

R7F0C003M2DFB、 R7F0C004M2DFB

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不同，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、输入/输出设定和寄存器的内容。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，原则上应先在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输入/输出上拉电源。因为输入信号或提供输入/输出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

- 对 象
- 本手册以理解 R7F0C003、R7F0C004 的功能并且以设计和开发其应用系统和程序的用户工程师为对象。
- 目 的
- 本手册以帮助用户理解下述结构中所示的功能为目的。
- 构 成
- R7F0C003、R7F0C004 的手册分为用户手册硬件篇（本手册）和用户手册软件篇（RL78 族通用）共 2 本。

R7F0C003M2DFB、R7F0C004M2DFB
用户手册 硬件篇

RL78 族
用户手册 软件篇

- 引脚功能
 - 内部块功能
 - 中断
 - 其他的内部外围功能
 - 电特性
- CPU 功能
 - 指令集
 - 指令的说明

- 阅读方法
- 阅读本手册的读者应具备电气、逻辑电路以及微控制器的基础知识。
- ☐ 要理解全部功能时

→ 请按照目录的顺序阅读本手册。

☐ 寄存器格式的阅读方法

→ 关于方框（□）内的位号，其位名称在 汇编程序中被定义为保留字，而在编译程序中被 #pragma sfr 指令定义为 sfr 变量。

☐ 要详细了解 R7F0C003、R7F0C004 的指令功能时

→ 请参照另一本手册 《RL78 family User's Manual: Software》（R01US0015E）。

- 凡 例
- 数据表示法：

左侧为高位，右侧为低位。

有效低电平表示法：

xxx （在引脚或者信号名称上标注上划线）

注：

正文中加“注”的说明

注意：

需要留心阅读的内容

备注：

正文的补充说明

数制表示法：

二进制 XXXX 或者 XXXXB

十进制 XXXX

十六进制 XXXXH

相关资料 相关资料中可能包括暂定版。但是，在以下资料中并未特别注明“暂定版”，请谅解。

产品的相关资料

资料名称	资料号
R7F0C003M2DFB、R7F0C004M2DFB 用户手册 硬件篇	本手册
RL78 family User's Manual: Software	R01US0015E

闪存编程器的相关资料（用户手册）

资料名称	资料号
PG-FP5 Flash Memory Programmer User's Manual	R20UT0008E

其他资料

资料名称	资料号
Renesas MPUs & MCUs RL78 Family	R01CP0003E
半导体封装安装手册	注
Quality Grades on NEC Semiconductor Devices	C11531E
Guide to Prevent Damage for Semiconductor Derices by Electrostatic Discharge (ESD)	C11892E
Semiconductor Reliability Handbook	R51ZZ0001E

注 请参阅“半导体封装安装手册”网页。
<http://cn.renesas.com/products/package/manual/index.jsp>

注意 上述相关资料的内容如有变更，恕不另行通知。设计等时请使用最新版本的资料。

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。

SuperFlash 是美国 Silicon Storage Technology, Inc. 在美国以及日本等国的注册商标。

注意：本产品使用已获得 Silicon Storage Technology, Inc. 授权的 SuperFlash®。

目 录

第 1 章	概述	1
1.1	特点	1
1.2	产品型号一览	3
1.3	引脚配置（俯视图）	4
1.4	引脚名称	5
1.5	框图	6
1.6	功能概要	7
第 2 章	引脚功能	9
2.1	端口功能	9
2.2	端口以外的引脚功能	12
2.3	引脚的输入 / 输出电路和未使用引脚的处理	14
第 3 章	CPU 体系结构	21
3.1	存储空间	21
3.1.1	内部程序存储器空间	25
3.1.2	镜像区	28
3.1.3	内部数据存储空间	29
3.1.4	特殊功能寄存器（SFR: Special Function Register）的区域	30
3.1.5	扩展特殊功能寄存器（2nd SFR: 2nd Special Function Register）的区域	30
3.1.6	数据存储器的寻址	30
3.2	处理器寄存器	32
3.2.1	控制寄存器	32
3.2.2	通用寄存器	34
3.2.3	ES 寄存器和 CS 寄存器	35
3.2.4	特殊功能寄存器（SFR: Special Function Register）	36
3.2.5	扩展特殊功能寄存器（2nd SFR: 2nd Special Funtion Register）	43
3.3	指令地址寻址	52
3.3.1	相对寻址	52
3.3.2	立即寻址	52
3.3.3	表间接寻址	53
3.3.4	寄存器直接寻址	53
3.4	处理数据地址的寻址	54
3.4.1	隐含寻址	54
3.4.2	寄存器寻址	54
3.4.3	直接寻址	55
3.4.4	短直接寻址	56
3.4.5	SFR 寻址	57
3.4.6	寄存器间接寻址	58
3.4.7	基址寻址	59
3.4.8	基址变址寻址	63
3.4.9	堆栈寻址	64
第 4 章	端口功能	68
4.1	端口功能	68
4.2	端口结构	68
4.2.1	端口 0	68
4.2.2	端口 1	75
4.2.3	端口 2	80

4.2.4	端口 3	84
4.2.5	端口 4	89
4.2.6	端口 5	97
4.2.7	端口 6	104
4.2.8	端口 7	106
4.2.9	端口 12	108
4.2.10	端口 13	113
4.3	控制端口功能的寄存器	116
4.3.1	端口模式寄存器 (PMxx)	120
4.3.2	端口寄存器 (Pxx)	121
4.3.3	上拉电阻选择寄存器 (PUxx)	122
4.3.4	端口输入模式寄存器 (PIMxx)	123
4.3.5	端口输出模式寄存器 (POMxx)	124
4.3.6	端口模式控制寄存器 (PMCxx)	125
4.3.7	A/D 端口配置寄存器 (ADPC)	126
4.3.8	外围 I/O 重定向寄存器 (PIOR)	127
4.3.9	LCD 端口功能寄存器 0 ~ 6 (PFSEG0 ~ PFSEG6)	128
4.3.10	LCD 输入切换控制寄存器 (ISLCD)	130
4.4	端口功能的运行	131
4.4.1	输入 / 输出端口的写操作	131
4.4.2	输入 / 输出端口的读操作	131
4.4.3	输入 / 输出端口的运算	131
4.4.4	不同电位 (1.8V、2.5V、3V) 的输入 / 输出缓冲器对应	132
4.5	使用复用功能时的寄存器设定	134
4.5.1	使用复用功能时的基本观点	134
4.5.2	不使用输出功能的复用功能的寄存器设定	135
4.5.3	使用的端口功能和复用功能的寄存器设定例子	135
4.5.4	SEGxx 引脚的复用端口的运行	143
4.5.5	V _{L3} 、CAPL、CAPH 引脚的复用端口的运行	145
4.5.6	P45 (模拟输入引脚 (IVREF0) 的复用端口)	146
4.6	使用端口功能时的注意事项	147
4.6.1	有关对端口寄存器 n (Pn) 的位操作指令的注意事项	147
4.6.2	引脚设定时的注意事项	148
第 5 章	时钟发生电路	149
5.1	时钟发生电路的功能	149
5.2	时钟发生电路的结构	151
5.3	控制时钟发生电路的寄存器	154
5.3.1	时钟运行模式控制寄存器 (CMC)	154
5.3.2	系统时钟控制寄存器 (CKC)	157
5.3.3	时钟运行状态控制寄存器 (CSC)	158
5.3.4	振荡稳定时间计数器的状态寄存器 (OSTC)	159
5.3.5	振荡稳定时间选择寄存器 (OSTS)	160
5.3.6	外围允许寄存器 0、1 (PER0、PER1)	161
5.3.7	副系统时钟提供模式控制寄存器 (OSMC)	164
5.3.8	高速内部振荡器频率选择寄存器 (HOCODIV)	165
5.3.9	高速内部振荡器的微调寄存器 (HIOTRM)	166
5.4	系统时钟振荡电路	167
5.4.1	X1 振荡电路	167
5.4.2	XT1 振荡电路	167
5.4.3	高速内部振荡器	171
5.4.4	低速内部振荡器	171

5.5	时钟发生电路的运行	171
5.6	时钟控制	173
5.6.1	高速内部振荡器的设定例子	173
5.6.2	X1 振荡电路的设定例子	174
5.6.3	XT1 振荡电路的设定例子	175
5.6.4	CPU 时钟状态转移图	176
5.6.5	CPU 时钟转移前的条件和转移后的处理	182
5.6.6	切换 CPU 时钟和系统时钟所需要的时间	184
5.6.7	时钟振荡停止前的条件	185
5.7	谐振器和振荡电路常数	186
第 6 章	定时器阵列单元	190
6.1	定时器阵列单元的功能	191
6.1.1	独立通道运行功能	191
6.1.2	多通道联动运行功能	193
6.1.3	8 位定时器运行功能（只限通道 1 和通道 3）	194
6.1.4	支持 LIN-bus 功能（只限通道 7）	194
6.2	定时器阵列单元的结构	195
6.2.1	定时器计数寄存器 mn（TCRmn）	201
6.2.2	定时器数据寄存器 mn（TDRmn）	202
6.3	控制定时器阵列单元的寄存器	203
6.3.1	外围允许寄存器 0（PER0）	204
6.3.2	定时器时钟选择寄存器 m（TPSm）	205
6.3.3	定时器模式寄存器 mn（TMRmn）	208
6.3.4	定时器状态寄存器 mn（TSRmn）	213
6.3.5	定时器通道允许状态寄存器 m（TEm）	214
6.3.6	定时器通道开始寄存器 m（TSm）	215
6.3.7	定时器通道停止寄存器 m（TTm）	216
6.3.8	定时器输入选择寄存器 0（TIS0）	217
6.3.9	定时器输出允许寄存器 m（TOEm）	218
6.3.10	定时器输出寄存器 m（TOM）	219
6.3.11	定时器输出电平寄存器 m（TOLm）	220
6.3.12	定时器输出模式寄存器 m（TOMm）	221
6.3.13	输入切换控制寄存器（ISC）	222
6.3.14	噪声滤波器允许寄存器 1（NFEN1）	223
6.3.15	定时器输入 / 输出引脚的端口功能控制寄存器	224
6.4	定时器阵列单元的基本规则	225
6.4.1	多通道联动运行功能的基本规则	225
6.4.2	8 位定时器运行功能的基本规则（只限通道 1 和通道 3）	226
6.5	计数器的操作	227
6.5.1	计数时钟（f _{TCLK} ）	227
6.5.2	计数器的开始时序	229
6.5.3	计数器的运行	230
6.6	通道输出（TOMn 引脚）的控制	235
6.6.1	TOMn 引脚输出电路的结构	235
6.6.2	TOMn 引脚的输出设定	236
6.6.3	通道输出运行的注意事项	237
6.6.4	TOMn 位的一次性操作	240
6.6.5	有关开始计数时的定时器中断和 TOMn 引脚输出	241

6.7	定时器输入 (TImn) 的控制	242
6.7.1	TImn 的输入电路结构	242
6.7.2	噪声滤波器	242
6.7.3	进行通道输入操作时的注意事项	243
6.8	定时器阵列单元的独立通道运行功能	244
6.8.1	作为间隔定时器 / 方波输出的运行	244
6.8.2	作为外部事件计数器的运行	249
6.8.3	作为输入脉冲间隔测量的运行	253
6.8.4	作为输入信号高 / 低电平宽度测量的运行	257
6.8.5	作为延迟计数器的运行	261
6.9	定时器阵列单元的多通道联动运行功能	265
6.9.1	作为单触发脉冲输出功能的运行	265
6.9.2	作为 PWM 功能的运行	272
6.9.3	作为多重 PWM 输出功能的运行	279
6.10	使用定时器阵列单元时的注意事项	286
6.10.1	使用定时器输出时的注意事项	286
第 7 章	实时时钟 2	287
7.1	实时时钟 2 的功能	287
7.2	实时时钟 2 的结构	288
7.3	控制高精度实时时钟的寄存器	290
7.3.1	外围允许寄存器 0 (PER0)	291
7.3.2	外围允许寄存器 1 (PER1)	292
7.3.3	副系统时钟提供模式控制寄存器 (OSMC)	293
7.3.4	上电复位状态控制寄存器 (PORSR)	294
7.3.5	实时时钟控制寄存器 0 (RTCC0)	295
7.3.6	实时时钟控制寄存器 1 (RTCC1)	297
7.3.7	秒计数寄存器 (SEC)	299
7.3.8	分钟计数寄存器 (MIN)	299
7.3.9	小时计数寄存器 (HOUR)	300
7.3.10	日计数寄存器 (DAY)	302
7.3.11	星期计数寄存器 (WEEK)	303
7.3.12	月计数寄存器 (MONTH)	304
7.3.13	年计数寄存器 (YEAR)	304
7.3.14	时钟误差校正寄存器 (SUBCUD)	305
7.3.15	闹钟分钟寄存器 (ALARMWM)	307
7.3.16	闹钟小时寄存器 (ALARMWH)	307
7.3.17	闹钟星期寄存器 (ALARMWW)	308
7.3.18	控制实时时钟 2 输出引脚的端口功能的寄存器	308
7.4	实时时钟 2 的运行	309
7.4.1	开始实时时钟 2 的运行	309
7.4.2	开始运行后向 HALT/STOP 模式的转移	310
7.4.3	读实时时钟 2 的计数器	311
7.4.4	写实时时钟 2 的计数器	312
7.4.5	实时时钟 2 的闹钟设定	313
7.4.6	实时时钟 2 的 1Hz 输出	314
7.4.7	时钟误差校正寄存器的设定步骤	315
7.4.8	实时时钟 2 的时钟误差校正例子	316
7.4.9	有关高精度 1Hz 输出	318

第 8 章	副系统时钟频率测量电路	319
8.1	副系统时钟频率测量电路	319
8.2	副系统时钟频率测量电路的结构	319
8.3	控制副系统时钟频率测量电路的寄存器	320
8.3.1	外围允许寄存器 1 (PER1)	320
8.3.2	副系统时钟提供模式控制寄存器 (OSMC)	321
8.3.3	频率测量计数寄存器 L (FMCRL)	321
8.3.4	频率测量计数寄存器 H (FMCRLH)	322
8.3.5	频率测量控制寄存器 (FMCTL)	323
8.4	副系统时钟频率测量电路的运行	324
8.4.1	副系统时钟频率测量电路的设定	324
8.4.2	副系统时钟频率测量电路的运行时序	325
第 9 章	12 位间隔定时器	326
9.1	12 位间隔定时器的功能	326
9.2	12 位间隔定时器的结构	326
9.3	控制 12 位间隔定时器的寄存器	327
9.3.1	外围允许寄存器 1 (PER1)	327
9.3.2	副系统时钟提供模式控制寄存器 (OSMC)	328
9.3.3	12 位间隔定时器控制寄存器 (ITMC)	329
9.4	12 位间隔定时器的运行	330
9.4.1	12 位间隔定时器的运行时序	330
9.4.2	从 HALT/STOP 模式返回后计数器开始运行并且再次转移到 HALT/STOP 模式	331
第 10 章	时钟输出 / 蜂鸣器输出控制电路	332
10.1	时钟输出 / 蜂鸣器输出控制电路的功能	332
10.2	时钟输出 / 蜂鸣器输出控制电路的结构	333
10.3	控制时钟输出 / 蜂鸣器输出控制电路的寄存器	333
10.3.1	时钟输出选择寄存器 n (CKSn)	333
10.3.2	副系统时钟提供模式控制寄存器 (OSMC)	335
10.3.3	控制时钟输出 / 蜂鸣器输出引脚的端口功能的寄存器	335
10.4	时钟输出 / 蜂鸣器输出控制电路的运行	336
10.4.1	输出引脚的运行	336
10.5	时钟输出 / 蜂鸣器输出控制电路的注意事项	336
第 11 章	看门狗定时器	337
11.1	看门狗定时器的功能	337
11.2	看门狗定时器的结构	337
11.3	控制看门狗定时器的寄存器	339
11.3.1	看门狗定时器的允许寄存器 (WDTE)	339
11.4	看门狗定时器的运行	340
11.4.1	看门狗定时器的运行控制	340
11.4.2	看门狗定时器上溢时间的设定	341
11.4.3	看门狗定时器窗口打开期间的设定	342
11.4.4	看门狗定时器间隔中断的设定	343
第 12 章	A/D 转换器	344
12.1	A/D 转换器的功能	344
12.2	A/D 转换器的结构	346
12.3	控制 A/D 转换器的寄存器	348
12.3.1	外围允许寄存器 0 (PER0)	349
12.3.2	A/D 转换器的模式寄存器 0 (ADM0)	350

12.3.3	A/D 转换器的模式寄存器 1 (ADM1)	358
12.3.4	A/D 转换器的模式寄存器 2 (ADM2)	359
12.3.5	10 位 A/D 转换结果寄存器 (ADCR)	362
12.3.6	8 位 A/D 转换结果寄存器 (ADCRH)	362
12.3.7	模拟输入通道指定寄存器 (ADS)	363
12.3.8	转换结果比较上限值设定寄存器 (ADUL)	364
12.3.9	转换结果比较下限值设定寄存器 (ADLL)	364
12.3.10	A/D 测试寄存器 (ADTES)	365
12.3.11	模拟输入引脚的端口功能控制寄存器	365
12.4	A/D 转换器的转换运行	366
12.5	输入电压和转换结果	368
12.6	A/D 转换器的运行模式	369
12.6.1	软件触发模式 (连续转换模式)	369
12.6.2	软件触发模式 (单次转换模式)	370
12.6.3	硬件触发无等待模式 (连续转换模式)	371
12.6.4	硬件触发无等待模式 (单次转换模式)	372
12.6.5	硬件触发等待模式 (连续转换模式)	373
12.6.6	硬件触发等待模式 (单次转换模式)	374
12.7	A/D 转换器的设定流程图	375
12.7.1	软件触发模式的设定	375
12.7.2	硬件触发无等待模式的设定	376
12.7.3	硬件触发等待模式的设定	377
12.7.4	使用温度传感器时的设定 (以软件触发模式、单次转换模式为例)	378
12.7.5	测试模式的设定	379
12.8	SNOOZE 模式功能	380
12.9	A/D 转换器特性表的阅读方法	383
12.10	A/D 转换器的注意事项	386
第 13 章	比较器	390
13.1	比较器的功能	390
13.2	比较器的结构	390
13.3	控制比较器的寄存器	391
13.3.1	外围允许寄存器 1 (PER1)	391
13.3.2	比较器模式设定寄存器 (COMPMDR)	392
13.3.3	比较器滤波器控制寄存器 (COMPFIR)	394
13.3.4	比较器输出控制寄存器 (COMPOCR)	395
13.3.5	控制模拟输入 / 输出引脚的端口功能的寄存器	396
13.4	运行说明	397
13.4.1	比较器 i 的数字滤波器 (i=0、1)	399
13.4.2	比较器 i 中断 (i=0、1)	399
13.4.3	比较器 i 输出 (i=0、1)	399
13.4.4	停止 / 提供比较器时钟	400
第 14 章	串行阵列单元	401
14.1	串行阵列单元的功能	402
14.1.1	3 线串行 I/O (CSI00)	402
14.1.2	UART (UART0 ~ UART3)	403
14.1.3	简易 I ² C (IIC00、IIC10)	404
14.2	串行阵列单元的结构	405
14.2.1	移位寄存器	408
14.2.2	串行数据寄存器 mn (SDRmn) 的低 8 位或者低 9 位	408

14.3	控制串行阵列单元的寄存器	410
14.3.1	外围允许寄存器 0 (PER0)	411
14.3.2	串行时钟选择寄存器 m (SPSm)	412
14.3.3	串行模式寄存器 mn (SMRmn)	413
14.3.4	串行通信操作设定寄存器 mn (SCRmn)	415
14.3.5	串行数据寄存器 mn (SDRmn)	417
14.3.6	串行标志清除触发寄存器 mn (SIRmn)	419
14.3.7	串行状态寄存器 mn (SSRmn)	420
14.3.8	串行通道开始寄存器 m (SSm)	422
14.3.9	串行通道停止寄存器 m (STm)	423
14.3.10	串行通道允许状态寄存器 m (SEm)	424
14.3.11	串行输出允许寄存器 m (SOEm)	425
14.3.12	串行输出寄存器 m (SOM)	426
14.3.13	串行输出电平寄存器 m (SOLm)	427
14.3.14	串行待机控制寄存器 m (SSCm)	429
14.3.15	输入切换控制寄存器 (ISC)	430
14.3.16	噪声滤波器允许寄存器 0 (NFEN0)	431
14.3.17	控制串行输入 / 输出引脚的端口功能的寄存器	432
14.4	运行停止模式	433
14.4.1	以单元为单位停止运行的情况	433
14.4.2	按通道停止运行的情况	434
14.5	3 线串行 I/O (CSI00) 通信的运行	435
14.5.1	主控发送	436
14.5.2	主控接收	444
14.5.3	主控的发送和接收	452
14.5.4	从属发送	460
14.5.5	从属接收	468
14.5.6	从属发送和接收	474
14.5.7	SNOOZE 模式功能	482
14.5.8	传送时钟频率的计算	486
14.5.9	在 3 线串行 I/O (CSI00) 通信过程中发生错误时的处理步骤	488
14.6	UART (UART0 ~ UART3) 通信的运行	489
14.6.1	UART 发送	491
14.6.2	UART 接收	500
14.6.3	SNOOZE 模式功能	507
14.6.4	波特率的计算	514
14.6.5	UART (UART0 ~ UART3) 通信过程中发生错误时的处理步骤	518
14.7	LIN 通信的运行	519
14.7.1	LIN 发送	519
14.7.2	LIN 接收	522
14.8	简易 I ² C (IIC00、IIC10) 通信的运行	526
14.8.1	地址字段发送	527
14.8.2	数据发送	532
14.8.3	数据接收	535
14.8.4	停止条件的产生	539
14.8.5	传送速率的计算	540
14.8.6	在简易 I ² C (IIC00、IIC10) 通信过程中发生错误时的处理步骤	542
第 15 章	串行接口 IICA	543
15.1	串行接口 IICA 的功能	543
15.2	串行接口 IICA 的结构	546
15.3	控制串行接口 IICA 的寄存器	549

15.3.1	外围允许寄存器 0 (PER0)	549
15.3.2	IICA 控制寄存器 n0 (IICCTLn0)	550
15.3.3	IICA 状态寄存器 n (IICSn)	555
15.3.4	IICA 标志寄存器 n (IICFn)	557
15.3.5	IICA 控制寄存器 n1 (IICCTLn1)	559
15.3.6	IICA 低电平宽度设定寄存器 n (IICWLn)	561
15.3.7	IICA 高电平宽度设定寄存器 n (IICWHn)	561
15.3.8	端口模式寄存器 6 (PM6)	562
15.4	I ² C 总线模式的功能	563
15.4.1	引脚结构	563
15.4.2	通过 IICWLn 寄存器和 IICWHn 寄存器设定传送时钟的方法	564
15.5	I ² C 总线的定义和控制方法	566
15.5.1	开始条件	566
15.5.2	地址	567
15.5.3	传送方向的指定	567
15.5.4	应答 (ACK)	568
15.5.5	停止条件	569
15.5.6	等待	570
15.5.7	等待的解除方法	572
15.5.8	中断请求 (INTIICAn) 的产生时序和等待控制	573
15.5.9	地址匹配的检测方法	574
15.5.10	错误的检测	574
15.5.11	扩展码	575
15.5.12	仲裁	576
15.5.13	唤醒功能	578
15.5.14	通信预约	581
15.5.15	其他注意事项	584
15.5.16	通信运行	585
15.5.17	I ² C 中断请求 (INTIICAn) 的产生时序	592
15.6	时序图	613
第 16 章	智能卡接口 (SMCI)	628
16.1	智能卡接口 (SMCI) 的功能	628
16.2	智能卡接口 (SMCI) 的结构	628
16.2.1	接收移位寄存器 n	629
16.2.2	发送移位寄存器 n	629
16.2.3	智能卡接收数据寄存器 n (SCRDRn)	630
16.2.4	智能卡发送数据寄存器 n (SCTDRn)	630
16.3	控制智能卡接口 SMCI 的寄存器	631
16.3.1	外围允许寄存器 1 (PER1)	631
16.3.2	智能卡串行模式寄存器 n (SCMRn)	632
16.3.3	智能卡串行控制寄存器 n (SCCRn)	634
16.3.4	智能卡串行状态寄存器 n (SCSRn)	636
16.3.5	智能卡位速率寄存器 n (SCBRRn)	639
16.3.6	端口输入模式寄存器 5 (PIM5)	640
16.3.7	端口输出模式寄存器 5 (POM5)	640
16.3.8	端口模式寄存器 5 (PM5)	641
16.3.9	端口寄存器 5 (P5)	641
16.4	智能卡接口 (SMCI) 的运行	642
16.4.1	与 IC 卡的连接例子	642
16.4.2	数据格式	643
16.4.3	接收数据的噪声滤波器	645

16.4.4	接收数据采样时序	646
16.4.5	SMCI 的初始化	647
16.4.6	串行数据的发送（通常模式）	648
16.4.7	串行数据的接收（通常模式）	650
16.4.8	时钟输出的控制	652
16.4.9	使用 DMA 进行的连续数据发送	654
16.4.10	使用 DMA 进行的连续数据接收	656
16.4.11	中断源	658
16.5	使用时的注意事项	659
16.5.1	外围允许寄存器 1（PER1）的设置	659
16.5.2	SCTDR 的写操作与 SCTDRE 标志的关系	659
16.5.3	使用 DMA 时的限制事项	659
16.5.4	向低功耗状态转移时的运行	660
16.5.5	允许接收前的 SCIO _n 引脚状态	661
16.5.6	不使用 SMCI 功能的情况	661
第 17 章	LCD 控制器 / 驱动器	662
17.1	LCD 控制器 / 驱动器的功能	663
17.2	LCD 控制器 / 驱动器的结构	664
17.3	控制 LCD 控制器 / 驱动器的寄存器	666
17.3.1	LCD 模式寄存器 0（LCDM0）	667
17.3.2	LCD 模式寄存器 1（LCDM1）	669
17.3.3	副系统时钟提供模式控制寄存器（OSMC）	671
17.3.4	LCD 时钟控制寄存器 0（LCDC0）	672
17.3.5	LCD 升压电平控制寄存器（VLCD）	673
17.3.6	LCD 输入切换控制寄存器（ISCLCD）	674
17.3.7	LCD 端口功能寄存器 0 ~ 6（PFSEG0 ~ PFSEG6）	676
17.3.8	端口模式寄存器 0 ~ 5、7、13（PM0 ~ PM5、PM7、PM13）	681
17.4	LCD 显示数据寄存器	682
17.5	LCD 显示寄存器的选择	686
17.5.1	A 图形区和 B 图形区的数据显示	686
17.5.2	闪烁显示（A 图形区和 B 图形区的数据的交替显示）	687
17.6	LCD 控制器 / 驱动器的设定	688
17.7	运行停止步骤	690
17.8	LCD 驱动电压 V_{L1} 、 V_{L2} 、 V_{L3} 和 V_{L4} 的提供	691
17.8.1	外部电阻分压方式	691
17.8.2	内部升压方式	693
17.9	公共信号和段信号	694
17.10	显示模式	700
17.10.1	静态显示例子	700
17.10.2	2 个时间片的显示例子	703
17.10.3	3 个时间片的显示例子	706
17.10.4	4 个时间片的显示例子	710
17.10.5	6 个时间片的显示例子	714
17.10.6	8 个时间片的显示例子	717
第 18 章	乘除器和乘加器	721
18.1	乘除器和乘加器的功能	721
18.2	乘除器和乘加器的结构	721
18.2.1	乘除数据寄存器 A（MDAH、MDAL）	723
18.2.2	乘除数据寄存器 B（MDBL、MDBH）	724
18.2.3	乘除数据寄存器 C（MDCL、MDCH）	725

18.3 控制乘除器和乘加器的寄存器	727
18.3.1 乘除控制寄存器 (MDUC)	727
18.4 乘除器和乘加器的运算	729
18.4.1 乘法运算 (无符号)	729
18.4.2 乘法运算 (带符号)	730
18.4.3 乘加运算 (无符号)	731
18.4.4 乘加运算 (带符号)	732
18.4.5 除法运算	734
第 19 章 DMA 控制器	735
19.1 DMA 控制器的功能	735
19.2 DMA 控制器的结构	736
19.2.1 DMA SFR 地址寄存器 n (DSAn)	736
19.2.2 DMA RAM 地址寄存器 n (DRAn)	737
19.2.3 DMA 字节计数寄存器 n (DBCn)	738
19.3 控制 DMA 控制器的寄存器	739
19.3.1 DMA 模式控制寄存器 n (DMCn)	739
19.3.2 DMA 运行控制寄存器 n (DRCn)	742
19.4 DMA 控制器的运行	743
19.4.1 运行步骤	743
19.4.2 传送模式	744
19.4.3 DMA 传送的结束	744
19.5 DMA 控制器的设定例子	745
19.5.1 CSI 连续发送	745
19.5.2 A/D 转换结果的连续读取	747
19.5.3 UART 连续接收 +ACK 发送	748
19.5.4 通过 DWAITn 位保留 DMA 传送	749
19.5.5 通过软件强制结束	750
19.6 DMA 控制器的注意事项	752
第 20 章 中断功能	754
20.1 中断功能的种类	754
20.2 中断源和结构	754
20.3 控制中断功能的寄存器	759
20.3.1 中断请求标志寄存器 (IF0L、IF0H、IF1L、IF1H、IF2L、IF2H、IF3L)	762
20.3.2 中断屏蔽标志寄存器 (MK0L、MK0H、MK1L、MK1H、MK2L、MK2H、MK3L)	764
20.3.3 优先级指定标志寄存器 (PR00L、PR00H、PR01L、PR01H、PR02L、PR02H、PR03L、 PR10L、PR10H、PR11L、PR11H、PR12L、PR12H、PR13L)	765
20.3.4 外部中断上升沿允许寄存器 (EGP0)、外部中断下降沿允许寄存器 (EGN0)	767
20.3.5 程序状态字 (PSW)	768
20.4 中断处理的操作	769
20.4.1 接受可屏蔽中断请求	769
20.4.2 接受软件中断请求	771
20.4.3 多重中断处理	771
20.4.4 中断请求的保留	775
第 21 章 待机功能	776
21.1 待机功能	776
21.2 待机功能的控制寄存器	777
21.3 待机功能的运行	777
21.3.1 HALT 模式	777
21.3.2 STOP 模式	782
21.3.3 SNOOZE 模式	787

第 22 章 复位功能	790
22.1 复位运行的时序	792
22.2 复位期间的运行状态	794
22.3 确认复位源的寄存器	796
22.3.1 复位控制标志寄存器 (RESF)	796
22.3.2 上电复位状态寄存器 (PORSR)	797
第 23 章 上电复位电路	799
23.1 上电复位电路的功能	799
23.2 上电复位电路的结构	800
23.3 上电复位电路的运行	800
第 24 章 电压检测电路	804
24.1 电压检测电路的功能	804
24.2 电压检测电路的结构	805
24.3 电压检测电路的控制寄存器	806
24.3.1 电压检测寄存器 (LVIM)	806
24.3.2 电压检测电平选择寄存器 (LVIS)	807
24.4 电压检测电路的运行	810
24.4.1 用作复位模式时的设定	810
24.4.2 用作中断模式时的设定	812
24.4.3 用作中断 & 复位模式时的设定	814
24.5 电压检测电路的注意事项	818
第 25 章 安全功能	820
25.1 安全功能的概要	820
25.2 安全功能使用的寄存器	821
25.3 安全功能的运行	821
25.3.1 闪存 CRC 运算功能 (高速 CRC)	821
25.3.2 CRC 运算功能 (通用 CRC)	824
25.3.3 RAM 奇偶校验错误检测功能	826
25.3.4 RAM 保护功能	827
25.3.5 SFR 保护功能	828
25.3.6 非法存储器存取检测功能	829
25.3.7 频率检测功能	831
25.3.8 A/D 测试功能	833
25.3.9 输入 / 输出引脚的数字输出信号电平检测功能	836
第 26 章 稳压器	837
26.1 稳压器的概述	837
第 27 章 选项字节	838
27.1 选项字节的功能	838
27.1.1 用户选项字节 (000C0H ~ 000C2H/010C0H ~ 010C2H)	838
27.1.2 片上调试选项字节 (000C3H/010C3H)	839
27.2 用户选项字节的格式	840
27.3 片上调试选项字节的格式	844
27.4 选项字节的设定	845
第 28 章 闪存	846
28.1 通过闪存编程器进行的串行编程	848
28.1.1 编程环境	849

28.1.2	通信方式	849
28.2	通过外部器件（内置 UART）进行的串行编程	851
28.2.1	编程环境	851
28.2.2	通信方式	851
28.3	电路板上的引脚处理	852
28.3.1	P40/TOOL0 引脚	852
28.3.2	RESET 引脚	853
28.3.3	端口引脚	853
28.3.4	REGC 引脚	853
28.3.5	X1 引脚和 X2 引脚	853
28.3.6	电源	853
28.4	串行编程方法	854
28.4.1	串行编程步骤	854
28.4.2	闪存编程模式	854
28.4.3	通信模式	856
28.4.4	通信命令	856
28.5	自编程	858
28.5.1	自编程步骤	859
28.5.2	引导交换功能	860
28.5.3	闪存屏蔽窗口功能	862
28.6	安全功能	863
第 29 章	片上调试功能	865
29.1	与 E1 片上调试仿真器的连接	865
29.2	片上调试安全 ID	866
29.3	用户资源的确保	866
第 30 章	十进制校正（BCD）电路	868
30.1	十进制校正电路的功能	868
30.2	十进制校正电路使用的寄存器	868
30.2.1	BCD 校正结果寄存器（BCDADJ）	868
30.3	十进制校正电路的运行	869
第 31 章	指令集的概要	871
31.1	凡例	871
31.1.1	操作数的表现形式和记述方法	871
31.1.2	操作栏的说明	872
31.1.3	标志栏的说明	873
31.1.4	PREFIX 指令	873
31.2	操作一览表	874
第 32 章	电特性	891
32.1	绝对最大额定值	891
32.2	振荡电路特性	894
32.2.1	X1、XT1 振荡电路特性	894
32.2.2	内部振荡器特性	894
32.3	DC 特性	895
32.3.1	引脚特性	895
32.3.2	电源电流特性	900
32.4	AC 特性	906
32.5	外围功能特性	910
32.5.1	串行阵列单元	910

32.5.2	串行接口 IICA	929
32.5.3	智能卡接口 (SMCI)	932
32.6	模拟特性	933
32.6.1	A/D 转换器特性	933
32.6.2	温度传感器 / 内部基准电压特性	935
32.6.3	比较器	936
32.6.4	POR 电路特性	936
32.6.5	LVD 电路特性	937
32.6.6	电源电压上升斜率特性	938
32.7	LCD 特性	939
32.7.1	外部电阻分压方式	939
32.7.2	内部升压方式	940
32.8	STOP 模式中的数据存储器的低电源电压数据保持特性	942
32.9	闪存编程特性	942
32.10	专用闪存编程器通信 (UART)	942
32.11	闪存编程模式的转移时序	943
第 33 章	封装尺寸图	944
附录 A	修订记录	945

第 1 章 概述

1.1 特点

超低功耗科技

- $V_{DD}=1.6\sim 5.5V$ 的单电源、能以 1.8V 的低电压运行。
- HALT 模式
- STOP 模式
- SNOOZE 模式

RL78 CPU 内核

- 3 段流水线的 CISC 哈佛体系结构
- 最短指令执行时间：能在高速（0.04167 μs ：以高速内部振荡器时钟 24MHz 运行时）到超低速（30.5 μs ：以副系统时钟 32.768kHz 运行时）之间变换。
- 地址空间：1M 字节
- 通用寄存器：8 位寄存器 $\times 8 \times 4$ 组
- 内部 RAM：8KB

代码闪存

- 代码闪存：96KB $\sim$ 128KB
- 块大小：1KB
- 禁止块擦除、改写（安全功能）
- 内置片上调试功能
- 自编程：具有引导交换功能和闪存屏蔽窗口功能

高速内部振荡器

- 可从 24MHz/16MHz/12MHz/8MHz/6MHz/4MHz/3MHz/2MHz/1MHz 中选择
- 高精度 $\pm 1.0\%$ （ $V_{DD}=1.8\sim 5.5V$ 、 $T_A=-20\sim +85^\circ C$ ）

工作环境温度

- $T_A=-40\sim +85^\circ C$

电源管理和复位功能

- 内置上电复位（POR）电路
- 内置电压检测（LVD）电路（从 14 种中选择中断和复位）

DMA（Direct Memory Access）控制器

- 内置 4 个通道
- 8 位/16 位的 SFR 与内部 RAM 间的传送为 2 个时钟

乘除器和乘加器

- 16 位 $\times$ 16 位 = 32 位（有符号/无符号）
- 32 位 $\div$ 32 位 = 32 位（无符号）
- 16 位 $\times$ 16 位 + 32 位 = 32 位（有符号/无符号）

串行接口

- CSI: 1 个通道
- UART/UART（支持 LIN-bus）: 4 个通道/1 个通道
- I²C/简易 I²C: 1 个通道/2 个通道
- 智能卡接口（SMCI）: 2 个通道

定时器

- 16 位定时器: 8 个通道
- 12 位间隔定时器: 1 个通道
- 实时时钟 2: 1 个通道（99 年日历、闹钟功能、时钟校正功能）
- 看门狗定时器: 1 个通道（能以专用低速内部振荡器时钟运行）

A/D 转换器

- 8/10 位分辨率 A/D 转换器（V_{DD}=1.6~5.5V）
- 模拟输入: 4 个通道
- 内部基准电压（1.45V）和内置温度传感器注 1

比较器

- 2 个通道
- 运行模式: 比较器高速模式、比较器低速模式、窗口模式
- 基准电压能从外部基准电压和内部基准电压中选择。

LCD 控制器 / 驱动器

- 段信号输出: 51（47）个注 2
- 公共信号输出: 4（8）个注 2
- 能进行内部升压和外部电阻分压的切换

- 注 1. 只能选择 HS（高速主）模式。
- 2.（）内为使用 8 COM 时的信号输出个数。

输入 / 输出端口

- I/O 端口: 65 个（N 沟道漏极开路输入/输出 [6V 耐压]: 2 个, N 沟道漏极开路输入/输出 [V_{DD} 耐压]: 18 个）
- 能进行 N 沟道漏极开路、TTL 输入缓冲、内部上拉的切换
- 能与不同电位（1.8/2.5/3V）运行的器件连接
- 内置时钟输出/蜂鸣器输出控制电路

其他

- 内置十进制校正（BCD）电路

备注 配置的功能因产品而不同。请参照“1.6 功能概要”。

○ ROM、RAM 容量

闪存 ROM	数据闪存	RAM	产品名称
128KB	—	8KB 注	R7F0C004M2DFB
96KB	—	8KB 注	R7F0C003M2DFB

注 在使用自编程功能时大约为 7KB（详细内容请参照“第 3 章 CPU 体系结构”）。

1.2 产品型号一览

图 1-1 产品型号、存储容量和封装

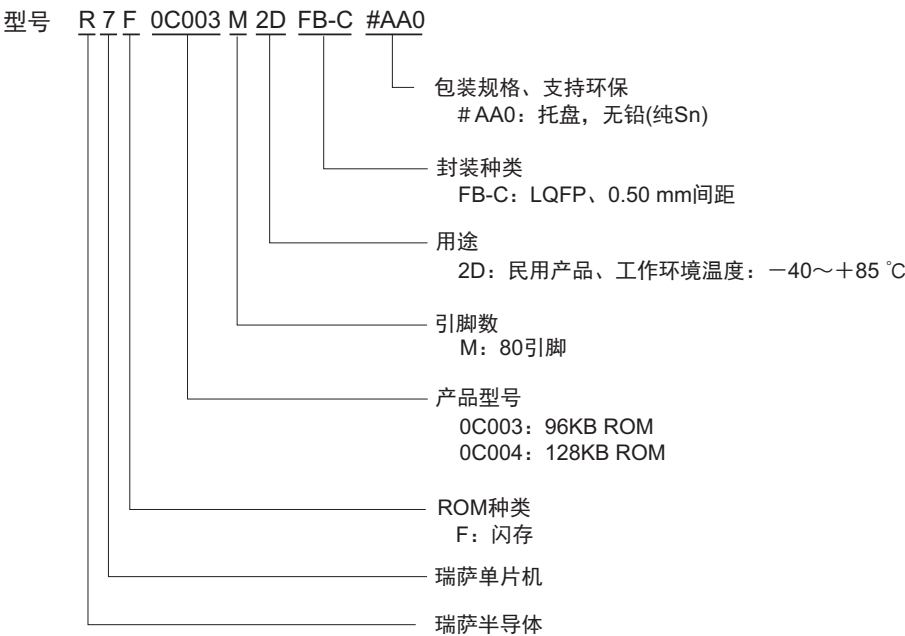
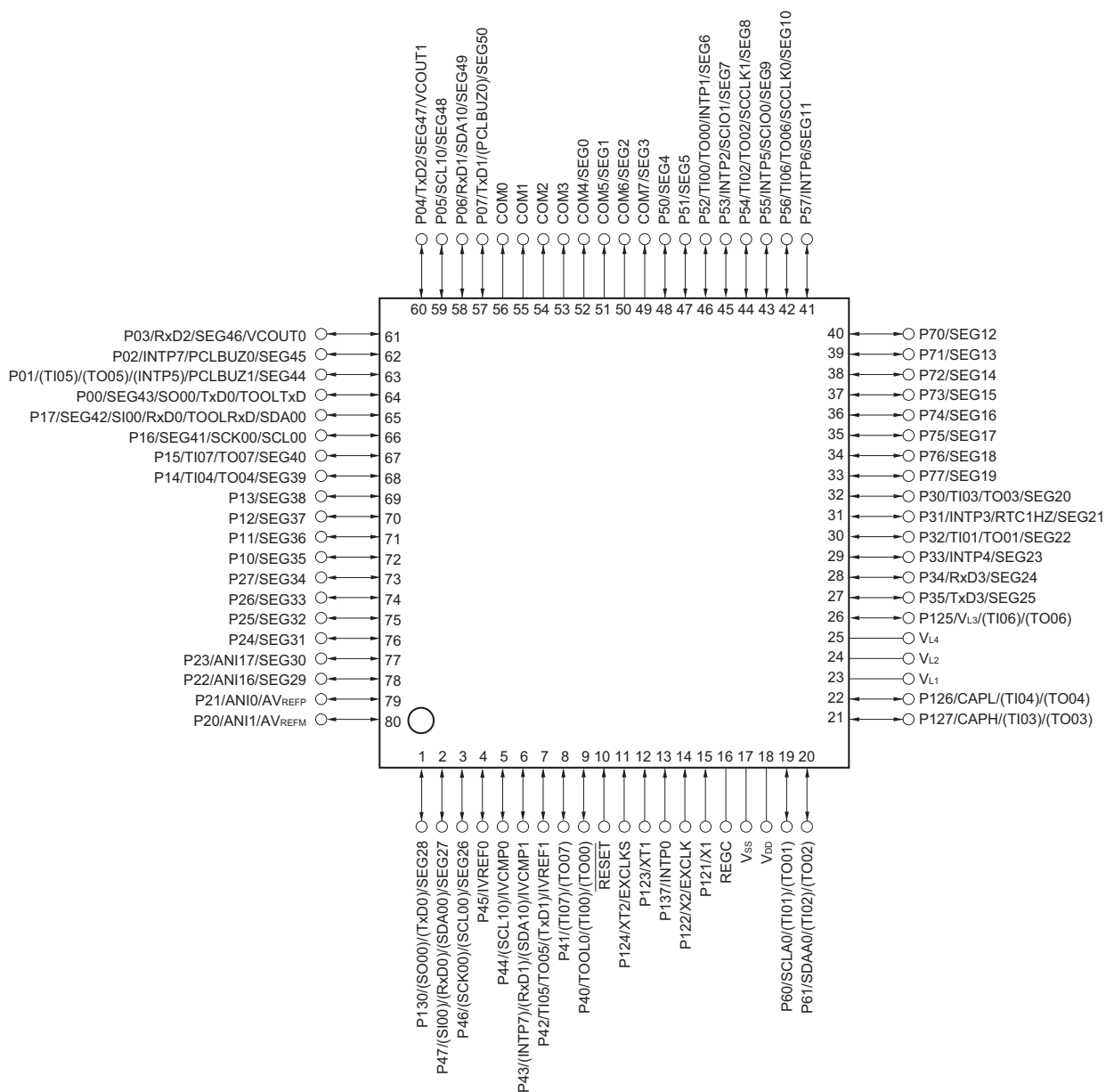


表 1-1 订购型号一览表

引脚数	封装	数据闪存	包装规格、支持环保	订购型号
80 引脚	80 引脚塑封 LQFP (细间距) (12×12)	未内置	托盘、无铅 (纯 Sn)	R7F0C003M2DFB-C#AA0 R7F0C004M2DFB-C#AA0

1.3 引脚配置（俯视图）



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

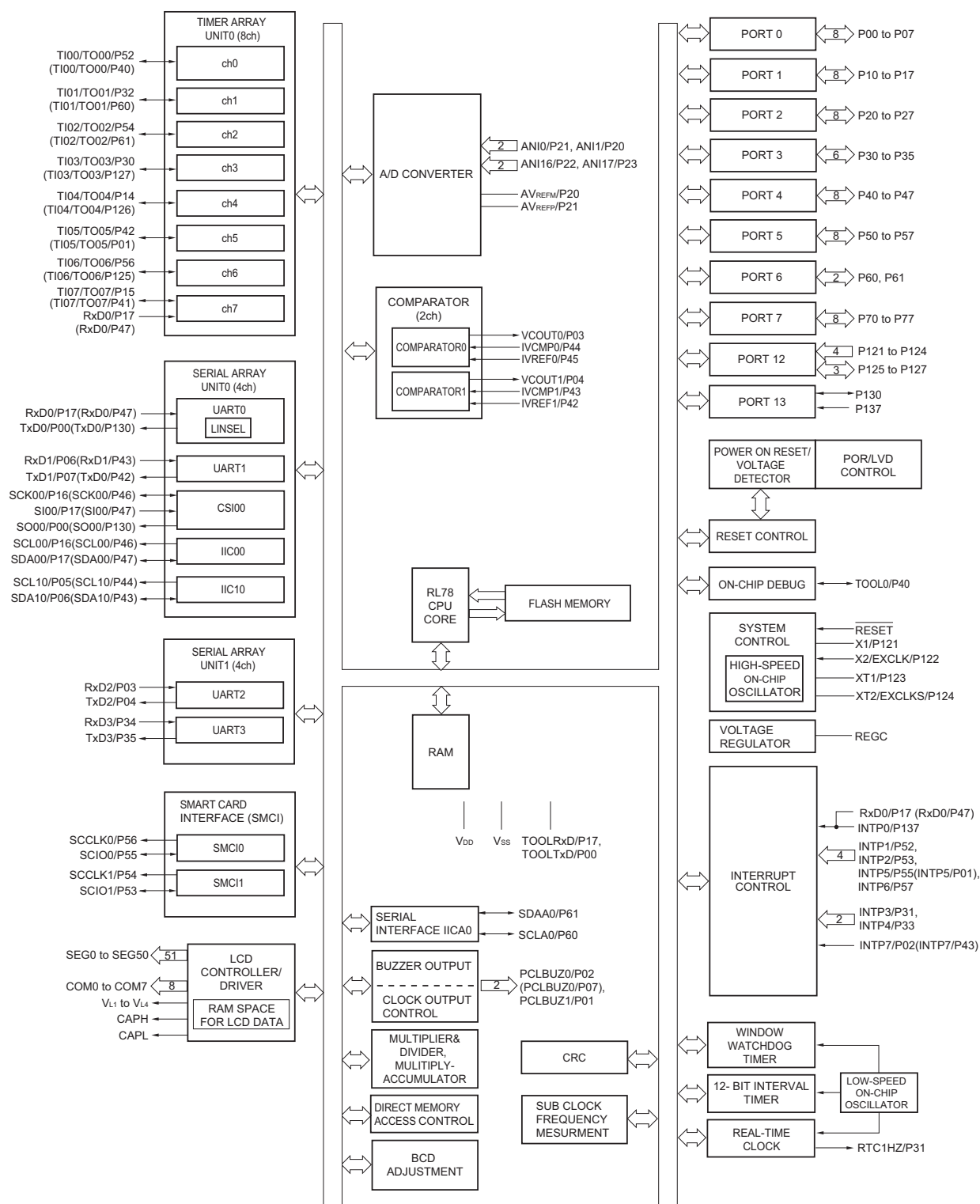
备注 1. 有关引脚名称，请参照“1.4 引脚名称”。

2. 可通过设定外围 I/O 重定向寄存器（PIOR），分配上图（ ）内的功能。详细内容请参照“图 4-42 外围 I/O 重定向寄存器（PIOR）的格式”。

1.4 引脚名称

ANI0、ANI1、 ANI16、ANI17:	Analog Input	PCLBUZ0、PCLBUZ1:	Programmable Clock Output/Buzzer Output
AVREFM:	Analog Reference Voltage Minus	REGC:	Regulator Capacitance
AVREFP:	Analog Reference Voltage Plus	$\overline{\text{RESET}}$:	Reset
CAPH、CAPL:	Capacitor for LCD	RTC1HZ:	Real-time Clock Correction Clock (1Hz) Output
COM0 ~ COM7:	LCD Common Output	RxD0 ~ RxD3:	Receive Data
EXCLK:	External Clock Input (Main System Clock)	SCK00、SCLA0:	Serial Clock Input/Output
EXCLKS:	External Clock Input (Subsystem Clock)	SCL00、SCL10、 SCCLK0、SCCLK1:	Serial Clock Output
INTP0 ~ INTP7:	External Clock Input	SDAA0、SDA00、SDA10、	Serial Data Input/Output
IVCMP0、IVCMP1:	Comparator Input	SCIO0、SCIO1:	
IVREF0、IVREF1:	Comparator Reference Input	SEG0 ~ SEG50:	LCD Segment Output
P00 ~ P07:	Port 0	SI00:	Serial Data Input
P10 ~ P17:	Port 1	SO00:	Serial Data Output
P20 ~ P27:	Port 2	TI00 ~ TI07:	Timer Input
P30 ~ P35:	Port 3	TO00 ~ TO07:	Timer Output
P40 ~ P47:	Port 4	TOOL0:	Data Input/Output for Tool
P50 ~ P57:	Port 5	TOOLRxD、TOOLTxD:	Data Input/Output for External Device
P60、P61:	Port 6	TxD0 ~ TxD3:	Transmit Data
P70 ~ P77:	Port 7	VCOUT0、VCOUT1:	Comparator Output
P121 ~ P127:	Port 12	V _{DD} :	Power Supply
P130、P137:	Port 13	V _{L1} ~ V _{L4} :	LCD Power Supply
		V _{SS} :	Ground
		X1、X2:	Crystal Oscillator (Main System Clock)
		XT1、XT2:	Crystal Oscillator (Subsystem Clock)

1.5 框图



备注 可通过设定外围 I/O 重定向寄存器 (PIOR)，分配上图 () 内的功能。详细内容请参照“图 4-42 外围 I/O 重定向寄存器 (PIOR) 的格式”。

1.6 功能概要

(1/2)

项目		R7F0C003M2DFB	R7F0C004M2DFB
代码闪存		96KB	128KB
数据闪存		—	—
RAM		8KB 注 1	
地址空间		1M 字节	
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） 1～20MHz：V _{DD} =2.7～5.5 V、1～8MHz：V _{DD} =1.8～2.7 V、1～4MHz：V _{DD} =1.6～1.8 V	
	高速内部振荡器时钟	HS（高速主）模式：1～24 MHz（V _{DD} =2.7～5.5 V）、 HS（高速主）模式：1～16 MHz（V _{DD} =2.4～5.5 V）、 LS（低速主）模式：1～8 MHz（V _{DD} =1.8～5.5 V）、 LV（低电压主）模式：1～4 MHz（V _{DD} =1.6～5.5 V）	
副系统时钟		XT1（晶体）振荡，外部副系统时钟输入（EXCLKS） 32.768kHz(TYP.): V _{DD} =1.6～5.5 V	
低速内部振荡器时钟		15kHz(TYP.)	
通用寄存器		(8 位寄存器 × 8 个) × 4 组	
最短指令执行时间		0.04167μs（高速内部振荡器时钟：f _{IH} =24MHz 运行时）	
		0.05μs（高速系统时钟：f _{MX} =20MHz 运行时）	
		30.5μs（副系统时钟：f _{SUB} =32.768kHz 运行时）	
指令集		• 数据传送（8/16 位） • 加减法 / 逻辑运算（8/16 位） • 乘法（8 位 × 8 位） • 旋转、桶式移位和位操作（置位、复位、测试和布尔运算）等	
I/O 端口	共计	65	
	CMOS 输入 / 输出	58 (N 沟道漏极开路输入 / 输出 [V _{DD} 耐压]：18)	
	CMOS 输入	5	
	CMOS 输出	—	
	N 沟道漏极开路输入 / 输出 (6 V 耐压)	2	
定时器	16 位定时器 TAU	8 个通道（定时器输出：8 个，PWM 输出：7 个注 2）	
	看门狗定时器	1 个通道	
	12 位间隔定时器（IT）	1 个通道	
	实时时钟 2	1 个通道	
	RTC 输出	1 个 • 1Hz（副系统时钟：f _{SUB} =32.768kHz）	

注 1. 在 8 KB 的情况下，使用自编程功能时，大约为 7 KB。

2. 输出个数因定时器的个数和所用通道的设定而变（参照“6.9.3 作为多重 PWM 输出功能的运行”）。

(2/2)

项目		R7F0C003M2DFB	R7F0C004M2DFB
副系统时钟频率测量功能		通过外部基准时钟测量副系统时钟的频率。 (用于校正实时时钟 2 的时钟误差)	
时钟输出 / 蜂鸣器输出		2 个 • 2.44kHz、4.88kHz、9.76kHz、1.25MHz、2.5MHz、5MHz、10MHz (主系统时钟: $f_{\text{MAIN}}=20\text{MHz}$ 运行时) • 256Hz、512Hz、1.024kHz、2.048kHz、4.096kHz、8.192kHz、16.384kHz、32.768kHz (副系统时钟: $f_{\text{SUB}}=32.768\text{kHz}$ 运行时)	
8/10 位分辨率 A/D 转换器		4 个通道	
比较器		2 个通道	
串行接口		• CSI: 1 个通道 / UART (支持 UART 的 LIN-bus): 1 个通道 / 简易 I ² C: 1 个通道 • UART: 1 个通道 / 简易 I ² C: 1 个通道 • UART: 2 个通道	
	I ² C 总线	1 个通道	
	智能卡接口 (SMCI)	2 个通道	
LCD 控制器 / 驱动器		可对内部升压方式和外部电阻分压方式进行切换。	
	段信号输出	51 (47) 个注 1	
	公共信号输出	4 (8) 个注 1	
乘除器和乘加器		乘法: 16 位 $\times$ 16 位 = 32 位 (无符号或者有符号) 除法: 32 位 $\div$ 32 位 = 32 位 (无符号) 乘加法: 16 位 $\times$ 16 位 + 32 位 = 32 位 (无符号或者有符号)	
DMA 控制器		4 个通道	
向量中断源	内部	41	
	外部	10	
复位		• 因 RESET 引脚产生复位 • 因看门狗定时器产生内部复位 • 因上电复位产生内部复位 • 因电压检测电路产生内部复位 • 因执行非法指令产生内部复位注 2 • 因 RAM 奇偶校验错误产生内部复位 • 因非法存取存储器产生内部复位	
上电复位电路		• 上电复位: 1.51V(TYP.) • 掉电复位: 1.50V(TYP.)	
电压检测电路		• 上升: 1.67V $\sim$ 4.06V (14 种) • 下降: 1.63V $\sim$ 3.98V (14 种)	
片上调试功能		有	
电源电压		$V_{\text{DD}}=1.6 \sim 5.5\text{V}$	
运行环境温度		$T_{\text{A}}=-40 \sim +85^{\circ}\text{C}$	

注 1. () 中的值为使用 8 COM 时信号输出的个数。

2. 执行指令代码 FFH 时, 产生非法指令。

通过在线仿真器或者片上调试仿真器进行仿真时, 不会因执行非法指令而产生内部复位。

第 2 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。

各个端口中设定的输入 / 输出、缓冲器、上拉电阻对复用功能也有效。

(1/3)

功能名称	输入 / 输出	功能	复位后	复用功能
P00	输入 / 输出	端口 0。 8 位输入 / 输出端口。 P03、P05 和 P06 的输入可以设定为 TTL 输入缓冲器。 P00、P04 ~ P07 的输出可以设定为 N 沟道漏极开路输出 (V_{DD} 耐压)。 可以以 1 位为单位指定为输入 / 输出。 输入端口时, 可通过软件设定使用内部上拉电阻。	数字输入无效 注 1	SEG43/SO00/TxD0/ TOOLTxD
P01				(TI05)/(TO05)/ (INTP5)/PCLBUZ1/ SEG44
P02				INTP7/PCLBUZ0/ SEG45
P03				RxD2/SEG46/ VCOUT0
P04				TxD2/SEG47/ VCOUT1
P05				SCL10/SEG48
P06				RxD1/SDA10/SEG49
P07				TxD1/(PCLBUZ0)/ SEG50
P10	输入 / 输出	端口 1。 8 位输入 / 输出端口。 P16 和 P17 的输入可以设定为 TTL 输入缓冲器。 P16 和 P17 的输出可以设定为 N 沟道漏极开路输出 (V_{DD} 耐压)。 可以以 1 位为单位指定为输入 / 输出。 输入端口时, 可通过软件设定使用内部上拉电阻。	数字输入无效 注 1、注 4	SEG35
P11				SEG36
P12				SEG37
P13				SEG38
P14			数字输入无效 注 1	TI04/TO04/SEG39
P15				TI07/TO07/SEG40
P16				SEG41/SCK00/ SCL00
P17				SEG42/SI00/RxD0/ TOOLRxD/SDA00
P20	输入 / 输出	端口 2。 8 位输入 / 输出端口。 P20 和 P21 可以设定为模拟输入 注 3。 P22 和 P23 可以设定为模拟输入 注 2。 可以以 1 位为单位指定为输入 / 输出。 输入端口时, P22 ~ P27 可通过软件设定使用内部上拉电阻。	模拟输入端口	ANI1/AV _{REFM}
P21				ANI0/AV _{REFP}
P22				ANI16/SEG29
P23				ANI17/SEG30
P24			数字输入无效 注 1、注 4	SEG31
P25				SEG32
P26				SEG33
P27				SEG34

注 1. 处于“数字输入无效”状态时, 禁止所有的数字输出、数字输入以及 LCD 输出。

2. 当将各引脚用作输入时, 必须通过端口模式控制寄存器 x (PMCx) 设定数字或者模拟 (能以 1 位为单位进行设定)。

3. 通过 A/D 端口配置寄存器 (ADPC) 将各引脚设定为数字或者模拟。

4. 当用作数字输入 / 输出或者 LCD 输出时, 必须将端口模式控制寄存器 x (PMCx) 设定为“0”。

备注 可通过设定外围 I/O 重定向寄存器 (PIOR), 分配上表 () 内的功能。详细内容请参照“图 4-42 外围 I/O 重定向寄存器 (PIOR) 的格式”。

(2/3)

功能名称	输入 / 输出	功能	复位后	复用功能
P30	输入 / 输出	端口 3。 6 位输入 / 输出端口。 P34 的输入可以设定为 TTL 输入缓冲器。 P35 的输出可以设定为 N 沟道漏极开路输出 (V_{DD} 耐压)。 可以以 1 位为单位指定为输入 / 输出。 输入端口时, 可通过软件设定使用内部上拉电阻。	数字输入无效 注 1	TI03/TO03/SEG20
P31				INTP3/RTC1HZ/SEG21
P32				TI01/TO01/SEG22
P33				INTP4/SEG23
P34				RxD3/SEG24
P35				TxD3/SEG25
P40	输入 / 输出	端口 4。 8 位输入 / 输出端口。 P43、P44、P46 和 P47 的输入可以设定为 TTL 输入缓冲器。 P42 ~ P44、P46 和 P47 的输出可以设定为 N 沟道漏极开路输出 (V_{DD} 耐压)。 P42 ~ P45 可以设定为模拟输入 注 2。 可以以 1 位为单位指定为输入 / 输出。 输入端口时, 可通过软件设定使用内部上拉电阻。	输入端口	TOOL0/(TI00)/(TO00)
P41				(TI07)/(TO07)
P42				TI05/TO05/(TxD1)/IVREF1
P43				(INTP7)/(RxD1)/(SDA10)/IVCMP1
P44				(SCL10)/IVCMP0
P45			数字输入无效 注 1	IVREF0
P46				(SCK00)/(SCL00)/SEG26
P47				(SI00)/(RxD0)/(SDA00)/SEG27
P50	输入 / 输出	端口 5。 8 位输入 / 输出端口。 P53 和 P55 的输入可以设定为 TTL 输入缓冲器。 P53 ~ P56 的输出可以设定为 N 沟道漏极开路输出 (V_{DD} 耐压)。 可以以 1 位为单位指定为输入 / 输出。 输入端口时, 可通过软件设定使用内部上拉电阻。	数字输入无效 注 1	SEG4
P51				SEG5
P52				TI00/TO00/INTP1/SEG6
P53				INTP2/SCIO1/SEG7
P54				TI02/TO02/SCCLK1/SEG8
P55				INTP5/SCIO0/SEG9
P56				TI06/TO06/SCCLK0/SEG10
P57				INTP6/SEG11
P60	输入 / 输出	端口 6。 2 位输入 / 输出端口。 可以设定为 N 沟道漏极开路输出 (6 V 耐压)。 可以以 1 位为单位指定为输入 / 输出。	输入端口	SCLA0/(TI01)/(TO01)
P61				SDAA0/(TI02)/(TO02)

注 1. 处于“数字输入无效”状态时, 禁止所有的数字输出、数字输入以及 LCD 输出。

2. 当将各引脚用作输入时, 必须通过端口模式控制寄存器 x (PMCx) 设定数字或者模拟 (能以 1 位为单位进行设定)。

备注 可通过设定外围 I/O 重定向寄存器 (PIOR), 分配上表 () 内的功能。详细内容请参照“图 4-42 外围 I/O 重定向寄存器 (PIOR) 的格式”。

(3/3)

功能名称	输入 / 输出	功能	复位后	复用功能
P70	输入 / 输出	端口 7。 8 位输入 / 输出端口。 可以以 1 位为单位指定为输入 / 输出。 输入端口时，可通过软件设定使用内部上拉电阻。	数字输入无效 注	SEG12
P71				SEG13
P72				SEG14
P73				SEG15
P74				SEG16
P75				SEG17
P76				SEG18
P77				SEG19
P121	输入	端口 12。 3 位输入 / 输出端口和 4 位输入专用端口。 只有 P125 ~ P127，可以以 1 位为单位指定为输入 / 输出。 输入端口时，只有 P125 ~ P127 可通过软件设定使用内部上拉电阻。	输入端口	X1
P122				X2/EXCLK
P123				XT1
P124				XT2/EXCLKS
P125	输入 / 输出		数字输入无效 注	V _{L3} /(TI06)/(TO06)
P126				CAPL/(TI04)/(TO04)
P127				CAPH/(TI03)/(TO03)
P130	输入 / 输出	端口 13。 1 位输入 / 输出端口和 1 位输入专用端口。 P130 的输出可以设定为 N 沟道漏极开路输出（V _{DD} 耐压）。 只有 P130，可以以 1 位为单位指定为输入 / 输出。 输入端口时，只有 P130 可通过软件设定使用内部上拉电阻。	数字输入无效 注	(SO00)/(TxD0)/ SEG28
P137	输入			输入端口

注 处于“数字输入无效”状态时，禁止所有的数字输出、数字输入以及 LCD 输出。

备注 可通过设定外围 I/O 重定向寄存器 (PIOR)，分配上表 () 内的功能。详细内容请参照“图 4-42 外围 I/O 重定向寄存器 (PIOR) 的格式”。

2.2 端口以外的引脚功能

(1/2)

功能名称	输入 / 输出	功能
ANI0、ANI1、ANI16、ANI17	输入	A/D 转换器的模拟输入（参照“图 12-38 模拟输入引脚的连接”）
INTP0 ~ INTP7	输入	外部中断请求输入 指定有效沿：上升沿、下降沿或上升下降的双边沿
IVCMP0、IVCMP1	输入	比较器的模拟电压输入
IVREF0、IVREF1	输入	比较器的基准电压输入
VCOUT0、VCOUT1	输出	比较器输出
PCLBUZ0、PCLBUZ1	输出	时钟输出 / 蜂鸣器输出
REGC	—	连接用于内部运行稳压器的输出稳定电容器。 通过电容器（0.47 ~ 1μF）连接至 V _{SS} 。 由于用于稳定内部电压，因此，必须使用特性良好的电容器。
RTC1HZ	输出	实时时钟校正时钟（1Hz）输出
RESET	输入	低电平有效的系统复位输入引脚。 未使用外部复位引脚时，直接或者通过电阻连接至 V _{DD} 。
RxD0 ~ RxD3	输入	串行接口 UART0 ~ UART3 的串行数据输入
TxD0 ~ TxD3	输出	串行接口 UART0 ~ UART3 的串行数据输出
SCK00	输入 / 输出	串行接口 CSI00 的串行时钟输入 / 输出
SI00	输入	串行接口 CSI00 的串行数据输入
SO00	输出	串行接口 CSI00 的串行数据输出
SCL00、SCL10	输出	串行接口 IIC00 和 IIC10 的串行时钟输出
SDA00、SDA10	输入 / 输出	串行接口 IIC00 和 IIC10 的串行数据输入 / 输出
SCLA0	输入 / 输出	串行接口 IICA0 的串行时钟输入 / 输出
SDAA0	输入 / 输出	串行接口 IICA0 的串行数据输入 / 输出
SCCLK0、SCCLK1	输出	智能卡接口 SMCI0 和 SMCI1 的串行时钟输出
SCIO0、SCIO1	输入 / 输出	智能卡接口 SMCI0 和 SMCI1 的串行数据输入 / 输出
TI00 ~ TI07	输入	向 16 位定时器 00 ~ 07 输入外部计数时钟 / 捕捉触发
TO00 ~ TO07	输出	16 位定时器 00 ~ 07 的定时器输出
V _{L1} ~ V _{L4}	—	LCD 驱动电压
CAPH、CAPL	—	连接至用于 LCD 控制器 / 驱动器的电容器
X1、X2	—	连接用于主系统时钟的谐振器
EXCLK	输入	输入用作主系统时钟的外部时钟
XT1、XT2	—	连接用于副系统时钟的谐振器
EXCLKS	输入	输入副系统时钟的外部时钟

(2/2)

功能名称	输入 / 输出	功能
V _{DD}	—	所有引脚的正电源
AV _{REFP}	输入	A/D 转换器的 (+) 基准电压输入
AV _{REFM}	输入	A/D 转换器的 (-) 基准电压输入
V _{SS}	—	所有引脚的接地电位
TOOLRxD	输入	用于闪存编程时连接外部器件的 UART 串行数据接收
TOOLTxD	输出	用于闪存编程时连接外部器件的 UART 串行数据发送
TOOL0	输入 / 输出	闪存编程器 / 调试器的数据输入 / 输出
COM0 ~ COM7	输出	LCD 控制器 / 驱动器的公共信号输出
SEG0 ~ SEG50	输出	LCD 控制器 / 驱动器的段信号输出

注意 解除复位后的 P40/TOOL0 和运行模式之间的关系如下所示。

表 2-1 解除复位后的 P40/TOOL0 和运行模式之间的关系

P40/TOOL0	运行模式
V _{DD}	通常运行模式
0V	闪存编程模式

详细内容请参照“28.4 串行编程方法”。

备注 作为防止噪声和门锁的对策，必须在 V_{DD} 与 V_{SS} 线路之间通过最短并且较粗的布线连接旁路电容器（0.1μF 左右）。

2.3 引脚的输入 / 输出电路和未使用引脚的处理

表 2-2 表示的是引脚的输入 / 输出电路的类型和未使用引脚的推荐连接。

表 2-2 各未使用引脚的处理 (1/3)

引脚名称	输入 / 输出电路类型	输入 / 输出	未使用时的推荐连接方法
P00/SEG43/SO00/TxD0/ TOOLTxD	45-A	输入 / 输出	< 设定端口的情况 > 输入：通过电阻单独连接至 V _{DD} 或者 V _{SS} 。 输出：必须置为开路。 < 设定段的情况 > 必须置为开路。
P01/(TI05)/(TO05)/(INTP5)/ PCLBUZ1/SEG44			
P02/INTP7/PCLBUZ0/SEG45			
P03/RxD2/SEG46/VCOUT0	45-B		
P04/TxD2/SEG47/VCOUT1	45-A		
P05/SCL10/SEG48	45-B		
P06/RxD1/SDA10/SEG49			
P07/TxD1/(PCLBUZ0)/SEG50			
P10/SEG35			
P11/SEG36			
P12/SEG37			
P13/SEG38			
P14/TI04/TO04/SEG39			
P15/TI07/TO07/SEG40			
P16/SEG41/SCK00/SCL00	45-B		< 设定端口的情况 > 输入：通过电阻单独连接至 V _{DD} 或者 V _{SS} 。 输出：必须置为开路。 < 设定段的情况 > 必须置为开路。
P17/SEG42/SI00/RxD0/ TOOLRxD/SDA00			
P20/ANI1/AV _{REFM}	11-T		输入：通过电阻单独连接至 V _{DD} 或者 V _{SS} 。 输出：必须置为开路。
P21/ANI0/AV _{REFP}			
P22/ANI16/SEG29	45-D		< 设定端口的情况 > 输入：通过电阻单独连接至 V _{DD} 或者 V _{SS} 。 输出：必须置为开路。 < 设定段的情况 > 必须置为开路。
P23/ANI17/SEG30			
P24/SEG31	45-A		
P25/SEG32			
P26/SEG33			
P27/SEG34			

备注 能通过设定外围 I/O 重定向寄存器 (PIOR)，分配上表 () 内的功能。

表 2-2 未使用引脚的连接 (2/3)

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
P30/TI03/TO03/SEG20	45-A	输入 / 输出	< 设定端口的情况 > 输入：通过电阻单独连接至 V _{DD} 或者 V _{SS} 。 输出：必须置为开路。 < 设定段的情况 > 必须置为开路。
P31/INTP3/RTC1HZ/SEG21			
P32/TI01/TO01/SEG22			
P33/INTP4/SEG23			
P34/RxD3/SEG24	45-B		
P35/TxD3/SEG25	45-A		
P40/TOOL0/(TI00)/(TO00)	8-R		
P41/(TI07)/(TO07)			
P42/TI05/TO05/(TxD1)/IVREF1	5-BB		< 设定端口的情况 > 输入：通过电阻单独连接至 V _{DD} 或者 V _{SS} 。 输出：必须置为开路。
P43/(INTP7)/(RxD1)/(SDA10)/IVCMP1	5-BC		
P44/(SCL10)/IVCMP0			
P45/IVREF0	5-BB		
P46/(SCK00)/(SCL00)/SEG26	45-B		< 设定端口的情况 > 输入：通过电阻单独连接至 V _{DD} 或者 V _{SS} 。 输出：必须置为开路。 < 设定段的情况 > 必须置为开路。
P47/(SI00)/(RxD0)/(SDA00)/SEG27			
P50/SEG4	45-A		< 设定端口的情况 > 输入：通过电阻单独连接至 V _{DD} 或者 V _{SS} 。 输出：必须置为开路。 < 设定段的情况 > 必须置为开路。
P51/SEG5			
P52/TI00/TO00/INTP1/SEG6			
P53/INTP2/SCIO1/SEG7	45-B		
P54/TI02/TO02/SCCLK1/SEG8	45-A		
P55/INTP5/SCIO0/SEG9	45-B		
P56/TI06/TO06/SCCLK0/SEG10	45-A		
P57/INTP6/SEG11			

备注 能通过设定外围 I/O 重定向寄存器 (PIOR)，分配上表 () 内的功能。

表 2-2 未使用引脚的连接 (3/3)

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
P60/SCLA0/(TI01)/(TO01)	13-R	输入 / 输出	输入：通过电阻单独连接至 V _{DD} 或者 V _{SS} 。 输出：将端口的输出锁存器置“0”并且置为开路， 或者将端口的输出锁存器置“1”并且通过电阻单独连接至 V _{DD} 或者 V _{SS} 。
P61/SDAA0/(TI02)/(TO02)			
P70/SEG12	45-A		< 设定端口的情况 > 输入：通过电阻单独连接至 V _{DD} 或者 V _{SS} 。 输出：必须置为开路。 < 当设定为段输出 > 必须置为开路。
P71/SEG13			
P72/SEG14			
P73/SEG15			
P74/SEG16			
P75/SEG17			
P76/SEG18			
P77/SEG19			
P121/X1	37-C	输入	通过电阻单独连接至 V _{DD} 或者 V _{SS} 。
P122/X2/EXCLK			
P123/XT1			
P124/XT2/EXCLKS			
P125/V _{L3} /(TI06)/(TO06)	5-AO	输入 / 输出	输入：通过电阻单独连接至 V _{DD} 或者 V _{SS} 。 输出：必须置为开路。
P126/CAPL/(TI04)/(TO04)	12-I		
P127/CAPH/(TI03)/(TO03)			
P130/(SO00)/(TxD0)/SEG28	45-A		< 设定端口的情况 > 输入：通过电阻单独连接至 V _{DD} 或者 V _{SS} 。 输出：必须置为开路。 < 设定段的情况 > 必须置为开路。
P137/INTP0	2	输入	通过电阻单独连接至 V _{DD} 或者 V _{SS} 。
RESET	2	输入	直接或者通过电阻连接至 V _{DD} 。
REGC	—	—	通过电容器（0.47 ～ 1μF）连接至 V _{SS} 。
COM0 ～ COM3	45	输出	必须置为开路。
COM4/SEG0			
COM5/SEG1			
COM6/SEG2			
COM7/SEG3			
V _{L1}	—	—	
V _{L2}	—	—	
V _{L4}	—	—	

备注 能通过设定外围 I/O 重定向寄存器 (PIOR)，分配上表 () 内的功能。

图 2-1 各引脚的输入 / 输出电路列表 (1/4)

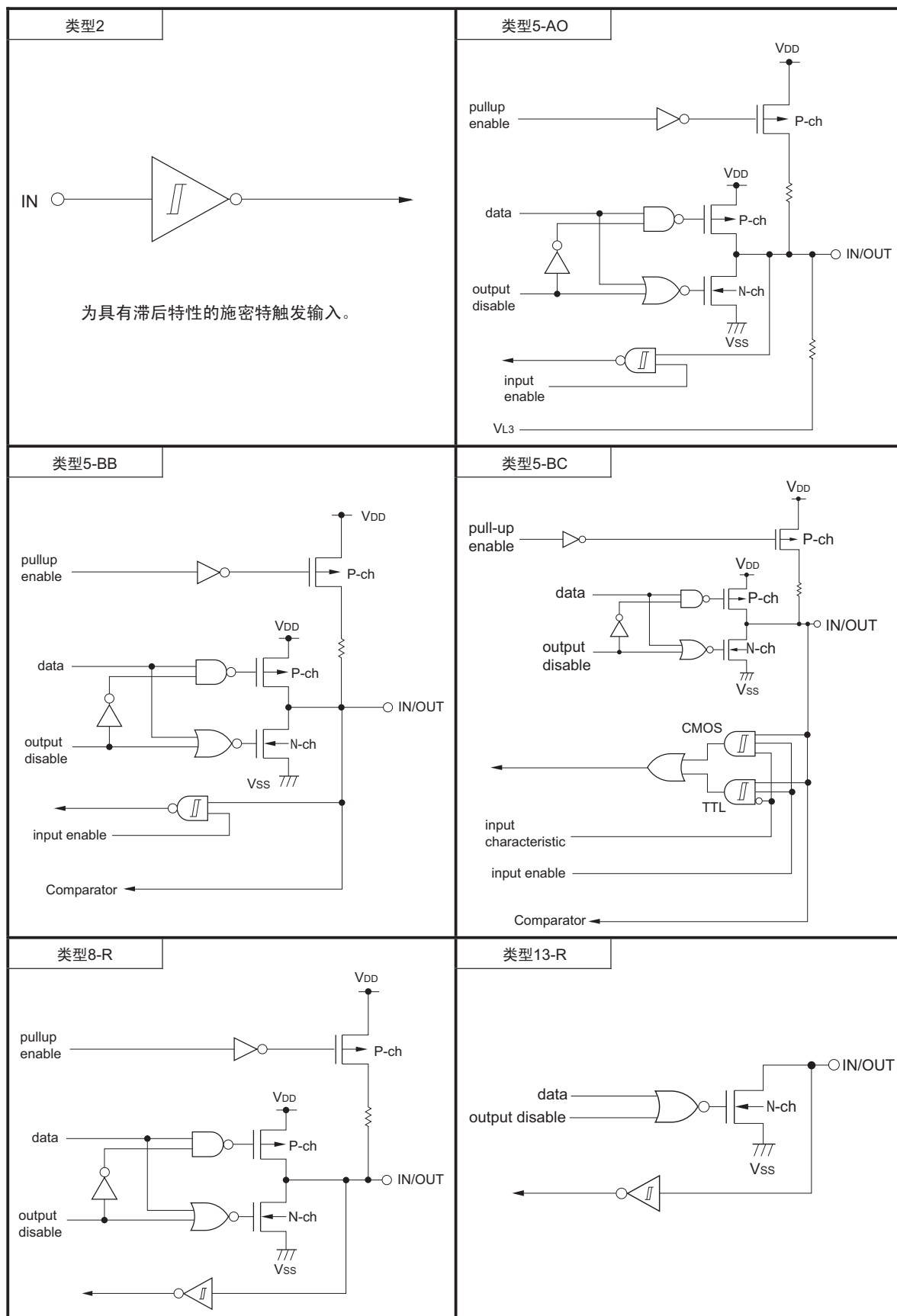


图 2-1 各引脚的输入 / 输出电路列表 (2/4)

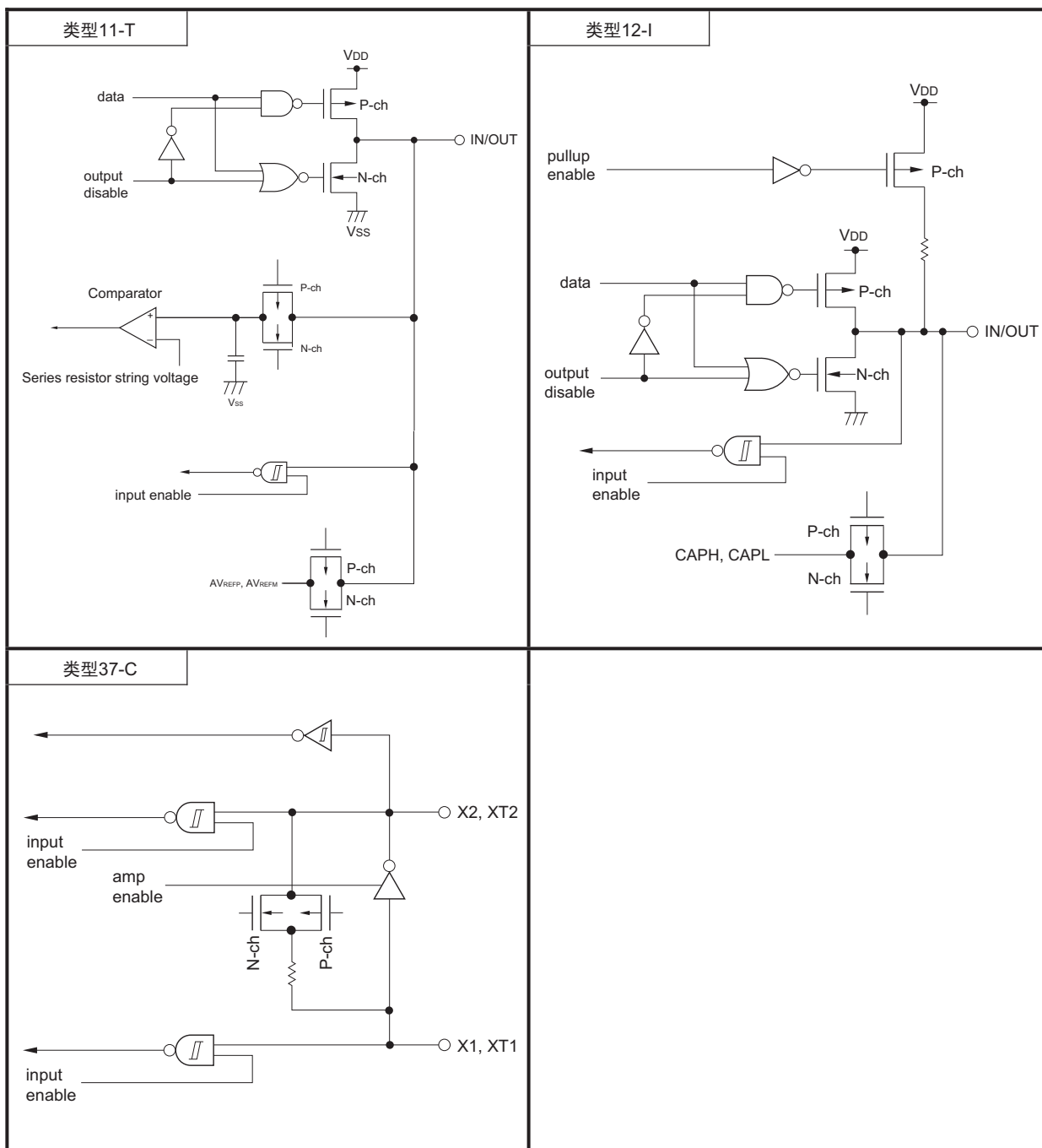


图 2-1 各引脚的输入 / 输出电路列表 (3/4)

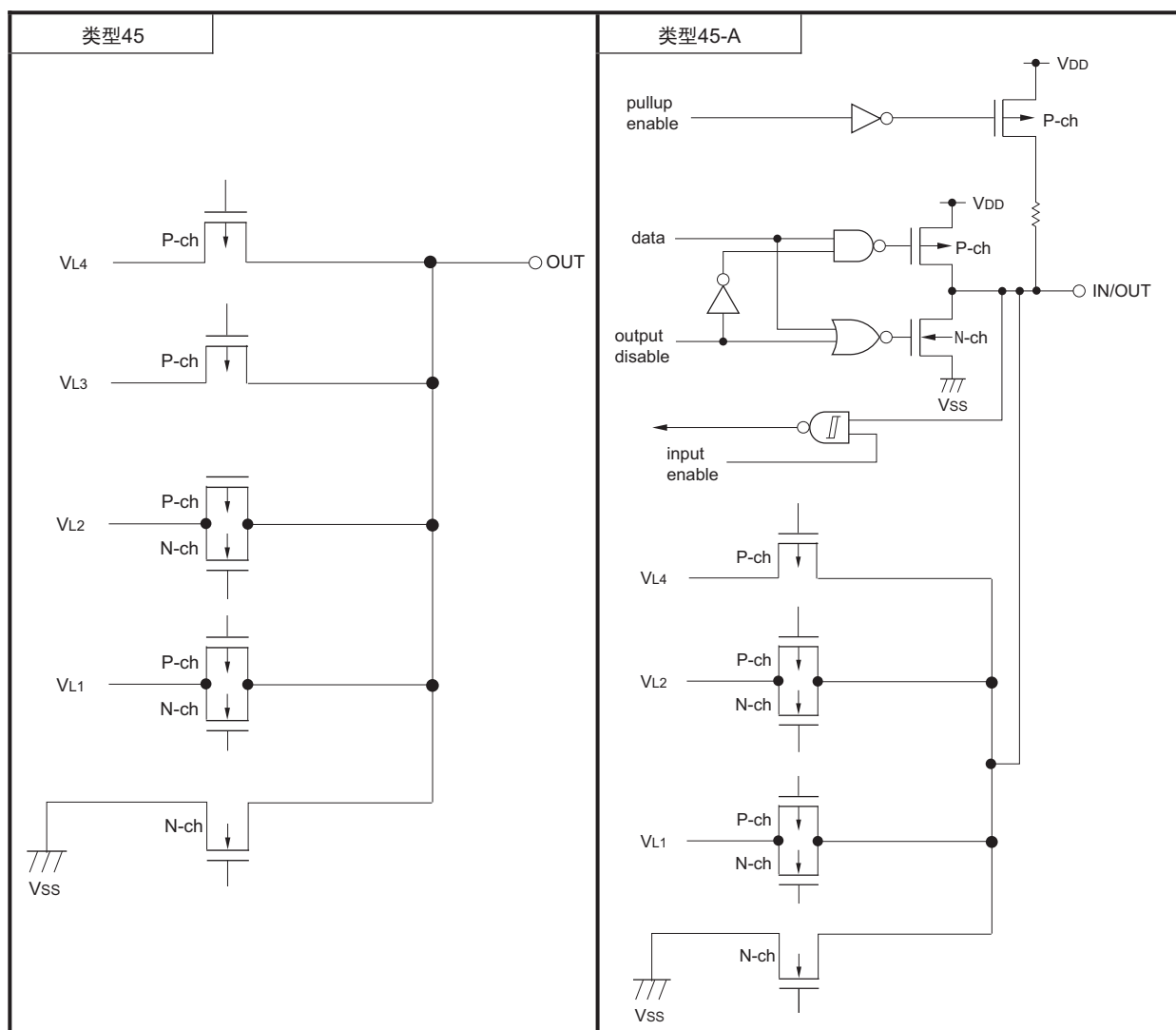
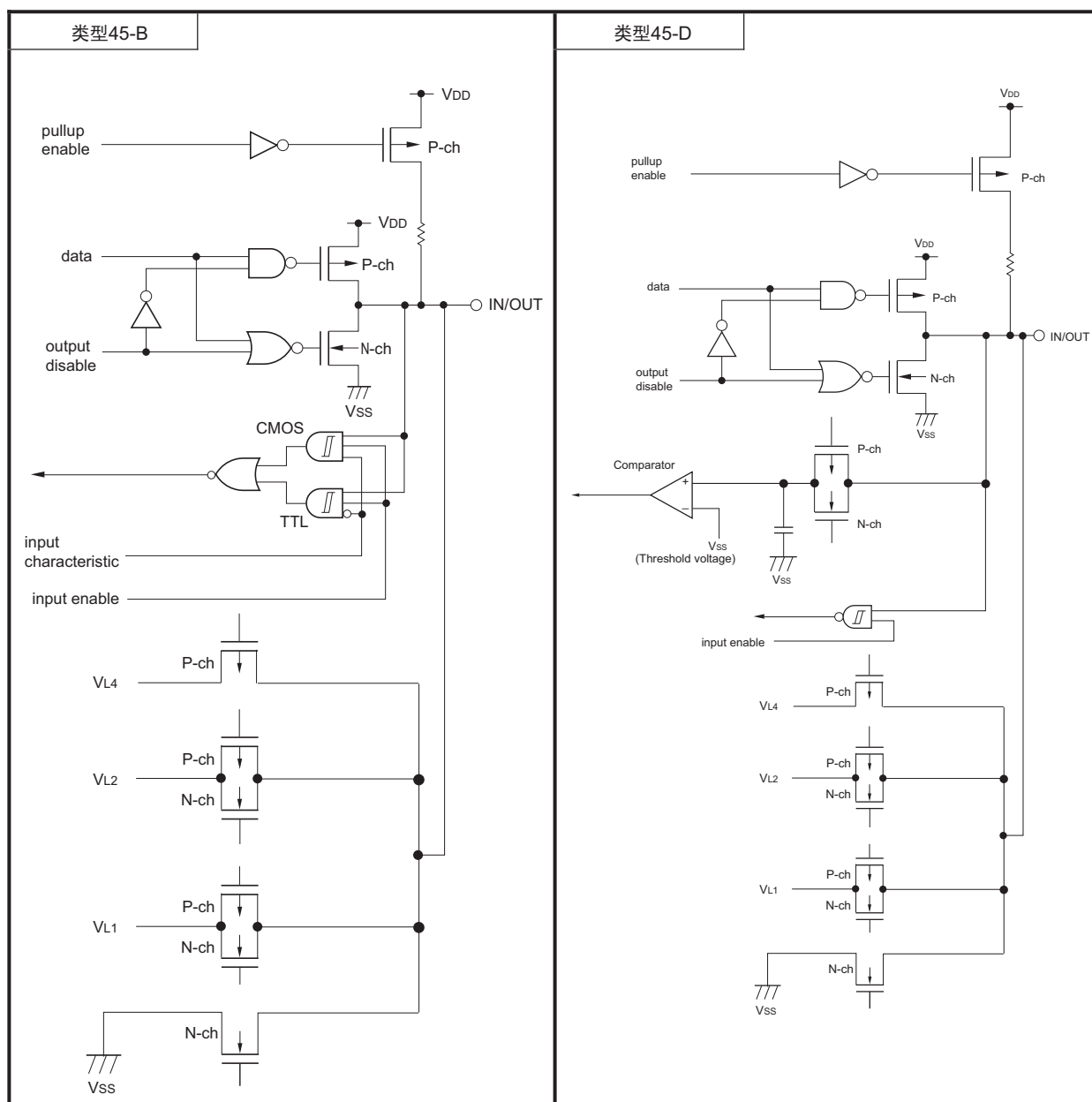


图 2-1 各引脚的输入 / 输出电路列表 (4/4)

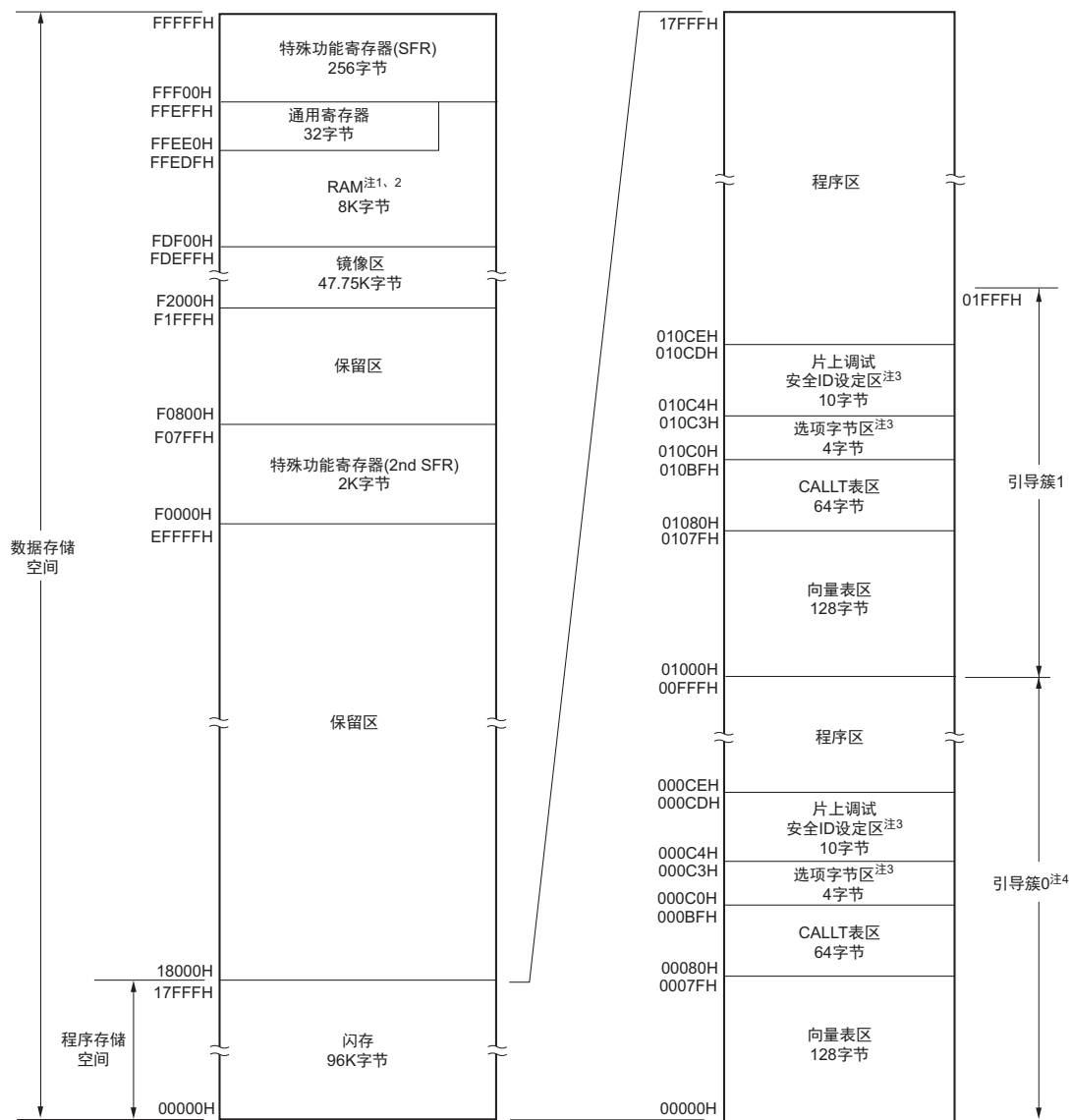


第 3 章 CPU 体系结构

3.1 存储空间

R7F0C003 和 R7F0C004 可以存取 1 M 字节的存储空间。存储器映射图如图 3-1 和图 3-2 所示。

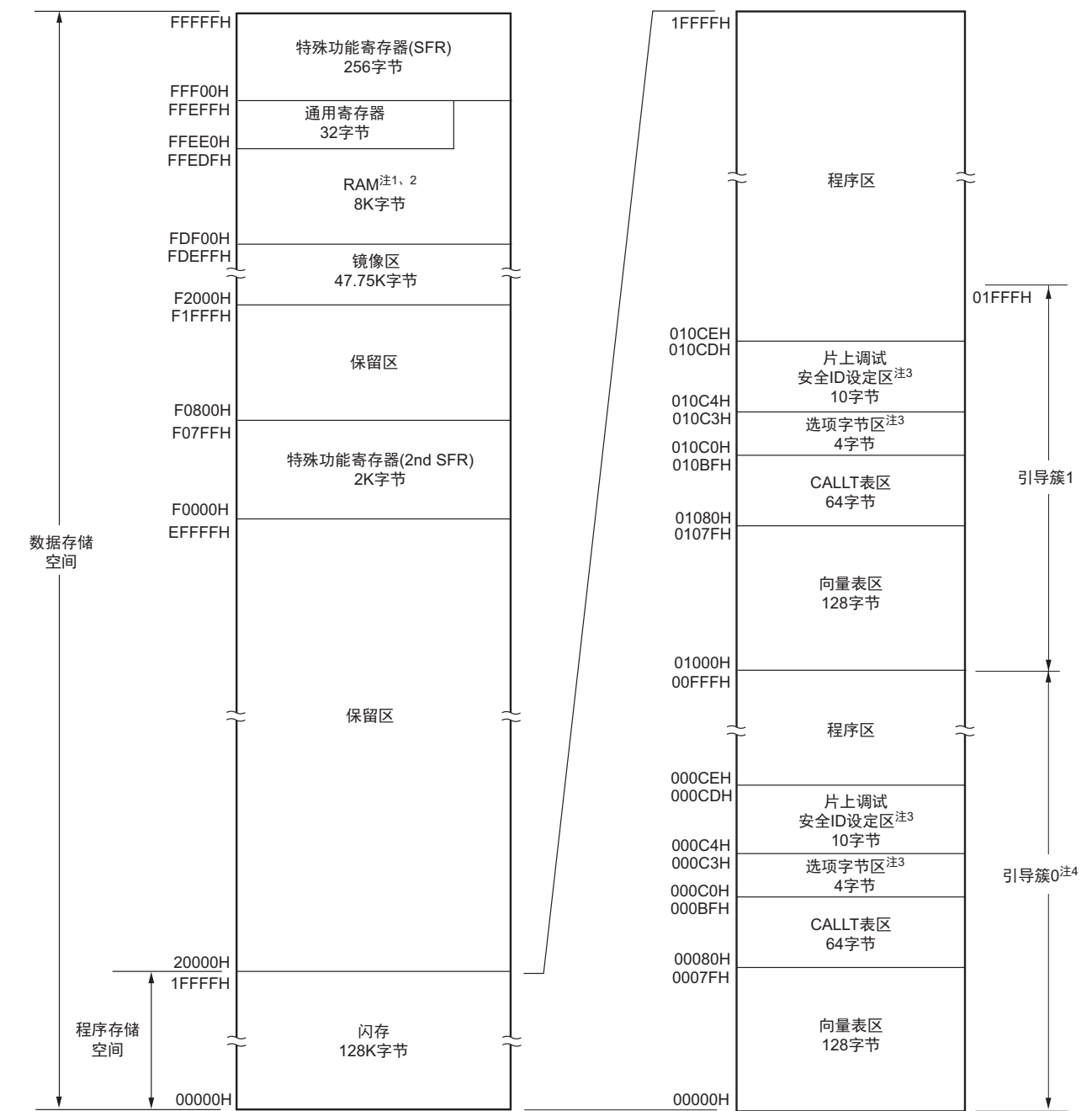
图 3-1 存储器映射图（R7F0C003）



- 注
1. 在自编程时，不能将堆栈、数据缓冲器、向量中断处理的转移目标以及 DMA 的转移目标 / 传送源使用的 RAM 地址分配到 FFE20H ~ FFEDFH 区。另外，FDF00H ~ FE309H 区用于程序库，所以禁止使用。
 2. 可以从通用寄存器以外的 RAM 区执行指令。
 3. 不使用引导交换功能时：给 000C0H ~ 000C3H 设定选项字节，并且给 000C4H ~ 000CDH 设定片上调试安全 ID。
使用引导交换功能时：给 000C0H ~ 000C3H 和 010C0H ~ 010C3H 设定选项字节，并且给 000C4H ~ 000CDH 和 010C4H ~ 010CDH 设定片上调试安全 ID。
 4. 可以通过安全设定禁止改写引导簇 0（参照“28.6 安全功能”）。

注意 在允许产生 RAM 奇偶校验错误复位（RPERDIS=0）的情况下，当存取数据时，必须对“所用 RAM 区”进行初始化；当从 RAM 区执行指令时，必须对“所用 RAM 区 +10 字节”的区域进行初始化。通过产生复位，进入允许产生 RAM 奇偶校验错误复位（RPERDIS=0）的状态。详细内容请参照“25.3.3 RAM 奇偶校验错误检测功能”。

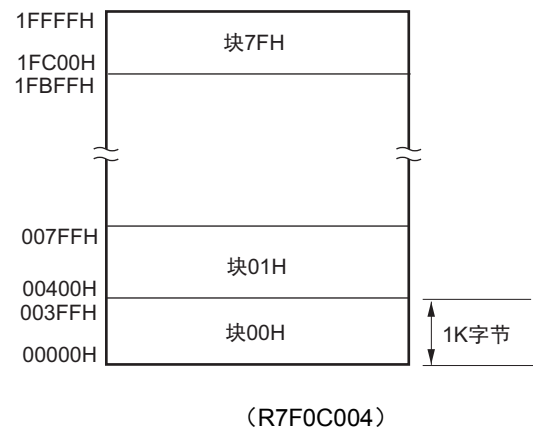
图 3-2 存储器映射图（R7F0C004）



- 注
- 1. 在自编程时，不能将堆栈、数据缓冲器、向量中断处理的转移目标以及 DMA 的转移目标 / 传送源使用的 RAM 地址分配到 FFE20H ~ FFEDFH 区。另外，FDF00H ~ FE309H 区用于程序库，所以禁止使用。
 - 2. 可以从通用寄存器以外的 RAM 区执行指令。
 - 3. 不使用引导交换功能时：给 000C0H ~ 000C3H 设定选项字节，并且给 000C4H ~ 000CDH 设定片上调试安全 ID。
使用引导交换功能时：给 000C0H ~ 000C3H 和 010C0H ~ 010C3H 设定选项字节，并且给 000C4H ~ 000CDH 和 010C4H ~ 010CDH 设定片上调试安全 ID。
 - 4. 可以通过安全设定禁止改写引导簇 0（参照“28.6 安全功能”）。

注意 在允许产生 RAM 奇偶校验错误复位（RPERDIS=0）的情况下，当存取数据时，必须对“所用 RAM 区”进行初始化；当从 RAM 区执行指令时，必须对“所用 RAM 区 +10 字节”的区域进行初始化。通过产生复位，进入允许产生 RAM 奇偶校验错误复位（RPERDIS=0）的状态。详细内容请参照“25.3.3 RAM 奇偶校验错误检测功能”。

备注 闪存分为多个块（1 块 =1K 字节）。有关地址值和块号，请参照“表 3-1 闪存的地址值和块号的对应”。



闪存地址值和块号的对应如下所示。

表 3-1 闪存地址值和块号的对应

地址值	块号	地址值	块号	地址值	块号	地址值	块号
00000H ~ 003FFH	00H	08000H ~ 083FFH	20H	10000H ~ 103FFH	40H	18000H ~ 183FFH	60H
00400H ~ 007FFH	01H	08400H ~ 087FFH	21H	10400H ~ 107FFH	41H	18400H ~ 187FFH	61H
00800H ~ 00BFFH	02H	08800H ~ 08BFFH	22H	10800H ~ 10BFFH	42H	18800H ~ 18BFFH	62H
00C00H ~ 00FFFH	03H	08C00H ~ 08FFFH	23H	10C00H ~ 10FFFH	43H	18C00H ~ 18FFFH	63H
01000H ~ 013FFH	04H	09000H ~ 093FFH	24H	11000H ~ 113FFH	44H	19000H ~ 193FFH	64H
01400H ~ 017FFH	05H	09400H ~ 097FFH	25H	11400H ~ 117FFH	45H	19400H ~ 197FFH	65H
01800H ~ 01BFFH	06H	09800H ~ 09BFFH	26H	11800H ~ 11BFFH	46H	19800H ~ 19BFFH	66H
01C00H ~ 01FFFH	07H	09C00H ~ 09FFFH	27H	11C00H ~ 11FFFH	47H	19C00H ~ 19FFFH	67H
02000H ~ 023FFH	08H	0A000H ~ 0A3FFH	28H	12000H ~ 123FFH	48H	1A000H ~ 1A3FFH	68H
02400H ~ 027FFH	09H	0A400H ~ 0A7FFH	29H	12400H ~ 127FFH	49H	1A400H ~ 1A7FFH	69H
02800H ~ 02BFFH	0AH	0A800H ~ 0ABFFH	2AH	12800H ~ 12BFFH	4AH	1A800H ~ 1ABFFH	6AH
02C00H ~ 02FFFH	0BH	0AC00H ~ 0AFFFH	2BH	12C00H ~ 12FFFH	4BH	1AC00H ~ 1AFFFH	6BH
03000H ~ 033FFH	0CH	0B000H ~ 0B3FFH	2CH	13000H ~ 133FFH	4CH	1B000H ~ 1B3FFH	6CH
03400H ~ 037FFH	0DH	0B400H ~ 0B7FFH	2DH	13400H ~ 137FFH	4DH	1B400H ~ 1B7FFH	6DH
03800H ~ 03BFFH	0EH	0B800H ~ 0BBFFH	2EH	13800H ~ 13BFFH	4EH	1B800H ~ 1BBFFH	6EH
03C00H ~ 03FFFH	0FH	0BC00H ~ 0BFFFH	2FH	13C00H ~ 13FFFH	4FH	1BC00H ~ 1BFFFH	6FH
04000H ~ 043FFH	10H	0C000H ~ 0C3FFH	30H	14000H ~ 143FFH	50H	1C000H ~ 1C3FFH	70H
04400H ~ 047FFH	11H	0C400H ~ 0C7FFH	31H	14400H ~ 147FFH	51H	1C400H ~ 1C7FFH	71H
04800H ~ 04BFFH	12H	0C800H ~ 0CBFFH	32H	14800H ~ 14BFFH	52H	1C800H ~ 1CBFFH	72H
04C00H ~ 04FFFH	13H	0CC00H ~ 0CFFFH	33H	14C00H ~ 14FFFH	53H	1CC00H ~ 1CFFFH	73H
05000H ~ 053FFH	14H	0D000H ~ 0D3FFH	34H	15000H ~ 153FFH	54H	1D000H ~ 1D3FFH	74H
05400H ~ 057FFH	15H	0D400H ~ 0D7FFH	35H	15400H ~ 157FFH	55H	1D400H ~ 1D7FFH	75H
05800H ~ 05BFFH	16H	0D800H ~ 0DBFFH	36H	15800H ~ 15BFFH	56H	1D800H ~ 1DBFFH	76H
05C00H ~ 05FFFH	17H	0DC00H ~ 0DFFFH	37H	15C00H ~ 15FFFH	57H	1DC00H ~ 1DFFFH	77H
06000H ~ 063FFH	18H	0E000H ~ 0E3FFH	38H	16000H ~ 163FFH	58H	1E000H ~ 1E3FFH	78H
06400H ~ 067FFH	19H	0E400H ~ 0E7FFH	39H	16400H ~ 167FFH	59H	1E400H ~ 1E7FFH	79H
06800H ~ 06BFFH	1AH	0E800H ~ 0EBFFH	3AH	16800H ~ 16BFFH	5AH	1E800H ~ 1EBFFH	7AH
06C00H ~ 06FFFH	1BH	0EC00H ~ 0EFFFH	3BH	16C00H ~ 16FFFH	5BH	1EC00H ~ 1EFFFH	7BH
07000H ~ 073FFH	1CH	0F000H ~ 0F3FFH	3CH	17000H ~ 173FFH	5CH	1F000H ~ 1F3FFH	7CH
07400H ~ 077FFH	1DH	0F400H ~ 0F7FFH	3DH	17400H ~ 177FFH	5DH	1F400H ~ 1F7FFH	7DH
07800H ~ 07BFFH	1EH	0F800H ~ 0FBFFH	3EH	17800H ~ 17BFFH	5EH	1F800H ~ 1FBFFH	7EH
07C00H ~ 07FFFH	1FH	0FC00H ~ 0FFFFH	3FH	17C00H ~ 17FFFH	5FH	1FC00H ~ 1FFFFH	7FH

备注 R7F0C003: 块号 00H ~ 5FH

R7F0C004: 块号 00H ~ 7FH

3.1.1 内部程序存储器空间

内部程序存储器空间保存程序和数据表。

R7F0C003 和 R7F0C004 内置的内部 ROM（闪存）如下所示。

表 3-2 ROM 容量

产品	内部 ROM	
	结构	容量
R7F0C003M2DFB	闪存	98304 × 8 位（00000H ~ 17FFFH）
R7F0C004M2DFB		131072 × 8 位（00000H ~ 1FFFFH）

内部程序存储器空间分为以下区域。

(1) 向量表区域

将 00000H ~ 0007FH 的 128 字节区域保留为向量表区，向量表区保存在复位或者产生各中断请求时需要转移的程序起始地址。另外，因为向量码为 2 字节，所以中断的转移目标地址为 00000H ~ 0FFFFH 的 64K 地址。

偶数地址保存 16 位地址中的低 8 位，奇数地址保存 16 位地址中的高 8 位。

在使用引导交换功能时，也必须给 01000H ~ 0107FH 设定向量表。

表 3-3 向量表 (1/2)

向量表地址	中断源
0000H	RESET、POR、LVD、WDT、TRAP、IAW、RPE
0004H	INTWDTI
0006H	INTLVI
0008H	INTP0
000AH	INTP1
000CH	INTP2
000EH	INTP3
0010H	INTP4
0012H	INTP5
0014H	INTST2
0016H	INTSR2
0018H	INTSRE2
001AH	INTDMA0
001CH	INTDMA1
001EH	INTST0/INTCSI00/INTIIC00
0020H	INTTM00
0022H	INTSR0
0024H	INTSRE0
	INTTM01H
0026H	INTST1/INTIIC10
0028H	INTSR1
002AH	INTSRE1
	INTTM03H
002CH	INTIICA0
002EH	INTRTIT
0030H	INTFM
0032H	INTTM01
0034H	INTTM02
0036H	INTTM03
0038H	INTAD
003AH	INTRTC
003CH	INTIT
0040H	INTST3
0042H	INTSR3

表 3-3 向量表 (2/2)

向量表地址	中断源
0046H	INTTM04
0048H	INTTM05
004AH	INTP6
004CH	INTP7
0050H	INTCMP0
0052H	INTCMP1
0054H	INTTM06
0056H	INTTM07
0058H	INTSCT0
005AH	INTSCR0
005CH	INTSRE3
005EH	INTMD
0060H	INTSCE0
0062H	INTFL
0064H	INTDMA2
0066H	INTDMA3
0068H	INTSCT1
006AH	INTSCR1
006CH	INTSCE1
007EH	BRK

(2) CALLT 指令表区

00080H ~ 000BFH 的 64 字节区域能保存 2 字节调用指令 (CALLT) 的子程序入口地址。必须给子程序入口地址设定 00000H ~ 0FFFFH 内的值 (因为地址码为 2 字节)。

在使用引导交换功能时, 也必须给 01080H ~ 010BFH 设定 CALLT 指令表。

(3) 选项字节区

000C0H ~ 000C3H 的 4 字节区域用作选项字节区。在使用引导交换功能时, 也必须给 010C0H ~ 010C3H 设定选项字节。详细内容请参照“第 27 章 选项字节”。

(4) 片上调试安全 ID 设定区

000C4H ~ 000CDH 和 010C4H ~ 010CDH 的 10 字节区域用作片上调试安全 ID 设定区。当不使用引导交换功能时, 必须给 000C4H ~ 000CDH 设定 10 字节的片上调试安全 ID; 当使用引导交换功能时, 必须给 000C4H ~ 000CDH 和 010C4H ~ 010CDH 设定 10 字节的片上调试安全 ID。详细内容请参照“第 29 章 片上调试功能”。

3.1.2 镜像区

R7F0C003 和 R7F0C004 将 00000H ~ 0FFFFH 或者 10000H ~ 1FFFFH 的闪存区镜像到 F0000H ~ FFFFFH (通过处理器模式控制寄存器 (PMC) 进行设定)。

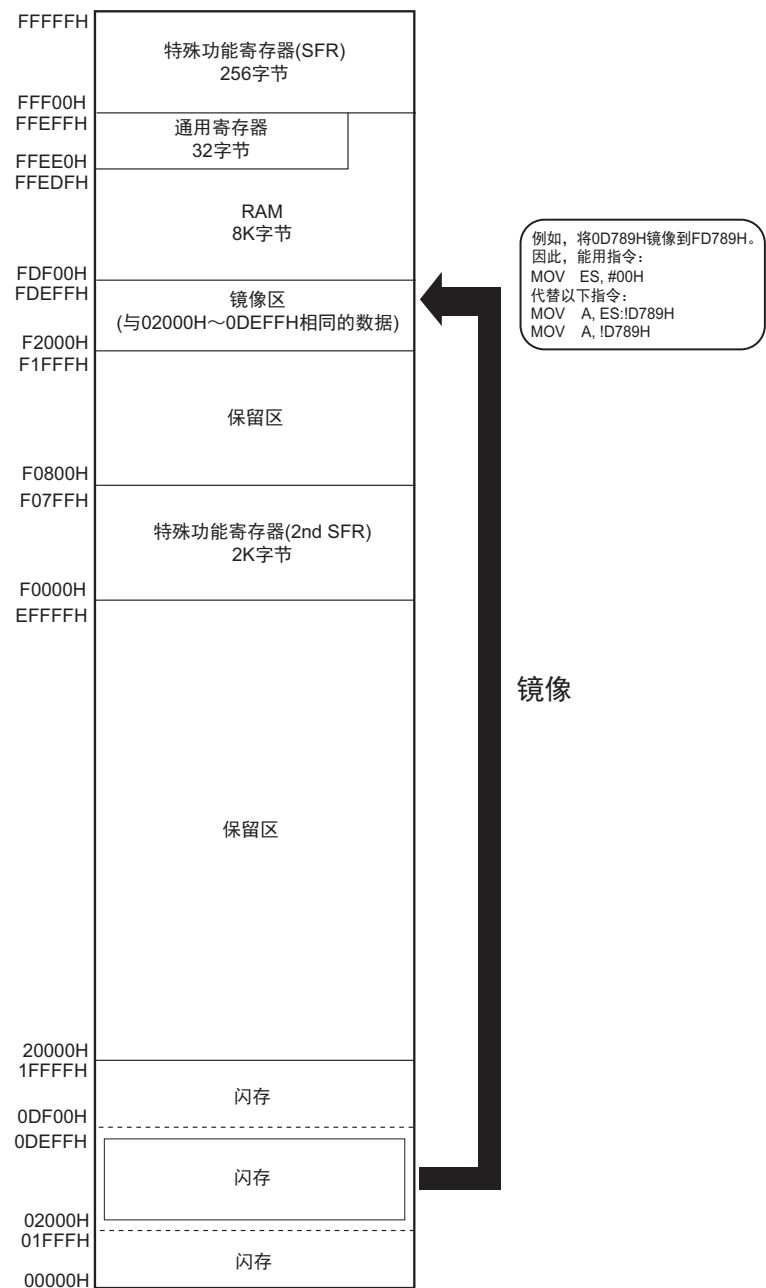
能通过从镜像目标的 F0000H ~ FFFFFH 读数据来使用操作数中不具有 ES 寄存器的指令，因此能用短代码读代码闪存的内容。但是，不能将代码闪存区镜像到 SFR、扩展 SFR、RAM 区以及保留区。

有关各产品的镜像区，请参照“3.1 存储空间”。

镜像区为只读区，不能从此区域取指令。

例子如下所示。

例子 R7F0C004 (闪存：128K 字节、RAM：8K 字节)



PMC 寄存器如下所示。

- 处理器模式控制寄存器（PMC）
这是设定要镜像到F0000H～FFFFFFH的闪存空间的寄存器。
通过1位或者8位存储器操作指令设定PMC寄存器。
在产生复位信号后，此寄存器的值变为“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	设定镜像到 F0000H ~ FFFFFFFH 的闪存空间
0	00000H ~ 0FFFFH 镜像到 F0000H ~ FFFFFFFH
1	10000H ~ 1FFFFH 镜像到 F0000H ~ FFFFFFFH

注意 设定 PMC 寄存器后，至少等待一条指令后才可以读取镜像区。

3.1.3 内部数据存储空间

R7F0C003 和 R7F0C004 内置以下 RAM。

表 3-4 内部 RAM 容量

产品	内部 RAM
R7F0C003、R7F0C004	8192 × 8 位（FDF00H ~ 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。给内部 RAM 区中的 FFEE0H ~ FFEFFH 的 32 字节区域分配了以 8 个 8 位寄存器为 1 组的 4 组通用寄存器。但是，不能通过通用寄存器执行指令。

堆栈存储器使用内部 RAM。

- 注意 1. 不能将分配通用寄存器（FFEE0H ~ FFEFFH）的空间用于取指令和堆栈区。
2. 在自编程和改写数据闪存时，不能将用于堆栈、数据缓冲器、向量中断处理的转移目标以及 DMA 的传送目标 / 传送源的 RAM 地址分配到 FFE20H ~ FFEDFH 的区域。
3. 在自编程和改写数据闪存时，以下产品的 RAM 区域用于程序库，所以禁止使用。
R7F0C003、R7F0C004：FDF00H ~ FE309H

3.1.4 特殊功能寄存器（SFR：Special Function Register）的区域

内部外围硬件的特殊功能寄存器（SFR）分配在 FFF00H ~ FFFFFH 的区域（参照“3.2.4 特殊功能寄存器（SFR：Special Function Register）”的表 3-5）。

注意 不能存取未分配 SFR 的地址。

3.1.5 扩展特殊功能寄存器（2nd SFR：2nd Special Function Register）的区域

内部外围硬件的扩展特殊功能寄存器（2nd SFR）分配在 F0000H ~ F07FFH 的区域（参照“3.2.5 扩展特殊功能寄存器（2nd SFR：2nd Special Function Register）”的表 3-6）。

在此区域中分配了 SFR 区（FFF00H ~ FFFFFH）以外的 SFR，但是扩展 SFR 区的存取指令比 SFR 区长 1 字节。

注意 不能存取未分配扩展 SFR 的地址。

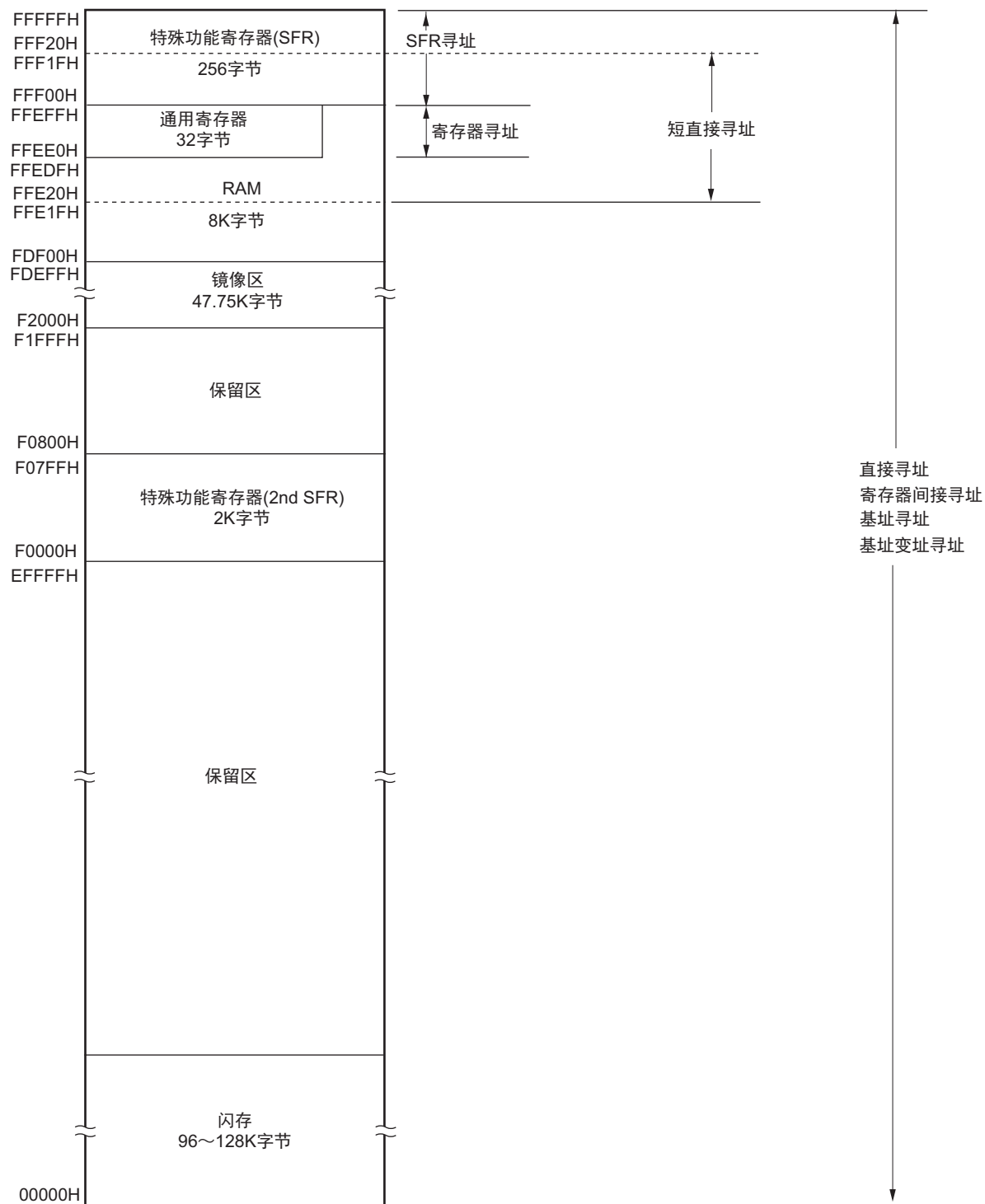
3.1.6 数据存储器的寻址

所谓寻址，是指定下一次要执行的指令地址以及指令执行操作对象的寄存器或者存储器等地址的方法。

对于指令执行操作对象的存储器的寻址，考虑到可操作性等，R7F0C003、R7F0C004 提供了丰富的寻址方式。尤其是能根据特殊功能寄存器（SFR）和通用寄存器等各种功能进行特殊的寻址。数据存储器和寻址的对应如图 3-4 所示。

有关各寻址的详细内容，请参照“3.4 处理数据地址的寻址”。

图 3-4 数据存储器和寻址的对应



3.2 处理器寄存器

R7F0C003 和 R7F0C004 内置以下处理器寄存器。

3.2.1 控制寄存器

这是具有控制程序顺序、状态和堆栈存储器等专用功能的寄存器。在控制寄存器中有程序计数器（PC）、程序状态字（PSW）和堆栈指针（SP）。

(1) 程序计数器（PC）

程序计数器是保存下一次要执行的程序地址信息的 20 位寄存器。

在通常运行时，根据预取的指令码字节数自动进行递增。在执行转移指令时，设定立即数或者寄存器的内容。

在产生复位信号后，给程序计数器设定地址 0000H 和 0001H 的复位向量表的值。

图 3-5 程序计数器的结构



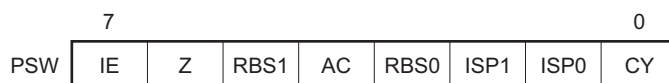
(2) 程序状态字（PSW）

程序状态字是由各种标志组成的 8 位寄存器，通过执行指令对这些标识进行置位和清除。

在接受向量中断请求以及执行 PUSH PSW 指令时，将程序状态字的内容保存到堆栈区，而在执行 RETB 指令、RETI 指令或者 POP PSW 指令时恢复程序状态字的内容。

在产生复位信号后，PSW 的值变为“06H”。

图 3-6 程序状态字的结构



(a) 中断允许标志（IE）

这是控制 CPU 的中断请求接受运行的标志。

当 IE 位是“0”时，为中断禁止（DI）状态，禁止全部可屏蔽中断请求。

当 IE 位是“1”时，为中断允许（EI）状态，通过优先级控制标志（ISP1、ISP0）、各中断源的中断屏蔽标志和优先级指定标志进行可屏蔽中断请求的接受控制。

通过执行 DI 指令或者接受中断，将此标志清“0”；通过执行 EI 指令，将此标志置“1”。

(b) 零标志（Z）

当运算或者比较结果为零或者相等时，将此标志置“1”。否则，将此标志清“0”。

(c) 寄存器组选择标志 (RBS0、RBS1)

这是从 4 组寄存器中选择 1 组的 2 位标志。

此标志保存通过执行 SEL RBn 指令所选寄存器组的 2 位信息。

(d) 辅助进位标志 (AC)

当运算结果在 bit3 产生进位或者借位时，将此标志置“1”。否则，将此标志清“0”。

(e) 优先级控制标志 (ISP1、ISP0)

这是管理能接受的可屏蔽向量中断优先级的标志。禁止接受优先级指定标志寄存器 (PRn0L、PRn0H、PRn1L、PRn1H、PRn2L、PRn2H、PRn3L) (参照 20.3.3) 指定的低于 ISP0 标志值和 ISP1 标志值的向量中断请求。另外，实际上通过中断允许标志 (IE) 的状态控制是否接受向量中断请求。

备注 n=0、1

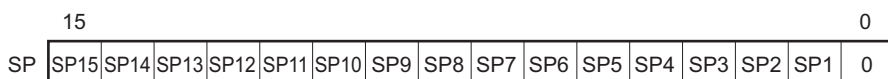
(f) 进位标志 (CY)

这是在执行加减运算指令时保存上溢和下溢的标志。另外，在执行循环指令时保存移出的值，并且在执行位操作指令时用作位累加器。

(3) 堆栈指针 (SP)

这是保存存储器堆栈区起始地址的 16 位寄存器。只有内部 RAM 区才能设定为堆栈区。

图 3-7 堆栈指针的结构



在通过堆栈指针进行堆栈寻址过程中，SP 在写堆栈存储器（压栈）时先递减，而在读堆栈存储器（退栈）后递增。

注意 1. 在产生复位信号后，SP 的内容变为不定值，因此必须在使用堆栈前对 SP 进行初始化。

2. 禁止将通用寄存器 (FFEE0H ~ FFEFFH) 的空间用作取指令或者堆栈区。

3. 在自编程和改写数据闪存时，不能将用于堆栈、数据缓冲器、向量中断处理的转移目标以及 DMA 的传送目标 / 传送源的 RAM 地址分配到 FFE20H ~ FFEDFH 的区。

4. 在自编程时，以下产品的 RAM 区用于各库，所以禁止使用。

R7F0C003、R7F0C004: FDF00H ~ FE309H

3.2.2 通用寄存器

通用寄存器被映像到数据存储器的特定地址（FFEE0H ~ FFEFFH），以 8 个 8 位寄存器（X、A、C、B、E、D、L、H）为 1 组的 4 组寄存器构成。

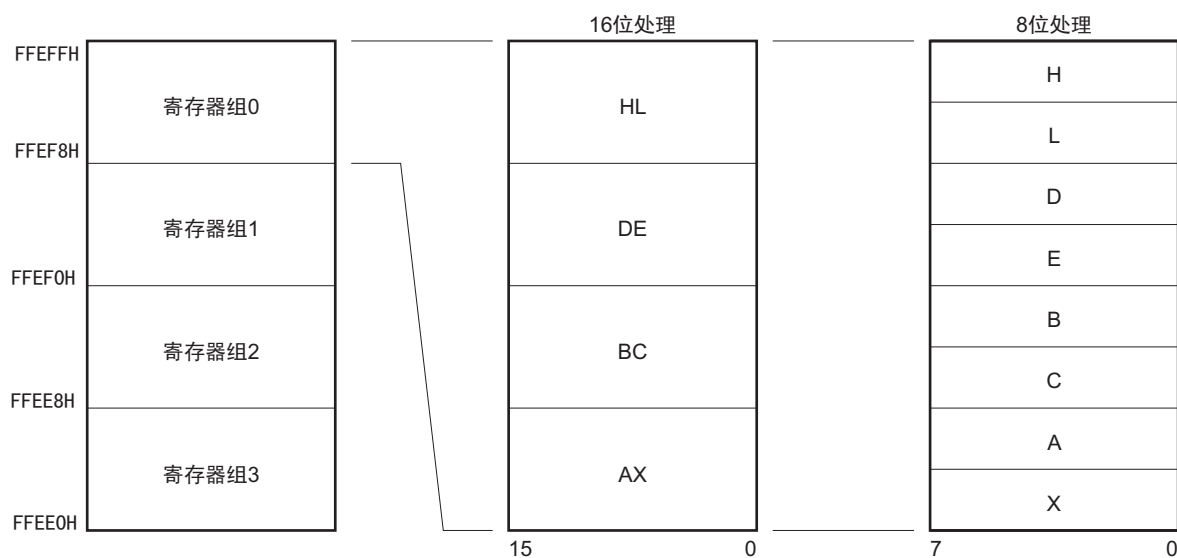
各寄存器除了能分别用作 8 位寄存器以外，还能将 2 个 8 位寄存器成对用作 1 个 16 位寄存器（AX、BC、DE、HL）。

通过 CPU 控制指令（SEL RBn）设定执行指令时使用的寄存器组。因为结构为 4 个寄存器组，所以能对通常处理使用的寄存器和中断处理使用的寄存器进行寄存器组的切换，建立高效率的程序。

注意 禁止将通用寄存器（FFEE0H ~ FFEFFH）的空间用于取指令和堆栈区。

图 3-8 通用寄存器的结构

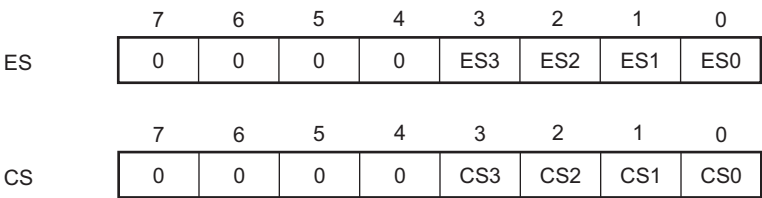
(a) 功能名称



3.2.3 ES 寄存器和 CS 寄存器

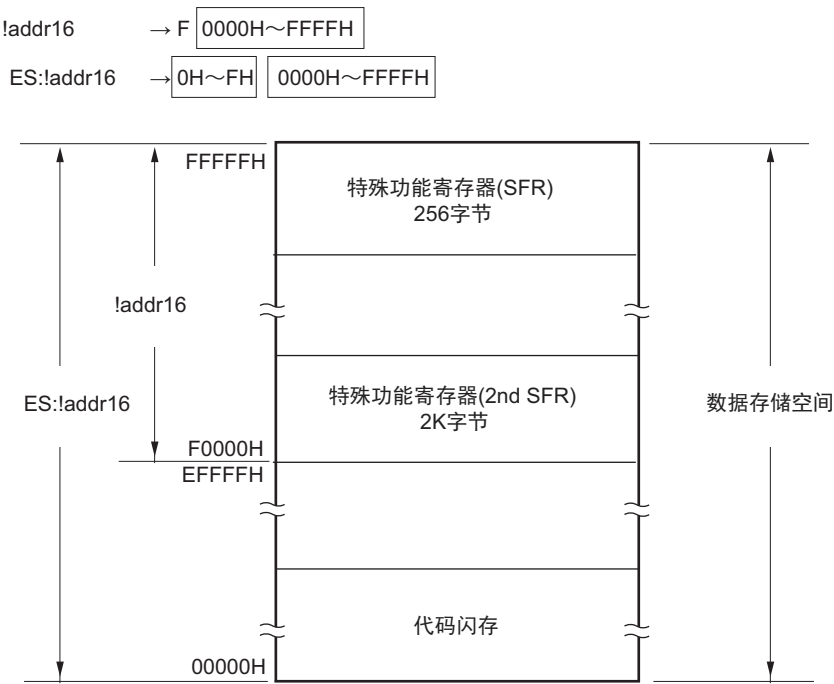
能通过 ES 寄存器指定数据存取并且通过 CS 寄存器（寄存器直接寻址）指定执行转移时的高位地址。
ES 寄存器复位后的初始值为“0FH”，CS 寄存器复位后的初始值为“00H”。

图 3-9 ES 和 CS 寄存器的结构



能通过 16 位地址进行存取的数据区是 F0000H ~ FFFFFH 的 64K 字节空间，但是如果附加“ES:”，就能扩展到 00000H ~ FFFFFH 的 1M 字节空间。

图 3-10 数据存取区的扩展



3.2.4 特殊功能寄存器（SFR：Special Function Register）

SFR 是和通用寄存器不同的并且分别具有特殊功能的寄存器。

SFR 空间分配在 FFF00H ~ FFFFFH 的区域。

和通用寄存器一样，能通过运算指令、传送指令和位操作指令来操作 SFR。能操作的位单位（1、8、16）因各 SFR 而不同。

各操作位单位的指定方法如下所示。

- 1位操作
给1位操作指令的操作数（sfr.bit）进行以下的记述：
已定义位名的情况：<位名>
没有定义位名的情况：<寄存器名>.<位号>或者<地址>.<位号>
- 8位操作
给8位操作指令的操作数（sfr）记述汇编程序定义的符号，也能指定地址。
- 16位操作
给16位操作指令的操作数（sfrp）记述汇编程序定义的符号。当指定地址时，必须记述偶数地址。

SFR 一览表如表 3-5 所示。表中的项目的含义如下所示。

- 符号
这是表示特殊功能寄存器地址的符号。在汇编程序中为保留字，在编译程序中通过 #pragma sfr 指令定义为 sfr 变量。在使用汇编程序、调试程序和仿真程序时，能记述为指令的操作数。
- R/W
表示能否读（Read）写（Write）相应的特殊功能寄存器。
R/W：可读写
R：只能读
W：只能写
- 可操作的位单位
“○”表示能操作的位单位（1、8、16）。“—”表示不能操作的位单位。
- 复位后
表示在产生复位信号后的各个寄存器的状态。

注意 不能存取未分配 SFR 的地址。

备注 有关扩展 SFR（2nd SFR），请参照“3.2.5 扩展特殊功能寄存器（2nd SFR：2nd Special Function Register）”。

表 3-5 SFR 列表 (1/5)

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
FFF05H	端口寄存器 5	P5	R/W	○	○	—	00H
FFF06H	端口寄存器 6	P6	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
FFF10H	串行数据寄存器 00	TXD0/ SIO00	R/W	—	○	○	0000H
FFF11H		—		—	—	—	
FFF12H	串行数据寄存器 01	RXD0	R/W	—	○	○	0000H
FFF13H		—		—	—	—	
FFF14H	串行数据寄存器 12	TXD3	R/W	—	○	○	0000H
FFF15H		—		—	—	—	
FFF16H	串行数据寄存器 13	RXD3	R/W	—	○	○	0000H
FFF17H		—		—	—	—	
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	—	○	○	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
FFF25H	端口模式寄存器 5	PM5	R/W	○	○	—	FFH
FFF26H	端口模式寄存器 6	PM6	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
FFF2DH	端口模式寄存器 13	PM13	R/W	○	○	—	FFH
FFF30H	A/D 转换器模式寄存器 0	ADM0	R/W	○	○	—	00H
FFF31H	模拟输入通道指定寄存器	ADS	R/W	○	○	—	00H
FFF32H	A/D 转换器模式寄存器 1	ADM1	R/W	○	○	—	00H

表 3-5 SFR 列表 (2/5)

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	
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
FFF40H	LCD 模式寄存器 0	LCDM0	R/W	—	○	—	00H
FFF41H	LCD 模式寄存器 1	LCDM1	R/W	○	○	—	00H
FFF42H	LCD 时钟控制寄存器 0	LCDC0	R/W	—	○	—	00H
FFF43H	LCD 引导等级控制寄存器	VLCD	R/W	—	○	—	04H
FFF44H	串行数据寄存器 02	TXD1	SDR02	R/W	—	○	0000H
FFF45H		—			—	—	
FFF46H	串行数据寄存器 03	RXD1	SDR03	R/W	—	○	0000H
FFF47H		—			—	—	
FFF48H	串行数据寄存器 10	TXD2	SDR10	R/W	—	○	0000H
FFF49H		—			—	—	
FFF4AH	串行数据寄存器 11	RXD2	SDR11	R/W	—	○	0000H
FFF4BH		—			—	—	
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
FFF58H	智能卡接收数据寄存器 0	SCRDR0	R	—	○	—	00H
FFF59H	智能卡发送数据寄存器 0	SCTDR0	R/W	—	○	—	FFH
FFF5AH	智能卡接收数据寄存器 1	SCRDR1	R	—	○	—	00H
FFF5BH	智能卡发送数据寄存器 1	SCTDR1	R/W	—	○	—	FFH
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	TDR03	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							

表 3-5 SFR 列表 (3/5)

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	
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	12 位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	—	○	—	不定
FFF93H	分钟计数寄存器	MIN	R/W	—	○	—	不定
FFF94H	小时计数寄存器	HOURL	R/W	—	○	—	不定
FFF95H	星期计数寄存器	WEEK	R/W	—	○	—	不定
FFF96H	日计数寄存器	DAY	R/W	—	○	—	不定
FFF97H	月计数寄存器	MONTH	R/W	—	○	—	不定
FFF98H	年计数寄存器	YEAR	R/W	—	○	—	不定
FFF9AH	闹钟分钟寄存器	ALARMWM	R/W	—	○	—	不定
FFF9BH	闹钟小时寄存器	ALARMWH	R/W	—	○	—	不定
FFF9CH	闹钟星期寄存器	ALARMWW	R/W	—	○	—	不定
FFF9DH	实时时钟控制寄存器 0	RTCC0	R/W	○	○	—	00H 注 1
FFF9EH	实时时钟控制寄存器 1	RTCC1	R/W	○	○	—	00H 注 1
FFFA0H	时钟运行模式控制寄存器	CMC	R/W	—	○	—	00H 注 1
FFFA1H	时钟运行状态控制寄存器	CSC	R/W	○	○	—	C0H 注 1
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
FFFA5H	时钟输出选择寄存器 0	CKS0	R/W	○	○	—	00H
FFFA6H	时钟输出选择寄存器 1	CKS1	R/W	○	○	—	00H
FFFA8H	复位控制标志寄存器	RESF	R	—	○	—	不定 注 2
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 2
FFFABH	看门狗定时器允许寄存器	WDTE	R/W	—	○	—	1AH/9AH 注 3
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

注 1. 只有在产生上电复位时，该寄存器被初始化。

2. 复位值因复位源不同，如下所示。

复位源		RESET 输入	POR 复位	因执行非法指令产生的复位	WDT 复位	因 RAM 奇偶校验错误产生的复位	因非法存取存储器产生的复位	LVD 复位
寄存器								
RESF	TRAP	清“0”		置“1”	保持			保持
	WDTRF			保持	置“1”	保持		
	RPERF			保持		置“1”	保持	
	IAWRF			保持			置“1”	
	LVIRF			保持				
LVIM	LVISEN	清“0”						保持
	LVIOMSK	保持						
	LVIF							
LVIS		清除（00H/01H/81H）						

3. WDTE 寄存器的复位值取决于选项字节的设定。

表 3-5 SFR 列表 (4/5)

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	
FFFB2H	DMA RAM 地址寄存器 0L	DRA0L	DRA0	R/W	—	○	○	00H
FFFB3H	DMA RAM 地址寄存器 0H	DRA0H		R/W	—	○		00H
FFFB4H	DMA RAM 地址寄存器 1L	DRA1L	DRA1	R/W	—	○	○	00H
FFFB5H	DMA RAM 地址寄存器 1H	DRA1H		R/W	—	○		00H
FFFB6H	DMA 字节计数寄存器 0L	DBC0L	DBC0	R/W	—	○	○	00H
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	—	○	○	00H
FFFB9H	DMA 字节计数寄存器 1H	DBC1H		R/W	—	○		00H
FFFBAH	DMA 模式控制寄存器 0	DMC0		R/W	○	○	—	00H
FFFB BH	DMA 模式控制寄存器 1	DMC1		R/W	○	○	—	00H
FFFBCH	DMA 运行控制寄存器 0	DRC0		R/W	○	○	—	00H
FFFB DH	DMA 运行控制寄存器 1	DRC1		R/W	○	○	—	00H
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
FFFD2H	中断请求标志寄存器 3L	IF3L	IF3	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
FFFD6H	中断屏蔽标志寄存器 3L	MK3L	MK3	R/W	○	○	○	FFH
FFFD8H	优先级指定标志寄存器 02L	PR02L	PR02	R/W	○	○	○	FFH
FFFD9H	优先级指定标志寄存器 02H	PR02H		R/W	○	○		FFH
FFFDAH	优先级指定标志寄存器 03L	PR03L	PR03	R/W	○	○	○	FFH
FFFDCH	优先级指定标志寄存器 12L	PR12L	PR12	R/W	○	○	○	FFH
FFDDH	优先级指定标志寄存器 12H	PR12H		R/W	○	○		FFH
FFFDEH	优先级指定标志寄存器 13L	PR13L	PR13	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 0H	IF0H		R/W	○	○		00H
FFFE2H	中断请求标志寄存器 1L	IF1L	IF1	R/W	○	○	○	00H
FFFE3H	中断请求标志寄存器 1H	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 0H	MK0H		R/W	○	○		FFH
FFFE6H	中断屏蔽标志寄存器 1L	MK1L	MK1	R/W	○	○	○	FFH
FFFE7H	中断屏蔽标志寄存器 1H	MK1H		R/W	○	○		FFH
FFFE8H	优先级指定标志寄存器 00L	PR00L	PR00	R/W	○	○	○	FFH
FFFE9H	优先级指定标志寄存器 00H	PR00H		R/W	○	○		FFH

表 3-5 SFR 列表 (5/5)

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	
FFFEAH	优先级指定标志寄存器 01L	PR01L	PR01	R/W	○	○	○	FFH
FFFEBH	优先级指定标志寄存器 01H	PR01H		R/W	○	○		FFH
FFFECH	优先级指定标志寄存器 10L	PR10L	PR10	R/W	○	○	○	FFH
FFFEDH	优先级指定标志寄存器 10H	PR10H		R/W	○	○		FFH
FFFEEH	优先级指定标志寄存器 11L	PR11L	PR11	R/W	○	○	○	FFH
FFFEFH	优先级指定标志寄存器 11H	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 有关扩展 SFR (2nd SFR)，请参照“表 3-6 扩展 SFR (2nd SFR) 列表”。

3.2.5 扩展特殊功能寄存器（2nd SFR: 2nd Special Function Register）

扩展 SFR（2nd SFR）是和通用寄存器不同的并且分别具有特殊功能的寄存器。

扩展 SFR 空间分配在 F0000H ~ F07FFH 的区域。在此区域中分配了 SFR 区（FFF00H ~ FFFFFH）以外的 SFR，但是扩展 SFR 区的存取指令比 SFR 区长 1 字节。

和通用寄存器一样，能通过运算指令、传送指令和位操作指令来操作扩展 SFR。能操作的位单位（1、8、16）因各扩展 SFR 而不同。

各操作位单位的指定方法如下所示。

- 1 位操作
给 1 位操作指令的操作数（!addr16.bit）进行以下的记述：
已定义位名的情况：<位名>
没有定义位名的情况：<寄存器名>.<位号>或者<地址>.<位号>
- 8 位操作
给 8 位操作指令的操作数（!addr16）记述汇编程序定义的符号，也能指定地址。
- 16 位操作
给 16 位操作指令的操作数（!addr16）记述汇编程序定义的符号。当指定地址时，必须记述偶数地址。

扩展 SFR 一览表如表 3-6 所示。表中的项目的含义如下所示。

- 符号
这是表示扩展 SFR 地址的符号。在汇编程序中为保留字，在编译程序中通过 #pragma sfr 指令定义为 sfr 变量。在使用汇编程序、调试程序和仿真程序时，能记述为指令的操作数。
- R/W
表示能否读（Read）写（Write）相应的扩展 SFR。
R/W：可读写
R：只能读
W：只能写
- 可操作的位单位
“○”表示能操作的位单位（1、8、16）。“—”表示不能操作的位单位。
- 复位后
表示产生复位信号后的各个寄存器的状态。

注意 不能存取未分配扩展 SFR（2nd SFR）的地址。

备注 有关 SFR 区的 SFR，请参照“3.2.4 特殊功能寄存器（SFR: Special Function Register）”。

表 3-6 扩展 SFR (2nd SFR) 列表 (1/8)

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定寄存器	ADUL	R/W	—	○	—	FFH
F0012H	转换结果比较下限值设定寄存器	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
F0032H	上拉电阻选择寄存器 2	PU2	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
F0035H	上拉电阻选择寄存器 5	PU5	R/W	○	○	—	00H
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
F003DH	上拉电阻选择寄存器 13	PU13	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
F0043H	端口输入模式寄存器 3	PIM3	R/W	○	○	—	00H
F0044H	端口输入模式寄存器 4	PIM4	R/W	○	○	—	00H
F0045H	端口输入模式寄存器 5	PIM5	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
F0053H	端口输出模式寄存器 3	POM3	R/W	○	○	—	00H
F0054H	端口输出模式寄存器 4	POM4	R/W	○	○	—	00H
F0055H	端口输出模式寄存器 5	POM5	R/W	○	○	—	00H
F005DH	端口输出模式寄存器 13	POM13	R/W	○	○	—	00H
F0061H	端口模式控制寄存器 1	PMC1	R/W	○	○	—	FFH
F0062H	端口模式控制寄存器 2	PMC2	R/W	○	○	—	FFH
F0064H	端口模式控制寄存器 4	PMC4	R/W	○	○	—	00H
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
F0077H	外围 I/O 重定向寄存器	PIOR	R/W	—	○	—	00H
F0078H	非法存储器存取检测控制寄存器	IAWCTL	R/W	—	○	—	00H
F007AH	外围允许寄存器 1	PER1	R/W	○	○	—	00H
F007BH	端口模式选择寄存器	PMS	R/W	○	○	—	00H

表 3-6 扩展 SFR (2nd SFR) 列表 (2/8)

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	
F00A0H	高速内部振荡器的微调寄存器	HIOTRM		R/W	—	○	—	不定注 1
F00A8H	高速内部振荡器频率选择寄存器	HOCODIV		R/W	—	○	—	不定注 2
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	副系统时钟提供模式控制寄存器	OSMC		R/W	—	○	—	00H
F00F5H	RAM 奇偶校验错误控制寄存器	RPECTL		R/W	○	○	—	00H
F00F9H	上电复位状态寄存器	PORSR		R/W	—	○	—	00H 注 3
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	—	—	○	0020H
F0111H								
F0112H	串行模式寄存器 01	SMR01		R/W	—	—	○	0020H
F0113H								
F0114H	串行模式寄存器 02	SMR02		R/W	—	—	○	0020H
F0115H								
F0116H	串行模式寄存器 03	SMR03		R/W	—	—	○	0020H
F0117H								

- 注 1. 复位值是产品发货时的调整值。
 2. 通过选项字节 000C2H 的 FRQSEL2 ~ FRQSEL0 设定的值。
 3. 只有在产生上电复位时, 该寄存器被初始化。

表 3-6 扩展 SFR (2nd SFR) 列表 (3/8)

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	
F0118H	串行通信运行设定寄存器 00	SCR00		R/W	—	—	○	0087H
F0119H								
F011AH	串行通信运行设定寄存器 01	SCR01		R/W	—	—	○	0087H
F011BH								
F011CH	串行通信运行设定寄存器 02	SCR02		R/W	—	—	○	0087H
F011DH								
F011EH	串行通信运行设定寄存器 03	SCR03		R/W	—	—	○	0087H
F011FH								
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	—	—	○	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
F0139H		—			—	—		
F0140H	串行状态寄存器 10	SSR10L	SSR10	R	—	○	○	0000H
F0141H		—			—	—		
F0142H	串行状态寄存器 11	SSR11L	SSR11	R	—	○	○	0000H
F0143H		—			—	—		
F0144H	串行状态寄存器 12	SSR12L	SSR12	R	—	○	○	0000H
F0145H		—			—	—		
F0146H	串行状态寄存器 13	SSR13L	SSR13	R	—	○	○	0000H
F0147H		—			—	—		
F0148H	串行标志清除触发寄存器 10	SIR10L	SIR10	R/W	—	○	○	0000H
F0149H		—			—	—		
F014AH	串行标志清除触发寄存器 11	SIR11L	SIR11	R/W	—	○	○	0000H
F014BH		—			—	—		
F014CH	串行标志清除触发寄存器 12	SIR12L	SIR12	R/W	—	○	○	0000H
F014DH		—			—	—		
F014EH	串行标志清除触发寄存器 13	SIR13L	SIR13	R/W	—	○	○	0000H
F014FH		—			—	—		
F0150H	串行模式寄存器 10	SMR10		R/W	—	—	○	0020H
F0151H								

表 3-6 扩展 SFR (2nd SFR) 列表 (4/8)

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	
F0152H	串行模式寄存器 11	SMR11		R/W	—	—	○	0020H
F0153H								
F0154H	串行模式寄存器 12	SMR12		R/W	—	—	○	0020H
F0155H								
F0156H	串行模式寄存器 13	SMR13		R/W	—	—	○	0020H
F0157H								
F0158H	串行通信运行设定寄存器 10	SCR10		R/W	—	—	○	0087H
F0159H								
F015AH	串行通信运行设定寄存器 11	SCR11		R/W	—	—	○	0087H
F015BH								
F015CH	串行通信运行设定寄存器 12	SCR12		R/W	—	—	○	0087H
F015DH								
F015EH	串行通信运行设定寄存器 13	SCR13		R/W	—	—	○	0087H
F015FH								
F0160H	串行通道允许状态寄存器 1	SE1L	SE1	R	○	○	○	0000H
F0161H		—			—	—		
F0162H	串行通道开始寄存器 1	SS1L	SS1	R/W	○	○	○	0000H
F0163H		—			—	—		
F0164H	串行通道停止寄存器 1	ST1L	ST1	R/W	○	○	○	0000H
F0165H		—			—	—		
F0166H	串行时钟选择寄存器 1	SPS1L	SPS1	R/W	—	○	○	0000H
F0167H		—			—	—		
F0168H	串行输出寄存器 1	SO1		R/W	—	—	○	0F0FH
F0169H								
F016AH	串行输出允许寄存器 1	SOE1L	SOE1	R/W	○	○	○	0000H
F016BH		—			—	—		
F0174H	串行输出电平寄存器 1	SOL1L	SOL1	R/W	—	○	○	0000H
F0175H		—			—	—		
F0178H	串行待机控制寄存器 1	SSC1L	SSC1	R/W	—	○	○	0000H
F0179H		—			—	—		
F0180H	定时器计数寄存器 00	TCR00		R	—	—	○	FFFFH
F0181H								
F0182H	定时器计数寄存器 01	TCR01		R	—	—	○	FFFFH
F0183H								
F0184H	定时器数据寄存器 02	TCR02		R	—	—	○	FFFFH
F0185H								
F0186H	定时器计数寄存器 03	TCR03		R	—	—	○	FFFFH
F0187H								
F0188H	定时器计数寄存器 04	TCR04		R	—	—	○	FFFFH
F0189H								

表 3-6 扩展 SFR (2nd SFR) 列表 (5/8)

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	
F018AH	定时器计数寄存器 05	TCR05	R	—	—	○	FFFFH
F018BH				—	—	○	
F018CH	定时器计数寄存器 06	TCR06	R	—	—	○	FFFFH
F018DH				—	—	○	
F018EH	定时器计数寄存器 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				—	—	○	
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	R	—	○	○	0000H
F01A1H		—		—	—		
F01A2H	定时器状态寄存器 01	TSR01L	R	—	○	○	0000H
F01A3H		—		—	—		
F01A4H	定时器数据寄存器 02	TSR02L	R	—	○	○	0000H
F01A5H		—		—	—		
F01A6H	定时器状态寄存器 03	TSR03L	R	—	○	○	0000H
F01A7H		—		—	—		
F01A8H	定时器状态寄存器 04	TSR04L	R	—	○	○	0000H
F01A9H		—		—	—		
F01AAH	定时器状态寄存器 05	TSR05L	R	—	○	○	0000H
F01ABH		—		—	—		
F01ACH	定时器状态寄存器 06	TSR06L	R	—	○	○	0000H
F01ADH		—		—	—		
F01AEH	定时器状态寄存器 07	TSR07L	R	—	○	○	0000H
F01AFH		—		—	—		
F01B0H	定时器通道允许状态寄存器 0	TE0L	R	○	○	○	0000H
F01B1H		—		—	—		
F01B2H	定时器通道开始寄存器 0	TS0L	R/W	○	○	○	0000H
F01B3H		—		—	—		

表 3-6 扩展 SFR (2nd SFR) 列表 (6/8)

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	
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	—	—	○	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		—			—	—		
F0200H	DMA SFR 地址寄存器 2	DSA2		R/W	—	○	—	00H
F0201H	DMA SFR 地址寄存器 3	DSA3		R/W	—	○	—	00H
F0202H	DMA RAM 地址寄存器 2L	DRA2L	DRA2	R/W	—	○	○	00H
F0203H	DMA RAM 地址寄存器 2H	DRA2H		R/W	—	○		00H
F0204H	DMA RAM 地址寄存器 3L	DRA3L	DRA3	R/W	—	○	○	00H
F0205H	DMA RAM 地址寄存器 3H	DRA3H		R/W	—	○		00H
F0206H	DMA 字节计数寄存器 2L	DBC2L	DBC2	R/W	—	○	○	00H
F0207H	DMA 字节计数寄存器 2H	DBC2H		R/W	—	○		00H
F0208H	DMA 字节计数寄存器 3L	DBC3L	DBC3	R/W	—	○	○	00H
F0209H	DMA 字节计数寄存器 3H	DBC3H		R/W	—	○		00H
F020AH	DMA 模式控制寄存器 2	DMC2		R/W	○	○	—	00H
F020BH	DMA 模式控制寄存器 3	DMC3		R/W	○	○	—	00H
F020CH	DMA 运行控制寄存器 2	DRC2		R/W	○	○	—	00H
F020DH	DMA 运行控制寄存器 3	DRC3		R/W	○	○	—	00H
F0230H	IICA 控制寄存器 00	IICCTL00		R/W	○	○	—	00H
F0231H	IICA 控制寄存器 01	IICCTL01		R/W	○	○	—	00H
F0232H	IICA 低电平宽度设定寄存器 0	IICWL0		R/W	—	○	—	FFH
F0233H	IICA 高电平宽度设定寄存器 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
F0300H	LCD 端口功能寄存器 0	PFSEG0		R/W	○	○	—	F0H
F0301H	LCD 端口功能寄存器 1	PFSEG1		R/W	○	○	—	FFH
F0302H	LCD 端口功能寄存器 2	PFSEG2		R/W	○	○	—	FFH
F0303H	LCD 端口功能寄存器 3	PFSEG3		R/W	○	○	—	FFH
F0304H	LCD 端口功能寄存器 4	PFSEG4		R/W	○	○	—	FFH
F0305H	LCD 端口功能寄存器 5	PFSEG5		R/W	○	○	—	FFH
F0306H	LCD 端口功能寄存器 6	PFSEG6		R/W	○	○	—	0FH
F0308H	LCD 输入切换控制寄存器	ISCLCD		R/W	○	○	—	00H

表 3-6 扩展 SFR (2nd SFR) 列表 (7/8)

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	
F0310H	时钟误差修正寄存器	SUBCUD	R/W	—	—	○	0020H 注
F0311H							
F0312H	频率测量计数寄存器 L	FMCRL	R	—	—	○	0000H
F0313H							
F0314H	频率测量计数寄存器 H	FMCRLH	R	—	—	○	0000H
F0315H							
F0316H	频率测量控制寄存器	FMCTL	R/W	○	—	—	00H
F0322H	智能卡串行模式寄存器 0	SCMR0	R/W	—	○	—	18H
F0323H	智能卡串行控制寄存器 0	SCCR0	R/W	—	○	—	00H
F0324H	智能卡串行状态寄存器 0	SCSR0	R/W	—	○	—	84H
F0325H	智能卡位速率寄存器 0	SCBRR0	R/W	—	○	—	0FH
F032AH	频率测量控制寄存器 1	SCMR1	R/W	—	○	—	18H
F032BH	智能卡串行模式寄存器 1	SCCR1	R/W	—	○	—	00H
F032CH	智能卡串行控制寄存器 1	SCSR1	R/W	—	○	—	84H
F032DH	智能卡串行状态寄存器 1	SCBRR1	R/W	—	○	—	0FH
F0340H	比较器模式设定寄存器	COMPMDR	R/W	○	○	—	00H
F0341H	比较器的滤波器控制寄存器	COMPFIR	R/W	○	○	—	00H
F0342H	比较器输出控制寄存器	COMPOCR	R/W	○	○	—	00H
F0400H	LCD 显示数据存储器 0	SEG0	R/W	—	○	—	00H
F0401H	LCD 显示数据存储器 1	SEG1	R/W	—	○	—	00H
F0402H	LCD 显示数据存储器 2	SEG2	R/W	—	○	—	00H
F0403H	LCD 显示数据存储器 3	SEG3	R/W	—	○	—	00H
F0404H	LCD 显示数据存储器 4	SEG4	R/W	—	○	—	00H
F0405H	LCD 显示数据存储器 5	SEG5	R/W	—	○	—	00H
F0406H	LCD 显示数据存储器 6	SEG6	R/W	—	○	—	00H
F0407H	LCD 显示数据存储器 7	SEG7	R/W	—	○	—	00H
F0408H	LCD 显示数据存储器 8	SEG8	R/W	—	○	—	00H
F0409H	LCD 显示数据存储器 9	SEG9	R/W	—	○	—	00H
F040AH	LCD 显示数据存储器 10	SEG10	R/W	—	○	—	00H
F040BH	LCD 显示数据存储器 11	SEG11	R/W	—	○	—	00H
F040CH	LCD 显示数据存储器 12	SEG12	R/W	—	○	—	00H
F040DH	LCD 显示数据存储器 13	SEG13	R/W	—	○	—	00H

注 只有在产生上电复位时，该寄存器被复位。

表 3-6 扩展 SFR (2nd SFR) 列表 (8/8)

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	
F040EH	LCD 显示数据存储器 14	SEG14	R/W	—	○	—	00H
F040FH	LCD 显示数据存储器 15	SEG15	R/W	—	○	—	00H
F0410H	LCD 显示数据存储器 16	SEG16	R/W	—	○	—	00H
F0411H	LCD 显示数据存储器 17	SEG17	R/W	—	○	—	00H
F0412H	LCD 显示数据存储器 18	SEG18	R/W	—	○	—	00H
F0413H	LCD 显示数据存储器 19	SEG19	R/W	—	○	—	00H
F0414H	LCD 显示数据存储器 20	SEG20	R/W	—	○	—	00H
F0415H	LCD 显示数据存储器 21	SEG21	R/W	—	○	—	00H
F0416H	LCD 显示数据存储器 22	SEG22	R/W	—	○	—	00H
F0417H	LCD 显示数据存储器 23	SEG23	R/W	—	○	—	00H
F0418H	LCD 显示数据存储器 24	SEG24	R/W	—	○	—	00H
F0419H	LCD 显示数据存储器 25	SEG25	R/W	—	○	—	00H
F041AH	禁止使用						
F041BH	LCD 显示数据存储器 26	SEG26	R/W	—	○	—	00H
F041CH	LCD 显示数据存储器 27	SEG27	R/W	—	○	—	00H
F041DH	LCD 显示数据存储器 28	SEG28	R/W	—	○	—	00H
F041EH	LCD 显示数据存储器 29	SEG29	R/W	—	○	—	00H
F041FH	LCD 显示数据存储器 30	SEG30	R/W	—	○	—	00H
F0420H	LCD 显示数据存储器 31	SEG31	R/W	—	○	—	00H
F0421H	LCD 显示数据存储器 32	SEG32	R/W	—	○	—	00H
F0422H	LCD 显示数据存储器 33	SEG33	R/W	—	○	—	00H
F0423H	LCD 显示数据存储器 34	SEG34	R/W	—	○	—	00H
F0424H	LCD 显示数据存储器 35	SEG35	R/W	—	○	—	00H
F0425H	LCD 显示数据存储器 36	SEG36	R/W	—	○	—	00H
F0426H	LCD 显示数据存储器 37	SEG37	R/W	—	○	—	00H
F0427H	LCD 显示数据存储器 38	SEG38	R/W	—	○	—	00H
F0428H	LCD 显示数据存储器 39	SEG39	R/W	—	○	—	00H
F0429H	LCD 显示数据存储器 40	SEG40	R/W	—	○	—	00H
F042AH	LCD 显示数据存储器 41	SEG41	R/W	—	○	—	00H
F042BH	LCD 显示数据存储器 42	SEG42	R/W	—	○	—	00H
F042CH	LCD 显示数据存储器 43	SEG43	R/W	—	○	—	00H
F042DH	LCD 显示数据存储器 44	SEG44	R/W	—	○	—	00H
F042EH	LCD 显示数据存储器 45	SEG45	R/W	—	○	—	00H
F042FH	LCD 显示数据存储器 46	SEG46	R/W	—	○	—	00H
F0430H	LCD 显示数据存储器 47	SEG47	R/W	—	○	—	00H
F0431H	LCD 显示数据存储器 48	SEG48	R/W	—	○	—	00H
F0432H	LCD 显示数据存储器 49	SEG49	R/W	—	○	—	00H
F0433H	LCD 显示数据存储器 50	SEG50	R/W	—	○	—	00H

备注 有关 SFR 区的 SFR，请参照“表 3-5 SFR 列表”。

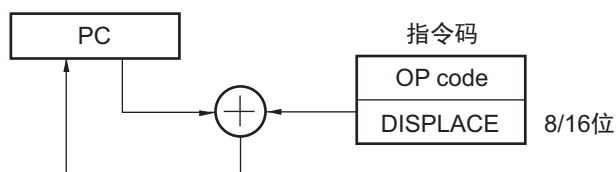
3.3 指令地址寻址

3.3.1 相对寻址

[功能]

相对寻址将指令码中的位移量（带符号的补码数据：-128 ~ +127 或者 -32768 ~ +32767）加上程序计数器（PC）的值（下一条指令的起始地址），结果保存在程序计数器（PC）并且指定转移目标的程序地址。相对寻址只适用于转移指令。

图 3-11 相对寻址的概要



3.3.2 立即寻址

[功能]

立即寻址将指令码中的立即数保存到程序计数器，指定转移目标的程序地址。

在立即寻址中有指定 20 位地址的 CALL !!addr20/BR !!addr20 和指定 16 位地址的 CALL !addr16/BR !addr16。当指定 16 位地址时，将高 4 位置“0000”。

图 3-12 CALL !!addr20/BR !!addr20 的例子

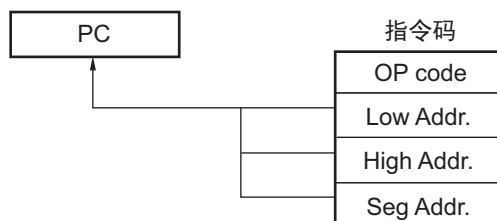
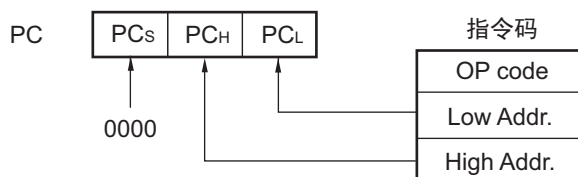


图 3-13 CALL !addr16/BR !addr16 的例子



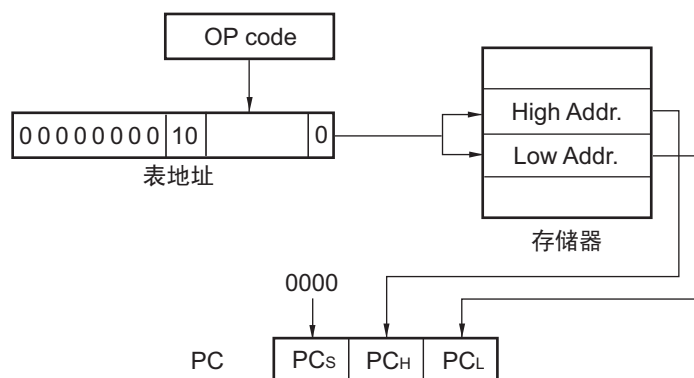
3.3.3 表间接寻址

[功能]

表间接寻址通过指令码中的 5 位立即数指定 CALLT 表区（0080H ~ 00BFH）中的表地址，将此内容和其后续的地址内容作为 16 位数据保存到程序计数器（PC），指定程序地址。表间接寻址只适用于 CALLT 指令。

RL78 单片机只能在 00000H ~ 0FFFFH 的 64K 字节空间中进行转移。

图 3-14 表间接寻址的概要

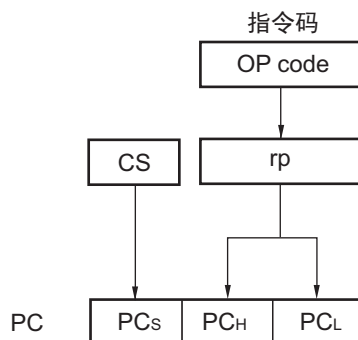


3.3.4 寄存器直接寻址

[功能]

寄存器直接寻址将指令码指定的当前寄存器组的通用寄存器对（AX/BC/DE/HL）和 CS 寄存器的内容作为 20 位数据保存到程序计数器（PC），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 3-15 寄存器直接寻址的概要



3.4 处理数据地址的寻址

3.4.1 隐含寻址

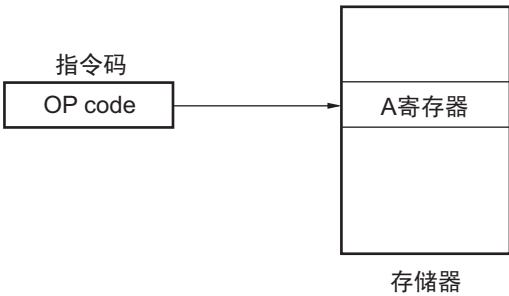
[功能]

对于具有累加器等特殊功能寄存器的存取指令，指令码中没有寄存器指定字段，而通过指令码直接指定。

[操作数形式]

隐含寻址只适用于 MULU X 指令。

图 3-16 隐含寻址的概要



3.4.2 寄存器寻址

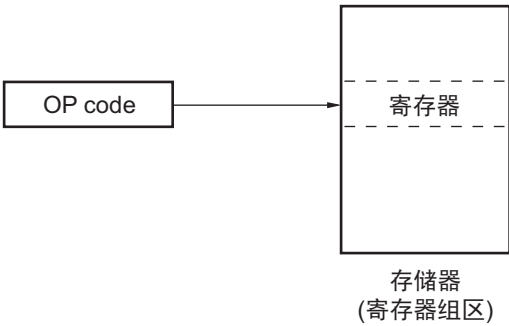
[功能]

寄存器寻址是将通用寄存器作为操作数进行存取的寻址方式。当指定 8 位寄存器时，通过指令码中的 3 位选择寄存器；当指定 16 位寄存器时，通过指令码中的 2 位选择寄存器。

[操作数形式]

表现形式	记述方法
r	X、A、C、B、E、D、L、H
rp	AX、BC、DE、HL

图 3-17 寄存器寻址的图示



3.4.3 直接寻址

[功能]

直接寻址是以指令码中的立即数为操作数地址来直接指定对象地址的寻址方式。

[操作数形式]

表现形式	记述方法
!addr16	标号或者 16 位立即数 (只能指定 F0000H ~ FFFFFH 的空间)
ES:!addr16	标号或者 16 位立即数 (通过 ES 寄存器指定高 4 位地址)

图 3-18 !addr16 的例子

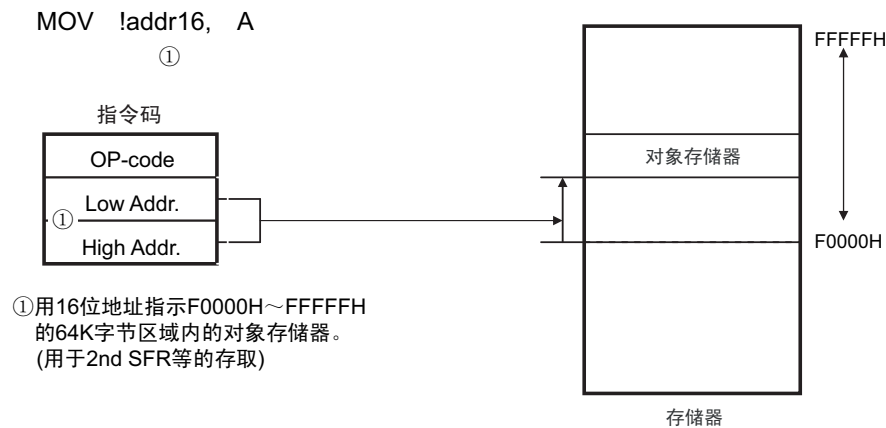
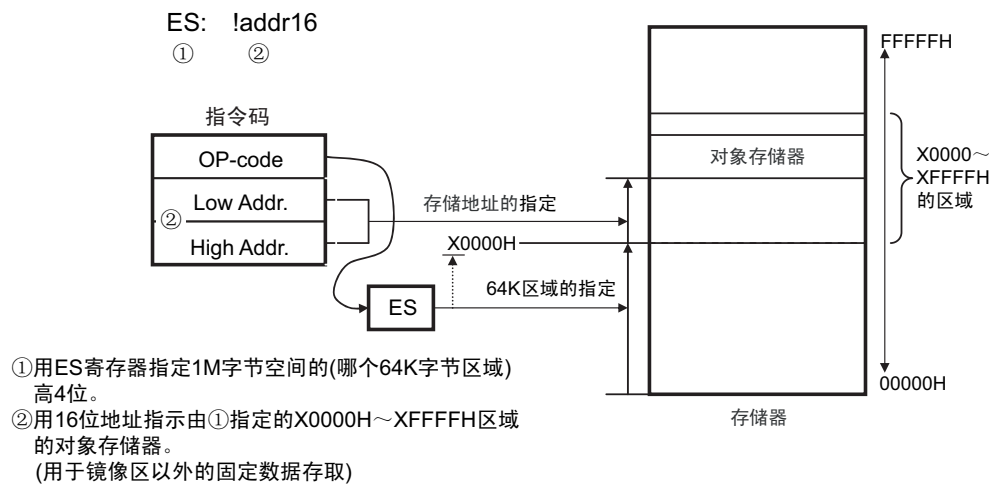


图 3-19 ES:!addr16 的例子



3.4.4 短直接寻址

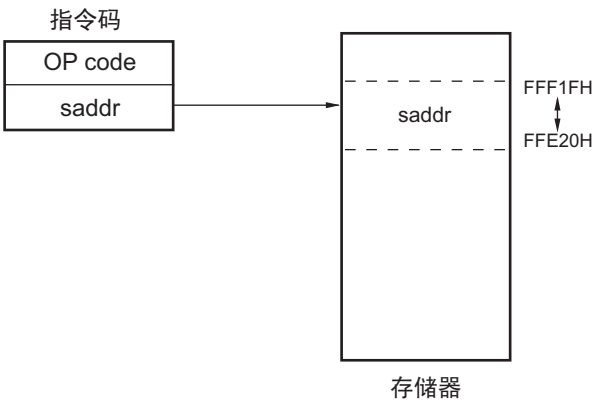
[功能]

短直接寻址是通过指令码中的 8 位数据直接指定对象地址的寻址方式。此寻址方式只适用于 FFE20H ~ FFF1FH 的空间。

[操作数形式]

表现形式	记述方法
SADDR	标号、 FFE20H ~ FFF1FH 的立即数或者 0FE20H ~ 0FF1FH 的立即数 (只能指定 FFE20H ~ FFF1FH 的空间)
SADDRP	标号、 FFE20H ~ FFF1FH 的立即数或者 0FE20H ~ 0FF1FH 的立即数 (只限于偶数地址) (只能指定 FFE20H ~ FFF1FH 的空间)

图 3-20 短直接寻址的图示



备注 SADDR 和 SADDRP 能通过 16 位立即数（省略了实际地址的高 4 位）记述 FE20H ~ FF1FH 的值，并且能通过 20 位立即数记述 FFE20H ~ FFF1FH 的值。
但是，无论使用哪种形式，都指定存储器的 FFE20H ~ FFF1FH 空间地址。

3.4.5 SFR 寻址

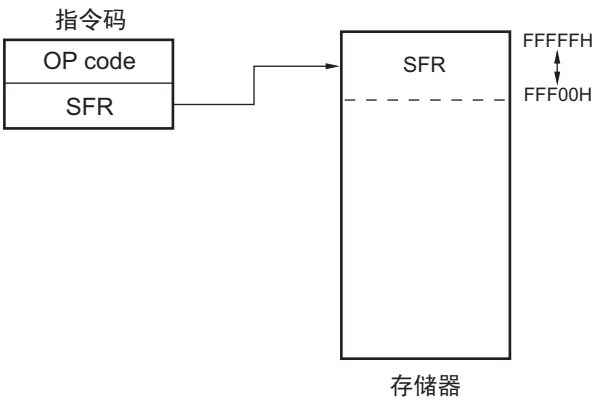
[功能]

SFR 寻址是通过指令码中的 8 位数据直接指定对象 SFR 地址的寻址方式。此寻址方式只适用于 FFF00H ~ FFFFFH 的空间。

[操作数形式]

表现形式	记述方法
SFR	SFR 名称
SFRP	16 位可操作 SFR 名称 （只限偶数地址）

图 3-21 SFR 寻址的图示



3.4.6 寄存器间接寻址

[功能]

寄存器间接寻址以指令码指定的寄存器对的内容为操作数地址，指定对象地址。

[操作数形式]

表现形式	记述方法
—	[DE]、[HL] (只能指定 F0000H ~ FFFFFH 的空间)
—	ES:[DE]、ES:[HL] (通过 ES 寄存器指定高 4 位地址)

图 3-22 [DE]、[HL] 的例子

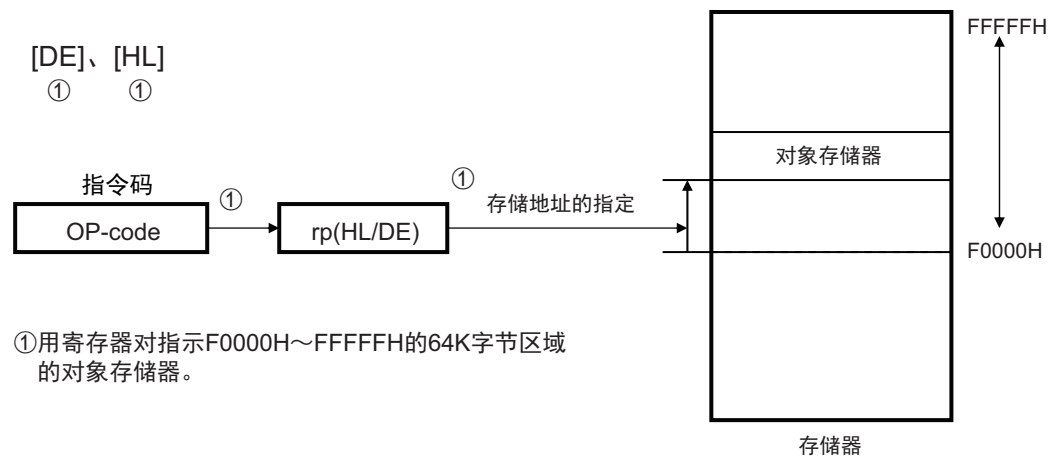
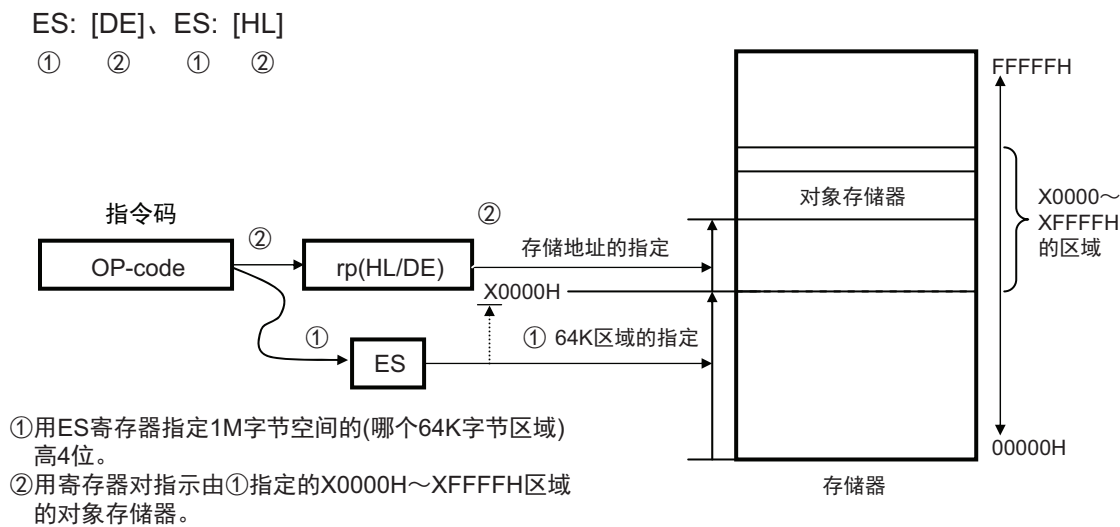


图 3-23 ES:[DE]、ES:[HL] 的例子



3.4.7 基址寻址

[功能]

基址寻址以指令码指定的寄存器对的内容或者 16 位立即数为基址，以 8 位立即数或者 16 位立即数为偏移量，用基址和偏移量的相加结果指定对象地址。

[操作数形式]

表现形式	记述方法
—	[HL+byte]、[DE+byte]、[SP+byte] (只能指定 F0000H ~ FFFFFH 的空间)
—	word[B]、word[C] (只能指定 F0000H ~ FFFFFH 的空间)
—	word[BC] (只能指定 F0000H ~ FFFFFH 的空间)
—	ES:[HL+byte]、ES:[DE+byte] (通过 ES 寄存器指定高 4 位地址)
—	ES:word[B]、ES:word[C] (通过 ES 寄存器指定高 4 位地址)
—	ES:word[BC] (通过 ES 寄存器指定高 4 位地址)

图 3-24 [SP+byte] 的例子

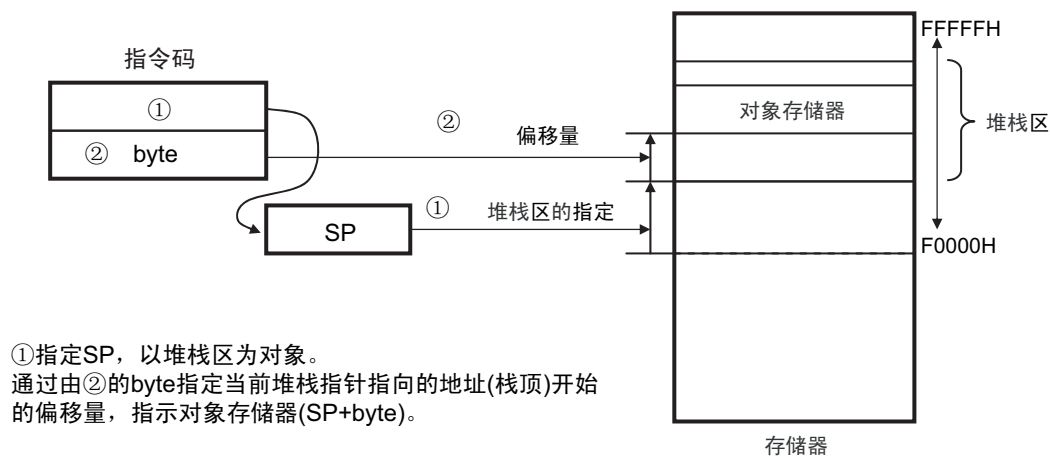


图 3-25 [HL + byte]、[DE + byte] 的例子

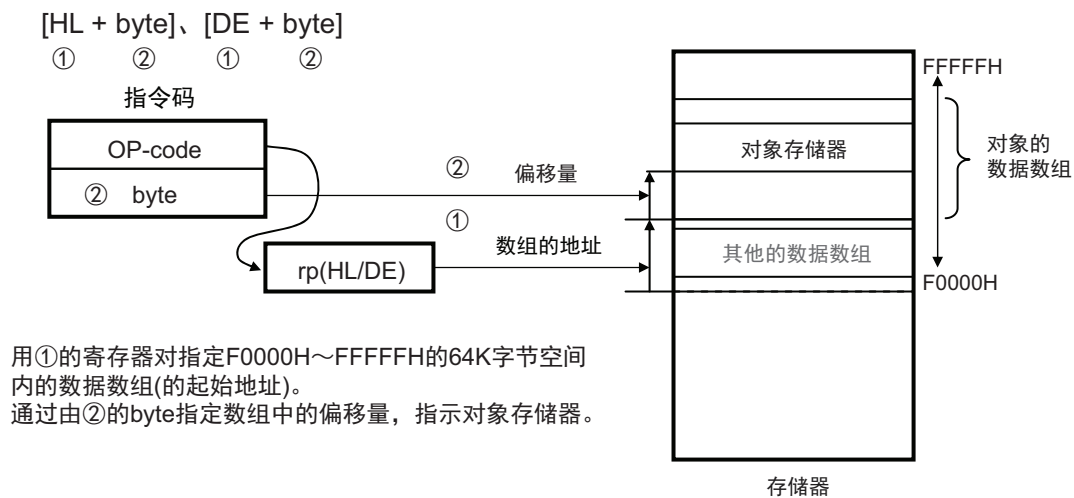


图 3-26 word[B]、Word[C] 的例子

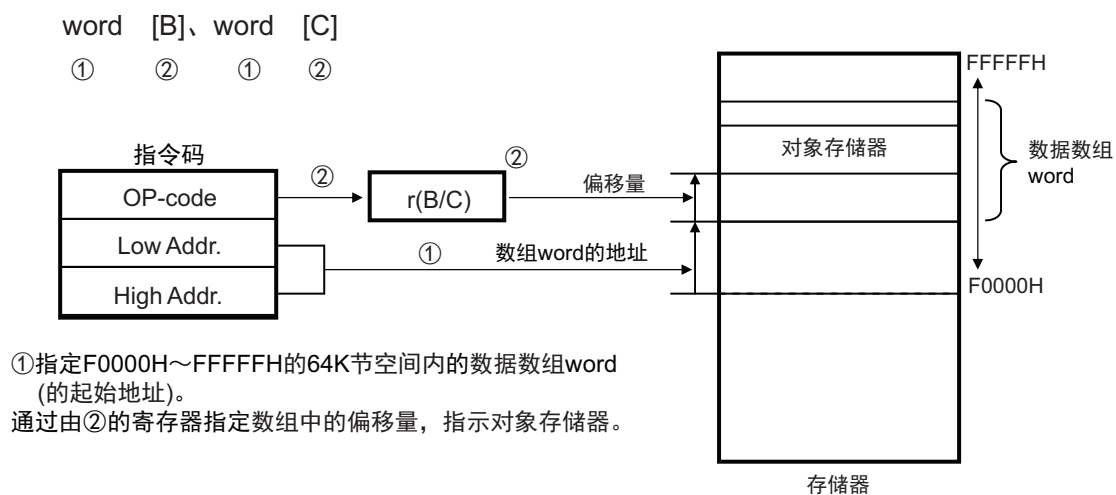


图 3-27 word[BC] 的例子

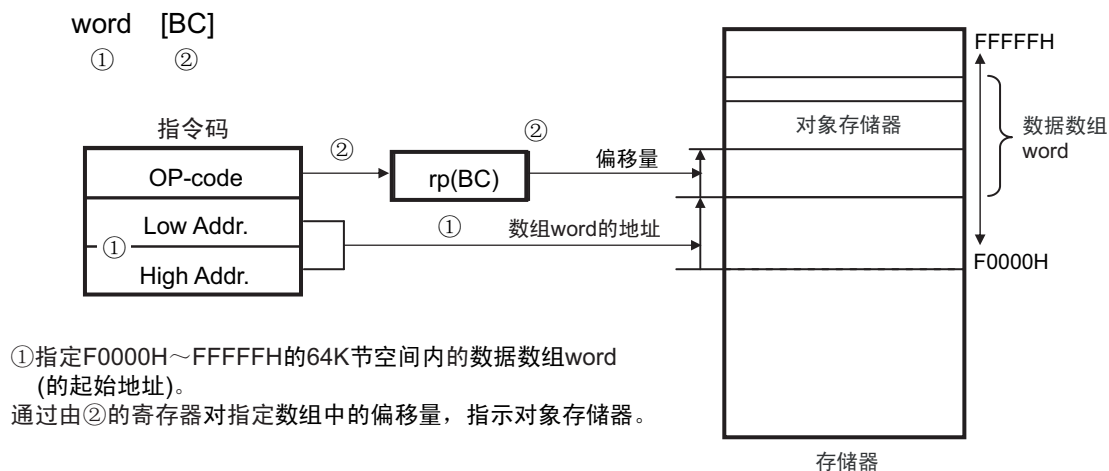


图 3-28 ES:[HL + byte]、ES:[DE + byte] 的例子

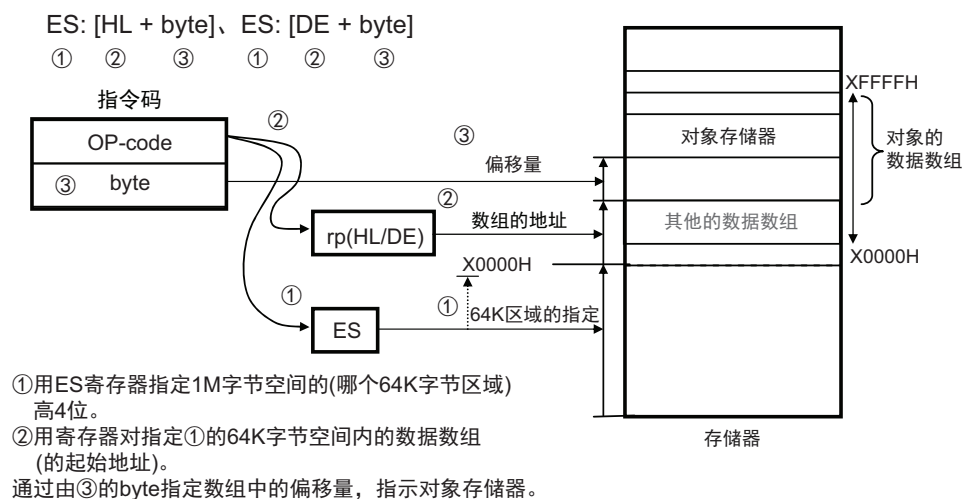


图 3-29 ES:word[B]、ES:word[C] 的例子

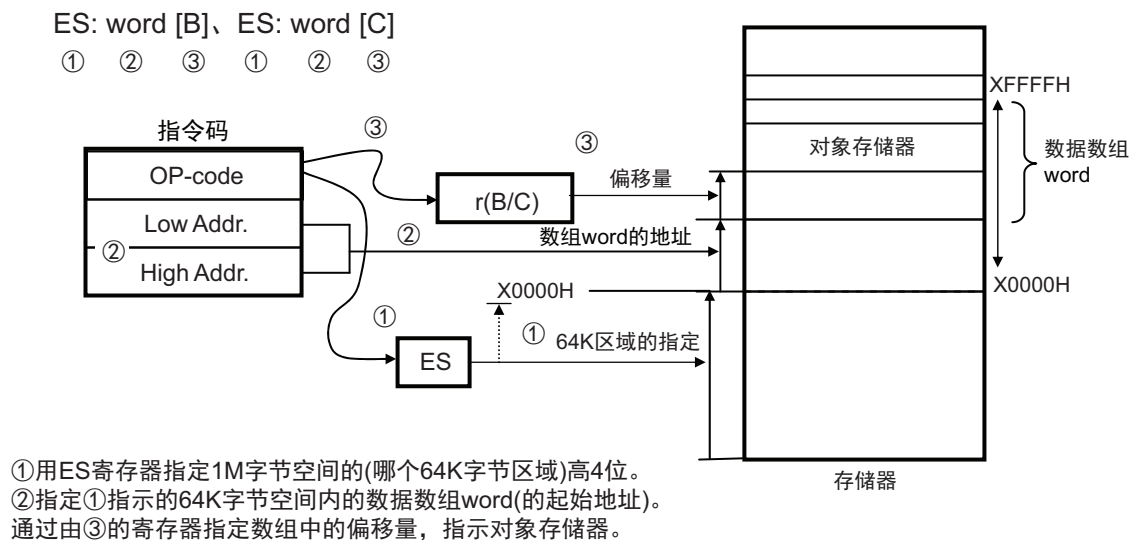
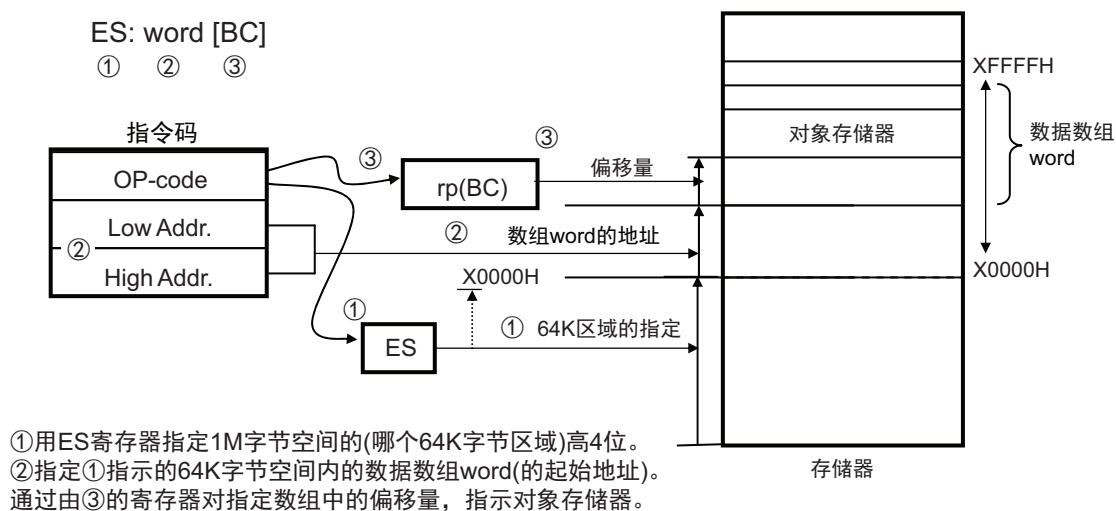


图 3-30 ES:word[BC] 的例子



3.4.8 基址变址寻址

[功能]

基址变址寻址以指令码指定的寄存器对的内容为基址，以指令码指定的 B 寄存器或者 C 寄存器的内容为偏移地址，用基址和偏移地址的相加结果指定对象地址。

[操作数形式]

表现形式	记述方法
—	[HL+B]、 [HL+C] (只能指定 F0000H ~ FFFFFH 的空间)
—	ES:[HL+B]、 ES:[HL+C] (通过 ES 寄存器指定高 4 位地址)

图 3-31 [HL+B]、 [HL+C] 的例子

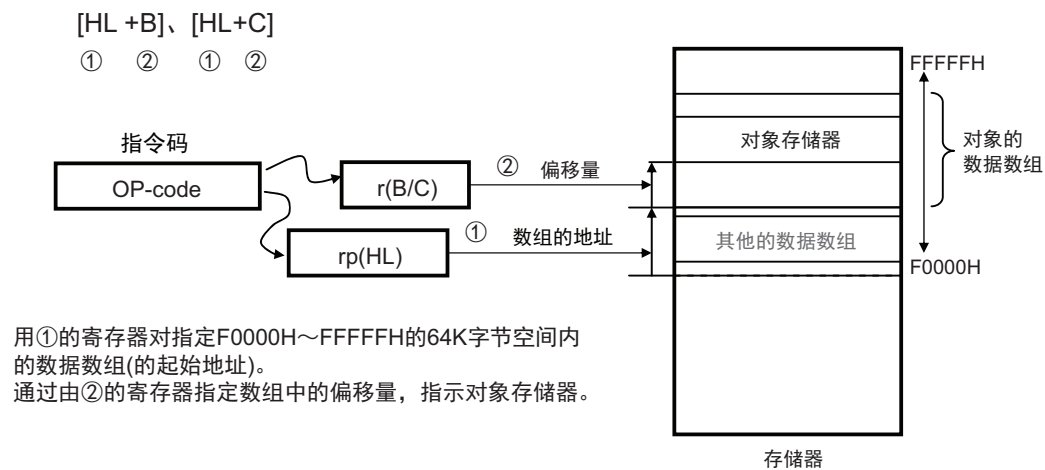
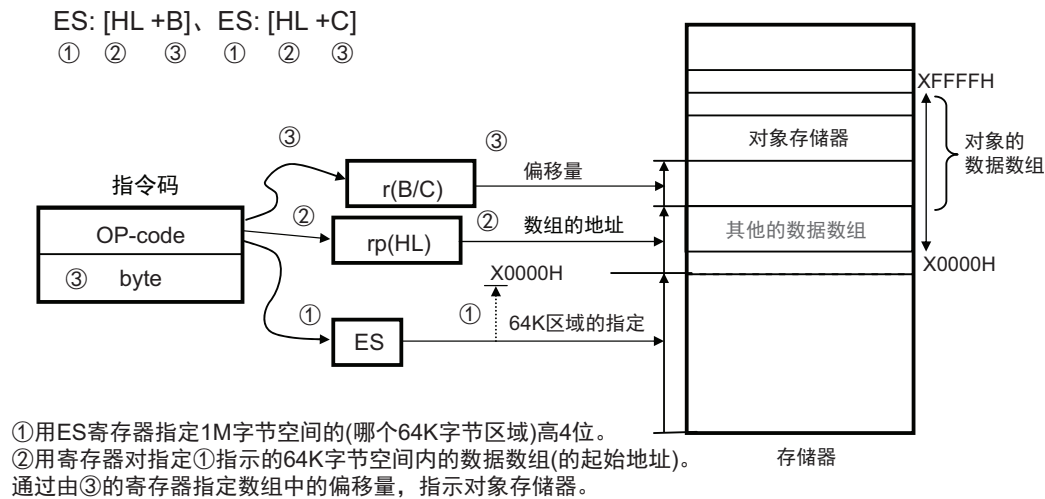


图 3-32 ES:[HL+B]、 ES:[HL+C] 的例子



3.4.9 堆栈寻址

[功能]

堆栈寻址是通过堆栈指针（SP）的值间接指定堆栈区域的寻址方式。当执行 PUSH、POP、子程序调用和返回指令时，或者在因产生中断请求而保存 / 恢复寄存器时，自动使用这种寻址方式。
只能将堆栈区设定在 RAM 区域。

[操作数形式]

表现形式	记述方法
—	PUSH PSW AX/BC/DE/HL POP PSW AX/BC/DE/HL CALL/CALLT RET BRK RETB (产生中断请求) RETI

根据各堆栈的运行，压栈和退栈的数据如图 3-33 ~图 3-38 所示。

图 3-33 PUSH rp 的例子

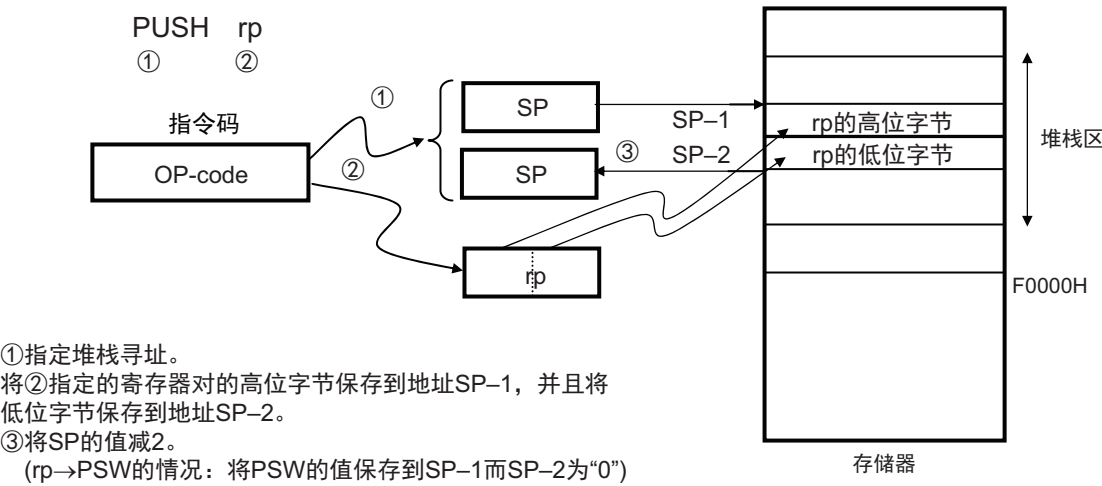


图 3-34 POP 的例子

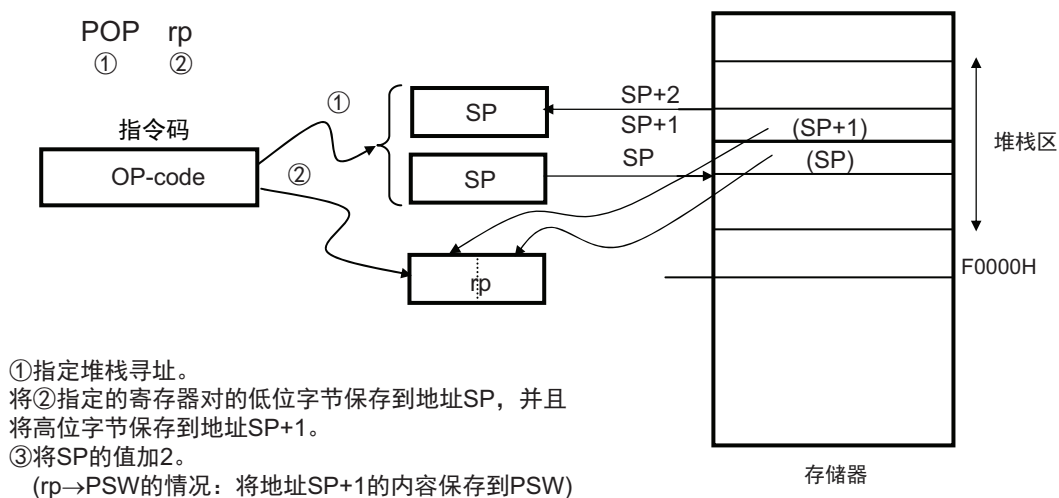


图 3-35 CALL、CALLT 的例子

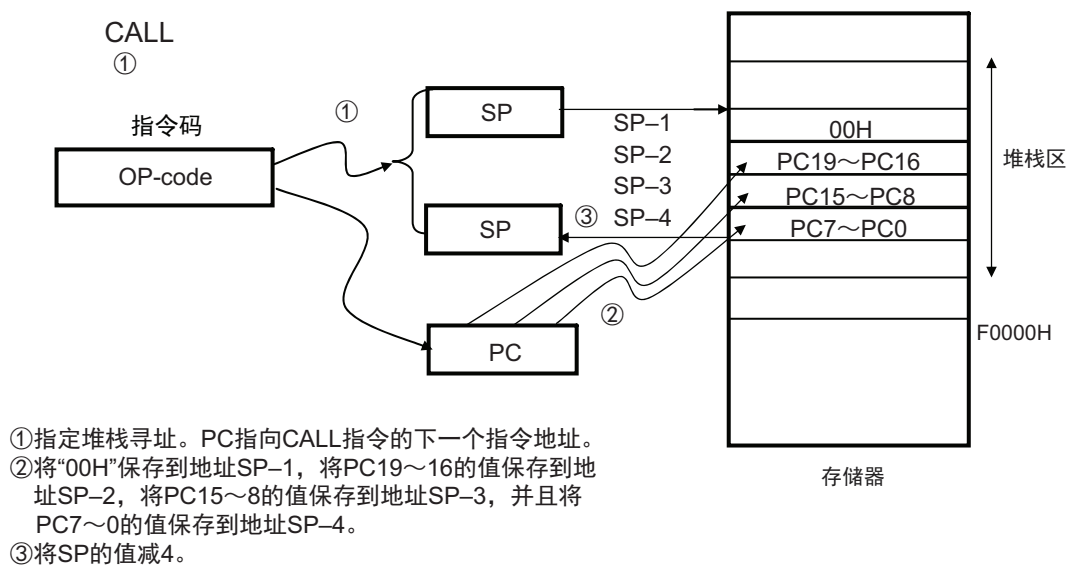


图 3-36 RET 的例子

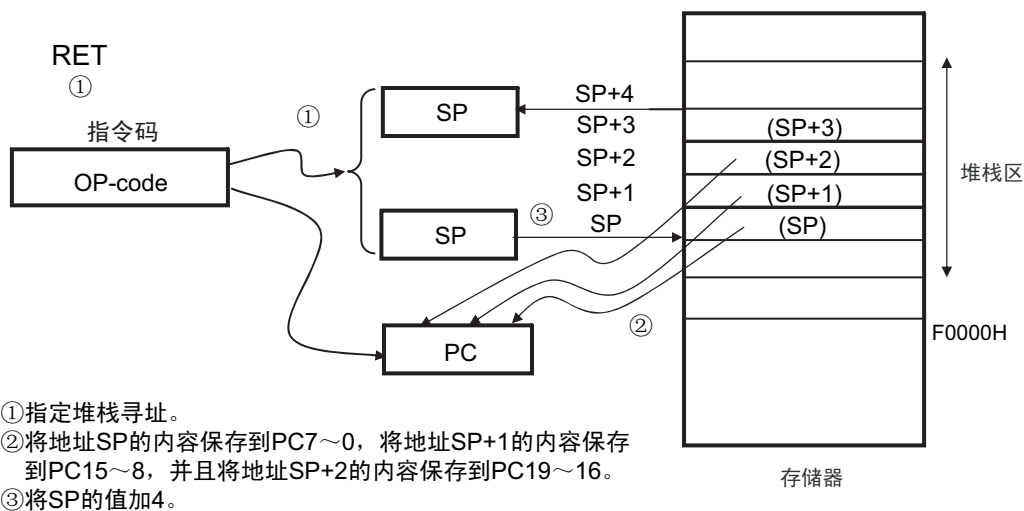


图 3-37 中断、BRK 的例子

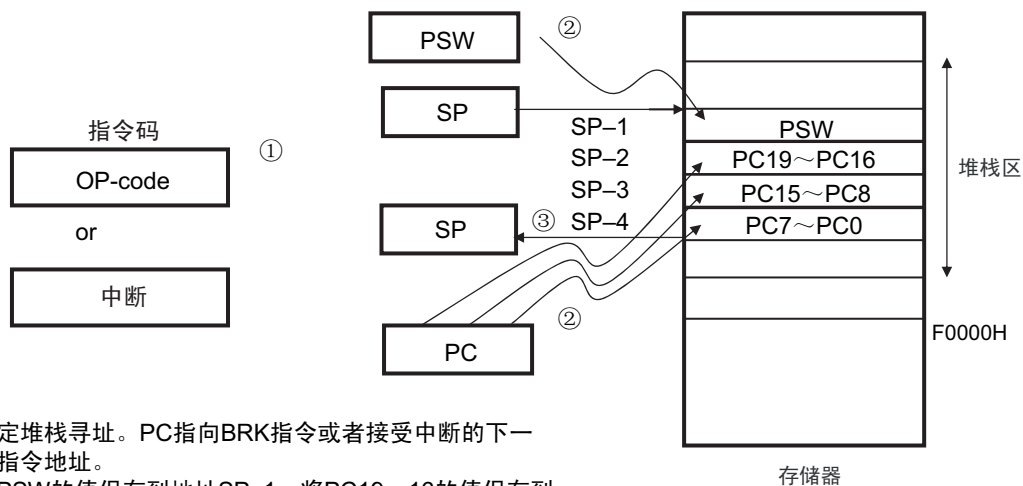
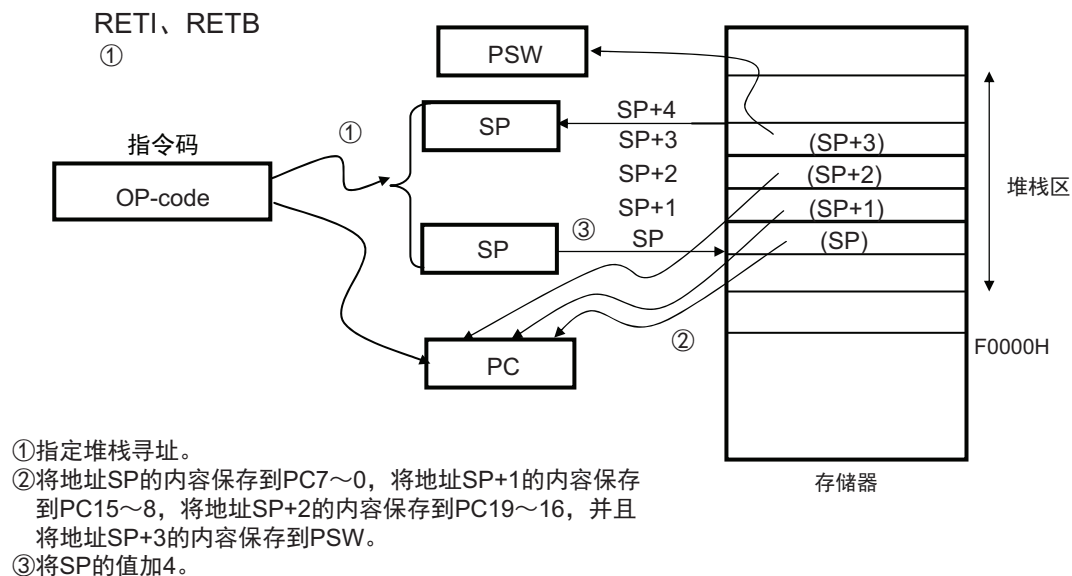


图 3-38 RETI、RETB 的例子



第4章 端口功能

4.1 端口功能

R7F0C003 和 R7F0C004 提供数字输入 / 输出端口，能进行各种各样的控制。

除了作为数字输入 / 输出端口的功能以外，还提供各种复用功能。有关复用功能，请参照“第2章 引脚功能”。

4.2 端口结构

端口由以下硬件构成。

表 4-1 端口结构

项目	结构
控制寄存器	端口模式寄存器 (PM0 ~ PM7、PM12) 端口寄存器 (P0 ~ P7、P12、P13) 上拉电阻选择寄存器 (PU0 ~ PU5、PU7、PU12、PU13) 端口输入模式寄存器 (PIM0、PIM1、PIM3 ~ PIM5) 端口输出模式寄存器 (POM0、POM1、POM3 ~ POM5、POM13) 端口模式控制寄存器 (PMC1、PMC2、PMC4) A/D 端口配置寄存器 (ADPC) 外围 I/O 重定向寄存器 (PIOR) LCD 端口功能寄存器 (PFSEG0 ~ PFSEG6) LCD 输入切换控制寄存器 (ISCLCD)
端口	共计：65 个 (CMOS 输入 / 输出：58 个，CMOS 输入：5 个，N 沟道漏极开路输入 / 输出：2 个)

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 位为单位指定为输入模式或者输出模式。当 P00 ~ P07 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 位为单位使用内部上拉电阻。

P03、P05 和 P06 引脚的输入可通过设定端口输入模式寄存器 0 (PIM0) 以 1 位为单位指定为通常的输入缓冲器或者 TTL 输入缓冲器。

P00 和 P04 ~ P07 引脚的输出可通过端口输出模式寄存器 0 (POM0) 以 1 位为单位设定为 N 沟道漏极开路输出 (V_{DD} 耐压)。

将全部引脚用作数字输入 / 输出端口时，可通过 LCD 端口功能寄存器 5、6 (PFSEG5、PFSEG6) 设定为“用作端口 (段输出除外)” (以 1 位单位设定)。

端口 0 作为复用功能，有编程 UART 的发送、LCD 控制器 / 驱动器的段输出、串行接口的数据输入 / 输出、时钟输入 / 输出、定时器的输入 / 输出、时钟 / 蜂鸣器输出、外部中断请求输入以及比较器输出。

产生复位信号后，端口 0 为数字输入无效注。

注 数字输入无效是指数字输出、数字输入、模拟输入以及 LCD 输出均处于无效的状态。

表 4-2 使用端口 0 时的寄存器设定

名称	输入 / 输出	PM0x	PIM0x	POM0x	PFSEGxx	复用功能设定 注 6	备注
P00	输入	1	—	x	0	x	
	输出	0		0	0	SO00/TxD0 输出 = 1 注 1	CMOS 输出
		0		1			N 沟道漏极开路输出
P01	输入	1	—	—	0	x	
	输出	0			0	PCLBUZ1 输出 = 0 注 2 (TO05 输出 = 0 注 3)	
P02	输入	1	—	—	0	x	
	输出	0			0	PCLBUZ0 输出 = 0 注 2	
P03	输入	1	0	—	0	x	CMOS 输入
		1	1			x	TTL 输入
	输出	0	x		0	VCOUT0 输出 = 0 注 4	
P04	输入	1	—	x	0	x	
	输出	0		0	0	TxD2 输出 = 1 注 1	CMOS 输出
		0		1		VCOUT1 输出 = 0 注 4	N 沟道漏极开路输出
P05	输入	1	0	x	0	x	CMOS 输入
		1	1	x		x	TTL 输入
	输出	0	x	0	0	SCL10 输出 = 1 注 1	CMOS 输出
		0	x	1			N 沟道漏极开路输出
P06	输入	1	0	x	0	x	CMOS 输入
		1	1	x		x	TTL 输入
	输出	0	x	0	0	SDA10 输出 = 1 注 1	CMOS 输出
		0	x	1			N 沟道漏极开路输出
P07	输入	1	—	x	0	x	
	输出	0		0	0	TxD1 输出 = 1 注 1	CMOS 输出
		0		1		(PCLBUZ0 输出 = 0 注 5)	N 沟道漏极开路输出

- 注 1. 将 P00/SEG43/SO00/TxD0/TOOLTxD、P04/TxD2/SEG47/VCOUT1、P05/SCL10/SEG48、P06/RxD1/SDA10/SEG49 或者 P07/TxD1/(PCLBUZ0)/SEG50 用作通用端口时，必须将串行通道允许状态寄存器 0 (SE0) 的 bit0 和 bit2 (SE00、SE02)、串行输出寄存器 0 (SO0) 的 bit0 和 bit2 (SO00、SO02) 以及串行输出允许寄存器 0 (SOE0) 的 bit0 和 bit2 (SOE00、SOE02) 设定为与其初始状态相同的值。
2. P01/(TI05)/(TO05)/(INTP5)/PCLBUZ1/SEG44、P02/INTP7/PCLBUZ0/SEG45 用作通用端口时，必须将时钟输出选择寄存器 0、1 (CKS0、CKS1) 的 bit7 (PCLOE0、PCLOE1) 设定为与其初始状态相同的值“0”。
3. 如果在设定 PIOR0=1 时 P01 用作通用端口，必须将定时器输出寄存器 0 (TO0) 的 bit5 (TO05) 和定时器输出允许寄存器 0 (TOE0) 的 bit5 (TOE05) 设定为与其初始状态相同的值“0”。
4. 当将复用比较器输出功能的引脚用作通用端口时，必须将比较器输出控制寄存器 (COMPOCR) 的 CnOE 位设定为与其初始状态相同的值。(n = 0、1)
5. 如果在设定 PIOR3=1 时 P07 用作通用端口，必须将时钟输出选择寄存器 0 (CKS0) 的 bit7 (PCLOE0) 设定为与其初始状态相同的值“0”。
6. () 中是 PIORx = 1 时的情况。

备注 x: 忽略

PM0x: 端口模式寄存器 0

PIM0x: 端口输入模式寄存器 0

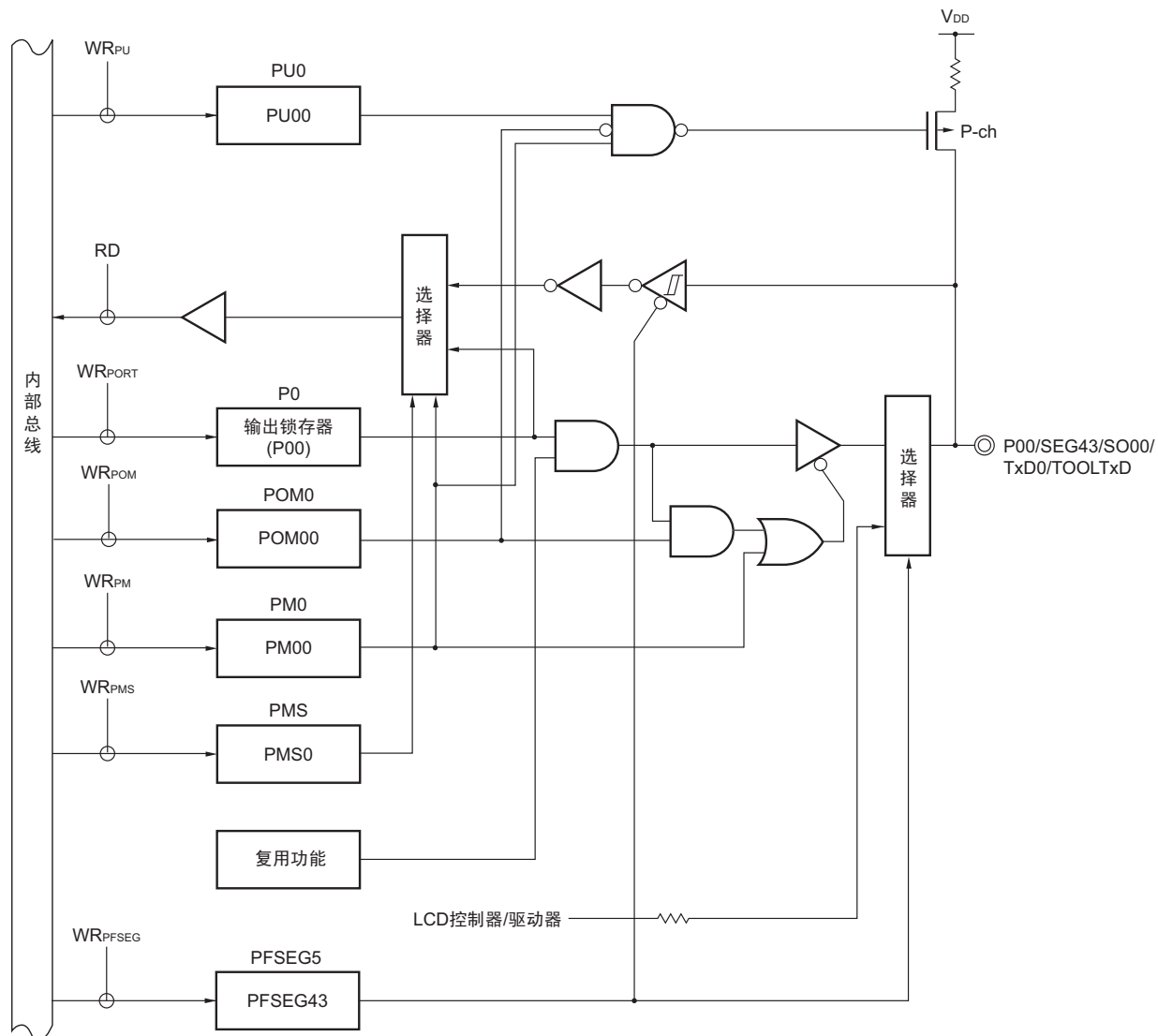
POM0x: 端口输出模式寄存器 0

PFSEGxx: LCD 端口功能寄存器

PIORx: 外围 I/O 重定向寄存器

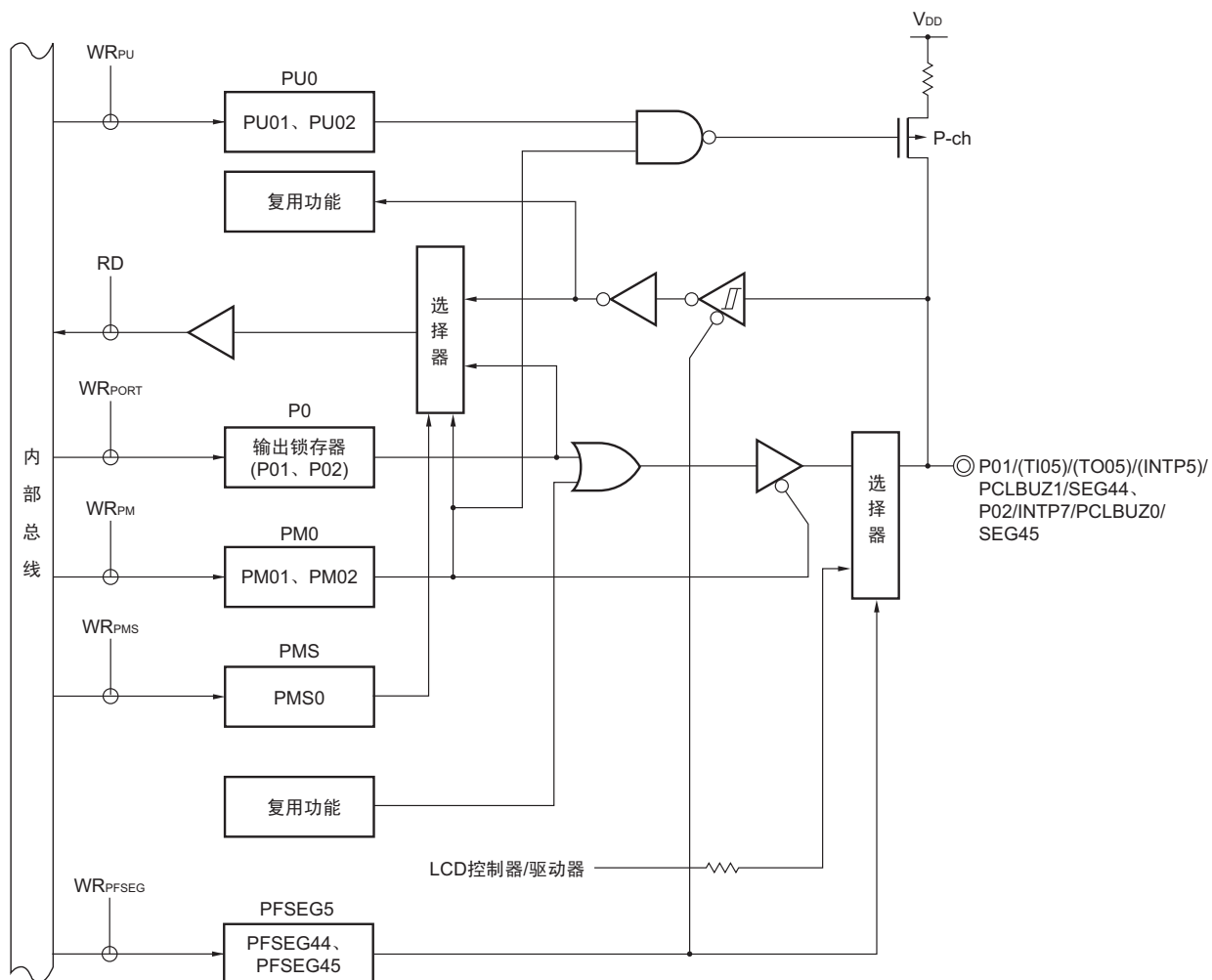
端口0的框图如图4-1～图4-5所示。

图4-1 P00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
 PMS: 端口模式选择寄存器
 PFSEG5: LCD 端口功能寄存器 5
 RD: 读信号
 WRxx: 写信号

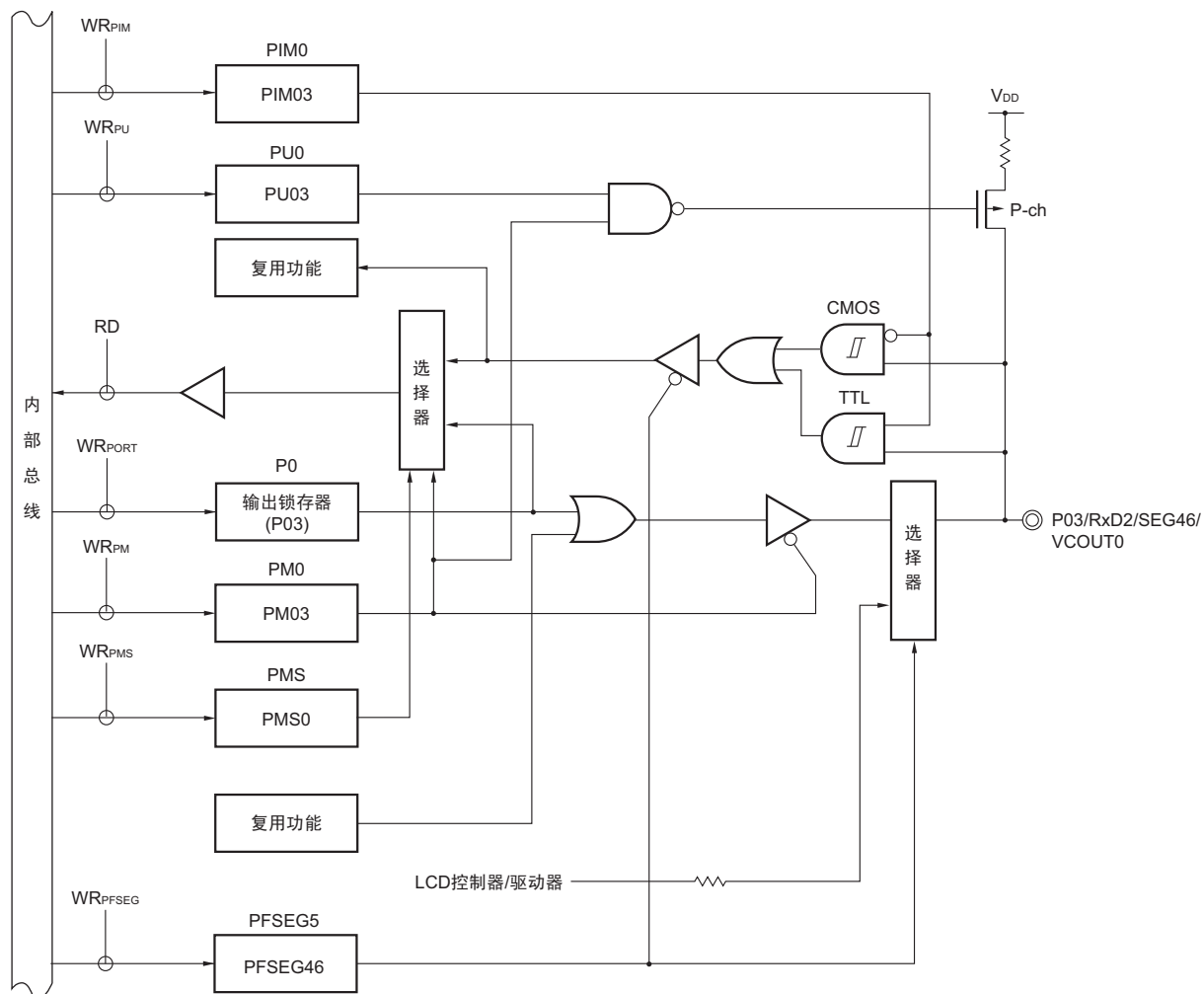
图 4-2 P01 和 P02 的框图



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
 PMS: 端口模式选择寄存器
 PFSEG5: LCD 端口功能寄存器 5
 RD: 读信号
 WRxx: 写信号

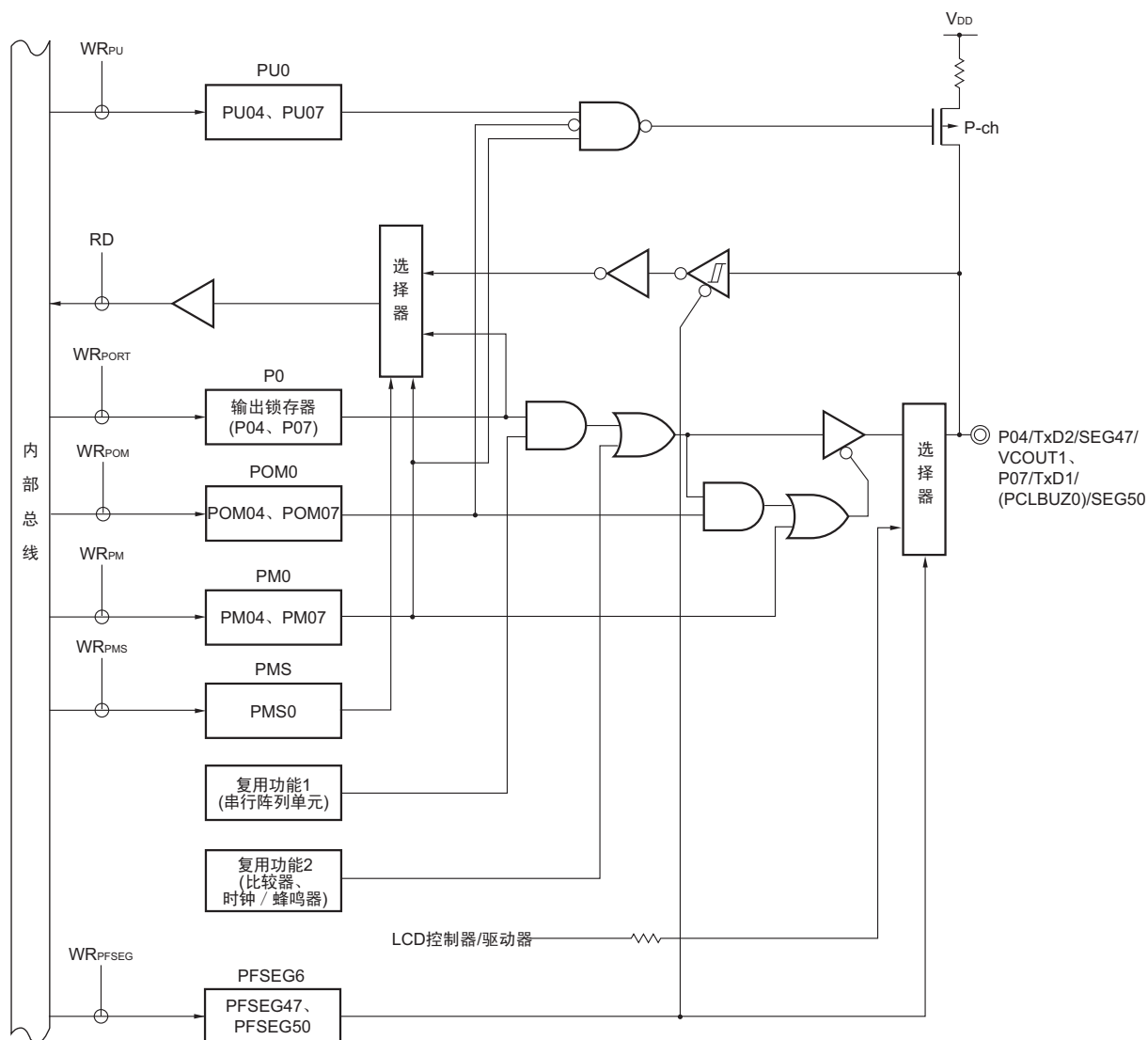
备注 可以通过设定外围 I/O 重定向寄存器 (PIOR) 来指定上图括号内的功能。请参照“图 4-42 外围 I/O 重定向寄存器 (PIOR) 的格式”。

图 4-3 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
 PMS: 端口模式选择寄存器
 PFSEG5: LCD 端口功能寄存器 5
 RD: 读信号
 WRxx: 写信号

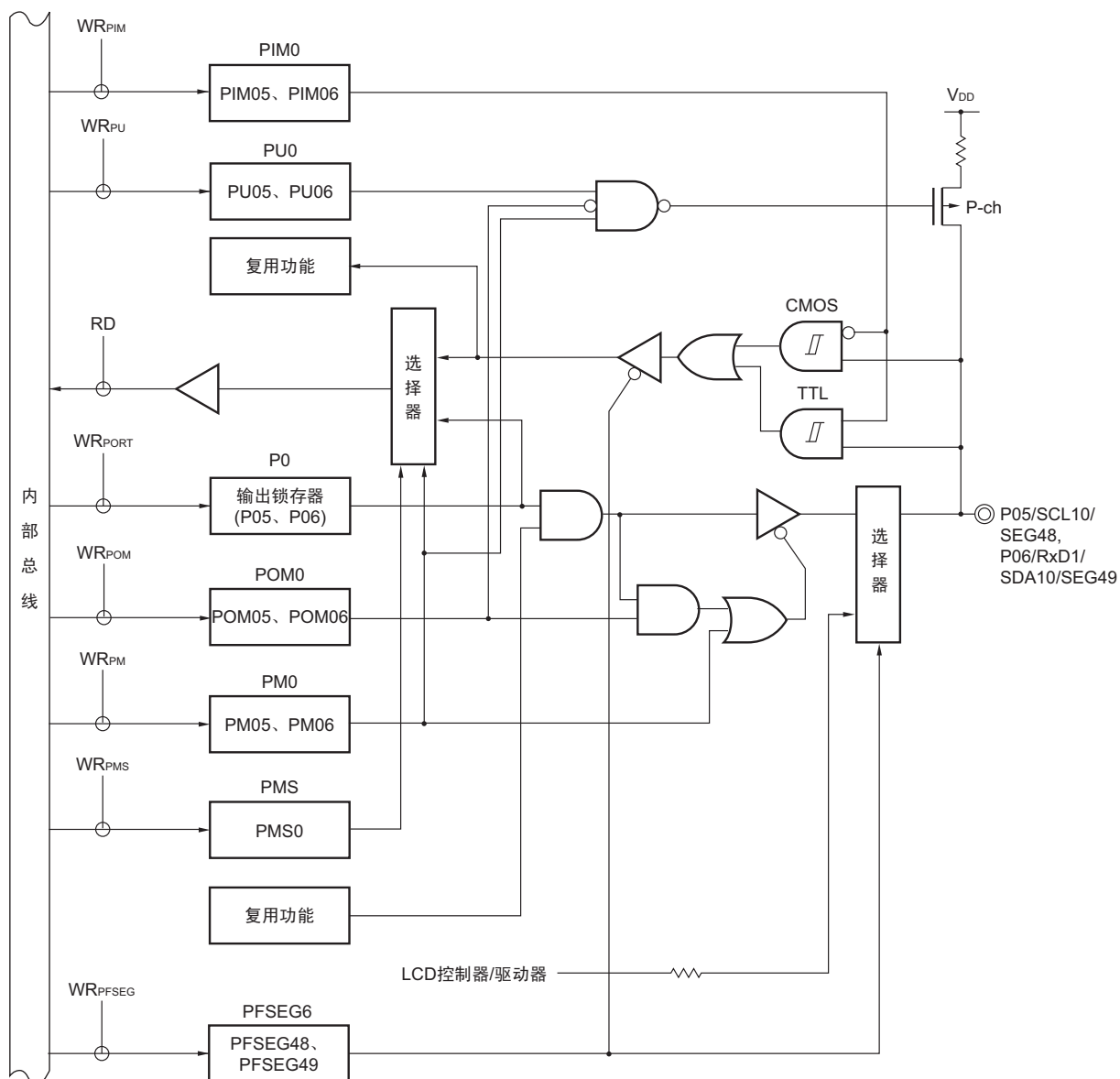
图 4-4 P04 和 P07 的框图



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
 PMS: 端口模式选择寄存器
 PFSEG6: LCD 端口功能寄存器 6
 RD: 读信号
 WRxx: 写信号

备注 可以通过设定外围 I/O 重定向寄存器 (PIOR) 来指定上图括号内的功能。请参照“图 4-42 外围 I/O 重定向寄存器 (PIOR) 的格式”。

图 4-5 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
 PIM0: 端口输入模式寄存器 0
 POM0: 端口输出模式寄存器 0
 PMS: 端口模式选择寄存器
 PFSEG6: LCD 端口功能寄存器 6
 RD: 读信号
 WRxx: 写信号

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 位为单位指定为输入模式或者输出模式。当 P10 ~ P17 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 位为单位使用内部上拉电阻。

P16 和 P17 引脚的输入可通过设定端口输入模式寄存器 1 (PIM1) 以 1 位为单位指定为通常的输入缓冲器或者 TTL 输入缓冲器。

P16 和 P17 引脚的输出可通过端口输出模式寄存器 1 (POM1) 以 1 位为单位设定为 N 沟道漏极开路输出 (V_{DD} 耐压)。

将 P10 ~ P13 引脚用作数字输入 / 输出端口时, 可通过端口模式控制寄存器 1 (PMC1) 设定为“数字输入 / 输出”, 并且可通过 LCD 端口功能寄存器 4 (PFSEG4) 设定为“用作端口 (段输出除外)” (以 1 位单位设定)。

将 P14 ~ P17 引脚用作数字输入 / 输出端口时, 可通过 LCD 端口功能寄存器 5 (PFSEG5) 设定为“用作端口 (段输出除外)” (以 1 位单位设定)。

端口 1 作为复用功能, 有 LCD 控制器 / 驱动器的段输出、串行接口的数据输入 / 输出、时钟输入 / 输出、定时器的输入 / 输出和编程 UART 的接收。

产生复位信号后, P10 ~ P17 引脚为数字输入无效注。

注 数字输入无效是指数字输出、数字输入、模拟输入以及 LCD 输出均处于无效的状态。

表 4-3 使用端口 1 时的寄存器设定

名称	输入 / 输出	PM1x	PIM1x	POM1x	PMC1x	PFSEGxx	复用功能设定	备注
P10	输入	1	—	—	0	0	×	
	输出	0			0	0	×	
P11	输入	1	—	—	0	0	×	
	输出	0			0	0	×	
P12	输入	1	—	—	0	0	×	
	输出	0			0	0	×	
P13	输入	1	—	—	0	0	×	
	输出	0			0	0	×	
P14	输入	1	—	—	—	0	×	
	输出	0				0	TO04 输出 = 0 注 1	
P15	输入	1	—	—	—	0	×	
	输出	0				0	TO07 输出 = 0 注 1	
P16	输入	1	0	×	—	0	×	CMOS 输入
		1	1	×			×	TTL 输入
	输出	0	×	0		0	SCK00/SCL00 输出 = 1 注 2	CMOS 输出
		0	×	1				N 沟道漏极 开路输出
P17	输入	1	0	×	—	0	×	CMOS 输入
		1	1	×			×	TTL 输入
	输出	0	×	0		0	SDA00 输出 = 1 注 2	CMOS 输出
		0	×	1				N 沟道漏极 开路输出

注 1. 将 P14/TI04/TO04/SEG39、P15/TI07/TO07/SEG40 用作通用端口时，必须将定时器输出寄存器 0（TO0）的 bit4 和 bit7（TO04、TO07）、定时器输出允许寄存器 0（TOE0）的 bit4 和 bit7（TOE04、TOE07）设定为与其初始状态相同的值“0”。

2. 将 P16/SEG41/SCK00/SCL00、P17/SEG42/SI00/RxD0/TOOLRxD/SDA00 用作通用端口时，必须将串行通道允许状态寄存器 0（SE0）的 bit0（SE00）、串行输出寄存器 0（SO0）的 bit0（SO00）和串行输出允许寄存器 0（SOE0）的 bit0（SOE00）设定为与其初始状态相同的值。

备注 ×： 忽略

PM1x： 端口模式寄存器 1

PIM1x： 端口输入模式寄存器 1

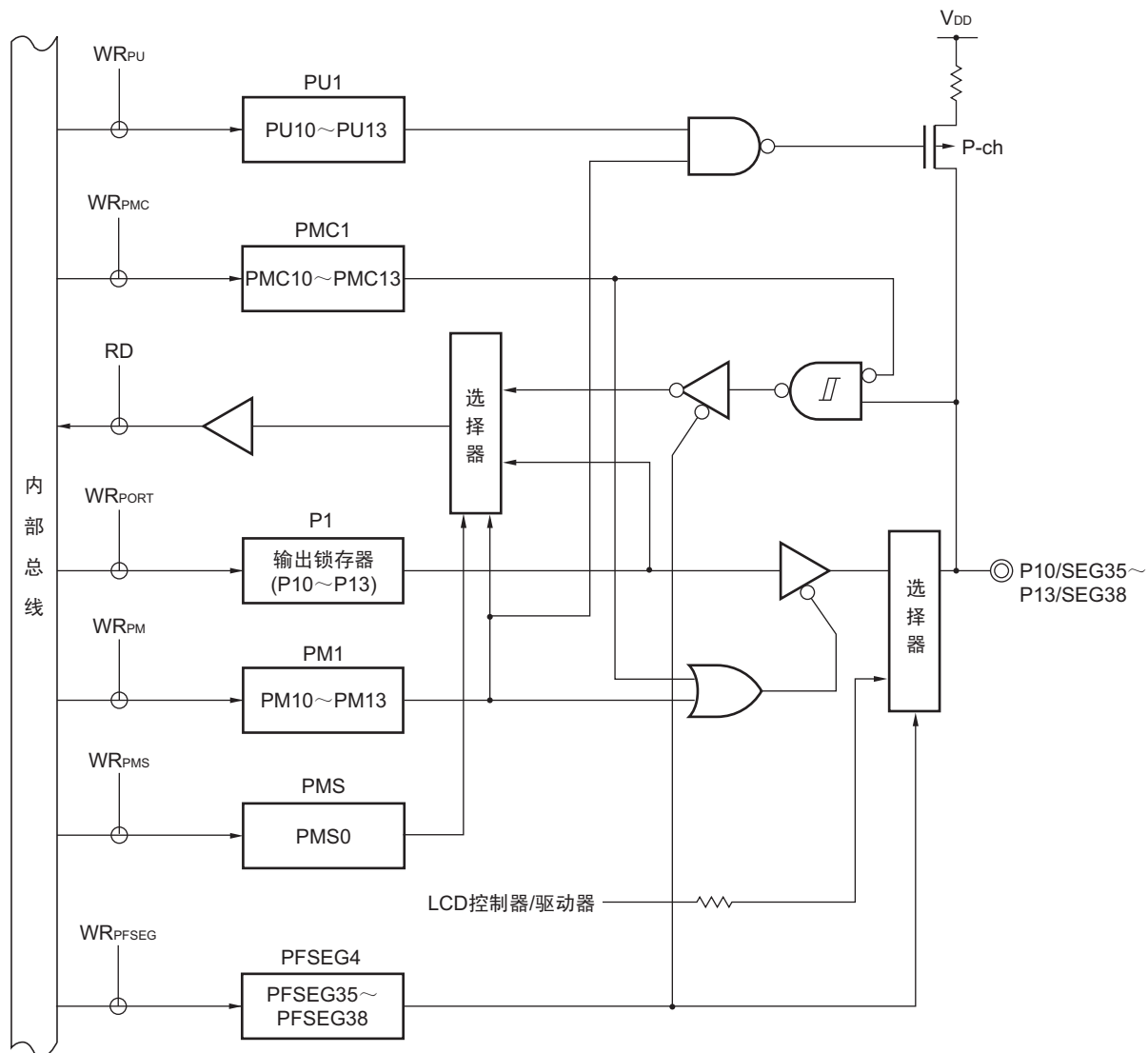
POM1x： 端口输出模式寄存器 1

PMC1x： 端口模式控制寄存器 1

PFSEGxx： LCD 端口功能寄存器

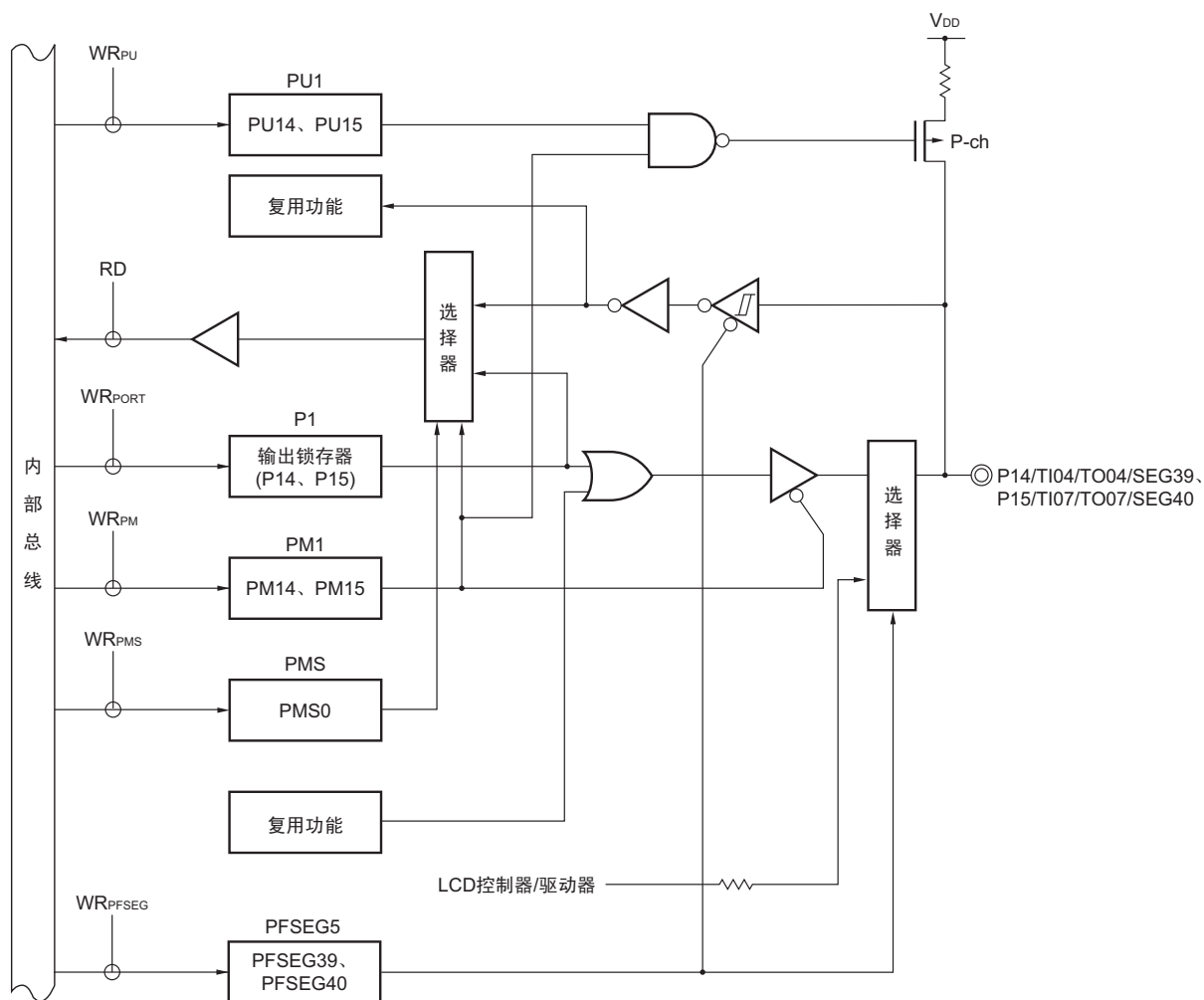
端口 1 的框图如图 4-6 ~ 图 4-8 所示。

图 4-6 P10 ~ P13 的框图



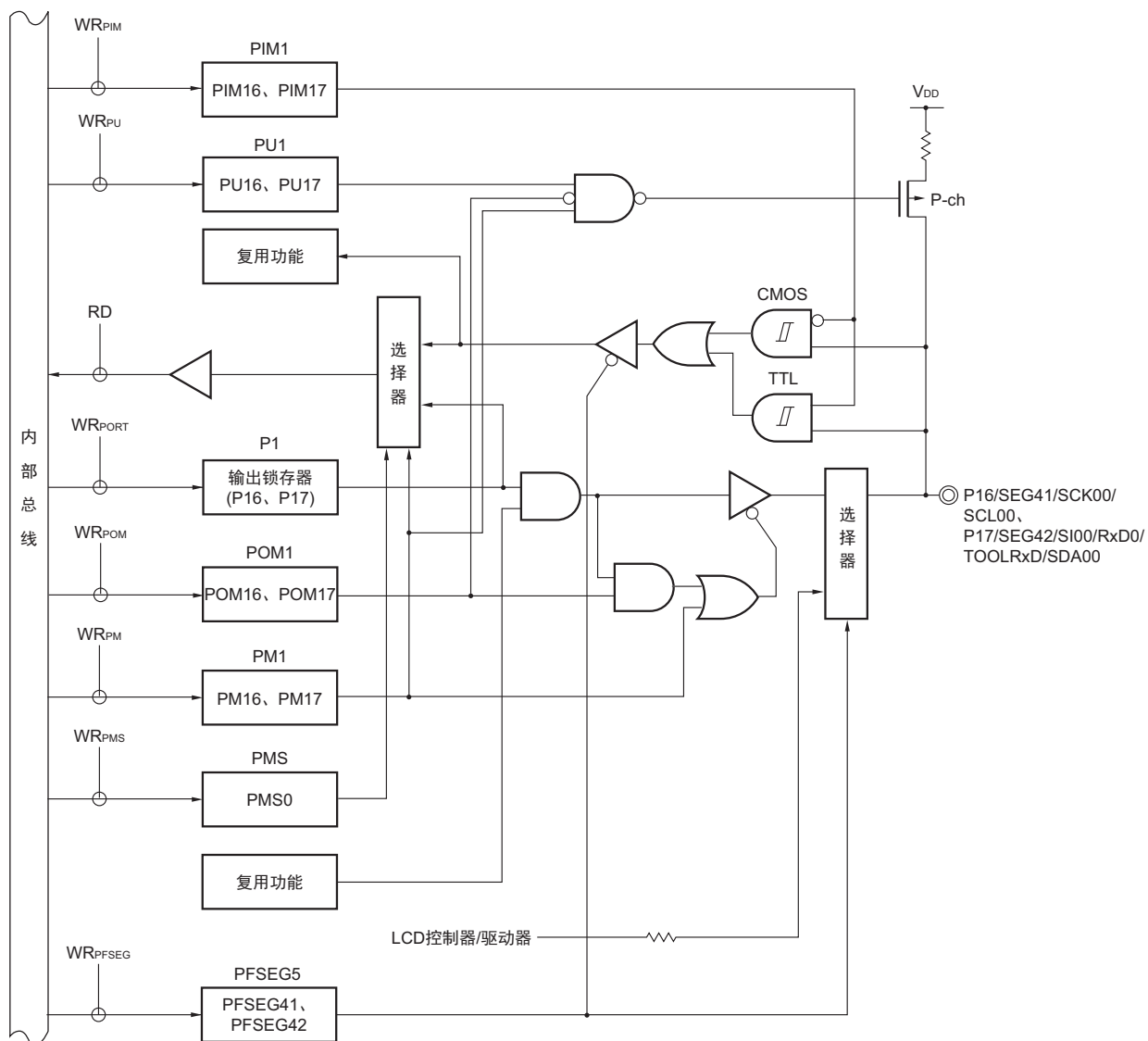
P1:	端口寄存器 1
PU1:	上拉电阻选择寄存器 1
PM1:	端口模式寄存器 1
PMC1:	端口模式控制寄存器 1
PMS:	端口模式选择寄存器
PFSEG4:	LCD 端口功能寄存器 4
RD:	读信号
WRxx:	写信号

图 4-7 P14 和 P15 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
 PMS: 端口模式选择寄存器
 PFSEG5: LCD 端口功能寄存器 5
 RD: 读信号
 WRxx: 写信号

图 4-8 P16 和 P17 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
PMS:	端口模式选择寄存器
PFSEG5:	LCD 端口功能寄存器 5
RD:	读信号
WRxx:	写信号

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 位为单位指定为输入模式或者输出模式。当 P22 ~ P27 引脚用作输入端口时，可通过上拉电阻选择寄存器 2（PU2）以 1 位为单位使用内部上拉电阻。

端口 2 作为复用功能，有 A/D 转换器模拟输入、(+) A/D 转换器的基准电压输入、(-) A/D 转换器的基准电压输入以及 LCD 控制器 / 驱动器的段输出。

将 P20/ANI1 ~ P21/ANI0 用作数字输入时，必须通过 A/D 端口配置寄存器（ADPC）设定为“数字输入 / 输出”后，从上位开始使用。

将 P20/ANI1 和 P21/ANI0 用作模拟输入引脚时，必须通过 A/D 端口配置寄存器（ADPC）设定为模拟输入模式，并且通过 PM2 寄存器设定为输入模式后，从低位开始使用。

表 4-4 使用端口 20 和端口 21 时的寄存器设定

名称	输入 / 输出	PM2x	ADPC	复用功能设定	备注
P2n	输入	1	01 ~ n+1H	—	用作端口时，从高位开始使用。
	输出	0	01 ~ n+1H		

表 4-5 P20/ANI1 和 P21/ANI0 引脚功能的设定

ADPC 寄存器	PM2 寄存器	ADS 寄存器	P20/ANI1 和 P21/ANI0 引脚
选择数字输入 / 输出	输入模式	—	数字输入
	输出模式	—	数字输出
选择模拟输入	输入模式	选择 ANI	模拟输入（转换对象）
		不选择 ANI	模拟输入（非转换对象）
	输出模式	选择 ANI	禁止设定
		不选择 ANI	

将 P22 和 P23 引脚用作输入时，可通过端口模式控制寄存器 2（PMC2）以 1 位为单位设定为数字输入或者模拟输入。

将 P24 ~ P27 引脚用作数字输入 / 输出或者 LCD 输出时，必须将端口模式控制寄存器 2（PMC2）设定为“0”（可以以 1 位单位设定）。

表 4-6 使用端口 22 ~ 端口 27 时的寄存器设定

名称	输入 / 输出	PM2x	PMC2x	PFSEGxx	复用功能设定	备注
P22 ~ P27	输入	1	0	0	×	
	输出	0	0	0	×	

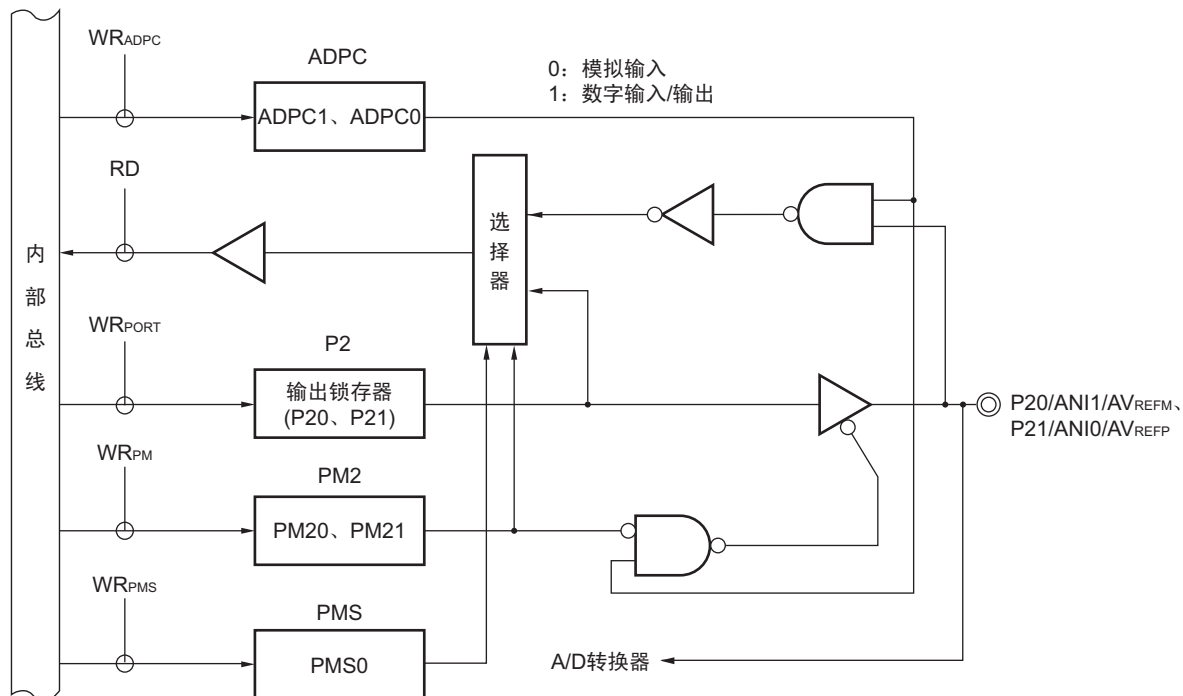
备注 ×: 忽略
 PM2x: 端口模式寄存器 2
 ADPC: A/D 端口配置寄存器
 PMC2x: 端口模式控制寄存器 2
 PFSEGxx: LCD 端口功能寄存器

在产生复位信号后，P20 ~ P23 为模拟输入，P24 ~ P27 为数字输入无效注。

注 数字输入无效是指数字输出、数字输入、模拟输入以及 LCD 输出均处于无效的状态。

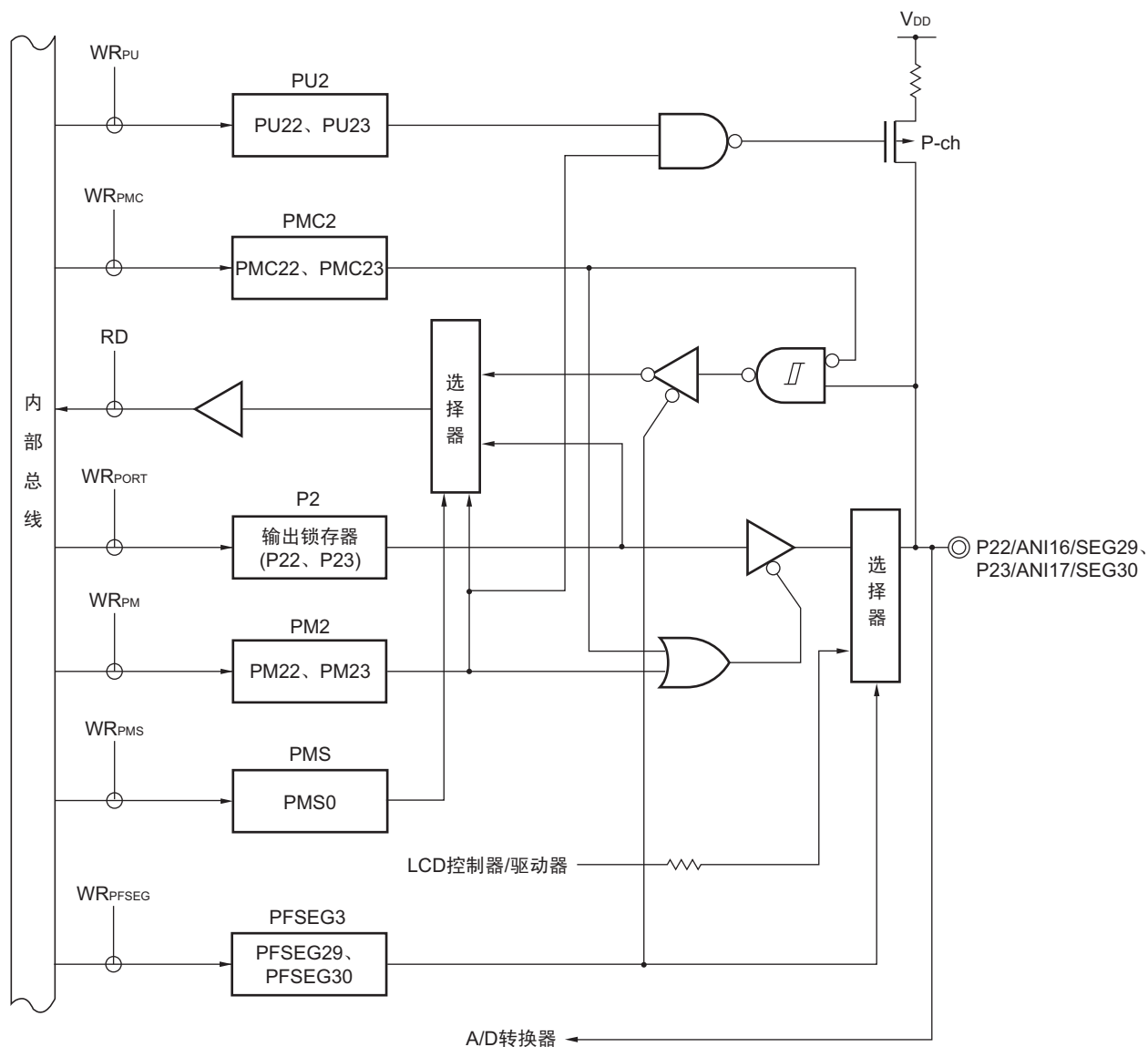
端口 2 的框图如图 4-9 ~ 图 4-11 所示。

图 4-9 P20 和 P21 的框图



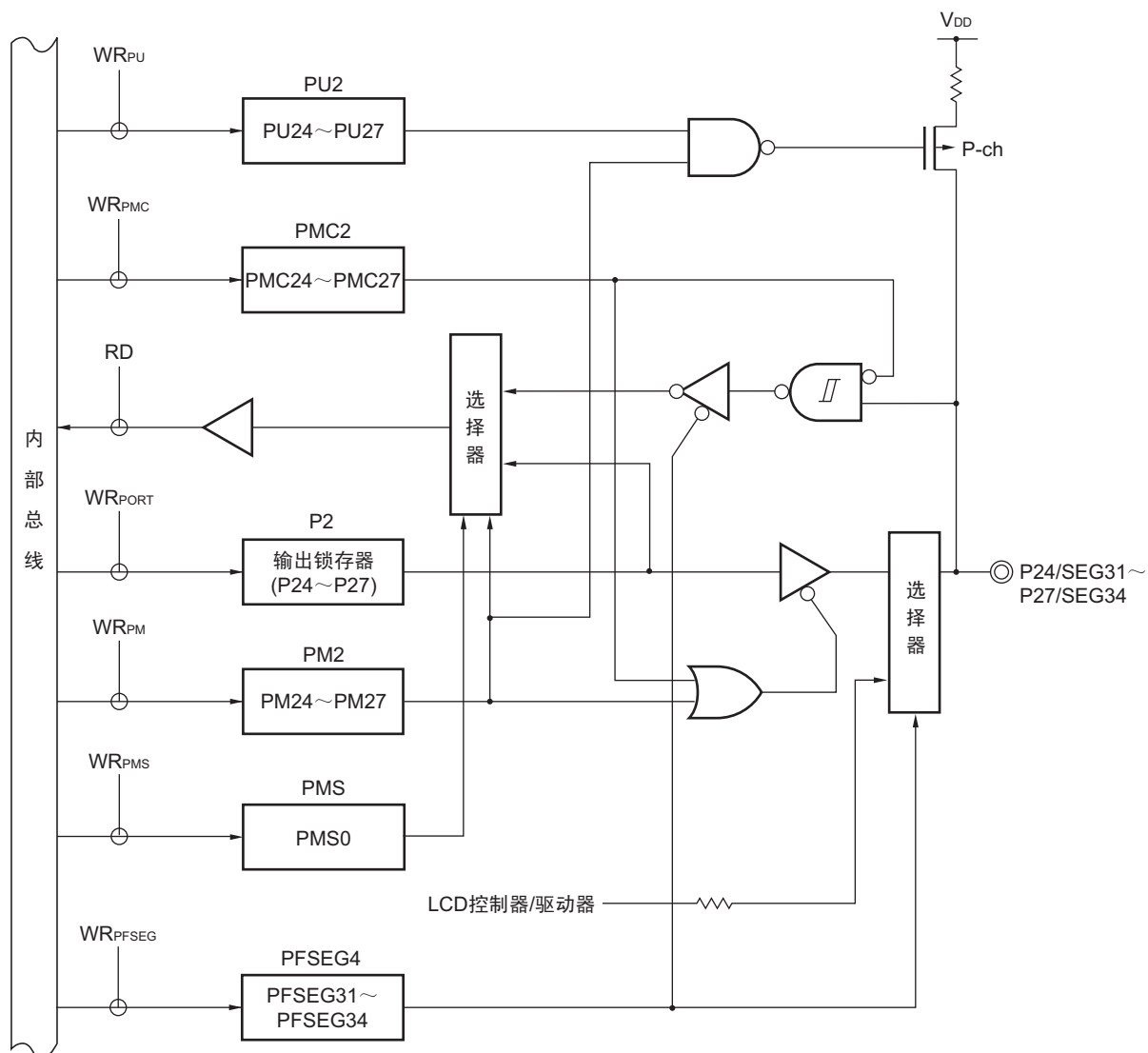
ADPC: A/D 端口配置寄存器
P2: 端口寄存器 2
PM2: 端口模式寄存器 2
PMS: 端口模式选择寄存器
RD: 读信号
WRxx: 写信号

