器设置内容示例(2/2)

(e) 串行输出允许寄存器 m (SOEm) ...仅将目标通道的位设置为 1。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOEm	0	0	0	0	0	0	0	0	0	0	0	0	0	SOEm2 ×	0	SOEm0 0/1

(f) 串行通道开始寄存器 m (SSm) ...仅将目标通道的位设置为 1。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSm	0	0	0	0	0	0	0	0	0	0	0	0	SSm3 ×	SSm2 ×	SSm1 ×	SSm0 0/1

注 仅限串行阵列单元 0。

- 备注 1. m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)
 2. : 禁止设置 (设为初始值)
 ×: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)
 0/1: 可根据客户的用途设为 0 或 1

(2) 操作步骤

图 15-42. 主发送/接收的初始设定步骤

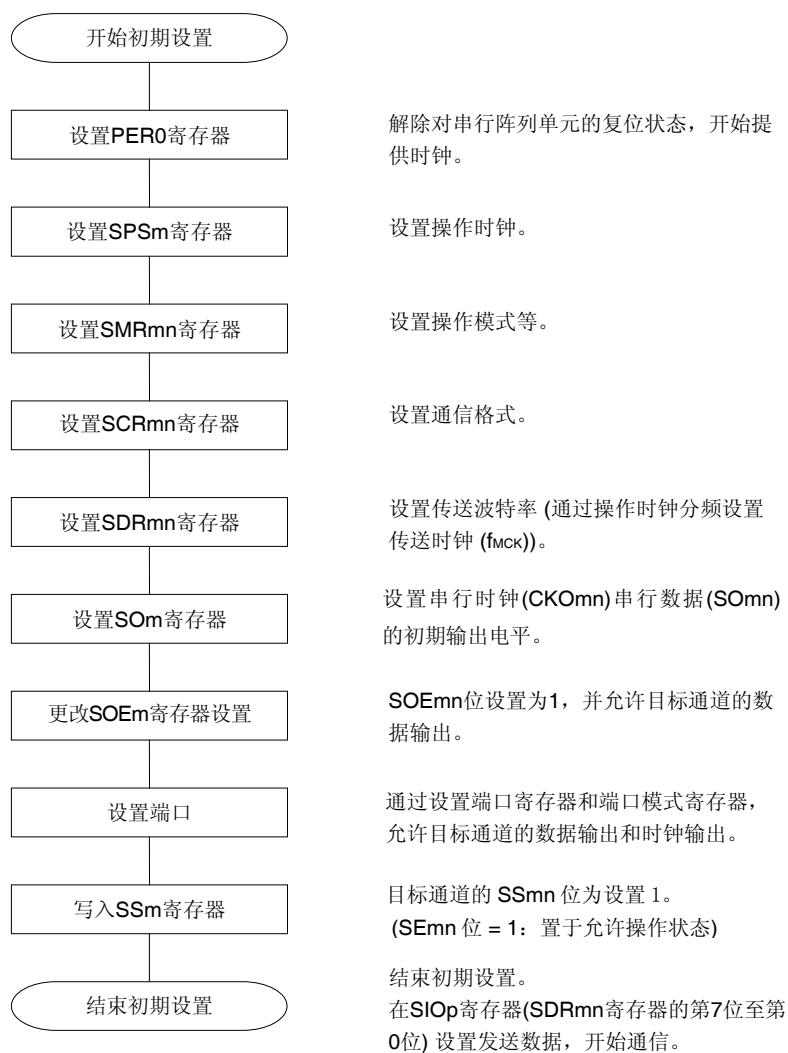


图 15-43. 停止主发送/接收的步骤

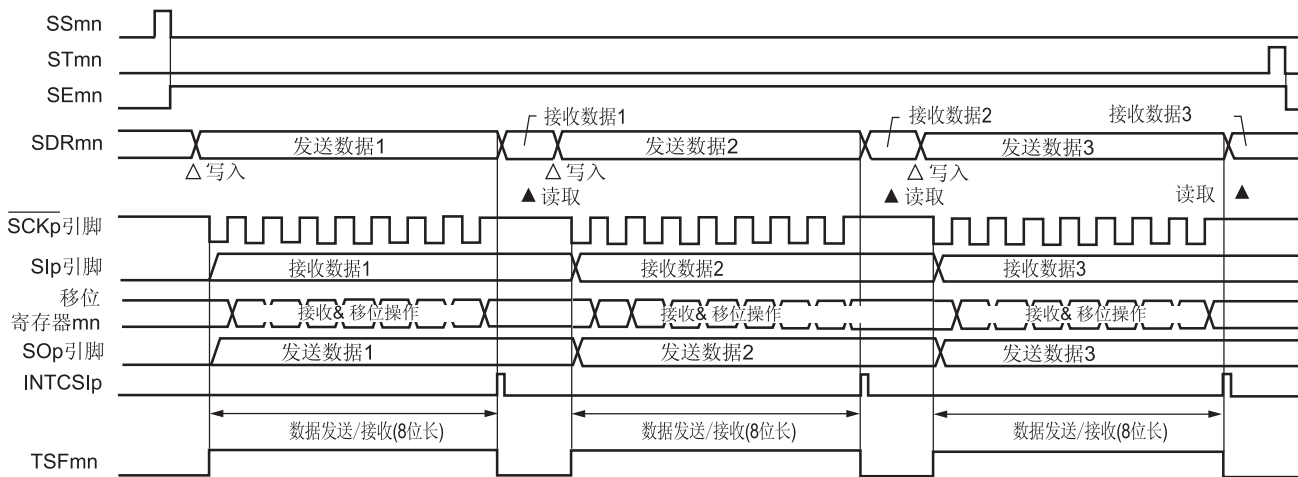


图 15-44. 恢复主发送/接收的步骤



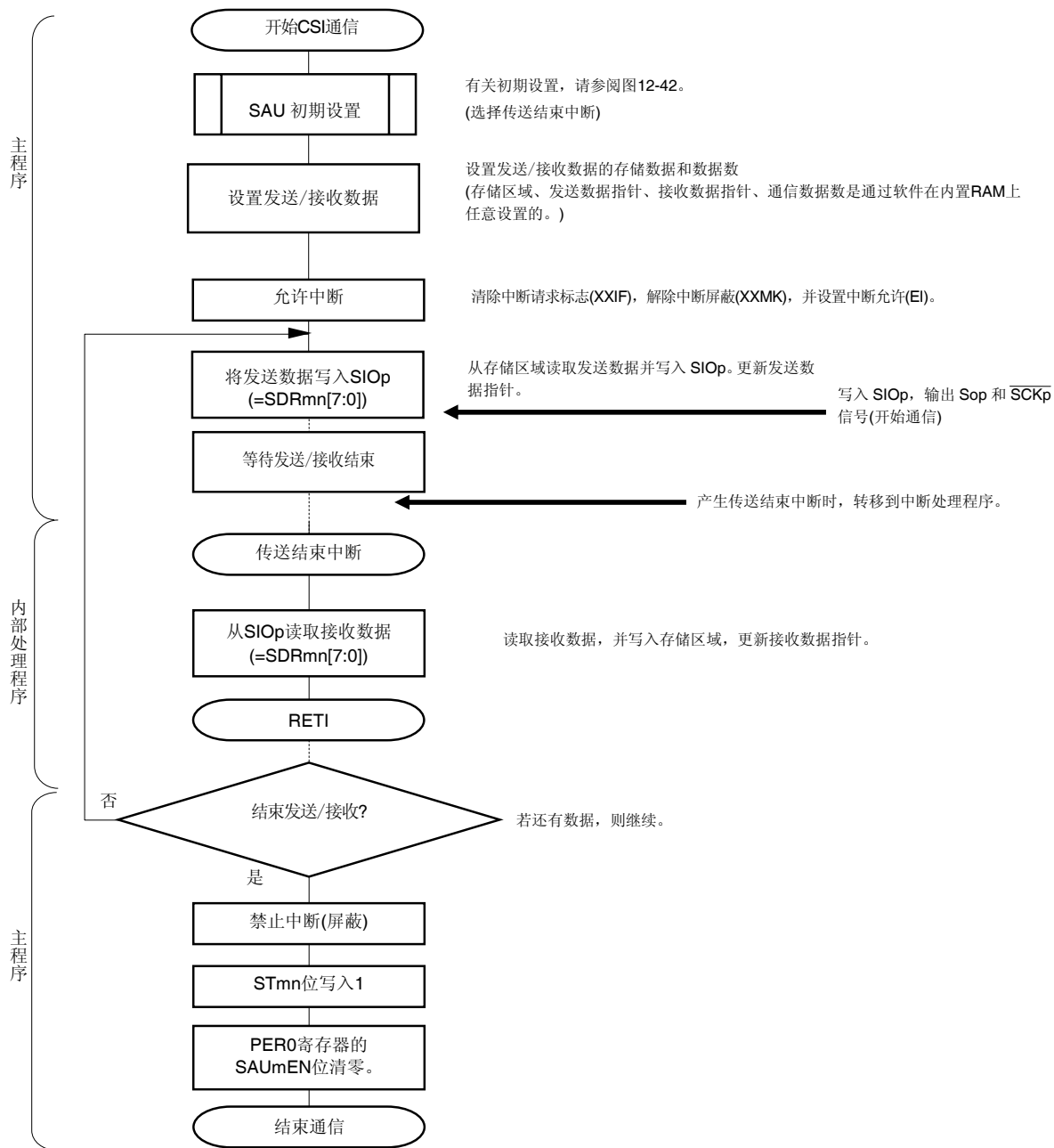
(3) 处理流程 (单发送/接收模式时)

图 15-45. 主发送/接收 (单发送/接收模式时)的时序图
(类型 1: DAPmn = 0、CKPmn = 0)



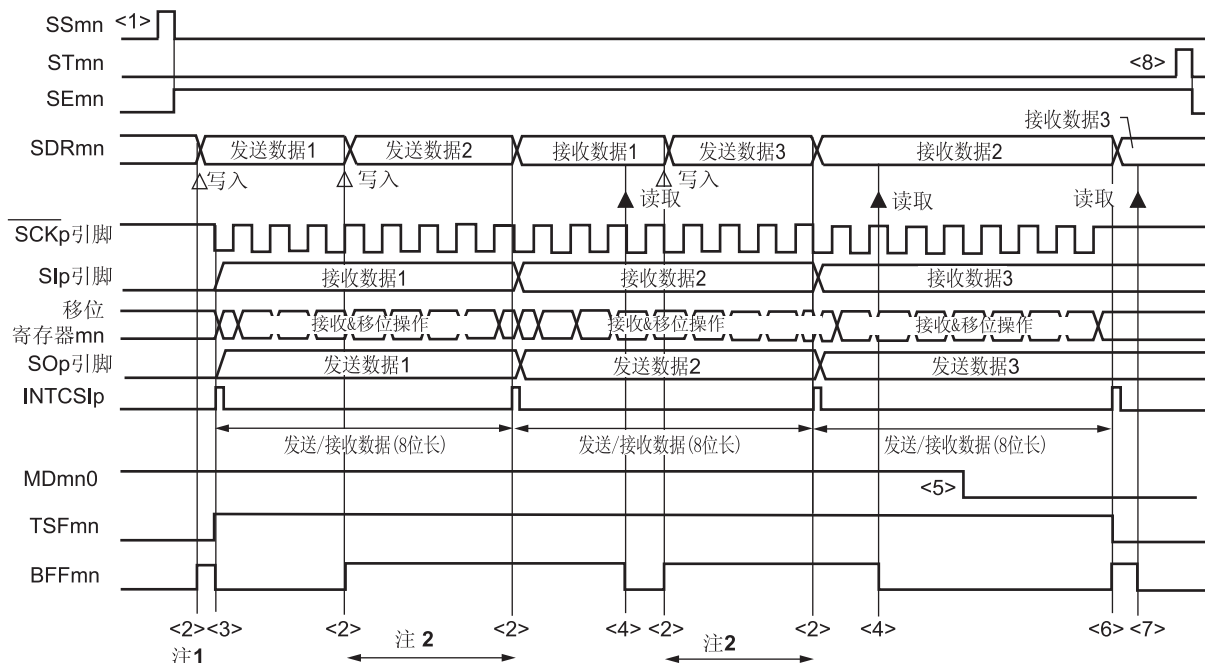
备注 m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)

图 15-46. 主发送/接收 (单发送/接收模式时)的流程图



(4) 处理流程 (连续发送/接收模式时)

图 15-47. 主发送/接收 (连续发送/接收模式时)的时序图
(类型 1: DAPmn = 0、CKPmn = 0)

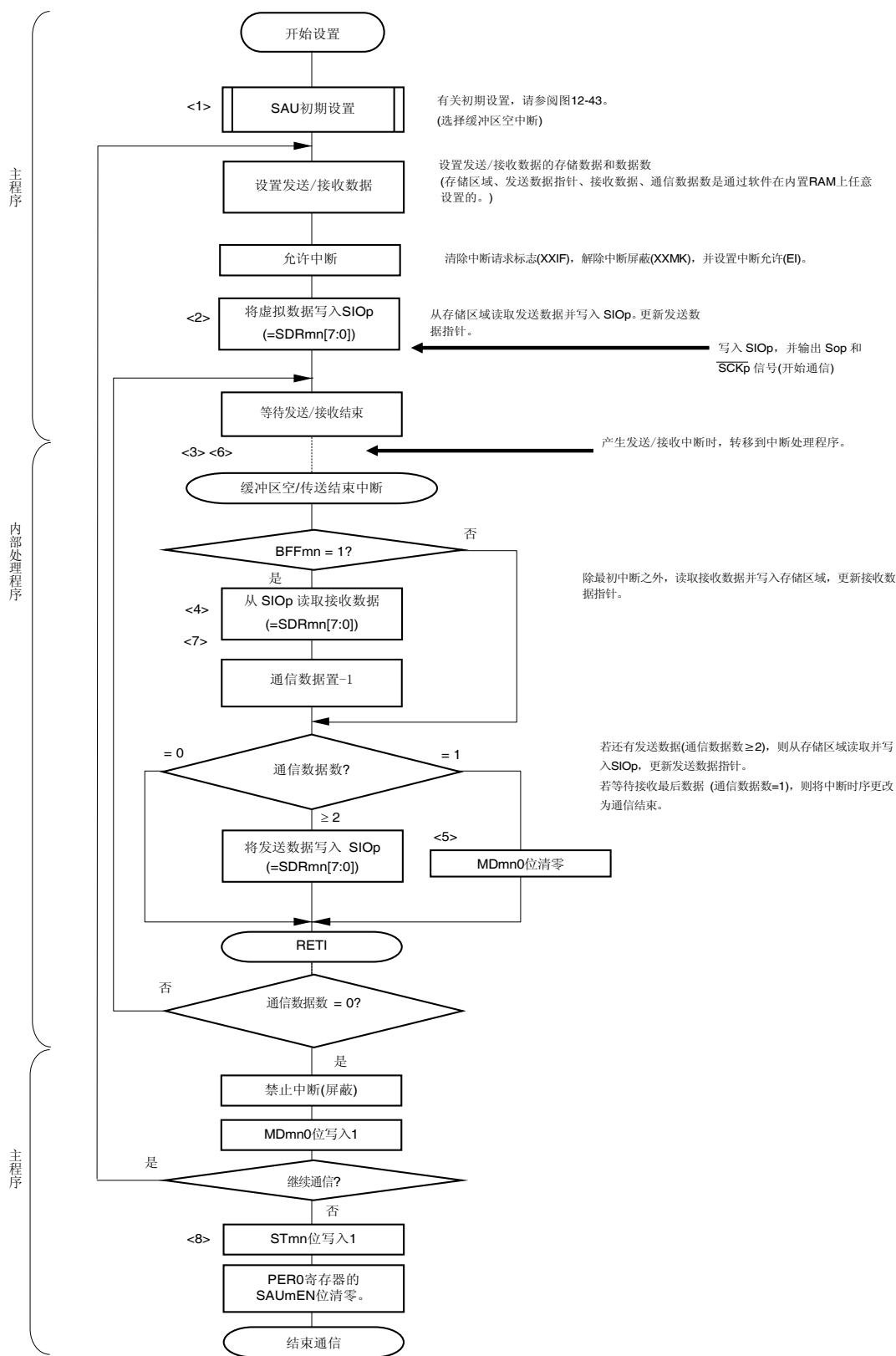


- 注 1. 如果在串行状态寄存器 mn (SSRmn)的 BFFmn 位为 1 时将发送数据写入 SDRmn 寄存器 (将有效数据存储于串行数据寄存器 mn (SDRmn)中), 则发送数据将被覆盖。
- 2. 在此过程中可以通过读取 SDRmn 寄存器来读取发送数据。此时, 传送操作不受影响。

注意事项 即使在操作过程中, 也可改写串行模式寄存器 mn (SMRmn)的 MDmn0 位。
然而必须在开始传送最后一位之前改写, 以便在最后发送数据的传送结束中断之前完成改写。

- 备注 1. 此图中的<1>至<8>对应于图 15-48. 主发送/接收 (连续发送/接收模式时)的流程图中的<1>至<8>。
- 2. m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)

图 15-48. 主发送/接收 (连续发送/接收模式时)的流程图



备注 此图中的<1>至<8>对应于图 15-47. 主发送/接收 (连续发送/接收模式时)的时序图中的<1>至<8>。

15.5.4 从发送

从发送是指，当接收来自另一个器件的传送时钟输入时，RL78/I1A 将数据发送至其他器件。

3线串行输入/输出	CSI00
目标通道	SAU0的通道0
使用引脚	SCK00, SO00
中断	INTCSI00
	可选择传送结束中断(单传送模式时)或缓冲器空中断(连续传送模式时)
错误检测标志	仅限溢出错误检测标志 (OVFmn)
传送数据长度	7或8位
传送速率	Max. $f_{mck}/6$ [Hz] ^{1,2} .
数据相位	可通过SCRmn寄存器的DAPmn位选择 <ul style="list-style-type: none"> • DAPmn = 0: 从开始串行时钟操作时开始数据输出。 • DAPmn = 1: 从开始串行时钟操作的半个时钟前开始数据输出。
时钟相位	可通过SCRmn寄存器的CKPmn位选择 <ul style="list-style-type: none"> • CKPmn = 0: 正相 • CKPmn = 1: 反相
数据方向	MSB或LSB优先

注 1. 输入至 SCK00 引脚的外部串行时钟是在内部通过采样使用，因此最大传送速率为 $f_{mck}/6$ [Hz]。通过设置 SPSm 寄存器，使该频率为最小 $f_{sck}/2$ （由 SDRmn 寄存器设置）。

2. 在满足以上条件及电特性中的外围功能特性(参阅第三十二章 电特性)的范围内使用此操作。

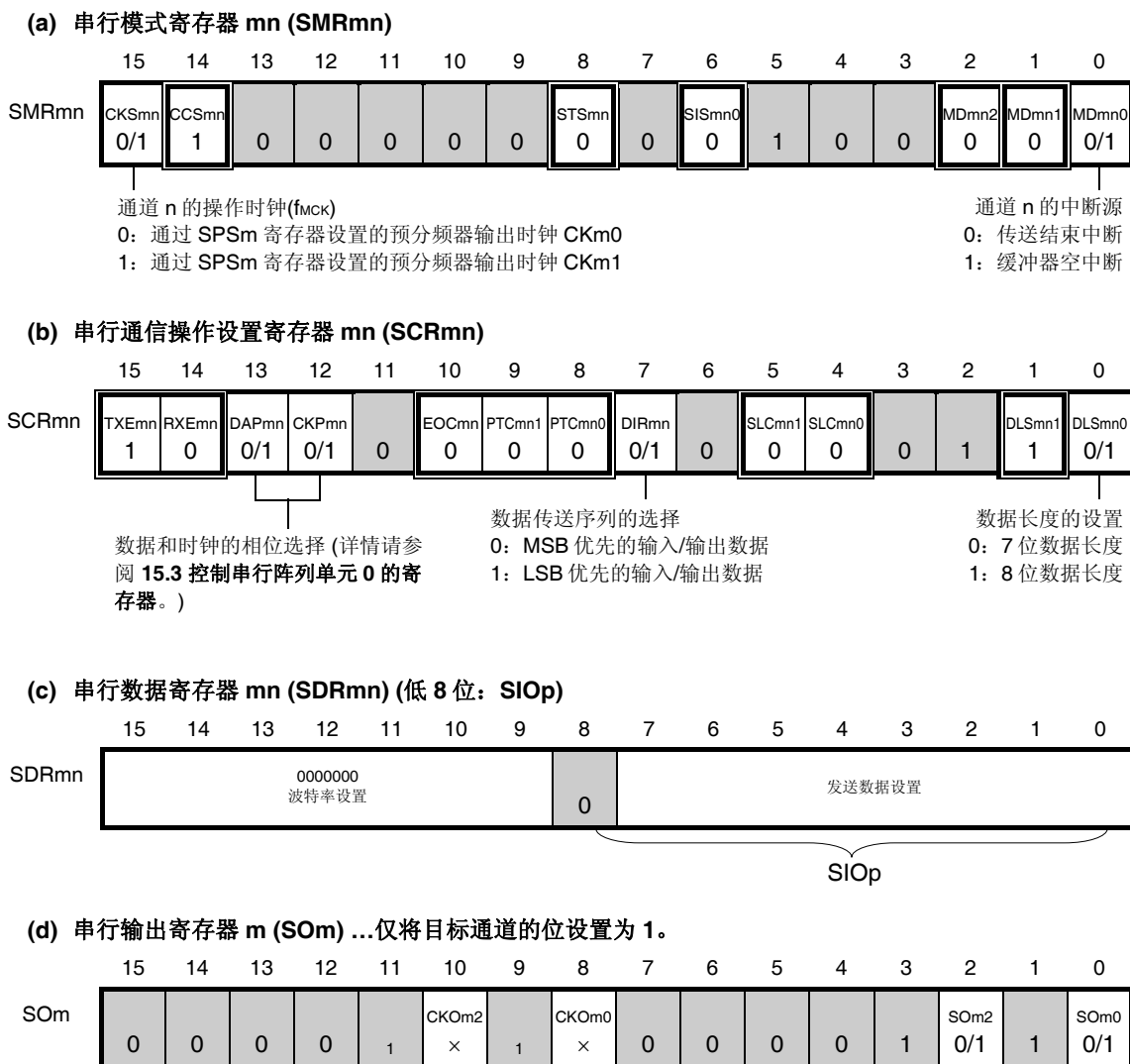
备注 1. f_{mck} : 目标通道的操作时钟频率

f_{sck} : 串行时钟频率

2. m: 单元编号(m = 0), n: 通道编号(n = 0)

(1) 寄存器设置

图 15-49. 3 线串行输入/输出(CSI00)的从发送时的寄存器设置内容示例(1/2)



- 备注 1. m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)
2. : 设置固定于 CSI 从传送模式, : 禁止设置 (设为初始值)
 x: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)
 0/1: 可根据用户的用途设为 0 或 1

图 15-49. 3 线串行输入/输出(CSI00)的从发送时的寄存器设置内容示例(2/2)

(e) 串行输出允许寄存器 m (SOEm) ... 仅将目标通道的位设置为 1。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOEm	0	0	0	0	0	0	0	0	0	0	0	0	0	SOEm2 0/1	0	SOEm0 0/1

(f) 串行通道开始寄存器 m (SSm) ... 仅将目标通道的位设置为 1。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSm	0	0	0	0	0	0	0	0	0	0	0	0	SSm3 ×	SSm2 ×	SSm1 ×	SSm0 0/1

- 备注 1. m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)
2. : 禁止设置 (设为初始值)
 x: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)
 0/1: 可根据客户的用途设为 0 或 1

(2) 操作步骤

图 15-50. 从发送的初始设定步骤



图 15-51. 停止从发送的步骤

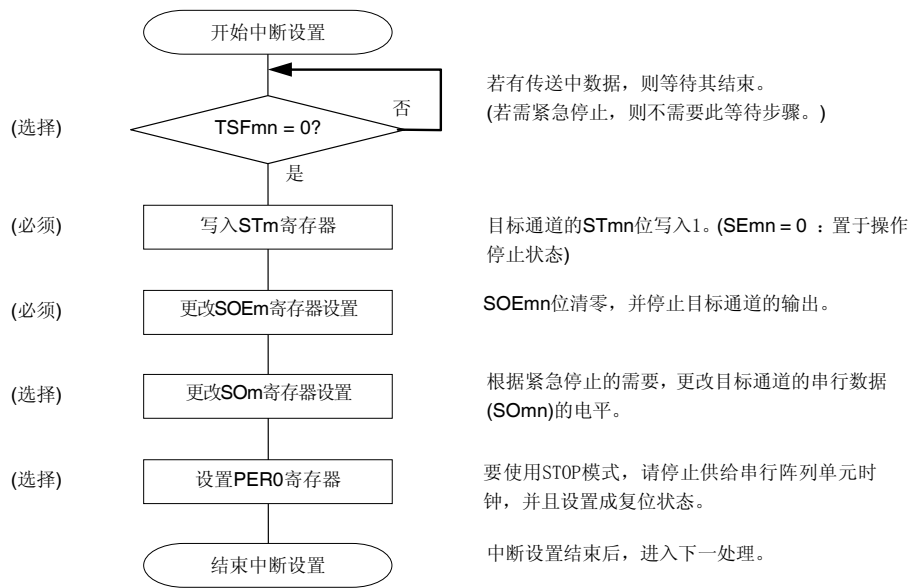


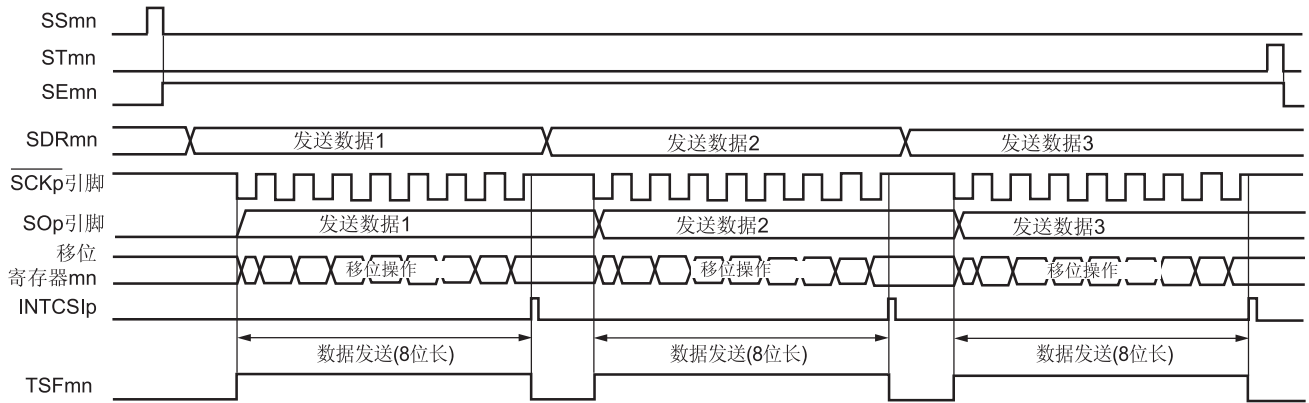
图 15-52. 恢复从发送的步骤



备注 在停止从发送的步骤中改写 PER0 而停止时钟供应时，请等待通信目标（主设备）停止或通信结束，然后执行初始设置，而不是重传设置。

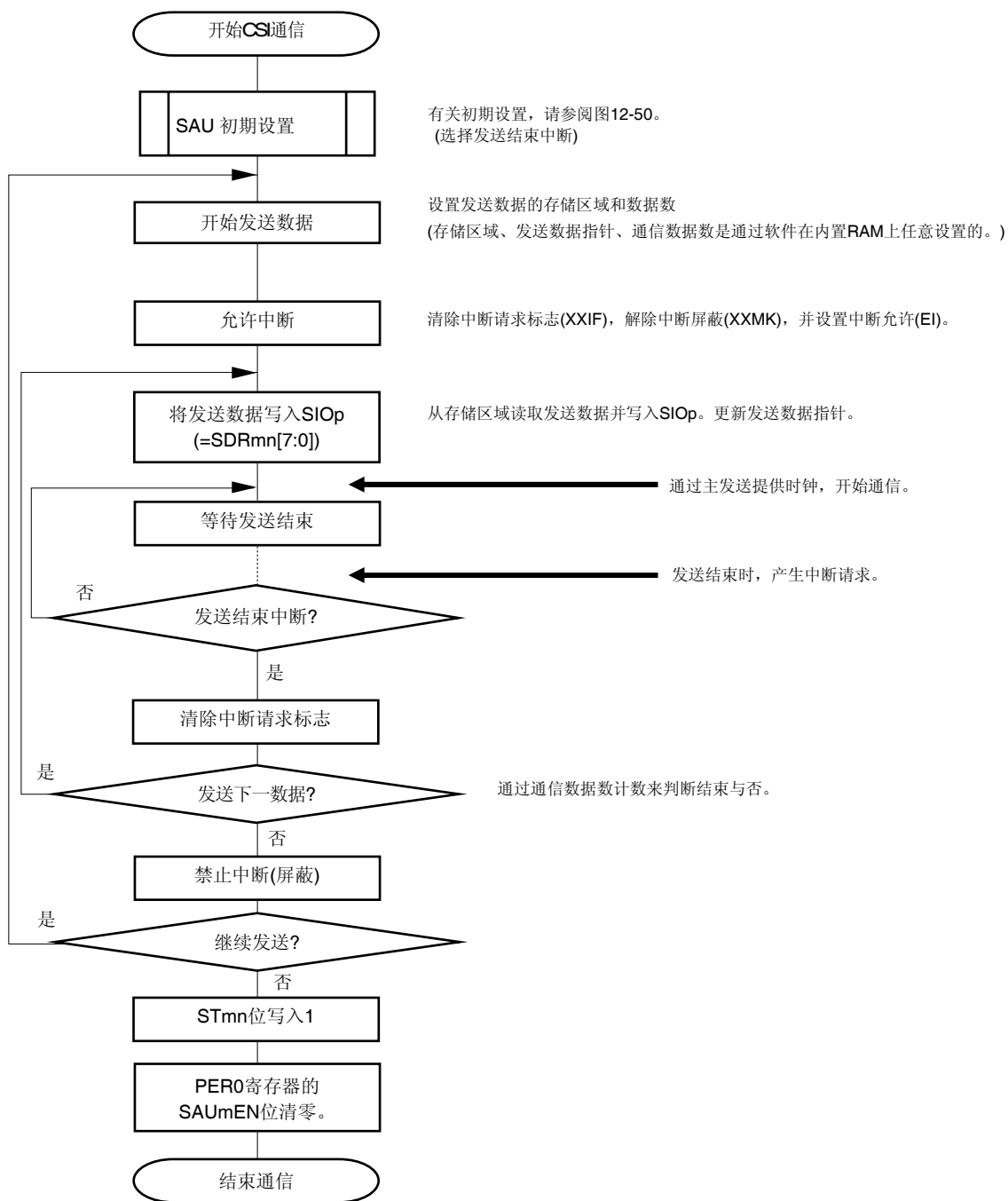
(3) 处理流程 (单发送模式时)

图 15-53. 从发送(单发送模式时)的时序图
(类型 1: DAPmn = 0、CKPmn = 0)



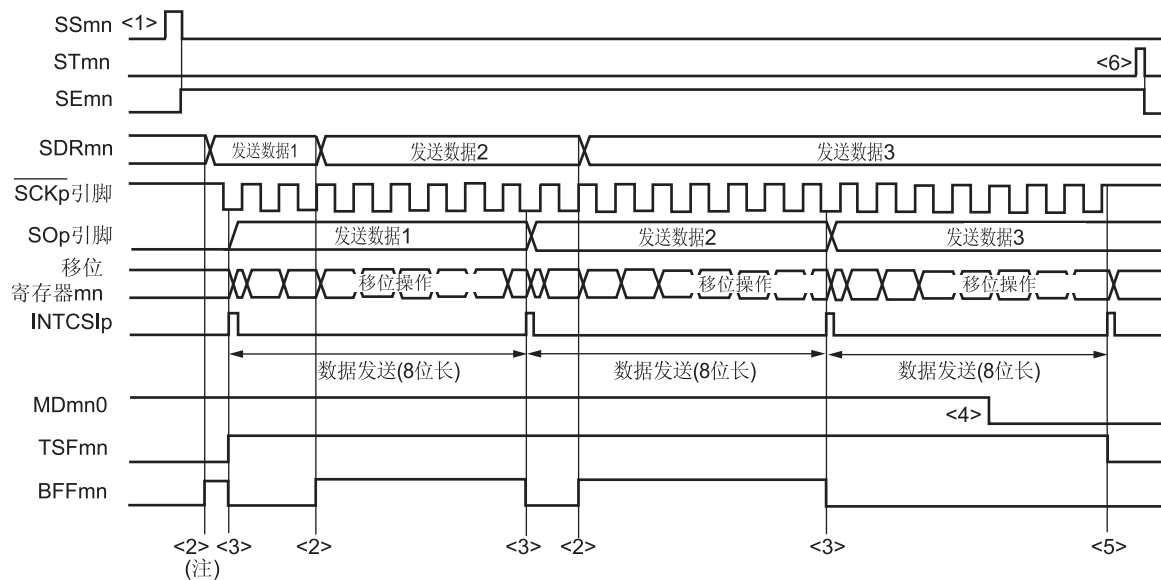
备注 m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)

图 15-54. 从发送(单发送模式时)的流程图



(4) 处理流程 (连续传送模式时)

图 15-55. 从发送(连续传送模式时)的时序图
(类型 1: DAPmn = 0、CKPmn = 0)

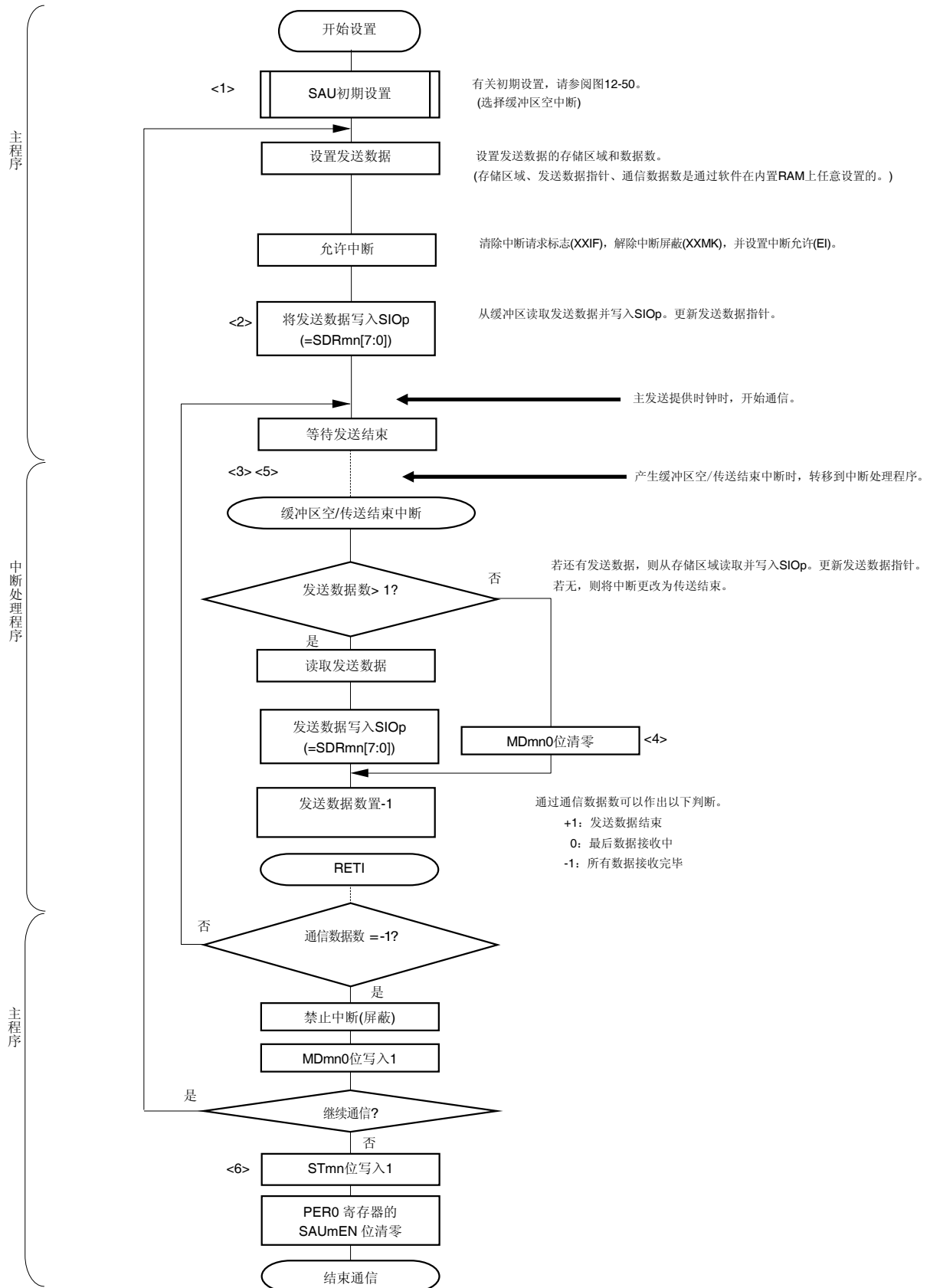


注 如果在串行状态寄存器 mn (SSRmn) 的 BFFmn 位为 1 时将发送数据写入 SDRmn 寄存器 (将有效数据存储于串行数据寄存器 mn (SDRmn) 中)，则发送数据将被覆盖。

注意事项 即使在操作过程中，也可改写串行模式寄存器 mn (SMRmn) 的 MDmn0 位。然而，要在开始传送最后一位之前改写。

备注 m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)

图 15-56. 从发送(连续传送模式时)的流程图



备注 此图中的<1>至<6>对应于图 15-55 从发送(连续发送模式时)的时序图中的<1>至<6>。

15.5.5 从接收

从接收是指，当接收来自另一个器件的传送时钟输入时，RL78/I1A 从其他器件接收数据。

3线串行输入/输出	CSI00
目标通道	SAU0的通道0
使用引脚	$\overline{\text{SCK00}}$, SI00
中断	INTCSI00
	仅限传送结束中断(禁止设置缓冲器空中断。)
错误检测标志	仅限溢出错误检测标志 (OVFmn)
传送数据长度	7或8位
传送速率	Max. $f_{\text{mck}}/6$ [Hz] ^{※1, 2}
数据相位	可通过SCRmn寄存器的DAPmn位选择 <ul style="list-style-type: none"> • DAPmn = 0: 从开始串行时钟操作时开始数据输入。 • DAPmn = 1: 从开始串行时钟操作的半个时钟前开始数据输入。
时钟相位	可通过SCRmn寄存器的CKPmn位选择 <ul style="list-style-type: none"> • CKPmn = 0: 正相 • CKPmn = 1: 反相
数据方向	MSB或LSB优先

注 1. 输入至 $\overline{\text{SCK00}}$ 引脚的外部串行时钟是在内部通过采样使用，因此最大传送速率为 $f_{\text{mck}}/6$ [Hz]。通过设置 SPSm 寄存器，使该频率为最小 $f_{\text{sc}}/2$ （由 SDRmn 寄存器设置）。

2. 在满足以上条件及电特性中的外围功能特性(参阅**第三十二章 电特性**)的范围内使用此操作。

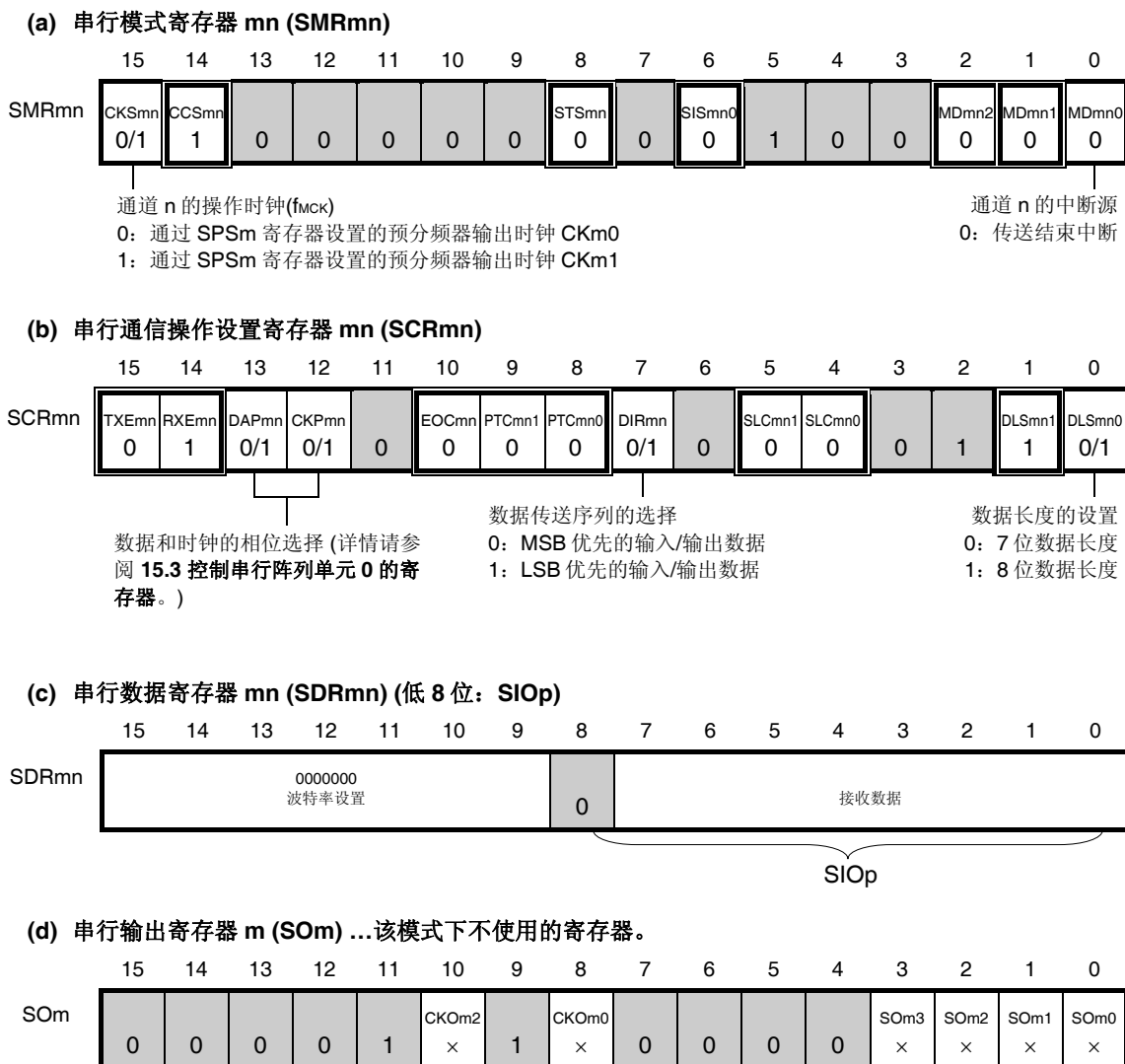
备注 1. f_{mck} : 目标通道的操作时钟频率

f_{sc} : 串行时钟频率

2. m: 单元编号(m = 0), n: 通道编号(n = 0)

(1) 寄存器设置

图 15-57. 3 线串行输入/输出 (CSI00)的从接收时的寄存器设置内容示例(1/2)



- 备注 1. m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)
2. □: 设置固定于 CSI 从发送模式, ■ 禁止设置 (设为初始值)
 ×: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)
 0/1: 可根据客户的用途设为 0 或 1

图 15-57. 3 线串行输入/输出 (CSI00)的从接收时的寄存器设置内容示例(2/2)

(e) 串行输出允许寄存器 m (SOEm) ...该模式下不使用的寄存器。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOEm	0	0	0	0	0	0	0	0	0	0	0	0	0	SOEm2 ×	0	SOEm0 ×

(f) 串行通道开始寄存器 m (SSm) ...仅将目标通道的位设置为 1。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSm	0	0	0	0	0	0	0	0	0	0	0	0	SSm3 ×	SSm2 ×	SSm1 ×	SSm0 0/1

- 备注 1. m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)
2. : 禁止设置(设置为初始值)
 x: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)
 0/1: 可根据客户的用途设为 0 或 1

(2) 操作步骤

图 15-58. 从接收的初始设定步骤

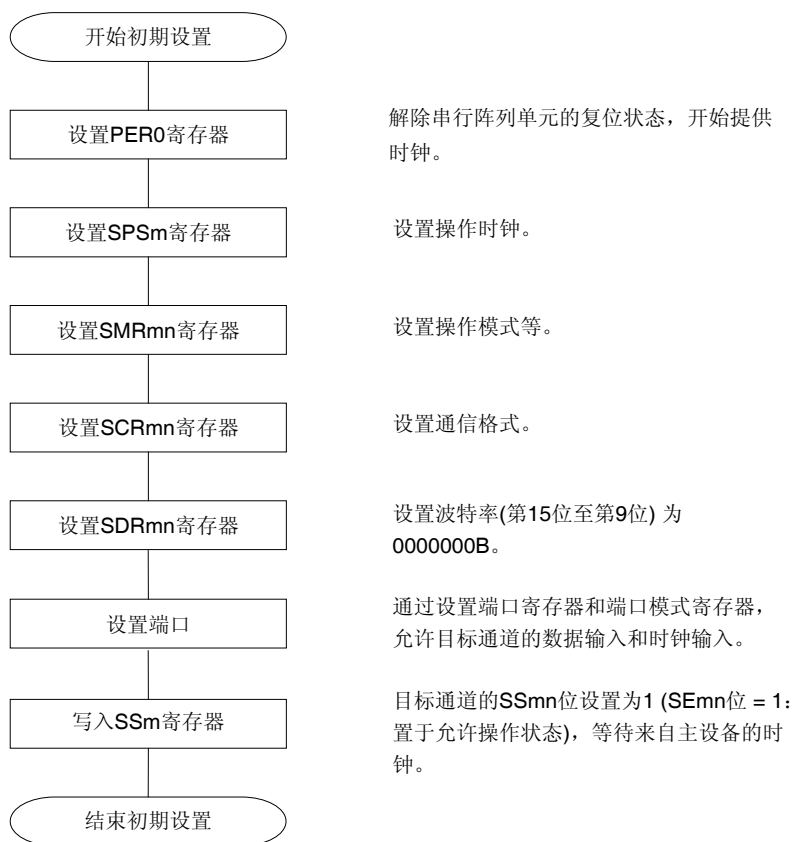


图 15-59. 停止从接收的步骤

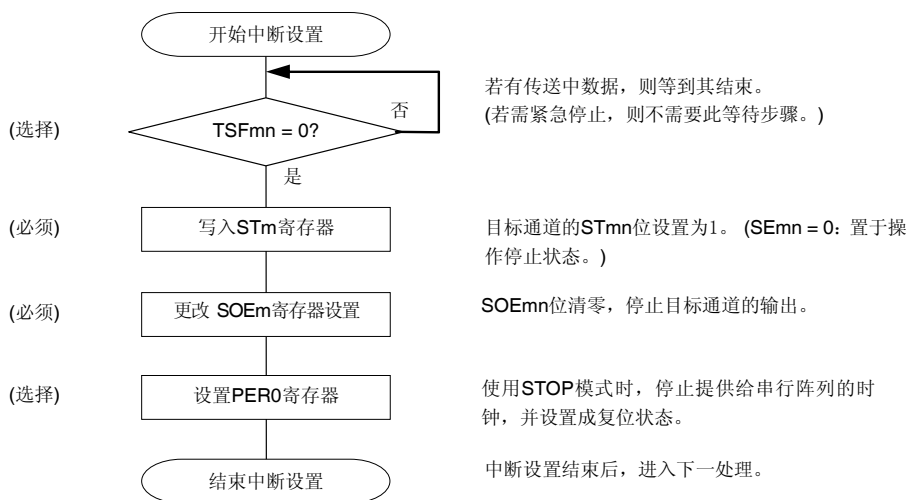
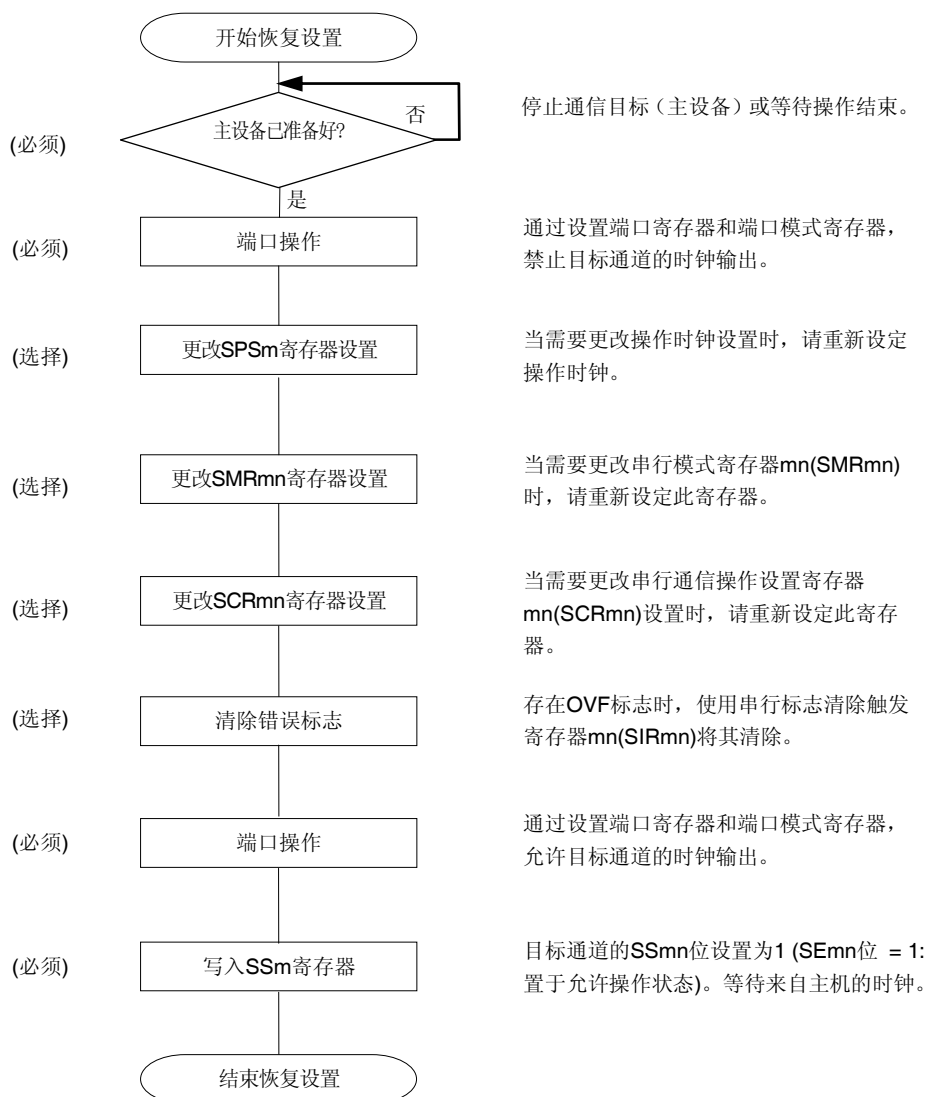


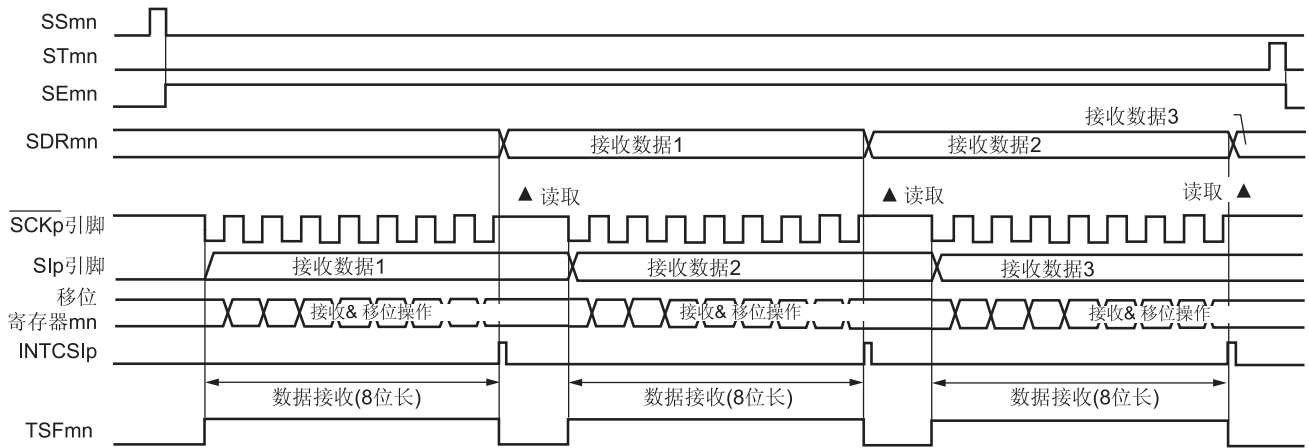
图 15-60. 恢复从接收的步骤



备注 在停止从接收的步骤中改写 PER0 而停止时钟供应时, 请等待通信目标 (主设备) 停止或者通信结束, 然后执行初始设置, 而不是重传设置。

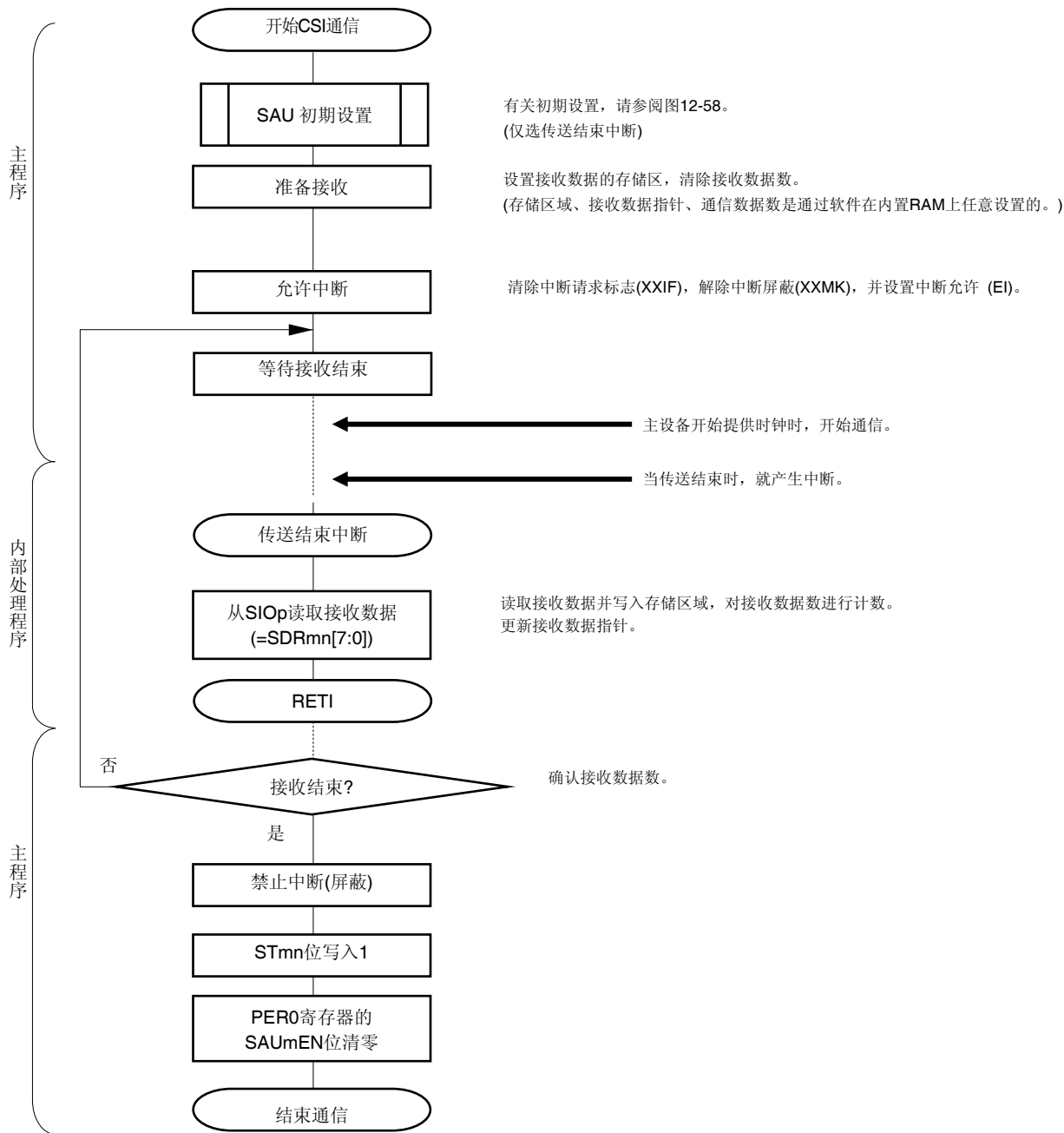
(3) 处理流程 (单接收模式时)

图 15-61. 从接收(单接收模式时)的时序图
(类型 1: DAPmn = 0、CKPmn = 0)



备注 m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)

图 15-62. 从接收(单接收模式时)的流程图



15.5.6 从发送/接收

从发送/接收是指，当接收来自另一个器件的传送时钟输入时，RL78/I1A 发送数据至其他器件或者从其他器件接收数据。

3线串行输入/输出	CSI00
目标通道	SAU0的通道0
使用引脚	$\overline{\text{SCK00}}$, SI00, SO00
中断	INTCSI00
	可选择传送结束中断(单传送模式时)或缓冲器空中断(连续传送模式时)
错误检测标志	仅限溢出错误检测标志 (OVFmn)
传送数据长度	7或8位
传送速率	Max. $f_{\text{MCK}}/6$ [Hz] ^{1,2}
数据相位	可通过SCRmn寄存器的DAPmn位选择 <ul style="list-style-type: none"> • DAPmn = 0: 从开始串行时钟操作时开始数据输入/输出 • DAPmn = 1: 从开始串行时钟操作的半个时钟前开始数据输入/输出。
时钟相位	可通过SCRmn寄存器的CKPmn位选择 <ul style="list-style-type: none"> • CKPmn = 0: 正相 • CKPmn = 1: 反相
数据方向	MSB或LSB优先

注 1. 输入至 $\overline{\text{SCK00}}$ 引脚的外部串行时钟是在内部通过采样使用，因此最大传送速率为 $f_{\text{MCK}}/6$ [Hz]。通过设置 SPSm 寄存器，使该频率为最小 $f_{\text{SCK}}/2$ （由 SDRmn 寄存器设置）。

2. 在满足以上条件及电特性中的外围功能特性(参阅第三十二章 电特性)的范围内使用此操作。

备注 1. f_{MCK} : 目标通道的操作时钟频率

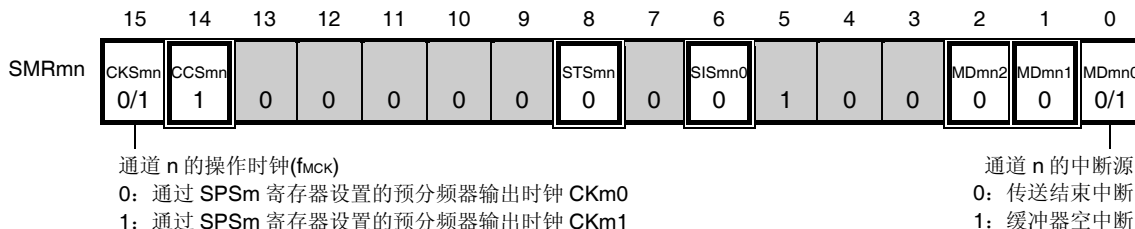
f_{CLK} : 串行时钟频率

2. m: 单元编号(m = 0), n: 通道编号(n = 0)

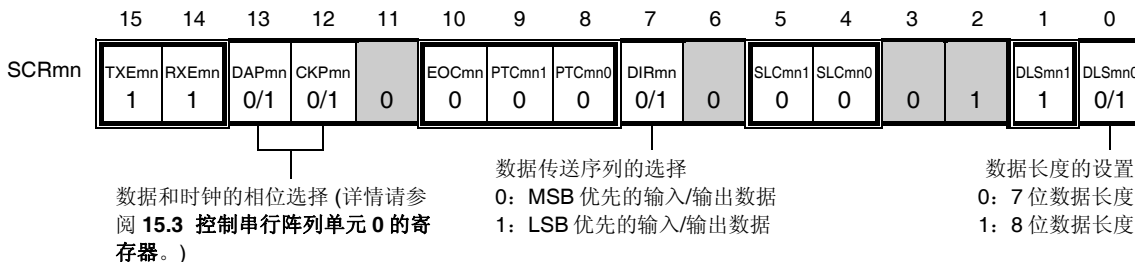
(1) 寄存器设置

图 15-63. 3 线串行输入/输出 (CSI00)的从发送/接收时的寄存器设置内容示例(1/2)

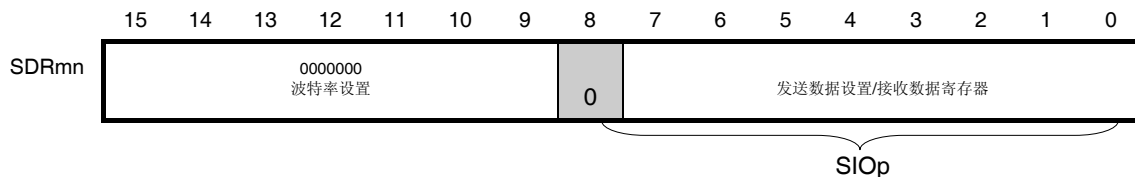
(a) 串行模式寄存器 mn (SMRmn)



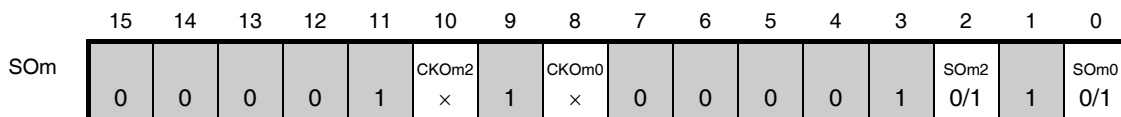
(b) 串行通信操作设置寄存器 mn (SCRmn)



(c) 串行数据寄存器 mn (SDRmn) (低 8 位: SIOp)



(d) 串行输出寄存器 m (SOM) ... 仅将目标通道的位设置为 1。



注意事项 必须在启动来自主机的时钟之前，把发送数据设置至 SIOp 寄存器。

- 备注**
- m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)
 - : 设置固定于 CSI 从发送/接收模式,
 ■: 禁止设置 (设为初始值)
 x: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)
 0/1: 可根据用户的用途设为 0 或 1

图 15-63. 3 线串行输入/输出 (CSI00)的从发送/接收时的寄存器设置内容示例(2/2)

(e) 串行输出允许寄存器 m (SOEm) ...仅将目标通道的位设置为 1。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOEm	0	0	0	0	0	0	0	0	0	0	0	0	0	SOEm2 0/1	0	SOEm0 0/1

(f) 串行通道开始寄存器 m (SSm) ...仅将目标通道的位设置为 1。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSm	0	0	0	0	0	0	0	0	0	0	0	0	SSm3 ×	SSm2 ×	SSm1 ×	SSm0 0/1

- 备注 1. m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)
2. : 禁止设置 (设为初始值)
 x: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)
 0/1: 可根据客户的用途设为 0 或 1

(2) 操作步骤

图 15-64. 从发送/接收的初始设定步骤

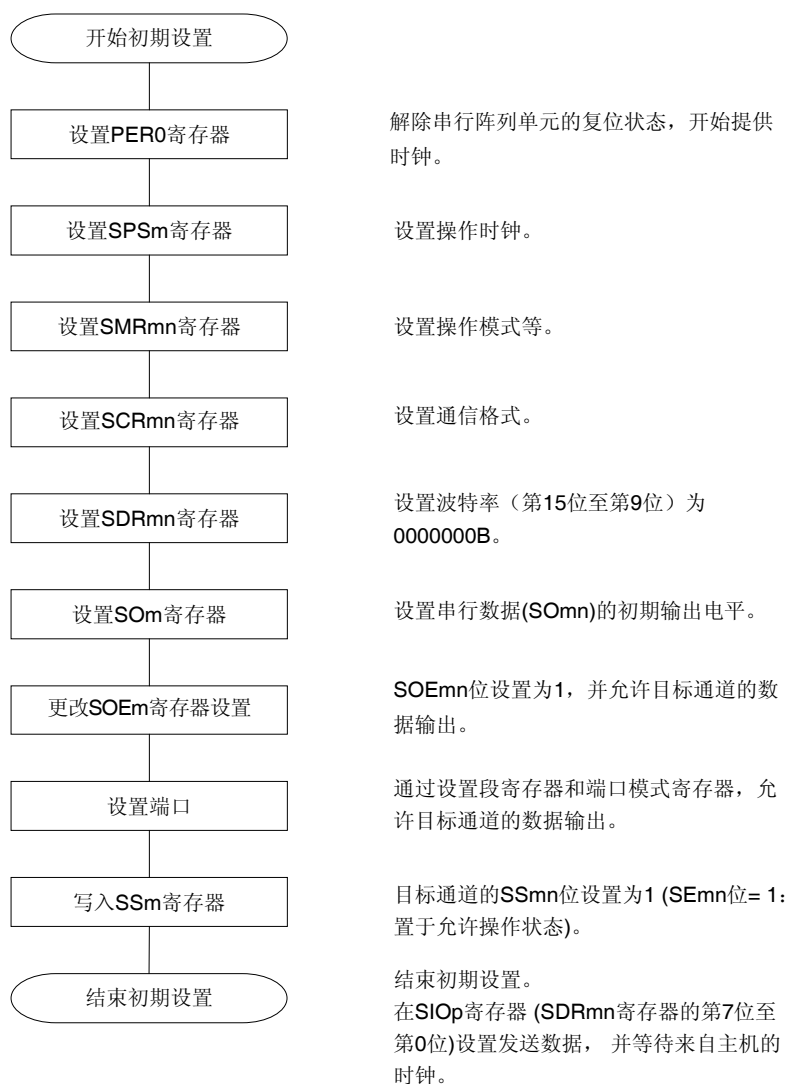


图 15-65. 停止从发送/接收的步骤

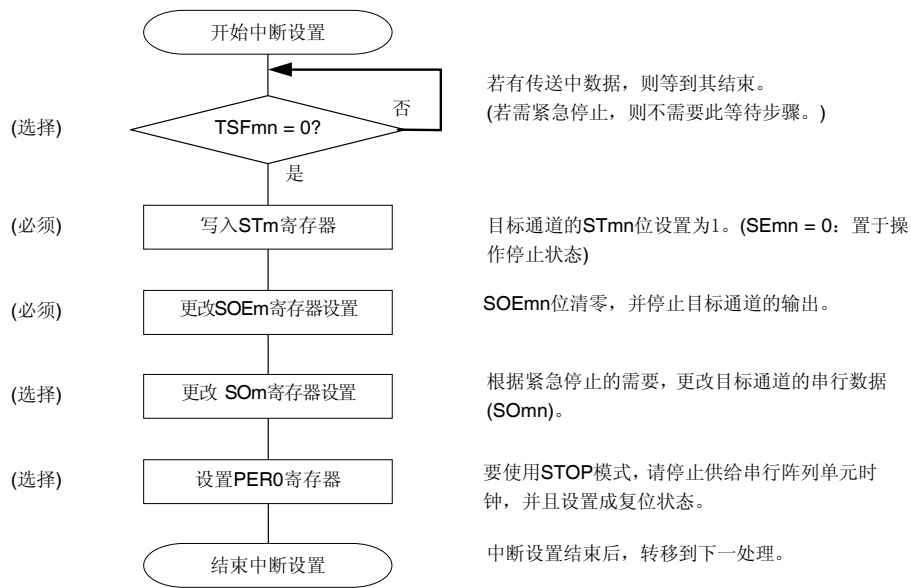


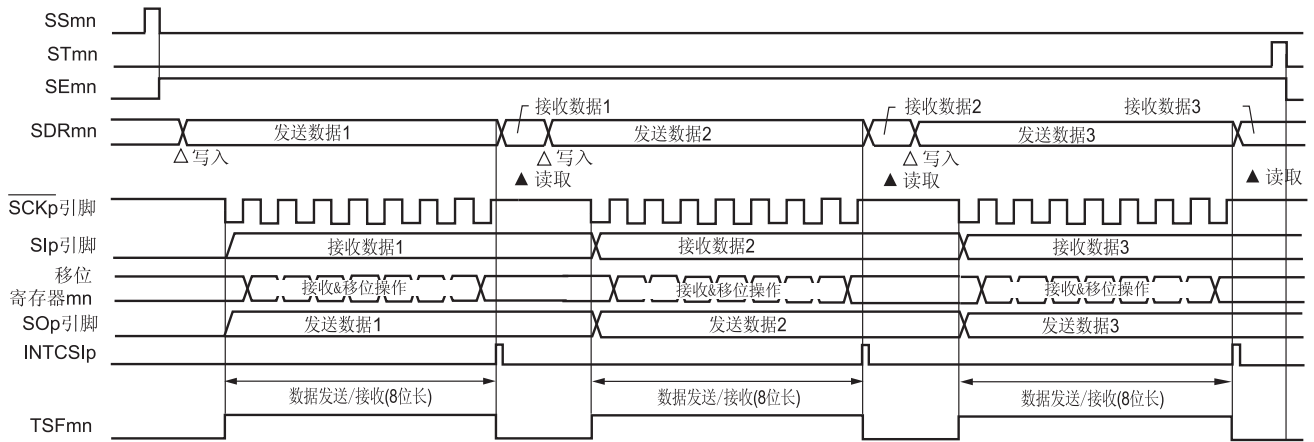
图 15-66. 恢复从发送/接收的步骤



- 注意事项**
1. 必须在启动来自主机的时钟之前，把发送数据设置至 SIOp 寄存器。
 2. 在停止从发送/接收的步骤中改写 PER0 而停止时钟供应时，请等待通信目标（主设备）停止或者通信结束，然后执行初始设置，而不是重传设置。

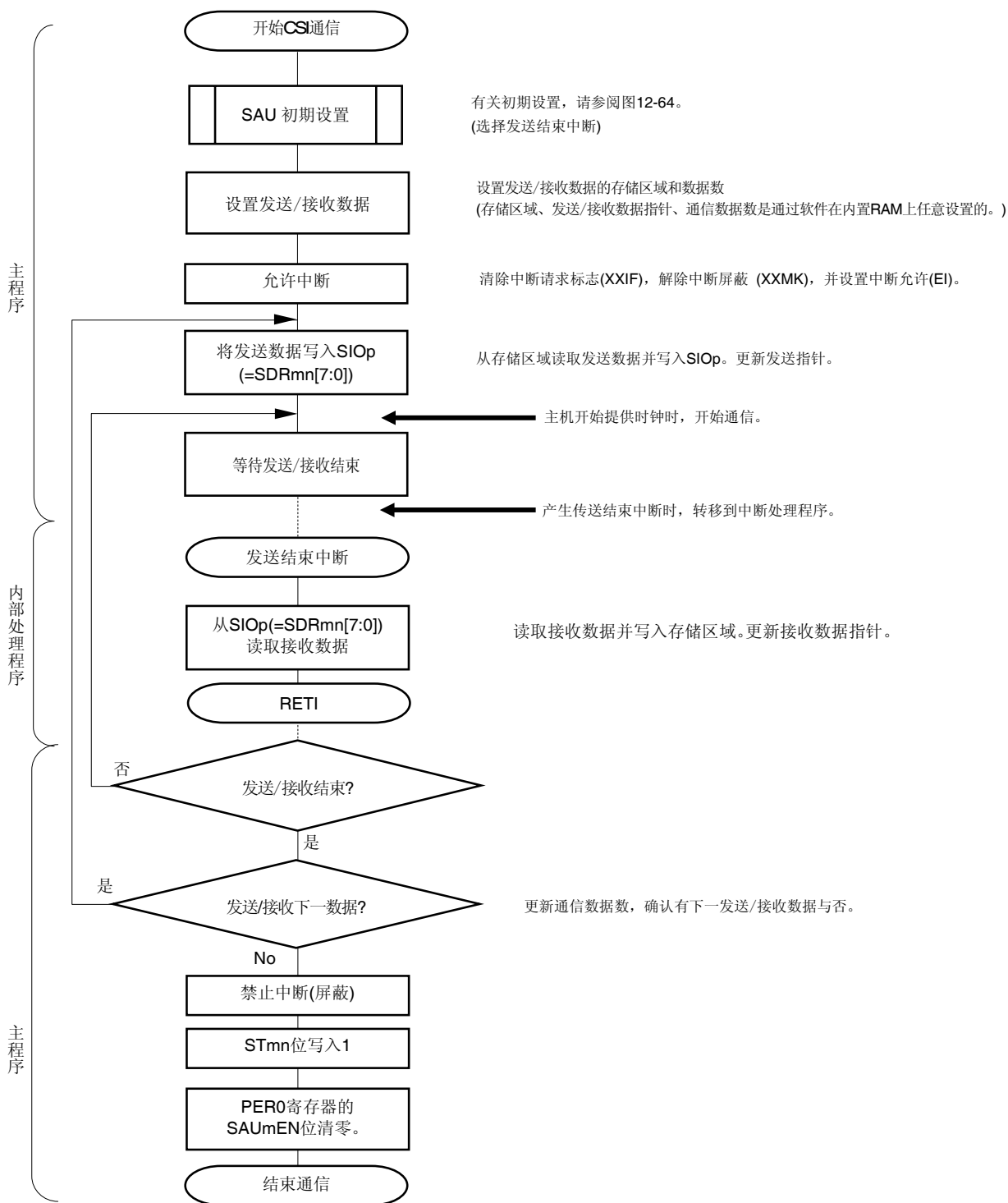
(3) 处理流程 (单发送/接收模式时)

图 15-67. 从发送/接收(单发送/接收模式时)的时序图
(类型 1: DAPmn = 0、CKPmn = 0)



备注 m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)

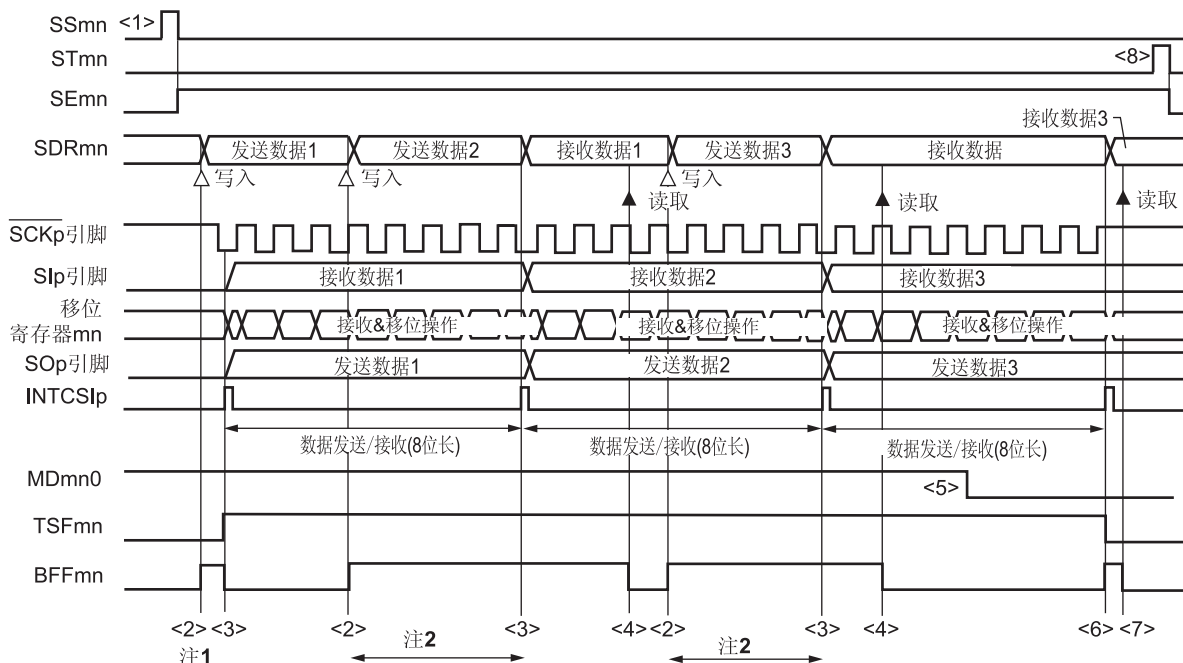
图 15-68. 从发送/接收(单发送/接收模式时)的流程图



注意事项 必须在启动来自主机的时钟之前, 把发送数据设置至 SIOp 寄存器。

(4) 处理流程 (连续发送/接收模式时)

图 15-69. 从发送/接收(连续发送/接收模式时)的时序图
(类型 1: DAPmn = 0、CKPmn = 0)

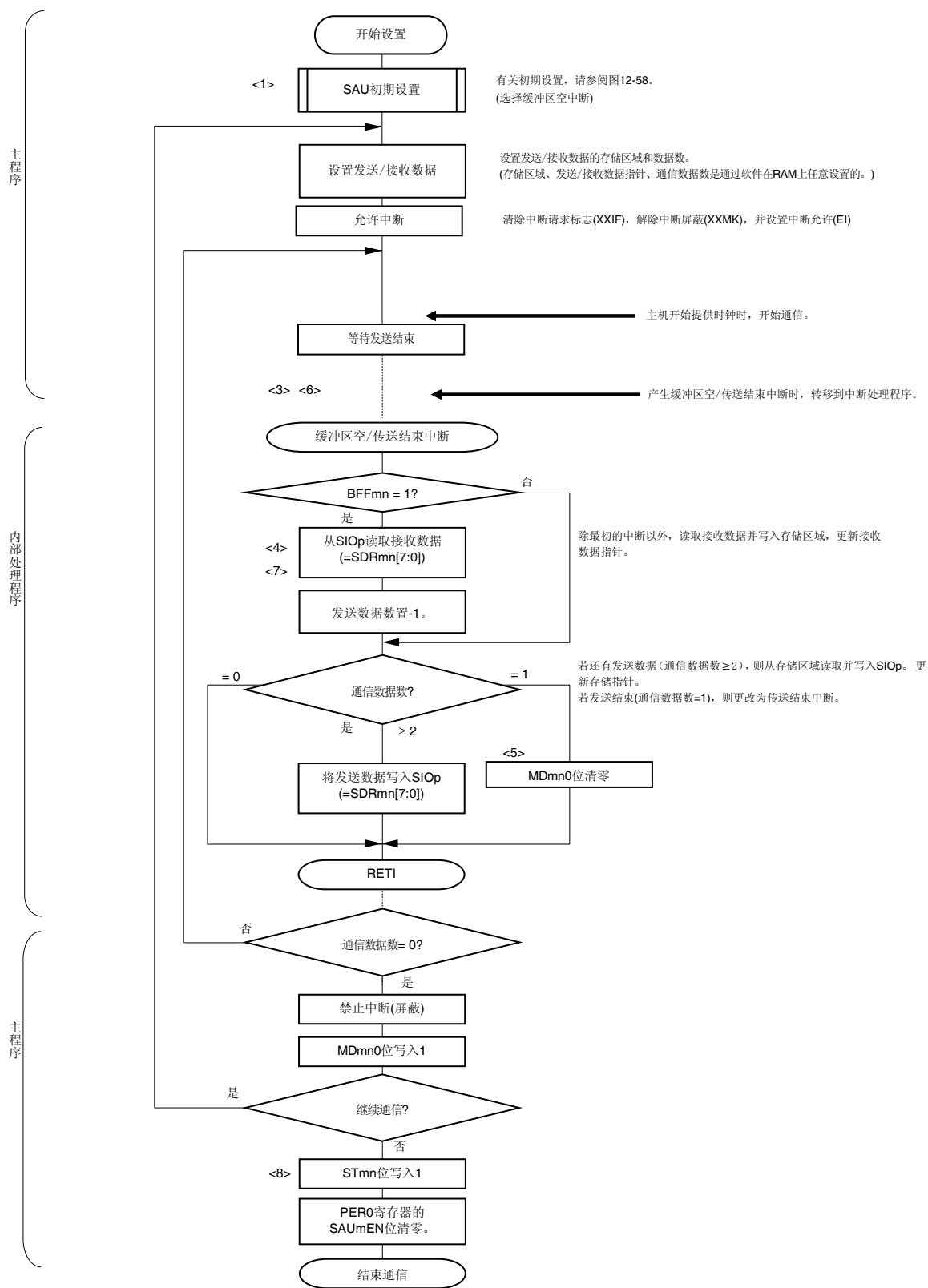


- 注 1. 如果在串行状态寄存器 mn (SSRmn)的 BFFmn 位为 1 时将发送数据写入 SDRmn 寄存器 (将有效数据存储于串行数据寄存器 mn (SDRmn)中), 则发送数据将被覆盖。
- 2. 在此过程中可以通过读取 SDRmn 寄存器来读取发送数据。此时, 传送操作不受影响。

注意事项 即使在操作过程中, 也可改写串行模式寄存器 mn (SMRmn)的 MDmn0 位。然而必须在开始传送最后一位之前改写, 以便在最后发送数据的传送结束中断之前完成改写。

- 备注 1. 此图中的<1>至<8>对应于图 15-70 从发送/接收(连续发送/接收模式时)的流程图中的<1>至<8>。
- 2. m: 单元编号(m = 0), n: 通道编号(n = 0), p: CSI 编号(p = 00)

图 15-70. 从发送/接收(连续发送/接收模式时)的流程图



注意事项 必须在启动来自主机的时钟之前, 把发送数据设置至 SIOp 寄存器。

备注 此图中的<1>至<8>对应于图 15-69. 从发送/接收(连续发送/接收模式时)的时序图中的<1>至<8>。

15.5.7 SNOOZE模式功能 (仅限CSI00)

利用 SNOOZE 模式，在 STOP 模式下检测到 $\overline{\text{SCKp}}$ 引脚输入时可以执行 CSI 接收操作。通常在 STOP 模式下 CSI 停止通信。但是在此模式下，可以在检测到 $\overline{\text{SCKp}}$ 引脚输入时不启动 CPU 就执行数据接收。仅限 CSI00 可以设置 SNOOZE 模式。

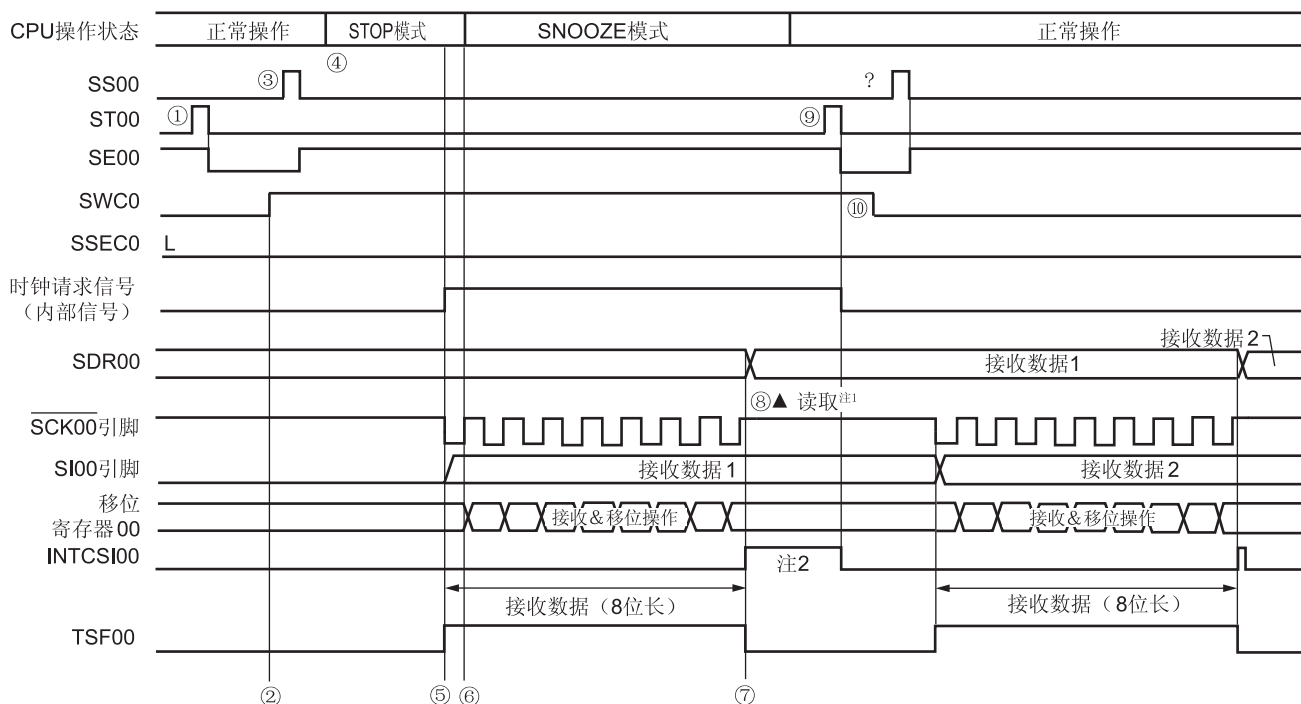
使用 SNOOZE 模式功能时，须在切换至 STOP 模式之前将串行待机控制寄存器 m (SSCm)的 SWCm 位设置为 1。

注意事项 1. 仅限选择高速片上振荡器时钟作为 f_{CLK}时，才可以使用 SNOOZE 模式。

2. 在 SNOOZE 模式下使用 CSIp 时的最大传送率为 1 Mbps。

(1) SNOOZE 模式操作 (一次启动)

图 15-71. SNOOZE 模式操作(一次启动)时的时序图(类型 1: DAPmn = 0、CKPmn = 0)



注 1. 仅当 SWCm = 1 且检测到 $\overline{\text{SCKp}}$ 引脚输入的下一个边沿之前，读取接收数据。

2. 在 SWCm 被清除为 0 时或者检测到 $\overline{\text{SCKp}}$ 引脚输入的下一个边沿时，传送结束中断 (INTCSIp)将被清除。

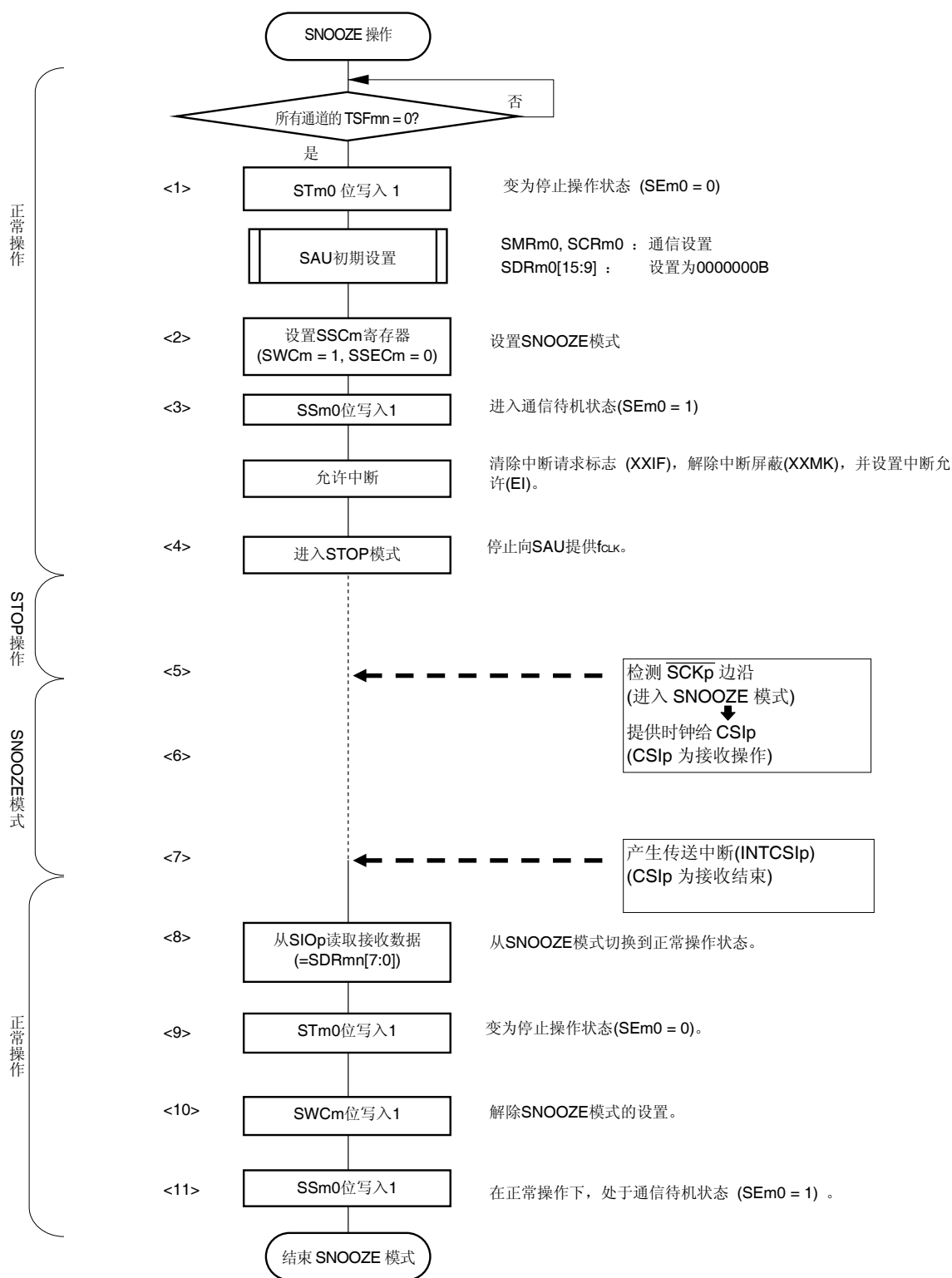
注意事项 在切换至 SNOOZE 模式之前，或者在 SNOOZE 模式下的接收操作完成之后，必须将 STm0 位设为 1 (清除 SEm0 位，并停止操作)。

并且在完成接收操作之后，将 SWCm 位清除为 0 (释放 SNOOZE 模式)。

备注 1. 此图中的<1>至<11>对应于图 15-72. SNOOZE 模式操作(一次启动)时的流程图中的<1>至<11>。

2. m: 单元编号(m = 0), p: CSI 编号(p = 00)

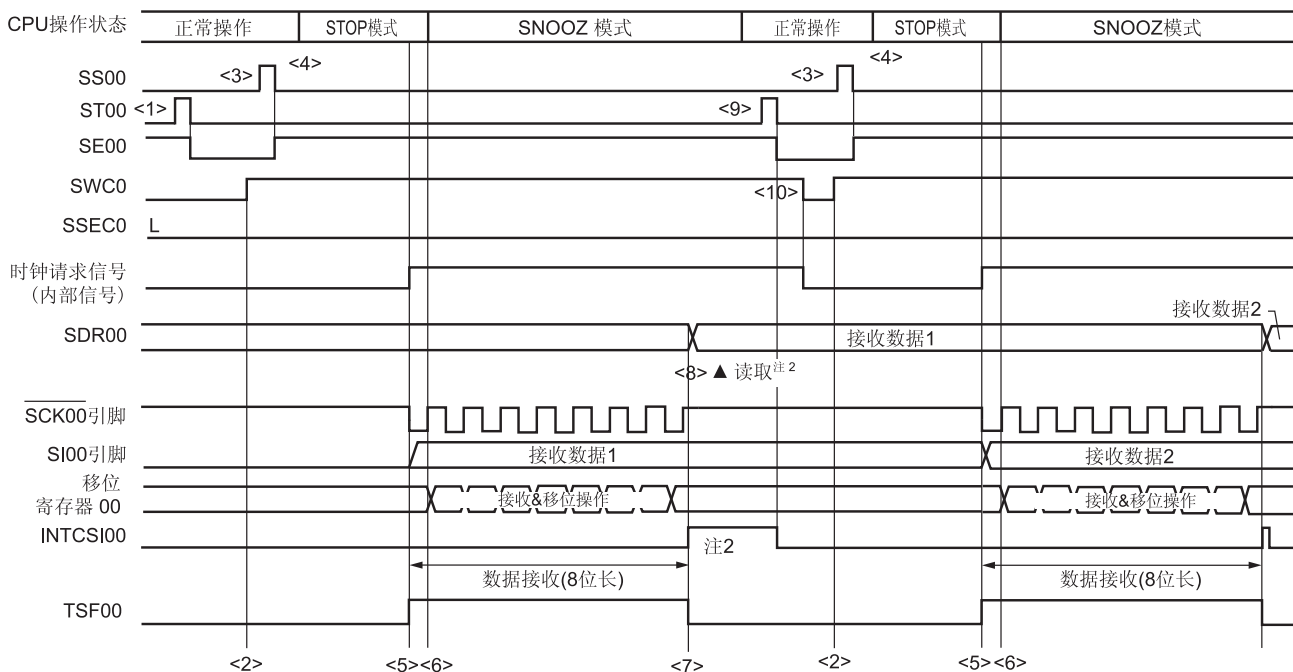
图 15-72. SNOOZE 模式操作(一次启动)时的流程图



备注 1. 此图中的<1>至<11>对应于图 15-71. SNOOZE 模式操作(一次启动)时的时序图中的<1>至<11>。
2. m: 单元编号(m = 0), p: CSI 编号(p = 00)

(2) SNOOZE 模式操作 (连续启动)

图 15-73. SNOOZE 模式操作(连续启动)时的时序图(类型 1: DAPmn = 0、CKPmn = 0)

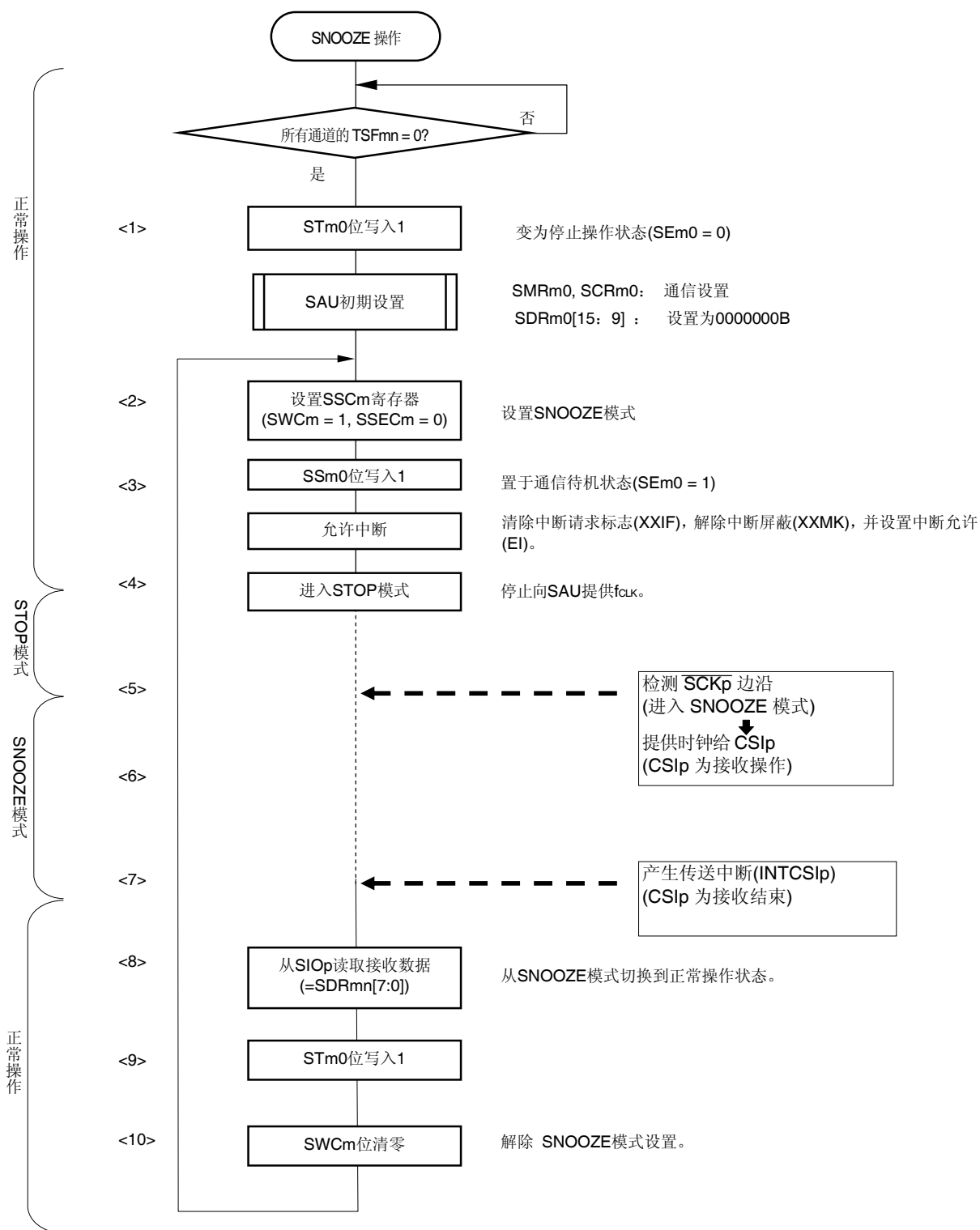


- 注 1. 仅当 SWCm = 1 且检测到 SCKp 引脚输入的下一个边沿之前，读取接收数据。
- 2. 在 SWCm 被清除至 0 时或者检测到 SCKp 引脚输入的下一个边沿时，传送结束中断 (INTCSIp) 将被清除。

注意事项 在切换至 SNOOZE 模式之前，或者在 SNOOZE 模式下的接收操作完成之后，必须将 STm0 位设为 1 (清除 SEm0 位，并停止操作)。并且在完成接收操作之后，将 SWCm 位清除为 0 (释放 SNOOZE 模式)。

- 备注 1. 此图中的<1>至<10>对应于图 15-74. SNOOZE 模式操作(连续启动)时的流程图中的<1>至<10>。
- 2. m: 单元编号(m = 0), p: CSI 编号(p = 00)

图 15-74. SNOOZE 模式操作(连续启动)时的流程图



备注 1. 此图中的<1>至<10>对应于图 15-73. SNOOZE 模式操作(连续启动)时的时序图中的<1>至<10>。
2. m: 单元编号(m = 0), p: CSI 编号(p = 00)

15.5.8 计算传送时钟频率

三线串行输入/输出 (CSI00)通信的传送时钟频率可以通过以下表达式计算。

(1) 主设备

$$(\text{传送时钟频率}) = \{ \text{目标通道的操作时钟}(f_{\text{MCK}}) \text{ 频率} \} \div (\text{SDRmn}[15:9] + 1) \div 2 \text{ [Hz]}$$

(2) 从设备

$$(\text{传送时钟频率}) = \{ \text{由主机提供的串行时钟}(SCK) \text{ 频率} \}^{\#} \text{ [Hz]}$$

注 允许的最大传送时钟频率为 $f_{\text{MCK}}/6$ 。

备注 SDRmn[15:9]的值为串行数据寄存器 mn (SDRmn)的位 15 至位 9 的值(0000000B 至 1111111B)，即 0 至 127。

操作时钟(f_{MCK}) 取决于串行时钟选择寄存器 m (SPSm) 以及串行模式寄存器 mn (SMRmn)的位 15 (CKSmn)。

表 15-2. 用于 3 线串行输入/输出的操作时钟的选择

SMRmn 寄存器	SPSm 寄存器								操作时钟(f _{CLK}) [#]	
	CKSmn	PRS m13	PRS m12	PRS m11	PRS m10	PRS m03	PRS m02	PRS m01	PRS m00	f _{CLK} = 32 MHz
0	X	X	X	X	0	0	0	0	f _{CLK}	32 MHz
	X	X	X	X	0	0	0	1	f _{CLK} /2	16 MHz
	X	X	X	X	0	0	1	0	f _{CLK} /2 ²	8 MHz
	X	X	X	X	0	0	1	1	f _{CLK} /2 ³	4 MHz
	X	X	X	X	0	1	0	0	f _{CLK} /2 ⁴	2 MHz
	X	X	X	X	0	1	0	1	f _{CLK} /2 ⁵	1 MHz
	X	X	X	X	0	1	1	0	f _{CLK} /2 ⁶	500 kHz
	X	X	X	X	0	1	1	1	f _{CLK} /2 ⁷	250 kHz
	X	X	X	X	1	0	0	0	f _{CLK} /2 ⁸	125 kHz
	X	X	X	X	1	0	0	1	f _{CLK} /2 ⁹	62.5 kHz
	X	X	X	X	1	0	1	0	f _{CLK} /2 ¹⁰	31.25 kHz
	X	X	X	X	1	0	1	1	f _{CLK} /2 ¹¹	15.63 kHz
	X	X	X	X	1	1	0	0	f _{CLK} /2 ¹²	7.81 kHz
	X	X	X	X	1	1	0	1	f _{CLK} /2 ¹³	3.91 kHz
	X	X	X	X	1	1	1	0	f _{CLK} /2 ¹⁴	1.95 kHz
X	X	X	X	1	1	1	1	f _{CLK} /2 ¹⁵	977 Hz	
1	0	0	0	0	X	X	X	X	f _{CLK}	32 MHz
	0	0	0	1	X	X	X	X	f _{CLK} /2	16 MHz
	0	0	1	0	X	X	X	X	f _{CLK} /2 ²	8 MHz
	0	0	1	1	X	X	X	X	f _{CLK} /2 ³	4 MHz
	0	1	0	0	X	X	X	X	f _{CLK} /2 ⁴	2 MHz
	0	1	0	1	X	X	X	X	f _{CLK} /2 ⁵	1 MHz
	0	1	1	0	X	X	X	X	f _{CLK} /2 ⁶	500 kHz
	0	1	1	1	X	X	X	X	f _{CLK} /2 ⁷	250 kHz
	1	0	0	0	X	X	X	X	f _{CLK} /2 ⁸	125 kHz
	1	0	0	1	X	X	X	X	f _{CLK} /2 ⁹	62.5 kHz
	1	0	1	0	X	X	X	X	f _{CLK} /2 ¹⁰	31.25 kHz
	1	0	1	1	X	X	X	X	f _{CLK} /2 ¹¹	15.63 kHz
	1	1	0	0	X	X	X	X	f _{CLK} /2 ¹²	7.81 kHz
	1	1	0	1	X	X	X	X	f _{CLK} /2 ¹³	3.91 kHz
	1	1	1	0	X	X	X	X	f _{CLK} /2 ¹⁴	1.95 kHz
1	1	1	1	X	X	X	X	f _{CLK} /2 ¹⁵	977 Hz	
其他									禁止设置	

注 要更改被选为 f_{CLK} 的时钟（通过更改系统时钟控制寄存器(CKC)的值）时，必须在停止串行阵列单元(SAU)操作（串行通道停止寄存器 m (STm) = 000FH）之后才可以执行。

- 备注 1. X: 忽略
2. m: 单元编号(m = 0), n: 通道编号(n = 0)

15.5.9 3线串行输入/输出 (CSI00)通信过程中发生错误时的处理步骤

在三线串行输入/输出 (CSI00)通信过程中发生错误时的处理步骤如图 12-76 所示。

图 15-75. 发生溢出错误时的处理步骤

软件操作	硬件状态	备注
读取串行数据寄存器mn (SDRmn)。→	SSRmn寄存器的BFFmn位被设置为0，并允许通道n接收数据。	这是为了防止在处理错误的过程中完成了下一次接收时所发生的溢出错误。
读取串行状态寄存器mn (SSRmn)。		识别出错误类型，并用读取值来清除错误标志。
将1写入串行标志清除触发寄存器mn (SIRmn)。→	错误标志被清除。	只有在读取期间可以清除错误，其方法将从SSRmn寄存器读取的值不做修改地直接写入SIRmn寄存器。

备注 m: 单元编号(m = 0), n: 通道编号(n = 0)

15.6 UART(UART0、UART1)通信的操作

这是一种异步功能，采用两条线路：串行/数据传送 (TxD)和串行/数据接收 (RxD) 线路。利用这两条通信线路，各数据帧（由一个起始位、数据、奇偶校验位和停止位构成）在单片机与其他通信方之间（以内部波特率）异步传送。全双工异步 UART 通信可以利用一个发送专用通道（偶数通道）和一个接收专用通道（奇数通道）来实现。可以结合使用定时器阵列单元和外部中断(INTP0)来实现支持 LIN-bus 和 DMX512。

[数据的发送/接收]

- 数据长度为 7、8 或 9 位^注
- MSB/LSB 选择
- 发送/接收数据的电平设置（选择是否反转电平）
- 附加奇偶校验位和奇偶校验功能
- 停止位附加、停止位校验功能

[中断功能]

- 传送结束中断/缓冲器空中断
- 出现帧错误、奇偶检验错误或溢出错误时的错误中断

[错误检测标志]

- 帧错误、奇偶校验错误、溢出错误

另外，UART0 接收(通道 1)支持 SNOOZE 模式。借助 SNOOZE 模式，在 STOP 模式下检测到 RxD 输入时不需要 CPU 就执行数据接收。

并且，UART0 (通道 0 和 1)支持 LIN-bus 和 DMX512。

[LIN-bus 功能]

- 唤醒信号检测
- 间隔段(BF)检测
- 同步段测量、波特率计算

使用外部中断(INTP0) 和
定时器阵列单元 0

[DMX512 功能]

- BREAK 信号检测
- 脉冲宽度检测

使用外部中断(INTP0) 和
定时器阵列单元 0

注 仅限 UART0 可以指定 9 位的数据长度。

UART0 使用通道 0 和 1。

UART1 使用通道 2 和 3。

• 20 和 32 引脚产品

通道	用作 CSI	用作 UART	用作简易 I ² C
0	-	UART0 (支持 LIN-bus、DMX512)	-
1	-		-
2	-	-	-
3	-		-

• 30 引脚产品

通道	用作 CSI	用作 UART	用作简易 I ² C
0	-	UART0 (支持 LIN-bus、DMX512)	-
1	-		-
2	-	UART1	-
3	-		-

• 38 引脚产品

通道	用作 CSI	用作 UART	用作简易 I ² C
0	CSI00	UART0 (支持 LIN-bus、DMX512)	-
1	-		-
2	-	UART1	-
3	-		-

可以对每个通道选择任何一个功能。但是，仅限所选的功能有效。例如，对于单元 0 的通道 0 和 1，如果选择 UART0，则这些通道就不能使用 CSI00 和 CSI01 功能。同时，同一单元的通道 2、3 或其他通道可以使用 UART0 以外的功能（例如 CSI10、UART1 和 IIC10）。

注意事项 在将串行阵列单元用作 UART 时，发送侧（偶数通道）和接收侧（奇数通道）的通道都只能用作 UART。

UART 有以下几种类型的通信操作。

- UART 发送 (参阅 15.6.1)
- UART 接收 (参阅 15.6.2)
- LIN 发送(仅限 UART0) (参阅 15.7.1)
- LIN 接收(仅限 UART0) (参阅 15.7.2)
- DMX512 接收(仅限 UART0) (参阅 15.8)

15.6.1 UART发送

UART 发送操作用于将数据从 RL78/I1A 异步发送至另一器件（开始-停止同步）。

在用于 UART 的两个通道中，偶数通道用于 UART 发送。

UART	UART0	UART1
目标通道	SAU0的通道0	SAU0的通道2
使用引脚	TxD0	TxD1
中断	INTST0	INTST1
	可选择发送结束中断(单发送模式时)或缓冲器空中断(连续发送模式时)	
错误检测标志	无	
传送数据长度	7、8或9位 ^{注1}	
传送速率	Max. $f_{mck}/6$ [bps] (SDRmn [15:9] = 2或以上), Min. $f_{clk}/(2 \times 2^{15} \times 128)$ [bps] ^{注2}	
数据相位	正相输出 (默认: 高电平) 反相输出 (默认: 低电平)	
奇偶校验位	以下可选 <ul style="list-style-type: none"> • 无奇偶校验位 • 附加零校验 • 附加偶校验 • 附加奇校验 	
停止位	以下可选 <ul style="list-style-type: none"> • 附加1位 • 附加2位 	
数据方向	MSB或LSB优先	

注 1. 仅限 UART0 可以指定 9 位的数据长度。

2. 在满足以上条件及电特性中的外围功能特性(参阅第三十二章 电特性)的范围内使用此操作。

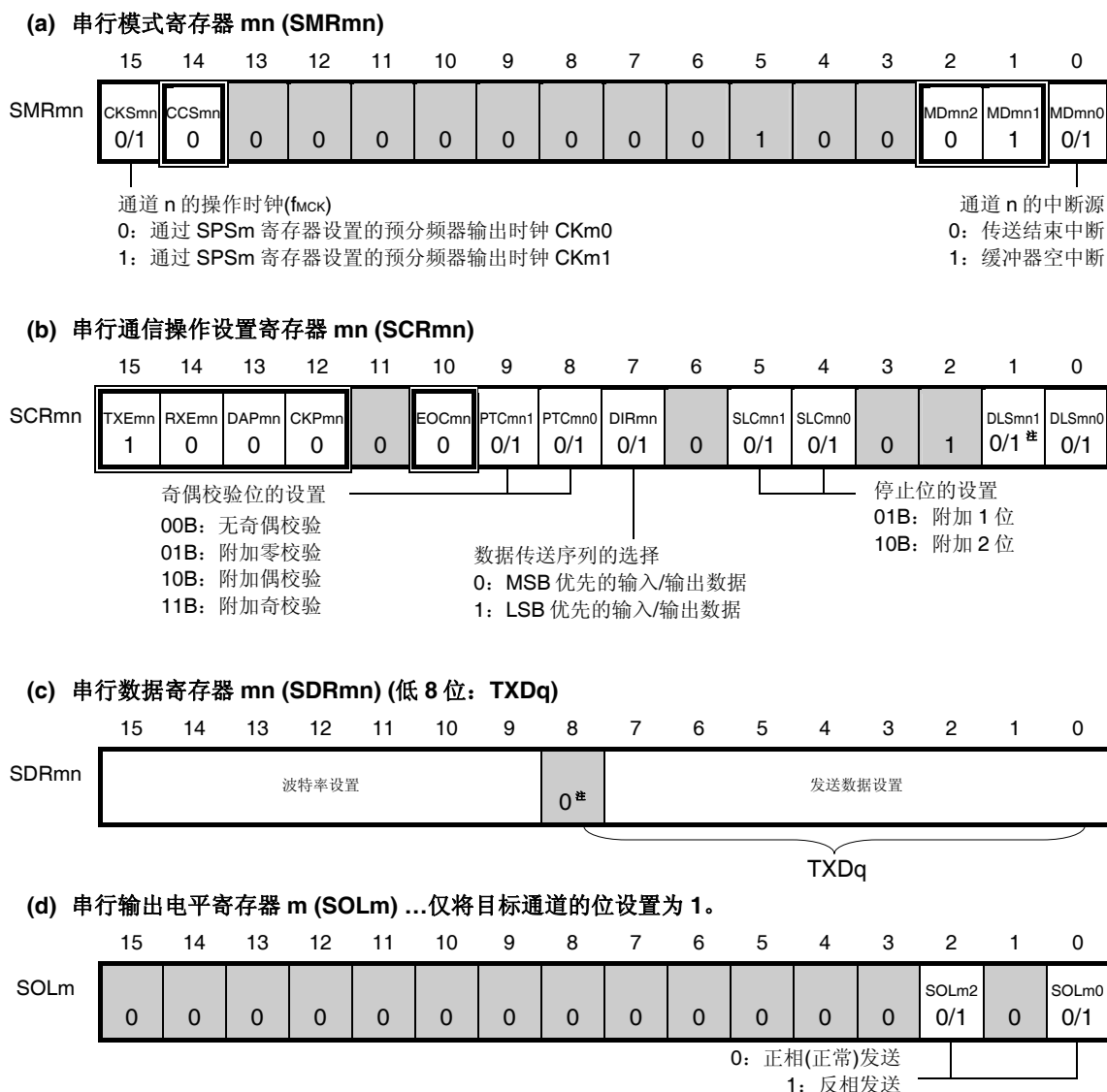
备注 1. f_{mck} : 目标通道的操作时钟频率

f_{clk} : 系统时钟频率

2. m: 单元编号(m = 0), n: 通道编号(n = 0, 2), mn = 00, 02

(1) 寄存器设置

图 15-76. UART(UART0, UART1)的 UART 发送时的寄存器设置内容示例(1/2)



- 注 1. 仅限 SCR00 寄存器(UART0)。SCR02 寄存器的该位固定为 1。
2. 当 UART0 执行 9 位通信时（通过将 SCR00 寄存器的 DLS001 和 DLS000 位设置为 1），SDR00 寄存器的 0 至位 8 用作发送数据指定区。仅限 UART0 可以指定 9 位的数据长度。

- 备注 1. m: 单元编号(m = 0), n: 通道编号(n = 0, 2), q: UART 编号(q = 0, 1), mn = 00, 02
2. □: 设置固定于 UART 发送模式, ■: 禁止设置 (设为初始值)
- x: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)
- 0/1: 可根据客户的用途设为 0 或 1

图 15-76. UART(UART0, UART1)的 UART 发送时的寄存器设置内容示例(2/2)

(e) 串行输出寄存器 m (SOm) ... 仅将目标通道的位设置为 1。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOm	0	0	0	0	1	CKOm2 ×	1	CKOm0 ×	0	0	0	0	1	SOm2 0/1 ^注	1	SOm0 0/1 ^注

0: 串行数据输出值为 0
1: 串行数据输出值为 1

(f) 串行输出允许寄存器 m (SOEm) ... 仅将目标通道的位设置为 1。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOEm	0	0	0	0	0	0	0	0	0	0	0	0	0	SOEm2 0/1	0	SOEm0 0/1

(g) 串行通道开始寄存器 m (SSm) ... 仅将目标通道的位设置为 1。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSm	0	0	0	0	0	0	0	0	0	0	0	0	SSm3 ×	SSm2 0/1	SSm1 ×	SSm0 0/1

注 开始发送前，当目标通道的 SOLmn 位为 0 时，必须设置为 1，当目标通道的 SOLmn 位为 1 时，必须清除为 0。在通信操作过程中，该值因通信数据而异。

- 备注** 1. m: 单元编号(m = 0), n: 通道编号(n = 0, 2), q: UART 编号(q = 0, 1),
mn = 00, 02
2. : 禁止设置(设置为初始值)
x: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)
0/1: 可根据客户的用途设为 0 或 1

(2) 操作步骤

图 15-77. UART 发送的初始设定步骤

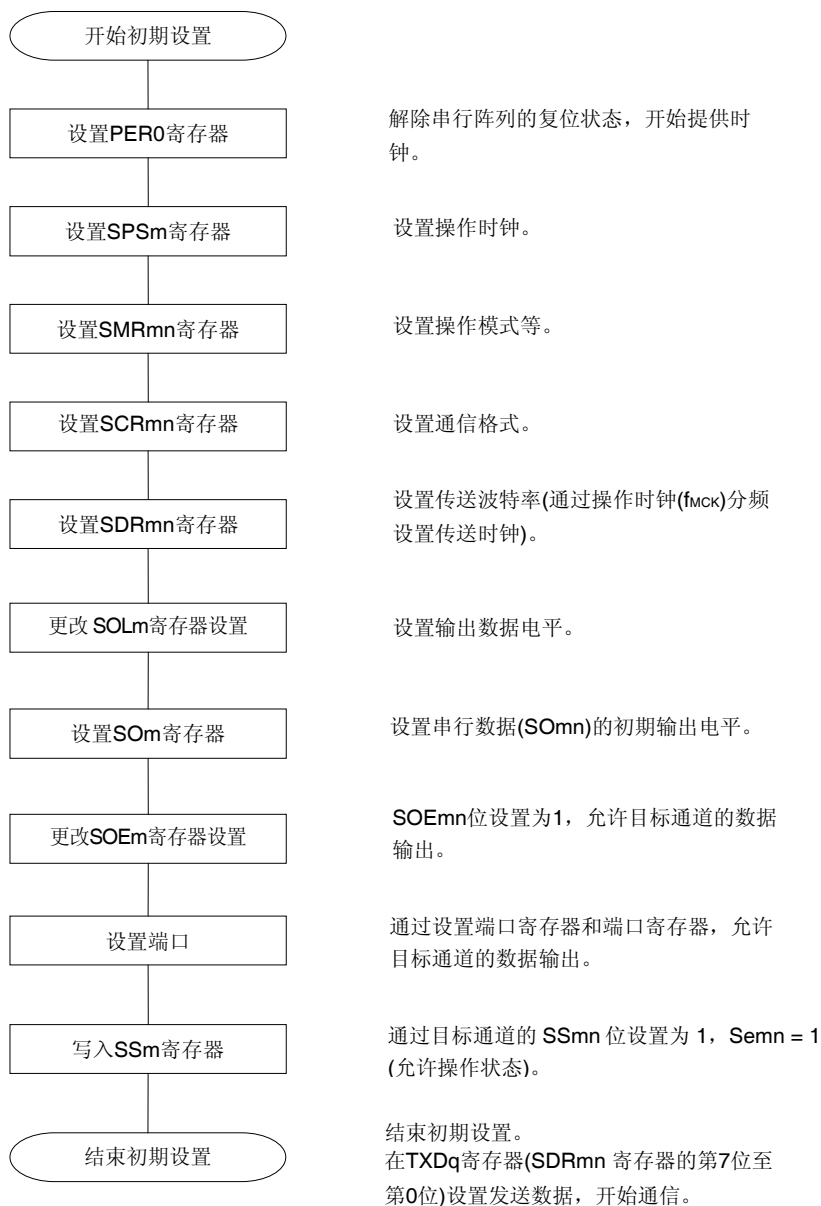


图 15-78. 停止 UART 发送的步骤

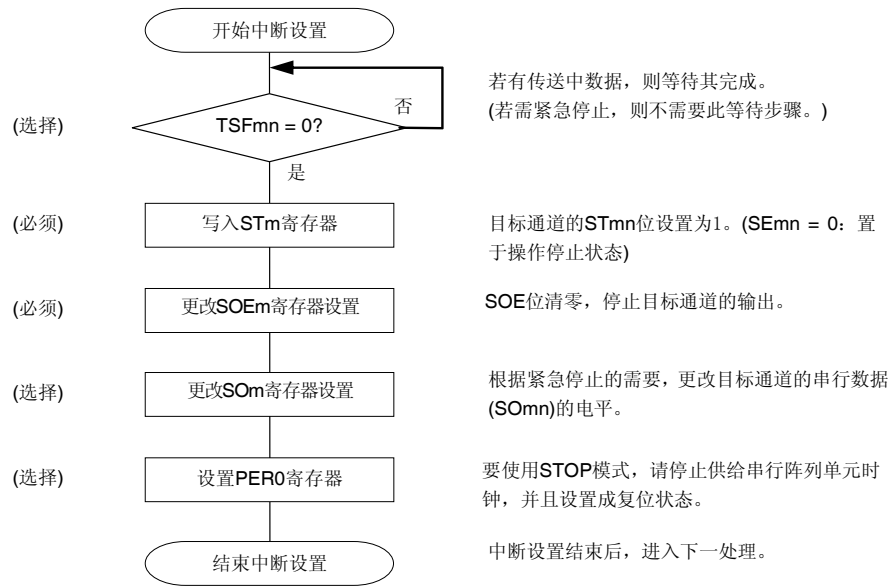
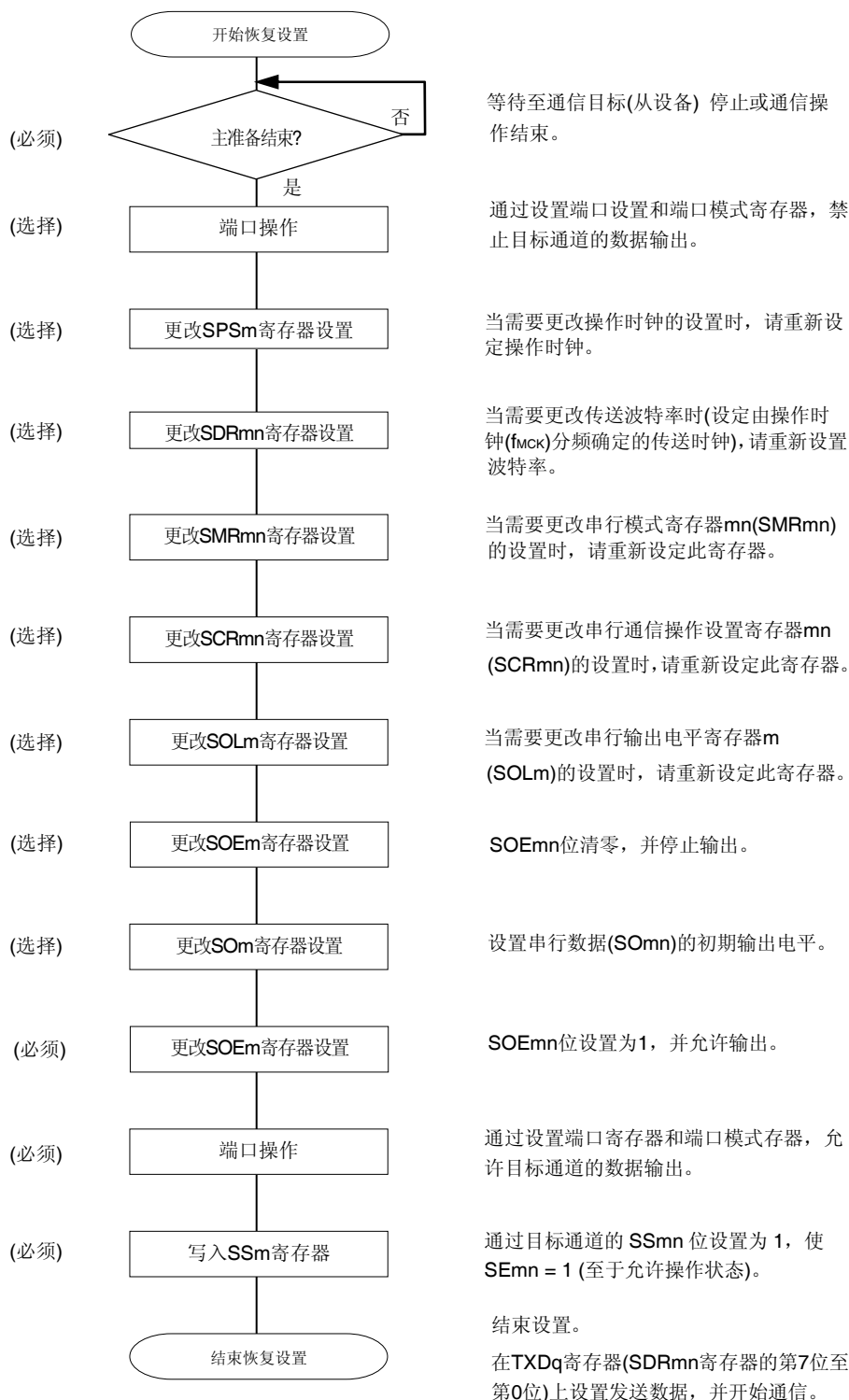


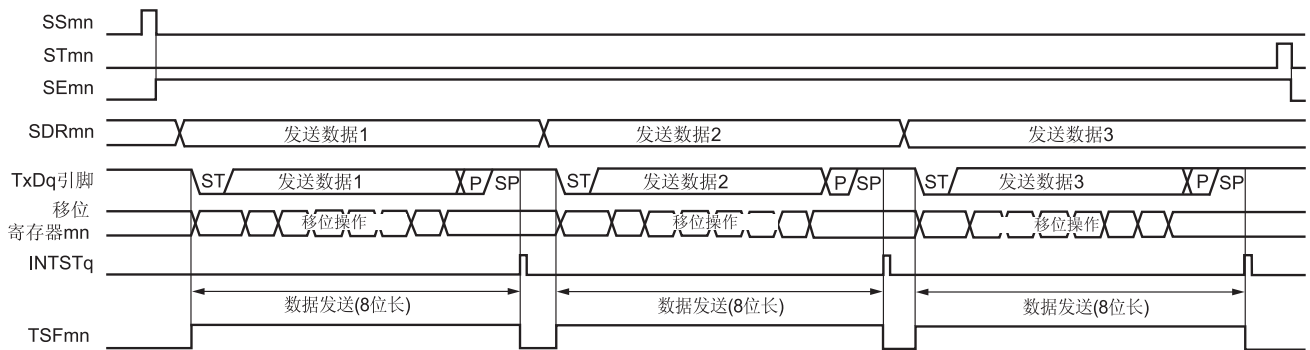
图 15-79. 恢复 UART 发送的步骤



备注 在停止 UART 发送的步骤中改写 PER0 而停止时钟供应时, 请等待通信目标停止或者通信结束, 然后再执行初始设置而不是重传设置。

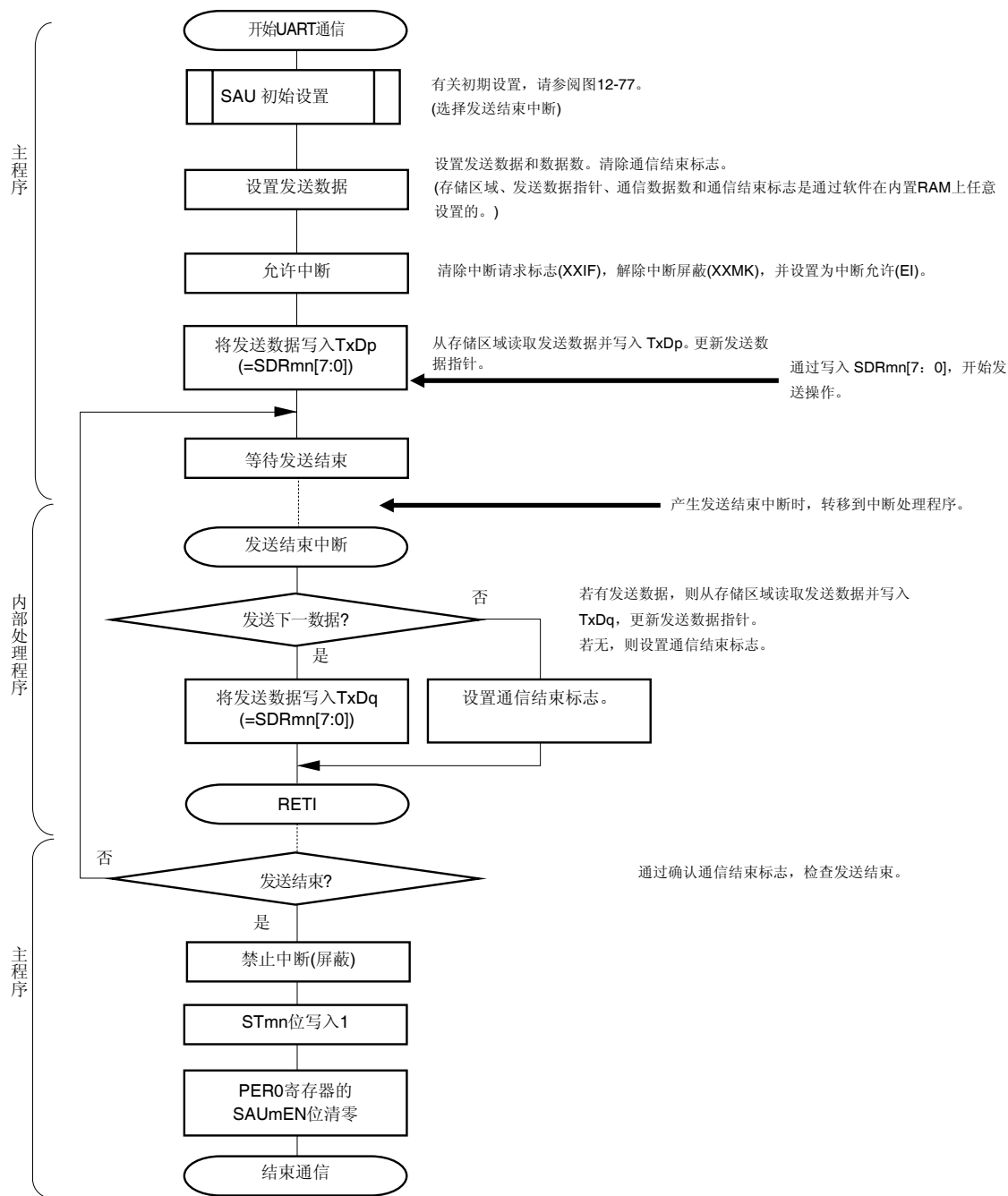
(3) 处理流程 (单发送模式时)

图 15-80. UART 发送(单发送模式时)的时序图



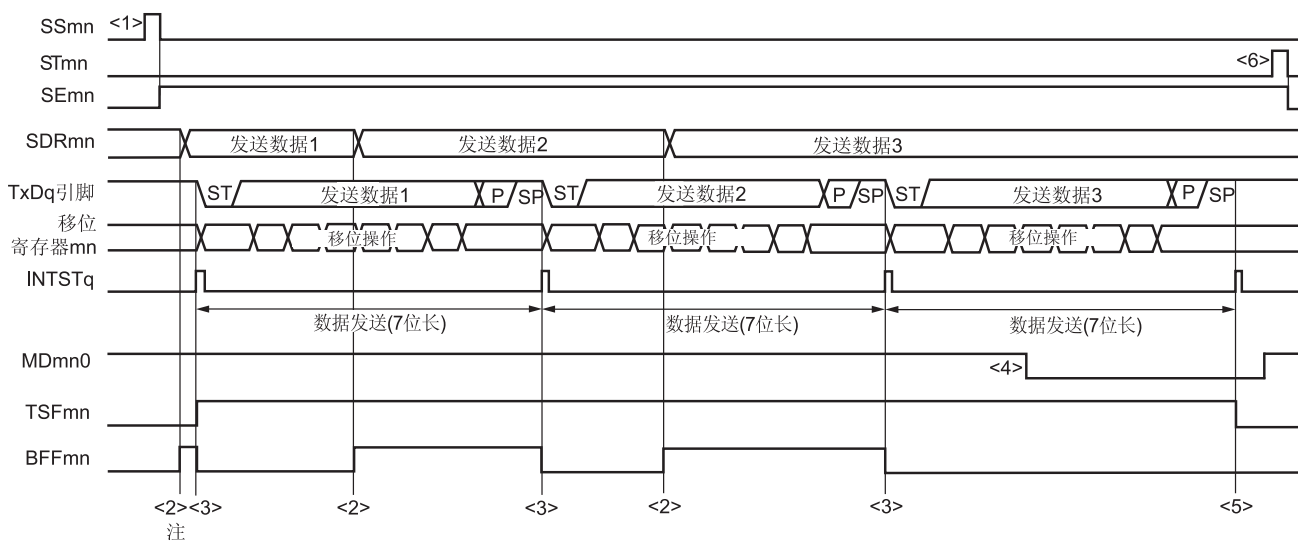
备注 m: 单元编号(m = 0), n: 通道编号(n = 0, 2), q: UART 编号(q = 0, 1), mn = 00, 02

图 15-81. UART 发送(单发送模式时)的流程图



(4) 处理流程 (连续发送模式时)

图 15-82. UART 发送(连续发送模式时)的时序图

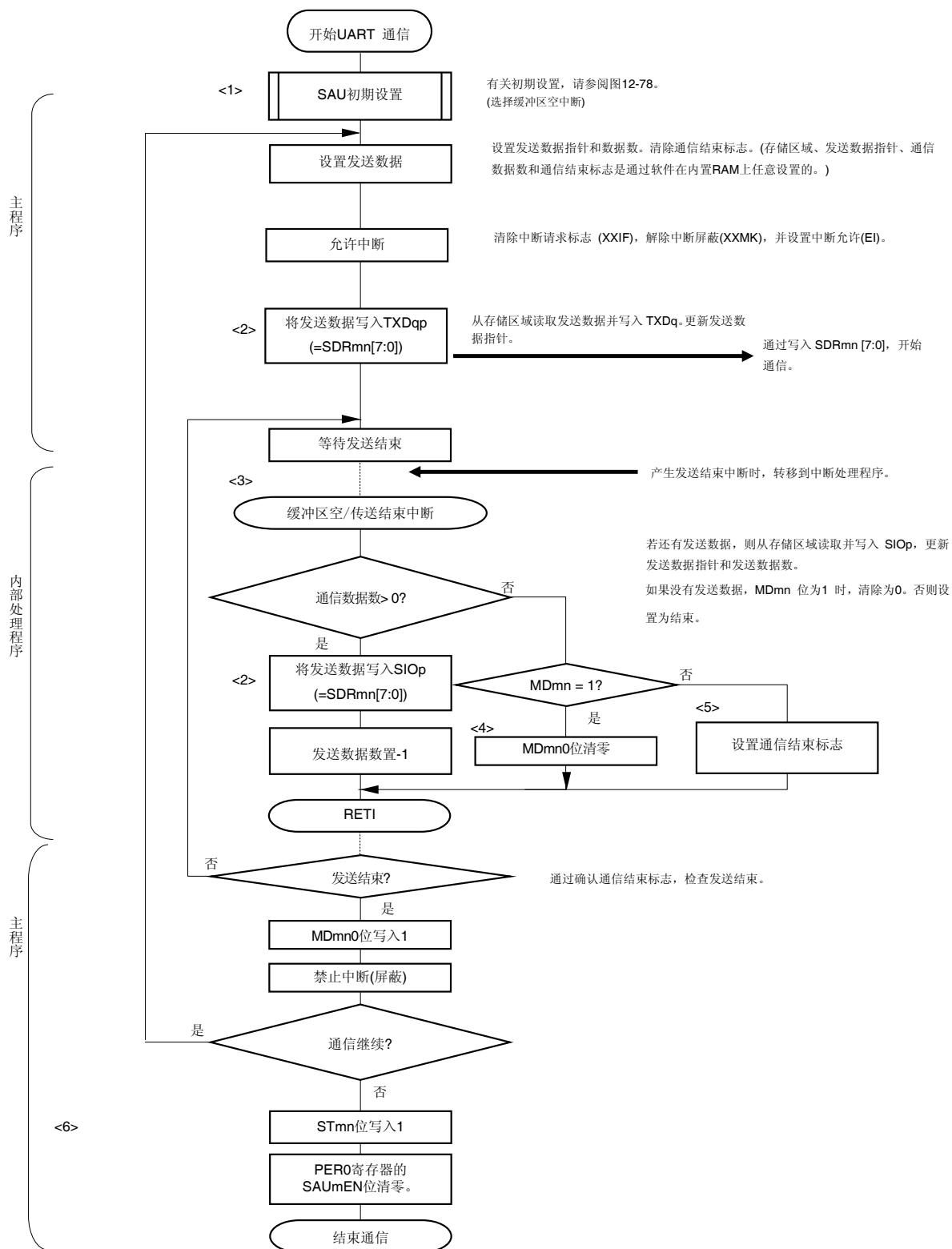


注 如果在串行状态寄存器 mn (SSRmn) 的 BFFmn 位为 1 时将发送数据写入 SDRmn 寄存器 (将有效数据存储于串行数据寄存器 mn (SDRmn) 中), 则发送数据将被覆盖。

注意事项 即使在操作过程中, 也可重写串行模式寄存器 mn (SMRmn) 的 MDmn0 位。然而必须在开始传送最后一位之前改写, 以便在最后发送数据的传送结束中断前完成改写。

备注 m: 单元编号(m = 0), n: 通道编号(n = 0, 2), q: UART 编号(q = 0, 1), mn = 00, 02

图 15-83. UART 发送(连续发送模式时)的流程图



备注 此图中的<1>至<6>对应于图 15-83. UART 发送(连续发送模式时)的时序图中的<1>至<6>。

15.6.2 UART接收

UART 接收操作指 RL78/I1A 从另一器件异步接收数据（调步同期）。

用于 UART 的两个通道中的奇数通道被用作 UART 接收。必须设置奇数和偶数通道的 SMR 寄存器。

UART	UART0	UART1
目标通道	SAU0的通道1	SAU0的通道3
使用引脚	RxD0	RxD1
中断	INTSR0	INTSR1
	仅限传送结束中断(禁止设置缓冲器空中断。)	
错误中断	INTSRE0	INTSRE1
错误检测标志	<ul style="list-style-type: none"> • 帧错误检测标志 (FEFmn) • 奇偶校验错误检测标志 (PEFmn) • 溢出错误检测标志 (OVFmn) 	
传送数据长度	7、8或9位 ^{注1}	
传送速率	Max. $f_{mck}/6$ [bps] (SDRmn [15:9] = 2或以上), Min. $f_{clk}/(2 \times 2^{15} \times 128)$ [bps] ^{注2}	
数据相位	正相输出 (默认: 高电平) 反相输出 (默认: 低电平)	
奇偶校验位	以下可选 <ul style="list-style-type: none"> • 无奇偶校验位 (无奇偶校验) • 无奇偶校验判断(0奇偶校验) • 偶校验 • 奇校验 	
停止位	1位验证	
数据方向	MSB或LSB优先	

注 1. 仅限 UART0 可以指定 8 位的数据长度。

2. 在满足以上条件及电特性中的外围功能特性(参阅第三十二章 电特性)的范围内使用此操作。

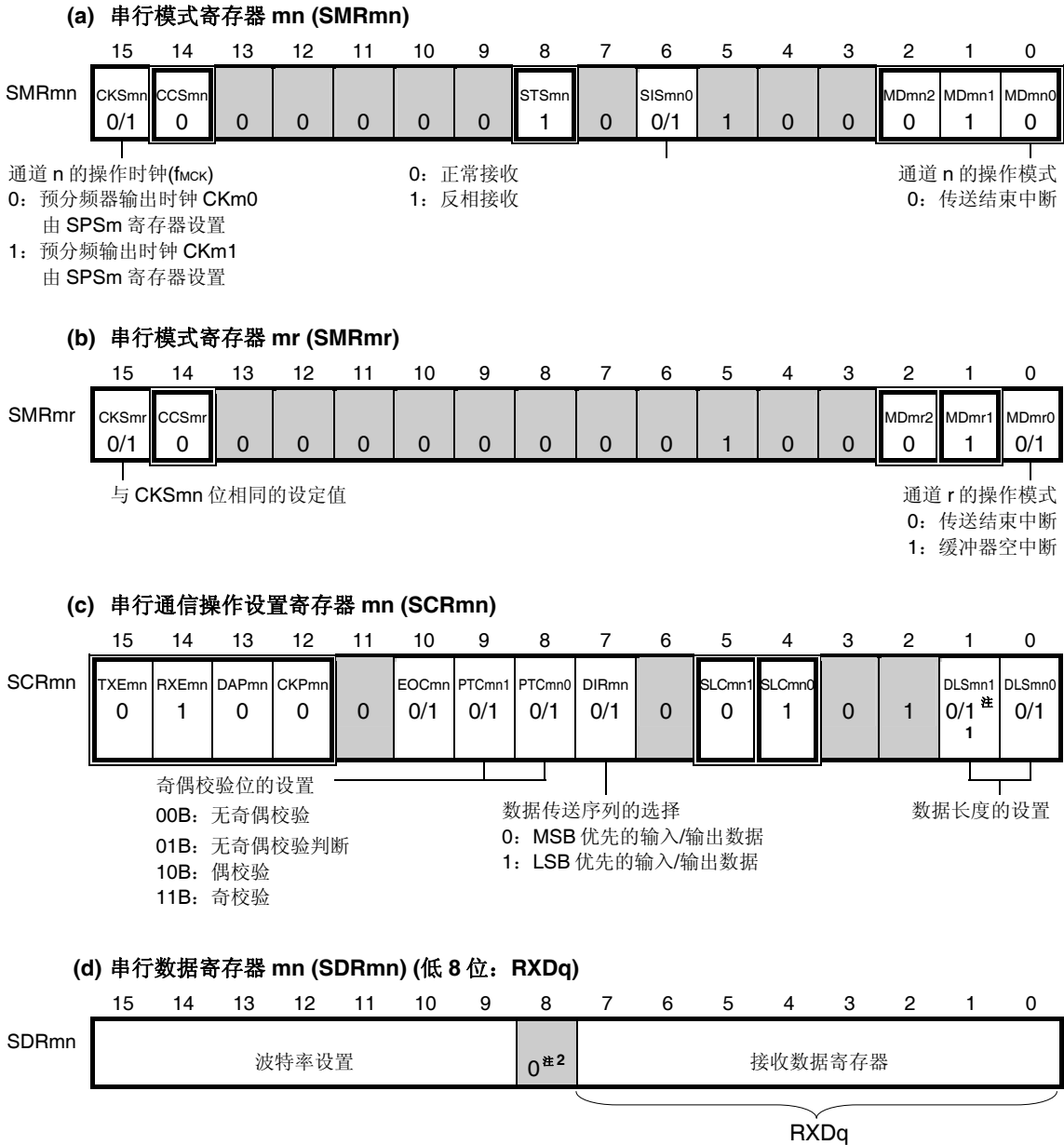
备注 1. f_{mck} : 目标通道的操作时钟频率

f_{clk} : 系统时钟频率

2. m: 单元编号(m = 0), n: 通道编号(n = 1, 3), mn = 01, 03

(1) 寄存器设置

图 15-84. UART(UART0, UART1)的 UART 接收时的寄存器设置内容示例(1/2)



注 1. 仅限 SCR00 寄存器(UART0)。SCR02 寄存器的该位固定为 1。
 2. 当 UART 执行 9 位通信时，SDRm1 寄存器的位 0 至位 8 用作发送数据指定区。仅限 UART0 可以指定 8 位的数据长度。

注意事项 对于 UART 接收，必须将通道 r 的 SMRmr 寄存器设置为 UART 发送模式，使其与通道 n 配对。

备注 1. m: 单元编号(m = 0), n: 通道编号(n = 1, 3), mn = 01, 03
 r: 通道编号(r = n - 1), q: UART 编号(q = 0, 1)
 2. □: 设置固定于 UART 接收模式, ■: 禁止设置 (设为初始值)
 x: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)
 0/1: 可根据客户的用途设为 0 或 1

图 15-84. UART(UART0, UART1)的 UART 接收时的寄存器设置内容示例(2/2)

(e) 串行输出寄存器 m (SOm) ...此模式下不使用的寄存器。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOm	0	0	0	0	1	CKOm2 ×	1	CKOm0 ×	0	0	0	0	1	SOM2 ×	1	SOM0 ×

(f) 串行输出允许寄存器 m (SOEm) ...此模式下不使用的寄存器。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOEm	0	0	0	0	0	0	0	0	0	0	0	0	0	SOEm2 ×	0	SOEm0 ×

(g) 串行通道开始寄存器 m (SSm) ...仅将目标通道的位设置为 1。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSm	0	0	0	0	0	0	0	0	0	0	0	0	SSm3 0/1	SSm2 ×	SSm1 0/1	SSm0 ×

注意事项 对于 UART 接收，必须将通道 r 的 SMRmr 寄存器设置为 UART 发送模式，使其与通道 n 配对。

备注 1. m: 单元编号(m = 0), n: 通道编号(n = 1, 3), mn = 01, 03

r: 通道编号(r = n - 1), q: UART 编号(q = 0, 1)

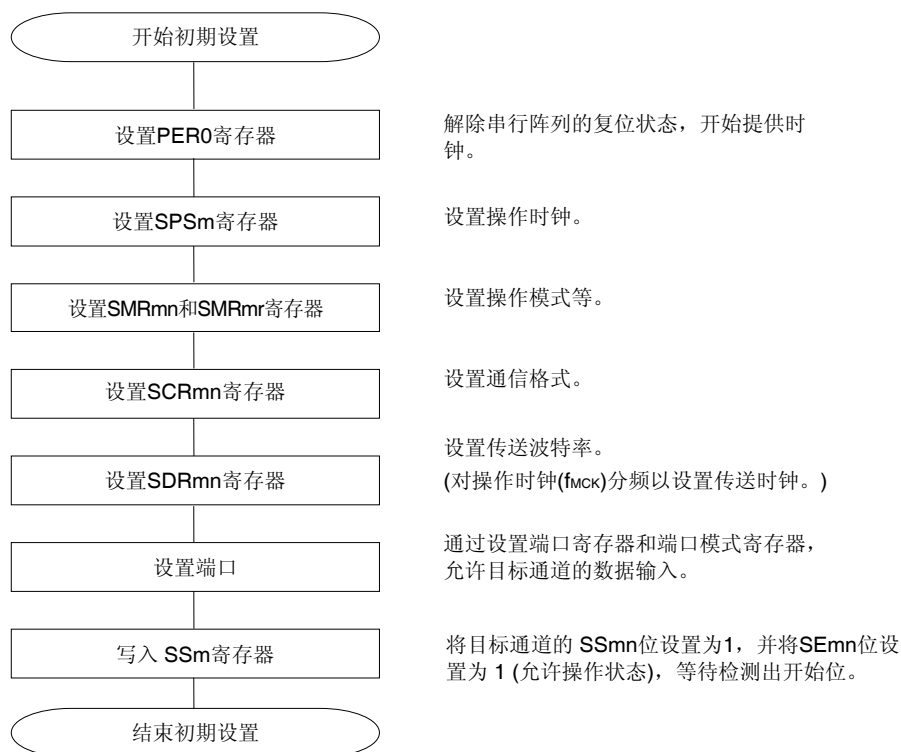
2. : 设置固定于 UART 接收模式, : 禁止设置 (设为初始值)

x: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)

0/1: 可根据客户的用途设为 0 或 1

(2) 操作步骤

图 15-85. UART 接收的初始设定步骤



注意事项 将 SCR_{mn} 寄存器的 RXE_{mn} 位设置为 1，然后确保在经过 4 个或更多 f_{mck} 时钟之后将 SS_{mn} 设置为 1。

图 15-86. 停止 UART 接收的步骤

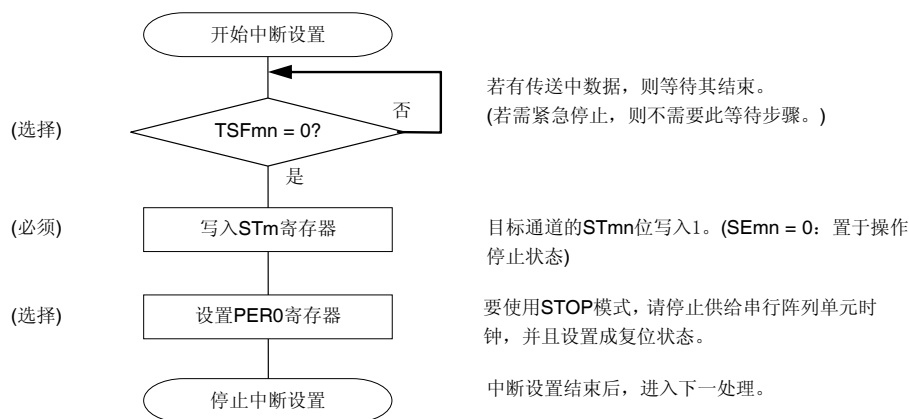
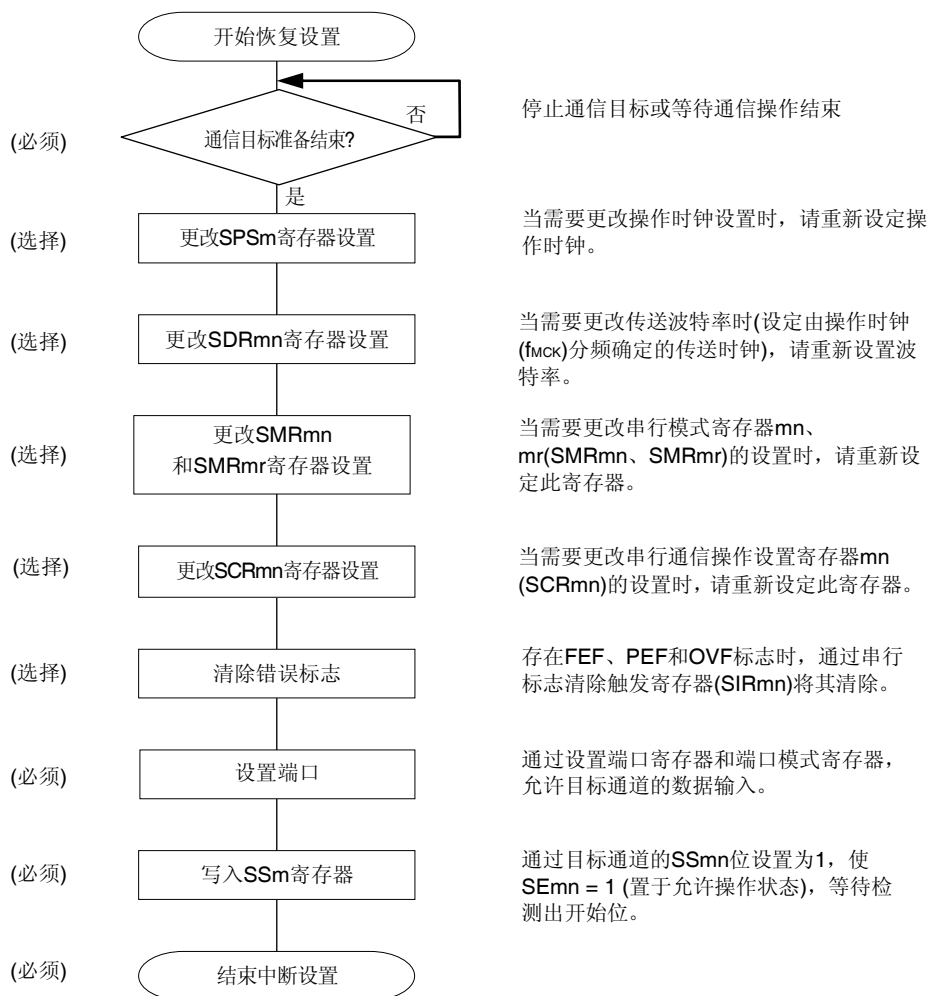