图 4-10 P22 和 P23 的框图



- P2: 端口寄存器 2
 PU2: 上拉电阻选择寄存器 2
 PM2: 端口模式寄存器 2
 PMC2: 端口模式控制寄存器 2
 PMS: 端口模式选择寄存器
 PFSEG3: LCD 端口功能寄存器 3
 RD: 读信号
 WR_{xx}: 写信号

图 4-11 P24 ~ P27 的框图



P2:	端口寄存器 2
PU2:	上拉电阻选择寄存器 2
PM2:	端口模式寄存器 2
PMC2:	端口模式控制寄存器 2
PMS:	端口模式选择寄存器
PFSEG4:	LCD 端口功能寄存器 4
RD:	读信号
WRxx:	写信号

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 位为单位指定为输入模式或输出模式。当 P30 ~ P35 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 位为单位使用内部上拉电阻。

P34 引脚的输入可通过设定端口输入模式寄存器 3（PIM3）指定为通常的输入缓冲器或者 TTL 输入缓冲器。

P35 引脚的输出可通过端口输出模式寄存器 3（POM3）设定为 N 沟道漏极开路输出（V_{DD} 耐压）。

将 P30 ~ P35 引脚用作数字输入 / 输出时，必须通过 LCD 端口功能寄存器 2、3（PFSEG2、PFSEG3）设定为“用作端口（段输出除外）”（可以以 1 位单位设定）。

端口 3 作为复用功能，有外部中断请求输入、实时时钟的校正时钟输出、定时器的输入 / 输出、LCD 控制器 / 驱动器的段输出以及串行接口的数据输入 / 输出。

在产生复位信号后，端口 3 为数字输入无效注。

注 数字输入无效是指数字输出、数字输入、模拟输入以及 LCD 输出均处于无效的状态。

表 4-7 使用端口 3 时的寄存器设定

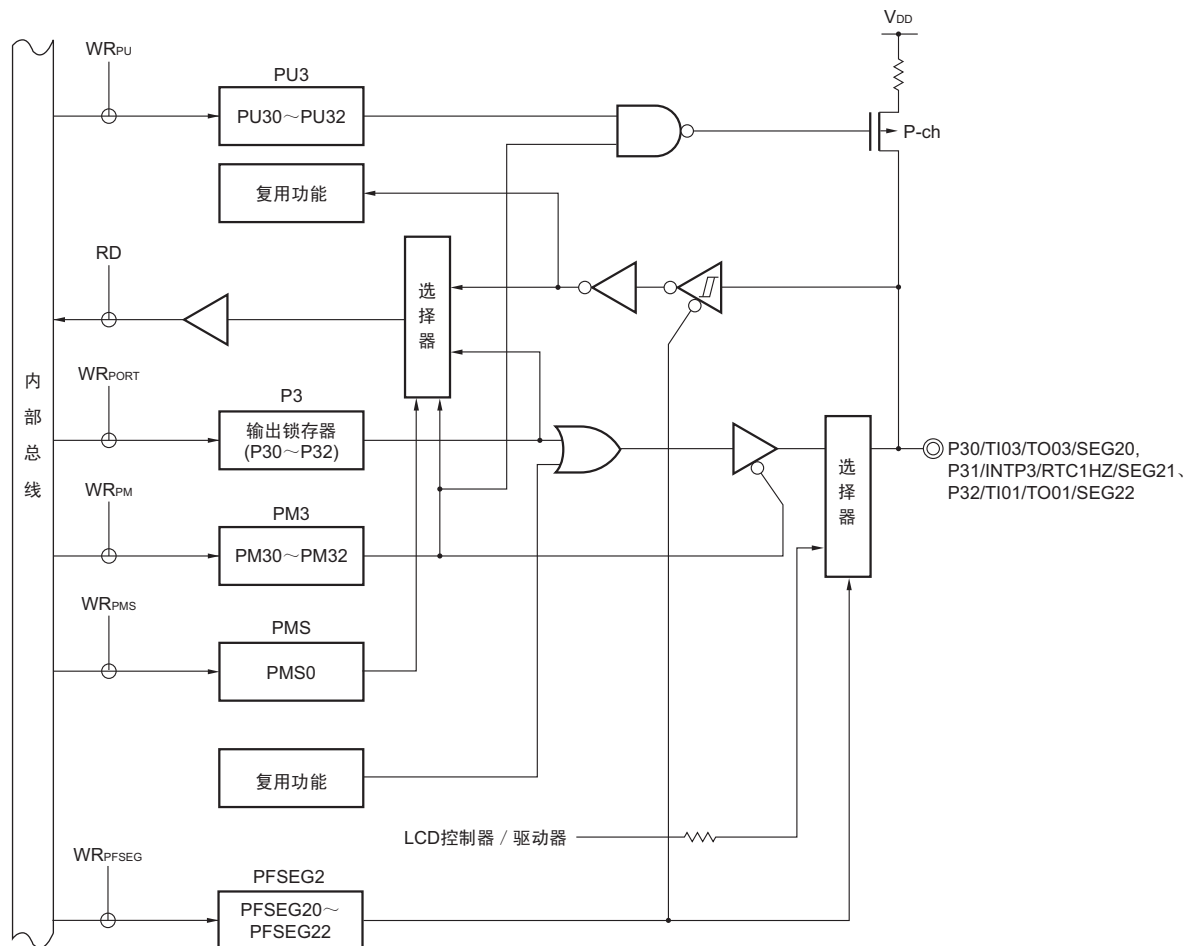
名称	输入 / 输出	PM3x	PIM3x	POM3x	PFSEGxx	复用功能设定	备注
P30	输入	1	—	—	0	×	
	输出	0			0	TO03 输出 = 0 注 1	
P31	输入	1	—	—	0	×	
	输出	0			0	RTC1HZ 输出 = 0 注 2	
P32	输入	1	—	—	0	×	
	输出	0			0	TO01 输出 = 0 注 1	
P33	输入	1	—	—	0	×	
	输出	0			0	×	
P34	输入	1	0	—	0	×	CMOS 输入
		1	1		0	×	TTL 输入
	输出	0	×			×	
P35	输入	1	—	×	0	×	
	输出	0		0	0	TxD3 输出 = 1 注 3	CMOS 输出
		0		1			N 沟道漏极开路输出

- 注 1. 将 P30/TI03/TO03/SEG20、P32/TI01/TO01/SEG22 用作通用端口时，必须将定时器输出寄存器 0（TO0）的 bit1 和 bit3（TO01、TO03）、定时器输出允许寄存器 0（TOE0）的 bit1 和 bit3（TOE01、TOE03）设定为与其初始状态相同的值“0”。
2. 将 P31/INTP3/RTC1HZ/SEG21 用作通用端口时，必须将实时时钟控制寄存器 0（RTCC0）的 bit5（CLOE1）设定为与其初始状态相同的值“0”。
3. 将 P35/TxD3/SEG25 用作通用端口时，必须将串行通道允许状态寄存器 1（SE1）的 bit2（SE12）、串行输出寄存器 1（SO1）的 bit2（SO12）和串行输出允许寄存器 1（SOE1）的 bit2（SOE12）设定为与其初始状态相同的值。

备注 ×： 忽略
 PM3x： 端口模式寄存器 3
 PIM3x： 端口输入模式寄存器 3
 POM3x： 端口输出模式寄存器 3
 PFSEGxx： LCD 端口功能寄存器

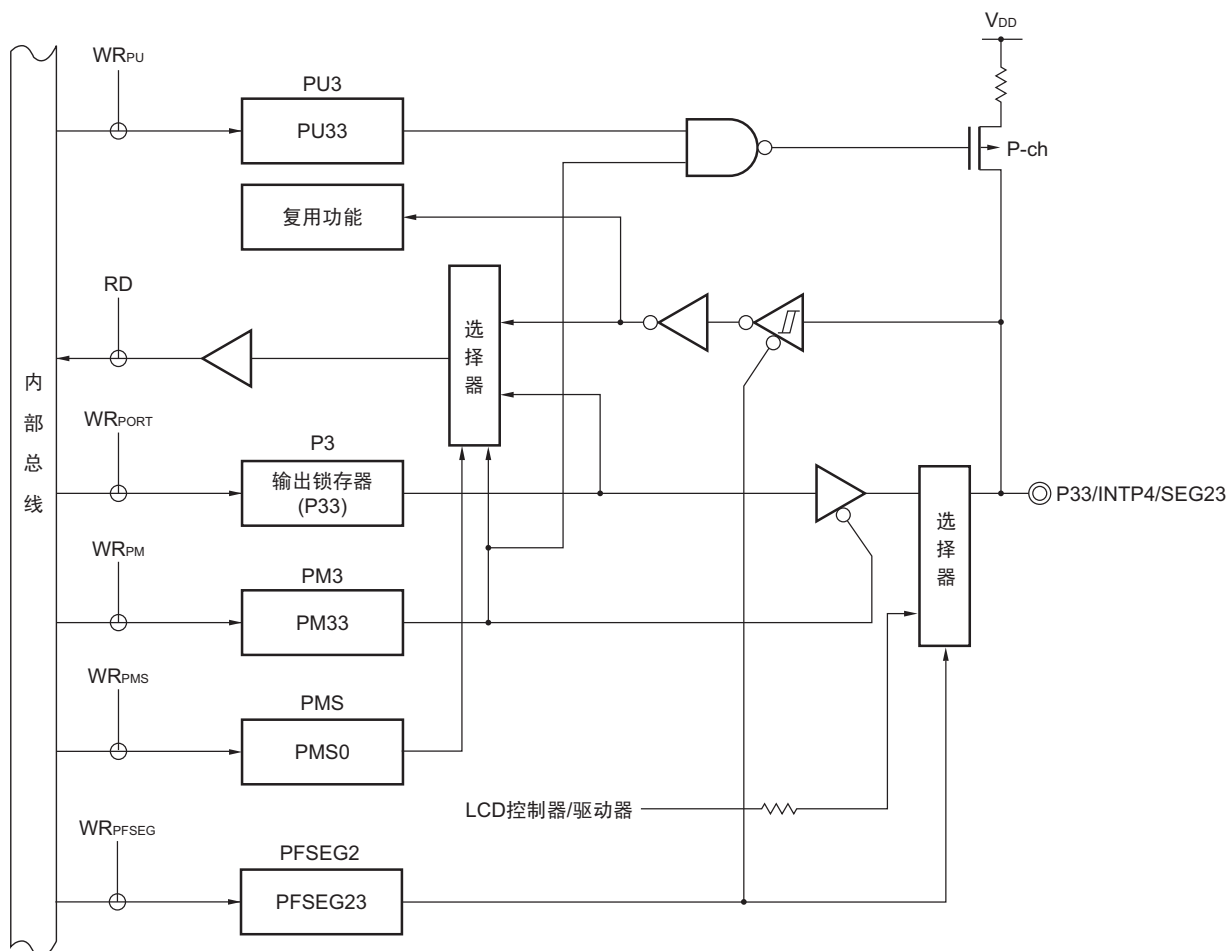
端口 3 的框图如图 4-12 ~ 图 4-15 所示。

图 4-12 P30 ~ P32 的框图



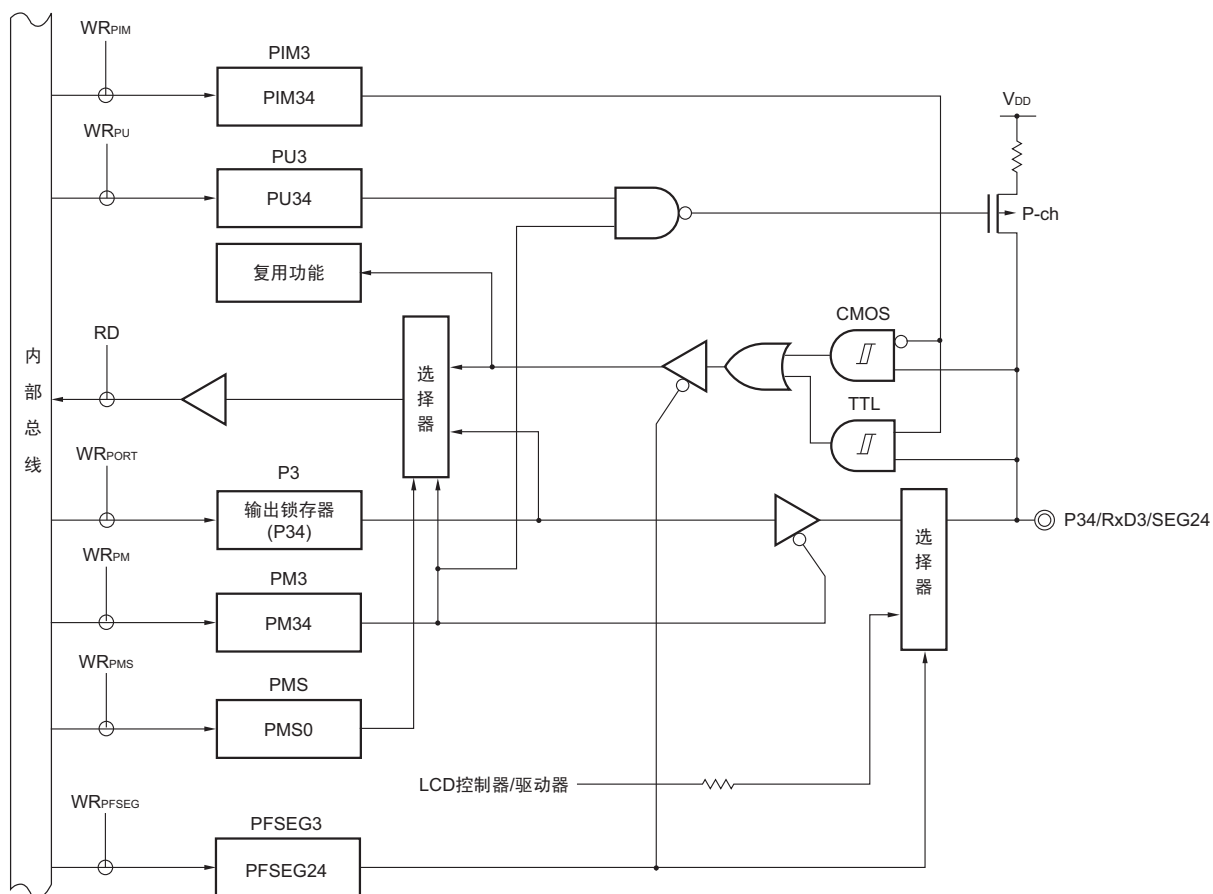
P3:	端口寄存器 3
PU3:	上拉电阻选择寄存器 3
PM3:	端口模式寄存器 3
PMS:	端口模式选择寄存器
PFSEG2:	LCD 端口功能寄存器 2
RD:	读信号
WRxx:	写信号

图 4-13 P33 的框图



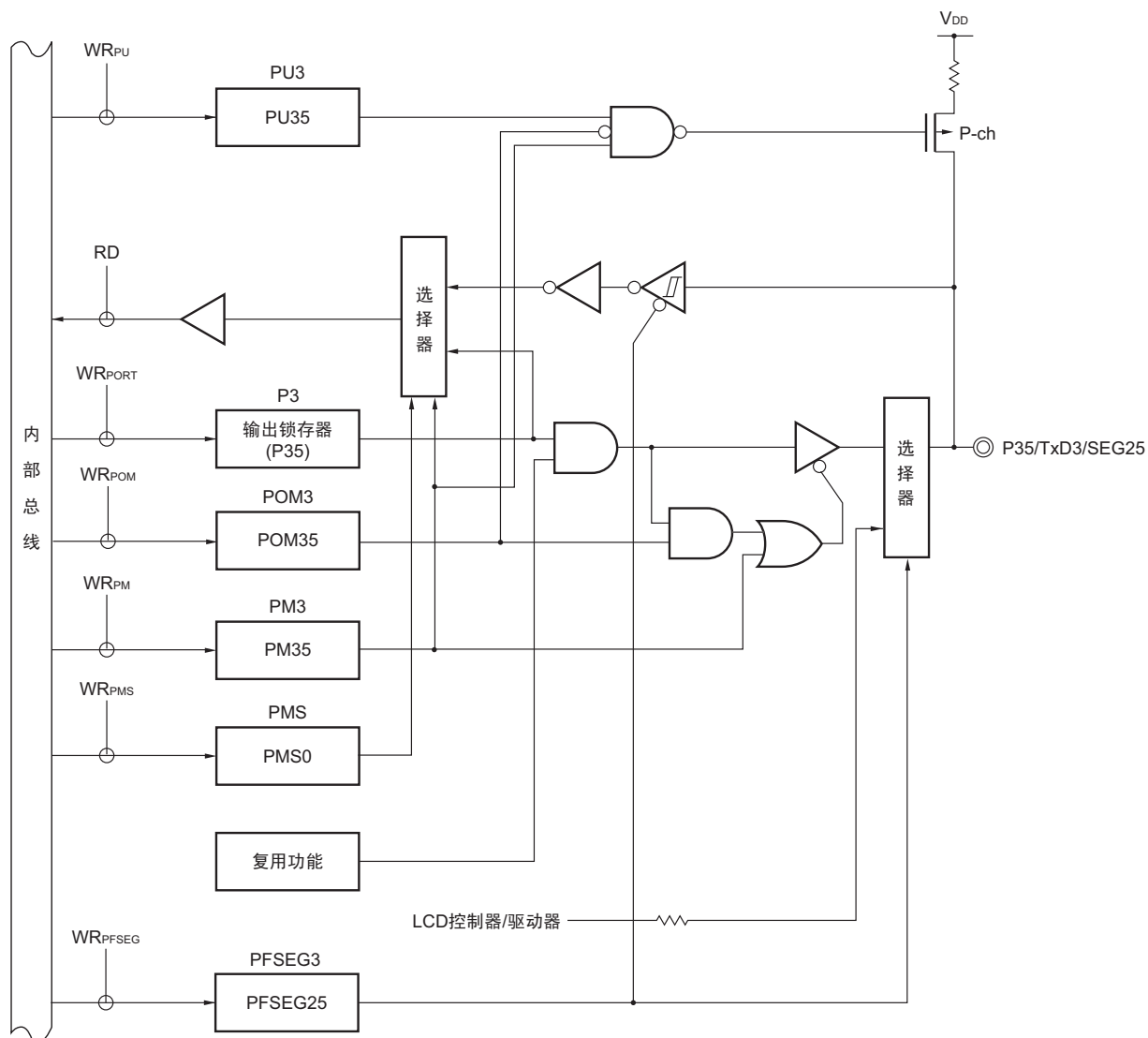
- P3: 端口寄存器 3
 PU3: 上拉电阻选择寄存器 3
 PM3: 端口模式寄存器 3
 PMS: 端口模式选择寄存器
 PFSEG2: LCD 端口功能寄存器 2
 RD: 读信号
 WRxx: 写信号

图 4-14 P34 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
 PIM3: 端口输入模式寄存器 3
 PMS: 端口模式选择寄存器
 PFSEG3: LCD 端口功能寄存器 3
 RD: 读信号
 WRxx: 写信号

图 4-15 P35 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
 POM3: 端口输出模式寄存器 3
 PMS: 端口模式选择寄存器
 PFSEG3: LCD 端口功能寄存器 3
 RD: 读信号
 WRxx: 写信号

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 位为单位指定为输入模式或者输出模式。当 P40 ~ P47 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 位为单位使用内部上拉电阻。

P43、P44、P46 和 P47 引脚的输入可通过设定端口输入模式寄存器 4（PIM4）以 1 位为单位指定为通常的输入缓冲器或者 TTL 输入缓冲器。

P42 ~ P44、P46 和 P47 引脚的输出可通过端口输出模式寄存器 4（POM4）以 1 位为单位设定为 N 沟道漏极开路输出（ V_{DD} 耐压）。

将 P45 引脚用作数字输入 / 输出端口时，必须通过 LCD 端口功能寄存器 3（PFSEG3）的 bit2（PFDEG）设定为“用作端口”。

将 P46、P47 引脚用作数字输入 / 输出端口时，必须通过 LCD 端口功能寄存器 3（PFSEG3）设定为“用作端口（段输出除外）”（可以以 1 位为单位设定）。

端口 4 作为复用功能，有串行接口的数据输入 / 输出、时钟输入 / 输出、闪存编程器 / 调试器的数据输入 / 输出、定时器的输入 / 输出、比较器的基准电压输入、比较器的模拟电压输入和 LCD 控制器 / 驱动器的段输出。

在产生复位信号后，P40 ~ P44 引脚为输入模式，P45 ~ P47 引脚为数字输入无效注。

注 数字输入无效是指数字输出、数字输入、模拟输入以及 LCD 输出均处于无效的状态。

表 4-8 使用端口 4 时的寄存器设定

名称	输入 / 输出	PM4x	PIM4x	POM4x	PMC4x	PFSEGxx、PFDEG	复用功能设定 注 4	备注
P40	输入	1	—	—	—	—	×	
	输出	0					(TO00 输出 = 0 注 1)	
P41	输入	1	—	—	—	—	×	
	输出	0					(TO07 输出 = 0 注 1)	
P42	输入	1	—	×	0	—	×	
	输出	0		0	0		TO05 输出 = 0 注 2	CMOS 输出
		0		1	0		(TxD1 输出 = 1 注 3)	N 沟道漏极开路输出
P43	输入	1	0	×	0	—	×	CMOS 输入
		1	1	×	0		×	TTL 输入
	输出	0	×	0	0		(SDA10 输出 = 1 注 3)	CMOS 输出
		0	×	1	0			N 沟道漏极开路输出
P44	输入	1	0	×	0	—	×	CMOS 输入
		1	1	×	0		×	TTL 输入
	输出	0	×	0	0		(SCL10 输出 = 1 注 3)	CMOS 输出
		0	×	1	0			N 沟道漏极开路输出
P45	输入	1	—	—	0	PFDEG=0	×	
	输出	0			0	PFDEG=0	×	
P46	输入	1	0	×	—	PFSEGxx=0	×	CMOS 输入
		1	1	×			×	TTL 输入
	输出	0	×	0		PFSEGxx=0	(SCK00/SCL00 输出 = 1 注 3)	CMOS 输出
		0	×	1				N 沟道漏极开路输出
P47	输入	1	0	×	—	PFSEGxx=0	×	CMOS 输入
		1	1	×			×	TTL 输入
	输出	0	×	0		PFSEGxx=0	(SDA00 输出 = 1 注 3)	CMOS 输出
		0	×	1				N 沟道漏极开路输出

- 注 1. 如果在设定 PIOR0=1 时 P40 和 P41 用作通用端口，必须将定时器输出寄存器 0 (TO0) 的 bit0 和 bit7 (TO00、TO07) 以及定时器输出允许寄存器 0 (TOE0) 的 bit0 和 bit7 (TOE00、TOE07) 设定为与其初始状态相同的值“0”。
2. 将 P42/TI05/TO05/(TxD1)/IVREF1 用作通用端口时，将定时器输出寄存器 0 (TO0) 的 bit5 (TO05)、定时器输出允许寄存器 0 (TOE0) 的 bit5 (TOE05) 设定为与其初始状态相同的值“0”。
3. 如果在设定 PIOR1、PIOR2=1 时 P42 ~ P44、P46 和 P47 用作通用端口，必须将串行通道允许状态寄存器 0 (SE0)、串行输出寄存器 0 (SO0) 和串行输出允许寄存器 0 (SOE0) 设定为与其初始状态相同的值。
4. () 中是 PIORx = 1 时的情况。

注意 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

PIM4x: 端口输入模式寄存器 4

POM4x: 端口输出模式寄存器 4

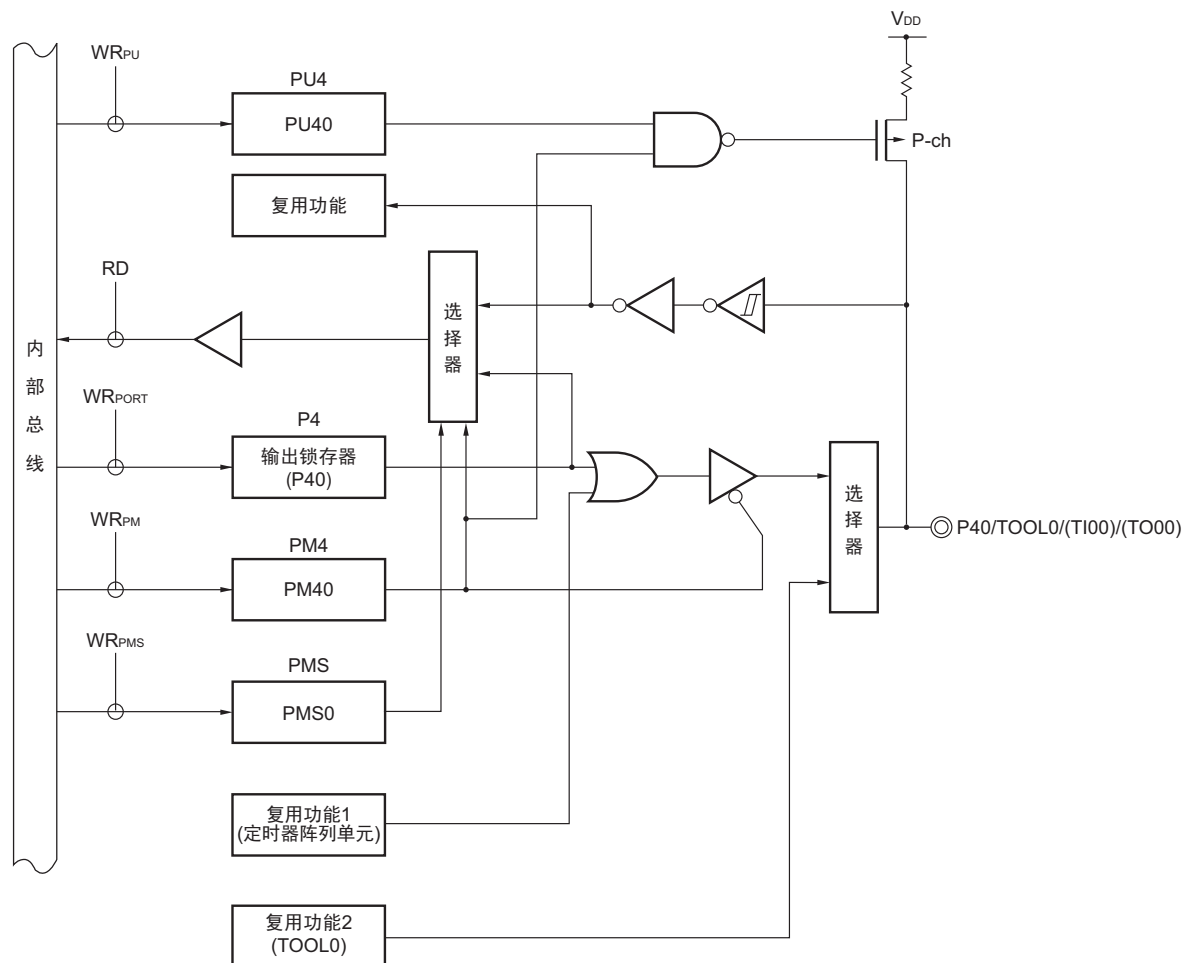
PMC4x: 端口模式控制寄存器 4

PFSEGxx: LCD 端口功能寄存器

PIORx: 外围 I/O 重定向寄存器

端口 4 的框图如图 4-16 ~ 图 4-21 所示。

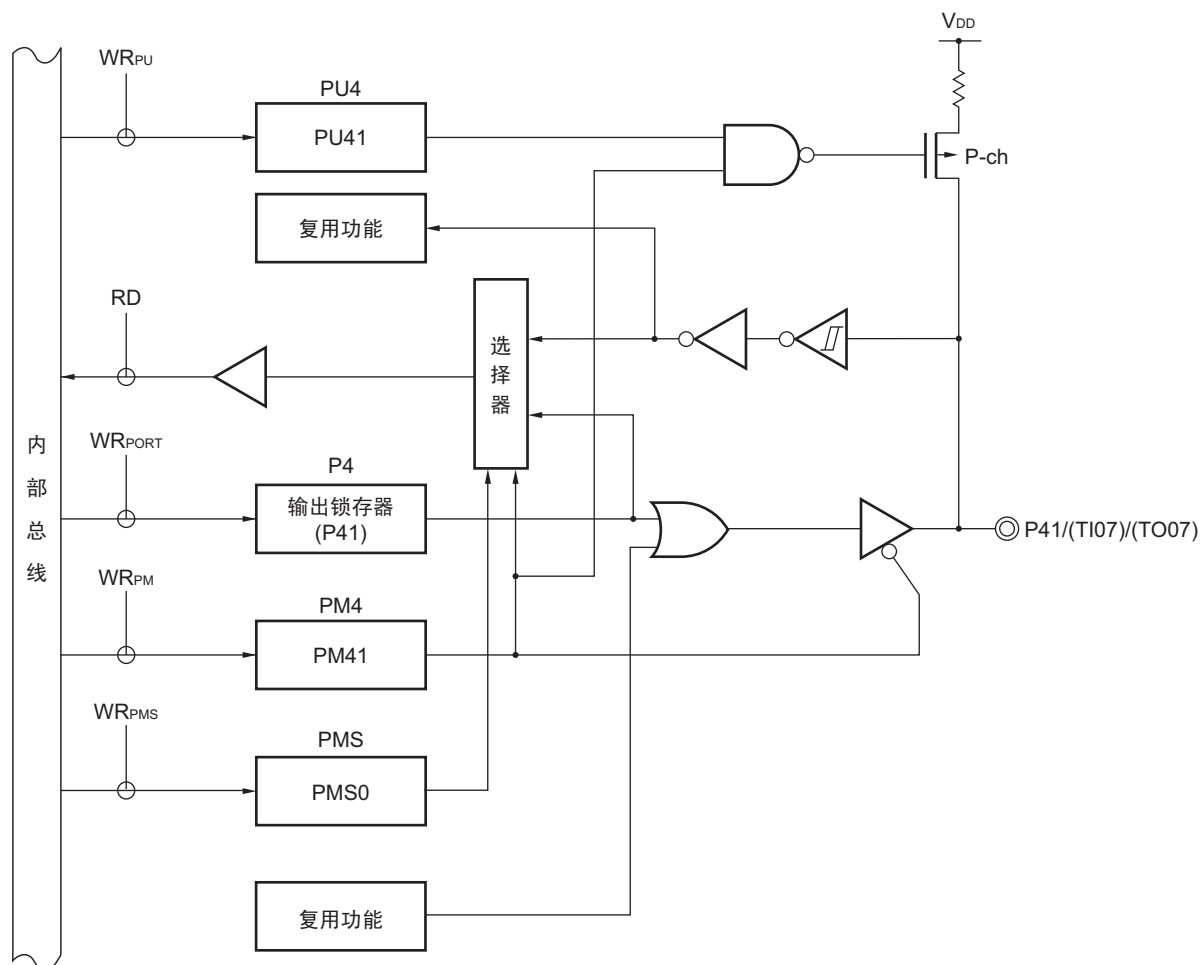
图 4-16 P40 的框图



- 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
 PMS: 端口模式选择寄存器
 RD: 读信号
 WRxx: 写信号

备注 可通过设定外围 I/O 重定向寄存器 (PIOR)，分配上图 () 内的功能。详细内容请参照“图 4-42 外围 I/O 重定向寄存器 (PIOR) 的格式”。

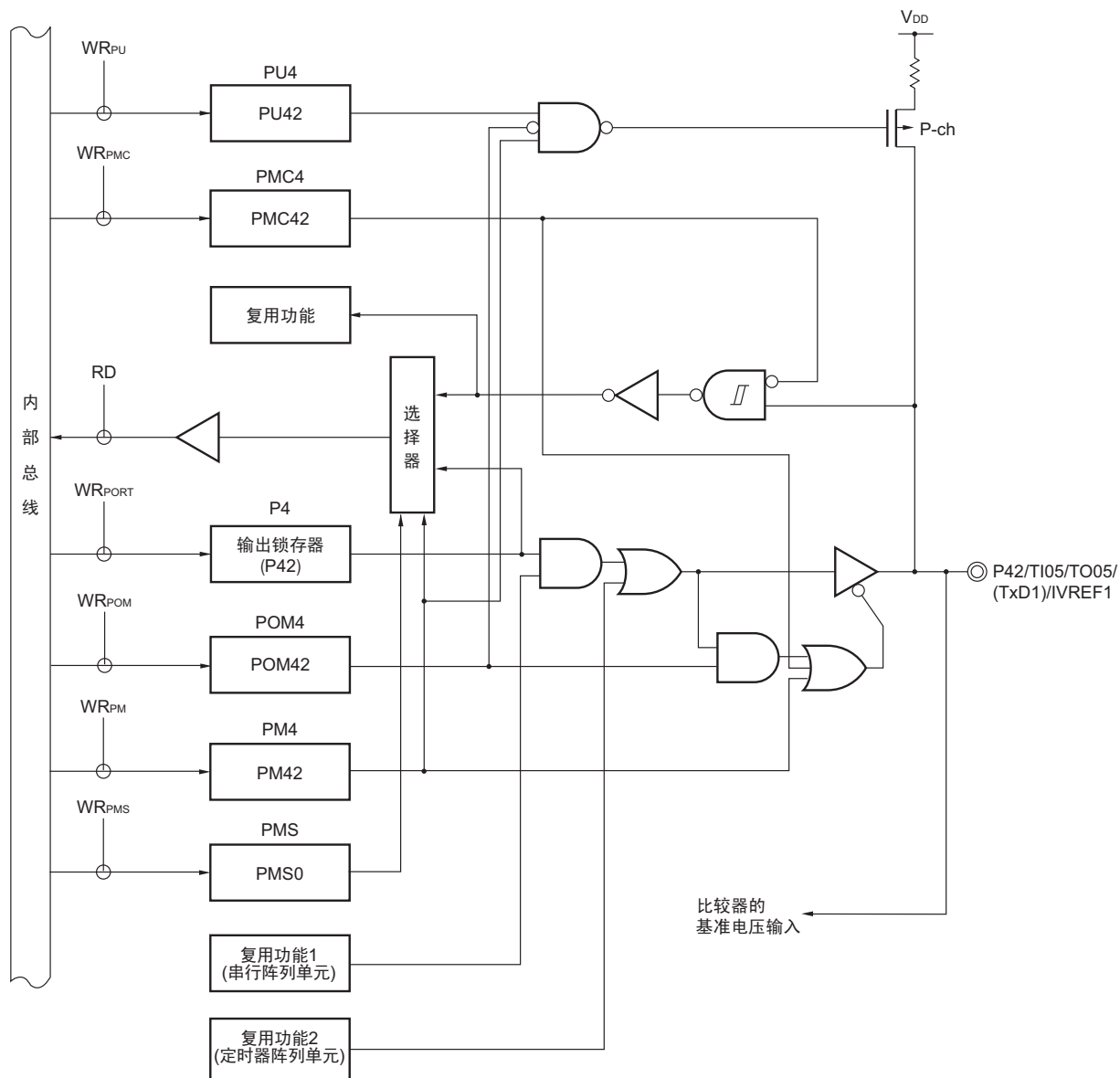
图 4-17 P41 的框图



- 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
 PMS: 端口模式选择寄存器
 RD: 读信号
 WRxx: 写信号

备注 可通过设定外围 I/O 重定向寄存器 (PIOR)，分配上图 () 内的功能。详细内容请参照“图 4-42 外围 I/O 重定向寄存器 (PIOR) 的格式”。

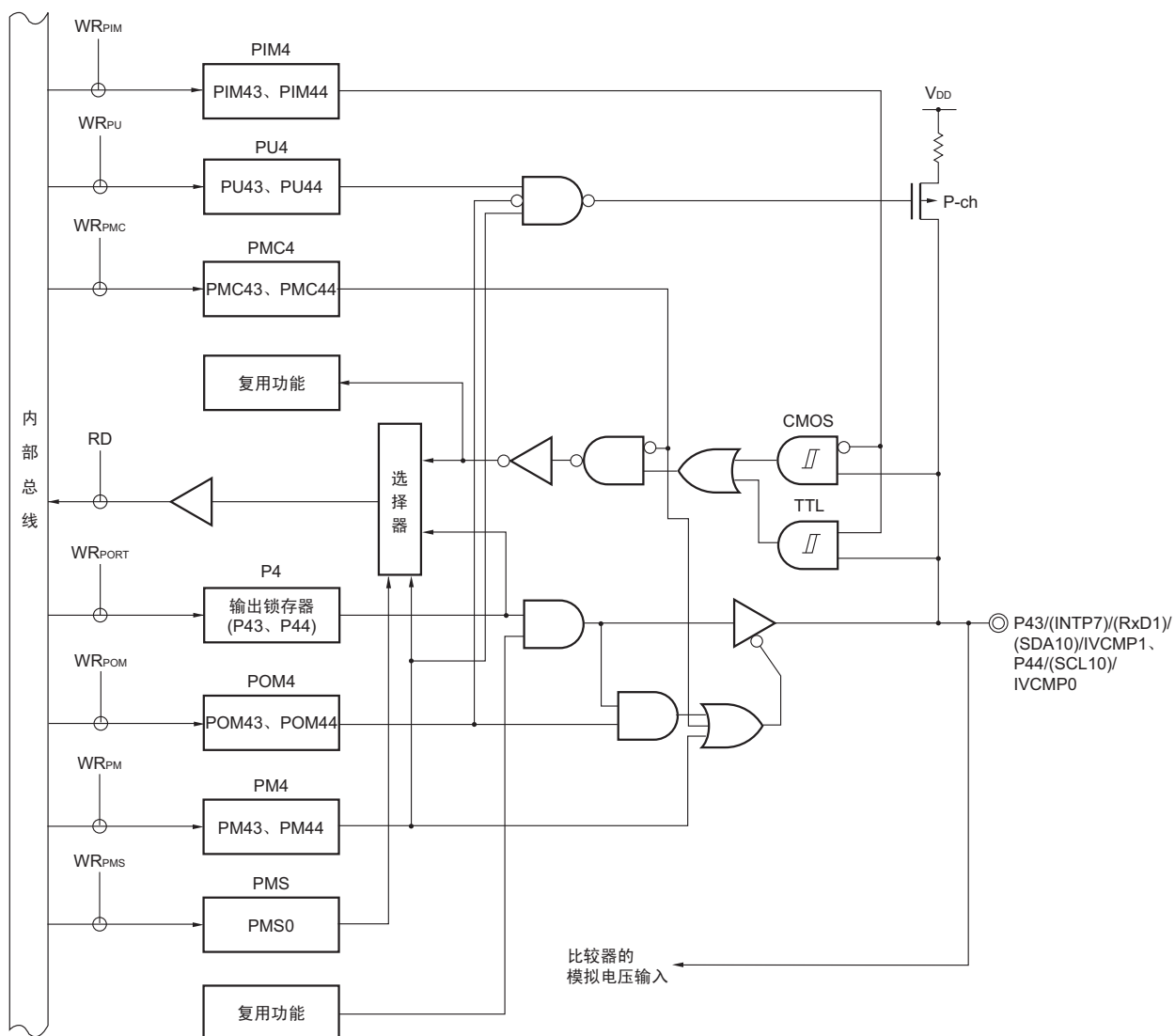
图 4-18 P42 的框图



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
POM4:	端口输出模式寄存器 4
PMC4:	端口模式控制寄存器 4
PMS:	端口模式选择寄存器
RD:	读信号
WRxx:	写信号

备注 可通过设定外围 I/O 重定向寄存器 (PIOR)，分配上图 () 内的功能。详细内容请参照“图 4-42 外围 I/O 重定向寄存器 (PIOR) 的格式”。

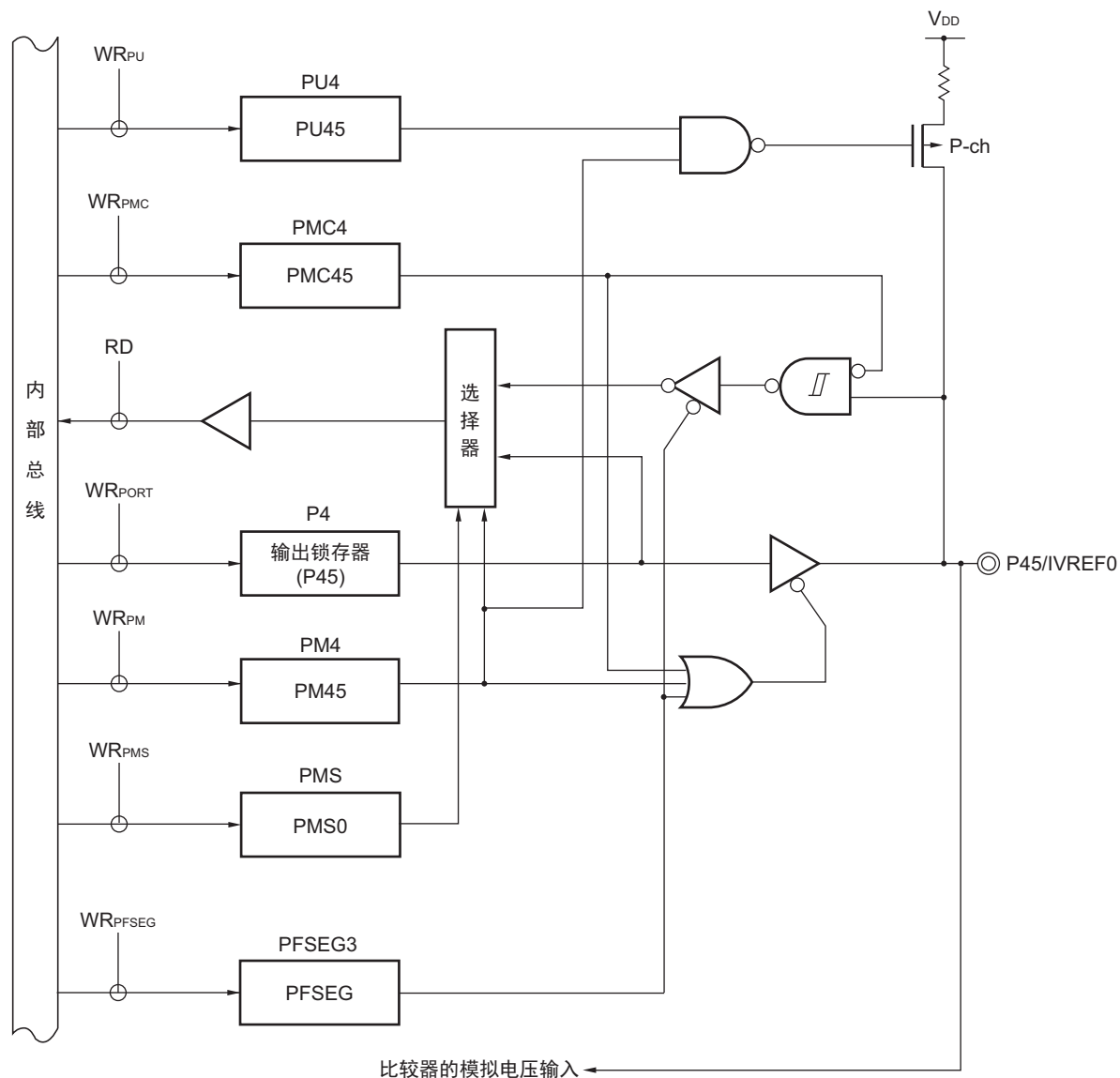
图 4-19 P43 和 P44 的框图



- 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
 PIM4: 端口输入模式寄存器 4
 POM4: 端口输出模式寄存器 4
 PMC4: 端口模式控制寄存器 4
 PMS: 端口模式选择寄存器
 RD: 读信号
 WRxx: 写信号

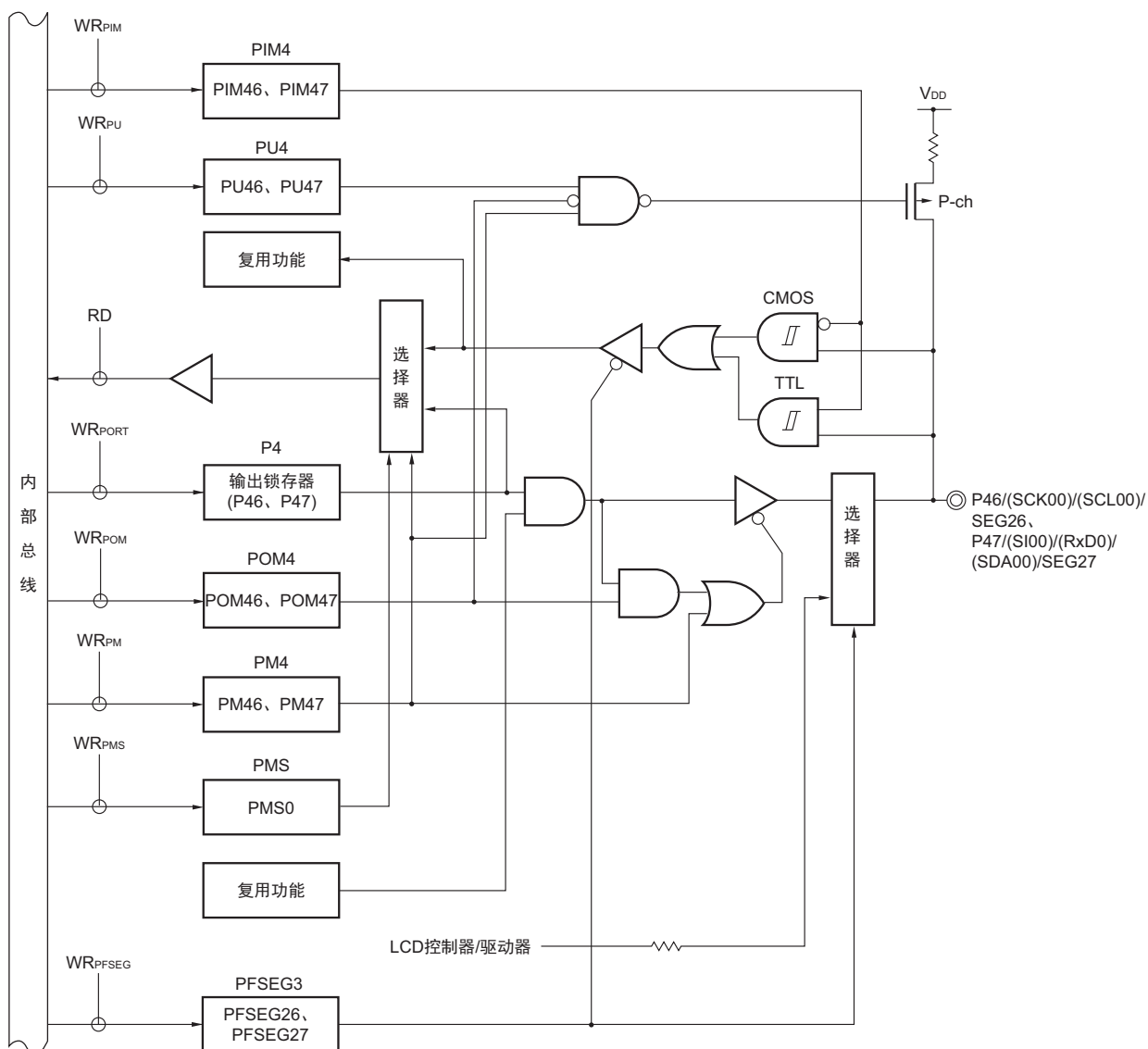
备注 可通过设定外围 I/O 重定向寄存器 (PIOR)，分配上图 () 内的功能。详细内容请参照“图 4-42 外围 I/O 重定向寄存器 (PIOR) 的格式”。

图 4-20 P45 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
 PMC4: 端口模式控制寄存器 4
 PMS: 端口模式选择寄存器
 PFSEG3: LCD 端口功能寄存器 3
 RD: 读信号
 WRxx: 写信号

图 4-21 P46 和 P47 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
 PIM4: 端口输入模式寄存器 4
 POM4: 端口输出模式寄存器 4
 PMS: 端口模式选择寄存器
 PFSEG3: LCD 端口功能寄存器 3
 RD: 读信号
 WRxx: 写信号

备注 可通过设定外围 I/O 重定向寄存器 (PIOR)，分配上图 () 内的功能。详细内容请参照“图 4-42 外围 I/O 重定向寄存器 (PIOR) 的格式”。

4.2.6 端口 5

端口 5 是具有输出锁存器的输入 / 输出端口。端口 5 可以通过端口模式寄存器 5（PM5）以 1 位为单位指定为输入模式或者输出模式。当 P50 ~ P57 引脚用作输入端口时，可通过上拉电阻选择寄存器 5（PU5）以 1 位为单位使用内部上拉电阻。

P53 和 P55 引脚的输入可通过设定端口输入模式寄存器 5（PIM5）以 1 位为单位指定为通常的输入缓冲器或者 TTL 输入缓冲器。

P53 ~ P56 引脚的输出可通过端口输出模式寄存器 5（POM5）以 1 位为单位设定为 N 沟道漏极开路输出（V_{DD} 耐压）。

将全部引脚用作数字输入 / 输出端口时，必须通过 LCD 端口功能寄存器 0、1（PFSEG0、PFSEG1）设定为“用作端口（段输出除外）”（可以以 1 位为单位设定）。

端口 5 作为复用功能，有 LCD 控制器 / 驱动器的段输出、定时器的输入 / 输出、外部中断请求输入、串行接口的数据输入 / 输出和时钟输入 / 输出。

在产生复位信号后，端口 5 为数字输入无效注。

注 数字输入无效是指数字输出、数字输入、模拟输入以及 LCD 输出均处于无效的状态。

表 4-9 使用端口 5 时的寄存器设定

名称	输入 / 输出	PM5x	PIM5x	POM5x	PFSEGxx	复用功能设定	备注
P50	输入	1	—	—	0	×	
	输出	0			0	×	
P51	输入	1	—	—	0	×	
	输出	0			0	×	
P52	输入	1	—	—	0	×	
	输出	0			0	TO00 输出 = 0 注 1	
P53	输入	1	0	×	0	×	CMOS 输入
		1	1	×		×	TTL 输入
	输出	0	×	0	0	SCIO1 输出 = 1 注 2	CMOS 输出
		0	×	1			N 沟道漏极开路输出
P54	输入	1	—	×	0	×	
	输出	0		0	0	TO02 输出 = 0 注 1	CMOS 输出
		0		1		SCCLK1 输出 = 1 注 2	N 沟道漏极开路输出
P55	输入	1	0	×	0	×	CMOS 输入
		1	1	×		×	TTL 输入
	输出	0	×	0	0	SCIO0 输出 = 1 注 2	CMOS 输出
		0	×	1			N 沟道漏极开路输出
P56	输入	1	—	×	0	×	
	输出	0		0	0	TO06 输出 = 0 注 1	CMOS 输出
		0		1		SCCLK0 输出 = 1 注 2	N 沟道漏极开路输出
P57	输入	1	—	—	0	×	
	输出	0			0	×	

注 1. 将 P52/TI00/TO00/INTP1/SEG6、P54/TI02/TO02/SCCLK1/SEG8、P56/TI06/TO06/SCCLK0/SEG10 用作通用端口时，必须将定时器输出寄存器 0 (TO0) 的 bit0、2、6 (TO00、TO002、TO06)、定时器输出允许寄存器 0 (TOE0) 的 bit0、2、6 (TOE00、TOE02、TOE06) 设定为与其初始状态相同的值“0”。另外，必须将外围允许寄存器 1 的 bit1 和 bit2 (SMCI0EN、SMCI1EN) 设定为“0”。

2. 将 P53/INTP2/SCIO1/SEG7、P54/TI02/TO02/SCCLK1/SEG8、P55/INTP5/SCIO0/SEG9、P56/TI06/TO06/SCCLK0/SEG10 用作通用端口时，必须将外围允许寄存器 1 的 bit1 和 bit2 (SMCI0EN、SMCI1EN) 设定为“0”。

备注 ×: 忽略

PM5x: 端口模式寄存器 5

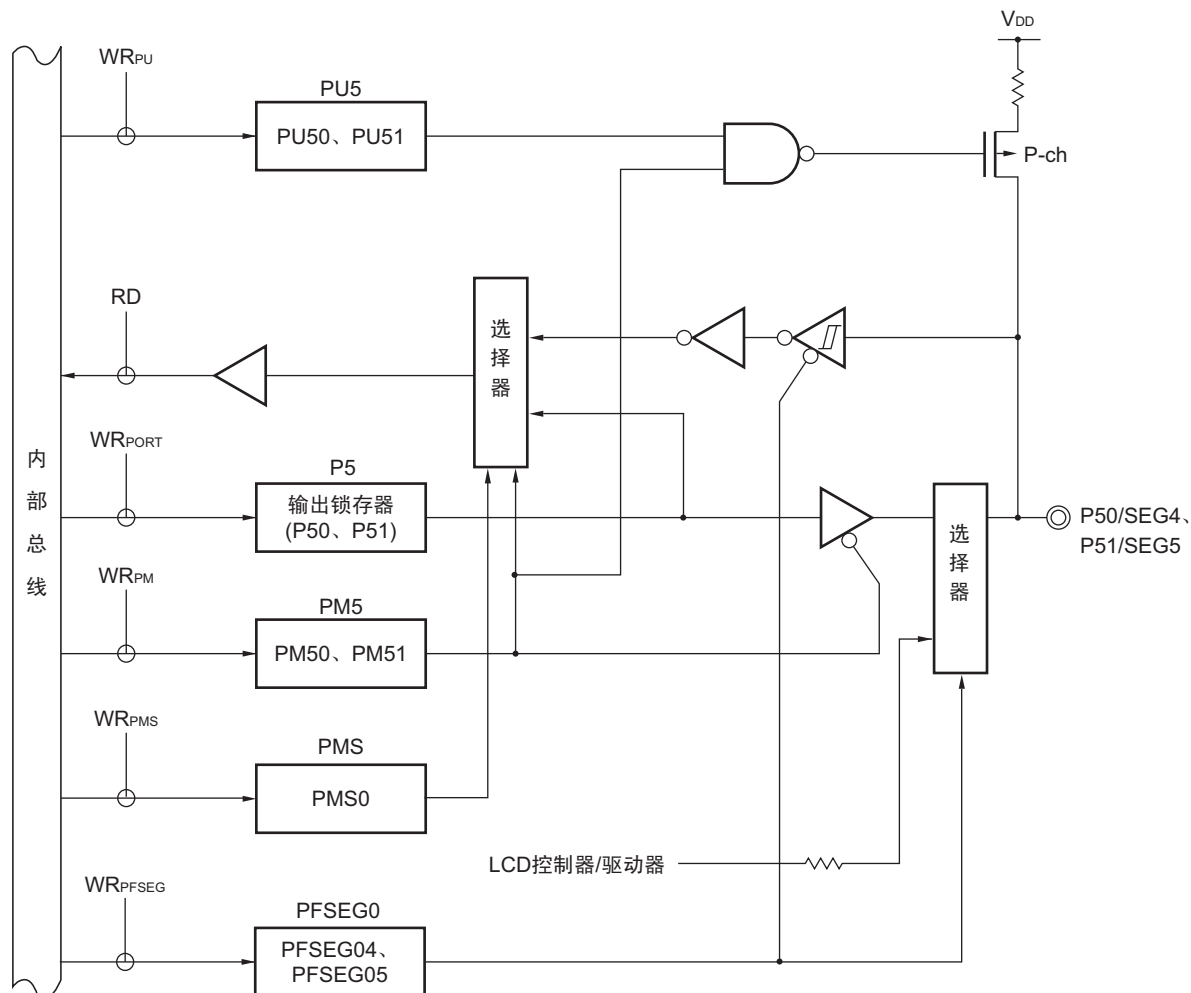
PIM5x: 端口输入模式寄存器 5

POM5x: 端口输出模式寄存器 5

PFSEGxx: LCD 端口功能寄存器

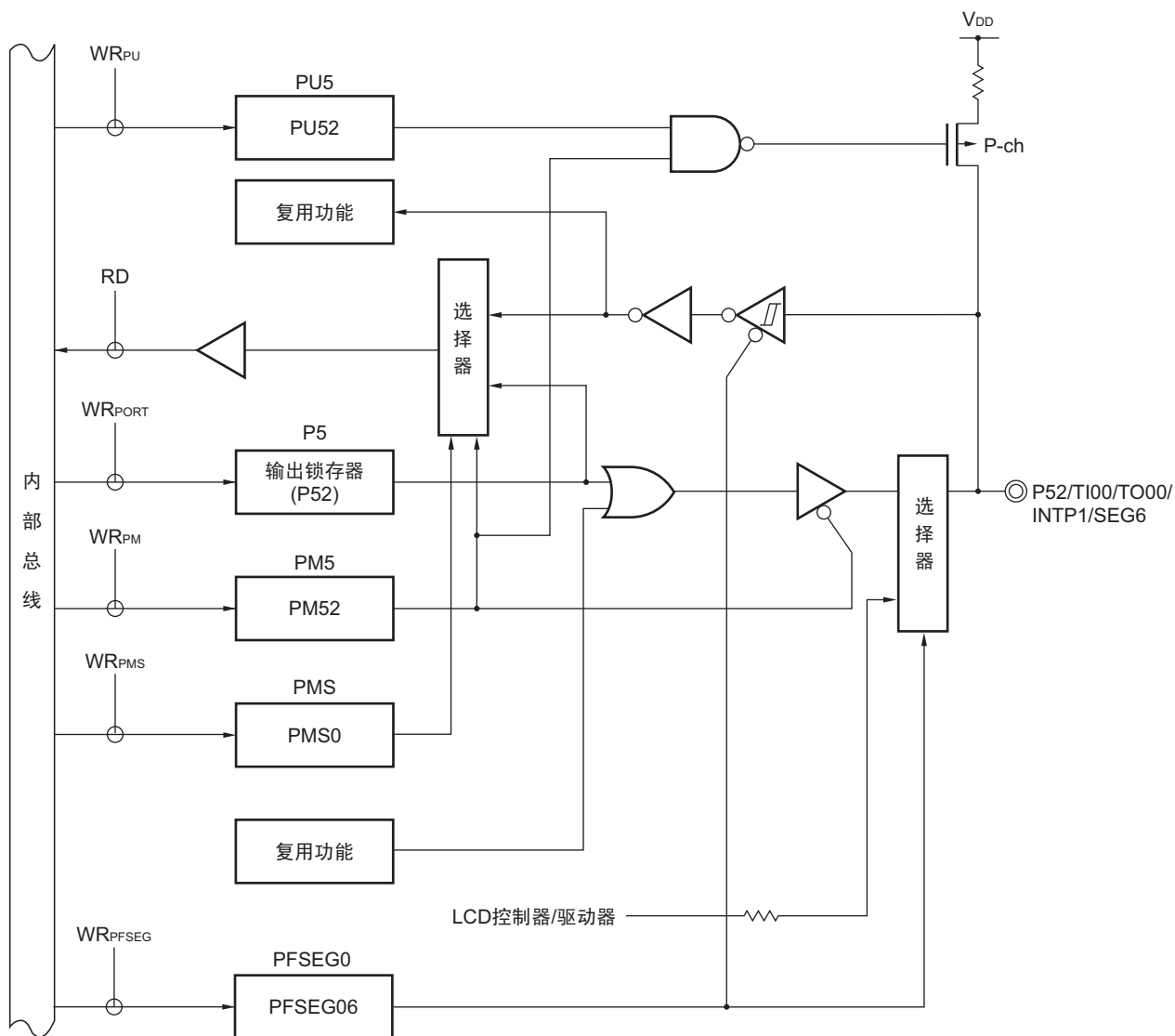
端口 5 的框图如图 4-22 ~ 图 4-26 所示。

图 4-22 P50 和 P51 的框图



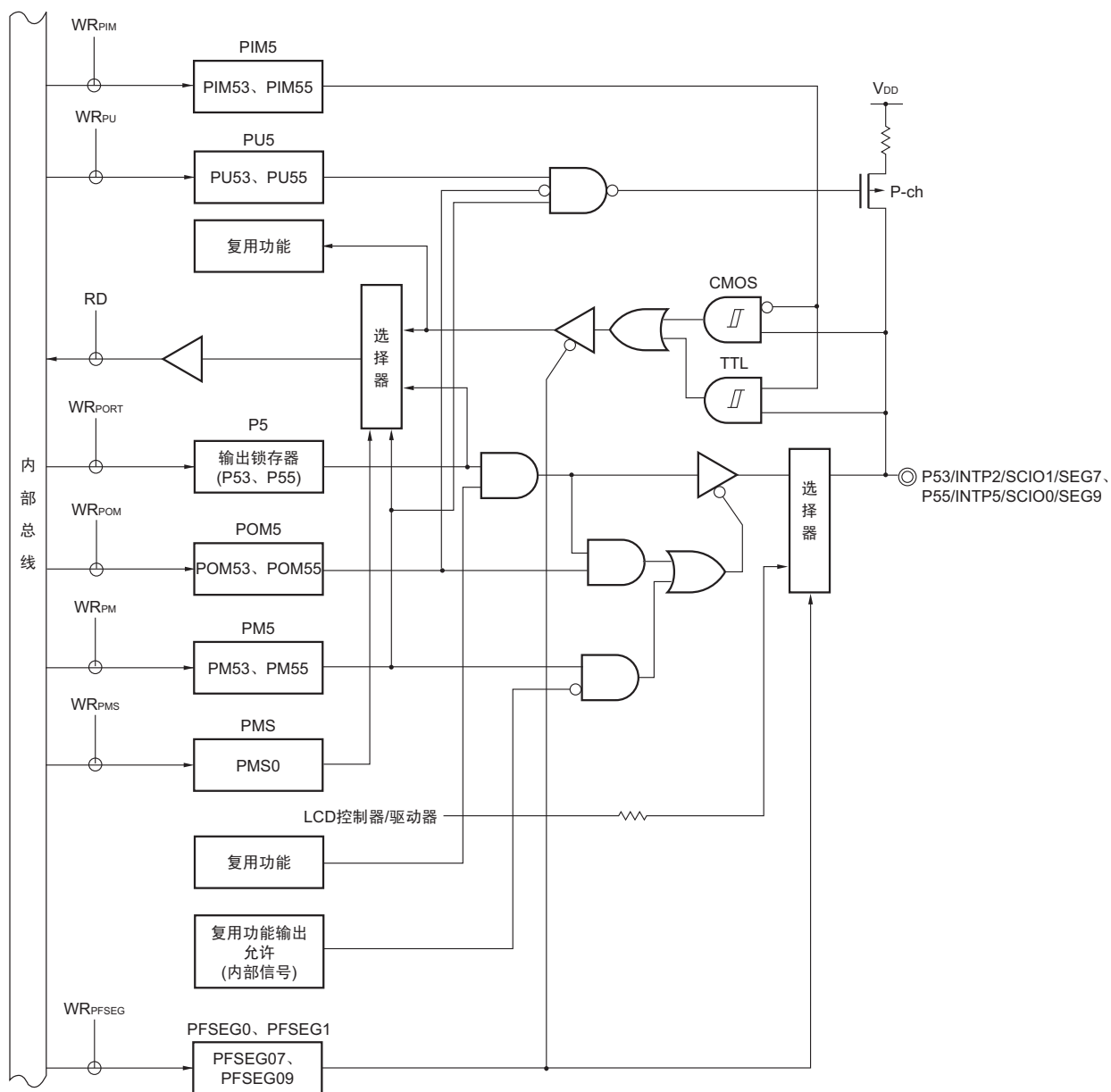
- P5: 端口寄存器 5
- PU5: 上拉电阻选择寄存器 5
- PM5: 端口模式寄存器 5
- PMS: 端口模式选择寄存器
- PFSEG0: LCD 端口功能寄存器 0
- RD: 读信号
- WRxx: 写信号

图 4-23 P52 的框图



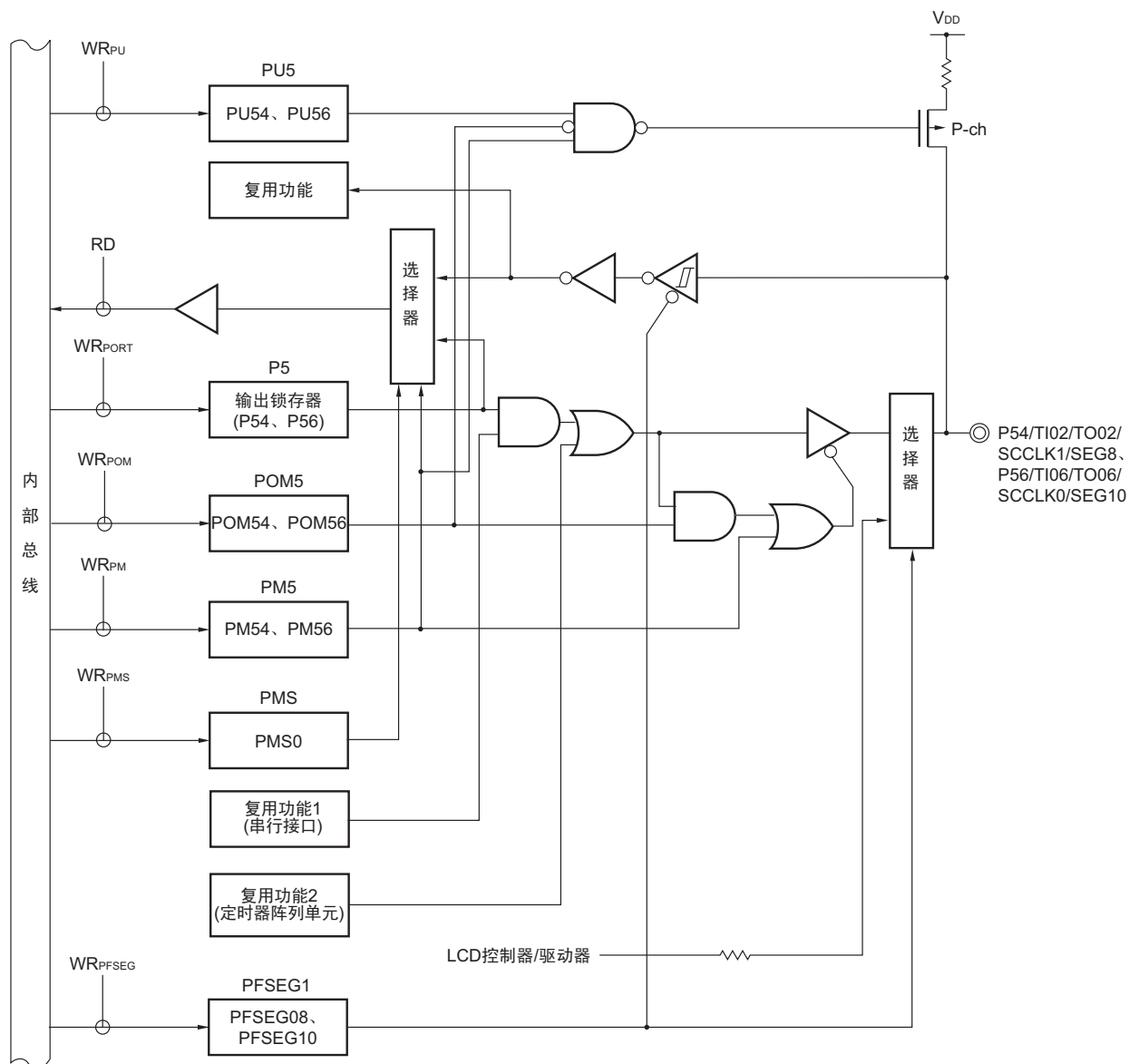
- P5: 端口寄存器 5
 PU5: 上拉电阻选择寄存器 5
 PM5: 端口模式寄存器 5
 PMS: 端口模式选择寄存器
 PFSEG0: LCD 端口功能寄存器 0
 RD: 读信号
 WRxx: 写信号

图 4-24 P53 和 P55 的框图



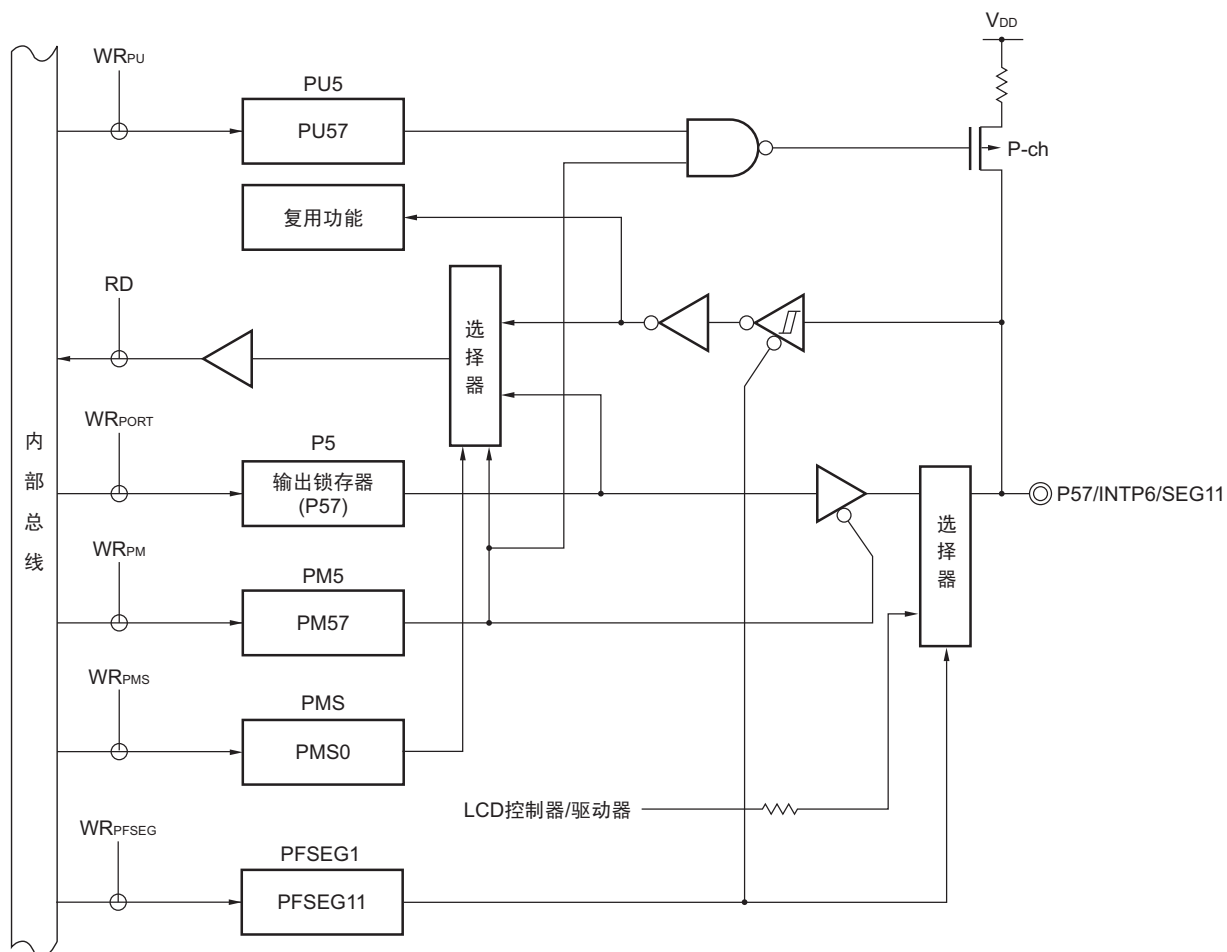
- P5: 端口寄存器 5
 PU5: 上拉电阻选择寄存器 5
 PM5: 端口模式寄存器 5
 PIM5: 端口输入模式寄存器 5
 POM5: 端口输出模式寄存器 5
 PMS: 端口模式选择寄存器
 PFSEG0、PFSEG1: LCD 端口功能寄存器 0、1
 RD: 读信号
 WRxx: 写信号
 SCTE: 智能卡串行控制寄存器 n (SCCRn) 的 bit5

图 4-25 P54 和 P56 的框图



- P5:** 端口寄存器 5
PU5: 上拉电阻选择寄存器 5
PM5: 端口模式寄存器 5
POM5: 端口输出模式寄存器 5
PMS: 端口模式选择寄存器
PFSEG1: LCD 端口功能寄存器 1
RD: 读信号
WRxx: 写信号

图 4-26 P57 的框图



- P5: 端口寄存器 5
 PU5: 上拉电阻选择寄存器 5
 PM5: 端口模式寄存器 5
 PMS: 端口模式选择寄存器
 PFSEG1: LCD 端口功能寄存器 1
 RD: 读信号
 WRxx: 写信号

4.2.7 端口 6

端口 6 是具有输出锁存器的输入 / 输出端口。端口 6 可以通过端口模式寄存器 6（PM6）以 1 位为单位指定为输入模式或者输出模式。

P60 和 P61 引脚的输出为 N 沟道漏极开路输出（6 V 耐压）。

端口 6 作为复用功能，有串行接口的数据输入 / 输出、时钟输入 / 输出和定时器的输入 / 输出。

在产生复位信号后，端口 6 为输入模式。

表 4-10 使用端口 6 时的寄存器设定

名称	输入 / 输出	PM6x	复用功能设定 注 3	备注
P60	输入	1	SCLA0 输出 = 0 注 1 (TO01 输出 = 0 注 2)	
	输出	0		
P61	输入	1	SDAA0 输出 = 0 注 1 (TO02 输出 = 0 注 2)	
	输出	0		

- 注 1. 将 P60/SCLA0/(TI01)/(TO01) 和 P61/SDAA0/(TI02)/(TO02) 用作通用端口时，必须停止对应的串行接口 IICA 的运行。
2. 如果在设定 PIOR0=1 时 P60、P61 用作通用端口，必须将定时器输出寄存器 0（TO0）的 bit1 和 bit2（TO01、TO02）、定时器输出允许寄存器 0（TOE0）的 bit1 和 bit2（TOE01、TOE02）设定为与其初始状态相同的值“0”。
3. () 中是 PIOR0 = 1 时的情况。

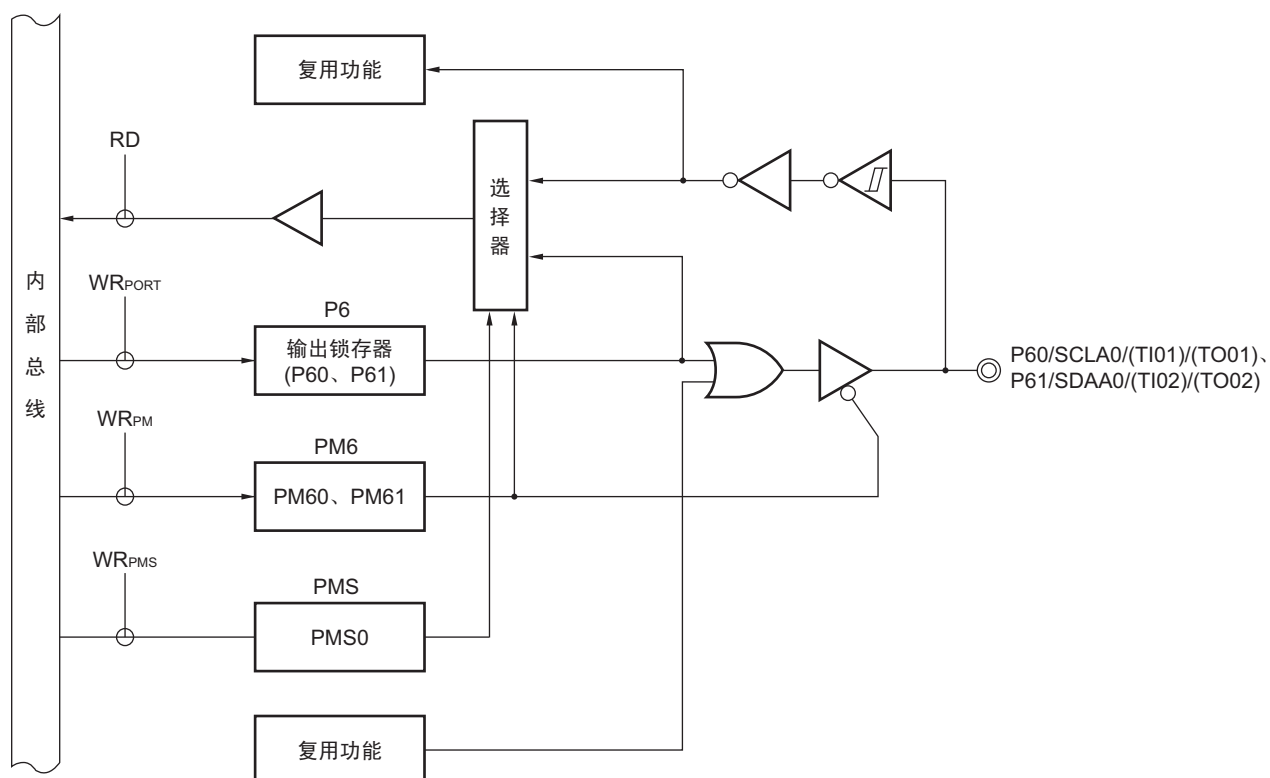
备注 x: 忽略

PM6x: 端口模式寄存器 6

PIORx: 外围输 I/O 重定向寄存器

端口 6 的框图如图 4-27 所示。

图 4-27 P60 和 P61 的框图



P6: 端口寄存器 6
 PM6: 端口模式寄存器 6
 PMS: 端口模式选择寄存器
 RD: 读信号
 WRxx: 写信号

备注 可通过设定外围 I/O 重定向寄存器 (PIOR)，分配上图 () 内的功能。详细内容请参照“图 4-42 外围 I/O 重定向寄存器 (PIOR) 的格式”。

4.2.8 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 位为单位指定为输入模式或者输出模式。当用作输入端口时，可通过上拉电阻选择寄存器 7（PU7）以 1 位为单位使用内部上拉电阻。

将全部引脚用作数字输入 / 输出端口时，必须通过 LCD 端口功能寄存器 1、2（PFSEG1、PFSEG2）设定为“用作端口（端输出除外）”（可以以 1 位为单位设定）。

端口 7 作为复用功能，有 LCD 控制器 / 驱动器的段输出。

在产生复位信号后，端口 7 为数字输入无效注。

注 数字输入无效是指数字输出、数字输入、模拟输入以及 LCD 输出均处于无效的状态。

表 4-11 使用端口 7 时的寄存器设定

名称	输入 / 输出	PM7x	PFSEGxx	复用功能设定	备注
P70	输入	1	0	×	
	输出	0	0	×	
P71	输入	1	0	×	
	输出	0	0	×	
P72	输入	1	0	×	
	输出	0	0	×	
P73	输入	1	0	×	
	输出	0	0	×	
P74	输入	1	0	×	
	输出	0	0	×	
P75	输入	1	0	×	
	输出	0	0	×	
P76	输入	1	0	×	
	输出	0	0	×	
P77	输入	1	0	×	
	输出	0	0	×	

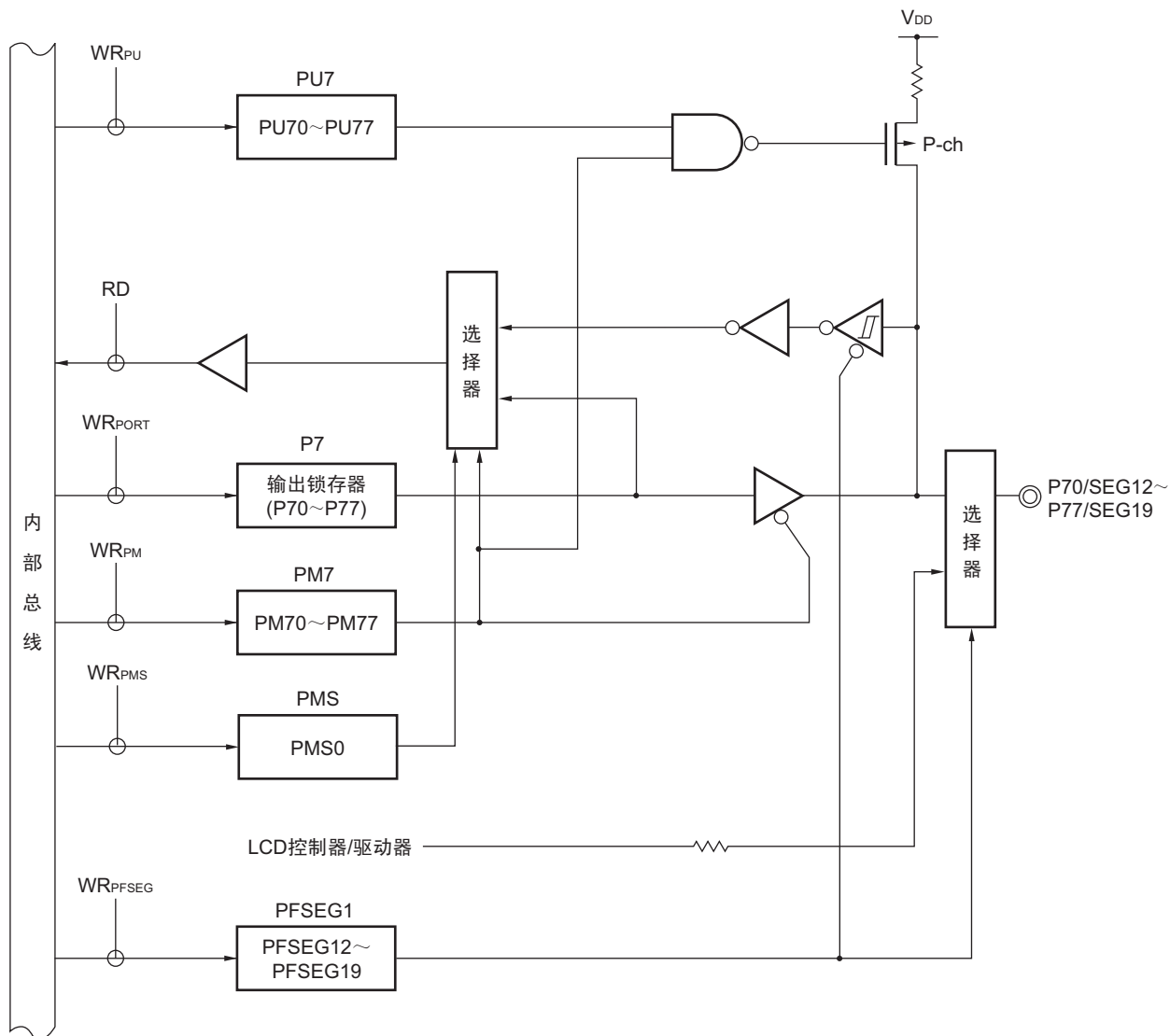
备注 ×： 忽略

PM7x： 端口模式寄存器 7

PFSEGxx： LCD 端口功能寄存器

端口 7 的框图如图 4-28 所示。

图 4-28 P70 ~ P77 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
PMS:	端口模式选择寄存器
PFSEG1:	LCD 端口功能寄存器 1
RD:	读信号
WRxx:	写信号

4.2.9 端口 12

P125 ~ P127 是具有输出锁存器的 3 位输入 / 输出端口。可以通过端口模式寄存器 12（PM12）以 1 位为单位指定为输入模式或者输出模式。当用作输入端口时，可通过上拉电阻选择寄存器 12（PU12）使用内部上拉电阻。

P121 ~ P124 是 4 位输入专用端口。

将 P125 ~ P127 用作数字输入 / 输出端口时，必须通过 LCD 输入切换控制寄存器（ISCLCD）设定为“允许数字输入”（可以以 1 位为单位设定）。

端口 12 作为复用功能，有主系统时钟的谐振器连接、副系统时钟的谐振器连接、主系统时钟的外部时钟输入、副系统时钟的外部时钟输入、定时器的输入 / 输出、LCD 控制器 / 驱动器的电容器连接以及用于驱动 LCD 的电压引脚。

在产生复位信号后，P121 ~ P124 引脚为输入模式，P125 ~ P127 引脚为数字输入无效注。

注 数字输入无效是指数字输出、数字输入、模拟输入以及 LCD 输出均处于无效的状态。

表 4-12 使用端口 12 时的寄存器设定

名称	输入 / 输出	PM12x	ISCLCD	复用功能设定 注 2	备注
P121	输入	—	—	CMC 寄存器的 OSCSEL 位 = 0 或者 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 或者 EXCLKS 位 = 1	
P124	输入	—	—	CMC 寄存器的 OSCSELS 位 = 0	
P125	输入	1	ISCVL3 = 1	×	
	输出	0		(TO06 输出 = 0 注 1)	
P126	输入	1	ISCCAP = 1	×	
	输出	0		(TO04 输出 = 0 注 1)	
P127	输入	1	ISCCAP = 1	×	
	输出	0		(TO03 输出 = 0 注 1)	

注 1. 如果在设定 PIOR0=1 时 P125 ~ P127 用作通用端口，必须将定时器输出寄存器 0（TO0）的 bit3、4、6（TO03、TO04、TO06）和定时器输出允许寄存器 0（TOE0）的 bit3、4、6（TOE03、TOE04、TOE06）设定为与其初始状态相同的值“0”。

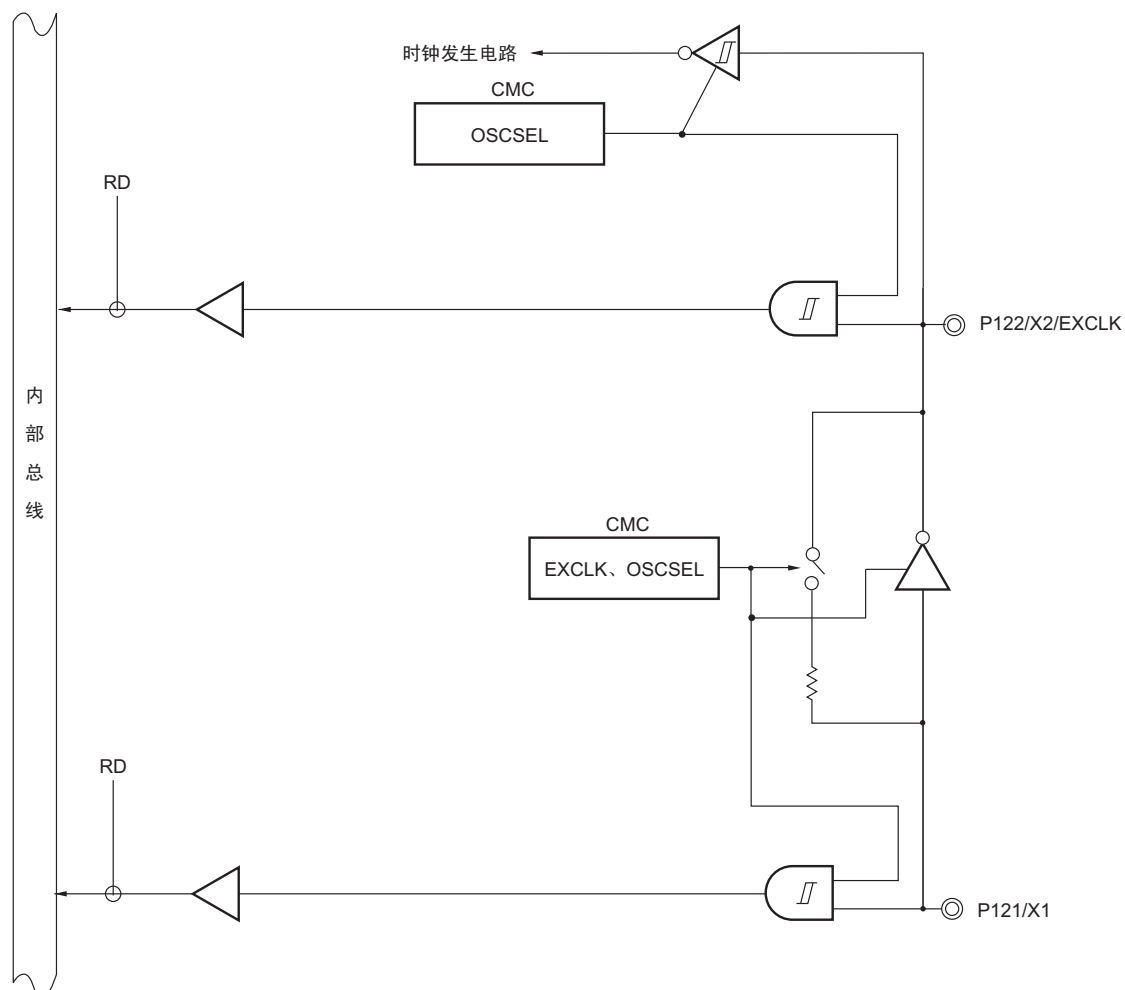
2. () 中是 PIORx = 1 时的情况。

注意 解除复位后只能进行一次 P121 ~ P124 的功能设定。一旦将端口设定为 X1、XT1 振荡或者外部时钟输入，除非复位，否则不能用作输入端口。

备注 ×: 忽略
PM12x: 端口模式寄存器 12
ISCLCD: LCD 输入切换控制寄存器
PIORx: 外围 I/O 重定向寄存器

端口 12 的框图如图 4-29 ~ 图 4-32 所示。

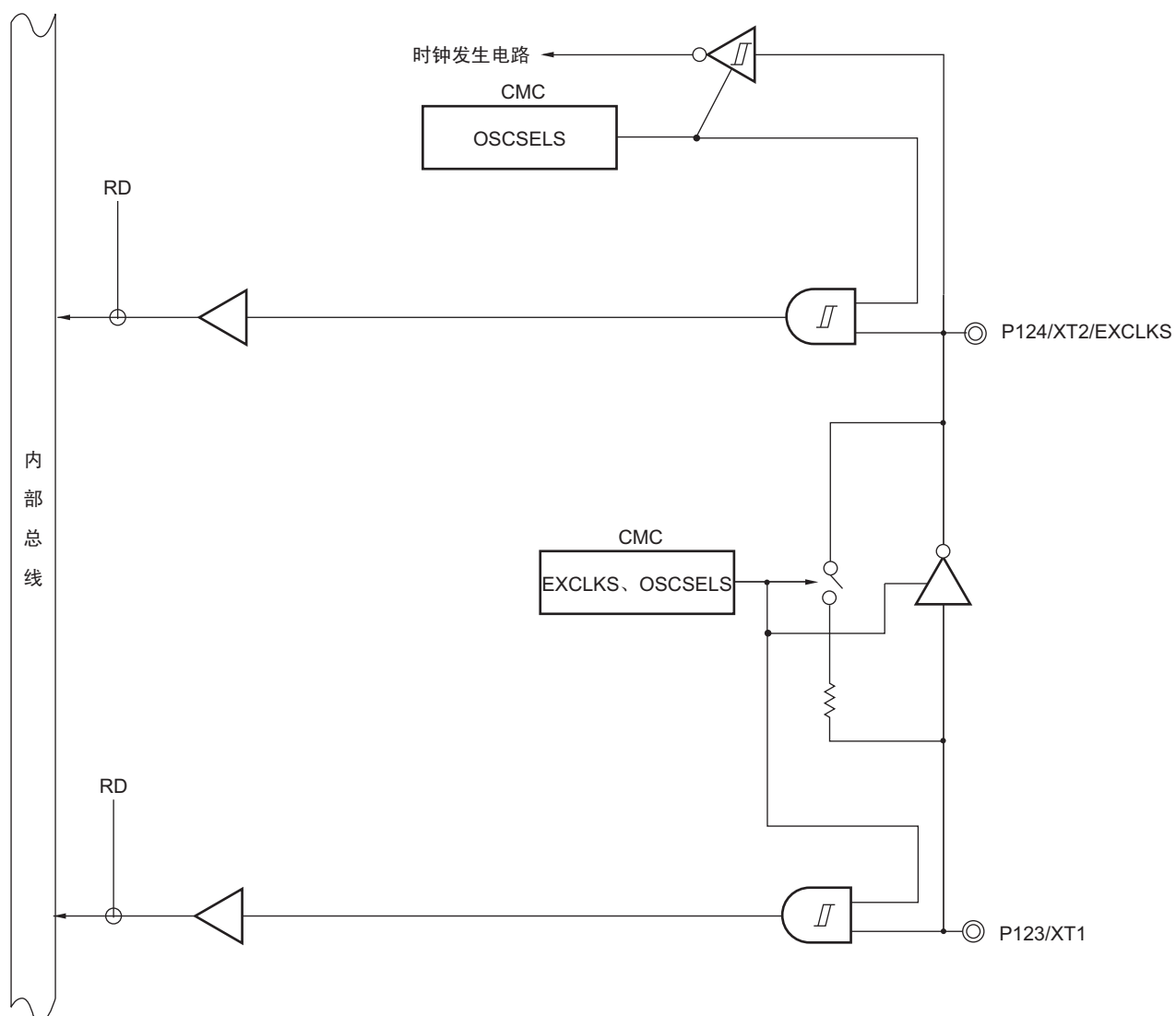
图 4-29 P121 和 P122 的框图



CMC: 时钟运行模式控制寄存器

RD: 读信号

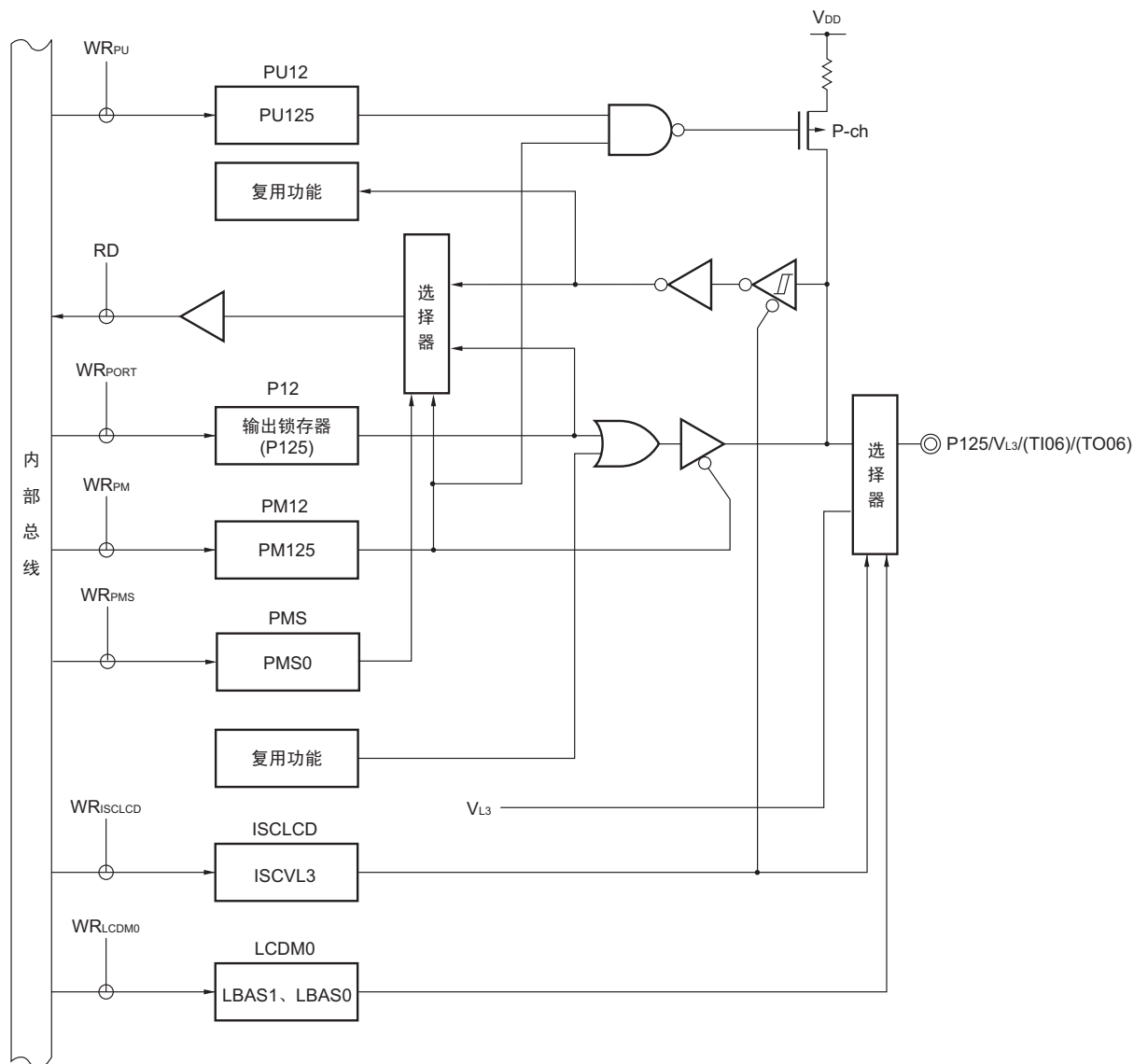
图 4-30 P123 和 P124 的框图



CMC: 时钟运行模式控制寄存器

RD: 读信号

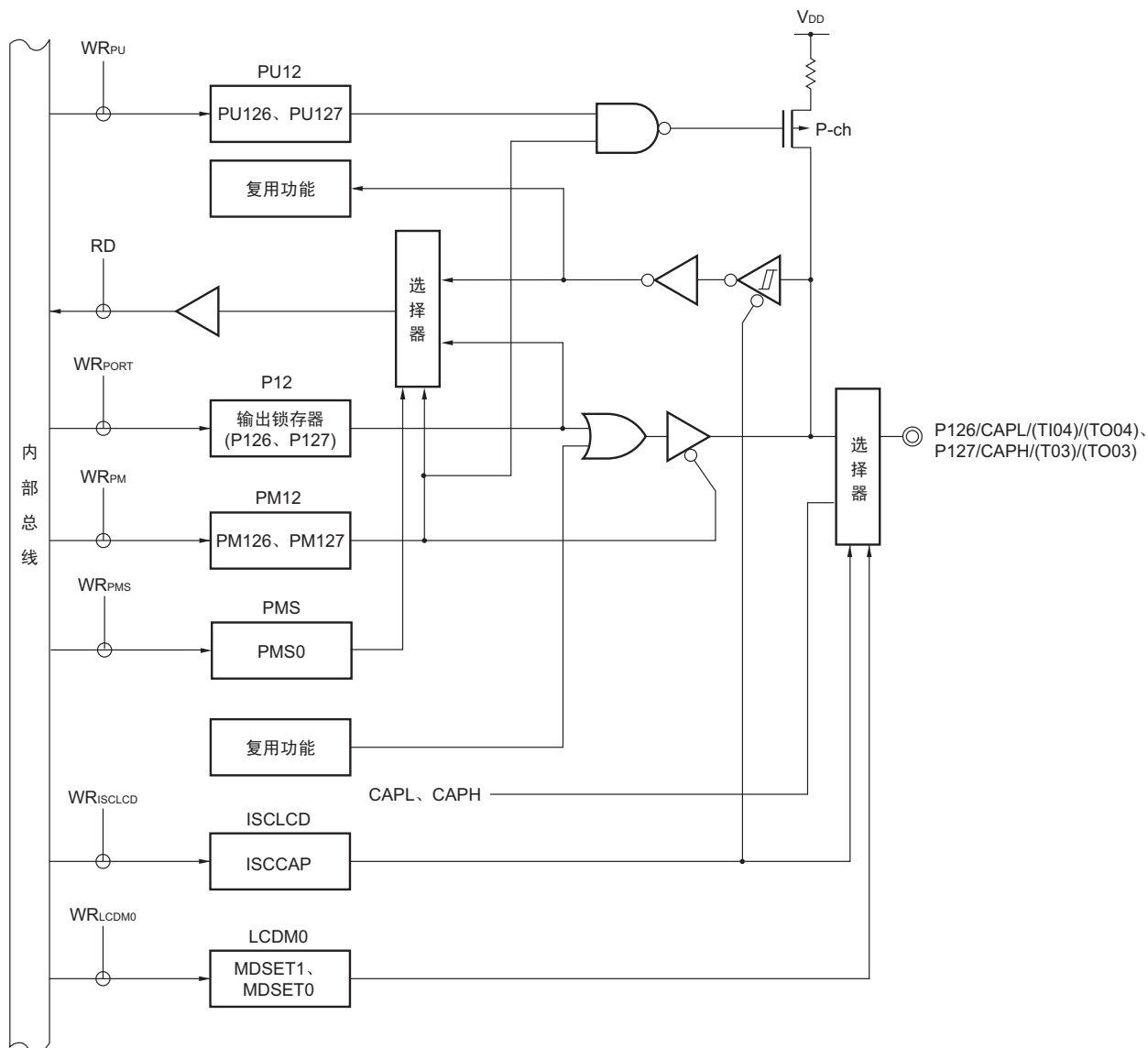
图 4-31 P125 的框图



- 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
 PMS: 端口模式选择寄存器
 ISCLCD: LCD 输入切换控制寄存器
 LCDM0: LCD 模式寄存器 0
 RD: 读信号
 WRxx: 写信号

备注 可通过设定外围 I/O 重定向寄存器 (PIOR)，分配上图 () 内的功能。详细内容请参照“图 4-42 外围 I/O 重定向寄存器 (PIOR) 的格式”。

图 4-32 P126 和 P127 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
 PMS: 端口模式选择寄存器
 ISCLCD: LCD 输入切换控制寄存器
 LCDM0: LCD 模式寄存器 0
 RD: 读信号
 WRxx: 写信号

备注 可通过设定外围 I/O 重定向寄存器 (PIOR)，分配上图 () 内的功能。详细内容请参照“图 4-42 外围 I/O 重定向寄存器 (PIOR) 的格式”。

4.2.10 端口 13

P130 是具有输出锁存器的输入 / 输出端口。可以通过端口模式寄存器 13（PM13）以 1 位为单位指定为输入模式或者输出模式。当用作输入端口时，可通过上拉电阻选择寄存器 13（PU13）以 1 位为单位使用内部上拉电阻。

P137 是 1 位输入专用端口。

将 P130 引脚用作数字输入 / 输出端口时，必须通过 LCD 端口功能寄存器 3（PFSEG3）设定为“用作端口（段输出除外）”。

端口 13 作为复用功能，有外部中断请求输入、串行接口的数据输入 / 输出和 LCD 控制器 / 驱动器的段输出。

在产生复位信号后，P137 引脚为输入模式，P130 引脚为数字输入无效注。

注 数字输入无效是指数字输出、数字输入、模拟输入以及 LCD 输出均处于无效的状态。

表 4-13 使用端口 13 时的寄存器设定

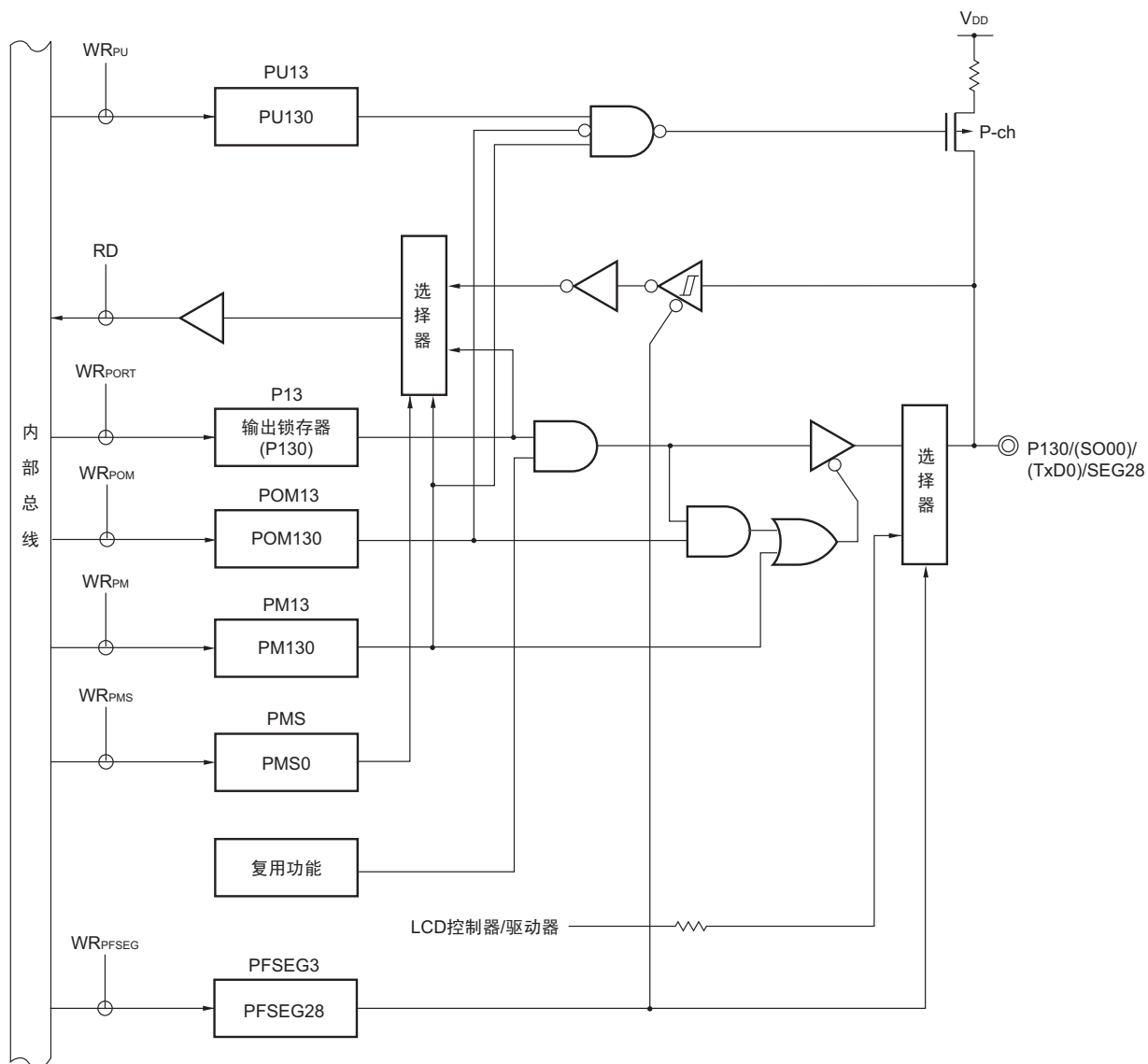
名称	输入 / 输出	PM13x	POM13x	PFSEGxx	复用功能设定 注 2	备注
P130	输入	1	×	0	×	
	输出	0	0	0	(SO00/TxD0 输出 = 1 注 1)	CMOS 输出
		0	1			N 沟道漏极开路输出
P137	输入	—	—	—	×	

注 1. 如果设定 PIOR1=1 时 P130 用作通用端口，必须将串行通道允许状态寄存器 0（SE0）、串行输出寄存器 0（SO0）和串行输出允许寄存器 0（SOE0）设定为与其初始状态相同的值。
2. （ ）中的是 PIORx = 1 时的情况。

备注 ×: 忽略
PM13: 端口模式寄存器 13
POM13x: 端口输出模式寄存器 13
PFSEGxx: LCD 端口功能寄存器
PIORx: 外围 I/O 重定向寄存器

端口 13 的框图如图 4-33 和图 4-34 所示。

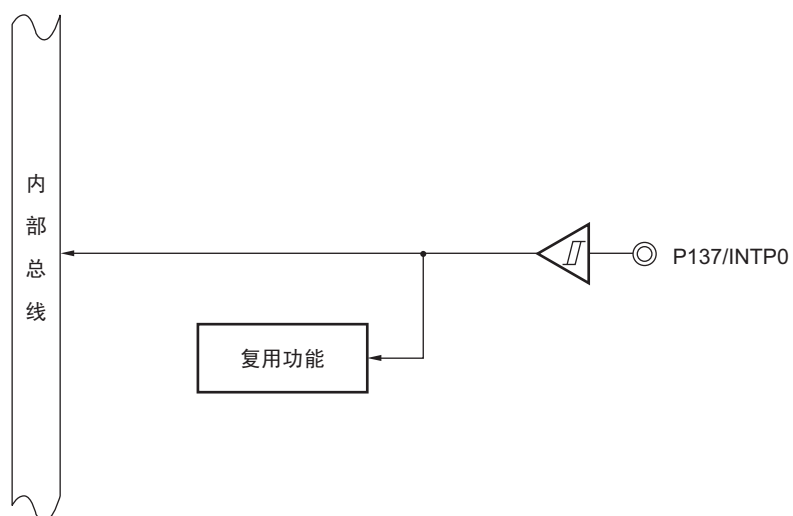
图 4-33 P130 的框图



- P13: 端口寄存器 13
 PU13: 上拉电阻选择寄存器 13
 PM13: 端口模式寄存器 13
 POM13: 端口输出模式寄存器 13
 PMS: 端口模式选择寄存器
 PFSEG3: LCD 端口功能寄存器 3
 RD: 读信号
 WRxx: 写信号

备注 可通过设定外围 I/O 重定向寄存器 (PIOR)，分配上图 () 内的功能。详细内容请参照“图 4-42 外围 I/O 重定向寄存器 (PIOR) 的格式”。

图 4-34 P137 的框图



4.3 控制端口功能的寄存器

使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 I/O 重定向寄存器 (PIOR)
- LCD 端口功能寄存器 (PFSEG0 ~ PFSEG6)
- LCD 输入切换控制寄存器 (ISCLCD)

注意 配置的寄存器和位因产品而不同。有关各产品配置的寄存器和位，请参照表 4-14。另外，必须给未配置的位设定初始值。

表 4-14 各产品配置的 PMxx、Pxx、PUxx、PIMxx、POMxx、PMCxx 寄存器及其位 (1/3)

端口		位名称					
		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	PM00	P00	PU00	—	POM00	—
	1	PM01	P01	PU01	—	—	—
	2	PM02	P02	PU02	—	—	—
	3	PM03	P03	PU03	PIM03	—	—
	4	PM04	P04	PU04	—	POM04	—
	5	PM05	P05	PU05	PIM05	POM05	—
	6	PM06	P06	PU06	PIM06	POM06	—
	7	PM07	P07	PU07	—	POM07	—
端口 1	0	PM10	P10	PU10	—	—	PMC10
	1	PM11	P11	PU11	—	—	PMC11
	2	PM12	P12	PU12	—	—	PMC12
	3	PM13	P13	PU13	—	—	PMC13
	4	PM14	P14	PU14	—	—	—
	5	PM15	P15	PU15	—	—	—
	6	PM16	P16	PU16	PIM16	POM16	—
	7	PM17	P17	PU17	PIM17	POM17	—
端口 2	0	PM20	P20	—	—	—	—
	1	PM21	P21	—	—	—	—
	2	PM22	P22	PU22	—	—	PMC22
	3	PM23	P23	PU23	—	—	PMC23
	4	PM24	P24	PU24	—	—	PMC24
	5	PM25	P25	PU25	—	—	PMC25
	6	PM26	P26	PU26	—	—	PMC26
	7	PM27	P27	PU27	—	—	PMC27
端口 3	0	PM30	P30	PU30	—	—	—
	1	PM31	P31	PU31	—	—	—
	2	PM32	P32	PU32	—	—	—
	3	PM33	P33	PU33	—	—	—
	4	PM34	P34	PU34	PIM34	—	—
	5	PM35	P35	PU35	—	POM35	—
	6	—	—	—	—	—	—
	7	—	—	—	—	—	—

表 4-14 各产品配置的 PMxx、Pxx、PUxx、PIMxx、POMxx、PMCxx 寄存器及其位 (2/3)

端口		位名称					
		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 4	0	PM40	P40	PU40	—	—	—
	1	PM41	P41	PU41	—	—	—
	2	PM42	P42	PU42	—	POM42	PMC42
	3	PM43	P43	PU43	PIM43	POM43	PMC43
	4	PM44	P44	PU44	PIM44	POM44	PMC44
	5	PM45	P45	PU45	—	—	PMC45
	6	PM46	P46	PU46	PIM46	POM46	—
	7	PM47	P47	PU47	PIM47	POM47	—
端口 5	0	PM50	P50	PU50	—	—	—
	1	PM51	P51	PU51	—	—	—
	2	PM52	P52	PU52	—	—	—
	3	PM53	P53	PU53	PIM53	POM53	—
	4	PM54	P54	PU54	—	POM54	—
	5	PM55	P55	PU55	PIM55	POM55	—
	6	PM56	P56	PU56	—	POM56	—
	7	PM57	P57	PU57	—	—	—
端口 6	0	PM60	P60	—	—	—	—
	1	PM61	P61	—	—	—	—
	2	—	—	—	—	—	—
	3	—	—	—	—	—	—
	4	—	—	—	—	—	—
	5	—	—	—	—	—	—
	6	—	—	—	—	—	—
	7	—	—	—	—	—	—
端口 7	0	PM70	P70	PU70	—	—	—
	1	PM71	P71	PU71	—	—	—
	2	PM72	P72	PU72	—	—	—
	3	PM73	P73	PU73	—	—	—
	4	PM74	P74	PU74	—	—	—
	5	PM75	P75	PU75	—	—	—
	6	PM76	P76	PU76	—	—	—
	7	PM77	P77	PU77	—	—	—

表 4-14 各产品配置的 PMxx、Pxx、PUxx、PIMxx、POMxx、PMCxx 寄存器及其位 (3/3)

端口		位名称					
		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 12	0	—	—	—	—	—	—
	1	—	P121	—	—	—	—
	2	—	P122	—	—	—	—
	3	—	P123	—	—	—	—
	4	—	P124	—	—	—	—
	5	PM125	P125	PU125	—	—	—
	6	PM126	P126	PU126	—	—	—
	7	PM127	P127	PU127	—	—	—
端口 13	0	PM130	P130	PU130	—	POM130	—
	1	—	—	—	—	—	—
	2	—	—	—	—	—	—
	3	—	—	—	—	—	—
	4	—	—	—	—	—	—
	5	—	—	—	—	—	—
	6	—	—	—	—	—	—
	7	—	P137	—	—	—	—

4.3.1 端口模式寄存器（PMxx）

这是以 1 位为单位指定端口的输入或者输出的寄存器。
通过 1 位或者 8 位存储器操作指令设定 PMxx 寄存器。
在产生复位信号后，这些寄存器的值变为“FFH”。
当将端口引脚用作复用功能的引脚时，必须参照“4.5 使用复用功能时的寄存器设定”进行设定。

图 4-35 端口模式寄存器的格式

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PM0	PM07	PM06	PM05	PM04	PM03	PM02	PM01	PM00	FFF20H	FFH	R/W
PM1	PM17	PM16	PM15	PM14	PM13	PM12	PM11	PM10	FFF21H	FFH	R/W
PM2	PM27	PM26	PM25	PM24	PM23	PM22	PM21	PM20	FFF22H	FFH	R/W
PM3	1	1	PM35	PM34	PM33	PM32	PM31	PM30	FFF23H	FFH	R/W
PM4	PM47	PM46	PM45	PM44	PM43	PM42	PM41	PM40	FFF24H	FFH	R/W
PM5	PM57	PM56	PM55	PM54	PM53	PM52	PM51	PM50	FFF25H	FFH	R/W
PM6	1	1	1	1	1	1	PM61	PM60	FFF26H	FFH	R/W
PM7	PM77	PM76	PM75	PM74	PM73	PM72	PM71	PM70	FFF27H	FFH	R/W
PM12	PM127	PM126	PM125	1	1	1	1	1	FFF2CH	FFH	R/W
PM13	1	1	1	1	1	1	1	PM130	FFF2DH	FFH	R/W
PMmn	Pmn 引脚输入 / 输出模式的选择 (m = 0 ~ 7、12、13 ; n = 0 ~ 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 必须给未配置的位设定初始值。

4.3.2 端口寄存器 (Pxx)

这是设定端口输出锁存器的值的寄存器。
 读时，在输入模式中读引脚电平，而在输出模式中读端口的输出锁存器的值。
 通过 1 位或者 8 位存储器操作指令设定 Pxx 寄存器。
 在产生复位信号后，这些寄存器的值变为“00H”。

注 在将 P10 ~ P13、P20 ~ P23 设定为 A/D 转换器的模拟输入功能或者将 P42 ~ P45 设定为比较器的模拟输入功能的情况下，如果在输入模式时读端口，读取值就不是引脚电平而总是“0”。

图 4-36 端口寄存器的格式

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
P0	P07	P06	P05	P04	P03	P02	P01	P00	FFF00H	00H (输出锁存器)	R/W
P1	P17	P16	P15	P14	P13	P12	P11	P10	FFF01H	00H (输出锁存器)	R/W
P2	P27	P26	P25	P24	P23	P22	P21	P20	FFF02H	00H (输出锁存器)	R/W
P3	0	0	P35	P34	P33	P32	P31	P30	FFF03H	00H (输出锁存器)	R/W
P4	P47	P46	P45	P44	P43	P42	P41	P40	FFF04H	00H (输出锁存器)	R/W
P5	P57	P56	P55	P54	P53	P52	P51	P50	FFF05H	00H (输出锁存器)	R/W
P6	0	0	0	0	0	0	P61	P60	FFF06H	00H (输出锁存器)	R/W
P7	P77	P76	P75	P74	P73	P72	P71	P70	FFF07H	00H (输出锁存器)	R/W
P12	P127	P126	P125	P124	P123	P122	P121	0	FFF0CH	不定	R/W 注 1
P13	P137	0	0	0	0	0	0	P130	FFF0DH	注 2	R/W 注 1

Pmn	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 ~ P124 和 P137 为只读位。
 2. P137: 不定
 P130: 0 (输出锁存器)

注意 必须给未配置的位设定初始值。

备注 m = 0 ~ 7、12、13 ; n = 0 ~ 7

4.3.3 上拉电阻选择寄存器（PUxx）

这是设定是否使用内部上拉电阻的寄存器。只能对通过上拉电阻选择寄存器指定为使用内部上拉电阻的引脚并且设定为通常输出模式（POMmn=0）而且设定为输入模式（PMmn=1）的位，以位为单位使用内部上拉电阻。对于设定为输出模式的位，与上拉电阻选择寄存器的设定无关，不连接内部上拉电阻。当用作复用功能的输出引脚或者设定为模拟功能（PMC=1、ADPC=1）时也相同。

通过 1 位或者 8 位存储器操作指令设定 PUxx 寄存器。
在产生复位信号后，这些寄存器的值变为“00H”（只有 PU4 为“01H”）。

注意 如果要带 PIMn 寄存器的端口从不同电位器件设定为 TTL 缓冲器时，必须将 PUm_n 位置“0”，并且通过外部电阻上拉到不同电位器件的电源。

图 4-37 上拉电阻选择寄存器的格式

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PU0	PU07	PU06	PU05	PU04	PU03	PU02	PU01	PU00	F0030H	00H	R/W
PU1	PU17	PU16	PU15	PU14	PU13	PU12	PU11	PU10	F0031H	00H	R/W
PU2	PU27	PU26	PU25	PU24	PU23	PU22	0	0	F0032H	00H	R/W
PU3	0	0	PU35	PU34	PU33	PU32	PU31	PU30	F0033H	00H	R/W
PU4	PU47	PU46	PU45	PU44	PU43	PU42	PU41	PU40	F0034H	01H	R/W
PU5	PU57	PU56	PU55	PU54	PU53	PU52	PU51	PU50	F0035H	00H	R/W
PU7	PU77	PU76	PU75	PU74	PU73	PU72	PU71	PU70	F0037H	00H	R/W
PU12	PU127	PU126	PU125	0	0	0	0	0	F003CH	00H	R/W
PU13	0	0	0	0	0	0	0	PU130	F003DH	00H	R/W
PUmn	Pmn 引脚内部上拉电阻的选择 (m = 0 ~ 5、7、12、13 ; n = 0 ~ 7)										
0	不连接内部上拉电阻										
1	连接内部上拉电阻										

注 必须给未配置的位设定初始值。

4.3.4 端口输入模式寄存器（PIMxx）

这是以位为单位设定输入缓冲器的寄存器。
能在和不同电位的外部器件进行串行通信等时选择 TTL 输入缓冲器。
通过 1 位或者 8 位存储器操作指令设定 PIMxx 寄存器。
在产生复位信号后，这些寄存器的值变为“00H”。

图 4-38 端口输入模式寄存器的格式

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PIM0	0	PIM06	PIM05	0	PIM03	0	0	0	F0040H	00H	R/W
PIM1	PIM17	PIM16	0	0	0	0	0	0	F0041H	00H	R/W
PIM3	0	0	0	PIM34	0	0	0	0	F0043H	00H	R/W
PIM4	PIM47	PIM46	0	PIM44	PIM43	0	0	0	F0044H	00H	R/W
PIM5	0	0	PIM55	0	PIM53	0	0	0	F0045H	00H	R/W
PIMmn	Pmn 引脚的输入缓冲器的选择 (m = 0、1、3 ~ 5 ; n = 3 ~ 7)										
0	通常输入缓冲器										
1	TTL 输入缓冲器										

注意 必须给未配置的位设定初始值。

4.3.5 端口输出模式寄存器（POMxx）

这是以位为单位设定输出模式的寄存器。

在和不同电位的外部器件进行串行通信以及和同电位的外部器件进行简易 I²C 通信时，SDA00、SDA10 引脚能选择 N 沟道漏极开路输出（V_{DD} 耐压）模式。

另外，通过 POMxx 寄存器和 PUxx 寄存器设定是否使用内部上拉电阻。

通过 1 位或者 8 位存储器操作指令设定 POMxx 寄存器。

在产生复位信号后，这些寄存器的值变为“00H”。

注意 设定为 N 沟道漏极开路输出（V_{DD} 耐压）模式（POMmn=1）的位，不能连接内部上拉电阻。

图 4-39 端口输出模式寄存器的格式

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
POM0	POM07	POM06	POM05	POM04	0	0	0	POM00	F0050H	00H	R/W
POM1	POM17	POM16	0	0	0	0	0	0	F0051H	00H	R/W
POM3	0	0	POM35	0	0	0	0	0	F0053H	00H	R/W
POM4	POM47	POM46	0	POM44	POM43	POM42	0	0	F0054H	00H	R/W
POM5	0	POM56	POM55	POM54	POM53	0	0	0	F0055H	00H	R/W
POM13	0	0	0	0	0	0	0	POM130	F005DH	00H	R/W
POMmn	Pmn 引脚输出模式的选择 (m = 0、1、3 ~ 5、13 ; n = 0、2 ~ 7)										
0	通常输出模式										
1	N 沟道漏极开路输出（V _{DD} 耐压）模式										

4.3.6 端口模式控制寄存器（PMCxx）

这是以位为单位设定数字输入 / 输出或者模拟输入、数字输入 / 输出或者数字输入无效的寄存器。

通过 1 位或者 8 位存储器操作指令设定 PMCxx 寄存器。

在产生复位信号后，PMC1、PMC2 的值变为“FFH”，PMC4 的值变为“00H”。

图 4-40 端口模式控制寄存器的格式

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PMC1	1	1	1	1	PMC13	PMC12	PMC11	PMC10	F0061H	FFH	R/W
PMC2	PMC27	PMC26	PMC25	PMC24	PMC23	PMC22	1	1	F0062H	FFH	R/W
PMC4	0	0	PMC45	PMC44	PMC43	PMC42	0	0	F0064H	00H	R/W

PMCmn	选择 Pmn 引脚数字输入 / 输出或者模拟输入 (mn = 22、23、42 ~ 45)
0	数字输入 / 输出（模拟输入以外的复用功能）
1	模拟输入

PMCmn	选择 Pmn 引脚数字输入 / 输出或者数字输入无效 (mn = 10 ~ 13、24 ~ 27)
0	数字输入 / 输出（复用功能）
1	数字输入无效

- 注意 1. 必须通过端口模式寄存器 1、2、4（PM1、PM2、PM4）将通过 PMCxx 寄存器设定为模拟输入的端口设定为输入模式。
2. 对于由 PMCxx 寄存器设定为数字输入 / 输出的引脚，不能通过模拟输入通道指定寄存器（ADS）进行设定。
3. 当 P10 ~ P13、P24 ~ P27 引脚用作数字输入 / 输出或者 LCD 输出时，必须将 PMC10 ~ PMC13、PMC24 ~ PMC27 设定为“0”。
4. 数字输入无效是指数字输出、数字输入、模拟输入以及 LCD 输出均处于无效的状态。
5. 必须给未配置的位设定初始值。

4.3.7 A/D 端口配置寄存器（ADPC）

这是将 ANI0/P21、ANI1/P20 引脚切换为 A/D 转换器的模拟输入或者端口的数字输入 / 输出的寄存器。
通过 8 位存储器操作指令设定 ADPC 寄存器。
在产生复位信号后，此寄存器的值变为“00H”。

图 4-41 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	0	0	ADPC1	ADPC0

ADPC1	ADPC0	模拟输入（A）和数字输入 / 输出（D）的切换	
		ANI1/P20	ANI0/P21
0	0	A	A
0	1	D	D
1	0	D	A
1	1	禁止设定	

- 注意 1. 对于由 ADPC 寄存器设定为模拟输入的端口，必须通过端口模式寄存器 2（PM2）将其设定为输入模式。
2. 对于由 ADPC 寄存器设定为数字输入 / 输出的引脚，不能通过模拟输入通道指定寄存器（ADS）进行设定。
3. 如果要使用 AV_{REFP} 和 AV_{REFM} ，必须将 ANI0 和 ANI1 设定为模拟输入，并且通过端口模式寄存器设定为输入模式。

4.3.8 外围 I/O 重定向寄存器 (PIOR)

这是设定允许或者禁止外围 I/O 重定向功能的寄存器。

外围 I/O 重定向功能切换被分配了复用功能的端口。

必须通过 PIOR 寄存器将端口分配给重定向功能，然后置为允许重定向功能的运行。

能在允许重定向功能的运行前更改重定向的设定。

通过 8 位存储器操作指令设定 PIOR 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 4-42 外围 I/O 重定向寄存器 (PIOR) 的格式

地址: F0077H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
PIOR	0	0	0	PIOR4	PIOR3	PIOR2	PIOR1	PIOR0

位	功能	设定值	
		0	1
PIOR4	INTP7	P02	P43
	INTP5	P55	P01
PIOR3	PCLBUZ0	P02	P07
PIOR2	TxD1	P07	P42
	RxD1	P06	P43
	SCL10	P05	P44
	SDA10	P06	P43
PIOR1	TxD0	P00	P130
	RxD0	P17	P47
	SCL00	P16	P46
	SDA00	P17	P47
	SI00	P17	P47
	SO00	P00	P130
	SCK00	P16	P46
PIOR0	TI00/TO00	P52	P40
	TI01/TO01	P32	P60
	TI02/TO02	P54	P61
	TI03/TO03	P30	P127
	TI04/TO04	P14	P126
	TI05/TO05	P42	P01
	TI06/TO06	P56	P125
	TI07/TO07	P15	P41

4.3.9 LCD 端口功能寄存器 0～6（PFSEG0～PFSEG6）

这些是设定将 P00～P07、P10～P17、P22～P27、P30～P35、P46、P47、P50～P57、P70～P77、P130 引脚用作端口（段输出除外）或者段输出，以及将 P45 引脚选择用作端口或模拟输入的寄存器。

通过 1 位或者 8 位存储器操作指令设定 PFSEG0～PFSEG6 寄存器。

在产生复位信号后，这些寄存器的值变为“FFH”（PFSEG0 为“F0H”，PFSEG6 为“0FH”）。

备注 段输出引脚（SEGxx）与 PFSEG 寄存器（PFSEGxx 位）的对应以及各产品中是否配置 SEGxx 引脚如“表 4-15 与段输出引脚对应的 PFSEG 寄存器（PFSEG 位）”。

图 4-43 LCD 端口功能寄存器（PFSEG0～PFSEG6）的格式

地址：F0300H 复位后：F0H R/W

符号	7	6	5	4	3	2	1	0
PFSEG0	PFSEG07	PFSEG06	PFSEG05	PFSEG04	0	0	0	0

地址：F0301H 复位后：FFH R/W

符号	7	6	5	4	3	2	1	0
PFSEG1	PFSEG15	PFSEG14	PFSEG13	PFSEG12	PFSEG11	PFSEG10	PFSEG09	PFSEG08

地址：F0302H 复位后：FFH R/W

符号	7	6	5	4	3	2	1	0
PFSEG2	PFSEG23	PFSEG22	PFSEG21	PFSEG20	PFSEG19	PFSEG18	PFSEG17	PFSEG16

地址：F0303H 复位后：FFH R/W

符号	7	6	5	4	3	2	1	0
PFSEG3	PFSEG30	PFSEG29	PFSEG28	PFSEG27	PFSEG26	PFDEG	PFSEG25	PFSEG24

地址：F0304H 复位后：FFH R/W

符号	7	6	5	4	3	2	1	0
PFSEG4	PFSEG38	PFSEG37	PFSEG36	PFSEG35	PFSEG34	PFSEG33	PFSEG32	PFSEG31

地址：F0305H 复位后：FFH R/W

符号	7	6	5	4	3	2	1	0
PFSEG5	PFSEG46	PFSEG45	PFSEG44	PFSEG43	PFSEG42	PFSEG41	PFSEG40	PFSEG39

地址：F0306H 复位后：0FH R/W

符号	7	6	5	4	3	2	1	0
PFSEG6	0	0	0	0	PFSEG50	PFSEG49	PFSEG48	PFSEG47

PFSEGxx (xx = 04～50)	Pmn 引脚的端口（非段输出）/ 段输出设定 (mn = 00～07、10～17、22～27、30～35、46、47、50～57、70～77、130)
0	用作端口（非段输出）
1	用作段输出

PFDEG	P45 引脚的端口 / 模拟输入（IVREF0）的指定
0	用作端口
1	用作模拟输入（IVREF0）

注意 必须给未配置的位设定初始值。

备注 用作段输出引脚（PFSEGxx = 1）时，必须将 PUm 寄存器的 PUm_n 位、POMm 寄存器的 POM_{mn} 位以及 PIMm 寄存器的 PIM_{mn} 位设定为“0”。

表 4-15 与段输出引脚对应的 PFSEG 寄存器 (PFSEG 位)

PFSEG 寄存器的位名称	对应的 SEGxx 引脚	复用端口
PFSEG04	SEG4	P50
PFSEG05	SEG5	P51
PFSEG06	SEG6	P52
PFSEG07	SEG7	P53
PFSEG08	SEG8	P54
PFSEG09	SEG9	P55
PFSEG10	SEG10	P56
PFSEG11	SEG11	P57
PFSEG12	SEG12	P70
PFSEG13	SEG13	P71
PFSEG14	SEG14	P72
PFSEG15	SEG15	P73
PFSEG16	SEG16	P74
PFSEG17	SEG17	P75
PFSEG18	SEG18	P76
PFSEG19	SEG19	P77
PFSEG20	SEG20	P30
PFSEG21	SEG21	P31
PFSEG22	SEG22	P32
PFSEG23	SEG23	P33
PFSEG24	SEG24	P34
PFSEG25	SEG25	P35
PFSEG26	SEG26	P46
PFSEG27	SEG27	P47
PFSEG28	SEG28	P130
PFSEG29	SEG29	P22
PFSEG30	SEG30	P23
PFSEG31	SEG31	P24
PFSEG32	SEG32	P25
PFSEG33	SEG33	P26
PFSEG34	SEG34	P27
PFSEG35	SEG35	P10
PFSEG36	SEG36	P11
PFSEG37	SEG37	P12
PFSEG38	SEG38	P13
PFSEG39	SEG39	P14
PFSEG40	SEG40	P15
PFSEG41	SEG41	P16
PFSEG42	SEG42	P17
PFSEG43	SEG43	P00
PFSEG44	SEG44	P01
PFSEG45	SEG45	P02
PFSEG46	SEG46	P03
PFSEG47	SEG47	P04
PFSEG48	SEG48	P05
PFSEG49	SEG49	P06
PFSEG50	SEG50	P07

4.3.10 LCD 输入切换控制寄存器 (ISCLCD)

这是设定将 P125 ~ P127 引脚用作端口 (LCD 功能除外) 或者 LCD 功能 (V_{L3} 、CAPL、CAPH) 的寄存器。

通过 1 位或者 8 位存储器操作指令设定 ISCLCD 寄存器。

在产生复位信号后, 这些寄存器的值为“00H”。

图 4-44 LCD 输入切换控制寄存器 (ISCLCD) 的格式

地址: F0308H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ISCLCD	0	0	0	0	0	0	ISCVL3	ISCCAP

ISCVL3	V_{L3} /P125 引脚的施密特触发缓冲器的控制
0	数字输入无效 (用作 LCD 功能 (V_{L3}))
1	数字输入有效

ISCCAP	CAPL/ P126 和 CAPH/P127 引脚的施密特触发缓冲器的控制
0	数字输入无效 (用作 LCD 功能 (CAPL、CAPH))
1	数字输入有效

注意 当 ISCVL3=0、ISCCAP=0 时, 必须如下设定对应的端口控制寄存器。

PU12 寄存器的 PU127=0, P12 寄存器的 P127=0

PU12 寄存器的 PU126=0, P12 寄存器的 P126=0

PU12 寄存器的 PU125=0, P12 寄存器的 P125=0

4.4 端口功能的运行

如下所示，端口的运行因输入 / 输出模式的设定而不同。

4.4.1 输入 / 输出端口的写操作

(1) 输出模式的情况

能通过传送指令给输出锁存器写值，并且从引脚输出输出锁存器的内容。

保持被写在输出锁存器中的数据，直到下次写数据为止。

在产生复位信号时清除输出锁存器的数据。

(2) 输入模式的情况

通过传送指令，给输出锁存器写值，但是因输出缓冲器处于 OFF 状态而引脚的状态不发生变化。因此，能对同时具有输入和输出功能的端口进行字节写操作。

保持被写在输出锁存器中的数据，直到下次写数据为止。

在产生复位信号时清除输出锁存器的数据。

4.4.2 输入 / 输出端口的读操作

(1) 输出模式的情况

能通过传送指令读输出锁存器的内容，但是输出锁存器的内容不发生变化。

(2) 输入模式的情况

能通过传送指令读引脚状态，但是输出锁存器的内容不发生变化。

4.4.3 输入 / 输出端口的运算

(1) 输出模式的情况

对输出锁存器的内容进行运算，将结果写到输出锁存器，并且从引脚输出输出锁存器的内容。

保持被写在输出锁存器中的数据，直到下次写数据为止。

在产生复位信号时清除输出锁存器的数据。

(2) 输入模式的情况

读引脚电平并且对其内容进行运算，将运算结果写到输出锁存器，但是因输出缓冲器处于 OFF 状态而引脚的状态不发生变化。因此，能对同时具有输入和输出功能的端口进行字节写操作。

在产生复位信号时清除输出锁存器的数据。

4.4.4 不同电位（1.8V、2.5V、3V）的输入 / 输出缓冲器对应

能通过端口输入模式寄存器（PIMxx）和端口输出模式寄存器（POMxx）切换输入 / 输出缓冲器，连接以不同电位（1.8V、2.5V、3V）运行的外部器件。

在接受不同电位（1.8V、2.5V、3V）的外部器件的输入时，能通过按位设定端口输入模式寄存器 0、1、3、4、5（PIM0、PIM1、PIM3、PIM4、PIM5），切换通常输入（CMOS）/TTL 输入缓冲器。

在向不同电位（1.8V、2.5V、3V）的外部器件进行输出时，能通过按位设定端口输出模式寄存器 0、1、3、4、5、13（POM0、POM1、PIM3、POM4、POM5、POM13），切换通常输出（COMS）/N 沟道漏极开路（V_{DD} 耐压）。

以下说明串行接口的连接。

(1) 将 UART0 ~ UART3、CSI00 功能的输入端口用作 TTL 输入缓冲器时的设定步骤

UART0 的情况：P17(P47)

UART1 的情况：P06(P43)

UART2 的情况：P03

UART3 的情况：P34

CSI00 的情况： P17、P16(P47、P46)

备注 能通过设定外围 I/O 重定向寄存器（PIOR），分配上述（）内的功能。

- ① 通过外部电阻将使用的输入引脚上拉到对象器件的电源（不能使用内部上拉电阻）。
- ② 将 PIM0、PIM1、PIM3、PIM4 寄存器的对应位置“1”，切换到 TTL 输入缓冲器。另外，V_{IH} 和 V_{IL} 参照选择 TTL 输入缓冲器时的 DC 特性。
- ③ 允许串行阵列单元运行，并且设定为 UART/CSI 模式。

(2) 将 UART0 ~ UART3、CSI00 功能的输出端口用作 N 沟道漏极开路输出时的设定步骤

UART0 的情况：P00(P130)

UART1 的情况：P07(P42)

UART2 的情况：P04

UART3 的情况：P35

CSI00 的情况： P10、P16(P130、P46)

备注 能通过设定外围 I/O 重定向寄存器（PIOR），分配上述（）内的功能。

- ① 通过外部电阻将使用的输出引脚上拉到对象器件的电源（不能使用内部上拉电阻）。
- ② 在解除复位后，端口模式为输入模式（Hi-Z）。
- ③ 将对应的端口输出锁存器置“1”。
- ④ 将 POM0、POM1、POM3、POM4、POM13 寄存器的对应位置“1”，设定为 N 沟道漏极开路输出（V_{DD} 耐压）模式。
- ⑤ 允许串行阵列单元运行，并且设定为 UART/CSI 模式。
- ⑥ 通过 PM0、PM1、PM3、PM4、PM13 寄存器设定为输出模式。此时，因为输出数据是高电平，所以引脚处于 Hi-Z 状态。

(3) 将 IIC00、IIC10 功能的输入 / 输出端口用作不同电位（1.8V、2.5V、3V）时的设定步骤

IIC00的情况： P16、P17(P46、P47)

IIC10的情况： P05、P06(P44、P43)

备注 能通过设定外围 I/O 重定向寄存器（PIOR），分配上述（）内的功能。

- ① 通过外部电阻将使用的输入引脚上拉到对象器件的电源（不能使用内部上拉电阻）。
- ② 在解除复位后，端口模式为输入模式（Hi-Z）。
- ③ 将对应的端口输出锁存器置“1”。
- ④ 将POM0、POM1、POM4寄存器的对应位置“1”，设定为N沟道漏极开路输出（ V_{DD} 耐压注1/ EV_{DD} 耐压注2）模式。
- ⑤ 将PIM0、PIM1、PIM4寄存器的对应位置“1”，切换到TTL输入缓冲器。另外， V_{IH} 和 V_{IL} 参照选择TTL输入缓冲器时的DC特性。
- ⑥ 允许串行阵列单元运行，并且设定为简易I²C模式。
- ⑦ 将PM0、PM1、PM4寄存器的对应位设定为输出模式（能在输出模式中进行数据的输入/输出）。此时，因为输出数据是高电平，所以引脚处于Hi-Z状态。

4.5 使用复用功能时的寄存器设定

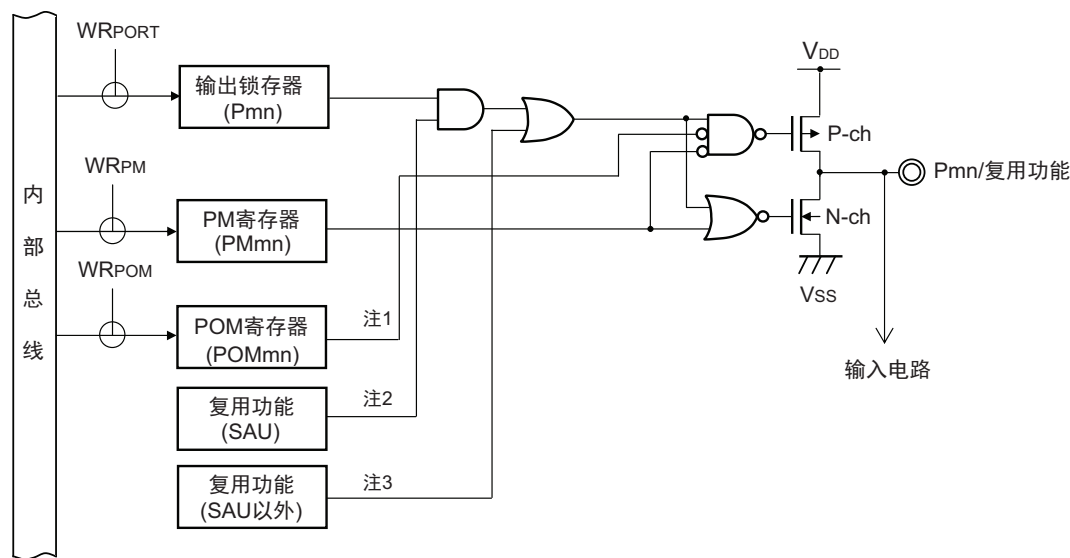
4.5.1 使用复用功能时的基本观点

通过 A/D 端口配置寄存器（ADPC）设定是将与模拟输入复用的引脚用作模拟输入还是用作数字输入 / 输出。

通过 LCD 端口功能寄存器（PFSEGx）和端口模式控制寄存器（PMCxx）设定是将与模拟输入和段输出复用的引脚用作模拟输入 / 段输出或者数字输入 / 输出。

用作数字输入 / 输出的引脚的输出电路的基本结构如图 4-45 所示。与端口输出锁存器的输出复用的 SAU 功能的输出被输入到 AND 门。AND 门的输出被输入到 OR 门。OR 门的其他输入与复用的 SAU 以外的功能（TAU、RTC2、时钟 / 蜂鸣器的输出、IICA 等）的输出连接。将这样的引脚用作端口功能或者复用功能时，必须注意不使用的复用功能不能影响要使用的功能的输出。此时的基本设定的观点如表 4-16 所示。

图 4-45 引脚输出电路的基本结构



- 注 1. 在无 POM 寄存器时，此信号为 Low 电平（0）。
2. 在无复用功能时，此信号为 High 电平（1）。
3. 在无复用功能时，此信号为 Low 电平（0）。

备注 m: 端口号（m=0 ~ 17、12、13）、n: 位号（n=0 ~ 7）

表 4-16 基本设定的观点

使用引脚的输出功能	不使用的复用功能的输出设定		
	端口功能	SAU 的输出功能	SAU 以外的输出功能
端口输出功能	—	High 电平输出（1）	Low 电平输出（0）
SAU 的输出功能	High（1）	—	Low 电平输出（0）
SAU 以外的输出功能	Low（0）	High 电平输出（1）	Low 电平输出（0）注

注 因为 1 个引脚有可能复用多个 SAU 以外的输出功能，所以必须将不使用的复用功能的输出设定为 Low 电平（0）。有关详细设定方法请参照“4.5.2 不使用输出功能的复用功能的寄存器设定”。

4.5.2 不使用输出功能的复用功能的寄存器设定

不使用引脚复用功能的输出时，必须执行以下设定。另外，在为外围 I/O 重定向功能的对象时，能通过设定外围 I/O 重定向寄存器（PIOR），将输出切换到其他引脚。这样就能使用分配到对象引脚的端口功能和其他复用功能。

1. $SOP=1/TxDq=1$ （不使用SAU的串行输出（SOP/TxDq）时的设定）
只使用SAU的串行输入，不使用串行输出（SOP/TxDq）时，将与不使用的输出对应的串行输出允许寄存器m（SOEm）的位置“0”（禁止输出），并且将串行输出寄存器m（SOM）的SOMn位置“1”（High电平）。这是与初始状态相同的设定。
2. $SCKp=1/SDAr=1/SCLr=1$ （不使用SAU的通道n时的设定）
不使用SAU时，将串行通道允许状态寄存器m（SEm）的bit n（SEmn）置“0”（运行停止状态），并且将与不使用的输出对应的串行输出允许寄存器m（SOEm）的位置“0”（禁止输出），串行输出寄存器m（SOM）的SOMn位和CKOMn位置“1”（High电平）。这是与初始状态相同的设定。
3. $TOmn=0$ （不使用TAU的通道n的输出时的设定）
不使用TAU的TOmn输出时，将与不使用的输出对应的定时器输出允许寄存器0（TOE0）的位置“0”（禁止输出），并且将定时器输出寄存器0（TO0）的位置“0”（Low电平）。这是与初始状态相同的设定。
4. $SDAAn=0/SCLAn=0$ （不使用IICA时的设定）
不使用IICA时，将IICA控制寄存器n0（IICCTLn0）的IICEn位置“0”（运行停止）。这是与初始状态相同的设定。
5. $PCLBUZn=0$ （不使用时钟输出/蜂鸣器输出时的设定）
不使用时钟输出/蜂鸣器输出时，将时钟输出选择寄存器n（CKSn）的PCLOEn位置“0”（禁止输出）。这是与初始状态相同的设定。
6. $VCOUTn=0$ （不使用VCOUTn时的设定）
不使用比较器的VCOUTn时，将比较器输出控制寄存器（COMPOCR）的bit5和bit1置“0”（禁止比较器的VCOUTn引脚输出）。这是与初始状态相同的设定。

4.5.3 使用的端口功能和复用功能的寄存器设定例子

使用的端口功能和复用功能的寄存器设定例子如表 4-17 所示。端口功能的控制寄存器的设定如表 4-17 所示。另外，有关表 4-17 的中的记载说明请参照以下备注。

备注	—:	对象外
	x:	忽略
	PIORx:	外围 I/O 重定向寄存器
	POMxx:	端口输出模式寄存器
	PMCxx:	端口模式控制寄存器
	PMxx:	端口模式寄存器
	Pxx:	端口的输出锁存器
	PFSEGxx、PFDEG:	LCD 端口功能寄存器
	能通过设定外围 I/O 重定向寄存器（PIOR），分配 () 内的功能。	

表 4-17 使用复用功能时端口相关寄存器的设定 (1/7)

引脚名称	复用功能		PIORx	PFSEGxx	POMxx	PMCxx	PMxx	Pxx
	功能名称	输入 / 输出						
P00	SEG43	输出	×	PFSEG43 = 1	0	—	0	0
	SO00	输出	0	PFSEG43 = 0	0/1	—	0	1
	TxD0	输出	0	PFSEG43 = 0	0/1	—	0	1
	TOOLTxD	输出	×	PFSEG43 = 0	0/1	—	0	1
P01	(TI05)	输入	1	PFSEG44 = 0	—	—	1	×
	(TO05)	输出	1	PFSEG44 = 0	—	—	0	0
	(INTP5)	输入	1	PFSEG44 = 0	—	—	1	×
	PCLBUZ1	输出	×	PFSEG44 = 0	—	—	0	0
	SEG44	输出	×	PFSEG44 = 1	—	—	0	0
P02	INTP7	输入	0	PFSEG45 = 0	—	—	1	×
	PCLBUZ0	输出	0	PFSEG45 = 0	—	—	0	0
	SEG45	输出	×	PFSEG45 = 1	—	—	0	0
P03	RxD2	输入	×	PFSEG46 = 0	—	—	1	×
	SEG46	输出	×	PFSEG46 = 1	—	—	0	0
	VCOUT0	输出	×	PFSEG46 = 0	—	—	0	0
P04	TxD2	输出	×	PFSEG47 = 0	0/1	—	0	1
	SEG47	输出	×	PFSEG47 = 1	0	—	0	0
	VCOUT1	输出	×	PFSEG47 = 0	0	—	0	0
P05	SCL10	输出	0	PFSEG48 = 0	0/1	—	0	1
	SEG48	输出	×	PFSEG48 = 1	0	—	0	0
P06	RxD1	输入	0	PFSEG49 = 0	×	—	1	×
	SDA10	输入 / 输出	0	PFSEG49 = 0	1	—	0	1
	SEG49	输出	×	PFSEG49 = 1	0	—	0	0
P07	TxD1	输出	0	PFSEG50 = 0	0/1	—	0	1
	(PCLBUZ0)	输出	1	PFSEG50 = 0	0	—	0	0
	SEG50	输出	×	PFSEG50 = 1	0	—	0	0

备注 1. ×: 忽略

PIORx: 外围 I/O 重定向寄存器

PFSEGxx: LCD 端口功能寄存器

POMxx: 端口输出模式寄存器

PMCxx: 端口模式控制寄存器

PMxx: 端口模式寄存器

Pxx: 端口输出锁存器

2. 能通过设定外围 I/O 重定向寄存器 (PIOR)，分配 () 内的功能。

表 4-17 使用复用功能时端口相关寄存器的设定 (2/7)

引脚名称	复用功能		PIORx	PFSEGxx	POMxx	PMCxx	PMxx	Pxx
	功能名称	输入 / 输出						
P10	SEG35	输出	×	PFSEG35 = 1	—	0	0	0
P11	SEG36	输出	×	PFSEG36 = 1	—	0	0	0
P12	SEG37	输出	×	PFSEG37 = 1	—	0	0	0
P13	SEG38	输出	×	PFSEG38 = 1	—	0	0	0
P14	TI04	输入	0	PFSEG39 = 0	—	—	1	×
	TO04	输出	0	PFSEG39 = 0	—	—	0	0
	SEG39	输出	×	PFSEG39 = 1	—	—	0	0
P15	TI07	输入	0	PFSEG40 = 0	—	—	1	×
	TO07	输出	0	PFSEG40 = 0	—	—	0	0
	SEG40	输出	×	PFSEG40 = 1	—	—	0	0
P16	SEG41	输出	×	PFSEG41 = 1	0	—	0	0
	SCK00	输入	0	PFSEG41 = 0	×	—	1	×
		输出	0	PFSEG41 = 0	0/1	—	0	1
	SCL00	输出	0	PFSEG41 = 0	0/1	—	0	1
P17	SEG42	输出	×	PFSEG42 = 1	0	—	0	0
	SI00	输入	0	PFSEG42 = 0	×	—	1	×
	RxD0	输入	0	PFSEG42 = 0	×	—	1	×
	TOOLRxD	输入	×	PFSEG42 = 0	×	—	1	×
	SDA00	输入 / 输出	0	PFSEG42 = 0	1	—	0	1

备注 1. ×: 忽略

PIORx: 外围 I/O 重定向寄存器

PFSEGxx: LCD 端口功能寄存器

POMxx: 端口输出模式寄存器

PMCxx: 端口模式控制寄存器

PMxx: 端口模式寄存器

Pxx: 端口输出锁存器

2. 有关段输出引脚 (SEGxx) 的复用端口的详细内容, 请参照“4.5.4 SEGxx 引脚的复用端口的运行”。

表 4-17 使用复用功能时端口相关寄存器的设定 (3/7)

引脚名称	复用功能		PIORx	PFSEGxx	POMxx	PMCxx	PMxx	Pxx
	功能名称	输入 / 输出						
P20	ANI1 注 2	输入	×	—	—	—	1	×
	AV _{REFM} 注 2	输入	×	—	—	—	1	×
P21	ANI0 注 2	输入	×	—	—	—	1	×
	AV _{REFP} 注 2	输入	×	—	—	—	1	×
P22	ANI16 注 1	输入	×	PFSEG29 = 1	—	1	1	×
	SEG29	输出	×	PFSEG29 = 1	—	0	0	0
P23	ANI17 注 1	输入	×	PFSEG30 = 1	—	1	1	×
	SEG30	输出	×	PFSEG30 = 1	—	0	0	0
P24	SEG31	输出	×	PFSEG31 = 1	—	0	0	0
P25	SEG32	输出	×	PFSEG32 = 1	—	0	0	0
P26	SEG33	输出	×	PFSEG33 = 1	—	0	0	0
P27	SEG34	输出	×	PFSEG34 = 1	—	0	0	0
P30	TI03	输入	0	PFSEG20 = 0	—	—	1	×
	TO03	输出	0	PFSEG20 = 0	—	—	0	0
	SEG20	输出	×	PFSEG20 = 1	—	—	0	0
P31	INTP3	输入	×	PFSEG21 = 0	—	—	1	×
	RTC1HZ	输出	×	PFSEG21 = 0	—	—	0	0
	SEG21	输出	×	PFSEG21 = 1	—	—	0	0
P32	TI01	输入	0	PFSEG22 = 0	—	—	1	×
	TO01	输出	0	PFSEG22 = 0	—	—	0	0
	SEG22	输出	×	PFSEG22 = 1	—	—	0	0

备注 1. ×: 忽略

PIORx: 外围 I/O 重定向寄存器

PFSEGxx: LCD 端口功能寄存器

POMxx: 端口输出模式寄存器

PMCxx: 端口模式控制寄存器

PMxx: 端口模式寄存器

Pxx: 端口输出锁存器

2. 有关段输出引脚 (SEGxx) 的复用端口的详细内容, 请参照“4.5.4 SEGxx 引脚的复用端口的运行”。

表 4-17 使用复用功能时端口相关寄存器的设定 (4/7)

引脚名称	复用功能		PIORx	PFSEGxx、 PFDEG	POMxx	PMCxx	PMxx	Pxx
	功能名称	输入 / 输出						
P33	INTP4	输入	×	PFSEG23 = 0	—	—	1	×
	SEG23	输出	×	PFSEG23 = 1	—	—	0	0
P34	RxD3	输入	×	PFSEG24 = 0	—	—	1	×
	SEG24	输出	×	PFSEG24 = 1	—	—	0	0
P35	TxD3	输出	×	PFSEG25 = 0	0/1	—	0	1
	SEG25	输出	×	PFSEG25 = 1	0	—	0	0
P40	TOOL0	输入 / 输出	×	—	—	—	×	×
	(TI00)	输入	1	—	—	—	1	×
	(TO00)	输出	1	—	—	—	0	0
P41	(TI07)	输入	1	—	—	—	1	×
	(TO07)	输出	1	—	—	—	0	0
P42	TI05	输入	0	—	×	0	1	×
	TO05	输出	0	—	0	0	0	0
	(TxD1)	输出	1	—	0/1	0	0	1
	IVREF1	输入	×	—	×	1	1	×
P43	(INTP7)	输入	1	—	×	0	1	×
	(RxD1)	输入	1	—	×	0	1	×
	(SDA10)	输入 / 输出	1	—	1	0	0	1
	IVCMP1	输入	×	—	×	1	1	×
P44	(SCL10)	输出	1	—	0/1	0	0	1
	IVCMP0	输入	×	—	×	1	1	×
P45	IVREF0	输入	×	PFDEG = 1	—	1	1	×
P46	(SCK00)	输入	1	PFSEG26 = 0	×	—	1	×
		输出	1	PFSEG26 = 0	0/1	—	0	1
	(SCL00)	输出	1	PFSEG26 = 0	0/1	—	0	1
	SEG26	输出	×	PFSEG26 = 1	0	—	0	0

备注 1. ×: 忽略

PIORx: 外围 I/O 重定向寄存器

PFSEGxx: LCD 端口功能寄存器

POMxx: 端口输出模式寄存器

PMCxx: 端口模式控制寄存器

PMxx: 端口模式寄存器

Pxx: 端口输出锁存器

2. 能通过设定外围 I/O 重定向寄存器 (PIOR)，分配 () 内的功能。

3. 有关段输出引脚 (SEGxx) 的复用端口的详细内容，请参照“4.5.4 SEGxx 引脚的复用端口的运行”。

表 4-17 使用复用功能时端口相关寄存器的设定 (5/7)

引脚名称	复用功能		PIORx	PFSEGxx	POMxx	PMCxx	PMxx	Pxx
	功能名称	输入 / 输出						
P47	(SI00)	输入	1	PFSEG27 = 0	×	—	1	×
	(RxD0)	输入	1	PFSEG27 = 0	×	—	1	×
	(SDA00)	输入 / 输出	1	PFSEG27 = 0	1	—	0	1
	SEG27	输出	×	PFSEG27 = 1	0	—	0	0
P50	SEG4	输出	×	PFSEG04 = 1	—	—	0	0
P51	SEG5	输出	×	PFSEG05 = 1	—	—	0	0
P52	TI00	输入	0	PFSEG06 = 0	—	—	1	×
	TO00	输出	0	PFSEG06 = 0	—	—	0	0
	INTP1	输入	×	PFSEG06 = 0	—	—	1	×
	SEG6	输出	×	PFSEG06 = 1	—	—	0	0
P53	INTP2	输入	×	PFSEG07 = 0	×	—	1	×
	SCIO1	输入 / 输出	×	PFSEG07 = 0	0/1	—	1	1
	SEG7	输出	×	PFSEG07 = 1	0	—	0	0
P54	TI02	输入	0	PFSEG08 = 0	×	—	1	×
	TO02	输出	0	PFSEG08 = 0	0	—	0	0
	SCCLK1	输出	×	PFSEG08 = 0	0	—	0	1
	SEG8	输出	×	PFSEG08 = 1	0	—	0	0
P55	INTP5	输入	0	PFSEG09 = 0	×	—	1	×
	SCIO0	输入 / 输出	×	PFSEG09 = 0	0/1	—	1	1
	SEG9	输出	×	PFSEG09 = 1	0	—	0	0
P56	TI06	输入	0	PFSEG10 = 0	×	—	1	×
	TO06	输出	0	PFSEG10 = 0	0	—	0	0
	SCCLK0	输出	×	PFSEG10 = 0	0	—	0	1
	SEG10	输出	×	PFSEG10 = 1	0	—	0	0
P57	INTP6	输入	×	PFSEG11 = 0	—	—	1	×
	SEG11	输出	×	PFSEG11 = 1	—	—	0	0

备注 1. ×: 忽略

PIORx: 外围 I/O 重定向寄存器

PFSEGxx: LCD 端口功能寄存器

POMxx: 端口输出模式寄存器

PMCxx: 端口模式控制寄存器

PMxx: 端口模式寄存器

Pxx: 端口输出锁存器

2. 能通过设定外围 I/O 重定向寄存器 (PIOR)，分配 () 内的功能。

3. 有关段输出引脚 (SEGxx) 的复用端口的详细内容，请参照“4.5.4 SEGxx 引脚的复用端口的运行”。

表 4-17 使用复用功能时端口相关寄存器的设定 (6/7)

引脚名称	复用功能		PIORx	PFSEGxx	POMxx	PMCxx	PMxx	Pxx
	功能名称	输入 / 输出						
P60	SCLA0	输入 / 输出	×	—	—	—	0	0
	(TI01)	输入	1	—	—	—	1	×
	(TO01)	输出	1	—	—	—	0	0
P61	SDAA0	输入 / 输出	×	—	—	—	0	0
	(TI02)	输入	1	—	—	—	1	×
	(TO02)	输出	1	—	—	—	0	0
P70	SEG12	输出	×	PFSEG12 = 1	—	—	0	0
P71	SEG13	输出	×	PFSEG13 = 1	—	—	0	0
P72	SEG14	输出	×	PFSEG14 = 1	—	—	0	0
P73	SEG15	输出	×	PFSEG15 = 1	—	—	0	0
P74	SEG16	输出	×	PFSEG16 = 1	—	—	0	0
P75	SEG17	输出	×	PFSEG17 = 1	—	—	0	0
P76	SEG18	输出	×	PFSEG18 = 1	—	—	0	0
P77	SEG19	输出	×	PFSEG19 = 1	—	—	0	0

备注 1. ×: 忽略

PIORx: 外围 I/O 重定向寄存器

PFSEGxx: LCD 端口功能寄存器

POMxx: 端口输出模式寄存器

PMCxx: 端口模式控制寄存器

PMxx: 端口模式寄存器

Pxx: 端口输出锁存器

2. 能通过设定外围 I/O 重定向寄存器 (PIOR)，分配 () 内的功能。

3. 有关段输出引脚 (SEGxx) 的复用端口的详细内容，请参照“4.5.4 SEGxx 引脚的复用端口的运行”。

表 4-17 使用复用功能时端口相关寄存器的设定 (7/7)

引脚名称	复用功能		PIORx	PFSEGxx	POMxx	PMCxx	PMxx	Pxx	ISCLCD
	功能名称	输入 / 输出							
P125	V _{L3}	输入 / 输出	×	—	—	—	1	0	ISCVL3 = 0
	(TI06)	输入	1	—	—	—	1	×	ISCVL3 = 1
	(TO06)	输出	1	—	—	—	0	0	ISCVL3 = 1
P126	CAPL	输出	×	—	—	—	1	0	ISCCAP = 0
	(TI04)	输入	1	—	—	—	1	×	ISCCAP = 1
	(TO04)	输出	1	—	—	—	0	0	ISCCAP = 1
P127	CAPH	输出	×	—	—	—	1	0	ISCCAP = 0
	(TI03)	输入	1	—	—	—	1	×	ISCCAP = 1
	(TO03)	输出	1	—	—	—	0	0	ISCCAP = 1
P130	(SO00)	输出	1	PFSEG28 = 0	0/1	—	0	1	—
	(TxD0)	输出	1	PFSEG28 = 0	0/1	—	0	1	—
	SEG28	输出	×	PFSEG28 = 1	0	—	0	0	—
P137	INTP0	输入	×	—	—	—	1	×	—

备注 1. ×: 忽略

PIORx: 外围 I/O 重定向寄存器

PFSEGxx: LCD 端口功能寄存器

POMxx: 端口输出模式寄存器

PMCxx: 端口模式控制寄存器

PMxx: 端口模式寄存器

Pxx: 端口输出锁存器

ISCLCD: LCD 输入切换控制寄存器

2. 能通过设定外围 I/O 重定向寄存器 (PIOR)，分配 () 内的功能。

3. 有关 V_{L3}、CAPL 和 CAPH 引脚的复用端口的详细内容，请参照“4.5.5 V_{L3}、CAPL、CAPH 引脚的复用端口的运行”。

注 1. ANI16/P22 和 ANI17/P23 引脚的功能取决于端口模式控制寄存器 2 (PMC2)、模拟输入通道指定寄存器 (ADS) 和 PM2 寄存器的设定。

表 4-18 ANI16/P22 和 ANI17/P23 引脚功能的设定

PMC2 寄存器	PM2 寄存器	ADS 寄存器	ANI16/P22 和 ANI17/P23 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 (转换对象)
		不选择 ANI	模拟输入 (非转换对象)
	输出模式	选择 ANI	禁止设定
		不选择 ANI	

2. ANI0/P21、ANI1/P20 引脚的功能可通过 A/D 端口配置寄存器 (ADPC)、模拟输入通道指定寄存器 (ADS) 和端口模式寄存器 2 (PM2) 进行选择。

表 4-19 ANI0/P21 和 ANI1/P20 引脚功能的设定

ADPC 寄存器	PM2 寄存器	ADS 寄存器	ANI0/P21、ANI1/P20 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 (转换对象)
		不选择 ANI	模拟输入 (非转换对象)
	输出模式	选择 ANI	禁止设定
		不选择 ANI	

备注 ×: 忽略

4.5.4 SEGxx 引脚的复用端口的运行

段输出引脚（SEGxx）的复用端口功能取决于端口模式控制寄存器（PMCxx）、端口模式寄存器（PMxx）和 LCD 端口功能寄存器 0 ~ 6（PFSEG0 ~ PFSEG6）的设置。

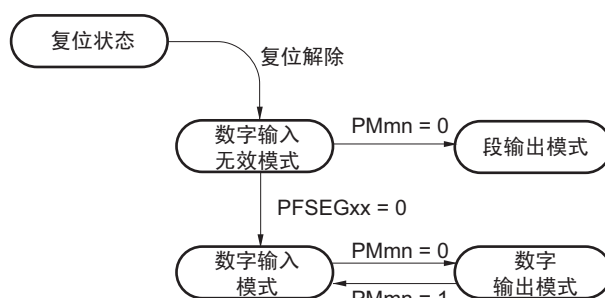
- (1) P00 ~ P07、P14 ~ P17、P30 ~ P35、P46、P47、P50 ~ P57、P70 ~ P77、P130（不复用作模拟输入引脚（ANLxx）的端口）

表 4-20 SEGxx/ 端口引脚功能的设定

PFSEG0 ~ PFSEG6 寄存器的 PFSEGxx 位	PMxx 寄存器的 PMxx 位	引脚功能	初始状态
1	1	数字输入无效模式	○
0	0	数字输出模式	—
0	1	数字输入模式	—
1	0	段输出模式	—

SEGxx/ 端口引脚功能的状态转移如下所示。

图 4-46 SEGxx/ 端口引脚功能状态转移图



注意 只有在设定为开始段输出前（LCD 模式寄存器 1（LCDM1）的 SCOC 为“0”期间）才能设定为段输出模式。

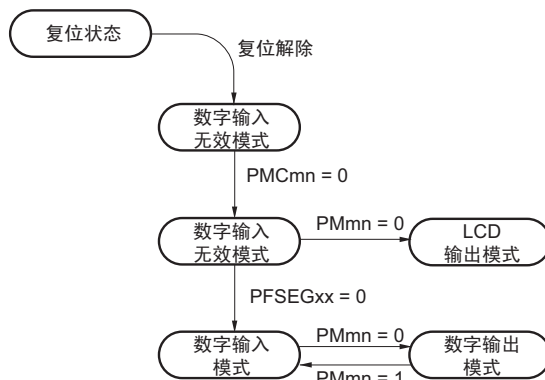
- (2) P10 ~ P13 和 P24 ~ P27（不用做模拟输入引脚（ANLxx）的端口）

表 4-21 SEGxx/ 端口引脚功能的设定

PMCxx 寄存器的 PMCxx 位	PFSEG3 和 PFSEG4 寄存器的 PFSEGxx 位	PMxx 寄存器的 PMxx 位	引脚功能	初始状态
1	1	1	数字输入无效模式	○
0	0	0	数字输出模式	—
0	0	1	数字输入模式	—
0	1	0	段输出模式	—
0	1	1	数字输入无效模式	—
上述以外			禁止设定	

SEGxx/ 端口引脚功能的状态转移如下所示。

图 4-47 SEGxx/ 端口引脚功能的状态转移图



注意 只有在设定为开始段输出前（LCD 模式寄存器 1（LCMD1）的 SCOC 为“0”期间）才能设定为段输出模式。

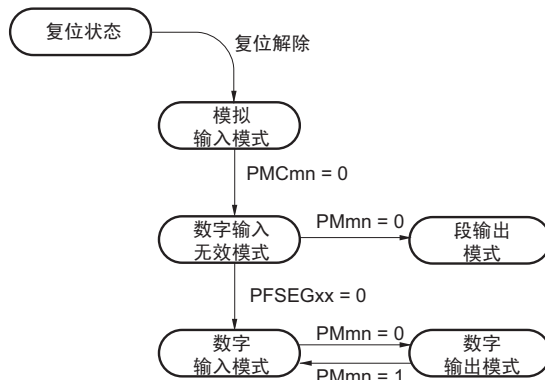
(3) P22 和 P23 (模拟输入引脚 (ANLxx) 的复用端口)

表 4-22 ANIxx/SEGxx/ 端口引脚功能的设定

PMCxx 寄存器的 PMCxx 位	PFSEG3 寄存器的 PFSEGxx 位	PMxx 寄存器的 PMxx 位	引脚功能	初始状态
1	1	1	模拟输入模式	○
0	0	0	数字输出模式	—
0	0	1	数字输入模式	—
0	1	0	段输出模式	—
0	1	1	数字输入无效模式	—
上述以外			禁止设定	

ANLxx/SEGxx/ 端口引脚功能的状态转移如下所示。

图 4-48 ANIxx/SEGxx/ 端口引脚功能状态转换的框图



注意 只有在设定为开始段输出前 (LCD 模式寄存器 1 (LCMD1) 的 SCOC 为“0”期间) 才能设定为段输出模式。

4.5.5 V_{I3}、CAPL、CAPH 引脚的复用端口的运行

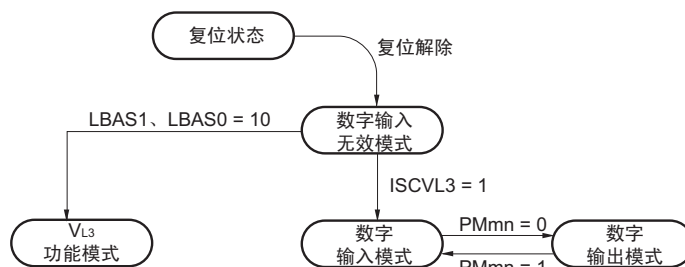
V_{L3}/P125、CAPL/P126、CAPH/P127 引脚的功能取决于 LCD 输入切换控制寄存器 (ISCLCD)、LCD 模式寄存器 0 (LCDM0) 和端口模式寄存器 12 (PM12) 的设定。

(1) $V_{L3}/P125$

表 4-23 V₁₃/P125 引脚功能的设定

偏置设定 (LCDS0 寄存器的 LBAS1 和 LBAS0 位)	ISCLCD 寄存器的 ISCVL3 位	PM12 寄存器的 PM125 位	引脚功能	初始状态
非 1/4 偏置方式 (LBAS1、LBAS0 = 00 或 01)	0	1	数字输入无效模式	○
	1	0	数字输出模式	—
	1	1	数字输入模式	—
1/4 偏置方式 (LBAS1、LBAS0 = 10)	0	1	V _{L3} 功能模式	—
上述以外			禁止设定	

V_{L3}/P125 引脚功能的状态转移如下所示。

图 4-49 $V_{I3}/P125$ 引脚功能的状态转移图

注意 只有在设定为开始段输出前 (LCD 模式寄存器 1 (LCMD1) 的 SCOC 为“0”期间) 才能设定为 V_{13} 功能模式。

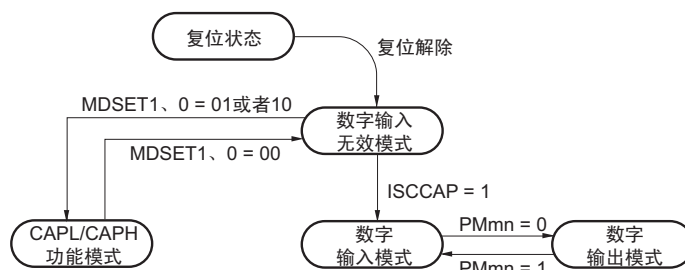
(2) CAPL/P126、CAPH/P127

表 4-24 CAPL/P126、CAPH/P127 引脚功能的设定

LCD 驱动器电压产生器 (LCDM0 寄存器的 MDSET1 和 MDSET0 位)	ISCLCD 寄存器的 ISCCAP 位	PM12 寄存器的 PM126、 PM127 位	引脚功能	初始状态
外部电阻分压 (MDSET1、 MDSET0 = 00)	0	1	数字输入无效模式	○
	1	0	数字输出模式	—
	1	1	数字输入模式	—
内部升压 (MDSET1、 MDSET0 = 01)	0	1	CAPL/CAPH 功能 模式	—
上述以外			禁止设定	

CAPL/P126 和 CAPH/P127 引脚功能的状态转移如下所示。

图 4-50 CAPL/P126 和 CAPH/P127 引脚功能状态转移图



注意 只有在设定为开始段输出前（LCD 模式寄存器 1（LCMD1）的 SCOC 为“0”期间）才能设定为 CAPL/CAPH 功能模式。

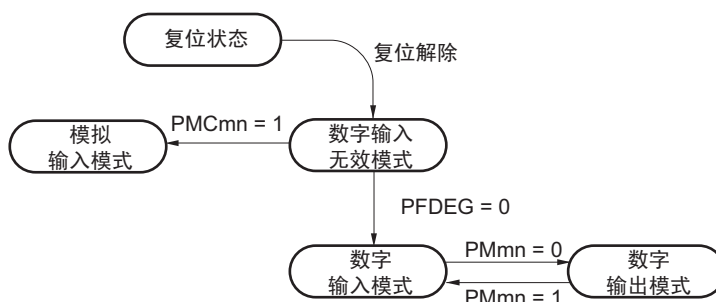
4.5.6 P45（模拟输入引脚（IVREF0）的复用端口）

表 4-25 IVREF0/ 端口引脚功能的设定

PMC4 寄存器的 PMC45 位	PFSEG3 寄存器的 PFDEG 位	PM4 寄存器的 PM45 位	引脚功能	初始状态
1	1	1	模拟输入模式	—
0	0	0	数字输出模式	—
0	0	1	数字输入模式	—
0	1	1	数字输入无效模式	○
上述以外			禁止设定	

IVREF0/ 端口引脚功能的状态转移如下所示。

图 4-51 IVREF0/ 端口引脚功能的状态转移图



4.6 使用端口功能时的注意事项

4.6.1 有关对端口寄存器 n (Pn) 的位操作指令的注意事项

在对同时具有输入和输出功能的端口执行位操作指令时，除了操作对象的位以外，也可能改写非操作对象的输入端口的输出锁存器的值。

因此，建议在将任意的端口从输入模式切换为输出模式前重新写输出锁存器的值。

<例> 当 P10 为输出端口并且 P11 ~ P17 为输入端口（引脚状态全部为高电平）而且端口 1 的输出锁存器值为“00H”时，如果通过位操作指令将输出端口 P10 的输出从低电平改为高电平，端口 1 的输出锁存器的值就变为“FFH”。

说明：PMnm 位为“1”的端口的 Pn 寄存器的写对象是输出锁存器，而读对象是引脚状态。

在 R7F0C003、R7F0C004 内部，位操作指令按照以下顺序执行：

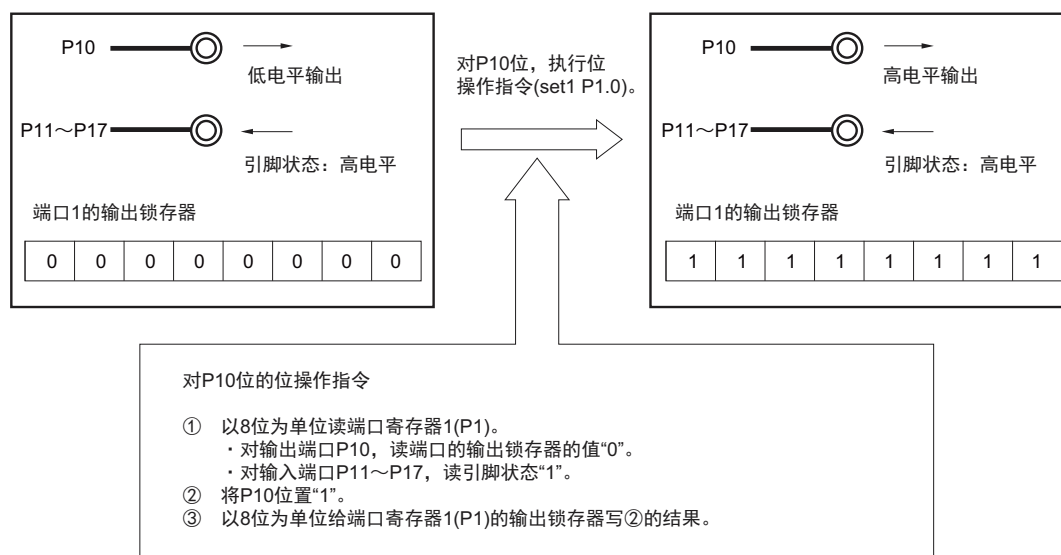
- <1> 以 8 位为单位读 Pn 寄存器。
- <2> 操作 1 位对象位。
- <3> 以 8 位为单位写 Pn 寄存器。

在 <1> 时，读输出端口 P10 的输出锁存器的值“0”，并且读输入端口 P11 ~ P17 的引脚状态。此时，如果 P11 ~ P17 的引脚状态为高电平，读取值就为“FEH”。

通过 <2> 的操作，值变为“FFH”。

通过 <3> 的操作，给输出锁存器写“FFH”。

图 4-52 位存储器操作指令（P10）



4.6.2 引脚设定时的注意事项

如果没有给使用的输出引脚分配其他的复用输出功能，就需要把不使用的复用功能的输出设定为与初始状态相同的值（防止输出冲突）。该注意事项也适用于通过设定外围 I/O 重定向寄存器（PIOR）分配的功能。有关复用输出功能，请参照“4.5 使用复用功能时的寄存器设定”。

因为输入引脚的复用输出功能无效（缓冲器输出为 Hi-Z），所以不需要进行处理。

具体的对象和处理方法如下所示。

表 4-26 不使用的复用功能的处理

对象单元	不使用的复用输出 或者输入 / 输出引脚	不使用的复用功能的处理
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 bit n（TOEmn）设定为与初始状态相同的值“0”。
时钟 / 蜂鸣器输出	PCLBUZn	将时钟输出选择寄存器 n（CKSn）的 bit7（PCLOEn）设定为与初始状态相同的值“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 bit 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）设定为与初始状态相同的值（SOMn 为“1”，其他为“0”）。
IICA	SCAA0、SDAA0	将 IICA 设定为停止运行（IICCTL00 寄存器的 bit7（IICE0）置“0”）。
SMCI	SCCLKn、SCIO n	必须将外围允许寄存器 1（PER1）的 bit1 和 bit2（SMCI0EN 和 SMCI1EN）设定为“0”。
实时时钟 2	RTC1HZ	必须将实时时钟控制寄存器 0（RTCC0）的 bit5（CLOE1）设定为与初始状态相同的值“0”。。
比较器	VCOUTn	必须将比较器输出控制寄存器（COMPOCR）的位（CnOE）设定为与初始状态相同的值“0”。。

例子：P52/TI00/TO00/INTP1/SEG6 引脚的情况

当用作 TO00 输出时

SEG6： 通过将 LCD 端口功能寄存器的 PFSEG06 置“0”设定为“数字输入 / 输出”。

P52： 通过将端口模式寄存器 5 的 PM52 设定置“0”设定为“输出”。

TI00、INTP1： 输入引脚，因此不适用于此注意事项。

如同将 P16/SEG41/SCK00/SCL00 引脚用作“SCK00 输入 / 输出”时的“SCL00”一样，相同串行通道内的复用功能，不能通过模式切换变为有效，因此也不适用于此注意事项。（如果指定为 CSI（MD002 = MD001 = 0），就不作为简易 I²C 运行，因此“SCL00 输出”无效。）

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

第 5 章 时钟发生电路

5.1 时钟发生电路的功能

时钟发生电路是产生给 CPU 和外围硬件提供时钟的电路。

有以下 3 种系统时钟和时钟振荡电路。

(1) 主系统时钟

① X1 振荡电路

能通过给 X1 和 X2 连接谐振器使 $f_X=1\sim 20\text{MHz}$ 的时钟振荡，并且能通过执行 STOP 指令或者设定 MSTOP 位（时钟运行状态控制寄存器（CSC）的 bit7）使振荡停止。

② 高速内部振荡器

能通过选项字节（000C2H）从 $f_{\text{HOCO}}=24\text{MHz}$ 、16MHz、12MHz、8MHz、6MHz、4MHz、3MHz、2MHz 和 1MHz(TYP.) 中选择频率进行振荡。在解除复位后，CPU 一定以此高速内部振荡器时钟开始运行。能通过执行 STOP 指令或者设定 HIOSTOP 位（CSC 寄存器的 bit0）使振荡停止。

能通过高速内部振荡器的频率选择寄存器（HOCODIV）更改选项字节设定的频率。有关频率，请参照“图 5-10 高速内部振荡器频率选择寄存器（HOCODIV）的格式”。

能通过高速内部振荡器设定的振荡频率如下所示（能通过选项字节和高速内部振荡器的频率选择寄存器（HOCODIV）选择的种类）。

电源电压	振荡频率（MHz）								
	1	2	3	4	6	8	12	16	24
$2.7\text{ V} \leq V_{\text{DD}} \leq 5.5\text{ V}$	○	○	○	○	○	○	○	○	○
$2.4\text{ V} \leq V_{\text{DD}} \leq 5.5\text{ V}$	○	○	○	○	○	○	○	○	—
$1.8\text{ V} \leq V_{\text{DD}} \leq 5.5\text{ V}$	○	○	○	○	○	○	—	—	—
$1.6\text{ V} \leq V_{\text{DD}} \leq 5.5\text{ V}$	○	○	○	○	—	—	—	—	—

另外，能由 EXCLK/X2/P122 引脚提供外部主系统时钟（ $f_{\text{EX}}=1\sim 20\text{MHz}$ ），并且能通过执行 STOP 指令或者设定 MSTOP 位将外部主系统时钟的输入置为无效。

能通过设定 MCM0 位（系统时钟控制寄存器（CKC）的 bit4）进行高速系统时钟（X1 时钟或者外部主系统时钟）和高速内部振荡器时钟的切换。

(2) 副系统时钟

• XT1 振荡电路

能通过给 XT1 和 XT2 连接 32.768kHz 的谐振器使 $f_{\text{XT}}=32.768\text{kHz}$ 的时钟振荡，并且能通过设定 XTSTOP 位（时钟运行状态控制寄存器（CSC）的 bit6）使振荡停止。

另外，能由 EXCLKS/XT2/P124 引脚提供外部副系统时钟（ $f_{\text{EXS}}=32.768\text{kHz}$ ），并且能通过设定 XTSTOP 位将外部副系统时钟的输入置为无效。

(3) 低速内部振荡器时钟

能使 $f_{IL}=15\text{kHz}$ (TYP.) 的时钟振荡。

不能将低速内部振荡器时钟用作 CPU 时钟。

只有以下外围硬件能通过低速内部振荡器时钟运行：

- 看门狗定时器
- 实时时钟 2
- 12 位间隔定时器
- LCD 控制器/驱动器

当选项字节 (000C0H) 的 bit4 (WDTON) 和副系统时钟提供模式控制寄存器 (OSMC) 的 bit4 (WUTMMCK0) 中的一个或者两个都为“1”时，低速内部振荡器时钟振荡。

但是，在 WDTON 位为“1”并且 WUTMMCK0 位为“0”而且选项字节 (000C0H) 的 bit0 (WDSTBYON) 为“0”时，如果执行 HALT 指令或者 STOP 指令，低速内部振荡器就停止振荡。

注意 只有在使用固定周期中断功能时，才能选择低速内部振荡器时钟 (f_{IL}) 作为实时时钟 2 的计数时钟。

备注	f_X :	X1 时钟振荡频率
	f_{IH} :	高速内部振荡器的时钟频率
	f_{EX} :	外部主系统时钟频率
	f_{XT} :	XT1 时钟振荡频率
	f_{EXS} :	外部副系统时钟频率
	f_{IL} :	低速内部振荡器的时钟频率

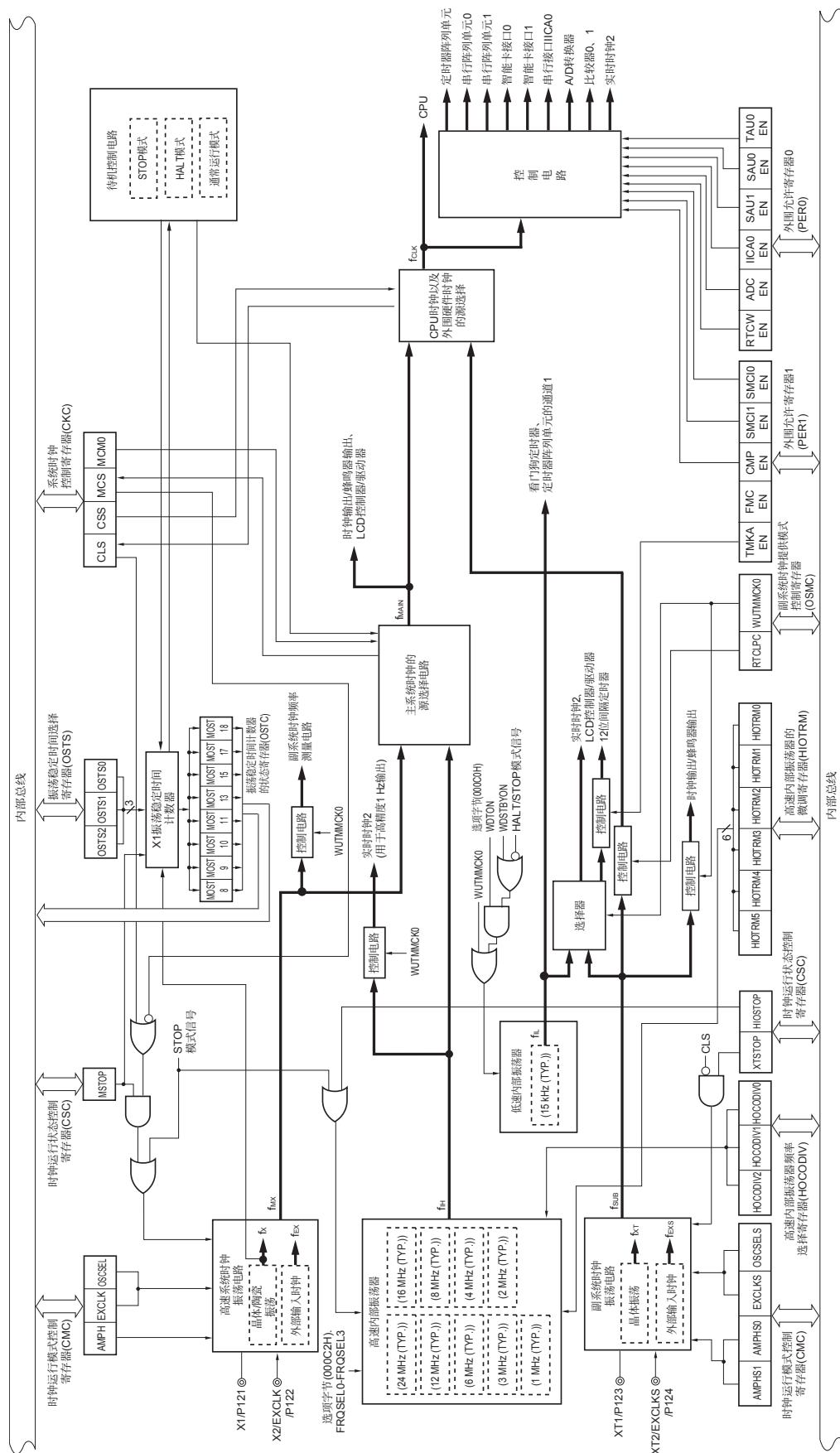
5.2 时钟发生电路的结构

时钟发生电路包括以下硬件。

表 5-1 时钟发生电路的结构

项目	结构
控制寄存器	时钟运行模式控制寄存器 (CMC) 系统时钟控制寄存器 (CKC) 时钟运行状态控制寄存器 (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) 外围允许寄存器 0、1 (PER0、PER1) 副系统时钟提供模式控制寄存器 (OSMC) 高速内部振荡器频率选择寄存器 (HOCODIV) 高速内部振荡器的微调寄存器 (HIOTRM)
振荡电路	X1 振荡电路 XT1 振荡电路 高速内部振荡器 低速内部振荡器

图 5-1 时钟发生电路的框图



备注	f_X :	X1 时钟振荡频率
	f_{IH} :	高速内部振荡器时钟频率
	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_{EXS} :	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} :	低速内部振荡器时钟频率

注 将 WUTMMCK0 位置“1”时，禁止选择 f_{SUB} 作为时钟输出 / 蜂鸣器输出的输出时钟。

5.3 控制时钟发生电路的寄存器

由以下寄存器控制时钟发生电路。

- 时钟运行模式控制寄存器 (CMC)
- 系统时钟控制寄存器 (CKC)
- 时钟运行状态控制寄存器 (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)
- 外围允许寄存器0、1 (PER0、PER1)
- 副系统时钟提供模式控制寄存器 (OSMC)
- 高速内部振荡器频率选择寄存器 (HOCODIV)
- 高速内部振荡器的微调寄存器 (HIOTRM)

注意 配置的寄存器和位因产品而不同。必须给未配置的位设定初始值。

5.3.1 时钟运行模式控制寄存器 (CMC)

这是设定 X1/P121、X2/EXCLK/P122、XT1/P123、XT2/EXCLKS/P124 引脚的运行模式以及选择振荡电路增益的寄存器。

在解除复位后，只能通过 8 位存储器操作指令写 1 次 CMC 寄存器。能通过 8 位存储器操作指令读此寄存器。在产生复位信号后，此寄存器的值变为“00H”。

注意 只在产生上电复位时初始化 EXCLKS、OSCSELS、AMPHS1 和 AMPHS0 位，当产生其他复位源引起的复位时保持原值不变。

图 5-2 时钟运行模式控制寄存器（CMC）的格式

地址：FFFA0H 复位后：00H 注 R/W

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

EXCLK	OSCSEL	高速系统时钟引脚的运行模式	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	副系统时钟引脚的运行模式	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 振荡电路的振荡模式
0	0	低功耗振荡（默认）
0	1	通常振荡
1	0	超低功耗振荡
1	1	禁止设定

AMPH	X1 时钟振荡频率的控制
0	$1\text{MHz} \leq f_X \leq 10\text{MHz}$
1	$10\text{MHz} < f_X \leq 20\text{MHz}$

注 只在产生上电复位时初始化 EXCLKS、OSCSELS、AMPHS1 和 AMPHS0 位，当产生其他复位源引起的复位时保持原值不变。

- 注意 1. 在解除复位后，只能通过 8 位存储器操作指令写 1 次 CMC 寄存器。当以初始值（“00H”）使用 CMC 寄存器时，为了防止程序失控时的误动作（如果误写“00H”以外的值就不能恢复），必须在解除复位后将 CMC 寄存器置“00H”。
2. 在解除复位后并且在通过设定时钟运行状态控制寄存器（CSC）开始 X1 或者 XT1 振荡前，必须设定 CMC 寄存器。
3. 在 X1 时钟振荡频率大于 10MHz 时，必须将 AMPH 位置“1”。
4. 在解除复位后，必须在选择 f_{IH} 作为 f_{CLK} 的状态（将 f_{CLK} 切换为 f_{MX} 前的状态）下设定 AMPH 位、AMPHS1 位和 AMPHS0 位。
5. 必须通过软件对 f_{XT} 的振荡稳定时间进行计数。
6. 系统时钟的频率上限是 24MHz，但是 X1 振荡电路的频率上限为 20MHz。
7. 在写 CMC 寄存器后，如果产生上电复位以外的复位，必须在解除复位后设定与产生复位前相同的值以防止程序失控时的误动作。

注意 8. 为了实现低功耗，XT1 振荡电路是低增幅电路。必须在设计时注意以下几点：

- 引脚和电路板含有寄生电容。因此，必须通过实际使用的电路板进行振荡评估，确认是否有问题。
- 在将 XT1 振荡电路的模式用于超低功耗振荡（AMPHS1、AMPHS0=1、0）时，必须在对“5.7 谐振器和振荡电路常数”记载的谐振器进行充分评估后使用。
- 必须尽量缩短 XT1 引脚、XT2 引脚和谐振器之间的布线，减小寄生电容和布线电阻。尤其在选择超低功耗振荡（AMPHS1、AMPHS0=1、0）时要注意。
- 必须使用寄生电容和布线电阻小的电路板材料构成电路。
- 必须尽量在 XT1 振荡电路的附近配置和 V_{SS} 同电位的接地图形。
- XT1 引脚、XT2 引脚和谐振器的信号线不能和其他信号线交叉，并且不能接近有变化的大电流流过的布线。
- 在高湿度环境中，因电路板的吸湿或者结露可能导致 XT1 引脚和 XT2 引脚之间的阻抗下降而影响振荡。在这样的环境中使用时，必须对电路板进行涂层等防潮措施。
- 在对电路板进行涂层处理时，必须在 XT1 引脚和 XT2 引脚之间使用不产生电容或者漏电流的材料。

备注 f_X : X1 时钟频率

5.3.2 系统时钟控制寄存器（CKC）

这是选择 CPU/ 外围硬件时钟和主系统时钟的寄存器。
通过 1 位或者 8 位存储器操作指令设定 CKC 寄存器。
在产生复位信号后，此寄存器的值变为“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 注 1

符号	7	6	5	4	3	2	1	0
CKC	CLS	CSS	MCS	MCM0	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 注 2	副系统时钟 (f _{SUB})

MCS	主系统时钟 (f _{MAIN}) 的状态
0	高速内部振荡器时钟 (f _{IH})
1	高速系统时钟 (f _{MX})

MCM0 注 2	主系统时钟 (f _{MAIN}) 的运行控制
0	选择高速内部振荡器时钟 (f _{IH}) 作为主系统时钟 (f _{MAIN})
1	选择高速系统时钟 (f _{MX}) 作为主系统时钟 (f _{MAIN})

注 1. bit7 和 bit5 为只读位。
 2. 禁止在将 CSS 位置“1”的状态下更改 MCM0 位的值。

备注 f_{IH}: 高速内部振荡器的时钟频率
 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

注意 1. 必须将 bit0 ~ 3 置“0”。

2. 给 CPU 和外围硬件提供 CSS 位设定的时钟。如果更改 CPU 时钟，就同时更改外围硬件的时钟（实时时钟 2、副系统时钟频率测量电路、12 位间隔定时器、时钟输出 / 蜂鸣器输出、LCD 控制器 / 驱动器和看门狗定时器除外）。因此，如果要更改 CPU/ 外围硬件的时钟，就必须停止各外围功能。

3. 如果将副系统时钟用作外围硬件时钟，就无法保证 A/D 转换器和 IICA 的运行。有关外围硬件的运行特性，请参照各外围硬件的章节和“第 32 章 电特性”。

5.3.3 时钟运行状态控制寄存器（CSC）

这是控制高速系统时钟、高速内部振荡器时钟和副系统时钟（低速内部振荡器时钟除外）运行的寄存器。通过 1 位或者 8 位存储器操作指令设定 CSC 寄存器。
在产生复位信号后，此寄存器的值变为“C0H”。

注意 只在产生上电复位时初始化 XTSTOP 位，当产生其他复位源引起的复位时保持原值不变。

图 5-4 时钟运行状态控制寄存器（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 注	0	0	0	0	0	HIOSTOP

MSTOP	高速系统时钟的运行控制		
	X1 振荡模式	外部时钟输入模式	输入端口模式
0	X1 振荡电路运行	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	副系统时钟的运行控制		
	XT1 振荡模式	外部时钟输入模式	输入端口模式
0	XT1 振荡电路运行	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	高速内部振荡器时钟的运行控制		
0	高速内部振荡器运行		
1	高速内部振荡器停止		

注 只在产生上电复位时初始化 XTSTOP 位，当产生其他复位源引起的复位时保持原值不变。

- 注意 1. 在解除复位后，必须在设定时钟运行模式控制寄存器（CMC）后设定 CSC 寄存器。
2. 在解除复位后并且在将 MSTOP 位置“0”前，必须设定振荡稳定时间选择寄存器（OSTS）。但是，当以初始值使用 OSTS 寄存器时，不需要设定 OSTS 寄存器。
3. 要通过设定 MSTOP 位开始 X1 振荡时，必须通过振荡稳定时间计数器的状态寄存器（OSTC）确认 X1 时钟的振荡稳定时间。
4. 要通过设定 XTSTOP 位开始 XT1 振荡时，必须通过软件等待副系统时钟所需的振荡稳定时间。
5. 不能通过 CSC 寄存器停止选择为 CPU/ 外围硬件时钟（f_{CLK}）的时钟。
6. 有关用于停止时钟振荡（外部时钟输入无效）的寄存器的标志设定和停止前的条件，请参照表 5-2。

表 5-2 停止时钟的方法

时钟	停止时钟（外部时钟输入无效）前的条件	CSC 寄存器的标志设定
X1 时钟	CPU/ 外围硬件时钟以高速系统时钟以外的时钟运行。 （CLS=0 并且 MCS=0，或者 CLS=1）	MSTOP=1
外部主系统时钟		
XT1 时钟	CPU/ 外围硬件时钟以副系统时钟以外的时钟运行。 （CLS=0）	XTSTOP=1
外部副系统时钟		
高速内部振荡器时钟	CPU/ 外围硬件时钟以高速内部振荡器时钟以外的时钟运行。 （CLS=0 并且 MCS=1，或者 CLS=1）	HIOSTOP=1

5.3.4 振荡稳定时间计数器的状态寄存器（OSTC）

这是表示 X1 时钟的振荡稳定时间计数器计数状态的状态寄存器。

能在以下情况下确认 X1 时钟的振荡稳定时间：

- 当 CPU 时钟为高速内部振荡器时钟或者副系统时钟运行并且开始 X1 时钟的振荡时
- 当 CPU 时钟为高速内部振荡器时钟并且在 X1 时钟振荡的状态下转移到 STOP 模式后解除 STOP 模式时

能通过 1 位或者 8 位存储器操作指令读 OSTC 寄存器。

通过复位信号的产生、STOP 指令或者 MSTOP 位（时钟运行状态控制寄存器（CSC）的 bit7）为“1”，此寄存器的值变为“00H”。

备注 在以下情况下，振荡稳定时间计数器开始计数：

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

图 5-5 振荡稳定时间计数器的状态寄存器（OSTC）的格式

地址：FFFA2H 复位后：00H R

符号 7 6 5 4 3 2 1 0

OSTC MOST8 MOST9 MOST10 MOST11 MOST13 MOST15 MOST17 MOST18

MOST 8	MOST 9	MOST 10	MOST 11	MOST 13	MOST 15	MOST 17	MOST 18	振荡稳定时间状态		
									$f_X=10\text{MHz}$	$f_X=20\text{MHz}$
0	0	0	0	0	0	0	0	小于 $2^8/f_X$	小于 25.6μs	小于 12.8μs
1	0	0	0	0	0	0	0	至少 $2^8/f_X$	至少 25.6μs	至少 12.8μs
1	1	0	0	0	0	0	0	至少 $2^9/f_X$	至少 51.2μs	至少 25.6μs
1	1	1	0	0	0	0	0	至少 $2^{10}/f_X$	至少 102μs	至少 51.2μs
1	1	1	1	0	0	0	0	至少 $2^{11}/f_X$	至少 204μs	至少 102μs
1	1	1	1	1	0	0	0	至少 $2^{13}/f_X$	至少 819μs	至少 409μs
1	1	1	1	1	1	0	0	至少 $2^{15}/f_X$	至少 3.27ms	至少 1.63ms
1	1	1	1	1	1	1	0	至少 $2^{17}/f_X$	至少 13.1ms	至少 6.55ms
1	1	1	1	1	1	1	1	至少 $2^{18}/f_X$	至少 26.2ms	至少 13.1ms

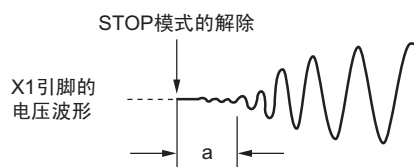
注意 1. 在经过上述时间后，各位从 MOST8 位开始依次变为“1”并且保持“1”的状态。

2. 振荡稳定时间计数器只在振荡稳定时间选择寄存器（OSTS）所设振荡稳定时间内进行计数。

在以下情况下，OSTS 寄存器的振荡稳定时间的设定值必须大于通过 OSTC 寄存器确认的计数值。

- 当 CPU 时钟为高速内部振荡器时钟或者副系统时钟并且要开始 X1 时钟的振荡时
- 当 CPU 时钟为高速内部振荡器时钟并且在 X1 时钟振荡的状态下转移到 STOP 模式后解除 STOP 模式时（因此必须注意，在解除 STOP 模式后只给 OSTC 寄存器设定 OSTS 寄存器所设振荡稳定时间内的状态）

3. X1 时钟的振荡稳定时间不包含时钟开始振荡前的时间（下图 a）。



备注 f_X : X1 时钟振荡频率

5.3.5 振荡稳定时间选择寄存器（OSTS）

这是选择 X1 时钟的振荡稳定等待时间的寄存器。

如果要使 X1 时钟振荡，就在 X1 振荡电路运行（MSTOP=0）后，自动等待 OSTS 寄存器设定的时间。

在将 CPU 时钟从高速内部振荡器时钟或者副系统时钟切换到 X1 时钟时，或者在 CPU 时钟为高速内部振荡器时钟并且在 X1 时钟振荡的状态下转移到 STOP 模式后解除 STOP 模式时，必须通过振荡稳定时间计数器的状态寄存器（OSTC）确认是否经过振荡稳定时间。能通过 OSTC 寄存器确认 OSTS 寄存器事先设定的时间。

通过 8 位存储器操作指令设定 OSTS 寄存器。

在产生复位信号后，此寄存器的值变为“07H”。

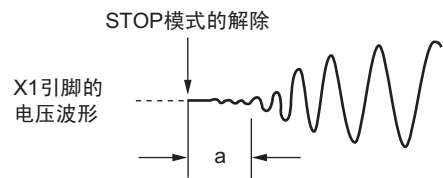
图 5-6 振荡稳定时间选择寄存器（OSTS）的格式

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		
				f _X =10MHz	f _X =20MHz
0	0	0	2 ⁸ /f _X	25.6μs	12.8μs
0	0	1	2 ⁹ /f _X	51.2μs	25.6μs
0	1	0	2 ¹⁰ /f _X	102μs	51.2μs
0	1	1	2 ¹¹ /f _X	204μs	102μs
1	0	0	2 ¹³ /f _X	819μs	409μs
1	0	1	2 ¹⁵ /f _X	3.27ms	1.64ms
1	1	0	2 ¹⁷ /f _X	13.1ms	6.55ms
1	1	1	2 ¹⁸ /f _X	26.2ms	13.1ms

- 注意 1. 要更改 OSTS 寄存器的设定时，必须在将时钟运行状态控制寄存器（CSC）的 MSTOP 位置“0”前进行更改。
2. 振荡稳定时间计数器只在 OSTS 寄存器所设振荡稳定时间内进行计数。
- 在以下情况下，OSTS 寄存器的振荡稳定时间的设定值必须大于在开始振荡后通过 OSTC 寄存器确认的计数值。
- 当 CPU 时钟为高速内部振荡器时钟或者副系统时钟并且要开始 X1 时钟的振荡时
 - 当 CPU 时钟为高速内部振荡器时钟并且在 X1 时钟振荡的状态下转移到 STOP 模式后解除 STOP 模式时（因此必须注意，在解除 STOP 模式后只给 OSTC 寄存器设定 OSTS 寄存器所设振荡稳定时间内的状态）
3. X1 时钟的振荡稳定时间不包含时钟开始振荡前的时间（下图 a）。



备注 f_X: X1 时钟振荡频率

5.3.6 外围允许寄存器 0、 1 （PER0、 PER1）

这是设定允许或者禁止给各外围硬件提供时钟的寄存器。通过停止给不使用的硬件提供时钟，降低功耗和噪声。

当使用由此寄存器控制的以下外围功能时，必须在进行外围功能的初始设定前将对应的位置“1”。

- 实时时钟 2
- A/D 转换器
- 串行接口 IICA0
- 串行阵列单元 1
- 串行阵列单元 0
- 定时器阵列单元
- 12 位间隔定时器
- 副系统时钟频率测量电路
- 比较器 0、 1
- 智能卡接口 1
- 智能卡接口 0

通过 1 位或者 8 位存储器操作指令设定 PER0 寄存器和 PER1 寄存器。
在产生复位信号后，此寄存器的值变为“00H”。

图 5-7 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	RTCWEN	0	ADCEN	IICA0EN	SAU1EN	SAU0EN	0	TAU0EN

RTCWEN	控制实时时钟 2（RTC2）的输入时钟的提供
0	停止提供输入时钟。 • 不能写实时时钟 2（RTC2）使用的 SFR。 • 实时时钟 2（RTC2）能运行。
1	提供输入时钟。 • 能读写实时时钟 2（RTC2）使用的 SFR。 • 实时时钟 2（RTC2）能运行。

ADCEN	控制 A/D 转换器的输入时钟的提供
0	停止提供输入时钟。 • 不能写 A/D 转换器使用的 SFR。 • A/D 转换器处于复位状态。
1	提供输入时钟。 • 能读写 A/D 转换器使用的 SFR。

注意 必须将 bit6 和 bit1 置“0”。

图 5-7 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	RTCWEN	0	ADCEN	IICA0EN	SAU1EN	SAU0EN	0	TAU0EN

IICA0EN	控制串行接口 IICA0 的输入时钟的提供
0	停止提供输入时钟。 - 不能写串行接口 IICA0 使用的 SFR。 - 串行接口 IICA0 处于复位状态。
1	提供输入时钟。 能读写串行接口 IICA0 使用的 SFR。

SAU1EN	控制串行阵列单元 1 的输入时钟的提供
0	停止提供输入时钟。 - 不能写串行阵列单元 1 使用的 SFR。 - 串行阵列单元 1 处于复位状态。
1	提供输入时钟。 能读写定时器阵列单元 1 使用的 SFR。

SAU0EN	控制串行阵列单元 0 的输入时钟的提供
0	停止提供输入时钟。 - 不能写串行阵列单元 0 使用的 SFR。 - 串行阵列单元 0 处于复位状态。
1	提供输入时钟。 能读写定时器阵列单元 0 使用的 SFR。

TAU0EN	控制定时器阵列单元的输入时钟的提供
0	停止提供输入时钟。 - 不能写定时器阵列单元使用的 SFR。 - 定时器阵列单元处于复位状态。
1	提供输入时钟。 能读写定时器阵列单元使用的 SFR。

注意 必须将 bit6 和 bit1 置“0”。

图 5-8 外围允许寄存器 1（PER1）的格式

地址：F007AH 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
PER1	TMKAEN	FMCEN	CMPEN	0	0	SMCI1EN	SMCI0EN	0

TMKAEN	控制 12 位间隔定时器的输入时钟的提供
0	停止提供输入时钟。 - 不能写 12 位间隔定时器使用的 SFR。 12 位间隔定时器处于复位状态。
1	提供输入时钟。 能读写 12 位间隔定时器使用的 SFR。

FMCEN	控制副系统时钟频率测量电路的输入时钟的提供
0	停止提供输入时钟。 - 不能写副系统时钟频率测量电路使用的 SFR。 - 不能写实时时钟 2（RTC2）使用的 SUBCUD 寄存器。 - 副系统时钟频率测量电路处于复位状态。
1	提供输入时钟。 - 能读写副系统时钟频率测量电路使用的 SFR。 - 能读写实时时钟 2（RTC2）使用的 SUBCUD 寄存器。

CMPEN	控制比较器 0、1 的输入时钟的提供
0	停止提供输入时钟。 - 不能写比较器 0、1 使用的 SFR。 - 比较器 0、1 处于复位状态。
1	提供输入时钟。 能读写比较器 0、1 使用的 SFR。

SMCI1EN	控制智能卡接口 1 的输入时钟的提供
0	停止提供输入时钟。 - 不能写智能卡接口 1 使用的 SFR。 - 智能卡接口 1 处于复位状态。
1	提供输入时钟。 能读写智能卡接口 1 使用的 SFR。

SMCI0EN	控制智能卡接口 0 的输入时钟的提供
0	停止提供输入时钟。 - 不能写智能卡接口 0 使用的 SFR。 - 智能卡接口 0 处于复位状态。
1	提供输入时钟。 能读写智能卡接口 0 使用的 SFR。

注意 必须将 bit4、3、0 置“0”。

5.3.7 副系统时钟提供模式控制寄存器（OSMC）

这是通过停止不需要的时钟功能来降低功耗的寄存器。

如果将 RTCLPC 位置“1”，就在 STOP 模式和 CPU 以副系统时钟运行的 HALT 模式中停止给实时时钟 2、12 位间隔定时器、时钟输出 / 蜂鸣器输出和 LCD 控制器 / 驱动器以外的外围功能提供时钟，因此能减小功耗。

另外，能通过 OSMC 寄存器选择实时时钟 2、12 位间隔定时器、时钟输出 / 蜂鸣器输出、LCD 控制器 / 驱动器和副系统时钟频率测量电路的运行时钟。

通过 8 位存储器操作指令设定 OSMC 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 5-9 副系统时钟提供模式控制寄存器（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 模式和 CPU 以副系统时钟运行的 HALT 模式中的设定
0	允许给外围功能提供副系统时钟 (有关允许运行的外围功能，请参照表 21-1 ~ 表 21-3)。
1	停止向实时时钟 2、12 位间隔定时器、时钟输出 / 蜂鸣器输出和 LCD 控制器 / 驱动器以外的外围功能提供副系统时钟。

WUTMMCK0	选择实时时钟 2、12 位间隔定时器和 LCD 控制器 / 驱动器的运行时钟	选择时钟输出 / 蜂鸣器输出的 PCLBUZn 引脚的输出时钟	副系统时钟频率测量电路的运行
0	副系统时钟 (f_{SUB})	允许选择副系统时钟 (f_{SUB})。	有效
1	低速内部振荡器时钟 (f_{IL})	禁止选择副系统时钟 (f_{SUB})。	无效

注意 1. 副系统时钟振荡时，必须选择副系统时钟（WUTMMCK0 位 = 0）。

2. 如果将 WUTMMCK0 位置“1”，低速内部振荡器时钟就振荡。

3. 在将 WUTMMCK0 位设定为“1”时，只可使用实时时钟 2 的固定周期中断功能，不可使用年、月、星期、日、小时、分钟、秒的计数功能以及 1Hz 输出功能。固定周期中断的间隔可通过以下方式算出：

固定周期（由 RTCC0 寄存器所选择的值） $\times f_{SUB} / f_{IL}$ 。

4. 只有在实时时钟 2、12 位间隔定时器和 LCD 控制器 / 驱动器的所有功能都停止时，才能通过 WUTMMCK0 位切换副系统时钟和低速内部振荡器时钟。

5. 在将 WUTMMCK0 位设定为“1”时，禁止选择 f_{SUB} 作为时钟输出 / 蜂鸣器输出的输出时钟。

6. 在将 WUTMMCK0 位设定为“1”时，LCD 控制器 / 驱动器不能使用以下模式。

• 内部升压方式

7. 在将 WUTMMCK0 位设定为“1”时，不能使用副系统时钟频率测量电路。

8. 低功耗 RTC 模式（副系统时钟提供模式控制寄存器（OSMC）的 RTCLPC 位 = 1 时）下，不能从 PCLBUZn 引脚输出副系统时钟（ f_{SUB} ）。

5.3.8 高速内部振荡器频率选择寄存器（HOCODIV）

这是更改选项字节（000C2H）设定的高速内部振荡器频率的寄存器。但是，能选择的频率因选项字节（000C2H）的 FRQSEL3 位的值而不同。

通过 8 位存储器操作指令设定 HOCODIV 寄存器。

在产生复位信号后，此寄存器的值变为通过选项字节（000C2H）的 FRQSEL2 ~ FRQSEL0 设定的值。

图 5-10 高速内部振荡器频率选择寄存器（HOCODIV）的格式

地址：F00A8H 复位后：不定 R/W

符号	7	6	5	4	3	2	1	0
HOCODIV	0	0	0	0	0	HOCODIV2	HOCODIV1	HOCODIV0

HOCODIV2	HOCODIV1	HOCODIV0	高速内部振荡器时钟频率选择	
			FRQSEL3=0	FRQSEL3=1
0	0	0	$f_{IH}=24\text{MHz}$	禁止设定
0	0	1	$f_{IH}=12\text{MHz}$	$f_{IH}=16\text{MHz}$
0	1	0	$f_{IH}=6\text{MHz}$	$f_{IH}=8\text{MHz}$
0	1	1	$f_{IH}=3\text{MHz}$	$f_{IH}=4\text{MHz}$
1	0	0	禁止设定	$f_{IH}=2\text{MHz}$
1	0	1	禁止设定	$f_{IH}=1\text{MHz}$
上述以外			禁止设定	

注意 1. 更改频率前后，都必须在选项字节（000C2H）所设闪存运行模式的可工作电压范围内进行 HOCODIV 寄存器的设定。

选项字节（000C2H）的值		闪存运行模式	工作频率范围	工作电压范围
CMODE1	CMODE2			
0	0	LV（低电压主）模式	1MHz～4MHz	1.6V～5.5V
1	0	LS（低速主）模式	1MHz～8MHz	1.8V～5.5V
1	1	HS（高速主）模式	1MHz～16MHz	2.4V～5.5V
			1MHz～24MHz	2.7V～5.5V
其他		禁止设定		

2. 必须在选择高速内部振荡器时钟（ f_{IH} ）作为 CPU/ 外围硬件时钟（ f_{CLK} ）的状态下设定 HOCODIV 寄存器。

3. 在通过 HOCODIV 寄存器更改频率后，经过以下转移时间之后进行频率切换：

- 以更改前的频率，最多 3 个时钟运行。
- 以更改后的频率，最多等待 3 个 CPU/ 外围硬件时钟。

5.3.9 高速内部振荡器的微调寄存器（HIOTRM）

这是校正高速内部振荡器精度的寄存器。

能使用高精度的外部时钟输入的定时器（定时器阵列单元）等进行高速内部振荡器频率的自测量和精度校正。

通过 8 位存储器操作指令设定 HIOTRM 寄存器。

注意 如果在校正精度后温度和 V_{DD} 引脚的电压发生变化，频率就发生变化。

在温度和 V_{DD} 引脚的电压发生变化的情况下，需要在要求频率的精度前或者定期地进行校正。

图 5-11 高速内部振荡器的微调寄存器（HIOTRM）的格式

地址：F00A0H 复位后：不定注 R/W

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

HIOTRM5	HIOTRM4	HIOTRM3	HIOTRM2	HIOTRM1	HIOTRM0	高速内部振荡器
0	0	0	0	0	0	最低速
0	0	0	0	0	1	↑
0	0	0	0	1	0	
0	0	0	0	1	1	
0	0	0	1	0	0	
⋮						
1	1	1	1	1	0	↓
1	1	1	1	1	1	最高速

注 复位值为出厂时调整后的值。

注意 1. 能对 HIOTRM 寄存器的每个位进行 0.05% 的高速内部振荡器时钟精度校正。

2. HIOTRM 寄存器的使用例子请参照《RL78/G13 High-speed On-chip Oscillator (HOCO) Clock Frequency Correction》Application Note（R01AN0464）”。

5.4 系统时钟振荡电路

5.4.1 X1 振荡电路

X1 振荡电路通过连接 X1 引脚和 X2 引脚的晶体振荡器或者陶瓷谐振器（1 ~ 20MHz）进行振荡。也能输入外部时钟，此时必须给 EXCLK 引脚输入时钟信号。

当使用 X1 振荡电路时，必须对时钟运行模式控制寄存器（CMC）的 bit7 和 bit6（EXCLK、OSCSEL）进行以下的设定：

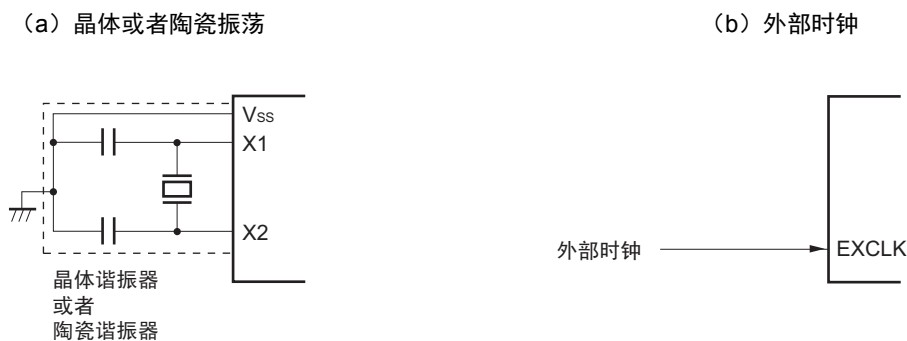
- 晶体或者陶瓷振荡：EXCLK、OSCSEL=0、1
- 外部时钟输入：EXCLK、OSCSEL=1、1

当不使用 X1 振荡电路时，必须设定为输入端口模式（EXCLK、OSCSEL=0、0）。

而且，当也不用作输入端口引脚时，请参照“表 2-2 各未使用引脚的处理”。

X1 振荡电路的外接电路例子如图 5-12 所示。

图 5-12 X1 振荡电路的外接电路例子



注意事项如下页所示。

5.4.2 XT1 振荡电路

XT1 振荡电路通过连接 XT1 引脚和 XT2 引脚的晶体振荡器（32.768kHz(TYP.))进行振荡。

当使用 XT1 振荡电路时，必须将时钟运行模式控制寄存器（CMC）的 bit4（OSCSELS）置“1”。

也能输入外部时钟，此时必须给 EXCLKS 引脚输入时钟信号。

当使用 XT1 振荡电路时，必须对时钟运行模式控制寄存器（CMC）的 bit5 和 bit4（EXCLKS、OSCSELS）进行以下的设定：

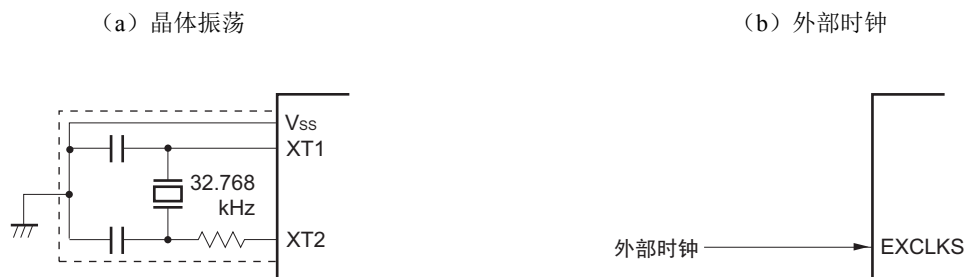
- 晶体或者陶瓷振荡：EXCLKS、OSCSELS=0、1
- 外部时钟输入：EXCLKS、OSCSELS=1、1

当不使用 XT1 振荡电路时，必须设定为输入端口模式（EXCLKS、OSCSELS=0、0）。

而且，当也不用作输入端口引脚时，请参照“表 2-2 各未使用引脚的处理”。

XT1 振荡电路的外接电路例子如图 5-13 所示。

图 5-13 XT1 振荡电路的外接电路例子



注意 当使用 X1 振荡电路和 XT1 振荡电路时，为了避免布线电容等的影响，必须通过以下方法对图 5-12 和图 5-13 中的虚线部分中进行布线：

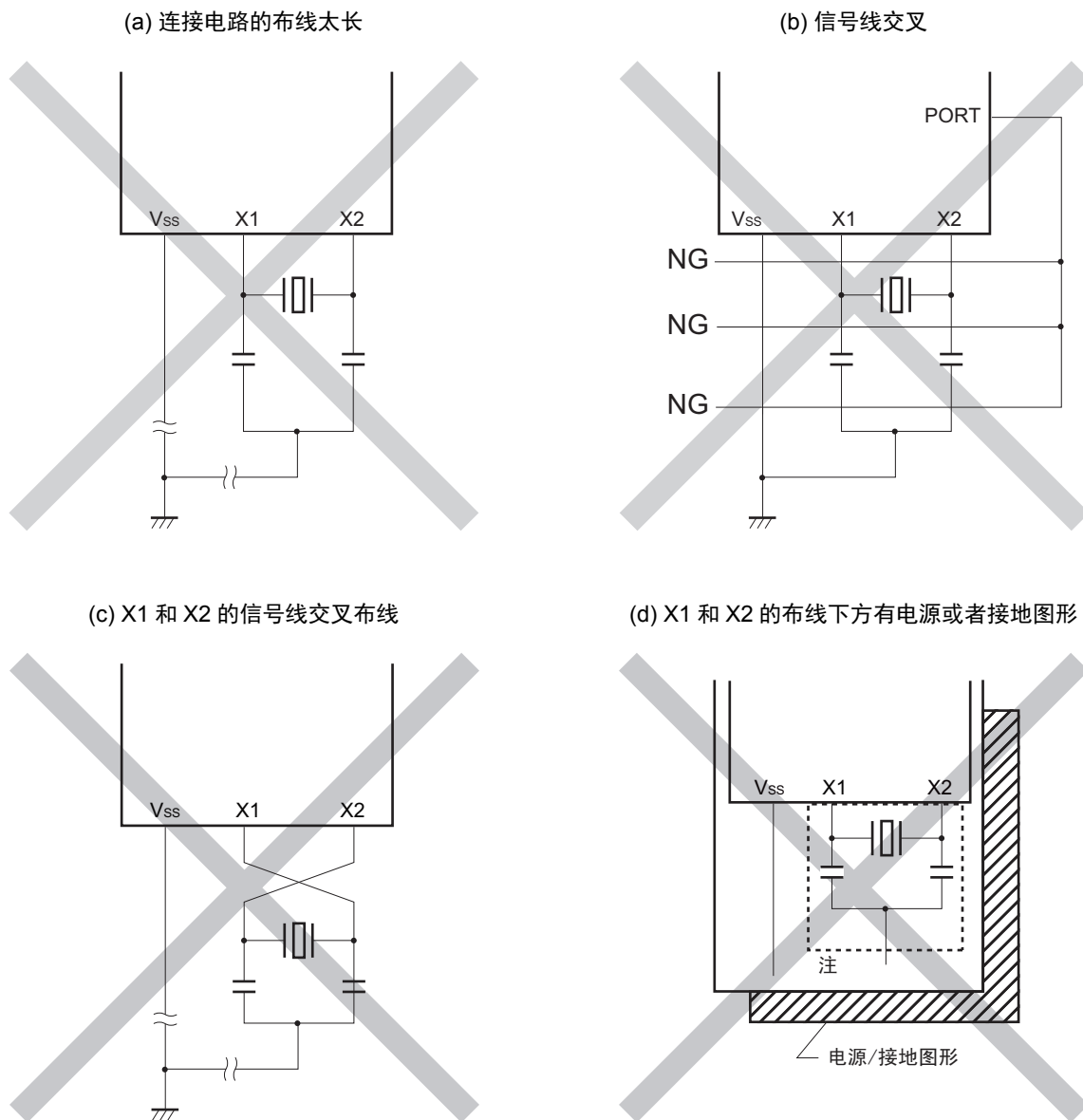
- 必须尽量缩短布线。
- 不能和其他的信号线交叉，并且不能接近有变化的大电流流过的布线。
- 必须始终保持振荡电路的电容器接地点和 V_{SS} 同电位，而且不能给大电流流过的接地图形接地。
- 不能从振荡电路取出信号。

尤其是为了实现低功耗，XT1 振荡电路是低增幅电路。必须在设计时注意以下几点：

- 引脚和电路板含有寄生电容。因此，必须通过实际使用的电路板进行振荡评估，确认是否有问题。
- 在将 XT1 振荡电路的模式用于超低功耗振荡（AMPHS1、AMPHS0=1、0）时，必须在对“5.7 谐振器和振荡电路常数”记载的谐振器进行充分评估后使用。
- 必须尽量缩短 XT1 引脚、XT2 引脚和谐振器之间的布线，减小寄生电容和布线电阻。尤其在选超低功耗振荡（AMPHS1、AMPHS0=1、0）时要注意。
- 必须使用寄生电容和布线电阻小的电路板材料构成电路。
- 必须尽量在 XT1 振荡电路的附近配置和 V_{SS} 同电位的接地图形。
- XT1 引脚、XT2 引脚和谐振器的信号线不能和其他信号线交叉，并且不能接近有变化的大电流流过的布线。
- 在高湿度环境中，因电路板的吸湿或者结露可能导致 XT1 引脚和 XT2 引脚之间的阻抗下降而影响振荡。在这样的环境中使用时，必须对电路板进行涂层等防潮措施。
- 在对电路板进行涂层处理时，必须在 XT1 引脚和 XT2 引脚之间使用不产生电容或者漏电流的材料。

不正确的谐振器连接例子如图 5-14 所示。

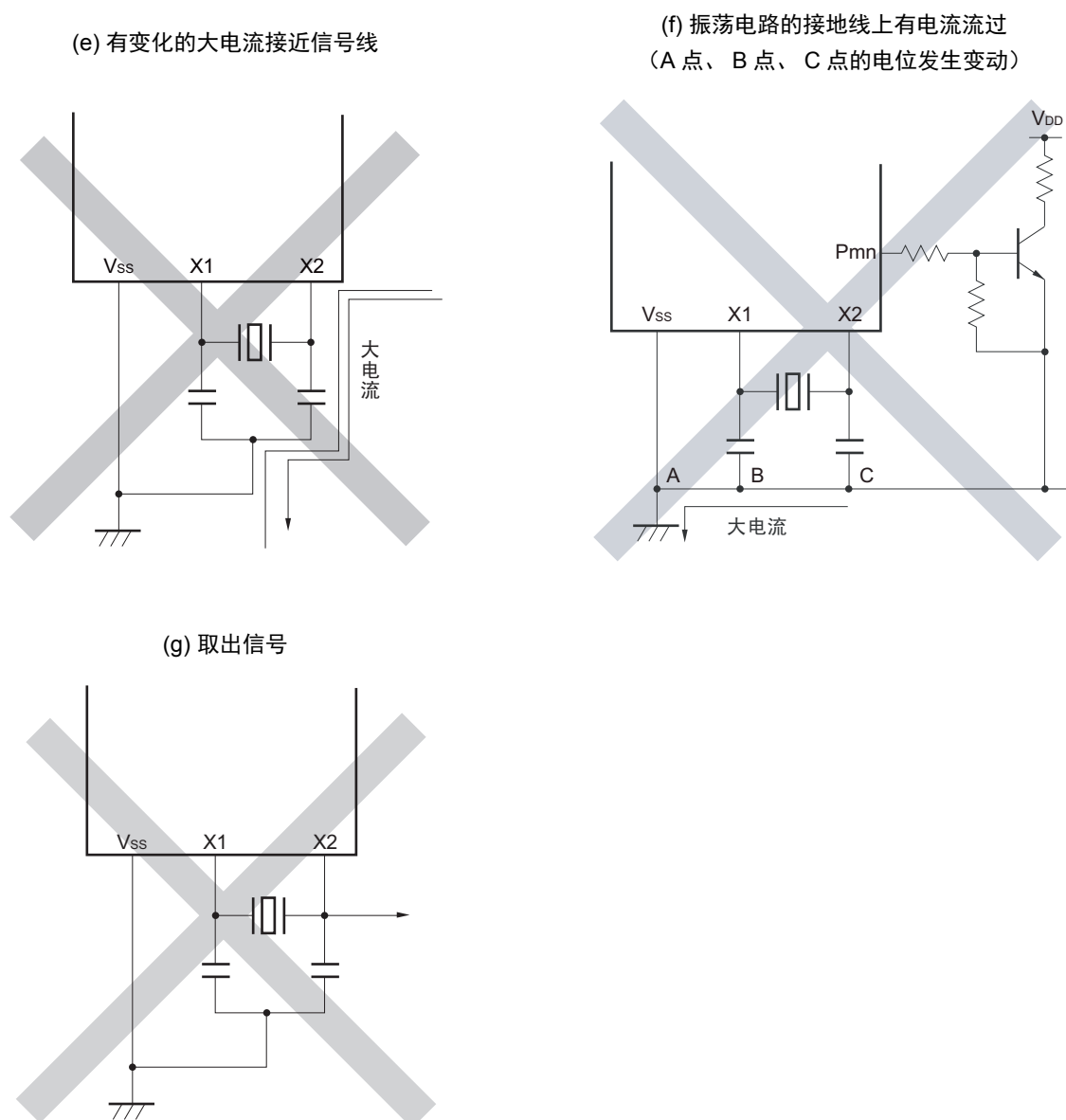
图 5-14 不正确的谐振器连接例子 (1/2)



注 在多层板或者双面板中，不能在 X1 引脚、X2 引脚和谐振器的布线区（图中虚线部分）下方配置电源或者接地图形。布线不能产生电容成分而影响振荡特性。

备注 在使用副系统时钟的情况下，请阅读时分别用 XT1 和 XT2 代替 X1 和 X2，并且在 XT2 侧插入串联电阻。

图 5-14 不正确的谐振器连接例子 (2/2)



注意 当 X2 和 XT1 并行布线时，X2 的串扰噪声会叠加到 XT1 而导致误动作。

备注 在使用副系统时钟的情况下，请阅读时分别用 XT1 和 XT2 代替 X1 和 X2，并且在 XT2 侧插入串联电阻。

5.4.3 高速内部振荡器

R7F0C003 和 R7F0C004 内置高速内部振荡器。能通过选项字节（000C2H）从 24MHz、16MHz、12MHz、8MHz、6MHz、4MHz、3MHz、2MHz 和 1MHz 中选择频率。能通过时钟运行状态控制寄存器（CSC）的 bit0（HIOSSTOP）控制振荡。

在解除复位后，高速内部振荡器自动开始振荡。

5.4.4 低速内部振荡器

R7F0C003 和 R7F0C004 内置低速内部振荡器。

低速内部振荡器时钟用作看门狗定时器、实时时钟 2、12 位间隔定时器和 LCD 控制器 / 驱动器的时钟，但是不能用作 CPU 时钟。

当选项字节（000C0H）的 bit4（WDTON）和副系统时钟提供模式控制寄存器（OSMC）的 bit4（WUTMMCK0）中的 1 个或者两者都为“1”时，低速内部振荡器运行。

当看门狗定时器停止运行，并且在 WUTMMCK0 位不为“0”时，低速内部振荡器继续振荡。但是，当看门狗定时器运行，并且只有在 WUTMMCK0=0 时，WDSTBYON 为“0”并且在 HALT、STOP、SNOOZE 模式中低速内部振荡器停止振荡。在看门狗定时器运行时，即使程序失控，低速内部振荡器时钟也不停止运行。

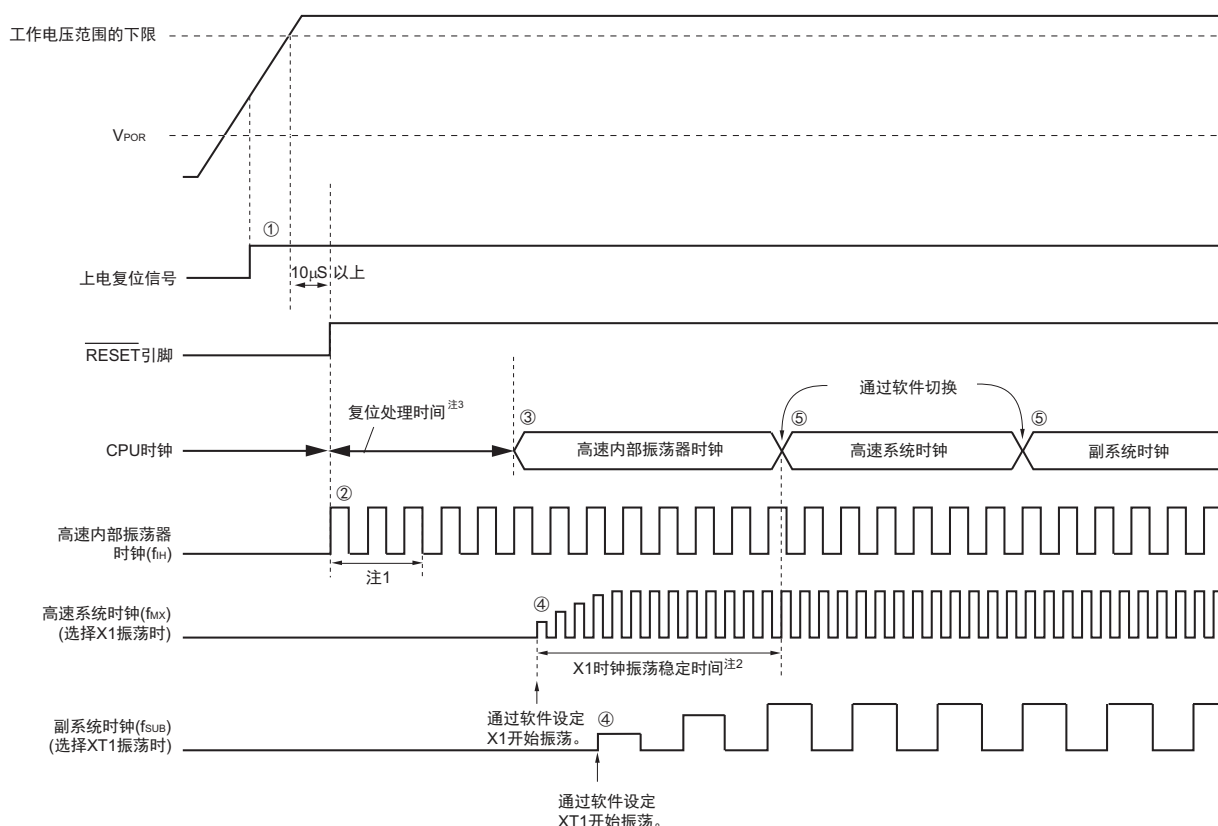
5.5 时钟发生电路的运行

时钟发生电路产生以下所示各种时钟，并且控制待机模式等 CPU 的运行模式（参照图 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}
 - 高速内部振荡器时钟 f_{IH}
- 副系统时钟 f_{SUB}
 - XT1 时钟 f_{XT}
 - 外部副系统时钟 f_{EXS}
- 低速内部振荡器时钟 f_{IL}
- CPU/外围硬件时钟 f_{CLK}

R7F0C003 和 R7F0C004 在解除复位后，CPU 通过高速内部振荡器的输出开始运行。接通电源时的时钟发生电路的运行如图 5-15 所示。

图 5-15 接通电源时时钟发生电路的操作



- ① 在接通电源后，通过上电复位（POR）电路产生内部复位信号。
但是，在达到“32.4 AC 特性”中所示的工作电压范围前，通过电压检测电路或者外部复位保持复位状态（上图为使用外部复位时的例子）。
- ② 如果解除复位，高速内部振荡器就自动开始振荡。
- ③ 在解除复位后，进行电压稳定等待和复位处理，然后CPU以高速内部振荡器时钟开始运行。
- ④ 必须通过软件设定X1时钟或者XT1时钟的开始振荡（参照“5.6.2 X1振荡电路的设定例子”和“5.6.3 XT1振荡电路的设定例子”）。
- ⑤ 如果要将CPU时钟切换到X1时钟或者XT1时钟，就必须在等待时钟振荡稳定后通过软件设定切换（参照“5.6.2 X1振荡电路的设定例子”和“5.6.3 XT1振荡电路的设定例子”）。

- 注 1. 复位处理时间包含高速内部振荡器时钟的振荡精度稳定等待时间。
2. 当解除复位时，必须通过振荡稳定时间计数器的状态寄存器（OSTC）确认X1时钟的振荡稳定时间。
3. 有关复位处理时间，请参照“第20章 上电复位电路”。

注意 如果使用EXCLK引脚输入的外部时钟，就不需要振荡稳定等待时间。

5.6 时钟控制

5.6.1 高速内部振荡器的设定例子

在解除复位后，CPU/ 外围硬件时钟 (f_{CLK}) 一定以高速内部振荡器时钟运行。能通过选项字节 (000C2H) 的 FRQSEL0 ~ FRQSEL3 位，从 24MHz、16MHz、12MHz、8MHz、6MHz、4MHz、3MHz、2MHz 和 1MHz 中选择高速内部振荡器的频率。另外，能通过高速内部振荡器的频率选择寄存器 (HOCODIV) 更改频率。

【选项字节的设定】

地址：000C2H

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

CMODE1	CMODE0	闪存运行模式的设定	
0	0	LV (低电压主) 模式	$V_{DD}=1.6V \sim 5.5V@1MHz \sim 4MHz$
1	0	LS (低速主) 模式	$V_{DD}=1.8V \sim 5.5V@1MHz \sim 8MHz$
1	1	HS (高速主) 模式	$V_{DD}=2.4V \sim 5.5V@1MHz \sim 16MHz$ $V_{DD}=2.7V \sim 5.5V@1MHz \sim 24MHz$
上述以外		禁止设定	

FRQSEL3	FRQSEL2	FRQSEL1	FRQSEL0	高速内部振荡器的频率 (f_{IH})
0	0	0	0	24MHz
1	0	0	1	16MHz
0	0	0	1	12MHz
1	0	1	0	8MHz
0	0	1	0	6MHz
1	0	1	1	4MHz
0	0	1	1	3MHz
1	1	0	0	2MHz
1	1	0	1	1MHz
上述以外				禁止设定

【高速内部振荡器的频率选择寄存器 (HOCODIV) 的设定】

地址：F00A8H

HOCODIV	7	6	5	4	3	2	1	0
	0	0	0	0	0	HOCODIV2	HOCODIV1	HOCODIV0

HOCODIV2	HOCODIV1	HOCODIV0	高速内部振荡器时钟频率选择	
			FRQSEL3=0	FRQSEL3=1
0	0	0	$f_{IH}=24MHz$	禁止设定
0	0	1	$f_{IH}=12MHz$	$f_{IH}=16MHz$
0	1	0	$f_{IH}=6MHz$	$f_{IH}=8MHz$
0	1	1	$f_{IH}=3MHz$	$f_{IH}=4MHz$
1	0	0	禁止设定	$f_{IH}=2MHz$
1	0	1	禁止设定	$f_{IH}=1MHz$
上述以外			禁止设定	

5.6.2 X1 振荡电路的设定例子

在解除复位后，CPU/ 外围硬件时钟（ f_{CLK} ）一定以高速内部振荡器时钟运行。此后，如果改为 X1 振荡时钟，就通过振荡稳定时间选择寄存器（OSTS）、时钟运行模式控制寄存器（CMC）和时钟运行状态控制寄存器（CSC）进行振荡电路的设定和控制振荡的开始，并且通过振荡稳定时间计数器的状态寄存器（OSTC）等待振荡稳定。在等待振荡稳定后通过系统时钟控制寄存器（CKC）将 X1 振荡时钟设定为 f_{CLK} 。

【寄存器的设定】 必须按照①～⑤的顺序设定寄存器。

- ① 将CMC寄存器的OSCSSEL位置“1”，当 f_X 大于10MHz时，将AMPHS1位置“1”，使X1振荡电路运行。

	7	6	5	4	3	2	1	0
CMC	EXCLK	OSCSSEL	EXCLKS	OSCSSELS		AMPHS1	AMPHS0	AMPH
	0	1	0	0	0	0	0	0/1

- ② 通过OSTS寄存器选择解除STOP模式时的X1振荡电路的振荡稳定时间。

例）要通过10MHz谐振器至少等待102.4 μ s时，必须设定为以下的值。

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

- ③ 将CSC寄存器的MSTOP位清“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。

例）要通过10MHz谐振器至少等待102.4 μ s时，必须等到各位变为以下的值。

	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 通过CKC寄存器的MCM0位将X1振荡时钟设定为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

注意 只在上电复位时初始化 EXCLKS、OSCSSELS、AMPHS1、AMPHS0 和 XTSTOP 位，在因其他复位源产生的复位时保持原值不变。

5.6.3 XT1 振荡电路的设定例子

在解除复位后，CPU/ 外围硬件时钟 (f_{CLK}) 一定以高速内部振荡器时钟运行。此后，如果改为 XT1 振荡时钟，就通过副系统时钟提供模式控制寄存器 (OSMC)、时钟运行模式控制寄存器 (CMC) 和时钟运行状态控制寄存器 (CSC) 进行振荡电路的设定和控制振荡的开始，并且通过系统时钟控制寄存器 (CKC) 将 XT1 振荡时钟设定为 f_{CLK} 。

【寄存器的设定】必须按照①～⑤的顺序设定寄存器。

- ① 在 STOP 模式和 CPU 以副系统时钟运行的 HALT 模式中，只让实时时钟 2、12 位间隔定时器以及 LCD 控制器/驱动器以副系统时钟运行（超低消耗电流）时，必须将 RTCLPC 位置“1”。

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

- ② 将 CMC 寄存器的 OSCSELS 位置“1”，使 XT1 振荡电路运行。

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

AMPHS0 位和 AMPHS1 位：设定 XT1 振荡电路的振荡模式。

- ③ 将 CSC 寄存器的 XTSTOP 位清“0”，使 XT1 振荡电路开始振荡。

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

- ④ 必须通过软件和定时器功能等，等待副系统时钟所需的振荡稳定时间。

- ⑤ 通过 CKC 寄存器的 CSS 位将 XT1 振荡时钟设定为 CPU/ 外围硬件时钟。

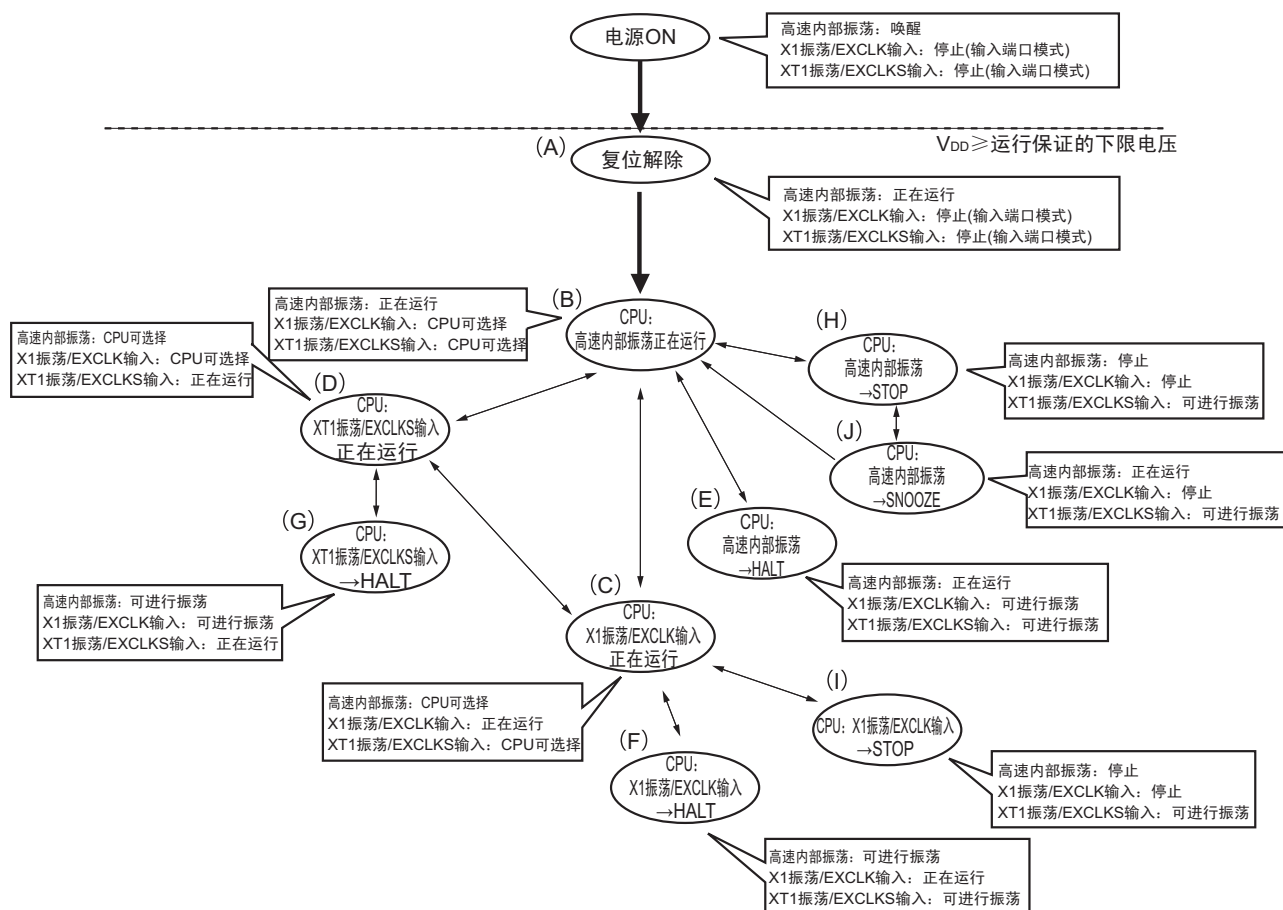
	7	6	5	4	3	2	1	0
CKC	CLS 0	CSS 1	MCS 0	MCM0 0	0	0	0	0

注意 只在上电复位时初始化 EXCLKS、OSCSELS、AMPHS1、AMPHS0 和 XTSTOP 位，在因其他复位源产生的复位时保持原值不变。

5.6.4 CPU 时钟状态转移图

本产品的 CPU 时钟状态转移图如图 5-16 所示。

图 5-16 CPU 时钟状态转移图



CPU 时钟的转移和 SFR 寄存器的设定例子等如表 5-3 所示。

表 5-3 CPU 时钟的转移和 SFR 寄存器的设定例子 (1/5)

(1) 在解除复位(A)后, CPU 转移到高速内部振荡器时钟运行(B)。

状态转移	SFR 寄存器的设定
(A)→(B)	不需要设定 SFR 寄存器 (解除复位后的初始状态)。

(2) 在解除复位(A)后, CPU 转移到高速系统时钟运行(C)。

(CPU 在解除复位后立即以高速内部振荡器时钟运行(B))

(SFR 寄存器的设定顺序) →

SFR 寄存器的设定标志 状态转移	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		
(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	不需要 确认	1

注 1. 在解除复位后, 只能通过 8 位存储器操作指令写 1 次时钟运行模式控制寄存器 (CMC)。

2. 必须对振荡稳定时间选择寄存器 (OSTS) 的振荡稳定时间进行以下的设定:

- 期待的振荡稳定时间计数器的状态寄存器 (OSTC) 的振荡稳定时间 ≤ OSTS 寄存器设定的振荡稳定时间

注意 必须在电源电压达到设定的时钟可运行电压 (参照“第 32 章”)后设定时钟。

(3) 在解除复位(A)后, CPU 转移到副系统时钟运行(D)。

(CPU 在解除复位后立即以高速内部振荡器时钟运行(B))

(SFR 寄存器的设定顺序) →

SFR 寄存器的设定标志 状态转移	CMC 寄存器注				CSC 寄存器	振荡稳定 的等待	CKC 寄存器
	EXCLKS	OSCSELS	AMPHS1	AMPHS0	XTSTOP		CSS
(A)→(B)→(D) (XT1 时钟)	0	1	0/1	0/1	0	需要	1
(A)→(B)→(D) (外部副时钟)	1	1	×	×	0	需要	1

注 在解除复位后, 只能通过 8 位存储器操作指令写 1 次时钟运行模式控制寄存器 (CMC)。

备注 1. ×: 忽略

2. 表 5-3 的 (A) ~ (J) 对应图 5-15 的 (A) ~ (J)。

表 5-3 CPU 时钟的转移和 SFR 寄存器的设定例子 (2/5)

(4) CPU 从高速内部振荡器时钟运行(B)转移到高速系统时钟运行(C)。

(SFR 寄存器的设定顺序) →

SFR 寄存器的设定标志 状态转移	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: 1MHz ≤ f _X ≤ 10MHz)	0	1	0	注 2	0	需要确认	1
(B)→(C) (X1 时钟: 10MHz < f _X ≤ 20MHz)	0	1	1	注 2	0	需要确认	1
(B)→(C) (外部主时钟)	1	1	×	注 2	0	不需要 确认	1

如果已设定就不需要。

在高速系统时钟运行中不需要。

注 1. 在解除复位后, 只能通过 8 位存储器操作指令写 1 次时钟运行模式控制寄存器 (CMC)。如果已设定就不需要。

2. 必须对振荡稳定时间选择寄存器 (OSTS) 的振荡稳定时间进行以下的设定:

- 期待的振荡稳定时间计数器的状态寄存器 (OSTC) 的振荡稳定时间 ≤ OSTS 寄存器设定的振荡稳定时间

注意 必须在电源电压达到设定的时钟可运行电压 (参照“第 32 章 电特性”) 后设定时钟。

(5) CPU 从高速内部振荡器时钟运行(B)转移到副系统时钟运行(D)。

(SFR 寄存器的设定顺序) →

SFR 寄存器的设定标志 状态转移	CMC 寄存器注			CSC 寄存器	振荡稳定的 等待	CKC 寄存器
	EXCLKS	OSCSELS	AMPHS1、 AMPHS0	XTSTOP		CSS
(B)→(D) (XT1 时钟)	0	1	00: 低功耗振荡 01: 通常振荡 10: 超低功耗振荡	0	需要	1
(B)→(D) (外部副时钟)	1	1		0	需要	1

如果已设定就不需要。

在副速系统时钟运行中不需要。

注 在解除复位后, 只能通过 8 位存储器操作指令写 1 次时钟运行模式控制寄存器 (CMC)。如果已设定就不需要。

备注 1. ×: 忽略

2. 表 5-3 的 (A) ~ (J) 对应图 5-16 的 (A) ~ (J)。

表 5-3 CPU 时钟的转移和 SFR 寄存器的设定例子 (3/5)

(6) CPU 从高速系统时钟运行 (C) 转移到高速内部振荡器时钟运行 (B)。

(SFR 寄存器的设定顺序) →

SFR 寄存器的设定标志 状态转移	CSC 寄存器	振荡精度稳定的等待	CKC 寄存器
	HIOSTOP		MCM0
(C)→(B)	0	18μs ~ 65μs	0

在高速内部振荡器时钟运行中不需要。

(7) CPU 从高速系统时钟运行 (C) 转移到副系统时钟运行 (D)。

(SFR 寄存器的设定顺序) →

SFR 寄存器的设定标志 状态转移	CSC 寄存器	振荡稳定的等待	CKC 寄存器
	XTSTOP		CSS
(C)→(D)	0	需要	1

在副系统时钟运行中不需要。

(8) CPU 从副系统时钟运行 (D) 转移到高速内部振荡器时钟运行 (B)。

(SFR 寄存器的设定顺序) →

SFR 寄存器的设定标志 状态转移	CSC 寄存器	振荡精度稳定的等待	CKC 寄存器	
	HIOSTOP		CSS	MCM0
(D)→(B)	0	18μs ~ 65μs	0	0

在高速内部振荡器时钟运行中不需要。

如果已设定就不需要。

备注 1. 表 5-3 的 (A) ~ (J) 对应图 5-16 的 (A) ~ (J)。

2. 高速内部振荡器时钟的振荡精度稳定的等待因温度条件和 STOP 模式期间而变。

表 5-3 CPU 时钟的转移和 SFR 寄存器的设定例子（4/5）

(9) CPU 从副系统时钟运行(D)转移到高速系统时钟运行(C)。

(SFR 寄存器的设定顺序) →

状态转移 \ SFR 寄存器的设定标志	OSTS 寄存器	CSC 寄存器	OSTC 寄存器	CKC 寄存器
		MSTOP		CSS
(D)→(C) (X1 时钟: $1\text{MHz} \leq f_X \leq 10\text{MHz}$)	注	0	需要确认	0
(D)→(C) (X1 时钟: $10\text{MHz} < f_X \leq 20\text{MHz}$)	注	0	需要确认	0
(D)→(C) (外部主时钟)	注	0	不需要确认	0

在高速系统时钟运行中不需要。

注 必须对振荡稳定时间选择寄存器（OSTS）的振荡稳定时间进行以下的设定：

- 期待的振荡稳定时间计数器的状态寄存器（OSTC）的振荡稳定时间 ≤ OSTS 寄存器设定的振荡稳定时间

注意 必须在电源电压达到设定的时钟可运行电压（参照“第 32 章 电特性”）后设定时钟。

- (10) • CPU 在高速内部振荡器时钟运行中(B)转移到HALT模式(E)。
- CPU 在高速系统时钟运行中(C)转移到HALT模式(F)。
 - CPU 在副系统时钟运行中(D)转移到HALT模式(G)。

状态转移	设定内容
(B)→(E) (C)→(F) (D)→(G)	执行 HALT 指令。

备注 表 5-3 的 (A) ~ (J) 对应图 5-16 的 (A) ~ (J)。

表 5-3 CPU 时钟的转移和 SFR 寄存器的设定例子 (5/5)

- (11) • CPU在高速内部振荡器时钟运行中(B)转移到STOP模式(H)
- CPU在高速系统时钟运行中(C)转移到STOP模式(I)。

(设定顺序) →

状态转移		设定内容		
(B)→(H)		停止禁止在 STOP 模式中运行的外围功能。	—	执行 STOP 指令。
(C)→(I)	X1 振荡		设定 OSTS 寄存器。	
	外部时钟		—	

- (12) • CPU从STOP模式(H)转移到SNOOZE模式(J)。

有关从 STOP 模式转移到 SNOOZE 模式的设定，请参照“12.8 SNOOZE 模式功能”、“14.5.7 SNOOZE 模式功能”和“14.6.3 SNOOZE 模式功能”。

备注 表 5-3 的 (A) ~ (J) 对应图 5-16 的 (A) ~ (J)。

5.6.5 CPU 时钟转移前的条件和转移后的处理

CPU 时钟转移前的条件和转移后的处理如下所示。

表 5-4 有关 CPU 时钟的转移 (1/2)

CPU 时钟		转移前的条件	转移后的处理
转移前	转移后		
高速内部振荡器时钟	X1 时钟	X1 振荡稳定。 • OSCSEL=1, EXCLK=0, MSTOP=0 • 经过振荡稳定时间后	如果停止高速内部振荡器的振荡 (HIOSTOP=1), 就能减小工作电流。
	外部主系统时钟	将 EXCLK 引脚输入的外部时钟置为有效。 • OSCSEL=1, EXCLK=1, MSTOP=0	
	XT1 时钟	XT1 振荡稳定。 • OSCSELS=1, EXCLKS=0, XTSTOP=0 • 经过振荡稳定时间后	
	外部副系统时钟	将 EXCLKS 引脚输入的外部时钟置为有效。 • OSCSELS=1, EXCLKS=1, XTSTOP=0	
X1 时钟	高速内部振荡器时钟	允许高速内部振荡器的振荡。 • HIOSTOP=0 • 经过振荡精度稳定时间后	能停止 X1 的振荡 (MSTOP=1)。
	外部主系统时钟	不能转移。	—
	XT1 时钟	XT1 振荡稳定。 • OSCSELS=1, EXCLKS=0, XTSTOP=0 • 经过振荡稳定时间后	能停止 X1 的振荡 (MSTOP=1)。
	外部副系统时钟	将 EXCLKS 引脚输入的外部时钟置为有效。 • OSCSELS=1, EXCLKS=1, XTSTOP=0	能停止 X1 的振荡 (MSTOP=1)。
外部主系统时钟	高速内部振荡器时钟	允许高速内部振荡器的振荡。 • HIOSTOP=0 • 经过振荡精度稳定时间后	能将外部主系统时钟的输入置为无效 (MSTOP=1)。
	X1 时钟	不能转移。	—
	XT1 时钟	XT1 振荡稳定。 • OSCSELS=1, EXCLKS=0, XTSTOP=0 • 经过振荡稳定时间后	能将外部主系统时钟的输入置为无效 (MSTOP=1)。
	外部副系统时钟	将 EXCLKS 引脚输入的外部时钟置为有效。 • OSCSELS=1, EXCLKS=1, XTSTOP=0	能将外部主系统时钟的输入置为无效 (MSTOP=1)。

表 5-4 有关 CPU 时钟的转移 (2/2)

CPU 时钟		转移前的条件	转移后的处理
转移前	转移后		
XT1 时钟	高速内部振荡器时钟	高速内部振荡器正在振荡并且选择高速内部振荡器时钟作为主系统时钟。 • HIOSTOP=0, MCS=0	能停止 XT1 的振荡 (XTSTOP=1)。
	X1 时钟	X1 振荡稳定并且选择高速系统时钟作为主系统时钟。 • OSCSEL=1, EXCLK=0, MSTOP=0 • 经过振荡稳定时间后 • MCS=1	
	外部主系统时钟	将 EXCLK 引脚输入的外部时钟置为有效并且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	不能转移。	—
外部副系统时钟	高速内部振荡器时钟	高速内部振荡器振荡正在振荡并且选择高速内部振荡器时钟作为主系统时钟 • HIOSTOP=0, MCS=0	能将外部副系统时钟的输入置为无效 (XTSTOP=1)。
	X1 时钟	X1 振荡稳定并且选择高速系统时钟作为主系统时钟。 • OSCSEL=1, EXCLK=0, MSTOP=0 • 经过振荡稳定时间后 • MCS=1	
	外部主系统时钟	将 EXCLK 引脚输入的外部时钟置为有效并且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	不能转移。	—

5.6.6 切换 CPU 时钟和系统时钟所需要的时间

能通过设定系统时钟控制寄存器（CKC）的 bit4 和 bit6（MCM0、CSS）进行 CPU 时钟的切换（主系统时钟 ↔ 副系统时钟）和主系统时钟的切换（高速内部振荡器时钟 ↔ 高速系统时钟）。

在改写 CKC 寄存器后不立即进行实际的切换，而是在更改 CKC 寄存器后仍然以切换前的时钟继续运行数个时钟（参照表 5-5 ~ 表 5-7）。

能通过 CKC 寄存器的 bit7（CLS）来判断 CPU 是以主系统时钟还是以副系统时钟运行。能通过 CKC 寄存器的 bit5（MCS）来判断主系统时钟是以高速系统时钟还是以高速内部振荡器时钟运行。

如果切换 CPU 时钟，就同时切换外围硬件时钟。

表 5-5 切换系统时钟所需要的最长时间

时钟 A	切换方向	时钟 B	备注
f_{IH}	↔	f_{MX}	参照表 5-6
f_{MAIN}	↔	f_{SUB}	参照表 5-7

表 5-6 $f_{IH} \leftrightarrow f_{MX}$ 所需要的最大时钟数

切换前的设定值		切换后的设定值	
MCM0		MCM0	
		0 ($f_{MAIN}=f_{IH}$)	1 ($f_{MAIN}=f_{MX}$)
0 ($f_{MAIN}=f_{IH}$)	$f_{MX} \geq f_{IH}$		2 个时钟
	$f_{MX} < f_{IH}$		2 个 f_{IH}/f_{MX} 时钟
1 ($f_{MAIN}=f_{MX}$)	$f_{MX} \geq f_{IH}$	2 个 f_{MX}/f_{IH} 时钟	
	$f_{MX} < f_{IH}$	2 个时钟	

表 5-7 $f_{MAIN} \leftrightarrow f_{SUB}$ 所需要的最大时钟数

切换前的设定值		切换后的设定值	
CSS		CSS	
		0 ($f_{CLK}=f_{MAIN}$)	1 ($f_{CLK}=f_{SUB}$)
0 ($f_{CLK}=f_{MAIN}$)			1+2 个 f_{MAIN}/f_{SUB} 时钟
1 ($f_{CLK}=f_{SUB}$)		3 个时钟	

备注 1. 表 5-6 和表 5-7 中的时钟数是切换前的 CPU 时钟数。
2. 表 5-6 和表 5-7 中的时钟数是舍入小数部分的时钟数。
例 主系统时钟从高速系统时钟切换到高速内部振荡器时钟的情况（选择 $f_{IH}=8\text{MHz}$ 、 $f_{MX}=10\text{MHz}$ 振荡时）
 $2f_{MX}/f_{IH}=2(10/8)=2.5 \rightarrow 3$ 个时钟

5.6.7 时钟振荡停止前的条件

停止时钟振荡（外部时钟输入无效）的寄存器的标志设定和停止前的条件如下所示。

表 5-8 时钟振荡停止前的条件和标志设定

时钟	时钟停止前的条件（外部时钟输入无效）	SFR 寄存器的标志设定
高速内部振荡器时钟	MCS=1 或者 CLS=1 (CPU 以高速内部振荡器时钟以外的时钟运行)	HIOSTOP=1
X1 时钟	MCS=0 或者 CLS=1 (CPU 以高速系统时钟以外的时钟运行)	MSTOP=1
外部主系统时钟		
XT1 时钟	CLS=0 (CPU 以副系统时钟以外的时钟运行)	XTSTOP=1
外部副系统时钟		

5.7 谐振器和振荡电路常数

经工作验证的谐振器及其振荡电路常数（供参考）如下所示。

注意 1. 此振荡电路常数是基于谐振器厂商在特定环境下进行评估的参考值。在实际应用中，请谐振器厂商给予安装电路中的评估。

在更改其他产品的单片机和电路板时，请谐振器厂商再次给予安装电路中的评估。

2. 振荡电压和振荡频率原则上是表示振荡电路的特性。有关 RL78 微控制器的内部工作条件，请在 DC、AC 特性的规格内使用。

图 5-17 外接振荡电路的例子



(1) X1 振荡

截至 2014 年 3 月 (1/2)

厂商	谐振器	产品名称注 3	SMD/ 引线	频率 (MHz)	闪存运行 模式注 1	振荡电路常数注 2 (参考)			电压范围 (V)	
						C1(pF)	C2(pF)	Rd(kΩ)	MIN.	MAX.
株式会社 村田制作 所注 3	陶瓷 谐振器	CSTCC2M00G56-R0	SMD	2.0	LV	(47)	(47)	0	1.6	5.5
		CSTCR4M00G55-R0	SMD	4.0		(39)	(39)	0		
		CSTLS4M00G53-B0	引线			(15)	(15)	0		
		CSTCC2M00G56-R0	SMD	2.0	LS	(47)	(47)	0	1.8	5.5
		CSTCR4M00G55-R0	SMD	4.0		(39)	(39)	0		
		CSTLS4M00G53-B0	引线			(15)	(15)	0		
		CSTCR4M19G55-R0	SMD	4.194		(39)	(39)	0		
		CSTLS4M19G53-B0	引线			(15)	(15)	0		
		CSTCR4M91G53-R0	SMD	4.915		(15)	(15)	0		
		CSTLS4M91G53-B0	引线			(15)	(15)	0		
		CSTCR5M00G53-R0	SMD	5.0		(15)	(15)	0		
		CSTLS5M00G53-B0	引线			(15)	(15)	0		
		CSTCR6M00G53-R0	SMD	6.0		(15)	(15)	0		
		CSTLS6M00G53-B0	引线			(15)	(15)	0		
		CSTCE8M00G52-R0	SMD	8.0		(10)	(10)	0		
		CSTLS8M00G53-B0	引线			(15)	(15)	0		
		CSTCE8M38G52-R0	SMD	8.388	HS	(10)	(10)	0	2.4	5.5
		CSTLS8M38G53-B0	引线			(15)	(15)	0		
		CSTCE10M0G52-R0	SMD	10.0		(10)	(10)	0		
		CSTLS10M0G53-B0	引线			(15)	(15)	0		
		CSTCE12M0G52-R0	SMD	12.0		(10)	(10)	0		
		CSTCE16M0V53-R0	SMD	16.0		(15)	(15)	0		
		CSTLS16M0X51-B0	引线			(5)	(5)	0		
		CSTCE20M0V51-R0	SMD	20.0		(5)	(5)	0	2.7	5.5
		CSTLS20M0X51-B0	引线			(5)	(5)	0		

注 1. 通过选项字节 (000C2H) 的 CMODE1 位和 CMODE0 位设定闪存运行模式。

2. C1 和 C2 栏的 () 内表示内部电容值。

3. 在使用此振荡器时, 有关匹配的详细内容, 请向株式会社村田制作所 (<http://www.murata-china.com>) 询问。

备注 工作电压范围、CPU 工作频率和运行模式的关系如下所示:

HS (高速主) 模式: $2.7V \leq V_{DD} \leq 5.5V @ 1MHz \sim 24MHz$

$2.4V \leq V_{DD} \leq 5.5V @ 1MHz \sim 16MHz$

LS (低速主) 模式: $1.8V \leq V_{DD} \leq 5.5V @ 1MHz \sim 8MHz$

LV (低电压主) 模式: $1.6V \leq V_{DD} \leq 5.5V @ 1MHz \sim 4MHz$

截至 2013 年 3 月 (2/2)

厂商	谐振器	产品名称注 2	SMD/ 引线	频率 (kHz)	闪存运行 模式注 1	振荡电路常数 (参考)			电压范围 (V)	
						C1(pF)	C2(pF)	Rd(kΩ)	MIN.	MAX.
日本电波 工业	晶体 振荡器	NX8045GB 注 3	SMD	8.0	注 3					
		NX5032GA 注 3	SMD	16.0						
		NX3225HA 注 3	SMD	20.0						
KYOCERA Crystal Device	晶体 振荡器	CX8045GB04000D0PPTZ1 注 4	SMD	4.0	LV	12	12	0	1.6	5.5
			SMD		LS				1.8	5.5
		CX8045GB04915D0PPTZ1 注 4	SMD	4.915	LS	12	12	0	1.8	5.5
		CX8045GB08000D0PPTZ1 注 4	SMD	8.0		12	12	0		
		CX8045GB10000D0PPTZ1 注 4	SMD	10.0	HS	12	12	0	2.4	5.5
		CX3225GB12000B0PPTZ1 注 4	SMD	12.0		5	5	0		
		CX3225GB16000B0PPTZ1 注 4	SMD	16.0		5	5	0		
		CX3225SB20000B0PPTZ1 注 4	SMD	20.0		5	5	0	2.7	5.5
RIVER ELETEC CORPORATION	晶体 振荡器	FCX-03-8.000MHZ-J21140 注 4	SMD	8.0	HS	3	3	0	2.4	5.5
		FCX-04C-10.000MHZ-J21139 注 4	SMD	10.0		4	4	0		
		FCX-05-12.000MHZ-J21138 注 4	SMD	12.0		6	6	0		
		FCX-06-16.000MHZ-J21137 注 4	SMD	16.0		4	4	0		

注 1. 通过选项字节 (000C2H) 的 CMODE1 位和 CMODE0 位设定闪存运行模式。

2. 此振荡器支持到 85°C。

3. 在使用此振荡器时, 有关匹配的详细内容, 请向日本电波工业 (<http://www.ndk.com>) 询问。

4. 在使用此振荡器时, 有关匹配的详细内容, 请向 KYOCERA Crystal Device (<http://www.kyocera-crystal.jp/eng>、<http://global.kyocera.com>) 询问。

5. 在使用此振荡器时, 有关匹配的详细内容, 请向 RIVER ELETEC CORPORATION (<http://www.river-ele.co.jp/english>) 询问。

备注 工作电压范围、CPU 工作频率和运行模式的关系如下所示:

HS (高速主) 模式: $2.7V \leq V_{DD} \leq 5.5V @ 1MHz \sim 24MHz$
 $2.4V \leq V_{DD} \leq 5.5V @ 1MHz \sim 16MHz$

LS (低速主) 模式: $1.8V \leq V_{DD} \leq 5.5V @ 1MHz \sim 8MHz$

LV (低电压主) 模式: $1.6V \leq V_{DD} \leq 5.5V @ 1MHz \sim 4MHz$

(2) XT1 振荡 (晶体振荡器)

截至 2013 年 3 月

厂商	产品名称注 2	SMD/ 引线	频率 (kHz)	负载 电容 C _L (pF)	XT1 振荡 模式注 1	振荡电路常数 (参考)			电压范围 (V)	
						C3(pF)	C4(pF)	Rd(kΩ)	MIN.	MAX.
精工电子 有限公司	SSP-T7-F 注 3	SMD	32.768	7	通常振荡	11	11	0	1.6	5.5
	SSP-T7-FL 注 3			6	低功耗振荡	9	9	0		
				6		9	9	0		
				4.4		6	5	0		
				4.4		超低功耗振荡	6	5		
				3.7	4	4	0			
	VT-200-FL 注 3	引线		6	通常振荡	9	9	0		
				6	低功耗振荡	9	9	0		
				4.4	超低功耗振荡	6	5	0		
				4.4		6	5	0		
				3.7		4	4	0		
日本电波 工业	NX3215SA 注 4	SMD	32.768	6	通常振荡	7	7	0	1.6	5.5
					低功耗振荡					
					超低功耗振荡					
	NX2012SA 注 4	SMD	32.768	6	通常振荡	7	7	0		
					低功耗振荡					
					超低功耗振荡					
KYOCERA Crystal Device	ST3215SB 注 5	SMD	32.768	7	通常振荡	10	10	0	1.6	5.5
					低功耗振荡					
					超低功耗振荡					
RIVER ELETEC CORPORATION	TFX-02-32.768KHZ- J20986 注 6	SMD	32.768	9	通常振荡	12	10	0	1.6	5.5
					低功耗振荡					
	TFX-03-32.768KHZ- J13375 注 6	SMD	32.768	7	通常振荡	12	10	0		

注 1. 通过时钟运行模式控制寄存器 (CMC) 的 AMPHS0 位和 AMPHS1 位设定 XT1 振荡模式。

2. 此振荡器支持到 85°C。

3. 此振荡器是用于低功耗的产品。在使用时, 有关匹配的详细内容, 请向精工电子有限公司 (<http://www.sii-crystal.com>) 询问。

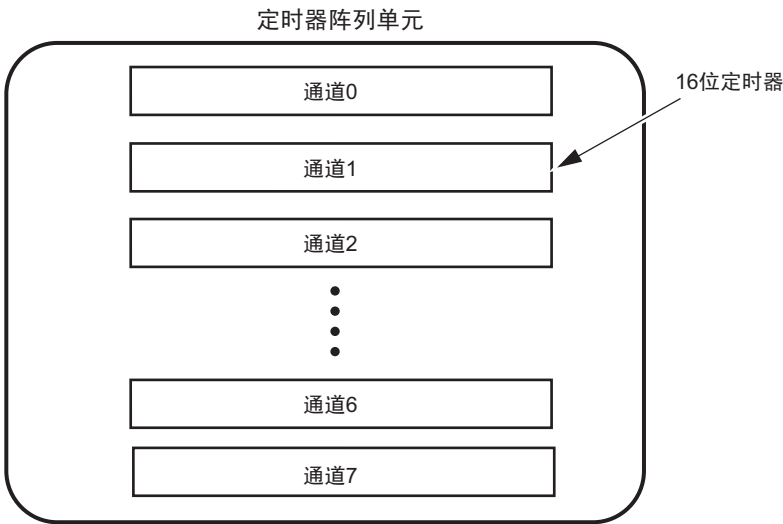
4. 在使用此振荡器时, 有关匹配的详细内容, 请向日本电波工业 (<http://www.ndk.com>) 询问。

5. 在使用此振荡器时, 有关匹配的详细内容, 请向 KYOCERA Crystal Device (<http://www.kyocera-crystal.jp/eng>、<http://global.kyocera.com>) 询问。

6. 在使用此振荡器时, 有关匹配的详细内容, 请向 RIVER ELETEC CORPORATION (<http://www.river-ele.co.jp/english>) 询问。

第 6 章 定时器阵列单元

定时器阵列单元有 8 个 16 位定时器。
各 16 位定时器称为“通道”，既能分别用作独立的定时器，也能组合多个通道用作高级的定时器功能。



有关每个功能的详细内容，请参照下表。

单通道运行功能	多通道联动运行功能
• 间隔定时器（→ 参照 6.8.1） • 方波输出（→ 参照 6.8.1） • 外部事件计数器（→ 参照 6.8.2） • 输入脉冲间隔测量（→ 参照 6.8.3） • 输入信号的高 / 低电平宽度的测量（→ 参照 6.8.4） • 延迟计数器（→ 参照 6.8.5）	• 单触发脉冲输出（→ 参照 6.9.1） • PWM 输出（→ 参照 6.9.2） • 多重 PWM 输出（→ 参照 6.9.3）

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）。可将通道 1 和通道 3 用作 8 位定时器的功能如下：

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

通道 7 能和串行阵列单元的 UART0 一起使用，实现 LIN-bus 通信。

6.1 定时器阵列单元的功能

定时器阵列单元有以下功能。

6.1.1 独立通道运行功能

独立通道运行功能是能不受其他通道运行模式的影响而独立使用任意通道的功能。

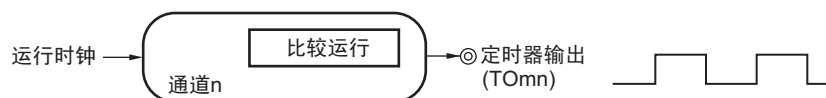
(1) 间隔定时器

能用作以固定间隔产生中断（INTTMmn）的基准定时器。



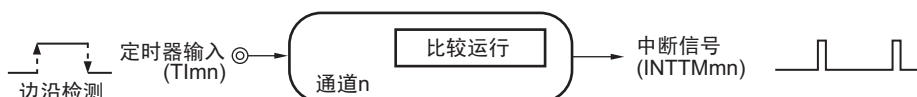
(2) 方波输出

每当产生 INTTMmn 中断时，就进行交替运行并且从定时器输出引脚（TOMn）输出 50% 占空比的方波。



(3) 外部事件计数器

对定时器输入引脚（TIMn）的输入信号的有效边沿进行计数，如果达到规定次数，就能用作产生中断的事件计数器。



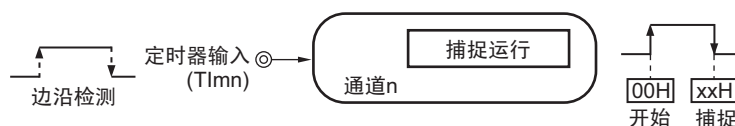
(4) 输入脉冲间隔测量

在定时器输入引脚（TIMn）的输入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。



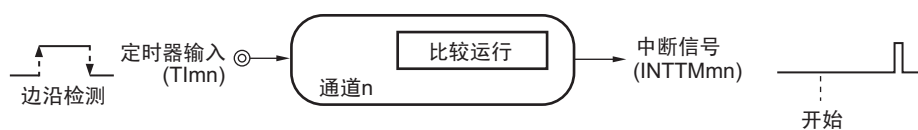
(5) 输入信号的高 / 低电平宽度的测量

在定时器输入引脚（TIMn）的输入信号的一个边沿开始计数并且在另一个边沿捕捉计数值，从而测量输入信号的高电平或者低电平的宽度。



(6) 延迟计数器

在定时器输入引脚（TIMn）的输入信号的有效边沿开始计数并且在经过任意延迟期间后产生中断。



备注 m: 单元号 (m = 0)、n: 通道号 (n = 0 ~ 7)

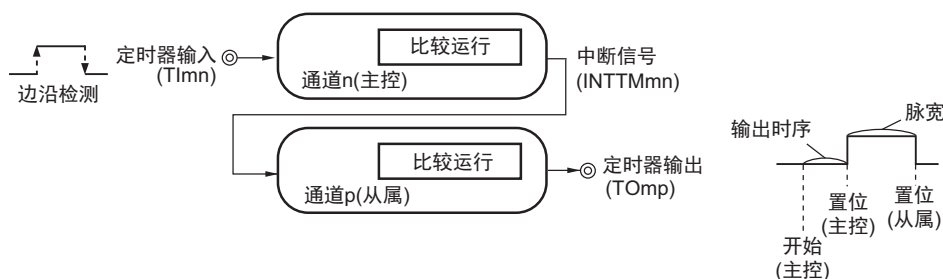
6.1.2 多通道联动运行功能

多通道联动运行功能是将主控通道（主要控制周期的基准定时器）和从属通道（遵从主控通道运行的定时器）组合实现的功能。

多通道联动运行功能能用作以下模式。

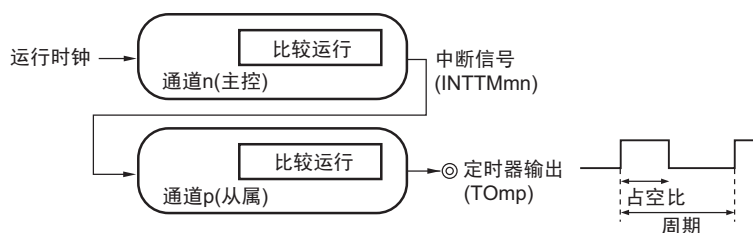
(1) 单触发脉冲输出

将 2 个通道成对使用，生成能任意设定输出时序和脉宽的单触发脉冲。



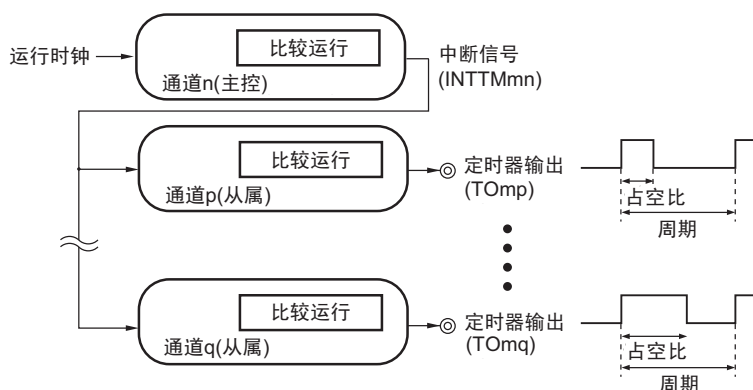
(2) PWM (Pulse Width Modulation) 输出

将 2 个通道成对使用，生成能任意设定周期和占空比的脉冲。



(3) 多重 PWM (Pulse Width Modulation) 输出

能通过扩展 PWM 功能并且使用 1 个主控通道和多个从属通道，以固定周期生成最多 7 种任意占空比的 PWM 信号。



备注 m: 单元号 (m = 0)、n: 通道号 (n = 0 ~ 7)

p、q: 从属通道号 (n < p < q ≤ 7)

6.1.3 8 位定时器运行功能（只限通道 1 和通道 3）

8 位定时器运行功能是将 16 位定时器通道用作 2 个 8 位定时器通道的功能。只能使用通道 1 和通道 3。

注意 在使用 8 位定时器运行功能时，有几个规则。

详细内容请参照“6.4.2 8 位定时器运行功能的基本规则（只限通道 1 和通道 3）”。

6.1.4 支持 LIN-bus 功能（只限通道 7）

通过定时器阵列单元检查 LIN-bus 通信中的接收信号是否适合 LIN-bus 通信格式。

(1) 唤醒信号的检测

在 UART0 串行数据输入引脚（RxD0）的输入信号的下降沿开始计数并且在上升沿捕捉计数值，从而测量低电平宽度。如果该低电平宽度大于等于某固定值，就认为是唤醒信号。

(2) 间隔场的检测

在检测到唤醒信号后，从 UART0 串行数据输入引脚（RxD0）的输入信号的下降沿开始计数并且在上升沿捕捉计数值，从而测量低电平宽度。如果该低电平宽度大于等于某固定值，就认为是间隔场。

(3) 同步场的脉宽测量

在检测到间隔场后，测量 UART0 串行数据输入引脚（RxD0）的输入信号的低电平宽度和高电平宽度。根据以此方式测量的同步场的位间隔，计算波特率。

备注 有关 LIN-bus 支持功能的运行设定，请参照“6.3.13 输入切换控制寄存器（ISC）”和“6.8.4 作为输入信号高 / 低电平宽度测量的运行”。

6.2 定时器阵列单元的结构

定时器阵列单元由以下硬件构成。

表 6-1 定时器阵列单元的结构

项目	结构
定时器 / 计数器	定时器计数寄存器 mn (TCRmn)
寄存器	定时器数据寄存器 mn (TDRmn)
定时器输入	TI00 ~ TI07、RxD0 引脚 (用于 LIN-bus)
定时器输出	TO00 ~ TO07 引脚, 输出控制电路
控制寄存器	< 单元设定部的寄存器 > 外围允许寄存器 0 (PER0) 定时器时钟选择寄存器 m (TPSm) 定时器通道允许状态寄存器 m (TEm) 定时器通道开始寄存器 m (TSm) 定时器通道停止寄存器 m (TTm) 定时器输入选择寄存器 0 (TIS0) 定时器输出允许寄存器 m (TOEm) 定时器输出寄存器 m (TOm) 定时器输出电平寄存器 m (TOLm) 定时器输出模式寄存器 m (TOMm) < 每个通道的寄存器 > 定时器模式寄存器 mn (TMRmn) 定时器状态寄存器 mn (TSRmn) 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 (PMCxx) 注 端口模式寄存器 (PMxx) 注 端口寄存器 (Pxx) 注

注 设定的端口模式控制寄存器 (PMCxx)、端口模式寄存器 (PMxx) 和端口寄存器 (Pxx) 因产品而不同。详细内容请参照“6.3.15 定时器输入 / 输出引脚的端口功能控制寄存器”

备注 m: 单元号 (m = 0)、n: 通道号 (n = 0 ~ 7)

定时器阵列单元中各通道的定时器输入 / 输出引脚的端口复用如下所示。

表 6-2 定时器输入 / 输出引脚

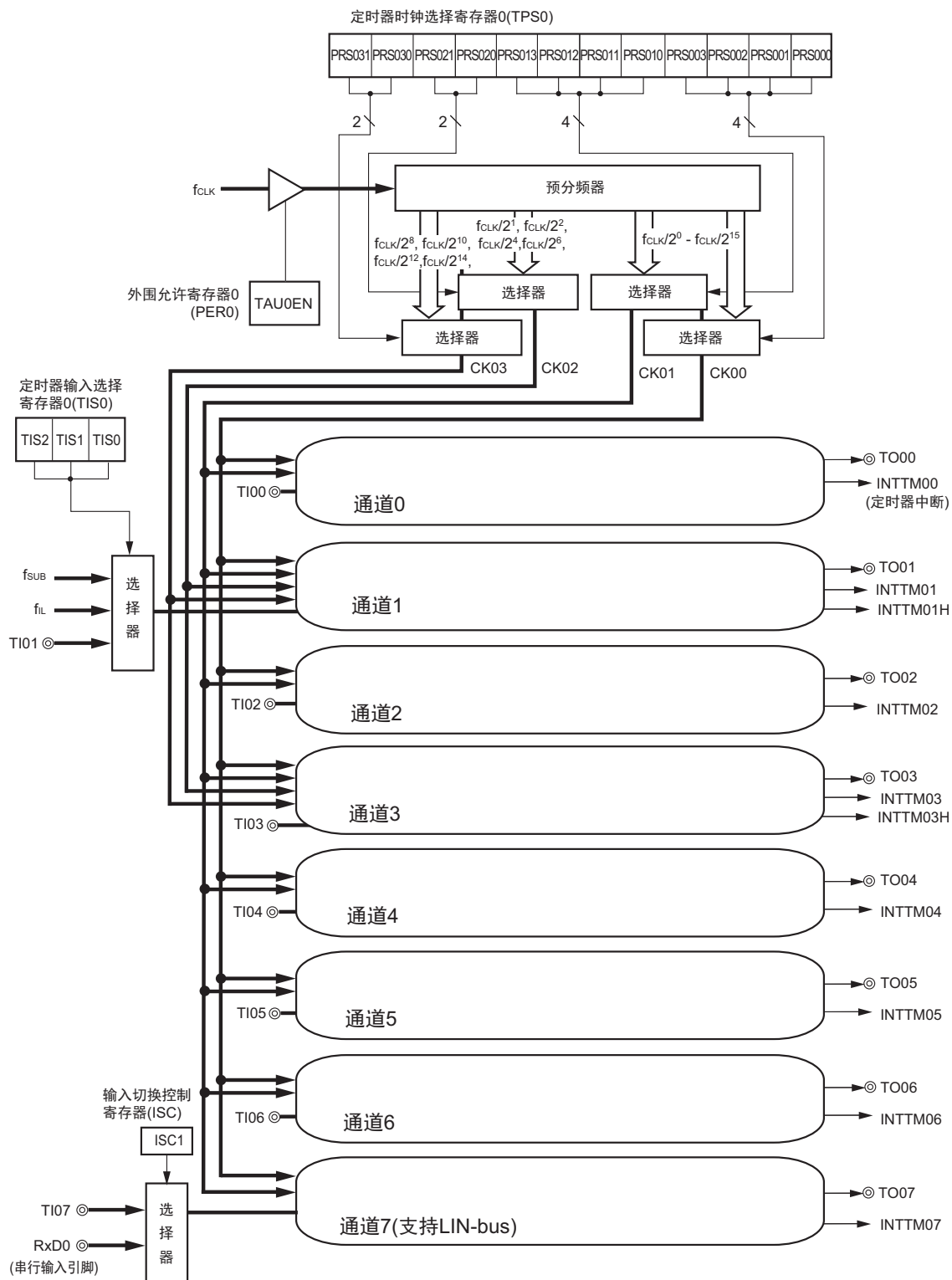
定时器阵列单元通道	定时器输入 / 输出引脚
通道 0	P52/TI00/TO00 (P40)
通道 1	P32/TI01/TO01 (P60)
通道 2	P54/TI02/TO02 (P61)
通道 3	P30/TI03/TO03 (P127)
通道 4	P14/TI04/TO04 (P126)
通道 5	P42/TI05/TO05 (P01)
通道 6	P56/TI06/TO06 (P125)
通道 7	P15/TI07/TO07 (P41)

备注 1. 在定时器输入和定时器输出被同一个引脚复用时，只能用作定时器输入或者定时器输出。

2. () 是将外围 I/O 重定向寄存器 (PIOR) 的 bit0 置“1”时的复用端口。

定时器阵列单元的框图如图 6-1 所示。

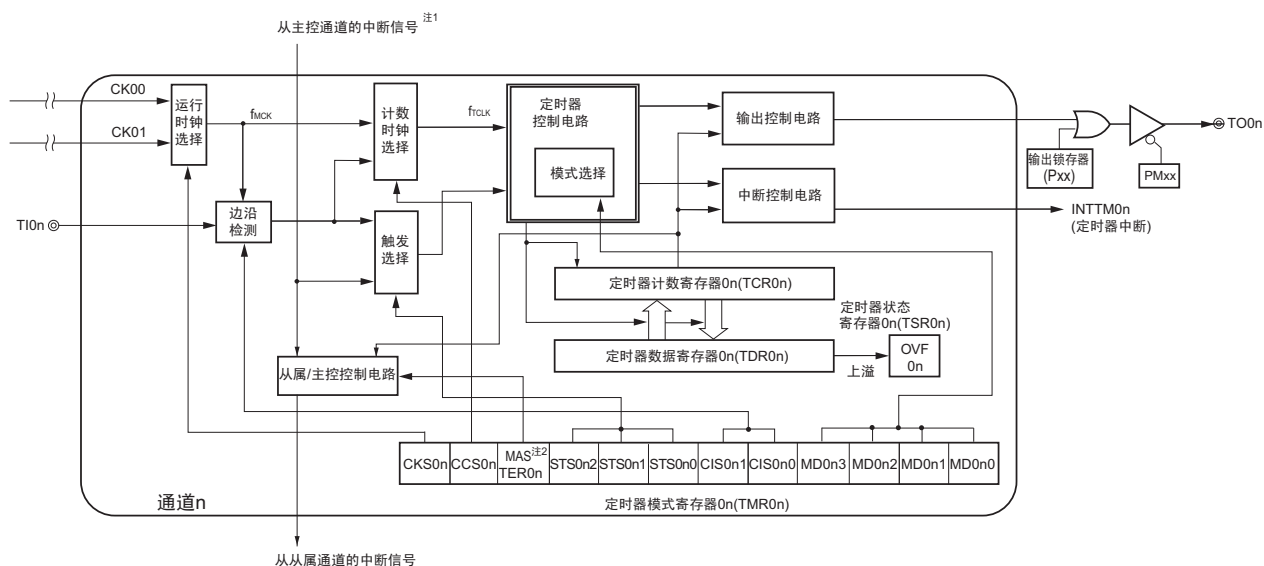
图 6-1 定时器阵列单元的整体框图



备注 f_{SUB} : 副系统时钟频率

f_{IL} : 低速内部振荡器时钟频率

图 6-2 定时器阵列单元的通道 0、2、4、6 的内部框图



注 1. 只限通道 2、4、6

2. 只限 $n = 2, 4, 6$

备注 $n = 0, 2, 4, 6$

图 6-3 定时器阵列单元的通道 1 的内部框图

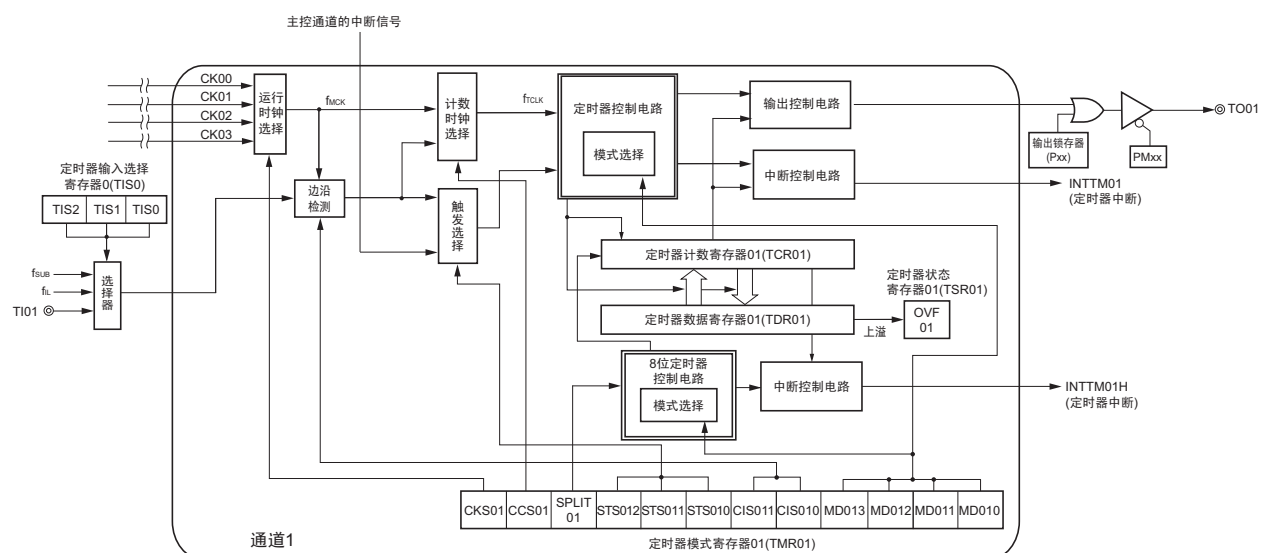


图 6-4 定时器阵列单元的通道 3 的内部框图

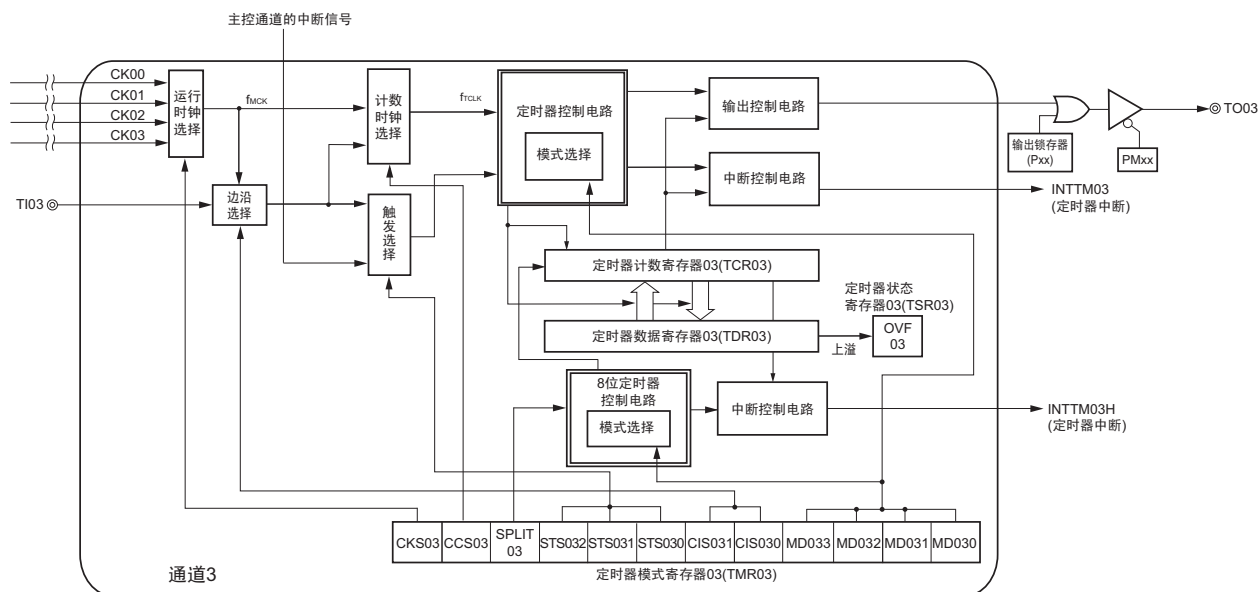


图 6-5 定时器阵列单元的通道 5 的内部框图

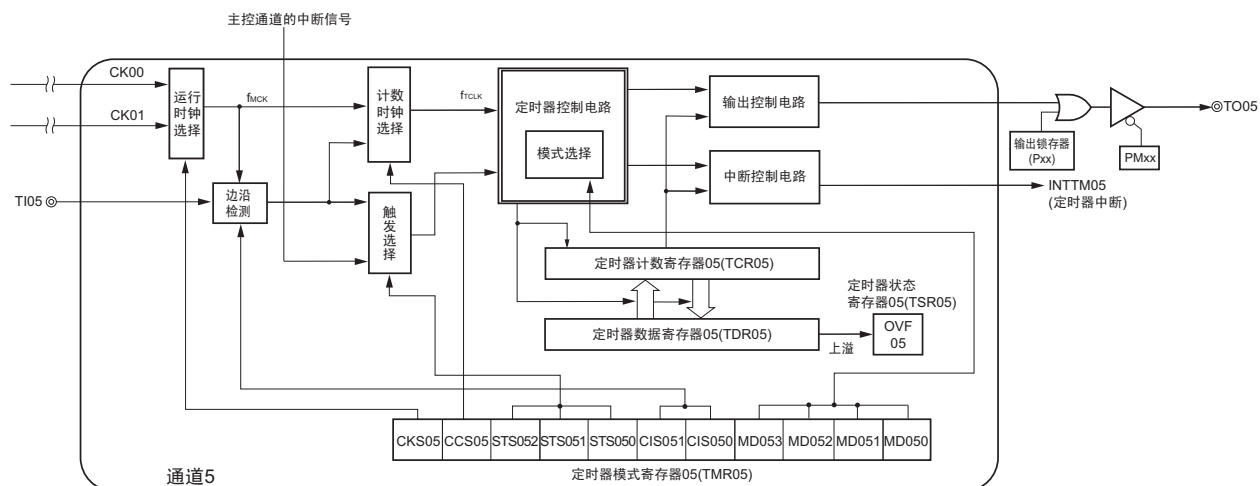
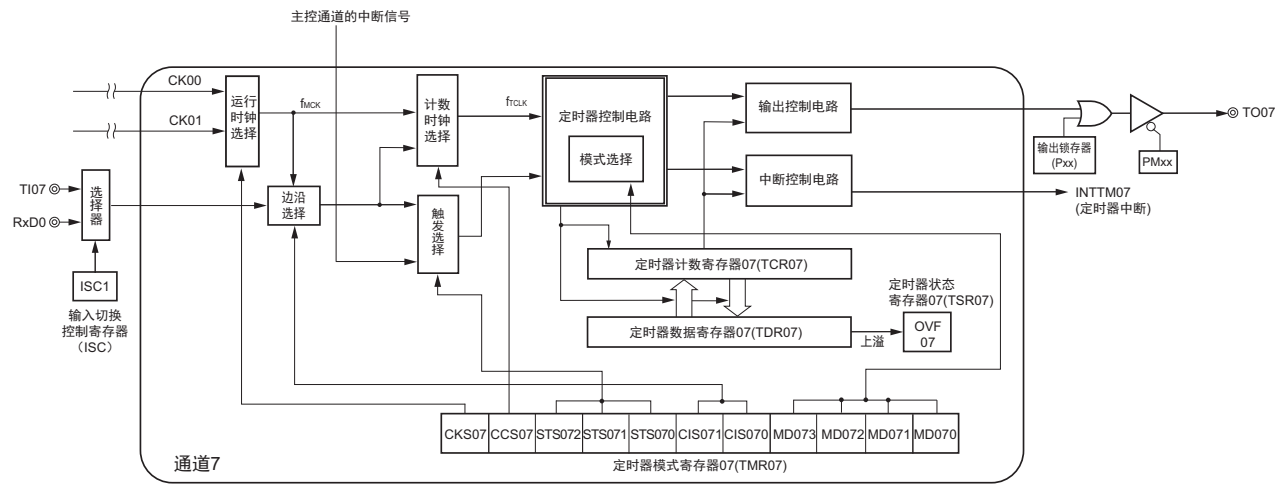


图 6-6 定时器阵列单元的通道 7 的内部框图



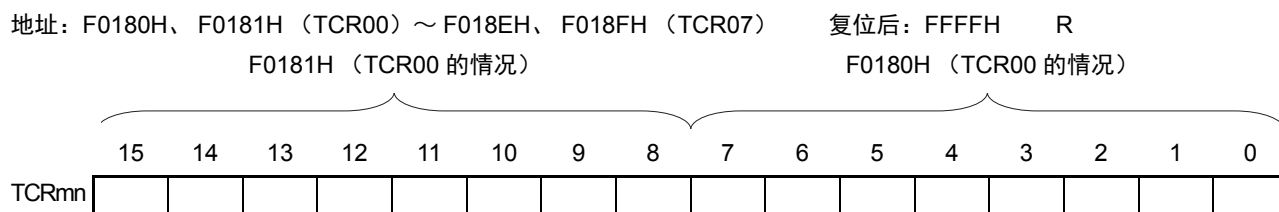
6.2.1 定时器计数寄存器 mn (TCRmn)

TCRmn 寄存器是对计数时钟进行计数的 16 位只读寄存器。

与计数时钟的上升沿同步进行递增或者递减计数。

通过定时器模式寄存器 mn (TMRmn) 的 MDmn3 ~ MDmn0 位来选择运行模式，进行递增和递减计数的切换（参照“6.3.3 定时器模式寄存器 mn (TMRmn)”）。

图 6-7 定时器计数寄存器 mn (TCRmn) 的格式



备注 m：单元号 (m = 0)、n：通道号 (n = 0 ~ 7)

能通过读定时器计数寄存器 mn (TCRmn) 来读计数值。

在以下情况下，计数值变为“FFFFH”。

- 当产生复位信号时
- 当清除外围允许寄存器0 (PER0) 的TAUmEN位时
- 在PWM输出模式中从属通道的计数结束时
- 在延迟计数模式中从属通道的计数结束时
- 在单脉冲输出模式中主控/从属通道的计数结束时
- 在多重PWM输出模式中从属通道的计数结束时

在以下情况下，计数值变为“0000H”。

- 在捕捉模式中输入开始触发时
- 在捕捉模式中捕捉结束时

注意 即使读 TCRmn 寄存器，也不将计数值捕捉到定时器数据寄存器 mn (TDRmn)。

如下所示，TCRmn 寄存器的读取值因运行模式和运行状态而不同。

表 6-3 各运行模式中的定时器计数寄存器 mn (TCRmn) 的读取值

运行模式	计数方式	定时器计数寄存器 mn (TCRmn) 的读取值 ^注			
		解除复位后更改运行模式时的值	计数暂停 (TTmn=1) 时的值	计数暂停 (TTmn=1) 后更改运行模式时的值	单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	停止时的值	不定	—
捕捉模式	递增计数	0000H	停止时的值	不定	—
事件计数器模式	递减计数	FFFFH	停止时的值	不定	—
单次计数模式	递减计数	FFFFH	停止时的值	不定	FFFFH
捕捉 & 单次计数模式	递增计数	0000H	停止时的值	不定	TDRmn 寄存器的捕捉值 + 1

注 表示通道 n 为定时器运行停止状态 (TEmn=0) 并且计数允许状态 (TSmn=1) 时的 TCRmn 寄存器的读取值。将此值保持在 TCRmn 寄存器，直到开始计数为止。

备注 m：单元号 (m = 0)、n：通道号 (n = 0 ~ 7)

6.2.2 定时器数据寄存器 mn (TDRmn)

这是能进行捕捉功能和比较功能切换使用的 16 位寄存器。通过定时器模式寄存器 mn (TMRmn) 的 MDmn3 ~ MDmn0 位来选择运行模式，进行捕捉功能和比较功能的切换。

能随时改写 TDRmn 寄存器的值。

能以 16 位为单位读写此寄存器。

在 8 位定时器模式中（定时器模式寄存器 m1、m3 (TMRm1、TMRm3) 的 SPLIT 位为“1”），能以 8 位为单位读写 TDRm1 寄存器和 TDRm3 寄存器，其中 TDRm1H 和 TDRm3H 用作高 8 位，TDRm1L 和 TDRm3L 用作低 8 位。

在产生复位信号后，TDRmn 寄存器的值变为“0000H”。

图 6-8 定时器数据寄存器 mn (TDRmn) 的格式 (n=0、2、4~7)

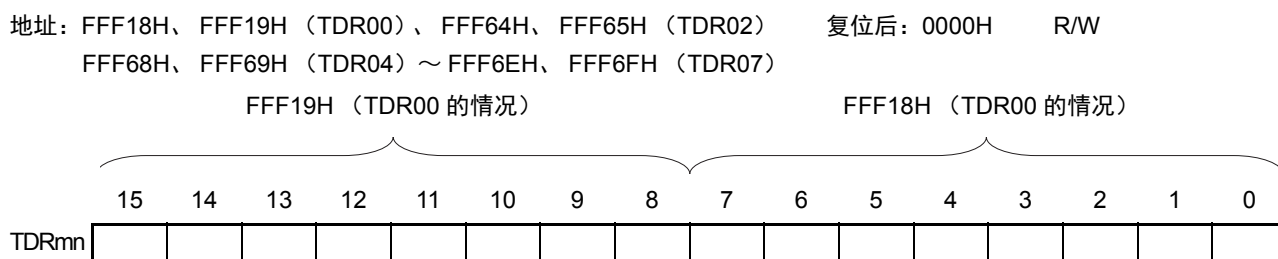
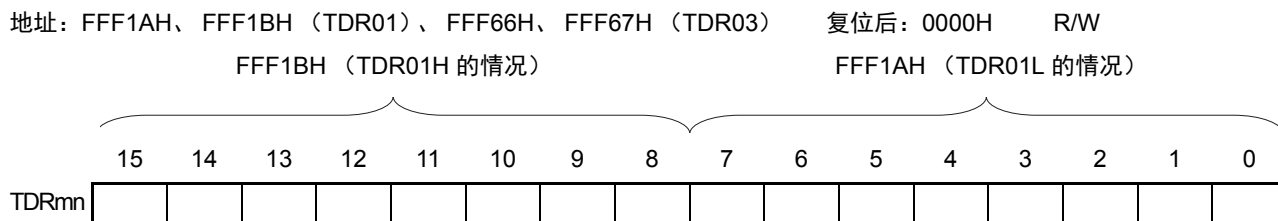


图 6-9 定时器数据寄存器 mn (TDRmn) 的格式 (n=1、3)



(i) 定时器数据寄存器 mn (TDRmn) 用作比较寄存器的情况

从 TDRmn 寄存器的设定值开始递减计数，当计数值变为“0000H”时，产生中断信号 (INTTMmn)。保持 TDRmn 寄存器的值，直到被改写为止。

注意 即使输入捕捉触发信号，设定为比较功能的 TDRmn 寄存器也不进行捕捉运行。

(ii) 定时器数据寄存器 mn (TDRmn) 用作捕捉寄存器的情况

通过输入捕捉触发，将定时器计数寄存器 mn (TCRmn) 的计数值捕捉到 TDRmn 寄存器。

能选择 TIImn 引脚的有效边沿作为捕捉触发信号。通过定时器模式寄存器 mn (TMRmn) 来设定捕捉触发的选择。

备注 m: 单元号 (m=0) n: 通道号 (n=0~7)

6.3 控制定时器阵列单元的寄存器

控制定时器阵列单元的寄存器如下所示。

- 外围允许寄存器0 (PER0)
- 定时器时钟选择寄存器m (TPSm)
- 定时器模式寄存器mn (TMRmn)
- 定时器状态寄存器mn (TSRmn)
- 定时器通道允许状态寄存器m (TEm)
- 定时器通道开始寄存器m (TSm)
- 定时器通道停止寄存器m (TTm)
- 定时器输入选择寄存器0 (TIS0)
- 定时器输出允许寄存器m (TOEm)
- 定时器输出寄存器m (TOM)
- 定时器输出电平寄存器m (TOLm)
- 定时器输出模式寄存器m (TOMm)
- 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 (PMCxx)
- 端口模式寄存器 (PMxx)
- 端口寄存器 (Pxx)

注意 配置的寄存器和位因产品而不同。必须将未配置的位设定为初始值。

备注 m: 单元号 (m = 0)、n: 通道号 (n = 0 ~ 7)

6.3.1 外围允许寄存器 0（PER0）

PER0 寄存器是设定允许或者禁止给各外围硬件提供时钟的寄存器。通过停止给不使用的硬件提供时钟，降低功耗和噪声。

要使用定时器阵列单元时，必须将 bit0（TAU0EN）置“1”。

通过 1 位或者 8 位存储器操作指令设定 PER0 寄存器。

在产生复位信号后，PER0 寄存器的值变为“00H”。

图 6-10 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	RTCWEN	0	ADCEN	IICA0EN	SAU1EN	SAU0EN	0	TAU0EN

TAU0EN	控制定时器阵列单元的输入时钟
0	停止提供输入时钟。 - 不能写定时器阵列单元使用的 SFR。 - 定时器阵列单元处于复位状态。
1	提供输入时钟。 可以读取 / 写入用于定时器阵列单元的 SFR。

注意 1. 要设定定时器阵列单元时，必须在 TAU0EN 位为“1”的状态下，设定以下寄存器。当 TAU0EN 位为“0”时，定时器阵列单元的控制寄存器为初始值，并且忽视写操作（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 4（PMC4）、端口模式寄存器 0、1、3、4、5（PM0、PM1、PM3、PM4、PM5）和端口寄存器 0、1、3、4、5（P0、P1、P3、P4、P5）除外）。

- 定时器时钟选择寄存器 m（TPSm）
- 定时器模式寄存器 mn（TMRmn）
- 定时器状态寄存器 mn（TSRmn）
- 定时器通道允许状态寄存器 m（TEm）
- 定时器通道开始寄存器 m（TSm）
- 定时器通道停止寄存器 m（TTm）
- 定时器输出允许寄存器 m（TOEm）
- 定时器输出寄存器 m（TOM）
- 定时器输出电平寄存器 m（TOLm）
- 定时器输出模式寄存器 m（TOMm）

2. 必须将 bit1 和 bit6 置“0”。

6.3.2 定时器时钟选择寄存器 m (TPSm)

TPSm 寄存器是选择共同提供给各通道的 2 种或者 4 种运行时钟 (CKm0、CKm1、CKm2、CKm3) 的 16 位寄存器。通过 TPSm 寄存器的 bit3 ~ 0 选择 CKm0, 通过 TPSm 寄存器的 bit7 ~ 4 选择 CKm1。另外, 对于通道 1 和通道 3 还可选择 CKm2 和 CKm3, 通过 TPSm 寄存器的 bit9 和 bit8 选择 CKm2, 通过 TPSm 寄存器的 bit13 和 bit12 选择 CKm3。

只有在以下情况下才能改写定时器运行中的 TPSm 寄存器。

能改写 PRSm00 ~ PRSm03 位的情况 (n=0 ~ 7):

选择 CKm0 作为运行时钟 (CKSmn1、CKSmn0=0、0) 的通道全部处于停止状态 (TEmn=0)。

能改写 PRSm10 ~ PRSm13 位的情况 (n=0 ~ 7):

选择 CKm1 作为运行时钟 (CKSmn1、CKSmn0=0、1) 的通道全部处于停止状态 (TEmn=0)。

能改写 PRSm20 ~ PRSm21 位的情况 (n=1、3):

选择 CKm2 作为运行时钟 (CKSmn1、CKSmn0=1、0) 的通道全部处于停止状态 (TEmn=0)。

能改写 PRSm30 ~ PRSm31 位的情况 (n=1、3):

选择 CKm3 作为运行时钟 (CKSmn1、CKSmn0=1、1) 的通道全部处于停止状态 (TEmn=0)。

通过 16 位存储器操作指令设定 TPSm 寄存器。

在产生复位信号后, TPSm 寄存器的值变为“0000H”。

图 6-11 定时器时钟选择寄存器 m (TPSm) 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
TPSm	0	0	PRS m31	PRS m30	0	0	PRS m21	PRS m20	PRS m13	PRS m12	PRS m11	PRS m10	PRS m03	PRS m02	PRS m01	PRS m00

PRS mk3	PRS mk2	PRS mk1	PRS mk0	运行时钟 (CKmk) 的选择注 (k=0、1)					
					f _{CLK} = 2MHz	f _{CLK} = 5MHz	f _{CLK} = 10MHz	f _{CLK} = 20MHz	f _{CLK} = 24MHz
0	0	0	0	f _{CLK}	2MHz	5MHz	10MHz	20MHz	24MHz
0	0	0	1	f _{CLK} /2	1MHz	2.5MHz	5MHz	10MHz	12MHz
0	0	1	0	f _{CLK} /2 ²	500kHz	1.25MHz	2.5MHz	5MHz	6MHz
0	0	1	1	f _{CLK} /2 ³	250kHz	625kHz	1.25MHz	2.5MHz	3MHz
0	1	0	0	f _{CLK} /2 ⁴	125kHz	313kHz	625kHz	1.25MHz	1.50MHz
0	1	0	1	f _{CLK} /2 ⁵	62.5kHz	156kHz	313kHz	625kHz	750kHz
0	1	1	0	f _{CLK} /2 ⁶	31.3kHz	78.1kHz	156kHz	313kHz	375kHz
0	1	1	1	f _{CLK} /2 ⁷	15.6kHz	39.1kHz	78.1kHz	156kHz	188kHz
1	0	0	0	f _{CLK} /2 ⁸	7.81kHz	19.5kHz	39.1kHz	78.1kHz	93.8kHz
1	0	0	1	f _{CLK} /2 ⁹	3.91kHz	9.76kHz	19.5kHz	39.1kHz	46.9kHz
1	0	1	0	f _{CLK} /2 ¹⁰	1.95kHz	4.88kHz	9.76kHz	19.5kHz	23.4kHz
1	0	1	1	f _{CLK} /2 ¹¹	977Hz	2.44kHz	4.88kHz	9.76kHz	11.7kHz
1	1	0	0	f _{CLK} /2 ¹²	488Hz	1.22kHz	2.44kHz	4.88kHz	5.86kHz
1	1	0	1	f _{CLK} /2 ¹³	244Hz	610Hz	1.22kHz	2.44kHz	2.93kHz
1	1	1	0	f _{CLK} /2 ¹⁴	122Hz	305Hz	610Hz	1.22kHz	1.46kHz
1	1	1	1	f _{CLK} /2 ¹⁵	61.0Hz	153Hz	305Hz	610Hz	732Hz

注 在更改选择为 f_{CLK} 的时钟 (更改系统时钟控制寄存器 (CKC) 的值) 的情况下, 必须停止定时器阵列单元 (TTm=00FFH)。

注意 1. 必须将 bit15、14、11、10 置“0”。

2. 如果选择 f_{CLK} (无分频) 作为运行时钟 (CKmk) 并且将 TDRnm 置“0000H” (n=0、m=0 ~ 7), 就不能使用来自定时器阵列单元的中断请求。

备注 1. f_{CLK}: CPU/ 外围硬件的时钟频率

2. TPSm 寄存器选择的时钟波形从上升沿开始只有 1 个 f_{CLK} 周期为高电平 (m=1 ~ 15)。详细内容请参照“6.5.1 计数时钟 (f_{TCLK})”。

图 6-11 定时器时钟选择寄存器 m (TPSm) 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
TPSm	0	0	PRS m31	PRS m30	0	0	PRS m21	PRS m20	PRS m13	PRS m12	PRS m11	PRS m10	PRS m03	PRS m02	PRS m01	PRS m00

PRS m21	PRS m20	运行时钟 (CKm2) 的选择注					
			$f_{CLK}=2MHz$	$f_{CLK}=5MHz$	$f_{CLK}=10MHz$	$f_{CLK}=20MHz$	$f_{CLK}=24MHz$
0	0	$f_{CLK}/2$	1MHz	2.5MHz	5MHz	10MHz	12MHz
0	1	$f_{CLK}/2^2$	500kHz	1.25MHz	2.5MHz	5MHz	6MHz
1	0	$f_{CLK}/2^4$	125kHz	313kHz	625MHz	1.25MHz	1.5MHz
1	1	$f_{CLK}/2^6$	31.3kHz	78.1kHz	156kHz	313kHz	375kHz

PRS m31	PRS m30	运行时钟 (CKm3) 的选择注					
			$f_{CLK}=2MHz$	$f_{CLK}=5MHz$	$f_{CLK}=10MHz$	$f_{CLK}=20MHz$	$f_{CLK}=24MHz$
0	0	$f_{CLK}/2^8$	7.81kHz	19.5kHz	39.1kHz	78.1kHz	93.8kHz
0	1	$f_{CLK}/2^{10}$	1.95kHz	4.88kHz	9.76kHz	19.5kHz	23.4kHz
1	0	$f_{CLK}/2^{12}$	488Hz	1.22kHz	2.44kHz	4.88kHz	5.86kHz
1	1	$f_{CLK}/2^{14}$	122Hz	305Hz	610Hz	1.22kHz	1.46kHz

注 在更改选择为 f_{CLK} 的时钟 (更改系统时钟控制寄存器 (CKC) 的值) 的情况下, 必须停止定时器阵列单元 (TTm=00FFH)。即使在选择了 CKSmn0 位和 CKSmn1 位指定的运行时钟 (f_{MCK}) 或者 TImn 引脚的输入信号的任意有效边沿时, 也需要停止定时器阵列单元。

注意 必须将 bit15、14、11、10 置“0”。

如果在 8 位定时器模式中使用通道 1 和通道 3 并且将 CKm2 和 CKm3 作为运行时钟, 就能通过间隔定时器功能实现表 6-4 所示的间隔时间。

表 6-4 运行时钟 CKSm2 和 CKSm3 能设定的间隔时间

时钟		间隔时间注 ($f_{CLK}=20MHz$)			
		16μs	160μs	1.6ms	16ms
CKm2	$f_{CLK}/2$	○	—	—	—
	$f_{CLK}/2^2$	○	—	—	—
	$f_{CLK}/2^4$	○	○	—	—
	$f_{CLK}/2^6$	○	○	—	—
CKm3	$f_{CLK}/2^8$	—	○	○	—
	$f_{CLK}/2^{10}$	—	○	○	—
	$f_{CLK}/2^{12}$	—	—	○	○
	$f_{CLK}/2^{14}$	—	—	○	○

注 ○包含 5% 以内的误差。

备注 1. f_{CLK} : CPU/ 外围硬件的时钟频率

2. 有关 TPSm 寄存器所选 $f_{CLK}/2^i$ 波形的详细内容, 请参照“6.5.1 计数时钟 (f_{TCLK})”。

6.3.3 定时器模式寄存器 mn (TMRmn)

TMRmn 寄存器是设定通道 n 运行模式的寄存器，进行运行时钟 (f_{MCK}) 的选择、计数时钟的选择、主控 / 从属的选择、16 位 / 8 位定时器的选择（只限于通道 1 和通道 3）、开始触发和捕捉触发的设定、定时器输入有效边沿的选择以及运行模式（间隔、捕捉、事件计数器、单次计数、捕捉 & 单次计数）的设定。

禁止在运行中 (TEmn=1) 改写 TMRmn 寄存器。但是，能在一部分的功能运行中 (TEmn=1) 改写 bit7 和 bit6 (CISmn1、CISmn0)（详细内容请参照“6.8 定时器阵列单元的独立通道运行功能”和“6.9 定时器阵列单元的多通道联动运行功能”）。

通过 16 位存储器操作指令设定 TMRmn 寄存器。

在产生复位信号后，TMRmn 寄存器的值变为“0000H”。

注意 TMRmn 寄存器的 bit11 因通道而不同。

TMRm2、TMRm4、TMRm6: MASTERmn 位 (n=2、4、6)

TMRm1、TMRm3: SPLITmn 位 (n=1、3)

TMRm0、TMRm5、TMRm7: 固定为“0”。

图 6-12 定时器模式寄存器 mn (TMRmn) 的格式 (1/4)

地址: F0190H、F0191H (TMR00) ~ 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
TMRmn (n=2, 4, 6)	CKS mn1	CKS mn0	0	CCS mn	MAS TERmn	STS mn2	STS mn1	STS mn0	CIS mn1	CIS mn0	0	0	MD mn3	MD mn2	MD mn1	MD mn0

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n=1, 3)	CKS mn1	CKS mn0	0	CCS mn	SPLIT mn	STS mn2	STS mn1	STS mn0	CIS mn1	CIS mn0	0	0	MD mn3	MD mn2	MD mn1	MD mn0

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n=0, 5, 7)	CKS mn1	CKS mn0	0	CCS mn	0注	STS mn2	STS mn1	STS mn0	CIS mn1	CIS mn0	0	0	MD mn3	MD mn2	MD mn1	MD mn0

CKS mn1	CKS mn0	通道 n 运行时钟 (f_{MCK}) 的选择
0	0	定时器时钟选择寄存器 m (TPSm) 设定的运行时钟 CKm0
0	1	定时器时钟选择寄存器 m (TPSm) 设定的运行时钟 CKm2
1	0	定时器时钟选择寄存器 m (TPSm) 设定的运行时钟 CKm1
1	1	定时器时钟选择寄存器 m (TPSm) 设定的运行时钟 CKm3
运行时钟 (f_{MCK}) 用于边沿检测电路。通过设定 CCSmn 位来产生采样时钟和计数时钟 (f_{TCLK})。 只有通道 1 和通道 3 才能选择运行时钟 CKm2 和 CKm3。		

CCS mn	通道 n 计数时钟 (f_{TCLK}) 的选择
0	CKSmn0 位和 CKSmn1 位指定的运行时钟 (f_{MCK})
1	TImn 引脚的输入信号的有效边沿 在通道 1 的情况下, 为 TIS0 选择的输入信号的有效边沿。
计数时钟 (f_{TCLK}) 用于计数器、输出控制电路和中断控制电路。	

注 bit11 是只读位, 固定为“0”, 忽视写操作。

注意 1. 必须将 bit13、5、4 置“0”。

2. 要更改选择为 f_{CLK} 的时钟 (更改系统时钟控制寄存器 (CKC) 的值) 时, 即使选择了 CKSmn0 位和 CKSmn1 位指定的运行时钟 (f_{MCK}) 或者 TImn 引脚的输入信号的有效边沿作为计数时钟 (f_{TCLK}), 也必须停止定时器阵列单元 (TTm=00FFH)。

备注 m: 单元号 (m=0) n: 通道号 (n=0 ~ 7)

图 6-12 定时器模式寄存器 mn (TMRmn) 的格式 (2/4)

地址: F0190H、F0191H (TMR00) ~ 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
TMRmn (n=2, 4, 6)	CKS mn1	CKS mn0	0	CCS mn	MAS TERmn	STS mn2	STS mn1	STS mn0	CIS mn1	CIS mn0	0	0	MD mn3	MD mn2	MD mn1	MD mn0

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n=1, 3)	CKS mn1	CKS mn0	0	CCS mn	SPLIT mn	STS mn2	STS mn1	STS mn0	CIS mn1	CIS mn0	0	0	MD mn3	MD mn2	MD mn1	MD mn0

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n=0, 5, 7)	CKS mn1	CKS mn0	0	CCS mn	0注	STS mn2	STS mn1	STS mn0	CIS mn1	CIS mn0	0	0	MD mn3	MD mn2	MD mn1	MD mn0

(TMRmn (n=2、4、6) 的 bit11)

MAS TER mn	通道 n 的独立通道运行 / 多通道联动运行 (从属或者主控) 的选择
0	用作独立通道运行功能或者多通道联动运行功能的从属通道。
1	用作多通道联动运行功能的主控通道。
只有通道 2、4、6 才能设定主控通道 (MASTERmn=1)。 通道 0、5、7 固定为“0” (因为通道 0 为最高位通道, 所以与此位的设定无关, 用作主控通道)。 对于用作独立通道运行功能的通道, 将 MASTERmn 位置“0”。	

(TMRmn (n=1、3) 的 bit11)

SPLIT mn	通道 1 和通道 3 的 8 位定时器 / 16 位定时器的运行选择
0	用作 16 位定时器。 (用作独立通道运行功能或者多通道联动运行功能的从属通道)
1	用作 8 位定时器。

STS mn2	STS mn1	STS mn0	通道 n 的开始触发和捕捉触发的设定
0	0	0	只有软件触发开始有效 (不选择其他触发源)。
0	0	1	将 TImn 引脚输入的有效边沿用于开始触发和捕捉触发。
0	1	0	将 TImn 引脚输入的双边沿分别用于开始触发和捕捉触发。
1	0	0	使用主控通道的中断信号 (多通道联动运行功能的从属通道的情况)。
上述以外			禁止设定。

注 bit11 是只读位, 固定为“0”, 忽视写操作。

备注 m: 单元号 (m=0) n: 通道号 (n=0 ~ 7)

图 6-12 定时器模式寄存器 mn（TMRmn）的格式（3/4）

地址：F0190H、F0191H（TMR00）～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
TMRmn (n=2、4、6)	CKS mn1	CKS mn0	0	CCS mn	MAS TERmn	STS mn2	STS mn1	STS mn0	CIS mn1	CIS mn0	0	0	MD mn3	MD mn2	MD mn1	MD mn0

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n=1、3)	CKS mn1	CKS mn0	0	CCS mn	SPLIT mn	STS mn2	STS mn1	STS mn0	CIS mn1	CIS mn0	0	0	MD mn3	MD mn2	MD mn1	MD mn0

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n=0、5、7)	CKS mn1	CKS mn0	0	CCS mn	0注	STS mn2	STS mn1	STS mn0	CIS mn1	CIS mn0	0	0	MD mn3	MD mn2	MD mn1	MD mn0

CIS mn1	CIS mn0	Tlmn 引脚有效边沿的选择
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 STSmn2～STSmn0 位不为“010B”并且使用双边沿指定时，必须将 CISmn1～CISmn0 位置“10B”。		

注 bit11 是只读位，固定为“0”，忽视写操作。

备注 m：单元号（m=0）、 n：通道号（n=0～7）

图 6-12 定时器模式寄存器 mn (TMRmn) 的格式 (4/4)

地址: F0190H、F0191H (TMR00) ~ 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
TMRmn (n=2, 4, 6)	CKS mn1	CKS mn0	0	CCS mn	MAS TERmn	STS mn2	STS mn1	STS mn0	CIS mn1	CIS mn0	0	0	MD mn3	MD mn2	MD mn1	MD mn0

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n=1, 3)	CKS mn1	CKS mn0	0	CCS mn	SPLIT mn	STS mn2	STS mn1	STS mn0	CIS mn1	CIS mn0	0	0	MD mn3	MD mn2	MD mn1	MD mn0

符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n=0, 5, 7)	CKS mn1	CKS mn0	0	CCS mn	0 注 1	STS mn2	STS mn1	STS mn0	CIS mn1	CIS mn0	0	0	MD mn3	MD mn2	MD mn1	MD mn0

MD mn3	MD mn2	MD mn1	通道 n 运行模式的设定	对应功能	TCR 的计数运行
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 （主控）	递减计数
0	1	0	捕捉模式	输入脉冲间隔的测量	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	单次计数模式	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 & 单次计数模式	输入信号的高 / 低电平宽度的测量	递增计数
上述以外			禁止设定。		
各模式的运行因 MDmn0 位而变 （参照下表）。					

运行模式 (MDmn3 ~ MDmn1 位的设定 (参照上表))	MD mn0	开始计数和中断的设定
- 间隔定时器模式 (0、0、0) - 捕捉模式 (0、1、0)	0	在开始计数时不产生定时器中断 (定时器的输出也不发生变化)。
	1	在开始计数时产生定时器中断 (定时器的输出也发生变化)。
事件计数器模式 (0、1、1)	0	在开始计数时不产生定时器中断 (定时器的输出也不发生变化)。
单次计数模式注 2 (1、0、0)	0	计数运行中的开始触发无效。此时不产生中断。
	1	计数运行中的开始触发有效注 3。此时不产生中断。
捕捉 & 单次计数模式 (1、1、0)	0	在开始计数时不产生定时器中断 (定时器的输出也不发生变化)。计数运行中的开始触发无效。此时不产生中断。

注 1. bit11 是只读位, 固定为“0”, 忽视写操作。

2. 在单次计数模式中, 不控制开始计数时的中断输出 (INTTMmn) 和 TOMn 输出。

3. 如果在运行中发生开始触发 (TSmn=1), 就对计数器进行初始化并且重新开始计数 (不产生中断请求)。

备注 m: 单元号 (m=0)、n: 通道号 (n=0 ~ 7)

6.3.4 定时器状态寄存器 mn (TSRmn)

TSRmn 寄存器是表示通道 n 计数器的上溢状态的寄存器。

TSRmn 寄存器只在捕捉模式 (MDmn3~MDmn1=010B) 和捕捉&单次计数模式 (MDmn3~MDmn1=110B) 中有效。有关各运行模式中的 OVF 位的变化和设定 / 清除条件, 请参照表 6-5。

通过 16 位存储器操作指令读 TSRmn 寄存器。

能用 TSRmnL 并且通过 8 位存储器操作指令读 TSRmn 寄存器的低 8 位。

在产生复位信号后, TSRmn 寄存器的值变为“0000H”。

图 6-13 定时器状态寄存器 mn (TSRmn) 的格式

地址: F01A0H、F01A1H (TSR00) ~ 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

TSRmn

0

0

0

0

0

0

0

0

0

0

0

0

0

0

0

OVF

OVF

通道 n 的计数器上溢状态

0

没有发生上溢。

1

发生上溢。

如果 OVF 位为“1”，就在下一次计数不发生上溢并且捕捉到计数值时清除此标志（OVF=0）。

备注 m: 单元号 (m = 0)、n: 通道号 (n = 0 ~ 7)

表 6-5 各运行模式中的 OVF 位的变化和设定 / 清除条件

定时器运行模式	OVF 位	设定 / 清除条件
- 捕捉模式 - 捕捉 & 单次计数模式	清除	在捕捉时没有发生上溢的情况
	置位	在捕捉时发生上溢的情况
- 间隔定时器模式 - 事件计数器模式 - 单次计数模式	清除	— (不能使用)
	置位	

备注 在计数器发生上溢后, OVF 位不立即发生变化, 而在此后的捕捉时发生变化。

6.3.5 定时器通道允许状态寄存器 m（TEm）

TEm 寄存器是表示各通道定时器运行的允许或者停止状态的寄存器。

TEm 寄存器的各位对应定时器通道开始寄存器 m（TSm）和定时器通道停止寄存器 m（TTm）的各位。如果将 TSm 寄存器的各位置“1”，TEm 寄存器的对应位就被置“1”。如果将 TTm 寄存器的各位置“1”，就将其对应位清“0”。

通过 16 位存储器操作指令读 TEm 寄存器。

能用 TEmL 并且通过 1 位或者 8 位存储器操作指令读 TEm 寄存器的低 8 位。

在产生复位信号后，TEm 寄存器的值变为“0000H”。

图 6-14 定时器通道允许状态寄存器 m（TEm）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
TEm	0	0	0	0	TEH m3	0	TEH m1	0	TEm 7	TEm 6	TEm 5	TEm 4	TEm 3	TEm 2	TEm 1	TEm 0

TEH m3	通道 3 为 8 位定时器模式时的高 8 位定时器的运行允许或者停止状态的表示
0	运行停止状态
1	运行允许状态

TEH m1	通道 1 为 8 位定时器模式时的高 8 位定时器的运行允许或者停止状态的表示
0	运行停止状态
1	运行允许状态

TEmn	通道 n 的运行允许或者停止状态的表示
0	运行停止状态
1	运行允许状态

在通道 1 和通道 3 为 8 位定时器模式时，TEm1 和 TEm3 表示低 8 位定时器的运行允许或者停止状态。

备注 m: 单元号 (m = 0)、n: 通道号 (n = 0 ~ 7)

6.3.6 定时器通道开始寄存器 m (TSm)

TSm 寄存器是对定时器计数寄存器 mn (TCRmn) 进行初始化并且设定各通道计数运行开始的触发寄存器。

如果将各位置“1”，定时器通道允许状态寄存器 m (TEm) 的对应位就被置“1”。因为 TSmn 位、TSHm1 位和 TSHm3 位是触发位，所以如果变为运行允许状态 (TEmn、TEHm1、TEHm3=1)，就立即清除 TSmn 位、TSHm1 位和 TSHm3 位。

通过 16 位存储器操作指令设定 TSm 寄存器。

能用 TSmL 并且通过 1 位或者 8 位存储器操作指令设定 TSm 寄存器的低 8 位。

在产生复位信号后，TSm 寄存器的值变为“0000H”。

图 6-15 定时器通道开始寄存器 m (TSm) 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
TSm	0	0	0	0	TSH m3	0	TSH m1	0	TSm 7	TSm 6	TSm 5	TSm 4	TSm 3	TSm 2	TSm 1	TSm 0

TSH m3	通道 3 为 8 位定时器模式时的高 8 位定时器的运行允许（开始）触发
0	没有触发运行。
1	将 TEHm3 位置“1”，进入计数允许状态。 如果在计数允许状态下开始 TCRm3 寄存器的计数，就进入间隔定时器模式（参照“6.5.2 计数器的开始时序”的表 6-6）。

TSH m1	通道 1 为 8 位定时器模式时的高 8 位定时器的运行允许（开始）触发
0	没有触发运行。
1	将 TEHm1 位置“1”，进入计数允许状态。 如果在计数允许状态下开始 TCRm1 寄存器的计数，就进入间隔定时器模式（参照“6.5.2 计数器的开始时序”的表 6-6）。

TSmn	通道 n 的运行允许（开始）触发
0	没有触发运行。
1	将 TEmn 位置“1”，进入计数允许状态。 计数允许状态下的 TCRmn 寄存器的计数开始因运行模式而不同（参照“6.5.2 计数器的开始时序”的表 6-6）。 在通道 1 和通道 3 为 8 位定时器模式时，TSm1 和 TSm3 为低 8 位定时器的运行允许（开始）触发。

- 注意 1. 必须将 bit15 ~ 12、10、8 置“0”。
2. 在从不使用 TImn 引脚输入的功能切换到使用 TImn 引脚输入的功能时，从设定定时器模式寄存器 mn (TMRmn) 到将 TSmn (TSHm1、TSHm3) 位置“1”为止，需要以下期间的等待：
TImn 引脚噪声滤波器有效时 (TNFENmn=1)：4 个运行时钟 (f_{MCK})
TImn 引脚噪声滤波器无效时 (TNFENmn=0)：2 个运行时钟 (f_{MCK})

- 备注 1. TSm 寄存器的读取值总是“0”。
2. m: 单元号 (m=0)、n: 通道号 (n=0 ~ 7)

6.3.7 定时器通道停止寄存器 m（TTm）

TTm 寄存器是设定各通道计数停止的触发寄存器。

如果将各位置“1”，定时器通道允许状态寄存器 m（TEm）的对应位就被清“0”。因为 TTmn 位、TTHm1 位和 TTHm3 位是触发位，所以如果变为运行停止状态（TEmn、TEHm1、TEHm3=0），就立即清除 TTmn 位、TTHm1 位和 TTHm3 位。

通过 16 位存储器操作指令设定 TTm 寄存器。

能用 TTmL 并且通过 1 位或者 8 位存储器操作指令设定 TTm 寄存器的低 8 位。

在产生复位信号后，TTm 寄存器的值变为“0000H”。

图 6-16 定时器通道停止寄存器 m（TTm）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
TTm	0	0	0	0	TTH m3	0	TTH m1	0	TTm 7	TTm 6	TTm 5	TTm 4	TTm 3	TTm 2	TTm 1	TTm 0
TTH m3	通道 3 为 8 位定时器模式时的高 8 位定时器的运行停止触发															
0	没有触发运行。															
1	将 TEHm3 位清“0”，进入计数运行停止状态。															
TTH m1	通道 1 为 8 位定时器模式时的高 8 位定时器的运行停止触发															
0	没有触发运行。															
1	将 TEHm1 位清“0”，进入计数运行停止状态。															
TTmn	通道 n 的运行停止触发															
0	没有触发运行。															
1	将 TEmn 位清“0”，进入计数运行停止状态。 在通道 1 和通道 3 为 8 位定时器模式时，TTm1 和 TTm3 为低 8 位定时器的运行停止触发。															

注意 必须将 bit15 ~ 12、10、8 置“0”。

备注 1. TTm 寄存器的读取值总是“0”。

2. m: 单元号 (m=0)、n: 通道号 (n=0 ~ 7)

6.3.8 定时器输入选择寄存器 0（TIS0）

TIS0 寄存器是选择通道 1 的定时器输入的寄存器。
通过 8 位存储器操作指令设定 TIS0 寄存器。
在产生复位信号后，TIS0 寄存器的值变为“00H”。

图 6-17 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 1 使用的定时器输入
0	0	0	定时器输入引脚 （TI01）的输入信号
0	0	1	
0	1	0	
0	1	1	
1	0	0	低速内部振荡器时钟 （f _{IL} ）
1	0	1	副系统时钟 （f _{SUB} ）
其他			禁止设定

注意 选择的定时器输入的高 / 低电平宽度必须大于等于 1/f_{MCK}+10ns。
因此，在选择 f_{SUB} 作为 f_{CLK} 时（CKS 寄存器的 CSS=1），不能将 TIS02 位置“1”。

6.3.9 定时器输出允许寄存器 m（TOEm）

TOEm 寄存器是设定允许或者禁止各通道定时器输出的寄存器。

对于允许定时器输出的通道 n，无法通过软件改写后述的定时器输出寄存器 m（TOM）的 TOMn 位的值，并且由计数运行的定时器输出功能反映的值从定时器的输出引脚（TOMn）输出。

通过 16 位存储器操作指令设定 TOEm 寄存器。

能用 TOEmL 并且通过 1 位或者 8 位存储器操作指令设定 TOEm 寄存器的低 8 位。

在产生复位信号后，TOEm 寄存器的值变为“0000H”。

图 6-18 定时器输出允许寄存器 m（TOEm）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
TOEm	0	0	0	0	0	0	0	0	TOE m7	TOE m6	TOE m5	TOE m4	TOE m3	TOE m2	TOE m1	TOE m0

TOE mn	允许 / 禁止通道 n 的定时器输出
0	禁止定时器的输出。 定时器的运行不反映到 TOMn 位，固定输出。 能写 TOMn 位，并且从 TOMn 引脚输出给 TOMn 位设定的电平。
1	允许定时器的输出。 定时器的运行反映到 TOMn 位，产生输出波形。 忽视 TOMn 位的写操作。

注意 必须将 bit15 ~ 8 置“0”。

备注 m: 单元号（m=0）、n: 通道号（n=0 ~ 7）

6.3.10 定时器输出寄存器 m（TOM）

TOM 寄存器是各通道定时器输出的缓冲寄存器。
此寄存器各位的值从各通道的定时器的输出引脚（TOMn）输出。

只有在禁止定时器输出（TOEmn=0）时才能通过软件改写此寄存器的 TOMn 位。当允许定时器输出时（TOEmn=1），忽视通过软件的改写操作，而只通过定时器的运行更改其值。

要将 P52/TI00/TO00、P32/TI01/TO01、P54/TI02/TO02、P30/TI03/TO03、P14/TI04/TO04、P42/TI05/TO05、P56/TI06/TO06 或者 P15/TI07/TO07 引脚用作端口功能时，必须将相应的 TOMn 位置“0”。

通过 16 位存储器操作指令设定 TOM 寄存器。
能用 TOML 并且通过 8 位存储器操作指令设定 TOM 寄存器的低 8 位。
在产生复位信号后，TOM 寄存器的值变为“0000H”。

图 6-19 定时器输出寄存器 m（TOM）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
TOM	0	0	0	0	0	0	0	0	TOM 7	TOM 6	TOM 5	TOM 4	TOM 3	TOM 2	TOM 1	TOM 0

TOMn	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 必须将 bit15～8 置“0”。

备注 m: 单元号（m=0）、n: 通道号（n=0～7）

6.3.11 定时器输出电平寄存器 m（TOLm）

TOLm 寄存器是控制各通道定时器输出电平的寄存器。

当允许定时器输出（TOEmn=1）并且为多通道联动运行功能（TOMmn=1）时，在定时器输出信号的置位和复位时序，反映此寄存器进行的各通道 n 的反相设定。在主导通道输出模式（TOMmn=0）中，此寄存器的设定无效。

通过 16 位存储器操作指令设定 TOLm 寄存器。

能用 TOLmL 并且通过 8 位存储器操作指令设定 TOLm 寄存器的低 8 位。

在产生复位信号后，TOLm 寄存器的值变为“0000H”。

图 6-20 定时器输出电平寄存器 m（TOLm）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
TOLm	0	0	0	0	0	0	0	0	TOL m7	TOL m6	TOL m5	TOL m4	TOL m3	TOL m2	TOL m1	0

TOL mn	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	负逻辑输出 （低电平有效）														

注意 必须将 bit15 ~ 8 和 bit0 置“0”。

备注 1. 如果在定时器运行中改写此寄存器的值，就在下一次定时器输出信号发生变化的时序反相定时器的输出逻辑，而不是在改写后立即反相。

2. m: 单元号（m=0）、n: 通道号（n=0 ~ 7）

6.3.12 定时器输出模式寄存器 m（TOMm）

TOMm 寄存器是控制各通道定时器输出模式的寄存器。

当用作独立通道运行功能时，将所用通道的对应位置“0”。

当用作多通道联动运行功能（PWM 输出、单触发脉冲输出和多重 PWM 输出）时，将主控通道的对应位置“0”并且将从属通道的对应位置“1”。

当允许定时器输出（TOEmn=1）时，在定时器输出信号的置位和复位时序，反映此寄存器进行的各通道 n 的设定。

通过 16 位存储器操作指令设定 TOMm 寄存器。

能用 TOMmL 并且通过 8 位存储器操作指令设定 TOMm 寄存器的低 8 位。

在产生复位信号后，TOMm 寄存器的值变为“0000H”。

图 6-21 定时器输出模式寄存器 m（TOMm）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
TOMm	0	0	0	0	0	0	0	0	TOM m7	TOM m6	TOM m5	TOM m4	TOM m3	TOM m2	TOM m1	0

TOM mn	通道 n 的定时器输出模式的控制
0	主控通道输出模式（通过定时器中断请求信号（INTTMmn）进行交替输出）
1	从属通道输出模式（通过主控通道的定时器中断请求信号（INTTMmn）将输出置位，并且通过从属通道的定时器中断请求信号（INTTMmp）对输出进行复位）

注意 必须将 bit15～8 和 bit0 置“0”。

备注 m: 单元号（m=0）
 n: 通道号
 n=0～7（主控通道时：n=0、2、4、6）
 p: 从属通道号
 n<p≤7
 （有关主控通道和从属通道关系的详细内容，请参照“6.4.1 多通道联动运行功能的基本规则”。）

6.3.13 输入切换控制寄存器（ISC）

ISC 寄存器的 ISC1 位和 ISC0 位用于将通道 7 和串行阵列单元联动来实现 LIN-bus 通信运行。如果将 ISC1 位置“1”，就选择串行数据输入引脚（RxD0）的输入信号作为定时器的输入信号。

通过 1 位或者 8 位存储器操作指令设定 ISC 寄存器。

在产生复位信号后，ISC 寄存器的值变为“00H”。

图 6-22 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	将 TI07 引脚的输入信号作为定时器输入（通常运行）。
1	将 RxD0 引脚的输入信号作为定时器的输入（唤醒信号检测、间隔场低电平宽度测量和同步场脉宽测量）。

ISC0	切换外部中断（INTP0）的输入
0	将 INTP0 引脚的输入信号作为外部中断的输入（通常运行）。
1	将 RxD0 引脚的输入信号作为外部中断的输入（唤醒信号检测）。

注意 必须将 bit7 ～ 2 置“0”。

备注 要使用 LIN-bus 通信时，必须将 ISC1 位置“1”，选择 RxD0 引脚的输入信号。

6.3.14 噪声滤波器允许寄存器 1（NFEN1）

NFEN1 寄存器设定噪声滤波器是否用于各通道定时器输入引脚的输入信号。

对于需要消除噪声的引脚，必须将对应的位置“1”，使噪声滤波器有效。

当噪声滤波器有效时，通过对象通道的运行时钟（ f_{MCK} ）进行同步后，执行 2 个时钟的一致检测。当噪声滤波器无效时，只通过对象通道的运行时钟（ f_{MCK} ）进行同步注。

通过 1 位或者 8 位存储器操作指令设定 NFEN1 寄存器。

在产生复位信号后，NFEN1 寄存器的值变为“00H”。

注 详细内容请参照“6.5.1 (2) 选择 TImn 引脚输入信号的有效边沿的情况（CCSmn=1）”和“6.5.2 计数器的开始时序”。

图 6-23 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	TNFEN04	TNFEN03	TNFEN02	TNFEN01	TNFEN00
	TNFEN07	TI07 引脚或者 RxD0 引脚输入信号的噪声滤波器的使用与否注						
	0	噪声滤波器 OFF						
	1	噪声滤波器 ON						
	TNFEN06	TI06 引脚输入信号的噪声滤波器的使用与否						
	0	噪声滤波器 OFF						
	1	噪声滤波器 ON						
	TNFEN05	TI05 引脚输入信号的噪声滤波器的使用与否						
	0	噪声滤波器 OFF						
	1	噪声滤波器 ON						
	TNFEN04	TI04 引脚输入信号的噪声滤波器的使用与否						
	0	噪声滤波器 OFF						
	1	噪声滤波器 ON						
	TNFEN03	TI03 引脚输入信号的噪声滤波器的使用与否						
	0	噪声滤波器 OFF						
	1	噪声滤波器 ON						
	TNFEN02	TI02 引脚输入信号的噪声滤波器的使用与否						
	0	噪声滤波器 OFF						
	1	噪声滤波器 ON						
	TNFEN01	TI01 引脚输入信号的噪声滤波器的使用与否						
	0	噪声滤波器 OFF						
	1	噪声滤波器 ON						
	TNFEN00	TI00 引脚输入信号的噪声滤波器的使用与否						
	0	噪声滤波器 OFF						
	1	噪声滤波器 ON						

注 能通过设定输入切换控制寄存器（ISC）的 ISC1 位来切换适用的引脚。

ISC1=0：可选择是否使用 TI07 引脚的噪声滤波器。

ISC1=1：可选择是否使用 RxD0 引脚的噪声滤波器。

6.3.15 定时器输入 / 输出引脚的端口功能控制寄存器

在使用定时器阵列单元时，必须设定对象通道的复用端口功能的控制寄存器（端口模式寄存器（PMxx）、端口寄存器（Pxx）、端口模式控制寄存器（PMCxx））。详细内容请参照“4.3.1 端口模式寄存器（PMxx）”、“4.3.2 端口寄存器（Pxx）”和“4.3.6 端口模式控制寄存器（PMCxx）”。

需要设定的端口模式寄存器（PMxx）、端口寄存器（Pxx）和端口模式控制寄存器（PMCxx）因产品而不同。详细内容请参照“4.5.3 使用的端口功能和复用功能的寄存器设定例子”。

在将定时器输出引脚的复用端口（P14/TI04/TO04、P42/TI05/TO05 等）用作定时器的输出时，必须将各端口对应的端口模式控制寄存器（PMCxx）的位、端口模式寄存器（PMxx）的位和端口寄存器（Pxx）的位置“0”。

例）将 P14/TI04/TO04 用作定时器输出的情况

将端口模式寄存器 1 的 PM14 位置“0”。

将端口寄存器 1 的 P14 位置“0”。

将 LCD 端口功能寄存器（PFSEGx）的位置“0”。

在将定时器输入引脚的复用端口（P14/TI04/TO04、P42/TI05/TO05 等）用作定时器的输入时，必须将各端口对应的端口模式寄存器（PMxx）的位置“1”并且将端口模式控制寄存器（PMCxx）的位置“0”。此时，端口寄存器（Pxx）的位可以是“0”或者“1”。

例）将 P14/TI04/TO04 用作定时器输入的情况

将端口模式寄存器 1 的 PM14 位置“1”。

将端口寄存器 1 的 P14 位置“0”或者“1”。

将 LCD 端口功能寄存器（PFSEGx）的位置“0”。

6.4 定时器阵列单元的基本规则

6.4.1 多通道联动运行功能的基本规则

多通道联动运行功能是将主控通道（主要对周期进行计数的基准定时器）和从属通道（遵从主控通道运行的定时器）组合实现的功能，使用时需要遵守几个规则。

多通道联动运行功能的基本规则如下所示。

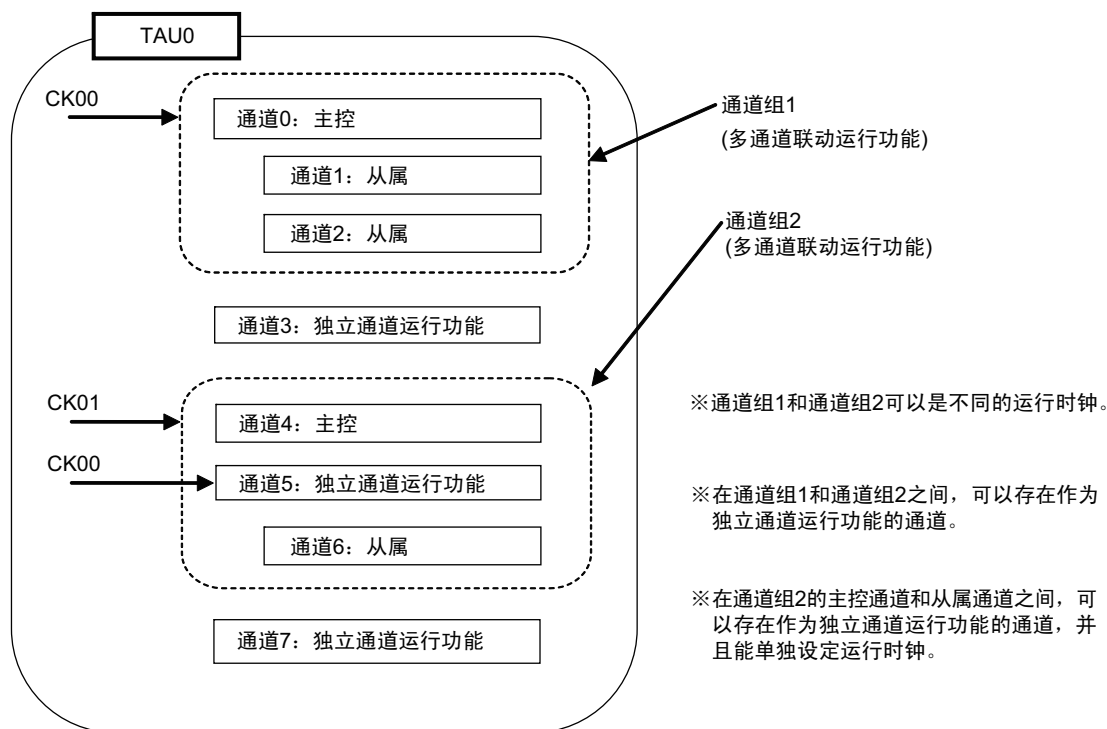
- (1) 只能将偶数通道（通道0、通道2、通道4、.....）设定为主控通道。
- (2) 通道0以外的任何通道都能设定为从属通道。
- (3) 只能将主控通道的低位通道设定为从属通道。
例 在将通道2设定为主控通道时，能将通道3开始的通道（通道3、通道4、通道5、.....）设定为从属通道。
- (4) 能对1个主控通道设定多个从属通道。
- (5) 当使用多个主控通道时，不能设定跨越主控通道的从属通道。
例 在将通道0和通道4设定为主控通道时，能将通道1～3设定为主控通道0的从属通道，而不能将通道5～7设定为主控通道0的从属通道。
- (6) 和主控通道联动的从属通道需要设定与主控通道相同的运行时钟。和主控通道联动的从属通道的CKSmn0位和CKSmn1位（定时器模式寄存器mn（TMRmn）的bit15和bit14）的值需要是与主控通道相同的设定值。
- (7) 主控通道能将INTTMmn（中断）、开始软件触发和计数时钟传给低位通道。
- (8) 从属通道能将主控通道的INTTMmn（中断）、开始软件触发和计数时钟用作源时钟，但是不能将自己的INTTMmn（中断）、开始软件触发和计数时钟传给低位通道。
- (9) 主控通道不能将其他高位主控通道的INTTMmn（中断）、开始软件触发和计数时钟用作源时钟。
- (10) 为了同时启动要联动的通道，需要同时设定联动通道的通道开始触发位（TSmn）。
- (11) 只有联动的全部通道或者主控通道才能使用计数运行中的TSmn位的设定。不能只使用从属通道的TSmn位的设定。
- (12) 为了同时停止要联动的通道，需要同时设定联动通道的通道停止触发位（TTmn）。
- (13) 在联动运行时，因为主控通道和从属通道需要相同的运行时钟，所以不能选择CKm2/CKm3。
- (14) 定时器模式寄存器m0（TMRm0）没有主控位而固定为“0”。但是，因为通道0是最高位的通道，所以在联动运行时能将通道0用作主控通道。

多通道联动运行功能的基本规则是适用于通道组（形成1个多通道联动运行功能的主控通道和从属通道的集合）的规则。

如果设定2个或者更多的相互不联动的通道组，通道组之间就不适用上述的基本规则。

备注 m: 单元号（m=0）、n: 通道号（n=0～7）

例



6.4.2 8 位定时器运行功能的基本规则（只限通道 1 和通道 3）

8 位定时器运行功能是将 16 位定时器的通道用作 2 个 8 位定时器的通道的功能。

只有通道 1 和通道 3 才能使用 8 位定时器运行功能，使用时需要遵守几个规则。

8 位定时器运行功能的基本规则如下所示。

- (1) 8 位定时器运行功能只适用于通道 1 和通道 3。
- (2) 当用作 8 位定时器时，将定时器模式寄存器 mn（TMRmn）的 SPLIT 位置“1”。
- (3) 高 8 位定时器能用作间隔定时器功能。
- (4) 在开始运行时，高 8 位定时器输出 INTTm1H/INTTm3H（中断）（和 MDmn0 位为“1”的运行相同）。
- (5) 高 8 位定时器的运行时钟的选择取决于低位 TMRmn 寄存器的 CKSmn1 位和 CKSmn0 位的设定。
- (6) 对于高 8 位定时器，通过操作 TSHm1/TSHm3 位来开始通道的运行，并且通过操作 TTHm1/TTHm3 位来停止通道的运行。能通过 TEHm1/TEHm3 位确认通道的状态。
- (7) 低 8 位定时器的运行取决于 TMRmn 寄存器的设定，有以下 3 种支持低 8 位定时器运行的功能：
 - 间隔定时器功能
 - 外部事件计数器功能
 - 延迟计数功能
- (8) 对于低 8 位定时器，通过操作 TSm1/TSm3 位来开始通道的运行，并且通过操作 TTm1/TTm3 位来停止通道的运行。能通过 TEm1/TEm3 位确认通道的状态。
- (9) 在 16 位定时器运行时，TSHm1/TSHm3/TTHm1/TTHm3 位的操作无效。通过操作 TSm1/TSm3 位和 TTm1/TTm3 位来控制通道 1 和通道 3 的运行。TEHm3 位和 TEHm1 位不变。
- (10) 8 位定时器功能不能使用联动运行功能（单触发脉冲、PWM 和多重 PWM）。

备注 m：单元号（m=0）、n：通道号（n=1、3）

6.5 计数器的操作

6.5.1 计数时钟 (f_{TCLK})

定时器阵列单元的计数时钟 (f_{TCLK}) 能通过定时器模式寄存器 mn (TMRmn) 的 CCSmn 位选择以下任意一个时钟:

- CKSmn0 位和 CKSmn1 位指定的运行时钟 (f_{MCK})
- TImn 引脚的输入信号的有效边沿

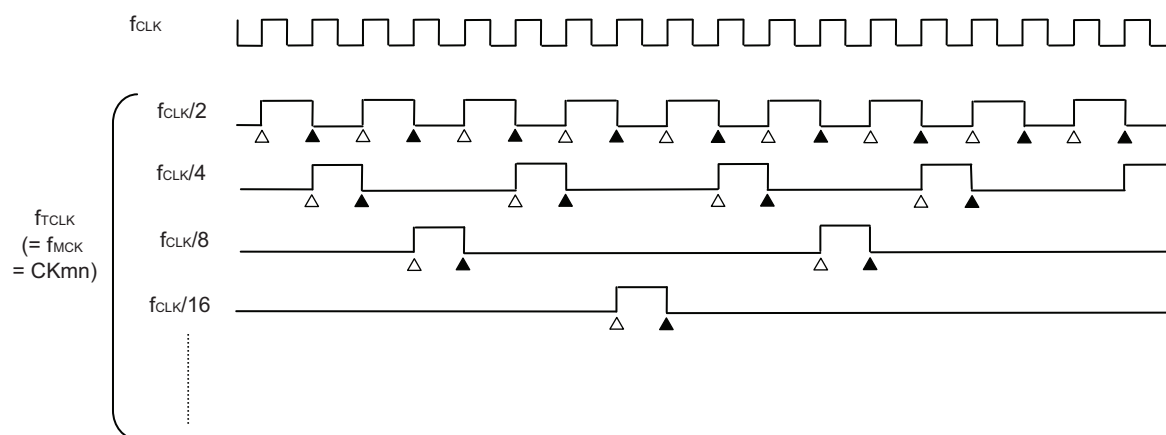
定时器阵列单元被设计为与 f_{CLK} 同步运行, 因此计数时钟 (f_{TCLK}) 的时序如下。

(1) 选择 CKSmn0 位和 CKSmn1 位指定的运行时钟 (f_{MCK}) 的情况 (CCSmn=0)

根据定时器时钟选择寄存器 m (TPSm) 的设定, 计数时钟 (f_{TCLK}) 为 $f_{CLK} \sim f_{CLK}/2^{15}$ 。但是, 当选择 f_{CLK} 的分频时, TPSm 寄存器选择的时钟是从上升沿开始只有 1 个 f_{CLK} 周期为高电平的信号。当选择 f_{CLK} 时, 固定为高电平。

为了取得与 f_{CLK} 的同步, 定时器计数寄存器 mn (TCRmn) 从计数时钟的上升沿开始延迟 1 个 f_{CLK} 时钟后进行计数, 出于方便, 将其称为“在计数时钟的上升沿进行计数”。

图 6-24 f_{CLK} 和计数时钟 (f_{TCLK}) 的时序 (CCSmn = 0 时)



备注 1. $\triangle$: 计数时钟的上升沿

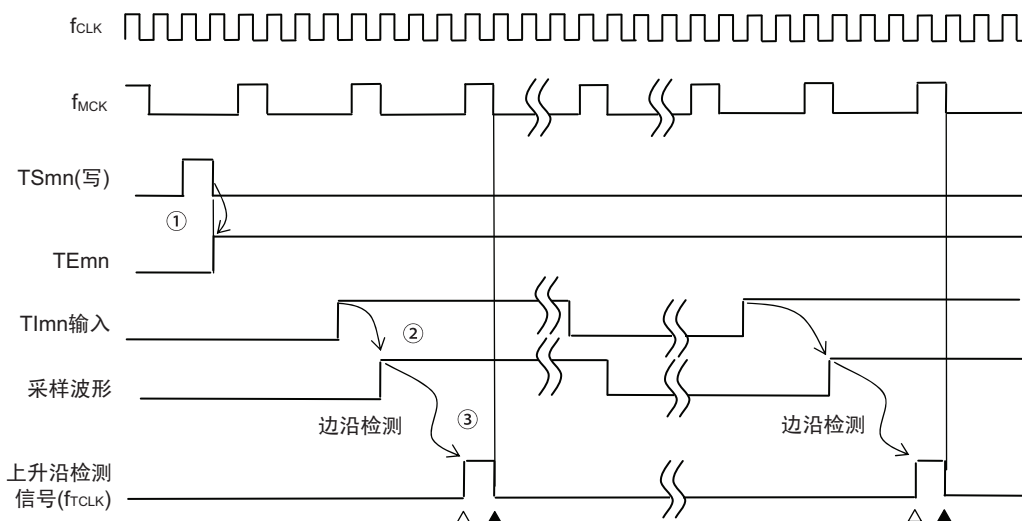
$\blacktriangle$: 同步、计数器的递增 / 递减

2. f_{CLK} : CPU/ 外围硬件时钟

(2) 选择 TImn 引脚输入信号的有效边沿的情况 (CCSmn=1)

计数时钟 (f_{TCLK}) 是检测 TImn 引脚输入信号的有效边沿并且与下一个 f_{MCK} 上升沿同步的信号。实际上, 这是比 TImn 引脚的输入信号延迟了 1~2 个 f_{MCK} 时钟的信号 (在使用噪声滤波器时, 延迟 3~4 个 f_{MCK} 时钟)。

为了取得与 f_{CLK} 的同步, 定时器计数寄存器 mn (TCRmn) 从计数时钟的上升沿开始延迟 1 个 f_{CLK} 时钟后进行计数, 出于方便, 将其称为“在 TImn 引脚输入信号的有效边沿进行计数”。

图 6-25 计数时钟 (f_{TCLK}) 的时序 (CCSmn = 1、未使用噪声滤波器时)

- ① 通过将 TSmn 位置位来开始定时器运行, 并且等待 TImn 输入的有效边沿。
- ② 通过 f_{MCK} 对 TImn 输入的上升沿进行采样。
- ③ 在采样信号的上升沿检测边沿, 并且输出检测信号 (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 的运行时钟

3. 输入脉冲间隔的测量、输入信号高 / 低电平测量、延迟计数器和单触发脉冲输出功能的 TImn 输入也是同样的波形。

6.5.2 计数器的开始时序

通过将定时器通道开始寄存器 m (TSMn) 的 TSMn 位置位, 定时器计数寄存器 mn (TCRmn) 进入运行允许状态。

从计数运行允许状态到定时器计数寄存器 mn (TCRmn) 开始计数为止的运行如表 6-6 所示。

表 6-6 从计数运行允许状态到定时器计数寄存器 mn (TCRmn) 开始计数为止的运行

定时器的运行模式	将 TSMn 位置“1”后的运行
• 间隔定时器模式	从检测到开始触发 (TSMn=1) 到产生计数时钟为止, 不执行任何操作。 通过第 1 个计数时钟将 TDRmn 寄存器的值加载到 TCRmn 寄存器, 并且通过后续的计数时钟进行递减计数 (参照“6.5.3 (1) 间隔定时器模式的运行”)。
• 事件计数器模式	通过给 TSMn 位写“1”, 将 TDRmn 寄存器的值加载到 TCRmn 寄存器。 如果检测到 TIMn 的输入边沿, 就通过后续的计数时钟进行递减计数 (参照“6.5.3 (2) 事件计数器模式的运行”)。
• 捕捉模式	从检测到开始触发 (TSMn=1) 到产生计数时钟为止, 不执行任何操作。 通过第 1 个计数时钟将“0000H”加载到 TCRmn 寄存器, 并且通过后续的计数时钟进行递增计数 (参照“6.5.3 (3) 捕捉模式的运行 (输入脉冲的间隔测量)”)。
• 单次计数模式	通过在定时器停止运行 (TEMn=0) 的状态下给 TSMn 位写“1”, 进入开始触发的等待状态。 从检测到开始触发到产生计数时钟为止, 不执行任何操作。 通过第 1 个计数时钟将 TDRmn 寄存器的值加载到 TCRmn 寄存器, 并且通过后续的计数时钟进行递减计数 (参照“6.5.3 (4) 单次计数模式的运行”)。
• 捕捉 & 单次计数模式	通过在定时器停止运行 (TEMn=0) 的状态下给 TSMn 位写“1”, 进入开始触发的等待状态。 从检测到开始触发到产生计数时钟为止, 不执行任何操作。 通过第 1 个计数时钟将“0000H”加载到 TCRmn 寄存器, 并且通过后续的计数时钟进行递增计数 (参照“6.5.3 (5) 捕捉 & 单次计数模式中的运行 (测量高电平宽度)”)。

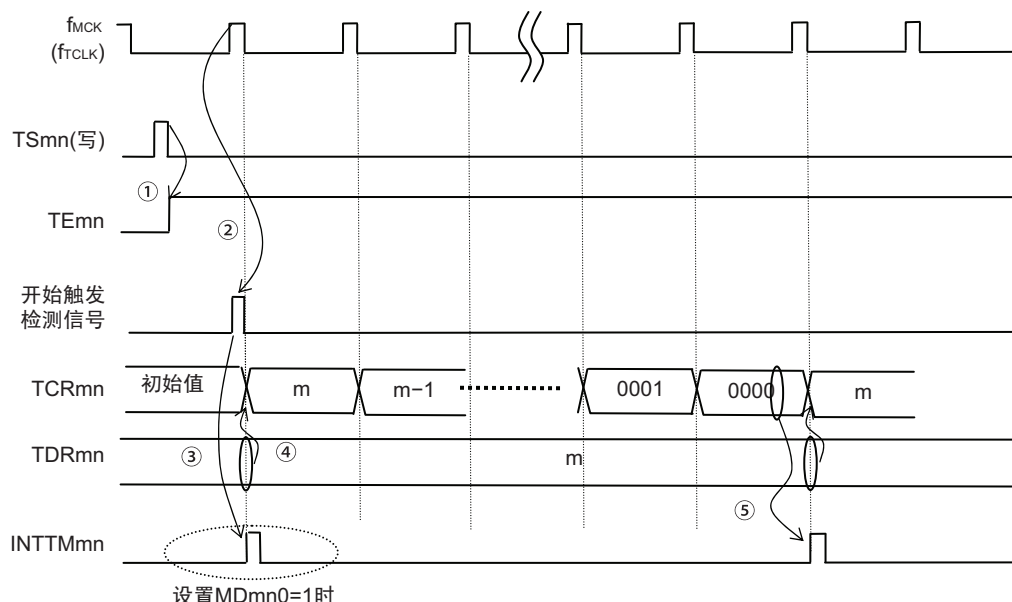
6.5.3 计数器的运行

以下说明各模式的计数器运行。

(1) 间隔定时器模式的运行

- ① 通过给 TSmn 位写“1”，进入运行允许状态 (TEmn=1)。定时器计数寄存器 mn (TCRmn) 保持初始值，直到产生计数时钟为止。
- ② 通过允许运行后的第1个计数时钟 (f_{MCK}) 产生开始触发信号。
- ③ 当 MDmn0 位为“1”时，通过开始触发信号产生 INTTMmn。
- ④ 通过允许运行后的第1个计数时钟将定时器数据寄存器 mn (TDRmn) 的值加载到 TCRmn 寄存器，并且以间隔定时器模式开始计数。
- ⑤ 如果 TCRmn 寄存器递减计数到“0000H”，就通过下一个计数时钟 (f_{MCK}) 产生 INTTMmn，并且在将定时器数据寄存器 mn (TDRmn) 的值加载到 TCRmn 寄存器后继续进行计数。

图 6-26 运行时序 (间隔定时器模式)



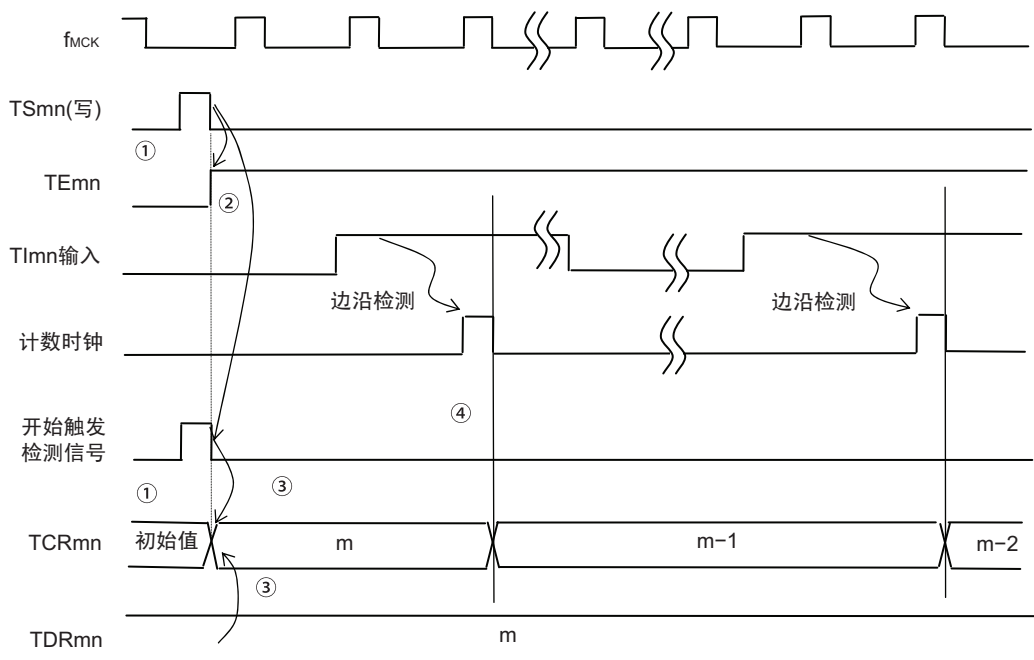
注意 因为第1个计数时钟周期的运行在写 TSmn 位后并且在产生计数时钟前会延迟开始计数，所以产生最大为1个时钟周期的误差。另外，如果需要开始计数时序的信息，就将 MDmn0 位置“1”，以便能在开始计数时产生中断。

备注 f_{MCK} 、开始触发检测信号和 INTTMmn 与 f_{CLK} 同步并且在1个时钟内有效。

(2) 事件计数器模式的运行

- ① 在运行停止状态 ($TE_{mn}=0$) 的期间, 定时器计数寄存器 mn (TCR_{mn}) 保持初始值。
- ② 通过给 TS_{mn} 位写“1”, 进入运行允许状态 ($TE_{mn}=1$)。
- ③ 在 TS_{mn} 位和 TE_{mn} 位都变为“1”的同时将定时器数据寄存器 mn (TDR_{mn}) 的值加载到 TCR_{mn} 寄存器, 并且开始计数。
- ④ 此后, 在 TI_{mn} 输入的有效边沿, 通过计数时钟对 TCR_{mn} 寄存器的值进行递减计数。

图 6-27 运行时序 (事件计数器模式)

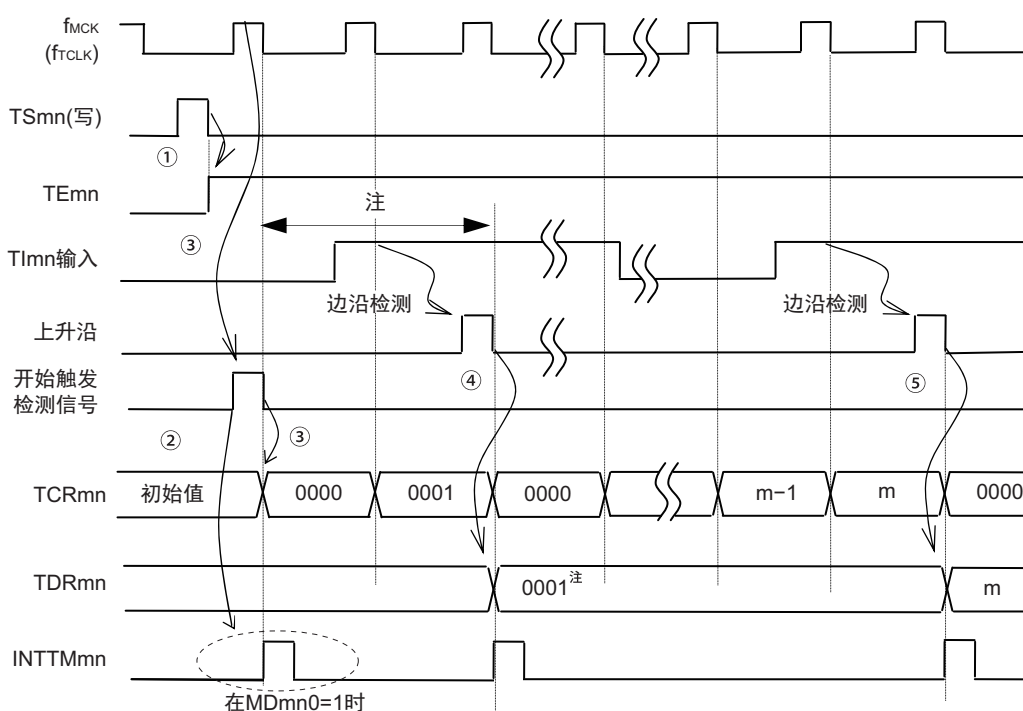


备注 这是不使用噪声滤波器时的时序。如果使用噪声滤波器, 边沿检测就从 TI_{mn} 输入开始再延迟 2 个 f_{MCK} 周期 (合计 3 ~ 4 个周期)。因为 TI_{mn} 输入与计数时钟 (f_{MCK}) 异步, 所以产生 1 个周期的误差。

(3) 捕捉模式的运行（输入脉冲的间隔测量）

- ① 通过给 TSmn 位写“1”，进入运行允许状态（TEmn=1）。
- ② 定时器计数寄存器 mn（TCRmn）保持初始值，直到产生计数时钟为止。
- ③ 通过允许运行后的第 1 个计数时钟（ f_{MCK} ）产生开始触发信号。然后，将“0000H”加载到 TCRmn 寄存器并且以捕捉模式开始计数（当 MDmn0 位为“1”时，通过开始触发产生 INTTMmn）。
- ④ 如果检测到 TImn 输入的有效边沿，就将 TCRmn 寄存器的值捕捉到 TDRmn 寄存器，并且产生 INTTMmn。此时的捕捉值没有意义。TCRmn 寄存器从“0000H”开始继续进行计数。
- ⑤ 如果检测到下一个 TImn 输入的有效边沿，就将 TCRmn 寄存器的值捕捉到 TDRmn 寄存器，并且发生 INTTMmn 中断。

图 6-28 运行时序（捕捉模式：输入脉冲的间隔测量）



注 在开始前将时钟输入到 TImn（有触发）时，即使没有检测边沿也通过触发检测来开始计数，因此第 1 次捕捉时（④）的捕捉值不是脉冲间隔（在此例子中，0001：2 个时钟间隔），必须忽视。

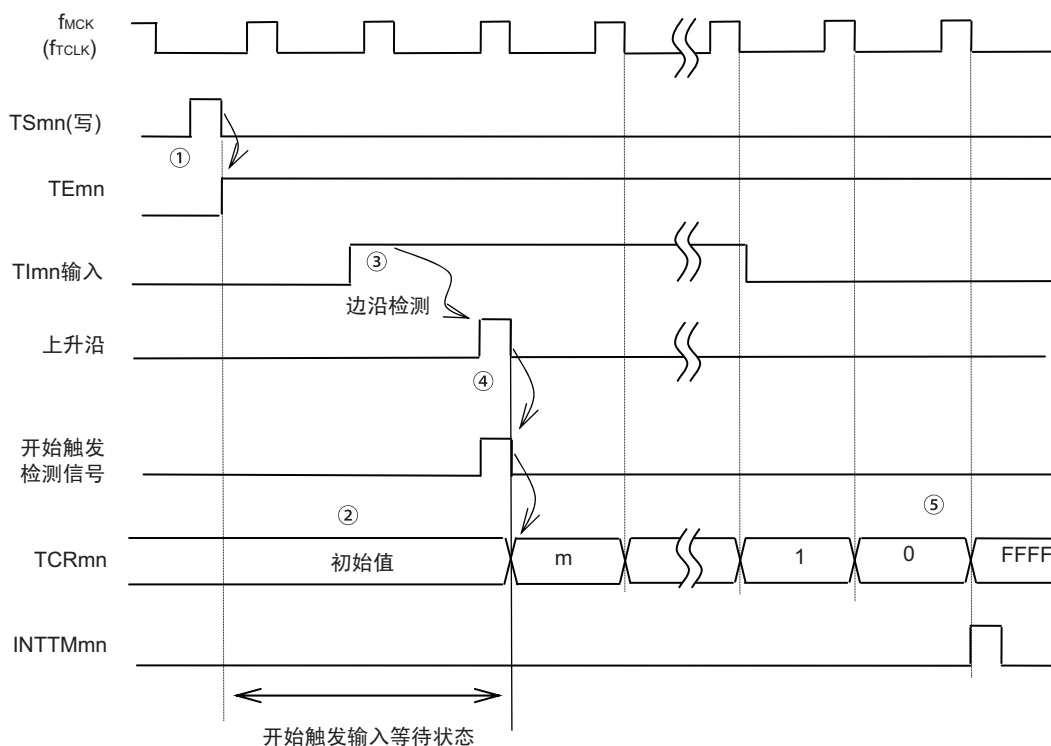
注意 因为第 1 个计数时钟周期的运行在写 TSmn 位后并且在产生计数时钟前会延迟开始计数，所以产生最大为 1 个计数时钟周期的误差。另外，如果需要开始计数时序的信息，就将 MDmn0 位置“1”，以便能在开始计数时产生中断。

备注 这是不使用噪声滤波器时的时序。如果使用噪声滤波器，边沿检测就从 TImn 输入开始再延迟 2 个 f_{MCK} 周期（合计 3 ~ 4 个周期）。因为 TImn 输入与计数时钟（ f_{MCK} ）异步，所以产生 1 个周期的误差。

(4) 单次计数模式的运行

- ① 通过给 TSmn 位写“1”，进入运行允许状态 (TEmn=1)。
- ② 定时器计数寄存器 mn (TCRmn) 保持初始值，直到产生开始触发信号为止。
- ③ 检测 TImn 输入的上升沿。
- ④ 在产生开始触发信号后将 TDRmn 寄存器的值 (m) 加载到 TCRmn 寄存器，并且开始计数。
- ⑤ 当 TCRmn 寄存器递减计数到“0000H”时，发生 INTTMmn 中断，并且 TCRmn 寄存器的值变为“FFFF”，停止计数。

图 6-29 运行时序 (单次计数模式)

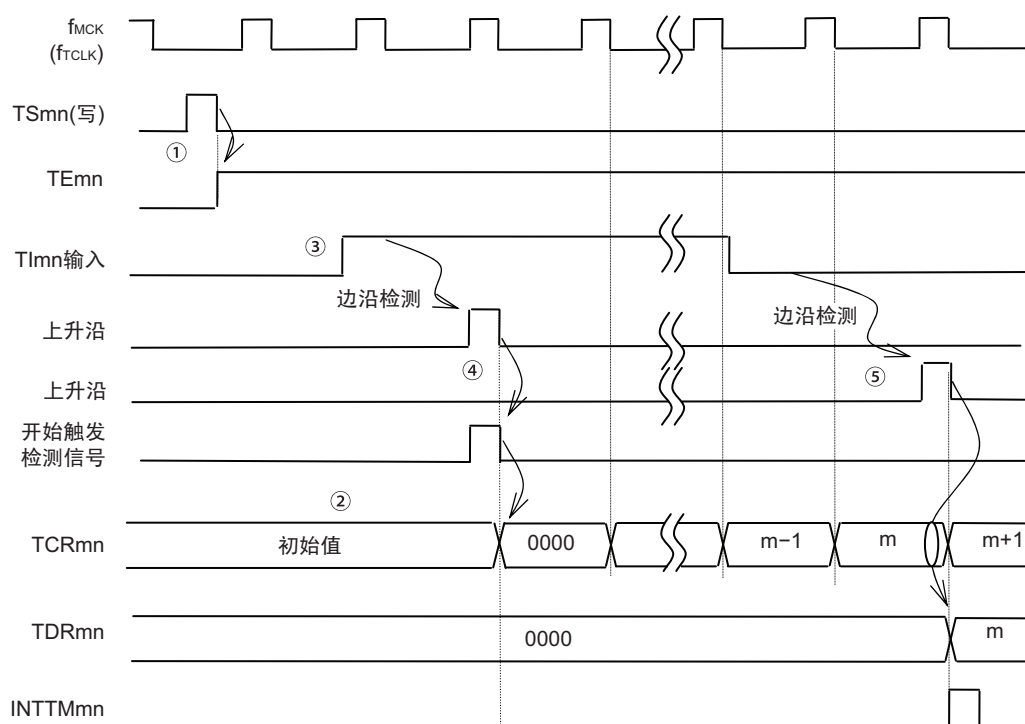


备注 这是不使用噪声滤波器时的时序。如果使用噪声滤波器，边沿检测就从 TImn 输入开始再延迟 2 个 f_{MCK} 周期 (合计 3 ~ 4 个周期)。因为 TImn 输入与计数时钟 (f_{MCK}) 异步，所以产生 1 个周期的误差。

(5) 捕捉 & 单次计数模式中的运行（测量高电平宽度）

- ① 通过给定时器通道开始寄存器 m （ TSm ）的 $TSmn$ 位写“1”，进入运行允许状态（ $TEmn=1$ ）。
- ② 定时器计数寄存器 mn （ $TCRmn$ ）保持初始值，直到产生开始触发信号为止。
- ③ 检测 $TI mn$ 输入的上升沿。
- ④ 在产生开始触发信号后将“0000H”加载到 $TCRmn$ 寄存器，并且开始计数。
- ⑤ 如果检测到 $TI mn$ 输入的下降沿，就将 $TCRmn$ 寄存器的值捕捉到 $TDRmn$ 寄存器，并且发生 $INTTMmn$ 中断。

图 6-30 运行时序（捕捉 & 单次计数模式：测量高电平宽度）

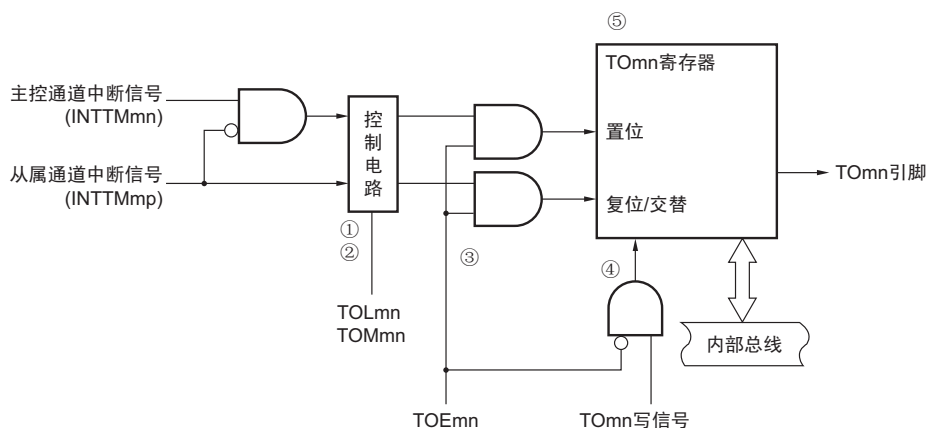


备注 这是不使用噪声滤波器时的时序。如果使用噪声滤波器，边沿检测就从 $TI mn$ 输入开始再延迟 2 个 f_{MCK} 周期（合计 3 ~ 4 个周期）。因为 $TI mn$ 输入与计数时钟（ f_{MCK} ）异步，所以产生 1 个周期的误差。

6.6 通道输出（T_{Om}n 引脚）的控制

6.6.1 T_{Om}n 引脚输出电路的结构

图 6-31 输出电路的结构图



以下说明 T_{Om}n 引脚的输出电路。

- ① 当 TOMmn 位为“0”（主控通道输出模式）时，忽视定时器输出电平寄存器 m（TOLm）的设定值，只将 INTTMmp（从属通道定时器中断）传给定时器输出寄存器 m（TOM）。
- ② 当 TOMmn 位为“1”（从属通道输出模式）时，将 INTTMmn（主控通道定时器中断）和 INTTMmp（从属通道定时器中断）传给 TOM 寄存器。
此时，TOLm 寄存器有效并且如下控制信号：
TOLmn=0 时：正逻辑输出（INTTMmn→置位、INTTMmp→复位）
TOLmn=1 时：负逻辑输出（INTTMmn→复位、INTTMmp→置位）
当同时产生 INTTMmn 和 INTTMmp 时（PWM 输出的 0% 输出），优先 INTTMmp（复位信号）而屏蔽 INTTMmn（置位信号）。
- ③ 在定时器输出允许状态（TOEmn=1）下，将 INTTMmn（主控通道定时器中断）和 INTTMmp（从属通道定时器中断）传给 TOM 寄存器。TOM 寄存器的写操作（TOMn 写信号）无效。
当 TOEmn 位为“1”时，除了中断信号以外，不改变 TOMn 引脚的输出。
要对 TOMn 引脚的输出电平进行初始化时，需要在设定为定时器输出禁止状态（TOEmn=0）后给 TOM 寄存器写值。
- ④ 在定时器输出禁止状态（TOEmn=0）下，对象通道的 TOMn 位的写操作（TOMn 写信号）有效。当定时器输出为禁止状态（TOEmn=0）时，不将 INTTMmn（主控通道定时器中断）和 INTTMmp（从属通道定时器中断）传给 TOM 寄存器。
- ⑤ 能随时读 TOM 寄存器，并且能确认 TOMn 引脚的输出电平。

注意 将 P60、P61 用作通道输出时，为 N 沟道漏极开路输出，因此需要外部上拉电阻。

备注 m：单元号（m = 0）

n：通道号

n = 0 ~ 7（主控通道时 n = 0、2、4、6）

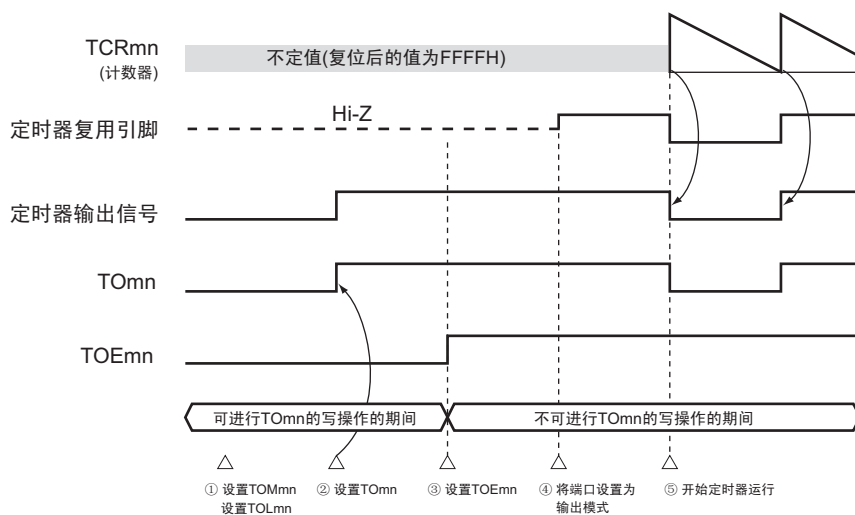
p：从属通道号

n < p ≤ 7

6.6.2 TOmn 引脚的输出设定

从 TOmn 输出引脚的初始设定到定时器开始运行的步骤和状态变化如下所示。

图 6-32 从设定定时器的输出到开始运行的状态变化



① 设定定时器输出的运行模式。

- TOMmn 位 (0: 主控通道输出模式、1: 从属通道输出模式)
- TOLmn 位 (0: 正逻辑输出、1: 负逻辑输出)

② 通过设定定时器输出寄存器m (TOm)，将定时器输出信号设定为初始状态。

③ 给TOEmn 位写“1”，允许定时器输出 (禁止写TOm 寄存器)。

④ 通过端口模式控制寄存器 (PMCxx) 将端口设定为数字输入/输出。

⑤ 将端口的输入/输出设定为输出 (参照“6.3.15 定时器输入/输出引脚的端口功能控制寄存器”)。

⑥ 允许定时器运行 (TSmn=1)。

备注 m: 单元号 (m = 0)、n: 通道号 (n = 0 ~ 7)

6.6.3 通道输出运行的注意事项

(1) 有关定时器运行中的 TOm、TOEm、TOLm 寄存器的设定值变更

定时器运行（定时器计数寄存器 mn（TCRmn）和定时器数据寄存器 mn（TDRmn）的运行）和 T0mn 输出电路相互独立。因此，定时器输出寄存器 m（T0m）、定时器输出允许寄存器 m（T0Em）和定时器输出电平寄存器 m（T0Lm）的设定值的变更不会影响定时器运行，能在定时器运行中更改设定值。但是，为了在各定时器的运行中从 T0mn 引脚输出期待的波形，必须设定为 6.8 和 6.9 所示的各运行的寄存器设定内容例子的值。

如果在产生各通道的定时器中断 (INTTMmn) 信号前后更改除了 TOm 寄存器以外的 TOEm 寄存器和 TOLm 寄存器的设定值, 就根据是在产生定时器中断 (INTTMmn) 信号前更改还是在产生后更改, TOmn 引脚输出的波形可能不同。

备注 m: 单元号 (m = 0)、n: 通道号 (n = 0 ~ 7)

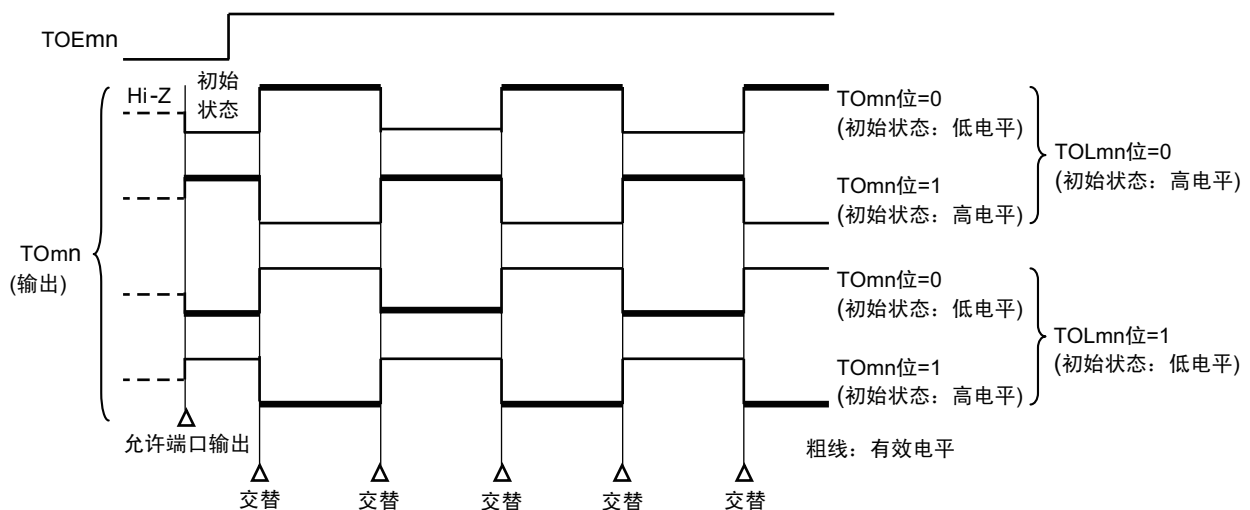
(2) 有关 T0mn 引脚的初始电平和定时器开始运行后的输出电平

在允许端口输出前并且在定时器输出禁止 (TOEmn=0) 的状态下写定时器输出寄存器 m (TOM)，在更改初始电平后设定为定时器输出允许状态 (TOEmn=1) 时的 TOMn 引脚输出电平的变化如下所示。

(a) 在主导通道输出模式 (TOMmn=0) 中开始运行的情况

在主控通道输出模式（TOMmn=0）中，定时器输出电平寄存器 m（TOLm）的设定无效。如果在设定初始电平后开始定时器的运行，就通过产生交替信号反相 TOMn 引脚的输出电平。

图 6-33 交替输出时 (TOMmn = 0) 的 TOMn 引脚输出状态



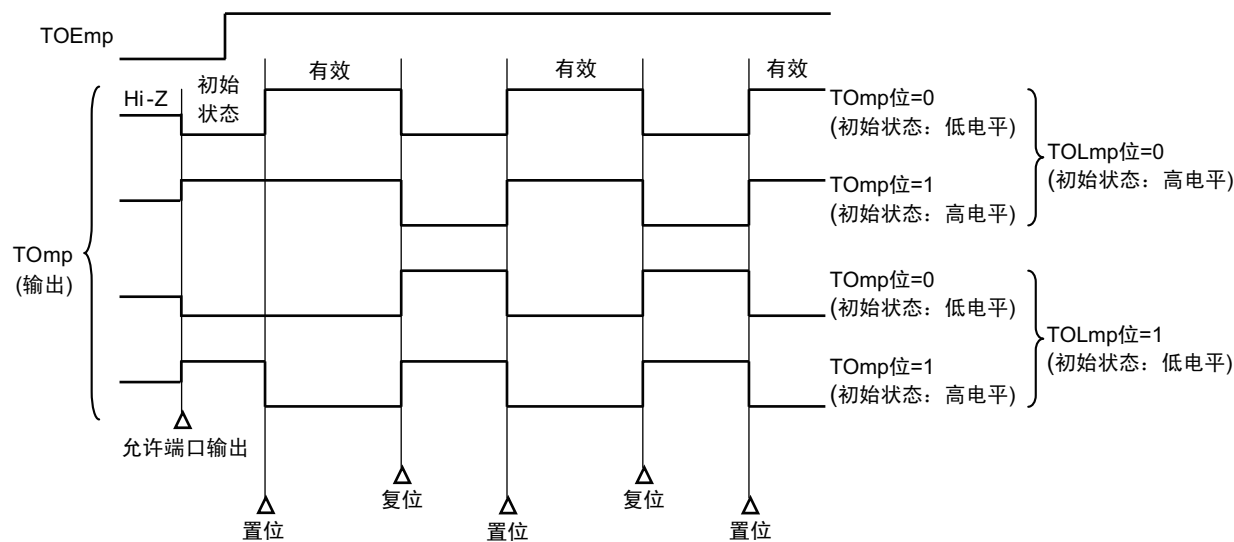
备注 1. 交替：反相 T0mn 引脚的输出状态

2. m: 单元号 (m=0)、n: 通道号 (n=0~7)

(b) 在从属通道输出模式（TOMmp=1）中开始运行的情况（PWM 输出）

在从属通道输出模式（TOMmp=1）中，有效电平取决于定时器输出电平寄存器 m（TOLm）的设定。

图 6-34 PWM 输出时（TOMmp = 1）的 TOmp 引脚输出状态



备注 1. 置位：TOmp 引脚的输出信号从无效电平变为有效电平。

复位：TOmp 引脚的输出信号从有效电平变为无效电平。

2. m：单元号（m=0）、p：通道号（p=1～7）

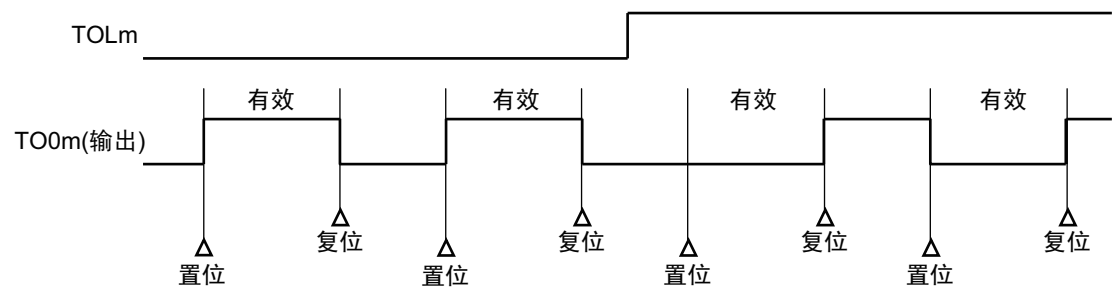
(3) 有关从属通道输出模式（TOMmn=1）的 TOMn 引脚变化

(a) 在定时器运行中更改定时器输出电平寄存器 m（TOLm）的设定的情况

如果在定时器运行中更改 TOLm 寄存器的设定，就在满足 TOMn 引脚变化条件时设定有效。无法通过改写 TOLm 寄存器来改变 TOMn 引脚的输出电平。

当 TOMmn 位为“1”时，在定时器运行中（TEmn=1）更改 TOLm 寄存器的值时的运行如下所示。

图 6-35 在定时器运行中更改 TOLm 寄存器的内容时的运行



备注 1. 置位：TOMn 引脚的输出信号从无效电平变为有效电平。

复位：TOMn 引脚的输出信号从有效电平变为无效电平。

2. m：单元号（m=0） n：通道号（n=0～7）

(b) 置位 / 复位时序

为了在 PWM 输出时实现 0%/100% 输出，通过从属通道将产生主控通道定时器中断（INTTMmn）时的 TOMn 引脚 / TOMn 位的设定时序延迟 1 个计数时钟。

当置位条件和复位条件同时产生时，优先复位条件。

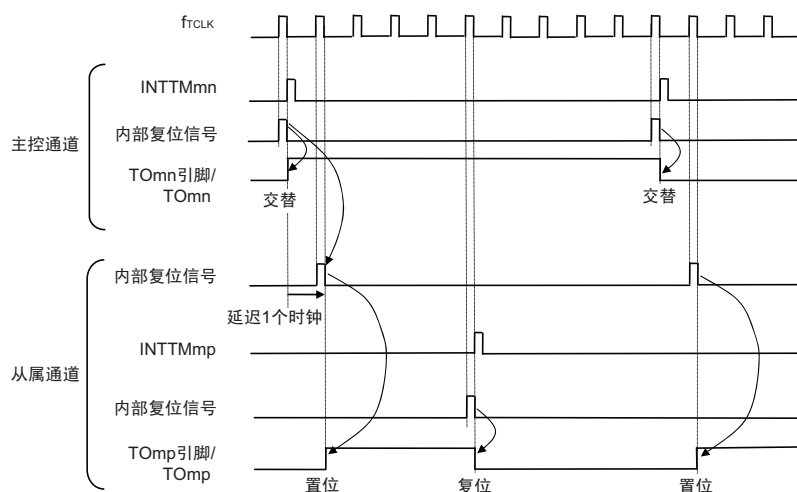
按照以下方法设定主控 / 从属通道时的置位 / 复位运行状态如图 6-36 所示。

主控通道：TOEmn=1、TOMmn=0、TOLmn=0

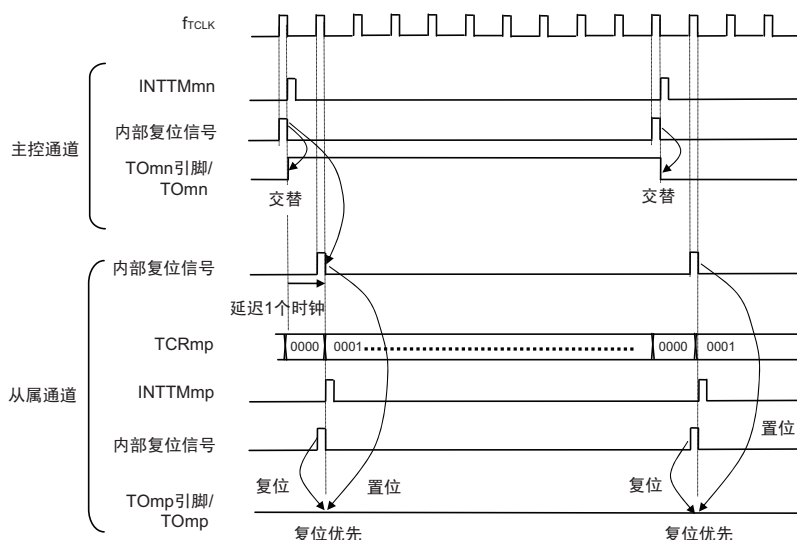
从属通道：TOEmp=1、TOMmp=1、TOLmp=0

图 6-36 置位 / 复位时序运行状态

(1) 基本运行时序



(2) 0 % 占空时的运行时序



备注 1. 内部复位信号：TOMn 引脚复位 / 交替信号

内部置位信号：TOMn 引脚置位信号

2. m: 单元号 (m = 0)

n: 通道号

n = 0 ~ 7 (主控通道时 n = 0、2、4、6)

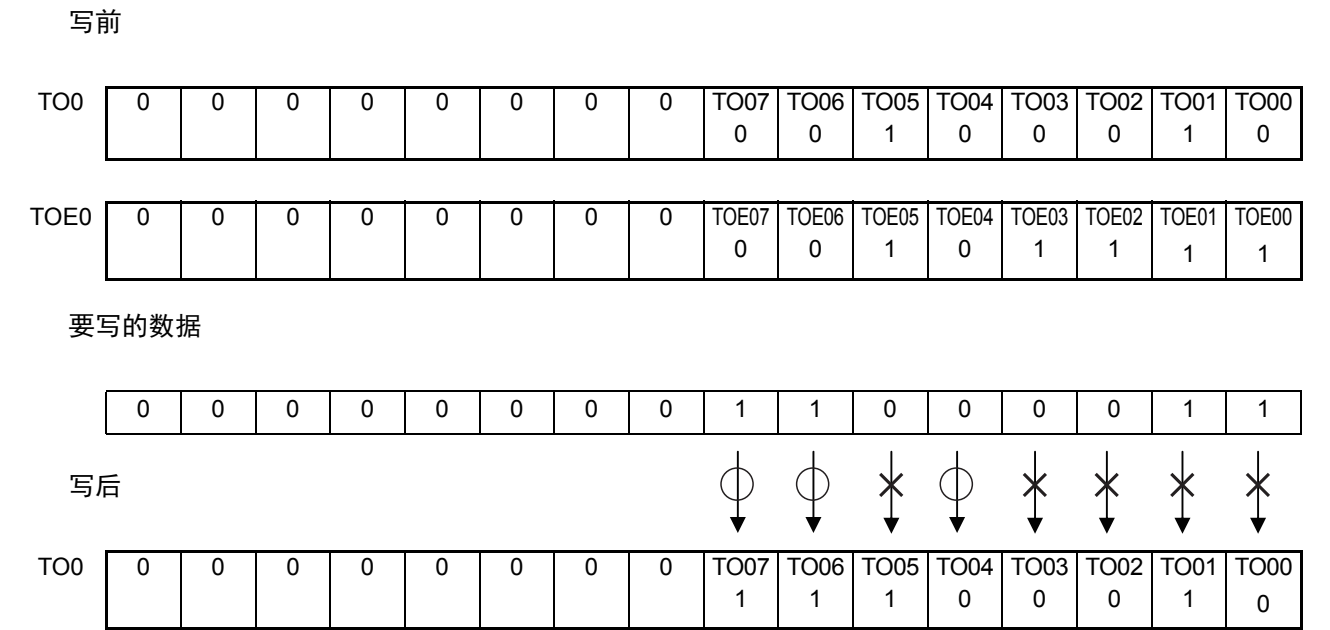
p: 从属通道号

n < p ≤ 7

6.6.4 TOmn 位的一次性操作

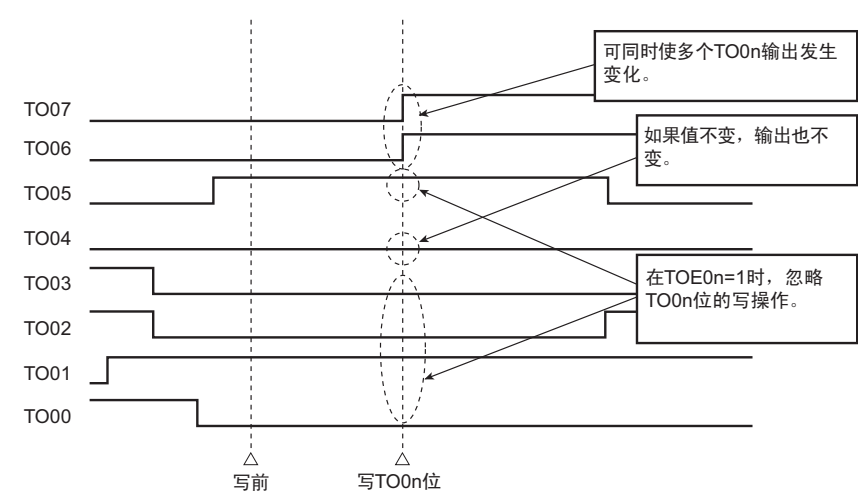
和定时器通道开始寄存器 m (TSm) 相同，定时器输出寄存器 m (TOm) 有全部通道的设定位 (TOmn)，因此能一次性地操作全部通道的 TOmn 位。另外，能只给想作为操作对象的通道输出 (TOmn) 的 TOmn 位写数据 (TOEmn=0)，因此也能只操作任意的位。

图 6-37 TO0n 位的一次性操作例子



只能写 TOEmn 位为“0”的 TOmn 位，忽视 TOEmn 位为“1”的 TOmn 位的写操作。
TOEmn 位为“1”的 TOmn（通道输出）不受写操作的影响，即使写 TOmn 位也被忽视，由定时器运行引起的输出变化正常进行。

图 6-38 一次性操作 TO0n 位时的 TO0n 引脚状态



注意 在定时器输出允许状态 (TOEmn=1) 下，即使各通道的定时器中断 (INTTMmn) 引起的输出和 TOmn 位的写操作发生竞争，TOmn 引脚也能正常输出。

备注 m: 单元号 (m=0)、n: 通道号 (n=0 ~ 7)

6.6.5 有关开始计数时的定时器中断和 TOn 引脚输出

在间隔定时器模式或者捕捉模式中，定时器模式寄存器 mn（TMRmn）的 MDmn0 位是设定是否在开始计数时产生定时器中断的位。

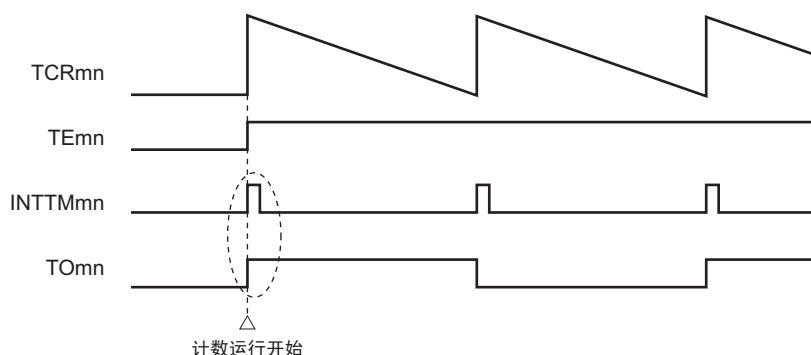
当 MDmn0 位为“1”时，能通过产生定时器中断（INTTMmn）得知计数的开始时序。

在其他模式中，不控制开始计数时的定时器中断和 TOn 输出。

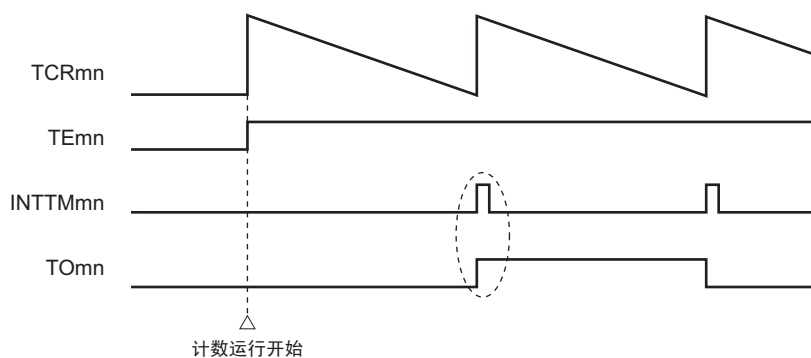
设定为间隔定时器模式（TOEmn=1、TOMmn=0）时的运行例子如下所示。

图 6-39 开始计数时的定时器中断和 TOn 输出的运行例子

(a) MDmn0 位为“1”的情况



(b) MDmn0 位为“0”的情况



当 MDmn0 位为“1”时，在开始计数时输出定时器中断（INTTMmn）并且 TOn 进行交替输出。

当 MDmn0 位为“0”时，在开始计数时不输出定时器中断（INTTMmn）并且 TOn 也不发生变化，而在对 1 个周期进行计数后输出 INTTMmn 并且 TOn 进行交替输出。

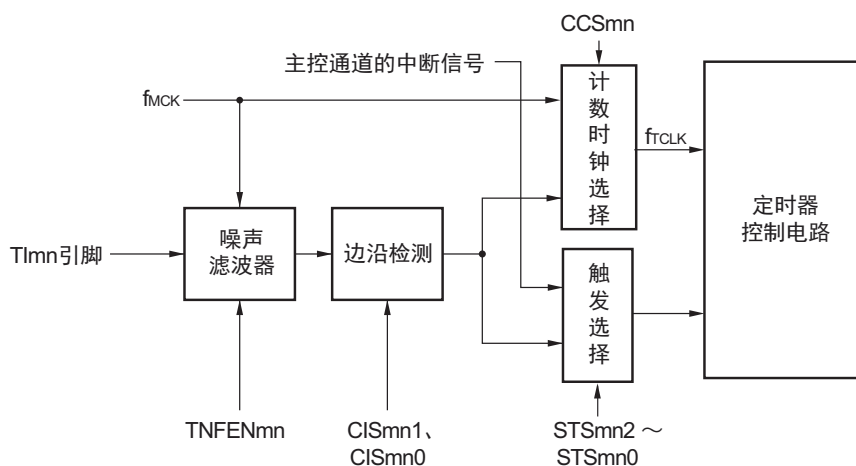
备注 m: 单元号 (m=0)、n: 通道号 (n=0 ~ 7)

6.7 定时器输入 (TImn) 的控制

6.7.1 TImn 的输入电路结构

定时器输入引脚的信号经过噪声滤波器和边沿检测电路输入到定时器控制电路。对于需要消除噪声的引脚，必须将对应引脚的噪声滤波器设定为有效。输入电路的结构图如图 6-40 所示。

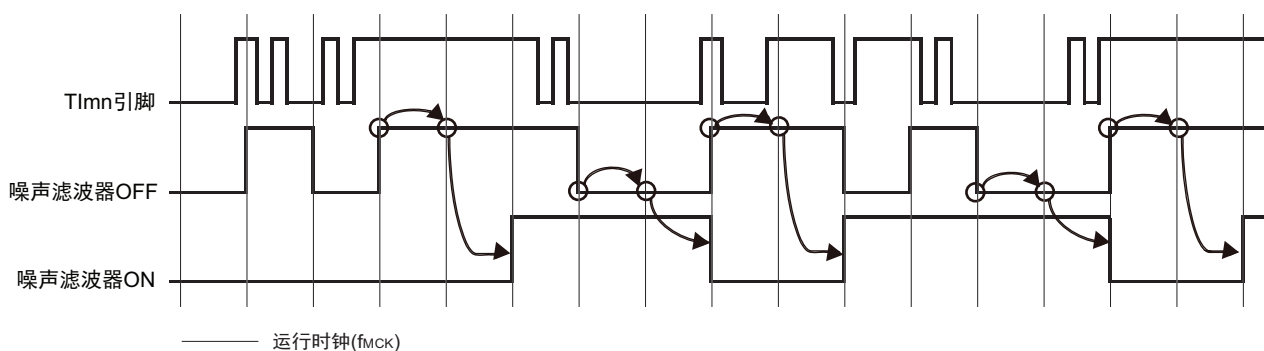
图 6-40 输入电路的结构图



6.7.2 噪声滤波器

噪声滤波器无效时，只通过通道 n 的运行时钟 (f_{MCK}) 进行同步。噪声滤波器有效时，在通过通道 n 的运行时钟 (f_{MCK}) 进行同步后，执行 2 个时钟的一致检测。在 TImn 输入引脚的噪声滤波器 ON/OFF 时经过噪声滤波器电路后的波形如图 6-41 所示。

图 6-41 TImn 输入引脚的噪声滤波器 ON/OFF 时的采样波形



注意 TImn 引脚的输入波形用于说明噪声滤波器 ON/OFF 的运行。实际使用时，请参照“32.4”所示的输入高 / 低电平宽度。

6.7.3 进行通道输入操作时的注意事项

在设定为不使用定时器输入引脚时，不向噪声滤波器电路提供运行时钟。因此，在设定为使用定时器输入引脚后，到设定定时器输入引脚的对应通道的运行允许触发前，需要以下的等待时间。

1. 噪声滤波器 OFF 的情况

在从定时器模式寄存器 mn (TMR mn) 的 bit12 (CCS mn)、bit9 (STS mn 1)、bit8 (STS mn 0) 全部为“0”的状态，到将其中 1 位置位后，必须在至少经过 2 个运行时钟 (f_{MCK}) 周期后，才能设定定时器通道开始寄存器 (TS m) 的运行允许触发。

2. 噪声滤波器 ON 的情况

在从定时器模式寄存器 mn (TMR mn) 的 bit12 (CCS mn)、bit9 (STS mn 1)、bit8 (STS mn 0) 全部为“0”的状态，到将其中 1 位置位后，必须在至少经过 4 个运行时钟 (f_{MCK}) 周期后，才能设定定时器通道开始寄存器 (TS m) 的运行允许触发。

6.8 定时器阵列单元的独立通道运行功能

6.8.1 作为间隔定时器 / 方波输出的运行

(1) 间隔定时器

能用作以固定间隔产生 INTTMmn（定时器中断）的基准定时器。

中断产生周期能用以下计算式进行计算：

$$\text{INTTMmn (定时器中断) 的产生周期} = \text{计数时钟周期} \times (\text{TDRmn 的设定值} + 1)$$

(2) 作为方波输出的运行

TOmn 在产生 INTTMmn 的同时进行交替输出，输出占空比为 50% 的方波。

TOmn 输出方波的周期和频率能用以下计算式进行计算：

$$\bullet \text{ TOmn 输出的方波周期} = \text{计数时钟周期} \times (\text{TDRmn 的设定值} + 1) \times 2$$

$$\bullet \text{ TOmn 输出的方波频率} = \text{计数时钟频率} / \{(\text{TDRmn 的设定值} + 1) \times 2\}$$

在间隔定时器模式中，定时器计数寄存器 mn（TCRmn）用作递减计数器。

在将定时器通道开始寄存器 m（TSM）的通道开始触发位（TSmn、TSHm1、TSHm3）置“1”后，通过第 1 个计数时钟将定时器数据寄存器 mn（TDRmn）的值加载到 TCRmn 寄存器。此时，如果定时器模式寄存器 mn（TMRmn）的 MDmn0 位为“0”，就不输出 INTTMmn 并且 TOmn 也不进行交替输出。如果 TMRmn 寄存器的 MDmn0 位为“1”，就输出 INTTMmn 并且 TOmn 进行交替输出。

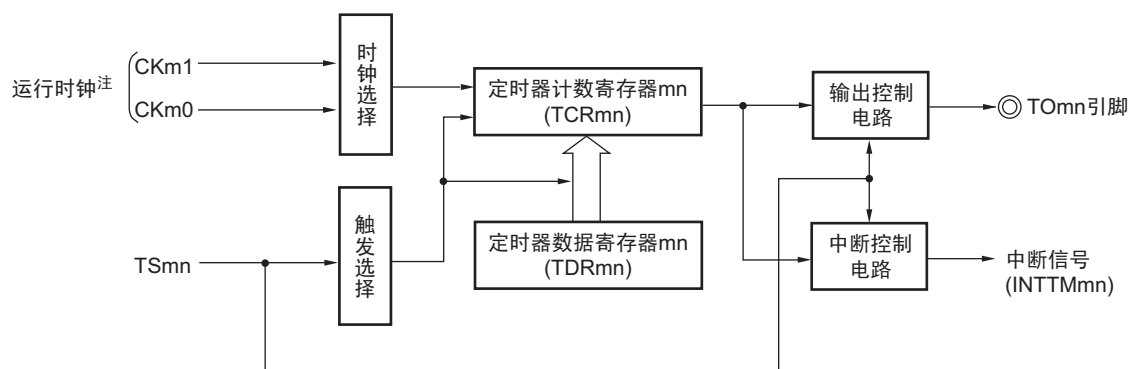
然后，TCRmn 寄存器通过计数时钟进行递减计数。

如果 TCRmn 变为“0000H”，就通过下一个计数时钟输出 INTTMmn 并且 TOmn 进行交替输出。同时，再次将 TDRmn 寄存器的值加载到 TCRmn 寄存器。此后，继续同样的运行。

能随时改写 TDRmn 寄存器，改写的 TDRmn 寄存器的值从下一个周期开始有效。

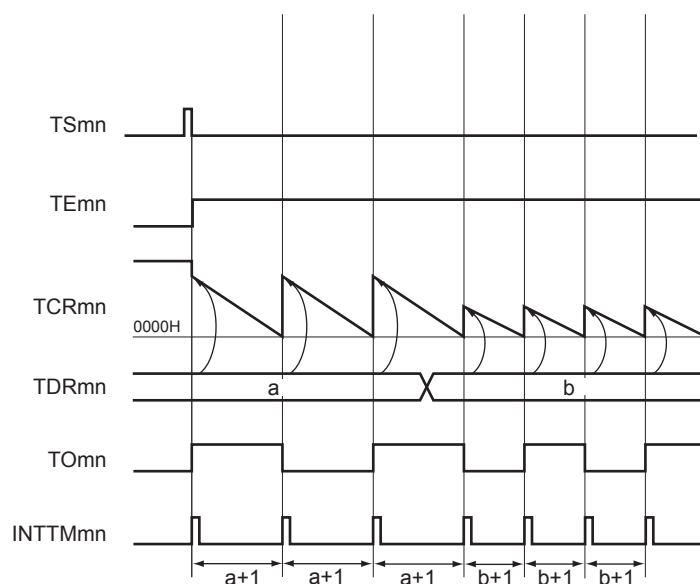
备注 m：单元号（m = 0）、n：通道号（n = 0 ~ 7）

图 6-42 作为间隔定时器 / 方波输出运行的框图



注 在通道 1 和通道 3 时，能从 CKm0、CKm1、CKm2 和 CKm3 中选择时钟。

图 6-43 作为间隔定时器 / 方波输出运行的基本时序例子 (MDmn0=1)



备注 1. m : 单元号 ($m = 0$)、 n : 通道号 ($n = 0 \sim 7$)

2. TSmn: 定时器通道开始寄存器 m (TS_m) 的 bit n

TE_{mn}: 定时器通道允许状态寄存器 m (TE_m) 的 bit n

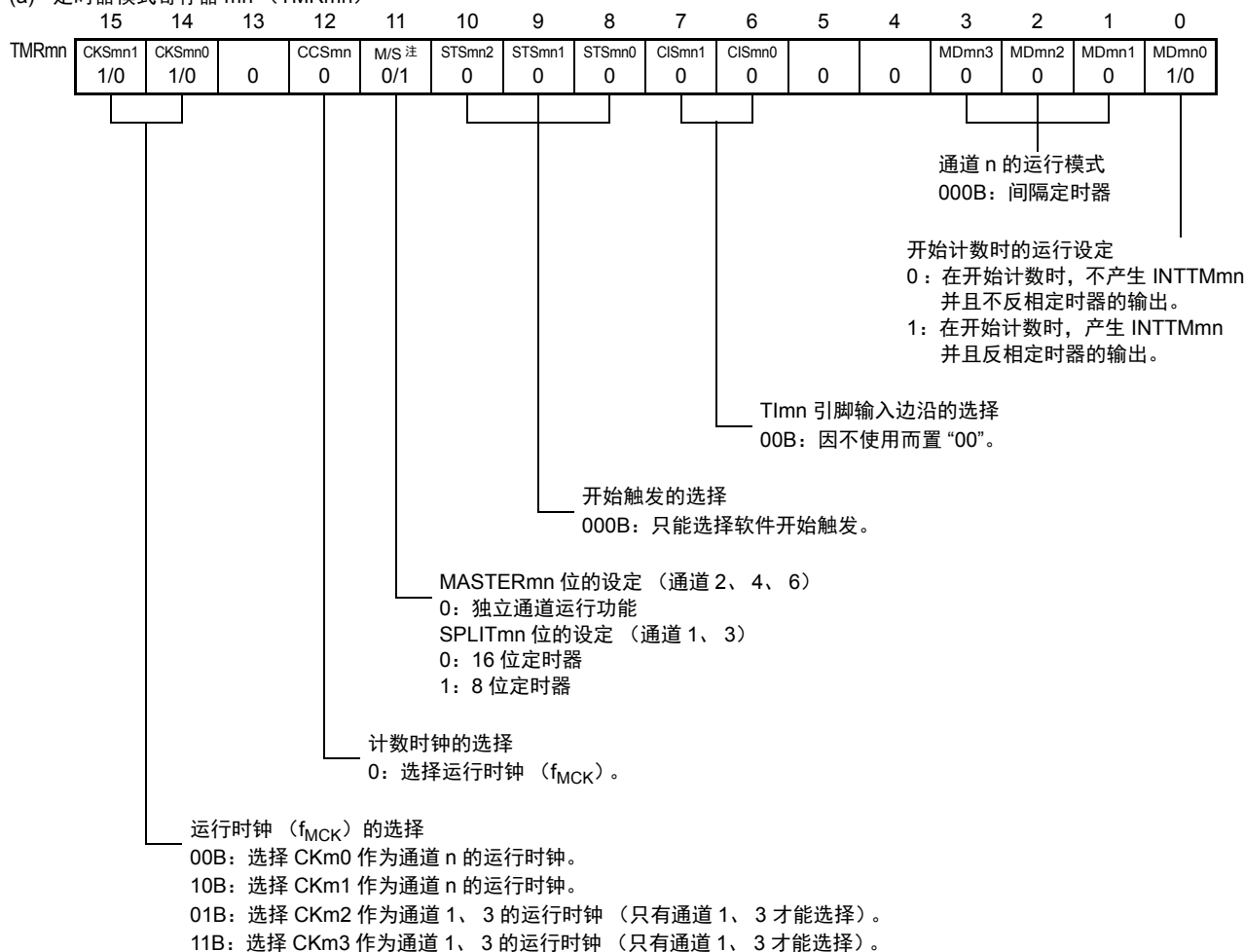
TCR_{mn}: 定时器计数寄存器 mn (TCR_{mn})

TDR_{mn}: 定时器数据寄存器 mn (TDR_{mn})

TO_{mn}: TO_{mn} 引脚输出信号

图 6-44 间隔定时器 / 方波输出时的寄存器设定内容例子

(a) 定时器模式寄存器 mn (TMRmn)



(b) 定时器输出寄存器 m (TOM)

bit n	
TOMn	0: 由 TOMn 输出“0”。
1/0	1: 由 TOMn 输出“1”。

(c) 定时器输出允许寄存器 m (TOEm)

bit n	
TOEmn	0: 停止由计数运行进行的 TOMn 输出。
1/0	1: 允许由计数运行进行的 TOMn 输出。

(d) 定时器输出电平寄存器 m (TOLm)

bit n	
TOLmn	0: 在 TOMmn=0 (主控通道输出模式) 时置“0”。
0	

(e) 定时器输出模式寄存器 m (TOMm)

bit n	
TOMmn	0: 设定主控通道输出模式。
0	

注 TMRm2、TMRm4、TMRm6: MASTERmn 位
TMRm1、TMRm3: SPLITmn 位
TMRm0、TMRm5、TMRm7: 固定为“0”。

备注 m: 单元号 (m=0)、n: 通道号 (n=0 ~ 7)

图 6-45 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	将外围允许寄存器 0 (PER0) 的 TAUmEN 位置“1”。→	断电状态 (停止提供时钟, 不能写各寄存器。)
	设定定时器时钟选择寄存器 m (TPSm)。 确定 CKm0 ~ CKm3 的时钟频率。	上电状态, 各通道处于运行停止状态。 (开始提供时钟, 能写各寄存器。)
通道初 始设定	设定定时器模式寄存器 mn (TMRmn) (确定通道的运行模式)。	通道处于运行停止状态。 (提供时钟, 消耗一部分电力。)
	给定时器数据寄存器 mn (TDRmn) 设定间隔 (周期) 值。	
	使用 TOMn 输出的情况: 将定时器输出模式寄存器 m (TOMm) 的 TOMmn 位置“0” (主控通道输出模式)。 将 TOLmn 位置“0”。	TOMn 引脚处于 Hi-Z 输出状态。
	设定 TOMn 位, 确定 TOMn 输出的初始电平。→ 将 TOEmn 位置“1”, 允许 TOMn 的运行。→ 将端口寄存器和端口模式寄存器置“0”。→	当端口模式寄存器为输出模式并且端口寄存器为“0”时, 输出 TOMn 初始设定的电平。 因为通道处于运行停止状态, 所以 TOMn 不变。 TOMn 引脚输出 TOMn 设定的电平。
重新 开始 运行	开始运行 (只在使用 TOMn 输出并且重新开始时, 将 TOEmn 位置“1”) 将 TSmn (TSHm1、TSHm3) 位置“1”。→ 因为 TSmn (TSHm1、TSHm3) 位是触发位, 所以自动返回到“0”。	TEmn (TEHm1、TEHm3) 位为“1”并且开始计数。 将 TDRmn 寄存器的值加载到定时器计数寄存器 mn (TCRmn)。当 TMRmn 寄存器的 MDmn0 位为“1”时, 产生 INTTMmn 并且 TOMn 进行交替输出。
	运行中 能任意更改 TDRmn 寄存器的设定值。 随时能读 TCRmn 寄存器。 不使用 TSRmn 寄存器。 能更改 TOM 寄存器和 TOEm 寄存器的设定值。 禁止更改 TMRmn 寄存器、TOMmn 位和 TOLmn 位的设定值。	计数器 (TCRmn) 进行递减计数。如果计数到“0000H”, 就再次将 TDRmn 寄存器的值加载到 TCRmn 寄存器, 继续进行计数。当检测到 TCRmn 为“0000H”时, 产生 INTTMmn 并且 TOMn 进行交替输出。 此后, 重复此运行。
	停止运行 将 TTmn (TTHm1、TTHm3) 位置“1”。→ 因为 TTmn (TTHm1、TTHm3) 位是触发位, 所以自动返回到“0”。 将 TOEmn 位置“0”并且给 TOMn 位设定值。→	TEmn (TEHm1、TEHmn) 位为“0”并且停止计数。 TCRmn 寄存器保持计数值而停止计数。 TOMn 输出不被初始化而保持状态。 TOMn 引脚输出 TOMn 位设定的电平。

图 6-45 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	要保持 TOmn 引脚输出电平的情况： 在端口寄存器设定要保持的值后将 TOmn 位置“0”。—————→	通过端口功能保持 TOmn 引脚的输出电平。
	不需要保持 TOmn 引脚输出电平的情况： 不需要设定。 ----- 将 PER0 寄存器的 TAUmEN 位置“0”。—————→	断电状态 对全部电路和各通道的 SFR 进行初始化。 (TOmn 位变为“0”并且 TOmn 引脚变为端口功能)

备注 m: 单元号 (m=0)、n: 通道号 (n=0 ~ 7)

6.8.2 作为外部事件计数器的运行

能用作事件计数器，对检测到的 TIMn 引脚输入的有效边沿（外部事件）进行计数，如果达到规定的计数值，就产生中断。规定的计数值能用以下计算式进行计算：

$$\text{规定的计数值} = \text{TDRmn 的设定值} + 1$$

在事件计数器模式中，定时器计数寄存器 mn（TCRmn）用作递减计数器。

通过将定时器通道开始寄存器 m（TSMn）的任意通道开始触发位（TSmn、TSHm1、TSHm3）置“1”，将定时器数据寄存器 mn（TDRmn）的值加载到 TCRmn 寄存器。

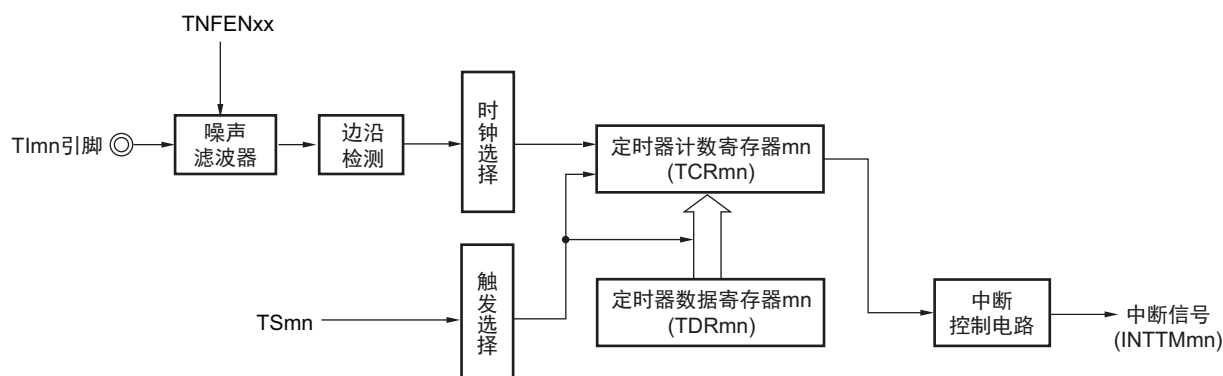
TCRmn 寄存器在检测到 TIMn 引脚输入的有效边沿的同时进行递减计数。如果 TCRmn 变为“0000H”，就再次加载到 TDRmn 寄存器的值并且输出 INTTMmn。

此后，继续同样的运行。

因为 TOMn 引脚根据外部事件输出不规则的波形，所以必须将定时器输出允许寄存器 m（TOEm）的 TOEmn 位置“0”，停止输出。

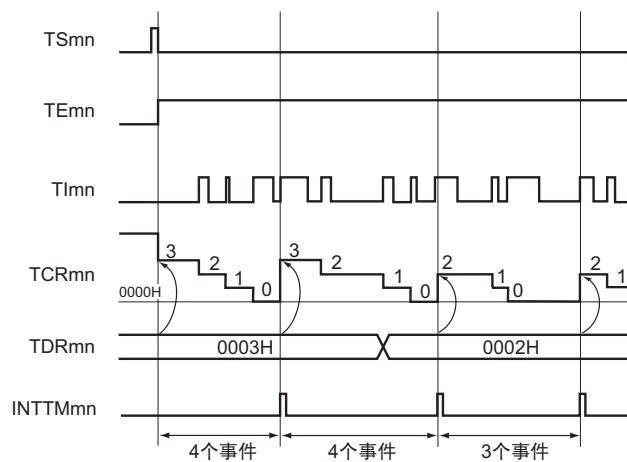
能随时改写 TDRmn 寄存器，改写的 TDRmn 寄存器的值在下一个计数期间有效。

图 6-46 作为外部事件计数器运行的框图



备注 m: 单元号 (m=0)、n: 通道号 (n=0 ~ 7)

图 6-47 作为外部事件计数器运行的基本时序例子

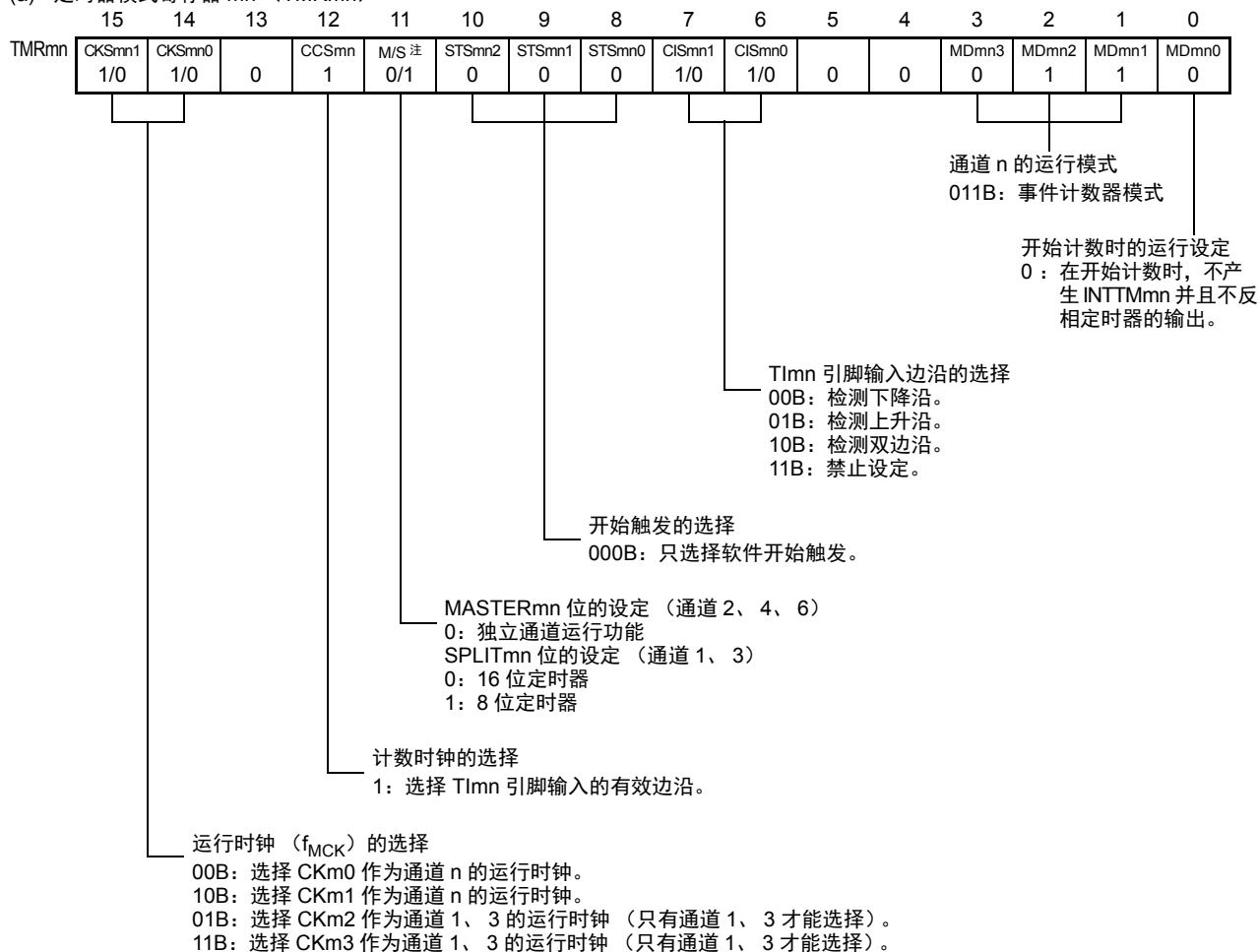


备注 1. m: 单元号 (m=0) n: 通道号 (n=0 ~ 7)

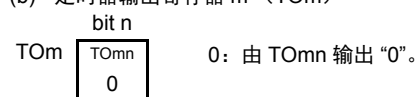
2. TSmn: 定时器通道开始寄存器 m (TSm) 的 bit n
TEmn: 定时器通道允许状态寄存器 m (TEm) 的 bit n
TImn: TImn 引脚输入信号
TCRmn: 定时器计数寄存器 mn (TCRmn)
TDRmn: 定时器数据寄存器 mn (TDRmn)

图 6-48 外部事件计数器模式时的寄存器设定内容例子

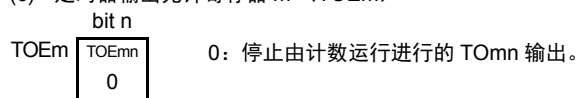
(a) 定时器模式寄存器 mn (TMRmn)



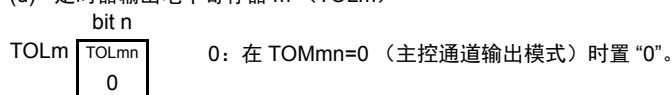
(b) 定时器输出寄存器 m (TOM)



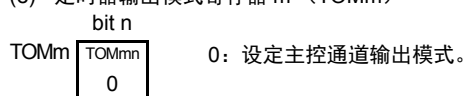
(c) 定时器输出允许寄存器 m (TOEm)



(d) 定时器输出电平寄存器 m (TOLm)



(e) 定时器输出模式寄存器 m (TOMm)



注 TMRm2、TMRm4、TMRm6: MASTERmn 位
TMRm1、TMRm3: SPLITmn 位
TMRm0、TMRm5、TMRm7: 固定为 "0"。

备注 m: 单元号 (m=0)、n: 通道号 (n=0 ~ 7)

图 6-49 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 (停止提供时钟, 不能写各寄存器。)
	将外围允许寄存器 0 (PER0) 的 TAUmEN 位置“1”。	上电状态, 各通道处于运行停止状态。 (开始提供时钟, 能写各寄存器。)
	设定定时器时钟选择寄存器 m (TPSm)。 确定 CKm0 ~ CKm3 的时钟频率。	
通道初 始设定	将噪声滤波器允许寄存器 1 (NFEN1) 的对应位置“0” (OFF) 或者“1” (ON)。 设定定时器模式寄存器 mn (TMRmn) (确定通道的运行模式)。 给定时器数据寄存器 mn (TDRmn) 设定计数值。 将定时器输出允许寄存器 m (TOEm) 的 TOEmn 位置“0”。	通道处于运行停止状态。 (提供时钟, 消耗一部分电力。)
重新 开始 运行	开始 运行	将 TSmn 位置“1”。 因为 TSmn 位是触发位, 所以自动返回到“0”。 TEmn 位为“1”并且开始计数。 将 TDRmn 寄存器的值加载到定时器计数寄存器 mn (TCRmn), 进入 TImn 引脚输入边沿的检测等待状态。
	运行中	能任意更改 TDRmn 寄存器的设定值。 随时能读 TCRmn 寄存器。 不使用 TSRmn 寄存器。 禁止更改 TMRmn 寄存器、TOMmn 位、TOLmn 位、TOmn 位和 TOEmn 位的设定值。 每当检测到 TImn 引脚的输入边沿时, 计数器 (TCRmn) 就进行递减计数。如果计数到“0000H”, 就再次将 TDRmn 寄存器的值加载到 TCRmn 寄存器, 继续进行计数。当检测到 TCRmn 为“0000H”时, 产生 INTTMmn。 此后, 重复此运行。
	停止 运行	将 TTmn 位置“1”。 因为 TTmn 位是触发位, 所以自动返回到“0”。 TEmn 位为“0”并且停止计数。 TCRmn 寄存器保持计数值而停止计数。
	TAU 停止	将 PER0 寄存器的 TAUmEN 位置“0”。 断电状态 对全部电路和各通道的 SFR 进行初始化。

备注 m: 单元号 (m=0) n: 通道号 (n=0 ~ 7)

6.8.3 作为输入脉冲间隔测量的运行

能在 TImn 有效边沿捕捉计数值，测量 TImn 输入脉冲的间隔。另外，在 TEmn 为“1”的期间，如果将软件操作（TSmn=1）设定为捕捉触发，就能捕捉计数值。

脉冲间隔能用以下计算式进行计算：

$$\text{TImn 输入脉冲间隔} = \text{计数时钟的周期} \times ((10000\text{H} \times \text{TSRmn:OVF}) + (\text{TDRmn 的捕捉值} + 1))$$

注意 因为通过定时器模式寄存器 mn（TMRmn）的 CKSmn 位选择的运行时钟对 TImn 引脚输入进行采样，所以产生最大 1 个运行时钟的误差。

在捕捉模式中，定时器计数寄存器 mn（TCRmn）用作递增计数器。

如果将定时器通道开始寄存器 m（TSM）的通道开始触发位（TSmn）置“1”，TCRmn 寄存器就通过计数时钟从“0000H”开始递增计数。

如果检测到 TImn 引脚输入的有效边沿，就将 TCRmn 寄存器的计数值传送（捕捉）到定时器数据寄存器 mn（TDRmn），同时将 TCRmn 寄存器清“0000H”，然后输出 INTTMmn。此时，如果计数器发生上溢，就将定时器状态寄存器 mn（TSRmn）的 OVF 位置“1”。如果计数器没有发生上溢，就清除 OVF 位。此后，继续同样的运行。

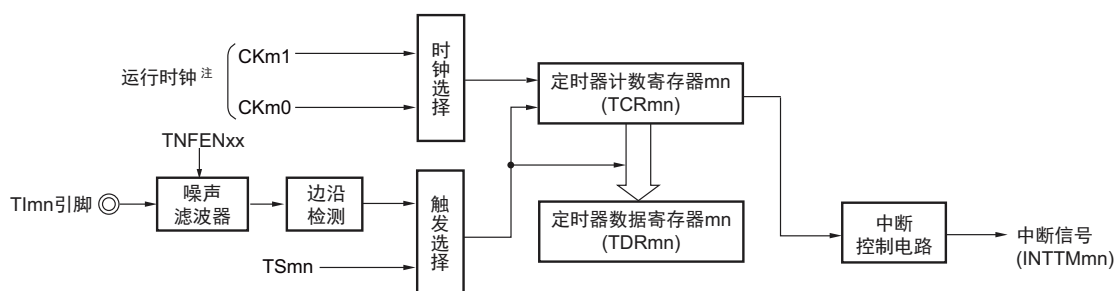
在将计数值捕捉到 TDRmn 寄存器的同时，根据在测量期间是否发生上溢，更新 TSRmn 寄存器的 OVF 位，并且能确认捕捉值的上溢状态。

即使计数器进行了 2 个或者 2 个周期以上的完整计数，也认为发生上溢而将 TSRmn 寄存器的 OVF 位置“1”。但是，在发生 2 次或者 2 次以上的上溢时，无法通过 OVF 位正常测量间隔值。

将 TMRmn 寄存器的 STSmn2 ~ STSmn0 位置“001B”，将 TImn 的有效边沿用于开始触发和捕捉触发。

在 TEmn=1 时，可以不使用 TImn 引脚输入，将软件操作（TSmn=1）用作捕捉触发。

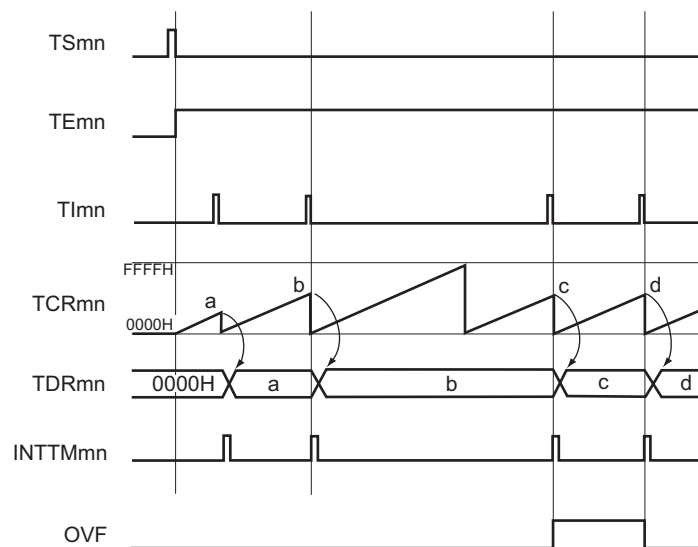
图 6-50 作为输入脉冲间隔测量运行的框图



注 在通道 1 和通道 3 时，能从 CKm0、CKm1、CKm2 和 CKm3 中选择时钟。

备注 m：单元号（m=0）、n：通道号（n=0 ~ 7）

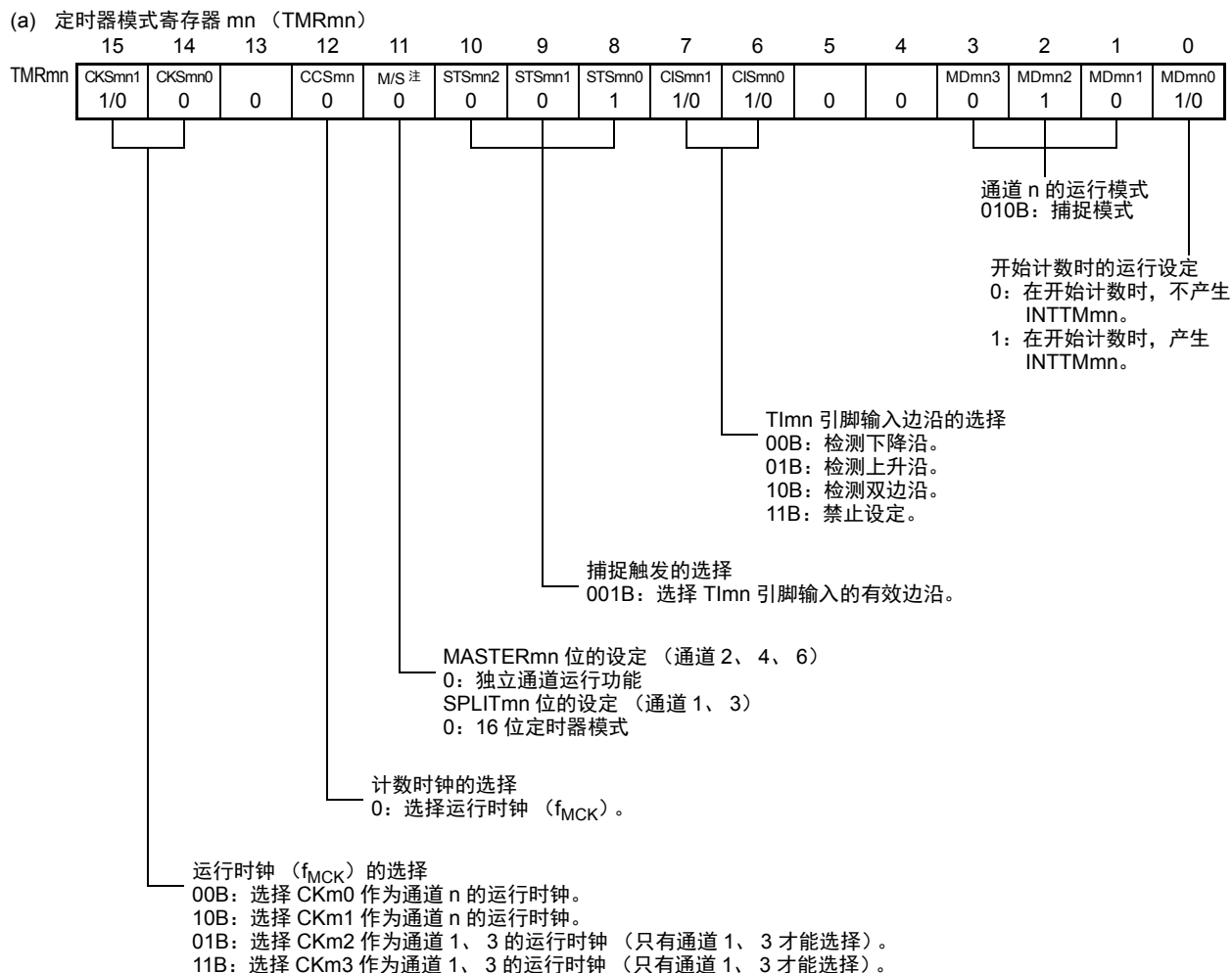
图 6-51 作为输入脉冲间隔测量的运行基本时序例子 (MDmn0=0)



备注 1. m: 单元号 (m=0)、n: 通道号 (n=0 ~ 7)

2. TSmn: 定时器通道开始寄存器 m (TSm) 的 bit n
- TEmn: 定时器通道允许状态寄存器 m (TEm) 的 bit n
- TImn: TImn 引脚输入信号
- TCRmn: 定时器计数寄存器 mn (TCRmn)
- TDRmn: 定时器数据寄存器 mn (TDRmn)
- OVF: 定时器状态寄存器 mn (TSRmn) 的 bit0

图 6-52 输入脉冲间隔测量时的寄存器设定内容例子



(b) 定时器输出寄存器 m (TOM)

bit n

TOMn 0: 由 TOMn 输出 "0"。

(c) 定时器输出允许寄存器 m (TOEm)

bit n

TOEmn 0: 停止由计数运行进行的 TOMn 输出。

(d) 定时器输出电平寄存器 m (TOLm)

bit n

TOLmn 0: 在 TOMmn=0 (主控通道输出模式) 时置 "0"。

(e) 定时器输出模式寄存器 m (TOMm)

bit n

TOMmn 0: 设定主控通道输出模式。

注 TMRm2、TMRm4、TMRm6: MASTERmn 位
TMRm1、TMRm3: SPLITmn 位
TMRm0、TMRm5、TMRm7: 固定为 "0"。

备注 m: 单元号 (m=0)、n: 通道号 (n=0 ~ 7)

图 6-53 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	将外围允许寄存器 m 的 TAUmEN 位置“1”。→	断电状态 (停止提供时钟, 不能写各寄存器。)
	设定定时器时钟选择寄存器 m (TPSm)。 确定 CKm0 ~ CKm3 的时钟频率。	上电状态, 各通道处于运行停止状态。 (开始提供时钟, 能写各寄存器。)
通道初 始设定	将噪声滤波器允许寄存器 1 (NFEN1) 的对应位置 “0” (OFF) 或者“1” (ON)。 设定定时器模式寄存器 mn (TMRmn) (确定通道 的运行模式)。	通道处于运行停止状态。 (提供时钟, 消耗一部分电力。)
开始 运行	将 TSmn 位置“1”。→ 因为 TSmn 位是触发位, 所以自动返回到“0”。	TEmn 位为“1”并且开始计数。 将定时器计数寄存器 mn (TCRmn) 清 “0000H”。当 TMRmn 寄存器的 MDmn0 位为“1” 时, 产生 INTTMmn。
运行中	只能更改 TMRmn 寄存器的 CISmn1 位和 CISmn0 位的设定值。 随时能读 TDRmn 寄存器。 随时能读 TCRmn 寄存器。 随时能读 TSRmn 寄存器。 禁止更改 TOMmn 位、TOLmn 位、TOMn 位和 TOEmn 位的设定值。	计数器 (TCRmn) 从“0000H”开始递增计数, 如果 检测到 TImn 引脚输入的有效边沿, 就将计数值传送 (捕捉) 到定时器数据寄存器 mn (TDRmn), 同 时将 TCRmn 寄存器清“0000H”并且产生 INTTMmn。 此时, 如果发生上溢, 就将定时器状态寄存器 mn (TSRmn) 的 OVF 位置位。如果不发生上溢, 就清 除 OVF 位。 此后, 重复此运行。
停止 运行	将 TTmn 位置“1”。→ 因为 TTmn 位是触发位, 所以自动返回到“0”。	TEmn 位变为“0”并且停止计数。 TCRmn 寄存器保持计数值而停止计数。 保持 TSRmn 寄存器的 OVF 位。
TAU 停止	将 PER0 寄存器的 TAUmEN 位置“0”。→	断电状态 对全部电路和各通道的 SFR 进行初始化。

备注 m: 单元号 (m=0)、n: 通道号 (n=0 ~ 7)

6.8.4 作为输入信号高 / 低电平宽度测量的运行

注意 当用作 LIN-bus 支持功能时，必须将输入切换控制寄存器（ISC）的 bit1（ISC1）置“1”。另外，在以下说明中，请用 RxD0 代替 TImn。

能通过 TImn 引脚输入的一个边沿开始计数并且在另一个边沿捕捉计数值，测量 TImn 的信号宽度（高电平宽度 / 低电平宽度）。TImn 的信号宽度能用以下计算式进行计算。

$$\text{TImn 输入的信号宽度} = \text{计数时钟的周期} \times ((10000\text{H} \times \text{TSRmn:OVF}) + (\text{TDRmn 的捕捉值} + 1))$$

注意 因为通过定时器模式寄存器 mn（TMRmn）的 CKSmn 位选择的运行时钟对 TImn 引脚输入进行采样，所以产生最大 1 个运行时钟的误差。

在捕捉 & 单次计数模式中，定时器计数寄存器 mn（TCRmn）用作递增计数器。

如果将定时器通道开始寄存器 m（TSM）的通道开始触发位（TSmn）置“1”，TEmn 位就变为“1”，并且进入 TImn 引脚的开始边沿检测等待状态。

如果检测到 TImn 引脚输入的开始边沿（在测量高电平宽度时为 TImn 引脚输入的上升沿），就与计数时钟同步，从“0000H”开始递增计数。然后，如果检测到有效捕捉边沿（在测量高电平宽度时为 TImn 引脚输入的下降沿），就在将计数值传送到定时器数据寄存器 mn（TDRmn）的同时，输出 INTTMmn。此时，如果计数器发生上溢，就将定时器状态寄存器 mn（TSRmn）的 OVF 位置位。如果计数器没有发生上溢，就清除 OVF 位。TCRmn 寄存器的值变为“传送到 TDRmn 寄存器的值 + 1”而停止计数，并且进入 TImn 引脚的开始边沿检测等待状态。此后，继续同样的运行。

在将计数值捕捉到 TDRmn 寄存器的同时，根据在测量期间是否发生上溢，更新 TSRmn 寄存器的 OVF 位，并且能确认捕捉值的上溢状态。

即使计数器进行了 2 个或者 2 个周期以上的完整计数，也认为发生上溢而将 TSRmn 寄存器的 OVF 位置“1”。但是，在发生 2 次或者 2 次以上的上溢时，无法通过 OVF 位正常测量间隔值。

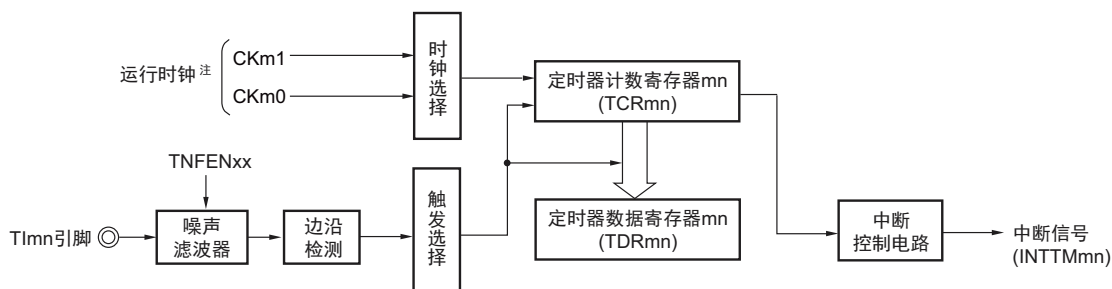
能通过 TMRmn 寄存器的 CISmn1 位和 CISmn0 位来设定是测量 TImn 引脚输入的高电平宽度还是低电平宽度。

此功能是以测量 TImn 引脚的输入信号宽度为目的，因此不能在 TE mn 位为“1”的期间将 TSmn 位置“1”。

TMRmn 寄存器的 CISmn1、CISmn0=10B：测量低电平宽度。

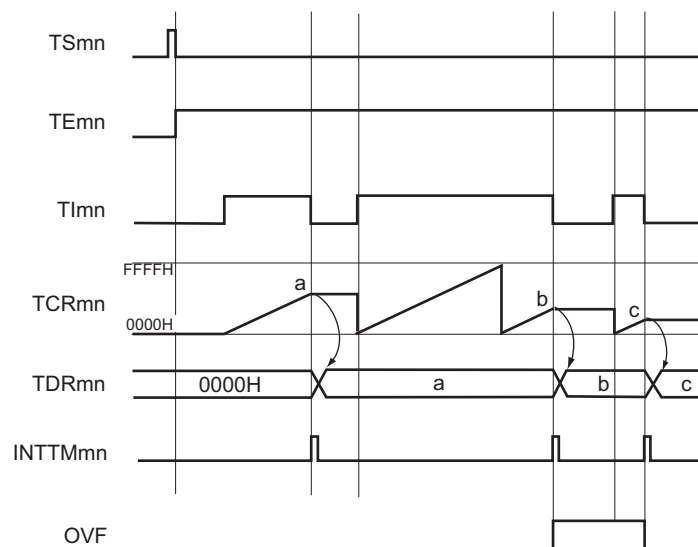
TMRmn 寄存器的 CISmn1、CISmn0=11B：测量高电平宽度。

图 6-54 作为输入信号高 / 低电平宽度测量运行的框图



注 在通道 1 和通道 3 时，能从 CKm0、CKm1、CKm2 和 CKm3 中选择时钟。

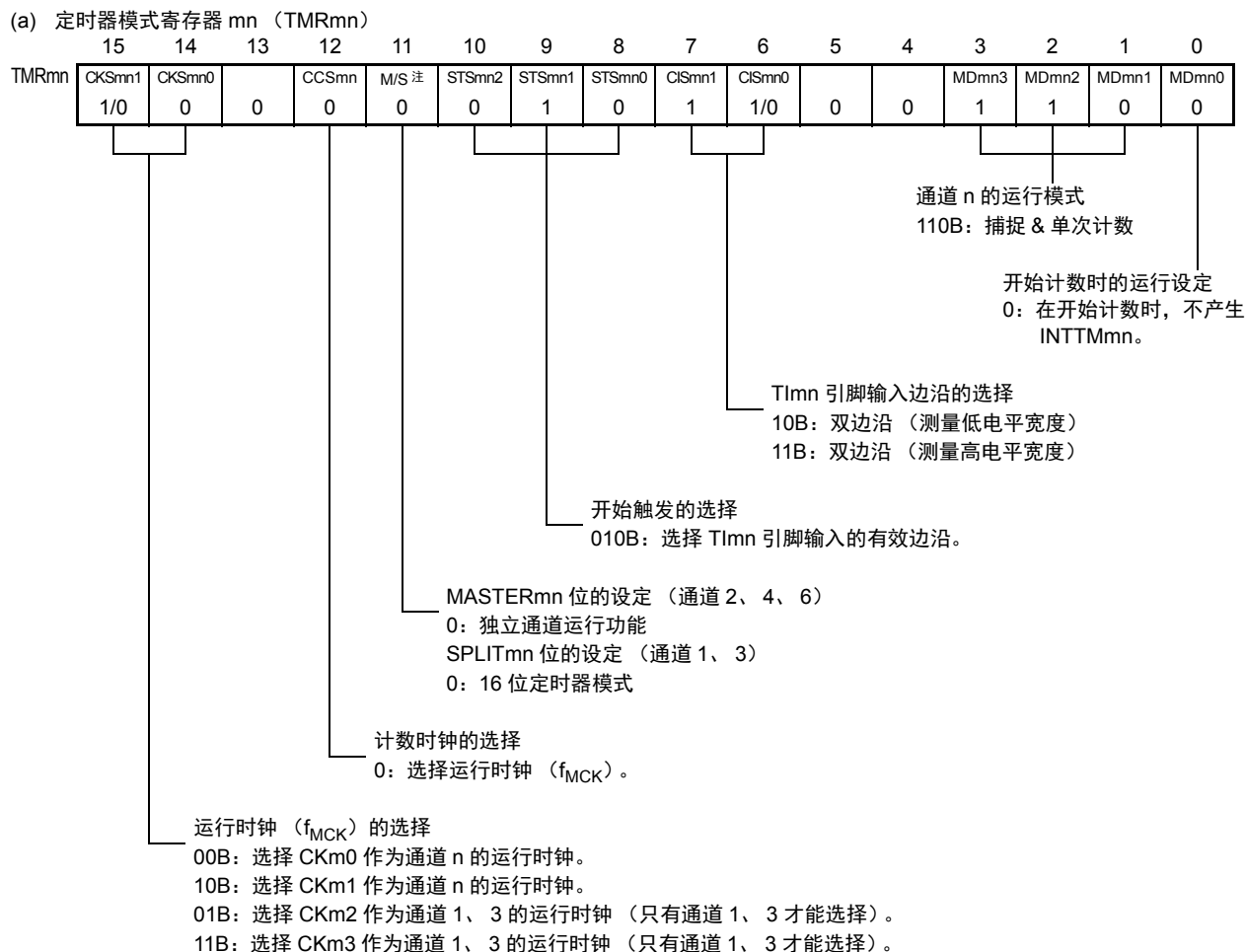
图 6-55 作为输入信号高 / 低电平宽度测量的运行基本时序例子



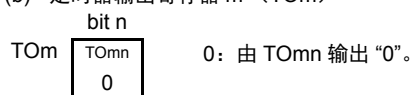
备注 1. m: 单元号 (m = 0)、n: 通道号 (n = 0 ~ 7)

2. TSmn: 定时器通道开始寄存器 m (TSm) 的 bit n
- TEmn: 定时器通道允许状态寄存器 m (TEm) 的 bit n
- TImn: TImn 引脚输入信号
- TCRmn: 定时器计数寄存器 mn (TCRmn)
- TDRmn: 定时器数据寄存器 mn (TDRmn)
- OVF: 定时器状态寄存器 mn (TSRmn) 的 bit0

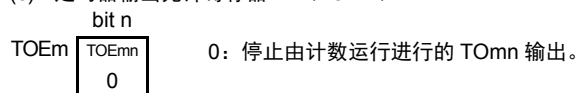
图 6-56 测量输入信号的高 / 低电平宽度时的寄存器设定内容例子



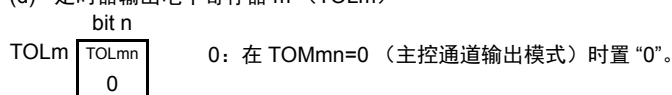
(b) 定时器输出寄存器 m (TOM)



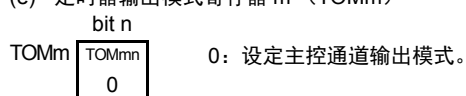
(c) 定时器输出允许寄存器 m (TOEm)



(d) 定时器输出电平寄存器 m (TOLm)



(e) 定时器输出模式寄存器 m (TOMm)



注 TMRm2、TMRm4、TMRm6: MASTERmn 位
TMRm1、TMRm3: SPLITmn 位
TMRm0、TMRm5、TMRm7: 固定为 "0"。

备注 m: 单元号 (m=0)、n: 通道号 (n=0 ~ 7)

图 6-57 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 (停止提供时钟, 不能写各寄存器。)
	将外围允许寄存器 0 (PER0) 的 TAUmEN 位置“1”。	上电状态, 各通道处于运行停止状态。 (开始提供时钟, 能写各寄存器。)
	设定定时器时钟选择寄存器 m (TPSm)。 确定 CKm0 ~ CKm3 的时钟频率。	
通道初 始设定	将噪声滤波器允许寄存器 1 (NFEN1) 的对应位置“0” (OFF) 或者“1” (ON)。 设定定时器模式寄存器 mn (TMRmn) (确定通道的运行模式)。 将 TOEmn 位置“0”, 并且停止 TOmn 运行。	通道处于运行停止状态。 (提供时钟, 消耗一部分电力。)
开始 运行	将 TSmn 位置“1”。 因为 TSmn 位是触发位, 所以自动返回到“0”。	TEmn 位为“1”并且进入 TImn 引脚的开始边沿检测等待状态。
	检测 TImn 引脚输入的计数开始边沿。	将定时器计数寄存器 mn (TCRmn) 清“0000H”并且开始递增计数。
运行中	能任意更改 TDRmn 寄存器的设定值。 随时能读 TCRmn 寄存器。 随时能读 TSRmn 寄存器。 禁止更改 TMRmn 寄存器、TOMmn 位、TOLmn 位、TOmn 位和 TOEmn 位的设定值。	在检测到 TImn 引脚的开始边沿后, 计数器 (TCRmn) 从“0000H”开始递增计数。如果检测到 TImn 引脚的捕捉边沿, 就将计数值传送到定时器数据寄存器 mn (TDRmn), 并且产生 INTTMmn。此时, 如果发生上溢, 就将定时器状态寄存器 mn (TSRmn) 的 OVF 位置位。如果没有发生上溢, 就清除 OVF 位。TCRmn 寄存器在检测到下一个 TImn 引脚的开始边沿前停止计数。此后, 重复此运行。
停止 运行	将 TTmn 位置“1”。 因为 TTmn 位是触发位, 所以自动返回到“0”。	TEmn 位为“0”并且停止计数。 TCRmn 寄存器保持计数值而停止计数。 保持 TSRmn 寄存器的 OVF 位。
TAU 停止	将 PER0 寄存器的 TAUmEN 位置“0”。	断电状态 对全部电路和各通道的 SFR 进行初始化。

重新
开始
运行

备注 m: 单元号 (m=0)、n: 通道号 (n=0 ~ 7)

6.8.5 作为延迟计数器的运行

能通过 TIMn 引脚输入的有效边沿检测（外部事件）开始递减计数，并且以任意的设定间隔产生 INTTMmn（定时器中断）。

在 TEmn 位为“1”的期间，能通过软件将 TSmn 位置“1”，开始递减计数，并且以任意的设定间隔产生 INTTMmn（定时器中断）。

中断产生周期能用以下计算式进行计算：

$$\text{INTTMmn（定时器中断）的产生周期} = \text{计数时钟的周期} \times (\text{TDRmn 的设定值} + 1)$$

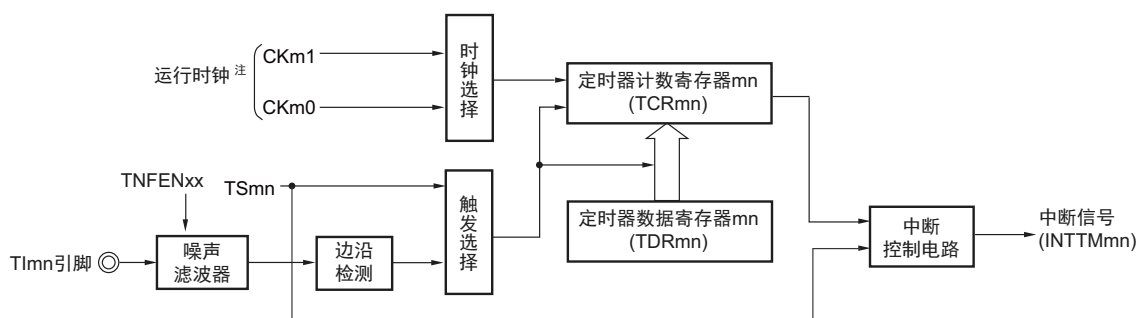
在单次计数模式中，定时器计数寄存器 mn（TCRmn）用作递减计数器。

如果将定时器通道开始寄存器 m（TSM）的通道开始触发位（TSmn、TSHm1、TSHm3）置“1”，TEmn 位、TEHm1 位和 TEHm3 位就变为“1”，并且进入 TIMn 引脚的有效边沿检测等待状态。

通过 TIMn 引脚输入的有效边沿检测，开始 TCRmn 寄存器的运行，并且加载到定时器数据寄存器 mn（TDRmn）的值。TCRmn 寄存器通过计数时钟，从被加载的 TDRmn 寄存器的值开始递减计数。如果 TCRmn 变为“0000H”，就输出 INTTMmn，并且在检测到下一个 TIMn 引脚输入的有效边沿前停止计数。

能随时改写 TDRmn 寄存器，改写的 TDRmn 寄存器的值从下一个周期开始有效。

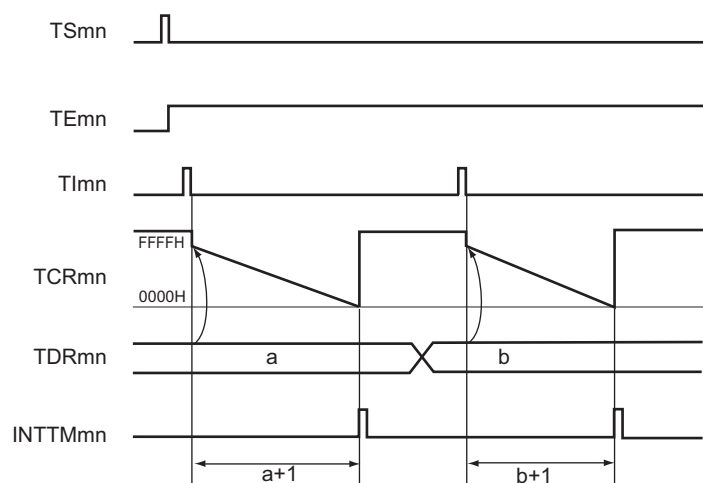
图 6-58 作为延迟计数器运行的框图



注 在通道 1 和通道 3 时，能从 CKm0、CKm1、CKm2 和 CKm3 中选择时钟。

备注 m: 单元号 (m=0)、n: 通道号 (n=0 ~ 7)

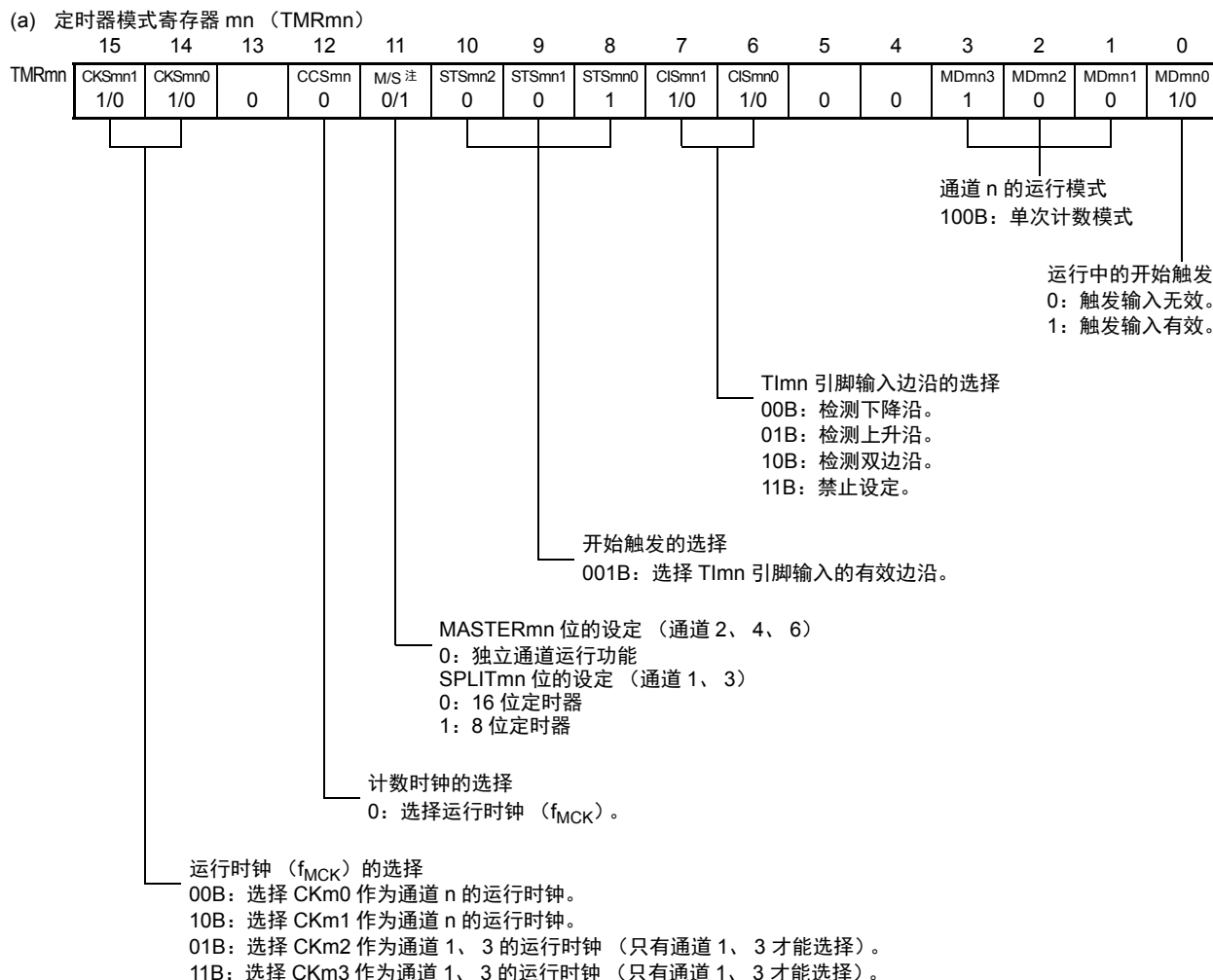
图 6-59 作为延迟计数器的运行基本时序例子



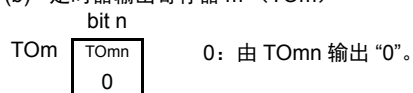
备注 1. m: 单元号 (m = 0)、n: 通道号 (n = 0 ~ 7)

2. TSmn: 定时器通道开始寄存器 m (TSM) 的 bit n
- TE mn: 定时器通道允许状态寄存器 m (TEM) 的 bit n
- TImn: TImn 引脚输入信号
- TCRmn: 定时器计数寄存器 mn (TCRmn)
- TDRmn: 定时器数据寄存器 mn (TDRmn)

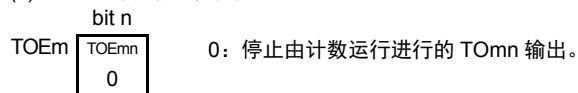
图 6-60 延迟计数器功能时的寄存器设定内容例子



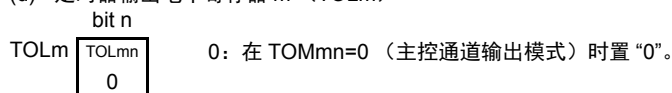
(b) 定时器输出寄存器 m (TOM)



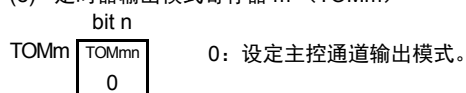
(c) 定时器输出允许寄存器 m (TOEm)



(d) 定时器输出电平寄存器 m (TOLm)



(e) 定时器输出模式寄存器 m (TOMm)



注 TMRm2、TMRm4、TMRm6: MASTERmn 位
TMRm1、TMRm3: SPLITmn 位
TMRm0、TMRm5、TMRm7: 固定为 "0"。

备注 m: 单元号 (m=0)、n: 通道号 (n=0 ~ 7)

图 6-61 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	将外围允许寄存器 0（PER0）的 TAUmEN 位置“1”。—————▶	断电状态 （停止提供时钟，不能写各寄存器。）	
	设定定时器时钟选择寄存器 m（TPSm）。 确定 CKm0 ～ CKm3 的时钟频率。	上电状态，各通道处于运行停止状态。 （开始提供时钟，能写各寄存器。）	
	通道初始设定	将噪声滤波器允许寄存器 1（NFEN1）的对应位置“0”（OFF）或者“1”（ON）。 设定定时器模式寄存器 mn（TMRmn）（确定通道 n 的运行模式）。 给定时器数据寄存器 mn（TDRmn）设定输出延迟。 将 TOEmn 位置“0”并且停止 TOMn 运行。	通道处于运行停止状态。 （提供时钟，消耗一部分电力。）
重新 开始 运行	开始运行	将 TSmn 位置“1”。—————▶ 因为 TSmn 位是触发位，所以自动返回到“0”。 检测到 TImn 引脚输入的有效边沿 —————▶	TEmn 位为“1”并且进入开始触发检测（检测 TImn 引脚的有效边沿，或者将 TSmn 位置“1”）等待状态。 将 TDRmn 寄存器的值加载到定时器计数寄存器 mn（TCRmn）。
	运行中	能任意更改 TDRmn 寄存器的设定值。 随时能读 TCRmn 寄存器。 不使用 TSRmn 寄存器。	计数器（TCRmn）进行递减计数。如果 TCRmn 计数到“0000H”，就输出 INTTMmn，直到下一个 TImn 引脚输入前 TCRmn 为“0000H”而且停止计数。
	停止运行	将 TTmn 位置“1”。—————▶ 因为 TTmn 位是触发位，所以自动返回到“0”。	TEmn 位为“0”并且停止计数。 TCRmn 寄存器保持计数值而停止计数。
	TAU 停止	将 PER0 寄存器的 TAUmEN 位置“0”。—————▶	断电状态 对全部电路和各通道的 SFR 进行初始化。

备注 m: 单元号 (m=0) n: 通道号 (n=0 ~ 7)

6.9 定时器阵列单元的多通道联动运行功能

6.9.1 作为单触发脉冲输出功能的运行

将 2 个通道成对使用，能通过 TIMn 引脚的输入生成任意延迟脉宽的单触发脉冲。
延迟和脉宽能用以下计算式进行计算：

$$\begin{aligned} \text{延迟} &= \{\text{TDRmn (主控) 的设定值} + 2\} \times \text{计数时钟周期} \\ \text{脉宽} &= \{\text{TDRmp (从属) 的设定值}\} \times \text{计数时钟周期} \end{aligned}$$

在单次计数模式中，主控通道运行并且对延迟进行计数。通过开始触发的检测，主控通道的定时器计数寄存器 mn (TCRmn) 开始运行并且加载到定时器数据寄存器 mn (TDRmn) 的值。TCRmn 寄存器通过计数时钟，从被加载的 TDRmn 寄存器的值开始递减计数。如果 TCRmn 变为“0000H”，就输出 INTTMmn，并且在检测到下一个开始触发前停止计数。

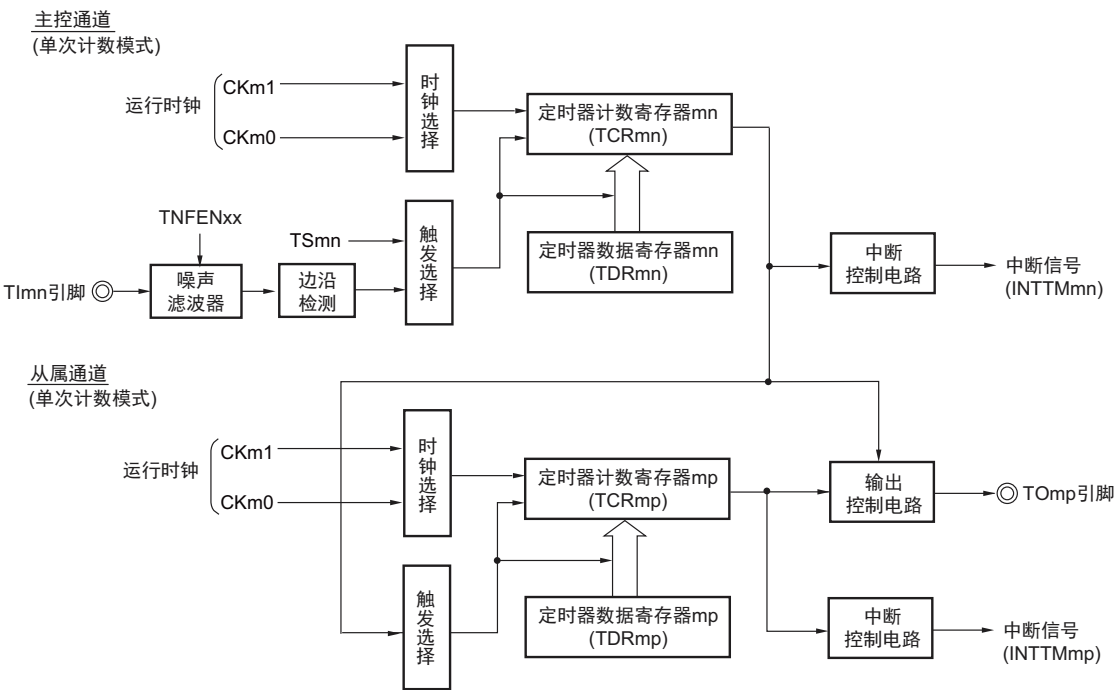
在单次计数模式中，从属通道运行并且对脉宽进行计数。将主控通道的 INTTMmn 作为开始触发，从属通道的 TCRmp 寄存器开始运行并且加载到 TDRmp 寄存器的值。TCRmp 寄存器通过计数时钟，从被加载的 TDRmp 寄存器值开始递减计数。如果计数值变为“0000H”，就输出 INTTMmp，并且在检测到下一个开始触发（主控通道的 INTTMmn）前停止计数。在从主控通道产生 INTTMmn 并且经过 1 个计数时钟后，TOMP 的输出电平变为有效电平，如果 TCRmp 变为“0000H”，就变为无效电平。

也能不使用 TIMn 引脚输入而将软件操作 (TSmn=1) 作为开始触发来输出单触发脉冲。

注意 因为主控通道的定时器数据寄存器 mn (TDRmn) 和从属通道的 TDRmp 寄存器的加载时序不同，所以如果在运行中改写 TDRmn 寄存器和 TDRmp 寄存器，就输出不正常的波形。必须在产生 INTTMmn 后改写 TDRmn 寄存器，并且在产生 INTTMmp 后改写 TDRmp 寄存器。

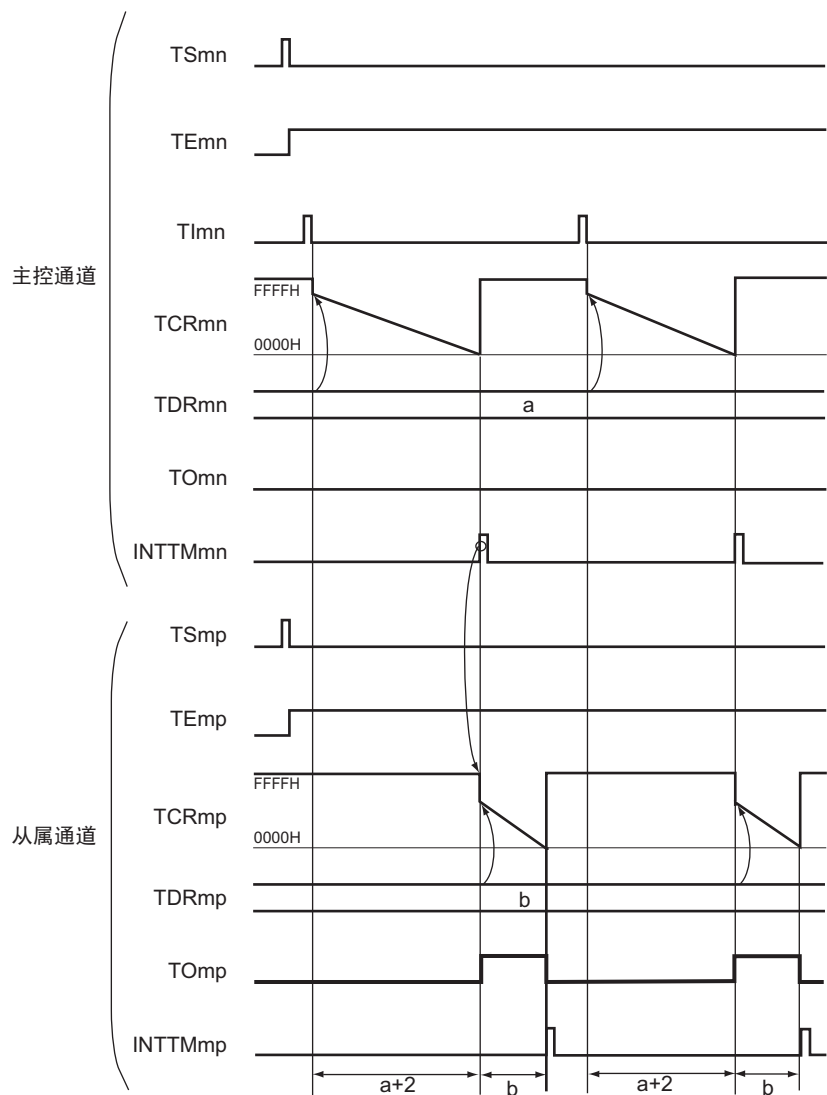
备注 m: 单元号 (m=0) n: 主控通道号 (n=0、2、4、6)
p: 从属通道号 (n < p ≤ 7)

图 6-62 作为单触发脉冲输出功能运行的框图



备注 m: 单元号 (m=0) n: 主控通道号 (n=0、2、4、6)
p: 从属通道号 (n<p≤7)

图 6-63 作为单触发脉冲输出功能的运行基本时序例子



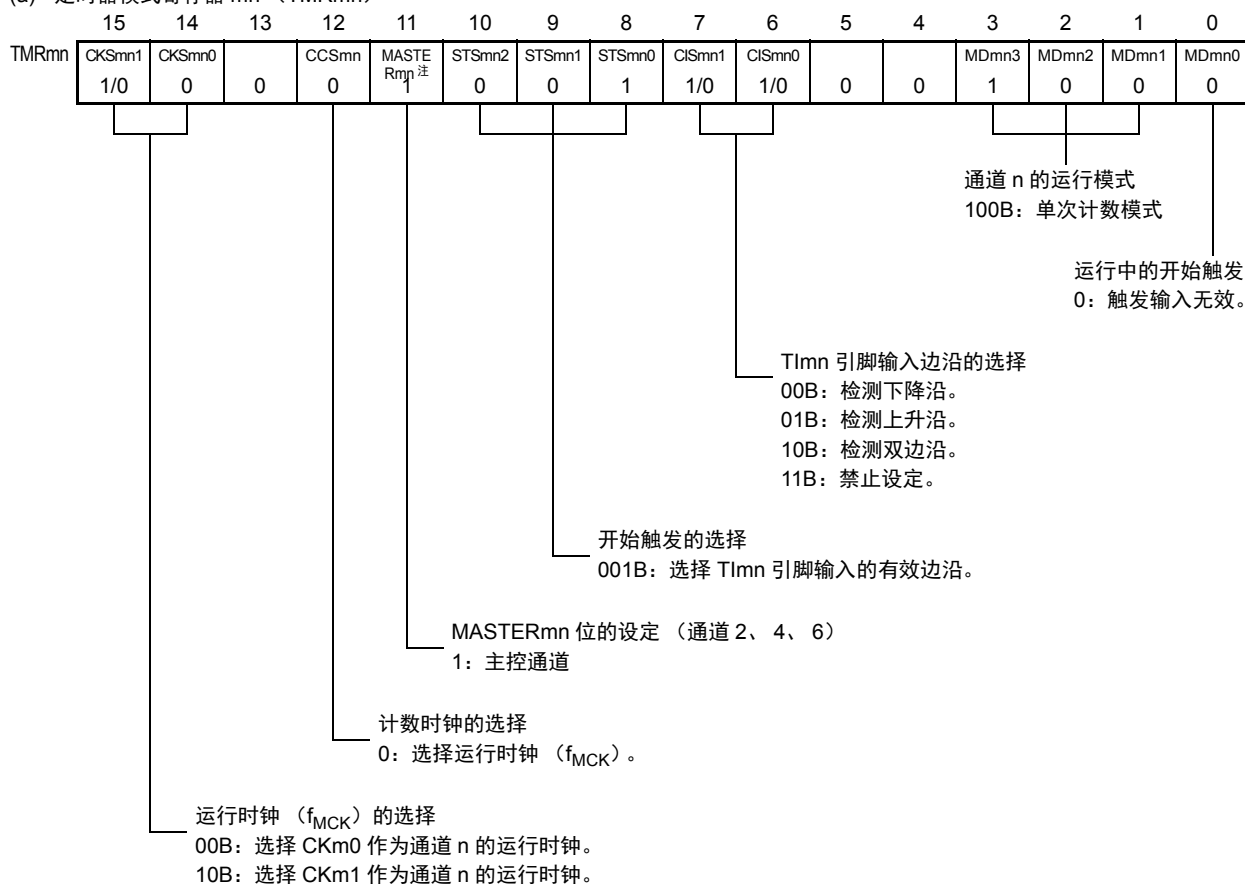
备注 1. m: 单元号 ($m = 0$)、n: 通道号 ($n = 0, 2, 4, 6$)

p: 从属通道号 ($n < p \leq 7$)

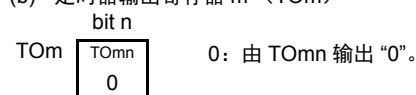
2. TSmn、TSmp: 定时器通道开始寄存器 m (TSM) 的 bit n、p
- TEmn、TEmp: 定时器通道允许状态寄存器 m (TEM) 的 bit n、p
- TImn、TImp: TIMn 和 TIMp 引脚输入信号
- TCRmn、TCRmp: 定时器计数寄存器 mn、mp (TCRmn、TCRmp)
- TDRmn、TDRmp: 定时器数据寄存器 mn、mp (TDRmn、TDRmp)
- TOmn、TOmp: TOmn 和 TOmp 引脚输出信号

图 6-64 单触发脉冲输出功能时（主控通道）的寄存器设定内容例子

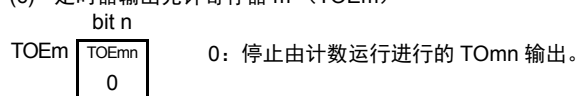
(a) 定时器模式寄存器 mn (TMRmn)



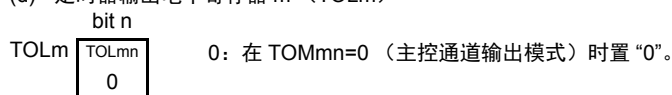
(b) 定时器输出寄存器 m (TOM)



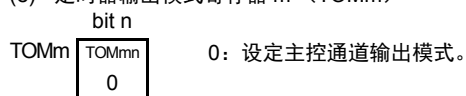
(c) 定时器输出允许寄存器 m (TOEm)



(d) 定时器输出电平寄存器 m (TOLm)



(e) 定时器输出模式寄存器 m (TOMm)

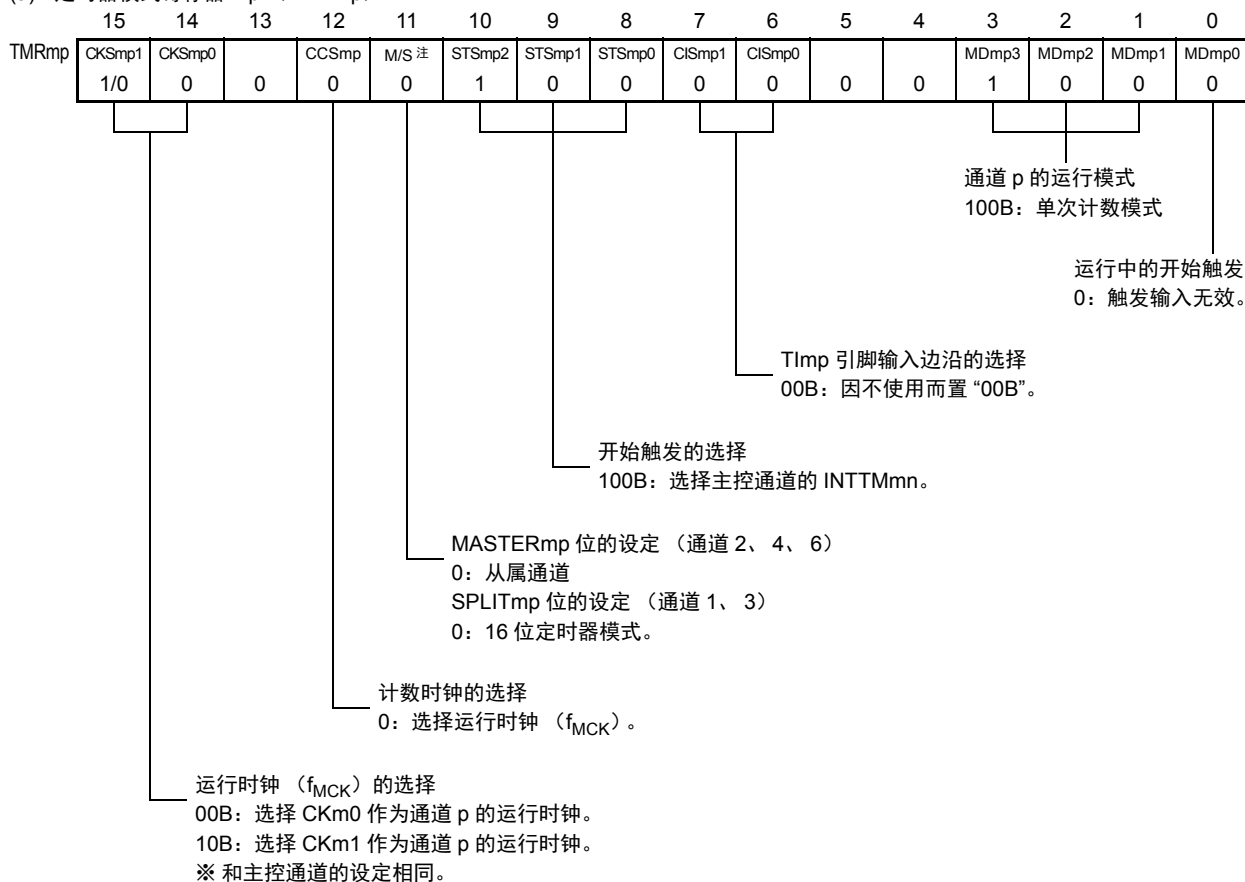


注 TMRm2、TMRm4、TMRm6: MASTERmn=1
TMRm0: 固定为“0”

备注 m: 单元号 (m=0) n: 主控通道号 (n=0、2、4、6)

图 6-65 单触发脉冲输出功能时（从属通道）的寄存器设定内容例子

(a) 定时器模式寄存器 mp (TMRmp)



(b) 定时器输出寄存器 m (TOM)

bit p	
TOM	TOMP
	1/0
	0: 由 TOMp 输出“0”。
	1: 由 TOMp 输出“1”。

(c) 定时器输出允许寄存器 m (TOEm)

bit p	
TOEm	TOEmp
	1/0
	0: 停止由计数运行进行的 TOMp 输出。
	1: 允许由计数运行进行的 TOMp 输出。

(d) 定时器输出电平寄存器 m (TOLm)

bit p	
TOLm	TOLmp
	1/0
	0: 正逻辑输出（高电平有效）
	1: 负逻辑输出（低电平有效）

(e) 定时器输出模式寄存器 m (TOMm)

bit p	
TOMm	TOMmp
	1
	1: 设定从属通道输出模式。

注 TMRm2、TMRm4、TMRm6: MASTERmp 位
TMRm1、TMRm3: SPLITmp 位
TMRm5、TMRm7: 固定为“0”。

备注 m: 单元号 (m=0) n: 主控通道号 (n=0、2、4、6)
p: 从属通道号 (n < p ≤ 7)

图 6-66 单触发脉冲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 (停止提供时钟, 不能写各寄存器。)
	将外围允许寄存器 0 (PER0) 的 TAUmEN 位置“1”。	上电状态, 各通道处于运行停止状态。 (开始提供时钟, 能写各寄存器。)
	设定定时器时钟选择寄存器 m (TPSm)。 确定 CKm0 和 CKm1 的时钟频率。	
通道初 始设定	将噪声滤波器允许寄存器 1 (NFEN1) 的对应位置“0” (OFF) 或者置“1” (ON)。 设定使用的 2 个通道的定时器模式寄存器 mn、mp (TMRmn、TMRmp) (确定通道的运行模式)。 给主控通道的定时器数据寄存器 mn (TDRmn) 设定输出延迟时间, 并且给从属通道的 TDRmp 寄存器设定脉宽。	通道处于运行停止状态。 (提供时钟, 消耗一部分电力。)
	从属通道的设定 将定时器输出模式寄存器 m (TOMm) 的 TOMmp 位置“1” (从属通道输出模式)。 设定 TOLmp 位。 设定 TOmp 位并且确定 TOmp 输出的初始电平。	TOmp 引脚处于 Hi-Z 输出状态。
	将 TOEmp 位置“1”, 允许 TOmp 运行。	当端口模式寄存器为输出模式并且端口寄存器为“0”时, 输出 TOmp 初始设定的电平。
	将端口寄存器和端口模式寄存器置“0”。	因为通道处于运行停止状态, 所以 TOmp 不变。
		TOmp 引脚输出 TOmp 设定的电平。

图 6-66 单触发脉冲输出功能时的操作步骤 (2/2)

	软件操作	硬件状态
重新开始运行	开始运行 将 TOEmp 位 (从属) 置 “1” (只限于重新开始运行)。 将定时器通道开始寄存器 m (TSm) 的 TSmn (主控) 和 TSmp (从属) 位同时置 “1”。——→ 因为 TSmn 位和 TSmp 位是触发位, 所以自动返回到 “0”。 通过检测主控通道的开始触发, 开始主控通道的计数运行。 - 检测到 TImn 引脚输入的有效边沿 - 通过软件将主控通道的 TSmn 位置 “1” 注 注 不能将从属通道的 TSmn 位置 “1”。	TEmn 位和 TEm p 位都为 “1”, 主控通道进入开始触发检测 (检测 TImn 引脚输入的有效边沿, 或者将主控通道的 TSmn 位置 “1”) 等待状态。 计数器还处于停止状态。 主控通道开始计数。
	运行中 只能更改 TMRmn 寄存器的 CISmn1 位和 CISmn0 位的设定值。 禁止更改 TMRmp、TDRmn、TDRmp 寄存器以及 TOMmn 位、TOMmp 位、TOLmn 位和 TOLmp 位的设定值。 随时能读 TCRmn 寄存器和 TCRmp 寄存器。 不使用 TSRmn 寄存器和 TSRmp 寄存器。 能更改从属通道的 TOm 寄存器和 TOEm 寄存器的设定值。	如果主控通道通过开始触发检测 (检测 TImn 引脚输入的有效边沿, 或者将主控通道的 TSmn 位置 “1”), 就将 TDRmn 寄存器的值加载到定时器计数寄存器 mn (TCRmn), 并且开始递减计数。 如果 TCRmn 计数到 “0000H”, 就输出 INTTMmn, 并且在下一个 TImn 引脚输入前停止计数。 从属通道以主控通道的 INTTMmn 作为触发, 将 TDRmp 寄存器的值加载到 TCRmp 寄存器并且开始递减计数。在从主控通道输出 INTTMmn 并且经过 1 个计数时钟后, 将 TOmp 的输出电平置为有效电平。然后, 如果 TCRmp 计数到 “0000H”, 就在将 TOmp 的输出电平置为无效电平后停止计数。此后, 重复此运行。
	停止运行 将 TTmn 位 (主控) 和 TTmp 位 (从属) 同时置 “1”。——→ 因为 TTmn 位和 TTmp 位是触发位, 所以自动返回到 “0”。 将从属通道的 TOEmp 位置 “0” 并且给 TOmp 位设定值。——→	TEmn 位和 TEm p 位都变为 “0” 并且停止计数。 TCRmn 寄存器和 TCRmp 寄存器保持计数值而停止计数。TOmp 输出不被初始化而保持状态。 TOmp 引脚输出 TOmp 设定的电平。
	TAU 停止 要保持 TOmp 引脚输出电平的情况: 在给端口寄存器设定要保留的值后将 TOmp 位置 “0”。——→ 不需要保持 TOmp 引脚输出电平的情况: 不需要设定。 将 PER0 寄存器的 TAUmEN 位置 “0”。——→	通过端口功能保持 TOmp 引脚的输出电平。 断电状态 对全部电路和各通道的 SFR 进行初始化。 (TOmp 位变为 “0” 并且 TOmp 引脚变为端口功能)

备注 m: 单元号 (m=0) n: 主控通道号 (n=0、2、4、6)
p: 从属通道号 (n < p ≤ 7)

6.9.2 作为 PWM 功能的运行

将 2 个通道成对使用，能生成任意周期和占空比的脉冲。

输出脉冲的周期和占空比能用以下计算式进行计算：

脉冲周期 = {TDRmn (主控) 的设定值 + 1} × 计数时钟周期
 占空比 [%] = {TDRmp (从属) 的设定值} / {TDRmn (主控) 的设定值 + 1} × 100
 0% 输出: TDRmp (从属) 的设定值 = 0000H
 100% 输出: TDRmp (从属) 的设定值 ≥ {TDRmn (主控) 的设定值 + 1}

备注 当 TDRmp (从属) 的设定值 > {TDRmn (主控) 的设定值 + 1} 时，占空比超过 100%，但是为 100% 输出。

主控通道用作间隔定时器模式。如果将定时器通道开始寄存器 m (TSm) 的通道开始触发位 (TSmn) 置“1”，就输出中断 (INTTMmn)，然后将定时器数据寄存器 mn (TDRmn) 的设定值加载到定时器计数寄存器 mn (TCRmn)，并且通过计数时钟进行递减计数。当计数到“0000H”时，在输出 INTTMmn 中断后再次将 TDRmn 寄存器的值加载到 TCRmn 寄存器，并且进行递减计数。此后，在将定时器通道停止寄存器 m (TTm) 的通道停止触发位 (TTmn) 置“1”前，重复此运行。

当用作 PWM 功能时，主控通道进行递减计数，在计数到“0000H”为止的期间为 PWM 输出 (TOmp) 周期。

从属通道用作单次计数模式。以主控通道的 INTTMmn 中断为开始触发，将 TDRmp 寄存器的值加载到 TCRmp 寄存器，并且进行递减计数，计数到“0000H”为止。当计数到“0000H”时，输出 INTTMmp，并且等待下一个开始触发 (主控通道的 INTTMmn)。

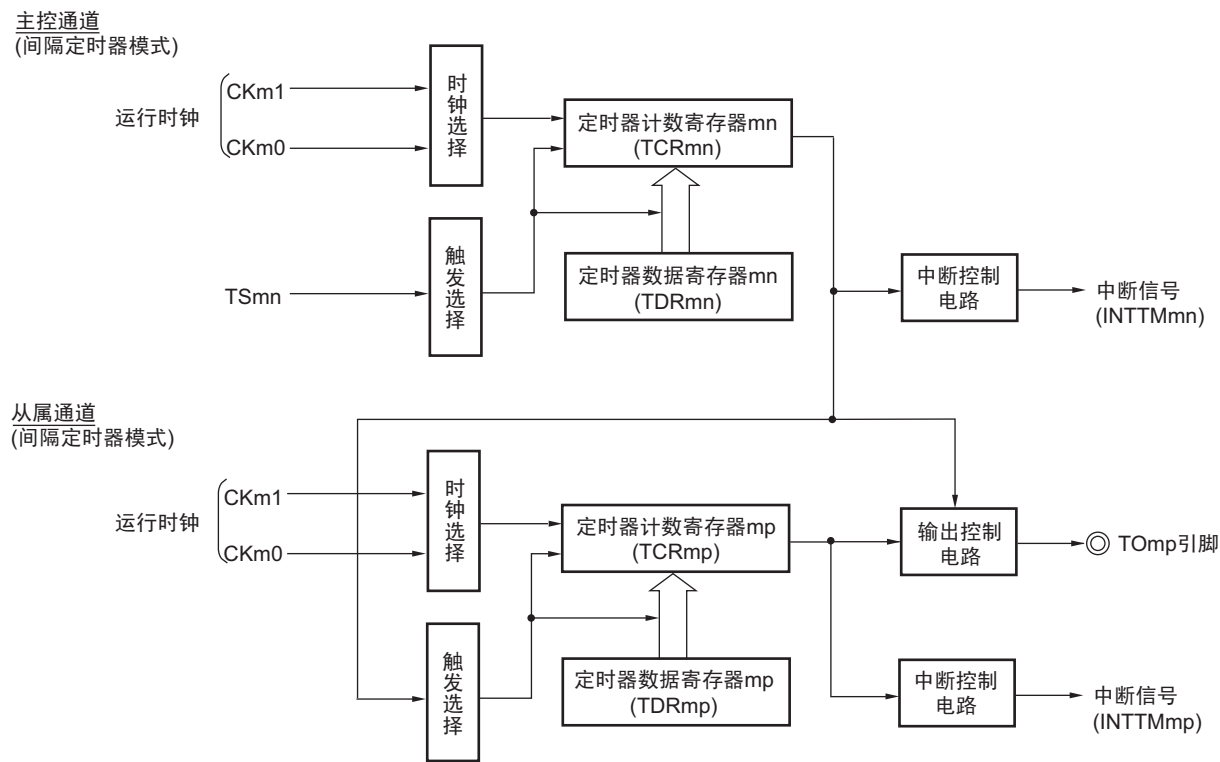
当用作 PWM 功能时，从属通道进行递减计数，在计数到“0000H”为止的期间为 PWM 输出 (TOmp) 的占空比。

在从主控通道产生 INTTMmn 中断并且经过 1 个时钟后，PWM 输出 (TOmp) 变为有效电平，并且在从属通道的 TCRmp 寄存器的值变为“0000H”时变为无效电平。

注意 要同时改写主控通道的定时器数据寄存器 mn (TDRmn) 和从属通道的 TDRmp 寄存器时，需要 2 次写存取。因为在主控通道产生 INTTMmn 时将 TDRmn 寄存器和 TDRmp 寄存器的值加载到 TCRmn 寄存器和 TCRmp 寄存器，所以如果分别在主控通道产生 INTTMmn 前和产生后进行改写，TOmp 引脚就不能输出期待的波形。因此，要同时改写主控的 TDRmn 寄存器和从属的 TDRmp 寄存器时，必须在主控通道产生 INTTMmn 后立即改写这 2 个寄存器。

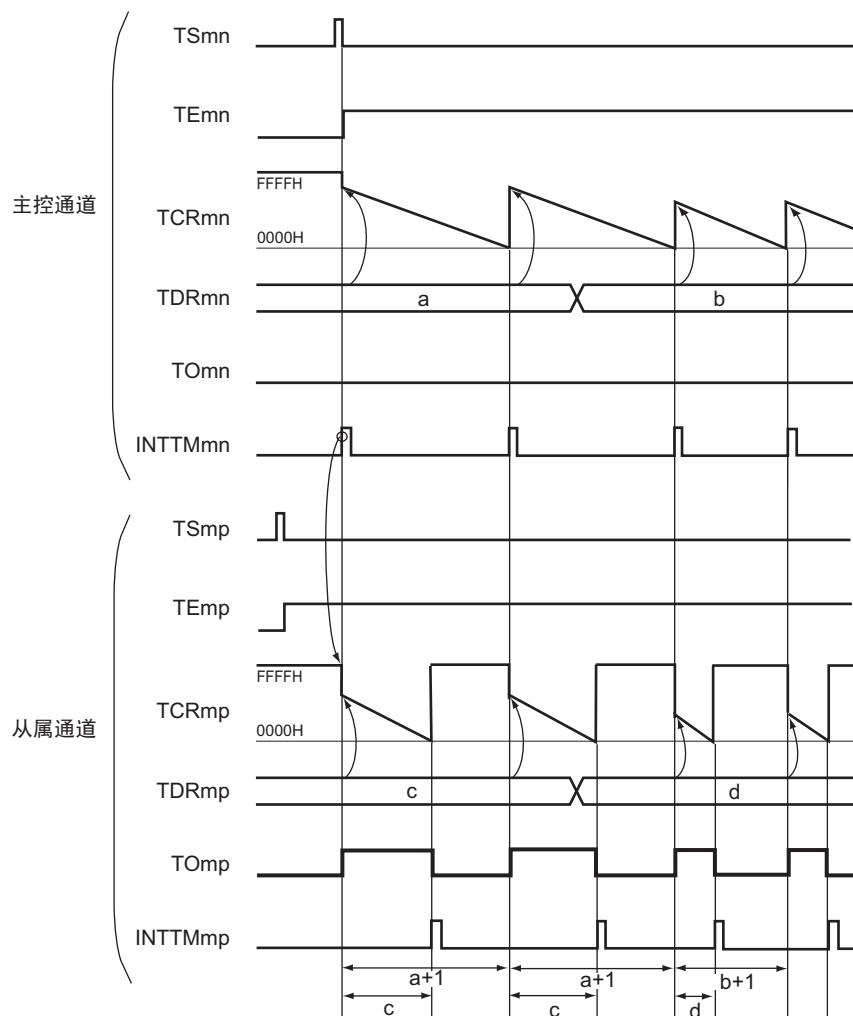
备注 m: 单元号 (m=0) n: 主控通道号 (n=0、2、4、6)
 p: 从属通道号 (n < p ≤ 7)

图 6-67 作为 PWM 功能运行的框图



备注 m: 单元号 (m = 0)、n: 通道号 (n = 0、2、4、6)
p: 从属通道号 (n < p ≤ 7)

图 6-68 作为 PWM 功能运行的基本时序例子

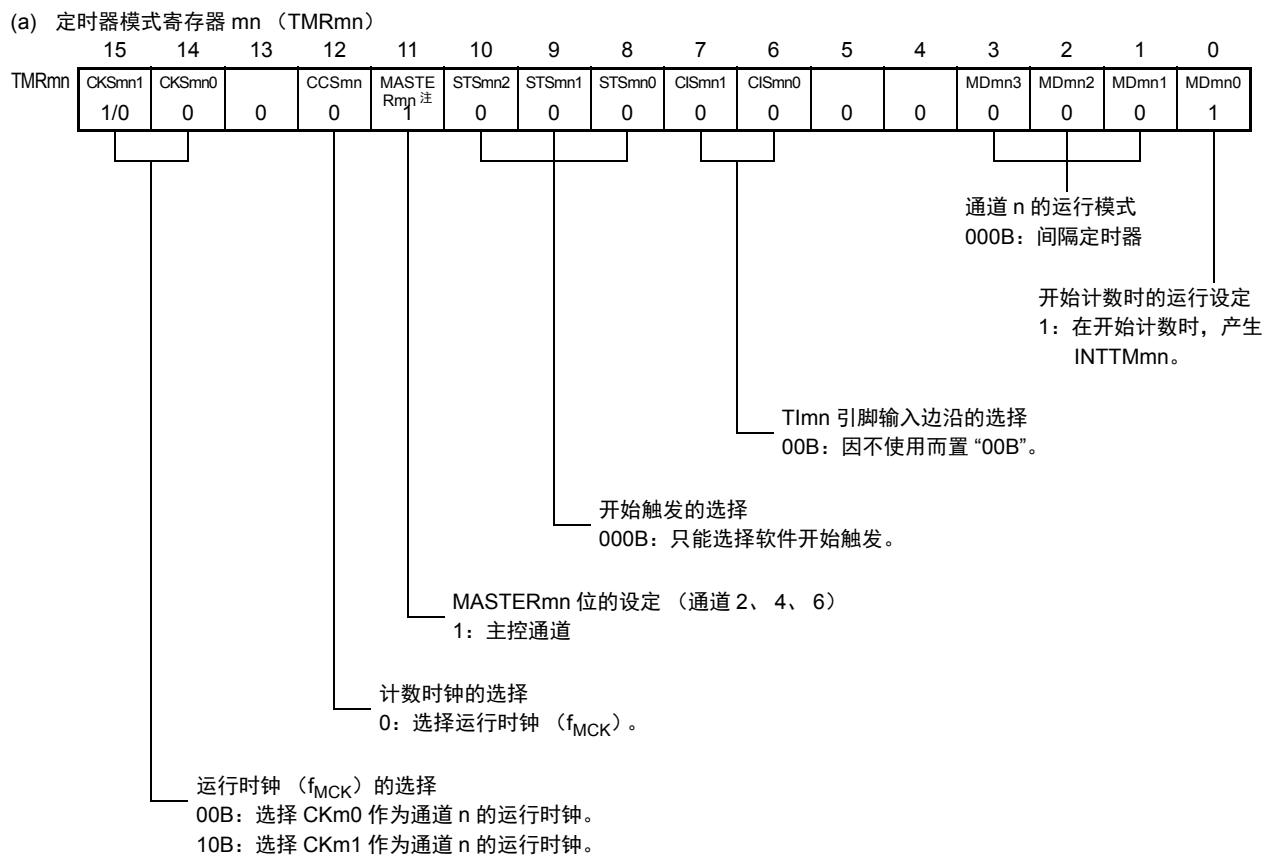


备注 1. *m*: 单元号 ($m = 0$)、*n*: 通道号 ($n = 0、2、4、6$)

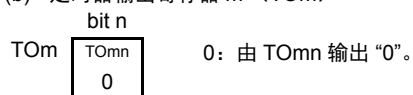
p: 从属通道号 ($n < p \leq 7$)

2. **TSmn、TSmp:** 定时器通道开始寄存器 *m* (TSm) 的 bit *n*、*p*
TEmn、TEmp: 定时器通道允许状态寄存器 *m* (TEm) 的 bit *n*、*p*
TCRmn、TCRmp: 定时器计数寄存器 *mn*、*mp* (TCRmn、TCRmp)
TDRmn、TDRmp: 定时器数据寄存器 *mn*、*mp* (TDRmn、TDRmp)
TOmn、TOmp: TOmn 和 TOmp 引脚输出信号

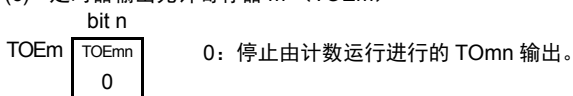
图 6-69 PWM 功能时（主控通道）的寄存器设定内容例子



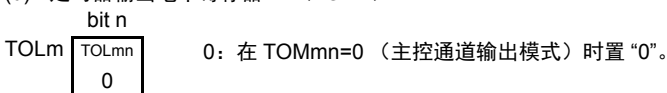
(b) 定时器输出寄存器 m (TOM)



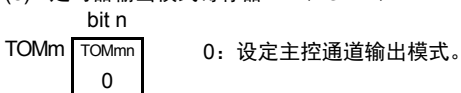
(c) 定时器输出允许寄存器 m (TOEm)



(d) 定时器输出电平寄存器 m (TOLm)



(e) 定时器输出模式寄存器 m (TOMm)

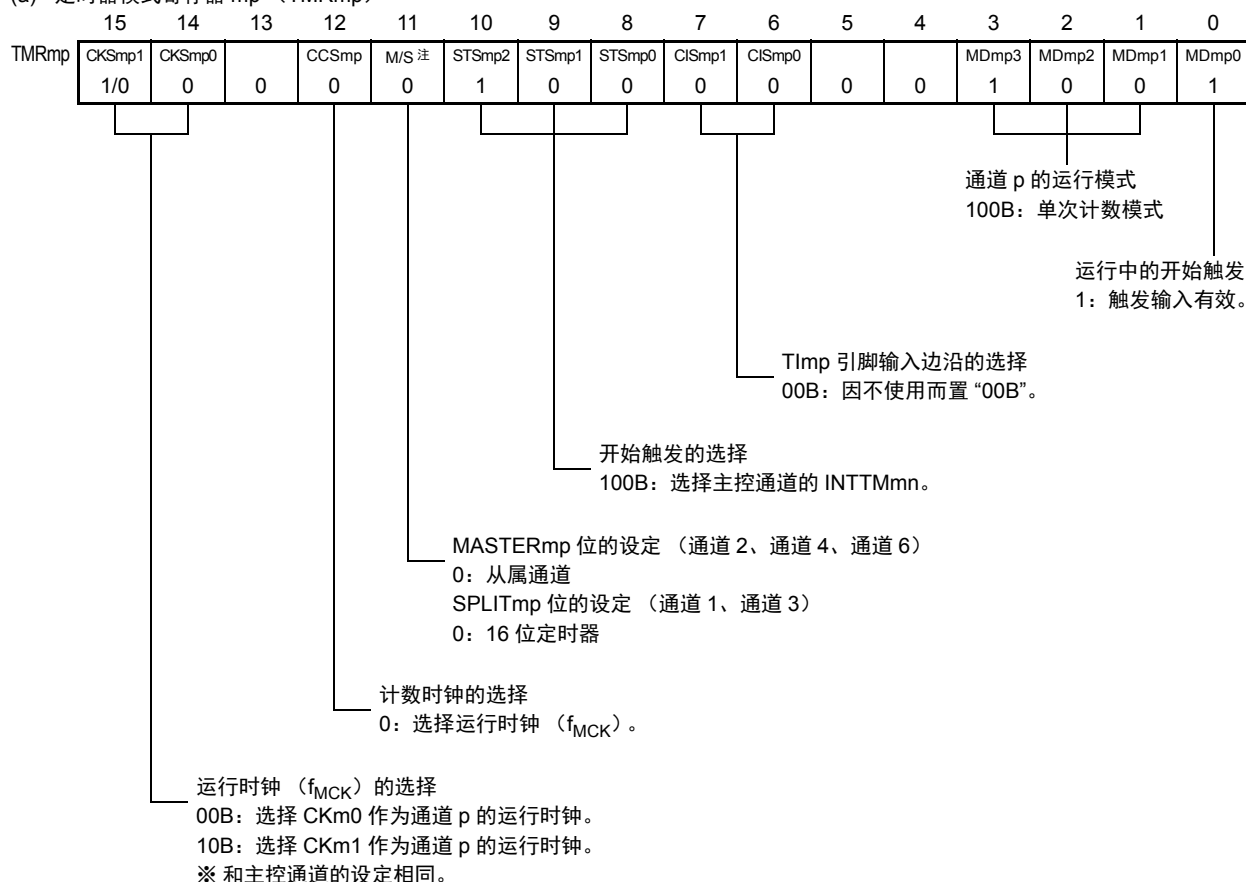


注 TMRm2、TMRm4、TMRm6: MASTERmn=1
TMRm0: 固定为“0”