图 15-87. 恢复 UART 接收的步骤

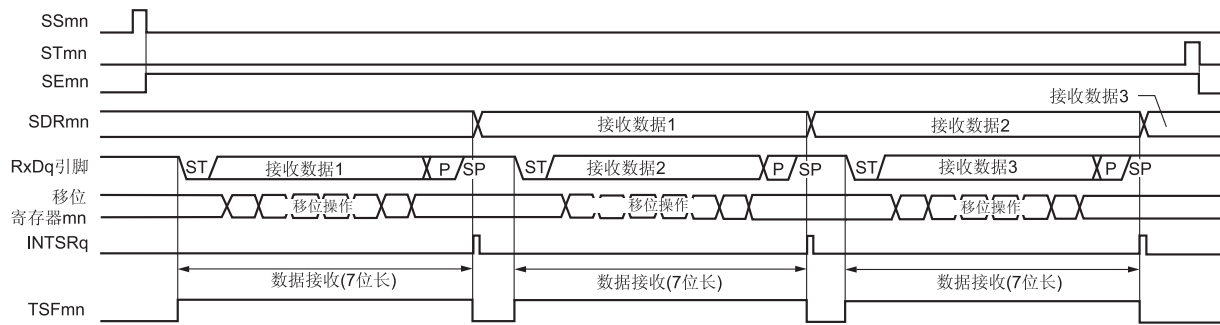


注意事项 将 SCRmn 寄存器的 RXEmn 位设置为 1 后, 等待至少 4 个 f_{mck} 时钟的间隔后再设置 SSmn = 1。

备注 在停止 UART 接收的步骤中改写 PER0 而停止时钟供应时, 请等待通信目标停止或者通信结束, 然后执行初始设置, 而不是重传设置。

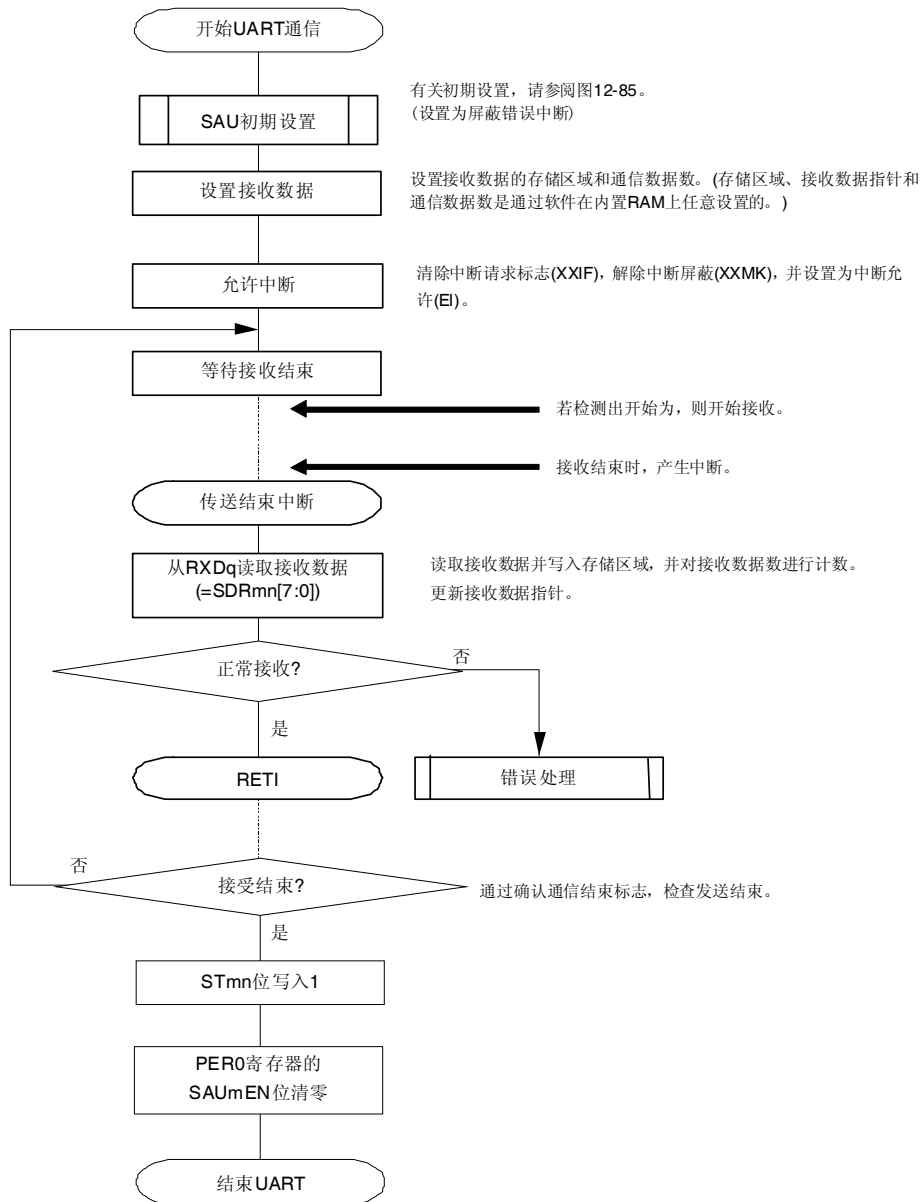
(3) 处理流程

图 15-88. UART 接收的时序图



备注 m: 单元编号(m = 0), n: 通道编号(n = 1, 3), mn = 01, 03
 r: 通道编号(r = n - 1), q: UART 编号(q = 0, 1)

图 15-89. UART 接收的流程图



15.6.3 SNOOZE模式功能 (仅限UART0 接收)

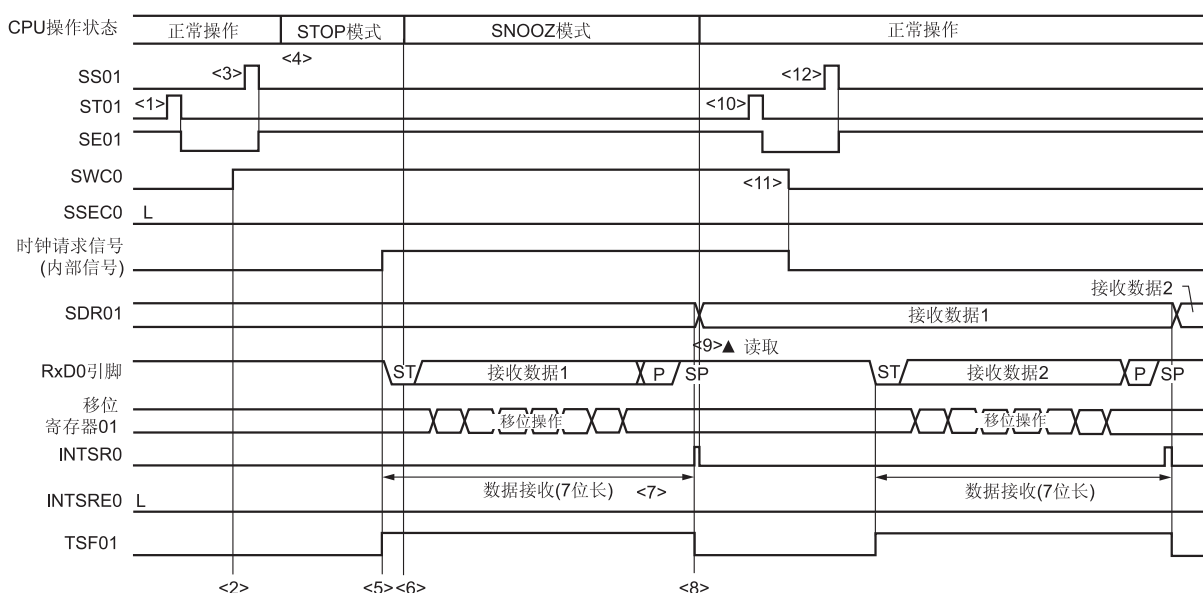
利用 SNOOZE 模式，在 STOP 模式下检测到 RxD0 引脚输入时可以执行 UART 接收操作。通常在 STOP 模式下 UART 停止通信。但是，利用 SNOOZE 模式，可以在检测到 RxD0 引脚输入时不启动 CPU 就执行 UART 接收。仅限 UART0 可以设置 SNOOZE 模式。

使用 SNOOZE 模式功能时，须在切换至 STOP 模式之前将串行待机控制寄存器 0 (SSC0)的 SWC0 位设置为 1。

- 1. 仅限选择高速片上振荡器时钟作为 f_{CLK}时可以设置 SNOOZE 模式。
- 2. 在 SNOOZE 模式下使用 UART0 时的最大传送速率为 9600 bps。

(1) SNOOZE 模式操作(正常操作)

图 15-90. SNOOZE 模式操作(正常操作)的时序图



注 当 SWC0 = 1 时，读取接收数据。

- 注意事项 在切换至 SNOOZE 模式之前，或者在 SNOOZE 模式下的接收操作完成之后，必须将 ST01 位设为 1（清除 SE01 位，并停止操作）。并且在完成接收操作之后，将 SWC0 位清除为 0（释放 SNOOZE 模式）。

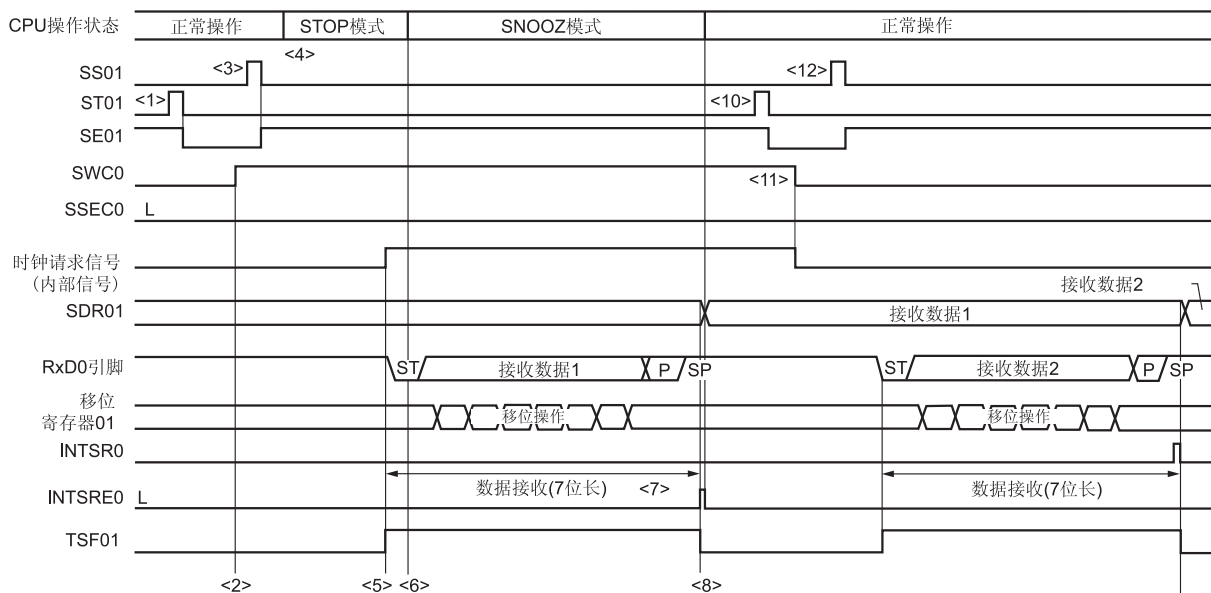
备注 此图中的<1>至<11>对应于图 15-92. SNOOZE 模式操作(正常操作/异常操作<1>)的流程图中的<1>至<11>。

(2) SNOOZE 模式操作(异常操作<1>)

异常操作<1> 指在 SSEC0 = 0 的情况下发生通信错误时执行的操作。

由于 SSEC0 = 0，所以在发生通信错误时将产生错误中断 (INTSRE0)。

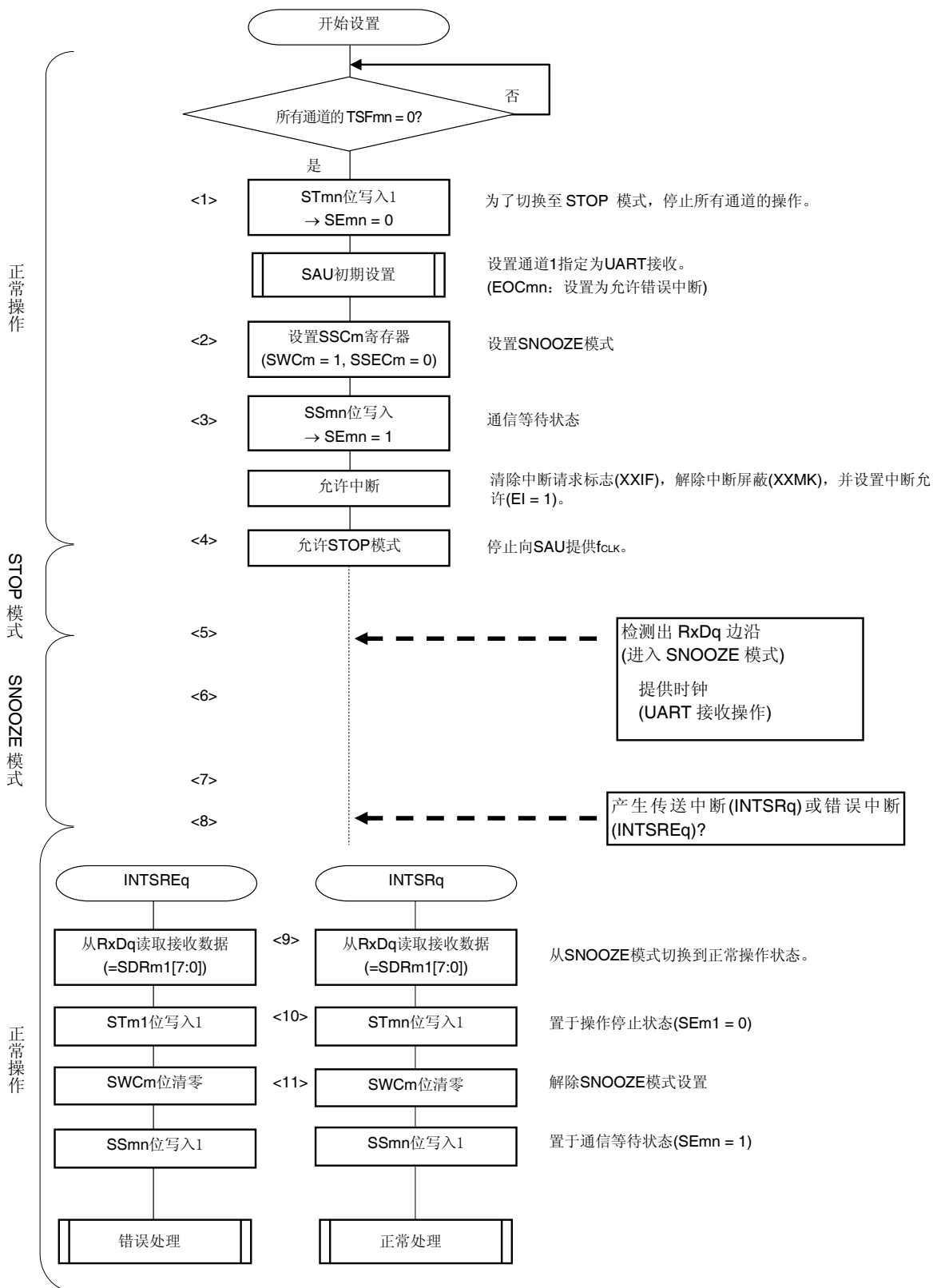
图 15-91. SNOOZE 模式操作(异常操作<1>)的时序图



注意事项 在切换至 SNOOZE 模式之前，或者在 SNOOZE 模式下的接收操作完成之后，必须将 ST01 位设为 1（清除 SE01 位，并停止操作）。
并且在完成接收操作之后，将 SWC0 位清除为 0（释放 SNOOZE 模式）。

备注 此图中的<1>至<11>对应于图 15-92. SNOOZE 模式操作(正常操作/异常操作<1>)的流程图中的<1>至<11>。

图 15-92. SNOOZE 模式操作(正常操作/异常操作<1>)的流程图

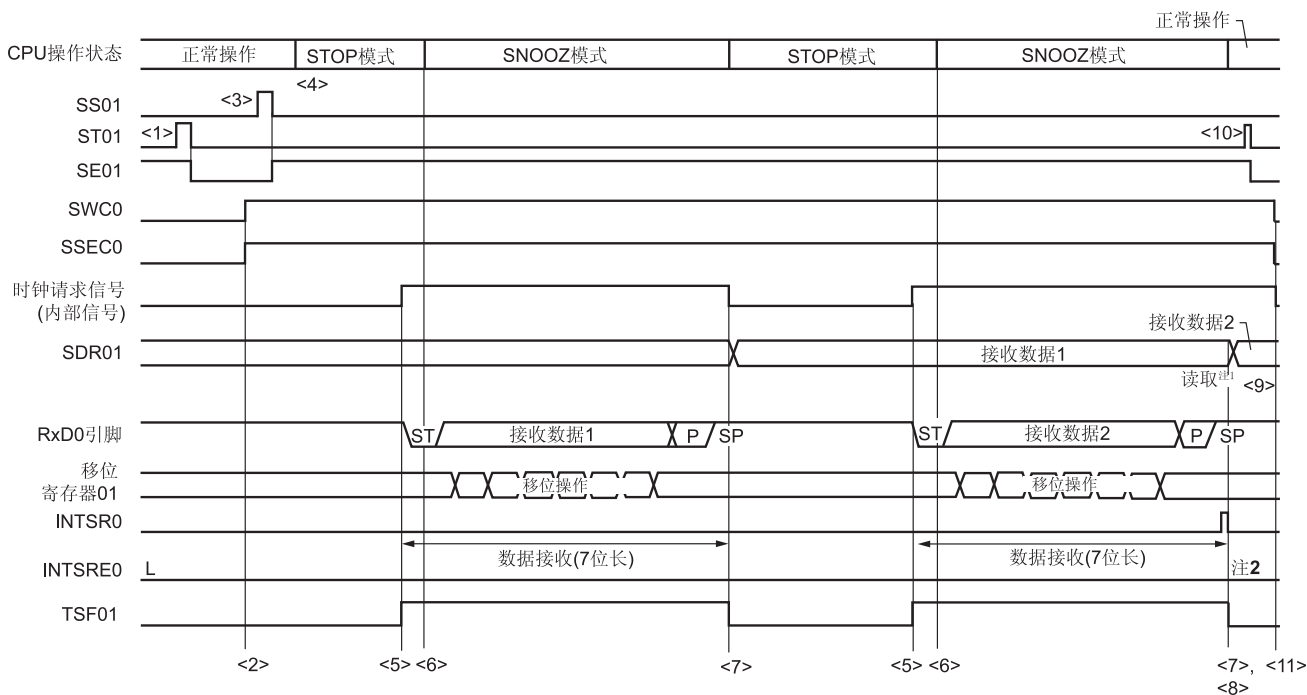


备注 此图中的<1>至<11>对应于图 15-90. SNOOZE 模式操作(正常操作)的时序图和图 15-91. SNOOZE 模式操作(异常操作<1>)的时序图中的<1>至<11>。

(3) SNOOZE 模式操作(异常操作<2>)

异常操作<2> 指在 $SSEC0 = 1$ 的情况下发生通信错误时执行的操作。
 由于 $SSEC0 = 1$ ，所以在发生通信错误时不产生错误中断 (INTSRE0)。

图 15-93. SNOOZE 模式操作(异常操作<2>)的时序图

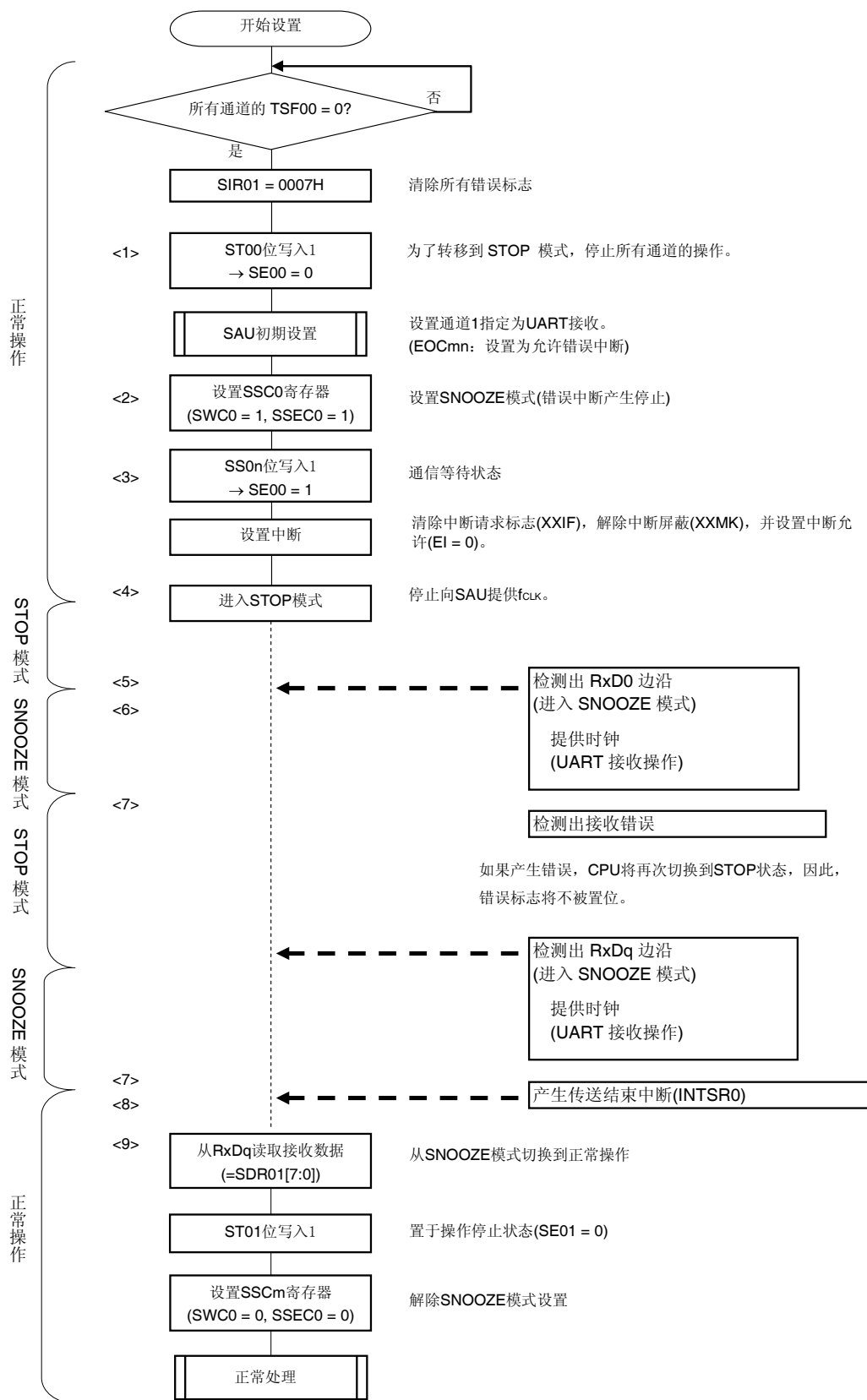


- 注 1. 仅当 $SWC0 = 1$ 时才读取接收数据。
- 2. SNOOZE 模式下当 UART0 成功完成接收时，不需要更改设置就可以继续执行正常的接收操作，但是，由于 $SSEC0 = 1$ ，因此，即使发生帧错误或奇偶校验错误，PEF01 和 FEF01 位也不会被置位。另外，也不会产生错误中断 (INTSRE0)。

- 注意事项**
1. 在切换至 SNOOZE 模式之前，或者在 SNOOZE 模式下的接收操作完成之后，必须将 ST01 位设为 1（清除 SE01 位，并停止操作）。
 并且在完成接收操作之后，将 SWC0 位清除为 0（释放 SNOOZE 模式）。
 2. 当在 SSEC0 被设置为 1 的情况下使用 SNOOZE 模式时，不会发生溢出错误。因此，当使用 SNOOZE 模式时，要在切换至 STOP 模式之前读取 SDR01 寄存器的位 7 至位 0 (RxD0)。

备注 此图中的<1>至<9>对应于图 15-94. SNOOZE 模式操作(异常操作<2>)的流程图中的<1>至<9>。

图 15-94. SNOOZE 模式操作(异常操作<2>)的流程图



(注意事项和备注如下页所示。)

注意事项 当在 **SSEC0** 被设置为 **1** 的情况下使用 **SNOOZE** 模式时，不会发生溢出错误。因此，当使用 **SNOOZE** 模式时，要在切换至 **STOP** 模式之前读取 **SDR01** 寄存器的位 **7** 至位 **0 (RxD0)**。

备注 此图中的<1>至<9>对应于图 15-93. **SNOOZE** 模式操作(异常操作<2>)的时序图中的<1>至<9>。

15.6.4 计算波特率

(1) 波特率计算公式

用于 UART (UART0, UART1)通信的波特率可以通过以下表达式计算。

$$\text{(波特率)} = \{\text{目标通道的操作时钟}(f_{\text{MCK}})\text{频率}\} \div (\text{SDRmn}[15:9] + 1) \div 2 \text{ [bps]}$$

注意事项 禁止将串行数据寄存器 mn(SDRmn)设置为 $\text{SDRmn}[15:9] = (0000000\text{B}, 0000001\text{B})$ 。

备注 1. 当使用 UART 时, SDRmn[15:9]的值为 SDRmn 寄存器位 15 至位 9 的值(0000010B 至 1111111B), 即 2 至 127。

2. m: 单元编号(m = 0), n: 通道编号(n = 0), mn = 00 至 03

操作时钟(f_{MCK}) 取决于串行时钟选择寄存器 m (SPSm) 以及串行模式寄存器 mn (SMRmn)的位 15 (CKSmn)。

表 15-3. UART 操作时钟的选择

SMRmn 寄存器	SPSm 寄存器								操作时钟(f _{CLK}) [#]	
	CKSmn	PRS m13	PRS m12	PRS m11	PRS m10	PRS m03	PRS m02	PRS m01	PRS m00	f _{CLK} = 32 MHz
0	X	X	X	X	0	0	0	0	f _{CLK}	32 MHz
	X	X	X	X	0	0	0	1	f _{CLK} /2	16 MHz
	X	X	X	X	0	0	1	0	f _{CLK} /2 ²	8 MHz
	X	X	X	X	0	0	1	1	f _{CLK} /2 ³	4 MHz
	X	X	X	X	0	1	0	0	f _{CLK} /2 ⁴	2 MHz
	X	X	X	X	0	1	0	1	f _{CLK} /2 ⁵	1 MHz
	X	X	X	X	0	1	1	0	f _{CLK} /2 ⁶	500 kHz
	X	X	X	X	0	1	1	1	f _{CLK} /2 ⁷	250 kHz
	X	X	X	X	1	0	0	0	f _{CLK} /2 ⁸	125 kHz
	X	X	X	X	1	0	0	1	f _{CLK} /2 ⁹	62.5 kHz
	X	X	X	X	1	0	1	0	f _{CLK} /2 ¹⁰	31.25 kHz
	X	X	X	X	1	0	1	1	f _{CLK} /2 ¹¹	15.63 kHz
	X	X	X	X	1	1	0	0	f _{CLK} /2 ¹²	7.81 kHz
	X	X	X	X	1	1	0	1	f _{CLK} /2 ¹³	3.91 kHz
	X	X	X	X	1	1	1	0	f _{CLK} /2 ¹⁴	1.95 kHz
X	X	X	X	1	1	1	1	f _{CLK} /2 ¹⁵	977 Hz	
1	0	0	0	0	X	X	X	X	f _{CLK}	32 MHz
	0	0	0	1	X	X	X	X	f _{CLK} /2	16 MHz
	0	0	1	0	X	X	X	X	f _{CLK} /2 ²	8 MHz
	0	0	1	1	X	X	X	X	f _{CLK} /2 ³	4 MHz
	0	1	0	0	X	X	X	X	f _{CLK} /2 ⁴	2 MHz
	0	1	0	1	X	X	X	X	f _{CLK} /2 ⁵	1 MHz
	0	1	1	0	X	X	X	X	f _{CLK} /2 ⁶	500 kHz
	0	1	1	1	X	X	X	X	f _{CLK} /2 ⁷	250 kHz
	1	0	0	0	X	X	X	X	f _{CLK} /2 ⁸	125 kHz
	1	0	0	1	X	X	X	X	f _{CLK} /2 ⁹	62.5 kHz
	1	0	1	0	X	X	X	X	f _{CLK} /2 ¹⁰	31.25 kHz
	1	0	1	1	X	X	X	X	f _{CLK} /2 ¹¹	15.63 kHz
	1	1	0	0	X	X	X	X	f _{CLK} /2 ¹²	7.81 kHz
	1	1	0	1	X	X	X	X	f _{CLK} /2 ¹³	3.91 kHz
	1	1	1	0	X	X	X	X	f _{CLK} /2 ¹⁴	1.95 kHz
1	1	1	1	X	X	X	X	f _{CLK} /2 ¹⁵	977 Hz	
其他									禁止设置	

注 改变 f_{CLK} 的时钟选择时（通过更改系统时钟控制寄存器(CKC)的值），必须在停止串行阵列单元(SAU)操作（串行通道停止寄存器 m (STm) = 000FH）之后才可以执行。

备注 1. X: 忽略

2. m: 单元编号(m = 0), n: 通道编号(n = 0 至 3), mn = 00 至 03

(2) 发送期间的波特率误差

发送过程中 UART(UART0, UART1)通信的波特率误差可以用以下计算式算出。确保发送侧的波特率处于接收侧允许的波特率范围之内。

$$(\text{波特率误差}) = (\text{计算波特率值}) \div (\text{目标波特率}) \times 100 - 100 [\%]$$

f_{CLK} = 32 MHz 时的 UART 波特率的设置示例如下所示。

UART波特率 (目标波特率)	f _{CLK} = 32 MHz			
	操作时钟(f _{MCK})	SDRmn[15:9]	计算波特率	与目标波特率之间的误差
300 bps	f _{CLK} /2 ⁹	103	300.48 bps	+0.16 %
600 bps	f _{CLK} /2 ⁸	103	600.96 bps	+0.16 %
1200 bps	f _{CLK} /2 ⁷	103	1201.92 bps	+0.16 %
2400 bps	f _{CLK} /2 ⁶	103	2403.85 bps	+0.16 %
4800 bps	f _{CLK} /2 ⁵	103	4807.69 bps	+0.16 %
9600 bps	f _{CLK} /2 ⁴	103	9615.38 bps	+0.16 %
19200 bps	f _{CLK} /2 ³	103	19230.8 bps	+0.16 %
31250 bps	f _{CLK} /2 ³	63	31250.0 bps	±0.0 %
38400 bps	f _{CLK} /2 ²	103	38461.5 bps	+0.16 %
76800 bps	f _{CLK} /2	103	76923.1 bps	+0.16 %
153600 bps	f _{CLK}	103	153846 bps	+0.16 %
312500 bps	f _{CLK}	50	312500 bps	±0.39 %

备注 m: 单元编号(m = 0), n: 通道编号(n = 0, 2), mn = 00, 02

(3) 接收时的波特率允许范围

UART (UART0, UART1)通信过程中允许接收的波特率范围可以用以下计算式算出。确保发送侧的波特率处于接收侧允许的波特率范围之内。

$$\text{(最大可接收波特率)} = \frac{2 \times k \times \text{Nfr}}{2 \times k \times \text{Nfr} - k + 2} \times \text{Brate}$$

$$\text{(最小可接收波特率)} = \frac{2 \times k \times (\text{Nfr} - 1)}{2 \times k \times \text{Nfr} - k - 2} \times \text{Brate}$$

Brate: 接收方的计算波特率值 (参阅 15.6.4 (1) 波特率计算公式。)

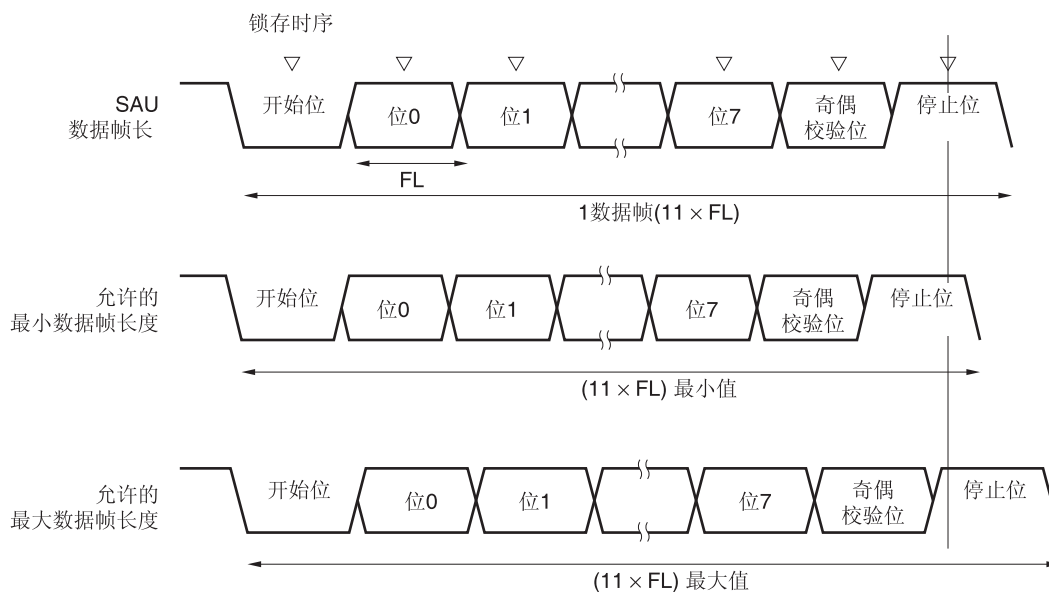
k: SDRmn[15:9] + 1

Nfr: 1 数据帧长度 [位]

= (开始位) + (数据长度) + (奇偶校验位) + (停止位)

备注 m: 单元编号(m = 0), n: 通道编号(n = 1, 3), mn = 01, 03

图 15-95. 接收时的允许波特率范围 (1 数据帧长度 = 11 位时)



如图 15-95 所示，接收数据的锁存时序取决于检测到起始位之后由串行数据寄存器 mn (SDRmn)的位 15 至位 9 设置的分频比。如果在该锁存时间之前收到最后数据（停止位），则可以正常进行接收。

15.6.5 UART(UART0, UART1)通信过程中发生错误时的处理步骤

在 UART (UART0, UART1)通信过程中发生的错误的处理步骤如图 15-96 和 15-97 所示。

图 15-96. 发生奇偶校验错误或溢出错误时的处理步骤

软件操作	硬件状态	备注
读取串行数据寄存器 mn (SDRmn)。	SSRmn寄存器的BFFmn位被设置为0，并允许通道n接收数据。	这是为了防止在处理错误的过程中完成了下一次接收时所发生的溢出错误。
读取串行状态寄存器mn (SSRmn)。		识别出错误类型，并用读取值来清除错误标志。
将1写入串行标志清除触发寄存器mn (SIRmn)。	错误标志被清除。	将SSRmn寄存器的读取值直接写入SIRmn寄存器，从而可以只清除读取时的错误。

图 15-97. 发生帧错误时的处理步骤

软件操作	硬件状态	备注
读取串行数据寄存器 mn (SDRmn)。	SSRmn寄存器的BFFmn位被清除为0，并允许通道n接收数据。	这是为了防止在处理错误的过程中完成了下一次接收时所发生的溢出错误。
读取串行状态寄存器mn (SSRmn)。		识别出错误类型，并用读取值来清除错误标志。
写入串行标志清除触发寄存器mn (SIRmn)。	错误标志被清除。	将SSRmn寄存器的读取值直接写入SIRmn寄存器，从而可以只清除读取时的错误。
将串行通道停止寄存器m(STm)的STmn位设置为1。	串行通道允许状态寄存器m (SEm)的SEmn位被清除为0，通道n停止工作。	
与其他通信方同步		起始位的偏移会使系统判断为出现了帧错误，因此重建与其他通信方的同步并恢复通信。
将串行通道开始寄存器m(SSm)的SSmn位设置为1。	串行通道允许状态寄存器m (SEm)的SEmn位被设置为1，并允许操作通道n。	

备注 m: 单元编号(m = 0), n: 通道编号(n = 0 至 3), mn = 00 至 03

15.7 LIN通信操作

15.7.1 LIN发送

在 UART 发送中，UART0 支持 LIN 通信。

LIN 发送使用单元 0 的通道 1。

UART	UART0	UART1
支持LIN通信	支持	不支持
目标通道	SAU1的通道0	-
使用引脚	TxD0	-
中断	INTST0	-
	可选择发送结束中断(单发送模式时)或缓冲器空中断(连续发送模式时)	
错误检测标志	无	
传送数据长度	8位	
传送速率	Max. $f_{MCK}/6$ [bps] (SDR10 [15:9] = 2或以上), Min. $f_{CLK}/(2 \times 2^{15} \times 128)$ [bps] *	
数据相位	正相输出 (默认: 高电平) 反相输出 (默认: 低电平)	
奇偶校验位	无奇偶校验位	
停止位	附加1位	
数据方向	LSB优先	

注 在满足以上条件及电特性中的外围功能特性(参阅第三十二章 电特性)的范围内使用此操作。此外，LIN 通信中通常使用 2.4/9.6/19.2 kbps。

备注 f_{MCK} : 目标通道的操作时钟频率

f_{CLK} : 系统时钟频率

LIN 是 Local Interconnect Network 的简称，是一种低速（1 至 20 kbps）串行通信协议，有助于降低汽车网络的成本。

LIN 通信为单主通信，一台主设备最多可以连接 15 台从设备。

LIN 从设备用于控制开关、传动装置和传感器，这些装置通过 LIN 连接至主设备。

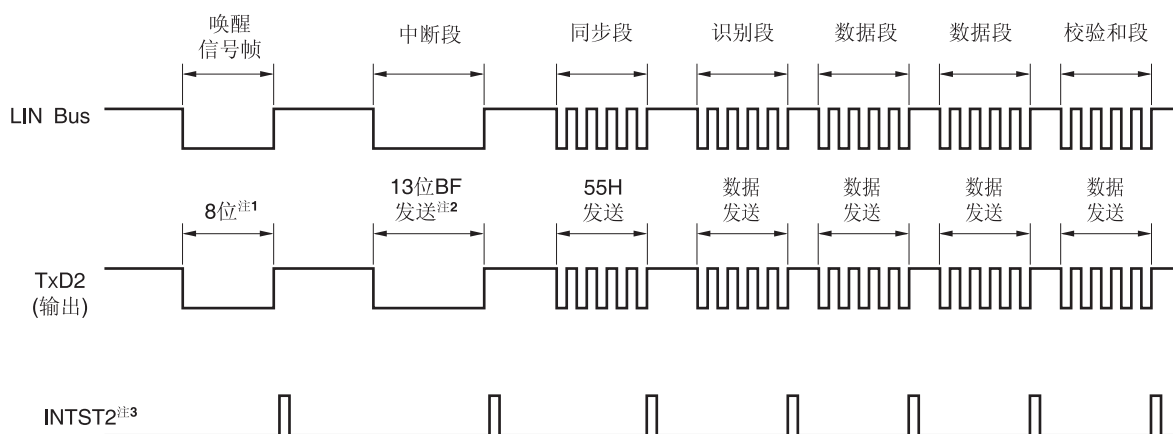
通常，主设备连接至例如 CAN（Controller Area Network）之类的网络。

LIN-BUS 是单线总线，通过符合 ISO9141 的收发器连接有节点。

根据 LIN 协议，主设备通过附加波特率信息来发送帧。从设备接收该帧，并修正来自主设备的波特率误差。如果从设备的波特率误差在±15% 以内，则可建立通信。

LIN 的主发送操作的概况如图 15-98 所示。

图 15-98. LIN 的主发送操作



注 1. 执行 80H 的数据发送。

2. 间隔段被定义为 13 位宽度并输出低电平。当主设备传送的波特率为 N [bps] 时，间隔段的波特率按以下方式计算。

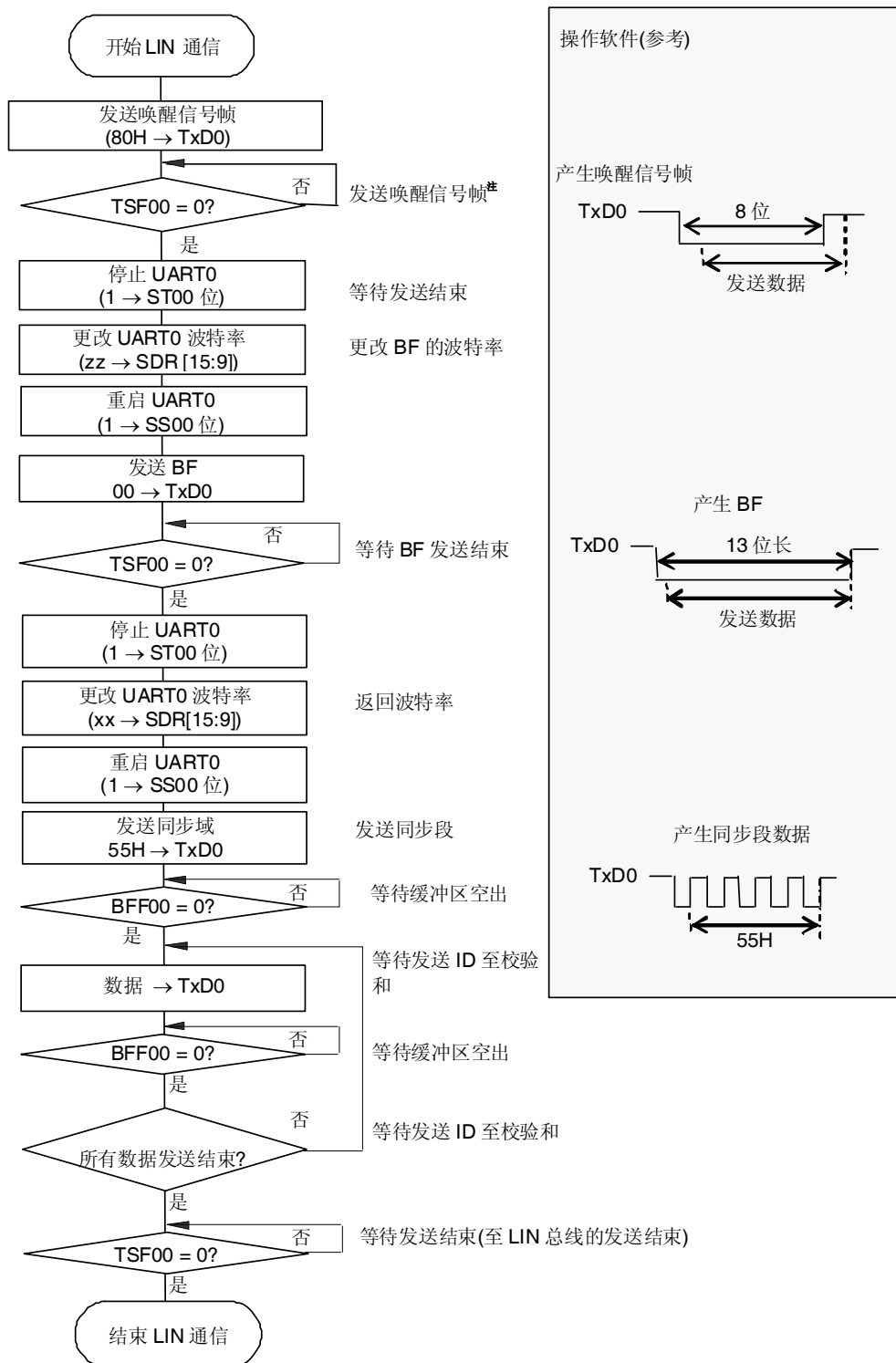
$$(\text{间隔段的波特率}) = 9/13 \times N$$

通过以该波特率发送 00H 的数据来产生间隔段。

3. 发送结束时输出 INTST0。

备注 段之间的间隔由软件控制。

图 15-99. LIN 发送的流程图



注 仅限当 LIN-bus 从休眠状态启动时。

备注 完成了 UART 的初始设置，从发送允许状态开始的流程。

15.7.2 LIN接收

在 UART 接收中，UART0 支持 LIN 通信。

LIN 接收使用单元 1 的通道 1。

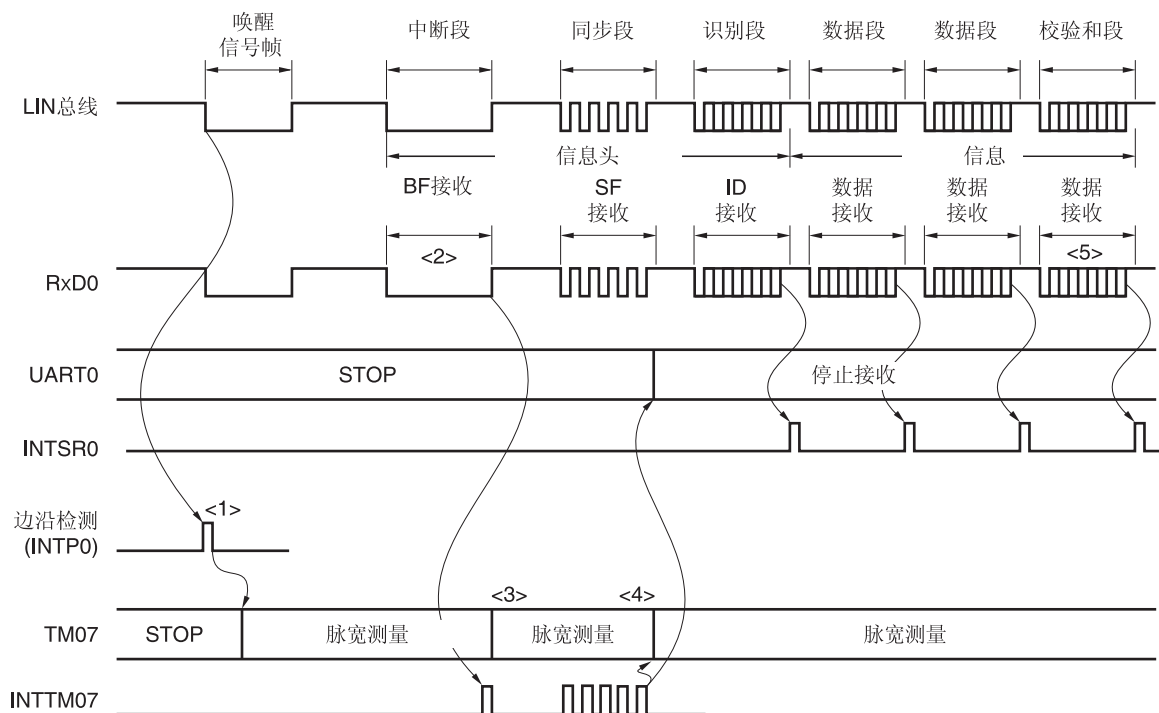
UART	UART0	UART1	UART2	UART3
支持LIN通信	不支持	不支持	支持	不支持
目标通道	通道1	-		-
使用引脚	RxD0	-		-
中断	INTSR0	-		-
	仅限传送结束中断(禁止设置缓冲器空中断。)			
错误中断	INTSRE0	-		-
错误检测标志	<ul style="list-style-type: none"> • 帧错误检测标志 (FEF11) • 溢出错误检测标志 (OVF11) 			
传送数据长度	8位			
传送速率	Max. $f_{MCK}/6$ [bps] (SDR11 [15:9] = 2或以上), Min. $f_{CLK}/(2 \times 2^{15} \times 128)$ [bps] *			
数据相位	正相输出 (默认: 高电平) 反相输出 (默认: 低电平)			
奇偶校验位	无奇偶校验位(不检查奇偶校验位。)			
停止位	检查第1位			
数据方向	LSB优先			

注 在满足以上条件及电特性中的外围功能特性(参阅第三十二章 电特性)的范围内使用此操作。

备注 f_{MCK} : 目标通道的操作时钟频率
 f_{CLK} : 系统时钟频率

LIN 的接收操作的概况如图 15-100 所示。

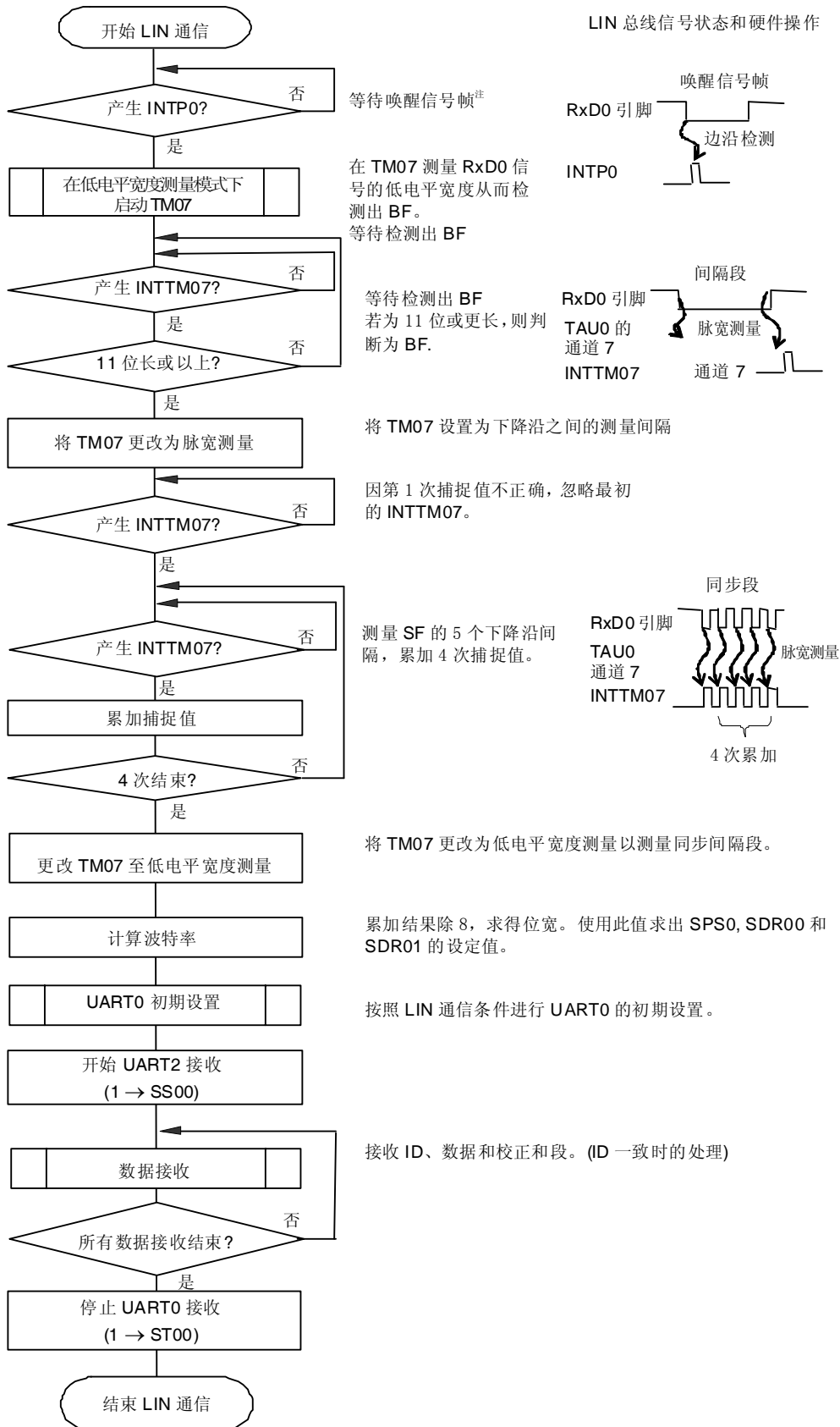
图 15-100. LIN 的接收操作



信号处理的流程如下所示。

- <1> 检测唤醒信号的方法是检测某个引脚上的中断边沿 (INTP0)。当检测出唤醒信号时，将 TM07 设置为脉冲宽度测量，以测量 BF 信号的低电平宽度。进入 BF 信号接收等待状态。
- <2> TM07 在检测到 BF 信号的下降沿时开始测量低电平宽度，在检测到 BF 信号的上升沿时捕捉数据。依据捕捉到的数据来判断是否为 BF 信号。
- <3> BF 信号正常接收时，将 TM07 设置为脉冲间隔测量并对同步段的 RxD0 信号下降沿之间的间隔进行四次测量。
- <4> 当 BF 接收正确完成时，启动定时器阵列单元的通道 7，并测量同步段的位间隔（脉冲宽度）(参阅 6.7.3 作为输入脉冲间隔测量的操作)。
- <5> 根据同步段 (SF) 的位间隔计算波特率误差。停止 UART2 一次，并调节（复位）波特率。
- <6> 通过软件来辨认校验和段。另外，校验和段接收完成之后初始化 UART0 的操作以及此时设置为 BF 接收等待状态的操作也应该通过软件执行。

图 15-101. LIN 接收的流程图



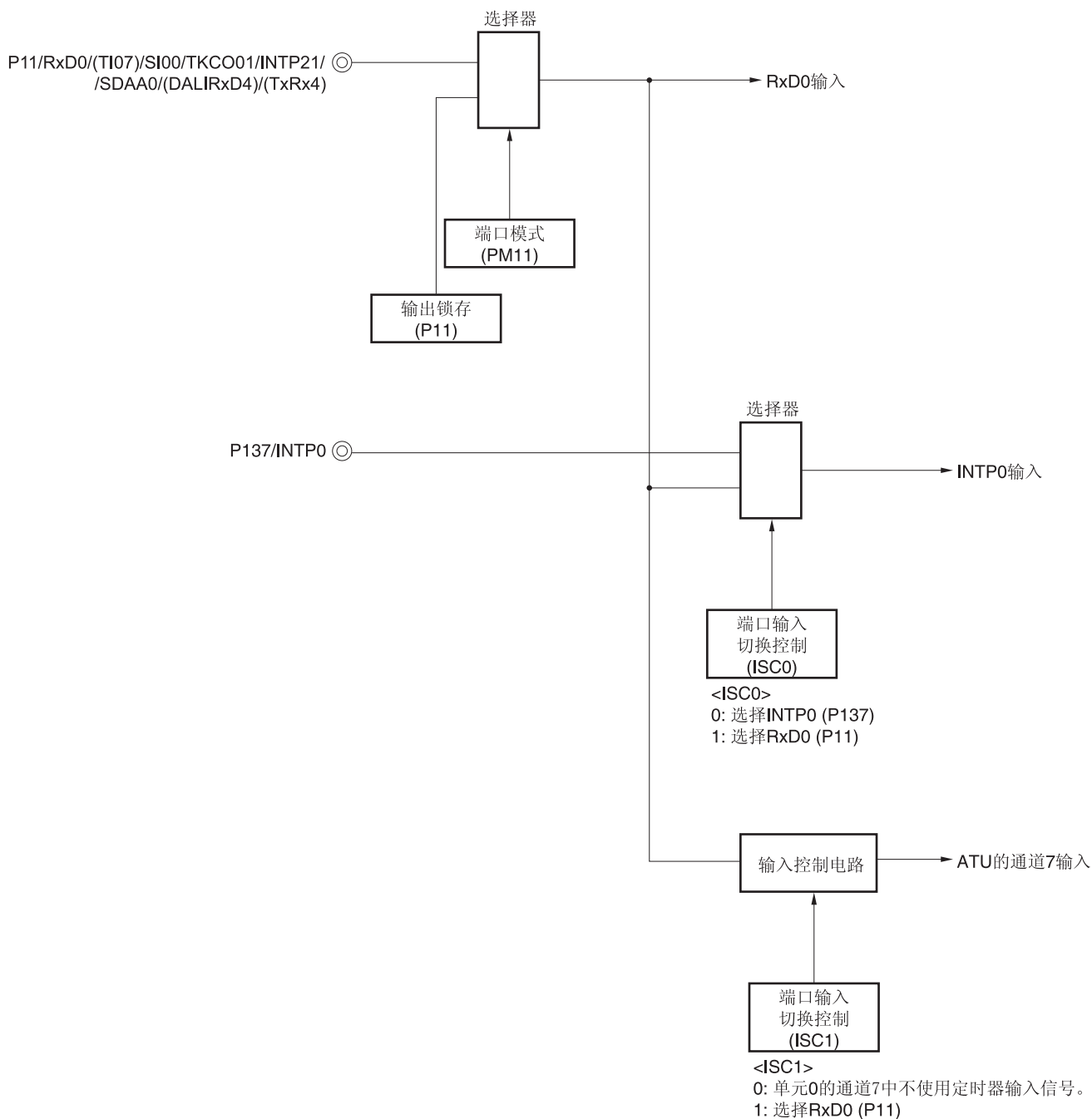
注 仅在休眠状态时需要。

用于操作 LIN 接收的端口的配置如图 15-102 和图 15-103 所示。

通过检测外部中断(INTP0)的边沿来接收从 LIN 主发送的唤醒信号。可以用定时器阵列单元 0 的外部事件捕捉操作来测量从 LIN 主发送的同步段的长度，从而计算出波特率误差。

通过端口输入切换控制(ISC0/ISC1)，可以将接收用端口输入(RxD0)的输入源输入至外部中断引脚(INTP0) 和定时器阵列单元。

图 15-102. LIN 的接收操作作用的端口配置(30, 32, 36, 40 引脚产品时)



备注 ISC0, ISC1: 输入切换控制寄存器(ISC)的位 0 和 1 (参阅图 15-19。)

用于 LIN 通信操作的外围功能如下所示。

<所用外围功能>

- 外部中断 (INTP0)；唤醒信号检测
用途：检测唤醒信号的边沿和检测通信的开始
- 定时器阵列单元的通道 7；波特率误差检测，间隔段检测
用途：检测同步段(SF)的长度，并除以位数以检测波特率误差(在捕捉模式下测量 RxD2 输入边沿的间隔)。
测量低电平宽度，判断是否为间隔段 (BF)。
- 串行阵列单元 0 (SAU0)的通道 0 和 1(UART0)

15.8 DMX512 通信操作

在 UART 接收中，UART0 支持 DMX512 通信。

DMX512 接收使用通道 1。

UART	UART0	UART1
支持DMX512通信	支持	不支持
目标通道	通道1	-
使用引脚	RxD0	-
中断	INTSR0	-
	仅限传送结束中断(禁止设置缓冲器空中断。)	
错误中断	INTSRE0	-
错误检测标志	<ul style="list-style-type: none"> • 帧错误检测标志(FEF01) • 溢出错误检测标志(OVF01) 	
传送数据长度	8位	
传送速率	250 [kbps]	
数据相位	正相输出 (默认: 高电平)	
奇偶校验位	无奇偶校验位(不检查奇偶校验位。)	
停止位	附加2位	
数据方向	MSB优先或LSB优先	

关于 DMX512 的接收操作，请参阅 **16.11 DMX512 通信操作**。

第十六章 串行阵列单元 4 (DALI/UART4)

串行阵列单元 4 的通道 0 和 1 用作 DALI/UART4 功能支持 DALI 通信。与 UART0 和 UART1 一样，DALI/UART4 也可用作 UART 通信。

16.1 串行阵列单元 4 (DALI/UART4)的功能

(1) 异步串行通信(UART)模式

这是一种起止式同步功能，采用两条线路：串行发送数据(DALITxD4)和串行接收数据(DALIRxD4)线路。利用这两条通信线路，各数据帧（由一个起始位、数据、奇偶校验位和停止位构成）在微控制器与其他通信方之间（以内部波特率）异步传输（开始-停止同步通信）。全双工开始-停止同步通信可以利用一个发送专用通道（偶数通道）和一个接收专用通道（奇数通道）来实现。

有关设置详情，请参阅 **16.5.1 UART 发送**或 **16.5.2 UART 接收**。

[数据的发送/接收]

- 支持 7、8 或 9 位的数据长度
- MSB/LSB 选择
- 支持发送/接收数据的电平设置（反相选择）
- 附加奇偶校验位和奇偶校验功能
- 停止位附加、停止位校验功能

[中断功能]

- 传送结束中断/缓冲器空中断
- 出现帧错误、奇偶检验错误或溢出错误时的错误中断

[错误检测标志]

- 帧错误、奇偶校验错误、溢出错误

另外，以下通道的 UART 支持 SNOOZE 模式。当在 STOP 模式下检测到 RxD 输入时，借助 SNOOZE 模式，不需要 CPU 动作就可以接收数据。对应异步接收。

(2) DALI 模式

该模式时执行 DALI (Digital Addressable Lighting Interface)的从属发送/接收。

有关详情, 请参阅 **16.6.1 DALI 发送**或 **16.6.2 DALI 接收**。

备注 DALI 是一项国际开放标准照明控制通信协议, 主要用作多支荧光灯或 LED 照明的调光控制。DALI 通过最大 64 个短地址和 16 个组地址构成网络, 在一个主和一个从属或者多个从属之间执行半双工命令通信。
DALI 命令用于以 8 位精度设置调光控制等级和保存或切换最多 16 种的任意调光控制等级。

[数据的发送/接收]

- 发送 8、16 或 24 位长度的数据
- 接收 16、17 或 24 位长度的数据
- MSB 优先
- 支持发送/接收数据的电平设置 (反相选择)
- 停止位附加 2 位、停止位校验功能

[中断功能]

- 传送结束中断
- 接收中断
- 出现曼彻斯特帧错误、帧错误或溢出错误时的错误中断

[错误检测标志]

- 曼彻斯特帧错误、帧错误、溢出错误

(3) SNOOZE 模式

DALI 接收以及 UART 接收(通道 1)支持 SNOOZE 模式。借助 SNOOZE 模式, 在 STOP 模式下检测到 DALIRxD4 引脚输入时不需要 CPU 而执行数据接收。

注意事项 当 DALI/UART4 被设置为 SNOOZE 模式时, CSI00、UART0 和 A/D 转换器不能被设置为 SNOOZE 模式。可以同时将 CSI00、UART0 和 A/D 转换器设置为 SNOOZE 模式。

(4) 单线数据模式

在此半双工通信模式下, 同一个引脚被兼用为数据发送引脚和接收引脚。TxRx4 引脚用作发送/接收引脚。有关详情, 请参阅 **16.8 单线数据模式**。

16.2 串行阵列单元 4 (DALI/UART4)的配置

定时器阵列单元 4 (DALI/UART4)包含以下硬件。

图 16-1. 串行阵列单元 4 (DALI/UART4)的配置

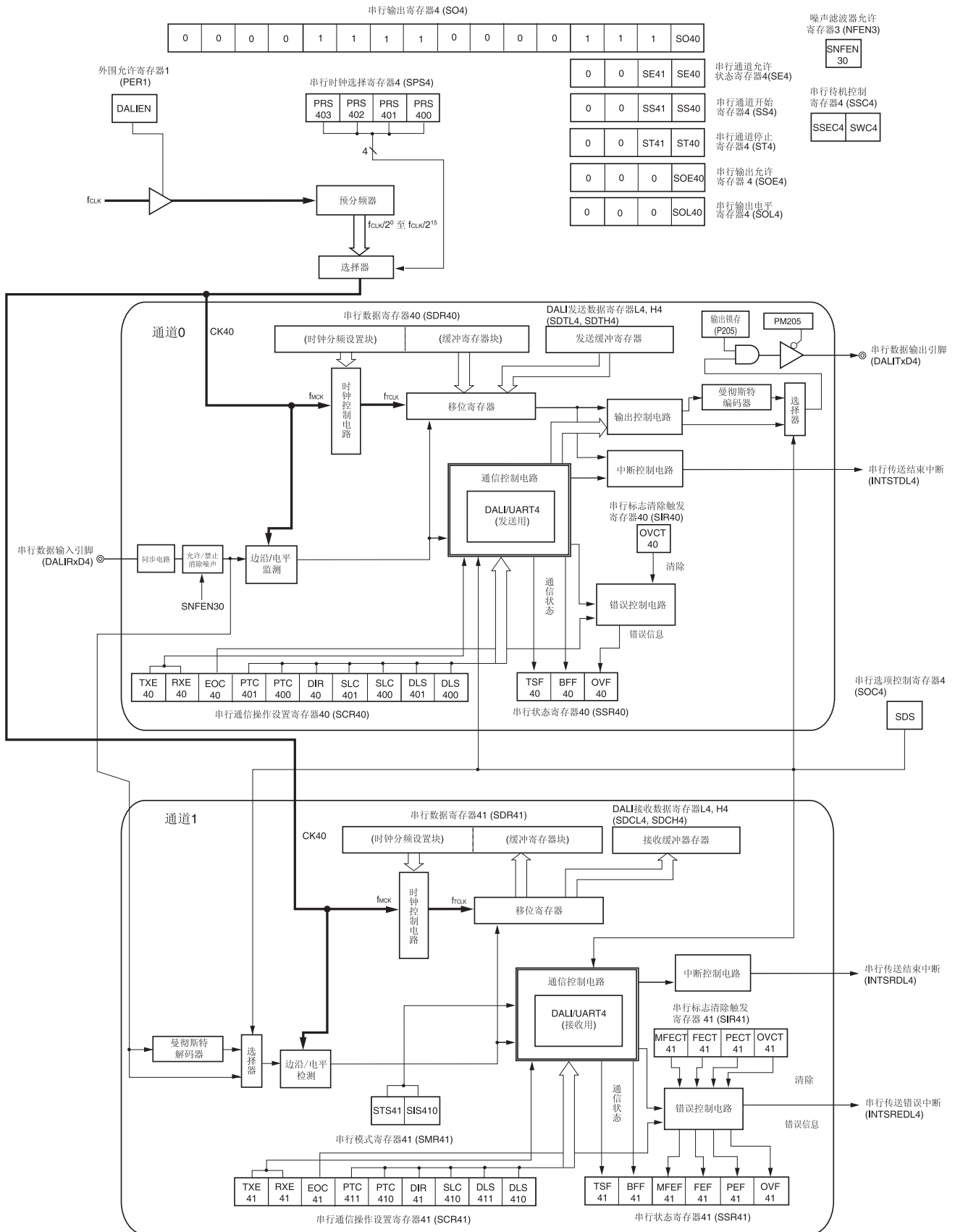
项目	配置
移位寄存器	9 位
缓冲寄存器	串行数据寄存器 40 (SDR40)的低 9 位 ^注 DALI 发送数据寄存器 H4, L4 (SDTH4, SDTL4) DALI 接收数据寄存器 H4, L4 (SDCH4, SDCL4)
串行数据输入	DALIRxD4 引脚
串行数据输出	DALITxD4 引脚
串行数据 I/O	TxRx4 引脚
控制寄存器	<p><单元设置块的寄存器></p> <ul style="list-style-type: none"> • 外围允许寄存器 1 (PER1) • 串行时钟选择寄存器 4 (SPS4) • 串行通道允许状态寄存器 4 (SE4) • 串行通道开始寄存器 4 (SS4) • 串行通道停止寄存器 4 (ST4) • 串行输出允许寄存器 4 (SOE4) • 串行输出寄存器 4 (SO4) • 串行输出电平寄存器 4 (SOL4) • 串行待机控制寄存器 4 (SSC4) • 单线 UART 控制寄存器(SUCTL) • 噪声滤波器允许寄存器 3 (NFEN3) • 串行选项控制寄存器 4 (SOC4) <p><每个通道的寄存器></p> <ul style="list-style-type: none"> • 串行数据寄存器 40, 41 (SDR40, SDR41) • 串行模式寄存器 40, 41 (SMR40, SMR41) • 串行通信操作设置寄存器 40, 41 (SCR40, SCR41) • 串行状态寄存器 40, 41 (SSR40, SSR41) • 串行标志清除触发寄存器 40, 41 (SIR40, SIR41) <ul style="list-style-type: none"> • 端口输出模式寄存器 20 (POM20) • 端口模式寄存器 20 (PM20) • 端口寄存器 20 (P20) • 外围输入/输出重定向寄存器(PIOR1)

注 依据通信模式，串行数据寄存器 41 (SDR41)的低 8 位可以按以下 SFR 名称读取或写入。

- UART 接收 ... RXD4 (UART4 接收数据寄存器)
- UART 发送 ... TXD4 (UART4 发送数据寄存器)

串行阵列单元 4 (DALI/UART4)的框图如图 16-1 所示。

图 16-1. 串行阵列单元 4 (DALI/UART4)的框图



(1) 移位寄存器

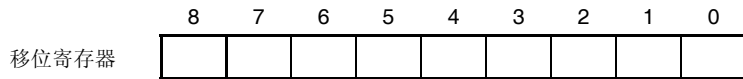
这是一个 9 位寄存器，可将并行数据转换为串行数据，反之亦然。

接收期间，将输入至串行输入引脚的数据转换为并行数据。

发送数据时，将设置到该寄存器的值作为串行数据从串行输出引脚输出。

不能用程序直接操作移位寄存器。

使用串行数据寄存器 4n (SDR4n)的低 9 位或 DALI 发送数据寄存器 H4, L4 (SDTH4, SDTL4)、DALI 接收数据寄存器 H4, L4 (SDCH4, SDCL4)来读取或写入移位寄存器。



(2) 串行数据寄存器 4n (SDR4n)的低 9 位

SDR4n 寄存器是通道 n 的发送/接收数据寄存器（16 位）。SDR4n 的位 8 至位 0（低 9 位）作为发送/接收缓冲寄存器，位 15 至位 9 作为设置操作时钟(fmck) 分频比的寄存器。

在 UART 模式下接收数据时，由移位寄存器转换的并行数据被存储于低 9 位。在 UART 模式下发送数据时，把传送到移位寄存器的发送值设置到低 9 位。

依据串行通信操作设置寄存器 4n (SCR4n)位 0 和位 1 (DLS4n0、DLS4n1)的设置，存储于该寄存器低 9 位的数据的情况如下所示（与数据输出顺序无关）：

- 9 位数据长度 (存储于 SDR4n 寄存器的位 0 至位 8)
- 7 位数据长度 (存储于 SDR4n 寄存器的位 0 至位 6)
- 8 位数据长度 (存储于 SDR4n 寄存器的位 0 至位 7)

SDR4n 寄存器可以按照 16 位单位进行读取或写入。

依据通信模式，SDR4n 寄存器的低 9 位可以按以下 SFR 名称读取或写入。

- UART 接收 ... RXD4 (UART4 接收数据寄存器)
- UART 发送 ... TXD4 (UART4 发送数据寄存器)

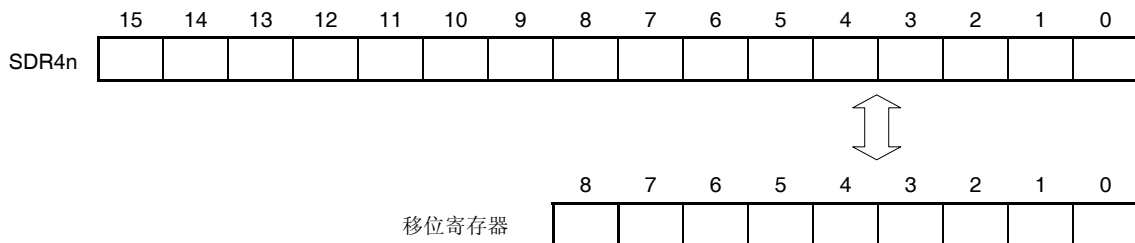
产生复位信号后，SDR4n 寄存器被清除为 0000H。

备注 1. 数据接收完成后，0 将存储于位 0 至位 8 中超过数据长度部分的位。

2. n: 通道编号(n = 0, 1), q: UART 编号(q = 4)

图 16-2. 串行数据寄存器 4n (SDR4n)的格式

地址: F0570H, F0571H (SDR40), F0572H, F0573H (SDR41) 复位后: 0000H R/W



备注 关于 SDR4n 寄存器的高 7 位功能，请参阅 16.3 控制串行阵列单元 4 (DALI/UART4)的寄存器。

(3) DALI 发送数据寄存器 H4, L4 (SDTH4, SDTL4)

SDTH4、SDTL4 寄存器是 DALI 的发送数据寄存器(16 位)。

关于 DALI 通信期间发送至移位寄存器的发送数据，位 15 至 0 被存储在 SDTL4 寄存器，位 23 至 16 被存储在 SDTH4 寄存器。

依据串行通信操作设置寄存器 4n (SCR4n)位 0 和位 1 (DLS4n0、DLS4n1)的设置，存储于该寄存器的数据的情况如下所示（与数据输出顺序无关）：

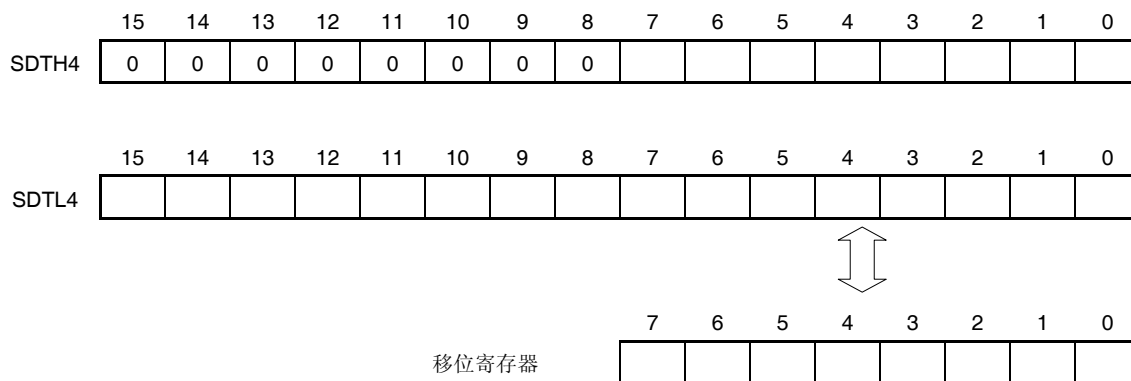
- 24 位数据长度 (存储于 SDTH4 寄存器的位 0 至位 7 和 SDTL4 寄存器的位 0 至位 15)
- 16 位数据长度 (存储于 SDTL4 寄存器的位 0 至位 15)
- 8 位数据长度 (存储于 SDTL4 寄存器的位 0 至位 7)

使用 16 位存储器操作指令设置 SDTH4 和 SDTL4 寄存器。

产生复位信号后，SDTH4 和 SDTL4 寄存器被清除为 0000H。

图 16-3. DALI 发送数据寄存器 H4, L4 (SDTH4, SDTL4)的格式

地址：F0578H (SDTL4), F057AH (SDTH4) 复位后：0000H R/W



- 注意事项**
1. 必须将 SDTH4 寄存器的位 8 至 15 清除为 0。
 2. 当数据被写入到 SDTL4 寄存器时开始发送数据。如果设置的数据长度超过 16 位，将首先写入到 SDTH4 寄存器，然后写入到 SDTL4 寄存器。

备注 n: 通道编号 (n = 0, 1)

(4) DALI 接收数据寄存器 H4, L4 (SDCH4, SDCL4)

SDCH4、SDCL4 寄存器是 DALI 的接收数据寄存器(16 位)。

接收 DALI 时，由移位寄存器转换的并行数据从 SDCL4 寄存器的 LSB 开始写入。

依据串行通信操作设置寄存器 4n (SCR4n)位 0 和位 1 (DLS4n0、DLS4n1)的设置，存储于该寄存器的数据的情况如下所示（与数据输出顺序无关）：

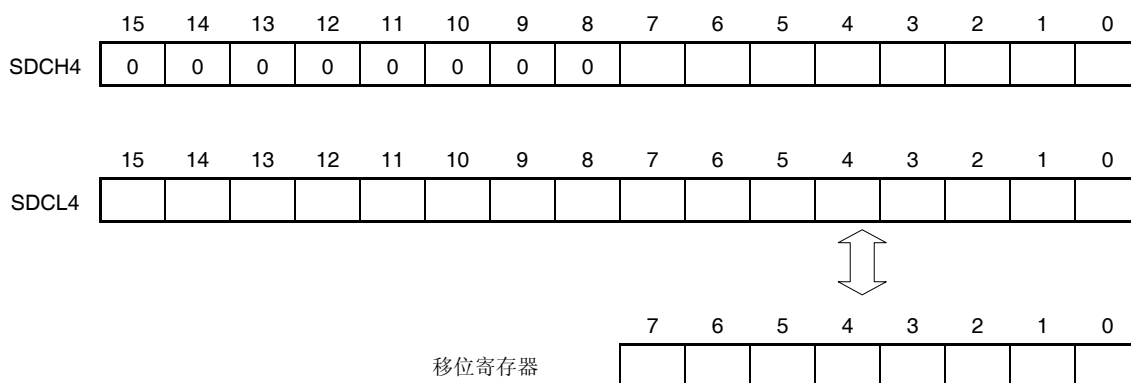
- 24 位数据长度 (存储于 SDCH4 寄存器的位 0 至位 7 和 SDCL4 寄存器的位 0 至位 15)
- 17 位数据长度 (存储于 SDCH4 寄存器的位和 SDCL4 寄存器的位 0 至位 15)
- 16 位数据长度 (存储于 SDCL4 寄存器的位 0 至位 15)

使用 16 位存储器操作指令读取 SDCH4 和 SDCL4 寄存器。

产生复位信号后，SDCH4 和 SDCL4 寄存器被清除为 0000H。

图 16-4. DALI 接收数据寄存器 H4, L4 (SDCH4, SDCL4)的格式

地址: F057CH (SDCL4), F057EH (SDCH4) 复位后: 0000H R



注意事项 SDCH4 寄存器的位 8 至位 15 固定为 0。

- 备注**
1. 接收结束后，超过数据长度的部分将存储为 0。
 2. n: 通道编号 (n = 0, 1)

16.3 控制串行阵列单元 4 (DALI/UART4)的寄存器

串行阵列单元 4 (DALI/UART4)由下列寄存器控制。

- 外围允许寄存器 1(PER1)
- 串行时钟选择寄存器 4 (SPS4)
- 串行模式寄存器 40, 41 (SMR40, SMR41)
- 串行通信操作设置寄存器 40, 41 (SCR40, SCR41)
- 串行数据寄存器 40, 41 (SDR40, SDR41)
- 串行标志清除触发寄存器 40, 41 (SIR40, SIR41)
- 串行状态寄存器 40, 41 (SSR40, SSR41)
- 串行选项控制寄存器 4 (SOC4)
- 串行通道开始寄存器 4 (SS4)
- 串行通道停止寄存器 4 (ST4)
- 串行通道允许状态寄存器 4 (SE4)
- 串行输出允许寄存器 4 (SOE4)
- 串行输出电平寄存器 4 (SOL4)
- 串行输出寄存器 4 (SO4)
- 串行待机控制寄存器 4 (SSC4)
- 单线 UART 控制寄存器(SUCTL)
- 噪声滤波器允许寄存器 3 (NFEN3)
- 端口输出模式寄存器 1, 20 (POM1, POM20)
- 端口模式寄存器 1, 20 (PM1, PM20)
- 端口寄存器 1, 20 (P1, P20)
- 外围输入/输出重定向寄存器(PIOR1)

(1) 外围允许寄存器 1(PER1)

PER1 用于允许或禁止向外围硬件供应时钟。并且中止向未使用的硬件供应时钟，以减少电力消耗和噪声。

当使用串行阵列单元 4(DALI/UART4)时，必须将该寄存器的位 0 (DALIEN)设置为 1。

PER1 寄存器可利用一条 1 位或 8 位存储器操作指令进行设置。

产生复位信号后，PER1 寄存器被清除为 00H。

图 16-5. 外围允许寄存器 1 (PER1)的格式

地址: F0508H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	<0>
PER1	0	0	0	0	0	0	0	DALIEN

DALIEN	串行阵列单元4 (DALI/UART4)输入时钟供应的控制
0	停止输入时钟的供应。 <ul style="list-style-type: none"> 不可写入用于串行阵列单元4 (DALI/UART4)的SFR。 串行阵列单元4 (DALI/UART4)处于复位状态。
1	允许输入时钟供应。 <ul style="list-style-type: none"> 可以读取和写入用于串行阵列单元4 (DALI/UART4)的SFR。

- 注意事项**
1. 设置串行阵列单元 4 (DALI/UART4)时，必须首先将 DALIEN 位设置为 1。如果 DALIEN = 0，对于串行阵列单元 4 (DALI/UART4)的控制寄存器进行的写入操作被忽略，而且该寄存器的读取值总为初始值（串行待机控制寄存器 4 (SSC4)、噪声滤波器允许寄存器 3 (NFEN3)、端口输出模式寄存器 20 (POM20)、端口模式寄存器 20 (PM20)和端口寄存器 20 (P20)除外）。
 2. 在将 DALIEN 位设置为 1 后，需经过 f_{CLK} 的 4 个或更多时钟后才可以设置串行时钟选择寄存器 4 (SPS4)。
 3. 必须将位 1 至 7 清除为 0。

(2) 串行时钟选择寄存器 4 (SPS4)

SPS4 寄存器为 16 位寄存器，用于选择供应给各个通道的操作时钟(CK40)。

当寄存器在运行（当 SE4n = 1）时，禁止改写 SPS4 寄存器。

当选择 SPS4 = 0003H 至 000FH 作为操作时钟时，即便 PLL 功能从"使用"切换到"不使用"，所选择时钟频率也保持相同。因此，当 CPU 从正常操作模式切换至待机模式时，供应给串行阵列单元 4 的时钟被允许继续并保持可接收，并且可切换 PLL 功能的使用/不使用设置。

使用 16 位存储器操作指令设置 SPS4 寄存器。

通过 SPS4L 使用 8 位存储器操作指令设置 SPS4 寄存器的低 8 位。

产生复位信号后，SPS4 寄存器被清除为 0000H。

图 16-6. 串行时钟选择寄存器 4 (SPS4)的格式

地址: F05A6H, F05A7H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPS4	0	0	0	0	0	0	0	0	0	0	0	0	PRS 403	PRS 402	PRS 401	PRS 400

PRS 403	PRS 402	PRS 401	PRS 400	操作时钟(CK40) ^注 的选择					
				f _{CLK} = 4 MHz (不使用PLL)		f _{CLK} = 16 MHz (使用PLL)		f _{CLK} = 32 MHz (使用PLL)	
0	0	0	0	f _{CLK}	4 MHz	f _{CLK}	16 MHz	f _{CLK}	32 MHz
0	0	0	1	f _{CLK}	4 MHz	f _{CLK}	16 MHz	f _{CLK} /2	16 MHz
0	0	1	0	f _{CLK}	4 MHz	f _{CLK} /2	8 MHz	f _{CLK} /2 ²	8 MHz
0	0	1	1	f _{CLK}	4 MHz	f _{CLK} /2 ²	4 MHz	f _{CLK} /2 ³	4 MHz
0	1	0	0	f _{CLK} /2	2 MHz	f _{CLK} /2 ³	2 MHz	f _{CLK} /2 ⁴	2 MHz
0	1	0	1	f _{CLK} /2 ²	1 MHz	f _{CLK} /2 ⁴	1 MHz	f _{CLK} /2 ⁵	1 MHz
0	1	1	0	f _{CLK} /2 ³	500 kHz	f _{CLK} /2 ⁵	500 kHz	f _{CLK} /2 ⁶	500 kHz
0	1	1	1	f _{CLK} /2 ⁴	250 kHz	f _{CLK} /2 ⁶	250 kHz	f _{CLK} /2 ⁷	250 kHz
1	0	0	0	f _{CLK} /2 ⁵	125 kHz	f _{CLK} /2 ⁷	125 kHz	f _{CLK} /2 ⁸	125 kHz
1	0	0	1	f _{CLK} /2 ⁶	62.5 kHz	f _{CLK} /2 ⁸	62.5 kHz	f _{CLK} /2 ⁹	62.5 kHz
1	0	1	0	f _{CLK} /2 ⁷	31.3 kHz	f _{CLK} /2 ⁹	31.3 kHz	f _{CLK} /2 ¹⁰	31.3 kHz
1	0	1	1	f _{CLK} /2 ⁸	15.6 kHz	f _{CLK} /2 ¹⁰	15.6 kHz	f _{CLK} /2 ¹¹	15.6 kHz
1	1	0	0	f _{CLK} /2 ⁹	7.81 kHz	f _{CLK} /2 ¹¹	7.81 kHz	f _{CLK} /2 ¹²	7.81 kHz
1	1	0	1	f _{CLK} /2 ¹⁰	3.91 kHz	f _{CLK} /2 ¹²	3.91 kHz	f _{CLK} /2 ¹³	3.91 kHz
1	1	1	0	f _{CLK} /2 ¹¹	1.95 kHz	f _{CLK} /2 ¹³	1.95 kHz	f _{CLK} /2 ¹⁴	1.95 kHz
1	1	1	1	f _{CLK} /2 ¹²	977 Hz	f _{CLK} /2 ¹⁴	977 Hz	f _{CLK} /2 ¹⁵	977 Hz
其它				禁止设置					

注 更改被选为 f_{CLK} 的时钟时（通过更改系统时钟控制寄存器(CKC)的值），必须在停止串行阵列单元 4 (DALI/UART4)操作（串行通道停止寄存器 4 (ST4) = 000FH）之后才可以执行。

- 注意事项
1. 必须将位 15 至 4 清除为 0。
 2. 将 PER1 寄存器的位 0 (DALIEN)设置为 1 后，需经过 4 个或更多 f_{CLK} 时钟后才可以设置串行时钟选择寄存器 4 (SPS4)。

备注 f_{CLK}: CPU/外围硬件时钟频率
f_{SUB}: 副系统时钟频率

(3) 串行模式寄存器 4n (SMR4n)

SMR4n 寄存器用于设置通道 n 的操作模式。该寄存器也可用于设置开始触发和选择中断源。该寄存器还以使接收数据的电平反相。

当寄存器在运行（当 SE4n = 1）时，禁止改写 SMR4n 寄存器。然而，可以在操作过程中改写 MD4n0 位。

使用 16 位存储器操作指令设置 SMR4n 寄存器。

产生复位信号后，SMR4n 寄存器被设置为 0020H。

图 16-7. 串行模式寄存器 4n (SMR4n)的格式

地址： F0590H, F0591H (SMR40), F0592H, F0593H (SMR41) 复位后： 0020H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMR4n	0	0	0	0	0	0	0	STS 4n [※]	0	SIS 4n0 [※]	1	0	0	0	1	MD 4n0

STS 4n [※]	开始触发源的选择
0	仅限软件触发有效 (UART发送时选择)。
1	DALIRxD4引脚的有效沿 (UART接收时选择)。
将SS4寄存器设置为1并且满足以上条件时，开始传送。	

SIS 4n0 [※]	UART模式下，通道n接收数据的电平反相的控制
0	下降沿被检测为起始位。 按原样捕捉被输入的通信数据。
1	上升沿被检测为起始位。 将被输入的通信数据反相并捕捉。

MD 4n0	通道n中断源的选择
0	传送结束中断
1	缓冲器空中断 (发生于把数据从SDR4n寄存器传送至移位寄存器时。)
连续发送时，将MD4n0位设置为1，并且在SDR4n的数据为空时，写入下一个发送数据。	

注 仅限 SMR41 寄存器。

注意事项 必须将位 15 至 9、7、4 至 2 (或者 SMR40 寄存器的位 15 至 6、4 至 2)清除为 0。必须将位 5 和 1 设置为 1。

备注 n: 通道编号(n = 0, 1), q: UART 编号(q = 4)

(4) 串行通信操作设置寄存器 4n (SCR4n)

SCR4n 寄存器是通道 n 的通信操作设置寄存器。该寄存器用于设置数据发送/接收模式、是否屏蔽错误信号、奇偶校验位、起始位、停止位和数据长度。

当寄存器在运行（当 SE4n = 1）时，禁止改写 SCR4n 寄存器。

使用 16 位存储器操作指令设置 SCR4n 寄存器。

产生复位信号后，SCR4n 寄存器被设置为 0087H。

图 16-8. 串行通信操作设置寄存器 4n (SCR4n)的格式(1/2)

地址： F0598H, F0599H (SCR40), F059AH, F059BH (SCR41) 复位后： 0087H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCR4n	TXE 4n	RXE 4n	0	0	0	EOC 4n	PTC 4n1	PTC 4n0	DIR 4n	0	SLC 4n1 ^{#1}	SLC 4n0	0	1	DLS 4n1	DLS 4n0

TXE 4n	RXE 4n	通道n操作模式的设置	
0	0	禁止通信	
0	1	仅接收(SCR41的设置)	
1	0	仅发送(SCR40的设置)	
1	1	发送/接收	

EOC 4n	选择是否屏蔽错误中断信号(INTSREDL4)	
0	屏蔽错误中断INTSREDL4（不屏蔽INTSRDL4）。	
1	允许产生错误中断INTSREDL4（如果出现错误，INTSRDL4将被屏蔽）。	
UART发送期间设置为EOC40 = 0。		

PTC 4n1	PTC 4n0	UART模式下奇偶校验位的设置	
		发送	接收
0	0	不输出奇偶校验位。	无奇偶校验的接收。
0	1	输出零奇偶校验 ^{#2} 。	无奇偶校验判断。
1	0	输出偶数校验。	偶数校验判断。
1	1	输出奇数校验。	奇数校验判断。

- 注 1. 仅限 SCR40 寄存器。
- 2. 不论数据内容如何，始终附加 0。

注意事项 必须将位 3、6、11 至 13(或者 SCR41 寄存器的位 3、5、6、11 至 13)清除为 0。必须将位 2 设置为 1。

备注 n: 通道编号 (n = 0, 1)

图 16-8. 串行通信操作设置寄存器 4n (SCR4n)的格式(2/2)

地址: F0598H, F0599H (SCR40), F059AH, F059BH (SCR41) 复位后: 0087H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCR4n	TXE 4n	RXE 4n	0	0	0	EOC 4n	PTC 4n1	PTC 4n0	DIR 4n	0	SLC 4n1 [※]	SLC 4n0	0	1	DLS 4n1	DLS 4n0

DIR 4n	数据传送序列的选择	
0	MSB优先的输入/输出数据。	
1	LSB优先的输入/输出数据	

SLC 4n1 [※]	SLC 4n0	停止位的设置
0	0	无停止位
0	1	停止位长度 = 1位
1	0	停止位长度 = 2位(仅限SLC401, SLC400)
1	1	禁止设置
选择传送结束中断时, 传送完所有停止位时产生中断。 UART接收时设置为1位(SLC4n1, SLC4n0 = 0, 1)。		

DLS 4n1	DLS 4n0	数据长度的设置	
		UART通信模式	DALI通信模式
0	0	禁止设置	24位数据长度 (存储于SDCH4和SDTH4寄存器的位7至位0以及SDCL4和SDTL4寄存器的位15至位0)
0	1	9位数据长度 (存储于SDR4n寄存器(mn = 40, 41)的位0至位8)	17位数据长度(仅限接收) (存储于SDCH4寄存器的位0以及SDCL4寄存器的位15至位0)
1	0	7位数据长度 (存储于SDR4n寄存器的位0至位6)	16位数据长度 (存储于SDCL4和SDTL4的位15至0)
1	1	8位数据长度 (存储于SDR4n寄存器的位0至位7)	8位数据长度(仅限发送) (存储于SDTL4的位7至0)
其它		禁止设置	

注 仅限 SCR40 寄存器。

注意事项 必须将位 3、6、11 至 13(或者 SCR41 寄存器的位 3、5、6、11 至 13)清除为 0。必须将位 2 设置为 1。

备注 n: 通道编号 (n = 0, 1)

(5) 串行数据寄存器 4n (SDR4n)的高 7 位

SDR4n 寄存器为通道 n 的发送/接收数据寄存器（16 位）。SDR4n 寄存器的位 8 至位 0（低 9 位）作为发送/接收缓冲寄存器，位 15 至位 9 作为设置操作时钟(fmck) 分频比的寄存器。

使用 SDR4n 寄存器的高 7 位设置为操作时钟分频的时钟可用作传送时钟。

SDR4n 寄存器的低 9 位用作发送/接收缓冲寄存器。接收过程中，由移位寄存器转换的并行数据将存储于低 9 位中，在发送过程中，发送至移位寄存器的数据将被设置到低 9 位。

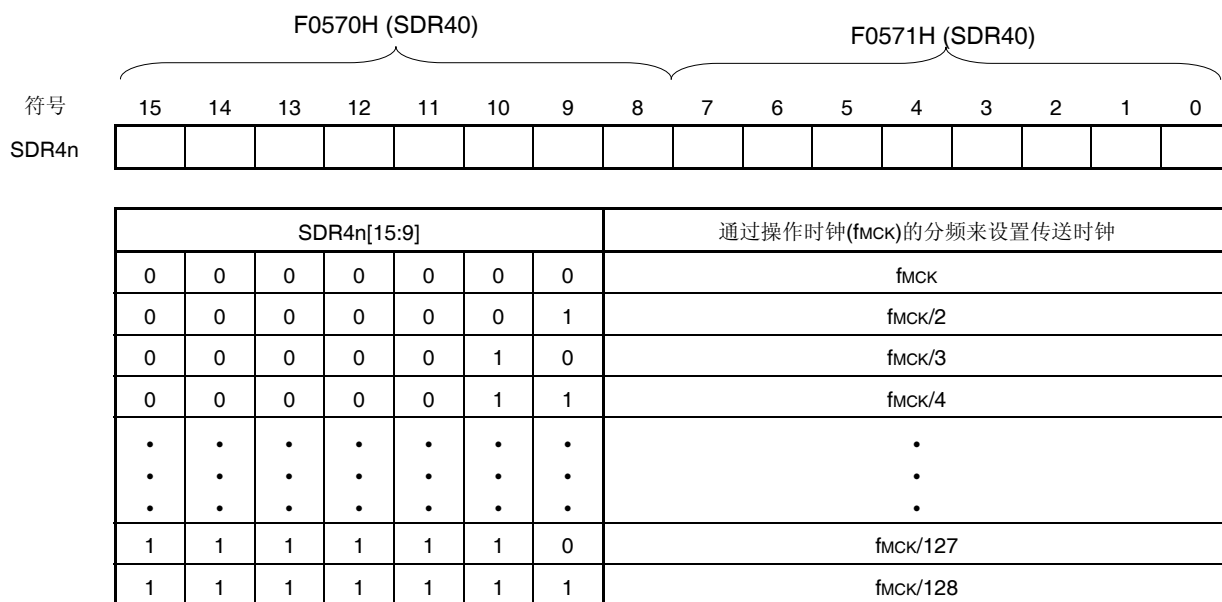
SDR4n 寄存器可以按照 16 位单位进行读取或写入。

然而，仅当停止操作(SE4n = 0)时，才可写入或读取高 7 位。操作过程(SE4n = 1)中，只能将值写入到 SDR4n 寄存器的低 9 位。操作过程中 SDR4n 寄存器的读取值始终为 0。

产生复位信号后，SDR4n 寄存器被清除为 0000H。

图 16-9. 串行数据寄存器 4n (SDR4n)的格式

地址： F0570H, F0571H (SDR40), F0572H, F0573H (SDR41), 复位后： 0000H R/W



- 注意事项**
1. 禁止设置为 SDR4n[15:9] = (0000000B, 0000001B, 0000010B)。
 2. 停止操作 (SE4n = 0)时，不要向低 8 位执行 8 位写入。（如果写入这些位，高 7 位将被清除为 0。）
 3. 如果在 BFF4n = 1 时将数据写入 SDR4n 寄存器，存储于该寄存器中的发送/接收数据将被破坏，并检测到溢出错误(OVE4n = 1)。

- 备注**
1. 关于 SDR4n 寄存器的低 9 位功能，请参阅 16.2 串行阵列单元 4 (DALI/UART4)的配置。
 2. n: 通道编号 (n = 0, 1)

(6) 串行状态寄存器 4n (SSR4n)

SSR4n 寄存器用于指示通道 n 的通信状态和错误发生状态。该寄存器指示的错误为帧错误、曼彻斯特帧错误、奇偶校验错误和溢出错误。

使用 16 位存储器操作指令读取 SSR4n 寄存器。

通过 SSR4nL 使用 8 位存储器操作指令设置 SSR4n 寄存器的低 8 位。

产生复位信号后，SSR4n 寄存器被清除为 0000H。

图 16-10. 串行状态寄存器 4n (SSR4n)的格式(1/3)

地址： F0580H, F0581H (SSR40), F0582H, F0583H (SSR41), 复位后： 0000H R

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSR4n	0	0	0	0	0	0	0	0	MFEF 4n [#]	TSF 4n	BFF 4n	0	0	FEF 4n [#]	PEF 4n [#]	OVF 4n

MFEF 4n [#]	通道n的曼彻斯特帧错误检测标志
0	无错误发生。
1	错误发生(DALI接收期间)。
<清除条件>	
• 将1写入SIR4n寄存器的MFECT4n位。	
<设置条件>	
• 当DALI接收结束时未检测到停止位。	

TSF4n	通道n的通信状态显示标志
0	通信停止或暂停。
1	通信正在进行。
<清除条件>	
• ST4寄存器的ST4n位被设置为1(通信停止)或者SS4寄存器的SS4n位被设置为1(通信暂停)。	
• 通信结束。	
<设置条件>	
• 通信开始。	

注 仅限 SSR41 寄存器。

注意事项 如果在 BFF4n = 1 时将数据写入 SDR4n 寄存器，存储于该寄存器中的发送/接收数据将被破坏，并检测到溢出错误(OVE4n = 1)。

备注 n: 通道编号 (n = 0, 1)

图 16-10. 串行状态寄存器 4n (SSR4n)的格式(2/3)

地址: F0580H, F0581H (SSR40), F0582H, F0583H (SSR41), 复位后: 0000H R

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSR4n	0	0	0	0	0	0	0	0	MFEF 4n [※]	TSF 4n	BFF 4n	0	0	FEF 4n [※]	PEF 4n [※]	OVF 4n

BFF 4n	通道n的缓冲寄存器状态显示标志
0	SDR4n寄存器中不存储有效数据。
1	SDR4n寄存器中存储有效数据。
<p><清除条件></p> <ul style="list-style-type: none"> • 发送时, 从SDR4n寄存器至移位寄存器的发送数据的传送结束。 • 接收时, 从SDR4n寄存器的接收数据的读取结束。 • ST4寄存器的ST4n位被设置为1 (通信停止)或者SS4寄存器的SS4n位被设置为1(允许通信)。 <p><设置条件></p> <ul style="list-style-type: none"> • 当SCR4n寄存器的TXE4n位被设置为1 (各种通信模式下的发送或者发送和接收模式) 时, 发送数据被写入SDR4n寄存器。 • 当SCR4n寄存器的RXE4n位被设置为1 (各种通信模式下的发送或者发送和接收模式) 时, 接收数据被存储于SDR4n寄存器中。 • 发生接收错误。 	

FEF 4n [※]	通道n的帧错误检测标志
0	无错误发生。
1	错误发生 (UART接收期间)。
<p><清除条件></p> <ul style="list-style-type: none"> • 将1写入SIR4n寄存器的FECT4n位。 <p><设置条件></p> <ul style="list-style-type: none"> • 当UART接收结束时未检测到停止位。 	

PEF4 n	通道n的奇偶校验错误检测标志
0	无错误发生。
1	错误发生 (UART接收期间)。
<p><清除条件></p> <ul style="list-style-type: none"> • 将1写入SIR4n寄存器的PECT4n位。 <p><设置条件></p> <ul style="list-style-type: none"> • 当UART接收结束时, 发送数据的奇偶校验和奇偶校验位不匹配(奇偶校验错误)。 	

注 仅限 SSR41 寄存器。

备注 n: 通道编号 (n = 0, 1)

图 16-10. 串行状态寄存器 4n (SSR4n)的格式(3/3)

地址: F0580H, F0581H (SSR40), F0582H, F0583H (SSR41), 复位后: 0000H R

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSR4n	0	0	0	0	0	0	0	0	MFEF 4n [※]	TSF 4n	BFF 4n	0	0	FEF 4n [※]	PEF 4n [※]	OVF 4n

OVF 4n	通道n的溢出错误检测标志
0	无错误发生。
1	错误发生。
<p><清除条件></p> <ul style="list-style-type: none"> • 将1写入SIR4n寄存器的OVCT4n位。 <p><设置条件></p> <ul style="list-style-type: none"> • 当SCR4n寄存器的RXE4n位被设置为1（各种通信模式下的接收或者发送和接收模式）时，即使接收数据存储于SDR4n寄存器中，也不会读取这些数据，而是写入发送数据或下一接收数据。 	

注 仅限 SSR41 寄存器。

备注 n: 通道编号 (n = 0, 1)

(7) 串行标志清除触发寄存器 4n (SIR4n)

SIR4n 寄存器是用于清除通道 n 各个错误标志的触发寄存器。

当该寄存器的各个位(MFECT4n、FECT4n、PECT4n、OVCT4n)被设置为 1 时，串行状态寄存器 4n 的对应位 (MFEF4n、FEF4n、PEF4n、OVF4n)将被清除为 0。由于 SIR4n 寄存器是触发寄存器，当 SSR4n 寄存器的对应位被清除时，SIR4n 寄存器立即被清除。

使用 16 位存储器操作指令设置 SIR4n 寄存器。

通过 SIR4nL 使用 8 位存储器操作指令设置 SIR4n 寄存器的低 8 位。

产生复位信号后，SIR4n 寄存器被清除为 0000H。

图 16-11. 串行标志清除触发寄存器 4n (SIR4n)的格式

地址: F0588H, F0589H (SIR40), F058AH, F058BH (SIR41) 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SIR4n	0	0	0	0	0	0	0	0	MFECT 4n [#]	0	0	0	0	FECT 4n [#]	PECT 4n [#]	OVCT 4n

MFECT 4n [#]	通道n的曼彻斯特帧错误标志的清除触发
0	不清除
1	清除SSR4n寄存器的MFEF4n位为0。

FECT 4n [#]	通道n的帧错误标志的清除触发
0	不清除
1	清除SSR4n寄存器的FEF4n位为0。

PECT 4n [#]	通道n的奇偶校验错误标志的清除触发
0	不清除
1	清除SSR4n寄存器的PEF4n位为0。

OVCT 4n	通道n的溢出错误标志的清除触发
0	不清除
1	清除SSR4n寄存器的OVF4n位为0。

注 仅限 SIR41 寄存器。

注意事项 必须将位 15 至 8、6 至 3(或者 SIR40 寄存器的位 15 至 1)清除为 0。

- 备注 1. n: 通道编号 (n = 0, 1)
2. SIR4n 寄存器的读取值总为 0000H。

(8) 串行通道开始寄存器 4 (SS4)

SS4 寄存器是对于各个通道进行设置以允许开始通信/计数的触发寄存器。

将 1 写入该寄存器(SS4n)的位时, 串行通道允许状态寄存器 4 (SE4)的对应位(SE4n)将被设置为 1 (允许操作)。由于 SS4n 位为触发位, 所以当 SE4n = 1 时, 该位将立即被清除。

使用 16 位存储器操作指令设置 SS4 寄存器。

通过 SS4L 使用 1 位或 8 位存储器操作指令设置 SS4 寄存器的低 8 位。

产生复位信号后, SS4 寄存器被清除为 0000H。

图 16-12. 串行通道开始寄存器 4 (SS4)的格式

地址: F05A2H, F05A3H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SS4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SS41	SS40

SS4n	通道n的操作开始触发
0	无触发操作。
1	将SE4n位设置为1, 进入通信等待状态 ^注 。

注 如果在通信操作中设置 SS4n = 1, 则会停止通信进入等待状态。
此时保持控制寄存器和移位寄存器、 $\overline{\text{SCK4n}}$ 和 SO4n 引脚、以及 FEF4n、PEF4n、OVF4n 标志的状态值。

注意事项 1. 必须将位 15 至 2 清除为 0。

2. 对于 UART 接收, 将 SCR4n 寄存器的 RXE4n 位设置为 1, 然后确保在经过 4 个或更多 f_{MCK} 时钟之后将 SS4n 设置为 1。

备注 1. n: 通道编号 (n = 0, 1)

2. SS4 寄存器的读取值总为 0000H。

(9) 串行通道停止寄存器 4 (ST4)

ST4 寄存器是对于各个通道进行设置以允许停止通信/计数的触发寄存器。

将 1 写入该寄存器(ST4n)的位时，串行通道允许状态寄存器 4 (SE4)的对应位(SE4n)将被清除为 0（停止操作）。由于 ST4n 位为触发位，所以当 SE4n = 0 时，该位将立即被清除。

用 16 位存储器操作指令写入 ST4 寄存器。

通过 ST4L 使用 1 位或 8 位存储器操作指令设置 ST4 寄存器的低 8 位。

产生复位信号后，ST4 寄存器被清除为 0000H。

图 16-13. 串行通道停止寄存器 4 (ST4)的格式

地址: F05A4H, F05A5H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ST4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	ST41	ST40

ST4n	通道n的操作停止触发
0	无触发操作。
1	将SE4n位清除为0，停止通信操作 ^注 。

注 保持控制寄存器和移位寄存器，SCK4n 和 SO4n 引脚，以及 FEF4n、PEF4n、OVF4n 标志的状态值。

注意事项 必须将位 15 至 2 清除为 0。

- 备注 1. n: 通道编号 (n = 0, 1)
2. ST4 寄存器的读取值总为 0000H。

(10) 串行通道允许状态寄存器 4 (SE4)

SE4 寄存器用于确认各个通道的串行发送/接收操作处于允许或停止状态。

将 1 写入串行通道开始寄存器 4 (SS4)的位时，该寄存器的对应位将被设置为 1。将 1 写入串行通道停止寄存器 4 (ST4)的位时，对应位将被清除为 0。

使用 16 位存储器操作指令读取 SE4 寄存器。

通过 SE4L 使用 1 位或 8 位存储器操作指令设置 SE4 寄存器的低 8 位。

产生复位信号后，SE4 寄存器被清除为 0000H。

图 16-14. 串行通道允许状态寄存器 4 (SE4)的格式

地址: F05A0H, F05A1H 复位后: 0000H R

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SE4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SE41	SE40

SE4n	表示通道n的操作允许或停止状态
0	停止操作。
1	允许操作。

备注 n: 通道编号 (n = 0, 1)

(11) 串行输出允许寄存器 4 (SOE4)

SOE4 寄存器用于设置是否允许或停止通道 0 的串行通信操作的输出。

对于允许串行输出的通道 0，将不能通过软件改写串行输出寄存器 4 (SO4)（描述见后文）的 SO40 位的值，同时从串行数据输出引脚输出一个由通信操作所反映的值。

对于停止串行输出的通道 0，可以通过软件设置 SO4 寄存器的 SO40 位的值，并从串行数据输出引脚输出该值。

使用 16 位存储器操作指令设置 SOE4 寄存器。

通过 SOE4L 使用 1 位或 8 位存储器操作指令设置 SOE4 寄存器的低 8 位。

产生复位信号后，SOE4 寄存器被清除为 0000H。

图 16-15. 串行输出允许寄存器 4 (SOE4)的格式

地址: F05AAH, F05ABH 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOE4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SOE 40

SOE 40	通道0的串行输出允许/停止
0	通过串行通信操作来停止输出。
1	通过串行通信操作来允许输出。

注意事项 必须将位 15 至 1 清除为 0。

(12) 串行输出寄存器 4 (SO4)

SO4 寄存器是用于通道 0 串行输出的缓冲器寄存器。

该寄存器的 SO40 位的值从通道 0 的串行数据输出引脚输出。

该寄存器的 SO40 位只有在串行输出被禁止时(SOE40 = 0)才可用软件改写。当允许串行输出(SOE40 = 1)时，用软件改写将被忽略，且该寄存器的值只能通过串行通信操作更改。

要将串行接口 4 的引脚用作端口功能引脚，须将对应的 SO40 位设置为 1。

使用 16 位存储器操作指令设置 SO4 寄存器。

产生复位信号后，SO4 寄存器被清除为 0F0FH。

图 16-16. 串行输出寄存器 4 (SO4)的格式

地址: F05A8H, F05A9H 复位后: 0F0FH R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SO4	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	SO 40

SO40	通道0的串行数据输出
0	串行数据输出值为0。
1	串行数据输出值为1。

注意事项 必须将位 11 至 8、3 至 1 设置为 1。必须将位 15 至 12、7 至 4 清除为 0。

(13) 串行输出电平寄存器 4 (SOL4)

SOL4 寄存器是用于设置通道 0 的数据输出电平反相的寄存器。

仅当允许串行输出时 (SOE40 = 1)，通过该寄存器对通道 n 所作的反相设置才会体现在引脚输出上。当串行输出被禁止(SOE40 = 0)时，SO40 位的值按原样输出。

当寄存器在运行 (当 SE40 = 1) 时，禁止改写 SOL4 寄存器。

使用 16 位存储器操作指令设置 SOL4 寄存器。

通过 SOL4L 使用 8 位存储器操作指令设置 SOL4 寄存器的低 8 位。

产生复位信号后，SOL4 寄存器被清除为 0000H。

图 16-17. 串行输出电平寄存器 4 (SOL4)的格式

地址: F05B4H, F05B5H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
SOL4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SOL 40

SOL 40	通道0的发送数据电平反相的选择
0	通信数据按原样输出。
1	通信数据反相并输出。

注意事项 必须将位 15 至 1 清除为 0。

(14) 串行待机控制寄存器 4 (SSC4)

SSC4 寄存器可通过接收 DALI/UART4 的串行数据，来控制 **STOP** 模式下开始接收（**SNOOZE** 模式）。

使用 16 位存储器操作指令设置 SSC4 寄存器。

通过 SSC4L 使用 8 位存储器操作指令设置 SSC4 寄存器的低 8 位。

产生复位信号后，SSC4 寄存器被清除为 0000H。

注意事项 1. 仅限选择高速片上振荡器时钟作为 f_{CLK} 时可以设置 **SNOOZE** 模式。并且，**SNOOZE** 模式时不能执行 **PLL** 输出。

从 **SNOOZE** 模式返回正常操作模式后，可以执行 **PLL** 输出。这种情况下，时钟可能产生“+1.125 sec (max.)至-0.406 sec”范围内的误差(使用 **PLL** 时 f_{CLK} = 32 MHz ↔ 4 MHz)。

DALI 通信时使用曼彻斯特协议，由于传送率为 1200 bps，该误差对其不造成影响。

2. 在 **SNOOZE** 模式下使用 **UART** 时，最大传送率为 9600 bps。

图 16-18. 串行待机控制寄存器 4 (SSC4)的格式

地址: F05B8H, F05B9H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSC4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SSEC 4	SWC4

SSEC 4	选择是否允许或停止产生传送结束中断
0	允许产生错误中断(INTSREDL4)。 以下情况时，时钟发生器的时钟请求信号（内部信号）也将被清除： <ul style="list-style-type: none"> • SWC4位被清除为0 • 当误测到DALI/UART4接收开始位时
1	停止产生错误中断(INTSREDL4)。 以下情况时，时钟发生器的时钟请求信号（内部信号）也将被清除： <ul style="list-style-type: none"> • SWC4位被清除为0 • 当误测到DALI/UART4接收开始位时 • 由于奇偶检验错误或帧错误而产生传送结束中断的时序

SWC 4	选择允许或停止在STOP模式下开始DALI/UART4的接收操作
0	不使用SNOOZE模式功能。
1	使用SNOOZE模式功能。 <ul style="list-style-type: none"> • 由于STOP模式下的硬件触发信号，退出STOP模式，并且在不操作CPU的情况下执行A/D转换（SNOOZE模式）。 • 仅在选择高速片上振荡器时钟作为CPU/外围硬件时钟(f_{CLK})时可以指定SNOOZE模式功能。如果选用任何其他时钟，则禁止指定该模式。 • 即便使用 SNOOZE模式，请在正常操作模式中设置 SWC4位为0，并且在切换至STOP模式之前将AWC位更改为1。 此外，从STOP模式恢复到正常操作模式前，请更改SWC4位为0。

(15) 串行选项控制寄存器 4 (SOC4)

SOC4 寄存器用于控制 DALI/UART4 通信模式。

使用 16 位存储器操作指令设置 SOC4 寄存器。

产生复位信号后，SOC4 寄存器被清除为 0000H。

图 16-19. 串行选项控制寄存器 4 (SOC4)的格式

地址: F05BAH, F05BBH 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
SOC4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SDS

SDS	DALI/UART模式的选择
0	UART模式
1	DALI模式

(16) 单线 UART 控制寄存器(SUCTL)

该寄存器用于设置 DALI/UART4 通信方式。

通过将 SUCTL 位设置为 1，可以执行单线 DALI/UART4 通信。

选择单线模式时，TxRx4/TKCO05/DALIRxD4/INTP23/P206 引脚兼用为发送和接收。

使用 1 位或 8 位存储器操作指令设置 SUCTL 寄存器。

产生复位信号后，SUCTL 寄存器被清除为 00H。

图 16-20. 单线 UART 控制寄存器(SUCTL)的格式

地址: F05C7H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
SUCTL	0	0	0	0	0	0	0	SUCTL0

SUCTL0	DALI/UART4通信方式的切换
0	双线数据通信 (使用DALIRxD4和DALITxD4引脚)
1	单线数据通信 (使用TxRx4引脚)

(17) 噪声滤波器允许寄存器 3 (NFEN3)

NFEN3 寄存器是按照每个通道设置对于串行数据输入引脚的输入信号是否使用噪声滤波器。
 允许噪声滤波器时，对 CPU/外围硬件时钟(fCLK)执行双时钟匹配检测以及同步。
 使用 1 位或 8 位存储器操作指令设置 NFEN3 寄存器。
 产生复位信号后，NFEN3 寄存器被清除为 00H。

图 16-21. 噪声滤波器允许寄存器 3 (NFEN3)的格式

地址: F05C1H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
NFEN3	0	0	0	0	0	0	0	SNFEN30

SNFEN30	RxD4引脚滤波器的使用(DALIRxD4/TxRx4/TKCO05/INTP23/P206)
0	噪声滤波器OFF
1	噪声滤波器ON

作为DALIRxD4引脚使用时，将SNFEN30设为1。
 作为DALIRxD4引脚以外使用时，将SNFEN30清除为0。

注意事项 必须将位 7 至 1 清除为 0。

(18) 端口输出模式寄存器 1, 20 (POM1, POM20)

该寄存器按照 1 位单位设置端口 1 和 20 的输出模式。
 用 1 位或 8 位存储器操作指令来设置 POM1 和 POM20 寄存器。
 另外，通过设置 POM1 和 POM20 寄存器以及 PUxx 寄存器来选择是否使用片上上拉电阻。
 产生复位信号后，POM1 和 POM20 寄存器被清除为 00H。

图 16-22. 端口输出模式寄存器 1, 20 (POM1, POM20)的格式

地址: F0051H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
POM1	0	0	0	0	0	POM12	POM11	POM10

地址: F0530H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
POM20	0	POM206	POM205	POM204	POM203	POM202	POM201	POM200

POMmn	Pmn引脚的输出缓冲器的选择(m = 1, 20; n = 0至6)
0	正常输出模式 输入时，允许PUmn位。
1	N沟开漏输出 (VDD耐压/EVDD耐压)模式 输入时，禁用PUmn位。

(19) 端口模式寄存器 1, 20 (PM1, PM20)

该寄存器按照 1 位单位设置端口 1、20 的输入/输出。

将兼用为串行数据输出引脚的端口(PM10/SO00/TxD0/TKCO00/INTP20/SCLA0/(DALITxD4)、

P205/DALITxD4/TKBO21/TKCO04)用作串行数据输出时, 设置端口模式寄存器(PM1)的 PM10 位或者端口模式寄存器(PM20)的 PM 205 位为 1, 此时端口寄存器(P1)的 P10 位或者端口寄存器(P20)的 P206 位可以为 0 或 1。

将兼用为串行数据输入引脚的端口 (P11/SI00/RxD0/TKCO01/INTP21/SDAA0/(TI07)/(DALIRxD4)/(TxRx4)、P206/DALIRxD4/TxRx4/TKCO05/INTP23)用作串行数据输入时, 设置端口模式寄存器(PM1)的 PM11 位或者端口模式寄存器(PM20)的 PM 206 位为 1, 此时端口寄存器(P1)的 P11 位或者端口寄存器(P20)的 P206 位可以为 0 或 1。

使用 1 位或 8 位存储器操作指令来设置 PM1、PM20 寄存器。

产生复位信号后, PM1 和 PM20 寄存器被设置为 FFH。

图 16-23. 端口模式寄存器 1, 20 (PM1, PM20)的格式(38 引脚产品)

地址: FFF21H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM1	1	1	1	1	1	PM12	PM11	PM10

地址: F0510H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM20	1	PM206	PM205	PM204	PM203	PM202	PM201	PM200

PMmn	Pmn引脚的输入/输出模式选择(m = 1, 20; n = 0至6)
0	输出模式 (输出缓冲器on)
1	输入模式 (输出缓冲器off)

注意事项 必须将 PM1 寄存器的位 3 至 7 以及 PM20 寄存器的位 7 设置为 1。

对于 32, 30 和 20 引脚产品, 在解除复位后, 必须通过软件将以下的位设置为输出模式 (通过设置端口寄存器和端口模式寄存器为 0 来指定)。

32 引脚产品: PM1 寄存器的位 2

30 引脚产品: PM1 寄存器的位 2

20 引脚产品: PM1 寄存器的位 2 和 PM20 寄存器的位 4 至 6

(20) 外围输入/输出重定向寄存器(PIOR1)

该寄存器用于指定是否允许或禁止外围输入/输出重定向功能。

该功能用于切换被分配了复用功能的端口。

使用 8 位存储器操作指令来设置 PIOR1 寄存器。

产生复位信号后，该寄存器被设置为 00H。

图 16-24. 外围输入/输出重定向寄存器(PIOR1)的格式

地址: F05C0H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
PIOR1	0	0	0	0	PIOR13	PIOR12	PIOR11	PIOR10

位	功能	20引脚		30引脚		32引脚		38引脚	
		设置值		设置值		设置值		设置值	
		0	1	0	1	0	1	0	1
PIOR11	DALITxD4/ DALIRxD4	–	P10/P11	P205/ P206	P10/P11	P205/ P206	P10/P11	P205/ P206	P10/P11

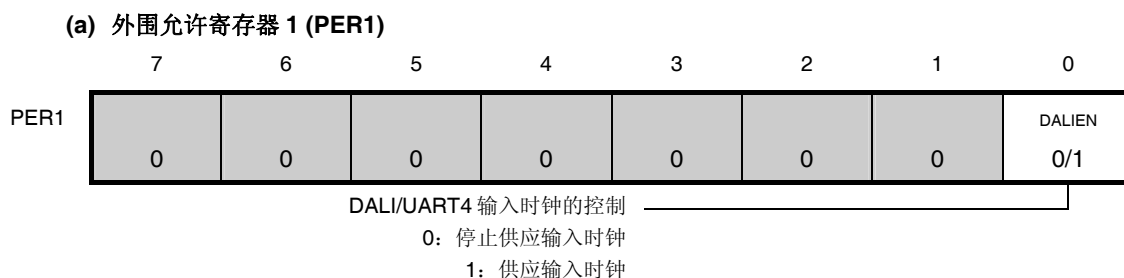
16.4 操作停止模式

串行阵列单元 4 (DALI/UART4)具有操作停止模式。
在此模式下，不能执行串行通信，因而可降低功耗。
另外，在该模式下，用于串行接口 4 的引脚可用作端口功能引脚。

16.4.1 以单元单位停止操作

使用外围允许寄存器 1 (PER1)设置按照单元单位停止操作。
PER1 寄存器用于允许或禁止向外围硬件供应时钟。并且中止向未使用的硬件供应时钟，以减少电力消耗和噪声。
将位 0 (DALIEN)设置为 0 以停止操作串行阵列单元 4。

图 16-25. 以单元单位停止操作时外围允许寄存器 1 (PER1)的设置



注意事项 1. 如果 DALIEN = 0，对于串行阵列单元 4 (DALI/UART4)的控制寄存器进行的写入操作被忽略，而且该寄存器的读取值总为初始值。

但如下的寄存器除外：

- 噪声滤波器允许寄存器 3 (NFEN3)
- 串行待机控制寄存器 4 (SSC4)
- 端口输出模式寄存器 20 (POM20)
- 端口模式寄存器 20 (PM20)
- 端口寄存器 20 (P20)

2. 必须将位 1 至 7 清除为 0。

备注 ：禁止设置(固定为初始值)

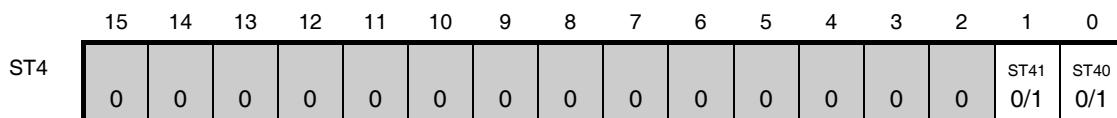
0/1：可根据用户的用途设为 0 或 1

16.4.2 以通道单位停止操作时

使用以下各种寄存器的设置，从而以通道单位停止操作。

图 16-26. 以通道单位停止操作时每个寄存器的设置

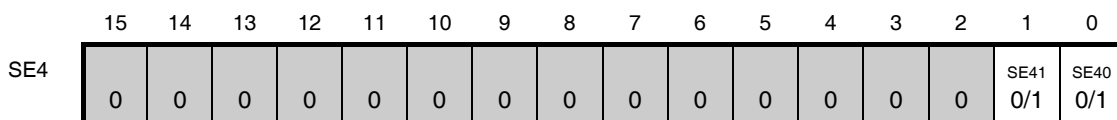
(a) 串行通道停止寄存器 4 (ST4) ... 该寄存器是用于设置是否允许各个通道的通信/计数的触发寄存器。



1: 将 SE4n 位清除为 0 并且停止通信操作

* 由于 ST4n 位为触发位，所以当 SE4n = 0 时，该位将立即被清除。

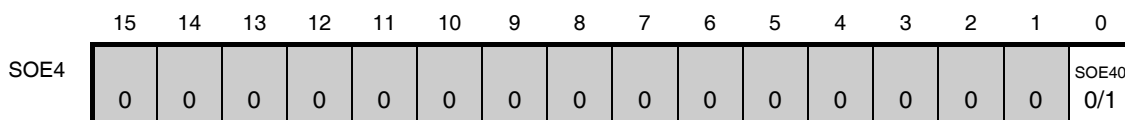
(b) 串行通道允许状态寄存器 4 (SE4) ... 该寄存器指示是否允许或停止各个通道的数据发送/接收操作。



0: 停止操作

* SE4 寄存器是 read-only 状态寄存器，通过 ST4 寄存器停止操作该寄存器。

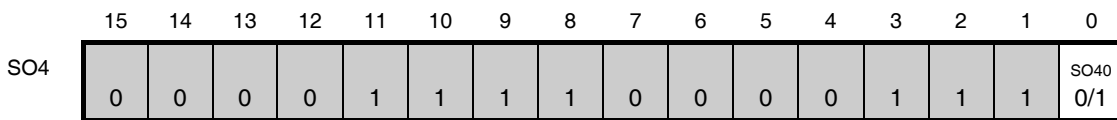
(c) 串行输出允许寄存器 4 (SOE4) ... 该寄存器用于设置允许或停止各个通道的串行通信操作的输出。



0: 利用串行通信操作来停止输出

* 对于通道 n (其串行输出已停止)，可以用软件设置 SO4 寄存器的 SO4n 位的值。

(d) 串行输出寄存器 4 (SO4) ... 该寄存器是各个通道的串行输出的缓冲寄存器。



1: 串行数据输出值为 1

* 在将应用于各个通道的引脚用作端口功能引脚时，将相应的 SO4n 位设置为 1。

备注 1. n: 通道编号 (n = 0, 1)

2. : 禁止设置(固定为初始值), 0/1: 可根据用户的用途设为 0 或 1

16.5 UART通信的操作

16.5.1 UART发送

UART 发送操作用于将数据从 RL78/I1A 异步发送至另一器件（调步同期）。
在用于 UART 的两个通道中，偶数通道用于 UART 发送。

UART发送	
目标通道	通道0
使用引脚	DALITxD4
传送数据长度	7, 8或9位
传送速率	Max. $f_{MCK}/6$ [bps] (SDR4n [15:9] = 3或以上), Min. $f_{CLK}/(2 \times 2^{11} \times 128)$ [bps] [※]
数据方向	MSB或LSB优先
数据相位	正相输出 (默认: 高电平) 反相输出 (默认: 低电平)
奇偶校验位	以下可选 <ul style="list-style-type: none"> • 无奇偶校验位 • 附加零校验 • 附加偶校验 • 附加奇校验
停止位	以下可选 <ul style="list-style-type: none"> • 附加1位 • 附加2位
错误检测标志	无
中断	INTSTDL4
	可选择发送结束中断(单发送模式时)或缓冲器空中断(连续发送模式时)

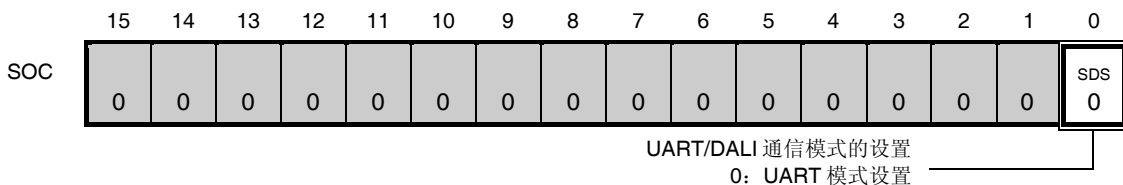
注 在满足以上条件及电特性中的 AC 特性(参阅第三十二章 电特性)的范围内使用此操作。

- 备注 1.** f_{MCK} : 目标通道的操作时钟频率
 f_{CLK} : 系统时钟频率
- 2.** n: 通道编号 (n = 0)

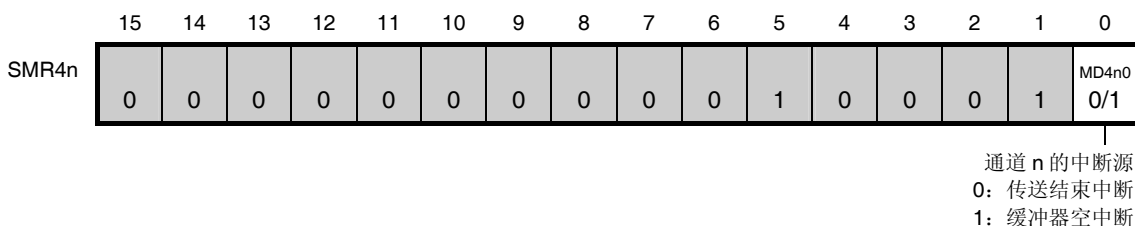
(1) 寄存器设置

图 16-27. UART 发送时的寄存器设置内容示例 (1/2)

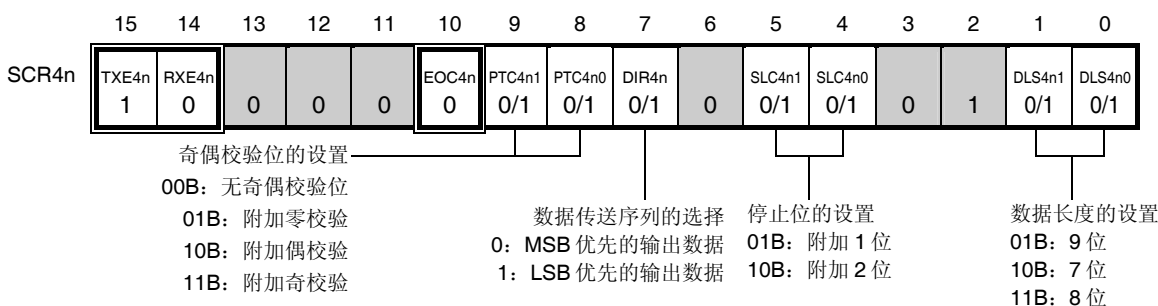
(a) 串行选项控制寄存器 4 (SOC4)



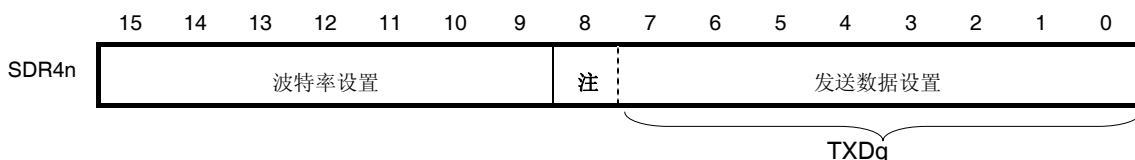
(b) 串行模式寄存器 4n (SMR4n)



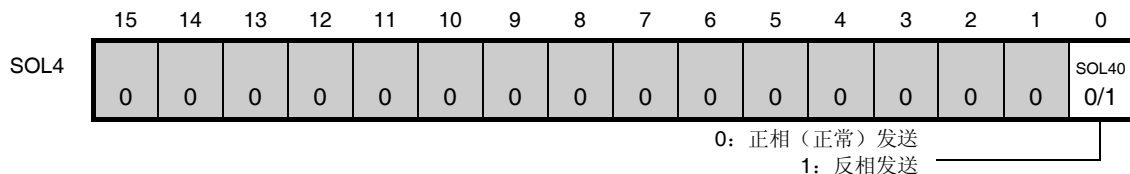
(c) 串行通信操作设置寄存器 4n (SCR4n)



(d) 串行数据寄存器 4n (SDR4n) (低 8 位: TXDq)



(e) 串行输出电平寄存器 4 (SOL4) ...仅设置目标通道的位。

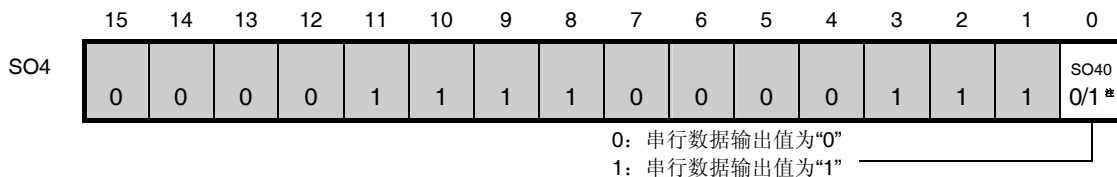


注 当执行 9 位数据长度通信(DLS401, DLS400 = 0, 1)时, SDR40 寄存器的位 0 至位 8 用作发送数据指定区。

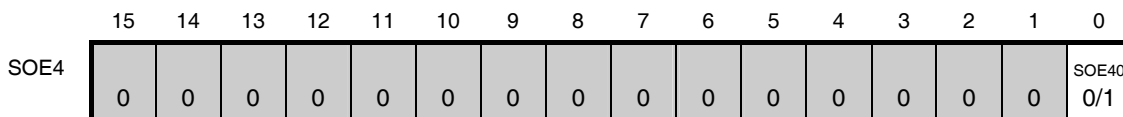
- 备注 1. n: 通道编号(n = 0), q: UART 编号(q = 4),
4n = 00, 02, 10
2. : 设置固定于 CSI 主发送模式, : 禁止设置 (设为初始值)
x: 此模式下不能使用的位 (任意模式下均不使用设设为初始值)
0/1: 可根据客户的用途设为 0 或 1

图 16-27. UART 发送时的寄存器设置内容示例 (2/2)

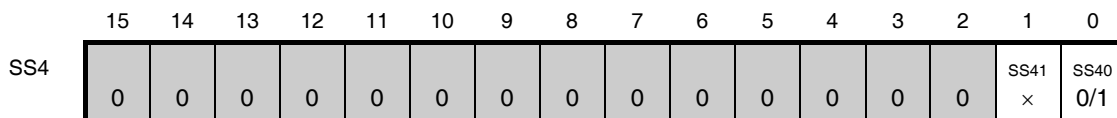
(f) 串行输出寄存器 4 (SO4) ...仅设置目标通道的位。



(g) 串行输出允许寄存器 4 (SOE4) ...仅将目标通道的位设置为 1。



(h) 串行通道开始寄存器 4 (SS4) ...仅将目标通道的位设置为 1。



注 开始发送前，当目标通道的 SOL4n 位被设置为 0 时，必须设置为 1，当目标通道的 SOL4n 位被设置为 1 时，则设置为 0。在通信操作过程中，该值因通信数据而异。

- 备注 1. n: 通道编号 (n = 0)
 2. : 禁止设置(设置为初始值)
 x: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)
 0/1: 可根据客户的用途设为 0 或 1

(2) 操作步骤

图 16-28. UART 发送的初始设定步骤

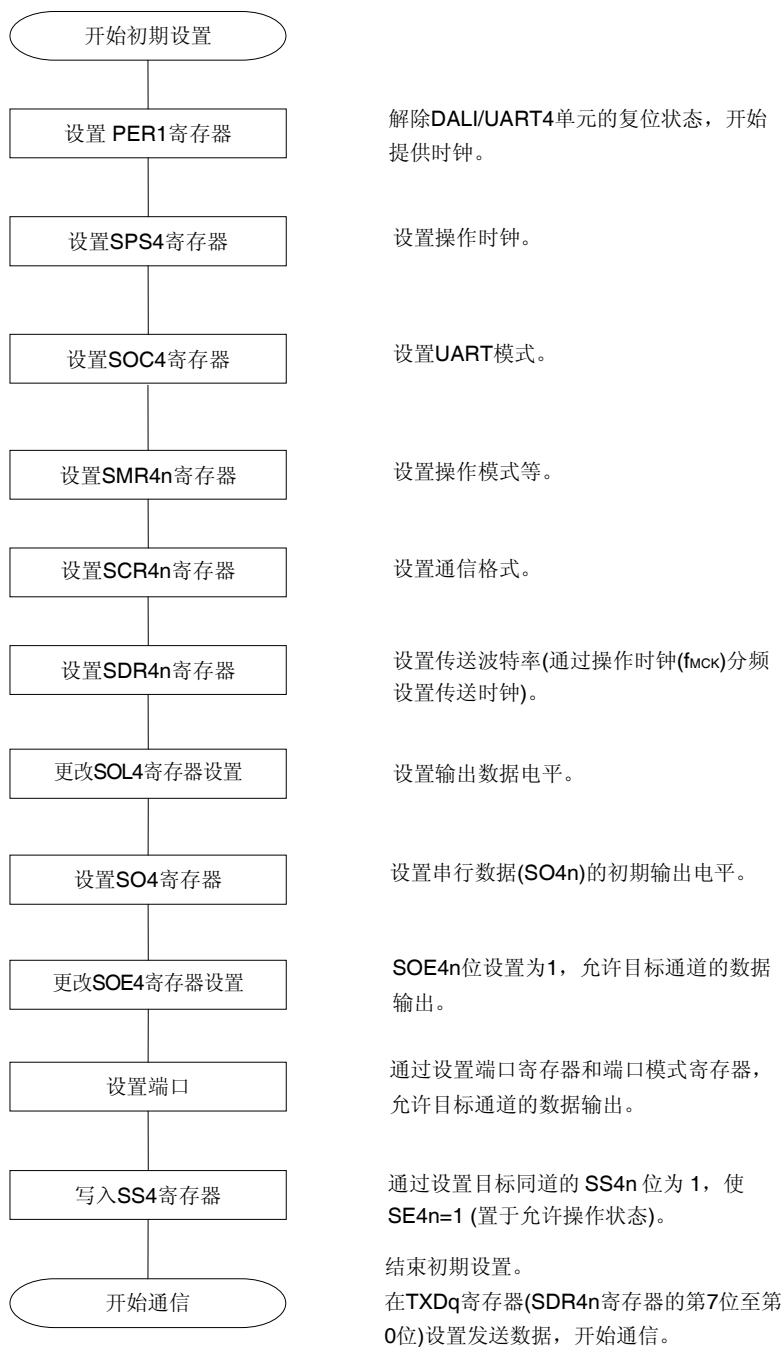


图 16-29. 停止 UART 发送的步骤

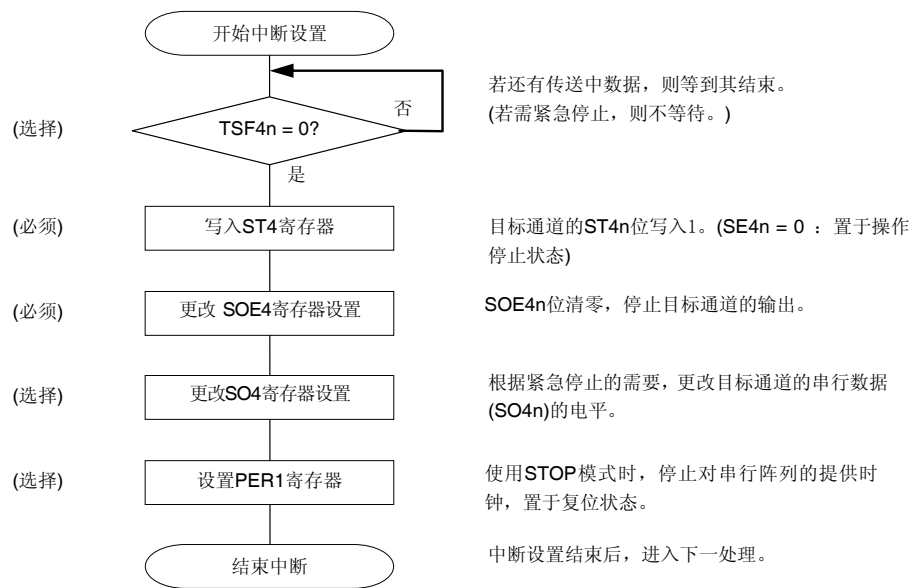
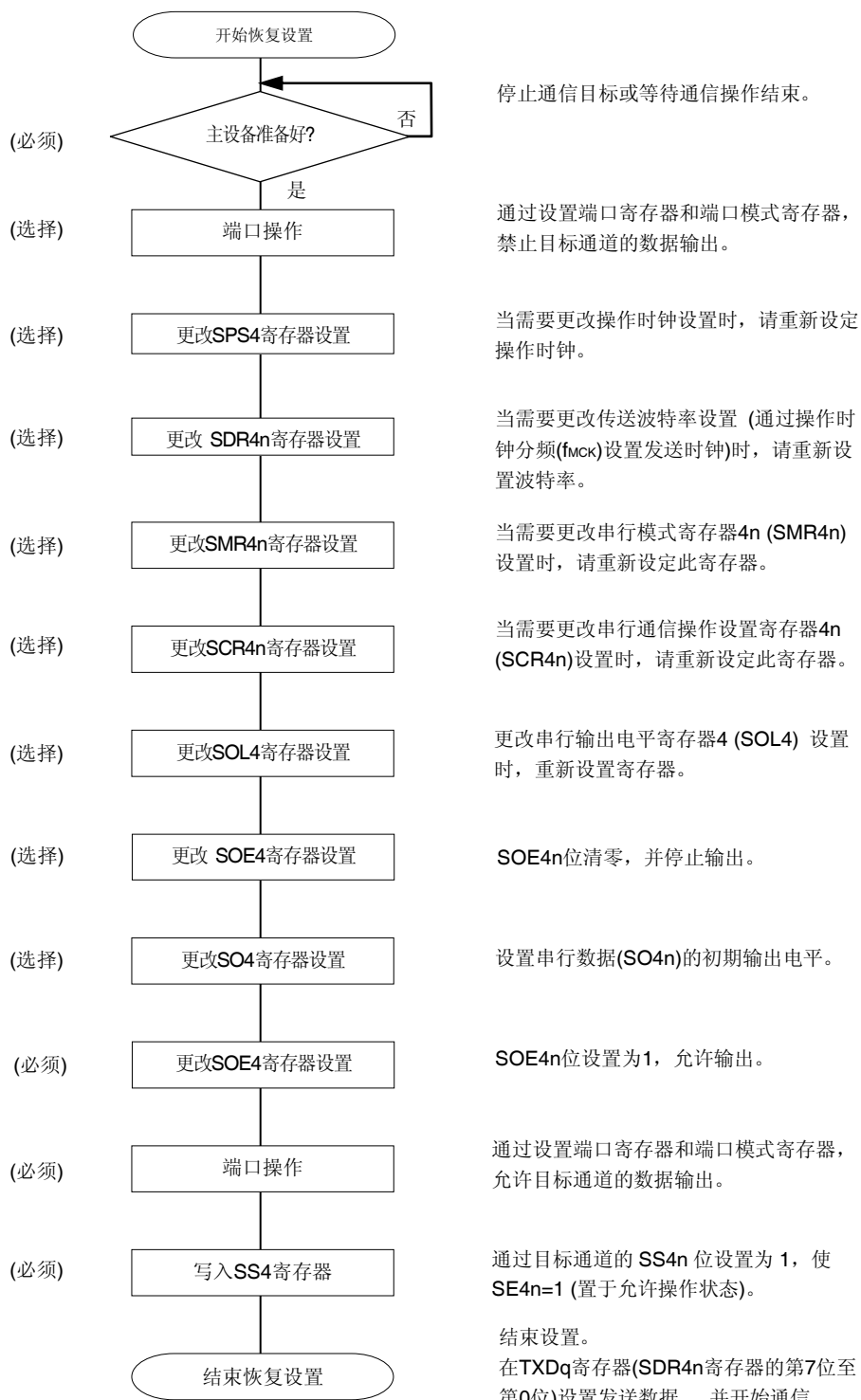
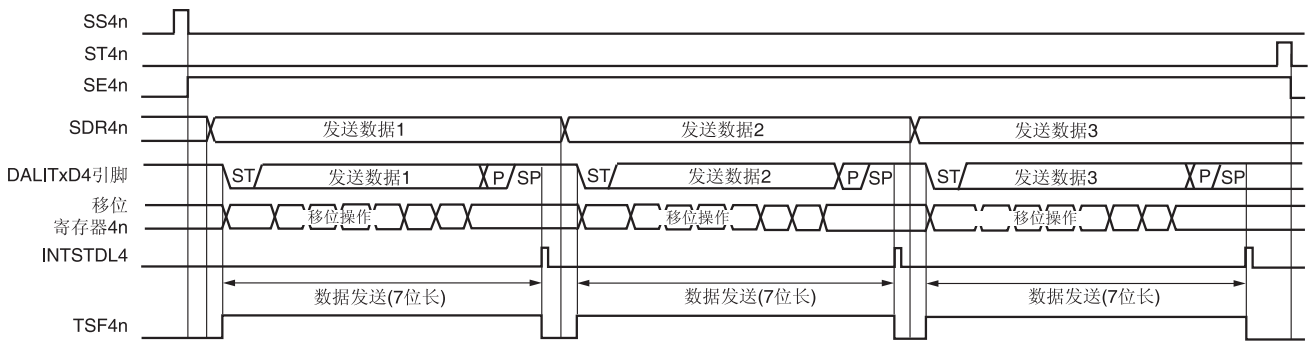


图 16-30. 恢复 UART 发送的步骤



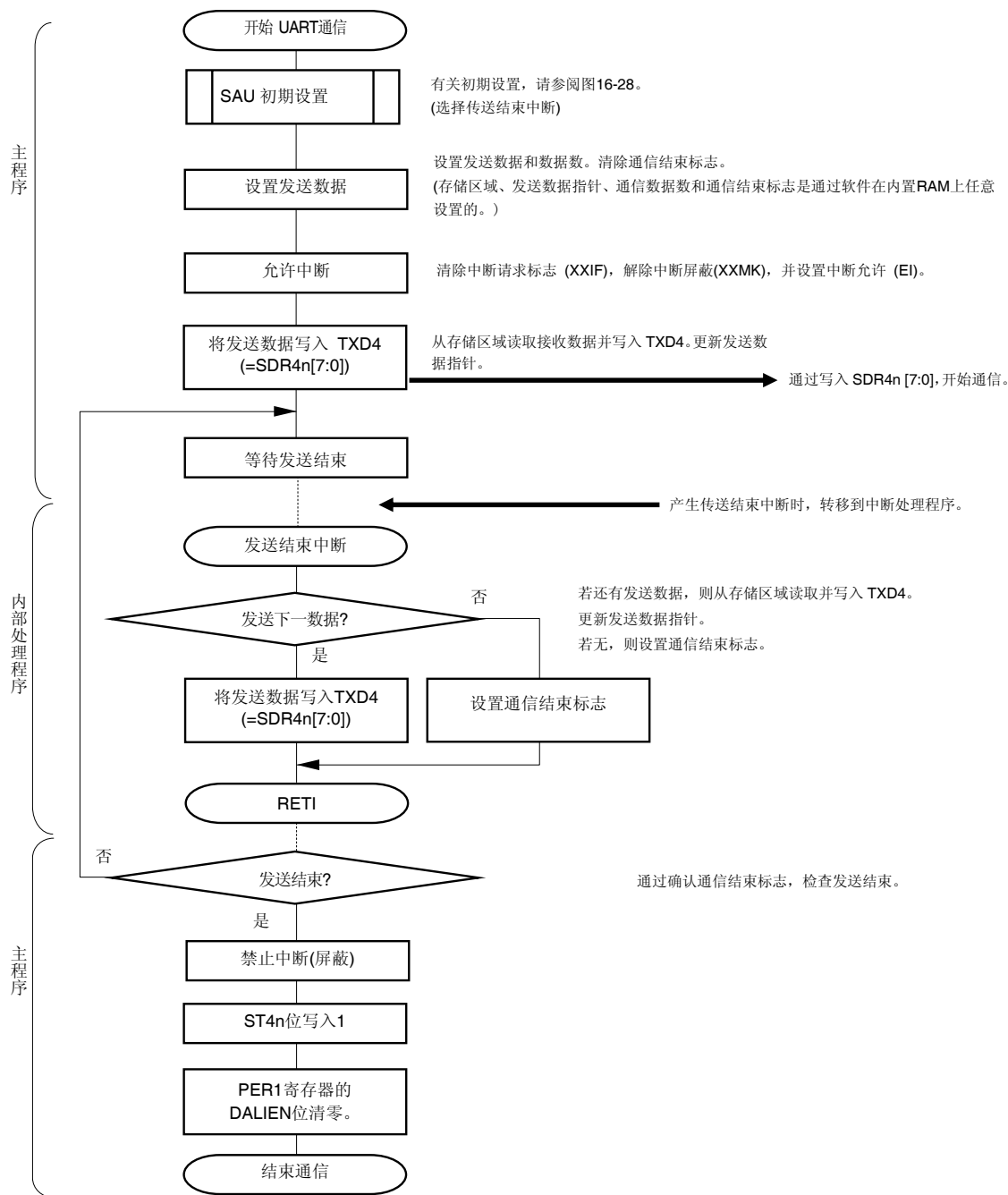
(3) 处理流程 (单发送模式时)

图 16-31. UART 发送(单发送模式时)的时序图



备注 n: 通道编号(n = 0)

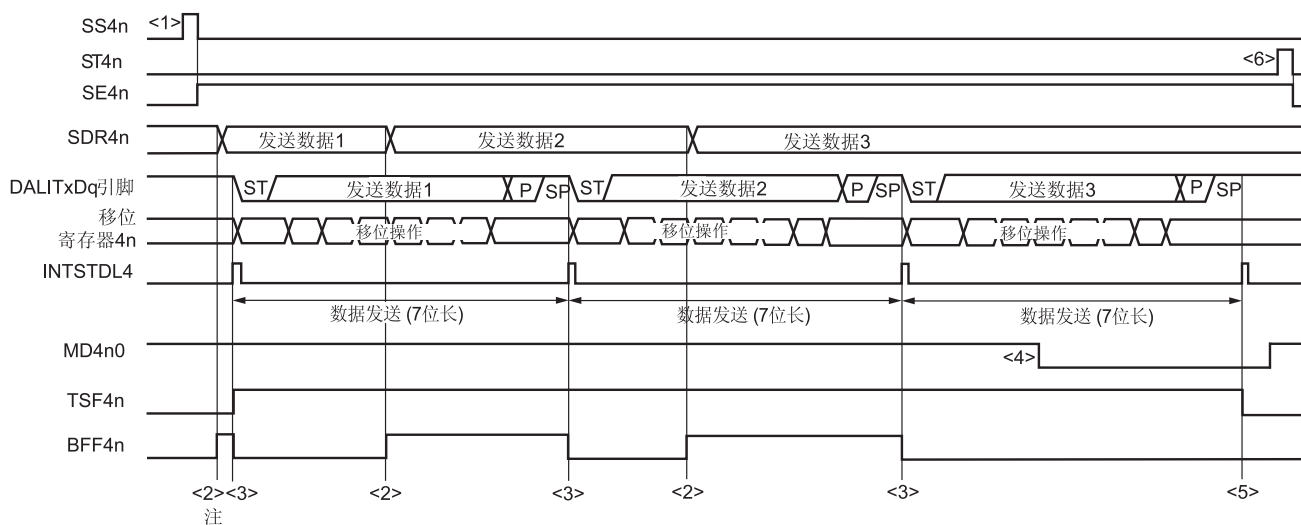
图 16-32. UART 发送(单发送模式时)的流程图



注意事项 在设置外围允许寄存器 1 (PER1)的 DALIEN 位为 1 后, 必须在经过 4 个或更多 fCLK 时钟后设置串行时钟选择寄存器 4 (SPS4)。

(4) 处理流程 (连续发送模式时)

图 16-33. UART 发送(连续发送模式时)的时序图

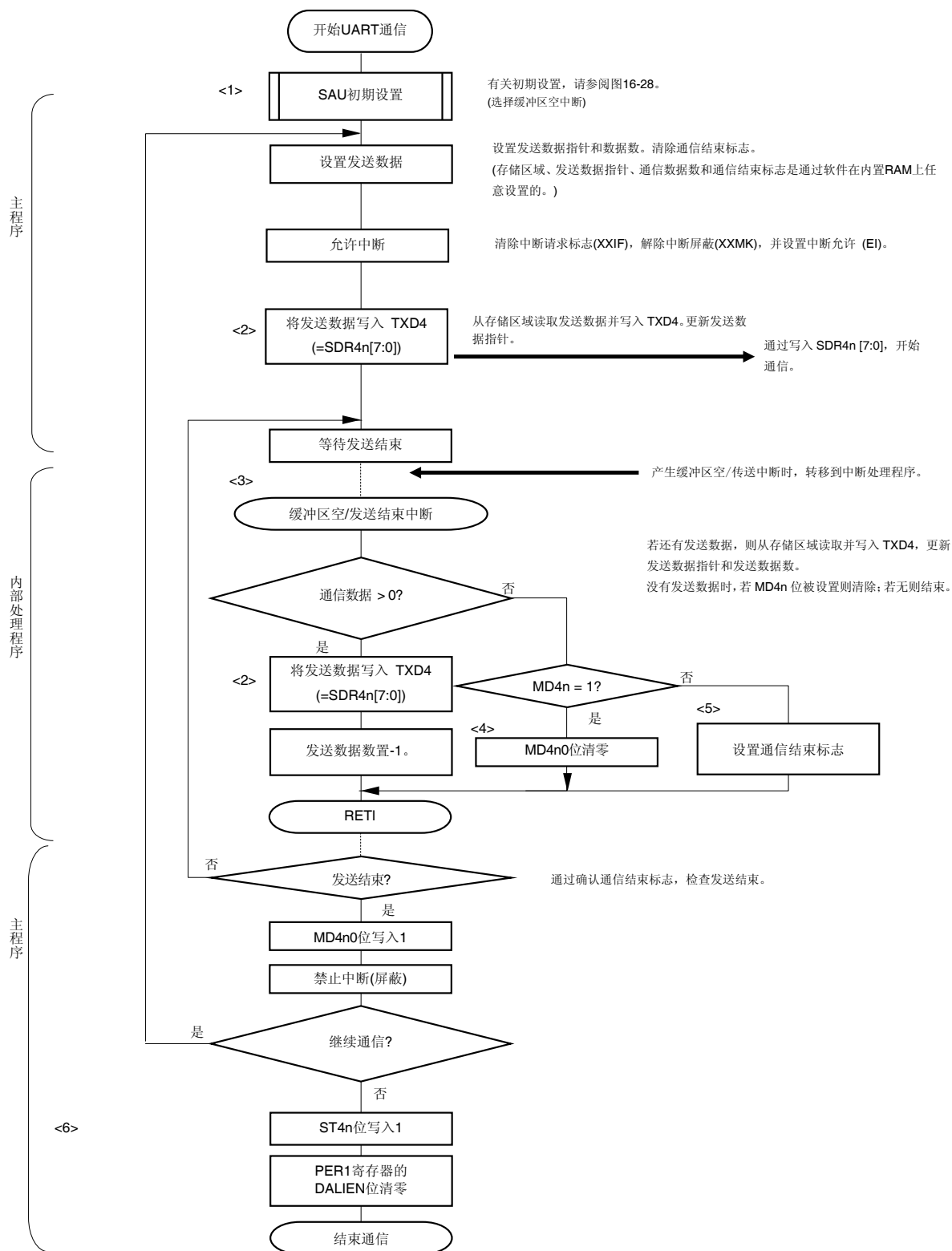


注 如果在串行状态寄存器 4n (SSR4n)的 BFF4n 位为 1 时将发送数据写入 SDR4n 寄存器 (将有效数据存储于串行数据寄存器 4n (SDR4n)中)，则发送数据将被覆盖。

注意事项 即使在操作过程中，也可改写串行模式寄存器 4n (SSR4n)的 MD4n0 位。
然而必须在开始传送最后一位之前改写，以便在最后发送数据的传送结束中断前完成改写。

备注 n: 通道编号(n = 0)

图 16-34. UART 发送(连续发送模式时)的流程图



注意事项 在设置外围允许寄存器 1 (PER1)的 DALIEN 位为 1 后, 必须在经过 4 个或更多 fCLK 时钟后设置串行时钟选择寄存器 4 (SPS4)。

备注 此图中的<1>至<6>对应于图 16-33. UART 发送(连续发送模式时)的时序图中的<1>至<6>。

16.5.2 UART接收

UART 接收操作指 RL78/I1A 从另一器件异步接收数据（开始-停止同步）。

用于 UART 的两个通道中的奇数通道被用作 UART 接收。必须设置奇数和偶数通道的 SMR 寄存器。

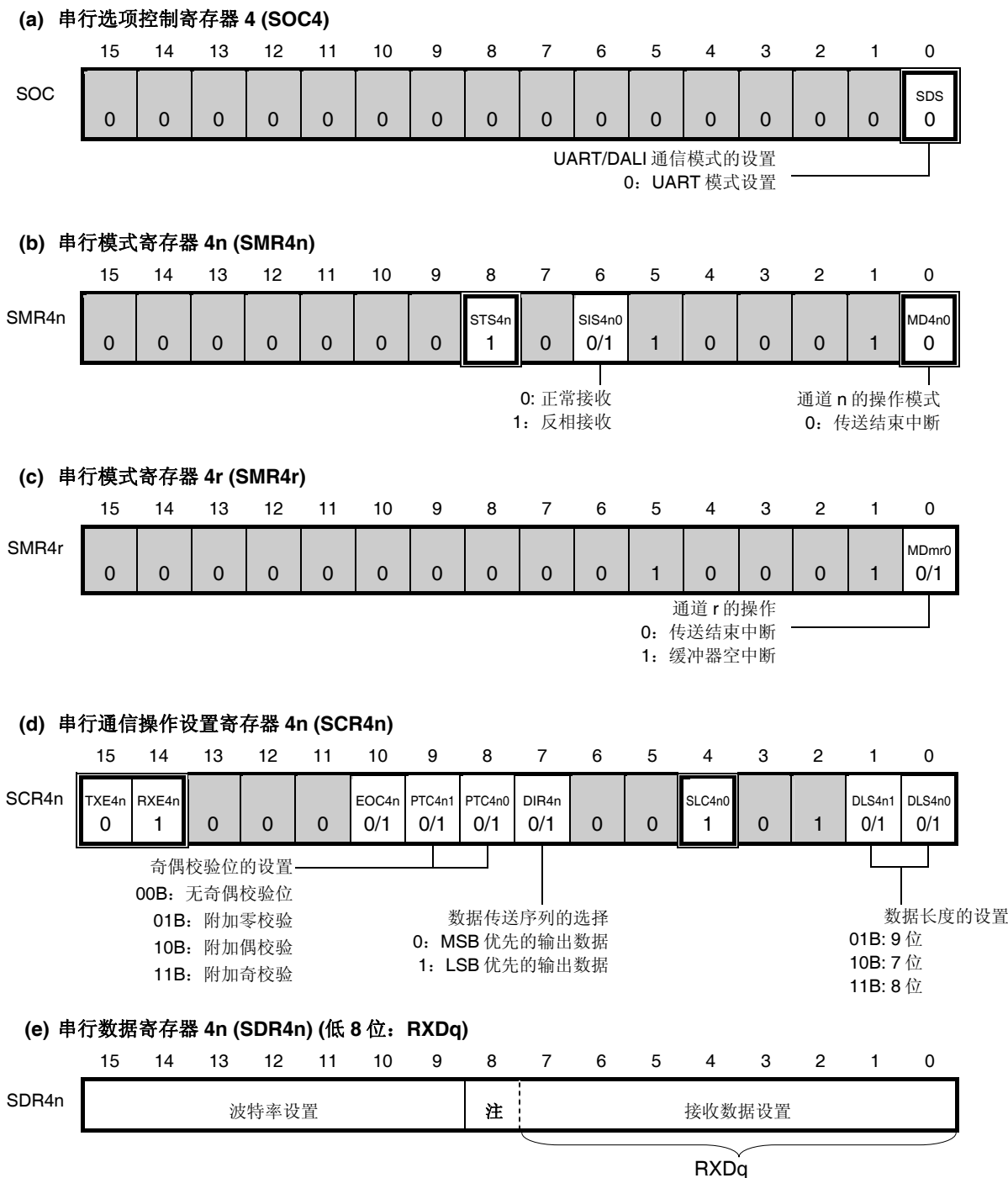
UART接收	
目标通道	通道1
使用引脚	DALIRxD4
传送数据长度	7, 8或9位
传送速率	Max. $f_{MCK}/6$ [bps] (SDR4n [15:9] = 3或以上), Min. $f_{CLK}/(2 \times 2^{11} \times 128)$ [bps] [※]
数据方向	MSB或LSB优先
数据相位	正相输出 (默认: 高电平) 反相输出 (默认: 低电平)
奇偶校验位	以下可选 <ul style="list-style-type: none"> • 无奇偶校验位 (无奇偶校验) • 附加零校验 (无奇偶校验) • 附加偶校验 • 附加奇校验
停止位	附加1位
错误检测标志	<ul style="list-style-type: none"> • 帧错误检测标志(FEF4n) • 奇偶校验错误检测标志(PEF4n) • 溢出错误检测标志(OVF4n)
中断	INTSRDL4
	仅限传送结束中断(禁止设置缓冲器空中断。)
错误中断	INTSREDL4

注 在满足以上条件及电特性中的 AC 特性(参阅第三十二章 电特性)的范围内使用此操作。

- 备注 1. f_{MCK} : 目标通道的操作时钟频率
 f_{CLK} : 系统时钟频率
2. n: 通道编号(n = 1)

(1) 寄存器设置

图 16-35. UART 接收时的寄存器设置内容示例 (1/2)



注 当执行 9 位数据长度通信(DLS411, DLS410 = 0, 1)时, SDR41 寄存器的位 0 至位 8 用作发送数据指定区。

注意事项 对于 UART 接收, 必须设置通道 r 的 SMR4r 寄存器, 使其与通道 n 配对。

- 备注 1. n: 通道编号(n = 1), r: 通道编号(r = 0), q: UART 编号(q = 4)
2. □: 设置固定于 CSI 主发送模式, ■: 禁止设置 (设为初始值)
x: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)
0/1: 可根据客户的用途设为 0 或 1

图 16-35. UART 接收时的寄存器设置内容示例 (2/2)

(f) 串行输出寄存器 4 (SO4) ...此模式下不使用的寄存器。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
SO4	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1		SO40 ×

(g) 串行输出允许寄存器 4 (SOE4) ...此模式下不使用的寄存器。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
SOE4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		SOE40 ×

(h) 串行通道开始寄存器 4 (SS4) ...仅将目标通道的位设置为 1。

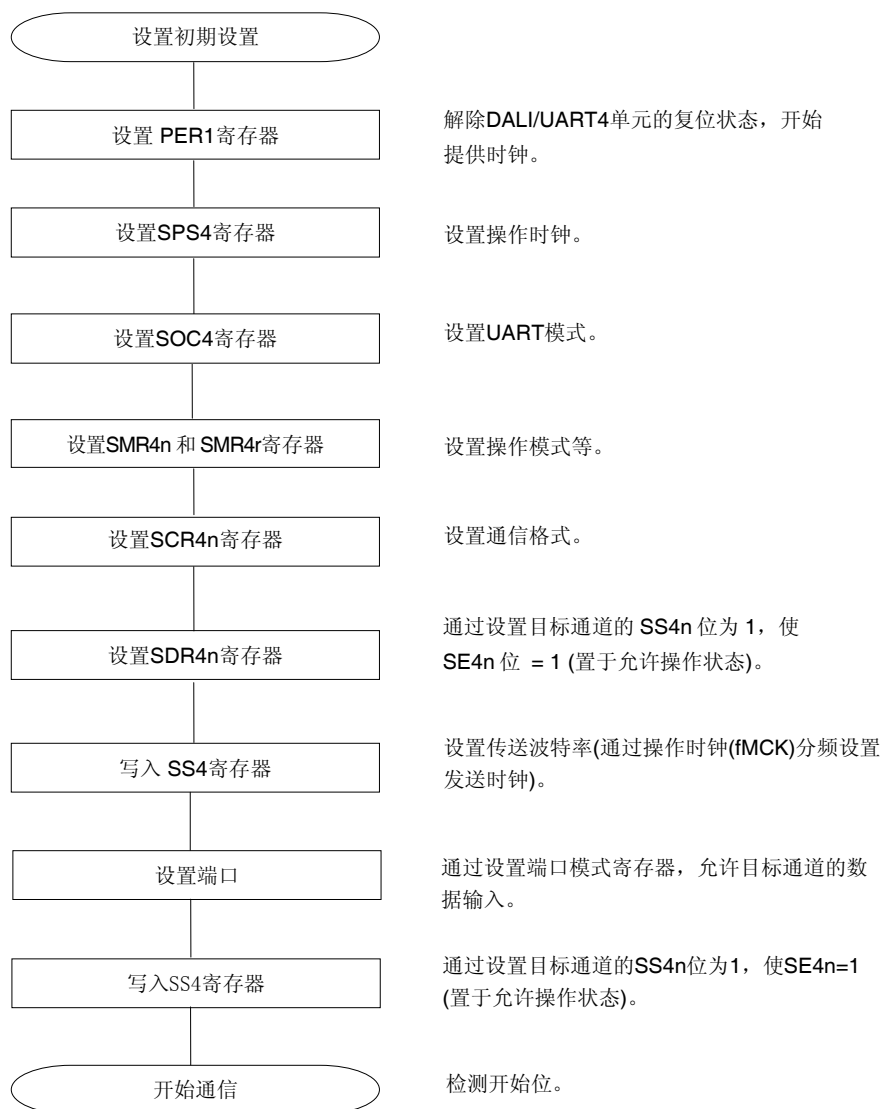
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
SS4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		SS41 0/1	SS40 ×

注意事项 对于 UART 接收，必须设置通道 r 的 SMR4r 寄存器，使其与通道 n 配对。

- 备注 1. n: 通道编号(n = 1), r: 通道编号(r = 0), q: UART 编号(q = 4)
2. : 设置固定于 CSI 主发送模式, : 禁止设置 (设为初始值)
 x: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)
 0/1: 可根据客户的用途设为 0 或 1

(2) 操作步骤

图 16-36. UART 接收的初始设定步骤



注意事项 将 SCR4n 寄存器的 RXE4n 位设置为 1，然后确保在经过 4 个或更多 fMCK 时钟之后将 SS4n 设置为 1。

图 16-37. 停止 UART 接收的步骤

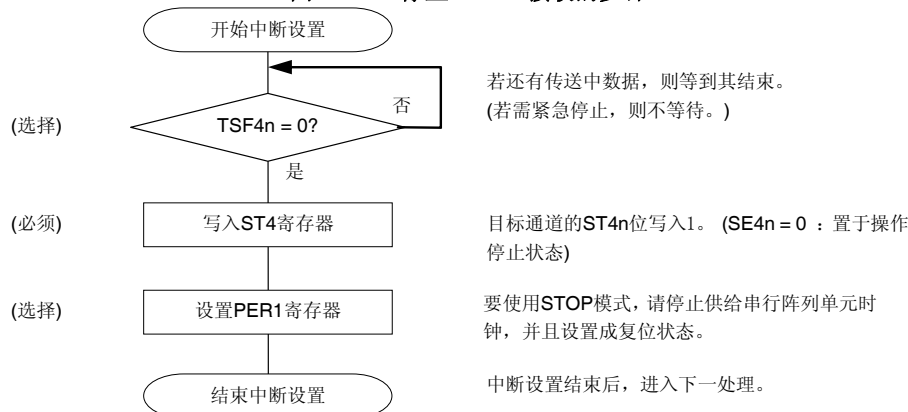
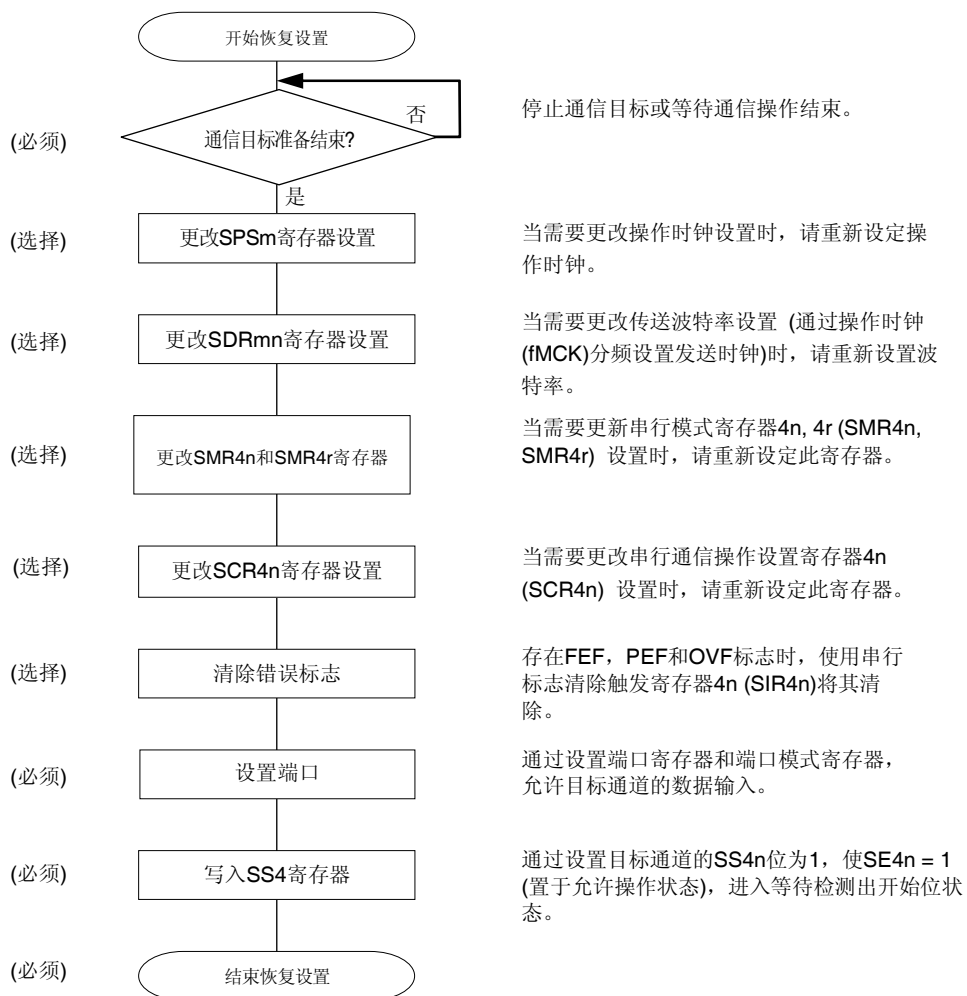
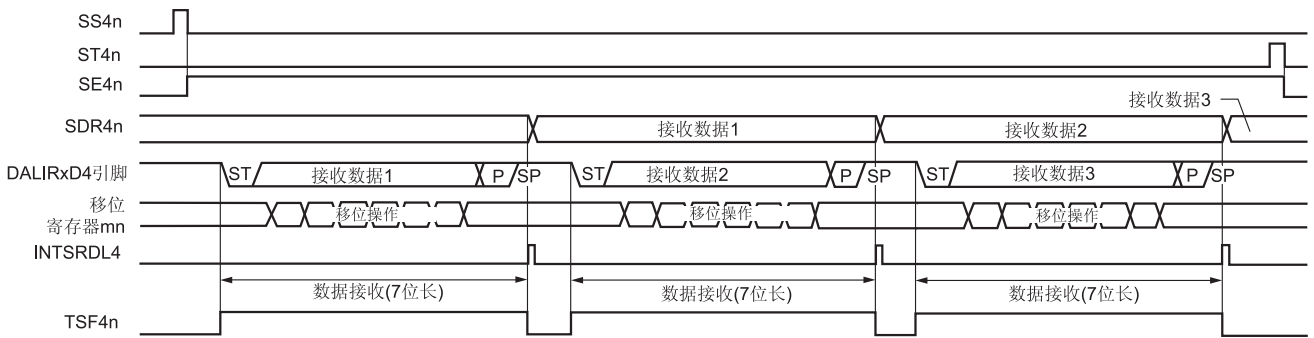


图 16-38. 恢复 UART 接收的步骤



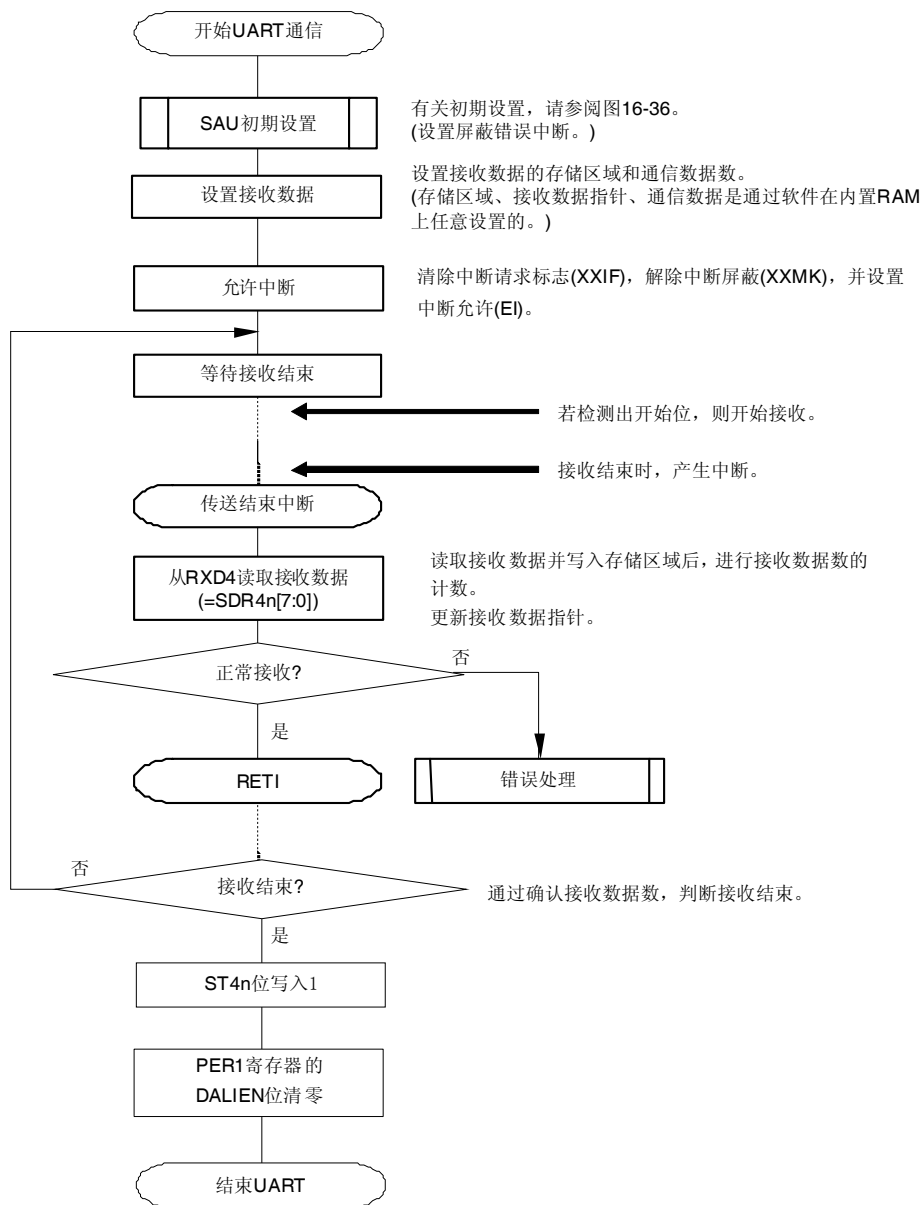
(3) 处理流程

图 16-39. UART 接收的时序图



备注 n: 通道编号(n = 1), r: 通道编号(r = 0)

图 16-40. UART 接收的流程图



注意事项 在设置外围允许寄存器 1 (PER1)的 DALIEN 位为 1 后，必须在经过 4 个或更多 fCLK 时钟后设置串行时钟选择寄存器 4 (SPS4)。

16.6 DALI模式

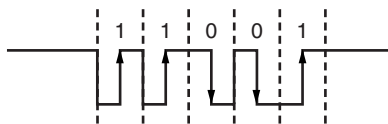
该模式时执行 DALI (Digital Addressable Lighting Interface)的从属发送/接收。

DALI 按照以下协议进行通信。

注意事项 关于 DALI 通信标准，请参阅最新标准规范。

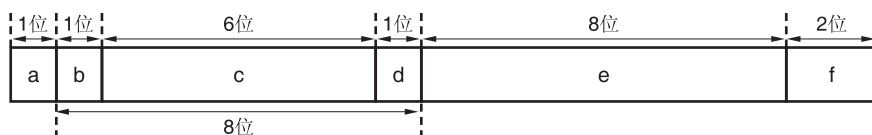
(1) 数据结构**<1> 位定义**

DALI 通信是使用曼彻斯特码，因此它的位定义是：下降沿为“0”，上升沿为“1”。在不执行通信时，DALI 通信固定为高电平。

图 16-41. 位定义**<2> 帧**

• 正向帧

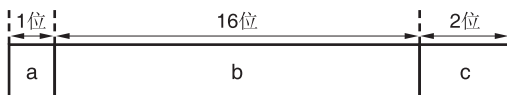
主向从属发送时使用该帧。一个帧包含 19、20 或 27 位。

图 16-42. 正向帧结构 (19 位)

- a: 开始位
显示帧的开始。总保持和“1”相同的波形。
- b-d: 地址字节
指定帧的发送目的地。
- e: 数据字节
指定命令。
- f: 停止位
显示帧的结束。固定为高电平。

• 反向帧

从属向主发送时使用该帧。一个帧包含 11、19 或 27 位。

图 16-43. 反向帧结构 (19 位)

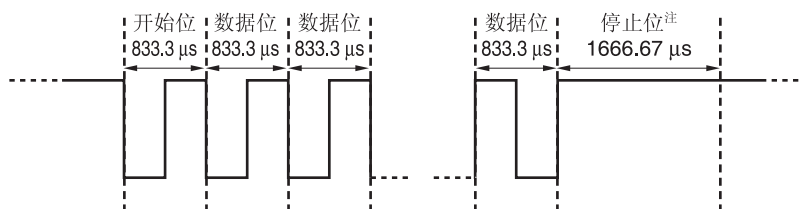
- a: 开始位
显示帧的开始。总保持和“1”相同的波形。
- b: 数据字节
回复主。
- c: 停止位
显示帧的结束。固定为高电平。

(2) 发送/接收时序规则

<1> 帧内的时序

正向帧和反向帧中的 DALI 位宽度均为 1 位(= 833.3 μ s \pm 10%)。

图 16-44. 帧内的时序



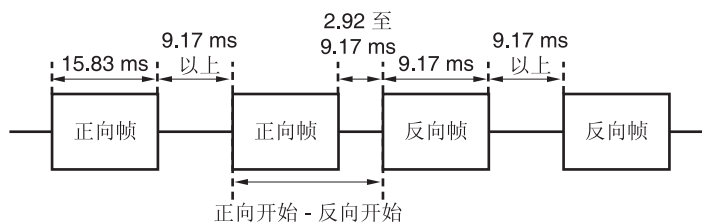
注 停止位是 2 个位，即 1666.67 μ s。

<2> 帧之间的时序

DALI 须按帧单位执行以下时序控制：

- 正向帧宽度： 15.83 ms \pm 10% (19 位), 16.67 ms \pm 10% (20 位), 22.5 ms \pm 10% (27 位)
- 反向帧宽度： 9.17 ms \pm 10% (11 位), 16.67 ms \pm 10% (20 位), 22.5 ms \pm 10% (27 位)
- 正向帧与反向帧之间的通信间隔： 2.92 至 9.17 ms
- 正向帧与下一个正向帧之间的通信间隔： 9.17 ms min.
- 反向帧与下一个正向帧之间的通信间隔： 9.17 ms min.