备注 m: 单元号 (m=0) n: 主控通道号 (n=0、2、4、6)

图 6-70 PWM 功能时（从属通道）的寄存器设定内容例子

(a) 定时器模式寄存器 mp (TMRmp)



(b) 定时器输出寄存器 m (TOM)

bit p	
TOM	TOMP
	1/0
	0: 由 TOMP 输出“0”。
	1: 由 TOMP 输出“1”。

(c) 定时器输出允许寄存器 m (TOEm)

bit p	
TOEm	TOEmp
	1/0
	0: 停止由计数运行进行的 TOMP 输出。
	1: 允许由计数运行进行的 TOMP 输出。

(d) 定时器输出电平寄存器 m (TOLm)

bit p	
TOLm	TOLmp
	1/0
	0: 正逻辑输出（高电平有效）
	1: 负逻辑输出（低电平有效）

(e) 定时器输出模式寄存器 m (TOMm)

bit p	
TOMm	TOMmp
	1
	1: 设定从属通道输出模式。

注 TMRm2、TMRm4、TMRm6: MASTERmn 位
TMRm1、TMRm3: SPLITmp 位
TMRm5、TMRm7: 固定为“0”。

备注 m: 单元号 (m=0) n: 主控通道号 (n=0、2、4、6)
p: 从属通道号 (n < p ≤ 7)

图 6-71 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 (停止提供时钟, 不能写各寄存器。)
	将外围允许寄存器 0 (PER0) 的 TAUmEN 位置“1”。→	上电状态, 各通道处于运行停止状态。 (开始提供时钟, 能写各寄存器。)
	设定定时器时钟选择寄存器 m (TPSm)。 确定 CKm0 和 CKm1 的时钟频率。	
通道初 始设定	设定使用的 2 个通道的定时器模式寄存器 mn、mp (TMRmn、TMRmp) (确定通道的运行模式)。 给主控通道的定时器数据寄存器 mn (TDRmn) 设定间隔 (周期) 值, 并且给从属通道的 TDRmp 寄存器设定占空比的值。	通道处于运行停止状态。 (提供时钟, 消耗一部分电力。)
	从属通道的设定 将定时器输出模式寄存器 m (TOMm) 的 TOMmp 位置“1” (从属通道输出模式)。 设定 TOLmp 位。 设定 TOmp 位并且确定 TOmp 输出的初始电平。→	TOmp 引脚处于 Hi-Z 输出状态。
	将 TOEmp 位置“1”, 允许 TOmp 运行。→	当端口模式寄存器为输出模式并且端口寄存器为“0”时, 输出 TOmp 初始设定的电平。
	将端口寄存器和端口模式寄存器置“0”。→	因为通道处于运行停止状态, 所以 TOmp 不变。
		TOmp 引脚输出 TOmp 设定的电平。

图 6-71 PWM 功能时的操作步骤 (2/2)

	软件操作	硬件状态
重新开始运行	开始运行 将 TOEmp 位（从属）置“1”（只限于重新开始运行）。 将定时器通道开始寄存器 m（TSm）的 TSmn 位（主控）和 TSmp 位（从属）同时置“1”。——→ 因为 TSmn 位和 TSmp 位是触发位，所以自动返回到“0”。	TEmn 位和 TEmn 位都变为“1”。 主控通道开始计数并且产生 INTTMmn。以此为触发，从属通道也开始计数。
	运行中 禁止更改 TMRmn 寄存器和 TMRmp 寄存器以及 TOMmn 位、TOMmp 位、TOLmn 位和 TOLmp 位的设定值。 能在主控通道产生 INTTMmn 后更改 TDRmn 寄存器和 TDRmp 寄存器的设定值。 随时能读 TCRmn 寄存器和 TCRmp 寄存器。 不使用 TSRmn 寄存器和 TSRmp 寄存器。	主控通道将 TDRmn 寄存器的值加载到定时器计数寄存器 mn（TCRmn），并且进行递减计数。如果 TCRmn 计数到“0000H”，就产生 INTTMmn。同时，将 TDRmn 寄存器的值加载到 TCRmn 寄存器，并且重新开始递减计数。 从属通道以主控通道的 INTTMmn 为触发，将 TDRmp 寄存器的值加载到 TCRmp 寄存器，并且计数器进行递减计数。在从主控通道输出 INTTMmn 并且经过 1 个计数时钟后，将 TOmp 输出电平置为有效电平。然后，如果 TCRmp 计数到“0000H”，就在将 TOmp 的输出电平置为无效电平后停止计数。 此后，重复此运行。
	停止运行 将 TTmn 位（主控）和 TTmp 位（从属）同时置“1”。——→ 因为 TTmn 位和 TTmp 位是触发位，所以自动返回到“0”。	TEmn 位和 TEmn 位变为“0”并且停止计数。 TCRmn 寄存器和 TCRmp 寄存器保持计数值而停止计数。 TOmp 输出不被初始化而保持状态。
	将从属通道的 TOEmp 位置“0”并且给 TOmp 位设定值。——→	TOmp 引脚输出 TOmp 设定的电平。
TAU 停止	要保持 TOmp 引脚输出电平的情况： 在给端口寄存器设定要保留的值后将 TOmp 位置“0”。——→ 不需要保持 TOmp 引脚输出电平的情况： 不需要设定。	通过端口功能保持 TOmp 引脚的输出电平。
	将 PER0 寄存器的 TAUmEN 位置“0”。——→	断电状态 对全部电路和各通道的 SFR 进行初始化。 （TOmp 位变为“0”并且 TOmp 引脚变为端口功能）

备注 m: 单元号 (m=0) n: 主控通道号 (n=0、2、4、6)

p: 从属通道号 (n < p ≤ 7)

6.9.3 作为多重 PWM 输出功能的运行

这是通过扩展 PWM 功能并且使用多个从属通道进行不同占空比的多个 PWM 输出的功能。

例如，使用 2 个从属通道时，输出脉冲的周期和占空比能用以下计算式进行计算：

$$\begin{aligned} \text{脉冲周期} &= \{\text{TDRmn (主控) 的设定值} + 1\} \times \text{计数时钟周期} \\ \text{占空比 1 [\%]} &= \{\text{TDRmp (从属 1) 的设定值}\} / \{\text{TDRmn (主控) 的设定值} + 1\} \times 100 \\ \text{占空比 2 [\%]} &= \{\text{TDRmq (从属 2) 的设定值}\} / \{\text{TDRmn (主控) 的设定值} + 1\} \times 100 \end{aligned}$$

备注 当 TDRmp (从属 1) 的设定值 > {TDRmn (主控) 的设定值 + 1} 或者 {TDRmq (从属 2) 的设定值} > {TDRmn (主控) 的设定值 + 1} 时，占空比超过 100%，但是为 100% 输出。

在间隔定时器模式中，主控通道的定时器计数寄存器 mn (TCRmn) 运行并且对周期进行计数。

在单次计数模式中，从属通道 1 的 TCRmp 寄存器运行并且对占空比进行计数以及从 TOmp 引脚输出 PWM 波形。以主控通道的 INTTMmn 为开始触发，将定时器数据寄存器 mp (TDRmp) 的值加载到 TCRmp 寄存器并且开始递减计数。如果 TCRmp 变为“0000H”，就输出 INTTMmp，并且在输入下一个开始触发（主控通道的 INTTMmn）前停止计数。在从主控通道产生 INTTMmn 并且经过 1 个计数时钟后，TOmp 的输出电平变为有效电平，如果 TCRmp 变为“0000H”，就变为无效电平。

和从属通道 1 的 TCRmp 寄存器相同，在单次计数模式中，从属通道 2 的 TCRmq 寄存器运行并且对占空比进行计数以及从 TOmq 引脚输出 PWM 波形。以主控通道的 INTTMmn 为开始触发，将 TDRmq 寄存器的值加载到 TCRmq 寄存器并且开始递减计数。如果 TCRmq 变为“0000H”，就输出 INTTMmq，并且在输入下一个开始触发（主控通道的 INTTMmn）前停止计数。在从主控通道产生 INTTMmn 并且经过 1 个计数时钟后，TOmq 的输出电平变为有效电平，如果 TCRmq 变为“0000H”，就变为无效电平。

当通过如此的运行将通道 0 用作主控通道时，最多能同时输出 7 种 PWM 信号。

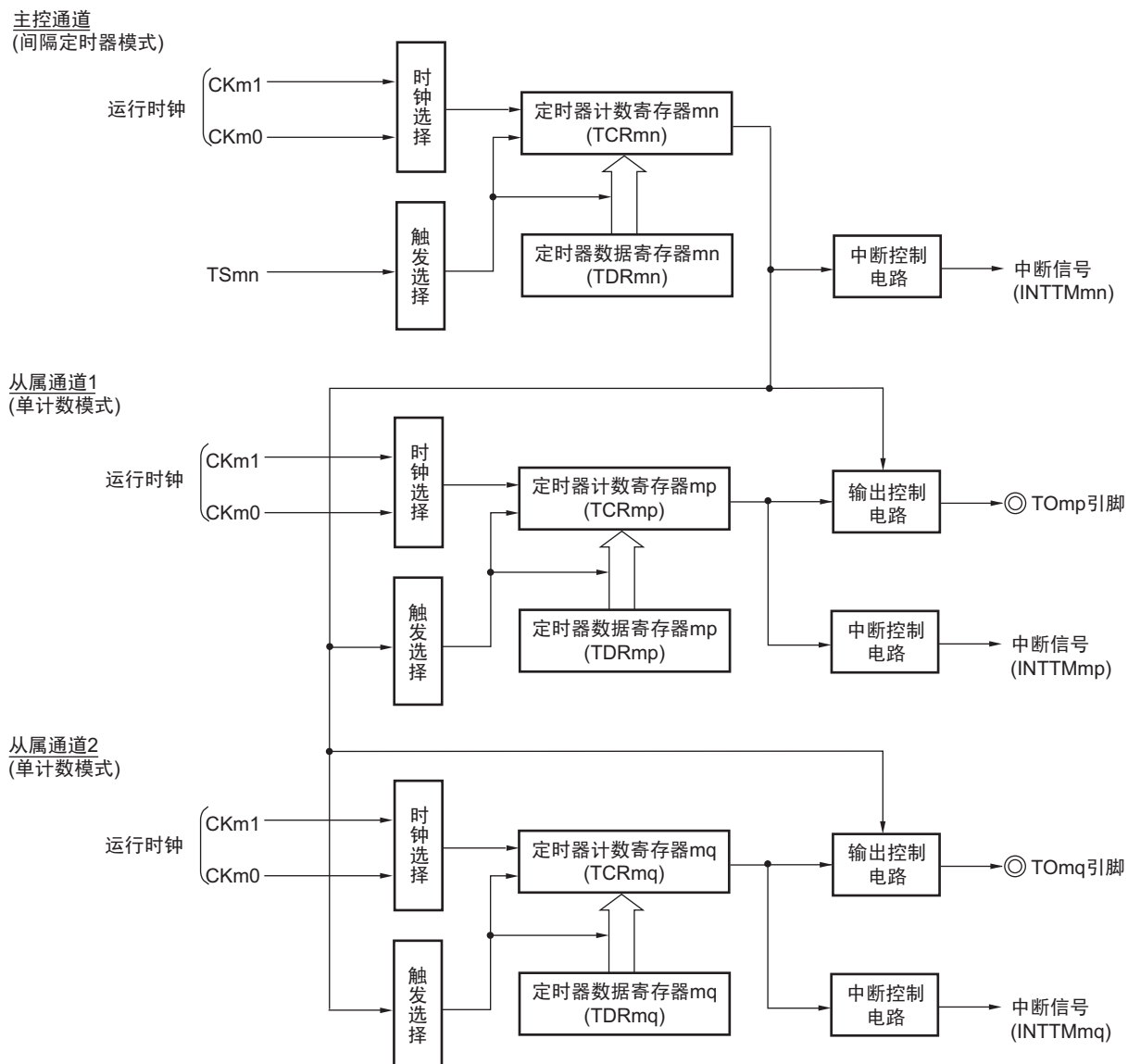
注意 要同时改写主控通道的定时器数据寄存器 mn (TDRmn) 和从属通道 1 的 TDRmp 寄存器时，至少需要 2 次写存取。因为在主控通道产生 INTTMmn 时将 TDRmn 寄存器和 TDRmp 寄存器的值加载到 TCRmn 寄存器和 TCRmp 寄存器，所以如果分别在主控通道产生 INTTMmn 前和产生后进行改写，TOmp 引脚就不能输出期待的波形。因此，要同时改写主控的 TDRmn 寄存器和从属的 TDRmp 寄存器时，必须在主控通道产生 INTTMmn 后立即改写这 2 个寄存器（同样也适用于从属通道 2 的 TDRmq 寄存器）。

备注 m: 单元号 (m=0) n: 主控通道号 (n=0、2、4)

p: 从属通道号 q: 从属通道号

$n < p < q \leq 7$ (p 和 q 是大于 n 的整数)

图 6-72 作为多重 PWM 输出功能运行的框图（输出 2 种 PWM 的情况）

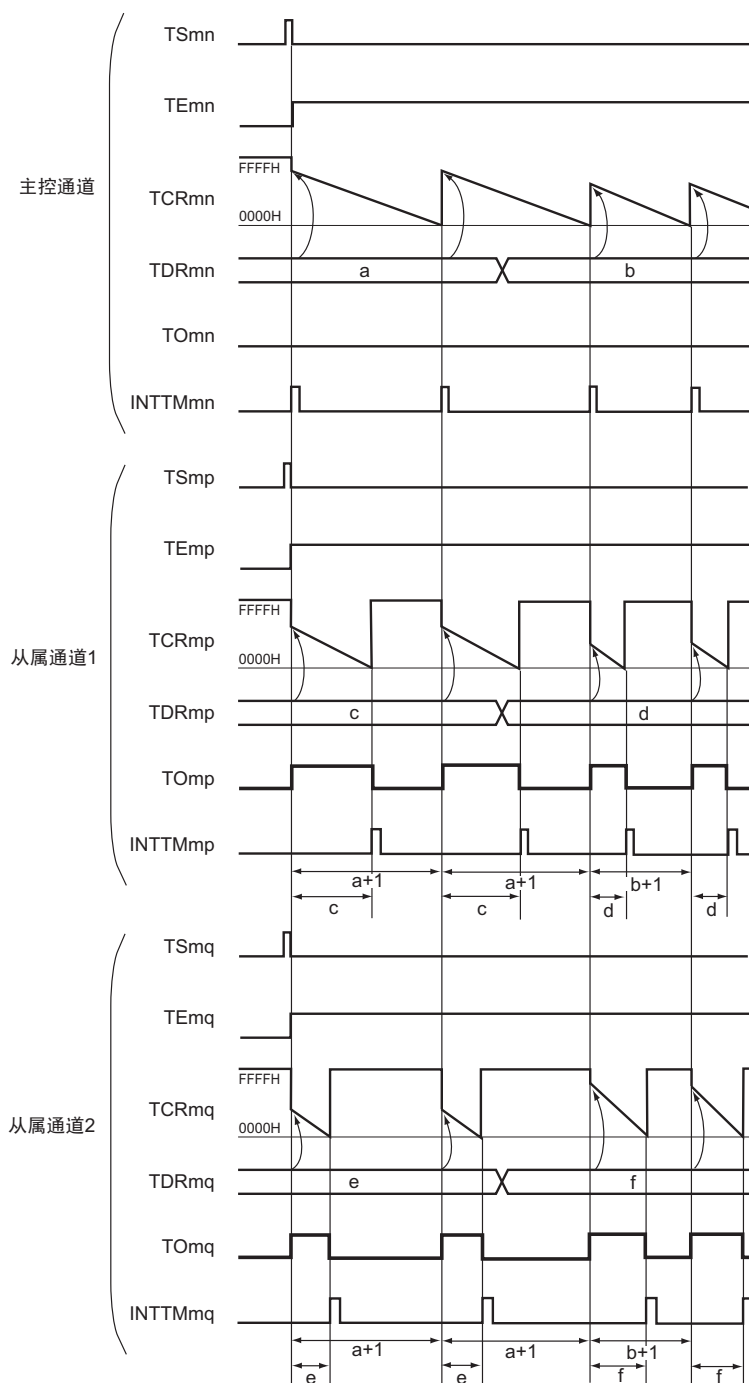


备注 m: 单元号 (m = 0)、n: 通道号 (n = 0、2、4)

p: 从属通道号、q: 从属通道号

$n < p < q \leq 7$ (p 和 q 为大于 n 的整数)

图 6-73 作为多重 PWM 输出功能的运行基本时序例子（输出 2 种 PWM 的情况）



备注 1. m: 单元号 (m = 0)、n: 通道号 (n = 0、2、4)

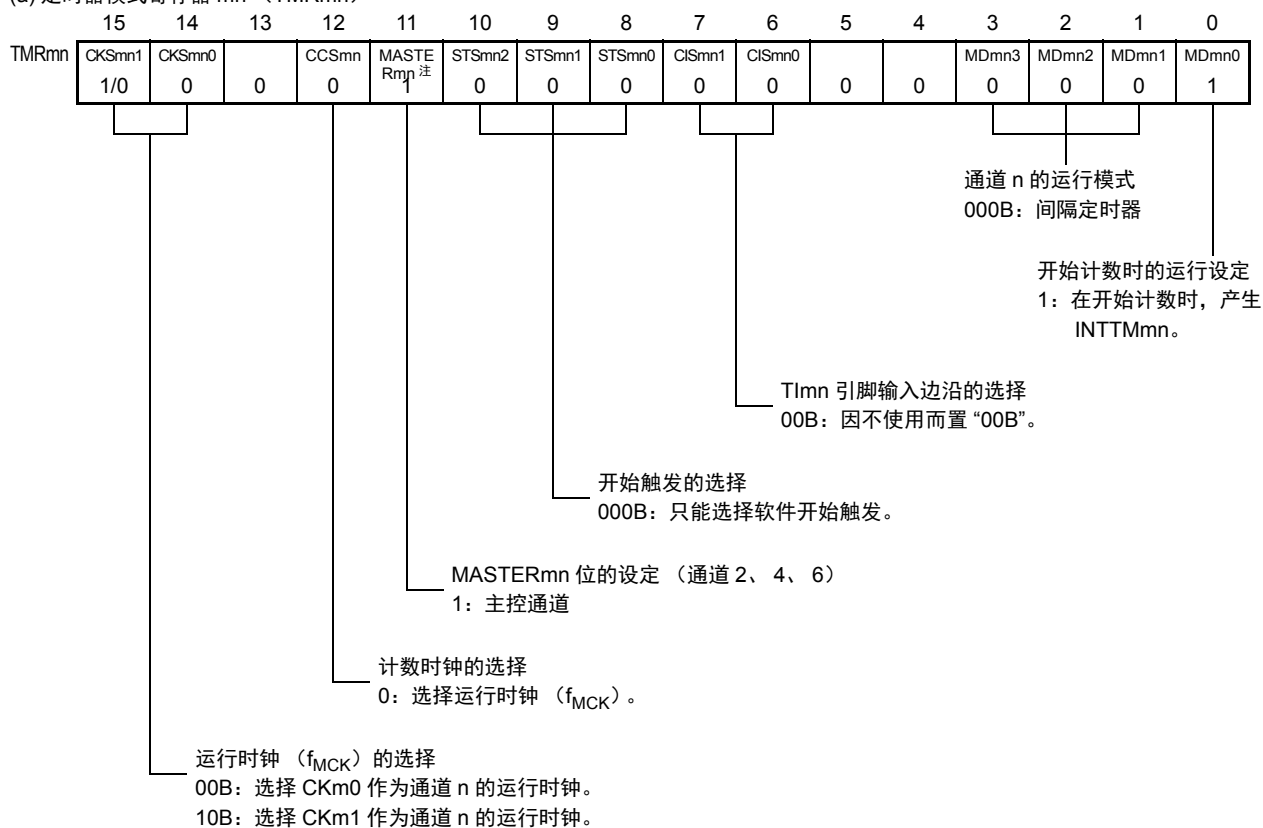
p: 从属通道号、q: 从属通道号

$n < p < q \leq 7$ (p 和 q 为大于 n 的整数)

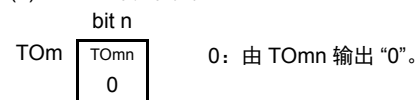
2. TS_{mn}、TS_{mp}、TS_{mq}: 定时器通道开始寄存器 m (TS_m) 的 bit n、p、q
 TE_{mn}、TE_{mp}、TE_{mq}: 定时器通道允许状态寄存器 m (TE_m) 的 bit n、p、q
 TCR_{mn}、TCR_{mp}、TCR_{mq}: 定时器计数寄存器 mn、mp、mq (TCR_{mn}、TCR_{mp}、TCR_{mq})
 TDR_{mn}、TDR_{mp}、TDR_{mq}: 定时器数据寄存器 mn、mp、mq (TDR_{mn}、TDR_{mp}、TDR_{mq})
 TO_{mn}、TO_{mp}、TO_{mq}: TO_{mn}、TO_{mp}、和 TO_{mq} 引脚输出信号

图 6-74 多重 PWM 输出功能时（主控通道）的寄存器设定内容例子

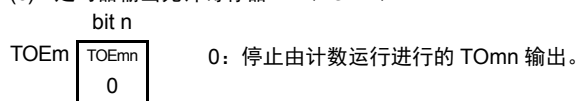
(a) 定时器模式寄存器 mn (TMRmn)



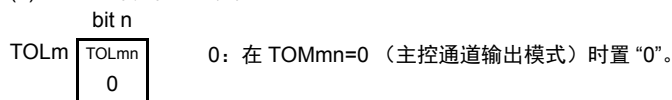
(b) 定时器输出寄存器 m (TOM)



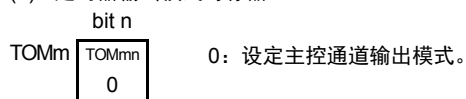
(c) 定时器输出允许寄存器 m (TOEm)



(d) 定时器输出电平寄存器 m (TOLm)



(e) 定时器输出模式寄存器 m (TOMm)

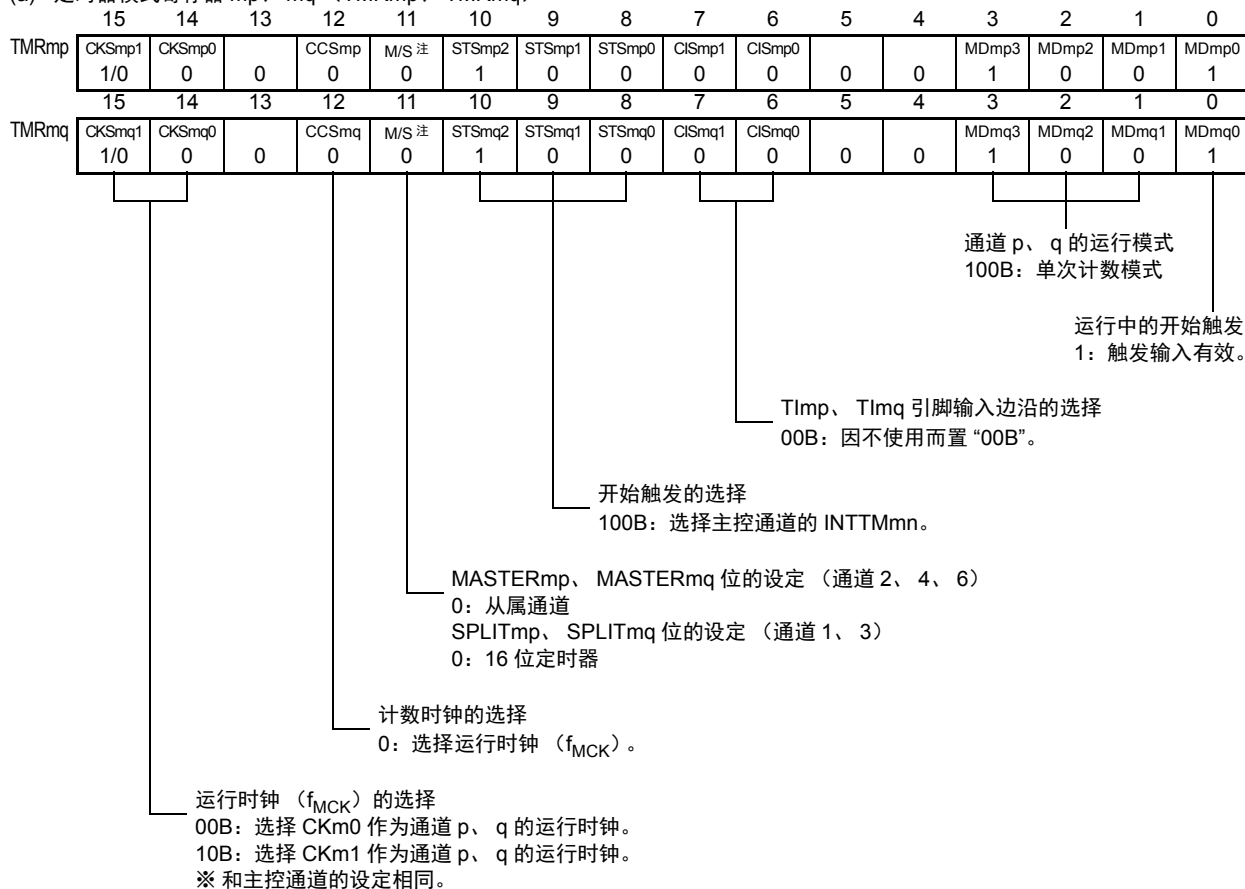


注 TMRm2、TMRm4、TMRm6: MASTERmn=1
TMRm0: 固定为“0”

备注 m: 单元号 (m=0) n: 主控通道号 (n=0、2、4)

图 6-75 多重 PWM 输出功能时（从属通道）的寄存器设定内容例子（输出 2 种 PWM 的情况）

(a) 定时器模式寄存器 mp、mq (TMRmp、TMRmq)



(b) 定时器输出寄存器 m (TOm)

	bit q	bit p	
TOm	TOmq 1/0	TOmp 1/0	0: 由 TOmp 和 TOmq 输出“0”。 1: 由 TOmp 和 TOmq 输出“1”。

(c) 定时器输出允许寄存器 m (TOEm)

	bit q	bit p	
TOEm	TOEmq 1/0	TOEmp 1/0	0: 停止由计数运行进行的 TOmp 和 TOmq 输出。 1: 允许由计数运行进行的 TOmp 和 TOmq 输出。

(d) 定时器输出电平寄存器 m (TOLm)

	bit q	bit p	
TOLm	TOLmq 1/0	TOLmp 1/0	0: 正逻辑输出（高电平有效） 1: 负逻辑输出（低电平有效）

(e) 定时器输出模式寄存器 m (TOMm)

	bit q	bit p	
TOMm	TOMmq 1	TOMmp 1	1: 设定从属通道输出模式。

注 TMRm2、TMRm4、TMRm6: MASTERmp 位、MASTERmq 位
TMRm1、TMRm3: SPLITmp 位、SPLITmq 位
TMRm5、TMRm7: 固定为“0”。

备注 m: 单元号 (m=0) n: 主控通道号 (n=0、2、4)
p: 从属通道号 q: 从属通道号
n < p < q ≤ 7 (p 和 q 是大于 n 的整数)

图 6-76 多重 PWM 输出功能时的操作步骤（输出 2 种 PWM 的情况）(1/2)

	软件操作	硬件状态
TAU 初始 设定		断电状态 (停止提供时钟, 不能写各寄存器。)
	将外围允许寄存器 0 (PER0) 的 TAUmEN 位置“1”。————→	上电状态, 各通道处于运行停止状态。 (开始提供时钟, 能写各寄存器。)
	设定定时器时钟选择寄存器 m (TPSm)。 确定 CKm0 和 CKm1 的时钟频率。	
通道初 始设定	设定使用的各通道的定时器模式寄存器 mn、mp、mq (TMRmn、TMRmp、TMRmq) (确定通道的运行模式)。 给主控通道的定时器数据寄存器 mn (TDRmn) 设定间隔 (周期) 值, 并且给从属通道的 TDRmp 寄存器和 TDRmq 寄存器设定占空比的值。	通道处于运行停止状态。 (提供时钟, 消耗一部分电力。)
	从属通道的设定 将定时器输出模式寄存器 m (TOMm) 的 TOMmp 位和 TOMmq 位置“1” (从属通道输出模式)。 设定 TOLmp 位和 TOLmq 位。	TOmp 引脚和 TOmq 引脚处于 Hi-Z 输出状态。
	设定 TOmp 位和 TOmq 位, 并且确定 TOmp 和 TOmq 输出的初始电平。————→	当端口模式寄存器为输出模式并且端口寄存器为“0”时, 输出 TOmp 和 TOmq 初始设定的电平。
	将 TOEmp 位和 TOEmq 位置“1”, 允许 TOmp 和 TOmq 的运行。————→	因为通道处于运行停止状态, 所以 TOmp 和 TOmq 不变。
	将端口寄存器和端口模式寄存器置“0”。————→	TOmp 引脚和 TOmq 引脚输出 TOmp 和 TOmq 设定的电平。

图 6-76 多重 PWM 输出功能时的操作步骤 (输出 2 种 PWM 的情况) (2/2)

	软件操作	硬件状态
重新开始运行	开始运行 (只在重新开始运行时将 TOEmp 位和 TOEmq 位 (从属) 置“1”)。 将定时器通道开始寄存器 m (TSm) 的 TSmn 位 (主控)、TSmp 位和 TSmq 位 (从属) 同时置“1”。 因为 TSmn 位、TSmp 位和 TSmq 位是触发位, 所以自动返回到“0”。	TEmn 位、TEmp 位和 TEMq 位变为“1”。 主控通道开始计数并且产生 INTTMmn。以此为触发, 从属通道也开始计数。
	运行中 禁止更改 TMRmn、TMRmp、TMRmq 寄存器以及 TOMmn 位、TOMmp 位、TOMmq 位、TOLmn 位、TOLmp、TOLmq 位的设定值。 能在主控通道产生 INTTMmn 后更改 TDRmn、TDRmp、TDRmq 寄存器的设定值。 随时能读 TCRmn、TCRmp、TCRmq 寄存器。 不使用 TSRmn、TSRmp、TSRmq 寄存器。	主控通道将 TDRmn 寄存器的值加载到定时器计数寄存器 mn (TCRmn), 并且进行递减计数。如果 TCRmn 计数到“0000H”, 就产生 INTTMmn。同时, 将 TDRmn 寄存器的值加载到 TCRmn 寄存器, 并且重新开始递减计数。 从属通道 1 以主控通道的 INTTMmn 信号为触发, 将 TDRmp 寄存器的值传送到 TCRmp 寄存器, 并且计数器开始递减计数。在从主控通道输出 INTTMmn 并且经过 1 个计数时钟后, 将 TOmp 输出电平置为有效电平。然后, 如果计数到“0000H”, 就在将 TOmp 的输出电平置为无效电平后停止计数。 从属通道 2 以主控通道的 INTTMmn 信号为触发, 将 TDRmq 寄存器的值传送到 TCRmq 寄存器, 并且计数器开始递减计数。在从主控通道输出 INTTMmn 并且经过 1 个计数时钟后, 将 TOMq 输出电平置为有效电平。然后, 如果计数到“0000H”, 就在将 TOMq 的输出电平置为无效电平后停止计数。 此后, 重复此运行。
	停止运行 将 TTmn 位 (主控)、TTmp 位和 TTmq 位 (从属) 位同时置“1”。 因为 TTmn 位、TTmp 位和 TTmq 位是触发位, 所以自动返回到“0”。 将从属通道的 TOEmp 位和 TOEmq 位置“0”并且给 TOmp 位和 TOMq 位设定值。	TEmn 位、TEmp 位和 TEMq 位都为“0”并且停止计数。 TCRmn、TCRmp、TCRmq 寄存器保持计数值而停止计数。 TOmp 和 TOMq 输出不被初始化而保持状态。 TOmp 引脚和 TOMq 引脚输出 TOmp 和 TOMq 设定的电平。
	TAU 停止 要保持 TOmp 引脚和 TOMq 引脚的输出电平的情况: 在给端口寄存器设定要保持的值后将 TOmp 位和 TOMq 位置“0”。 不需要保持 TOmp 引脚和 TOMq 引脚的输出电平的情况: 不需要设定。 将 PER0 寄存器的 TAUmEN 位置“0”。	通过端口功能保持 TOmp 引脚和 TOMq 引脚的输出电平。 断电状态 对全部电路和各通道的 SFR 进行初始化。 (TOmp 位和 TOMq 位变为“0”并且 TOmp 引脚和 TOMq 引脚变为端口功能)

备注 m: 单元号 (m=0) n: 主控通道号 (n=0、2、4)

p: 从属通道号 q: 从属通道号

 $n < p < q \leq 7$ (p 和 q 是大于 n 的整数)

6.10 使用定时器阵列单元时的注意事项

6.10.1 使用定时器输出时的注意事项

根据产品，分配了定时器输出功能的引脚也可能被分配其他复用功能的输出。在这种情况下使用定时器输出时，需要将其他复用功能输出设定为初始状态。。

(1) 使用 TO00 ~ TO07 输出时

除了将端口模式寄存器（PMxx 位）和端口寄存器（Pxx 位）置“0”外，还必须将 LCD 端口功能寄存器 0 ~ 6（PFSEG0 ~ PFSEG6）的对应位置“0”。详细内容请参照“表 4-17 使用复用功能时端口相关寄存器的设定（1/7）”。

(2) 使用分配给 P42 的 TO05 输出时（PIOR2=1 时）

为了将复用 SO10/TxD1 输出置“1”，除了将端口模式寄存器（PM42 位）和端口寄存器（P42 位）置“0”外，还必须将串行通道允许状态寄存器 0（SE0）、串行输出寄存器 0（SO0）和串行输出允许寄存器 0（SOE0）设定为与初始状态相同的值。

第 7 章 实时时钟 2

7.1 实时时钟 2 的功能

实时时钟 2 (RTC2) 具有以下功能。

- 具有年、月、星期、日、小时、分钟和秒的计数器，最长能计数到99年。
- 固定周期中断功能（周期：0.5秒、1秒、1分钟、1小时、1日、1个月）
- 闹钟中断功能（闹钟：星期、小时、分钟）
- 1Hz的引脚输出功能（通常1Hz输出、高精度1Hz输出）

实时时钟中断信号 (INTRTC) 能用作 STOP 模式的唤醒或者 A/D 转换器的 SNOOZE 模式的触发。

注意 1. 只有在选择副系统时钟 ($f_{\text{SUB}}=32.768\text{kHz}$) 作为实时时钟 2 的运行时钟的情况下，才能进行年、月、星期、日、小时、分钟和秒的计数。当选择低速内部振荡器时钟 ($f_{\text{IL}}=15\text{kHz}$) 时，只能使用固定周期中断功能。

选择 f_{IL} 时的固定周期中断间隔用以下计算式进行计算：

固定周期 (RTCC0 寄存器选择的值) $\times f_{\text{SUB}}/f_{\text{IL}}$ 。

2. 使用高精度 1Hz 的引脚输出时，必须给高速内部振荡器时钟 (f_{IH}) 选择 24 MHz。

7.2 实时时钟 2 的结构

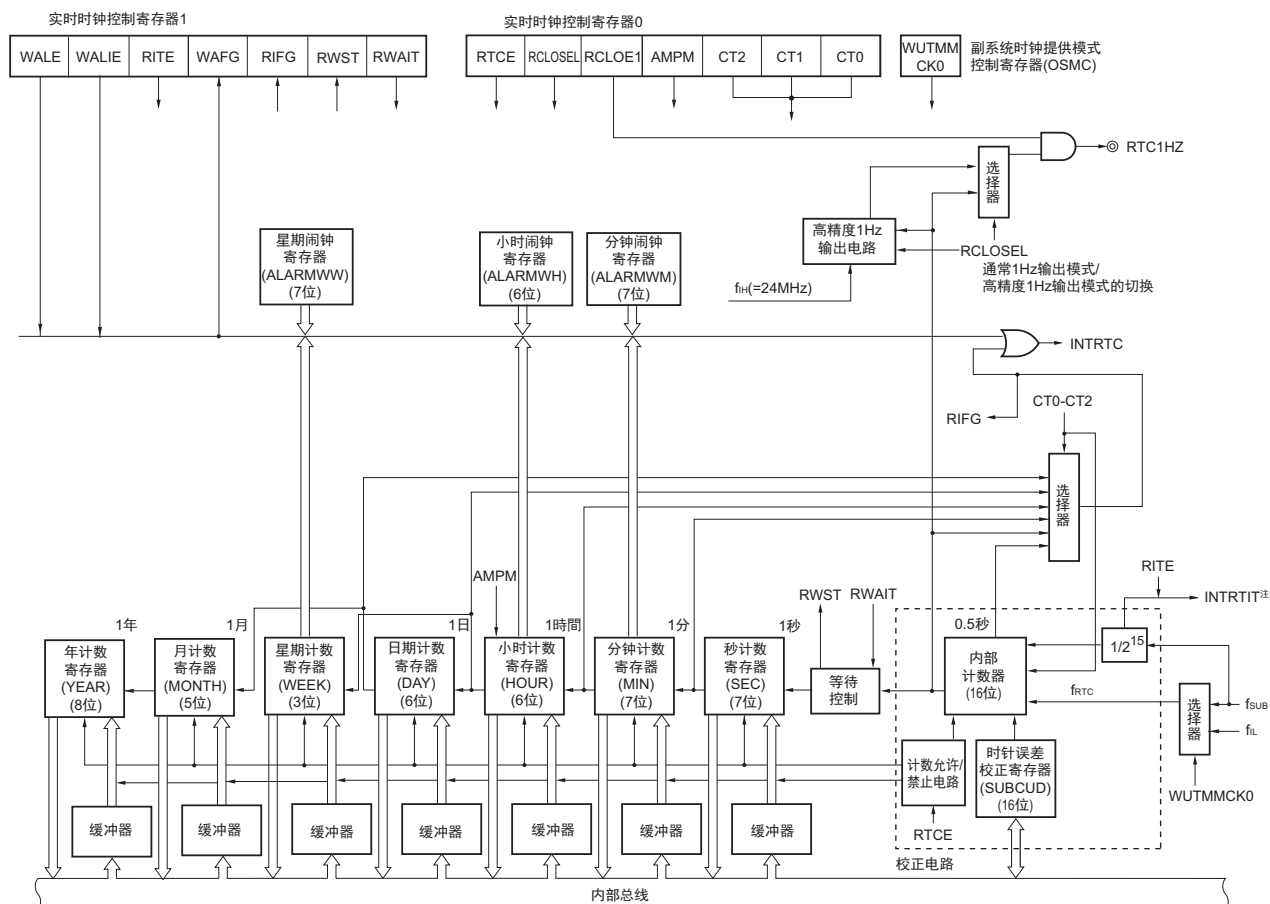
实时时钟 2 由以下硬件构成。

表 7-1 实时时钟 2 的结构

项目	结构
计数器	计数器（16 位）
控制寄存器	外围允许寄存器 0（PER0）
	外围允许寄存器 1（PER1）
	副系统时钟提供模式控制寄存器（OSMC）
	实时时钟控制寄存器 0（RTCC0）
	实时时钟控制寄存器 1（RTCC1）
	上电复位状态寄存器（PORSR）
	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）
	闹钟分钟寄存器（ALARMWM）
	闹钟小时寄存器（ALARMWH）
	闹钟星期寄存器（ALARMWW）

实时时钟 2 的框图如图 7-1 所示。

图 7-1 高精度实时时钟的框图



注 该中断表示从时钟误差校正寄存器（SUBCUD）取校正值的时序。以1秒为间隔（ f_{SUB} 基准）取校正值。

7.3 控制高精度实时时钟的寄存器

以下寄存器用于控制实时时钟。

- 外围允许寄存器0 (PER0)
- 外围允许寄存器1 (PER1)
- 副系统时钟提供模式控制寄存器 (OSMC)
- 上电复位状态寄存器 (PORSR)
- 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)
- 闹钟分钟寄存器 (ALARMWM)
- 闹钟小时寄存器 (ALARMWH)
- 闹钟星期寄存器 (ALARMWW)
- 端口模式寄存器3 (PM3)
- 端口寄存器3 (P3)

产生各复位源时寄存器的状态如下所示。

复位源	系统关联寄存器 注1	日历关联寄存器 注2
POR	复位	不复位
外部复位	保持	保持
WDT	保持	保持
TRAP	保持	保持
LVD	保持	保持
其他内部复位	保持	保持

注 1. RTCC0、RTCC1、SUBCUD

2. SEC、MIN、HOUR、DAY、WEEK、MONTH、YEAR、ALARMWM、ALARMWH、ALARMWW (计数器)

产生复位信号后，SEC、MIN、HOUR、WEEK、DAY、MONTH、YEAR、ALARMWM、ALARMWH 和 ALARMWW 寄存器不复位。因此，上电后必须对全部的寄存器进行初始设定。

7.3.1 外围允许寄存器 0（PER0）

PER0 寄存器是设定允许或者禁止给各外围硬件提供时钟的寄存器。通过停止给不使用的硬件提供时钟，降低功耗和噪声。

要使用实时时钟 2 时，必须将 bit7（RTCWEN）置“1”。

通过 1 位或者 8 位存储器操作指令设定 PER0 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 7-2 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	RTCWEN	0	ADCEN	IICA0EN	SAU1EN	SAU0EN	0	TAU0EN	

RTCWEN	控制实时时钟 2 的输入时钟的提供
0	停止提供输入时钟。 - 不能写实时时钟 2 使用的 SFR。 - 实时时钟 2 能运行。
1	提供输入时钟。 - 能读写实时时钟 2 使用的 SFR。 - 实时时钟 2 可能运行。

- 注意 1. 可以通过将外围允许寄存器 0（PER0）的 RTCWEN 置“1”读写时钟误差校正寄存器（SUBCUD）。
2. 如果要使用实时时钟 2，就必须首先在计数时钟（ f_{RTC} ）振荡稳定的状态下将 RTCWEN 位置“1”，然后设定以下的寄存器。当 RTCWEN 位为“0”时，忽视实时时钟 2 的控制寄存器的写操作，而且读取值为 RTCWEN=1 时的值（副系统时钟提供模式控制寄存器（OSMC）、上电复位状态寄存器（PORSR）、端口模式寄存器 3（PM3）和端口寄存器 3（P3）除外）。
- 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）
 - 时钟误差校正寄存器（SUBCUD）
 - 闹钟分钟寄存器（ALARMWM）
 - 闹钟小时寄存器（ALARMWH）
 - 闹钟星期寄存器（ALARMWW）
3. 必须将 bit1 和 bit6 置“0”。

7.3.2 外围允许寄存器 1（PER1）

PER1 寄存器是设定允许或者禁止给各外围硬件提供时钟的寄存器。通过停止给不使用的硬件提供时钟，降低功耗和噪声。

可通过将 bit6（FMCEN）置“1”，对控制实时时钟 2 的寄存器中的时钟误差校正寄存器（SUBCUD）进行设定。

通过 1 位或者 8 位存储器操作指令设定 PER1 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 7-3 外围允许寄存器 1（PER1）的格式

地址: F007AH	复位后: 00H	R/W						
符号	7	6	5	4	3	2	1	0
PER1	TMKAEN	FMCEN	CMPEEN	0	0	SMCI1EN	SMCI0EN	0

FMCEN	控制副系统时钟频率测量电路的输入时钟的提供
0	停止提供输入时钟。 - 不能写副系统时钟频率测量电路使用的 SFR。 - 不能写实时时钟 2 使用的 SUBCUD 寄存器。 - 副系统时钟频率测量电路处于复位状态。
1	提供输入时钟。 - 能读写入副系统时钟频率测量电路使用的 SFR。 - 能读写实时时钟 2 使用的 SUBCUD 寄存器。

注意 1. 可以通过将外围允许寄存器 0（PER0）的 RTCWEN 设定为“1”，或者将外围允许寄存器 1（PER1）的 FMCEN 位设定为“1”，读写时钟误差校正寄存器（SUBCUD）。

2. 必须将 bit0、3、4 设定为“0”。

7.3.3 副系统时钟提供模式控制寄存器（OSMC）

OSMC 寄存器通过停止不需要的时钟功能实现低功耗。

如果将 RTCLPC 位置“1”，就在 STOP 模式或者 CPU 以副系统时钟运行时的 HALT 模式中，停止向除实时时钟 2、12 位间隔定时器、时钟输出 / 蜂鸣器输出和 LCD 控制器 / 驱动器以外的外围功能提供时钟，因此可降低功耗。

另外，可通过 OSMC 寄存器选择实时时钟 2、12 位间隔定时器、时钟输出 / 蜂鸣器输出、LCD 控制器 / 驱动器和副系统时钟频率测量电路的运行时钟。

通过 8 位存储器操作指令设定 OSMC 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 7-4 副系统时钟提供模式控制寄存器（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	在 STOP 模式或者 CPU 以副系统时钟运行的 HALT 模式中的设定
0	允许向外围功能提供副系统时钟。 (有关允许运行的外围功能，请参照表 21-1 ~ 表 21-3。)
1	停止向除实时时钟 2、12 位间隔定时器、时钟输出 / 蜂鸣器输出和 LCD 控制器 / 驱动器以外的外围功能提供副系统时钟。

WUTMMCK0	选择实时时钟 2、12 位间隔定时器和 LCD 控制器 / 驱动器的运行时钟	选择时钟输出 / 蜂鸣器输出的 PCLBUZn 引脚的输出时钟	副系统时钟频率测量电路的运行
0	副系统时钟 (f_{SUB})	允许选择副系统时钟 (f_{SUB})	有效
1	低速内部振荡器时钟 (f_{IL})	禁止选择副系统时钟 (f_{SUB})	有效

注意 1. 在副系统时钟振荡时，必须选择副系统时钟（WUTMMCK0 位 = 0）。

2. 如果将 WUTMMCK0 设定为“1”，低速内部振荡器时钟就振荡。

3. 在 WUTMMCK0 设定为“1”时，只能使用实时时钟 2 的固定周期中断功能。而不能使用年、月、星期、日、小时、分钟、秒的计数功能以及 1Hz 输出功能。

固定周期中断的间隔可通过以下方式算出：固定周期（由 RTCC0 寄存器选择的值） $\times f_{SUB} / f_{IL}$

4. 只有在实时时钟 2、12 位间隔定时器和 LCD 控制器 / 驱动器的全部功能都停止运行时，才可通过 WUTMMCK0 位进行副系统时钟和低速内部振荡器时钟的切换。

7.3.4 上电复位状态控制寄存器（PORSR）

PORSR 寄存器是确认发生上电复位的寄存器。
给 PORSR 寄存器的 bit0（PORF）写“1”有效，写“0”无效。
确认是否发生上电复位时，必须事先给 PORF 位写“1”。
通过 8 位存储器操作指令设定 PORSR 寄存器。
在产生复位信号后，此寄存器的值变为“00H”。

- 注意 1. 只有在上电复位时初始化 PORSR 寄存器。在因其他复位源引起的复位时，保持原值。
2. 在 PORF 位为“1”时，保证没有发生上电复位。但是，不保证 RAM 保持原值。

图 7-5 上电复位状态寄存器（PORSR）的格式

地址: F00F9H	复位后: 00H	R/W						
符号	7	6	5	4	3	2	1	0
PORSR	0	0	0	0	0	0	0	PORF

PORF	确认是否发生上电复位
0	没有写“1”或者发生上电复位。
1	没有发生上电复位。

7.3.5 实时时钟控制寄存器 0（RTCC0）

这是设定实时时钟 2 的运行开始或者停止、RTC1HZ 引脚的控制、12/24 小时系统和固定周期中断功能的 8 位寄存器。

通过 1 位或者 8 位存储器操作指令设定 RTCC0 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 7-6 实时时钟控制寄存器 0（RTCC0）的格式（1/2）

地址：FFF9DH 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
RTCC0	RTCE	RCLOSEL	RCLOE1	0	AMPM	CT2	CT1	CT0

RTCE 注 1	控制实时时钟 2 的运行
0	停止计数器运行。
1	开始计数器运行。

RCLOSEL	控制 RTC1HZ 引脚的输出模式
0	通常 1Hz 输出
1	高精度 1Hz 输出

RCLOE1 注 2	RTC1HZ 引脚的输出控制
0	禁止 RTC1HZ 引脚的输出（1Hz）
1	允许 RTC1HZ 引脚的输出（1Hz）
因为在 RTCE=0 时时钟计数器不运行，所以不进行 1Hz 输出。	

- 注
1. 在将 RTCE 设定为“1”后转移到 STOP 模式时，必须按照“图 7-20”向 STOP 模式转移。

2. 如果在时钟计数器运行（RTCE = 1）时设定 RCLOE1 位，就有可能向 1 Hz 输出引脚（RTC1HZ）输出毛刺（干扰）。
- 注意
1. 只有在选择 f_{1H} 为 24MHz 时才能使用高精度 1Hz 输出。另外，在使用高精度 1Hz 输出时，必须使用时钟误差校正。

2. 必须将 bit4 设定为“0”。

图 7-6 实时时钟控制寄存器 0 (RTCC0) 的格式 (2/2)

地址: FFF9DH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
RTCC0	RTCE	RCLOSEL	RCLOE1	0	AMPM	CT2	CT1	CT0

AMPM	12 小时系统 /24 小时系统的选择
0	12 小时系统 (表示上午或者下午)
1	24 小时系统

要在时钟计数器运行 (RTCE=1) 中更改 AMPM 位的值时, 必须在将 RWAIT 位 (RTCC1 的 bit0) 置“1”后进行改写, 并且重新设定小时计数器 (HOUR)。
AMPM 位为“0”时是 12 小时系统, 为“1”时是 24 小时系统。时间位的表示如表 7-3 所示。

CT2	CT1	CT0	固定周期中断 (INTRTC) 的选择
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)
0	1	0	1 秒一次 (与秒累加同时)
0	1	1	1 分钟一次 (每分钟的 00 秒)
1	0	0	1 小时一次 (每小时的 00 分 00 秒)
1	0	1	1 日一次 (每日的 00 时 00 分 00 秒)
1	1	×	1 个月一次 (每月的 1 日上午 00 时 00 分 00 秒)

要在计数器运行中 (RTCE=1) 更改 CT2 ~ CT0 位的值时, 必须在通过中断屏蔽标志寄存器将 INTRTC 设定为禁止中断处理后进行改写, 并且必须在改写后清除 RIFG 标志和 RTCIF 标志, 然后再设定为允许中断处理。

注意 必须将 bit4 设定为“0”。

表 7-2 RTCE、RCLOSEL 和 RCLOE1 的设定值与状态的关系

寄存器的设定值			状态	
RTCE	RCLOSEL	RCLOE1	实时时钟 2 的状态	RTC1HZ 引脚输出
0	×	×	计数停止	不输出
1	0	0	计数	不输出
		1	计数	通常 1Hz 输出
	1	0	计数	不输出
		1	计数	高精度 1Hz 输出

备注 ×: 忽略

7.3.6 实时时钟控制寄存器 1（RTCC1）

RTCC1 寄存器是控制闹钟中断功能和计数器等待的 8 位寄存器。
通过 1 位或者 8 位存储器操作指令设定 RTCC1 寄存器。
在产生复位信号后，此寄存器的值变为“00H”。

图 7-7 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	RITE	WAFG	RIFG	0	RWST	RWAIT

WALE	闹钟的运行控制
0	一致运行无效。
1	一致运行有效。
要在计数器运行中（RTCE=1）并且 WALIE 位为“1”的情况下设定 WALE 位的值时，必须在通过中断屏蔽标志寄存器将 INTRTC 设定为禁止中断处理后进行改写，并且必须在改写后清除 WAFG 标志和 RTCIF 标志。要设定闹钟的各寄存器（RTCC1 寄存器的 WALIE 标志、闹钟分钟寄存器（ALARMWM）、闹钟小时寄存器（ALARMWH）和闹钟星期寄存器（ALARMWW））时，必须将 WALE 位置“0”（一致运行无效）。	

WALIE	闹钟中断（INTRTC）功能的运行控制
0	不产生闹钟一致中断。
1	产生闹钟一致中断。

RITE	校正时序信号中断（INTRTIT）功能的运行控制
0	不产生校正时序信号中断。
1	产生校正时序信号中断。

WAFG	闹钟检测状态标志
0	闹钟不一致。
1	检测到闹钟一致。
这是表示检测到闹钟一致的状态标志。只在 WALE 位为“1”时有效，在检测到闹钟一致并且经过 1 个时钟（32.768kHz）后变为“1”。 通过给此标志写“0”来清除此标志。写“1”的操作无效。	

注意 如果通过 1 位操作指令写 RTCCT1 寄存器，就可能清除 RIFG 标志和 WAFG 标志。因此，必须通过 8 位操作指令写 RTCCT1。为了在写时不清除 RIFG 标志和 WAFG 标志，必须将对应位置“1”（写操作无效）。在不使用 RIFG 标志和 WAFG 标志而且即使值被改写也没有问题的情况下，也能通过 1 位操作指令写 RTCCT1 寄存器。

图 7-7 实时时钟控制寄存器 1（RTCC1）的格式（2/2）

地址：FFF9EH 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
RTCC1	WALE	WALIE	RITE	WAFG	RIFG	0	RWST	RWAIT

RIFG	固定周期中断状态标志
0	没有产生固定周期中断。
1	产生固定周期中断。
这是表示产生固定周期中断的状态标志。当产生固定周期中断时，此标志为“1”。 通过给此标志写“0”来清除此标志。写“1”的操作无效。	

RWST	实时时钟 2 的等待状态标志
0	计数器运行。
1	正处于计数器值的读写模式。
这是表示 RWAIT 位的设定是否有效的状态标志。 必须在确认此标志为“1”后读写计数器的值。当正在进行计数器的写操作时，即使给 RWAIT 位设定“0”，RWST 位也不变为“0”。在写操作结束后，RWST 位变为“0”。	

RWAIT	实时时钟 2 的等待控制
0	设定计数器运行。
1	设定停止 SEC ~ YEAR 计数器，并且为计数器值的读写模式。
此位控制计数器的运行。 要读写计数器值时，必须给此位写“1”。 因为内部计数器（16 位）继续运行，所以必须在 1 秒内结束读写，然后返回“0”。 从将 RWAIT 位置“1”到能读写计数器值（RWST=1）为止，最多需要 1 个 f _{RTC} 时钟。 如果在 RWAIT 位为“1”时发生内部计数器（16 位）上溢，就保持发生上溢的状态，在 RWAIT 位变为“0”后进行递增计数。 但是，当写秒计数寄存器时，不保持发生上溢的状态。	

注意 如果通过 1 位操作指令写 RTCCT1 寄存器，就可能清除 RIFG 标志和 WAFG 标志。因此，必须通过 8 位操作指令写 RTCCT1。为了在写时不清除 RIFG 标志和 WAFG 标志，必须将对应位置“1”（写操作无效）。在不使用 RIFG 标志和 WAFG 标志而且即使值被改写也没有问题的情况下，也能通过 1 位操作指令写 RTCCT1 寄存器。

备注 1. 固定周期中断和闹钟一致中断使用相同中断源（INTRTC）。在同时使用这 2 个中断的情况下，能在发生 INTRTC 时通过确认固定周期中断状态标志（RIFG）和闹钟检测状态标志（WAFG）来判断发生的是哪个中断。
2. 如果写秒计数寄存器（SEC），就清除内部计数器（16 位）。

7.3.7 秒计数寄存器（SEC）

这是取 0 ~ 59（十进制）的值表示秒计数值的 8 位寄存器。

通过内部计数器（16 位）的上溢进行递增计数的十进制计数器。

在写时，数据先被写到缓冲器，在经过最多 2 个 f_{RTC} 时钟后被写到计数器。以 BCD 码设定十进制的 00 ~ 59。

通过 8 位存储器操作指令设定 SEC 寄存器。

在产生复位信号后，此寄存器的值不被初始化。

图 7-8 秒计数寄存器（SEC）的格式

地址：FFF92H 复位后：不定 R/W

符号	7	6	5	4	3	2	1	0
SEC	0	SEC40	SEC20	SEC10	SEC8	SEC4	SEC2	SEC1

备注 如果写秒计数寄存器（SEC），就清除内部计数器（16 位）。

注意 要在时钟计数器运行中（RTCE=1）读写此寄存器时，必须按照“7.4.3 读实时时钟 2 的计数器”、“7.4.4 写实时时钟 2 的计数器”的步骤进行。

7.3.8 分钟计数寄存器（MIN）

这是取 0 ~ 59（十进制）的值表示分钟计数值的 8 位寄存器。

通过秒计数器的上溢进行递增计数的十进制计数器。

在写时，数据先被写到缓冲器，在经过最多 2 个 f_{RTC} 时钟后被写到计数器。在写操作过程中忽视秒计数寄存器的上溢并且设定为写入值。以 BCD 码设定十进制的 00 ~ 59。

通过 8 位存储器操作指令设定 MIN 寄存器。

在产生复位信号后，此寄存器的值不被初始化。

图 7-9 分钟计数寄存器（MIN）的格式

地址：FFF93H 复位后：不定 R/W

符号	7	6	5	4	3	2	1	0
MIN	0	MIN40	MIN20	MIN10	MIN8	MIN4	MIN2	MIN1

注意 要在时钟计数器运行中（RTCE=1）读写此寄存器时，必须按照“7.4.3 读实时时钟 2 的计数器”、“7.4.4 写实时时钟 2 的计数器”的步骤进行。

7.3.9 小时计数寄存器 (HOUR)

这是取 00 ~ 23 或者 01 ~ 12、21 ~ 32（十进制）的值表示小时计数值的 8 位寄存器。

通过分钟计数器的上溢进行递增计数的十进制计数器。

在写时，数据先被写到缓冲器，在经过最多 2 个 f_{RTC} 时钟后被写到计数器。在写操作过程中忽视分钟计数寄存器的上溢并且设定为写入值。

根据实时时钟控制寄存器 0 (RTCC0) 的 bit3 (AMPM) 设定的时间系统，以 BCD 码设定十进制的 00 ~ 23 或者 01 ~ 12、21 ~ 32。

如果更改 AMPM 位的值，HOUR 寄存器的值就变为设定的时间系统所对应的值。

通过 8 位存储器操作指令设定 HOUR 寄存器。

在产生复位信号后，此寄存器的值不被初始化。

图 7-10 小时计数寄存器 (HOUR) 的格式

地址: FFF94H 复位后: 不定 R/W

符号	7	6	5	4	3	2	1	0
HOUR	0	0	HOUR20	HOUR10	HOUR8	HOUR4	HOUR2	HOUR1

注意 1. 当选择 AMPM 位为“0”（12 小时系统）时，HOUR 寄存器的 bit5 (HOUR20) 表示 AM (0) /PM (1)。

2. 要在时钟计数器运行中 (RTCE=1) 读写此寄存器时，必须按照“7.4.3 读实时时钟 2 的计数器”、“7.4.4 写实时时钟 2 的计数器”的步骤进行。

AMPM 位的设定值以及小时计数寄存器（HOUR）值与时间的关系如表 7-3 所示。

表 7-3 时间位的显示

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	AM12 时	12H
1	01H	AM1 时	01H
2	02H	AM2 时	02H
3	03H	AM3 时	03H
4	04H	AM4 时	04H
5	05H	AM5 时	05H
6	06H	AM6 时	06H
7	07H	AM7 时	07H
8	08H	AM8 时	08H
9	09H	AM9 时	09H
10	10H	AM10 时	10H
11	11H	AM11 时	11H
12	12H	PM12 时	32H
13	13H	PM1 时	21H
14	14H	PM2 时	22H
15	15H	PM3 时	23H
16	16H	PM4 时	24H
17	17H	PM5 时	25H
18	18H	PM6 时	26H
19	19H	PM7 时	27H
20	20H	PM8 时	28H
21	21H	PM9 时	29H
22	22H	PM10 时	30H
23	23H	PM11 时	31H

当 AMPM 位为“0”时，HOUR 寄存器的值为 12 小时显示；当 AMPM 位为“1”时，HOUR 寄存器的值为 24 小时显示。

在 12 小时表示时，HOUR 寄存器的 bit5 显示上午 / 下午。上午（AM）为“0”，下午（PM）为“1”。

7.3.10 日计数寄存器 (DAY)

这是取 1 ~ 31 (十进制) 的值表示日计数值的 8 位寄存器。

通过小时计数器的上溢进行递增计数的十进制计数器。

计数器进行以下的计数。

[DAY 的计数值]

- 01 ~ 31 (1、3、5、7、8、10、12月)
- 01 ~ 30 (4、6、9、11月)
- 01 ~ 29 (2月 闰年)
- 01 ~ 28 (2月 平年)

在写时, 数据先被写到缓冲器, 在经过最多 2 个 f_{RTC} 时钟后被写到计数器。在写操作过程中忽视小时计数寄存器的上溢并且设定为写入值。以 BCD 码设定十进制的 01 ~ 31。

通过 8 位存储器操作指令设定 DAY 寄存器。

在产生复位信号后, 此寄存器的值不被初始化。

图 7-11 日计数寄存器 (DAY) 的格式

地址: FFF96H 复位后: 不定 R/W

符号	7	6	5	4	3	2	1	0
DAY	0	0	DAY20	DAY10	DAY8	DAY4	DAY2	DAY1

注意 要在时钟计数器运行中 (RTCE=1) 读写此寄存器时, 必须按照“7.4.3 读实时时钟 2 的计数器”、“7.4.4 写实时时钟 2 的计数器”的步骤进行。

7.3.11 星期计数寄存器（WEEK）

这是取 0 ～ 6 （十进制）的值表示星期计数值的 8 位寄存器。

在向日计数器进位时进行递增计数的十进制计数器。

在写时，数据先被写到缓冲器，在经过最多2个f_{RTC}时钟后被写到计数器。以BCD码设定十进制的00～06。

通过 8 位存储器操作指令设定 WEEK 寄存器。

在产生复位信号后，此寄存器的值不被初始化。

图 7-12 星期计数寄存器（WEEK）的格式

地址：FFF95H	复位后：不定	R/W								
符号	7	6	5	4	3	2	1	0		
WEEK	0	0	0	0	0	WEEK4	WEEK2	WEEK1		

注意 1. 月计数寄存器（MONTH）和日计数寄存器（DAY）的对应值不自动保存到星期寄存器（WEEK）。

必须在解除复位后进行以下的设定：

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. 要在时钟计数器运行中（RTCE=1）读写此寄存器时，必须按照“7.4.3 读实时时钟 2 的计数器”、“7.4.4 写实时时钟 2 的计数器”的步骤进行。

7.3.12 月计数寄存器 (MONTH)

这是取 1 ~ 12 (十进制) 的值表示月计数值的 8 位寄存器。

通过日计数器的上溢进行递增计数的十进制计数器。

在写时, 数据先被写到缓冲器, 在经过最多 2 个 f_{RTC} 时钟后被写到计数器。在写操作过程中忽视日计数寄存器的上溢并且设定为写入值。以 BCD 码设定十进制的 01 ~ 12。

通过 8 位存储器操作指令设定 MONTH 寄存器。

在产生复位信号后, 此寄存器的值不被初始化。

图 7-13 月计数寄存器 (MONTH) 的格式

地址: FFF97H 复位后: 不定 R/W

符号	7	6	5	4	3	2	1	0
MONTH	0	0	0	MONTH10	MONTH8	MONTH4	MONTH2	MONTH1

注意 要在时钟计数器运行中 (RTCE=1) 读写此寄存器时, 必须按照“7.4.3 读实时时钟 2 的计数器”、“7.4.4 写实时时钟 2 的计数器”的步骤进行。

7.3.13 年计数寄存器 (YEAR)

这是取 0 ~ 99 (十进制) 的值表示年计数值的 8 位寄存器。

通过月计数器 (MONTH) 的上溢进行递增计数的十进制计数器。

00、04、08、.....、92、96 是闰年。

在写时, 数据先被写到缓冲器, 在经过最多 2 个 f_{RTC} 时钟后被写到计数器。在写操作过程中忽视 MONTH 寄存器的上溢并且设定为写入值。以 BCD 码设定十进制的 00 ~ 99。

通过 8 位存储器操作指令设定 YEAR 寄存器。

在产生复位信号后, 此寄存器的值不被初始化。

图 7-14 年计数寄存器 (YEAR) 的格式

地址: FFF98H 复位后: 不定 R/W

符号	7	6	5	4	3	2	1	0
YEAR	YEAR80	YEAR40	YEAR20	YEAR10	YEAR8	YEAR4	YEAR2	YEAR1

注意 要在时钟计数器运行中 (RTCE=1) 读写此寄存器时, 必须按照“7.4.3 读实时时钟 2 的计数器”、“7.4.4 写实时时钟 2 的计数器”的步骤进行。

7.3.14 时钟误差校正寄存器（SUBCUD）

SUBCUD 寄存器通过每秒校正计数器的值，可以以最小分辨率 0.96 ppm 精度校正变快或者变慢的时钟。
SUBCUD 的 F8 ~ F0 是 9 位固定小数点形式（2 的补数）的寄存器。详细内容请参照“表 7-5 时钟误差校正值”。

通过 16 位存储器操作指令设定 SUBCUD 寄存器。

在发生因上电复位电路引起的内部复位后，此寄存器的值变为“0020H”。

图 7-15 时钟误差校正寄存器（SUBCUD）的格式

地址: F0310H	复位后: 0020H	R/W														
符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SUBCUD	F15	0	0	0	0	0	0	F8	F7	F6	F5	F4	F3	F2	F1	F0

F15	允许时钟误差校正
0	停止时钟误差校正。
1	允许时钟误差校正。

可通过时钟误差校正寄存器（SUBCUD）进行的晶体谐振器的振荡频率偏差的校正范围如表 7-4 所示。

表 7-4 晶体谐振器的振荡频率偏差的可校正范围

项目	值
可校正范围	-274.6ppm ~ +212.6ppm
最大量化误差	±0.48 ppm
最小分辨率	0.96 ppm

表 7-5 时钟误差校正值

SUBCUD										目标校正值
F15	F8	F7	F6	F5	F4	F3	F2	F1	F0	
1	1	0	0	0	0	0	0	0	0	-274.6 ppm
	1	0	0	0	0	0	0	0	1	-273.7 ppm
	1	0	0	0	0	0	0	1	0	-272.7 ppm
	•	•	•	•	•	•	•	•	•	•
	•	•	•	•	•	•	•	•	•	•
	•	•	•	•	•	•	•	•	•	•
	1	1	1	1	1	1	1	0	1	-33.3 ppm
	1	1	1	1	1	1	1	1	0	-32.4 ppm
	1	1	1	1	1	1	1	1	1	-31.4 ppm
	0	0	0	0	0	0	0	0	0	-30.5 ppm
	0	0	0	0	0	0	0	0	1	-29.6 ppm
	0	0	0	0	0	0	0	1	0	-28.6 ppm
	•	•	•	•	•	•	•	•	•	•
	•	•	•	•	•	•	•	•	•	•
	•	•	•	•	•	•	•	•	•	•
	0	0	0	0	1	1	1	1	1	-0.95 ppm
	0	0	0	1	0	0	0	0	0	0 ppm
	0	0	0	1	0	0	0	0	1	0.95 ppm
	•	•	•	•	•	•	•	•	•	•
	•	•	•	•	•	•	•	•	•	•
	•	•	•	•	•	•	•	•	•	•
	0	1	1	1	1	1	1	0	1	210.7 ppm
	0	1	1	1	1	1	1	1	0	211.7 ppm
	0	1	1	1	1	1	1	1	1	212.6 ppm
0	×	×	×	×	×	×	×	×	×	停止时钟误差校正

使用目标修正值以以下计算式计算 SUBCUD 寄存器的 F8 ~ F0 值。

$$\text{SUBCUD}[8:0] = \left(\frac{\text{目标校正值}[\text{ppm}] \times 2^{15}}{10^6} \right) \text{ 2 的补数 (9 位固定小数点形式) } + 0001.00000\text{B}$$

注意 目标校正值是晶体谐振器的振荡频率偏差（单位为 [ppm]）。有关校正值的计算方法，请参照“7.4.8 实时时钟 2 的时钟误差校正例子”。

例 1. 当目标校正值 = 18.3[ppm] 时

$$\begin{aligned} \text{SUBCUD}[8:0] &= (8.3 \times 2^{15} / 10^6) \text{ 2 的补数 (9 位固定小数点形式) } + 0001.00000\text{B} \\ &= (0.59375) \text{ 2 的补数 (9 位固定小数点形式) } + 0001.00000\text{B} \\ &= 0000.10011\text{B} + 0001.00000\text{B} \\ &= 0001.10011\text{B} \end{aligned}$$

例 2. 当目标校正值 = -18.3[ppm] 时

$$\begin{aligned} \text{SUBCUD}[8:0] &= (-18.3 \times 2^{15} / 10^6) \text{ 2 的补数 (9 位固定小数点形式) } + 0001.00000\text{B} \\ &= (-0.59965) \text{ 2 的补数 (9 位固定小数点形式) } + 0001.00000\text{B} \\ &= 1111.01101\text{B} + 0001.00000\text{B} \\ &= 0000.01101\text{B} \end{aligned}$$

7.3.15 闹钟分钟寄存器 (ALARMWM)

这是设定闹钟分钟的寄存器。

通过 8 位存储器操作指令设定 ALARMWM 寄存器。

在产生复位信号后，此寄存器的值不被初始化。

图 7-16 闹钟分钟寄存器 (ALARMWM) 的格式

地址: FFF9AH 复位后: 不定 R/W

符号	7	6	5	4	3	2	1	0
ALARMWM	0	WM40	WM20	WM10	WM8	WM4	WM2	WM1

注意 以 BCD 码设定十进制的 00 ~ 59。如果设定范围以外的值，就不检测闹钟。

7.3.16 闹钟小时寄存器 (ALARMWH)

这是设定闹钟小时的寄存器。

通过 8 位存储器操作指令设定 ALARMWH 寄存器。

在产生复位信号后，此寄存器的值不被初始化。

图 7-17 闹钟小时寄存器 (ALARMWH) 的格式

地址: FFF9BH 复位后: 不定 R/W

符号	7	6	5	4	3	2	1	0
ALARMWH	0	0	WH20	WH10	WH8	WH4	WH2	WH1

注意 1. 以 BCD 码设定十进制的 00 ~ 23 或者 01 ~ 12、21 ~ 32。如果设定范围以外的值，就不检测闹钟。

2. 当选择 AMPM 位为“0”（12 小时系统）时，ALARMWH 寄存器的 bit5（WH20）表示 AM（0）/PM（1）。

7.3.17 闹钟星期寄存器 (ALARMWW)

这是设定闹钟星期的寄存器。

通过 8 位存储器操作指令设定 ALARMWW 寄存器。

在产生复位信号后，此寄存器的值不被初始化。

图 7-18 闹钟星期寄存器 (ALARMWW) 的格式

地址: FFF9CH 复位后: 不定 R/W

符号	7	6	5	4	3	2	1	0
ALARMWW	0	WW6	WW5	WW4	WW3	WW2	WW1	WW0

闹钟时间的设定例子如表 7-6 所示。

表 7-6 闹钟时间的设定例子

闹钟设定时间	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								
	W W 0	W W 1	W W 2	W W 3	W W 4	W W 5	W W 6	10 时	1 时	10 分	1 分	10 时	1 时	10 分	1 分
每天 上午 0 时 00 分	1	1	1	1	1	1	1	1	2	0	0	0	0	0	0
每天 上午 1 时 30 分	1	1	1	1	1	1	1	0	1	3	0	0	1	3	0
每天 上午 11 时 59 分	1	1	1	1	1	1	1	1	1	5	9	1	1	5	9
星期一~星期五 下午 0 时 00 分	0	1	1	1	1	1	0	3	2	0	0	1	2	0	0
星期日 下午 1 时 30 分	1	0	0	0	0	0	0	2	1	3	0	1	3	3	0
星期一、星期三、星期五 下午 11 时 59 分	0	1	0	1	0	1	0	3	1	5	9	2	3	5	9

7.3.18 控制实时时钟 2 输出引脚的端口功能的寄存器

在使用实时时钟 2 时，必须设定控制对象通道的复用端口功能的寄存器（LCD 端口功能寄存器 0 ~ 6（PFSEG0 ~ PFSEG6）、端口模式寄存器（PMxx）、端口寄存器（Pxx））。详细内容请参照“4.3.9 LCD 端口功能寄存器 0 ~ 6（PFSEG0 ~ PFSEG6）”、“4.3.1 端口模式寄存器（PMxx）”和“4.3.2 端口寄存器（Pxx）”。

将实时时钟 2 输出引脚的复用端口（P31/RTC1HZ）用作实时时钟 2 时，必须将与端口对应的 LCD 端口功能寄存器（PFSEGx）、端口模式寄存器（PMxx）以及端口寄存器（Pxx）的位设定为“0”。

例) 将 P31/RTC1HZ 用作实时时钟 2 输出时

将 LCD 端口功能寄存器（PFSEG2）的 PFSEG21 位置“0”。

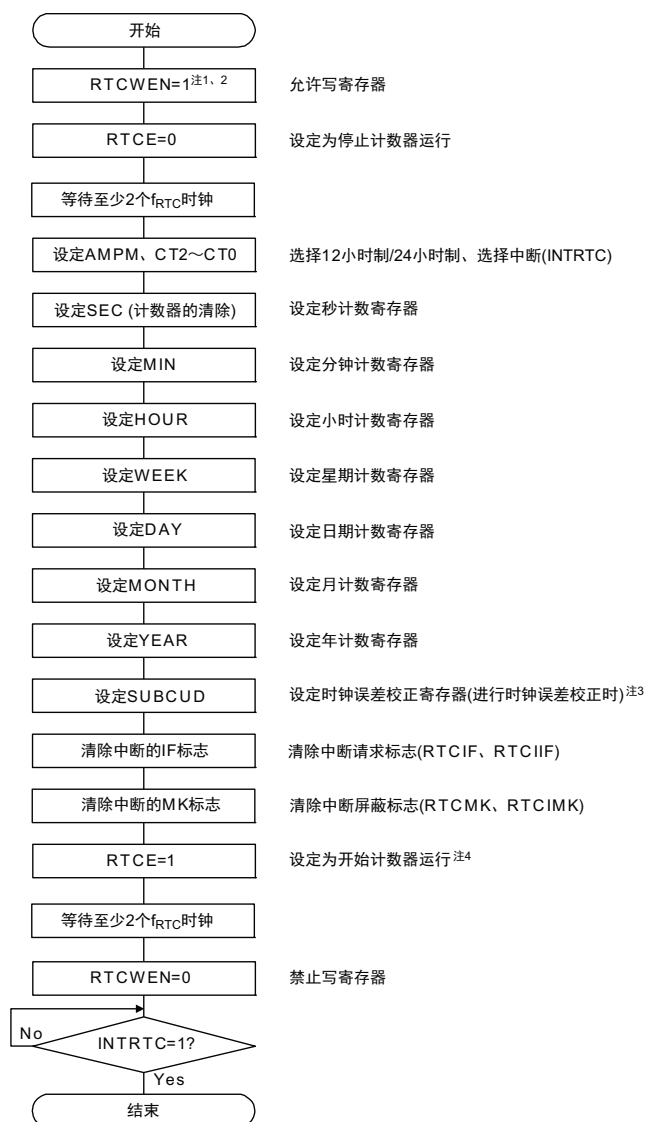
将端口模式寄存器（PM3）的 PM31 位置“0”。

将端口寄存器（P3）的 P31 位置“0”。

7.4 实时时钟2的运行

7.4.1 开始实时时钟2的运行

图 7-19 实时时钟2的运行开始步骤



- 注 1. 除存取 RTC 寄存器时以外，必须将 RTCWEN 位设定为“0”，以防止误写时钟计数器。
2. 在计数时钟（ f_{RTC} ）处于振荡稳定状态时，首先要将 RTCWEN 位设定为“1”。
3. 这是只需要校正时钟误差的情况。有关校正值的计算方法，请参照“7.4.8 实时时钟2的时钟误差校正例子”。
4. 如果在 RTCE=1 后，不等到 INTRTC 变为“1”就转移到 HALT/STOP 模式时，必须确认“7.4.2 开始运行后向 HALT/STOP 模式的转移”。

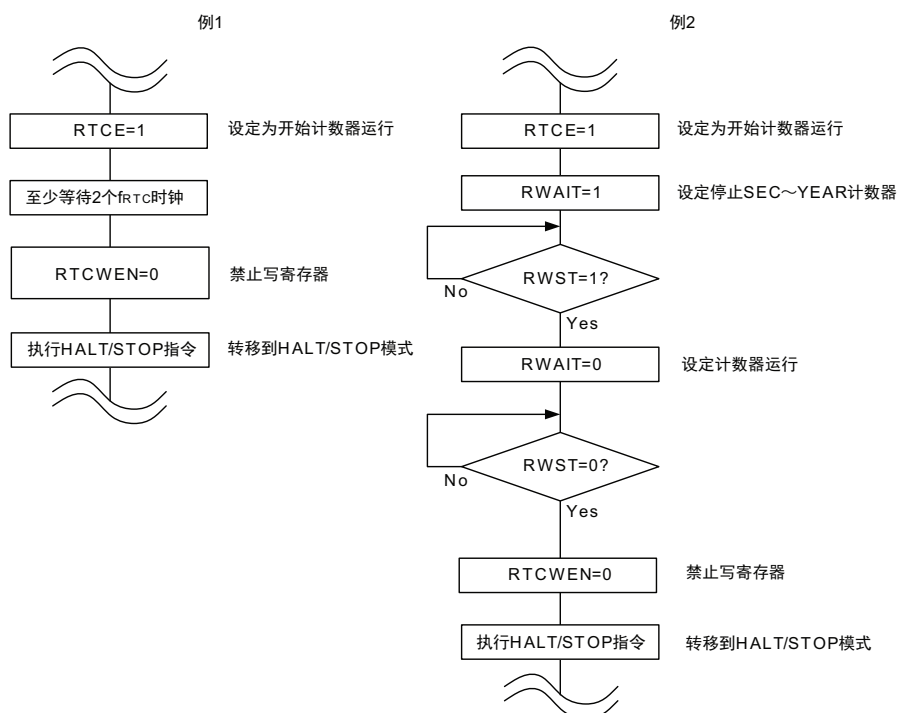
7.4.2 开始运行后向 HALT/STOP 模式的转移

如果要在将 RTCE 位置“1”后立即转移到 HALT/STOP 模式，就必须进行以下某种处理。

但是，在将 RTCE 位置“1”后，如果要在发生第 1 个 INTRTC 中断后转移到 HALT/STOP 模式，就不需要这些处理。

- 在将 RTCE 位置“1”后至少经过 2 个计数时钟 (f_{RTC}) 之后转移到 HALT/STOP 模式（参照图 7-20 的例 1）。
- 在将 RTCE 位置“1”后将 RWAIT 位置“1”，通过轮询确认 RWST 位变为“1”。然后，将 RWAIT 位置“0”并且再次通过轮询确认 RWST 位变为“0”，然后转移到 HALT/STOP 模式（参照图 7-20 的例 2）。

图 7-20 将 RTCE 位设定为“1”后向 HALT/STOP 模式转移的步骤



7.4.3 读实时时钟 2 的计数器

必须先将 RWAIT 位置“1”，然后读计数器。

必须在读计数器后将 RWAIT 位置“0”。

图 7-21 实时时钟 2 的计数器读操作步骤



注 1. 在计数器停止（RTCE=0）时，RWST 位不变为“1”。

2. 必须在转移到 STOP 模式前，确认 RWST 位为“0”。

注意 必须在 1 秒内进行将 RWAIT 位从“1”置“0”的处理。

备注 不限制秒计数寄存器（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）的读操作顺序。
可以不读全部寄存器而只读部分寄存器。

7.4.4 写实时时钟 2 的计数器

必须先将 RWAIT 位置“1”，然后写计数器。

必须在写计数器后将 RWAIT 位置“0”。

图 7-22 实时时钟 2 的计数器写操作步骤



注 1. 在计数器停止（RTCE=0）时，RWST 位不变为“1”。

2. 必须在转移到 STOP 模式前，确认 RWST 位为“0”。

注意 1. 必须在 1 秒内进行将 RWAIT 位从“1”置“0”的处理。

2. 如果要在计数器运行期间（RTCE = 1）改写 SEC、MIN、HOUR、WEEK、DAY、MONTH 和 YEAR 寄存器，必须通过中断屏蔽标志寄存器将 INTRTC 设定为禁止中断处理后，再进行改写。另外，进行改写后，必须清除 WAFG、RIFG 和 RTCIF 标志。

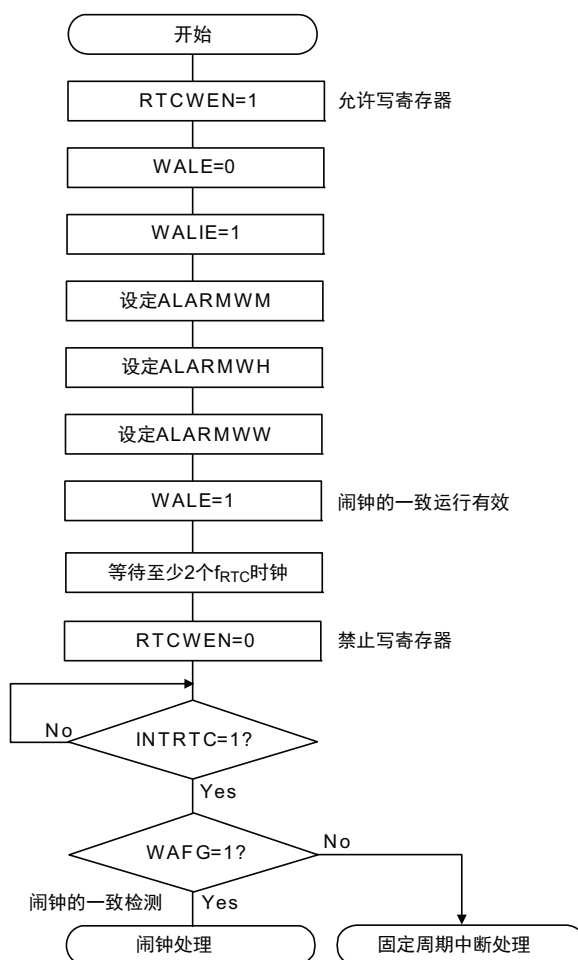
备注 不限制秒计数寄存器（SEC）、分钟计数寄存器（MIN）、小时计数寄存器（HOUR）、星期计数寄存器（WEEK）、日计数寄存器（DAY）、月计数寄存器（MONTH）和年计数寄存器（YEAR）的写操作顺序。

可以不设定全部寄存器而只改写部分寄存器。

7.4.5 实时时钟2的闹钟设定

必须先将 WALE 位置“0”（闹钟运行无效），然后设定闹钟时间。

图 7-23 闹钟设定步骤



备注 1. 不限制闹钟分钟寄存器（ALARMWM）、闹钟小时寄存器（ALARMWH）和闹钟星期寄存器（ALARMWW）的写操作顺序。

2. 固定周期中断和闹钟一致中断使用相同中断源（INTRTC）。在同时使用这 2 个中断的情况下，能在发生 INTRTC 时通过确认固定周期中断状态标志（RIFG）和闹钟检测状态标志（WAFG）来判断发生的是哪个中断。

7.4.6 实时时钟 2 的 1Hz 输出

图 7-24 1Hz 输出的设定步骤



注意 使用高精度 1Hz 引脚输出时，必须给高速内部振荡器时钟 (f_{IH}) 选择 24 MHz。

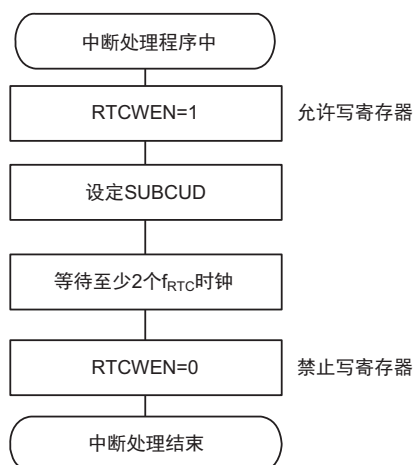
7.4.7 时钟误差校正寄存器的设定步骤

设定时钟误差校正寄存器（SUBCUD）时，必须执行以下任意一种处理。

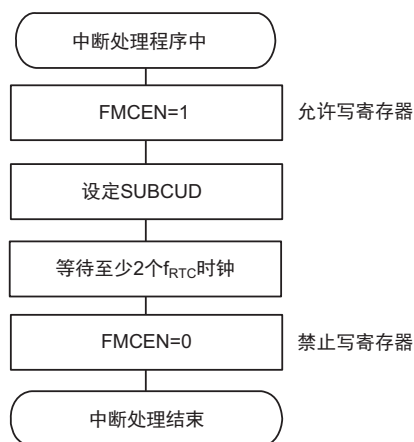
为了防止误写时钟寄存器，在进行 SUBCUD 寄存器的改写时，推荐 (2) 通过 FMCEN 允许写操作。

当时钟误差校正寄存器（SUBCUD）的改写与校正时序发生竞争时，有可能无法正常进行 RTC 的校正。为了避免校正时序与改写 SUBCUD 寄存器发生竞争，必须以与校正时序同步产生的校正时序中断（INTRTIT）或者固定周期中断（INTRTC）为起点，并且在产生下一个校正时序前（约 0.5 秒以内）完成改写 SUBCUD 寄存器。

- (1) 在设定 RTCWEN=1 后设定时钟误差校正寄存器，之后再设定 RTCWEN=0。



- (2) 在设定 FMCEN=1 后设定时钟误差校正寄存器，之后再设定 FMCEN=0。



7.4.8 实时时钟 2 的时钟误差校正例子

通过给时钟误差校正寄存器（SUBCUD）设定值，就能每秒以最小分辨率 0.96ppm 精度对变快或者变慢的时钟进行校正。

目标校正值的计算方法和使用目标校正值计算时钟误差校正寄存器的 F8 ~ F0 的值的方法如下所示。

目标校正值的计算方法 1

（使用 RTC1HZ 引脚的输出频率）

【测量振荡频率】

在时钟误差校正寄存器（SUBCUD）的 F15 为“0”（停止时钟误差校正）时，通过从 RTC1HZ 引脚执行 1Hz 输出，测量各产品的振荡频率注。

注 RTC1HZ 的输出步骤，请参照“7.4.6 实时时钟 2 的 1Hz 输出”。

【计算目标校正值】

（当 RTC1HZ 引脚的输出频率为 0.9999817 Hz 时）

振荡频率 = $32768 \times 0.9999817 \approx 32767.40$ Hz

假设目标频率为 32768 Hz，目标修正值就为：

$$\begin{aligned} \text{目标修正值} &= (\text{振荡频率} - \text{目标频率}) \div \text{目标频率} \\ &= (32767.40 - 32768.00) \div 32768.00 \\ &\approx -18.3 \text{ ppm} \end{aligned}$$

备注 1. 振荡频率为输入时钟（ f_{RTC} ）的值。可以通过停止时钟误差校正时的 RTC1HZ 输出频率 $\times 32768$ 计算振荡频率。

2. 目标校正值是晶体谐振器的振荡频率偏差（单位为 [ppm]）。

3. 目标频率为进行时钟误差校正后的频率。

目标校正值的计算方法 2

（使用副系统时钟频率测量电路）

【测量振荡频率】

通过副系统时钟频率测量电路测量各产品的振荡频率^注。能通过以下计算式计算振荡频率。

$$\text{振荡频率} = \frac{f_{\text{MX}} \text{ 频率 [Hz]} \times \text{运行触发分频比}}{(\text{频率测量计数寄存器 H、L (FMCRLH、FMCRL) 值})_{\text{十进制}}} \text{ [Hz]}$$

注 副系统时钟频率测量功能的运行步骤，请参照“8.4.1 副系统时钟频率测量电路的设定”。

【计算目标校正值】

（当频率测量计数寄存器 H、L 的值为“9999060D”时）

- 高速系统时钟频率（ f_{MX} ）=10MHz
- 频率测量控制寄存器的FMDIV2～FMDIV0 = 111B （运行触发分频比 = 2^{15} ）

上述条件下的振荡频率为：

$$\begin{aligned} \text{振荡频率} &= f_{\text{MX}} \text{ 频率 [Hz]} \times \text{运行触发分频比} \div (\text{FMCRLH、FMCRL) 值} \\ &= 10 \times 10^6 \times 2^{15} \div 9999060D \\ &= 32771.0804816 \text{ Hz} \end{aligned}$$

假设目标频率为 32768 Hz，目标校正值就为：

$$\begin{aligned} \text{目标修正值} &= \text{振荡频率} \div \text{目标频率} - 1 \\ &= 32771.0804816 \div 32768 - 1 \\ &\approx 94.0 \text{ ppm} \end{aligned}$$

备注 1. 运行触发分频比是由频率测量控制寄存器的 FMDIV2 ～ FMDIV0 设定的 f_{SUB} 的分频比。在 FMDIV2 ～ FMDIV0 = 000B 时，运行触发分频比为 2^8 ，在 FMDIV2 ～ FMDIV0 = 111B 时，运行触发分频比为 2^{15} 。

2. 目标校正值是晶体谐振器的振荡频率偏差（单位为 [ppm]）。

3. 目标频率为进行时钟误差校正后的频率。

时钟误差校正寄存器（SBUCUD）的 F8 ~ F0 值的计算方法

能使用目标校正值通过以下计算式计算 SUBCUD 寄存器的 F8 ~ F0 的值。

$$\text{SUBCUD}[8:0] = \left\lceil \frac{\text{目标校正值}[\text{ppm}] \times 2^{15}}{10^6} \right\rceil \text{ 2 的补数 (9 位固定小数点形式) } + 0001.00000\text{B}$$

例 1. 当目标校正值 = -18.3[ppm] 时

$$\begin{aligned} \text{SUBCUD}[8:0] &= (-18.3 \times 2^{15} / 10^6) \text{ 2 的补数 (9 位固定小数点形式) } + 0001.00000\text{B} \\ &= (-0.59965) \text{ 2 的补数 (9 位固定小数点形式) } + 0001.00000\text{B} \\ &= 1111.01101\text{B} + 0001.00000\text{B} \\ &= 0000.01101\text{B} \end{aligned}$$

例 2. 当目标校正值 = 94.0 [ppm] 时

$$\begin{aligned} \text{SUBCUD}[8:0] &= (94.0 \times 2^{15} / 10^6) \text{ 2 的补数 (9 位固定小数点形式) } + 0001.00000\text{B} \\ &= (+3.8019) \text{ 2 的补数 (9 位固定小数点形式) } + 0001.00000\text{B} \\ &= 0011.00011\text{B} + 0001.00000\text{B} \\ &= 0100.00011\text{B} \end{aligned}$$

7.4.9 有关高精度 1Hz 输出

使用时钟误差校正寄存器进行的时钟校正是指每 0.5 秒以最小分辨率 0.96ppm 对寄存器的值进行校正。因为计数器与 f_{RTC} 同步运行，因此因计数器的上溢生成的 1Hz 输出的最小分辨率为 1/32.768KHz ($\approx 30.5\mu\text{s}=30.5\text{ppm}$)。即，长时间的通常 1Hz 输出虽然具有 0.96 ppm 的精度，但是每个 1Hz 输出包括最大 30.5ppm 的误差。

对此，高精度 1Hz 输出是使用时钟误差校正寄存器的校正值，并且通过 f_{IH} 对校正时间进行计数，以最小分辨率 0.96ppm 对每个 1Hz 输出进行校正后再输出注。

注 实际的高精度 1Hz 输出包括 f_{IH} 精度或者校正时间计数时的量化误差。

使用高精度 1Hz 输出时，必须给高速内部振荡器时钟 (f_{IH}) 选择 24MHz，并且运行高速内部振荡器 (HISTOP=0)。不需要对 CPU 时钟进行选择。

第 8 章 副系统时钟频率测量电路

8.1 副系统时钟频率测量电路

副系统时钟频率测量电路通过从外部输入高精度的基准时钟测量副系统时钟（ f_{SUB} ）的频率。

注意 只有在选择副系统时钟（ $f_{SUB} = 32.768\text{ kHz}$ ）为运行时钟时（选择 OSMC 的寄存器的 WUTMMCK0 =0 时），才可使用副系统时钟频率测量电路。

8.2 副系统时钟频率测量电路的结构

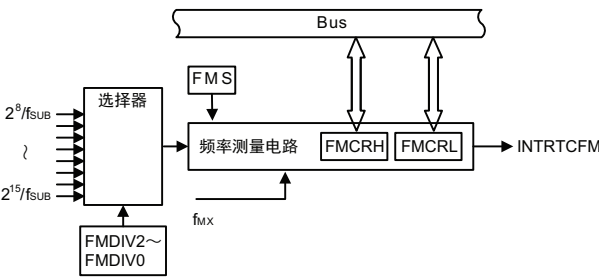
副系统时钟频率测量电路由以下硬件构成。

表 8-1 副系统时钟频率测量电路的结构

项目	结构
计数器	计数器（32 位）
控制寄存器	外围允许寄存器 1（PER1）
	副系统时钟提供模式控制寄存器（OSMC）
	频率测量计数寄存器 L（FMCRL）
	频率测量计数寄存器 H（FMCRH）
	频率测量控制寄存器（FMCTL）

副系统时钟频率测量电路的框图如图 8-1 所示。

图 8-1 副系统时钟频率测量电路的框图



8.3 控制副系统时钟频率测量电路的寄存器

使用以下的寄存器控制副系统时钟频率测量电路。

- 外围允许寄存器 1（PER1）
- 副系统时钟提供模式控制寄存器（OSMC）
- 频率测量计数寄存器 L（FMCRL）
- 频率测量计数寄存器 H（FMCRH）
- 频率测量控制寄存器（FMCTL）

8.3.1 外围允许寄存器 1（PER1）

PER1 寄存器是设定允许或者禁止向各外围硬件提供时钟的寄存器。通过停止向不使用的硬件提供时钟，实现低功耗和低噪声。

可通过将 bit6（FMCEN）设定为“1”，对控制副系统时钟频率测量电路和实时时钟 2 的寄存器中的时钟误差校正寄存器（SUBCUD）进行设定。

通过 1 位或者 8 位存储器操作指令设定 PER1 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 8-2 外围允许寄存器 1（PER1）的格式

地址: F007AH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
PER1	TMKAEN	FMCEN	CMPEEN	0	0	SMCI1EN	SMCI0EN	0

FMCEN	控制副系统时钟频率测量电路的输入时钟的提供
0	停止提供输入时钟。 • 不能写副系统时钟频率测量电路使用的 SFR。 • 不能写实时时钟 2 使用的 SUBCUD 寄存器。 • 副系统时钟频率测量电路处于复位状态。
1	提供输入时钟。 • 能读写副系统时钟频率测量电路使用的 SFR。 • 能读写实时时钟 2 使用的 SUBCUD 寄存器。

- 注意 1. 可通过将外围允许寄存器 0（PER0）的 RTCWEN 位设定为“1”，或将外围允许寄存器 1（PER1）的 FMCEN 位设定为“1”，读写时钟误差校正寄存器（SUBCUD）。
2. 必须将 bit0、3、4 设定为“0”。

8.3.2 副系统时钟提供模式控制寄存器（OSMC）

OSMC 寄存器通过停止不需要的时钟功能实现低功耗。

如果在 STOP 模式或者 CPU 以副系统时钟运行的 HALT 模式中将 RTCLPC 设定为“1”，就停止向实时时钟 2、12 位间隔定时器、时钟输出 / 蜂鸣器输出、LCD 控制器 / 驱动器以外的外围功能提供时钟，因此可降低功耗。

另外，OSMC 寄存器能选择实时时钟 2、12 位间隔定时器、时钟输出 / 蜂鸣器输出、LCD 控制器 / 驱动器以及副系统时钟频率测量电路的运行时钟。

通过 8 位存储器操作指令设定 OSMC 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 8-3 副系统时钟提供模式控制寄存器（OSMC）的格式

复位后：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 2、12 位间隔定时器和 LCD 控制器 / 驱动器的运行时钟	选择时钟输出 / 蜂鸣器输出的 PCLBUZn 引脚的输出时钟	副系统时钟频率测量电路的运行
0	副系统时钟 (f_{SUB})	可以选择副系统时钟 (f_{SUB})	有效
1	低速内部振荡器时钟 (f_{IL})	禁止选择副系统时钟 (f_{SUB})	无效

注意 1. 在副系统时钟振荡时，必须选择副系统时钟（WUTMMCK0 = 0）。

2. 如果将 WUTMMCK0 位设定为“1”，低速内部振荡器时钟就振荡。

3. 如果将 WUTMMCK0 位设定为“1”，就无法使用副系统时钟频率测量电路。

8.3.3 频率测量计数寄存器 L（FMCRL）

FMCRL 寄存器表示频率测量电路中的频率测量计数寄存器（FMCR）的低 16 位。

通过 16 位存储器操作指令读 FMCRL 寄存器。

在产生复位信号后，此寄存器的值变为“0000H”。

图 8-4 频率测量计数寄存器 L（FMCRL）的格式

复位后：F0312H 复位后：0000H R

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

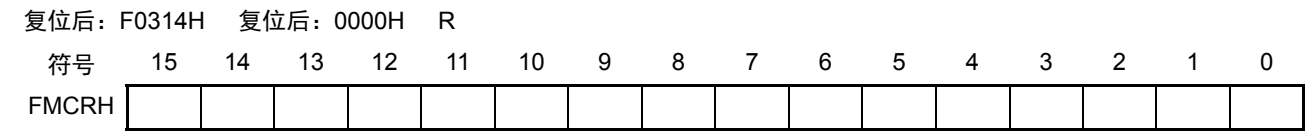
注意 1. 在 FMS=1 时，禁止读 FMCRL 的值。

2. 在产生频率测量结束中断后读 FMCRL 的值。

8.3.4 频率测量计数寄存器 H（FMCRH）

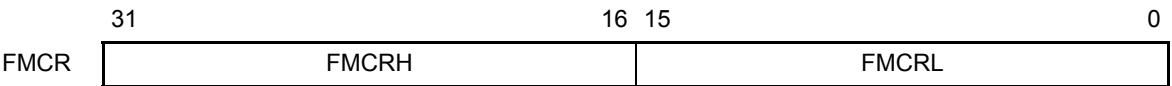
FMCRH 寄存器表示频率测量电路中的频率测量计数寄存器（FMCR）的高 16 位。
通过 16 位存储器操作指令读 FMCRH 寄存器。
在产生复位信号后，此寄存器的值变为“0000H”。

图 8-5 频率测量计数寄存器 H（FMCRH）的格式



- 注意 1. 在 FMS=1 时，禁止读 FMCRH 的值。
2. 在产生频率测量结束中断后读 FMCRH 的值。

图 8-6 频率测量计数寄存器（FMCRH、FMCRL）



8.3.5 频率测量控制寄存器（FMCTL）

FMCTL 寄存器设定副系统时钟频率测量电路的运行。对频率测量运行的开始和频率测量的期间进行设定。

通过 1 位或 8 位存储器操作指令设定 FMCTL 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 8-7 频率测量控制寄存器（FMCTL）的格式

复位后：F0316H 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
FMCTL	FMS	0	0	0	0	FMDIV2	FMDIV1	FMDIV0

FMS	允许频率测量电路运行
0	停止频率测量电路。
1	频率测量电路运行。 在运行时钟的上升沿开始计数，在下一个运行时钟的下降沿停止计数。

FMDIV2	FMDIV1	FMDIV0	设定频率测量期间
0	0	0	$2^8/f_{SUB}$ （7.8125 ms）
0	0	1	$2^9/f_{SUB}$ （15.625 ms）
0	1	0	$2^{10}/f_{SUB}$ （31.25 ms）
0	1	1	$2^{11}/f_{SUB}$ （62.5 ms）
1	0	0	$2^{12}/f_{SUB}$ （0.125 s）
1	0	1	$2^{13}/f_{SUB}$ （0.25 s）
1	1	0	$2^{14}/f_{SUB}$ （0.5 s）
1	1	1	$2^{15}/f_{SUB}$ （1 s）

注意 在 FMS = 1 时，禁止更改 FMDIV2 ~ FMDIV0 位的设定。

备注 可以通过以下的计算式计算频率测量分辨率。

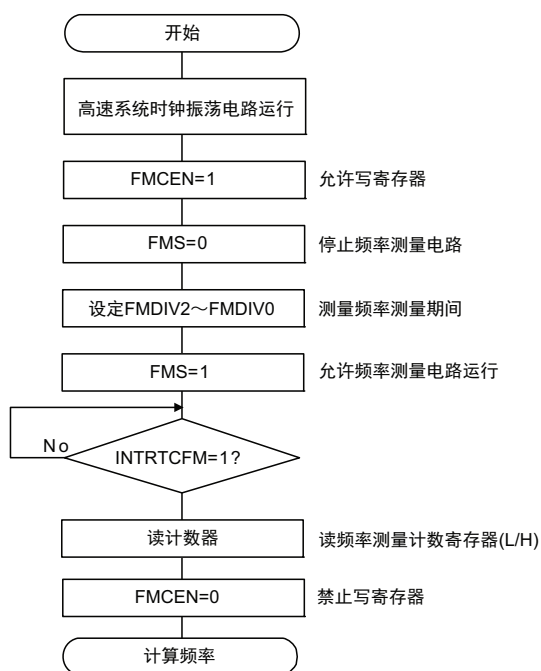
- 频率测量分辨率 = $10^6 / (\text{频率测量期间 [S]} \times \text{基准时钟频率 (f}_{MX}) \text{ [Hz]})$ [ppm]
例 1) 在 FMDIV2 ~ FMDIV0 = 000B、f_{MX} = 20MHz 时，测量分辨率 = 6.4ppm
例 2) 在 FMDIV2 ~ FMDIV0 = 111B、f_{MX} = 1MHz 时，测量分辨率 = 1ppm

8.4 副系统时钟频率测量电路的运行

8.4.1 副系统时钟频率测量电路的设定

必须在将 FMS 设定为“0”后设定频率测量电路。

图 8-8 副系统时钟频率测量电路的设定步骤



注意 必须在读频率测量计数寄存器（L/H）后，将 FMCEN 位设定为“0”。

通过以下计算式计算 f_{SUB} 振荡频率。

$$f_{\text{SUB}} \text{ 振荡频率} = \frac{\text{基准时钟频率 [Hz]} \times \text{运行触发分频比}}{\text{频率测量计数寄存器值 (FMCR)}} \text{ [Hz]}$$

例如，在以下条件下测量频率：

- 计数时钟频率： $f_{\text{MX}} = 10 \text{ MHz}$
- 频率测量周期设定寄存器：FMDIV2 ~ FMDIV0 = 111B（运行触发分频比： 2^{15} ）

当测量结果如下所示时：

- 频率测量计数寄存器： $\text{FMCR} = 10000160\text{D}$

f_{SUB} 振荡频率如下所示：

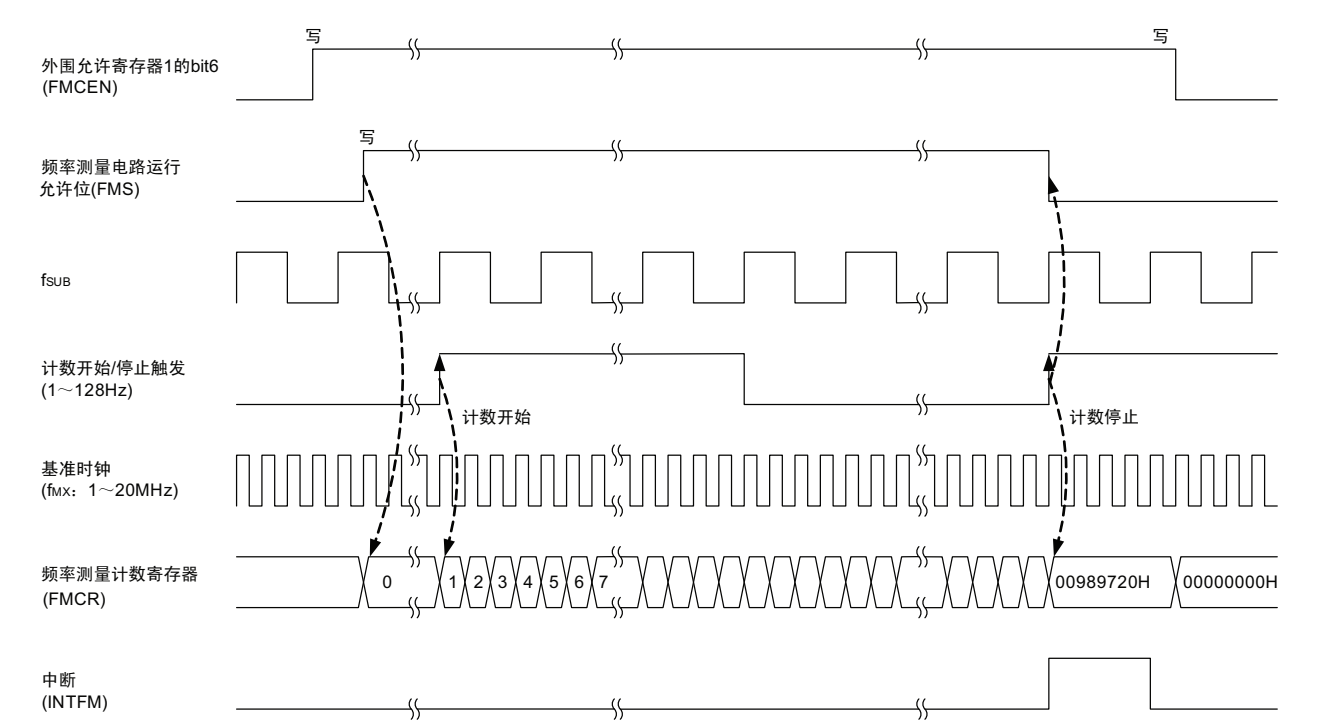
$$f_{\text{SUB}} \text{ 振荡频率} = \frac{(10 \times 10^6) \times 2^{15}}{10000160} = 32767.47572 \text{ [Hz]}$$

8.4.2 副系统时钟频率测量电路的运行时序

副系统时钟频率测量电路的运行时序如图 8-9 所示。

在将频率测量电路运行允许位（FMS）设定为“1”后，通过由频率测量期间设定位（FMDIV2 ~ FMDIV0）设定的计数开始触发开始计数，并且通过下一个触发停止计数。计数停止后，保持计数值，并且频率测量电路运行允许位（FMS）被复位为“0”。另外，产生 1 个 f_{SUB} 时钟的中断。结束频率测量电路的运行并且在读频率测量计数寄存器（L/H）后，必须将外围允许寄存器 1 的 bit6（FMCEN）设定为“0”。

图 8-9 副系统时钟频率测量电路的运行时序



第 9 章 12 位间隔定时器

9.1 12 位间隔定时器的功能

按事先设定的任意时间间隔产生中断（INTIT），能用于从 STOP 模式的唤醒以及 A/D 转换器的 SNOOZE 模式的触发。

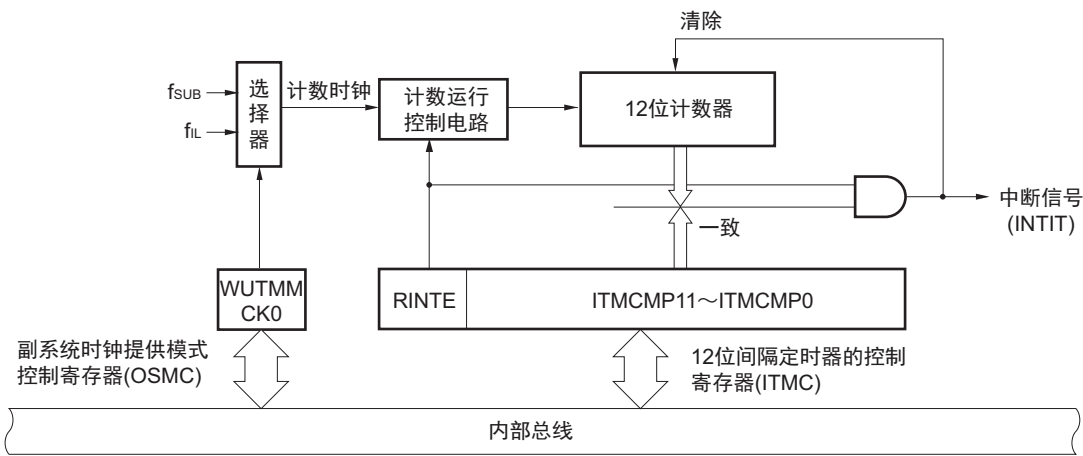
9.2 12 位间隔定时器的结构

12 位间隔定时器由以下硬件构成。

表 9-1 12 位间隔定时器的结构

项目	结构
计数器	12 位计数器
控制寄存器	外围允许寄存器 1（PER1）
	副系统时钟提供模式控制寄存器（OSMC）
	12 位间隔定时器控制寄存器（ITMC）

图 9-1 12 位间隔定时器的框图



9.3 控制 12 位间隔定时器的寄存器

通过以下寄存器控制 12 位间隔定时器。

- 外围允许寄存器 1（PER1）
- 副系统时钟提供模式控制寄存器（OSMC）
- 12 位间隔定时器控制寄存器（ITMC）

9.3.1 外围允许寄存器 1（PER1）

PER1 寄存器是设定允许或者禁止给各外围硬件提供时钟的寄存器。通过停止给不使用的硬件提供时钟，降低功耗和噪声。

通过 1 位或者 8 位存储器操作指令设定 PER1 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 9-2 外围允许寄存器 1（PER1）的格式

地址: F007AH 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
PER1	TMKAEN	FMCEN	CMPEN	0	0	SMCI1EN	SMCI0EN	0

TMKAEN	控制 12 位间隔定时器的输入时钟的提供
0	停止提供输入时钟。 • 不能写 12 位间隔定时器使用的 SFR。 • 12 位间隔定时器处于复位状态。
1	提供输入时钟。 • 能读写 12 位间隔定时器使用的 SFR。

- 注意 1. 如果要使用 12 位间隔定时器，就必须先将 TMKAEN 位设定为“1”。当 TMKAEN 位为“0”时，忽视 12 位间隔定时器的控制寄存器的写操作，而且读取值为初始值（副系统时钟提供模式控制寄存器（OSMC）除外）。
2. 能通过将副系统时钟提供模式控制寄存器（OSMC）的 RTCLPC 位置“1”，在以副系统时钟运行的 HALT 模式中停止给实时时钟 2、12 位间隔定时器、时钟输出 / 蜂鸣器输出和 LCD 控制器 / 驱动器以外的外围功能提供时钟。此时，必须将 TMKAEN 位设定为“1”，PER1 寄存器的 bit0 ~ 6 设定为“0”。
3. 必须将 bit0、3、4 设定为“0”。

9.3.2 副系统时钟提供模式控制寄存器（OSMC）

OSMC 寄存器通过停止不需要的时钟功能实现低功耗。

如果在 STOP 模式或者 CPU 以副系统时钟运行的 HALT 模式中将 RTCLPC 设定为“1”，就停止向实时时钟 2、12 位间隔定时器、时钟输出 / 蜂鸣器输出、LCD 控制器 / 驱动器以外的外围功能提供时钟，因此可降低功耗。

另外，OSMC 寄存器能选择实时时钟 2、12 位间隔定时器、时钟输出 / 蜂鸣器输出、LCD 控制器 / 驱动以及副系统时钟频率测量电路的运行时钟。

通过 8 位存储器操作指令设定 OSMC 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 9-3 副系统时钟提供模式控制寄存器（OSMC）的格式

复位后：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 模式和 CPU 以副系统时钟运行的 HALT 模式时的设定
0	允许向外围功能提供副系统时钟。 (有关允许运行的外围功能，请参照表 21-1 ~ 表 21-3。)
1	停止向实时时钟 2、12 位间隔定时器、时钟输出 / 蜂鸣器输出和 LCD 控制器 / 驱动器以外的外围功能提供时钟。

WUTMMCK0	选择实时时钟 2、12 位间隔定时器和 LCD 控制器 / 驱动器的运行时钟	选择时钟输出 / 蜂鸣器输出的 PCLBUZn 引脚的输出时钟	副系统时钟频率测量电路的运行
0	副系统时钟 (f_{SUB})	可以选择副系统时钟 (f_{SUB})	有效
1	低速内部振荡器时钟 (f_{IL})	禁止选择副系统时钟 (f_{SUB})	无效

注意 1. 在副系统时钟振荡时，必须选择副系统时钟（WUTMMCK0 = 0）。

2. 如果将 WUTMMCK0 位设定为“1”，低速内部振荡器时钟就振荡。

3. 只有在实时时钟 2、12 位间隔定时器和 LCD 控制器 / 驱动器的全部功能都停止运行时，才能通过 WUTMMCK0 位进行副系统时钟和低速内部振荡器时钟的切换。

9.3.3 12 位间隔定时器控制寄存器 (ITMC)

ITMC 寄存器是设定 12 位间隔定时器的运行开始和停止以及比较值的寄存器。
通过 16 位存储器操作指令设定 ITMC 寄存器。
在产生复位信号后，此寄存器的值变为“0FFFH”。

图 9-4 12 位间隔定时器控制寄存器 (ITMC) 的格式

地址: FFF90H 复位后: 0FFFH R/W

符号	15	14	13	12	11 ~ 0
ITMC	RINTE	0	0	0	ITMCMP11 ~ ITMCMP0

RINTE	控制 12 位间隔定时器的运行
0	停止计数器运行 (计数清除)
1	开始计数器运行

ITMCMP11 ~ ITMCMP0	设定 12 位间隔定时器的比较值
001H	产生 “计数时钟周期 × (ITMCMP 设定值 + 1)” 的固定周期中断。
⋮	
FFFH	
000H	禁止设定
ITCMP11 ~ ITCMP0 为 “001H” 或者 “FFFH” 时的中断周期例子	
• ITCMP11 ~ ITCMP0 = 001H, 计数时钟: f _{SUB} =32.768kHz 1/32.768[kHz] × (1+1) = 0.06103515625[ms] ≈ 61.03[μs]	
• ITCMP11 ~ ITCMP0 = FFFH, 计数时钟: f _{SUB} =32.768kHz 1/32.768[kHz] × (4095+1) = 125[ms]	

- 注意 1. 要将 RINTE 位从 “1” 改为 “0” 时，必须在通过中断屏蔽标志寄存器来禁止 INTIT 中断处理后改写 RINTE 位。要重新开始运行 (从 “0” 改为 “1”) 时，必须在清除 ITIF 标志后允许中断处理。
2. RINTE 位的读取值在设定 RINTE 位后的 1 个计数时钟之后被反映。
3. 要在从待机模式返回后设定 RINTE 位并且再次转移到待机模式时，必须确认在 RINTE 位的写入值被反映后或者在从待机模式返回后至少经过 1 个计数时钟的时间之后再转移到待机模式。
4. 要更改 ITCMP11 ~ ITCMP0 位的设定时，必须在 RINTE 位为 “0” 的状态下进行。
但是，能在将 RINTE 位从 “0” 改为 “1” 或者从 “1” 改为 “0” 的同时更改 ITCMP11 ~ ITCMP0 位的设定。

9.4 12 位间隔定时器的运行

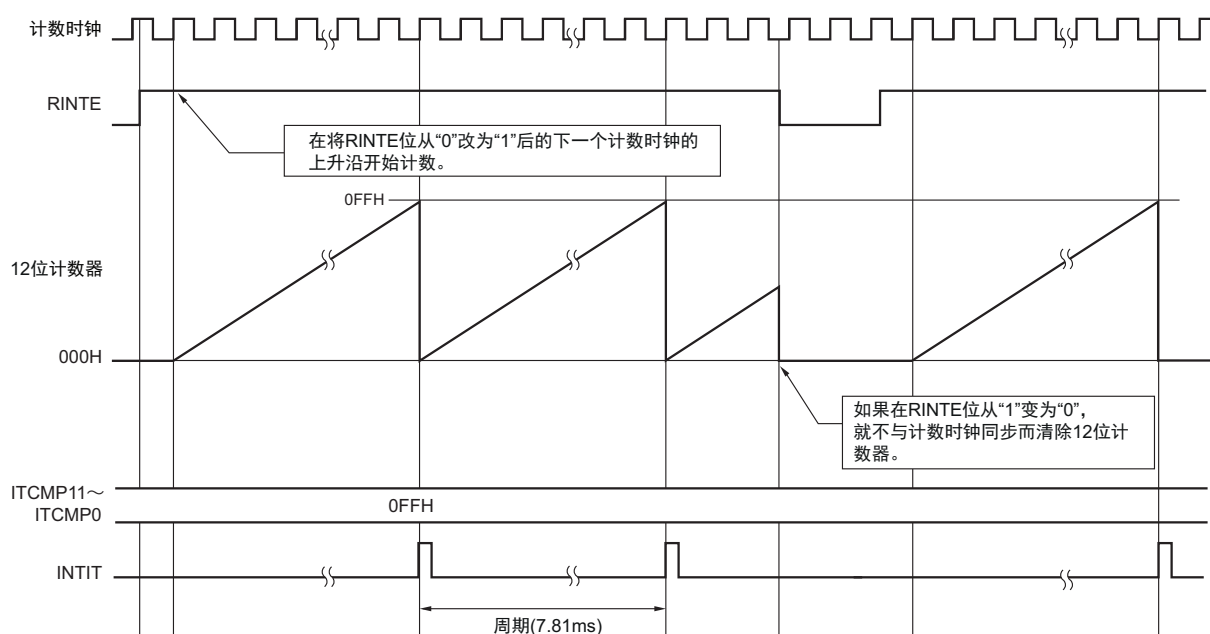
9.4.1 12 位间隔定时器的运行时序

以 ITCMP11 ~ ITCMP0 位设定的计数值为间隔，作为重复产生中断请求 (INTIT) 的 12 位间隔定时器运行。
如果将 RINTE 位置“1”，12 位计数器就开始计数。

当 12 位计数器的值和 ITCMP11 ~ ITCMP0 位的设定值相同时，将 12 位计数器清“0”并且继续进行计数，同时产生中断请求信号 (INTIT)。

12 位间隔定时器的基本运行如图 9-5 所示。

图 9-5 12 位间隔定时器操作时序
(ITCMP11 ~ ITCMP0 = 0FFH, 计数时钟: $f_{\text{SUB}} = 32.768 \text{ kHz}$)

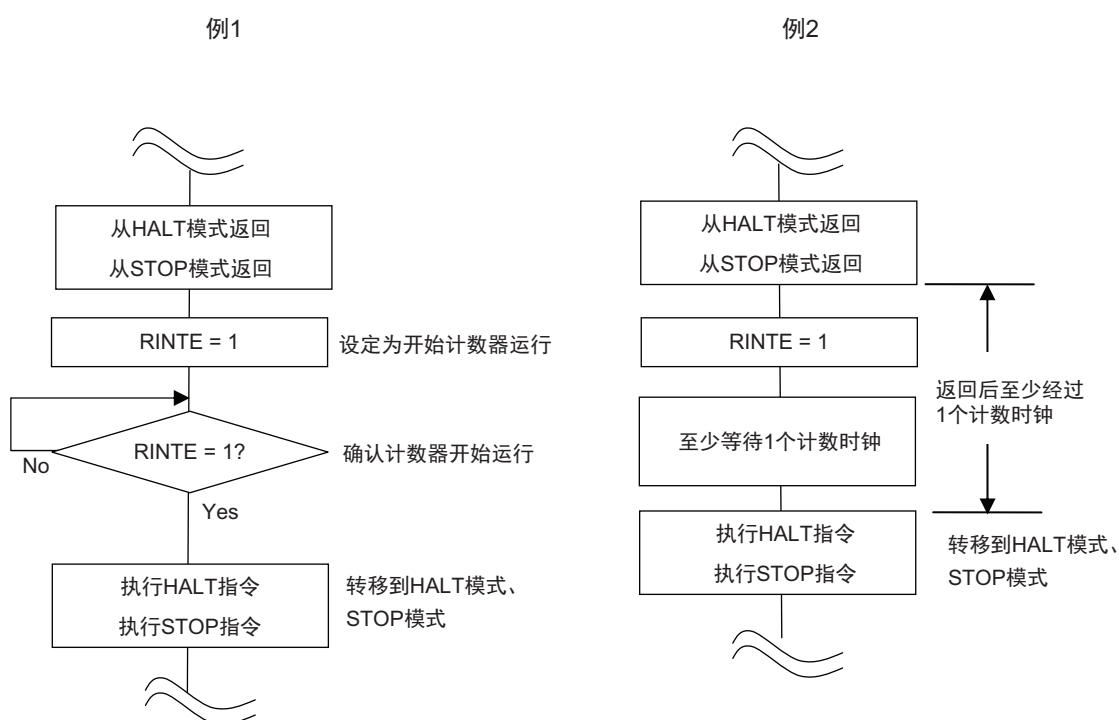


9.4.2 从 HALT/STOP 模式返回后计数器开始运行并且再次转移到 HALT/STOP 模式

如果从 HALT 模式或者 STOP 模式返回后将 RINTE 置“1”，并且要再次转移到 HALT 模式或者 STOP 模式，就必须在将 RINTE 置“1”后，确认是否已反映 RINTE 位的写入值，或者在返回后是否至少经过 1 个计数时钟，然后再进行转移。

- 将 RINTE 位置“1”后，通过轮询确认 RINTE 位已变为“1”，然后转移到 HALT 模式或者 STOP 模式（参照图 9-6 的例 1）。
- 将 RINTE 位置“1”后，至少经过 1 个计数时钟，然后转移到 HALT 模式或者 STOP 模式（参照图 9-6 的例 2）。

图 9-6 将 RINTE 位置“1”后向 HALT 模式或者 STOP 模式转移的步骤



第 10 章 时钟输出 / 蜂鸣器输出控制电路

10.1 时钟输出 / 蜂鸣器输出控制电路的功能

时钟输出是输出提供给外围 IC 的时钟的功能，蜂鸣器输出是输出蜂鸣器频率的方波的功能。

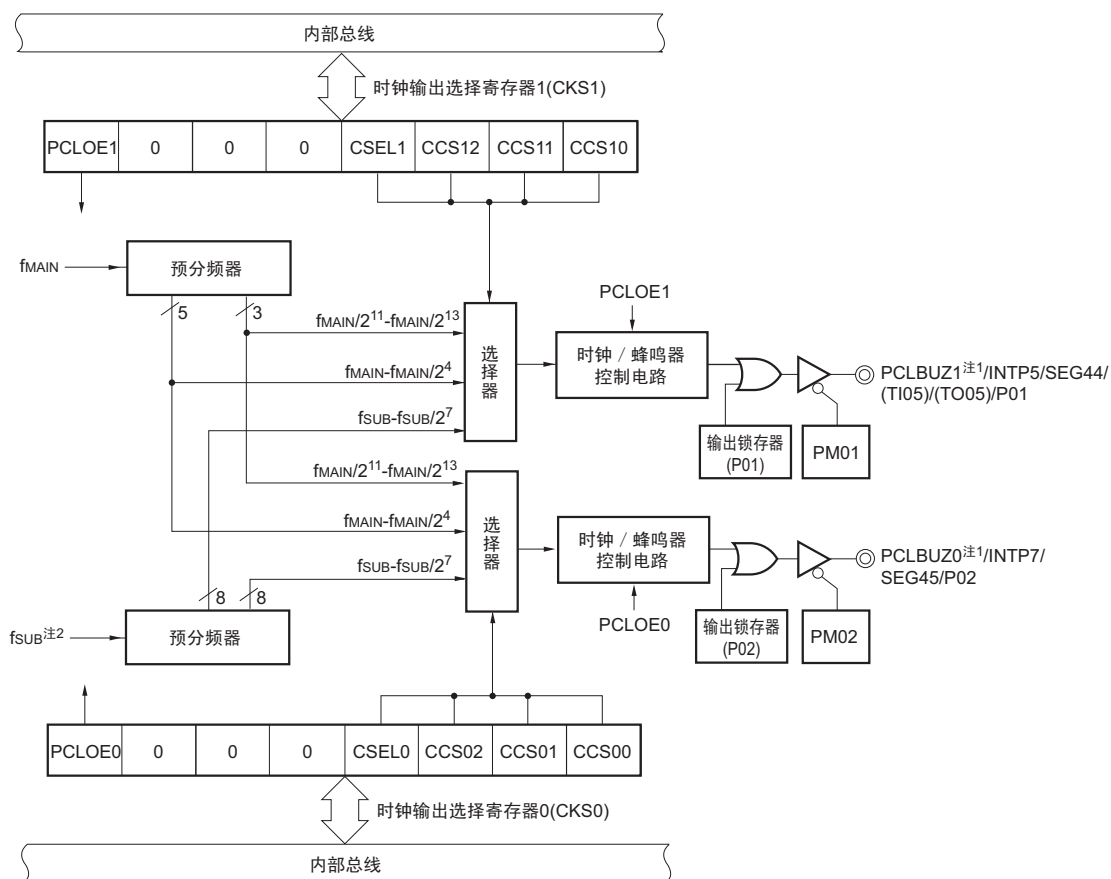
能通过 1 个引脚在选择用作时钟输出或者蜂鸣器输出后进行输出。

PCLBUZn 引脚输出由时钟输出选择寄存器 n (CKSn) 选择的时钟。

时钟输出 / 蜂鸣器输出控制电路的框图如图 10-1 所示。

备注 n = 0、1

图 10-1 时钟输出 / 蜂鸣器输出控制电路的框图



- 注 1. 有关可从 PCLBUZ0 和 PCLBUZ1 引脚输出的频率，请参照“32.4 AC 特性”。
2. 当 OSMC 寄存器的 WUTMMCK0 位设定为“1”时，禁止选择 f_{SUB} 作为时钟输出 / 蜂鸣器输出的输出时钟。

备注 此图中的时钟输出 / 蜂鸣器输出引脚是 PIOR3 位为“0”的情况。

其他情况下的引脚名、输出锁存器 (Pxx) 以及 PMxx 不同，必须替换。

10.2 时钟输出 / 蜂鸣器输出控制电路的结构

时钟输出 / 蜂鸣器输出控制电路由以下硬件构成。

表 10-1 时钟输出 / 蜂鸣器输出控制电路的结构

项目	结构
控制寄存器	时钟输出选择寄存器 n (CKSn) 副系统时钟提供模式控制寄存器 (OSMC) 端口模式寄存器 0 (PM0) 端口寄存器 0 (P0)

10.3 控制时钟输出 / 蜂鸣器输出控制电路的寄存器

通过以下两种寄存器控制时钟输出 / 蜂鸣器输出控制电路。

- 时钟输出选择寄存器 n (CKSn)
- 副系统时钟提供模式控制寄存器 (OSMC)
- 端口模式寄存器 0 (PM0)

10.3.1 时钟输出选择寄存器 n (CKSn)

这是允许或者禁止时钟输出引脚或者蜂鸣器频率输出引脚 (PCLBUZn) 的输出以及设定输出时钟的寄存器。

通过 CKSn 寄存器选择 PCLBUZn 引脚输出的时钟。

通过 1 位或者 8 位存储器操作指令设定 CKSn 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 10-2 时钟输出选择寄存器 n (CKSn) 的格式

地址: FFFA5H (CKS0)、FFFA6H (CKS1) 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
CKSn	PCLOEn	0	0	0	CSELn	CCSn2	CCSn1	CCSn0

PCLOEn	设定允许 / 禁止 PCLBUZn 引脚的输出
0	禁止输出 (默认)
1	允许输出

CSELn	CCSn2	CCSn1	CCSn0		选择 PCLBUZn 引脚的输出时钟			
					f _{MAIN} = 5 MHz	f _{MAIN} = 10MHz	f _{MAIN} = 20 MHz	f _{MAIN} = 24 MHz
0	0	0	0	f _{MAIN}	5MHz	10MHz 注 1	禁止设定注 1	禁止设定注 1
0	0	0	1	f _{MAIN} /2	2.5MHz	5MHz	10MHz 注 1	12MHz 注 1
0	0	1	0	f _{MAIN} /2 ²	1.25MHz	2.5MHz	5MHz	6MHz
0	0	1	1	f _{MAIN} /2 ³	625kHz	1.25MHz	2.5MHz	3MHz
0	1	0	0	f _{MAIN} /2 ⁴	312.5kHz	625kHz	1.25MHz	1.5MHz
0	1	0	1	f _{MAIN} /2 ¹¹	2.44kHz	4.88kHz	9.77kHz	11.7kHz
0	1	1	0	f _{MAIN} /2 ¹²	1.22kHz	2.44kHz	4.88kHz	5.86kHz
0	1	1	1	f _{MAIN} /2 ¹³	610Hz	1.22kHz	2.44kHz	2.93kHz
1	0	0	0	f _{SUB} 注 2	32.768kHz			
1	0	0	1	f _{SUB} /2 注 2	16.384kHz			
1	0	1	0	f _{SUB} /2 ² 注 2	8.192kHz			
1	0	1	1	f _{SUB} /2 ³ 注 2	4.096kHz			
1	1	0	0	f _{SUB} /2 ⁴ 注 2	2.048kHz			
1	1	0	1	f _{SUB} /2 ⁵ 注 2	1.024kHz			
1	1	1	0	f _{SUB} /2 ⁶ 注 2	512Hz			
1	1	1	1	f _{SUB} /2 ⁷ 注 2	256Hz			

注 1. 在 16MHz 的范围内使用输出时钟。详细内容请参照“32.4 AC 特性”。

2. 当 OSMC 寄存器的 WUTMMCK0 位设定为“1”时，禁止选择 f_{SUB} 作为时钟输出 / 蜂鸣器输出的输出时钟。

注意 1. 必须在设定为禁止输出 (PCLOEn=0) 后，才可以进行输出时钟的切换。

2. 如果在选择主系统时钟 (CSELn=0) 时转移到 STOP 模式，就必须在执行 STOP 指令之前将 PCLOEn 设定为“0”。在选择副系统时钟 (CSELn=1) 时，因为能在 STOP 模式中输出时钟，所以可以将 PCLOEn 设定为“1”。

备注 1. n = 0、1

2. f_{MAIN}: 主系统时钟频率f_{SUB}: 副系统时钟频率

10.3.2 副系统时钟提供模式控制寄存器（OSMC）

OSMC 寄存器通过停止不需要的时钟功能实现低功耗。

如果在 STOP 模式或者 CPU 以副系统时钟运行的 HALT 模式中将 RTCLPC 设定为“1”，就停止向实时时钟 2、12 位间隔定时器、时钟输出 / 蜂鸣器输出、LCD 控制器 / 驱动器以外的外围功能提供时钟，因此可降低功耗。

另外，OSMC 寄存器能选择实时时钟 2、12 位间隔定时器、时钟输出 / 蜂鸣器输出、LCD 控制器 / 驱动以及副系统时钟频率测量电路的运行时钟。

通过 8 位存储器操作指令设定 OSMC 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 10-3 副系统时钟提供模式控制寄存器（OSMC）的格式

复位后：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 模式和 CPU 以副系统时钟运行的 HALT 模式时的设定
0	允许向外围功能提供副系统时钟。 (有关允许运行的外围功能，请参照表 21-1 ~ 表 21-3。)
1	停止向实时时钟 2、12 位间隔定时器、时钟输出 / 蜂鸣器输出和 LCD 控制器 / 驱动器以外的外围功能提供时钟。

WUTMMCK0	选择实时时钟 2、12 位间隔定时器和 LCD 控制器 / 驱动器的运行时钟	选择时钟输出 / 蜂鸣器输出的 PCLBUZn 引脚的输出时钟	副系统时钟频率测量电路的运行
0	副系统时钟 (f_{SUB})	可以选择副系统时钟 (f_{SUB})	有效
1	低速内部振荡器时钟 (f_{IL})	禁止选择副系统时钟 (f_{SUB})	无效

注意 1. 在副系统时钟振荡时，必须选择副系统时钟（WUTMMCK0 = 0）。

2. 如果将 WUTMMCK0 位设定为“1”，低速内部振荡器时钟就振荡。

3. 在将 WUTMMCK0 置“1”后，禁止选择 f_{SUB} 作为时钟输出 / 蜂鸣器输出的输出时钟。

10.3.3 控制时钟输出 / 蜂鸣器输出引脚的端口功能的寄存器

在用作时钟输出 / 蜂鸣器输出功能时，必须设定与对象通道复用的端口功能的控制寄存器（端口模式寄存器（PMxx）、端口寄存器（Pxx））。详细内容请参照“4.3.1 端口模式寄存器（PMxx）”和“4.3.2 端口寄存器（Pxx）”。

在将与时钟输出 / 蜂鸣器输出引脚复用的端口（P02/SEG45/INTP7/PCLBUZ0、P01/SEG44/(TI05)/(TO05)/(INTP5)/PCLBUZ1 等）用作时钟输出 / 蜂鸣器输出时，必须将与各端口对应的端口模式寄存器（PMxx）的位和端口寄存器（Pxx）的位置“0”。

例 将 P02/SEG45/INTP7/PCLBUZ0 用作时钟输出 / 蜂鸣器输出时

将端口模式寄存器 0 的 PM02 位置“0”。

将端口寄存器 0 的 P02 位置“0”。

10.4 时钟输出 / 蜂鸣器输出控制电路的运行

能通过 1 个引脚在选择用作时钟输出或者蜂鸣器输出后进行输出。

PCLBUZ0 引脚输出由时钟输出选择寄存器 0 (CKS0) 选择的时钟 / 蜂鸣器。

PCLBUZ1 引脚输出由时钟输出选择寄存器 1 (CKS1) 选择的时钟 / 蜂鸣器。

10.4.1 输出引脚的运行

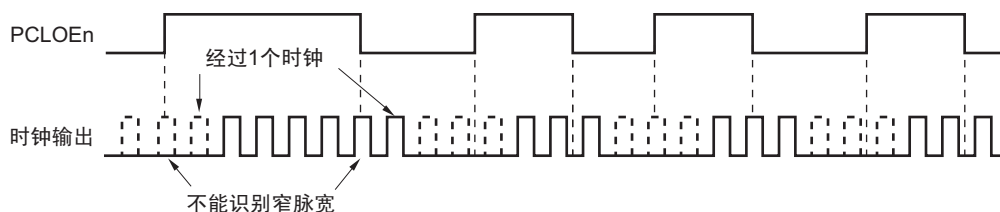
PCLBUZn 引脚按照以下步骤进行输出：

- ① 将与用作 PCLBUZn 引脚的端口对应的端口模式寄存器 (PMxx) 和端口寄存器 (Pxx) 的位置“0”。
- ② 通过 PCLBUZn 引脚的时钟输出选择寄存器 (CKSn) 的 bit0~3 (CCSn0~CCSn2、CSELn) 选择输出频率 (输出为禁止状态)。
- ③ 将 CKSn 寄存器的 bit7 (PCLOEn) 置“1”，允许时钟/蜂鸣器的输出。

备注 1. 用作时钟输出时的控制电路在允许或者禁止时钟输出 (PCLOEn 位) 后的 1 个时钟之后，开始或者停止时钟输出。此时不输出宽度窄的脉冲。通过 PCLOEn 位允许或者停止输出以及时钟输出的时序如图 10-4 所示。

2. n=0、1

图 10-4 PCLBUZn 引脚的时钟输出时序



10.5 时钟输出 / 蜂鸣器输出控制电路的注意事项

当选择主系统时钟作为 PCLBUZn 输出 (CSELn=0) 时，如果在设定停止输出 (PCLOEn=0) 后的 1.5 个主系统时钟内转移到 STOP 模式，PCLBUZn 的输出宽度就变窄。

第 11 章 看门狗定时器

11.1 看门狗定时器的功能

看门狗定时器通过选项字节（000C0H）设定计数运行。

看门狗定时器以低速内部振荡器时钟（ f_{LL} ）运行。

看门狗定时器用于检测程序失控。在检测到程序失控时，产生内部复位信号。

下述情况判断为程序失控。

- 当看门狗定时器计数器发生上溢时
- 当对看门狗定时器的允许寄存器（WDTE）执行1位操作指令时
- 当给WDTE寄存器写“ACH”以外的数据时
- 在窗口关闭期间给WDTE寄存器写数据时

当因看门狗定时器而发生复位时，将复位控制标志寄存器（RESF）的 bit4（WDTRF）置“1”。有关 RESF 寄存器的详细内容，请参照“第 22 章 复位功能”。

当达到上溢时间的 $75\% + 1/2f_{LL}$ 时，能产生间隔中断。

11.2 看门狗定时器的结构

看门狗定时器由以下硬件构成。

表 11-1 看门狗定时器的结构

项目	结构
计数器	内部计数器（17 位）
控制寄存器	看门狗定时器的允许寄存器（WDTE）

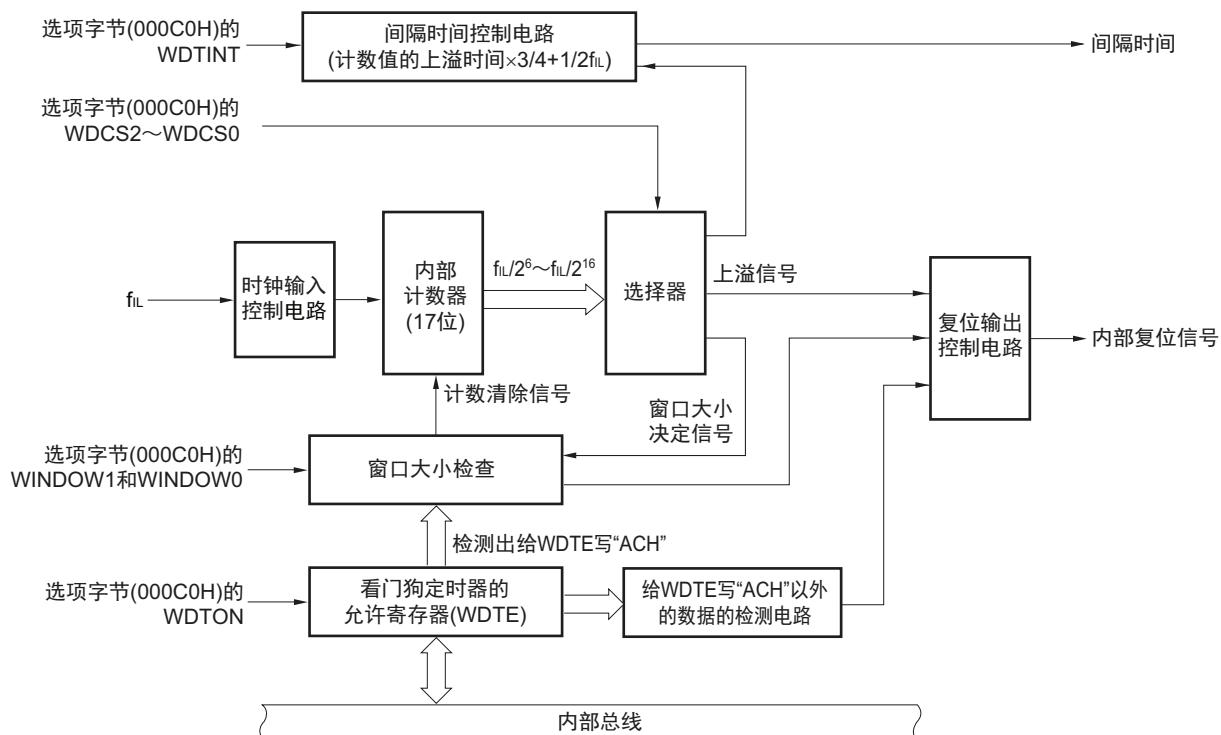
通过选项字节控制计数器的运行以及设定上溢时间、窗口打开期间和间隔中断。

表 11-2 选项字节和看门狗定时器的设定内容

看门狗定时器的设定内容	选项字节（000C0H）
看门狗定时器的间隔中断的设定	bit7（WDTINT）
窗口打开期间的设定	bit6 和 bit5（WINDOW1、WINDOW0）
看门狗定时器的计数器运行控制	bit4（WDTON）
看门狗定时器的上溢时间的设定	bit3 ~ 1（WDSC2 ~ WDSC0）
看门狗定时器的计数器运行控制（HALT/STOP 模式时）	bit0（WDSTBYON）

备注 有关选项字节，请参照“第 27 章 选项字节”。

图 11-1 看门狗定时器的框图



备注 f_{IL} : 低速内部振荡器时钟

11.3 控制看门狗定时器的寄存器

通过看门狗定时器的允许寄存器（WDTE）控制看门狗定时器。

11.3.1 看门狗定时器的允许寄存器（WDTE）

通过给 WDTE 寄存器写“ACH”，清除看门狗定时器的计数器并且重新开始计数。

通过 8 位存储器操作指令设定 WDTE 寄存器。

在产生复位信号后，此寄存器的值变为“9AH”或者“1AH”注。

图 11-2 看门狗定时器允许寄存器（WDTE）的格式

地址: FFFABH	复位后: 9AH/1AH 注	R/W						
符号	7	6	5	4	3	2	1	0
WDTE								

注 WDTE 寄存器的复位值因选项字节（000C0H）的 WDTON 位的设定值而不同。要使看门狗定时器运行时，必须将 WDTON 位置“1”。

WDTON 位的设定值	WDTE 寄存器的复位值
0（禁止看门狗定时器的计数运行）	1AH
1（允许看门狗定时器的计数运行）	9AH

- 注意 1. 当给 WDTE 寄存器写“ACH”以外的值时，产生内部复位信号。
2. 当对 WDTE 寄存器执行 1 位操作指令时，产生内部复位信号。
3. WDTE 寄存器的读取值为“9AH/1AH”（和写入值（“ACH”）不同）。

11.4 看门狗定时器的运行

11.4.1 看门狗定时器的运行控制

- 当使用看门狗定时器时，通过选项字节（000C0H）设定以下内容：
 - 必须将选项字节（000C0H）的 bit4（WDTON）置“1”，允许看门狗定时器的计数运行（在解除复位后，计数器开始运行）（详细内容请参照第 27 章）。

WDTON	看门狗定时器的计数器
0	禁止计数运行（解除复位后停止计数）。
1	允许计数运行（解除复位后开始计数）。

- 必须通过选项字节（000C0H）的 bit3～1（WDSC2～WDSC0）设定上溢时间（详细内容请参照 11.4.2 和第 27 章）。
 - 必须通过选项字节（000C0H）的 bit6 和 bit5（WINDOW1、WINDOW0）设定窗口打开期间（详细内容请参照 11.4.3 和第 27 章）。
- 在解除复位后，看门狗定时器开始计数。
 - 在开始计数后并且在选项字节所设上溢时间前，如果给看门狗定时器的允许寄存器（WDTE）写“ACH”，就清除看门狗定时器并且重新开始计数。
 - 此后，解除复位后的第 2 次以及以后的 WDTE 寄存器的写操作必须在窗口打开期间进行。如果在窗口关闭期间写 WDTE 寄存器，就产生内部复位信号。
 - 如果不给 WDTE 寄存器写“ACH”而超过上溢时间，就产生内部复位信号。
以下情况会产生内部复位信号：
 - 当对 WDTE 寄存器执行 1 位操作指令时
 - 当给 WDTE 寄存器写“ACH”以外的数据时

- 注意 1. 在解除复位后第 1 次写看门狗定时器的允许寄存器（WDTE）时，与窗口打开期间无关，只要在上溢时间前的任何时刻写 WDTE，就都清除看门狗定时器，并且重新开始计数。
- 从给 WDTE 寄存器写“ACH”到清除看门狗定时器的计数器，可能产生最大 2 个 f_{L} 时钟的误差。
 - 在计数值发生上溢前，都能清除看门狗定时器。
 - 如下所示，看门狗定时器在 HALT 模式、STOP 模式或者 SNOOZE 模式中的运行因选项字节（000C0H）的 bit0（WDSTBYON）的设定值而不同。

	WDSTBYON=0	WDSTBYON=1
HALT 模式	停止看门狗定时器运行。	继续看门狗定时器运行。
STOP 模式		
SNOOZE 模式		

当 WDSTBYON 位为“0”时，在解除 HALT 模式或者 STOP 模式后重新开始看门狗定时器的计数。此时，将计数器清“0”，开始计数。

当解除 STOP 模式后以 X1 振荡时钟运行时，CPU 在经过振荡稳定时间后开始运行。

如果从解除 STOP 模式到看门狗定时器发生上溢的时间较短，就会在振荡稳定时间内发生看门狗上溢而产生复位。

因此，在通过间隔中断解除 STOP 模式后，如果要以 X1 振荡时钟运行并且要清除看门狗定时器，因为在经过振荡稳定时间后才清除看门狗定时器，所以必须考虑这种情况进行上溢时间的设定。

11.4.2 看门狗定时器上溢时间的设定

通过选项字节（000C0H）的 bit3 ~ 1（WDCS2 ~ WDCS0）设定看门狗定时器的上溢时间。

在发生上溢时，产生内部复位信号。在上溢时间前的窗口打开期间，通过给看门狗定时器的允许寄存器（WDTE）写“ACH”，清除计数并且重新开始计数。

能设定的上溢时间如下所示。

表 11-3 看门狗定时器上溢时间的设定

WDCS2	WDCS1	WDCS0	看门狗定时器的上溢时间 ($f_{IL}=17.25\text{kHz}(\text{MAX.})$ 的情况)
0	0	0	$2^6/f_{IL}$ (3.71ms)
0	0	1	$2^7/f_{IL}$ (7.42ms)
0	1	0	$2^8/f_{IL}$ (14.84ms)
0	1	1	$2^9/f_{IL}$ (29.68ms)
1	0	0	$2^{11}/f_{IL}$ (118.72ms)
1	0	1	$2^{13}/f_{IL}$ (474.89ms)
1	1	0	$2^{14}/f_{IL}$ (949.79ms)
1	1	1	$2^{16}/f_{IL}$ (3799.18ms)

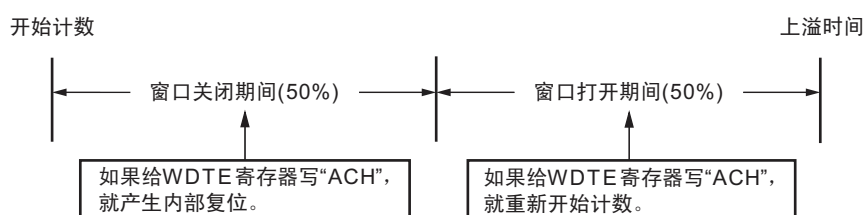
备注 f_{IL} ：低速内部振荡器的时钟频率

11.4.3 看门狗定时器窗口打开期间的设定

通过选项字节（000C0H）的 bit6 和 bit5（WINDOW1、WINDOW0）设定看门狗定时器的窗口打开期间。窗口概要如下：

- 如果在窗口打开期间给看门狗定时器的允许寄存器（WDTE）写“ACH”，就清除看门狗定时器，并且重新开始计数。
- 在窗口关闭期间，即使给 WDTE 寄存器写“ACH”，也会检测到异常并且产生内部复位信号。

例 窗口打开期间为 50% 的情况



注意 在解除复位后第 1 次写看门狗定时器的允许寄存器（WDTE）时，与窗口打开期间无关，只要在上溢时间前的任何时刻写 WDTE，就都清除看门狗定时器，并且重新开始计数。

能设定的窗口打开期间如下所示。

表 11-4 看门狗定时器窗口打开期间的设定

WINDOW1	WINDOW0	看门狗定时器的窗口打开期间
0	0	禁止设定。
0	1	50%
1	0	75%
1	1	100%

注意 当选项字节（000C0H）的 bit0（WDSTBYON）为“0”时，与 WINDOW1 位和 WINDOW0 位的值无关，窗口打开期间为 100%。

备注 当将上溢时间设定为 $2^9/f_{IL}$ 时，窗口关闭时间和打开时间如下所示。

	窗口打开期间的设定		
	50%	75%	100%
窗口关闭时间	0 ~ 20.08ms	0 ~ 10.04ms	无
窗口打开时间	20.08 ~ 29.68ms	10.04 ~ 29.68ms	0 ~ 29.68ms

<当窗口打开期间为 50% 时>

- 上溢时间：
 $2^9/f_{IL}(\text{MAX.}) = 2^9/17.25\text{kHz} = 29.68\text{ms}$
- 窗口关闭时间：
 $0 \sim 2^9/f_{IL}(\text{MIN.}) \times (1-0.5) = 0 \sim 2^9/12.75\text{kHz} \times 0.5 = 0 \sim 20.08\text{ms}$
- 窗口打开时间：
 $2^9/f_{IL}(\text{MIN.}) \times (1-0.5) \sim 2^9/f_{IL}(\text{MAX.}) = 2^9/12.75\text{kHz} \times 0.5 \sim 2^9/17.25\text{kHz} = 20.08 \sim 29.68\text{ms}$

11.4.4 看门狗定时器间隔中断的设定

能通过设定选项字节（000C0H）的 bit7（WDTINT），在达到上溢时间的 $75\% + 1/2f_{IL}$ 时产生间隔中断（INTWDTI）。

表 11-5 看门狗定时器间隔中断的设定

WDTINT	看门狗定时器间隔中断的使用 / 不使用
0	不使用间隔中断。
1	在达到上溢时间的 $75\% + 1/2f_{IL}$ 时，产生间隔中断。

注意 当解除 STOP 模式后以 X1 振荡时钟运行时，CPU 在经过振荡稳定时间后开始运行。
 如果从解除 STOP 模式到看门狗定时器发生上溢的时间较短，就会在振荡稳定时间内发生看门狗上溢而产生复位。
 因此，在通过间隔中断解除 STOP 模式后，如果要以 X1 振荡时钟运行并且要清除看门狗定时器，因为在经过振荡稳定时间后才清除看门狗定时器，所以必须考虑这种情况进行上溢时间的设定。

备注 即使在产生 INTWDTI 后也继续计数（直到给看门狗定时器的允许寄存器（WDTE）写“ACH”为止）。如果在上溢时间前不给 WDTE 寄存器写“ACH”，就产生内部复位信号。

第 12 章 A/D 转换器

12.1 A/D 转换器的功能

A/D 转换器是将模拟输入转换为数字值的转换器，最多能控制 4 个通道的 A/D 转换器模拟输入（ANI0、ANI1、ANI16、ANI17）。通过 A/D 转换器的模式寄存器 2（ADM2）的 ADTYP 位，选择 10 位分辨率或者 8 位分辨率。

A/D 转换器有以下功能。

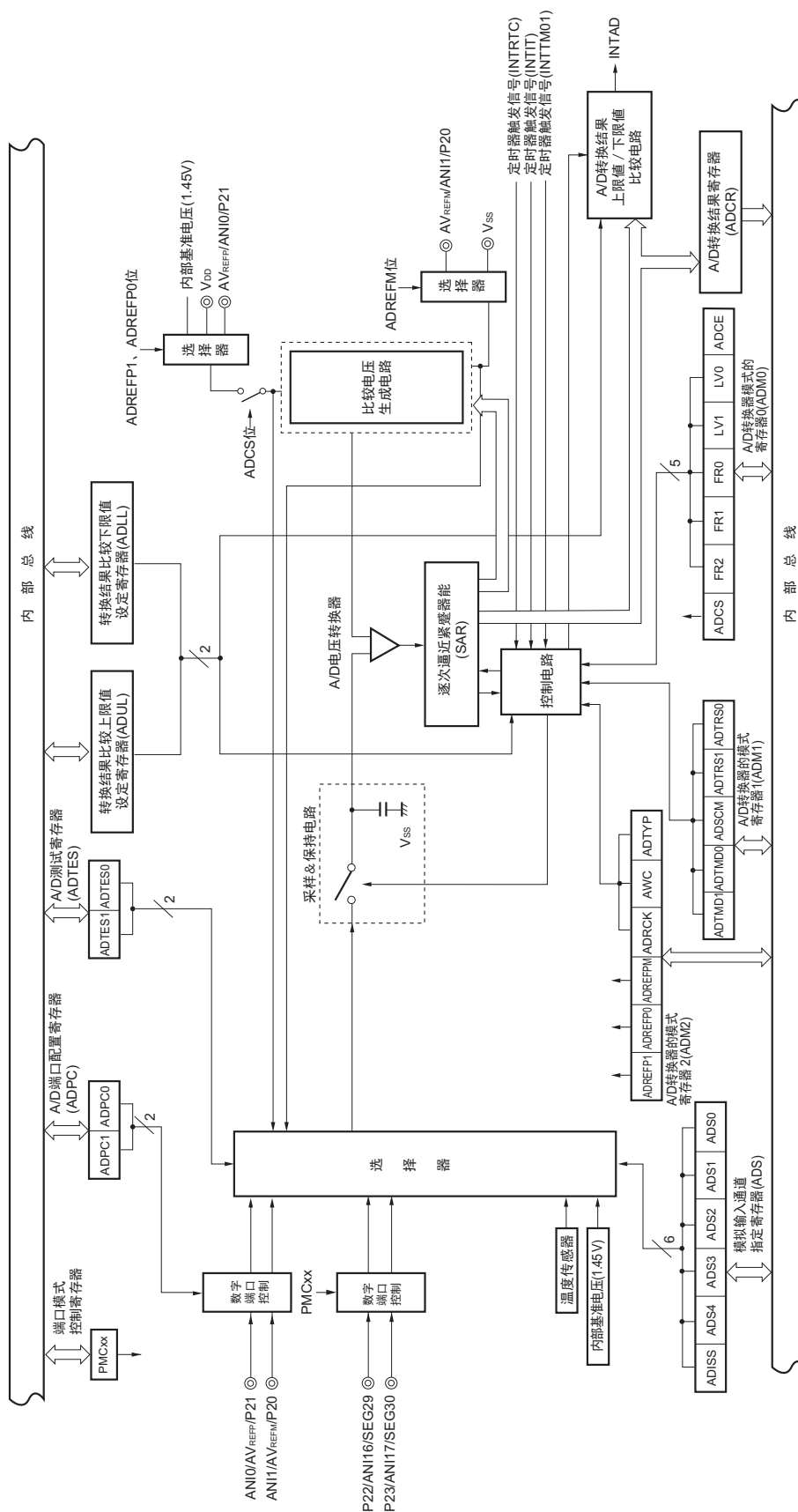
- 10 位/8 位分辨率的 A/D 转换

从 ANI0、ANI1、ANI16、ANI17 中选择 1 个通道的模拟输入，重复进行 10 位/8 位分辨率的 A/D 转换。每结束 1 次 A/D 转换，就产生中断请求（INTAD）（选择模式的情况）。

能通过下述的模式组合设定各种 A/D 转换模式。

触发模式	软件触发	通过软件操作，开始转换运行。
	硬件触发无等待模式	通过检测到硬件触发，开始转换运行。
	硬件触发等待模式	在切断电源的转换待机状态下，通过检测到硬件触发，接通电源，并且在经过 A/D 电源稳定等待时间后自动开始转换运行。在使用 SNOOZE 模式功能时，必须选择硬件触发等待模式。
转换运行模式	单次转换模式	对所选通道进行 1 次 A/D 转换。
	连续转换模式	对所选通道进行连续的 A/D 转换，直到通过软件停止为止。
工作电压模式	标准 1/ 标准 2 模式	在 $2.7V \leq V_{DD} \leq 5.5V$ 的工作电压范围内进行转换运行时选择此模式。
	低电压 1/ 低电压 2 模式	能在 $1.6V \leq V_{DD} \leq 5.5V$ 的工作电压范围内进行转换运行。 在低电压下进行转换运行时选择此模式。因为是以低电压运行，所以在转换时内部门会进行升压。
采样时间的选择	采样时钟数：7 个 f_{AD}	标准 1/ 低电压模式 1 的采样时间为 7 个转换时钟（ f_{AD} ）。在模拟输入源的输出阻抗高并且需要采样时间长的情况下选择此采样时钟数。
	采样时钟数：5 个 f_{AD}	标准 2/ 低电压模式 2 的采样时间为 5 个转换时钟（ f_{AD} ）。在模拟输入源的输出阻抗低等能确保足够的采样时间时选择此采样时钟数。

图 12-1 A/D 转换器的框图



12.2 A/D 转换器的结构

A/D 转换器由以下硬件构成。

(1) ANI0、ANI1、ANI16、ANI17 引脚

这些是 A/D 转换器的 4 个通道的模拟输入引脚，输入要进行 A/D 转换的模拟信号。没有被选择为模拟输入的引脚能用作输入 / 输出端口。

(2) 采样 & 保持电路

采样 & 保持电路依次对来自输入电路的模拟输入电压进行采样并且送给 A/D 电压比较器。在 A/D 转换过程中，保持采样到的模拟输入电压。

(3) A/D 电压比较器

通过 A/D 电压比较器将比较电压生成电路的分接头电压和模拟输入电压进行比较。如果比较结果是模拟输入电压大于基准电压 ($1/2AV_{REF}$)，就将逐次逼近寄存器 (SAR) 的最高位 (MSB) 置位。如果模拟输入电压小于基准电压 ($1/2AV_{REF}$)，就对 SAR 寄存器的 MSB 位进行复位。

然后，自动将 SAR 寄存器的 bit8 置位并且进行下一次比较。在此，根据 bit9 被设定的结果值，选择比较电压生成电路的分接头电压。

bit9=0: ($1/4AV_{REF}$)

bit9=1: ($3/4AV_{REF}$)

将比较电压生成电路的分接头电压和模拟输入电压进行比较，根据比较结果操作 SAR 寄存器的 bit8。

模拟输入电压 $\geq$ 比较电压生成电路的分接头电压: bit8=1

模拟输入电压 $\leq$ 比较电压生成电路的分接头电压: bit8=0

继续此比较直到 SAR 寄存器的 bit0 为止。

当以 8 位分辨率进行 A/D 转换时，继续此比较直到 SAR 寄存器的 bit2 为止。

备注 AV_{REF} 是 A/D 转换器的正 (+) 基准电压。可选择 AV_{REFP} 、内部基准电压 (1.45V) 或者 V_{DD} 。

(4) 比较电压生成电路

比较电压生成电路生成模拟输入电压的比较电压。

(5) 逐次逼近寄存器 (SAR: Successive Approximation Register)

SAR 寄存器从最高位 (MSB) 开始按位设定比较电压生成电路的分接头电压值和模拟输入引脚的电压值相同的数据。

如果设定到 SAR 寄存器的最低位 (LSB) (A/D 转换结束)，就将 SAR 寄存器的内容 (转换结果) 保持到 A/D 转换结果寄存器 (ADCR)。而且，如果指定的全部 A/D 转换都结束，就产生 A/D 转换结束中断请求信号 (INTAD)。

(6) 10 位 A/D 转换结果寄存器 (ADCR)

每当 A/D 转换结束时，就将逐次逼近寄存器的转换结果加载到 ADCR 寄存器，A/D 转换结果保持在高 10 位（低 6 位固定为“0”）。

(7) 8 位 A/D 转换结果寄存器 (ADCRH)

每当 A/D 转换结束时，就将逐次逼近寄存器的转换结果加载到 ADCRH 寄存器，保存 A/D 转换结果的高 8 位。

(8) 控制电路

此电路控制要进行 A/D 转换的模拟输入的转换时间、转换的开始或者停止等。当 A/D 转换结束时，经过 A/D 转换结果上限值 / 下限值比较电路，产生 INTAD 信号。

(9) AV_{REFP} 引脚

这是从外部输入基准电压 (AV_{REFP}) 的引脚。

要将 AV_{REFP} 用作 A/D 转换器的正 (+) 基准电压时，必须将 A/D 转换器的模式寄存器 2 (ADM2) 的 ADREFP1 位置“0”、ADREFP0 位置“1”。

根据加载的 AV_{REFP} 和负 (-) 基准电压 (AV_{REFM}/V_{SS}) 之间的电压，将 ANI0、ANI1、ANI16、ANI17 输入的模拟信号转换为数字信号。

除了 AV_{REFP} 以外，可选择 V_{DD} 或者内部基准电压 (1.45V) 作为 A/D 转换器的正 (+) 基准电压。

(10) AV_{REFM} 引脚

这是从外部输入基准电压 (AV_{REFM}) 的引脚。要将 AV_{REFM} 用作 A/D 转换器的负 (-) 基准电压时，必须将 ADM2 寄存器的 ADREFM 位置“1”。

除了 AV_{REFM} 以外，可选择 V_{SS} 作为 A/D 转换器的负 (-) 基准电压。

12.3 控制 A/D 转换器的寄存器

A/D 转换器使用以下寄存器。

- 外围允许寄存器0 (PER0)
- A/D转换器的模式寄存器0 (ADM0)
- A/D转换器的模式寄存器1 (ADM1)
- A/D转换器的模式寄存器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)
- 模拟输入通道指定寄存器 (ADS)
- 转换结果比较上限值设定寄存器 (ADUL)
- 转换结果比较下限值设定寄存器 (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2 (PMC2)
- 端口模式寄存器2 (PM2)

12.3.1 外围允许寄存器 0（PER0）

PER0 寄存器是设定允许或者禁止给各外围硬件提供时钟的寄存器。通过停止给不使用的硬件提供时钟，降低功耗和噪声。

要使用 A/D 转换器时，必须将 bit5（ADCEN）置“1”。

通过 1 位或者 8 位存储器操作指令设定 PER0 寄存器。

在产生复位信号后，此寄存器的值变为“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	RTCWEN	0	ADCEN	IICA0EN	SAU1EN	SAU0EN	0	TAU0EN

ADCEN	控制 A/D 转换器的输入时钟
0	停止提供输入时钟。 - 不能写 A/D 转换器使用的 SFR。 - A/D 转换器处于复位状态。
1	提供输入时钟。 能读写 A/D 转换器使用的 SFR。

注意 1. 在设定 A/D 转换器时，必须先在 ADCEN 位为“1”的状态下设定以下寄存器。如果 ADCEN 位为“0”，A/D 转换器的控制寄存器都为初始值，并且忽视写操作（端口模式寄存器 2（PM2）、端口模式控制寄存器 2（PMC2）以及 A/D 端口配置寄存器（ADPC）除外）。

- A/D 转换器的模式寄存器 0（ADM0）
- A/D 转换器的模式寄存器 1（ADM1）
- A/D 转换器的模式寄存器 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）
- 模拟输入通道指定寄存器（ADS）
- 转换结果比较上限值设定寄存器（ADUL）
- 转换结果比较下限值设定寄存器（ADLL）
- A/D 测试寄存器（ADTES）

2. 必须将 bit1 和 bit6 置“0”。

12.3.2 A/D 转换器的模式寄存器 0 (ADM0)

这是设定要进行 A/D 转换的模拟输入的转换时间、转换开始或者停止的寄存器。

通过 1 位或者 8 位存储器操作指令设定 ADM0 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 12-3 A/D 转换器的模式寄存器 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	0	FR2 注 1	FR1 注 1	FR0 注 1	LV1 注 1	LV0 注 1	ADCE

ADCS	A/D 转换运行的控制
0	停止转换运行。 [读时] 停止转换运行 / 待机状态
1	允许转换运行。 [读时] 软件触发模式时: 转换运行状态 硬件触发等待模式时: A/D 电源稳定等待状态 + 转换运行状态

ADCE	A/D 电压比较器的运行控制注 2
0	停止 A/D 电压比较器的运行。
1	允许 A/D 电压比较器的运行。

- 注 1. 有关 FR2 ~ FR0 位、LV1 位、LV0 位和 A/D 转换的详细内容，请参照“表 12-3 A/D 转换时间的选择”。
2. 在软件触发模式或者硬件触发无等待模式中，ADCS 位和 ADCE 位控制 A/D 电压比较器的运行，从开始运行到稳定需要 1μs。因此，在将 ADCE 位置“1”后至少经过 1μs，然后将 ADCS 位置“1”，从而从第 1 个转换数据开始有效。如果不至少等待 1μs 就将 ADCS 位置“1”，则忽视第 1 个转换数据。

注意 1. 必须在转换停止状态 (ADCS=0、ADCE=0) 下更改 FR2 ~ FR0 位、LV1 位和 LV0 位。

- 禁止进行 ADCS=1、ADCE=0 的设定。
- 禁止通过 8 位操作指令将 ADCS=0、ADCE=0 的状态设定为 ADCS=1、ADCE=1。必须按照“12.7 A/D 转换器的设定流程图”的步骤进行设定。
- 必须将 bit6 设定为“0”。

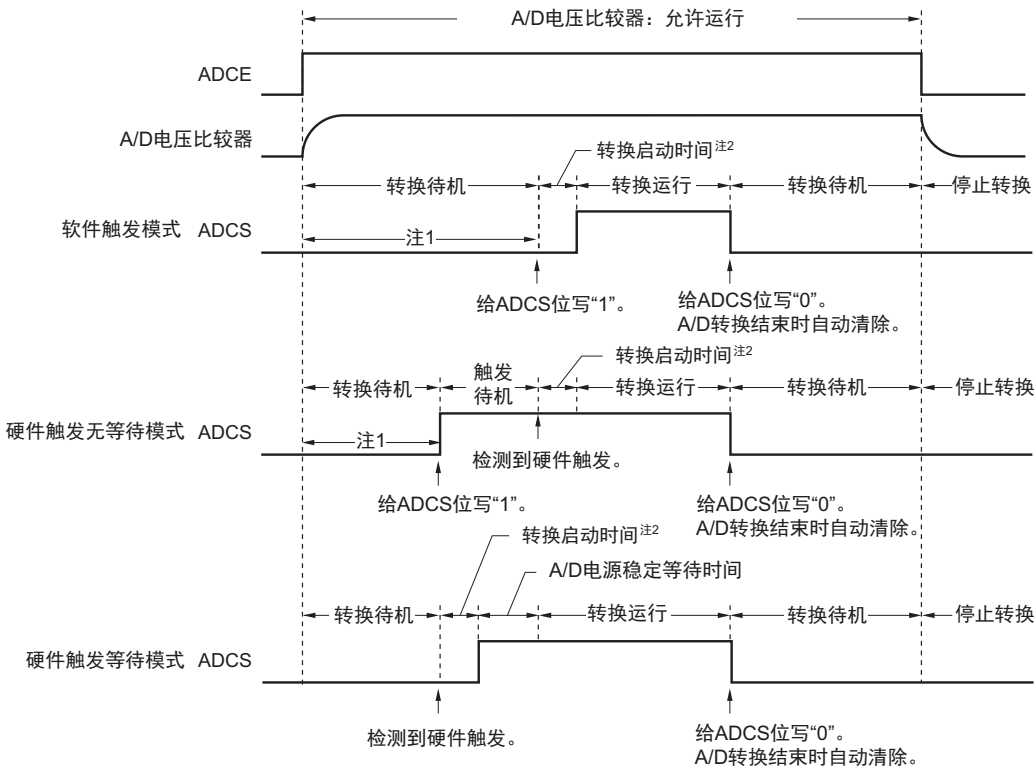
表 12-1 ADCS 位和 ADCE 位的设定

ADCS	ADCE	A/D 转换运行
0	0	转换停止状态
0	1	转换待机状态
1	0	禁止设定。
1	1	转换运行状态

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	连续转换模式	当给 ADCS 位写“1”时	当给 ADCS 位写“0”时
	单次转换模式		- 当给 ADCS 位写“0”时 - 在 A/D 转换结束时自动清“0”。
硬件触发无等待模式	连续转换模式		当给 ADCS 位写“0”时
	单次转换模式		当给 ADCS 位写“0”时
硬件触发等待模式	连续转换模式	当输入硬件触发时	当给 ADCS 位写“0”时
	单次转换模式		- 当给 ADCS 位写“0”时 - 在 A/D 转换结束时自动清“0”。

图 12-4 使用 A/D 电压比较器时的时序图



- 注 1. 在软件触发模式和硬件触发无等待模式中，为了稳定内部电路，从 ADCE 位上升到 ADCS 位上升的时间至少需要 1 μ s。
2. 在启动转换时最长需要以下的时间：

ADM0			转换时钟 (f_{AD})	启动时间 (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，从连续转换模式的第 2 次转换开始，在检测到硬件触发后不产生转换启动时间和 A/D 电源稳定等待时间。

- 注意 1. 要使用硬件触发等待模式时，禁止将 ADCS 位置“1”（当检测到硬件触发信号时，自动切换为“1”）。但是，为了设定为 A/D 转换待机状态，能将 ADCS 位置“0”。
2. 在硬件触发无等待模式的单次转换模式中，ADCS 标志在 A/D 转换结束时不自动清“0”，而保持“1”的状态。
3. 必须在 ADCS 位为“0”（停止转换 / 转换待机状态）时改写 ADCE 位。
4. 为了结束 A/D 转换，必须至少将硬件触发间隔设定为以下时间：
硬件触发无等待模式时：2 个 f_{CLK} 时钟 + 转换启动时间 + A/D 转换时间
硬件触发等待模式时：2 个 f_{CLK} 时钟 + 转换启动时间 + A/D 电源稳定等待时间 + A/D 转换时间

备注 f_{CLK} ：CPU/ 外围硬件的时钟频率

表 12-3 A/D 转换时间的选择 (1/4)

(1) 无 A/D 电源稳定等待时间 标准模式 1、2 (软件触发模式 / 硬件触发无等待模式)

A/D 转换器的模式寄存器 0 (ADM0)					模式	转换时钟 (f _{AD})	转换 时钟数注	转换时间	10 位分辨率时的转换时间						
FR2	FR1	FR0	LV1	LV0					2.7V ≤ V _{DD} ≤ 5.5V						
									f _{CLK} = 1MHz	f _{CLK} = 4MHz	f _{CLK} = 8MHz	f _{CLK} = 16MHz	f _{CLK} = 24MHz		
0	0	0	0	0	标准 1	f _{CLK} /64	19 个 f _{AD} (采样时 钟数: 7 个 f _{AD})	1216/f _{CLK}	禁止设定	禁止设定	禁止设定	76μs	50.6667μs		
0	0	1				f _{CLK} /32		608/f _{CLK}				76μs	38μs	25.3333μs	
0	1	0				f _{CLK} /16		304/f _{CLK}				76μs	38μs	19μs	12.6667μs
0	1	1				f _{CLK} /8		152/f _{CLK}				38μs	19μs	9.5μs	6.3333μs
1	0	0				f _{CLK} /6		114/f _{CLK}				28.5μs	14.25μs	7.125μs	4.75μs
1	0	1				f _{CLK} /5		95/f _{CLK}	95μs	23.75μs	11.875μs	5.938μs	3.9583μs		
1	1	0				f _{CLK} /4		76/f _{CLK}	76μs	19μs	9.5μs	4.75μs	3.1667μs		
1	1	1				f _{CLK} /2		38/f _{CLK}	38μs	9.5μs	4.75μs	2.375μs	禁止设定		
0	0	0	0	1	标准 2	f _{CLK} /64	17 个 f _{AD} (采样时 钟数: 5 个 f _{AD})	1088/f _{CLK}	禁止设定	禁止设定	禁止设定	68μs	45.3333μs		
0	0	1				f _{CLK} /32		544/f _{CLK}				68μs	34μs	22.6667μs	
0	1	0				f _{CLK} /16		272/f _{CLK}				68μs	34μs	17μs	11.3333μs
0	1	1				f _{CLK} /8		136/f _{CLK}				34μs	17μs	8.5μs	5.6667μs
1	0	0				f _{CLK} /6		102/f _{CLK}				25.5μs	12.75μs	6.375μs	4.25μs
1	0	1				f _{CLK} /5		85/f _{CLK}	85μs	21.25μs	10.625μs	5.3125μs	3.5417μs		
1	1	0				f _{CLK} /4		68/f _{CLK}	68μs	17μs	8.5μs	4.25μs	2.8333μs		
1	1	1				f _{CLK} /2		34/f _{CLK}	34μs	8.5μs	4.25μs	2.125μs	禁止设定		

注 这是 10 位分辨率时的转换时钟数。如果选择 8 位分辨率，转换时钟数就会减少 2 个转换时钟 (f_{AD})。

注意 1. 必须在“32.6.1 A/D 转换器特性”中所示的转换时间 (t_{CONV}) 范围内使用 A/D 转换时间。

2. 要将 FR2 ~ FR0 位、LV1 位和 LV0 位改写为不同数据时，必须在停止转换的状态下 (ADCS=0、ADCE=0) 进行。

3. 转换时间不包含转换启动时间，因此在第 1 次转换时必须加上转换启动时间。另外，转换时间不包含时钟频率的误差，因此在选择转换时间时必须考虑时钟频率的误差。

备注 f_{CLK} : CPU/ 外围硬件的时钟频率

表 12-3 A/D 转换时间的选择 (2/4)

(2) 无 A/D 电源稳定等待时间 低电压模式 1、2 (软件触发模式 / 硬件触发无等待模式)

A/D 转换器的模式寄存器 0 (ADM0)					模式	转换时钟 (f _{AD})	转换 时钟数 ^{注 4}	转换时间	10 位分辨率时的转换时间						
FR2	FR1	FR0	LV1	LV0					1.6V ≤ V _{DD} ≤ 5.5V		注 1	注 2	注 3		
									f _{CLK} = 1MHz	f _{CLK} = 4MHz	f _{CLK} = 8MHz	f _{CLK} = 16MHz	f _{CLK} = 24MHz		
0	0	0	1	0	低电压1	f _{CLK} /64	19 个 f _{AD} (采样时 钟数: 7 个 f _{AD})	1216/f _{CLK}	禁止设定	禁止设定	禁止设定	76μs	50.6667μs		
0	0	1				f _{CLK} /32		608/f _{CLK}			76μs	38μs	25.3333μs		
0	1	0				f _{CLK} /16		304/f _{CLK}			76μs	38μs	19μs	12.6667μs	
0	1	1				f _{CLK} /8		152/f _{CLK}			38μs	19μs	9.5μs	6.3333μs	
1	0	0				f _{CLK} /6		114/f _{CLK}			28.5μs	14.25μs	7.125μs	4.75μs	
1	0	1				f _{CLK} /5		95/f _{CLK}			95μs	23.75μs	11.875μs	5.938μs	3.9587μs
1	1	0				f _{CLK} /4		76/f _{CLK}			76μs	19μs	9.5μs	4.75μs	3.1667μs
1	1	1				f _{CLK} /2		38/f _{CLK}			38μs	9.5μs	4.75μs	2.375μs	禁止设定
0	0	0	1	1	低电压2	f _{CLK} /64	17 个 f _{AD} (采样时 钟数: 5 个 f _{AD})	1088/f _{CLK}	禁止设定	禁止设定	禁止设定	68μs	45.3333μs		
0	0	1				f _{CLK} /32		544/f _{CLK}			68μs	34μs	22.6667μs		
0	1	0				f _{CLK} /16		272/f _{CLK}			68μs	34μs	17μs	11.3333μs	
0	1	1				f _{CLK} /8		136/f _{CLK}			34μs	17μs	8.5μs	5.6667μs	
1	0	0				f _{CLK} /6		102/f _{CLK}			25.5μs	12.75μs	6.375μs	4.25μs	
1	0	1				f _{CLK} /5		85/f _{CLK}			85μs	21.25μs	10.625μs	5.3125μs	3.5417μs
1	1	0				f _{CLK} /4		68/f _{CLK}			68μs	17μs	8.5μs	4.25μs	2.8333μs
1	1	1				f _{CLK} /2		34/f _{CLK}			34μs	8.5μs	4.25μs	2.125μs	禁止设定

注 1. $1.8V \leq V_{DD} \leq 5.5V$ 2. $2.4V \leq V_{DD} \leq 5.5V$ 3. $2.7V \leq V_{DD} \leq 5.5V$ 4. 这是 10 位分辨率时的转换时钟数。如果选择 8 位分辨率，转换时钟数就会减少 2 个转换时钟 (f_{AD})。注意 1. 必须在“32.6.1 A/D 转换器特性”中所示的转换时间 (t_{CONV}) 范围内使用 A/D 转换时间。

2. 要将 FR2 ~ FR0 位、LV1 位和 LV0 位改写为不同数据时，必须在停止转换的状态下 (ADCS=0、ADCE=0) 进行。

3. 转换时间不包含转换启动时间，因此在第 1 次转换时必须加上转换启动时间。另外，转换时间不包含时钟频率的误差，因此在选择转换时间时必须考虑时钟频率的误差。

备注 f_{CLK} : CPU/ 外围硬件的时钟频率

表 12-3 A/D 转换时间的选择 (3/4)

(3) 有 A/D 电源稳定等待时间 标准模式 1、2 (硬件触发等待模式注 1)

A/D 转换器的模式寄存器 0 (ADM0)					模式	转换 时钟 (f _{AD})	稳定 等待 时钟数	转换 时钟数注 2	稳定等待 时间 + 转换时间	稳定等待时间 +10 位分辨率时的转换时间							
FR2	FR1	FR0	LV1	LV0						2.7V ≤ V _{DD} ≤ 5.5V							
										f _{CLK} = 1MHz	f _{CLK} = 4MHz	f _{CLK} = 8MHz	f _{CLK} = 16MHz	f _{CLK} = 24MHz			
0	0	0	0	0	标准 1	f _{CLK} /64	8 个 f _{AD}	19 个 f _{AD} (采样时 钟数: 7 个 f _{AD})	1728/f _{CLK}	禁止设定	禁止设定	禁止设定	108μs	72μs			
0	0	1				f _{CLK} /32			864/f _{CLK}				108μs	54μs	36μs		
0	1	0				f _{CLK} /16			432/f _{CLK}				108μs	54μs	27μs	18μs	
0	1	1				f _{CLK} /8			216/f _{CLK}				54μs	27μs	13.5μs	9μs	
1	0	0				f _{CLK} /6			162/f _{CLK}				40.5μs	20.25μs	10.125μs	6.75μs	
1	0	1				f _{CLK} /5			135/f _{CLK}				135μs	33.75μs	16.875μs	8.4375μs	5.625μs
1	1	0				f _{CLK} /4			108/f _{CLK}				108μs	27μs	13.5μs	6.75μs	4.5μs
1	1	1				f _{CLK} /2			54/f _{CLK}				54μs	13.5μs	6.75μs	3.375μs	禁止设定
0	0	0	0	1	标准 2	f _{CLK} /64	8 个 f _{AD}	17 个 f _{AD} (采样时 钟数: 5 个 f _{AD})	1600/f _{CLK}	禁止设定	禁止设定	禁止设定	100μs	66.6667μs			
0	0	1				f _{CLK} /32			800/f _{CLK}				100μs	50μs	33.3333μs		
0	1	0				f _{CLK} /16			400/f _{CLK}				100μs	50μs	25μs	16.6667μs	
0	1	1				f _{CLK} /8			200/f _{CLK}				50μs	25μs	12.5μs	8.3333μs	
1	0	0				f _{CLK} /6			150/f _{CLK}				37.5μs	18.75μs	9.375μs	6.25μs	
1	0	1				f _{CLK} /5			125/f _{CLK}				125μs	31.25μs	15.625μs	7.8125μs	5.2083μs
1	1	0				f _{CLK} /4			100/f _{CLK}				100μs	25μs	12.5μs	6.25μs	4.1667μs
1	1	1				f _{CLK} /2			50/f _{CLK}				50μs	12.5μs	6.25μs	3.125μs	禁止设定

注 1. 从连续转换模式的第 2 次转换开始, 在检测到硬件触发后不产生转换启动时间和 A/D 电源稳定等待时间 (参照表 12-3 (1/4))。

2. 这是 10 位分辨率时的转换时钟数。如果选择 8 位分辨率, 转换时钟数就会减少 2 个转换时钟 (f_{AD})。

注意 1. 必须在“32.6.1 A/D 转换器特性”中所示的转换时间 (t_{CONV}) 范围内使用 A/D 转换时间。

2. 要将 FR2 ~ FR0 位、LV1 位和 LV0 位改写为不同数据时, 必须在停止转换的状态下 (ADCS=0、ADCE=0) 进行。

3. 转换时间不包含转换启动时间, 因此在第 1 次转换时必须加上转换启动时间。另外, 转换时间不包含时钟频率的误差, 因此在选择转换时间时必须考虑时钟频率的误差。

4. 硬件触发等待模式中的转换时间包含检测到硬件触发后的稳定等待时间。

备注 f_{CLK} : CPU/ 外围硬件的时钟频率

表 12-3 A/D 转换时间的选择 (4/4)

(4) 有 A/D 电源稳定等待时间 低电压模式 1、2 (硬件触发等待模式注 1)

A/D 转换器的模式寄存器 0 (ADM0)					模式	转换时钟 (f _{AD})	稳定等待时钟数	转换时钟数注 5	稳定等待时间 + 转换时间	稳定等待时间 + 10 位分辨率时的转换时间				
FR2	FR1	FR0	LV1	LV0						1.6V ≤ V _{DD} ≤ 5.5V		注 2	注 3	注 4
										f _{CLK} =1MHz	f _{CLK} =4MHz	f _{CLK} =8MHz	f _{CLK} =16MHz	f _{CLK} =24MHz
0	0	0	1	0	低电压1	f _{CLK} /64	2 个 f _{AD}	19 个 f _{AD} (采样时钟数: 7 个 f _{AD})	1344/f _{CLK}	禁止设定	禁止设定	禁止设定	84μs	56μs
0	0	1				f _{CLK} /32			672/f _{CLK}				84μs	42μs
0	1	0				f _{CLK} /16			336/f _{CLK}				84μs	21μs
0	1	1				f _{CLK} /8			168/f _{CLK}				42μs	10.5μs
1	0	0				f _{CLK} /6			126/f _{CLK}				31.25μs	7.875μs
1	0	1				f _{CLK} /5			105/f _{CLK}	105μs	26.25μs	13.125μs	6.5625μs	4.375μs
1	1	0				f _{CLK} /4			84/f _{CLK}	84μs	21μs	10.5μs	5.25μs	3.5μs
1	1	1				f _{CLK} /2			42/f _{CLK}	42μs	10.5μs	5.25μs	2.625μs	禁止设定
0	0	0	1	1	低电压2	f _{CLK} /64	2 个 f _{AD}	17 个 f _{AD} (采样时钟数: 5 个 f _{AD})	1216/f _{CLK}	禁止设定	禁止设定	禁止设定	76μs	50.6667μs
0	0	1				f _{CLK} /32			608/f _{CLK}				76μs	38μs
0	1	0				f _{CLK} /16			304/f _{CLK}				76μs	19μs
0	1	1				f _{CLK} /8			152/f _{CLK}				38μs	9.5μs
1	0	0				f _{CLK} /6			114/f _{CLK}				28.5μs	7.125μs
1	0	1				f _{CLK} /5			96/f _{CLK}	96μs	23.75μs	12μs	5.938μs	4.0μs
1	1	0				f _{CLK} /4			76/f _{CLK}	76μs	19μs	9.5μs	4.75μs	3.1667μs
1	1	1				f _{CLK} /2			38/f _{CLK}	38μs	9.5μs	4.75μs	2.375μs	禁止设定

注 1. 从连续转换模式的第 2 次转换开始, 在检测到硬件触发后不产生转换启动时间和 A/D 电源稳定等待时间 (参照表 12-3 (2/4))。

2. $1.8V \leq V_{DD} \leq 5.5V$

3. $2.4V \leq V_{DD} \leq 5.5V$

4. $2.7V \leq V_{DD} \leq 5.5V$

5. 这是 10 位分辨率时的转换时钟数。如果选择 8 位分辨率, 转换时钟数就会减少 2 个转换时钟 (f_{AD})。

注意 1. 必须在“32.6.1 A/D 转换器特性”中所示的转换时间 (t_{CONV}) 范围内使用 A/D 转换时间。

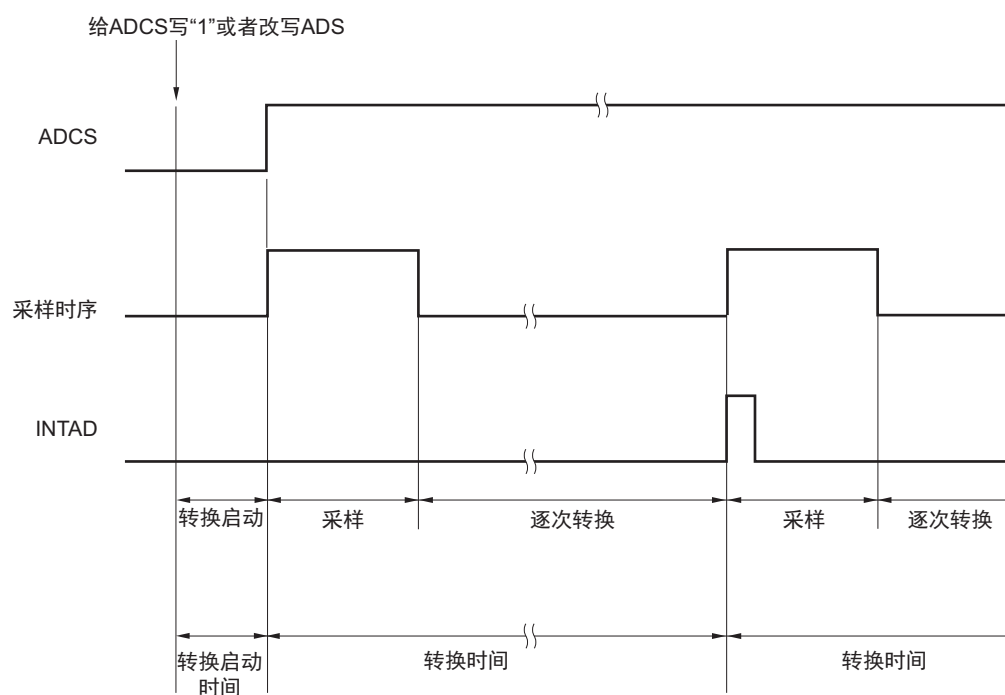
2. 要将 FR2 ~ FR0 位、LV1 位和 LV0 位改写为不同数据时, 必须在停止转换的状态下 (ADCS=0、ADCE=0) 进行。

3. 转换时间不包含转换启动时间, 因此在第 1 次转换时必须加上转换启动时间。另外, 转换时间不包含时钟频率的误差, 因此在选择转换时间时必须考虑时钟频率的误差。

4. 硬件触发等待模式中的转换时间包含检测到硬件触发后的稳定等待时间。

备注 f_{CLK}: CPU/ 外围硬件的时钟频率

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）



12.3.3 A/D 转换器的模式寄存器 1（ADM1）

这是设定 A/D 转换触发、转换模式和硬件触发信号的寄存器。
通过 1 位或者 8 位存储器操作指令设定 ADM1 寄存器。
在产生复位信号后，此寄存器的值变为“00H”。

图 12-6 A/D 转换器的模式寄存器 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	×	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 转换运行模式的设定
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 01 的计数结束或者捕捉结束中断信号（INTTM01）
0	1	禁止设定
1	0	实时时钟 2 中断信号（INTRTC）
1	1	12 位间隔定时器中断信号（INTIT）

- 注意 1. 要改写 ADM1 寄存器时，必须在处于转换停止状态（A/D 转换器的模式寄存器 0（ADM0）的 ADCS=0）时进行。
2. 为了结束 A/D 转换，必须至少将硬件触发间隔设定为以下时间：
硬件触发无等待模式时：2 个 f_{CLK} 时钟 + 转换启动时间 + A/D 转换时间
硬件触发等待模式时：2 个 f_{CLK} 时钟 + 转换启动时间 + A/D 电源稳定等待时间 + A/D 转换时间
3. 在 SNOOZE 功能以外的模式中，在输入 INTRTC、INTIT 后的最多 4 个 f_{CLK} 时钟期间内，下一个 INTRTC 和 INTIT 的输入触发无效。

备注 1. ×：忽略
2. f_{CLK} ：CPU/ 外围硬件时钟频率

12.3.4 A/D 转换器的模式寄存器 2（ADM2）

这是选择 A/D 转换器的基准电压和分辨率并且检查 A/D 转换结果的上限值和下限值以及设定 SNOOZE 模式的寄存器。

通过 1 位或者 8 位存储器操作指令设定 ADM2 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 12-7 A/D 转换器的模式寄存器 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 转换器的正（+）基准电压的选择
0	0	由 V _{DD} 提供。
0	1	由 P21/AV _{REFP} /ANI0 提供。
1	0	由内部基准电压（1.45V）提供注。
1	1	禁止设定。
• 要改写 ADREFP1 位和 ADREFP0 位时，必须按照以下步骤进行设定： ①将 ADCE 位置“0”。 ②更改 ADREFP1 位和 ADREFP0 位的值。 ③等待基准电压稳定等待时间（A）。 ④将 ADCE 位置“1”。 ⑤等待基准电压稳定等待时间（B）。 当将 ADREFP1 位和 ADREFP0 位改为“1”和“0”时，A=5μs，B=1μs。 当将 ADREFP1 位和 ADREFP0 位改为“0”和“0”或者“0”和“1”时，A 不需等待，B=1μs。 必须在等待⑤的稳定时间后开始 A/D 转换。 • 当将 ADREFP1 位和 ADREFP0 位置“1”和“0”时，不能对温度传感器输出电压和内部基准电压（1.45V）进行 A/D 转换。 必须在 ADISS 位为“0”时进行 A/D 转换。		

ADREFM	A/D 转换器的负（-）基准电压的选择
0	由 V _{SS} 提供。
1	由 P20/AV _{REFM} /ANI1 提供。

注 只有在 HS（高速主）模式中才能选择。

- 注意 1. 要改写 ADM2 寄存器时，必须在处于转换停止状态（ADCS=0、ADCE=0）时进行。
2. 要转移到 STOP 模式时或者要在 CPU 以副系统时钟运行中转移到 HALT 模式时，不能将 ADREFP1 位置“1”。在选择内部基准电压（ADREFP1、ADREFP0=1、0）时，加上“32.3.2 电源电流特性”所示的 A/D 转换器基准电压电流（I_{ADREF}）的电流值。
3. 当使用 AV_{REFP}和 AV_{REFM}时，必须将 ANI0和 ANI1设定为模拟输入通道，并且通过端口模式寄存器设定为输入模式。

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	转换结果上限值和下限值的检查
0	当 ADLL 寄存器≤ ADCR 寄存器≤ ADUL 寄存器（AREA1）时，产生中断信号（INTAD）。
1	当 ADCR 寄存器< ADLL 寄存器（AREA2）或者 ADUL 寄存器< ADCR 寄存器（AREA3）时，产生中断信号（INTAD）。
AREA1～AREA3 的中断信号（INTAD）的产生范围如图 12-8 所示。	

AWC	SNOOZE 模式的设定
0	不使用 SNOOZE 模式功能。
1	使用 SNOOZE 模式功能。

通过 STOP 模式中的硬件触发信号来解除 STOP 模式，并且在 CPU 不运行的情况下进行 A/D 转换（SNOOZE 模式）。

- 只有在选择高速内部振荡器时钟作为 CPU/ 外围硬件的时钟（f_{CLK}）的情况下才能设定 SNOOZE 模式功能。在选择高速内部振荡器以外的时钟的情况下，禁止设定。
- 在软件触发模式或者硬件触发无等待模式中，禁止使用 SNOOZE 模式功能。
- 在连续转换模式中，禁止使用 SNOOZE 模式功能。
- 在使用 SNOOZE 模式功能时，必须至少将硬件触发间隔设定为“SNOOZE 模式的转移时间注 + 转换启动时间 + A/D 电源稳定等待时间 + A/D 转换时间 + 2 个 f_{CLK} 时钟”。
- 即使使用 SNOOZE 功能，也必须在通常运行时将 AWC 位置“0”，并且在即将要转移到 STOP 模式前将 AWC 位改为“1”。

另外，必须在从 STOP 模式返回到通常运行后将 AWC 位改为“0”。

如果 AWC 位保持“1”的状态，就无法正常开始 A/D 转换，而与 SNOOZE 模式和通常运行无关。

ADTYP	A/D 转换分辨率的选择
0	10 位分辨率
1	8 位分辨率

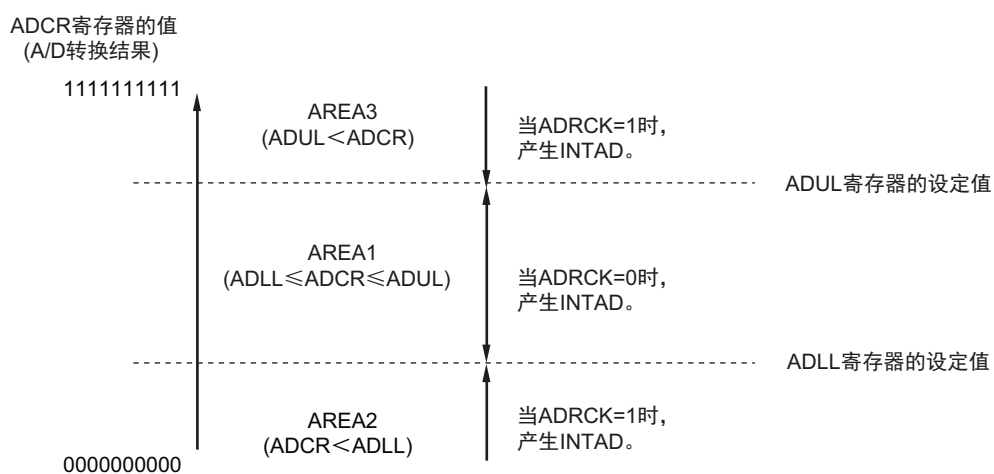
注

请参照“21.3.3 SNOOZE 模式”的“STOP 模式 → SNOOZE 模式的转移时间”。

注意

要改写 ADM2 寄存器时，必须在处于转换停止状态（ADCS=0、ADCE=0）时进行。

图 12-8 ADRCK 位中断信号产生范围



注意 当不发生 INTAD 时，A/D 转换结果不保存到 ADCR 寄存器和 ADCRH 寄存器。

12.3.5 10 位 A/D 转换结果寄存器 (ADCR)

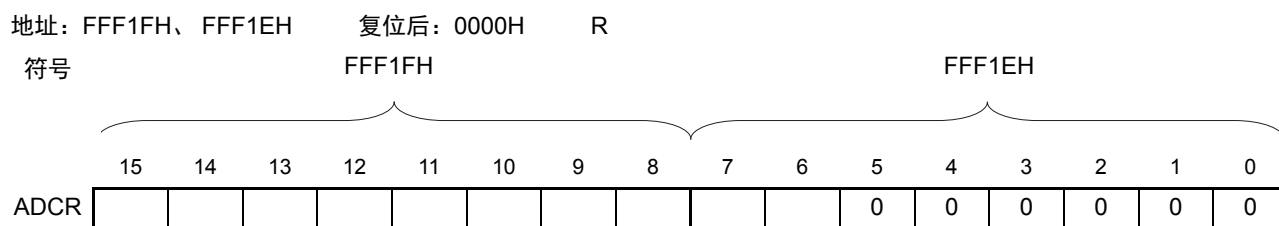
这是保存 A/D 转换结果的 16 位寄存器，低 6 位固定为“0”。每当 A/D 转换结束时，就从逐次逼近寄存器 (SAR) 加载转换结果。转换结果的高 8 位保存到 FFF1FH，低 2 位保存到 FFF1EH 的高 2 位注。

通过 16 位存储器操作指令读 ADCR 寄存器。

在产生复位信号后，此寄存器的值变为“0000H”。

注 如果 A/D 转换结果的值超出 A/D 转换结果比较功能（通过 ADRCK 位和 ADUL/ADLL 寄存器进行设定（参照图 12-8））的设定值范围，就不保存 A/D 转换结果。

图 12-9 10 位 A/D 转换结果寄存器 (ADCR) 的格式



注意 1. 如果在选择 8 位分辨率 A/D 转换（A/D 转换器的模式寄存器 2 (ADM2) 的 ADTYP=1）时读 ADCR 寄存器，低 2 位（ADCR 寄存器的 bit7 和 bit6）的读取值就为“0”。

2. 当对 ADCR 寄存器进行 16 位存取时，能从 ADCR 寄存器的 bit15 依次读转换结果的高 10 位。

12.3.6 8 位 A/D 转换结果寄存器 (ADCRH)

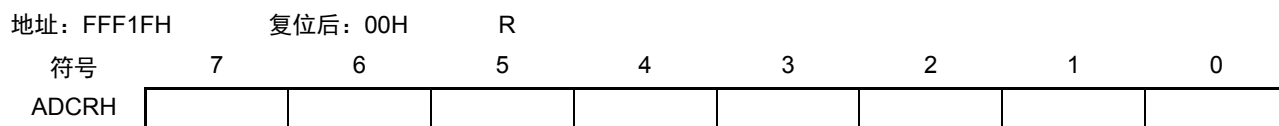
这是保存 A/D 转换结果的 8 位寄存器，保存 10 位分辨率的高 8 位注。

通过 8 位存储器操作指令读 ADCRH 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

注 如果 A/D 转换结果的值超出 A/D 转换结果比较功能（通过 ADRCK 位和 ADUL/ADLL 寄存器进行设定（参照图 12-8））的设定值范围，就不保存 A/D 转换结果。

图 12-10 8 位 A/D 转换结果寄存器 (ADCRH) 的格式



注意 当写 A/D 转换器的模式寄存器 0 (ADM0)、模拟输入通道指定寄存器 (ADS) 和 A/D 端口配置寄存器 (ADPC) 时，ADCRH 寄存器的内容可能不定。必须在转换结束后并且在写 ADM0、ADS、ADPC 寄存器前读转换结果。否则，就可能读不到正确的转换结果。

12.3.7 模拟输入通道指定寄存器（ADS）

这是指定要进行 A/D 转换的模拟电压输入通道的寄存器。

通过 1 位或者 8 位存储器操作指令设定 ADS 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 12-11 模拟输入通道指定寄存器（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

ADISS	ADS4	ADS3	ADS2	ADS1	ADS0	模拟输入通道	输入源
0	0	0	0	0	0	ANI0	P21/ANI0/AV _{REFP} 引脚
0	0	0	0	0	1	ANI1	P20/ANI1/AV _{REFM} 引脚
0	1	0	0	0	0	ANI16	P22/ANI16 引脚
0	1	0	0	0	1	ANI17	P23/ANI17 引脚
1	0	0	0	0	0	—	温度传感器输出注
1	0	0	0	0	1	—	内部基准电压 (1.45 V) 注
其他						禁止设定	

注 只有在 HS（高速主）模式中才能选择。

注意 1. 必须将 bit5 和 bit6 置“0”。

- 对于由 ADPC 寄存器和 PMCx 寄存器设定为模拟输入的端口，必须通过端口模式寄存器 2（PM2）设定为输入模式。
- 对于由 A/D 端口配置寄存器（ADPC）设定为数字输入 / 输出的引脚，不能通过 ADS 寄存器进行设定。
- 对于由端口模式控制寄存器 2（PMC2）设定为数字输入 / 输出的引脚，不能通过 ADS 寄存器进行设定。
- 必须在 A/D 电压转换器运行停止状态（A/D 转换器模式寄存器 0（ADM0）的 ADCE=0）下改写 ADISS 位。
- 当将 AV_{REFP} 用作 A/D 转换器的正（+）基准电压时，不能选择 ANI0 作为 A/D 转换通道。
- 当将 AV_{REFM} 用作 A/D 转换器的负（-）基准电压时，不能选择 ANI1 作为 A/D 转换通道。
- 在将 ADISS 位置“1”的情况下，不能将内部基准电压（1.45V）用作正（+）基准电压。
另外，在将 ADISS 位置“1”后，不能使用第一次转换的转换结果。详细内容请参照“12.7.4 使用温度传感器时的设定”。
- 要转移到 STOP 模式时或者要在 CPU 以副系统时钟运行中转移到 HALT 模式时，不能将 ADISS 位置“1”。如果将 ADISS 位置“1”，就加上“32.3.2 电源电流特性”所示的 A/D 转换器基准电压电流（I_{ADREF}）的电流值。
- 当产品中不存在对应的 ANI 引脚时，忽视转换结果。

12.3.8 转换结果比较上限值设定寄存器（ADUL）

这是设定用于检查 A/D 转换结果上限值的寄存器。

将 A/D 转换结果和 ADUL 寄存器的值进行比较，并且在 A/D 转换器的模式寄存器 2（ADM2）的 ADRCCK 位的设定范围内（参照图 12-8）控制中断信号（INTAD）的产生。

通过 8 位存储器操作指令设定 ADUL 寄存器。

在产生复位信号后，此寄存器的值变为“FFH”。

注意 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定寄存器（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

12.3.9 转换结果比较下限值设定寄存器（ADLL）

这是设定用于检查 A/D 转换结果下限值的寄存器。

将 A/D 转换结果和 ADLL 寄存器的值进行比较，并且在 A/D 转换器的模式寄存器 2（ADM2）的 ADRCCK 位的设定范围内（参照图 12-8）控制中断信号（INTAD）的产生。

通过 8 位存储器操作指令设定 ADLL 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 12-13 转换结果比较下限值设定寄存器（ADLL）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 1. 当选择 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。

2. 必须在停止转换的状态（ADCS=0、ADCE=0）下改写 ADUL 寄存器和 ADLL 寄存器。

3. ADUL 寄存器和 ADLL 寄存器必须设定为 ADUL > ADLL。

12.3.10 A/D 测试寄存器（ADTES）

这是选择 A/D 转换器的正（+）基准电压、负（-）基准电压、模拟输入通道（ANLxx）、温度传感器输出电压和内部基准电压（1.45V）作为 A/D 转换对象的寄存器。

用作 A/D 测试功能时，必须如下设定。

- 测量零刻度时，选择负（-）基准电压作为 A/D 转换对象
- 测量满刻度时，选择正（+）基准电压作为 A/D 转换对象

通过 8 位存储器操作指令设定 ADTES 寄存器。

在产生复位信号后，此寄存器的值变为“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 转换对象
0	0	ANLxx/ 温度传感器输出电压注 / 内部基准电压（1.45V）注 （通过模拟输入通道指定寄存器（ADS）进行设定）
1	0	负（-）基准电压（通过 ADM2 寄存器的 ADREFM 位选择）
1	1	正（+）基准电压（通过 ADM2 寄存器的 ADREFP1 和 ADREFP0 位选择）
上述以外		禁止设定。

注 只能在 HS（高速主）模式中选择温度传感器输出电压和内部基准电压（1.45V）。

12.3.11 模拟输入引脚的端口功能控制寄存器

必须设定与 A/D 转换器的模拟输入复用的端口功能的控制寄存器（端口模式寄存器（PMxx）、端口模式控制寄存器（PMCxx）、A/D 端口配置寄存器（ADPC））。详细内容请参照“4.3.1 端口模式寄存器（PMxx）”、“4.3.6 端口模式控制寄存器（PMCxx）”、“4.3.7 A/D 端口配置寄存器（ADPC）”。

将 ANI0、ANI1 引脚用作 A/D 转换器的模拟输入时，必须将各端口对应的端口模式寄存器（PMxx）的位置“1”，并且通过 A/D 端口配置寄存器（ADPC）设定为模拟输入。

将 ANI16、ANI17 引脚用作 A/D 转换器的模拟输入时，必须将各端口对应的端口模式寄存器（PMxx）和端口模式控制寄存器（PMCxx）的位置“1”。

12.4 A/D 转换器的转换运行

A/D 转换器的转换运行如下所示。

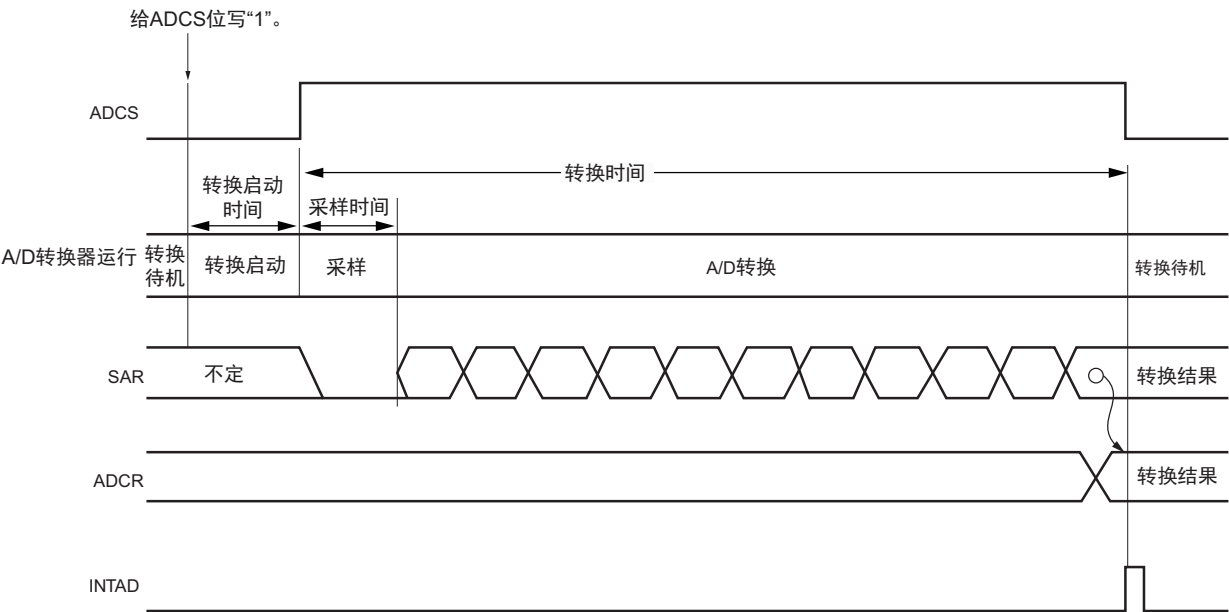
- ① 通过采样 & 保持电路对所选模拟输入通道的输入电压进行采样。
- ② 如果进行了一定时间的采样，采样 & 保持电路就为保持状态，并且保持采样电压，直到 A/D 转换结束为止。
- ③ 将逐次逼近寄存器（SAR）的 bit9 置位，并且通过分接头选择器将串联电阻串的分接头电压设定为 $(1/2)AV_{REF}$ 。
- ④ 通过 A/D 电压比较器将串联电阻串的分接头电压和采样电压的电压差进行比较。如果模拟输入大于 $(1/2)AV_{REF}$ ，SAR 寄存器的 MSB 位就保持被置位的状态。如果模拟输入小于 $(1/2)AV_{REF}$ ，就对 MSB 位进行复位。
- ⑤ 然后，自动将 SAR 寄存器的 bit8 置位并且进行下一次比较。在此，根据已被设定的 bit9 的结果值选择串联电阻串的分接头电压。
 - bit9=1: $(3/4)AV_{REF}$
 - bit9=0: $(1/4)AV_{REF}$
 将此分接头电压和采样电压进行比较，根据比较结果如下操作 SAR 寄存器的 bit8。
 - 采样电压 $\geq$ 分接头电压: bit8=1
 - 采样电压 $<$ 分接头电压: bit8=0
- ⑥ 继续此比较直到 SAR 寄存器的 bit0 为止。
- ⑦ 当 10 位比较结束时，SAR 寄存器保存了有效的数字结果，并且将该值传送到 A/D 转换结果寄存器（ADCR、ADCRH），而且被锁存注 1。
同时，能产生 A/D 转换结束中断请求（INTAD）注 1。
- ⑧ 重复①～⑦的运行，直到 ADCS 位变为“0”注 2。
要停止 A/D 转换器时，必须将 ADCS 位置“0”。

- 注 1. 如果 A/D 转换结果的值超出 A/D 转换结果比较功能（通过 ADRCK 位和 ADUL/ADLL 寄存器进行设定（参照图 12-8））的设定值范围，就不产生 A/D 转换结束中断请求信号（INTAD），并且 ADCR 寄存器和 ADCRH 寄存器不保存 A/D 转换结果。
2. 在连续转换模式中，ADCS 标志不自动清“0”。即使在硬件触发无等待模式的单次转换模式中，ADCS 标志也不自动清“0”而保持“1”的状态。

备注 1. 有 2 种 A/D 转换结果寄存器。

- ADCR 寄存器（16 位）：保存 10 位的 A/D 转换值。
 - ADCRH 寄存器（8 位）：保存 8 位的 A/D 转换值。
2. AV_{REF} 是 A/D 转换器的正（+）基准电压，可选择 AV_{REFP} 、内部基准电压（1.45V）或者 V_{DD} 。

图 12-15 A/D 转换器的转换运行（软件触发模式的情况）



在单次转换模式中进行 A/D 转换运行时，如果 A/D 转换结束，ADCS 位就自动清“0”。

连续进行连续转换模式中的 A/D 转换直到通过软件将 A/D 转换器的模式寄存器 0（ADM0）的 bit7（ADCS）置“0”。

在 A/D 转换过程中，如果改写或者重写模拟输入通道指定寄存器（ADS），就中止正在执行的转换运行，并且执行通过 ADS 寄存器重新指定的模拟输入的 A/D 转换。废弃转换途中的数据。

在产生复位信号后，A/D 转换结果寄存器（ADCR、ADCRH）的值变为“0000H”或者“00H”。

12.5 输入电压和转换结果

模拟输入引脚（ANI0、ANI1、ANI16、ANI17）的模拟输入电压和理论上的 A/D 转换结果（10 位 A/D 转换结果寄存器（ADCR））有以下表达式的关系。

$$SAR = INT(\frac{V_{AIN}}{AV_{REF}} \times 1024 + 0.5)$$
$$ADCR = SAR \times 64$$

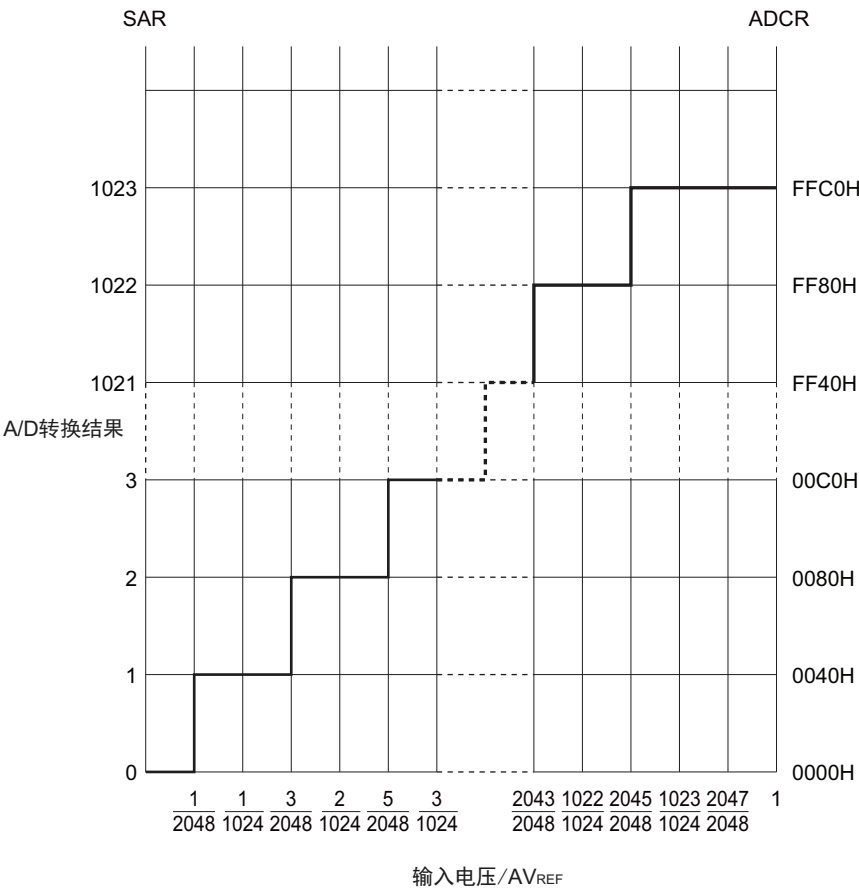
或者

$$(\frac{ADCR}{64} - 0.5) \times \frac{AV_{REF}}{1024} \leq V_{AIN} < (\frac{ADCR}{64} + 0.5) \times \frac{AV_{REF}}{1024}$$

- INT(): 将括号中的数值的整数部分返回的函数
- V_{AIN}: 模拟输入电压
- AV_{REF}: AV_{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

模拟输入电压和 A/D 转换结果的关系如图 12-16 所示。

图 12-16 模拟输入电压和 A/D 转换结果的关系



备注 AV_{REF} 是 A/D 转换器的正（+）基准电压，可选择 AV_{REFP}、内部基准电压（1.45V）或者 V_{DD}。

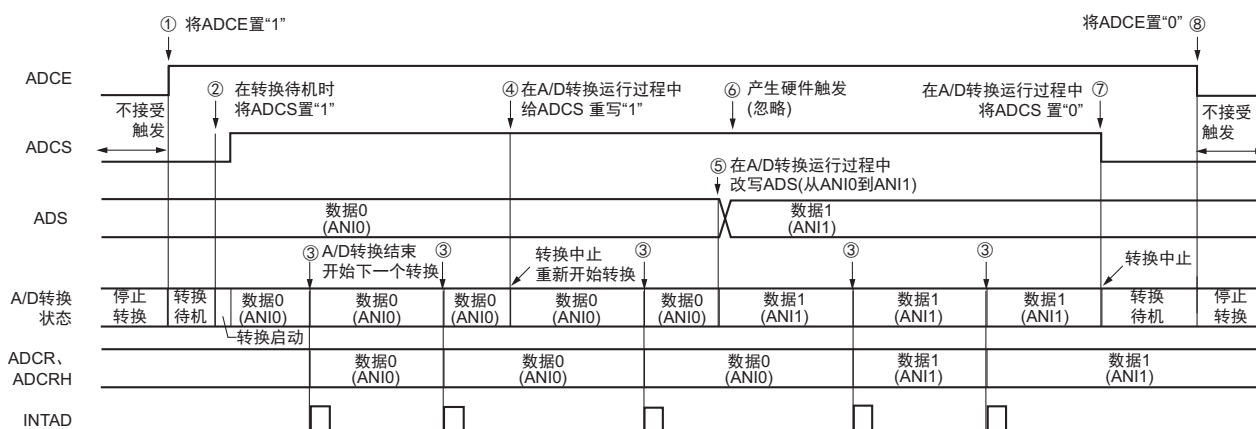
12.6 A/D 转换器的运行模式

A/D 转换器的各模式的运行如下所示。有关各模式的设定步骤，请参照“12.7 A/D 转换器的设定流程图”。

12.6.1 软件触发模式（连续转换模式）

- ① 在停止状态下，将 A/D 转换器的模式寄存器 0（ADM0）的 ADCE 位置“1”，进入 A/D 转换待机状态。
- ② 在通过软件对稳定等待时间（1 μ s）进行计数后，将 ADM0 寄存器的 ADCS 位置“1”，对模拟输入通道指定寄存器（ADS）指定的模拟输入进行 A/D 转换。
- ③ 如果 A/D 转换结束，就将转换结果保存到 A/D 转换结果寄存器（ADCR、ADCRH），并且产生 A/D 转换结束中断请求信号（INTAD）。在 A/D 转换结束后立即开始下一轮的 A/D 转换。
- ④ 如果在转换过程中给 ADCS 位重写“1”，就中止当前的 A/D 转换而重新开始转换。取消转换途中的数据。
- ⑤ 如果在转换过程中改写或者重写 ADS 寄存器，就中止当前的 A/D 转换而对由 ADS 寄存器重新指定的模拟输入进行 A/D 转换。取消转换途中的数据。
- ⑥ 即使在转换过程中输入硬件触发也不开始 A/D 转换。
- ⑦ 如果在转换过程中将 ADCS 位置“0”，就中止当前的 A/D 转换而进入 A/D 转换待机状态。
- ⑧ 如果在 A/D 转换待机状态下将 ADCE 位置“0”，A/D 转换器就进入停止状态。当 ADCE 位为“0”时，即使将 ADCS 位置“1”也被忽视，不开始 A/D 转换。

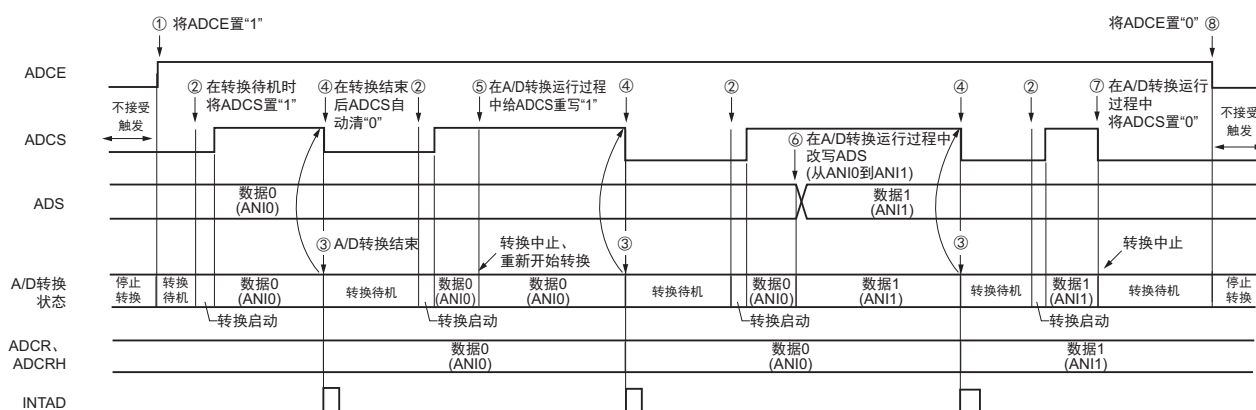
图 12-17 软件触发模式（连续转换模式）的运行时序例子



12.6.2 软件触发模式（单次转换模式）

- ① 在停止状态下，将A/D转换器的模式寄存器0（ADM0）的ADCE位置“1”，进入A/D转换待机状态。
- ② 在通过软件对稳定等待时间（1 μ s）进行计数后，将ADM0寄存器的ADCS位置“1”，对模拟输入通道指定寄存器（ADS）指定的模拟输入进行A/D转换。
- ③ 如果A/D转换结束，就将转换结果保存到A/D转换结果寄存器（ADCR、ADCRH），并且产生A/D转换结束中断请求信号（INTAD）。
- ④ 在A/D转换结束后，ADCS位自动清“0”，进入A/D转换待机状态。
- ⑤ 如果在转换过程中给ADCS位重写“1”，就中止当前的A/D转换而重新开始转换。取消转换途中的数据。
- ⑥ 如果在转换过程中改写或者重写ADS寄存器，就中止当前的A/D转换而对由ADS寄存器重新指定的模拟输入进行A/D转换。取消转换途中的数据。
- ⑦ 如果在转换过程中将ADCS位置“0”，就中止当前的A/D转换而进入A/D转换待机状态。
- ⑧ 如果在A/D转换待机状态下将ADCE位置“0”，A/D转换器就进入停止状态。当ADCE位为“0”时，即使将ADCS位置“1”也被忽视，不开始A/D转换。即使在A/D转换待机的状态下输入硬件触发也不开始A/D转换。

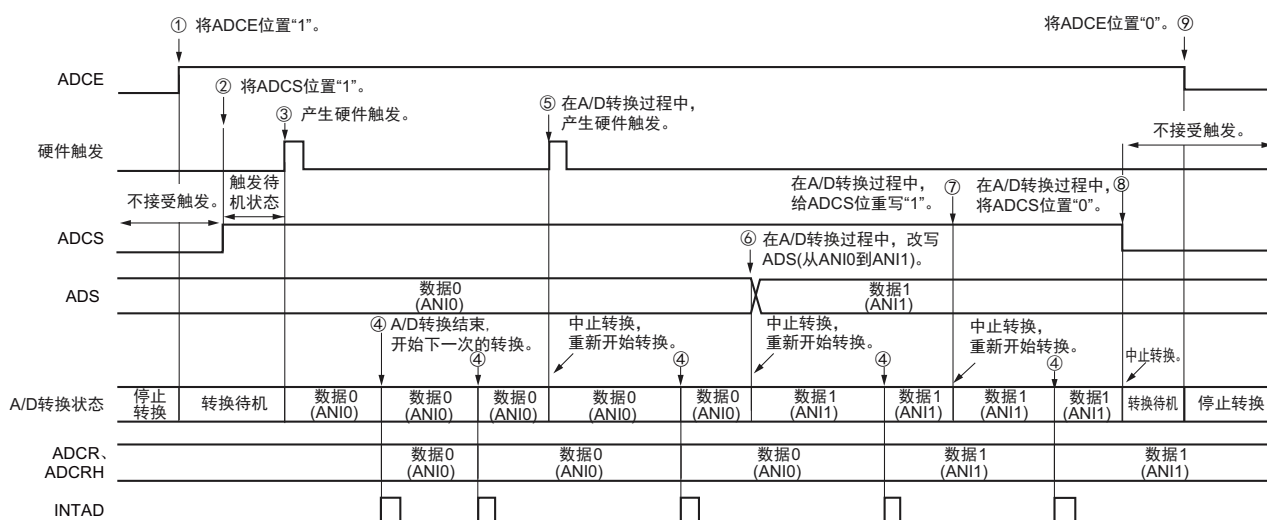
图 12-18 软件触发模式（单次转换模式）的运行时序例子



12.6.3 硬件触发无等待模式（连续转换模式）

- ① 在停止状态下，将 A/D 转换器的模式寄存器 0（ADM0）的 ADCE 位置“1”，进入 A/D 转换待机状态。
- ② 在通过软件对稳定等待时间（1μs）进行计数后，将 ADM0 寄存器的 ADCS 位置“1”，进入硬件触发待机状态（此阶段不开始转换）。在硬件触发待机状态时，即使将 ADCS 位置“1”也不开始 A/D 转换。
- ③ 如果在 ADCS 位为“1”的状态下输入硬件触发，就对由模拟输入通道指定寄存器（ADS）指定的模拟输入进行 A/D 转换。
- ④ 如果 A/D 转换结束，就将转换结果保存到 A/D 转换结果寄存器（ADCR、ADCRH），并且产生 A/D 转换结束中断请求信号（INTAD）。在 A/D 转换结束后立即开始下一轮的 A/D 转换。
- ⑤ 如果在转换过程中输入硬件触发，就中止当前的 A/D 转换而重新开始转换。取消转换途中的数据。
- ⑥ 如果在转换过程中改写或者重写 ADS 寄存器，就中止当前的 A/D 转换而对由 ADS 寄存器重新指定的模拟输入进行 A/D 转换。取消转换途中的数据。
- ⑦ 如果在转换过程中给 ADCS 位重写“1”，就中止当前的 A/D 转换而重新开始转换。取消转换途中的数据。
- ⑧ 如果在转换过程中将 ADCS 位置“0”，就中止当前的 A/D 转换而进入 A/D 转换待机状态。但是，在此状态下 A/D 转换器不进入停止状态。
- ⑨ 如果在 A/D 转换待机状态下将 ADCE 位置“0”，A/D 转换器就进入停止状态。当 ADCS 位为“0”时，即使输入硬件触发也被忽视，不开始 A/D 转换。

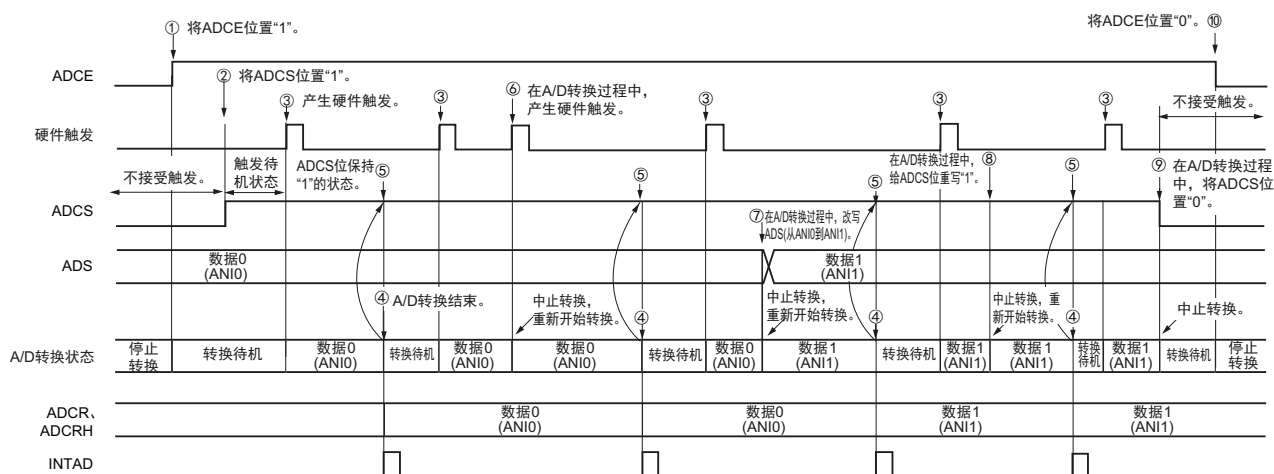
图 12-19 硬件触发无等待模式（连续转换模式）的运行时序例子



12.6.4 硬件触发无等待模式（单次转换模式）

- ① 在停止状态下，将 A/D 转换器的模式寄存器 0（ADM0）的 ADCE 位置“1”，进入 A/D 转换待机状态。
- ② 在通过软件对稳定等待时间（1 μ s）进行计数后，将 ADM0 寄存器的 ADCS 位置“1”，进入硬件触发待机状态（此阶段不开始转换）。在硬件触发待机状态时，即使将 ADCS 位置“1”也不开始 A/D 转换。
- ③ 如果在 ADCS 位为“1”的状态下输入硬件触发，就对由模拟输入通道指定寄存器（ADS）指定的模拟输入进行 A/D 转换。
- ④ 如果 A/D 转换结束，就将转换结果保存到 A/D 转换结果寄存器（ADCR、ADCRH），并且产生 A/D 转换结束中断请求信号（INTAD）。
- ⑤ 在 A/D 转换结束后，ADCS 位保持“1”的状态，进入 A/D 转换待机状态。
- ⑥ 如果在转换过程中输入硬件触发，就中止当前的 A/D 转换而重新开始转换。取消转换途中的数据。
- ⑦ 如果在转换过程中改写或者重写 ADS 寄存器，就中止当前的 A/D 转换而对由 ADS 寄存器重新指定的模拟输入进行 A/D 转换。取消转换途中的数据。
- ⑧ 如果在转换过程中给 ADCS 位重写“1”，就中止当前的 A/D 转换而重新开始转换。取消转换途中的数据。
- ⑨ 如果在转换过程中将 ADCS 位置“0”，就中止当前的 A/D 转换而进入 A/D 转换待机状态。但是，在此状态下 A/D 转换器不进入停止状态。
- ⑩ 如果在 A/D 转换待机状态下将 ADCE 位置“0”，A/D 转换器就进入停止状态。当 ADCS 位为“0”时，即使输入硬件触发也被忽视，不开始 A/D 转换。

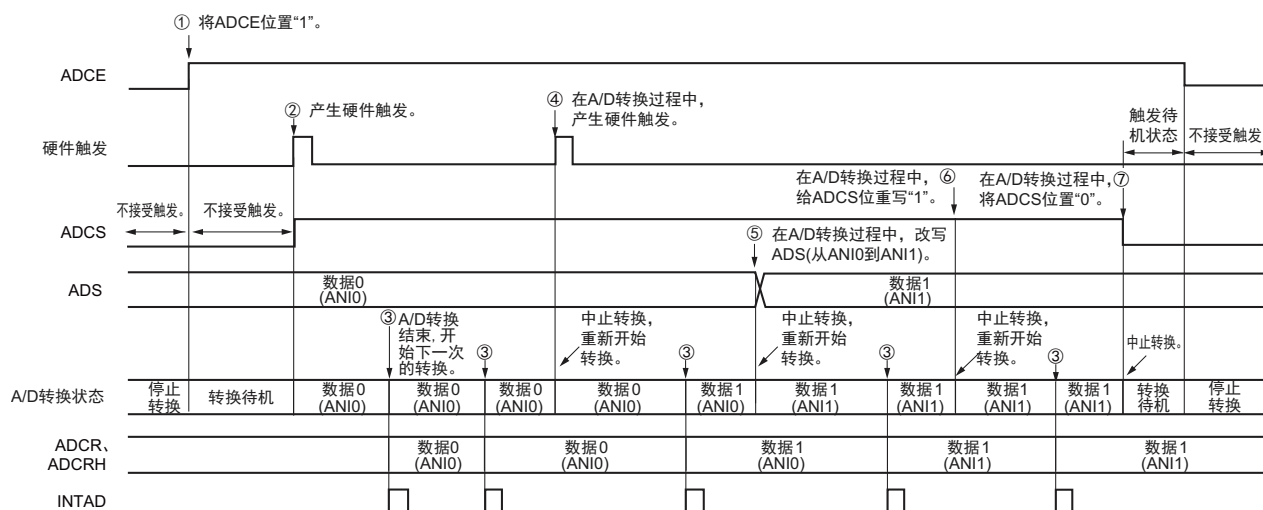
图 12-20 硬件触发无等待模式（单次转换模式）的运行时序例子



12.6.5 硬件触发等待模式（连续转换模式）

- ① 在停止状态下，将A/D转换器的模式寄存器0（ADM0）的ADCE位置“1”，进入硬件触发待机状态。
- ② 如果在硬件触发待机状态下输入硬件触发，就对由模拟输入通道指定寄存器（ADS）指定的模拟输入进行A/D转换。在输入硬件触发的同时自动将ADM0寄存器的ADCS位置“1”。
- ③ 如果A/D转换结束，就将转换结果保存到A/D转换结果寄存器（ADCR、ADCRH），并且产生A/D转换结束中断请求信号（INTAD）。在A/D转换结束后立即开始下一次的A/D转换（此时，不需要硬件触发）。
- ④ 如果在转换过程中输入硬件触发，就中止当前的A/D转换而重新开始转换。取消转换途中的数据。
- ⑤ 如果在转换过程中改写或者重写ADS寄存器，就中止当前的A/D转换而对由ADS寄存器重新指定的模拟输入进行A/D转换。取消转换途中的数据。
- ⑥ 如果在转换过程中给ADCS位重写“1”，就中止当前的A/D转换而重新开始转换。取消转换途中的数据。
- ⑦ 如果在转换过程中将ADCS位置“0”，就中止当前的A/D转换而进入硬件触发待机状态，并且A/D转换器进入停止状态。当ADCE位为“0”时，即使输入硬件触发也被忽视，不开始A/D转换。

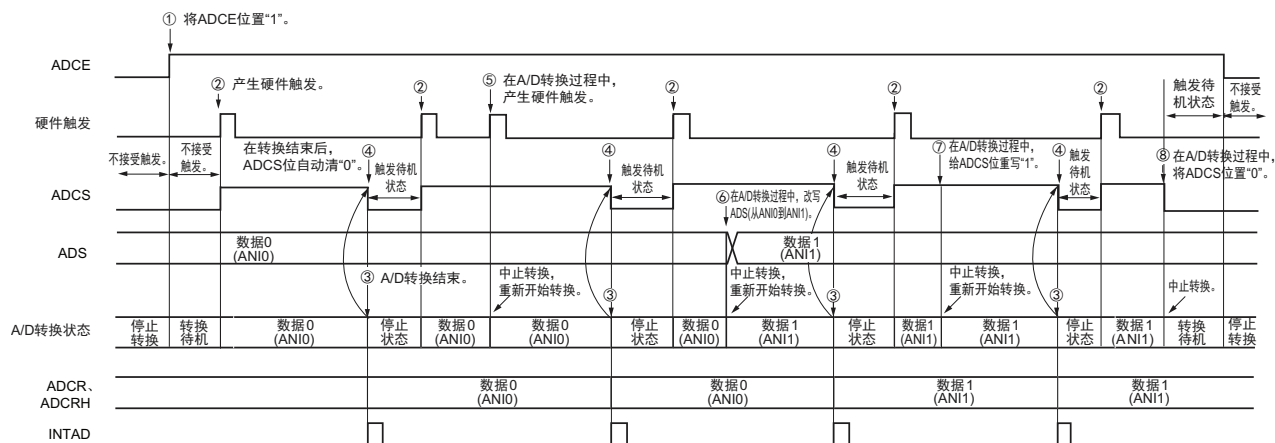
图 12-21 硬件触发等待模式（连续转换模式）的运行时序例子



12.6.6 硬件触发等待模式（单次转换模式）

- ① 在停止状态下，将A/D转换器的模式寄存器0（ADM0）的ADCE位置“1”，进入硬件触发待机状态。
- ② 如果在硬件触发待机状态下输入硬件触发，就对由模拟输入通道指定寄存器（ADS）指定的模拟输入进行A/D转换。在输入硬件触发的同时自动将ADM0寄存器的ADCS位置“1”。
- ③ 如果A/D转换结束，就将转换结果保存到A/D转换结果寄存器（ADCR、ADCRH），并且产生A/D转换结束中断请求信号（INTAD）。
- ④ 在A/D转换结束后，ADCS位自动清“0”，A/D转换器进入停止状态。
- ⑤ 如果在转换过程中输入硬件触发，就中止当前的A/D转换而重新开始转换。取消转换途中的数据。
- ⑥ 如果在转换过程中改写或者重写ADS寄存器，就中止当前的A/D转换而对由ADS寄存器重新指定的模拟输入进行A/D转换。取消转换途中的数据。
- ⑦ 如果在转换过程中给ADCS位重写“1”，就中止当前的A/D转换而重新开始转换。取消转换途中的数据。
- ⑧ 如果在转换过程中将ADCS位置“0”，就中止当前的A/D转换而进入硬件触发待机状态，并且A/D转换器进入停止状态。当ADCE位为“0”时，即使输入硬件触发也被忽视，不开始A/D转换。

图 12-22 硬件触发等待模式（单次转换模式）操作时序的示例

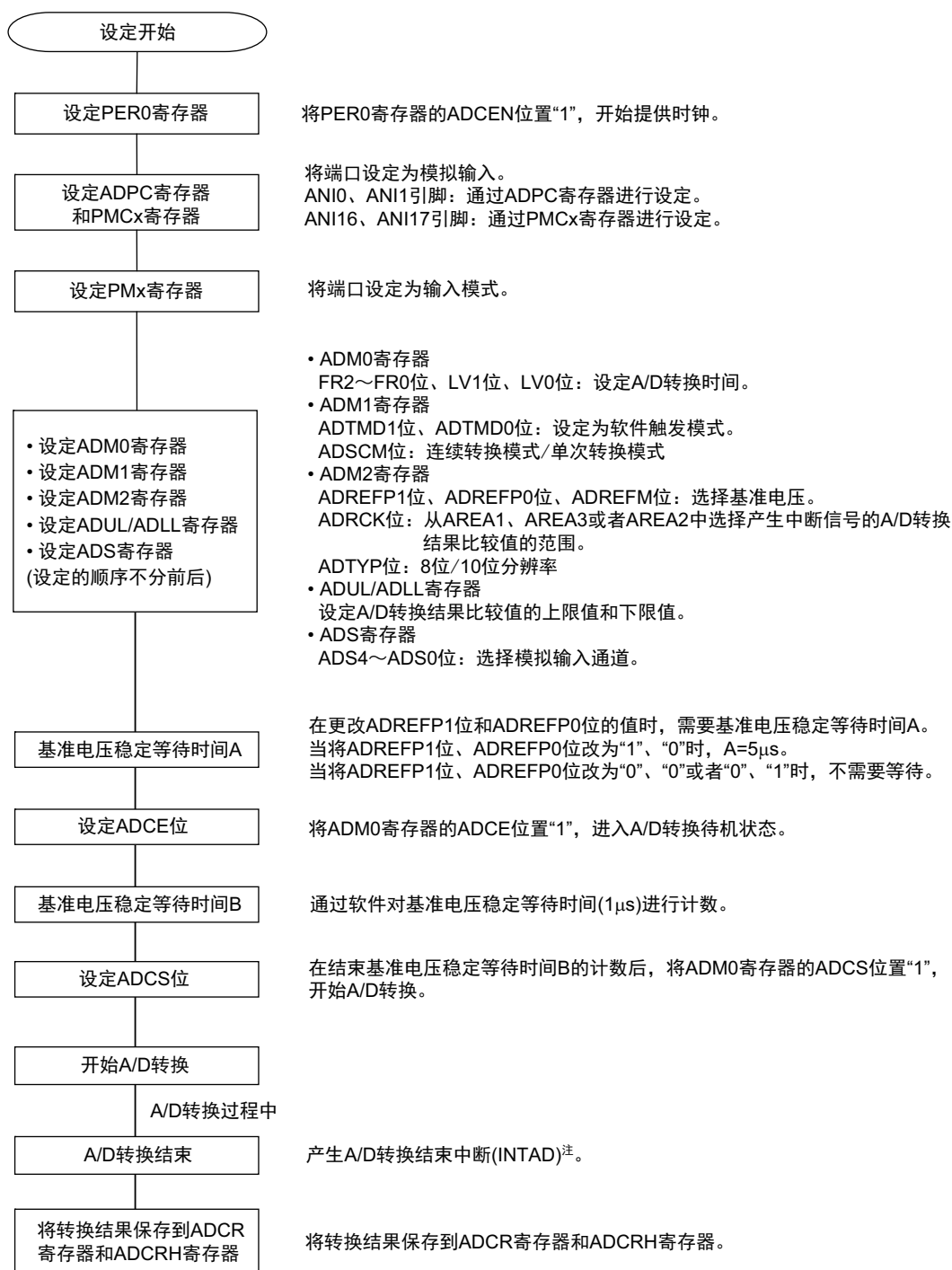


12.7 A/D 转换器的设定流程图

各运行模式的 A/D 转换器的设定流程图如下所示。

12.7.1 软件触发模式的设定

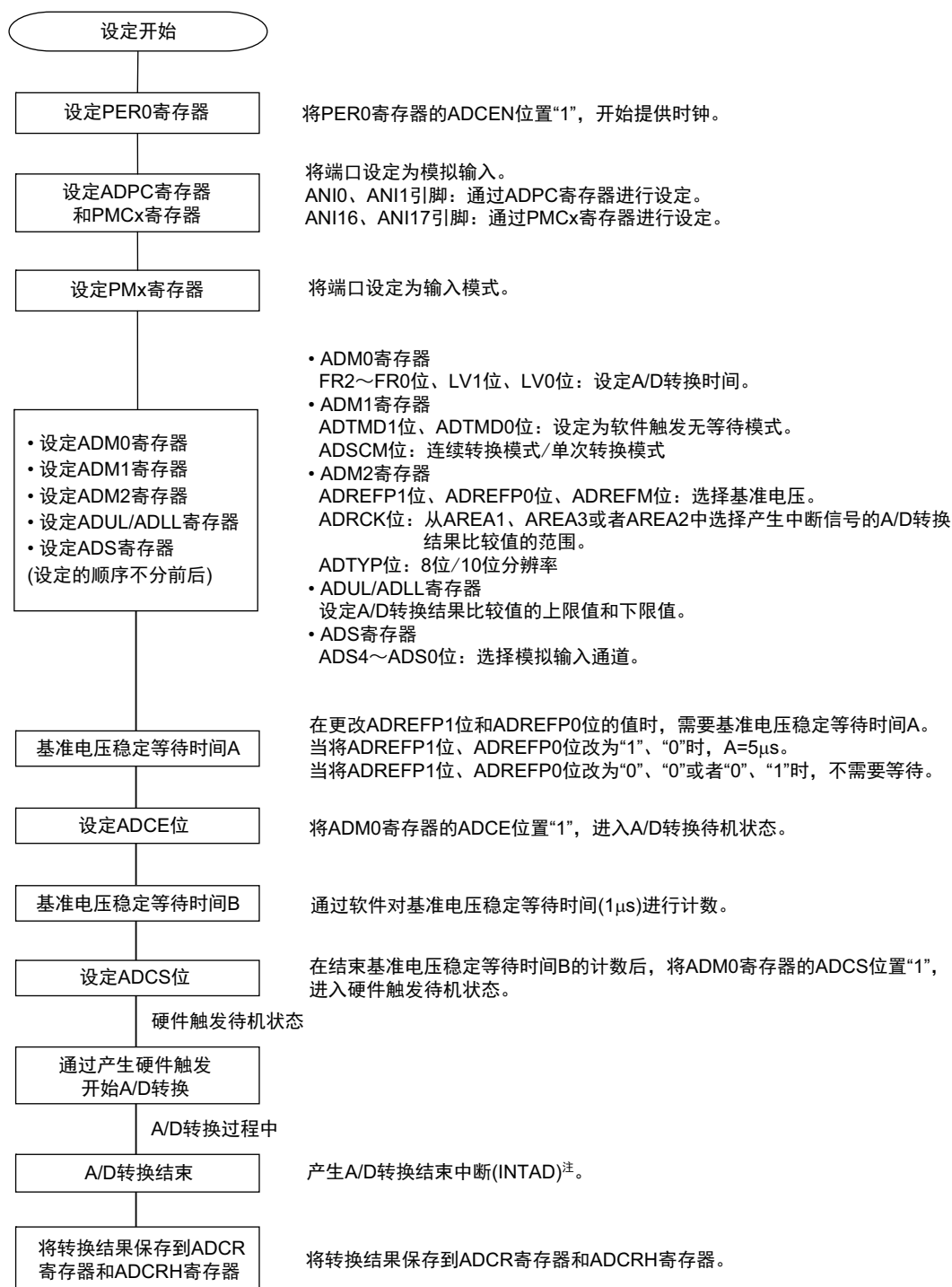
图 12-23 软件触发模式的设定



注 根据 ADRCK 位和 ADUL/ADLL 寄存器的设定，有可能不产生中断信号。此时，不将结果保存到 ADCR 寄存器和 ADCRH 寄存器。

12.7.2 硬件触发无等待模式的设定

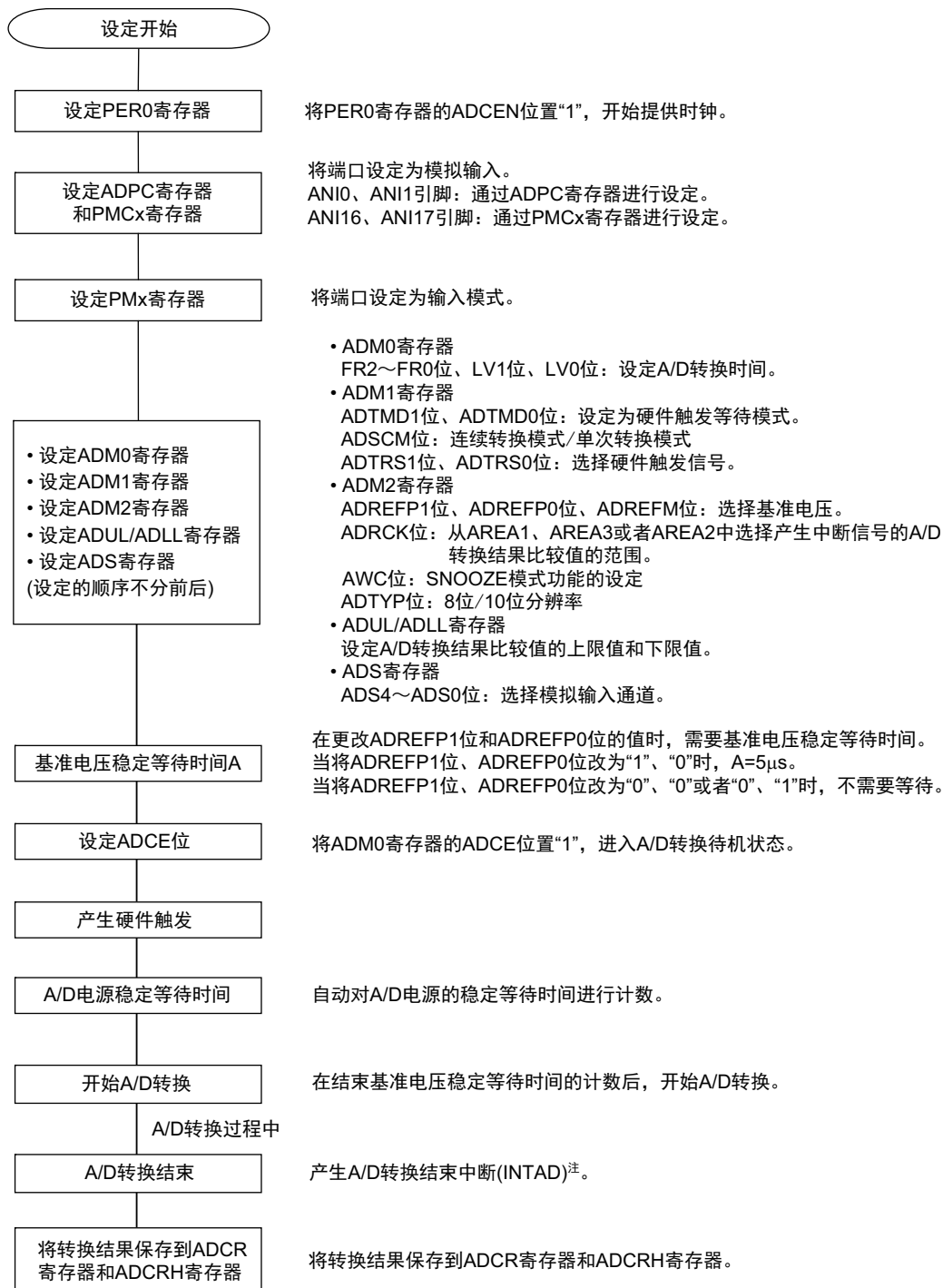
图 12-24 硬件触发无等待模式的设定



注 根据 ADRCK 位和 ADUL/ADLL 寄存器的设定，有可能不产生中断信号。此时，不将结果保存到 ADCR 寄存器和 ADCRH 寄存器。

12.7.3 硬件触发等待模式的设定

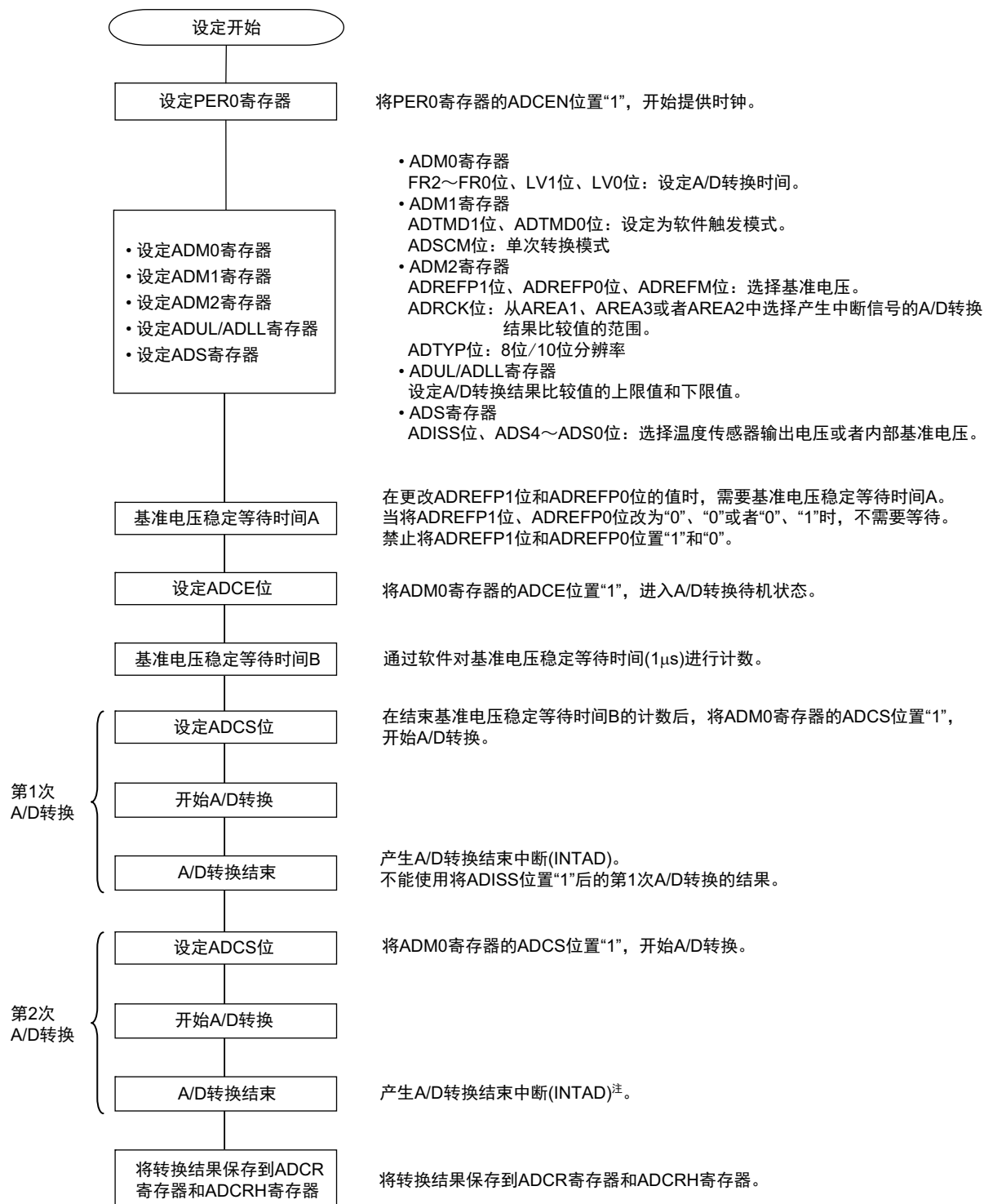
图 12-25 硬件触发等待模式的设定



注 根据 ADRCK 位和 ADUL/ADLL 寄存器的设定，有可能不产生中断信号。此时，不将结果保存到 ADCR 寄存器和 ADCRH 寄存器。

12.7.4 使用温度传感器时的设定（以软件触发模式、单次转换模式为例）

图 12-26 选择温度传感器输出电压 / 内部基准电压时的设定

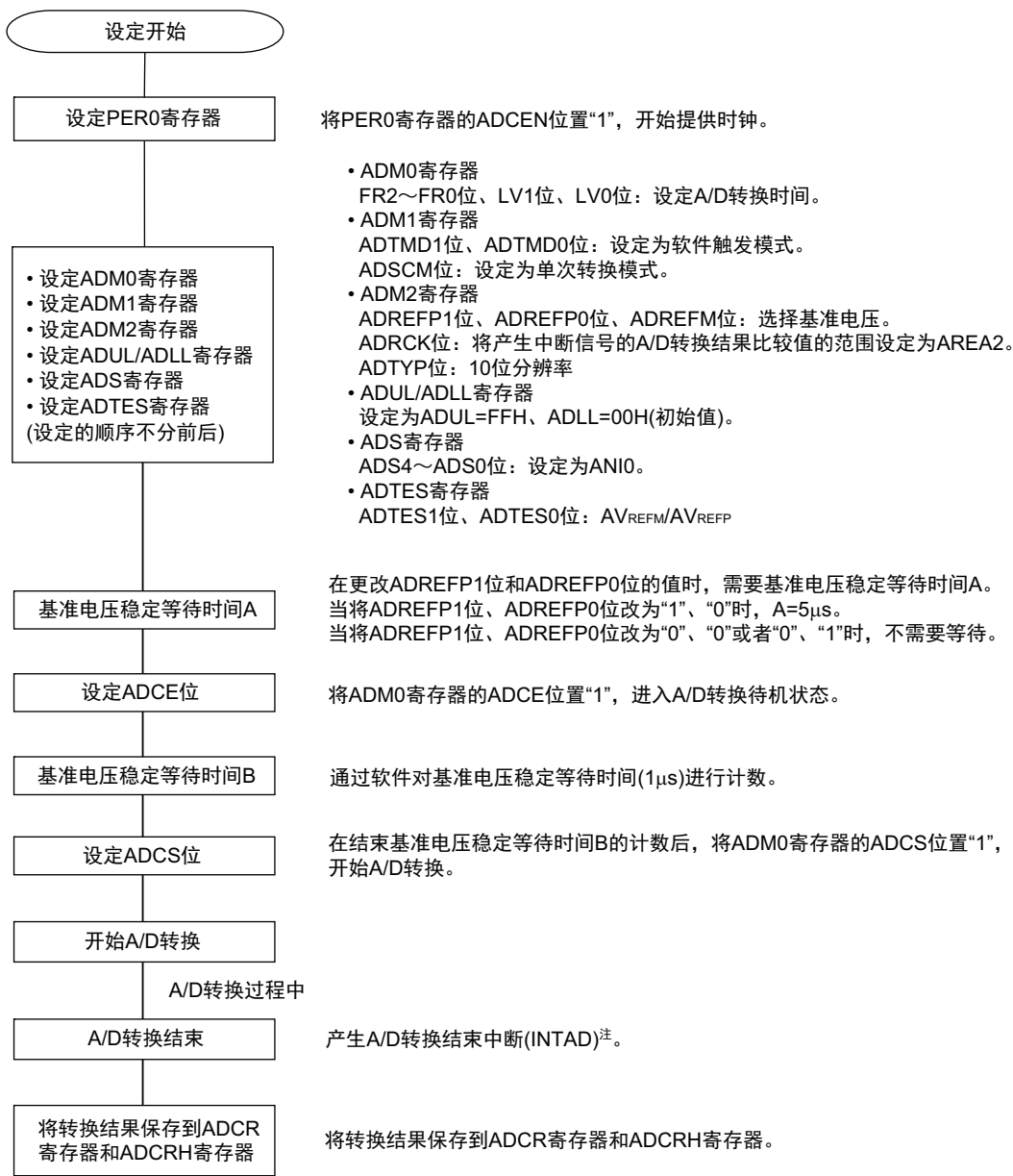


注 根据 ADRCCK 位和 ADUL/ADLL 寄存器的设定，有可能不产生中断信号。此时，不将结果保存到 ADCR 寄存器和 ADCRH 寄存器。

注意 只有在 HS（高速主）模式中才能选择。

12.7.5 测试模式的设定

图 12-27 测试模式的设定



注 根据 ADRCK 位和 ADUL/ADLL 寄存器的设定，有可能不产生中断信号。此时，不将结果保存到 ADCR 寄存器和 ADCRH 寄存器。

注意 有关 A/D 转换器的测试方法，请参照“25.3.8 A/D 测试功能”。

12.8 SNOOZE 模式功能

这是在 STOP 模式中通过输入硬件触发进行 A/D 转换的模式。在通常的 STOP 模式中 A/D 转换停止，但是通过使用 SNOOZE 模式功能，就能在 CPU 不运行的状态下进行 A/D 转换，减少工作电流。

在 SNOOZE 模式中，如果通过 ADUL 寄存器和 ADLL 寄存器指定转换结果的范围，就能每隔一段时间判断 A/D 转换结果。从而能判断电源电压监视或者通过 A/D 输入进行的键输入。

在 SNOOZE 模式中，只能使用以下转换模式：

- 硬件触发等待模式（单次转换模式）

注意 只有在选择高速内部振荡器时钟作为 f_{CLK} 时才能设定 SNOOZE 模式。

图 12-28 使用 SNOOZE 模式功能时的框图



当使用 SNOOZE 模式功能时，在转移到 STOP 模式前进行各寄存器的初始设定（参照“12.7.3 硬件触发等待模式的设定”注2）。在即将要转移到 STOP 模式前，将 A/D 转换器的模式寄存器 2（ADM2）的 bit2（AWC）置“1”。在初始设定结束后，将 A/D 转换器的模式寄存器 0（ADM0）的 bit0（ADCE）置“1”。

如果在转移到 STOP 模式后输入硬件触发，就给 A/D 转换器提供高速内部振荡器时钟。在提供高速内部振荡器时钟后，自动对 A/D 电源稳定等待时间进行计数，开始 A/D 转换。

A/D 转换结束后的 SNOOZE 模式的运行因是否产生中断信号而不同注1。

注 1. 根据 A/D 转换结果比较功能的设定（ADRCK 位、ADUL/ADLL 寄存器），有可能不产生中断请求信号。

2. 必须将 ADM1 寄存器置“E2H”或者“E3H”。

备注 硬件触发为 INTRTC 或者 INTIT。

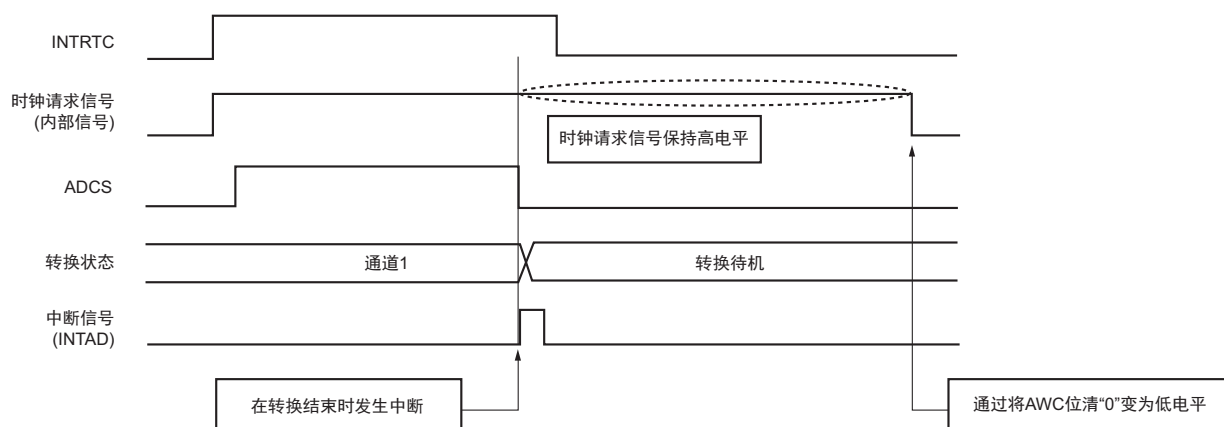
必须通过 A/D 转换器的模式寄存器 1（ADM1）设定硬件触发。

(1) 在 A/D 转换结束后发生中断的情况

如果 A/D 转换结果的值不超出 A/D 转换结果比较功能（通过 ADRCK 位和 ADUL/ADLL 寄存器进行设定）的设定值范围，就产生 A/D 转换结束中断请求信号（INTAD）。

如果在 A/D 转换结束后产生 A/D 转换结束中断请求信号（INTAD），A/D 转换器就从 SNOOZE 模式转移到通常运行模式。在此，必须清除 A/D 转换器的模式寄存器 2（ADM2）的 bit2（AWC=0：解除 SNOOZE 模式）。如果 AWC 位保持“1”的状态，就无法正常开始 A/D 转换，而与之之后的 SNOOZE 模式和通常运行模式无关。

图 12-29 在 A/D 转换结束后发生中断时的运行例子



(2) 在 A/D 转换结束后不发生中断的情况

如果 A/D 转换结果的值不超出 A/D 转换结果比较功能（通过 ADRCK 位和 ADUL/ADLL 寄存器进行设定）的设定值范围，就不产生 A/D 转换结束中断请求信号（INTAD）。

如果不发生 A/D 转换结束中断请求信号（INTAD），时钟请求信号（内部信号）就在 A/D 转换结束后自动变为低电平，并且停止提供高速内部振荡器时钟。此后，如果输入硬件触发，就再次在 SNOOZE 模式中进行 A/D 转换。

图 12-30 A/D 转换结束后不产生中断时的操作示例

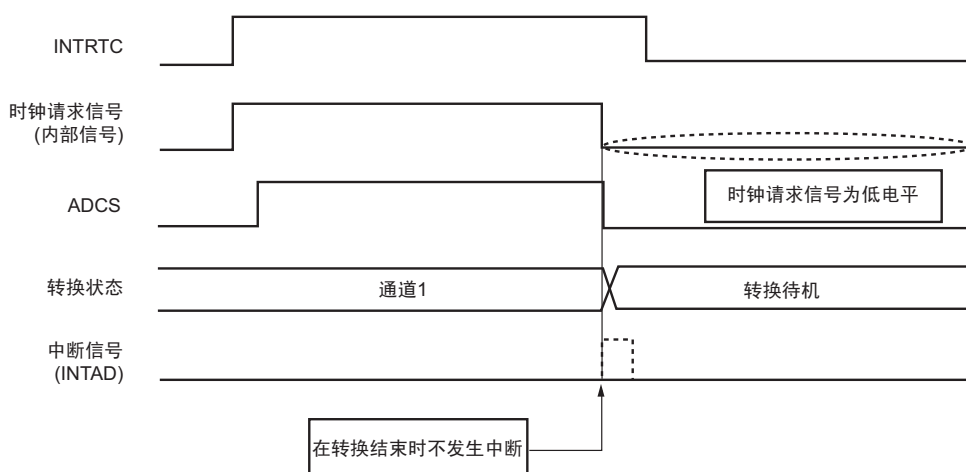
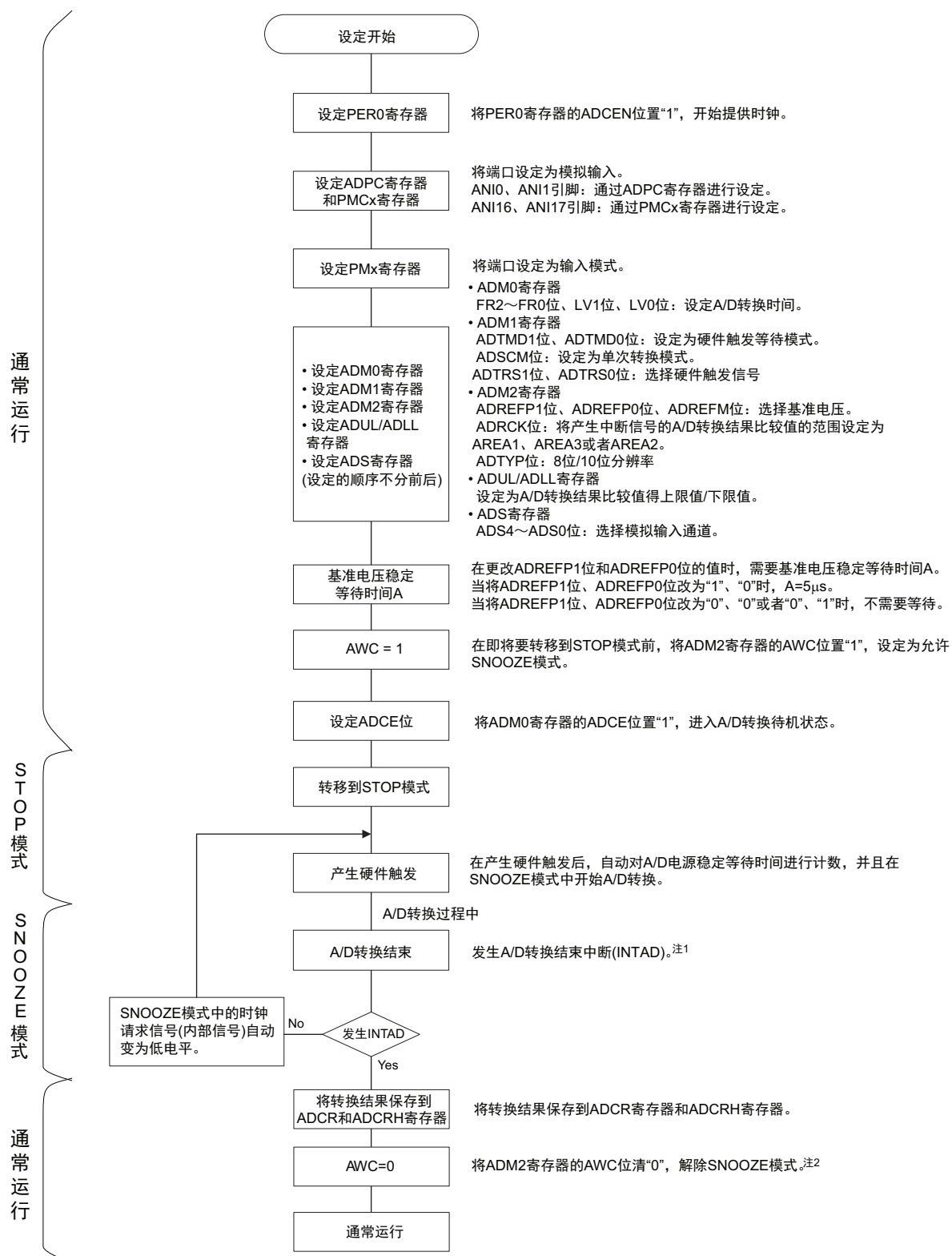


图 12-31 设定 SNOOZE 模式的流程图



注 1. 在通过设定 ADRCK 位 ADUL/ADLL 寄存器，没有产生 A/D 转换结束中断请求信号（INTAD）时，结果不保存到 ADCR 寄存器和 ADCRH 寄存器，重新转移到 STOP 模式。此后，如果输入硬件触发，就再次在 SNOOZE 模式中进行 A/D 转换运行。

2. 在AWC保持“1”的状态下，与之后的SNOOZE模式和通常运行模式无关，不能正常开始A/D转换。必须将AWC置“0”。

12.9 A/D 转换器特性表的阅读方法

以下说明 A/D 转换器特有的专业术语。

(1) 分辨率

分辨率是能分辨的最小模拟输入电压。也就是说，数字输出的每 1 位与模拟输入电压的比率称为 1 LSB (Least Significant Bit)。将对 1 LSB 满刻度的比率表示为 %FSR (Full Scale Range)。

当分辨率为 10 位时，

$$\begin{aligned} 1 \text{ LSB} &= 1/2^{10} = 1/1024 \\ &= 0.098\% \text{FSR} \end{aligned}$$

精度与分辨率无关而取决于综合误差。

(2) 综合误差

综合误差是指实际测量值和理论值的最大差值，是将零刻度误差、满刻度误差、积分线性误差、微分线性误差以及这些组合所产生的误差综合起来的误差。

特性表中的综合误差不包含量化误差。

(3) 量化误差

在将模拟值转换为数字值时，必然会出现 $\pm 1/2 \text{LSB}$ 的误差。A/D 转换器将 $\pm 1/2 \text{LSB}$ 范围内的模拟输入电压转换为相同的数字码，因此不能避免量化误差。

特性表中的综合误差、零刻度误差、满刻度误差、积分线性误差和微分线性误差不包含量化误差。

图 12-32 综合误差

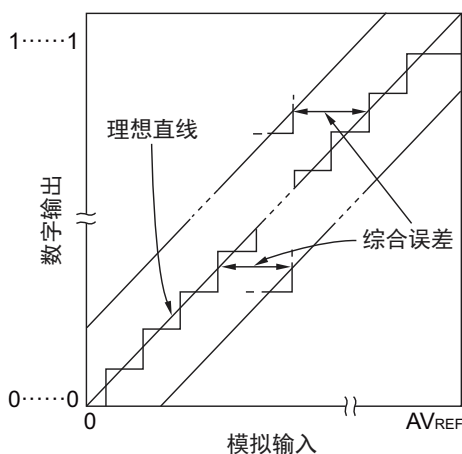
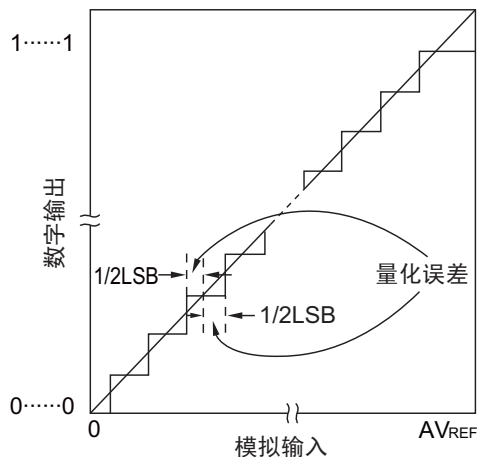


图 12-33 量化误差



(4) 零刻度误差

零刻度误差是指数字输出从 0.....000 变为 0.....001 时的模拟输入电压的实际测量值和理论值（1/2 LSB）的差。如果实际测量值大于理论值，零刻度误差就是指数码输出从 0.....001 变为 0.....010 时的模拟输入电压的实际测量值和理论值（3/2 LSB）的差。

(5) 满刻度误差

满刻度误差是指数字输出从 1.....110 变为 1.....111 时的模拟输入电压的实际测量值和理论值（满刻度 -3/2 LSB）的差。

(6) 积分线性误差

积分线性误差是指转换特性从理想线性关系偏离的程度，是零刻度误差和满刻度误差为“0”时的实际测量值和理想直线的最大差值。

(7) 微分线性误差

微分线性误差是指在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和理想值的差值。

图 12-34 零刻度误差

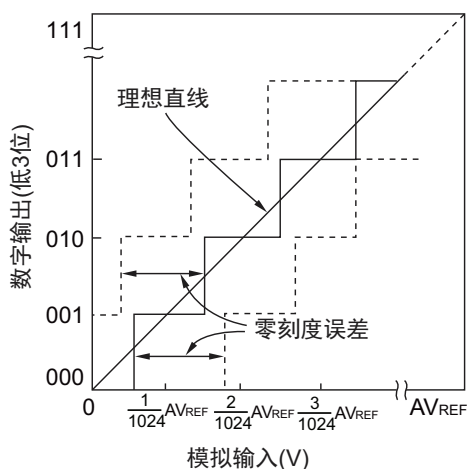


图 12-35 满刻度误差

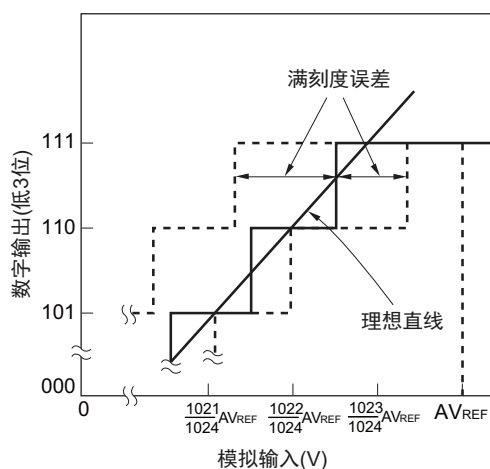


图 12-36 积分线性误差

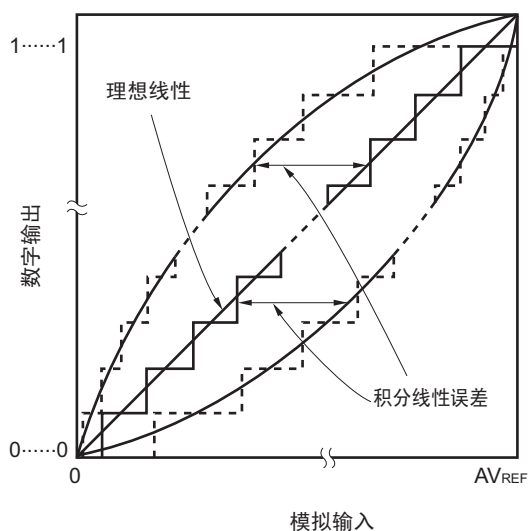
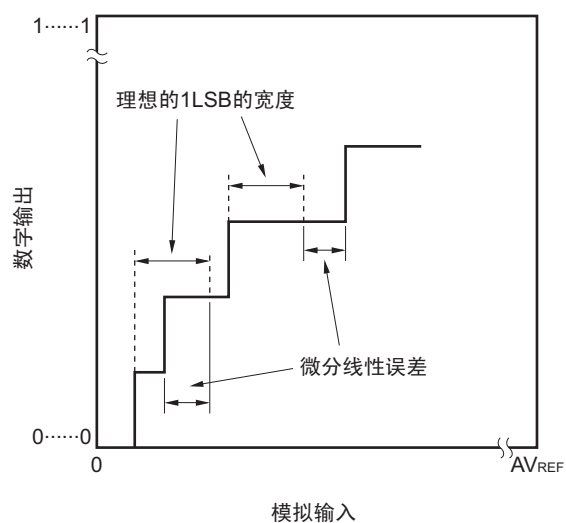


图 12-37 微分线性误差

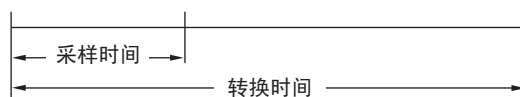


(8) 转换时间

转换时间是指从开始采样到取得数字输出的时间。
特性表中的转换时间包含采样时间。

(9) 采样时间

采样时间是指为了将模拟电压取入采样 & 保持电路而使模拟开关开启的时间。



12.10 A/D 转换器的注意事项

(1) 有关 STOP 模式中的工作电流

要转移到 STOP 模式时，必须在停止 A/D 转换器（将 A/D 转换器的模式寄存器 0 (ADM0) 的 bit7 (ADCS) 置“0”) 后进行。此时，还能通过将 ADM0 寄存器的 bit0 (ADCE) 置“0”来减少工作电流。

要从待机状态重新运行时，必须在将中断请求标志寄存器 1H (IF1H) 的 bit0 (ADIF) 清“0”后开始运行。

(2) 有关 ANI0、ANI1、ANI16 和 ANI17 引脚的输入范围

必须在规格范围内使用 ANI0、ANI1、ANI16 和 ANI17 引脚的输入电压。如果输入大于 V_{DD} 和 AV_{REFP} 或者小于 V_{SS} 和 AV_{REFM} （即使在绝对最大额定范围内）的电压，该通道的转换值就为不定值，并且还可能影响其他通道的转换值。

当选择内部基准电压（1.45V）作为 A/D 转换器的正（+）基准电压时，不能给 ADS 寄存器选择的引脚输入大于内部基准电压（1.45V）的电压。但是，能给 ADS 寄存器没有选择的引脚输入大于内部基准电压（1.45V）的电压。

注意 只有在 HS（高速主）模式中才能选择内部基准电压（1.45V）。

(3) 有关竞争

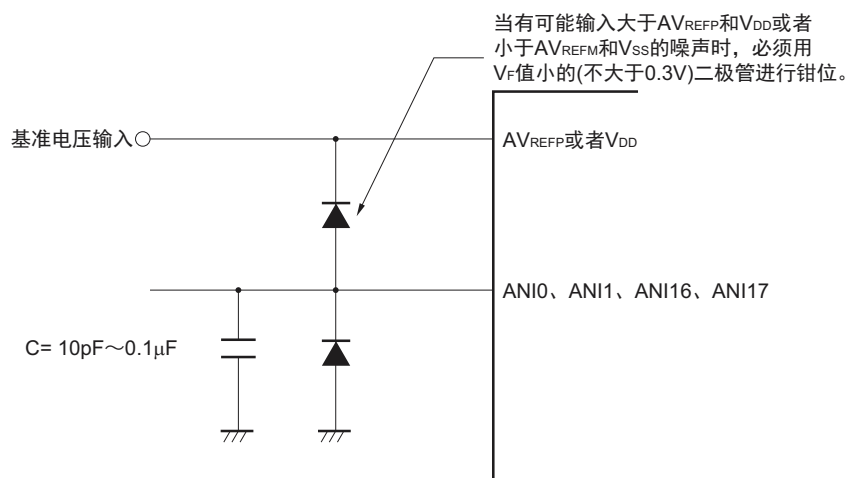
- ① 转换结束时的 A/D 转换结果寄存器 (ADCR、ADCRH) 的写操作与 ADCR 寄存器和 ADCRH 寄存器的指令读操作的竞争
优先读 ADCR 寄存器和 ADCRH 寄存器。在读后，将新的转换结果写到 ADCR 寄存器和 ADCRH 寄存器。
- ② 转换结束时的 ADCR 寄存器和 ADCRH 寄存器的写操作与 A/D 转换器的模式寄存器 0 (ADM0)、模拟输入通道指定寄存器 (ADS) 或者 A/D 端口配置寄存器 (ADPC) 的写操作的竞争
优先写 ADM0、ADS、ADPC 寄存器。不写 ADCR 寄存器和 ADCRH 寄存器，也不产生转换结束中断信号 (INTAD)。

(4) 有关噪声对策

为了保持 10 位分辨率，必须注意输入到 AV_{REFP} 、 V_{DD} 、ANI0、ANI1、ANI16 和 ANI17 引脚的噪声。

- ① 连接电源的电容器必须使用等效电阻小并且频率响应好的电容器（0.01 μ F 左右），而且最好以最短距离并且使用比较粗布线进行连接。
- ② 模拟输入源的输出阻抗越高影响就越大，因此为了降低噪声，建议按照图 12-38 的方法外接电容器。
- ③ 不能在转换过程中切换其他引脚。
- ④ 如果在转换开始后立即设定为 HALT 模式，就会提高精度。

图 12-38 模拟输入引脚的连接



(5) 模拟输入 (ANIn) 引脚

- ① 模拟输入 (ANI0、ANI1) 引脚和输入端口 (P20、P21) 引脚复用。
要选择 ANI0、ANI1 中的任意引脚进行 A/D 转换时，不能在转换过程中更改 P20、P21 的输出值。否则，可能降低转换精度。
- ② 如果将正在进行 A/D 转换的引脚的相邻引脚用作数字输入/输出端口，就可能因耦合噪声而取得与期待值不同的 A/D 转换值。因此，在 A/D 转换过程中，不能输入或者输出会发生急剧变化 (如数字信号) 的脉冲。

(6) 有关模拟输入 (ANIn) 引脚的输入阻抗

此 A/D 转换器在采样时间内给内部的采样电容充电并且进行采样。

因此，在不采样时只有漏电流流过，而在采样时还有电容充电的电流流过。所以输入阻抗根据是否采样而发生变动。

但是，为了充分进行采样，必须将模拟输入源的输出阻抗保持在 $1k\Omega$ 以下。如果不能将输出阻抗保持在 $1k\Omega$ 以下，建议加长采样时间，或者给 ANI0、ANI1、ANI16 和 ANI17 引脚外接 $0.1\mu F$ 左右的电容器 (参照图 12-38)。另外，在转换运行过程中将 ADCS 置“0”以及重新开始转换时，采样电容器的充电电压为不定值。因此，在将 ADCS 置“0”时，下一个转换从不定状态开始充电；在重新转换时，该转换从不定状态开始充电。为了确保足够的充电，与模拟信号变化的大小无关，必须将模拟输入源的输出阻抗降低或者确保充分的采样时间。

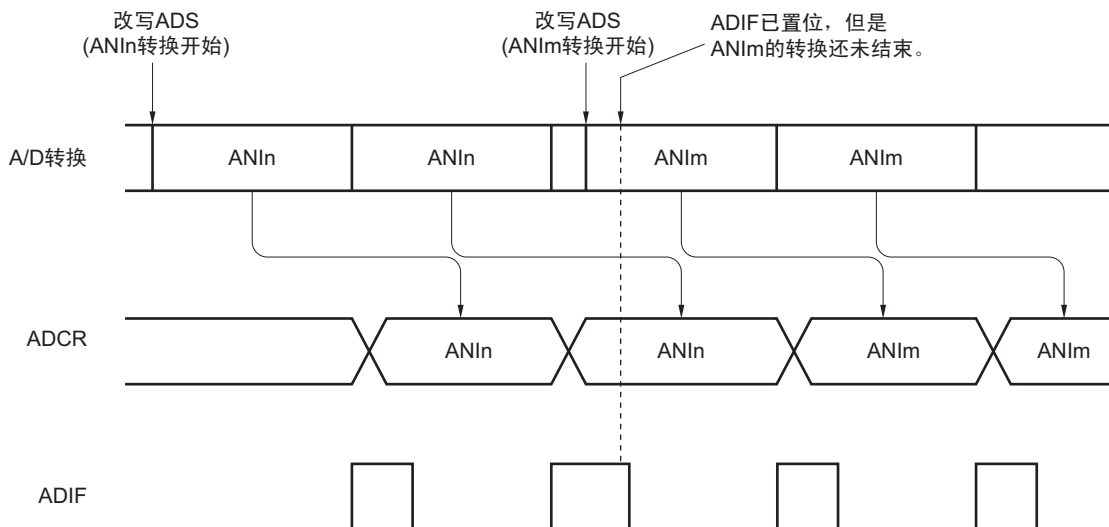
(7) 有关中断请求标志 (ADIF)

即使更改模拟输入通道指定寄存器 (ADS)，也不将中断请求标志 (ADIF) 清“0”。

因此，如果在 A/D 转换过程中更改模拟输入引脚，就可能在即将要改写 ADS 寄存器前，更改前的模拟输入的 A/D 转换结果和 ADIF 标志有可能被设定。必须注意：如果在改写 ADS 寄存器后立即读 ADIF 标志，尽管更改后的模拟输入的 A/D 转换还未结束，也会将 ADIF 标志置位。

另外，在暂停后重新开始 A/D 转换时，必须在重新开始前将 ADIF 标志清“0”。

图 12-39 产生 A/D 转换结束中断请求的时序



(8) 有关 A/D 转换开始后的初次转换结果

在软件触发模式或者硬件触发无等待模式中，如果在将 ADCE 位置“1”后的 $1\mu\text{s}$ 内将 ADCS 位置“1”，开始 A/D 转换后的初次 A/D 转换值就可能不满足额定值。必须采取轮询 A/D 转换结束中断请求 (INTAD) 并且取消初次的转换结果等对策。

(9) 有关 A/D 转换结果寄存器 (ADCR、ADCRH) 的读操作

当写 A/D 转换器的模式寄存器 0 (ADM0)、模拟输入通道指定寄存器 (ADS)、A/D 端口配置寄存器 (ADPC) 和端口模式控制寄存器 (PMC_x) 时，ADCR 寄存器和 ADCRH 寄存器的内容可能不定。必须在转换结束后并且在写 ADM0、ADS、ADPC、PMC_x 寄存器前读转换结果。否则，就可能读不到正确的转换结果。

(10) 有关内部等效电路

模拟输入部的等效电路如下所示。

图 12-40 ANIn 引脚的内部等效电路

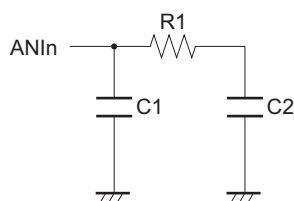


表 12-4 等效电路的电阻和电容值（参考值）

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、ANI1	14	8	2.5
	ANI16、ANI17	18	8	7.0
$2.7\text{ V} \leq V_{DD} \leq 3.6\text{ V}$	ANI0、ANI1	39	8	2.5
	ANI16、ANI17	53	8	7.0
$1.8\text{ V} \leq V_{DD} \leq 2.7\text{ V}$	ANI0、ANI1	231	8	2.5
	ANI16、ANI17	321	8	7.0
$1.6\text{ V} \leq V_{DD} < 2.7\text{ V}$	ANI0、ANI1	632	8	2.5
	ANI16、ANI17	902	8	7.0

备注 表 12-4 的各电阻和电容值不是保证值。

(11) 有关 A/D 转换器的运行开始

必须在 AV_{REFP} 和 V_{DD} 的电压稳定后开始 A/D 转换器的运行。

第 13 章 比较器

比较器对基准输入电压和模拟输入电压进行比较。由两个独立的比较器（比较器 0 和比较器 1）构成。

13.1 比较器的功能

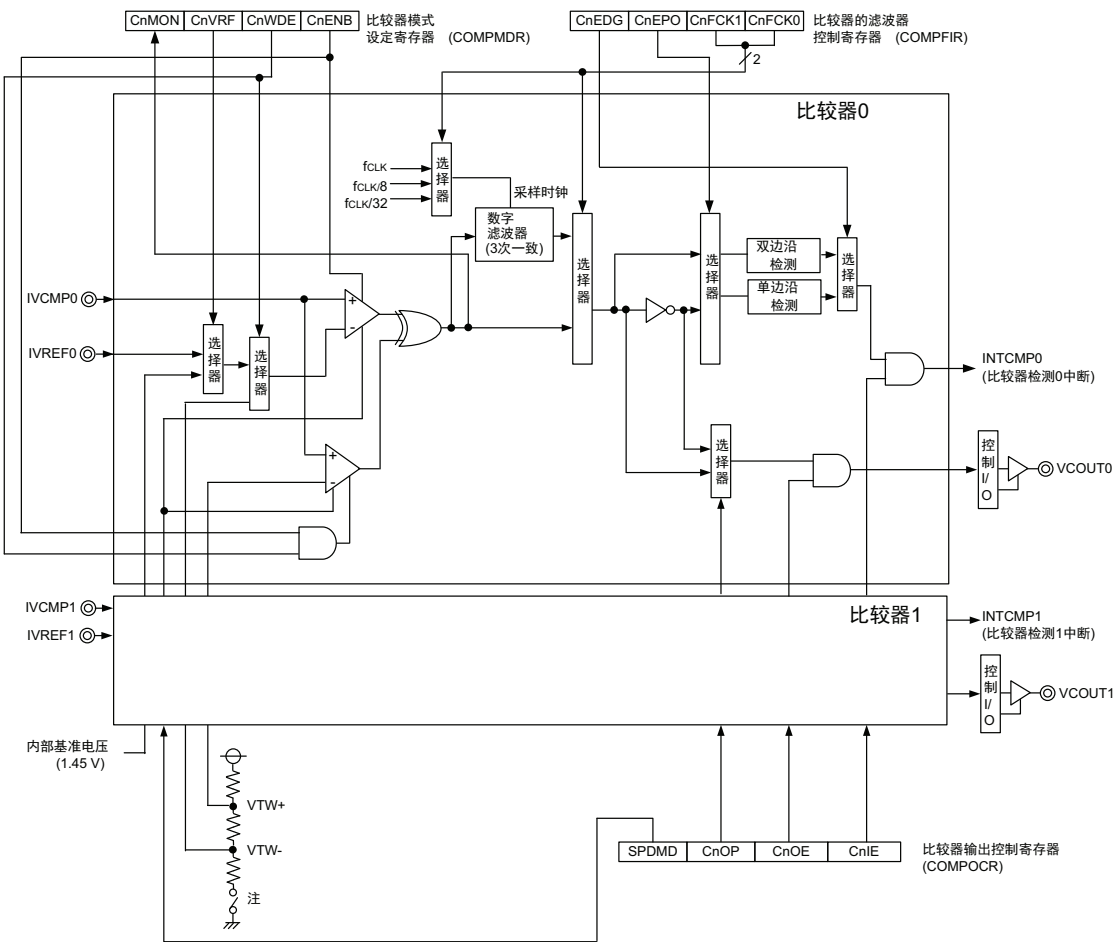
比较器具有以下功能。

- 可以选择比较器高速模式/比较器低速模式/比较器窗口模式。
- 可以选择外部基准电压或者内部基准电压作为基准电压。
- 可以选择噪声消除数字滤波器的消除宽度
- 可以检测比较器输出的有效边沿并且产生中断信号。

13.2 比较器的结构

比较器的框图如图 13-1 所示。

图 13-1 比较器的框图



注 如果将 C0WDE 位和 C1WDE 位中的 1 个或者 2 个都置“1”，该开关就为“ON”，并且用于产生比较电压的分压电阻为有效。

备注 n=0、1

13.3 控制比较器的寄存器

控制比较器的寄存器如表 13-1 所示。

表 13-1 控制比较器的寄存器

寄存器名称	符号
外围允许寄存器 1	PER1
比较器模式设定寄存器	COMPMDR
比较器的滤波器控制寄存器	COMPFIR
比较器输出控制寄存器	COMPOCR
端口模式控制寄存器 4	PMC4
LCD 端口功能寄存器 3	PFSEG3
端口模式寄存器 0、4	PM0、PM4
端口寄存器 0、4	P0、P4

13.3.1 外围允许寄存器 1（PER1）

PER1 寄存器设定允许或者禁止向各外围硬件提供时钟。通过停止向不使用的硬件供应时钟，可以降低功耗和噪声。

当使用比较器时，必须将 bit5（CMPEN）设定为“1”。

通过 1 位或 8 位存储器操作指令设定 PER1 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 13-2 外围允许寄存器 1（PER1）的格式

地址：F007AH 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
PER1	TMKAEN	FMCEN	CMPEN	0	0	SMCI1EN	SMCI0EN	0

CMPEN	控制比较器的输入时钟的
0	停止提供输入时钟。 - 不能写入比较器使用的 SFR。 - 比较器处于复位状态。
1	提供输入时钟。 能读写比较器使用的 SFR。

注意 1. 要设定比较器时，必须最先在 CMPEN 位为“1”的状态写设定以下寄存器。当 CMPEN=0 时，比较器的控制寄存器的值为初始值，并且忽视写操作（端口模式寄存器 0、4（PM0、PM4）、端口寄存器 0、4（P0、P4）、端口模式控制寄存器 4（PMC4）和 LCD 端口功能寄存器 3（PFSEG3）除外）。

- 比较器模式设定寄存器（COMPMDR）
- 比较器的滤波器控制寄存器（COMPFIR）
- 比较器输出控制寄存器（COMPOCR）

2. 必须将 bit4、3、0 设定为“0”。

13.3.2 比较器模式设定寄存器 (COMPMDR)

图 13-3 比较器模式设定寄存器 (COMPMDR) 的格式

地址: F0340H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
COMPMDR	C1MON	C1VRF	C1WDE	C1ENB	C0MON	C0VRF	C0WDE	C0ENB

C1MON	比较器 1 的监控标志 注 3、注 7
0	基本模式时: IVCMP1 < 比较器 1 的基准电压或者停止比较器 1 窗口模式时: IVCMP1 < 低电压的基准电压或者 IVCMP1 > 高电压的基准电压
1	基本模式时: IVCMP1 > 比较器 1 的基准电压 窗口模式时: 低电压的基准电压 < IVCMP1 < 高电压的基准电压

C1VRF	选择比较器 1 的基准电压 注 1、注 4、注 5、注 6
0	比较器 1 的基准电压为 IVREF1 输入
1	比较器 1 的基准电压为内部基准电压 (1.45 V)

C1WDE	选择比较器 1 窗口模式 注 2
0	比较器 1 基本模式
1	比较器 1 窗口模式

C1ENB	允许比较器 1 运行
0	禁止比较器 1 运行
1	允许比较器 1 运行

C0MON	比较器 0 的监控标志 注 3、注 7
0	基本模式时: IVCMP0 < 比较器 0 的基准电压或者停止比较器 0 窗口模式时: IVCMP0 < 低电压的基准电压或者 IVCMP0 > 高电压的基准电压
1	基本模式时: IVCMP0 > 比较器 0 的基准电压 窗口模式时: 低电压基准电压 < IVCMP0 < 高电压的基准电压

C0VRF	选择比较器 0 的基准电压选择 注 1、注 4、注 5、注 6
0	比较器 0 的基准电压为 IVREF0 输入
1	比较器 0 的基准电压为内部基准电压 (1.45 V)

COWDE	选择比较器 0 窗口模式注 2
0	比较器 0 正常模式
1	比较器 0 窗口模式

C0ENB	允许比较器 0 运行
0	禁止比较器 0 运行
1	允许比较器 0 运行

- 注
1. 只在在选择基本模式时有效。在窗口模式下，与此位的设定无关，选择比较器中的基准电压。
 2. 在选择低速模式时（COMPOCR 寄存器的 SPDMD 位为“0”），不能设定为窗口模式。
 3. 虽然刚解除复位后的初始值为“0”，但是，一旦在允许比较器运行后，将 C0ENB 和 C1ENB 设定为“0”，此值就变为不定值。
 4. 在 HS（高速主）模式中可选择内部基准电压（1.45 V）。在 HS（高速主）模式中选择内部基准电压（1.45 V）时，不可通过 A/D 转换器对温度传感器输出进行 A/D 转换。
 5. 在 STOP 模式中不可选择内部基准电压。
 6. 当选择副系统时钟（ f_{XT} ）为 CPU 时钟，并且高速系统时钟（ f_{MX} ）和高速内部振荡器时钟（ f_{IH} ）都停止时，不能选择内部基准电压。
 7. 忽略写入此位的值。

13.3.3 比较器滤波器控制寄存器 (COMPFIR)

图 13-4 比较器滤波器控制寄存器 (COMPFIR) 的格式

地址: F0341H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
COMPFIR	C1EDG	C1EPO	C1FCK1	C1FCK0	C0EDG	C0EPO	C0FCK1	C0FCK0

C1EDG	选择比较器 1 边沿检测 ^{注 1}
0	比较器 1 单边沿检测的中断请求
1	比较器 1 双边沿检测的中断请求

C1EPO	比较器 1 边沿极性的切换 ^{注 1}
0	比较器 1 上升沿的中断请求
1	比较器 1 下降沿的中断请求

C1FCK1	C1FCK0	选择比较器 1 滤波器 ^{注 1}
0	0	没有比较器 1 滤波器
0	1	有比较器 1 滤波器、以 f_{CLK} 采样
1	0	有比较器 1 滤波器、以 $f_{CLK}/8$ 采样
1	1	有比较器 1 滤波器、以 $f_{CLK}/32$ 采样

C0EDG	选择比较器 0 边沿检测 ^{注 2}
0	比较器 0 单边沿检测的中断请求
1	比较器 0 双边沿检测的中断请求

C0EPO	比较器 0 边沿极性的切换 ^{注 2}
0	比较器 0 上升沿的中断请求
1	比较器 0 下降沿的中断请求

C0FCK1	C0FCK0	选择比较器 0 滤波器 ^{注 2}
0	0	没有比较器 0 滤波器
0	1	有比较器 0 滤波器、以 f_{CLK} 采样
1	0	有比较器 0 滤波器、以 $f_{CLK}/8$ 采样
1	1	有比较器 0 滤波器、以 $f_{CLK}/32$ 采样

- 注 1. 如果更改 C1FCK1 位、C1FCK0 位、C1EPO 位以及 C1EDG 位, 就有可能产生比较器 1 中断请求。在更改这些位后, 必须将中断请求标志寄存器 2L (IF2L) 的 bit7 (CMPIF1) 清“0”。另外, 如果将 C1FCK1 位和 C1FCK0 位从“00B” (无比较器 1 滤波器) 更改为“00B”以外的值 (有比较器 1 滤波器), 就在经过更新滤波器输出前的 4 次采样后, 使用比较器 1 中断请求。
- 注 2. 如果更改 C0FCK1 位、C0FCK0 位、C0EPO 位以及 C0EDG 位, 就有可能产生比较器 0 中断请求。在更改这些位后, 必须将中断请求标志寄存器 2L (IF2L) 的 bit6 (CMPIF0) 清“0”。另外, 如果将 C0FCK1 位和 C0FCK0 位从“00B” (无比较器 0 滤波器) 更改为“00B”以外的值 (有比较器 0 滤波器), 就在经过更新滤波器输出前的 4 次采样后, 使用比较器 0 中断请求。

13.3.4 比较器输出控制寄存器（COMPOCR）

图 13-5 比较器输出控制寄存器（COMPOCR）的格式

地址：F0342H 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
COMPOCR	SPDMD	C1OP	C1OE	C1IE	0	C0OP	C0OE	C0IE

SPDMD	选择比较器速度注 1
0	比较器低速模式
1	比较器高速模式

C1OP	选择 VCOUT1 输出极性
0	向 VCOUT1 输出比较器 1 输出
1	向 VCOUT1 输出比较器 1 输出的反相

C1OE	允许 VCOUT1 引脚输出
0	禁止比较器 1 的 VCOUT1 引脚输出
1	允许比较器 1 的 VCOUT1 引脚输出

C1IE	允许比较器 1 中断请求 注 2
0	禁止比较器 1 中断请求
1	允许比较器 1 中断请求

C0OP	选择 VCOUT0 输出极性
0	向 VCOUT0 输出比较器 0 输出
1	向 VCOUT0 输出比较器 0 输出的反相

C0OE	允许 VCOUT0 引脚输出
0	禁止比较器 0 的 VCOUT0 引脚输出
1	允许比较器 0 的 VCOUT0 引脚输出

C0IE	允许比较器 0 中断请求注 3
0	禁止比较器 0 中断请求
1	允许比较器 0 中断请求

- 注 1. 在改写 SPDMD 位时，必须在将 COMPMDR 寄存器中的 CiENB 位（i = 0 或 1）设定为“0”后再改写。
2. 如果 C1IE 从“0”（禁止中断请求）变为“1”（允许中断请求），中断请求标志寄存器 2L（IF2L）的 bit7（CMPIF1）就有可能变为“1”（有中断请求），因此必须在将中断请求标志寄存器 2L（IF2L）的 bit7（CMPIF1）清“0”后再进行中断。
3. 如果 C0IE 从“0”（禁止中断请求）变为“1”（允许中断请求），中断请求标志寄存器 2L（IF2L）的 bit6（CMPIF0）就有可能变为“1”（有中断请求），因此必须在将中断请求标志寄存器 2L（IF2L）的 bit6（CMPIF0）清“0”后再进行中断。

13.3.5 控制模拟输入 / 输出引脚的端口功能的寄存器

将 IVREF0 引脚用作比较器的模拟输入时，必须将 LCD 端口功能寄存器 3（PFSEG3）的 bit2，端口模式寄存器 4（PM4）和端口模式控制寄存器 4（PMC4）的 bit5 设定为“1”。

将 IVCMP0 引脚、IVCMP1 引脚以及 IVREF1 引脚用作比较器的模拟输入时，必须将各端口对应的端口模式寄存器（PMxx）和端口模式控制寄存器（PMCxx）的位设定为“1”。

在使用 VCOUT0、VCOUT1 功能时，必须设定对象通道的复用端口功能的控制寄存器（端口模式寄存器（PMxx）、端口寄存器（Pxx））。

详细内容请参照“4.3.1 端口模式寄存器（PMxx）”、“4.3.2 端口寄存器（Pxx）”和“4.3.9 LCD 端口功能寄存器 0～6（PFSEG0～PFSEG6）”。

13.4 运行说明

比较器 0 和比较器 1 分别独立运行。设定方法不同，但是运行相同。比较器相关寄存器的设定步骤如表 13-2 所示。

表 13-2 比较器相关寄存器的设定步骤

顺序	寄存器	位	设定值		
1	PER1	CMPEN	1（提供输入时钟）		
2	PFSEG3	PFDEG	选择 IVCMPi 和 IVREFi 引脚的功能。		
	PMC4	PMC4n	给 PFDEG 位设定“1”（数字输入无效）（只限使用 IVREF0 的情况）。		
	PM4	PM4n	给 PMC4n 位设定“1”（模拟输入）。 给 PM4n 位设定“1”（输入模式）。		
3	COMPOCR	SPDMD	选择比较器响应速度（0：低速模式 /1：高速模式）注 1		
4	COMPMDR	CiWDE	0（基本模式）		1（窗口模式）注 2
		CiVRF	0 （基准 = IVREFi 输入）	1 （基准 = 内部基准电压（1.45 V））注 4	窗口比较器运行 （基准 = 内部 VREF）
		CiENB	1（允许运行）		
5	等待比较器稳定时间 t _{CMP}				
6	COMPFIR	CiFCK1、 CiFCK0	使用 / 不使用数字滤波器、选择选择采样时钟。		
		CiEPO、 CiEDG	选择中断请求的边沿检测条件（上升沿 / 下降沿 / 双边沿）。		
7	COMPOCR	CiOP、 CiOE	设定 VCOUTi 输出（选择极性、设定允许 / 禁止输出）。参照“13.4.3 比较器 i 输出（i=0、1）”		
		CiIE	设定允许 / 禁止中断请求输出。参照“13.4.3 比较器 i 输出（i=0、1）”。		
8	PR2L	CMPPR0i、 CMPPR1i	使用中断时：选择中断优先级。		
9	MK2L	CMPMKi	使用中断时：选择中断屏蔽。		
10	IF2L	CMPIFi	使用中断时：0（没有中断请求：初始化）注 3		

- 注 1. 不能单独设定比较器 0 和比较器 1。
 2. 只能在高速模式（SPDMD=1）中进行设定。
 3. 设定比较器后，在运行稳定之前有可能产生不必要的中断，因此须初始化中断标志。
 4. 能在 HS（高速主）模式中进行设定。

备注 i = 0、1, n = 2 ~ 5

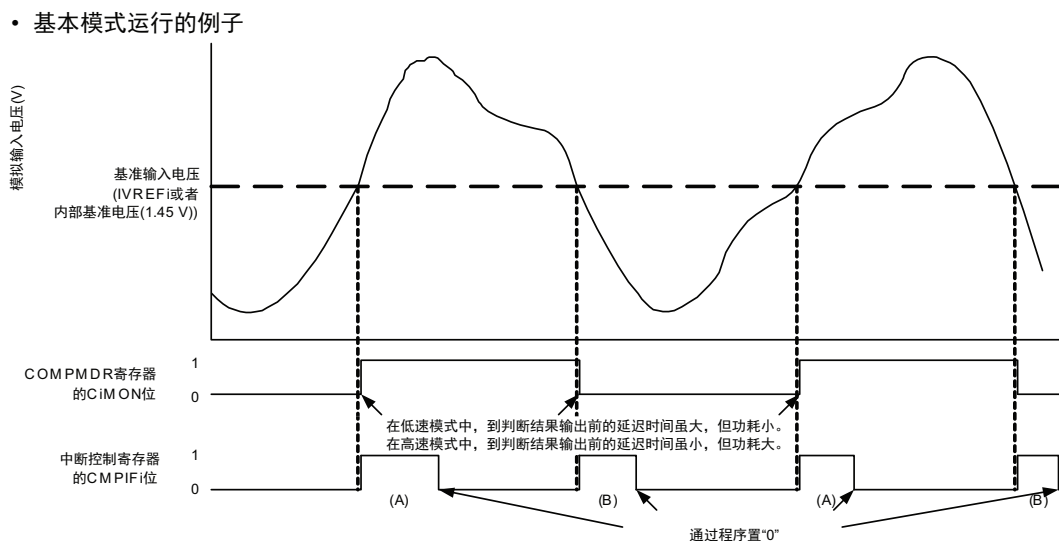
比较器 i（i=0、1）的运行例子如图 13-6 和图 13-7 所示。在基本模式中，当模拟输入电压高于基准输入电压时，COMPMDR 寄存器的 CiMON 位变为“1”，当模拟输入电压低于基准输入电压时，CiMON 位变为“0”。

在窗口模式中，当模拟输入电压满足以下条件时，COMPMDR 寄存器的 CiMON 位就变为“1”，当模拟输入电压不满足以下条件时，CiMON 位变为“0”。

低电压的基准电压 < 模拟输入电压 < 高电压的基准电压

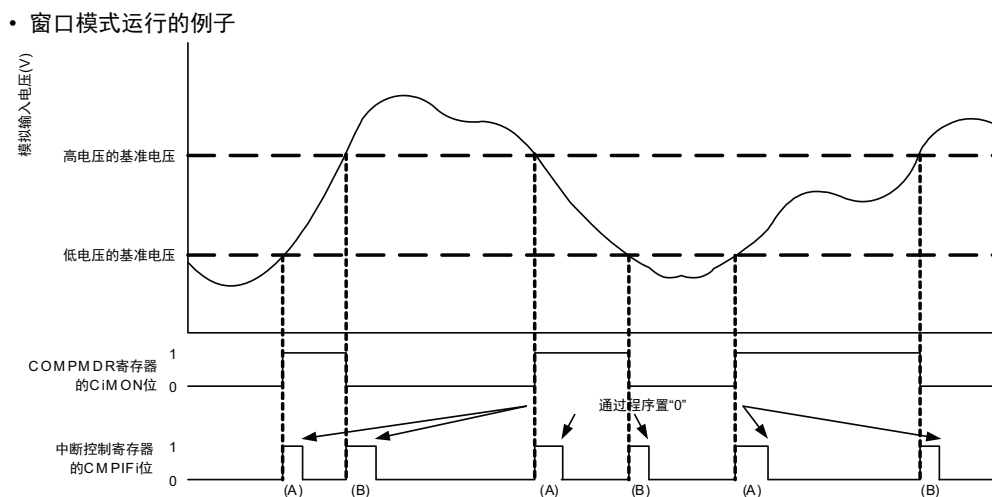
在使用比较器 i 中断时，必须将 COMPOCR 寄存器的 CiIE 位设定为“1”（允许中断请求）。如果此时比较结果发生变化，就产生比较器 i 中断请求。有关中断请求的详细内容，请参照“13.4.2 比较器 i 中断（i=0、1）”。

图 13-6 比较器 i (i = 0、1) 的运行例子 (基本模式)



注意 上图是 COMPFIR 寄存器的 CiFCK1、CiFCK0 位都为 “00B” (无滤波器) 并且 CiEDG 位为 “1” (双边沿) 的情况。(CiEDG = 0、CiEOP = 0 (上升沿) 时的 CMPIFi 只有 (A) 的变化。CiEDG = 0、CiEPO = 1 (下降沿) 时的 CMPIFi 只有 (B) 的变化。

图 13-7 比较器 i (i = 0、1) 的运行例子 (窗口模式)

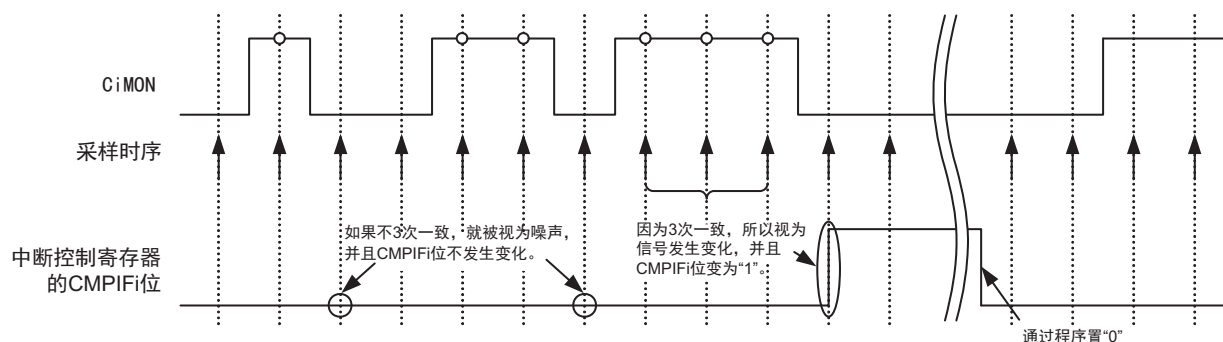


注意 上图是 COMPFIR 寄存器的 CiFCK1、CiFCK0 位都为 “00B” (无滤波器) 并且 CiEDG 位为 “1” (双边沿) 的情况。(CiEDG = 0、CiEOP = 0 (上升沿) 时的 CMPIFi 只有 (A) 的变化。CiEDG = 0、CiEPO = 1 (下降沿) 时的 CMPIFi 只有 (B) 的变化。

13.4.1 比较器 i 的数字滤波器 (i=0、1)

比较器 i 内置数字滤波器。通过 COMPFIR 寄存器的 CiFCK1 位和 CiFCK0 位选择采样时钟。各采样时钟对比较器 i 的输出信号进行采样，在输出信号的电平 3 次一致后的下一个采样时钟时，数字滤波器输出变为此值。比较器 i 的数字滤波器和中断运行的例子如图 13-8 所示。

图 13-8 比较器 i (i=0、1) 的数字滤波器和中断运行的例子



注意 上图是 COMPFIR 寄存器的 CiFCK1 ~ CiFCK0 位为 01B、10B 或者 11B 时（有数字滤波器）的运行例子。

13.4.2 比较器 i 中断 (i=0、1)

比较器产生比较器 0 和比较器 1 的这两个中断请求。比较器 i 中断分别具有 1 个优先级指定标志、中断屏蔽标志、中断请求标志和中断向量。

在使用比较器 i 中断时，必须将 COMPOCR 寄存器的 CiIE 位设定为“1”（允许中断请求输出）。通过 COMPFIR 寄存器设定产生中断请求的条件。另外，还可给比较器输出配置数字滤波器。数字滤波器可选择三种采样时钟。

寄存器设定和产生中断请求的对应，请参照“13.3.3 比较器滤波器控制寄存器（COMPFIR）”和“13.3.4 比较器输出控制寄存器（COMPOCR）”。

13.4.3 比较器 i 输出 (i=0、1)

可以向外部引脚输出比较器的比较结果。可通过 COMPOCR 寄存器的 CiOP 位和 CiOE 位设定输出极性（输出 / 取反输出）以及允许或者禁止输出。寄存器设定和比较器输出的对应，请参照“13.3.4 比较器输出控制寄存器（COMPOCR）”。

向 VCOUTi 输出引脚输出比较器的比较结果时，必须按照以下步骤进行设定（复位后，端口为输入设定。）。

1. 设定比较器的模式（“表 13-2 比较器相关寄存器的设定步骤”的步骤 2～5。）
2. 设定比较器的 VCOUTi 输出（设定 COMPOCR 寄存器、选择极性，并且设定为允许输出。）
3. 将 VCOUTi 输出引脚对应的端口模式控制寄存器的位置“0”
4. 将 VCOUTi 输出引脚对应的端口寄存器的位置“0”。
5. 将 VCOUTi 输出引脚对应的端口方向寄存器设定为输出（从引脚开始输出）。

13.4.4 停止 / 提供比较器时钟

如果要通过设定外围允许寄存器 1（PER1）来停止比较器的时钟，必须按照以下步骤进行设定。

1. 将COMPMDR寄存器的CiENB位置“0”（停止比较器）
2. 将IF2L寄存器的CMPiFi位置“0”（清除停止比较前不需要的中断）
3. 将PER1寄存器的CMPEN位置“0”

如果通过设定 PER1 停止时钟，比较器内部的寄存器就全部被初始化。再次使用比较器时，必须根据表 13-2 的步骤设定寄存器。

注意 在将比较器模式设定寄存器（COMPMDR）的比较器 n 基准电压选择位（CnVRF）置“1”（比较器 n 基准电压为内部基准电压（1.45 V））时，不能通过 A/D 转换器对温度传感器输出进行 A/D 转换。（n = 0、1）

第 14 章 串行阵列单元

串行阵列单元的每个单元最多有 4 个串行通道。各通道能实现 3 线串行（CSI）、UART 和简易 I²C 的通信功能。

R7F0C003 和 R7F0C004 支持的各通道的功能分配如下所示。

单元	通道	用作 CSI	用作 UART	用作简易 I ² C
0	0	CSI00	UART0（支持 LIN-bus）	IIC00
	1	—		—
	2	—	UART1	IIC10
	3	—		—
1	0	—	UART2	—
	1	—		—
	2	—	UART3	—
	3	—		—

在使用单元 0 的通道 0 和通道 1 的“UART0”时，不能使用 CSI00 和 IIC00，但是能使用通道 2 和通道 3 的 UART1 和 IIC10。

14.1 串行阵列单元的功能

R7F0C003 和 R7F0C004 支持的各个串行接口的特征如下所示。

14.1.1 3 线串行 I/O (CSI00)

与主控设备输出的串行时钟 (SCK) 同步进行数据的发送和接收。

这是使用 1 条串行时钟 (SCK)、1 条发送串行数据 (SO) 和 1 条接收串行数据 (SI) 共 3 条通信线进行通信的时钟同步通信功能。

有关具体的设定例子, 请参照请参照“14.5 3 线串行 I/O (CSI00) 通信的运行”。

[数据的发送和接收]

- 7 位或者 8 位的数据长度
- 发送和接收数据的相位控制
- MSB/LSB 优先的选择

[时钟控制]

- 主控或者从属的选择
- 输入/输出时钟的相位控制
- 设定由预分频器和通道内部计数器产生的传送周期。
- 最大传送速率^注

主控通信: $\text{Max. } f_{\text{CLK}}/2$

从属通信: $\text{Max. } f_{\text{MCK}}/6$

[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

CSI00 支持 SNOOZE 模式。SNOOZE 模式功能是指: 如果在 STOP 模式的状态下检测到 SCK 的输入, 就不需要 CPU 运行而接收数据。只能设定支持异步接收的 CSI00。

注 必须在满足 SCK 周期时间 (t_{KCY}) 特性的范围内使用。详细内容请参照“第 32 章 电特性”。

14.1.2 UART (UART0 ~ UART3)

这是通过串行数据发送 (TxD) 和串行数据接收 (RxD) 共 2 条线进行异步通信的功能。使用这 2 条通信线,按数据帧 (由起始位、数据、奇偶校验位和停止位构成) 与其他通信方进行异步 (使用内部波特率) 的数据发送和接收。能通过使用发送专用 (偶数通道) 和接收专用 (奇数通道) 共 2 个通道来实现全双工 UART 通信, 而且还能通过组合定时器阵列单元和外部中断 (INTP0) 来支持 LIN-bus。

有关具体的设定例子, 请参照“14.6 UART (UART0 ~ UART3) 通信的运行”。

[数据的发送和接收]

- 7位、8位或者9位的数据长度注
- MSB/LSB 优先的选择
- 发送和接收数据的电平设定、反相的选择
- 奇偶校验位的附加、奇偶校验功能
- 停止位的附加

[中断功能]

- 传送结束中断、缓冲器空中断
- 帧错误、奇偶校验错误或者溢出错误引起的错误中断

[错误检测标志]

- 帧错误、奇偶校验错误、溢出错误

UART0、UART2 接收支持 SNOOZE 模式。SNOOZE 模式功能是指: 如果在 STOP 模式的状态下检测到 RxD 的输入, 就不需要 CPU 运行而接收数据。只能对支持接收时的波特率调整功能的 UART0、UART2 进行设定。

UART0 (单元 0 的通道 0 和通道 1) 支持 LIN-bus。

[LIN-bus 功能]

• 唤醒信号的检测 • 间隔场 (BF) 的检测 • 同步场的测量、波特率的计算	}	使用外部中断 (INTP0) 和定时器阵列单元。

注 只有 UART0 和 UART2 支持 9 位的数据长度。

14.1.3 简易 I²C (IIC00、IIC10)

这是通过串行时钟 (SCL) 和串行数据 (SDA) 共 2 条线与多个设备进行时钟同步通信的功能。因为此简易 I²C 是为了与 EEPROM、闪存、A/D 转换器等设备进行单通信而设计的，所以只能用作主控设备。

开始条件和停止条件与操作控制寄存器一样，必须遵守 AC 特性，通过软件进行处理。

有关具体的设定例子，请参照“14.8 简易 I²C (IIC00、IIC10) 通信的运行”。

[数据的发送和接收]

- 主控发送、主控接收 (只限于单主控的主控功能)
- ACK 输出功能注、ACK 检测功能
- 8 位数据长度 (在发送地址时，用高 7 位指定地址，用最低位进行 R/W 控制)
- 手动产生开始条件和停止条件

[中断功能]

- 传送结束中断

[错误检测标志]

- ACK 错误、溢出错误

※ [简易 I²C 不支持的功能]

- 从属发送、从属接收
- 仲裁失败检测功能
- 等待检测功能

注 在接收最后的数据时，如果给 SOEmn 位 (串行输出允许寄存器 m (SOEm)) 写 “0” 来停止串行通信数据的输出，就不输出 ACK。详细内容请参照“14.8.3(2) 处理流程”。

备注 1. 在使用全功能 I²C 总线时，请参照“第 15 章 串行接口 IICA”。

2. m: 单元号 (m=0) n: 通道号 (n=0、2)

14.2 串行阵列单元的结构

串行阵列单元由以下硬件构成。

表 14-1 串行阵列单元的结构

项目	结构
移位寄存器	8 位或者 9 位 注 1
缓冲寄存器	串行数据寄存器 mn (SDRmn) 的低 8 位或者 9 位 注 1、注 2
串行时钟输入 / 输出	SCK00 引脚 (用于 3 线串行 I/O), SCL00、SCL10 引脚 (用于简易 I ² C)
串行数据输入	SI00 引脚 (用于 3 线串行 I/O), RxD1 ~ RxD3 引脚 (用于 UART), RxD0 引脚 (用于支持 LIN-bus 的 UART)
串行数据输出	SO00 引脚 (用于 3 线串行 I/O), TxD1 ~ TxD3 引脚 (用于 UART), TxD0 引脚 (用于支持 LIN-bus 的 UART)
串行数据输入 / 输出	SDA00、SDA10 引脚 (用于简易 I ² C)
控制寄存器	< 单元设定部的寄存器 > • 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 m (SSCm) • 输入切换控制寄存器 (ISC) • 噪声滤波器允许寄存器 0 (NFEN0) < 各通道部的寄存器 > • 串行数据寄存器 mn (SDRmn) • 串行模式寄存器 mn (SMRmn) • 串行通信运行设定寄存器 mn (SCRmn) • 串行状态寄存器 mn (SSRmn) • 串行标志清除触发寄存器 mn (SIRmn) • 端口输入模式寄存器 0、1、3 (PIM0、PIM1、PIM3) • 端口输出模式寄存器 0、1、3 (POM0、POM1、POM3) • 端口模式寄存器 0、1、3 (PM0、PM1、PM3) • 端口寄存器 0、1、3 (P0、P1、P3)

注 1. 用作移位寄存器和缓冲寄存器的位数因单元和通道而不同。

- mn = 00、01、10、11: 低 9 位
- 上述以外: 低 8 位

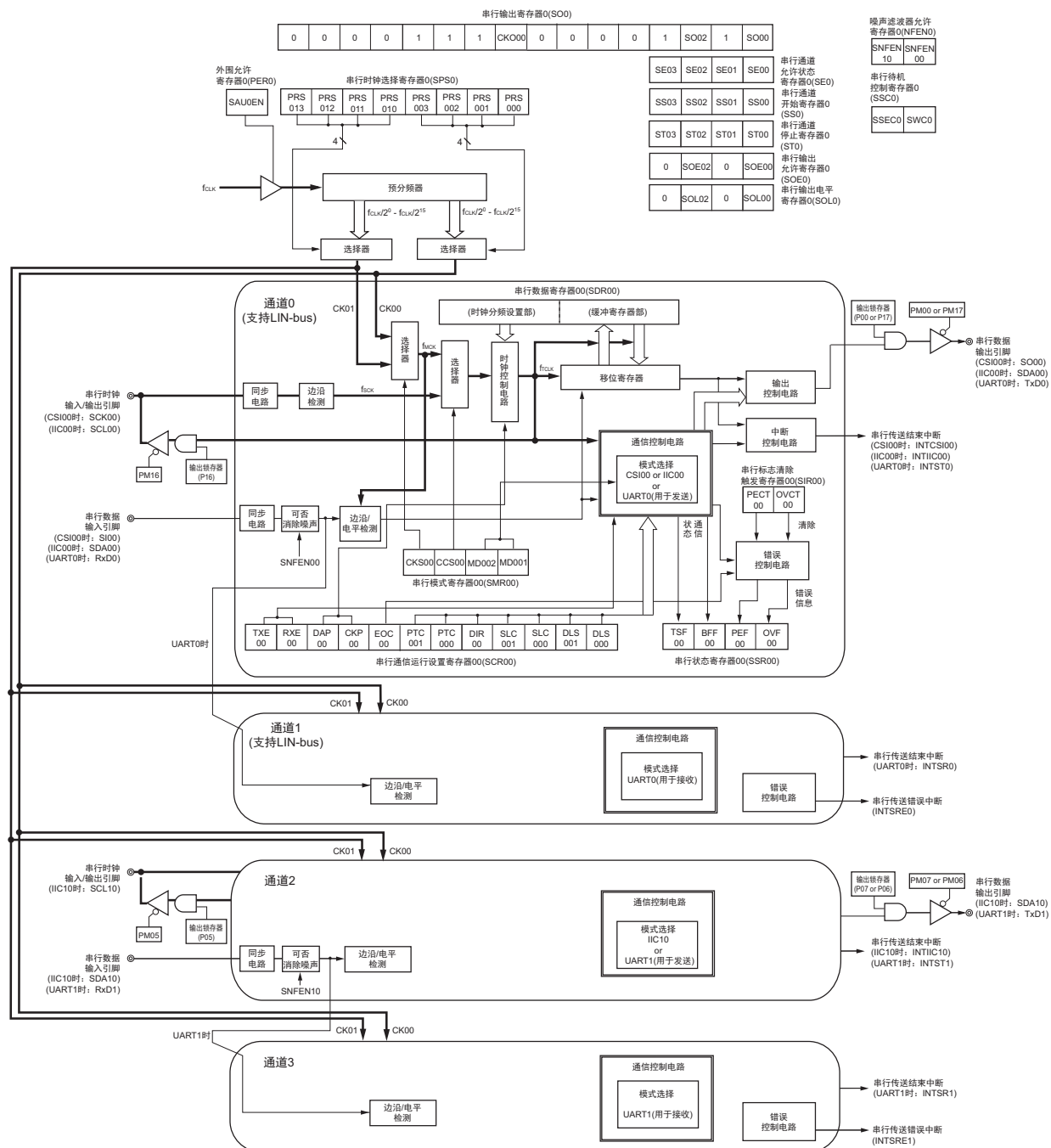
2. 根据通信模式, 能用以下 SFR 名称读写串行数据寄存器 mn (SDRmn) 的低 8 位。

- 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)
- IICr 通信 ... SIOr (IICr 数据寄存器)

备注 m: 单元号 (m = 0、1)、n: 通道号 (n = 0 ~ 3)、p: CSI 号 (p = 00)、
q: UART 号 (q = 0 ~ 3)、r: IIC 号 (r = 00、10)

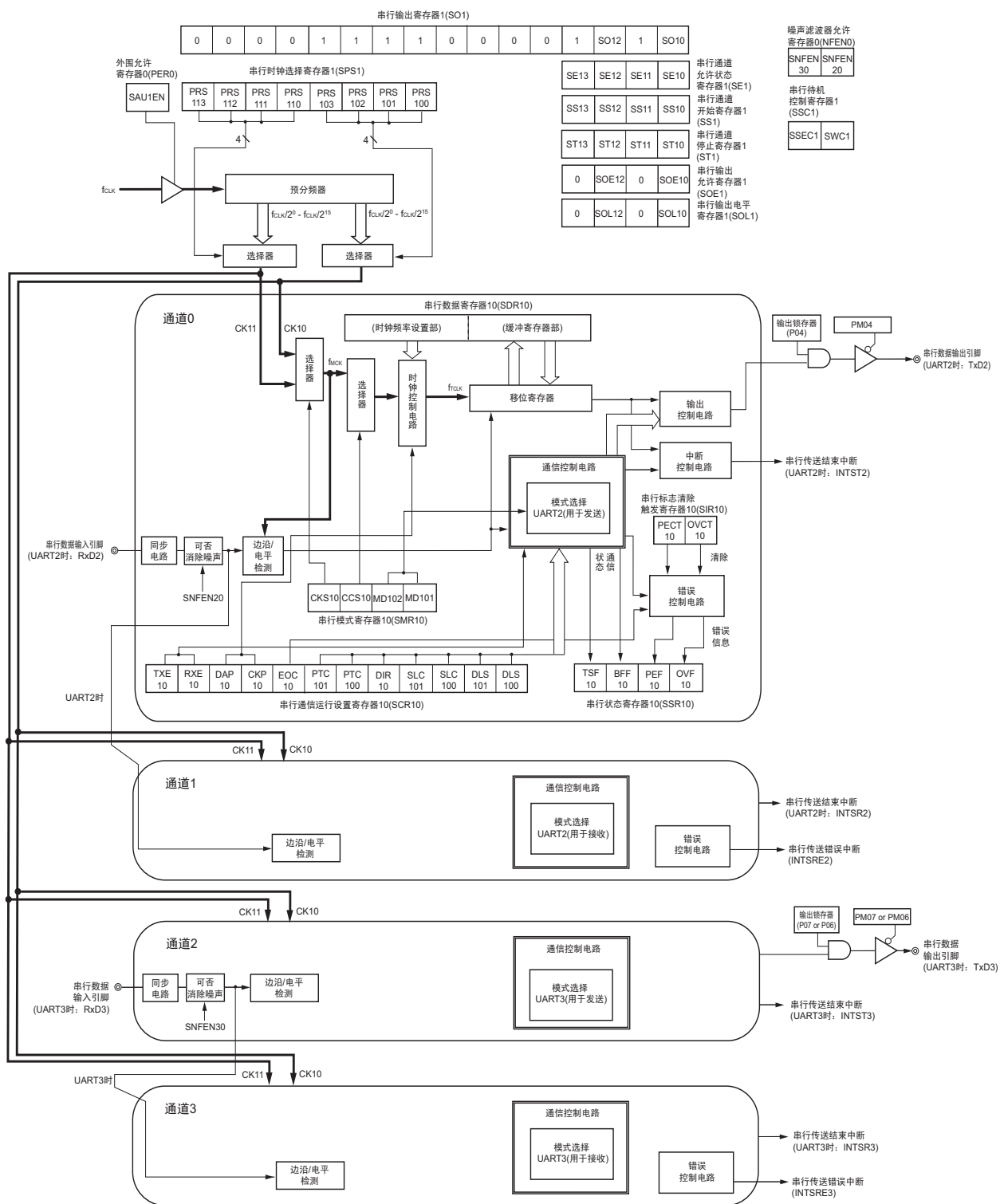
串行阵列单元 0 的框图如图 14-1 所示。

图 14-1 串行阵列单元 0 的框图



串行阵列单元 1 的框图如图 14-2 所示。

图 14-2 串行阵列单元 1 的框图



14.2.1 移位寄存器

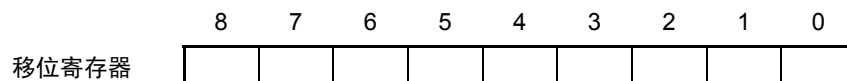
这是进行并行和串行相互转换的 9 位寄存器。

在以 9 位数据长度进行 UART 通信时，使用 9 位（bit0 ~ 8）注 1。

在接收数据时，将串行输入引脚的输入数据转换为并行数据；在发送数据时，将被传送到此寄存器的值作为串行数据从串行输出引脚输出。

不能通过程序直接操作移位寄存器。

要读写移位寄存器的数据时，使用串行数据寄存器 mn（SDRmn）的低 8 位或者低 9 位。



14.2.2 串行数据寄存器 mn（SDRmn）的低 8 位或者低 9 位

SDRmn 寄存器是通道 n 的发送和接收数据寄存器（16 位）。

bit8 ~ 0（低 9 位）注 1 或者 bit7 ~ 0（低 8 位）用作发送和接收缓冲寄存器，bit15 ~ 9 用作运行时钟（f_{MCK}）的分频设定寄存器。

在接收数据时，将由移位寄存器转换的并行数据保存到低 8 位或者低 9 位；在发送数据时，将被传送到移位寄存器的发送数据设定到低 8 位或者低 9 位。

与数据的输出顺序无关，根据串行通信运行设定寄存器 mn（SCRmn）的 bit0 和 bit1（DLSmn0、DLSmn1）的设定，保存到低 8 位或者低 9 位的数据如下所示：

- 7 位数据长度（保存在 SDRmn 寄存器的 bit0 ~ 6）
- 8 位数据长度（保存在 SDRmn 寄存器的 bit0 ~ 7）
- 9 位数据长度（保存在 SDRmn 寄存器的 bit0 ~ 8）注 1

能以 16 位为单位读写 SDRmn 寄存器。

根据通信模式，能用以下 SFR 名称，以 8 位为单位读写 SDRmn 寄存器的低 8 位或者低 9 位注 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）
- IICr 通信 SIOr（IICr 数据寄存器）

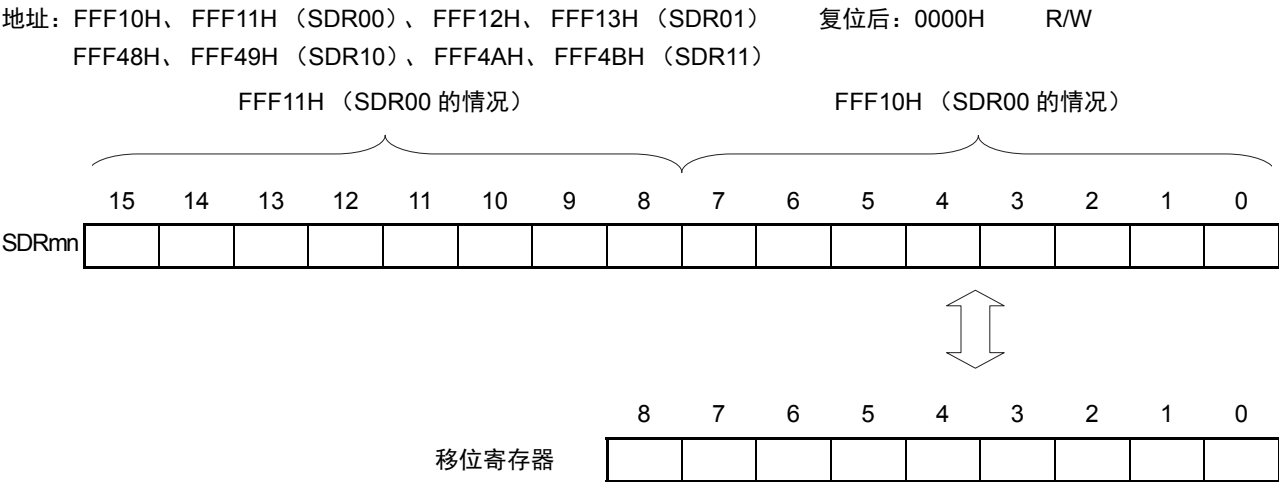
在产生复位信号后，SDRmn 寄存器的值变为“0000H”。

- 注 1. 只有 UART0 和 UART2 支持 9 位的数据长度。
2. 在停止运行（SEmn=0）时，禁止以 8 位为单位的写操作。

备注 1. 在接收结束后，bit0 ~ 8 中超过数据长度的部分的位为“0”。

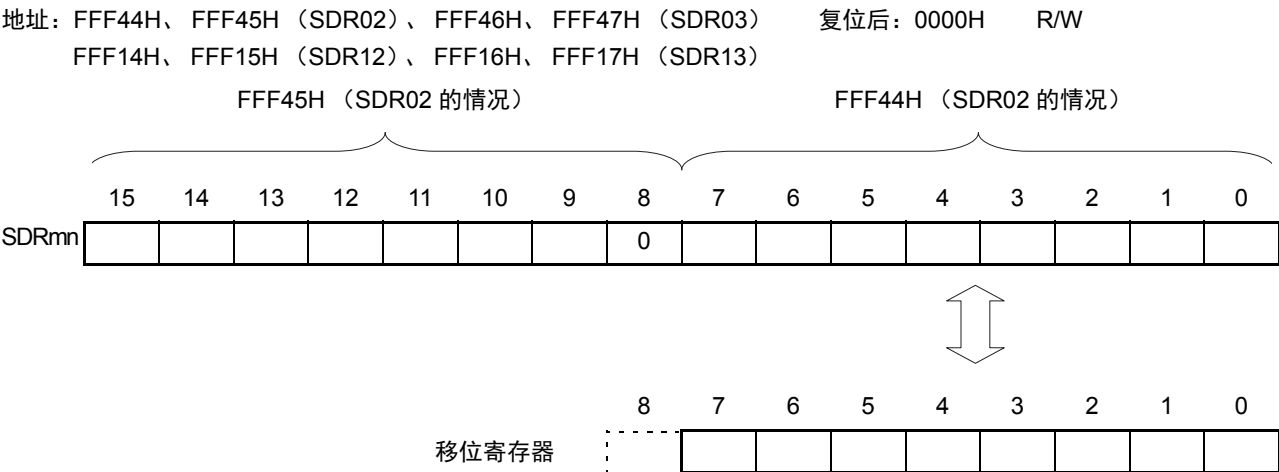
2. m: 单元号（m = 0、1）、n: 通道号（n = 0 ~ 3）、p: CSI 号（p = 00）
- q: UART 号（q = 0 ~ 3）、r: IIC 号（r = 00、10）

图 14-3 串行数据寄存器 mn（SDRmn）（mn=00、 01、 10、 11）的格式



备注 有关 SDRmn 寄存器的高 7 位的功能，请参照“14.3 控制串行阵列单元的寄存器”。

图 14-4 串行数据寄存器 mn（SDRmn）（mn=02、 03、 12、 13）的格式



注意 必须将 bit8 置“0”。

备注 有关 SDRmn 寄存器的高 7 位的功能，请参照“14.3 控制串行阵列单元的寄存器”。

14.3 控制串行阵列单元的寄存器

串行阵列单元由下列寄存器控制。

- 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)
- 串行通信运行设定寄存器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m (SSCm)
- 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、3 (PIM0、PIM1、PIM3)
- 端口输出模式寄存器0、1、3 (POM0、POM1、POM3)
- 端口模式寄存器0、1、3 (PM0、PM1、PM3)
- 端口寄存器0、1、3 (P0、P1、P3)

备注 m: 单元号 (m = 0、1)、n: 通道号 (n = 0 ~ 3)

14.3.1 外围允许寄存器 0 (PER0)

PER0 寄存器是设定允许或者禁止给各外围硬件提供时钟的寄存器。通过停止给不使用的硬件提供时钟，降低功耗和噪声。

要使用串行阵列单元 0 时，必须将 bit2 (SAU0EN) 置“1”。

要使用串行阵列单元 1 时，必须将 bit3 (SAU1EN) 置“1”。

通过 1 位或者 8 位存储器操作指令设定 PER0 寄存器。

在产生复位信号后，PER0 寄存器的值变为“00H”。

图 14-5 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	SAU1EN	SAU0EN	0	TAU0EN

SAUmEN	提供串行阵列单元 m 的输入时钟的控制
0	停止提供输入时钟。 - 不能写串行阵列单元 m 使用的 SFR。 - 串行阵列单元 m 处于复位状态。
1	允许提供输入时钟。 能读写入串行阵列单元 m 使用的 SFR。

注意 1. 在设定串行阵列单元 m 时，必须在 SAUmEN 位为“1”的状态下设定以下寄存器。如果 SAUmEN 位为“0”，串行阵列单元 m 的控制寄存器为初始值，并且忽略写操作（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、3 (PIM0、PIM1、PIM3)、端口输出模式寄存器 0、1、3 (POM0、POM1、POM3)、端口模式寄存器 0、1、3 (PM0、PM1、PM3) 以及端口寄存器 0、1、3 (P0、P1、P3) 除外）。

- 串行时钟选择寄存器 m (SPSm)
- 串行模式寄存器 mn (SMRmn)
- 串行通信运行设定寄存器 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 m (SSCm)

2. 必须将 bit1 和 bit6 设定为“0”。

14.3.2 串行时钟选择寄存器 m (SPSm)

SPSm 寄存器是 16 位寄存器，选择提供给各通道的 2 种公共运行时钟 (CKm0、CKm1)。通过 SPSm 寄存器的 bit7 ~ 4 选择 CKm1，通过 bit3 ~ 0 选择 CKm0。

禁止在运行过程中 (SEmn=1) 改写 SPSm 寄存器。

通过 16 位存储器操作指令设定 SPSm 寄存器。

能用 SPSmL 并且通过 8 位存储器操作指令设定 SPSm 寄存器的低 8 位。

在产生复位信号后，SPSm 寄存器的值变为“0000H”。

图 14-6 串行时钟选择寄存器 m (SPSm) 的格式

地址: F0126H、F0127H (SPS0)、F0166H、F0167H (SPS1) 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	运行时钟 (CKmk) 的选择注					
					f _{CLK} =2MHz	f _{CLK} =5MHz	f _{CLK} =10MHz	f _{CLK} =20MHz	f _{CLK} =24MHz
0	0	0	0	f _{CLK}	2MHz	5MHz	10MHz	20MHz	24MHz
0	0	0	1	f _{CLK} /2	1MHz	2.5MHz	5MHz	10MHz	12MHz
0	0	1	0	f _{CLK} /2 ²	500kHz	1.25MHz	2.5MHz	5MHz	6MHz
0	0	1	1	f _{CLK} /2 ³	250kHz	625kHz	1.25MHz	2.5MHz	3MHz
0	1	0	0	f _{CLK} /2 ⁴	125kHz	313kHz	625kHz	1.25MHz	1.5MHz
0	1	0	1	f _{CLK} /2 ⁵	62.5kHz	156kHz	313kHz	625kHz	750kHz
0	1	1	0	f _{CLK} /2 ⁶	31.3kHz	78.1kHz	156kHz	313kHz	375kHz
0	1	1	1	f _{CLK} /2 ⁷	15.6kHz	39.1kHz	78.1kHz	156kHz	187.5kHz
1	0	0	0	f _{CLK} /2 ⁸	7.81kHz	19.5kHz	39.1kHz	78.1kHz	93.8kHz
1	0	0	1	f _{CLK} /2 ⁹	3.91kHz	9.77kHz	19.5kHz	39.1kHz	46.9kHz
1	0	1	0	f _{CLK} /2 ¹⁰	1.95kHz	4.88kHz	9.77kHz	19.5kHz	23.4kHz
1	0	1	1	f _{CLK} /2 ¹¹	977Hz	2.44kHz	4.88kHz	9.77kHz	11.7kHz
1	1	0	0	f _{CLK} /2 ¹²	488Hz	1.22kHz	2.44kHz	4.88kHz	5.86kHz
1	1	0	1	f _{CLK} /2 ¹³	244Hz	610Hz	1.22kHz	2.44kHz	2.93kHz
1	1	1	0	f _{CLK} /2 ¹⁴	122Hz	305Hz	610Hz	1.22kHz	1.46kHz
1	1	1	1	f _{CLK} /2 ¹⁵	61Hz	153kHz	305Hz	610Hz	732Hz

注 要在串行阵列单元 (SAU) 运行过程中更改 f_{CLK} 选择的时钟 (更改系统时钟控制寄存器 (CKC) 的值) 时，必须在停止 SAU 的运行 (串行通道停止寄存器 m (STm) =000FH) 后进行更改。

注意 必须将 bit15 ~ 8 置“0”。

备注 1. f_{CLK}: CPU/ 外围硬件的时钟频率

2. m: 单元号 (m=0、1)

3. k=0、1

14.3.3 串行模式寄存器 mn (SMRmn)

SMRmn 寄存器是设定通道 n 运行模式的寄存器，进行运行时钟 (f_{MCK}) 的选择、能否使用串行时钟 (f_{SCK}) 输入的指定、开始触发的设定、运行模式 (CSI、UART、简易 I²C) 的设定以及中断源的选择。另外，只在 UART 模式中设定接收数据的反相电平。

禁止在运行过程中 (SEmn=1) 改写 SMRmn 寄存器，但是能在运行过程中改写 MDmn0 位。

通过 16 位存储器操作指令设定 SMRmn 寄存器。

在产生复位信号后，SMRmn 寄存器的值变为“0020H”。

图 14-7 串行模式寄存器 mn (SMRmn) 的格式 (1/2)

地址: F0110H、F0111H (SMR00) ~ F0116H、F0117H (SMR03) 复位后: 0020H R/W
F0150H、F0151H (SMR10) ~ F0156H、F0157H (SMR13)

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注	0	SISmn 0注	1	0	0	MD mn2	MD mn1	MD mn0

CKS mn	通道 n 运行时钟 (f_{MCK}) 的选择
0	SPSm 寄存器设定的运行时钟 CKm0
1	SPSm 寄存器设定的运行时钟 CKm1
运行时钟 (f_{MCK}) 用于边沿检测电路。通过设定 CCSmn 位和 SDRmn 寄存器的高 7 位，生成传送时钟 (f_{TCLK})。	

CCS mn	通道 n 传送时钟 (f_{TCLK}) 的选择
0	CKSmn 位指定的运行时钟 f_{MCK} 的分频时钟
1	来自 SCKp 引脚的输入时钟 f_{SCK} (CSI 模式的从属传送)
传送时钟 f_{TCLK} 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 位为“0”时，通过 SDRmn 寄存器的高 7 位进行运行时钟 (f_{MCK}) 的分频设定。	

STS mn注	开始触发源的选择
0	只有软件触发有效 (在 CSI、UART 发送、简易 I ² C 时选择)。
1	RxDq 引脚的有效边沿 (在 UART 接收时选择)
在将 SSm 寄存器置“1”后满足上述条件时，开始传送。	

注 只限 SMR01、SMR03、SMR11、SMR13 寄存器。

注意 必须将 bit13 ~ 9、7、4、3 (在 SMR00、SMR02、SMR10、SMR12 寄存器时，为 bit13 ~ 6、4、3) 置“0”，并且将 bit5 置“1”。

备注 m: 单元号 (m = 0、1)、n: 通道号 (n = 0 ~ 3)、p: CSI 号 (p = 00)

q: UART 号 (q = 0 ~ 3)、r: IIC 号 (r = 00、10)

图 14-7 串行模式寄存器 mn（SMRmn）的格式（2/2）

地址：F0110H、F0111H（SMR00）～F0116H、F0117H（SMR03） 复位后：0020H R/W
F0150H、F0151H（SMR10）～F0156H、F0157H（SMR13）

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注	0	SISmn 0注	1	0	0	MD mn2	MD mn1	MD mn0

SISmn 0注	UART 模式中的通道 n 接收数据的电平反相控制
0	将下降沿检测为起始位。 不将输入的通信数据进行反相。
1	将上升沿检测为起始位。 将输入的通信数据进行反相。

MD mn2	MD mn1	通道 n 运行模式的设定
0	0	CSI 模式
0	1	UART 模式
1	0	简易 I ² C 模式
1	1	禁止设定。

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 (在数据从 SDRmn 寄存器传送到移位寄存器时发生。)
在连续发送时，如果 MDmn0 位为“1”并且 SDRmn 的数据为空，就写下一个发送数据。	

注 只限 SMR01、SMR03、SMR11、SMR13 寄存器。

注意 必须将 bit13～9、7、4、3（在 SMR00、SMR02、SMR10、SMR12 寄存器时，为 bit13～6、4、3）置“0”，并且将 bit5 置“1”。

备注 m：单元号（m = 0、1）、n：通道号（n = 0～3）、p：CSI 号（p = 00）
q：UART 号（q = 0～3）、r：IIC 号（r = 00、10）

14.3.4 串行通信操作设定寄存器 mn (SCRmn)

SCRmn 寄存器是通道 n 的通信运行设定寄存器，设定数据发送和接收模式、数据和时钟相位、是否屏蔽错误信号、奇偶校验位、起始位、停止位和数据长度等。

禁止在运行过程中 (SEmn=1) 改写 SCRmn 寄存器。

通过 16 位存储器操作指令设定 SCRmn 寄存器。

在产生复位信号后，SCRmn 寄存器的值变为“0087H”。

图 14-8 串行通信运行设定寄存器 mn (SCRmn) 的格式 (1/2)

地址: F0118H、F0119H (SCR00) ~ F011EH、F011FH (SCR03) 复位后: 0087H R/W
F0158H、F0159H (SCR10) ~ F015EH、F015FH (SCR13)

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 注1	SLC mn0	0	1	DLSm n1 注2	DLS mn0

TXE mn	RXE mn	通道 n 运行模式的设定
0	0	禁止通信
0	1	只接收
1	0	只发送
1	1	发送和接收

DAP mn	CKP mn	CSI 模式中的数据和时钟的相位选择	类型
0	0		1
0	1		2
1	0		3
1	1		4

在 UART 模式和简易 I²C 模式中，必须将 DAPmn 位和 CKPmn 位都置“0”。

EOC mn	错误中断信号 (INTSREx (x=0 ~ 3)) 的屏蔽控制
0	禁止发生错误中断 INTSREx (发生 INTSRx)。
1	允许发生错误中断 INTSREx (在发生错误时不发生 INTSRx)。

在 CSI 模式和简易 I²C 模式中或者在 UART 发送时，必须将 EOCmn 位置“0”注3。

- 注 1. 只限 SCR00、SCR02、SCR10、SCR12 寄存器。
2. 只限 SCR00、SCR01、SCR10、SCR11 寄存器，其他固定为“1”。
3. 在 EOCmn 位为“0”并且不使用 CSImn 时，有可能产生错误中断 INTSREn。

注意 必须将 bit3、6、11 置“0” (也必须将 SCR01、SCR03、SCR11、SCR13 寄存器的 bit5 置“0”)，并且将 bit2 置“1”。

备注 m: 单元号 (m = 0、1)、n: 通道号 (n = 0 ~ 3)、p: CSI 号 (p = 00)

图 14-8 串行通信运行设定寄存器 mn (SCRmn) 的格式 (2/2)

地址: F0118H、F0119H (SCR00) ~ F011EH、F011FH (SCR03) 复位后: 0087H R/W
F0158H、F0159H (SCR10) ~ F015EH、F015FH (SCR13)

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 注1	SLC mn0	0	1	DLSm n1 注2	DLS mn0

PTC mn1	PTC mn0	UART 模式中的奇偶校验位的设定	
		发送	接收
0	0	不输出奇偶校验位。	接收时没有奇偶校验。
0	1	输出零校验注 3。	不判断奇偶校验。
1	0	输出偶校验。	判断偶校验。
1	1	输出奇校验。	判断奇校验。

在 CSI 模式和简易 I²C 模式中, 必须将 PTCmn1 位和 PTCmn0 位都置“0”。

DIR mn	CSI 和 UART 模式中的数据传送顺序的选择
0	MSB 优先的输入 / 输出
1	LSB 优先的输入 / 输出

在简易 I²C 模式中, 必须将 DIRmn 位置“0”。

SLCm n1 注1	SLC mn0	UART 模式中的停止位的设定
0	0	无停止位
0	1	停止位长度 =1 位
1	0	停止位长度 =2 位 (只限于 mn=00、02、10、12)
1	1	禁止设定。

如果选择了传送结束中断, 就在传送完所有停止位后产生中断。在 UART 接收时或者在简易 I²C 模式中, 必须设定为 1 个停止位 (SLCmn1、SLCmn0=0、1)。在 CSI 模式中, 必须设定为无停止位 (SLCmn1、SLCmn0=0、0)。在 UART 发送时, 设定为 1 个 (SLCmn1、SLCmn0=0、1) 或者 2 个 (SLCmn1、SLCmn0=1、0) 停止位。

DLSm n1 注2	DLS mn0	CSI 和 UART 模式中的数据长度的设定
0	1	9 位数据长度 (保存到 SDRmn 寄存器的 bit0 ~ 8) (只在 UART 模式中可选择)
1	0	7 位数据长度 (保存到 SDRmn 寄存器的 bit0 ~ 6)
1	1	8 位数据长度 (保存到 SDRmn 寄存器的 bit0 ~ 7)
其他		禁止设定。

在简易 I²C 模式中, 必须将 DLSmn1 位和 DLSmn0 位都置“1”。

- 注 1. 只限于 SCR00、SCR02、SCR10、SCR12 寄存器。
2. 只限于 SCR00、SCR01、SCR10 和 SCR11 寄存器, 其他固定为“1”。
3. 与数据的内容无关, 总是附加“0”。

注意 必须将 bit3、6、11 置“0” (也必须将 SCR01、SCR03、SCR11、SCR13 寄存器的 bit5 置“0”), 并且将 bit2 置“1”。

备注 m: 单元号 (m = 0、1)、n: 通道号 (n = 0 ~ 3)、p: CSI 号 (p = 00)

14.3.5 串行数据寄存器 mn (SDRmn)

SDRmn 寄存器是通道 n 发送和接收的数据寄存器 (16 位)。

SDR00、SDR01、SDR10、SDR11 的 bit8 ~ 0 (低 9 位) 或者 SDR02、SDR03、SDR12、SDR13 的 bit7 ~ 0 (低 8 位) 用作发送和接收缓冲寄存器, bit15 ~ 9 用作运行时钟 (f_{MCK} 、 f_{SCK}) 的分频设定寄存器。

如果将串行模式寄存器 mn (SMRmn) 的 CCSmn 位置“0”, 由 SDRmn 寄存器的高 7 位设定的运行时钟的分频时钟就用作传送时钟。

如果将 CCSmn 位置“1”, 就必须将 SDR00、SDR01、SDR10、SDR11 的 bit15 ~ 9 (高 7 位) 置“0000000B”。SCKp 引脚的输入时钟 f_{SCK} (CSI 模式的从属传送) 为传送时钟。

SDRmn 寄存器的低 8 位或者低 9 位用作发送和接收缓冲寄存器。在接收数据时, 将移位寄存器转换的并行数据保存到低 8 位或者低 9 位; 在发送数据时, 将被传送到移位寄存器的发送数据设定到低 8 位或者低 9 位。

能以 16 位为单位读写 SDRmn 寄存器。但是, 只有在运行停止状态 (SEmn=0) 时才能读写高 7 位。在运行中 (SEmn=1) 只能写 SDRmn 寄存器的低 8 位或者低 9 位, 而且运行中 SDRmn 寄存器的读取值总是为“0”。

在产生复位信号后, SDRmn 寄存器的值变为“0000H”。

图 14-9 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
FFF48H、FFF49H (SDR10)、FFF4AH、FFF4BH (SDR11)

FFF11H (SDR00 的情况)

FFF10H (SDR00 的情况)

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

地址: FFF44H、FFF45H (SDR02)、FFF46H、FFF47H (SDR03) 复位后: 0000H R/W
FFF14H、FFF15H (SDR12)、FFF16H、FFF17H (SDR13)

FFF45H (SDR02 的情况)

FFF44H (SDR02 的情况)

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

SDRmn[15:9]							运行时钟分频的传送时钟设定	
0	0	0	0	0	0	0	$f_{MCK}/2$	
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 1. 必须将 SDR02、SDR03、SDR12、SDR13 寄存器的 bit8 置“0”。

2. 在使用 UART 时，禁止设定为 SDRmn[15:9]= (0000000B、0000001B)。
3. 在使用简易 I²C 时，禁止设定为 SDRmn[15:9]=0000000B，必须设定为 SDRmn[15:9] ≥ 0000001B。
4. 在运行停止状态 (SEmn=0) 时，不能以 8 位为单位写低 8 位 (否则，高 7 位被清“0”)。

备注 1. 有关 SDRmn 寄存器的低 8 位或者低 9 位的功能，请参照“14.2 串行阵列单元的结构”。

2. m: 单元号 (m=0、1) n: 通道号 (n=0 ~ 3)

14.3.6 串行标志清除触发寄存器 mn（SIRmn）

这是用于清除通道 n 各错误标志的触发寄存器。

如果将各位（FECTmn、PECTmn、OVCTmn）置“1”，就将串行状态寄存器 mn（SSRmn）的对应位（FEFmn、PEFmn、OVFmn）清“0”。因为 SIRmn 寄存器是触发寄存器，所以如果清除 SSRmn 寄存器的对应位，也会立即清除 SIRmn 寄存器。

通过 16 位存储器操作指令设定 SIRmn 寄存器。

能用 SIRmnL 并且通过 8 位存储器操作指令设定 SIRmn 寄存器的低 8 位。

在产生复位信号后，SIRmn 寄存器的值变为“0000H”。

图 14-10 串行标志清除触发寄存器 mn（SIRmn）的格式

地址: F0108H、F0109H (SIR00) ~ F010EH、F010FH (SIR03) 复位后: 0000H R/W

F0148H、F0149H (SIR10) ~ F014EH、F014FH (SIR13)

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 帧错误标志的清除触发
0	不清除。
1	将 SSRmn 寄存器的 FEFmn 位清“0”。

PEC Tmn	通道 n 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“0”。

OVC Tmn	通道 n 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“0”。

注 只限 SIR01、SIR03、SIR11、SIR13 寄存器。

注意 必须将 bit15～3（在 SIR00、SIR02、SIR10、SIR12 寄存器时，为 bit15～2）置“0”。

备注 1. m：单元号（m=0、1） n：通道号（n=0～3）

2. SIRmn 寄存器的读取值总是“0000H”。

14.3.7 串行状态寄存器 mn (SSRmn)

SSRmn 寄存器表示通道 n 的通信状态和发生错误的情况。表示的错误为帧错误、奇偶校验错误和溢出错误。

通过 16 位存储器操作指令读取 SSRmn 寄存器。

能用 SSRmnL 并且通过 8 位存储器操作指令读取 SSRmn 寄存器的低 8 位。

在产生复位信号后，SSRmn 寄存器的值变为“0000H”。

图 14-11 串行状态寄存器 mn (SSRmn) 的格式 (1/2)

地址: F0100H、F0101H (SSR00) ~ F0106H、F0107H (SSR03) 复位后: 0000H R
F0140H、F0141H (SSR10) ~ F0146H、F0147H (SSR13)

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	FEF mn注	PEF mn	OVF mn

TSF mn	通道 n 通信状态的表示标志
0	通信处于停止或者待机状态。
1	通信处于运行状态。

[清除条件]

- 当将 STm 寄存器的 STmn 位置 “1”（通信停止状态）或者将 SSm 寄存器的 SSmn 位置 “1”（通信待机状态）时
- 当通信结束时

[置位条件]

- 当开始通信时

BFF mn	通道 n 缓冲寄存器的状态表示标志
0	SDRmn 寄存器没有保存有效数据。
1	SDRmn 寄存器保存了有效数据。

[清除条件]

- 在发送过程中将发送数据从 SDRmn 寄存器传送到移位寄存器时
- 在接收过程中从 SDRmn 寄存器读到接收数据时
- 将 STm 寄存器的 STmn 位置“1”（通信停止状态）或者将 SSm 寄存器的 SSmn 位置“1”（通信允许状态）。

[置位条件]

- 在 SCRmn 寄存器的 TXEmn 位为“1”（各通信模式中的发送模式、发送和接收模式）的状态下给 SDRmn 寄存器写发送数据时
- 在 SCRmn 寄存器的 RXEmn 位为“1”（各通信模式中的接收模式、发送和接收模式）的状态下将接收数据保存到 SDRmn 寄存器时
- 当发生接收错误时

注 只限 SSR01、SSR03、SSR11、SSR13 寄存器。

注意 如果在 BFFmn=1 时写 SDRmn 寄存器，就破坏已保存的发送 / 接收数据，并且检测到溢出错误（OVermn=1）。

备注 m: 单元号 (m=0、1) n: 通道号 (n=0 ~ 3)

图 14-11 串行状态寄存器 mn (SSRmn) 的格式 (2/2)

地址: F0100H、F0101H (SSR00) ~ F0106H、F0107H (SSR03) 复位后: 0000H R
 F0140H、F0141H (SSR10) ~ F0146H、F0147H (SSR13)

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	FEF mn ^注	PEF mn	OVF mn

FEFmn	通道 n 帧错误的检测标志
0	没有发生错误。
1	发生错误 (UART 接收时)。
[清除条件]	
• 当给 SIRmn 寄存器的 FECTmn 位写“1”时。	
[置位条件]	
• 在 UART 接收结束时没有检测到停止位时	

PEFmn	通道 n 奇偶校验 /ACK 错误的检测标志
0	没有发生错误。
1	发生奇偶校验错误 (UART 接收时) 或者未检测到 ACK (I ² C 发送时)。
[清除条件]	
• 当给 SIRmn 寄存器的 PECTmn 位写“1”时。	
[置位条件]	
• 在 UART 接收结束时发送数据的奇偶校验和奇偶校验位不同 (奇偶校验错误) 时	
• 在 I ² C 发送时并且在 ACK 接收时序从属方没有返回 ACK 信号 (未检测到 ACK) 时。	

OVFmn	通道 n 溢出错误的检测标志
0	没有发生错误。
1	发生错误。
[清除条件]	
• 当给 SIRmn 寄存器的 OVCTmn 位写“1”时。	
[置位条件]	
• 在 SCRmn 寄存器的 RXEmn 位为“1” (各通信模式中的接收模式、发送和接收模式) 的状态下, 虽然接收数据被保存在 SDRmn 寄存器中, 但是没有读接收数据而写发送数据或者写下一个接收数据时	
• 在 CSI 模式的从属发送或者从属发送和接收过程中未准备好发送数据时	

注 只限 SSR01、SSR03、SSR11、SSR13 寄存器。

备注 m: 单元号 (m=0、1) n: 通道号 (n=0 ~ 3)

14.3.8 串行通道开始寄存器 m（SSm）

SSm 寄存器是设定允许各通道的通信 / 开始计数的触发寄存器。

如果给各位（SSmn）写“1”，就将串行通道允许状态寄存器 m（SEm）的对应位（SEmn）置“1”（运行允许状态）。因为 SSmn 位是触发位，所以如果 SEmn 位为“1”就立即清除 SSmn 位。

通过 16 位存储器操作指令设定 SSm 寄存器。

能用 SSmL 并且通过 1 位或者 8 位存储器操作指令设定 SSm 寄存器的低 8 位。

在产生复位信号后，SSm 寄存器的值变为“0000H”。

图 14-12 串行通道开始寄存器 m（SSm）的格式

地址: F0122H、F0123H (SS0)

复位后: 0000H

R/W

符号

15

14

13

12

11

10

9

8

7

6

5

4

3

2

1

0

SS0

0

0

0

0

0

0

0

0

0

0

0

0

SS03

SS02

SS01

SS00

地址: F0162H、F0163H (SS1)

复位后: 0000H

R/W

符号

15

14

13

12

11

10

9

8

7

6

5

4

3

2

1

0

SS1

0

0

0

0

0

0

0

0

0

0

0

0

SS13

SS12

SS11

SS10

SSmn	通道 n 的运行开始触发
0	没有触发。
1	将 SEmn 位置“1”，转移到通信待机状态注。

注 如果在通信中将 SSmn 位置“1”，就停止通信进入待机状态。此时，控制寄存器和移位寄存器的值、SCKmn 引脚和 SOMn 引脚、FEFmn 标志、PEFmn 标志和 OV Fmn 标志保持状态。

- 注意 1. 必须将 SS0 寄存器和 SS1 寄存器的 bit15 ~ 4 置“0”。
2. 在 UART 接收时，必须在将 SCRmn 寄存器的 RXEmn 位置“1”后至少间隔 4 个 f_{MCK} 时钟，然后将 SSmn 置“1”。

备注 1. m: 单元号 (m=0、1) n: 通道号 (n=0 ~ 3)

2. SSm 寄存器的读取值总是“0000H”。

14.3.9 串行通道停止寄存器 m（STm）

STm 寄存器是设定允许各通道的通信 / 停止计数的触发寄存器。

如果给各位（STmn）写“1”，就将串行通道允许状态寄存器 m（SEm）的对应位（SEmn）清“0”（运行停止状态）。因为 STmn 位是触发位，所以如果 SEmn 位为“0”就立即清除 STmn 位。

通过 16 位存储器操作指令设定 STm 寄存器。

能用 STmL 并且通过 1 位或者 8 位存储器操作指令设定 STm 寄存器的低 8 位。

在产生复位信号后，STm 寄存器的值变为“0000H”。

图 14-13 串行通道停止寄存器 m（STm）的格式

地址：F0124H、F0125H （ST0）

复位后：0000H

R/W

符号

15

14

13

12

11

10

9

8

7

6

5

4

3

2

1

0

ST0

0

0

0

0

0

0

0

0

0

0

0

0

ST03

ST02

ST01

ST00

地址：F0164H、F0165H （ST1）

复位后：0000H

R/W

符号

15

14

13

12

11

10

9

8

7

6

5

4

3

2

1

0

ST1

0

0

0

0

0

0

0

0

0

0

0

0

ST13

ST12

ST11

ST10

STmn	通道 n 的运行停止触发
0	没有触发。
1	将 SEmn 位清“0”，停止通信运行注。

注 控制寄存器和移位寄存器的值、SCKmn 引脚和 SOMn 引脚以及 FEFmn 标志、PEFmn 标志和 OVFmn 标志保持状态。

注意 必须将 ST0 寄存器和 ST1 寄存器的 bit15 ~ 4 置“0”。

备注 1. m: 单元号 (m=0、1) n: 通道号 (n=0 ~ 3)
2. STm 寄存器的读取值总是“0000H”。

14.3.10 串行通道允许状态寄存器 m（SEm）

SEm 寄存器用于确认各通道的串行发送和接收的允许或者停止状态。

如果给串行开始允许寄存器 m（SSm）的各位写“1”，就将其对应的位置“1”。如果给串行通道停止寄存器 m（STm）的各位写“1”，就将其对应的位清“0”。

对于允许运行的通道 n，无法通过软件改写后述的串行输出寄存器 m（SOm）的 CKOmn 位（通道 n 的串行时钟输出）的值，而从串行时钟引脚输出由通信运行反映的值。

对于停止运行的通道 n，能通过软件设定 SOm 寄存器的 CKOmn 位的值，并且从串行时钟引脚输出该值。从而，能通过软件生成开始条件或者停止条件等的任意波形。

通过 16 位存储器操作指令读取 SEm 寄存器。

能用 SEmL 并且通过 1 位或者 8 位存储器操作指令读取 SEm 寄存器的低 8 位。

在产生复位信号后，SEm 寄存器的值变为“0000H”。

图 14-14 串行通道允许状态寄存器 m（SEm）的格式

地址: F0120H、 F0121H (SE0)	复位后: 0000H	R
符号	1514131211109876543210	
SE0	00000000000000	SE03SE02SE01SE00

地址: F0160H、 F0161H (SE1)	复位后: 0000H	R
符号	1514131211109876543210	
SE1	00000000000000	SE13SE12SE11SE10

SEmn	通道 n 运行的允许或者停止状态的表示
0	运行停止状态。
1	运行允许状态。

备注 m: 单元号 (m=0、 1) n: 通道号 (n=0 ~ 3)

14.3.11 串行输出允许寄存器 m（SOEm）

SOEm 寄存器设定允许或者停止各通道的串行通信的输出。

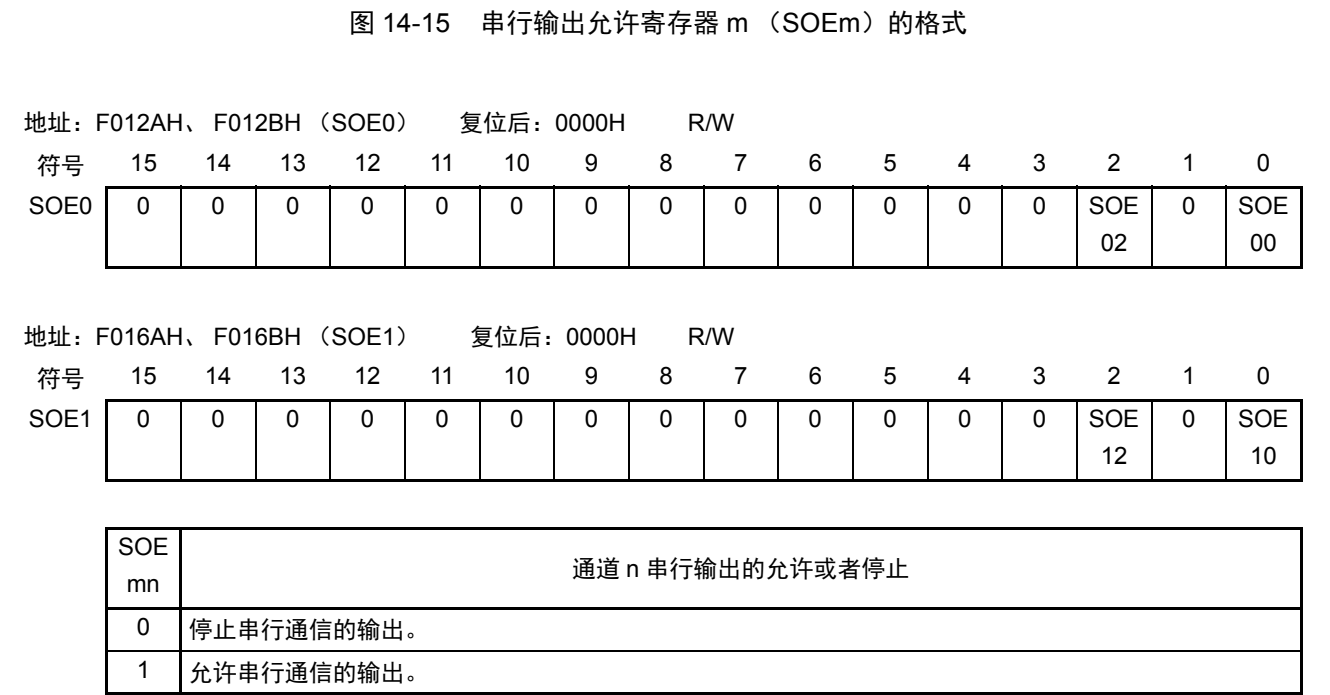
对于允许串行输出的通道 n，无法通过软件改写后述的串行输出寄存器 m（SOm）的 SOmn 位的值，而从串行数据输出引脚输出由通信运行反映的值。

对于停止串行输出的通道 n，能通过软件设定 SOm 寄存器的 SOmn 位的值，并且从串行数据输出引脚输出该值。从而，能通过软件生成开始条件或者停止条件等的任意波形。

通过 16 位存储器操作指令设定 SOEm 寄存器。

能用 SOEmL 并且通过 1 位或者 8 位存储器操作指令设定 SOEm 寄存器的低 8 位。

在产生复位信号后，SOEm 寄存器的值变为“0000H”。



注意 必须将 SOE0 寄存器和 SOE1 寄存器的 bit15～3、bit1 置“0”。

备注 m: 单元号（m=0、1） n: 通道号（n=0、2）

14.3.12 串行输出寄存器 m (SOm)

SOm 寄存器是各通道串行输出的缓冲寄存器。

从通道 n 的串行数据输出引脚输出此寄存器的 SOmn 位的值。

从通道 n 的串行时钟输出引脚输出此寄存器的 CKOmn 位的值。

只有在禁止串行输出时 (SOEmn=0) 才能通过软件改写此寄存器的 SOmn 位。当允许串行输出 (SOEmn=1) 时，忽视通过软件的改写而只能通过串行通信更改此寄存器的 SOmn 位的值。

只有在停止通道运行时 (SEmn=0) 才能通过软件改写此寄存器的 CKOmn 位。当允许通道运行 (SEmn=1) 时，忽视通过软件的改写而只能通过串行通信更改此寄存器的 CKOmn 位的值。

要将串行接口引脚用作端口功能时，必须将相应的 CKOmn 位和 SOmn 位置“1”。

通过 16 位存储器操作指令设定 SOm 寄存器。

在产生复位信号后，SOm 寄存器的值变为“0F0FH”。

图 14-16 串行输出寄存器 m (SOm) 的格式

地址: F0128H、 F0129H (SO0)		复位后: 0F0FH		R/W												
符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SO0	0	0	0	0	1	1	1	CKO00	0	0	0	0	1	SO02	1	SO00

地址: F0168H、 F0169H (SO1)		复位后: 0F0FH		R/W												
符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SO1	0	0	0	0	1	1	1	1	0	0	0	0	1	SO12	1	SO10

CKOmn	通道 n 的串行时钟输出														
0	串行时钟的输出值为“0”。														
1	串行时钟的输出值为“1”。														

SOmn	通道 n 的串行数据输出														
0	串行数据的输出值为“0”。														
1	串行数据的输出值为“1”。														

注意 必须将 SO0 寄存器的 bit15 ~ 12 和 bit7 ~ 4 置“0”、bit11 ~ 9、3、1 置“1”。
必须将 SO1 寄存器的 bit15 ~ 12 和 bit7 ~ 4 置“0”、bit11 ~ 8、3、1 置“1”。

备注 m: 单元号 (m=0、1) n: 通道号 (n=0 ~ 3)

14.3.13 串行输出电平寄存器 m（SOLm）

SOLm 寄存器是设定各通道的数据输出电平反相的寄存器。

只有在 UART 模式中才能设定此寄存器。在 CSI 模式和简易 I²C 模式中，必须将对应的位置“0”。

只在允许串行输出时（SOEmn=1），将此寄存器的各通道 n 反相设定反映到引脚输出。在禁止串行输出时（SOEmn=0），将 SOMn 位的值直接输出而不反相输出。

禁止在运行过程中（SEmn=1）改写 SOLm 寄存器。

通过 16 位存储器操作指令设定 SOLm 寄存器。

能用 SOLmL 并且通过 8 位存储器操作指令设定 SOLm 寄存器的低 8 位。

在产生复位信号后，SOLm 寄存器的值变为“0000H”。

图 14-17 串行输出电平寄存器 m（SOLm）的格式

地址: F0134H、F0135H (SOL0)		复位后: 0000H		R/W												
符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOL0	0	0	0	0	0	0	0	0	0	0	0	0	0	SOL 02	0	SOL 00

地址: F0174H、F0175H (SOL1)		复位后: 0000H		R/W												
符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOL1	0	0	0	0	0	0	0	0	0	0	0	0	0	SOL 12	0	SOL 10

SOLmn	UART 模式中的通道 n 发送数据电平反相的选择														
0	将通信数据直接输出。														
1	将通信数据取反输出。														

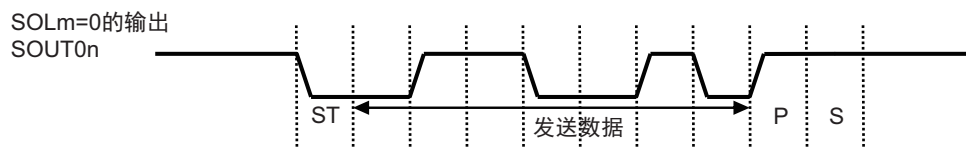
注意 必须将 SOL0 寄存器和 SOL1 寄存器的 bit15 ~ 3 和 bit1 置“0”。

备注 m: 单元号 (m=0、1) n: 通道号 (n=0、2)

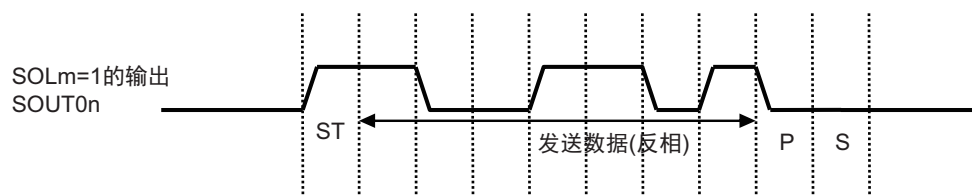
在进行 UART 发送时，发送数据的电平反相例子如图 14-18 所示。

图 14-18 发送数据的电平反相例子

(a)非反相输出(SOLmn=0)



(b)反相输出(SOLmn=1)



备注 m: 单元号 (m=0、1) n: 通道号 (n=0、2)

14.3.14 串行待机控制寄存器 m (SSCm)

SSC0 寄存器是控制在 CSI00 或者 UART0 的串行数据接收时从 STOP 模式启动接收运行 (SNOOZE 模式) 的寄存器。

SSC1 寄存器是控制在 UART2 的串行数据接收时从 STOP 模式下启动接收运行 (SNOOZE 模式) 的寄存器。通过 16 位存储器操作指令设定 SSCm 寄存器。

能用 SSCmL 并且通过 8 位存储器操作指令设定 SSCm 寄存器的低 8 位。

在产生复位信号后, SSCm 寄存器的值变为“0000H”。

注意 SNOOZE 模式中的最大传送速率如下所示:

- CSI00: 1Mbps
- UART0、UART2: 9600bps

图 14-19 串行待机控制寄存器 m (SSCm) 的格式

地址: F0138H (SSC0)、F0178H (SSC1)							复位后: 0000H			R/W						
符号	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSCm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SS ECm	SWC m

SS ECm	选择允许 / 禁止发生 SNOOZE 模式中的通信错误中断
0	允许发生错误中断 (INTSRE0/INTSRE2)。
1	禁止发生错误中断 (INTSRE0/INTSRE2)。
• 在 SNOOZE 模式中的 UART 接收时, 只有在 SWCm=1 并且 EOCmn=1 时, 能将 SSECm 位置“1”或者“0”。其他情况下, 将 SSECm 位置“0”。 • 禁止设定为 SSECm、SWCm=1、0	

SWC m	SNOOZE 模式的设定
0	不使用 SNOOZE 模式功能。
1	使用 SNOOZE 模式功能。
• 在 STOP 模式中, 通过硬件触发信号解除 STOP 模式, 并且在 CPU 不运行的状态下进行 CSI/UART 的接收 (SNOOZE 模式)。 • 只有在选择高速内部振荡器时钟作为 CPU/ 外围硬件时钟 (f_{CLK}) 时才能设定 SNOOZE 模式功能, 而在选择其他时钟的情况下禁止设定。 • 即使使用 SNOOZE 模式, 也必须在通常运行模式中将 SWC 位置“0”并且即将要转移到 STOP 模式前将 SWC 位置“1”。 另外, 必须在从 STOP 模式恢复到通常运行模式后将 SWC 位置“0”。	

图 14-20 在 SNOOZE 模式中进行 UART 接收时的中断

EOCmn 位	SSECm 位	正常接收时	接收错误时
0	0	发生 INTSRx	发生 INTSRx
0	1	发生 INTSRx	发生 INTSRx
1	0	发生 INTSRx	发生 INTSREx
1	1	发生 INTSRx	不发生中断

14.3.15 输入切换控制寄存器（ISC）

在通过 UART0 实现 LIN-bus 通信时，ISC 寄存器的 ISC1 位和 ISC0 位用于协调外部中断和定时器阵列单元的运行。

如果将 bit0 置“1”，就选择串行数据输入（RxD0）引脚的输入信号为外部中断的输入（INTP0），因此能通过 INTP0 中断检测唤醒信号。

如果将 bit1 置“1”，就选择串行数据输入（RxD0）引脚的输入信号为定时器的输入，因此能通过定时器检测唤醒信号并且测量间隔场的低电平宽度和同步场的脉宽。

通过 1 位或者 8 位存储器操作指令设定 ISC 寄存器。

在产生复位信号后，ISC 寄存器的值变为“00H”。

图 14-21 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	定时器阵列单元的通道 7 的输入切换
0	将 TI07 引脚的输入信号用作定时器的输入（通常运行）。
1	将 RxD0 引脚的输入信号用作定时器的输入（检测唤醒信号并且测量间隔场的低电平宽度和同步场的脉宽）。

ISC0	外部中断（INTP0）的输入切换
0	将 INTP0 引脚的输入信号用作外部中断的输入（通常运行）。
1	将 RxD0 引脚的输入信号用作外部中断的输入（检测唤醒信号）。

注意 必须将 bit7 ~ 2 置“0”。

14.3.16 噪声滤波器允许寄存器 0 (NFEN0)

NFEN0 寄存器设定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。

对于用于 CSI 或者简易 I²C 通信的引脚，必须将对应位置“0”，使噪声滤波器无效。

对于用于 UART 通信的引脚，必须将对应位置“1”，使噪声滤波器有效。

在噪声滤波器有效时，通过对象通道的运行时钟 (f_{MCK}) 进行同步后，执行 2 个时钟的一致检测。在噪声滤波器无效时，只通过对象通道的运行时钟 (f_{MCK}) 进行同步。

通过 1 位或者 8 位存储器操作指令设定 NFEN0 寄存器。

在产生复位信号后，NFEN0 寄存器的值变为“00H”。

图 14-22 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	SNFEN30	0	SNFEN20	0	SNFEN10	0	SNFEN00

SNFEN30	RxD3 引脚的噪声滤波器的使用与否
0	噪声滤波器 OFF
1	噪声滤波器 ON
当用作 RxD3 引脚时，必须将 SNFEN30 位置“1”。	
当用作 RxD3 引脚以外的功能时，必须将 SNFEN30 位置“0”。	

SNFEN20	RxD2 引脚的噪声滤波器的使用与否
0	噪声滤波器 OFF
1	噪声滤波器 ON
当用作 RxD2 引脚时，必须将 SNFEN20 位置“1”。	
当用作 RxD2 引脚以外的功能时，必须将 SNFEN20 位置“0”。	

SNFEN10	RxD1 引脚的噪声滤波器的使用与否
0	噪声滤波器 OFF
1	噪声滤波器 ON
当用作 RxD1 引脚时，必须将 SNFEN10 位置“1”。	
当用作 RxD1 引脚以外的功能时，必须将 SNFEN10 位置“0”。	

SNFEN00	RxD0 引脚的噪声滤波器的使用与否
0	噪声滤波器 OFF
1	噪声滤波器 ON
当用作 RxD0 引脚时，必须将 SNFEN00 位置“1”。	
当用作 RxD0 引脚以外的功能时，必须将 SNFEN00 位置“0”。	

注意 必须将 bit7、5、3、1 置“0”。

14.3.17 控制串行输入 / 输出引脚的端口功能的寄存器

在使用串行阵列单元时，必须设定与对象通道复用的端口相关寄存器（端口模式寄存器（PMxx）、端口寄存器（Pxx）、端口输入模式寄存器（PIMxx）、端口输出模式寄存器（POMxx）和端口模式控制寄存器（PMCxx））。

详细内容请参照“4.3.1 端口模式寄存器（PMxx）”、“4.3.2 端口寄存器（Pxx）”、“4.3.4 端口输入模式寄存器（PIMxx）”、“4.3.5 端口输出模式寄存器（POMxx）”、“4.3.6 端口模式控制寄存器（PMCxx）”。

在将串行数据输出或者串行时钟输出引脚的复用端口（P07/TxD1/SEG50 等）用作串行数据输出或者串行时钟输出时，必须将与各端口对应的端口模式控制寄存器（PMCxx）的位和端口模式寄存器（PMxx）的位置“0”，将端口寄存器（Pxx）的位置“1”。

另外，在用作 N 沟道漏极开路输出（V_{DD} 耐压）模式时，必须将与各端口对应的端口输出模式寄存器（POMxx）的位置“1”。在连接以不同电位（1.8V、2.5V、3V）运行的外部器件时，请参照“4.4.4 不同电位（1.8V、2.5V、3V）的输入 / 输出缓冲器对应”。

例）P07/TxD1/SEG50 用作串行数据输出的情况

将端口模式控制寄存器 0 的 PM07 位置“0”。

将端口寄存器 0 的 P07 位置“1”。

在将串行数据输入引脚或者串行时钟输入引脚的复用端口（P06/RxD1/SDA10/SEG49 等）用作串行数据输入或者串行时钟输入时，必须将各端口的端口模式寄存器（PMxx）的对应位置“1”，并且将端口模式控制寄存器（PMCxx）的对应位置“0”。此时，端口寄存器（Pxx）的对应位可以是“0”或者“1”。

另外，在用作 TTL 输入缓冲器时，必须将与各端口对应的端口输入模式寄存器（PIMxx）的位置“1”。在连接以不同电位（1.8V、2.5V、3V）运行的外部器件时，请参照“4.4.4 不同电位（1.8V、2.5V、3V）的输入 / 输出缓冲器对应”。

例）P06/RxD1/SDA10/SEG49 用作串行数据输入的情况

将端口模式寄存器 0 的 PM06 位置“1”。

将端口寄存器 0 的 P06 位置“0”或者“1”。

14.4 运行停止模式

串行阵列单元的各串行接口有运行停止模式。

在运行停止模式中不能进行串行通信，因此能降低功耗。

另外，在运行停止模式中能将用于串行接口的引脚用作端口功能。

14.4.1 以单元为单位停止运行的情况

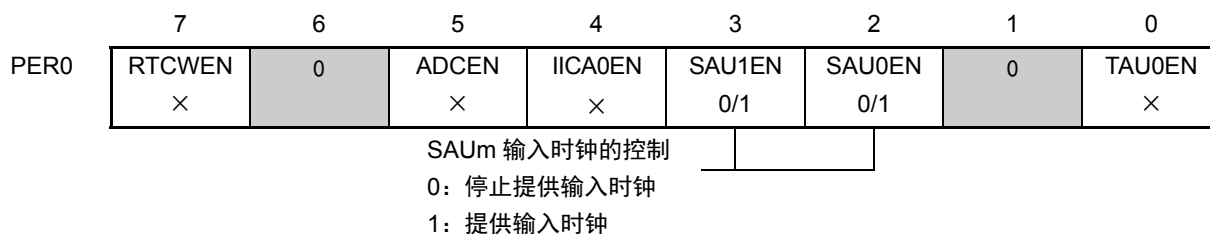
通过外围允许寄存器 0（PER0）设定以单元为单位的停止运行。

PER0 寄存器是设定允许或者禁止给各外围硬件提供时钟的寄存器。通过给不使用的硬件停止提供时钟，降低功耗和噪声。

要停止串行阵列单元 0 时，必须将 bit2（SAU0EN）置“0”；要停止串行阵列单元 1 时，必须将 bit3（SAU1EN）置“0”。

图 14-23 以单元为单位停止运行时的外围允许寄存器 0（PER0）的设定

(a) 外围允许寄存器 0（PER0）..... 只将要停止 SAUm 的对应位置“0”。



注意 1. 当 SAUmEN 位为“0”时，忽视串行阵列单元 m 的控制寄存器的写操作，而且读取值都为初始值。

但是，以下的寄存器除外：

- 输入切换控制寄存器（ISC）
- 噪声滤波器允许寄存器 0（NFEN0）
- 端口输入模式寄存器 0、1、3（PIM0、PIM1、PIM3）
- 端口输出模式寄存器 0、1、3（POM0、POM1、POM3）
- 端口模式寄存器 0、1、3（PM0、PM1、PM3）
- 端口寄存器 0、1、3（P0、P1、P3）

2. 必须将 bit1 和 bit6 置“0”。

备注 ×：这是串行阵列单元未使用的位（取决于其他外围功能的设定）。

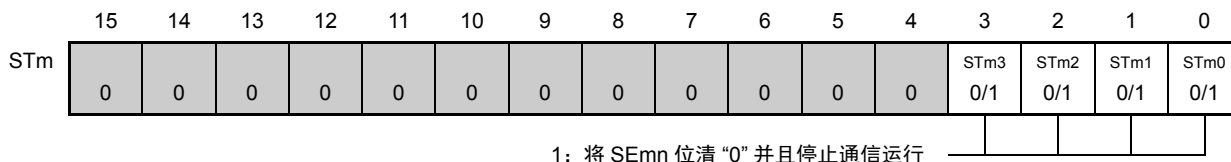
0/1：根据用户的用途置“0”或者“1”。

14.4.2 按通道停止运行的情况

通过以下各寄存器设定按通道停止运行。

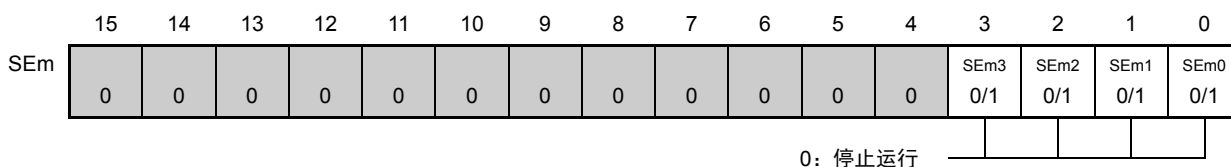
图 14-24 按通道停止运行时的各寄存器的设定

(a) 串行通道停止寄存器 m (STm) 这是设定允许各通道的通信 / 停止计数的寄存器。



※ 因为 STmn 位是触发位，所以如果 SEmn 位为“0”就立即清除 STmn 位。

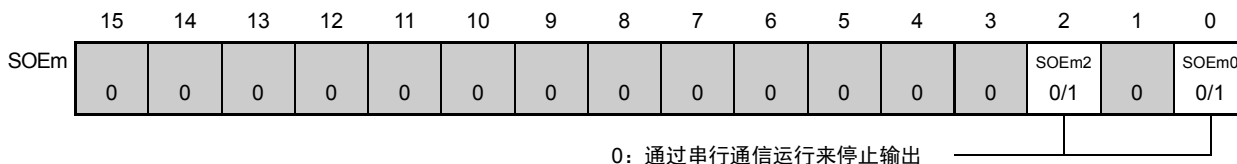
(b) 串行通道允许状态寄存器 m (SEm) 此寄存器表示各通道的数据发送和接收的运行或者停止状态。



※SEm 寄存器是只读状态寄存器，通过 STm 寄存器来停止运行。

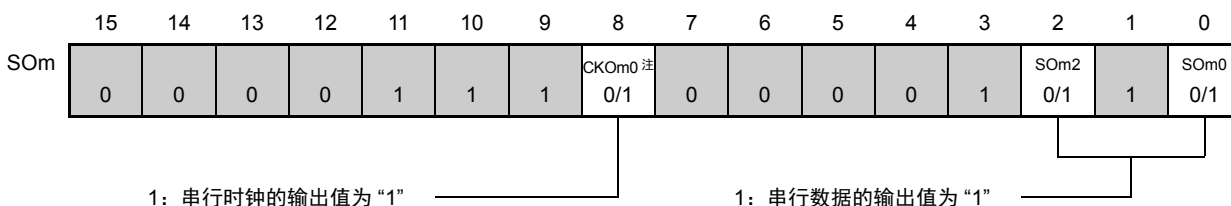
对于已经停止运行的通道，能通过软件设定 SOm 寄存器的 CKOmn 位的值。

(c) 串行输出允许寄存器 m (SOEm) 这是设定允许或者停止各通道串行通信输出的寄存器。



※ 对于已经停止串行输出的通道，能通过软件设定 SOm 寄存器的 SOmn 位的值。

(d) 串行输出寄存器 m (SOM) 这是各通道串行输出的缓冲寄存器。



※ 当将各通道对应的引脚用作端口功能时，必须将相应的 CKOmn 位和 SOMn 位置“1”。

注 只限串行阵列单元 0。

备注 1. m: 单元号 (m=0、1) n: 通道号 (n=0 ~ 3)

2. : 不能设定 (设定初始值)。 0/1: 根据用户的用途置“0”或者“1”。

14.5 3 线串行 I/O（CSI00）通信的运行

这是通过串行时钟（SCK）和串行数据（SI 和 SO）共 3 条线实现的时钟同步通信功能。

[数据的发送和接收]

- 7 位或者 8 位的数据长度
- 发送和接收数据的相位控制
- MSB/LSB 优先的选择

[时钟控制]

- 主控或者从属的选择
- 输入/输出时钟的相位控制
- 设定由预分频器和通道内部计数器产生的传送周期。
- 最大传送速率注

主控通信: $\text{Max. } f_{\text{CLK}}/2$

从属通信: $\text{Max. } f_{\text{MCK}}/6$

[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

CSI00 支持 SNOOZE 模式。SNOOZE 模式功能是指：如果在 STOP 模式的状态下检测到 SCK 的输入，就不需要 CPU 运行而接收数据的。

注 必须在满足 SCK 周期时间（ t_{KCY} ）特性的范围内使用。详细内容请参照“第 32 章 电特性”。

支持 3 线串行 I/O（CSI00）的通道为 SAU0 的通道 0。

单元	通道	用作 CSI	用作 UART	用作简易 I ² C
0	0	CSI00	UART0	IIC00
	1	—	(支持 LIN-bus)	—
	2	—	UART1	IIC10
	3	—		—
1	0	—	UART2	—
	1	—		—
	2	—	UART3	—
	3	—		—

3 线串行 I/O（CSI00）有以下 7 种通信运行：

- 主控发送 (参照 14.5.1)
- 主控接收 (参照 14.5.2)
- 主控发送和接收 (参照 14.5.3)
- 从属发送 (参照 14.5.4)
- 从属接收 (参照 14.5.5)
- 从属发送和接收 (参照 14.5.6)
- SNOOZE 模式功能 (参照 14.5.7)

14.5.1 主控发送

主控发送是指此 RL78 微控制器输出传送时钟并且将数据发送到其他设备的运行。

3 线串行 I/O	CSI00
对象通道	SAU0 的通道 0
使用的引脚	SCK00、SO00
中断	INTCSI00 可选择传送结束中断（单次传送模式）或者缓冲器空中断（连续传送模式）。
错误检测标志	无
传送数据长度	7 位或者 8 位
传送速率注	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 位进行选择。 • DAPmn=0: 在串行时钟开始运行时, 开始数据输出。 • DAPmn=1: 在串行时钟开始运行的半个时钟前, 开始数据输出。
时钟相位	可通过 SCRmn 寄存器的 CKPmn 位进行选择。 • CKPmn = 0: 不反相（在 SCK 的下降沿输出数据, 在上升沿输入数据） • CKPmn = 1: 反相（在 SCK 的上升沿输出数据, 在下降沿输入数据）
数据方向	MSB 优先或者 LSB 优先

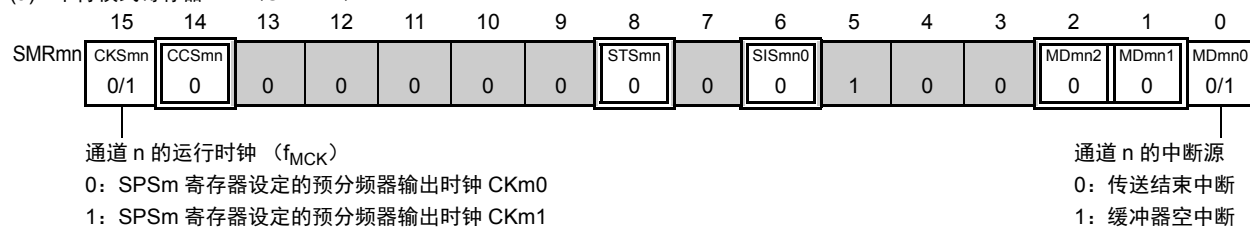
注 必须在满足此条件并且满足电特性的外围功能特性（参照“第 32 章 电特性”）的范围内使用。

备注 m: 单元号 (m=0) n: 通道号 (n=0) mn=00

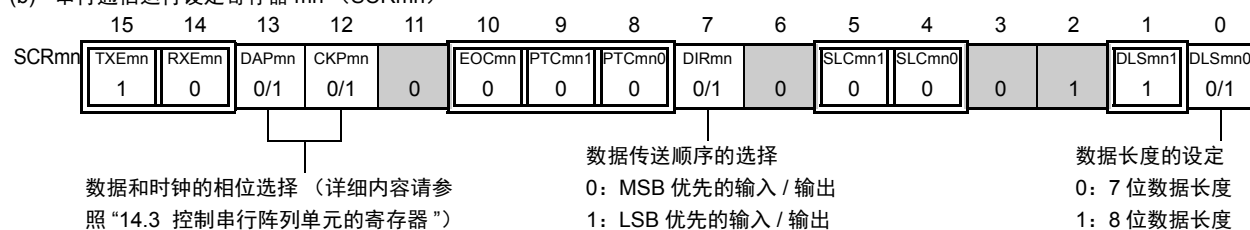
(1) 寄存器的设定

图 14-25 3 线串行 I/O (CSI00) 主控发送时的寄存器设定内容例子

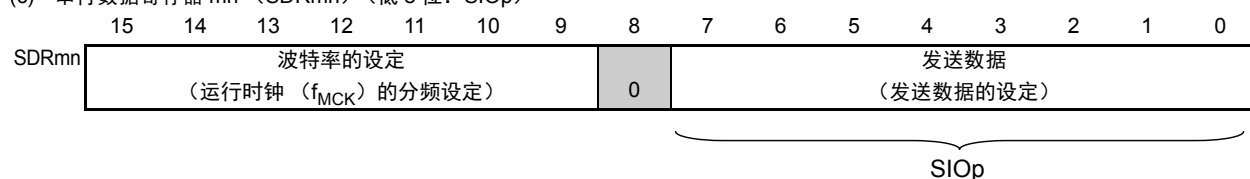
(a) 串行模式寄存器 mn (SMRmn)



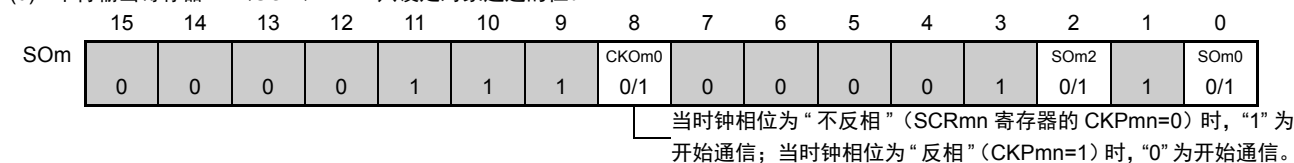
(b) 串行通信运行设定寄存器 mn (SCRmn)



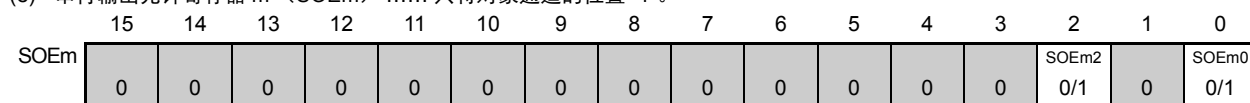
(c) 串行数据寄存器 mn (SDRmn) (低 8 位: SIOp)



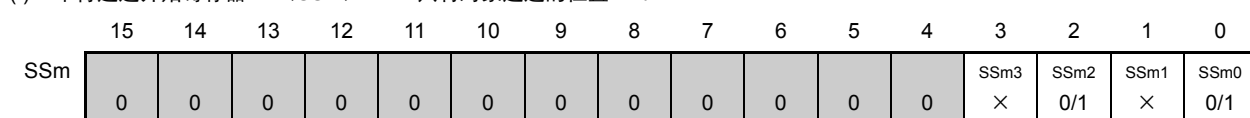
(d) 串行输出寄存器 m (SOm) 只设定对象通道的位。



(e) 串行输出允许寄存器 m (SOEm) 只将对象通道的位置“1”。



(f) 串行通道开始寄存器 m (SSm) 只将对象通道的位置“1”。



备注 1. m: 单元号 (m=0) n: 通道号 (n=0) p: CSI 号 (p=00) mn=00

2. : 在 CSI 主控发送模式中为固定设定。 : 不能设定 (设定初始值)。

×: 这是在此模式中不能使用的位 (在其他模式中也不使用的情况下, 设定初始值)。

0/1: 根据用户的用途置“0”或者“1”。

(2) 操作步骤

图 14-26 主控发送的初始设定步骤

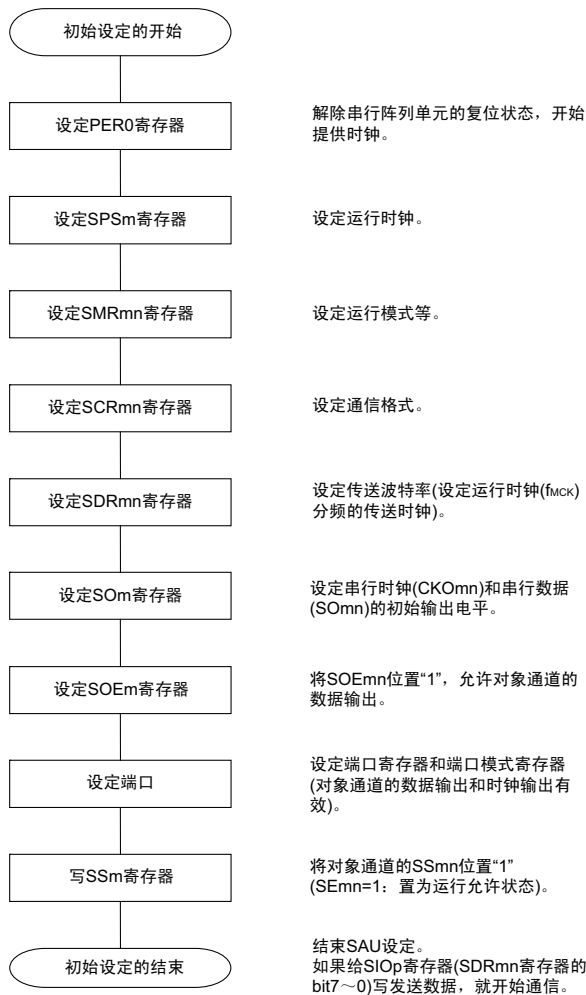


图 14-27 主控发送的中止步骤

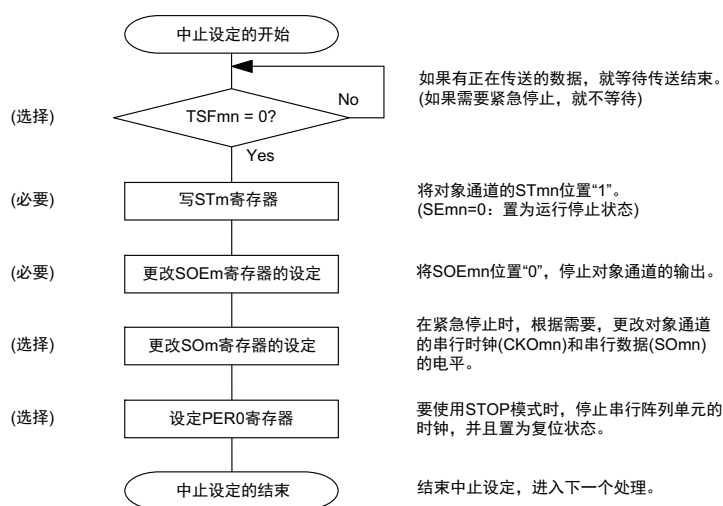


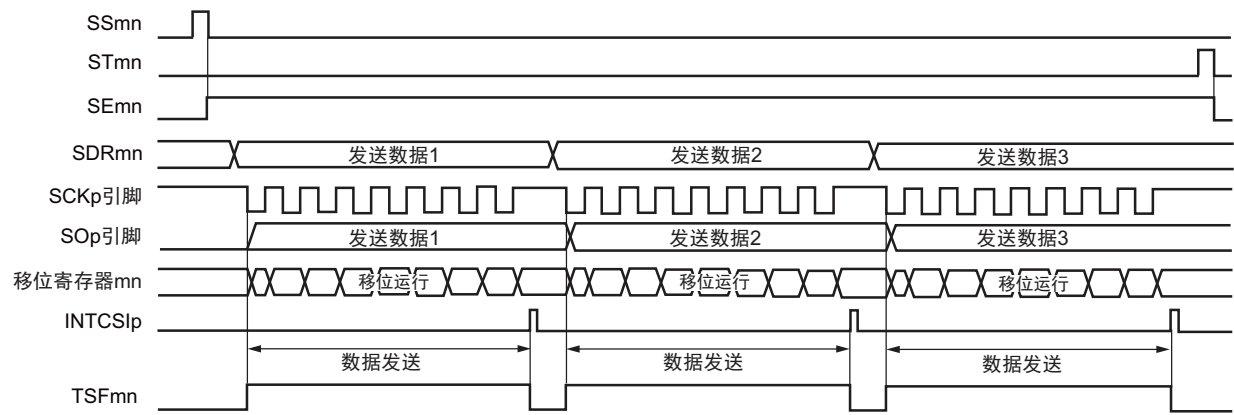
图 14-28 重新开始主控发送的设定步骤



备注 如果在中止设定中改写 PER0 来停止提供时钟，就必须在等待通信对象（从属设备）的停止或者通信结束后进行初始设定而不是进行重新开始设定。

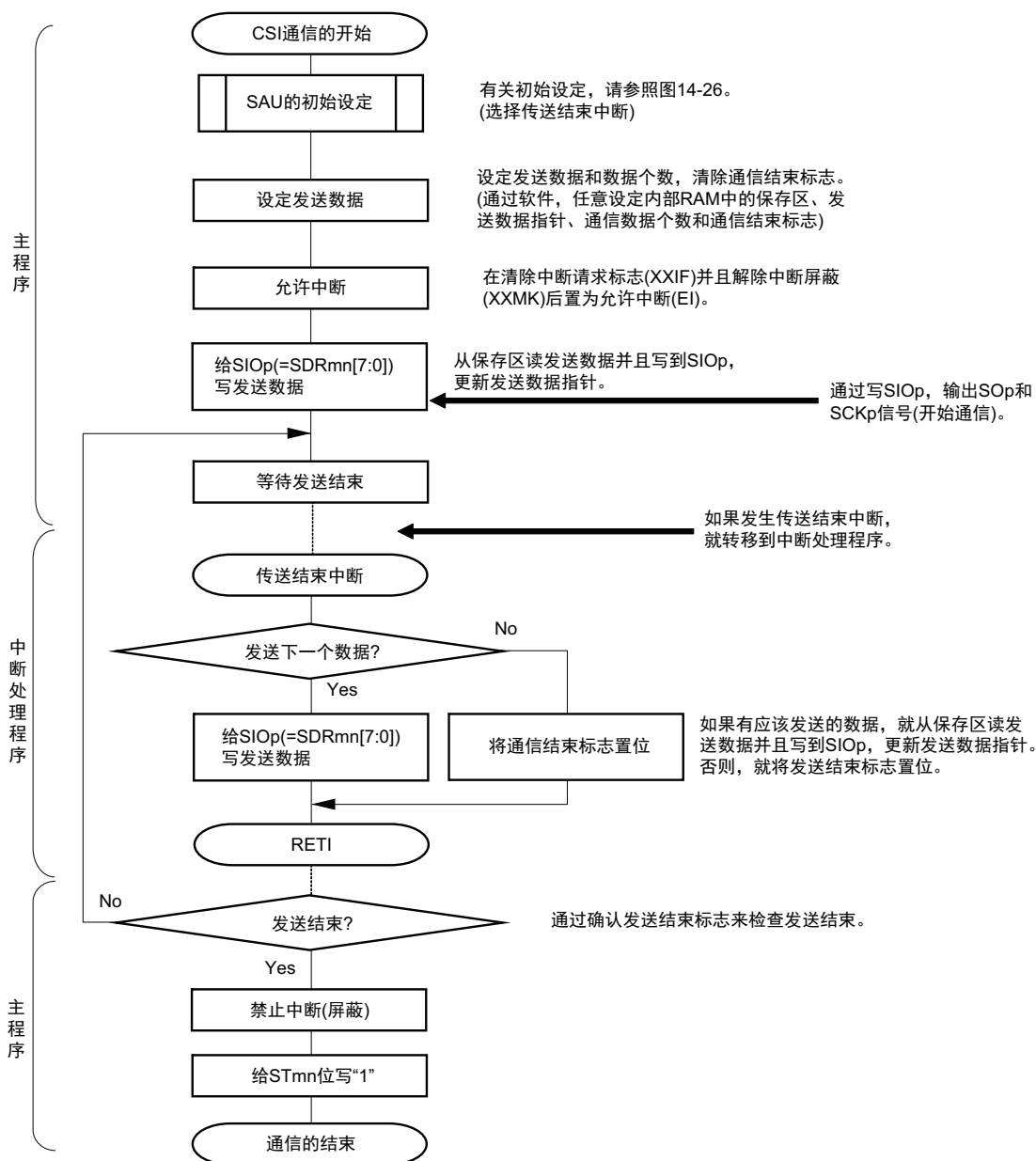
(3) 处理流程（单次发送模式）

图 14-29 主控发送（单次发送模式）的时序图（类型 1：DAPmn=0、CKPmn=0）



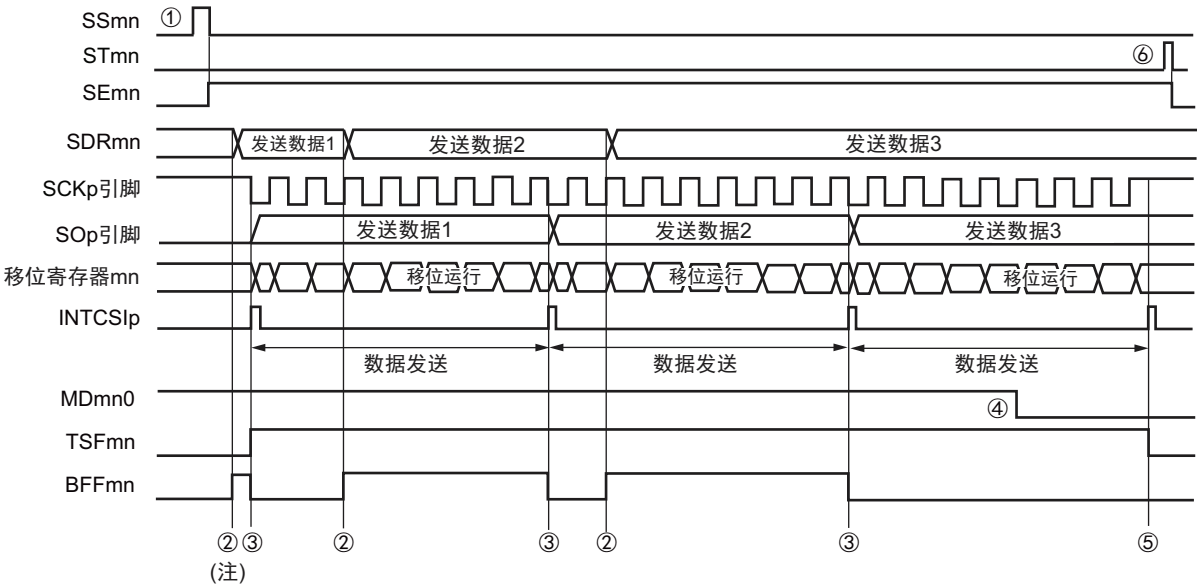
备注 m: 单元号 (m=0) n: 通道号 (n=0) p: CSI 号 (p=00) mn=00

图 14-30 主控发送（单次发送模式）的流程图



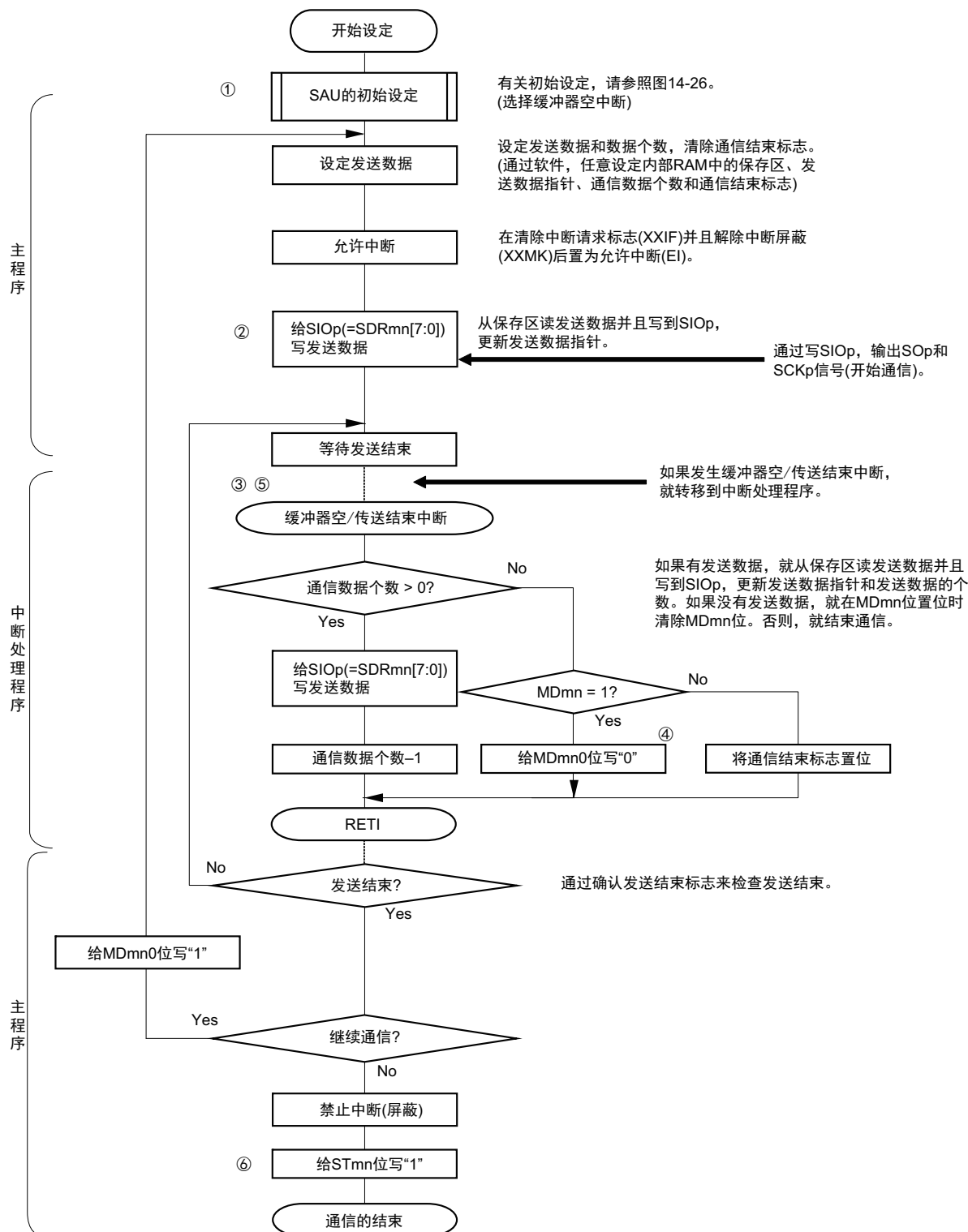
(4) 处理流程（连续发送模式）

图 14-31 主控发送（连续发送模式）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”期间（有效数据保存在串行数据寄存器 mn（SDRmn）时）给 SDRmn 寄存器写发送数据，就重写发送数据。
- 注意 即使在运行中也能改写串行模式寄存器 mn（SMRmn）的 MDmn0 位。但是，为了能赶上最后发送数据的传送结束中断，必须在开始传送最后一位之前进行改写。
- 备注 m: 单元号（m=0） n: 通道号（n=0） p: CSI 号（p=00） mn=00

图 14-32 主控发送（连续发送模式）的流程图



备注 图中的①~⑥对应“图 14-31 主控发送（连续发送模式）的时序图”中的①~⑥。

14.5.2 主控接收

主控接收是 RL78 微控制器输出传送时钟并且从其他设备接收数据的运行。

3 线串行 I/O	CSI00
对象通道	SAU0 的通道 0
使用的引脚	SCK00、SI00
中断	INTCSI00 可选择传送结束中断（单次传送模式）或者缓冲器空中断（连续传送模式）。
错误检测标志	只有溢出错误检测标志（OVFmn）。
传送数据长度	7 位或者 8 位
传送速率注	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 位进行选择。 • DAPmn=0: 在串行时钟开始运行时，开始数据输出。 • DAPmn=1: 在串行时钟开始运行的半个时钟前，开始数据输出。
时钟相位	可通过 SCRmn 寄存器的 CKPmn 位进行选择。 • CKPmn=0: 不反相 • CKPmn=1: 反相
数据方向	MSB 优先或者 LSB 优先

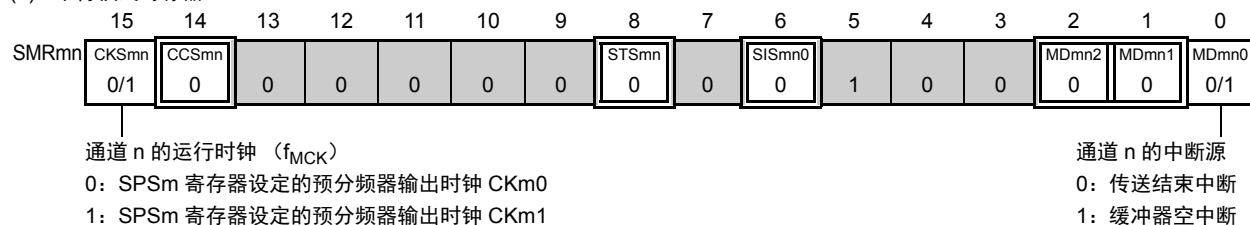
注 必须在满足此条件并且满足电特性的外围功能特性（参照“第 32 章 电特性”）的范围内使用。

备注 m: 单元号 (m=0) n: 通道号 (n=0) mn=00

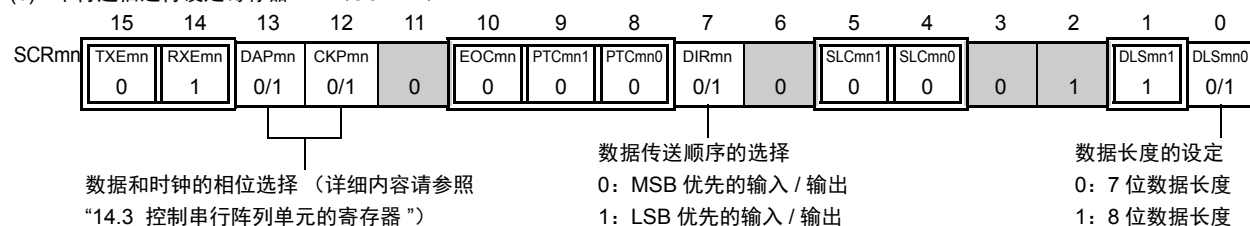
(1) 寄存器的设定

图 14-33 3 线串行 I/O (CSI00) 主控接收时的寄存器设定内容例子

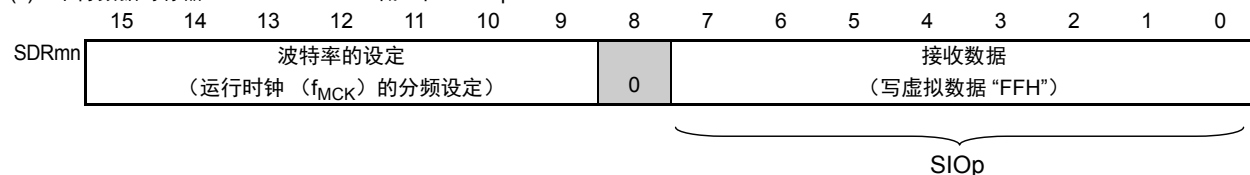
(a) 串行模式寄存器 mn (SMRmn)



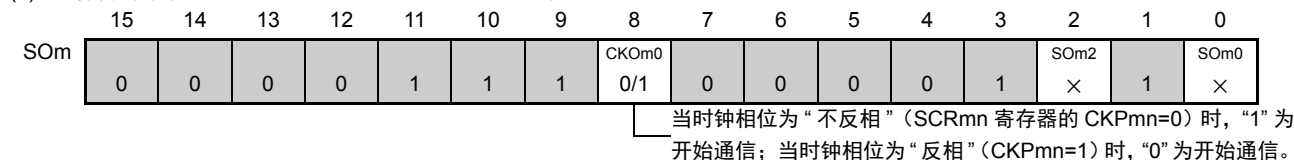
(b) 串行通信运行设定寄存器 mn (SCRmn)



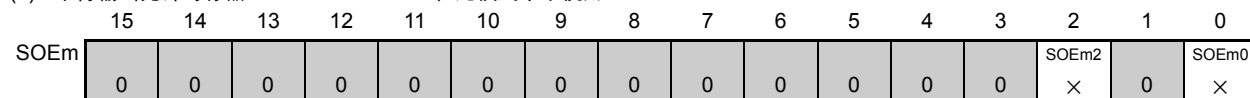
(c) 串行数据寄存器 mn (SDRmn) (低 8 位: SIOp)



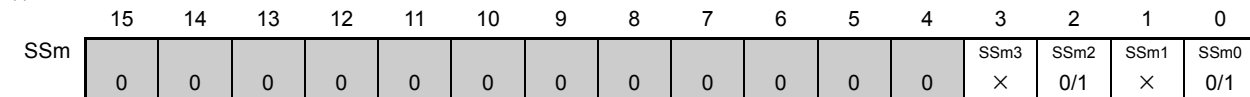
(d) 串行输出寄存器 m (SOM) 只设定对象通道的位。



(e) 串行输出允许寄存器 m (SOEm) 在此模式中不使用。



(f) 串行通道开始寄存器 m (SSm) 只将对象通道的位置“1”。



备注 1. m: 单元号 (m=0) n: 通道号 (n=0) p: CSI 号 (p=00) mn=00

2. □: 在 CSI 主控接收模式中为固定设定。■: 不能设定 (设定初始值)。

×: 这是在此模式中不能使用的位 (在其他模式中也不使用的情况下, 设定初始值)。

0/1: 根据用户的用途置“0”或者“1”。

(2) 操作步骤

图 14-34 主控接收的初始设定步骤

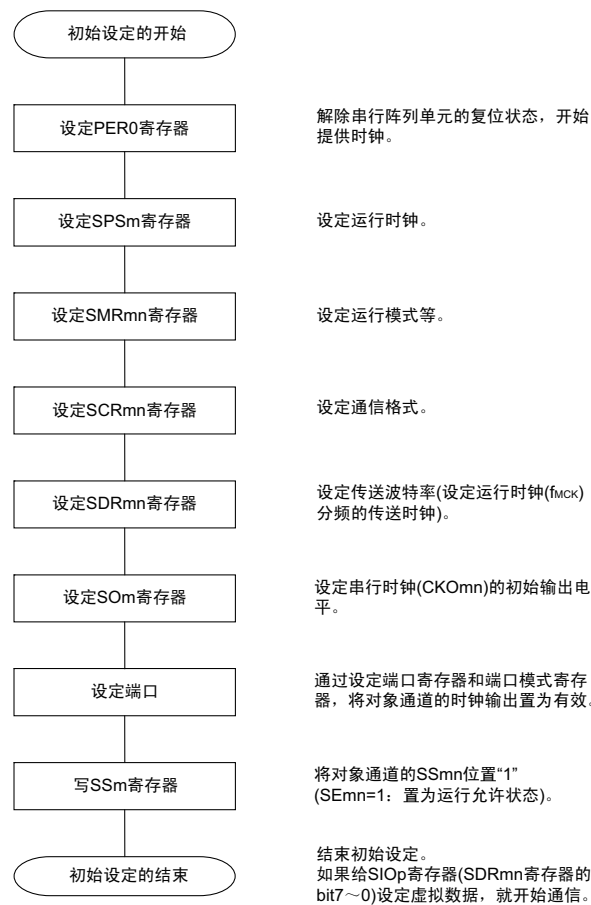


图 14-35 主控接收的中止步骤

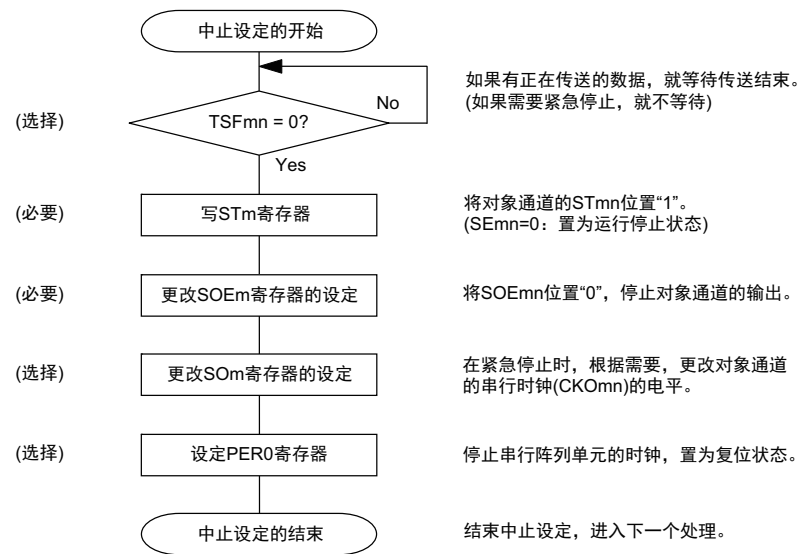
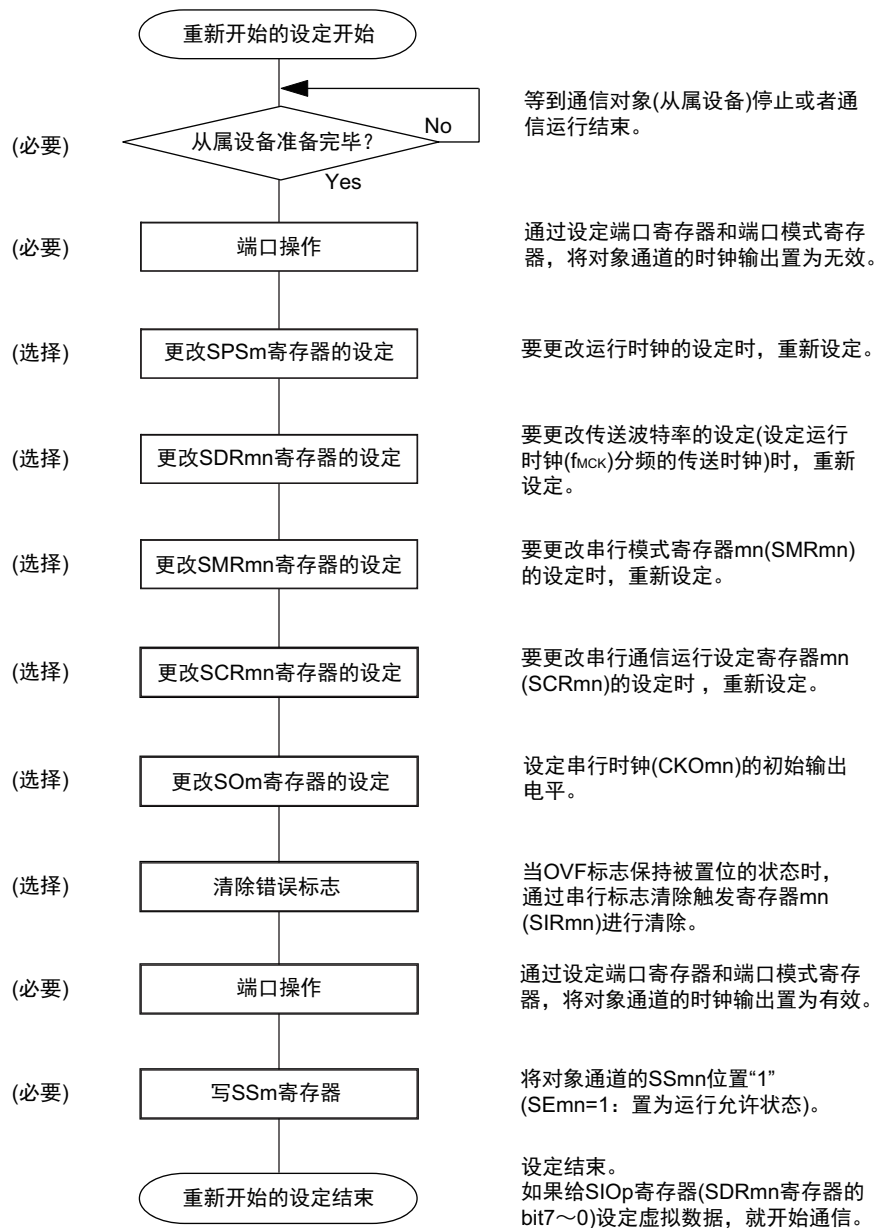


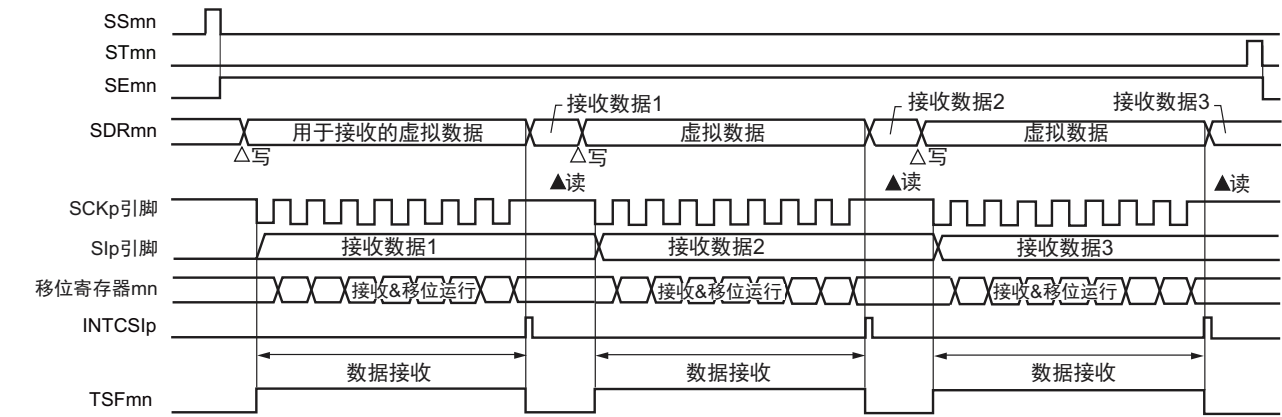
图 14-36 重新开始主控接收的设定步骤



备注 如果在中止设定中改写 PER0 来停止提供时钟，就必须在等待通信对象（从属设备）的停止或者通信结束后进行初始设定而不是进行重新开始设定。

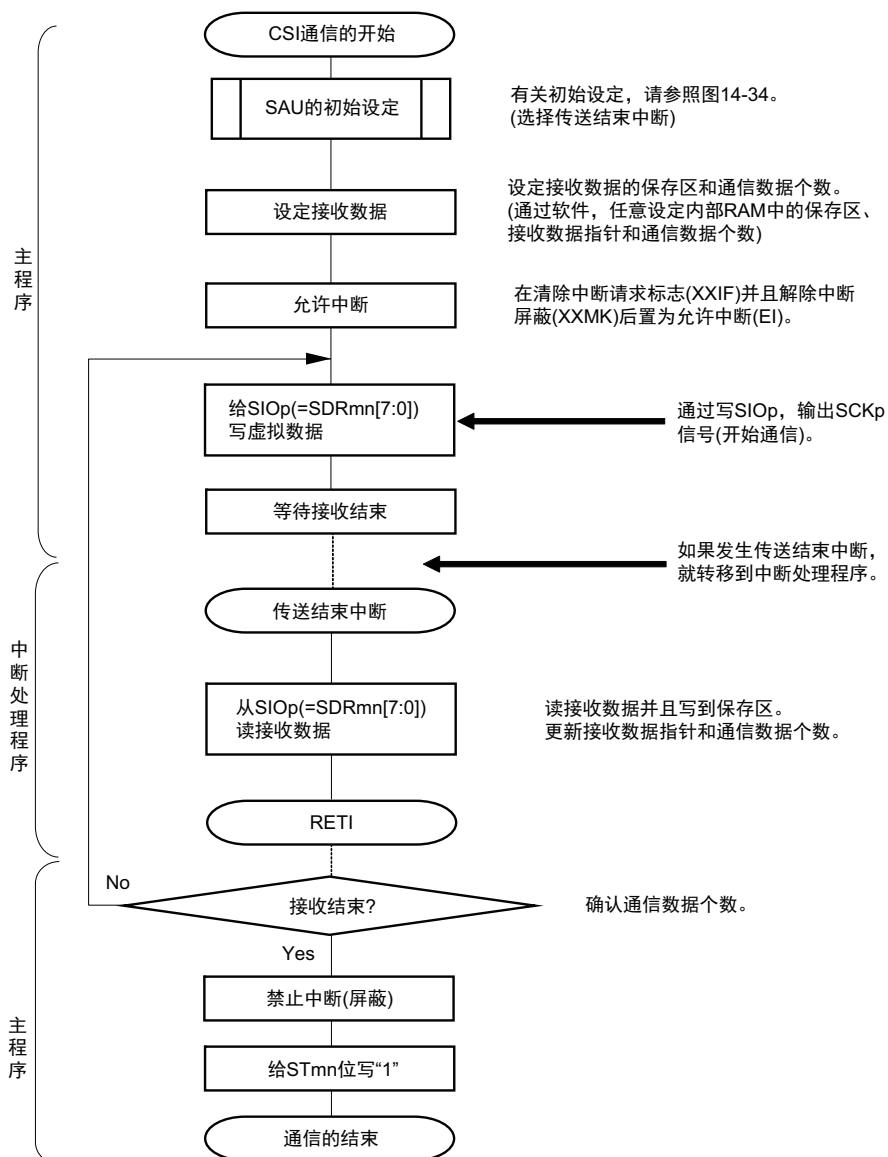
(3) 处理流程（单次接收模式）

图 14-37 主控接收（单次接收模式）的时序图（类型 1：DAPmn=0、CKPmn=0）



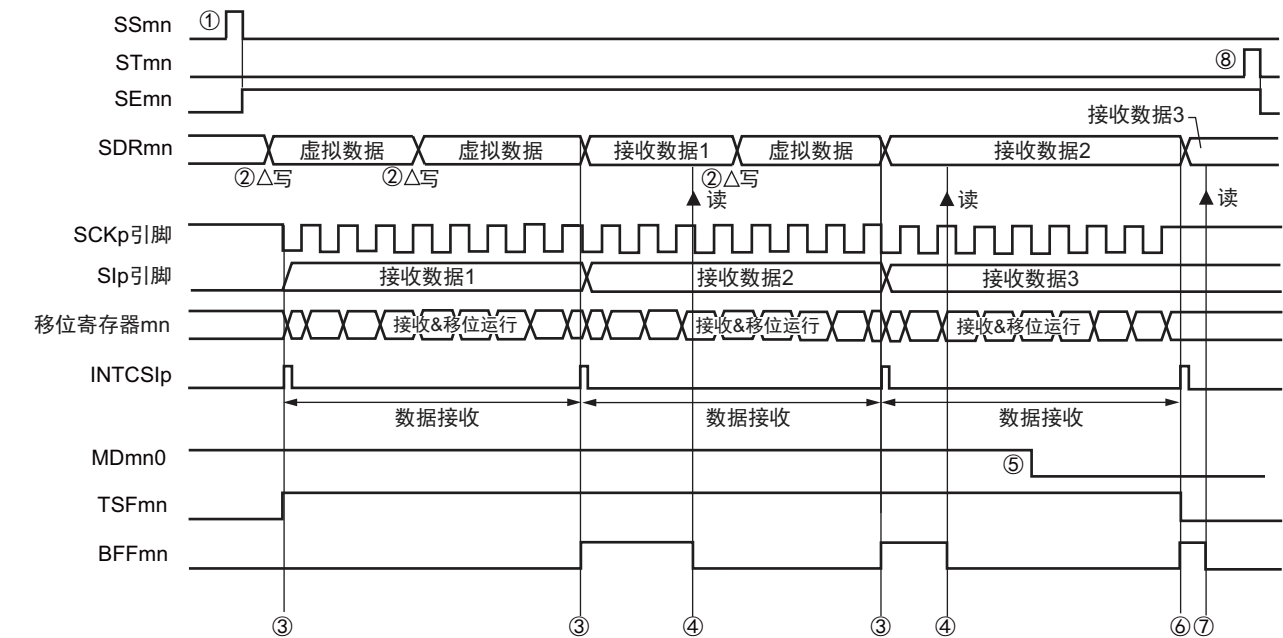
备注 m: 单元号 (m=0) n: 通道号 (n=0) p: CSI 号 (p=00) mn=00

图 14-38 主控接收（单次接收模式）的流程图



(4) 处理流程（连续接收模式）

图 14-39 主控接收（连续接收模式）的时序图（类型 1：DAPmn=0、CKPmn=0）

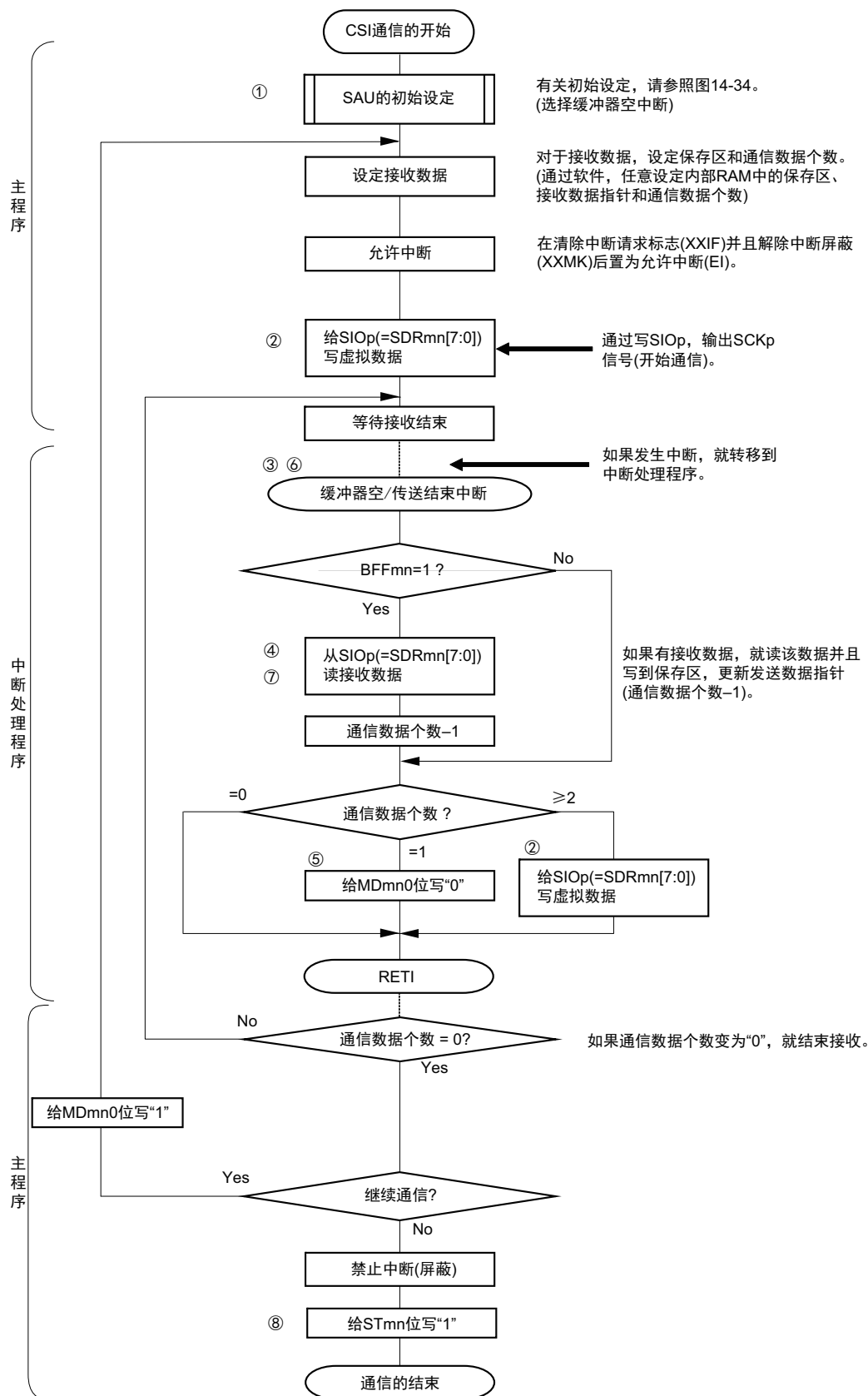


注意 即使在运行过程中也能改写 MDmn0 位。但是，为了能赶上最后接收数据的传送结束中断，必须在开始接收最后一位之前进行改写。

备注 1. 图中的①～⑧对应“图 14-40 主控接收（连续接收模式）的流程图”中的①～⑧。

2. m: 单元号 (m=0) n: 通道号 (n=0) p: CSI 号 (p=00) mn=00

图 14-40 主控接收（连续接收模式）的流程图



备注 图中的①~⑧对应“图 14-39 主控接收（连续接收模式）的时序图”中的①~⑧。

14.5.3 主控的发送和接收

主控的发送和接收是 RL78 微控制器输出传送时钟并且和其他设备进行数据发送和接收的运行。

3 线串行 I/O	CSI00
对象通道	SAU0 的通道 0
使用的引脚	SCK00、SI00、SO00
中断	INTCSI00 可选择传送结束中断（单次传送模式）或者缓冲器空中断（连续传送模式）。
错误检测标志	只有溢出错误检测标志（OVFmn）。
传送数据长度	7 位或者 8 位
传送速率 ^注	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 位进行选择。 • DAPmn=0: 在串行时钟开始运行时，开始数据输出。 • DAPmn=1: 在串行时钟开始运行的半个时钟前，开始数据输出。
时钟相位	可通过 SCRmn 寄存器的 CKPmn 位进行选择。 • CKPmn=0: 不反相 • CKPmn=1: 反相
数据方向	MSB 优先或 LSB 优先

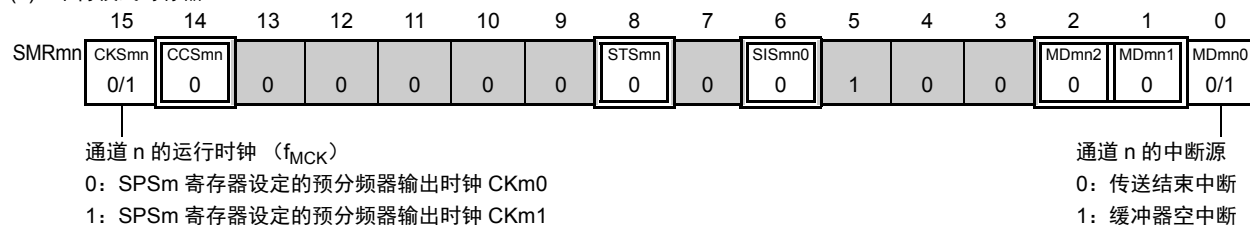
注 必须在满足此条件并且满足电特性的外围功能特性（参照“第 32 章 电特性”）的范围内使用。

备注 m: 单元号 (m=0) n: 通道号 (n=0) p: CSI 号 (p=00) mn=00

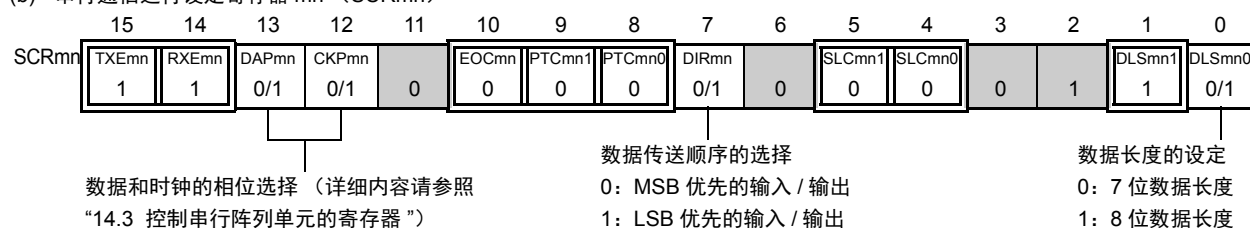
(1) 寄存器的设定

图 14-41 3 线串行 I/O (CSI00) 主控发送和接收时的寄存器设定内容例子

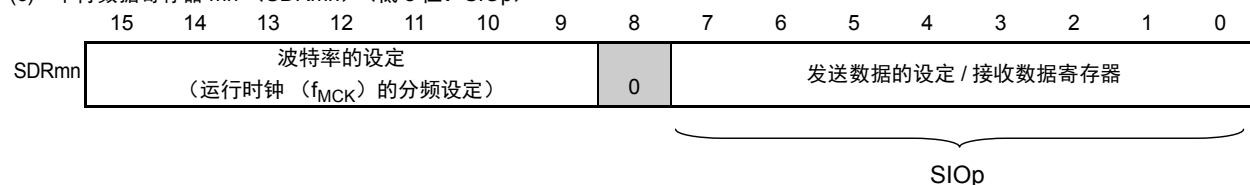
(a) 串行模式寄存器 mn (SMRmn)



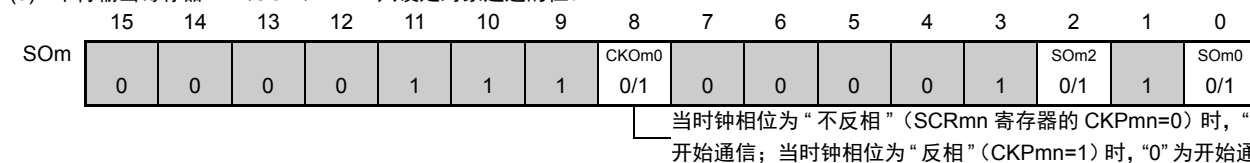
(b) 串行通信运行设定寄存器 mn (SCRmn)



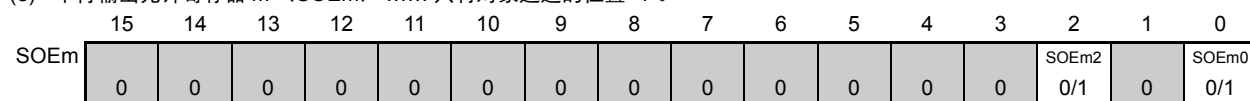
(c) 串行数据寄存器 mn (SDRmn) (低 8 位: SIOp)



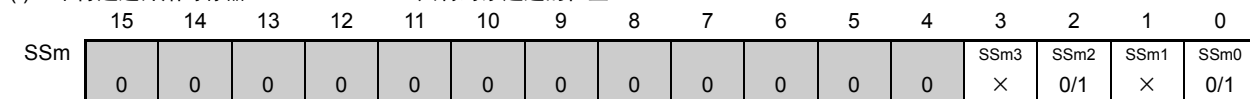
(d) 串行输出寄存器 m (SOM) 只设定对象通道的位。



(e) 串行输出允许寄存器 m (SOEm) 只将对象通道的位置“1”。



(f) 串行通道开始寄存器 m (SSm) 只将对象通道的位置“1”。



备注 1. m: 单元号 (m=0) n: 通道号 (n=0) p: CSI 号 (p=00) mn=00

2. : 在 CSI 主控发送和接收模式中为固定设定。 : 不能设定 (设定初始值)。

×: 这是在此模式中不能使用的位 (在其他模式中也不使用的情况下, 设定初始值)。

0/1: 跟据用户的用途置“0”或者“1”。

(2) 操作步骤

图 14-42 主控发送和接收的初始设定步骤

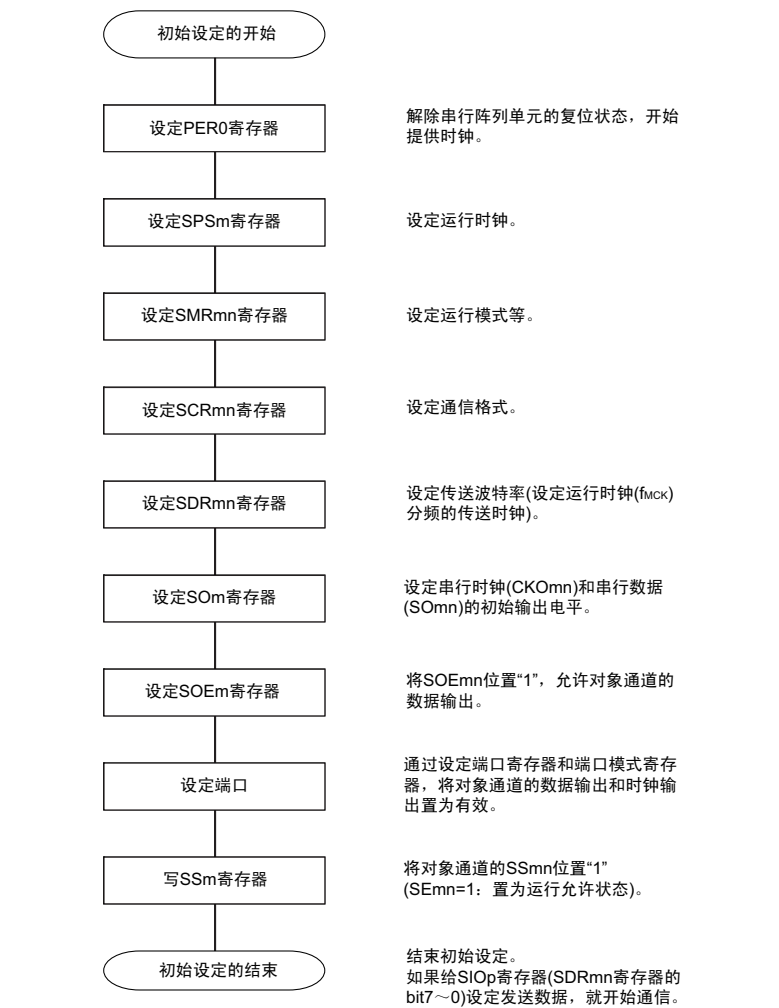


图 14-43 主控发送和接收的中止步骤

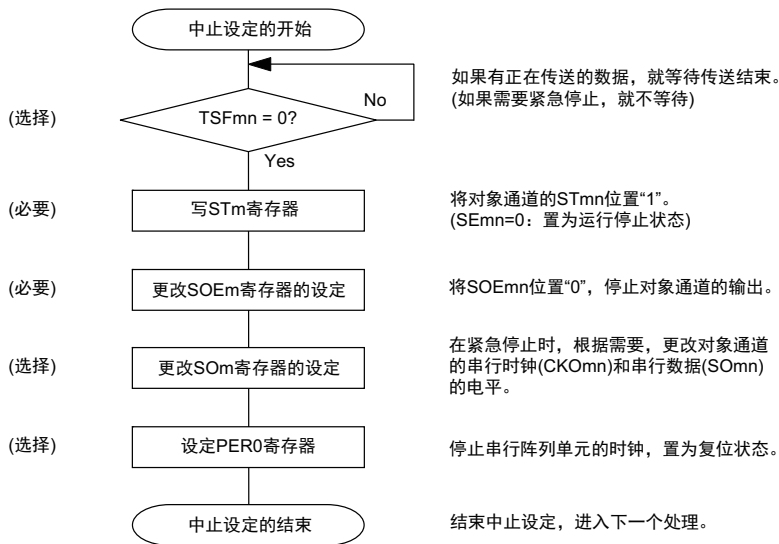
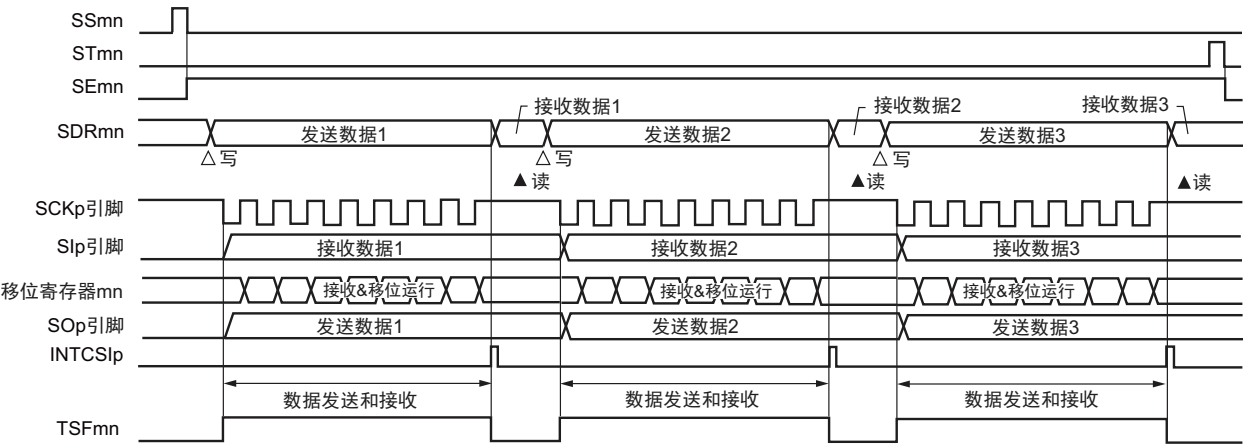


图 14-44 重新开始主控发送和接收的设定步骤



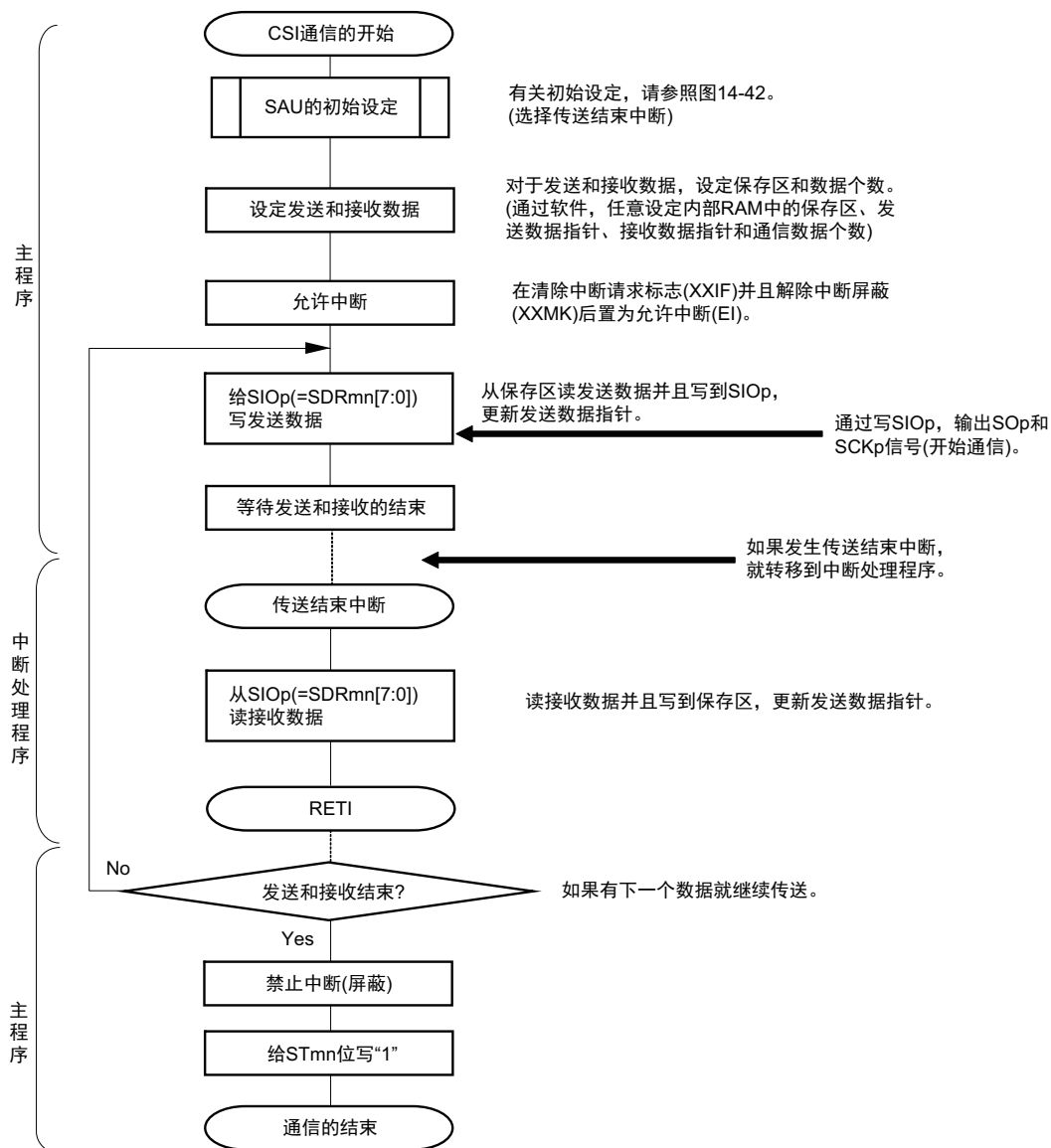
(3) 处理流程（单次发送和接收模式）

图 14-45 主控发送和接收（单次发送和接收模式）的时序图（类型 1：DAPmn=0、CKPmn=0）



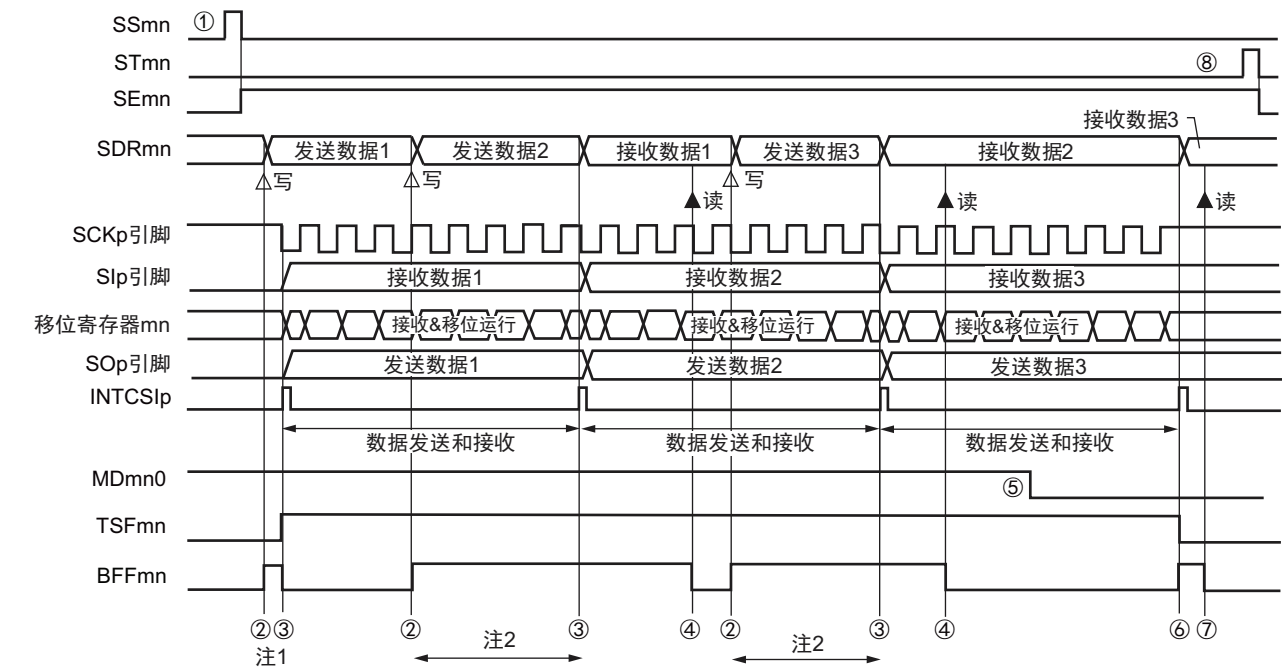
备注 m: 单元号 (m=0) n: 通道号 (n=0) p: CSI 号 (p=00) mn=00

图 14-46 主控发送和接收（单次发送和接收模式）的流程图



(4) 处理流程（连续发送和接收模式）

图 14-47 主控发送和接收（连续发送和接收模式）的时序图（类型 1：DAPmn=0、CKPmn=0）



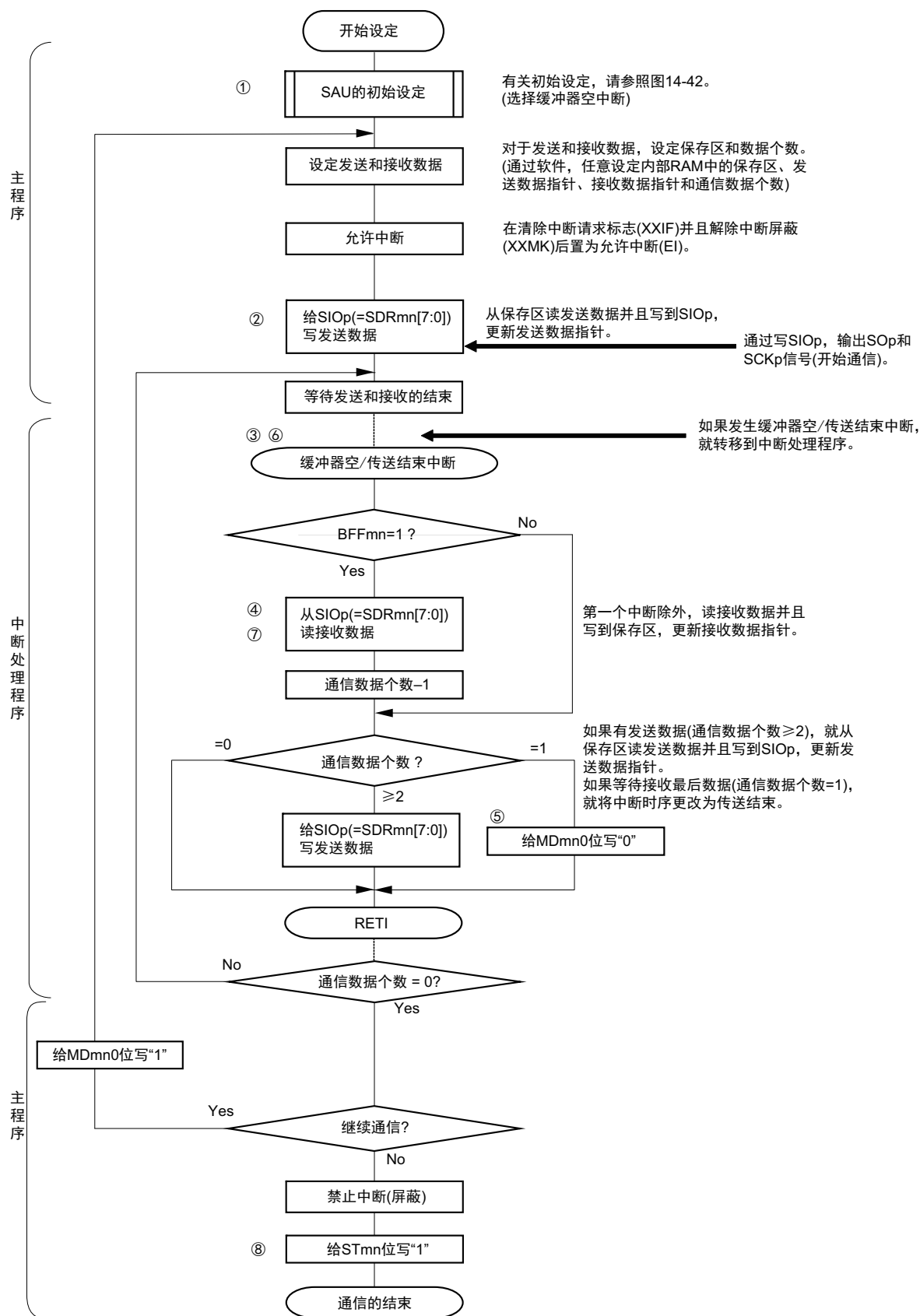
- 注 1. 如果在串行状态寄存器 mn（SSRmn）的 BFFmn 位为“1”期间（有效数据保存在串行数据寄存器 mn（SDRmn）时）给 SDRmn 寄存器写发送数据，就重写发送数据。
2. 如果在此期间读取 SDRmn 寄存器，就能读发送数据。此时，不影响传送运行。

注意 即使在运行中也能改写串行模式寄存器 mn（SMRmn）的 MDmn0 位。但是，为了能赶上最后发送数据的传送结束中断，必须在开始传送最后一位之前进行改写。

备注 1. 图中的①～⑧对应“图 14-48 主控发送和接收（连续发送和接收模式）的流程图”中的①～⑧。

2. m: 单元号（m=0） n: 通道号（n=0） p: CSI 号（p=00） mn=00

图 14-48 主控发送和接收（连续发送和接收模式）的流程图



备注 图中的①~⑧对应“图 14-47 主控发送和接收（连续发送和接收模式）的时序图”中的①~⑧。

14.5.4 从属发送

从属发送是在从其他设备输入传送时钟的状态下 RL78 微控制器将数据发送给其他设备的运行。

3 线串行输入 / 输出	CSI00
对象通道	SAU0 的通道 0
使用的引脚	SCK00、SO00
中断	INTCSI00 可选择传送结束中断（单次传送模式）或者缓冲器空中断（连续传送模式）。
错误检测标志	只有溢出错误检测标志（OVFmn）。
传送数据长度	7 位或者 8 位
传送速率	Max. $f_{MCK}/6$ [Hz] 注 1、注 2
数据相位	可通过 SCRmn 寄存器的 DAPmn 位进行选择。 • DAPmn=0: 在串行时钟开始运行时，开始数据输出。 • DAPmn=1: 在串行时钟开始运行的半个时钟前，开始数据输出。
时钟相位	可通过 SCRmn 寄存器的 CKPmn 位进行选择。 • CKPmn=0: 不反相 • CKPmn=1: 反相
数据方向	MSB 优先或者 LSB 优先

注 1. 因为在内部对 SCK00 引脚输入的外部串行时钟进行采样后使用，所以最大传送速率为 $f_{MCK}/6$ [Hz]。

2. 必须在满足此条件并且满足电特性的外围功能特性（参照“第 32 章 电特性”）的范围内使用。

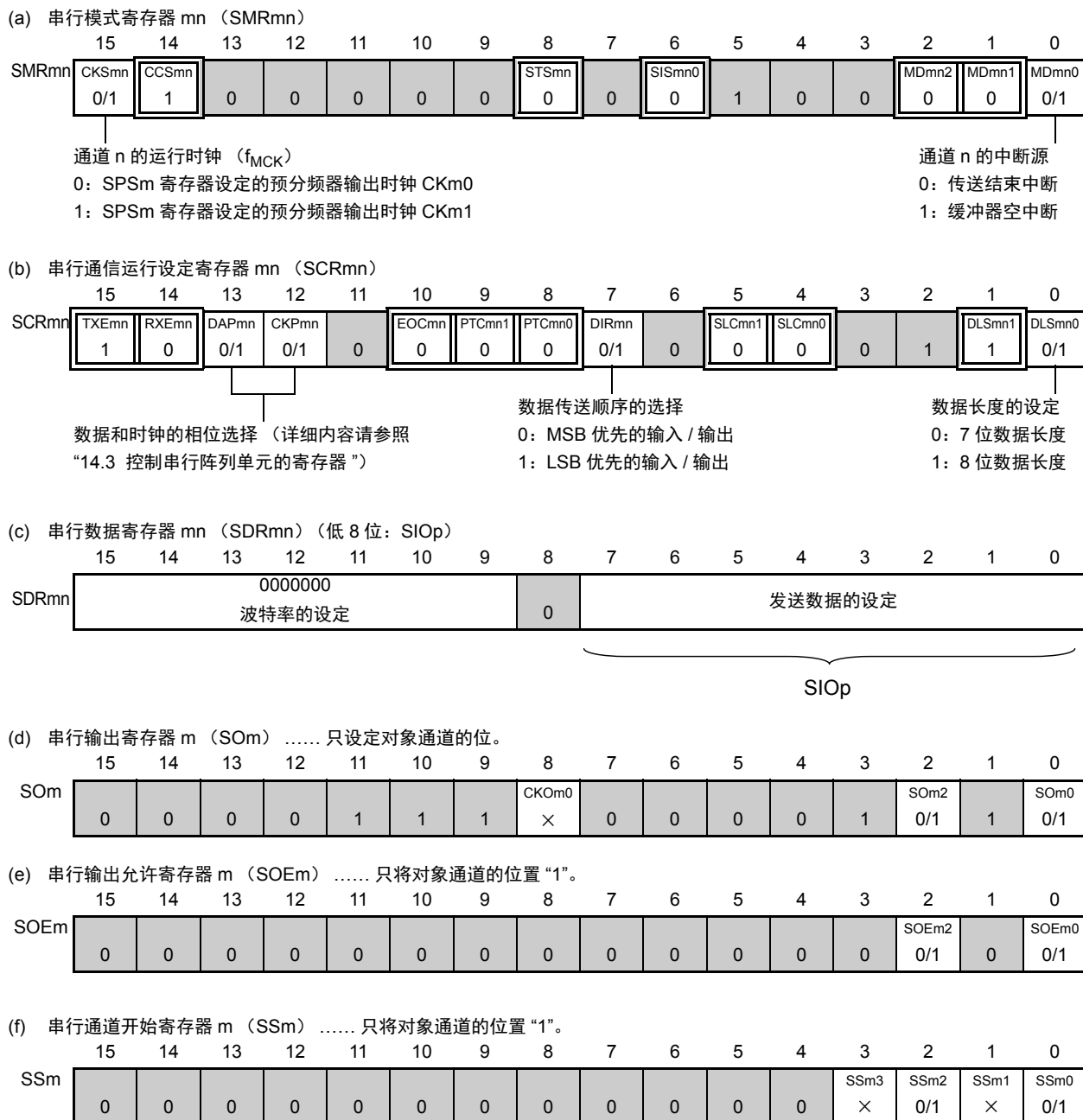
备注 1. f_{MCK} : 对象通道的运行时钟频率

f_{SCK} : 串行时钟频率

2. m: 单元号 (m=0) n: 通道号 (n=0) mn=00

(1) 寄存器的设定

图 14-49 3 线串行 I/O (CSI00) 从属发送时的寄存器设定内容例子



备注 1. m: 单元号 (m=0) n: 通道号 (n=0) p: CSI 号 (p=00) mn=00

2. □: 在 CSI 从属发送模式中为固定设定。■: 不能设定 (设定初始值)。

×: 这是在此模式中不能使用的位 (在其他模式中也不使用的情况下, 设定初始值)。

0/1: 根据用户的用途置“0”或者“1”。

(2) 操作步骤

图 14-50 从属发送的初始设定步骤

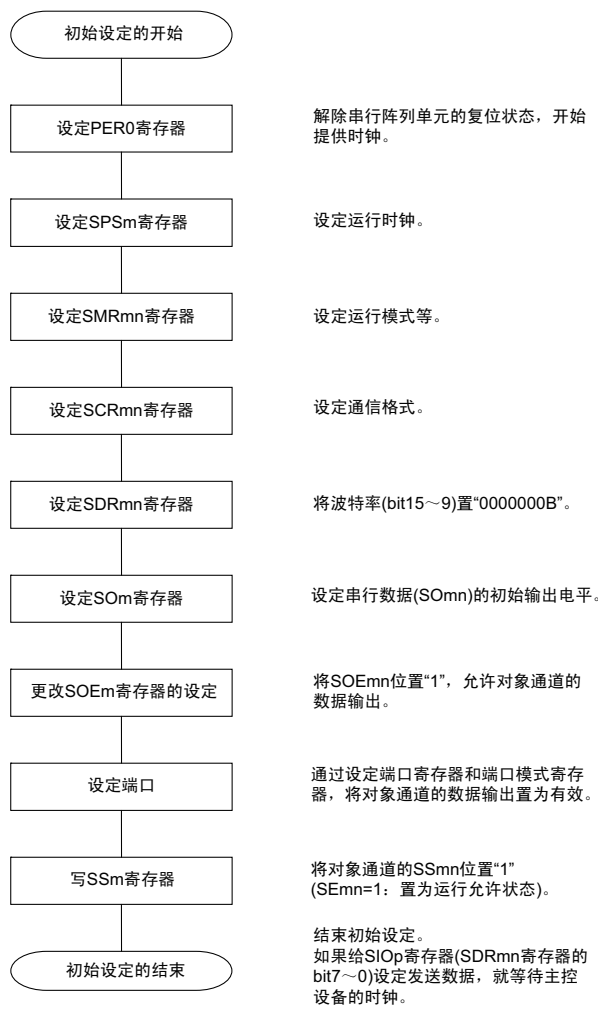


图 14-51 从属发送的中止步骤

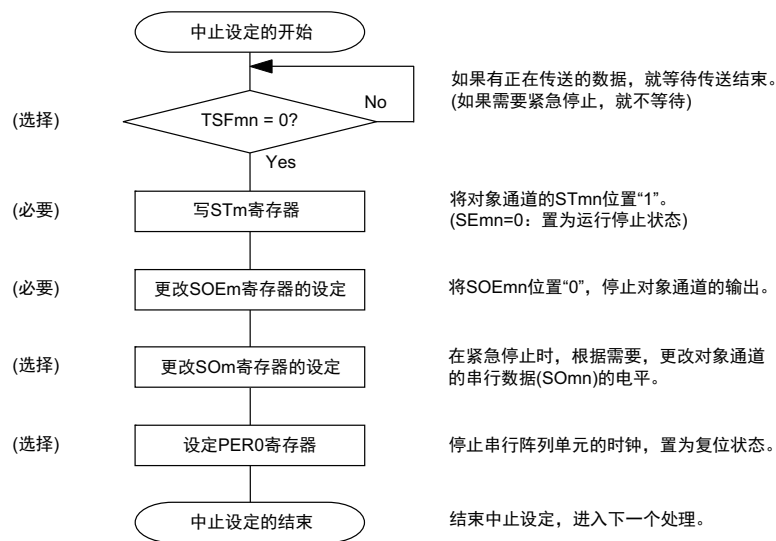
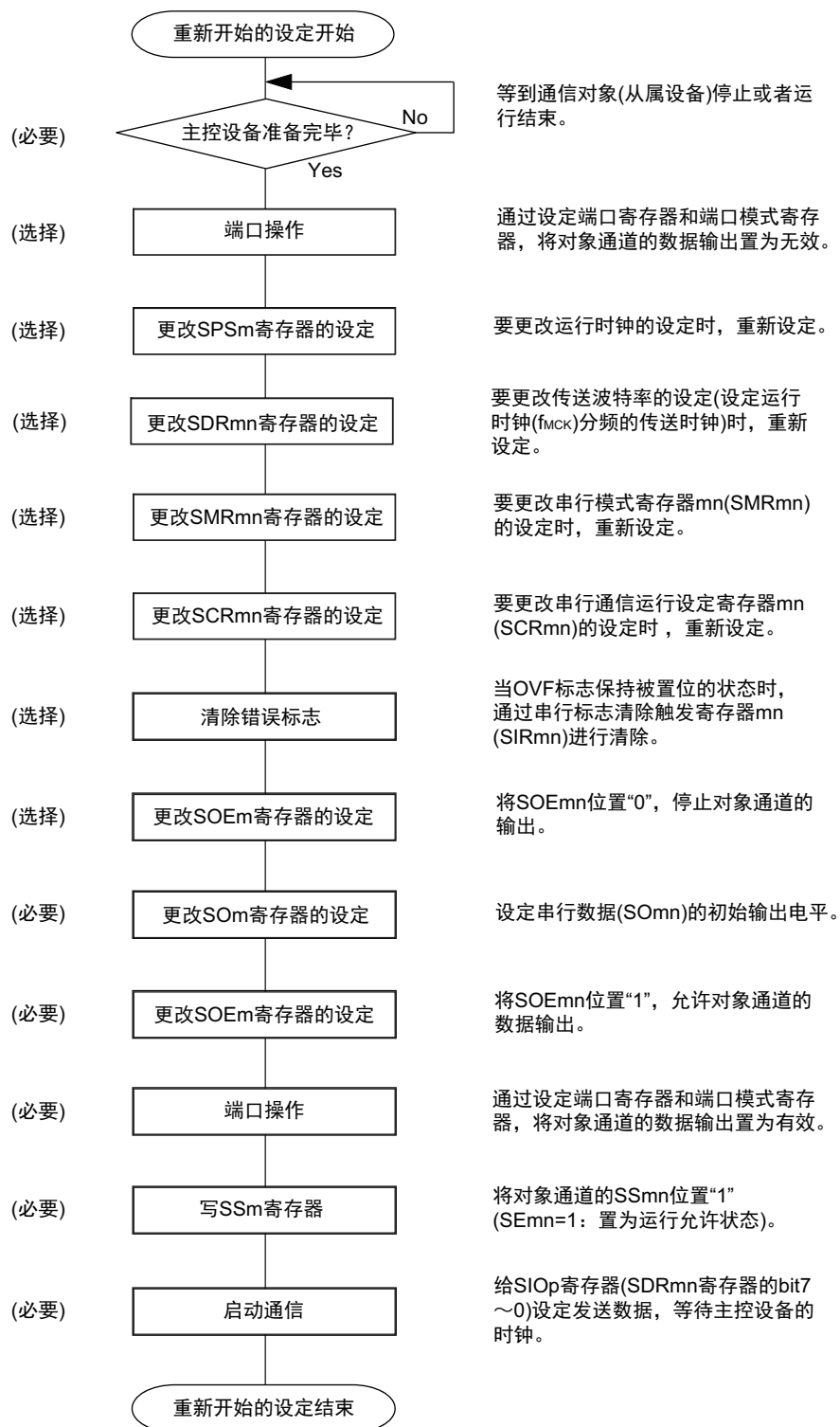


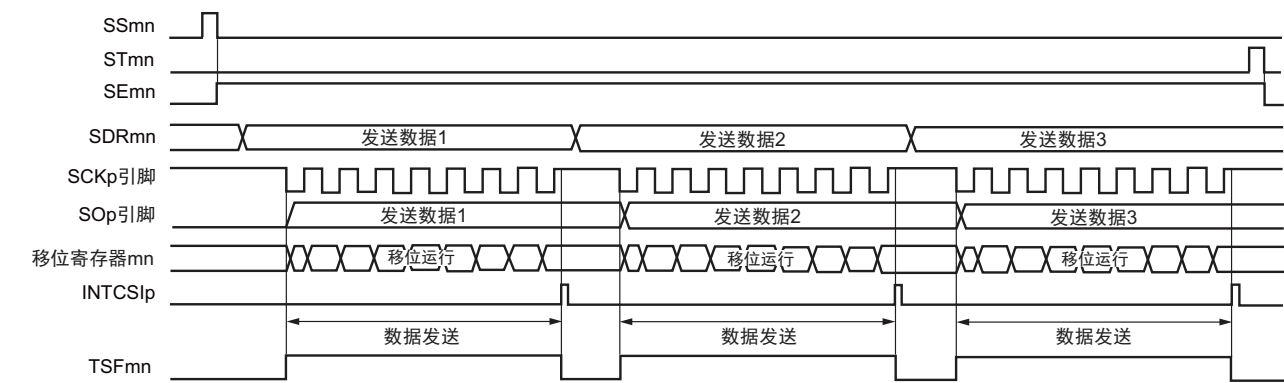
图 14-52 重新开始从属发送的设定步骤



备注 如果在中止设定中改写 PER0 来停止提供时钟，就必须在等待通信对象（主控设备）的停止或者通信结束后进行初始设定而不是进行重新开始的设定。

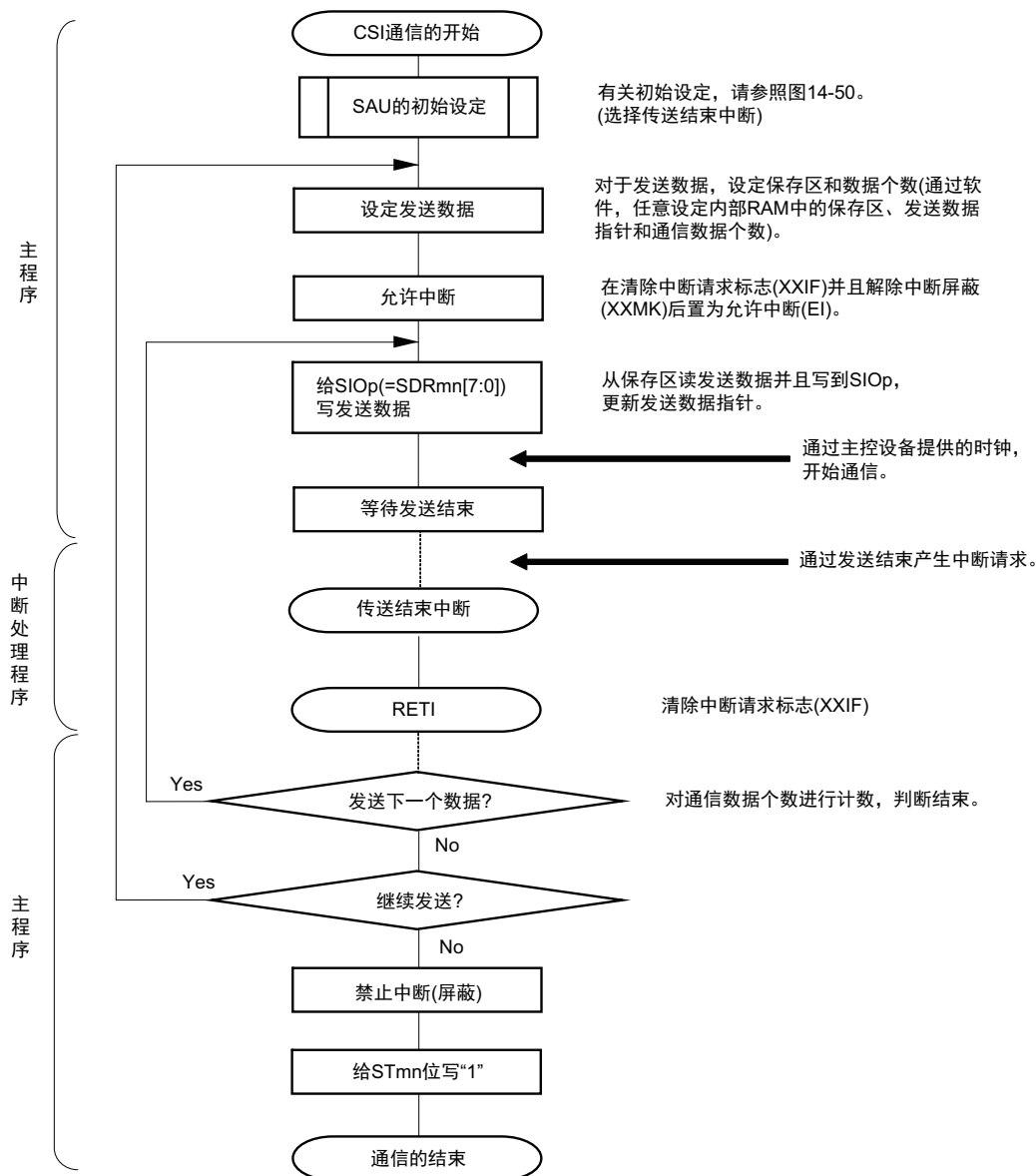
(3) 处理流程（单次发送模式）

图 14-53 从属发送（单次发送模式）的时序图（类型 1：DAPmn=0、CKPmn=0）



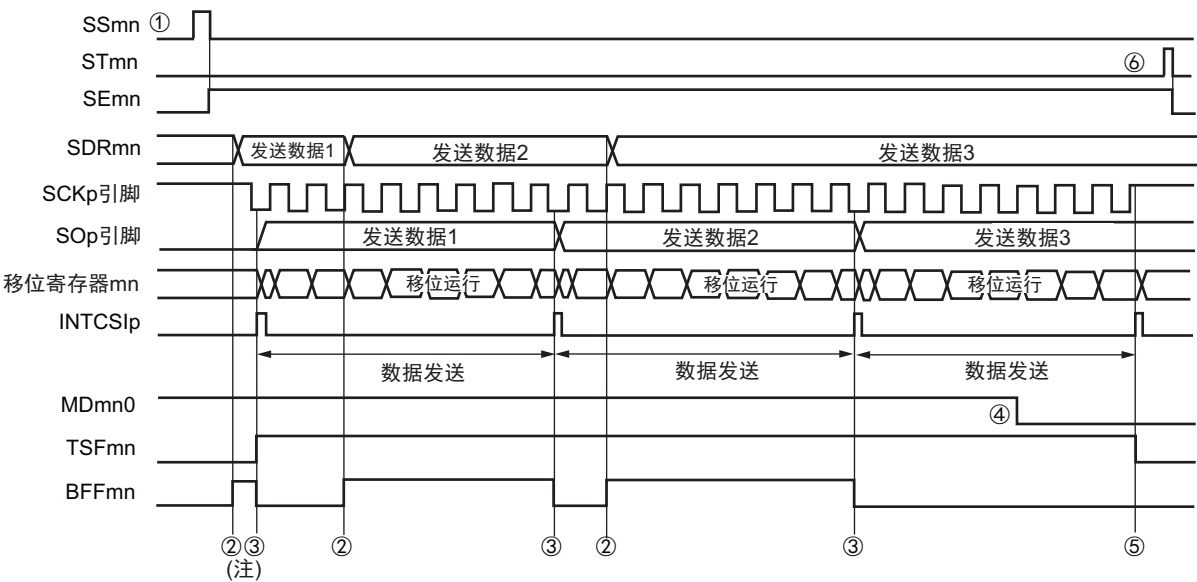
备注 m: 单元号 (m=0) n: 通道号 (n=0) p: CSI 号 (p=00) mn=00

图 14-54 从属发送（单次发送模式）的流程图



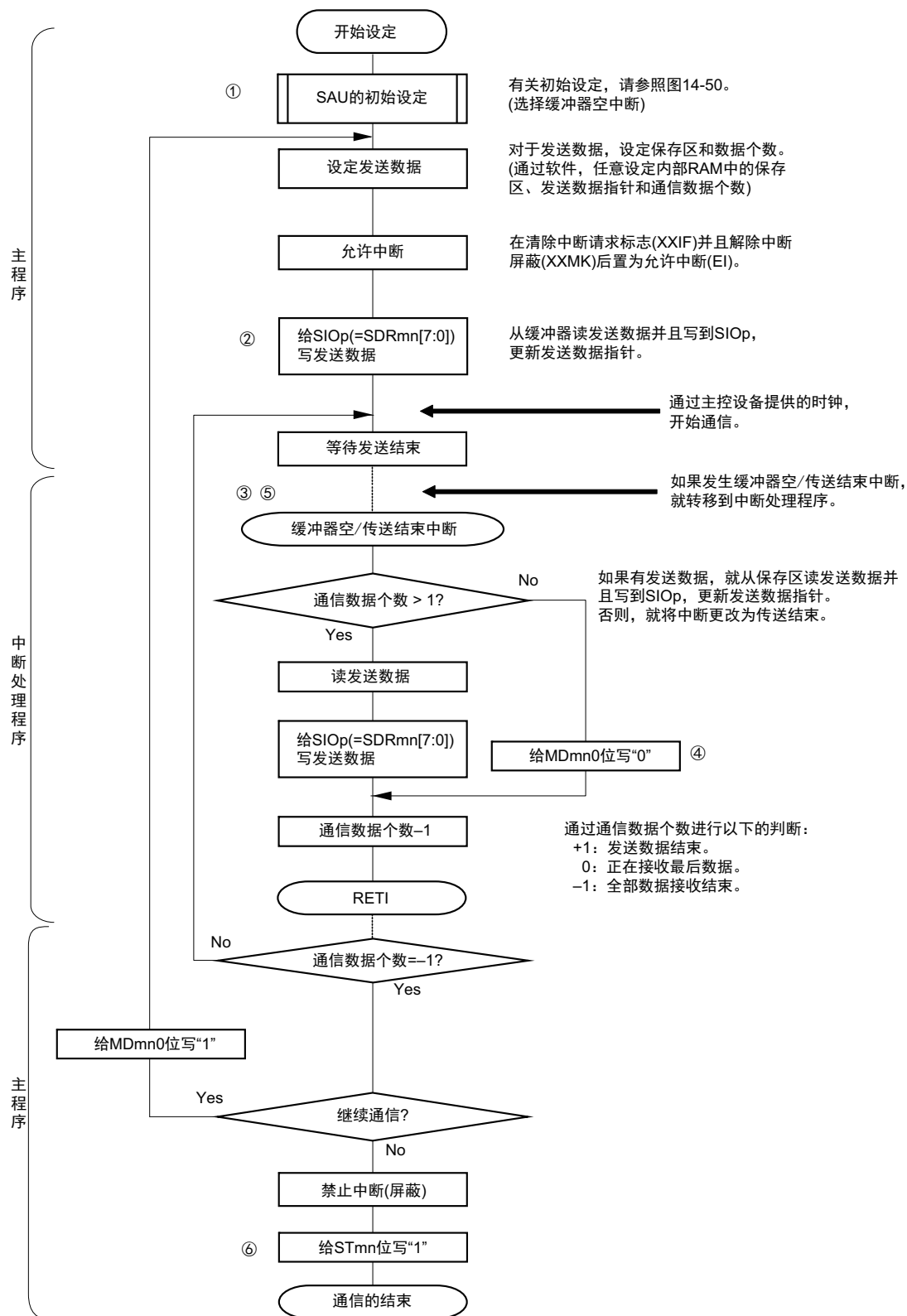
(4) 处理流程（连续发送模式）

图 14-55 从属发送（连续发送模式）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”期间（有效数据保存在串行数据寄存器 mn（SDRmn）时）给 SDRmn 寄存器写发送数据，就重写发送数据。
- 注意 即使在运行中也能改写串行模式寄存器 mn（SMRmn）的 MDmn0 位。但是，必须在开始传送最后一位之前进行改写。
- 备注 m: 单元号（m=0） n: 通道号（n=0） p: CSI 号（p=00） mn=00

图 14-56 从属发送（连续发送模式）的流程图



备注 图中的①~⑥对应“图 14-55 从属发送（连续发送模式）的时序图”中的①~⑥。

14.5.5 从属接收

从属接收是在其他设备输入传送时钟的状态下 RL78 微控制器从其他设备接收数据的运行。

3 线串行输入 / 输出	CSI00
对象通道	SAU0 的通道 0
使用的引脚	SCK00、SI00
中断	INTCSI00 只限于传送结束中断（禁止设定缓冲器空中断）。
错误检测标志	只有溢出错误检测标志（OVFmn）。
传送数据长度	7 位或者 8 位
传送速率	Max. $f_{MCK}/6$ [Hz] 注 1、注 2
数据相位	可通过 SCRmn 寄存器的 DAPmn 位进行选择。 • DAPmn=0: 在串行时钟开始运行时，开始数据输出。 • DAPmn=1: 在串行时钟开始运行的半个时钟前，开始数据输出。
时钟相位	可通过 SCRmn 寄存器的 CKPmn 位进行选择。 • CKPmn=0: 不反相 • CKPmn=1: 反相
数据方向	MSB 优先或者 LSB 优先

注 1. 因为在内部对 SCK00 引脚输入的外部串行时钟进行采样后使用，所以最大传送速率为 $f_{MCK}/6$ [Hz]。

2. 必须在满足此条件并且满足电特性的外围功能特性（参照“第 32 章 电特性”）的范围内使用。

备注 1. f_{MCK} : 对象通道的运行时钟频率

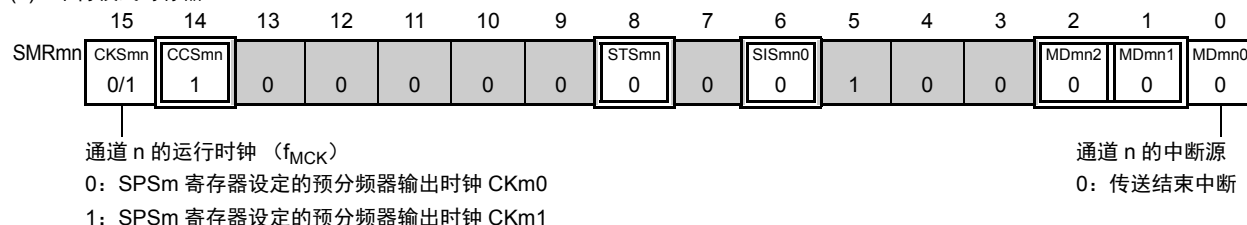
f_{SCK} : 串行时钟频率

2. m: 单元号 (m=0) n: 通道号 (n=0) mn=00

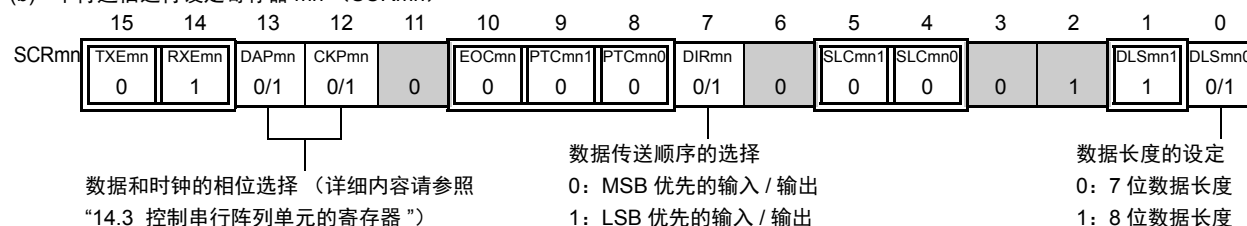
(1) 寄存器的设定

图 14-57 3 线串行 I/O (CSI00) 从属接收时的寄存器设定内容例子

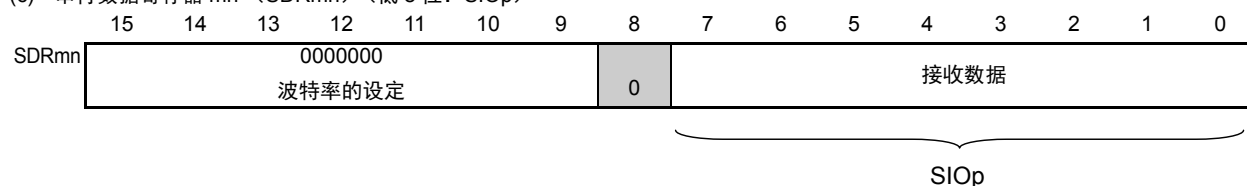
(a) 串行模式寄存器 mn (SMRmn)



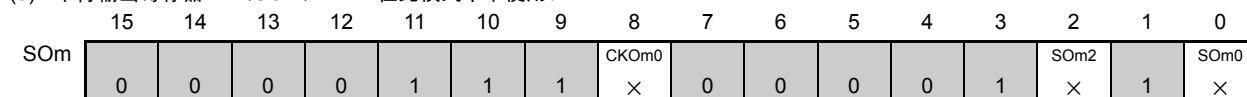
(b) 串行通信运行设定寄存器 mn (SCRmn)



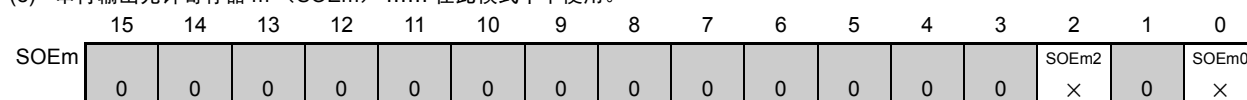
(c) 串行数据寄存器 mn (SDRmn) (低 8 位: SIOp)



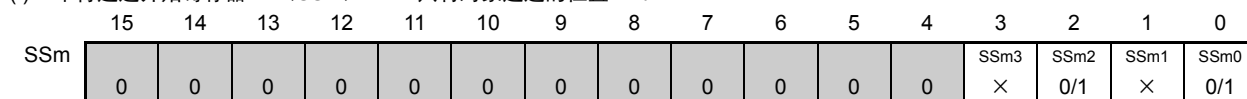
(d) 串行输出寄存器 m (SOM) 在此模式中不使用。



(e) 串行输出允许寄存器 m (SOEm) 在此模式中不使用。



(f) 串行通道开始寄存器 m (SSm) 只将对象通道的位置“1”。



备注 1. m: 单元号 (m=0) n: 通道号 (n=0) p: CSI 号 (p=00) mn=00

2. □: 在 CSI 从属接收模式中为固定设定。■: 不能设定 (设定初始值)。

×: 这是在此模式中不能使用的位 (在其他模式中也不使用的情况下, 设定初始值)。

0/1: 根据用户的用途置“0”或者“1”。

(2) 操作步骤

图 14-58 从属接收的初始设定步骤

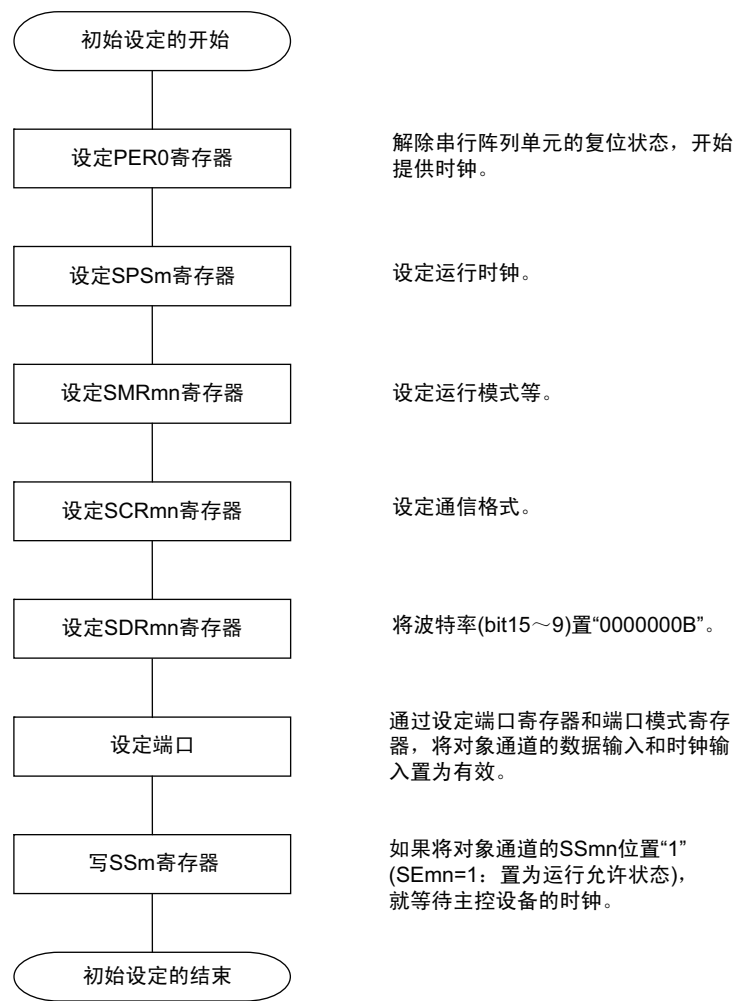


图 14-59 从属接收的中止步骤

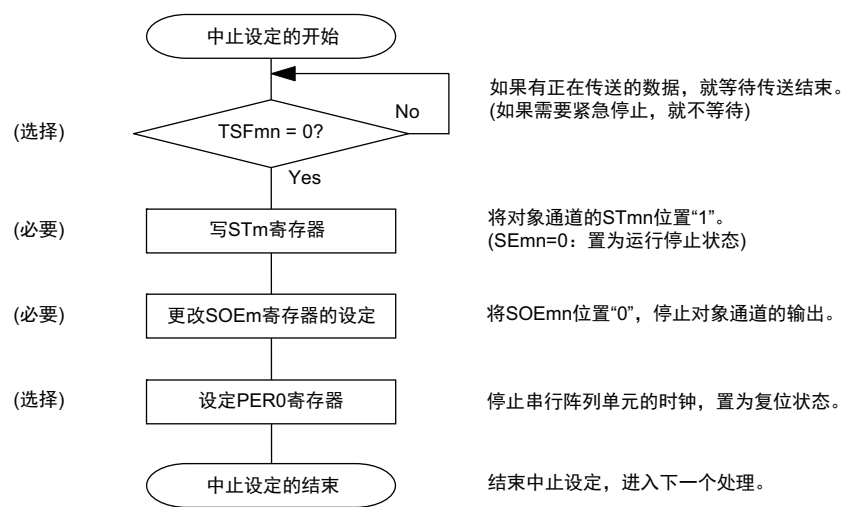
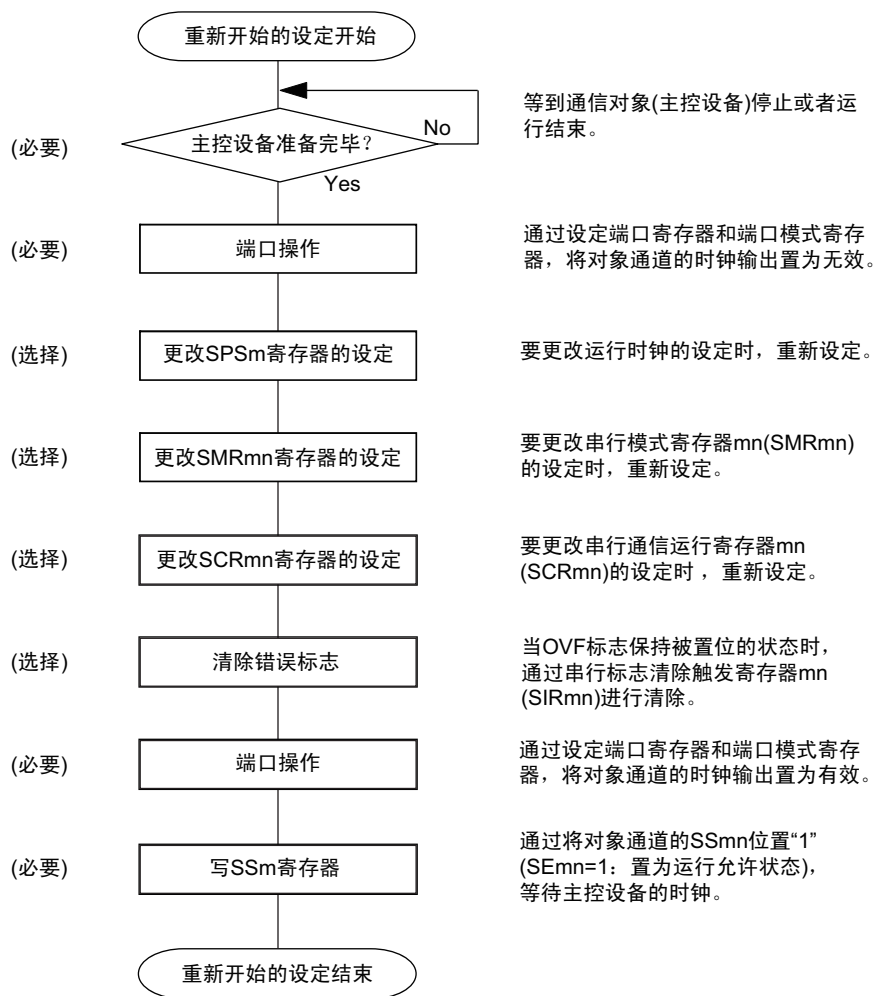


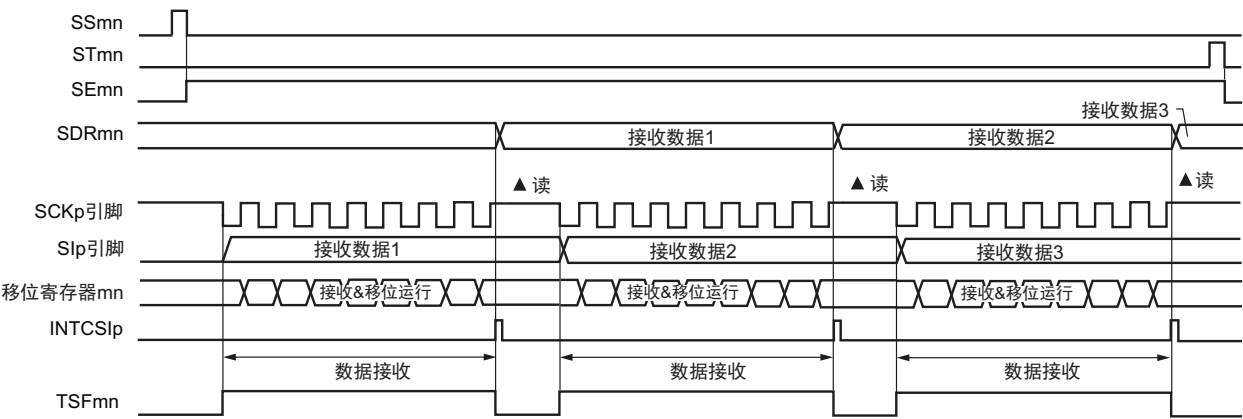
图 14-60 重新开始从属接收的设定步骤



备注 如果在中止设定中改写 PER0 来停止提供时钟，就必须在等待通信对象（主控设备）的停止或者通信结束后进行初始设定而不是进行重新开始的设定。

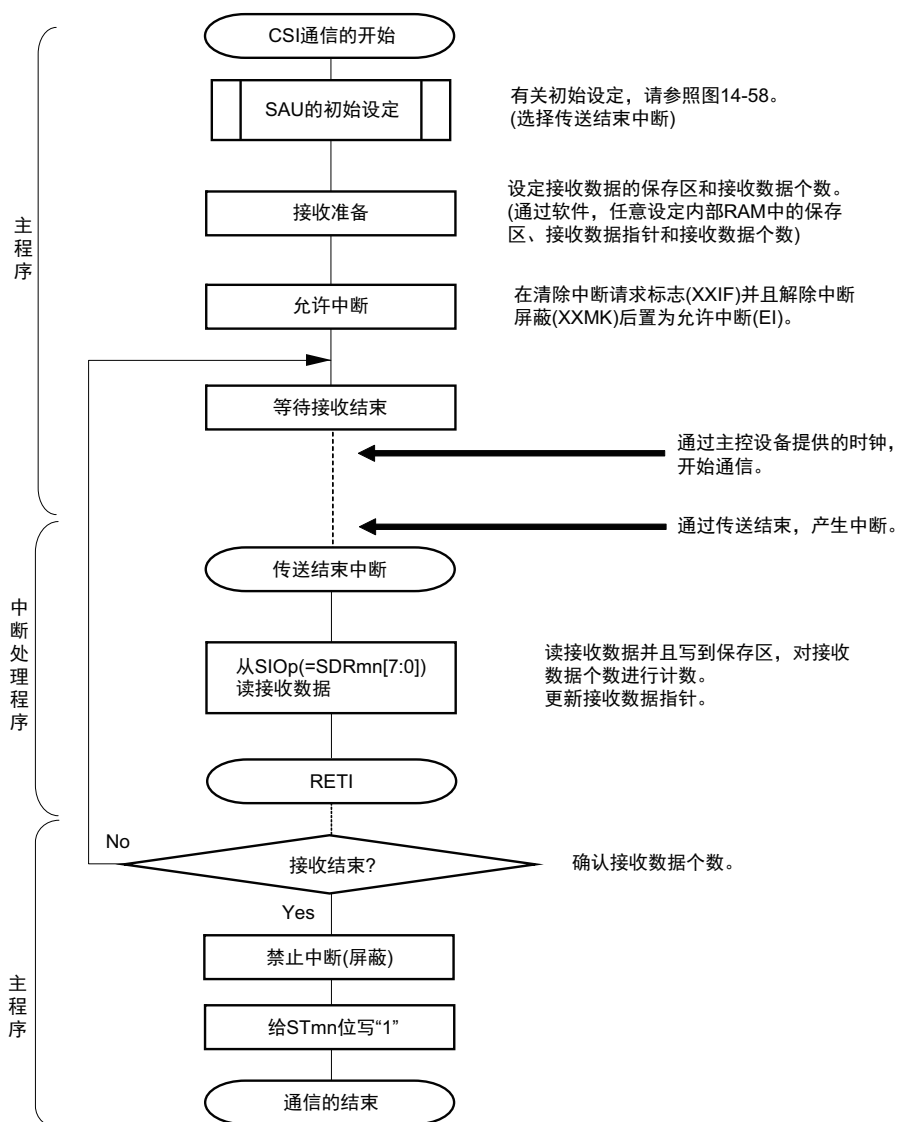
(3) 处理流程（单次接收模式）

图 14-61 从属接收（单次接收模式）的时序图（类型 1：DAPmn=0、CKPmn=0）



备注 m: 单元号 (m=0) n: 通道号 (n=0) p: CSI 号 (p=00) mn=00

图 14-62 从属接收（单次接收模式）的流程图



14.5.6 从属发送和接收

从属发送和接收是在其他设备输入传送时钟的状态下 RL78 微控制器和其他设备进行数据发送和接收的运行。

3 线串行输入 / 输出	CSI00
对象通道	SAU0 的通道 0
使用的引脚	SCK00、SI00、SO00
中断	INTCSI00 可选择传送结束中断（单次传送模式时）或者缓冲器空中断（连续传送模式时）。
错误检测标志	只有溢出错误检测标志（OVFmn）。
传送数据长度	7 位或者 8 位
传送速率	Max. $f_{MCK}/6$ [Hz] 注 1、注 2
数据相位	可通过 SCRmn 寄存器的 DAPmn 位进行选择。 • DAPmn=0: 在串行时钟开始运行时，开始数据输出。 • DAPmn=1: 在串行时钟开始运行的半个时钟前，开始数据输出。
时钟相位	可通过 SCRmn 寄存器的 CKPmn 位进行选择。 • CKPmn=0: 不反相 • CKPmn=1: 反相
数据方向	MSB 优先或者 LSB 优先

注 1. 因为在内部对 SCK00 引脚输入的外部串行时钟进行采样后使用，所以最大传送速率为 $f_{MCK}/6$ [Hz]。

2. 必须在满足此条件并且满足电特性的外围功能特性（参照“第 32 章 电特性”）的范围内使用。

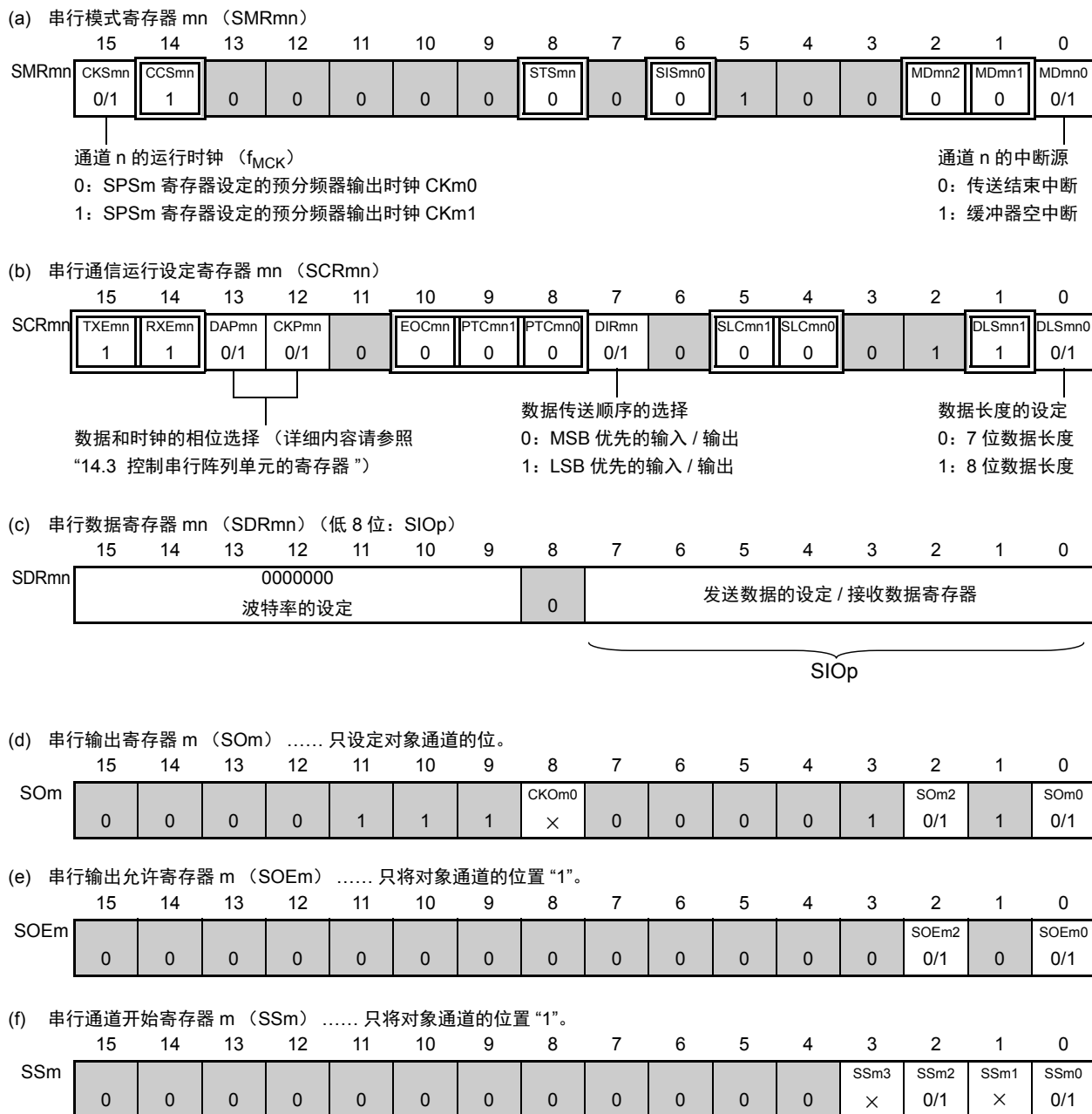
备注 1. f_{MCK} : 对象通道的运行时钟频率

f_{SCK} : 串行时钟频率

2. m: 单元号 (m=0) n: 通道号 (n=0) mn=00

(1) 寄存器的设定

图 14-63 3 线串行 I/O (CSI00) 从属发送和接收时的寄存器设定内容例子



注意 在主控设备开始输出时钟前，必须给 SI0p 寄存器设定发送数据。

备注 1. m: 单元号 (m=0) n: 通道号 (n=0) p: CSI 号 (p=00) mn=00

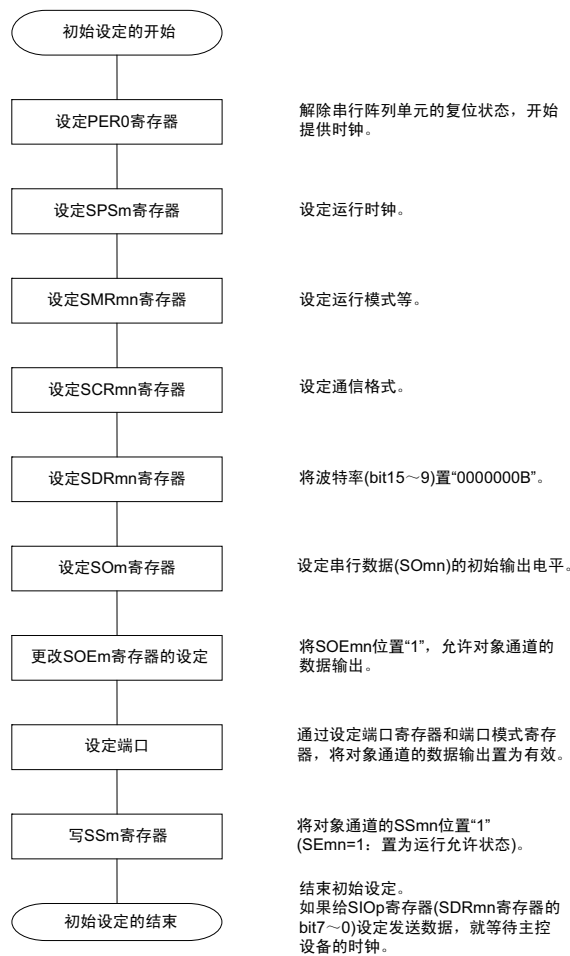
2. : 在 CSI 从属发送和接收模式中为固定设定。■: 不能设定 (设定初始值)。