图 16-45. 帧之间的时序



备注 以上是 1200 bps 时的示例。

16.6.1 DALI发送

DALI 发送操作用于将数据从 RL78/I1A 异步发送至另一器件（开始-停止同步）。

在用于 DALI 的两个通道中，偶数通道用于 DALI 发送。

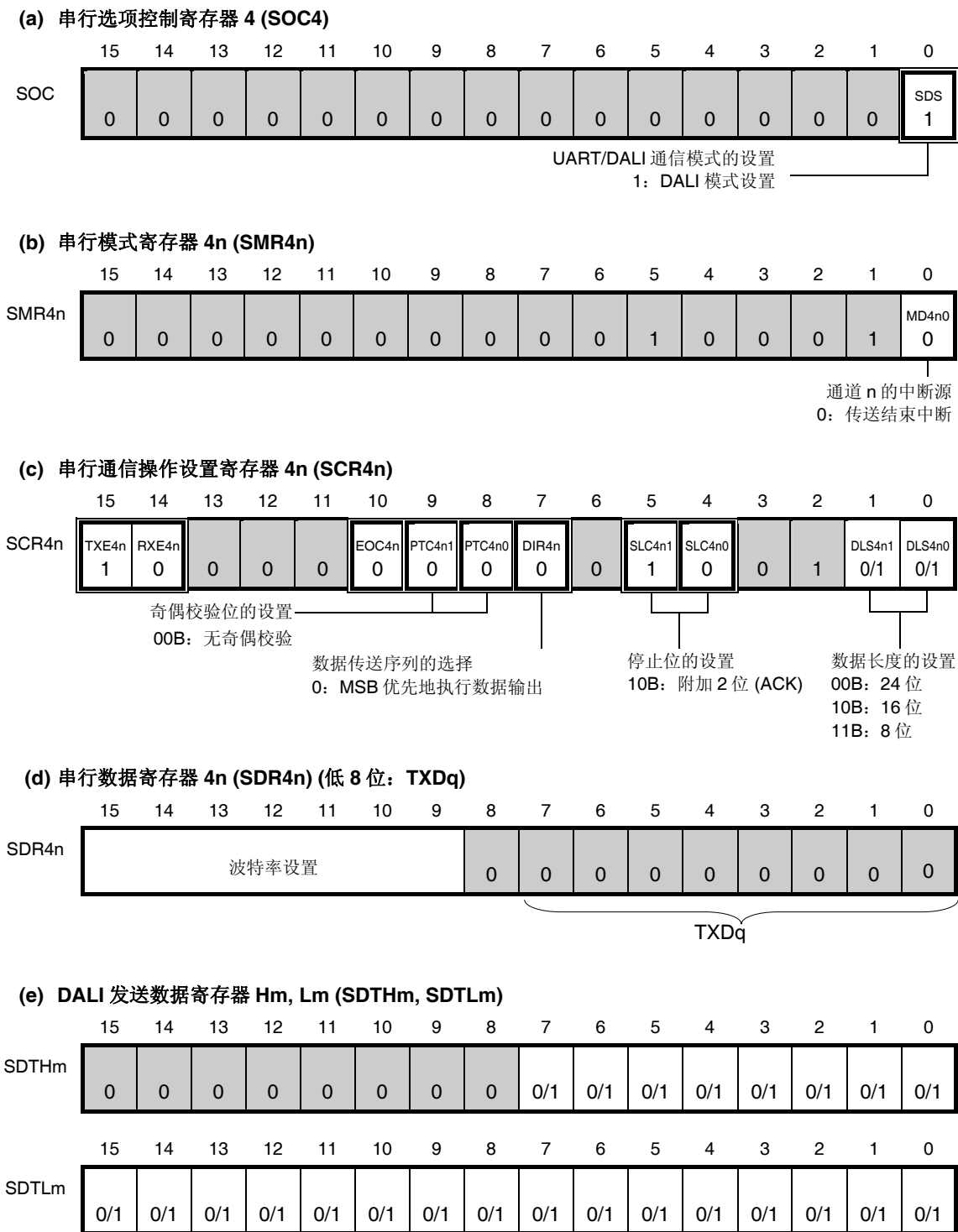
DALI发送	
目标通道	通道0
使用引脚	DALITxD4
传送数据长度	8, 16或24位
传送速率	Max. $f_{MCK}/12$ [bps] (SDR4n [15:9] = 3或以上), Min. $f_{CLK}/(2 \times 2^{11} \times 256)$ [bps] ^注
数据方向	MSB优先
数据相位	正相输出 (默认: 高电平)
奇偶校验位	无奇偶校验位
停止位	附加2位
错误检测标志	无
中断	INTSTDL4
	传送结束中断

注 在满足以上条件及电特性中的 AC 特性(参阅第三十二章 电特性)的范围内使用此操作。

- 备注 1. f_{CLK} : 系统时钟频率
 2. n: 通道编号(n = 0)

(1) 寄存器设置

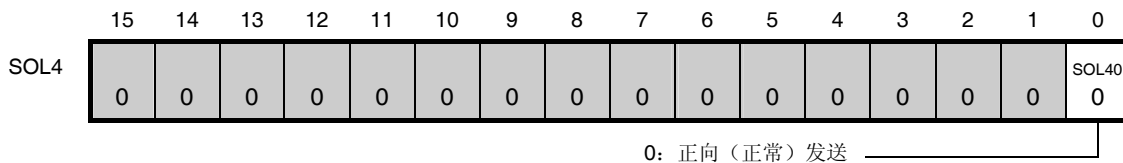
图 16-46. DALI 发送时的寄存器设置内容示例 (1/2)



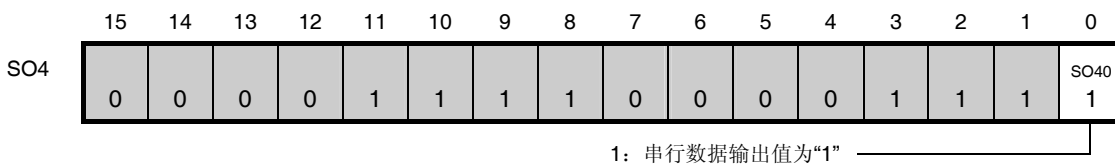
备注 1. n: 通道编号(n = 0), q: DALI/UART 编号(q = 4)
 2. □: 设置固定于 DALI 发送模式, ■: 禁止设置 (设为初始值)
 x: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)
 0/1: 可根据客户的用途设为 0 或 1

图 16-46. DALI 发送时的寄存器设置内容示例 (2/2)

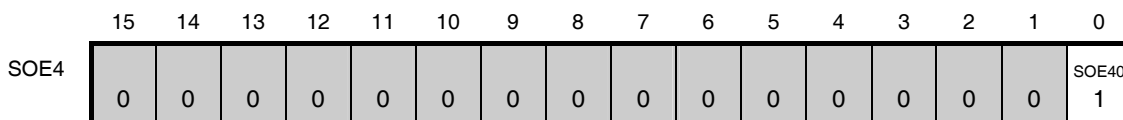
(f) 串行输出电平寄存器 4 (SOL4) ...仅设置目标通道的位。



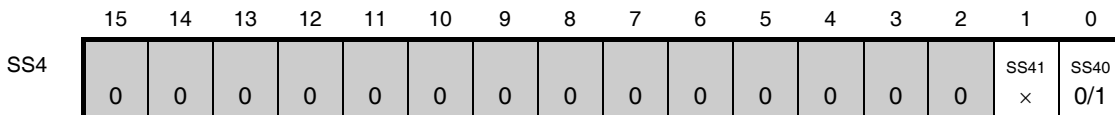
(g) 串行输出寄存器 4 (SO4) ...仅设置目标通道的位。



(h) 串行输出允许寄存器 4 (SOE4) ...仅将目标通道的位设置为 1。



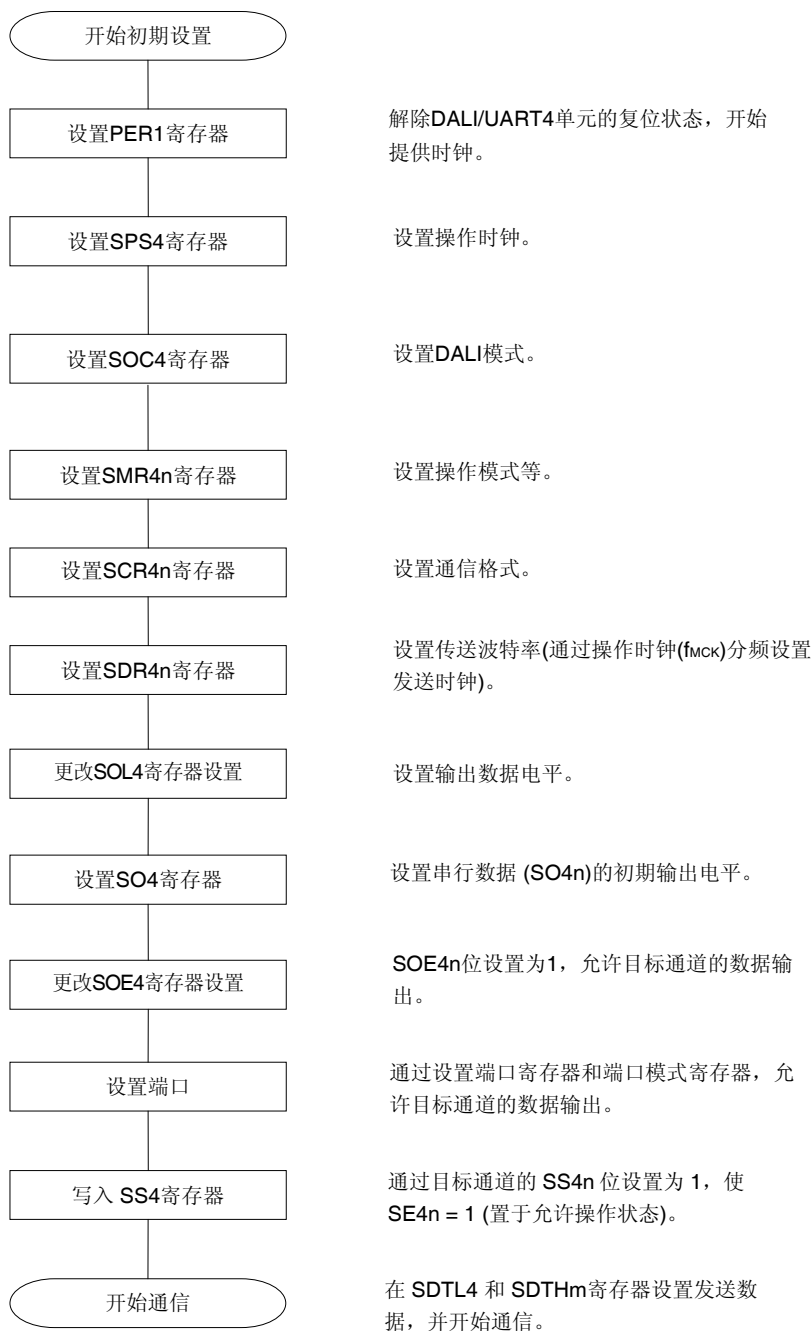
(i) 串行通道开始寄存器 4 (SS4) ...仅将目标通道的位设置为 1。



- 备注 1. n: 通道编号(n = 0)
2. : 设置固定于 DALI 发送模式, : 禁止设置 (设为初始值)
- x: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)
- 0/1: 可根据客户的用途设为 0 或 1

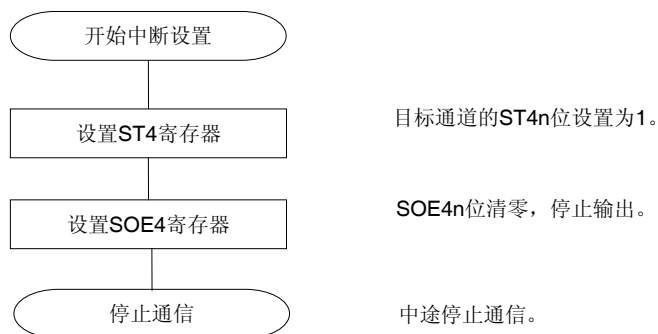
(2)操作步骤

图 16-47. DALI 发送的初始设定步骤



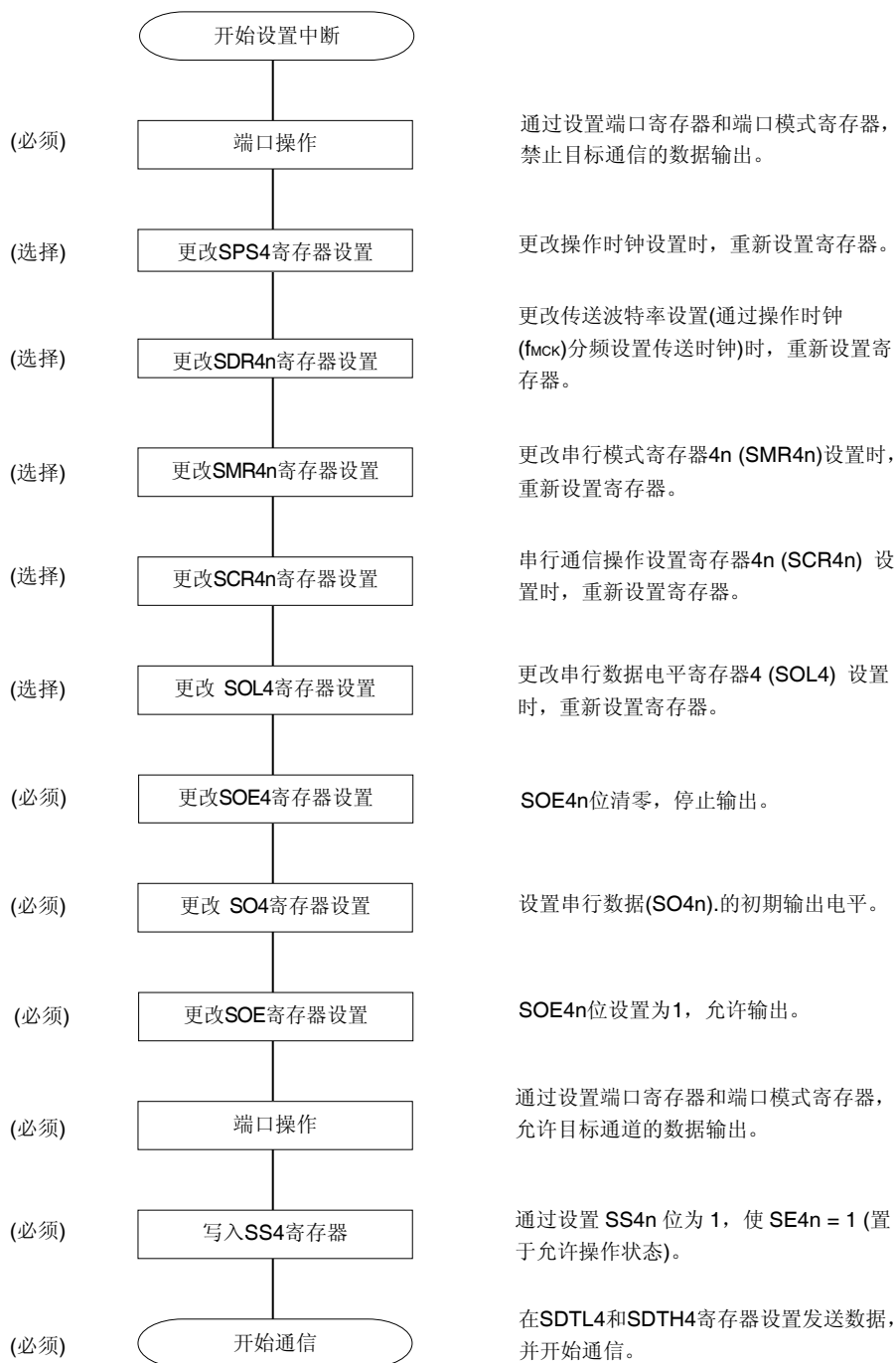
注意事项 在设置外围允许寄存器 1 (PER1)的 DALIEN 位为 1 后，必须在经过 4 个或更多 f_{CLK} 时钟后设置串行时钟选择寄存器 4 (SPS4)。

图 16-48. 停止 DALI 发送的步骤



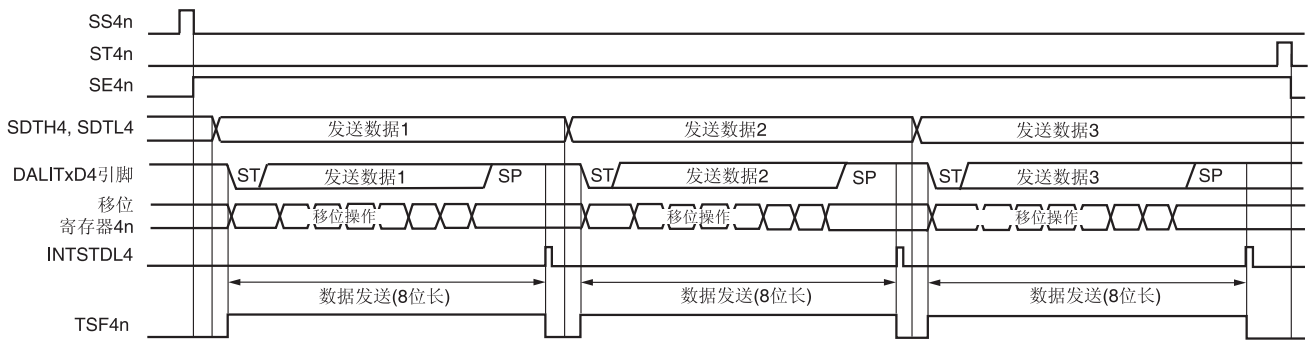
备注 即使在停止通信之后也保持引脚电平。将串行输出寄存器 4 (SO4)复位，以恢复操作 (参阅图 16-49. 恢复 DALI 发送的步骤)。

图 16-49. 恢复 DALI 发送的步骤



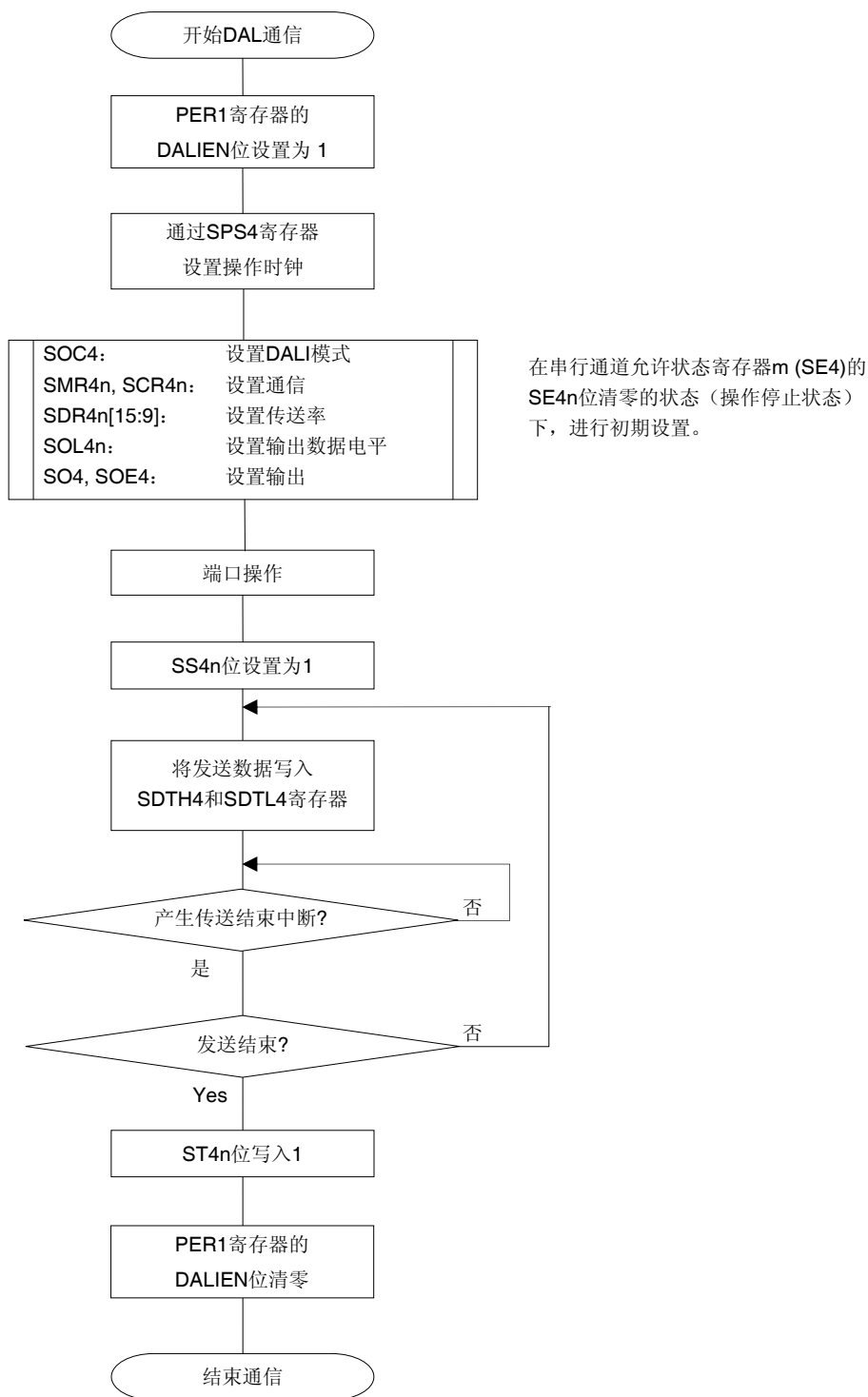
(3) 处理流程

图 16-50. DALI 发送的时序图



备注 n: 通道编号(n = 0)

图 16-51. DALI 发送的流程图



注意事项 在设置外围允许寄存器 1 (PER1)的 DALIEN 位为 1 后，必须在经过 4 个或更多 fCLK 时钟后设置串行时钟选择寄存器 4 (SPS4)。

16.6.2 DALI接收

DALI 接收操作指 RL78/I1A 从另一器件异步接收数据（开始-停止同步）。

用于 DALI 的两个通道中的奇数通道被用作 DALI 接收。必须设置奇数和偶数通道的 SMR 寄存器。

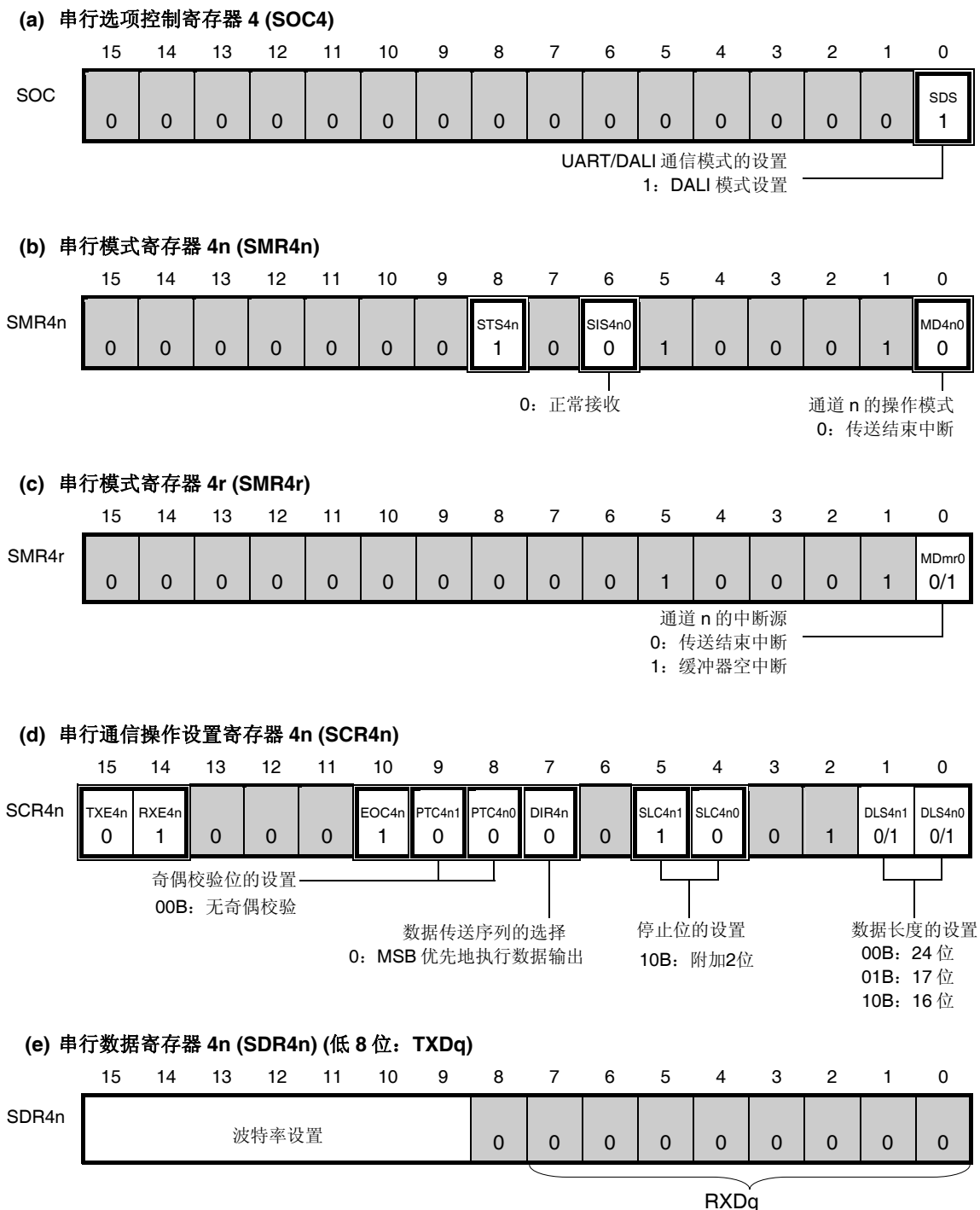
DALI接收	
目标通道	通道1
使用引脚	DALIRxD4
传送数据长度	16, 17或24位
传送速率	Max. $f_{MCK}/12$ [bps] (SDR4n [15:9] = 3或以上), Min. $f_{CLK}/(2 \times 2^{11} \times 256)$ [bps] ^注
数据方向	MSB优先
数据相位	正相输出 (默认: 高电平)
奇偶校验位	无奇偶校验位
停止位	附加2位
错误检测标志	<ul style="list-style-type: none"> • 帧错误检测标志(FEF4n) • 曼彻斯特帧错误检测标志(MFEF4n) • 溢出错误检测标志(OVF4n)
中断	INTSRDL4
	仅限传送结束中断(禁止设置缓冲器空中断。)
错误中断	INTSREDL4

注 在满足以上条件及电特性中的 AC 特性(参阅第三十二章 电特性)的范围内使用此操作。

- 备注 1.** f_{MCK} : 目标通道的操作时钟频率
 f_{CLK} : 系统时钟频率
- 2.** n: 通道编号(n = 1)

(1) 寄存器设置

图 16-52. DALI 接收时的寄存器设置内容示例(1/2)



注意事项 对于 DALI 接收，必须设置通道 r 的 SMR4r 寄存器，使其与通道 n 配对。

- 备注 1. n: 通道编号(n = 1), r: 通道编号(r = 0), q: DALI/UART 编号(q = 4)
 2. □: 设置固定于 DALI 接收模式, ■: 禁止设置 (设为初始值)
 x: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)
 0/1: 可根据客户的用途设为 0 或 1

图 16-52. DALI 接收时的寄存器设置内容示例(2/2)

(f) DALI 接收数据寄存器(SDCHm, SDCLm)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SDCHm	0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
SDCLm	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

(e) 串行输出寄存器 4 (SO4) ...此模式下不使用的寄存器。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SO4	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	SO40 ×

(f) 串行输出允许寄存器 4 (SOE4) ...此模式下不使用的寄存器。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOE4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SOE40 ×

(g) 串行通道开始寄存器 4 (SS4) ...仅将目标通道的位设置为 1。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SS4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SS41 0/1	SS40 ×

注意事项 对于 DALI 接收，必须设置通道 r 的 SMR4r 寄存器，使其与通道 n 配对。

- 备注 1. n: 通道编号(n = 1), r: 通道编号(r = 0), q: DALI/UART 编号(q = 4)
 2. □: 设置固定于 DALI 接收模式, ■: 禁止设置 (设为初始值)
 x: 此模式下不能使用的位 (任意模式下均不使用时设为初始值)
 0/1: 可根据客户的用途设为 0 或 1

(2) 操作步骤

图 16-53. DALI 接收的初始设定步骤



注意事项 在设置外围允许寄存器 1 (PER1)的 DALIEN 位为 1 后，必须在经过 4 个或更多 f_{CLK} 时钟后设置串行时钟选择寄存器 4 (SPS4)。

图 16-54. 停止 DALI 接收的步骤

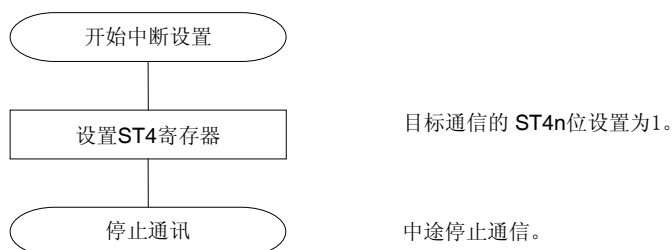
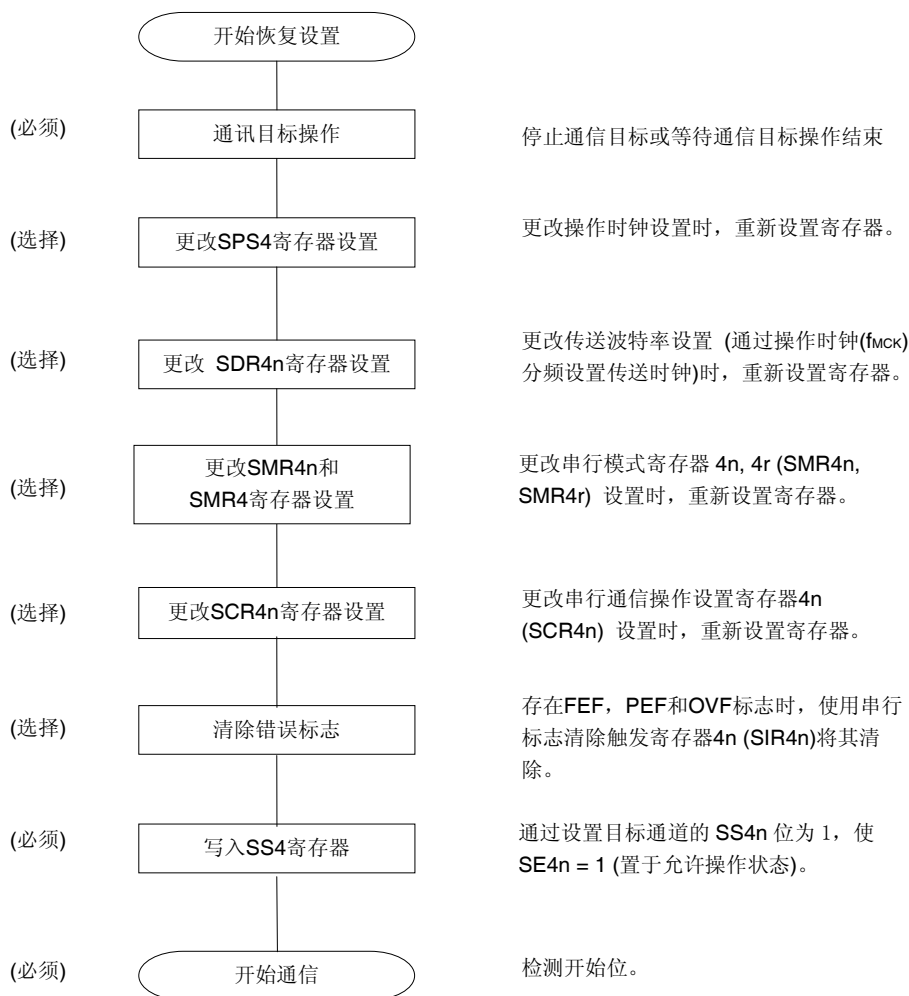
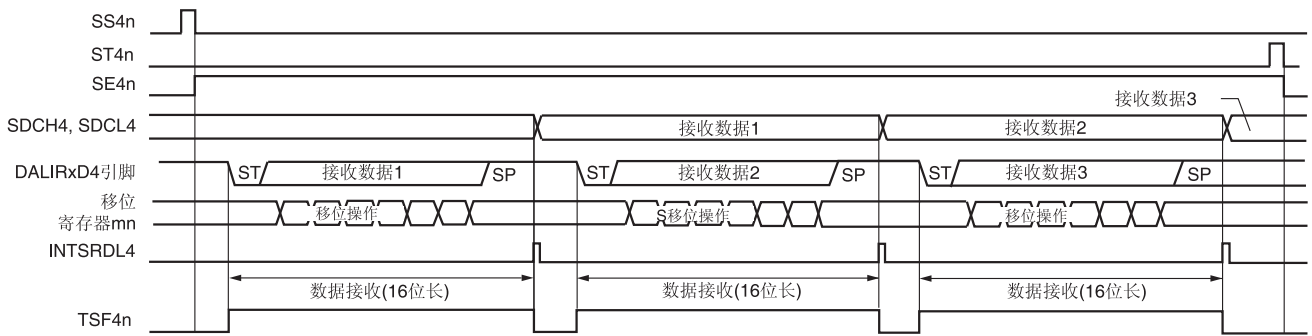


图 16-55. 恢复 DALI 接收的步骤



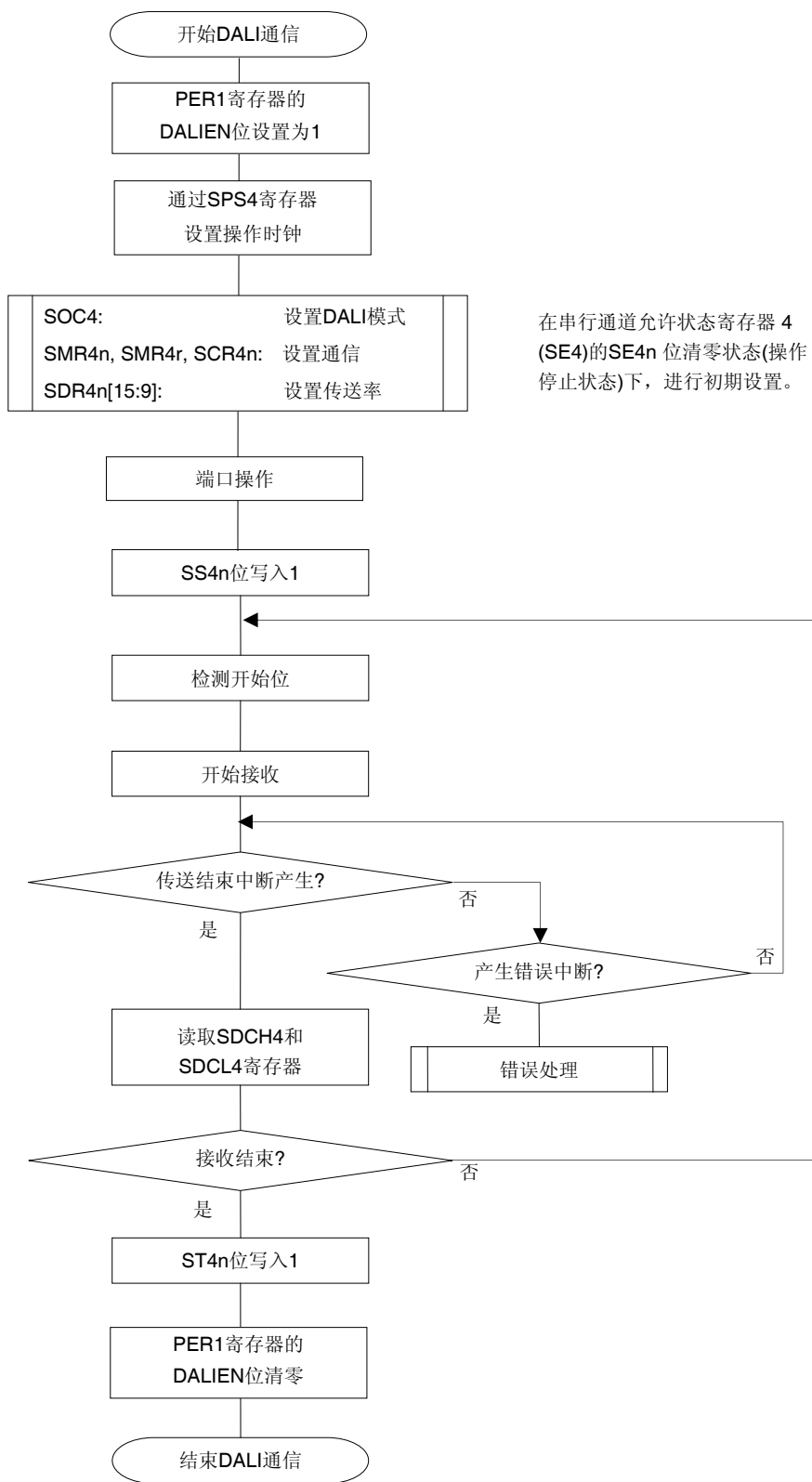
(3) 处理流程

图 16-56. DALI 接收的时序图



备注 n: 通道编号(n = 1), r: 通道编号(r = 0)

图 16-57. DALI 接收的流程图



在串行通道允许状态寄存器 4 (SE4)的SE4n 位清零状态(操作停止状态)下, 进行初期设置。

注意事项 在设置外围允许寄存器 1 (PER1)的 DALIEN 位为 1 后, 必须在经过 4 个或更多 fCLK 时钟后设置串行时钟选择寄存器 4 (SPS4)。

16.7 SNOOZE模式功能 (仅限DALI/UART4 接收)

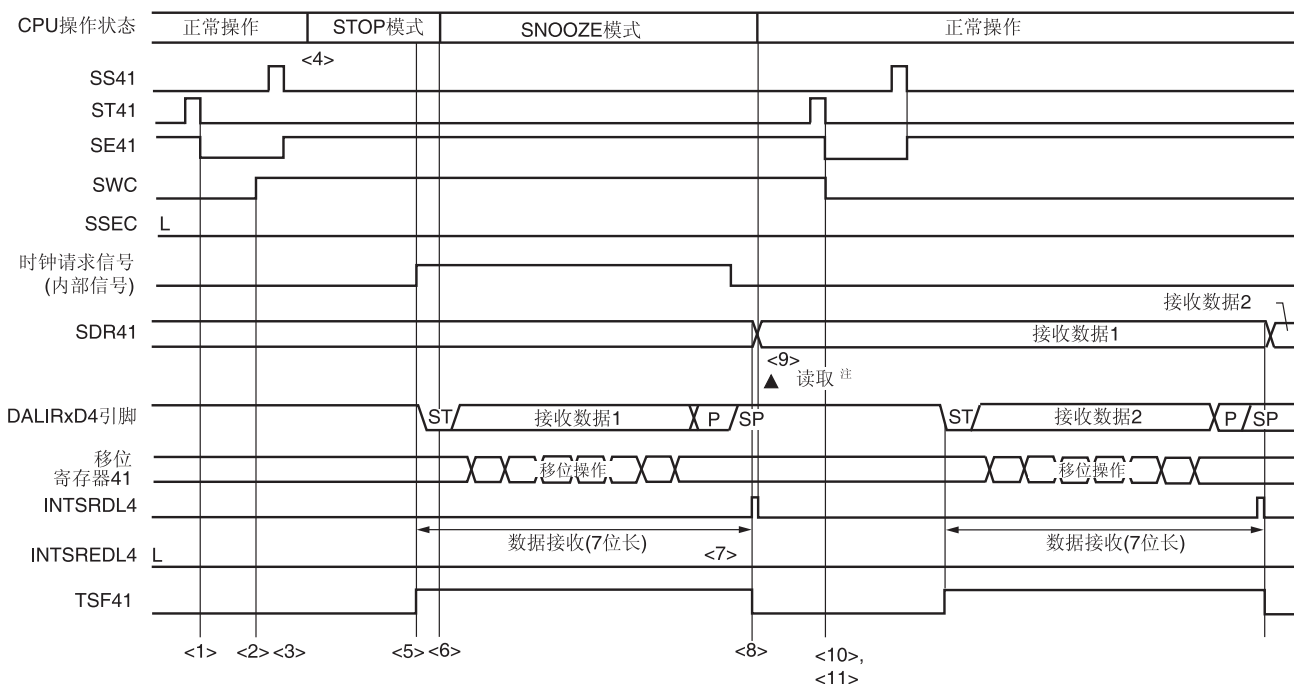
DALI 接收以及 UART 接收(通道 1)支持 SNOOZE 模式。借助 SNOOZE 模式，在 STOP 模式下检测到 DALIRxD4 引脚输入时不需要 CPU 而执行数据接收。

当使用 SNOOZE 模式功能时，须在切换至 STOP 模式之前将串行待机控制寄存器 4 (SSC4)的 SWC 位设置为 1。

- 注意事项 1.** 仅限选择高速片上振荡器时钟作为 f_{CLK}时可以设置 SNOOZE 模式。并且，SNOOZE 模式时不能执行 PLL 输出。
 从 SNOOZE 模式返回正常操作模式后，可以执行 PLL 输出。这种情况下，时钟可能产生“+1.125 sec (max.) 至 -0.406 sec”范围内的误差(使用 PLL 时 f_{CLK} = 32 MHz ↔ 4 MHz)。
 由于 DALI 通信是采用曼彻斯特协议，该误差对其不产生影响。
- 2.** 在 SNOOZE 模式下使用 UART 时，最大传送率为 9600 bps。
3. 在 DALI 模式下使用时，最大传送率为 4800 bps。

(1) SNOOZE 模式操作(正常操作)

图 16-58. SNOOZE 模式操作(正常操作模式)的时序图(UART 接收)



注 仅当 SWC = 1 且在检测到 DALIRxD4 引脚输入的下一个边沿时才读取接收的数据。

注意事项 在切换至 SNOOZE 模式之前，以及在 SNOOZE 模式下完成接收操作之后，必须将 ST41 位设置为 1 并清除 SE41 位（停止操作）。

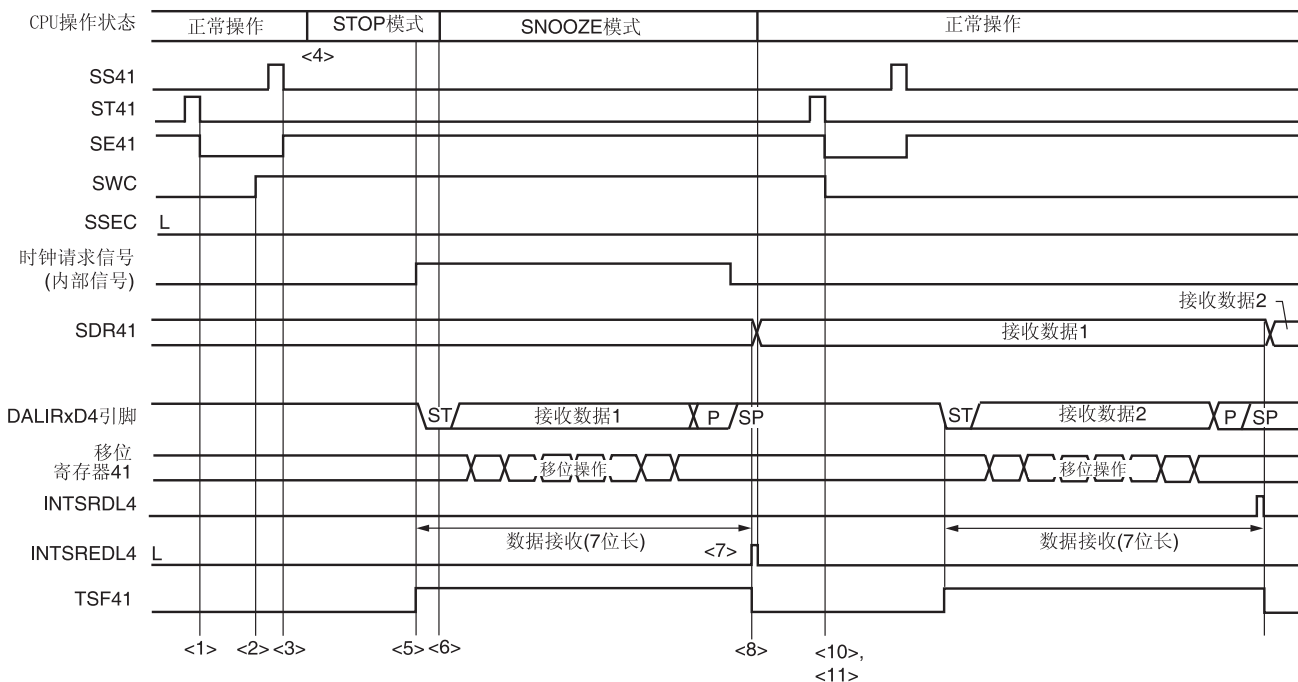
备注 此图中的<1>至<11>对应于图 16-60. SNOOZE 模式操作(正常操作/异常操作<1>)的流程图中的<1>至<11>。

(2) SNOOZE 模式操作(异常操作<1>)

异常操作<1> 指在 SSEC = 0 的情况下发生通信错误时执行的操作。

由于 SSEC = 0，所以在发生通信错误时将产生一个错误中断(INTSREDL4)。

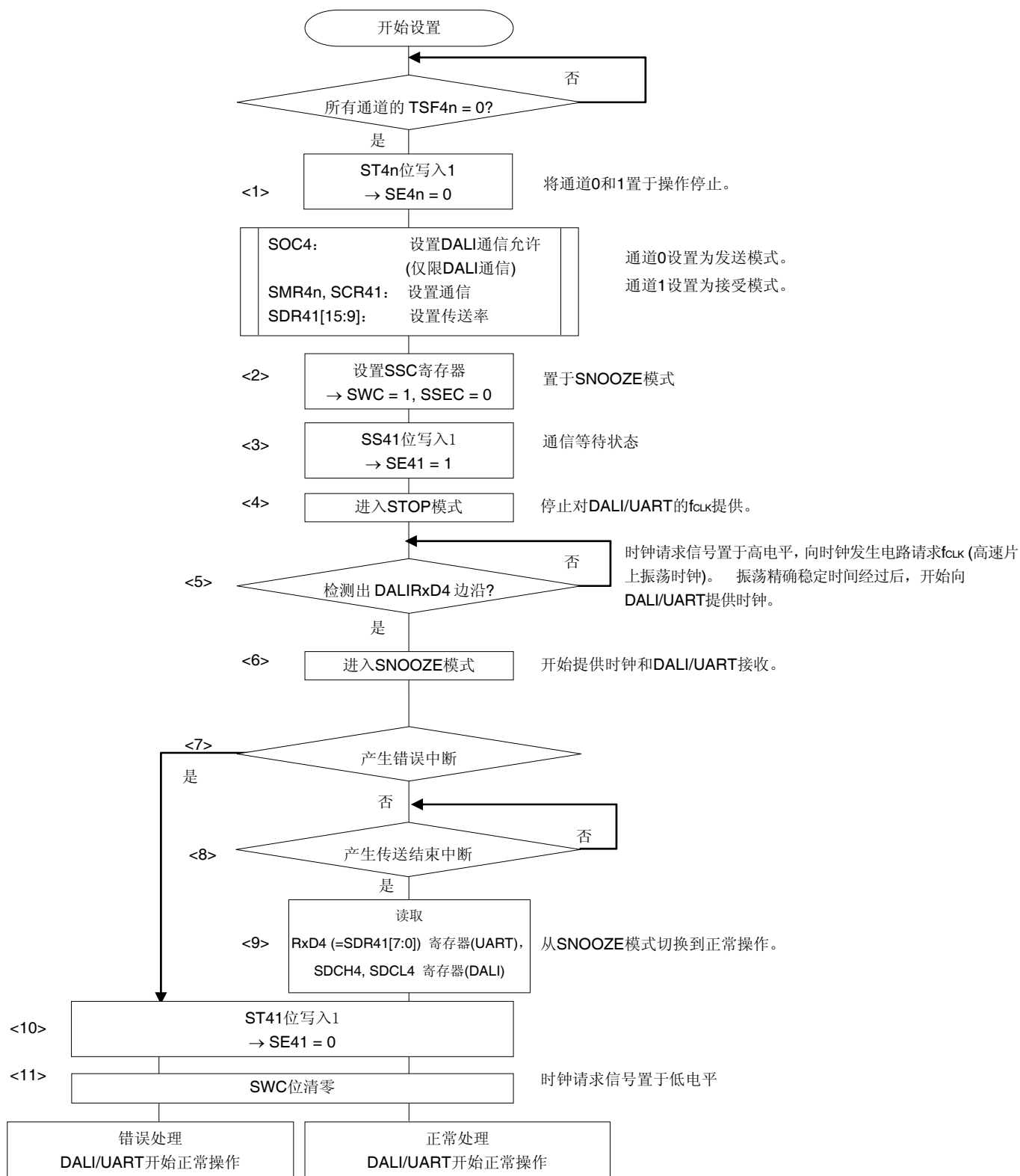
图 16-59. SNOOZE 模式操作(异常操作<1>)的时序图(UART 接收)



注意事项 在切换至 SNOOZE 模式之前，以及在 SNOOZE 模式下完成接收操作之后，必须将 ST41 位设置为 1 并清除 SE41 位（停止操作）。

备注 此图中的<1>至<11>对应于图 16-60. SNOOZE 模式操作(正常操作/异常操作<1>)的流程图中的<1>至<11>。

图 16-60. SNOOZE 模式操作(正常操作/异常操作<1>)的流程图



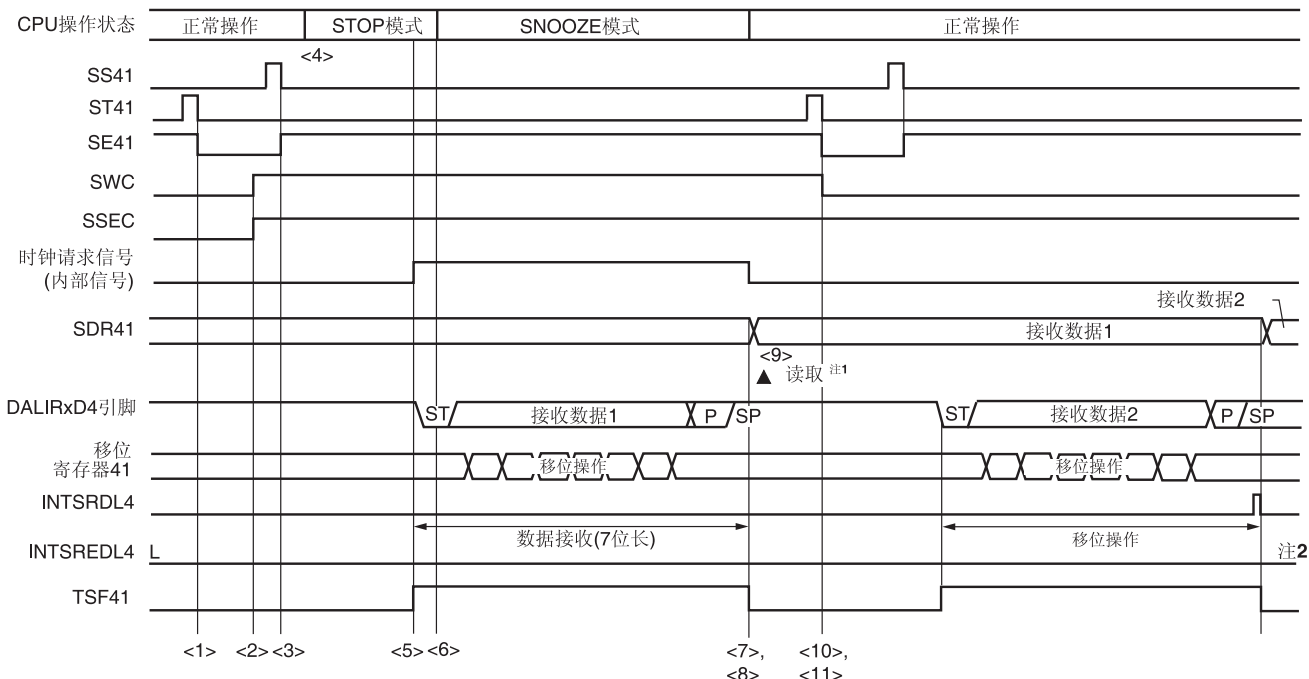
备注 此图中的<1>至<11>对应于图 16-58. SNOOZE 模式操作(正常操作模式)的时序图(UART 接收)和图 16-59. SNOOZE 模式操作(异常操作<1>)的时序图(UART 接收)中的<1>至<11>。

(3) SNOOZE 模式操作(异常操作<2>)

异常操作<2> 指在 SSEC = 1 的情况下发生通信错误时执行的操作。

由于 SSEC = 1，所以在发生通信错误时不产生错误中断 (INTSREDL4)。

图 16-61. SNOOZE 模式操作(异常操作<2>)的时序图(UART 接收)

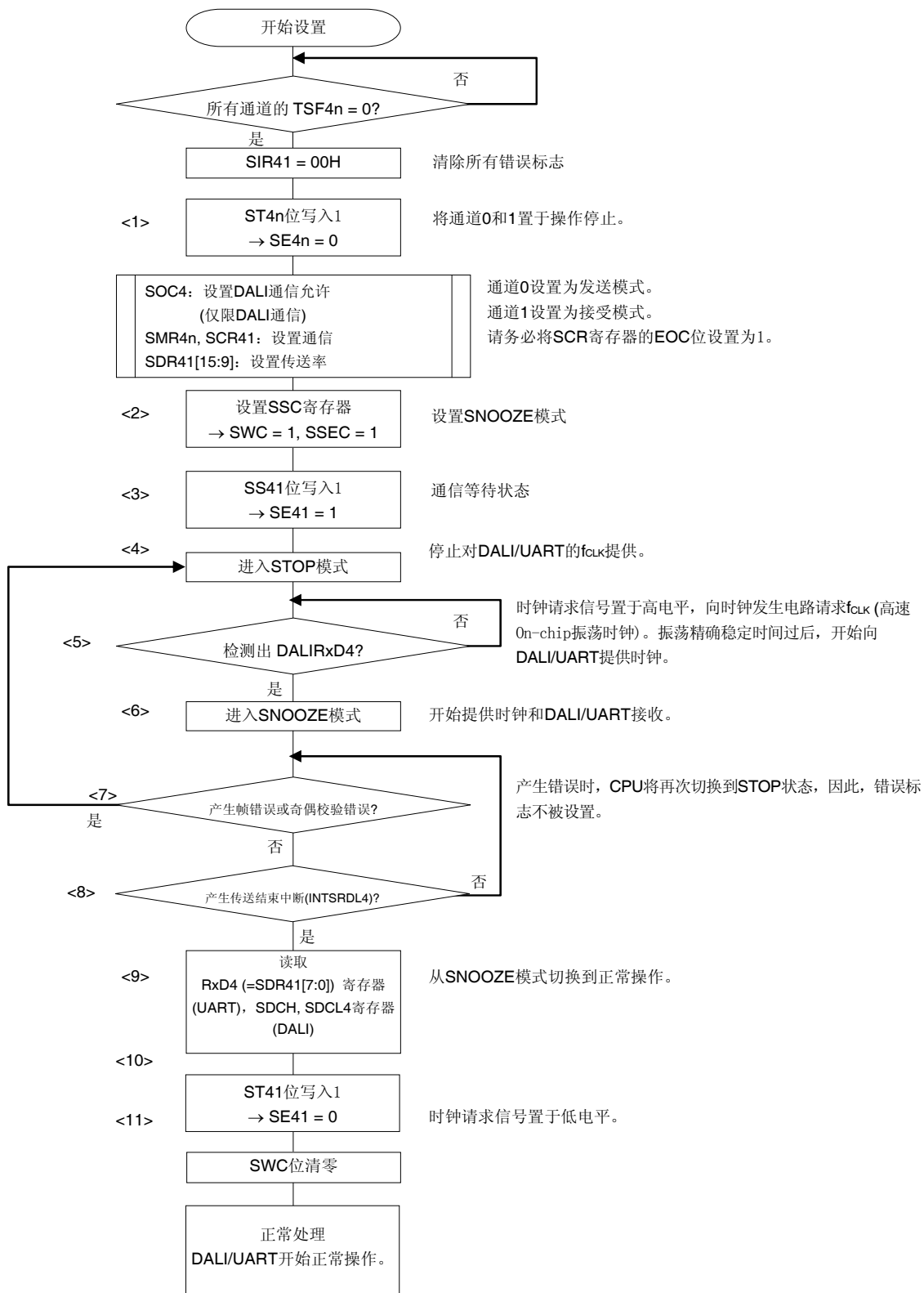


- 注 1. 仅当 SWC = 1 且在检测到 DALIRxD4 引脚输入的下一个边沿时才读取接收的数据。
- 2. SNOOZE 模式下当 DALI/UART4 成功完成接收时，不需要更改设置就可以继续执行正常的接收操作，但是，由于 SSEC = 1，因此，即使发生帧错误或奇偶校验错误，PEF41 和 FEF41 位也不会被设置。另外，也不会产生错误中断 (INTSREDL4)。

- 注意事项 1. 在切换至 SNOOZE 模式之前，以及在 SNOOZE 模式下完成接收操作之后，必须将 ST41 位设置为 1 并清除 SE41 位（停止操作）。
- 2. 当在 SSEC 被设置为 1 的情况下使用 SNOOZE 模式时，不会发生溢出错误。因此，当使用 SNOOZE 模式时，要在切换至 STOP 模式之前读取 SDR41 寄存器的位 7 至位 0 (RxD4)。

备注 此图中的<1>至<11>对应于图 16-62. SNOOZE 模式操作(异常操作<2>)的流程图中的<1>至<11>。

图 16-62. SNOOZE 模式操作(异常操作<2>)的流程图



(注意事项和备注如下页所示。)

注意事项 当在 SSEC 被设置为 1 的情况下使用 SNOOZE 模式时，不会发生溢出错误。因此，当使用 SNOOZE 模式时，要在切换至 STOP 模式之前读取 SDR41 寄存器的位 7 至位 0 (RxD4)(DALI 接收时，SDCH4 和 SDCL4 寄存器)。

备注 此图中的<1>至<11>对应于图 11-61. SNOOZE 模式操作(异常操作<2>)的时序图(UART 接收)中的<1>至<11>。

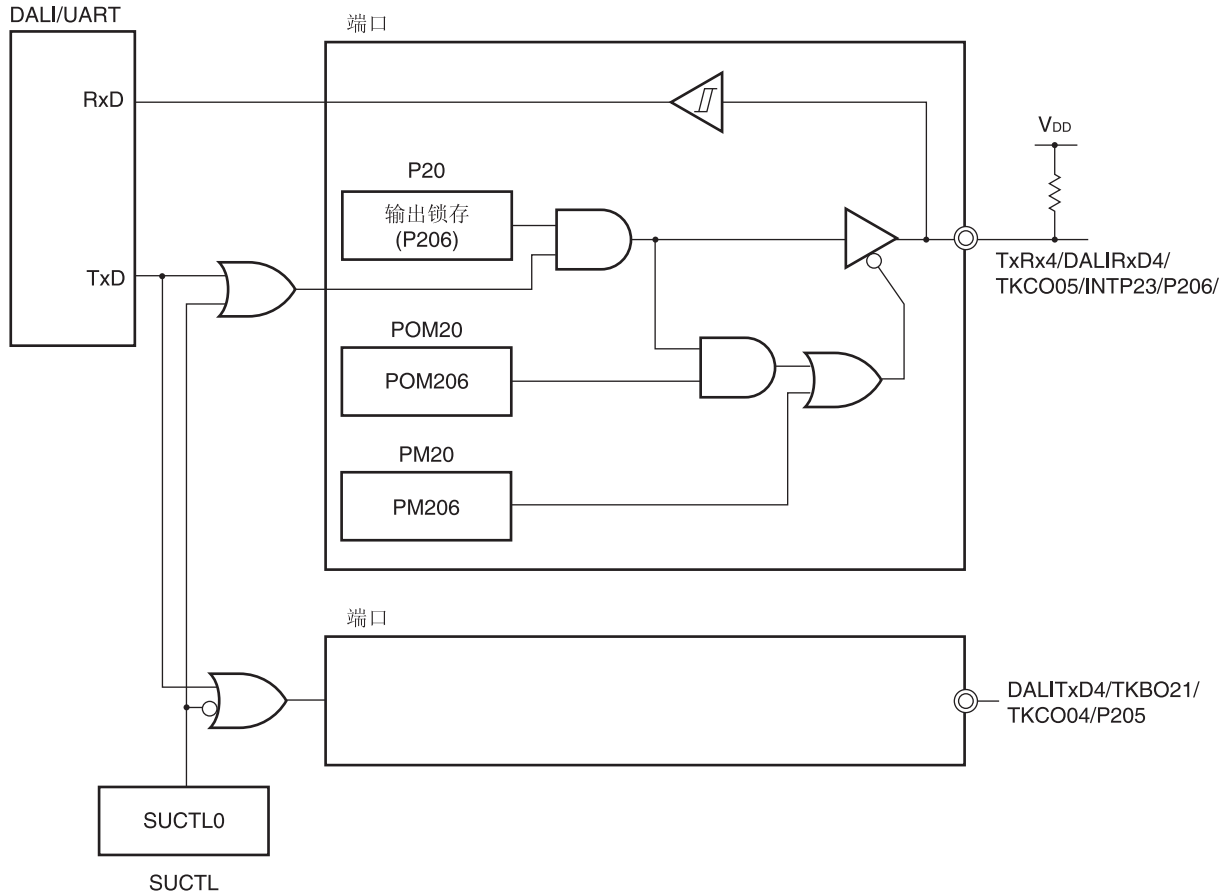
16.8 单线数据模式

通过将单线数据控制寄存器(SUCTL)的 SUCTL0 位设置为 1，可以执行单线 DALI/UART4 通信。

使用单线模式时，P206/TxRx4/TKCO05/DALIRxD4/INTP23 引脚兼用于发送和接收。

单线数据模式的框图如图 16-63 所示。

图 16-63. 单线数据模式的框图



- 注意事项**
1. 在单线模式下使用 DALI/UART4 时，在发送的同时，发送数据被接收。
 2. 在不使用且 POM20 寄存器中 POM206 = 1(N 沟道开漏输出模式)时，可能发生外部信号冲突。

备注 P20: 端口寄存器 20
 PM20: 端口模式寄存器 20
 POM20: 端口输出模式寄存器 20

16.9 计算波特率

(1) 波特率计算公式

用于 DALI/UART4 通信的波特率可以通过以下表达式计算。

UART 通信

$$\text{(波特率)} = \{\text{目标通道的操作时钟}(f_{\text{MCK}})\text{频率}\} \div (\text{SDR4n}[15:9] + 1) \div 2 \text{ [bps]}$$

DALI 通信

$$\text{(波特率)} = \{\text{目标通道的操作时钟}(f_{\text{MCK}})\text{频率}\} \div (\text{SDR4n}[15:9] + 1) \div 4 \text{ [bps]}$$

注意事项 禁止将串行数据寄存器 4n (SDR4n) 设置为 $\text{SDR4n}[15:9] = (0000000\text{B}, 0000001\text{B}, 0000010\text{B})$ 。

- 备注 1.** 当使用 DALI/UART4 时，SDR4n[15:9] 的值为 SDR4n 寄存器位 15 至位 9 的值(0000011B 至 1111111B)，即 3 至 127。
- 2.** UART 通信的 2 位对应于 DALI 通信的 1 位，因此当 f_{MCK} 和 SDR4n 的设置相同时，DALI 通信的波特率为 UART 通信波特率的一半。
- 3.** n: 通道编号 (n = 0, 1)

利用串行时钟选择寄存器 4 (SPS4) 来指定操作时钟(f_{MCK})。

表 16-2. DALI/UART4 操作时钟的选择

SPS4 寄存器				操作时钟(fmck) [*]					
PRSm03	PRSm02	PRSm01	PRSm00	f _{CLK} = 4MHz (不使用 PLL)		f _{CLK} = 16MHz (使用 PLL)		f _{CLK} = 32 MHz (使用 PLL)	
0	0	0	0	f _{CLK}	4 MHz	f _{CLK}	16 MHz	f _{CLK}	32 MHz
0	0	0	1	f _{CLK}	4 MHz	f _{CLK}	16 MHz	f _{CLK} / 2	16 MHz
0	0	1	0	f _{CLK}	4 MHz	f _{CLK} / 2	8 MHz	f _{CLK} / 2 ²	8 MHz
0	0	1	1	f _{CLK}	4 MHz	f _{CLK} / 2 ²	4 MHz	f _{CLK} / 2 ³	4 MHz
0	1	0	0	f _{CLK} / 2	2 MHz	f _{CLK} / 2 ³	2 MHz	f _{CLK} / 2 ⁴	2 MHz
0	1	0	1	f _{CLK} / 2 ²	1 kHz	f _{CLK} / 2 ⁴	1 kHz	f _{CLK} / 2 ⁵	1 kHz
0	1	1	0	f _{CLK} / 2 ³	500 kHz	f _{CLK} / 2 ⁵	500 kHz	f _{CLK} / 2 ⁶	500 kHz
0	1	1	1	f _{CLK} / 2 ⁴	250 kHz	f _{CLK} / 2 ⁶	250 kHz	f _{CLK} / 2 ⁷	250 kHz
1	0	0	0	f _{CLK} / 2 ⁵	125 kHz	f _{CLK} / 2 ⁷	125 kHz	f _{CLK} / 2 ⁸	125 kHz
1	0	0	1	f _{CLK} / 2 ⁶	62.5 kHz	f _{CLK} / 2 ⁸	62.5 kHz	f _{CLK} / 2 ⁹	62.5 kHz
1	0	1	0	f _{CLK} / 2 ⁷	31.25 kHz	f _{CLK} / 2 ⁹	31.25 kHz	f _{CLK} / 2 ¹⁰	31.25 kHz
1	0	1	1	f _{CLK} / 2 ⁸	15.63 kHz	f _{CLK} / 2 ¹⁰	15.63 kHz	f _{CLK} / 2 ¹¹	15.63 kHz
1	1	0	0	f _{CLK} / 2 ⁹	7.81 kHz	f _{CLK} / 2 ¹¹	7.81 kHz	f _{CLK} / 2 ¹²	7.81 kHz
1	1	0	1	f _{CLK} / 2 ¹⁰	3.91 kHz	f _{CLK} / 2 ¹²	3.91 kHz	f _{CLK} / 2 ¹³	3.91 kHz
1	1	1	0	f _{CLK} / 2 ¹¹	1.95 kHz	f _{CLK} / 2 ¹³	1.95 kHz	f _{CLK} / 2 ¹⁴	1.95 kHz
1	1	1	1	f _{CLK} / 2 ¹²	977Hz	f _{CLK} / 2 ¹⁴	977Hz	f _{CLK} / 2 ¹⁵	977Hz

注 更改被选为 f_{CLK} 的时钟时（通过更改系统时钟控制寄存器(CKC)的值），必须在停止串行阵列单元 4 (DALI/UART4)操作（串行通道停止寄存器 4 (STm) = 000FH）之后才可以执行。

备注 n: 通道编号 (n = 0, 1)

(2) 发送期间的波特率误差

发送过程中 DALI/UART4 通信的波特率误差可以用以下计算式算出。确保发送侧的波特率处于接收侧允许的波特率范围之内。

$$\text{(波特率误差)} = (\text{计算波特率值}) \div (\text{目标波特率}) \times 100 - 100 [\%]$$

f_{CLK} = 32 MHz 时波特率的设置示例如下所示。

• UART 通信

UART波特率 (目标波特率)	f _{CLK} = 32 MHz			
	操作时钟(f _{MCK})	SDR4n[15:9]	计算波特率	与目标波特率之间的误差
300 bps	f _{CLK} /2 ⁹	103	300.48 bps	+0.16 %
600 bps	f _{CLK} /2 ⁸	103	600.96 bps	+0.16 %
1200 bps	f _{CLK} /2 ⁷	103	1201.92 bps	+0.16 %
2400 bps	f _{CLK} /2 ⁶	103	2403.85 bps	+0.16 %
4800 bps	f _{CLK} /2 ⁵	103	4807.69 bps	+0.16 %
9600 bps	f _{CLK} /2 ⁴	103	9615.38 bps	+0.16 %
19200 bps	f _{CLK} /2 ³	103	19230.8 bps	+0.16 %
31250 bps	f _{CLK} /2 ³	63	31250.0 bps	±0.0 %
38400 bps	f _{CLK} /2 ²	103	38461.5 bps	+0.16 %
76800 bps	f _{CLK} /2	103	76923.1 bps	+0.16 %
153600 bps	f _{CLK}	103	153846 bps	+0.16 %
312500 bps	f _{CLK}	50	312500 bps	+0.39 %

• DALI 通信

UART 波特率 (目标波特率)	操作频率(f _{CLK})	操作时钟(f _{MCK})	SDR4n [15:9]	计算波特率	与目标波特率之间的 误差
1200 bps	32 MHz	f _{CLK} /2 ⁶	103	1202 bps	+0.16 %
		f _{CLK} /2 ⁷	51	1202 bps	+0.16 %
		f _{CLK} /2 ⁸	25	1202 bps	+0.16 %
		f _{CLK} /2 ⁹	12	1202 bps	+0.16 %
	16 MHz	f _{CLK} /2 ⁵	103	1202 bps	+0.16 %
		f _{CLK} /2 ⁶	51	1202 bps	+0.16 %
		f _{CLK} /2 ⁷	25	1202 bps	+0.16 %
		f _{CLK} /2 ⁸	12	1202 bps	+0.16 %
	4 MHz	f _{CLK} /2 ³	103	1202 bps	+0.16 %
		f _{CLK} /2 ⁴	51	1202 bps	+0.16 %
		f _{CLK} /2 ⁵	25	1202 bps	+0.16 %
		f _{CLK} /2 ⁶	12	1202 bps	+0.16 %
	24 MHz	f _{CLK} /2 ⁶	77	1202 bps	+0.16 %
		f _{CLK} /2 ⁷	38	1202 bps	+0.16 %
		f _{CLK} /2 ⁸	19	1172 bps	+2.34 %
		f _{CLK} /2 ⁸	18	1234 bps	+2.8 %
		f _{CLK} /2 ⁹	9	1172 bps	+2.34 %
		f _{CLK} /2 ¹⁰	4	1172 bps	+2.34 %
	20 MHz	f _{CLK} /2 ⁶	64	1202 bps	+0.16 %
		f _{CLK} /2 ⁷	32	1184 bps	+1.73 %
f _{CLK} /2 ⁷		31	1221 bps	+1.36 %	
f _{CLK} /2 ⁸		15	1221 bps	+1.36 %	
f _{CLK} /2 ⁹		7	1221 bps	+1.36 %	
f _{CLK} /2 ¹⁰		3	1221 bps	+1.36 %	

备注 n: 通道编号(n = 0)

(3) 接收时的波特率允许范围

DALI/UART4 通信过程中允许接收的波特率范围可以用以下计算式算出。确保发送侧的波特率处于接收侧允许的波特率范围之内。

$$\text{(最大可接收波特率)} = \frac{2 \times k \times \text{Nfr}}{2 \times k \times \text{Nfr} - k + 2} \times \text{Brate}$$

$$\text{(最小可接收波特率)} = \frac{2 \times k \times (\text{Nfr} - 1)}{2 \times k \times \text{Nfr} - k - 2} \times \text{Brate}$$

Brate: 接收方的计算波特率值 (参阅 16.9 (1) 波特率计算公式。)

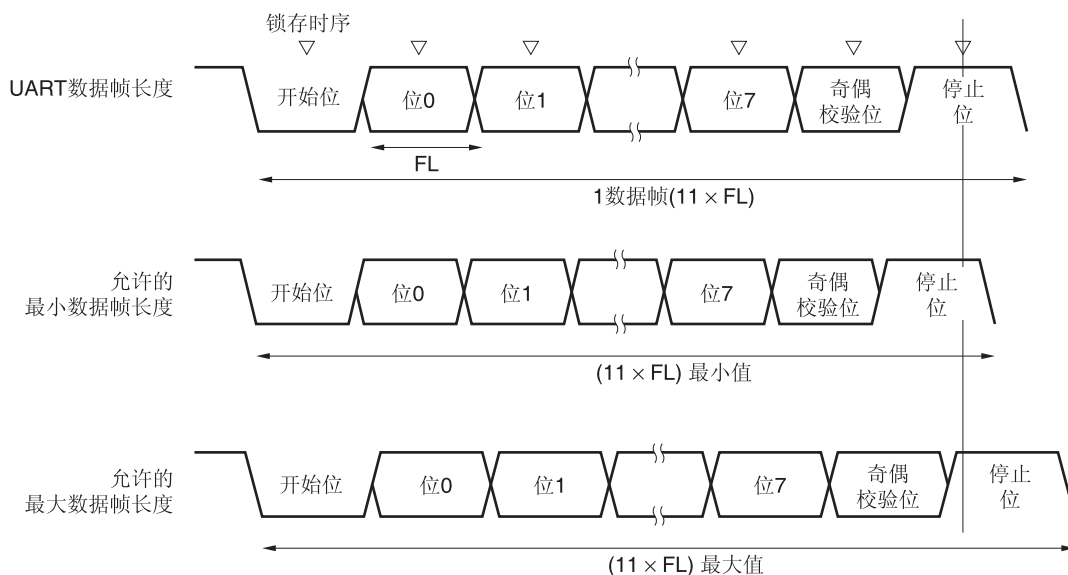
k: SDR4n[15:9] + 1

Nfr: 1 数据帧长度 [位]

= (开始位) + (数据长度) + (奇偶校验位) + (停止位)

备注 n: 通道编号(n = 1), 4n = 01, 03

图 16-64. 接收时的允许波特率范围(UART 通信, 1 数据帧长度 = 11 位时)



如图 16-64 所示, 接收数据的锁存时序取决于检测到起始位之后由串行数据寄存器 4n (SDR4n)的位 15 至位 9 设置的分频比。如果在该锁存时间之前收到最后数据 (停止位), 则可以正常进行接收。

16.10 DALI/UART4 通信时的发生错误时的处理步骤

在 DALI/UART4 通信过程中发生的错误的处理步骤如图 16-65 和图 16-66 所示。

图 16-65. 发生奇偶校验错误或溢出错误时的处理步骤

软件操作	硬件状态	备注
读取串行数据寄存器 4n (SDR4n)。→	SSR4n寄存器的BFF4n位被设置为0，并允许通道n接收数据。	这是为了防止在处理错误的过程中完成了下一次接收时所发生的溢出错误。
读取串行状态寄存器4n (SSR4n)。		识别出错误类型，并用读取值来清除错误标志。
将1写入串行标志清除触发寄存器4n (SIR4n)。→	错误标志被清除。	将SSR4n寄存器的读取值直接写入SIR4n寄存器，从而可以只清除读取时的错误。

图 16-66. 发生帧错误和曼彻斯特帧错误时的处理步骤

软件操作	硬件状态	备注
读取串行数据寄存器 4n (SDR4n) ^注 。→	SSR4n寄存器的BFF4n位被设置为0，并允许通道n接收数据。	这是为了防止在处理错误的过程中完成了下一次接收时所发生的溢出错误。
读取串行状态寄存器4n (SSR4n)。		识别出错误类型，并用读取值来清除错误标志。
写入串行标志清除触发寄存器4n (SIR4n) →	错误标志被清除。	将SSR4n寄存器的读取值直接写入SIR4n寄存器，从而可以只清除读取时的错误。
将串行通道停止寄存器4 (STm)的ST4n位设置为1。→	串行通道允许状态寄存器4 (SEm)的SE4n位被设置为0，通道n停止工作。	
与其他通信方同步		起始位的偏移会使系统判断为出现了帧错误，因此重建与其他通信方的同步并恢复通信。
将串行通道开始寄存器4 (SS4)的SS4n位设置为1。→	串行通道允许状态寄存器4 (SEm)的SE4n位被设置为1，并允许操作通道n。	

注 在执行 DALI 接收或发生曼彻斯特帧错误时，读取 DALI 接收数据寄存器(SDCHm, SDCLm)。

备注 n: 通道编号(n = 0, 1), 4n = 40, 41

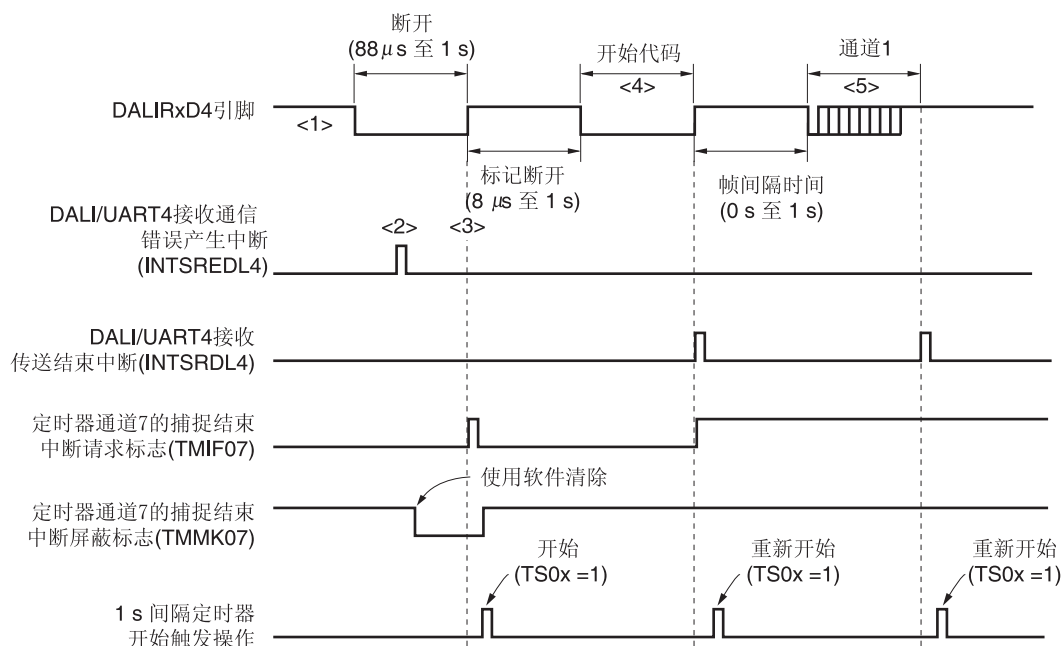
16.11 DMX512 通信操作

UART4 支持 DMX512 通信。

UART	UART4
支持DMX512通信	支持
目标通道	通道1
使用引脚	DALIRxD4
中断	INTSRDL4
	仅限传送结束中断(禁止设置缓冲器空中断。)
错误中断	INTSREDL4
错误检测标志	<ul style="list-style-type: none"> • 帧错误检测标志(FEF41) • 溢出错误检测标志(OVF41)
传送数据长度	8位
传送速率	250 [kbps]
数据相位	正相输出 (默认: 高电平)
奇偶校验位	无奇偶校验位(不检查奇偶校验位。)
停止位	附加2位
数据方向	MSB优先或LSB优先

DMX512 的接收操作的概况如图 16-67 所示。

图 16-67. DMX512 的接收操作



为了在 DALIUART4 接收 DMX512，需要结合使用 TAU 的间隔时钟/脉冲宽度测量功能。

具体方法是，使用 TAU 的定时器输入(TI07)来测量 Break 宽度，在 TAU 的一个单独通道创造 1 秒的间隔时钟，并且测量 MarkBreak 脉冲是否满足标准。

<接收前的设置>

- 在 DALIUART4 UART 模式下设置为初始设置 250 kbps。
- 将 TI07 设置为低宽度的脉冲宽度测量功能。
- 将 TAU (任意通道)设置为 1 秒间隔定时器。

<接收步骤>

状态<1> (待机状态)

- 低电平宽度测量定时器的中断屏蔽(TMMK07 = 1)
- DALIUART4 = 允许接收，不屏蔽中断(SREDLMK4 = 0, SRDLMK4 = 0)
- 1 秒间隔定时器 = 停止

状态<2> (Break)

- 由于 Break 信号而产生 INTSREDL4 中断
- 清除 DALIUART4 帧错误
- 通过软件来取消低电平宽度测量定时器的中断屏蔽(TMMK07 = 0)

状态<3> (Break 结束)

- 由于上升信号而产生低电平宽度测量定时器的 INTTM07 中断
- 计算捕捉到的 Break 长度。
- 如果不符合标准，则跳转至状态(1)
- 如果 Break 宽度符合标准，则激活 1 秒间隔定时器(TS0x = 1)并跳转至状态<4>。

状态<4> (接收 StartCode)

- 由于接收开始代码而产生 INTSRDL4 中断
- 检查数据：如果 1 秒数据不为 0，则跳转至状态<1>。如果发生接收错误(INTSREDL4)，则清除错误并跳转至状态(1)。
- 清除并启动 1 秒间隔定时器(TS0x = 1)

状态<5> (接收 Slotx)

- 由于执行插槽接收而产生 INTSRDL4 中断 -> 检查数据
- 如果发生接收错误(INTSREDL4)，则清除错误并跳转至状态(1)。
- 清除并启动 1 秒间隔定时器(TS0x = 1)

重复状态<5>，直到没有可接收数据

状态<6> (发生 MarkBreak, TimeBetweenFrame 错误时)

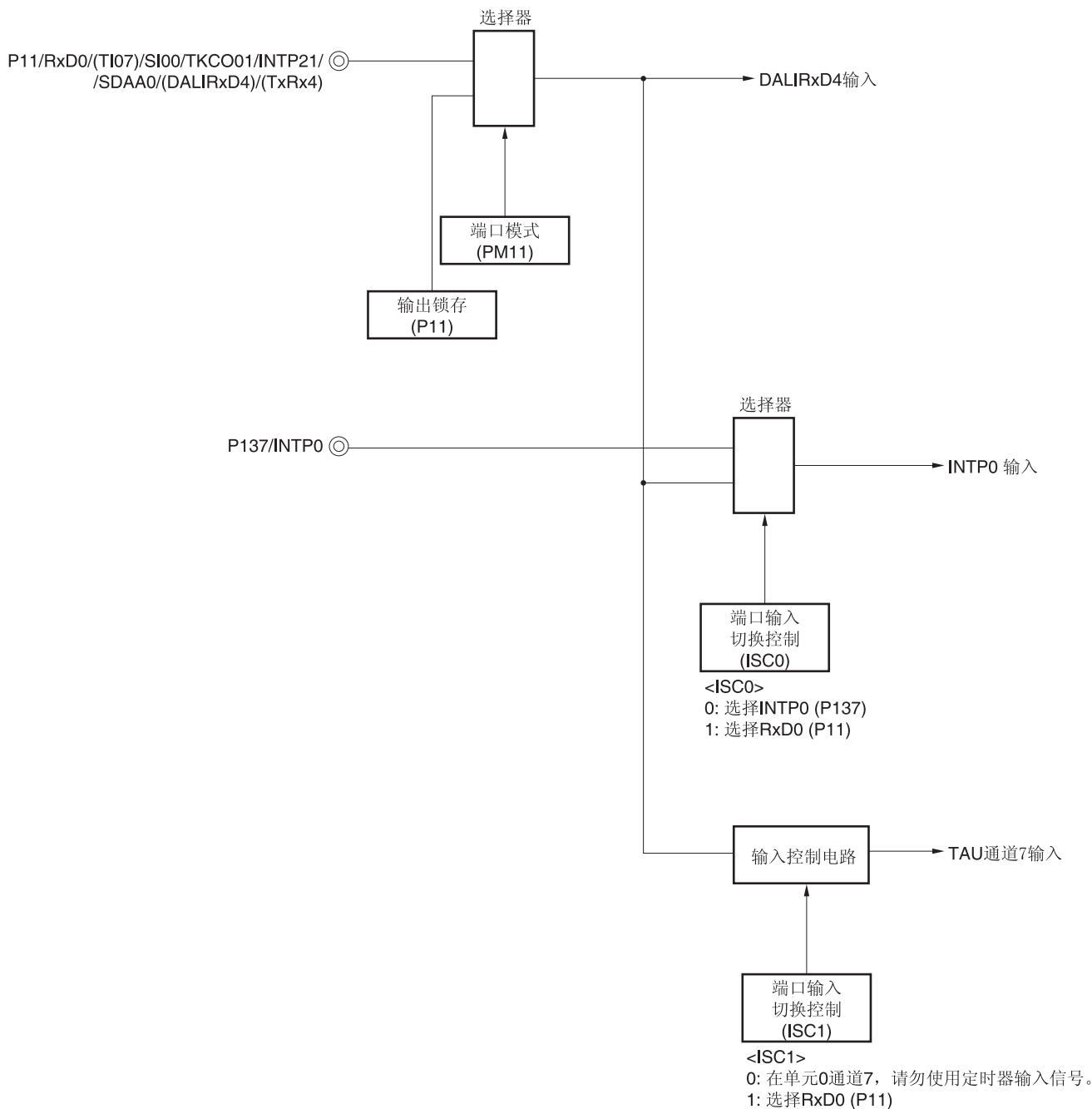
- 产生 1 秒间隔定时器中断(INTTM0x)
- 跳转至状态<1>

用于操作 DMX512 接收的端口的配置如图 16-68 所示。

通过检测外部中断(INTP0)的边沿来接收从 DMX512 主发送的 BREAK 信号。可以用定时器阵列单元的外部事件捕捉操作来测量从主发送的输入信号宽度。

可以将接收用端口输入(DALIRxD0)的输入源输入至外部中断引脚(INTP0)和定时器阵列单元。

图 16-68. DMX512 的接收操作作用端口配置



备注 ISC0, ISC1: 输入切换控制寄存器(ISC)的位 0 和 1 (参阅图 15-18。)

用于 DMX512 通信操作的外围功能如下所示。

<所用外围功能>

- 外部中断 (INTP0); BREAK 信号检测
用途: 检测 BREAK 信号的边沿和通信的开始
- 定时器阵列单元的通道 7; 输入信号宽度检测
用途: 测量输入信号宽度 (在捕捉模式下测量 DALIRxD4 输入边沿的间隔。)
- 串行阵列单元 04(SAU)的通道 0 和 1(UART0)

第十七章 串行接口 IICA

注意事项 本章的下述内容大多是以 38 引脚产品为例进行说明的。

17.1 串行接口IICA的功能

串行接口 IICA 具有以下三种模式。

(1) 操作停止模式

不执行串行传输时使用该模式。可以降低功耗。

(2) I²C 总线模式 (支持多主)

该模式使用两条线路（串行时钟 (SCLA0) 线路和串行数据总线 (SDAA0) 线路）与多个设备之间进行 8 位数据传送。该模式符合 I²C 总线格式，主设备设备可以产生并通过串行数据总线向从设备发送“开始条件”、“地址”、“传送方向指示”、“数据”和“停止条件”数据。从设备可以通过硬件自动检测出这些接收到的状态和数据。这个功能可以简化应用程序中控制 I²C 总线的部分。

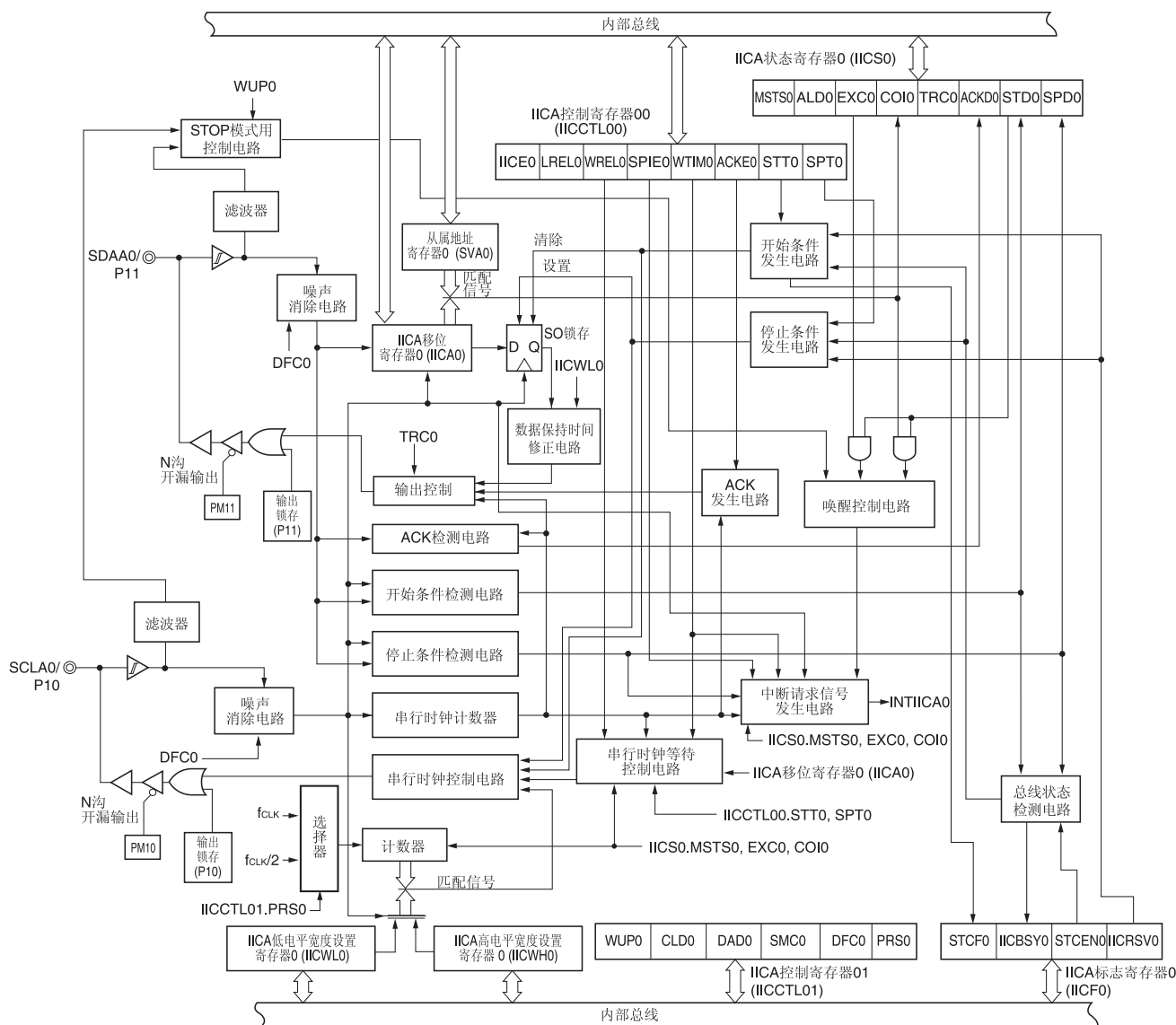
由于 SCLA0 和 SDAA0 引脚采用 N 沟开漏输出，因此串行接口 IICA 中，串行时钟线和串行数据总线上须连接上拉电阻。

(3) 唤醒模式

当在 STOP 模式下收到来自自主设备的扩展代码或者本地地址时，可以通过产生中断请求信号(INTIICA0) 来解除 STOP 模式。这可以用 IICA 控制寄存器 01 (IICCTL01)的 WUP0 位来设置。

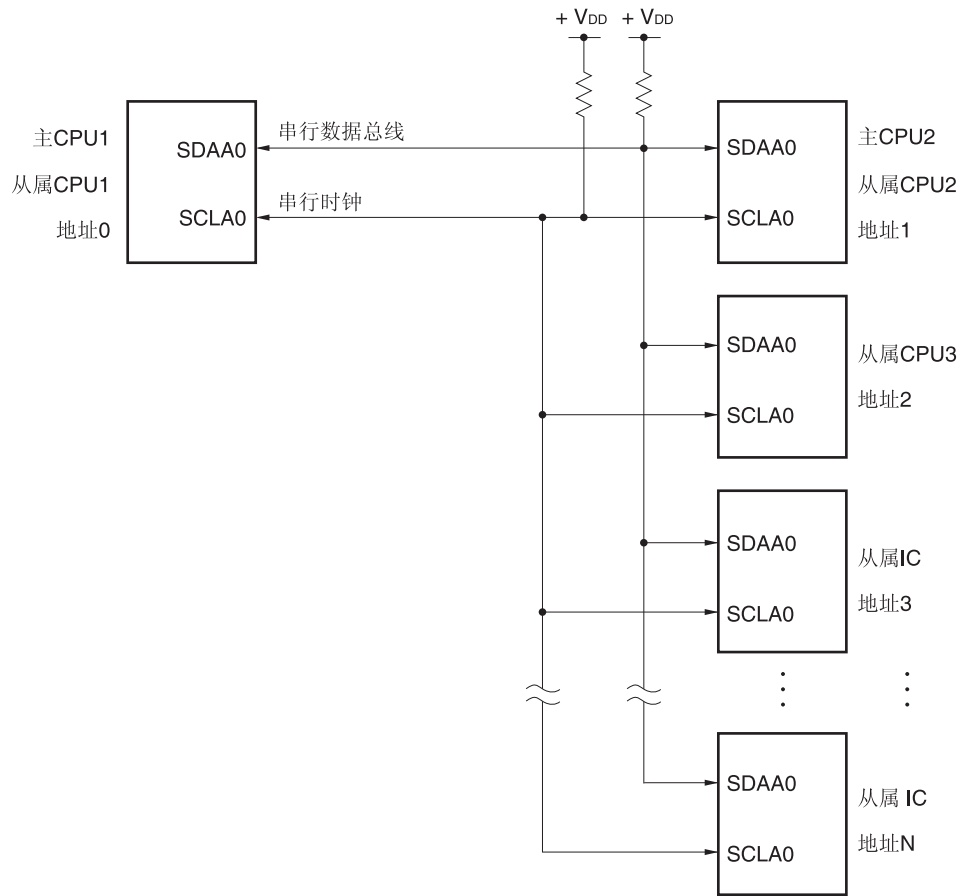
串行接口 IICA 的框图如图 17-1 所示。

图 17-1. 串行接口 IICA 的框图



串行总线配置示例如图 17-2 所示。

图 17-2. 使用 I²C 总线的串行总线配置示例



17.2 串行接口IICA的配置

串行接口 IICA 具包含以下硬件。

表 17-1. 串行接口 IICA 的配置

项目	配置
寄存器	IICA 移位寄存器 0 (IICA0) 从设备地址寄存器 0 (SVA0)
控制寄存器	外围允许寄存器 0 (PER0) IICA 控制寄存器 00 (IICCTL00) IICA 状态寄存器 0 (IICS0) IICA 标志寄存器 0 (IICF0) IICA 控制寄存器 01 (IICCTL01) IICA 低电平宽度设置寄存器 0 (IICWLO) IICA 高电平宽度设置寄存器 0 (IICWHO) 端口模式寄存器 1 (PM1) 端口输出模式寄存器 1 (POM1) 端口寄存器 1 (P1)

(1) IICA 移位寄存器 0 (IICA0)

IICA0 寄存器使用串行同步时钟，将接收的 8 位串行数据转换为 8 位并行数据，并将发送的 8 位并行数据转换为 8 位串行数据。IICA0 寄存器可以用于发送和接收。

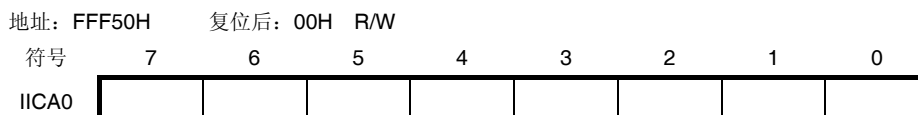
可以通过写入和读取 IICA0 寄存器来控制实际的发送和接收操作。

在等待期间，通过向 IICA0 寄存器写入数据以解除等待，并开始传送数据。

使用 8 位存储器操作指令设置 IICA0 寄存器。

产生复位信号后，IICA0 寄存器被清除为 00H。

图 17-3. IICA 移位寄存器 0 (IICA0)的格式



- 注意事项**
- 不要在数据传送过程中向 IICA0 寄存器写入数据。
 - 只能在等待期间写入或读取 IICA0 寄存器。除了等待期间，还禁止在通信状态下存取 IICA0 寄存器。但是，当器件用作主设备时，只能在通信触发位(STT0)被设置为 1 之后写入 IICA0 寄存器一次。
 - 当预约通信时，在检测到由停止条件触发的中断之后将数据写入 IICA0 寄存器。

(2) 从设备地址寄存器 0 (SVA0)

该寄存器在从设备模式下存储 7 位本地地址: {A6, A5, A4, A3, A2, A1, A0}。

使用 8 位存储器操作指令设置 SVA0 寄存器。

但是, 当 $STD0 = 1$ (即检测到开始条件) 时, 禁止改写该寄存器。

产生复位信号后, SVA0 寄存器被清除为 00H。

图 17-4. 从设备地址寄存器 0 (SVA0)的格式

地址: F0234H	复位后: 00H	R/W						
符号	7	6	5	4	3	2	1	0
SVA0	A6	A5	A4	A3	A2	A1	A0	0 ^注

注 位 0 固定为 0。

(3) SO 锁存器

SO 锁存器用于保持 SDAA0 引脚的输出电平。

(4) 唤醒模式

当该寄存器接收到的地址与设置至从设备地址寄存器 0 (SVA0)的地址值相匹配时, 或者当接收到扩展代码时, 该电路将产生一个中断请求 (INTIICA0)。

(5) 串行时钟计数器

该计数器计算发送/接收操作期间输出/输入的串行时钟, 并验证 8 位数据已被发送或接收是否完成。

(6) 中断请求信号发生电路

该电路控制产生中断请求信号(INTIICA0)。

I²C 中断请求由以下两种触发产生。

- 串行时钟的第 8 个或第 9 个时钟的下降沿 (由 WTIMO 位设置)
- 当检测到停止条件时产生中断请求 (由 SPIE0 位设置)

备注 WTIMO 位: IICA 控制寄存器 00 (IICCTL00)的位 3

SPIE0 位: IICA 控制寄存器 00 (IICCTL00)的位 4

(7) 串行时钟控制电路

在主设备模式下, 该电路从采样时钟生成串行时钟并经由 SCLA0 引脚输出。

(8) 串行时钟等待控制电路

该电路控制等待时序。

(9) ACK 生成电路, 停止条件检测电路, 开始条件检测电路以及 ACK 检测电路

这些电路用来生成和检测各种状态。

(10) 数据保持时间修正电路

此电路生成串行时钟下降沿对应的数据保持时间。

(11) 开始条件生成电路

此电路会在 STT0 位被设置为 1 时生成开始条件。

但是, 在通信预约功能禁止状态下 (IICRSV 位 = 1), 且总线未释放 (IICBSY 位 = 1)时, 开始条件请求会被忽略, 并且将 STCF 位设为 1。

(12) 停止条件生成电路

此电路会在 SPT0 位被设置为 1 时生成停止条件。

(13) 总线状态检测电路

该电路通过检测开始和停止条件来检测总线是否被释放。

但是，由于操作之后不能立即检测出总线状态，需使用 STCEN 位设置总线状态检测电路的初始状态。

备注 STT0 位: IICA 控制寄存器 00 (IICCTL00)的位 1
SPT0 位: IICA 控制寄存器 00 (IICCTL00)的位 0
IICRSV0 位: IICA 标志寄存器 0 (IICF0)的位 0
IICBSY0 位: IICA 标志寄存器 0 (IICF0)的位 6
STCF0 位: IICA 标志寄存器 0 (IICF0)的位 7
STCEN0 位: IICA 标志寄存器 0 (IICF0)的位 1

17.3 控制串行接口IICA的寄存器

串行接口 IICA 由以下十种寄存器控制。

- 外围允许寄存器 0 (PER0)
- IICA 控制寄存器 00 (IICCTL00)
- IICA 标志寄存器 0 (IICF0)
- IICA 状态寄存器 0 (IICS0)
- IICA 控制寄存器 01 (IICCTL01)
- IICA 低电平宽度设置寄存器 0 (IICWLO)
- IICA 高电平宽度设置寄存器 0 (IICWHO)
- 端口模式寄存器 1 (PM1)
- 端口输出模式寄存器 1 (POM1)
- 端口寄存器 1 (P1)

(1) 外围允许寄存器 0 (PER0)

该寄存器用于允许或禁止向外围硬件供应时钟。并且中止向未使用的硬件供应时钟，以减少电力消耗和噪声。

当使用串行接口 IICA 时，必须将该寄存器的位 4 (IICA0EN) 设置为 1。

使用 1 位或 8 位存储器操作指令设置 PER0 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 17-5. 外围允许寄存器 0 (PER0) 的格式

地址: F00F0H 复位后: 00H R/W

符号	<7>	6	<5>	<4>	3	<2>	1	<0>
PER0	RTCEN	0	ADCEN	IICA0EN	0	SAU0EN	0	TAU0EN

IICA0EN	串行接口 IICA 输入时钟供应的控制
0	停止输入时钟供应。 <ul style="list-style-type: none"> • 不可以写入串行接口 IICA 所使用的 SFR。 • 串行接口 IICA 处于复位状态。
1	允许输入时钟供应。 <ul style="list-style-type: none"> • 可以读取/写入串行接口 IICA 所使用的 SFR。

注意事项 1. 要设置串行接口 IICA，必须首先将 IICA0EN 位设置为 1。如果 IICA0EN = 0，将忽略对串行接口 IICA 控制寄存器进行的写操作，而且寄存器的读取值均为初始值（端口模式寄存器 1 (PM1)、端口输出模式寄存器 1 (POM1) 和端口寄存器 1 (P1) 除外）。

2. 必须将位 1、3 和 6 清除为 0。

(2) IICA 控制寄存器 00 (IICCTL00)

该寄存器用于允许/停止 I²C 操作、设置等待时间和设置其他 I²C 操作。

使用 1 位或 8 位存储器操作指令设置 IICCTL00 寄存器。但是，在 IICE0 = 0 时或者等待期间，可以设置 SPIE0、WTIM0 和 ACKE0 位。可以在将 IICE0 位从 0 设置为 1 时，同时设置这些位。

产生复位信号后，该寄存器被清除为 00H。

图 17-6. IICA 控制寄存器 00 (IICCTL00)的格式 (1/4)

地址: F0230H 复位后: 00H R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IICCTL00	IICE0	LRELO	WRELO	SPIE0	WTIM0	ACKE0	STT0	SPT0

IICE0	允许 I ² C 操作
0	停止操作。将 IICA 状态寄存器 0 (IICS0)复位 ^{注1} 。同时停止内部操作。
1	允许操作。
SCLA0 和 SDAA0 线路是高电平时，请确保将该位设置为 1。	
清除条件 (IICE0 = 0)	设置条件 (IICE0 = 1)
<ul style="list-style-type: none"> 由指令清除 复位 	<ul style="list-style-type: none"> 由指令设置

LRELO ^{注2,3}	退出通信
0	正常操作
1	<p>从当前通信退出，并设置为待机模式。执行后该位自动清除为 0。</p> <p>当接收到本地无关扩展码等情况下使用。</p> <p>SCLA0 和 SDAA0 线路被设置为高阻状态。</p> <p>IICA 控制寄存器 00 (IICCTL00)和 IICA 状态寄存器 0 (IICS0)的下列标志被清除为 0。</p> <p>• STT0 • SPT0 • MSTS0 • EXC0 • COI0 • TRC0 • ACKD0 • STD0</p>
从通信中退出之后的待机状态一直保持有效，直到满足下面的通信参加条件。	
<ul style="list-style-type: none"> 检测到停止条件后，在主设备模式下重新开始。 在检测到开始条件以后，发生地址匹配或扩展代码接收。 	
清除条件(LRELO = 0)	设置条件(LRELO = 1)
<ul style="list-style-type: none"> 执行后自动清除。 复位 	<ul style="list-style-type: none"> 由指令设置

WRELO ^{注2,3}	取消等待
0	不取消等待
1	取消等待。等待取消以后，该设置被自动清除。
发送模式下 (TRC0 = 1)，在等待第九个时钟脉冲的过程中设置 WRELO 位（取消等待）时，SDAA0 线路将进入高阻抗状态 (TRC0 = 0)。	
清除条件(WRELO = 0)	设置条件(WRELO = 1)
<ul style="list-style-type: none"> 执行后自动清除。 复位 	<ul style="list-style-type: none"> 由指令设置

注 1. IICA 状态寄存器 0 (IICS0)、IICA 标志寄存器 0 (IICF0)的 STCF 和 IICBSY 位以及 IICA 控制寄存器 01 (IICCTL01)的 CLD0 和 DAD0 位复位。

- 当 IICE0 为 0 时，该位的信号无效。
- LRELO 和 WRELO 位的读取值始终为 0。

注意事项 如果在 SCLA0 线路为高电平、SDAA0 线路为低电平且开启数字滤波器 (IICCTL01 寄存器的 DFC0 位 = 1) 时允许 I²C 操作(IICE0 = 1)，则会立即检测到开始条件。这种情况下，在允许 I²C 操作(IICE0 = 1)之后，接着通过 1 位存储器操作指令，将 LRELO 位设置为 1。

图 17-6. IICA 控制寄存器 00 (IICCTL00)的格式 (2/4)

SPIE0 ^{注1}	当检测到停止条件时，允许/禁止产生中断请求	
0	禁止	
1	允许	
如果 IICA 控制寄存器 01 (IICCTL01)的 WUP0 位为 1，即使 SPIE0 = 1，也不会产生停止条件中断。		
清除条件(SPIE0 = 0)		设置条件(SPIE0 = 1)
<ul style="list-style-type: none"> 由指令清除 复位 		<ul style="list-style-type: none"> 由指令设置

WTIMO ^{注1}	控制产生等待和中断请求	
0	在第 8 个时钟的下降沿产生中断请求信号。 主模式： 在输出 8 个时钟以后，时钟输出被设置为低电平并等待。 从模式： 在输入 8 个时钟以后，时钟被设置为低电平并等待主设备。	
1	在第 9 个时钟的下降沿产生中断请求信号。 主模式： 在输出 9 个时钟以后，时钟输出被设置为低电平并等待。 从模式： 在输入 9 个时钟以后，时钟被设置为低电平并等待主设备。	
地址传送期间，不论该位如何设置，均将在第 9 个时钟的下降沿处产生中断。地址传送完成后该位的设置有效。主设备模式下，在地址传送期间的第 9 个时钟下降沿处，插入一个等待。对于接收到本地地址的从设备，会在执行应答(ACK)之后的第 9 个时钟下降沿处插入一个等待。但是，当从设备已经接收到扩展代码时，则会在第 8 个时钟下降沿处插入一个等待。		
清除条件(WTIMO = 0)		设置条件(WTIMO = 1)
<ul style="list-style-type: none"> 由指令清除 复位 		<ul style="list-style-type: none"> 由指令设置

ACKE0 ^{注1,2}	应答控制	
0	禁止应答	
1	允许应答。在第 9 个时钟期间，SDAA0 线路设置为低电平。	
清除条件(ACKE0 = 0)		设置条件(ACKE0 = 1)
<ul style="list-style-type: none"> 由指令清除 复位 		<ul style="list-style-type: none"> 由指令设置

- 注 1. 当 IICE0 为 0 时，该位的信号无效。在这个期间设置该位。
2. 在地址传送期间且代码不是扩展代码时，设置值无效。
当器件为从设备且地址匹配时，不论设置值如何均产生应答。

图 17-6. IICA 控制寄存器 00 (IICCTL00)的格式 (3/4)

STT0 ^注	开始条件触发	
0	不发送开始条件。	
1	当总线被释放 (待机状态下, 当 IICBSY = 0): 当这个位被设置为 1, 产生一个开始条件 (作为主设备启动)。 当第三方正在通信时: <ul style="list-style-type: none"> • 当允许通信预约功能 (IICRSV = 0) 用作开始条件预约标志。设置为 1 时, 会在释放总线后自动发送开始条件。 • 当禁止通信预约功能 (IICRSV = 1) 即使该位被设置为 1, 也将清除 STT0 位, 并将 STT0 清除标志(STCF)设置为 1。不发送开始条件。 处于等待状态 (作为主设备): 解除等待后产生重启条件。	
设置时序的有关注意事项: <ul style="list-style-type: none"> • 主设备接收: 传输期间不能设置为 1。只能在 ACKE0 位已经被清除为 0, 且从设备已经完成接收时, 才可在等待期间设置为 1。 • 主设备发送: 在应答期间, 通常不能发送开始条件。在第 9 个时钟输出后的等待期间设置为 1。 • 不能与停止条件触发(SPT0)同时设置为 1。 • 一旦将 STT0 设置为 1, 则不允许在清除条件之前再次将其设置为 1。 		
清除条件(STT0 = 0)		设置条件(STT0 = 1)
<ul style="list-style-type: none"> • 在禁止通信预约时, 通过将 STT0 位设置为 1 来进行清除。 • 仲裁失败后清除。 • 主设备发送开始条件后清除。 • 由 LREL0 = 1 清除 (退出通信) • 当 IICE0 = 0 (停止操作) • 复位 		<ul style="list-style-type: none"> • 由指令设置

注 当 IICE0 为 0 时, 该位的信号无效。

- 备注 1. 设置数据之后读取时, 位 1 (STT0) 为 0。
 2. IICRSV0: IIC 标志寄存器 0 (IICF0)的位 0
 STCF0: IIC 标志寄存器 0 (IICF0)的位 7

图 17-6. IICA 控制寄存器 00 (IICCTL00)的格式 (4/4)

SPT0	停止条件触发
0	不产生停止条件。
1	产生停止条件（作为主设备的传输结束）。
设置时序的有关注意事项： <ul style="list-style-type: none"> 主接收：传输期间不能设置为 1。 只能在 ACKE0 位已经清除为 0，且从设备已经完成接收时，才可在等待期间设置为 1。 主发送：在应答期间，通常不能产生停止条件。在第 9 个时钟输出后的等待期间对其进行设置。 不能与开始条件触发(STT0)同时设置为 1。 只有在主设备模式下，才可将 SPT0 位设置为 1。 在 WTIMO 位已经清除为 0 时，如果在 8 个时钟输出后的等待期间将 SPT0 位设置为 1，注意，将在第 9 个时钟的高电平期间产生停止条件。在 8 个时钟输出后的等待期间，WTIMO 位应该从 0 改变为 1，且在第 9 个时钟输出后的等待期间应该将 SPT0 位设置为 1。 一旦将 SPT0 设置为 1，则不允许在清除条件之前再次将其设置为 1。 	
清除条件(SPT0 = 0)	设置条件(SPT0 = 1)
<ul style="list-style-type: none"> 仲裁失败后清除。 检测到停止条件后，自动清除。 由 LRELO = 1 清除 (退出通信) 当 IICE0 = 0 (停止操作) 复位 	<ul style="list-style-type: none"> 由指令设置

注意事项 当 IICA 状态寄存器 0 (IICS0)的位 3 (TRC0)被设置为 1（传输状态）时，IICA 控制寄存器 00 (IICCTL00)的位 5 (WRELO)将在第 9 个时钟被设置为 1 并取消等待，之后，TRC0 位被清除（接收状态），SDAA0 线路设置为高阻抗。通过写入 IICA 移位寄存器 0 来解除当 TRC0 位为 1（发送状态）时执行的等待。

备注 设置数据之后读取时，位 0 (SPT0) 为 0。

(3) IICA 状态寄存器 0 (IICS0)

该寄存器指示 I²C 的状态。

仅当 STT0 = 1 且处于等待期间时，可以使用 1 位或 8 位的存储器操作指令读取 IICS0 寄存器。

产生复位信号后，该寄存器被清除为 00H。

注意事项 当允许在 STOP 模式下允许地址匹配唤醒功能 (WUP0 = 1) 时，禁止读取 IICS0 寄存器。当 WUP0 位从 1 改变为 0 (停止唤醒操作) 时，无论是否有 INTIICA0 中断请求，检测到下一个开始条件或停止条件时才会更新状态的变化。因此，要使用唤醒功能，须允许 (SPIE0 = 1) 检测到停止条件时产生中断，并在检测到中断之后读取 IICS0 寄存器。

备注 STT0: IICA 控制寄存器 00 (IICCTL00) 的位 1

WUP0: IICA 控制寄存器 01 (IICCTL01) 的位 7

图 17-7. IICA 状态寄存器 0 (IICS0) 的格式 (1/3)

地址: FFF51H 复位后: 00H R

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IICS0	MSTS0	ALD0	EXC0	COI0	TRC0	ACKD0	STD0	SPD0

MSTS0	主设备状态检查标志	
0	从状态或通信待机状态	
1	主设备通信状态	
清除条件(MSTS0 = 0)		设置条件(MSTS0 = 1)
<ul style="list-style-type: none"> 当检测到停止条件时 当 ALD0 = 1 (仲裁失败) 由 LREL0 = 1 清除 (退出通信) 当 IICE0 位由 1 变为 0 (停止操作) 复位 		<ul style="list-style-type: none"> 当发送开始条件时
ALD0	仲裁失败的检测	
0	该状态表示不存在总线仲裁，或仲裁结果为“成功”。	
1	该状态表示仲裁结果为“失败”。MSTS0 位被清除。	
清除条件(ALD0 = 0)		设置条件(ALD0 = 1)
<ul style="list-style-type: none"> IICS0 寄存器被读取后自动清除^注 当 IICE0 位由 1 变为 0 (停止操作) 复位 		<ul style="list-style-type: none"> 当仲裁结果为“失败”。

注 对于 IICS0 寄存器的其他位执行 1 位存储器操作指令时该寄存器也被清除。因此，使用 ALD0 位时，要在读取其他位的数据之前读取该位的数据。

备注 LREL0: IICA 控制寄存器 00 (IICCTL00) 的位 6

IICE0: IICA 控制寄存器 00 (IICCTL00) 的位 7

图 17-7. IICA 状态寄存器 0 (IICS0)的格式 (2/3)

EXC0	扩展码接收的检测	
0	未接收到扩展码。	
1	接收到扩展码。	
清除条件(EXC0 = 0)		设置条件(EXC0 = 1)
<ul style="list-style-type: none"> 当检测到开始条件时 当检测到停止条件时 由 LRELO = 1 清除 (退出通信) 当 IICE0 位由 1 变为 0 (停止操作) 复位 		<ul style="list-style-type: none"> 当接收地址数据的高 4 位是 0000 或 1111 时 (在第 8 个时钟的上升沿设置)。
COI0	匹配地址的检测	
0	地址不匹配。	
1	地址匹配。	
清除条件(COI0 = 0)		设置条件(COI0 = 1)
<ul style="list-style-type: none"> 当检测到开始条件时 当检测到停止条件时 由 LRELO = 1 清除 (退出通信) 当 IICE0 位由 1 变为 0 (停止操作) 复位 		<ul style="list-style-type: none"> 当接收到的地址和本地地址(从设备地址寄存器 0 (SVA0))匹配时 (在第 8 个时钟的上升沿设置)。
TRC0	发送/接收状态的检测	
0	接收状态 (除发送状态外)。SDAA0 线路被设置为高阻态。	
1	发送状态。允许 SO0 锁存器内的数值输出至 SDAA0 线路 (在第一个字节的第 9 个时钟下降沿开始生效)。	
清除条件(TRC0 = 0)		设置条件(TRC0 = 1)
<主设备和从设备的双方> <ul style="list-style-type: none"> 当检测到停止条件时 由 LRELO = 1 清除 (退出通信) 当 IICE0 位由 1 变为 0 (停止操作) 由 WRELO = 1 清除 (取消等待) 当 ALD0 位由 0 变为 1 (仲裁失败) 复位 不用于通信时 (MSTS0, EXC0, COI0 = 0) <主设备> <ul style="list-style-type: none"> 当输出 1 至第一个字节的 LSB (传送方向指示位) 时。 <从设备> <ul style="list-style-type: none"> 当检测到开始条件时 将 0 输入至第一个字节的 LSB (传送方向指示位) 时。 		<主设备> <ul style="list-style-type: none"> 当产生开始条件时 当 (在地址传送期间) 将 0 (主设备发送) 输出至第一个字节的 LSB (传送方向指示位) 时 <从设备> <ul style="list-style-type: none"> 当 (在地址传送期间) 主设备将 1 (从设备发送) 输入至第一个字节的 LSB (传送方向指示位) 时

注 当 IICA 状态寄存器 0 (IICS0)的位 3 (TRC0) 被设置为 1 (传输状态) 时, IICA 控制寄存器 00 (IICCTL00)的位 5 (WRELO) 将在第 9 个时钟被设置为 1 并取消等待, 然后, TRC0 位被清除 (接收状态), SDAA0 线路设置为高阻抗。通过写入 IICA 移位寄存器 0 来解除当 TRC0 位为 1 (发送状态) 时执行的等待。

备注 LRELO: IICA 控制寄存器 00 (IICCTL00)的位 6
IICE0: IICA 控制寄存器 00 (IICCTL00)的位 7

图 17-7. IICA 状态寄存器 0 (IICS0)的格式 (3/3)

ACKD0	应答的检测(ACK)	
0	未检测到应答。	
1	检测到应答。	
清除条件(ACKD0 = 0)		设置条件(ACKD0 = 1)
<ul style="list-style-type: none"> 当检测到停止条件时 在下一字节第 1 个时钟的上升沿。 由 LREL0 = 1 清除 (退出通信) 当 IICE0 位由 1 变为 0 (停止操作) 复位 		<ul style="list-style-type: none"> 在 SCLA0 线路第 9 个时钟的上升沿将 SDAA0 线路设置为低电平后
STD0	开始条件的检测	
0	未检测到开始条件。	
1	检测到开始条件。这表示处于地址传送期间。	
清除条件(STD0 = 0)		设置条件(STD0 = 1)
<ul style="list-style-type: none"> 当检测到停止条件时 在传送地址后下一个字节第一个时钟的上升沿。 由 LREL0 = 1 清除 (退出通信) 当 IICE0 位由 1 变为 0 (停止操作) 复位 		<ul style="list-style-type: none"> 当检测到开始条件时
SPD0	停止条件的检测	
0	未检测到停止条件。	
1	检测到停止条件。终结主设备通信，并且释放总线。	
清除条件(SPD0 = 0)		设置条件(SPD0 = 1)
<ul style="list-style-type: none"> 设置该位后，在检测到开始条件时，地址传送位的第 1 个时钟上升沿处 当 WUP0 位由 1 变为 0 当 IICE0 位由 1 变为 0 (停止操作) 复位 		<ul style="list-style-type: none"> 当检测到停止条件时

备注 LREL0: IICA 控制寄存器 00 (IICCTL00)的位 6

IICE0: IICA 控制寄存器 00 (IICCTL00)的位 7

(4) IICA 标志寄存器 0 (IICF0)

该寄存器设置 I²C 的操作模式，并指示 I²C 总线的状态。

使用 1 位或 8 位存储器操作指令设置 IICF0 寄存器。其中，STT0 清除标志 0 (STCF0)和 I²C 总线状态标志 0 (IICBSY0)位为只读位。

IICRSV0 位用于允许/禁止通信预约功能。

STCEN0 位用于为 IICBSY0 位设置初始值。

只能在禁止 I²C 操作 (IICA 控制寄存器 00 (IICCTL00)的位 7 (IICE0) = 0) 时写入 IICRSV0 和 STCEN0 位。允许操作时，IICF0 寄存器可以被读取。

产生复位信号后，该寄存器被清除为 00H。

图 17-8. IICA 标志寄存器 0 (IICF0)的格式

地址: FFF52H 复位后: 00H R/W^注符号

7	6	5	4	3	2	1	0
---	---	---	---	---	---	---	---

IICF0	STCF0	IICBSY0	0	0	0	0	STCEN0	IICRSV0
-------	-------	---------	---	---	---	---	--------	---------

STCF0	STT0清除标志
0	发送开始条件。
1	不发送开始条件, 清除STT0标志。
清零条件 (STCF0 = 0)	
<ul style="list-style-type: none"> • 通过STT0 = 1清除 • 当IICE0 = 0 (停止操作) 时 • 复位时 	
置位条件 (STCF0 = 1)	
<ul style="list-style-type: none"> • 因禁止通信预约 (IICRSV0 = 1), 而不产生开始条件, 并且STT0位清零时。 	

IICBSY0	I ² C总线状态标志
0	总线解除状态 (STCEN0 = 1时, 置于通信初期状态)。
1	总线通信状态 (STCEN0 = 0时, 置于通信初期状态)。
清零条件 (IICBSY0 = 0)	
<ul style="list-style-type: none"> • 检测出停止条件时 • IICE0 = 0 (操作停止) 时 • 复位时 	
置位条件 (IICBSY0 = 1)	
<ul style="list-style-type: none"> • 检测开始条件时 • STCEN0 = 0时的IICE0位设置 	

STCEN0	初期开始允许触发
0	允许操作 (IICE0 = 1) 后, 通过检测出停止条件, 允许发送开始条件。
1	允许操作 (IICE0 = 1) 后, 不通过检测停止条件, 允许发送开始条件。
清零条件 (STCEN0 = 0)	
<ul style="list-style-type: none"> • 通过指令清零 • 检测出开始条件时 • 复位时 	
置位条件 (STCEN0 = 1)	
<ul style="list-style-type: none"> • 通过指令置位 	

IICRSV0	通信预约功能禁止位
0	允许通信预约。
1	禁止通信预约。
清零条件 (IICRSV0 = 0)	
<ul style="list-style-type: none"> • 通过指令清零 • 复位时 	
置位条件 (IICRSV0 = 1)	
<ul style="list-style-type: none"> • 通过指令置位 	

注 位 6 和位 7 为只读位。

- 注意事项
1. 只能在停止操作(IICE0 = 0)时才可以写入 STCEN0 位。
 2. 当 STCEN0 = 1 时, 不论实际总线状态如何都认为处于总线释放状态 (IICBSY0 = 0), 因此, 要产生第一个开始条件 (STT0 = 1)时, 为了避免破坏其他通信, 必须确认没有正在进行的第三方通信。
 3. 只能在停止操作(IICE0 = 0)时才可以写入 IICRSV0。

备注 STT0: IICA 控制寄存器 00 (IICCTL00)的位 1

IICE0: IICA 控制寄存器 00 (IICCTL00)的位 7

(5) IICA 控制寄存器 01 (IICCTL01)

该寄存器用于设置 I²C 的操作模式，并检测 SCLA0 和 SDA A0 引脚的状态。

使用 1 位或 8 位存储器操作指令设置 IICCTL01 寄存器。其中，CLD0 和 DAD0 位是只读位。

在禁止操作 I²C (IICA 控制寄存器 00 (IICCTL00)的位 7 (IICE0) 为 0) 时设置 IICCTL01 寄存器 (WUP0 位除外)。产生复位信号后，该寄存器被清除为 00H。

图 17-9. IICA 控制寄存器 01 (IICCTL01)的格式 (1/2)

地址: F0231H 复位后: 00H R/W ^{注1}

符号	7	6	<5>	<4>	<3>	<2>	1	<0>
IICCTL01	WUP0	0	CLD0	DAD0	SMC0	DFC0	0	PRS0

WUP0	控制地址匹配唤醒
0	停止 STOP 模式下地址匹配唤醒功能的操作。
1	允许 STOP 模式下地址匹配唤醒功能的操作。
<p>要在 WUP0 = 1 时切换至 STOP 模式，则须在将 WUP0 位设置为 1 的至少三个时钟周期之后执行 STOP 指令(参阅图 17-23. 设置 WUP0 = 1 的流程)。</p> <p>在地址匹配或者收到扩展代码之后，将 WUP0 位清除为 0。可以通过将 WUP0 位清除为 0 进入后续通信。(必须在 WUP0 位清除为 0 之后才能解除等待以及写入发送数据。)</p> <p>在 WUP0 = 1 时，当地址匹配或者收到扩展代码时，产生中断的时序与 WUP0 = 0 时产生中断的时序相同。(将发生时钟采样差异延迟。)另外，当 WUP0 = 1 时，即使 SPIE0 位被设置为 1，也不产生停止条件中断。</p> <p>当通过串行接口 IICA 的中断以外的源设置 WUP0 = 0 时，在检测到后续开始条件或停止条件之前，不能执行主设备操作。如果不等待检测到后续开始条件或停止条件，则不得通过设置 STT0 位为(1) 来输出开始条件。</p>	
清除条件(WUP0 = 0)	设置条件(WUP0 = 1)
<ul style="list-style-type: none"> 由指令清除 (在地址匹配或接收扩展码后) 	<ul style="list-style-type: none"> 由指令设置 (当 MSTS0、EXC0 和 COI0 位为 0，而且 STD0 位也为 0 (不参与通信))^{注2}

注1. 位 4 和位 5 为只读位。

2. 必须在如下所示的期间内确认 IICA 状态寄存器 0 (IICS0)的状态并设置 WUP0 位。

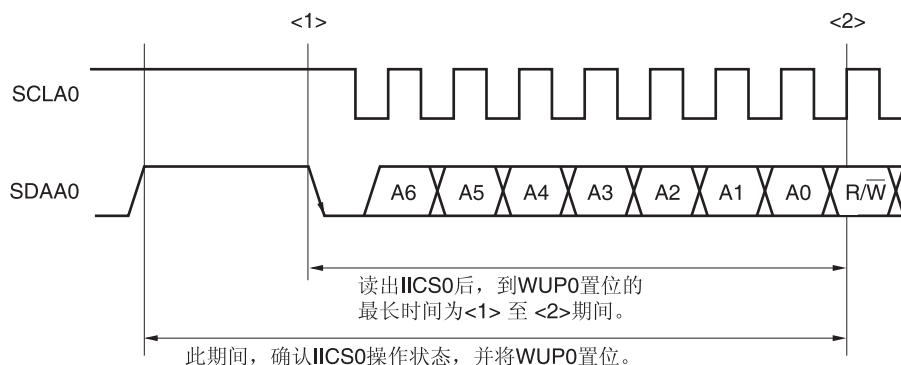


图 17-9. IICA 控制寄存器 01 (IICCTL01)的格式 (2/2)

CLD0	SCLA0 引脚电平的检测 (仅限 IICE0 = 1 时有效)	
0	SCLA0 引脚检测为低电平。	
1	SCLA0 引脚检测为高电平。	
清除条件(CLD0 = 0)		设置条件(CLD0 = 1)
<ul style="list-style-type: none"> 当 SCLA0 引脚处于低电平 当 IICE0 = 0 (停止操作) 复位 		<ul style="list-style-type: none"> 当 SCLA0 引脚处于高电平
DAD0	SDAA0 引脚电平的检测 (仅限 IICE0 = 1 时有效)	
0	SDAA0 引脚检测为低电平。	
1	SDAA0 引脚检测为高电平。	
清除条件(DAD0 = 0)		设置条件(DAD0 = 1)
<ul style="list-style-type: none"> 当 SDAA0 引脚处于低电平 当 IICE0 = 0 (停止操作) 复位 		<ul style="list-style-type: none"> 当 SDAA0 引脚处于高电平
SMC0	操作模式切换	
0	在标准模式下操作 (最大传送速率: 100 kbps)	
1	在快速模式下操作 (最大传送速率: 400 kbps)	
DFC0	数字滤波器操作控制	
0	数字滤波器 OFF。	
1	数字滤波器 ON。	
<p>仅限快速模式下可以使用数字滤波器。 在快速模式下, 无论 DFC0 位设置为 1 还是清除为 0, 传送时钟不变。 数字滤波器用于消除快速模式下的噪声。</p>		
PRS0	操作时钟的分频	
0	选择 f _{CLK} 作为操作时钟。	
1	选择 f _{CLK} /2 作为操作时钟。	

注意事项 串行接口 IICA 操作时钟的最大工作频率为 20 MHz(Max.)。如果 f_{CLK} 超过 20 MHz, 则通过将 PRS0 位设置为 1 以设置时钟为 f_{CLK} /2。

备注 IICE0: IICA 控制寄存器 00 (IICCTL00)的位 7

(6) IICA 低电平宽度设置寄存器 0 (IICWLO)

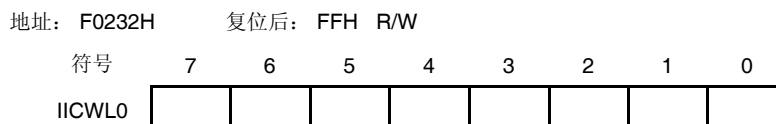
该寄存器用于设置串行接口 IICA 输出的 SCLA0 引脚信号的低电平宽度 (tLow)。

使用 8 位存储器操作指令设置 IICWLO 寄存器。

禁止操作 I²C (IICA 控制寄存器 00 (IICCTL00)的位 7 (IICE0) 为 0) 后设置 IICWLO 寄存器。

产生复位信号后, 该寄存器被设置为 FFH。

图 17-10. IICA 低电平宽度设置寄存器 0 (IICWLO)的格式

**(7) IICA 高电平宽度设置寄存器 0 (IICWHO)**

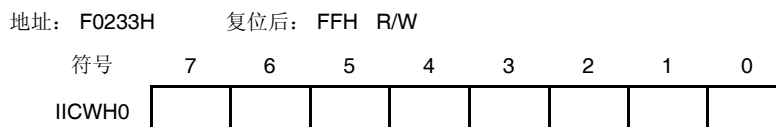
该寄存器用于设置串行接口 IICA 输出的 SCLA0 引脚信号的高电平宽度。

使用 8 位存储器操作指令设置 IICWHO 寄存器。

禁止操作 I²C (IICA 控制寄存器 00 (IICCTL00)的位 7 (IICE0) 为 0) 后设置 IICWHO 寄存器。

产生复位信号后, 该寄存器被设置为 FFH。

图 17-11. IICA 高电平宽度设置寄存器 0 (IICWHO)的格式



备注 关于利用 IICWLO 和 IICWHO 寄存器设置传送时钟的详情, 请参阅 17.4.2 利用 IICWLO 和 IICWHO 寄存器设置传送时钟。

(8) 端口模式寄存器 1 (PM1)