×: 这是在此模式中不能使用的位 (在其他模式中也不使用的情况下, 设定初始值)。

0/1: 根据用户的用途置“0”或者“1”。

(2) 操作步骤

图 14-64 从属发送和接收的初始设定步骤



注意 在主控设备开始输出时钟前，必须给 SIOp 寄存器设定发送数据。

图 14-65 从属发送和接收的中止步骤

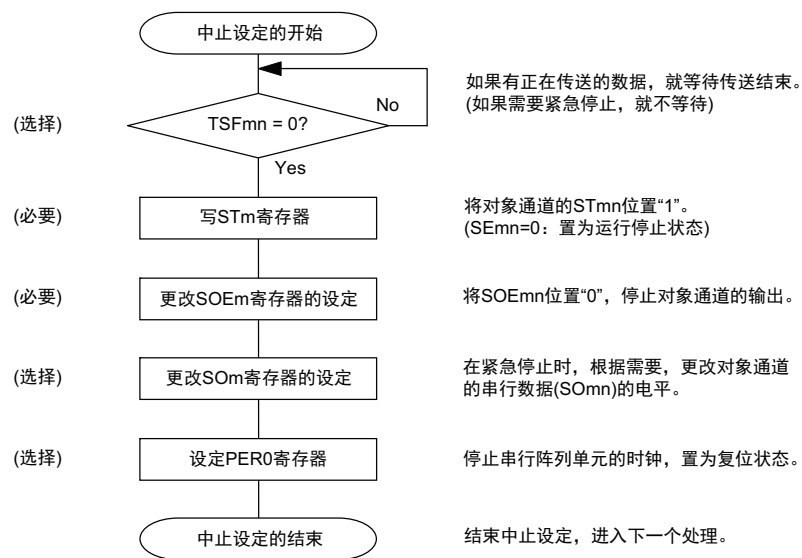
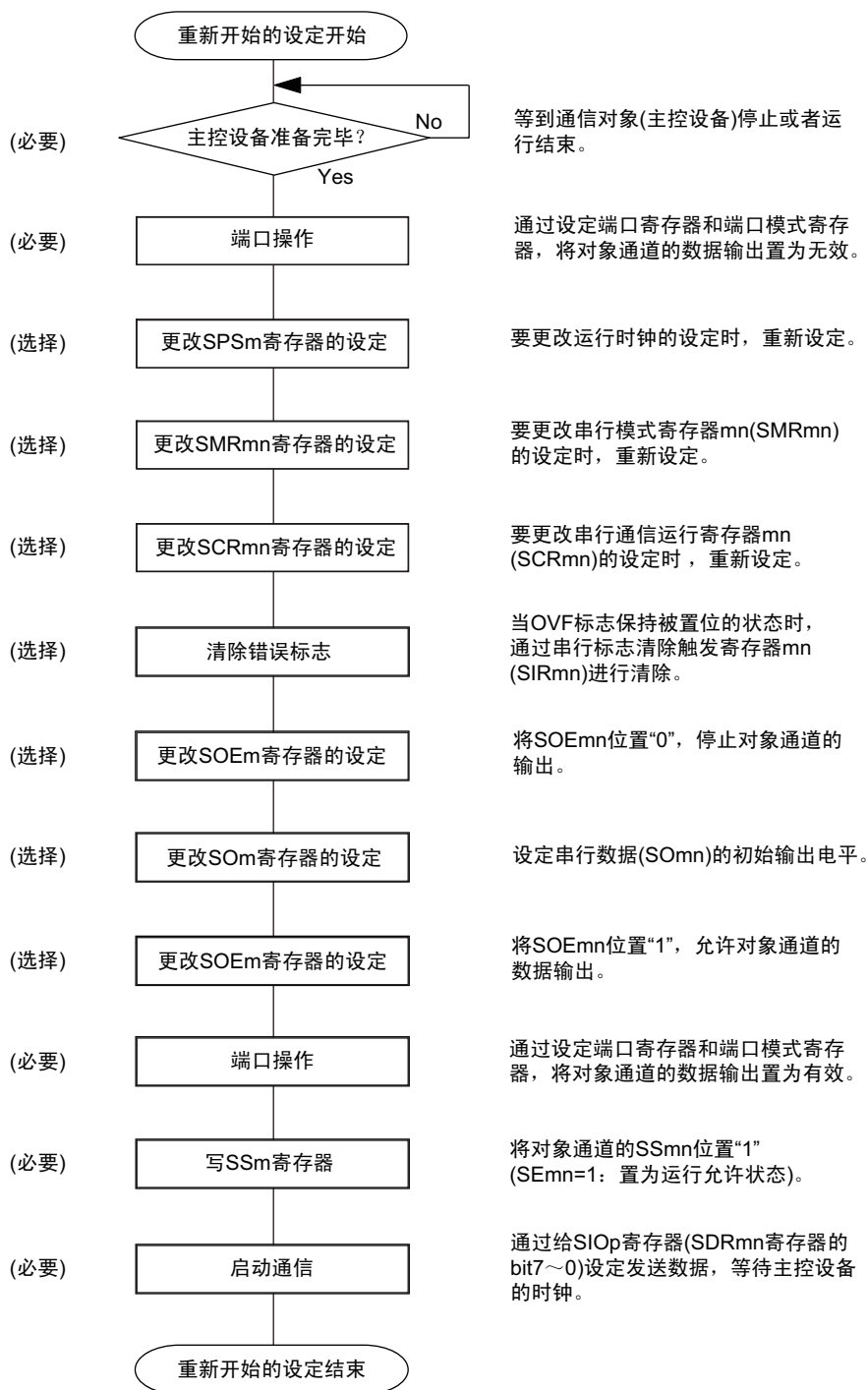


图 14-66 重新开始从属发送和接收的设定步骤

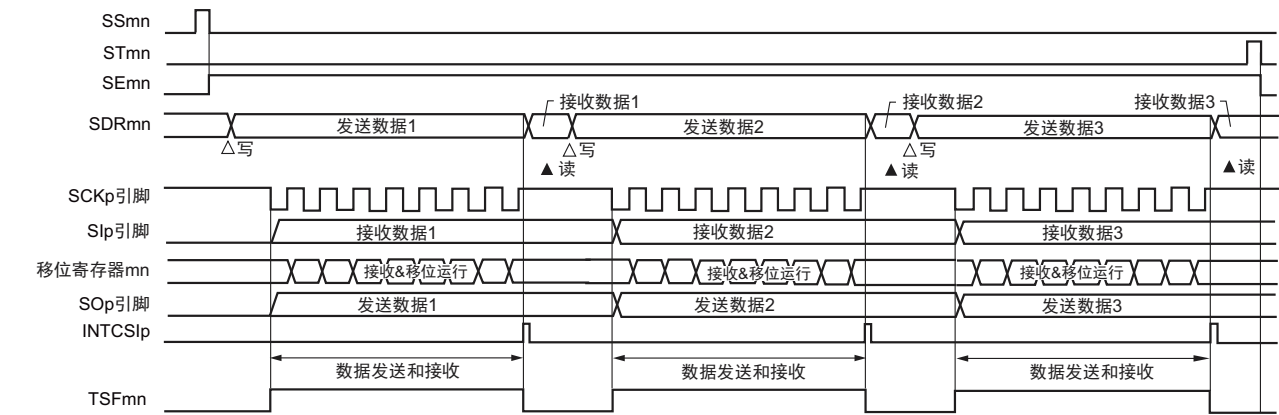


注意 1. 在主控设备开始输出时钟前, 必须给 SIOp 寄存器设定发送数据。

2. 如果在中止设定中改写 PER0 来停止提供时钟, 就必须在等待通信对象 (主控设备) 的停止或者通信结束后进行初始设定而不是进行重新开始的设定。

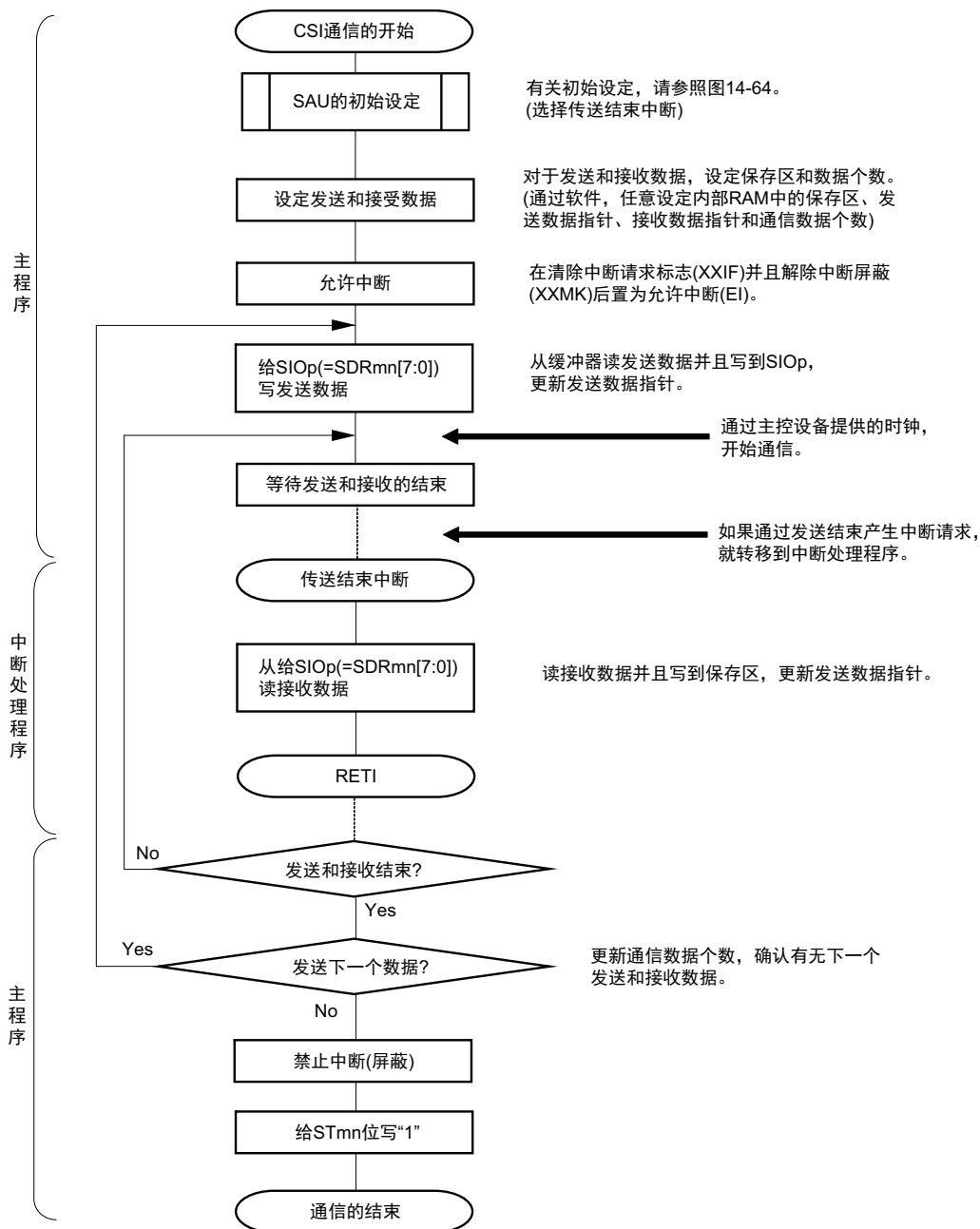
(3) 处理流程（单次发送和接收模式）

图 14-67 从属发送和接收（单次发送和接收模式）的时序图（类型 1：DAPmn=0、CKPmn=0）



备注 m: 单元号 (m=0) n: 通道号 (n=0) p: CSI 号 (p=00) mn=00

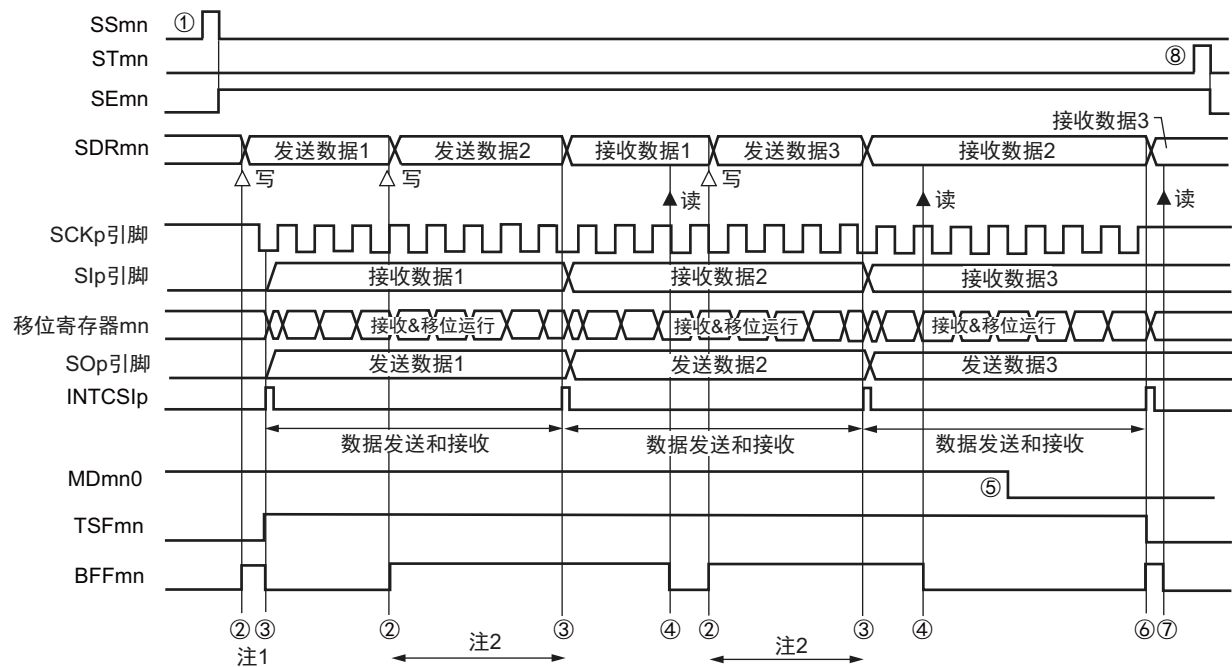
图 14-68 从属发送和接收（单次发送和接收模式）的流程图



注意 在主控设备开始输出时钟前，必须给 SIOp 寄存器设定发送数据。

(4) 处理流程（连续发送和接收模式）

图 14-69 从属发送和接收（连续发送和接收模式）的时序图（类型 1：DAPmn=0、CKPmn=0）



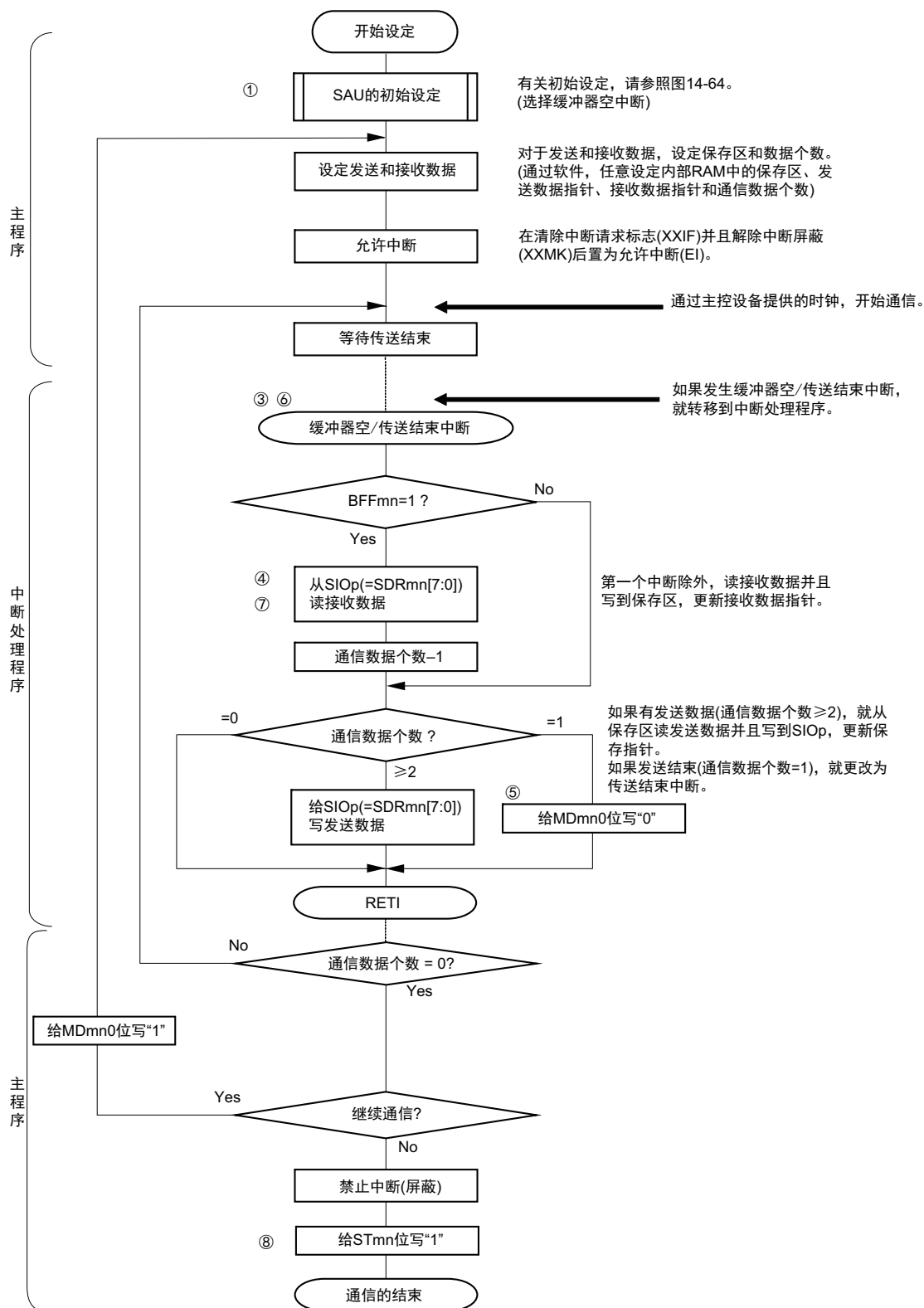
- 注 1. 如果在串行状态寄存器 mn（SSRmn）的 BFFmn 位为“1”期间（有效数据保存在串行数据寄存器 mn（SDRmn）时）给 SDRmn 寄存器写发送数据，就重写发送数据。
2. 如果在此期间读取 SDRmn 寄存器，就能读发送数据。此时，不影响传送运行。

注意 即使在运行中也能改写串行模式寄存器 mn（SMRmn）的 MDmn0 位。但是，为了能赶上最后发送数据的传送结束中断，必须在开始传送最后一位之前进行改写。

备注 1. 图中的①～⑧对应“图 14-70 从属发送和接收（连续发送和接收模式）的流程图”中的①～⑧。

2. m: 单元号（m=0） n: 通道号（n=0） p: CSI 号（p=00） mn=00

图 14-70 从属发送和接收（连续发送和接收模式）的流程图



注意 在主控设备开始输出时钟前, 必须给 SIOp 寄存器设定发送数据。

备注 图中的①~⑧对应“图 14-69 从属发送和接收（连续发送和接收模式）的时序图”中的①~⑧。

14.5.7 SNOOZE 模式功能

这是在 STOP 模式中通过检测 SCKp 引脚的输入使 CSI 进行接收的模式。通常，在 STOP 模式中 CSI 停止通信，但是使用此模式并且通过检测 SCKp 引脚的输入，能在 CPU 不运行的状态下进行 CSI 的接收。只有 CSI00 才能设定 SNOOZE 模式。

要在 SNOOZE 模式中使用 CSI 时，必须在即将要转移到 STOP 模式前进行以下设定（参照“图 14-72 SNOOZE 模式运行（启动一次）的流程图”、“图 14-74 SNOOZE 模式运行（连续启动）的流程图”）。

- 在即将要转移到 STOP 模式前，将串行待机控制寄存器 m（SSCm）的 SWCm 位置“1”。初始设定结束后，将串行通道开始寄存器 m（SSm）的 SSm1 位置“1”。

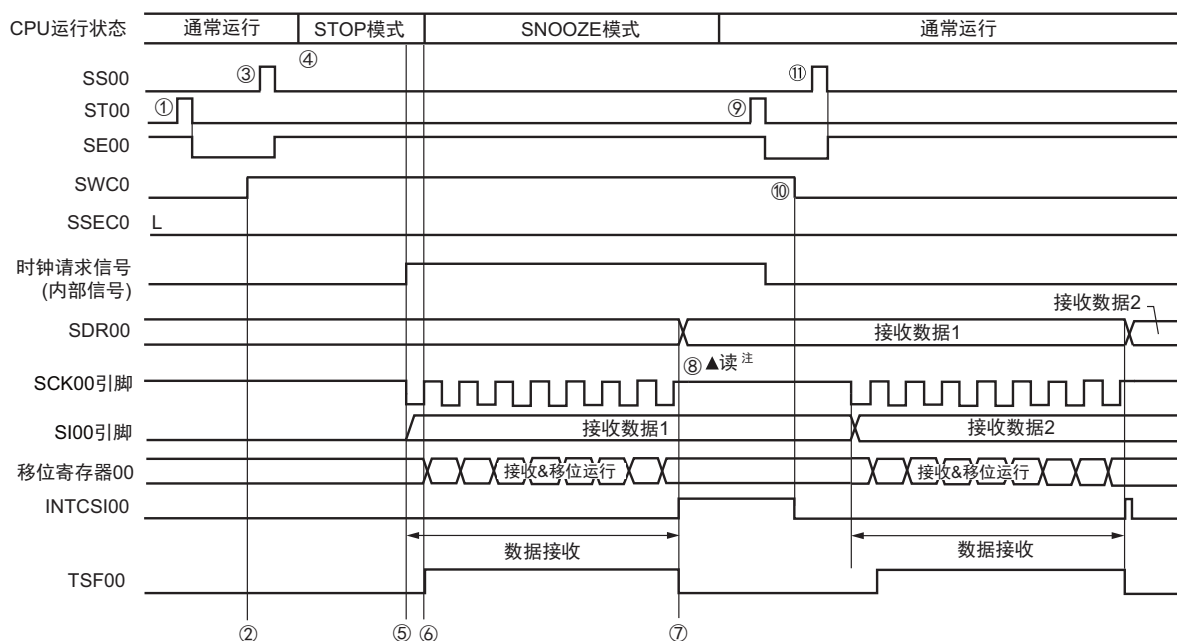
在转移到 STOP 模式后，如果检测到 SCKp 引脚的边沿，就开始 CSI 接收。

注意 1. 只有在选择高速内部振荡器时钟作为 f_{CLK} 时才能设定 SNOOZE 模式。

2. 在 SNOOZE 模式中使用时，最大传送速率为 1Mbps。

(1) SNOOZE 模式运行（启动一次）

图 14-71 SNOOZE 模式运行（启动一次）的时序图（类型 1：DAPmn=0、CKPmn=0）



注 必须在 SWCm 位为“1”的状态下并且检测到下一个 SCKp 引脚的边沿前读接收数据。

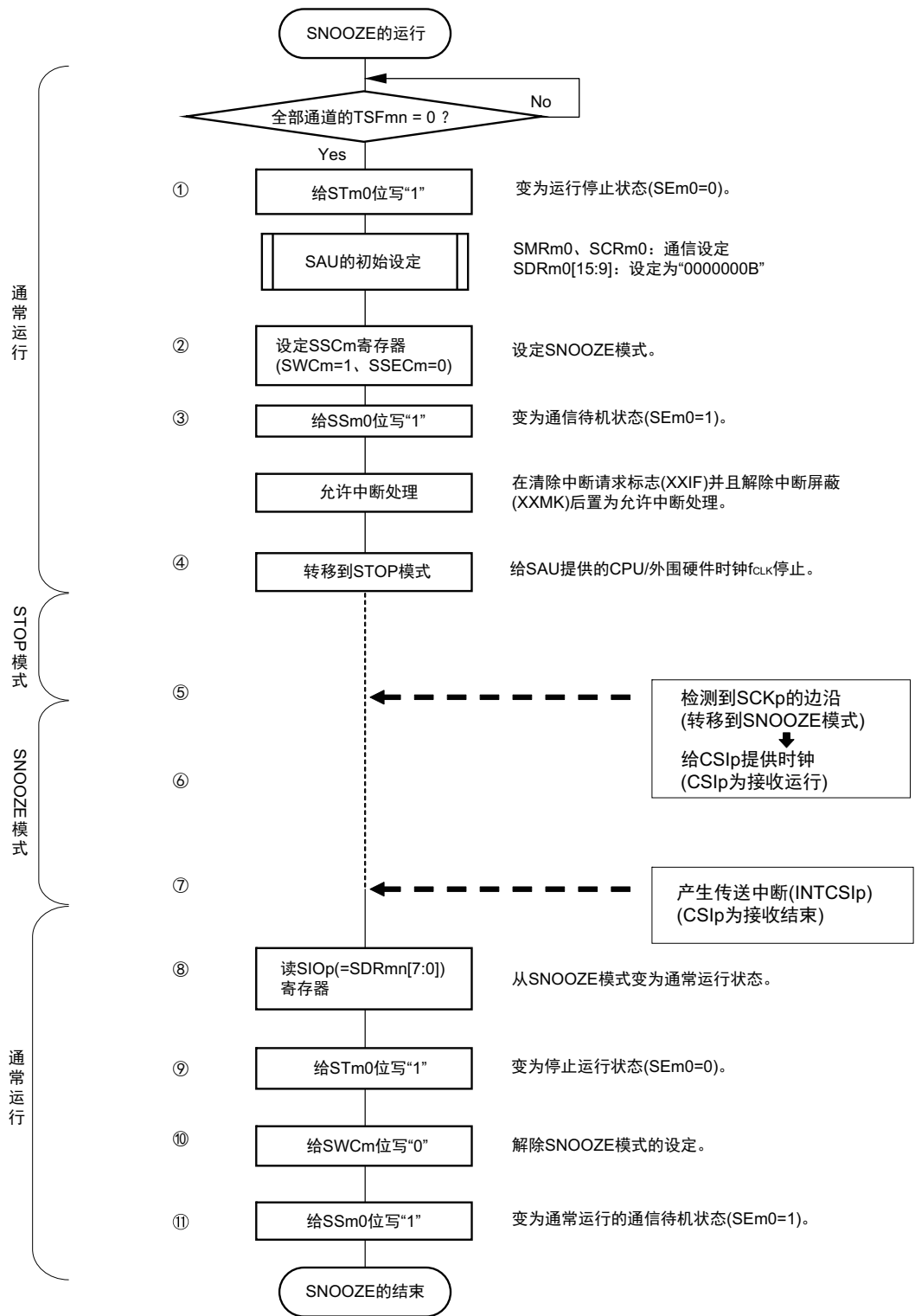
注意 1. 在向 SNOOZE 模式转移前或者在 SNOOZE 模式中接收结束后，必须将 STm0 位置“1”（清除 SEm0 位并且停止运行），而且还必须在接收结束后清除 SWCm 位（解除 SNOOZE 模式）。

2. 在 SWCm=1 时，BFFm1、OVFm1 标志不发生变化。

备注 1. 图中的①～⑪对应“图 14-72 SNOOZE 模式运行（启动一次）的流程图”中的①～⑪。

2. m=0、p=00

图 14-72 SNOOZE 模式运行（启动一次）的流程图

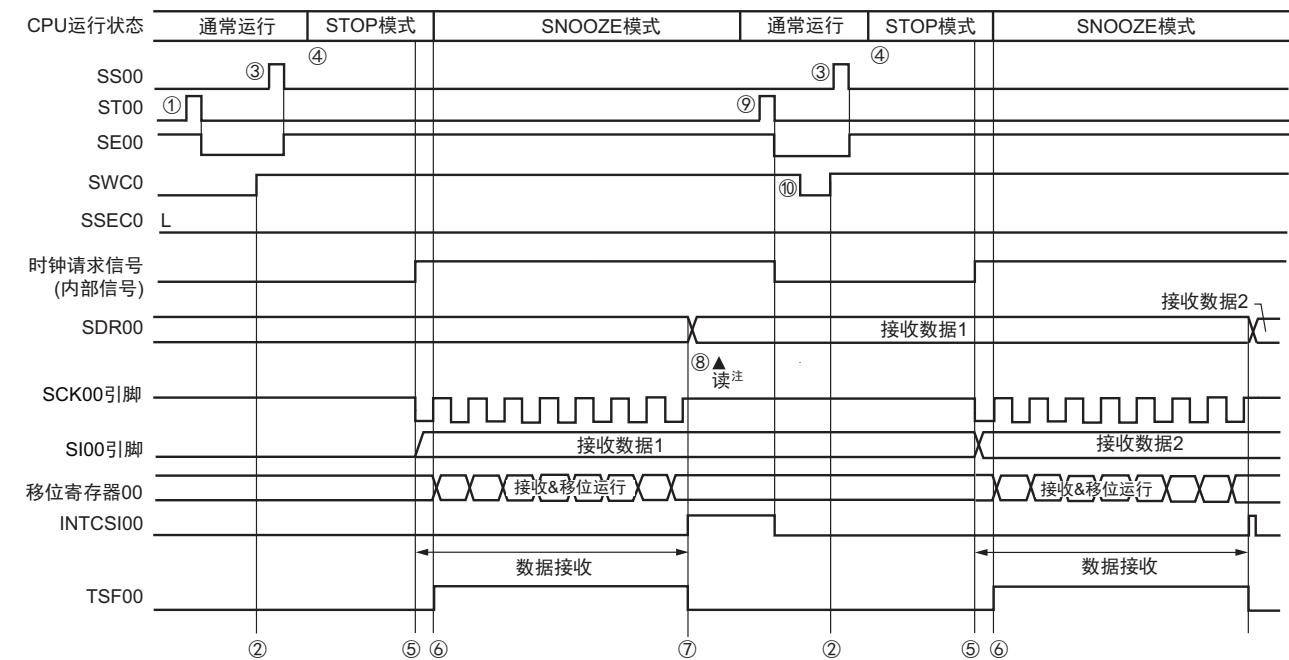


备注 1. 图中的①～⑪对应“图 14-71 SNOOZE 模式运行（启动一次）的时序图”中的①～⑪。

2. m=0、p=00

(2) SNOOZE 模式运行（连续启动）

图 14-73 SNOOZE 模式运行（连续启动）的时序图（类型 1：DAPmn=0、CKPmn=0）



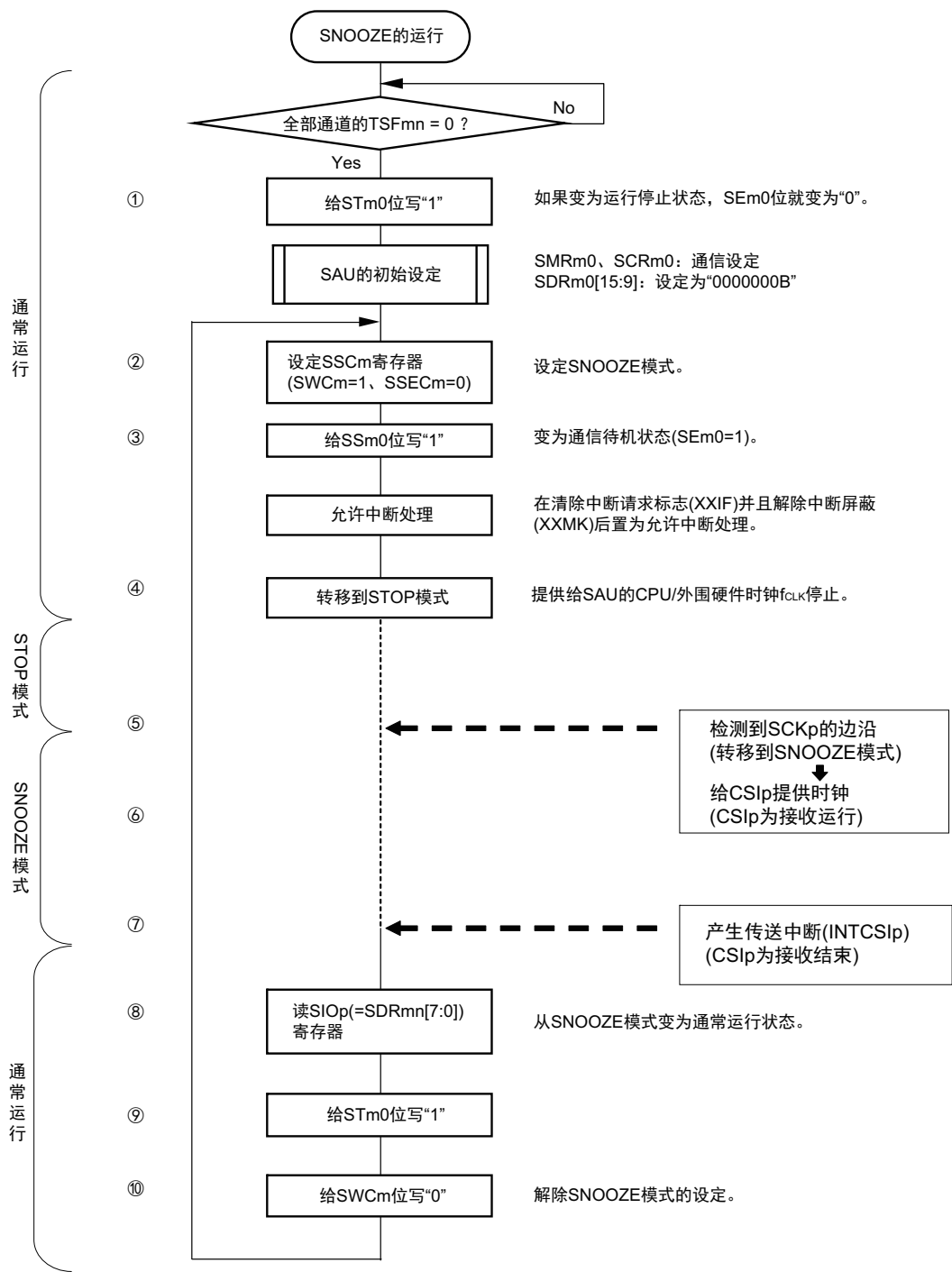
注 必须在 SWCm 位为“1”的状态下并且在检测到下一个 SCKp 引脚的边沿前读接收数据。

- 注意 1. 在向 SNOOZE 模式转移前或者在 SNOOZE 模式中接收结束后，必须将 STm0 位置“1”（清除 SEm0 位并且停止运行），而且还必须在接收结束后清除 SWCm 位（解除 SNOOZE 模式）。
2. 在 SWCm=1 时，BFFm1、OVFm1 标志不发生变化。

备注 1. 图中的①~⑩对应“图 14-74 SNOOZE 模式运行（连续启动）的流程图”中的①~⑩。

2. m=0, p=00

图 14-74 SNOOZE 模式运行（连续启动）的流程图



备注 1. 图中的①～⑩对应“图 14-73 SNOOZE 模式运行（连续启动）的时序图”中的①～⑩。

2. m=0, p=00

14.5.8 传送时钟频率的计算

3 线串行 I/O（CSI00）通信的传送时钟频率能用以下计算式进行计算。

(1) 主控设备

$$(\text{传送时钟频率}) = \{ \text{对象通道的运行时钟 (f}_{\text{MCK}}) \text{ 频率} \} \div (\text{SDRmn}[15:9] + 1) \div 2 [\text{Hz}]$$

(2) 从属设备

$$(\text{传送时钟频率}) = \{ \text{主控设备提供的串行时钟 (SCK) 频率} \} \text{注} \quad [\text{Hz}]$$

注 容许的最大传送时钟频率为 $f_{\text{MCK}}/6$ 。

备注 因为SDRmn[15:9]的值为串行数据寄存器mn（SDRmn）的bit15～9的值（0000000B～1111111B），所以为0～127。

运行时钟（ f_{MCK} ）取决于串行时钟选择寄存器 m（SPSm）和串行模式寄存器 mn（SMRmn）的 bit15（CKSmn）。

表 14-2 用于 3 线串行输入 / 输出的运行时钟的选择

SMRmn 寄存器	SPSm 寄存器								运行时钟（f _{MCK} ）注	
CKSmn	PRS m13	PRS m12	PRS m11	PRS m10	PRS m03	PRS m02	PRS m01	PRS m00		f _{CLK} =24 MHz 运行时
0	X	X	X	X	0	0	0	0	f _{CLK}	24 MHz
	X	X	X	X	0	0	0	1	f _{CLK} /2	12 MHz
	X	X	X	X	0	0	1	0	f _{CLK} /2 ²	6 MHz
	X	X	X	X	0	0	1	1	f _{CLK} /2 ³	3 MHz
	X	X	X	X	0	1	0	0	f _{CLK} /2 ⁴	1.5 MHz
	X	X	X	X	0	1	0	1	f _{CLK} /2 ⁵	750 kHz
	X	X	X	X	0	1	1	0	f _{CLK} /2 ⁶	375 kHz
	X	X	X	X	0	1	1	1	f _{CLK} /2 ⁷	187.5 kHz
	X	X	X	X	1	0	0	0	f _{CLK} /2 ⁸	93.8 kHz
	X	X	X	X	1	0	0	1	f _{CLK} /2 ⁹	46.9 kHz
	X	X	X	X	1	0	1	0	f _{CLK} /2 ¹⁰	23.4 kHz
	X	X	X	X	1	0	1	1	f _{CLK} /2 ¹¹	11.7 kHz
	X	X	X	X	1	1	0	0	f _{CLK} /2 ¹²	5.86 kHz
	X	X	X	X	1	1	0	1	f _{CLK} /2 ¹³	2.93 kHz
	X	X	X	X	1	1	1	0	f _{CLK} /2 ¹⁴	1.46 kHz
	X	X	X	X	1	1	1	1	f _{CLK} /2 ¹⁵	732 Hz
1	0	0	0	0	X	X	X	X	f _{CLK}	24 MHz
	0	0	0	1	X	X	X	X	f _{CLK} /2	12 MHz
	0	0	1	0	X	X	X	X	f _{CLK} /2 ²	6 MHz
	0	0	1	1	X	X	X	X	f _{CLK} /2 ³	3 MHz
	0	1	0	0	X	X	X	X	f _{CLK} /2 ⁴	1.5 MHz
	0	1	0	1	X	X	X	X	f _{CLK} /2 ⁵	750 kHz
	0	1	1	0	X	X	X	X	f _{CLK} /2 ⁶	375 kHz
	0	1	1	1	X	X	X	X	f _{CLK} /2 ⁷	187.5 kHz
	1	0	0	0	X	X	X	X	f _{CLK} /2 ⁸	93.8 kHz
	1	0	0	1	X	X	X	X	f _{CLK} /2 ⁹	46.9 kHz
	1	0	1	0	X	X	X	X	f _{CLK} /2 ¹⁰	23.4 kHz
	1	0	1	1	X	X	X	X	f _{CLK} /2 ¹¹	11.7 kHz
	1	1	0	0	X	X	X	X	f _{CLK} /2 ¹²	5.86 kHz
	1	1	0	1	X	X	X	X	f _{CLK} /2 ¹³	2.93 kHz
	1	1	1	0	X	X	X	X	f _{CLK} /2 ¹⁴	1.46 kHz
	1	1	1	1	X	X	X	X	f _{CLK} /2 ¹⁵	732 Hz

注 要更改被选择为 f_{CLK} 的时钟（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）的运行（串行通道停止寄存器 m（STm）=000FH）后进行更改。

备注 1. X: 忽略

2. m: 单元号 (m = 0) n: 通道号 (n = 0) mn = 00

14.5.9 在 3 线串行 I/O（CSI00）通信过程中发生错误时的处理步骤

在 3 线串行 I/O（CSI00）通信过程中发生的错误时的处理步骤如图 14-75 所示。

图 14-75 发生溢出错误时的处理步骤

软件操作	硬件状态	备注
读串行数据寄存器 mn（SDRmn）。→	SSRmn 寄存器的 BFFmn 位为“0”并且通道 n 为可接收状态。	这是为了防止在错误处理的过程中结束下一次接收而发生溢出错误。
读串行状态寄存器 mn（SSRmn）。		判断错误的种类，读取值用于清除错误标志。
给串行标志清除触发寄存器 mn（SIRmn）写“1”。→	清除错误标志。	通过将 SSRmn 寄存器的读取值直接写到 SIRmn 寄存器，只能清除读操作时的错误。

备注 m：单元号（m=0） n：通道号（n=0） mn=00

14.6 UART (UART0 ~ UART3) 通信的运行

这是通过串行数据发送 (TxD) 和串行数据接收 (RxD) 共 2 条线进行异步通信的功能。使用这 2 条通信线, 按数据帧 (由起始位、数据、奇偶校验位和停止位构成) 与其他通信方进行异步 (使用内部波特率) 的数据发送和接收。能通过使用发送专用 (偶数通道) 和接收专用 (奇数通道) 共 2 个通道来实现全双工异步 UART 通信, 而且还能通过组合 UART0、定时器阵列单元 0 (通道 7) 和外部中断 (INTP0) 来支持 LIN-bus。

[数据的发送和接收]

- 7 位、8 位或者 9 位的数据长度注
- MSB/LSB 优先的选择
- 发送和接收数据的电平设定、反相的选择
- 奇偶校验位的附加、奇偶校验功能
- 停止位的附加、停止位检查功能

[中断功能]

- 传送结束中断、缓冲器空中断
- 帧错误、奇偶校验错误和溢出错误引起的错误中断

[错误检测标志]

- 帧错误、奇偶校验错误、溢出错误

UART0、UART2 接收支持 SNOOZE 模式。SNOOZE 模式功能是指: 如果在 STOP 模式的状态下检测到 RxD 的输入, 就不需要 CPU 运行而接收数据。只能对支持接收时的波特率调整功能的 UART0、UART2 进行设定。

UART0 (单元 0 的通道 0 和通道 1) 支持 LIN-bus。

[LIN-bus 功能]

• 唤醒信号的检测 • 间隔场 (BF) 的检测 • 同步场的测量、波特率的计算	}	使用外部中断 (INTP0) 和定时器阵列单元 0 (通道 7)。

注 只有 UART0、UART2 支持 9 位的数据长度。

UART0 使用 SAU0 的通道 0 和通道 1。

UART1 使用 SAU0 的通道 2 和通道 3。

UART2 使用 SAU1 的通道 0 和通道 1。

UART3 使用 SAU1 的通道 2 和通道 3。

单元	通道	用作 CSI	用作 UART	用作简易 I ² C
0	0	CSI00	UART0（支持 LIN-bus）	IIC00
	1	—		—
	2	—	UART1	IIC10
	3	—		—
1	0	—	UART2	—
	1	—		—
	2	—	UART3	—
	3	—		—

各通道任意选择一个功能使用，除了所选功能以外，其他功能不能运行。例如，当通过单元 0 的通道 0 和通道 1 使用 UART0 时，不能使用 CSI00。但是，在使用 UART0 的同时，能将不同的通道 2 和通道 3 用于 UART1 或者 IIC10。

注意 当用作 UART 时，只能发送方（偶数通道）或者接收方（奇数通道）用作 UART。

UART 有以下 4 种通信运行：

- UART 发送 （参照 14.6.1。）
- UART 接收 （参照 14.6.2。）
- LIN 发送（只限 UART2） （参照 14.7.1。）
- LIN 接收（只限 UART2） （参照 14.7.2。）

14.6.1 UART 发送

UART 发送是 RL78 微控制器将数据异步发送到其他设备的运行。

UART 使用的 2 个通道中的偶数通道用于 UART 发送。

UART	UART0	UART1	UART2	UART3
对象通道	SAU0 的通道 0	SAU0 的通道 2	SAU1 的通道 0	SAU1 的通道 2
使用的引脚	TxD0	TxD1	TxD2	TxD3
中断	INTST0	INTST1	INTST2	INTST3
	可选择发送结束中断（单次发送模式）或者缓冲器空中断（连续发送模式）。			
错误检测标志	无			
传送数据长度	7 位、8 位或者 9 位 ^{注 1}			
传送速率 ^{注 2}	Max. $f_{MCK}/6[\text{bps}]$ ($\text{SDRmn}[15:9] \geq 2$)、Min. $f_{CLK}/(2 \times 2^{15} \times 128)[\text{bps}]$			
数据相位	不反相输出（默认值：高电平）。 反相输出（默认值：低电平）。			
奇偶校验位	可选择以下内容： • 无奇偶校验位 • 附加零校验 • 附加偶校验 • 附加奇校验			
停止位	可选择以下内容： • 附加 1 位 • 附加 2 位			
数据方向	MSB 优先或者 LSB 优先			

注 1. 只有 UART0、UART2 支持 9 位的数据长度。

2. 必须在满足此条件并且满足电特性的外围功能特性（参照“第 32 章 电特性”）的范围内使用。

备注 1. f_{MCK} : 对象通道的运行时钟频率

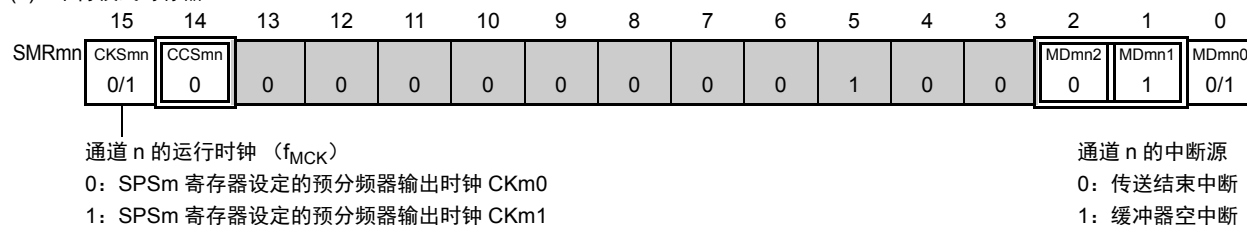
f_{CLK} : 系统时钟频率

2. m: 单元号 (m=0、1) n: 通道号 (n=0、2) mn=00、02、10、12

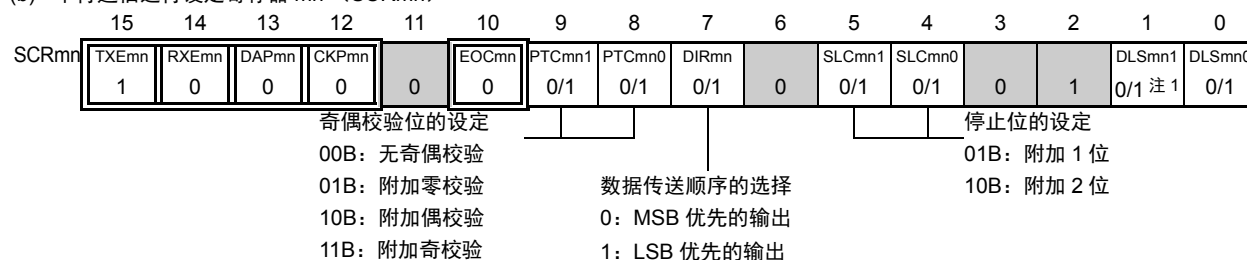
(1) 寄存器的设定

图 14-76 UART (UART0 ~ UART3) 的 UART 发送时的寄存器设定内容例子 (1/2)

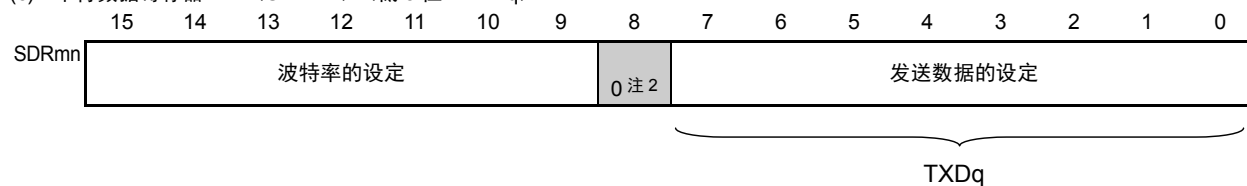
(a) 串行模式寄存器 mn (SMRmn)



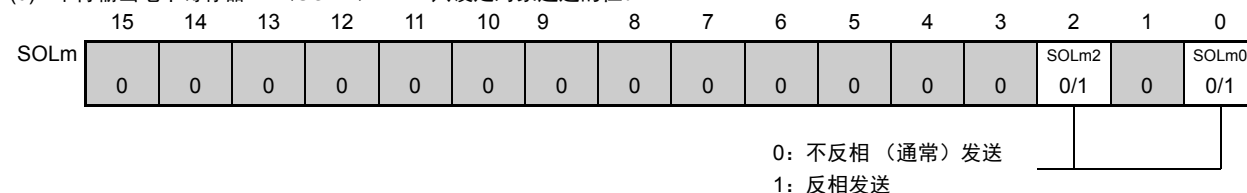
(b) 串行通信运行设定寄存器 mn (SCRmn)



(c) 串行数据寄存器 mn (SDRmn) (低 8 位: TXDq)



(d) 串行输出电平寄存器 m (SOLm) 只设定对象通道的位。



注 1. 只限 SCR00、SCR01、SCR10 和 SCR11 寄存器, 其他固定为“1”。

2. 当进行 9 位数据长度的通信时, SDRm0 寄存器的 bit0 ~ 8 为发送数据的设定区。只有 UART0、UART2 能进行 9 位数据长度的通信。

备注 1. m: 单元号 (m=0、1) n: 通道号 (n=0、2) q: UART 号 (q=0 ~ 3) mn=00、02、10、12

2. : 在 UART 发送模式中为固定设定。■: 不能设定 (设定初始值)。

0/1: 根据用户的用途置“0”或者“1”。

图 14-76 UART (UART0 ~ UART3) 的 UART 发送时的寄存器设定内容例子 (2/2)

(e) 串行输出寄存器 m (SOM) 只设定对象通道的位。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOM	0	0	0	0	1	1	1	CKOm0 注2 ×	0	0	0	0	1	SOM2 0/1 注1	1	SOM0 0/1 注1

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) 只将对象通道的位置“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) 只将对象通道的位置“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 1. 在开始发送前, 当对应通道的 SOLmn 位为“0”时, 必须置“1”; 当对应通道的 SOLmn 位为“1”时, 必须置“0”。在通信过程中, 值因通信数据而变。
2. 只限串行阵列单元 0。

备注 1. m: 单元号 (m=0、1) n: 通道号 (n=0、2) q: UART 号 (q=0 ~ 3) mn=00、02、10、12

2. ☐ : 在 UART 发送模式中为固定设定。 ☐ : 不能设定 (设定初始值)。
 × : 这是在此模式中不能使用的位 (在其他模式中也不使用的情况下, 设定初始值)。
 0/1 : 根据用户的用途置“0”或者“1”。

(2) 操作步骤

图 14-77 UART 发送的初始设定步骤

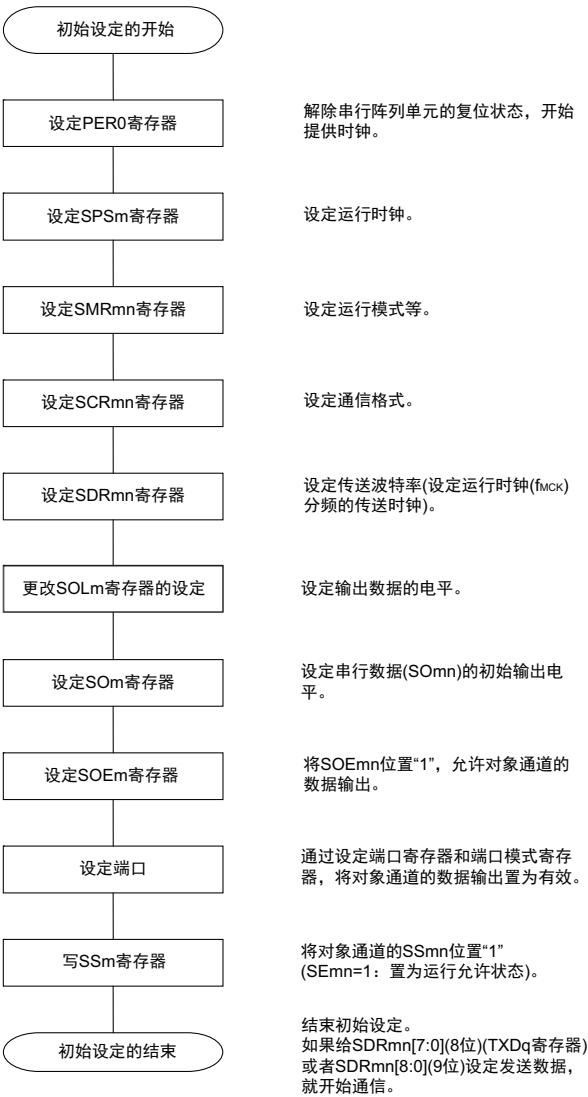


图 14-78 UART 发送的中止步骤

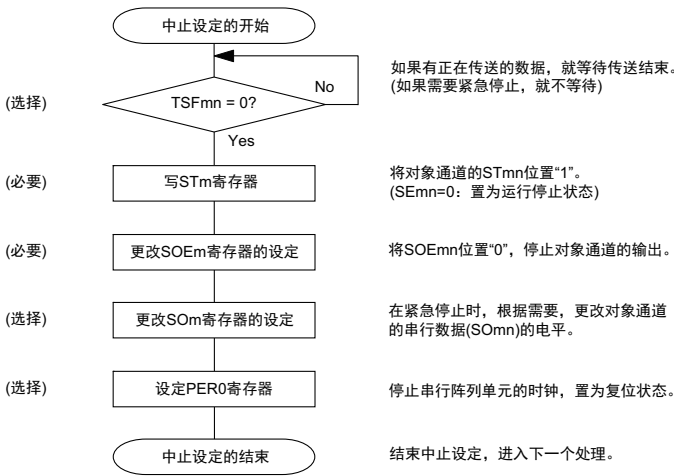
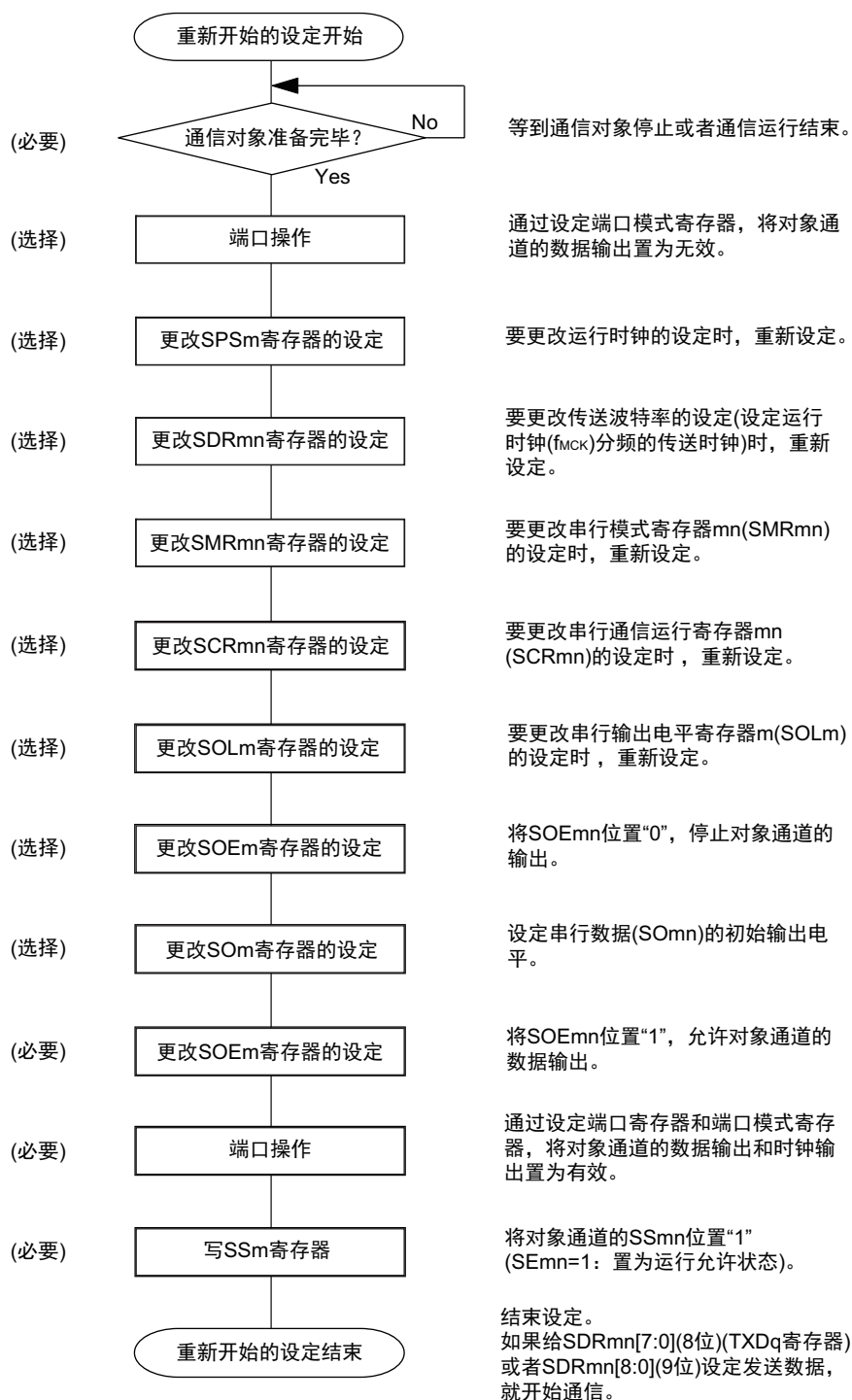


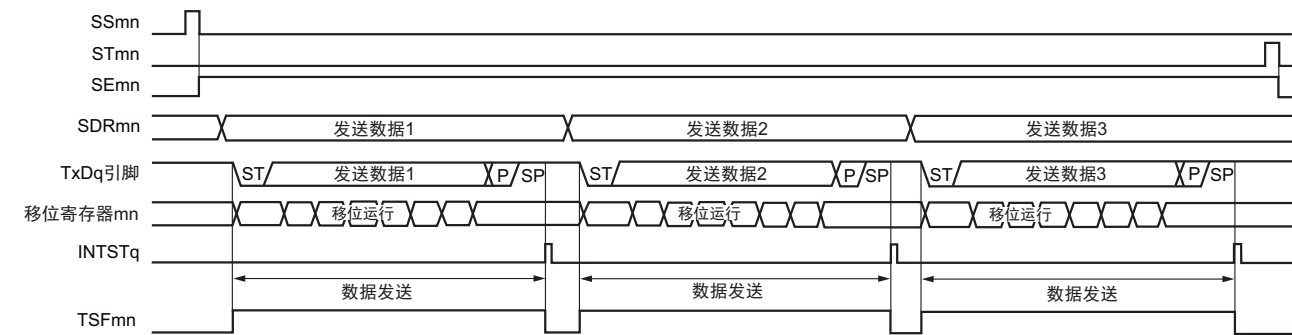
图 14-79 重新开始 UART 发送的设定步骤



备注 如果在中止设定中改写 PER0 来停止提供时钟, 就必须在等待通信对象的停止或者通信结束后进行初始设定而不是进行重新开始设定。

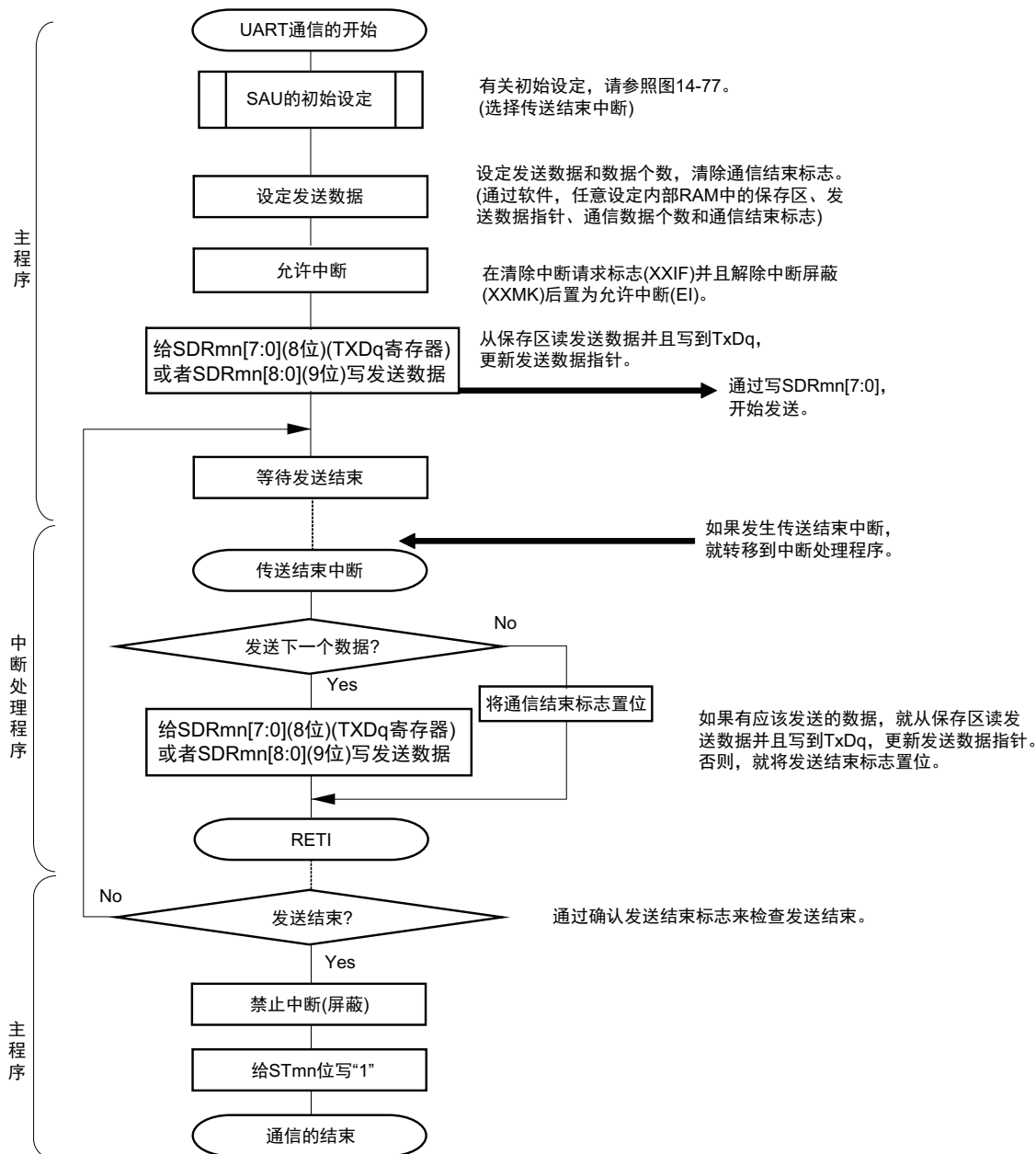
(3) 处理流程（单次发送模式）

图 14-80 UART 发送（单次发送模式）的时序图



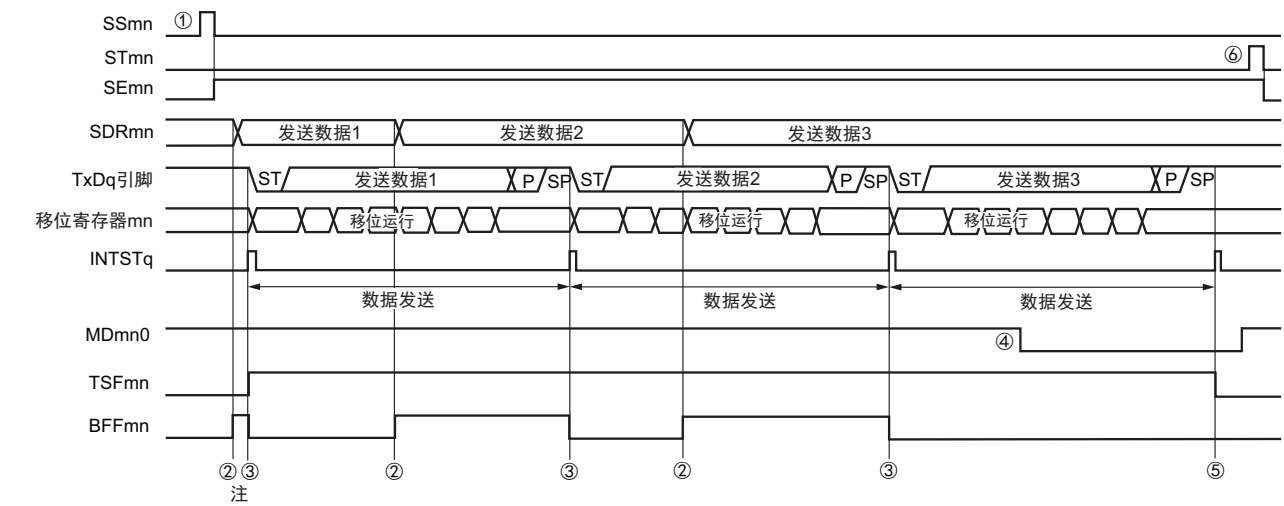
备注 m: 单元号 (m=0、1) n: 通道号 (n=0、2) q: UART 号 (q=0 ~ 3) mn=00、02、10、12

图 14-81 UART 发送（单次发送模式）的流程图



(4) 处理流程（连续发送模式）

图 14-82 UART 发送（连续发送模式）的时序图

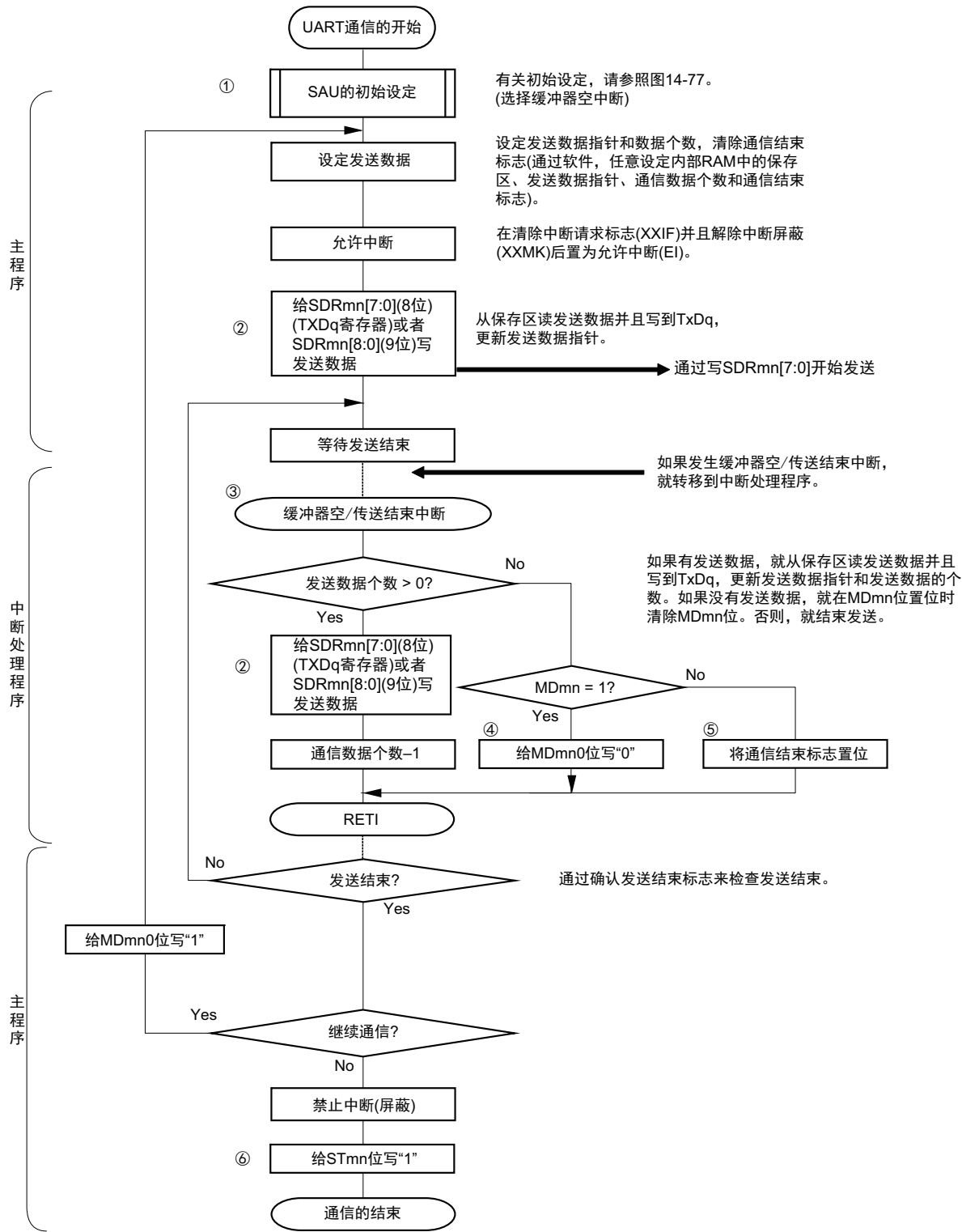


注 如果在串行状态寄存器 mn（SSRmn）的 BFFmn 位为“1”期间（有效数据保存在串行数据寄存器 mn（SDRmn）时）给 SDRmn 寄存器写发送数据，就重写发送数据。

注意 即使在运行中也能改写串行模式寄存器 mn（SMRmn）的 MDmn0 位。但是，为了能赶上最后发送数据的传送结束中断，必须在开始传送最后一位之前进行改写。

备注 m: 单元号（m=0、1） n: 通道号（n=0、2） q: UART 号（q=0～3） mn=00、02、10、12

图 14-83 UART 发送（连续发送模式）的流程图



备注 图中的①～⑥对应“图 14-82 UART 发送（连续发送模式）的时序图”中的①～⑥。

14.6.2 UART 接收

UART 接收是 RL78 微控制器从其他设备异步接收数据的运行。

UART 使用的 2 个通道中的奇数通道用于 UART 接收。但是，需要设定奇数通道和偶数通道的 SMR 寄存器。

UART	UART0	UART1	UART2	UART3
对象通道	SAU0 的通道 1	SAU0 的通道 3	SAU1 的通道 1	SAU1 的通道 3
使用的引脚	RxD0	RxD1	RxD2	RxD3
中断	INTSR0	INTSR1	INTSR2	INTSR3
	只限于传送结束中断（禁止设定缓冲器空中断）。			
错误中断	INTSRE0	INTSRE1	INTSRE2	INTSRE3
错误检测标志	• 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 位、8 位或者 9 位注 1			
传送速率注 2	Max. $f_{MCK}/6[\text{bps}]$ ($\text{SDRmn}[15:9] \geq 2$)、Min. $f_{CLK}/(2 \times 2^{15} \times 128)$ [bps]			
数据相位	不反相输出（默认值：高电平）。 反相输出（默认值：低电平）。			
奇偶校验位	可选择以下内容： • 无奇偶校验位（无奇偶校验） • 不判断奇偶校验（零校验） • 偶校验 • 奇校验			
停止位	附加 1 位。			
数据方向	MSB 优先或者 LSB 优先			

注 1. 只有 UART0、UART2 支持 9 位的数据长度。

2. 必须在满足此条件并且满足电特性的外围功能特性（参照“第 32 章 电特性”）的范围内使用。

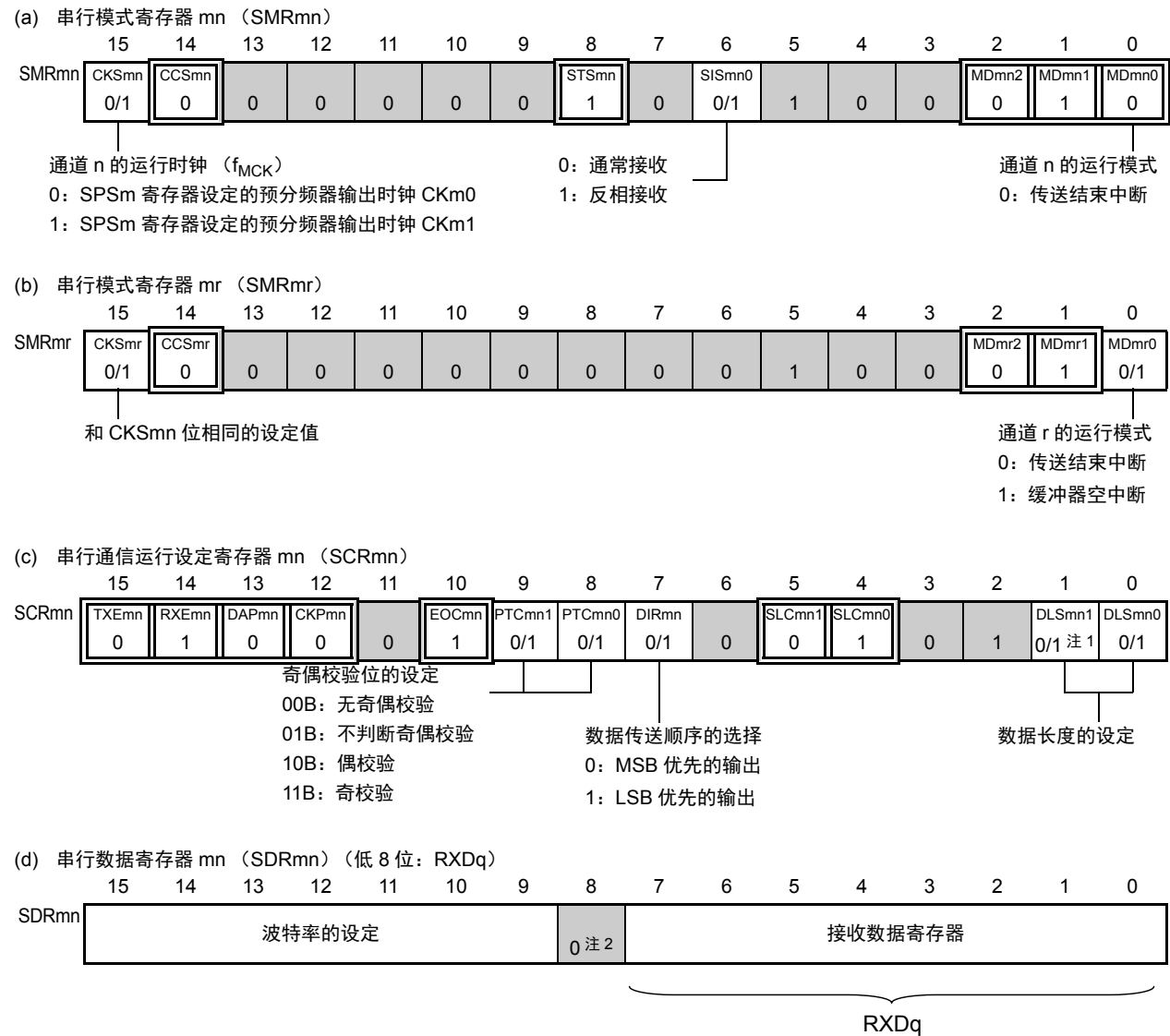
备注 1. f_{MCK} ：对象通道的运行时钟频率

f_{CLK} ：系统时钟频率

2. m：单元号（m=0、1） n：通道号（n=1、3） mn=01、03、11、13

(1) 寄存器的设定

图 14-84 UART（UART0～UART3）的 UART 接收时的寄存器设定内容例子（1/2）



注 1. 只限 SCR00、SCR01、SCR10 和 SCR11 寄存器，其他固定为“1”。
2. 当进行 9 位数据长度的通信时，SDRm1 寄存器的 bit0～8 为发送数据的设定区。只有 UART0、UART2 能进行 9 位数据长度的通信。

注意 在 UART 接收时，还必须将与通道 n 成对的通道 r 的 SMRmr 寄存器设定为 UART 发送模式。

备注 1. m: 单元号（m=0、1） n: 通道号（n=1、3） mn=01、03、11、13
r: 通道号（r=n-1） q: UART 号（q=0～3）
2. □: 在 UART 接收模式中为固定设定。■: 不能设定（设定初始值）。
0/1: 根据用户的用途置“0”或者“1”。

图 14-84 UART (UART0 ~ UART3) 的 UART 接收时的寄存器设定内容例子 (2/2)

(e) 串行输出寄存器 m (SOM) 在此模式中不使用。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOM	0	0	0	0	1	1	1	CKOm0 注 ×	0	0	0	0	1	SOM2 ×	1	SOM0 ×

(f) 串行输出允许寄存器 m (SOEm) 在此模式中不使用。

	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) 只将对象通道的位置“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 只限串行阵列单元 0。

在 UART 接收时, 还必须将与通道 n 成对的通道 r 的 SMRmr 寄存器设定为 UART 发送模式。

备注 1. m: 单元号 (m=0、1) n: 通道号 (n=1、3) mn=01、03、11、13

r: 通道号 (r=0~1) q: UART 号 (q=0~3)

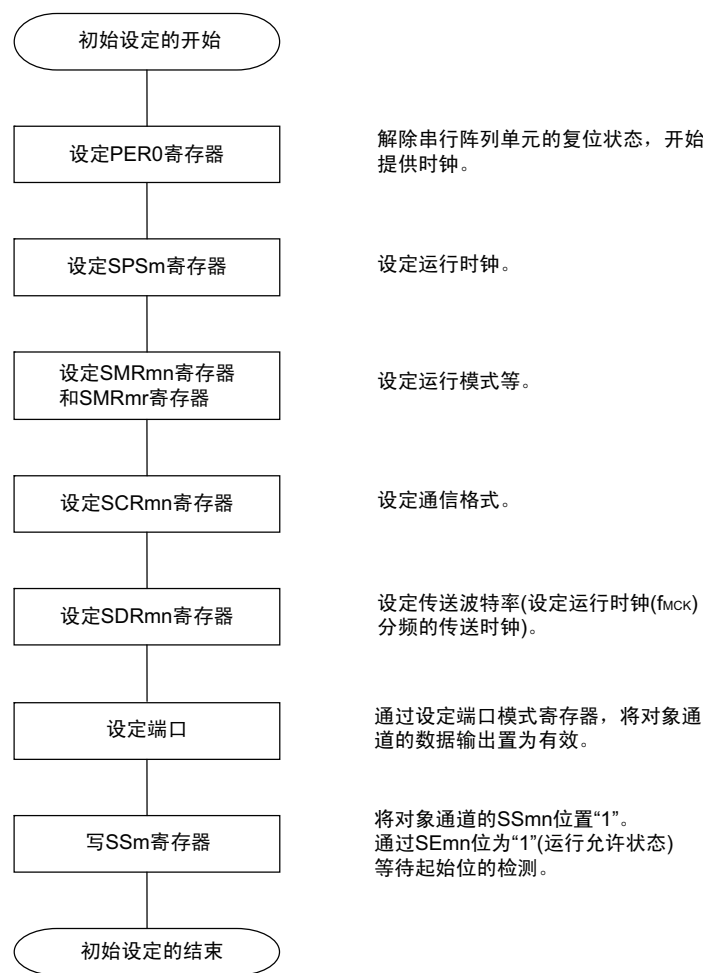
2.  : 不能设定 (设定初始值)。

× : 这是在此模式中不能使用的位 (在其他模式中也不使用的情况下, 设定初始值)。

0/1 : 根据用户的用途置“0”或者“1”。

(2) 操作步骤

图 14-85 UART 接收的初始设定步骤



注意 必须在将 SCRmn 寄存器的 RXEmn 位置“1”后至少经过 4 个 f_{MCK} 时钟，然后将 SSmn 位置“1”。

图 14-86 UART 接收的中止步骤

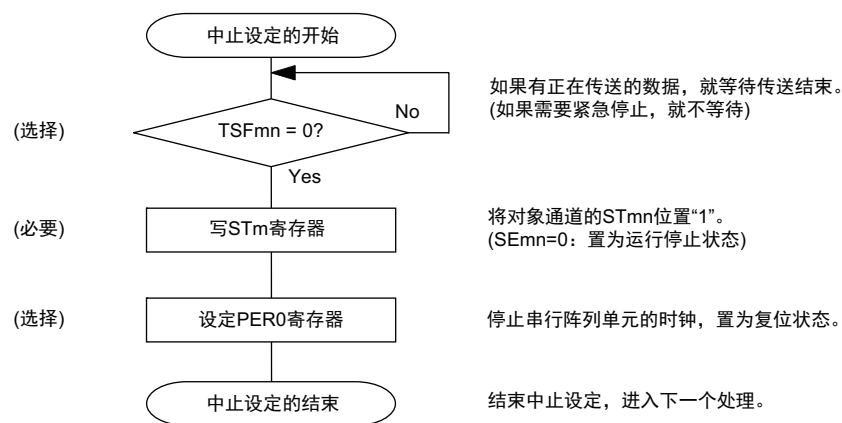
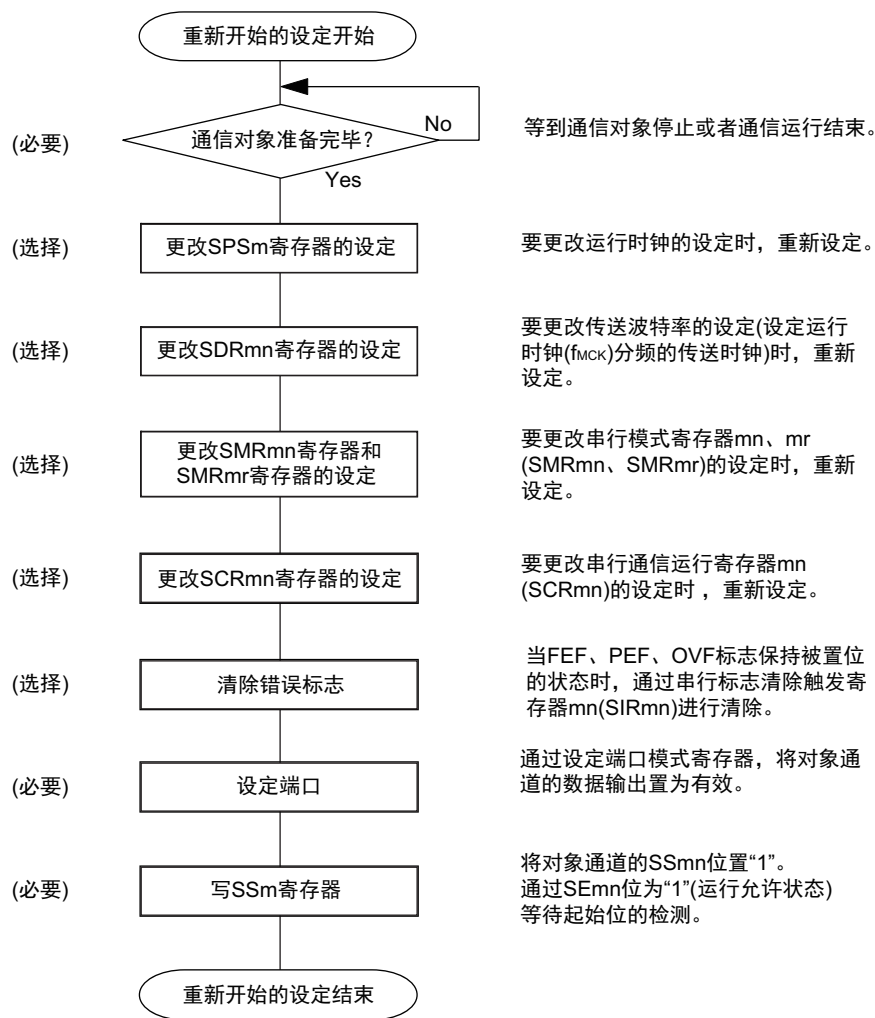


图 14-87 重新开始 UART 接收的设定步骤

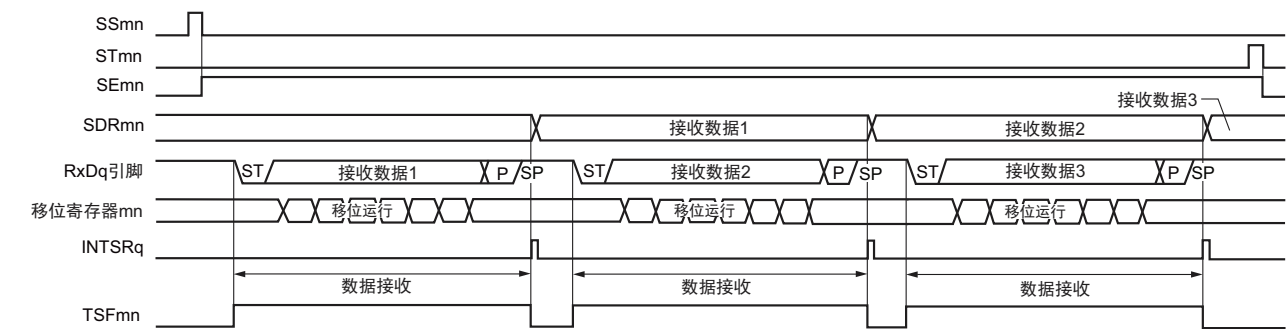


注意 必须在将 SCRmn 寄存器的 RXEmn 位置“1”后至少经过 4 个 f_{MCK} 时钟，然后将 SSmn 位置“1”。

备注 如果在中止设定中改写 PER0 来停止提供时钟，就必须在等待通信对象的停止或者通信结束后进行初始设定而不是进行重新开始设定。

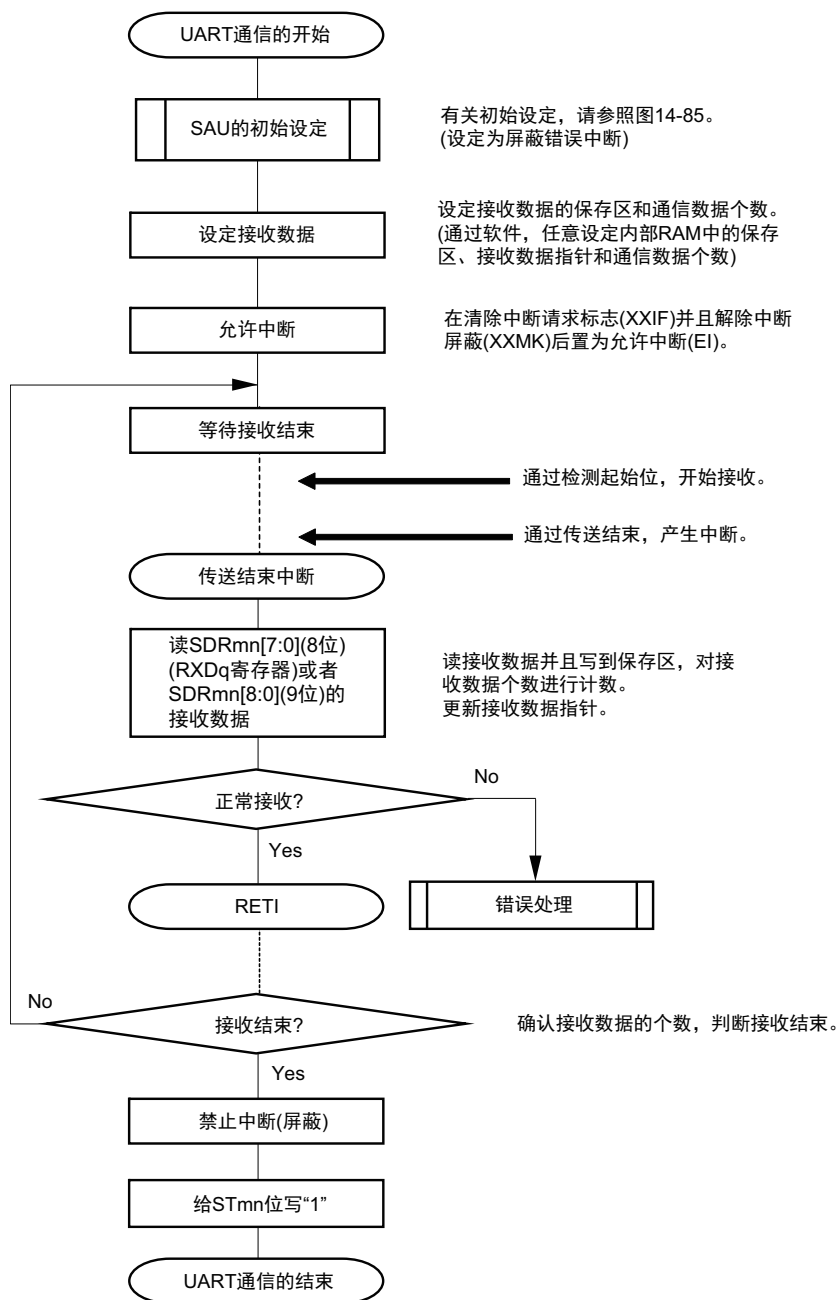
(3) 处理流程

图 14-88 UART 接收的时序图



备注 m: 单元号 (m=0、1) n: 通道号 (n=1、3) mn=01、03、11、13
r: 通道号 (r=n-1) q: UART 号 (q=0 ~ 3)

图 14-89 UART 接收的流程图



14.6.3 SNOOZE 模式功能

这是在 STOP 模式中通过检测 RxDq 引脚的输入开始 UART 接收的模式。通常，在 STOP 模式中 UART 停止通信，但是通过使用 SNOOZE 模式功能，能在 CPU 不工作的状态下进行 UART 的接收。

只有 UART0、UART2 能设定 SNOOZE 模式。

如果要在 SNOOZE 模式中使用 UARTq，必须在即将要转移到 STOP 模式前进行以下设定（参照图“图 14-92 SNOOZE 模式运行（EOCm1=0、SSECm=0/1 或者 EOCm1=1、SSECm=0）的流程图”、“图 14-94 SNOOZE 模式运行（EOCm1=1、SSECm=1）的流程图”）。

- 在 SNOOZE 模式中，需要将 UART 接收波特率的设定更改为与通常运行时不同的值。参照表 14-3 设定 SPSm 寄存器和 SDRmn 寄存器的 bit15~9。
- 设定 EOCmn 位和 SSECm 位。能在发生通信错误时设定允许或者禁止发生错误中断（INTSRE0）。
- 在即将要转移到 STOP 模式前将串行待机控制寄存器 m（SSCm）的 SWCm 位置“1”。初始设定结束后，将串行通道开始寄存器 m（SSm）的 SSm1 位置“1”。

在转移到 STOP 模式后，如果检测到 RxDq 的边沿（输入起始位），就开始 UART 接收。

注意 1. 只有在选择高速内部振荡器时钟（ f_{IH} ）作为 f_{CLK} 时才能使用 SNOOZE 模式功能。

2. SNOOZE 模式中的传送速率只能为 4800bps。

3. 在 SWCm 位为“1”的情况下，只能在 STOP 模式中开始接收时使用 UARTq。通过同时使用 SNOOZE 模式功能和中断，在以下非 STOP 模式的状态下开始接收时，有可能无法正常接收数据，并且有可能发生帧错误或者奇偶校验错误。

- 将 SWCm 位置“1”后，转移到 STOP 模式前开始接收时
- 在其他 SNOOZE 模式中开始接收时
- 通过中断等从 STOP 模式返回到通常运行后，将 SWCm 位置“0”前开始接收时

4. 在 SSECm 位为“1”的情况下发生奇偶校验错误、帧错误或者溢出错误时，PEFmn、FEFmn、OVFmn 标志不被置位，而且也不发生错误中断（INTSREq）。因此如果要在 SSECm 位为“1”时使用 SNOOZE 模式功能，必须在将 SWC0 置“1”前，清除 PEFmn、FEFmn、OVFmn 标志，并且读 SDRm1 寄存器的 bit7~0（RxDq）。

表 14-3 SNOOZE 模式中的 UART 接收波特率的设定

高速内部振荡器 (f_{IH})	SNOOZE 模式中的 UART 接收波特率			
	波特率 4800bps			
	运行时钟 (f_{MCK})	SDRmn[15:9]	最大容许值	最小容许值
24MHz±1.0% 注	$f_{CLK}/2^5$	79	1.60%	-2.18%
16MHz±1.0% 注	$f_{CLK}/2^4$	105	2.27%	-1.53%
12MHz±1.0% 注	$f_{CLK}/2^4$	79	1.60%	-2.19%
8MHz±1.0% 注	$f_{CLK}/2^3$	105	2.27%	-1.53%
6MHz±1.0% 注	$f_{CLK}/2^3$	79	1.60%	-2.19%
4MHz±1.0% 注	$f_{CLK}/2^2$	105	2.27%	-1.53%
3MHz±1.0% 注	$f_{CLK}/2^2$	79	1.60%	-2.19%
2MHz±1.0% 注	$f_{CLK}/2$	105	2.27%	-1.54%
1MHz±1.0% 注	f_{CLK}	105	2.27%	-1.57%

注 在高速内部振荡器时钟频率的精度为 ±1.5%、±2.0% 时，容许范围将会变小，如下所示。

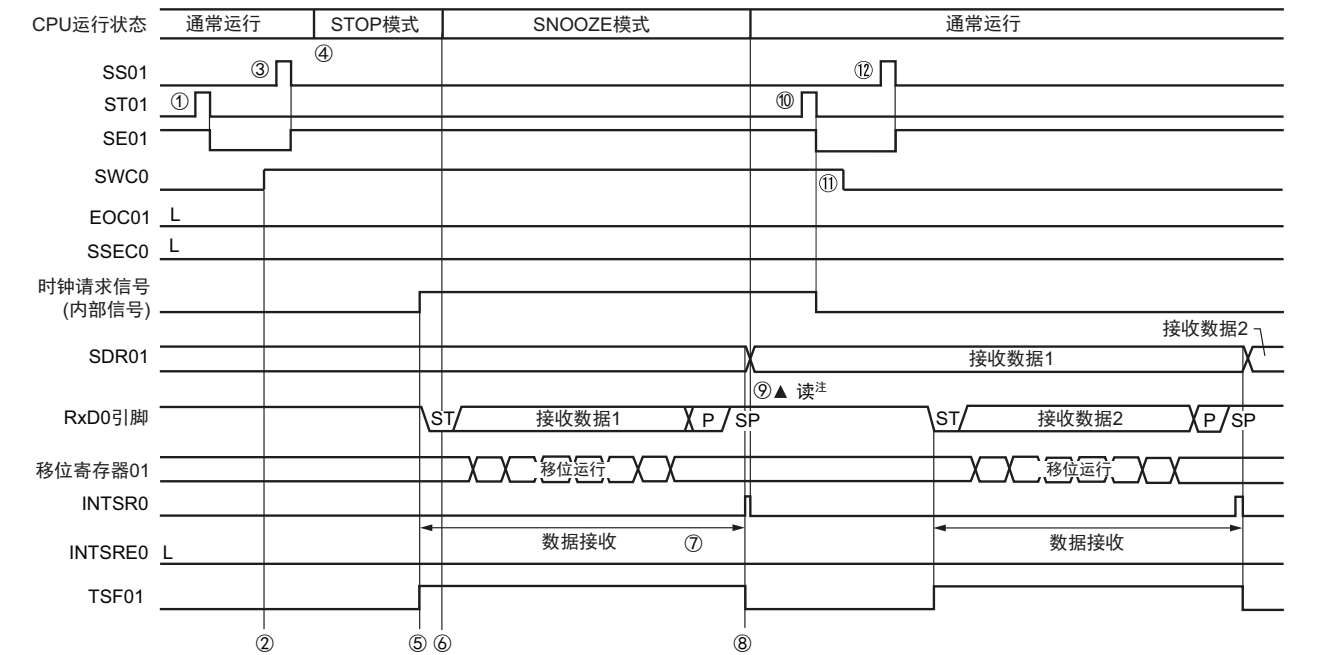
- 在 f_{IH} ±1.5% 时，上表中的最大容许值必须减 0.5%，最小容许值必须加 0.5%。
- 在 f_{IH} ±2.0% 时，上表中的最大容许值必须减 1.0%，最小容许值必须加 1.0%。

备注 最大容许值和最小容许值是 UART 接收时的波特率容许值。
发送方的波特率必须设定在此范围内。

(1) SNOOZE 模式运行（EOCm1=0、SSECm=0/1）

因为 EOCm1=0，所以与 SSECm 位的设定无关，即使发生通信错误，也不发生错误中断（INTSREq）。但是发生传送结束中断（INTSRq）。

图 14-90 SNOOZE 模式运行（EOCm1=0、SSECm=0/1）的时序图



注 必须在 SWCm 位为“1”的状态下读接收数据。

注意 在向 SNOOZE 模式转移前或者在 SNOOZE 模式中接收结束后，必须将 STm1 位置“1”（清除 SEm1 位并且停止运行），而且还必须在接收结束后清除 SWCm 位（解除 SNOOZE 模式）。

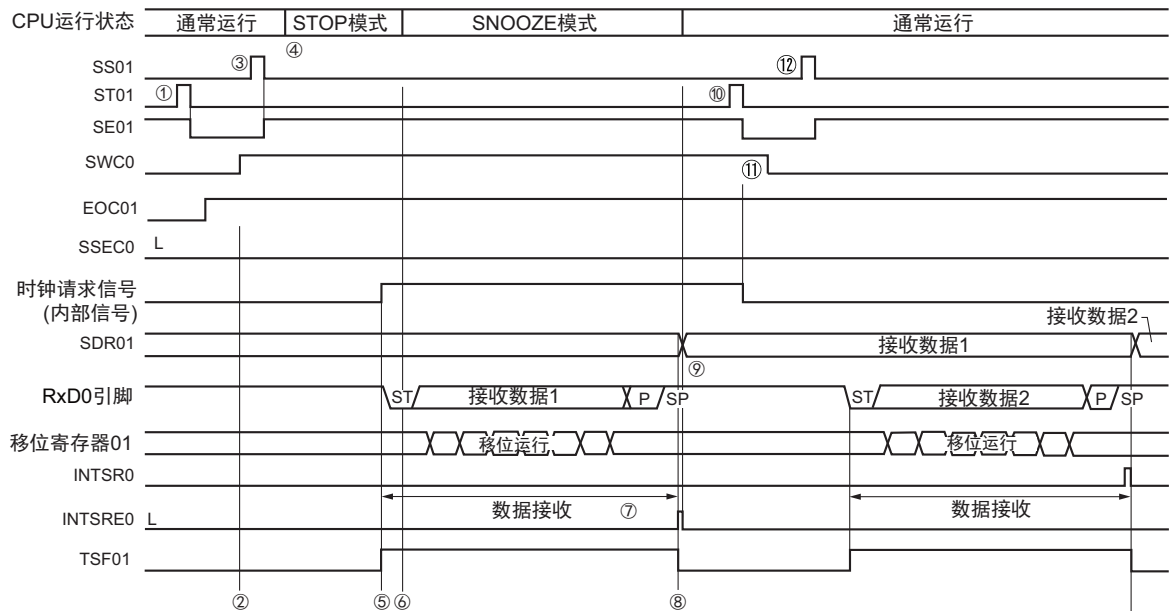
备注 1. 图中的①～⑫对应“图 14-92 SNOOZE 模式运行（EOCm1=0、SSECm=0/1 或者 EOCm1=1、SSECm=0）的流程图”中的①～⑫。

2. m=0、1，q=0、2

(2) SNOOZE 模式运行（EOCm1=1、SSECm=0：允许发生错误中断（INTSREq））

因为 EOCm1=1、SSECm=0，所以在发生通信错误时发生错误中断（INTSREq）。

图 14-91 SNOOZE 模式运行（EOCm1=1、SSECm=0）的时序图

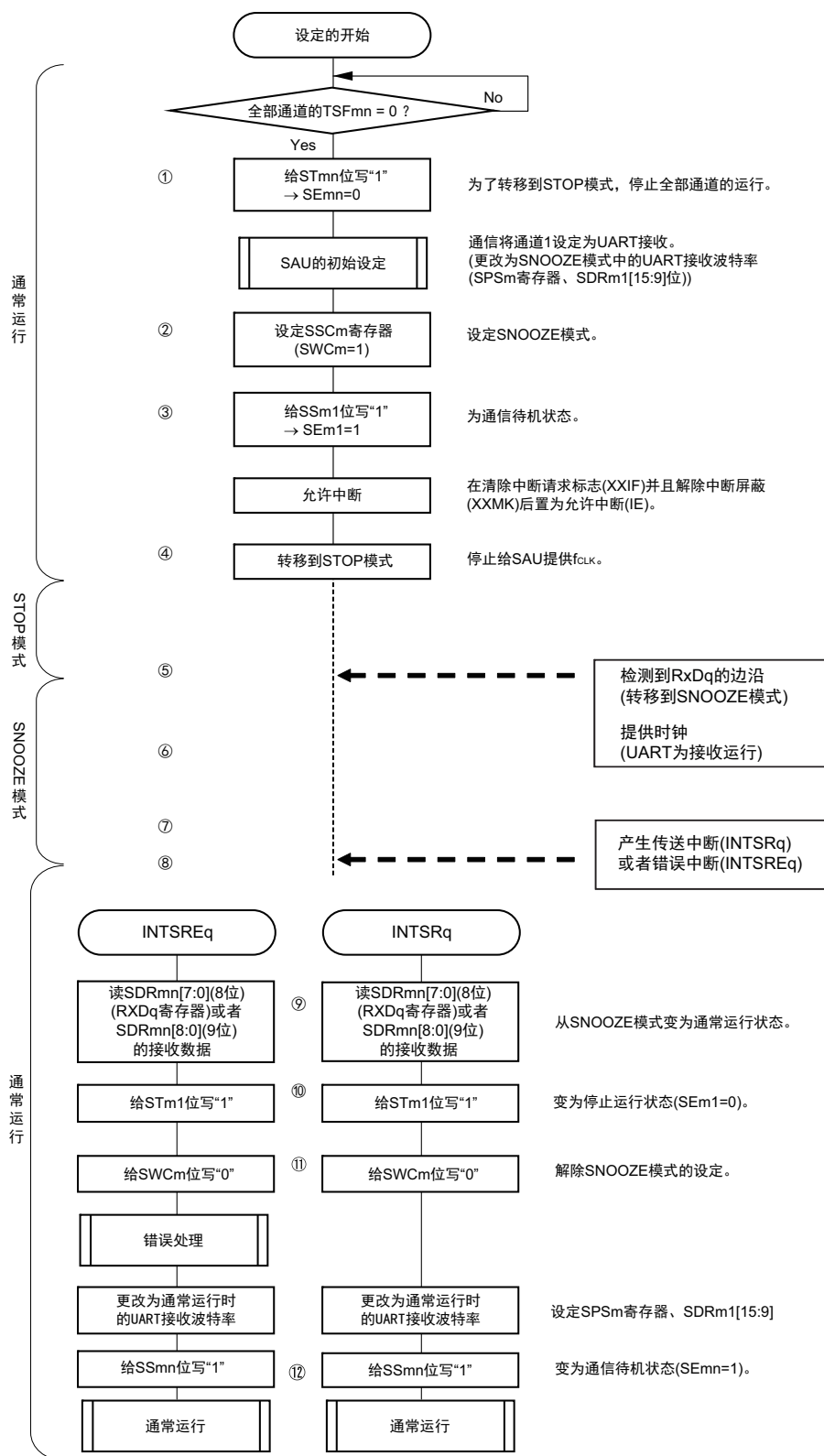


注意 在向 SNOOZE 模式转移前或者在 SNOOZE 模式中接收结束后，必须将 STm1 位置“1”（清除 SEm1 位并且停止运行），而且还必须在接收结束后清除 SWCm 位（解除 SNOOZE 模式）。

备注 1. 图中的①～⑫对应“图 14-92 SNOOZE 模式运行（EOCm1=0、SSECm=0/1 或者 EOCm1=1、SSECm=0）的流程图”中的①～⑫。

2. m=0、1，q=0、2

图 14-92 SNOOZE 模式运行 (EOCm1=0、SSECm=0/1 或者 EOCm1=1、SSECm=0) 的流程图



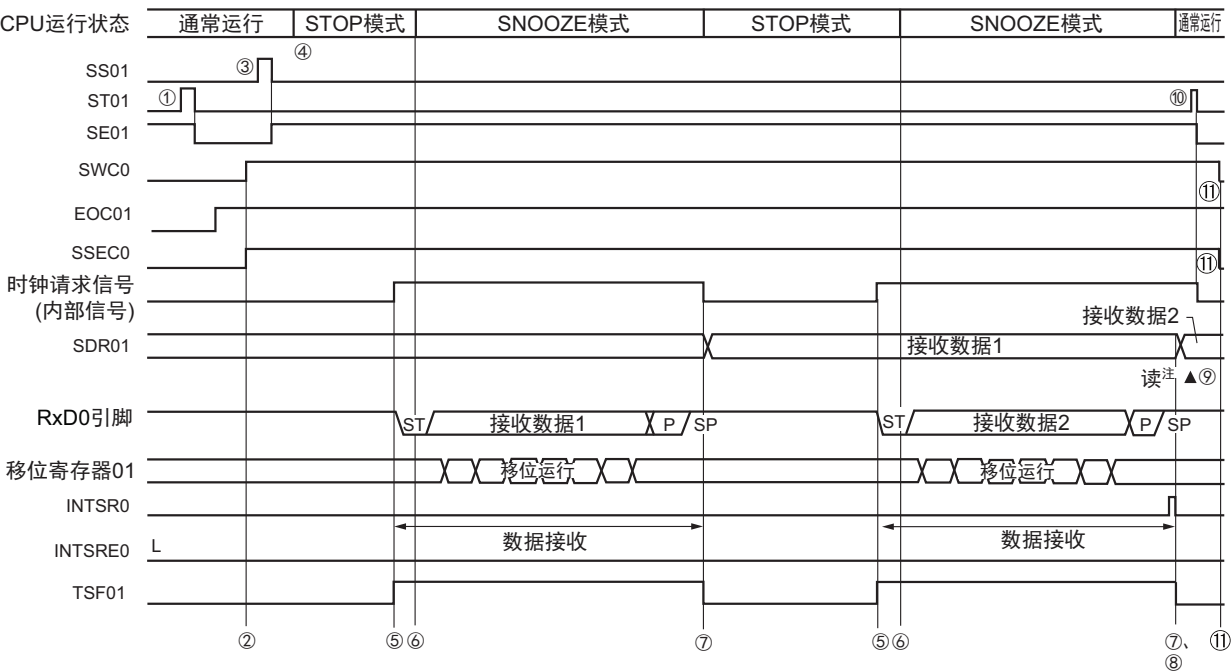
备注 1. 图中的①~⑫对应“图 14-90 SNOOZE 模式运行 (EOCm1=0、SSECm=0/1) 的时序图”和“图 14-91 SNOOZE 模式运行 (EOCm1=1、SSECm=0) 的时序图”中的①~⑫。

2. m=0、1，q=0、2

(3) SNOOZE 模式运行（EOCm1=1、SSECm=1：禁止发生错误中断（INTSREq））

因为 EOCm1=1、SSECm=1，所以在发生通信错误时不发生错误中断（INTSREq）。

图 14-93 SNOOZE 模式运行（EOCm1=1、SSECm=1）的时序图



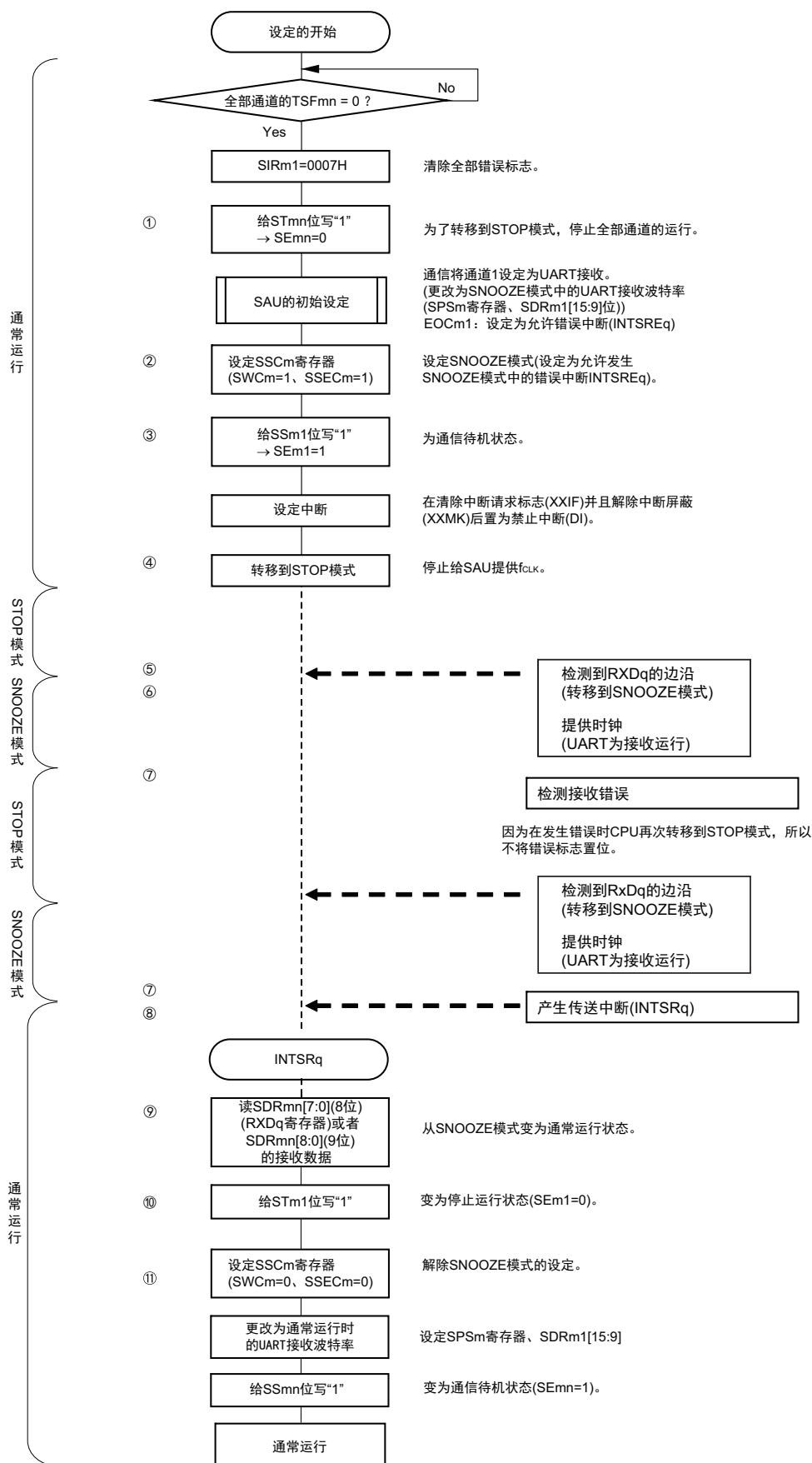
注 必须在 SWCm 位为“1”的状态下读接收数据。

- 注意 1. 在向 SNOOZE 模式转移前或者在 SNOOZE 模式中接收结束后，必须将 STm1 位置“1”（清除 SEm1 位并且停止运行），而且还必须在接收结束后清除 SWCm 位（解除 SNOOZE 模式）。
2. 在 SSECm 位为“1”时，如果发生奇偶校验错误、帧错误、溢出错误，PEFm1、FEFm1、OVFm1 标志就不置位，也不发生错误中断（INTSREq）。因此，在 SSECm=1 时使用 SNOOZE 模式功能时，必须在将 SWCm 位置“1”前，清除 PEFm1、FEFm1、OVFm1 标志，并且读 SDRm1[7:0]（8 位）（RxDq 寄存器）或者 SDRm1[8:0]（9 位）。

备注 1. 图中的①～⑪对应“图 14-94 SNOOZE 模式运行（EOCm1=1、SSECm=1）的流程图”中的①～⑪。

2. m=0、1，q=0、2

图 14-94 SNOOZE 模式运行 (EOCm1=1、SSECm=1) 的流程图



注意 在 SSECm 位为“1”时，如果发生奇偶校验错误、帧错误、溢出错误，PEFm1、FEFm1、OVFm1 标志就不置位，也不发生错误中断（INTSREq）。因此，在 SSECm=1 时使用 SNOOZE 模式功能时，必须在将 SWCm 位置“1”前，清除 PEFm1、FEFm1、OVFm1 标志，并且读 SDRm1[7:0]（8 位）（RxDq 寄存器）或者 SDRm1[8:0]（9 位）。

备注 1. 图中的①～⑪对应“图 14-93 SNOOZE 模式运行（EOCm1=1、SSECm=1）的时序图”中的①～⑪。

2. m=0、1，q=0、2

14.6.4 波特率的计算

(1) 波特率的计算式

UART (UART0 ~ UART3) 通信的波特率能用以下计算式进行计算:

$$(\text{波特率}) = \{ \text{对象通道的运行时钟 (f}_{\text{MCK}}) \text{ 频率} \} \div (\text{SDRmn}[15:9] + 1) \div 2 [\text{bps}]$$

注意 禁止设定串行数据寄存器 mn (SDRmn) 的 SDRmn[15:9]=(0000000B、0000001B)。

备注 1. 因为在使用 UART 时 SDRmn[15:9] 的值为 SDRmn 寄存器的 bit15 ~ 9 的值 (0000010B ~ 1111111B)，所以为 2 ~ 127。

2. m: 单元号 (m=0、1) n: 通道号 (n=0 ~ 3) mn=00 ~ 03、10 ~ 13

运行时钟 (f_{MCK}) 取决于串行时钟选择寄存器 m (SPSm) 和串行模式寄存器 mn (SMRmn) 的 bit15 (CKSmn 位)。

表 14-4 UART 运行时钟的选择

SMRmn 寄存器	SPSm 寄存器								运行时钟 (f_{CLK}) 注	
CKSmn	PRS m13	PRS m12	PRS m11	PRS m10	PRS m03	PRS m02	PRS m01	PRS m00	$f_{CLK}=24\text{ MHz}$ 运行时	
0	X	X	X	X	0	0	0	0	f_{CLK}	24 MHz
	X	X	X	X	0	0	0	1	$f_{CLK}/2$	12 MHz
	X	X	X	X	0	0	1	0	$f_{CLK}/2^2$	6 MHz
	X	X	X	X	0	0	1	1	$f_{CLK}/2^3$	3 MHz
	X	X	X	X	0	1	0	0	$f_{CLK}/2^4$	1.5 MHz
	X	X	X	X	0	1	0	1	$f_{CLK}/2^5$	750 kHz
	X	X	X	X	0	1	1	0	$f_{CLK}/2^6$	375 kHz
	X	X	X	X	0	1	1	1	$f_{CLK}/2^7$	187.5 kHz
	X	X	X	X	1	0	0	0	$f_{CLK}/2^8$	93.8 kHz
	X	X	X	X	1	0	0	1	$f_{CLK}/2^9$	46.9 kHz
	X	X	X	X	1	0	1	0	$f_{CLK}/2^{10}$	23.4 kHz
	X	X	X	X	1	0	1	1	$f_{CLK}/2^{11}$	11.7 kHz
	X	X	X	X	1	1	0	0	$f_{CLK}/2^{12}$	5.86 kHz
	X	X	X	X	1	1	0	1	$f_{CLK}/2^{13}$	2.93 kHz
	X	X	X	X	1	1	1	0	$f_{CLK}/2^{14}$	1.46 kHz
	X	X	X	X	1	1	1	1	$f_{CLK}/2^{15}$	732 Hz
1	0	0	0	0	X	X	X	X	f_{CLK}	24 MHz
	0	0	0	1	X	X	X	X	$f_{CLK}/2$	12 MHz
	0	0	1	0	X	X	X	X	$f_{CLK}/2^2$	6 MHz
	0	0	1	1	X	X	X	X	$f_{CLK}/2^3$	3 MHz
	0	1	0	0	X	X	X	X	$f_{CLK}/2^4$	1.5 MHz
	0	1	0	1	X	X	X	X	$f_{CLK}/2^5$	750 kHz
	0	1	1	0	X	X	X	X	$f_{CLK}/2^6$	375 kHz
	0	1	1	1	X	X	X	X	$f_{CLK}/2^7$	187.5 kHz
	1	0	0	0	X	X	X	X	$f_{CLK}/2^8$	93.8 kHz
	1	0	0	1	X	X	X	X	$f_{CLK}/2^9$	46.9 kHz
	1	0	1	0	X	X	X	X	$f_{CLK}/2^{10}$	23.4 kHz
	1	0	1	1	X	X	X	X	$f_{CLK}/2^{11}$	11.7 kHz
	1	1	0	0	X	X	X	X	$f_{CLK}/2^{12}$	5.86 kHz
	1	1	0	1	X	X	X	X	$f_{CLK}/2^{13}$	2.93 kHz
	1	1	1	0	X	X	X	X	$f_{CLK}/2^{14}$	1.46 kHz
	1	1	1	1	X	X	X	X	$f_{CLK}/2^{15}$	732 Hz

注 只有在停止串行阵列单元 (SAU) 的运行 (串行通道停止寄存器 m (STm) = 000FH) 后才能改 f_{CLK} 的选择时钟 (更改系统时钟控制寄存器 (CKC) 的值)。

备注 1. X: 忽略

2. m: 单元号 (m = 0、1)、n: 通道号 (n = 0 ~ 3)、mn = 00 ~ 03、10 ~ 13

(2) 发送时的波特率误差

UART (UART0 ~ UART3) 通信发送时的波特率误差能用以下计算式进行计算, 必须将发送方的波特率设定在接收方的波特率容许范围内。

$$(\text{波特率误差}) = (\text{波特率的计算值}) \div (\text{目标波特率的值}) \times 100 - 100 [\%]$$

$f_{\text{CLK}}=24\text{MHz}$ 时的 UART 波特率的设定例子如下所示。

UART 波特率 (目标波特率)	$f_{\text{CLK}}=24\text{MHz}$			
	运行时钟 (f_{CLK})	SDRmn[15:9]	波特率的计算值	与目标波特率的误差
300bps	$f_{\text{CLK}}/2^9$	77	300.48bps	+0.16%
600bps	$f_{\text{CLK}}/2^8$	77	600.96bps	+0.16%
1200bps	$f_{\text{CLK}}/2^7$	77	1201.92bps	+0.16%
2400bps	$f_{\text{CLK}}/2^6$	77	2403.85bps	+0.16%
4800bps	$f_{\text{CLK}}/2^5$	77	4807.69bps	+0.16%
9600bps	$f_{\text{CLK}}/2^4$	77	9615.38bps	+0.16%
19200bps	$f_{\text{CLK}}/2^3$	77	19230.8bps	+0.16%
31250bps	$f_{\text{CLK}}/2^3$	47	31250.0bps	$\pm 0.0\%$
38400bps	$f_{\text{CLK}}/2^2$	77	38461.5bps	+0.16%
76800bps	$f_{\text{CLK}}/2$	77	76923.1bps	+0.16%
153600bps	f_{CLK}	77	153846bps	+0.16%
312500bps	f_{CLK}	37	315789bps	+1.05%

备注 m: 单元号 (m=0、1) n: 通道号 (n=0、2) mn=00、02、10、12

(3) 接收时的波特率容许范围

UART (UART0 ~ UART3) 通信接收时的波特率容许范围能用以下计算式进行计算，必须将发送方的波特率设定在接收方的波特率容许范围内。

$$(\text{可接收的最大波特率}) = \frac{2 \times k \times \text{Nfr}}{2 \times k \times \text{Nfr} - k + 2} \times \text{Brate}$$

$$(\text{可接收的最小波特率}) = \frac{2 \times k \times (\text{Nfr} - 1)}{2 \times k \times \text{Nfr} - k - 2} \times \text{Brate}$$

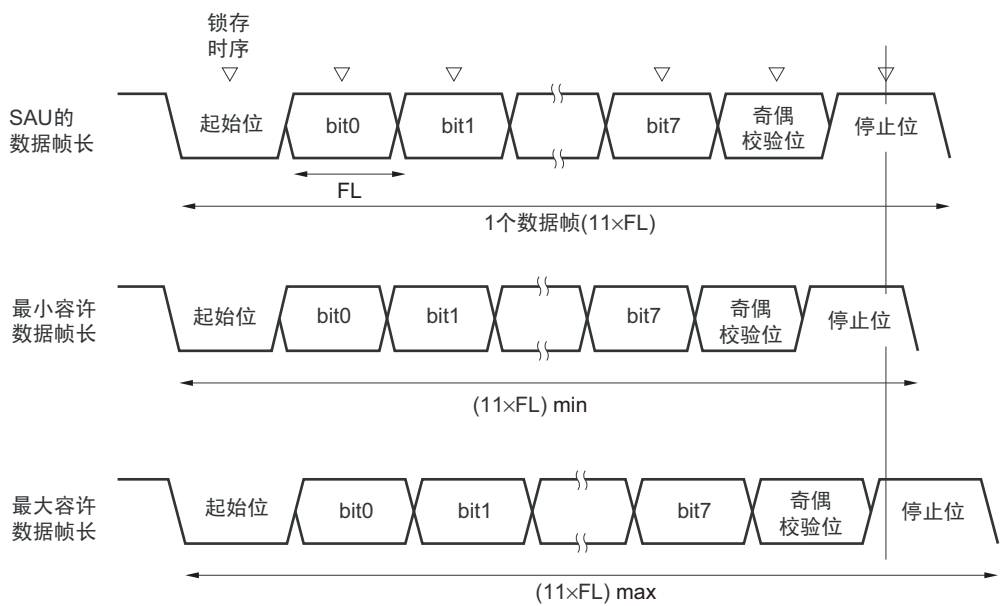
Brate: 接收方的波特率的计算值 (参照“14.6.4(1) 波特率的计算式”)

K: SDRmn[15:9]+1

Nfr: 1 个数据的数据帧长 [位]
=(起始位) + (数据长度) + (奇偶校验位) + (停止位)

备注 m: 单元号 (m=0、1) n: 通道号 (n=1、3) mn=01、03、11、13

图 14-95 接收时的波特率容许范围 (1 个数据的数据帧长 =11 位的情况)



如图 14-95 所示，在检测到起始位后，接收数据的锁存时序取决于串行数据寄存器 mn (SDRmn) 的 bit15 ~ 9 设定的分频比。如果最后的数据 (停止位) 能赶上此锁存时序，就能正常接收。

14.6.5 UART (UART0 ~ UART3) 通信过程中发生错误时的处理步骤

在 UART (UART0 ~ UART3) 通信过程中发生错误时的处理步骤如图 14-96 和图 14-97 所示。

图 14-96 发生奇偶校验错误或者溢出错误时的处理步骤

软件操作	硬件状态	备注
读串行数据寄存器 mn (SDRmn)。	SSRmn 寄存器的 BFFmn 位为“0”并且通道 n 为可接收状态。	这是为了防止在错误处理的过程中结束下一次接收而出现溢出错误。
读串行状态寄存器 mn (SSRmn)。		判断错误种类，读取值用于清除错误标志。
给串行标志清除触发寄存器 mn (SIRmn) 写“1”。	清除错误标志。	通过将 SSRmn 寄存器的读取值直接写到 SIRmn 寄存器，只能清除读操作时的错误。

图 14-97 发生帧错误时的处理步骤

软件操作	硬件状态	备注
读串行数据寄存器 mn (SDRmn)。	SSRmn 寄存器的 BFFmn 位为“0”并且通道 n 为可接收状态。	这是为了防止在错误处理的过程中结束下一次接收而出现溢出错误。
读串行状态寄存器 mn (SSRmn)。		判断错误种类，读取值用于清除错误标志。
写串行标志清除触发寄存器 mn (SIRmn)。	清除错误标志。	通过将 SSRmn 寄存器的读取值直接写到 SIRmn 寄存器，只能清除读操作时的错误。
将串行通道停止寄存器 m (STm) 的 STmn 位置“1”。	串行通道允许状态寄存器 m (SEm) 的 SEMn 位为“0”并且通道 n 为停止运行状态。	
与通信方进行同步处理。		因为起始位偏移，所以可认为发生了帧错误。因此，需要与通信方重新取得同步，重新开始通信。
将串行通道开始寄存器 m (SSm) 的 SSmn 位置“1”。	串行通道允许状态寄存器 m (SEm) 的 SEMn 位为“1”并且通道 n 为可运行状态。	

备注 m: 单元号 (m=0、1) n: 通道号 (n=0 ~ 3) mn=00 ~ 03、10 ~ 13

14.7 LIN 通信的运行

14.7.1 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 0。

UART	UART0	UART1	UART2	UART3
支持 LIN 通信	支持	不支持	不支持	不支持
对象通道	SAU0 的通道 0	—	—	—
使用的引脚	TxD0	—	—	—
中断	INTST0	—	—	—
	可选择传送结束中断（单次发送模式）或者缓冲器空中断（连续发送模式）。			
错误检测标志	无			
传送数据长度	8 位			
传送速率注	Max. $f_{MCK}/6[bps]$ ($SDR10[15:9] \geq 2$)、Min. $f_{CLK}/(2 \times 2^{15} \times 128)[bps]$			
数据相位	不反相输出（默认值：高电平）。 反相输出（默认值：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			

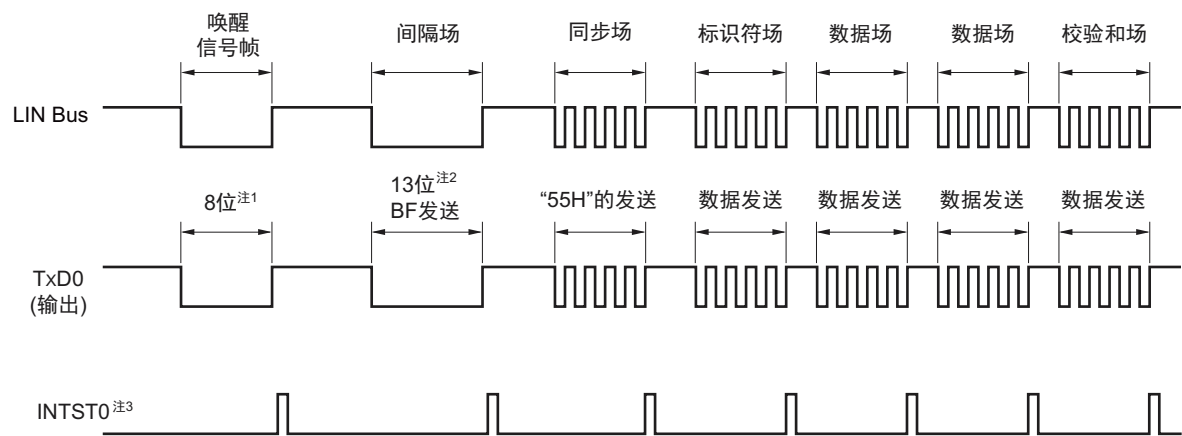
注 必须在满足此条件并且满足电特性的外围功能特性（参照“第 32 章 电特性”）的范围内使用，并且在 LIN 通信中经常使用 2.4/9.6/19.2kbps。

备注 f_{MCK} ：目标通道的运行时钟频率
 f_{CLK} ：系统时钟频率

LIN 是 Local Interconnect Network 的简称，是为降低汽车网络成本的低速（1 ~ 20kbps）串行通信协议。
LIN 通信是单主控通信，一个主控设备最多可连接 15 台从属设备。
LIN 从属设备用于开关、传动装置和传感器等的控制，这些装置通过 LIN 连接到主控设备。
LIN 主控一般连接 CAN（Controller Area Network）等的网络。
LIN 总线是单线方式的总线，通过符合 ISO9141 的收发器连接各节点。
根据 LIN 协议，主控设备发送附加波特率信息的帧，从属设备接收此帧并且校正与主控设备的波特率误差。
因此，如果从属设备的波特率误差不大于 ±15%，就能进行通信。

LIN 的发送操作的概要如图 14-98 所示。

图 14-98 LIN 的发送操作



- 注 1. 设定满足唤醒信号规定的波特率，并且通过发送“80H”的数据进行对应。
2. 间隔场规定为 13 位宽的低电平输出，因此假设主传送使用的波特率为 N[bps]，间隔场使用的波特率如下：

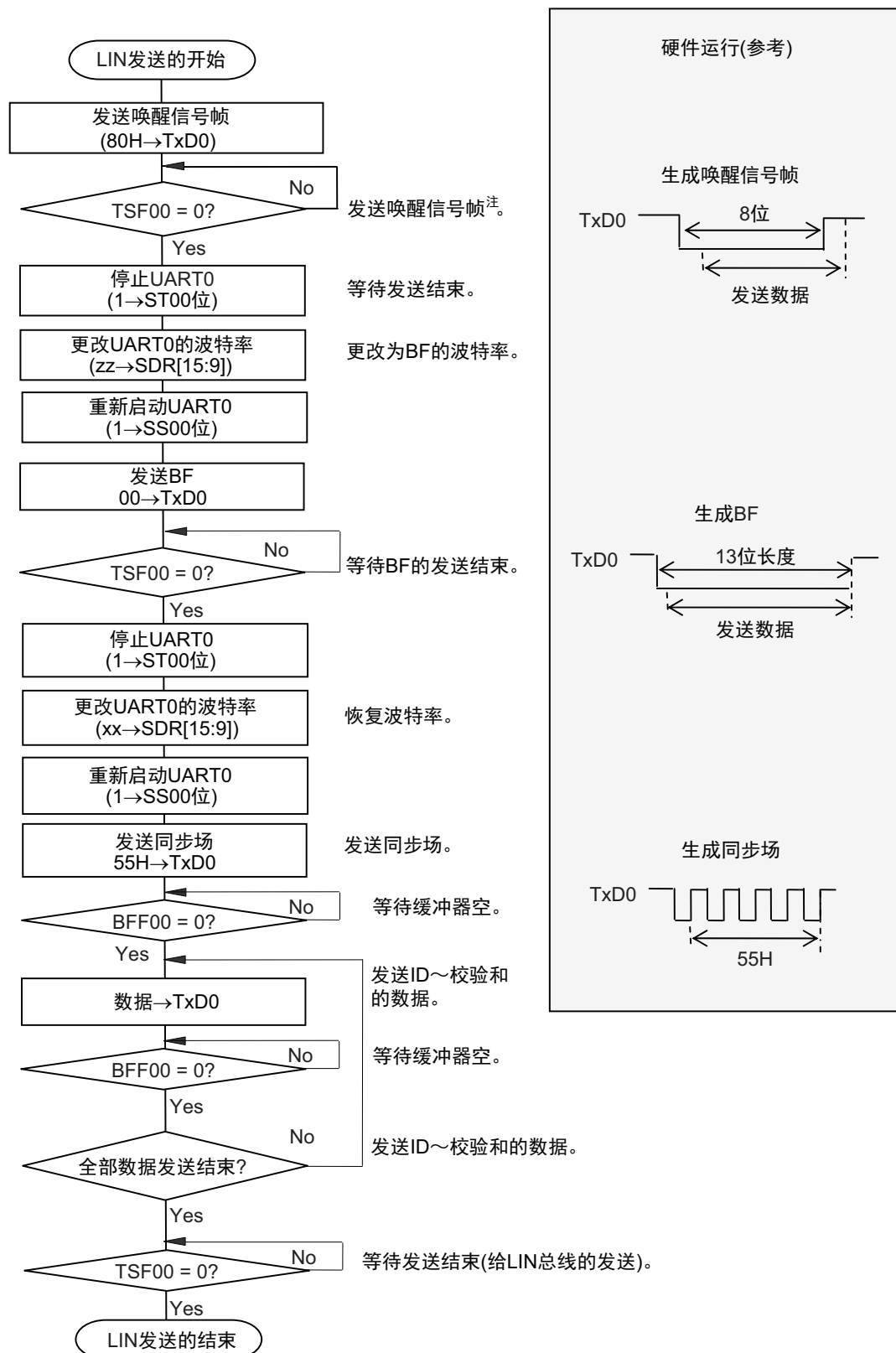
$$(\text{间隔场的波特率}) = 9/13 \times N$$

通过此波特率发送“00H”的数据，生成间隔场。

3. 在各数据发送结束时，输出 INTST0。而且在 BF 发送时也输出 INTST0。

备注 由软件控制各场间的间隔。

图 14-99 LIN 发送的流程图



注 只限于从 LIN-bus 睡眠状态启动的情况。

备注 这是从结束 UART 的初始设定并且允许从属发送开始的流程。

14.7.2 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	支持	不支持	不支持	不支持
对象通道	SAU0 的通道 1	—	—	—
使用的引脚	RxD0	—	—	—
中断	INTSR0	—	—	—
	只限于传送结束中断（禁止设定缓冲器空中断）。			
错误中断	INTSRE0	—	—	—
错误检测标志	- 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			
传送速率注	Max. $f_{MCK}/6[\text{bps}]$ （ $\text{SDR01}[15:9] \geq 2$ ）、Min. $f_{CLK}/(2 \times 2^{15} \times 128)[\text{bps}]$			
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			

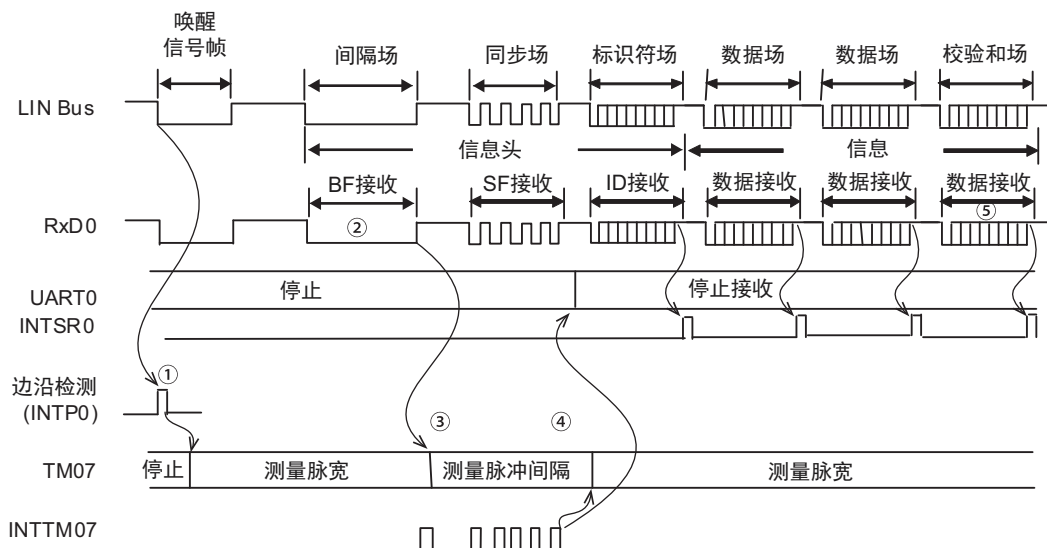
注 在满足以上条件及电特性中的外围功能特性（参照“第 32 章 电特性”）的范围内进行此操作。

备注 f_{MCK} ：目标通道的运行时钟频率

f_{CLK} ：系统时钟频率

LIN 的接收操作的概要如图 14-100 所示。

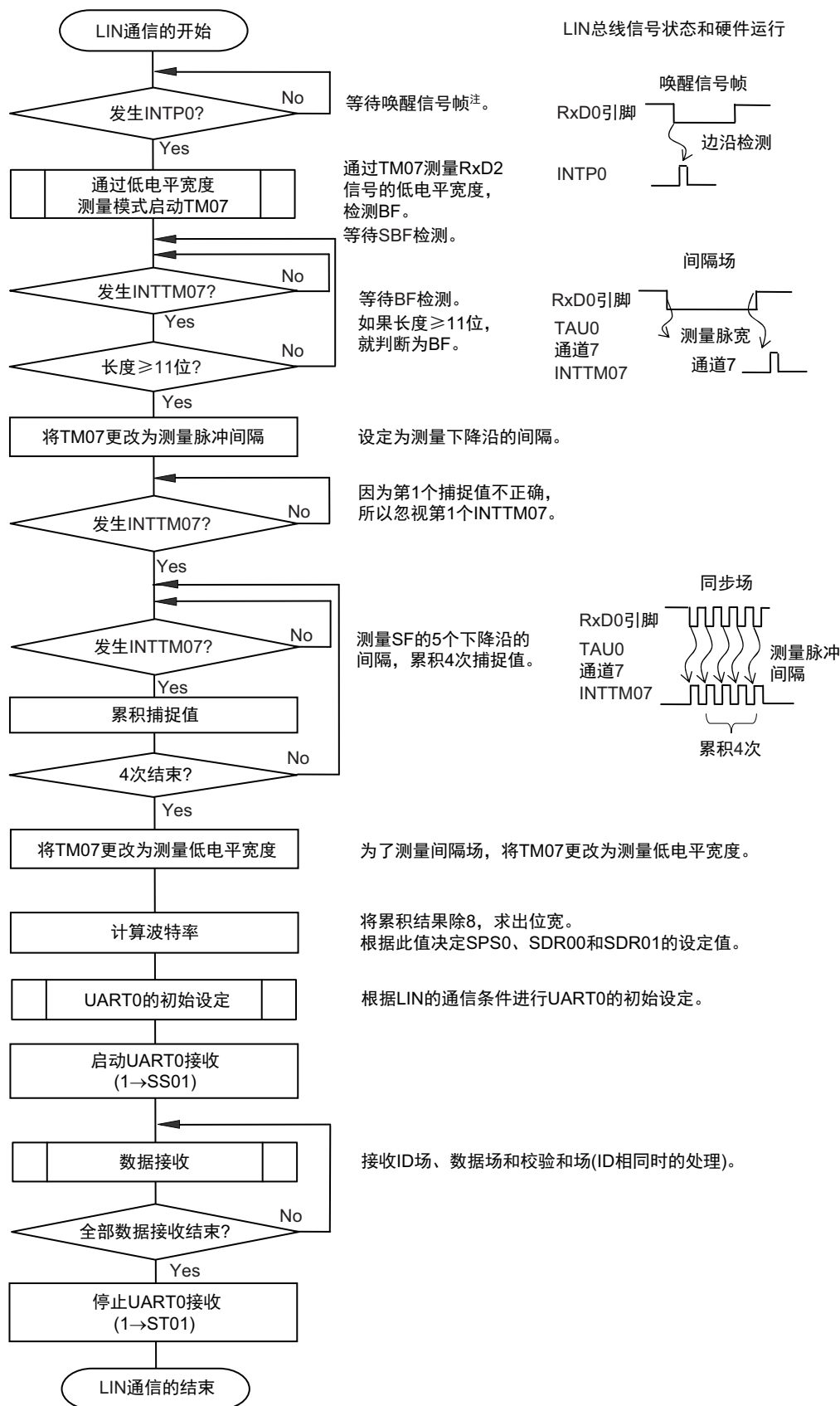
图 14-100 LIN 的接收操作



信号处理的流程如下：

- ① 通过检测引脚的中断边沿 (INTP0) 来检测唤醒信号。当检测到唤醒信号时，为了测量BF的低电平宽度，将TM07设定为测量脉宽，然后进入BF接收等待状态。
- ② 如果检测到BF的下降沿，TM07就开始测量低电平的宽度，并且在BF的上升沿进行捕捉。根据捕捉到的值判断是否为BF信号。
- ③ 当BF接收正常结束时，必须将TM07设定为测量脉冲间隔，并且测量4次同步场的RxD0信号下降沿的间隔（参照“6.8.3 作为输入脉冲间隔测量的运行”）。
- ④ 根据同步场（SF）的位间隔计算波特率误差。然后，必须在暂停UART0运行后调整（重新设定）波特率。
- ⑤ 必须通过软件区分校验和场。还必须通过软件在接收校验和场后对UART0进行初始化并且再次设定为BF接收等待状态。

图 14-101 LIN 接收的流程图



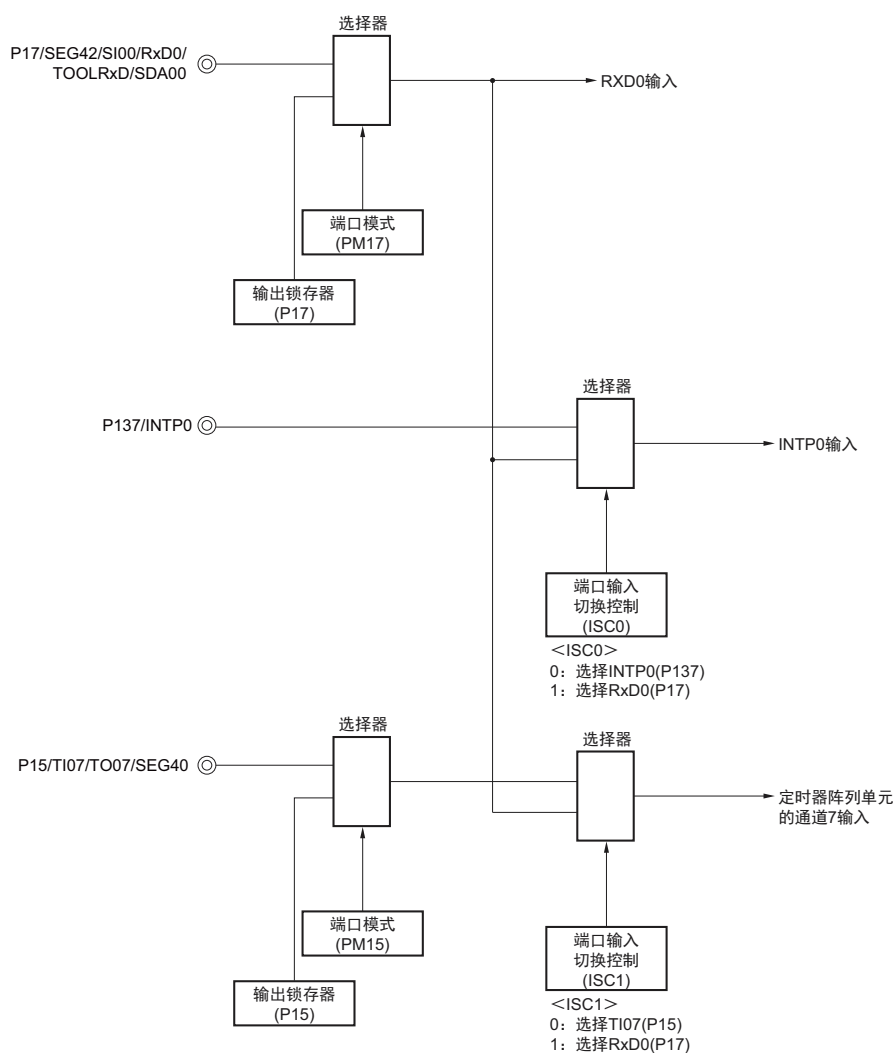
注 只在睡眠状态下才需要。

用于 LIN 接收操作的端口结构图如图 14-102 所示。

通过外部中断（INTP0）的边沿检测，接收 LIN 主控发送的唤醒信号。能通过定时器阵列单元 0 的外部事件捕捉运行，测量 LIN 主控发送的同步场的长度以及计算波特率误差。

通过端口输入切换控制（ISC0/ISC1），能不在外部连线而将用于接收的端口输入（RxD0）的输入源输入到外部中断（INTP0）和定时器阵列单元。

图 14-102 用于 LIN 接收操作的端口结构图



备注 ISC0、ISC1：输入切换控制寄存器（ISC）的 bit0 和 bit1（参照图 14-21。）

用于 LIN 通信运行的外围功能总结如下：

<使用的外围功能>

- 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、间隔场（BF）检测
用途：检测同步场（SF）的长度，并且通过将其长度除以位数来检测波特率误差（通过捕捉模式测量 RxD0 输入边沿的间隔）。
测量低电平宽度，判断是否为间隔场（BF）。
- 串行阵列单元 0（SAU0）的通道 0 和通道 1（UART0）

14.8 简易 I²C（IIC00、IIC10）通信的运行

这是通过串行时钟（SCL）和串行数据（SDA）共 2 条线与多个设备进行时钟同步通信的功能。因此简易 I²C 是为了与 EEPROM、闪存、A/D 转换器等设备进行单通信而设计的，所以只用作主控设备。

对于开始条件和停止条件，必须遵守 I²C 总线的总线特性，通过软件操作控制寄存器。

[数据的发送和接收]

- 主控发送、主控接收（只限于单主控的主控功能）
- ACK 输出功能注、ACK 检测功能
- 8 位数据长度（在发送地址时，用高 7 位指定地址，用最低位进行 R/W 控制）
- 通过软件产生开始条件和停止条件

[中断功能]

- 传送结束中断

[错误检测标志]

- 奇偶校验错误（ACK 错误）

※ [简易 I²C 不支持的功能]

- 从属发送、从属接收
- 多主控功能（仲裁失败检测功能）
- 等待检测功能

注 在接收最后的数据时，如果给 SOEmn 位（SOEm 寄存器）写“0”来停止串行通信数据的输出，就不输出 ACK。详细内容请参照“14.8.3(2) 处理流程”。

备注 m：单元号（m=0） n：通道号（n=0、2） mn=00、02

支持简易 I²C（IIC00、IIC10）的通道为 SAU0 的通道 0 和通道 2。

单元	通道	用作 CSI	用作 UART	用作简易 I ² C
0	0	CSI00	UART0（支持 LIN-bus）	IIC00
	1	—		—
	2	—	UART1	IIC10
	3	—		—
1	0	—	UART2	—
	1	—		—
	2	—	UART3	—
	3	—		—

简易 I²C（IIC00、IIC10）有如下 4 种通信运行：

- 地址字段发送（参照 14.8.1。）
- 数据发送（参照 14.8.2。）
- 数据接收（参照 14.8.3。）
- 停止条件的产生（参照 14.8.4。）

14.8.1 地址字段发送

地址字段发送是为了特别指定传送对象（从属设备）而在 I²C 通信时最先进行的发送运行。在产生开始条件后，将地址（7 位）和传送方向（1 位）作为 1 帧进行发送。

简易 I ² C	IIC00	IIC10
对象通道	SAU0 的通道 0	SAU0 的通道 2
使用的引脚	SCL00、SDA00 注 1	SCL10、SDA10 注 1
中断	INTIIC00	INTIIC10
	只限于传送结束中断（不能选择缓冲器空中断）。	
错误检测标志	ACK 错误检测标志（PEFmn）	
传送数据长度	8 位（将高 7 位作为地址并且将低 1 位作为 R/W 控制进行发送）	
传送速率注 2	Max. $f_{MCK}/4$ [Hz]（SDRmn[15:9] ≥ 1） f_{MCK} ：对象通道的运行时钟频率 但是，必须在 I ² C 的各模式中满足以下条件： • Max. 1MHz（增强型快速模式） • Max. 400kHz（快速模式） • Max. 100kHz（标准模式）	
数据电平	不反相输出（默认值：高电平）。	
奇偶校验位	无奇偶校验位	
停止位	附加 1 位（用于 ACK 接收时序）	
数据方向	MSB 优先	

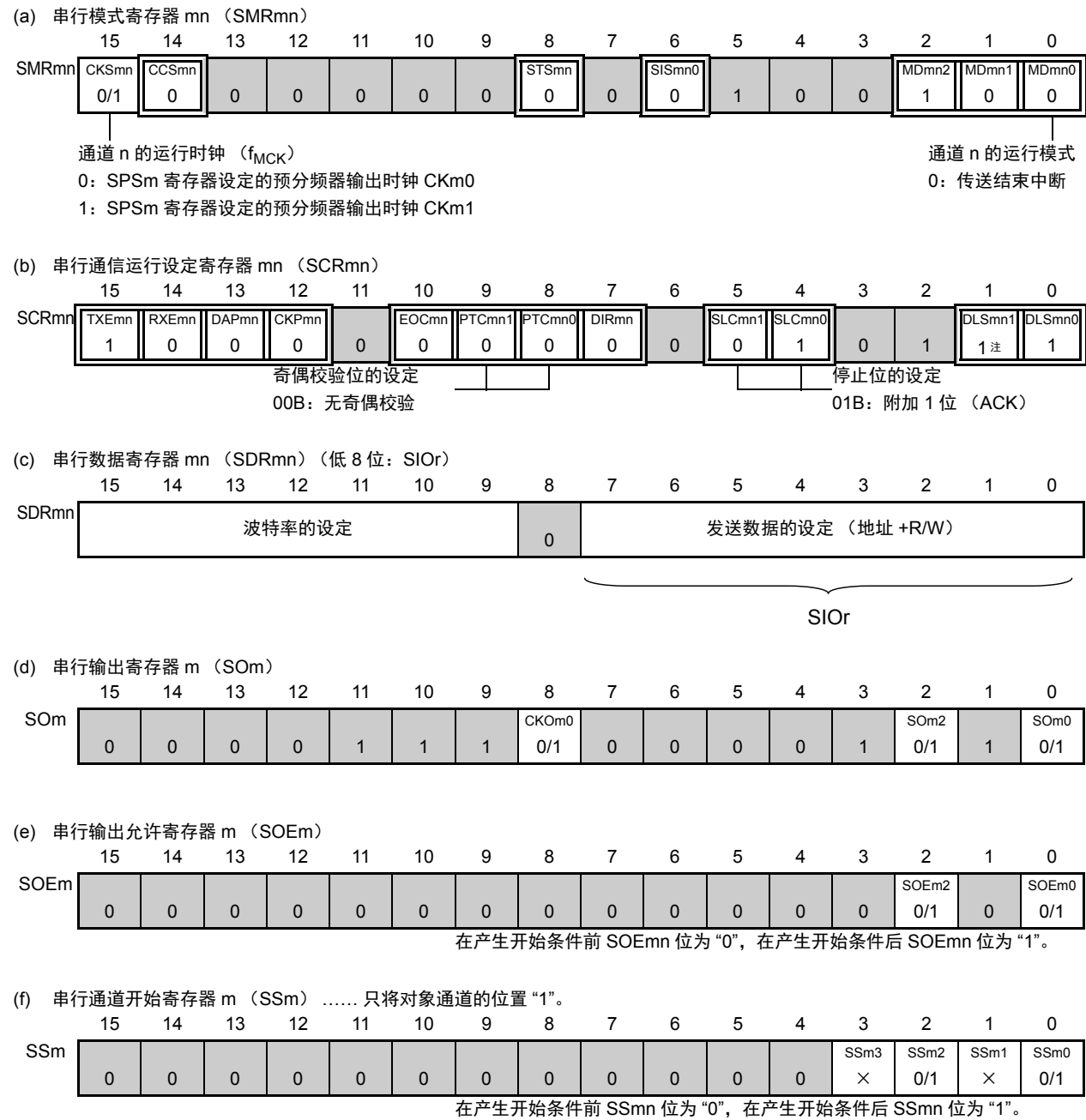
注 1. 要通过简易 I²C 进行通信时，必须通过端口输出模式寄存器（POM0、POM1）设定 N 沟道漏极开路输出（V_{DD} 耐压）模式（POM06、POM17=1）。（详细内容请参照“4.3 端口功能的控制寄存器”。）要将 IIC00、IIC10 与不同电位的外部器件通信时，也必须同样地将时钟输入 / 输出引脚（SCL00、SCL10）设定为 N 沟道漏极开路输出（V_{DD} 耐压）模式（POM05、POM16=1）。详细内容请参照“4.4.4 不同电位（1.8V、2.5V、3V）的输入 / 输出缓冲器对应”。

2. 必须在满足此条件并且满足电特性的外围功能特性（参照“第 32 章 电特性”）的范围内使用。

备注 m：单元号（m=0） n：通道号（n=0、2） mn=00、02

(1) 寄存器的设定

图 14-103 简易 I²C（IIC00、IIC10）地址字段发送时的寄存器设定内容例子



注 只限 SCR00 寄存器，其他固定为“1”。

备注 1. m: 单元号 (m=0) n: 通道号 (n=0、2) r: IIC 号 (r=00、10) mn=00、02

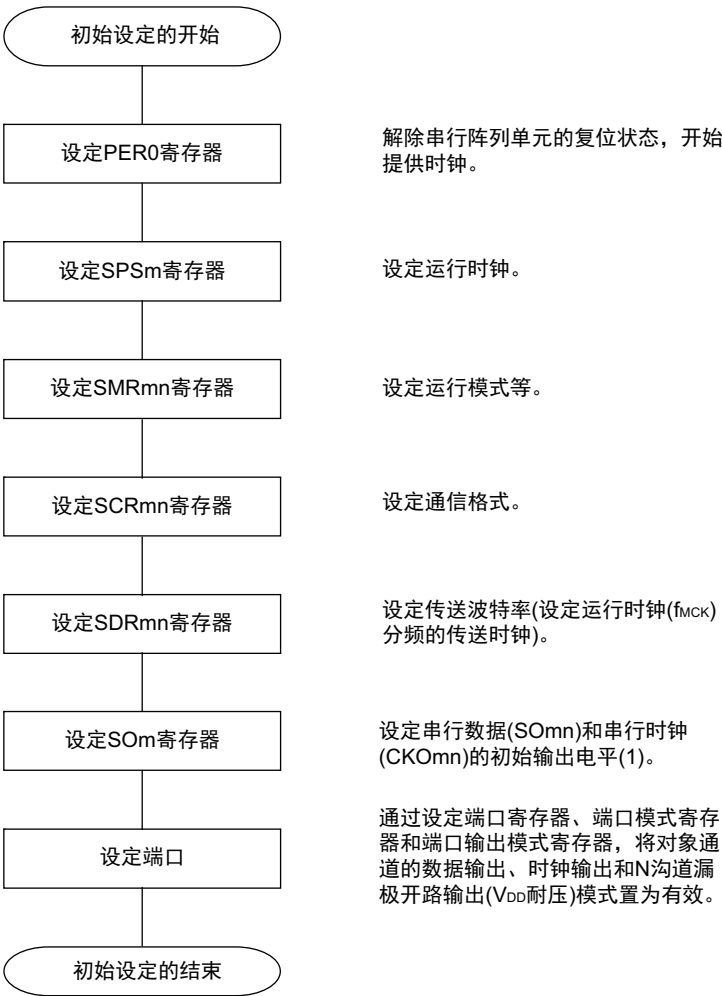
2. □: 在 IIC 模式中为固定设定。■: 不能设定 (设定初始值)。

×: 这是在此模式中不能使用的位 (在其他模式中也不使用的情况下，设定初始值)。

0/1: 根据用户的用途置“0”或者“1”。

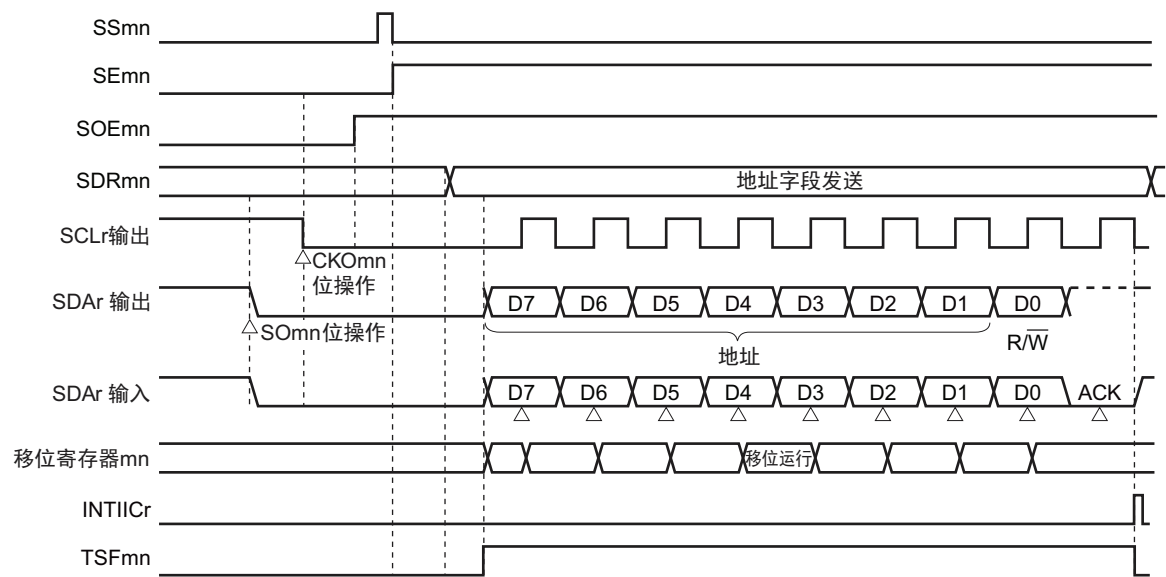
(2) 操作步骤

图 14-104 简易 I2C 地址字段发送的初始设定步骤



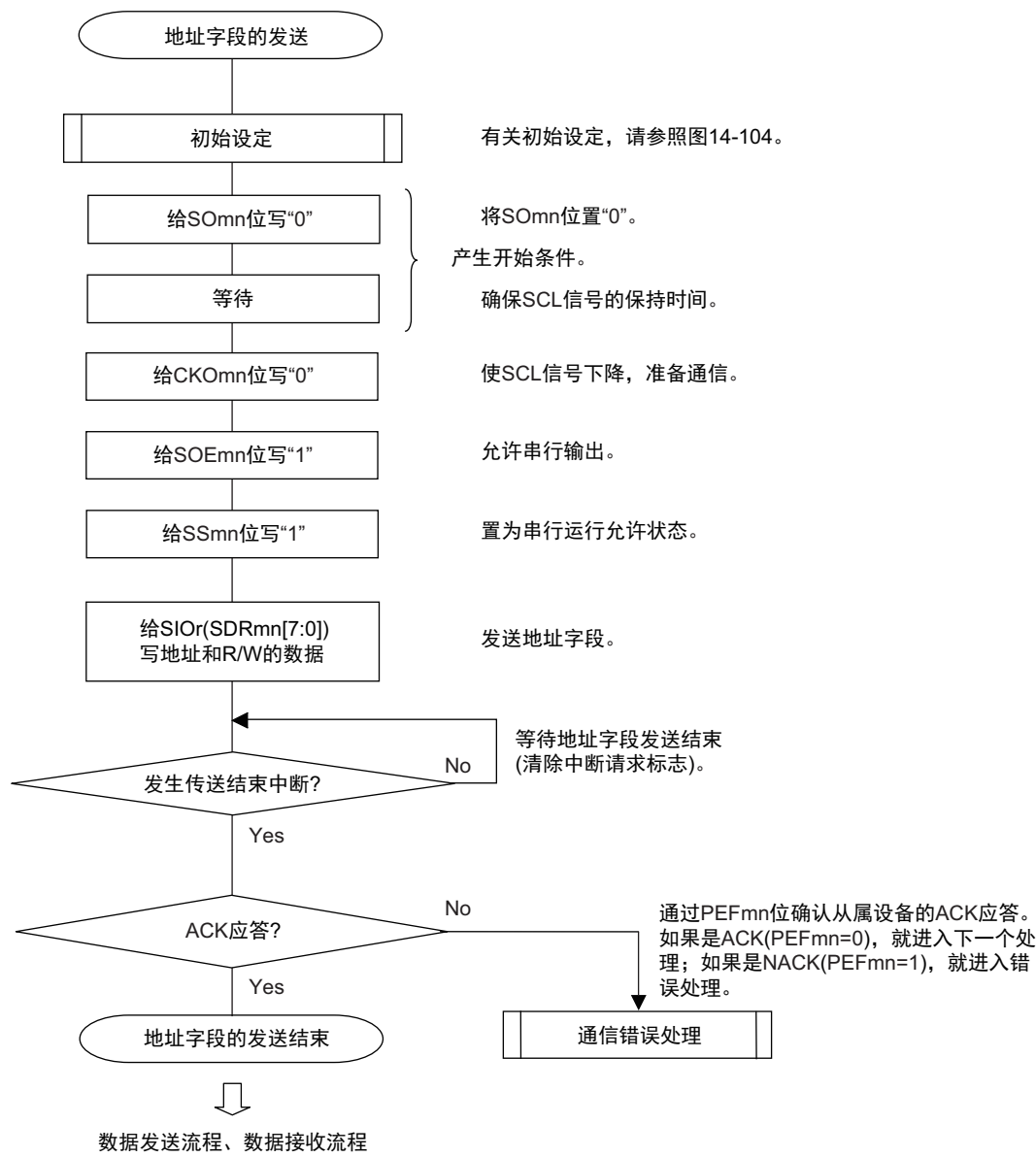
(3) 处理流程

图 14-105 地址字段发送的时序图



备注 m: 单元号 (m=0) n: 通道号 (n=0、2) r: IIC 号 (r=00、10) mn=00、02

图 14-106 简易 I2C 地址字段发送的流程图



14.8.2 数据发送

数据发送是在发送地址字段后给该传送对象（从属设备）发送数据的运行。在给对象从属设备发送全部数据后产生停止条件并且释放总线。

简易 I ² C	IIC00	IIC10
对象通道	SAU0 的通道 0	SAU0 的通道 2
使用的引脚	SCL00、SDA00 注 1	SCL10、SDA10 注 1
中断	INTIIC00	INTIIC10
	只限于传送结束中断（不能选择缓冲器空中断）。	
错误检测标志	ACK 错误标志（PEFmn）	
传送数据长度	8 位	
传送速率注 2	Max. $f_{MCK}/4[Hz]$ （ $SDRmn[15:9] \geq 1$ ） f_{MCK} ：对象通道的运行时钟频率 但是，必须在 I ² C 的各模式中满足以下条件： • Max. 1MHz（增强型快速模式） • Max. 400kHz（快速模式） • Max. 100kHz（标准模式）	
数据电平	不反相输出（默认值：高电平）。	
奇偶校验位	无奇偶校验位	
停止位	附加 1 位（用于 ACK 接收时序）	
数据方向	MSB 优先	

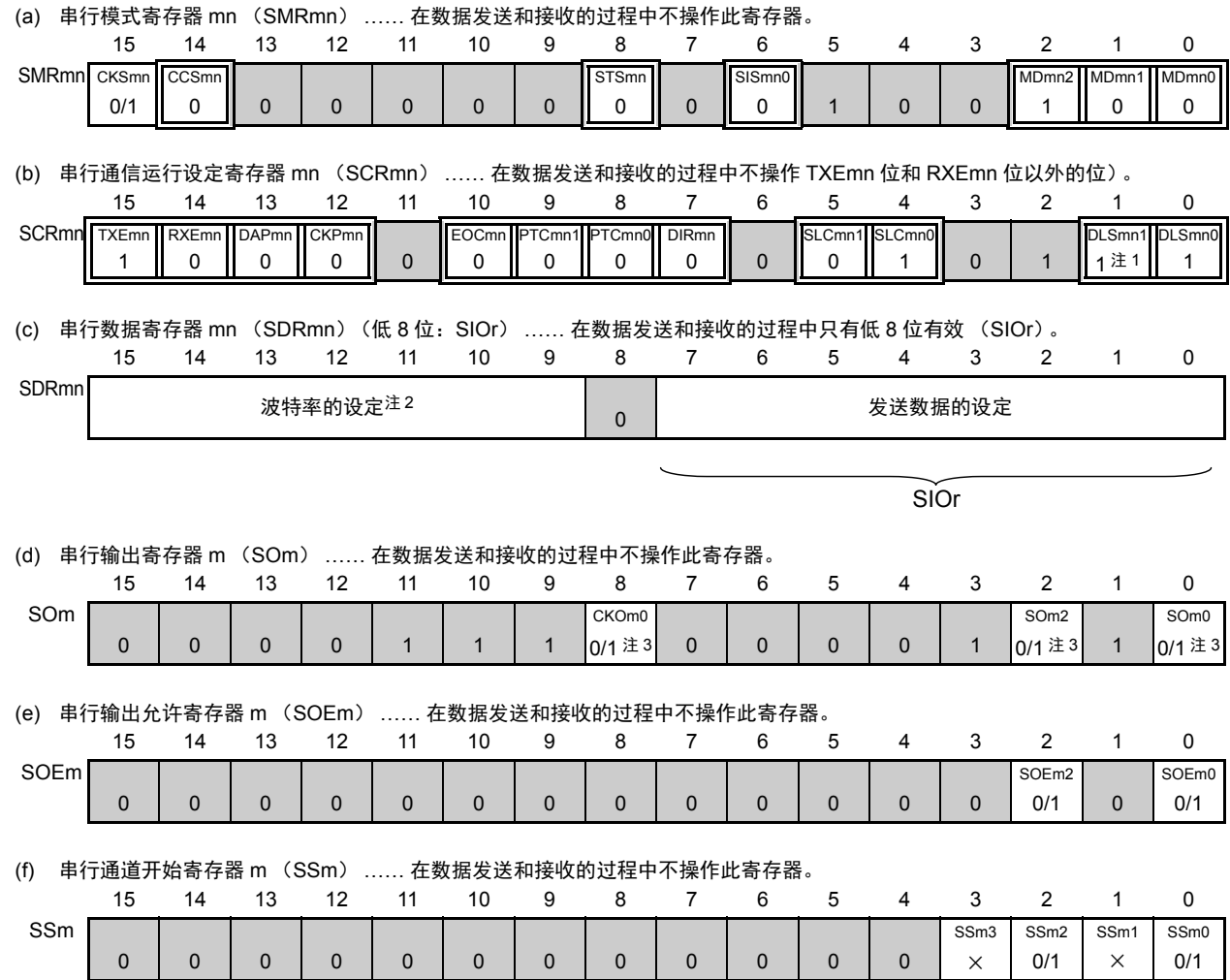
注 1. 要通过简易 I²C 进行通信时，必须通过端口输出模式寄存器（POM0、POM1）设定 N 沟道漏极开路输出（V_{DD} 耐压）模式（POM06、POM17=1）。（详细内容请参照“4.3 端口功能的控制寄存器”。）要将 IIC00、IIC10 与不同电位的外部器件通信时，也必须同样地将时钟输入 / 输出引脚（SCL00、SCL10）设定为 N 沟道漏极开路输出（V_{DD} 耐压）模式（POM05、POM16=1）。详细内容请参照“4.4.4 不同电位（1.8V、2.5V、3V）的输入 / 输出缓冲器对应”。

2. 必须在满足此条件并且满足电特性的外围功能特性（参照“第 32 章 电特性”）的范围内使用。

备注 m：单元号（m=0） n：通道号（n=0、2） mn=00、02

(1) 寄存器的设定

图 14-107 简易 I²C（IIC00、IIC10）数据发送时的寄存器设定内容例子



注 1. 只限 SCR00，其他固定为“1”。。

2. 因为已经在发送地址字段时设定，所以不需要设定。

3. 在通信运行过程中，值因通信数据而变。

备注 1. m：单元号（m=0） n：通道号（n=0、2） r：IIC 号（r=00、10） mn=00、02

2. □：在 IIC 模式中为固定设定。■：不能设定（设定初始值）。
×：这是在此模式中不能使用的位（在其他模式中也不使用的情况下，设定初始值）。
0/1：根据用户的用途置“0”或者“1”。

(2) 处理流程

图 14-108 数据发送的时序图

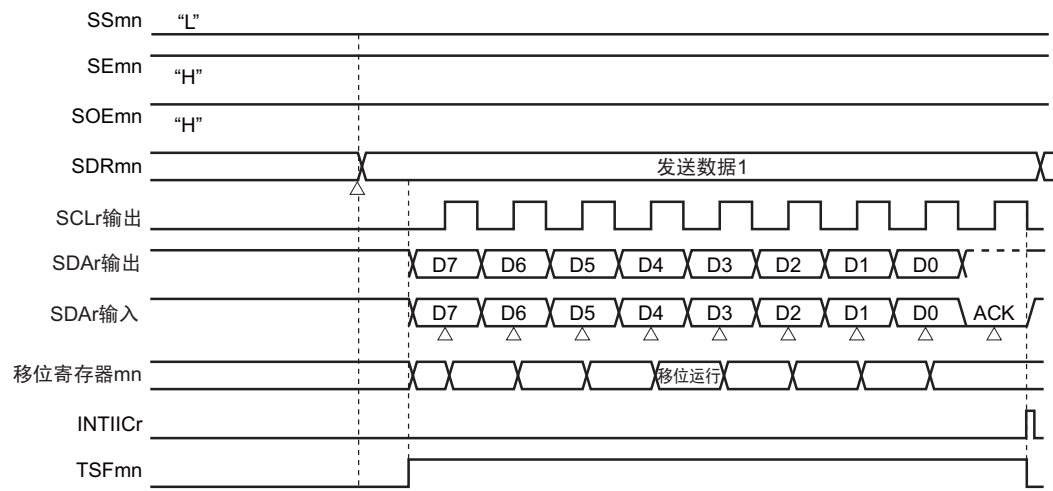
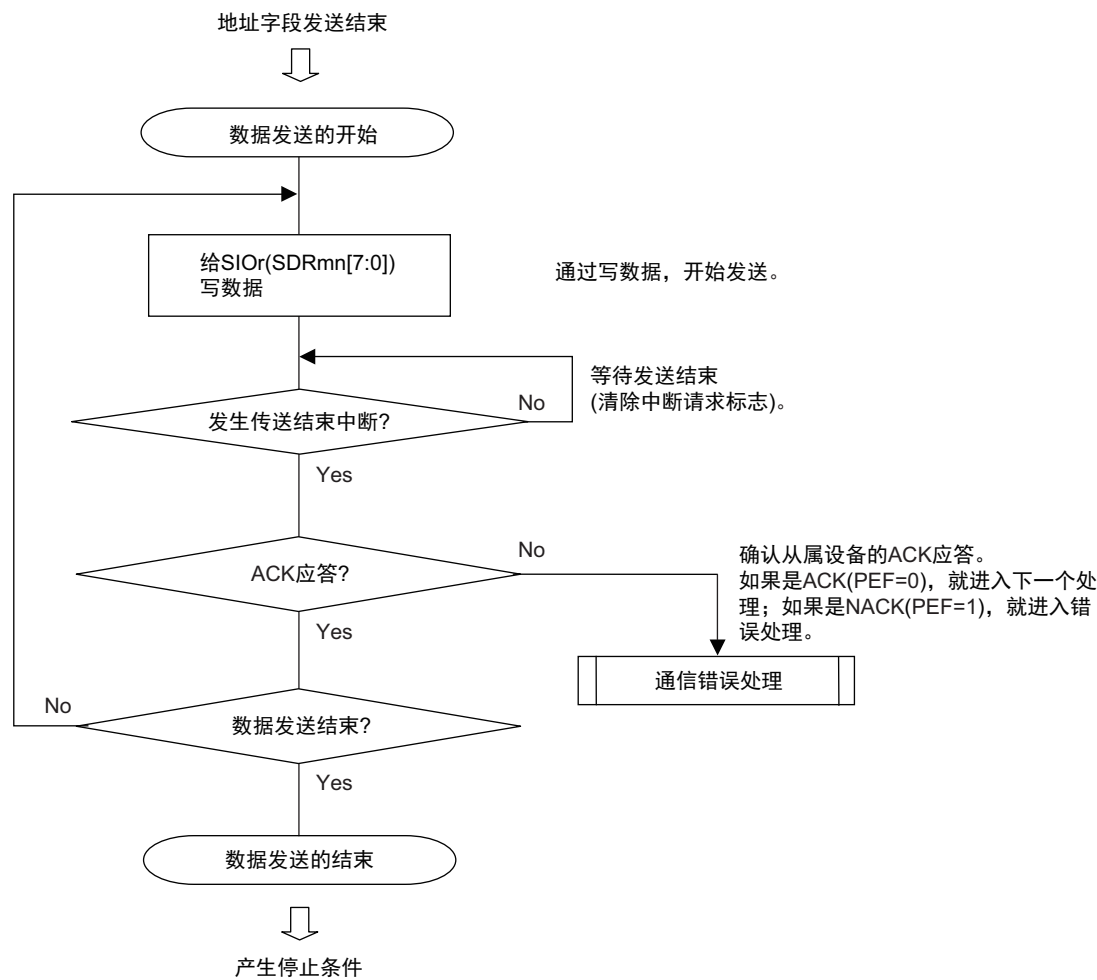


图 14-109 简易 I²C 数据发送的流程图



14.8.3 数据接收

数据接收是在发送地址字段后从传送对象（从属设备）接收数据的运行。在从对象从属设备接收全部数据后产生停止条件并且释放总线。

简易 I ² C	IIC00	IIC10
对象通道	SAU0 的通道 0	SAU0 的通道 2
使用的引脚	SCL00、SDA00 注 1	SCL10、SDA10 注 1
中断	INTIIC00	INTIIC10
	只限于传送结束中断（不能选择缓冲器空中断）。	
错误检测标志	只有溢出错误检测标志（OVFmn）。	
传送数据长度	8 位	
传送速率注 2	Max. $f_{MCK}/4$ [Hz] ($SDRmn[15:9] \geq 1$) f_{MCK} : 对象通道的运行时钟频率 但是，必须在 I ² C 的各模式中满足以下条件： • Max. 1MHz（增强型快速模式） • Max. 400kHz（快速模式） • Max. 100kHz（标准模式）	
数据电平	不反相输出（默认值：高电平）。	
奇偶校验位	无奇偶校验位	
停止位	附加 1 位（ACK 发送）	
数据方向	MSB 优先	

注 1. 要通过简易 I²C 进行通信时，必须通过端口输出模式寄存器（POM0、POM1）设定 N 沟道漏极开路输出（V_{DD} 耐压）模式（POM06、POM17=1）。（详细内容请参照“4.3 端口功能的控制寄存器”。）要将 IIC00、IIC10 与不同电位的外部器件通信时，也必须同样地将时钟输入 / 输出引脚（SCL00、SCL10）设定为 N 沟道漏极开路输出（V_{DD} 耐压）模式（POM05、POM16=1）。详细内容请参照“4.4.4 不同电位（1.8V、2.5V、3V）的输入 / 输出缓冲器对应”。

2. 必须在满足此条件并且满足电特性的外围功能特性（参照“第 32 章 电特性”）的范围内使用。

备注 m: 单元号（m=0） n: 通道号（n=0、2） mn=00、02

(1) 寄存器的设定

图 14-110 简易 I²C (IIC00、IIC01) 数据接收时的寄存器设定内容例子

(a) 串行模式寄存器 mn (SMRmn) 在数据发送和接收的过程中不操作此寄存器。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMRmn	CKSmn	CCSmn						STSmn		SISmn0				MDmn2	MDmn1	MDmn0
	0/1	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0

(b) 串行通信运行设定寄存器 mn (SCRmn) 在数据发送和接收的过程中不操作 TXEmn 位和 RXEmn 位以外的位。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCRmn	TXEmn	RXEmn	DAPmn	CKPmn		EOCmn	PTCmn1	PTCmn0	DIRmn		SLCmn1	SLCmn0			DLSmn1	DLSmn0
	0	1	0	0	0	0	0	0	0	0	0	1	0	1	1 注 1	1

(c) 串行数据寄存器 mn (SDRmn) (低 8 位: SIO_r)

(c) 串行发送寄存器 Rmn (SDRmn) 数据位 SDRmn																
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SDRmn	波特率的设定注 2							0	虚拟发送数据的设定 (FFH)							
									SIO _r							

(d) 串行输出寄存器 m (SOM_m) 在数据发送和接收的过程中不操作此寄存器。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOM _m								CKOm0						SOM2		SOM0
	0	0	0	0	1	1	1	0/1 注 3	0	0	0	0	1	0/1 注 3	1	0/1 注 3

(e) 串行输出允许寄存器 m (SOEm_m) 在数据发送和接收的过程中不操作此寄存器。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOEm _m														SOEm2		SOEm0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0/1	0	0/1

(f) 串行通道开始寄存器 m (SSm_m) 在数据发送和接收的过程中不操作此寄存器。

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSm _m													SSm3	SSm2	SSm1	SSm0
	0	0	0	0	0	0	0	0	0	0	0	0	×	0/1	×	0/1

注 1. 只限 SCR00 寄存器, 其他固定为“1”。

2. 因为已经在发送地址字段时设定, 所以不需要设定。

3. 在通信运行过程中, 值因通信数据而变。

备注 1. m: 单元号 (m=0) n: 通道号 (n=0、2) r: IIC 号 (r=00、10) mn=00、02

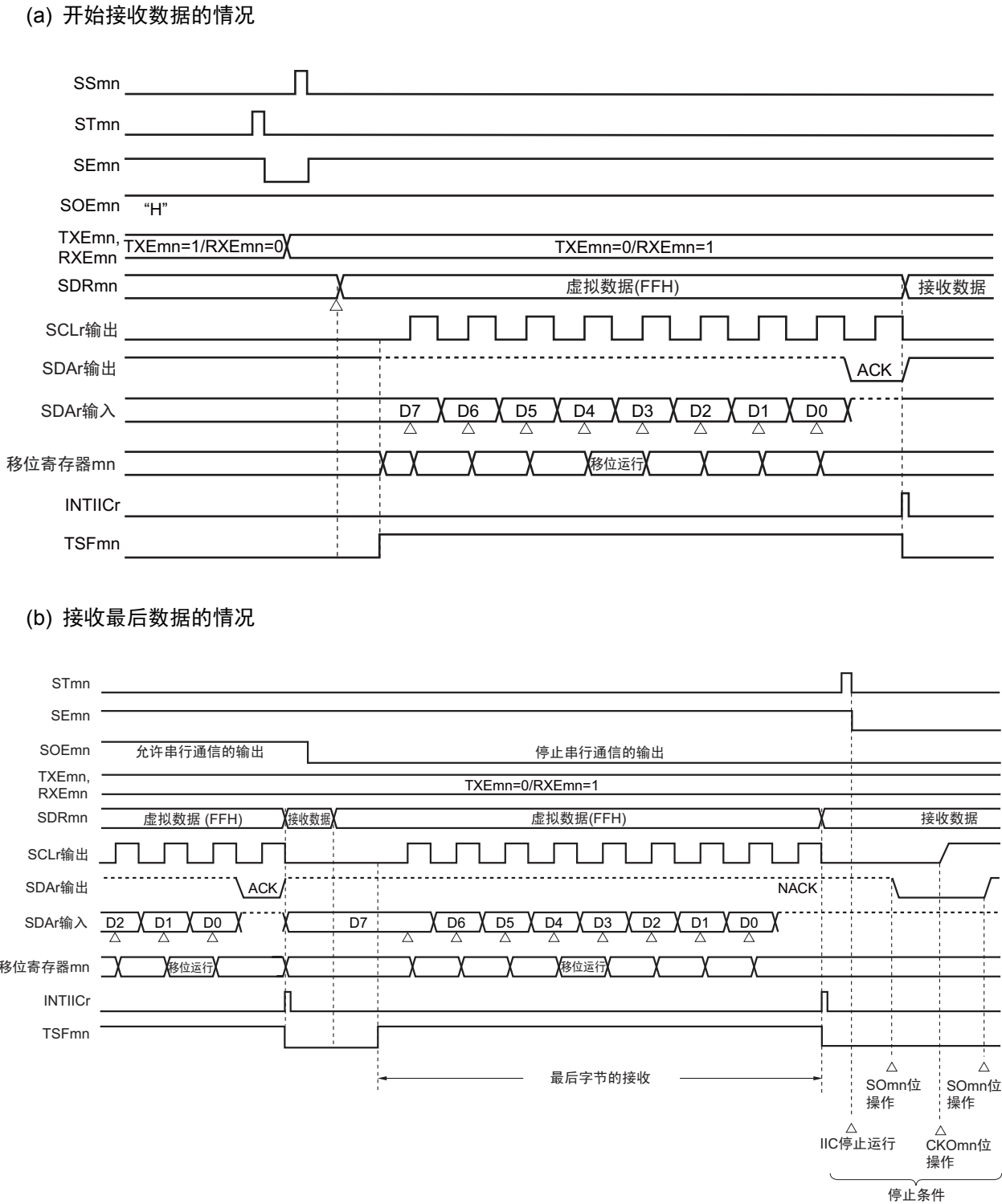
2. : 在 IIC 模式中为固定设定。 : 不能设定 (设定初始值)。

×: 这是在此模式中不能使用的位 (在其他模式中也不使用的情况下, 设定初始值)。

0/1: 根据用户的用途置“0”或者“1”。

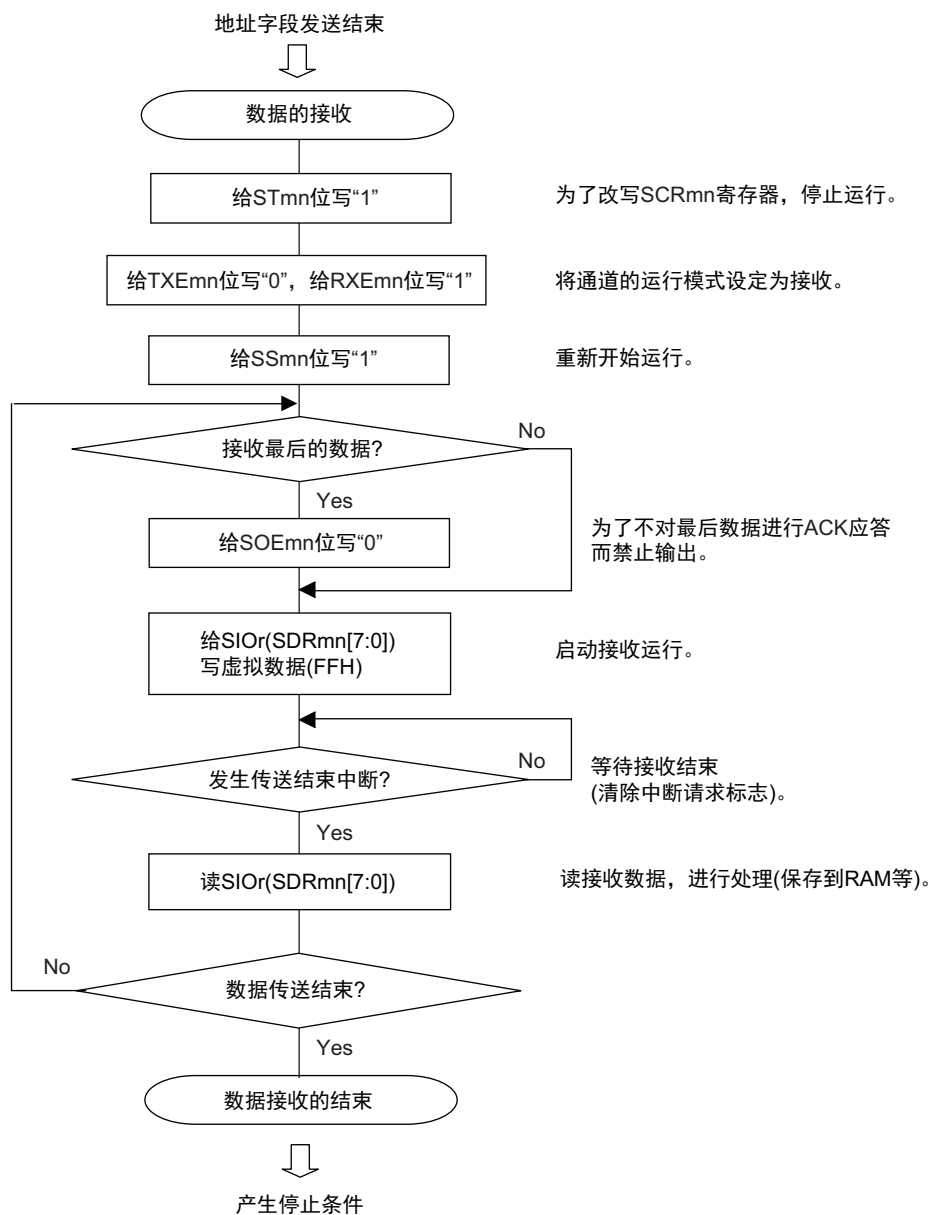
(2) 处理流程

图 14-111 数据接收的时序图



备注 m: 单元号 (m=0) n: 通道号 (n=0、2) r: IIC 号 (r=00、10) mn=00、02

图 14-112 数据接收的流程图



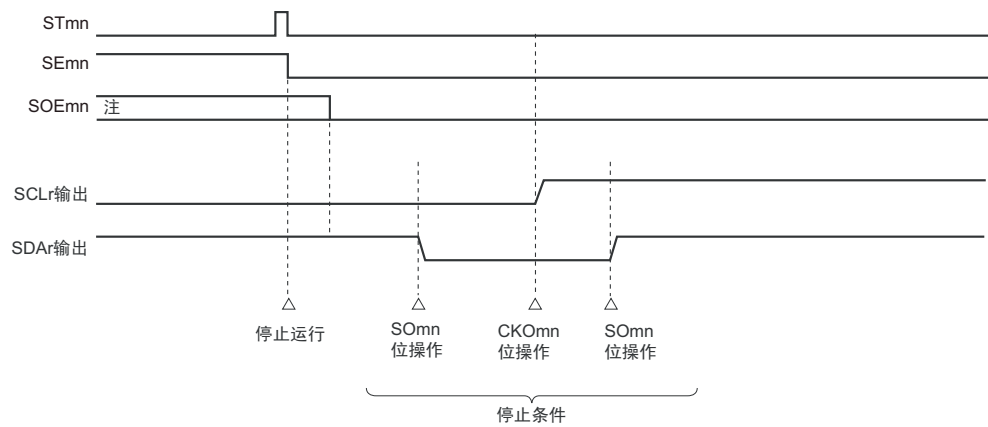
注意 在接收最后数据时不输出 ACK (NACK)。此后，先通过将串行通道停止寄存器 m (STm) 的 STmn 位置“1”来停止运行，然后产生停止条件来结束通信。

14.8.4 停止条件的产生

在与对象从属设备进行了全部数据的发送和接收后，产生停止条件并且释放总线。

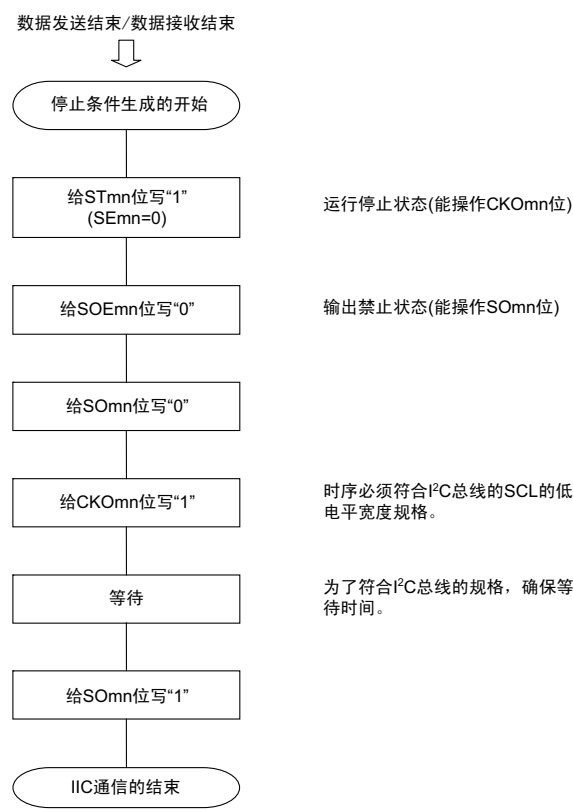
(1) 处理流程

图 14-113 产生停止条件的时序图



注 在接收时，在接收最后数据前已经将串行输出允许寄存器 m（SOEm）的 SOEmn 位置“0”。

图 14-114 产生停止条件的流程图



14.8.5 传送速率的计算

简易 I²C（IIC00、IIC10）通信的传送速率能用以下计算式进行计算。

$$(\text{传送速率}) = \{ \text{对象通道的运行时钟 } (f_{\text{MCK}}) \text{ 频率} \} \div (\text{SDRmn}[15:9] + 1) \div 2$$

注意 禁止将 SDRmn[15:9] 设定为“0000000B”，SDRmn[15:9] 的设定值必须大于等于“0000001B”。简易 I²C 输出的 SCL 信号的占空比为 50%。在 I²C 总线规格中，SCL 信号的低电平宽度大于高电平宽度。因此，如果设定为快速模式的 400kbps 或者增强型快速模式的 1Mbps，SCL 信号输出的低电平宽度就小于 I²C 总线的规格值。必须给 SDRmn[15:9] 设定能满足 I²C 总线规格的值。

备注 1. 因为 SDRmn[15:9] 的值为串行数据寄存器 mn（SDRmn）的 bit15～9 的值（0000001B～1111111B），所以为 1～127。

2. m: 单元号（m=0） n: 通道号（n=0、2） mn=00、02

运行时钟（ f_{MCK} ）取决于串行时钟选择寄存器 m（SPSm）和串行模式寄存器 mn（SMRmn）的 bit15（CKSmn 位）。

表 14-5 简易 I²C 运行时钟的选择

SMRmn 寄存器	SPSm 寄存器								运行时钟 (f _{MCK}) 注	
CKSmn	PRS m13	PRS m12	PRS m11	PRS m10	PRS m03	PRS m02	PRS m01	PRS m00		f _{CLK} =24 MH 运行时
0	X	X	X	X	0	0	0	0	f _{CLK}	24MHz
	X	X	X	X	0	0	0	1	f _{CLK} /2	12MHz
	X	X	X	X	0	0	1	0	f _{CLK} /2 ²	6MHz
	X	X	X	X	0	0	1	1	f _{CLK} /2 ³	3MHz
	X	X	X	X	0	1	0	0	f _{CLK} /2 ⁴	1.5MHz
	X	X	X	X	0	1	0	1	f _{CLK} /2 ⁵	750kHz
	X	X	X	X	0	1	1	0	f _{CLK} /2 ⁶	375kHz
	X	X	X	X	0	1	1	1	f _{CLK} /2 ⁷	187.5kHz
	X	X	X	X	1	0	0	0	f _{CLK} /2 ⁸	93.8kHz
	X	X	X	X	1	0	0	1	f _{CLK} /2 ⁹	46.9kHz
	X	X	X	X	1	0	1	0	f _{CLK} /2 ¹⁰	23.4kHz
	X	X	X	X	1	0	1	1	f _{CLK} /2 ¹¹	11.7kHz
1	0	0	0	0	X	X	X	X	f _{CLK}	24MHz
	0	0	0	1	X	X	X	X	f _{CLK} /2	12MHz
	0	0	1	0	X	X	X	X	f _{CLK} /2 ²	6MHz
	0	0	1	1	X	X	X	X	f _{CLK} /2 ³	3MHz
	0	1	0	0	X	X	X	X	f _{CLK} /2 ⁴	1.5MHz
	0	1	0	1	X	X	X	X	f _{CLK} /2 ⁵	750kHz
	0	1	1	0	X	X	X	X	f _{CLK} /2 ⁶	375kHz
	0	1	1	1	X	X	X	X	f _{CLK} /2 ⁷	187.5kHz
	1	0	0	0	X	X	X	X	f _{CLK} /2 ⁸	93.8kHz
	1	0	0	1	X	X	X	X	f _{CLK} /2 ⁹	46.9kHz
	1	0	1	0	X	X	X	X	f _{CLK} /2 ¹⁰	23.4kHz
	1	0	1	1	X	X	X	X	f _{CLK} /2 ¹¹	11.7kHz
上述以外									禁止设定	

注 要更改被选择为 f_{CLK} 的时钟（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）的运行（串行通道停止寄存器 m（STm）=000FH）后进行更改。

备注 1. X：忽略

2. m：单元号（m = 0） n：通道号（n = 0、2） mn = 00、02

f_{MCK} = f_{CLK} = 24 MHz 时的 I²C 传送速率的设定例子如下所示。

I ² C 传送模式 (期待的传送速率)	f _{CLK} =24MHz			
	运行时钟 (f _{MCK})	SDRmn[15:9]	计算传送速率	与期待的传送速率的误差
100 kHz	f _{CLK} /2	59	100kHz	0.0%
400 kHz	f _{CLK}	29	380kHz	5.0% 注
1 MHz	f _{CLK}	5	0.84MHz	16.0% 注

注 因为 SCL 信号的占空比为 50%，所以不能将误差设定为“0%”左右。

14.8.6 在简易 I²C（IIC00、IIC10）通信过程中发生错误时的处理步骤

在简易 I²C（IIC00、IIC10）通信过程中发生错误时的处理步骤分别如图 14-115、图 14-116 所示。

图 14-115 发生溢出错误时的处理步骤

软件操作	硬件状态	备注
读串行数据寄存器 mn（SDRmn）。	SSRmn 寄存器的 BFFmn 位变为“0”并且通道 n 进入可接收状态。	在错误处理过程中结束下一个接收时执行此操作，可防止发生溢出错误。
读串行状态寄存器 mn（SSRmn）。		判断错误种类，读取值用于清除错误标志。
给串行标志清除触发寄存器 mn（SIRmn）写“1”。	清除错误标志。	通过将 SSRmn 寄存器的读取值直接写到 SIRmn 寄存器，只能清除读操作时的错误。

图 14-116 发生简易 I²C 模式中的奇偶校验错误（ACK 错误）时的处理步骤

软件操作	硬件状态	备注
读串行状态寄存器 mn（SSRmn）。		判断错误种类，读取值用于清除错误标志。
写串行标志清除触发寄存器 mn（SIRmn）。	清除错误标志。	通过将 SSRmn 寄存器的读取值直接写到 SIRmn 寄存器，只能清除读操作时的错误。
将串行通道停止寄存器 m（STm）的 STmn 位置“1”。	串行通道允许状态寄存器 m（SEm）的 SEmn 位为“0”并且通道 n 为运行停止状态。	因为没有返回 ACK，所以从属设备没有做好接收的准备。因此，生成停止条件并且释放总线，再次从开始条件开始通信，或者也能生成重新开始条件并且从地址发送开始重新进行。
生成停止条件。		
生成开始条件。		
将串行通道开始寄存器 m（SSm）的 SSmn 位置“1”。	串行通道允许状态寄存器 m（SEm）的 SEmn 位为“1”并且通道 n 为可运行状态。	

备注 m: 单元号 (m=0) n: 通道号 (n=0、2) r: IIC 号 (r=00、10) mn=00、02

第 15 章 串行接口 IICA

15.1 串行接口 IICA 的功能

串行接口 IICA 有以下 3 种模式。

(1) 运行停止模式

这是用于不进行串行传送时的模式，能降低功耗。

(2) I²C 总线模式（支持多主控）

此模式通过串行时钟（SCLAn）和串行数据总线（SDAAn）的 2 条线，与多个设备进行 8 位数据传送。

符合 I²C 总线格式，主控设备能在串行数据总线上给从属设备生成“开始条件”、“地址”、“传送方向的指示”、“数据”和“停止条件”。从属设备通过硬件自动检测接收到的状态和数据。能通过此功能简化应用程序的 I²C 总线控制部分。

因为串行接口 IICA 的 SCLAn 引脚和 SDAAn 引脚用作漏极开路输出，所以串行时钟线和串行数据总线需要上拉电阻。

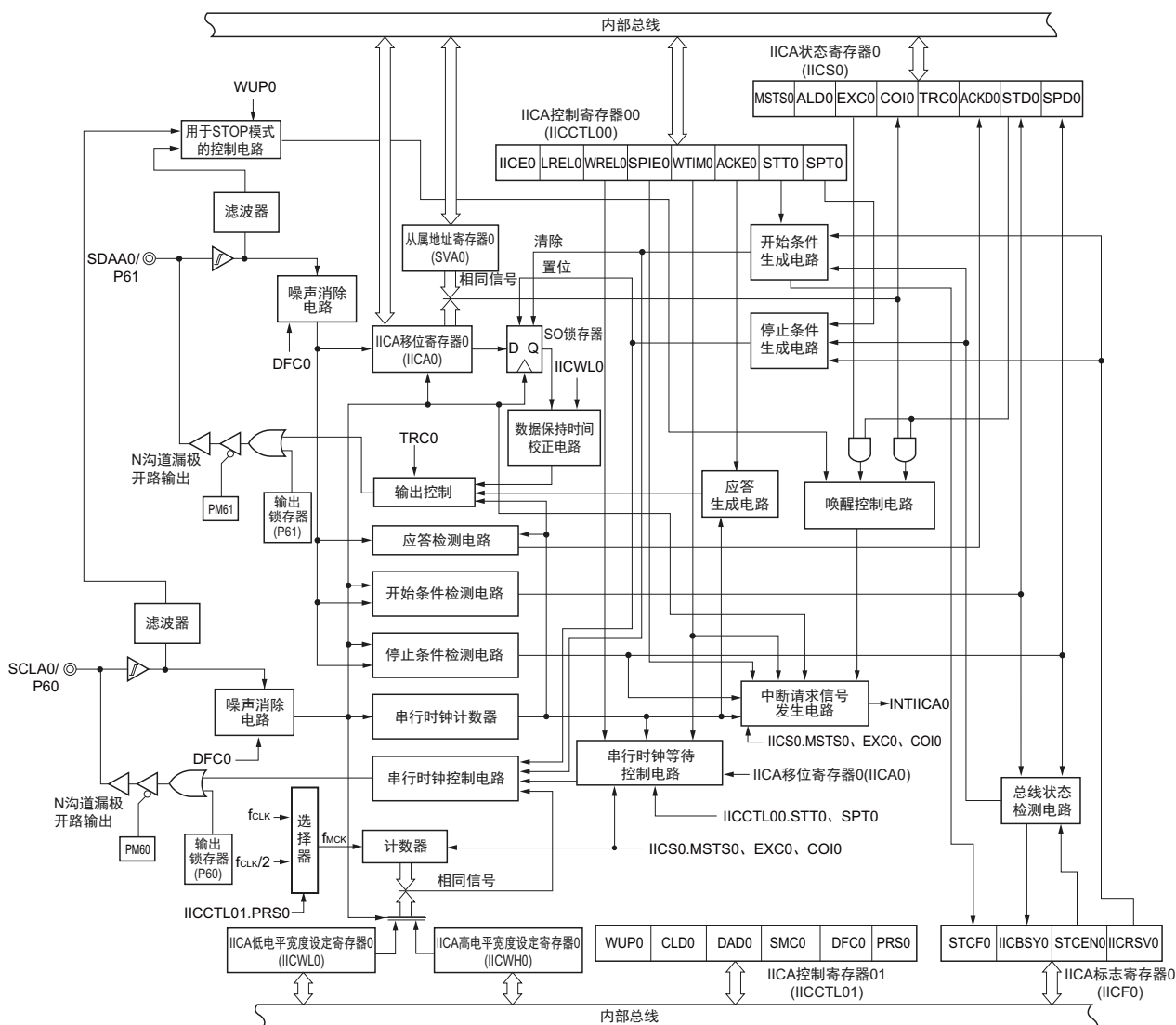
(3) 唤醒模式

在 STOP 模式中，当接收到来自主控设备的扩展码或者本地站地址时，能通过产生中断请求信号（INTIICAn）解除 STOP 模式。通过 IICA 控制寄存器 n1（IICCTLn1）的 WUPn 位进行设定。

串行接口 IICA 的框图如图 15-1 所示。

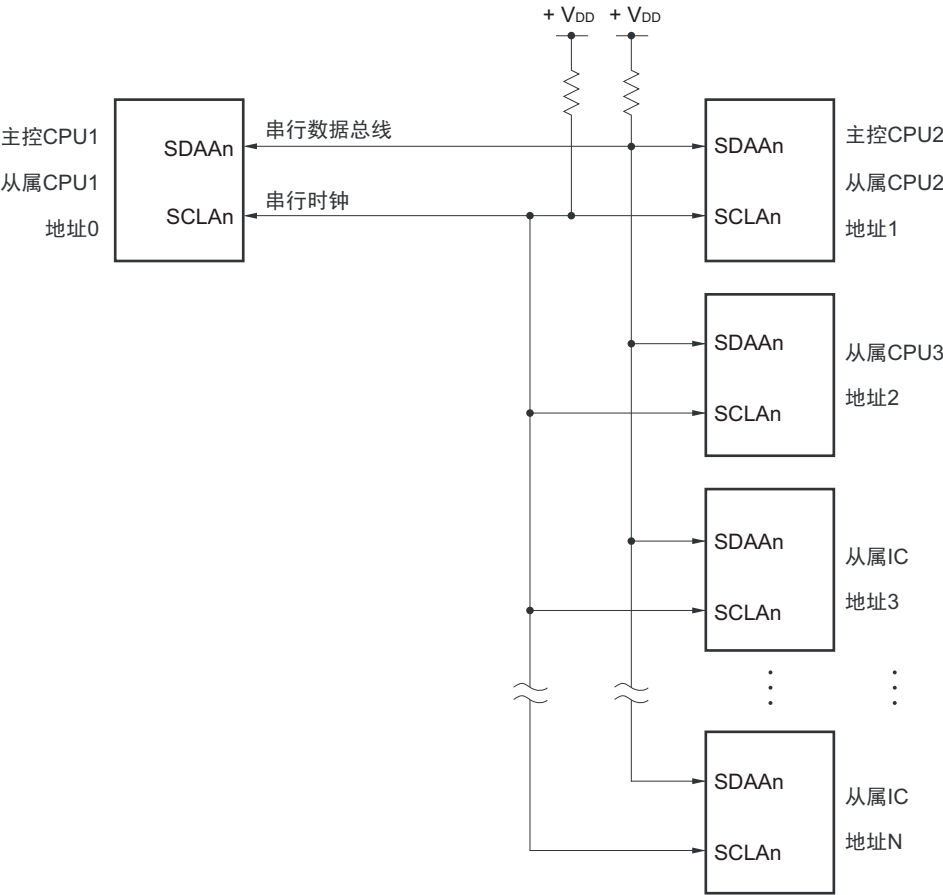
备注 n=0

图 15-1 串行接口 IICA0 的框图



串行总线的结构例子如图 15-2 所示。

图 15-2 I²C 总线的串行总线结构例子



备注 n=0

15.2 串行接口 IICA 的结构

串行接口 IICA 由以下硬件构成。

表 15-1 串行接口 IICA 的结构

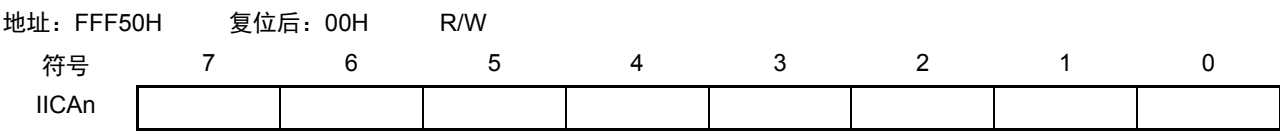
项目	结构
寄存器	IICA 移位寄存器 n (IICAn) 从属地址寄存器 n (SVAn)
控制寄存器	外围允许寄存器 0 (PER0) IICA 控制寄存器 n0 (IICCTLn0) IICA 状态寄存器 n (IICSn) IICA 标志寄存器 n (IICFn) IICA 控制寄存器 n1 (IICCTLn1) IICA 低电平宽度设定寄存器 n (IICWLn) IICA 高电平宽度设定寄存器 n (IICWHn) 端口模式寄存器 6 (PM6) 端口寄存器 6 (P6)

备注 n=0

(1) IICA 移位寄存器 n (IICAn)

IICAn 寄存器是与串行时钟同步进行 8 位串行数据和 8 位并行数据相互转换的寄存器，用于发送和接收。
能通过读写 IICAn 寄存器来控制实际的接收和发送。
在等待期间，通过写 IICAn 寄存器来解除等待，开始传送数据。
通过 8 位存储器操作指令设定 IICAn 寄存器。
在产生复位信号后，此寄存器的值变为“00H”。

图 15-3 IICAn 移位寄存器 n (IICAn) 的格式



- 注意 1. 在数据传送过程中，不能给 IICAn 寄存器写数据。
2. 只能在等待期间读写 IICAn 寄存器。除了等待期间以外，禁止在通信状态下存取 IICAn 寄存器。但是，在主控设备的情况下，能在将通信触发位 (STTn) 置“1”后写一次 IICAn 寄存器。
3. 当预约通信时，必须在检测到由停止条件产生的中断后给 IICAn 寄存器写数据。

备注 n=0

(2) 从属地址寄存器 n (SVAn)

这是在用作从属设备时保存 7 位本地站地址 {A6, A5, A4, A3, A2, A1, A0} 的寄存器。

通过 8 位存储器操作指令设定 SVAn 寄存器。但是，在 STDn 位为“1”（检测到开始条件）时，禁止改写此寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 15-4 从属地址寄存器 n (SVAn) 的格式

地址: F0234H	复位后: 00H	R/W						
符号	7	6	5	4	3	2	1	0
SVAn	A6	A5	A4	A3	A2	A1	A0	0 注

注 bit0 固定为“0”。

(3) SO 锁存器

SO 锁存器保持 SDAAn 引脚的输出电平。

(4) 唤醒控制电路

当设定在从属地址寄存器 n (SVAn) 的地址值和接收到的地址相同时或者当接收到扩展码时，此电路产生中断请求 (INTIICAn)。

(5) 串行时钟计数器

在发送或者接收过程中，此计数器对输出或者输入的串行时钟进行计数，检查是否进行了 8 位数据的发送和接收。

(6) 中断请求信号发生电路

此电路控制产生中断请求信号 (INTIICAn)。

由以下 2 种触发产生 I²C 中断请求。

- 第 8 个或者第 9 个串行时钟的下降沿（通过 WTIMn 位进行设定）
- 因检测到停止条件而产生中断请求（通过 SPIEn 位进行设定）。

备注 WTIMn 位: IICA 控制寄存器 n0 (IICCTLn0) 的 bit3

SPIEn 位: IICA 控制寄存器 n0 (IICCTLn0) 的 bit4

(7) 串行时钟控制电路

在主机模式中，此电路从采样时钟生成输出到 SCLAn 引脚的时钟。

(8) 串行时钟等待控制电路

此电路控制等待时序。

备注 n=0

(9) 应答生成电路、停止条件检测电路、开始条件检测电路、应答检测电路

这些电路生成并且检测各种状态。

(10) 数据保持时间校正电路

此电路生成对串行时钟下降的数据保持时间。

(11) 开始条件生成电路

如果将 STTn 位置“1”，此电路就生成开始条件。

但是，在禁止预约通信的状态下（IICRSVn 位 =1）并且没释放总线（IICBSYn 位 =1）时，忽视开始条件请求并且将 STCFn 位置“1”。

(12) 停止条件生成电路

如果将 SPTn 位置“1”，此电路就生成停止条件。

(13) 总线状态检测电路

此电路通过检测开始条件和停止条件来检测总线是否被释放。

但是，在刚运行时不能立即检测总线状态，因此必须通过 STCENn 位设定总线状态检测电路的初始状态。

备注 STTn 位: IICA 控制寄存器 n0 (IICCTLn0) 的 bit1
 SPTn 位: IICA 控制寄存器 n0 (IICCTLn0) 的 bit0
 IICRSVn 位: IICA 标志寄存器 n (IICFn) 的 bit0
 IICBSYn 位: IICA 标志寄存器 n (IICFn) 的 bit6
 STCFn 位: IICA 标志寄存器 n (IICFn) 的 bit7
 STCENn 位: IICA 标志寄存器 n (IICFn) 的 bit1

备注 n=0

15.3 控制串行接口 IICA 的寄存器

通过以下寄存器控制串行接口 IICA。

- 外围允许寄存器 0（PER0）
- IICA 控制寄存器 n0（IICCTLn0）
- IICA 标志寄存器 n（IICFn）
- IICA 状态寄存器 n（IICSn）
- IICA 控制寄存器 n1（IICCTLn1）
- IICA 低电平宽度设定寄存器 n（IICWLn）
- IICA 高电平宽度设定寄存器 n（IICWHn）
- 端口模式寄存器 6（PM6）
- 端口寄存器 6（P6）

备注 n=0

15.3.1 外围允许寄存器 0（PER0）

PER0 寄存器是设定允许或者禁止给各外围硬件提供时钟的寄存器。通过停止给不使用的硬件提供时钟，降低功耗和噪声。

要使用串行接口 IICAn 时，必须将 bit4（IICA0EN）置“1”。

通过 1 位或者 8 位存储器操作指令设定 PER0 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 15-5 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	SAU1EN	SAU0EN	0	TAU0EN

IICAnEN	控制串行接口 IICAn 的输入时钟的提供
0	停止提供输入时钟。 • 不能写串行接口 IICAn 使用的 SFR。 • 串行接口 IICAn 处于复位状态。
1	允许提供输入时钟。 • 能读写串行接口 IICAn 使用的 SFR。

注意 1. 在设定串行接口 IICA 时，必须在 IICAnEN 位为“1”的状态下设定以下寄存器。如果 IICAnEN 位为“0”，串行接口 IICA 的控制寄存器为初始值，并且忽视写操作（端口模式寄存器 6（PM6）和端口寄存器 6（P6）除外）。

- IICA 控制寄存器 n0（IICCTLn0）
- IICA 标志寄存器 n（IICFn）
- IICA 状态寄存器 n（IICSn）
- IICA 控制寄存器 n1（IICCTLn1）
- IICA 低电平宽度设定寄存器 n（IICWLn）
- IICA 高电平宽度设定寄存器 n（IICWHn）

2. 必须将 bit1 和 bit6 置“0”。

备注 n=0

15.3.2 IICA 控制寄存器 n0 (IICCTLn0)

这是允许或者停止 I²C 运行、设定等待时序以及设定其他 I²C 运行的寄存器。

通过 1 位或者 8 位存储器操作指令设定 IICCTLn0 寄存器。但是，必须在 IICEn 位为“0”时或者等待期间设定 SPIEn 位、WTIMn 位和 ACKEn 位，而且在将 IICEn 位从“0”置为“1”时能同时设定这些位。

在产生复位信号后，此寄存器的值变为“00H”。

备注 n=0

图 15-6 IICA 控制寄存器 n0 (IICCTLn0) 的格式 (1/4)

地址: F0230H	复位: 00H	R/W						
符号	7	6	5	4	3	2	1	0
IICCTLn0	IICEn	LRELn	WRELn	SPIEn	WTIMn	ACKEn	STTn	SPTn

IICEn	I ² C 运行的允许
0	停止运行。对 IICA 状态寄存器 n (IICSn) 进行复位 ^{注 1} , 并且停止内部运行。
1	允许运行。
必须在 SCLAn 线和 SDAAn 线为高电平的状态下将此位置“1”。	
清除条件 (IICEn=0)	置位条件 (IICEn=1)
- 通过指令清除。 - 当复位时	通过指令置位。

LRELn ^{注 2、3}	通信的退出
0	通常运行
1	退出当前的通信, 进入待机状态。执行后自动清“0”。 在接收到与本站无关的扩展码等情况时使用。 SCLAn 线和 SDAAn 线变为高阻状态。 IICA 控制寄存器 n0 (IICCTLn0) 和 IICA 状态寄存器 n (IICSn) 中的以下标志被清“0”: •STTn •SPTn •MSTSn •EXCn •COIn •TRCn •ACKDn •STDn
变为退出通信的待机状态, 直到满足以下的通信参加条件为止。	
- 在检测到停止条件后作为主控设备启动。 - 在检测到开始条件后地址匹配或者接收到扩展码。	
清除条件 (LRELn=0)	置位条件 (LRELn=1)
- 在执行后自动清除。 - 当复位时	通过指令置位。

WRELn ^{注 2、3}	等待的解除
0	不解除等待。
1	解除等待。在解除等待后自动清除。
如果在发送状态下 (TRCn=1) 的第 9 个时钟等待期间将 WRELn 位 (解除等待) 置位, SDAAn 线就变为高阻抗状态 (TRCn=0)。	
清除条件 (WRELn=0)	置位条件 (WRELn=1)
- 在执行后自动清除。 - 当复位时	通过指令置位。

- 注 1. 对 IICA 状态寄存器 n (IICA0)、IICA 标志寄存器 n (IICF0) 的 STCFn 位和 IICBSYn 位以及 IICA 控制寄存器 n1 (IICCTLn1) 的 CLDn 位和 DADn 位进行复位。
2. 在 IICEn 位为“0”的状态下, 此位的信号无效。
3. LRELn 位和 WRELn 位的读取值总是“0”。

注意 如果在 SCLAn 线为高电平、SDAAn 线为低电平并且数字滤波器为 ON (IICCTLn1 寄存器的 DFCn=1) 时允许 I²C 运行 (IICEn=1), 就立即检测开始条件。此时, 必须在允许 I²C 运行 (IICEn=1) 后连续通过 1 位存储器操作指令将 LRELn 位置“1”。

备注 n=0

图 15-6 IICA 控制寄存器 n0 (IICCTLn0) 的格式 (2/4)

SPIEn 注 1	允许或者禁止停止条件检测产生的中断请求
0	禁止
1	允许
当 IICA 控制寄存器 n1 (IICCTLn1) 的 WUPn 位为“1”时, 即使将 SPIEn 位置“1”也不产生停止条件中断。	
清除条件 (SPIEn=0)	置位条件 (SPIEn=1)
- 通过指令清除。 - 当复位时	通过指令置位。

WTIMn 注 1	等待和中断请求的控制
0	在第 8 个时钟的下降沿产生中断请求信号。 主控设备: 在输出 8 个时钟后, 将时钟输出置为低电平进行等待。 从属设备: 在输入 8 个时钟后, 将时钟置为低电平, 然后等待主控设备。
1	在第 9 个时钟的下降沿产生中断请求信号。 主控设备: 在输出 9 个时钟后, 将时钟输出置为低电平进行等待。 从属设备: 在输入 9 个时钟后, 将时钟置为低电平, 然后等待主控设备。
在地址传送期间, 与此位的设定无关, 在第 9 个时钟的下降沿产生中断; 在地址传送结束后, 此位的设定有效。主控设备在地址传送期间的第 9 个时钟下降沿进入等待。接收到本地站地址的从属设备在产生应答 (ACK) 后的第 9 个时钟下降沿进入等待, 但是接收到扩展码的从属设备在第 8 个时钟下降沿进入等待。	
清除条件 (WTIMn=0)	置位条件 (WTIMn=1)
- 通过指令清除。 - 当复位时	通过指令置位。

ACKEn 注 1、2	应答控制
0	禁止应答。
1	允许应答。在第 9 个时钟期间将 SDAAn 线置为低电平。
清除条件 (ACKEn=0)	置位条件 (ACKEn=1)
- 通过指令清除。 - 当复位时	通过指令置位。

注 1. 在 IICEn 位为“0”的状态下, 此位的信号无效。必须在此期间设定此位。

2. 在地址传送过程中并且不是扩展码时, 设定值无效。

当为从属设备并且地址匹配时, 与设定值无关而生成应答。

备注 n=0

图 15-6 IICA 控制寄存器 n0 (IICCTLn0) 的格式 (3/4)

STTn 注 1、注 2	开始条件的触发				
0	不生成开始条件。				
1	当总线被释放时（待机状态，IICBSYn 位为“0”）： 如果将此位置“1”，就生成开始条件（作为主控设备的启动）。 当第三方正在通信时： • 允许通信预约功能的情况（IICRSVn=0） 用作开始条件预约标志。如果将此位置“1”，就在释放总线后自动生成开始条件。 • 禁止通信预约功能的情况（IICRSVn=1） 即使将此位置“1”，也清除 STTn 位并且将 STTn 清除标志（STCFn）置“1”，不生成开始条件。 等待状态（主控设备）： 在解除等待后生成重新开始条件。				
有关置位时序的注意事项： • 主控接收：禁止在传送过程中将此位置“1”。只有在将 ACKEn 位置“0”并且通知从属设备接收已经完成后的等待期间才能将此位置“1”。 • 主控发送：在应答期间，可能不能正常生成开始条件。必须在输出第 9 个时钟后的等待期间将此位置“1”。 • 禁止与停止条件的触发（SPTn）同时置“1”。 • 在将 STTn 位置“1”后，禁止在满足清除条件前再次将此位“1”。					
<table border="1"> <thead> <tr> <th>清除条件（STTn=0）</th><th>置位条件（STTn=1）</th></tr> </thead> <tbody> <tr> <td> • 在禁止通信预约的状态下将 STTn 位置“1”。 • 在仲裁失败时 • 主控设备生成开始条件。 • 因 LRELn 位为“1”（退出通信）而进行的清除 • 当 IICEn 位为“0”（停止运行）时 • 当复位时 </td><td> • 通过指令置位。 </td></tr> </tbody> </table>		清除条件（STTn=0）	置位条件（STTn=1）	• 在禁止通信预约的状态下将 STTn 位置“1”。 • 在仲裁失败时 • 主控设备生成开始条件。 • 因 LRELn 位为“1”（退出通信）而进行的清除 • 当 IICEn 位为“0”（停止运行）时 • 当复位时	• 通过指令置位。
清除条件（STTn=0）	置位条件（STTn=1）				
• 在禁止通信预约的状态下将 STTn 位置“1”。 • 在仲裁失败时 • 主控设备生成开始条件。 • 因 LRELn 位为“1”（退出通信）而进行的清除 • 当 IICEn 位为“0”（停止运行）时 • 当复位时	• 通过指令置位。				

注 1. 在 IICEn 位为“0”的状态下，此位的信号无效。
2. STTn 位的读取值总是为“0”。

备注 1. 在设定数据后，bit1（STTn）的读取值为“0”。
2. IICRSVn：IIC 标志寄存器 n（IICFn）的 bit0
STCFn：IIC 标志寄存器 n（IICFn）的 bit7
3. n=0

图 15-6 IICA 控制寄存器 n0 (IICCTLn0) 的格式 (4/4)

SPTn 注	停止条件的触发
0	不生成停止条件。
1	生成停止条件（作为主控设备的传送结束）。
有关置位时序的注意事项：	
• 主控接收：禁止在传送过程中将此位置“1”。只有在将 ACKEn 位置“0”并且通知从属设备接收已经完成后的等待期间才能将此位置“1”。 • 主控发送：在应答期间，可能不能正常生成停止条件。必须在输出第 9 个时钟后的等待期间将此位置位。 • 禁止与开始条件的触发（STTn）同时置“1”。 • 只有在主控设备的情况下才能将 SPTn 位置“1”。 • 在 WTIMn 位为“0”时，必须注意：如果在输出 8 个时钟后的等待期间将 SPTn 位置“1”，就在解除等待后的第 9 个时钟的高电平期间生成停止条件。必须在输出 8 个时钟后的等待期间将 WTIMn 位从“0”置为“1”并且在输出第 9 个时钟后的等待期间将 SPTn 位置“1”。 • 在将 SPTn 位置“1”后，禁止在满足清除条件前再次将此位置“1”。	
清除条件（SPTn=0）	置位条件（SPTn=1）
• 当仲裁失败时 • 在检测到停止条件后自动清除。 • 因 LRELn 位为“1”（退出通信）而进行的清除 • 当 IICEn 位为“0”（停止运行）时 • 当复位时	• 通过指令置位。

注 SPTn 的读取值总是为“0”。

注意 在 IICA 状态寄存器 n (IICSn) 的 bit3 (TRCn) 为“1” (发送状态) 时，如果在第 9 个时钟将 IICCTLn0 寄存器的 bit5 (WRELn) 置“1”来解除等待，就在清除 TRCn 位 (接收状态) 后将 SDAAn 线置为高阻抗。必须通过写 IICA 移位寄存器 n 进行 TRCn 位为“1” (发送状态) 时的等待解除。

备注 1. 如果在设定数据后读 bit0 (SPTn)，此位就变为“0”。

2. n=0

15.3.3 IICA 状态寄存器 n（IICSn）

这是表示 I²C 状态的寄存器。
只有在 STTn 位为“1”并且等待期间，才能通过 1 位或者 8 位存储器操作指令读 IICSn 寄存器。
在产生复位信号后，此寄存器的值变为“00H”。

注意 在 STOP 模式中允许地址匹配唤醒功能（WUPn=1）状态下，禁止读 IICSn 寄存器。在 WUPn 位为“1”的状态下，与 INTIICAn 中断请求无关，如果将 WUPn 位从“1”改为“0”（停止唤醒运行），就在检测到下一个开始条件或者停止条件后才会反映状态的变化。因此，要使用唤醒功能时，必须允许（SPIEn=1）因检测到停止条件而产生的中断，并且在检测到中断后读 IICSn 寄存器。

备注 STTn: IICA 控制寄存器 n0（IICCTLn0）的 bit1
WUPn: IICA 控制寄存器 n1（IICCTLn1）的 bit7

图 15-7 IICA 状态寄存器 n（IICSn）的格式（1/3）



注 即使对 IICSn 寄存器以外的位执行 1 位存储器操作指令，也清除此位。因此，在使用 ALDn 位时，必须在读其他位前先读 ALDn 位的数据。

备注 1. LRELn: IICA 控制寄存器 n0（IICCTLn0）的 bit6
IICEn: IICA 控制寄存器 n0（IICCTLn0）的 bit7
2. n=0

图 15-7 IICA 状态寄存器 n (IICSn) 的格式 (2/3)

EXCn	扩展码的接收检测	
0	未接收到扩展码。	
1	接收到扩展码。	
清除条件 (EXCn=0)		置位条件 (EXCn=1)
- 当检测到开始条件时 - 当检测到停止条件时 - 因 LRELn 位为“1” (退出通信) 而进行的清除 - 当 IICEn 位从“1”变为“0” (停止运行) 时 - 当复位时		当接收的地址数据的高 4 位为“0000”或者“1111”时 (在第 8 个时钟的上升沿进行置位)。

COIn	匹配地址的检测	
0	地址不同。	
1	地址相同。	
清除条件 (COIn=0)		置位条件 (COIn=1)
- 当检测到开始条件时 - 当检测到停止条件时 - 因 LRELn 位为“1” (退出通信) 而进行的清除 - 当 IICEn 位从“1”变为“0” (停止运行) 时 - 当复位时		当接收地址和本地站地址 (从属地址寄存器 n (SVAn)) 相同时 (在第 8 个时钟的上升沿置位)。

TRCn	发送 / 接收的状态检测	
0	接收状态 (发送状态除外)。将 SDAAn 线置为高阻抗。	
1	发送状态。设定为能将 SOn 锁存器的值输出到 SDAAn 线 (在第 1 字节的第 9 个时钟的下降沿以后有效)。	
清除条件 (TRCn=0)		置位条件 (TRCn=1)
<主控设备和从属设备> - 当检测到停止条件时 - 因 LRELn 位为“1” (退出通信) 而进行的清除 - 当 IICEn 位从“1”变为“0” (停止运行) 时 - 因 WRELn 位为“1” (解除等待) 而进行的清除注 - 当 ALDn 位从“0”变为“1” (仲裁失败) 时 - 当复位时 - 不参加通信的情况 (MSTSn、EXCn、COIn=0) <主控设备> - 当第 1 字节的 LSB (传送方向指示位) 输出“1”时。 <从属设备> - 当检测到开始条件时 - 当第 1 字节的 LSB (传送方向指示位) 输入“0”时		<主控设备> - 当生成开始条件时 - 当将第 1 字节 (地址传送) 的 LSB (传送方向指示位) 输出“0” (主控发送) 时 <从属设备> - 当来自主控设备的第 1 字节 (地址传送) 的 LSB (传送方向指示位) 输入“1” (从属发送) 时

注 在 IICA 状态寄存器 n (IICSn) 的 bit3 (TRCn) 为“1” (发送状态) 时, 如果在第 9 个时钟将 IICA 控制寄存器 n0 (IICCTLn0) 的 bit5 (WRELn) 置“1”来解除等待, 就在清除 TRCn 位 (接收状态) 后将 SDAAn 线置为高阻抗。必须通过写 IICA 移位寄存器 n 进行 TRCn 位为“1” (发送状态) 时的等待解除。

备注 1. LRELn: IICA 控制寄存器 n0 (IICCTLn0) 的 bit6

IICEn: IICA 控制寄存器 n0 (IICCTLn0) 的 bit7

2. n=0

图 15-7 IICA 状态寄存器 n (IICSn) 的格式 (3/3)

ACKDn	应答 (ACK) 的检测
0	未检测到应答。
1	检测到应答。
清除条件 (ACKDn=0)	置位条件 (ACKDn=1)
- 当检测到停止条件时 - 当下一字节的第 1 个时钟上升时 - 因 LRELn 位为“1” (退出通信) 而进行的清除 - 当 IICEn 位从“1”变为“0” (停止运行) 时 - 当复位时	在 SCLAn 线的第 9 个时钟上升沿将 SDAAn 线为低电平时

STDn	开始条件的检测
0	未检测到开始条件。
1	检测到开始条件, 表示处于地址传送期间。
清除条件 (STDn=0)	置位条件 (STDn=1)
- 当检测到停止条件时 - 在地址传送后的下一个字节的第 1 个时钟上升时 - 因 LRELn 位为“1” (退出通信) 而进行的清除 - 当 IICEn 位从“1”变为“0” (停止运行) 时 - 当复位时	当检测到开始条件时

SPDn	停止条件的检测
0	未检测到停止条件。
1	检测到停止条件, 主控设备结束通信并且已释放总线。
清除条件 (SPDn=0)	置位条件 (SPDn=1)
- 在将此位置位后, 在检测到开始条件后的地址传送位的第 1 个时钟上升时 - 当 WUPn 位从“1”变为“0”时 - 当 IICEn 位从“1”变为“0” (停止运行) - 当复位时	当检测到停止条件时

备注 1. LRELn: IICA 控制寄存器 n0 (IICCTLn0) 的 bit6

IICEn: IICA 控制寄存器 n0 (IICCTLn0) 的 bit7

2. n=0

15.3.4 IICA 标志寄存器 n (IICFn)

这是设定 I²C 运行模式以及表示 I²C 总线状态的寄存器。

通过 1 位或者 8 位存储器操作指令设定 IICFn 寄存器。但是, STTn 清除标志 (STCFn) 和 I²C 总线状态标志 (IICBSYn) 位为只读位。

通过 IICRSVn 位设定允许或者禁止通信预约功能, 并且通过 STCENn 位设定 IICBSYn 位的初始值。

只有在禁止 I²C 运行 (IICA 控制寄存器 n0 (IICCTLn0) 的 bit7 (IICEn) =0) 时才能写 IICRSVn 位和 STCENn 位。在允许运行后, IICFn 寄存器为只读寄存器。

在产生复位信号后, 此寄存器的值变为“00H”。

图 15-8 IICA 标志寄存器 n (IICFn) 的格式

地址: FFF52H 复位后: 00H R/W 注

符号	7	6	5	4	3	2	1	0
IICFn	STCFn	IICBSYn	0	0	0	0	STCENn	IICRSVn

STCFn	STTn 清除标志
0	发行开始条件。
1	无法发行开始条件而清除 STTn 标志。
清除条件 (STCFn=0)	
- 因 STTn 位为“1”而进行的清除 - 当 IICEn 位为“0” (停止运行) 时 - 当复位时	
置位条件 (STCFn=1)	
在设定为禁止通信预约 (IICRSVn=1) 的状态下无法发行开始条件而将 STTn 位清“0”时	

IICBSYn	I ² C 总线状态标志
0	总线释放状态 (STCENn=1 时的通信初始状态)
1	总线通信状态 (STCENn=0 时的通信初始状态)
清除条件 (IICBSYn=0)	
- 当检测到停止条件时 - 当 IICEn 位为“0” (停止运行) 时 - 当复位时	
置位条件 (IICBSYn=1)	
- 当检测到开始条件时 - STCENn 位为“0”时的 IICEn 位的置位	

STCENn	初始开始允许触发
0	在允许运行 (IICEn=1) 后, 通过检测停止条件而允许生成开始条件。
1	在允许运行 (IICEn=1) 后, 不通过检测停止条件而允许而生成开始条件。
清除条件 (STCENn=0)	
- 通过指令清除。 - 当检测到开始条件时 - 当复位时	
置位条件 (STCENn=1)	
通过指令置位。	

IICRSVn	通信预约功能禁止位
0	允许通信预约。
1	禁止通信预约。
清除条件 (IICRSVn=0)	
- 通过指令清除。 - 当复位时	
置位条件 (IICRSVn=1)	
通过指令置位。	

注 bit6 和 bit7 是只读位。

注意 1. 只有在停止运行 (IICEn=0) 时才能写 STCENn 位。

2. 如果 STCENn 位为“1”, 就与实际的总线状态无关而认为总线为释放状态 (IICBSYn=0), 因此为了避免在发行第 1 个开始条件 (STTn=1) 时破坏其他通信, 必须确认没有正在通信的第三方。

3. 只有在停止运行 (IICEn=0) 时才能写 IICRSVn。

备注 1. STTn: IICA 控制寄存器 n0 (IICCTLn0) 的 bit1

2. IICEn: IICA 控制寄存器 n0 (IICCTLn0) 的 bit7

15.3.5 IICA 控制寄存器 n1 (IICCTLn1)

这是用于设定 I²C 运行模式以及检测 SCLAn 引脚和 SDAAn 引脚状态的寄存器。

通过 1 位或者 8 位存储器操作指令设定 IICCTLn1 寄存器。但是，CLDn 位和 DADn 位是只读位。

除了 WUPn 位以外，必须在禁止 I²C 运行（IICA 控制寄存器 n0 (IICCTLn0) 的 bit7 (IICEn) =0) 时设定 IICCTLn1 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

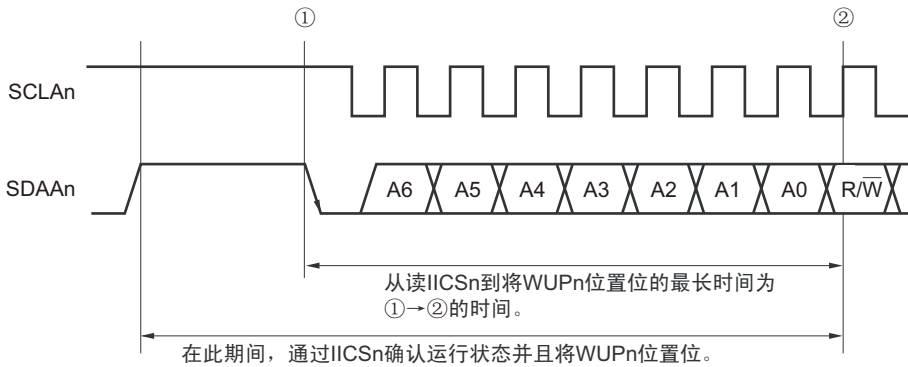
图 15-9 IICA 控制寄存器 n1 (IICCTLn1) 的格式 (1/2)

地址: F0231H	复位后: 00H	R/W 注 1						
符号	7	6	5	4	3	2	1	0
IICCTLn1	WUPn	0	CLDn	DADn	SMCn	DFCn	0	PRSn

WUPn	地址匹配唤醒的控制
0	在 STOP 模式中，停止地址匹配唤醒功能的运行。
1	在 STOP 模式中，允许地址匹配唤醒功能的运行。
要通过将 WUPn 位置“1”来转移到 STOP 模式时，必须在将 WUPn 位置“1”后至少经过 3 个 f _{MCK} 时钟，然后执行 STOP 指令（参照“图 15-22 将 WUPn 位置“1”时的流程”）。	
在地址匹配或者接收到扩展码后，必须将 WUPn 位清“0”。能通过将 WUPn 位清“0”参加后续的通信（需要在将 WUPn 位清“0”后解除等待以及写发送数据）。	
在 WUPn 位为“1”的状态下，地址匹配或者接收到扩展码时的中断时序与 WUPn 位为“0”时的中断时序相同（根据时钟产生采样误差的延迟差）。另外，当 WUPn 位为“1”时，即使将 SPIEn 位置“1”也不产生停止条件中断。	
清除条件（WUPn=0）	置位条件（WUPn=1）
• 通过指令清除（在地址匹配或者接收到扩展码后）。	• 通过指令置位（MSTSn=0、EXCn=0、COIn=0 并且 STDn=0（不参加通信））注 2。

注 1. bit4 和 bit5 是只读位。

2. 在以下所示的期间，需要确认 IICA 状态寄存器 n (IICCSn) 的状态。



备注 n=0

图 15-9 IICA 控制寄存器 n1 (IICCTLn1) 的格式 (2/2)

CLDn	SCLAn 引脚的电平检测 (只在 IICEn 位为“1”时有效)
0	检测到 SCLAn 引脚为低电平。
1	检测到 SCLAn 引脚为高电平。
清除条件 (CLDn=0)	
置位条件 (CLDn=1)	
- 当 SCLAn 引脚为低电平时 - 当 IICEn 位为“0” (停止运行) 时 - 当复位时	当 SCLAn 引脚为高电平时

DADn	SDAAn 引脚的电平检测 (只在 IICEn 位为“1”时有效)
0	检测到 SDAAn 引脚为低电平。
1	检测到 SDAAn 引脚为高电平。
清除条件 (DADn=0)	
置位条件 (DADn=1)	
- 当 SDAAn 引脚为低电平时 - 当 IICEn 位为“0” (停止运行) 时 - 当复位时	当 SDAAn 引脚为高电平时

SMCn	运行模式的切换
0	在标准模式中运行 (最大传送速率: 100kbps)。
1	在快速模式 (最大传送速率: 400kbps) 或者增强型快速模式 (最大传送速率: 1Mbps) 中运行。

DFCn	数字滤波器的运行控制
0	数字滤波器 OFF
1	数字滤波器 ON
只有在快速模式或者增强型快速模式中才能使用数字滤波器。 在快速模式或者增强型快速模式中, 无论是将 DFCn 位置“1”还是清“0”, 传送时钟都不变。 在快速模式或者增强型快速模式中, 数字滤波器用于消除噪声。	

PRSn	IICA 运行时钟 (f_{MCK}) 的控制
0	选择 f_{CLK} ($1MHz \leq f_{CLK} \leq 20MHz$)
1	选择 $f_{CLK}/2$ ($20MHz < f_{CLK}$)

注意 1. IICA 运行时钟 (f_{MCK}) 的最大工作频率为 20MHz(Max.)。

只有在 f_{CLK} 大于 20MHz 时, 将 IICA 控制寄存器 n1 (IICCTLn1) 的 bit0 (PRSn) 设定为“1”。

2. 在设定传送时钟时, 必须注意 f_{CLK} 的最小工作频率。串行接口 IICA 的 f_{CLK} 的最小工作频率取决于模式。

快速模式中: $f_{CLK}=3.5MHz(\text{Min.})$

增强型快速模式中: $f_{CLK}=10MHz(\text{Min.})$

标准模式中: $f_{CLK}=1MHz(\text{Min.})$

3. 增强型快速模式只用于 A: 民用产品 ($T_A=-40 \sim +85^{\circ}\text{C}$)。

备注 1. IICEn: IICA 控制寄存器 n0 (IICCTLn0) 的 bit7

2. n=0

15.3.6 IICA 低电平宽度设定寄存器 n (IICWLn)

此寄存器控制串行接口 IICA 输出的 SCLAn 引脚信号低电平宽度 (t_{LOW})。

通过 8 位存储器操作指令设定 IICWLn 寄存器。

必须在禁止 I²C 运行 (IICA 控制寄存器 n0 (IICCTLn0) 的 bit7 (IICEn) =0) 时设定 IICWLn 寄存器。

在产生复位信号后, 此寄存器的值变为“FFH”。

有关 IICWLn 寄存器的设定方法, 请参照“15.4.2 通过 IICWLn 寄存器和 IICWHn 寄存器设定传送时钟的方法”。

图 15-10 IICA 低电平宽度设定寄存器 n (IICWLn) 的格式

地址: F0232H	复位后: FFH	R/W						
符号	7	6	5	4	3	2	1	0
IICWLn								

15.3.7 IICA 高电平宽度设定寄存器 n (IICWHn)

此寄存器控制串行接口 IICA 输出的 SCLAn 引脚信号高电平宽度。

通过 8 位存储器操作指令设定 IICWHn 寄存器。

必须在禁止 I²C 运行 (IICA 控制寄存器 n0 (IICCTLn0) 的 bit7 (IICEn) =0) 时设定 IICWHn 寄存器。

在产生复位信号后, 此寄存器的值变为“FFH”。

图 15-11 IICA 高电平宽度设定寄存器 n (IICWHn) 的格式

地址: F0233H	复位后: FFH	R/W						
符号	7	6	5	4	3	2	1	0
IICWHn								

备注 1. 有关主控方的传送时钟的设定方法, 请参照“15.4.2(1) 主控方传送时钟的设定方法”, 有关从属方的 IICWLn、IICWHn 寄存器的设定方法, 请参照“15.4.2(2) 从属方 IICWLn 寄存器和 IICWHn 寄存器的设定方法”。

2. n=0

15.3.8 端口模式寄存器 6（PM6）

此寄存器以位为单位设定端口 6 的输入 / 输出。

在将 P60/SCLA0 引脚用作时钟输入 / 输出并且将 P61/SDAA0 引脚用作串行数据输入 / 输出时，必须将 PM60 和 PM61 以及 P60 和 P61 的输出锁存器置“0”。

当 IICEn 位（IICA 控制寄存器 n0（IICCTLn0）的 bit7）为“0”时，P60/SCLA0 引脚和 P61/SDAA0 引脚为低电平输出（固定），因此必须在将 IICEn 位置“1”后切换到输出模式。

通过 1 位或者 8 位存储器操作指令设定 PM6 寄存器。

在产生复位信号后，此寄存器的值变为“FFH”。

图 15-12 端口模式寄存器 6（PM6）的格式

地址：FFF26H	复位后：FFH	R/W						
符号	7	6	5	4	3	2	1	0
PM6	1	1	1	1	1	1	PM61	PM60

PM6n	P6n 引脚输入 / 输出模式的选择（n=0、1）
0	输出模式（输出缓冲器 ON）
1	输入模式（输出缓冲器 OFF）

15.4 I²C 总线模式的功能

15.4.1 引脚结构

串行时钟引脚（SCLAn）和串行数据总线引脚（SDAAn）的结构如下。

(1) SCLAn..... 串行时钟的输入/输出引脚

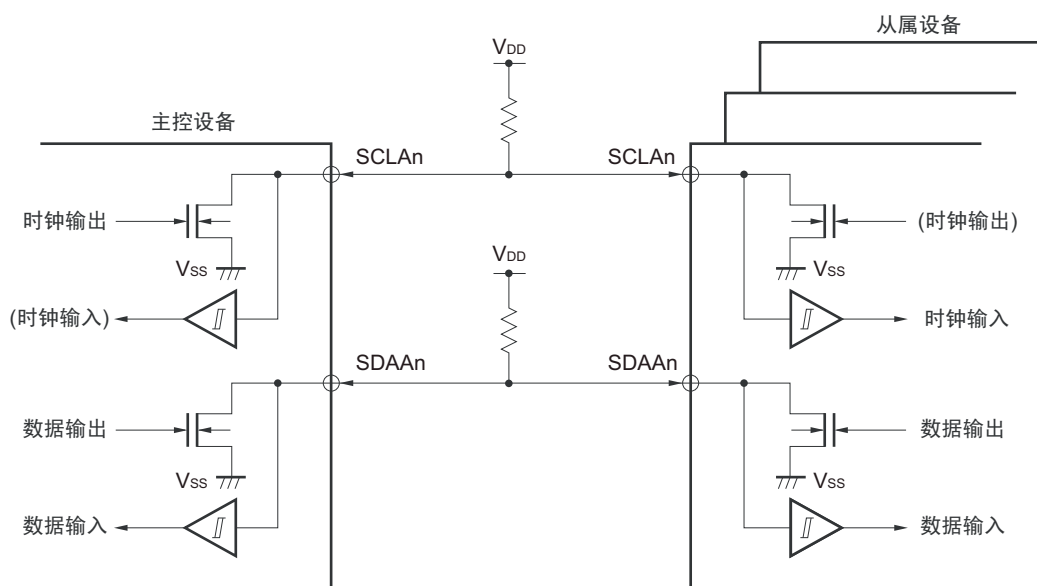
主控设备和从属设备的输出都为N沟道漏极开路输出，输入都为施密特输入。

(2) SDAAn..... 串行数据的输入/输出复用引脚

主控设备和从属设备的输出都为N沟道漏极开路输出，输入都为施密特输入。

因为串行时钟线和串行数据总线的输出为 N 沟道漏极开路输出，所以需要外接上拉电阻。

图 15-13 引脚结构图



备注 n=0

15.4.2 通过 IICWL_n 寄存器和 IICWH_n 寄存器设定传送时钟的方法

(1) 主控方传送时钟的设定方法

$$\text{传送时钟} = \frac{f_{\text{MCK}}}{\text{IICWL} + \text{IICWH} + f_{\text{CLK}}(t_{\text{R}} + t_{\text{F}})}$$

此时，IICWL_n 寄存器和 IICWH_n 寄存器的最佳设定值如下：
(全部设定值的小数部分都舍入)

- 快速模式

$$\text{IICWL}_n = \frac{0.52}{\text{传送时钟}} \times f_{\text{MCK}}$$

$$\text{IICWH}_n = \left(\frac{0.48}{\text{传送时钟}} - t_{\text{R}} - t_{\text{F}} \right) \times f_{\text{MCK}}$$

- 标准模式

$$\text{IICWL}_n = \frac{0.47}{\text{传送时钟}} \times f_{\text{MCK}}$$

$$\text{IICWH}_n = \left(\frac{0.53}{\text{传送时钟}} - t_{\text{R}} - t_{\text{F}} \right) \times f_{\text{MCK}}$$

- 增强型快速模式

$$\text{IICWL}_n = \frac{0.50}{\text{传送时钟}} \times f_{\text{MCK}}$$

$$\text{IICWH}_n = \left(\frac{0.50}{\text{传送时钟}} - t_{\text{R}} - t_{\text{F}} \right) \times f_{\text{MCK}}$$

(2) 从属方 IICWL_n 寄存器和 IICWH_n 寄存器的设定方法

(全部设定值的小数部分都舍入)

- 快速模式

$$\text{IICWL}_n = 1.3\mu\text{s} \times f_{\text{MCK}}$$

$$\text{IICWH}_n = (1.2\mu\text{s} - t_R - t_F) \times f_{\text{MCK}}$$
- 标准模式

$$\text{IICWL}_n = 4.7\mu\text{s} \times f_{\text{MCK}}$$

$$\text{IICWH}_n = (5.3\mu\text{s} - t_R - t_F) \times f_{\text{MCK}}$$
- 增强型快速模式

$$\text{IICWL}_n = 0.50\mu\text{s} \times f_{\text{MCK}}$$

$$\text{IICWH}_n = (0.50\mu\text{s} - t_R - t_F) \times f_{\text{MCK}}$$

注意 1. 串行接口 IICA 运行时钟 (f_{MCK}) 的最大工作频率为 20MHz(Max.)。

只有在 f_{CLK} 大于 20MHz 时, 将 IICA 控制寄存器 n1 (IICCTLn1) 的 bit0 (PRSn) 位设定为“1”。

2. 在设定传送时钟的情况下, 必须注意 f_{CLK} 的最小工作频率。串行接口 IICA 的 f_{CLK} 最小工作频率取决于运行模式。

快速模式: $f_{\text{CLK}}=3.5\text{MHz}(\text{Min.})$

增强型快速模式: $f_{\text{CLK}}=10\text{MHz}(\text{Min.})$

标准模式: $f_{\text{CLK}}=1\text{MHz}(\text{Min.})$

备注 1. 因为 SDAAn 信号和 SCLAn 信号的上升时间 (t_R) 和下降时间 (t_F) 因上拉电阻和布线负载而不同, 所以必须各自计算。

2. IICWL_n: IICA 低电平宽度设定寄存器 n

IICWH_n: IICA 高电平宽度设定寄存器 n

t_F : SDAAn 信号和 SCLAn 信号的下降时间

t_R : SDAAn 信号和 SCLAn 信号的上升时间

f_{MCK} : IICA 运行时钟频率

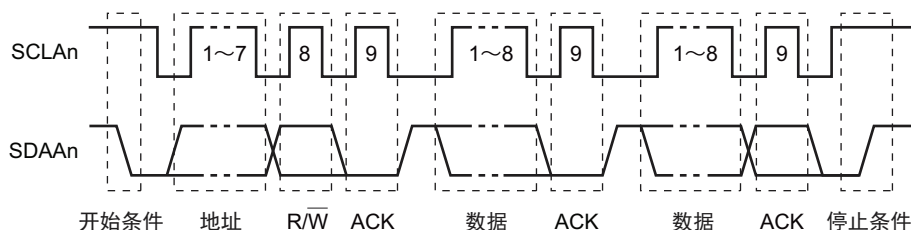
3. n=0

15.5 I²C 总线的定义和控制方法

以下说明 I²C 总线的串行数据通信格式和使用的信号。

I²C 总线的串行数据总线上生成的“开始条件”、“地址”、“数据”和“停止条件”的各传送时序如图 15-14 所示。

图 15-14 I²C 总线的串行数据传送时序



主控设备生成开始条件、从属地址和停止条件。

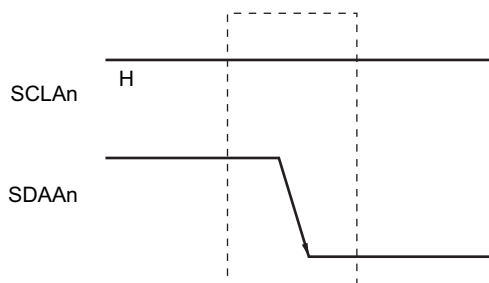
主控设备和从属设备都能生成应答（ACK）（在一般情况下，接收方输出 8 位数据）。

主控设备连续输出串行时钟（SCLAn）。但是，从属设备能延长 SCLAn 引脚的低电平期间并且插入等待。

15.5.1 开始条件

在 SCLAn 引脚为高电平时，如果 SDAAn 引脚从高电平变为低电平，就生成开始条件。SCLAn 引脚和 SDAAn 引脚的开始条件是在主控设备对从属设备开始串行传送时生成的信号。从属设备能检测到开始条件。

图 15-15 开始条件



在检测到停止条件（SPDn: IICA 状态寄存器 n（IICSn）的 bit0=1）的状态下，如果将 IICA 控制寄存器 n0（IICCTLn0）的 bit1（STTn）置“1”，就输出开始条件。如果检测到开始条件，就将 IICSn 寄存器的 bit1（STDn）置“1”。

备注 n=0

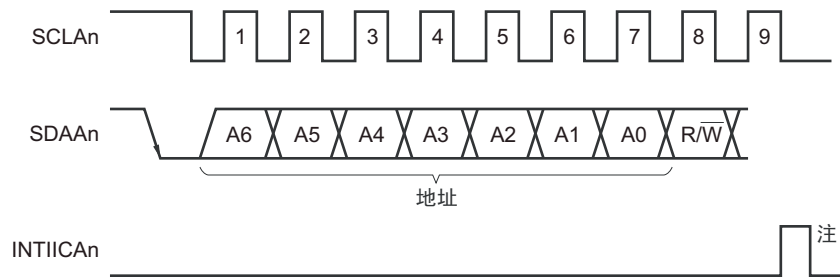
15.5.2 地址

开始条件的后续 7 位数据被定义为地址。

地址是主控设备为了从连接在总线的多个从属设备中选择特定的从属设备而输出的 7 位数据。因此，总线上的从属设备需要设定完全不同的地址。

从属设备通过硬件检测到开始条件，并且检查 7 位数据是否和从属地址寄存器 n（SVAn）的内容相同。此时，如果 7 位数据和 SVAn 寄存器的值相同，该从属设备就被选中，在 主控设备生成开始条件或者停止条件前，与主控设备进行通信。

图 15-16 地址



注 如果在从属设备运行时接收到本地站地址或者扩展码以外的数据，就不产生 INTIICAn。

如果将从属地址和“15.5.3 传送方向的指定”中说明的传送方向构成的 8 位数据写到 IICA 移位寄存器 n（IICAn），就输出地址。接收到的地址被写到 IICAn 寄存器。

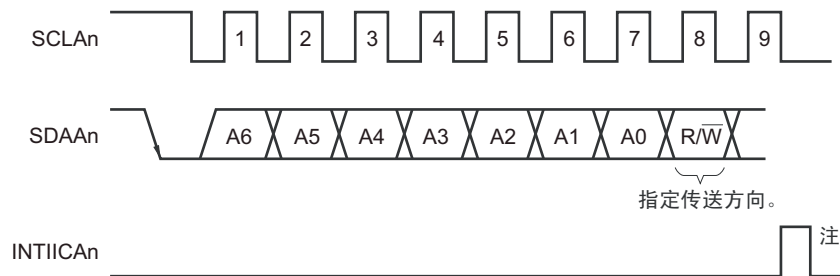
从属地址分配在 IICAn 寄存器的高 7 位。

15.5.3 传送方向的指定

主控设备在 7 位地址之后发送 1 位指定传送方向的数据。

当此传送方向指定位为“0”时，表示主控设备向从属设备发送数据；当此传送方向指定位为“1”时，表示主控设备从从属设备接收数据。

图 15-17 传送方向的指定



注 如果在从属设备运行时接收到本地站地址或者扩展码以外的数据，就不产生 INTIICAn。

备注 n=0

15.5.4 应答 (ACK)

能通过应答 (ACK) 确认发送方和接收方的串行数据状态。

接收方在每次接收到 8 位数据时返回应答。

通常, 发送方在发送 8 位数据后接收应答。当接收方返回应答时, 认为已正常接收, 继续处理。能通过 IICA 状态寄存器 n (IICSn) 的 bit2 (ACKDn) 确认应答的检测。

在主机设备为接收状态下接收到最后的数据时, 不返回应答而生成停止条件。在从属设备接收数据后不返回应答时, 主机设备输出停止条件或者重新开始条件, 中止传送。不返回应答的原因如下:

- ① 没有正常接收。
- ② 已结束最后数据的接收。
- ③ 不存在地址指定的接收方。

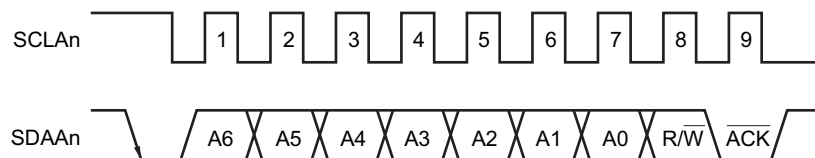
接收方在第 9 个时钟将 SDAAn 线置为低电平, 生成应答 (正常接收)。

通过将 IICA 控制寄存器 n0 (IICCTLn0) 的 bit2 (ACKEn) 置“1”, 变为能够自动生成应答的状态。能通过 7 位地址信息后续的第 8 位数据设定 IICSn 寄存器的 bit3 (TRCn)。在接收 (TRCn=0) 的情况下, 通常必须将 ACKEn 位置“1”。

在从属接收运行过程中 (TRCn=0) 不能接收数据或者不需要下一个数据时, 必须将 ACKEn 位清“0”, 通知主机方不能接收数据。

在主机接收运行过程中 (TRCn=0) 不需要下一个数据时, 为了不生成应答, 必须将 ACKEn 位清“0”, 通知从属发送方数据的结束 (停止发送)。

图 15-18 应答



当接收到本地站的地址时, 与 ACKEn 位的值无关, 自动生成应答; 当接收到非本地站的地址时, 不生成应答 (NACK)。

通过事先将 ACKEn 位置“1”, 在接收到扩展码时生成应答。

接收数据时的应答生成方法因等待时序的设定而不同, 如下所示。

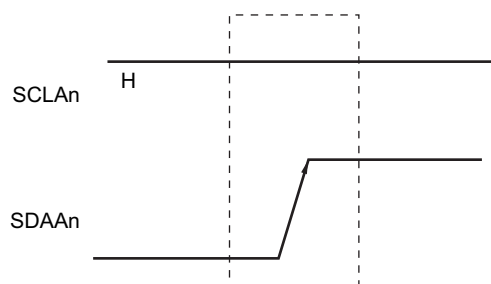
- 当选择 8 个时钟等待时 (IICCTLn0 寄存器的 bit3 (WTIMn) =0):
通过在解除等待前将 ACKEn 位置“1”, 与 SCLAn 引脚的第 8 个时钟下降同步生成应答。
- 当选择 9 个时钟等待时 (IICCTLn0 寄存器的 bit3 (WTIMn) =1):
通过事先将 ACKEn 位置“1”, 生成应答。

备注 n=0

15.5.5 停止条件

在 SCLAn 引脚为高电平时，如果 SDAAn 引脚从低电平变为高电平，就生成停止条件。
停止条件是在主控设备结束对从属设备的串行传送时生成的信号。从属设备能检测到停止条件。

图 15-19 停止条件



如果将 IICA 控制寄存器 n0 (IICCTLn0) 的 bit0 (SPTn) 置“1”，就生成停止条件。如果检测到停止条件，就将 IICA 状态寄存器 n (IICSn) 的 bit0 (SPDn) 置“1”，并且在 IICCTLn0 寄存器的 bit4 (SPIEn) 为“1”时产生 INTIICAn。

备注 n=0

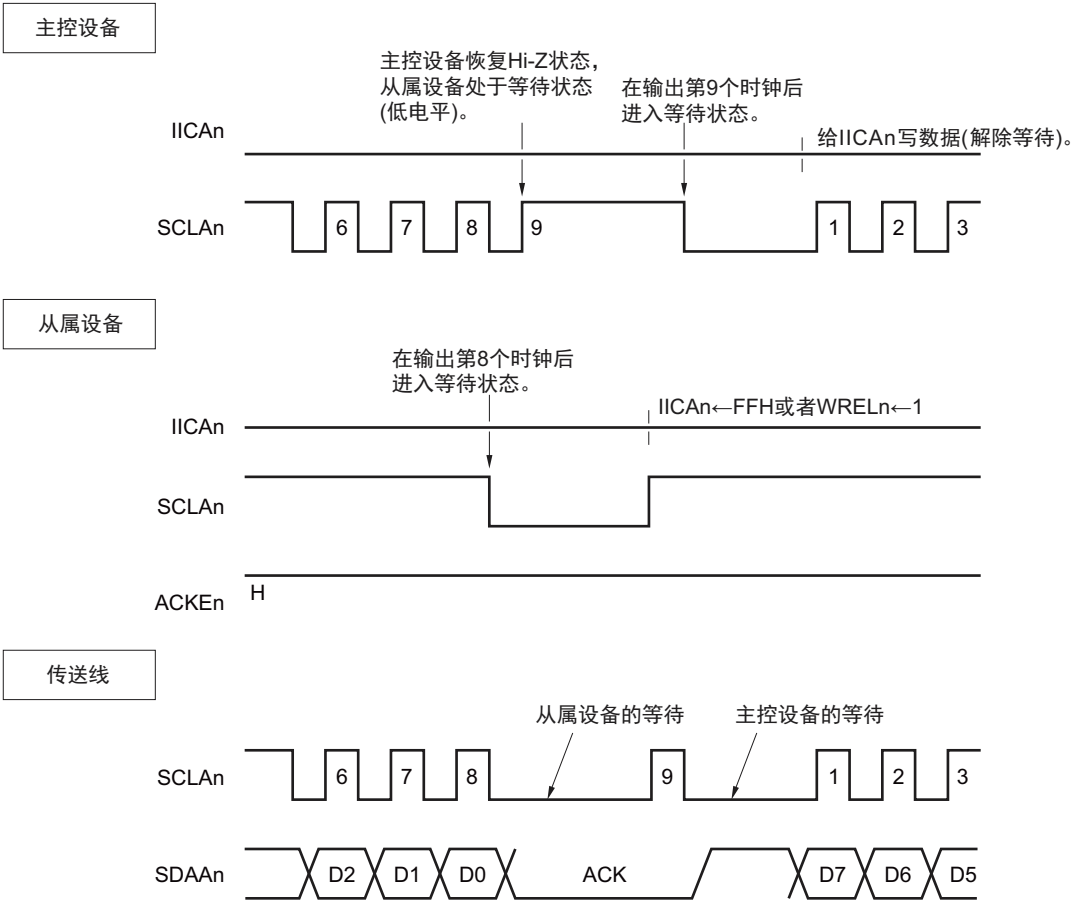
15.5.6 等待

通过等待来通知对方主控设备或者从属设备正在准备数据的接收 / 发送（等待状态）。

通过将 SCLAn 引脚置为低电平，通知对方处于等待状态。如果主控设备和从属设备的等待状态都解除，就能开始下一次传送。

图 15-20 等待（1/2）

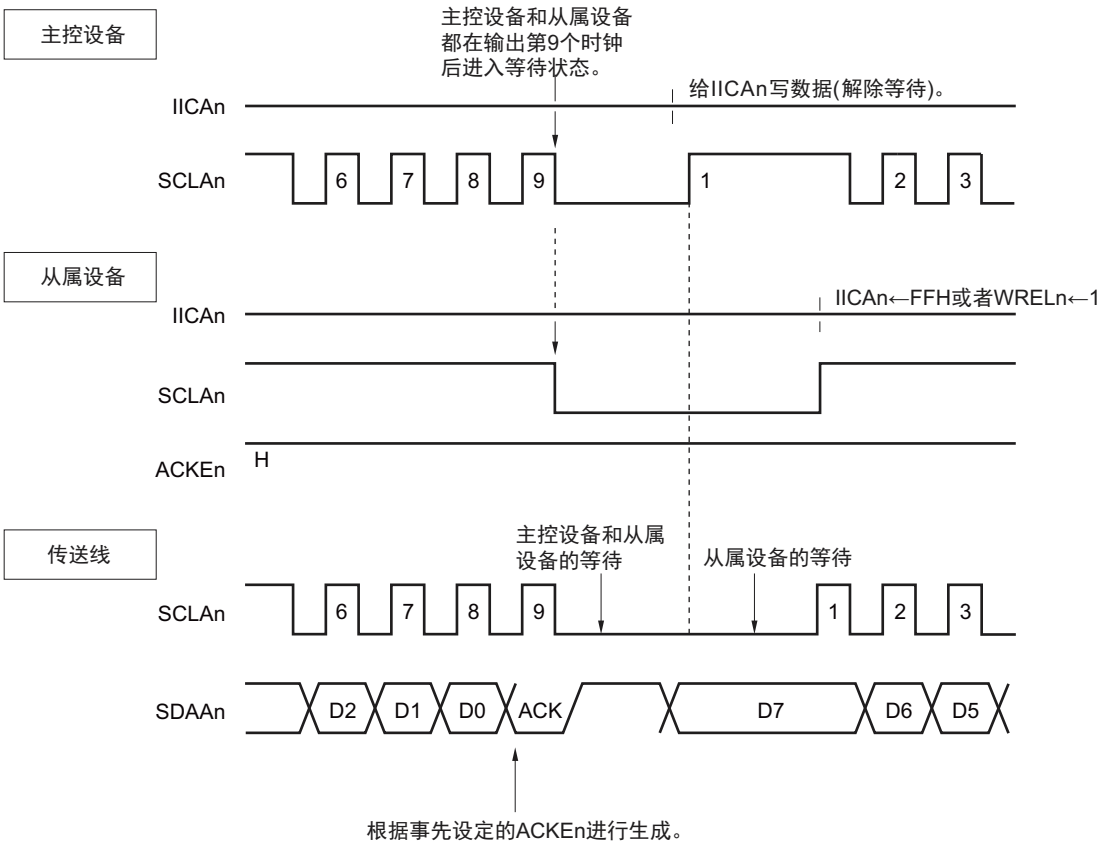
- (1) 主控设备为 9 个时钟等待，从属设备为 8 个时钟等待的情况
（主控设备：发送，从属设备：接收，ACKEn=1）



备注 n=0

图 15-20 等待 (2/2)

(2) 主控设备和从属设备都为 9 个时钟等待的情况
(主控设备：发送，从属设备：接收，ACKEn=1)



备注 ACKEn: IICA 控制寄存器 n0 (IICCTLn0) 的 bit2
WRELn: IICA 控制寄存器 n0 (IICCTLn0) 的 bit5

通过设定 IICA 控制寄存器 n0 (IICCTLn0) 的 bit3 (WTIMn) 自动产生等待。

通常，在接收方，如果 IICCTLn0 寄存器的 bit5 (WRELn) 为“1”或者给 IICA 移位寄存器 n (IICAn) 写“FFH”，就解除等待；在发送方，如果给 IICAn 寄存器写数据，就解除等待。

主控设备还能通过以下方法解除等待：

- 将 IICCTLn0 寄存器的 bit1 (STTn) 置“1”。
- 将 IICCTLn0 寄存器的 bit0 (SPTn) 置“1”。

备注 n=0

15.5.7 等待的解除方法

在一般情况下，I²C 通过以下的处理来解除等待。

- 给 IICA 移位寄存器 n（IICAn）写数据。
- 将 IICA 控制寄存器 n0（IICCTLn0）的 bit5（WRELn）置位（解除等待）。
- 将 IICCTLn0 寄存器的 bit1（STTn）置位（生成开始条件）注。
- 将 IICCTLn0 寄存器的 bit0（SPTn）置位（生成停止条件）注。

注 只限于主控设备。

如果执行了这些等待的解除处理，I²C 就解除等待，重新开始通信。

要在解除等待后发送数据（包括地址）时，必须给 IICAn 寄存器写数据。

要在解除等待后接收数据或者结束发送数据，就必须将 IICCTLn0 寄存器的 bit5（WRELn）置“1”。

要在解除等待后生成重新开始条件，就必须将 IICCTLn0 寄存器的 bit1（STTn）置“1”。

要在解除等待后生成停止条件，就必须将 IICCTLn0 寄存器的 bit0（SPTn）置“1”。

对于一次等待只能执行一次解除处理。

例如，如果在通过将 WRELn 位置“1”来解除等待后给 IICAn 寄存器写数据，SDAAn 线的变化时序与 IICAn 寄存器的写时序就可能发生冲突，导致将错误的值输出到 SDAAn 线。

除了这些处理以外，在中途中止通信的情况下，如果将 IICEn 位清“0”，就停止通信，因此能解除等待。

在 I²C 总线状态因噪声而被死锁的情况下，如果将 IICCTLn0 寄存器的 bit6（LRELn）置“1”，就退出通信，因此能解除等待。

注意 如果在 WUPn 位为“1”时执行等待的解除处理，就不解除等待。

备注 n=0

15.5.8 中断请求（INTIICAn）的产生时序和等待控制

通过设定 IICA 控制寄存器 n0（IICCTLn0）的 bit3（WTIMn），在表 15-2 所示的时序产生 INTIICAn 并且进行等待控制。

表 15-2 INTIICAn 的产生时序和等待控制

WTIMn	从属设备			主控设备		
	地址	数据接收	数据发送	地址	数据接收	数据发送
0	9 注 1、2	8 注 2	8 注 2	9	8	8
1	9 注 1、2	9 注 2	9 注 2	9	9	9

注 1. 只有在接收的地址和从属地址寄存器 n（SVAn）的设定值相同时，从属设备才在第 9 个时钟的下降沿产生 INTIICAn 信号并且进入等待状态。

此时，与 IICCTLn0 寄存器 bit2（ACKEn）的设定无关，生成应答。接收到扩展码的从属设备在第 8 个时钟的下降沿产生 INTIICAn。如果在重新开始后地址不同，就在第 9 个时钟的下降沿产生 INTIICAn，但是不进入等待状态。

2. 如果接收的地址和从属地址寄存器 n（SVAn）的内容不同并且未接收到扩展码，就不产生 INTIICAn 并且也不进入等待状态。

备注 表中的数字表示串行时钟的时钟数。中断请求和等待控制都与串行时钟的下降沿同步。

(1) 地址的发送和接收

- 从属设备：与 WTIMn 位无关，通过上述注 1 和注 2 的条件决定中断和等待的时序。
- 主控设备：与 WTIMn 位无关，在第 9 个时钟的下降沿产生中断和等待的时序。

(2) 数据的接收

- 主控设备/从属设备：通过 WTIMn 位决定中断和等待的时序。

(3) 数据的发送

- 主控设备/从属设备：通过 WTIMn 位决定中断和等待的时序。

备注 n=0

(4) 等待的解除方法

等待的解除方法有以下 4 种：

- 给 IICA 移位寄存器 n (IICAn) 写数据。
- 将 IICA 控制寄存器 n0 (IICCTLn0) 的 bit5 (WRELn) 置位 (解除等待)。
- 将 IICCTLn0 寄存器的 bit1 (STTn) 置位 (生成开始条件) 注。
- 将 IICCTLn0 寄存器的 bit0 (SPTn) 置位 (生成停止条件) 注。

注 只限于主控设备。

当选择 8 个时钟等待 (WTIMn=0) 时，需要在解除等待前决定是否生成应答。

(5) 停止条件的检测

如果检测到停止条件，就产生 INTIICAn (只限于 SPIEn=1)。

15.5.9 地址匹配的检测方法

在 I²C 总线模式中，主控设备能通过发送从属地址来选择特定的从属设备。

通过硬件自动检测地址匹配。当主控设备送来的从属地址和从属地址寄存器 n (SVAn) 的设定地址相同或者接收到扩展码时，产生 INTIICAn 中断请求。

15.5.10 错误的检测

在 I²C 总线模式中，因为发送过程中的串行数据总线 (SDAAn) 的状态被取到发送器件的 IICA 移位寄存器 n (IICAn)，所以能通过将开始发送前和发送结束的 IICA 数据进行比较来检测发送错误。此时，如果 2 个数据不同，就判断为发生了发送错误。

备注 n=0

15.5.11 扩展码

- (1) 当接收地址的高4位为“0000”或者“1111”时，作为接收到扩展码，将扩展码接收标志（EXCn）置“1”，并且在第8个时钟的下降沿产生中断请求（INTIICAn）。
不影响保存在从属地址寄存器n（SVAn）的本地站地址。
- (2) 当SVAn寄存器的设定值为“11110xx0”时，如果通过10位地址传送从主控设备发送“11110xx0”，就发生以下的置位。但是，在第8个时钟的下降沿产生中断请求（INTIICAn）。
 - 高4位数据相同：EXCn=1
 - 7位数据相同：COIn=1

备注 EXCn：IICA 状态寄存器 n（IICSn）的 bit5

COIn：IICA 状态寄存器 n（IICSn）的 bit4

- (3) 中断请求发生后的处理因扩展码的后续数据而不同，通过软件进行处理。
如果在从属设备运行时接收到扩展码，即使地址不同也在参加通信。
例如，在接收到扩展码后不想作为从属设备运行时，必须将 IICA 控制寄存器 n0（IICCTLn0）的 bit6（LRELn）置“1”，进入下次通信的待机状态。

表 15-3 主要扩展码的位定义

从属地址	R/W 位	说明
0 0 0 0 0 0 0	0	全呼地址
1 1 1 1 0 x x	0	10 位从属地址的指定（地址认证时）
1 1 1 1 0 x x	1	10 位从属地址的指定（在地址相同后发行读命令时）

备注 1. 有关上述以外的扩展码，请参照 NXP 公司发行的 I²C 总线规格书。

2. n=0

15.5.12 仲裁

当多个主控设备同时生成开始条件时（在 STDn 位变为“1”前将 STTn 位置“1”的情况），边调整时钟边进行主控设备的通信，直到数据不同为止。此运行称为仲裁。

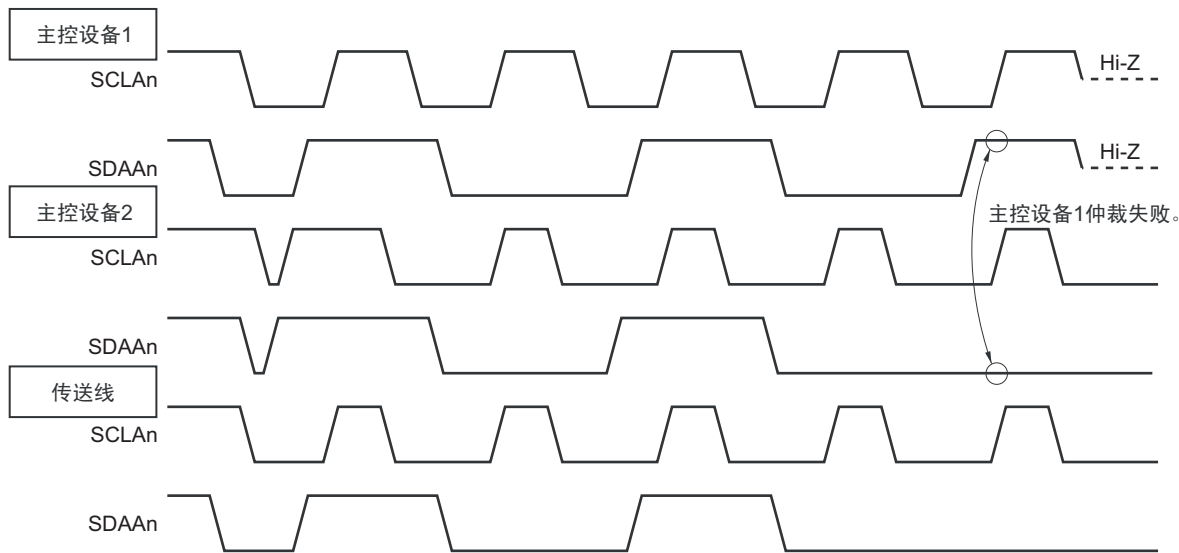
在仲裁失败时，仲裁失败的主控设备将 IICA 状态寄存器 n（IICSn）的仲裁失败标志（ALDn）置“1”，并且将 SCLAn 线和 SDAAn 线都置为高阻抗状态，以释放总线。

在发生下一次中断请求时（例如：在第 8 或者第 9 个时钟检测到停止条件），用软件通过 ALDn 位为“1”来检测仲裁的失败。

有关中断请求的产生时序，请参照“15.5.8 中断请求（INTIICAn）的产生时序和等待控制”。

备注 STDn：IICA 状态寄存器 n（IICSn）的 bit1
STTn：IICA 控制寄存器 n0（IICCTLn0）的 bit1

图 15-21 仲裁时序例子



备注 n=0

表 15-4 发生仲裁时的状态和中断请求的产生时序

发生仲裁时的状态	中断请求的产生时序
地址发送过程中	在字节传送后的第 8 或者第 9 个时钟的下降沿 ^{注 1}
发送地址后的读写信息	
扩展码发送过程中	
发送扩展码后的读写信息	
数据发送过程中	
发送数据后的应答传送过程中	
在数据传送过程中检测到重新开始条件。	
在数据传送过程中检测到停止条件。	在生成停止条件时（SPIEn=1） ^{注 2}
要生成重新开始条件，但是数据为低电平。	在字节传送后的第 8 或者第 9 个时钟的下降沿 ^{注 1}
要生成重新开始条件，但是检测到停止条件。	在生成停止条件时（SPIEn=1） ^{注 2}
要生成停止条件，但是数据为低电平。	在字节传送后的第 8 或者第 9 个时钟的下降沿 ^{注 1}
要生成重新开始条件，但是 SCLAn 为低电平。	

注 1. 当 WTIMn 位（IICA 控制寄存器 n0（IICCTLn0）的 bit3）为“1”时，在第 9 个时钟的下降沿产生中断请求。当 WTIMn 位为“0”并且接收到扩展码的从属地址时，在第 8 个时钟的下降沿产生中断请求。

2. 当有可能发生仲裁时，必须在主控设备时将 SPIEn 位置“1”。

备注 1. SPIEn: IICA 控制寄存器 n0（IICCTLn0）的 bit4

2. n=0

15.5.13 唤醒功能

这是 I²C 的从属功能，是在接收到本地站地址和扩展码时产生中断请求信号（INTIICAn）的功能。

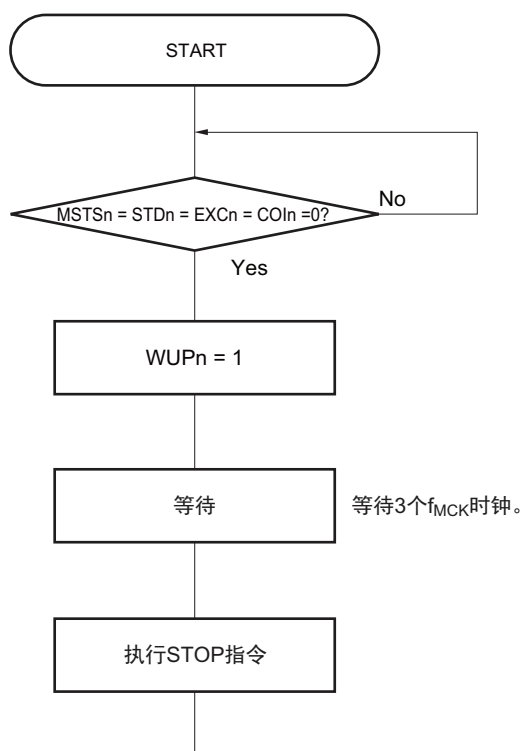
在地址不同的情况下不产生不需要的 INTIICAn 信号，能提高处理效率。

如果检测到开始条件，就进入唤醒待机状态。因为主控设备（已经生成开始条件的情况）也有可能因仲裁失败而变为从属设备，所以在发送地址的同时进入唤醒待机状态。

要在 STOP 模式中使用唤醒功能时，必须将 WUPn 位置“1”。与运行时钟无关而能接收地址。即使在这种情况下，也在接收到本地站地址和扩展码时产生中断请求信号（INTIICAn）。在产生此中断后，通过指令将 WUPn 位清“0”，返回通常运行。

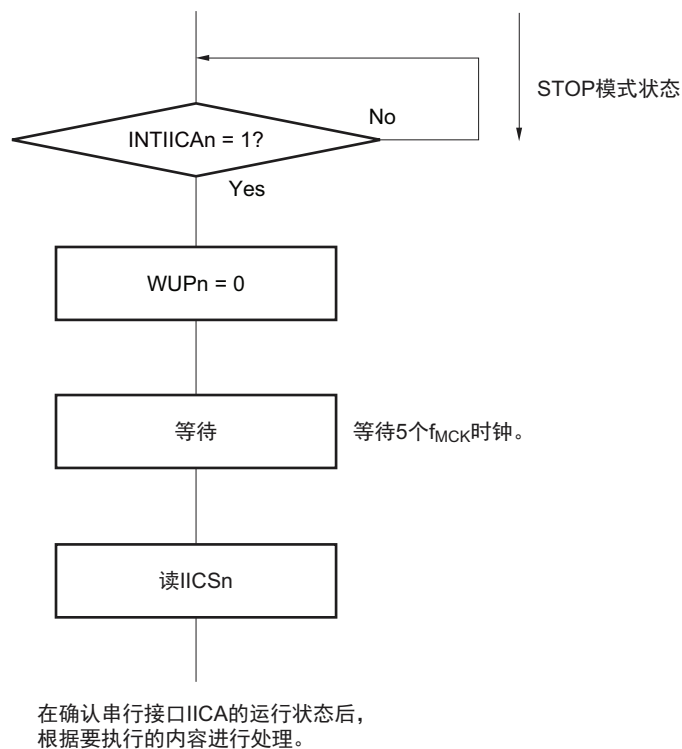
将 WUPn 位置“1”时的流程如图 15-22 所示，通过地址匹配将 WUPn 位置“0”时的流程如图 15-23 所示。

图 15-22 将 WUPn 位置“1”时的流程



备注 n=0

图 15-23 通过地址匹配将 WUPn 位置“0”时的流程（包括接收扩展码）

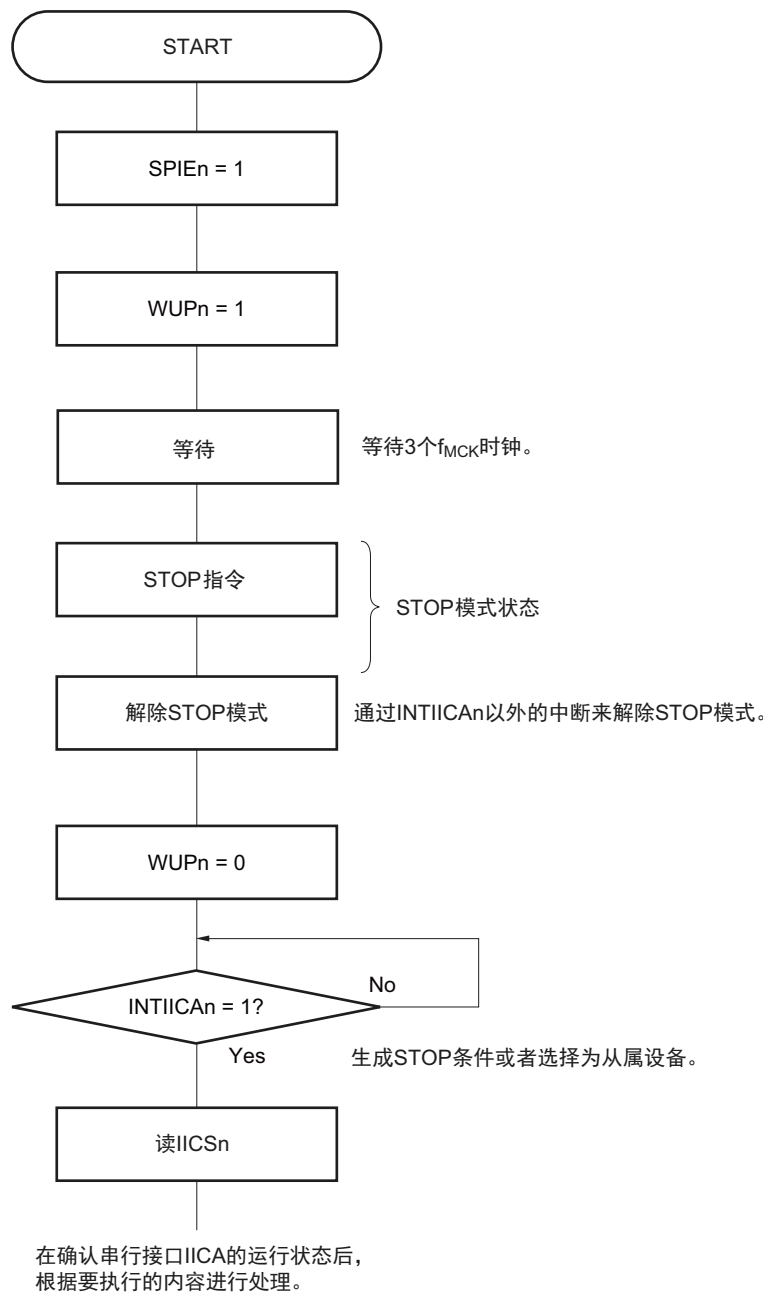


除了串行接口 IICA 产生的中断请求（INTIICAn）以外，必须通过以下的流程解除 STOP 模式。

- 将下一个 IIC 通信作为主控设备运行的情况：图 15-24 的流程
- 将下一个 IIC 通信作为从属设备运行的情况：
 - 通过 INTIICAn 中断返回时：和图 15-23 的流程相同。
 - 通过 INTIICAn 中断以外的中断返回时：保持 WUPn=1 时的运行直到发生 INTIICAn 中断。

备注 n=0

图 15-24 在通过 INTIICAn 以外的方法解除 STOP 模式后作为主控设备运行的情况



备注 n=0

15.5.14 通信预约

(1) 允许通信预约功能的情况 (IICA 标志寄存器 n (IICFn) 的 bit0 (IICRSVn) =0)

要在不加入总线的状态下进行下次的主控通信时，能通过通信预约在释放总线时发送开始条件。此时的不加入总线包括以下 2 种状态：

- 在仲裁结果既不是主控设备也不是从属设备时
- 在接收到扩展码后不作为从属设备运行时（不返回应答而将 IICA 控制寄存器 n0 (IICCTLn0) 的 bit6 (LRELn) 置“1”，退出通信后释放了总线)。

如果在不加入总线的状态下将 IICCTLn0 寄存器的 bit1 (STTn) 置“1”，就在释放总线后（检测到停止条件）自动生成开始条件，进入等待状态。

将 IICCTLn0 寄存器的 bit4 (SPIEn) 置“1”，在通过产生的中断请求信号 (INTIICAn) 检测到总线的释放（检测到停止条件）后，如果给 IICA 移位寄存器 n (IICAn) 写地址，就自动作为主控设备开始通信。在检测到停止条件前，给 IICAn 寄存器写的的数据无效。

当将 STTn 位置“1”时，根据总线状态决定是作为开始条件运行还是作为通信预约运行。

- 总线处于释放状态时 生成开始条件
- 总线未处于释放状态（待机状态）时 通信预约

在将 STTn 位置“1”并且经过等待时间后，通过 MSTSn 位 (IICA 状态寄存器 n (IICSn) 的 bit7) 确认是否作为通信预约运行。

必须通过软件确保以下计算式计算的等待时间：

从将 STTn 位置“1”到确认 MSTSn 标志为止的等待时间 (f_{MCK} 的时钟数)：
 $(IICWLn \text{ 的设定值} + IICWHn \text{ 的设定值} + 4) + t_F \times 2 \times f_{MCK} [\text{时钟}]$

备注 1. IICWLn: IICA 低电平宽度设定寄存器 n

ICWHn: IICA 高电平宽度设定寄存器 n

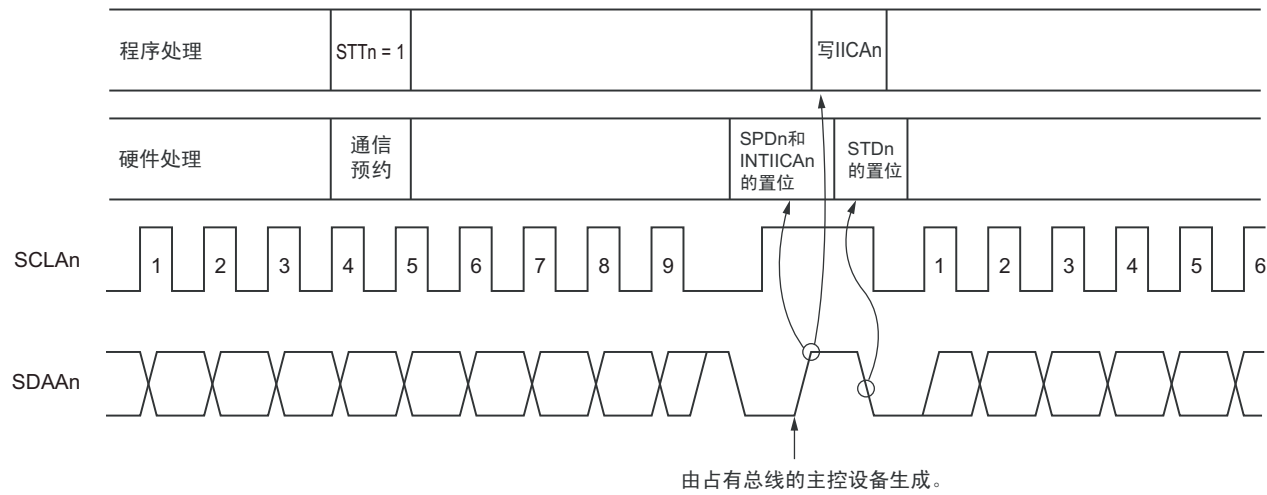
t_F : SDAAn 信号和 SCLAn 信号的下降时间

f_{MCK} : IICA 运行时钟频率

2. n=0

通信预约时序如图 15-25 所示。

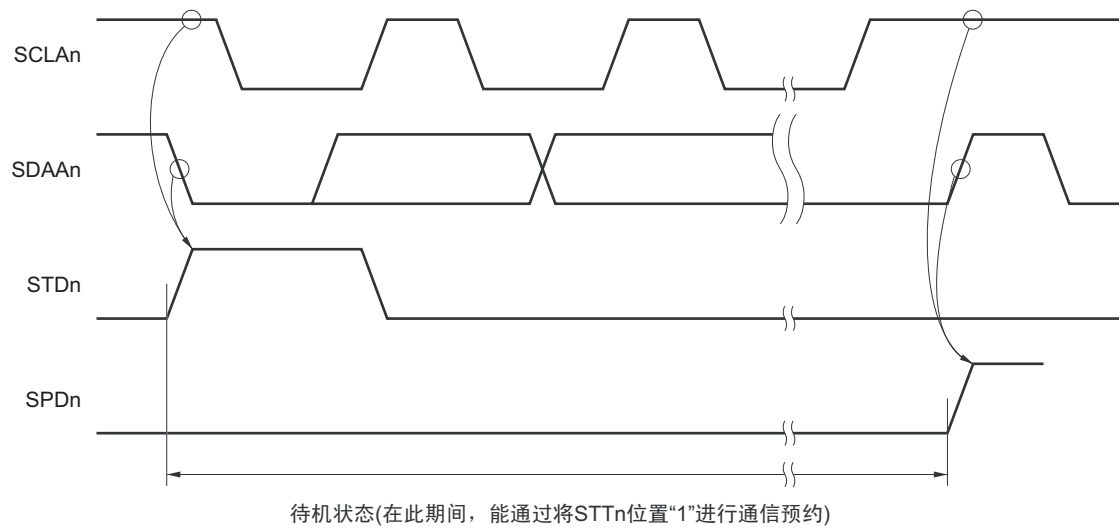
图 15-25 通信预约时序



备注 IICAn: IICA 移位寄存器 n
STTn: IICA 控制寄存器 n0 (IICCTLn0) 的 bit1
STDn: IICA 状态寄存器 n (IICSn) 的 bit1
SPDn: IICA 状态寄存器 n (IICSn) 的 bit0

通过图 15-26 所示的时序接受通信预约。在 IICA 状态寄存器 n (IICSn) 的 bit1 (STDn) 变为“1”后并且在检测到停止条件前，将 IICA 控制寄存器 n0 (IICCTLn0) 的 bit1 (STTn) 置“1”进行通信预约。

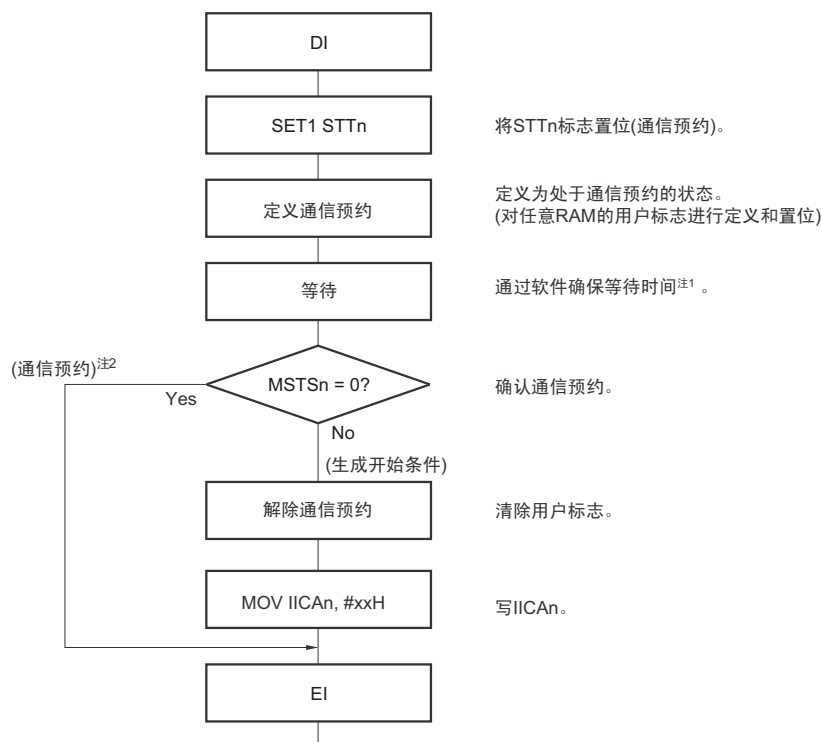
图 15-26 通信预约的接受时序



通信预约步骤如图 15-27 所示。

备注 n=0

图 15-27 通信预约步骤



注 1. 等待时间如下：

(IICWL_n 的设定值 + IICWH_n 的设定值 + 4) + $t_F \times 2 \times f_{CLK}$ [时钟]

2. 在通信预约运行时，通过停止条件中断请求写 IICA 移位寄存器 n (IICAn)。

备注 1. STT_n: IICA 控制寄存器 n0 (IICCTLn0) 的 bit1

MSTS_n: IICA 状态寄存器 n (IICS_n) 的 bit7

IICAn: IICA 移位寄存器 n

IICWL_n: IICA 低电平宽度设定寄存器 n

IICWH_n: IICA 高电平宽度设定寄存器 n

t_F : SDAAn 信号和 SCLAn 信号的下降时间

f_{CLK} : CPU/ 外围硬件的时钟频率

2. n=0

(2) 禁止通信预约功能的情况 (IICA 标志寄存器 n (IICFn) 的 bit0 (IICRSVn) =1)

在总线通信过程中，如果在不参加此通信的状态下将 IICA 控制寄存器 n0 (IICCTLn0) 的 bit1 (STT_n) 置“1”，就拒绝此请求而且不生成开始条件。此时的不加入总线包括以下 2 种状态：

- 在仲裁结果既不是主控设备也不是从属设备时
- 在接收到扩展码后不作为从属设备运行时（不返回应答而将 IICCTLn0 寄存器的 bit6 (LREL_n) 置“1”，退出通信后释放了总线）。

通过 STCF_n (IICFn 寄存器的 bit7) 来确认是生成了开始条件还是拒绝了请求。因为从 STT_n 位为“1”到将 STCF_n 位置“1”需要 5 个 f_{MCK} 时钟的时间，所以必须通过软件确保此时间。

备注 n=0

15.5.15 其他注意事项

(1) STCENn 位为“0”的情况

在刚允许 I²C 运行 (IICEn=1) 后, 与实际的总线状态无关而视为通信状态 (IICBSYn=1)。要在没有检测到停止条件的状态下进行主控通信时, 必须先生成停止条件, 在释放总线后进行主控通信。

对于多主控, 在总线未释放 (未检测到停止条件) 的状态下不能进行主控通信。

按照以下顺序生成停止条件:

- ① 设定 IICA 控制寄存器 n1 (IICCTLn1)。
- ② 将 IICA 控制寄存器 n0 (IICCTLn0) 的 bit7 (IICEn) 置“1”。
- ③ 将 IICCTLn0 寄存器的 bit0 (SPTn) 置“1”。

(2) STCENn 位为“1”的情况

在刚允许 I²C 运行 (IICEn=1) 后, 与实际的总线状态无关而视为释放状态 (IICBSYn=0)。因此在生成第 1 个开始条件 (STTn=1) 时, 为了不破坏其他的通信, 需要确认总线已被释放。

(3) 正在和其他设备进行 I²C 通信的情况

在 SDAAn 引脚为低电平并且 SCLAn 引脚为高电平时, 如果允许 I²C 运行而且中途参加通信, I²C 的宏就视为 SDAAn 引脚从高电平变为低电平 (检测到开始条件)。如果此时总线上的值是能识别为扩展码的值, 就返回应答而妨碍和其他设备的 I²C 通信。为了避免这种情况, 必须按照以下顺序启动 I²C:

- ① 将 IICCTLn0 寄存器的 bit4 (SPIEn) 清“0”, 禁止在检测到停止条件时产生中断请求信号 (INTIICAn)。
- ② 将 IICCTLn0 寄存器的 bit7 (IICEn) 置“1”, 允许 I²C 运行。
- ③ 等待检测到开始条件。
- ④ 在返回应答前 (将 IICEn 位置“1”后的 4 ~ 72 个 f_{MCK} 时钟内) 将 IICCTLn0 寄存器的 bit6 (LRELn) 置“1”, 强制将检测置为无效。

(4) 在将 STTn 位和 SPTn 位 (IICCTLn0 寄存器的 bit1 和 bit0) 置位后, 禁止清“0”前的再置位。

(5) 如果进行了通信预约, 就必须将 SPIEn 位 (IICCTLn0 寄存器的 bit4) 置“1”, 在检测到停止条件时产生中断请求。在产生中断请求后, 通过给 IICA 移位寄存器 n (IICAn) 写通信数据, 开始传送。如果在检测到停止条件时不发生中断, 就因为开始通信时不产生中断请求而停止在等待状态。但是, 当通过软件检测 MSTSn 位 (IICA 状态寄存器 n (IICSn) 的 bit7) 时, 不需要将 SPIEn 位置“1”。

备注 n=0

15.5.16 通信运行

在此通过流程图表示以下 3 个运行步骤。

(1) 单主控系统的主控运行

在单主控系统中用作主控设备的流程图如下所示。

此流程大体分为“初始设定”和“通信处理”。在启动时执行“初始设定”部分，如果需要和从属设备进行通信，就在进行通信时所需的准备后执行“通信处理”部分。

(2) 多主控系统的主控运行

在 I²C 总线的多主控系统中，只根据 I²C 总线的规格无法判断在参加通信的阶段总线是处于释放状态还是处于使用状态。在此，如果数据和时钟在一定时间内（1 帧）为高电平，就将总线作为释放状态而参加通信。

此流程大体分为“初始设定”、“通信等待”和“通信处理”。在此省略因仲裁失败而被指定为从属设备的处理，只表示用作主控设备的处理。在启动时执行“初始设定”部分后加入总线，然后通过“通信等待”，等待主控设备的通信请求或者从属设备的指定。实际进行通信的是“通信处理”部分，除了支持与从属设备进行数据发送和接收以外，还支持与其他主控设备的仲裁。

(3) 从属运行

用作 I²C 总线从属设备的例子如下所示。

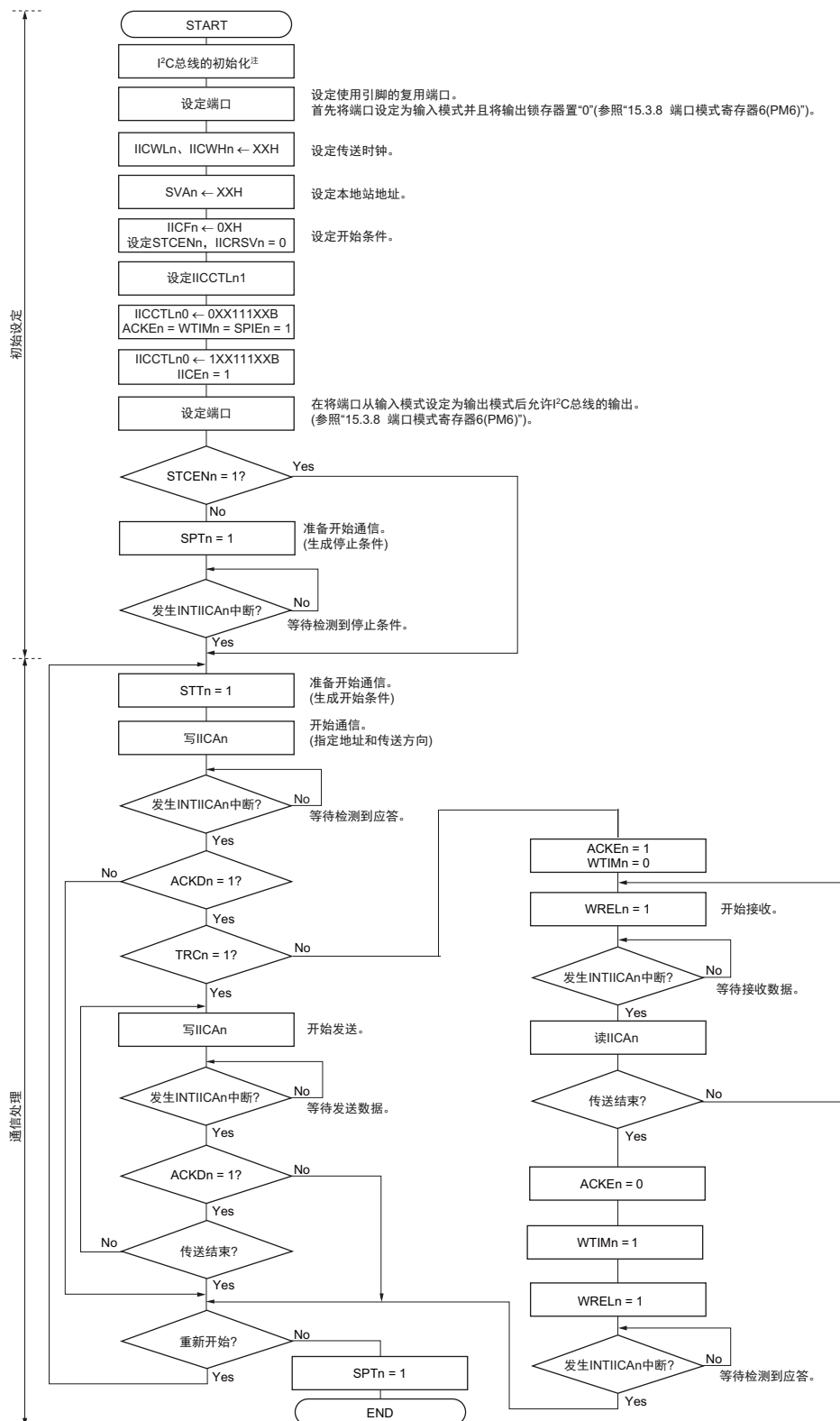
当用作从属设备时，通过中断开始运行。在启动时执行“初始设定”部分，然后通过“通信等待”，等待 INTIICAn 中断的发生。如果发生 INTIICAn 中断，就判断通信状态并且将标志传递给主处理部。

通过检查各标志，进行所要的“通信处理”。

备注 n=0

(1) 单主控系统的主控运行

图 15-28 单主控系统的主控运行



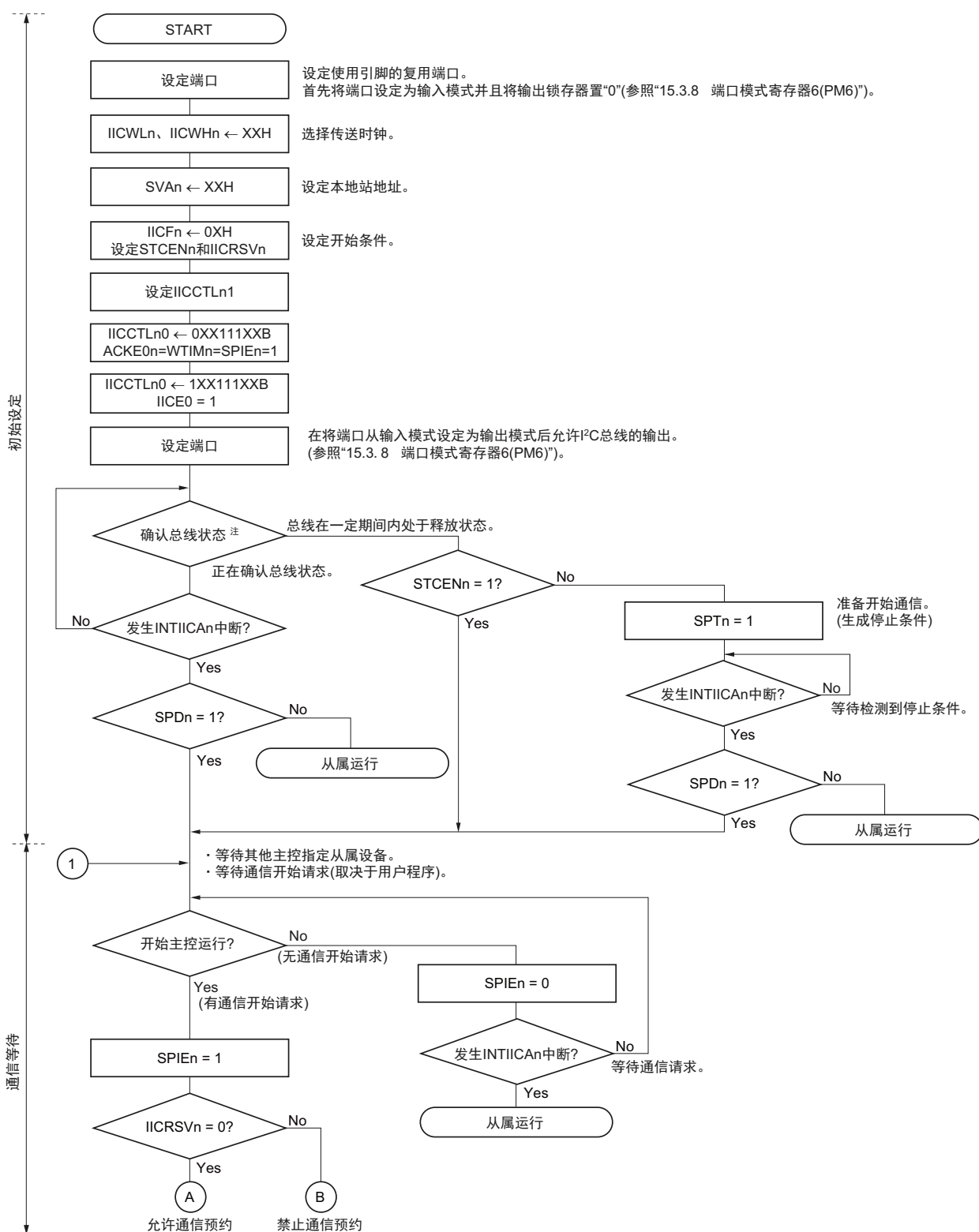
注 必须根据通信中的产品的规格，释放 I²C 总线（SCLAn 引脚和 SDAAn 引脚为高电平）。例如，如果 EEPROM 处于给 SDAAn 引脚输出低电平的状态，就必须将 SCLAn 引脚设定为输出端口，并且在 SDAAn 引脚固定为高电平前从输出端口输出时钟脉冲。

备注 1. 发送和接收格式必须符合通信中的产品的规格。

2. $n=0$

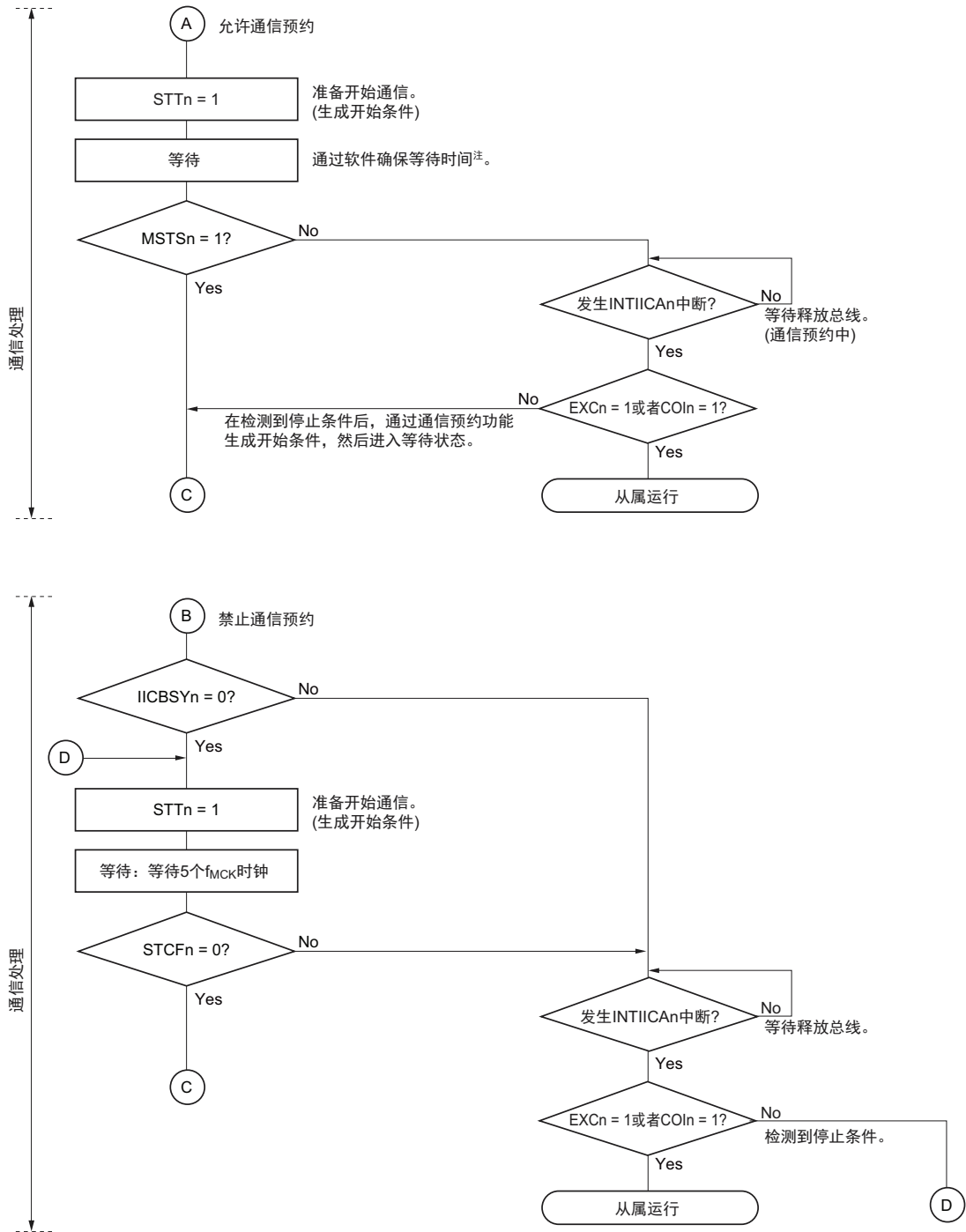
(2) 多主控系统的主控运行

图 15-29 多主控系统的主控运行 (1/3)



备注 n=0

图 15-29 多主控系统的主控运行（2/3）



注 等待时间（ f_{MCK} 的时钟数）如下：
(IICWLn 的设定值 + IICWHn 的设定值 + 4) / $t_F \times 2 \times f_{MCK}$ [时钟]

备注 1. IICWLn: IICA 低电平宽度设定寄存器 n

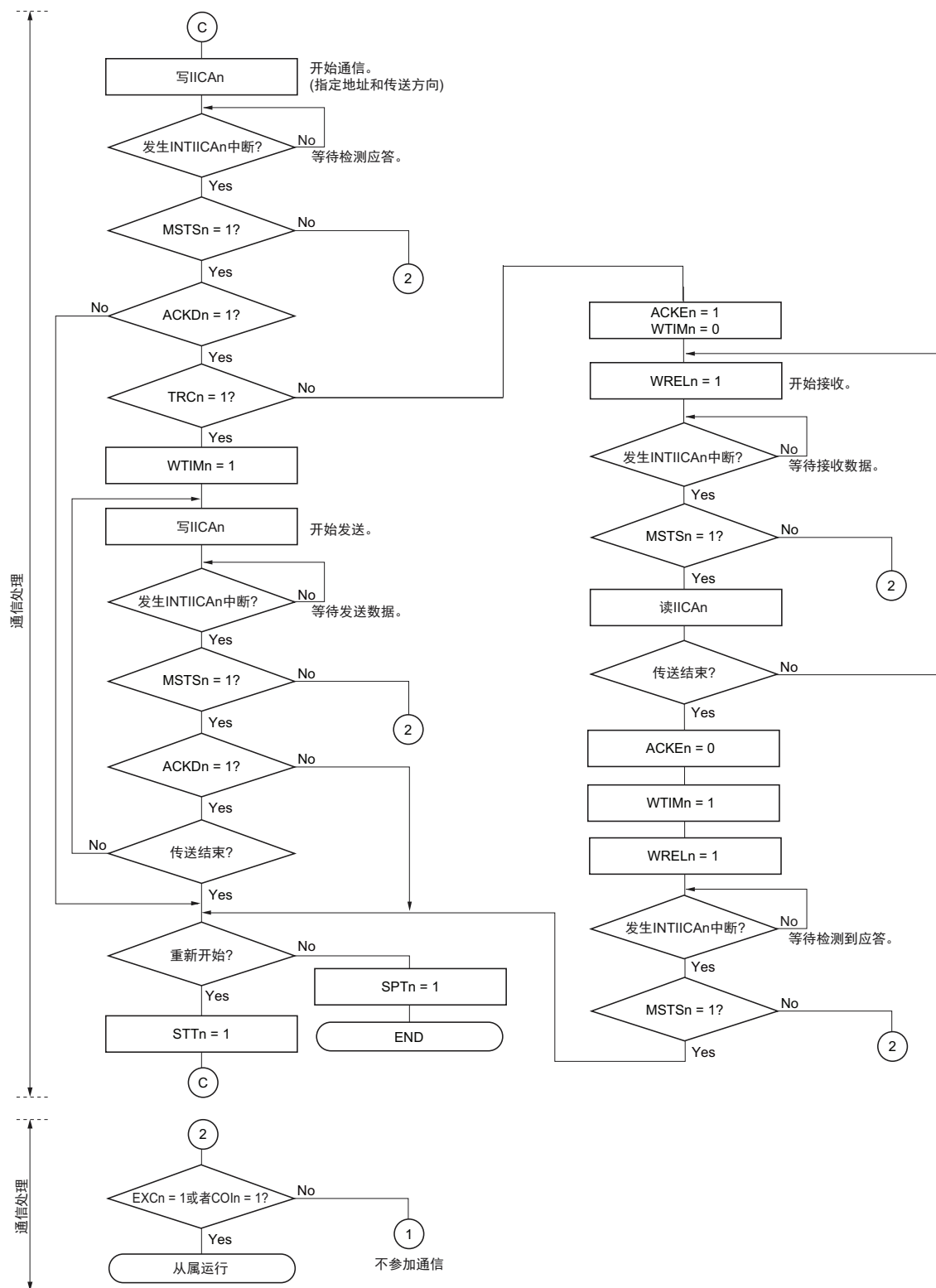
IICWHn: IICA 高电平宽度设定寄存器 n

t_F : SDAAn 信号和 SCLAn 信号的下降时间

f_{MCK} : IICA 运行时钟频率

2. n=0

图 15-29 多主控系统的主控运行 (3/3)



备注 1. 传送和接收格式必须符合通信中的产品的规格。

2. 在多主控系统中用作主控设备的情况下，必须在每次发生 INTIICAn 中断时读 MSTSn 位，确认仲裁结果。

3. 在多主控系统中用作从属设备的情况下，必须在每次发生 INTIICAn 中断时通过 IICA 状态寄存器 n (IICSn) 和 IICA 标志寄存器 n (IICFn) 确认状态，决定以后的处理。

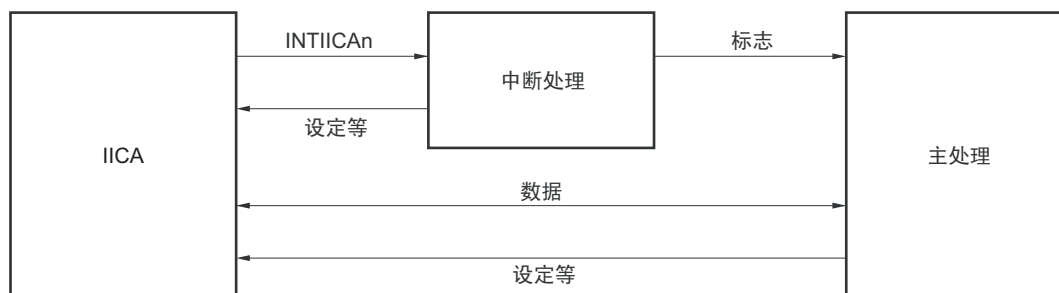
4. n=0

(3) 从属运行

从属运行的处理步骤如下所示。

从属运行基本上由事件驱动，因此需要通过 INTIICAn 中断进行处理（需要对通信中的停止条件检测等的运行状态进行很大的变更处理）。

在此说明中，假设数据通信不支持扩展码，INTIICAn 中断处理只进行状态转移处理并且实际的数据通信由主处理部进行。



因此，准备以下 3 个标志并且代替 INTIICAn 将标志传递给主处理部，进行数据通信处理。

① 通信模式标志

此标志表示以下 2 种通信状态：

- 清除模式：不在进行数据通信的状态
- 通信模式：正在进行数据通信的状态（有效地址的检测～停止条件的检测，未检测到主控设备的应答，地址不同）

② 就绪标志

此标志表示可进行数据通信。在通常的数据通信中，和 INTIICAn 中断相同，由中断处理部置位而由主处理部清除。在开始通信时，由中断处理部清除标志。但是，在发送第 1 个数据时，中断处理部不将就绪标志置位，因此在不清除标志的状态下发送第 1 个数据（地址匹配被解释为下次的请求）。

③ 通信方向标志

此标志表示通信方向，和 TRCn 位的值相同。

备注 n=0

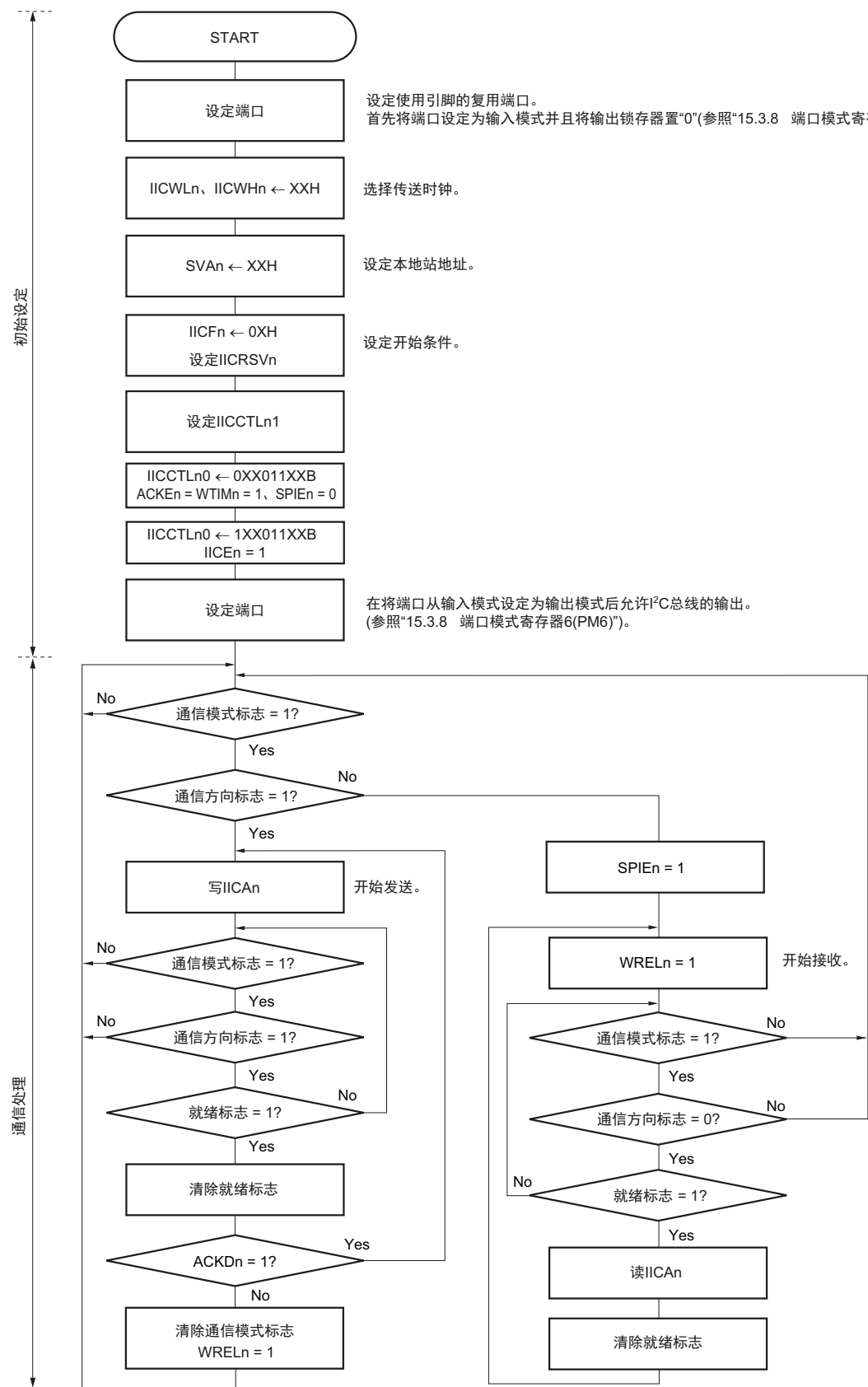
从属运行的主处理部的运行如下所示。

启动串行接口 IICA，等待变为可通信的状态。如果变为可通信的状态，就使用通信模式标志和就绪标志进行通信（通过中断进行停止条件和开始条件的处理，在此通过标志确认状态）。

在发送时，重复发送，直到主控设备不返回应答为止。如果主控设备不返回应答，就结束通信。

在接收时，接收所需数量的数据。如果通信结束，就在下一个数据时不返回应答。此后，主控设备生成停止条件或者重新开始条件，从而退出通信状态。

图 15-30 从属运行步骤 (1)



备注 1. 传送和接收格式必须符合通信中的产品的规格。

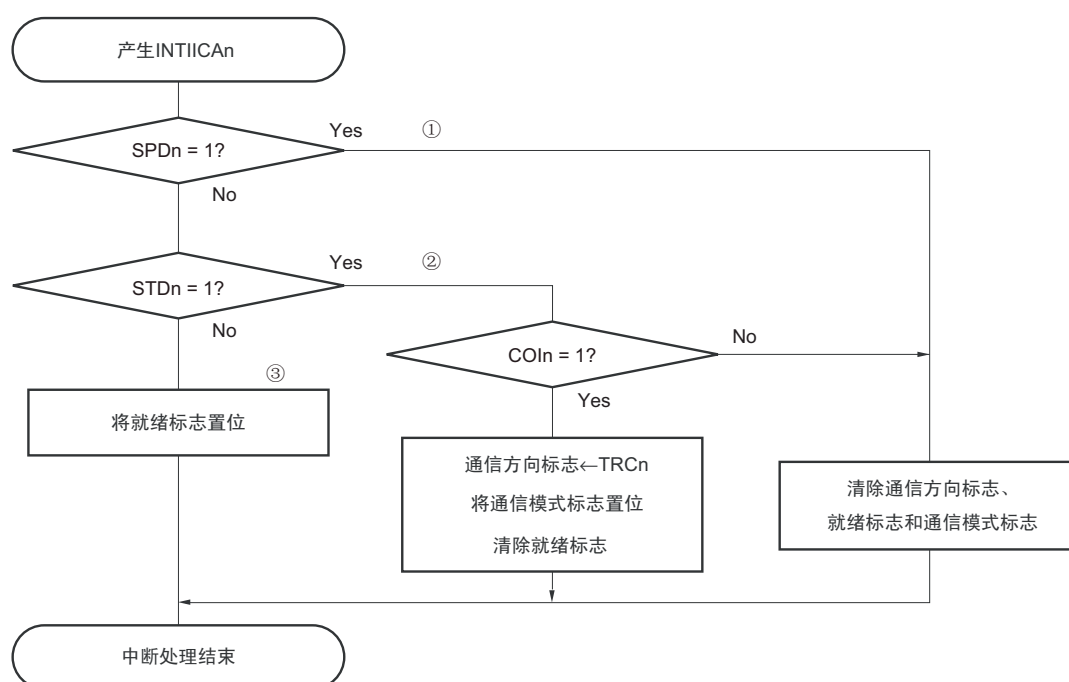
2. n=0

从属设备通过 INTIICAn 中断进行处理的步骤例子如下所示（在此假设没有用扩展码进行处理）。通过 INTIICAn 中断确认状态并且进行以下处理。

- ① 如果生成停止条件，就结束通信。
- ② 如果生成开始条件，就确认地址。如果地址不同，就结束通信。如果地址相同，就设定为通信模式并且解除等待，然后从中断返回（清除就绪标志）。
- ③ 当发送和接收数据时，只要将就绪标志置位，I²C 总线就保持等待状态并且从中断返回。

备注 上述的①～③对应“图 15-31 从属运行步骤 (2)”的①～③。

图 15-31 从属运行步骤 (2)



备注 n=0

15.5.17 I²C 中断请求 (INTIICAn) 的产生时序

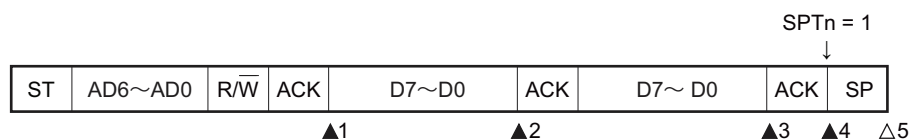
数据的发送和接收时序、INTIICAn 中断请求信号的产生时序以及产生 INTIICAn 信号时的 IICA 状态寄存器 n (IICSn) 的值如下所示。

- 备注 1. ST: 开始条件
 AD6 ~ AD0: 地址
 R/W: 传送方向的指定
 ACK: 应答
 D7 ~ D0: 数据
 SP: 停止条件
2. n=0

(1) 主控运行

(a) Start ~ Address ~ Data ~ Data ~ Stop (发送和接收)

(i) WTIMn=0 的情况



▲ 1: IICSn=1000×110B

▲ 2: IICSn=1000×000B

▲ 3: IICSn=1000×000B (将 WTIMn 位置“1”) 注

▲ 4: IICSn=1000××00B (将 SPTn 位置“1”)

△ 5: IICSn=00000001B

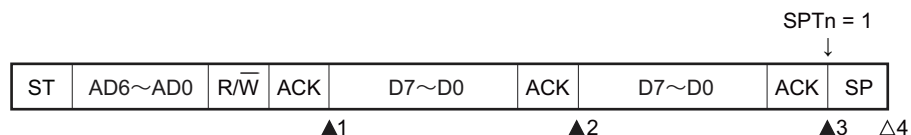
注 为了生成停止条件，必须将 WTIMn 位置“1”并且更改 INTIICAn 中断请求信号的产生时序。

备注 ▲ 一定产生。

 △ 只在 SPIEn 位为“1”时产生。

 × 任意

(ii) WTIMn=1 的情况



▲ 1: IICSn=1000×110B

▲ 2: IICSn=1000×100B

▲ 3: IICSn=1000××00B (将 SPTn 位置“1”)

△ 4: IICSn=00000001B

备注 ▲ 一定产生。

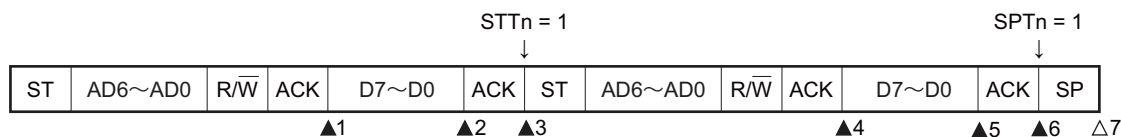
 △ 只在 SPIEn 位为“1”时产生。

 × 任意

备注 n=0

(b) Start ~ Address ~ Data ~ Start ~ Address ~ Data ~ Stop (重新开始)

(i) WTIMn=0 的情况

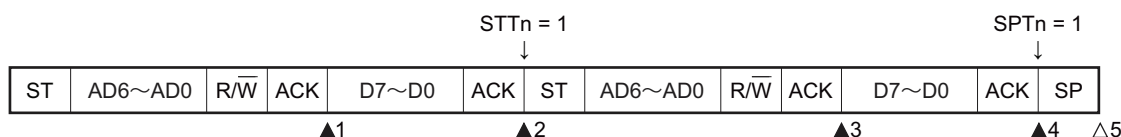


- ▲ 1: IICSn=1000×110B
 ▲ 2: IICSn=1000×000B (将 WTIMn 位置“1”) 注 1
 ▲ 3: IICSn=1000××00B (将 WTIMn 位清“0”注 2 并且将 STTn 位置“1”)
 ▲ 4: IICSn=1000×110B
 ▲ 5: IICSn=1000×000B (将 WTIMn 位置“1”) 注 3
 ▲ 6: IICSn=1000××00B (将 SPTn 位置“1”)
 △ 7: IICSn=00000001B

注 1. 为了生成开始条件, 必须将 WTIMn 位置“1”并且更改 INTIICAn 中断请求信号的产生时序。
 2. 为了恢复原来的设定, 必须将 WTIMn 位清“0”。
 3. 为了生成停止条件, 必须将 WTIMn 位置“1”并且更改 INTIICAn 中断请求信号的产生时序。

备注 ▲ 一定产生
 △ 只在 SPIEn 位为“1”时产生。
 × 任意

(ii) WTIMn=1 的情况



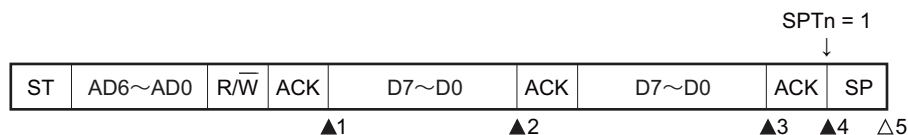
- ▲ 1: IICSn=1000×110B
 ▲ 2: IICSn=1000××00B (将 STTn 位置“1”)
 ▲ 3: IICSn=1000×110B
 ▲ 4: IICSn=1000××00B (将 SPTn 位置“1”)
 △ 5: IICSn=00000001B

备注 ▲ 一定产生。
 △ 只在 SPIEn 位为“1”时产生。
 × 任意

备注 n=0

(c) Start ~ Code ~ Data ~ Data ~ Stop (发送扩展码)

(i) WTIMn=0 的情况



▲ 1: IICSn=1010×110B

▲ 2: IICSn=1010×000B

▲ 3: IICSn=1010×000B (将 WTIMn 位置“1”) 注

▲ 4: IICSn=1010××00B (将 SPTn 位置“1”)

△ 5: IICSn=00000001B

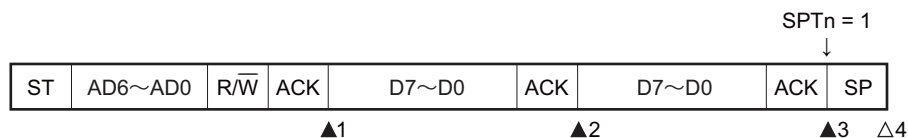
注 为了生成停止条件，必须将 WTIMn 位置“1”并且更改 INTIICAn 中断请求信号的产生时序。

备注 ▲ 一定产生。

△ 只在 SPIEn 位为“1”时产生。

× 任意

(ii) WTIMn=1 的情况



▲ 1: IICSn=1010×110B

▲ 2: IICSn=1010×100B

▲ 3: IICSn=1010××00B (将 SPTn 位置“1”)

△ 4: IICSn=00001001B

备注 ▲ 一定产生。

△ 只在 SPIEn 位为“1”时产生。

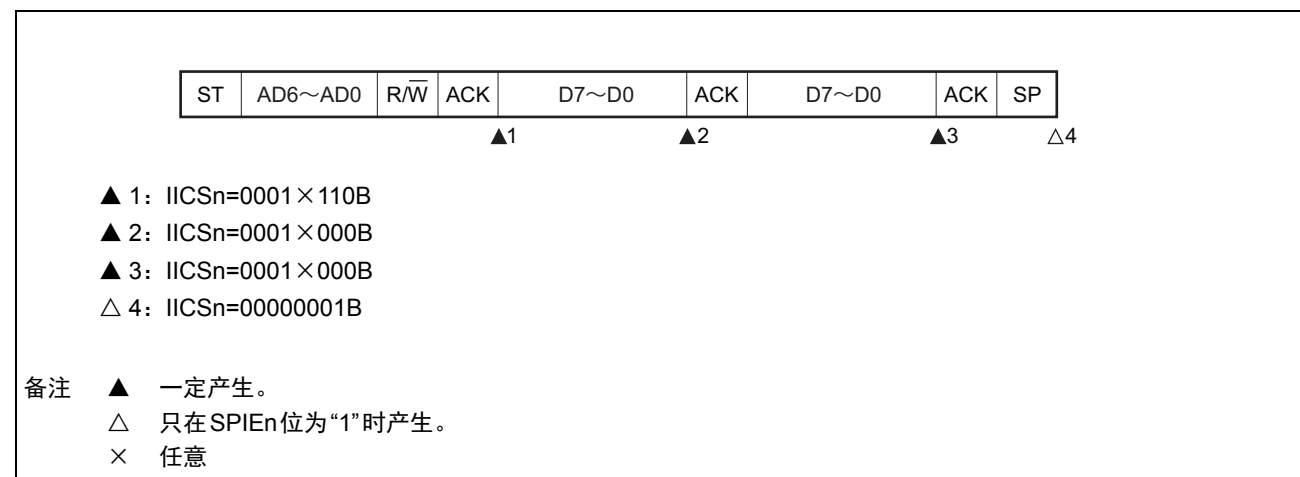
× 任意

备注 n=0

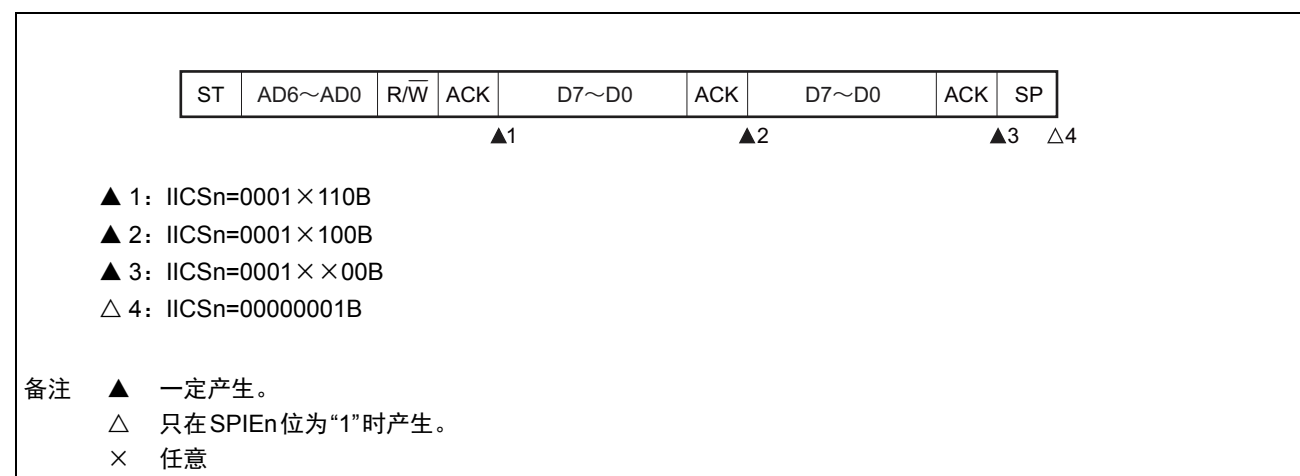
(2) 从属运行（接收从属地址的情况）

(a) Start ~ Address ~ Data ~ Data ~ Stop

(i) WTIMn=0 的情况



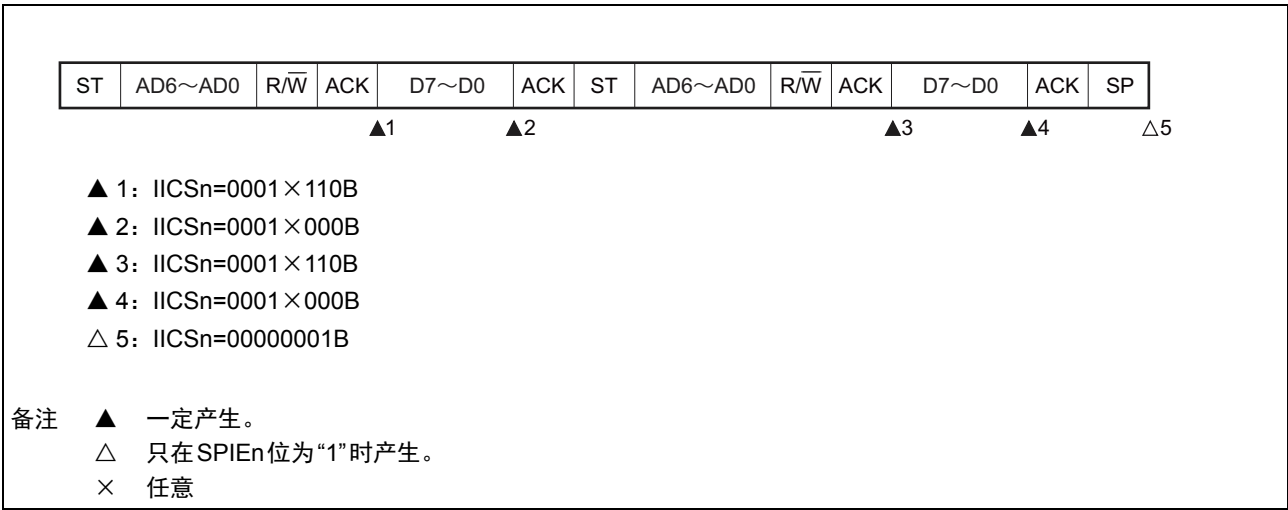
(ii) WTIMn=1 的情况



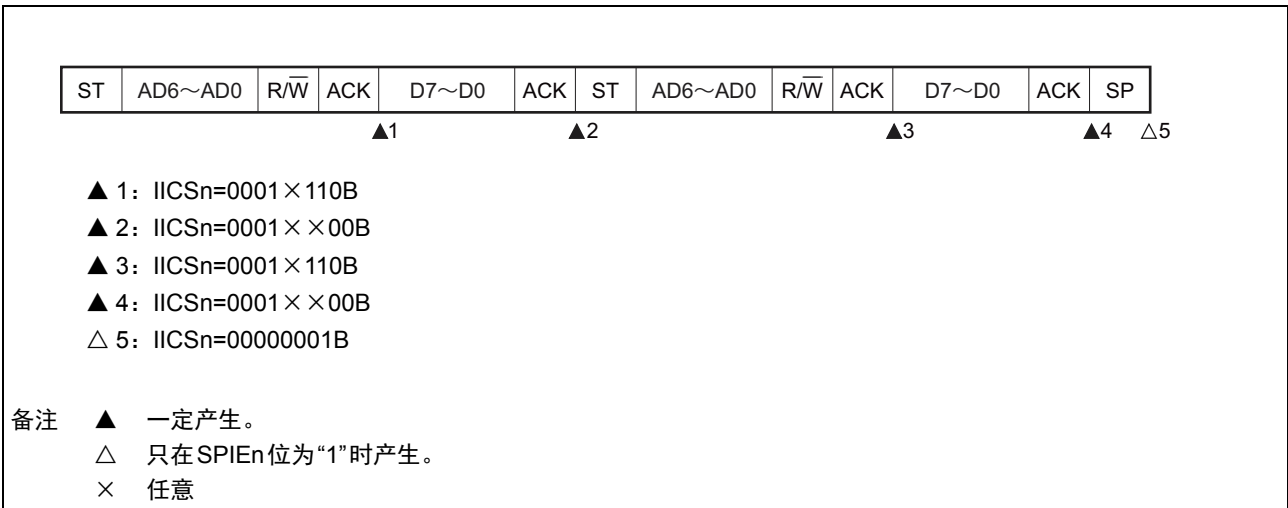
备注 n=0

(b) Start ~ Address ~ Data ~ Start ~ Address ~ Data ~ Stop

(i) WTIMn=0的情况（在重新开始后SVAn相同）



(ii) WTIMn=1的情况（在重新开始后SVAn相同）



备注 n=0

(c) Start ~ Address ~ Data ~ Start ~ Code ~ Data ~ Stop

(i) WTIMn=0的情况（在重新开始后地址不同（扩展码））



▲ 1: IICSn=0001×110B

▲ 2: IICSn=0001×000B

▲ 3: IICSn=0010×010B

▲ 4: IICSn=0010×000B

△ 5: IICSn=00000001B

备注 ▲ 一定产生。

△ 只在SPIEn位为“1”时产生。

× 任意

(ii) WTIMn=1的情况（在重新开始后地址不同（扩展码））



▲ 1: IICSn=0001×110B

▲ 2: IICSn=0001××00B

▲ 3: IICSn=0010×010B

▲ 4: IICSn=0010×110B

▲ 5: IICSn=0010××00B

△ 6: IICSn=00000001B

备注 ▲ 一定产生。

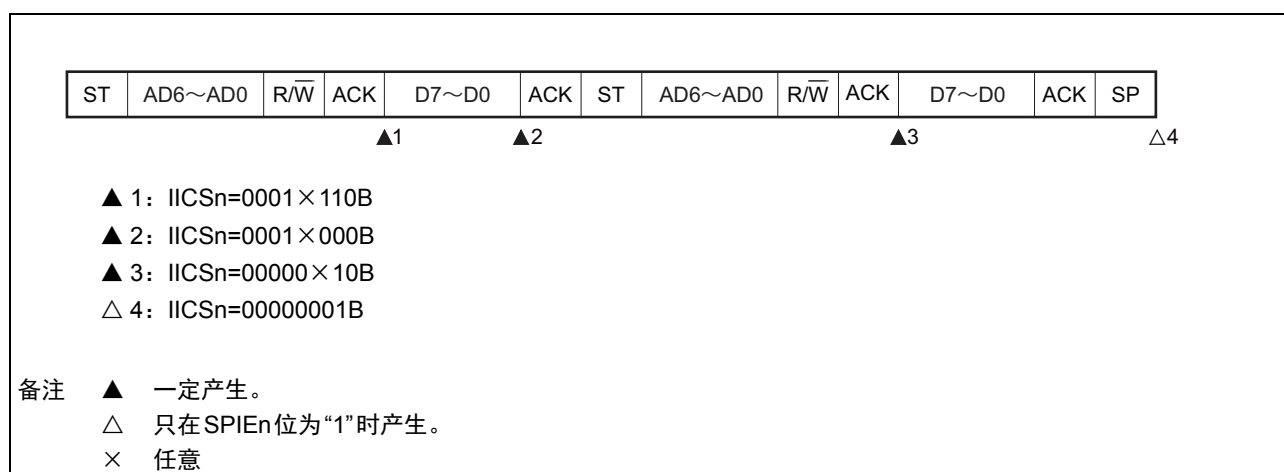
△ 只在SPIEn位为“1”时产生。

× 任意

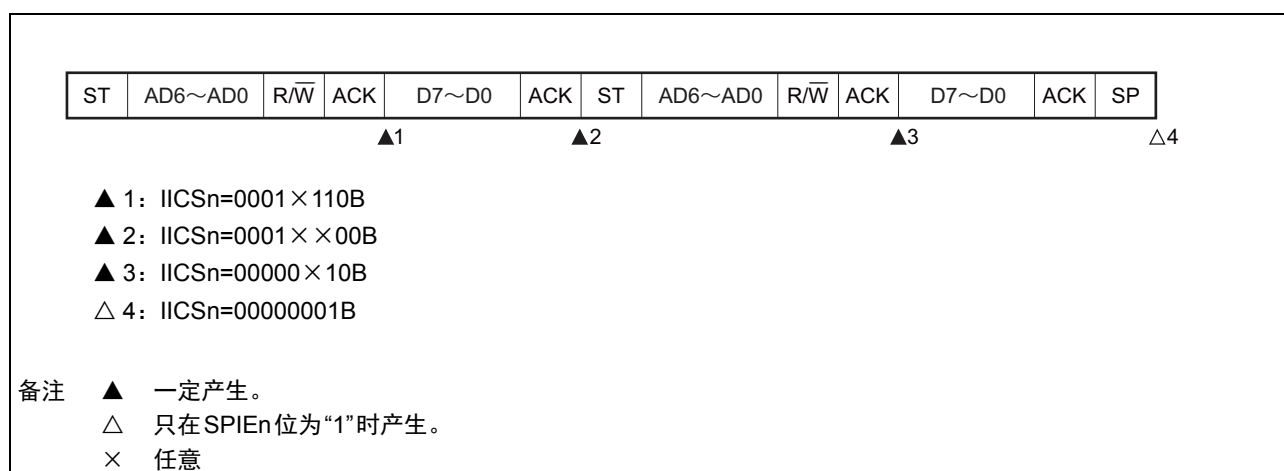
备注 n=0

(d) Start ~ Address ~ Data ~ Start ~ Address ~ Data ~ Stop

(i) WTIMn=0 的情况（在重新开始后地址不同（非扩展码））



(ii) WTIMn=1 的情况（在重新开始后地址不同（非扩展码））



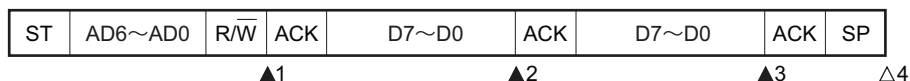
备注 n=0

(3) 从属运行（接收扩展码的情况）

在接收扩展码时，始终参加通信。

(a) Start ~ Code ~ Data ~ Data ~ Stop

(i) WTIMn=0 的情况



▲ 1: IICSn=0010×010B

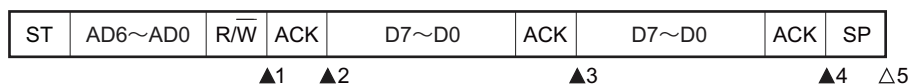
▲ 2: IICSn=0010×000B

▲ 3: IICSn=0010×000B

△ 4: IICSn=00000001B

备注 ▲ 一定产生。
 △ 只在 SPIEn 位为“1”时产生。
 × 任意

(ii) WTIMn=1 的情况



▲ 1: IICSn=0010×010B

▲ 2: IICSn=0010×110B

▲ 3: IICSn=0010×100B

▲ 4: IICSn=0010××00B

△ 5: IICSn=00000001B

备注 ▲ 一定产生。
 △ 只在 SPIEn 位为“1”时产生。
 × 任意

备注 n=0

(b) Start ~ Code ~ Data ~ Start ~ Address ~ Data ~ Stop

(i) WTIMn=0 的情况 (在重新开始后 SVAn 相同)



▲ 1: IICSn=0010×010B

▲ 2: IICSn=0010×000B

▲ 3: IICSn=0001×110B

▲ 4: IICSn=0001×000B

△ 5: IICSn=00000001B

备注 ▲ 一定产生。
 △ 只在 SPIEn 位为“1”时产生。
 × 任意

(ii) WTIMn=1 的情况 (在重新开始后 SVAn 相同)



▲ 1: IICSn=0010×010B

▲ 2: IICSn=0010×110B

▲ 3: IICSn=0010××00B

▲ 4: IICSn=0001×110B

▲ 5: IICSn=0001××00B

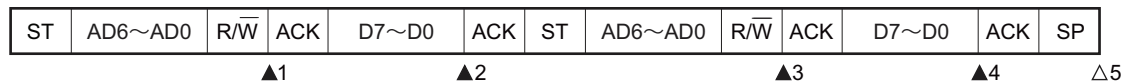
△ 6: IICSn=00000001B

备注 ▲ 一定产生。
 △ 只在 SPIEn 位为“1”时产生。
 × 任意

备注 n=0

(c) Start ~ Code ~ Data ~ Start ~ Code ~ Data ~ Stop

(i) WTIMn=0 的情况（在重新开始后接收扩展码）



▲ 1: IICSn=0010×010B

▲ 2: IICSn=0010×000B

▲ 3: IICSn=0010×010B

▲ 4: IICSn=0010×000B

△ 5: IICSn=00000001B

备注 ▲ 一定产生。

△ 只在SPIEn位为“1”时产生。

× 任意

(ii) WTIMn=1 的情况（在重新开始后接收扩展码）



▲ 1: IICSn=0010×010B

▲ 2: IICSn=0010×110B

▲ 3: IICSn=0010××00B

▲ 4: IICSn=0010×010B

▲ 5: IICSn=0010×110B

▲ 6: IICSn=0010××00B

△ 7: IICSn=00000001B

备注 ▲ 一定产生。

△ 只在SPIEn位为“1”时产生。

× 任意

备注 n=0

(d) Start ~ Code ~ Data ~ Start ~ Address ~ Data ~ Stop

(i) WTIMn=0的情况（在重新开始后地址不同（非扩展码））



▲ 1: IICSn=0010×010B

▲ 2: IICSn=0010×000B

▲ 3: IICSn=00000×10B

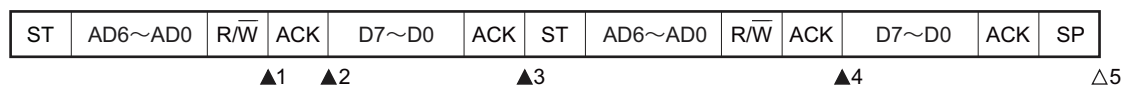
△ 4: IICSn=00000001B

备注 ▲ 一定产生。

△ 只在SPIEn位为“1”时产生。

× 任意

(ii) WTIMn=1的情况（在重新开始后地址不同（非扩展码））



▲ 1: IICSn=0010×010B

▲ 2: IICSn=0010×110B

▲ 3: IICSn=0010××00B

▲ 4: IICSn=00000×10B

△ 5: IICSn=00000001B

备注 ▲ 一定产生。

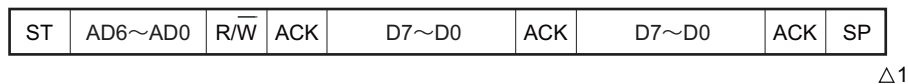
△ 只在SPIEn位为“1”时产生。

× 任意

备注 n=0

(4) 不参加通信的运行

(a) Start ~ Code ~ Data ~ Data ~ Stop



△ 1: IICSn=00000001B

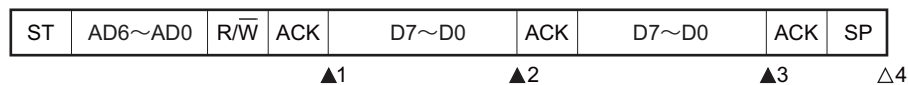
备注 △: 只在 SPIEn 位为“1”时产生。

(5) 仲裁失败的运行（在仲裁失败后作为从属设备运行）

在多主控系统中用作主控设备时，必须在每次产生 INTIICAn 中断请求信号时读 MSTSn 位，确认仲裁结果。

(a) 在发送从属地址数据的过程中仲裁失败的情况

(i) WTIMn=0 的情况



▲ 1: IICSn=0101×110B

▲ 2: IICSn=0001×000B

▲ 3: IICSn=0001×000B

△ 4: IICSn=00000001B

备注 ▲ 一定产生。

△ 只在 SPIEn 位为“1”时产生。

× 任意

备注 n=0

(ii) WTIMn=1 的情况

ST	AD6~AD0	R/W	ACK	D7~D0	ACK	D7~D0	ACK	SP
			▲1		▲2		▲3	△4

▲ 1: IICSn=0101×110B
▲ 2: IICSn=0001×100B
▲ 3: IICSn=0001××00B
△ 4: IICSn=00000001B

备注 ▲ 一定产生。
 △ 只在SPIEn位为“1”时产生。
 × 任意

(b) 在发送扩展码的过程中仲裁失败的情况

(i) WTIMn=0 的情况

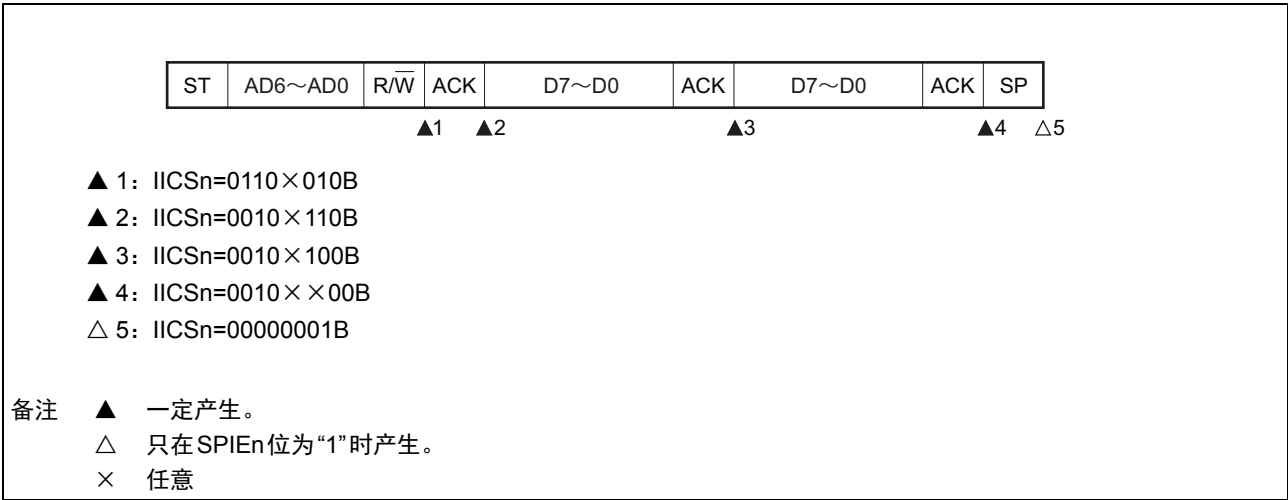
ST	AD6~AD0	R/W	ACK	D7~D0	ACK	D7~D0	ACK	SP
			▲1		▲2		▲3	△4

▲ 1: IICSn=0110×010B
▲ 2: IICSn=0010×000B
▲ 3: IICSn=0010×000B
△ 4: IICSn=00000001B

备注 ▲ 一定产生。
 △ 只在SPIEn位为“1”时产生。
 × 任意

备注 n=0

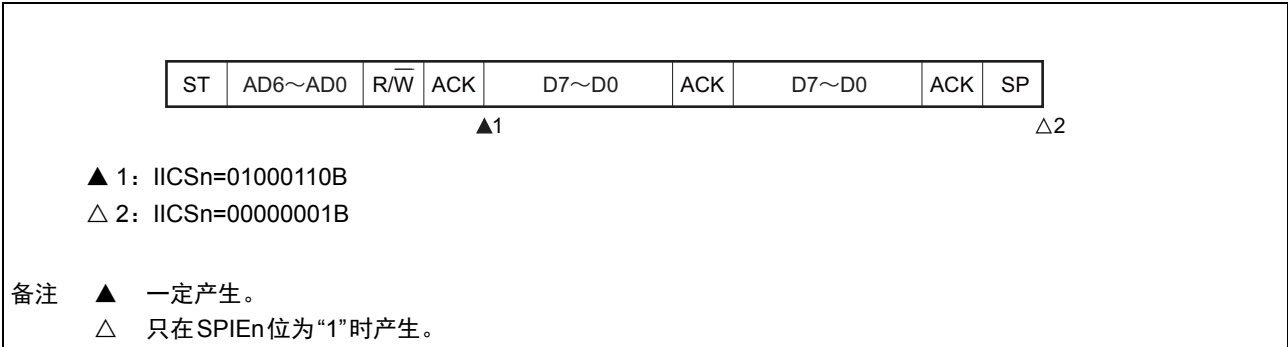
(ii) WTIMn=1 的情况



(6) 仲裁失败的运行（在仲裁失败后不参加通信）

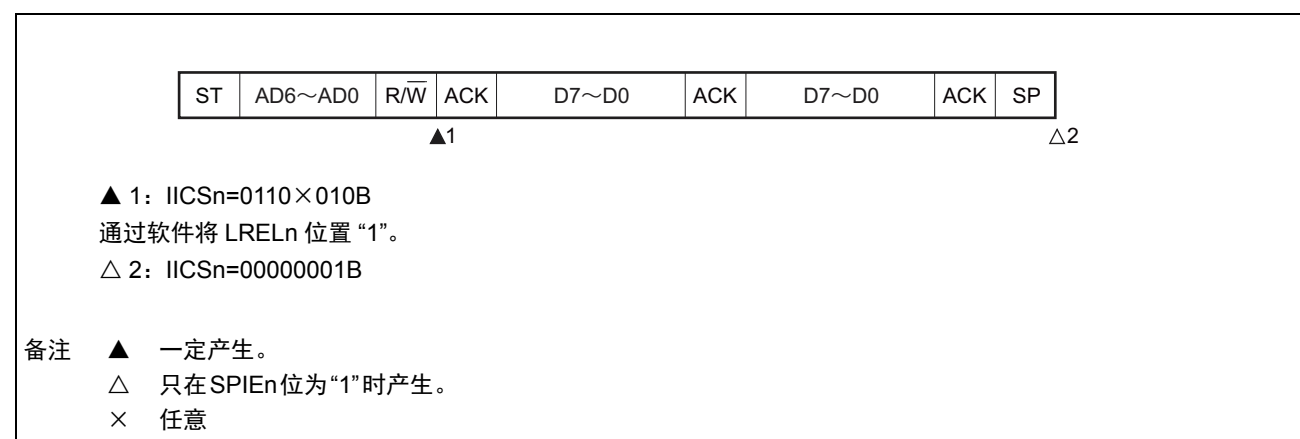
在多主控系统中用作主控设备时，必须在每次产生 INTIICAn 中断请求信号时读 MSTSn 位，确认仲裁结果。

(a) 在发送从属地址数据的过程中仲裁失败的情况（WTIMn=1）



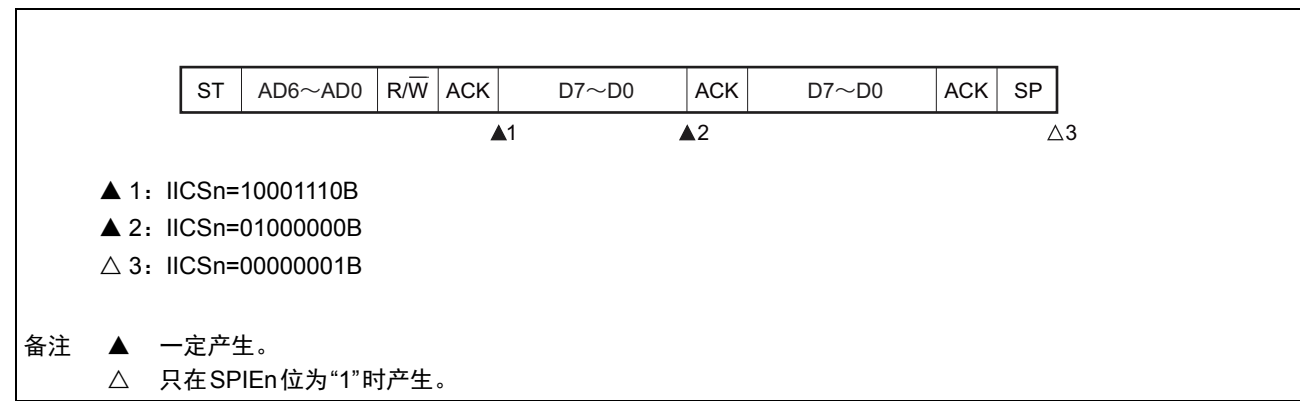
备注 n=0

(b) 在发送扩展码的过程中仲裁失败的情况



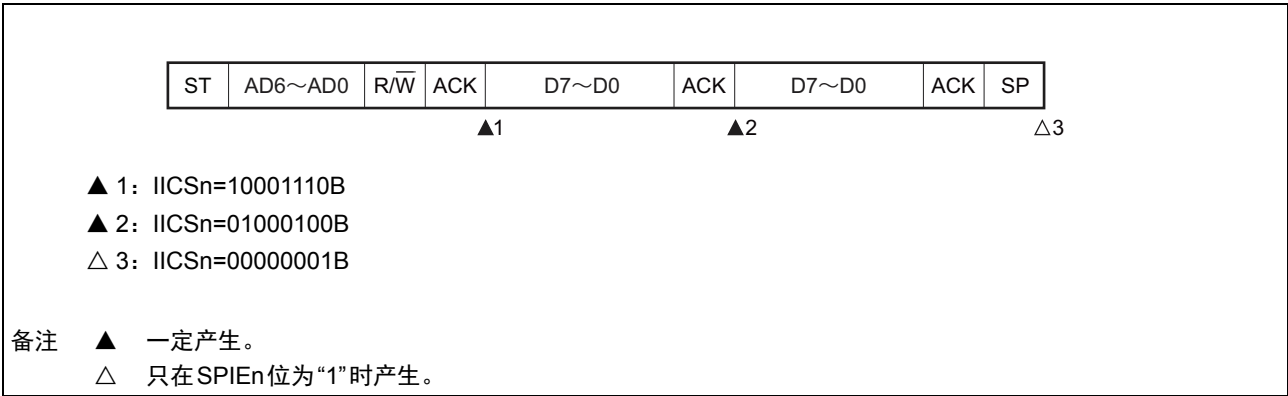
(c) 在发送数据的过程中仲裁失败的情况

(i) WTIMn=0 的情况



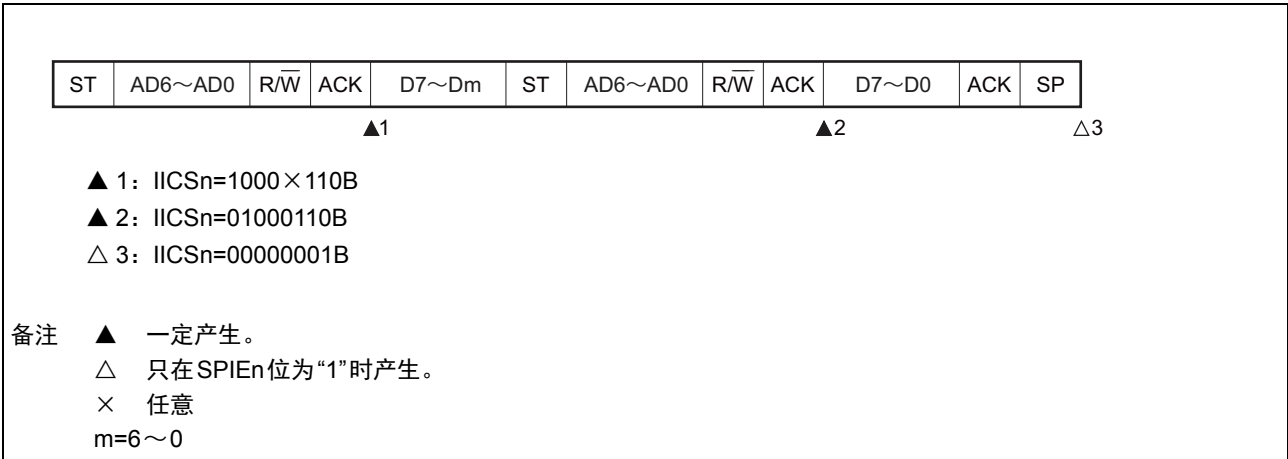
备注 n=0

(ii) WTIMn=1



(d) 在传送数据时因重新开始条件而仲裁失败的情况

(i) 非扩展码（例如，SVAn不同）



备注 n=0

(ii) 扩展码

ST	AD6~AD0	R/W	ACK	D7~Dm	ST	AD6~AD0	R/W	ACK	D7~D0	ACK	SP
				▲1				▲2			△3

▲ 1: IICSn=1000×110B
▲ 2: IICSn=01100010B
通过软件将 LRELn 位置“1”。
△ 3: IICSn=00000001B

备注 ▲ 一定产生。
△ 只在 SPIEn 位为“1”时产生。
× 任意
m=6~0

(e) 在传送数据时因停止条件而仲裁失败的情况

ST	AD6~AD0	R/W	ACK	D7~Dm	SP
				▲1	△2

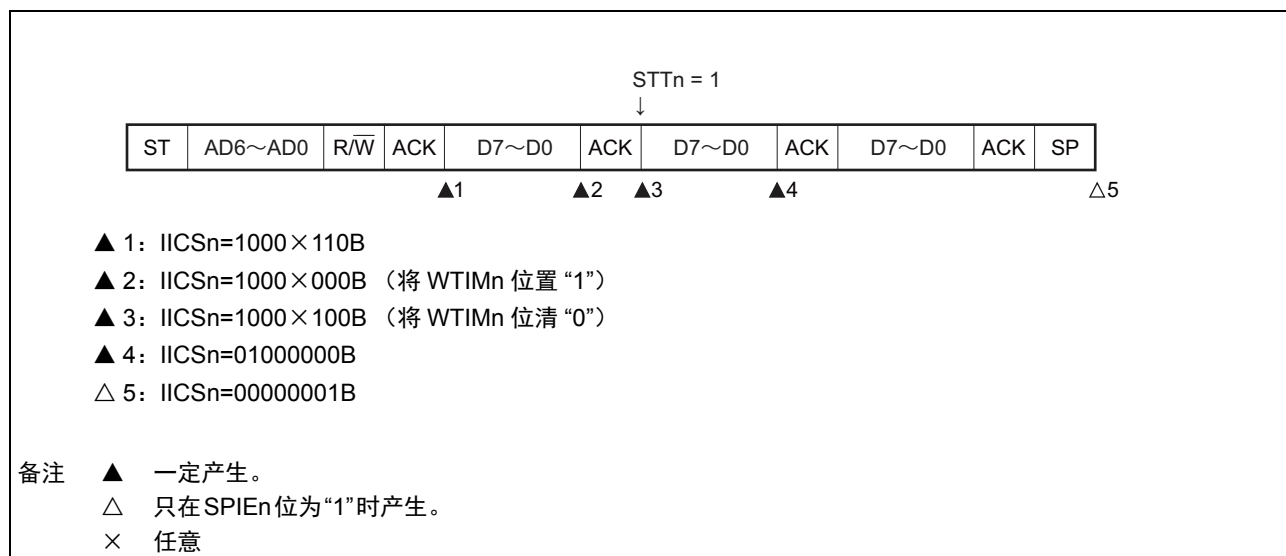
▲ 1: IICSn=10000110B
△ 2: IICSn=01000001B

备注 ▲ 一定产生。
△ 只在 SPIEn 位为“1”时产生。
× 任意
m=6~0

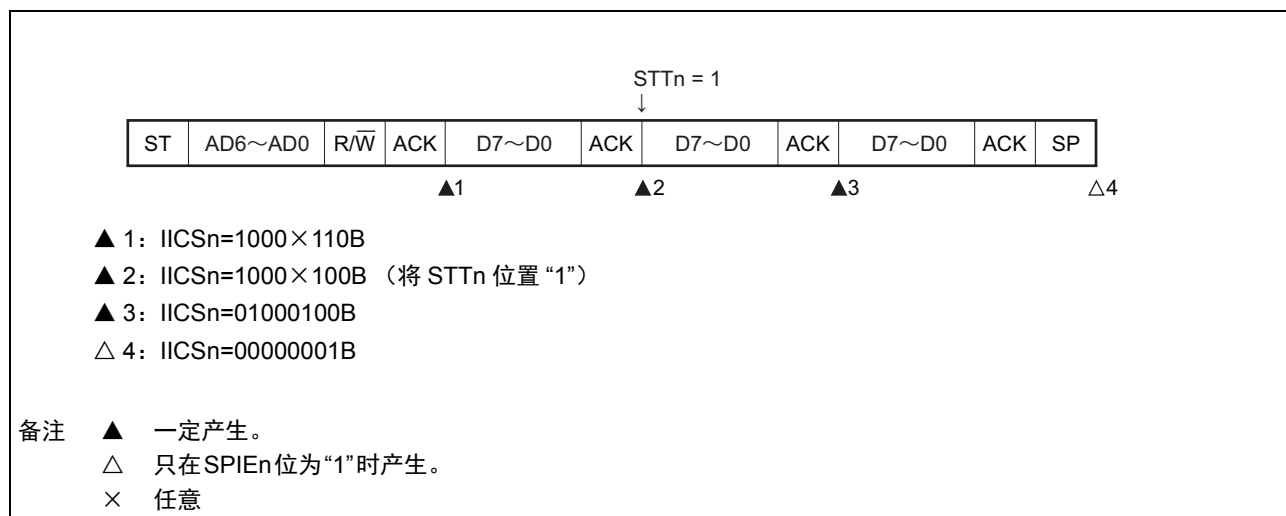
备注 n=0

(f) 在想生成重新开始条件时因数据为低电平而仲裁失败的情况

(i) WTIMn=0 的情况



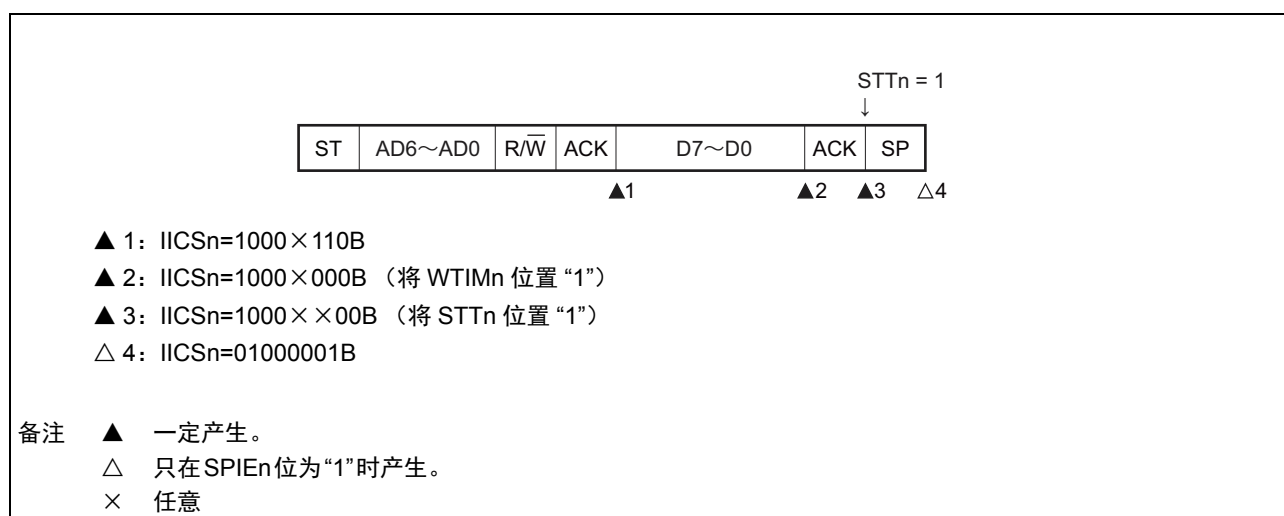
(ii) WTIMn=1 的情况



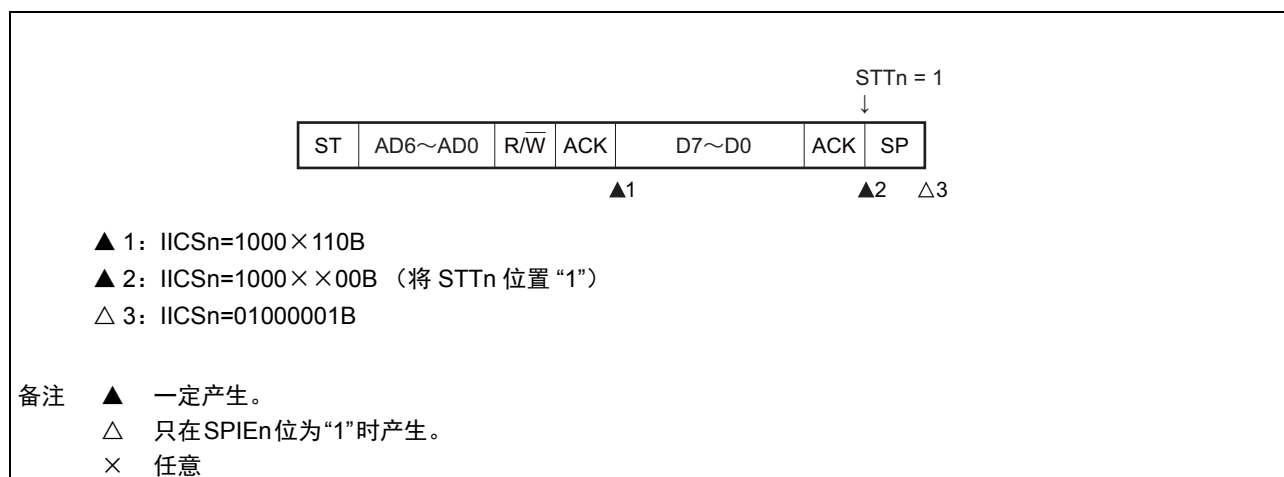
备注 n=0

(g) 在想生成重新开始条件时因停止条件而仲裁失败的情况

(i) WTIMn=0 的情况



(ii) WTIMn=1 的情况



备注 n=0

(h) 在想生成停止条件时因数据为低电平而仲裁失败的情况

(i) WTIMn=0 的情况

SPTn = 1
↓

ST	AD6~AD0	R/W	ACK	D7~D0	ACK	D7~D0	ACK	D7~D0	ACK	SP
			▲1		▲2	▲3		▲4		△5

▲ 1: IICSn=1000×110B
▲ 2: IICSn=1000×000B (将 WTIMn 位置“1”)
▲ 3: IICSn=1000×100B (将 WTIMn 位清“0”)
▲ 4: IICSn=01000100B
△ 5: IICSn=00000001B

备注 ▲ 一定产生。
 △ 只在 SPIEn 位为“1”时产生。
 × 任意

(ii) WTIMn=1 的情况

SPTn = 1
↓

ST	AD6~AD0	R/W	ACK	D7~D0	ACK	D7~D0	ACK	D7~D0	ACK	SP
			▲1		▲2		▲3			△4

▲ 1: IICSn=1000×110B
▲ 2: IICSn=1000×100B (将 SPTn 位置“1”)
▲ 3: IICSn=01000100B
△ 4: IICSn=00000001B

备注 ▲ 一定产生。
 △ 只在 SPIEn 位为“1”时产生。
 × 任意

备注 n=0

15.6 时序图

在 I²C 总线模式中，主控设备通过给串行总线输出地址，从多个从属设备选择一个通信对象的从属设备。

主控设备在从属设备地址之后发送表示数据传送方向的 TRCn 位（IICA 状态寄存器 n（IICSn）的 bit3），开始与从属设备进行串行通信。

数据通信的时序图如图 15-32 和图 15-33 所示。

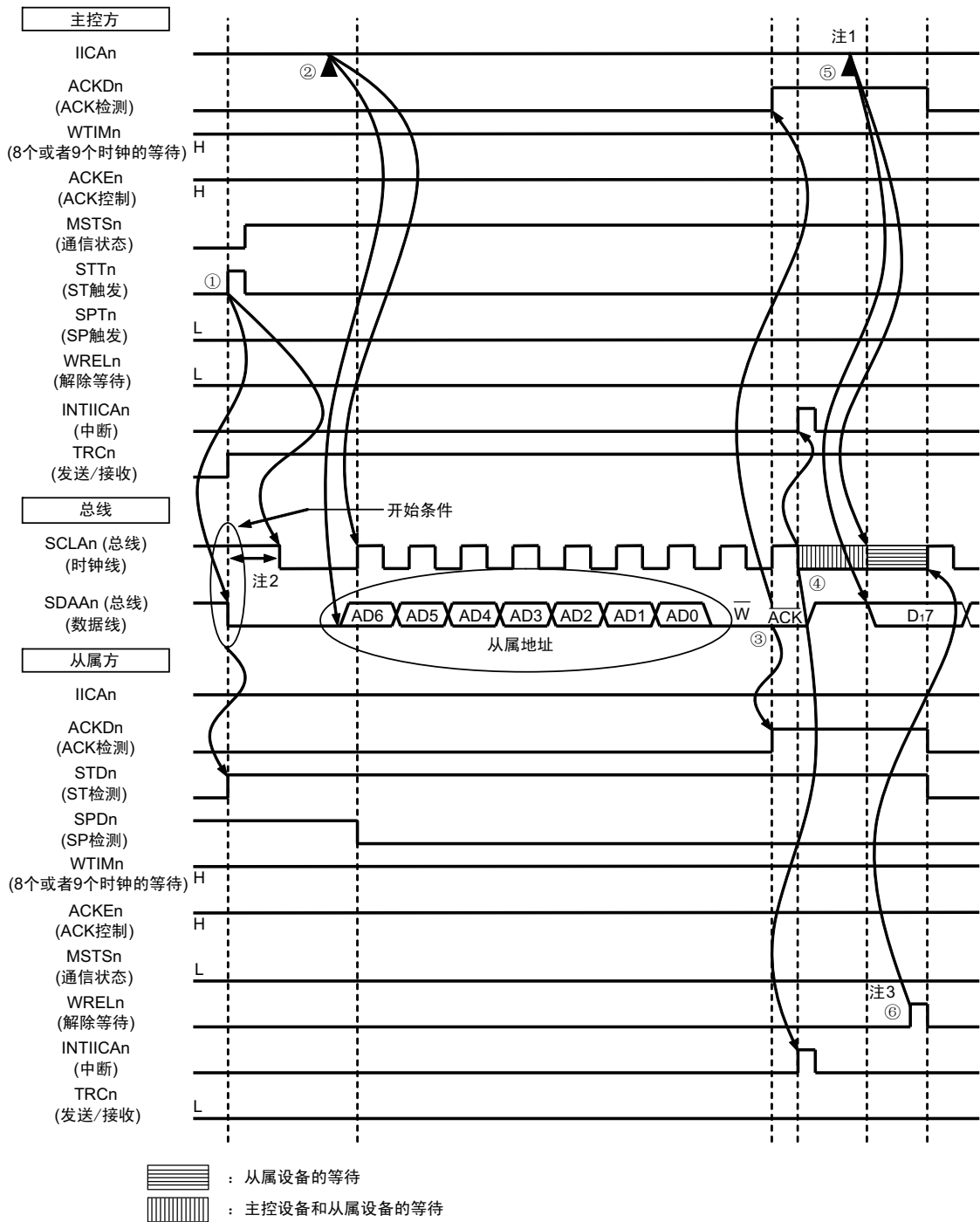
与串行时钟（SCLAn）的下降沿同步进行 IICA 移位寄存器 n（IICAn）的移位，并且将发送数据传送到 SO 锁存器，以 MSB 优先从 SDAAn 引脚输出数据。

在 SCLAn 的上升沿将 SDAAn 引脚输入的数据取到 IICAn。

备注 n=0

图 15-32 主控设备 → 从属设备的通信例子
(主控设备：选择 9 个时钟的等待，从属设备：选择 9 个时钟的等待) (1/4)

(1) 开始条件～地址～数据



- 注 1. 要在主控方的发送期间解除等待时，必须给 IICAn 写数据而不是将 WRELn 位置位。
2. 从 SDAAn 引脚信号下降到 SCLAn 引脚信号下降的时间，在设定为标准模式时至少为 4.0μs，在设定为快速模式时至少为 0.6μs。
3. 要在从属方的接收期间解除等待时，必须将 IICAn 置“FFH”或者将 WRELn 位置位。

备注 n=0

图 15-32 的“(1) 开始条件～地址～数据”的①～⑥的说明如下：

- ① 如果在主控方将开始条件触发置位（STTn=1），总线数据线（SDAAn）就下降，生成开始条件（通过 SCLAn=1 使 SDAAn 从“1”变为“0”）。此后，如果检测到开始条件，主控方就进入主控通信状态（MSTS_n=1），在经过保持时间后总线时钟线下降（SCLAn=0），结束通信准备。
- ② 如果主控方给 IICA 移位寄存器 n（IICAn）写地址+W（发送），就发送从属地址。
- ③ 在从属方，如果接收地址和本地站地址（SVAn 的值）相同注，就通过硬件给主控方发送 ACK。主控方在第 9 个时钟的上升沿检测到 ACK（ACKDn=1）。
- ④ 主控方在第 9 个时钟的下降沿产生中断（INTIICAn：地址发送结束中断）。相同地址的从属设备进入等待状态（SCLAn=0），并且产生中断（INTIICAn：地址匹配中断）注。
- ⑤ 主控方给 IICAn 寄存器写发送数据，解除主控方的等待。
- ⑥ 如果从属方解除等待（WRELn=1），主控方就开始给从属方传送数据。

注 如果发送的地址和从属地址不同，从属方就不给主控方返回 ACK（NACK：SDAAn=1），并且不产生 INTIICAn 中断（地址匹配中断），也不进入等待状态。

但是，主控方对于 ACK 和 NACK 都产生 INTIICAn 中断（地址发送结束中断）。

备注 1. 图 15-32 的①～⑮是通过 I²C 总线进行数据通信的一系列运行步骤。

图 15-32 的“(1) 开始条件～地址～数据”说明步骤①～⑥。

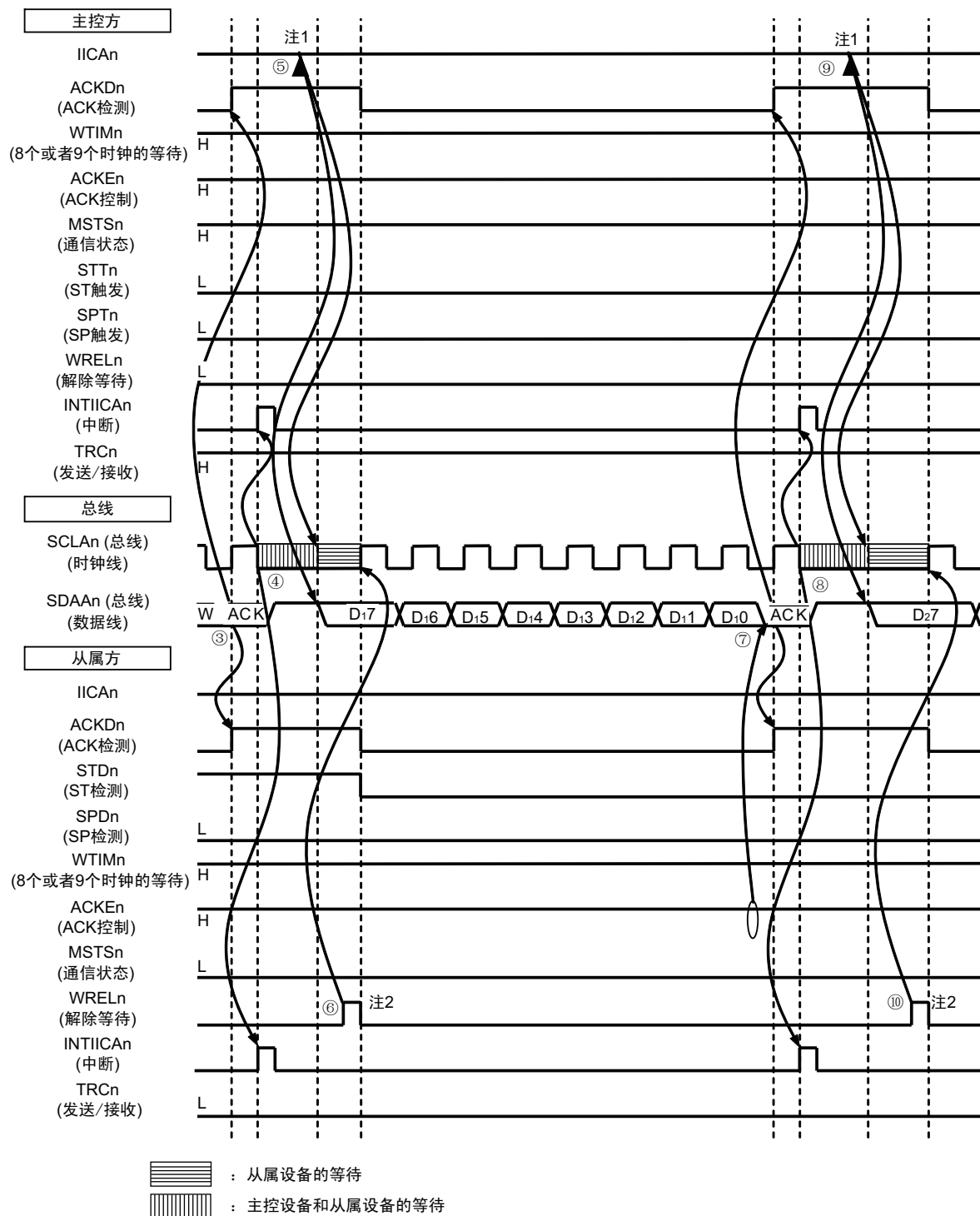
图 15-32 的“(2) 地址～数据～数据”说明步骤③～⑩。

图 15-32 的“(3) 数据～数据～停止条件”说明步骤⑦～⑮。

2. n=0

图 15-32 主控设备 → 从属设备的通信例子
 (主控设备: 选择 9 个时钟的等待, 从属设备: 选择 9 个时钟的等待) (2/4)

(2) 地址~数据~数据



- 注 1. 要在主控方的发送期间解除等待时, 必须给 IICAn 写数据而不是将 WRELn 位置位。
 2. 要在从属方的接收期间解除等待时, 必须将 IICAn 置 "FFH" 或者将 WRELn 位置位。

备注 n=0

图 15-32 的“(2) 地址~数据~数据”的③~⑩的说明如下:

- ③ 在从属方, 如果接收地址和本地站地址 (SVAn 的值) 相同^注, 就通过硬件给主控方发送 ACK。主控方在第 9 个时钟的上升沿检测到 ACK (ACKDn=1)。
- ④ 主控方在第 9 个时钟的下降沿产生中断 (INTIICAn: 地址发送结束中断)。相同地址的从属设备进入等待状态 (SCLAn=0), 并且产生中断 (INTIICAn: 地址匹配中断)^注。
- ⑤ 主控方给 IICA 移位寄存器 n (IICAn) 写发送数据, 解除主控方的等待。
- ⑥ 如果从属方解除等待 (WRELn=1), 主控方就开始给从属方传送数据。
- ⑦ 在数据传送结束后, 因为从属方的 ACKEn 位为“1”, 所以通过硬件给主控方发送 ACK。主控方在第 9 个时钟的上升沿检测到 ACK (ACKDn=1)。
- ⑧ 主控方和从属方在第 9 个时钟的下降沿进入等待状态 (SCLAn=0), 并且都产生中断 (INTIICAn: 传送结束中断)。
- ⑨ 主控方给 IICAn 寄存器写发送数据, 解除主控方的等待。
- ⑩ 如果从属方读接收数据并且解除等待 (WRELn=1), 主控方就开始给从属方传送数据。

注 如果发送的地址和从属地址不同, 从属方就不给主控方返回 ACK (NACK: SDAA_n=1), 并且不产生 INTIICAn 中断 (地址匹配中断), 也不进入等待状态。

但是, 主控方对于 ACK 和 NACK 都产生 INTIICAn 中断 (地址发送结束中断)。

备注 1. 图 15-32 的①~⑮是通过 I²C 总线进行数据通信的一系列运行步骤。

图 15-32 的“(1) 开始条件~地址~数据”说明步骤①~⑥。

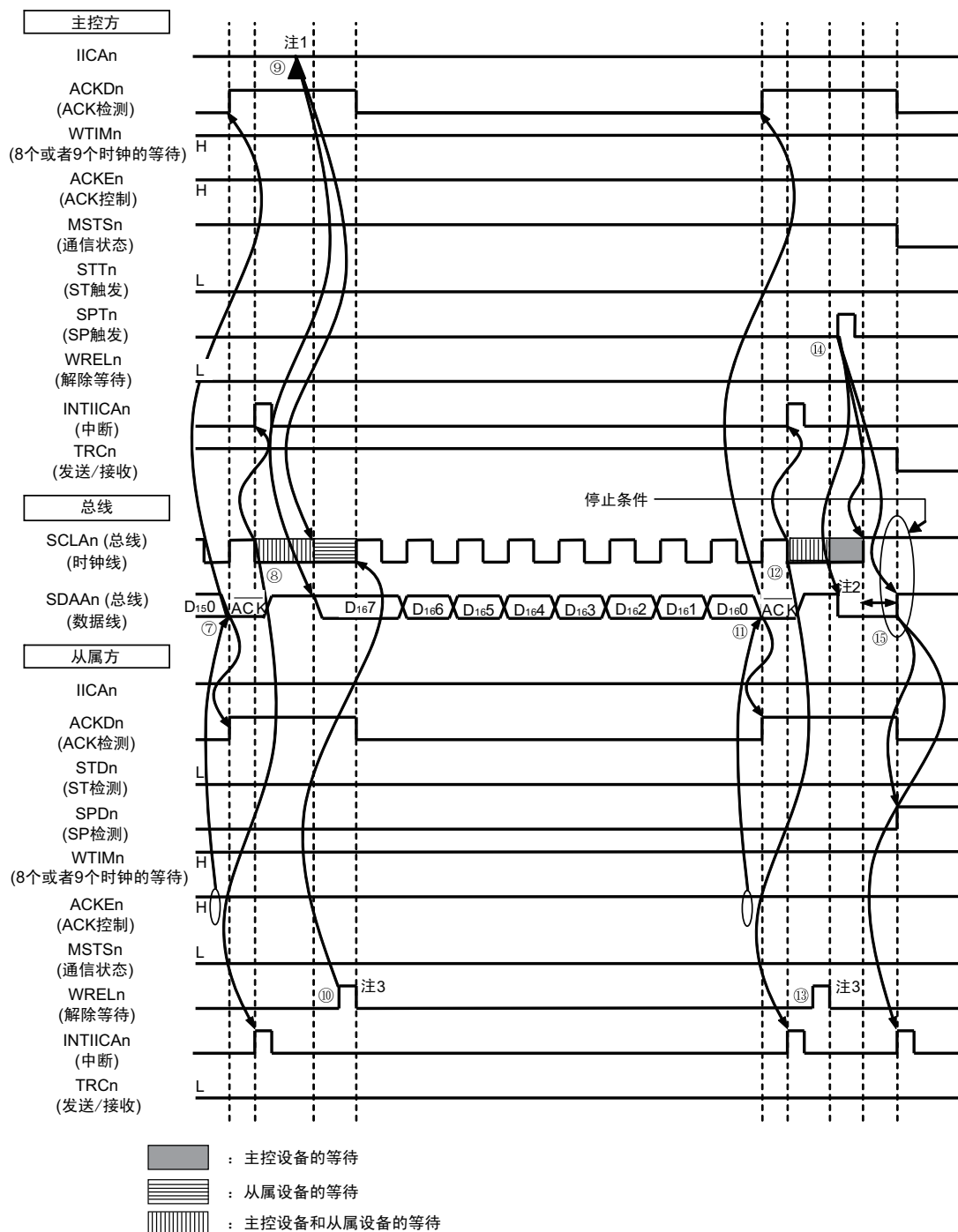
图 15-32 的“(2) 地址~数据~数据”说明步骤③~⑩。

图 15-32 的“(3) 数据~数据~停止条件”说明步骤⑦~⑮。

2. n=0

图 15-32 主控设备 → 从属设备的通信例子
(主控设备: 选择 9 个时钟的等待, 从属设备: 选择 9 个时钟的等待) (3/4)

(3) 数据~数据~停止条件



- 注 1. 要在主控方的发送期间解除等待时, 必须给 IICAn 写数据而不是将 WRELn 位置位。
2. 在发行停止条件后, 从 SCLAn 引脚信号上升到生成停止条件的时刻, 在设定为标准模式时至少为 4.0μs, 在设定为快速模式时至少为 0.6μs。
3. 要在从属方的接收期间解除等待时, 必须将 IICAn 置“FFH”或者将 WRELn 位置位。

备注 n=0

图 15-32 的“(3) 数据~数据~停止条件”的⑦~⑮的说明如下:

- ⑦ 在数据传送结束后, 因为从属方的 ACKEn 位为“1”, 所以通过硬件给主控方发送 ACK。主控方在第 9 个时钟的上升沿检测到 ACK (ACKDn=1)。
- ⑧ 主控方和从属方在第 9 个时钟的下降沿进入等待状态 (SCLAn=0), 并且都产生中断 (INTIICAn: 传送结束中断)。
- ⑨ 主控方给 IICA 移位寄存器 n (IICAn) 写发送数据, 解除主控方的等待。
- ⑩ 如果从属方读接收数据并且解除等待 (WRELn=1), 主控方就开始给从属方传送数据。
- ⑪ 在数据传送结束后, 从属方 (ACKEn=1) 通过硬件给主控方发送 ACK。主控方在第 9 个时钟的上升沿检测到 ACK (ACKDn=1)。
- ⑫ 主控方和从属方在第 9 个时钟的下降沿进入等待状态 (SCLAn=0), 并且都产生中断 (INTIICAn: 传送结束中断)。
- ⑬ 从属方读接收数据, 解除等待 (WRELn=1)。
- ⑭ 如果在主控方将停止条件触发置位 (SPTn=1), 就清除总线数据线 (SDAAn=0) 并且将总线时钟线置位 (SCLAn=1), 在经过停止条件的准备时间后将总线数据线置位 (SDAAn=1), 生成停止条件 (通过 SCLAn=1 使 SDAAn 从“0”变为“1”)。
- ⑮ 如果生成停止条件, 从属方就检测到停止条件并且产生中断 (INTIICAn: 停止条件中断)。

备注 1. 图 15-32 的①~⑮是通过 I²C 总线进行数据通信的一系列运行步骤。

图 15-32 的“(1) 开始条件~地址~数据”说明步骤①~⑥。

图 15-32 的“(2) 地址~数据~数据”说明步骤③~⑩。

图 15-32 的“(3) 数据~数据~停止条件”说明步骤⑦~⑮。

2. n=0

图 15-32 主控设备 → 从属设备的通信例子
(主控设备: 选择 9 个时钟的等待, 从属设备: 选择 9 个时钟的等待) (4/4)

(4) 数据~重新开始条件~地址

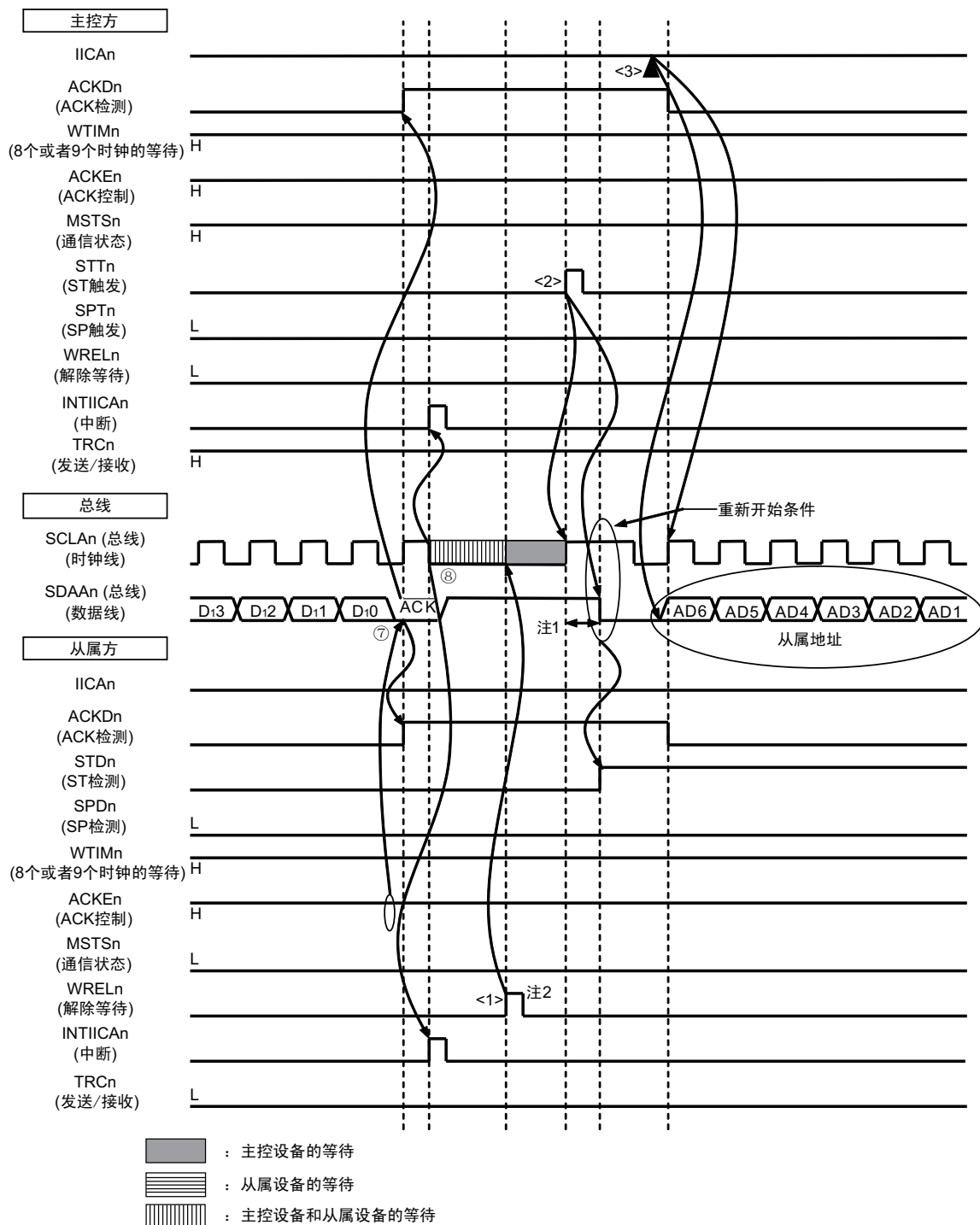


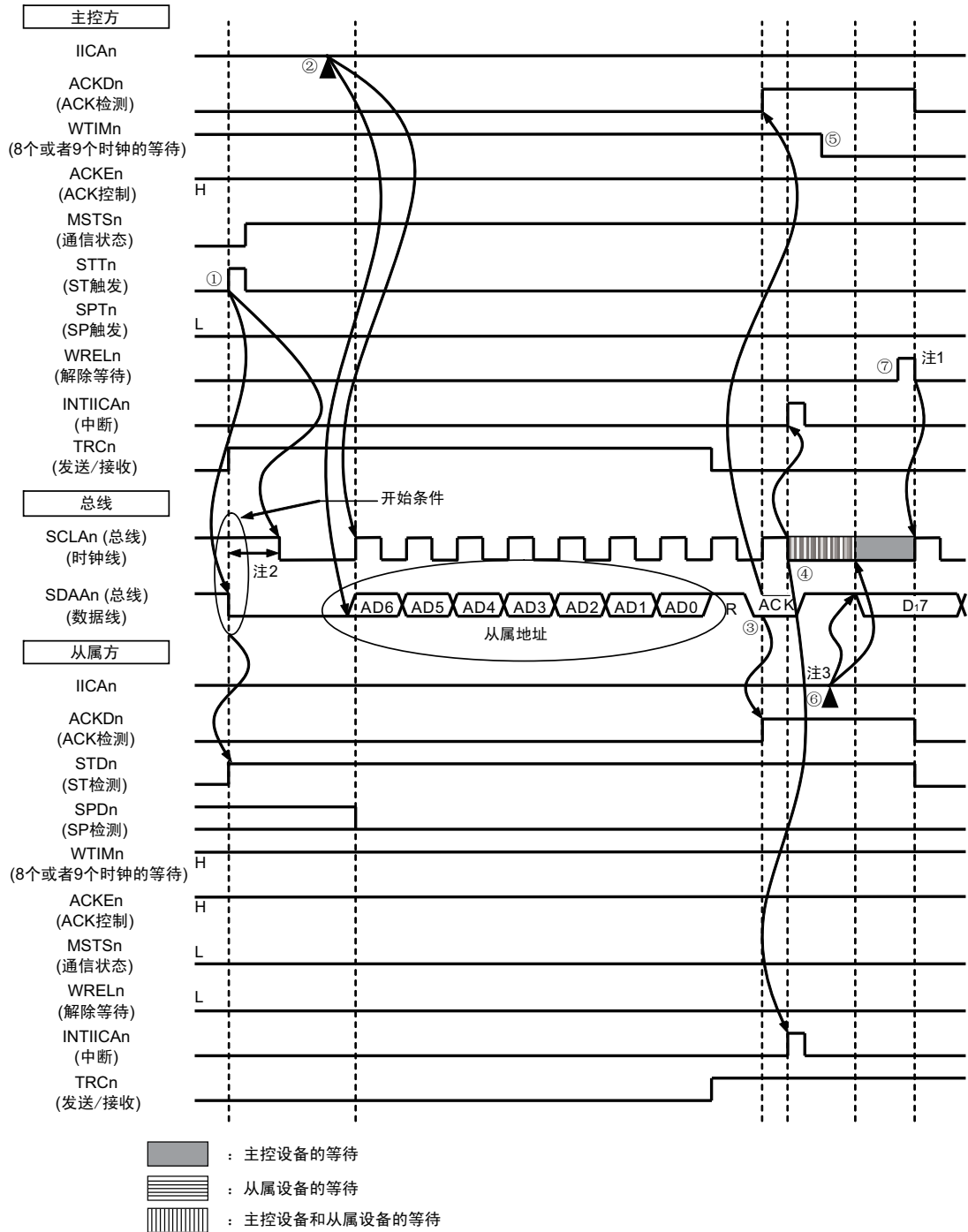
图 15-32 的“(4) 数据～重新开始条件～地址”的运行说明如下。在执行步骤⑦和⑧后执行<1>～<3>，从而返回步骤③的数据发送步骤。

- ⑦ 在数据传送结束后，因为从属方的 ACKEn 位为“1”，所以通过硬件给主控方发送 ACK。主控方在第 9 个时钟的上升沿检测到 ACK (ACKDn=1)。
- ⑧ 主控方和从属方在第 9 个时钟的下降沿进入等待状态 (SCLAn=0)，并且都产生中断 (INTIICAn: 传送结束中断)。
- <1> 从属方读接收数据，解除等待 (WRELn=1)。
- <2> 如果在主控方再次将开始条件触发置位 (STTn=1)，总线时钟线就上升 (SCLAn=1)，而且在经过重新开始条件的准备时间后总线数据线下降 (SDAAn=0)，生成开始条件 (通过 SCLAn=1 使 SDAAn 从“1”变为“0”)。然后，如果检测到开始条件，就在经过保持时间后总线时钟线下降 (SCLAn=0)，结束通信准备。
- <3> 如果主控方给 IICA 移位寄存器 n (IICAn) 写地址 +R/W (发送)，就发送从属地址。

备注 n=0

图 15-33 从属设备 → 主控设备的通信例子
(主控设备：选择 8 个时钟等待，从属设备：选择 9 个时钟等待) (1/3)

(1) 开始条件～地址～数据



- 注 1. 要在主控方的接收期间解除等待时，必须将 IICAn 置“FFH”或者将 WRELn 位置位。
2. 从 SDAAn 引脚信号下降到 SCLAn 引脚信号下降的时间，在设定为标准模式时至少为 4.0μs，在设定为快速模式时至少为 0.6μs。
3. 要在从属方的发送期间解除等待时，必须给 IICAn 写数据而不是将 WRELn 位置位。

备注 n=0

图 15-33 的“(1) 开始条件～地址～数据”的①～⑦的说明如下：

- ① 如果在主控方将开始条件触发置位（STTn=1），总线数据线（SDAAn）就下降，生成开始条件（通过 SCLAn=1 使 SDAAn 从“1”变为“0”）。此后，如果检测到开始条件，主控方就进入主控通信状态（MSTS_n=1），在经过保持时间后总线时钟线下降（SCLAn=0），结束通信准备。
- ② 如果主控方给 IICA 移位寄存器 n（IICAn）写地址+R（接收），就发送从属地址。
- ③ 在从属方，如果接收地址和本地站地址（SVAn 的值）相同注，就通过硬件给主控方发送 ACK。主控方在第 9 个时钟的上升沿检测到 ACK（ACKDn=1）。
- ④ 主控方在第 9 个时钟的下降沿产生中断（INTIICAn：地址发送结束中断）。相同地址的从属设备进入等待状态（SCLAn=0），并且产生中断（INTIICAn：地址匹配中断）注。
- ⑤ 主控方将等待时序更改为第 8 个时钟（WTIMn=0）。
- ⑥ 从属方给 IICAn 寄存器写发送数据，解除从属方的等待。
- ⑦ 主控方解除等待（WRELn=1），开始来自从属设备的数据传送。

注 如果发送的地址和从属地址不同，从属方就不给主控方返回 ACK（NACK：SDAAn=1），并且不产生 INTIICAn 中断（地址匹配中断），也不进入等待状态。

但是，主控方对于 ACK 和 NACK 都产生 INTIICAn 中断（地址发送结束中断）。

备注 1. 图 15-33 的①～⑱是通过 I²C 总线进行数据通信的一系列运行步骤。

图 15-33 的“(1) 开始条件～地址～数据”说明步骤①～⑦。

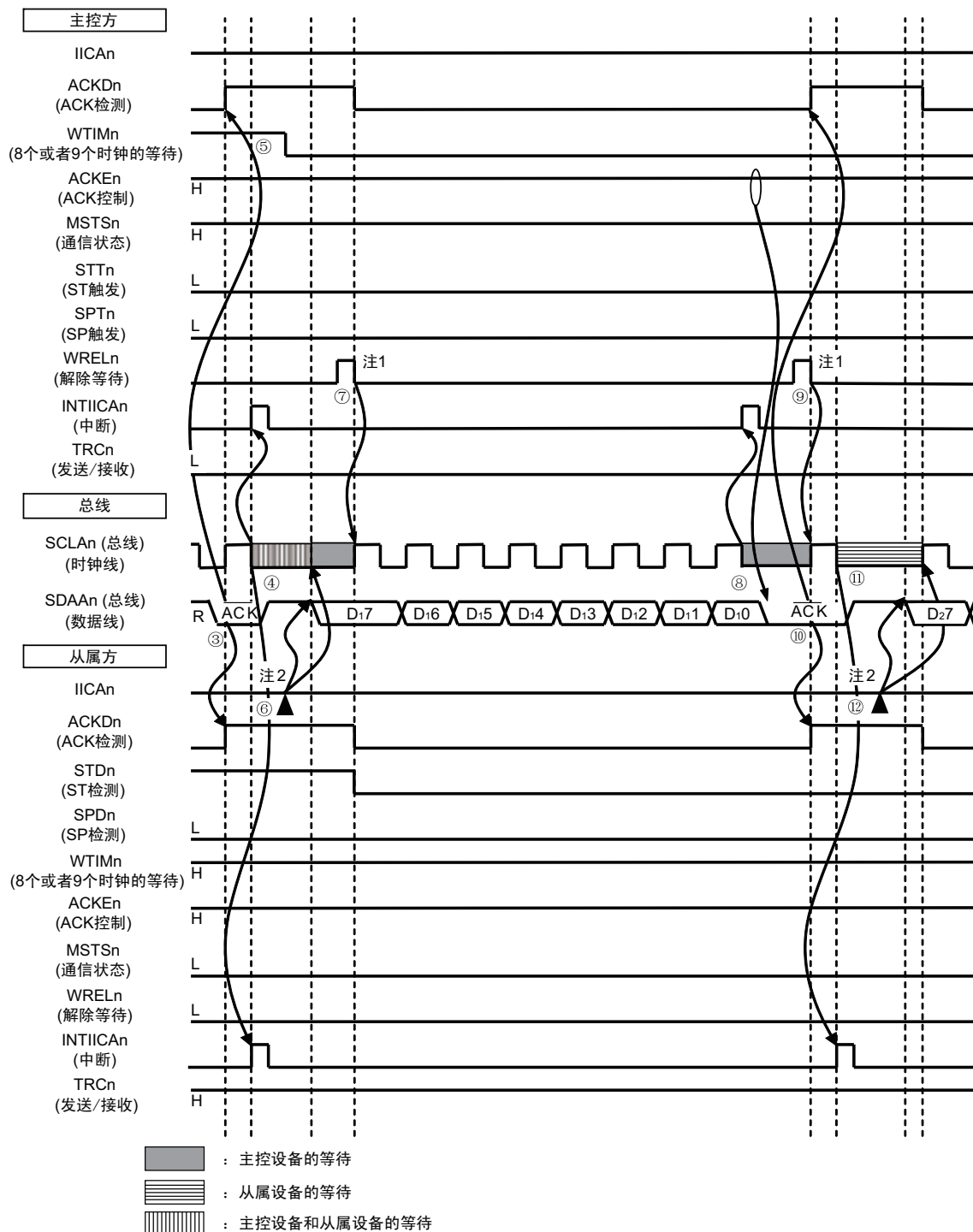
图 15-33 的“(2) 地址～数据～数据”说明步骤③～⑫。

图 15-33 的“(3) 数据～数据～停止条件”说明步骤⑧～⑱。

2. n=0

图 15-33 从属设备 → 主控设备的通信例子
 (主控设备：选择 8 个时钟等待，从属设备：选择 9 个时钟等待) (2/3)

(2) 地址~数据~数据



- 注 1. 要在主控方的接收期间解除等待时，必须将 IICAn 置“FFH”或者将 WRELn 位置位。
 2. 要在从属方的发送期间解除等待时，必须给 IICAn 写数据而不是将 WRELn 位置位。

备注 n=0

图 15-33 的“(2) 地址~数据~数据”的③~⑫的说明如下:

- ③ 在从属方, 如果接收地址和本地站地址 (SVAn 的值) 相同注, 就通过硬件给主控方发送 ACK。主控方在第 9 个时钟的上升沿检测到 ACK (ACKDn=1)。
- ④ 主控方在第 9 个时钟的下降沿产生中断 (INTIICAn: 地址发送结束中断)。相同地址的从属设备进入等待状态 (SCLAn=0), 并且产生中断 (INTIICAn: 地址匹配中断)注。
- ⑤ 主控方将等待时序改为第 8 个时钟 (WTIMn=0)。
- ⑥ 从属方给 IICA 移位寄存器 n (IICAn) 写发送数据, 解除从属方的等待。
- ⑦ 主控方解除等待 (WRELn=1), 开始来自从属设备的数据传送。
- ⑧ 主控方在第 8 个时钟的下降沿进入等待状态 (SCLAn=0), 并且产生中断 (INTIICAn: 传送结束中断)。因为主控方的 ACKEn 位为“1”, 所以通过硬件给从属方发送 ACK。
- ⑨ 主控方读接收数据, 解除等待 (WRELn=1)。
- ⑩ 从属方在第 9 个时钟的上升沿检测到 ACK (ACKDn=1)。
- ⑪ 从属方在第 9 个时钟的下降沿进入等待状态 (SCLAn=0), 并且产生中断 (INTIICAn: 传送结束中断)。
- ⑫ 如果从属方给 IICAn 寄存器写发送数据, 就解除从属方的等待, 开始从属设备到主控设备的数据传送。

注 如果发送的地址和从属地址不同, 从属方就不给主控方返回 ACK (NACK: SDAAn=1), 并且不产生 INTIICAn 中断 (地址匹配中断), 也不进入等待状态。

但是, 主控方对于 ACK 和 NACK 都产生 INTIICAn 中断 (地址发送结束中断)。

备注 1. 图 15-33 的①~⑭是通过 I²C 总线进行数据通信的一系列运行步骤。

图 15-33 的“(1) 开始条件~地址~数据”说明步骤①~⑦。

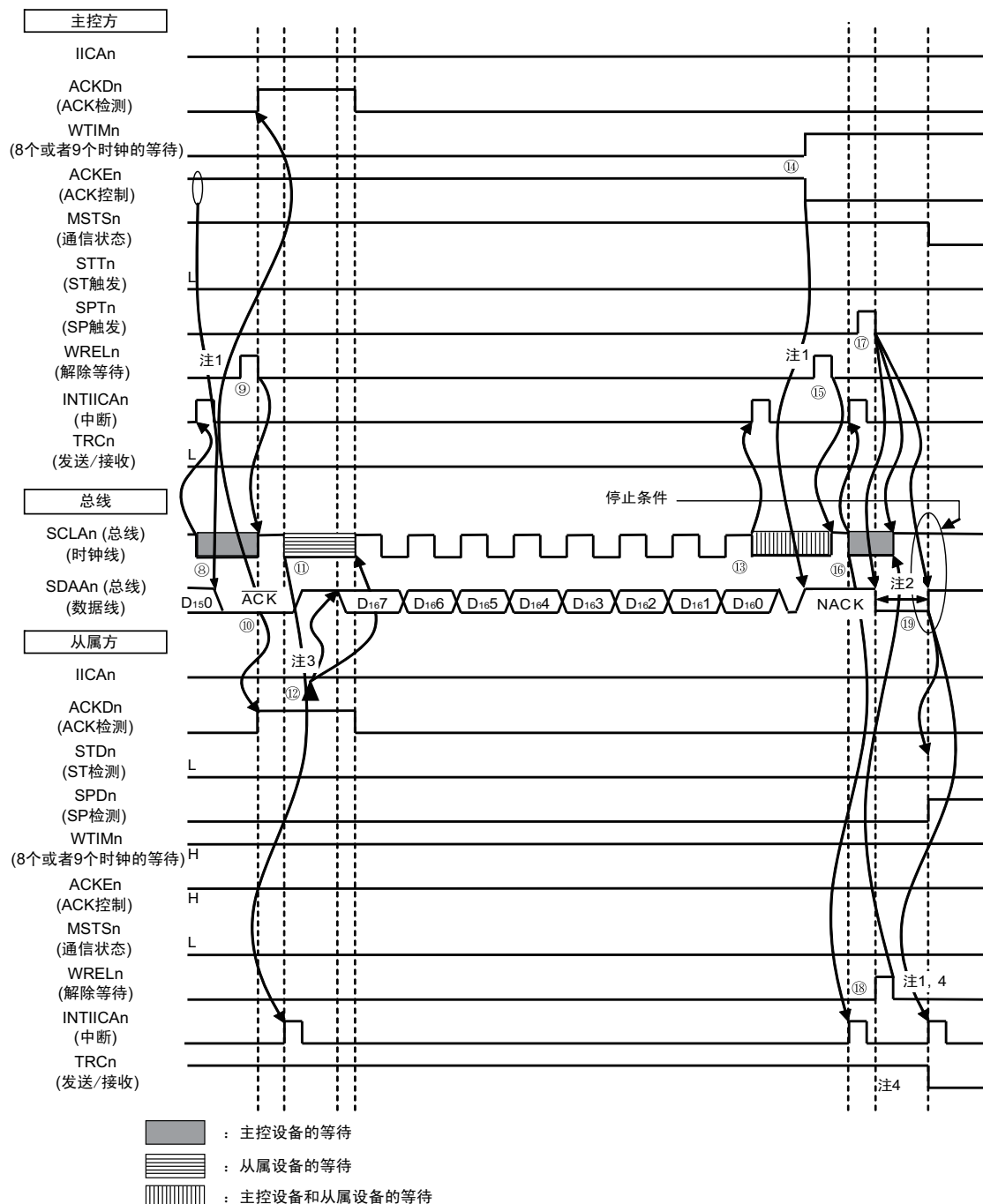
图 15-33 的“(2) 地址~数据~数据”说明步骤③~⑫。

图 15-33 的“(3) 数据~数据~停止条件”说明步骤⑧~⑭。

2. n=0

图 15-33 从属设备 → 主控设备的通信例子
(主控设备: 选择 8 个 → 9 个时钟等待, 从属设备: 选择 9 个时钟等待) (3/3)

(3) 数据~数据~停止条件



- 注 1. 要解除等待时, 必须将 IICAn 置 "FFH" 或者将 WRELn 位置位。
2. 在发行停止条件后, 从 SCLAn 引脚信号上升到生成停止条件的的时间, 在设定为标准模式时至少为 4.0μs, 在设定为快速模式时至少为 0.6μs。
3. 要在从属方的发送期间解除等待时, 必须给 IICAn 写数据而不是将 WRELn 位置位。
4. 在从属方的发送期间, 如果通过 WRELn 位的置位来解除等待, 就清除 TRCn 位。

备注 n=0

图 15-33 的“(3) 数据～数据～停止条件”的⑧～⑲的说明如下：

- ⑧ 主控方在第 8 个时钟的下降沿进入等待状态（ $SCLAn=0$ ），并且产生中断（INTIICAn：传送结束中断）。因为主控方的 $ACKEn$ 位为“0”，所以通过硬件给从属方发送 ACK。
- ⑨ 主控方读接收数据，解除等待（ $WRELn=1$ ）。
- ⑩ 从属方在第 9 个时钟的上升沿检测到 ACK（ $ACKDn=1$ ）。
- ⑪ 从属方在第 9 个时钟的下降沿进入等待状态（ $SCLAn=0$ ），并且产生中断（INTIICAn：传送结束中断）。
- ⑫ 如果从属方给 IICA 移位寄存器 n（IICAn）写发送数据，就解除从属方的等待，开始从属设备到主控设备的数据传送。
- ⑬ 主控方在第 8 个时钟的下降沿产生中断（INTIICAn：传送结束中断），并且进入等待状态（ $SCLAn=0$ ）。因为进行 ACK 控制（ $ACKEn=1$ ），所以此阶段的总线数据线变为低电平（ $SDAAn=0$ ）。
- ⑭ 主控方设定为 NACK 应答（ $ACKEn=0$ ），并且将等待时序更改为第 9 个时钟（ $WTIMn=1$ ）。
- ⑮ 如果主控方解除等待（ $WRELn=1$ ），从属方就在第 9 个时钟的上升沿检测到 NACK（ $ACKDn=0$ ）。
- ⑯ 主控方和从属方在第 9 个时钟的下降沿进入等待状态（ $SCLAn=0$ ），并且都产生中断（INTIICAn：传送结束中断）。
- ⑰ 如果主控方发行停止条件（ $SPTn=1$ ），就清除总线数据线（ $SDAAn=0$ ），并且解除主控方的等待。此后，主设备处于待机状态，直到将总线时钟线置位（ $SCLAn=1$ ）为止。
- ⑱ 从属方在确认 NACK 后停止发送，为了结束通信，解除等待（ $WRELn=1$ ）。如果解除从属方的等待，就将总线时钟线置位（ $SCLAn=1$ ）。
- ⑲ 如果主控方确认到总线时钟线被置位（ $SCLAn=1$ ），就在经过停止条件准备时间后将总线数据线置位（ $SDAAn=1$ ），然后发行停止条件（通过 $SCLAn=1$ 使 $SDAAn$ 从“0”变为“1”）。如果生成停止条件，从属方就检测到停止条件，并且产生中断（INTIICAn：停止条件中断）。

备注 1. 图 15-33 的①～⑲是通过 I²C 总线进行数据通信的一系列运行步骤。

图 15-33 的“(1) 开始条件～地址～数据”说明步骤①～⑦。

图 15-33 的“(2) 地址～数据～数据”说明步骤③～⑫。

图 15-33 的“(3) 数据～数据～停止条件”说明步骤⑧～⑲。

2. $n=0$

第 16 章 智能卡接口 (SMCI)

16.1 智能卡接口 (SMCI) 的功能

智能卡接口 (SMCI) 进行异步串行通信, 是支持符合 ISO/IEC 7816-3 (Identification Card) 的智能卡 (IC 卡) 接口。

智能卡接口的特征如下所示。

- 智能卡接口的主控功能 (提供输出时钟)
- 异步串行数据通信。
 - 数据长度: 8 位
 - 如果在接收时检测到奇偶校验错误, 就自动发送错误信号 (奇偶校验错误)。
 - 如果在发送时接收到错误信号, 就自动重新发送数据。
 - 支持正向约定和反向约定
- 可通过内置波特率发生器选择任意的波特率
- 前导功能
- 中断源: 3 个
有发送结束、接收数据满、接收错误 3 个中断源。
另外, 可通过发送结束中断和接收数据满中断启动 DMA。

16.2 智能卡接口 (SMCI) 的结构

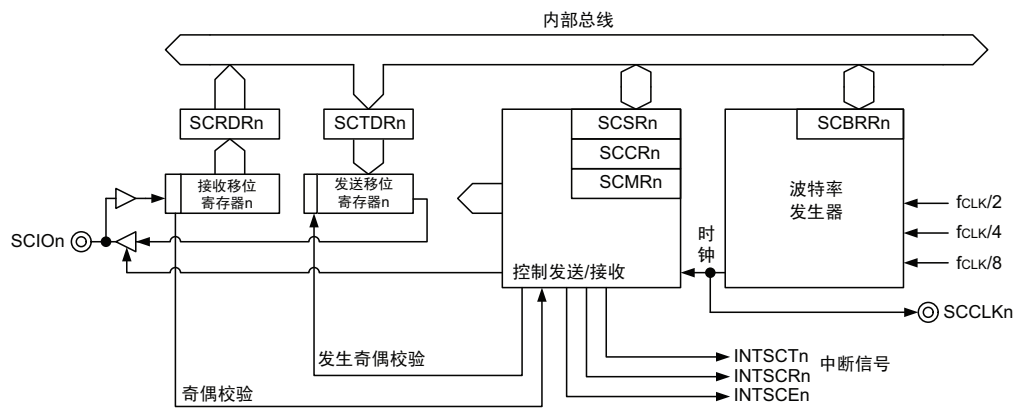
智能卡接口 (SMCI) 由以下硬件构成。

表 16-1 智能卡接口的结构

项目	结构
移位寄存器	接收移位寄存器 n、发送移位寄存器 n
缓冲寄存器	智能卡接收数据寄存器 n (SCRDRn) 智能卡发送数据寄存器 n (SCTDRn)
串行时钟输出	SCCLKn
串行数据输入 / 输出	SCION
控制寄存器	外围允许寄存器 1 (PER1)
	智能卡串行模式寄存器 n (SCMRn)
	智能卡串行控制寄存器 n (SCCRn)
	智能卡串行状态寄存器 n (SCSRn)
	智能卡位速率寄存器 n (SCBRRn)
	端口输入模式寄存器 5 (PIM5)
	端口输出模式寄存器 5 (POM5)
	端口模式寄存器 5 (PM5)
	端口寄存器 5 (P5)

SMCI 的框图如图 16-1 所示。

图 16-1 SMCI 的框图



备注 n: 通道号 (n = 0、1)

16.2.1 接收移位寄存器 n

接收移位寄存器 n 将从 SCIO_n 引脚输入的串行数据转换为并行数据。如果接收到 1 帧的数据，就将该数据自动传送到 SCRDR_n。CPU 不能直接访问接收移位寄存器 n。

图 16-2 接收移位寄存器 n 的格式

地址：— 复位后：00H

7	6	5	4	3	2	1	0
接收移位寄存器 n [7:0]							

接收移位寄存器 n [7:0]	功能
bit7 ~ 0	用于接收的移位寄存器

16.2.2 发送移位寄存器 n

发送移位寄存器 n 用于发送串行数据。通过将写入 SCTDR_n 寄存器的发送数据自动发送到发送移位寄存器 n，再发送到 SCIO_n 引脚来执行串行数据的发送。CPU 不能直接访问发送移位寄存器 n。

图 16-3 发送移位寄存器 n 的格式

地址：— 复位后：FFH

7	6	5	4	3	2	1	0
发送移位寄存器 n [7:0]							

发送移位寄存器 n [7:0]	功能
bit7 ~ 0	用于发送的移位寄存器

备注 n: 通道号 (n = 0、1)

16.2.3 智能卡接收数据寄存器 n（SCRDRn）

SCRDRn 寄存器是用于保存接收数据的 8 位寄存器。如果接收到 1 帧的数据，就将接收数据从接收移位寄存器 n 传送到该寄存器，并且接收移位寄存器 n 可接收下一个数据。

必须在确认 SCSSRn 的 SCRDRF 标志设定为“1”后才能读 SCRDRn 寄存器。

通过 8 位存储器操作指令读 SCRDRn 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 16-4 智能卡接收数据寄存器 n（SCRDRn）的格式

地址：FFF58H（SCRDR0）、FFF5AH（SCRDR1）		复位后：00H		R				
符号	7	6	5	4	3	2	1	0
SCRDRn	SCRDRn [7:0]							
SCRDRn [7:0]				功能				
bit7 ~ 0				用于接收的寄存器				

16.2.4 智能卡发送数据寄存器 n（SCTDRn）

SCTDRn 寄存器是用于保存发送数据的 8 位寄存器。如果检测到发送移位寄存器 n 为空，就将已写入 SCTDRn 的发送数据发送到发送移位寄存器 n，开始发送。如果在发送 1 帧的数据后，SCTDRn 中已写入下一个发送数据，就将该发送数据传送到发送移位寄存器 n，继续发送。CPU 可以随时读 SCTDRn 寄存器。必须在确认 SCSRn 寄存器的 SCTDRE 标志设定为“1”后才能给 SCTDRn 寄存器写发送数据。

通过 8 位存储器操作指令设定 SCTDRn 寄存器。

在产生复位信号后，此寄存器的值变为“FFH”。

图 16-5 智能卡发送数据寄存器 n（SCTDRn）的格式

地址：FFF59H （SCTDR0）、FFF5BH （SCTDR1）		复位后：FFH		R/W				
符号	7	6	5	4	3	2	1	0
SCTDRn	SCTDRn [7:0]							
SCTDRn [7:0]				功能				
bit7 ～ 0				用于发送的寄存器				

备注 n: 通道号（n = 0、1）

16.3 控制智能卡接口 SMCI 的寄存器

通过以下寄存器控制智能卡接口（SMCI）。

- 外围允许寄存器 1（PER1）
- 智能卡串行模式寄存器 n（SCMRn）
- 智能卡串行控制寄存器 n（SCCRn）
- 智能卡串行状态寄存器 n（SCSRn）
- 智能卡位速率寄存器 n（SCBRRn）
- 端口输入模式寄存器 5（PIM5）
- 端口输出模式寄存器 5（POM5）
- 端口模式寄存器 5（PM5）
- 端口寄存器 5（P5）

备注 n：通道号（n = 0、1）

16.3.1 外围允许寄存器 1（PER1）

PER1 寄存器设定允许或禁止向各外围硬件提供时钟。通过停止向未使用的硬件提供时钟，降低功耗和噪声。

当使用 SMCI_n 时，必须将 bit2 和 bit1（SMCI1EN、SMCI0EN）设定为“1”。

通过 1 位或者 8 位存储器操作指令设定 PER1 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 16-6 外围允许寄存器 1（PER1）的格式

地址：F007AH 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
PER1	TMKAEN	FMCEN	CMPEN	0	0	SMCI1EN	SMCI0EN	0

SMCI _n EN	控制智能卡接口 n 的输入时钟的提供
0	停止提供输入时钟。 • 不能写智能卡接口 n 使用的 SFR。 • 智能卡接口 n 处于复位状态。
1	提供输入时钟。 • 能读写智能卡接口 n 的 SFR。

注意 1. 当不使用 SMCI 功能时，必须将 PER1 寄存器的 SMCI_nEN 位设定为“0”（停止提供输入时钟）。

2. 必须将 bit4、3、0 设定为“0”。

备注 n：通道号（n = 0、1）

16.3.2 智能卡串行模式寄存器 n (SCMRn)

SCMRn 寄存器选择通信格式和内置波特率发生器的时钟源。

通过 8 位存储器操作指令设定 SCMRn 寄存器。

在产生复位信号后，此寄存器的值变为“18H”。

图 16-7 智能卡串行模式寄存器 n (SCMRn) 的格式 (1/2)

地址: F0322H (SCMR0)、F032AH (SCMR1) 复位后: 18H R/W

符号	7	6	5	4	3	2	1	0
SCMRn 注	SCGSM	SCBLK	SCOE	SCBCP2	SCBCP1	SCBCP0	SCCKS1	SCCKS0

SCGSM	GSM 模式的选择
0	以通常模式运行
1	以 GSM 模式运行
如果将 SCGSM 位设定为“1”就以 GSM 模式运行。 在 GSM 模式中，将 SCTEND 标志的置位时序从起始开始提前 11.0 etu (eut: Elementary Time Unit, 1 位的传送时间)，并且追加时钟输出控制功能。 详细内容请参照“16.4.8 时钟输出的控制”。	

SCBLK	块传送模式的选择
0	以通常模式运行
1	以块传送模式运行
如果将此 SCBLK 位设定为“1”就以块传送模式运行。 有关块传送模式的详细内容，请参照“16.4.2 数据格式”。	

SCOE	奇偶校验的选择
0	以偶校验发送 / 接收
1	以奇校验发送 / 接收
有关使用方法的详细内容，请参照“16.4.2 数据格式”。	

SCBCP2	SCBCP1	SCBCP0	基本时钟脉冲 1 和基本时钟脉冲 0
0	0	0	93 个时钟周期 (S = 93)
0	0	1	128 个时钟周期 (S = 128)
0	1	0	186 个时钟周期 (S = 186)
0	1	1	512 个时钟周期 (S = 512)
1	0	0	32 个时钟周期 (S = 32)
1	0	1	64 个时钟周期 (S = 64)
1	1	0	372 个时钟周期 (S = 372) (初始值)
1	1	1	256 个时钟周期 (S = 256)
详细内容请参照“16.4.4 接收数据采样时序”。 S 表示“16.3.5 智能卡位速率寄存器 n (SCBRRn)”中 S 的值。			

注 只在 SCTE 位和 SCRE 位都为“0”时才能进行写操作。

备注 n: 通道号 (n = 0、1)

图 16-7 智能卡串行模式寄存器 n（SCMRn）的格式（2/2）

地址：F0322H（SCMR0）、F032AH（SCMR1） 复位后：18H R/W

符号	7	6	5	4	3	2	1	0
SCMRn 注	SCGSM	SCBLK	SCOE	SCBCP2	SCBCP1	SCBCP0	SCCKS1	SCCKS0

SCCKS1	SCCKS0	时钟的选择
0	0	f _{CLK} （n = 0）
0	1	f _{CLK} /2（n = 1）
1	0	f _{CLK} /4（n = 2）
1	1	f _{CLK} /8（n = 3）
选择内置波特率发生器的时钟源。 有关 SCCKS1、SCCKS0 位的设定值和波特率的关系，请参照“16.3.5 智能卡位速率寄存器 n（SCBRRn）”。 n 为设定值的十进制，表示“16.3.5 智能卡位速率寄存器 n（SCBRRn）”中 n 的值。		

注 只在 SCTE 位和 SCRE 位都为“0”才能进行写操作。

备注 n：通道号（n = 0、1）

16.3.3 智能卡串行控制寄存器 n（SCCRn）

SCCRn 寄存器控制发送 / 接收和中断，以及选择发送 / 接收的时钟源。有关各中断请求，请参照“16.4.11 中断源”。

通过 8 位存储器操作指令设定 SCCRn 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 16-8 智能卡串行控制寄存器 n（SCCRn）的格式（1/2）

地址：F0323H（SCCR0）、F032BH（SCCR1） 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
SCCRn	0	0	SCTE	SCRE	SCDIR	SCINV	SCCKE1	SCCKE0

SCTE	选择允许 / 禁止发送
0	禁止串行发送
1	允许串行发送
如果给 SCTE 位设定“1”，就在通过前导功能输出 1 帧长度的高电平后，允许发送运行。 在此状态下，将发送数据写入 SCTDRn，并且在将 SCSRn 的 SCTDRE 标志清“0”时开始串行发送。另外，在将 SCTE 位设为“1”前，必须先设定 SCMRn，选择发送格式。 如果通过将 SCTE 位置“0”来禁止发送运行，就必须将 SCSRn 的 SCTDRE 标志固定为“1”。	

SCRE	选择允许 / 禁止接收
0	禁止串行接收
1	允许串行接收
如果给 SCRE 位设定“1”，就允许接收运行。在此状态下一旦检测到起始位，就开始串行接收。另外，在将 SCRE 位设为“1”前，必须先设定 SCMRn，选择接收格式。 即使通过将 SCRE 位置“0”来禁止接收运行，SCRDRF、SCPER 和 SCORER 标志也不受影响，保持之前的状态。	

SCDIR 注	数据传送顺序的选择
0	以 LSB 优先方式发送 / 接收
1	以 MSB 优先方式发送 / 接收
选择串行 / 并行转换的方向。	

注 只在 SCTE 位和 SCRE 位都为“0”时才能进行写操作。该写操作可与将“0”写入 SCTE 位和 SCRE 位同时进行。

备注 n：通道号（n = 0、1）

图 16-8 智能卡串行控制寄存器 n（SCCRn）的格式（2/2）

地址：F0323H（SCCR0）、F032BH（SCCR1） 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
SCCRn	0	0	SCTE	SCRE	SCDIR	SCINV	SCCKE1	SCCKE0

SCINV 注	选择数据取反 / 非取反
0	发送 SCTDRn 的内容，并且将接收数据保存到 SCRDRn。
1	对 SCTDRn 的内容进行取反后发送，并且对接收数据进行取反后保存到 SCRDRn。
如果将 SCINV 位设定为“1”，就在取反发送 / 接收数据后进行保存。 SCINV 不影响奇偶校验位。 如果要取反奇偶校验位，必须通过 SCMRn 的 SCOE 位进行设定。	

SCCKE1 注	SCCKE0 注	时钟输出的控制
• 当 SCMRn 寄存器的 SCGSM = 0 时		
0	0	时钟停止（初始值为“H”）
0	1	时钟输出
1	0	禁止设定
1	1	禁止设定
• 当 SCMRn 寄存器的 SCGSM = 1 时		
0	0	固定为低电平输出
0	1	时钟输出
1	0	固定为高电平输出
1	1	时钟输出
控制从 SCCLKn 引脚的时钟输出。在 GSM 模式中，可动态切换时钟输出。详细内容请参照“16.4.8 时钟输出的控制”。		

注 只在 SCTE 位和 SCRE 位都为“0”时才能进行写操作。该写操作可与将“0”写入 SCTE 位和 SCRE 位同时进行。

备注 n：通道号（n = 0、1）

16.3.4 智能卡串行状态寄存器 n（SCSRn）

SCSRn 寄存器由 SMCI 的状态标志构成。只能清除 SCTDRE、SCRDRF、SCORER、SCERS 和 SCPER 标志。

通过 8 位存储器操作指令设定 SCSRn 寄存器。
在产生复位信号后，此寄存器的值变为“84H”。

图 16-9 智能卡串行状态寄存器 n（SCSRn）的格式（1/3）

地址：F0324H（SCSR0）、F032CH（SCSR1） 复位后：84H R/W

符号	7	6	5	4	3	2	1	0
SCSRn	SCTDRE	SCRDRF	SCORER	SCERS	SCPER	SCTEND	0	0

SCTDRE 注	发送数据寄存器空标志
0	SCTDRn 中有发送数据。
1	SCTDRn 中无发送数据。

表示 SCTDRn 中有无发送数据。

[置位条件]

- SCCRn 的 SCTE 位为“0”时
- 数据从 SCTDRn 传送到发送移位寄存器 n 并且允许将数据写入 SCTDRn 时

[清除条件]

- 在读取“SCTDRE =1”后向 SCTDRE 写“0”时
- 将发送数据传送到 SCTDRn 时

SCRDRF 注	接收数据寄存器满标志
0	SCRDRn 中无接收数据。
1	SCRDRn 中有接收数据。

表示 SCRDRn 中有无接收数据。

[置位条件]

- 在接收正常结束后接收数据从接收移位寄存器 n 传送到 SCRDRn 时

[清除条件]

- 在读取“SCRDRF=1”后向 SCRDRF 写“0”时
- 从 SCRDRn 传送数据时

即使清除 SCCRn 的 SCRE 位，SCRDRF 标志也不受影响，保持之前的状态。

如果在 SCRDRF 标志为“1”的状态下完成下一个数据收的接收，就发生溢出错误并且丢失接收到的数据，因此必须注意。

注 只能写用于清除标志的“0”。

备注 n: 通道号 (n = 0、1)

图 16-9 智能卡串行状态寄存器 n (SCSRn) 的格式 (2/3)

地址: F0324H (SCSR0)、F032CH (SCSR1) 复位后: 84H R/W

符号	7	6	5	4	3	2	1	0
SCSRn	SCTDRE	SCRDRF	SCORER	SCERS	SCPER	SCEND	0	0

SCORER 注	溢出错误标志
0	未发生溢出错误
1	发生溢出错误

表示在接收时发生溢出错误，并且异常结束。

[置位条件]

- 在 SCRDRF=1 的状态下接受到下一个数据时

[清除条件]

- 在读取“SCORER=1”后向 SCORER 写“0”时

SCRDRn 保持发生溢出错误前的接收数据，丢失发生溢出错误后接收到的数据。在 SCORER 标志被设定为“1”的状态下，不能继续进行之后的串行接收。

即使清除 SCCRn 的 SCRE 位，SCORER 标志也不受影响，保持之前的状态。

SCERS 注	错误信号状态标志
0	无 Low 错误信号响应
1	有 Low 错误信号响应

[置位条件]

- 对 Low 错误信号进行采样时

[清除条件]

- 在读取“SCERS=1”后向 SCERS 写“0”时

即使清除 SCCRn 的 SCTE 位，SCERS 标志也不受影响，保持之前的状态。

SCPER 注	奇偶校验错误标志
0	未发生奇偶校验错误
1	发生奇偶校验错误

表示在接收时发生奇偶校验错误，并且异常结束。

[置位条件]

- 在接收时检测到奇偶校验错误时

[清除条件]

- 在读取“SCPER=1”后向 SCPER 写“0”时

在发生奇偶校验错误时，虽然接收到的数据被传送到 SCRDRn，但是 SCRDRF 标志不被置位。另外，在 SCPER 标志被设为“1”的状态下，之后接收到的数据不被传送到 SCRDRn。

即使清除 SCCRn 的 SCRE 位，SCPER 标志也不受影响，保持之前的状态。

注 只能写用于清除标志的“0”。

备注 n: 通道号 (n = 0、1)

图 16-9 智能卡串行状态寄存器 n（SCSRn）的格式（3/3）

地址：F0324H（SCSR0）、F032CH（SCSR1） 复位后：84H R/W

符号	7	6	5	4	3	2	1	0
SCSRn	SCTDRE	SCRDRF	SCORER	SCERS	SCPER	SCTEND	0	0

SCTEND 注	发送结束标志
0	正在发送数据。
1	发送数据结束。
如果没有接收方的错误信号响应，就在允许将下一个发送数据传送到 SCTDRn 时设定此寄存器。 [置位条件] • 当 SCCRn 的 SCTE = 0 时 • 在进行一定期间的 1 字节数据发送后 SCERS = 0 并且 SCTDRE = 1 时 设定时序根据寄存器的设定而不同，如下所示。 SCGSM、SCBLK： 0、0 时：开始发送后的 12.5etu 之后 0、1 时：开始发送后的 11.5etu 之后 1、0 时：开始发送后的 11.0etu 之后 1、1 时：开始发送后的 11.0etu 之后 [清除条件] • 在读取“SCTDRE= 1”后向 SCTDRE 标志写“0”时 • 将发送数据传送到 SCTDRn 时	

注 只能读。

备注 n：通道号（n = 0、1）

16.3.5 智能卡位速率寄存器 n (SCBRRn)

SCBRRn 寄存器用于调整位速率。在 SCCKE1 位和 SCCKE0 位都为“0”（停止时钟输出时）时，设定此寄存器。

只在 SCTE 位和 SCRE 位都为“0”时才能写 SCBRRn。

在产生复位信号后，此寄存器的值变为“0FH”。

图 16-10 智能卡位速率寄存器 n (SCBRRn) 的格式

地址: F0325H (SCBRR0)、F032DH (SCBRR1) 复位后: 0FH R/W

符号	7	6	5	4	3	2	1	0
SCBRRn	0	0	0	0	SCBRRn [3:0]			

SCBRRn [3:0]	功能
bit3 ~ 0	位速率寄存器的设定值 N (N = 0 ~ 15)

因为 SMCI 在每个通道都有独立的波特率发生器，所以可以设定不同的位速率。

SCBRRn 的设定值 N 与位速率 B 的关系如下。

位速率:

$$B = \frac{f_{CLK} \times 10^6}{S \times 2^n \times 2 (N + 1)}$$

SCBRRn 的初始值为“0FH”。输出时钟的频率为 $f = B \times S$ 。

备注 1. B: 位速率 (bit/s)

N: 波特率发生器的 SCBRRn 的设定值 ($0 \leq N \leq 15$)

f_{CLK} : 工作频率 (MHz)

n 和 S: 取决于下表所示的 SCMRn 的设定值。

SCMRn 的设定值		n	SCMRn 的设定值			S
SCCKS1	SCCKS0		SCBCP2	SCBCP1	SCBCP0	
0	0	0	0	0	0	93
0	1	1	0	0	1	128
1	0	2	0	1	0	186
1	1	3	0	1	1	512
			1	0	0	32
			1	0	1	64
			1	1	0	372
			1	1	1	256

2. n: 通道号 (n = 0、1)

16.3.6 端口输入模式寄存器 5 (PIM5)

PIM5 寄存器以 1 位为单位设定输入缓冲器。

通过 1 位或者 8 位存储器操作指令设定 PIM5 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 16-11 端口输入模式寄存器 5 (PIM5) 的格式

地址: F0045H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
PIM5	0	0	PIM55	0	PIM53	0	0	0

PIM5n	P5n 引脚的输入缓冲器的选择 (n = 3、5)
0	通常输入缓冲器
1	TTL 输入缓冲器

16.3.7 端口输出模式寄存器 5 (POM5)

POM5 寄存器以 1 位为单位设定输出模式。

通过设定 POM5 寄存器和 PU5 寄存器设定是否使用内部上拉电阻。

通过 1 位或者 8 位存储器操作指令设定 POM5 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 16-12 端口输出模式寄存器 5 (POM5) 的格式

地址: F0055H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
POM5	0	POM56	POM55	POM54	POM53	0	0	0

POM5n	选择 P5n 引脚的输出模式 (n = 3 ~ 6)
0	通常输出模式 输入时，将 PU5n 位设定为有效。
1	N 沟道漏极开路输出 (V _{DD} 耐压) 模式 输入时，将 PU5n 位设定为无效。

16.3.8 端口模式寄存器 5 (PM5)

PM5 寄存器以 1 位为单位指定端口的输入或者输出。

通过 1 位或者 8 位存储器操作指令设定 PM5 寄存器。

在产生复位信号后，此寄存器的值变为“FFH”。

当端口引脚用作复用功能的引脚时，参照“4.5.3 使用的端口功能和复用功能的寄存器设定例子”设定此寄存器。

图 16-13 端口模式寄存器 5 (PM5) 的格式

地址: FFF25H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PM5	PM57	PM56	PM55	PM54	PM53	PM52	PM51	PM50

PM5n	选择 P5n 引脚的输入 / 输出模式 (n = 0 ~ 7)
0	输出模式 (输出缓冲器 ON)
1	输入模式 (输出缓冲器 OFF)

16.3.9 端口寄存器 5 (P5)

P5 寄存器设定端口的输出锁存器的值。

如果读 P5 寄存器，输入模式时的读取值就为引脚电平，输出模式时的读取值就为端口输出锁存器的值。

通过 1 位或者 8 位存储器操作指令设定 P5 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 16-14 端口寄存器 5 (P5) 的格式

地址: FFF05H 复位后: 00H (输出锁存器) R/W

符号	7	6	5	4	3	2	1	0
P5	P57	P56	P55	P54	P53	P52	P51	P50

P5n	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

备注 n = 0 ~ 7

16.4 智能卡接口 (SMCI) 的运行

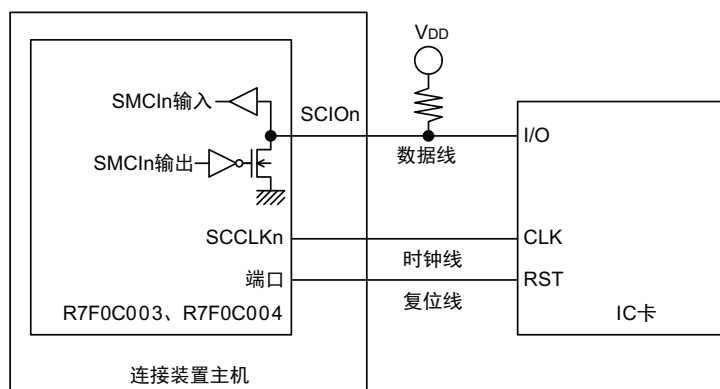
SMCI 支持与符合 ISO/IEC 7816-3 (Identification Card) 的智能卡 (IC 卡) 的连接。

16.4.1 与 IC 卡的连接例子

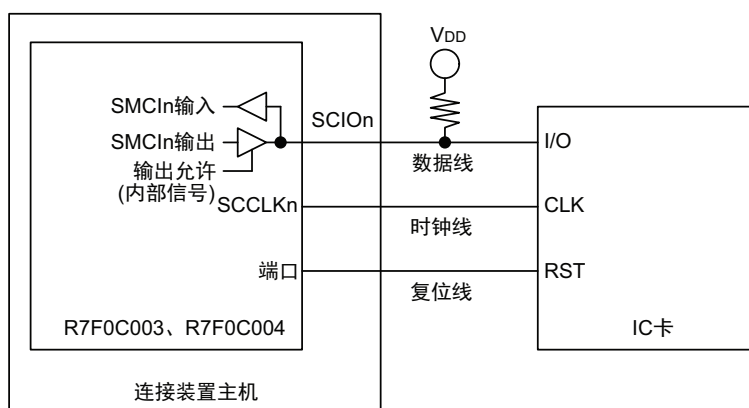
智能卡接口的引脚连接概要如图 16-15 所示。通过 1 条数据传送线与 IC 卡进行发送 / 接收。连接 SCIO_n 引脚和 IC 卡引脚。此时如果用作漏极开路, 就必须通过电阻将数据传送线上拉至电源 V_{CC}。如果在不连接 IC 卡的状态下将 SCRE 位和 SCTE 位设为“1”, 就能进行自诊断。要向 IC 卡提供 SMCI 生成的时钟时, 必须将 SCCLK_n 引脚的输出输入到 IC 卡的 CLK 引脚。可将输出端口用于复位信号的输出。

图 16-15 智能卡接口的引脚连接概要

(1) 用作漏极开路输出时



(2) 用作 CMOS 输出时



备注 n: 通道号 (n = 0、1)

因为在保护时间中 SMCI 输出和 IC 卡输出都有可能变为 Hi-Z 状态, 因此, 在用作 CMOS 输出时, 将数据线连接上拉电阻。

16.4.2 数据格式

除通常的运行外，智能卡接口还具有块传送模式。

通常运行与块传送模式的不同点如表 16-2 所示。因为块传送模式不执行错误信号的发送 / 接收，所以串行数据的发送 / 接收与 UART 相同。

表 16-2 运行模式的不同点

项目			智能卡接口的通常运行	智能卡接口的块传送模式
错误信号的发送 / 接收和重新发送			有	无
发送	保护时间		至少 3etu	至少 2etu
	SCTEND 标志	SCGSM = 0	开始发送后的 12.5etu 之后	开始发送后的 11.5etu 之后
	置位时序	SCGSM = 1	开始发送后的 11.0etu 之后	开始发送后的 11.0etu 之后
	SCERS		变化	保持
接收	SCPER		变化	变化

数据格式如图 16-16 所示。

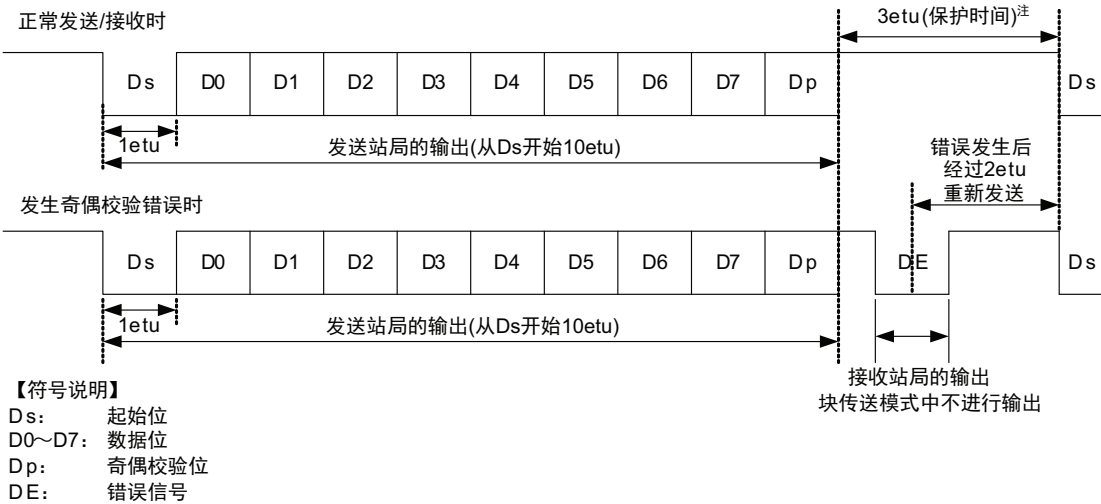
- 异步时，1 帧由 8 位数据和奇偶校验位构成。
- 发送时，从奇偶校验位的结束到下 1 帧的开始，至少需要 3etu（Elementary Time Unit：1 位的传送期间）的保护时间。
- 如果在通常运行的接收时检测到奇偶校验错误，就从起始位开始经过 10.5etu 后，输出 1etu 期间的 Low 错误信号。
- 如果在通常运行的发送时对错误信号进行采样，就在至少经过 2etu 后，自动重新发送相同的数据。

块传送模式与通常的智能卡接口运行比较有以下不同点。

- 虽然在接收时进行奇偶校验，但是即使检测到错误也不输出错误信号。因为 SCSRn 的 SCPER 标志置位，所以必须在接收下 1 帧的奇偶校验位前清除该标志。
- 从发送时的奇偶校验位结束到下 1 帧的开始，至少需要 2etu 的保护时间。
- 因为在发送时不执行重新发送，所以 SCTEND 标志在开始发送后的 11.5etu 之后被置位。
- 因为不进行错误信号的发送 / 接收，所以 SCERS 标志不发生变化。

备注 n：通道号（n = 0、1）

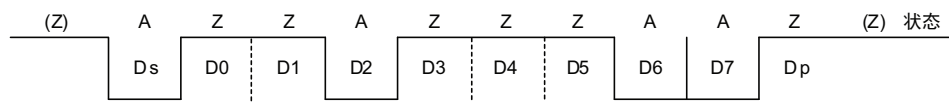
图 16-16 数据格式



注 块传送模式中 2etu。

如下进行与正向约定型和反向约定型的 IC 卡之间的发送 / 接收。

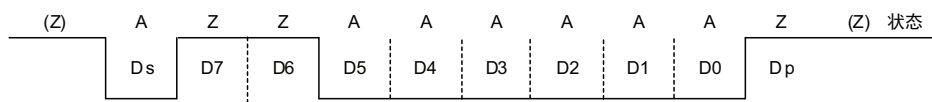
图 16-17 正向约定（SCDIR = SCINV = SCOE = 0）



如上述起始字符的例子，正向约定型是以逻辑 1 电平对应状态 Z，逻辑 0 电平对应状态 A，并且以 LSB 优先进行发送 / 接收。上述起始字符为数据“3BH”。

在正向约定型时，必须将 SCCRn 的 SCDIR 位和 SCINV 位设定为“0”。另外，还必须将 SCMRn 的 SCOE 位设定为“0”，以满足是偶校验的智能卡规则。

图 16-18 反向约定（SCDIR = SCINV = SCOE = 1）



反向约定型是以逻辑 1 电平对应状态 A，逻辑 0 电平对应状态 Z，并且以 MSB 优先进行发送 / 接收。上述起始字符为数据“3FH”。在反向约定型时，必须将 SCCRn 的 SCDIR 位和 SCINV 设定为“1”。根据智能卡的规则奇偶校验位必须是偶校验并且为逻辑 0 以状态 Z 对应。R7F0C003、R7F0C004 中的 SCINV 只取反数据位 D7 ~ D0，因此在接收和发送时都必须将 SCMRn 的 SCOE 位设定为“1”，以取反奇偶校验位。

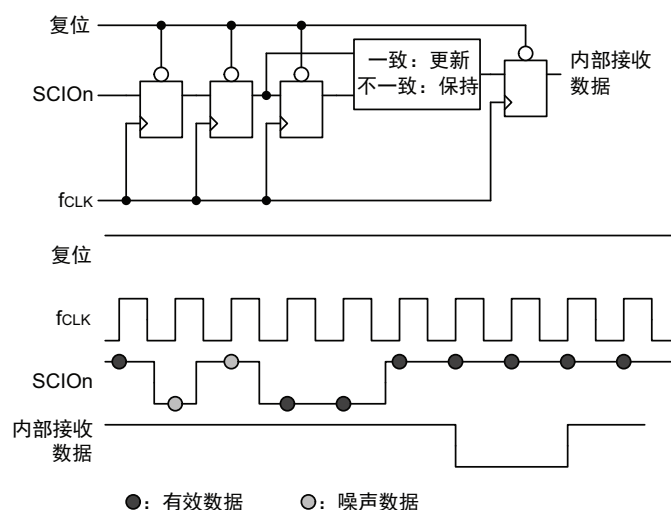
备注 n: 通道号 (n = 0、1)

16.4.3 接收数据的噪声滤波器

对接收数据 (SCIO_n) 进行采样, 如果 2 次一致就认为电平确定。

因电路结构, 接收数据最多延迟 3 个运行时钟。噪声滤波器电路与时序图如图 16-19 所示。可对通过 f_{CLK} 采样的数据执行小于 2 个 f_{CLK} 的噪声消除。

图 16-19 噪声滤波器电路和时序图



备注 n: 通道号 (n = 0、1)

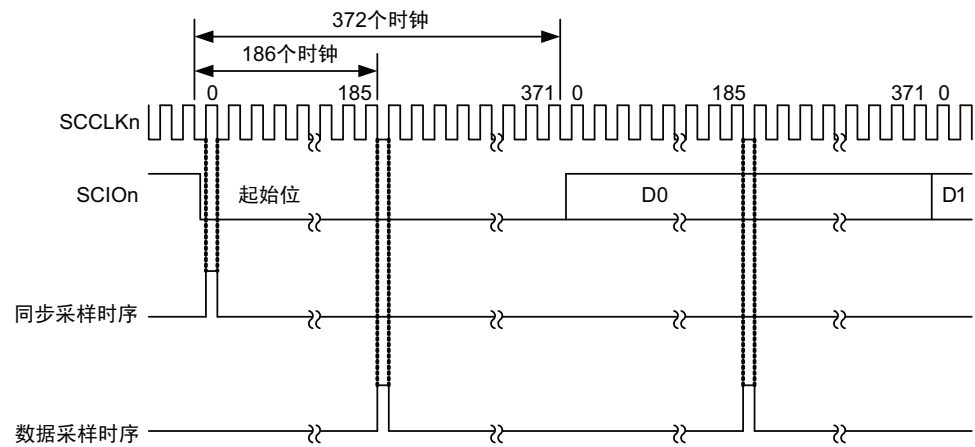
16.4.4 接收数据采样时序

发送 / 接收时钟是由内部波特率发生器生成的内部时钟。SMCI 以 SCMRn 寄存器的 SCBCP2、SCBCP1、SCBCP0 位设定的频率是位速率的 32、64、372、256、93、128、186 或者 512 倍的基本时钟运行。接收时，通过将起始位的开始（低电平）作为基本时钟进行采样，取得内部同步。

如图 16-20 所示，通过分别在基准时钟的第 16、32、186、128、46、64、93 和 256 个上升沿对接收数据进行采样，可以在各位的中间取得数据。

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

图 16-20 智能卡接口的接收数据采样时序（使用 372 倍的时钟时）



备注 n: 通道号 (n = 0、1)

16.4.5 SMCI 的初始化

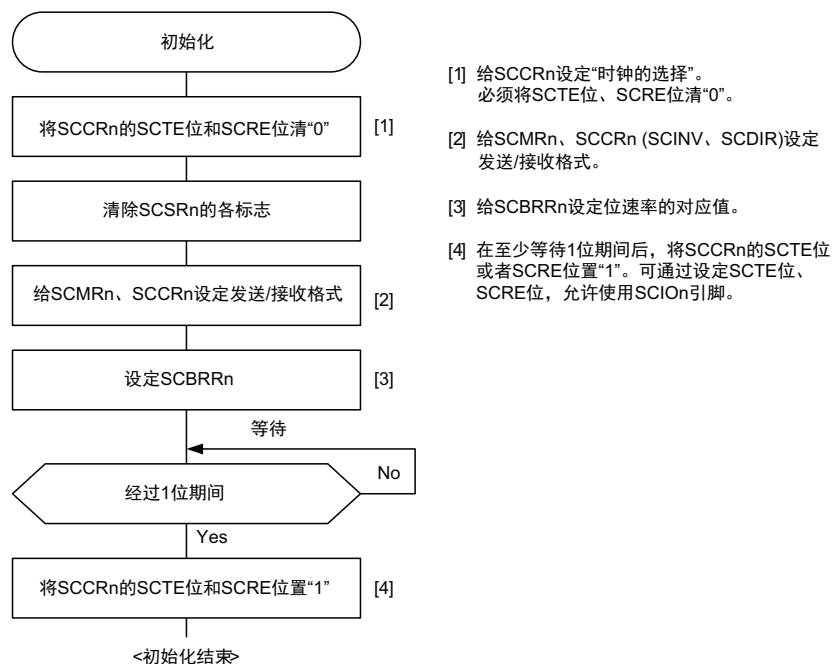
在发送和接收数据前，通过以下步骤初始化 SMCI。在从发送模式切换至接收模式或者从接收模式切换至发送模式时也必须进行初始化（图 16-21）。

1. 作为端口的寄存器设定，必须将 SCION 引脚的端口寄存器（P5m）设定为“1”，将端口模式寄存器（PM5m）也设定为“1”（输入）。如果要发送数据作为 CMOS 输出时，必须将端口输出模式寄存器（POM5m）设定为“0”，作为漏极开路输出时，必须将端口输出模式寄存器（POM5m）设定为“1”。另外，必须将 SCCLKn 引脚的端口寄存器（P5p）设定为“1”，将端口模式寄存器（PM5p）设定为“0”（输出）。
2. 将 SCCRN 的 SCTE 位和 SCRE 位清“0”。
3. 将 SCSRn 的错误标志（SCERS、SCPER、SCORER）都清“0”。
4. 设定 SCMRn 的 SCGSM、SCBLK、SCOE、SCBCP2、SCBCP1、SCBCP0、SCCKS1 和 SCCKS0 位。
5. 给 SCBRRn 设定与位速率对应的值。
6. 设定 SCCRN 的 SCDIR、SCINV、SCCKE1 和 SCCKE0 位。此时，必须将 SCTE 位和 SCRE 位设定为“0”。在将 SCCKE0 位设定为“1”时，从 SCCLKn 引脚输出时钟。
7. 在等待至少 1 位期间后，设定 SCCRN 的 SCTE 位和 SCRE 位。除自诊断外，不能同时设定 SCTE 和 SCRE 位。

从接收模式切换至传送模式时，必须在确认接收运行已经结束后开始初始化，并且将 SCRE 位设定为“0”，将 SCTE 位设定为“1”。可通过 SCRDRF、SCPER、或者 SCORER 标志，确认接收运行是否结束。从发送模式切换至接受模式时，必须在确认发送运行已经结束后开始初始化，并且将 SCTE 位设定为“0”，将 SCRE 设定为“1”。可通过 SCTEND 标志，确认发送运行是否结束。

备注 n: 通道号 (n = 0、1)、m = 3、5, p = 4、6

图 16-21 SMCI 初始化的采样流程图



注意 复位后 SCTE = SCRE = 0。

备注 n: 通道号 (n=0、1)

16.4.6 串行数据的发送 (通常模式)

发送时的重新传送运行如图 16-22 所示。图中的数字与以下说明的序号对应。

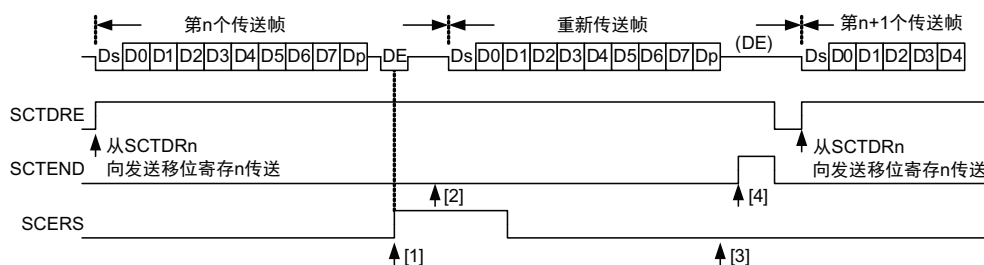
1. 在结束 1 帧的发送后, 如果对接收方的错误信号进行采样, SCSRn 的 SCERS 标志就被设定为“1”。此时, 产生 INTSCEn 中断请求。必须在对下一个奇偶校验位进行采样前, 清除 SCERS 标志。
2. 对于接收到错误信号的帧, SCSRn 的 SCTEND 标志不被置位。自动重新发送发送移位寄存器 n 的内容。
3. 如果错误信号未从接收方返回, SCSRn 的 SCERS 标志就不被置位。
4. 在认为已经结束 1 帧的发送 (包括重新传送) 后, SCSRn 的 SCTEND 标志被置位。此时, 产生 INTSCTn 中断请求。通过将发送数据写入 SCTDRn, 开始发送下一个数据。

发送流程图的例子如图 16-24 所示。通过 INTSCTn 中断源启动 DMA, 可自动执行此一系列的处理。在发送运行时, 如果将 SCSRn 的 SCTEND 标志设定为“1”, 就产生 INTSCTn 中断请求。如果事先将 INTSCTn 中断请求设定为 DMA 的启动源, 就在通过 INTSCTn 中断请求启动 DMA 后, 进行发送数据的传送。SCTDRE 标志和 SCTEND 标志在数据传送时自动清“0”。如果发生错误, SMCI 就自动重新发送相同的数据。在此期间 SCTEND 标志保持“0”, DMA 未被启动。因此, SMCI 和 DMA 自动发送指定的字节数, 包括发生错误时的重新发送。但是, 因为在发生错位时不自动清除 SCERS 标志, 所以必须在发生错位时产生 INTSCEn 中断请求, 清除 SCERS 标志。

使用 DMA 进行发送 / 接收时, 必须先设定 DMA, 并且设定为允许状态后, 再设定 SMCI。

注意 在将 SCCRn 寄存器的 SCTE 位从“0”设定为“1”时, 必须通过前导功能输出一次 1 帧长度的高电平。如果在通信过程 (接收) 中发生异常结束, 可通过前导功能避免开始发送时的智能卡的发送与数据冲突。

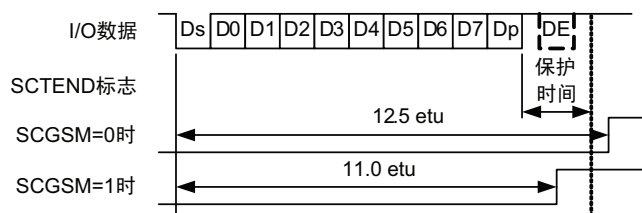
图 16-22 SMCI 发送模式中的重新传送运行
(重新传送不包括从 SCTDRn 向发送移位寄存器 n 的传送)



备注 n: 通道号 (n = 0、1)

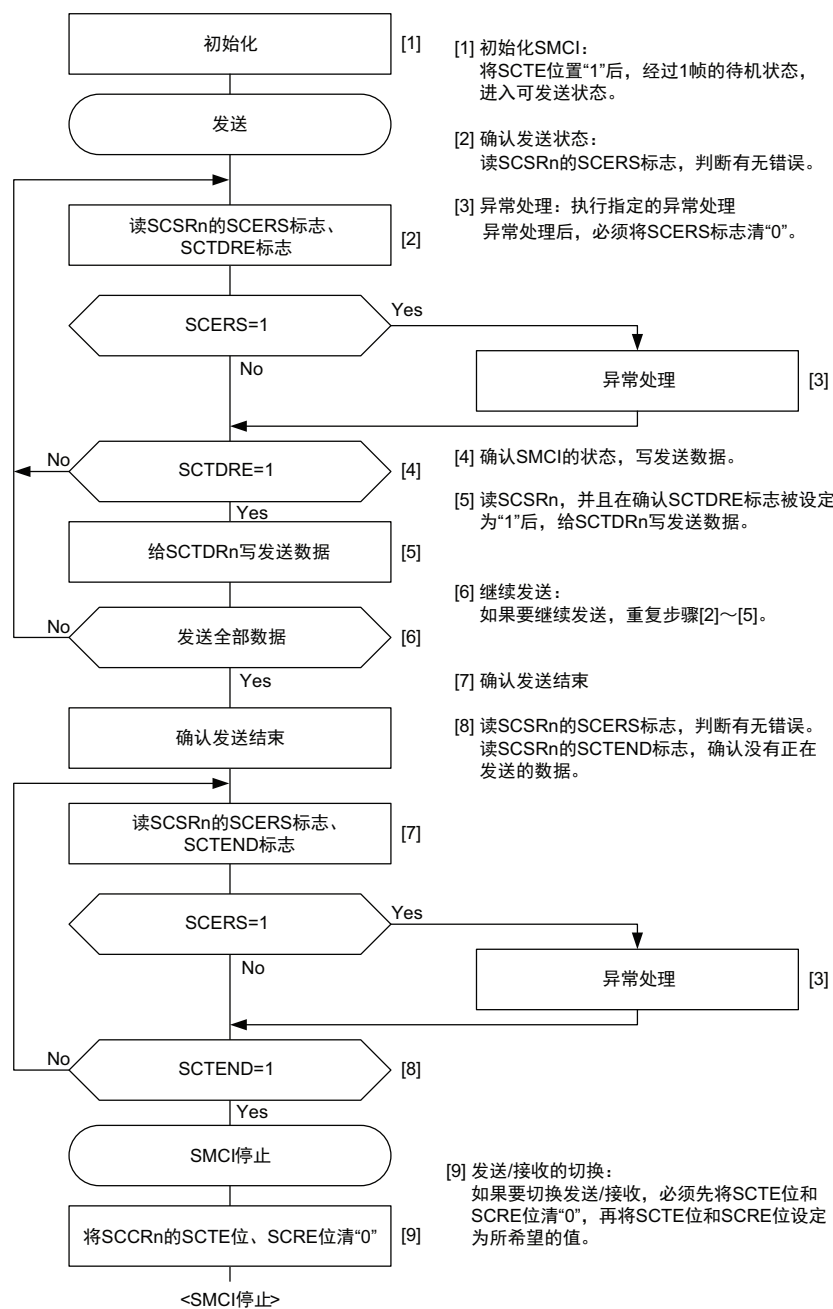
SCTEND 标志的置位时序根据 SCMRn 的 GSM 位的设定而不同。发送运行时的 SCTEND 标志产生时序如图 16-23 所示。

图 16-23 发送运行时的 SCTEND 标志产生时序



备注 n: 通道号 (n = 0、1)

图 16-24 发送流程图的例子



注意 在通过发送数据空中断 (INTSCTn) 请求启动 DMA, 并且将发送数据写入 SCTDRn 时, 自动进行 SCTDRE 标志的确认和清除。

备注 n: 通道号 (n = 0、1)

16.4.7 串行数据的接收 (通常模式)

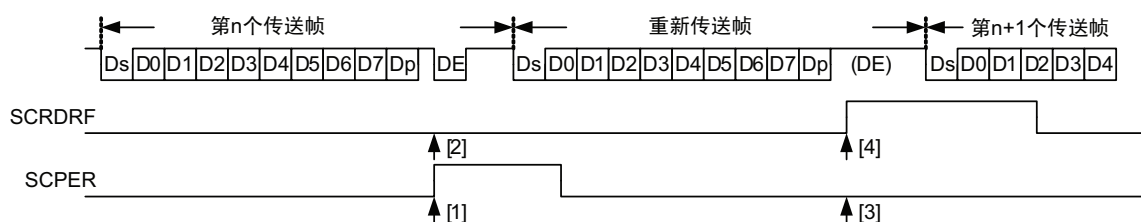
SMCI 接收模式时的重新传送运行如图 16-25 所示。图中的数字与以下说明的序号对应。

1. 如果在接收数据中检测到奇偶校验错误, SCSRn 的 SCPER 标志就被设定为“1”。此时, 产生 INTSCEn 中断请求。必须在对下一个奇偶校验位进行采样前, 清除 SCPER 标志。
2. 对于检测到奇偶校验错误的帧, SCSRn 的 SCRDRF 标志不被置位。
3. 在没有检测到奇偶校验错误时, SCSRn 的 SCPER 标志不被置位。
4. 在认为已正常结束接收后, SCSRn 的 SCRDRF 标志被设定为“1”。此时, 产生 INTSCRn 中断请求。

接收流程图的例子如图 16-26 所示。通过 INTSCRn 中断源启动 DMA, 可自动执行此一系列的处理。在接收运行时, 如果将 SCRDRF 标志设定为“1”, 就产生 INTSCRn 中断请求。如果事先将 INTSCRn 中断请求设定为 DMA 的启动源, 就在通过 INTSCRn 中断请求启动 DMA 后, 进行接收数据的传送。一旦数据传送 SCRDRF 标志就自动清“0”。如果在接收时发生错误, 并且 SCORER 标志或者 SCPER 标志被设定为“1”, 就产生接收错误中断 (INTSCEn) 请求, 所以必须清除错误标志。在发生错误时, 不启动 DMA, 并且跳过接收数据, 所以只传送 DMA 指定的字节数的接收数据。另外, 即使在接收时发生奇偶校验错误, 并且 SCPER 标志被设定为“1”, 接收的数据也被传送到 SCRDRn, 因此可读此数据。

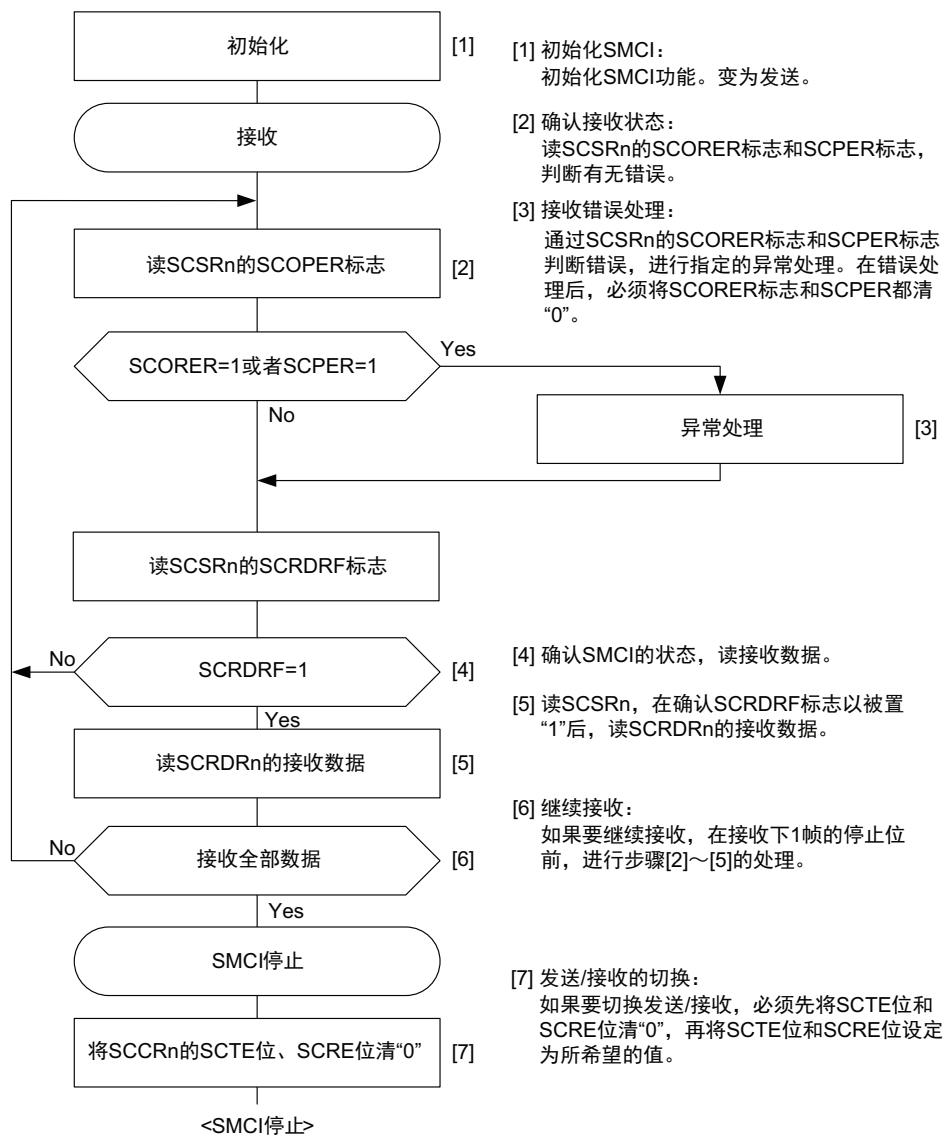
注意 块传送模式中没有错误信号 (DE), 因此不输出低电平, 也不执行重新传送。

图 16-25 SMCI 接收模式中的重新传送运行



备注 n: 通道号 (n = 0、1)

图 16-26 接收流程图的例子



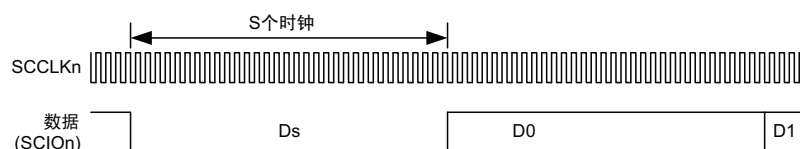
注意 在读 SCRDRn 的接收数据时，自动进行 SCRDRF 标志的确认和清除。

备注 n: 通道号 (n = 0、1)

16.4.8 时钟输出的控制

时钟输出时，如图 16-27 所示，输出的时钟的频率为位速率的 S 倍。

图 16-27 输出时钟和发送数据的相位关系 (SCBCP2 ~ SCBCP0=100B、 $S=32$ 的例子)



(1) 在通常模式 (SCMRn 寄存器的 SCGSM 位为“0”) 中使用的情况

在 SCMRn 的 SCGSM 位设定为“0”时，通过 SCCRn 的 SCCKE0 位控制时钟输出。一旦在输出时钟 (SCCKE0=1) 后又停止时钟 (SCCKE0=0) 时，保持最后的时钟输出状态。将 SCCLKn 引脚用作 SMCI 的时钟输出后，如果再用作端口功能或者其他复用功能，就必须将 PER1 寄存器的 SMCIInEN 位设定为“0”。

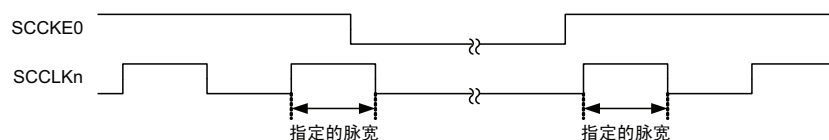
在 SCCKE0=0 (时钟停止) 时才能向 STOP 模式转移。

(2) 在 GSM 模式 (SCMRn 寄存器的 SCGSM 位为“1”) 中使用的情况

在 SCMRn 的 SCGSM 位设定为“1”时，可通过 SCCRn 的 SCCKE1 位和 SCCKE0 位固定时钟输出。但是输出时钟的最小脉宽 (高电平 / 低电平期间的最小宽度) 为 SCMRn 寄存器的 SCCKS1、SCCKS0、SCBCP2、SCBCP1 和 SCBCP0 位以及 SCBRRn 寄存器设定的 SCCLKn 的脉宽。

时钟输出的固定时序如图 16-28 所示。这是在 SCGSM = 1、SCCKE1 = 0、并且 SCCKE0 位控制时的例子。

图 16-28 时钟输出的固定时序



在上电以及向 STOP 模式转移或者从 STOP 模式返回时，为了确保时钟的占空比，必须按照以下的步骤进行处理。

[上电时的步骤]

为了确保时钟的占空比，必须按照以下的切换步骤进行处理。

1. 初始状态为端口输入并且为高阻抗状态。要固定电位时，必须将上拉电阻选择寄存器 (PU5p) 设定为“1”。
2. 将 SCMRn 的 SCGSM 位设定为“1”。
3. 通过 SCCRn 的 SCCKE1 位将 SCCLKn 引脚固定为指定的输出。
4. 设定 SCCRn 寄存器的 SCINV 位和 SCDIR 位。
5. 将 SCCRn 的 SCCKE0 位设定为“1”，开始时钟输出。

备注 n: 通道号 (n = 0、1)、p = 4、6

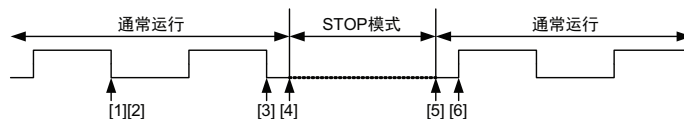
[向 STOP 模式的转移]

1. 给 SCCRn 中 SCTE 位和 SCRE 位写“0”，停止发送/接收运行。
同时，将 SCCKE1 位设定为 STOP 模式中输出固定状态的值。
2. 给 SCCRn 的 SCCKE0 位写“0”，停止时钟。
3. 等待 1 个串行时钟周期。
在此期间，保持占空比，并且以指定的电平固定时钟输出。
4. 向 STOP 模式转移。

[从 STOP 模式的返回]

5. 解除 STOP 模式。
6. 给 SCCRn 的 SCCKE0 位写“1”，输出时钟。通过正常的占空比开始产生信号。

图 16-29 时钟停止和重新启动的步骤



备注 n: 通道号 (n = 0、1)

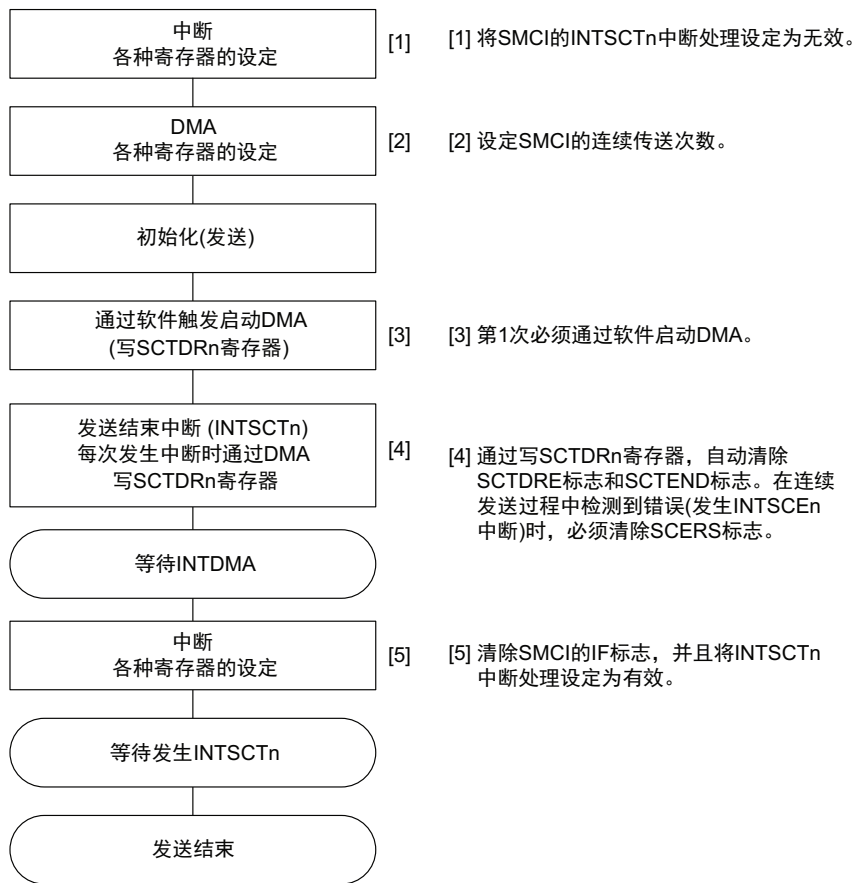
16.4.9 使用 DMA 进行的连续数据发送

通过将发送结束中断（INTSCTn）作为触发，启动 DMA 来实现连续发送的方法如下所示。

- 1. 通过中断屏蔽标志寄存器禁止处理INTSCTn中断。
- 2. 设定DMA。将传送次数设定为“实际的传送次数”。
- 3. 设定SMCI的发送。
- 4. 第一次是通过软件触发器执行DAM传送（写SCTDRn寄存器）。
- 5. 通过将INTSCTn中断作为触发，并且通过DMA将数据写入SCTDRn寄存器，实现连续发送。
- 6. 如果在连续发送期间检测到错误（INTSCEn中断），就清除SCERS标志。
- 7. 如果要结束连续发送，就在产生INTDMA后清除中断请求，并且将INTSCTn中断设定为有效。
- 8. 在发生INTSCTn中断后，结束设定次数的发送。

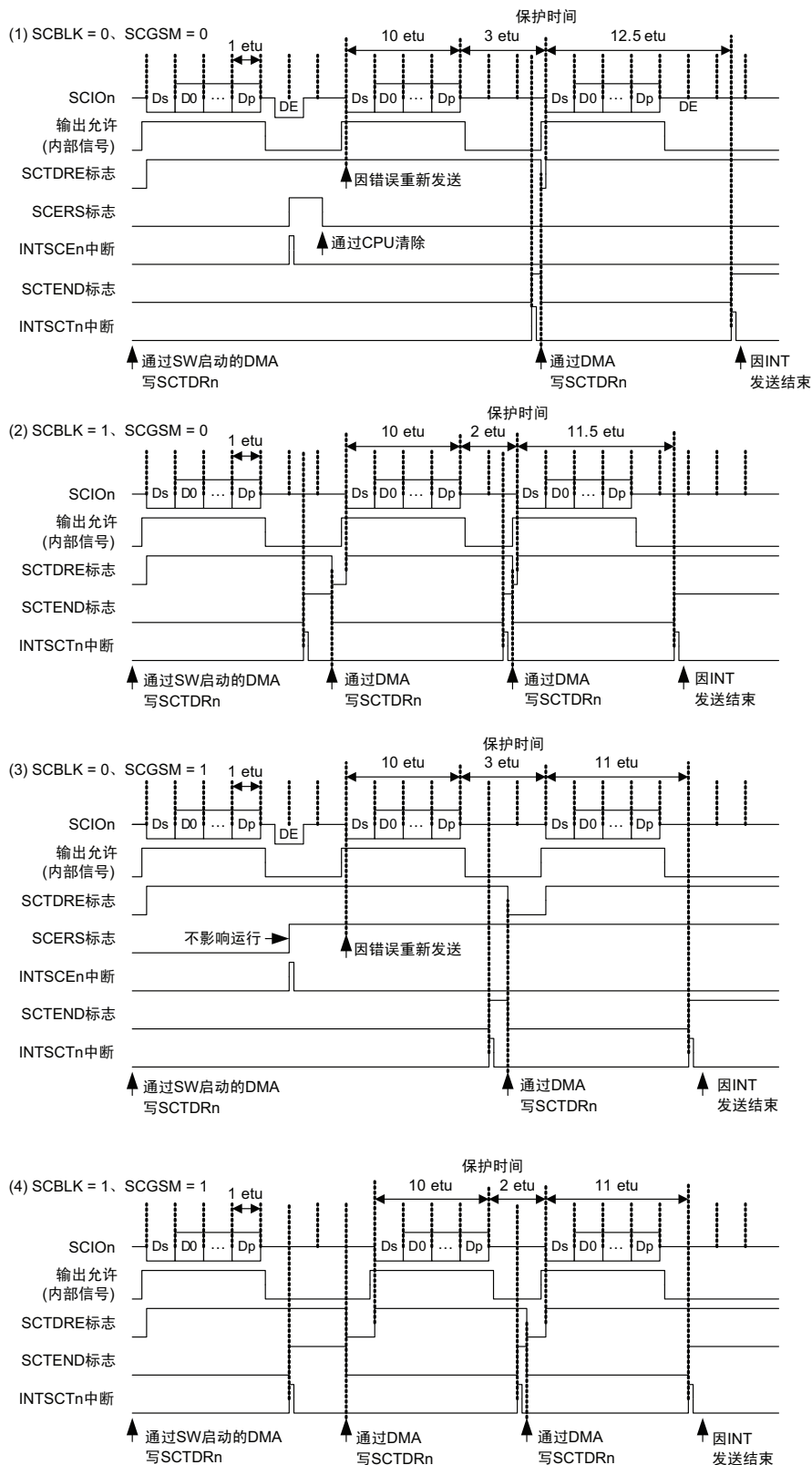
使用 DMA 进行连续数据发送的流程如图 16-30，使用 DMA 进行连续数据发送的时序图如图 16-31 所示。

图 16-30 使用 DMA 进行连续数据发送的流程



备注 n: 通道号 (n = 0、1)

图 16-31 使用 DMA 进行连续数据发送的时序图



备注 n: 通道号 (n = 0、1)

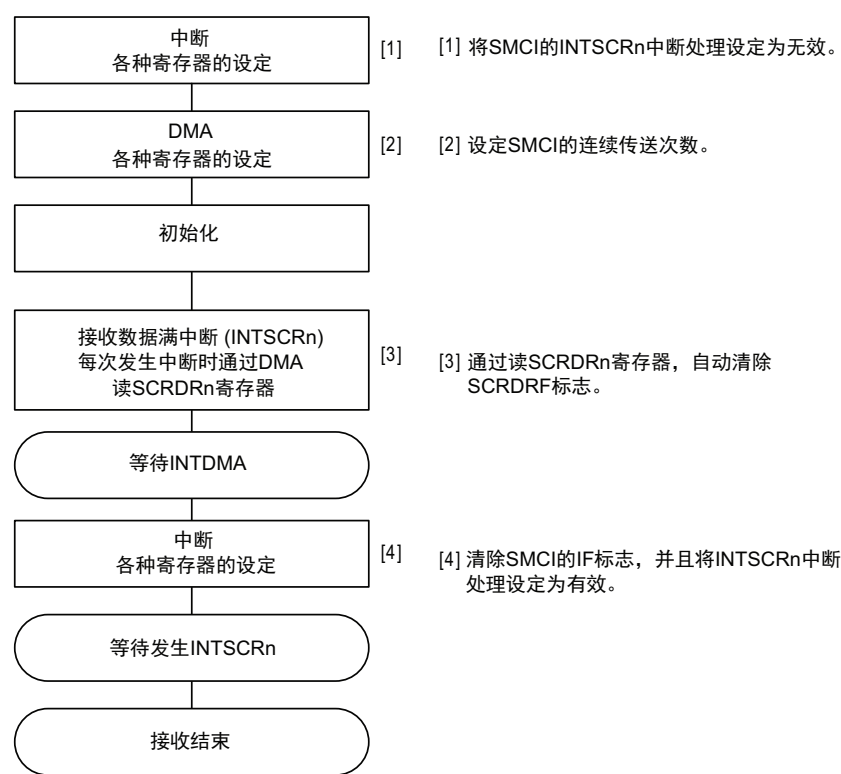
16.4.10 使用 DMA 进行的连续数据接收

通过将接收数据满中断（INTSCRn）作为触发，启动 DMA 来实现连续接收的方法如下所示，

- 1. 通过中断屏蔽标志寄存器禁止处理 INTSCRn 中断。
- 2. 设定 DMA （设定传送次数）。
- 3. 设定 SMCI 的接收。
- 4. 通过将 INTSCRn 中断作为触发，并且通过 DMA 读 SCRDRn 寄存器，实现连续接收。
- 5. 如果要结束连续接收，就在产生 INTDMA 后清除中断请求，并且将 INTSCRn 中断设定为有效。
- 6. 在发生 INTSCRn 后，结束设定次数的接收。

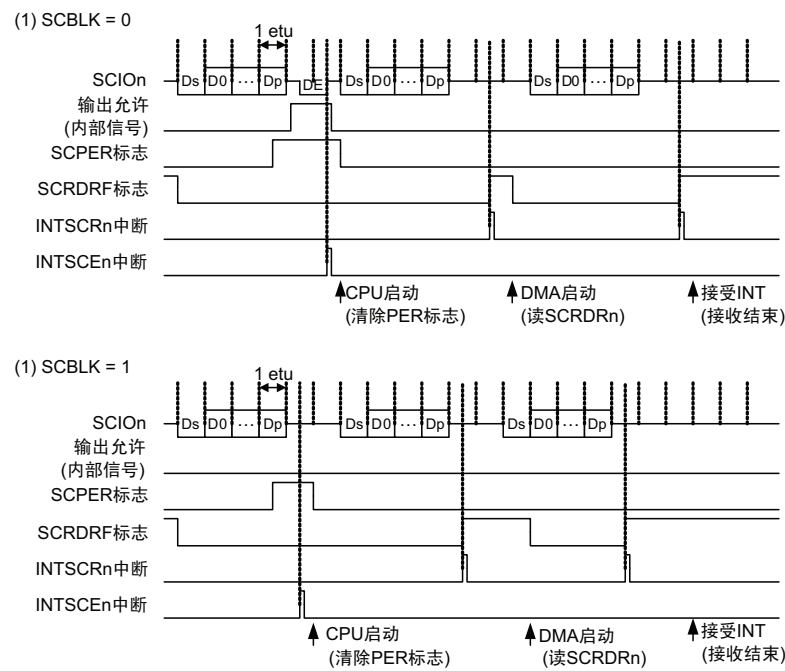
使用 DMA 进行连续数据接收的流程如图 16-32，使用 DMA 进行连续数据接收的时序图如图 16-33 所示。

图 16-32 使用 DMA 进行连续数据接收的流程



备注 n: 通道号 (n = 0、1)

图 16-33 使用 DMA 进行连续数据接收的时序图



备注 n: 通道号 (n = 0、1)

16.4.11 中断源

智能卡接口的中断源如表 16-3 所示。

表 16-3 中断源

名称	中断源	中断标志名称	DMA 启动
INTSCEn	接收错误或者检测到错误信号	SCORER、SCPER、SCERS	不可
INTSCRn	接收数据满	SCRDRF	可以
INTSCTn	发送结束	SCTEND	可以

可使用 DMA 进行发送 / 接收。在发送运行时，如果将 SCSRn 的 SCTEND 标志设定为“1”，就产生 INTSCTn 中断请求。如果事先将 INTSCTn 中断请求设定为 DMA 的启动源，就在通过 INTSCTn 中断请求启动 DMA 后，进行发送数据的传送。SCTDRE 标志和 SCTEND 标志在数据传送时自动清“0”。如果发生错误，SMCI 就自动重新发送相同的数据。在此期间 SCTEND 标志保持“0”，DMA 未被启动。因此，SMCI 和 DMA 自动发送指定的字节数，包括发生错误时的重新发送。但是，因为在发生错位时不自动清除 SCERS 标志，所以必须在发生错位时产生 INTSCEn 中断请求，清除 SCERS 标志。

在接收运行时，如果将 SCSRn 的 SCRDRF 标志设定为“1”，就产生 INTSCRn 中断请求。如果事先将 INTSCRn 中断请求设定为 DMA 的启动源，就在通过 INTSCRn 中断请求启动 DMA 后，进行接收数据的传送。一旦数据传送 SCRDRF 标志就自动清“0”。如果在接收时发生错误，SCRDR 标志不被置位而错误标志置位。因此，不启动 DMA，但是，对 CPU 产生 INTSCEn 中断请求，所以必须清除错误标志。在结束下一次接收前必须清除错误标志（SCPER、SCORER）。

使用 DMA 进行发送 / 接收时，必须先设定 DMA，并且在设定为允许状态后再设定 SMCI。

备注 n: 通道号 (n = 0、1)

16.5 使用时的注意事项

16.5.1 外围允许寄存器 1 (PER1) 的设定

通过外围允许寄存器 1 (PER1) 的 SMCInEN 位, 控制内部时钟 (f_{CLK}) 的提供 / 停止以及复位的 ON/OFF 状态。在 SMCInEN 位为“0”时, 为复位状态。如果使用 SMCI, 就必须设定为 SMCI 运行状态 (SMCInEN=1)。

为了进入复位状态, 不保持输出时钟的占空比。

16.5.2 SCTDR 的写操作与 SCTDRE 标志的关系

SCSRn 的 SCTDRE 标志是状态标志, 表示发送数据已经从 SCTDRn 传送到发送移位寄存器 n。如果 SMCI 将数据从 SCTDRn 传送到发送移位寄存器 n, SCTDRE 标志就被设定为“1”。

与 SCTDRE 标志的状态无关, 都可向 SCTDRn 写入数据。但是, 如果在 SCTDRE 标志为“0”的状态下将新数据写入 SCTDRn, 保存在 SCTDRn 的数据就会因为没有传送到发送移位寄存器 n 而丢失。因此, 必须在确认 SCTDRE 标志被设定为“1”后, 再向 SCTDRn 写发送数据。。

16.5.3 使用 DMA 时的限制事项

在通过 DMA 读 SCRDRn 时, 必须将启动源设定为对应 SMCI 的接收数据满中断 (INTSCRn)。同样, 在通过 DMA 写 SCTDRn 时, 必须将启动源设定为对应 SMCI 的发送结束中断 (INTSCTn)。

备注 n: 通道号 (n = 0、1)

16.5.4 向低功耗状态转移时的运行

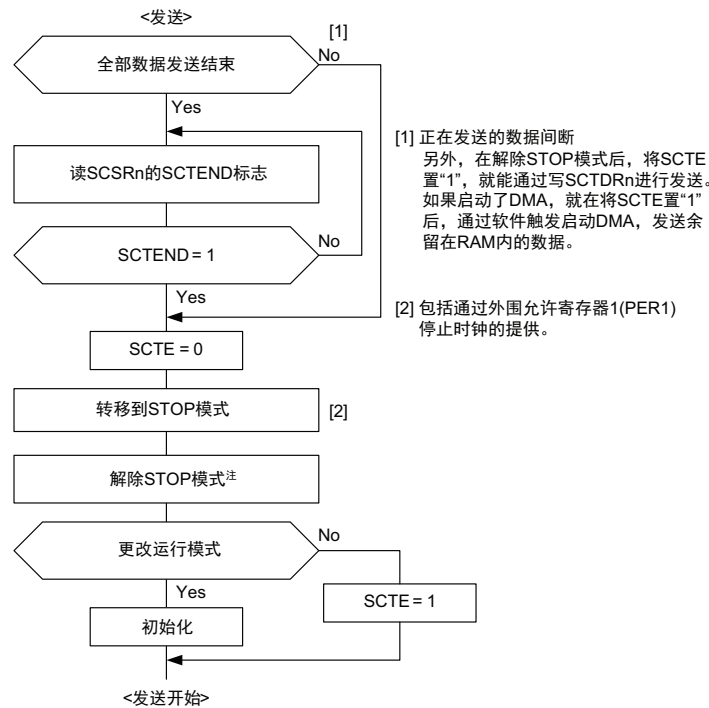
(1) 发送运行时

必须在停止运行（SCTE=0）后才能向 STOP 模式转移。通过将 SCTE 位清“0”，发送移位寄存器、SCTDRE 标志以及 SCTEND 标志置位。STOP 模式期间的输出引脚状态取决于端口的设定，解除 STOP 模式后的输出引脚为高电平输出。在不更改发送模式的情况下重新开始发送时，将 SCTE 位设定为“1”，并且通过向 SCTDRn 写发送数据，开始发送。在更改发送模式的情况下进行发送时，必须从初始设定开始执行。发送时的模式转移流程图例子如图 16-34 所示。模式转移时的端口引脚状态如图 16-35 所示。如果在发送过程中转移到 STOP 模式，发送中的数据变为不定。

必须在停止运行（SCTE=0）后才能从通过 DMA 进行的发送向 STOP 模式转移。

如果要在解除 STOP 模式后通过 DAM 进行发送，就必须将 SCTE 位设定为“1”，并且通过软件启动 DMA 后，开始发送。

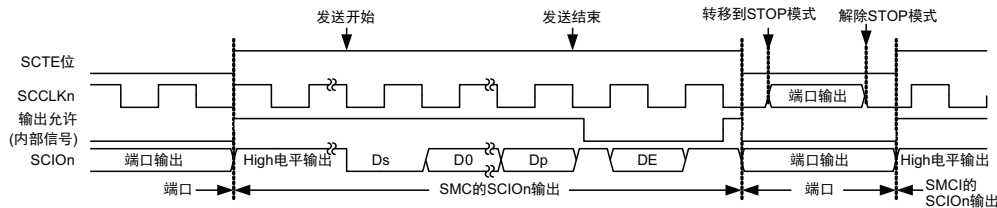
图 16-34 发送时的模式转移流程图例子



注 在将 PER1 寄存器的 SMCI_nEN 位设定为“0”后向 STOP 模式转移时，必须执行初 SMCI 的初始化处理。

备注 n: 通道号（n = 0、1）

图 16-35 模式转移时的端口引脚状态



注意 SCCLK_n 的端口输出是输出由 SCCR_n 寄存器的 SCCKE1 位和 SCCKE0 位设定的状态。

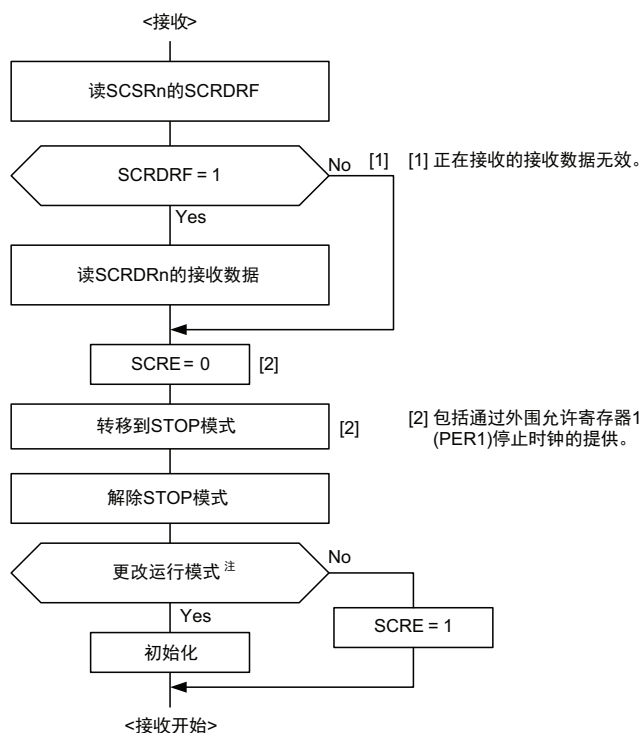
(2) 接收运行时

必须在停止接收运行 (SCRE=0) 后才能向 STOP 模式转移。如果在接收时向 STOP 模式转移, 接收中的数据就变为无效。

在解除 STOP 模式后, 如果在不更改接收模式的情况下进行接收, 就在将 SCRE 位设定为“1”后开始接收, 如果在更改发送模式的情况下进行接收, 必须从初始设定开始执行。

接收时的模式转移流程图例子如图 16-36 所示。

图 16-36 接收时的模式转移流程图例子



注意 在将 PER1 寄存器的 SMCI_nEN 位设定为“0”后向 STOP 模式转移时, 必须执行初 SMCI 的初始化处理。

备注 n: 通道号 (n = 0、1)

16.5.5 允许接收前的 SCIO_n 引脚状态

如果在 SCIO_n 引脚为低电平的状态下给 SCRE 位写“1”, 就在检测到起始位后开始接收。但是, 如果在起始位的采样点前恢复到高电平, 就返回空闲状态。

备注 n: 通道号 (n = 0、1)

16.5.6 不使用 SMCI 功能的情况

在不使用 SMCI 功能时, 必须将 PER1 寄存器的 SMCI1EN 位和 SMCI0EN 位设定为“0”。

第 17 章 LCD 控制器 / 驱动器

LCD 显示功能的引脚个数如下表所示。

表 17-1 各产品的 LCD 显示功能引脚

项目	R7F0C003、R7F0C004							
LCD 控制器 / 驱动器	段信号输出：51（47）注 公共信号输出：8							
复用 I/O 端口	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
P0	SEG50	SEG49	SEG48	SEG47	SEG46	SEG45	SEG44	SEG43
P1	SEG42	SEG41	SEG40	SEG39	SEG38	SEG37	SEG36	SEG35
P2	SEG34	SEG33	SEG32	SEG31	SEG30	SEG29	—	—
P3	—	—	SEG25	SEG24	SEG23	SEG22	SEG21	SEG20
P4	SEG27	SEG26	—	—	—	—	—	—
P5	SEG11	SEG10	SEG9	SEG8	SEG7	SEG6	SEG5	SEG4
P7	SEG19	SEG18	SEG17	SEG16	SEG15	SEG14	SEG13	SEG12
P13	—	—	—	—	—	—	—	SEG28
不复用 I/O 端口								
COM4	SEG0							
COM5	SEG1							
COM6	SEG2							
COM7	SEG3							

注 （ ）内是使用 8com 时的信号输出个数。

17.1 LCD 控制器 / 驱动器的功能

R7F0C003 和 R7F0C004 内置的 LCD 控制器 / 驱动器的功能如下所示。

- (1) 能选择 A 波形或者 B 波形。
- (2) LCD 驱动电压生成电路能进行内部升压和外部电阻分压的切换。
- (3) 能通过自动读显示数据寄存器进行段信号和公共信号的自动输出。
- (4) 能从升压电路运行时生成的 16 种基准电压（调整对比度）中选择。
- (5) 能进行 LCD 闪烁。

在各显示模式中能显示的最大像素数如表 17-2 所示。

表 17-2 最大显示像素数

LCD 驱动器的 驱动波形	LCD 驱动器的驱 动电压生成电路	偏压法	时间片	最大像素数
A 波形	外部电阻分压	—	静态	51（51 段 × 1 个公共）
		1/2	2	102（51 段 × 2 个公共）
			3	153（51 段 × 3 个公共）
		1/3	3	
			4	204（51 段 × 4 个公共）
		1/4	8	376（47 段 × 8 个公共）
	内部升压	1/3	3	153（51 段 × 3 个公共）
			4	204（51 段 × 4 个公共）
		1/4	6	294（49 段 × 6 个公共）
			8	376（47 段 × 8 个公共）
B 波形	外部电阻分压、 内部升压	1/3	4	204（51 段 × 4 个公共）
		1/4	8	376（47 段 × 8 个公共）

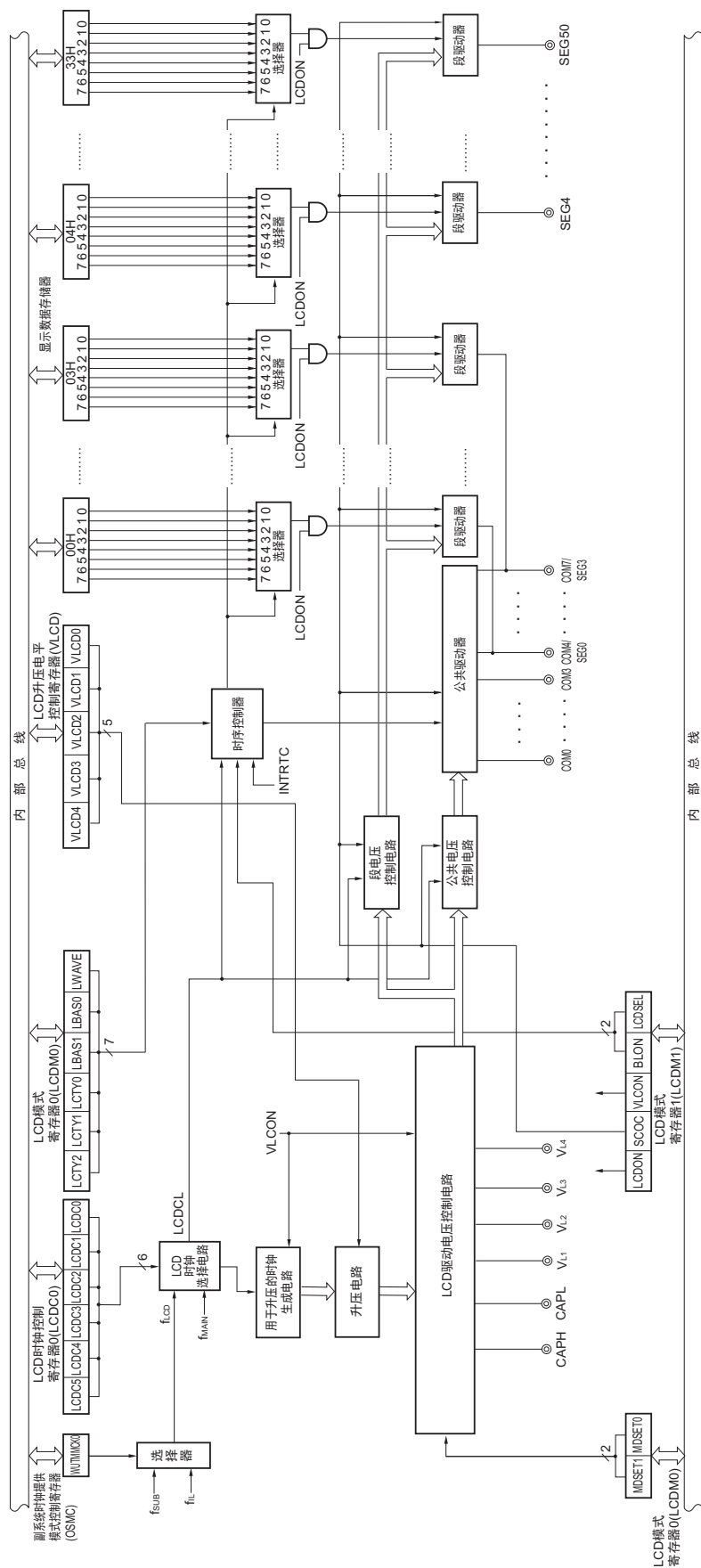
17.2 LCD 控制器 / 驱动器的结构

LCD 控制器 / 驱动器包含由以下硬件构成。

表 17-3 LCD 控制器 / 驱动器的结构

项目	结构
控制寄存器	LCD 模式寄存器 0 (LCDM0) LCD 模式寄存器 1 (LCDM1) 副系统时钟提供模式控制寄存器 (OSMC) LCD 时钟控制寄存器 0 (LCDC0) LCD 升压电平控制寄存器 (VLCD) LCD 输入切换控制寄存器 (ISCLCD) LCD 端口功能寄存器 0 ~ 6 (PFSEG0 ~ PFSEG6) 端口模式寄存器 0 ~ 5、7、13 (PM0 ~ PM5、PM7、PM13)

图 17-1 LCD 控制器 / 驱动器的框图



17.3 控制 LCD 控制器 / 驱动器的寄存器

通过以下 10 种寄存器控制 LCD 控制器 / 驱动器。

- LCD 模式寄存器 0 (LCDM0)
- LCD 模式寄存器 1 (LCDM1)
- 副系统时钟提供模式控制寄存器 (OSMC)
- LCD 时钟控制寄存器 0 (LCDC0)
- LCD 升压电平控制寄存器 (VLCD)
- LCD 输入切换控制寄存器 (ISCLCD)
- LCD 端口功能寄存器 0~6 (PFSEG0~PFSEG6)
- 端口模式寄存器 0~5、7、13 (PM0~PM5、PM7、PM13)

17.3.1 LCD 模式寄存器 0 (LCDM0)

LCDM0 寄存器设定 LCD 的运行。

通过 8 位存储器操作指令设定 LCDM0 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 17-2 LCD 模式寄存器 0 (LCDM0) 的格式

地址：FFF40H 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
LCDM0	MDSET1	MDSET0	LWAVE	LDTY2	LDTY1	LDTY0	LBAS1	LBAS0

MDSET1	MDSET0	LCD 驱动电压生成电路的选择
0	0	外部电阻分压方式
0	1	内部升压方式
上述以外		禁止设定

LWAVE	LCD 显示波形的选择
0	A 波形
1	B 波形

LDTY2	LDTY1	LDTY0	选择 LCD 显示的时间片
0	0	0	静态
0	0	1	2 个时间片
0	1	0	3 个时间片
0	1	1	4 个时间片
1	0	0	6 个时间片
1	0	1	8 个时间片
上述以外			禁止设定

LBAS1	LBAS0	选择 LCD 显示的偏压法
0	0	1/2 偏压法
0	1	1/3 偏压法
1	0	1/4 偏压法
1	1	禁止设定

注意 1. 当 LCDM1 寄存器的 SCOC 位为“1”时，不能改写 LCDM0 寄存器的值。

2. 当选择静态 (LDTY2 ~ LDTY0=000B) 时，必须将 LBAS1 位和 LBAS0 位置初始值 (“00B”)。如果设定初始值以外的值，就不保证运行。

3. 只对应表 17-4 所示的显示波形、时间片、偏压法的组合设定。

禁止表 17-4 所示以外的组合设定。

表 17-4 显示波形、时间片、偏压法与帧频率的组合

显示模式			设定值						驱动电压产生方法	
显示波形	时间片个数	偏压法	LWAVE	LDTY2	LDTY1	LDTY0	LBAS1	LBAS0	外部电阻分压	内部升压
A 波形	8	1/4	0	1	0	1	1	0	○ (24 ~ 128Hz)	○ (24 ~ 64Hz)
A 波形	6	1/4	0	1	0	0	1	0	X	○ (32 ~ 86Hz)
A 波形	4	1/3	0	0	1	1	0	1	○ (24 ~ 128Hz)	○ (24 ~ 128Hz)
A 波形	3	1/3	0	0	1	0	0	1	○ (32 ~ 128Hz)	○ (32 ~ 128Hz)
A 波形	3	1/2	0	0	1	0	0	0	○ (32 ~ 128Hz)	X
A 波形	2	1/2	0	0	0	1	0	0	○ (24 ~ 128Hz)	X
A 波形	静态		0	0	0	0	0	0	○ (24 ~ 128Hz)	X
B 波形	8	1/4	1	1	0	1	1	0	○ (24 ~ 128Hz)	○ (24 ~ 64Hz)
B 波形	4	1/3	1	0	1	1	0	1	○ (24 ~ 128Hz)	○ (24 ~ 128Hz)

备注 ○: 支持

X: 不支持

17.3.2 LCD 模式寄存器 1（LCDM1）

LCDM1 寄存器设定允许或者禁止显示运行、升压电路的运行、显示数据区以及低电压模式。
通过 1 位或者 8 位存储器操作指令设定 LCDM1 寄存器。
在产生复位信号后，此寄存器的值变为“00H”。

图 17-3 LCD 模式寄存器 1（LCDM1）的格式（1/2）

地址：FFF41H 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
LCDM1	LCDON	SCOC	VLCON	BLON	LCDSEL	0	0	LCDVLM

SCOC	LCDON	LCD 显示的允许 / 禁止 输出 A 波形 /B 波形时
0	0	向段引脚 / 公共引脚输出接地电平
0	1	
1	0	显示 OFF（段输出全部为非选择信号输出）
1	1	显示 ON

VLCON	运行 / 停止升压电路的运行
0	停止升压电路的运行
1 注 1	允许升压电路的运行

BLON 注 2	LCDSEL	显示数据区域控制
0	0	显示 A 图形区（LCD 显示数据寄存器的低 4 位）的数据。
0	1	显示 B 图形区（LCD 显示数据寄存器的高 4 位）的数据。
1	0	交替显示 A 图形区和 B 图形区的数据（显示与实时时钟 2（RTC2）的固定周期中断（INTRTC）时序对应的闪烁）
1	1	

注 1. 在外部电阻分压方式时禁止设定。
2. 选择 f_{IL} 作为 LCD 源时钟（f_{LCD}）时，必须将 BLON 位设为“0”。

图 17-3 LCD 模式寄存器 1 (LCDM1) 的格式 (2/2)

地址: FFF41H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
LCDM1	LCDON	SCOC	VLCON	BLON	LCDSEL	0	0	LCDVLM

LCDVLM 注	控制升压引脚初始值的切换
0	$V_{DD} \geq 2.7\text{ V}$ 的情况。
1	$V_{DD} \leq 4.2\text{ V}$ 的情况。

注 这是在使用升压电路时通过设定 V_{Lx} 引脚的初始状态缩短升压稳定时间的功能。

在升压开始时的 V_{DD} 电压不小于 2.7V 的情况下, 必须将 LCDVLM 位置“0”; 在 V_{DD} 电压不大于 4.2V 的情况下, 必须将 LCDVLM 位置“1”。但是, 当 $2.7\text{ V} \leq V_{DD} \leq 4.2\text{ V}$ 时, 无论在 LCDVLM 位为“0”或者 LCDVLM 位为“1”的情况下都能运行。

- 注意 1. 如果在使用升压电路时不进行 LCD 显示, 为了降低功耗, 必须将 SCOC 位和 VLCON 位都置“0”, 并且将 MDSET1 位和 MDSET0 位置“00B”。在 MDSET1 位和 MDSET0 位为“01B”时, 内部基准电压生成部运行, 因此会消耗功率。
2. 在设定外部电阻分压方式 (LCDM0 的 MDSET1、MDSET0=00B) 时, 必须将 LCDVLM 位设定为“0”。
3. 当 SCOC 位为“1”时, 不能改写 VLCON 位和 LCDVLM 位。
4. 当选择 8 个时间片的显示模式时, 必须将 BLON 位和 LCDSEL 位设定为“0”。
5. 当使用内部升压方式时, 必须在通过 VLCD 寄存器设定基准电压后 (当以默认值使用基准电压时, 在选择为内部升压方式 (LCDM0 寄存器的 MDSET1、MDSET0=01B) 后) 等待基准电压准备时间 (5ms(MIN.)), 然后将 VLCON 位设定为“1”。

17.3.3 副系统时钟提供模式控制寄存器（OSMC）

OSMC 寄存器是通过停止不需要的时钟功能来降低功耗的寄存器。

如果将 RTCLPC 位置“1”，就在 STOP 模式和 CPU 以副系统时钟运行的 HALT 模式中停止给实时时钟 2、12 位间隔定时器、时钟输出 / 蜂鸣器输出以及 LCD 控制器 / 驱动器以外的外围功能提供时钟，因此能降低功耗。

能通过 OSMC 寄存器选择实时时钟 2、12 位间隔定时器、时钟输出 / 蜂鸣器输出、LCD 控制器 / 驱动器以及副系统时钟频率测量电路的运行时钟。

通过 8 位存储器操作指令设定 OSMC 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 17-4 副系统时钟提供模式控制寄存器（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 模式以及 CPU 以副系统时钟运行的 HALT 模式中的设定
0	允许给外围功能提供副系统时钟 (有关允许运行的外围功能，请参照表 21-1 ~ 表 21-3)。
1	停止给实时时钟 2、12 位间隔定时器、时钟输出 / 蜂鸣器输出、LCD 控制器 / 驱动器以外的外围功能提供副系统时钟。

WUTMMCK0	选择实时时钟 2、12 位间隔定时器和 LCD 控制器 / 驱动器的运行时钟	选择时钟输出 / 蜂鸣器输出的 PCLBUZn 引脚的时钟输出	副系统时钟频率测量电路的运行
0	副系统时钟 (f_{SUB})	允许选择副系统时钟 (f_{SUB})。	有效
1	低速内部振荡器时钟 (f_{IL})	禁止选择副系统时钟 (f_{SUB})。	无效

注意 1. 副系统时钟振荡时，必须选择副系统时钟（WUTMMCK0 位 = 0）。

2. 如果将 WUTMMCK0 设定为“1”，低速内部振荡器时钟就振荡。

3. 只有在停止实时时钟 2、12 位间隔定时器和 LCD 控制器 / 驱动器的全部功能时，才能通过 WUTMMCK0 位进行副系统时钟和低速内部振荡器时钟的切换。

备注 RTCE：实时时钟控制寄存器 0（RTCC0）的 bit7

RINTE：间隔定时器控制寄存器（ITMC）的 bit15

SCOC：LCD 模式寄存器 1（LCDM1）的 bit6

VLCON：LCD 模式寄存器 1（LCDM1）的 bit5

17.3.4 LCD 时钟控制寄存器 0 (LCDC0)

LCDC0 寄存器是设定 LCD 时钟的寄存器。

帧频率取决于 LCD 时钟和时间片的个数。

通过 8 位存储器操作指令设定 LCDC0 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 17-5 LCD 时钟控制寄存器 0 (LCDC0) 的格式

地址: FFF42H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
LCDC0	0	0	LCDC05	LCDC04	LCDC03	LCDC02	LCDC01	LCDC00

LCDC05	LCDC04	LCDC03	LCDC02	LCDC01	LCDC00	LCD 时钟 (LCDCL)
0	0	0	1	0	0	$f_{\text{SUB}}/2^5$ or $f_{\text{IL}}/2^5$
0	0	0	1	0	1	$f_{\text{SUB}}/2^6$ or $f_{\text{IL}}/2^6$
0	0	0	1	1	0	$f_{\text{SUB}}/2^7$ or $f_{\text{IL}}/2^7$
0	0	0	1	1	1	$f_{\text{SUB}}/2^8$ or $f_{\text{IL}}/2^8$
0	0	1	0	0	0	$f_{\text{SUB}}/2^9$ or $f_{\text{IL}}/2^9$
0	0	1	0	0	1	$f_{\text{SUB}}/2^{10}$
0	1	0	0	1	1	$f_{\text{MAIN}}/2^{10}$
0	1	0	1	0	0	$f_{\text{MAIN}}/2^{11}$
0	1	0	1	0	1	$f_{\text{MAIN}}/2^{12}$
0	1	0	1	1	0	$f_{\text{MAIN}}/2^{13}$
0	1	0	1	1	1	$f_{\text{MAIN}}/2^{14}$
0	1	1	0	0	0	$f_{\text{MAIN}}/2^{15}$
0	1	1	0	0	1	$f_{\text{MAIN}}/2^{16}$
0	1	1	0	1	0	$f_{\text{MAIN}}/2^{17}$
0	1	1	0	1	1	$f_{\text{MAIN}}/2^{18}$
1	0	1	0	1	1	$f_{\text{MAIN}}/2^{19}$
上述以外						禁止设定