该寄存器以 1 位为单位设置端口 1 的输入/输出。

当使用 P10/SCLA0/SO00/TxD0/TKCO00/INTP20/(DALITxD4)引脚作为时钟输入/输出而且使用 P11/SDAA0/SI00/RxD0/TKCO01/INTP21/(TI07)/(DALIRxD4)/(TxRx4)引脚作为串行数据输入/输出时，将 PM10 和 PM11 以及 P10 和 P11 的输出锁存器清除为 0。

在设置输出模式前，将 IICE0 位（IICA 控制寄存器 00 (IICCTL00)的位 7）设置为 1，因为

P10/SCLA0/SO00/TxD0/TKCO00/INTP20/(DALITxD4)和

P11/SDAA0/SI00/RxD0/TKCO01/INTP21/(TI07)/(DALIRxD4)/(TxRx4)引脚在 IICE0 位为 0 时输出低电平（固定）。

使用 1 位或 8 位存储器操作指令设置 PM1 寄存器。

产生复位信号后，该寄存器被设置为 FFH。

图 17-12. 端口模式寄存器 1 (PM1)的格式

地址: FFF21H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM1	1	1	1	1	1	PM12	PM11	PM10

PM1n	P1n 引脚的输出缓冲器的选择(n = 0 至 2)
0	输出模式 (输出缓冲器 ON)
1	输入模式 (输出缓冲器 OFF)

注意事项 必须将 PM1 寄存器的位 3 至位 7 设置为 1。

对于 32、30 和 20 引脚产品，在解除复位后，必须通过软件将 PM1 寄存器的位 2 设置为输出模式 (通过设置端口寄存器和端口模式寄存器为 0 来指定)。

(9) 端口输出模式寄存器 1 (POM1)

该寄存器以 1 位为单位设置 P10 至 P12 的输出模式。

I²CA 通信时，P11/SDAA0/SI00/RxD0/TKCO01/INTP21/(TI07)/(DALIRxD4)/(TxRx4)引脚可以选择 N 沟开漏输出 (V_{DD} 耐压)模式。

使用 1 位或 8 位存储器操作指令来设置 POM1 寄存器。

产生复位信号后，POM1 寄存器被清除为 00H。

图 17-13. 端口输出模式寄存器 1 (POM1)的格式

地址: F0051H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
POM1	0	0	0	0	0	POM12	POM11	POM10

POM1n	P1n 引脚的输出缓冲器的选择(n = 0 至 2)
0	正常输出模式
1	N 沟开漏输出 (V _{DD} 耐压)模式

17.4 I²C总线模式功能

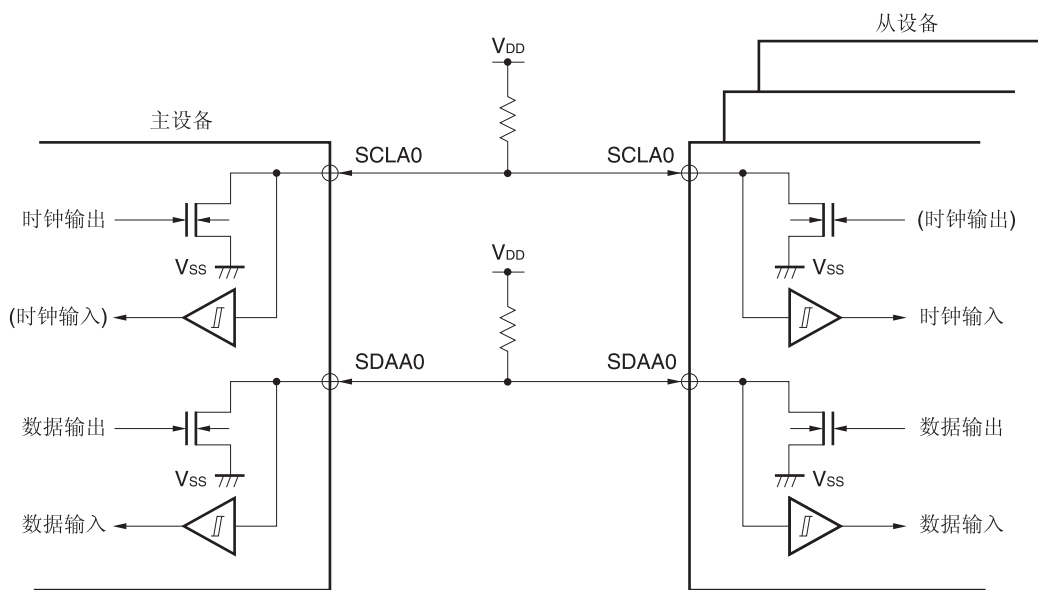
17.4.1 引脚配置

串行时钟引脚 (SCLA0)和串行数据总线引脚 (SDAA0)的配置如下所示。

- (1) SCLA0 .该引脚用于串行时钟输入和输出。
对于主设备和从设备，该引脚都是 N 沟开漏输出。输入为施密特输入。
- (2) SDAA0..该引脚用于串行数据输入和输出。
对于主设备和从设备，该引脚都是 N 沟开漏输出。输入为施密特输入。

由于串行时钟线路和串行数据总线输出都是 N 沟开漏输出，所以都需要外接上拉电阻。

图 17-14. 引脚配置图



17.4.2 利用IICWLO 和IICWH0 寄存器设置传送时钟

(1) 设置主设备方的传送时钟

$$\text{传送时钟} = \frac{f_{\text{CLK}}}{\text{IICWLO} + \text{IICWH0} + f_{\text{CLK}}(t_{\text{R}} + t_{\text{F}})}$$

此时，IICWLO 和 IICWH0 寄存器的最佳设置值如下所示。
(所有设置值的小数部分均经舍入处理。)

• 快速模式下

$$\begin{aligned} \text{IICWLO} &= \frac{0.52}{\text{Transfer clock}} \times f_{\text{CLK}} \\ \text{IICWH0} &= \left(\frac{0.48}{\text{Transfer clock}} - t_{\text{R}} - t_{\text{F}} \right) \times f_{\text{CLK}} \end{aligned}$$

• 标准模式下

$$\begin{aligned} \text{IICWLO} &= \frac{0.47}{\text{Transfer clock}} \times f_{\text{CLK}} \\ \text{IICWH0} &= \left(\frac{0.53}{\text{Transfer clock}} - t_{\text{R}} - t_{\text{F}} \right) \times f_{\text{CLK}} \end{aligned}$$

(2) 设置从设备方的 IICWLO 和 IICWH0 寄存器

(所有设置值的小数部分均经舍入处理。)

• 快速模式下

$$\begin{aligned} \text{IICWLO} &= 1.3 \mu\text{s} \times f_{\text{CLK}} \\ \text{IICWH0} &= (1.2 \mu\text{s} - t_{\text{R}} - t_{\text{F}}) \times f_{\text{CLK}} \end{aligned}$$

• 标准模式下

$$\begin{aligned} \text{IICWLO} &= 4.7 \mu\text{s} \times f_{\text{CLK}} \\ \text{IICWH0} &= (5.3 \mu\text{s} - t_{\text{R}} - t_{\text{F}}) \times f_{\text{CLK}} \end{aligned}$$

注意事项 设置传送时钟时要注意最小 f_{CLK} 工作频率。串行接口 IICA 的最小 f_{CLK} 工作频率取决于操作模式。

快速模式: $f_{\text{CLK}} = 3.5 \text{ MHz (MIN.)}$

标准模式: $f_{\text{CLK}} = 1 \text{ MHz (MIN.)}$

此外，串行接口 IICA 操作时钟的最大工作频率为 **20 MHz(Max.)**。如果 f_{CLK} 超过 **20 MHz**，则通过将 IICCTL01 寄存器的 PRS0 位设置为 **1** 将传送时钟设置为 $f_{\text{CLK}}/2$ 。

备注 1. 要分开计算 SDAA0 和 SCLA0 信号的上升时间 (t_{R})和下降时间 (t_{F})，因为它们因上拉电阻和线路负载而存在差异。

2. IICWLO: IICA 低电平宽度设置寄存器 0

IICWH0: IICA 高电平宽度设置寄存器 0

t_{F} : SDAA0 和 SCLA0 信号的下降时间

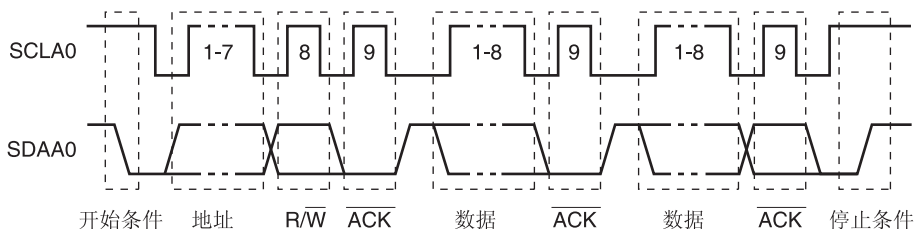
t_{R} : SDAA0 和 SCLA0 信号的上升时间

f_{CLK} : CPU/外围硬件时钟频率

17.5 I²C总线定义和控制方法

以下章节描述 I²C 总线的串行数据通信格式和 I²C 总线使用的信号。通过 I²C 总线的串行数据总线输出“开始条件”、“地址”、“数据”和“停止条件”时的传送时序如图 17-15 所示。

图 17-15. I²C 总线串行数据传送时序



由主设备生成开始条件，从设备地址和停止条件。

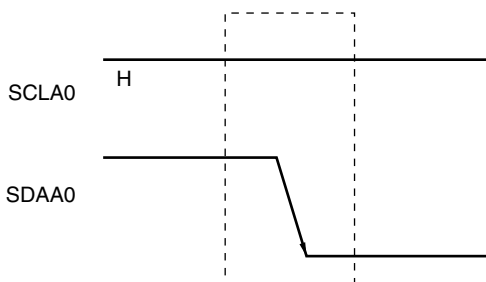
应答(ACK)信号既可由主设备产生，也可由从设备产生(通常情况下，它由接收 8 位数据的器件输出)。

串行时钟 (SCLA0) 由主设备连续输出。但是，从设备可以延长 SCLA0 引脚的低电平期间，并可插入等待。

17.5.1 开始条件

当 SCLA0 引脚保持高电平并且 SDAA0 引脚由高电平变为低电平时，形成一个开始条件。SCLA0 和 SDAA0 引脚的开始条件是开始串行传送时主设备对从设备发送的信号。当器件作为从设备时，可以检测到开始条件。

图 17-16. 开始条件



检测到停止条件(SPD0: IICA 状态寄存器 0 (IICS0)的位 0 = 1)后，如果 IICA 控制寄存器 00 (IICCTL00)的位 1 (STT0) 被设置为 1 时，输出开始条件。当检测到开始条件时，IICS0 寄存器的位 1 (STD0)被设置为 1。

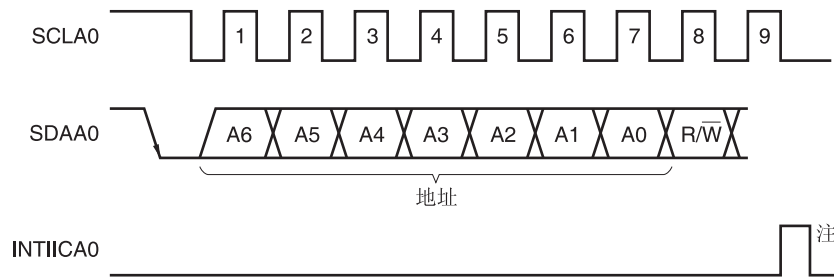
17.5.2 地址

开始条件的后续 7 位数据定义为地址。

主设备通过向总线发送 7 位的地址数据段来选择一个目标从设备。所以，每一个接入总线的从设备必须具有一个独一无二的地址。

从设备通过硬件检测开始条件，并检查 7 位地址数据与从设备地址寄存器 0 (SVA0)中保存的数据值是否匹配。如果地址数据和 SVA0 寄存器内储存的数据值相匹配，这个从设备就被选择，并且和主设备进行通信，直到主设备发送一个开始条件或结束条件。

图 17-17. 地址



注 在从设备操作期间，如果接收到本地地址或扩展代码以外的数据，不产生 INTIICA0。

地址是将从设备地址和 17.5.3 指定传送方向中所示传送方向合并起来构成的 8 位数据，将其写入 IICA 移位寄存器 0 (IICA0)并输出。另外，接收到的地址被写入 IICA0 寄存器。

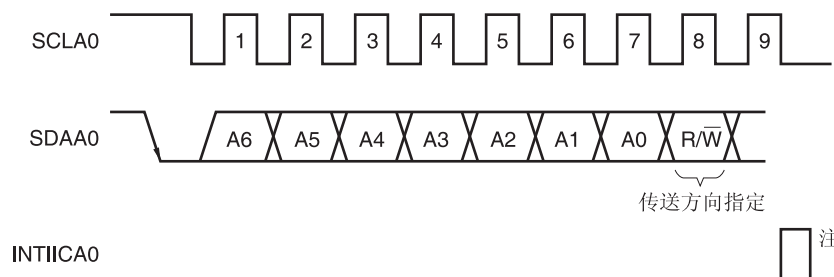
从设备地址被分配至 IICA0 寄存器的高 7 位。

17.5.3 指定传送方向

除了 7 位地址数据之外，主设备发送 1 位数据以指示传送方向。

当该传送方向指示位为 0 时，表示主设备向从设备发送数据。当该传送方向指示位为 1 时，表示主设备由从设备接收数据。

图 17-18. 指定传送方向



注 在从设备操作期间，如果接收到本地地址或扩展代码以外的数据，不产生 INTIICA0。

17.5.4 应答(ACK)

ACK 用于检查发送和接收方的串行数据的状态。

每收到 8 位数据，接收方都会返回 ACK 信号。

每发送 8 位数据，发送方通常会接收 ACK 信号。当从接收方返回 ACK 信号时，将假定接收已正确完成，继续处理。可以利用 IICA 状态寄存器 0 (IICS0)的位 2 (ACKD0) 来确认是否已检测到 ACK 信号。

当主设备接收时，接收到最后数据后，不返回 ACK 信号而是产生停止条件。从设备接收时，如果接收到数据后不返回 ACK 信号，主设备会输出停止条件或者重启条件，并停止传输。不返回 ACK 信号的可能原因如下所示。

- <1> 没有正常执行接收。
- <2> 已结束最后数据的接收。
- <3> 地址指定的接收方不存在。

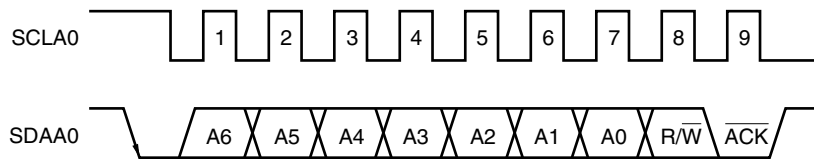
接收方通过在第 9 个时钟处使 SDAA0 线路处于低电平来产生 ACK (正常接收)。

通过将 IICA 控制寄存器 00 (IICCTL00)的位 2 (ACKE0)设置为 1，进入能够自动产生 ACK 的状态。由 7 位地址信息之后的第 8 位数据设置 IICS0 寄存器的位 3 (TRC0)。接收 (TRC0 = 0) 时，通常设置 ACEK0 位为 1。

从设备接收操作过程中(TRC0 = 0)，如果不能接收更多数据或者不需要下一个数据时，须将 ACEK0 位清除为 0，以不产生 ACK，从而告知主设备不能接收更多数据。

在主设备接收操作过程中(TRC0 = 0)，不需要下一个数据时，须将 ACEK0 位清除为 0，以不产生 ACK。这样的话，告知从设备发送方数据的结束 (停止发送)。

图 17-19. ACK



接收到本地地址后，不管 ACEK0 位的设置值如何，都会自动产生 ACK 信号。当接收到非本地以外的地址时，则不产生 ACK (NACK)。

如果 ACEK0 位被预先设置为 1，则在收到扩展代码时将产生 ACK 信号。

在接收到数据时的 ACK 产生方式各不相同，取决于等待时间的设置。

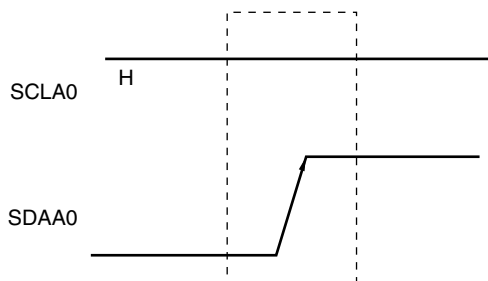
- 当选择 8 时钟等待时 (IICCTL00 寄存器的位 3(WTIM0) = 0):
通过在解除等待状态之前将 ACEK0 位设置为 1，则会在 SCLA0 引脚第 8 个时钟的下降沿产生 ACK。
- 当选择 9 时钟等待时 (IICCTL00 寄存器的位 3(WTIM0) = 1):
如果 ACEK0 位被预先设置为 1，则产生 ACK 信号。

17.5.5 停止条件

当 SCLA0 引脚保持高电平时，并且 SDAA0 引脚由低电平变为高电平，即产生一个停止条件。

停止条件是在完成串行传输时主设备向从设备发送的一种信号。当器件用作从设备时，可以检测到停止条件。

图 17-20. 停止条件



当 IICA 控制寄存器 00 (IICCTL00) 的位 0 (SPT0) 被设置为 1 时，将产生一个停止条件。当检测到停止条件时，IICA 状态寄存器 0 (IICS0) 的位 0 (SPD0) 将设置为 1，并在 IICCTL00 寄存器的位 4 (SPIE0) 被设置为 1 时产生 INTIICA0。

17.5.6 等待

等待用来通知通信伙伴，器件（主设备或从设备）已经准备就绪可以进行收发数据（即：处于等待状态）。

将 SCLA0 引脚设置为低电平，通知通信伙伴本机已经处于等待状态。当主设备和从设备的等待状态都被取消时，可以开始下一次传送。

图 17-21. 等待 (1/2)

(1) 主设备处于 9 时钟等待状态，从设备处于 8 时钟等待状态时
(主设备：发送，从设备：接收，ACKE0 = 1)

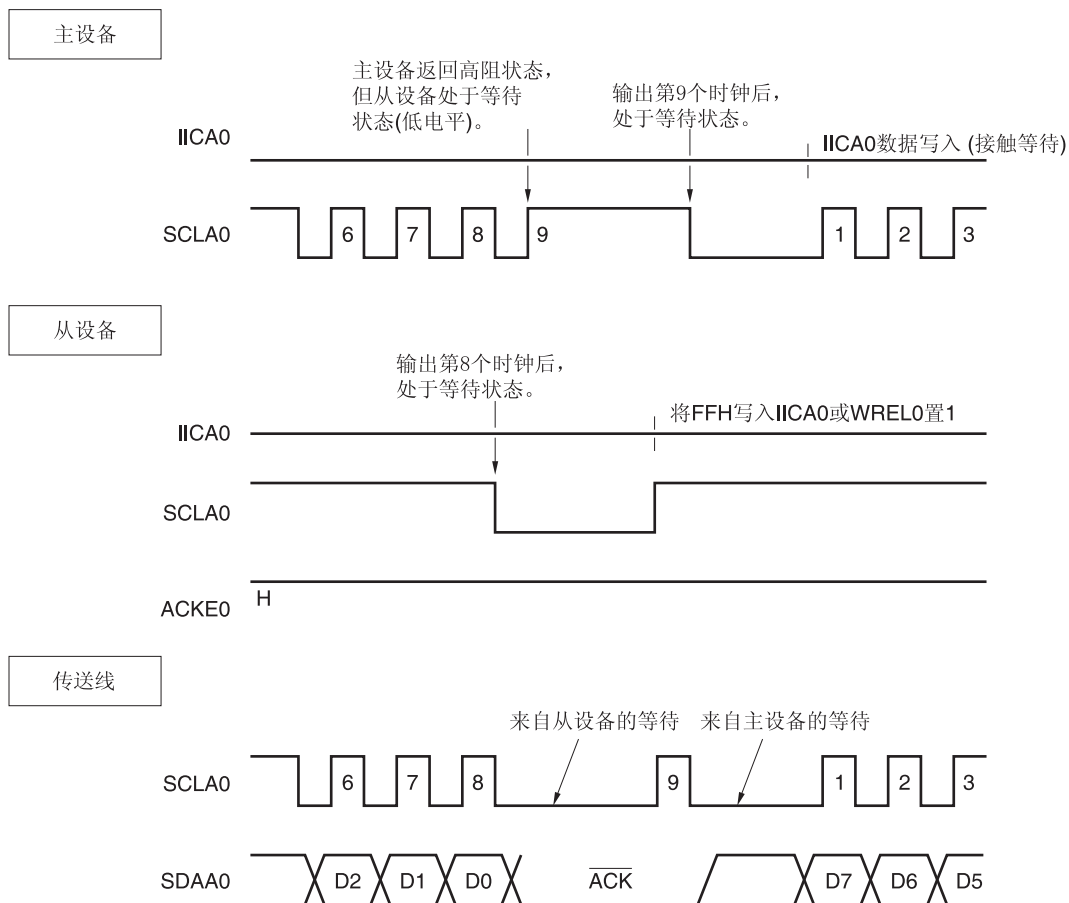
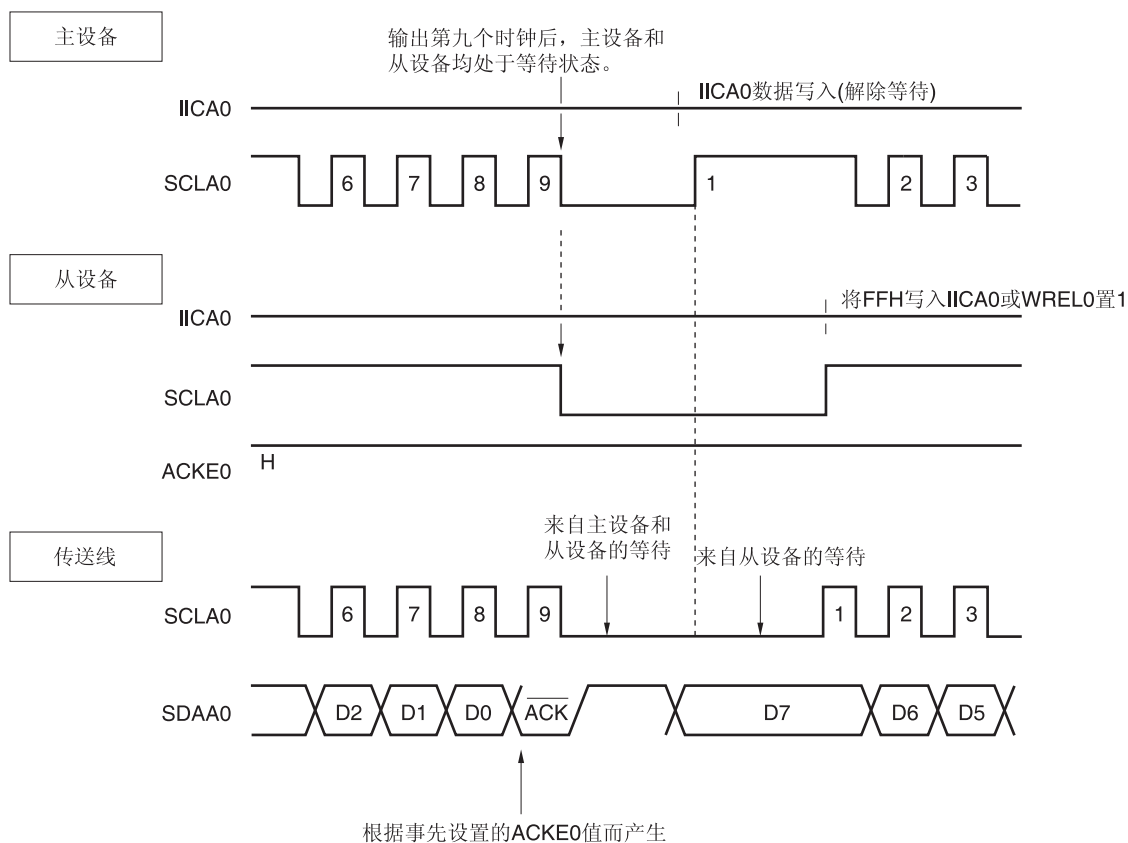


图 17-21. 等待 (2/2)

(2) 主设备和从设备都处于 9 时钟等待状态时

(主设备：发送，从设备：接收， $ACKEO = 1$)**备注** ACKEO: IICA 控制寄存器 00 (IICCTL00)的位 2

WRELO: IICA 控制寄存器 00 (IICCTL00)的位 5

通过设置 IICA 控制寄存器 00 (IICCTL00)的位 3 (WTIM0)可以自动产生等待。

通常，当 IICCTL00 寄存器的位 5 (WRELO) 设置为 1 或当 FFH 写入 IICA 移位寄存器 0 (IICA0)时，接收方会取消等待状态，而当数据写入 IICA0 寄存器时，发送方会取消等待状态。

主设备还可以通过以下方式取消等待状态。

- 通过将 IICCTL00 寄存器的位 1(STT0)设置为 1
- 通过将 IICCTL00 寄存器的位 0(SPT0)设置为 1

17.5.7 取消等待

I²C 通常通过以下处理来取消等待状态。

- 将数据写入 IICA 移位寄存器 0 (IICA0)
- 设置 IICA 控制寄存器 00 (IICCTL00)的位 5(WRELO) (取消等待)
- 设置 IICCTL00 寄存器的位 1(STT0) (发送开始条件)[※]
- 设置 IICCTL00 寄存器的位 0(SPT0) (发送停止条件)[※]

注 仅限主设备。

当执行以上等待取消处理时，I²C 将取消等待状态并恢复通信。

要取消等待状态并发送数据（包括地址），须向 IICA0 寄存器写入数据。

要在取消等待状态之后接收数据，或完成数据发送，则须将 IICCTL00 寄存器的位 5 (WRELO)设置为 1。

要在取消等待状态之后产生一个重启条件，则须将 IICCTL00 寄存器的位 1 (STT0)设置为 1。

要在取消等待状态之后产生一个停止条件，则须将 IICCTL00 寄存器的位 0 (SPT0)清除为 0。

一个等待状态只执行一次取消处理。

例如，如果在通过将 WRELO 位设置为 1 取消等待状态之后向 IICA0 寄存器写入数据，则可能向 SDA0 线路输出错误的值，因为更改 SDA0 线路的时序与写入 IICA0 寄存器的时序相冲突。

除此以外，如果 IICE0 位在中止通信时清除为 0，将停止通信，取消等待状态。

如果 I²C 总线因噪声而死锁，则通过设置 IICCTL00 寄存器的位 6 (LRELO) 来退出通信并保留处理，以取消等待状态。

注意事项 如果在 WUP0 = 1 时执行用于取消等待状态的处理，则等待状态不会被取消。

17.5.8 中断请求(INTIICA0)的产生时序和等待控制

IICA 控制寄存器 00 (IICCTL00)的位 3 (WTIMO) 的设置决定产生 INTIICA0 的时序以及相应的等待控制，如表 17-2 所示。

表 17-2. 控制 INTIICA0 产生时序和等待

WTIMO	从设备操作期间			主设备操作期间		
	地址	数据接收	数据发送	地址	数据接收	数据发送
0	9 ^{#1,2}	8 ^{#2}	8 ^{#2}	9	8	8
1	9 ^{#1,2}	9 ^{#2}	9 ^{#2}	9	9	9

注 1. 仅当接收的地址和从设备地址寄存器 0 (SVA0) 设定的值相匹配时，在第 9 个时钟的下降沿产生从设备的 INTIICA0 信号和等待。

此时，无论 IICCTL00 寄存器位 2 (ACKE0) 的设置值如何，都将产生 $\overline{\text{ACK}}$ 信号。对于已经接收到扩展代码的从设备，INTIICA0 将在第 8 个时钟的下降沿发生。

但是，如果重启后地址不匹配，INTIICA0 在第 9 个时钟的下降沿发生，但不发生等待。

2. 如果接收地址和从设备地址寄存器 0 (SVA0) 的内容不匹配，并且未接收到扩展代码，则既不产生 INTIICA0，也不产生等待。

备注 表中的数量表示串行时钟的时钟信号数。中断请求和等待控制都和这些时钟信号的下降沿同步。

(1) 地址发送/接收期间

- 从设备操作时： 中断和等待时序由上述注 1 和注 2 描述的条件决定，与 WTIMO 位无关。
- 主设备操作时： 中断和等待时序都在第 9 个时钟下降沿发生，与 WTIMO 位无关。

(2) 数据接收期间

- 主设备/从设备操作： 中断和等待时序由 WTIMO 位决定。

(3) 数据发送期间

- 主设备/从设备操作： 中断和等待时序由 WTIMO 位决定。

(4) 等待取消方法

四种等待取消方法如下所示：

- 将数据写入 IICA 移位寄存器 0 (IICA0)
- 设置 IICA 控制寄存器 00 (IICCTL00)的位 5(WRELO) (取消等待)
- 设置 IICCTL00 寄存器的位 1(STT0) (发送开始条件)[#]
- 设置 IICCTL00 寄存器的位 0(SPT0) (产生停止条件)[#]

注 仅限主设备。

当已经选择 8 时钟等待时 (WTIMO = 0)，在取消等待之前必须确定是否产生 $\overline{\text{ACK}}$ 。

(5) 停止条件检测

当检测到停止条件时，会产生 INTIICA0(仅限 SPIE0 = 1 时)。

17.5.9 地址匹配检测方法

在 I²C 总线模式下，通过发送从设备的地址，主设备可以选择特定的从设备。

地址匹配可以由硬件自动检测。当设置至从设备地址寄存器 0 (SVA0)的地址值与主设备发送的从设备地址相匹配时，或者当接收到扩展代码时，将产生一个中断请求(INTIICA0)。

17.5.10 错误检测

在 I²C 总线模式下，串行数据总线 (SDAA0) 在数据发送期间的状态可被发送器件的 IICA 移位寄存器 0 (IICA0)捕捉，于是，可以将发送之前的 IICA 数据与已发送的 IICA 数据进行比较，用来检测发送错误是否发生。当比较值不匹配时，判断为发生发送错误。

17.5.11 扩展代码

(1) 当接收地址的高 4 位是 0000 或 1111 时接收扩展代码，将扩展代码接收标志(EXC0)设置为 1，且在第 8 个时钟的下降沿处产生一个中断请求(INTIICA0)。存储在从设备地址寄存器 0 (SVA0)中的本地地址不受影响。

(2) 在 SVA0 寄存器被设置为 11110xx0 时，如果通过 10 位地址传送从主设备发送 11110xx0，则以下设置被指定。注意，INTIICA0 发生于第 8 个时钟的下降沿。

- 数据匹配的高 4 位: EXC0 = 1
- 数据匹配的 7 位: COI0 = 1

备注 EXC0: IICA 状态寄存器 0 (IICS0)的位 5
COI0: IICA 状态寄存器 0 (IICS0)的位 4

(3) 根据扩展代码之后的数据，中断请求发生后的处理各不相同，这些处理由软件执行。

如果在从设备工作期间收到扩展代码，即使地址不匹配，从设备也在参与通信。

例如，在接收到扩展代码之后，如果不希望将目标器件用作从设备，可将 IICA 控制寄存器 00 (IICCTL00)的位 6 (LRELO) 设置为 1，为下一个通信操作设置待机模式。

表 17-3. 主设备要扩展码的位定义

从设备地址	R/W 位	说明
0 0 0 0 0 0 0	0	通用呼叫地址
1 1 1 1 0 x x	0	10 位从设备地址规范 (地址认证期间)
1 1 1 1 0 x x	1	10 位从设备地址规范 (地址匹配后，发出读取命令时)

备注 有关上述以外扩展代码的详细信息，请参阅恩智浦半导体发布的 I²C 总线规范。

17.5.12 仲裁

当多个主设备同时发送开始条件时（在 STD0 位被设置为 1 之前，STT0 位已设置为 1），需要调整时钟数目直到数据不同，多个主设备之间的通信才能执行。这种操作被称为仲裁。

当一个主设备在仲裁中失败时，通过仲裁失败发生时序将 IICA 状态寄存器 0 (IICS0)中的仲裁失败标志(ALD0)设置为 1，并将 SCLA0 和 SDAA0 线路都被设置为高阻抗，从而释放总线。

仲裁失败的检测基于下次中断请求的时序（例如，在第 8 或第 9 个时钟，当检测到停止条件时）和通过软件设置的 ALD0 = 1。

关于中断请求时序的详情，请参阅 17.5.8 中断请求(INTIICA0)的产生时序和等待控制。

备注 STD0: IICA 状态寄存器 0 (IICS0)的位 1
STT0: IICA 控制寄存器 00 (IICCTL00)的位 1

图 17-22. 仲裁时序示例

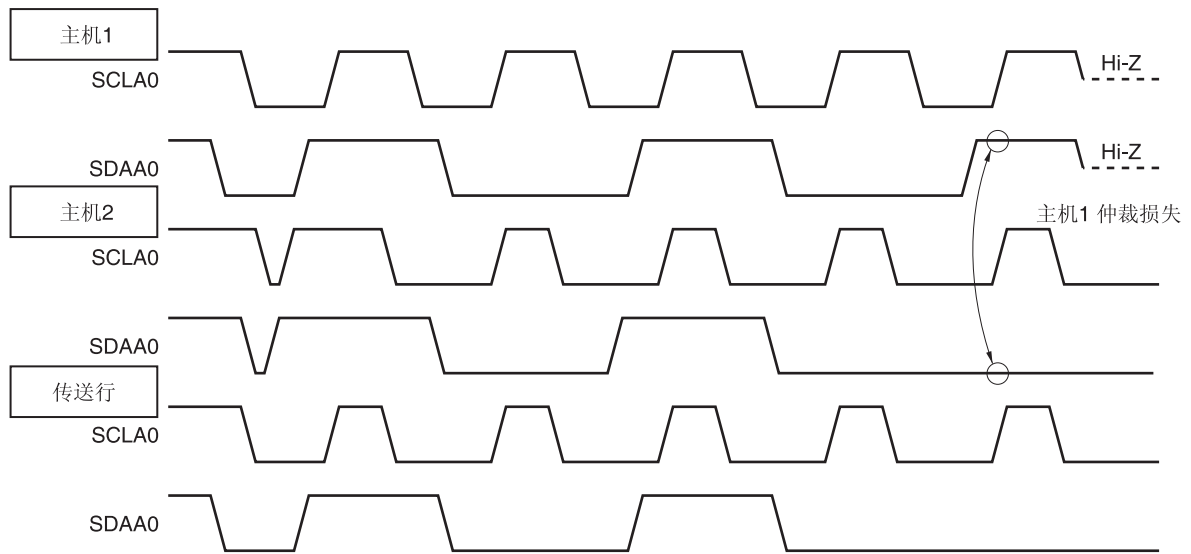


表 17-4. 产生仲裁时的状态和产生中断请求的时序

产生仲裁时的状态	产生中断请求的时序
地址发送期间	字节传输之后的第 8 个或第 9 个时钟的下降沿 ^{注1}
发送地址后读出/写入数据	
扩展码发送期间	
发送扩展码后读出/写入数据	
数据发送期间	
数据发送后的 $\overline{\text{ACK}}$ 传送期间	
在数据传输期间检测到重新开始条件	
在数据传输期间检测到停止条件	当产生停止条件时 (当 SPIE0 = 1) ^{注2}
尝试产生复位条件的同时数据处于低电平时	字节传输之后的第 8 个或第 9 个时钟的下降沿 ^{注1}
尝试产生复位条件的同时检测到停止条件时	当产生停止条件时 (当 SPIE0 = 1) ^{注2}
尝试产生停止条件的同时数据处于低电平时	字节传输之后的第 8 个或第 9 个时钟的下降沿 ^{注1}
尝试产生复位条件的同时 SCLA0 处于低电平时	

注 1. 当 WTIM0 位 (IICA 控制寄存器 00 (IICCTL00)的位 3) = 1 时, 将在第 9 个时钟的下降沿发生一个中断请求。当 WTIM0 = 0 并收到扩展代码的从设备地址时, 则会在第 8 个时钟的下降沿发生一个中断请求。

2. 当有可能发生仲裁时, 对于主设备操作, 则须设置 SPIE0 = 1。

备注 SPIE0: IICA 控制寄存器 00 (IICCTL00)的位 4

17.5.13 唤醒功能

I²C 总线从设备功能用于在收到本地地址和扩展代码时产生一个中断请求信号(INTIICA0)。

当地址数据不匹配时，该功能就会阻止产生不必要的 INTIICA0 信号，使处理更有效率。

当检测到开始条件时，设置唤醒待机模式。仲裁失败可能将主设备（已经产生开始条件）变为从设备，在这种情况下发送地址时，唤醒待机模式有效。

要在 STOP 模式下使用唤醒功能，则须将 WUP0 位设置为 1。无论操作时钟为何，均可接收到地址。在收到本地地址和扩展代码时，会产生一个中断请求信号(INTIICA0)。在产生该中断之后，通过一条指令将 WUP0 位清除为 0，可以返回正常操作。

图 17-23 所示为设置 WUP0 = 1 的流程，图 17-24 所示为在地址匹配时设置 WUP0 = 0 的流程。

图 17-23. 设置 WUP0 = 1 的流程

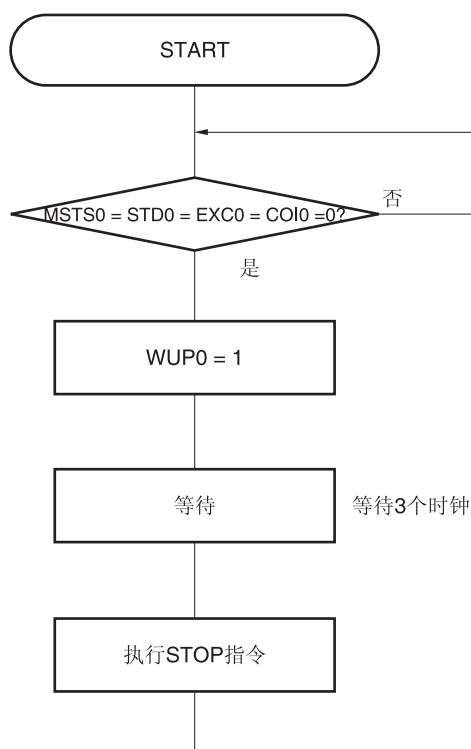
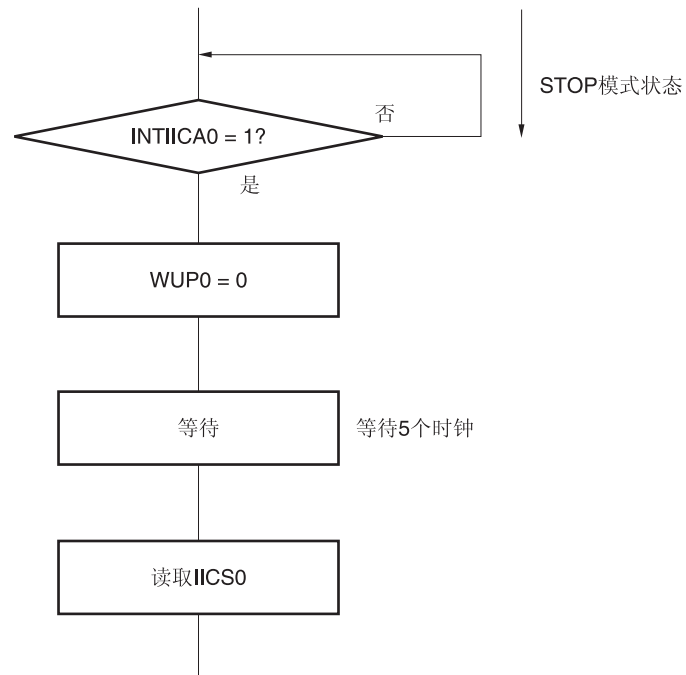


图 17-24. 地址匹配时设置 WUP0 = 0 的流程 (包括扩展码接收)

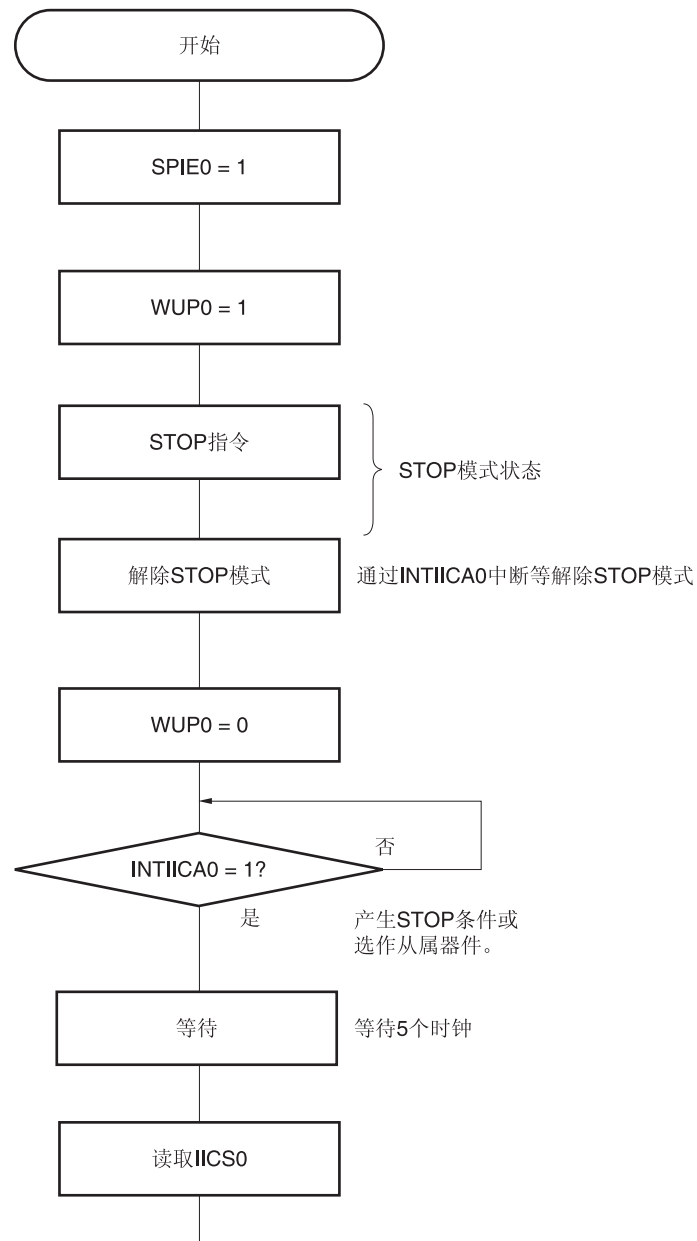


确认串行接口IICA的操作状态后，
按照要执行的操作处理。

用以下流程进行处理以解除 STOP 模式，而不是利用串行接口 IICA 产生的中断请求 (INTIICA0)。

- 主设备操作：流程如图 17-25 所示
- 从设备操作：与图 17-24 所示流程相同

图 17-25. 在通过 INTIICA0 之外的方式解除 STOP 模式后工作于主设备模式



确认串行接口IICA的操作状态后，
根据要执行的操作进行处理。

17.5.14 通信预约

(1) 当允许通信预约功能 (IICA 标志寄存器 0 (IICF0)的位 0 (IICRSV) = 0)

如果当前未使用总线但希望开始主设备通信，可使用通信预约功能，这样可以在释放总线后允许发送开始条件。这里的未使用总线包括以下两种模式。

- 当仲裁结果既不是主设备操作，也不是从设备操作
- 当接收到扩展代码且禁止从设备操作时（不返回 ACK 信号，并且通过将 IICA 控制寄存器 00 (IICCTL00)的位 6 (LRELO) 设置为 1 来保留通信并释放总线）。

如果在未使用总线时将 IICCTL00 寄存器的位 1 (STT0)设置为 1，总线被释放后（在检测到停止条件），会自动发送开始条件并设置为等待状态。

如果在 IICCTL00 寄存器的位 4 (SPIE0)设置为 1 之后，并且通过产生一个中断请求信号(INTIICA0)而检测到总线已被释放（检测到停止条件）后，向 IICA 移位寄存器 0 (IICA0)写入地址，则器件自动作为主设备开始通信。直到检测到停止条件为止，写入 IICA0 寄存器的数据无效。

当 STT0 位被设置为 1 后，根据总线状态决定工作模式（开始条件或通信预约）。

- 如果总线被释放 产生一个开始条件
- 如果总线未被释放（待机模式） 通信预约

将 STT0 位设置为 1 且经过等待时间后，利用 MSTSO 位(IICA 状态寄存器 0 (IICS0)的位 7)确认是否工作于通信预约模式。

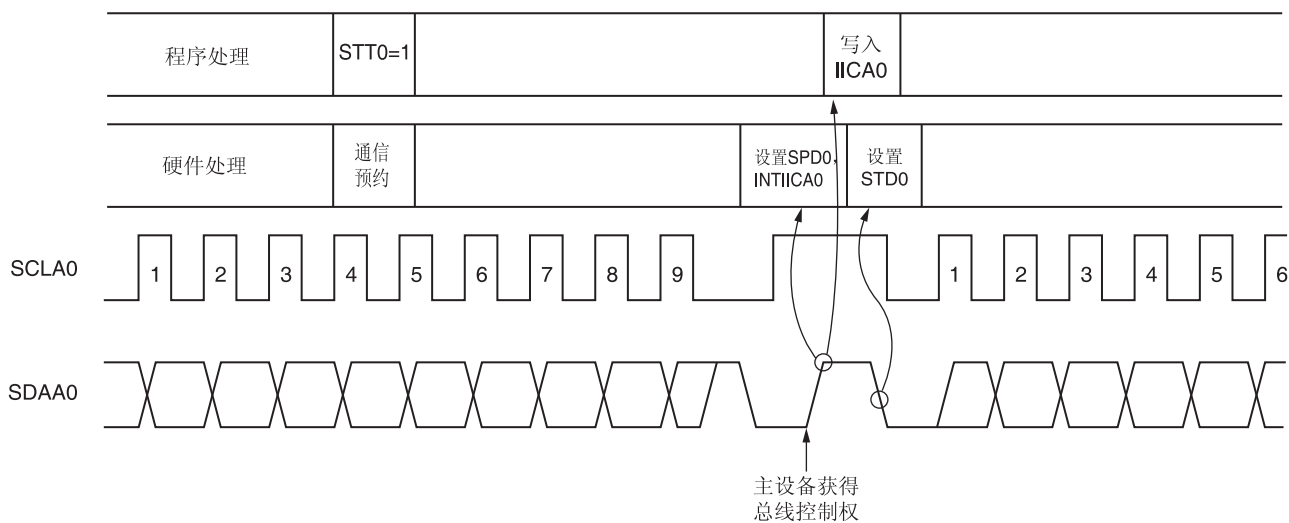
用软件确保通过以下公式计算的等待时间。

从设置 STT0 = 1 到确认 MSTSO 标志的等待时间：
 $(IICWLO \text{ 设定值} + IICWHO \text{ 设定值} + 4) + t_F \times 2 \times f_{CLK}$ [时钟]

备注 IICWLO: IICA 低电平宽度设置寄存器 0
 IICWHO: IICA 高电平宽度设置寄存器 0
 t_F: SDAA0 和 SCLA0 信号的下降时间
 f_{CLK}: CPU/外围硬件时钟频率

通信预约时序如图 17-26 所示。

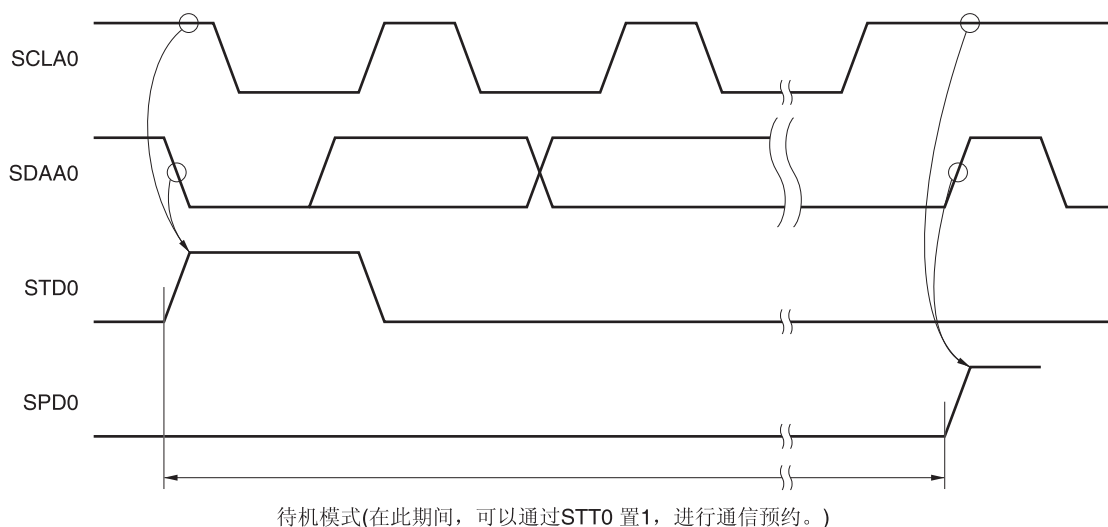
图 17-26. 通信预约时序



- 备注**
- IICA0: IICA 移位寄存器 0
 - STT0: IICA 控制寄存器 00 (IICCTL00)的位 1
 - STD0: IICA 状态寄存器 0 (IICS0)的位 1
 - SPD0: IICA 状态寄存器 0 (IICS0)的位 0

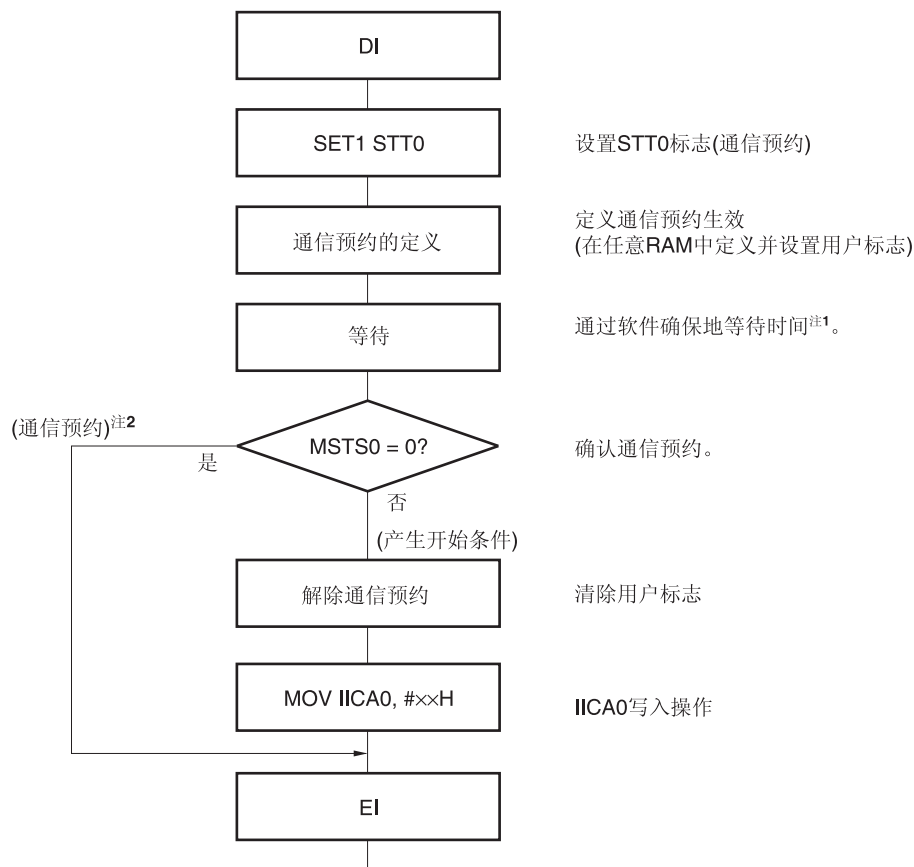
通过如图 17-27 所示的时序处理通信预约。在将 IICA 状态寄存器 0 (IICS0)的位 1 (STD0) 设置为 1 之后，可以在检测到停止条件之前将 IICA 控制寄存器 00 (IICCTL00)的位 1 (STT0)设置为 1 来预约通信。

图 17-27. 处理通信预约的时序



通信预约协议如图 17-28 所示。

图 17-28. 通信预约协议



注 1. 等待时间的计算方法如下所示。

$$(IICWLO \text{ 设定值} + IICWHO \text{ 设定值} + 4) + t_f \times 2 \times f_{CLK} [\text{时钟}]$$

2. 通信预约操作期间如发生停止条件中断请求，则写入 IICA 移位寄存器 0 (IICA0)。

备注 STT0: IICA 控制寄存器 00 (IICCTL00)的位 1

MSTS0: IICA 状态寄存器 0 (IICS0)的位 7

IICA0: IICA 移位寄存器 0

IICWLO: IICA 低电平宽度设置寄存器 0

IICWHO: IICA 高电平宽度设置寄存器 0

t_f: SDAA0 和 SC, LA0 信号的下降时间

f_{CLK}: CPU/外围硬件时钟频率

(2) 当禁止通信预约功能 (IICA 标志寄存器 0 (IICF0)的位 0 (IICRSV) = 1)

如果在总线通信期间，未在使用总线进行通信时将 IICA 控制寄存器 00 (IICCTL00)的位 1 (STT0)设置为 1，则该预约请求将被拒绝且不会发送开始条件。这里的未使用总线包括以下两种状态：

- 当仲裁结果既不是主设备操作，又不是从设备操作
- 当接收扩展代码并且禁止从设备操作时(不返回 $\overline{\text{ACK}}$ ，并且通过将 IICCTL00 寄存器的位 6 (LREL0)设置为 1 来保留通信并释放总线)。

可以通过 STCF0 (IICF0 寄存器的位 7)来确认是产生了开始条件还是请求被拒绝。设置 STT0 = 1 后，一直到 STCF0 位设置为 1 为止，需要 5 个时钟。须通过软件来确保这段时间。

17.5.15 注意事项

(1) 当 STCEN = 0

在允许 I²C 操作(IICE0 = 1)之后, 无论总线的实际状态如何, 默认处于总线通信状态 (IICBSY = 1)。在没有检测到停止条件的状态下希望执行主设备通信时, 须先发送一个停止条件以释放总线, 然后执行主设备通信。

使用多个主设备时, 总线未被释放(未检测到停止信号时)的状态下不能执行主设备通信。

通过以下步骤来产生停止条件。

- <1> 设置 IICA 控制寄存器 01 (IICCTL01)。
- <2> 将 IICA 控制寄存器 00 (IICCTL00)的位 7 设置为 1。
- <3> 将 IICCTL00 寄存器的位 0 (SPT0)设置为 1。

(2) 当 STCEN = 1

刚允许操作 I²C 操作 (IICE0 = 1)后, 不管总线的实际状态如何, 默认处于总线释放状态 (IICBSY = 0)。若要产生第一个开始条件 (STT0 = 1), 为了不干扰其他的通信, 必需确认总线已经被释放。

(3) 如果正在和其他方进行 I²C 通信时

如果在 SDA0 引脚为低电平、SCLA0 引脚为高电平时允许操作 I²C 并且设备正在通信, 则 I²C 宏将认为 SDA0 引脚从高电平改变为低电平(检测到一个开始条件)。如果此时总线上的值可以识别为扩展代码, 则会返回 ACK, 将干扰其他 I²C 通信。为了避免这种情况, 须按以下顺序启动 I²C。

- <1> 将 IICCTL00 寄存器的位 4 (SPIE0)清除为 0, 并禁止检测到停止条件时产生中断请求信号(INTIICA0)。
- <2> 将 IICCTL00 寄存器的位 7 (IICE0)设置为 1 以允许 I²C 操作。
- <3> 等待开始条件的检测。
- <4> 返回 ACK (将 IICE0 位设置为 1 之后经过了 4 至 80 个时钟周期)之前, 将 IICCTL00 寄存器的位 6 (LRELO)设置为 1, 强制性地禁止检测。

(4) 设置 STT0 和 SPT0 位 (IICCTL00 寄存器的位 1 和位 0) 之后、被清除为 0 之前, 禁止对其再次进行设置。

(5) 进行通信预约时, 将 SPIE0 位 (IICCTL0 寄存器的位 4) 设置为 1, 以允许在检测到停止条件时产生中断请求。当产生中断请求之后将通信数据写入 IICA 移位寄存器 0 (IICA0), 开始传送。在检测到停止条件时不产生中断, 否则, 器件将停止于等待状态, 因为开始通信时不产生中断请求。但是, 当通过软件检测到 MSTS0 位 (IICA 状态寄存器 0 (IICS0)的位 7) 时, 不必将 SPIE0 位设置为 1。

17.5.16 通信操作

下面通过流程图展示三种操作过程。

(1) 单主设备(线)系统下的主设备操作

在单主设备系统中将 RL78/I1A 用作主设备的流程图如下所示。

操作流程大体可以分为初始设置和通信处理两部分。在启动时执行初始设置。如果需要和从设备进行通信，进行通信准备后，执行通信处理。

(2) 多主设备(线)系统下的主设备操作

在 I²C 总线多主设备系统中，参与通信之后，仅根据 I²C 总线规范不能判断总线是被释放还是正在使用。此时，当数据和时钟在一定时期内（1 帧）处于高电平时，视为处于总线被释放的状态，RL78/I1A 参与通信。

操作流程可以大体分为初始设置、通信等待和通信处理。RL78/I1A 在仲裁失败并被指定为从设备的处理过程这里予以省略，仅显示其用作主设备的处理操作过程。启动时执行初始设置以便参与总线。然后，作为主设备等待通信请求或作为从设备等待指定。在通信处理部分执行实际通信，且支持与从设备的数据发送/接收操作以及与其他主设备的仲裁。

(3) 从设备操作

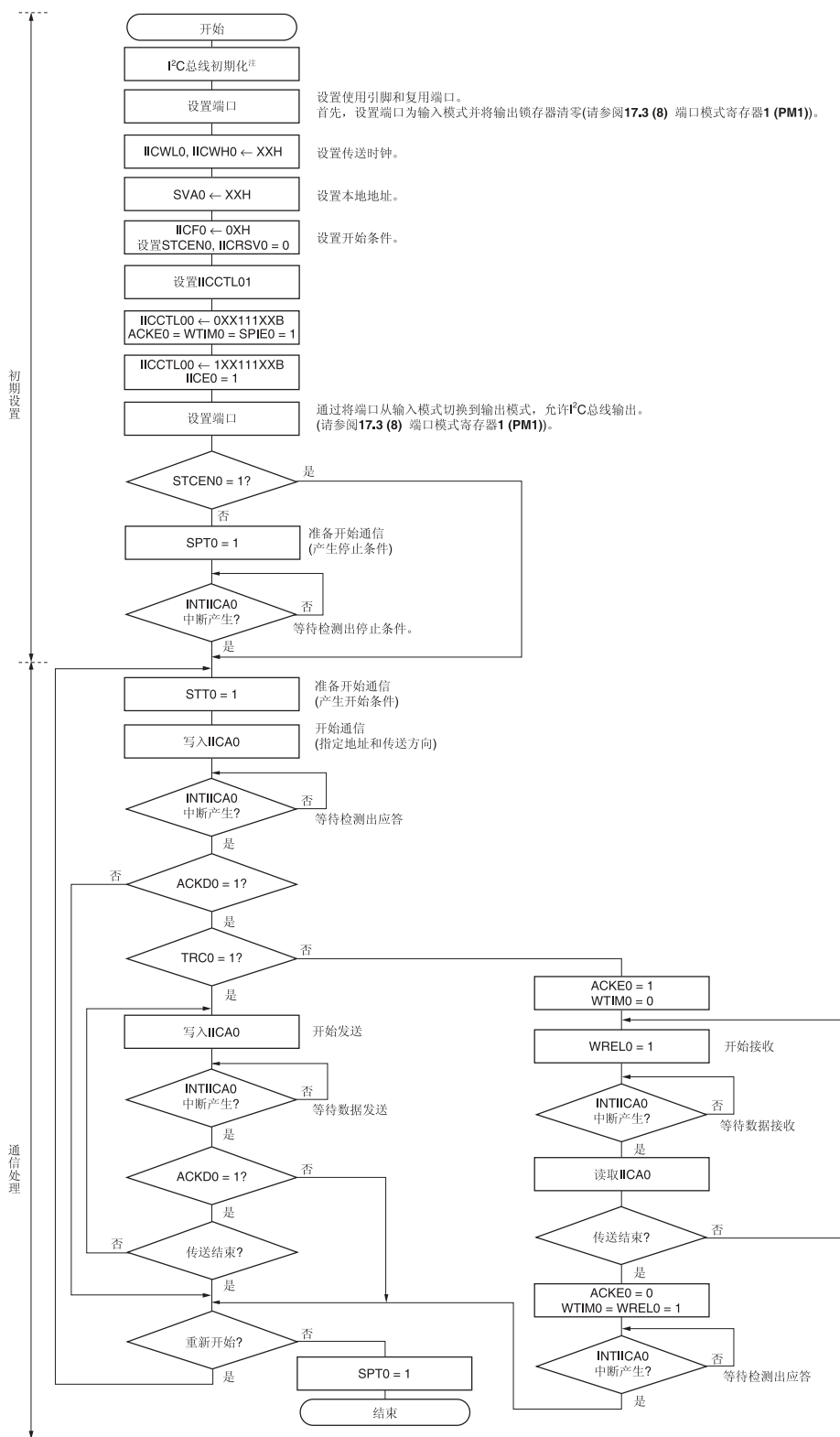
将 RL78/I1A 用作 I²C 总线从设备的示例如下所示。

当用作从设备时，由一个中断启动操作。在启动时执行初始设置，然后，等待 INTIICA0 中断信号的发生（通信等待）。当产生 INTIICA0 中断时，判断通信状态，并将判断结果作为一个标志传送给主程序。

通过检查该标志，来执行必要的通信处理。

(1) 单主系统下的主设备操作

图 17-29. 单主系统下的主设备操作

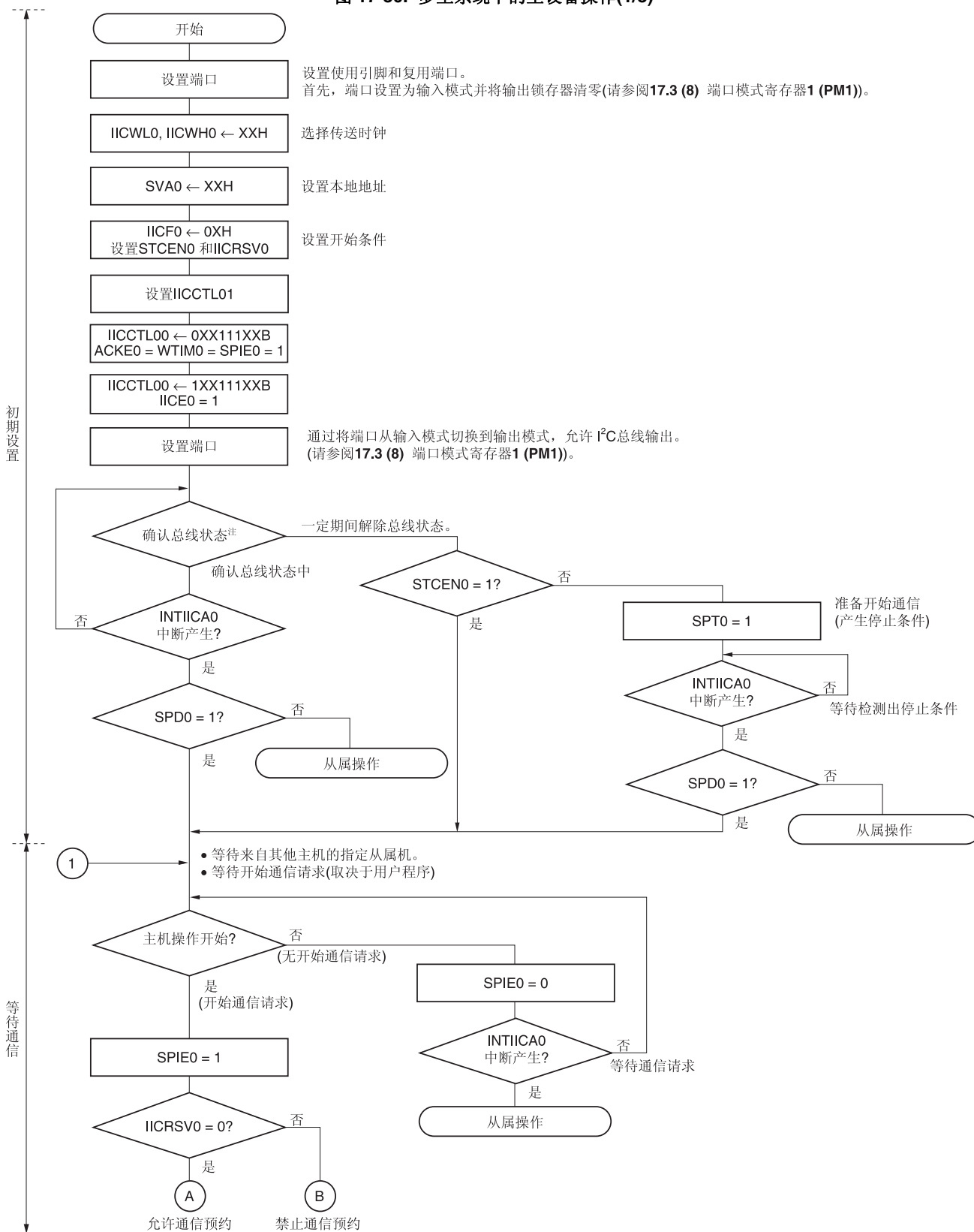


注 根据通信中产品的相关规格释放 I²C 总线 (SCLA0 和 SDAA0 引脚 = 高电平)。例如，当 EEPROM 向 SDAA0 引脚输出一个低电平时，将 SCLA0 引脚设置为输出端口模式，并从该输出端口输出时钟脉冲，直至 SDAA0 引脚为稳定的高电平为止。

备注 须遵循通信中产品规格的发送和接收格式要求。

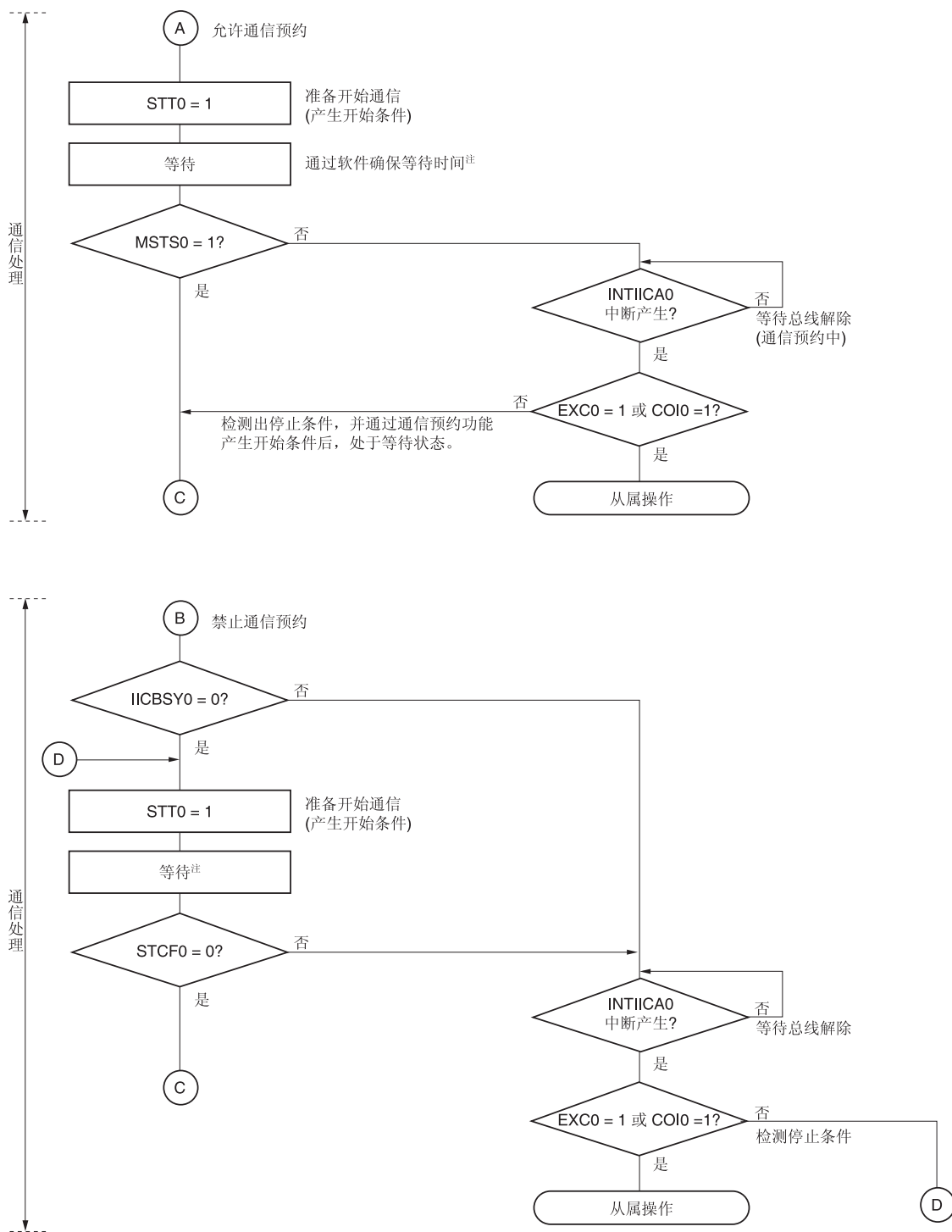
(2) 多主系统下的主设备操作

图 17-30. 多主系统下的主设备操作(1/3)



注 确认在一定时间内(例如，一帧)处于总线被释放的状态(CLD0位=1, DAD0位=1)。如果SDAA0引脚稳定处于低电平，则根据通信中产品的相关规格，决定是否释放I²C总线(SCLA0和SDAA0引脚=高电平)。

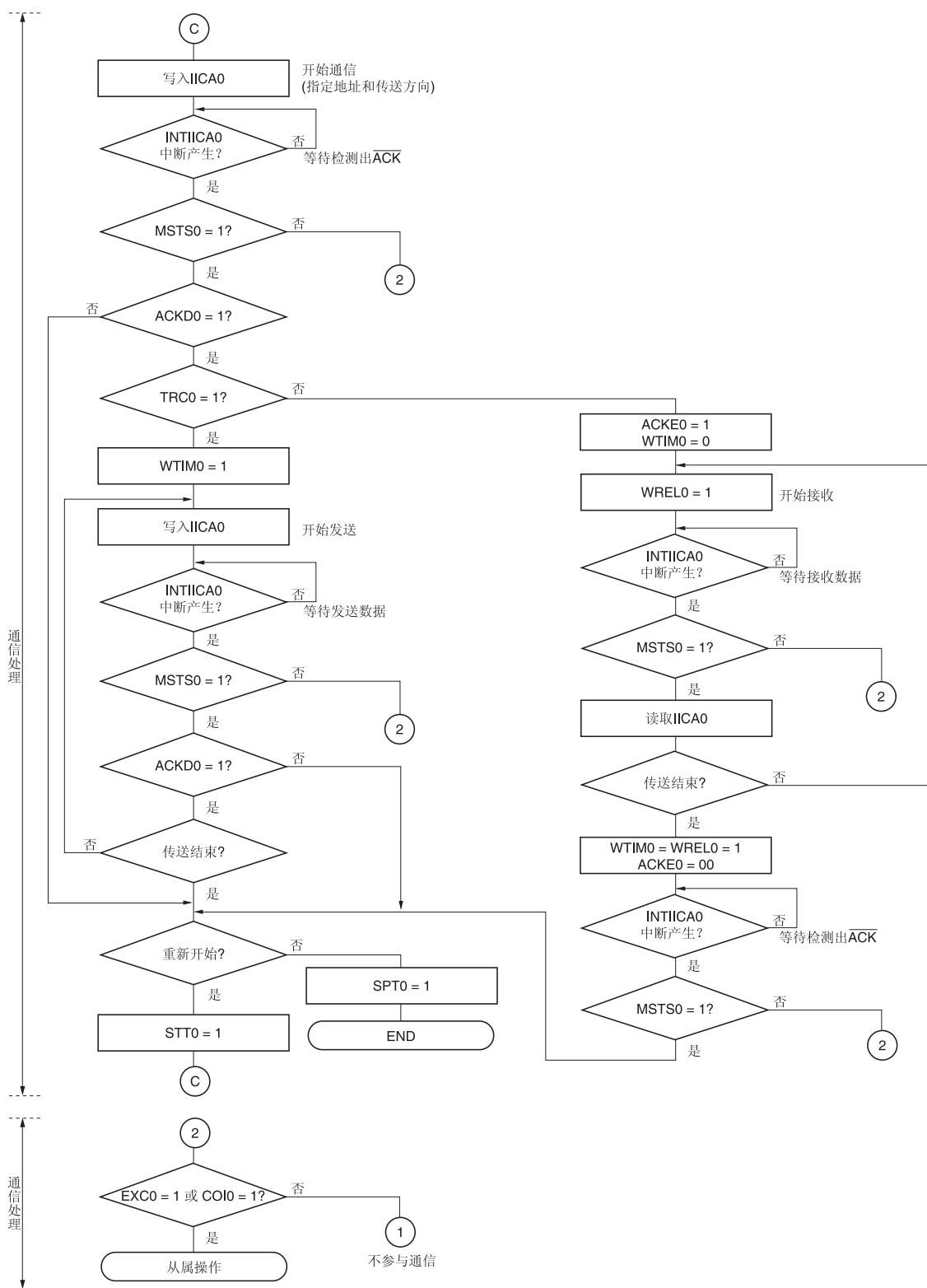
图 17-30. 多主系统下的主设备操作(2/3)



注 等待时间的计算方法如下所示。
 $(IICWLO \text{ 设定值} + IICWHO \text{ 设定值} + 4) \div f_{CLK} + t_F \times 2$ [时钟]

备注 IICWLO: IICA 低电平宽度设置寄存器 0
 IICWHO: IICA 高电平宽度设置寄存器 0
 t_F: SDAA0 和 SCLA0 信号的下降时间
 f_{CLK}: CPU/外围硬件时钟频率

图 17-30. 多主系统下的主设备操作(3/3)



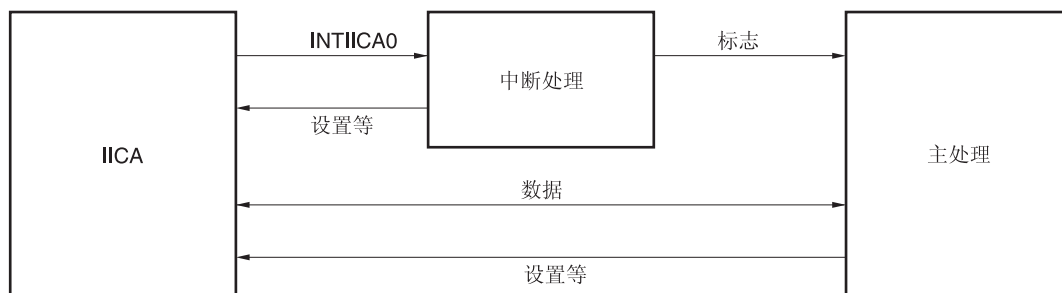
- 备注 1.** 须遵循通信中产品规格的发送和接收格式要求。
- 2.** 在多主设备系统中将器件用作主设备的情况下，每当发生中断 INTIICA0 时均需读取 MSTS0 位以检查仲裁结果。
- 3.** 在多主设备系统中将器件用作从设备的情况下，每当发生 INTIICA0 中断时均需利用 IICA 状态寄存器 0 (IICS0) 和 IICA 标志寄存器 0 (IICF0) 来检查状态，并确认接下来将执行的处理。

(3) 从设备操作

从设备操作的处理步骤如下所示。

基本而言，从设备操作都是由事件驱动的。所以，必需通过 INTIICA0 中断进行处理（使操作状态发生重大改变的处理，比如，通信过程中检测到停止条件）。

以下说明假定数据通信不支持扩展代码。还假定 INTIICA0 中断处理只执行状态改变处理，并且实际的数据通信由主程序执行。



因此，数据通信处理的执行需要准备以下三个标志来代替 INTIICA0，并将其传送至主程序。

<1> 通信模式标志

该标志表示以下两种通信状态。

- 清除模式：不执行数据通信的状态
- 通信模式：执行数据通信的状态 (从有效地址检测到停止条件检测，不检测来自主设备的 $\overline{\text{ACK}}$ 和地址不匹配)

<2> 就绪标志

该标志表示允许数据通信。对于正常的通信，其功能与 INTIICA0 中断相同。该标志由中断处理设置，并由主程序清除。在开始通信时先以中断处理清除该标志。但是，在发送第一个数据时，不通过中断处理设置就绪标志。因此，在该标志不被清除的情况下发送第一个数据 (地址匹配被理解为对下一个数据的请求)。

<3> 通信方向标志

该标志表示通信方向。它具有与 TRC0 位相同的值。

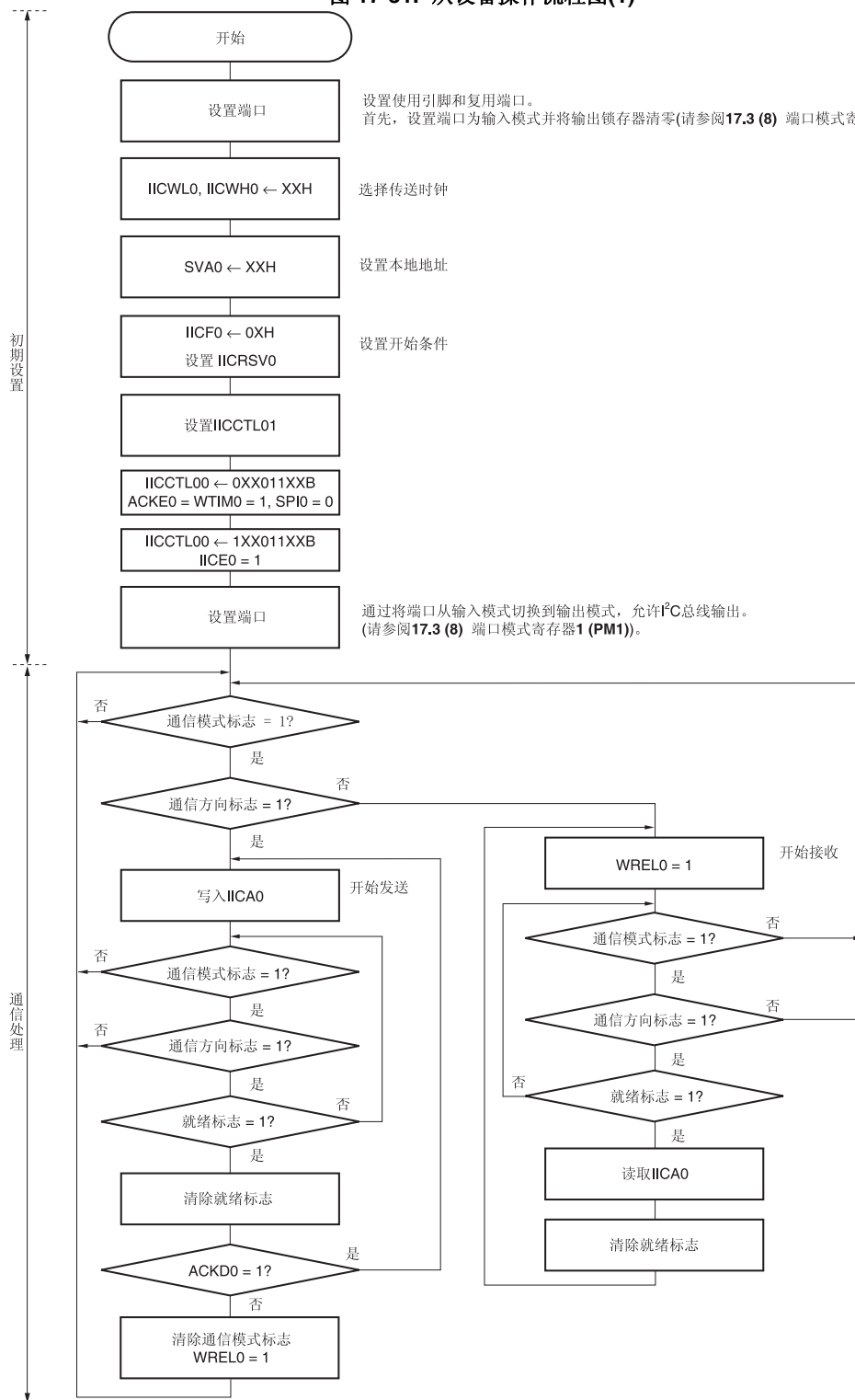
接下来说明从设备操作时主程序的操作。

启动串行接口 IICA 并等到允许通信的状态。允许通信状态后，使用通信模式标志和就绪标志执行通信（停止条件和开始条件处理由中断执行。在这里只用标志来检查状态）。

重复发送操作，直到主设备停止返回 \overline{ACK} 信号。如果主设备不返回 \overline{ACK} ，则完成通信。

对于接收需要收到必要数量的数据，当通信完成后，在下一个数据不返回 \overline{ACK} 。此后，主设备产生一个停止条件或重启条件。至此，退出通信状态。

图 17-31. 从设备操作流程(1)



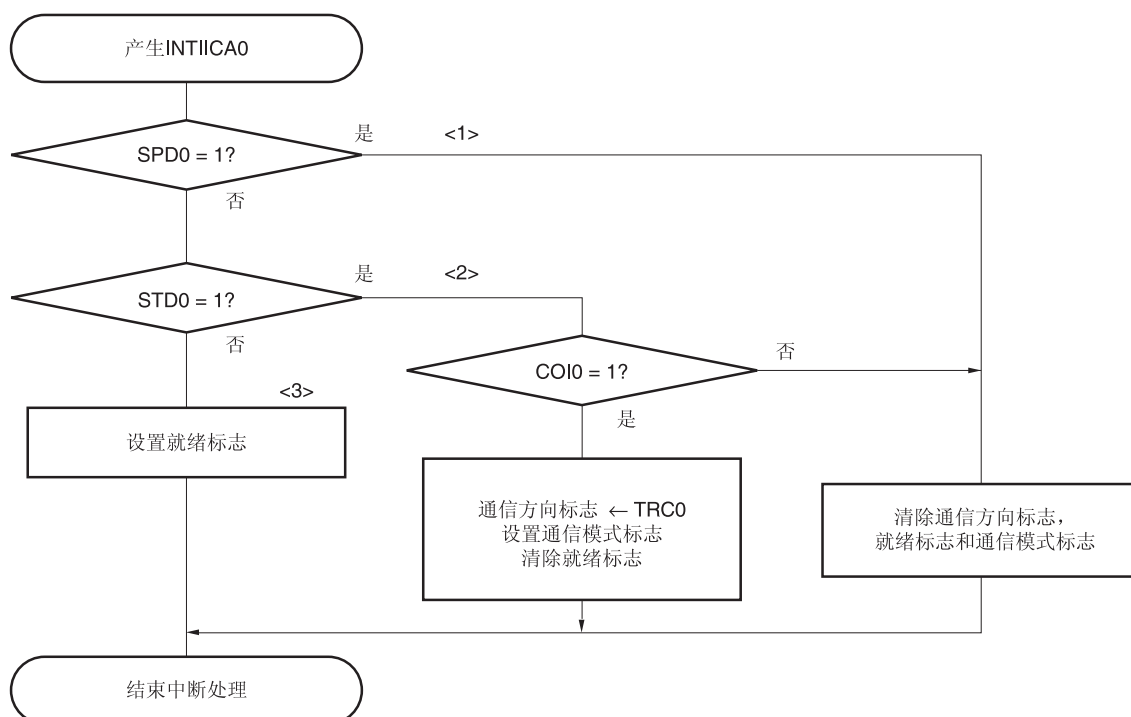
备注 须遵循通信中产品规格的发送和接收格式要求。

下面举例说明从设备利用 INTIICA0 中断进行处理的步骤 (假定处理中未使用扩展代码)。通过 INTIICA0 中断来检查状态, 并执行以下操作。

- <1> 如果产生停止条件, 则结束通信。
- <2> 如果发送开始条件, 则确认地址, 如果地址不匹配, 则结束通信。如果地址匹配, 则设置为通信模式, 解除等待, 并从中断返回处理 (就绪标志被清除)。
- <3> 发送/接收数据时, 仅设置就绪标志。I²C 总线保持等待状态, 从中断返回处理操作。

备注 上述<1>至<3>对应于图 17-32. 从设备操作流程图(2)中的<1>至<3>。

图 17-32. 从设备操作流程图(2)



17.5.17 I²C 中断请求(INTIICA0)的发生时序

发送或接收数据的时序、产生中断请求信号 INTIICA0 的时序以及产生 INTIICA0 信号时 IICA 状态寄存器 0 (IICS0) 的值如下所示。

- 备注**
- ST: 开始条件
 - AD6 至 AD0: 地址
 - R/W: 传送方向指示
 - ACK: 应答
 - D7 至 D0: 数据
 - SP: 停止条件

(1) 主设备操作

(a) 开始 ~ 地址 ~ 数据 ~ 数据 ~ 停止 (发送/接收)

(i) 当 $WTIMO = 0$ 时

▲1: IICS0 = 1000×110B

▲2: IICS0 = 1000×000B

▲3: IICS0 = 1000×000B (将 $WTIMO$ 位设置为 1)[※]▲4: IICS0 = 1000××00B (将 $SPT0$ 位设置为 1)[※]

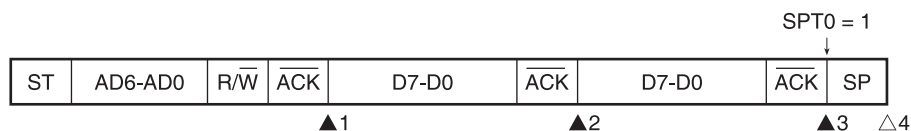
△5: IICS0 = 00000001B

注 若要产生停止条件，须将 $WTIMO$ 位设置为 1，并更改产生 $INTIICA0$ 中断请求信号的时序。

备注 ▲: 总是产生

△: 仅当 $SPIE0 = 1$ 时产生

×: 忽略

(ii) 当 $WTIMO = 1$ 时

▲1: IICS0 = 1000×110B

▲2: IICS0 = 1000×100B

▲3: IICS0 = 1000××00B (将 $SPT0$ 位设置为 1)

△4: IICS0 = 00000001B

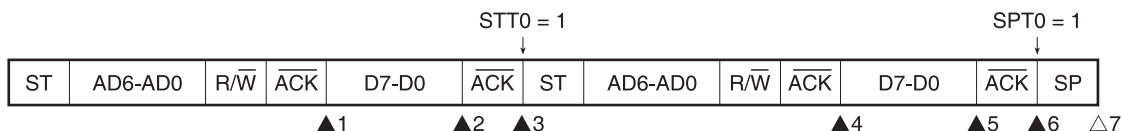
备注 ▲: 总是产生

△: 仅当 $SPIE0 = 1$ 时产生

×: 忽略

(b) 开始 ~ 地址 ~ 数据 ~ 开始 ~ 地址 ~ 数据 ~ 停止 (重启)

(i) 当 $WTIM0 = 0$ 时

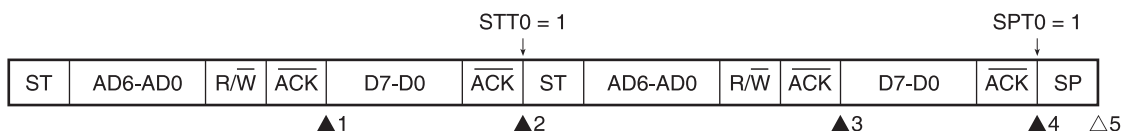


- ▲1: IICS0 = 1000x110B
- ▲2: IICS0 = 1000x000B (将 $WTIM0$ 位设置为 1)^{注1}
- ▲3: IICS0 = 1000xx00B (将 $WTIM0$ 位清除为 0^{注2}, 将 $STT0$ 位设置为 1)
- ▲4: IICS0 = 1000x110B
- ▲5: IICS0 = 1000x000B (将 $WTIM0$ 位设置为 1)^{注3}
- ▲6: IICS0 = 1000xx00B (将 $SPT0$ 位设置为 1)
- △7: IICS0 = 00000001B

- 注 1. 若要产生开始条件, 须将 $WTIM0$ 位设置为 1, 并更改产生 $INTIICA0$ 中断请求信号的时序。
 2. 将 $WTIM0$ 位清除为 0, 恢复原始设置。
 3. 若要产生停止条件, 须将 $WTIM0$ 位设置为 1, 并更改产生 $INTIICA0$ 中断请求信号的时序。

备注 ▲: 总是产生
 △: 仅当 $SPIE0 = 1$ 时产生
 x: 忽略

(ii) 当 $WTIM0 = 1$ 时



- ▲1: IICS0 = 1000x110B
- ▲2: IICS0 = 1000xx00B (将 $STT0$ 位设置为 1)
- ▲3: IICS0 = 1000x110B
- ▲4: IICS0 = 1000xx00B (将 $SPT0$ 位设置为 1)
- △5: IICS0 = 00000001B

备注 ▲: 总是产生
 △: 仅当 $SPIE0 = 1$ 时产生
 x: 忽略

(c) 开始 ~ 代码 ~ 数据 ~ 数据 ~ 停止 (扩展代码发送)

(i) 当 $WTIM0 = 0$ 时

▲1: IICS0 = 1010×110B

▲2: IICS0 = 1010×000B

▲3: IICS0 = 1010×000B (将 $WTIM0$ 位设置为 1)[#]▲4: IICS0 = 1010××00B (将 $SPT0$ 位设置为 1)

△5: IICS0 = 00000001B

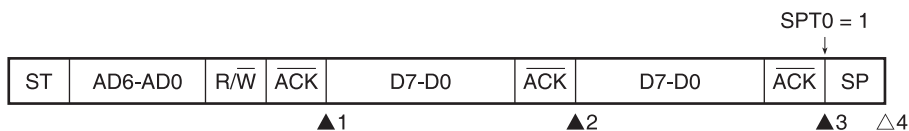
注 若要产生停止条件，须将 $WTIM0$ 位设置为 1，并更改产生 $INTIICA0$ 中断请求信号的时序。

备注

▲: 总是产生

△: 仅当 $SPIE0 = 1$ 时产生

x: 忽略

(ii) 当 $WTIM0 = 1$ 时

▲1: IICS0 = 1010×110B

▲2: IICS0 = 1010×100B

▲3: IICS0 = 1010××00B (将 $SPT0$ 位设置为 1)

△4: IICS0 = 00001001B

备注

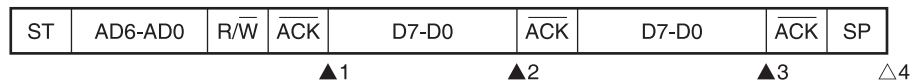
▲: 总是产生

△: 仅当 $SPIE0 = 1$ 时产生

x: 忽略

(2) 从设备操作 (从设备地址数据接收)

开始 ~ 地址 ~ 数据 ~ 数据 ~ 停止

(i) 当 $WTIMO = 0$ 时

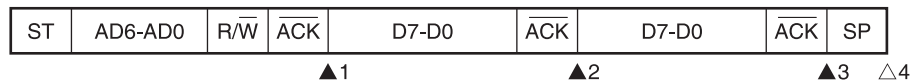
▲1: IICS0 = 0001×110B

▲2: IICS0 = 0001×000B

▲3: IICS0 = 0001×000B

△4: IICS0 = 00000001B

备注 ▲: 总是产生
 △: 仅当 $SPIE0 = 1$ 时产生
 ×: 忽略

(ii) 当 $WTIMO = 1$ 时

▲1: IICS0 = 0001×110B

▲2: IICS0 = 0001×100B

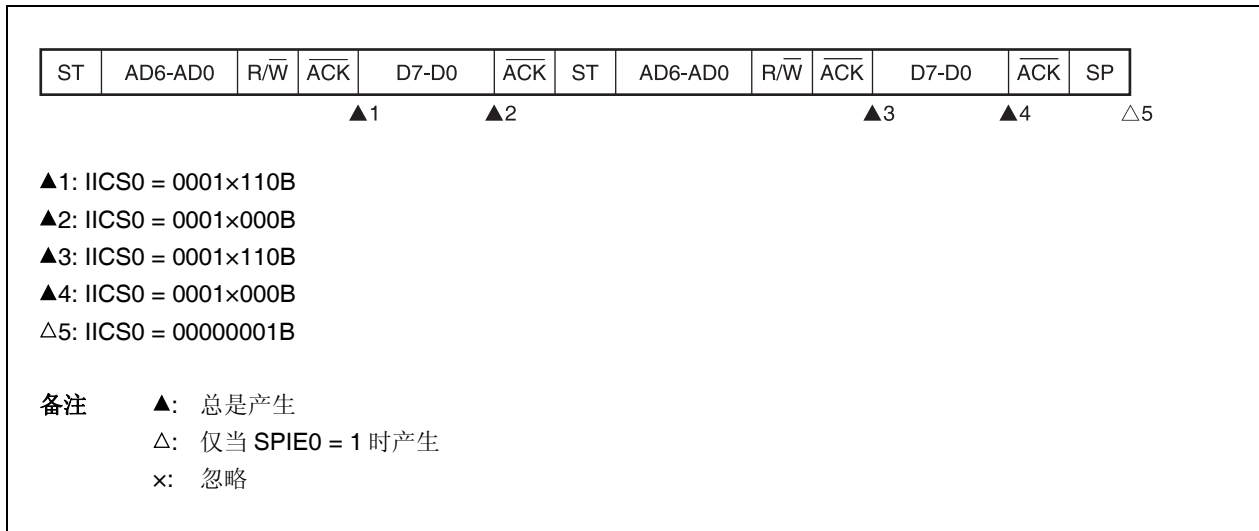
▲3: IICS0 = 0001××00B

△4: IICS0 = 00000001B

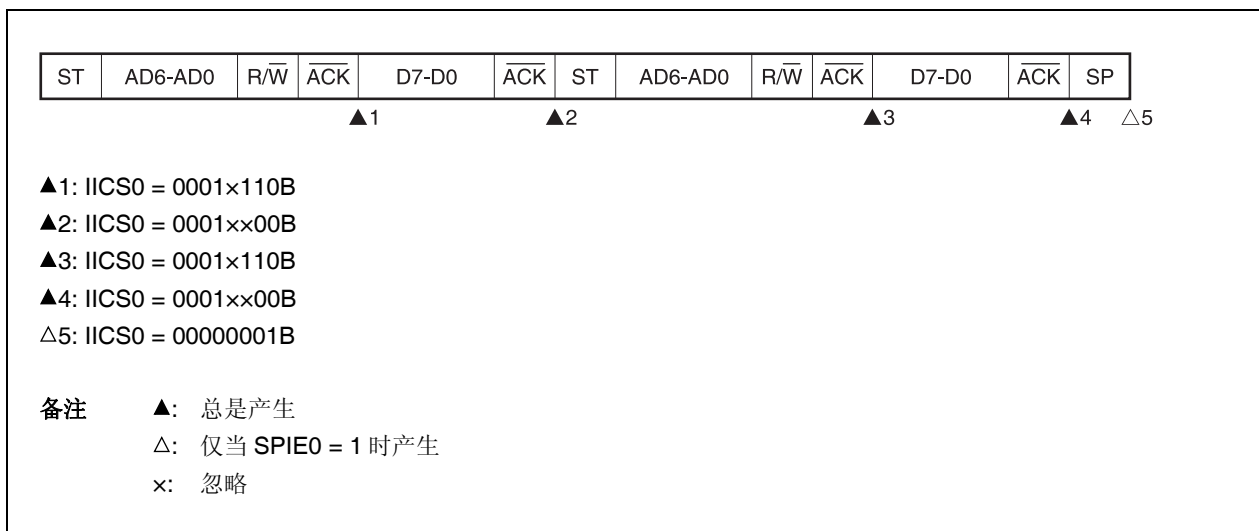
备注 ▲: 总是产生
 △: 仅当 $SPIE0 = 1$ 时产生
 ×: 忽略

(b) 开始 ~ 地址 ~ 数据 ~ 开始 ~ 地址 ~ 数据 ~ 停止

(i) 当 $WTIM0 = 0$ 时 (重启后, 与 $SVA0$ 匹配)

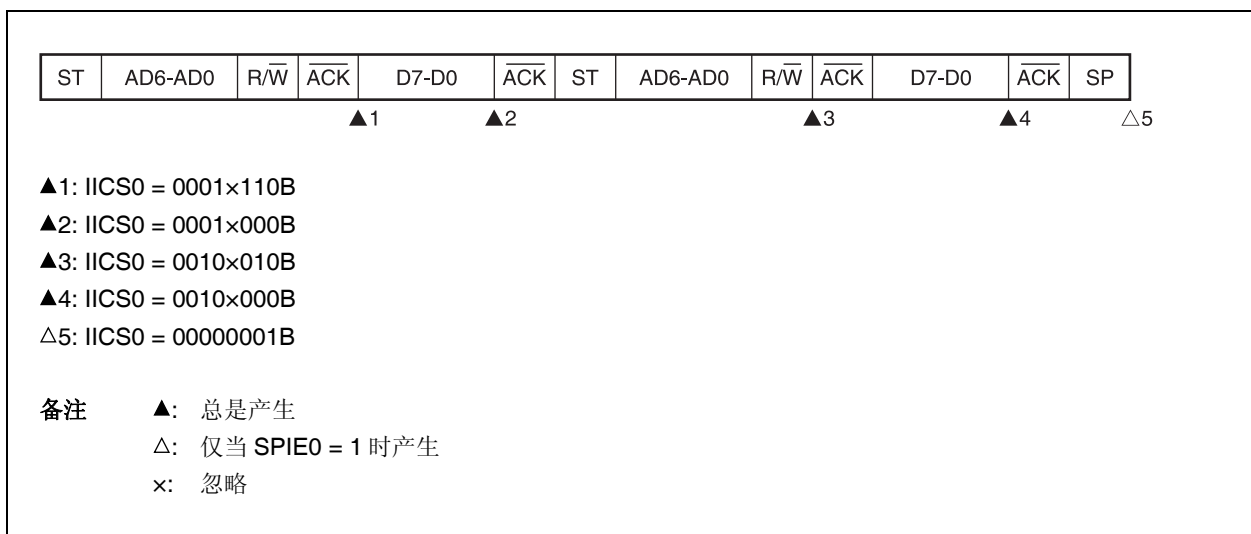


(ii) 当 $WTIM0 = 1$ 时 (重启后, 与 $SVA0$ 匹配)

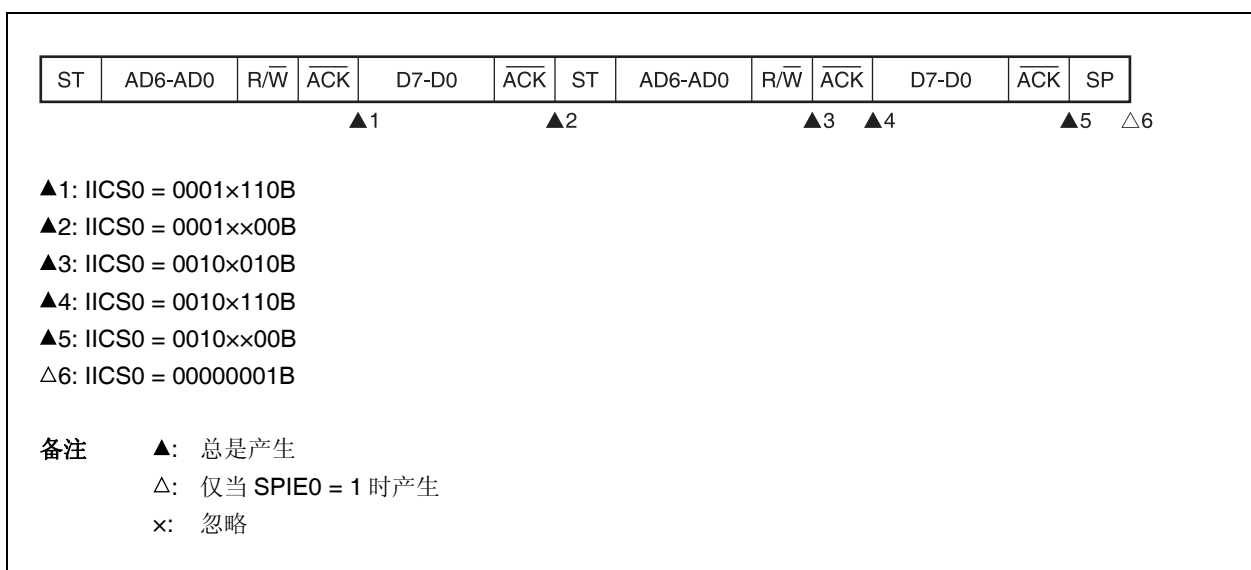


(c) 开始 ~ 地址 ~ 数据 ~ 开始 ~ 代码 ~ 数据 ~ 停止

(i) 当 $WTIMO = 0$ 时 (重启后, 地址不匹配 (扩展代码))

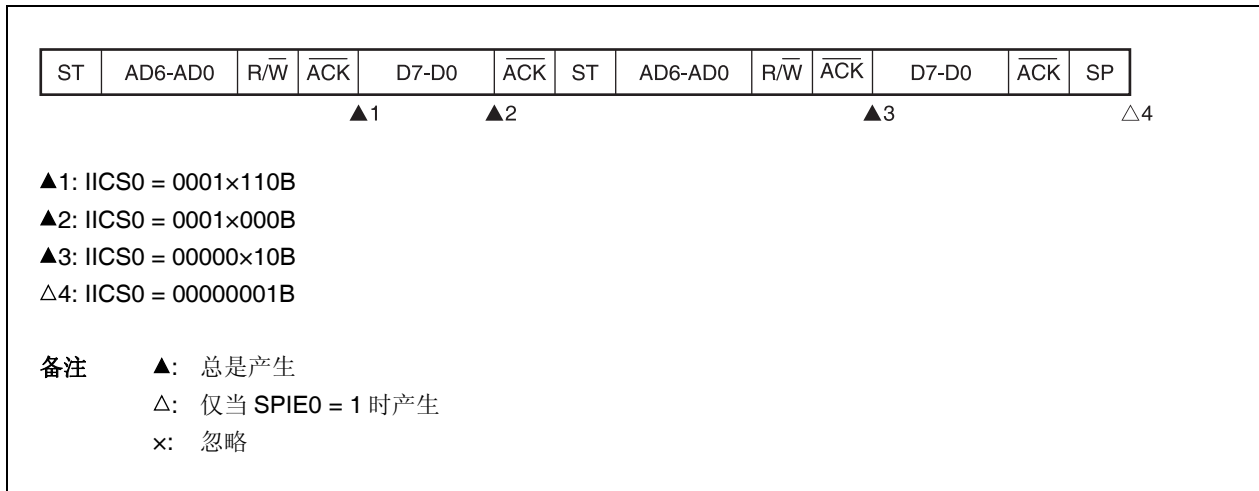


(ii) 当 $WTIMO = 1$ 时 (重启后, 地址不匹配 (扩展代码))

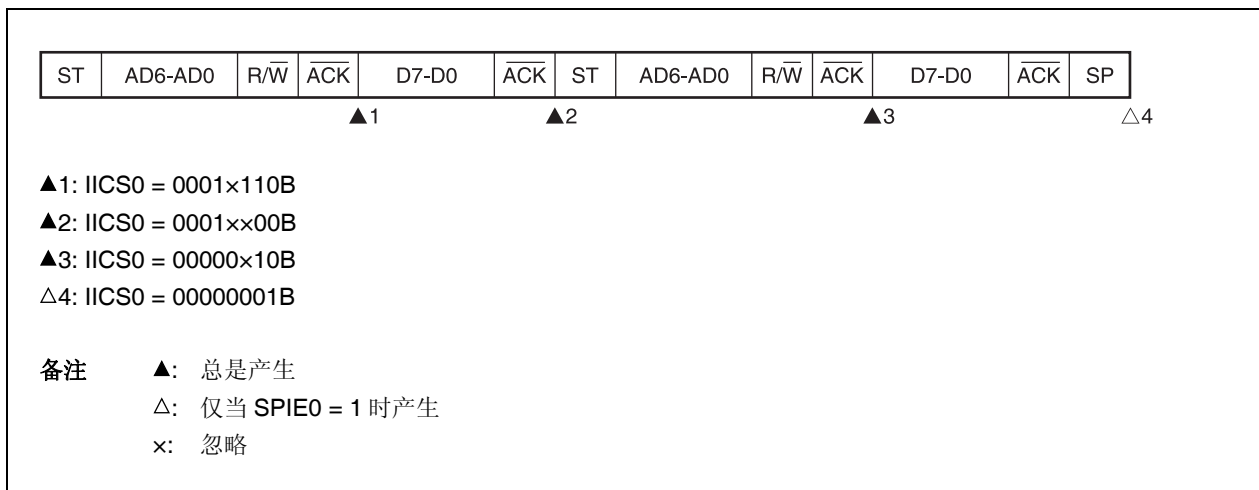


(d) 开始 ~ 地址 ~ 数据 ~ 开始 ~ 地址 ~ 数据 ~ 停止

(i) 当 $WTIMO = 0$ 时 (重启后, 地址不匹配 (扩展代码以外))

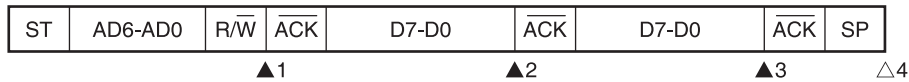


(ii) 当 $WTIMO = 1$ 时 (重启后, 地址不匹配 (扩展代码以外))



(3) 从设备操作 (扩展代码接收时)

接收到扩展代码时，器件始终参与通信。

(a) 开始 ~ 代码 ~ 数据 ~ 数据 ~ 停止**(i) 当 $WTIMO = 0$ 时**

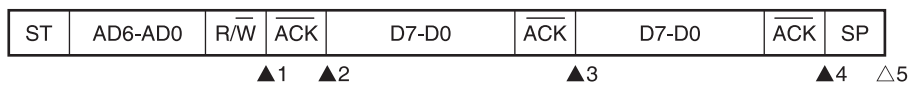
▲1: IICS0 = 0010x010B

▲2: IICS0 = 0010x000B

▲3: IICS0 = 0010x000B

△4: IICS0 = 00000001B

备注 ▲: 总是产生
 △: 仅当 $SPIE0 = 1$ 时产生
 x: 忽略

(ii) 当 $WTIMO = 1$ 时

▲1: IICS0 = 0010x010B

▲2: IICS0 = 0010x110B

▲3: IICS0 = 0010x100B

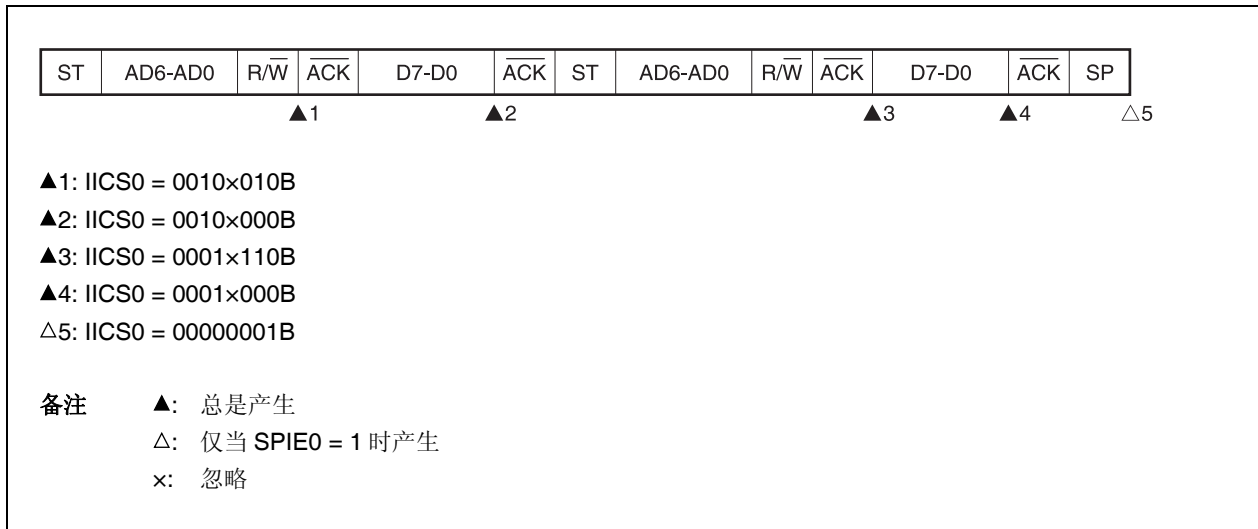
▲4: IICS0 = 0010xx00B

△5: IICS0 = 00000001B

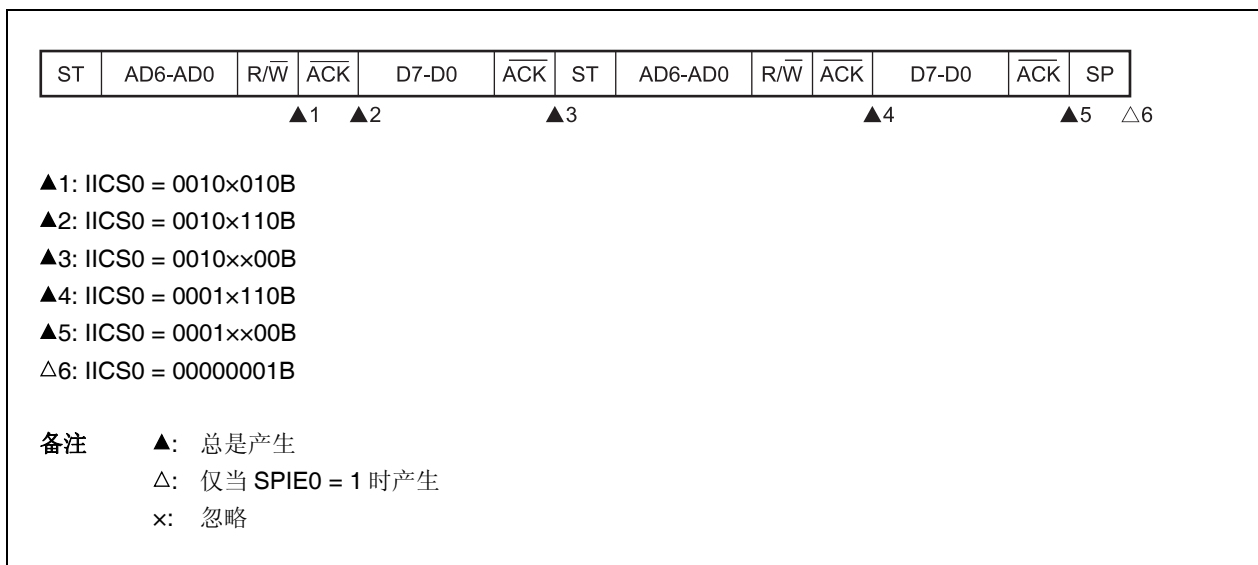
备注 ▲: 总是产生
 △: 仅当 $SPIE0 = 1$ 时产生
 x: 忽略

(b) 开始 ~ 代码 ~ 数据 ~ 开始 ~ 地址 ~ 数据 ~ 停止

(i) 当 $WTIM0 = 0$ 时 (重启后, 与 $SVA0$ 匹配)

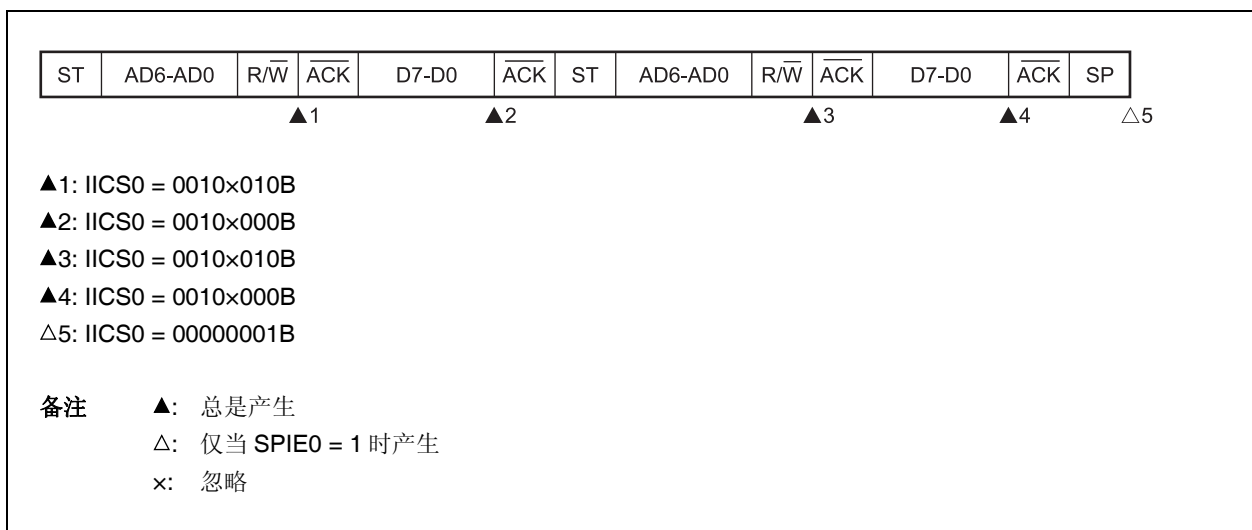


(ii) 当 $WTIM0 = 1$ 时 (重启后, 与 $SVA0$ 匹配)

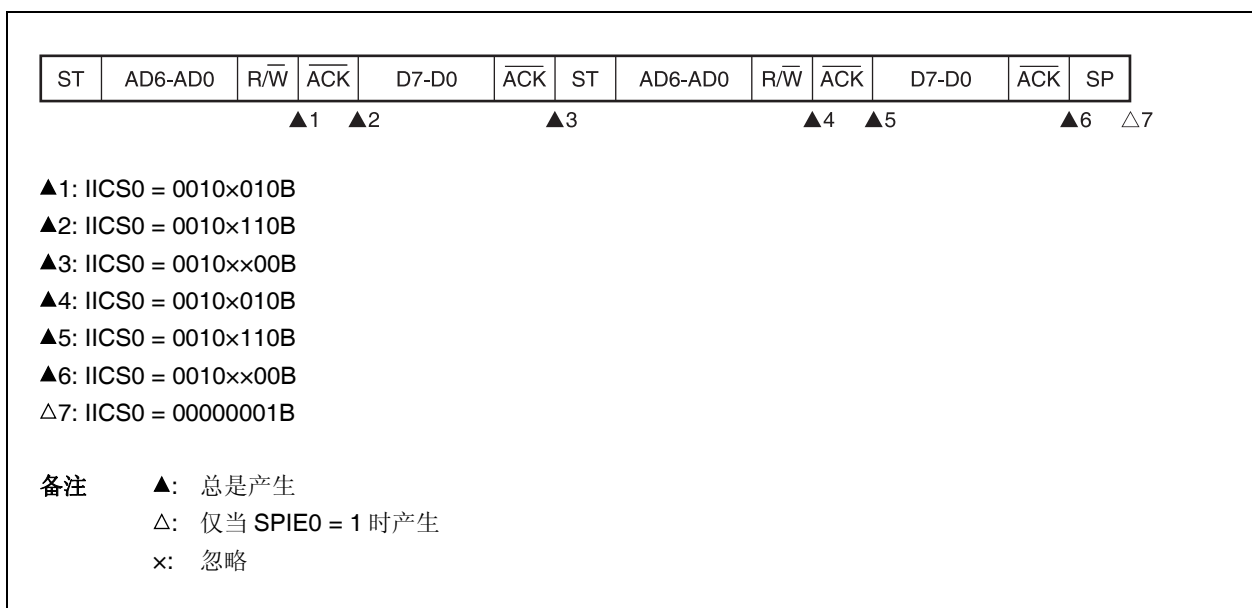


(c) 开始 ~ 代码 ~ 数据 ~ 开始 ~ 代码 ~ 数据 ~ 停止

(i) 当 $WTIMO = 0$ 时 (重启后, 扩展代码接收)

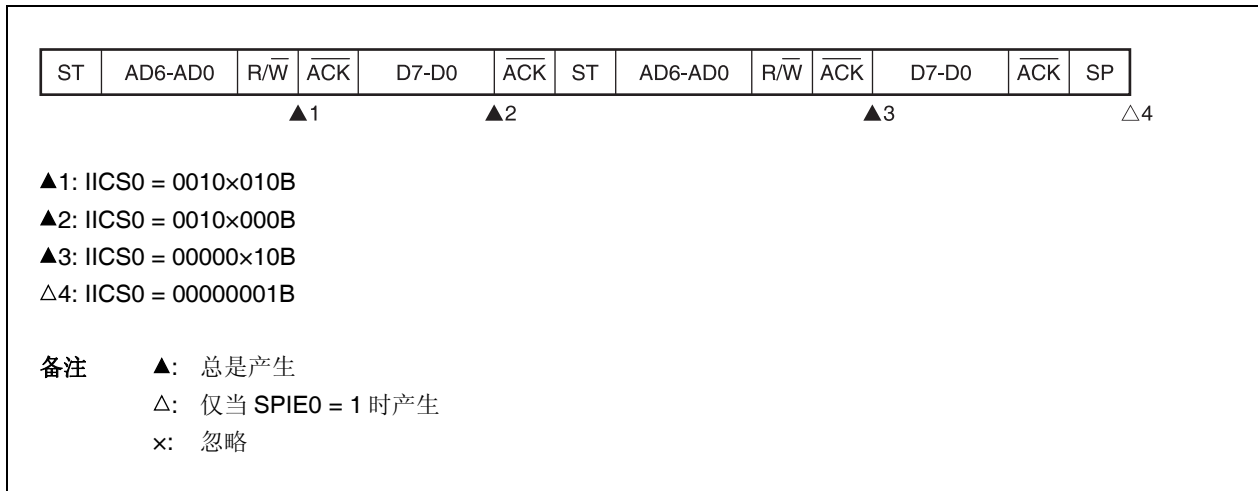


(ii) 当 $WTIMO = 1$ 时 (重启后, 扩展代码接收)

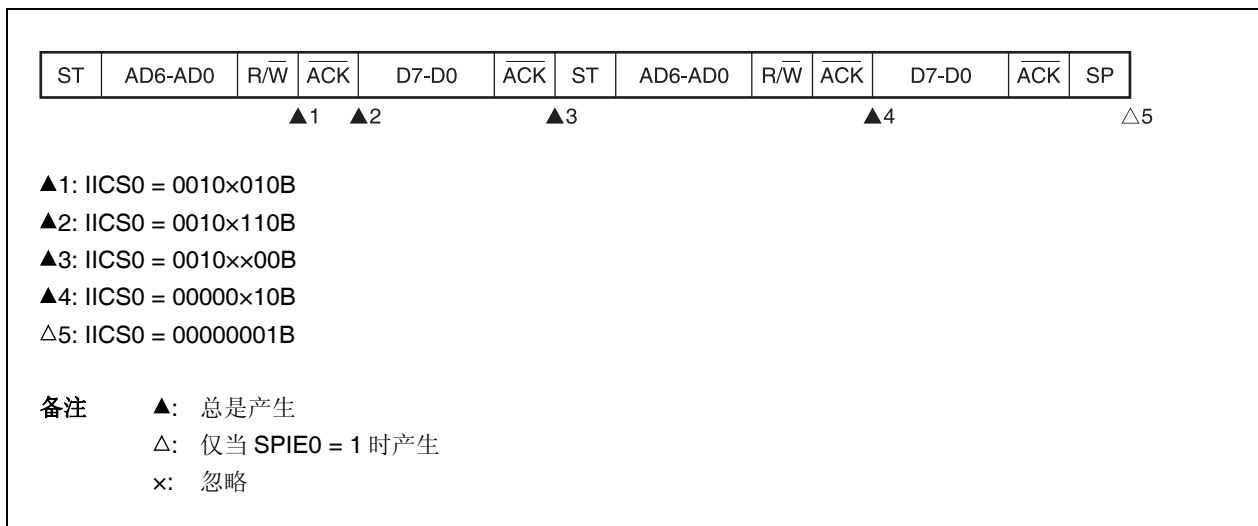


(d) 开始 ~ 代码 ~ 数据 ~ 开始 ~ 地址 ~ 数据 ~ 停止

(i) 当 $WTIMO = 0$ 时 (重启后, 地址不匹配 (扩展代码以外))

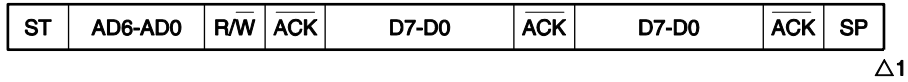


(ii) 当 $WTIMO = 1$ 时 (重启后, 地址不匹配 (扩展代码以外))



(4) 不参与通信时的操作

(a) 开始 ~ 代码 ~ 数据 ~ 数据 ~ 停止



△1: IICS0 = 00000001B

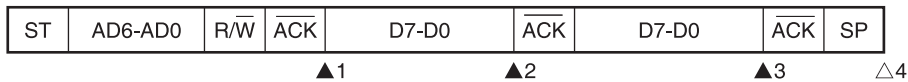
备注 △: 仅当 SPIE0 = 1 时产生

(5) 仲裁失败的操作 (仲裁失败后作为从设备操作)

当在多主设备系统中将器件用作主设备时，每当发生中断请求信号 INTIICA0 时均需读取 MSTS0 位以检查仲裁结果。

(a) 当从设备地址数据的发送期间发生仲裁失败时

(i) 当 WTIMO = 0 时



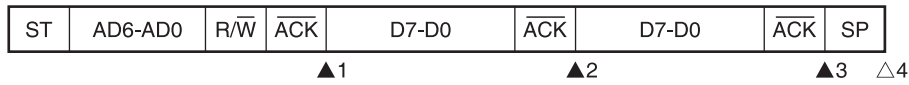
▲1: IICS0 = 0101×110B

▲2: IICS0 = 0001×000B

▲3: IICS0 = 0001×000B

△4: IICS0 = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE0 = 1 时产生
 ×: 忽略

(ii) 当 $WTIMO = 1$ 时

▲1: IICS0 = 0101×110B

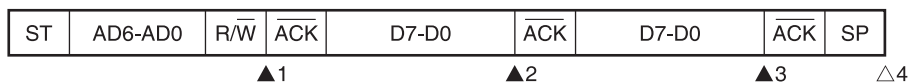
▲2: IICS0 = 0001×100B

▲3: IICS0 = 0001××00B

△4: IICS0 = 00000001B

备注 ▲: 总是产生
 △: 仅当 $SPIE0 = 1$ 时产生
 x: 忽略

(b) 当扩展代码的发送期间发生仲裁失败时

(i) 当 $WTIMO = 0$ 时

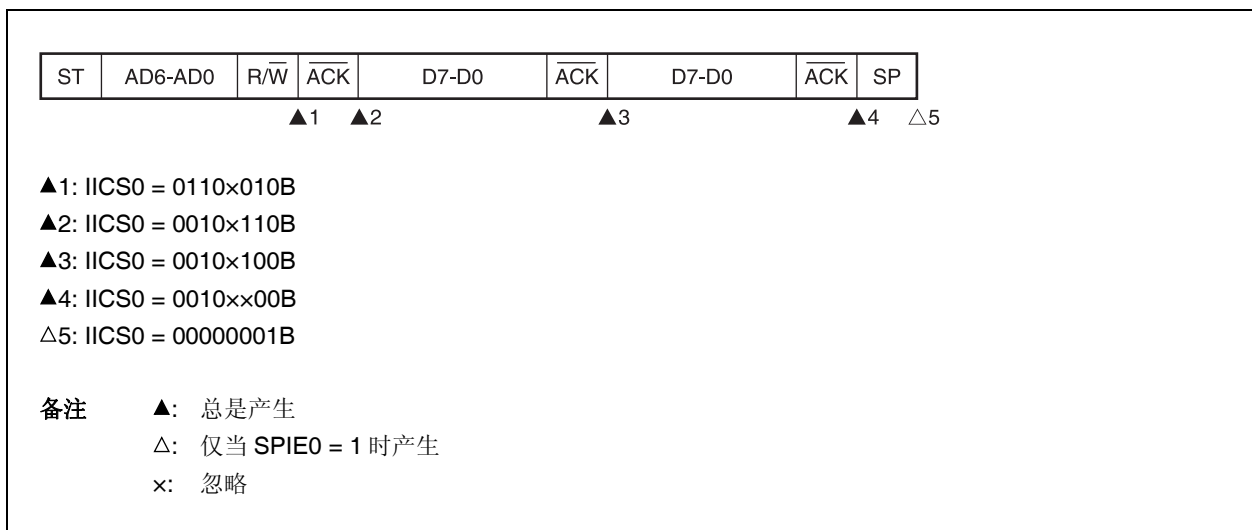
▲1: IICS0 = 0110×010B

▲2: IICS0 = 0010×000B

▲3: IICS0 = 0010×000B

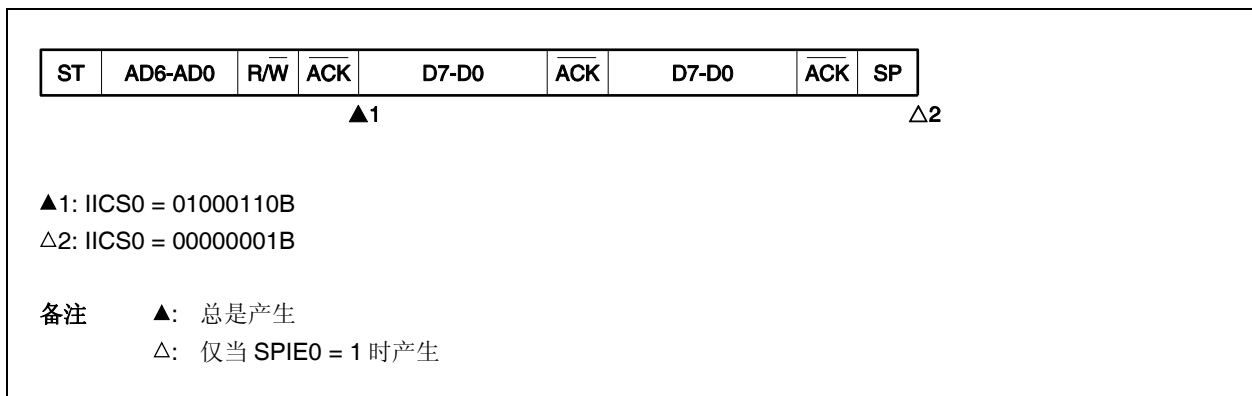
△4: IICS0 = 00000001B

备注 ▲: 总是产生
 △: 仅当 $SPIE0 = 1$ 时产生
 x: 忽略

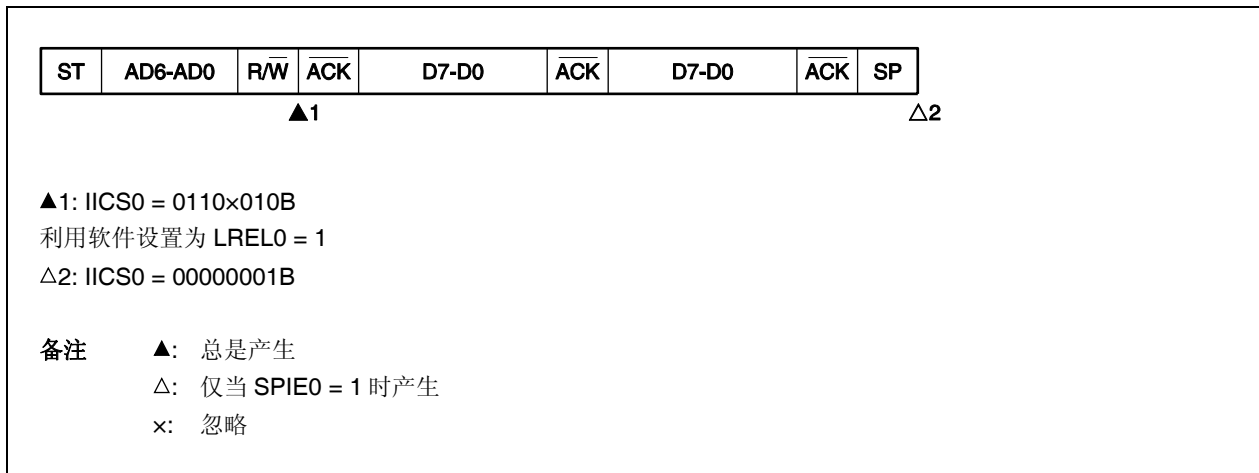
(ii) 当 $WTIMO = 1$ 时

(6) 发生仲裁失败时的操作 (仲裁失败后不参与通信)

当在多主设备系统中将器件用作主设备时，每当发生中断请求信号 INTIICA0 时均需读取 MSTS0 位以检查仲裁结果。

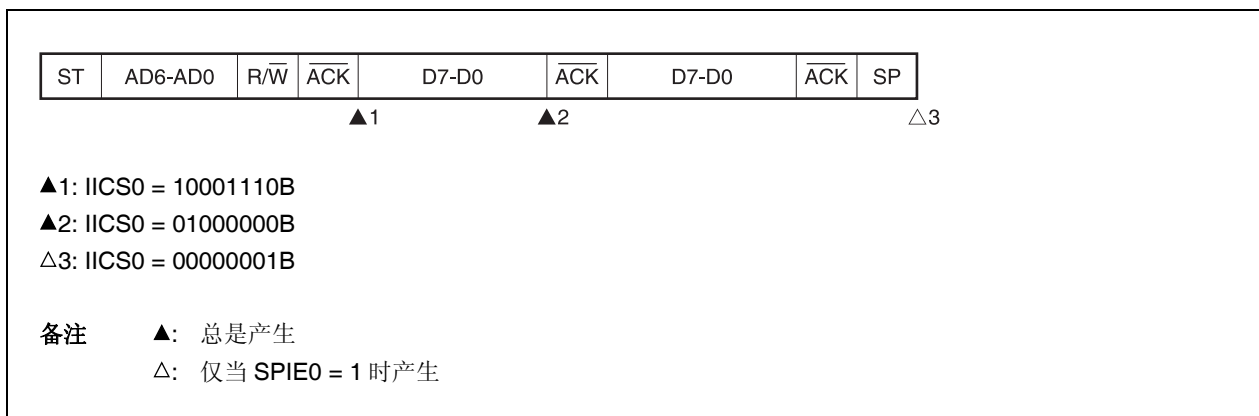
(a) 当从设备地址数据的发送期间发生仲裁失败时 (当 $WTIMO = 1$ 时)

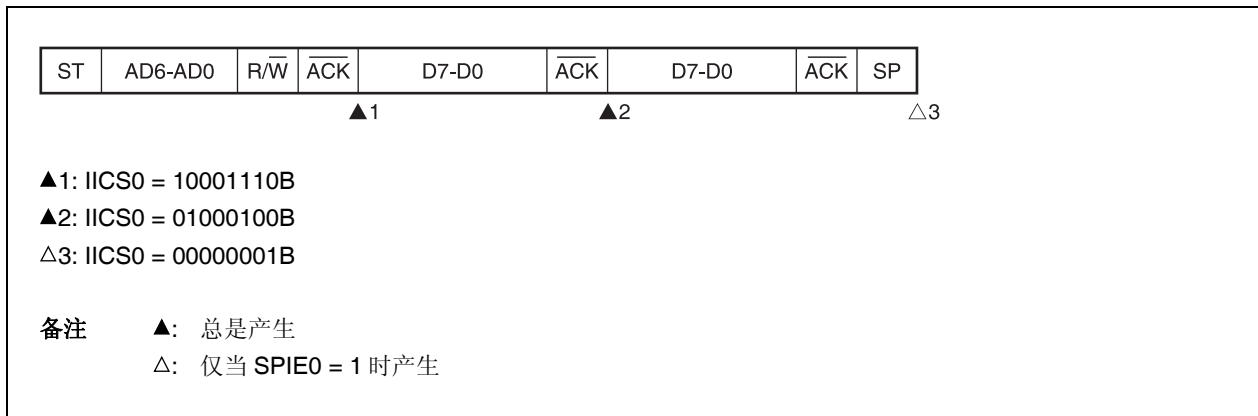
(b) 当扩展代码的发送期间发生仲裁失败时



(c) 当数据发送期间发生仲裁失败时

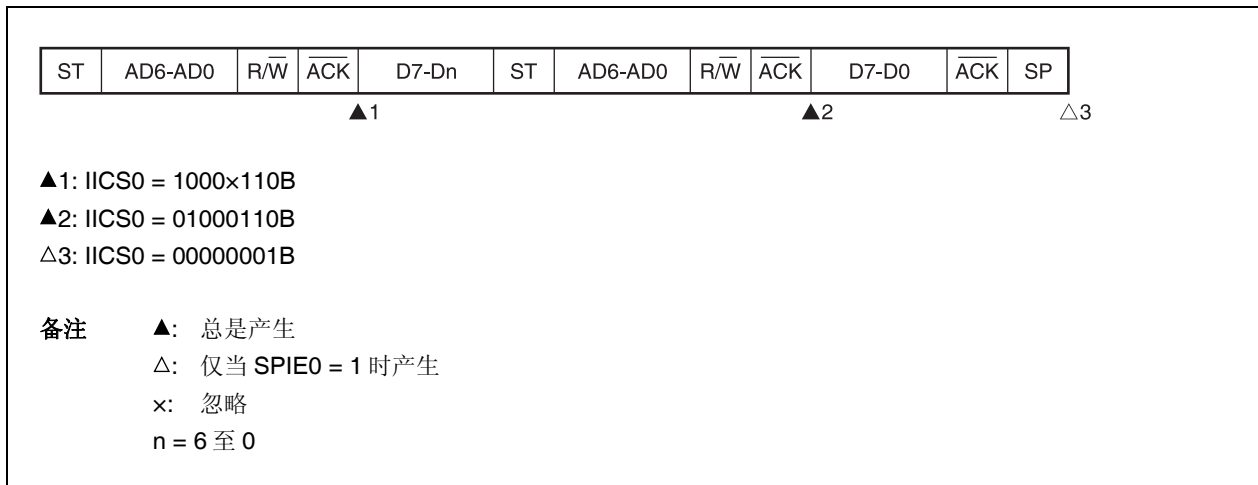
(i) 当 WTIMO = 0 时



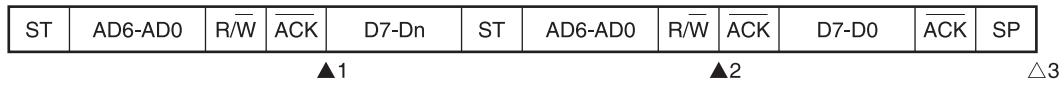
(ii) 当 $WTIMO = 1$ 时

(d) 数据传输期间由于重启条件而发生失败时

(i) 扩展代码以外 (示例: 与 SVA0 不匹配)



(ii) 扩展代码



▲1: IICS0 = 1000×110B

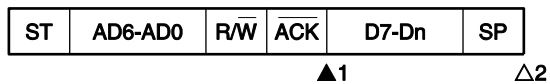
▲2: IICS0 = 01100010B

利用软件设置为 LRELO = 1

△3: IICS0 = 00000001B

备注 ▲: 总是产生
 △: 仅当 SPIE0 = 1 时产生
 x: 忽略
 n = 6 至 0

(e) 数据传输期间由于停止条件而发生失败时



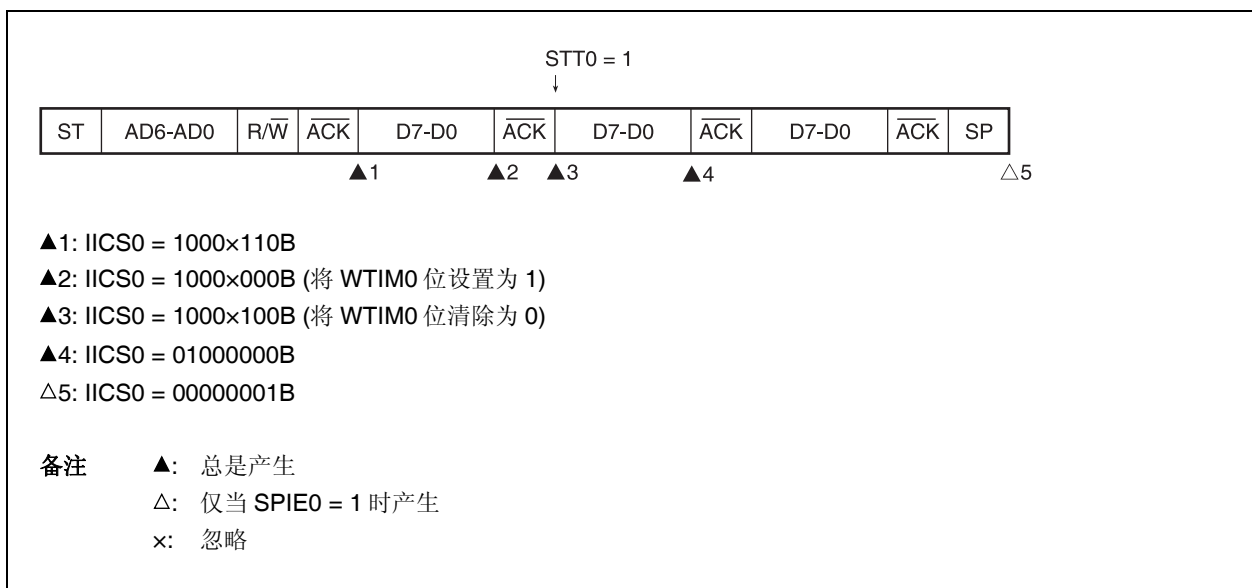
▲1: IICS0 = 10000110B

△2: IICS0 = 01000001B

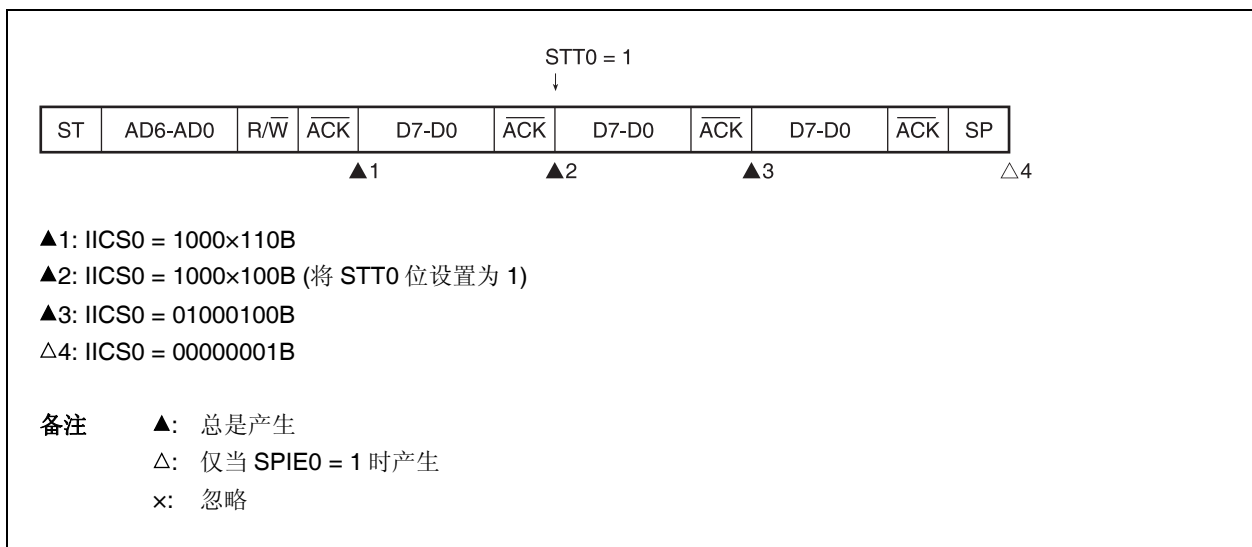
备注 ▲: 总是产生
 △: 仅当 SPIE0 = 1 时产生
 x: 忽略
 n = 6 至 0

(f) 当试图产生重启条件时，由于低电平数据发生仲裁失败

(i) 当 $WTIMO = 0$ 时

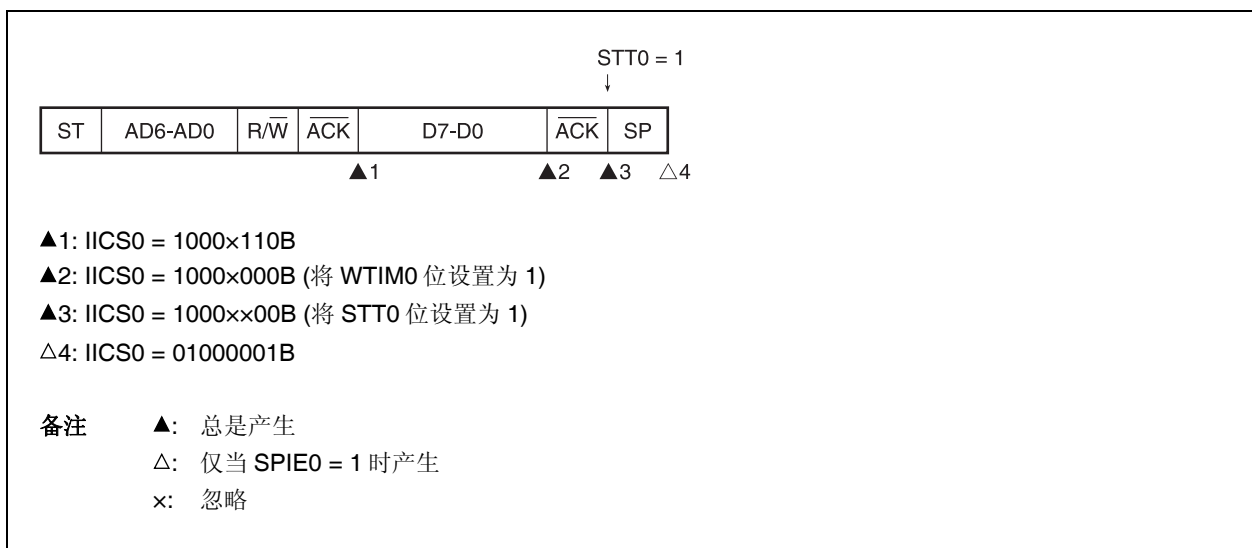


(ii) 当 $WTIMO = 1$ 时

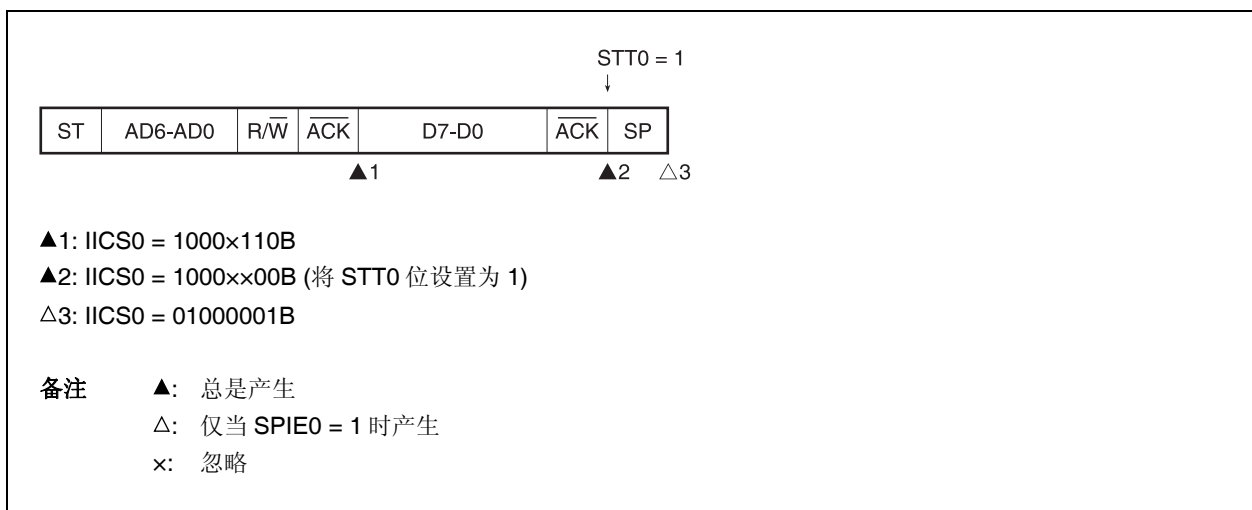


(g) 当试图产生重启条件时，由于停止条件而发生仲裁失败

(i) 当 $WTIM0 = 0$ 时

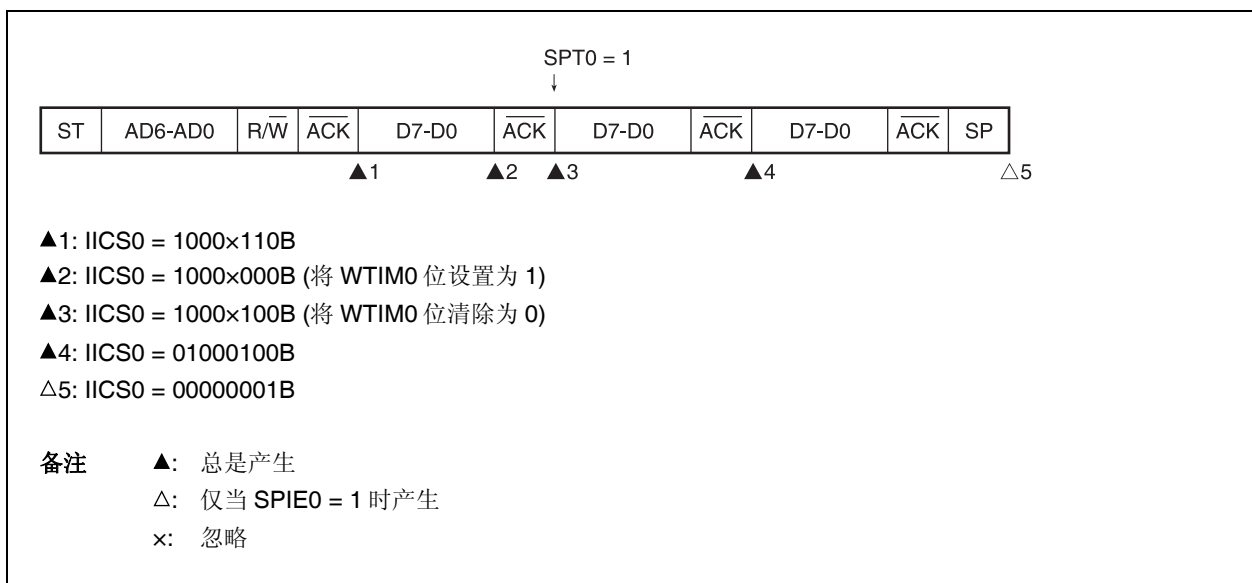


(ii) 当 $WTIM0 = 1$ 时

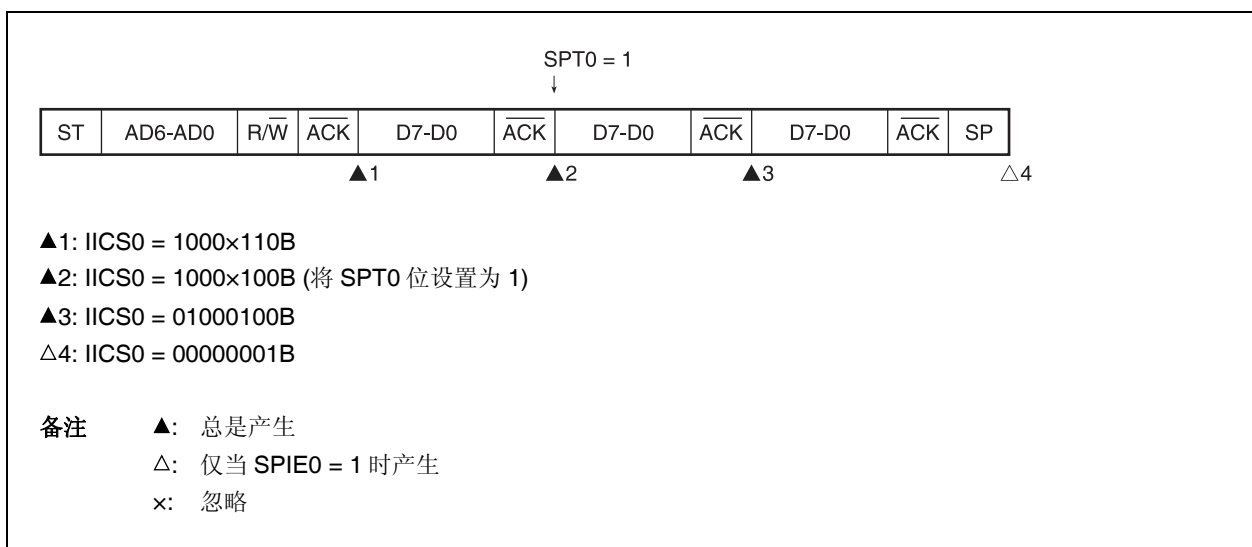


(h) 当试图产生停止条件时，由于低电平数据发生仲裁失败

(i) 当 $WTIM0 = 0$ 时



(ii) 当 $WTIM0 = 1$ 时



17.6 时序图

当使用 I²C 总线模式时，主设备通过串行总线输出一个地址，在多个从设备中选择其中一个作为自己的通信对象。

输出从设备地址以后，主设备发送 TRC0 位(IICA 状态寄存器 0 (IICS0)的位 3)指定数据传输方向，然后，开始与从设备进行串行通信。

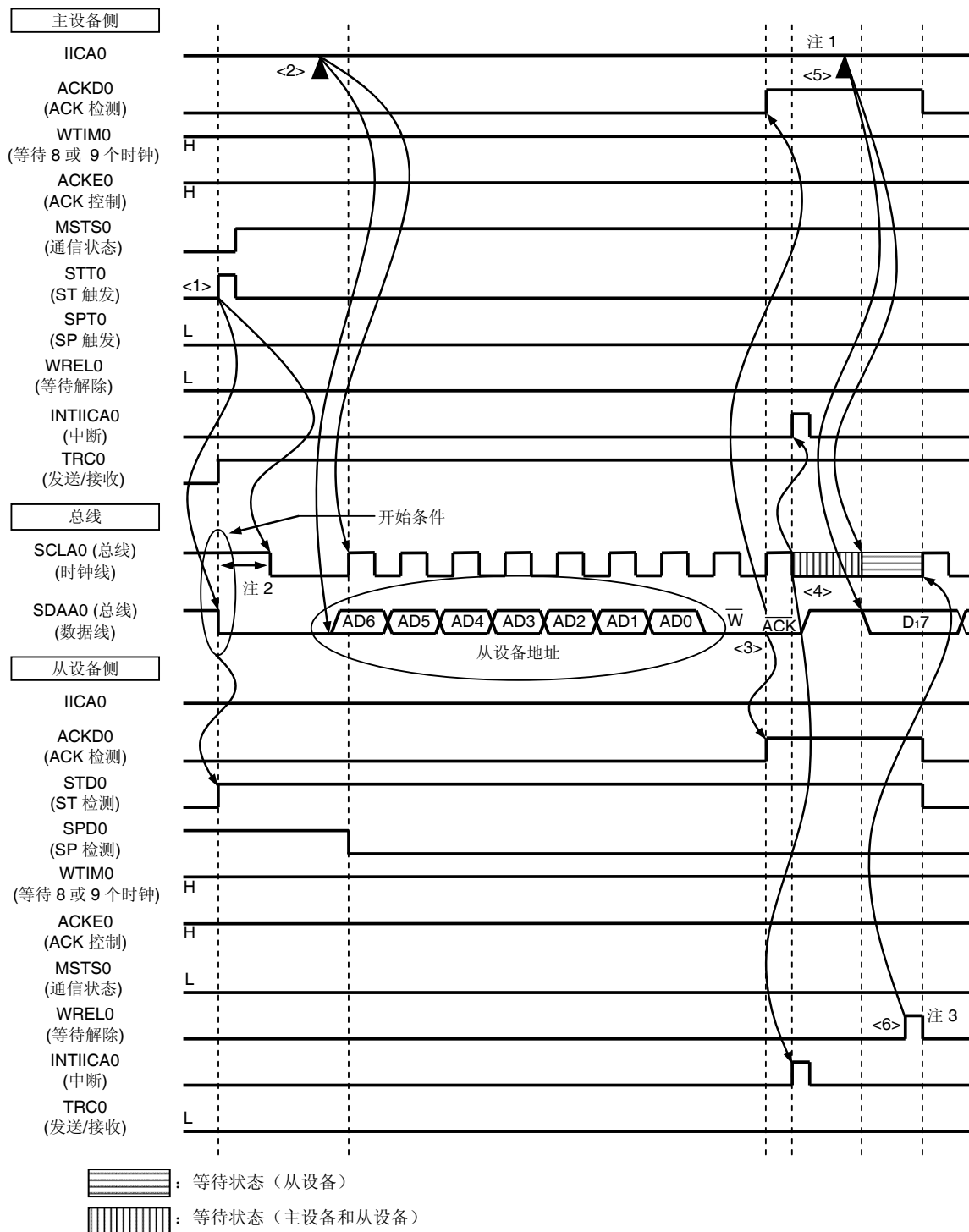
数据通信时序图如图 17-33 和 17-34 所示。

IICA 移位寄存器 0(IICA0)的移位操作和串行时钟 (SCLA0) 的下降沿同步。发送数据被传送到 SO 锁存器并经由 SDAA0 引脚输出 (MSB 先行)。

由 SDAA0 引脚输入的数据在 SCLA0 的上升沿被捕捉至 IICA0。

图 17-33. 主设备→从设备通信示例
(主设备: 选择 9 时钟等待, 从设备: 选择 9 时钟等待) (1/4)

(1) 开始条件 ~ 地址 ~ 数据



- 注 1. 将数据写入 IICA0, 不设置 WRELO 位, 以在主设备发送期间取消等待状态。
- 注 2. 确保从 SDAA0 引脚信号下降到 SCLA0 引脚信号下降的时间在指定标准模式时至少为 4.0 μ s, 在指定快速模式时至少为 0.6 μ s。
- 注 3. 要取消从设备等待状态, 须将“FFH”写入 IICA0, 或者设置 WRELO 位。

图 17-33 中，(1) 开始条件~地址~数据中的 <1>至<6> 的意义解释如下。

- <1> 开始条件触发由主设备设置 (STT0 = 1)，当总线数据线变成低电平 (SDAA0 = 0)时将产生一个开始条件 (SDAA0 = 0 和 SCLA0 = 1)。当接下来检测到开始条件时，主设备将进入主设备通信状态 (MSTS0 = 1)。在经过保持时间之后，一旦总线时钟线变成低电平 (SCLA0 = 0)，主设备即可进行通信。
- <2> 主设备将地址 + W (发送) 写入 IICA 移位寄存器 0 (IICA0)，并发送从设备地址。
- <3> 如果收到的地址与从设备地址相匹配^注，则从设备将通过硬件向主设备发送 ACK。主设备在第 9 个时钟的上升沿检测 ACK (ACKD0 = 1)。
- <4> 主设备在第 9 个时钟的下降沿产生一个中断(INTIICA0: 地址发送结束)，和发送的地址相匹配的地址的从设备也产生一个中断(INTIICA0: 地址匹配)。地址匹配时，主设备和从设备还设置等待状态(SCLA0 = 0)^注。
- <5> 主设备将待发送的数据写入 IICA0 寄存器，并解除由主设备设置的等待状态。
- <6> 如果从设备解除等待状态(WRELO = 1)，则主设备将开始向从设备传输数据。

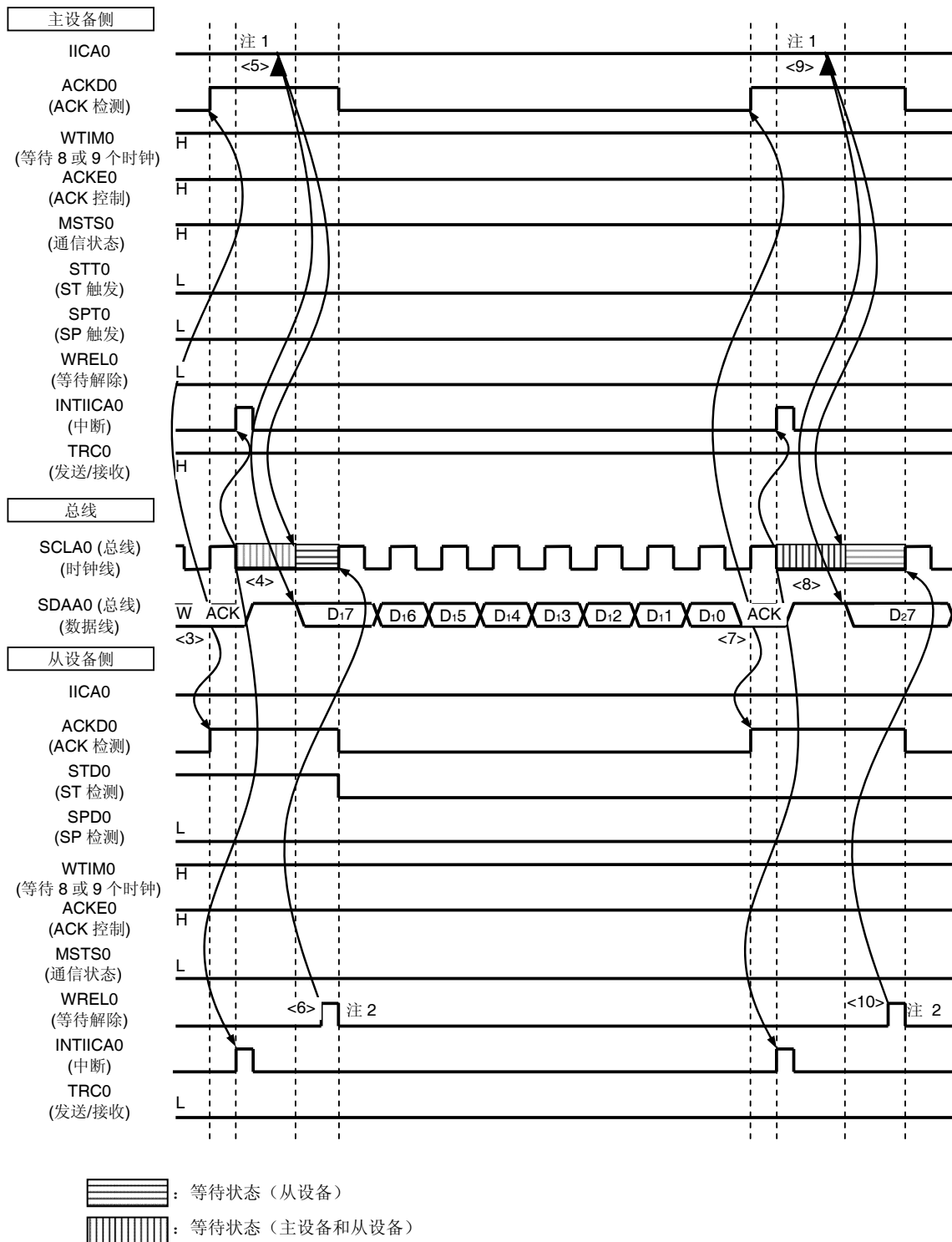
注 如果发送的地址和从设备的地址不匹配，则从设备不会向主设备返回 ACK(NACK: SDAA0 = 1)。从设备也不会产生 INTIICA0 中断 (地址匹配)，不会设置等待状态。但是，主设备将产生 INTIICA0 中断 (地址发送结束)，无论是收到 ACK 还是 NACK。

备注 图 17-33 中，<1>至<15>代表利用 I²C 总线进行数据通信的整个步骤。

图 17-33 (1)开始条件~ 地址~ 数据表示从<1> 到 <6> 的处理步骤，图 17-33 (2)地址 ~ 数据 ~ 数据表示从<3>到<10>的处理步骤，图 17-33 (3) 数据 ~ 数据 ~ 停止条件则表示从<7>到<15>的处理步骤。

图 17-33. 主设备→从设备通信示例
(主设备: 选择 9 时钟等待, 从设备: 选择 9 时钟等待) (2/4)

(2) 地址 ~ 数据 ~ 数据



- 注 1. 将数据写入 IICA0, 不设置 WRELO 位, 以在主设备发送期间取消等待状态。
- 2. 要取消从设备等待状态, 须将“FFH”写入 IICA0, 或者设置 WRELO 位。

图 17-33 中，(2) 地址~数据~数据中的 <3>至<10> 的意义解释如下。

- <3> 如果收到的地址与从设备地址相匹配^注，则从设备将通过硬件向主设备发送 ACK。主设备在第 9 个时钟的上升沿检测 ACK (ACKD0 = 1)。
- <4> 主设备在第 9 个时钟的下降沿产生一个中断(INTIICA0: 地址发送结束)，和发送的地址相匹配的地址的从设备也产生一个中断(INTIICA0: 地址匹配)。地址匹配时，主设备和从设备还设置等待状态(SCLA0 = 0)^注。
- <5> 主设备将待发送的数据写入 IICA 移位寄存器 0 (IICA0)，并解除由主设备设置的等待状态。
- <6> 如果从设备解除等待状态(WRELO = 1)，则主设备将开始向从设备传输数据。
- <7> 数据传输完成时，从设备将通过硬件向主设备发送 ACK。主设备在第 9 个时钟的上升沿检测 ACK (ACKD0 = 1)。
- <8> 主设备和从设备在第 9 个时钟的下降沿设置等待状态 (SCLA0 = 0)，而且主设备和从设备都将产生一个中断 (INTIICA0: 传送结束)。
- <9> 主设备将待发送的数据写入 IICA0 寄存器，并解除由主设备设置的等待状态。
- <10> 从设备读取接收到的数据，并解除等待状态(WRELO = 1)。主设备开始向从设备发送数据。

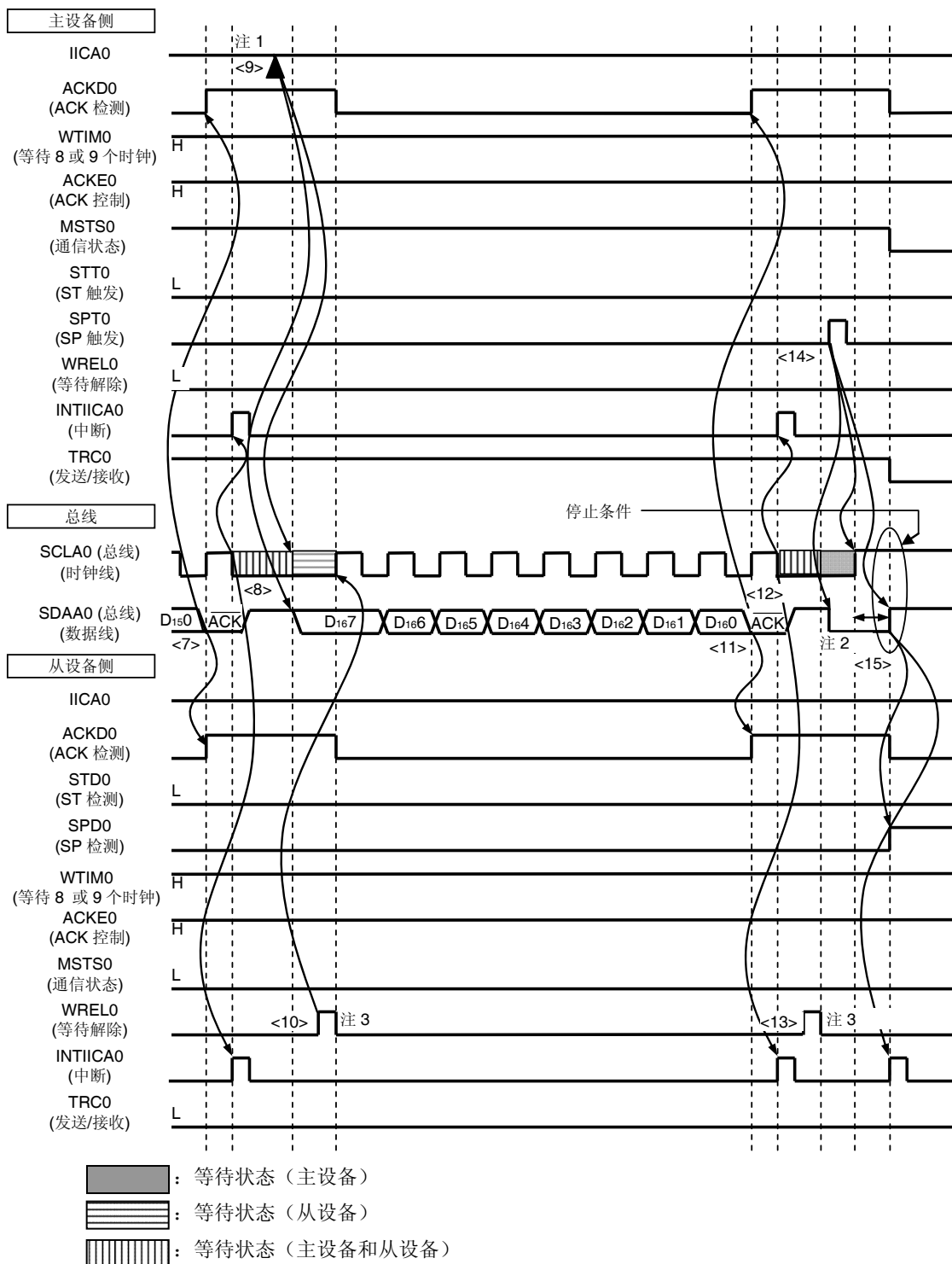
注 如果发送的地址和从设备的地址不匹配，则从设备不会向主设备返回 ACK(NACK: SDAA0 = 1)。从设备也不会产生 INTIICA0 中断（地址匹配），不会设置等待状态。但是，主设备将产生 INTIICA0 中断（地址发送结束），无论是收到 ACK 还是 NACK。

备注 图 17-33 中，<1>至<15>代表利用 I²C 总线进行数据通信的整个步骤。

图 17-33 (1)开始条件~ 地址~ 数据表示从<1> 到 <6> 的处理步骤，图 17-33 (2)地址 ~ 数据 ~ 数据表示从 <3>到<10>的处理步骤，图 17-33 (3) 数据 ~ 数据 ~ 停止条件则表示从<7>到<15>的处理步骤。

图 17-33. 主设备→从设备通信示例
(主设备: 选择 9 时钟等待, 从设备: 选择 9 时钟等待) (3/4)

(3) 数据 ~ 数据 ~ 停止条件



- 注 1. 将数据写入 IICA0, 不设置 WRELO 位, 以在主设备发送期间取消等待状态。
- 注 2. 确保在产生停止条件之后, 从 SCLA0 引脚信号上升到产生停止条件的时间在指定标准模式时至少为 4.0 μs, 在指定快速模式时至少为 0.6 μs。
- 注 3. 要取消从设备等待状态, 须将“FFH”写入 IICA0, 或者设置 WRELO 位。

图 17-33 中，(3) 数据~数据~停止条件中的 <7>至<15> 的意义解释如下。

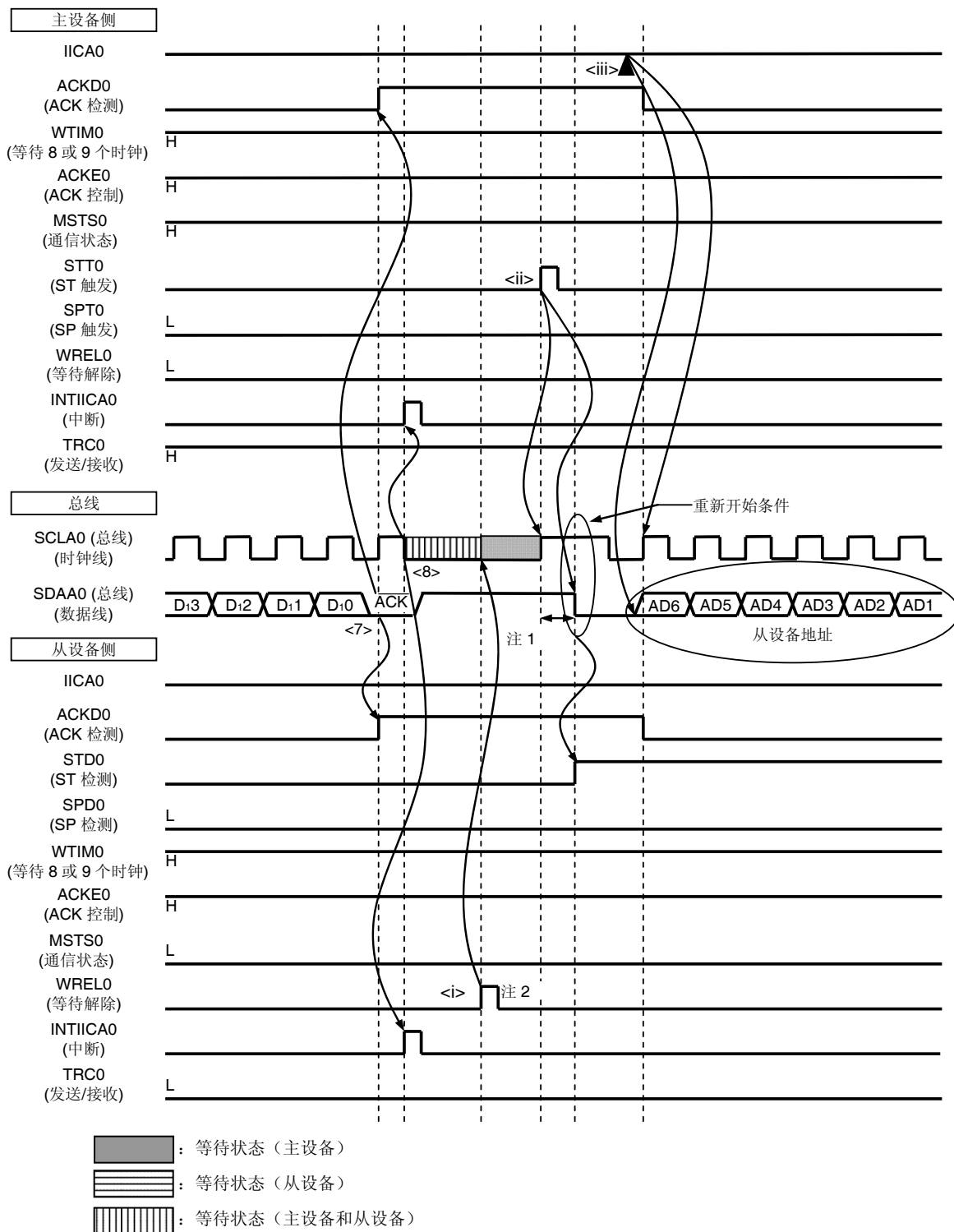
- <7> 数据传输完成时，从设备将通过硬件向主设备发送 ACK。主设备在第 9 个时钟的上升沿检测 ACK (ACKD0 = 1)。
- <8> 主设备和从设备在第 9 个时钟的下降沿设置等待状态 (SCLA0 = 0)，而且主设备和从设备都将产生一个中断 (INTIICA0: 传送结束)。
- <9> 主设备将待发送的数据写入 IICA 移位寄存器 0 (IICA0)，并解除由主设备设置的等待状态。
- <10> 从设备读取接收到的数据，并解除等待状态(WRELO = 1)。主设备开始向从设备发送数据。
- <11> 数据传输完成时，从设备将通过硬件向主设备发送 ACK。主设备在第 9 个时钟的上升沿检测 ACK (ACKD0 = 1)。
- <12> 主设备和从设备在第 9 个时钟的下降沿设置等待状态 (SCLA0 = 0)，而且主设备和从设备都将产生一个中断 (INTIICA0: 传送结束)。
- <13> 从设备读取接收到的数据，并解除等待状态(WRELO = 1)。
- <14> 设置了停止条件触发后，总线数据线被清零(SDAA0 = 0)，并设置总线时钟线(SCLA0 = 1)。经过了停止条件设置时间后，通过设置总线数据线(SDAA0 = 1)来产生停止条件。
- <15> 在产生停止条件时，从设备将检测停止条件并产生一个中断(INTIICA0: 停止条件)。

备注 图 17-33 中，<1>至<15>代表利用 I²C 总线进行数据通信的整个步骤。

图 17-33 (1)开始条件~ 地址~ 数据表示从<1> 到 <6> 的处理步骤，图 17-33 (2)地址 ~ 数据 ~ 数据表示从<3>到<10>的处理步骤，图 17-33 (3) 数据 ~ 数据 ~ 停止条件则表示从<7>到<15>的处理步骤。

图 17-33. 主设备→从设备通信示例
(主设备：选择 9 时钟等待， 从设备：选择 9 时钟等待) (4/4)

(4) 数据 ~ 重启条件 ~ 地址



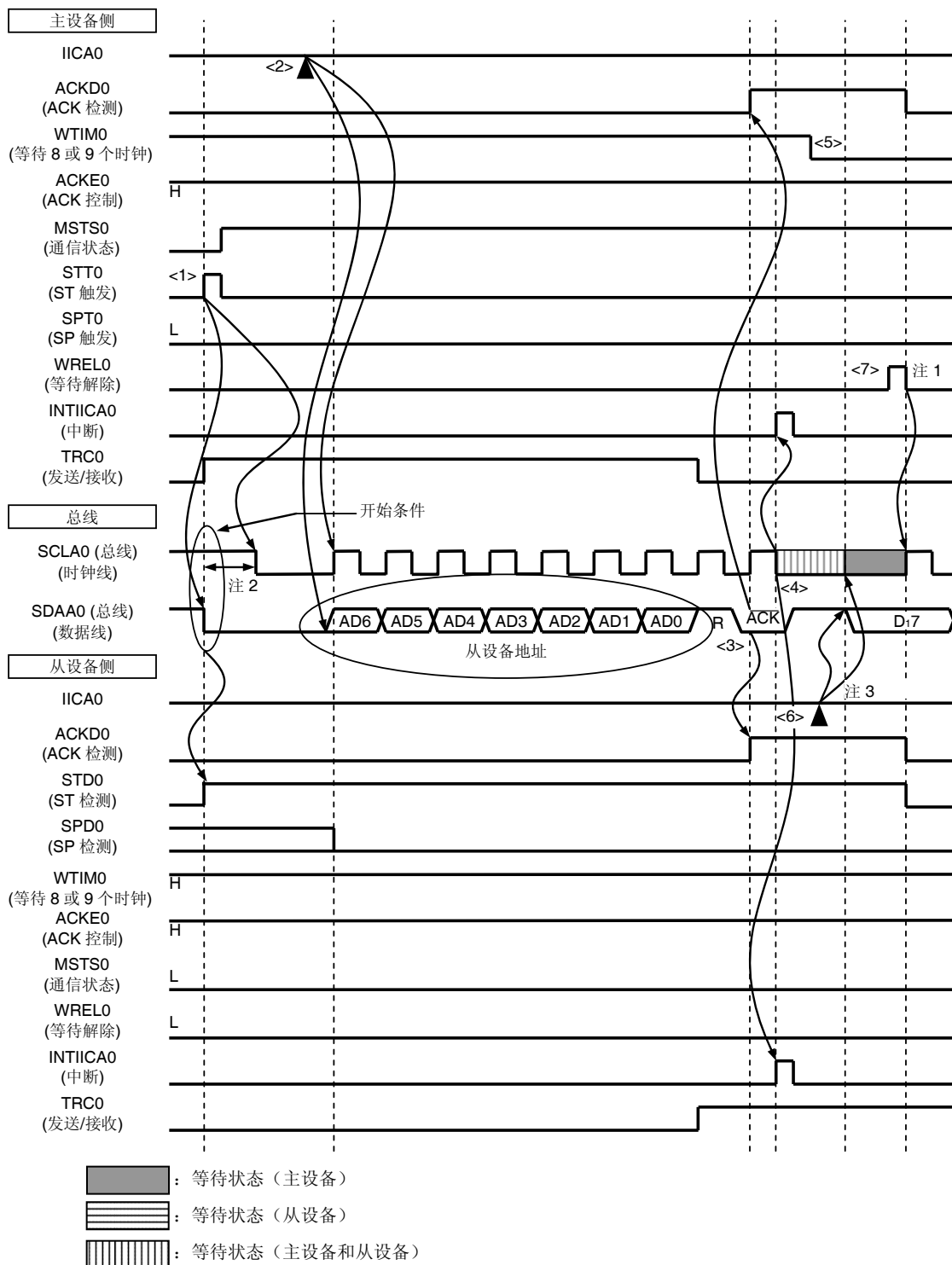
注 1. 确保在产生重启条件之后，从 SCLA0 引脚信号上升到产生开始条件的时间在指定标准模式时至少为 4.7 μs，在指定快速模式时至少为 0.6 μs。
2. 要取消从设备等待状态，须将“FFH”写入 IICA0，或者设置 WRELO 位。

以下说明图 17-33 中(4) 数据 ~ 重启条件 ~ 地址的操作。在执行第<7>和第<8>步中的操作之后，将执行第<i>至第<ii>步中的操作。这些步骤使处理返回第<3>步，即数据发送步骤。

- <7> 数据传输完成时，从设备将通过硬件向主设备发送 ACK。主设备在第 9 个时钟的上升沿检测 ACK (ACKD0 = 1)。
- <8> 主设备和从设备在第 9 个时钟的下降沿设置等待状态 (SCLA0 = 0)，而且主设备和从设备都将产生一个中断 (INTIICA0: 传送结束)。
- <i> 从设备读取接收到的数据，并解除等待状态(WRELO = 1)。
- <ii> 开始条件触发再次由主设备设置(STT0 = 1)，在经过重启条件设置时间之后，当总线时钟线变成高电平 (SCLA0 = 1)且总线数据线变成低电平(SDAA0 = 0)时，将产生一个开始条件 (SDAA0 = 0 和 SCLA0 = 1)。当接下来检测到开始条件时，在经过保持时间之后，一旦总线时钟线变成低电平 (SCLA0 = 0)，主设备即可通信。
- <iii> 主设备将地址 + R/W (发送) 写入 IICA 移位寄存器 0 (IICA0)，以发送从设备地址。

图 17-34. 从设备→主设备通信示例
(主设备：选择 8 时钟等待， 从设备：选择 9 时钟等待) (1/3)

(1) 开始条件 ~ 地址 ~ 数据



- 注 1. 要取消主设备等待状态，须将“FFH”写入 IICA0，或者设置 WRELO 位。
- 2. 确保从 SDAA0 引脚信号下降到 SCLA0 引脚信号下降的时间在指定标准模式时至少为 4.0 μs，在指定快速模式时至少为 0.6 μs。
- 3. 将数据写入 IICA0，不设置 WRELO 位，以在从设备发送期间取消等待状态。

图 17-34 中，(1) 开始条件~地址~数据中的 <1>至<7> 的意义解释如下。

- <1> 开始条件触发由主设备设置 (STT0 = 1)，当总线数据线变成低电平 (SDAA0)时将产生一个开始条件 (即 SCLA0 = 1 使 SDAA0 从 1 变为 0)。当接下来检测到开始条件时，主设备将进入主设备通信状态 (MSTS0 = 1)。在经过保持时间之后，一旦总线时钟线变成低电平 (SCLA0 = 0)，主设备即可进行通信。
- <2> 主设备将地址 + R (接收) 写入 IICA 移位寄存器 0 (IICA0)，并发送从设备地址。
- <3> 在从设备中，如果收到的地址与从设备的地址 (SVA0 值) 相匹配^注，则从设备将通过硬件向主设备发送一个 ACK。主设备在第 9 个时钟的上升沿检测 ACK (ACKD0 = 1)。
- <4> 主设备在第 9 个时钟的下降沿产生一个中断 (INTIICA0: 地址发送结束)。地址与发送的从设备地址相匹配的从设备将设置等待状态(SCLA0 = 0)，并产生一个中断(INTIICA0: 地址匹配)^注。
- <5> 主设备设置等待状态的时序变成第 8 个时钟 (WTIM0 = 0)。
- <6> 从设备将待发送的数据写入 IICA0 寄存器，并解除由从设备设置的等待状态。
- <7> 如果主设备解除等待状态(WREL0 = 1)，则从设备将开始向主设备传送数据。

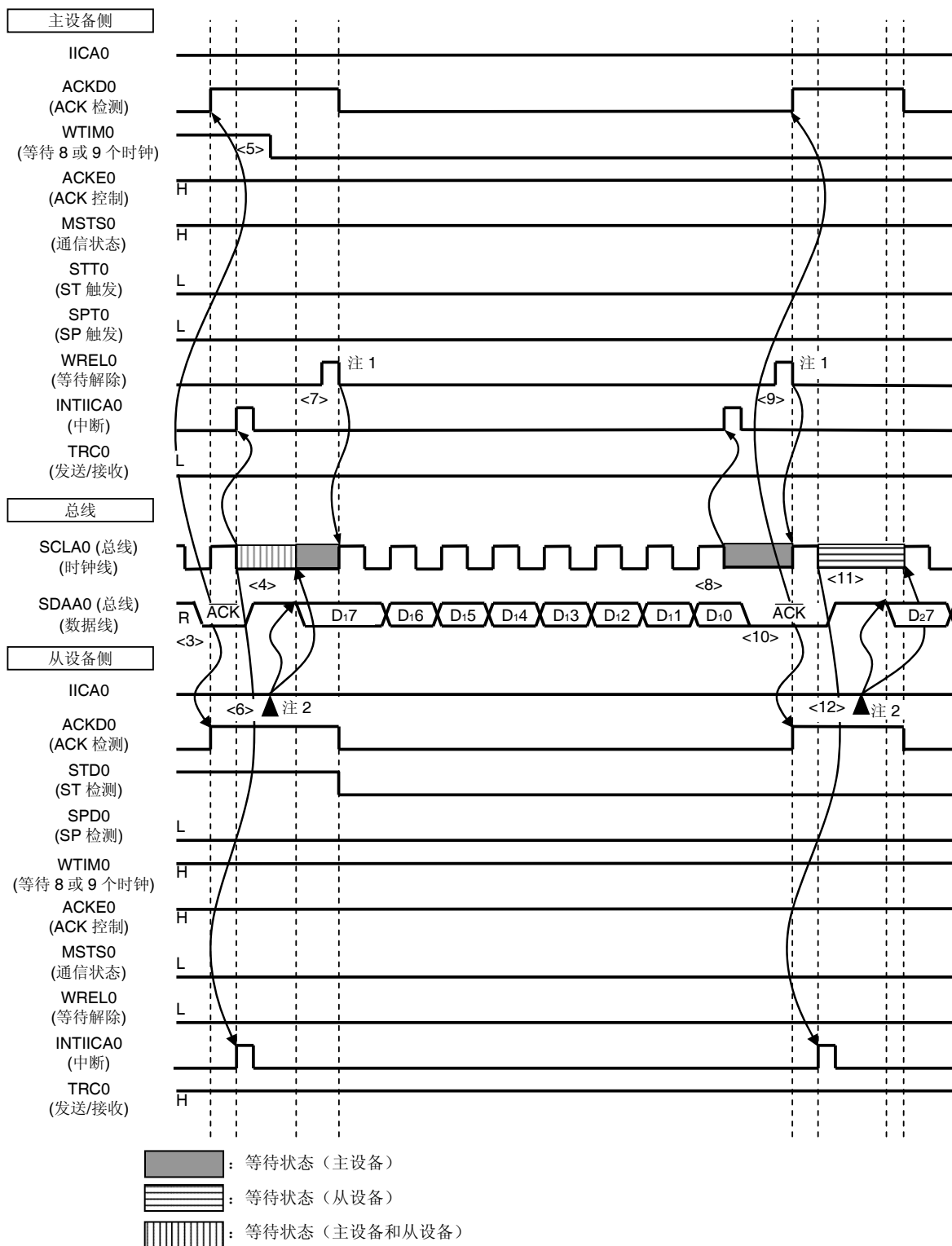
注 如果发送的地址和从设备的地址不匹配，则从设备不会向主设备返回 ACK(NACK: SDAA0 = 1)。从设备也不会产生 INTIICA0 中断 (地址匹配)，不会设置等待状态。但是，主设备将产生 INTIICA0 中断 (地址发送结束)，无论是收到 ACK 还是 NACK。

备注 图 17-34 中，<1>至<19>代表利用 I²C 总线进行数据通信的整个步骤。

图 17-34 (1)开始条件~ 地址~ 数据表示从<1> 到 <7> 的处理步骤，图 17-34 (2)地址 ~ 数据 ~ 数据表示从<3>到<12>的处理步骤，图 17-34 (3) 数据 ~ 数据 ~ 停止条件则表示从<8>到<19>的处理步骤。

图 17-34. 从设备→主设备通信示例
(主设备：选择 8 时钟等待， 从设备：选择 9 时钟等待) (2/3)

(2) 地址 ~ 数据 ~ 数据



注 1. 要取消主设备等待状态，须将“FFH”写入 IICA0，或者设置 WRELO 位。
 2. 将数据写入 IICA0，不设置 WRELO 位，以在从设备发送期间取消等待状态。

图 17-34 中，(2) 地址~数据~数据中的 <3>至<12> 的意义解释如下。

- <3> 如果收到的地址与从设备地址相匹配^註，则从设备将通过硬件向主设备发送 ACK。主设备在第 9 个时钟的上升沿检测 ACK (ACKD0 = 1)。
- <4> 主设备在第 9 个时钟的下降沿产生一个中断(INTIICA0: 地址发送结束)，和发送的地址相匹配的地址的从设备也产生一个中断(INTIICA0: 地址匹配)。地址匹配时，主设备和从设备还设置等待状态(SCLA0 = 0)^註。
- <5> 主设备设置等待状态的时序变成第 8 个时钟 (WTIM0 = 0)。
- <6> 从设备将待发送的数据写入 IICA 移位寄存器 0 (IICA0)，并解除由从设备设置的等待状态。
- <7> 如果主设备解除等待状态(WREL0 = 1)，则从设备将开始向主设备传送数据。
- <8> 主设备在第 8 个时钟下降沿设置一个等待状态 (SCLA0 = 0)，并产生一个中断 (INTIICA0: 传送结束)。然后主设备通过硬件向从设备发送 ACK。
- <9> 主设备读取接收到的数据，并解除等待状态(WREL0 = 1)。
- <10> 从设备在第 9 个时钟的上升沿检测 ACK (ACKD0 = 1)。
- <11> 从设备在第 9 个时钟的下降沿设置等待状态 (SCLA0 = 0)，而且从设备产生一个中断(INTIICA0: 传送结束)。
- <12> 从设备将待发送的数据写入 IICA0 寄存器，并解除由从设备设置的等待状态。然后，从设备开始向主设备传送数据。

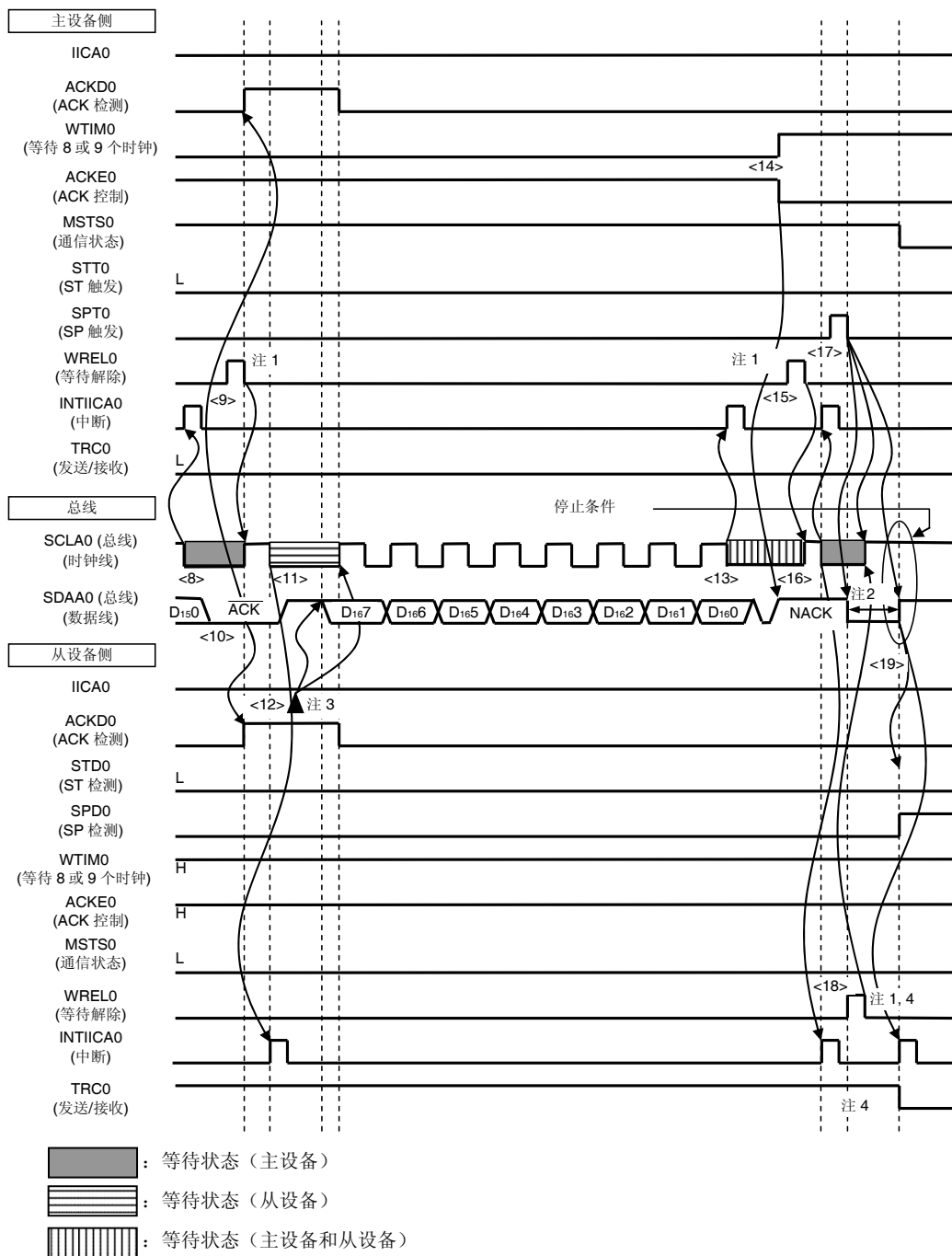
注 如果发送的地址和从设备的地址不匹配，则从设备不会向主设备返回 ACK(NACK: SDA0 = 1)。从设备也不会产生 INTIICA0 中断（地址匹配），不会设置等待状态。但是，主设备将产生 INTIICA0 中断（地址发送结束），无论是收到 ACK 还是 NACK。

备注 图 17-34 中，<1>至<19>代表利用 I²C 总线进行数据通信的整个步骤。

图 17-34 (1)开始条件~ 地址~ 数据表示从<1> 到 <7> 的处理步骤，图 17-34 (2)地址 ~ 数据 ~ 数据表示从<3>到<12>的处理步骤，图 17-34 (3) 数据 ~ 数据 ~ 停止条件则表示从<8>到<19>的处理步骤。

图 17-34. 从设备→主设备通信示例
(主设备：选择 8 时钟和 9 时钟等待， 从设备：选择 9 时钟等待) (3/3)

(3) 数据 ~ 数据 ~ 停止条件



- 注 1. 要取消等待状态，须将“FFH”写入 IICA0，或者设置 WREL0 位。
- 2. 确保在产生停止条件之后，从 SCLA0 引脚信号上升到产生停止条件的时间在指定标准模式时至少为 4.0 μs，在指定快速模式时至少为 0.6 μs。
- 3. 将数据写入 IICA0，不设置 WREL0 位，以在从设备发送期间取消等待状态。
- 4. 如果在从设备发送期间，通过设置 WREL0 位来取消等待状态，则 TRC0 位将被清除。

图 17-34 中，(3) 数据~数据~停止条件中的 <8>至<19> 的意义解释如下。

- <8> 主设备在第 8 个时钟下降沿设置一个等待状态 ($SCLA0 = 0$)，并产生一个中断 (INTIICA0: 传送结束)。然后主设备通过硬件向从设备发送 ACK。
- <9> 主设备读取接收到的数据，并解除等待状态($WRELO = 1$)。
- <10> 从设备在第 9 个时钟的上升沿检测 ACK ($ACKD0 = 1$)。
- <11> 从设备在第 9 个时钟的下降沿设置等待状态 ($SCLA0 = 0$)，而且从设备产生一个中断(INTIICA0: 传送结束)。
- <12> 从设备将待发送的数据写入 IICA 移位寄存器 0 (IICA0)，并解除由从设备设置的等待状态。然后，从设备开始向主设备传送数据。
- <13> 主设备在第 8 个时钟下降沿产生一个中断 (INTIICA0: 传送结束)，并设置一个等待状态 ($SCLA0 = 0$)。由于执行 ACK 控制($ACKE0 = 1$)，所以，总线数据线在此阶段处于低电平($SDAA0 = 0$)。
- <14> 作为响应，主设备设置 NACK ($ACKE0 = 0$)，并将其设置等待状态的时序更改为第 9 个时钟。
- <15> 如果主设备解除等待状态 ($WRELO = 1$)，从设备将在第 9 个时钟的上升沿检测 NACK ($ACK = 0$)。
- <16> 主设备和从设备在第 9 个时钟的下降沿设置等待状态 ($SCLA0 = 0$)，而且主设备和从设备都将产生一个中断 (INTIICA0: 传送结束)。
- <17> 当主设备产生一个停止条件($SPT0 = 1$)时，总线数据线被清零($SDAA0 = 0$)，主设备解除等待状态。然后，主设备处于等待状态，直到总线时钟线被设置为止($SCLA0 = 1$)。
- <18> 从设备确认 NACK，中止发送，并解除等待状态($WRELO = 1$)，以结束通信。主设备一旦解除等待状态，总线时钟线将被设置 ($SCLA0 = 1$)。
- <19> 当主设备认识到总线时钟线被设置($SCLA0 = 1$)，并在经过停止条件设置时间之后，主设备设置总线数据线 ($SDAA0 = 1$)，并产生一个停止条件。从设备检测产生的停止条件，并且主设备和从设备都将产生一个中断 (INTIICA0: 停止条件)。

备注 图 17-34 中，<1>至<19>代表利用 I²C 总线进行数据通信的整个步骤。

图 17-34 (1)开始条件~ 地址~ 数据表示从<1> 到 <7> 的处理步骤，图 17-34 (2)地址 ~ 数据 ~ 数据表示从<3>到<12>的处理步骤，图 17-34 (3) 数据 ~ 数据 ~ 停止条件则表示从<8>到<19>的处理步骤。

第十八章 乘除法器 and 乘加器

18.1 乘除法器 and 乘加器的功能

乘除法器 and 乘加器具有以下功能。

- 16 位 × 16 位 = 32 位 (无符号)
- 16 位 × 16 位 = 32 位 (有符号)
- 16 位 × 16 位 + 32 位 = 32 位 (无符号)
- 16 位 × 16 位 + 32 位 = 32 位 (有符号)
- 32 位 ÷ 32 位 = 32 位, 32 位余数 (无符号)

18.2 乘除法器 and 乘加器的配置

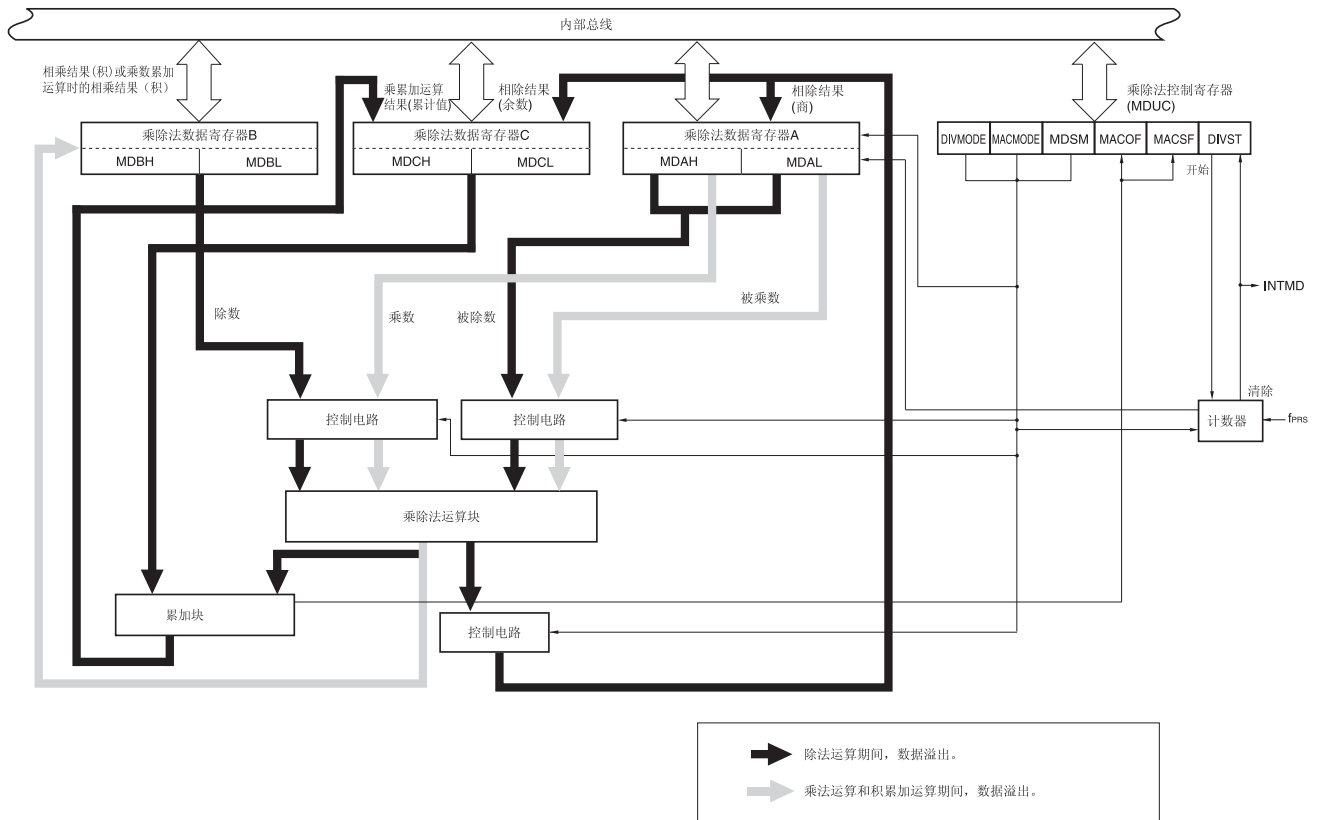
乘除法器 and 乘加器包含以下硬件。

表 18-1. 乘除法器 and 乘加器的配置

项目	配置
寄存器	乘法/除法数据寄存器 A (L) (MDAL) 乘法/除法数据寄存器 A (H) (MDAH) 乘法/除法数据寄存器 B (L) (MDBL) 乘法/除法数据寄存器 B (H) (MDBH) 乘法/除法数据寄存器 C (L) (MDCL) 乘法/除法数据寄存器 C (H) (MDCH)
控制寄存器	乘法/除法控制寄存器(MDUC)

乘除法器 and 乘加器的框图如图 18-1 所示。

图 18-1. 乘除法和乘加器的框图



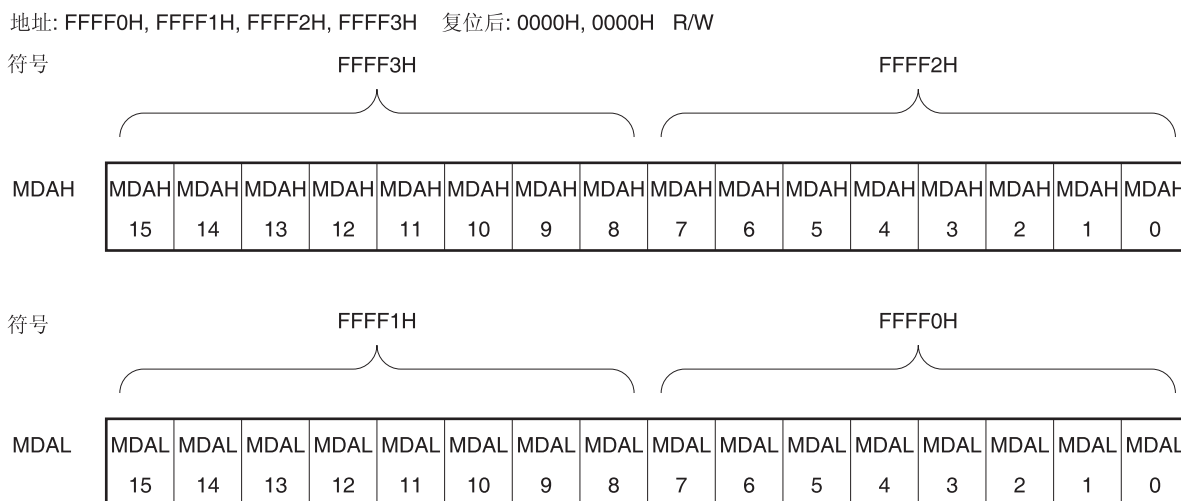
(1) 乘法/除法数据寄存器 A (MDAH, MDAL)

MDAH 和 MDAL 寄存器设置用于乘法或除法运算的值，并存储运算结果。在乘法模式或乘法累加器模式下，它们设置乘数和被乘数，在除法模式下，则设置被除数。另外，在除法模式下，运算结果（商）存储于 MDAH 和 MDAL 寄存器中。

使用 16 位操作指令设置 MDAH 和 MDAL 寄存器。

产生复位信号后，该寄存器被清除为 0000H。

图 18-2. 乘法/除法数据寄存器 A (MDAH, MDAL)的格式



- 注意事项**
1. 在除法运算过程中，不要重写 MDAH 和 MDAL 寄存器值(当乘法/除法控制寄存器(MDUC)的值为 81H 或 C1H)。虽然会执行运算，但运算结果为不确定的值。
 2. 在除法运算处理过程中读取的 MDAH 和 MDAL 寄存器值（即当 MDUC 寄存器值为 81H 或 C1H 时）将得不到保证。
 3. 在乘法模式（有符号）或乘法累加器模式（有符号）下，数据为二进制补码格式。

执行运算期间 MDAH 和 MDAL 寄存器的功能如下表所示。

表 18-2. 执行运算期间 MDAH 和 MDAL 寄存器的功能

操作模式	设置	运算结果
乘法模式 (无符号) 乘法累加器模式 (无符号)	MDAH: 乘数 (无符号) MDAL: 被乘数 (无符号)	—
乘法模式 (有符号) 乘法累加器模式 (有符号)	MDAH: 乘数 (有符号) MDAL: 被乘数 (有符号)	—
除法模式 (无符号)	MDAH: 被除数 (无符号) 高 16 位 MDAL: 被除数 (无符号) 低 16 位	MDAH: 商 (无符号) 高 16 位 MDAL: 商 (无符号) 低 16 位

(2) 乘法/除法数据寄存器 B (MDBL, MDBH)

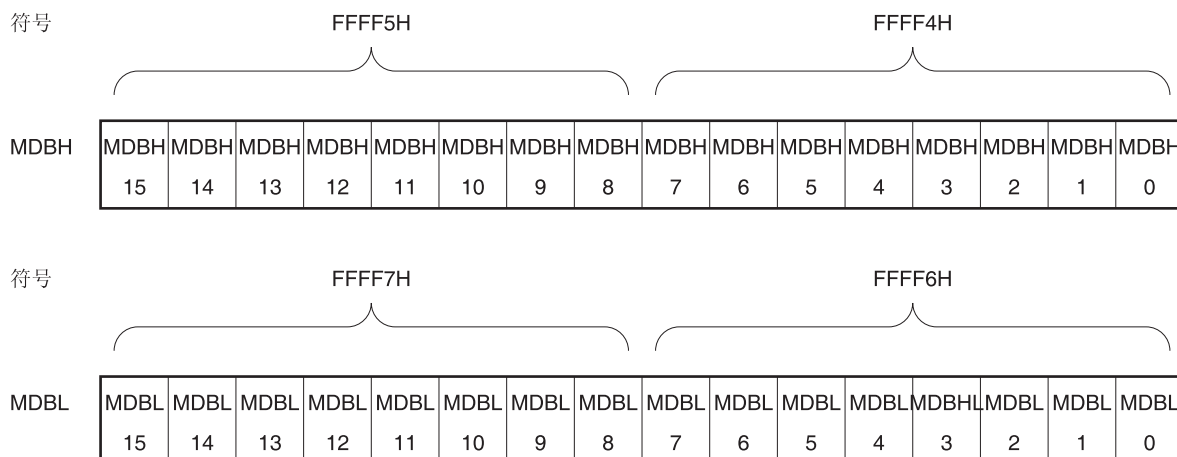
MDBH 和 MDBL 寄存器设置用于乘法或除法运算的值，并存储运算结果。在乘法模式和乘法累加器模式下，它们存储运算结果（积），在除法模式下，则设置除数。

使用 16 位操作指令设置 MDBH 和 MDBL 寄存器。

产生复位信号后，该寄存器被清除为 0000H。

图 18-3. 乘法/除法数据寄存器 B (MDBH, MDBL)的格式

地址: FFFF4H, FFFF5H, FFFF6H, FFFF7H 复位后: 0000H, 0000H R/W



- 注意事项**
1. 在除法运算处理(当乘法/除法控制寄存器(MDUC)的值为 81H 或 C1H)或者乘法累加运算处理过程中，不得重写 MDBH 和 MDBL 寄存器值。否则，运算结果将为不确定的值。
 2. 在除法模式下，不要将 MDBH 和 MDBL 寄存器设置为 0000H。如果设置的话，运算结果将为不确定的值。
 3. 在乘法模式（有符号）或乘法累加器模式（有符号）下，数据为二进制补码格式。

执行运算期间 MDBH 和 MDBL 寄存器的功能如下表所示。

表 18-3. 执行运算期间 MDBH 和 MDBL 寄存器的功能

操作模式	设置	运算结果
乘法模式 (无符号) 乘法累加器模式 (无符号)	-	MDBH: 乘法结果(积) (无符号) 高 16 位 MDBL: 乘法结果(积) (无符号) 低 16 位
乘法模式 (有符号) 乘法累加器模式 (有符号)	-	MDBH: 乘法结果(积) (有符号) 高 16 位 MDBL: 乘法结果(积) (有符号) 低 16 位
除法模式 (无符号)	MDBH: 除数 (无符号) 高 16 位 MDBL: 除数 (无符号) 低 16 位	-

(3) 乘法/除法数据寄存器 C (MDCL, MDCH)

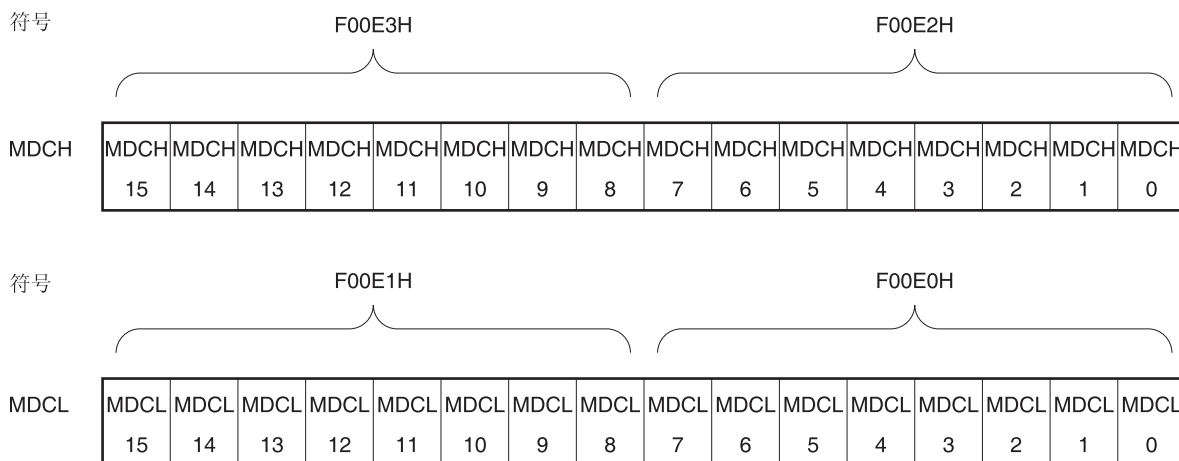
在乘法累加器模式下，MDCH 和 MDCL 寄存器用于存储累加结果，在除法模式下，则存储运算结果的余数。在乘法模式下，将不使用这些寄存器。

使用 16 位操作指令设置 MDCH 和 MDCL 寄存器。

产生复位信号后，该寄存器被清除为 0000H。

图 18-4. 乘法/除法数据寄存器 C (MDCH, MDCL)的格式

地址: F00E0H, F00E1H, F00E2H, F00E3H 复位后: 0000H, 0000H R/W



- 注意事项**
1. 在除法运算处理过程中(当乘法/除法控制寄存器(MDUC)的值为 81H 或 C1H)读取的 MDCH 和 MDCL 寄存器值将得不到保证。
 2. 在乘法累加器处理过程中，不得使用软件来重写 MDCH 和 MDCL 寄存器的值。如果重写，则运算结果将不确定。
 3. 在乘法累加器模式（有符号）下，数据为二进制补码格式。

表 18-4. 执行运算期间 MDCH 和 MDCL 寄存器的功能

操作模式	设置	运算结果
乘法模式 (无符号或有符号)	-	-
乘法累加器模式 (无符号)	MDCH: 累加初始值(无符号) 高 16 位 MDCL: 累加初始值(无符号) 低 16 位	MDCH: 累加值(无符号) 高 16 位 MDCL: 累加值(无符号) 低 16 位
乘法累加器模式 (有符号)	MDCH: 累加初始值(有符号) 高 16 位 MDCL: 累加初始值(有符号) 低 16 位	MDCH: 累加值(有符号) 高 16 位 MDCL: 累加值(有符号) 低 16 位
除法模式 (无符号)	-	MDCH: 余数 (无符号) 高 16 位 MDCL: 余数 (无符号) 低 16 位

寄存器配置在执行乘法和除法时有所不同，如下所示。

- 乘法运算期间的寄存器配置

<乘数 A> <乘数 B> <积>
 MDAL (位 15 至 0) × MDAH (位 15 至 0) = [MDBH (位 15 至 0), MDBL (位 15 至 0)]

- 乘法累加期间的寄存器配置

<乘数 A> <乘数 B> <累加值> <累加结果>
 MDAL (位 15 至 0) × MDAH (位 15 至 0) + MDC (位 31 至 0) = [MDCH (位 15 至 0), MDCL (位 15 至 0)]
 (乘法运算结果存储于 MDBH (位 15 至 0) 和 MDBL (位 15 至 0)。)

- 除法运算期间的寄存器配置

<被除数> <除数>
 [MDAH (位 15 至 0), MDAL (位 15 至 0)] ÷ [MDBH (位 15 至 0), MDBL (位 15 至 0)] =
 <商> <余数>
 [MDAH (位 15 至 0), MDAL (位 15 至 0)] ... [MDCH (位 15 至 0), MDCL (位 15 至 0)]

18.3 控制乘除法和乘加器的寄存器

利用乘法/除法控制寄存器(MDUC)来控制乘除法和乘加器。

(1) 乘法/除法控制寄存器(MDUC)

MDUC 寄存器是用于控制乘除法和乘加器操作的 8 位寄存器。

使用 1 位或 8 位存储器操作指令设置 MDUC 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 18-5. 乘法/除法控制寄存器(MDUC)的格式

地址: F00E8H 复位后: 00H R/W

符号	<7>	<6>	5	4	<3>	<2>	<1>	<0>
MDUC	DIVMODE	MACMODE	0	0	MDSM	MACOF	MACSF	DIVST

DIVMODE	MACMODE	MDSM	操作模式选择
0	0	0	乘法模式 (无符号) (默认)
0	0	1	乘法模式 (有符号)
0	1	0	乘法累加器模式 (无符号)
0	1	1	乘法累加器模式 (有符号)
1	0	0	除法模式 (无符号), 除法运算结束时, 产生 INTMD 中断
1	1	0	除法模式 (无符号), 除法运算结束时, 无 INTMD 中断
其它			禁止设置

MACOF	乘法累加结果(累加值)的溢出标志
0	无溢出
1	有溢出
<设置条件> •对于乘法累加器模式 (无符号) 当累加值超出 00000000h 至 FFFFFFFFh 的范围时, 该位被设置。 •对于乘法累加器模式 (有符号) 当一个正积与一个正累加值之和超过 7FFFFFFFh 且为负时, 或者当一个负积与一个负累加值之和超过 80000000h 且为正时, 该位被设置。	

MACSF	乘法累加结果(累加值)的符号标志
0	累加值为正数。
1	累加值为负数。
乘法累加器模式(无符号): 该位常为 0。 乘法累加器模式(有符号): 该位表示累加值的符号位。	

DIVST ^注	除法运算操作的开始/停止
0	除法运算处理结束
1	除法运算开始/除法运算处理中

(注和注意事项如下页所示。)

注 只有在除法模式下才可将 DIVST 位置位 1。在除法模式下，通过将 DIVST 位置位 1 开始除法运算。当运算结束时，DIVST 被自动清除 0。在乘法模式下，将乘数和被乘数设置到乘法/除法数据寄存器 A (MDAH, MDAL)，将自动开始运算。

- 注意事项**
1. 不可在运算处理过程中(当 DIVST 位为 1)重写 DIVMODE、MDSM 位。如果重写，运算结果将为不确定的值。
 2. 在除法运算过程中(当 DIVST 位为 1 时)，不能使用软件将 DIVST 位清除 0。

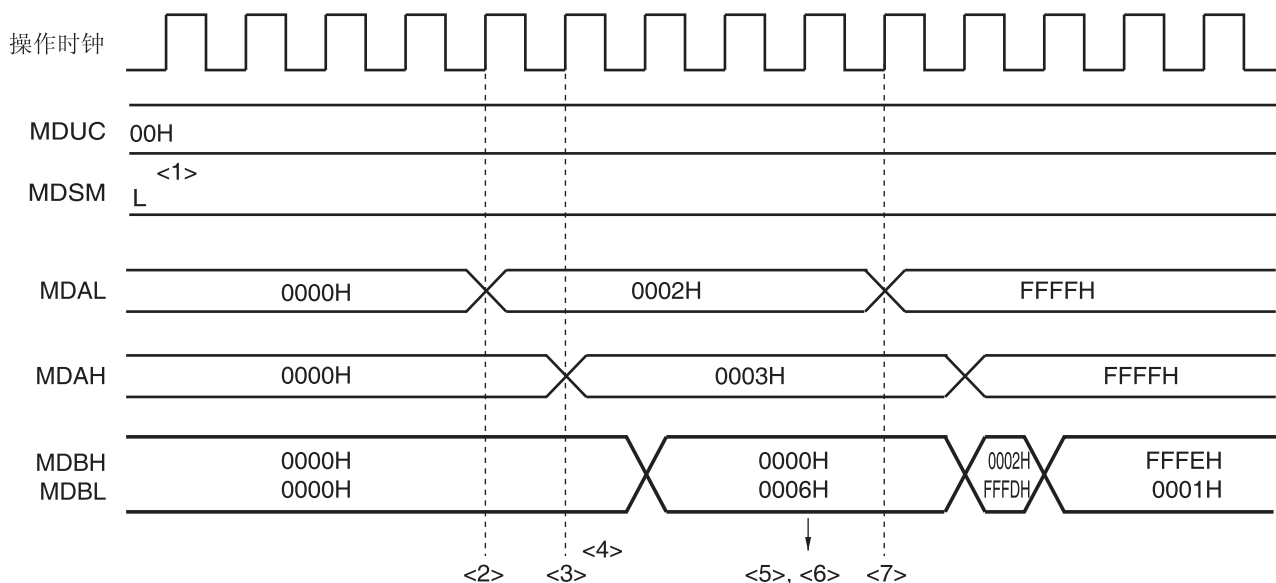
18.4 乘除法器 and 乘加器的操作

18.4.1 乘法(无符号)操作

- 初始设置
 - <1> 将乘法/除法控制寄存器(MDUC)设置为 00H。
 - <2> 将被乘数设置至乘法/除法数据寄存器 A (L) (MDAL)。
 - <3> 将乘数设置至乘法/除法数据寄存器 A (H) (MDAH)。
 - (在执行第<2>和第<3>步时没有优先顺序之分。当乘数和被乘数被分别设置至 MDAH 和 MDAL 寄存器时，自动开始乘法运算。)
- 运算处理
 - <4> 至少等待一个时钟。直到运算结束将占用一个时钟。
- 运算结束
 - <5> 从乘法/除法数据寄存器 B (L) (MDBL)读取积低 16 位。
 - <6> 从乘法/除法数据寄存器 B (H) (MDBH)读取积高 16 位。
 - (步骤<5>和步骤<6>的读取没有优先顺序之分。)
- 后续运算
 - <7> 下次执行乘法、除法或乘法累加运算时，从各个步骤的初始设置开始。

备注 步骤<1>至<7>对应于图 18-6 中的<1>至<7>。

图 18-6. 乘法(无符号)操作的时序图(2 × 3 = 6)



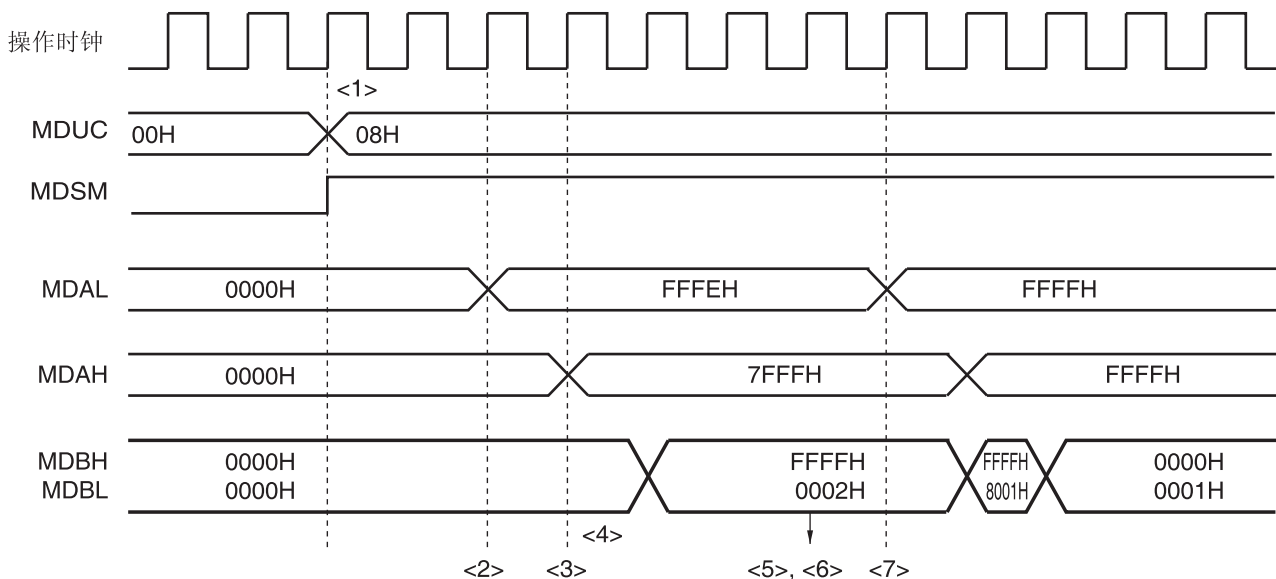
18.4.2 乘法(有符号)操作

- 初始设置
 - <1> 将乘法/除法控制寄存器(MDUC)设置为 08H。
 - <2> 将被乘数设置至乘法/除法数据寄存器 A (L) (MDAL)。
 - <3> 将乘数设置至乘法/除法数据寄存器 A (H) (MDAH)。
 - (在执行第<2>和第<3>步时没有优先顺序之分。当乘数和被乘数被分别设置至 MDAH 和 MDAL 寄存器时, 自动开始乘法运算。)
- 运算处理
 - <4> 至少等待一个时钟。直到运算结束将占用一个时钟。
- 运算结束
 - <5> 从乘法/除法数据寄存器 B (L) (MDBL)读取积低 16 位。
 - <6> 从乘法/除法数据寄存器 B (H) (MDBH)读取积高 16 位。
 - (步骤<5>和步骤<6>的读取没有优先顺序之分。)
- 后续运算
 - <7> 下次执行乘法、除法或乘法累加运算时, 从各个步骤的初始设置开始。

注意事项 在乘法模式 (有符号) 下, 数据为二进制补码格式。

备注 步骤<1>至<7>对应于图 18-7 中的<1>至<7>。

图 18-7. 乘法(有符号)操作的时序图(-2 × 32767 = -65534)

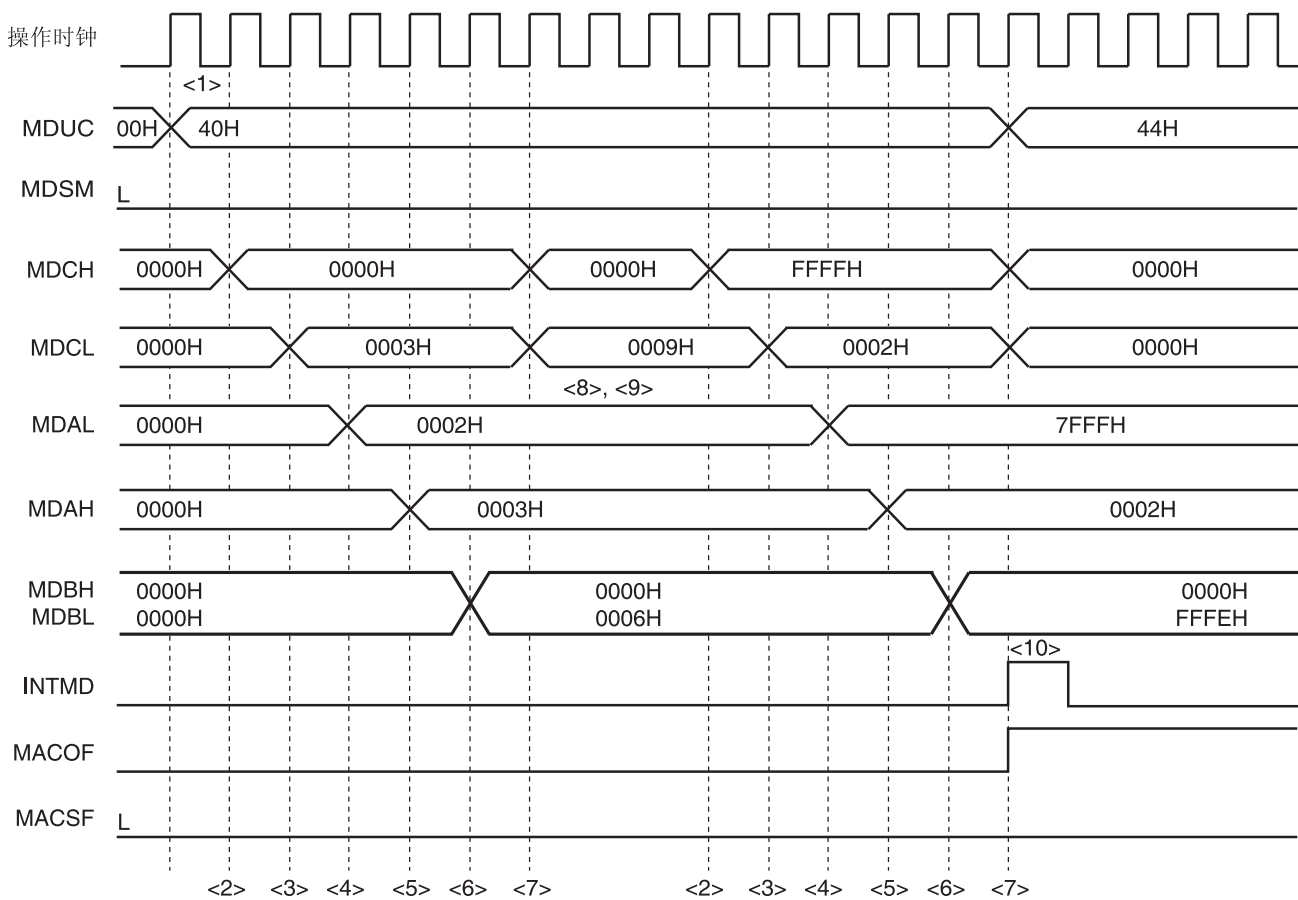


18.4.3 乘法累加(无符号)操作

- 初始设置
 - <1> 将乘法/除法控制寄存器(MDUC)设置为 40H。
 - <2> 将高 16 位初始累加值设置至乘法/除法数据寄存器 C (H) (MDCH)。
 - <3> 将低 16 位初始累加值设置至乘法/除法数据寄存器 C (L) (MDCL)。
 - <4> 将被乘数设置至乘法/除法数据寄存器 A (L) (MDAL)。
 - <5> 将乘数设置至乘法/除法数据寄存器 A (H) (MDAH)。
(在执行第<2>、第<3>和第<4>步时没有优先顺序之分。当乘数被分别设置至 MDAH 寄存器时, 将自动开始乘法运算。)
- 运算处理中
 - <6> 乘法运算在一个时钟周期内完成。
(乘法结果存储在乘法/除法数据寄存器 B (L) (MDBL)和乘法/除法数据寄存器 B (H) (MDBH)中。)
 - <7> 在第<6>步后, 乘法累加运算还需一个时钟周期才能结束。(在完成指定初始设置(<5>)之后, 至少要等两个时钟周期。)
- 运算结束
 - <8> 从 MDCL 寄存器读取累加值低 16 位。
 - <9> 从 MDCH 寄存器读取累加值高 16 位。
(步骤<8>和步骤<9>的读取没有优先顺序之分。)
 - (<10> 如果乘法累加运算的结果导致溢出, 则 MACOF 位被设置为 1, 并产生 INTMD 信号。)
- 后续运算
 - <11> 下次执行乘法、除法或乘法累加运算时, 从各个步骤的初始设置开始。

备注 步骤<1>至<10>对应于图 18-8 中的<1>至<10>。

图 18-8. 乘法累加(无符号)操作的时序图
 $(2 \times 3 + 3 = 9 \rightarrow 32767 \times 2 + 4294901762 = 0 \text{ (产生溢出)})$



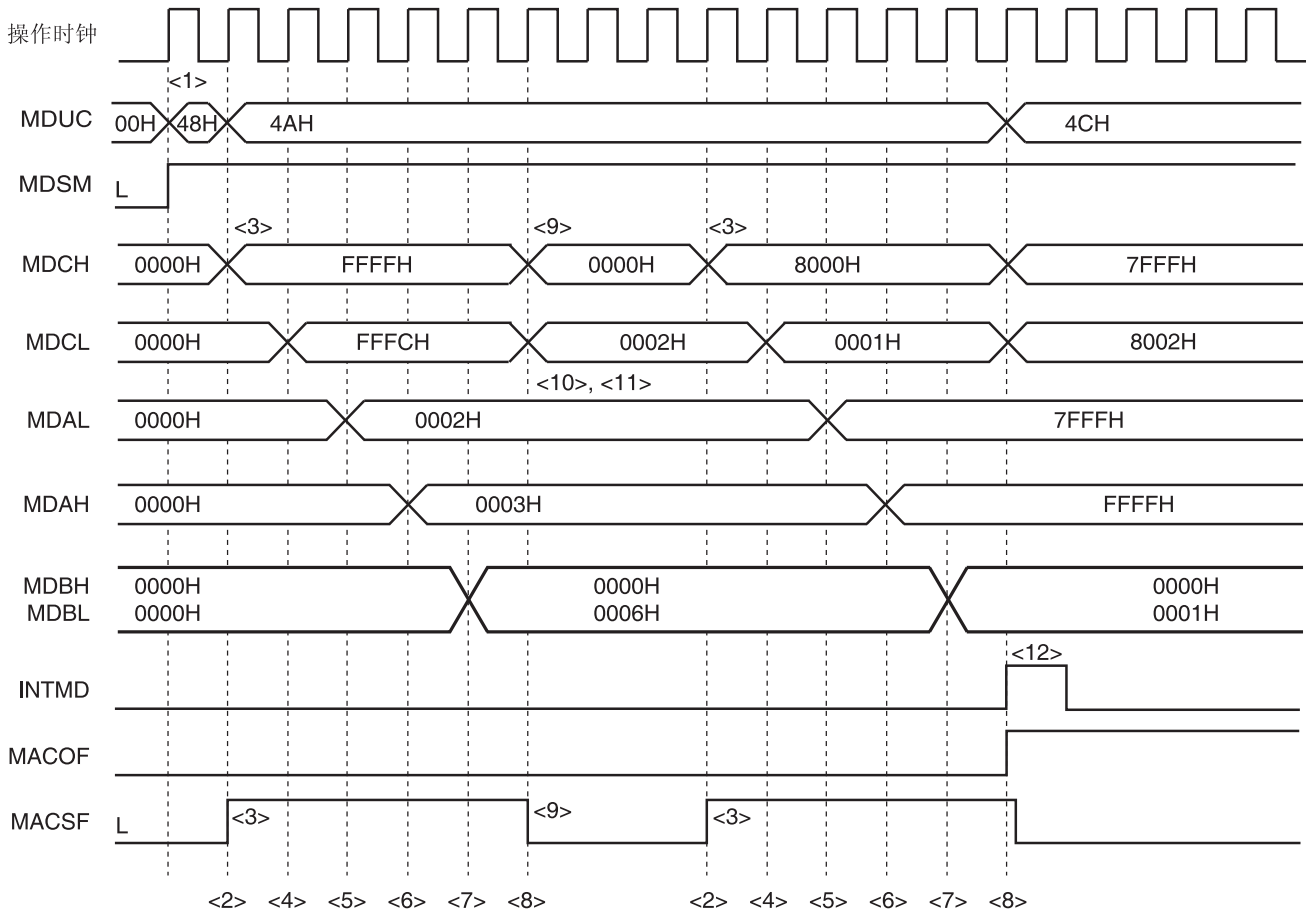
18.4.4 乘法累加（有符号）操作

- 初始设置
 - <1> 将乘法/除法控制寄存器(MDUC)设置为 48H。
 - <2> 将高 16 位初始累加值设置至乘法/除法数据寄存器 C (H) (MDCH)。
 - <3> 如果 MDCH 寄存器中的累加值为负, 则 MACSF 位将被设置为 1。
 - <4> 将低 16 位初始累加值设置至乘法/除法数据寄存器 C (L) (MDCL)。
 - <5> 将被乘数设置至乘法/除法数据寄存器 A (L) (MDAL)。
 - <6> 将乘数设置至乘法/除法数据寄存器 A (H) (MDAH)。
(步骤<2>、步骤<4>和步骤<5>的设置没有优先顺序之分。当乘数被设置至<6>的 MDAH 寄存器时, 自动开始乘法运算。)
- 运算处理
 - <7> 乘法运算在一个时钟周期内完成。
(乘法结果存储在乘法/除法数据寄存器 B (L) (MDBL)和乘法/除法数据寄存器 B (H) (MDBH)中。)
 - <8> 在第<7>步后, 乘法累加运算还需一个时钟周期才结束。(完成指定初始设置(<6>)之后, 至少要等两个时钟周期。)
- 运算结束
 - <9> 如果 MDCL 和 MDCH 寄存器中存储的累加值为正, MACSF 位将被清除为 0。
 - <10> 从 MDCL 寄存器读取累加值 (低 16 位)。
 - <11> 从 MDCH 寄存器读取累加值 (高 16 位)。
(步骤<10>和步骤<11>的读取没有优先顺序之分。)
 - <12> 如果乘法累加运算的结果导致溢出, 则 MACOF 位被设置为 1, 并产生 INTMD 信号。
- 后续运算
 - <13> 下次执行乘法、除法或乘法累加运算时, 从各个步骤的初始设置开始。

注意事项 在乘法累加（有符号）运算时, 数据为二进制补码格式。

备注 步骤<1>至<12>对应于图 18-9 中的<1>至<12>。

图 18-9. 乘法累加(有符号)操作的时序图
 $(2 \times 3 + (-4) = 2 \rightarrow 32767 \times (-1) + (-2147483647) = -2147450882$ (产生溢出。))

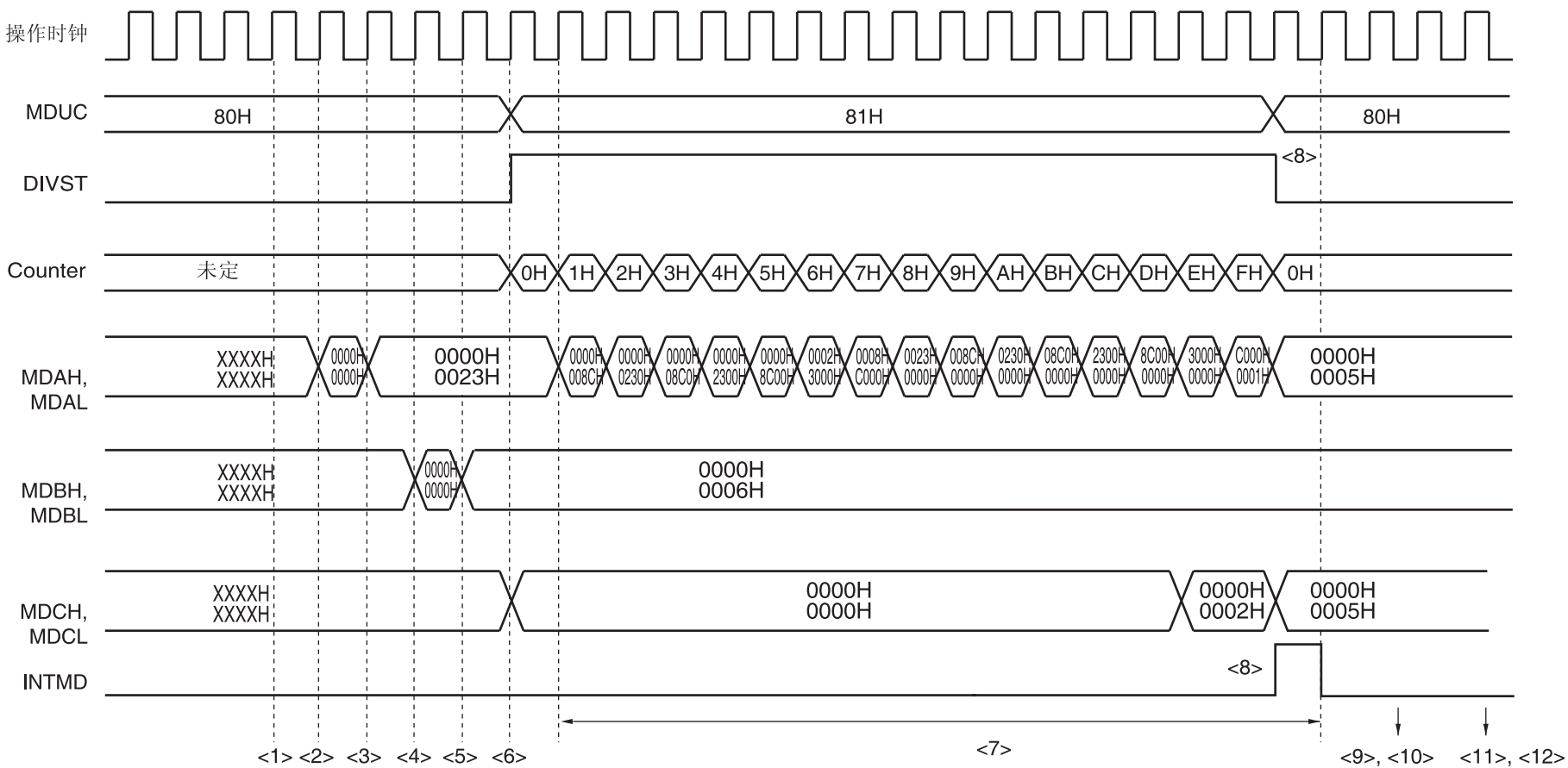


18.4.5 除法操作

- 初始设置
 - <1> 将乘法/除法控制寄存器(MDUC)设置为 80H。
 - <2> 将被除数高 16 位设置至乘法/除法数据寄存器 A (H) (MDAH)。
 - <3> 将被除数低 16 位设置至乘法/除法数据寄存器 A (L) (MDAL)。
 - <4> 将除数高 16 位设置至乘法/除法数据寄存器 B (H) (MDBH)。
 - <5> 将除数低 16 位设置至乘法/除法数据寄存器 B (L) (MDBL)。
 - <6> 将 MDUC 寄存器的位 0(DIVST)设置为 1。
(在执行第<2>至第<5>步时没有优先顺序之分。)
- 运算处理
 - <7> 当下列处理中的任意一个完成时, 结束运算。
 - 等待至少 16 个时钟(产生 16 个时钟后结束运算。)
 - 检查 DIVST 位是否已经被清除。
(在运算处理过程中, MDBL、MDBH、MDCL 和 MDCH 寄存器的读取值将得不到保证。)
- 运算结束
 - <8> DIVST 位被清除并结束运算。此时, 若执行 MACMODE = 0, 则产生一个中断请求信号 (INTMD)。
 - <9> 从 MDAL 寄存器读取商低 16 位。
 - <10> 从 MDAH 寄存器读取商高 16 位。
 - <11> 从乘法/除法数据寄存器 C (L) (MDCL)读取余数低 16 位。
 - <12> 从乘法/除法数据寄存器 C (H) (MDCH)读取余数高 16 位。
(在执行第<9>至第<12>步时没有优先顺序之分。)
- 后续运算
 - <13> 下次执行乘法、除法或乘法累加运算时, 从各个步骤的初始设置开始。

备注 步骤<1>至<12>对应于图 18-10 中的<1>至<12>。

图 18-10. 除法操作的时序图(示例: $35 \div 6 = 5$, 余数 5)



第十九章 DMA 控制器

RL78/I1A 内置了一个 DMA（直接存储器存取）控制器。

可以在支持 DMA 的外围硬件的 SFR 和内部 RAM 之间无需通过 CPU 自动传送数据。

因此，可以并行执行 CPU 的正常内部操作和数据传送以及 SFR 和内部 RAM 之间的传送，从而可以进行大量数据的处理。另外，也可实现对通信、定时器和 A/D 的实时控制。

19.1 DMA控制器的功能

- DMA 通道数量：2 个通道
- 传送单位：8 位或 16 位
- 最大传送单位：1024 次
- 传送类型：2 周期传送(利用 2 个时钟执行一次传送，并且传送期间 CPU 停止动作。)
- 传送模式：单次传送模式
- 传送请求：选自下列外围硬件中断
 - A/D 转换器
 - 串行接口
(CSI00, UART0, UART1)
 - 定时器(通道 0, 1, 2, 3)
- 传送目标：在 SFR 与内部 RAM 之间

使用 DMA 功能的示例如下所示。

- 串行接口的连续传送
- 模拟数据的批量传送
- 按照固定间隔读入 A/D 转换结果
- 按照固定间隔读入端口值

19.2 DMA控制器的配置

DMA 控制器包含以下硬件。

表 19-1. DMA 控制器的配置

项目	配置
地址寄存器	<ul style="list-style-type: none"> • DMA SFR 地址寄存器 0, 1 (DSA0, DSA1) • DMA RAM 地址寄存器 0, 1 (DRA0, DRA1)
计数寄存器	<ul style="list-style-type: none"> • DMA 字节计数寄存器 0, 1 (DBC0, DBC1)
控制寄存器	<ul style="list-style-type: none"> • DMA 模式控制寄存器 0, 1 (DMC0, DMC1) • DMA 操作控制寄存器 0, 1 (DRC0, DRC1)

(1) DMA SFR 地址寄存器 n (DSAn)

这是一个用于设置 SFR 地址的 8 位寄存器，该地址是 DMA 通道 n 的传送源或传送目的地。

设置 SFR 地址 FFF00H 至 FFFFFH 的低 8 位。

该寄存器不会自动递增，而是一个固定值。

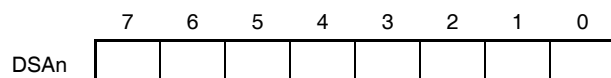
在 16 位传送模式下，忽略最低位，作为偶数地址处理。

可以以 8 位为单位读取或写入 DSAn 寄存器。然而，不能在 DMA 传送过程中写入。

产生复位信号后，该寄存器被清除为 00H。

图 19-1. DMA SFR 地址寄存器 n (DSAn)的格式

地址: FFFB0H (DSA0), FFFB1H (DSA1) 复位后: 00H R/W



备注 n: DMA 通道编号 (n = 0, 1)

(2) DMA RAM 地址寄存器 n (DRAn)

这是一个用于设置 RAM 地址的 16 位寄存器，该地址是 DMA 通道 n 的传送源或传送目的地。

可将通用寄存器以外的内部 RAM 区域的地址（见表 19-2）设置于该寄存器。

设置 RAM 地址的低 16 位。

该寄存器将在 DMA 传送开始后自动递增。在 8 位传送模式下，递增单位为+1，16 位传送模式下，递增单位为+2。

DMA 传送从设置至该 DRAn 寄存器的地址开始。当最后地址的数据传送完成后，DMA 传送停止工作，此时，在 8 位传送模式下，停止值为最后地址+1，在 16 位传送模式下，停止值为最后地址+2。

在 16 位传送模式下，忽略最低位，作为偶数地址处理。

可以以 8 位或 16 位为单位读取或写入 DRAn 寄存器。然而，不能在 DMA 传送过程中写入。

产生复位信号后，该寄存器被清除为 0000H。

图 19-2. DMA RAM 地址寄存器 n (DRAn)的格式

地址: FFFB2H, FFFB3H (DRA0), FFFB4H, FFFB5H (DRA1), 复位后: 0000H R/W

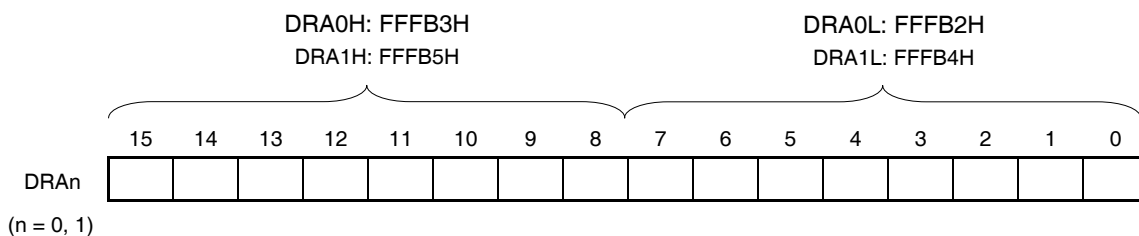


表 19-2. 通用寄存器以外的内部 RAM 区域

产品编号	通用寄存器以外的内部RAM区域
R5F1076C, R5F107AC, R5F107BC	FF700H至FFEDFH
R5F107AE, R5F107DE	FEF00H至FFEDFH

备注 n: DMA 通道编号 (n = 0, 1)

(3) DMA 字节计数寄存器 n (DBCn)

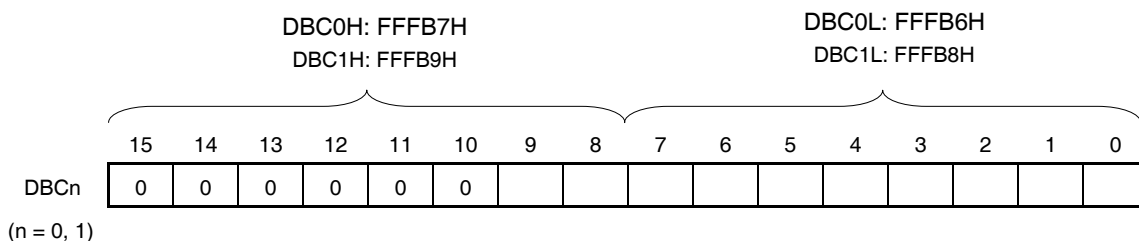
这是一个 10 位寄存器，用于设置 DMA 通道 n 执行传送的次数。必须在执行 DMA 传送之前将传送次数设置至该 DBCn 寄存器（最多 1024 次）。

每次执行一次 DMA 传送之后，该寄存器将自动递减。通过在 DMA 传送过程中读取该 DBCn 寄存器，可以获知剩余的传送次数。

可以以 8 位或 16 位为单位读取或写入 DBCn 寄存器。然而，不能在 DMA 传送过程中写入。产生复位信号后，该寄存器被清除为 0000H。

图 19-3. DMA 字节计数寄存器 n (DBCn)的格式

地址: FFFB6H, FFFB7H (DBC0), FFFB8H, FFFB9H (DBC1) 复位后: 0000H R/W



DBCn[9:0]	传送次数 (当 DBCn 处于写入状态)	剩余传送次数 (当 DBCn 处于读取状态)
000H	1024	传送结束或者等待 1024 次的 DMA 传送
001H	1	等待剩余 1 次的 DMA 传送
002H	2	等待剩余 2 次的 DMA 传送
003H	3	等待剩余 3 次的 DMA 传送
•	•	•
•	•	•
•	•	•
3FEH	1022	等待剩余 1022 次的 DMA 传送
3FFH	1023	等待剩余 1023 次的 DMA 传送

- 注意事项**
1. 必须将位 15 至 10 清除为 0。
 2. 如果指定了通用寄存器，或者因连续传送而超过了内部 RAM 空间，则会写入或读取通用寄存器或 SFR 空间，导致这些空间中的数据丢失。必须使传送次数处于内部 RAM 空间范围之内。

备注 n: DMA 通道编号 (n = 0, 1)

19.3 控制DMA控制器的寄存器

DMA 控制器由以下两个寄存器控制。

- DMA 模式控制寄存器 n (DMCn)
- DMA 操作控制寄存器 n (DRCn)

备注 n: DMA 通道编号 (n = 0, 1)

(1) DMA 模式控制寄存器 n (DMCn)

DMCn 寄存器用于设置 DMA 通道 n 的传送模式。可以选择传送方向、数据宽度、挂起设置和触发源。位 7 (STGn) 为软件触发，用于启动 DMA。

禁止在 DMA 操作过程中（当 DSTn = 1 时）改写 DMCn 寄存器的位 6、位 5 以及位 3 至位 0。

使用 1 位或 8 位存储器操作指令设置 DMCn 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 19-4. DMA 模式控制寄存器 n (DMCn)的格式 (1/2)

地址：FFFBAH (DMC0), FFFBBH (DMC1) 复位后：00H R/W

符号	<7>	<6>	<5>	<4>	3	2	1	0
DMCn	STGn	DRSn	DSn	DWAITn	IFCn3	IFCn2	IFCn1	IFCn0
STGn ^{#1}	DMA传送开始软件触发							
0	无触发触发							
1	当允许DMA操作(DENn = 1)时，开始DMA传送。							
允许DMA操作（DENn = 1）时，通过向STGn位写入1来执行一次DMA传送。 该位的读取值始终为0。								
DRSn	选择DMA传送方向							
0	SFR至内部RAM							
1	内部RAM至SFR							
DSn	指定DMA传送的传送数据宽度							
0	8位							
1	16位							
DWAITn ^{#2}	挂起DMA传送							
0	在收到DMA开始请求时执行DMA传送（不挂起）。							
1	即使收到DMA开始请求也挂起。							
可以通过将DWAITn位的值清除为0来开始被挂起的DMA传送。 从DWAITn位的值被设置为1开始，直到DMA传送被实际挂起，需要两个时钟周期。								

- 注 1. 无论 IFCn0 至 IFCn3 位为何值，均可使用软件触发(STGn)。
2. 在使用双 DMA 通道时，如果挂起 DMA 传送，则必须挂起双通道的 DMA 传送（将 DWAIT0 和 DWAIT1 位设置为 1）。

备注 n: DMA 通道编号 (n = 0, 1)

图 19-4. DMA 模式控制寄存器 n (DMCn)的格式 (2/2)

地址: FFFBAH (DMC0), FFFBBH (DMC1) 复位后: 00H R/W

符号	<7>	<6>	<5>	<4>	3	2	1	0
DMCn	STGn	DRSn	DSn	DWAItn	IFCn3	IFCn2	IFCn1	IFCn0

(当 n = 0 或 1)

IFCn 3	IFCn 2	IFCn 1	IFCn 0	选择DMA触发源 ^注	
				触发信号	触发内容
0	0	0	0	-	禁止中断触发DMA传送。 (仅允许软件触发。)
0	0	0	1	INTAD	A/D转换结束中断
0	0	1	0	INTTM00	定时器通道0的计数结束或读入中断
0	0	1	1	INTTM01	定时器通道1的计数结束或读入中断
0	1	0	0	INTTM02	定时器通道2的计数结束或读入中断
0	1	0	1	INTTM03	定时器通道3的计数结束或读入中断
0	1	1	0	INTST0/INTCSI00	UART0发送的传送结束或缓冲器空中断 /CSI00传送结束或缓冲器空中断
0	1	1	1	INTSR0	UART0接收的传送结束中断
1	0	0	0	INTST1	UART1发送的传送结束或缓冲器空中断
1	0	0	1	INTSR1	UART1接收的传送结束中断
其他				禁止设置	

注 无论 IFCn0 至 IFCn3 位为何值，均可使用软件触发(STGn)。

备注 n: DMA 通道编号 (n = 0, 1)

(2) DMA 操作控制寄存器 n (DRCn)

DRCn 寄存器用于允许或禁止 DMA 通道 n 的传送。

禁止在 DMA 操作过程中（当 DSTn = 1 时）改写该寄存器的位 7 (DENn)。

使用 1 位或 8 位存储器操作指令设置 DRCn 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 19-5. DMA 操作控制寄存器 n (DRCn)的格式

地址: FFFBCH (DRC0), FFFBDH (DRC1) 复位后: 00H R/W

符号	<7>	6	5	4	3	2	1	<0>
DRCn	DENn	0	0	0	0	0	0	DSTn

DENn	DMA操作允许标志
0	禁止DMA通道n的操作 (DMA工作时钟停止)。
1	允许DMA通道n的操作。
允许DMA 操作(DENn = 1)之后，当DSTn = 1时，DMA将进入等待DMA触发的状态。	

DSTn	DMA传送模式标志
0	DMA通道n的DMA传送结束。
1	DMA通道n的DMA传送未结束(仍在执行)。
允许DMA操作 (DENn = 1)之后，当DSTn = 1时，DMA将进入等待DMA触发的状态。 当输入一个软件触发(STGn)或者输入由IFCn3至IFCn0位设置的触发源时，开始DMA传送。 完成DMA传送后，该位被自动清除为0。 向该位写入0将强制终止执行中的DMA传送。	

注意事项 完成 DMA 传送后，DSTn 标志将被自动清除为 0。

仅当 DSTn = 0 时允许写入 DENn 标志。不等待产生 DMA 中断(INTDMA)就终止 DMA 传送时，先将 DSTn 位设置为 0，然后将 DENn 位设置为 0 (有关详情，请参阅 19.5.5 利用软件的强制结束)。

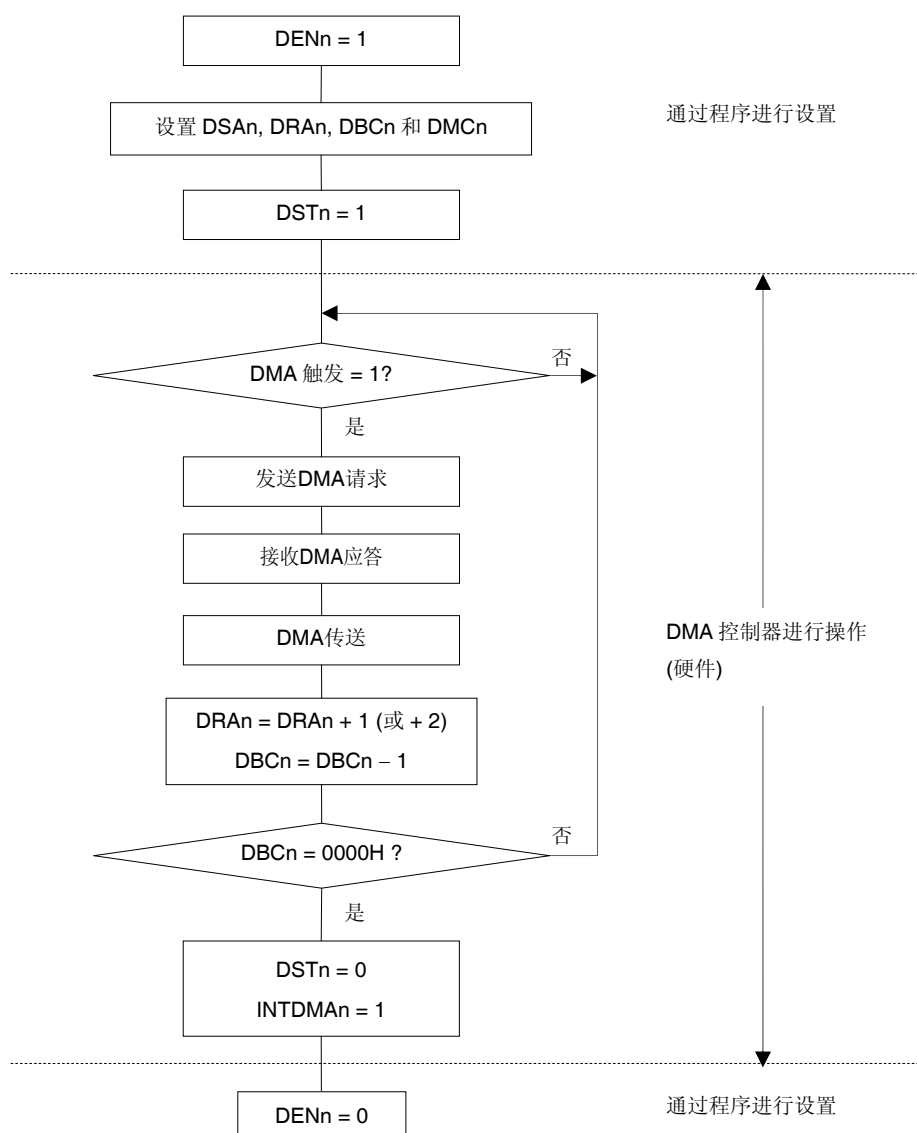
备注 n: DMA 通道编号 (n = 0, 1)

19.4 DMA控制器的操作

19.4.1 操作步骤

- <1> 当 DENn = 1 时，允许操作 DMA 控制器。在写入其他寄存器之前，必须将 DENn 位设置为 1。用 8 位操作指令进行写入操作时，写入 80H。
- <2> 将 DMA 传送的 SFR 地址、RAM 地址、传送次数，以及传送模式设置至 DMA SFR 地址寄存器 n (DSAn)、DMA RAM 地址寄存器 n (DRAn)、DMA 字节计数寄存器 n (DBCn)和 DMA 模式控制寄存器 n (DMCn)。
- <3> 当 DSTn = 1 时，DMA 控制器等待 DMA 触发。用 8 位操作指令进行写入操作时，写入 81H。
- <4> 当输入软件触发(STGn)或者输入由 IFCn3 至 IFCn0 位指定的触发源时，开始 DMA 传送。
- <5> 当由 DBCn 寄存器设置的传送次数变成 0 时，传送完成，产生中断(INTDMA n)时，传送自动终止。
- <6> 不使用 DMA 控制器时，将 DENn 位清除为 0，以停止操作 DMA 控制器。

图 19-6. 操作步骤



备注 n: DMA 通道编号 (n = 0, 1)

19.4.2 传送模式

通过 DMA 模式控制寄存器 n (DMCn) 的位 6 和位 5 (DRSn 和 DS_n)，可以选择以下四种 DMA 传送模式。

DRSn	DS _n	DMA传送模式
0	0	从1字节数据（固定地址）的SFR传送至RAM（地址递增+1）
0	1	从2字节数据（固定地址）的SFR传送至RAM（地址递增+2）
1	0	从1字节数据（地址递增+1）的RAM传送至SFR（固定地址）
1	1	从2字节数据（地址递增+2）的RAM传送至SFR（固定地址）

通过使用这些传送模式，可以利用串行接口最多连续传送 1024 字节数据，也可以连续传送 A/D 转换产生的数据，还可以利用定时器以固定间隔扫描端口数据。

19.4.3 结束DMA传送

当 DBCn = 00H 完成 DMA 传送时，DSTn 位将被自动清除为 0。结果将产生一个中断请求 (INTDMA_n) 并终止传送。

当 DSTn 位被清除为 0 以强制终止 DMA 传送时，DMA 字节计数器寄存器 n (DBCn) 和 DMA RAM 地址寄存器 n (DRAn) 将保持传送终止时的值。

如果强制终止传送，则不产生中断请求 (INTDMA_n)。

备注 n: DMA 通道编号 (n = 0, 1)

19.5 DMA控制器的设置示例

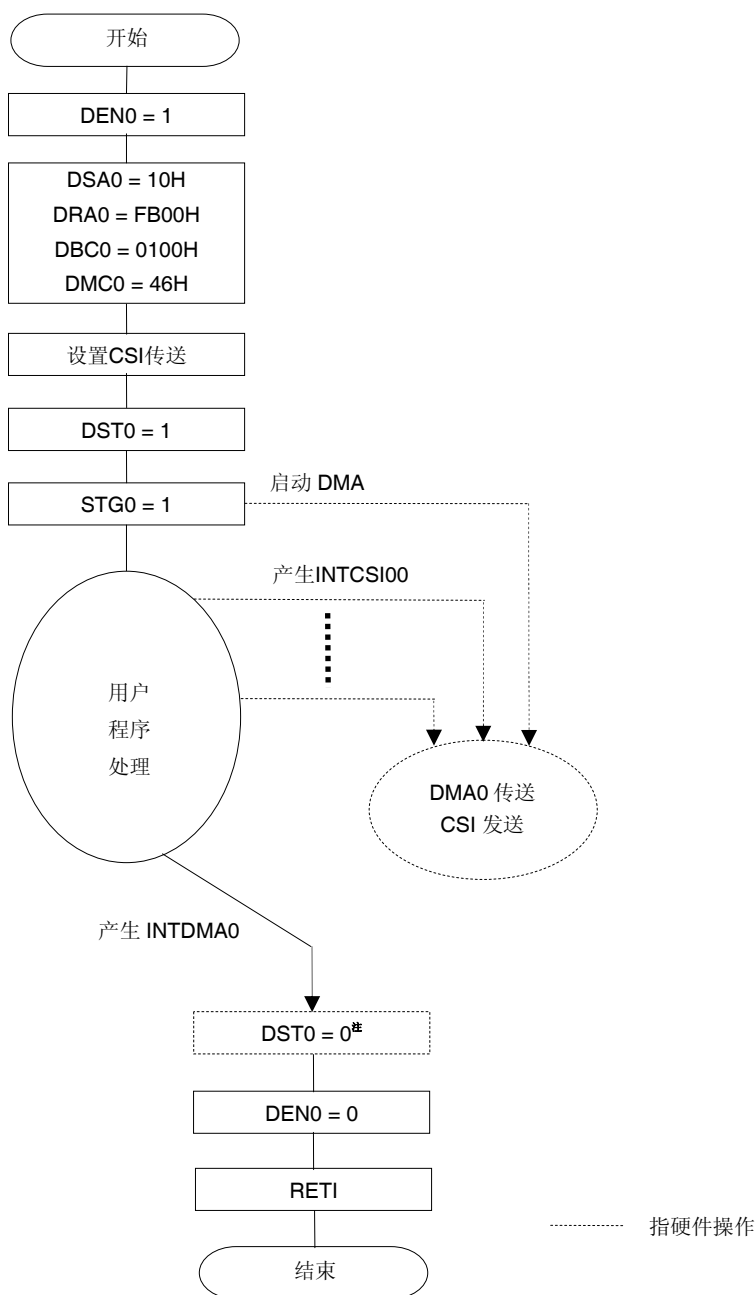
19.5.1 CSI连续发送

CSI 连续发送的设置示例如下页流程图所示。

- CSI00 的连续发送(256 字节)
- DMA 传送时使用 DMA 通道 0
- DMA 触发源: INTCSI00 (第一个触发源仅限软件触发(STG0))
- CSI00 的中断由 IFC03 至 IFC00 = 0110B 指定
- 将 RAM 的 FFB00H 至 FFBFFH (256 字节)传送至 CSI 的数据寄存器(SIO00)的 FFF10H

备注 IFC03 至 IFC00: DMA 模式控制寄存器 0 (DMC0)的位 3 至 0

图 19-7. CSI 连续发送的设置示例



注 当完成 DMA 传送时，DST0 标志将被自动清除为 0。

仅当 DST0 = 0 时允许写入 DEN0 标志。不等待产生 DMA0 中断(INTDMA0)就终止 DMA 传送时，先将 DST0 位设置为 0，然后将 DEN0 位设置为 0 (有关详情，请参阅 19.5.5 利用软件的强制结束)。

连续发送时，CSI 中断不能启动第一次发送。在这种情况下，通过软件触发启动。

第二次及以后的 CSI 发送被自动执行。

当最后发送数据被写入数据寄存器时，将产生一个 DMA 中断(INTDMA0)。

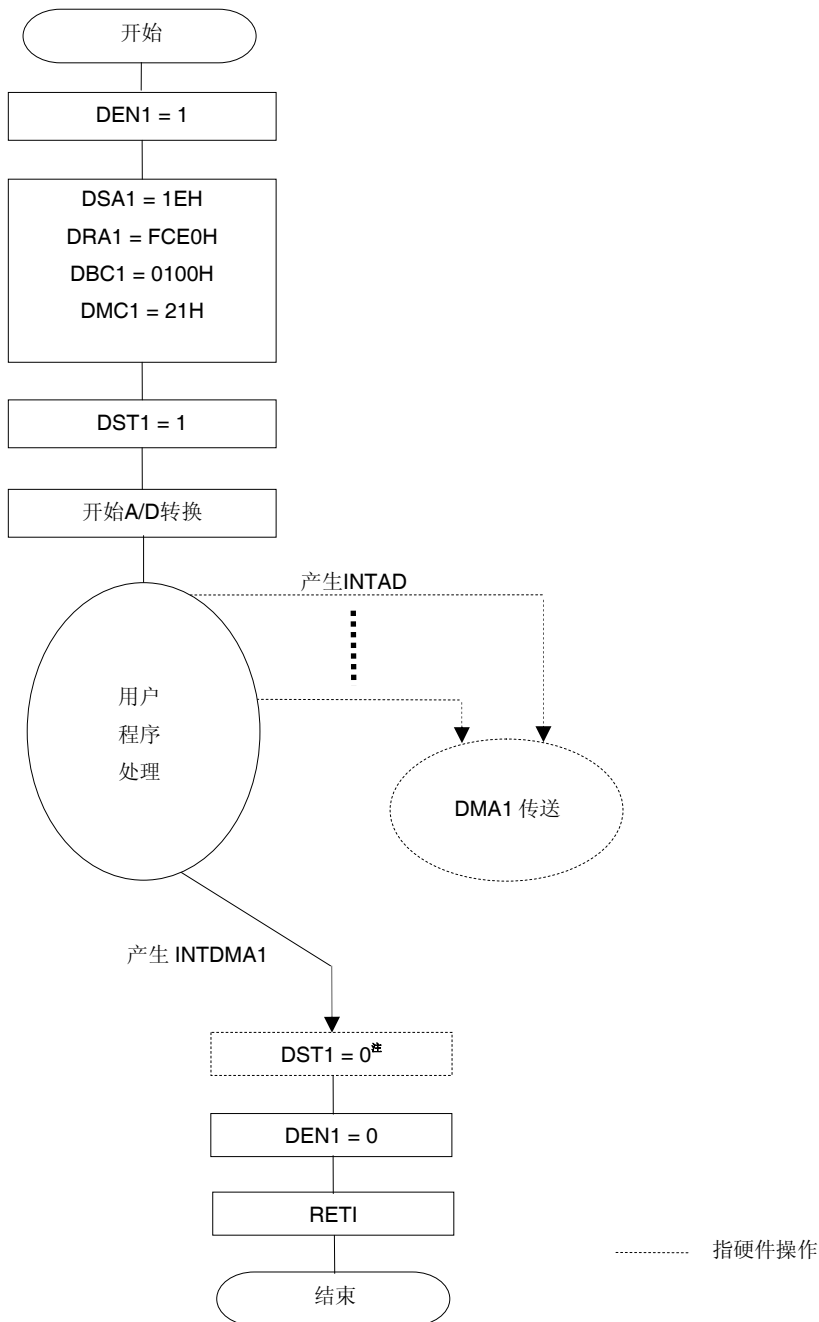
19.5.2 A/D转换结果的连续读入

设置 A/D 转换结果连续读入的示例如下页流程图所示。

- A/D 转换结果的连续读入
- DMA 传送时使用 DMA 通道 1
- DMA 触发源：INTAD
- A/D 的中断由 IFC13 至 IFC10 = 0001B 指定
- 将 10 位 A/D 转换结果寄存器(ADCR)的 FFF1EH 和 FFF1FH (2 字节) 传送至 RAM 的 FFCE0H 至 FFEDFH (512 字节)

备注 IFC13 至 IFC10: DMA 模式控制寄存器 1 (DMC1)的位 3 至 0

图 19-8. A/D 转换结果的连续读入的设置示例



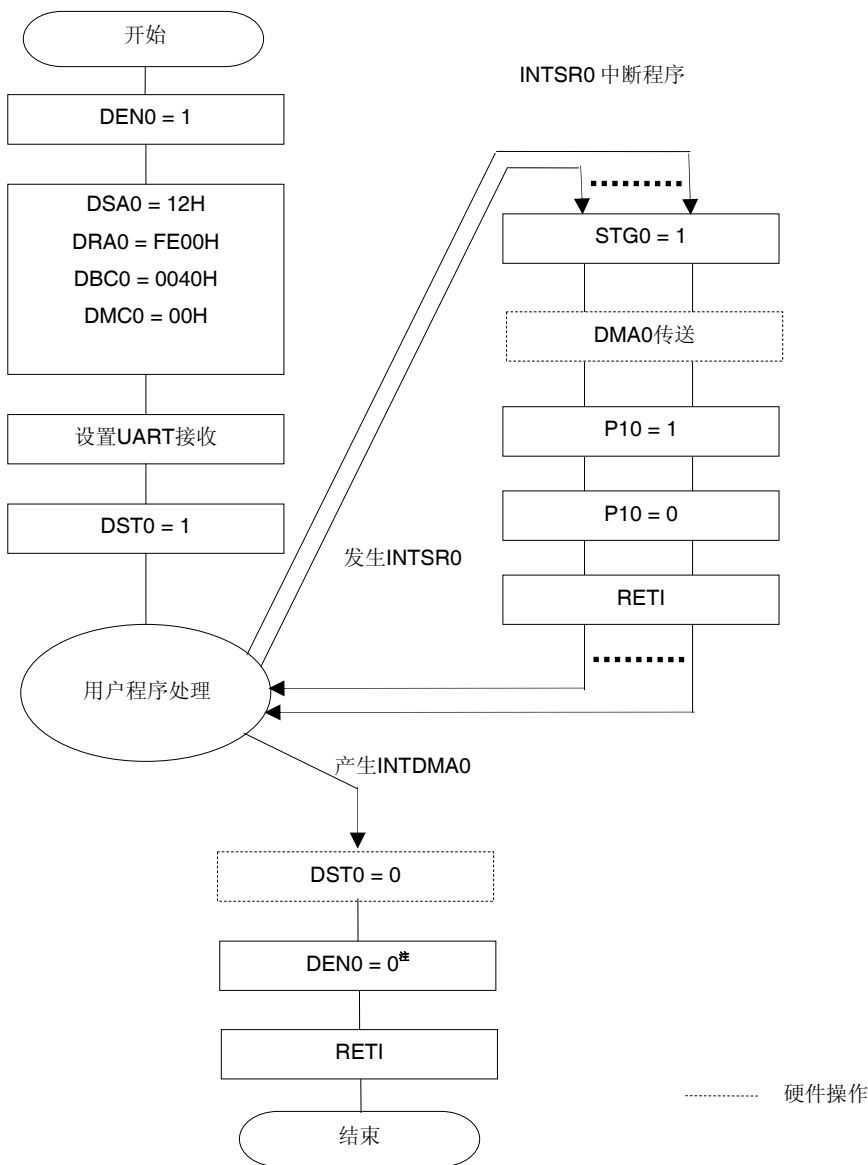
注 DMA 传送结束时，DST1 标志将被自动清除为 0。
 仅当 DST1 = 0 时允许写入 DEN1 标志。不等待产生 DMA1 中断(INTDMA1)就终止 DMA 传送时，先将 DST1 位设置为 0，然后将 DEN1 位设置为 0 (有关详情，请参阅 19.5.5 利用软件的强制结束)。

19.5.3 UART连续接收 + ACK发送

UART 连续接收 + ACK 发送的设置示例如下页的流程图所示。

- 从 UART0 连续接收数据，并在接收完成时将 ACK 输出至 P10
- DMA 传送时使用 DMA 通道 0
- DMA 触发源：软件触发(通过中断禁止 DMA 传送)
- 将 UART 接收数据寄存器 0 (RXD0)的 FFF12H 传送至 RAM 的 FFE00H 至 FFE3FH 的 64 字节

图 19-9. UART 连续接收 + ACK 发送的设置示例



注 DMA 传送结束时，DST0 标志将被自动清除为 0。

仅当 DST0 = 0 时允许写入 DENO 标志。不等待产生 DMA0 中断(INTDMA0)就终止 DMA 传送时，先将 DST0 位设置为 0，然后将 DENO 位设置为 0 (有关详情，请参阅 19.5.5 利用软件的强制结束)。

备注 这是将软件触发用作 DMA 触发源的一个示例。

如果不发送 ACK，仅从 UART 连续接收数据，则可利用 UART 接收结束中断(INTSR0)来启动 DMA，以进行数据的接收。

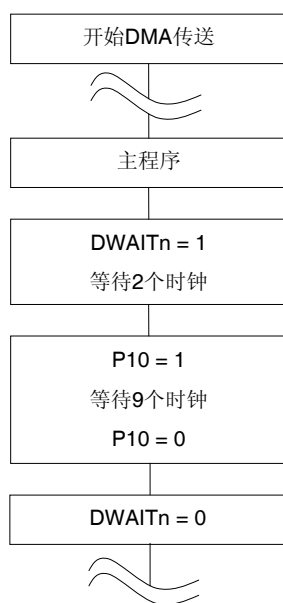
19.5.4 利用DWAITn位挂起DMA传送

开始 DMA 传送时，在执行指令时进行传送。此时，停止操作 CPU，并延迟 2 个时钟周期。如果这给设定系统的工作造成了问题，则可将 DWAITn 位设置为 1 从而挂起 DMA 传送。对于在 DMA 传送挂起期间发生的传送触发，在取消挂起状态后执行由其启动的 DMA 传送。然而，由于每个通道只能有一个传送触发可以被挂起，因此，即使在挂起状态期间一个通道产生了多个传送触发，在取消挂起状态后，只会执行一个 DMA 传送。

例如，当 P10 引脚输出宽度为操作频率的 10 个时钟的脉冲时，如果途中启动 DMA 传送，则增加至 12 个时钟。这种情况下，可通过将 DWAITn 位设置为 1，从而挂起 DMA 传送。

在将 DWAITn 位设置为 1 之后，需要经过两个时钟周期才挂起 DMA 传送。

图 19-10. 利用 DWAITn 位挂起 DMA 传送的设置示例



注意事项 在使用双 DMA 通道时，如果挂起 DMA 传送，则必须挂起双通道的 DMA 传送（将 DWAIT0 和 DWAIT1 位设置为 1）。如果在挂起通道的 DMA 传送期间执行了另一个通道的 DMA 传送，则可能无法挂起其他通道的 DMA 传送。

- 备注**
1. n: DMA 通道编号 (n = 0, 1)
 2. 1 时钟: 1/fCLK (fCLK: CPU 时钟)

19.5.5 利用软件的强制结束

通过软件将 DSTn 位设置为 0 之后，需要经过 2 个时钟周期，DMA 传送才会停止，之后，DSTn 位被设置为 0。若要通过软件强制终止 DMA 传送而不等待产生 DMAn 中断(INTDMA_n)，则需执行以下处理的任意一个。

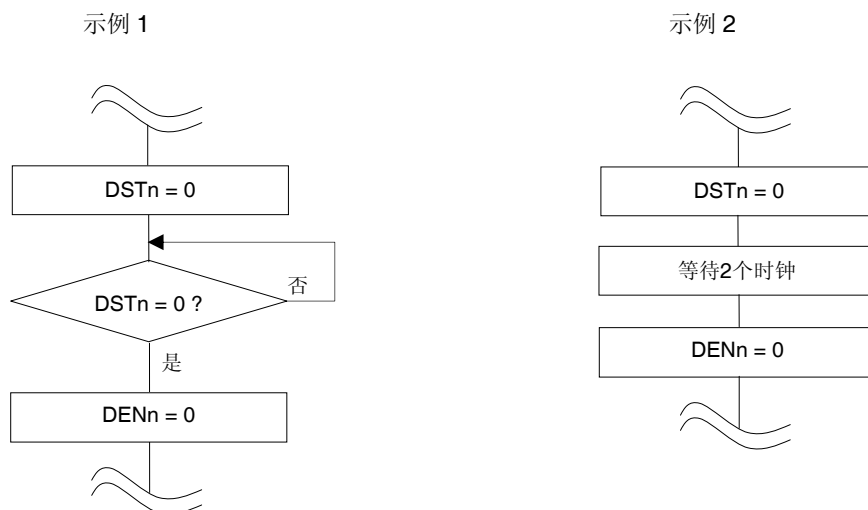
<当使用一个通道的 DMA 时>

- 使用软件将 DSTn 位设置为 0（通过 8 位操作指令写入 DRCn = 80H），通过轮询确认 DSTn 位已被清除为 0，然后将 DENn 位设置为 0（通过 8 位操作指令写入 DRCn = 00H）。
- 使用软件将 DSTn 位设置为 0（通过 8 位操作指令写入 DRCn = 80H），两个或多个时钟之后将 DENn 位设置为 0（通过 8 位操作指令写入 DRCn = 00H）。

<当使用双 DMA 通道时>

- 在使用双 DMA 通道时，若要通过软件强制终止 DMA 传送（通过将 DSTn 设置为 0），则须在将双通道的 DWAIT0 和 DWAIT1 位设置为 1 从而挂起 DMA 传送之后，将 DSTn 位清除为 0。接下来，将双通道的 DWAIT0 和 DWAIT1 位清除为 0，以解除挂起状态，然后将 DENn 位清除为 0。

图 19-11. DMA 传送的强制结束 (1/2)

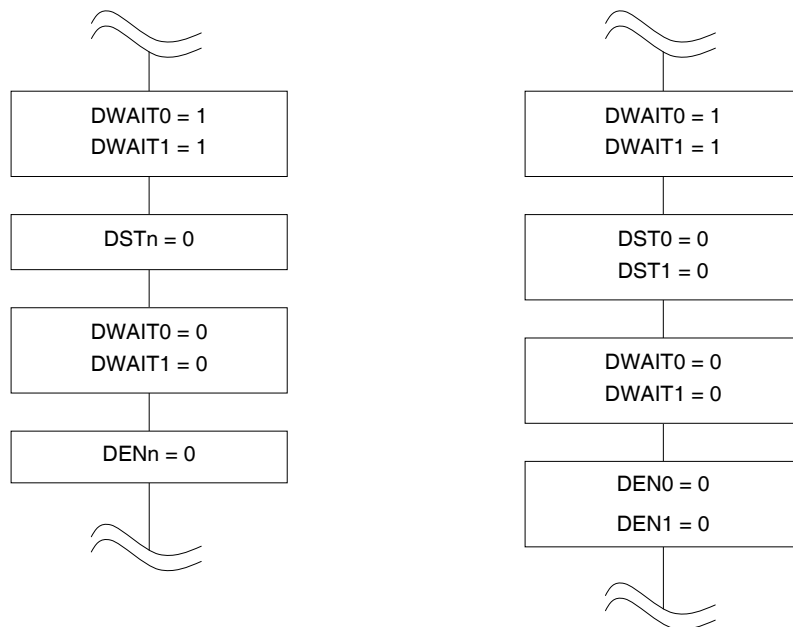


- 备注 1. n: DMA 通道编号 (n = 0, 1)
 2. 1 时钟: 1/f_{CLK} (f_{CLK}: CPU 时钟)

图 19-11. DMA 传送的强制结束 (2/2)

示例 3

- 使用双通道时，强制终止其中一个通道的 DMA 传送的步骤
- 使用双通道时，强制终止双通道的 DMA 传送的步骤



注意事项 在示例 3 中，系统不必在 $DWAITn$ 位被设置为 1 之后等待 2 个时钟周期。另外，不必在将 $DSTn$ 位清除为 0 之后等待 2 个时钟周期，因为从 $DSTn$ 位被清除为 0 到 $DENn$ 位被清除为 0，已经经过了超过 2 个时钟周期的时间。

- 备注**
1. n: DMA 通道编号 (n = 0, 1)
 2. 1 时钟: $1/f_{CLK}$ (f_{CLK} : CPU 时钟)

19.6 使用DMA控制器时的注意事项

(1) DMA 的优先级

在 DMA 传送过程中，即使另一个 DMA 通道产生了一个请求，该请求也会被挂起。挂起的 DMA 传送将在当前进行中的 DMA 传送结束后被启动。如果同时产生两个或多个 DMA 请求，DMA 通道 0 的优先级高于 DMA 通道 1。如果同时产生一个 DMA 请求和一个中断请求，则 DMA 传送优先，然后执行中断服务。

(2) DMA 响应时间

DMA 传送的响应时间如下所示。

表 19-3. DMA 传送的响应时间

	最短时间	最长时间
响应时间	3个时钟	10个时钟 [#]

注 执行内部 RAM 中的指令时，响应的最长时间为 16 个时钟周期。

注意事项 1. 以上响应时间不包括 DMA 传送需要的两个时钟周期。

2. 在执行 DMA 挂起指令（见 19.6 (4)）时，各条件下的最长响应时间满足该条件下挂起指令的执行时间。
3. 不能在最长响应时间与一个时钟周期之和的范围内连续触发同一通道的传送，否则，有可能会被忽略。

备注 1 时钟: 1/fCLK (fCLK: CPU 时钟)

(3) 待机模式下的操作

待机模式下 DMA 控制器的操作如下所示。

表 19-4. 待机模式下的 DMA 操作

状态	DMA操作
HALT模式	正常操作
STOP模式	停止操作 如果DMA传送和STOP指令的执行相冲突，DMA传送可能会遭到破坏。因此，在执行STOP指令之前，要停止DMA。

(4) DMA 挂起指令

紧随着以下指令之后即使产生了 DMA 请求，DMA 也会被挂起。

- CALL !addr16
- CALL \$!addr20
- CALL !!addr20
- CALL rp
- CALLT [addr5]
- BRK
- IF0L、IF0H、IF1L、IF1H、IF2L、IF2H、MK0L、MK0H、MK1L、MK1H、MK2L、MK2H、PR00L、PR00H、PR01L、PR01H、PR02L、PR02H、PR10L、PR10H、PR11L、PR11H、PR12L、PR12H 和 PSW 的各个寄存器的位操作指令
- 从 F0500H 至 F06FFH 的第二 SFR 地址空间内寄存器的存取指令
- 存取数据闪存的指令

(5) 指定了通用寄存器区域内或内部 RAM 区域外的地址时的操作

DMA RAM 地址寄存器 n (DRAn) 指示的地址将在 DMA 传送过程中递增。如果该地址递增至通用寄存器区域内或者超过内部 RAM 区域，则将执行以下操作。

- 从 SFR 传送至 RAM 模式下
该地址的数据丢失。
- 从 RAM 传送至 SFR 模式下
未定义的数据被传送至 SFR。

这两种情况下均可能发生功能故障或者系统损坏。因此，须确保地址在通用寄存器区以外的内部 RAM 区域之内。

**(6) 从 F0500H 至 F06FFH 的第二 SFR 地址空间内寄存器的存取**

在产生 DMA 传送的一个指令后，若存取上述寄存器，将插入一个时钟周期的等待。

(7) 存取数据闪存区域时的操作

- 由于存取数据闪存区域时将暂停 DMA 传送，所以必须添加 DMA 挂起指令。
如果 DMA 传送后的一条指令之后要存取数据闪存区域，则该条指令的执行需等待 3 个时钟周期。

指令 1

DMA 传送

指令 2 ← 发生 3 个时钟周期的等待。

MOV A, !DataFlash area

第二十章 中断功能

中断功能是指可将执行中的程序切换为其他所需处理的一种功能。分支处理结束后，返回到原来中断了的执行程序。中断源的数量因产品而异。

		20引脚	30引脚	32引脚	38引脚
可屏蔽 中断	外部	7	10	10	11
	内部	27	30	28	30

20.1 中断功能的类型

中断功能有以下两种。

(1) 可屏蔽中断

受到屏蔽控制的中断。通过设置优先级指定标志寄存器(PR00L、PR00H、PR01L、PR01H、PR02L、PR02H、PR10L、PR10H、PR11L、PR11H、PR12L、PR12H)，可以将可屏蔽中断的优先级分成四组。

产生高优先级中断时，对低优先级中断可以执行嵌套中断处理。如果同时产生优先级相同的两个或多个中断请求，则根据向量中断处理的默认优先级进行处理。关于默认优先级，请参阅表 20-1。

产生待机解除信号，解除 STOP、HALT 和 SNOOZE 模式。

可屏蔽中断分为外部中断请求和内部中断请求。

(2) 软件中断

这是通过执行 BRK 指令而产生的向量中断。即使在禁止中断时也能得到受理。软件中断不受中断优先级控制。

20.2 中断源和配置

中断源包括可屏蔽中断和软件中断。另外还具有最多七种的复位源(参阅表 20-1)。当因产生复位或各种中断请求而分支时，存储着程序起始地址的向量代码各为两个字节，因此，中断跳转目标的地址为 00000H 至 0FFFFH 的 64 K 地址。

表 20-1. 中断源列表(1/3)

中断类型	默认优先级 ^{注1}	中断源		内部/外部	向量表地址	基本配置类型 ^{注2}	38 引脚	32 引脚	30 引脚	20 引脚
		名称	触发							
可屏蔽	0	INTWDTI	看门狗定时器间隔 ^{注3} (溢出时间的 75%+1/2f _L)	内部	0004H	(A)	√	√	√	√
	1	INTLVI	检测电压 ^{注4}		0006H		√	√	√	√
	2	INTP0	检测引脚输入边沿	外部	0008H	(B)	√	√	√	√
	3	INTP3			000EH		√	-	-	-
	4	INTP4			0010H		√	√	√	-
	5	INTDMA0	DMA0 的传送结束	内部	001AH	(A)	√	√	√	√
	6	INTDMA1	DMA1 的传送结束		001CH		√	√	√	√
	7	INTST0	UART0 发送的传送结束或缓冲器空中断		001EH		√	√	√	√
		INTCSI00	CSI00 传送结束或缓冲器空中断				√	-	-	-
	8	INTSR0	UART0 接收传送结束		0020H		√	√	√	√
	9	INTSRE0	发生 UART0 接收通信错误		0022H		√	√	√	√
		INTTM01H	定时器通道 1 的计数或捕捉结束 (高 8 位定时器工作时)				√	√	√	√
	10	INTST1	UART1 发送的传送结束或缓冲器空中断		0024H		√	-	√	-
	11	INTSR1	UART1 接收传送结束		0026H		√	-	√	-
	12	INTSRE1	发生 UART1 接收通信错误		0028H		√	-	√	-
		INTTM03H	定时器通道 3 的计数或捕捉结束 (高 8 位定时器工作时)				√	√	√	√
13	INTIICA0	IICA0 通信结束	002AH		√		√	√	√	
14	INTTM00	定时器通道 0 的计数或捕捉结束	002CH		√		√	√	√	
15	INTTM01	定时器通道 1 的计数或捕捉结束 (16 位/低 8 位定时器工作时)	002EH		√		√	√	√	
16	INTTM02	定时器通道 2 的计数或捕捉结束	0030H		√		√	√	√	

- 注 1. 如果同时发生两个或更多可屏蔽中断，默认优先级将决定中断的顺序。0 表示最高优先级，40 表示最低优先级。
2. 基本配置类型(A)至(D)分别对应于图 20-1 中的(A)至(D)。
3. 当选项字节(000C0H)的位 7(WDTINT)被设置为 1。
4. 当电压检测电平寄存器(LVIS)的位 7(LVIMD)被清除为 0。

表 20-1. 中断源列表(2/3)

中断类型	默认优先级 ^{注1}	中断源		内部/外部	向量表地址	基本配置类型 ^{注2}	38 引脚	32 引脚	30 引脚	20 引脚
		名称	触发							
可屏蔽	17	INTTM03	定时器通道 3 的计数或捕捉结束(16 位/低 8 位定时器工作时)	内部	0032H	(A)	√	√	√	√
	18	INTAD	A/D 转换结束		0034H		√	√	√	√
	19	INTRTC	检测实时时钟的固定周期信号/报警一致		0036H		√	√	√	√
	20	INTIT	12 位间隔定时器检测间隔信号		0038H		√	√	√	√
	21	INTSTDL4	DALI/UART4 发送的传送结束或缓冲器空中断		003CH		√	√	√	√
	22	INTSRDL4	DALI/UART4 接收的传送结束		003EH		√	√	√	√
		INTSREDL4	发生 DALI/UART4 接收通信错误				√	√	√	√
	23	INTP20	检测引脚输入边沿	外部	0040H	(B)	√	√	√	√
		INTP22					√	√	√	√
	24	INTTM04	定时器通道 4 的计数或捕捉结束	内部	0042H	(A)	√	√	√	√
	25	INTTM05	定时器通道 5 的计数或捕捉结束		0044H		√	√	√	√
	26	INTTM06	定时器通道 6 的计数或捕捉结束		0046H		√	√	√	√
	27	INTTM07	定时器通道 7 的计数或捕捉结束		0048H		√	√	√	√
	28	INTCMP0	比较器 0 边沿检测	外部	004AH	(B)	√	√	√	√
	29	INTCMP1	比较器 1 边沿检测		004CH		√	√	√	√
	30	INTCMP2	比较器 2 边沿检测		004EH		√	√	√	√
	31	INTP9	检测引脚输入边沿		0050H		√	√	-	-
		INTCMP3	比较器 3 边沿检测				√	√	√	√
	32	INTP10	检测引脚输入边沿		0052H		√	√	-	-
		INTCMP4	比较器 4 边沿检测				√	√	√	-
33	INTP11	检测引脚输入边沿	0054H		√		√	√	-	
	INTCMP5	比较器 5 边沿检测			√		-	√	-	

注 1. 如果同时发生两个或更多可屏蔽中断，默认优先级将决定中断的顺序。零表示最高优先级，40 表示最低优先级。

2. 基本配置类型(A)至(D)对应于图 20-1 中的(A)至(D)。

表 20-1. 中断源列表(3/3)

中断类型	默认优先级 ^{注1}	中断源		内部/外部	向量表地址	基本配置类型 ^{注2}	38 引脚	32 引脚	30 引脚	20 引脚
		名称	触发							
可屏蔽	34	INTTMKB0	定时器 KB0 计数结束	内部	0056H	(A)	√	√	√	√
	35	INTTMKB1	定时器 KB1 计数结束		0058H	√	√	√	√	
	36	INTTMKB2	定时器 KB2 计数结束		005AH	√	√	√	—	
	37	INTTMKC0	定时器 KC0 计数结束		005CH	√	√	√	√	
	38	INTMD	除法运算结束/发生溢出		005EH	√	√	√	√	
	39	INTP21	检测引脚输入边沿	外部	0060H	(B)	√	√	√	√
							INTP23	√	√	√
40	INTFL	定序器结束中断 ^{注3}	内部	0062H	(A)	√	√	√	√	
软件	—	BRK	执行 BRK 指令	—	007EH	(D)	√	√	√	√
复位	—	RESET	RESET 引脚输入	—	0000H	—	√	√	√	√
		POR	上电复位				√	√	√	√
		LVD	电压检测 ^{注4}				√	√	√	√
		WDT	看门狗定时器的溢出				√	√	√	√
		TRAP	执行非法指令 ^{注5}				√	√	√	√
		IAW	存取非法存储器				√	√	√	√
		RPE	RAM 奇偶校验错误				√	√	√	√

注 1. 如果同时发生两个或更多可屏蔽中断，默认优先级将决定中断的顺序。0 表示最高优先级，40 表示最低优先级。

2. 基本配置类型(A)至(D)分别对应于图 20-1 中的(A)至(D)。

3. 仅限于自编程序使用。

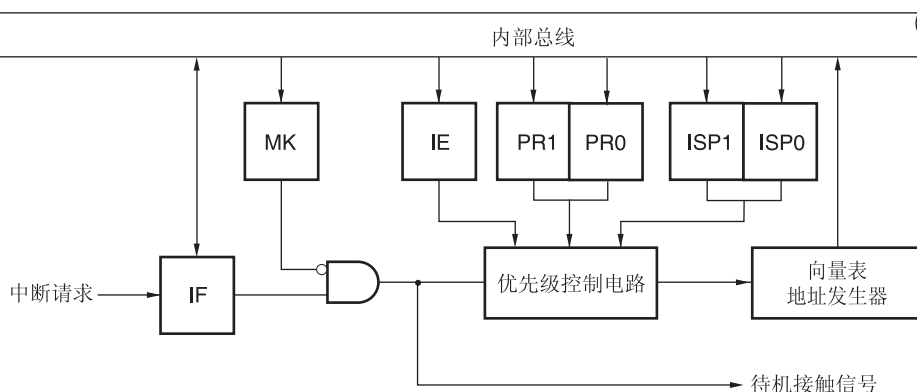
4. 当电压检测电平寄存器(LVIS)的位 7(LVIMD)被设置为 1。

5. 执行指令码 FFH 时发生。

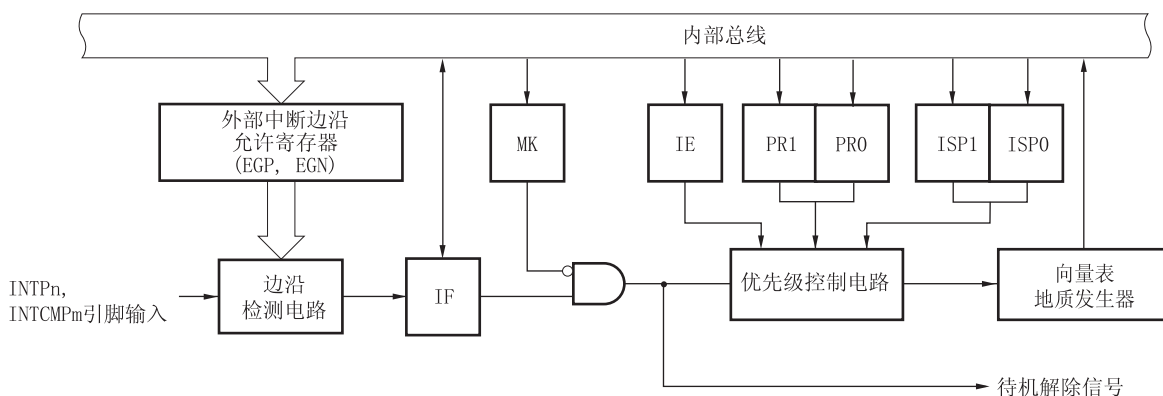
在线仿真器或片上调试仿真器的仿真过程中，不会因执行非法指令发生内部复位。

图 20-1. 中断功能的基本配置 (1/2)

(A) 内部可屏蔽中断



(B) 外部可屏蔽中断(INTPn, INTCMPm)

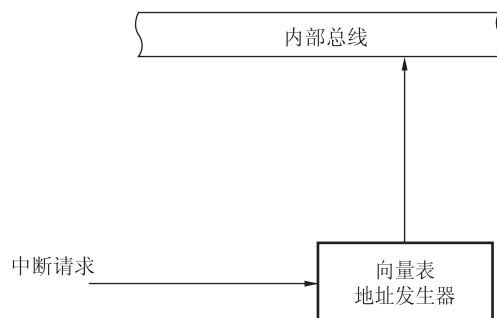


- IF: 中断请求标志
- IE: 中断允许标志
- ISP0: 受理中断的优先级标志 0
- ISP1: 受理中断的优先级标志 1
- MK: 中断屏蔽标志
- PR0: 优先级指定标志 0
- PR1: 优先级指定标志 1

- 备注
- 20 引脚: n = 0, 20, 21, 22, m = 0 至 3
 - 30 引脚: n = 0, 4, 11, 20 至 23, m = 0 至 5
 - 32 引脚: n = 0, 4, 9 至 11, 20 至 23, m = 0 至 4
 - 38 引脚: n = 0, 3, 4, 9 至 11, 20 至 23, m = 0 至 5

图 20-1. 中断功能的基本配置 (2/2)

(C) 软件中断



- IF: 中断请求标志
- IE: 中断允许标志
- ISP0: 受理中断的优先级标志 0
- ISP1: 受理中断的优先级标志 1
- MK: 中断屏蔽标志
- PR0: 优先级指定标志 0
- PR1: 优先级指定标志 1

20.3 控制中断功能的寄存器

以下寄存器用于控制中断功能。

- 中断请求标志寄存器 (IF0L、IF0H、IF1L、IF1H、IF2L、IF2H)
- 中断屏蔽标志寄存器 (MK0L、MK0H、MK1L、MK1H、MK2L、MK2H)
- 优先级指定标志寄存器 (PR00L、PR00H、PR01L、PR01H、PR02L、PR02H、PR10L、PR10H、PR11L、PR11H、PR12L、PR12H)
- 外部中断上升沿允许寄存器 (EGP0、EGP1、EGP2)
- 外部中断下降沿允许寄存器 (EGN0、EGN1、EGN2)
- 程序状态字 (PSW)
- 中断屏蔽标志寄存器 0 (INTMK0)
- 中断监视标志寄存器 0 (INTMF0)

各中断请求源对应的中断请求标志，中断屏蔽标志和优先级指定标志如表 20-2 所示。

表 20-2. 中断请求源对应的标志(1/3)

中断源	中断请求标志		中断屏蔽标志		优先级指定标志	寄存器	38 引脚	32 引脚	30 引脚	20 引脚
	寄存器	寄存器	寄存器	寄存器						
INTWDTI	WDTIIF	IF0L	WDTIMK	MK0L	WDTIPR0, WDTIPR1	PR00L, PR10L	√	√	√	√
INTLVI	LVIF		LVIMK		LVIPR0, LVIPR1		√	√	√	√
INTP0	PIF0		PMK0		PPR00, PPR10		√	√	√	√
INTP3	PIF3		PMK3		PPR03, PPR13		√	–	–	–
INTP4	PIF4		PMK4		PPR04, PPR14		√	√	√	–
INTDMA0	DMAIF0	IF0H	DMAMK0	MK0H	DMAPR00, DMAPR10	PR00H, PR10H	√	√	√	√
INTDMA1	DMAIF1		DMAMK1		DMAPR01, DMAPR11		√	√	√	√
INTST0 ^{注1}	STIF0 ^{注1}		STMK0 ^{注1}		STPR00, STPR10 ^{注1}		√	√	√	√
INTCSI00 ^{注1}	CSIIF00 ^{注1}		CSIMK00 ^{注1}		CSIPR000, CSIPR100 ^{注1}		√	–	–	–
INTSR0	SRIF0		SRMK0		SRPR00, SRPR10		√	√	√	√
INTSRE0 ^{注2}	SREIF0 ^{注2}		SREMK0 ^{注2}		SREPR00, SREPR10 ^{注2}		√	√	√	√
INTTM01H ^{注2}	TMIF01H ^{注2}		TMMK01H ^{注2}		TMPR001H, TMPR101H ^{注2}		√	√	√	√

- 注 1. 如果产生中断源 INTST0 和 INTCSI00 二者之一，设置 IF0H 寄存器的位 5 为 1。MK0H、PR00H 和 PR10H 寄存器的位 5 支持这两种中断源。
- 注 2. 不要同时使用 UART0 和 TAU0 的通道 1(高 8 位定时器工作时)，因为它们共用中断请求源标志。如果产生中断源 INTSRE0 和 INTTM01H 二者之一，设置 IF0H 寄存器的位 7 为 1。MK0H、PR00H 和 PR10H 寄存器的位 7 支持这两种中断源。

表 20-2. 中断请求源对应的标志(2/3)

中断源	中断请求标志		中断屏蔽标志		优先指定标志	寄存器	38 引脚	32 引脚	30 引脚	20 引脚
		寄存器		寄存器						
INTST1	STIF1	IF1L	STMK1	MK1L	STPR01, STPR11	PR01L, PR11L	√	-	√	-
INTSR1	SRIF1		SRMK1		SRPR01, SRPR11		√	-	√	-
INTSRE1 ^{注1}	SREIF1 ^{注1}		SREMK1 ^{注1}		SREPR01, SREPR11 ^{注1}		√	-	√	-
INTTM03H ^{注1}	TMIF03H ^{注1}		TMMK03H ^{注1}		TMPR003H, TMPR103H ^{注1}		√	√	√	√
INTIICA0	IICAIF0		IICAMK0		IICAPR00, IICAPR10		√	√	√	√
INTTM00	TMIF00		TMMK00		TMPR000, TMPR100		√	√	√	√
INTTM01	TMIF01		TMMK01		TMPR001, TMPR101		√	√	√	√
INTTM02	TMIF02		TMMK02		TMPR002, TMPR102		√	√	√	√
INTTM03	TMIF03		TMMK03		TMPR003, TMPR103		√	√	√	√
INTAD	ADIF	IF1H	ADMK	MK1H	ADPR0, ADPR1	PR01H, PR11H	√	√	√	√
INTRTC	RTCIF		RTCMK		RTCPR0, RTCPR1		√	√	√	√
INTIT	ITIF		ITMK		ITPR0, ITPR1		√	√	√	√
INTSTDL4	STDLIF4		STDLMK4		STDLPR04, STDLP14		√	√	√	√
INTSRDL4 ^{注2}	SRDLIF4 ^{注2}		SRDLMK4 ^{注2}		SRDLPR04, SRDLPR14 ^{注2}		√	√	√	√
INTSREDL4 ^{注2}	SREDLIF4 ^{注2}		SREDLMK4 ^{注2}		SREDLPR04, SREDLPR14 ^{注2}		√	√	√	√
INTP20 ^{注3}	PIF20 ^{注3}		PMK20 ^{注3}		PPR020, PPR120 ^{注3}		√	√	√	√
INTP22 ^{注3}	PIF22 ^{注3}		PMK22 ^{注3}		PPR022, PPR122 ^{注3}		√	√	√	√
INTTM04	TMIF04		TMMK04		TMPR004, TMPR104		√	√	√	√

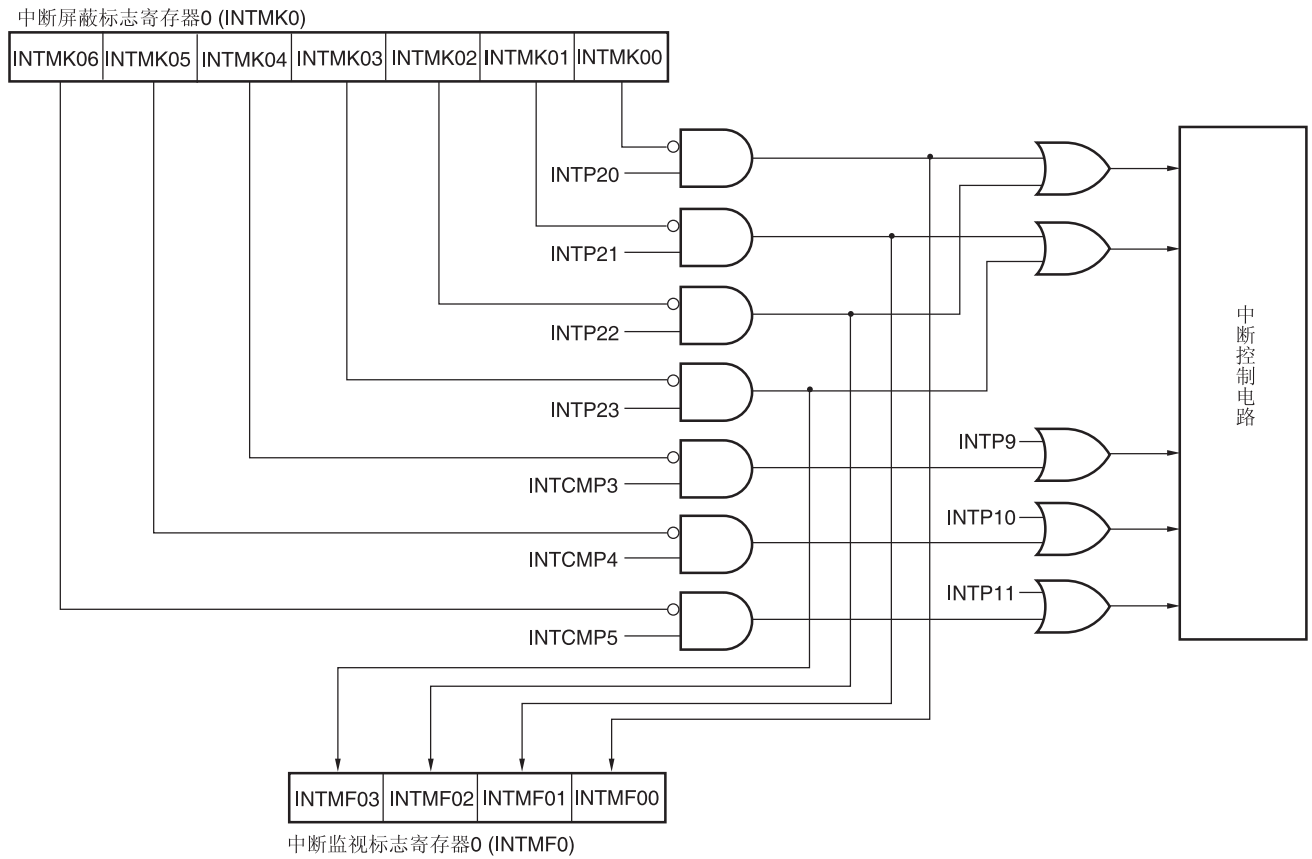
- 注 1. 不要同时使用 UART1 和 TAU0 的通道 3 (高 8 位定时器工作时), 因为它们共用中断请求源标志。如果产生中断源 INTSRE1 和 INTTM03H 二者之一, 设置 IF1L 寄存器的位 2 为 1。MK1L、PR01L 和 PR11L 寄存器的位 2 支持这两种中断源。
2. DALI/UART4 的 INTSRDL4 (传送结束中断)和 INTSREDL4 (错误中断)共用中断请求源标志。如果产生中断源 INTSRDL4 和 INTSREDL4 二者之一, 设置 IF1H 寄存器的位 5 为 1。MK1H、PR01H 和 PR11H 寄存器的位 5 支持这两种中断源。当产生错误中断 INTSREDL4 时, 串行状态寄存器 41 (SSR41)的错误标志被设置。依据错误标志的设置, 可以判断产生的是 INTSRDL4 (传送结束中断)还是 INTSREDL4 (错误中断)。
3. 不要同时使用外部中断 INTP20 和 INTP22, 因为它们共用中断请求源标志。如果产生中断源 INTP20 和 INTP22 二者之一, 设置 IF1H 寄存器的位 6 为 1。MK1H、PR01H 和 PR11H 寄存器的位 6 支持这两种中断源。可以通过中断屏蔽标志寄存器 0 (INTMK0)来屏蔽外部中断 INTP20 和 INTP22 的中断请求。而且, 可以通过中断监视标志寄存器 0 (INTMF0)来验证外部中断的产生状态。(参阅图 20-2。)

表 20-2. 中断请求源对应的标志(3/3)

中断源	中断请求标志		中断屏蔽标志		优先指定标志		38 引脚	32 引脚	30 引脚	20 引脚
		寄存器		寄存器		寄存器				
INTTM05	TMIF05	IF2L	TMMK05	MK2L	TMPR005, TMPR105	PR02L, PR12L	√	√	√	√
INTTM06	TMIF06		TMMK06		TMPR006, TMPR106		√	√	√	√
INTTM07	TMIF07		TMMK07		TMPR007, TMPR107		√	√	√	√
INTCMP0	CMPIF0		CMPMK0		CMPPR00, CMPPR10		√	√	√	√
INTCMP1	CMPIF1		CMPMK1		CMPPR01, CMPPR11		√	√	√	√
INTCMP2	CMPIF2		CMPMK2		CMPPR02, CMPPR12		√	√	√	√
INTP9 ^{注1}	PIF9 ^{注1}		PMK9 ^{注1}		PPR09, PPR19 ^{注1}		√	√	-	-
INTCMP3 ^{注1}	CMPIF3 ^{注1}		CMPMK3 ^{注1}		CMPPR03, CMPPR13 ^{注1}		√	√	√	√
INTP10 ^{注2}	PIF10 ^{注2}	IF2L	PMK10 ^{注2}	MK2L	PPR010, PPR110 ^{注2}	PR02L, PR12L	√	√	-	-
INTCMP4 ^{注2}	CMPIF4 ^{注2}		CMPMK4 ^{注2}		CMPPR04, CMPPR14 ^{注2}		√	√	√	-
INTP11 ^{注3}	PIF11 ^{注3}	IF2H	PMK11 ^{注3}	MK2H	PPR011, PPR111 ^{注3}	PR02H, PR12H	√	√	√	-
INTCMP5 ^{注3}	CMPIF5 ^{注3}		CMPMK5 ^{注3}		CMPPR05, CMPPR15 ^{注3}		√	-	√	-
INTTMKB0	TMKBIF0		TMKBMK0		TMKBPR00, TMKBPR10		√	√	√	√
INTTMKB1	TMKBIF1		TMKBMK1		TMKBPR01, TMKBPR11		√	√	√	√
INTTMKB2	TMKBIF2		TMKBMK2		TMKBPR02, TMKBPR12		√	√	√	-
INTTMKC0	TMKCIF0		TMKCMK0		TMKCPR00, TMKCPR10		√	√	√	√
INTMD	MDIF		MDMK		MDPR0, MDPR1		√	√	√	√
INTP21 ^{注4}	PIF21 ^{注4}		PMK21 ^{注4}		PPR021, PPR121 ^{注4}		√	√	√	√
INTP23 ^{注4}	PIF23 ^{注4}	PMK23 ^{注4}	PPR023, PPR123 ^{注4}	√	√	√	-			
INTFL	FLIF		FLMK		FLPR0, FLPR1		√	√	√	√

- 注 1. 不要同时使用外部中断 INTP9 和比较器 3，因为它们共用中断请求源标志。如果产生中断源 INTP9 和 INTCMP3 二者之一，设置 IF2L 寄存器的位 6 为 1。MK2L、PR02L 和 PR12L 寄存器的位 6 支持这两种中断源。而且，可以通过中断屏蔽标志寄存器 0 (INTMK0) 来屏蔽 INTCMP3 的中断请求（参阅图 20-2）。
2. 不要同时使用外部中断 INTP10 和比较器 4，因为它们共用中断请求源标志。如果产生中断源 INTP10 和 INTCMP4 二者之一，设置 IF2L 寄存器的位 7 为 1。MK2L、PR02L 和 PR12L 寄存器的位 7 支持这两种中断源。而且，可以通过中断屏蔽标志寄存器 0 (INTMK0) 来屏蔽 INTCMP4 的中断请求（参阅图 20-2）。
3. 不要同时使用外部中断 INTP11 和比较器 5，因为它们共用中断请求源标志。如果产生中断源 INTP11 和 INTCMP5 二者之一，设置 IF2H 寄存器的位 0 为 1。MK2H、PR02H 和 PR12H 寄存器的位 0 支持这两种中断源。而且，可以通过中断屏蔽标志寄存器 0 (INTMK0) 来屏蔽 INTCMP4 的中断请求（参阅图 20-2）。
4. 不要同时使用外部中断 INTP21 和 INTP23，因为它们共用中断请求源标志。如果产生中断源 INTP21 和 INTP23 二者之一，设置 IF2H 寄存器的位 6 为 1。MK2H、PR02H 和 PR12H 寄存器的位 6 支持这两种中断源。可以通过中断屏蔽标志寄存器 0 (INTMK0) 来屏蔽外部中断 INTP21 和 INTP23 的中断请求。而且，可以通过中断监视标志寄存器 0 (INTMFO) 来验证外部中断的产生状态（参阅图 20-2）。

图 20-2. 外部中断 INTP20 至 INTP23 以及比较器中断 INTCMP3 至 INTCMP5 的中断控制图



(1) 中断请求标志寄存器(IF0L, IF0H, IF1L, IF1H, IF2L, IF2H)

在产生相应的中断请求或者执行一条指令时，中断请求标志被设置为 1。受理中断请求或者产生复位信号时，执行指令后该寄存器被清除为 0。

当一个中断被受理时，中断请求标志将自动清零，然后进行中断处理。

使用 1 位或 8 位存储器操作指令设置 IF0L、IF0H、IF1L、IF1H、IF2L 和 IF2H 寄存器。当 IF0L 和 IF0H 寄存器、IF1L 和 IF1H 寄存器以及 IF2L 和 IF2H 寄存器相结合以构成 16 位寄存器 IF0、IF1 和 IF2 时，可以用 16 位存储器操作指令进行设置。

产生复位信号后，该寄存器被清除为 00H。

备注 执行向该寄存器写入数据的指令时，该指令执行时钟数将增加 2 个时钟。

图 20-3. 中断请求标志寄存器 (IF0L、IF0H、IF1L、IF1H、IF2L、IF2H)的格式 (38 引脚) (1/2)

地址: FFFE0H 复位后: 00H R/W

符号	7	<6>	<5>	4	3	<2>	<1>	<0>
IF0L	0	PIF4	PIF3	0	0	PIF0	LVIIIF	WDTIIF

地址: FFFE1H 复位后: 00H R/W

符号	<7>	<6>	<5>	<4>	<3>	2	1	0
IF0H	SREIF0	SRIF0	STIF0	DMAIF1	DMAIF0	0	0	0
	TMIF01H		CSIIIF00					

地址: FFFE2H 复位后: 00H R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF1L	TMIF03	TMIF02	TMIF01	TMIF00	IICAIF0	SREIF1	SRIF1	STIF1
						TMIF03H		

地址: FFFE3H 复位后: 00H R/W

符号	<7>	<6>	<5>	<4>	3	<2>	<1>	<0>
IF1H	TMIF04	PIF20	SRDLIF4	STDLIF4	0	ITIF	RTCIF	ADIF
		PIF22	SREDLIF4					

地址: FFFD0H 复位后: 00H R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF2L	PIF10	PIF9	CMPIF2	CMPIF1	CMPIF0	TMIF07	TMIF06	TMIF05
	CMPIF4	CMPIF3						

图 20-3. 中断请求标志寄存器 (IF0L、IF0H、IF1L、IF1H、IF2L、IF2H)的格式 (38 引脚) (2/2)

地址: FFFD1H 复位后: 00H R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF2H	FLIF	PIF21 PIF23	MDIF	TMKCIF0	TMKBIF2	TMKBIF1	TMKBIF1	PIF11 CMPIF5
XXIFX	中断请求标志							
0	不产生中断请求信号。							
1	产生中断请求, 处于中断请求状态。							

注意事项 1. 38 引脚产品的位布局如上所示。配置位因产品而异。关于每种产品配置位的详情, 请参阅表 20-2。必须将未配置的位清除为 0。

- 2.** 在操作中中断请求标志寄存器的标志时, 要使用 1 位存储器操作指令 (CLR1)。当以 C 语言描述时, 要使用位操作指令, 如“IF0L.0 = 0;”或“_asm(“clr1 IF0L, 0”);”, 因为编译后的汇编程序必须为 1 位存储器操作指令 (CLR1)。

如果利用 8 位存储器操作指令 (如“IF0L &= 0xfe;”) 以 C 语言描述程序并进行编译, 则将成为有三条指令的汇编程序。

```
mov a, IF0L
and a, #0FEH
mov IF0L, a
```

这种情况下, 即使在“mov a, IF0L”与“mov IF0L, a”之间将同一中断请求标志寄存器 (IF0L) 的另一位的请求标志设置为 1, 该标志也会在“mov IF0L, a”时清除为 0。因此, 在以 C 语言使用 8 位存储器操作指令时, 必须多加注意。

(2) 中断屏蔽标志寄存器 (MK0L, MK0H, MK1L, MK1H, MK2L, MK2H)

中断屏蔽标志用于允许/禁止相应的可屏蔽中断处理。

使用 1 位或 8 位存储器操作指令设置 MK0L、MK0H、MK1L、MK1H、MK2L 和 MK2H 寄存器。当 MK0L 和 MK0H 寄存器、MK1L 和 MK1H 寄存器以及 MK2L 和 MK2H 寄存器相结合以构成 16 位寄存器 MK0、MK1 和 MK2 时, 可以用 16 位存储器操作指令进行设置。

产生复位信号后, 该寄存器被设置为 FFH。

备注 执行向该寄存器写入数据的指令时, 该指令执行时钟数将增加 2 个时钟。

图 20-4. 中断屏蔽标志寄存器 (MK0L、MK0H、MK1L、MK1H、MK2L、MK2H)的格式(38 引脚)

地址: FFFE4H 复位后: FFH R/W

符号	7	<6>	<5>	4	3	<2>	<1>	<0>
MK0L	1	PMK4	PMK3	1	1	PMK0	LVIMK	WDTIMK

地址: FFFE5H 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	2	1	0
MK0H	SREMK0 TMMK01H	SRMK0	STMK0 CSIMK00	DMAMK1	DMAMK0	1	1	1

地址: FFFE6H 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
MK1L	TMMK03	TMMK02	TMMK01	TMMK00	IICAMK0	SREMK1 TMMK03H	SRMK1	STMK1

地址: FFFE7H 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	3	<2>	<1>	<0>
MK1H	TMMK04	PMK20 PMK22	SRDLMK4 SREDLMK4	STDLMK4	1	ITMK	RTCMK	ADMK

地址: FFFD4H 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
MK2L	PMK10 CMPMK4	PMK9 CMPMK3	CMPMK2	CMPMK1	CMPMK0	TMMK07	TMMK06	TMMK05

地址: FFFD5H 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
MK2H	FLMK	PMK21 PMK23	MDMK	TMKCMK0	TMKBMK2	TMKBMK1	TMKBMK0	PMK11 CMPMK5

XXMKX	中断处理的控制							
0	允许中断处理							
1	禁止中断处理							

注意事项 38 引脚产品的位布局如上所示。配置位因产品而异。关于每种产品配置位的详情，请参阅表 20-2。必须将未配置的位设置为 1。

(3) 优先级指定标志寄存器(PR00L, PR00H, PR01L, PR01H, PR02L, PR02H, PR10L, PR10H, PR11L, PR11H, PR12L, PR12H)

优先级指定标志寄存器用于设置相应的可屏蔽中断的优先级。

通过组合使用 PR0xy 和 PR1xy 寄存器(xy = 0L、0H、1L、1H、2L 或 2H)来设置优先级。

使用 1 位或 8 位存储器操作指令设置 PR00L、PR00H、PR01L、PR01H、PR02L、PR02H、PR10L、PR10H、PR11L、PR11H、PR12L 和 PR12H 寄存器。如果 PR00L 和 PR00H 寄存器、PR01L 和 PR01H 寄存器、PR02L 和 PR02H 寄存器、PR10L 和 PR10H 寄存器、PR11L 和 PR11H 寄存器以及 PR12L 和 PR12H 寄存器相结合以构成 16 位寄存器 PR00、PR01、PR02、PR10、PR11 和 PR12，则可以用 16 位存储器操作指令进行设置。

产生复位信号后，该寄存器被设置为 FFH。

备注 执行向该寄存器写入数据的指令时，该指令执行时钟数将增加 2 个时钟。

图 20-5. 优先级指定标志寄存器(PR00L、PR00H、PR01L、PR01H、PR02L、PR02H、PR10L、PR10H、PR11L、PR11H、PR12L、PR12H)的格式 (38 引脚) (1/2)

地址: FFFE8H 复位后: FFH R/W

符号	7	<6>	<5>	4	3	<2>	<1>	<0>
PR00L	1	PPR04	PPR03	1	1	PPR00	LVIPR0	WDTIPR0

地址: FFFECH 复位后: FFH R/W

符号	7	<6>	<5>	4	3	<2>	<1>	<0>
PR10L	1	PPR14	PPR13	1	1	PPR10	LVIPR1	WDTIPR1

地址: FFFE9H 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	2	1	0
PR00H	SREPR00 TMPR001H	SRPR00	STPR00 CSIPR000	DMAPR01	DMAPR00	1	1	1

地址: FFFEDH 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	2	1	0
PR10H	SREPR10 TMPR101H	SRPR10	STPR10 CSIPR100	DMAPR11	DMAPR10	1	1	1

地址: FFFEAH 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PR01L	TMPR003	TMPR002	TMPR001	TMPR000	IICAPR00	SREPR01 TMPR003H	SRPR01	STPR01

地址: FFFEEH 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PR11L	TMPR103	TMPR102	TMPR101	TMPR100	IICAPR10	SREPR11 TMPR103H	SRPR11	STPR11

图 20-5. 优先级指定标志寄存器(PR00L、PR00H、PR01L、PR01H、PR02L、PR02H、PR10L、PR10H、PR11L、PR11H、PR12L、PR12H)的格式 (38 引脚) (2/2)

地址: FFFEBH 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	3	<2>	<1>	<0>
PR01H	TMPR004	PPR020 PPR120	SRDLPR04 SREDLPR04	STDLP04	1	ITPR0	RTCPR0	ADPR0

地址: FFFEFH 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	3	<2>	<1>	<0>
PR11H	TMPR104	PPR022 PPR122	SRDLPR14 SREDLPR14	STDLP14	1	ITPR1	RTCPR1	ADPR1

地址: FFFD8H 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PR02L	PPR010 CMPPR04	PPR09 CMPPR03	CMPPR02	CMPPR01	CMPPR00	TMPR007	TMPR006	TMPR005

地址: FFFDCH 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PR12L	PPR110 CMPPR14	PPR19 CMPPR13	CMPPR12	CMPPR11	CMPPR10	TMPR107	TMPR106	TMPR105

地址: FFFD9H 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PR02H	FLPR0	PPR021 PPR023	MDPR0	TMKCPR00	TMKBPR02	TMKBPR01	TMKBPR00	PPR011 CMPR05

地址: FFFDDH 复位后: FFH R/W

符号	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PR12H	FLPR1	PPR121 PPR123	MDPR1	TMKCPR10	TMKBPR12	TMKBPR11	TMKBPR10	PPR111 CMPR15

XXPR1X	XXPR0X	选择优先等级
0	0	指定等级 0 (高优先等级)
0	1	指定等级 1
1	0	指定等级 2
1	1	指定等级 3 (低优先等级)

注意事项 38 引脚产品的位布局如上所示。配置位因产品而异。关于每种产品配置位的详情，请参阅表 20-2。必须将未配置的位设置为 1。

(4) 外部中断上升沿允许寄存器(EGP0, EGP1, EGP2), 外部中断下降沿允许寄存器(EGN0, EGN1, EGN2)

该寄存器指定 INTP0 至 INTP11 的有效边沿。

使用 1 位或 8 位存储器操作指令设置 EGP0 至 EGP2、EGN0 至 EGN2 寄存器。

产生复位信号后, 该寄存器被清除为 00H。

图 20-6. 外部中断上升沿允许寄存器(EGP0, EGP1, EGP2)和外部中断下降沿允许寄存器(EGN0, EGN1, EGN2)的格式(38 引脚)

地址: FFF38H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
EGP0	0	0	0	EGP4	EGP3	0	0	EGP0

地址: FFF39H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
EGN0	0	0	0	EGN4	EGN3	0	0	EGN0

地址: FFF3AH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
EGP1	0	0	0	0	EGP11	EGP10	EGP9	0

地址: FFF3BH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
EGN1	0	0	0	0	EGN11	EGN10	EGN9	0

地址: F0518H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
EGP2	0	0	0	0	EGP23	EGP22	EGP21	EGP20

地址: F0519H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
EGN2	0	0	0	0	EGN23	EGN22	EGN21	EGN20

EGPn	EGNn	选择 INTPn 引脚的有效边沿 (n = 0, 3, 4, 9 至 11, 20 至 23)
0	0	禁止检测边沿
0	1	下降沿
1	0	上升沿
1	1	上升和下降沿

对应于 EGPn 和 EGNn 位的端口如表 20-3 所示。

表 20-3. 对应于 EGPn 位和 EGNn 位的中断请求信号

检测允许位		边沿检测端口	中断请求信号	38 引脚	32 引脚	30 引脚	20 引脚
EGP0	EGN0	P137	INTP0	√	√	√	√
EGP3	EGN3	P30	INTP3	√	—	—	—
EGP4	EGN4	P31	INTP4	√	√	√	—
EGP9	EGN9	P75	INTP9	√	√	—	—
EGP10	EGN10	P76	INTP10	√	√	—	—
EGP11	EGN11	P77	INTP11	√	√	√	—
EGP20	EGN20	P10	INTP20	√	√	√	√
EGP21	EGN21	P11	INTP21	√	√	√	√
EGP22	EGN22	P200	INTP22	√	√	√	√
EGP23	EGN23	P206	INTP23	√	√	√	—

注意事项 从外部中断功能切换至端口功能时，可能会检测到边沿，因此将 EGPn 和 EGNn 位清除为 0 之后再选择端口模式。

备注 n = 0, 3, 4, 9 至 11, 20 至 23

(5) 中断屏蔽标志寄存器 0 (INTMK0)

中断屏蔽标志寄存器 0 (INTMK0)用于屏蔽 INTP20 至 INTP23 和 INTCMP3 至 INTCMP5 的中断请求标志。

使用 1 位或 8 位存储器操作指令设置 INTMK0 寄存器。

产生复位信号后，该寄存器被设置为 FFH。

图 20-7. 中断屏蔽标志寄存器 0 (INTMK0) (38 引脚)

地址: F05C2H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
INTMK0	1	INTMK06	INTMK05	INTMK04	INTMK03	INTMK02	INTMK01	INTMK00

INTMK06	INTCMP5 中断请求信号的输出允许/禁止
0	允许输出
1	禁止输出

INTMK05	INTCMP4 中断请求信号的输出允许/禁止
0	允许输出
1	禁止输出

INTMK04	INTCMP3 中断请求信号的输出允许/禁止
0	允许输出
1	禁止输出

INTMK03	INTP23 中断请求信号的输出允许/禁止
0	允许输出
1	禁止输出

INTMK02	INTP22 中断请求信号的输出允许/禁止
0	允许输出
1	禁止输出

INTMK01	INTP21 中断请求信号的输出允许/禁止
0	允许输出
1	禁止输出

INTMK00	INTP20 中断请求信号的输出允许/禁止
0	允许输出
1	禁止输出

(6) 中断监视标志寄存器 0 (INTMF0)

中断监视标志寄存器 0 (INTMF0)用于监视 INTP20 至 INTP23 的中断请求信号的产生状态。

通过产生 INTP20 至 INTP23 的中断请求信号，将 INTMF0 寄存器的对应标志设置为 1。然而，该标志不自动清除为 0。因此必须通过软件来清除该标志。

而且，在清除为(0)之前不输入 INTP20 至 INTP23 的中断请求信号。

可以使用 1 位存储器操作指令来设置或读取 INTMF0 寄存器，或者使用 8 位存储器操作指令来读取 INTMF0 寄存器。

清除为(0)时，执行 1 位存储器操作指令。

产生复位信号后，该寄存器被清除为 00H。

图 20-8. 中断监视标志寄存器 0 (INTMF0) (38 引脚)

地址: F05C3H 复位后: 00H R/W ^注

符号	7	6	5	4	3	2	1	0
INTMF0	0	0	0	0	INTMF03	INTMF02	INTMF01	INTMF00

INTMF03	显示 INTP23 的中断请求信号的产生状态
0	不产生中断请求信号。
1	产生中断请求信号。

INTMF02	显示 INTP22 的中断请求信号的产生状态
0	不产生中断请求信号。
1	产生中断请求信号。

INTMF01	显示 INTP21 的中断请求信号的产生状态
0	不产生中断请求信号。
1	产生中断请求信号。

INTMF00	显示 INTP20 的中断请求信号的产生状态
0	不产生中断请求信号。
1	产生中断请求信号。

注 使用 8 位存储器操作指令时，仅能读取。

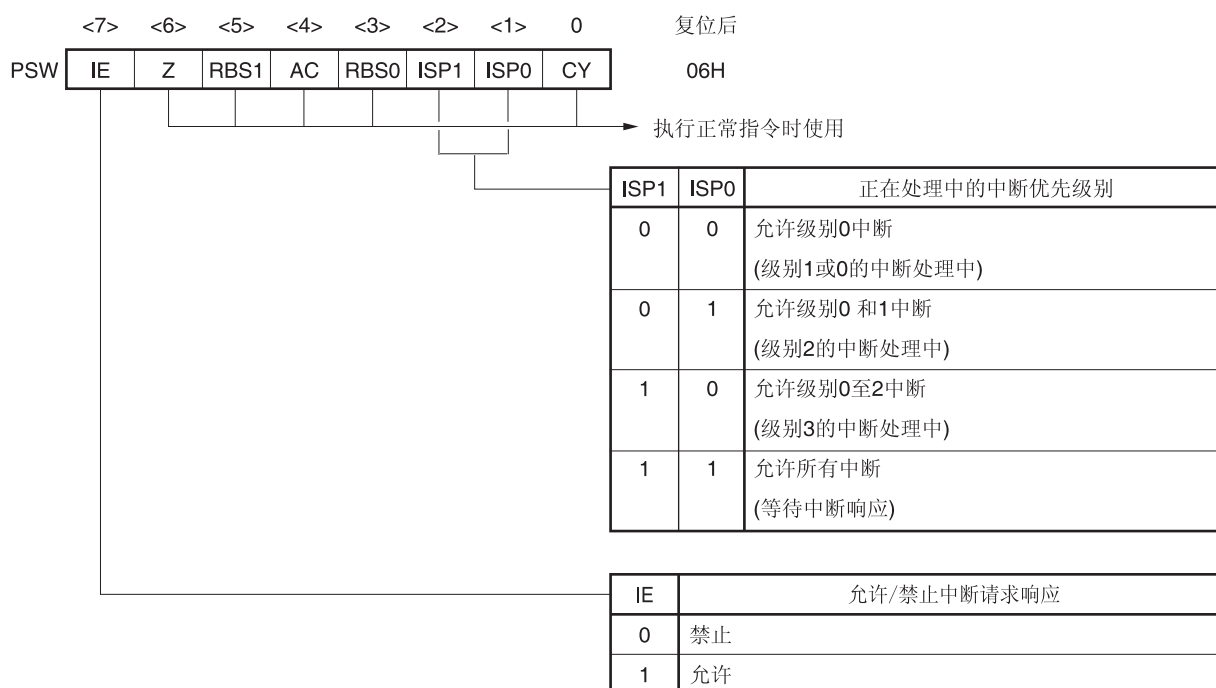
(7) 程序状态字(PSW)

程序状态字为用于保存指令执行结果和中断请求当前状态的寄存器。设置允许/禁止可屏蔽中断的 IE 标志以及控制嵌套中断处理的 ISP0 和 ISP1 标志被映射至 PSW。

除 8 位读/写以外，该寄存器还可以利用位操作指令和专用指令（EI 和 DI）来执行操作。当向量中断请求被受理时，如果执行 BRK 指令，则 PSW 的内容将被自动保存至堆栈，且 IE 标志复位至 0。如果可屏蔽中断请求被受理，则被受理中断的优先级指定标志的内容将被传送至 ISP0 和 ISP1 标志。PSW 的内容也可通过 PUSH PSW 指令保存至堆栈。这些内容可用 RETI、RETB 和 POP PSW 指令从堆栈中恢复。

产生复位信号后，PSW 被设置为 06H。

图 20-9. 程序状态字的配置



20.4 中断处理的操作

20.4.1 可屏蔽中断请求受理

当中断请求标志被设置为 1 且与该中断请求对应的屏蔽(MK)标志被清除为 0 时, 该可屏蔽中断可被受理。在中断允许状态(当 IE 标志被设置为 1)时, 向量中断请求被受理。然而, 在处理较高优先级中断请求期间, 不受理低优先级中断请求。

从产生可屏蔽中断请求到执行向量中断处理的时间如下面的表 20-4 所示。

关于中断请求受理时序, 请参阅图 20-11 和 20-12。

表 20-4. 从产生可屏蔽中断到处理的时间

	最短时间	最长时间 ^注
处理时间	9 个时钟	16 个时钟

注 执行内部 RAM 区的指令时, 不适用以上的最长时间。

备注 1 时钟: $1/f_{CLK}$ (f_{CLK} : CPU 时钟)

如果同时产生两个或更多可屏蔽中断请求, 则优先级指定标志所指定的优先级较高的请求将被首先受理。如果两个或更多中断请求有着相同的优先级, 则默认优先级最高的请求被首先受理。

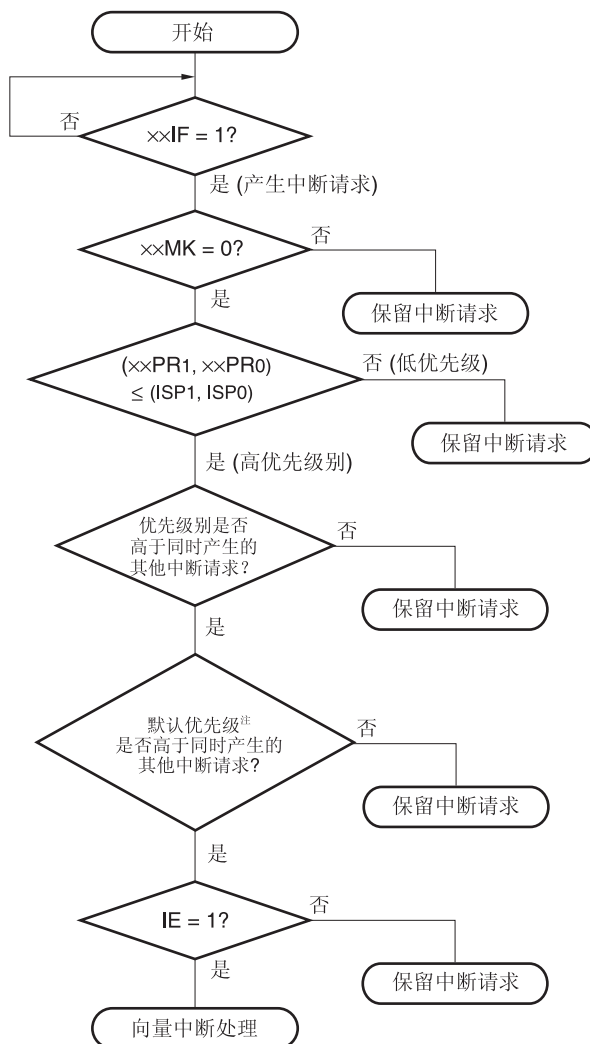
被保留的中断请求将在可以受理时被受理。

中断请求受理的流程如图 20-10 所示。

如果可屏蔽中断请求被受理, 则内容将按先 PSW 后 PC 的顺序保存至堆栈之中, 然后复位 IE 标志为(0), 并将与已受理中断对应的优先级指定标志的内容传送至 ISP1 和 ISP0 标志。为各中断请求确定的向量表数据将载入 PC 并分支。

可使用 RETI 指令从中断恢复。

图 20-10. 中断请求受理的处理流程



××IF: 中断请求标志

××MK: 中断屏蔽标志

××PR0: 优先级指定标志 0

××PR1: 优先级指定标志 1

IE: 用于控制可屏蔽中断请求的受理的标志(1 = 允许, 0 = 禁止)

ISPO, ISP1: 表示正在处理的中断的优先级的标志(参阅图 20-9)

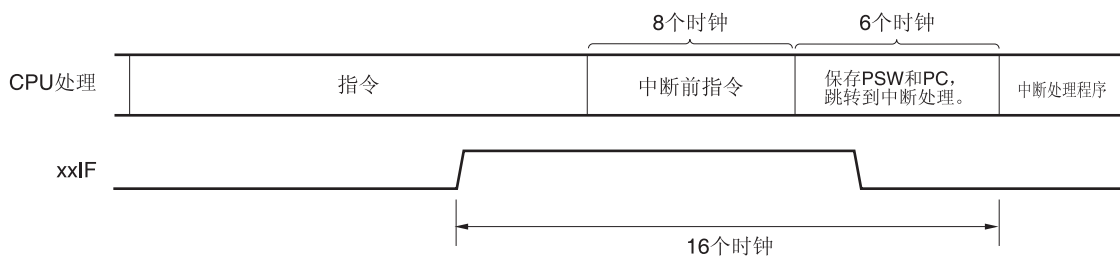
注 关于默认优先级, 请参阅表 20-1. 中断源列表。

图 20-11. 中断请求受理时序 (最短时间)



备注 1 时钟: 1/f_{CLK} (f_{CLK}: CPU 时钟)

图 20-12. 中断请求受理时序 (最长时间)



备注 1 时钟: 1/f_{CLK} (f_{CLK}: CPU 时钟)

20.4.2 软件中断请求受理

通过执行 **BRK** 指令受理软件中断请求。不能禁止软件中断。

如果软件中断请求被受理，则内容将按先程序状态字 (PSW)后程序计数器(PC)的顺序保存至堆栈之中，**IE** 标志被复位为(0)，向量表(0007EH, 0007FH)的内容载入 **PC** 并分支。

可利用 **RETB** 指令从软件中断恢复。

注意事项 不要使用 **RETI** 指令从软件中断恢复。

20.4.3 嵌套中断处理

当在执行一个中断的过程中受理了另一个中断请求时，这种情况叫做嵌套中断。

除非选择中断请求受理允许状态(**IE = 1**)，否则不会发生嵌套中断处理。当某个中断请求被受理时，禁止中断请求受理 (**IE = 0**)。因此，若要允许嵌套中断处理，则须在中断处理期间用 **EI** 指令将 **IE** 标志设置为(1)，以允许中断受理。

另外，即使允许中断，也可能不允许嵌套中断处理，这取决于中断优先级控制。可使用两种优先级控制：默认优先级控制和可编程优先级控制。可编程优先级控制用于嵌套中断处理。

在中断允许状态下，如果产生的中断请求的优先级高于正在处理的中断的优先级，则受理嵌套中断处理。如果在中断处理期间产生的中断的优先级等于或低于正在处理的中断的优先级，则不受理嵌套中断处理。因中断处于中断禁止状态或者优先级较低而不启用的中断请求将被保留。在当前中断的处理结束时，被保留的中断请求将在执行至少一条主处理指令之后被受理。

表 20-5 展示了允许嵌套中断处理的中断请求之间的关系，图 20-13 为嵌套中断处理示例。

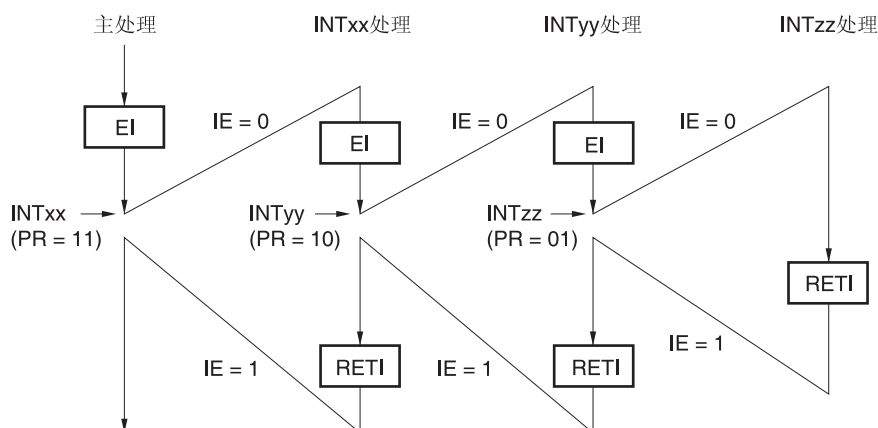
表 20-5. 中断处理期间允许嵌套中断处理的中断请求之间的关系

嵌套中断请求		可屏蔽中断请求								软件中断请求
		优先等级 0 (PR = 00)		优先等级 1 (PR = 01)		优先等级 2 (PR = 10)		优先等级 3 (PR = 11)		
		IE = 1	IE = 0	IE = 1	IE = 0	IE = 1	IE = 0	IE = 1	IE = 0	
正在处理的中断										
可屏蔽中断	ISP1 = 0 ISP0 = 0	√	-	-	-	-	-	-	-	√
	ISP1 = 0 ISP0 = 1	√	-	√	-	-	-	-	-	√
	ISP1 = 1 ISP0 = 0	√	-	√	-	√	-	-	-	√
软件中断		√	-	√	-	√	-	√	-	√

- 备注 1.** √: 允许嵌套中断处理
2. -: 禁止嵌套中断处理
3. ISP0、ISP1 和 IE 为包含在 PSW 中的标志。
 ISP1 = 0, ISP0 = 0: 正在处理等级 1 或等级 0 的中断。
 ISP1 = 0, ISP0 = 1: 正在处理等级 2 的中断。
 ISP1 = 1, ISP0 = 0: 正在处理等级 3 的中断。
 ISP1 = 1, ISP0 = 1: 等待中断受理。
 IE = 0: 禁止中断请求受理。
 IE = 1: 允许中断请求受理。
4. PR 为包含在 PR00L、PR00H、PR01L、PR01H、PR02L、PR02H、PR10L、PR10H、PR11L、PR11H、PR12L 和 PR12H 寄存器中的标志。
 PR = 00: 通过 $\times\times PR1\times = 0, \times\times PR0\times = 0$ 指定等级 0(较高优先级)
 PR = 01: 通过 $\times\times PR1\times = 0, \times\times PR0\times = 1$ 指定等级 1
 PR = 10: 通过 $\times\times PR1\times = 1, \times\times PR0\times = 0$ 指定等级 2
 PR = 11: 通过 $\times\times PR1\times = 1, \times\times PR0\times = 1$ 指定等级 3 (较低优先级)

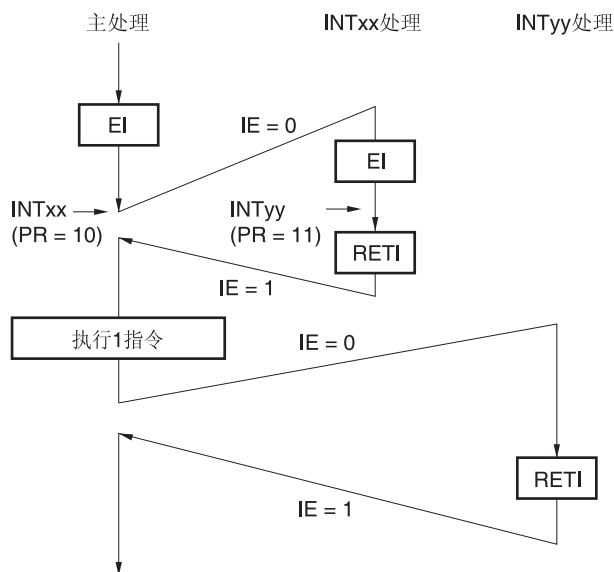
图 20-13. 嵌套中断处理示例 (1/2)

示例 1. 发生两次嵌套中断处理



在处理中断 INTxx 时，受理 INTyy 和 INTzz 这两个中断请求，发生嵌套中断处理。在受理各中断请求之前，必须执行 EI 指令以允许中断请求受理。

示例 2. 由于优先级控制，不发生嵌套中断处理。



在处理中断 INTxx 时产生的中断请求 INTyy 不被受理，因为其优先级低于 INTxx，不发生嵌套中断处理。INTyy 中断请求被保留，并在执行了一条主处理指令之后被受理。

PR = 00: 通过 $\times\times PR1\times = 0, \times\times PR0\times = 0$ 指定等级 0 (较高优先级)

PR = 01: 通过 $\times\times PR1\times = 0, \times\times PR0\times = 1$ 指定等级 1

PR = 10: 通过 $\times\times PR1\times = 1, \times\times PR0\times = 0$ 指定等级 2

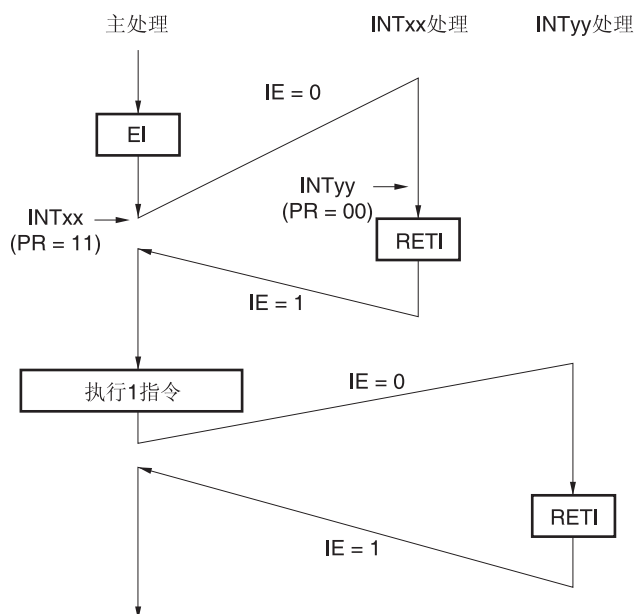
PR = 11: 通过 $\times\times PR1\times = 1, \times\times PR0\times = 1$ 指定等级 3 (较低优先级)

IE = 0: 禁止中断请求受理。

IE = 1: 允许中断请求受理。

图 20-13. 嵌套中断处理示例 (2/2)

示例 3. 因为不允许中断，所以不发生嵌套中断处理



在处理中断 INT_{xx} 时因为不允许中断（未使用 EI 指令），所以中断请求 INT_{yy} 不被受理，不发生嵌套中断处理。 INT_{yy} 中断请求被保留，并在执行了一条主处理指令之后被受理。

$PR = 00$: 通过 $\times\times PR1\times = 0, \times\times PR0\times = 0$ 指定等级 0(较高优先级)

$PR = 01$: 通过 $\times\times PR1\times = 0, \times\times PR0\times = 1$ 指定等级 1

$PR = 10$: 通过 $\times\times PR1\times = 1, \times\times PR0\times = 0$ 指定等级 2

$PR = 11$: 通过 $\times\times PR1\times = 1, \times\times PR0\times = 1$ 指定等级 3(较低优先级)

$IE = 0$: 禁止中断请求受理。

$IE = 1$: 允许中断请求受理。

20.4.4 保留中断请求

有些指令，即使在执行指令期间产生了中断请求，中断请求受理也会被保留，直到下一条指令执行完成为止。这些指令（中断请求保留指令）如下所示。

- MOV PSW, #byte
- MOV PSW, A
- MOV1 PSW. bit, CY
- SET1 PSW. bit
- CLR1 PSW. bit
- RETB
- RETI
- POP PSW
- BTCLR PSW. bit, \$addr20
- EI
- DI
- SKC
- SKNC
- SKZ
- SKNZ
- SKH
- SKNH
- 对 IF0L、IF0H、IF1L、IF1H、IF2L、IF2H、MK0L、MK0H、MK1L、MK1H、MK2L、MK2H、PR00L、PR00H、PR01L、PR01H、PR02L、PR02H、PR10L、PR10H、PR11L、PR11H、PR12L 和 PR12H 的各个寄存器的操作指令

保留中断请求的时序如图 20-14 所示。

图 20-14. 保留中断请求



- 备注**
1. 指令 N: 中断请求保留指令
 2. 指令 M: 中断请求保留指令以外的指令

第二十一章 待机功能

21.1 待机功能和构成

21.1.1 待机功能

待机功能可以降低系统的工作电流，有以下三种模式可供选择。

(1) HALT 模式

通过执行 HALT 指令设置为 HALT 模式。在 HALT 模式下，CPU 工作时钟停止。如果在设置 HALT 模式之前，高速系统时钟振荡器、高速片上振荡器或副系统时钟振荡器正在工作，则各时钟的振荡将继续。在该模式下，工作电流降低幅度不如 STOP 模式，但 HALT 模式在产生中断请求时立即重启工作以及经常执行间歇工作方面非常有效。

(2) STOP 模式

通过执行 STOP 指令设置为 STOP 模式。在 STOP 模式下，高速系统时钟振荡器和高速片上振荡器停止工作，并停止整个系统，从而大幅降低 CPU 工作电流。

由于可以通过中断请求解除该模式，因而可以执行间歇工作。然而，在选择 X1 时钟的情况下，由于解除 STOP 模式之后，需要一定的等待时间以确保振荡稳定时间，因此，如果需要在产生中断请求时立即开始处理，请选择 HALT 模式。

(3) SNOOZE 模式

当存在 CSI00、UART0 或 DALI/UART4 数据接收以及定时器触发信号（中断请求信号(INTRTC/INTIT)）发出的 A/D 转换请求时，退出 STOP 模式，接收 CSI00、UART0 或 DALI/UART4 数据而不工作 CPU，并执行 A/D 转换。仅当选择高速片上振荡器作为 CPU/外围硬件时钟(fCLK)时可以指定该模式。

在以上任何一种模式下，设置待机模式之前寄存器、标志和数据存储器的所有内容将被保持。同时还保持 I/O 端口输出锁存器和输出缓冲器状态。

- 注意事项**
1. 仅当 CPU 工作于主系统时钟时，才可使用 STOP 模式。当 CPU 以副系统时钟工作时，不能设置 STOP 模式。当 CPU 工作于主系统时钟或副系统时钟时，均可使用 HALT 模式。
 2. 切换至 STOP 模式时，必须在执行 STOP 指令（除 SNOOZE 模式设置单元外）之前，停止按主系统时钟工作的外围硬件。
 3. 在 SNOOZE 模式下使用 CSI00、UART0、DALI/UART4 或 A/D 转换器时，须在切换至 STOP 模式之前，设置串行待机控制寄存器 0 (SSC0)、串行待机控制寄存器 4 (SSC4) 和 A/D 转换器模式寄存器 2 (ADM2)。详情请参阅 15.3 控制串行阵列单元 0 的寄存器、16.3 控制串行阵列单元 4 (DALI/UART4) 的寄存器和 12.3 A/D 转换器中使用的寄存器。

4. 当 DALI/UART4 被设置为 SNOOZE 模式时，CSI00、UART0 和 A/D 转换器不能被设置为 SNOOZE 模式。可以同时将 CSI00、UART0 和 A/D 转换器设置为 SNOOZE 模式。
5. 使用待机功能时，建议通过以下步骤降低 A/D 转换器的功耗：先将 A/D 转换器模式寄存器 0 (ADM0) 的位 7 (ADCS)和位 0 (ADCE)清除为 0，以停止 A/D 转换工作，然后执行 STOP 指令。
6. 可通过选项字节选择是让低速片上振荡器继续振荡，还是停止于 HALT 或 STOP 模式。详情请参阅第二十七章 选项字节。

21.1.2 控制待机功能的寄存器

解除 STOP 模式时，使用以下两个寄存器控制振荡稳定时间。

- 振荡稳定时间计数器状态寄存器(OSTC)
- 振荡稳定时间选择寄存器(OSTS)

备注 关于启动、停止或选择时钟的寄存器，请参阅第五章 时钟发生电路。

(1) 振荡稳定时间计数器状态寄存器(OSTC)

该寄存器表示 X1 时钟振荡稳定时间计数器的计数状态。

在以下情况下，可以确认 X1 时钟振荡稳定时间。

- 高速片上振荡器时钟或副系统时钟用作 CPU 时钟时，X1 时钟开始振荡。
- 高速片上振荡器时钟用作 CPU 时钟且 X1 时钟振荡时，进入 STOP 模式然后解除该模式。

使用 1 位或 8 位存储器工作指令来读取 OSTC 寄存器。

在解除复位后（由 RESET 输入、POR、LVD、WDT，以及执行非法指令引起的复位），STOP 指令和 MSTOP 位（时钟工作状态控制寄存器(CSC)的位 7）= 1 将该寄存器清除为 00H。

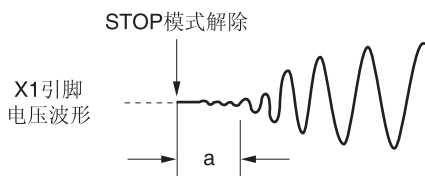
图 21-1. 振荡稳定时间计数器状态寄存器(OSTC) 的格式

地址: FFFA2H 复位后: 00H R

符号	7	6	5	4	3	2	1	0
OSTC	MOST 8	MOST 9	MOST 10	MOST 11	MOST 13	MOST 15	MOST 17	MOST 18

MOST 8	MOST 9	MOST 10	MOST 11	MOST 13	MOST 15	MOST 17	MOST 18	振荡稳定时间状态		
								fx = 10 MHz	fx = 20 MHz	
0	0	0	0	0	0	0	0	2 ⁸ /fx max.	25.6 μs max.	12.8 μs max.
1	0	0	0	0	0	0	0	2 ⁸ /fx min.	25.6 μs min.	12.8 μs min.
1	1	0	0	0	0	0	0	2 ⁹ /fx min.	51.2 μs min.	25.6 μs min.
1	1	1	0	0	0	0	0	2 ¹⁰ /fx min.	102.4 μs min.	51.2 μs min.
1	1	1	1	0	0	0	0	2 ¹¹ /fx min.	204.8 μs min.	102.4 μs min.
1	1	1	1	1	0	0	0	2 ¹³ /fx min.	819.2 μs min.	409.6 μs min.
1	1	1	1	1	1	0	0	2 ¹⁵ /fx min.	3.27 ms min.	1.64 ms min.
1	1	1	1	1	1	1	0	2 ¹⁷ /fx min.	13.11 ms min.	6.55 ms min.
1	1	1	1	1	1	1	1	2 ¹⁸ /fx min.	26.21 ms min.	13.11 ms min.

- 注意事项**
1. 经过上述时间后，从 MOST8 位开始各个位依次设置为 1 并保持 1 不变。
 2. 振荡稳定时间计数器仅在振荡稳定时间选择寄存器(OSTS)所设置的振荡稳定时间内计数。如果在高速片上振荡器时钟用作 CPU 时钟时切换至 STOP 模式然后解除该模式，则按以下方式设置振荡稳定时间。
 - 需要的 OSTC 寄存器振荡稳定时间 ≤ 由 OSTS 寄存器设置的振荡稳定时间
 注意，解除 STOP 模式后，只有 OSTS 寄存器所设置的振荡稳定时间内的状态被设置至 OSTC 寄存器。
 3. X1 时钟振荡稳定等待时间不包括开始时钟振荡前的时间（下图“a”）。



备注 fx: X1 时钟振荡频率

(2) 振荡稳定时间选择寄存器(OSTS)

该寄存器用于选择解除 STOP 模式时的 X1 时钟振荡稳定等待时间。

当 X1 时钟被用作 CPU 时钟时，在解除 STOP 模式后，系统等待由 OSTS 寄存器设置的时间。

当选用高速片上振荡器时钟作为 CPU 时钟时，解除 STOP 模式后，请通过振荡稳定时间计数器状态寄存器(OSTC)确认所需振荡稳定时间已过。在 OSTS 寄存器事先设置的时间范围内都可以通过 OSTC 寄存器进行确认。

使用 8 位存储器工作指令来设置 OSTS 寄存器。

产生复位信号后，该寄存器被设置为 07H。

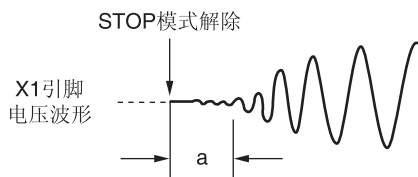
图 21-2. 振荡稳定时间选择寄存器(OSTS) 的格式

地址: FFFA3H 复位后: 07H R/W

符号	7	6	5	4	3	2	1	0
OSTS	0	0	0	0	0	OSTS2	OSTS1	OSTS0

OSTS2	OSTS1	OSTS0		选择振荡稳定时间	
				fx = 10 MHz	fx = 20 MHz
0	0	0	$2^8/f_x$	25.6 μ s	12.8 μ s
0	0	1	$2^9/f_x$	51.2 μ s	25.6 μ s
0	1	0	$2^{10}/f_x$	102.4 μ s	51.2 μ s
0	1	1	$2^{11}/f_x$	204.8 μ s	102.4 μ s
1	0	0	$2^{13}/f_x$	819.2 μ s	409.6 μ s
1	0	1	$2^{15}/f_x$	3.27 ms	1.64 ms
1	1	0	$2^{17}/f_x$	13.11 ms	6.55 ms
1	1	1	$2^{18}/f_x$	26.21 ms	13.11 ms

- 注意事项
1. 用 X1 时钟作为 CPU 时钟并进入 STOP 模式时，执行 STOP 指令之前须设置好 OSTS 寄存器。
 2. 在更改 OSTS 寄存器的设置之前，须确认 OSTC 寄存器的计数工作已完成。
 3. X1 时钟振荡稳定时间期间不要更改 OSTS 寄存器的值。
 4. 振荡稳定时间计数器仅在 OSTS 寄存器设置的振荡稳定时间内计数。如果在高速片上振荡器时钟用作 CPU 时钟时切换至 STOP 模式然后解除该模式，则按以下方式设置振荡稳定时间。
 - 需要的 OSTC 寄存器振荡稳定时间 \leq 由 OSTS 寄存器设置的振荡稳定时间
 注意，解除 STOP 模式后，只有 OSTS 寄存器所设置的振荡稳定时间内的状态被设置至 OSTC 寄存器。
 5. X1 时钟振荡稳定等待时间不包括开始时钟振荡前的时间（下图“a”）。



备注 fx: X1 时钟振荡频率

21.2 待机功能工作

21.2.1 HALT 模式

(1) HALT 模式

通过执行 HALT 指令设置 HALT 模式。无论设置之前的 CPU 时钟为高速系统时钟、高速片上振荡器时钟还是副系统时钟，均可设置 HALT 模式。

HALT 模式时的工作状态如下所示。

表 21-1. HALT 模式时的工作状态 (1/2)

HALT 模式设置		CPU 利用主系统时钟工作期间, 执行 HALT 指令时					
		CPU 工作于高速片上振荡器时钟 (f _H)时	CPU 工作于 X1 时钟(f _X)时	CPU 工作于外部主系统时钟 (f _{EX})时			
项目							
系统时钟		停止向 CPU 供应时钟。					
主系统时钟	f _H	继续工作(不可以停止)	禁止工作				
	f _X	禁止工作	继续工作(不可以停止)	不可工作			
	f _{EX}		不可工作	继续工作(不可以停止)			
副系统时钟	f _{XT}	保持设置为 HALT 模式前的状态					
	f _{EXT}						
f _{IL}		利用选项字节(000C0H)的位 0(WDSTBYON)和 4(WDTON), 以及工作速度模式控制寄存器(OSMC)的 WUTMMCK0 位进行设置 <ul style="list-style-type: none"> • WUTMMCK0 = 1: 振荡 • WUTMMCK0 = 0 和 WDTON = 0: 停止 • WUTMMCK0 = 0, WDTON = 1 和 WDSTBYON = 1: 振荡 • WUTMMCK0 = 0, WDTON = 1 和 WDSTBYON = 0: 停止 					
CPU		停止工作					
代码闪存		停止工作					
数据闪存							
RAM							
端口(锁存器)		保持设置为 HALT 模式前的状态					
定时器阵列单元		可以工作					
定时器 KB0 至 KB2							
定时器 KC0							
实时时钟 (RTC)							
12 位间隔定时器							
看门狗定时器		参阅第十一章 看门狗定时器					
A/D 转换器		可以工作					
可编程增益放大器							
比较器							
串行阵列单元 0 (SAU0)							
串行阵列单元 4 (DALI/UART4)							
串行接口(IICA)							
乘除法和乘加器							
DMA 控制器							
上电复位功能							
电压检测功能							
外部中断							
CRC 工作功能	高速 CRC				在 RAM 区域的计算中, 仅限执行 DMA 时可工作。		
	通用 CRC						
RAM 奇偶校验错误检测功能					仅限执行 DMA 时可工作。		
RAM 保护功能							
SFR 保护功能							
非法存储器存取检测功能							
PLL 功能		可以工作					

(备注如下页所示。)

备注 停止工作： 在切换到 **HALT** 模式时自动停止工作。
禁止工作： 在切换到 **HALT** 模式前使工作停止。

f_H: 高速片上振荡器时钟
f_L: 低速片上振荡器时钟
f_X: **X1** 时钟
f_{EX}: 外部主系统时钟
f_{XT}: **XT1** 时钟
f_{EXT}: 外部副系统时钟

表 21-1. HALT 模式时的工作状态 (2/2)

HALT 模式设置		当 CPU 工作于副系统时钟, 执行 HALT 指令时	
项目		CPU 工作于 XT1 时钟(f _{XT})时	CPU 工作于外部副系统时钟(f _{EXT})时
系统时钟		停止向 CPU 供应时钟。	
主系统时钟	f _{IH}	禁止工作	
	f _X		
	f _{EX}		
副系统时钟	f _{XT}	继续工作(不可以停止)	不可以工作
	f _{EXT}	不可以工作	继续工作(不可以停止)
f _{IL}		利用选项字节(000C0H)的位 0(WDSTBYON)和 4(WDTON), 以及工作速度模式控制寄存器(OSMC)的 WUTMMCK0 位进行设置 <ul style="list-style-type: none"> • WUTMMCK0 = 1: 振荡 • WUTMMCK0 = 0 和 WDTON = 0: 停止 • WUTMMCK0 = 0, WDTON = 1 和 WDSTBYON = 1: 振荡 • WUTMMCK0 = 0, WDTON = 1 和 WDSTBYON = 0: 停止 	
CPU		停止工作	
代码闪存		保持设置为 HALT 模式前的状态	
数据闪存			
RAM			
端口(锁存器)		保持设置为 HALT 模式前的状态	
定时器阵列单元		可以工作(低功耗 RTC 模式(当 OSMC 寄存器的 RTCLPC 位为 1)时禁止工作)	
定时器 KB0 至 KB2			
定时器 KC0			
实时时钟 (RTC)		可以工作	
12 位间隔定时器		参阅第十一章 看门狗定时器	
看门狗定时器			
A/D 转换器		禁止工作	
可编程增益放大器		可以工作(然而, 由于禁止工作 A/D 转换器(PGA 输出信号的输入目的地), 不能使用)	
比较器		可以工作(在低功耗 RTC 模式(OSMC 寄存器的 RTCLPC = 1)时, 仅当由于检测到比较器中断而解除 STOP 模式(PFSEL0 寄存器的 CMPnSTEN = 1)并且不使用噪声滤波器(n = 0, 2)时可以使用)	
串行阵列单元 0 (SAU0)		可以工作(低功耗 RTC 模式(当 OSMC 寄存器的 RTCLPC 位为 1)时禁止工作。)	
串行阵列单元 4 (DALI/UART4)		禁止工作	
串行接口(IICA)		可以工作(低功耗 RTC 模式(当 OSMC 寄存器的 RTCLPC 位为 1)时禁止工作)	
乘除法和乘加器			
DMA 控制器			
上电复位功能		可以工作	
电压检测功能		可以工作(在定时器 KB0 至 KB2 的 INTP 同步模式时, 禁止工作 INTP20 和 INTP21。)	
外部中断			
CRC 工作功能	高速 CRC	禁止工作	
	通用 CRC	在 RAM 区域的计算中, 仅限执行 DMA 时可工作。	
RAM 奇偶校验错误检测功能		仅限执行 DMA 时可工作。	
RAM 保护功能			
SFR 保护功能			
非法存储器存取检测功能			
PLL 功能			

(备注如下页所示。)

备注

停止工作： 在切换到 HALT 模式时自动停止工作。

禁止工作： 在切换到 HALT 模式前使工作停止。

f_H: 高速片上振荡器时钟

f_L: 低速片上振荡器时钟

f_X: X1 时钟

f_{EX}: 外部主系统时钟

f_{XT}: XT1 时钟

f_{EXT}: 外部副系统时钟

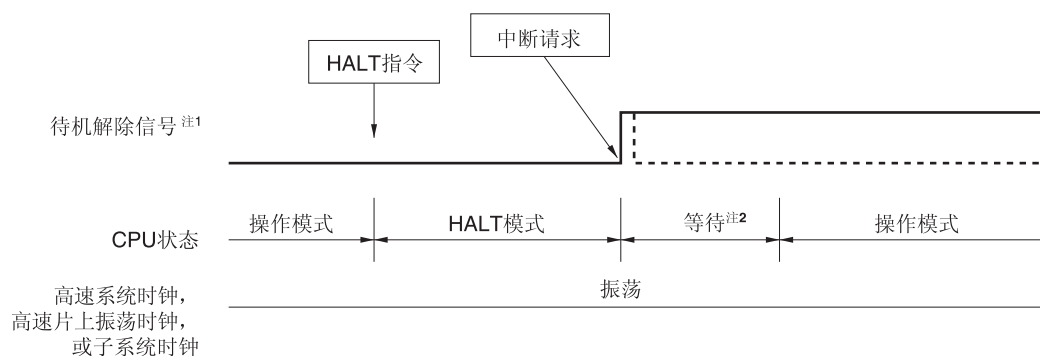
(2) HALT 模式的解除

可使用以下两种方法解除 HALT 模式。

(a) 通过未屏蔽中断请求解除

当产生一个未屏蔽中断请求时，HALT 模式被解除。如果允许受理中断，则执行向量中断处理。如果不允许受理中断，则执行下一个地址指令。

图 21-3. 通过产生中断请求解除 HALT 模式



注 1. 关于待机解除信号的详情，请参阅图 20-1。

2. 解除 HALT 模式的等待时间

- 执行向量中断处理时
 - 主系统时钟: 15 至 16 个时钟
 - 副系统时钟 (RTCLPC = 0): 10 至 11 个时钟
 - 副系统时钟 (RTCLPC = 1): 11 至 12 个时钟
- 不执行向量中断处理时:
 - 主系统时钟: 9 至 10 个时钟
 - 副系统时钟 (RTCLPC = 0): 4 至 5 个时钟
 - 副系统时钟 (RTCLPC = 1): 5 至 6 个时钟

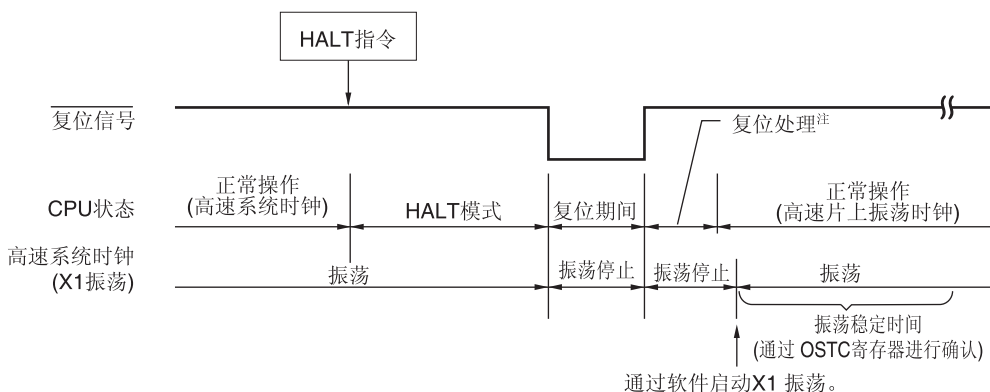
备注 虚线表示导致待机模式解除的中断请求被受理时的情况。

(b) 通过产生复位信号解除

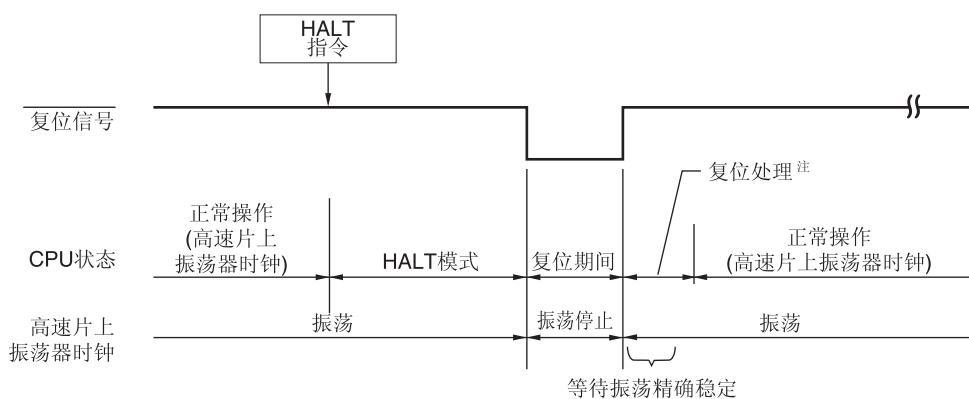
当产生复位信号时，HALT 模式被解除，然后，和正常复位操作一样，跳转至复位向量地址之后执行程序。

图 21-4. 通过复位解除 HALT 模式

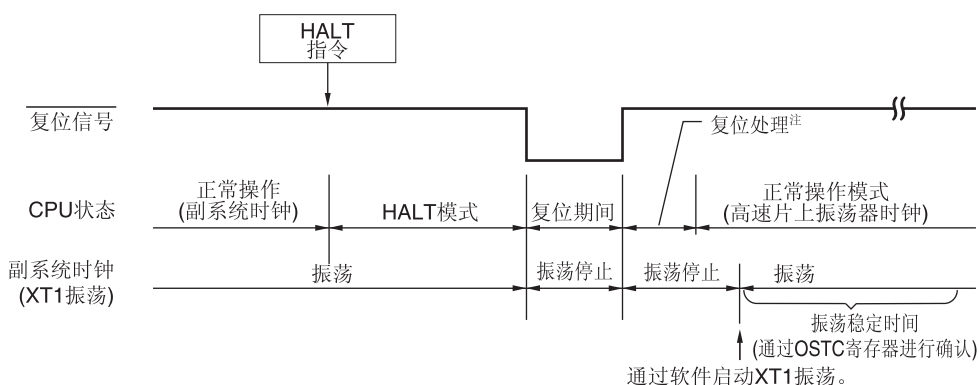
(1) CPU 时钟使用高速系统时钟时



(2) CPU 时钟使用高速片上振荡器时钟时



(3) CPU 时钟使用副系统时钟时



注 复位处理时间： 388 至 673 μ s (LVD 启用时)
156 至 360 μ s (LVD 关闭时)

21.2.2 STOP模式

(1) STOP模式的设置和工作状态

通过执行 STOP 指令设置 STOP 模式，仅限在设置之前 CPU 时钟为高速片上振荡器时钟、X1 时钟或外部主系统时钟时，可以设置该模式。

- 注意事项**
1. 由于中断请求信号用于解除 STOP 模式，因此，如果有一个中断请求标志已设置且中断屏蔽标志已复位的中断源，则 STOP 模式将立即被解除（若已设置）。因此，STOP 模式将在执行 STOP 指令之后立即复位至 HALT 模式，系统在经过由振荡稳定时间选择寄存器(OSTS) 设置的等待时间之后返回工作模式。
 2. 在 SNOOZE 模式下使用 CSI00、UART0、DALI/UART4 或 A/D 转换器时，须在切换至 STOP 模式之前，设置串行待机控制寄存器 0 (SSC0)、串行待机控制寄存器 4 (SSC4) 和 A/D 转换器模式寄存器 2 (ADM2)。详情请参阅 15.3 控制串行阵列单元 0 的寄存器、16.3 控制串行阵列单元 4 (DALI/UART4)的寄存器和 12.3 A/D 转换器中使用的寄存器。
 3. 当主系统时钟选择 PLL 输出时，不能设置 STOP 模式。请停止 PLL 功能(PLL 控制寄存器(PLLCTL)的 SELPLL = 0 → PLLON = 0)后，选择高速片上振荡时钟(f_{IH})或高速系统时钟(f_{MX})作为主系统时钟，再执行 STOP 指令。

STOP 模式时的工作状态如下所示。

表 21-2. STOP 模式时的工作状态

STOP 模式设置		CPU 利用主系统时钟工作期间，执行 STOP 指令时		
		CPU 工作于高速片上振荡器时钟(f_{IH})时	CPU 工作于 X1 时钟(f_x)时	CPU 工作于外部主系统时钟(f_{EX})时
项目				
系统时钟		停止向 CPU 供应时钟。		
主系统时钟	f_{IH}	停止		
	f_x			
	f_{EX}			
副系统时钟	f_{XT}	保持设置为 STOP 模式前的状态		
	f_{EXT}			
f_{IL}		利用选项字节(000C0H)的位 0(WDSTBYON)和 4(WDTON)，以及工作速度模式控制寄存器(OSMC)的 WUTMMCK0 位进行设置		
		<ul style="list-style-type: none"> • WUTMMCK0 = 1: 振荡 • WUTMMCK0 = 0 和 WDTON = 0: 停止 • WUTMMCK0 = 0, WDTON = 1 和 WDSTBYON = 1: 振荡 • WUTMMCK0 = 0, WDTON = 1 和 WDSTBYON = 0: 停止 		
CPU		停止工作		
代码闪存		禁止工作		
数据闪存				
RAM				
端口(锁存器)		保持设置为 STOP 模式前的状态		
定时器阵列单元		禁止工作		
定时器 KB0 至 KB2				
定时器 KC0				
实时时钟 (RTC)		可以工作		
12 位间隔定时器		参阅第十一章 看门狗定时器		
看门狗定时器				
A/D 转换器				
可编程增益放大器		可以工作		
比较器		可以工作(仅限设置为可以解除 STOP 模式的通道(不使用数字滤波器))		
串行阵列单元 0 (SAU0)		仅对 CSI00 和 UART0 允许唤醒工作(切换至 SNOOZE 模式) 禁止除了 CSI00 和 UART0 以外的所有工作		
串行阵列单元 4 (DALI/UART4)		允许唤醒工作(切换至 SNOOZE 模式)。		
串行接口(IICA)		可以工作使用地址匹配的唤醒		
乘除法和乘加器		禁止工作		
DMA 控制器				
上电复位功能		可以工作		
电压检测功能		可以工作(在定时器 KB0 至 KB2 的 INTP 同步模式时，禁止工作 INTP20 和 INTP21。)		
外部中断				
CRC 工作功能	高速 CRC	禁止工作		
	通用 CRC			
RAM 奇偶校验错误检测功能				
RAM 保护功能				
SFR 保护功能				
非法存储器存取检测功能				
PLL 功能				

(备注和注意事项如下页所示。)

备注	停止工作:	在切换到 STOP 模式时自动停止工作。
	禁止工作:	在切换到 STOP 模式前使工作停止。
	f _H :	高速片上振荡器时钟
	f _L :	低速片上振荡器时钟
	f _X :	X1 时钟
	f _{EX} :	外部主系统时钟
	f _{XT} :	XT1 时钟
	f _{EXT} :	外部副系统时钟

- 注意事项
1. 解除 STOP 模式后,若要使用在 STOP 模式时停止工作的外围硬件或选用了停止振荡的时钟的外围硬件,须重启外围硬件。
 2. 若要使低速片上振荡器时钟停止于 STOP 模式,必须预先设置选项字节使看门狗定时器工作停止于 HALT/STOP 模式(000C0H 的位 0 (WDSTBYON) = 0)。
 3. 当 CPU 工作于高速系统时钟 (X1 振荡) 时,要在解除 STOP 模式之后缩短振荡稳定时间,须在执行 STOP 指令之前临时将 CPU 时钟切换至高速片上振荡器时钟。在解除 STOP 模式之后,在将 CPU 时钟从高速片上振荡器时钟切换为高速系统时钟 (X1 振荡) 之前,须用振荡稳定时间计数器状态寄存器(OSTC) 确认振荡稳定时间。

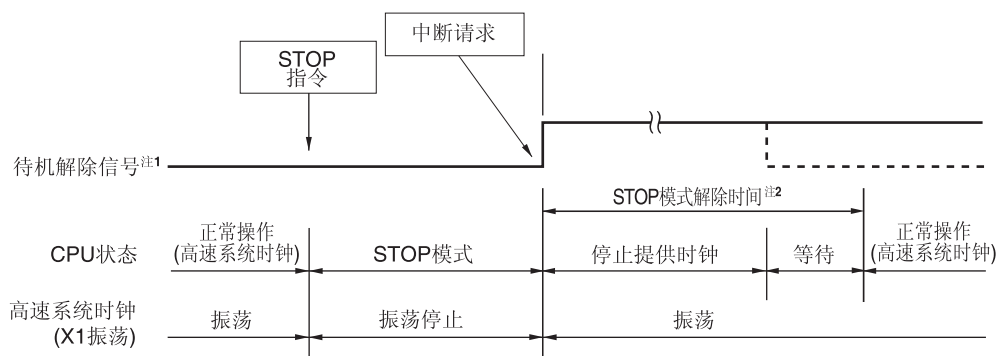
(2) STOP 模式的解除

可使用以下两种源解除 STOP 模式。

(a) 通过未屏蔽中断请求解除

当产生一个未屏蔽中断请求时，STOP 模式被解除。经过振荡稳定时间之后，如果允许受理中断，则执行向量中断处理。如果不允许受理中断，则执行下一个地址指令。

图 21-5. 通过中断请求解除 STOP 模式 (1/2)

(1) CPU 时钟使用高速系统时钟(X1 振荡)时

注 1. 关于待机解除信号的详情，请参阅图 20-1。

2. STOP 模式解除时间

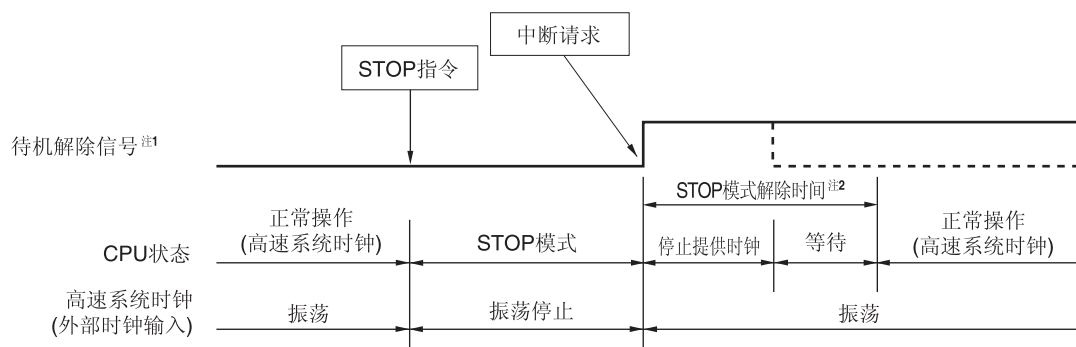
停止供应时钟：18.96 μs 至“28.95 μs 与振荡稳定时间(由 OSTC 设置)的两者中较长的一方”等待

- 执行向量中断处理时：10 至 11 个时钟
- 不执行向量中断处理时：4 至 5 个时钟

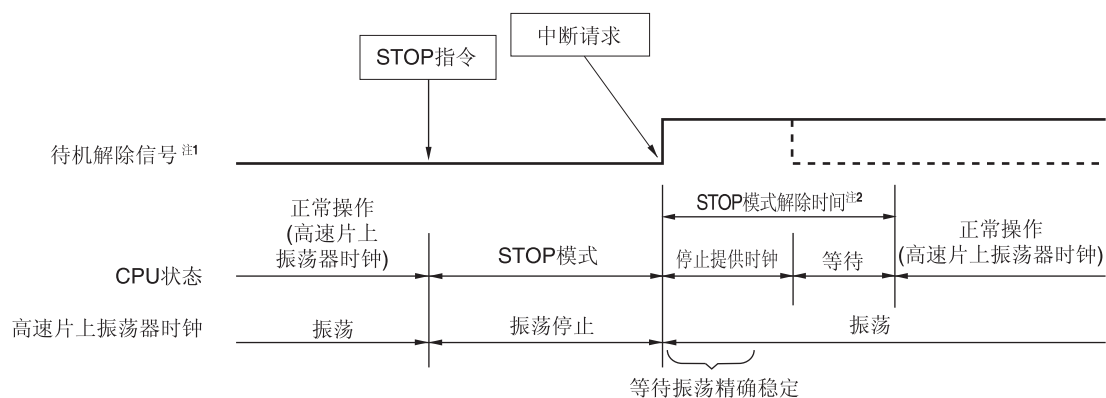
备注 虚线表示导致待机模式解除的中断请求被受理时的情况。

图 21-5. 通过中断请求解除 STOP 模式 (2/2)

(2) CPU 时钟使用高速系统时钟(外部时钟输入)时



(3) CPU 时钟使用高速片上振荡器时钟时



注 1. 关于待机解除信号的详情，请参阅图 20-1。

2. STOP 模式解除时间

停止供应时钟：19.08 至 32.99 μs

等待

- 执行向量中断处理时：7 个时钟
- 不执行向量中断处理时：1 个时钟

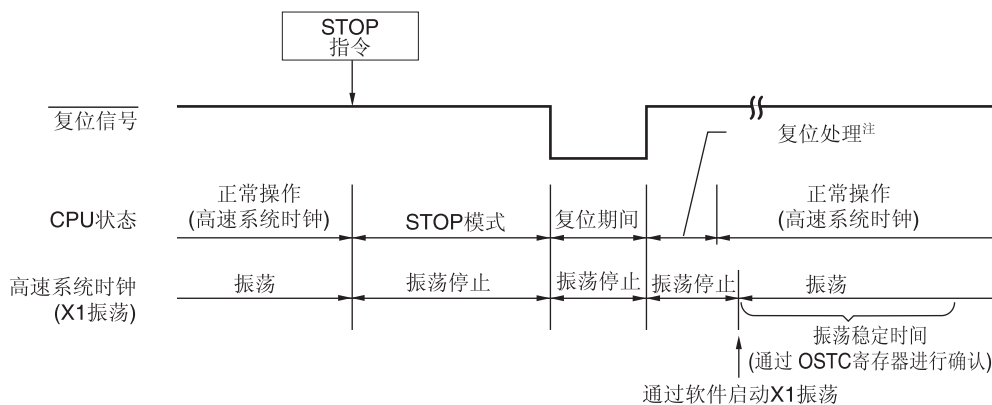
备注 虚线表示导致待机模式解除的中断请求被受理时的情况。

(b) 通过产生复位信号解除

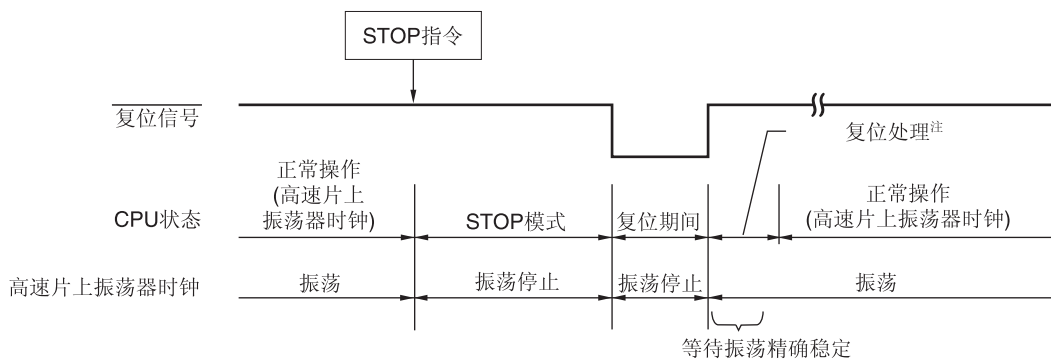
当产生复位信号时，STOP 模式被解除，然后，和正常复位操作一样，跳转至复位向量地址之后执行程序。

图 21-6. 通过复位解除 STOP 模式

(1) CPU 时钟使用高速系统时钟时



(2) CPU 时钟使用高速片上振荡器时钟时



注 复位处理时间: 388 至 673 μ s (LVD 启用时)
156 至 360 μ s (LVD 关闭时)

21.2.3 SNOOZE模式

(1) SNOOZE 模式设置和工作状态

只能为 CSI00、UART0、DALI/UART4 或 A/D 转换器指定 SNOOZE 模式。而且，仅当 CPU 时钟为高速片上振荡器时钟时，才可指定该模式。注意，在上述情况下不能使用 PLL 输出。

在 SNOOZE 模式下使用 CSI00、UART0 或 DALI/UART4 时，须在切换至 STOP 模式之前，设置串行待机控制寄存器 0 (SSC0)或串行待机控制寄存器 4 (SSC4)。详情请参阅 15.3 控制串行阵列单元 0 的寄存器和 16.3 控制串行阵列单元 4 (DALI/UART4)的寄存器。

在 SNOOZE 模式下使用 A/D 转换器时，须在切换至 STOP 模式之前，设置 A/D 转换器模式寄存器 2 (ADM2)。详情请参阅 12.3 A/D 转换器中使用的寄存器。

注意事项 当 DALI/UART4 被设置为 SNOOZE 模式时，CSI00、UART0 和 A/D 转换器不能被设置为 SNOOZE 模式。可以同时将 CSI00、UART0 和 A/D 转换器设置为 SNOOZE 模式。

SNOOZE 模式切换时，仅在以下时间内成为等待状态。

从 STOP 至 SNOOZE

HS (高速主) 模式:	18.96 至 28.95 μ s
LS (低速主) 模式:	20.24 至 28.95 μ s

从 SNOOZE 至正常工作

- 执行向量中断处理时:

HS (高速主) 模式:	6.79 至 12.4 μ s + 7 个时钟
LS (低速主) 模式:	2.58 至 7.8 μ s + 7 个时钟
- 不执行向量中断处理时:

HS (高速主) 模式:	6.79 至 12.4 μ s + 1 个时钟
LS (低速主) 模式:	2.58 至 7.8 μ s + 1 个时钟

SNOOZE 模式时的工作状态如下所示。

表 21-3. SNOOZE 模式时的工作状态

STOP 模式设置		当在 STOP 模式下输入 CSI00/UART0 数据接收信号或 A/D 转换器定时器触发信号时	
项目		CPU 工作于高速片上振荡器时钟(f_{IH})时	
系统时钟		停止向 CPU 供应时钟。	
主系统时钟	f_{IH}	开始工作	
	f_X	停止	
	f_{EX}		
副系统时钟	f_{XT}	继续保持 STOP 模式中的状态	
	f_{EXT}		
f_{IL}	利用选项字节(000C0H)的位 0(WDSTBYON)和 4(WDTON), 以及工作速度模式控制寄存器(OSMC)的 WUTMMCK0 位进行设置		
<ul style="list-style-type: none"> • WUTMMCK0 = 1: 振荡 • WUTMMCK0 = 0 和 WDTON = 0: 停止 • WUTMMCK0 = 0, WDTON = 1 和 WDSTBYON = 1: 振荡 • WUTMMCK0 = 0, WDTON = 1 和 WDSTBYON = 0: 停止 			
CPU		停止工作	
代码闪存			
数据闪存			
RAM			
端口(锁存器)		继续保持 STOP 模式中的状态	
定时器阵列单元		禁止工作	
定时器 KB0 至 KB2			
定时器 KC0			
实时时钟 (RTC)		可以工作	
12 位间隔定时器			
看门狗定时器		参阅第十一章 看门狗定时器	
A/D 转换器		可以工作	
可编程增益放大器		可以工作	
比较器		可以工作(仅限设置为可以解除 STOP 模式的通道(不使用数字滤波器))	
串行阵列单元 0 (SAU0)		仅可工作 CSI00 和 UART0。 禁止除了 CSI00 和 UART0 以外的所有工作。	
串行阵列单元 4 (DALI/UART4)		可以工作	
串行接口(IICA)		禁止工作	
乘除法和乘加器			
DMA 控制器			
上电复位功能		可以工作	
电压检测功能			
外部中断		可以工作(在定时器 KB0 至 KB2 的 INTP 同步模式时, 禁止工作 INTP20 和 INTP21。)	
CRC 工作功能		停止工作	
RAM 奇偶校验错误检测功能			
RAM 保护功能			
SFR 保护功能			
非法存储器存取检测功能			
PLL 功能		禁止工作	

(备注如下页所示。)

备注	停止工作:	在切换到 STOP 模式时自动停止工作。
	禁止工作:	在切换到 STOP 模式前使工作停止。
	f _H :	高速片上振荡器时钟
	f _L :	低速片上振荡器时钟
	f _X :	X1 时钟
	f _{EX} :	外部主系统时钟
	f _{XT} :	XT1 时钟
	f _{EXT} :	外部副系统时钟

第二十二章 复位功能

以下七种操作方法能够产生复位信号。

- (1) 经由 $\overline{\text{RESET}}$ 引脚输入外部复位
- (2) 通过看门狗定时器的程序失控检测的内部复位
- (3) 通过比较上电复位(POR)电路的电源电压与检测电压的内部复位
- (4) 通过比较电压检测电路(LVD) 的电源电压和检测电压的内部复位
- (5) 通过执行非法指令的内部复位^注
- (6) 通过 RAM 奇偶校验错误的内部复位
- (7) 通过存取非法存储器的内部复位

产生复位信号后，内部复位和外部复位都以地址 0000H 和 0001H 处的内容，作为程序的起始地址，开始执行程序。

当向 $\overline{\text{RESET}}$ 引脚输入低电平时、当看门狗定时器溢出时、检测 POR 或 LVD 电路电压时、执行非法指令^注，发生 RAM 奇偶检验错误或非法存储器存取时，复位生效，硬件的各个项目被设置为如表 22-1 所示状态。

向 $\overline{\text{RESET}}$ 引脚输入低电平时，器件复位。当向 $\overline{\text{RESET}}$ 引脚输入高电平时，复位状态解除，并在复位处理之后以高速片上振荡器时钟开始执行程序。由看门狗定时器引起的复位自动解除，并在复位处理之后以高速片上振荡器时钟开始执行程序(参阅图 22-2 至 22-4)。在复位后，当 $V_{DD} \geq V_{POR}$ 或 $V_{DD} \geq V_{LVD}$ 时，检测 POR 和 LVD 电路的电源电压所引起的复位自动解除，并在复位处理之后，以高速片上振荡器时钟开始执行程序(参阅第二十三章 上电复位电路和第二十四章 电压检测电路)。

注 执行指令代码 FFH 时，产生非法指令。

通过电路内置仿真器或片上调试仿真器进行仿真时，不会因执行非法指令发生内部复位。

注意事项 1. 向 $\overline{\text{RESET}}$ 引脚输入一个 10 μs 或以上的低电平，以执行外部复位。

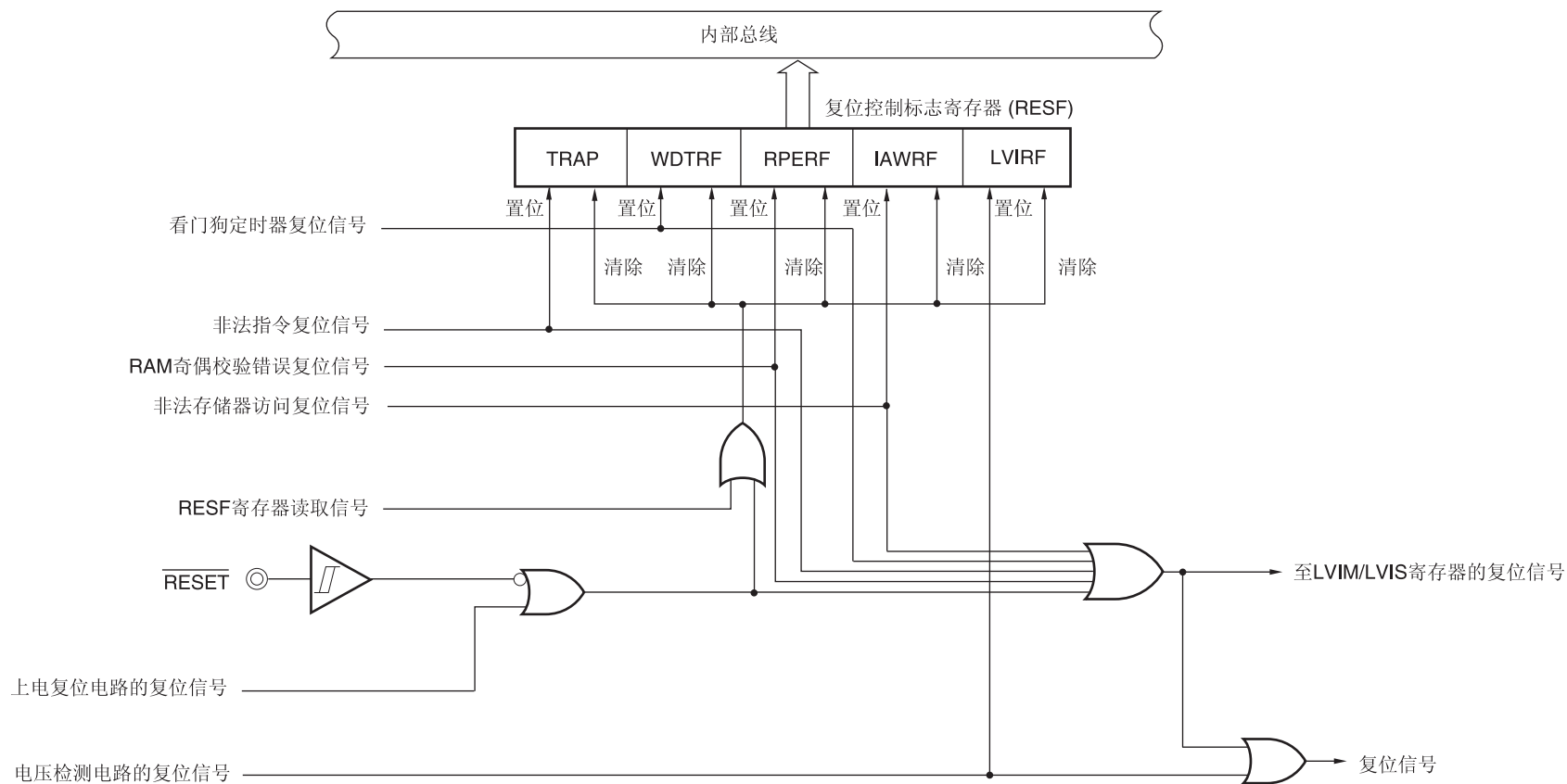
若要在上电时执行外部复位，必须在电源电压处于工作电压范围之内 ($V_{DD} \geq 2.7 \text{ V}$)后，再保持至少 10 μs 的低电平。

2. 产生复位信号时，X1 时钟、XT1 时钟、高速片上振荡器时钟和低速片上振荡器时钟停止振荡。外部主系统时钟输入和外部副系统时钟输入无效。

3. 复位后，因为各 SFR 和扩展 SFR 被初始化，所以端口成为高阻抗。

备注 V_{POR} : POR 电源上升检测电压

图 22-1. 复位功能框图



注意事项 LVD 电路产生内部复位时，LVD 电路不复位。

- 备注**
1. LVIM: 电压检测寄存器
 2. LVIS: 电压检测电平寄存器

图 22-2. 通过 RESET 输入的复位时序

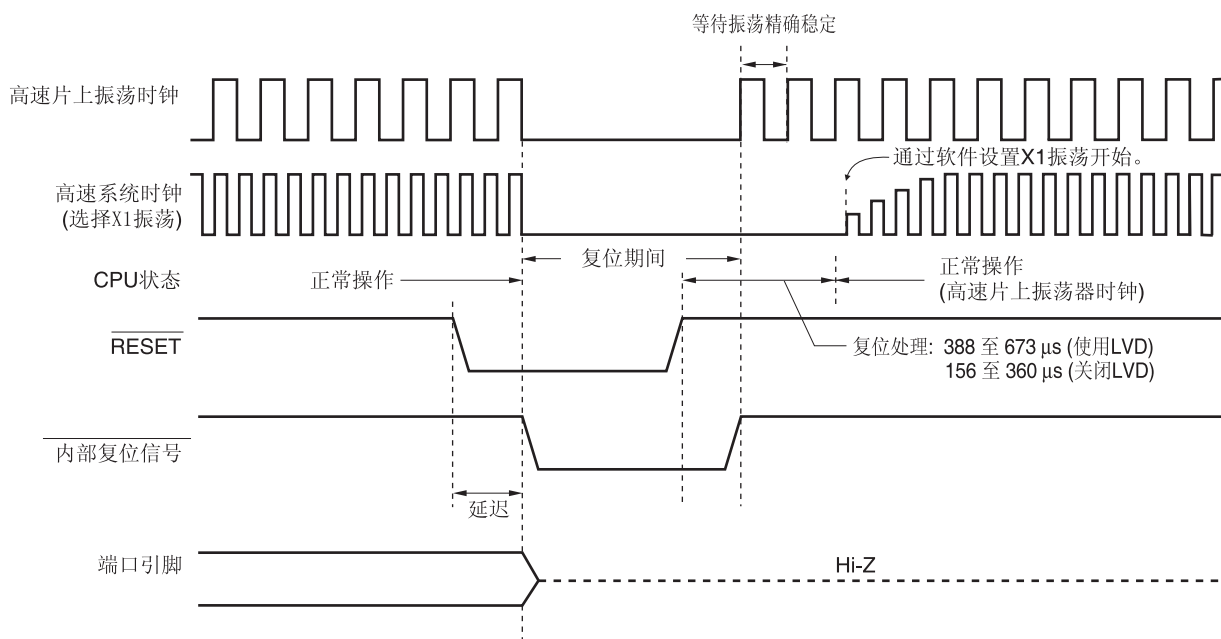
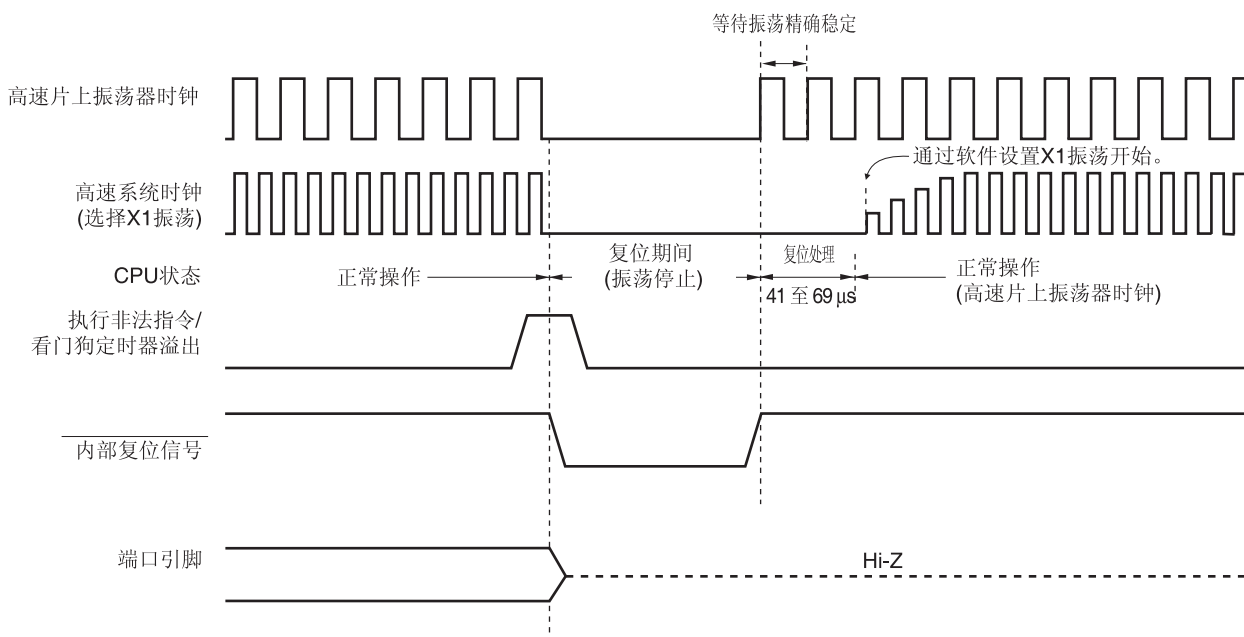
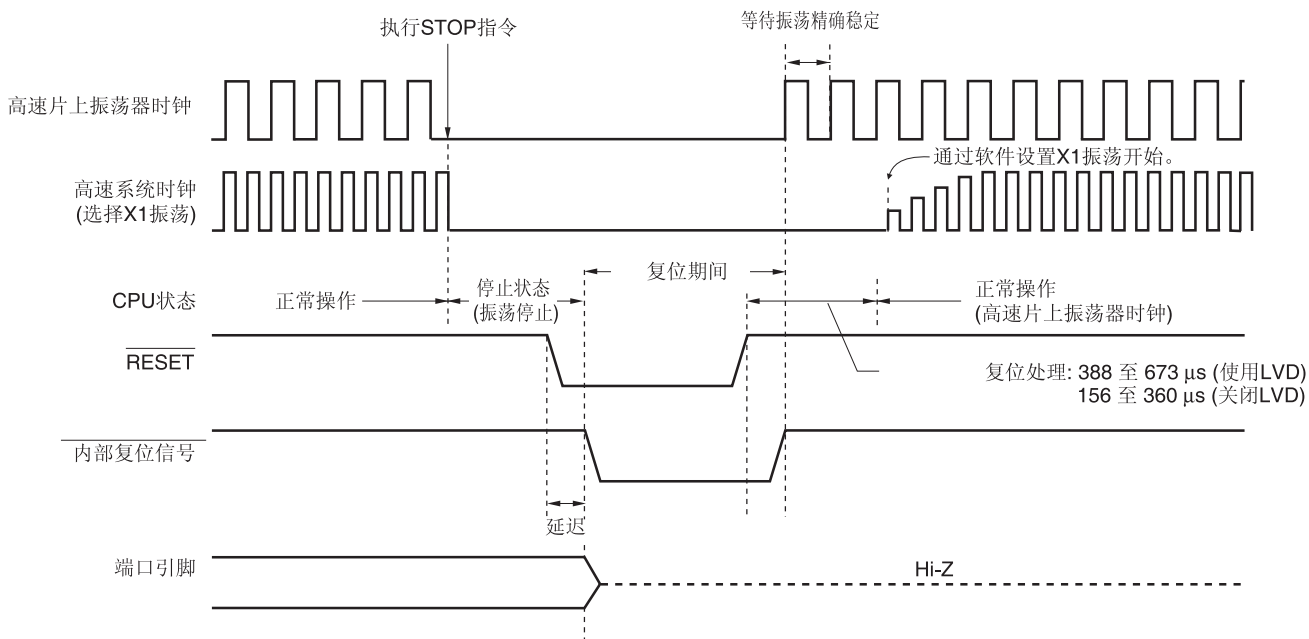


图 22-3. 通过执行非法指令或看门狗定时器溢出的复位时序



注意事项 看门狗定时器的内部复位使看门狗定时器复位。

图 22-4. S 至 P 模式时通过输入 $\overline{\text{RESET}}$ 的复位时序



备注 关于上电复位电路和电压检测电路的复位时序，请参阅第二十三章 上电复位电路和第二十四章 电压检测电路。

表 22-1. 复位期间的工作状态

项目	复位期间	
系统时钟	停止向 CPU 供应时钟。	
主系统时钟	f _{IH}	停止工作
	f _X	停止工作 (X1 和 X2 引脚处于输入端口模式)
	f _{EX}	允许时钟输入 (该引脚处于输入端口模式)
副系统时钟	f _{XT}	停止工作 (XT1 和 XT2 引脚处于输入端口模式)
	f _{EXT}	允许时钟输入 (该引脚处于输入端口模式)
f _{IL}	停止工作	
CPU		
代码闪存	停止工作(读取选项字节之后, 系统工作于 LV (低电压主)模式。)	
数据闪存	停止工作	
RAM	停止工作(当电压为至少上电复位检测电压时, 值被保留。)	
端口(锁存器)	设置 P40 为上拉(引脚复位, POC 复位以外的复位), 成为高阻抗。	
定时器阵列单元	停止工作	
定时器 KB0 至 KB2		
定时器 KC0		
实时时钟 (RTC)		
12 位间隔定时器		
看门狗定时器		
A/D 转换器		
可编程增益放大器		
比较器		
串行阵列单元 0 (SAU0)		
串行阵列单元 4 (DALI/UART4)		
串行接口(IICA)		
乘除法和乘加器		
DMA 控制器		
上电复位功能		可进行检测操作
电压检测功能		停止工作
外部中断	停止工作	
CRC 操作 功能		高速 CRC
		通用 CRC
RAM 奇偶校验错误检测功能		
RAM 保护功能		
SFR 保护功能		
非法存储器存取检测功能		
PLL 功能		

备注 f_{IH}: 高速片上振荡器时钟
 f_X: X1 振荡时钟
 f_{EX}: 外部主系统时钟

f_{XT}: XT1 振荡时钟
 f_{EXT}: 外部副系统时钟
 f_{IL}: 低速片上振荡器时钟

表 22-2. 受理复位后各硬件的状态 (1/6)

硬件		受理复位后的状态 ^{*1}
程序计数器(PC)		复位向量表(0000H, 0001H)的内容被设置。
堆栈指针(SP)		不定
程序状态字(PSW)		06H
RAM	数据存储器	不定
	通用寄存器	不定
处理器模式控制寄存器(PMC)		00H
端口寄存器(P0 至 P4, P7, P14, P20) (输出锁存器)		00H
端口寄存器(P12, P13) (输出锁存器)		不定
端口模式寄存器(PM0 至 PM4, PM7, PM12, PM14, PM20)		FFH
端口模式控制寄存器(PMC0, PMC12, PMC14)		FFH
端口输入模式寄存器(PIM0, PIM1)		00H
端口输出模式寄存器(POM0, POM1, POM20)		00H
上拉电阻选择寄存器(PU0, PU1, PU3, P4, PU7, PU12, PU14, PU20)		00H (PU4 为 01H)
外围输入/输出重定向寄存器 1 (PIOR1)		00H
时钟操作模式控制寄存器(CMC)		00H
时钟操作状态控制寄存器(CSC)		C0H
系统时钟控制寄存器(CKC)		00H
PLL 控制寄存器(PLLCTL)		00H
振荡稳定时间计数器状态寄存器(OSTC)		00H
振荡稳定时间选择寄存器(OSTS)		07H
噪声滤波器允许寄存器 0、1 (NFEN0, NFEN1)		00H
外围允许寄存器 0 至 2 (PER0 至 PER2)		00H
高速片上振荡器调节寄存器(HIOTRM)		注 2
操作速度模式控制寄存器(OSMC)		00H
外围功能切换寄存器 0 (PFSEL0)		00H
定时器阵列单元	定时器数据寄存器 00 至 07 (TDR00 至 TDR07)	0000H
	定时器模式寄存器 00 至 07 (TMR00 至 TMR07)	0000H
	定时器状态寄存器 00 至 07 (TSR00 至 TSR07)	0000H
	定时器输入选择寄存器 0 (TIS0)	00H
	定时器计数器寄存器 00 至 07 (TCR00 至 TCR07)	FFFFH
	定时器通道允许状态寄存器 0 (TE0)	0000H
	定时器通道开始寄存器 0 (TS0)	0000H
	定时器通道停止寄存器 0 (TT0)	0000H
	定时器时钟选择寄存器 0 (TPS0)	0000H
	定时器输出寄存器 0 (TO0)	0000H
	定时器输出允许寄存器 0 (TOE0)	0000H
	定时器输出电平寄存器 0 (TOL0)	0000H
	定时器输出模式寄存器 0 (TOM0)	0000H

注 1. 产生复位信号时以及等待振荡稳定期间, 各硬件状态中仅有 PC 的内容为不定。其他的硬件状态在复位后也保持不变。

2. 复位值因芯片而异。

备注 所安装的特殊功能寄存器(SFR)因产品而异。请参阅 3.1.4 特殊功能寄存器(SFR)区域和 3.1.5 扩展特殊功能寄存器(2nd SFR: 第二特殊功能寄存器)区域。

表 22-2. 受理复位后各硬件的状态 (2/6)

硬件		受理复位后的状态 ^{注1}
定时器KB0至KB2	16位定时器KB比较寄存器00至03, 10至13, 20至23 (TKBCR00至TKBCR03, TKBCR10至TKBCR13, TKBCR20至TKBCR23)	0000H
	16位定时器KB触发比较寄存器0, 1, 2 (TKBTGCR0, TKBTGCR1, TKBTGCR2)	0000H
	16位定时器KB软启动初始占空比寄存器00, 01, 10, 11, 20, 21 (TKBSIR00, TKBSIR01, TKBSIR10, TKBSIR11, TKBSIR20, TKBSIR21)	0000H
	16位定时器KB软启动步宽寄存器00, 01, 10, 11, 20, 21 (TKBSSR00, TKBSSR01, TKBSSR10, TKBSSR11, TKBSSR20, TKBSSR21)	00H
	16位定时器KB抖动计数寄存器00, 01, 10, 11, 20, 21 (TKBDNR00, TKBDNR01, TKBDNR10, TKBDNR11, TKBDNR20, TKBDNR21)	00H
	16位定时器KB触发寄存器0, 1, 2 (TKBTRG0, KBTTRG1, KBTTRG2)	00H
	16位定时器KB标志寄存器0, 1, 2 (TKBFLG0, KTBFLG1, KTBFLG2)	00H
	16位定时器KB比较1L&抖动计数寄存器00, 01, 10, 11, 20, 21 (TKBCRLD00, TKBCRLD01, TKBCRLD10, TKBCRLD11, TKBCRLD20, TKBCRLD21)	0000H
	16位定时器KB操作控制寄存器00, 01, 10, 11, 20, 21 (TKBCTL00, TKBCTL01, KTBCTL10, KTBCTL11, KTBCTL20, KTBCTL21)	0000H
	16位定时器KB最大频率限制设置寄存器0, 1, 2 (TKBMFR0, TKBMFR1, TKBMFR2)	0000H
	16位定时器KB输出控制寄存器00, 01, 10, 11, 20, 21 (TKBIOC00, TKBIOC01, KTBIOC10, KTBIOC11, KTBIOC20, KTBIOC21)	00H
	16位定时器KB标志清除触发寄存器0, 1, 2 (TKBCLR0, KTBCLR1, KTBCLR2)	00H
	强制输出停止功能控制寄存器00, 01, 10, 11, 20, 21 (TKBPACTL00, TKBPACTL01, KTBPACTL10, KTBPACTL11, KTBPACTL20, KTBPACTL21)	0000H
	强制输出停止功能开始触发寄存器0, 1, 2 (TKBPAHFS0, KTBPAHFS1, KTBPAHFS2)	00H
	强制输出停止功能停止触发寄存器0, 1, 2 (TKBPAHFT0, KTBPAHFT1, KTBPAHFT2)	00H
	强制输出停止功能标志寄存器0, 1, 2 (TKBPAFLG0, KTBPAFLG1, KTBPAFLG2)	00H
强制输出停止功能控制寄存器02, 12, 22 (TKBPACTL02, KTBPACTL12, KTBPACTL22)	00H	

注 产生复位信号时以及等待振荡稳定期间，各硬件状态中仅有 PC 的内容为不定。其他控制寄存器的状态与复位后保持相同。

备注 特殊功能寄存器(SFR)因产品而异。详情请参阅 3.1.4 特殊功能寄存器 (SFR) 区域和 3.1.5 扩展特殊功能寄存器 (2nd SFR: 第二特殊功能寄存器) 区域。

表 22-2. 受理复位后各硬件的状态 (3/6)

硬件		受理复位后的状态 ^{注1}
定时器 KC0	16 位定时器 KC 比较寄存器 0 (TKCCR0)	0000H
	16 位定时器 KC 占空比较寄存器 00 至 05 (TKCDUTY00 至 TKCDUTY05)	0000H
	16 位定时器 KC 触发寄存器 0 (TKCTRG0)	00H
	16 位定时器 KC 标志寄存器 0 (TKCFLG0)	00H
	16 位定时器 KC 输出控制寄存器 00 (TKCIOC00)	0000H
	16 位定时器 KC 输出控制寄存器 01 (TKCIOC01)	00H
	16 位定时器 KC 操作控制寄存器 0 (TKCCTL0)	00H
	16 位定时器 KC 输出标志寄存器 0 (TKCTOF0)	00H
实时时钟	秒计数寄存器(SEC)	00H
	分钟计数寄存器(MIN)	00H
	小时计数寄存器(HOUR)	12H
	星期计数寄存器(WEEK)	00H
	日计数寄存器(DAY)	01H
	月计数寄存器(MONTH)	01H
	年计数寄存器(YEAR)	00H
	时钟误差修正寄存器(SUBCUD)	00H
	报警分钟寄存器(ALARMWWM)	00H
	报警小时寄存器(ALARMWWMH)	12H
	报警星期寄存器(ALARMWWMW)	00H
	控制寄存器 0 (RTCC0)	00H
	控制寄存器 1 (RTCC1)	00H
12 位间隔定时器	控制寄存器(ITMC)	0FFFH
看门狗定时器	允许寄存器(WDTE)	1AH/9AH ^{注2}
A/D 转换器	10 位 A/D 转换结果寄存器(ADCR)	0000H
	8 位 A/D 转换结果寄存器(ADCRH)	00H
	模式寄存器 0 至 2 (ADM0 至 ADM2)	00H
	转换结果比较上限值设置寄存器(ADUL)	FFH
	转换结果比较下限值设置寄存器(ADLL)	00H
	A/D 测试寄存器(ADTES)	00H
	模拟输入通道选择寄存器(ADS)	00H
	A/D 端口配置寄存器(ADPC)	00H
可编程增益放大器	可编程增益放大器控制寄存器(PGACTL)	00H
	可编程增益放大器输入通道选择寄存器(PGAINS)	00H
比较器	窗口比较器功能设置寄存器(CMPWDC)	00H
	比较器 0 至 5 控制寄存器(C0CTL 至 5CTL)	00H
	比较器上升沿允许寄存器 0 (CMPEGP0)	00H
	比较器下降沿允许寄存器 0 (CMPEGN0)	00H
	外部中断控制寄存器(INTPCTL)	00H
	比较器内部基准电压控制寄存器(CVRCTL)	00H
	比较器内部基准电压选择寄存器 0 至 2 (C0RVM 至 C2RVM)	00H
	比较器输出监视寄存器(CMPMON)	00H
	比较器输入切换控制寄存器(CMPSEL)	00H

注 1. 产生复位信号时以及等待振荡稳定期间，硬件状态中仅有 PC 的内容为不定。其他的硬件状态在复位后也保持不变。

2. WDTE 的复位值取决于选项字节的设置。

备注 所安装的特殊功能寄存器(SFR)因产品而异。请参阅 3.1.4 特殊功能寄存器(SFR)区域和 3.1.5 扩展特殊功能寄存器(2nd SFR: 第二特殊功能寄存器) 区域。

表 22-2. 受理复位后各硬件的状态 (4/6)

硬件		受理复位后的状态 ^{注1}
串行阵列单元 0 (SAU0)	串行数据寄存器 00 至 03 (SDR00 至 SDR03)	0000H
	串行状态寄存器 00 至 03 (SSR00 至 SSR03)	0000H
	串行标志清除触发寄存器 00 至 03 (SIR00 至 SIR03)	0000H
	串行模式寄存器 00 至 03 (SMR00 至 SMR03)	0020H
	串行通信操作设置寄存器 00 至 03 (SCR00 至 SCR03)	0087H
	串行通道允许状态寄存器 0 (SE0)	0000H
	串行通道开始寄存器 0 (SS0)	0000H
	串行通道停止寄存器 0 (ST0)	0000H
	串行时钟选择寄存器 0 (SPS0)	0000H
	串行输出寄存器 0 (SO0)	0F0FH
	串行输出允许寄存器 0 (SOE0)	0000H
	串行输出电平寄存器 0 (SOL0)	0000H
	串行待机控制寄存器 0 (SSC0)	0000H
	输入切换控制寄存器 (ISC)	00H
串行阵列单元 4 (DALI/UART4)	串行数据寄存器 40, 41 (SDR40, SDR41)	0000H
	串行状态寄存器 40, 41 (SSR40, SSR41)	0000H
	串行标志清除触发寄存器 40, 41 (SIR40, SIR41)	0000H
	串行模式寄存器 40, 41 (SMR40, SMR41)	0020H
	串行通信操作设置寄存器 40, 41 (SCR40, SCR41)	0087H
	串行通道允许状态寄存器 4 (SE4)	0000H
	串行通道开始寄存器 4 (SS4)	0000H
	串行通道停止寄存器 4 (ST4)	0000H
	串行时钟选择寄存器 4 (SPS4)	0000H
	串行输出寄存器 4 (SO4)	0F0FH
	串行输出允许寄存器 4 (SOE4)	0000H
	串行输出电平寄存器 4 (SOL4)	0000H
	串行待机控制寄存器 4 (SSC4)	0000H
	DALI 发送数据寄存器 H4, L4 (SDTH4, SDTL4)	0000H
	DALI 接收数据寄存器 H4, L4 (SDCH4, SDCL4)	0000H
	串行选项控制寄存器 4 (SOC4)	0000H
	单线 UART 控制寄存器 (SUCTL4)	00H
串行接口 IICA	IICA 移位寄存器 0 (IICA0)	00H
	IICA 状态寄存器 0 (IICS0)	00H
	IICA 标志寄存器 0 (IICF0)	00H
	IICA 控制寄存器 00 (IICCTL00)	00H
	IICA 控制寄存器 01 (IICCTL01)	00H
	IICA 低电平宽度设置寄存器 0 (IICWL0)	FFH
	IICA 高电平宽度设置寄存器 0 (IICWH0)	FFH
	从属地址寄存器 0 (SVA0)	00H

注 产生复位信号时以及等待振荡稳定期间，硬件状态中仅有 PC 的内容为不定。其他控制寄存器的状态与复位后保持相同。

备注 特殊功能寄存器(SFR)因产品而异。详情请参阅 3.1.4 特殊功能寄存器 (SFR) 区域和 3.1.5 扩展特殊功能寄存器 (2nd SFR: 第二特殊功能寄存器) 区域。

表 22-2. 受理复位后各硬件的状态 (5/6)

硬件		受理复位后的状态 ^{注1}
乘除法和乘加器	乘法/除法数据寄存器 A (L) (MDAL)	0000H
	乘法/除法数据寄存器 A (H) (MDAH)	0000H
	乘法/除法数据寄存器 B (L) (MDBL)	0000H
	乘法/除法数据寄存器 B (H) (MDBH)	0000H
	乘法/除法数据寄存器 C (L) (MDCL)	0000H
	乘法/除法数据寄存器 C (H) (MDCH)	0000H
	乘法/除法控制寄存器(MDUC)	00H
复位功能	复位控制标志寄存器(RESF)	不确定 ^{注2}
电压检测电路(LVD)	电压检测寄存器(LVIM)	00H ^{注2}
	电压检测电平寄存器(LVIS)	00H/01H/81H ^{注2,3}
DMA 控制器	SFR 地址寄存器 0, 1 (DSA0, DSA1)	00H
	RAM 地址寄存器 0L, 0H, 1L, 1H (DRA0L, DRA0H, DRA1L, DRA1H)	00H
	字节计数寄存器 0L, 0H, 1L, 1H (DBC0L, DBC0H, DBC1L, DBC1H)	00H
	模式控制寄存器 0, 1 (DMC0, DMC1)	00H
	操作控制寄存器 0, 1 (DRC0, DRC1)	00H

注 1. 产生复位信号时以及等待振荡稳定期间，硬件状态中仅有 PC 的内容为不定。其他的硬件状态在复位后也保持不变。

2. 这些值因复位源而异。复位源和寄存器的关系如下所示。

复位源		RESET 输入	通过 POR 的复位	通过执行非法指令的复位	通过 WDT 的复位	通过 RAM 奇偶校验错误的复位	通过存取非法存储器的复位	通过 LVD 的复位
RESF	TRAP 位	清除 (0)		置位 (1)	保持			保持
	WDTRF 位			保持	置位 (1)	保持		
	RPERF 位			保持		置位 (1)	保持	
	IAWRF 位			保持			置位 (1)	
	LVIRF 位			保持				
LVIM	LVISEN 位	清除 (0)						保持
	LVIOMSK 位	保持						
	LVIF 位							
LVIS		清除(00H/01H/81H)						

3. LVD 复位以外的复位时设置如下。

- 当选项字节 LVIMDS1, LVIMDS0 = 1, 0: 00H
- 当选项字节 LVIMDS1, LVIMDS0 = 1, 1: 81H
- 当选项字节 LVIMDS1, LVIMDS0 = 0, 1: 01H

备注 所安装的特殊功能寄存器(SFR)因产品而异。请参阅 3.1.4 特殊功能寄存器(SFR)区域和 3.1.5 扩展特殊功能寄存器(2nd SFR: 第二特殊功能寄存器) 区域。

表 22-2. 受理复位后各硬件的状态 (6/6)

硬件		受理复位后的状态 ^{注1}
中断	请求标志寄存器 0L, 0H, 1L, 1H, 2L, 2H (IF0L, IF0H, IF1L, IF1H, IF2L, IF2H)	00H
	屏蔽标志寄存器 0L, 0H, 1L, 1H, 2L, 2H (MK0L, MK0H, MK1L, MK1H, MK2L, MK2H)	FFH
	优先级指定标志寄存器 00L, 00H, 01L, 01H, 02L, 02H, 10L, 10H, 11L, 11H, 12L, 12H (PR00L, PR00H, PR01L, PR01H, PR10L, PR10H, PR11L, PR11H, PR02L, PR02H, PR12L, PR12H)	FFH
	外部中断上升沿允许寄存器 0 至 2 (EGP0 至 EGP2)	00H
	外部中断下降沿允许寄存器 0 至 2 (EGN0 至 EGN2)	00H
	中断屏蔽标志寄存器 0 (INTMK0)	FFH
	中断监视标志寄存器 0 (INTMF0)	00H
安全功能	闪存 CRC 控制寄存器(CRC0CTL)	00H
	闪存 CRC 运算结果寄存器(PGCRCL)	0000H
	CRC 输入寄存器(CRCIN)	00H
	CRC 数据寄存器(CRCD)	0000H
	非法存储器存取检测控制寄存器 0, 1 (IAWCTL0, IAWCTL1)	00H
	RAM 奇偶校验错误控制寄存器(RPECTL)	00H
闪存	数据闪存控制寄存器(DFLCTL)	00H
十进制调整(BCD)电路	BCD 校正结果寄存器(BCDADJ)	不确定

注 产生复位信号时以及等待振荡稳定期间，硬件状态中仅有 PC 的内容为不定。其他控制寄存器的状态与复位后保持相同。

备注 特殊功能寄存器(SFR)因产品而异。详情请参阅 3.1.4 特殊功能寄存器 (SFR) 区域和 3.1.5 扩展特殊功能寄存器 (2nd SFR: 第二特殊功能寄存器) 区域。

22.1 确认复位源的寄存器

RL78/I1A 中存在着多种复位源。复位控制标志寄存器(RESF)用于存储产生了复位请求的源。

使用 8 位存储器操作指令读取 RESF 寄存器。

通过输入 RESET、上电复位 (POR)电路引起复位、以及读取 RESF 寄存器，可清除 TRAP、WDTRF、RPERF、IAWRF 和 LVIRF 标志。

图 22-5. 复位控制标志寄存器(RESF)的格式

地址: FFFA8H 复位后: 00H^{#1} R

符号	7	6	5	4	3	2	1	0
RESF	TRAP	0	0	WDTRF	0	RPERF	IAWRF	LVIRF
TRAP	执行非法指令的内部复位请求 ^{#2}							
0	无内部复位请求，或 RESF 寄存器被清除。							
1	产生内部复位请求。							
WDTRF	看门狗定时器(WDT) 的内部复位请求							
0	无内部复位请求，或 RESF 寄存器被清除。							
1	产生内部复位请求。							
RPERF	RAM 奇偶校验的内部复位请求							
0	无内部复位请求，或 RESF 寄存器被清除。							
1	产生内部复位请求。							
IAWRF	非法存储器存取的内部复位请求							
0	无内部复位请求，或 RESF 寄存器被清除。							
1	产生内部复位请求。							
LVIRF	电压检测电路 (LVD) 的内部复位请求							
0	无内部复位请求，或 RESF 寄存器被清除。							
1	产生内部复位请求。							

注 1. 复位后的值因复位源而异。

2. 执行指令代码 FFH 时，产生非法指令。

通过电路内置仿真器或片上调试仿真器进行仿真时，不会因执行非法指令发生内部复位。

注意事项 1. 不要用 1 位存储器操作指令读取数据。

2. 从 RAM 获取的指令代码，在执行过程中不受奇偶校验错误检测的影响。然而，由于 RAM 获取指令代码而引起的 RAM 数据读取要接受奇偶校验错误检测。

3. 由于 RL78 执行流水操作，CPU 会执行预取，所以 CPU 可能读取分配至所用 RAM 之后的未初始化的 RAM 区域，以至于产生 RAM 奇偶校验错误。因此，允许产生 RAM 奇偶校验错误复位 (RPERDIS = 0) 时，务必对所用 RAM 区域 + 10 字节的区域进行初始化。

产生复位请求时 RESF 寄存器的状态如图 22-3 所示。

表 22-3. 产生复位请求时 RESF 寄存器的状态

标志 \ 复位源	RESET 输入	通过 POR 的 复位	通过执行非法 指令的复位	通过 WDT 的 复位	通过 RAM 奇 偶校验错误 的复位	通过存取非 法存储器的 复位	通过 LVD 的 复位
TRAP 位	清除 (0)	清除 (0)	置位 (1)	保持	保持	保持	保持
WDTRF 位			保持	置位 (1)			
RPERF 位			保持	置位 (1)			
IAWRF 位			保持	置位 (1)			
LVIRF 位			保持	置位 (1)			

第二十三章 上电复位电路

23.1 上电复位电路的功能

上电复位电路 (POR)具有以下功能。

- 上电时产生内部复位信号。
当电源电压 (V_{DD})超过 $1.51\text{ V} \pm 0.03\text{ V}$ 时解除复位。
- 比较电源电压(V_{DD})和检测电压($V_{PDR} = 1.50\text{ V} \pm 0.03\text{ V}$)，在 $V_{DD} < V_{PDR}$ 时产生内部复位信号。

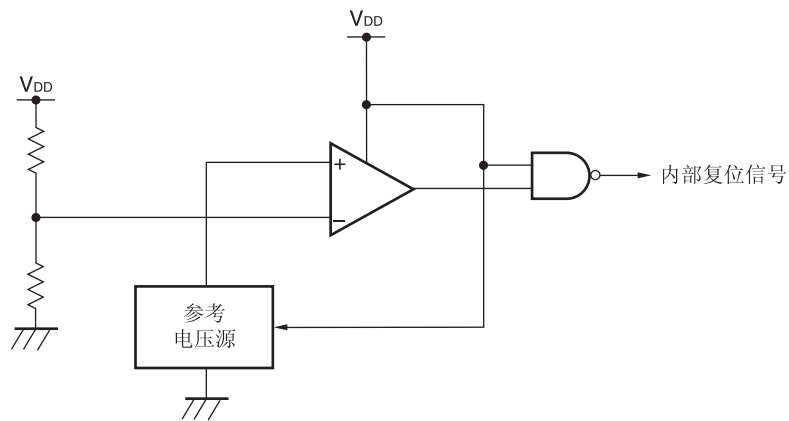
注意事项 如果 POR 电路中产生内部复位信号，则复位控制标志寄存器(RESF)的 TRAP、WDTRF、RPERF、IAWRF 和 LVIRF 标志将被清除。

备注 本产品含有多个能产生内部复位信号的硬件功能。由于看门狗定时器(WDT)、电压检测电路(LVD)、执行非法指令、RAM 奇偶校验错误或存取非法存储器而产生内部复位信号时，用于指示复位源的标志配置于复位控制标志寄存器(RESF)中。由看门狗定时器(WDT)、电压检测电路(LVD)、执行非法指令、RAM 奇偶校验错误或存取非法存储器而产生内部复位信号时，RESF 寄存器不会被清除为 00H，而是标志被设置为 1。
关于 RESF 寄存器的详情，请参阅第二十二章 复位功能。

23.2 上电复位电路的配置

上电复位电路的框图如图 23-1 所示。

图 23-1. 上电复位电路的框图



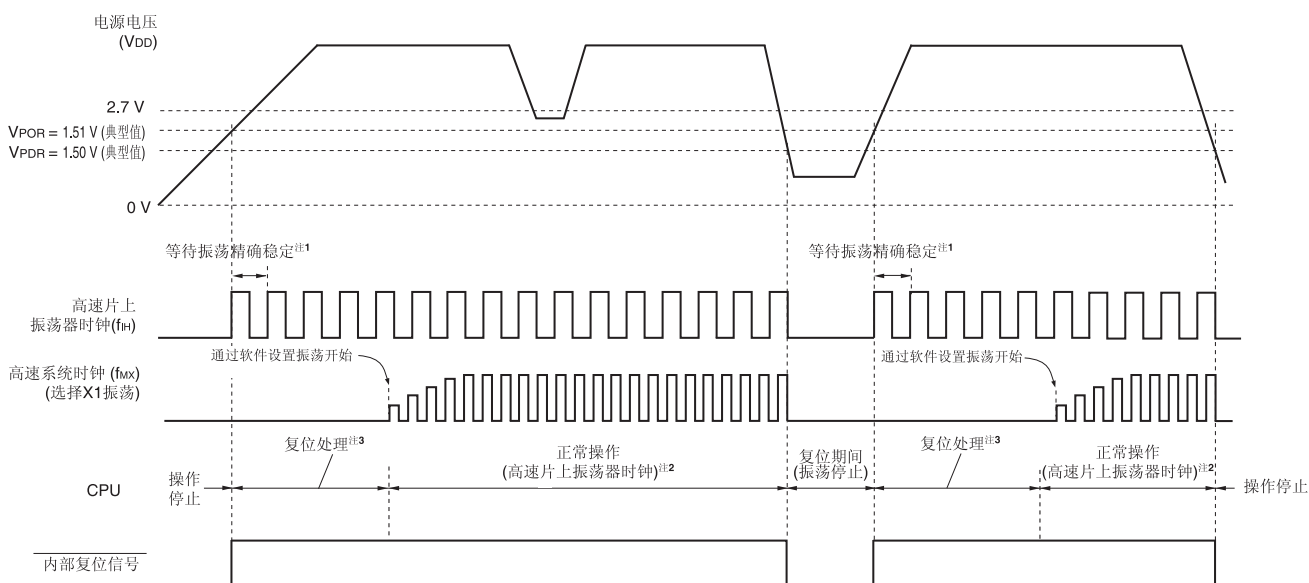
23.3 上电复位电路的操作

- 上电时产生内部复位信号。当电源电压 (V_{DD}) 超过检测电压 ($V_{PDR} = 1.51\text{ V} \pm 0.03\text{ V}$) 时解除复位。
- 比较电源电压 (V_{DD}) 和检测电压 ($V_{PDR} = 1.50\text{ V} \pm 0.03\text{ V}$)。在 $V_{DD} < V_{PDR}$ 时产生内部复位信号。

上电复位电路和电压检测电路产生的内部复位信号的时序如下所示。

图 23-2. 上电复位电路和电压检测电路产生的内部复位信号的时序 (1/2)

(1) 当 LVD 为 OFF 时 (选项字节 000C1H: VPOC2 = 1)



注 1. 高速片上振荡器时钟的振荡精度稳定等待时间包含在内部电压稳定等待的复位处理时间内。

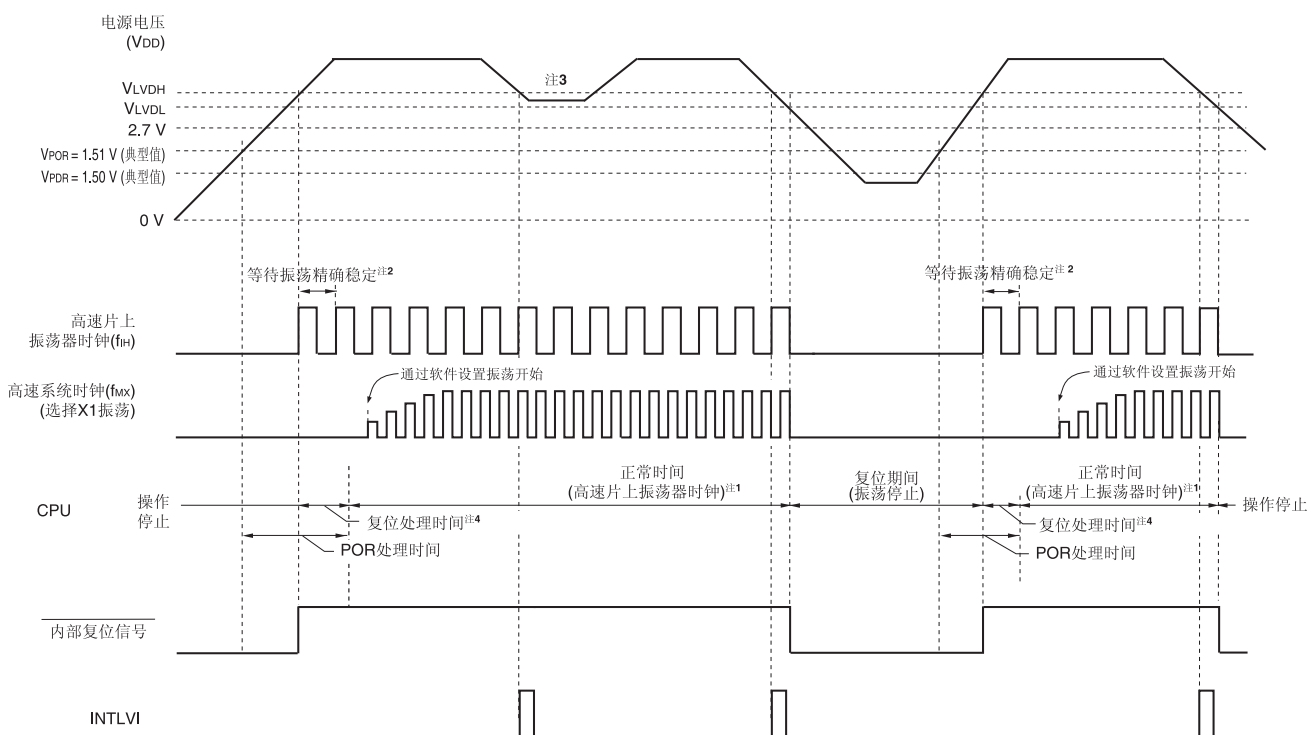
2. 可以将 CPU 时钟从高速片上振荡器时钟切换至高速系统时钟或副系统时钟。如要使用 X1 时钟，须利用振荡稳定时间计数器状态寄存器(OSTC)确认过振荡稳定时间之后再切换。如要使用 XT1 时钟，则须利用定时器功能来确认过振荡稳定时间再进行切换。

3. 复位处理时间： 265 至 407 μ s

备注 V_{POR}: POR 电源上升检测电压
V_{PDR}: POR 电源下降检测电压

图 23-2. 上电复位电路和电压检测电路产生的内部复位信号的时序 (2/2)

(2) 当 LVD 为中断&复位模式时 (选项字节 000C1: LVIMDS1, LVIMDS0 = 1, 0)



- 注 1. 可以将 CPU 时钟从高速片上振荡器时钟切换至高速系统时钟或副系统时钟。如要使用 X1 时钟，须利用振荡稳定时间计数器状态寄存器(OSTC)确认过振荡稳定时间之后再切换。如要使用 XT1 时钟，则须利用定时器功能来确认过振荡稳定时间再进行切换。
- 2. 高速片上振荡器时钟的振荡精度稳定等待时间包含在内部电压稳定等待的复位处理时间内。
- 3. 在产生第一个中断请求信号(INTLVI)之后，电压检测电平寄存器(LVIS)的 LVIL 和 LVIMD 位将被自动设置为 1。如果出现工作电压不低于电压检测电平(VLVL)而是返回 2.7 V 或更高的情况时，在产生 INTLVI 之后，执行所需的备份处理，并利用软件依次指定初始设置（请参阅图 24-8. 中断和复位模式的初始设置）。
- 4. 复位处理时间： 497 至 720 μs

备注 VLVDH、VLVDL: LVD 检测电压
 VPOR: POR 电源上升检测电压
 VPDR: POR 电源下降检测电压

23.4 上电复位电路的注意事项

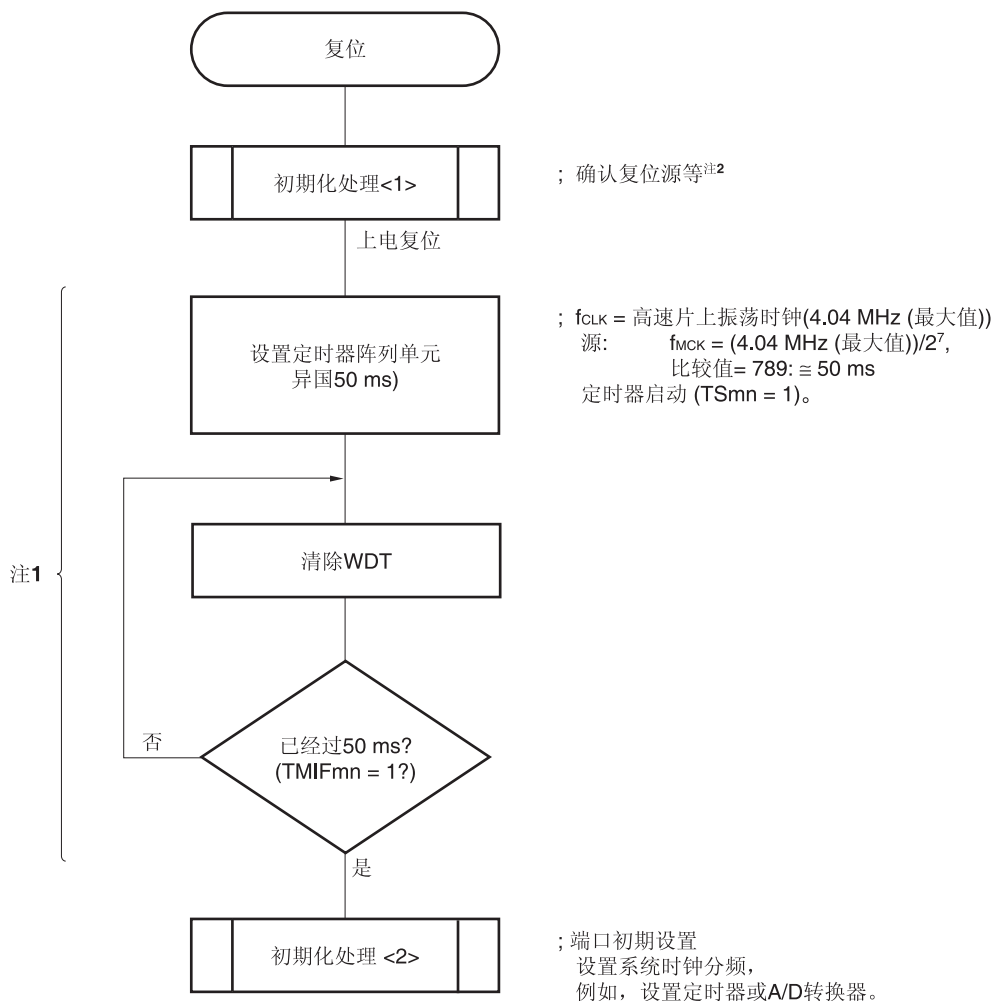
电源电压(V_{DD})在 POR 检测电压(V_{POR}、V_{PDR})附近在一定期间内波动的系统中，单片机可能会重复地进入复位状态/解除复位状态。通过以下方法，可以任意设置从解除复位到单片机开始动作的时间：

<处理>

解除复位后，通过基于定时器的软件计数器，依据各系统的不同，等待其电源电压波动期间过后，进行对端口等的初始化。

图 23-3. 复位解除后的软件处理示例 (1/2)

- 当 POR 检测电压附近的电源电压波动在 50 ms 以内时



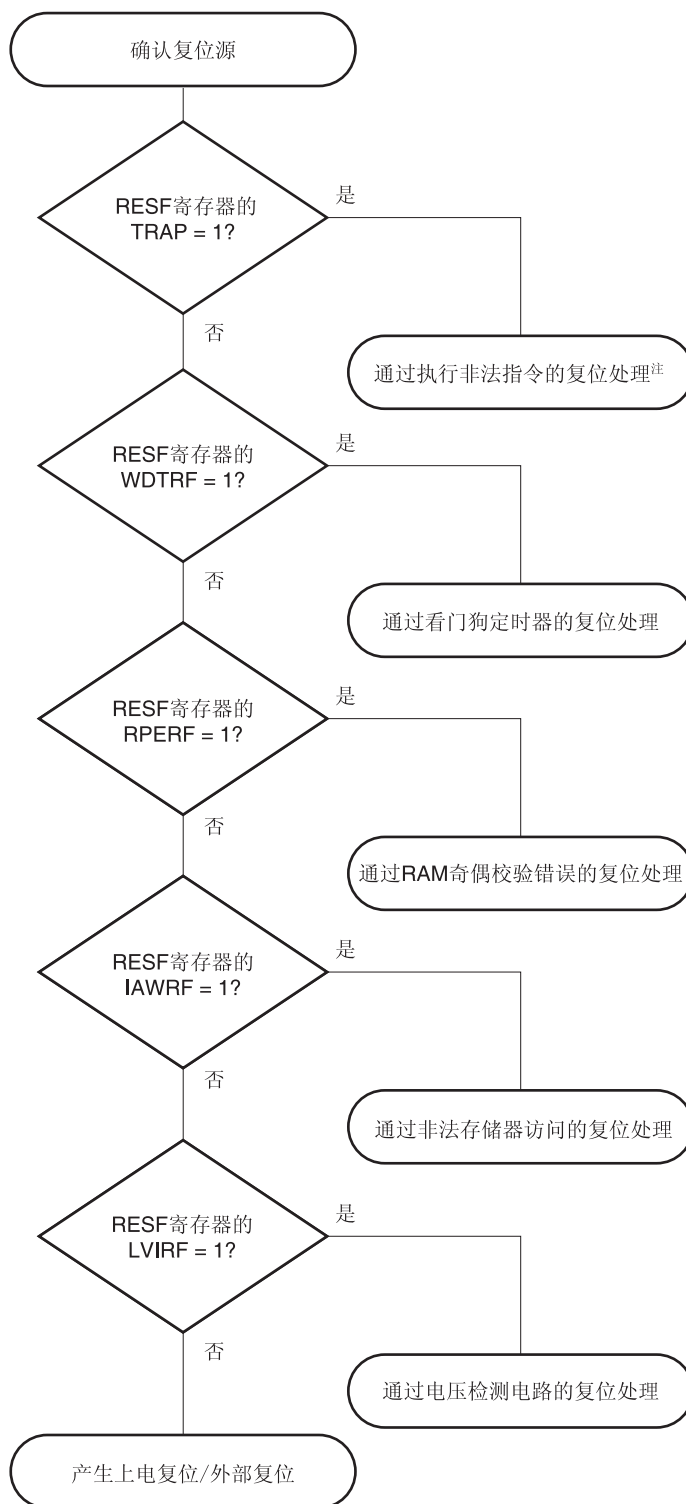
注 1. 如果在此期间再次产生复位，则不开始初始化处理<2>。

2. 流程图如下页所示。

备注 m = 0
n = 0 至 7

图 23-3. 复位解除后的软件处理示例 (2/2)

• 确认复位源



注 执行指令代码 FFH 时，产生非法指令。
通过电路内置仿真器或片上调试仿真器进行仿真时，不会因执行非法指令发生内部复位。

第二十四章 电压检测电路

24.1 电压检测电路的功能

电压检测(LVD)电路具有如下功能。

- 比较电源电压(V_{DD})和检测电压(V_{LVDH} , V_{LVDL}), 产生内部复位或内部中断信号。
- 电源检测电压(V_{LVDH} , V_{LVDL})的检测电平, 可通过选项字节从 6 阶段中选择(详情参阅**第二十七章 选项字节**)。
- 可在 STOP 模式下操作。
- 使用选项字节时可选择如下三种的操作模式。

(a) 中断&复位模式 (选项字节 LVIMDS1, LVIMDS0 = 1, 0)

对于由选项字节 000C1H 选择的两个检测电压, 高电压检测电平 (V_{LVDH}) 用于发生中断和解除复位, 低电压检测电平 (V_{LVDL}) 用于产生复位。

(b) 复位模式 (选项字节 LVIMDS1, LVIMDS0 = 1, 1)

通过选项字节 000C1H 选择的检测电压(V_{LVD}), 可用来产生和解除复位。

(c) 中断模式 (选项字节 LVIMDS1, LVIMDS0 = 0, 1)

通过选项字节 000C1H 选择的检测电压(V_{LVD}), 可用来产生和解除中断。

在中断&复位模式下可设置两个检测电压(V_{LVDH} , V_{LVDL}), 在复位模式以及中断模式下可设置一个检测电压(V_{LVD})。可通过选项字节 (LVIMDS0, LVIMDS1)的选择, 产生如下的复位和中断信号。

中断&复位模式 (LVIMDS1, LVIMDS0 = 1, 0)	复位模式 (LVIMDS1, LVIMDS0 = 1, 1)	中断模式 (LVIMDS1, LVIMDS0 = 0, 1)
当 $V_{DD} < V_{LVDH}$ 时, 产生内部中断信号, 当 $V_{DD} < V_{LVDL}$ 时, 则产生内部复位信号。 当 $V_{DD} \geq V_{LVDH}$ 时, 解除内部复位信号。	当 $V_{DD} < V_{LVD}$ 时, 产生内部复位信号, 当 $V_{DD} \geq V_{LVD}$ 时, 则解除复位信号。	当 V_{DD} 降至比 V_{LVD} 小时 ($V_{DD} < V_{LVD}$) 或者当 V_{DD} 变成 V_{LVD} 或更大时 ($V_{DD} \geq V_{LVD}$), 产生内部中断信号。 上电时若 $V_{DD} \geq V_{LVD}$, 则解除内部复位信号。

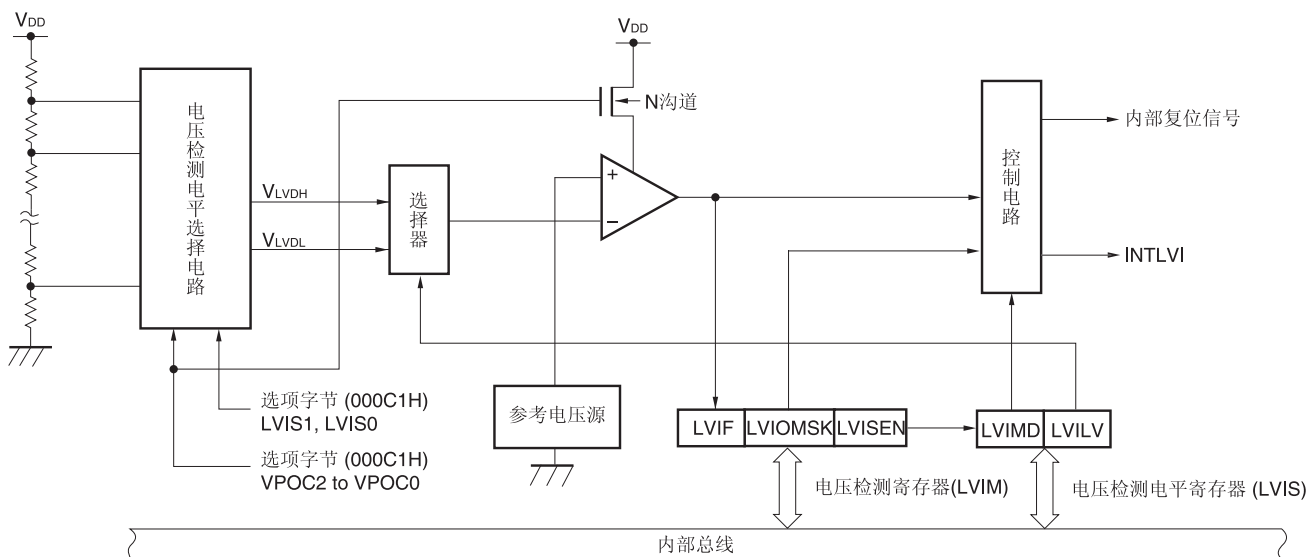
在电压检测电路工作时, 可以通过读取电压检测标志(LVIF: 电压检测寄存器 (LVIM)的位 0), 来检验电源电压或来自外部引脚的输入电压是大于还是小于检测电平。

如果发生复位, 则复位控制标志寄存器(RESF)的位 0 (LVIRF) 将被设为 1。关于 RESF 寄存器的详情, 请参阅**第二十二章 复位功能**。

24.2 电压检测电路的配置

电压检测电路的框图如图 24-1 所示。

图 24-1. 电压检测电路的框图



24.3 控制电压检测电路的寄存器

使用如下的寄存器控制电压检测电路。

- 电压检测寄存器(LVIM)
- 电压检测电平寄存器(LVIS)

(1) 电压检测寄存器(LVIM)

该寄存器用来设置电压检测电平寄存器 (LVIS)的允许/禁止改写，确认 LVD 输出的屏蔽状态。

使用 1 位或 8 位存储器操作指令来设置该寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 24-2. 电压检测寄存器(LVIM)的格式

地址: FFFA9H 复位后: 00H^{注1} R/W^{注2}

符号	<7>	6	5	4	3	2	<1>	<0>
LVIM	LVISEN	0	0	0	0	0	LVIOMSK	LVIF

LVISEN	电压检测电平寄存器 (LVIS)的允许/禁止改写的设置
0	禁止改写
1	允许改写 ^{注3}

LVIOMSK	LVD 输出的屏蔽状态标志
0	屏蔽无效
1	屏蔽有效 ^{注3,4}

LVIF	电压检测标志
0	电源电压 (V _{DD}) ≥ 检测电压 (V _{LVD})或当 LVD 操作被禁止时
1	电源电压 (V _{DD}) < 检测电压 (V _{LVD})

注 1. 复位值因复位源而异。

如果通过 LVD 使 LVIS 寄存器复位，则其不被复位而是保持当前值。其他复位时，LVISEN 被清除为 0。

2. 位 0 和位 1 为只读位。

3. 仅在选项字节设置 LVIMDS1 和 LVIMDS0 为 1 和 0 (中断&复位模式) 时可以设置 (其他模式无效)。

4. 在以下期间内，LVIOMSK 位被自动设置为 1，由 LVD 引起的复位或中断被屏蔽。

- LVISEN = 1 的期间
- 从发生 LVD 中断开始到 LVD 检测电压稳定为止的等待时间
- 从 LVILV 位的值发生变化到 LVD 检测电压稳定为止的等待时间

(2) 电压检测电平寄存器(LVIS)

该寄存器用来选择电压检测电平。

使用 1 位或 8 位存储器操作指令来设置该寄存器。

产生复位信号后，该寄存器被设置为 00H/01H/81H^{注1}。

图 24-3. 电压检测电平寄存器(LVIS)的格式

地址: FFFAAH 复位后: 00H/01H/81H^{注1} R/W

符号	<7>	6	5	4	3	2	1	<0>
LVIS	LVIMD	0	0	0	0	0	0	LVILV

LVIMD ^{注2}	电压检测的操作模式
0	中断模式
1	复位模式

LVILV ^{注2}	LVD 检测电平
0	高电压检测电平 (VLVDH)
1	低电压检测电平 (VLVDL 或 VLVD)

注 1. 复位值因复位源和选项字节的设置而异。

LVD 复位时，该寄存器不被清除为 00H。

LVD 复位以外的复位时设置如下。

- 选项字节 LVIMDS1, LVIMDS0 = 1, 0 时: 00H
- 选项字节 LVIMDS1, LVIMDS0 = 1, 1 时: 81H
- 选项字节 LVIMDS1, LVIMDS0 = 0, 1 时: 01H

2. 仅当 LVIMDS1 和 LVIMDS0 由选项字节设为 1 和 0（中断&复位模式）时，才可写入 0。在其他情况下不允许写入操作，通过产生复位或中断自动切换数值。

注意事项 1. 改写 LVIS 寄存器时，请先将 LVISEN 位(LVIM 寄存器的位 7)设置为 1 后再进行。

2. 可使用选项字节(000C1H)设置 LVD 操作模式和检测电压 (VLVDH, VLVDL)。选项字节(000C1H)的设置如表 24-1 所示。有关选项字节，请参阅第二十七章 选项字节。

表 24-1. 通过用户选项字节 (000C1H)设置 LVD 操作模式和检测电压 (1/2)

地址: 000C1H/010C1H[※]

7	6	5	4	3	2	1	0
VPOC2	VPOC1	VPOC0	1	LVIS1	LVIS0	LVIMDS1	LVIMDS0

• LVD 设置(中断&复位模式时的设置)

检测电压			选项字节设置值						
VLVDH		VLVDL	模式设置		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0
上升沿	下降沿	下降沿	LVIMDS1	LVIMDS0					
2.92 V	2.86 V	2.75 V	1	0	0	1	1	1	0
3.02 V	2.96 V							0	1
4.06 V	3.98 V							0	0
其他			禁止设置						

• LVD 设置(复位模式时的设置)

检测电压		选项字节设置值						
VLVD		模式设置		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0
上升沿	下降沿	LVIMDS1	LVIMDS0					
2.81 V	2.75 V	1	1	0	1	1	1	1
2.92 V	2.86 V			0	1	1	1	0
3.02 V	2.96 V			0	1	1	0	1
3.13 V	3.06 V			0	0	1	0	0
3.75 V	3.67 V			0	1	0	0	0
4.06 V	3.98 V			0	1	1	0	0
其他		禁止设置						

• LVD 设置(中断模式时的设置)

检测电压		选项字节设置值						
VLVD		模式设置		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0
上升沿	下降沿	LVIMDS1	LVIMDS0					
2.81 V	2.75 V	0	1	0	1	1	1	1
2.92 V	2.86 V			0	1	1	1	0
3.02 V	2.96 V			0	1	1	0	1
3.13 V	3.06 V			0	0	1	0	0
3.75 V	3.67 V			0	1	0	0	0
4.06 V	3.98 V			0	1	1	0	0
其他		禁止设置						

注 在使用引导交换操作时, 由于 000C1H 与 010C1H 相互切换, 须将 000C1H 设置为与 010C1H 相同的值。

表 24-1. 通过用户选项字节 (000C1H)设置 LVD 操作模式和检测电压 (2/2)

地址: 000C1H/010C1H[※]

7	6	5	4	3	2	1	0
VPOC2	VPOC1	VPOC0	1	LVIS1	LVIS0	LVIMDS1	LVIMDS0

• LVD 设置(LVD 关闭时)

检测电压		选项字节设置值						
V _{LVD}		模式设置		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0
上升沿	下降沿	LVIMDS1	LVIMDS0					
-	-	0/1	1	1	×	×	×	×
其他		禁止设置						

注 在使用引导交换操作时，由于 000C1H 与 010C1H 相互切换，须将 000C1H 设置为与 010C1H 相同的值。

备注 ×: 忽略

24.4 电压检测电路的操作

24.4.1 使用复位模式时

- 开始操作时

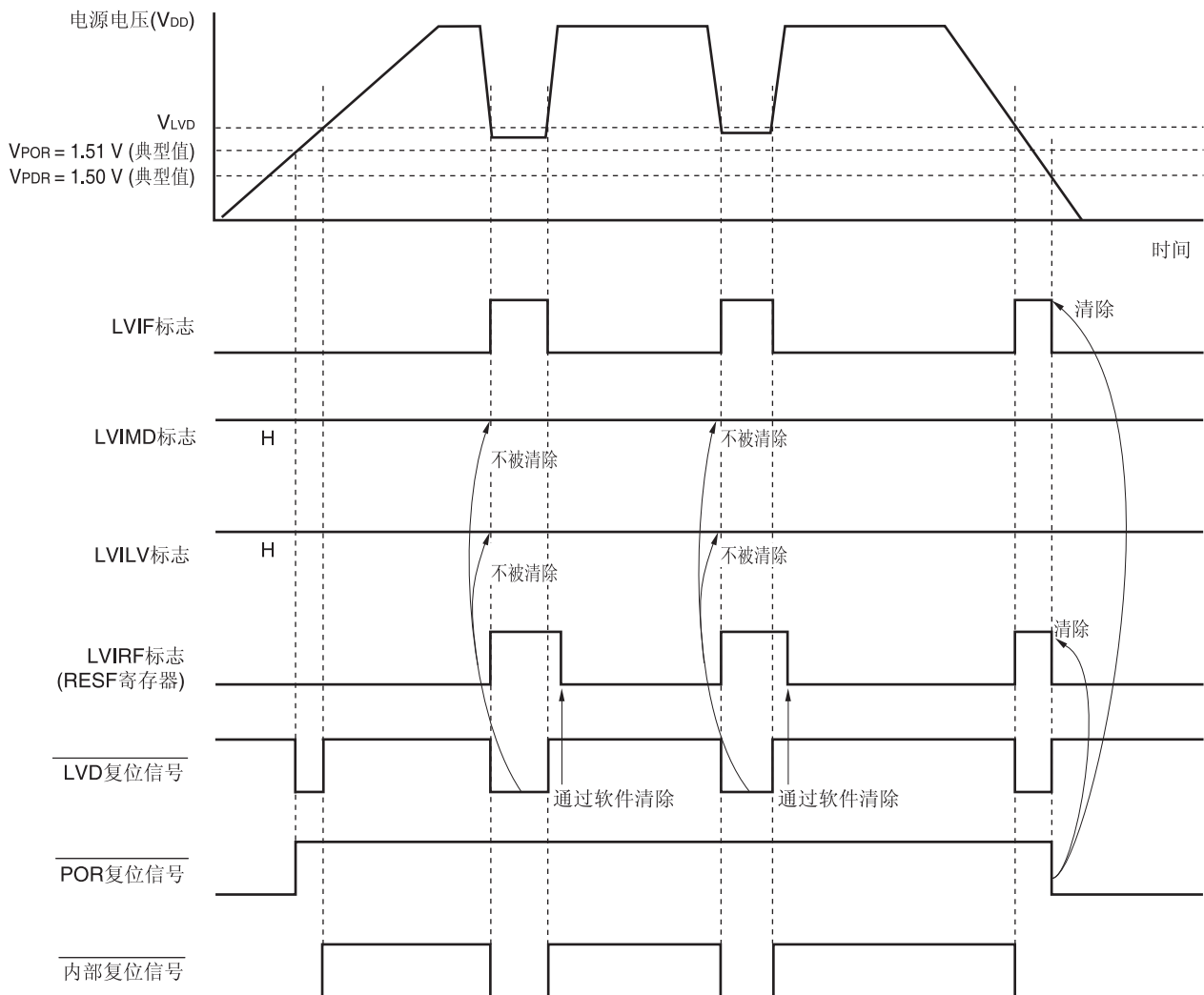
在如下的初始设置的状态下开始动作。

可使用选项字节 000C1H 设置操作模式 (复位模式 (LVIMDS1, LVIMDS0 = 1, 1)) 和检测电压 (VLVD)。

- 电压检测寄存器 (LVIM) 的位 7 (LVISEN) 被清除为 0 (禁止电压检测电平寄存器 (LVIS) 的改写)。
- 当选项字节 LVIMDS1 和 LVIMDS0 被设为 1 时, LVIS 寄存器的初始值被设为 81H。
 - 位 7 (LVIMD) 为 1 (复位模式)。
 - 位 0 (LVILV) 为 1 (低电压检测电平: VLVDL 或 VLVD)。

电压检测电路产生内部复位信号的时序如图 24-4 所示。

图 24-4. 电压检测电路产生内部复位信号的时序
(选项字节 LVIMDS1, LVIMDS0 = 1, 1)



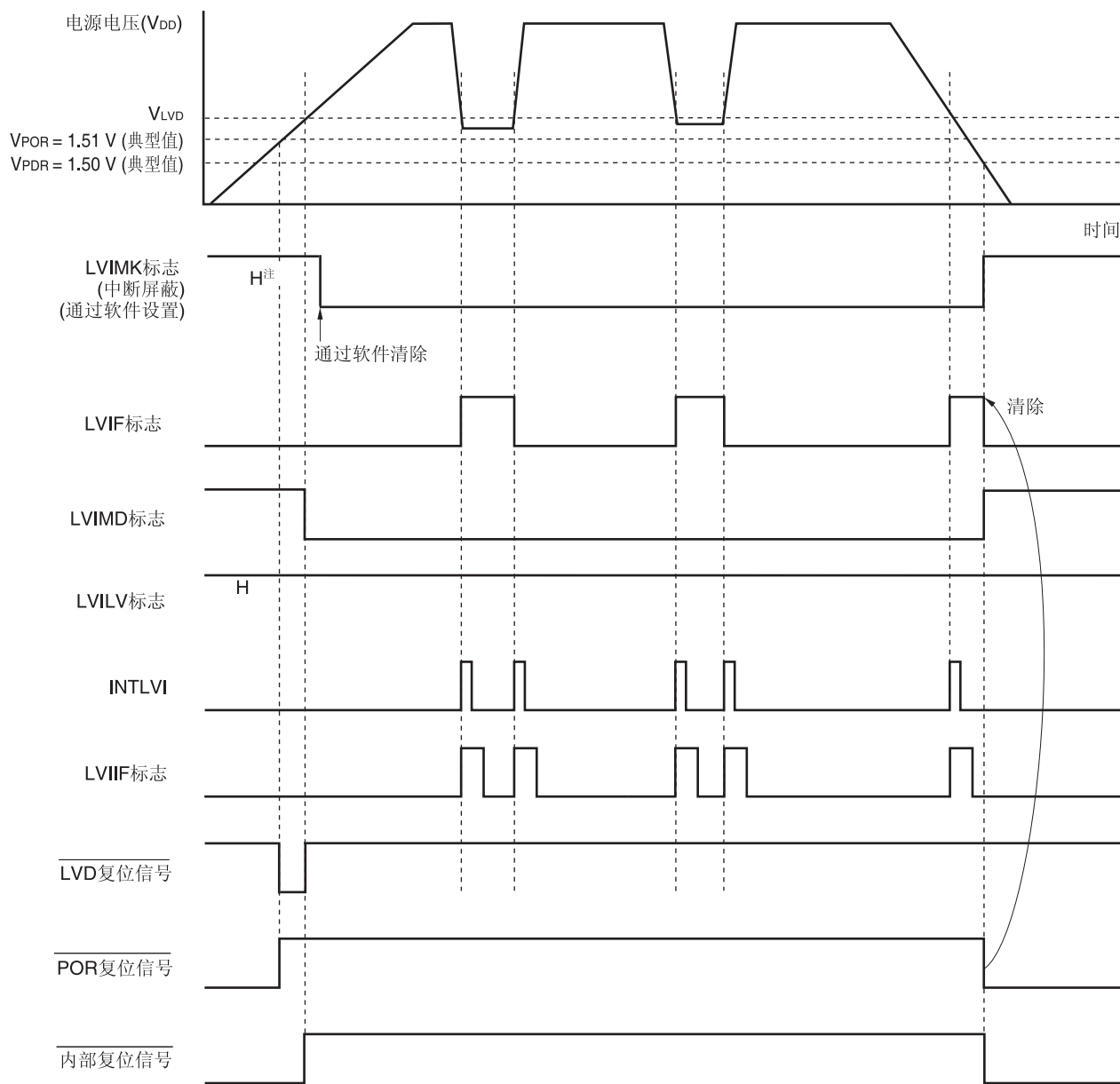
备注 V_{POR} : POR 电源上升检测电压
 V_{PDR} : POR 电源下降检测电压

24.4.2 使用中断模式时

- 开始操作时
可使用选项字节 000C1H/010C1H 设置操作模式 (中断模式 (LVIMDS1, LVIMDS0 = 0, 1)) 和检测电压 (VLvi)。在下列的初始设置的状态下开始动作。
 - 电压检测寄存器 (LVIM)的位 7 (LVISEN) 被清除为 0 (禁止电压检测电平寄存器 (LVIS)的改写)。
 - 当选项字节 LVIMDS1 清除为 0 且 LVIMDS0 被设为 1 时, LVIS 寄存器的初始值被设为 01H。
 - 位 7 (LVIMD)为 0 (中断模式)。
 - 位 0 (LVILV) 为 1 (低电压检测电平: VLvi)。

电压检测电路产生内部中断信号的时序如图 24-5 所示。

图 24-5. 电压检测电路产生内部中断信号的时序
(选项字节 LVIMDS1, LVIMDS0 = 0, 1)



注 通过产生复位信号，LVIMK 标志被设置为 1。

备注 V_{POR}: POR 电源上升检测电压
V_{PDR}: POR 电源下降检测电压

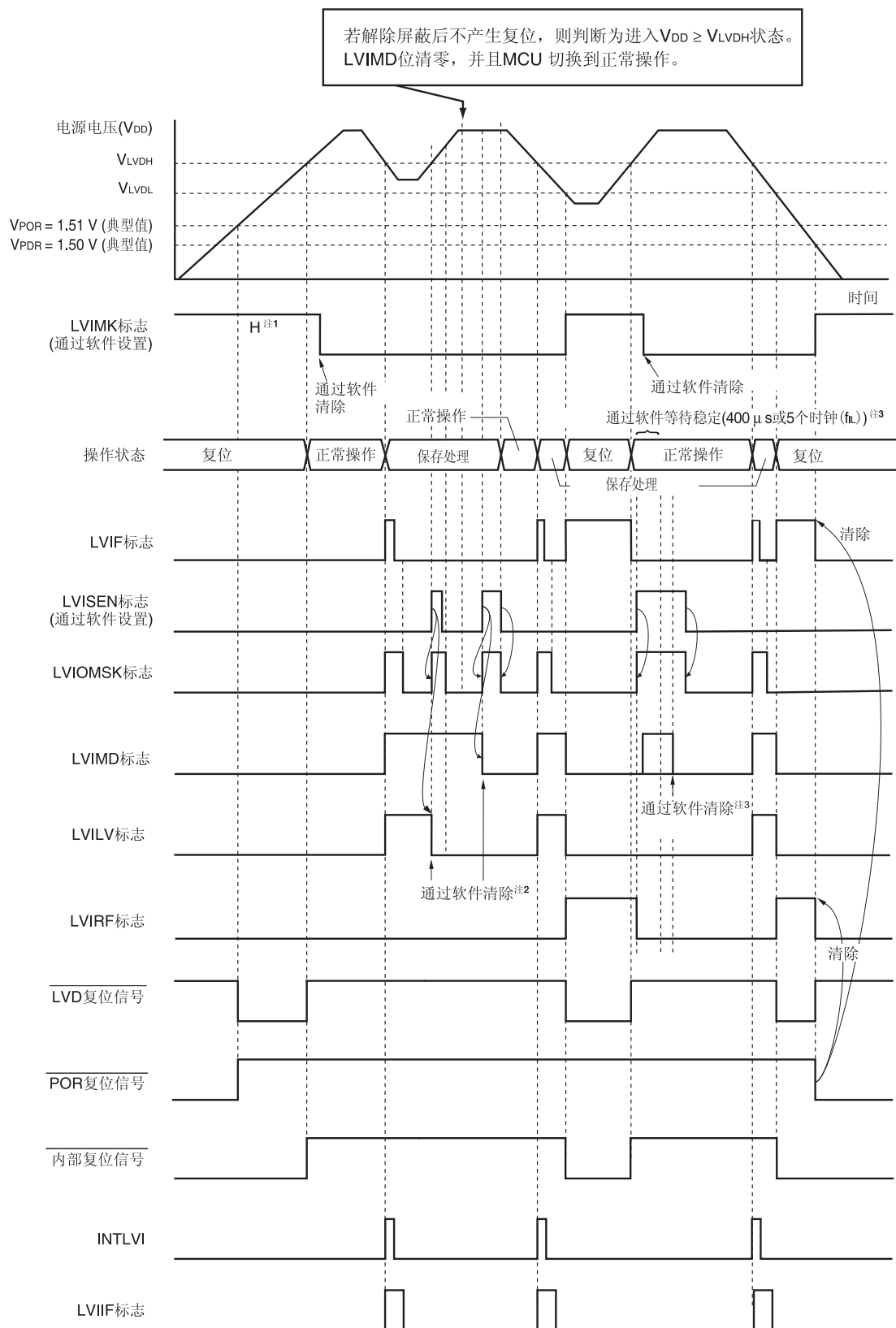
24.4.3 使用中断&复位模式时

- 开始操作时
可使用选项字节 000C1H 设置操作模式 (中断&模式 (LVIMDS1, LVIMDS0 = 1, 0)) 和检测电压 (VLVDH, VLVDL)。在下列的初始设置的状态下开始动作。
 - 电压检测寄存器 (LVIM)的位 7 (LVISEN) 被清除为 0 (禁止电压检测电平寄存器 (LVIS)的改写)。
 - 当选项字节 LVIMDS1 被设置为 1 且 LVIMDS0 被清除为 0 时, LVIS 寄存器的初始值被清除为 00H。
 - 位 7 (LVIMD)为 0 (中断模式)。
 - 位 0 (LVILV)为 0 (高电压检测电平: VLVDH)。

电压检测电路内部复位信号和中断信号产生的时序如图 24-6 所示。

请按照图 24-7. 发生中断后的处理步骤, 图 24-8. 中断&复位模式的初始设置所示的流程图的步骤执行。

图 24-6. 电压检测电路的发生内部复位信号和中断信号的时序
(选项字节 LVIMDS1, LVIMDS0 = 1, 0) (1/2)

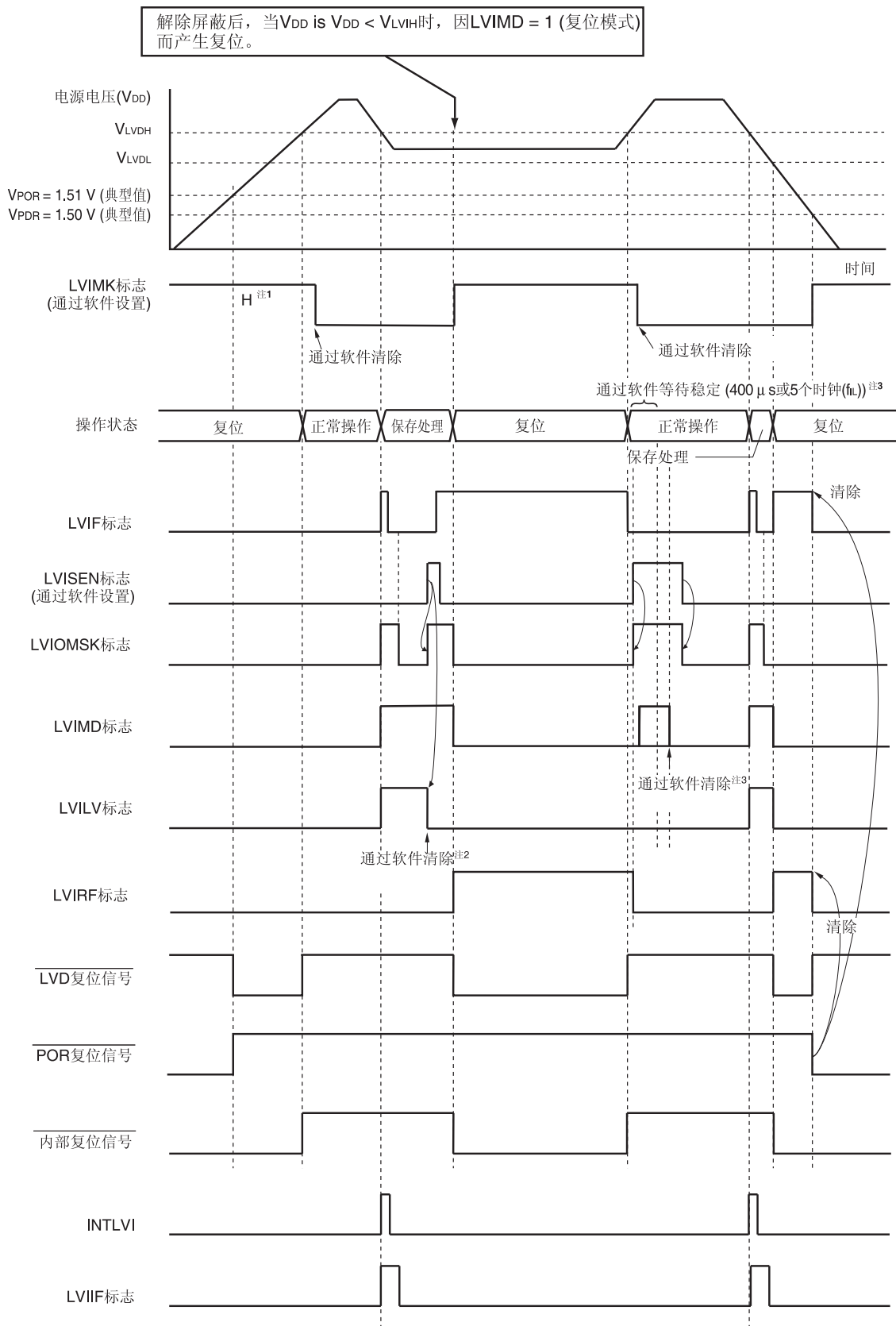


(注和备注如下页所示。)

- 注 1. 通过产生复位信号，LVIMK 标志被设置为 1。
2. 产生中断后，在中断&复位模式下根据图 24-7 发生中断后的处理步骤进行处理。
 3. 解除复位后，在中断&复位模式下根据图 24-8. 中断&复位模式的初始设置进行处理。

备注 VPOR: POR 电源上升检测电压
VPDR: POR 电源下降检测电压

图 24-6. 电压检测电路内部复位信号和中断信号产生的时序
(选项字节 LVIMDS1, LVIMDS0 = 1, 0) (2/2)

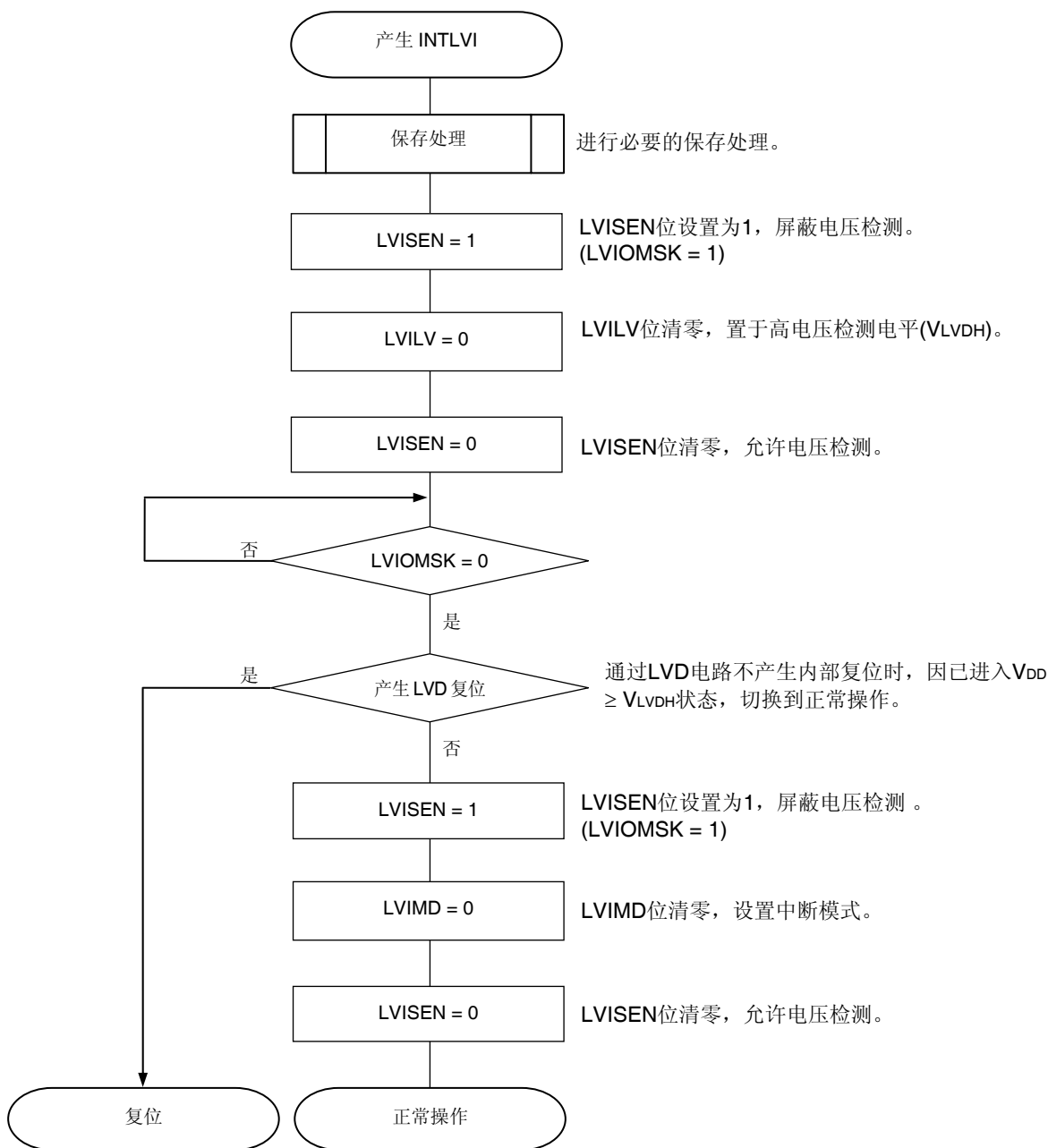


(注和备注如下页所示。)

- 注 1. 通过产生复位信号，LVIMK 标志被设置为 1。
- 2. 产生中断后，在中断&复位模式下根据图 24-7 发生中断后的处理步骤进行处理。
- 3. 解除复位后，在中断&复位模式下根据图 24-8. 中断&复位模式的初始设置进行处理。

备注 VPOR: POR 电源上升检测电压
 VPDR: POR 电源下降检测电压

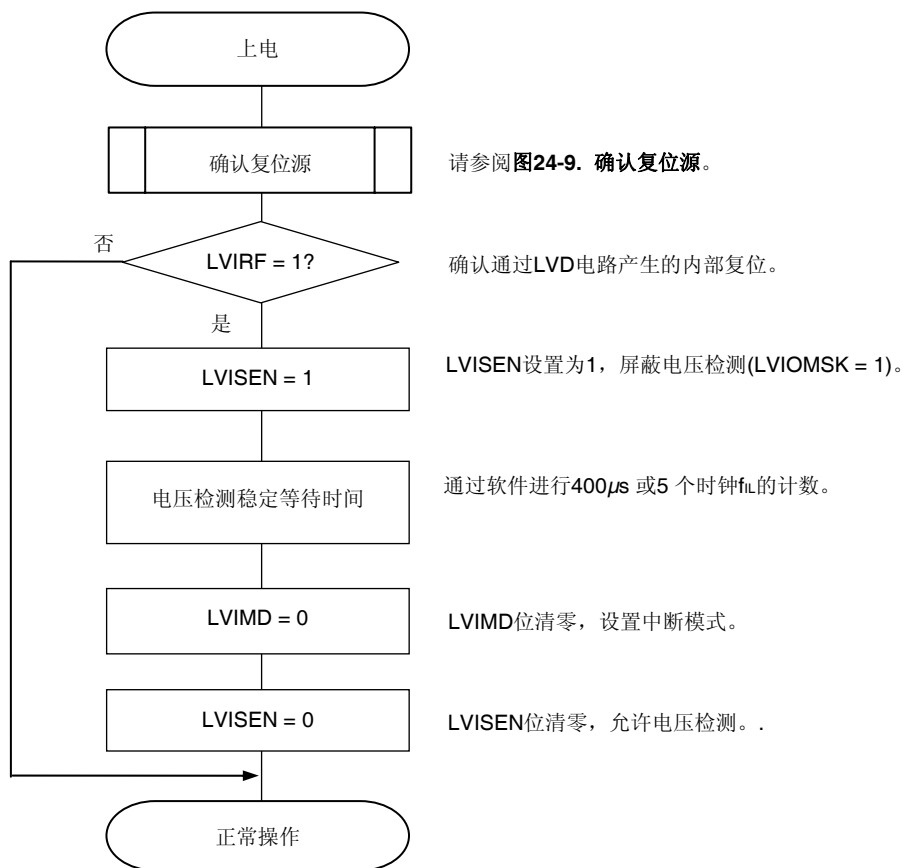
图 24-7. 发生中断后的处理步骤



当设置中断&复位模式(LVIMDS1, LVIMDS0 = 1, 0)时, 在解除 LVD 复位之后 (LVIRF = 1), 须有 400 μ s 或 5 个 f_{L} 时钟的电压检测稳定等待时间。等到电压检测稳定之后, 清除 LVIMD 位为 0, 使其初始化。当对电压检测稳定等待时间进行计数且改写 LVIMD 位之后, 将 LVISEN 设置为 1, 以屏蔽 LVD 引起的复位或中断产生。

中断&复位模式的初始设置的步骤如图 24-8 所示。

图 24-8. 中断和复位模式的初始设置



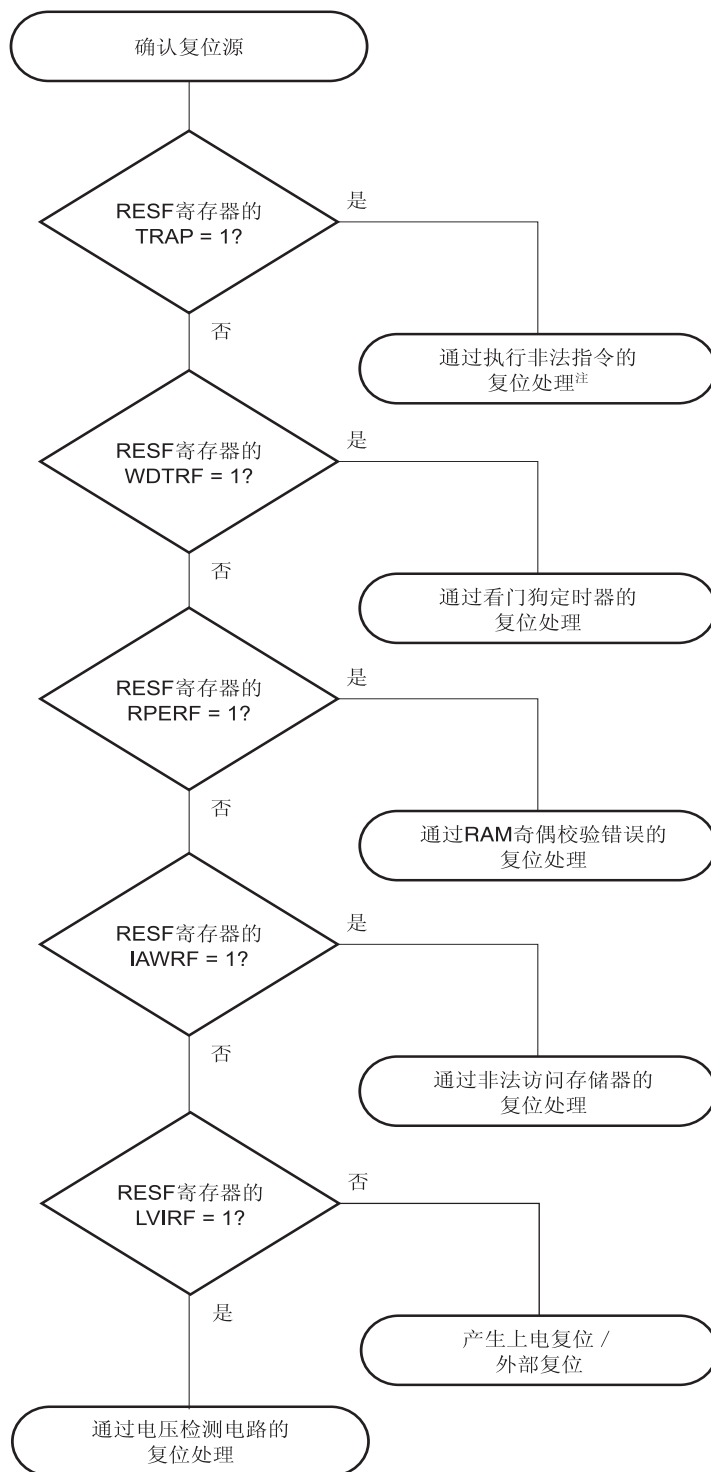
备注 f_L : 低速片上振荡器时钟频率

24.5 电压检测电路的注意事项

(1) 确认复位源

当发生复位时，通过以下方法确认复位源。

图 24-9. 确认复位源

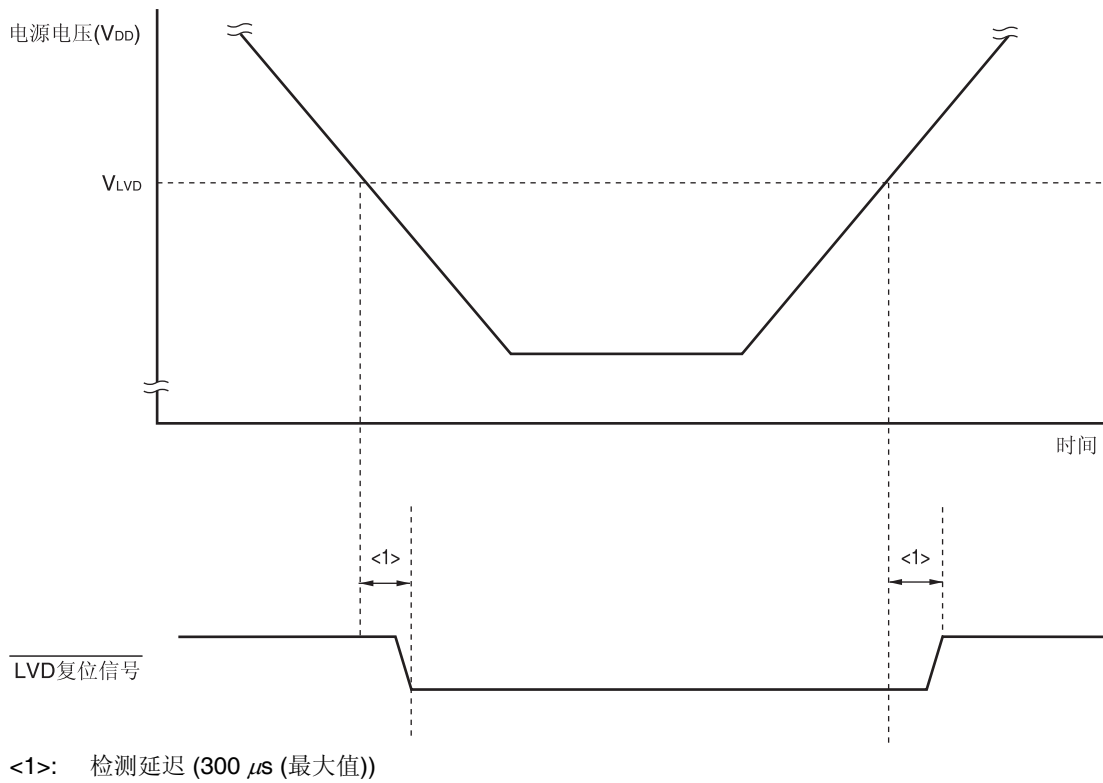


注 当执行指令代码 FFH 时产生。
通过电路内置仿真器或片上调试仿真器进行仿真时，不会因执行非法指令发生内部复位。

(2) 从产生 LVD 复位源到产生或解除 LVD 复位的延迟

从电源电压 (V_{DD}) < LVD 检测电压 (V_{LVD}) 到产生 LVD 复位的时间内有一定的延迟。

同理，从 LVD 检测电压 (V_{LVD}) \leq 电源电压 (V_{DD}) 到解除 LVD 复位的时间内也有一定的延迟（参阅图 24-10）。

图 24-10. 从产生 LVD 复位源到产生或解除 LVD 复位的延迟

第二十五章 安全功能

25.1 安全功能的概述

RL78/I1A 内置以下安全功能，符合 IEC60730 和 IEC61508 安全标准。
借助这些功能，单片机可以自行诊断异常问题，并在检测到异常时停止工作。

(1) 闪存 CRC 运算功能(高速 CRC、通用 CRC)

通过执行 CRC 运算检测闪存中的数据错误。

RL78/I1A 提供了两个 CRC 功能，可根据应用或用途加以使用。

- 高速 CRC：在初始化程序执行过程中，可以停止 CPU，并在整个代码闪存区域执行高速检验。
- 通用 CRC：在 CPU 运行期间，可以利用该功能检验包括代码闪存区域在内的各种数据。

(2) RAM 奇偶校验错误检测功能

当读取 RAM 数据时，可以检测奇偶校验错误。

(3) RAM 保护功能

防止当 CPU 失控时 RAM 数据被改写。

(4) SFR 保护功能

防止当 CPU 失控时 SFR 被改写。

(5) 非法存储器存取检测功能

检测对非法存储器区域(存储器不存在的区域,存取受限的区域)进行非法存取的功能。

(6) 频率检测功能

使用 TAU 可以检测振荡频率。

(7) A/D 测试功能

通过对内部基准电压执行 A/D 转换，从而对 A/D 转换执行自检。

备注 有关 IEC60730 标准所要求的功能，请参阅应用笔记（R01AN0749）。

25.2 安全功能使用的寄存器

安全功能的各种功能使用如下的寄存器。

寄存器	安全功能的各种功能
<ul style="list-style-type: none"> 闪存CRC控制寄存器(CRC0CTL) 闪存CRC运算结果寄存器(PGCRCL) 	闪存CRC运算功能 (高速CRC)
<ul style="list-style-type: none"> CRC输入寄存器(CRCIN) CRC数据寄存器(CRCD) 	CRC运算功能 (通用CRC)
<ul style="list-style-type: none"> RAM奇偶校验错误控制寄存器(RPECTL) 	RAM奇偶校验错误检测功能
<ul style="list-style-type: none"> 非法存储器存取检测控制寄存器(IAWCTL) 	RAM保护功能
	SFR保护功能
	非法存储器存取检测功能
<ul style="list-style-type: none"> 定时器输入选择寄存器0 (TIS0) 	频率检测功能
<ul style="list-style-type: none"> A/D测试寄存器(ADTES) 	A/D测试功能

各个寄存器的内容，在 **25.3 安全功能的操作** 中说明。

25.3 安全功能的操作

25.3.1 闪存CRC运算功能(高速CRC)

IEC60730 标准要求对闪存中的数据进行检验，并建议利用 CRC 来实现。RL78/I1A 提供的高速 CRC 可用于在初始化程序执行过程中，检验整个代码闪存区域。仅限通过 RAM 上的程序运行主系统时钟的 HALT 模式时，才可执行高速 CRC。

高速 CRC 是在停止 CPU 的状态下每个时钟从闪存读取 32 位数据来执行运算。因此，该功能可以在较短的时间内完成一次检查(例如，64 KB 闪存时 512 μ s@32 MHz)。

所用的生成 CRC 的多项式符合 CRC-16-CCITT 的“ $X^{16} + X^{12} + X^5 + 1$ ”公式。

高速 CRC 从位 31 到位 0 执行运算(MSB 格式)。

注意事项 片上调试时，配置有监视程序，因此会出现不同的 CRC 运算结果。

备注 高速 CRC 与通用 CRC 的运算结果是不一样的，因为通用 CRC 是从 LSB 开始执行的。

<控制寄存器>

(1) 闪存 CRC 控制寄存器(CRC0CTL)

该寄存器用于控制高速 CRC ALU 的操作，同时用于指定操作范围。

使用 1 位或 8 位存储器操作指令设置 CRC0CTL 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 25-1. 闪存 CRC 控制寄存器(CRC0CTL)的格式

地址: F02F0H 复位后: 00H R/W

符号	<7>	6	5	4	3	2	1	0
CRC0CTL	CRC0EN	0	FEA5	FEA4	FEA3	FEA2	FEA1	FEA0
CRC0EN	CRC ALU 的操作控制							
0	停止操作。							
1	根据 HALT 命令的执行开始运算。							
FEA5	FEA4	FEA3	FEA2	FEA1	FEA0	高速 CRC 运算范围		
0	0	0	0	0	0	0000H 至 3FFBH (16 K 减去 4 字节)		
0	0	0	0	0	1	0000H 至 7FFBH (32 K 减去 4 字节)		
0	0	0	0	1	0	0000H 至 BFFBH (48 K 减去 4 字节)		
0	0	0	0	1	1	0000H 至 FFFBH (64 K 减去 4 字节)		
其它						禁止设置		

备注 将用于比较的预期 CRC 运算结果值输入闪存的最低 4 个字节。注意，运算范围将因此减少 4 个字节。

(2) 闪存 CRC 运算结果寄存器(PGCRCL)

该寄存器用来存储高速 CRC 运算结果。

使用 16 位存储器操作指令设置 PGCRCL 寄存器。

产生复位信号后，该寄存器被清除为 0000H。

图 25-2. 闪存 CRC 运算结果寄存器(PGCRCL)的格式

地址: F02F2H 复位后: 0000H R/W

符号	15	14	13	12	11	10	9	8
PGCRCL	PGCRC15	PGCRC14	PGCRC13	PGCRC12	PGCRC11	PGCRC10	PGCRC9	PGCRC8
	7	6	5	4	3	2	1	0
	PGCRC7	PGCRC6	PGCRC5	PGCRC4	PGCRC3	PGCRC2	PGCRC1	PGCRC0
PGCRC15 至 0	高速 CRC 运算结果							
0000H 至 FFFFH	存储高速 CRC 运算结果							

注意事项 当 CRC0EN (CRC0CTL 寄存器的位 7) = 1 时，才可对 PGCRCL 寄存器执行写入操作。

闪存 CRC 运算功能(高速 CRC)的流程图如图 25-3 所示。

<操作流程>

图 25-3. 闪存 CRC 运算功能(高速 CRC)的流程图



- 注意事项
1. CRC 运算对象仅针对代码闪存。
 2. 将预期 CRC 运算值存储于代码闪存内运算范围以后的区域。
 3. 在执行 CRC 运算时，不会执行引导交换。
 4. 通过在 RAM 区域执行 HALT 指令，CRC 运算才能变为有效。
必须在 RAM 区域执行 HALT 指令。

预期 CRC 值可以利用 CubeSuite+ 开发环境等的工具来进行计算。（有关详情，请参阅 CubeSuite+ 用户手册。）

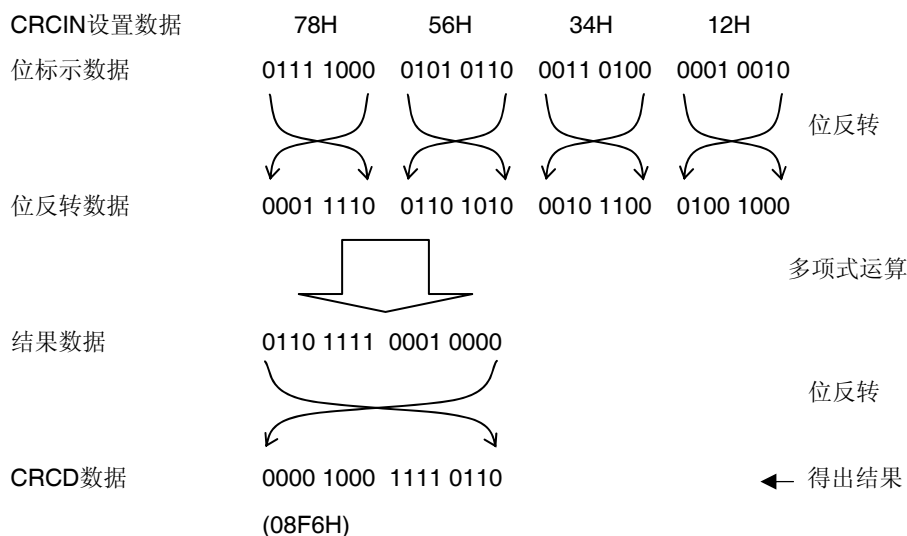
25.3.2 CRC运算功能(通用CRC)

为了保证操作过程中的安全，即使在 CPU 运行时根据 IEC61508 标准也要求检验数据。

在 RL78/I1A 中，可以在 CPU 运行时，把通用 CRC 运算当作外围功能来操作。通用 CRC 可用于检验代码闪存区域以外的各种数据。待检验的数据可通过软件（用户创建的程序）来指定。HALT 模式的 CRC 计算功能仅在 DMA 传送期间使用。

通用 CRC 运算可以在主系统时钟操作模式或副系统时钟操作模式下执行。

所用的生成 CRC 的多项式符合 CRC-16-CCITT 的“ $X^{16} + X^{12} + X^5 + 1$ ”公式。待输入的数据按位反转，然后进行计算以支持 LSB 优先的通信。例如，如果从 LSB 发送数据 12345678H，则写入 CRCIN 寄存器的值的顺序为 78H、56H、34H 和 12H，可从 CRCD 寄存器获取值 08F6H。这是针对将数据 12345678H 的位的排序反转后的位行（如下所示）执行 CRC 运算所获得的结果。



注意事项 执行程序时，调试器将设置软件断点的行改写为 BREAK 指令，因此在 CRC 运算目标区域中如果设置软件 BREAK 的话，将出现不同的 CRC 运算结果。

<控制寄存器>

(1) CRC 输入寄存器(CRCIN)

CRCIN 寄存器是设置通用 CRC 的 CRC 计算数据的 8 位寄存器。

可设置的范围为 00H 至 FFH。

使用 8 位存储器操作指令来设置 CRCIN 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 25-4. CRC 输入寄存器(CRCIN)的格式

地址: FFFACH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
CRCIN								
	位 7 至 0				功能			
	00H 至 FFH				数据输入			

(2) CRC 数据寄存器(CRCD)

该寄存器用来存储通用 CRC 运算结果。

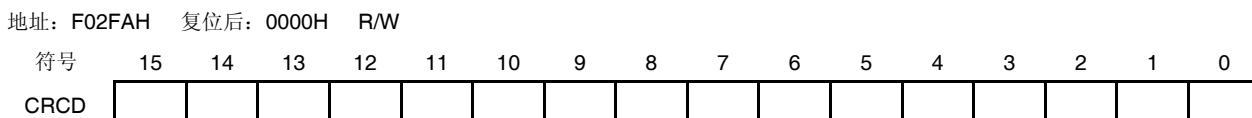
可设置的范围为 0000H 至 FFFFH。

从对 CRCIN 寄存器执行写入操作的时间开始经过 1 个 CPU/外围硬件时钟 (fCLK)的时钟周期之后，CRC 运算结果存储到 CRCD 寄存器中。

使用 16 位存储器操作指令设置 CRCD 寄存器。

产生复位信号后，该寄存器被清除为 0000H。

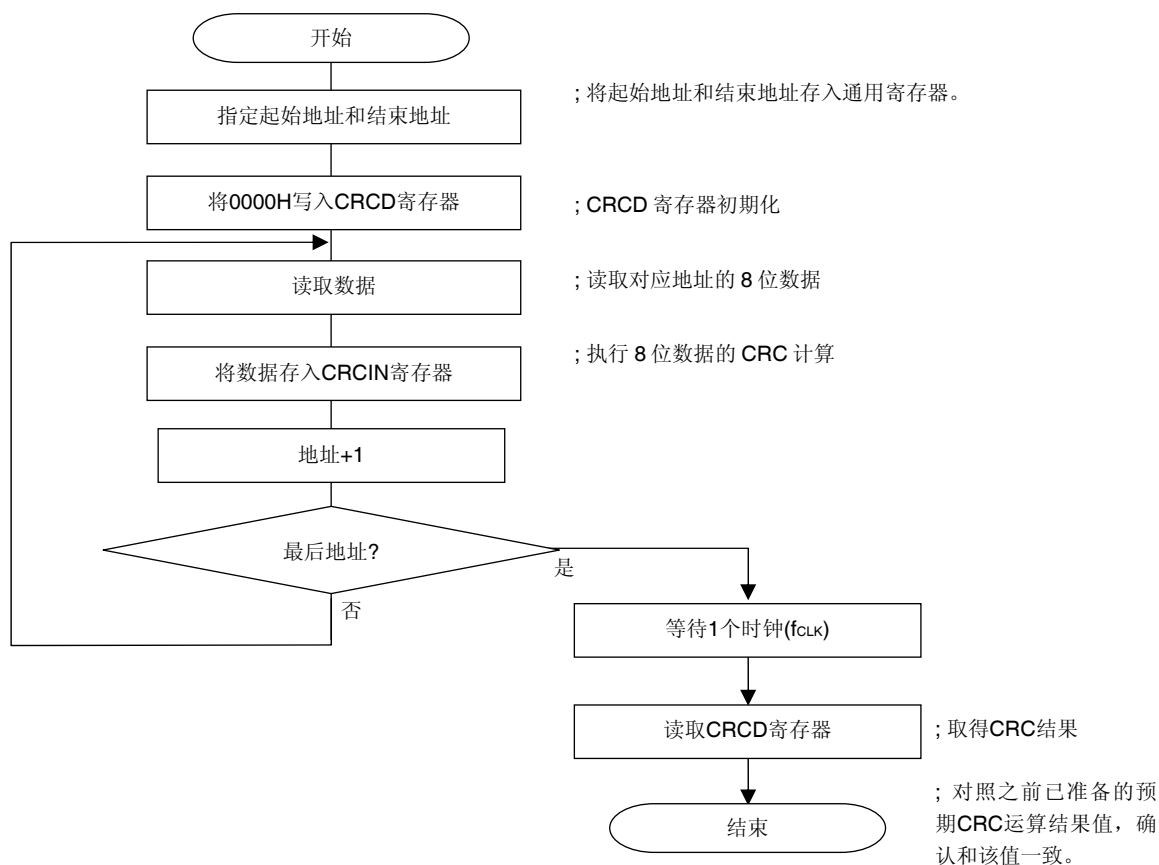
图 25-5. CRC 数据寄存器(CRCD)的格式



- 注意事项
1. 如需读出已写入 CRCD 寄存器的数值，请在写入 CRCIN 寄存器前执行读出操作。
 2. 如果向 CRCD 寄存器进行写入和存储运算结果发生冲突，则写入操作将被忽略。

<操作流程>

图 25-6. CRC 运算功能(通用 CRC)的流程图



25.3.3 RAM奇偶校验错误检测功能

根据 IEC60730 标准要求对 RAM 数据进行检验。因而，对 RL78/I1A 的 RAM 中的所有 8 位数据，都会添加由一个位构成的奇偶校验位。使用这个 RAM 奇偶校验错误检测功能，该奇偶校验位在写入数据时将被附加上去，并在读取数据时对该奇偶校验位进行检验。这个功能可以在发生奇偶校验错误时产生复位。

<控制寄存器>

• RAM 奇偶校验错误控制寄存器(RPECTL)

该寄存器用来控制奇偶校验错误的发生检查位和因奇偶校验错误而导致的发生复位。

使用 1 位或 8 位存储器操作指令设置 RPECTL 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 25-7. RAM 奇偶校验错误控制寄存器(RPECTL)的格式

地址: F00F5H 复位后: 00H R/W

符号	<7>	6	5	4	3	2	1	<0>
RPECTL	RPERDIS	0	0	0	0	0	0	RPEF
RPERDIS	奇偶校验错误复位屏蔽标志							
0	允许奇偶校验错误产生复位							
1	禁止奇偶校验错误产生复位							
RPEF	奇偶校验错误状态标志							
0	没有发生奇偶校验错误							
1	已发生奇偶校验错误							

注意事项 由于 RL78 执行流水操作，CPU 会执行预取，所以 CPU 可能读取分配至所用 RAM 之后的未初始化的 RAM 区域，以至于产生 RAM 奇偶校验错误。因此，允许产生 RAM 奇偶校验错误复位 (RPERDIS = 0) 时，必须对所用 RAM 区域 + 10 字节的区域进行初始化。当允许 RAM 奇偶校验错误复位 (RPERDIS = 0)，且使用自编程功能时，必须在改写前对“待改写的 RAM 区域 + 10 字节”的区域进行初始化。此外，通过 RAM 的指令读取的 RAM 数据受奇偶校验错误检测的影响。

- 备注**
- RAM 奇偶校验错误检查一直为允许状态，并且可通过 RPEF 标志来确认其结果。
 - 初始状态下允许奇偶校验错误产生复位 (RPERDIS = 0)。即使设置为禁止奇偶校验错误产生复位 (RPERDIS = 1)，发生奇偶校验错误时 RPEF 标志会被设为 1。
 - 因 RAM 奇偶校验错误 RPEF 标志被设为 1，可以通过写入 0 或任何复位源清零 (0)。当 RPEF = 1 时，即使读取的是未发生奇偶校验错误的 RAM，也会保持 RPEF=1 的状态。

25.3.4 RAM保护功能

为了保证操作过程中的安全，即使在 CPU 停止运行时根据 IEC61508 标准也要求保护存储于 RAM 的重要数据。

这个 RAM 保护功能用来保护指定的 RAM 空间的数据。

若设为 RAM 保护功能，则写入指定的 RAM 空间变为无效，但可以照常从该空间读取数据。

<控制寄存器>

● 非法存储器存取检测控制寄存器(IAWCTL)

该寄存器用来控制非法存储器存取的检测和 RAM/SFR 保护功能。

RAM 保护功能使用 GRAM1 和 GRAM0 位。

使用 8 位存储器操作指令来设置 IAWCTL 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 25-8. 非法存储器存取检测控制寄存器(IAWCTL)的格式

地址: F0078H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
IAWCTL	IAWEN	0	GRAM1	GRAM0	0	GPORT	GINT	GCSC

GRAM1	GRAM0	RAM 保护空间 ^注
0	0	无效。可以写入 RAM。
0	1	从低 RAM 地址开始的 128 字节
1	0	从低 RAM 地址开始的 256 字节
1	1	从低 RAM 地址开始的 512 字节

注 RAM 的起始地址因所用产品内置的 RAM 大小而异。

25.3.5 SFR保护功能

为了保证操作过程中的安全，即使在 CPU 停止运行时根据 IEC61508 标准也要求保护重要的 SFR，使其免遭改写。

SFR 保护功能用来保护端口功能，中断功能，时钟控制功能，电压检测电路和 RAM 奇偶校验错误检测功能的控制寄存器的数据。

若设为 SFR 保护功能，则写入被保护的 SFR 变为无效，但可以照常进行读出。

<控制寄存器>

• 非法存储器存取检测控制寄存器 0, 1 (IAWCTL0, IAWCTL1)

该寄存器用来控制非法存储器存取的检测和 RAM/SFR 保护功能。

SFR 保护功能使用 GPORT、GINT 和 GCSC 位。

使用 8 位存储器操作指令来设置 IAWCTL0 和 IAWCTL1 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 25-9. 非法存储器存取检测控制寄存器 0 (IAWCTL0)的格式

地址: F0078H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
IAWCTL0	IAWEN	0	GRAM1	GRAM0	0	GPORT	GINT	GCSC

GPORT	端口功能的控制寄存器的保护
0	无效。可读出或写入端口功能的控制寄存器。
1	有效。端口功能的控制寄存器的写入为无效。读出为有效。 [被保护的 SFR] PM0 至 PM3, PM4, PM7, PM12, PM14, PU0, PU1, PU3, PU4, PU7, PU12, PU14, PIM0, PIM1, POM0, POM1, PMC0, PMC12, PMC14, ADPC 和 PIOR1 ^{注1}

GINT	中断功能的寄存器的保护
0	无效。可读出或写入中断功能的寄存器。
1	有效。中断功能的寄存器的写入为无效。读出为有效。 [被保护的 SFR] IF0 至 IF2, MK0 至 MK2, PR00, PR01, PR02, PR10, PR11, PR12, EGP0, EGP1, EGN0 和 EGN1

GCSC ^{注2}	时钟控制功能，电压检测电路和 RAM 奇偶校验错误检测功能的控制寄存器的保护
0	无效。可读出或写入时钟控制功能，电压检测电路和 RAM 奇偶校验错误检测功能的控制寄存器。
1	有效。时钟控制功能，电压检测电路和 RAM 奇偶校验错误检测功能的控制寄存器为无效。读出为有效。 [被保护的 SFR] CMC, CSC, OSTs, CKC, PER0, OSMC, LVIM, LVIS 和 RPECTL

注 1. Pxx (端口寄存器)不被保护。

2. 在自编程/串行编程时，请清除 GCSC 位为 0。

图 25-10. 非法存储器存取检测控制寄存器 1 (IAWCTL1)的格式

地址: F05C4H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
IAWCTL1	0	0	0	0	0	GDPORT1	GDINT1	GDCG1

GDPORT1	端口功能的控制寄存器的保护
0	无效。可读出或写入端口功能的控制寄存器。
1	有效。写入端口功能的控制寄存器为无效。读出为有效。 [被保护的 SFR] PM20, PU20, POM20, PIOR1 和 SUCTL ^{注1}

GDINT1	中断功能的寄存器的保护
0	无效。可读出或写入中断功能的寄存器。
1	有效。写入中断功能的寄存器为无效。读出为有效。 [被保护的 SFR] INTMK0, EGP2, EGN2, CMPEGP0 和 CMPEGN0

GDCG1	时钟控制功能的控制寄存器的保护
0	无效。可读出或写入时钟控制功能, 电压检测电路和 RAM 奇偶校验错误检测功能的控制寄存器。
1	有效。写入时钟控制功能, 电压检测电路和 RAM 奇偶校验错误检测功能的控制寄存器为无效。读出为有效。 [被保护的 SFR] PER1, PER2 和 PLLCTL

注 Pxx (端口寄存器)不被保护。

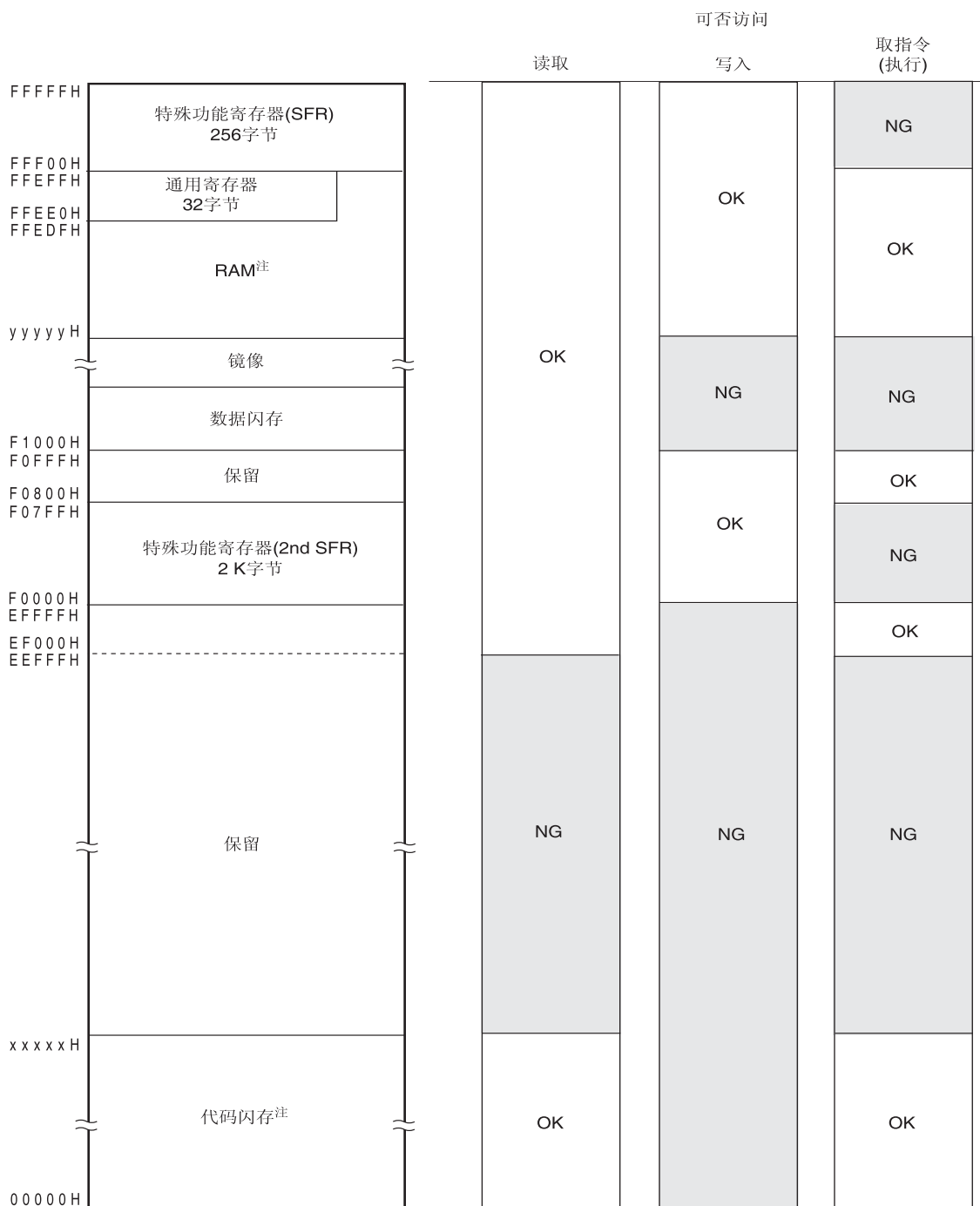
25.3.6 非法存储器存取检测功能

IEC60730 标准要求检验 CPU 和中断均工作正常。

如果对指定为禁止存取的存储器空间进行存取，则非法存储器存取检测功能将产生复位。

非法存储器存取检测功能适用于图 25-11 中表示为 NG 的区域。

图 25-11. 非法存取检测空间



注 各产品的代码闪存和 RAM 地址如下所示。

产品	代码闪存 (00000H至xxxxxH)	RAM (yyyyyH至FFEFFH)
R5F1076C, R5F107AC, R5F107BC	32768 × 8位(00000H-07FFFH)	2048 × 8位(FF700H-FFEFFH)
R5F107AE, R5F107DE	65536 × 8位(00000H-0FFFFH)	4096 × 8位(FEF00H-FFEFFH)

<控制寄存器>

• 非法存储器存取检测控制寄存器 0 (IAWCTL0)

该寄存器用来控制非法存储器存取的检测和 RAM/SFR 保护功能。

非法存储器存取检测功能使用 IAWEN 位。

使用 8 位存储器操作指令来设置 IAWCTL 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 25-12. 非法存储器存取检测控制寄存器 0 (IAWCTL0)的格式

地址: F0078H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
IAWCTL	IAWEN	0	GRAM1	GRAM0	0	GPORT	GINT	GCSC

IAWEN ^注	非法存储器存取的检测控制
0	非法存储器存取的检测无效
1	非法存储器存取的检测有效

注 只有对 IAWEN 位写 1 的时候才有效,对 IAWEN 写 1 以后再写 0 无效。

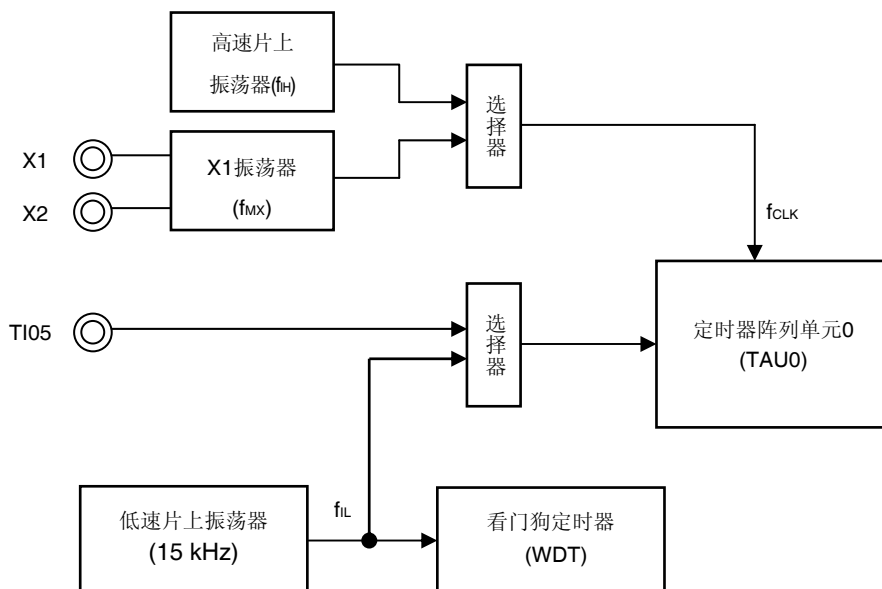
备注 若将选项字节指定为 WDTON = 1(看门狗定时器操作有效)，则即使 IAWEN = 0，非法存储器存取检测功能也将有效。

25.3.7 频率检测功能

根据 IEC60730 标准要求检验振荡频率正常。

频率检测功能可以检测时钟的工作频率是否异常，方法是通过将高速片上振荡器时钟或外接 X1 振荡器时钟与低速片上振荡器时钟 (15 kHz) 进行比较，能够检查出时钟的工作频率是否异常。

图 25-13. 频率检测功能的配置



<操作概述>

可以通过在以下条件下测量脉冲间隔来判断时钟频率是否正确：

- 将高速片上振荡器时钟 (f_H)或外接 X1 振荡时钟 (f_{MX})选为 CPU/外围硬件时钟 (f_{CLK})。
- 将低速片上振荡器时钟 (f_{IL} :15 kHz)选为定时器阵列单元 0 (TAU0) 通道 5 的定时器输入。

如果脉冲间隔测量得到异常值，则可得出结论认为，时钟频率异常。

若要了解脉冲间隔测量的方法，请参阅 6.7.3 作为输入脉冲间隔测量的操作。

<控制寄存器>

• 定时器输入选择寄存器 0 (TIS0)

该寄存器用于选择通道 5 的定时器输入。

通过将低速片上振荡器时钟选为定时器输入，则可测量其脉冲宽度，以确定低速片上振荡器时钟与定时器操作时钟之间的比例关系是否正确。

用 8 位存储器操作指令设置 TIS0 寄存器。

产生复位信号后，该寄存器被清除为 00H。

图 25-14. 定时器输入选择寄存器 0 (TIS0)的格式

地址: F0074H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
TIS0	0	0	0	0	0	TIS02	TIS01	TIS00

TIS02	TIS01	TIS00	选择通道5使用的定时器输入
0	0	0	定时器输入引脚 (TI05)的输入信号
0	0	1	
0	1	0	
0	1	1	
1	0	0	低速片上振荡器时钟(f _{IL})
1	0	1	副系统时钟 (f _{SUB})
其它			禁止设置

25.3.8 A/D测试功能

根据 IEC60730 标准要求对 A/D 转换器进行测试。A/D 测试功能用于检验 A/D 转换器是否工作正常，方法是对 0V 内部电压、AVREF 电压和内部基准电压(1.45 V) 进行 A/D 转换操作。

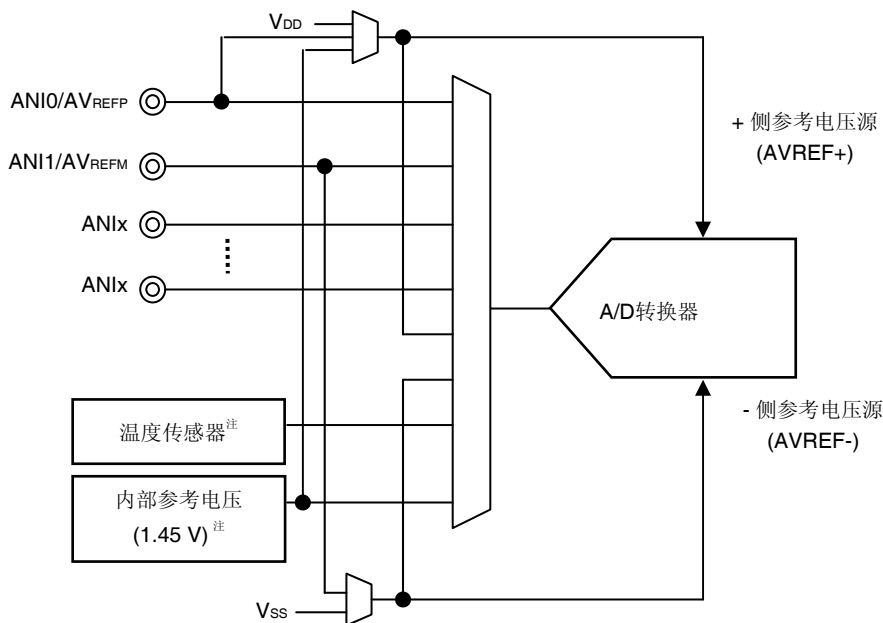
可以通过以下步骤来检查模拟多路转换器。

- (1) 对 ANIx 引脚进行 A/D 转换(转换结果 1)。
- (2) 使用 ADTES 寄存器选择 AVREFM，进行 A/D 转换，然后将 A/D 转换器的采样电容端子间的电压电位差设置为 0 V。
- (3) 对 ANIx 引脚执行 A/D 转换(转换结果 2)。
- (4) 使用 ADTES 寄存器选择 AVREFP，执行 A/D 转换，然后将 A/D 转换器的采样电容端子间的电压电位差设置为 AVREF。
- (5) 对 ANIx 引脚执行 A/D 转换(转换结果 3)。
- (6) 确保转换结果 1、2 和 3 为相同值。

通过以上步骤，可以确认已经选择模拟多路转换器以及所有引线已经连接好。

- 备注 1.** 在上述步骤<1>至<5>的 A/D 转换过程中，如果模拟输入电压为变化值，则采用别的方法来检查模拟多路转换器。
- 2.** 转换结果可能含有误差。对转换结果进行比较时，要考虑适当的误差。

图 25-15. A/D 测试功能的配置



注 此设置仅能在 HS (高速主) 模式下使用。

<控制寄存器>

(1) A/D 测试寄存器(ADTES)

该寄存器用于将 A/D 转换器的+端的基准电压 AV_{REFP} 、A/D 转换器的-端的基准电压 AV_{REFM} 或模拟输入通道(ANLxx)选为 A/D 转换的目标。

使用 A/D 测试功能时，指定以下的设置：

- 在转换内部 0 V 电压时，将 AV_{REFM} 选为 A/D 转换的目标。
- 在转换 AV_{REF} 时，将 AV_{REFP} 选为 A/D 转换的目标。

ADTES 寄存器可使用 8 位存储器操作指令进行设置。

产生复位信号后，该寄存器被清除为 00H。

图 25-16. A/D 测试寄存器(ADTES)的格式

地址：F0013H 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
ADTES	0	0	0	0	0	0	ADTES1	ADTES0

ADTES1	ADTES0	A/D 转换目标
0	0	ANLxx/温度传感器输出 ^注 /内部基准电压(1.45 V) ^注 (通过模拟输入通道指定寄存器(ADS)来指定。)
1	0	AV_{REFM}
1	1	AV_{REFP}
其它		禁止设置

注 温度传感器输出/内部基准电压 (1.45V) 仅限在 HS (高速主)模式下使用。

(2) 模拟输入通道指定寄存器(ADS)

该寄存器用于指定即将执行 A/D 转换的模拟电压的输入通道。

测量 ANI_{xx}/温度传感器输出/内部基准电压(1.45 V)时, 将 A/D 测试寄存器(ADTES)设置为 00H。

使用 1 位或 8 位存储器操作指令设置 ADS 寄存器。

产生复位信号后, 该寄存器被清除为 00H。

图 25-17. 模拟输入通道规格寄存器(ADS)的格式

地址: FFF31H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ADS	ADISS	0	0	ADS4	ADS3	ADS2	ADS1	ADS0

○ 选择模式 (ADMD = 0)

ADISS	ADS4	ADS3	ADS2	ADS1	ADS0	模拟输入通道	输入源
0	0	0	0	0	0	ANI0	P20/ANI0/AV _{REFP} 引脚
0	0	0	0	0	1	ANI1	P21/ANI1/AV _{REFM} 引脚
0	0	0	0	1	0	ANI2	P22/ANI2 引脚
0	0	0	0	1	1	PGAOUT	PGA 输出引脚
0	0	0	1	0	0	ANI4	P24/ANI4 引脚
0	0	0	1	0	1	ANI5	P25/ANI5 引脚
0	0	0	1	1	0	ANI6	P26/ANI6 引脚
0	0	0	1	1	1	ANI7	P27/ANI7 引脚
0	1	0	0	0	0	ANI16	P03/ANI16 引脚
0	1	0	0	0	1	ANI17	P02/ANI17 引脚
0	1	0	0	1	0	ANI18	P147/ANI18 引脚
0	1	0	0	1	1	ANI19	P120/ANI19 引脚
1	0	0	0	0	0	-	温度传感器的输出 ^注
1	0	0	0	0	1	-	内部基准电压输出(1.45 V) ^注
其它						禁止设置	

注 此设置仅能在 HS (高速主) 模式下使用。

注意事项 1. 必须将位 5 和位 6 清除为 0。

2. 仅在停止 A/D 电压比较器操作时 (即当 A/D 转换器模式寄存器 0(ADM0)的 ADCE 位为 0 时) 可以重写 ADISS 位的值。

3. 如果使用 AV_{REFP} 作为 A/D 转换器的+侧基准电压源, 则不得选择 ANI0 作为 A/D 转换通道。

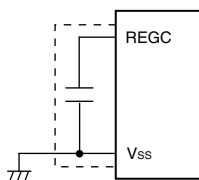
4. 如果使用 AV_{REFM} 作为 A/D 转换器的-侧基准电压源, 则不得选择 ANI1 作为 A/D 转换通道。

5. 如果将 ADISS 设置为 1, 则不得将内部基准电压 (1.45 V)用作+ 侧基准电压源。

第二十六章 稳压器

26.1 稳压器的概述

RL78/I1A 内置了一个可使器件内部以恒定电压操作的电路。此时，为了稳定稳压器的输出电压，须通过一个电容(0.47 至 1 μF)将 REGC 引脚连接至 Vss。并且，由于其作用是稳定内部电压，所以必须使用优质电容。



注意事项 请尽可能缩短上图中虚线部分的布线长度。

稳压器输出电压，请参阅表 26-1。

表 26-1. 稳压器输出电压条件

模式	输出电压	条件
LS (低速主)模式	1.8 V	-
HS (高速主)模式	1.8 V	STOP 模式时
		当高速系统时钟 (f _{MX}) 和高速片上振荡器时钟 (f _{IH}) 均在 CPU 以副系统时钟 (f _{XT}) 运行期间停止时
	2.1 V	当高速系统时钟 (f _{MX}) 和高速片上振荡器时钟 (f _{IH}) 均在 HALT 模式期间停止时 (已将 CPU 设为以副系统时钟 (f _{XT}) 运行)
		上述以外 (包括 OCD 模式时) ^注

注 当在片上调试期间，切换至副系统时钟运行模式或 STOP 模式时，稳压器输出电压保持于 2.1 V (不会下降至 1.8 V)。

第二十七章 选项字节

27.1 选项字节的功能

RL78/I1A 中的闪存的地址 000C0H 至 000C3H 为选项字节区域。

选项字节由用户选项字节 (000C0H 至 000C2H)和片上调试选项字节 (000C3H)构成。

上电或复位启动时，将自动按照选项字节为基准并设置指定的功能。在使用本产品时，必须利用选项字节来设置以下功能。

若要在自编程过程中使用引导交换操作时，由 010C0H 至 010C3H 与 000C0H 至 000C3H 相互切换。因此，须将 010C0H 至 010C3H 设置为与 000C0H 至 000C3H 相同的值。

27.1.1 用户选项字节 (000C0H至 000C2H/010C0H至 010C2H)

(1) 000C0H/010C0H

- 看门狗定时器的操作
 - HALT 或 STOP 模式时的操作停止或启用
- 看门狗定时器的溢出时间的设置
- 看门狗定时器的操作
 - 操作停止或启用。
- 看门狗定时器的窗口开启期间的设置
- 看门狗定时器的间隔中断的设置
 - 使用或不使用

注意事项 在使用引导交换操作时，由于 000C0H 与 010C0H 相互切换，须将 010C0H 设置为与 000C0H 相同的值。

(2) 000C1H/010C1H

- 设置 LVD 操作模式
 - 中断&复位模式
 - 复位模式
 - 中断模式
- 设置 LVD 检测水平 (VLVDH, VLVDL, VLVD)

注意事项 在使用引导交换操作时，由于 000C1H 与 010C1H 相互切换，须将 000C1H 设置为与 010C1H 相同的值。

(3) 000C2H/010C2H

- Flash 操作模式的设置
 - LS (低速主)模式
 - HS (高速主)模式
- 高速片上振荡器的频率的设置
 - 可从 1 MHz、4 MHz、8 MHz、12 MHz、16 MHz、24 MHz 和 32 MHz 中选择。

注意事项 使用引导交换操作时，由于 000C2H 与 010C2H 相互切换，须将 000C2H 设置为与 010C2H 相同的值。

27.1.2 片上调试选项字节 (000C3H/ 010C3H)

- 片上调试操作的控制
 - 禁止/允许片上调试操作
- 片上调试安全 ID 验证失败时闪存数据的处理
 - 在片上调试安全 ID 验证失败时闪存的数据将被擦除或不被擦除。

注意事项 在使用引导交换操作时，由于 000C3H 与 010C3H 相互切换，须将 000C3H 设置为与 010C3H 相同的值。

27.2 用户选项字节的格式

用户选项字节的格式如下所示。

图 27-1. 用户选项字节(000C0H/010C0H)的格式

地址: 000C0H/010C0H^{※1}

7	6	5	4	3	2	1	0
WDTINIT	WINDOW1	WINDOW0	WDTON	WDCS2	WDCS1	WDCS0	WDSTBYON
WDTINIT	看门狗定时器的间隔中断使用						
0	不使用间隔中断。						
1	当达到溢出时间的 75% + 1/2f _{IL} 时产生间隔中断。						
WINDOW1	WINDOW0	看门狗定时器的窗口开启期间 ^{※2}					
0	0	禁止设置					
0	1	50%					
1	0	75%					
1	1	100%					
WDTON	看门狗定时器的操作控制						
0	禁止计数器操作(复位后停止计数)						
1	允许计数器操作(复位后开始计数)						
WDCS2	WDCS1	WDCS0	看门狗定时器的溢出时间 (f _{IL} = 17.25 kHz (MAX.))				
0	0	0	2 ⁶ /f _{IL} (3.71 ms)				
0	0	1	2 ⁷ /f _{IL} (7.42 ms)				
0	1	0	2 ⁸ /f _{IL} (14.84 ms)				
0	1	1	2 ⁹ /f _{IL} (29.68 ms)				
1	0	0	2 ¹¹ /f _{IL} (118.72 ms)				
1	0	1	2 ¹³ /f _{IL} (474.90 ms)				
1	1	0	2 ¹⁴ /f _{IL} (949.80 ms)				
1	1	1	2 ¹⁶ /f _{IL} (3799.19m s)				
WDSTBYON	看门狗定时器的操作控制 (HALT/STOP 模式)						
0	HALT/STOP 模式时, 停止计数器操作 ^{※2}						
1	HALT/STOP 模式时, 允许计数器操作						

注 1. 在使用引导交换操作时, 由于 000C0H 与 010C0H 相互切换, 须将 010C0H 设置为与 000C0H 相同的值。

2. 当 WDSTBYON = 0 时, 窗口开启周期为 100%, 与 WINDOW1 和 WINDOW0 位的值无关。

注意事项 看门狗定时器在 EEPROM 仿真过程中也继续运行。处理过程中, 中断的受理时间将被延迟。在设置溢出时间和窗口尺寸时要考虑该延迟。

备注 f_{IL}: 低速片上振荡器时钟频率

图 27-2. 用户选项字节(000C1H/010C1H)的格式 (1/2)

地址: 000C1H/010C1H*

7	6	5	4	3	2	1	0
VPOC2	VPOC1	VPOC0	1	LVIS1	LVIS0	LVIMDS1	LVIMDS0

• LVD 设置(中断&复位模式时的设置)

检测电压			选项字节设置值						
VLVDH		VLVDL	模式设置		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0
上升沿	下降沿	下降沿	LVIMDS1	LVIMDS0					
2.92 V	2.86 V	2.75 V	1	0	0	1	1	1	0
3.02 V	2.96 V							0	1
4.06 V	3.98 V							0	0
其它			禁止设置						

• LVD 设置(复位模式时的设置)

检测电压		选项字节设置值						
VLVD		模式设置		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0
上升沿	下降沿	LVIMDS1	LVIMDS0					
2.81 V	2.75 V	1	1	0	1	1	1	1
2.92 V	2.86 V				1	1	1	0
3.02 V	2.96 V				1	1	0	1
3.13 V	3.06 V				0	1	0	0
3.75 V	3.67 V				1	0	0	0
4.06 V	3.98 V				1	1	0	0
其它		禁止设置						

• LVD 设置(中断模式时的设置)

检测电压		选项字节设置值						
VLVD		模式设置		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0
上升沿	下降沿	LVIMDS1	LVIMDS0					
2.81 V	2.75 V	0	1	0	1	1	1	1
2.92 V	2.86 V				1	1	1	0
3.02 V	2.96 V				1	1	0	1
3.13 V	3.06 V				0	1	0	0
3.75 V	3.67 V				1	0	0	0
4.06 V	3.98 V				1	1	0	0
其它		禁止设置						

注 在使用引导交换操作时，由于 000C1H 与 010C1H 相互切换，须将 000C1H 设置为与 010C1H 相同的值。

注意事项 必须将位 4 设置为 1。

备注 关于 LVD 设置，请参阅 24.1 电压检测电路的功能。

图 27-2. 用户选项字节(000C1H/010C1H)的格式 (2/2)

地址: 000C1H/010C1H[#]

7	6	5	4	3	2	1	0
VPOC2	VPOC1	VPOC0	1	LVIS1	LVIS0	LVIMDS1	LVIMDS0

• LVD 设置(LVD OFF)

检测电压		选项字节设置值						
V _{LVD}		模式设置		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0
上升沿	下降沿	LVIMDS1	LVIMDS0					
-	-	0/1	1	1	×	×	×	×
其它		禁止设置						

注 在使用引导交换操作时，由于 000C1H 与 010C1H 相互切换，须将 000C1H 设置为与 010C1H 相同的值。

注意事项 必须将位 4 设置为 1。

备注 1. ×: 忽略

2. 关于 LVD 设置，请参阅 24.1 电压检测电路的功能。

图 27-3. 选项字节(000C2H/010C2H)的格式

地址: 000C2H/010C2H^注

7	6	5	4	3	2	1	0
CMODE1	CMODE0	1	0	FRQSEL3	FRQSEL2	FRQSEL1	FRQSEL0

CMODE1	CMODE0	Flash 操作模式的设置		
			工作频率范围	工作电压范围
1	0	LS (低速主)模式	1 至 8 MHz	2.7 至 5.5 V
1	1	HS (高速主)模式	1 至 32 MHz	2.7 至 5.5 V
其它		禁止设置		

CMODE1	CMODE0	Flash 操作模式的设置
1	0	LS (低速主)模式
1	1	HS (高速主)模式
其它		禁止设置

FRQSEL3	FRQSEL2	FRQSEL1	FRQSEL0	高速片上振荡器的频率
1	0	0	0	32 MHz
0	0	0	0	24 MHz
1	0	0	1	16 MHz
0	0	0	1	12 MHz
1	0	1	0	8 MHz
1	0	1	1	4 MHz
1	1	0	1	1 MHz
其它				禁止设置

注 使用引导交换操作时，由于 000C2H 与 010C2H 相互切换，须将 000C2H 设置为与 010C2H 相同的值。

27.3 片上调试选项字节的格式

片上调试选项字节的格式如下所示。

图 27-4. 片上调试选项字节(000C3H/010C3H)的格式

地址: 000C3H/010C3H^注

7	6	5	4	3	2	1	0
OCDENSET	0	0	0	0	1	0	OCDERSD

OCDENSET	OCDERSD	片上调试操作控制
0	0	禁止片上调试操作
0	1	禁止设置
1	0	允许片上调试操作。 在片上调试安全 ID 验证失败时擦除闪存数据。
1	1	允许片上调试操作。 在片上调试安全 ID 验证失败时不擦除闪存数据。

注 在使用引导交换操作时，由于 000C3H 与 010C3H 相互切换，须将 000C3H 设置为与 010C3H 相同的值。

注意事项 仅限位 7 和 0 (OCDENSET 和 OCDERSD)，可以指定值。
必须将位 6 至位 1 设置为 000010B。

备注 位 3 至位 1 的值在使用 On-chip 调试功能时将被改写，因此在设置后将变得不确定。
然而，请确保在设置时将位 3 至位 1 设为默认值 (0、1 和 0)。

27.4 选项字节的设置

除通过源程序设置以外，用户选项字节和片上调试选项字节也可以通过编译器的连接器选项进行设置。设置时，利用连接器选项进行设置的内容优先，即使源程序中存在设置也是如此，如下所示。

以下为选项字节设置的软件设置示例。

OPT	CSEG	OPT_BYTE
DB	36H	; 禁用看门狗定时器的间隔中断, ; 允许看门狗定时器操作, ; 看门狗定时器的窗口开启期间为 50%, ; 看门狗定时器的溢出时间为 29/f _{IL} , ; HALT/STOP 模式时, 停止看门狗定时器操作
DB	7AH	; 给 VLVDL 选择 2.75 V ; 给 VLVDH 选择上升沿 2.92 V, 选择下降沿 2.86 V ; LVD 操作模式选择中断&复位模式
DB	ADH	; Flash 操作模式选择 LS (低速主)模式 高速片上振荡器频率选择 1 MHz
DB	85H	; 允许片上调试操作, 不擦除闪存 数据 (当安全 ID 验证失败时)

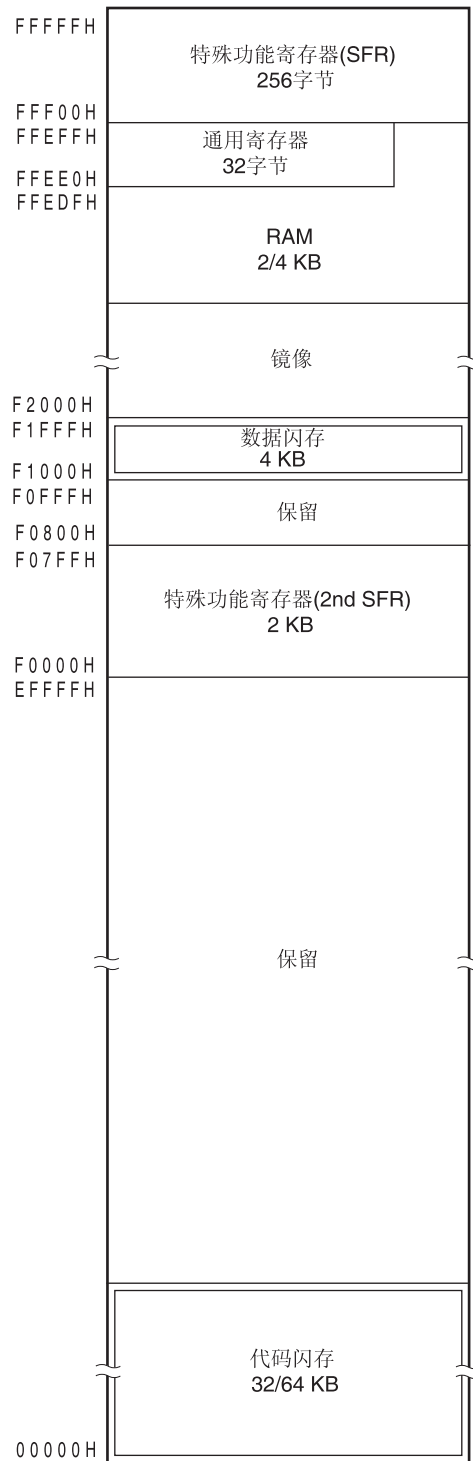
当在自编程过程中使用引导交换功能时，由于 000C0H 至 000C3H 与 010C0H 至 010C3H 相互切换。因此，须按以下方式将 010C0H 至 010C3H 设置为与 000C0H 至 000C3H 相同的值。

OPT2	CSEG	AT	010C0H
DB	36H		; 禁用看门狗定时器的间隔中断, ; 允许看门狗定时器操作, ; 看门狗定时器的窗口开启期间为 50%, ; 看门狗定时器的溢出时间为 29/f _{IL} , ; HALT/STOP 模式时, 停止看门狗定时器操作
DB	7AH		; 给 VLVDL 选择 2.75 V ; 给 VLVDH 选择上升沿 2.92 V, 选择下降沿 2.86 V ; LVD 操作模式选择中断&复位模式
DB	ADH		; Flash 操作模式选择 LS (低速主)模式 高速片上振荡器频率选择 1 MHz
DB	85H		; 允许片上调试操作, 不擦除闪存 数据 (当安全 ID 验证失败时)

注意事项 用汇编语言设置选项字节时，将 **OPT_BYTE** 用作 **CSEG** 伪指令的重定位属性。若要将选项字节设置给 **010C0H** 至 **010C3H** 以使用引导交换功能，须利用重定位属性 **AT** 来指定一个绝对地址。

第二十八章 闪存

RL78/I1A 内置闪存，安装在电路板上时可对其执行程序的写入、擦除和改写。闪存包含用于存储可执行程序的“代码闪存”和用于存储数据的“数据闪存”。



闪存区域的编程方法，共有如下 2 种：

- 使用闪存编程器写入闪存 (参阅 28.1)
- 自编程 (参阅 28.6)

28.1 使用闪存编程器写入闪存

下列专用闪存编程器可用于将数据写入 RL78/I1A 的内部闪存。

- PG-FP5, FL-PR5
- E1 片上调试仿真器

通过专用闪存编程器，可以板上或板外向闪存写入数据。

(1) 板上编程

闪存的内容可以在将 RL78/I1A 安装在目标系统之后改写。连接专用闪存编程器的连接器必须安装在目标系统中。

(2) 板外编程

在将 RL78/I1A 安装到目标系统之前，可以用一个专用编程适配器（FA 系列）将数据写入闪存。

备注 FL-PR5 和 FA 系列为 Naito Densai Machida Mfg. Co., Ltd.的产品。

表 28-1. RL78/I1A 和专用闪存编程器的连线表

专用闪存编程器的引脚配置				引脚名称	引脚编号			
信号名称		输入/输出	引脚功能		20 引脚	30 引脚	32 引脚	38 引脚
PG-FP5, FL-PR5	E1 片上调试 仿真器				TSSOP	SSOP	VQFN (5x6)	SSOP
-	TOOL0	输入/输出	发送/接收信号	TOOL0/P40	3	5	2	5
SI/RxD	-	输入/输出	发送/接收信号					
SCK	-	输出	-	-	-	-	-	-
CLK	-	输出	-	-	-	-	-	-
-	RESET	输出	复位信号	RESET	4	6	3	6
/RESET	-	输出						
FLMD0	-	输出	模式信号	-	-	-	-	-
V _{DD}		输入/输出	V _{DD} 电压生成/ 电源监视	V _{DD}	10	12	9	14
GND		-	接地	V _{SS}	9	11	8	13
				REGC ^注	8	10	7	12
EMV _{DD}		-	TOOL0 引脚驱动电源	V _{DD}	10	12	9	14

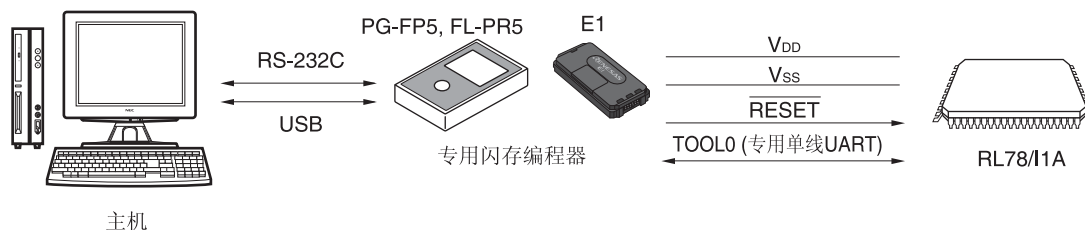
注 通过一个电容(默认: 0.47 至 1 μ F)将 REGC 引脚接地。

备注 在利用闪存编程器进行闪存编程时, 上表未列出的引脚可以保持开路。

28.1.1 编程环境

将程序写入 RL78/I1A 的闪存时所需环境如下所示。

图 28-1. 将程序写入闪存时所需环境



需要一台主机来控制专用闪存编程器。

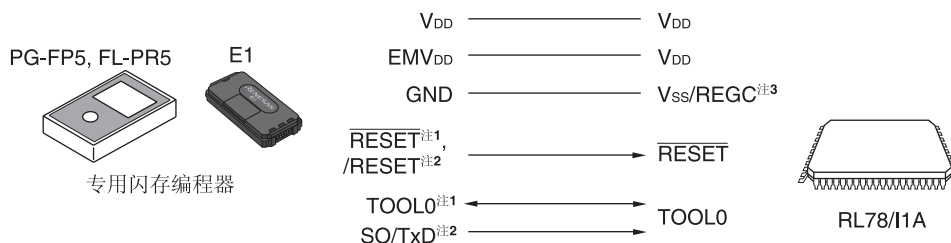
作为专用闪存编程器与 RL78/I1A 之间的接口，专用的单线 UART 利用 TOOL0 引脚执行写入、擦除等操作。

28.1.2 通信模式

专用闪存编程器与 RL78/I1A 之间的通信，是利用 RL78/G13 的专用单线 UART，通过 TOOL0 引脚，以串行通信方式实现的。

传送速率：1 M, 500 k, 250 k, 115.2 kbps

图 28-2. 与专用闪存编程器的通信



- 注 1. 使用 E1 片上调试仿真器时。
2. 使用 PG-FP5 或 FL-PR5 时。
3. 通过一个电容(默认: 0.47 至 1 μ F)将 REGC 引脚接地。

专用闪存编程器为 RL78/I1A 产生以下信号。有关详情，请参阅 PG-FP5、FL-PR5 或 E1 片上调试仿真器的手册。

表 28-2. 引脚连接

专用闪存编程器			RL78/I1A	连接	
信号名称		输入/输出	引脚名称		
PG-FP5, FL-PR5	E1 片上调试仿 真器				
FLMD0	–	输出	模式信号	–	×
V _{DD}		输入/输出	V _{DD} 电压生成/电源监视	V _{DD}	◎
GND		–	接地	V _{SS} , REGC ^注	◎
EMV _{DD}		–	TOOL 引脚驱动电源	V _{DD}	◎
CLK	–	输出	时钟输出	–	×
/RESET	–	输出	复位信号	RESET	◎
–	RESET	输出			
–	TOOL0	输入/输出	发送/接收信号	TOOL0	◎
SI/RxD	–	输入/输出	发送/接收信号		
SCK	–	输出	传送时钟	–	×

注 通过一个电容(默认: 0.47 至 1 μ F)将 REGC 引脚接地。

备注 ◎: 请务必连接此引脚。

×: 不需要连接此引脚。

28.2 板上引脚连接

若要利用闪存编程器板上写入闪存，则目标系统上必须提供连接专用闪存编程器的连接器。首先，电路板上要提供用于选择正常操作模式或闪存编程模式的功能。

当设置为闪存编程模式时，所有在闪存编程过程中未使用的引脚，其状态与复位后的瞬时状态相同。因此，如果外部器件未能在复位后立即识别出该状态，则必须按以下方式处理这些引脚。

备注 关于闪存编程模式，请参阅 **28.6 通过自编程对闪存编程**。

28.2.1 P40/TOOL0 引脚

在闪存编程模式下，通过一个外部 1 kΩ 上拉电阻将该引脚连接至专用闪存编程器。

当该引脚用作端口引脚时，须按以下方式使用该引脚。

用作输入引脚时： 引脚复位解除后，在 1 ms 之内禁止输入低电平。另外，通过下拉电阻使用该引脚时，须使用 500 kΩ 或更大的电阻。

用作输出引脚时： 通过下拉电阻使用该引脚时，须使用 500 kΩ 或更大的电阻。

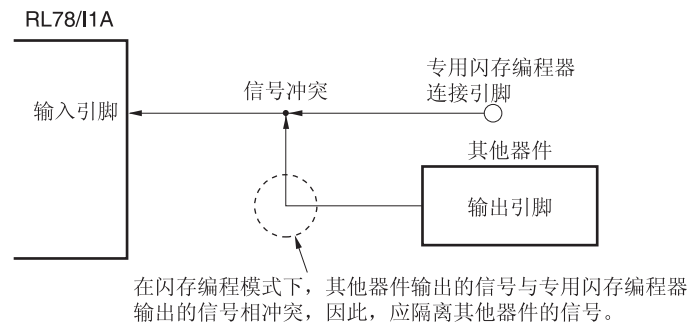
备注 SAU 和 IICA 引脚不用于 RL78/I1A 和专用闪存编程器之间的通信，因为使用的是单线 UART（TOOL0 引脚）。

28.2.2 $\overline{\text{RESET}}$ 引脚

如果将专用闪存编程器和外部器件的复位信号连接至 $\overline{\text{RESET}}$ 引脚，而该引脚已经和板上的复位信号发生电路相连时，发生信号冲突。为避免这种冲突，连接时须和复位信号发生电路相隔离。

在设置为闪存编程模式时，如果从用户系统输入复位信号，则不能对于闪存进行正确编程。不要输入专用闪存编程器的复位信号之外的任何信号。

图 28-3. 信号冲突 ($\overline{\text{RESET}}$ 引脚)



28.2.3 端口引脚

在设置为闪存编程模式时，所有在闪存编程过程中未使用的引脚将进入与复位后的瞬时状态相同的状态。如果连接至端口的外部器件未能在复位后立即识别出端口状态，则必须通过一个电阻将端口引脚连接至 V_{DD} 或 V_{SS} 。

28.2.4 REGC引脚

和正常操作相同的方式通过一个电容（0.47 至 1 μF ）将 REGC 引脚连接至 GND。并且，请使用具有良好特性的电容以稳定内部电压。

28.2.5 X1 和X2 引脚

X1、X2 的连接方法和正常操作模式相同。

备注 在闪存编程模式下，使用高速片上振荡器时钟 (f_{IH})。

28.2.6 电源

使用闪存编程器的电源输出时，须将 V_{DD} 引脚连接至闪存编程器的 V_{DD} ，并将 V_{SS} 引脚连接至闪存编程器的 GND。

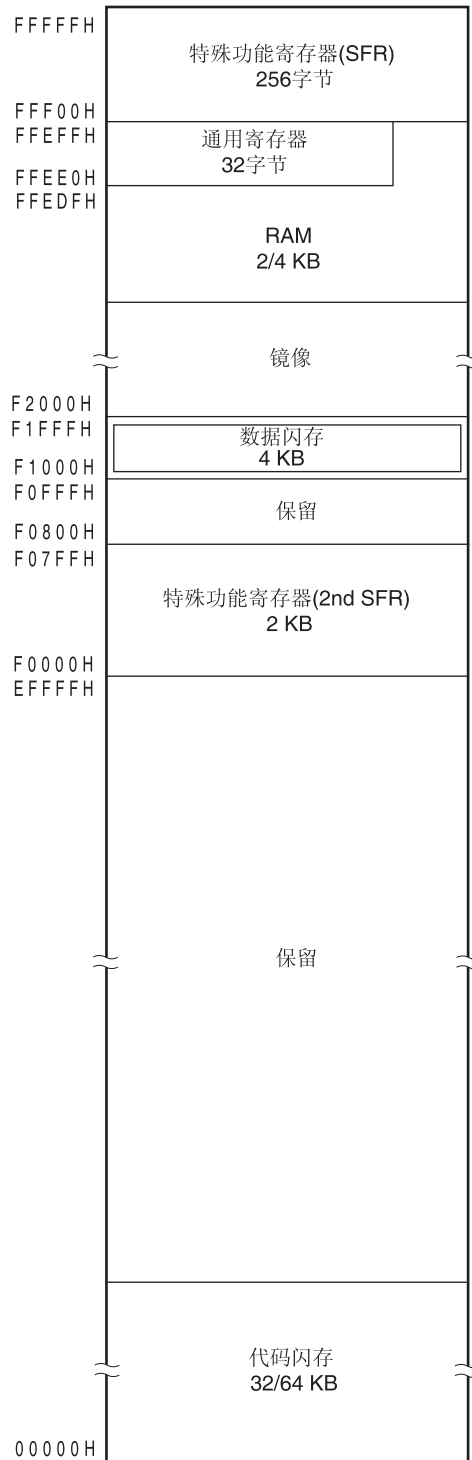
使用板上电源时，则须按正常操作模式连接。

然而，在利用闪存编程器向闪存进行写操作时，即使采用板上电源电压，也必须将 V_{DD} 和 V_{SS} 引脚连接至闪存编程器的 V_{DD} 和 GND，以在闪存编程器上使用电源监控功能。

28.3 数据闪存

28.3.1 数据闪存概述

内置数据闪存的 RL78/I1A 中，除了 32/64 KB 的代码闪存以外，还内置有 4 KB 的数据闪存以存储数据。



数据闪存的概述如下所示。

- 可以利用闪存编程器或外部器件对于数据闪存进行编程操作
- 编程单位为 8 位(半字写入)
- 可按 1 块=1 KB 单位擦除
- CPU 指令仅可执行字节读取 (1 个时钟周期 + 等待 3 个时钟周期)
- 由于数据闪存是数据专用区域, 因此不能用于执行指令 (取代码)
- 可以在改写数据闪存的同时从代码闪存执行指令 (即支持后台运行(BGO))
- 在改写代码闪存 (在自编程期间) 时不能存取数据闪存
- 由于数据闪存在复位解除后将停止工作, 所以必须设置数据闪存控制寄存器 (DFLCTL) 以使用数据闪存
- 在改写数据闪存时, 不能操作 DFLCTL 寄存器
- 在改写数据闪存时, 不能转换至 HALT/STOP 状态。
- 在存取数据闪存时, CPU 将等待三个时钟周期

备注 关于闪存编程模式, 请参阅 28.6 通过自编程对闪存编程。

28.3.2 控制数据闪存的寄存器

(1) 数据闪存控制寄存器(DFLCTL)

该寄存器用于允许或禁止存取数据闪存。

使用 1 位或 8 位存储器操作指令设置 DFLCTL 寄存器。

产生复位信号后, 该寄存器被清除为 00H。

图 28-4. 数据闪存的格式(DFLCTL)

地址: F0090H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	<0>
DFLCTL	0	0	0	0	0	0	0	DFLEN

DFLEN	数据闪存的存取控制
0	禁止数据闪存存取
1	允许数据闪存存取

注意事项 在改写数据闪存时, 禁止操作 DFLCTL 寄存器。

28.3.3 存取数据闪存的步骤

复位解除后，数据闪存的初始状态为停止工作，不能被存取（读或编程）。若要存取该存储器，则须执行以下步骤：

<1> 在数据闪存控制寄存器 (DFLCTL)的位 0 (DFLEN)写入 1。

<2> 通过软件定时器等来等待设置结束。

设置时间因主时钟的各模式而异。

<各主时钟模式下的设置时间>

- HS (高速主): 5 μ s
- LS (低速主): 720 ns

<3> 等待设置时间后，可以存取数据闪存。

注意事项 1. 设置时间内，禁止存取数据闪存。

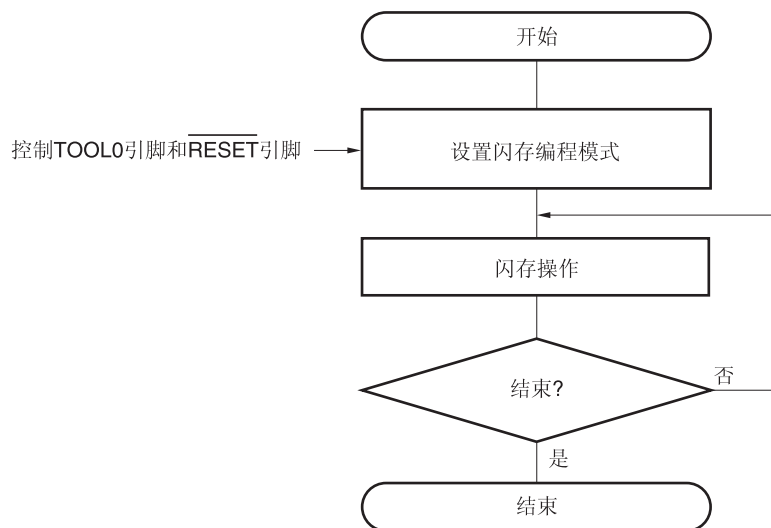
2. 设置时间内，在执行 STOP 指令之前，要暂时将 DFLEN 清除为 0。

28.4 编程方法

28.4.1 闪存的控制

操作闪存的步骤，如下图所示。

图 28-5. 闪存的操作步骤



28.4.2 闪存的编程模式

改写闪存的内容，须将 RL78/I1A 设置为闪存编程模式。若要进入该模式，须按以下方式设置。

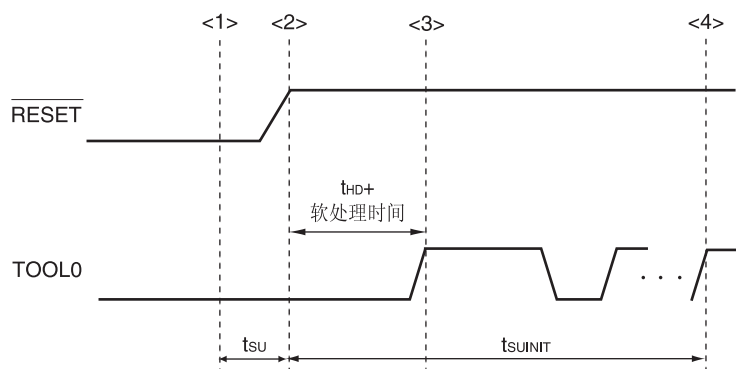
<使用专用闪存编程器进行编程时>

执行来自专用闪存编程器的通信，以自动切换到闪存编程模式。

<使用外部器件进行编程时>

将 TOOL0 引脚设置为低电平，然后解除复位。然后解除复位。从复位结束到 1 ms + 软件处理结束的这段时间内，使 TOOL0 引脚保持低电平，然后利用外部器件的 UART 通信发送来自外部器件的数据“00H”。在复位结束后的 100 ms 内完成 UART 通信的初始化设置。

图 28-6. 设置闪存的编程模式



<1> 低电位输入到 TOOL0 引脚。

<2> 引脚复位解除（必须在结束引脚复位之前结束 POR 和 LVD 复位）。

<3> 将 TOOL0 引脚设置为高电位。

<4> 通过 UART 接收设置闪存编程模式，并完成波特率设置。

备注 tsuINIT: 该段表示在解除复位后须在 100 ms 内完成初始通信设置的设置。

tsu: 从 TOOL0 引脚被设置为低电平到解除引脚复位的时间

tHD: 解除外部和内部复位后，将 TOOL0 引脚保持为低电平的时间(软件处理时间除外)

表 28-3. 复位解除时的 TOOL0 引脚和操作模式之间的关系

TOOL0	操作模式
V _{DD}	正常操作模式
0	闪存编程模式

28.4.3 通信模式

RL78/I1A 的通信模式如下所示。

表 28-4. 通信模式

通信模式	标准设置 ^{注1}				使用引脚
	端口	速率 ^{注2}	频率	乘法率	
单线模式 (使用闪存编程器时)	UART	115200 bps, 250000 bps, 500000 bps, 1 Mbps	–	–	TOOL0

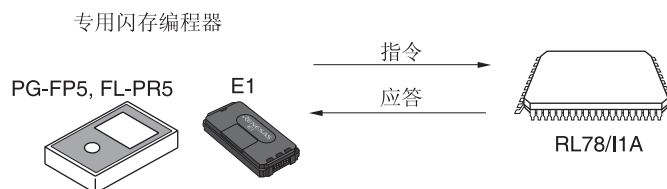
注 1. 闪存编程器的 GUI 上标准设置的设置项目。

2. 波特率误差之外的因素（如信号波形失真）也会影响 UART 通信，因此要充分评估失真和波特率误差。

28.4.4 通信命令

RL78/I1A 通过命令与专用闪存编程器进行通信。从闪存编程器或外部器件发送至 RL78/I1A 的信号称为命令，从 RL78/I1A 发送至专用闪存编程器或外部器件的信号称为响应。

图 28-7. 通信命令



RL78/I1A 的闪存控制命令列于下表之中。所有这些命令都来自编程器，RL78/I1A 执行与各命令相对应的处理。

表 28-5. 闪存控制命令

分类	命令名称	功能
校验	Verify	比较闪存中指定区域的内容和编程器发送的数据。
擦除	Block Erase	擦除闪存的指定区域。
查空	Block Blank Check	检查闪存的指定块是否已被正确擦除。
写入	Programming	将数据写入闪存的指定区域。
获取信息	Silicon Signature	获取 RL78/I1A 的信息（例如产品编号、闪存配置）。
	Version Get	获取 RL78/I1A 的固件版本。
	Checksum	获取指定区域的校验和数据。
安全	Security Set	设置安全信息。
	Security Release	解除写入禁止设置。
其他	Reset	用于检测通信同步状态。
	Baud Rate Set	选择 UART 通信模式时用于设置波特率。

对于专用闪存编程器发出的命令，RL78/I1A 返回一个响应。从 RL78/I1A 发送出的响应名称如下表所示。

表 28-6. 响应名称

响应名称	功能
ACK	确认命令/数据。
NAK	确认非法的命令/数据。

28.4.5 签名数据的说明

执行“Silicon Signature”命令时，可以获取 RL78/I1A 信息（如产品编号、闪存配置、编程用固件版本）。

表 28-7 和表 28-8 列出了签名数据列表和签名数据示例。

表 28-7. 签名数据列表

字段名称	说明	传输数据的数量
器件代码	分配给该器件的序列号	3 字节
器件名称	器件名称 (ASCII 码)	10 字节
代码闪存区域最后地址	代码闪存区域的最后地址 (从低位地址发送。 示例. 00000H 至 0FFFFH (64 KB) → FFH, 1FH, 00H)	3 字节
数据闪存区域最后地址	数据闪存区域的最后地址 (从低位地址发送。 示例. F1000H 至 F1FFFH (4 KB) → FFH, 1FH, 0FH)	3 字节
固件版本	编程固件版本信息 (从高位地址发送。 示例. 从 Ver. 1.23 → 01H, 02H, 03H)	3 字节

表 25-8. 签名数据示例

字段名称	说明	传输数据的数量	数据(十六进制)
器件代码	RL78 协议 A	3 字节	10 00 06
器件名称	R5F107AE	10 字节	52 = “R” 35 = “5” 46 = “F” 31 = “1” 30 = “0” 37 = “7” 41 = “A” 45 = “E” 20 = “ ” 20 = “ ”
代码闪存区域最后地址	代码闪存区域 00000H 至 0FFFFH (64 KB)	3 字节	FF FF 00
数据闪存区域最后地址	数据闪存区域 F1000H 至 F1FFFH (4 KB)	3 字节	FF 1F 0F
固件版本	Ver.1.23	3 字节	01 02 03

28.5 安全设置

RL78/I1A 支持安全功能，可以禁止对已经写入内部闪存的用户程序进行改写，因此，未经授权者不能更改程序。使用安全设置命令，可以实现以下所示的操作。

- 禁止块擦除
板上/板外编程过程中，禁止对闪存中指定块执行块擦除命令。然而，可以通过自编程方式擦除块。
- 禁止写入
板上/板外编程过程中，禁止对闪存的所有块执行写命令。然而，可以通过自编程方式写入块。
- 禁止改写引导群集 0
在这种设置下，禁用对闪存的引导群集 0 (00000H 至 00FFFH) 执行块擦除命令和写命令。

执行了安全设置后，通过安全解除命令来解除安全设置时，复位后生效。

在闪存出厂时，默认允许块擦除、写命令和改写引导群集 0。安全性可以通过板上/板外编程和自编程设置。各种安全设置均可组合使用。

表 28-9 表示 RL78/I1A 安全功能有效时擦除命令与写命令之间的关系。

注意事项 闪存编程器的安全功能不支持自编程。

备注 若要在自编程过程中禁止写和擦除，须使用闪存屏蔽窗口功能(详情参阅 28.6.2)。

表 28-9. 安全功能有效时与命令之间的关系

(1) 板上/板外编程时

有效的安全	执行的命令	
	块擦除	写入
禁止块擦除	不能擦除块。	可以执行。 ^注
禁止写入	可以擦除块。	不能执行。
禁止改写引导群集 0	不能擦除引导群集 0。	不能写入引导群集 0。

注 确认待写入区域未曾写入数据。由于在禁用块擦除之后无法擦除数据，因此，如果尚未擦除数据，则不要写入数据。

(2) 自编程时

有效的安全	执行的命令	
	块擦除	写入
禁止块擦除	可以擦除块。	可以执行。
禁止写入		
禁止改写引导群集 0	不能擦除引导群集 0。	不能写入引导群集 0。

备注 若要在自编程过程中禁止写和擦除，须使用闪存屏蔽窗口功能(详情参阅 28.6.2)。

表 28-10. 各编程模式下的安全设置

(1) 板上/板外编程

安全	安全设置	如何使安全设置无效
禁止块擦除	利用专用闪存编程器的 GUI 等设置。	设置后无法设置为无效。
禁止写入		执行安全解除命令。
禁止改写引导群集 0		设置后无法设置为无效。

注意事项 仅当块擦除禁止、引导集群 0 改写禁止，且代码闪存区和数据闪存区为空时，安全解除命令有效。

(2) 自编程

安全	安全设置	如何使安全设置无效
禁止块擦除	利用闪存自编程库设置。	设置后无法设置为无效。
禁止写入		在板上/板外编程时执行安全解除命令(自编程时无法设置为无效)。
禁止改写引导群集 0		设置后无法设置为无效。

28.6 通过自编程对闪存编程

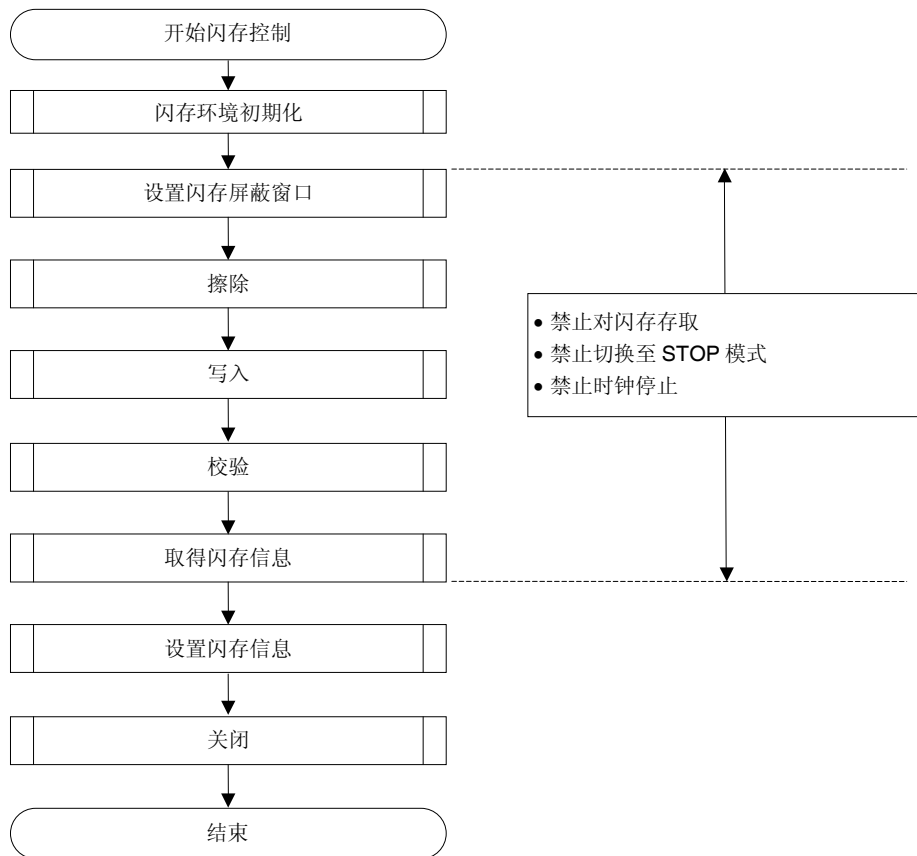
RL78/I1A 支持自编程功能，可用于通过用户程序改写闪存。由于该功能允许用户应用程序通过 RL78/I1A 自编程库改写闪存，因此可用该功能来现场升级程序。

- 注意事项**
1. 当 CPU 以副系统时钟运行时不能使用自编程功能。
 2. 若要禁止在自编程过程中产生中断，和通常操作模式一样，通过 DI 指令将 IE 标志清零(0)后执行自编程库。若要允许中断，通过 EI 指令将 IE 标志置位(1)并将所受理中断的中断屏蔽标志清零(0)，然后执行自编程库。
 3. 当允许 RAM 奇偶校验错误复位(RPERDIS = 0)时，必须在改写前对所用 RAM 区 + 10 字节的区域进行初始化。

- 备注**
1. 有关自编程功能和 RL78/I1A 自编程库的更多信息，请参阅 **RL78 单片机自编程库 01 类用户手册 (R01AN0350E)**。
 2. 有关执行自编程所需时间的详情，请参阅随闪存自编程库工具一起提供的应用笔记。

使用自编程库改写闪存的流程如下图所示。

图 28-8. 自编程流程 (改写闪存)



28.6.1 引导交换功能

如果因临时断电或其他原因导致改写引导区失败，以至于引导区数据遭到破坏，则通过复位或改写来重启程序的功能将被禁用。

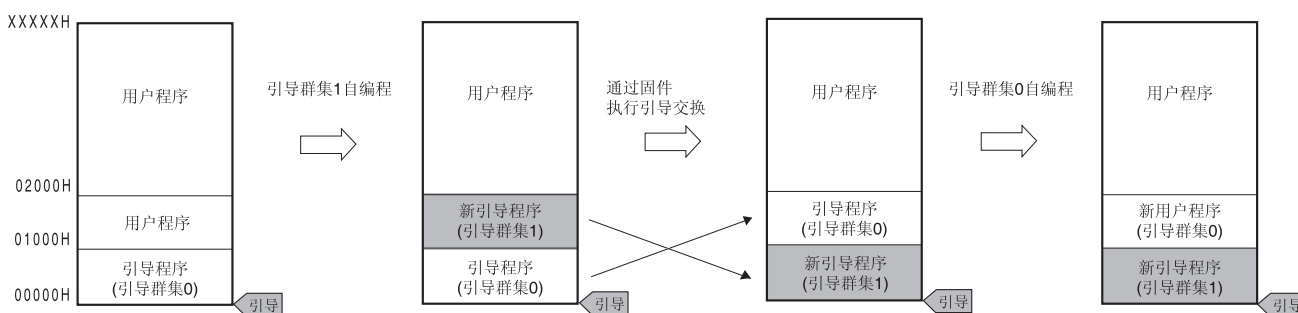
引导交换功能用于避免该问题。

在擦除引导群集 0*（引导程序区）之前，提前通过自编程向引导群集 1 写入一个新的引导程序。当程序已被正确写入引导群集 1 时，利用 RL78/I1A 固件的设置信息功能来交换引导群集 1 和引导群集 0，以使引导群集 1 被用作引导区。此后，擦除或写入原始引导程序区，即引导群集 0。

之后，即使在改写引导编程区的过程中断电，也能够正确执行程序，因为当程序被复位并重新启动时，是从被交换的引导群集 1 进行的引导。

注 引导群集是一个 4 KB 的区域，通过引导交换功能来交换引导群集 0 和 1。

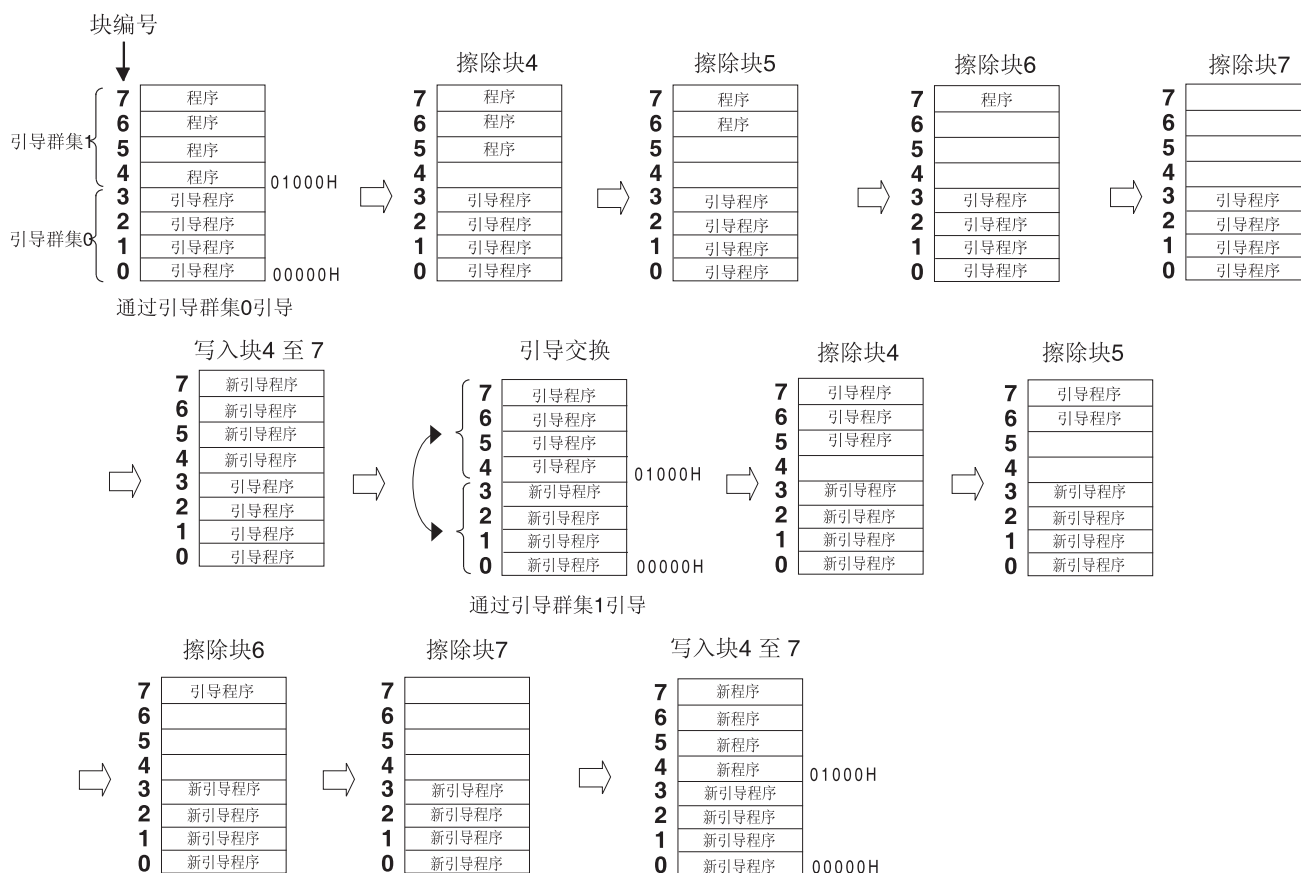
图 28-9. 引导交换功能



上图的示例如下所示。

- 引导群集 0: 引导交换前的引导程序区
- 引导群集 1: 引导交换后的引导程序区

图 28-10. 执行引导交换的示例



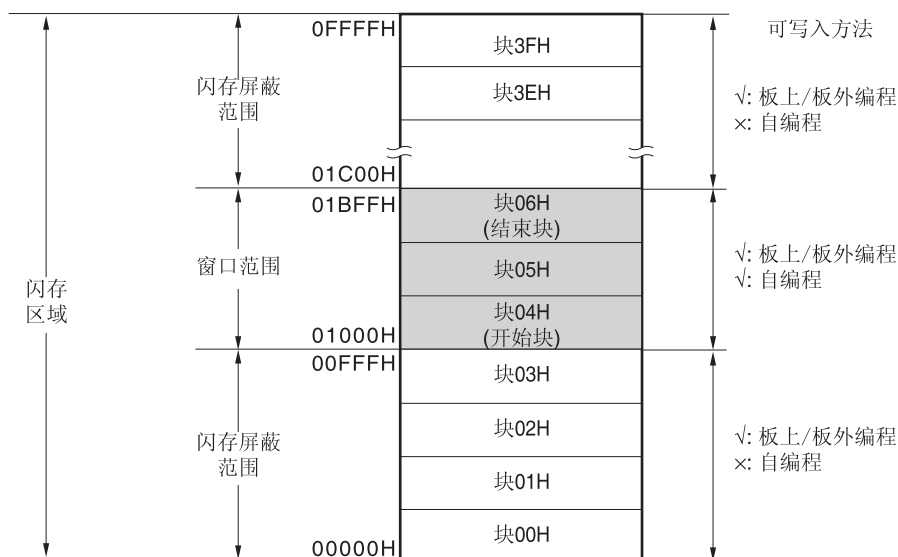
28.6.2 闪存屏蔽窗口功能

闪存屏蔽窗口功能是自编程的一种安全功能。该功能仅在自编程过程中，禁止写入和擦除指定窗口范围之外的区域。

窗口范围可通过指定起始块和结束块来设置。可以在板上/板外编程和自编程过程中设置或更改窗口范围。

在自编程过程中，禁止写入和擦除窗口范围之外的区域。然而，板上/板外编程时，可以写入和擦除指定窗口范围之外的区域。

图 28-11. 闪存屏蔽窗口设置示例
(对象器件: R5F107AE, 开始块: 04H, 结束块: 06H)



- 注意事项 1. 如果引导群集 0 的禁止改写区与 Flash 屏蔽窗口范围重叠，则优先禁止改写引导群集 0。
- 2. Flash 屏蔽窗口只能用作代码闪存（且不支持数据闪存）。

表28-11. 闪存屏蔽窗口功能设置/更改方法和命令之间的关系

编程条件	窗口范围设置/更改方法	执行的命令	
		块擦除	写入
自编程	通过闪存自编程库指定起始和结束块。	仅在窗口范围内允许块擦除。	仅在窗口范围内允许写入。
板上/板外编程	通过专用闪存编程器的 GUI 等指定起始和结束块。	在窗口范围外也允许块擦除。	在窗口范围外也允许写入。

备注 要想禁止板上/板外编程时的写入/擦除时，请参阅 28.5 安全设置。

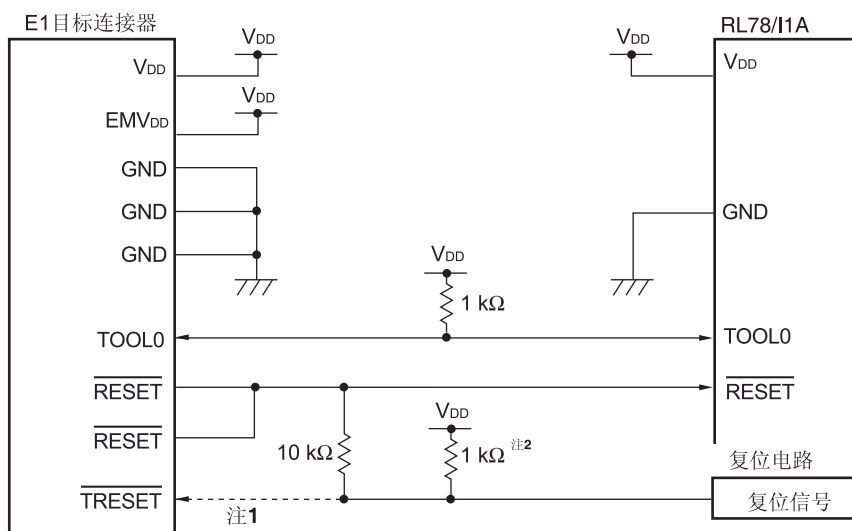
第二十九章 On-chip 调试功能

29.1 将E1 片上调试仿真器连接至RL78/I1A

RL78/I1A 用 V_{DD} 、 $\overline{\text{RESET}}$ 、 TOOL0 和 V_{SS} 引脚通过 E1 on-chip 调试仿真器与主机进行通信。通过使用 TOOL0 引脚的单线 UART 实现串行通信。

注意事项 RL78/I1A 提供用于开发和评估的 On-chip 调试功能。对于量产产品，不得使用 On-chip 调试功能，因为在使用该功能时，可能会超过闪存保证的可改写次数，因而会使产品可靠性无法保证。对于使用 On-chip 调试功能时发生的问题，瑞萨电子不承担任何责任。

图 29-1. E1 On-chip 调试仿真器和 RL78/I1A 的连接示例



注 1. 在闪存编程期间，不需要连接虚线。

2. 如果目标系统中的复位电路不具有缓冲器并且只通过电阻和电容产生复位信号，则没有必要使用该上拉电阻。

注意事项 该电路图假定从 N 沟道 O.D. 缓冲器输出复位信号（输出电阻：100Ω或更小）

29.2 片上调试安全ID

RL78/I1A 在闪存 000C3H 处有一个片上调试操作控制位（参阅**第二十七章 选项字节**），并在 000C4H 至 000CDH 处有一个片上调试安全 ID 设置区，用于防止第三方读取存储器内容。

在使用引导交换功能时，要提前在 010C3H 和 010C4H 至 010CDH 中设置相同的值，因为 000C3H、000C4H 至 000CDH 和 010C3H、010C4H 至 010CDH 被相互切换。

表 29-1. 片上调试安全 ID

地址	片上调试安全 ID
000C4H 至 000CDH	任意 10 个字节的 ID 码
010C4H 至 010CDH	

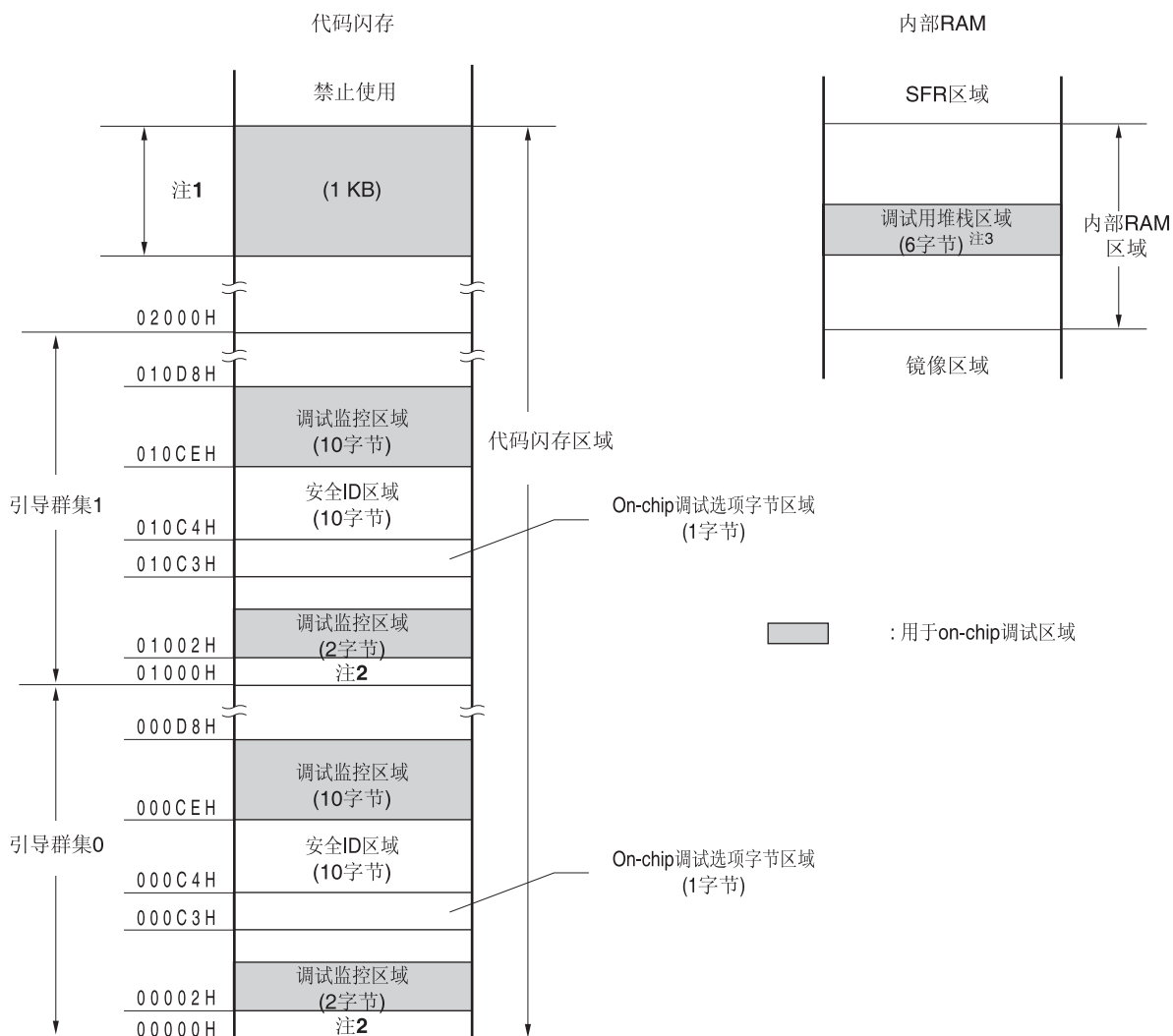
29.3 用户资源的预留

要实现 RL78/I1A 与 E1 On-chip 调试仿真器之间的通信以及各调试功能，需提前预留存储器区空间。如果使用瑞萨电子的编译器，则可通过连接器选项来进行设置。

(1) 存储器空间的预留

图 29-2 中的阴影部分是预留区，用于放置调试监控程序，因此不能将用户程序或数据分配在该空间。在使用 On-chip 调试功能时，必须保证这些空间不被用户程序使用。另外，禁止通过用户程序改写该区域。

图 29-2. 调试监控程序的内存空间分配



注 1. 如下所示，地址因产品而异。

产品	注 1 的地址
R5F1076C, R5F107AC, R5F107BC	07FFFH
R5F107AE, R5F107DE	0FFFFH

2. 当未使用实时 RAM 监控(RRM)功能和动态存储器修改(DMM)功能时，为 256 字节。
3. 在调试中，复位向量被改写至分配给监控程序的地址。
4. 由于该区域被分配在紧邻堆栈区之下，因此，该区域的地址随着堆栈的增减而变化。即所用堆栈区将额外占用 4 字节。
使用自编程时，所用堆栈区将额外占用 12 字节。

第三十章 十进制调整(BCD)电路

30.1 十进制调整电路的功能

通过该电路，可以获得 BCD 码（二进制编码的十进制）相加/减的 BCD 格式的结果。

通过执行以 A 寄存器为操作数的加/减运算，并对 BCD 校正结果寄存器(BCDADJ)执行加/减运算，获得十进制校正运算结果。

30.2 十进制调整电路使用的寄存器

十进制调整电路使用如下的寄存器。

- BCD 校正结果寄存器(BCDADJ)

(1) BCD 校正结果寄存器(BCDADJ)

通过以 A 寄存器为操作数的加/减指令，获得 BCD 码的加/减结果所需要的校正值被存储于 BCDADJ 寄存器。

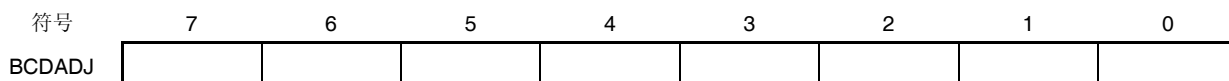
BCDADJ 寄存器的读取值因读取时 A 寄存器以及 CY 和 AC 标志的值而异。

使用 8 位存储器操作指令读取 BCDADJ 寄存器。

产生复位后，该寄存器为不定值。

图 30-1. BCD 校正结果寄存器的格式 (BCDADJ)

地址: F00FEH 复位后: 不确定 R



30.3 十进制调整电路的操作

十进制调整电路的基本操作如下所示。

(1) 加法：以 BCD 码值计算出 BCD 码值与另一个 BCD 码值相加的结果

- <1> 被执行加法运算的 BCD 码值存储于 A 寄存器中。
- <2> 通过将 A 寄存器的值与第二个操作数（被加的另一个 BCD 码值）在二进制格式下相加，二进制运算结果将存储于 A 寄存器中，校正值存储于 BCD 校正结果寄存器(BCDADJ)中。
- <3> 通过在二进制格式下将 A 寄存器的值（二进制的加法运算结果）与 BCDADJ 寄存器（校正值）相加来实现十进制校正运算，校正结果存储于 A 寄存器和 CY 标志中。

注意事项 BCDADJ 寄存器的读取值因读取时 A 寄存器以及 CY 和 AC 标志的值而异。因此，在执行指令 <2>之后，必须先执行指令<3>再执行其他指令。若要在中断允许状态下执行 BCD 校正，则须在中断函数中对 A 寄存器执行进栈和出栈操作。通过 RETI 指令来恢复 PSW（CY 标志和 AC 标志）。

示例如下所示。

示例 1: $99 + 89 = 188$

指令	A 寄存器	CY 标志	AC 标志	BCDADJ 寄存器
MOV A, #99H ; <1>	99H	-	-	-
ADD A, #89H ; <2>	22H	1	1	66H
ADD A, !BCDADJ ; <3>	88H	1	0	-

示例 2: $85 + 15 = 100$

指令	A 寄存器	CY 标志	AC 标志	BCDADJ 寄存器
MOV A, #85H ; <1>	85H	-	-	-
ADD A, #15H ; <2>	9AH	0	0	66H
ADD A, !BCDADJ ; <3>	00H	1	1	-

示例 3: $80 + 80 = 160$

指令	A 寄存器	CY 标志	AC 标志	BCDADJ 寄存器
MOV A, #80H ; <1>	80H	-	-	-
ADD A, #80H ; <2>	00H	1	0	60H
ADD A, !BCDADJ ; <3>	60H	1	0	-

(2) 减法：以 BCD 码值计算出从 BCD 码值减去另一个 BCD 码值的结果

- <1> 被执行减法运算的 BCD 码值存储于 A 寄存器中。
- <2> 在二进制格式下通过从 A 寄存器减去第二个操作数（被减的 BCD 码值），二进制的运算结果将存储于 A 寄存器中，校正值存储于 BCD 校正结果寄存器(BCDADJ)中。
- <3> 通过在二进制格式下从 A 寄存器（二进制的减运算结果）减去 BCDADJ 寄存器的值（校正值）来实现十进制校正运算，校正结果存储于 A 寄存器和 CY 标志中。

注意事项 BCDADJ 寄存器的读取值因读取时 A 寄存器以及 CY 和 AC 标志的值而异。因此，在执行指令 <2> 之后，必须先执行指令 <3> 再执行其他指令。若要在中断允许状态下执行 BCD 校正，则须在中断函数中对 A 寄存器执行进栈和出栈操作。通过 RETI 指令来恢复 PSW（CY 标志和 AC 标志）。

示例如下所示。

示例：91 - 52 = 39

指令	A 寄存器	CY 标志	AC 标志	BCDADJ 寄存器
MOV A, #91H ; <1>	91H	-	-	-
SUB A, #52H ; <2>	3FH	0	1	06H
SUB A, !BCDADJ ; <3>	39H	0	0	-

第三十一章 指令集

本章列出了 RL78 单片机指令集中的指令。有关各指令的详细操作及操作码，请参阅独立文档 **RL78 单片机用户手册：软件篇**。

31.1 操作列表使用规则

31.1.1 操作标识符和标识方法

遵循着指令操作数标识符的标识方法，操作数列于各指令的“操作数”一栏中，遵循指令操作数标识符的规范方法（详见汇编器规范）。当存在两种或更多的标识方法时，应选择其中一种。大写字母以及#、!、!!、\$、\$!、[] 和 ES: 符号为关键字，描述时应保持原样。每个符号的含义如下。

- #: 立即数
- !: 16 位绝对地址
- !!: 20 位绝对地址
- \$: 8 位相对地址
- \$!: 16 位相对地址
- []: 间接地址
- ES:: 扩展地址

立即数可以由一个对应的数值或标号来描述。使用标号时，必须使用 #、!、!!、\$、\$!、[] 和 ES: 符号描述。

用于操作数的寄存器标识符 r 和 rp，可使用功能名称（X、A、C 等）或绝对名称（下表括号中名称，R0、R1、R2 等）进行描述。

表31-1. 操作标识符和标识方法

标识符	标识方法
r	X (R0), A (R1), C (R2), B (R3), E (R4), D (R5), L (R6), H (R7)
rp	AX (RP0), BC (RP1), DE (RP2), HL (RP3)
sfr	特殊功能寄存器符号(SFR 符号) FFF00H 至 FFFFFH
sfrp	特殊功能寄存器符号(可 16 位操作的 SFR 符号。仅用于偶数地址 ^{*)} FFF00H 至 FFFFFH
saddr	FFE20H 至 FFF1FH 区立即数或标号
saddrp	FFE20H 至 FF1FH 区立即数或标号(仅用于偶数地址 ^{*)})
addr20	00000H 至 FFFFFH 区立即数或标号
addr16	0000H 至 FFFFH 区立即数或标号 (16 位数据时仅用于偶数地址 ^{*)})
addr5	0080H 至 00BFH 区立即数或标号 (仅用于偶数地址)
word	16 位立即数或标号
byte	8 位立即数或标号
bit	3 位立即数或标号
RBn	RB0 至 RB3

注 指定为偶数地址时，位0 = 0。

备注 特殊功能寄存器的符号，可用于描述操作数 sfr。关于特殊功能寄存器的符号，请参阅表 3-5. SFR 列表。扩展特殊功能寄存器的符号，可用于描述操作数!addr16。关于扩展特殊功能寄存器的符号，请参阅表 3-6. 扩展 SFR (2nd SFR)列表。

31.1.2 操作栏的说明

在各指令的操作栏中，指令执行时的动作作用以下的符号表示。

表31-2. 操作栏的符号

符号	功能
A	A 寄存器；8 位累加器
X	X 寄存器
B	B 寄存器
C	C 寄存器
D	D 寄存器
E	E 寄存器
H	H 寄存器
L	L 寄存器
ES	ES 寄存器
CS	CS 寄存器
AX	AX 寄存器对；16 位累加器
BC	BC 寄存器对
DE	DE 寄存器对
HL	HL 寄存器对
PC	程序计数器
SP	堆栈指针
PSW	程序状态字
CY	进位标志
AC	辅助进位标志
Z	零标志
RBS	寄存器组选择标志
IE	中断要求允许标志
()	()内的地址或寄存器的内容所指示的存储器的内容
X _H , X _L	16 位寄存器：X _H = 高 8 位, X _L = 低 8 位
X _S , X _H , X _L	20 位寄存器：X _S = (位 19 至 16), X _H = (位 15 至 8), X _L = (位 7 至 0)
∧	逻辑与 (AND)
∨	逻辑或 (OR)
⊕	异或 (exclusive OR)
—	反转数据
addr5	16 位立即数 (仅限 0080H 至 00BFH 的偶数地址)
addr16	16 位立即数
addr20	20 位立即数
jdisp8	有符号的 8 位数据 (偏移量)
jdisp16	有符号的 16 位数据 (偏移量)

31.1.3 标志栏的说明

在各指令的标志栏中，指令执行时标志值的变化用以下符号表示。

表31-3. 标志栏的符号

符号	标志值的变化
(Blank)	无变化
0	清除为 0
1	设置为 1
x	根据结果设置为 1/清除为 0
R	恢复以前保存的值

31.1.4 PREFIX 指令

通过将“ES:”作为 PREFIX 指令码置于指令之前，可以将可存取数据区域从 64 KB 空间（F0000H 至 FFFFFH 区）扩展至 1 MB 空间（00000H 至 FFFFFH 区）。把 PREFIX 指令码作为前缀附加到目标指令上时，只有正好位于 PREFIX 操作代码之后的一条指令按照添加了 ES 寄存器值的地址被执行。

在 PREFIX 指令代码与紧随其后的指令之间，不受理中断和 DMA 传送。

表31-4. PREFIX操作代码示例

指令	操作代码				
	1	2	3	4	5
MOV !addr16, #byte	CFH	!addr16		#byte	-
MOV ES:!addr16, #byte	11H	CFH	!addr16		#byte
MOV A, [HL]	8BH	-	-	-	-
MOV A, ES:[HL]	11H	8BH	-	-	-

注意事项 在执行 PREFIX 指令之前，以 MOV ES、A 等设置 ES 寄存器值。

31.2 操作列表

表31-5. 操作列表(1/17)

指令组	助记符	操作数	字节	时钟		时钟	标志		
				注1	注2		Z	AC	CY
8 位数据 传送	MOV	r, #byte	2	1	-	r ← byte			
		PSW, #byte	3	3	-	PSW ← byte	×	×	×
		CS, #byte	3	1	-	CS ← byte			
		ES, #byte	2	1	-	ES ← byte			
		laddr16, #byte	4	1	-	(addr16) ← byte			
		ES:laddr16, #byte	5	2	-	(ES, addr16) ← byte			
		saddr, #byte	3	1	-	(saddr) ← byte			
		sfr, #byte	3	1	-	sfr ← byte			
		[DE+byte], #byte	3	1	-	(DE+byte) ← byte			
		ES:[DE+byte], #byte	4	2	-	((ES, DE)+byte) ← byte			
		[HL+byte], #byte	3	1	-	(HL+byte) ← byte			
		ES:[HL+byte], #byte	4	2	-	((ES, HL)+byte) ← byte			
		[SP+byte], #byte	3	1	-	(SP+byte) ← byte			
		word[B], #byte	4	1	-	(B+word) ← byte			
		ES:word[B], #byte	5	2	-	((ES, B)+word) ← byte			
		word[C], #byte	4	1	-	(C+word) ← byte			
		ES:word[C], #byte	5	2	-	((ES, C)+word) ← byte			
		word[BC], #byte	4	1	-	(BC+word) ← byte			
		ES:word[BC], #byte	5	2	-	((ES, BC)+word) ← byte			
		A, r ^{#3}	1	1	-	A ← r			
		r, A ^{#3}	1	1	-	r ← A			
		A, PSW	2	1	-	A ← PSW			
		PSW, A	2	3	-	PSW ← A	×	×	×
		A, CS	2	1	-	A ← CS			
		CS, A	2	1	-	CS ← A			
		A, ES	2	1	-	A ← ES			
		ES, A	2	1	-	ES ← A			
		A, laddr16	3	1	4	A ← (addr16)			
		A, ES:laddr16	4	2	5	A ← (ES, addr16)			
		laddr16, A	3	1	-	(addr16) ← A			
ES:laddr16, A	4	2	-	(ES, addr16) ← A					
A, saddr	2	1	-	A ← (saddr)					
saddr, A	2	1	-	(saddr) ← A					

注 1. 内部 RAM 区域、SFR 区域及扩展 SFR 区域被存取，或者未存取数据时的 CPU 时钟数(fCLK)。

2. 程序存储器区域被存取时的 CPU 时钟数(fCLK)。

3. 除 r = A 以外。

备注 时钟数是指程序存储于内部 ROM (闪存) 区域时的时钟数。如果从内部 RAM 区域获取指令，最大为该值的 2 倍加 3 个时钟。

表31-5. 操作列表(2/17)

指令组	助记符	操作数	字节	时钟		时钟	标志		
				注1	注2		Z	AC	CY
8 位数据 传送	MOV	A, sfr	2	1	-	A ← sfr			
		sfr, A	2	1	-	sfr ← A			
		A, [DE]	1	1	4	A ← (DE)			
		[DE], A	1	1	-	(DE) ← A			
		A, ES:[DE]	2	2	5	A ← (ES, DE)			
		ES:[DE], A	2	2	-	(ES, DE) ← A			
		A, [HL]	1	1	4	A ← (HL)			
		[HL], A	1	1	-	(HL) ← A			
		A, ES:[HL]	2	2	5	A ← (ES, HL)			
		ES:[HL], A	2	2	-	(ES, HL) ← A			
		A, [DE+byte]	2	1	4	A ← (DE + byte)			
		[DE+byte], A	2	1	-	(DE + byte) ← A			
		A, ES:[DE+byte]	3	2	5	A ← ((ES, DE) + byte)			
		ES:[DE+byte], A	3	2	-	((ES, DE) + byte) ← A			
		A, [HL+byte]	2	1	4	A ← (HL + byte)			
		[HL+byte], A	2	1	-	(HL + byte) ← A			
		A, ES:[HL+byte]	3	2	5	A ← ((ES, HL) + byte)			
		ES:[HL+byte], A	3	2	-	((ES, HL) + byte) ← A			
		A, [SP+byte]	2	1	-	A ← (SP + byte)			
		[SP+byte], A	2	1	-	(SP + byte) ← A			
		A, word[B]	3	1	4	A ← (B + word)			
		word[B], A	3	1	-	(B + word) ← A			
		A, ES:word[B]	4	2	5	A ← ((ES, B) + word)			
		ES:word[B], A	4	2	-	((ES, B) + word) ← A			
		A, word[C]	3	1	4	A ← (C + word)			
		word[C], A	3	1	-	(C + word) ← A			
		A, ES:word[C]	4	2	5	A ← ((ES, C) + word)			
		ES:word[C], A	4	2	-	((ES, C) + word) ← A			
		A, word[BC]	3	1	4	A ← (BC + word)			
		word[BC], A	3	1	-	(BC + word) ← A			
		A, ES:word[BC]	4	2	5	A ← ((ES, BC) + word)			
		ES:word[BC], A	4	2	-	((ES, BC) + word) ← A			

注 1. 内部 RAM 区域、SFR 区域及扩展 SFR 区域被存取，或者未存取数据时的 CPU 时钟数(fCLK)。

2. 程序存储器区域被存取时的 CPU 时钟数(fCLK)。

备注 时钟数是指程序存储于内部 ROM (闪存) 区域时的时钟数。如果从内部 RAM 区域获取指令，最大为该值的 2 倍加 3 个时钟。

表31-5. 操作列表(3/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注1	注2		Z	AC	CY
8 位数据 传送	MOV	A, [HL+B]	2	1	4	$A \leftarrow (HL + B)$			
		[HL+B], A	2	1	-	$(HL + B) \leftarrow A$			
		A, ES:[HL+B]	3	2	5	$A \leftarrow ((ES, HL) + B)$			
		ES:[HL+B], A	3	2	-	$((ES, HL) + B) \leftarrow A$			
		A, [HL+C]	2	1	4	$A \leftarrow (HL + C)$			
		[HL+C], A	2	1	-	$(HL + C) \leftarrow A$			
		A, ES:[HL+C]	3	2	5	$A \leftarrow ((ES, HL) + C)$			
		ES:[HL+C], A	3	2	-	$((ES, HL) + C) \leftarrow A$			
		X, laddr16	3	1	4	$X \leftarrow (addr16)$			
		X, ES:laddr16	4	2	5	$X \leftarrow (ES, addr16)$			
		X, saddr	2	1	-	$X \leftarrow (saddr)$			
		B, laddr16	3	1	4	$B \leftarrow (addr16)$			
		B, ES:laddr16	4	2	5	$B \leftarrow (ES, addr16)$			
		B, saddr	2	1	-	$B \leftarrow (saddr)$			
		C, laddr16	3	1	4	$C \leftarrow (addr16)$			
		C, ES:laddr16	4	2	5	$C \leftarrow (ES, addr16)$			
	C, saddr	2	1	-	$C \leftarrow (saddr)$				
	ES, saddr	3	1	-	$ES \leftarrow (saddr)$				
	XCH	A, r ^{#3}	1 (r=X) 2 (r=X以外)	1	-	$A \leftrightarrow r$			
		A, laddr16	4	2	-	$A \leftrightarrow (addr16)$			
		A, ES:laddr16	5	3	-	$A \leftrightarrow (ES, addr16)$			
		A, saddr	3	2	-	$A \leftrightarrow (saddr)$			
		A, sfr	3	2	-	$A \leftrightarrow sfr$			
		A, [DE]	2	2	-	$A \leftrightarrow (DE)$			
		A, ES:[DE]	3	3	-	$A \leftrightarrow (ES, DE)$			
		A, [HL]	2	2	-	$A \leftrightarrow (HL)$			
		A, ES:[HL]	3	3	-	$A \leftrightarrow (ES, HL)$			
		A, [DE+byte]	3	2	-	$A \leftrightarrow (DE + byte)$			
A, ES:[DE+byte]		4	3	-	$A \leftrightarrow ((ES, DE) + byte)$				
A, [HL+byte]		3	2	-	$A \leftrightarrow (HL + byte)$				
A, ES:[HL+byte]	4	3	-	$A \leftrightarrow ((ES, HL) + byte)$					

注 1. 内部 RAM 区域、SFR 区域及扩展 SFR 区域被存取，或者未存取数据时的 CPU 时钟数(fCLK)。

2. 程序存储器区域被存取时的 CPU 时钟数(fCLK)。

3. 除 r = A 以外。

备注 时钟数是指程序存储于内部 ROM (闪存) 区域时的时钟数。如果从内部 RAM 区域获取指令，最大为该值的 2 倍加 3 个时钟。

表31-5. 操作列表(4/17)

指令组	助记符	操作数	字节	时钟		时钟	标志			
				注1	注2		Z	AC	CY	
8 位数据 传送	XCH	A, [HL+B]	2	2	-	A ←→ (HL+B)				
		A, ES:[HL+B]	3	3	-	A ←→ ((ES, HL)+B)				
		A, [HL+C]	2	2	-	A ←→ (HL+C)				
		A, ES:[HL+C]	3	3	-	A ←→ ((ES, HL)+C)				
	ONEB	A	1	1	-	A ← 01H				
		X	1	1	-	X ← 01H				
		B	1	1	-	B ← 01H				
		C	1	1	-	C ← 01H				
		!addr16	3	1	-	(addr16) ← 01H				
		ES:!addr16	4	2	-	(ES, addr16) ← 01H				
		saddr	2	1	-	(saddr) ← 01H				
	CLR B	A	1	1	-	A ← 00H				
		X	1	1	-	X ← 00H				
		B	1	1	-	B ← 00H				
		C	1	1	-	C ← 00H				
		!addr16	3	1	-	(addr16) ← 00H				
		ES:!addr16	4	2	-	(ES, addr16) ← 00H				
		saddr	2	1	-	(saddr) ← 00H				
	MOVS	[HL+byte], X	3	1	-	(HL+byte) ← X	x		x	
		ES:[HL+byte], X	4	2	-	(ES, HL+byte) ← X	x		x	
	16 位数据 传送	MOVW	rp, #word	3	1	-	rp ← word			
			saddrp, #word	4	1	-	(saddrp) ← word			
sfrp, #word			4	1	-	sfrp ← word				
AX, rp ^{※3}			1	1	-	AX ← rp				
rp, AX ^{※3}			1	1	-	rp ← AX				
AX, !addr16			3	1	4	AX ← (addr16)				
!addr16, AX			3	1	-	(addr16) ← AX				
AX, ES:!addr16			4	2	5	AX ← (ES, addr16)				
ES:!addr16, AX			4	2	-	(ES, addr16) ← AX				
AX, saddrp			2	1	-	AX ← (saddrp)				
saddrp, AX			2	1	-	(saddrp) ← AX				
AX, sfrp			2	1	-	AX ← sfrp				
sfrp, AX			2	1	-	sfrp ← AX				

注 1. 内部 RAM 区域、SFR 区域及扩展 SFR 区域被存取，或者未存取数据时的 CPU 时钟数(fCLK)。

2. 程序存储器区域被存取时的 CPU 时钟数(fCLK)。

3. 除 rp = AX 以外。

备注 时钟数是指程序存储于内部 ROM (闪存) 区域时的时钟数。如果从内部 RAM 区域获取指令，最大为该值的 2 倍加 3 个时钟。

表31-5. 操作列表(5/17)

指令组	助记符	操作数	字节	时钟		时钟	标志		
				注1	注2		Z	AC	CY
16 位数 数据传送	MOVW	AX, [DE]	1	1	4	AX ← (DE)			
		[DE], AX	1	1	-	(DE) ← AX			
		AX, ES:[DE]	2	2	5	AX ← (ES, DE)			
		ES:[DE], AX	2	2	-	(ES, DE) ← AX			
		AX, [HL]	1	1	4	AX ← (HL)			
		[HL], AX	1	1	-	(HL) ← AX			
		AX, ES:[HL]	2	2	5	AX ← (ES, HL)			
		ES:[HL], AX	2	2	-	(ES, HL) ← AX			
		AX, [DE+byte]	2	1	4	AX ← (DE+byte)			
		[DE+byte], AX	2	1	-	(DE+byte) ← AX			
		AX, ES:[DE+byte]	3	2	5	AX ← ((ES, DE) + byte)			
		ES:[DE+byte], AX	3	2	-	((ES, DE) + byte) ← AX			
		AX, [HL+byte]	2	1	4	AX ← (HL + byte)			
		[HL+byte], AX	2	1	-	(HL + byte) ← AX			
		AX, ES:[HL+byte]	3	2	5	AX ← ((ES, HL) + byte)			
		ES:[HL+byte], AX	3	2	-	((ES, HL) + byte) ← AX			
		AX, [SP+byte]	2	1	-	AX ← (SP + byte)			
		[SP+byte], AX	2	1	-	(SP + byte) ← AX			
		AX, word[B]	3	1	4	AX ← (B + word)			
		word[B], AX	3	1	-	(B+ word) ← AX			
		AX, ES:word[B]	4	2	5	AX ← ((ES, B) + word)			
		ES:word[B], AX	4	2	-	((ES, B) + word) ← AX			
		AX, word[C]	3	1	4	AX ← (C + word)			
		word[C], AX	3	1	-	(C + word) ← AX			
		AX, ES:word[C]	4	2	5	AX ← ((ES, C) + word)			
		ES:word[C], AX	4	2	-	((ES, C) + word) ← AX			
		AX, word[BC]	3	1	4	AX ← (BC + word)			
		word[BC], AX	3	1	-	(BC + word) ← AX			
AX, ES:word[BC]	4	2	5	AX ← ((ES, BC) + word)					
ES:word[BC], AX	4	2	-	((ES, BC) + word) ← AX					

注 1. 内部 RAM 区域、SFR 区域及扩展 SFR 区域被存取，或者未存取数据时的 CPU 时钟数(f_{CLK})。

2. 程序存储器区域被存取时的 CPU 时钟数(f_{CLK})。

备注 时钟数是指程序存储于内部 ROM（闪存）区域时的时钟数。如果从内部 RAM 区域获取指令，最大为该值的 2 倍加 3 个时钟。

表31-5. 操作列表(6/17)

指令组	助记符	操作数	字节	时钟		时钟	标志		
				注1	注2		Z	AC	CY
16 位数 数据传送	MOVW	BC, laddr16	3	1	4	BC ← (addr16)			
		BC, ES:laddr16	4	2	5	BC ← (ES, addr16)			
		DE, laddr16	3	1	4	DE ← (addr16)			
		DE, ES:laddr16	4	2	5	DE ← (ES, addr16)			
		HL, laddr16	3	1	4	HL ← (addr16)			
		HL, ES:laddr16	4	2	5	HL ← (ES, addr16)			
		BC, saddrp	2	1	-	BC ← (saddrp)			
		DE, saddrp	2	1	-	DE ← (saddrp)			
		HL, saddrp	2	1	-	HL ← (saddrp)			
	XCHW	AX, rp ^{注3}	1	1	-	AX ↔ rp			
	ONEW	AX	1	1	-	AX ← 0001H			
		BC	1	1	-	BC ← 0001H			
	CLRW	AX	1	1	-	AX ← 0000H			
		BC	1	1	-	BC ← 0000H			
8 位操作	ADD	A, #byte	2	1	-	A, CY ← A + byte	x	x	x
		saddr, #byte	3	2	-	(saddr), CY ← (saddr)+byte	x	x	x
		A, r ^{注4}	2	1	-	A, CY ← A + r	x	x	x
		r, A	2	1	-	r, CY ← r + A	x	x	x
		A, laddr16	3	1	4	A, CY ← A + (addr16)	x	x	x
		A, ES:laddr16	4	2	5	A, CY ← A + (ES, addr16)	x	x	x
		A, saddr	2	1	-	A, CY ← A + (saddr)	x	x	x
		A, [HL]	1	1	4	A, CY ← A + (HL)	x	x	x
		A, ES:[HL]	2	2	5	A, CY ← A + (ES, HL)	x	x	x
		A, [HL+byte]	2	1	4	A, CY ← A + (HL+byte)	x	x	x
		A, ES:[HL+byte]	3	2	5	A, CY ← A + ((ES, HL)+byte)	x	x	x
		A, [HL+B]	2	1	4	A, CY ← A + (HL+B)	x	x	x
		A, ES:[HL+B]	3	2	5	A, CY ← A + ((ES, HL)+B)	x	x	x
		A, [HL+C]	2	1	4	A, CY ← A + (HL+C)	x	x	x
A, ES:[HL+C]	3	2	5	A, CY ← A + ((ES, HL) + C)	x	x	x		

注 1. 内部 RAM 区域、SFR 区域及扩展 SFR 区域被存取，或者未存取数据时的 CPU 时钟数(fCLK)。

2. 程序存储器区域被存取时的 CPU 时钟数(fCLK)。
3. 除 rp = AX 以外。
4. 除 r = A 以外。

备注 时钟数是指程序存储于内部 ROM (闪存) 区域时的时钟数。如果从内部 RAM 区域获取指令，最大为该值的 2 倍加 3 个时钟。

表31-5. 操作列表(7/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注1	注2		Z	AC	CY
8 位操作	ADDC	A, #byte	2	1	-	A, CY ← A+byte+CY	x	x	x
		saddr, #byte	3	2	-	(saddr), CY ← (saddr) +byte+CY	x	x	x
		A, r ^{注3}	2	1	-	A, CY ← A + r + CY	x	x	x
		r, A	2	1	-	r, CY ← r + A + CY	x	x	x
		A, laddr16	3	1	4	A, CY ← A + (addr16)+CY	x	x	x
		A, ES:laddr16	4	2	5	A, CY ← A + (ES, addr16)+CY	x	x	x
		A, saddr	2	1	-	A, CY ← A + (saddr)+CY	x	x	x
		A, [HL]	1	1	4	A, CY ← A+ (HL) + CY	x	x	x
		A, ES:[HL]	2	2	5	A,CY ← A+ (ES, HL) + CY	x	x	x
		A, [HL+byte]	2	1	4	A, CY ← A+ (HL+byte) + CY	x	x	x
		A, ES:[HL+byte]	3	2	5	A,CY ← A+ ((ES, HL)+byte) + CY	x	x	x
		A, [HL+B]	2	1	4	A, CY ← A+ (HL+B) +CY	x	x	x
		A, ES:[HL+B]	3	2	5	A,CY ← A+((ES, HL)+B)+CY	x	x	x
		A, [HL+C]	2	1	4	A, CY ← A+ (HL+C)+CY	x	x	x
	A, ES:[HL+C]	3	2	5	A,CY ← A+ ((ES, HL)+C)+CY	x	x	x	
	SUB	A, #byte	2	1	-	A, CY ← A - byte	x	x	x
		saddr, #byte	3	2	-	(saddr), CY ← (saddr) - byte	x	x	x
		A, r ^{注3}	2	1	-	A, CY ← A - r	x	x	x
		r, A	2	1	-	r, CY ← r - A	x	x	x
		A, laddr16	3	1	4	A, CY ← A - (addr16)	x	x	x
		A, ES:laddr16	4	2	5	A, CY ← A - (ES, addr16)	x	x	x
		A, saddr	2	1	-	A, CY ← A - (saddr)	x	x	x
		A, [HL]	1	1	4	A, CY ← A - (HL)	x	x	x
		A, ES:[HL]	2	2	5	A,CY ← A - (ES, HL)	x	x	x
		A, [HL+byte]	2	1	4	A, CY ← A - (HL+byte)	x	x	x
		A, ES:[HL+byte]	3	2	5	A,CY ← A - ((ES, HL)+byte)	x	x	x
		A, [HL+B]	2	1	4	A, CY ← A - (HL+B)	x	x	x
		A, ES:[HL+B]	3	2	5	A,CY ← A - ((ES, HL)+B)	x	x	x
A, [HL+C]		2	1	4	A, CY ← A - (HL+C)	x	x	x	
A, ES:[HL+C]	3	2	5	A,CY ← A - ((ES, HL)+C)	x	x	x		

注 1. 内部 RAM 区域、SFR 区域及扩展 SFR 区域被存取，或者未存取数据时的 CPU 时钟数(fCLK)。

2. 程序存储器区域被存取时的 CPU 时钟数(fCLK)。

3. 除 r = A 以外。

备注 时钟数是指程序存储于内部 ROM (闪存) 区域时的时钟数。如果从内部 RAM 区域获取指令，最大为该值的 2 倍加 3 个时钟。

表31-5. 操作列表(8/17)

指令组	助记符	操作数	字节	时钟		时钟	标志		
				注1	注2		Z	AC	CY
8位操作	SUBC	A, #byte	2	1	-	A, CY ← A - byte - CY	×	×	×
		saddr, #byte	3	2	-	(saddr), CY ← (saddr) - byte - CY	×	×	×
		A, r ^{注3}	2	1	-	A, CY ← A - r - CY	×	×	×
		r, A	2	1	-	r, CY ← r - A - CY	×	×	×
		A, laddr16	3	1	4	A, CY ← A - (addr16) - CY	×	×	×
		A, ES:laddr16	4	2	5	A, CY ← A - (ES, addr16) - CY	×	×	×
		A, saddr	2	1	-	A, CY ← A - (saddr) - CY	×	×	×
		A, [HL]	1	1	4	A, CY ← A - (HL) - CY	×	×	×
		A, ES:[HL]	2	2	5	A, CY ← A - (ES, HL) - CY	×	×	×
		A, [HL+byte]	2	1	4	A, CY ← A - (HL+byte) - CY	×	×	×
		A, ES:[HL+byte]	3	2	5	A, CY ← A - ((ES, HL)+byte) - CY	×	×	×
		A, [HL+B]	2	1	4	A, CY ← A - (HL+B) - CY	×	×	×
		A, ES:[HL+B]	3	2	5	A, CY ← A - ((ES, HL)+B) - CY	×	×	×
		A, [HL+C]	2	1	4	A, CY ← A - (HL+C) - CY	×	×	×
		A, ES:[HL+C]	3	2	5	A, CY ← A - ((ES:HL)+C) - CY	×	×	×
	AND	A, #byte	2	1	-	A ← A ∧ byte	×		
		saddr, #byte	3	2	-	(saddr) ← (saddr) ∧ byte	×		
		A, r ^{注3}	2	1	-	A ← A ∧ r	×		
		r, A	2	1	-	R ← r ∧ A	×		
		A, laddr16	3	1	4	A ← A ∧ (addr16)	×		
		A, ES:laddr16	4	2	5	A ← A ∧ (ES:addr16)	×		
		A, saddr	2	1	-	A ← A ∧ (saddr)	×		
		A, [HL]	1	1	4	A ← A ∧ (HL)	×		
		A, ES:[HL]	2	2	5	A ← A ∧ (ES:HL)	×		
		A, [HL+byte]	2	1	4	A ← A ∧ (HL+byte)	×		
		A, ES:[HL+byte]	3	2	5	A ← A ∧ ((ES:HL)+byte)	×		
		A, [HL+B]	2	1	4	A ← A ∧ (HL+B)	×		
		A, ES:[HL+B]	3	2	5	A ← A ∧ ((ES:HL)+B)	×		
A, [HL+C]	2	1	4	A ← A ∧ (HL+C)	×				
A, ES:[HL+C]	3	2	5	A ← A ∧ ((ES:HL)+C)	×				

注 1. 内部 RAM 区域、SFR 区域及扩展 SFR 区域被存取，或者未存取数据时的 CPU 时钟数(f_{CLK})。

2. 程序存储器区域被存取时的 CPU 时钟数(f_{CLK})。

3. 除 r = A 以外。

备注 时钟数是指程序存储于内部 ROM (闪存) 区域时的时钟数。如果从内部 RAM 区域获取指令，最大为该值的 2 倍加 3 个时钟。

表31-5. 操作列表(9/17)

指令组	助记符	操作数	字节	时钟		操作	标志		
				注1	注2		Z	AC	CY
8位操作	OR	A, #byte	2	1	-	$A \leftarrow A \vee \text{byte}$		x	
		saddr, #byte	3	2	-	$(\text{saddr}) \leftarrow (\text{saddr}) \vee \text{byte}$		x	
		A, r ^{註3}	2	1	-	$A \leftarrow A \vee r$		x	
		r, A	2	1	-	$r \leftarrow r \vee A$		x	
		A, laddr16	3	1	4	$A \leftarrow A \vee (\text{addr16})$		x	
		A, ES:laddr16	4	2	5	$A \leftarrow A \vee (\text{ES:addr16})$		x	
		A, saddr	2	1	-	$A \leftarrow A \vee (\text{saddr})$		x	
		A, [HL]	1	1	4	$A \leftarrow A \vee (\text{H})$		x	
		A, ES:[HL]	2	2	5	$A \leftarrow A \vee (\text{ES:HL})$		x	
		A, [HL+byte]	2	1	4	$A \leftarrow A \vee (\text{HL}+\text{byte})$		x	
		A, ES:[HL+byte]	3	2	5	$A \leftarrow A \vee ((\text{ES:HL})+\text{byte})$		x	
		A, [HL+B]	2	1	4	$A \leftarrow A \vee (\text{HL}+\text{B})$		x	
		A, ES:[HL+B]	3	2	5	$A \leftarrow A \vee ((\text{ES:HL})+\text{B})$		x	
		A, [HL+C]	2	1	4	$A \leftarrow A \vee (\text{HL}+\text{C})$		x	
	A, ES:[HL+C]	3	2	5	$A \leftarrow A \vee ((\text{ES:HL})+\text{C})$		x		
	XOR	A, #byte	2	1	-	$A \leftarrow A \oplus \text{byte}$		x	
		saddr, #byte	3	2	-	$(\text{saddr}) \leftarrow (\text{saddr}) \oplus \text{byte}$		x	
		A, r ^{註3}	2	1	-	$A \leftarrow A \oplus r$		x	
		r, A	2	1	-	$r \leftarrow r \oplus A$		x	
		A, laddr16	3	1	4	$A \leftarrow A \oplus (\text{addr16})$		x	
		A, ES:laddr16	4	2	5	$A \leftarrow A \oplus (\text{ES:addr16})$		x	
		A, saddr	2	1	-	$A \leftarrow A \oplus (\text{saddr})$		x	
		A, [HL]	1	1	4	$A \leftarrow A \oplus (\text{HL})$		x	
		A, ES:[HL]	2	2	5	$A \leftarrow A \oplus (\text{ES:HL})$		x	
		A, [HL+byte]	2	1	4	$A \leftarrow A \oplus (\text{HL}+\text{byte})$		x	
		A, ES:[HL+byte]	3	2	5	$A \leftarrow A \oplus ((\text{ES:HL})+\text{byte})$		x	
		A, [HL+B]	2	1	4	$A \leftarrow A \oplus (\text{HL}+\text{B})$		x	
		A, ES:[HL+B]	3	2	5	$A \leftarrow A \oplus ((\text{ES:HL})+\text{B})$		x	
A, [HL+C]		2	1	4	$A \leftarrow A \oplus (\text{HL}+\text{C})$		x		
A, ES:[HL+C]	3	2	5	$A \leftarrow A \oplus ((\text{ES:HL})+\text{C})$		x			

注 1. 内部 RAM 区域、SFR 区域及扩展 SFR 区域被存取，或者未存取数据时的 CPU 时钟数(f_{CLK})。

2. 程序存储器区域被存取时的 CPU 时钟数(f_{CLK})。

3. 除 r = A 以外。

备注 时钟数是指程序存储于内部 ROM (闪存) 区域时的时钟数。如果从内部 RAM 区域获取指令，最大为该值的 2 倍加 3 个时钟。

表31-5. 操作列表(10/17)

指令组	助记符	操作数	字节	时钟		时钟	标志		
				注1	注2		Z	AC	CY
8 位操作	CMP	A, #byte	2	1	-	A - byte	x	x	x
		laddr16, #byte	4	1	4	(addr16) - byte	x	x	x
		ES:laddr16, #byte	5	2	5	(ES:addr16) - byte	x	x	x
		saddr, #byte	3	1	-	(saddr) - byte	x	x	x
		A, r ^{注3}	2	1	-	A - r	x	x	x
		r, A	2	1	-	r - A	x	x	x
		A, laddr16	3	1	4	A - (addr16)	x	x	x
		A, ES:laddr16	4	2	5	A - (ES:addr16)	x	x	x
		A, saddr	2	1	-	A - (saddr)	x	x	x
		A, [HL]	1	1	4	A - (HL)	x	x	x
		A, ES:[HL]	2	2	5	A - (ES:HL)	x	x	x
		A, [HL+byte]	2	1	4	A - (HL+byte)	x	x	x
		A, ES:[HL+byte]	3	2	5	A - ((ES:HL)+byte)	x	x	x
		A, [HL+B]	2	1	4	A - (HL+B)	x	x	x
		A, ES:[HL+B]	3	2	5	A - ((ES:HL)+B)	x	x	x
		A, [HL+C]	2	1	4	A - (HL+C)	x	x	x
	A, ES:[HL+C]	3	2	5	A - ((ES:HL)+C)	x	x	x	
	CMP0	A	1	1	-	A - 00H	x	0	0
		X	1	1	-	X - 00H	x	0	0
		B	1	1	-	B - 00H	x	0	0
		C	1	1	-	C - 00H	x	0	0
		laddr16	3	1	4	(addr16) - 00H	x	0	0
		ES:laddr16	4	2	5	(ES:addr16) - 00H	x	0	0
		saddr	2	1	-	(saddr) - 00H	x	0	0
	CMPS	X, [HL+byte]	3	1	4	X - (HL+byte)	x	x	x
		X, ES:[HL+byte]	4	2	5	X - ((ES:HL)+byte)	x	x	x

注 1. 内部 RAM 区域、SFR 区域及扩展 SFR 区域被存取，或者未存取数据时的 CPU 时钟数(fCLK)。

2. 程序存储器区域被存取时的 CPU 时钟数(fCLK)。

3. 除 r = A 以外。

备注 时钟数是指程序存储于内部 ROM (闪存) 区域时的时钟数。如果从内部 RAM 区域获取指令，最大为该值的 2 倍加 3 个时钟。

表31-5. 操作列表(11/17)

指令组	助记符	操作数	字节	时钟		时钟	标志		
				注1	注2		Z	AC	CY
16 位操作	ADDW	AX, #word	3	1	-	AX, CY ← AX+word	x	x	x
		AX, AX	1	1	-	AX, CY ← AX+AX	x	x	x
		AX, BC	1	1	-	AX, CY ← AX+BC	x	x	x
		AX, DE	1	1	-	AX, CY ← AX+DE	x	x	x
		AX, HL	1	1	-	AX, CY ← AX+HL	x	x	x
		AX, !addr16	3	1	4	AX, CY ← AX+(addr16)	x	x	x
		AX, ES:!addr16	4	2	5	AX, CY ← AX+(ES:addr16)	x	x	x
		AX, saddrp	2	1	-	AX, CY ← AX+(saddrp)	x	x	x
		AX, [HL+byte]	3	1	4	AX, CY ← AX+(HL+byte)	x	x	x
		AX, ES: [HL+byte]	4	2	5	AX, CY ← AX+((ES:HL)+byte)	x	x	x
	SUBW	AX, #word	3	1	-	AX, CY ← AX - word	x	x	x
		AX, BC	1	1	-	AX, CY ← AX - BC	x	x	x
		AX, DE	1	1	-	AX, CY ← AX - DE	x	x	x
		AX, HL	1	1	-	AX, CY ← AX - HL	x	x	x
		AX, !addr16	3	1	4	AX, CY ← AX - (addr16)	x	x	x
		AX, ES:!addr16	4	2	5	AX, CY ← AX - (ES:addr16)	x	x	x
		AX, saddrp	2	1	-	AX, CY ← AX - (saddrp)	x	x	x
		AX, [HL+byte]	3	1	4	AX, CY ← AX - (HL+byte)	x	x	x
		AX, ES: [HL+byte]	4	2	5	AX, CY ← AX - ((ES:HL)+byte)	x	x	x
		CMPW	AX, #word	3	1	-	AX - word	x	x
	AX, BC		1	1	-	AX - BC	x	x	x
	AX, DE		1	1	-	AX - DE	x	x	x
	AX, HL		1	1	-	AX - HL	x	x	x
	AX, !addr16		3	1	4	AX - (addr16)	x	x	x
	AX, ES:!addr16		4	2	5	AX - (ES:addr16)	x	x	x
	AX, saddrp		2	1	-	AX - (saddrp)	x	x	x
	AX, [HL+byte]		3	1	4	AX - (HL+byte)	x	x	x
AX, ES: [HL+byte]	4		2	5	AX - ((ES:HL)+byte)	x	x	x	
乘	MULU	X	1	1	-	AX ← A×X			

注 1. 内部 RAM 区域、SFR 区域及扩展 SFR 区域被存取，或者未存取数据时的 CPU 时钟数(f_{CLK})。

2. 程序存储器区域被存取时的 CPU 时钟数(f_{CLK})。

备注 时钟数是指程序存储于内部 ROM（闪存）区域时的时钟数。如果从内部 RAM 区域获取指令，最大为该值的 2 倍加 3 个时钟。

表31-5. 操作列表(12/17)

指令组	助记符	操作数	字节	时钟		时钟	标志		
				注1	注2		Z	AC	CY
递增/递减	INC	r	1	1	-	$r \leftarrow r+1$	x	x	
		laddr16	3	2	-	$(addr16) \leftarrow (addr16)+1$	x	x	
		ES:laddr16	4	3	-	$(ES, addr16) \leftarrow (ES, addr16)+1$	x	x	
		saddr	2	2	-	$(saddr) \leftarrow (saddr)+1$	x	x	
		[HL+byte]	3	2	-	$(HL+byte) \leftarrow (HL+byte)+1$	x	x	
		ES:[HL+byte]	4	3	-	$((ES:HL)+byte) \leftarrow ((ES:HL)+byte)+1$	x	x	
	DEC	r	1	1	-	$r \leftarrow r-1$	x	x	
		laddr16	3	2	-	$(addr16) \leftarrow (addr16)-1$	x	x	
		ES:laddr16	4	3	-	$(ES, addr16) \leftarrow (ES, addr16)-1$	x	x	
		saddr	2	2	-	$(saddr) \leftarrow (saddr)-1$	x	x	
		[HL+byte]	3	2	-	$(HL+byte) \leftarrow (HL+byte)-1$	x	x	
		ES:[HL+byte]	4	3	-	$((ES:HL)+byte) \leftarrow ((ES:HL)+byte)-1$	x	x	
	INCW	rp	1	1	-	$rp \leftarrow rp+1$			
		laddr16	3	2	-	$(addr16) \leftarrow (addr16)+1$			
		ES:laddr16	4	3	-	$(ES, addr16) \leftarrow (ES, addr16)+1$			
		saddrp	2	2	-	$(saddrp) \leftarrow (saddrp)+1$			
		[HL+byte]	3	2	-	$(HL+byte) \leftarrow (HL+byte)+1$			
		ES:[HL+byte]	4	3	-	$((ES:HL)+byte) \leftarrow ((ES:HL)+byte)+1$			
	DECW	rp	1	1	-	$rp \leftarrow rp-1$			
		laddr16	3	2	-	$(addr16) \leftarrow (addr16)-1$			
		ES:laddr16	4	3	-	$(ES, addr16) \leftarrow (ES, addr16)-1$			
		saddrp	2	2	-	$(saddrp) \leftarrow (saddrp)-1$			
		[HL+byte]	3	2	-	$(HL+byte) \leftarrow (HL+byte)-1$			
		ES:[HL+byte]	4	3	-	$((ES:HL)+byte) \leftarrow ((ES:HL)+byte)-1$			
移位	SHR	A, cnt	2	1	-	$(CY \leftarrow A_0, A_{m-1} \leftarrow A_m, A_7 \leftarrow 0) \times cnt$			x
	SHRW	AX, cnt	2	1	-	$(CY \leftarrow AX_0, AX_{m-1} \leftarrow AX_m, AX_{15} \leftarrow 0) \times cnt$			x
	SHL	A, cnt	2	1	-	$(CY \leftarrow A_7, A_m \leftarrow A_{m-1}, A_0 \leftarrow 0) \times cnt$			x
		B, cnt	2	1	-	$(CY \leftarrow B_7, B_m \leftarrow B_{m-1}, B_0 \leftarrow 0) \times cnt$			x
		C, cnt	2	1	-	$(CY \leftarrow C_7, C_m \leftarrow C_{m-1}, C_0 \leftarrow 0) \times cnt$			x
	SHLW	AX, cnt	2	1	-	$(CY \leftarrow AX_{15}, AX_m \leftarrow AX_{m-1}, AX_0 \leftarrow 0) \times cnt$			x
		BC, cnt	2	1	-	$(CY \leftarrow BC_{15}, BC_m \leftarrow BC_{m-1}, BC_0 \leftarrow 0) \times cnt$			x
	SAR	A, cnt	2	1	-	$(CY \leftarrow A_0, A_{m-1} \leftarrow A_m, A_7 \leftarrow A_7) \times cnt$			x
SARW	AX, cnt	2	1	-	$(CY \leftarrow AX_0, AX_{m-1} \leftarrow AX_m, AX_{15} \leftarrow AX_{15}) \times cnt$			x	

注 1. 内部 RAM 区域、SFR 区域及扩展 SFR 区域被存取，或者未存取数据时的 CPU 时钟数(fCLK)。

2. 程序存储器区域被存取时的 CPU 时钟数(fCLK)。

备注 1. 时钟数是指程序存储于内部 ROM（闪存）区域时的时钟数。如果从内部 RAM 区域获取指令，最大为该值的 2 倍加 3 个时钟。

2. cnt 为移位位数。

表31-5. 操作列表(13/17)

指令组	助记符	操作数	字节	时钟		时钟	标志		
				注1	注2		Z	AC	CY
循环运算	ROR	A, 1	2	1	-	$(CY, A_7 \leftarrow A_0, A_{m-1} \leftarrow A_m) \times 1$			x
	ROL	A, 1	2	1	-	$(CY, A_0 \leftarrow A_7, A_{m+1} \leftarrow A_m) \times 1$			x
	RORC	A, 1	2	1	-	$(CY \leftarrow A_0, A_7 \leftarrow CY, A_{m-1} \leftarrow A_m) \times 1$			x
	ROLC	A, 1	2	1	-	$(CY \leftarrow A_7, A_0 \leftarrow CY, A_{m+1} \leftarrow A_m) \times 1$			x
	ROLWC	AX,1	2	1	-	$(CY \leftarrow AX_{15}, AX_0 \leftarrow CY, AX_{m+1} \leftarrow AX_m) \times 1$			x
		BC,1	2	1	-	$(CY \leftarrow BC_{15}, BC_0 \leftarrow CY, BC_{m+1} \leftarrow BC_m) \times 1$			x
位操作	MOV1	CY, A.bit	2	1	-	$CY \leftarrow A.bit$			x
		A.bit, CY	2	1	-	$A.bit \leftarrow CY$			
		CY, PSW.bit	3	1	-	$CY \leftarrow PSW.bit$			x
		PSW.bit, CY	3	4	-	$PSW.bit \leftarrow CY$	x	x	
		CY, saddr.bit	3	1	-	$CY \leftarrow (saddr).bit$			x
		saddr.bit, CY	3	2	-	$(saddr).bit \leftarrow CY$			
		CY, sfr.bit	3	1	-	$CY \leftarrow sfr.bit$			x
		sfr.bit, CY	3	2	-	$sfr.bit \leftarrow CY$			
		CY, [HL].bit	2	1	4	$CY \leftarrow (HL).bit$			x
		[HL].bit, CY	2	2	-	$(HL).bit \leftarrow CY$			
	CY, ES:[HL].bit	3	2	5	$CY \leftarrow (ES, HL).bit$			x	
	ES:[HL].bit, CY	3	3	-	$(ES, HL).bit \leftarrow CY$				
	AND1	CY, A.bit	2	1	-	$CY \leftarrow CY \wedge A.bit$			x
		CY, PSW.bit	3	1	-	$CY \leftarrow CY \wedge PSW.bit$			x
		CY, saddr.bit	3	1	-	$CY \leftarrow CY \wedge (saddr).bit$			x
		CY, sfr.bit	3	1	-	$CY \leftarrow CY \wedge sfr.bit$			x
		CY, [HL].bit	2	1	4	$CY \leftarrow CY \wedge (HL).bit$			x
		CY, ES:[HL].bit	3	2	5	$CY \leftarrow CY \wedge (ES, HL).bit$			x
	OR1	CY, A.bit	2	1	-	$CY \leftarrow CY \vee A.bit$			x
		CY, PSW.bit	3	1	-	$CY \leftarrow CY \vee PSW.bit$			x
		CY, saddr.bit	3	1	-	$CY \leftarrow CY \vee (saddr).bit$			x
		CY, sfr.bit	3	1	-	$CY \leftarrow CY \vee sfr.bit$			x
		CY, [HL].bit	2	1	4	$CY \leftarrow CY \vee (HL).bit$			x
		CY, ES:[HL].bit	3	2	5	$CY \leftarrow CY \vee (ES, HL).bit$			x

注 1. 内部 RAM 区域、SFR 区域及扩展 SFR 区域被存取，或者未存取数据时的 CPU 时钟数(fCLK)。

2. 程序存储器区域被存取时的 CPU 时钟数(fCLK)。

备注 时钟数是指程序存储于内部 ROM (闪存) 区域时的时钟数。如果从内部 RAM 区域获取指令，最大为该值的 2 倍加 3 个时钟。

表31-5. 操作列表(14/17)

指令组	助记符	操作数	字节	时钟		时钟	标志		
				注1	注2		Z	AC	CY
位操作	XOR1	CY, A.bit	2	1	-	CY ← CY ∨ A.bit			×
		CY, PSW.bit	3	1	-	CY ← CY ∨ PSW.bit			×
		CY, saddr.bit	3	1	-	CY ← CY ∨ (saddr).bit			×
		CY, sfr.bit	3	1	-	CY ← CY ∨ sfr.bit			×
		CY, [HL].bit	2	1	4	CY ← CY ∨ (HL).bit			×
		CY, ES:[HL].bit	3	2	5	CY ← CY ∨ (ES, HL).bit			×
	SET1	A.bit	2	1	-	A.bit ← 1			
		PSW.bit	3	4	-	PSW.bit ← 1	×	×	×
		laddr16.bit	4	2	-	(addr16).bit ← 1			
		ES:laddr16.bit	5	3	-	(ES, addr16).bit ← 1			
		saddr.bit	3	2	-	(saddr).bit ← 1			
		sfr.bit	3	2	-	sfr.bit ← 1			
		[HL].bit	2	2	-	(HL).bit ← 1			
		ES:[HL].bit	3	3	-	(ES, HL).bit ← 1			
	CLR1	A.bit	2	1	-	A.bit ← 0			
		PSW.bit	3	4	-	PSW.bit ← 0	×	×	×
		laddr16.bit	4	2	-	(addr16).bit ← 0			
		ES:laddr16.bit	5	3	-	(ES, addr16).bit ← 0			
		saddr.bit	3	2	-	(saddr).bit ← 0			
		sfr.bit	3	2	-	sfr.bit ← 0			
		[HL].bit	2	2	-	(HL).bit ← 0			
		ES:[HL].bit	3	3	-	(ES, HL).bit ← 0			
	SET1	CY	2	1	-	CY ← 1			1
	CLR1	CY	2	1	-	CY ← 0			0
	NOT1	CY	2	1	-	CY ← CY			×

注 1. 内部 RAM 区域、SFR 区域及扩展 SFR 区域被存取，或者未存取数据时的 CPU 时钟数(f_{CLK})。

2. 程序存储器区域被存取时的 CPU 时钟数(f_{CLK})。

备注 时钟数是指程序存储于内部 ROM（闪存）区域时的时钟数。如果从内部 RAM 区域获取指令，最大为该值的 2 倍加 3 个时钟。

表31-5. 操作列表(15/17)

指令组	助记符	操作数	字节	时钟		时钟	标志		
				注1	注2		Z	AC	CY
调用 / 返回	CALL	rp	2	3	-	(SP - 2) ← (PC+2) _s , (SP - 3) ← (PC+2) _H , (SP - 4) ← (PC+2) _L , PC ← CS, rp, SP ← SP - 4			
		\$!addr20	3	3	-	(SP - 2) ← (PC+3) _s , (SP - 3) ← (PC+3) _H , (SP - 4) ← (PC+3) _L , PC ← PC+3+jdisp16, SP ← SP - 4			
		laddr16	3	3	-	(SP - 2) ← (PC+3) _s , (SP - 3) ← (PC+3) _H , (SP - 4) ← (PC+3) _L , PC ← 0000, addr16, SP ← SP - 4			
		!!addr20	4	3	-	(SP - 2) ← (PC+4) _s , (SP - 3) ← (PC+4) _H , (SP - 4) ← (PC+4) _L , PC ← addr20, SP ← SP - 4			
	CALLT	[addr5]	2	5	-	(SP - 2) ← (PC+2) _s , (SP - 3) ← (PC+2) _H , (SP - 4) ← (PC+2) _L , PC _s ← 0000, PC _H ← (0000, addr5+1), PC _L ← (0000, addr5), SP ← SP - 4			
	BRK	-	2	5	-	(SP - 1) ← PSW, (SP - 2) ← (PC+2) _s , (SP - 3) ← (PC+2) _H , (SP - 4) ← (PC+2) _L , PC _s ← 0000, PC _H ← (0007FH), PC _L ← (0007EH), SP ← SP - 4, IE ← 0			
	RET	-	1	6	-	PC _L ← (SP), PC _H ← (SP+1), PC _s ← (SP+2), SP ← SP+4			
RETI	-	2	6	-	PC _L ← (SP), PC _H ← (SP+1), PC _s ← (SP+2), PSW ← (SP+3), SP ← SP+4	R	R	R	
RETB	-	2	6	-	PC _L ← (SP), PC _H ← (SP+1), PC _s ← (SP+2), PSW ← (SP+3), SP ← SP+4	R	R	R	

注 1. 内部 RAM 区域、SFR 区域及扩展 SFR 区域被存取，或者未存取数据时的 CPU 时钟数(fCLK)。

2. 程序存储器区域被存取时的 CPU 时钟数(fCLK)。

备注 时钟数是指程序存储于内部 ROM（闪存）区域时的时钟数。如果从内部 RAM 区域获取指令，最大为该值的 2 倍加 3 个时钟。

表31-5. 操作列表(16/17)

指令组	助记符	操作数	字节	时钟		时钟	标志		
				注1	注2		Z	AC	CY
堆栈操作	PUSH	PSW	2	1	-	(SP - 1) ← PSW, (SP - 2) ← 00H, SP ← SP - 2			
		rp	1	1	-	(SP - 1) ← rpH, (SP - 2) ← rpL, SP ← SP - 2			
	POP	PSW	2	3	-	PSW ← (SP+1), SP ← SP + 2	R	R	R
		rp	1	1	-	rpL ← (SP), rpH ← (SP+1), SP ← SP + 2			
	MOVW	SP, #word	4	1	-	SP ← word			
		SP, AX	2	1	-	SP ← AX			
		AX, SP	2	1	-	AX ← SP			
		HL, SP	3	1	-	HL ← SP			
		BC, SP	3	1	-	BC ← SP			
		DE, SP	3	1	-	DE ← SP			
	ADDW	SP, #byte	2	1	-	SP ← SP + byte			
SUBW	SP, #byte	2	1	-	SP ← SP - byte				
无条件分支	BR	AX	2	3	-	PC ← CS, AX			
		\$addr20	2	3	-	PC ← PC + 2 + jdisp8			
		!addr20	3	3	-	PC ← PC + 3 + jdisp16			
		laddr16	3	3	-	PC ← 0000, addr16			
		!!addr20	4	3	-	PC ← addr20			
有条件分支	BC	\$addr20	2	2/4 ^{注3}	-	PC ← PC + 2 + jdisp8 if CY = 1			
	BNC	\$addr20	2	2/4 ^{注3}	-	PC ← PC + 2 + jdisp8 if CY = 0			
	BZ	\$addr20	2	2/4 ^{注3}	-	PC ← PC + 2 + jdisp8 if Z = 1			
	BNZ	\$addr20	2	2/4 ^{注3}	-	PC ← PC + 2 + jdisp8 if Z = 0			
	BH	\$addr20	3	2/4 ^{注3}	-	PC ← PC + 3 + jdisp8 if (Z∨CY)=0			
	BNH	\$addr20	3	2/4 ^{注3}	-	PC ← PC + 3 + jdisp8 if (Z∨CY)=1			
	BT	saddr.bit, \$addr20	4	3/5 ^{注3}	-	PC ← PC + 4 + jdisp8 if (saddr).bit = 1			
		sfr.bit, \$addr20	4	3/5 ^{注3}	-	PC ← PC + 4 + jdisp8 if sfr.bit = 1			
		A.bit, \$addr20	3	3/5 ^{注3}	-	PC ← PC + 3 + jdisp8 if A.bit = 1			
		PSW.bit, \$addr20	4	3/5 ^{注3}	-	PC ← PC + 4 + jdisp8 if PSW.bit = 1			
		[HL].bit, \$addr20	3	3/5 ^{注3}	6/7	PC ← PC + 3 + jdisp8 if (HL).bit = 1			
ES:[HL].bit, \$addr20	4	4/6 ^{注3}	7/8	PC ← PC + 4 + jdisp8 if (ES, HL).bit = 1					

注 1. 内部 RAM 区域、SFR 区域及扩展 SFR 区域被存取，或者未存取数据时的 CPU 时钟数(fCLK)。

2. 程序存储器区域被存取时的 CPU 时钟数(fCLK)。

3. 表示“当不符合条件/符合条件时”的时钟数。

备注 时钟数是指程序存储于内部 ROM（闪存）区域时的时钟数。如果从内部 RAM 区域获取指令，最大为该值的 2 倍加 3 个时钟。

表31-5. 操作列表(17/17)

指令组	助记符	操作数	字节	时钟		时钟	标志		
				注1	注2		Z	AC	CY
有条件分支	BF	saddr.bit, \$addr20	4	3/5 ^{#3}	–	PC ← PC + 4 + jdisp8 if (saddr).bit = 0			
		sfr.bit, \$addr20	4	3/5 ^{#3}	–	PC ← PC + 4 + jdisp8 if sfr.bit = 0			
		A.bit, \$addr20	3	3/5 ^{#3}	–	PC ← PC + 3 + jdisp8 if A.bit = 0			
		PSW.bit, \$addr20	4	3/5 ^{#3}	–	PC ← PC + 4 + jdisp8 if PSW.bit = 0			
		[HL].bit, \$addr20	3	3/5 ^{#3}	6/7	PC ← PC + 3 + jdisp8 if (HL).bit = 0			
		ES:[HL].bit, \$addr20	4	4/6 ^{#3}	7/8	PC ← PC + 4 + jdisp8 if (ES, HL).bit = 0			
	BTCLR	saddr.bit, \$addr20	4	3/5 ^{#3}	–	PC ← PC + 4 + jdisp8 if (saddr).bit = 1 then reset (saddr).bit			
		sfr.bit, \$addr20	4	3/5 ^{#3}	–	PC ← PC + 4 + jdisp8 if sfr.bit = 1 then reset sfr.bit			
		A.bit, \$addr20	3	3/5 ^{#3}	–	PC ← PC + 3 + jdisp8 if A.bit = 1 then reset A.bit			
		PSW.bit, \$addr20	4	3/5 ^{#3}	–	PC ← PC + 4 + jdisp8 if PSW.bit = 1 then reset PSW.bit	×	×	×
		[HL].bit, \$addr20	3	3/5 ^{#3}	–	PC ← PC + 3 + jdisp8 if (HL).bit = 1 then reset (HL).bit			
		ES:[HL].bit, \$addr20	4	4/6 ^{#3}	–	PC ← PC + 4 + jdisp8 if (ES, HL).bit = 1 then reset (ES, HL).bit			
有条件跳跃	SKC	–	2	1	–	Next instruction skip if CY = 1			
	SKNC	–	2	1	–	Next instruction skip if CY = 0			
	SKZ	–	2	1	–	Next instruction skip if Z = 1			
	SKNZ	–	2	1	–	Next instruction skip if Z = 0			
	SKH	–	2	1	–	Next instruction skip if (Z∨CY)=0			
	SKNH	–	2	1	–	Next instruction skip if (Z∨CY)=1			
CP 控制	SEL ^{#4}	Rbn	2	1	–	RBS[1:0] ← n			
	NOP	–	1	1	–	No Operation			
	EI	–	3	4	–	IE ← 1 (Enable Interrupt)			
	DI	–	3	4	–	IE ← 0 (Disable Interrupt)			
	HALT	–	2	3	–	Set HALT Mode			
	STOP	–	2	3	–	Set STOP Mode			

注 1. 内部 RAM 区域、SFR 区域及扩展 SFR 区域被存取，或者未存取数据时的 CPU 时钟数(fCLK)。

2. 程序存储器区域被存取时的 CPU 时钟数(fCLK)。
3. 表示“当不符合条件/符合条件时”的时钟数。

备注 时钟数是指程序存储于内部 ROM（闪存）区域时的时钟数。如果从内部 RAM 区域获取指令，最大为该值的 2 倍加 3 个时钟。

第三十二章 电特性

- 注意事项
1. **RL78/I1A** 单片机具有用于开发和评价的片上调试功能。对于量产产品，不得使用片上调试功能，因为在使用该功能时，可能会超过闪存的保证可重写次数，因而会使产品可靠性失去保障。对于使用片上调试功能时发生的问题，瑞萨电子不承担任何责任。
 2. 配置的引脚因产品而异。请参阅 2.1 端口功能至 2.2.1 各种产品的配置功能。

32.1 最大绝对额定值

最大绝对额定值 (T_A = 25°C) (1/2)

参数	符号	条件	额定值	单元
电源电压	V _{DD}		-0.5 至 +6.5	V
	V _{SS}		-0.5 至 +0.3	V
REGC 引脚输入电压	V _{IREGC}	REGC	-0.3 至 +2.8 和 -0.3 至 V _{DD} + 0.3 ^{注1}	V
输入电压	V _{I1}	P02, P03, P05, P06, P10 至 P12, P20 至 P22, P24 至 P27, P30, P31, P40, P75 至 P77, P120 至 P124, P137, P147, P200 至 P206, EXCLK, EXCLKS, $\overline{\text{RESET}}$	-0.3 至 V _{DD} + 0.3 ^{注2}	V
输出电压	V _{O1}	P02, P03, P05, P06, P10 至 P12, P20 至 P22, P24 至 P27, P30, P31, P40, P75 至 P77, P120, P147, P200 至 P206	-0.3 至 V _{DD} + 0.3 ^{注2}	V
模拟输入电压	V _{AI1}	ANI0 至 ANI2, ANI4 至 ANI7, ANI16 至 ANI19	-0.3 至 V _{DD} + 0.3 和 -0.3 至 AV _{REF(+)} + 0.3 ^{注2,3}	V

注 1. 通过一个电容 (0.47 至 1 μF) 将 REGC 引脚连接至 V_{SS}。该值调节 REGC 引脚的最大绝对额定值。不得在该引脚上输入电压。

2. 必须为 6.5 V 或更低。
3. A/D 转换目标引脚不可超过 AV_{REF(+)} + 0.3 V。

注意事项 任何参数如果超过最大额定值，即使只有一瞬间超过也可能损坏产品质量。也就是说，最大绝对额定值是对产品造成物理损坏的临界值，因此必须确保在不超过最大绝对额定值下使用产品。

- 备注 1. 除非另有说明，否则复用功能引脚的特性与端口引脚的特性相同。
2. AV_{REF(+)}：A/D 转换器的+侧基准电压。

最大绝对额定值 (TA = 25°C) (2/2)

参数	符号	条件		额定值	单元
输出电流, 高电位	I _{OH1}	每个引脚	P02, P03, P05, P06, P10 至 P12, P30, P31, P40, P75 至 P77, P120, P147, P200 至 P206	-40	mA
		所有引脚共计 -170 mA	P02, P03, P40, P120	-70	mA
			P05, P06, P10 至 P12, P30, P31, P75 至 P77, P147, P200 至 P206	-100	mA
	I _{OH2}	每个引脚	P20 至 P22, P24 至 P27	-0.5	mA
		所有引脚共计		-2	mA
	输出电流, 低电位	I _{OL1}	每个引脚	P02, P03, P05, P06, P10 至 P12, P30, P31, P40, P75 至 P77, P120, P147, P200 至 P206	40
所有引脚共计 170 mA			P02, P03, P40, P120	70	mA
			P05, P06, P10 至 P12, P30, P31, P75 至 P77, P147, P200 至 P206	100	mA
I _{OL2}		每个引脚	P20 至 P22, P24 至 P27	1	mA
		所有引脚共计		5	mA
运行环境温度		T _A	在普通操作模式下		-40 至 +105
	在内存编程模式下				
存储温度	T _{stg}			-65 至 +150	°C

注意事项 任何参数如果超过最大额定值, 即使只有一瞬间超过也可能损坏产品质量。也就是说, 最大绝对额定值是对产品造成物理损坏的临界值, 因此必须确保在不超过最大绝对额定值下使用产品。

备注 除非另有说明, 否则复用功能引脚的特性与端口引脚的特性相同。

32.2 振荡电路特性

32.2.1 X1, XT1 振荡电路特性

($T_A = -40$ 至 $+105^\circ\text{C}$, $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$)

参数	谐振器	推荐电路	条件	最小值	典型值	最大值	单元
X1 时钟振荡频率 (f_x) ^注	陶瓷谐振器/ 晶体谐振器			1.0		20.0	MHz
XT1 时钟振荡频率 (f_{XT1}) ^注	晶体谐振器			32	32.768	35	kHz

注 仅表示振荡电路特性。关于指令执行时间，请参阅 AC 特性。

注意事项 1. 使用 X1 振荡电路、XT1 振荡电路时，按照下面的要求对上图中虚线区域内的部分进行布线，从而避免布线电容的影响。

- 布线要保证尽可能短。
 - 振荡器布线不要和别的信号线互相交叉。
 - 请勿将振荡器的布线靠近通过大波动电流的信号线。
 - 始终保持振荡电路电容的接地点与 V_{SS} 具有相同电位。
 - 请勿将电容的接地点连接到通过大电流的接地布局。
 - 请勿从振荡电路取信号。
2. 解除复位后，使用高速片上振荡器时钟来启动 CPU，因此，用户须通过振荡稳定时间计数器状态寄存器 (OSTC) 来确认 X1 时钟振荡稳定时间。在使用的谐振器上对振荡稳定时间进行充分评估后，确定 OSTC 寄存器和振荡稳定时间选择寄存器 (OSTS) 的振荡稳定时间。
 3. 为了降低功耗，XT1 振荡电路采用低振幅电路，与 X1 振荡电路相比，更容易诱发噪声导致的误操作。因此，使用 XT1 时钟时，应特别注意布线方法。

32.2.2 片上振荡器特性

(TA = -40 至+105°C, 2.7 V ≤ VDD ≤ 5.5 V, VSS = 0 V)

振荡电路	参数	条件	最小值	典型值	最大值	单元
高速片上振荡器时钟频率 ^{注1}	f _{IH}		1		32	MHz
高速片上振荡器时钟频率精度 ^{注2}		TA = -20 至 85°C	-1		+1	%
		TA = -40 至 105°C	-1.5		+1.5	%
低速片上振荡器时钟频率	f _{IL}			15		kHz
低速片上振荡器时钟频率精度			-15		+15	%

注 1. 高速片上振荡器可以选择的频率。通过选项字节(000C2H/010C2H)的位 0 至 3 进行选择。

2. 仅表示振荡电路特性。关于指令执行时间，请参阅 AC 特性。

32.2.3 PLL特性

(TA = -40 至+105°C, 2.7 V ≤ VDD ≤ 5.5 V, VSS = 0 V)

参数	符号	条件	最小值	典型值	最大值	单元
PLL 输入时钟频率 ^注	f _{PLLIN}	选择高速系统时钟(f _{MX} = 4 MHz)	3.94	4.00	4.06	MHz
		选择高速片上振荡器时钟(f _{IH} = 4 MHz)	3.94	4.00	4.06	MHz
PLL 输出时钟频率 ^注	f _{PLL}		f _{PLLIN} × 16			MHz

注 仅表示振荡电路特性。关于指令执行时间，请参阅 AC 特性。

32.3 DC特性

32.3.1 引脚特性

(T_A = -40 至+105°C, 2.7 V ≤ V_{DD} ≤ 5.5 V, V_{SS} = 0 V)

项目	符号	条件	最小值	典型值	最大值	单元	
输出电流, 高电位 ^{注1}	I _{OH1}	P02, P03, P05, P06, P10 至 P12, P30, P31, P40, P75 至 P77, P120, P147, P200 至 P206 的任一引脚	4.0 V ≤ V _{DD} ≤ 5.5 V			-3.0 ^{注2}	mA
			2.7 V ≤ V _{DD} < 4.0 V			-1.0	mA
		P02, P03, P40, P120 的合计 (当占空比 = 70% ^{注3})	4.0 V ≤ V _{DD} ≤ 5.5 V			-12.0	mA
			2.7 V ≤ V _{DD} < 4.0 V			-4.0	mA
		P05, P06, P10 至 P12, P30, P31, P75 至 P77, P147, P200 至 P206 的合计 (当占空比 = 70% ^{注3})	4.0 V ≤ V _{DD} ≤ 5.5 V			-30.0	mA
			2.7 V ≤ V _{DD} < 4.0 V			-10.0	mA
	所有引脚的合计 (当占空比 = 70% ^{注3})	4.0 V ≤ V _{DD} ≤ 5.5 V			-30.0	mA	
		2.7 V ≤ V _{DD} < 4.0 V			-14.0	mA	
	I _{OH2}	P20 至 P22, P24 至 P27 的任一引脚	2.7 V ≤ V _{DD} ≤ 5.5 V			-0.1 ^{注2}	mA
			所有引脚的合计 (当占空比 = 70% ^{注3})	2.7 V ≤ V _{DD} ≤ 5.5 V			-0.7

注 1. 即使电流从 V_{DD} 引脚流向输出引脚时也可保证器件普通工作的电流值。

2. 不得超过总电流值。

3. 占空比为 70% 的条件下的规格。

改变了占空比的输出电流值可以用以下表达式算出 (占空比从 70% 变成 n% 时)。

• 所有引脚的输出电流共计 = (I_{OH} × 0.7)/(n × 0.01)

<示例> 当 n = 50% 和 I_{OH} = -10.0 mA 时

所有引脚的输出电流共计 = (-10.0 × 0.7)/(50 × 0.01) = -14.0 mA

然而, 允许流进引脚的电流并不因占空比而变化。不要使高于最大绝对额定值的电流不得流入引脚。

注意事项 在 N 沟开漏模式下, P02、P10 至 P12 不输出高电位。

备注 除非另有说明, 否则复用功能引脚的特性与端口引脚的特性相同。

(TA = -40 至+105°C, 2.7 V ≤ VDD ≤ 5.5 V, VSS = 0 V)

项目	符号	条件	最小值	典型值	最大值	单元
输出电流, 低电平 ^{#1}	I _{OL1}	P02, P03, P05, P06, P10 至 P12, P30, P31, P40, P75 至 P77, P120, P147, P200 至 P206 的任一引脚	4.0 V ≤ V _{DD} ≤ 5.5 V		8.5 ^{#2}	mA
			2.7 V ≤ V _{DD} < 4.0 V		1.5 ^{#2}	mA
		P02, P03, P40, P120 的合计 (当占空比 = 70% ^{#3})	4.0 V ≤ V _{DD} ≤ 5.5 V		40.0	mA
			2.7 V ≤ V _{DD} < 4.0 V		7.5	mA
		P05, P06, P10 至 P12, P30, P31, P75 至 P77, P147, P200 至 P206 的合计 (当占空比 = 70% ^{#3})	4.0 V ≤ V _{DD} ≤ 5.5 V		40.0	mA
			2.7 V ≤ V _{DD} < 4.0 V		17.5	mA
	所有引脚的合计 (当占空比为 70%时 ^{#3})	4.0 V ≤ V _{DD} ≤ 5.5 V		80.0	mA	
		2.7 V ≤ V _{DD} < 4.0 V		25.0	mA	
	I _{OL2}	P20 至 P22, P24 至 P27 的任一引脚	2.7 V ≤ V _{DD} ≤ 5.5 V		0.4 ^{#2}	mA
			所有引脚的合计 (当占空比为 70%时 ^{#3})	2.7 V ≤ V _{DD} ≤ 5.5 V		2.8

注 1. 即使电流从输出引脚流向 V_{SS} 引脚时也可保证器件普通工作的电流值。

2. 不得超过总电流值。

3. 占空比为 70% 的条件下的规格。

改变了占空比的输出电流值可以用以下表达式算出 (占空比从 70% 变成 n% 时)。

- 所有引脚的输出电流共计 = (I_{OL} × 0.7)/(n × 0.01)

<示例> 当 n = 50% 和 I_{OL} = 40.0 mA

所有引脚的输出电流共计 = (40.0 × 0.7)/(50 × 0.01) = 56.0 mA

然而, 允许流进引脚的电流并不因占空比而变化。不要使高于绝对最大额定值的电流流入引脚。

备注 除非另有说明, 否则复用功能引脚的特性与端口引脚的特性相同。

(TA = -40 至+105°C, 2.7 V ≤ VDD ≤ 5.5 V, VSS = 0 V)

项目	符号	条件	最小值	典型值	最大值	单元	
输入电压, 高电平	V _{IH1}	P02, P03, P05, P06, P10 至 P12, P20 至 P22, P24 至 P27, P30, P31, P40, P75 至 P77, P120 至 P124, P137, P147, P200 至 P206, EXCLK, EXCLKS, RESET	普通输入缓冲器	0.8V _{DD}		V _{DD}	V
	V _{IH2}	P03, P10, P11	TTL 输入缓冲器 4.0 V ≤ V _{DD} ≤ 5.5 V	2.1		V _{DD}	V
			TTL 输入缓冲器 3.3 V ≤ V _{DD} < 4.0 V	2.0		V _{DD}	V
			TTL 输入缓冲器 2.7 V ≤ V _{DD} < 3.3 V	1.5		V _{DD}	V
输入电压, 低电平	V _{IL1}	P02, P03, P05, P06, P10 至 P12, P20 至 P22, P24 至 P27, P30, P31, P40, P75 至 P77, P120 至 P124, P137, P147, P200 至 P206, EXCLK, EXCLKS, RESET	普通输入缓冲器	0		0.2V _{DD}	V
	V _{IL2}	P03, P10, P11	TTL 输入缓冲器 4.0 V ≤ V _{DD} ≤ 5.5 V	0		0.8	V
			TTL 输入缓冲器 3.3 V ≤ V _{DD} < 4.0 V	0		0.5	V
			TTL 输入缓冲器 2.7 V ≤ V _{DD} < 3.3 V	0		0.32	V

注意事项 即使在 N 沟开漏模式下, P02、P10 至 P12 引脚的 V_{IH} 的最大值也为 V_{DD}。

备注 除非另有说明, 否则复用功能引脚的特性与端口引脚的特性相同。

(TA = -40 至 +105°C, 2.7 V ≤ VDD ≤ 5.5 V, VSS = 0 V)

项目	符号	条件	最小值	典型值	最大值	单元
输出电压, 高电位	VOH1	P02, P03, P05, P06, P10 至 P12, P30, P31, P40, P75 至 P77, P120, P147, P200 至 P206	4.0 V ≤ VDD ≤ 5.5 V, IOH1 = -3.0 mA		VDD - 0.7	V
			2.7 V ≤ VDD ≤ 5.5 V, IOH1 = -1.0 mA		VDD - 0.5	V
	VOH2	P20 至 P22, P24 至 P27	2.7 V ≤ VDD ≤ 5.5 V, IOH2 = -100 μA		VDD - 0.5	V
输出电压, 低电位	VOL1	P02, P03, P05, P06, P10 至 P12, P30, P31, P40, P75 至 P77, P120, P147, P200 至 P206	4.0 V ≤ VDD ≤ 5.5 V, IOL1 = 8.5 mA		0.7	V
			4.0 V ≤ VDD ≤ 5.5 V, IOL1 = 4.0 mA		0.4	V
			2.7 V ≤ VDD ≤ 5.5 V, IOL1 = 1.5 mA		0.4	V
	VOL2	P20 至 P22, P24 至 P27	2.7 V ≤ VDD ≤ 5.5 V, IOL2 = 400 μA		0.4	V

注意事项 在 N 沟开漏模式下, P02、P10 至 P12 不输出高电位。

备注 除非另有说明, 否则复用功能引脚的特性与端口引脚的特性相同。

(TA = -40 至+105°C, 2.7 V ≤ VDD ≤ 5.5 V, VSS = 0 V)

项目	符号	条件	最小值	典型值	最大值	单元	
输入漏电流, 高电位	I _{LIH1}	P02, P03, P05, P06, P10 至 P12, P20 至 P22, P24 至 P27, P30, P31, P40, P75 至 P77, P120, P137, P147, P200 至 P206, $\overline{\text{RESET}}$	V _I = V _{DD}			1	μA
	I _{LIH2}	P121 至 P124 (X1, X2, XT1, XT2, EXCLK, EXCLKS)	V _I = V _{DD}	在输入端口或外部时钟输入时			1
			在谐振器连接时			10	μA
输入漏电流, 低电位	I _{LIL1}	P02, P03, P05, P06, P10 至 P12, P20 至 P22, P24 至 P27, P30, P31, P40, P75 至 P77, P120, P137, P147, P200 至 P206, $\overline{\text{RESET}}$	V _I = V _{SS}			-1	μA
	I _{LIL2}	P121 至 P124 (X1, X2, XT1, XT2, EXCLK, EXCLKS)	V _I = V _{SS}	在输入端口或外部时钟输入时			-1
			在谐振器连接时			-10	μA
内部上拉电阻	R _U	P02, P03, P05, P06, P10 至 P12, P20 至 P22, P24 至 P27, P30, P31, P40, P75 至 P77, P120 至 P124, P137, P147, P200 至 P206	V _I = V _{SS} , 输入端口时	10	20	100	kΩ

备注 除非另有说明, 否则复用功能引脚的特性与端口引脚的特性相同。

32.3.2 电源电流特性

(TA = -40 至+105°C, 2.7 V ≤ VDD ≤ 5.5 V, VSS = 0 V)

(1/2)

参数	符号	条件		最小值	典型值	最大值	单元		
电源电流	IDD1 ^{※1}	操作模式	HS (高速主) 模式 ^{※5}	f _{IH} = 32 MHz ^{※3}	V _{DD} = 5.0 V		5.0	7.5	mA
					V _{DD} = 3.0 V		5.0	7.5	mA
				f _{IH} = 24 MHz ^{※3}	V _{DD} = 5.0 V		3.9	5.8	mA
					V _{DD} = 3.0 V		3.9	5.8	mA
				f _{IH} = 16 MHz ^{※3}	V _{DD} = 5.0 V		2.9	4.2	mA
					V _{DD} = 3.0 V		2.9	4.2	mA
			LS (低速主) 模式 ^{※5}	f _{IH} = 8 MHz ^{※3} , TA = -40 至+ 85°C	V _{DD} = 3.0 V		1.3	2.0	mA
			HS (高速主) 模式 ^{※5}	f _{MX} = 20 MHz ^{※2} , V _{DD} = 5.0 V	方波输入		3.2	4.9	mA
					谐振器连接		3.3	5.0	mA
				f _{MX} = 20 MHz ^{※2} , V _{DD} = 3.0 V	方波输入		3.2	4.9	mA
					谐振器连接		3.3	5.0	mA
				f _{MX} = 10 MHz ^{※2} , V _{DD} = 5.0 V	方波输入		2.0	2.9	mA
					谐振器连接		2.0	2.9	mA
				f _{MX} = 10 MHz ^{※2} , V _{DD} = 3.0 V	方波输入		2.0	2.9	mA
					谐振器连接		2.0	2.9	mA
			LS (低速主) 模式 ^{※5}	f _{MX} = 8 MHz ^{※2} , V _{DD} = 3.0 V, TA = -40 至+ 85°C	方波输入		1.2	1.8	mA
					谐振器连接		1.2	1.8	mA
			HS (高速主) 模式 ^{※5}	f _{IH} = 4 MHz ^{※3} f _{PLL} = 64 MHz, f _{CLK} = 32 MHz	V _{DD} = 5.0 V		5.4	8.5	mA
					V _{DD} = 3.0 V		5.4	8.5	mA
				f _{IH} = 4 MHz ^{※3} f _{PLL} = 64 MHz, f _{CLK} = 16 MHz	V _{DD} = 5.0 V		3.3	5.7	mA
					V _{DD} = 3.0 V		3.3	5.7	mA
			副系统时钟操作	f _{SUB} = 32.768 kHz ^{※4} TA = -40°C	方波输入		4.2		μA
					谐振器连接		4.4		μA
				f _{SUB} = 32.768 kHz ^{※4} TA = +25°C	方波输入		4.2	6.0	μA
					谐振器连接		4.4	6.2	μA
				f _{SUB} = 32.768 kHz ^{※4} TA = +50°C	方波输入		4.3	7.2	μA
					谐振器连接		4.5	7.4	μA
				f _{SUB} = 32.768 kHz ^{※4} TA = +70°C	方波输入		4.4	8.1	μA
					谐振器连接		4.6	8.3	μA
				f _{SUB} = 32.768 kHz ^{※4} TA = +85°C	方波输入		5.2	11.4	μA
谐振器连接		5.4			11.6	μA			
f _{SUB} = 32.768 kHz ^{※4} TA = +105°C	方波输入		6.9	20.8	μA				
	谐振器连接		7.1	21.0	μA				

(注和备注如下页所示。)

- 注 1.** 流入 V_{DD} 的总电流，包括输入引脚电平固定至 V_{DD} 或 V_{SS} 时的输入漏电流。最大值栏下方的值包括外围操作电流（后台操作(BGO)除外）。然而，不包括流过 A/D 转换器、比较器、可编程增益放大器、LVD 电路、输入/输出端口和内部上拉/下拉电阻的电流。
2. 当高速片上振荡器和副系统时钟停止时。
 3. 当高速系统时钟和副系统时钟停止时。
 4. 当高速片上振荡器和高速系统时钟停止时。当实时计数器、看门狗定时器、可编程增益放大器和比较器停止时。当 $AMPHS1 = 1$ 时 (超低功耗振荡)。
 5. 工作电压宽度、CPU 操作频率和操作模式之间的关系如下所示。
HS (高速主) 模式: $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}@1\text{ MHz 至 }32\text{ MHz}$
LS (低速主) 模式: $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}@1\text{ MHz 至 }8\text{ MHz}$

- 备注 1.** f_{MX} : 高速系统时钟频率 (X1 时钟振荡频率或外部主系统时钟频率)
2. f_{IH} : 高速片上振荡器时钟频率
 3. f_{SUB} : 副系统时钟频率 (XT1 时钟振荡频率)
 4. 除副系统时钟操作以外的典型值的温度条件为 $T_A = 25^\circ\text{C}$

(TA = -40 至+105°C, 2.7 V ≤ VDD ≤ 5.5 V, VSS = 0 V)

(2/2)

参数	符号	条件		最小值	典型值	最大值	单元			
电源电流 ^{注1}	IDD2 ^{注2}	HALT 模式	HS (高速主) 模式 ^{注7}	f _{IH} = 32 MHz ^{注4}	V _{DD} = 5.0 V		0.72	2.9	mA	
					V _{DD} = 3.0 V		0.72	2.9	mA	
				f _{IH} = 24 MHz ^{注4}	V _{DD} = 5.0 V		0.57	2.3	mA	
					V _{DD} = 3.0 V		0.57	2.3	mA	
				f _{IH} = 16 MHz ^{注4}	V _{DD} = 5.0 V		0.50	1.7	mA	
					V _{DD} = 3.0 V		0.50	1.7	mA	
			LS (低速主) 模式 ^{注7}	f _{IH} = 8 MHz ^{注4} , TA = -40 至+85°C	V _{DD} = 3.0 V		320	910	μA	
			HS (高速主) 模式 ^{注7}	f _{MX} = 20 MHz ^{注3} , V _{DD} = 5.0 V	方波输入		0.40	1.9	mA	
					谐振器连接		0.50	2.0	mA	
					f _{MX} = 20 MHz ^{注3} , V _{DD} = 3.0 V	方波输入		0.40	1.9	mA
					谐振器连接		0.50	2.0	mA	
		f _{MX} = 10 MHz ^{注3} , V _{DD} = 5.0 V		方波输入		0.24	1.02	mA		
				谐振器连接		0.30	1.08	mA		
		f _{MX} = 10 MHz ^{注3} , V _{DD} = 3.0 V		方波输入		0.24	1.02	mA		
				谐振器连接		0.30	1.08	mA		
		LS (低速主) 模式 ^{注7}	f _{MX} = 8 MHz ^{注3} , V _{DD} = 3.0 V, TA = -40 至+85°C	方波输入		130	720	μA		
				谐振器连接		170	760	μA		
		HS (高速主) 模式 ^{注7}	f _{IH} = 4 MHz ^{注4} , f _{PLL} = 64 MHz, f _{CLK} = 32 MHz	V _{DD} = 5.0 V		1.15	4.0	mA		
				V _{DD} = 3.0 V		1.15	4.0	mA		
			f _{IH} = 4 MHz ^{注4} , f _{PLL} = 64 MHz, f _{CLK} = 16 MHz	V _{DD} = 5.0 V		0.95	3.2	mA		
				V _{DD} = 3.0 V		0.95	3.2	mA		
		副系统时钟操作	f _{SUB} = 32.768 kHz ^{注5} , TA = -40°C	方波输入		0.28		μA		
				谐振器连接		0.47		μA		
	f _{SUB} = 32.768 kHz ^{注5} , TA = +25°C		方波输入		0.33	0.70	μA			
			谐振器连接		0.52	0.89	μA			
	f _{SUB} = 32.768 kHz ^{注5} , TA = +50°C		方波输入		0.41	1.90	μA			
			谐振器连接		0.60	2.09	μA			
	f _{SUB} = 32.768 kHz ^{注5} , TA = +70°C		方波输入		0.54	2.80	μA			
			谐振器连接		0.73	2.99	μA			
	f _{SUB} = 32.768 kHz ^{注5} , TA = +85°C	方波输入		1.27	6.10	μA				
		谐振器连接		1.46	6.29	μA				
f _{SUB} = 32.768 kHz ^{注5} , TA = +105°C	方波输入		3.04	15.5	μA					
	谐振器连接		3.23	15.7	μA					
IDD3 ^{注6}	STOP 模式 ^{注8}	TA = -40°C			0.18		μA			
		TA = +25°C			0.23	0.50	μA			
		TA = +50°C			0.27	1.70	μA			
		TA = +70°C			0.44	2.60	μA			
		TA = +85°C			1.17	5.90	μA			
		TA = +105°C			2.94	15.3	μA			

(注和备注如下页所示。)

- 注 1.** 流入 V_{DD} 的总电流，包括输入引脚电平固定至 V_{DD} 或 V_{SS} 时的输入漏电流。最大值栏下方的值包括外围操作电流。然而，不包括流过 A/D 转换器、比较器、可编程增益放大器、LVD 电路、输入/输出端口和内部上拉/下拉电阻的电流。
2. 通过内存执行 HALT 指令时。
 3. 当高速片上振荡器和副系统时钟停止时。
 4. 当高速系统时钟和副系统时钟停止时。
 5. 当操作实时时钟(RTC)且设置为超低功耗(AMPHS1 = 1)时。当高速片上振荡器和高速系统时钟停止时。当看门狗定时器停止时。最大值栏下方的值包括漏电流。
 6. 当高速片上振荡器时钟、高速系统时钟和副系统时钟停止时。当看门狗定时器停止时。最大值栏下方的值包括漏电流。
 7. 工作电压宽度、CPU 操作频率和操作模式之间的关系如下所示。
HS (高速主) 模式: $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}@1\text{ MHz 至 }32\text{ MHz}$
LS (低速主) 模式: $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}@1\text{ MHz 至 }8\text{ MHz}$
 8. STOP 模式下操作副系统时钟时，和操作副系统时钟的 HALT 模式时相同。
- 备注 1.** f_{MX} : 高速系统时钟频率 (X1 时钟振荡频率或外部主系统时钟频率)
2. f_{IH} : 高速片上振荡器时钟频率
 3. f_{SUB} : 副系统时钟频率 (XT1 时钟振荡频率)
 4. 除副系统时钟操作和 STOP 模式以外的典型值的温度条件为 $T_A = 25^\circ\text{C}$

(TA = -40 至 +105°C, 2.7 V ≤ VDD ≤ 5.5 V, VSS = 0 V)

参数	符号	条件		最小值	典型值	最大值	单元
RTC 操作电流	IRTC ^{注1,2}	fSUB = 32.768 kHz	实时时钟操作		0.02		μA
			12 位间隔定时器操作				
看门狗定时器操作电流	IWDT ^{注2,3}	fIL = 15 kHz			0.22		μA
A/D 转换器操作电流	IADC ^{注4}	最高速转换时	标准模式, AVREFP = VDD = 5.0 V		1.3	1.7	mA
			低电压模式, AVREFP = VDD = 3.0 V		0.5	0.7	mA
A/D 转换器基准电压电流	IADREF				75.0		μA
温度传感器操作电流	ITMPS				75.0		μA
LVD 操作电流	ILVI ^{注5}				0.08		μA
可编程增益放大器操作电流	IPGA ^{注6}				0.21	0.31	mA
					0.18	0.29	mA
比较器操作电流	ICMP ^{注7}	当一个比较器通道工作时	AVREFP = VDD = 5.0 V		41.4	62	μA
			AVREFP = VDD = 3.0 V		37.2	59	μA
	IVREF	当一个内部基准电压电路工作时	AVREFP = VDD = 5.0 V		14.8	26	μA
			AVREFP = VDD = 3.0 V		8.9	20	μA
可编程增益放大器/比较器基准电流源	IREF ^{注8}				3.2	5.1	μA
					2.9	4.9	μA
BGO 操作电流	IBGO ^{注9}				2.50	12.2	mA

(注和备注如下页所示。)

- 注 1.** 仅流入实时时钟的电流（不包括 XT1 振荡器的操作电流）。当实时时钟工作于操作模式或 HALT 模式时，RL78/I1A 单片机的电流的值为 I_{DD1} 或 I_{DD2} 与 I_{RTC} 的值之和。副系统时钟操作时的 I_{DD2} 包括实时时钟的操作电流。
- 2.** 当高速片上振荡器和高速系统时钟停止工作时。
- 3.** 仅流入看门狗定时器的电流（包括低速片上振荡器的操作电流）。当 f_{CLK} = f_{SUB} 且看门狗定时器工作于 STOP 模式时，RL78/I1A 的电流值为 I_{DD1}、I_{DD2} 或 I_{DD3} 与 I_{WDT} 之和。
- 4.** 仅流入 A/D 转换器的电流。当 A/D 转换器工作于操作模式或 HALT 模式时，RL78/I1A 的电流值为 I_{DD1} 或 I_{DD2} 与 I_{ADC} 之和。
- 5.** 仅流入 LVD 电路的电流。当 LVD 电路工作于 HALT 或 STOP 模式时，RL78/I1A 的电流值为 I_{DD1}、I_{DD2} 或 I_{DD3} 与 I_{COMP} 之和。
- 6.** 仅流入可编程增益放大器的电流。当可编程增益放大器工作于操作模式或 HALT 模式时，RL78/I1A 的电流值为 I_{DD1}、I_{DD2} 或 I_{DD3} 与 I_{PGA} 之和。
- 7.** 仅流入比较器的电流。当比较器工作于操作模式或 HALT 模式时，RL78/I1A 的电流值为 I_{DD1}、I_{DD2} 或 I_{DD3} 与 I_{COMP} 之和。
- 8.** 流入电流电路的 V_{DD} 引脚的电流。使用可编程增益放大器或者比较器时需要该电流电路。
- 9.** 仅流入 BGO 的电流。当 BGO 工作于操作模式时，RL78/I1A 单片机的电流值为 I_{DD1} 或 I_{DD2} 与 I_{BGO} 之和。

- 备注 1.** fil: 低速片上振荡器时钟频率
- 2.** f_{SUB}: 副系统时钟频率 (XT1 时钟振荡频率)
- 3.** f_{CLK}: CPU/外围硬件时钟频率
- 4.** 典型值的温度条件为 T_A = 25°C
- 5.** 使用可编程增益放大器和比较器时计算电流值的示例。
- 示例 1) 操作比较器 3 通道、内部基准电压电路 1 通道以及 PGA 时的典型值工作电流(当 AV_{REFP} = V_{DD} = 5.0 V 时)

$$\begin{aligned} & I_{\text{COMP}} \times 3 + I_{\text{VREF}} + I_{\text{PGA}} + I_{\text{IREF}} \\ &= 41.4 [\mu\text{A}] \times 3 + 14.8 [\mu\text{A}] \times 1 + 210 [\mu\text{A}] + 3.2 [\mu\text{A}] \\ &= 352.2 [\mu\text{A}] \end{aligned}$$

示例 2) 使用比较器 2 通道并且不使用内部基准电压电路时的典型值工作电流(当 AV_{REFP} = V_{DD} = 5.0 V 时)

$$\begin{aligned} & I_{\text{COMP}} \times 2 + I_{\text{IREF}} \\ &= 41.4 [\mu\text{A}] \times 2 + 3.2 [\mu\text{A}] \\ &= 86.0 [\mu\text{A}] \end{aligned}$$

32.4 AC 特性

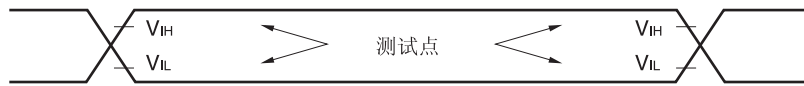
(T_A = -40 至+105°C, 2.7 V ≤ V_{DD} ≤ 5.5 V, V_{SS} = 0 V)

项目	符号	条件		最小值	典型值	最大值	单元	
指令周期(最短指令执行时间)	T _{CY}	主系统时钟 (f _{MAIN})操作	HS (高速主)模式	2.7 V ≤ V _{DD} ≤ 5.5 V	0.03125		1	μs
			LS (低速主)模式	2.7 V ≤ V _{DD} ≤ 5.5 V, T _A = -40 至+85°C	0.125		1	μs
		副系统时钟 (f _{SUB})操作			28.5	30.5	31.3	μs
		自编程模式	HS (高速主)模式	2.7 V ≤ V _{DD} ≤ 5.5 V	0.03125		1	μs
			LS (低速主)模式	2.7 V ≤ V _{DD} ≤ 5.5 V, T _A = -40 至+85°C	0.125		1	μs
外部系统时钟频率	f _{EX}	2.7 V ≤ V _{DD} ≤ 5.5 V		1.0		20.0	MHz	
	f _{EXT}			32		35	kHz	
外部系统时钟输入高电位, 低电位宽度	t _{EXH} , t _{EXL}	2.7 V ≤ V _{DD} ≤ 5.5 V		24			ns	
	t _{EXHS} , t _{EXLS}			13.7			μs	
TI03、TI05、TI06、TI07 输入高电位, 低电位宽度	t _{TIH} , t _{TIL}			2/f _{MCK} +10			ns	
TO03, TO05, TO06, TKBO00, TKBO01, TKBO10, TKBO11, TKBO20, TKBO21, TKCO00 至 TKCO05 输出频率(当 占空比 = 50%)	f _{TO}	HS (高速主)模式	4.0 V ≤ V _{DD} ≤ 5.5 V			8	MHz	
			2.7 V ≤ V _{DD} < 4.0 V			4	MHz	
		LS (低速主)模式, T _A = -40 至+85°C	4.0 V ≤ V _{DD} ≤ 5.5 V			4	MHz	
			2.7 V ≤ V _{DD} < 4.0 V			2	MHz	
中断输入高电位, 低电位宽度	t _{INTH} , t _{INTL}	INTP0, INTP3, INTP4, INTP9 至 INTP11, INTP20 至 INTP23	2.7 V ≤ V _{DD} ≤ 5.5 V	1			μs	
RESET 低电位宽度	t _{RSL}			10			μs	

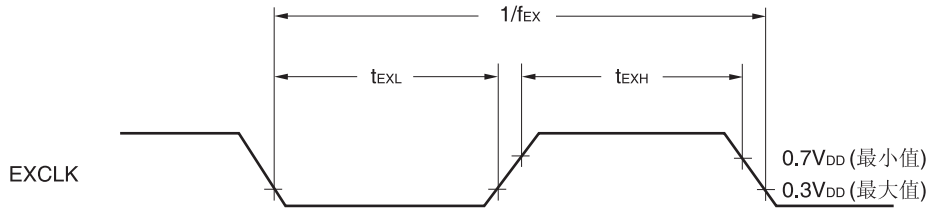
备注 f_{MCK}: 定时器阵列单元的操作时钟频率。

(由定时器模式寄存器 0n (TMR0n) 的 CKS0n 位设置的操作时钟。n: 通道编号 (n = 0 至 7))

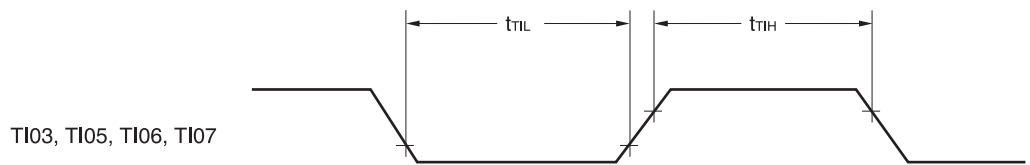
AC 时序测试点



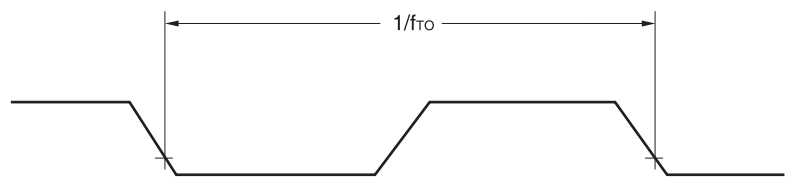
外部系统时钟时序



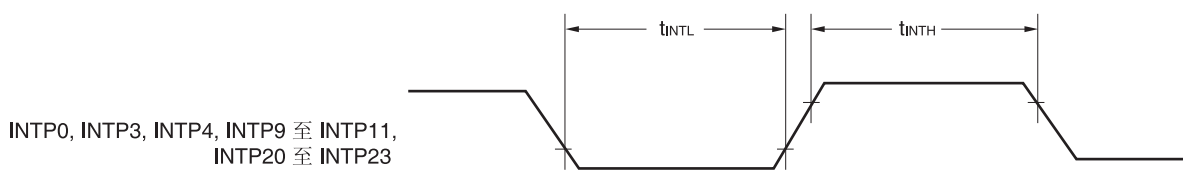
TI/TO 时序



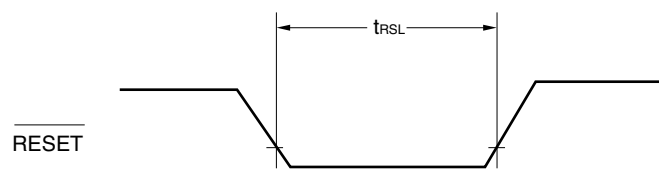
TO03, TO05, TO06, TKBO00, TKBO01, TKBO10, TKBO11, TKBO20, TKBO21, TKCO00 至 TKCO05



输入中断请求的时序



RESET 输入时序



32.5 外围功能特性

32.5.1 串行阵列单元 0, 4 (UART0, UART1, CSI00, DALI/UART4)

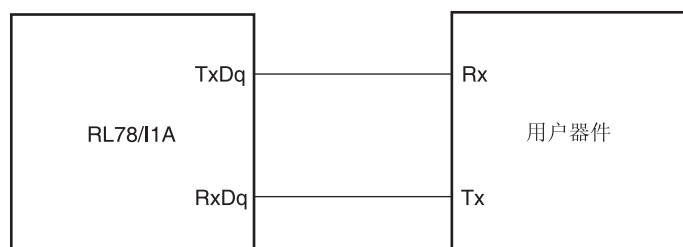
(1) 相同电位通信时(UART 模式) (专用波特率生成器输出)

($T_A = -40$ 至 $+105^\circ\text{C}$, $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$)

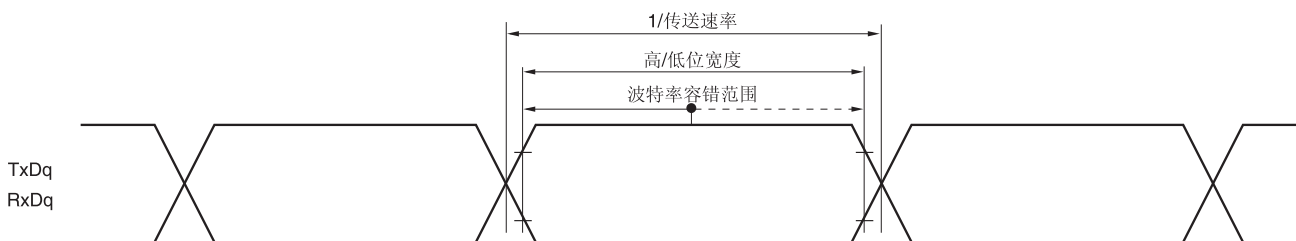
参数	符号	条件	最小值	典型值	最大值	单元
传送速率 ^注		$2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$			$f_{MCK}/6$	bps
		最大传送速率理论值 $f_{CLK} = 32\text{ MHz}$, $f_{MCK} = f_{CLK}$			5.3	Mbps

注 SNOOZE 模式下的传送速率为最大值 9600 bps, 最小值 4800 bps。

UART 模式连接图(相同电位通信时)



UART 模式时的位宽度(相同电位通信时) (参考)



注意事项 使用端口输入模式寄存器 g (PIM g)和端口输出模式寄存器 g (POM g), 将 RxDq 引脚选择为普通输入缓冲器, TxDq 引脚选择为普通输出模式。

备注 1. q: UART 编号($q = 0, 1$), g: PIM 和 POM 编号($g = 0, 1$)

2. f_{MCK} : 串行阵列单元的操作时钟频率。

(由串行模式寄存器 mn (SMR mn)的 CKS mn 位设置的操作时钟。m: 单元编号,

n: 通道编号 ($mn = 00$ 至 03))

(2) 相同电位通信时(CSI 模式) (主模式($f_{MCK}/4$), \overline{SCKp} ... 内部时钟输出)(T_A = -40 至 +105°C, 2.7 V ≤ V_{DD} ≤ 5.5 V, V_{SS} = 0 V)

参数	符号	条件	最小值	典型值	最大值	单元
\overline{SCKp} 周期	t _{KCY1}	2.7 V ≤ V _{DD} ≤ 5.5 V	125 ^{※1}			ns
\overline{SCKp} 高电位/低电位宽度	t _{KH1} , t _{KL1}	4.0 V ≤ V _{DD} ≤ 5.5 V	t _{KCY1} /2 - 12			ns
		2.7 V ≤ V _{DD} ≤ 5.5 V	t _{KCY1} /2 - 18			ns
Slp 建立时间 (至 $\overline{SCKp}\uparrow$) ^{※2}	t _{SIK1}	4.0 V ≤ V _{DD} ≤ 5.5 V	44			ns
		2.7 V ≤ V _{DD} ≤ 5.5 V	44			ns
Slp 保持时间 (从 $\overline{SCKp}\uparrow$) ^{※3}	t _{KSI1}		19			ns
从 $\overline{SCKp}\downarrow$ 至 SOp 输出的延迟时间 ^{※4}	t _{KSO1}	C = 30 pF ^{※5}			25	ns

注 1. 需设定为 4/f_{CLK} 或更高的值。

- 当 DAPmn = 0 和 CKPmn = 0, 或 DAPmn = 1 和 CKPmn = 1 时。DAPmn = 0 和 CKPmn = 1, 或 DAPmn = 1 和 CKPmn = 0 时, Slp 建立时间变为“至 $\overline{SCKp}\downarrow$ ”。
- 当 DAPmn = 0 和 CKPmn = 0, 或 DAPmn = 1 和 CKPmn = 1 时。DAPmn = 0 和 CKPmn = 1, 或 DAPmn = 1 和 CKPmn = 0 时, Slp 保持时间变为“从 $\overline{SCKp}\downarrow$ ”。
- 当 DAPmn = 0 和 CKPmn = 0, 或 DAPmn = 1 和 CKPmn = 1 时。DAPmn = 0 和 CKPmn = 1, 或 DAPmn = 1 和 CKPmn = 0 时, SOp 输出的延迟时间变为“从 $\overline{SCKp}\uparrow$ ”。
- C 为 \overline{SCKp} 和 SOp 输出线路的负载电容。

注意事项 使用端口输入模式寄存器 g (PIMg)和端口输出模式寄存器 g (POMg), 将 Slp 引脚选择为普通输入缓冲器, Sop 引脚和 \overline{SCKp} 引脚选择为普通输出模式。

备注 1. p: CSI 编号(p = 00), m: 单元编号(m = 0), n: 通道编号(n = 0),

g: PIM 和 POM 编号(g = 1)

- f_{MCK}: 串行阵列单元的操作时钟频率。

(由串行模式寄存器 mn (SMRmn)的 CKSmn 位设置的操作时钟。m: 单元编号,

n: 通道编号 (mn = 00))

(3) 相同电位通信时 (CSI 模式) (从属模式, $\overline{\text{SCKp}}$... 外部时钟输入)(T_A = -40 至 +105°C, 2.7 V ≤ V_{DD} ≤ 5.5 V, V_{SS} = 0 V)

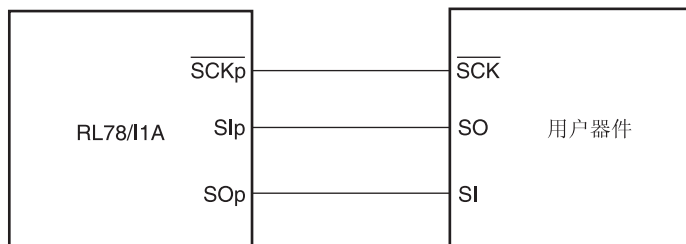
参数	符号	条件	最小值	典型值	最大值	单元
$\overline{\text{SCKp}}$ 周期 ^{注5}	t _{KCY2}	4.0 V ≤ V _{DD} ≤ 5.5 V	20 MHz < f _{MCK}	8/f _{MCK}		ns
			f _{MCK} ≤ 20 MHz	6/f _{MCK}		ns
		2.7 V ≤ V _{DD} < 4.0 V	16 MHz < f _{MCK}	8/f _{MCK}		ns
			f _{MCK} ≤ 16 MHz	6/f _{MCK}		ns
$\overline{\text{SCKp}}$ 高电位/低电位宽度	t _{KH2} , t _{KL2}		t _{KCY2} /2			ns
Slp 建立时间 (至 $\overline{\text{SCKp}}\uparrow$) ^{注1}	t _{SIK2}		1/f _{MCK} +20			ns
Slp 保持时间 (从 $\overline{\text{SCKp}}\uparrow$) ^{注2}	t _{KS12}		1/f _{MCK} +31			ns
从 $\overline{\text{SCKp}}\downarrow$ 至 SOp 输出的延迟时间 ^{注3}	t _{KSO2}	C = 30 pF ^{注4}			2/f _{MCK} +44	ns

- 注 1. 当 DAPmn = 0 和 CKPmn = 0, 或 DAPmn = 1 和 CKPmn = 1 时。DAPmn = 0 和 CKPmn = 1, 或 DAPmn = 1 和 CKPmn = 0 时, Slp 建立时间变为“至 $\overline{\text{SCKp}}\downarrow$ ”。
- 注 2. 当 DAPmn = 0 和 CKPmn = 0, 或 DAPmn = 1 和 CKPmn = 1 时。DAPmn = 0 和 CKPmn = 1, 或 DAPmn = 1 和 CKPmn = 0 时, Slp 保持时间变为“从 $\overline{\text{SCKp}}\downarrow$ ”。
- 注 3. 当 DAPmn = 0 和 CKPmn = 0, 或 DAPmn = 1 和 CKPmn = 1 时。DAPmn = 0 和 CKPmn = 1, 或 DAPmn = 1 和 CKPmn = 0 时, SOp 输出的延迟时间变为“从 $\overline{\text{SCKp}}\uparrow$ ”。
- 注 4. C 为 SOp 输出线路的负载电容。
- 注 5. SNOOZE 模式下的传送速率为最大值 1 Mbps。

注意事项 使用端口输入模式寄存器 g (PIMg)和端口输出模式寄存器 g (POMg)时, 将 Slp 引脚和 $\overline{\text{SCKp}}$ 引脚选择为普通输入缓冲器, SOp 引脚选择为普通输出模式。

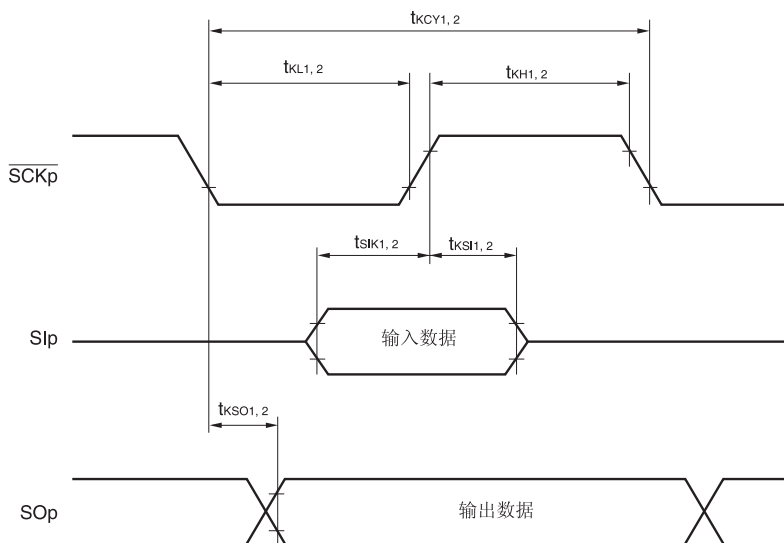
- 备注 1. p: CSI 编号(p = 00), m: 单元编号(m = 0), n: 通道编号(n = 0), g: PIM 和 POM 编号(g = 1)
2. f_{MCK}: 串行阵列单元的操作时钟频率。
(由串行模式寄存器 mn (SMRmn)的 CKSmn 位设置的操作时钟。m: 单元编号, n: 通道编号(mn = 00))

CSI 模式连接图(相同电位通信时)



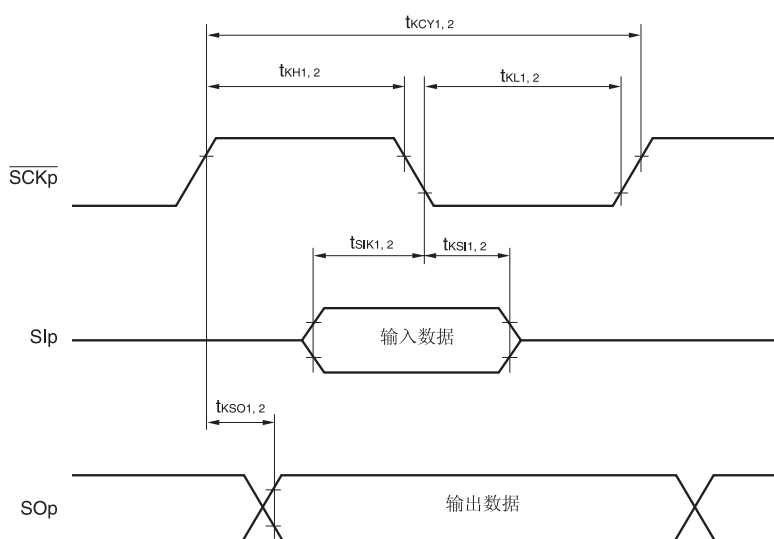
CSI 模式串行传送时序 (相同电位通信时)

($\text{DAPmn} = 0$ 和 $\text{CKPmn} = 0$, 或 $\text{DAPmn} = 1$ 和 $\text{CKPmn} = 1$ 时)



CSI 模式串行传送时序 (相同电位通信时)

($\text{DAPmn} = 0$ 和 $\text{CKPmn} = 1$, 或 $\text{DAPmn} = 1$ 和 $\text{CKPmn} = 0$ 时)



- 备注 1. p: CSI 编号(p = 00)
- 2. m: 单元编号, n: 通道编号(mn = 00)

(4) 不同电位通信时(2.5 V, 3 V) (UART 模式) (专用波特率生成器输出) (1/2)

(T_A = -40 至+105°C, 2.7 V ≤ V_{DD} ≤ 5.5 V, V_{SS} = 0 V)

参数	符号	条件		最小值	典型值	最大值	单元
传送速率		接收	4.0 V ≤ V _{DD} ≤ 5.5 V, 2.7 V ≤ V _b ≤ 4.0 V			f _{MCK} /6 ^注	bps
				最大传送速率的理论值 f _{CLK} = 32 MHz, f _{MCK} = f _{CLK}		5.3	Mbps
			2.7 V ≤ V _{DD} < 4.0 V, 2.3 V ≤ V _b ≤ 2.7 V			f _{MCK} /6 ^注	bps
				最大传送速率的理论值 f _{CLK} = 32 MHz, f _{MCK} = f _{CLK}		5.3	Mbps

注 SNOOZE 模式下的传送速率为最大值 9600 bps，最小值 4800 bps。

注意事项 使用端口输入模式寄存器 g (PIMg)和端口输出模式寄存器 g (POMg)，将 RxDq 引脚选择为 TTL 输入缓冲器，TxDq 引脚选择为 N 沟开漏输出 (V_{DD} 耐压)模式。

备注 1. V_b [V]: 通信线路电压

2. q: UART 编号(q = 0, 1), g: PIM 和 POM 编号(g = 0, 1)

3. f_{MCK}: 串行阵列单元的操作时钟频率。

(由串行模式寄存器 mn (SMRmn)的 CKSmn 位设置的操作时钟。m: 单元编号, n: 通道编号 (mn = 00 至 03))

(4) 不同电位通信时(2.5 V, 3 V) (UART 模式) (专用波特率生成器输出) (2/2)

(TA = -40 至+105°C, 2.7 V ≤ VDD ≤ 5.5 V, VSS = 0 V)

参数	符号	条件		最小值	典型值	最大值	单元
传送速率		发送	4.0 V ≤ VDD ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V			注 1, 2	bps
				最大传送速率理论值 Cb = 50 pF, Rb = 1.4 kΩ, Vb = 2.7 V			2.8 ^{#3}
			2.7 V ≤ VDD < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V			注 2, 4	bps
				最大传送速率理论值 Cb = 50 pF, Rb = 2.7 kΩ, Vb = 2.3 V			1.2 ^{#5}

注 1. 在 $f_{MCK}/6$ 和用以下表达式得到的最大传送速率中较小的一个为有效的最大传送速率。

当 $4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$ 时的传送速率计算公式示例

$$\text{最大传送速率} = \frac{1}{\{-C_b \times R_b \times \ln(1 - \frac{2.2}{V_b})\} \times 3} \text{ [bps]}$$

$$\text{波特率误差(理论值)} = \frac{\frac{1}{\text{传送速率} \times 2} - \{-C_b \times R_b \times \ln(1 - \frac{2.2}{V_b})\}}{(\frac{1}{\text{传送速率}}) \times \text{传送位数量}} \times 100 \text{ [%]}$$

* 该值为发送端与接收端之间的理论相对差值。

2. SNOOZE 模式下的传送速率为最大值 9600 bps, 最小值 4800 bps。
3. 此值作为一示例, 是在“条件”栏中所列条件满足时计算得到的。请参阅上面的注 1, 来计算用户使用条件下的最大传送速率。
4. 在 $f_{MCK}/6$ 和用以下表达式得到的最大传送速率中较小的一个为有效的最大传送速率。
当 $2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$ 时的传送速率计算公式示例

$$\text{最大传送速率} = \frac{1}{\{-C_b \times R_b \times \ln(1 - \frac{2.0}{V_b})\} \times 3} \text{ [bps]}$$

$$\text{波特率误差(理论值)} = \frac{\frac{1}{\text{传送速率} \times 2} - \{-C_b \times R_b \times \ln(1 - \frac{2.0}{V_b})\}}{(\frac{1}{\text{传送速率}}) \times \text{传送位数量}} \times 100 \text{ [%]}$$

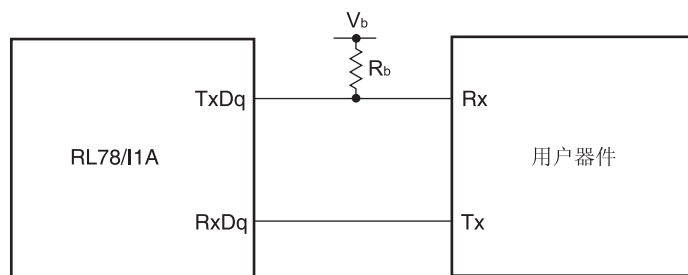
* 该值为发送端与接收端之间的理论相对差值。

5. 此值作为一示例, 是在“条件”栏中所列条件满足时计算得到的。请参阅上面的注 4, 来计算用户使用条件下的最大传送速率。

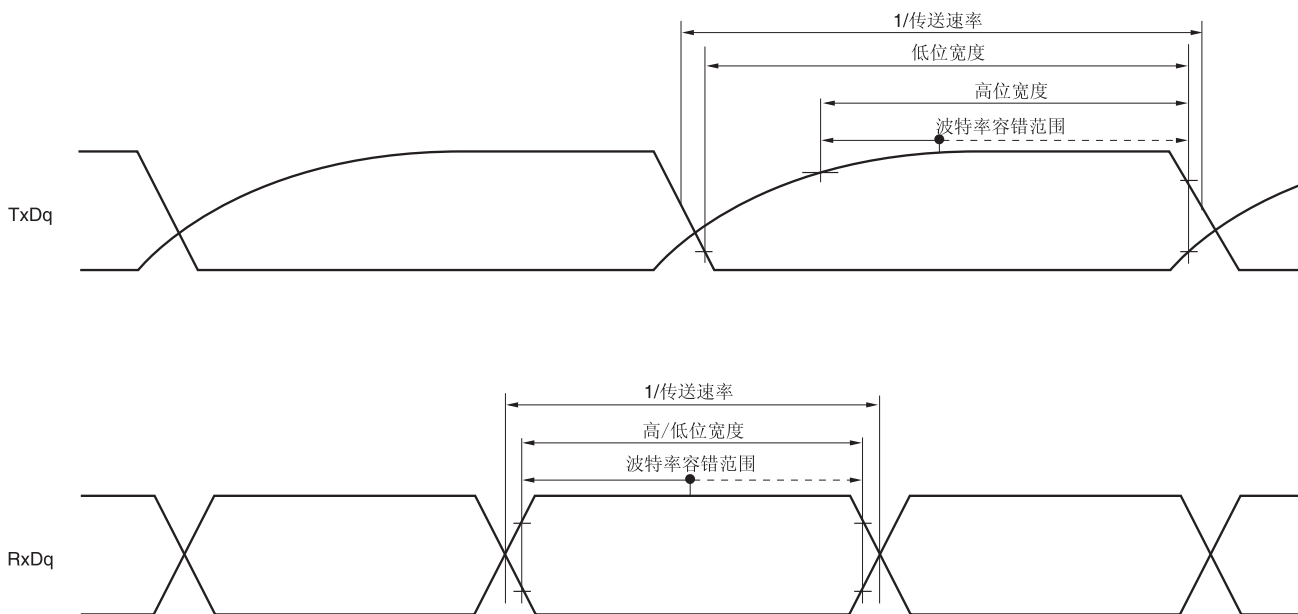
注意事项 使用端口输入模式寄存器 g (PIMg)和端口输出模式寄存器 g (POMg), 将 RxDq 引脚选择为 TTL 输入缓冲器, TxDq 引脚选择为 N 沟开漏输出 (VDD 耐压)模式。

- 备注 1.** Rb[Ω]: 通信线路 (TxDq) 上拉电阻,
Cb[F]: 通信线路 (TxDq)负载电容, Vb[V]: 通信线路电压
2. q: UART 编号(q = 0, 1), g: PIM 和 POM 编号(g = 0, 1)
 3. fMCK: 串行阵列单元的操作时钟频率。
(由串行模式寄存器 mn (SMRmn)的 CKSmn 位设置的操作时钟。
m: 单元编号, n: 通道编号 (mn = 00 至 03))

UART 模式连接图(不同电位通信时)



UART 模式位宽度 (不同电位通信时) (参考)



注意事项 使用端口输入模式寄存器 g (PIMg)和端口输出模式寄存器 g (POMg), 将 RxDq 引脚选择为 TTL 输入缓冲器, TxDq 引脚选择为 N 沟开漏输出 (V_{DD} 耐压)模式。

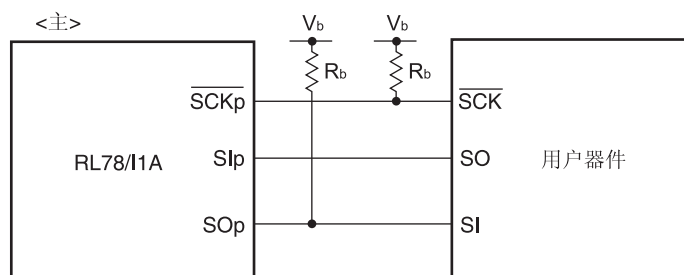
- 备注 1.** R_b[Ω]: 通信线路 (TxDq) 上拉电阻, V_b[V]: 通信线路电压
2. q: UART 编号(q = 0, 1), g: PIM 和 POM 编号(g = 0, 1)

(5) 不同电位(2.5 V, 3 V) (f_{MCK/2})通信时 (CSI 模式) (主模式, $\overline{\text{SCKp}}$... 内部时钟输出)(T_A = -40 至 +105°C, 2.7 V ≤ V_{DD} ≤ 5.5 V, V_{SS} = 0 V)

参数	符号	条件	最小值	典型值	最大值	单元
SCKp 周期	t _{KCY1}	4.0 V ≤ V _{DD} ≤ 5.5 V, 2.7 V ≤ V _b ≤ 4.0 V, C _b = 20 pF, R _b = 1.4 kΩ	200 ^{※1}			ns
		2.7 V ≤ V _{DD} < 4.0 V, 2.3 V ≤ V _b ≤ 2.7 V, C _b = 20 pF, R _b = 2.7 kΩ	300 ^{※1}			ns
SCKp 高电位宽度	t _{KH1}	4.0 V ≤ V _{DD} ≤ 5.5 V, 2.7 V ≤ V _b ≤ 4.0 V, C _b = 20 pF, R _b = 1.4 kΩ	t _{KCY1} /2 - 50			ns
		2.7 V ≤ V _{DD} < 4.0 V, 2.3 V ≤ V _b ≤ 2.7 V, C _b = 20 pF, R _b = 2.7 kΩ	t _{KCY1} /2 - 120			ns
SCKp 低电位宽度	t _{KL1}	4.0 V ≤ V _{DD} ≤ 5.5 V, 2.7 V ≤ V _b ≤ 4.0 V, C _b = 20 pF, R _b = 1.4 kΩ	t _{KCY1} /2 - 7			ns
		2.7 V ≤ V _{DD} < 4.0 V, 2.3 V ≤ V _b ≤ 2.7 V, C _b = 20 pF, R _b = 2.7 kΩ	t _{KCY1} /2 - 10			ns
Slp 建立时间 (至 SCKp↑) ^{※2}	t _{SIK1}	4.0 V ≤ V _{DD} ≤ 5.5 V, 2.7 V ≤ V _b ≤ 4.0 V, C _b = 20 pF, R _b = 1.4 kΩ	81			ns
		2.7 V ≤ V _{DD} < 4.0 V, 2.3 V ≤ V _b ≤ 2.7 V, C _b = 20 pF, R _b = 2.7 kΩ	177			ns
Slp 保持时间 (从 SCKp↑) ^{※2}	t _{KSl1}	4.0 V ≤ V _{DD} ≤ 5.5 V, 2.7 V ≤ V _b ≤ 4.0 V, C _b = 20 pF, R _b = 1.4 kΩ	10			ns
		2.7 V ≤ V _{DD} < 4.0 V, 2.3 V ≤ V _b ≤ 2.7 V, C _b = 20 pF, R _b = 2.7 kΩ	10			ns
从 $\overline{\text{SCKp}}$ ↓ 至 SOp 输出的延迟 时间 ^{※2}	t _{KSO1}	4.0 V ≤ V _{DD} ≤ 5.5 V, 2.7 V ≤ V _b ≤ 4.0 V, C _b = 20 pF, R _b = 1.4 kΩ			60	ns
		2.7 V ≤ V _{DD} < 4.0 V, 2.3 V ≤ V _b ≤ 2.7 V, C _b = 20 pF, R _b = 2.7 kΩ			130	ns
Slp 建立时间 (至 $\overline{\text{SCKp}}$ ↓) ^{※3}	t _{SIK1}	4.0 V ≤ V _{DD} ≤ 5.5 V, 2.7 V ≤ V _b ≤ 4.0 V, C _b = 20 pF, R _b = 1.4 kΩ	44			ns
		2.7 V ≤ V _{DD} < 4.0 V, 2.3 V ≤ V _b ≤ 2.7 V, C _b = 20 pF, R _b = 2.7 kΩ	44			ns
Slp 保持时间 (从 $\overline{\text{SCKp}}$ ↓) ^{※3}	t _{KSl1}	4.0 V ≤ V _{DD} ≤ 5.5 V, 2.7 V ≤ V _b ≤ 4.0 V, C _b = 20 pF, R _b = 1.4 kΩ	10			ns
		2.7 V ≤ V _{DD} < 4.0 V, 2.3 V ≤ V _b ≤ 2.7 V, C _b = 20 pF, R _b = 2.7 kΩ	10			ns
从 $\overline{\text{SCKp}}$ ↑ 至 SOp 输出的延迟 时间 ^{※3}	t _{KSO1}	4.0 V ≤ V _{DD} ≤ 5.5 V, 2.7 V ≤ V _b ≤ 4.0 V, C _b = 20 pF, R _b = 1.4 kΩ			10	ns
		2.7 V ≤ V _{DD} < 4.0 V, 2.3 V ≤ V _b ≤ 2.7 V, C _b = 20 pF, R _b = 2.7 kΩ			10	ns

(注、注意事项和备注如下页所示。)

CSI 模式连接图(不同电位通信时)



- 注 1. 且需设定为 $2/f_{CLK}$ 或更高的值。
 2. DAPmn = 0 和 CKPmn = 0, 或 DAPmn = 1 和 CKPmn = 1 时。
 3. DAPmn = 0 和 CKPmn = 1, 或 DAPmn = 1 和 CKPmn = 0 时。

注意事项 使用端口输入模式寄存器 g (PIMg)和端口输出模式寄存器 g (POMg), 将 Slp 引脚选择为 TTL 输入缓冲器, SOp 引脚和 SCKp 引脚选择为 N 沟开漏输出(V_{DD} 耐压)模式。

- 备注 1. $R_b[\Omega]$: 通信线路 (SCKp, SOp) 上拉电阻, $C_b[F]$: 通信线路 (SCKp, SOp)负载电容, $V_b[V]$: 通信线路电压
 2. p: CSI 编号(p = 00), m: 单元编号(m = 0), n: 通道编号(n = 0), g: PIM 和 POM 编号(g = 1)
 3. f_{MCK} : 串行阵列单元的操作时钟频率。
 (由串行模式寄存器 mn (SMRmn)的 CKSmn 位设置的操作时钟。
 m: 单元编号, n: 通道编号 (mn = 00))

(6) 不同电位(2.5 V, 3 V)通信时 ($f_{MCK}/4$) (CSI 模式) (主模式, \overline{SCKp} ... 内部时钟输出) (1/2)
 ($T_A = -40$ 至 $+105^\circ\text{C}$, $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$)

参数	符号	条件	最小值	典型值	最大值	单元
\overline{SCKp} 周期	t_{KCY1}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	300 ^注			ns
		$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 2.7\text{ k}\Omega$	500 ^注			ns
		$2.7\text{ V} \leq V_{DD} < 3.3\text{ V}$, $1.6\text{ V} \leq V_b \leq 2.0\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 5.5\text{ k}\Omega$	1150 ^注			ns
\overline{SCKp} 高电位宽度	t_{KH1}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	$t_{KCY1}/2 - 75$			ns
		$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 2.7\text{ k}\Omega$	$t_{KCY1}/2 - 170$			ns
		$2.7\text{ V} \leq V_{DD} < 3.3\text{ V}$, $1.6\text{ V} \leq V_b \leq 2.0\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 5.5\text{ k}\Omega$	$t_{KCY1}/2 - 458$			ns
\overline{SCKp} 低电位宽度	t_{KL1}	$4.0\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $2.7\text{ V} \leq V_b \leq 4.0\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 1.4\text{ k}\Omega$	$t_{KCY1}/2 - 12$			ns
		$2.7\text{ V} \leq V_{DD} < 4.0\text{ V}$, $2.3\text{ V} \leq V_b \leq 2.7\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 2.7\text{ k}\Omega$	$t_{KCY1}/2 - 18$			ns
		$2.7\text{ V} \leq V_{DD} < 3.3\text{ V}$, $1.6\text{ V} \leq V_b \leq 2.0\text{ V}$, $C_b = 30\text{ pF}$, $R_b = 5.5\text{ k}\Omega$	$t_{KCY1}/2 - 50$			ns

注 且需设定为 $4/f_{CLK}$ 或更高的值。

注意事项 使用端口输入模式寄存器 g (PIMg)和端口输出模式寄存器 g (POMg), 将 SIp 引脚选择为 TTL 输入缓冲器, SOp 引脚和 SCKp 引脚选择为 N 沟开漏输出(V_{DD} 耐压)模式。

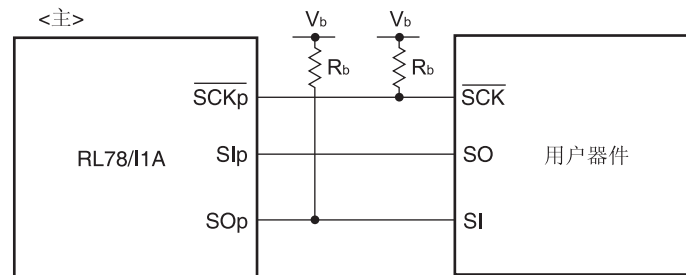
- 备注 1. $R_b[\Omega]$: 通信线路 (\overline{SCKp} , SOp) 上拉电阻, $C_b[\text{F}]$: 通信线路 (\overline{SCKp} , SOp)负载电容, $V_b[\text{V}]$: 通信线路电压
 2. p: CSI 编号(p = 00), m: 单元编号(m = 0), n: 通道编号(n = 0), g: PIM 和 POM 编号(g = 1)

(6) 不同电位(2.5 V, 3 V)通信时 ($f_{MCK}/4$) (CSI 模式) (主模式, \overline{SCKp} ... 内部时钟输出) (2/2)(T_A = -40 至 +105°C, 2.7 V ≤ V_{DD} ≤ 5.5 V, V_{SS} = 0 V)

参数	符号	条件	最小值	典型值	最大值	单元
Slp 建立时间 (至 $\overline{SCKp}\uparrow$) ^{※1}	t _{SIK1}	4.0 V ≤ V _{DD} ≤ 5.5 V, 2.7 V ≤ V _b ≤ 4.0 V, C _b = 30 pF, R _b = 1.4 kΩ	81			ns
		2.7 V ≤ V _{DD} < 4.0 V, 2.3 V ≤ V _b ≤ 2.7 V, C _b = 30 pF, R _b = 2.7 kΩ	177			ns
		2.7 V ≤ V _{DD} < 3.3 V, 1.6 V ≤ V _b ≤ 2.0 V, C _b = 30 pF, R _b = 5.5 kΩ	479			ns
Slp 保持时间 (从 $\overline{SCKp}\uparrow$) ^{※1}	t _{KS1}	4.0 V ≤ V _{DD} ≤ 5.5 V, 2.7 V ≤ V _b ≤ 4.0 V, C _b = 30 pF, R _b = 1.4 kΩ	19			ns
		2.7 V ≤ V _{DD} < 4.0 V, 2.3 V ≤ V _b ≤ 2.7 V, C _b = 30 pF, R _b = 2.7 kΩ	19			ns
		2.7 V ≤ V _{DD} < 3.3 V, 1.6 V ≤ V _b ≤ 2.0 V, C _b = 30 pF, R _b = 5.5 kΩ	19			ns
从 $\overline{SCKp}\downarrow$ 至 SOp 输出的延迟 时间 ^{※1}	t _{KSO1}	4.0 V ≤ V _{DD} ≤ 5.5 V, 2.7 V ≤ V _b ≤ 4.0 V, C _b = 30 pF, R _b = 1.4 kΩ			100	ns
		2.7 V ≤ V _{DD} < 4.0 V, 2.3 V ≤ V _b ≤ 2.7 V, C _b = 30 pF, R _b = 2.7 kΩ			195	ns
		2.7 V ≤ V _{DD} < 3.3 V, 1.6 V ≤ V _b ≤ 2.0 V, C _b = 30 pF, R _b = 5.5 kΩ			483	ns
Slp 建立时间 (至 $\overline{SCKp}\downarrow$) ^{※2}	t _{SIK1}	4.0 V ≤ V _{DD} ≤ 5.5 V, 2.7 V ≤ V _b ≤ 4.0 V, C _b = 30 pF, R _b = 1.4 kΩ	44			ns
		2.7 V ≤ V _{DD} < 4.0 V, 2.3 V ≤ V _b ≤ 2.7 V, C _b = 30 pF, R _b = 2.7 kΩ	44			ns
		2.7 V ≤ V _{DD} < 3.3 V, 1.6 V ≤ V _b ≤ 2.0 V, C _b = 30 pF, R _b = 5.5 kΩ	110			ns
Slp 保持时间 (从 $\overline{SCKp}\downarrow$) ^{※2}	t _{KS1}	4.0 V ≤ V _{DD} ≤ 5.5 V, 2.7 V ≤ V _b ≤ 4.0 V, C _b = 30 pF, R _b = 1.4 kΩ	19			ns
		2.7 V ≤ V _{DD} < 4.0 V, 2.3 V ≤ V _b ≤ 2.7 V, C _b = 30 pF, R _b = 2.7 kΩ	19			ns
		2.7 V ≤ V _{DD} < 3.3 V, 1.6 V ≤ V _b ≤ 2.0 V, C _b = 30 pF, R _b = 5.5 kΩ	19			ns
从 $\overline{SCKp}\uparrow$ 至 SOp 输出的延迟 时间 ^{※2}	t _{KSO1}	4.0 V ≤ V _{DD} ≤ 5.5 V, 2.7 V ≤ V _b ≤ 4.0 V, C _b = 30 pF, R _b = 1.4 kΩ			25	ns
		2.7 V ≤ V _{DD} < 4.0 V, 2.3 V ≤ V _b ≤ 2.7 V, C _b = 30 pF, R _b = 2.7 kΩ			25	ns
		2.7 V ≤ V _{DD} < 3.3 V, 1.6 V ≤ V _b ≤ 2.0 V, C _b = 30 pF, R _b = 5.5 kΩ			25	ns

(注、注意事项和备注如下页所示。)

CSI 模式连接图(不同电位通信时)

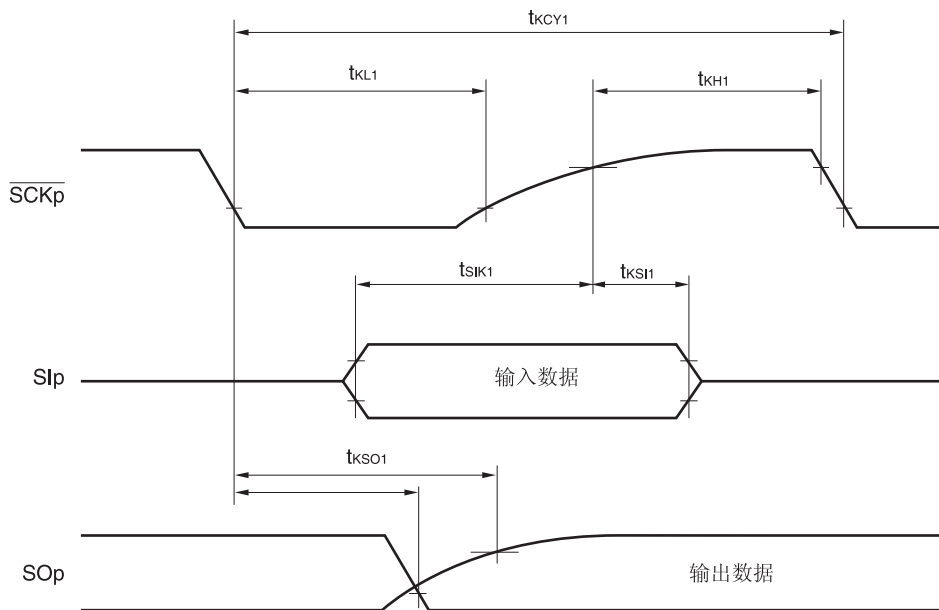


- 注 1. DAPmn = 0 和 CKPmn = 0, 或 DAPmn = 1 和 CKPmn = 1 时。
 2. DAPmn = 0 和 CKPmn = 1, 或 DAPmn = 1 和 CKPmn = 0 时。

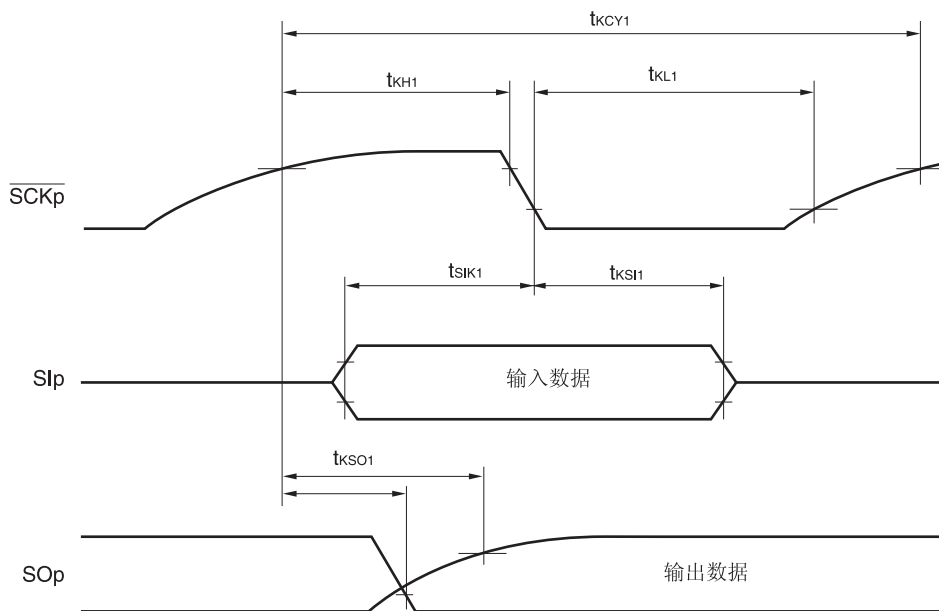
注意事项 使用端口输入模式寄存器 g (PIMg)和端口输出模式寄存器 g (POMg), 将 Slp 引脚选择为 TTL 输入缓冲器, SOp 引脚和 SCKp 引脚选择为 N 沟开漏输出(V_{DD} 耐压)模式。

- 备注 1. $R_b[\Omega]$: 通信线路 (\overline{SCKp} , SOp) 上拉电阻, $C_b[F]$: 通信线路 (\overline{SCKp} , SOp)负载电容, $V_b[V]$: 通信线路电压
 2. p: CSI 编号(p = 00), m: 单元编号(m = 0), n: 通道编号(n = 0), g: PIM 和 POM 编号(g = 1)

CSI 模式串行传送时序(主模式) (不同电位通信时)
 (DAPmn = 0 和 CKPmn = 0, 或 DAPmn = 1 和 CKPmn = 1 时)



CSI 模式串行传送时序(主模式) (不同电位通信时)
 (DAPmn = 0 和 CKPmn = 1, 或 DAPmn = 1 和 CKPmn = 0 时)



注意事项 使用端口输入模式寄存器 g (PIMg)和端口输出模式寄存器 g (POMg), 将 Slp 引脚选择为 TTL 输入缓冲器, SOp 引脚和 SCKp 引脚选择为 N 沟开漏输出(V_{DD} 耐压)模式。

备注 p: CSI 编号(p = 00), m: 单元编号(m = 0), n: 通道编号(n = 0), g: PIM 和 POM 编号(g = 1)

(7) DALI/UART4 模式

(TA = -40 至 +105°C, 2.7 V ≤ VDD ≤ 5.5 V, VSS = 0 V)

参数	符号	条件	最小值	最大值	单元
传送速率				$f_{MCK}/12$ ^{#1,2}	bps
		最大传送速率的理论值 $f_{CLK} = 32 \text{ MHz}, f_{MCK} = f_{CLK}$		2.6	Mbps

注 1. f_{MCK} 是 DALI/UART 的操作时钟频率。

2. SNOOZE 模式下的传送速率为最大值 4800 Mbps。

备注 f_{MCK} : 串行阵列单元的操作时钟频率。

(由串行模式寄存器 mn (SMRmn) 的 CKSmn 位设置的操作时钟。

m: 单元编号, n: 通道编号 (mn = 40))

32.5.2 串行接口IICA

(TA = -40 至+105°C, 2.7 V ≤ VDD ≤ 5.5 V, VSS = 0 V)

参数	符号	条件	标准模式		快速模式		单元
			最小值	最大值	最小值	最大值	
SCLA0 时钟频率	fSCL	快速模式: fCLK ≥ 3.5 MHz			0	400	kHz
		标准模式: fCLK ≥ 1 MHz	2.7 V ≤ VDD ≤ 5.5 V	0	100		
再开始条件的建立时间	tSU:STA		4.7		0.6		μs
保持时间 ^{注1}	tHD:STA		4.0		0.6		μs
SCLA0 = “L”时的保持时间	tLOW		4.7		1.3		μs
SCLA0 = “H”时的保持时间	tHIGH		4.0		0.6		μs
数据建立时间 (接收)	tSU:DAT		250		100		ns
数据保持时间 (传送) ^{注2}	tHD:DAT		0	3.45	0	0.9	μs
停止条件的建立时间	tSU:STO		4.0		0.6		μs
总线自由时间	tBUF		4.7		1.3		μs

注 1. 在检测到开始/再开始条件时, 将在该周期之后产生第一个时钟脉冲。

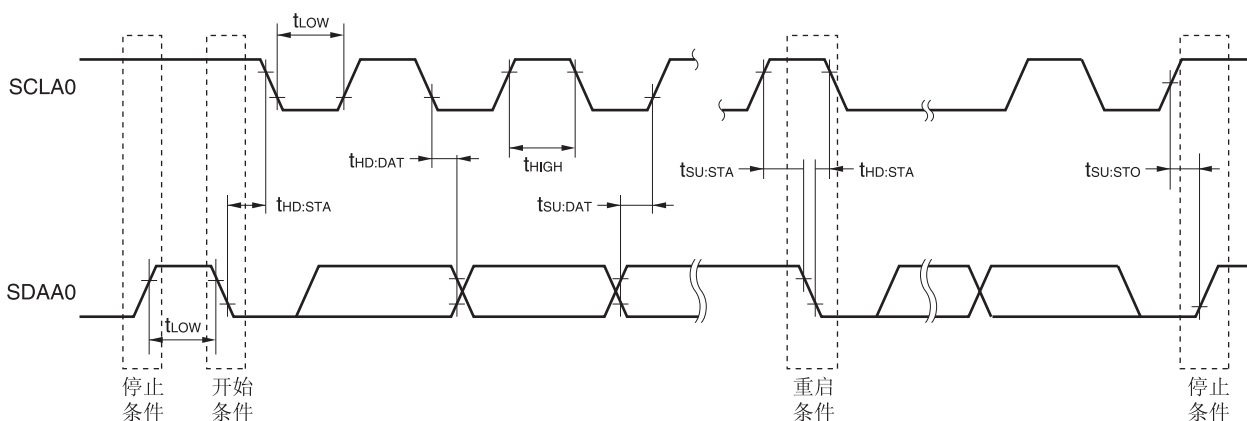
2. tHD:DAT 的最大值为正常传送过程中的值, 在 ACK (应答) 时序中, 需要等待 (从设备的应答)。

备注 此时, 各种模式下的 Cb (通信线路电容) 的最大值以及 Rb (通信线路上拉电阻) 的值如下所示:

标准模式: Cb = 400 pF, Rb = 2.70 kΩ

快速模式: Cb = 320 pF, Rb = 1.1 kΩ

IICA 串行传送时序



32.5.3 片上调试 (UART)

(TA = -40 至+105°C, 2.7 V ≤ VDD ≤ 5.5 V, VSS = 0 V)

参数	符号	条件	最小值	典型值	最大值	单元
传送速率			115.2 k		1 M	bps

32.6 模拟特性

32.6.1 A/D 转换器特性

(1) 选择 AVREF(+) = AVREFP/ANI0 (ADREFP1 = 0, ADREFP0 = 1), AVREF(-) = AVREFM/ANI1 (ADREFM = 1)时, 目标 ANI 引脚: ANI2、ANI14 至 ANI7

(TA = -40 至+105°C, 2.7 V ≤ VDD ≤ 5.5 V, VSS = 0 V, 基准电压(+) = AVREFP, 基准电压(-) = AVREFM = 0 V)

参数	符号	条件	最小值	典型值	最大值	单元	
分辨率	RES		8		10	bit	
总误差 ^{注1,2}	AINL	10 位分辨率		1.2	±3.5	LSB	
转换时间	tCONV	10 位分辨率	3.6 V ≤ VDD ≤ 5.5 V	2.125		39	μs
			2.7 V ≤ VDD < 5.5 V	3.1875		39	μs
零刻度误差 ^{注1,2}	EZS	10 位分辨率			±0.25	%FSR	
满刻度误差 ^{注1,2}	EFS	10 位分辨率			±0.25	%FSR	
积分线性误差 ^{注1}	ILE	10 位分辨率			±2.5	LSB	
微分线性误差 ^{注1}	DLE	10 位分辨率			±1.5	LSB	
基准电压 (+)	AVREFP		2.7		VDD	V	
模拟输入电压	VAIN		0		AVREFP	V	
	VBGR	选择内部基准电压输出 HS (高速主)模式	1.38	1.45	1.5	V	

注 1. 不包括量化误差 (±1/2 LSB)。

2. 该值表示针对满刻度值的比率 (%FSR)。

(2) 当 $AV_{REF(+)} = AV_{REFP}/ANI0$ ($ADREFP1 = 0, ADREFP0 = 1$), $AV_{REF(-)} = AV_{REFM}/ANI1$ ($ADREFM = 1$)时, 目标 ANI 引脚: ANI6 至 ANI19

($T_A = -40$ 至 $+105^\circ\text{C}$, $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$, 基准电压(+) = AV_{REFP} , 基准电压(-) = $AV_{REFM} = 0\text{ V}$)

参数	符号	条件	最小值	典型值	最大值	单元
分辨率	RES		8		10	bit
总误差 ^{注1,2}	AINL	10 位分辨率		1.2	± 5.0	LSB
转换时间	t_{CONV}	10 位分辨率	$3.6\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	2.125	39	μs
			$2.7\text{ V} \leq V_{DD} < 5.5\text{ V}$	3.1875	39	μs
零刻度误差 ^{注1,2}	E_{ZS}	10 位分辨率			± 0.35	%FSR
满刻度误差 ^{注1,2}	E_{FS}	10 位分辨率			± 0.35	%FSR
积分线性误差 ^{注1}	ILE	10 位分辨率			± 3.5	LSB
微分线性误差 ^{注1}	DLE	10 位分辨率			± 2.0	LSB
基准电压(+)	AV_{REFP}		2.7		V_{DD}	V
模拟输入电压	V_{AIN}		0		AV_{REFP} 和 V_{DD}	V
	V_{BGR}	选择内部基准电压输出 HS (高速主)模式	1.38	1.45	1.5	V

注 1. 不包括量化误差 ($\pm 1/2$ LSB)。

2. 该值表示针对满刻度值的比率 (%FSR)。

(3) 当 $AV_{REF(+)} = V_{DD}$ ($ADREFP1 = 0, ADREFP0 = 0$), $AV_{REF(-)} = V_{SS}$ ($ADREFM = 0$)时, 目标 ANI 引脚: ANI0 至 ANI2, ANI4 至 ANI7, ANI16 至 ANI19

($T_A = -40$ 至 $+105^{\circ}\text{C}$, $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$, 基准电压(+) = V_{DD} , 基准电压(-) = V_{SS})

参数	符号	条件		最小值	典型值	最大值	单元
分辨率	RES			8		10	bit
总误差 ^{注1,2}	AINL	10 位分辨率			1.2	± 7.0	LSB
转换时间	t_{CONV}	10 位分辨率	$3.6\text{ V} \leq V_{DD} \leq 5.5\text{ V}$	2.125		39	μs
			$2.7\text{ V} \leq V_{DD} < 5.5\text{ V}$	3.1875		39	μs
零刻度误差 ^{注1,2}	E_{ZS}	10 位分辨率				± 0.60	%FSR
满刻度误差 ^{注1,2}	E_{FS}	10 位分辨率				± 0.60	%FSR
积分线性误差 ^{注1}	ILE	10 位分辨率				± 4.0	LSB
微分线性误差 ^{注1}	DLE	10 位分辨率				± 2.0	LSB
模拟输入电压	V_{AIN}			0		V_{DD}	V
	V_{BGR}	选择内部基准电压输出 HS (高速主)模式		1.38	1.45	1.5	V

注 1. 不包括量化误差 ($\pm 1/2$ LSB)。

2. 该值表示针对满刻度值的比率 (%FSR)。

(4) 当 $AV_{REF(+)} =$ 内部基准电压 ($ADREFP1 = 1, ADREFP0 = 0$), $AV_{REF(-)} = AV_{REFM}/ANI1$ ($ADREFM = 1$)时, 目标 ANI 引脚: ANI0, ANI2, ANI4 至 ANI7, ANI16 至 ANI19

($T_A = -40$ 至 $+105^{\circ}\text{C}$, $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$, 基准电压(+) = V_{BGR} , 基准电压(-) = $AV_{REFM} = 0\text{ V}$, HS (高速主)模式)

参数	符号	条件	最小值	典型值	最大值	单元
分辨率	RES		8			bit
转换时间	t_{CONV}	8 位分辨率	17		39	μs
零刻度误差 ^{注1,2}	E_{ZS}	8 位分辨率			± 0.60	%FSR
积分线性误差 ^{注1}	ILE	8 位分辨率			± 2.0	LSB
微分线性误差 ^{注1}	DLE	8 位分辨率			± 1.0	LSB
基准电压 (+)	V_{BGR}		1.38	1.45	1.5	V
模拟输入电压	V_{AIN}		0		V_{BGR}	V

注 1. 不包括量化误差 ($\pm 1/2$ LSB)。

2. 该值表示针对满刻度值的比率 (%FSR)。

32.6.2 温度传感器特性

($T_A = -40$ 至 $+105^{\circ}\text{C}$, $2.7\text{ V} \leq V_{DD} \leq 5.5\text{ V}$, $V_{SS} = 0\text{ V}$, HS (高速主)模式)

参数	符号	条件	最小值	典型值	最大值	单元
温度传感器输出电压	V_{TMPS25}	设置 ADS 寄存器 = 80H, $T_A = +25^{\circ}\text{C}$		1.05		V
参考输出电压	V_{CONST}	设置 ADS 寄存器 = 81H	1.38	1.45	1.5	V
温度系数	F_{VTMPS}	取决于温度传感器电压的温度		-3.6		mV/C
操作稳定等待时间	t_{AMP}				5	μs

32.6.3 可编程增益放大器

(T_A = -40 至 +105°C, 2.7 V ≤ V_{DD} ≤ 5.5 V, V_{DD} = AV_{REFP}, V_{SS} = AV_{REFM} = 0 V)

参数	符号	条件		最小值	典型值	最大值	单元
输入偏移电压	V _{IOPGA}				±5	±10	mV
输入电压范围	V _{IPGA}			0		0.9V _{DD} /增益	V
增益误差 ^{注1}		4, 8 次				±1	%
		16 次				±1.5	%
		32 次				±2	%
摆动速率 ^{注1}	SRR _{PGA}	上升沿	4.0 V ≤ V _{DD} ≤ 5.5 V	4, 8 次	4		V/μs
				16, 32 次	1.4		V/μs
		2.7 V ≤ V _{DD} < 4.0 V	4, 8 次	1.8		V/μs	
			16, 32 次	0.5		V/μs	
	SR _{FPGA}	下降沿	4.0 V ≤ V _{DD} ≤ 5.5 V	4, 8 次	3.2		V/μs
				16, 32 次	1.4		V/μs
		2.7 V ≤ V _{DD} < 4.0 V	4, 8 次	1.2		V/μs	
			16, 32 次	0.5		V/μs	
操作稳定等待时间 ^{注2}	t _{PGA}	4, 8 次				5	μs
		16, 32 次				10	μs

注 1. 当 V_{IPGA} 为“0.1V_{DD}/增益”至“0.9V_{DD}/增益”。

2. 从允许 PGA 操作(PGAEN = 1)开始, 直到进入能够满足 PGA 操作的 DC 和 AC 特性的状态为止, 所需要的时间。

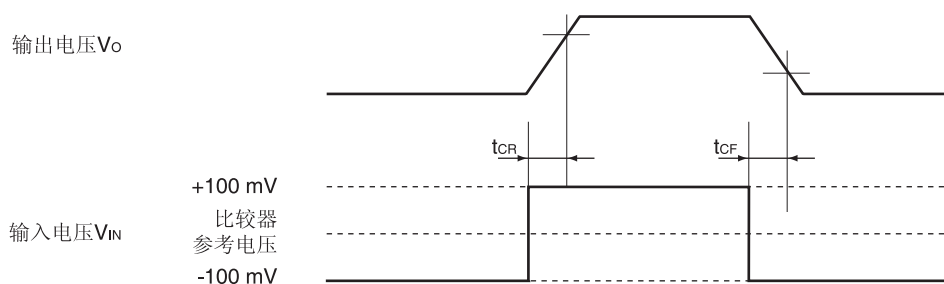
32.6.4 比较器

(T_A = -40 至 +105°C, 2.7 V ≤ V_{DD} ≤ 5.5 V, V_{DD} = AV_{REFP}, V_{SS} = AV_{REFM} = 0 V)

参数	符号	条件	最小值	典型值	最大值	单元
输入偏移电压	V _{IOCOMP}			±5	±40	mV
输入电压范围	V _{ICMP}	CMP0P 至 CMP5P	0		V _{DD}	V
		CMPCOM	0.045		0.9V _{DD}	V
内部基准电压偏差	ΔV _{IREF}	CmRVM 寄存器值: 7FH 至 80H (m = 0 至 2)			±2	LSB
		其它			±1	LSB
响应时间	t _{CR} , t _{CF}	输入振幅 = ±100 mV		70	150	ns
操作稳定等待时间 ^{注1}	t _{CMP}	3.3 V ≤ V _{DD} ≤ 5.5 V			1	μs
		2.7 V ≤ V _{DD} < 3.3 V			3	μs
基准电压稳定等待时间	t _{VR}	CVRE: 0→1 ^{注2}			10	μs

注 1. 从允许操作比较器(CMPnEN 位 = 1; n = 0 至 5)开始, 直到进入能够满足比较器操作的 DC 和 AC 特性的状态为止, 所需要的时间。

2. 允许执行内部基准电压生成(通过将 CVREm 位设置为 1; m = 0 至 2)并且等待操作稳定时间之后, 再允许比较器输出 (CnOE 位 = 1; n = 0 至 5)



32.6.5 POR电路特性

(T_A = -40 至 +105°C, V_{SS} = 0 V)

参数	符号	条件	最小值	典型值	最大值	单元
检测电压	V _{POR}	电源上升时间	1.48	1.51	1.58	V
	V _{PDR}	电源下降时间	1.47	1.50	1.57	V
最小脉冲宽度	T _{PW}		300			μs
检测延迟时间					350	μs

32.6.6 LVD电路特性

复位模式和中断模式的 LVD 检测电压

(TA = -40 至+105°C, VPDR ≤ VDD ≤ 5.5 V, VSS = 0 V)

参数		符号	条件	最小值	典型值	最大值	单元
检测电压	电源电压电位	VLV10	电源上升时间	3.97	4.06	4.14	V
			电源下降时间	3.89	3.98	4.06	V
		VLV11	电源上升时间	3.67	3.75	3.82	V
			电源下降时间	3.59	3.67	3.74	V
		VLV12	电源上升时间	3.06	3.13	3.19	V
			电源下降时间	2.99	3.06	3.12	V
		VLV13	电源上升时间	2.95	3.02	3.08	V
			电源下降时间	2.89	2.96	3.02	V
		VLV14	电源上升时间	2.85	2.92	2.97	V
			电源下降时间	2.79	2.86	2.91	V
		VLV15	电源上升时间	2.75	2.81	2.87	V
			电源下降时间	2.70	2.75	2.81	V
最小脉冲宽度		tLW		300			μs
检测延迟时间						300	μs

备注 VLV1(n-1) > VLV1n: n = 1 至 5

中断和复位模式的 LVD 检测电压

(TA = -40 至+105°C, VPDR ≤ VDD ≤ 5.5 V, VSS = 0 V)

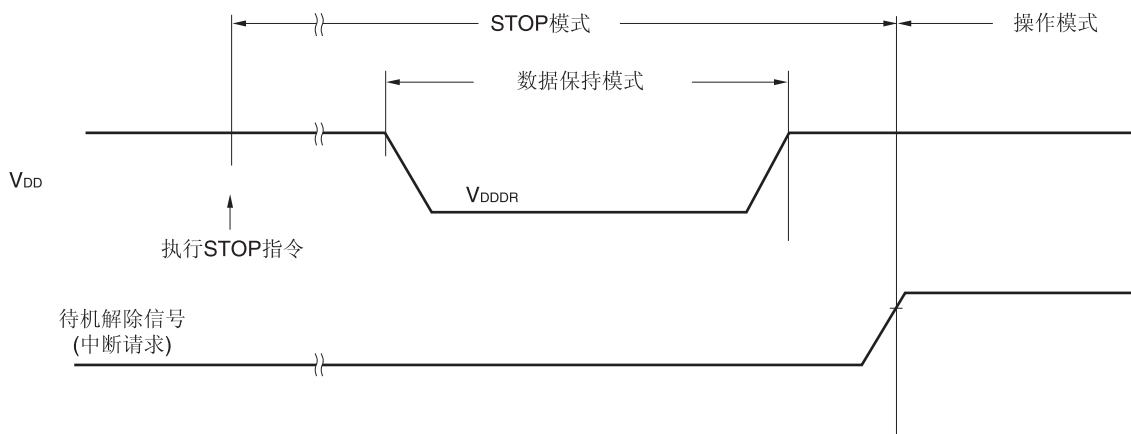
参数	符号	条件	最小值	典型值	最大值	单元	
中断&复位模式	VLV15	VPOC2, VPOC1, VPOC0 = 0, 1, 1, 下降复位电压: 2.7 V	2.70	2.75	2.81	V	
	VLV14	LVIS1, LVIS0 = 1, 0 (+0.1 V)	上升复位解除电压	2.85	2.92	2.97	V
			下降中断电压	2.79	2.86	2.91	V
	VLV13	LVIS1, LVIS0 = 0, 1 (+0.2 V)	上升复位解除电压	2.95	3.02	3.08	V
			下降中断电压	2.89	2.96	3.02	V
	VLV10	LVIS1, LVIS0 = 0, 0 (+1.2 V)	上升复位解除电压	3.97	4.06	4.14	V
			下降中断电压	3.89	3.98	4.06	V

32.7 数据存储器STOP模式低电源电压数据保持特性

(T_A = -40 至+105°C)

参数	符号	条件	最小值	典型值	最大值	单元
数据保持电源电压	V _{DDDR}		1.47 [※]		5.5	V

注 取决于 POR 检测电压。当电压下降时，在 POR 复位生效前，数据被保持，但当 POR 复位生效后，则数据不被保持。



32.8 闪存编程特性

(T_A = -40 至+105°C, 2.7 V ≤ V_{DD} ≤ 5.5 V, V_{SS} = 0 V)

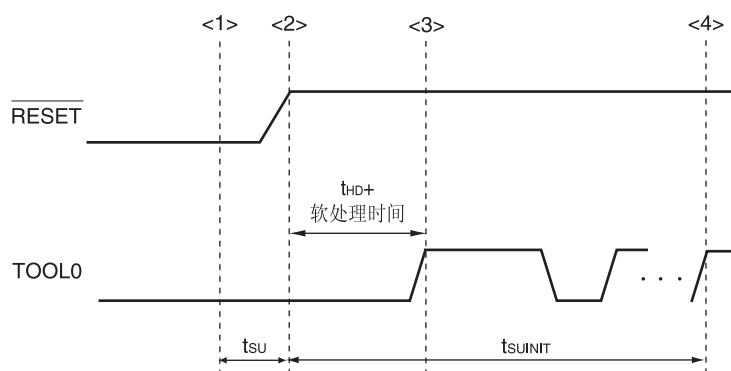
参数	符号	条件	最小值	典型值	最大值	单元
CPU/外围硬件时钟频率	f _{CLK}	2.7 V ≤ V _{DD} ≤ 5.5 V	1		32	MHz
代码闪存重写次数	C _{enwr}	擦除 1 次 + 擦除后写入 1 次被认为是 1 次重写。 保持年限为重写之后到下次重写为止的间隔时间。	保持 20 年 (自编程/串行编程) [※]			次
数据闪存重写次数			保持 1 年 (自编程/串行编程) [※]		1,000,000	
			保持 5 年 (自编程/串行编程) [※]	100,000		

注 当使用闪存编程器和瑞萨电子自编程库时。

备注 在多次更新数据时，请将闪存用于数据的更新。

32.9 切换闪存编程模式的时序规格

参数	符号	条件	最小值	典型值	最大值	单元
从解除引脚复位到完成初始通信设置为止的时间	$t_{SUIINIT}$	必须在解除引脚复位之前解除 POR 和 LVD 复位。			100	ms
从 TOOL0 引脚被设置为低电平到解除引脚复位的时间	t_{SU}	必须在解除引脚复位之前解除 POR 和 LVD 复位。	10			μs
解除复位后 TOOL0 引脚保持低电平的时间 (软件处理时间除外)	t_{HD}	必须在解除引脚复位之前解除 POR 和 LVD 复位。	1			ms



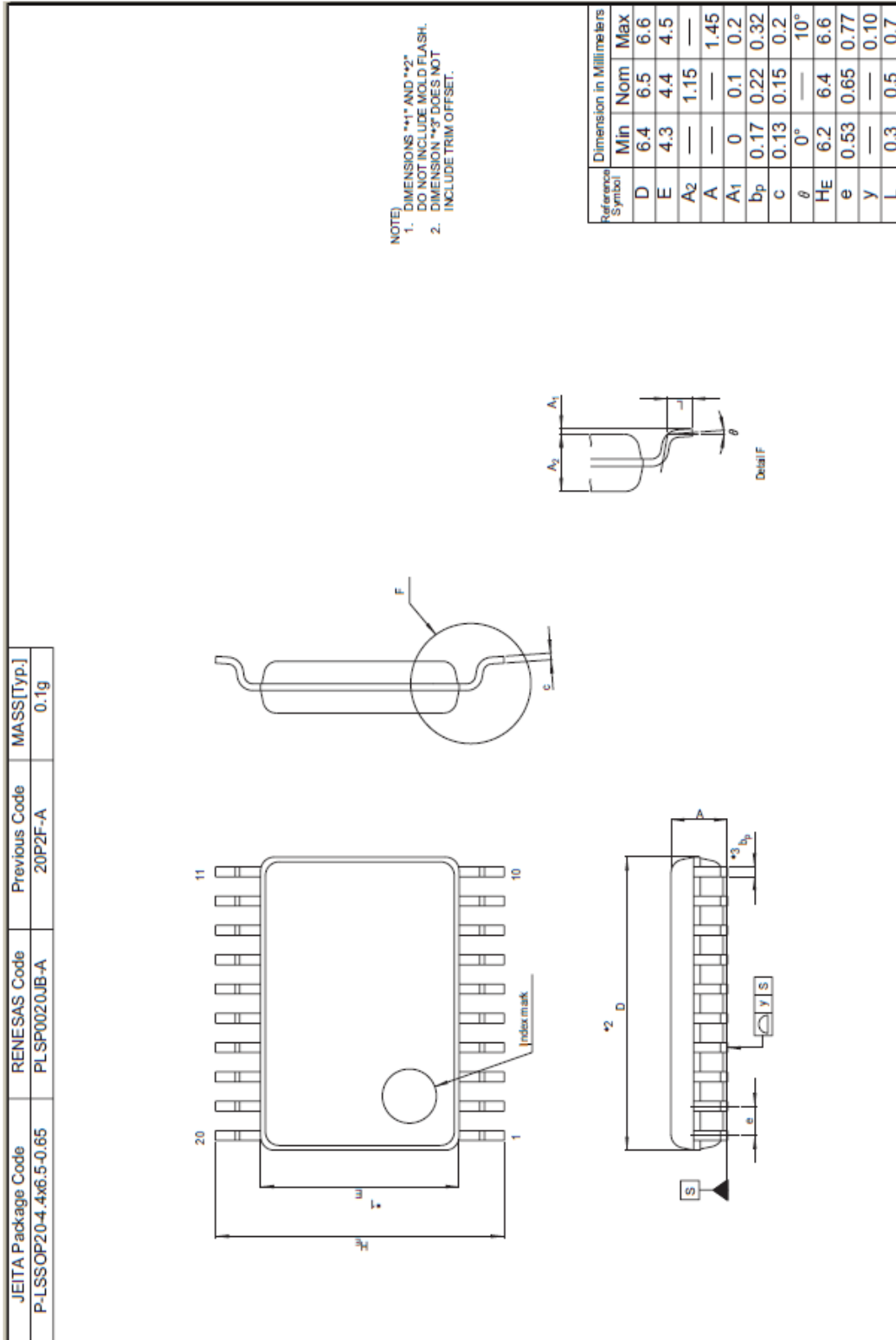
- <1> 低电位输入到 TOOL0 引脚。
 <2> 引脚复位解除（必须在解除引脚复位之前解除 POR 和 LVD 复位）。
 <3> 将 TOOL0 引脚设置为高电位。
 <4> 通过 UART 接收设置闪存编程模式，并完成波特率设置。

备注 $t_{SUIINIT}$: 该段表示在解除复位后须在 100 ms 内完成初始通信设置的指定。
 t_{SU} : 从 TOOL0 引脚被设置为低电平到解除引脚复位的时间
 t_{HD} : 解除外部和内部复位后，将 TOOL0 引脚保持为低电平的时间(软件处理时间除外)

第三十三章 封装尺寸图

33.1 20 引脚产品

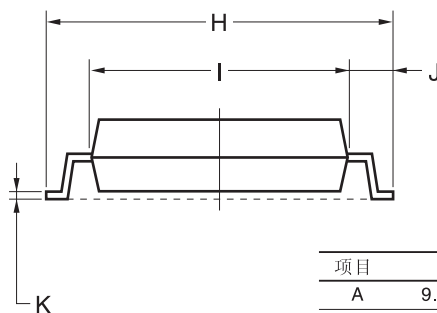
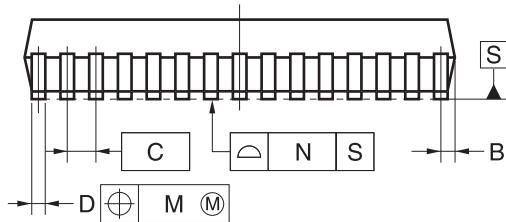
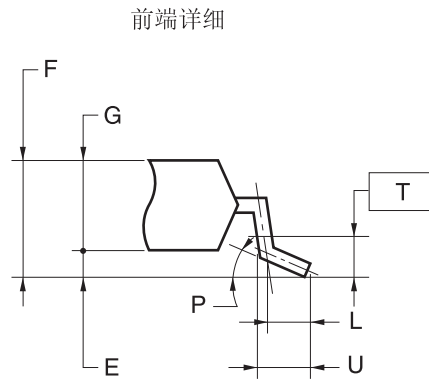
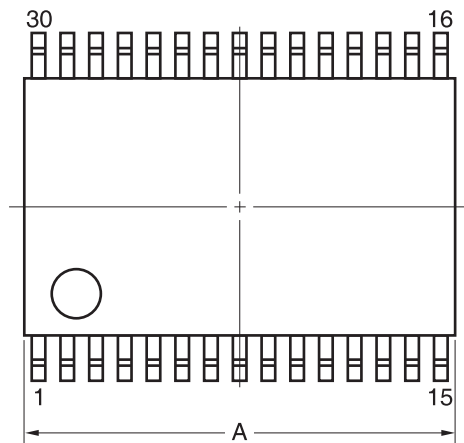
R5F1076CGSP



33.2 30 引脚产品

R5F107ACGSP, R5F107AEGSP

JEITA封装代码	RENESAS代码	先前代码	重量(典型值) [g]
P-LSSOP30-0300-0.65	PLSP0030JB-B	S30MC-65-5A4-3	0.18



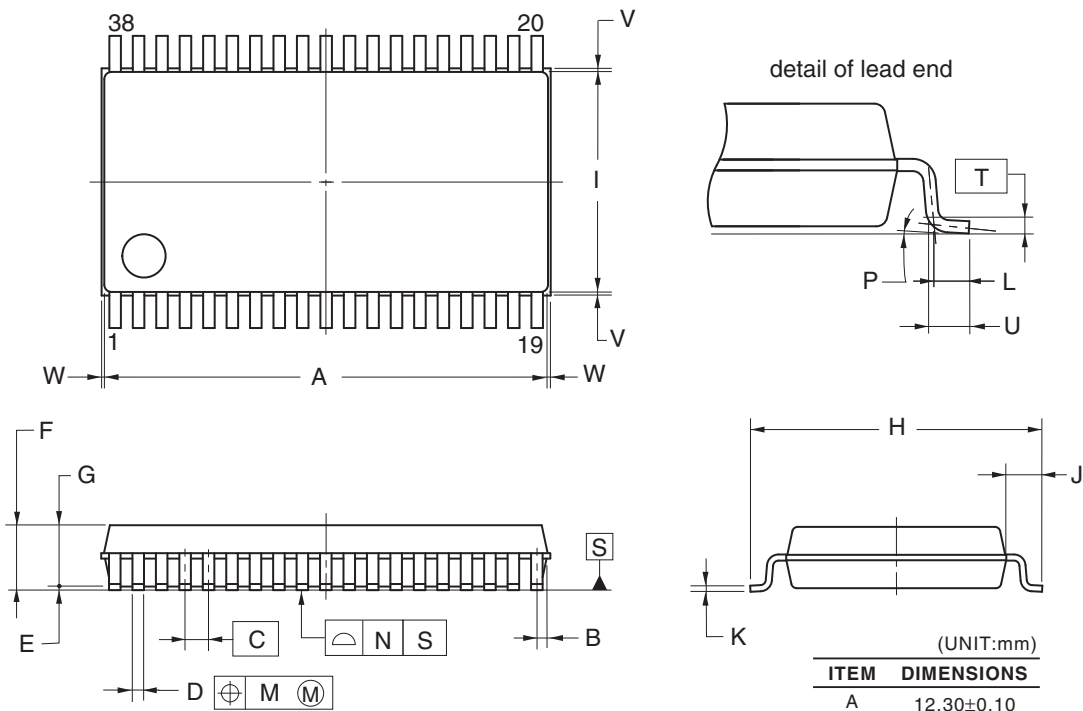
注：
在最大实体状态(MMC)下，各前端中心线在其实际位置(T.P.)的0.13毫米之间。

项目	规格
A	9.85±0.15
B	0.45 最大值
C	0.65 (T.P.)
D	0.24 ^{+0.08} _{-0.07}
E	0.1±0.05
F	1.3±0.1
G	1.2
H	8.1±0.2
I	6.1±0.2
J	1.0±0.2
K	0.17±0.03
L	0.5
M	0.13
N	0.10
P	3° ^{+5°} _{-3°}
T	0.25
U	0.6±0.15

33.4 38 引脚产品

R5F107DEGSP

JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP.) [g]
P-SSOP38-6.1x12.3-0.65	PRSP0038JA-B	P38MC-65-GAA-2	0.3



NOTE
 Each lead centerline is located within 0.10 mm of its true position (T.P.) at maximum material condition .

(UNIT:mm)

ITEM	DIMENSIONS
A	12.30±0.10
B	0.30
C	0.65 (T.P.)
D	0.30 ^{+0.10} _{-0.05}
E	0.125±0.075
F	2.00 MAX.
G	1.70±0.10
H	8.10±0.20
I	6.10±0.10
J	1.00±0.20
K	0.15 ^{+0.10} _{-0.05}
L	0.50
M	0.10
N	0.10
P	3° ^{+5°} _{-3°}
T	0.25(T.P.)
U	0.60±0.15
V	0.25 MAX.
W	0.15 MAX.

第三十四章 关于等待的注意事项

34.1 关于等待的注意事项

访问分配到扩展特殊功能寄存器(2nd SFR)区域中的 F0500H 至 F06FFH 地址的寄存器时，CPU 不开始处理下一个指令，而是变为等待状态。因此，当发生这种等待状态时，执行指令所需的时钟数就会增长等待时钟数。

34.2 产生等待的外围硬件

通过 CPU 执行存取时产生等待请求的寄存器列表以及 CPU 等待时钟数如表 34-1 所示。

表 34-1. 产生等待的寄存器以及 CPU 等待时钟数(1/2)

外围硬件	寄存器	存取	等待时钟数
时钟发生电路	PLLCTL, PER1, PER2 寄存器	读取	1 个时钟
		写入	1 个时钟
端口功能	P20, PM20, PU20, POM20 寄存器	读取	1 个时钟
		写入	1 个时钟
16 位定时器 KB0 至 KB2	TPS2, TKBCRn0 至 TKBCRn3, TKBTGCRn, TKBCTLn0, TKBCTLn1, TKBIOCn0, TKBIOCn1, TKBFLGn, TKBTRGn, TKBCLRn, TKBCRLDn0, TKBCRLDn1,TKBDNRn0, TKBDNRn1, TKBSIRn0, TKBSIRn1, TKBSSRn0, TKBSSRn1, TKBMFRn, TKBPACTLn0, TKBPACTLn1, TKBPACTLn2, TKBP AFLGn, TKBPAHFSn, TKBPAHFTn 寄存器 (n = 0 至 2)	读取	1 个时钟 (定时器 KB0 至 KB2: 64 MHz 操作; CPU: 16 MHz 操作)
			2 个时钟 (定时器 KB0 至 KB2: 64 MHz 操作; CPU: 32 MHz 操作)
		写入	1 个时钟

表 34-1. 产生等待的寄存器以及 CPU 等待时钟数(2/2)

外围硬件	寄存器	存取	等待时钟数
16 位定时器 KC0	TKCCR0, TKCDUTY00 至 TKCDUTY05, TKCCTL0, TKCIOC00, TKCIOC01, TKCTOF0, TKCFLG0, TKCTRG0 寄存器	读取	1 个时钟 (定时器 KC0: 64 MHz 操作; CPU: 16 MHz 操作)
			2 个时钟 (定时器 KC0: 64 MHz 操作; CPU: 32 MHz 操作)
		写入	1 个时钟
可编程增益放大器(PGA)	PGACTL, PGAINS 寄存器	读取	1 个时钟 (PGA: 64 MHz 操作; CPU: 16 MHz 操作)
			2 个时钟 (PGA: 64 MHz 操作; CPU: 32 MHz 操作)
		写入	1 个时钟
比较器	CnCTL, CVRCTL, CmRVM, CMPEGP0, CMPEGN0, CMPMON, CMPWDC, CMPSEL 寄存 器 (n = 0 至 5, m = 0 至 2)	读取	1 个时钟 (比较器: 64 MHz 操作; CPU: 16 MHz 操作)
			2 个时钟 (PGA: 64 MHz 操作; CPU: 32 MHz 操作)
		写入	1 个时钟
串行阵列单元 4 (DALI/UART4)	SDTL4, SDTH4, SDCL4, SDCH4, SDR40, SDR41, SSR40, SSR41, SIR40, SIR41, SMR40, SMR41, SCR40, SCR41, SE4, SS4, ST4, SPS4, SO4, SOE4, SOL4, SSC4, SOC4, NFEN3, SUCTL 寄 存器	读取	1 个时钟
		写入	1 个时钟
中断功能	EGP2, EGN2, INTPCTL, INTMK0, INTMF0 寄存器	读取	1 个时钟
		写入	1 个时钟
安全功能	IAWCTL1 寄存器	读取	1 个时钟
		写入	1 个时钟
其它功能	PIOR1, PFSEL0 寄存器	读取	1 个时钟
		写入	1 个时钟

附录 A 修订记录

A.1 修订记录

版本	说明	章
Rev.1.00	初次制作 2012.09.11	

[Memo]

[Memo]

RL78/I1A 用户手册 硬件篇

Publication Date: Rev.1.00 Sep 11, 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: +44-1628-651-700, Fax: +44-1628-651-804

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.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

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

RL78/I1A



瑞萨电子株式会社

R01UH0169CJ0100