注意 1. 在 LCDM1 寄存器的 SCOC 位为“1”时不能设定 LCDC0。

2. 必须将 bit6 和 bit7 设定为“0”。

3. 在设定为内部升压方式时，必须如下设定 LCD 时钟 (LCDCL)。

- 选择 f_{SUB} 时不大于 512Hz
- 选择 f_{IL} 时不大于 235Hz

详细内容请参照“表 17-4 显示波形、时间片、偏压法与帧频率的组合”。

备注 f_{MAIN} : 主系统时钟频率

f_{SUB} : 副系统时钟频率

f_{IL} : 低速内部振荡器时钟频率

17.3.5 LCD 升压电平控制寄存器 (VLCD)

VLCD 寄存器是选择升压电路运行时生成的基准电压（调整对比度）的寄存器。能选择 16 种基准电压。

通过 8 位存储器操作指令设定 VLCD 寄存器。

在产生复位信号后，此寄存器的值变为“04H”。

图 17-6 LCD 升压电平控制寄存器 (VLCD) 的格式

地址: FFF43H 复位后: 04H R/W

符号	7	6	5	4	3	2	1	0
VLCD	0	0	0	VLCD4	VLCD3	VLCD2	VLCD1	VLCD0

VLCD4	VLCD3	VLCD2	VLCD1	VLCD0	基准电压选择 (调整对比度)	V _{L4} 电压	
						1/3 偏压法	1/4 偏压法
0	0	1	0	0	1.00 V (默认)	3.00 V	4.00 V
0	0	1	0	1	1.05 V	3.15 V	4.20 V
0	0	1	1	0	1.10 V	3.30 V	4.40 V
0	0	1	1	1	1.15 V	3.45 V	4.60 V
0	1	0	0	0	1.20 V	3.60 V	4.80 V
0	1	0	0	1	1.25 V	3.75 V	5.00 V
0	1	0	1	0	1.30 V	3.90 V	5.20 V
0	1	0	1	1	1.35 V	4.05 V	禁止设定
0	1	1	0	0	1.40 V	4.20 V	禁止设定
0	1	1	0	1	1.45 V	4.35 V	禁止设定
0	1	1	1	0	1.50 V	4.50 V	禁止设定
0	1	1	1	1	1.55 V	4.65 V	禁止设定
1	0	0	0	0	1.60 V	4.80 V	禁止设定
1	0	0	0	1	1.65 V	4.95 V	禁止设定
1	0	0	1	0	1.70 V	5.10 V	禁止设定
1	0	0	1	1	1.75 V	5.25 V	禁止设定
上述以外					禁止设定		

注意 1. 只有在升压电路运行时，VLCD 寄存器的设定才有效。

2. 必须将 bit5 ~ 7 置“0”。

3. 必须在停止升压电路的运行 (VLCON=0) 后更改 VLCD 寄存器的值。

4. 当使用内部升压方式时，必须在通过 VLCD 寄存器设定基准电压后（当以默认值使用基准电压时，在选择为内部升压方式 (LCDM0 寄存器的 MDSET1、MDSET0=01B) 后）等待基准电压准备时间 (5ms(MIN.))，然后将 VLCON 位置“1”。

5. 在外部电阻分压方式下，VLCD 寄存器必须为初始值“04H”。

17.3.6 LCD 输入切换控制寄存器 (ISCLCD)

为了在设定 CAPL/P126、CAPH/P127、 V_{L3} /P125 引脚作为 LCD 功能运行的期间防止贯通电流的流入，需要禁止施密特触发缓冲器的输入。

通过 1 位或者 8 位存储器操作指令设定 ISCLCD 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 17-7 LCD 输入切换控制寄存器 (ISCLCD) 的格式

地址: F0308H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
ISCLCD	0	0	0	0	0	0	ISCVL3	ISCCAP

ISCVL3	V_{L3} /P125 引脚的施密特触发缓冲器的控制
0	输入无效
1	输入有效

ISCCAP	CAPL/P126 和 CAPH/P127 引脚的施密特触发缓冲器的控制
0	输入无效
1	输入有效

注意 当 ISCVL3 = 0、ISCCAP = 0 时，如下设定对应的端口控制寄存器。

PU12 寄存器的 PU127 位 = 0，P12 寄存器的 P127 位 = 0

PU12 寄存器的 PU126 位 = 0，P12 寄存器的 P126 位 = 0

PU12 寄存器的 PU125 位 = 0，P12 寄存器的 P125 位 = 0

(a) V_{L3} 、CAPL、CAPH 引脚复用端口的运行

V_{L3} /P125、CAPL/P126、CAPH/P127 引脚的功能取决于 LCD 输入切换控制寄存器 (ISCLCD)、LCD 模式寄存器 0 (LCDM0) 和端口模式寄存器 12 (PM12) 的设定。

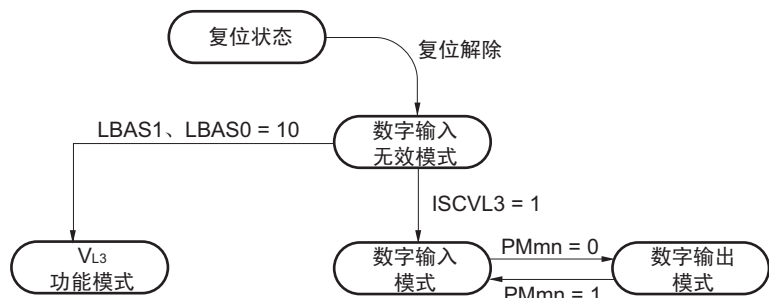
- V_{L3} /P125

表 17-5 V_{L3} /P125 引脚功能的设定

偏压法的设定 (通过 LCDM0 寄存器的 LBAS1 位和 LBAS0 位设定)	ISCLCD 寄存器的 ISCVL3 位	PM12 寄存器的 PM125 位	引脚功能	初始状态
1/4 偏压法以外 (LBAS1、LBAS0=00 或者 01)	0	1	数字输入无效模式	○
	1	0	数字输出模式	—
	1	1	数字输入模式	—
1/4 偏压法 (LBAS1、LBAS0=10)	0	1	V_{L3} 功能模式	—
上述以外			禁止设定	

V_{L3}/P125 引脚功能的状态转移如下所示。

图 17-8 V_{L3}/P125 引脚功能的状态转移图



注意 要设定为 V_{L3} 功能模式时，必须在段输出开始前（LCD 模式寄存器 1（LCDM1）的 SCOC=0 的期间）进行设定。

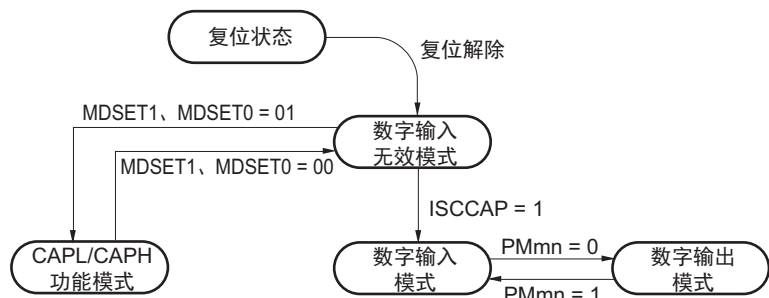
- CAPL/P126、CAPH/P127

表 17-6 CAPL/P126、CAPH/P127 引脚功能的设定

生成 LCD 驱动电压 (LCDM0 寄存器的 MDSET1 位 和 MDSET0 位)	ISCLCD 寄存器的 ISCCAP 位	PM12 寄存器的 PM126 位和 PM127 位	引脚功能	初始状态
外部电阻分压 (MDSET1、MDSET0=00)	0	1	数字输入无效模式	○
	1	0	数字输出模式	—
	1	1	数字输入模式	—
内部升压 (MDSET1、MDSET0=01)	0	1	CAPL/CAPH 功能模式	—
上述以外			禁止设定	

CAPL/P126 和 CAPH/P127 引脚功能的状态转移如下所示。

图 17-9 CAPL/P126 和 CAPH/P127 引脚功能的状态转移图



注意 要设定为 CAPL/CAPH 功能模式时，必须在段输出开始前（LCD 模式寄存器 1（LCDM1）的 SCOC=0 的期间）进行设定。

17.3.7 LCD 端口功能寄存器 0 ~ 6 (PFSEG0 ~ PFSEG6)

这些寄存器设定将 P00 ~ P07、P10 ~ P17、P22 ~ P27、P30 ~ P35、P46、P47、P50 ~ P57、P70 ~ P77、P130 引脚用作端口（段输出除外）或者段输出，将 P45 引脚用作端口或者模拟输入。通过 1 位或者 8 位存储器操作指令设定 PFSEG0 ~ PFSEG6 寄存器。

在产生复位信号后，此寄存器的值变为“FFH”（PFSEG0 为“F0H”、PFSEG6 为“0FH”）。

备注 段输出引脚（SEGxx）和 PFSEG 寄存器（PFSEGxx 位）的对应如“表 17-7 与段输出引脚对应的 PFSEG 寄存器（PFSEG 位）”所示。

图 17-10 LCD 端口功能寄存器的格式

地址: F0300H	复位后: F0H	R/W						
符号	7	6	5	4	3	2	1	0
PFSEG0	PFSEG07	PFSEG06	PFSEG05	PFSEG04	0	0	0	0

地址: F0301H		复位后: FFH		R/W				
符号	7	6	5	4	3	2	1	0
PFSEG1	PFSEG15	PFSEG14	PFSEG13	PFSEG12	PFSEG11	PFSEG10	PFSEG09	PFSEG08

地址: F0302H		复位后: FFH		R/W				
符号	7	6	5	4	3	2	1	0
PFSEG2	PFSEG23	PFSEG22	PFSEG21	PFSEG20	PFSEG19	PFSEG18	PFSEG17	PFSEG16

地址: F0303H		复位后: FFH		R/W				
符号	7	6	5	4	3	2	1	0
PFSEG3	PFSEG30	PFSEG29	PFSEG28	PFSEG27	PFSEG26	PFDEG	PFSEG25	PFSEG24

地址: F0304H		复位后: FFH		R/W				
符号	7	6	5	4	3	2	1	0
PFSEG4	PFSEG38	PFSEG37	PFSEG36	PFSEG35	PFSEG34	PFSEG33	PFSEG32	PFSEG31

地址: F0305H		复位后: FFH		R/W				
符号	7	6	5	4	3	2	1	0
PFSEG5	PFSEG46	PFSEG45	PFSEG44	PFSEG43	PFSEG42	PFSEG41	PFSEG40	PFSEG39

地址: F0306H	复位后: 0FH		R/W					
符号	7	6	5	4	3	2	1	0
PFSEG6	0	0	0	0	PFSEG50	PFSEG49	PFSEG48	PFSEG47

PFSEGxx (xx=04~50)	Pmn 引脚的端口（段输出除外）或者段输出的指定 (mn=00~07、10~17、22~27、30~35、46、47、50~57、70~77、130)
0	用作端口（段输出除外）。
1	用作段输出。

PFDEG	P45 引脚的端口或者模拟输入（IVREF）的指定
0	用作端口。
1	模拟输入（IVREF0）。

备注 当用作段输出（PFSEGxx=1）时，必须将 PUm 寄存器的 PUm_n 位、POMm 寄存器的 POM_n 位和 PIMm 寄存器的 PIM_n 位都置“0”。

表 17-7 与段输出引脚对应的 PFSEG 寄存器 (PFSEG 位)

PFSEG 寄存器的位名称	对应的 SEGxx 引脚	复用端口
PFSEG04	SEG4	P50
PFSEG05	SEG5	P51
PFSEG06	SEG6	P52
PFSEG07	SEG7	P53
PFSEG08	SEG8	P54
PFSEG09	SEG9	P55
PFSEG10	SEG10	P56
PFSEG11	SEG11	P57
PFSEG12	SEG12	P70
PFSEG13	SEG13	P71
PFSEG14	SEG14	P72
PFSEG15	SEG15	P73
PFSEG16	SEG16	P74
PFSEG17	SEG17	P75
PFSEG18	SEG18	P76
PFSEG19	SEG19	P77
PFSEG20	SEG20	P30
PFSEG21	SEG21	P31
PFSEG22	SEG22	P32
PFSEG23	SEG23	P33
PFSEG24	SEG24	P34
PFSEG25	SEG25	P35
PFSEG26	SEG26	P46
PFSEG27	SEG27	P47
PFSEG28	SEG28	P130
PFSEG29	SEG29	P22
PFSEG30	SEG30	P23
PFSEG31	SEG31	P24
PFSEG32	SEG32	P25
PFSEG33	SEG33	P26
PFSEG34	SEG34	P27
PFSEG35	SEG35	P10
PFSEG36	SEG36	P11
PFSEG37	SEG37	P12
PFSEG38	SEG38	P13
PFSEG39	SEG39	P14
PFSEG40	SEG40	P15
PFSEG41	SEG41	P16
PFSEG42	SEG42	P17
PFSEG43	SEG43	P00
PFSEG44	SEG44	P01
PFSEG45	SEG45	P02
PFSEG46	SEG46	P03
PFSEG47	SEG47	P04
PFSEG48	SEG48	P05
PFSEG49	SEG49	P06
PFSEG50	SEG50	P07

(a) SEGxx 引脚复用端口的运行

段输出引脚（SEGxx）的功能取决于端口模式控制寄存器（PMCxx）、端口模式寄存器（PMxx）和 LCD 端口功能寄存器 0 ~ 6（PFSEG0 ~ PFSEG6）的设定。

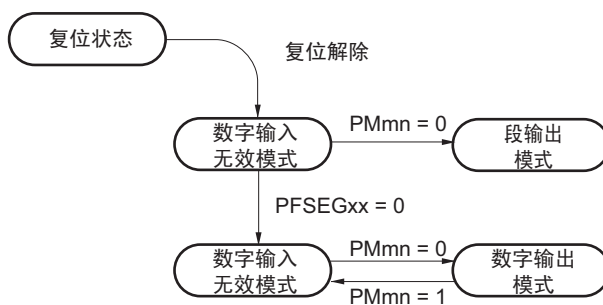
- P00 ~ P07、P14 ~ P17、P30 ~ P35、P46、P47、P50 ~ P57、P70 ~ P77、P130
（不复用模拟输入引脚（ANLxx）的端口）

表 17-8 SEGxx/ 端口引脚功能的设定

PFSEG0 ~ PFSEG6 寄存器的 PFSEGxx 位	PMxx 寄存器的 PMxx 位	引脚功能	初始状态
1	1	数字输入无效模式	○
0	0	数字输出模式	—
0	1	数字输入模式	—
1	0	段输出模式	—

SEGxx/ 端口引脚功能的状态转移如下所示。

图 17-11 SEGxx/ 端口引脚功能的状态转移图



注意 要设定为段输出模式时，必须在段输出开始前（LCD 模式寄存器 1（LCDM1）的 SCOC=0 的期间）进行设定。

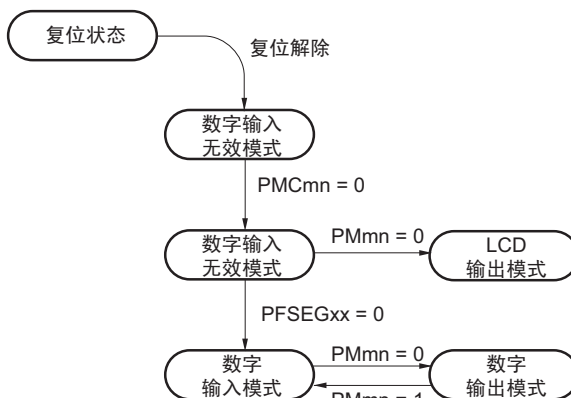
- P10 ~ P13、P24 ~ P27（不复用模拟输入引脚（ANLxx）的端口）

表 17-9 SEGxx/ 端口引脚功能的设定

PMCxx 寄存器的 PMCxx 位	PFSEG3 和 PFSEG4 寄存器的 PFSEGxx 位	PMxx 寄存器的 PMxx 位	引脚功能	初始状态
1	1	1	数字输入无效模式	○
0	0	0	数字输出模式	—
0	0	1	数字输入模式	—
0	1	0	段输出模式	—
0	1	1	数字输入无效模式	—
上述以外			禁止设定	

SEGxx/ 端口引脚功能的状态转移如下所示。

图 17-12 SEGxx/ 端口引脚功能的状态转移图



注意 要设定为段输出模式时，必须在段输出开始前（LCD 模式寄存器 1（LCDM1）的 SCOC=0 的期间）进行设定。

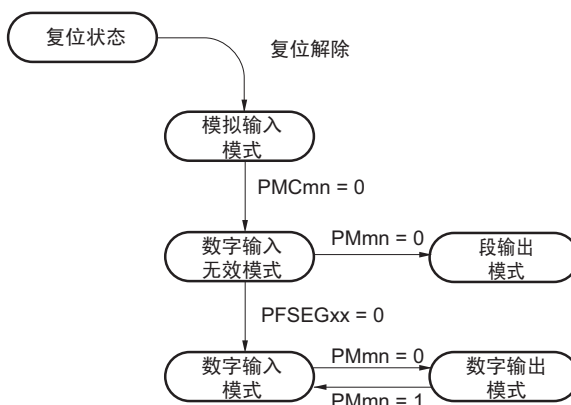
- P22、P23（复用模拟输入引脚（ANIdx）的端口）

表 17-10 ANIdx/SEGxx/ 端口引脚功能的设定

PMCxx 寄存器的 PMCxx 位	PFSEG3 寄存器的 PFSEGxx 位	PMxx 寄存器的 PMxx 位	引脚功能	初始状态
1	1	1	模拟输入模式	○
0	0	0	数字输出模式	—
0	0	1	数字输入模式	—
0	1	0	段输出模式	—
0	1	1	数字输入无效模式	—
上述以外			禁止设定	

ANIdx/SEGxx/ 端口引脚功能的状态转移如下所示。

图 17-13 ANIdx/SEGxx/ 端口引脚功能的状态转移图



注意 要设定为段输出模式时，必须在段输出开始前（LCD 模式寄存器 1（LCDM1）的 SCOC=0 的期间）进行设定。

17.3.8 端口模式寄存器 0 ~ 5、7、13 (PM0 ~ PM5、PM7、PM13)

这些寄存器以 1 位为单位设定端口 0 ~ 5、7 和 13 的输入 / 输出。

在将段输出引脚复用端口 (P00/SEG43/SO00/TxD0/TOOLTxD) 用作段输出时, 必须将各端口对应的端口模式寄存器 (PM_{xx}) 的位和端口寄存器 (P_{xx}) 的位置“0”。

例) 将 P00/SEG43/SO00/TxD0/TOOLTxD 用作段输出时

将端口模式寄存器 0 的 PM00 位设定为“0”。

将端口寄存器 0 的 P00 位设定为“0”。

通过 1 位或者 8 位存储器操作指令设定 PM0 ~ PM5、PM7、PM13 寄存器。

在产生复位信号后, 此寄存器的值变为“FFH”。

图 17-14 端口模式寄存器 0 ~ 5、7、13 (PM0 ~ PM5、PM7、PM13) 的格式

符号	7	6	5	4	3	2	1	0	地址	复位后	R/W
PM0	PM07	PM06	PM05	PM04	PM03	PM02	PM01	PM00	FFF20H	FFH	R/W
PM1	PM17	PM16	PM15	PM14	PM13	PM12	PM11	PM10	FFF21H	FFH	R/W
PM2	PM27	PM26	PM25	PM24	PM23	PM22	PM21	PM20	FFF22H	FFH	R/W
PM3	1	1	PM35	PM34	PM33	PM32	PM31	PM30	FFF23H	FFH	R/W
PM4	PM47	PM46	PM45	PM44	PM43	PM42	PM41	PM40	FFF24H	FFH	R/W
PM5	PM57	PM56	PM55	PM54	PM53	PM52	PM51	PM50	FFF25H	FFH	R/W
PM7	PM77	PM76	PM75	PM74	PM73	PM72	PM71	PM70	FFF27H	FFH	R/W
PM13	1	1	1	1	1	1	1	PM130	FFF2DH	FFH	R/W

PM _m n	选择 P _m n 引脚的输入 / 输出模式 (m = 0 ~ 5、7、13, n = 0 ~ 7)
0	输出模式 (输出缓冲器 ON)
1	输入模式 (输出缓冲器 OFF)

17.4 LCD 显示数据寄存器

LCD 显示数据寄存器的映像如表 17-11 所示。能通过更改 LCD 显示数据寄存器的内容，更改 LCD 的显示内容。

表 17-11 LCD 显示数据寄存器的内容与段输出 / 公共输出的关系 (1/4)

(a) 6 个时间片和 8 个时间片以外 (静态、2 个时间片、3 个时间片、4 个时间片) (1/2)

寄存器名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
		COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
SEG0	F0400H	SEG0 (B 图形区)				SEG0 (A 图形区)			
SEG1	F0401H	SEG1 (B 图形区)				SEG1 (A 图形区)			
SEG2	F0402H	SEG2 (B 图形区)				SEG2 (A 图形区)			
SEG3	F0403H	SEG3 (B 图形区)				SEG3 (A 图形区)			
SEG4	F0404H	SEG4 (B 图形区)				SEG4 (A 图形区)			
SEG5	F0405H	SEG5 (B 图形区)				SEG5 (A 图形区)			
SEG6	F0406H	SEG6 (B 图形区)				SEG6 (A 图形区)			
SEG7	F0407H	SEG7 (B 图形区)				SEG7 (A 图形区)			
SEG8	F0408H	SEG8 (B 图形区)				SEG8 (A 图形区)			
SEG9	F0409H	SEG9 (B 图形区)				SEG9 (A 图形区)			
SEG10	F040AH	SEG10 (B 图形区)				SEG10 (A 图形区)			
SEG11	F040BH	SEG11 (B 图形区)				SEG11 (A 图形区)			
SEG12	F040CH	SEG12 (B 图形区)				SEG12 (A 图形区)			
SEG13	F040DH	SEG13 (B 图形区)				SEG13 (A 图形区)			
SEG14	F040EH	SEG14 (B 图形区)				SEG14 (A 图形区)			
SEG15	F040FH	SEG15 (B 图形区)				SEG15 (A 图形区)			
SEG16	F0410H	SEG16 (B 图形区)				SEG16 (A 图形区)			
SEG17	F0411H	SEG17 (B 图形区)				SEG17 (A 图形区)			
SEG18	F0412H	SEG18 (B 图形区)				SEG18 (A 图形区)			
SEG19	F0413H	SEG19 (B 图形区)				SEG19 (A 图形区)			
SEG20	F0414H	SEG20 (B 图形区)				SEG20 (A 图形区)			
SEG21	F0415H	SEG21 (B 图形区)				SEG21 (A 图形区)			
SEG22	F0416H	SEG22 (B 图形区)				SEG22 (A 图形区)			
SEG23	F0417H	SEG23 (B 图形区)				SEG23 (A 图形区)			
SEG24	F0418H	SEG24 (B 图形区)				SEG24 (A 图形区)			
SEG25	F0419H	SEG25 (B 图形区)				SEG25 (A 图形区)			
禁止使用	F041AH	禁止使用							
SEG26	F041BH	SEG26 (B 图形区)				SEG26 (A 图形区)			
SEG27	F041CH	SEG27 (B 图形区)				SEG27 (A 图形区)			
SEG28	F041DH	SEG28 (B 图形区)				SEG28 (A 图形区)			
SEG29	F041EH	SEG29 (B 图形区)				SEG29 (A 图形区)			
SEG30	F041FH	SEG30 (B 图形区)				SEG30 (A 图形区)			
SEG31	F0420H	SEG31 (B 图形区)				SEG31 (A 图形区)			
SEG32	F0421H	SEG32 (B 图形区)				SEG32 (A 图形区)			

表 17-11 LCD 显示数据寄存器的内容与段输出 / 公共输出的关系 (2/4)

(a) 6 个时间片和 8 个时间片以外 (静态、2 个时间片、3 个时间片、4 个时间片) (2/2)

寄存器名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
		COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
SEG33	F0422H	SEG33 (B 图形区)				SEG33 (A 图形区)			
SEG34	F0423H	SEG34 (B 图形区)				SEG34 (A 图形区)			
SEG35	F0424H	SEG35 (B 图形区)				SEG35 (A 图形区)			
SEG36	F0425H	SEG36 (B 图形区)				SEG36 (A 图形区)			
SEG37	F0426H	SEG37 (B 图形区)				SEG37 (A 图形区)			
SEG38	F0427H	SEG38 (B 图形区)				SEG38 (A 图形区)			
SEG39	F0428H	SEG39 (B 图形区)				SEG39 (A 图形区)			
SEG40	F0429H	SEG40 (B 图形区)				SEG40 (A 图形区)			
SEG41	F042AH	SEG41 (B 图形区)				SEG41 (A 图形区)			
SEG42	F042BH	SEG42 (B 图形区)				SEG42 (A 图形区)			
SEG43	F042CH	SEG43 (B 图形区)				SEG43 (A 图形区)			
SEG44	F042DH	SEG44 (B 图形区)				SEG44 (A 图形区)			
SEG45	F042EH	SEG45 (B 图形区)				SEG45 (A 图形区)			
SEG46	F042FH	SEG46 (B 图形区)				SEG46 (A 图形区)			
SEG47	F0430H	SEG47 (B 图形区)				SEG47 (A 图形区)			
SEG48	F0431H	SEG48 (B 图形区)				SEG48 (A 图形区)			
SEG49	F0432H	SEG49 (B 图形区)				SEG49 (A 图形区)			
SEG50	F0433H	SEG50 (B 图形区)				SEG50 (A 图形区)			

表 17-11 LCD 显示数据寄存器的内容与段输出 / 公共输出的关系 (3/4)

(b) 6 个时间片和 8 个时间片 (1/2)

寄存器名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
		COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
SEG0	F0400H	SEG0 注							
SEG1	F0401H	SEG1 注							
SEG2	F0402H	SEG2 注							
SEG3	F0403H	SEG3 注							
SEG4	F0404H	SEG4							
SEG5	F0405H	SEG5							
SEG6	F0406H	SEG6							
SEG7	F0407H	SEG7							
SEG8	F0408H	SEG8							
SEG9	F0409H	SEG9							
SEG10	F040AH	SEG10							
SEG11	F040BH	SEG11							
SEG12	F040CH	SEG12							
SEG13	F040DH	SEG13							
SEG14	F040EH	SEG14							
SEG15	F040FH	SEG15							
SEG16	F0410H	SEG16							
SEG17	F0411H	SEG17							
SEG18	F0412H	SEG18							
SEG19	F0413H	SEG19							
SEG20	F0414H	SEG20							
SEG21	F0415H	SEG21							
SEG22	F0416H	SEG22							
SEG23	F0417H	SEG23							
SEG24	F0418H	SEG24							
SEG25	F0419H	SEG25							
禁止使用	F041AH	禁止使用							
SEG26	F041BH	SEG26							
SEG27	F041CH	SEG27							
SEG28	F041DH	SEG28							
SEG29	F041EH	SEG29							
SEG30	F041FH	SEG30							
SEG31	F0420H	SEG31							
SEG32	F0421H	SEG32							
SEG33	F0422H	SEG33							
SEG34	F0423H	SEG34							
SEG35	F0424H	SEG35							
SEG36	F0425H	SEG36							
SEG37	F0426H	SEG37							

注 COM4 ~ COM7 引脚与 SEG0 ~ SEG3 引脚复用。

表 17-11 LCD 显示数据寄存器的内容与段输出 / 公共输出的关系 (4/4)

(b) 6 个时间片和 8 个时间片 (2/2)

寄存器名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
		COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
SEG38	F0427H	SEG38							
SEG39	F0428H	SEG39							
SEG40	F0429H	SEG40							
SEG41	F042AH	SEG41							
SEG42	F042BH	SEG42							
SEG43	F042CH	SEG43							
SEG44	F042DH	SEG44							
SEG45	F042EH	SEG45							
SEG46	F042FH	SEG46							
SEG47	F0430H	SEG47							
SEG48	F0431H	SEG48							
SEG49	F0432H	SEG49							
SEG50	F0433H	SEG50							

当用于静态、2 个时间片、3 个时间片或者 4 个时间片时，LCD 显示数据寄存器的各地址的低 4 位为 A 图形区，高 4 位为 B 图形区。

A 图形区的数据和 COM 信号的对应为 bit0↔COM0、bit1↔COM1、bit2↔COM2、bit3↔COM3。

B 图形区的数据和 COM 信号的对应为 bit4↔COM0、bit5↔COM1、bit6↔COM2、bit7↔COM3。

当 BLON 位和 LCDSEL 位都为“0”时，LCD 显示屏显示 A 图形区的数据；当 BLON 位为“0”并且 LCDSEL 位为“1”时，LCD 显示屏显示 B 图形区的数据。

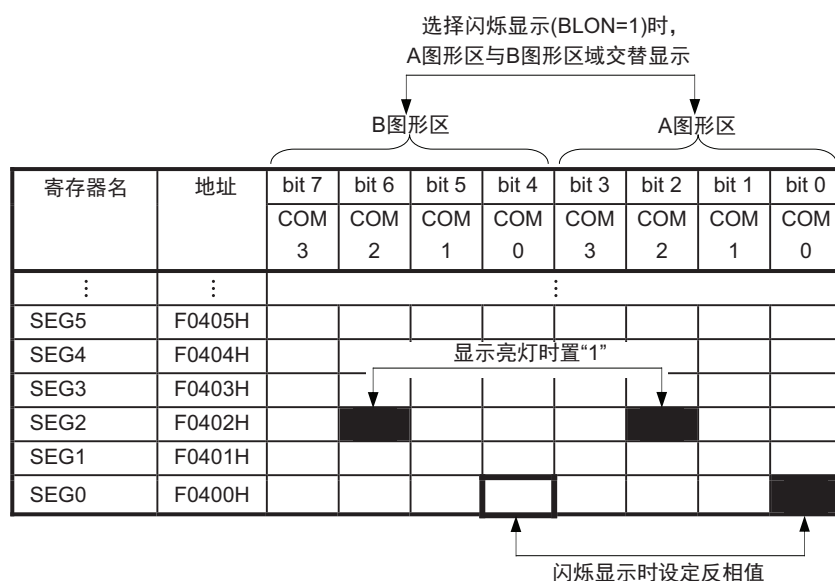
17.5 LCD 显示寄存器的选择

当用于静态、2 个时间片、3 个时间片或者 4 个时间片时，R7F0C003 和 R7F0C004 能通过设定 BLON 位和 LCDSEL 位，从以下 3 种选择 LCD 显示数据寄存器。

- A 图形区（LCD 显示数据寄存器的低 4 位）的数据显示
- B 图形区（LCD 显示数据寄存器的高 4 位）的数据显示
- 交替显示 A 图形区和 B 图形区的数据（显示与实时时钟 2（RTC2）的固定周期中断时序对应的闪烁）

注意 在使用 6 个时间片或者 8 个时间片时，不能选择 LCD 显示数据寄存器（A 图形、B 图形或者闪烁显示）。

图 17-15 图形切换显示时的 LCD 显示寄存器的设定例子



17.5.1 A 图形区和 B 图形区的数据显示

当 BLON 位和 LCDSEL 位都为“0”时，将 A 图形区（LCD 显示数据寄存器的低 4 位）的数据作为 LCD 显示寄存器进行输出。

当 BLON 位为“0”并且 LCDSEL 位为“1”时，将 B 图形区（LCD 显示数据寄存器的高 4 位）的数据作为 LCD 显示寄存器进行输出。

有关显示区请参照“17.4 LCD 显示数据寄存器”。

17.5.2 闪烁显示（A 图形区和 B 图形区的数据的交替显示）

当 BLON 位为“1”时，对应实时时钟 2（RTC2）的固定周期中断（INTRTC）时序，进行 A 图形区和 B 图形区的数据交替显示。有关 RTC 的固定周期中断（INTRTC、只限于 0.5s 的设定）时序的设定，请参照“第 7 章 实时时钟 2”。

当 LCD 闪烁显示时，必须给与 A 图形区的位对应的 B 图形区的位设定反相值（ex. 将 F0400H 的 bit0 置“1”，在闪烁显示时将 F0400H 的 bit4 置“0”）；当 LCD 不闪烁显示时，必须设定相同值（ex. 将 F0402H 的 bit2 置“1”，在亮灯显示时将 F0402H 的 bit6 置“1”）。

有关显示区请参照“17.4 LCD 显示数据寄存器”。

显示的切换时序如下所示。

图 17-16 从 A 图形显示到闪烁显示的切换运行

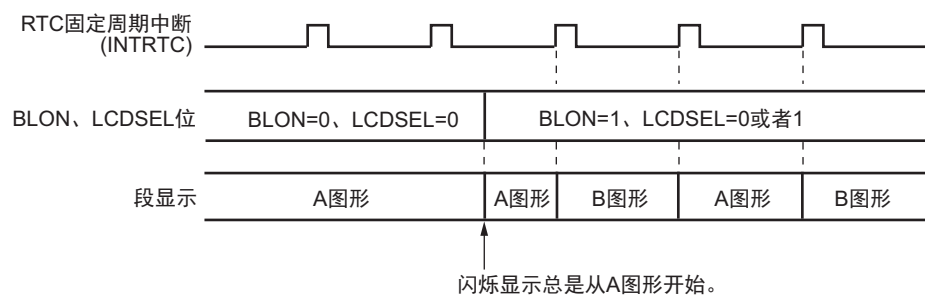
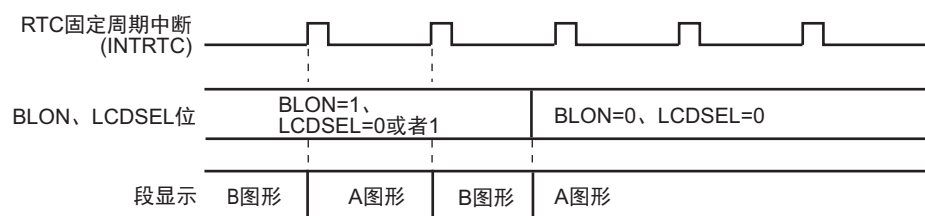


图 17-17 从闪烁显示到 A 图形显示的切换运行



17.6 LCD 控制器 / 驱动器的设定

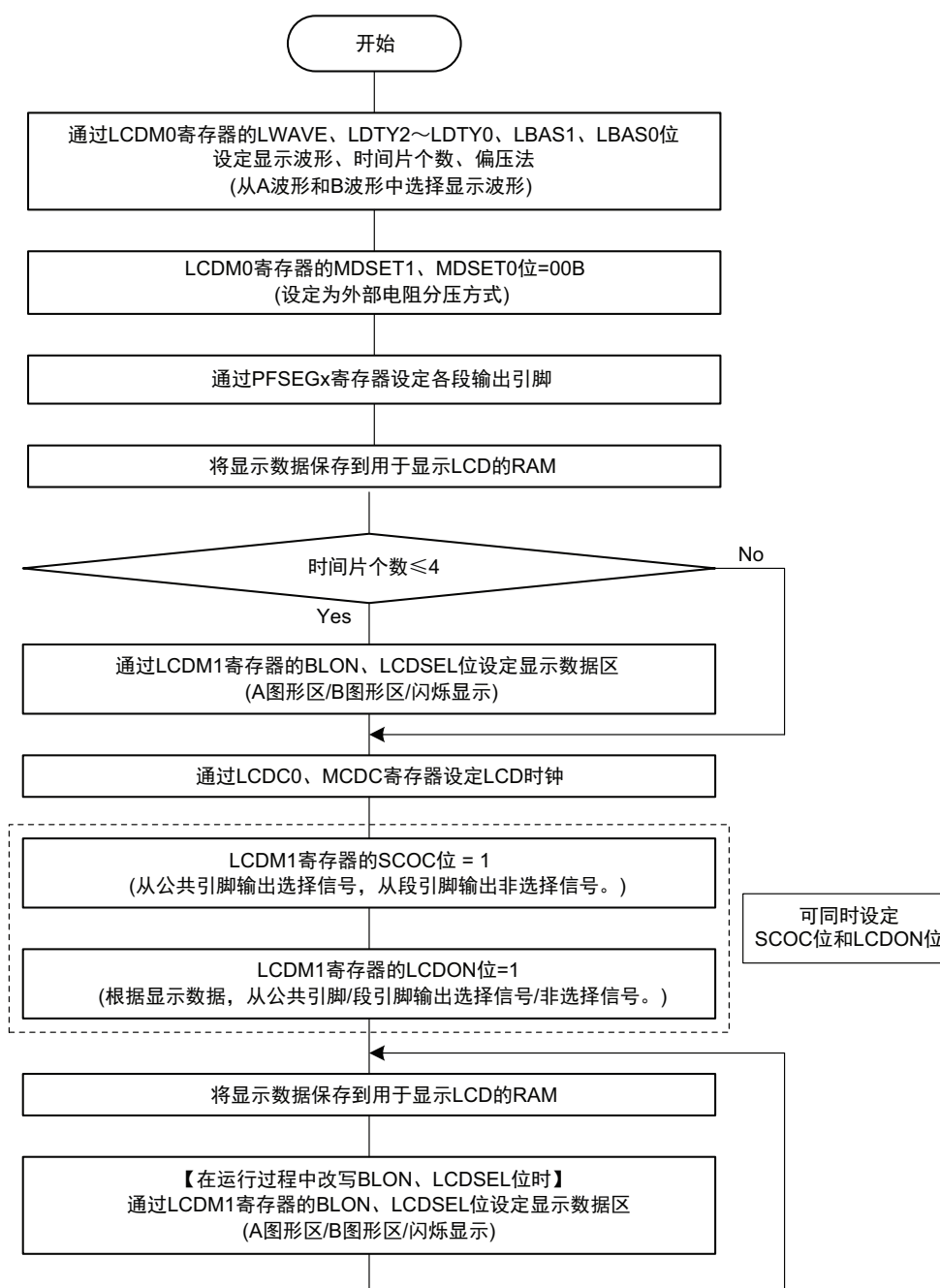
按照以下的步骤设定 LCD 控制器 / 驱动器。

注意 1. 要运行 LCD 控制器 / 驱动器时，必须按照 (1)、(2) 的设定步骤进行设定。如果不按照设定步骤进行，就不保证运行。

2. (1)、(2) 的设定步骤全部为 CPU 的处理。

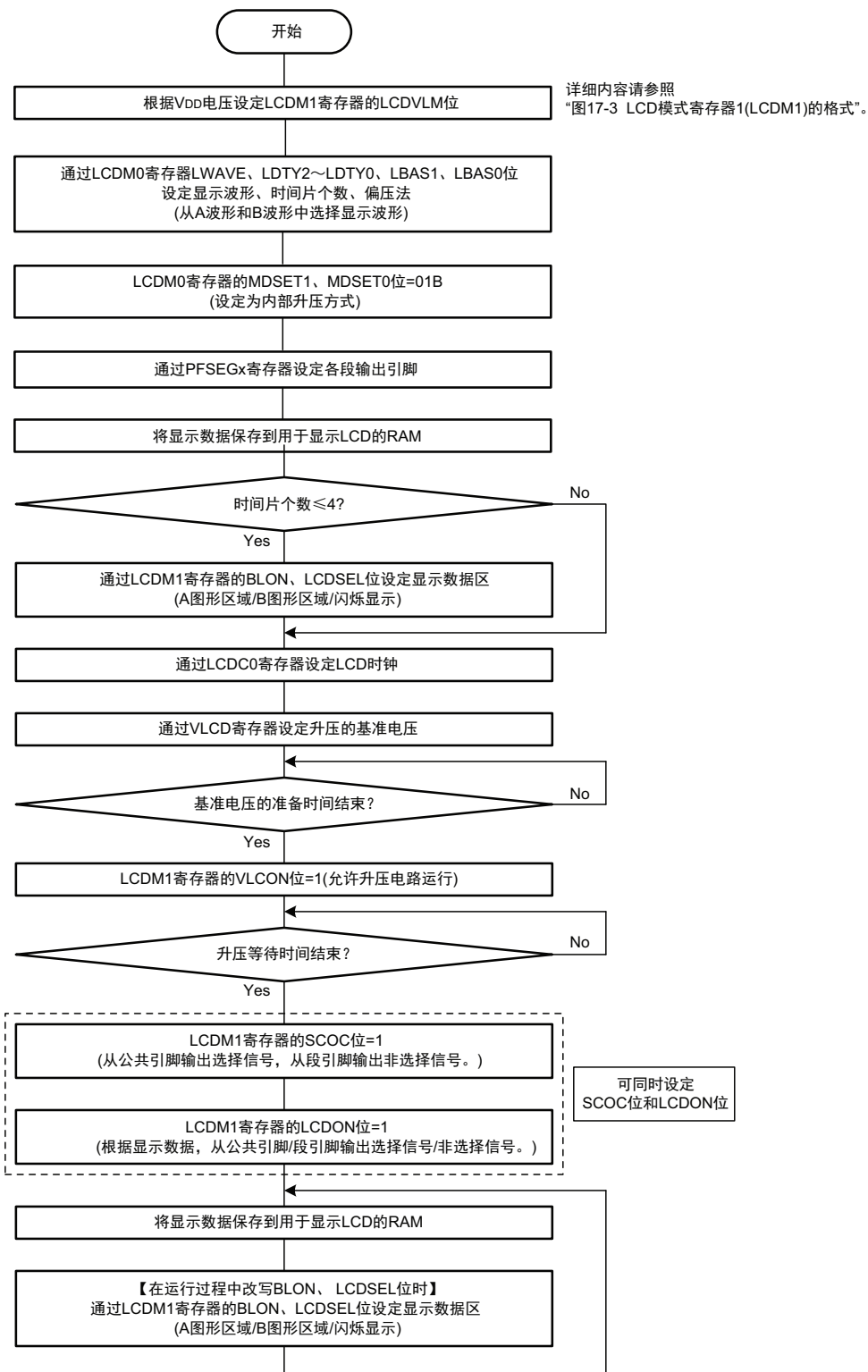
(1) 外部电阻分压方式

图 17-18 外部电阻分压方式的设定步骤



(2) 内部升压方式

图 17-19 内部升压方式的设定步骤



注意 1. 即使未更改 VLCD 寄存器，也要等待直到准备时间结束。

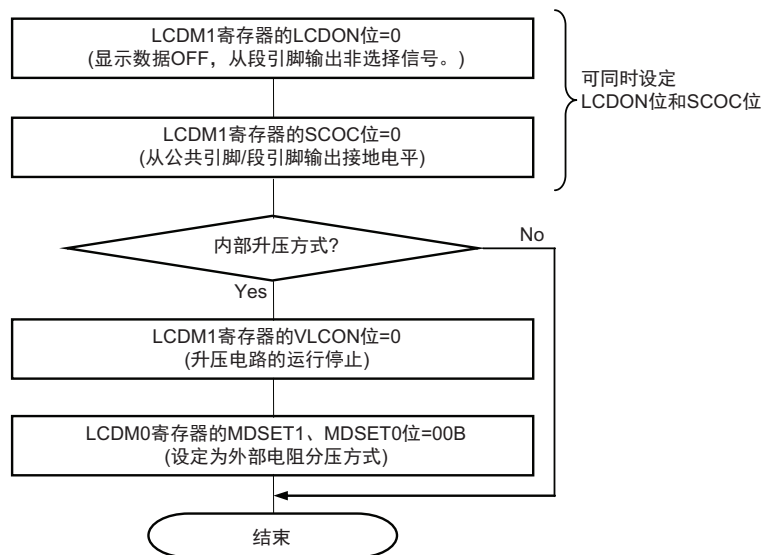
2. 有关基准电压准备时间和升压等待时间的规格，请参照“第 32 章 电特性”。

17.7 运行停止步骤

如果要在 LCD 显示屏正在显示时停止运行，就必须通过以下流程图的设定进行停止。

通过将 LCDM1 寄存器的 LCDON 位置“0”并且将 LCDM1 寄存器的 SCOC 位置“0”，停止 LCD 的运行。

图 17-20 运行停止步骤



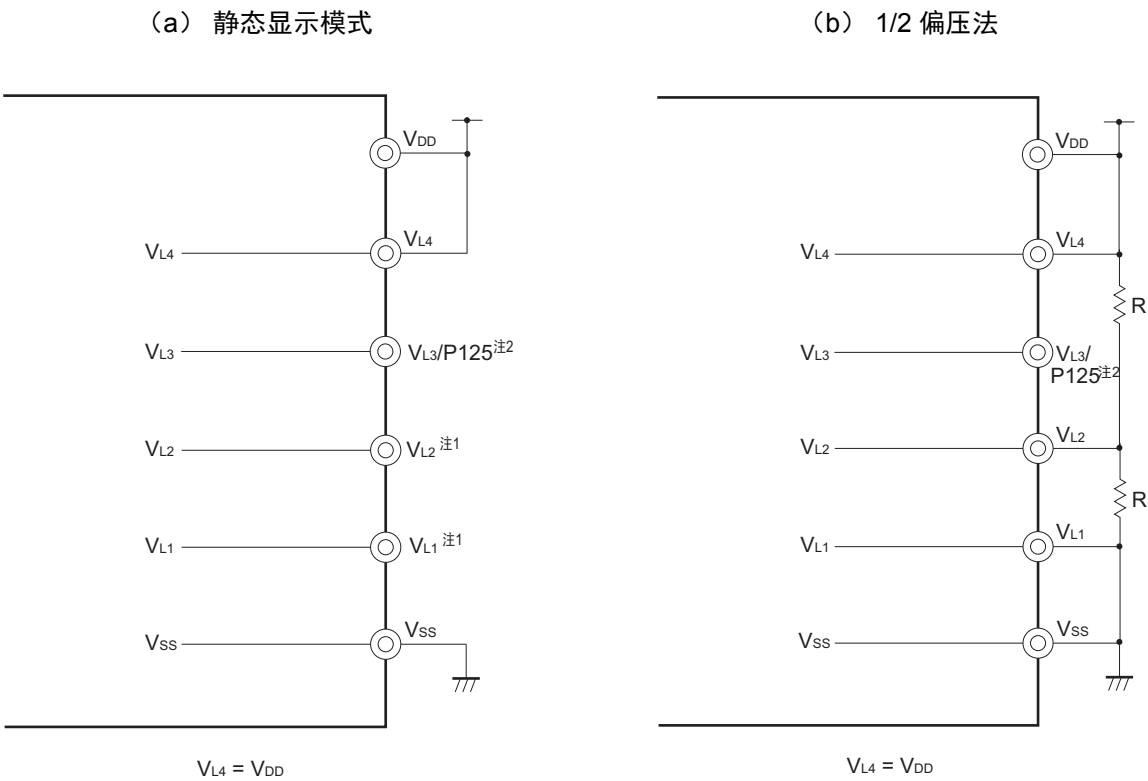
注意 在显示为 ON 状态（LCDM1 寄存器的 SCOC、LCDON 位 =11B）的情况下，禁止停止升压电路（LCDM1 寄存器的 VLCON 位 =0），否则不保证设定后的运行。必须在将显示置为 OFF（LCDM1 寄存器的 SCOC、LCDON 位 =10B）后停止升压电路（LCDM1 寄存器的 VLCON 位 =0）。

17.8 LCD 驱动电压 V_{L1} 、 V_{L2} 、 V_{L3} 和 V_{L4} 的提供

17.8.1 外部电阻分压方式

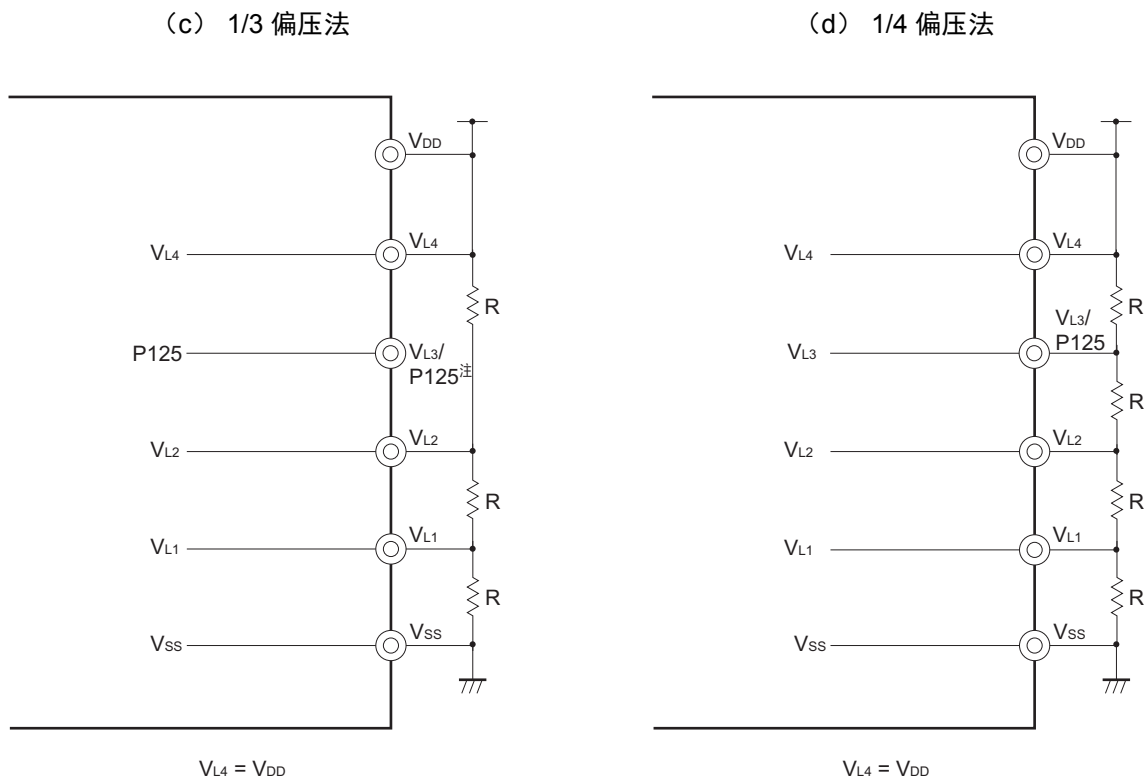
根据各偏压法进行 LCD 驱动电压连接的例子如图 17-21 所示。

图 17-21 LCD 驱动电源的连接例子（外部电阻分压方式）（1/2）



- 注 1. 将 V_{L1} 和 V_{L2} 接地或者置为开路。
2. V_{L3} 可用作端口（P125）。

图 17-21 LCD 驱动电源的连接例子（外部电阻分压方式）（2/2）



注 V_{L3} 可用作端口（P125）。

注意 用于外部电阻分解的电阻 R 的参考值为 $10\text{k}\Omega \sim 1\text{M}\Omega$ 。当要稳定 $V_{L1} \sim V_{L4}$ 引脚的电压时，必须根据需要，在 $V_{L1} \sim V_{L4}$ 引脚和 GND 之间连接电容器，电容器的参考值约为 $0.22\mu\text{F}$ ，取决于使用的 LCD 显示屏、段引脚数、公共引脚数、帧频和使用环境。必须在根据系统进行充分评估的基础上调整和决定电容值。

17.8.2 内部升压方式

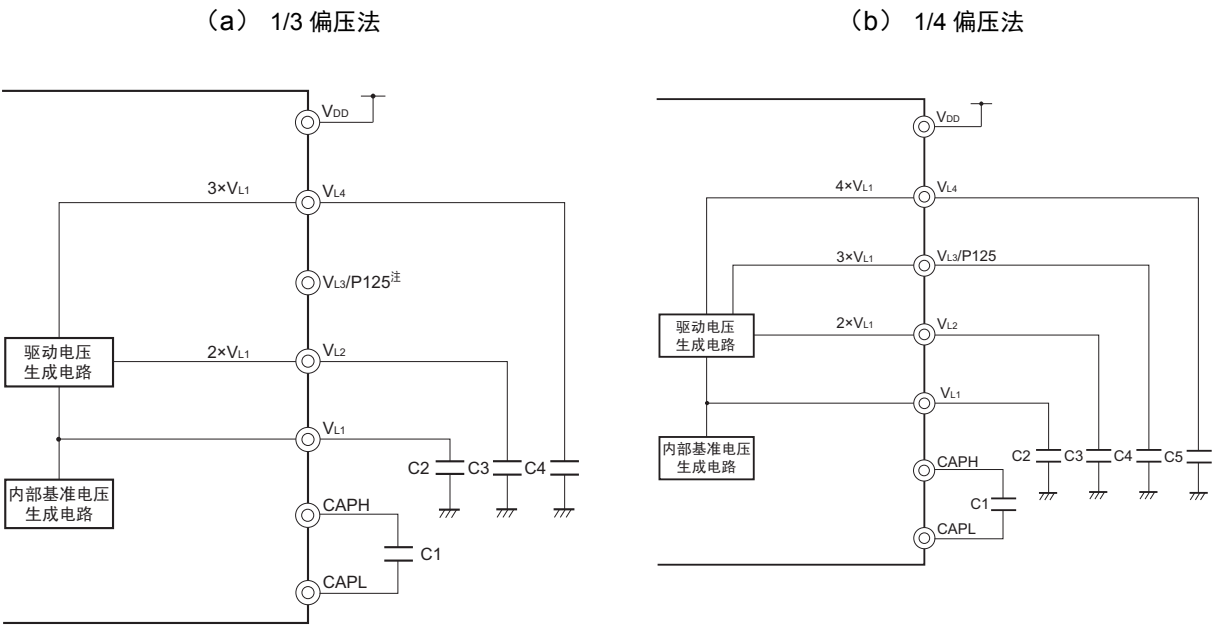
R7F0C003、R7F0C004 内置用于 LCD 驱动电源的内部升压电路。通过外接内部升压电路的电容器 (0.47μF±30%)，生成 LCD 驱动电压。内部升压方式只能使用 1/3 偏压法或者 1/4 偏压法。

内部升压方式的 LCD 驱动电压和主机本身不是同一个电源，因此与 V_{DD} 的变化无关，能提供固定的电压。能通过设定 LCD 升压控制寄存器 (VLCD) 来调整对比度。

表 17-12 LCD 驱动电压 (内部升压方式)

显示模式 驱动 LCD 的电源引脚	1/3 偏压方式	1/4 偏压方式
V _{L4}	3 × V _{L1}	4 × V _{L1}
V _{L3}	—	3 × V _{L1}
V _{L2}	2 × V _{L1}	2 × V _{L1}
V _{L1}	LCD 基准电压	LCD 基准电压

图 17-22 LCD 驱动电源的连接例子 (内部升压方式)



注 VL₃ 可用作端口 (P125)。

备注 必须尽量使用漏电流小的电容器。
C1 必须是无极性的电容器。

17.9 公共信号和段信号

当各像素对应的公共信号和段信号的电位差高于一定电压（LCD 驱动电压 V_{LCD} ）时，LCD 显示屏的各像素就亮灯。如果电位差低于 V_{LCD} ，各像素就熄灯。

如果给公共信号和段信号外加 DC 电压，LCD 显示屏就会变坏，因此通过 AC 电压进行驱动。

(1) 公共信号

根据设定的时间片个数，表 17-13 所示的顺序为公共信号的选择时序，并且以其为一个周期进行重复运行。在静态模式的情况下，COM0 ~ COM3 输出相同的信号。

必须将 2 个时间片时的 COM2 引脚和 COM3 引脚以及 3 个时间片时的 COM3 引脚置为开路。

必须将不是 6 个时间片或者 8 个时间片时的 COM4 ~ COM7 引脚和 6 个时间片时的 COM6、COM7 引脚置为开路或者用作段引脚。

表 17-13 COM 信号

COM信号 时间片	COM0	COM1	COM2	COM3	COM4	COM5	COM6	COM7
静态					注	注	注	注
2个时间片			开路	开路	注	注	注	注
3个时间片				开路	注	注	注	注
4个时间片					注	注	注	注
6个时间片							注	注
8个时间片								

注 必须置为开路或者用作段引脚。

(2) 段信号

段信号对应 LCD 显示数据寄存器（参照“17.4 LCD 显示数据寄存器”）。

在 8 个时间片方式下，各显示数据寄存器的 bit0 ~ bit7 对应 COM0 ~ COM7。与公共信号输出的各时序同步，读数据存储器的数据。如果各位的内容为“1”，就在转换为选择电压后输出到段引脚（SEG4 ~ SEG50）。如果各位的内容为“0”，就在转换为非选择电压后输出到段引脚（SEG4 ~ SEG50）。

在 6 个时间片方式下，各显示数据寄存器的 bit0 ~ bit5 对应 COM0 ~ COM5。与公共信号输出的各时序同步，读数据存储器的数据。如果各位的内容为“1”，就在转换为选择电压后输出到段引脚（SEG2 ~ SEG50）。如果各位的内容为“0”，就在转换为非选择电压后输出到段引脚（SEG2 ~ SEG50）。

在不是 6 个时间片和 8 个时间片方式下，在 A 图形区中各显示数据寄存器的 bit0 ~ bit3 对应 COM0 ~ COM3，在 B 图形区中各显示数据寄存器的 bit4 ~ bit7 对应 COM0 ~ COM3。与公共信号输出的各时序同步，读数据存储器的数据。如果各位的内容为“1”，就在转换为选择电压后输出到段引脚（SEG0 ~ SEG50）。如果各位的内容为“0”，就在转换为非选择电压后输出到段引脚（SEG0 ~ SEG50）。

因此，必须先确认使用的 LCD 显示屏的前面电极（对应段信号）和背面电极（对应公共信号）是如何组合形成显示图形的，然后给 LCD 数据寄存器写与想要显示的图形一一对应的位数据。

(3) 公共信号和段信号的输出波形

公共信号和段信号输出表 17-14 所示的电压。

只有在公共信号和段信号都为选择电压时才为 $\pm V_{LCD}$ 的亮灯电压，在其他组合时为熄灯电压。

表 17-14 LCD 驱动电压

(a) 静态显示模式

公共信号 \ 段信号		选择信号电平	非选择信号电平
		V_{SS}/V_{L4}	V_{L4}/V_{SS}
V_{L4}/V_{SS}		$-V_{LCD}/+V_{LCD}$	0V/0V

(b) 1/2 偏压法

公共信号 \ 段信号		选择信号电平	非选择信号电平
		V_{SS}/V_{L4}	V_{L4}/V_{SS}
选择信号电平	V_{L4}/V_{SS}	$-V_{LCD}/+V_{LCD}$	0V/0V
非选择信号电平	V_{L2}	$-\frac{1}{2}V_{LCD}/+\frac{1}{2}V_{LCD}$	$+\frac{1}{2}V_{LCD}/-\frac{1}{2}V_{LCD}$

(c) 1/3 偏压法 (A 波形、B 波形)

公共信号 \ 段信号		选择信号电平	非选择信号电平
		V_{SS}/V_{L4}	V_{L2}/V_{L1}
选择信号电平	V_{L4}/V_{SS}	$-V_{LCD}/+V_{LCD}$	$-\frac{1}{3}V_{LCD}/+\frac{1}{3}V_{LCD}$
非选择信号电平	V_{L4}/V_{L2}	$-\frac{1}{3}V_{LCD}/+\frac{1}{3}V_{LCD}$	$+\frac{1}{3}V_{LCD}/-\frac{1}{3}V_{LCD}$

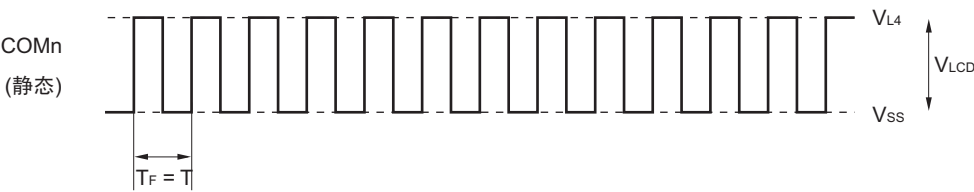
(d) 1/4 偏压法 (A 波形、B 波形)

公共信号 \ 段信号		选择信号电平	非选择信号电平
		V_{SS}/V_{L4}	V_{L2}
选择信号电平	V_{L4}/V_{SS}	$-V_{LCD}/+V_{LCD}$	$-\frac{1}{2}V_{LCD}/+\frac{1}{2}V_{LCD}$
非选择信号电平	V_{L1}/V_{L3}	$-\frac{1}{4}V_{LCD}/+\frac{1}{4}V_{LCD}$	$+\frac{1}{4}V_{LCD}/-\frac{1}{4}V_{LCD}$

公共信号波形如图 17-23 所示，公共信号和段信号的电压与相位如图 17-24 所示。

图 17-23 公共信号波形 (1/2)

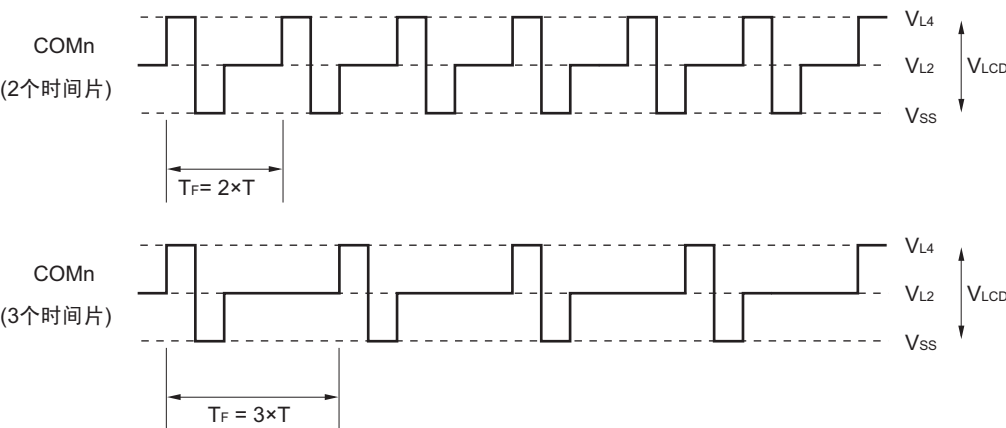
(a) 静态显示模式



T: 1 个 LCD 时钟周期

T_F: 帧频率

(b) 1/2 偏压法

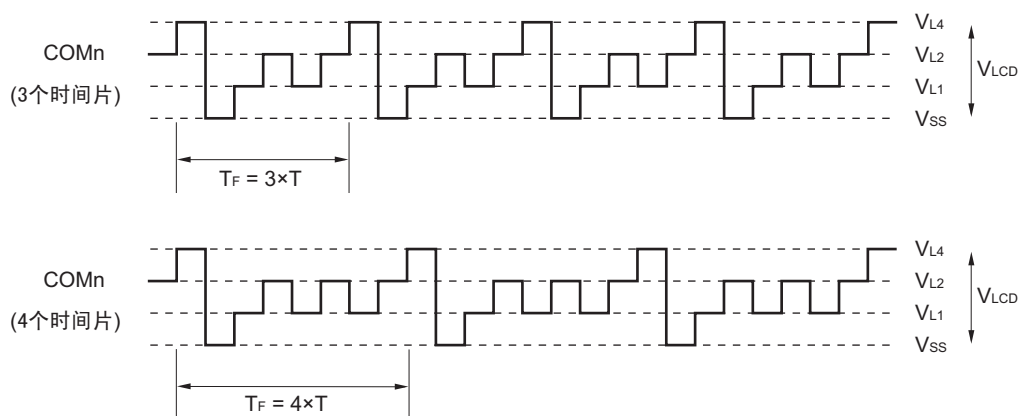


T: 1 个 LCD 时钟周期

T_F: 帧频率

图 17-23 公共信号波形 (2/2)

(c) 1/3 偏压法



T: 1 个 LCD 时钟周期

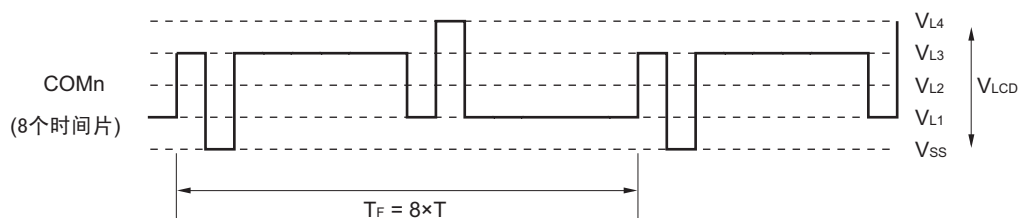
 T_F : 帧频率

< LCD 帧频率的计数例子 (使用 4 个时间片的情况) >

LCD 时钟: $32768/2^7 = 256\text{Hz}$ (设定为 $\text{LCDC0} = 06\text{H}$ 时)

LCD 帧频率: 64Hz

(d) 1/4 偏压法



T: 1 个 LCD 时钟周期

 T_F : 帧频率

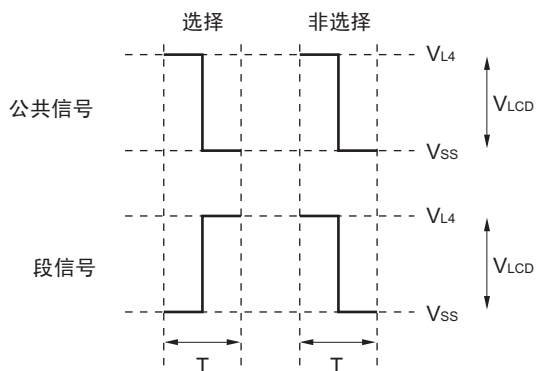
< LCD 帧频率的计算例子 (使用 8 个时间片的情况) >

LCD 时钟: $32768/2^7 = 256\text{Hz}$ (设定为 $\text{LCDC0} = 06\text{H}$ 时)

LCD 帧频率: 32Hz

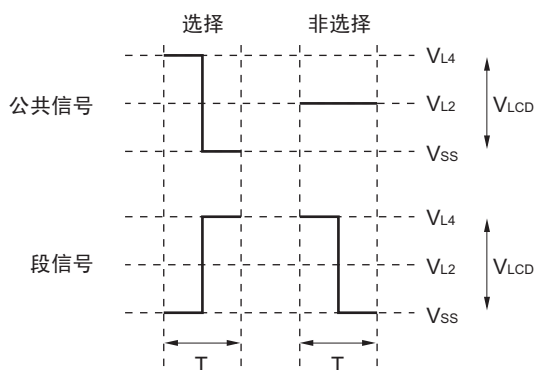
图 17-24 公共信号和段信号的电压与相位 (1/2)

(a) 静态显示模式 (A 波形)



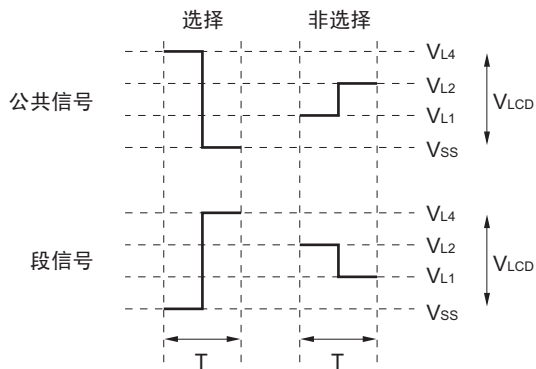
T: 1 个 LCD 时钟周期

(b) 1/2 偏压法 (A 波形)



T: 1 个 LCD 时钟周期

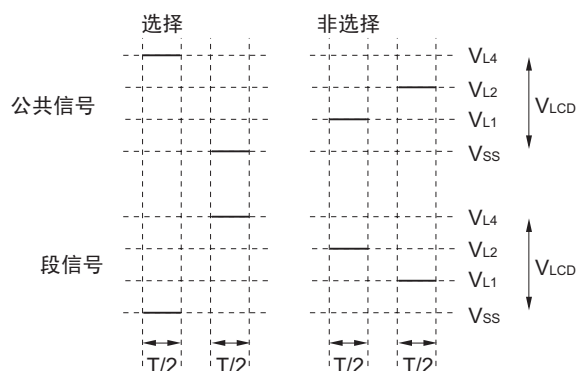
(c) 1/3 偏压法 (A 波形)



T: 1 个 LCD 时钟周期

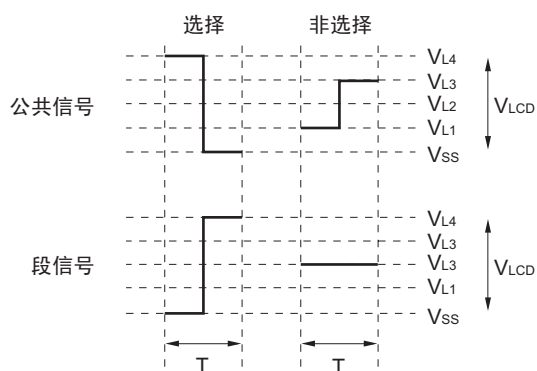
图 17-24 公共信号和段信号的电压与相位 (2/2)

(d) 1/3 偏置方式 (B 波形)



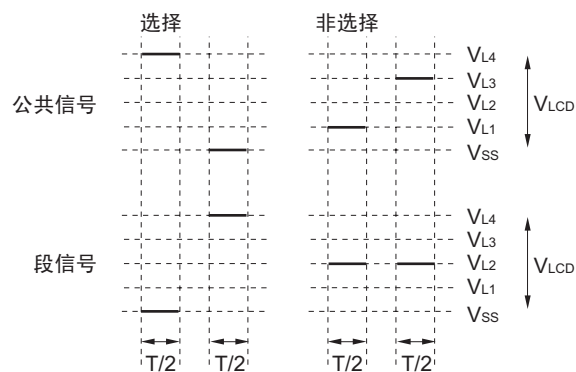
T: 1 个 LCD 时钟周期

(e) 1/4 偏压法 (A 波形)



T: 1 个 LCD 时钟周期

(f) 1/4 偏压法 (B 波形)



T: 1 个 LCD 时钟周期

17.10 显示模式

17.10.1 静态显示例子

图 17-25 显示图形的静态方式的 3 位 LCD 显示屏、段信号（SEG0 ~ SEG23）和公共信号（COM0）的连接如图 17-26 所示。显示例子是“12.3”，显示数据寄存器（F0400H ~ F0417H）的内容对应此显示例子。

在此，以第 2 位的“2.”（2.）为例进行说明。需要根据图 17-25 的显示图形并且通过 COM0 的公共信号的时序，将表 17-15 所示的选择电压和非选择电压输出到 SEG8 ~ SEG15 引脚。

表 17-15 选择和非选择电压（COM0）

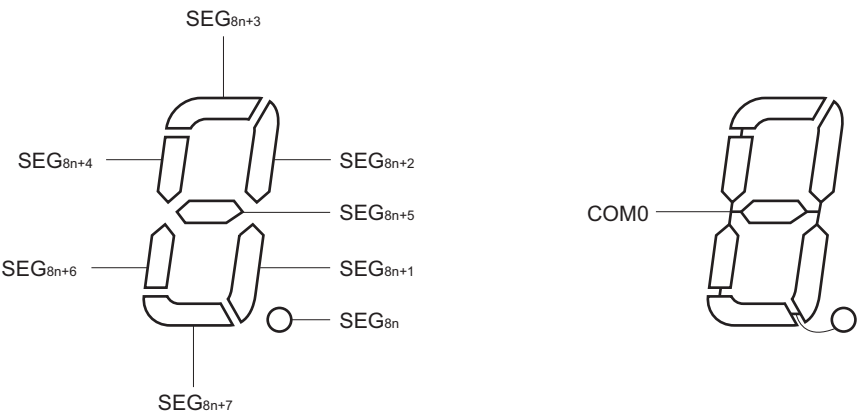
段信号 公共信号	SEG8	SEG9	SEG10	SEG11	SEG12	SEG13	SEG14	SEG15
COM0	选择	非选择	选择	选择	非选择	选择	选择	选择

因此，给 SEG8 ~ SEG15 对应的显示数据寄存器（F0408H ~ F040FH）的 bit0 准备“10110111”即可。

SEG11、SEG12 和 COM0 的 LCD 驱动波形如图 17-27 所示。如果在选择 COM0 时 SEG11 为选择电压，就知道产生 LCD 亮灯电平 $+V_{LCD}/-V_{LCD}$ 的交流矩形波。

因为 COM1 ~ COM3 输出和 COM0 相同的波形，所以能通过连接 COM0 ~ COM3 来提高驱动能力。

图 17-25 静态 LCD 显示图形和电极连线



备注 n = 0 ~ 5

图 17-26 静态 LCD 面板的连接例子

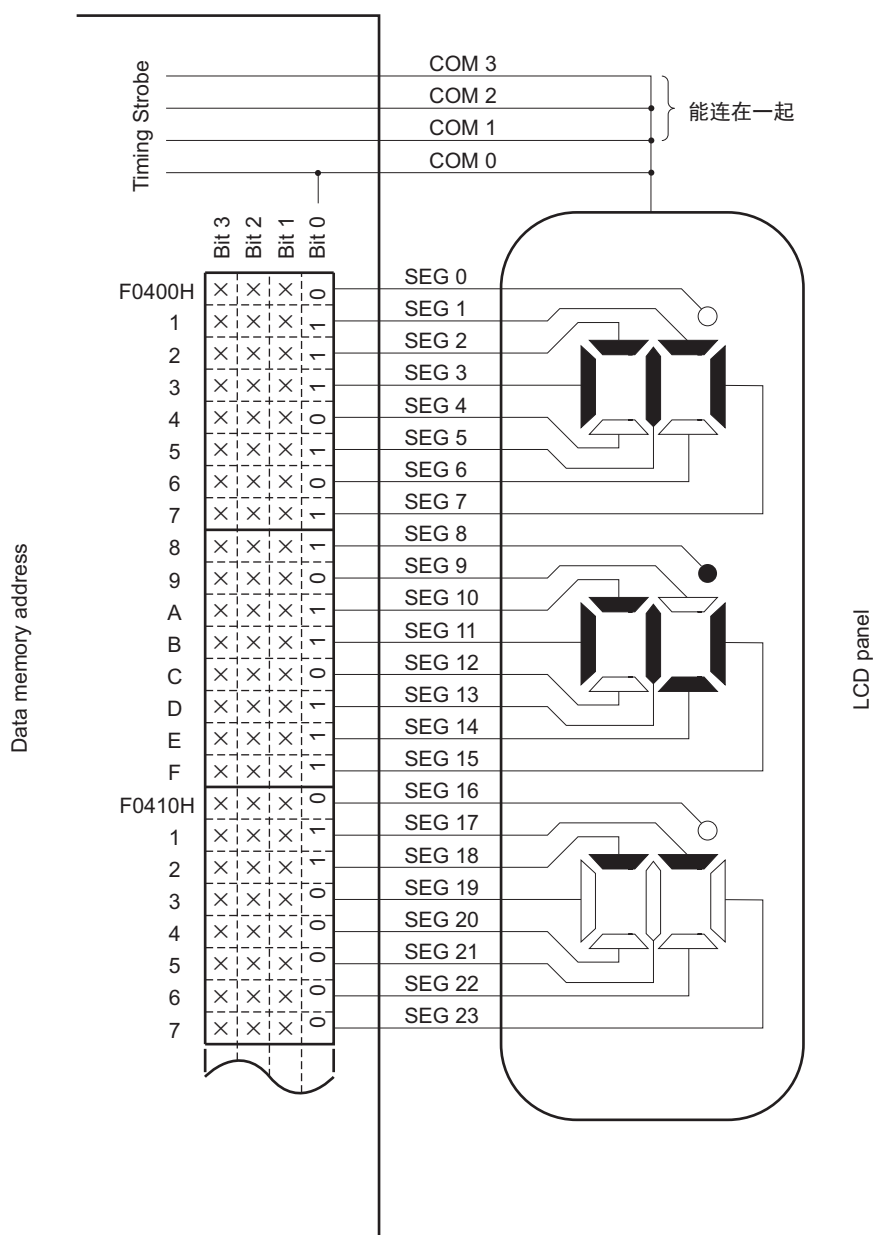
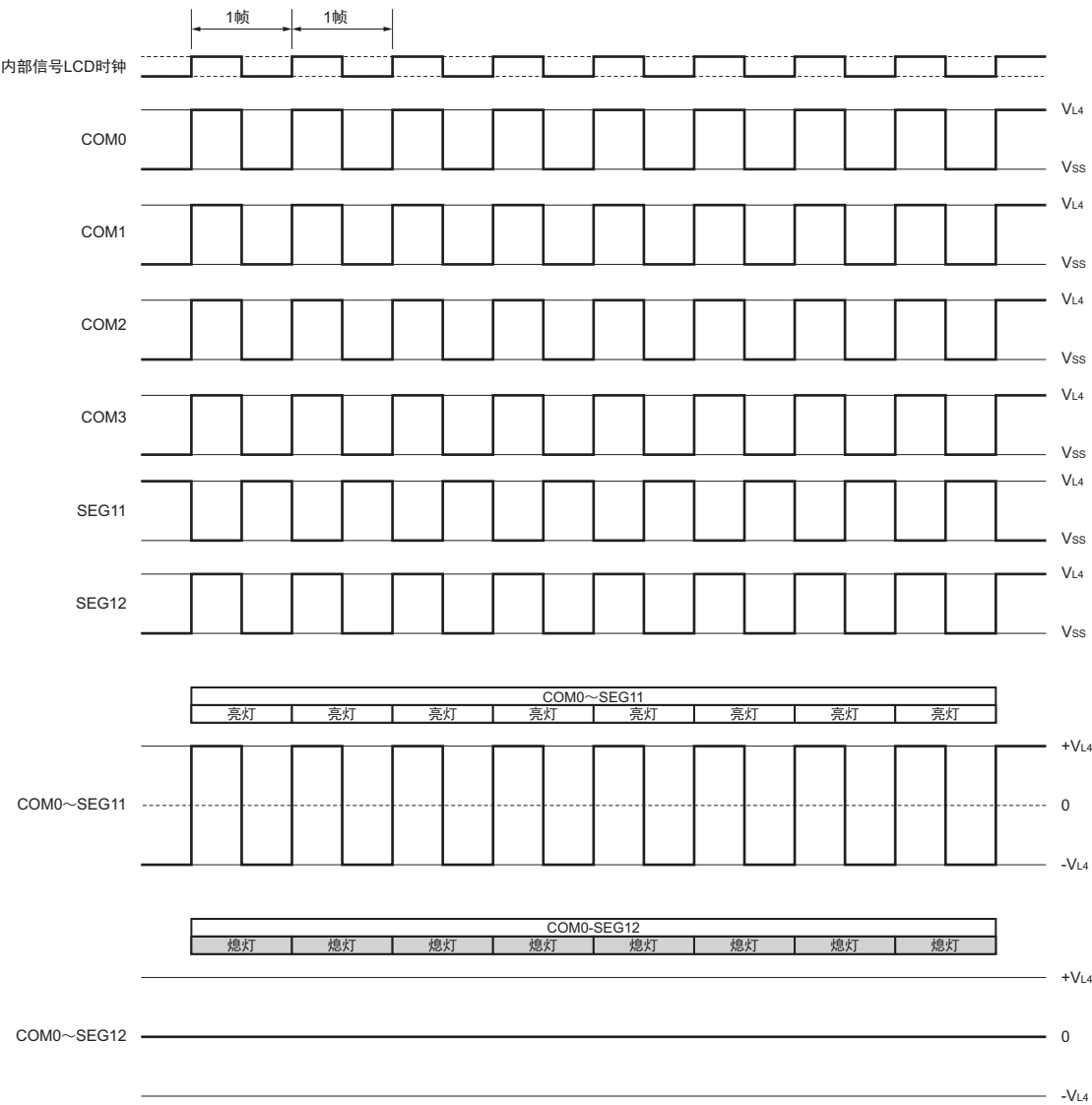


图 17-27 SEG11、SEG12 和 COM0 的静态 LCD 驱动波形例子



17.10.2 2 个时间片的显示例子

图 17-28 显示图形的 2 个时间片方式的 6 位 LCD 显示屏、段信号（SEG0 ～ SEG23）和公共信号（COM0、COM1）的连接如图 17-29 所示。显示例子是“12345.6”，显示数据寄存器（F0400H ～ F0417H）的内容对应此显示例子。

在此，以第 4 位的“3”（）为例进行说明。需要根据图 17-28 的显示图形并且通过 COM0 和 COM1 的公共信号的时序，将表 17-16 所示的选择和非选择电压输出到 SEG12 ～ SEG15 引脚。

表 17-16 选择和非选择电压（COM0 和 COM1）

段信号 公共信号	SEG12	SEG13	SEG14	SEG15
COM0	选择	选择	非选择	非选择
COM1	非选择	选择	选择	选择

因此，给 SEG15 对应的显示数据寄存器（地址 F040FH）准备“xx10”即可。

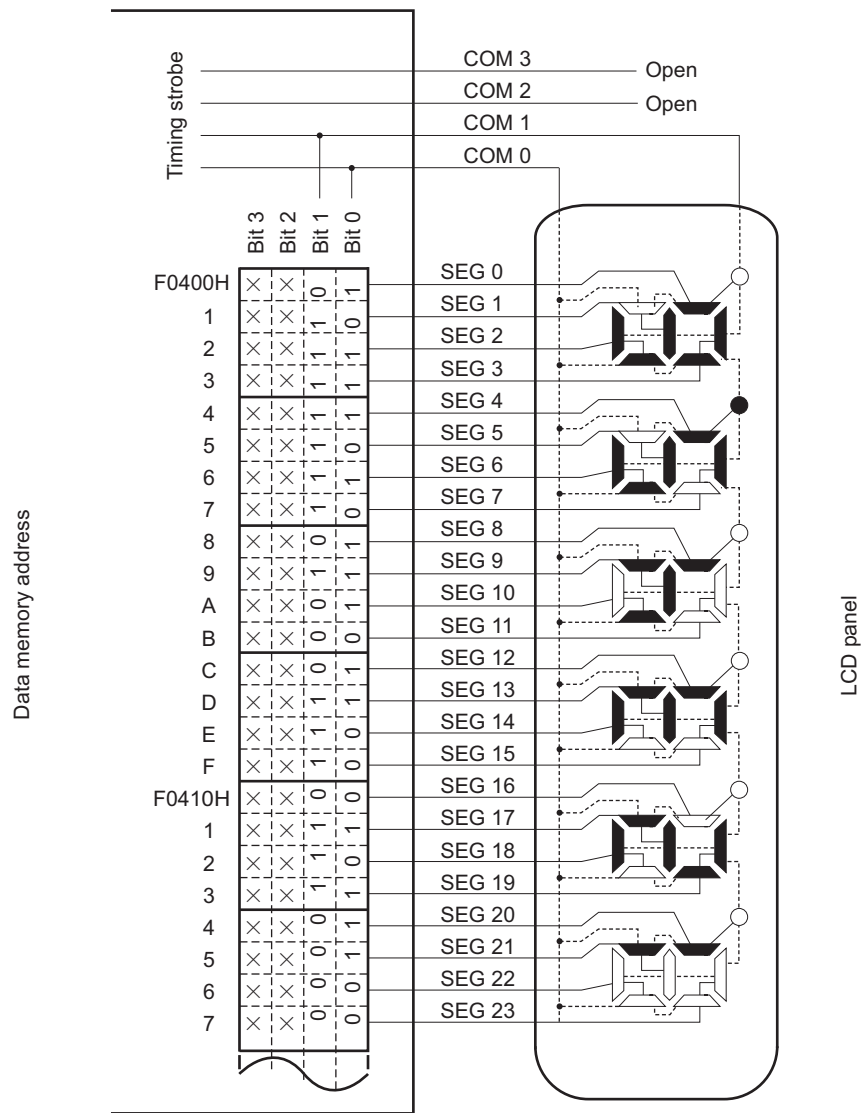
SEG15 和各公共信号之间的 LCD 驱动波形例子如图 17-30 所示。在选择 COM1 时 SEG15 为选择电压，就知道产生 LCD 亮灯电平 $+V_{LCD}/-V_{LCD}$ 的交流矩形波。

图 17-28 2 个时片的 LCD 显示图形和电极连线



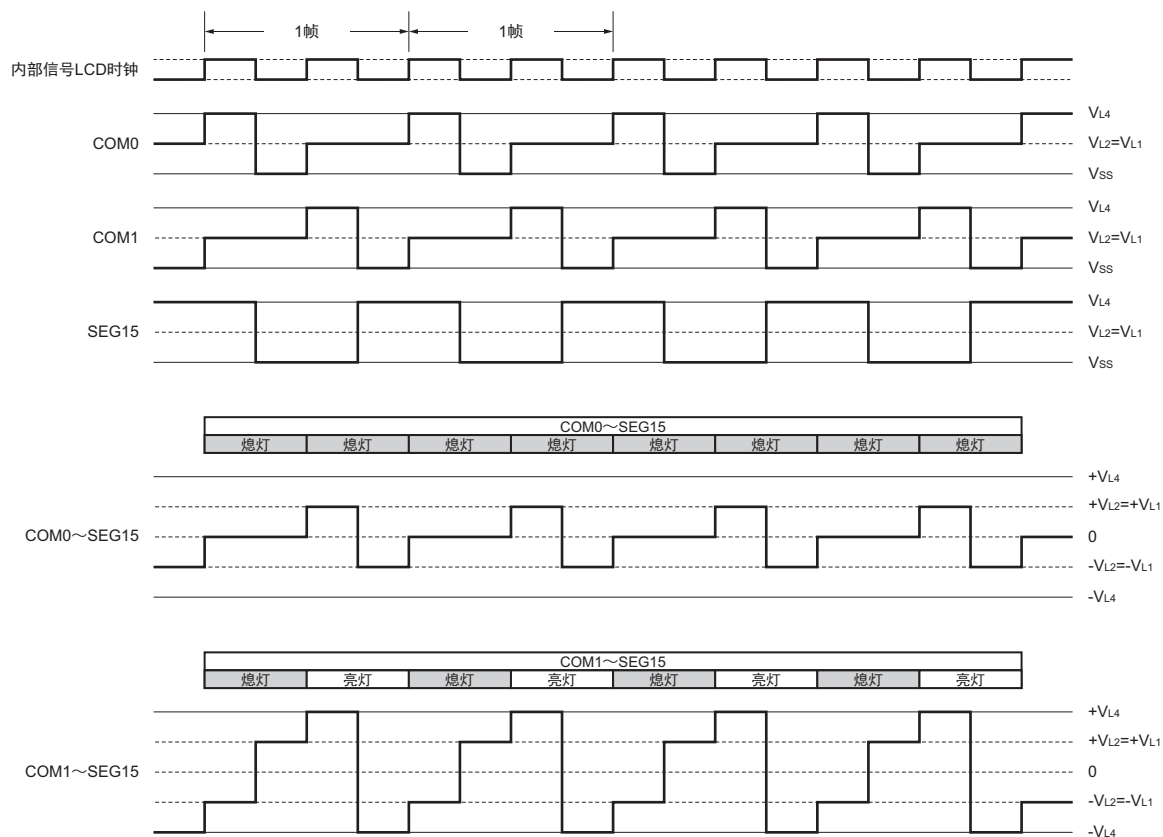
备注 n = 0 ～ 12

图 17-29 2 个时间片的 LCD 显示屏的连接例子



×: 因为是 2 个时间片显示, 所以可随时保存任意的数据。

图 17-30 SEG15 和各公共信号之间的 2 个时间片的 LCD 驱动波形例子 (1/2 偏压法)



17.10.3 3 个时间片的显示例子

图 17-31 显示图形的 3 个时间片方式的 8 位 LCD 显示屏、段信号（SEG0 ~ SEG23）和公共信号（COM0 ~ COM2）的连接如图 17-32 所示。显示例子是“123456.78”，显示数据寄存器（F0400H ~ F0417H）的内容对应此显示例子。

在此，以第 3 位的“6.”（) 为例进行说明。需要根据图 17-31 的显示图形并且通过 COM0 ~ COM2 的各公共信号的时序，将表 17-17 所示的选择电压和非选择电压输出到 SEG6 ~ SEG8 引脚。

表 17-17 选择和非选择电压（COM0 ~ COM2）

公共信号 \ 段信号	SEG6	SEG7	SEG8
COM0	非选择	选择	选择
COM1	选择	选择	选择
COM2	选择	选择	—

因此，给 SEG6 对应的显示数据寄存器（地址 F0406H）准备“x110”即可。

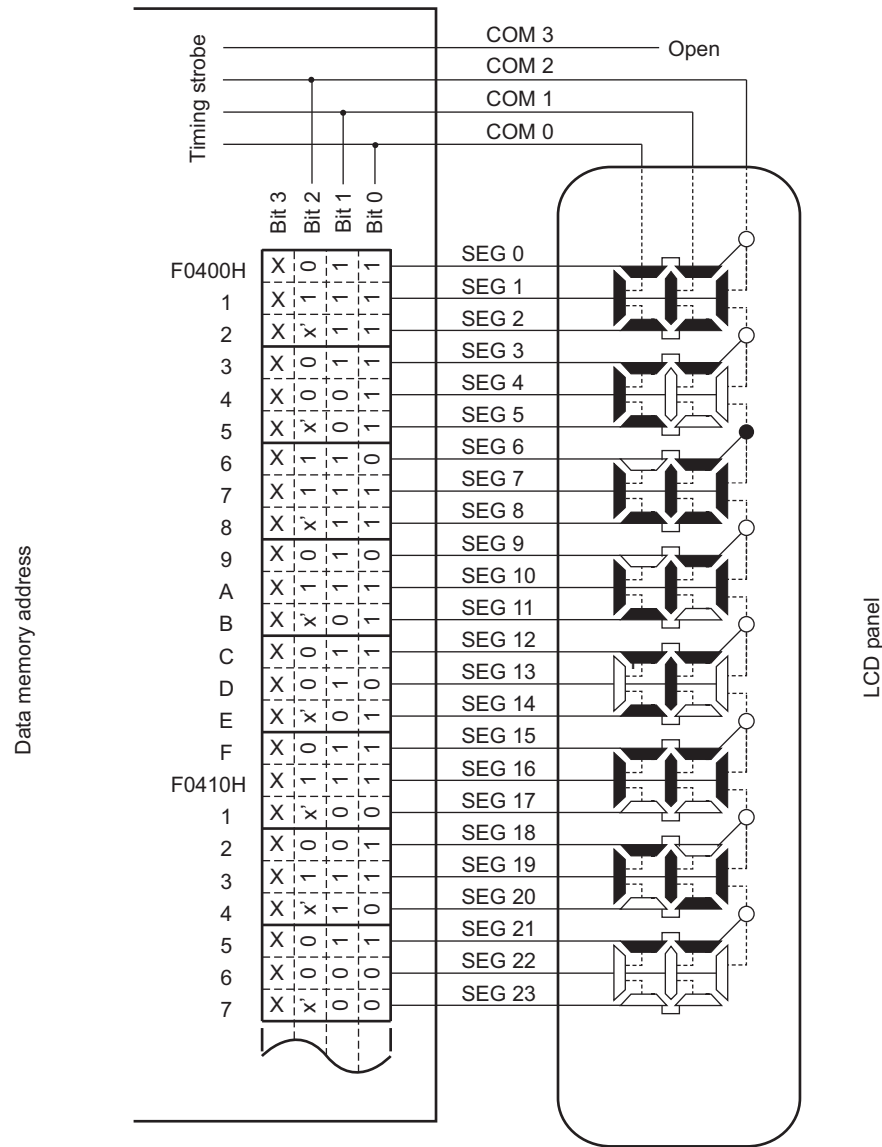
SEG6 和各公共信号之间的 LCD 驱动波形例子如图 17-33（1/2 偏压法）和图 17-34（1/3 偏压法）所示。在选择 COM1 或者 COM2 时 SEG6 为选择电压，就知道产生 LCD 亮灯电平 +V_{LCD}/-V_{LCD} 的交流矩形波。

图 17-31 3 个时间片的 LCD 显示图形和电极连线



备注 n = 0 ~ 16

图 17-32 3 个时间片的 LCD 显示屏的连接例子



×': 因为 LCD 显示屏中没有对应的段，所以可保存任意的数据。

×: 因为是 3 个时间片显示，所以可随时保存任意的数据。

图 17-33 SEG6 和各公共信号之间的 3 个时间片的 LCD 驱动波形例子 (1/2 偏压法)

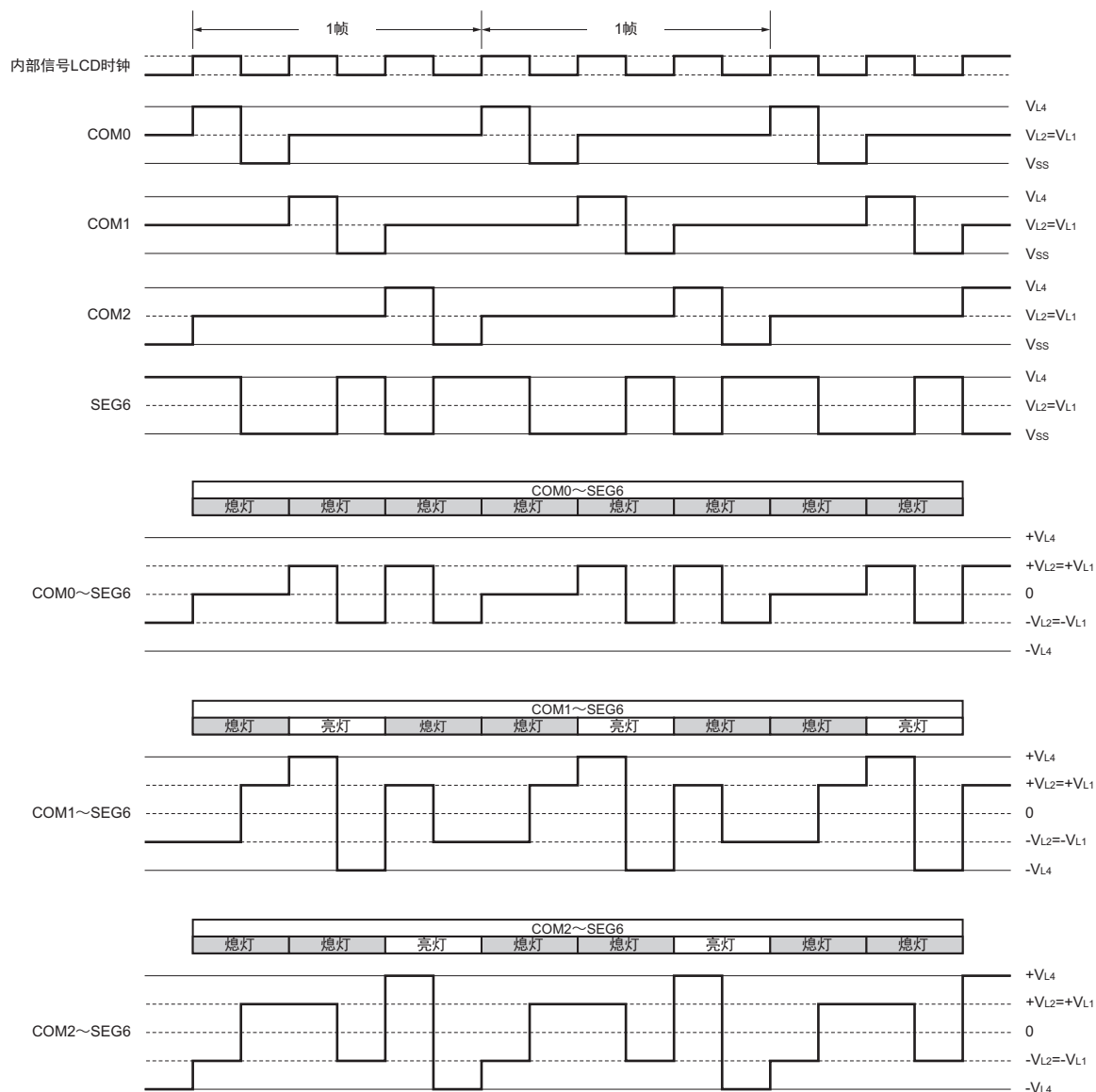
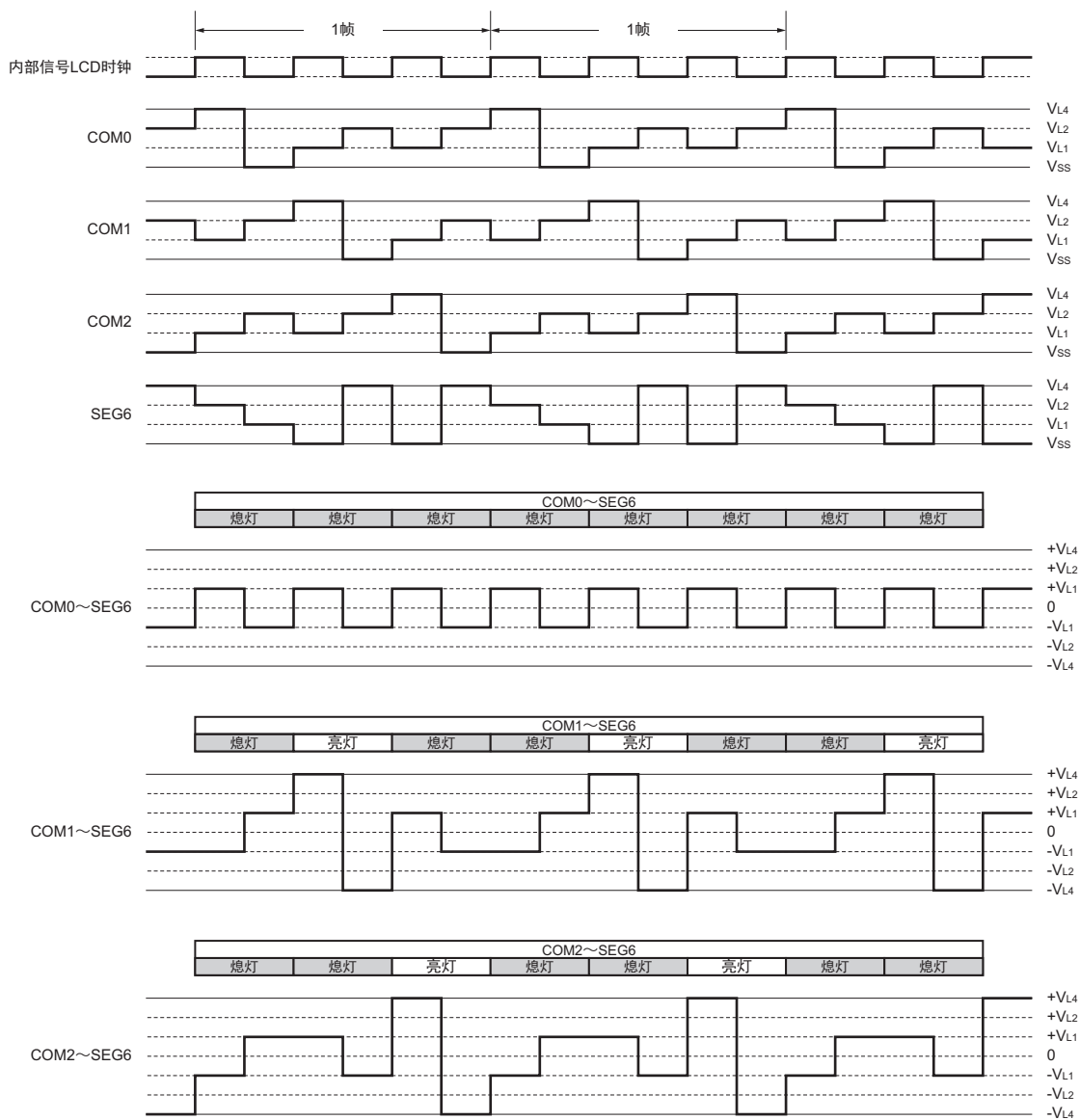


图 17-34 SEG6 和各公共信号之间的 3 个时间片的 LCD 驱动波形例子 (1/3 偏压法)



17.10.4 4 个时间片的显示例子

图 17-35 显示图形的 4 个时间片方式的 12 位 LCD 显示屏、段信号（SEG0 ～ SEG23）和公共信号（COM0 ～ COM3）的连接如图 17-36 所示。显示例子是“123456.789012”，显示数据寄存器（地址 F0400H ～ F0417H）的内容对应此显示例子。

在此，以第 7 位的“6.”（) 为例进行说明。需要根据图 17-35 的显示图形并且通过 COM0 ～ COM3 的各公共信号的时序，将表 17-18 所示的选择电压和非选择电压输出到 SEG12 引脚和 SEG13 引脚。

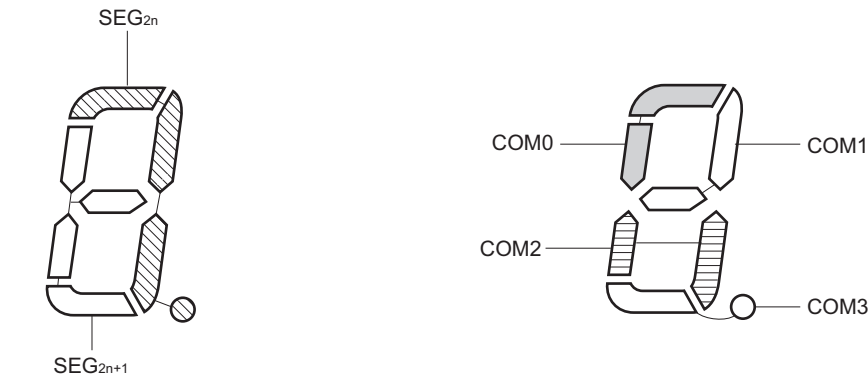
表 17-18 选择和取消选择电压（COM0 ～ COM3）

公共信号 \ 段信号	SEG12	SEG13
COM0	选择	选择
COM1	非选择	选择
COM2	选择	选择
COM3	选择	选择

因此，给 SEG12 对应的显示数据寄存器（地址 F040CH）准备“1101”即可。

SEG12 和各公共信号之间的 LCD 驱动波形例子如图 17-37 所示。在选择 COM0 时 SEG12 为选择电压，就知道产生 LCD 亮灯电平 $+V_{LCD}/-V_{LCD}$ 的交流矩形波。

图 17-35 4 个时间片的 LCD 显示图形和电极连线



备注 n = 0 ～ 25

图 17-36 4 个时间片的 LCD 显示屏的连接例子

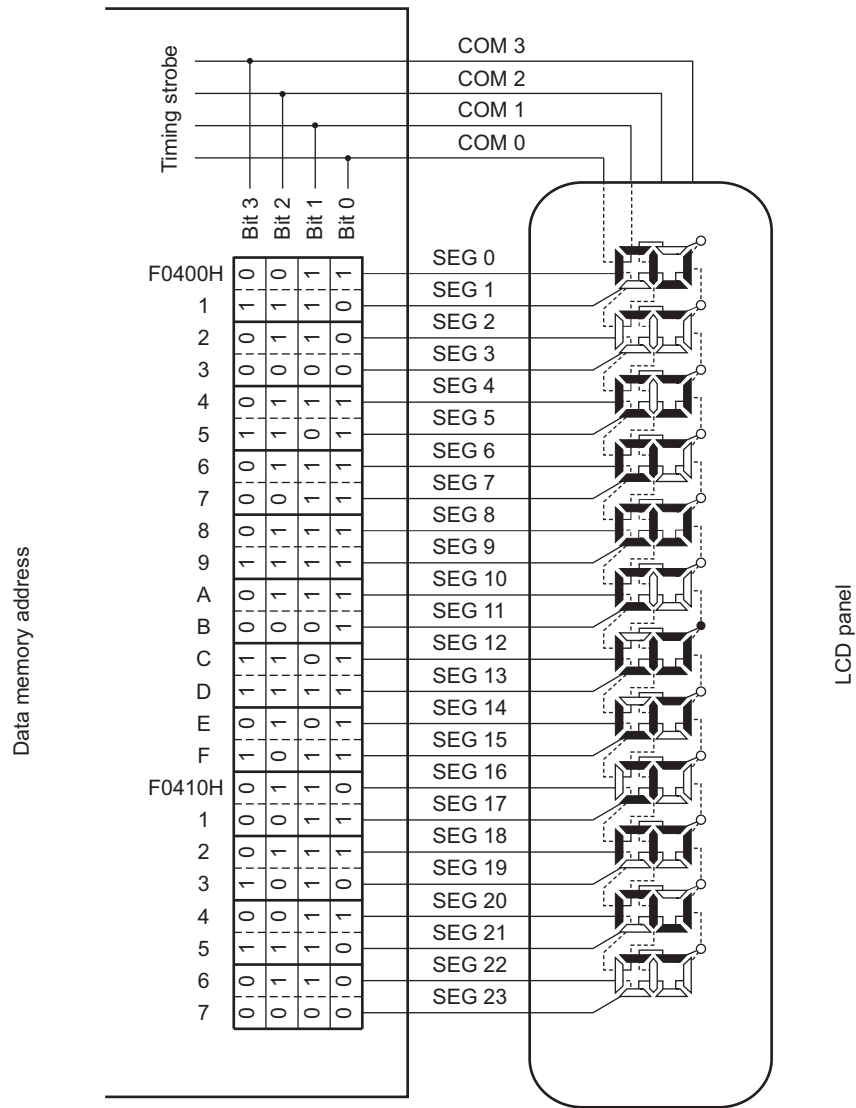


图 17-37 SEG12 和各公共信号之间的 4 个时间片的 LCD 驱动波形例子 (1/3 偏压法) (1/2)

(a) A 波形

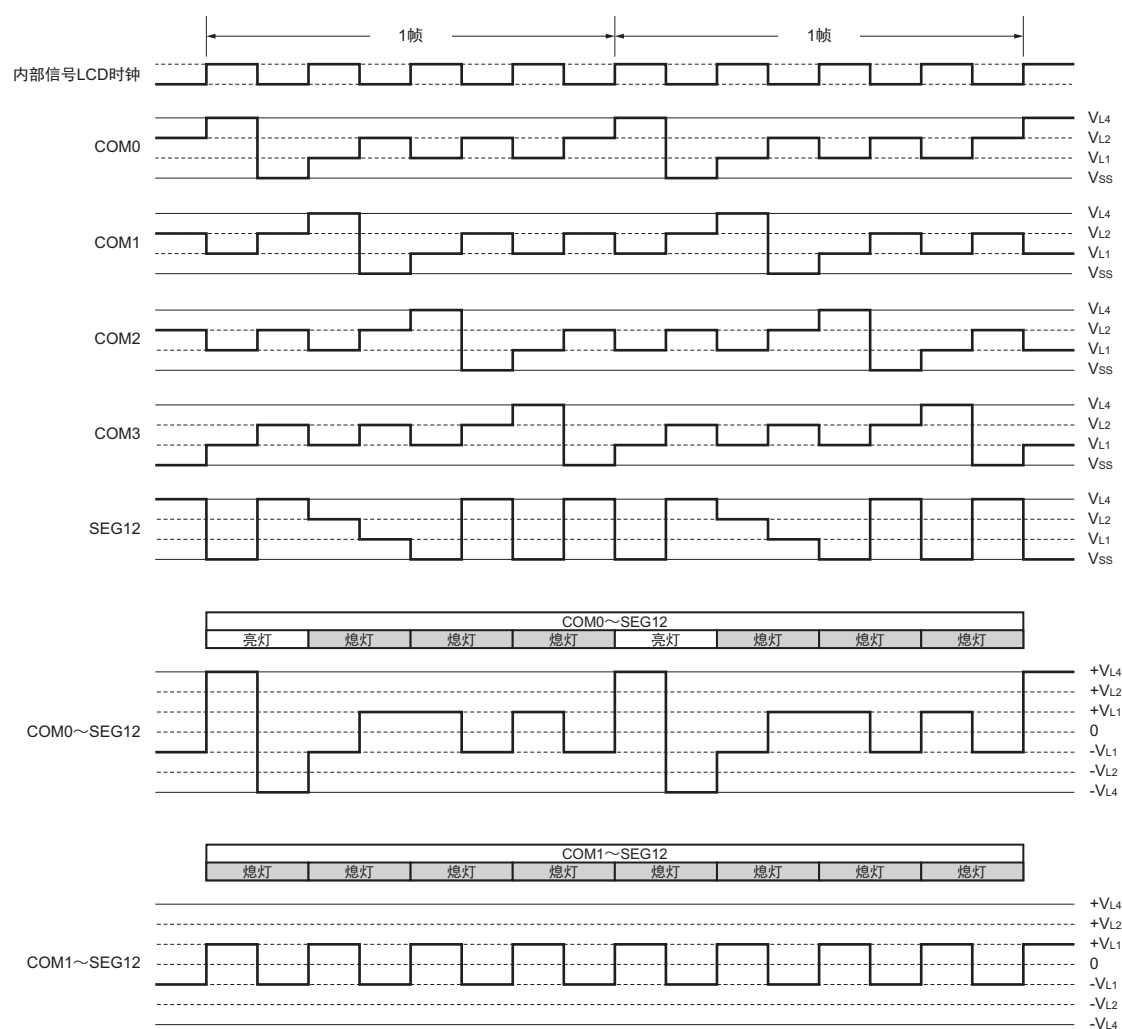
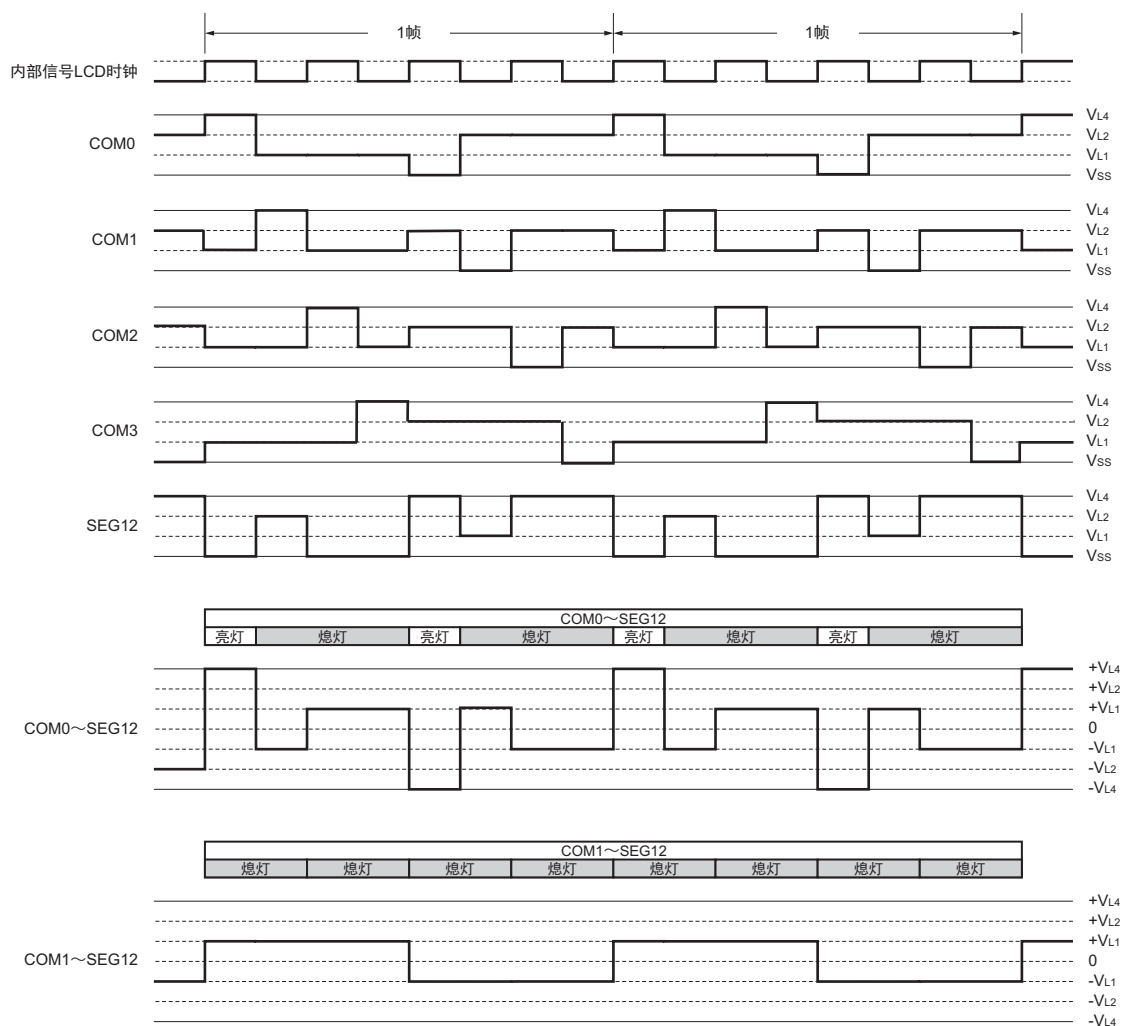


图 17-37 SEG12 和各公共信号之间的 4 个时间片的 LCD 驱动波形例子 (1/3 偏压法) (2/2)

(b) B 波形



17.10.5 6 个时间片的显示例子

图 17-38 显示图形的 6 个时间片方式的 15×6 点 LCD 显示屏、段信号（SEG2～SEG16）和公共信号（COM0～COM5）的连接如图 17-39 所示。显示例子是“123”，显示数据寄存器（地址 F0402H～F0410H）的内容对应此显示例子。

在此，以第 1 位的“3”（）为例进行说明。需要根据图 17-38 的显示图形并且通过 COM0～COM5 的各公共信号的时序，将表 17-19 所示的选择电压和非选择电压输出到 SEG2～SEG6 引脚。

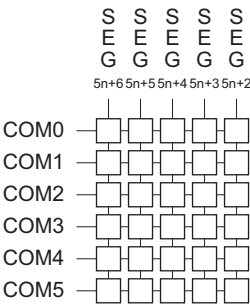
表 17-19 选择和非选择电压（COM0～COM5）

公共信号 \ 段信号	SEG2	SEG3	SEG4	SEG5	SEG6
COM0	选择	选择	选择	选择	选择
COM1	非选择	选择	非选择	非选择	非选择
COM2	非选择	非选择	选择	非选择	非选择
COM3	非选择	选择	非选择	非选择	非选择
COM4	选择	非选择	非选择	非选择	选择
COM5	非选择	选择	选择	选择	非选择

因此，给 SEG2 对应的显示数据寄存器（地址 F0402H）准备“010001”即可。

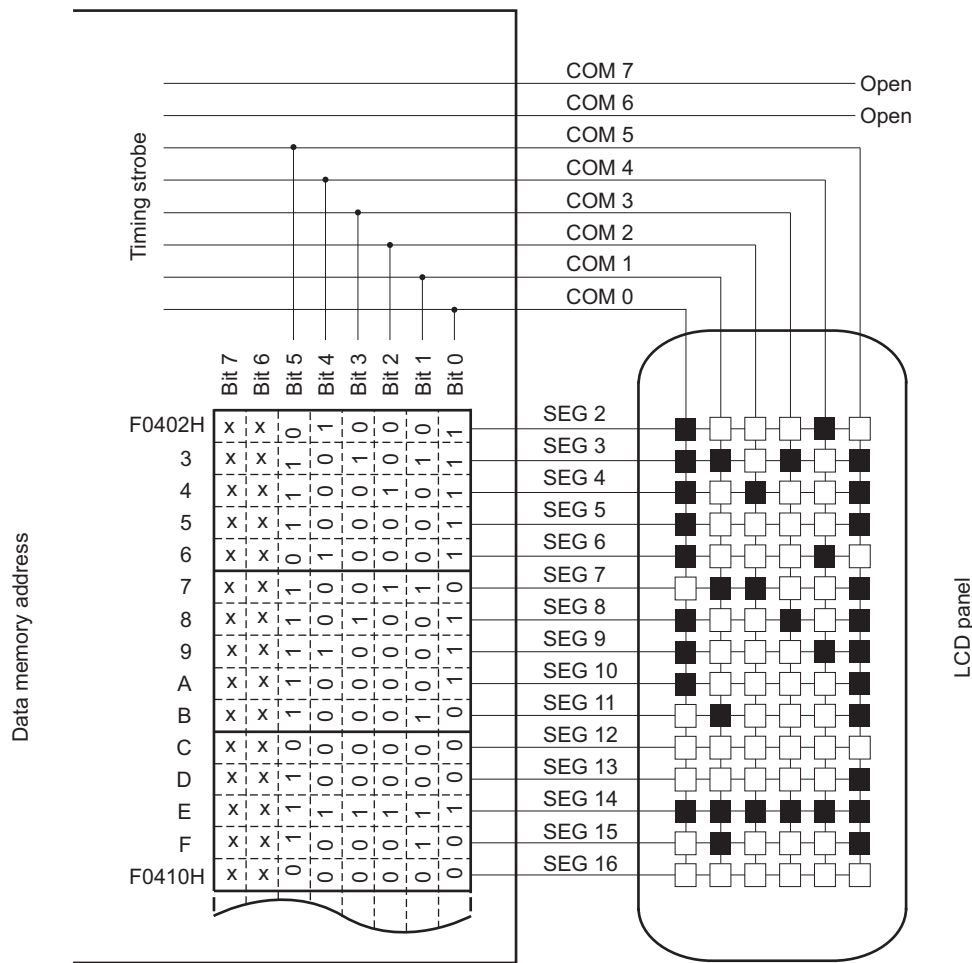
SEG2 和各公共信号之间的 LCD 驱动波形例子如图 17-40 所示。在选择 COM0 时 SEG2 为选择电压，就知道产生 LCD 亮灯电平的波形。

图 17-38 6 个时间片的 LCD 显示图形和电极连线



备注 n = 0～9

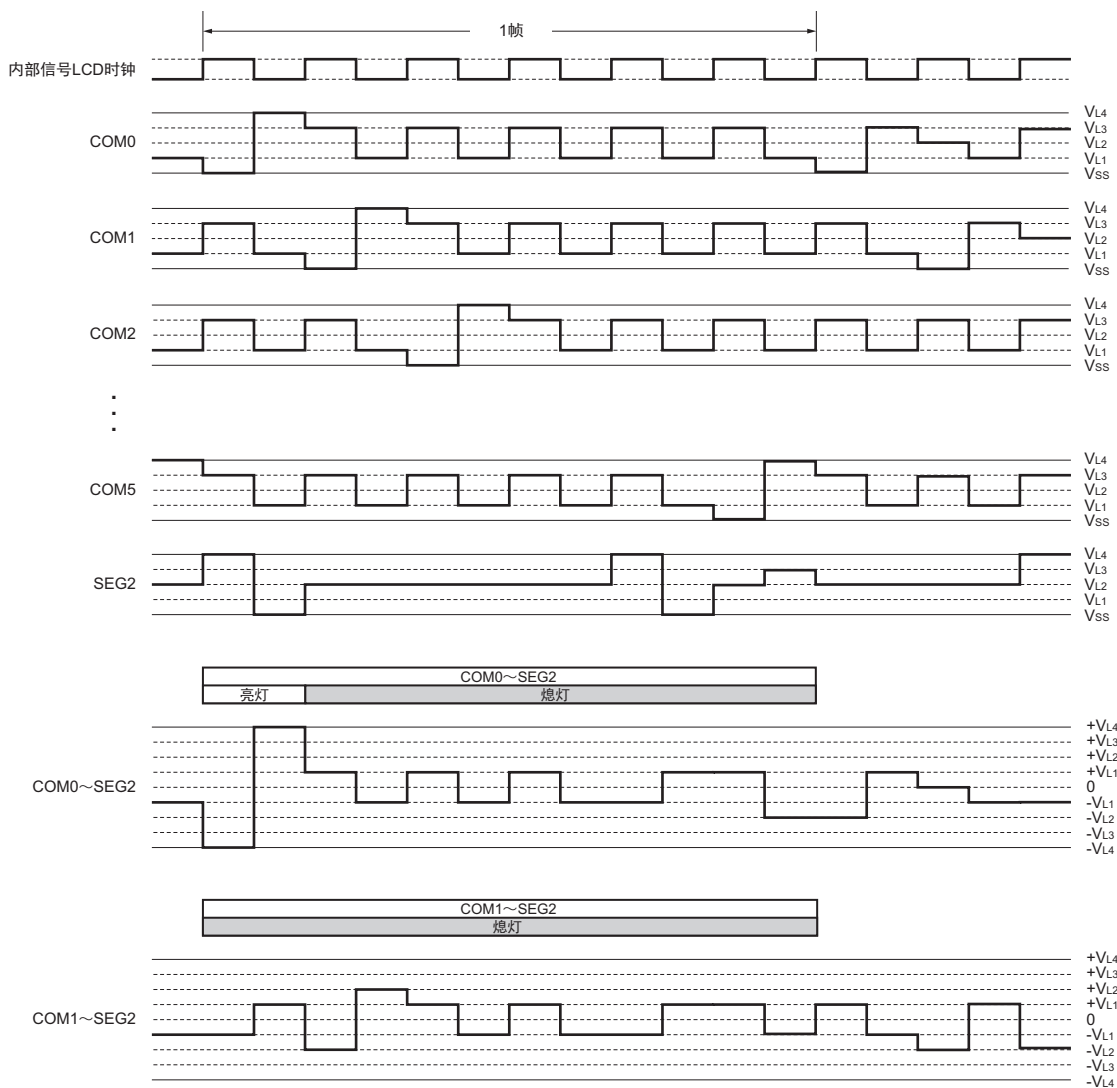
图 17-39 6 个时间片的 LCD 显示屏的连接例子



×: 因为是 6 个时间片显示, 所以可随时保存任意的数据。

图 17-40 SEG2 和各公共信号之间的 6 个时间片的 LCD 驱动波形例子 (1/4 偏压法)

(a) A 波形



17.10.6 8 个时间片的显示例子

图 17-41 显示图形的 8 个时间片方式的 15×8 点 LCD 显示屏、段信号（SEG4～SEG18）和公共信号（COM0～COM7）的连接如图 17-42 所示。显示例子是“123”，显示数据寄存器（地址 F0404H～F0412H）的内容对应此显示例子。

在此，以第 1 位的“3”（) 为例进行说明。需要根据图 17-41 的显示图形并且通过 COM0～COM7 的各公共信号的时序，将表 17-20 所示的选择电压和非选择电压输出到 SEG4～SEG8 引脚。

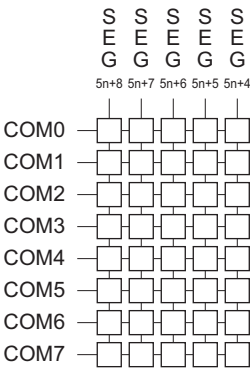
表 17-20 选择和非选择电压（COM0～COM7）

段信号 公共信号	SEG4	SEG5	SEG6	SEG7	SEG8
COM0	选择	选择	选择	选择	选择
COM1	非选择	选择	非选择	非选择	非选择
COM2	非选择	非选择	选择	非选择	非选择
COM3	非选择	选择	非选择	非选择	非选择
COM4	选择	非选择	非选择	非选择	非选择
COM5	选择	非选择	非选择	非选择	选择
COM6	非选择	选择	选择	选择	非选择
COM7	非选择	非选择	非选择	非选择	非选择

因此，给 SEG4 对应的显示数据寄存器（地址 F0404H）准备“00110001”即可。

SEG4 和各公共信号之间的 LCD 驱动波形例子如图 17-43 所示。在选择 COM0 时 SEG4 为选择电压，就知道产生 LCD 亮灯电平的波形。

图 17-41 8 个时间片的 LCD 显示图形和电极连线



备注 n = 0～8

图 17-42 8 个时间片的 LCD 显示屏的连接例子

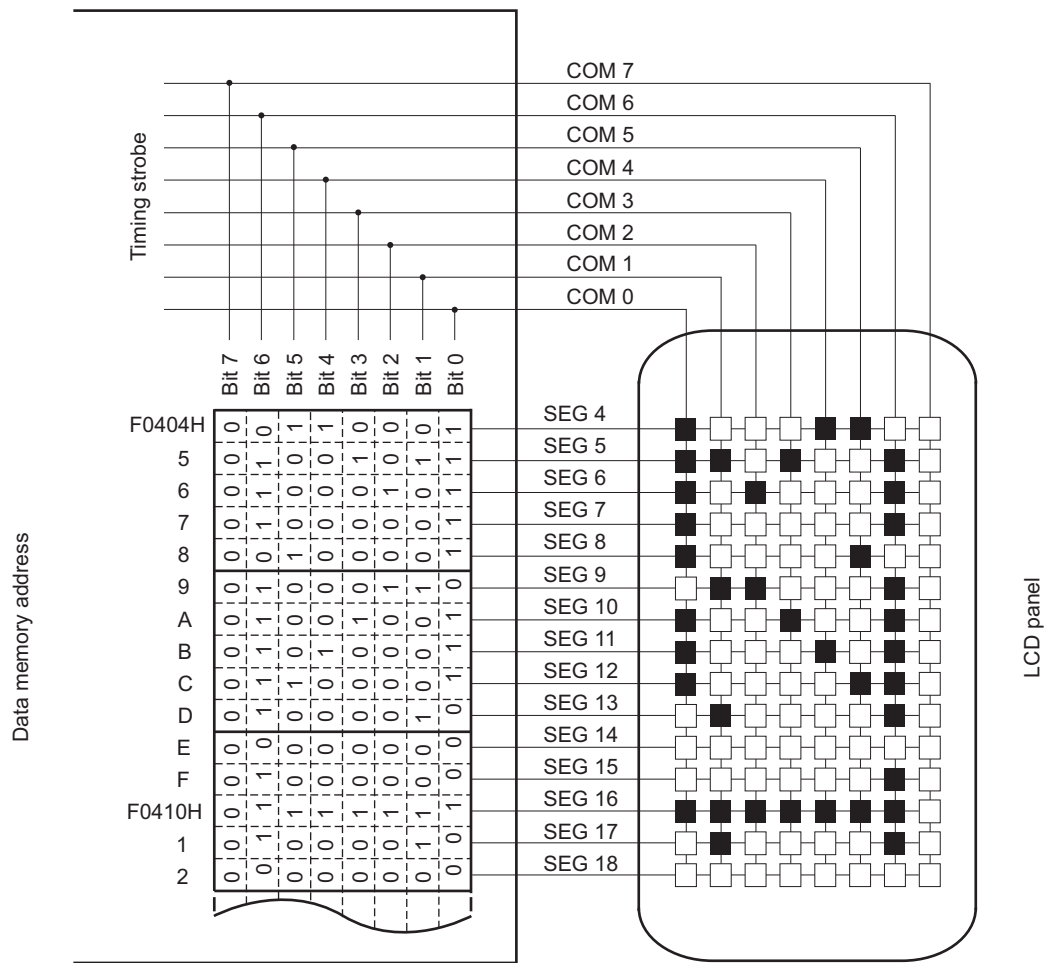


图 17-43 SEG4 和各公共信号之间的 8 个时间片的 LCD 驱动波形例子 (1/4 偏压法) (1/2)

(a) A 波形

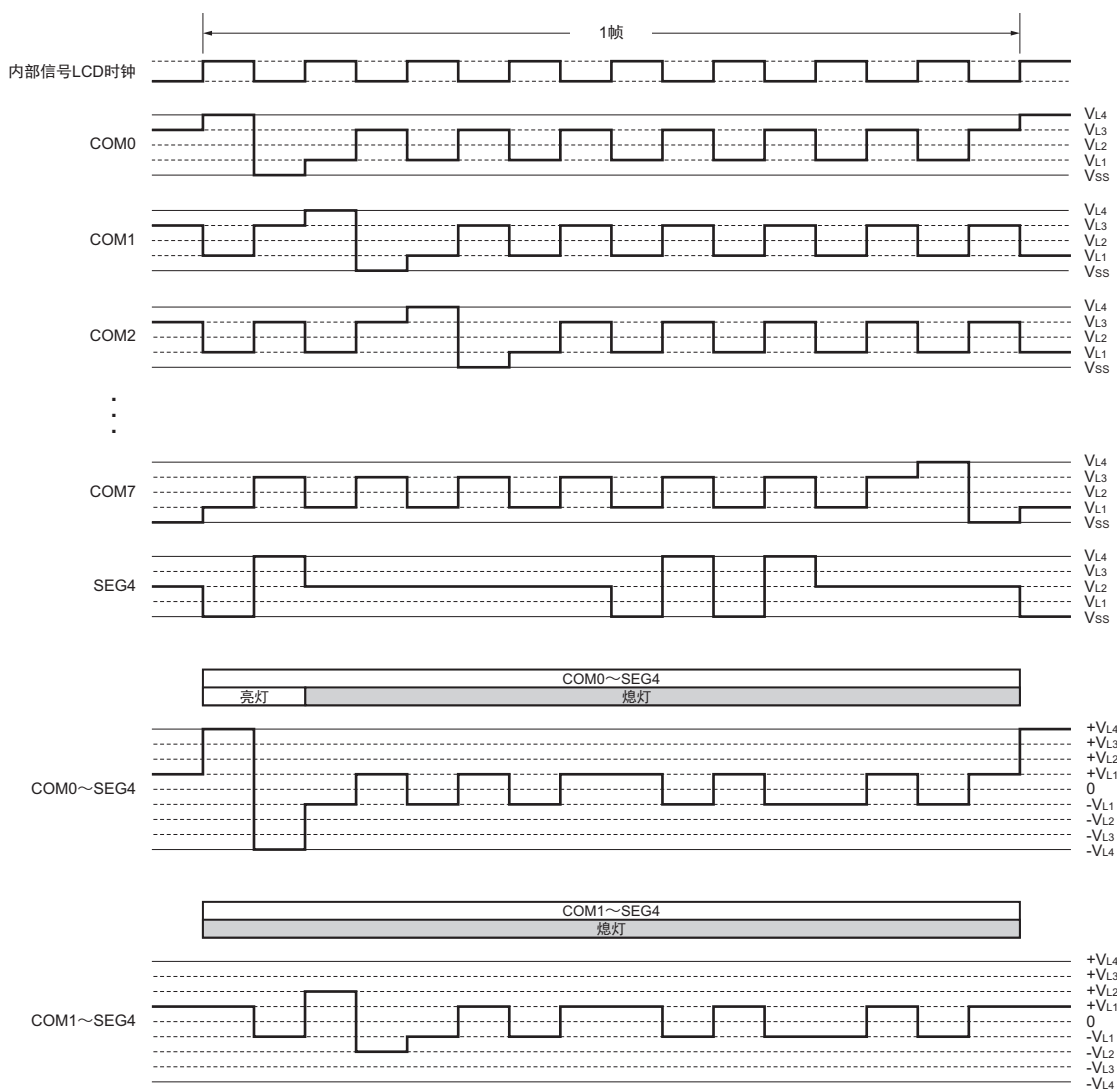
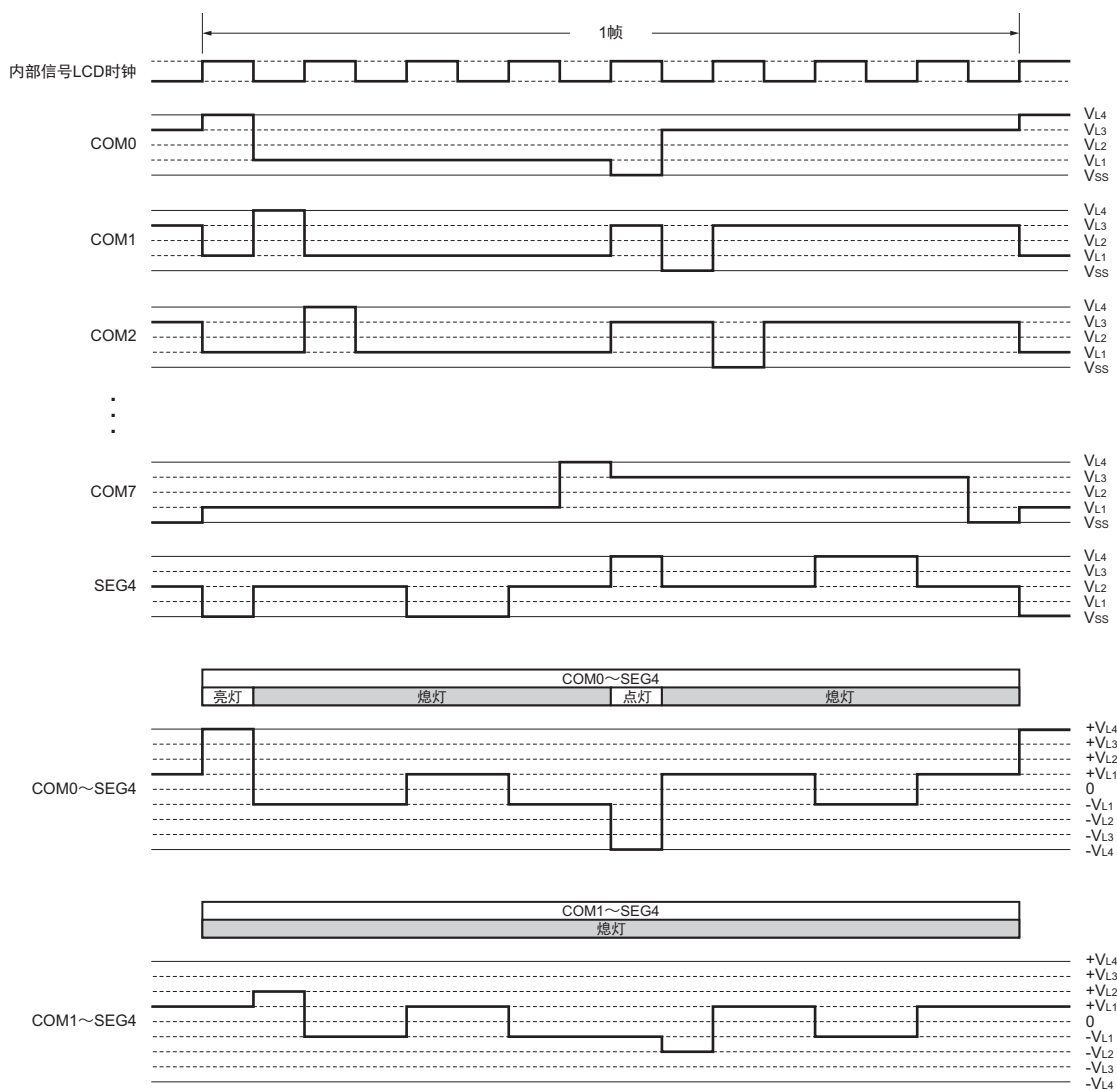


图 17-43 SEG4 和各公共信号之间的 8 个时间片的 LCD 驱动波形例子 (1/4 偏压法) (2/2)

(b) B 波形



第 18 章 乘除器和乘加器

18.1 乘除器和乘加器的功能

乘除器和乘加器有以下功能：

- 16 位×16 位=32 位（无符号）
- 16 位×16 位=32 位（带符号）
- 16 位×16 位+32 位=32 位（无符号）
- 16 位×16 位+32 位=32 位（带符号）
- 32 位÷32 位=32 位、余数 32 位（无符号）

18.2 乘除器和乘加器的结构

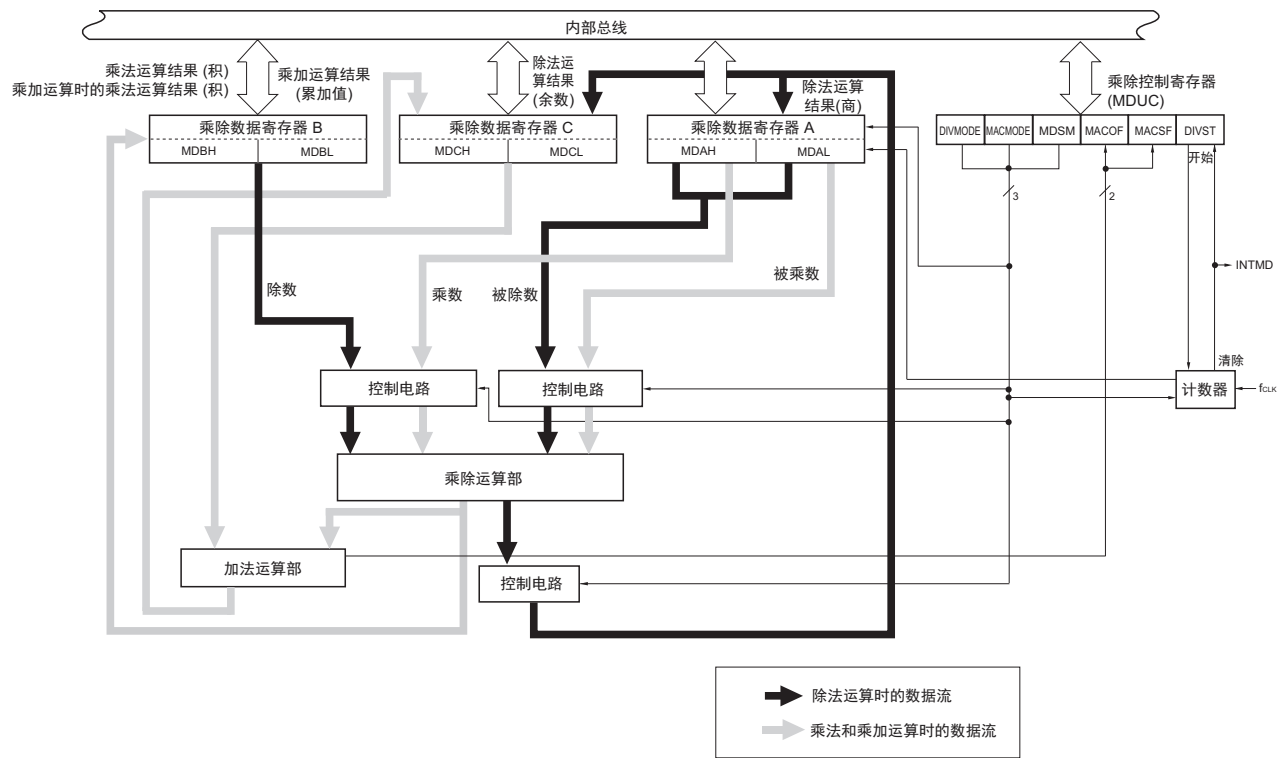
乘除器和乘加器由以下硬件构成。

表 18-1 乘除器和乘加器的结构

项目	结构
寄存器	乘除数据寄存器 A（L）（MDAL） 乘除数据寄存器 A（H）（MDAH） 乘除数据寄存器 B（L）（MDBL） 乘除数据寄存器 B（H）（MDBH） 乘除数据寄存器 C（L）（MDCL） 乘除数据寄存器 C（H）（MDCH）
控制寄存器	乘除控制寄存器（MDUC）

乘除器和乘加器的框图如图 18-1 所示。

图 18-1 乘除器和乘加器的框图



备注 f_{CLK}: CPU/ 外围硬件时钟频率

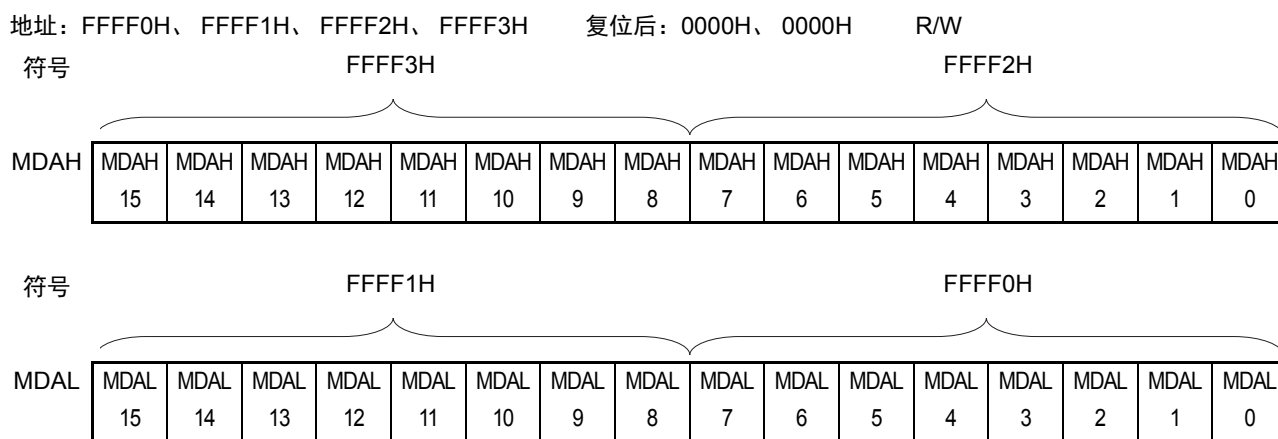
18.2.1 乘除数据寄存器 A (MDAH、MDAL)

MDAH 寄存器和 MDAL 寄存器设定用于乘除运算的值并且保存运算结果。在乘法运算模式或者乘加运算模式中，设定乘数和被乘数的数据；在除法运算模式中，设定被除数的数据。另外，在除法运算模式中，运算结果（商）保存在 MDAH 寄存器和 MDAL 寄存器。

通过 16 位操作指令设定 MDAH 寄存器和 MDAL 寄存器。

在产生复位信号后, 这些寄存器的值变为“0000H”。

图 18-2 乘除数据寄存器 A (MDAH、MDAL) 的格式



- 注意 1. 在除法运算过程中（乘除控制寄存器（MDUC）为“81H”或者“C1H”），不能改写 MDAH 寄存器和 MDAL 寄存器的值。否则，虽然进行运算，但是运算结果为不定值。
2. 在除法运算过程中（MDUC 寄存器为“81H”或者“C1H”），如果读 MDAH 寄存器和 MDAL 寄存器的值时，就不保证其读取值。
3. 在乘法运算模式（带符号）或者乘加运算模式（带符号）中，数据为 2 的补码。

运算时 MDAH 寄存器和 MDAL 寄存器的功能如下所示。

表 18-2 运算时 MDAH 寄存器和 MDAL 寄存器的功能

运算模式	设定	运算结果
乘法运算模式（无符号） 乘加运算模式（无符号）	MDAH：乘数（无符号） MDAL：被乘数（无符号）	—
乘法运算模式（带符号） 乘加运算模式（带符号）	MDAH：乘数（带符号） MDAL：被乘数（带符号）	—
除法运算模式（无符号）	MDAH：被除数（无符号） （高 16 位） MDAL：被除数（无符号） （低 16 位）	MDAH：除法运算结果（商）（无符号） 高 16 位 MDAL：除法运算结果（商）（无符号） 低 16 位

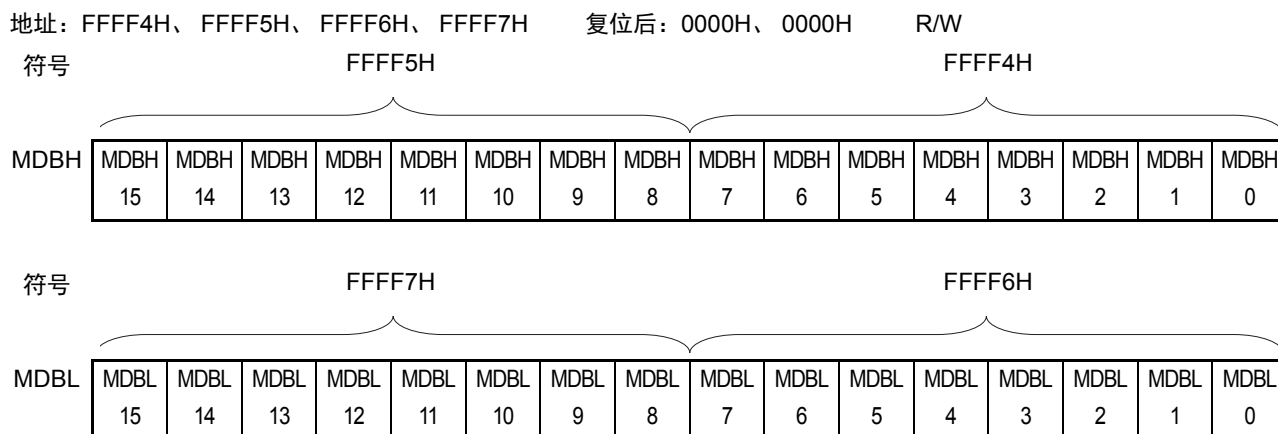
18.2.2 乘除数据寄存器 B (MDBL、MDBH)

MDBH 寄存器和 MDBL 寄存器设定用于乘除运算的值并且保存运算结果。在乘法运算模式或者乘加运算模式中，保存运算结果（积）；在除法运算模式中，设定除数的数据。

通过 16 位操作指令设定 MDBH 寄存器和 MDBL 寄存器。

在产生复位信号后, 这些寄存器的值变为“0000H”。

图 18-3 乘除数据寄存器 B (MDBH、MDBL) 的格式



- 注意 1. 在除法运算（乘除控制寄存器（MDUC）为“81H”或者“C1H”）或者乘加运算过程中，不能通过软件改写 MDBH 寄存器和 MDBL 寄存器的值。否则，运算结果为不定值。
2. 在除法运算模式中，不能将 MDBH 寄存器和 MDBL 寄存器置“0000H”。否则，运算结果为不定值。
3. 在乘法运算模式（带符号）或者乘加运算模式（带符号）中，数据为 2 的补码。

运算时 MDBH 寄存器和 MDBL 寄存器的功能如下所示。

表 18-3 运算时 MDBH 寄存器和 MDBL 寄存器的功能

运算模式	设定	运算结果
乘法运算模式（无符号）	—	MDBH：乘法运算结果（积）（无符号）高 16 位
乘加运算模式（无符号）		MDBL：乘法运算结果（积）（无符号）低 16 位
乘法运算模式（带符号）	—	MDBH：乘法运算结果（积）（带符号）高 16 位
乘加运算模式（带符号）		MDBL：乘法运算结果（积）（带符号）低 16 位
除法运算模式（无符号）	MDBH：除数（无符号） （高 16 位） MDBL：除数（无符号） （低 16 位）	—

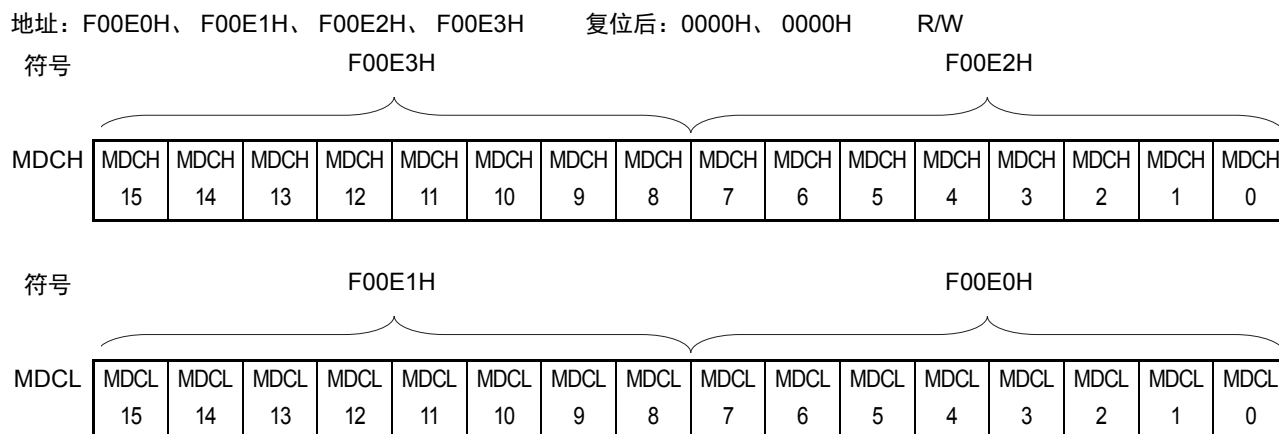
18.2.3 乘除数据寄存器 C (MDCL、MDCH)

MDCH 寄存器和 MDCL 寄存器在乘加运算模式中，保存累加结果；在除法运算模式中，保存运算结果的余数。在乘法运算模式中，不使用此寄存器。

通过 16 位操作指令设定 MDCH 寄存器和 MDCL 寄存器。

在产生复位信号后, 这些寄存器的值变为“0000H”。

图 18-4 乘除数据寄存器 C (MDCH、MDCL) 的格式



- 注意 1. 在除法运算过程中（乘除控制寄存器（MDUC）为“81H”或者“C1H”），如果读 MDCH 寄存器和 MDCL 寄存器的值，就不保证其读取值。
2. 在乘加运算过程中，不能通过软件改写 MDCH 寄存器和 MDCL 寄存器的值。否则，运算结果为不定值。
3. 在乘加运算模式（带符号）中，数据为 2 的补码。

表 18-4 运算时 MDCH 寄存器和 MDCL 寄存器的功能

运算模式	设定	运算结果
乘法运算模式 (无符号或者带符号)	—	—
乘加运算模式 (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)
乘加运算模式 (带符号)	MDCH: 累加初始值 (带符号) (高 16 位) MDCL: 累加初始值 (带符号) (低 16 位)	MDCH: 累加值 (带符号) (高 16 位) MDCL: 累加值 (带符号) (低 16 位)
除法运算模式 (无符号)	—	MDCH: 余数 (无符号) (高 16 位) MDCL: 余数 (无符号) (低 16 位)

在乘法运算和除法运算时的寄存器结构如下所示。

- 乘法运算时的寄存器结构

$\begin{array}{ccc} \text{<乘数A>} & \text{<乘数B>} & \text{<积>} \\ \text{MDAL (bit15~0)} \times \text{MDAH (bit15~0)} & = & [\text{MDBH (bit15~0)}, \text{MDBL (bit15~0)}] \end{array}$

- 乘加运算时的寄存器结构

$\begin{array}{ccc} \text{<乘数A>} & \text{<乘数B>} & \text{<累加值>} \\ \text{MDAL (bit15~0)} \times \text{MDAH (bit15~0)} + \text{MDC (bit31~0)} & = & \\ \text{<累加结果>} & & \\ [\text{MDCH (bit15~0)}, \text{MDCL (bit15~0)}] & & \\ (\text{MDBH (bit15~0)} \text{ 和 } \text{MDBL (bit15~0)}) \text{ 保存乘法运算结果} & & \end{array}$

- 除法运算时的寄存器结构

$\begin{array}{ccc} \text{<被除数>} & & \text{<除数>} \\ [\text{MDAH (bit15~0)}, \text{MDAL (bit15~0)}] \div [\text{MDBH (bit15~0)}, \text{MDBL (bit15~0)}] & = & \\ \text{<商>} & & \text{<余数>} \\ [\text{MDAH (bit15~0)}, \text{MDAL (bit15~0)}] \dots [\text{MDCH (bit15~0)}, \text{MDCL (bit15~0)}] & & \end{array}$

18.3 控制乘除器和乘加器的寄存器

通过乘除控制寄存器（MDUC）来控制乘除器和乘加器。

18.3.1 乘除控制寄存器（MDUC）

MDUC 寄存器是控制乘除器和乘加器运算的 8 位寄存器。

通过 1 位或者 8 位存储器操作指令设定 MDUC 寄存器。

乘加运算结果（累加值）的上溢标志（MACOF）和符号标志（MACSF）为只读标志。

在产生复位信号后，此寄存器的值变为“00H”。

图 18-5 乘除控制寄存器（MDUC）的格式

地址: F00E8H	复位后: 00H	R/W 注 1						
符号	7	6	5	4	3	2	1	0
MDUC	DIVMODE	MACMODE	0	0	MDSM	MACOF	MACSF	DIVST

DIVMODE	MACMODE	MDSM	运算模式的选择
0	0	0	乘法运算模式（无符号）（默认）
0	0	1	乘法运算模式（带符号）
0	1	0	乘加运算模式（无符号）
0	1	1	乘加运算模式（带符号）
1	0	0	除法运算模式（无符号）、发生除法运算结束中断（INTMD）
1	1	0	除法运算模式（无符号）、不发生除法运算结束中断（INTMD）
以上以外			禁止设定。

MACOF	乘加运算结果（累加值）的上溢标志
0	没有发生上溢。
1	发生上溢。

[置位条件]

- 乘加运算模式（无符号）
当累加值超出 00000000h ~ FFFFFFFFh 的范围时
- 乘加运算模式（带符号）
当正累加值与正积之和超过 7FFFFFFFh 而结果为负时
当负累加值与负积之和超过 80000000h 而结果为正时

MACSF	乘加结果（累加值）的符号标志
0	累加值为正。
1	累加值为负。

乘加运算模式（无符号）：总是“0”。
乘加运算模式（带符号）：表示累加值的符号位。

DIVST 注 2	除法运算的开始 / 停止
0	除法运算处理结束。
1	开始除法运算 / 正在除法运算。

注 1. bit1 和 bit2 为只读位。

2. 只有在除法运算模式中才能将 DIVST 位置“1”。在除法运算模式中，如果将 DIVST 位置“1”，就开始除法运算。在运算结束后，自动将 DIVST 位清“0”。在乘法运算模式中，通过给乘除数据寄存器 A（MDAH、MDAL）设定乘数和被乘数，自动开始运算。

注意 1. 在运算过程中（DIVST 位为“1”），不能改写 DIVMODE 位和 MDSM 位。否则，运算结果为不定值。

2. 在除法运算过程中（DIVST 位为“1”），不能通过软件将 DIVST 位清“0”。

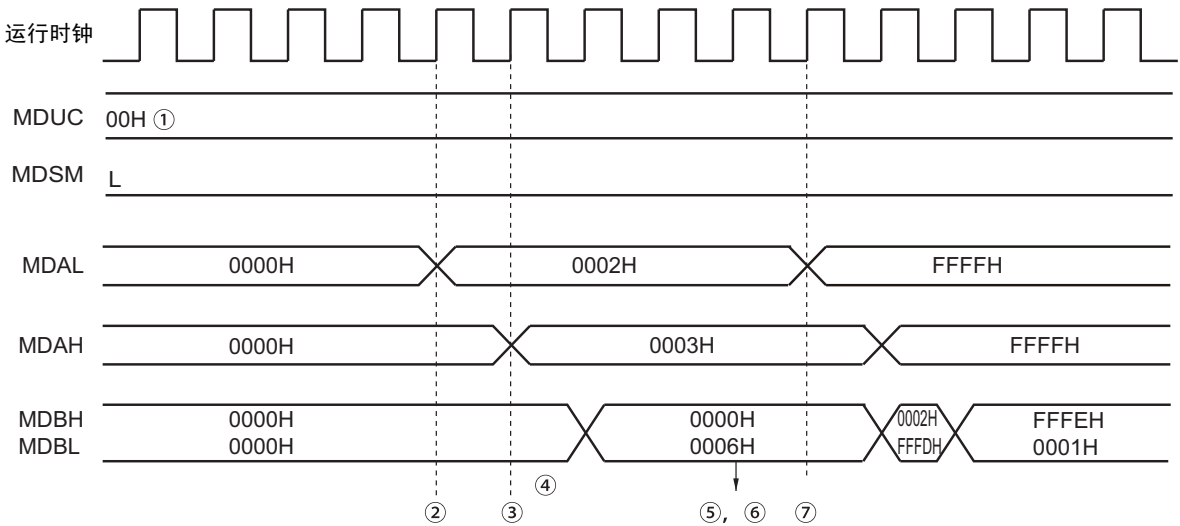
18.4 乘除器和乘加器的运算

18.4.1 乘法运算（无符号）

- 初始设定
 - ① 将乘除控制寄存器（MDUC）置“00H”。
 - ② 给乘除数据寄存器 A（L）（MDAL）设定被乘数。
 - ③ 给乘除数据寄存器 A（H）（MDAH）设定乘数。
（②和③的设定顺序不分先后。如果给MDAH寄存器和MDAL寄存器设定乘数和被乘数，就自动开始乘法运算）
- 运算处理
 - ④ 至少等待1个时钟。运算在1个时钟内完成。
- 运算结束
 - ⑤ 从乘除数据寄存器 B（L）（MDBL）读积（低16位）。
 - ⑥ 从乘除数据寄存器 B（H）（MDBH）读积（高16位）。
（⑤和⑥的读取顺序不分先后）
- 下次运算
 - ⑦ 如果要更改运算模式，必须从各运算步骤的初始设定开始执行。
如果继续使用相同运算模式，可省略①和②的设定。

备注 步骤①～⑦对应图 18-6 中的①～⑦。

图 18-6 乘法运算（无符号）的时序图（2×3=6）



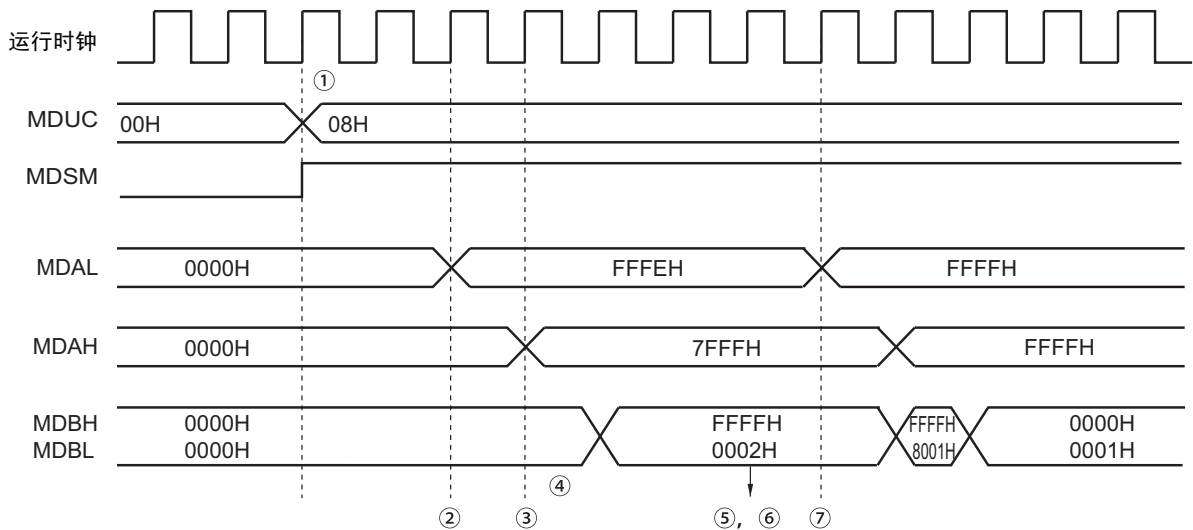
18.4.2 乘法运算（带符号）

- 初始设定
 - ① 将乘除控制寄存器（MDUC）置“08H”。
 - ② 给乘除数据寄存器 A（L）（MDAL）设定被乘数。
 - ③ 给乘除数据寄存器 A（H）（MDAH）设定乘数。
（②和③的设定顺序不分先后。如果给MDAH寄存器和MDAL寄存器设定乘数和被乘数，就自动开始乘法运算）
- 运算处理
 - ④ 至少等待1个时钟。运算在1个时钟内完成。
- 运算结束
 - ⑤ 从乘除数据寄存器 B（L）（MDBL）读积（低16位）。
 - ⑥ 从乘除数据寄存器 B（H）（MDBH）读积（高16位）。
（⑤和⑥的读取顺序不分先后）
- 下次运算
 - ⑦ 如果要更改运算模式，必须从各运算步骤的初始设定开始执行。
如果继续使用相同运算模式，可省略①和②的设定。

注意 在乘法运算模式（带符号）中，数据为 2 的补码。

备注 步骤①～⑦对应图 18-7 中的①～⑦。

图 18-7 乘法运算（带符号）的时序图（-2×32767=-65534）

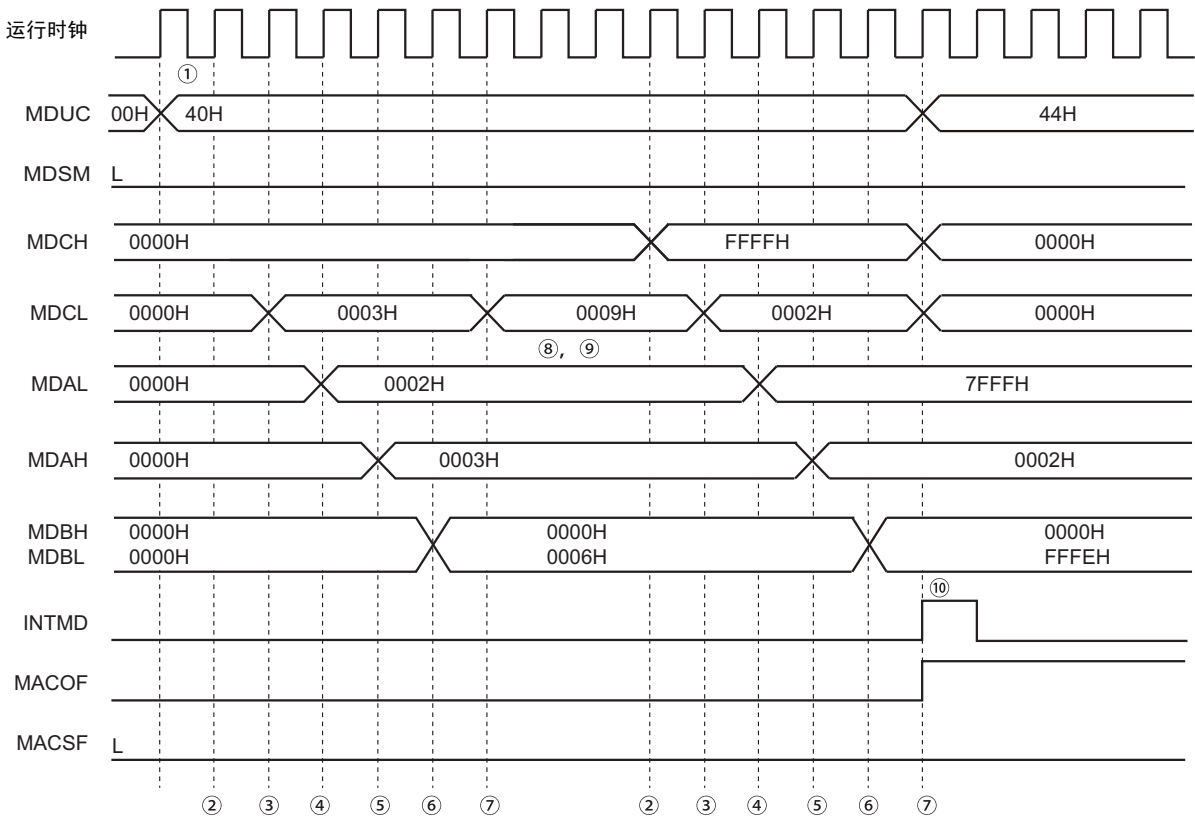


18.4.3 乘加运算（无符号）

- 初始设定
 - ① 将乘除控制寄存器（MDUC）置“40H”。
 - ② 给乘除数据寄存器C（H）（MDCH）设定累加初始值的高16位。
 - ③ 给乘除数据寄存器C（L）（MDCL）设定累加初始值的低16位。
 - ④ 给乘除数据寄存器A（L）（MDAL）设定被乘数。
 - ⑤ 给乘除数据寄存器A（H）（MDAH）设定乘数。
 - （②、③和④的设定顺序不分先后。如果给⑤的MDAH寄存器设定乘数，就自动开始乘法运算）
- 运算处理
 - ⑥ 乘法运算在1个时钟内完成。
 - （乘法结果保存在乘除数据寄存器B（L）（MDBL）和乘除数据寄存器B（H）（MDBH））
 - ⑦ 在⑥之后，还需要1个时钟完成乘加运算（在初始设定结束（⑤）后，至少等待2个时钟）。
- 运算结束
 - ⑧ 从乘除数据寄存器C（L）（MDCL）读累加值（低16位）。
 - ⑨ 从乘除数据寄存器C（H）（MDCH）读累加值（高16位）。
 - （⑧和⑨的读取顺序不分先后）
 - （⑩当乘加运算的结果发生上溢时，将MACOF位置“1”并且产生INTMD信号）
- 下次运算
 - ⑪ 如果要更改运算模式，必须从各运算步骤的初始设定开始执行。
 - 如果继续使用相同运算模式，可省略①～④的设定。

备注 步骤①～⑩对应图 18-8 中的①～⑩。

图 18-8 乘加运算（无符号）的时序图
($2 \times 3 + 3 = 9 \rightarrow 32767 \times 2 + 4294901762 = 0$ （发生上溢）)



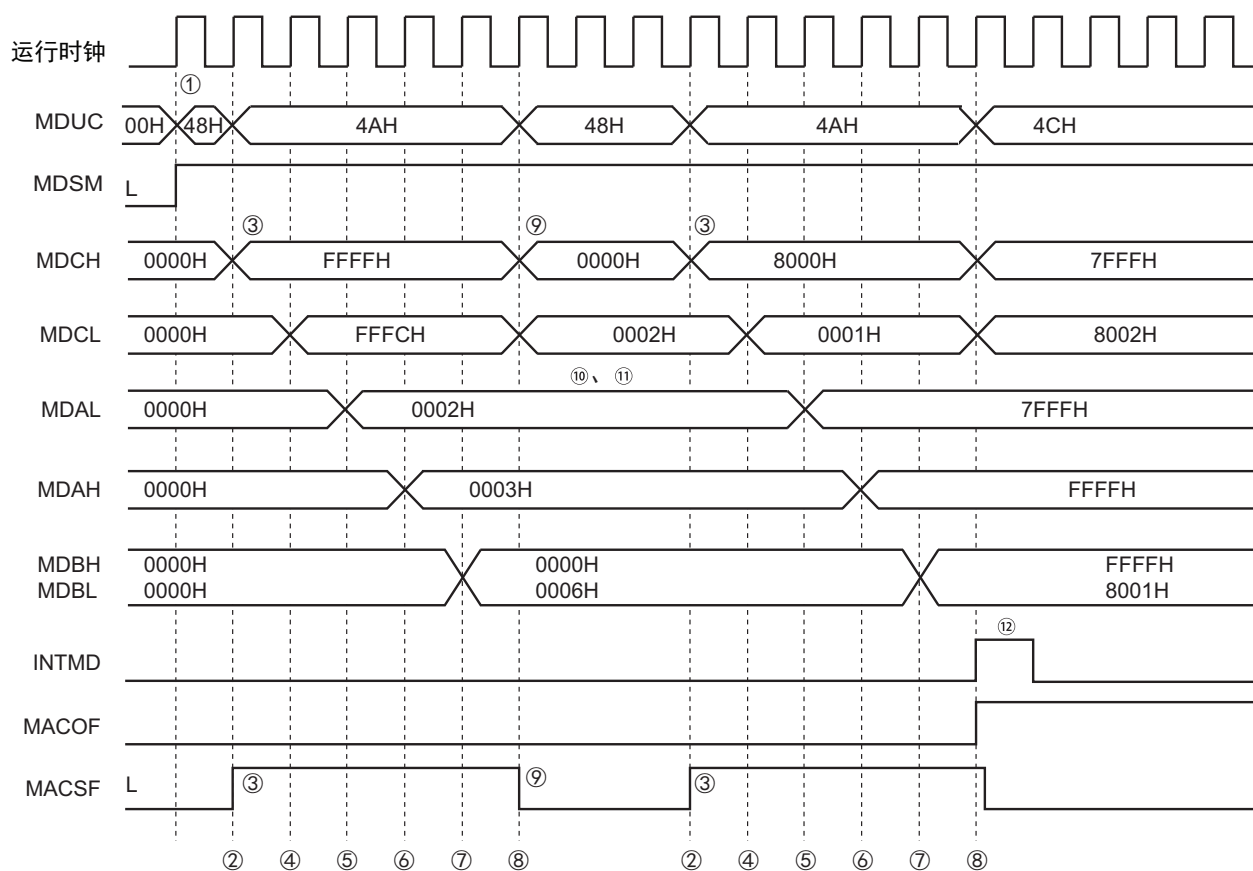
18.4.4 乘加运算（带符号）

- 初始设定
 - ① 将乘除控制寄存器（MDUC）置“48H”。
 - ② 给乘除数据寄存器C（H）（MDCH）设定累加初始值的高16位。
（③ 当MDCH寄存器的累加值为负值时，将MACSF位置“1”）
 - ④ 给乘除数据寄存器C（L）（MDCL）设定累加初始值的低16位。
 - ⑤ 给乘除数据寄存器A（L）（MDAL）设定被乘数。
 - ⑥ 给乘除数据寄存器A（H）（MDAH）设定乘数。
（②、④和⑤的设定顺序不分先后。如果给⑥的MDAH寄存器设定乘数，就自动开始乘法运算）
- 运算处理
 - ⑦ 乘法运算在1个时钟内完成。
（乘法结果保存在乘除数据寄存器B（L）（MDBL）和乘除数据寄存器B（H）（MDBH））
 - ⑧ 在⑦之后，还需要1个时钟完成乘加运算（在初始设定结束（⑥）后，至少等待2个时钟）。
- 运算结束
 - ⑨ 当保存在MDCL寄存器和MDCH寄存器中的累加值为正值时，将MACSF位清“0”。
 - ⑩ 从MDCL寄存器读累加值（低16位）。
 - ⑪ 从MDCH寄存器读累加值（高16位）。
（⑩和⑪的读取顺序不分先后）
 - （⑫ 当乘加运算的结果发生上溢时，将MACOF位置“1”并且产生INTMD信号）
- 下次运算
 - ⑬ 如果要更改运算模式，必须从各运算步骤的初始设定开始执行。
如果继续使用相同运算模式，可省略①～⑤的设定。

注意 在乘加运算模式（带符号）中，数据为2的补码。

备注 步骤①～⑫对应图 18-9 中的①～⑫。

图 18-9 乘加运算（带符号）的时序图
 $(2 \times 3 + (-4)) = 2 \rightarrow 32767 \times (-1) + (-2147483647) = -2147516414$ （发生上溢）

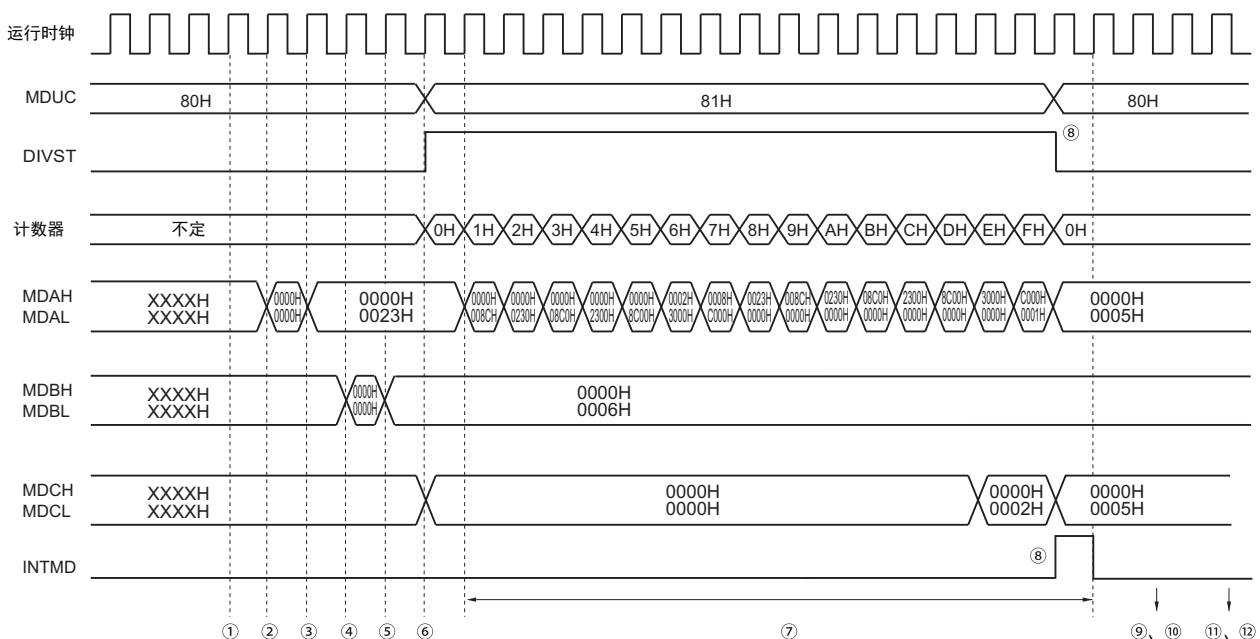


18.4.5 除法运算

- 初始设定
 - ① 将乘除控制寄存器（MDUC）置“80H”。
 - ② 给乘除数据寄存器A（H）（MDAH）设定被除数（高16位）。
 - ③ 给乘除数据寄存器A（L）（MDAL）设定被除数（低16位）。
 - ④ 给乘除数据寄存器B（H）（MDBH）设定除数（高16位）。
 - ⑤ 给乘除数据寄存器B（L）（MDBL）设定除数（低16位）。
 - ⑥ 将MDUC寄存器的bit0（DIVST）置“1”。
（②～⑤的顺序不分先后）
- 运算处理
 - ⑦ 如果完成以下某种处理，就结束运算。
 - 至少等待16个时钟（运算在16个时钟内完成）。
 - 确认DIVST位被清除。
（在运算过程中，不保证MDBL、MDBH、MDCL、MDCH寄存器的读取值）
- 运算结束
 - ⑧ 将DIVST位清“0”并且结束运算。此时，如果是MACMODE=0的运算，就产生中断请求信号（INTMD）。
 - ⑨ 从MDAL寄存器读商（低16位）。
 - ⑩ 从MDAH寄存器读商（高16位）。
 - ⑪ 从乘除数据寄存器C（L）（MDCL）读余数（低16位）。
 - ⑫ 从乘除数据寄存器C（H）（MDCH）读余数（高16位）。
（⑨～⑫的顺序不分先后）
- 下次运算
 - ⑬ 如果要更改运算模式，必须从各运算步骤的初始设定开始执行。
如果继续使用相同运算模式，可省略①～⑤的设定。

备注 步骤①~⑫对应图 18-10 中的①~⑫。

图 18-10 除法运算的时序图 ($35 \div 6 = 5$ 余数 5)



第 19 章 DMA 控制器

R7F0C003、R7F0C004 内置一个 DMA（Direct Memory Access）控制器。

能在支持 DMA 外围硬件的 SFR 和内部 RAM 之间不经过 CPU 而自动传送数据。

因为能在进行通常的 CPU 内部运算和数据传送的同时进行 SFR 和内部 RAM 之间的传送，所以能进行大容量数据的处理，并且还能实现通过通信、定时器和 A/D 进行的实时控制。

19.1 DMA 控制器的功能

- DMA 通道个数：4 个通道
- 传送单位：8 位或者 16 位
- 最大传送单位：1024 次
- 传送类型：2 个周期传送（通过 2 个时钟进行一次传送并且在传送期间 CPU 停止运行）
- 传送模式：单次传送模式
- 传送请求：选择以下外围硬件中断。
 - A/D 转换器
 - 串行接口（CSI00、UART0～UART3、SMCI0、SMCI1）
 - 定时器阵列单元（通道 0、1、2、3）
- 传送对象：SFR 和内部 RAM 之间的传送

使用 DMA 功能的例子如下所示：

- 串行接口的连续传送
- A/D 转换结果的连续取得
- 固定时间的 A/D 转换结果的取得
- 固定时间的端口值的取得

19.2 DMA 控制器的结构

DMA 控制器由以下硬件构成。

表 19-1 DMA 控制器的结构

项目	结构
地址寄存器	- DMA SFR 地址寄存器 0 ~ 3 (DSA0 ~ DSA3) - DMA RAM 地址寄存器 0 ~ 3 (DRA0 ~ DRA3)
计数寄存器	DMA 字节计数寄存器 0 ~ 3 (DBC0 ~ DBC3)
控制寄存器	- DMA 模式控制寄存器 0 ~ 3 (DMC0 ~ DMC3) - DMA 运行控制寄存器 0 ~ 3 (DRC0 ~ DRC3)

19.2.1 DMA SFR 地址寄存器 n (DSAn)

这是设定 SFR 地址的 8 位寄存器，SFR 地址是 DMA 通道 n 的传送源或者传送目标。

必须设定 SFR 地址 FFF00H ~ FFFFFH 的低 8 位。

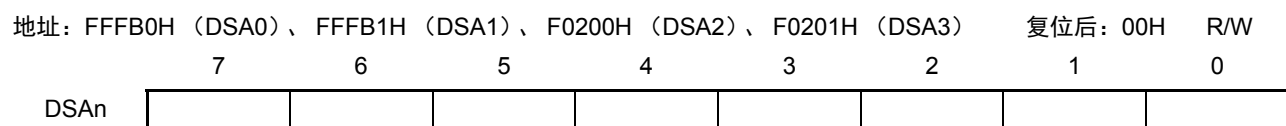
此寄存器为固定值而不会自动递增。

在 16 位传送模式中，忽略最低位而作为偶数地址处理。

能以 8 位为单位读写 DSAn 寄存器，但是不能在 DMA 传送过程中写此寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 19-1 DMA SFR 地址寄存器 n (DSAn) 的格式



备注 n: DMA 通道号 (n=0 ~ 3)

19.2.2 DMA RAM 地址寄存器 n（DRAn）

这是设定 RAM 地址的 16 位寄存器，RAM 地址是 DMA 通道 n 的传送源或者传送目标。
能设定通用寄存器以外的内部 RAM 区（参照表 19-2）的地址。
必须设定 RAM 地址的低 16 位。

如果开始 DMA 传送，此寄存器就自动递增。在 8 位传送模式中增 1；在 16 位传送模式中增 2。如果从此 DRAn 寄存器的设定地址开始传送到最后地址，DMA 传送就停止运行。此时，在 8 位传送模式中，DRAn 寄存器为最后地址 +1；在 16 位传送模式中，DRAn 寄存器为最后地址 +2。

在 16 位传送模式中，忽略最低位而作为偶数地址处理。
能以 8 位或者 16 位为单位读写 DRAn 寄存器，但是不能在 DMA 传送过程中写此寄存器。
在产生复位信号后，此寄存器的值变为“0000H”。

图 19-2 DMA RAM 地址寄存器 n（DRAn）的格式

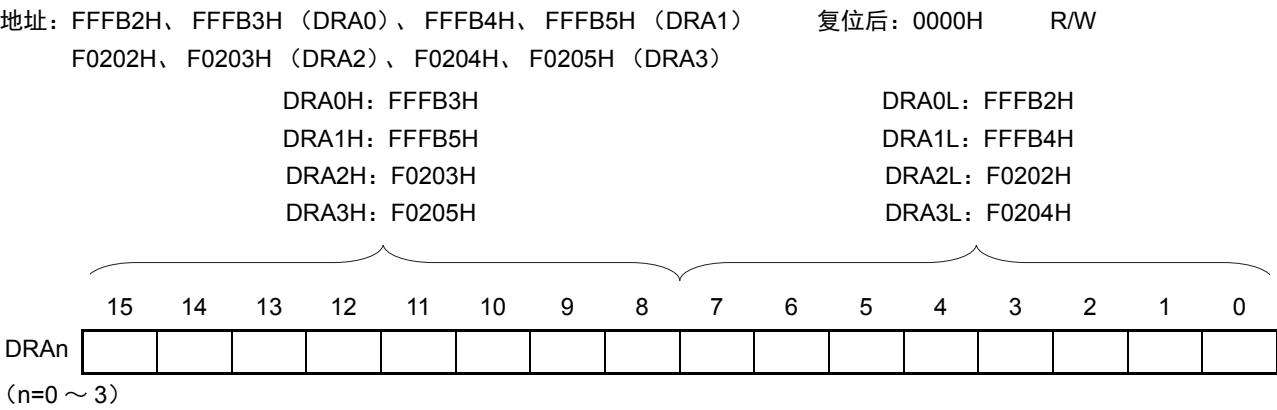


表 19-2 通用寄存器以外的内部 RAM 区

产品	通用寄存器以外的内部 RAM 区
R7F0C003、R7F0C004	FDF00H ~ FFEDFH

备注 n: DMA 通道号（n=0 ~ 3）

19.2.3 DMA 字节计数寄存器 n (DBCn)

这是设定 DMA 通道 n 传送次数的 10 位寄存器。必须在 DMA 传送前给此 DBCn 寄存器设定传送次数（最多 1024 次）。

每进行一次 DMA 传送，此寄存器就自动递减。在 DMA 传送过程中，能通过读此 DBCn 寄存器获得剩余的传送次数。

能以 8 位或者 16 位为单位读写 DBCn 寄存器，但是不能在 DMA 传送过程中写此寄存器。

在产生复位信号后，此寄存器的值变为“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)
F0206H、F0207H (DBC2)、F0208H、F0209H (DBC3)

复位后：0000H R/W

DBC0H: FFFB7H

DBC0L: FFFB6H

DBC1H: FFFB9H

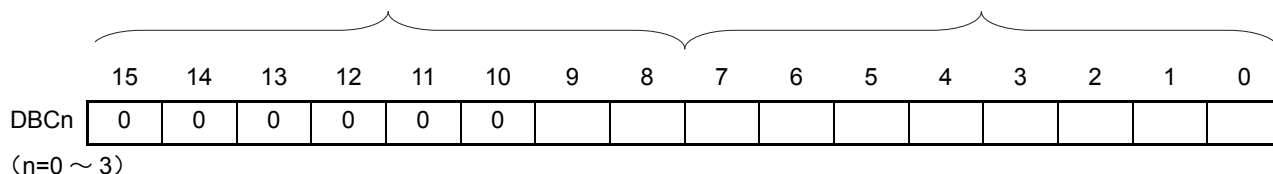
DBC1L: FFFB8H

DBC2H: F0207H

DBC2L: F0206H

DBC3H: F0209H

DBC3L: F0208H



DBCn[9:0]	传送次数的设定（写 DBCn）	剩余的传送次数（读 DBCn）
000H	1024 次	传送结束或者等待 1024 次的 DMA 传送
001H	1 次	等待剩余的 1 次 DMA 传送
002H	2 次	等待剩余的 2 次 DMA 传送
003H	3 次	等待剩余的 3 次 DMA 传送
⋮	⋮	⋮
3FEH	1022 次	等待剩余的 1022 次 DMA 传送
3FFH	1023 次	等待剩余的 1023 次 DMA 传送

注意 1. 必须将 bit15 ~ 10 置“0”。

2. 如果指定通用寄存器或者因连续传送而超过内部 RAM 空间，就读写通用寄存器或者 SFR 空间，导致数据的破坏。
必须将传送次数设定在内部 RAM 空间范围内。

备注 n: DMA 通道号 (n=0 ~ 3)

19.3 控制 DMA 控制器的寄存器

DMA 控制器由以下 2 个寄存器控制。

- DMA 模式控制寄存器 n (DMCn)
- DMA 运行控制寄存器 n (DRCn)

备注 n: DMA 通道号 (n=0 ~ 3)

19.3.1 DMA 模式控制寄存器 n (DMCn)

DMCn 寄存器设定 DMA 通道 n 的传送模式，选择传送方向、数据宽度、保留设定和启动源。bit7 (STGn) 为启动 DMA 的软件触发。

禁止在 DMA 运行过程中 (DSTn=1) 改写 DMCn 寄存器的 bit6、bit5 和 bit3 ~ 0。

通过 1 位或者 8 位存储器操作指令设定 DMCn 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 19-4 DMA 模式控制寄存器 n (DMCn) 的格式 (1/3)

地址: FFFBAH (DMC0)、FFFBH (DMC1)、F020AH (DMC2)、F020BH (DMC3) 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 注 1	DMA 传送开始的软件触发
0	软件触发不运行。
1	在允许 DMA 运行 (DENn=1) 时，开始 DMA 传送。
在允许 DMA 运行 (DENn=1) 时，通过给 STGn 位写“1”进行一次 DMA 传送。 此位的读取值总是“0”。	

DRSn	DMA 传送方向的选择
0	SFR→内部 RAM
1	内部 RAM→SFR

DSn	DMA 传送数据长度的指定
0	8 位
1	16 位

DWAITn 注 2	DMA 传送的保留
0	通过 DMA 启动请求进行 DMA 传送 (不保留)。
1	即使接收到 DMA 启动请求也保留 DMA 传送。
能通过将 DWAITn 位的值从“1”变为“0”来开始被保留的 DMA 传送。 在将 DWAITn 位的值从“0”变为“1”后，实际上需要经过 2 个时钟，然后保留 DMA 传送。	

注 1. 与 IFCn3 ~ IFCn0 位的值无关，能使用软件触发 (STGn)。

2. 要在使用 2 个或者更多通道的 DMA 过程中保留 DMA 传送时，必须保留全部通道的 DMA 传送 (DWAIT0=DWAIT1=DWAIT2=DWAIT3=1)。

备注 n: DMA 通道号 (n=0 ~ 3)

图 19-4 DMA 模式控制寄存器 n (DMCn) 的格式 (2/3)

地址: FFFBAH (DMC0)、FFFBH (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

(n=0、1)

IFCn 3	IFCn 2	IFCn 1	IFCn 0	DMA 启动源的选择注	
				触发信号	触发内容
0	0	0	0	—	通过中断禁止 DMA 传送。 (只能允许软件触发)
0	0	0	1	INTAD	A/D 转换结束中断
0	0	1	0	INTTM00	定时器通道 00 的计数结束或者捕捉中断
0	0	1	1	INTTM01	定时器通道 01 的计数结束或者捕捉中断
0	1	0	0	INTTM02	定时器通道 02 的计数结束或者捕捉中断
0	1	0	1	INTTM03	定时器通道 03 的计数结束或者捕捉中断
0	1	1	0	INTST0/INTCSI00	UART0 发送的传送结束或者缓冲器空中断/ CSI00 传送结束或者缓冲器空中断
0	1	1	1	INTSR0	UART0 接收的传送结束中断
1	0	0	0	INTST1	UART1 发送的传送结束或者缓冲器空中断
1	0	0	1	INTSR1	UART1 接收的传送结束中断
1	0	1	0	INTST2	UART2 发送的传送结束或者缓冲器空中断
1	0	1	1	INTSR2	UART2 接收的传送结束中断
1	1	0	0	INTSCR1	SMCI1 接收的数据满中断
1	1	0	1	INTSCT1	SMCI1 发送的传送结束中断
上述以外				禁止设定。	

注 与 IFCn3 ~ IFCn0 位的值无关, 能使用软件触发 (STGn)。

备注 n: DMA 通道号 (n=0、1)

图 19-4 DMA 模式控制寄存器 n（DMCn）的格式（3/3）

地址：F020AH（DMC2）、F020BH（DMC3）

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

(n=2、3)

IFCn	IFCn	IFCn	IFCn	DMA 启动源的选择注	
3	2	1	0	触发信号	触发内容
0	0	0	0	—	禁止通过中断进行的 DMA 传送。 (只能允许软件触发)
0	0	0	1	INTAD	A/D 转换结束中断
0	1	1	0	INTST3	UART3 发送的传送结束或者缓冲器空中断
0	1	1	1	INTSR3	UART3 接收的传送结束中断
1	0	0	0	INTST1	UART1 发送的传送结束或者缓冲器空中断
1	0	0	1	INTSR1	UART1 接收的传送结束中断
1	0	1	0	INTST2	UART2 发送的传送结束或者缓冲器空中断
1	0	1	1	INTSR2	UART2 接收的传送结束中断
1	1	0	0	INTSCR0	SMCI0 接收的数据满
1	1	0	1	INTSCT0	SMCI0 发送的传送结束
上述以外				禁止设定。	

注

与 IFCn3 ～ IFCn0 位的值无关，能使用软件触发（STGn）。

备注

n: DMA 通道号（n=2、3）

19.3.2 DMA 运行控制寄存器 n（DRCn）

DRCn 寄存器设定允许或者禁止 DMA 通道 n 的传送。

禁止在 DMA 运行中（DSTn=1）改写 DRCn 寄存器的 bit7（DENn）。

通过 1 位或者 8 位存储器操作指令设定 DRCn 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 19-5 DMA 运行控制寄存器 n（DRCn）的格式

地址：FFFBCH（DRC0）、FFBDH（DRC1）、F020CH（DRC2）、F020DH（DRC3）复位后：00HR/W

符号	7	6	5	4	3	2	1	0
DRCn	DENn	0	0	0	0	0	0	DSTn

DENn	DMA 运行允许标志
0	禁止 DMA 通道 n 的运行（停止 DMA 的运行时钟）。
1	允许 DMA 通道 n 的运行。

在设定为允许 DMA 运行（DENn=1）后，通过将 DSTn 标志置“1”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 传送没有结束（正在传送）。

在设定为允许 DMA 运行（DENn=1）后，通过将 DSTn 标志置“1”进入等待 DMA 触发的状态。
如果输入软件触发（STGn）或者输入由 IFCn3～IFCn0 位设定的启动源触发，就开始 DMA 传送。
此后，如果 DMA 传送结束，此标志就自动将清“0”。
要在 DMA 传送过程中强制结束传送时，给此标志写“0”。

注意 如果 DMA 传送结束，DSTn 标志就自动清“0”。

因为只有在 DSTn 标志为“0”时才允许写 DENn 标志，所以当不等待 DMA 中断（INTDMA_n）的发生而结束 DMA 传送时，必须在将 DSTn 标志置“0”后将 DENn 标志置“0”（有关详细内容，请参照“19.5.5 通过软件强制结束”）。

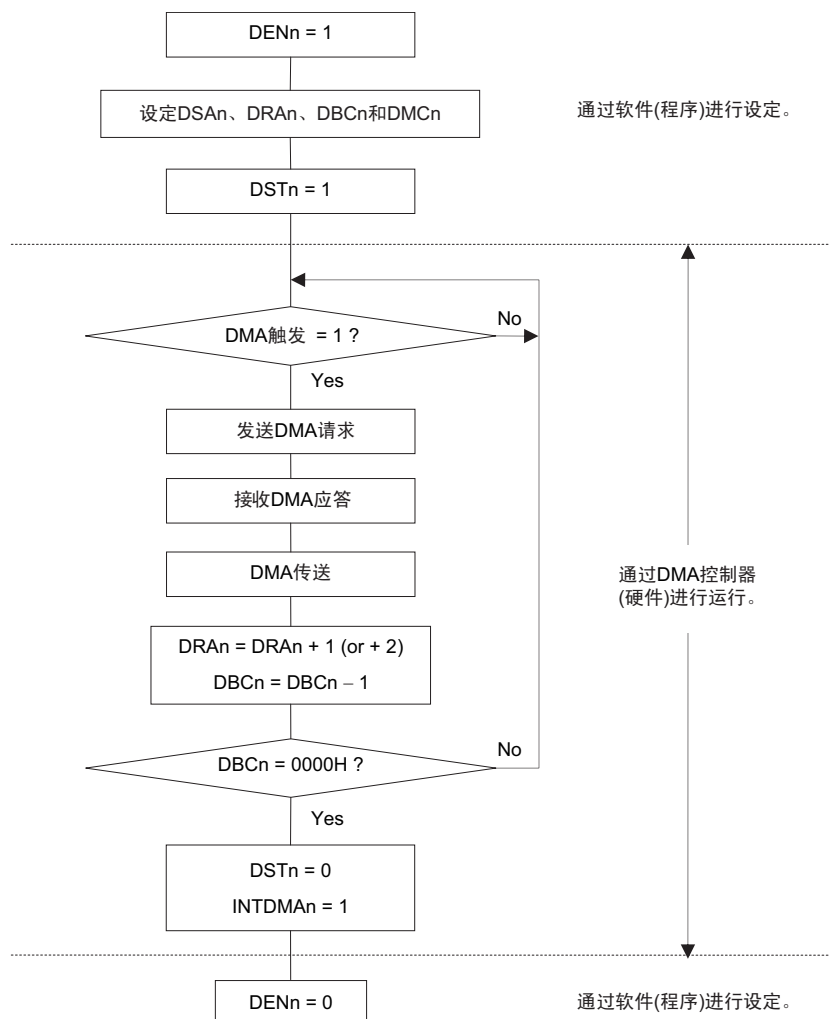
备注 n：DMA 通道号（n=0～3）

19.4 DMA 控制器的运行

19.4.1 运行步骤

- ① 通过将DENn位置“1”，使DMA控制器变为允许运行状态。必须在将DENn位置“1”后写其他寄存器。当通过8位操作指令进行写操作时，写“80H”。
- ② 给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）设定DMA 传送的SFR 地址、RAM 地址、传送次数和传送模式。
- ③ 通过将DSTn位置“1”，使DMA 变为等待触发的状态。当通过8位操作指令进行写操作时，写“81H”。
- ④ 如果输入软件触发（STGn）或者输入由IFCn3～IFCn0位设定的启动源触发，就开始DMA 传送。
- ⑤ 如果DBCn 寄存器设定的传送次数变为“0”，就完成传送并且产生中断请求（INTDMA n）而自动结束传送。
- ⑥ 此后，如果不使用DMA 控制器，就必须将DENn位置“0”，进入运行停止状态。

图 19-6 运行步骤



备注 n: DMA 通道号 (n=0 ~ 3)

19.4.2 传送模式

能通过设定 DMA 模式控制寄存器 n (DMCn) 的 bit6 和 bit5 (DRSn、DSn)，选择以下 4 种 DMA 传送模式。

DRSn	DSn	DMA 传送模式
0	0	从 1 字节数据（固定地址）的 SFR 到 RAM（地址递增 +1）的传送
0	1	从 2 字节数据（固定地址）的 SFR 到 RAM（地址递增 +2）的传送
1	0	从 1 字节数据（地址递增 +1）的 RAM 到 SFR（固定地址）的传送
1	1	从 2 字节数据（地址递增 +2）的 RAM 到 SFR（固定地址）的传送

通过这些传送模式，能利用串行接口最多连续传送 1024 字节的数据，并且能连续传送 A/D 转换结果的数据以及利用定时器以固定时间扫描端口的数据。

19.4.3 DMA 传送的结束

如果 DBCn 变为“00H”而完成 DMA 传送，DSTn 标志就自动清“0”，然后产生中断请求（INTDMA_n）而结束传送。

如果为了强制结束 DMA 传送而将 DSTn 位清“0”，DMA 字节计数寄存器 n (DBCn) 和 DMA RAM 地址寄存器 n (DRAn) 就保持停止时的值。

当强制结束传送时，不产生中断请求（INTDMA_n）。

备注 n: DMA 通道号 (n=0 ~ 3)

19.5 DMA 控制器的设定例子

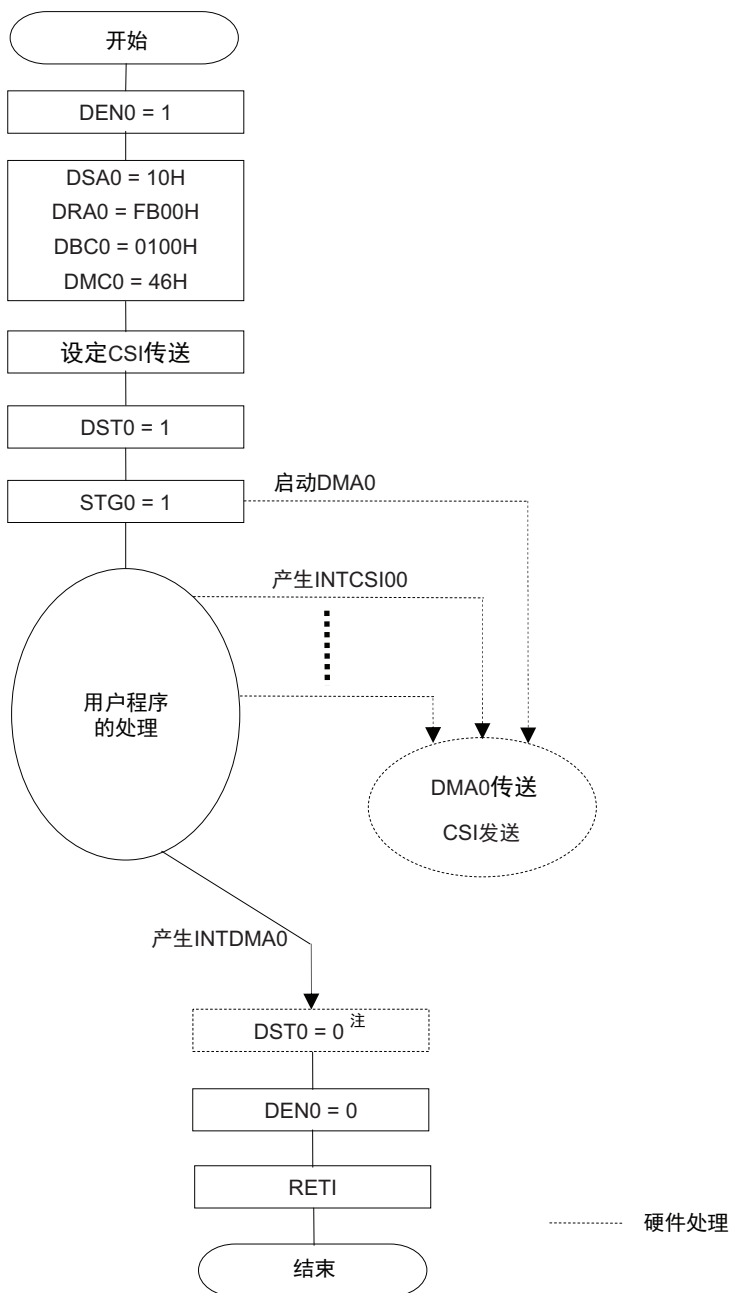
19.5.1 CSI 连续发送

CSI 连续发送的设定例子的流程图如下所示：

- CSI00 的连续发送（256 字节）
- DMA 传送使用 DMA 的通道 0。
- DMA 启动源：INTCSI00（软件触发（STG0）只限于最初的启动源）
- 通过 IFC03 ~ IFC00=0110B 指定 CSI00 的中断。
- 将 RAM 的 FFB00H ~ FFBFFH（256 字节）传送到 CSI 的数据寄存器（SIO00）的 FFF10H。

备注 IFC03 ~ IFC00：DMA 模式控制寄存器 0（DMC0）的 bit3 ~ 0

图 19-7 CSI 连续发送的设定例子



注 如果 DMA 传送结束，DST0 标志就自动清“0”。
因为只有在 DST0 标志为“0”时才允许写 DEN0 标志，所以当不等待 DMA0 中断（INTDMA0）的发生而结束 DMA 传送时，必须在将 DST0 标志置“0”后将 DEN0 标志置“0”（详细内容请参照“19.5.5 通过软件强制结束”）。

在连续发送时，CSI 中断不能启动第一次触发。在此例子中，通过软件触发进行启动。

自动进行第 2 次及以后的 CSI 发送。

在给数据寄存器写完最后的发送数据时，产生 DMA 中断（INTDMA0）。

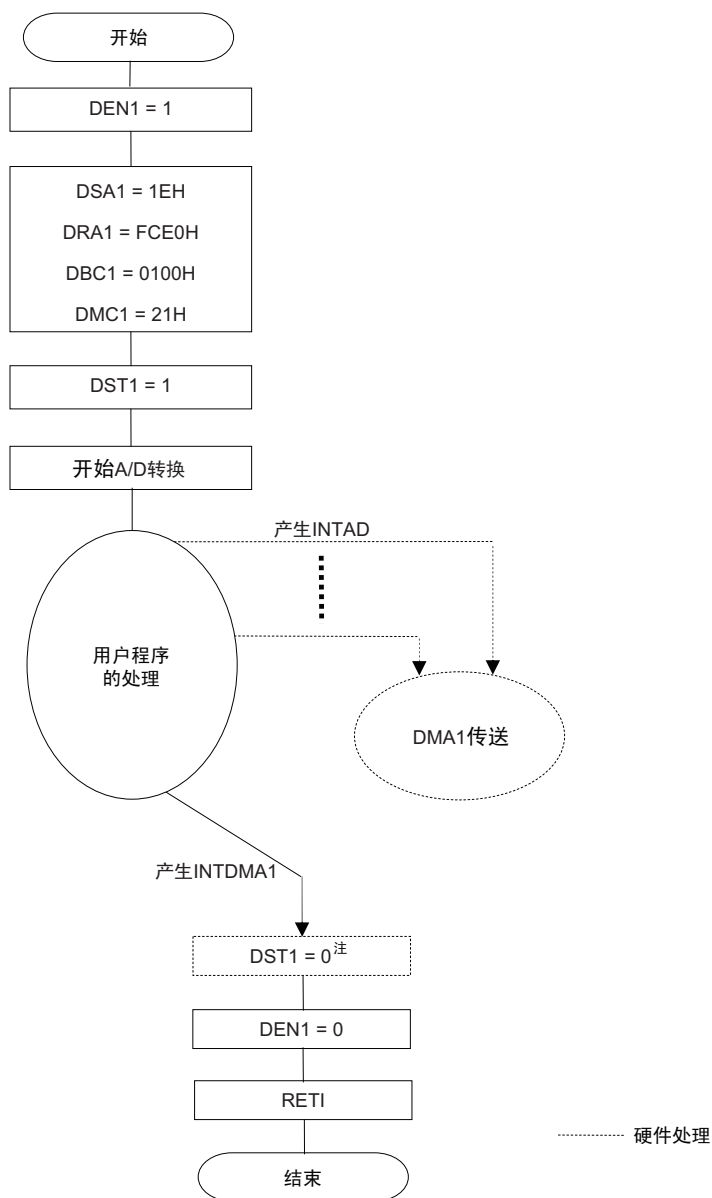
19.5.2 A/D 转换结果的连续读取

A/D 转换结果连续读取的设定例子的流程图如下所示：

- A/D 转换结果的连续读取
- DMA 传送使用 DMA 的通道 1。
- DMA 启动源：INTAD
- 通过 IFC13 ~ IFC10=0001B 指定 A/D 的中断
- 将 10 位 A/D 转换结果寄存器（ADCR）的 FFF1EH 和 FFF1FH（2 字节）传送到 RAM 的 FFCE0H ~ FFEDFH（512 字节）

备注 IFC13 ~ IFC10：DMA 模式控制寄存器 1（DMC1）的 bit3 ~ 0

图 19-8 A/D 转换结果连续读取的设定例子

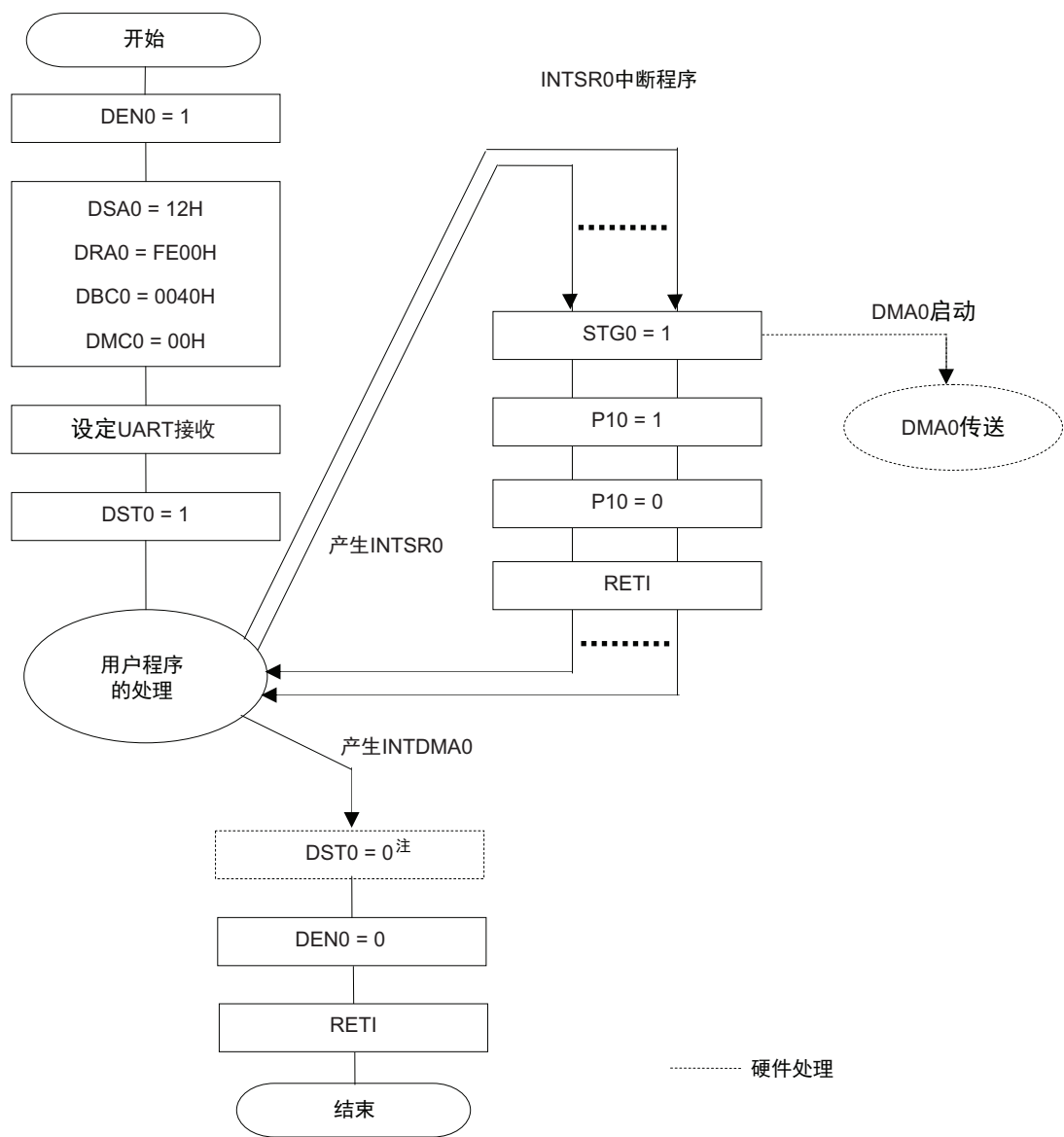


注 如果 DMA 传送结束，DST1 标志就自动清“0”。
因为只有在 DST1 标志为“0”时才允许写 DEN1 标志，所以当不等待 DMA1 中断（INTDMA1）的发生而结束 DMA 传送时，必须在将 DST1 标志置“0”后将 DEN1 标志置“0”（有关详细内容，请参照“19.5.5 通过软件强制结束”）。

19.5.3 UART 连续接收 +ACK 发送

- UART 连续接收 +ACK 发送的设定例子的流程图如下所示：
- UART0进行连续接收并且将接收结束的ACK输出到P10。
 - DMA 传送使用DMA 的通道0。
 - DMA 启动源：软件触发（禁止通过中断触发DMA 传送）
 - 将UART接收数据寄存器0（RXD0）的FFF12H传送到RAM的FFE00H～FFE3FH（64字节）。

图 19-9 UART 连续接收 +ACK 发送的设定例子



注 如果 DMA 传送结束，DST0 标志就自动清“0”。
因为只有在 DST0 标志为“0”时才允许写 DEN0 标志，所以当不等待 DMA0 中断（INTDMA0）的发生而结束 DMA 传送时，必须在将 DST0 标志置“0”后将 DEN0 标志置“0”（有关详细内容，请参照“19.5.5 通过软件强制结束”）。

备注 这是将软件触发用于 DMA 启动源的例子。
如果不发送 ACK 而只进行 UART 的连续接收，就能将 UART 接收结束中断（INTSR0）设定为 DMA 启动源进行数据接收。

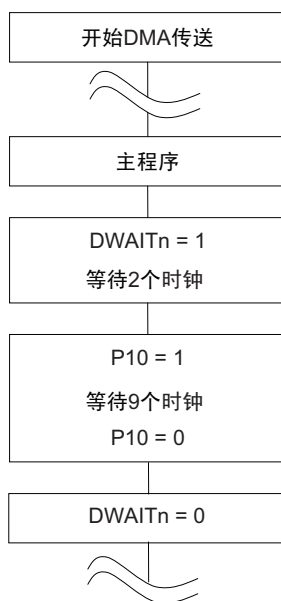
19.5.4 通过 DWAITn 位保留 DMA 传送

如果开始 DMA 传送，就在指令执行过程中进行传送。因此，此时 CPU 停止运行并且延迟 2 个时钟。如果这种情况会影响设备系统的正常运行，就能通过将 DWAITn 位置“1”，保留 DMA 传送。对于在保留期间发生的传送触发所对应的 DMA 传送，在解除保留后执行该 DMA 传送。但是，各通道只能保留一个传送触发，因此同一个通道即使在保留期间发生了 2 次或者更多次的传送触发，在解除保留后也只执行一次 DMA 传送。

以 P10 引脚输出 10 个时钟（工作频率）宽度的脉冲为例，如果在中途开始 DMA 传送，就增加到 12 个时钟。此时，能通过将 DWAITn 位置“1”来保留 DMA 传送。

在将 DWAITn 位置“1”后需要经过 2 个时钟，然后保留 DMA 传送。

图 19-10 通过 DWAITn 位保留 DMA 传送的设置例子



注意 要在使用 2 个或者更多通道 DMA 的过程中保留 DMA 传送时，必须保留全部通道的 DMA 传送（DWAIT0=DWAIT1=DWAIT2=DWAIT3=1）。如果在保留其他通道的 DMA 传送期间执行一个通道的 DMA 传送，就可能无法保留其他通道的 DMA。

备注 1. n: DMA 通道号（n=0 ~ 3）

2. 1 个时钟: $1/f_{\text{CLK}}$ （ f_{CLK} : CPU 时钟）

19.5.5 通过软件强制结束

在通过软件将 DSTn 位置“0”后需要经过 2 个时钟，然后停止 DMA 传送并且 DSTn 位变为“0”。因此，当不等待 DMA 中断（INTDMA_n）的发生而通过软件强制结束 DMA 传送时，必须进行以下某种处理。

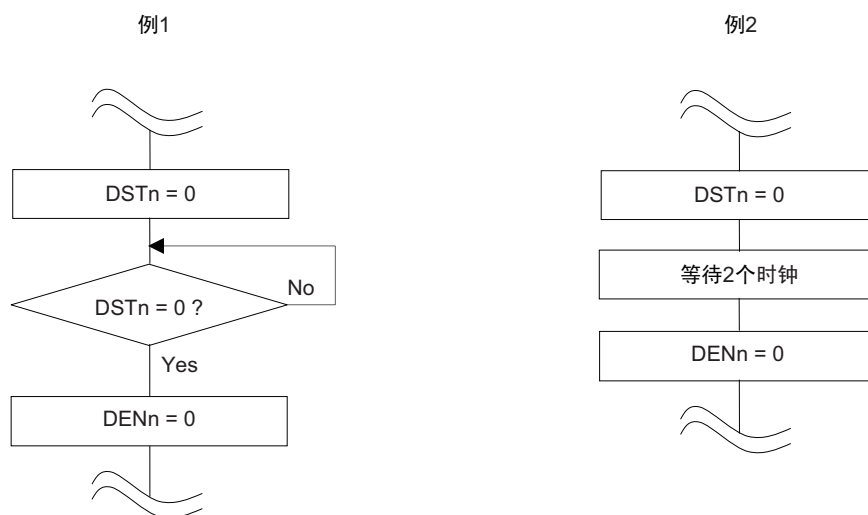
< 当使用 1 个通道的 DMA 时 >

- 在通过软件将 DSTn 位置“0”（当通过字节操作指令进行写操作时，DRCn=80H）后，通过轮询确认 DSTn 位已变为“0”，然后将 DENn 位置“0”（当通过字节操作指令进行写操作时，DRCn=00H）。
- 在通过软件将 DSTn 位置“0”（当通过字节操作指令进行写操作时，DRCn=80H）后经过 2 个时钟，然后将 DENn 位置“0”（当通过字节操作指令进行写操作时，DRCn=00H）。

< 当使用 2 个或者更多通道的 DMA 时 >

- 要在使用 2 个或者更多通道 DMA 的过程中通过软件强制结束 DMA 传送（DSTn=0）时，通过将所用全部通道的 DWAITn 位置“1”来保留 DMA 传送，然后将 DSTn 位清“0”。此后，通过将所用全部通道的 DWAITn 位清“0”来解除保留，然后将 DENn 位清“0”。

图 19-11 DMA 传送的强制结束（1/2）



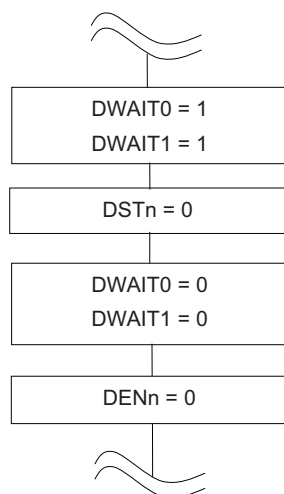
备注 1. n: DMA 通道号 (n=0 ~ 3)

2. 1 个时钟: $1/f_{CLK}$ (f_{CLK} : CPU 时钟)

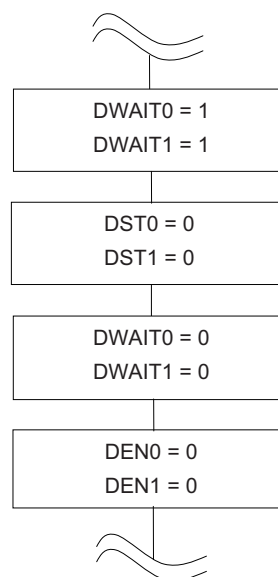
图 19-11 DMA 传送的强制结束 (2/2)

例3

- 在2个通道都使用时，其中1个通道强制结束的步骤



- 在2个通道都使用时，2个通道都强制结束的步骤



注意 例 3 中，在将 DWAITn 位置“1”后不需要等待 2 个时钟。因为在从 DSTn 位清“0”到 DENn 位清“0”的期间已经至少经过了 2 个时钟，所以在将 DSTn 位清“0”后不需要等待 2 个时钟。

备注 1. n: DMA 通道号 (n=0、1)

2. 1 个时钟: $1/f_{CLK}$ (f_{CLK} : CPU 时钟)

19.6 DMA 控制器的注意事项

(1) DMA 的优先级

在 DMA 传送过程中，即使发生其他 DMA 通道的请求，也保留该通道的请求。在 DMA 传送结束后开始被保留的 DMA 传送。但是，当多个 DMA 请求同时发生时，优先级为 DMA 通道 0 > DMA 通道 1 > DMA 通道 2 > DMA 通道 3。

当 DMA 请求和中断请求同时发生时，优先 DMA 传送，然后执行中断处理。

(2) DMA 应答时间

DMA 传送的应答时间如下所示。

表 19-3 DMA 传送的应答时间

	最短时间	最长时间
应答时间	3 个时钟	10 个时钟注

注 当从内部 RAM 执行指令时，最长时间为 16 个时钟。

注意 1. 以上的应答时间不包括 DMA 传送的 2 个时钟。

2. 在执行 DMA 保留指令（参照 19.6(4)）时，各条件的最长应答时间加上在其条件下保留指令的执行时间。

3. 最长应答时间 +1 个时钟内的同一个通道的连续传送触发可能被忽视，因此不能设定。

备注 1 个时钟：1/f_{CLK}（f_{CLK}：CPU 时钟）

(3) 待机模式中的运行

待机模式中的 DMA 控制器的运行如下所示。

表 19-4 待机模式中的 DMA 运行

状态	DMA 运行
HALT 模式	通常运行
STOP 模式	停止运行 当 DMA 传送和 STOP 指令发生竞争时，DMA 传送可能会遭到破坏，因此必须在执行 STOP 指令前停止 DMA。

(4) DMA 传送的保留指令

即使发生 DMA 请求，也会在紧随着以下指令之后保留 DMA 传送。

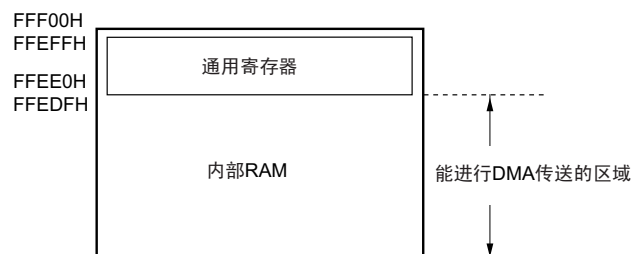
- CALL !addr16
- CALL \$!addr20
- CALL !!addr20
- CALL rp
- CALLT [addr5]
- BRK
- MOV PSW, #byte
- MOV PSW, A
- MOV1 PSW.bit, CY
- SET1 PSW.bit
- CLR1 PSW.bit
- POP PSW
- BTCLR PSW.bit, \$addr20
- EI
- DI
- IF0L、IF0H、IF1L、IF1H、IF2L、IF2H、IF3L、MK0L、MK0H、MK1L、MK1H、MK2L、MK2H、MK3L、PR00L、PR00H、PR01L、PR01H、PR02L、PR02H、PR03L、PR10L、PR10H、PR11L、PR11H、PR12L、PR12H和PR13L的各个寄存器的写指令
- 存取数据闪存的指令

(5) 指定通用寄存器区内或者内部 RAM 区外的地址时的运行

在 DMA 传送过程中，DMA RAM 地址寄存器 n（DRAn）所示的地址进行递增。当该地址进入通用寄存器区内或者超过内部 RAM 区时，会出现以下情况。

- 从SFR传送到RAM的模式
破坏该地址的数据。
- 从RAM传送到SFR的模式
将不定的数据传送到SFR。

以上2种情况都可能导致误动作或者系统的损坏，因此必须确保地址在通用寄存器区以外的内部RAM区内。



第 20 章 中断功能

中断功能是指在程序执行过程中将处理程序切换为其他所需处理程序的一种功能。在转移处理结束后返回到原来中断的执行程序。

20.1 中断功能的种类

中断功能有以下两种。

(1) 可屏蔽中断

这是接受屏蔽控制的中断。通过设定优先级指定标志寄存器（PR00L、PR00H、PR01L、PR01H、PR02L、PR02H、PR03L、PR10L、PR10H、PR11L、PR11H、PR12L、PR12H、PR13L），能将可屏蔽中断的优先级分成四组。高优先级的中断能对低优先级中断进行多重中断处理。如果同时发生相同优先级的多个中断请求，就根据向量中断处理的默认优先级进行处理。有关默认优先级，请参照表 20-1。

产生待机解除信号，解除 STOP 模式、HALT 模式和 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。即使在禁止中断的状态下也接受软件中断，而且软件中断不受中断优先级控制。

20.2 中断源和结构

中断源除了有可屏蔽中断和软件中断以外还有最多 7 种复位源（参照表 20-1）。当发生复位或者各种中断请求而进行转移时，因为保存程序起始地址的向量代码各为两个字节，所以中断的转移目标地址为 00000H ~ 0FFFFH 的 64K 地址。

表 20-1 中断源一览表 (1/3)

中断处理	默认优先级注 1	中断源		内部 / 外部	向量表地址	基本结构类型注 2
		名称	触发			
可屏蔽	0	INTWDTI	看门狗定时器间隔 注 3 (上溢时间的 75%+1/2f _{IL})	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	INTP1			000AH	
	4	INTP2			000CH	
	5	INTP3			000EH	
	6	INTP4			0010H	
	7	INTP5			0012H	
	8	INTST2	UART2 发送的传送结束或者缓冲器空中断	内部	0014H	(A)
	9	INTSR2	UART2 接收的传送结束		0016H	
	10	INTSRE2	发生 UART2 接收的通信错误		0018H	
	11	INTDMA0	DMA0 的传送结束		001AH	
	12	INTDMA1	DMA1 的传送结束		001CH	
	13	INTST0 /INTCSI00/INTIIC00	UART0 发送的传送结束或者缓冲器空中断 /CSI00 的传送结束或者缓冲器空中断 /IIC00 的传送结束		001EH	
	14	INTTM00	定时器通道 00 的计数结束或者捕捉结束		0020H	
	15	INTSR0	UART0 接收的传送结束		0022H	
	16	INTSRE0	发生 UART0 接收的通信错误		0024H	
		INTTM01H	定时器通道 01 的计数或者捕捉结束 (高 8 位定时器运行时)			
	17	INTST1/INTIIC10	UART1 发送的传送结束或者缓冲器空中断 /IIC10 的传送结束		0026H	
	18	INTSR1	UART1 接收的传送结束		0028H	
	19	INTSRE1	发生 UART1 接收的通信错误		002AH	
		INTTM03H	定时器通道 03 的计数结束或者捕捉结束 (高 8 位定时器运行时)			
	20	INTIICA0	IICA0 通信结束		002CH	
	21	INTRTIT	RTC 校正时序		002EH	
	22	INTFM	频率测量结束		0030H	
	23	INTTM01	定时器通道 01 的计数结束或者捕捉结束 (16 位 / 低 8 位定时器运行时)		0032H	

注 1. 在发生多个可屏蔽中断时, 默认优先级决定优先顺序。0 表示最高优先级, 49 表示最低优先级。

2. 基本构成类型 (A) ~ (C) 分别对应图 20-1 的 (A) ~ (C)。

3. 当将选项字节 (000C0H) 的 bit7 (WDTINT) 置“1”时。

4. 当将电压检测电平寄存器 (LVIS) 的 bit7 (LVIMD) 置“0”时。

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	基本结构类型注 2
		名称	触发			
可屏蔽	24	INTTM02	定时器通道 02 的计数或者捕捉结束	内部	0034H	(A)
	25	INTTM03	定时器通道 03 的计数结束或者捕捉结束 (16 位 / 低 8 位定时器工作时)		0036H	
	26	INTAD	A/D 转换结束		0038H	
	27	INTRTC	检测实时时钟 2 的固定周期信号 / 闹钟一致检测		003AH	
	28	INTIT	12 位间隔定时器的间隔信号检测		003CH	
	29	INTST3	UART3 发送的传送结束或者缓冲器空中断		0040H	
	30	INTSR3	UART3 接收的传送结束		0042H	
	31	INTTM04	定时器通道 04 的计数结束或者捕捉结束		0046H	
	32	INTTM05	定时器通道 05 的计数结束或者捕捉结束		0048H	
	33	INTP6	检测引脚输入边沿	外部	004AH	(B)
	34	INTP7			004CH	
	35	INTCMP0	比较器检测 0		0050H	
	36	INTCMP1	比较器检测 1		0052H	
	37	INTTM06	定时器通道 06 的计数结束或者捕捉结束	内部	0054H	(A)
	38	INTTM07	定时器通道 07 的计数结束或者捕捉结束		0056H	
	39	INTSCT0	SMCI0 发送的传送结束		0058H	
	40	INTSCR0	SMCI0 接收数据满		005AH	
	41	INTSRE3	发生 UART3 接收的通信错误		005CH	
	42	INTMD	除法运算结束 / 发生上溢		005EH	
	43	INTSCE0	发生 SMCI0 接收的通信错误 / 检测到错误信号		0060H	
	44	INTFL	定序器结束中断注 3		0062H	
	45	INTDMA2	DMA2 的传送结束		0064H	
	46	INTDMA3	DMA3 的传送结束		0066H	
	47	INTSCT1	SMCI1 发送的传送结束		0068H	
	48	INTSCR1	SMCI1 接收数据满		006AH	
	49	INTSCE1	发生 SMCI1 接收的通信错误 / 检测到错误信号		006CH	

- 注 1. 在发生多个可屏蔽中断时，默认优先级决定优先顺序。0 表示最高优先级，49 表示最低优先级。
2. 基本构成类型 (A) ~ (C) 分别对应图 20-1 的 (A) ~ (C)。
3. 在闪存自编程库中使用。

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	基本结构类型注 2
		名称	触发			
软件	—	BRK	执行 BRK 指令	—	007EH	(C)
复位	—	RESET	RESET 引脚输入	—	0000H	—
		POR	上电复位			
		LVD	电压检测 注 3			
		WDT	看门狗定时器的上溢			
		TRAP	非法指令的执行注 4			
		IAW	存取非法存储器			
		RPE	RAM 奇偶校验错误			

注 1. 在发生多个可屏蔽中断时，默认优先级决定优先顺序。0 表示最高优先级，49 表示最低优先级。

2. 基本构成类型 (A) ~ (C) 分别对应图 20-1 的 (A) ~ (C)。

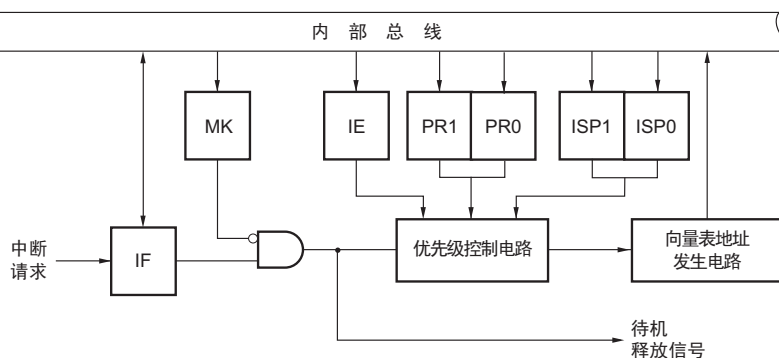
3. 当将电压检测电平寄存器 (LVIS) 的 bit7 (LVIMD) 置“1”时。

4. 在执行指令码 FFH 时发生。

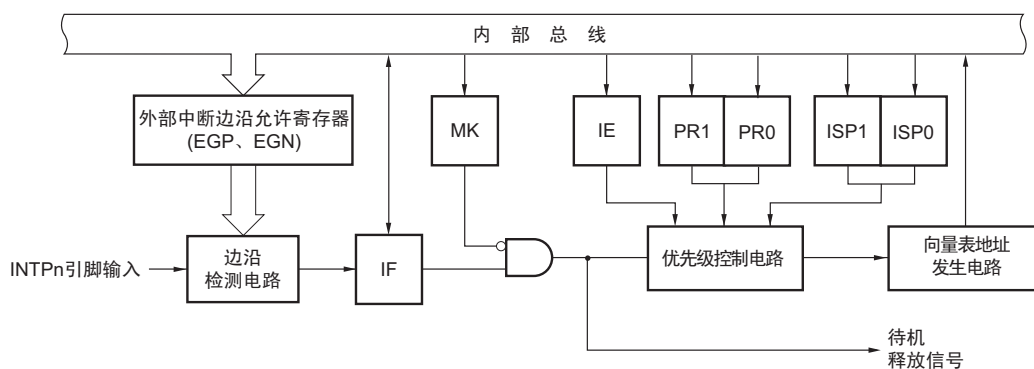
在通过在线仿真器或者片上调试仿真器进行仿真时，不会因执行非法指令而产生复位。

图 20-1 中断功能的基本结构

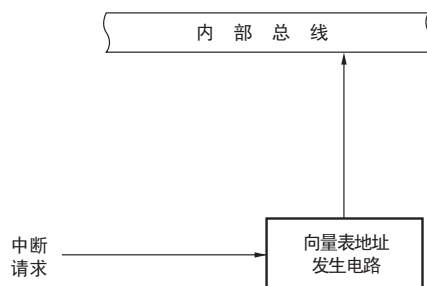
(A) 内部可屏蔽中断



(B) 外部可屏蔽中断 (INTPn)



(C) 软件中断



IF: 中断请求标志
 IE: 中断允许标志
 ISP0: 优先级控制标志 0
 ISP1: 优先级控制标志 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 n = 0 ~ 7

20.3 控制中断功能的寄存器

通过以下 6 种寄存器控制中断功能。

- 中断请求标志寄存器（IF0L、IF0H、IF1L、IF1H、IF2L、IF2H、IF3L）
- 中断屏蔽标志寄存器（MK0L、MK0H、MK1L、MK1H、MK2L、MK2H、MK3L）
- 优先级指定标志寄存器（PR00L、PR00H、PR01L、PR01H、PR02L、PR02H、PR03L、PR10L、PR10H、PR11L、PR11H、PR12L、PR12H、PR13L）
- 外部中断上升沿允许寄存器（EGP0）
- 外部中断下降沿允许寄存器（EGN0）
- 程序状态字（PSW）

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 中断请求源对应的各标志（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		寄存器		寄存器
INTWDTI	WDTIIF	IF0L	WDTIMK	MK0L	WDTIPR0、WDTIPR1	PR00L、PR10L
INTLVI	LVIIIF		LVIMK		LVIPR0、LVIPR1	
INTP0	PIF0		PMK0		PPR00、PPR10	
INTP1	PIF1		PMK1		PPR01、PPR11	
INTP2	PIF2		PMK2		PPR02、PPR12	
INTP3	PIF3		PMK3		PPR03、PPR13	
INTP4	PIF4		PMK4		PPR04、PPR14	
INTP5	PIF5		PMK5		PPR05、PPR15	
INTST2	STIF2	IF0H	STMK2	MK0H	STPR02、STPR12	PR00H、PR10H
INTSR2	SRIF2		SRMK2		SRPR02、SRPR12	
INTSRE2	SREIF2		SREMK2		SREPR02、SREPR12	
INTDMA0	DMAIF0		DMAMK0		DMAPR00、DMAPR10	
INTDMA1	DMAIF1		DMAMK1		DMAPR01、DMAPR11	
INTST0 注	STIF0 注		STMK0 注		STPR00、STPR10 注	
INTCSI00 注	CSIIIF00 注		CSIMK00 注		CSIPR000、CSIPR100 注	
INTIIC00 注	IICIF00 注		IICMK00 注		IICPR000、IICPR100 注	
INTTM00	TMIF00		TMMK00		TMPR000、TMPR100	
INTSR0	SRIF0		SRMK0		SRPR00、SRPR10	

注 如果产生 INTST0、INTCSI00 或者 INTIIC000 的中断源，IF0H 寄存器的 bit5 就被置“1”。MK0H、PR00H、PR10H 寄存器的 bit5 对应这 3 个中断源。

表 20-2 中断请求源对应的各标志 (2/3)

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		寄存器		寄存器
INTSRE0 注 1	SREIF0 注 1	IF1L	SREMK0 注 1	MK1L	SREPR00、SREPR10 注 1	PR01L、PR11L
INTTM01H 注 1	TMIF01H 注 1		TMMK01H 注 1		TMPR001H、TMPR101H 注 1	
INTST1 注 2	STIF1 注 2		STMK1 注 2		STPR01、STPR11 注 2	
INTIIC10 注 2	IICIF10 注 2		IICMK10 注 2		IICPR010、IICPR110 注 2	
INTSR1	SRIF1		SRMK1		SRPR01、SRPR11	
INTSRE1 注 3	SREIF1 注 3		SREMK1 注 3		SREPR01、SREPR11 注 3	
INTTM03H 注 3	TMIF03H 注 3		TMMK03H 注 3		TMPR003H、TMPR103H 注 3	
INTIICA0	IICAIF0		IICAMK0		IICAPR00、IICAPR10	
INTRTIT	RTITIF		RTITMK		RTITPR0、RTITPR1	
INTFM	FMIF		FMMK		FMPR0、FMPR1	
INTTM01	TMIF01		TMMK01		TMPR001、TMPR101	
INTTM02	TMIF02	IF1H	TMMK02	MK1H	TMPR002、TMPR102	PR01H、PR11H
INTTM03	TMIF03		TMMK03		TMPR003、TMPR103	
INTAD	ADIF		ADMK		ADPR0、ADPR1	
INTRTC	RTCIF		RTCMK		RTCPR0、RTCPR1	
INTIT	TMKAIF		TMKAMK		TMKAPR0、TMKAPR1	
INTST3	STIF3		STMK3		STPR03、STPR13	
INTSR3	SRIF3		SRMK3		SRPR03、SRPR13	
INTTM04	TMIF04	IF2L	TMMK04	MK2L	TMPR004、TMPR104	PR02L、PR12L
INTTM05	TMIF05		TMMK05		TMPR005、TMPR105	
INTP6	PIF6		PMK6		PPR06、PPR16	
INTP7	PIF7		PMK7		PPR07、PPR17	
INTCMP0	CMPIF0		CMPMK0		CMPPR00、CMPPR10	
INTCMP1	CMPIF1		CMPMK1		CMPPR01、CMPPR11	

- 注 1. 因为 UART0 接收的错误中断和 TAU0 的通道 1（高 8 位定时器运行时）的中断复用中断请求源对应的各种标志，所以不能同时使用。在不使用 UART0 接收的错误中断（EOC01=0）时，可同时使用 UART0、TAU0 的通道 1（高 8 位定时器运行时）。如果产生 INTSRE0 或者 INTTM01H 的中断源，IF1L 寄存器的 bit0 就被置“1”。MK1L、PR01L、PR11L 寄存器的 bit0 对应这 2 个中断源。
2. 如果产生 INTST1、INTIIC10 的中断源，IF1L 寄存器的 bit1 就被置“1”。MK1L、PR01L、PR11L 寄存器的 bit1 对应这 2 个中断源。
3. 因为 UART1 接收的错误中断和 TAU0 的通道 3（高 8 位定时器运行时）的中断复用中断请求源对应的各种标志，所以不能同时使用。在不使用 UATR1 接收的错误中断（EOC03=0）时，可同时使用 UART1、TAU0 的通道 3（高 8 位定时器运行时）。如果发生 INTSRE1 或者 INTTM03H 的中断源，IF1L 寄存器的 bit3 就被置“1”。MK1L、PR01L、PR11L 寄存器的 bit3 对应这 2 个中断源。

表 20-2 中断请求源对应的各标志 (3/3)

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		寄存器		寄存器
INTTM06	TMIF06	IF2H	TMMK06	MK2H	TMPR006、TMPR106	PR02H、 PR12H
INTTM07	TMIF07		TMMK07		TMPR007、TMPR107	
INTSCT0	SCTIF0		SCTMK0		SCTPR00、SCTPR10	
INTSCR0	SCRIF0		SCRMK0		SCRPR00、SCRPR10	
INTSRE3	SREIF3		SREMK3		SREPR03、SREPR13	
INTMD	MDIF		MDMK		MDPR0、MDPR1	
INTSCE0	SCEIF0		SCEMK0		SCEPR00、SCEPR10	
INTFL	FLIF		FLMK		FLPR0、FLPR1	
INTDMA2	DMAIF2	IF3L	DMAMK2	MK3L	DMAPR02、DMAPR12	PR03L、 PR13L
INTDMA3	DMAIF3		DMAMK3		DMAPR03、DMAPR13	
INTSCT1	SCTIF1		SCTMK1		SCTPR01、SCTPR11	
INTSCR1	SCRIF1		SCRMK1		SCRPR01、SCRPR11	
INTSCE1	SCEIF1		SCEMK1		SCEPR01、SCEPR11	

20.3.1 中断请求标志寄存器 (IF0L、IF0H、IF1L、IF1H、IF2L、IF2H、IF3L)

通过产生对应的中断请求或者执行指令，将中断请求标志置“1”。通过接受中断请求或者产生复位信号或者执行指令，将中断请求标志清“0”。

在接受中断时，首先自动清除中断请求标志，然后进入中断程序。

通过 1 位或者 8 位存储器操作指令设定 IF0L、IF0H、IF1L、IF1H、IF2L、IF2H、IF3L 寄存器。当将 IF0L 寄存器和 IF0H 寄存器、IF1L 寄存器和 IF1H 寄存器以及 IF2L 寄存器和 IF2H 寄存器一起用作 IF0、IF1、IF2 的 16 位寄存器时，通过 16 位存储器操作指令进行设定。

在产生复位信号后，这些寄存器的值变为“00H”。

备注 当执行写此寄存器的指令时，指令执行时钟数增加 2 个时钟。

图 20-2 中断请求标志寄存器 (IF0L、IF0H、IF1L、IF1H、IF2L、IF2H、IF3L) 的格式 (1/2)

地址: FFFE0H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
IF0L	PIF5	PIF4	PIF3	PIF2	PIF1	PIF0	LVIIIF	WDTIIF

地址: FFFE1H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
IF0H	SRIF0	TMIF00	STIF0 CSIIF00 IICIF00	DMAIF1	DMAIF0	SREIF2	SRIF2	STIF2

地址: FFFE2H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
IF1L	TMIF01	FMIF	RTITIF	IICAIF0	SREIF1 TMIF03H	SRIF1	STIF1 IICIF10	SREIF0 TMIF01H

地址: FFFE3H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
IF1H	SRIF3	STIF3	0	TMKAIF	RTCIF	ADIF	TMIF03	TMIF02

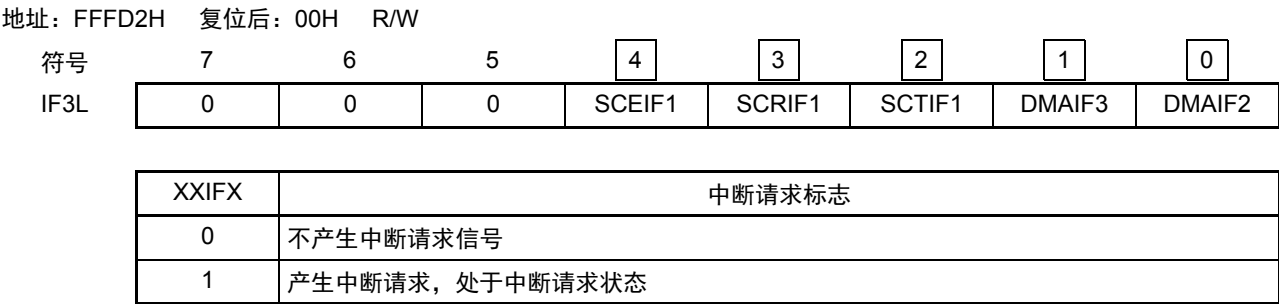
地址: FFFD0H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
IF2L	CMPIF1	CMPIF0	0	PIF7	PIF6	TMIF05	TMIF04	0

地址: FFFD1H 复位后: 00H R/W

符号	7	6	5	4	3	2	1	0
IF2H	FLIF	SCEIF0	MDIF	SREIF3	SCRIF0	SCTIF0	TMIF07	TMIF06

图 20-2 中断请求标志寄存器（IF0L、IF0H、IF1L、IF1H、IF2L、IF2H、IF3L）的格式（2/2）



注意 1. 必须将未配置的位设定为“0”。

2. 在操作中断请求标志寄存器的标志时，必须使用 1 位存储器操作指令（CLR1）。当使用 C 语言描述时，编译后的汇编程序需要变为 1 位存储器操作指令（CLR1），因此必须使用类似“IF0L.0=0;”或者“_asm("clr1 IF0L, 0");”的位操作指令。

如果在 C 语言中使用类似于“IF0L &= 0xfe;”的 8 位存储器操作指令描述的情况下进行编译，就变为 3 条指令的汇编程序。

```
mov a, IF0L
and a, #0FEH
mov IF0L, a
```

此时，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）的其他位的请求标志置“1”，该标志也会被“mov IF0L, a”清“0”。因此，在 C 语言中使用 8 位存储器操作指令时，必须注意。

20.3.2 中断屏蔽标志寄存器（MK0L、MK0H、MK1L、MK1H、MK2L、MK2H、MK3L）

中断屏蔽标志设定允许或者禁止对应的可屏蔽中断。

通过 1 位或者 8 位存储器操作指令设定 MK0L、MK0H、MK1L、MK1H、MK2L、MK2H、MK3L 寄存器。当将 MK0L 寄存器和 MK0H 寄存器、MK1L 寄存器和 MK1H 寄存器以及 MK2L 寄存器和 MK2H 寄存器一起用作 MK0、MK1、MK2 的 16 位寄存器时，通过 16 位存储器操作指令进行设定。

在产生复位信号后，这些寄存器的值变为“FFH”。

备注 当执行写此寄存器的指令时，指令执行时钟数增加 2 个时钟。

图 20-3 中断屏蔽标志寄存器（MK0L、MK0H、MK1L、MK1H、MK2L、MK2H、MK3L）的格式

地址: FFFE4H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
MK0L	PMK5	PMK4	PMK3	PMK2	PMK1	PMK0	LVIMK	WDTIMK

地址: FFFE5H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
MK0H	SRMK0	TMMK00	STMK0 CSIMK00 IICMK00	DMAMK1	DMAMK0	SREMK2	SRMK2	STMK2

地址: FFFE6H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
MK1L	TMMK01	FMMK	RTITMK	IICAMK0	SREMK1 TMMK03H	SRMK1	STMK1 IICMK10	SREMK0 TMMK01H

地址: FFFE7H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
MK1H	SRMK3	STMK3	1	TMKAMK	RTCMK	ADMK	TMMK03	TMMK02

地址: FFFD4H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
MK2L	CMPMK1	CMPMK0	1	PMK7	PMK6	TMMK05	TMMK04	1

地址: FFFD5H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
MK2H	FLMK	SCEMK0	MDMK	SREMK3	SCRMK0	SCTMK0	TMMK07	TMMK06

地址: FFFD6H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
MK3L	1	1	1	SCEMK1	SCRMK1	SCTMK1	DMAMK3	DMAMK2

XXMKX	中断处理的控制						
0	允许中断处理						
1	禁止中断处理						

注意 必须将未配置的位设定为“1”。

20.3.3 优先级指定标志寄存器 (PR00L、PR00H、PR01L、PR01H、PR02L、PR02H、PR03L、PR10L、PR10H、PR11L、PR11H、PR12L、PR12H、PR13L)

优先级指定标志寄存器设定对应的可屏蔽中断优先级。

通过组合 PR0xy 寄存器和 PR1xy 寄存器来设定优先级 (xy=0L、0H、1L、1H、2L、2H)。

通过 1 位或者 8 位存储器操作指令设定 PR00L、PR00H、PR01L、PR01H、PR02L、PR02H、PR03L、PR10L、PR10H、PR11L、PR11H、PR12L、PR12H、PR13L 寄存器。当将 PR00L 寄存器和 PR00H 寄存器、PR01L 寄存器和 PR01H 寄存器、PR02L 寄存器和 PR02H 寄存器、PR10L 寄存器和 PR10H 寄存器、PR11L 寄存器和 PR11H 寄存器以及 PR12L 寄存器和 PR12H 寄存器一起用作 PR00、PR01、PR02、PR10、PR11、PR12 的 16 位寄存器时，通过 16 位存储器操作指令进行设定。

在产生复位信号后，这些寄存器的值变为“FFH”。

备注 当执行写此寄存器的指令时，指令执行时钟数增加 2 个时钟。

图 20-4 优先级指定标志寄存器 (PR00L、PR00H、PR01L、PR01H、PR02L、PR02H、PR03L、PR10L、PR10H、PR11L、PR11H、PR12L、PR12H、PR13L) 的格式 (1/2)

地址: FFFE8H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PR00L	PPR05	PPR04	PPR03	PPR02	PPR01	PPR00	LVIPR0	WDTIPR0

地址: FFFECH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PR10L	PPR15	PPR14	PPR13	PPR12	PPR11	PPR10	LVIPR1	WDTIPR1

地址: FFFE9H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PR00H	SRPR00	TMPR000	STPR00 CSIPR000 CSIPR000	DMAPR01	DMAPR00	SREPR02	SRPR02	STPR02

地址: FFFEDH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PR10H	SRPR10	TMPR100	STPR10 CSIPR100 IICPR100	DMAPR11	DMAPR10	SREPR12	SRPR12	STPR12

地址: FFFEAH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PR01L	TMPR001	FMPR0	RTITPR0	IICAPR00	SREPR01 TMPR003H	SRPR01	STPR01 IICPR010	SREPR00 TMPR001H

地址: FFEEH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PR11L	TMPR101	FMPR1	RTITPR1	IICAPR10	SREPR11 TMPR103H	SRPR11	STPR11 IICPR110	SREPR10 TMPR101H

图 20-4 优先级指定标志寄存器 (PR00L、PR00H、PR01L、PR01H、PR02L、PR02H、PR03L、PR10L、PR10H、PR11L、PR11H、PR12L、PR12H、PR13L) 的格式 (2/2)

地址: FFFEBH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PR01H	SRPR03	STPR03	1	TMKAPR0	RTCPR0	ADPR0	TMPR003	TMPR002

地址: FFFE FH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PR11H	SRPR13	STPR13	1	TMKAPR1	RTCPR1	ADPR1	TMPR103	TMPR102

地址: FFFD8H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PR02L	CMPPR01	CMPPR00	1	PPR07	PPR06	TMPR005	TMPR004	1

地址: FFFDCH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PR12L	CMPPR11	CMPPR10	1	PPR17	PPR16	TMPR105	TMPR104	1

地址: FFFD9H 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PR02H	FLPR0	SCEPR00	MDPR0	SREPR03	SCRPR00	SCTPR00	TMPR007	TMPR006

地址: FFFDDH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PR12H	FLPR1	SCEPR10	MDPR1	SREPR13	SCRPR10	SCTPR10	TMPR107	TMPR106

地址: FFFDAH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PR03L	1	1	1	SCEPR01	SCRPR01	SCTPR01	DMAPR03	DMAPR02

地址: FFFDEH 复位后: FFH R/W

符号	7	6	5	4	3	2	1	0
PR13L	1	1	1	SCEPR11	SCRPR11	SCTPR11	DMAPR13	DMAPR12

XXPR1X	XXPR0X	选择优先等级
0	0	指定优先级 0 (高优先级)
0	1	指定优先 1
1	0	指定优先 2
1	1	指定优先 3 (低优先级)

注意 必须将未配置的位设定为“1”。

20.3.4 外部中断上升沿允许寄存器（EGP0）、外部中断下降沿允许寄存器（EGN0）

这些寄存器设定 INTP0 ~ INTP11 的有效边沿。

通过 1 位或者 8 位存储器操作指令设定 EGP0、EGN0 寄存器。

在产生复位信号后，这些寄存器的值变为“00H”。

图 20-5 外部中断上升沿允许寄存器（EGP0）和外部中断下降沿允许寄存器的格式（EGN0）

地址：FFF38H 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
EGP0	EGP7	EGP6	EGP5	EGP4	EGP3	EGP2	EGP1	EGP0

地址：FFF39H 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
EGN0	EGN7	EGN6	EGN5	EGN4	EGN3	EGN2	EGN1	EGN0

EGPn	EGNn	选择 INTPn 引脚的有效边沿 (n = 0 ~ 7)
0	0	禁止检测边沿
0	1	下降沿
1	0	上升沿
1	1	上升沿和下降沿的双边沿

EGPn 位和 EGNn 位对应的端口如表 20-3 所示。

表 20-3 EGPn 位和 EGNn 位对应的中断请求信号

检测允许位		中断请求信号
EGP0	EGN0	INTP0
EGP1	EGN1	INTP1
EGP2	EGN2	INTP2
EGP3	EGN3	INTP3
EGP4	EGN4	INTP4
EGP5	EGN5	INTP5
EGP6	EGN6	INTP6
EGP7	EGN7	INTP7

注意 如果要从外部中断功能切换至端口功能，就有可能进行边沿检测，因此必须在将 EGPn 位和 EGNn 位设定为“0”后，切换至端口模式。

备注 1. 有关边沿检测端口，请参照“2.1 端口功能”。

2. n = 0 ~ 7

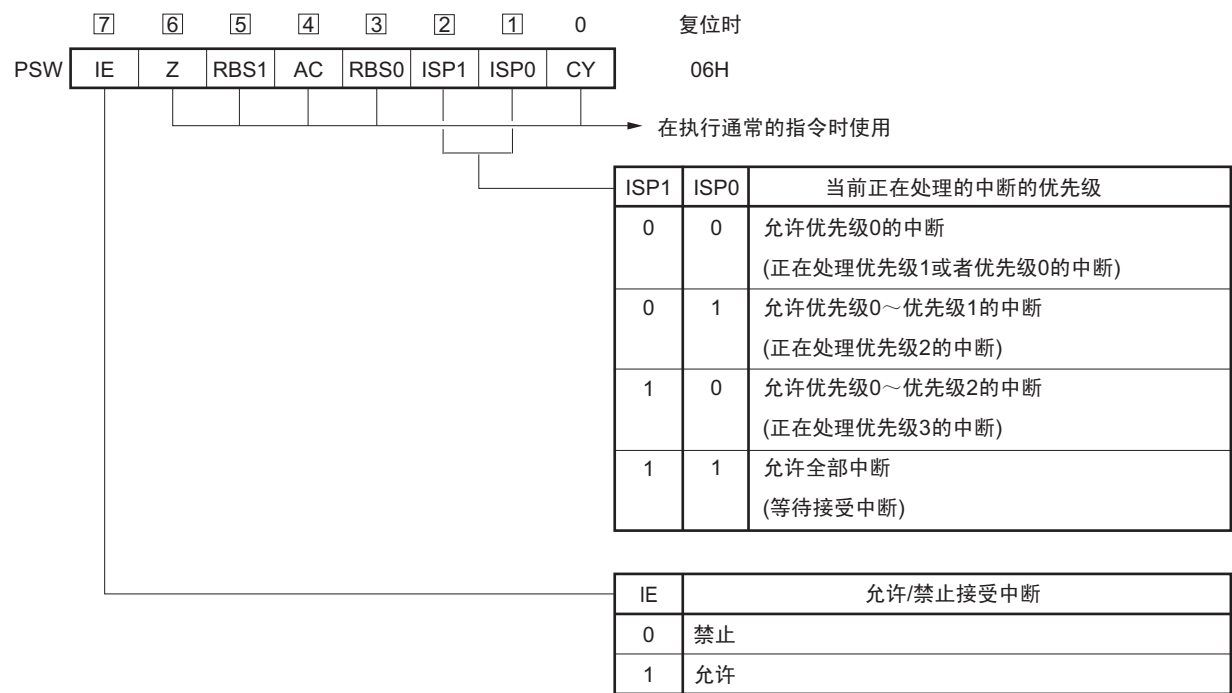
20.3.5 程序状态字（PSW）

程序状态字是保存指令执行结果和中断请求当前状态的寄存器，映射设定允许或者禁止可屏蔽中断的 IE 标志以及控制多重中断处理的 ISP0 标志和 ISP1 标志。

除了能以 8 位为单位读写 PSW 以外，还能利用位操作指令和专用指令（EI 和 DI）操作 PSW。在接受向量中断请求及执行 BRK 指令时，自动将 PSW 的内容压栈，并且将 IE 标志恢复为“0”。在接受可屏蔽中断请求时，将接受的中断优先级指定标志寄存器的内容（“00”以外）“- 1”后的值传送到 ISP0 标志和 ISP1 标志。PSW 的内容也能通过 PUSH PSW 指令进行压栈，通过 RETI、RETB、POP PSW 指令进行退栈。

在产生复位信号后，PSW 的值变为“06H”。

图 20-6 程序状态字的结构



20.4 中断处理的操作

20.4.1 接受可屏蔽中断请求

如果中断请求标志被置“1”并且该中断请求的屏蔽（MK）标志已被清“0”，就进入能接受可屏蔽中断请求的状态，在中断允许状态（IE 标志已被置“1”）下接受向量中断请求。但是，在处理优先级高的中断请求的过程中，不接受被指定的低优先级的中断请求。

从产生可屏蔽中断请求到进行向量中断处理的时间如表 20-4 所示。

有关中断请求的接受时序，请参照图 20-8 和图 20-9。

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 区的指令时除外。

备注 1 个时钟：1/f_{CLK}（f_{CLK}：CPU 时钟）

如果同时发生多个可屏蔽中断请求，就从优先级指定标志所指定的高优先级的请求开始接受。如果优先级指定标志所指定的优先级相同，就从默认优先级高的请求开始接受。

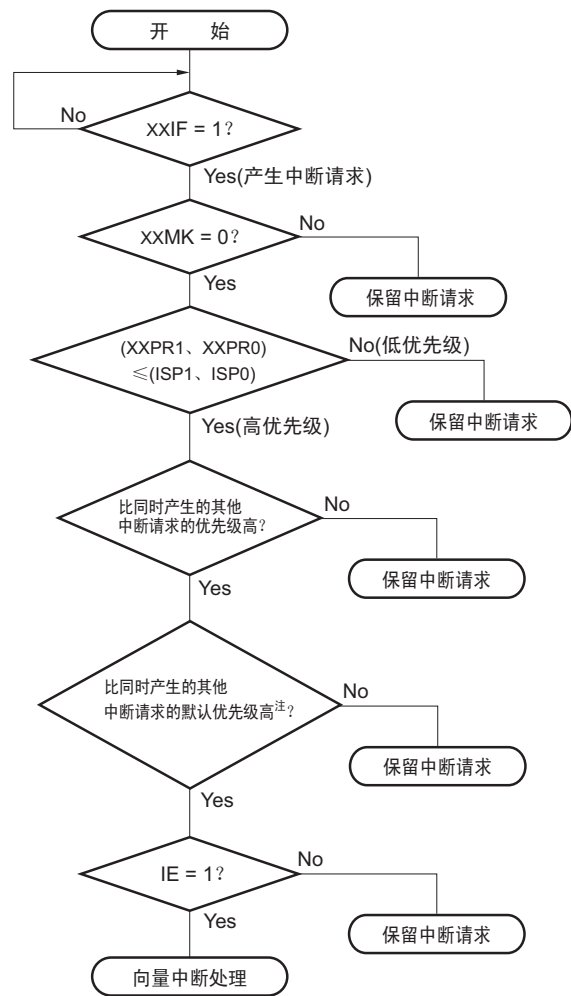
被保留的中断请求在可接受的状态时被接受。

接受中断请求的处理算法如图 20-7 所示。

如果接受可屏蔽中断请求，就按程序状态字（PSW）、程序计数器（PC）的顺序将内容压栈，将 IE 标志复位为“0”，并且将接受的中断优先级指定标志的内容传送到 ISP1 标志和 ISP0 标志。将各中断请求决定的向量表中的数据加载到 PC 并且进行转移。

能使用 RETI 指令从中断返回。

图 20-7 接受中断请求的处理算法



- XXIF: 中断请求标志
XXMK: 中断屏蔽标志
XXPR0: 优先级指定标志 0
XXPR1: 优先级指定标志 1
IE: 可屏蔽中断请求的接受控制标志 (1= 允许、0= 禁止)
ISP0、ISP1: 表示正在处理的中断优先级的标志 (参照图 20-6)

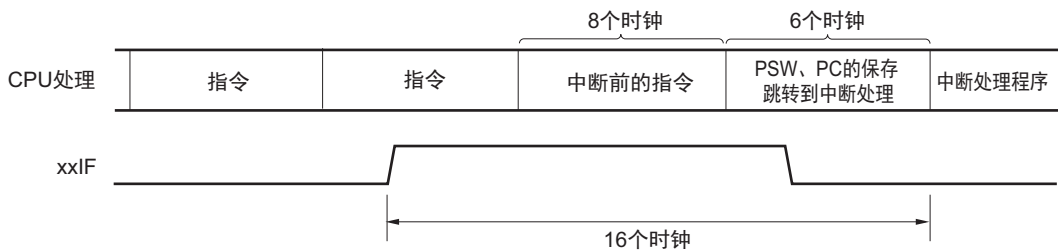
注 有关默认优先级，请参照“表 20-1 中断源一览表”。

图 20-8 中断请求的接受时序（最短时间）



备注 1 个时钟：1/ f_{CLK} (f_{CLK} ：CPU 时钟)

图 20-9 中断请求的接受时序（最长时间）



备注 1 个时钟：1/ f_{CLK} (f_{CLK} ：CPU 时钟)

20.4.2 接受软件中断请求

通过执行 BRK 指令接受软件中断请求，不能禁止软件中断。

如果接受软件中断请求，就按程序状态字（PSW）、程序计数器（PC）的顺序将内容压栈，将 IE 标志复位为“0”，并且将向量表（0007EH、0007FH）的内容加载到 PC 进行转移。

能使用 RETB 指令从软件中断返回。

注意 不能使用 RETI 指令从软件中断返回。

20.4.3 多重中断处理

在中断处理过程中接受了另一个中断请求的情况称为多重中断。

除非中断请求为接受允许状态（IE=1），否则不会发生多重中断。在接受中断请求时，中断请求为接受被禁止状态（IE=0）。因此，如果要允许多重中断，就需要在中断处理过程中用 EI 指令将 IE 标志置“1”，进入中断允许状态。

另外，即使在中断允许状态下也可能不允许多重中断，由中断优先级进行控制。中断优先级有默认优先级和可编程优先级，通过可编程优先级控制多重中断。

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优先级，就作为多重中断进行接受。如果产生的中断请求的优先级等于或者低于正在处理的中断优先级，就不作为多重中断进行接受。但是，如果在处理优先级 0 的中断过程中将 IE 标志置“1”，就允许优先级 0 的其他中断。

保留因中断被禁止或者优先级低而不允许多重中断的中断请求，在当前的中断处理结束后至少执行一条主处理的指令，然后接受被保留的中断请求。

能多重中断的中断请求的关系和多重中断的例子如表 20-5 和图 20-10 所示。

表 20-5 在中断处理过程中能多重中断的中断请求的关系

多重中断请求 正在处理的中断		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	
可屏蔽 中断	ISP1=0 ISP0=0	○	×	×	×	×	×	×	×	○
	ISP1=0 ISP0=1	○	×	○	×	×	×	×	×	○
	ISP1=1 ISP0=0	○	×	○	×	○	×	×	×	○
	ISP1=1 ISP0=1	○	×	○	×	○	×	○	×	○
软件中断		○	×	○	×	○	×	○	×	○

备注 1. ○：能多重中断

2. ×：不能多重中断

3. ISP0、ISP1 和 IE 是 PSW 中的标志。

ISP1=0、ISP0=0：正在处理优先级 1 或者优先级 0 的中断。

ISP1=0、ISP0=1：正在处理优先级 2 的中断。

ISP1=1、ISP0=0：正在处理优先级 3 的中断。

ISP1=1、ISP0=1：等待接受中断（允许全部中断）。

IE=0：禁止接受中断请求。

IE=1：允许接受中断请求。

4. PR 是 PR00L、PR00H、PR01L、PR01H、PR02L、PR02H、PR03L、PR10L、PR10H、PR11L、PR11H、PR12L、PR12H、PR13L 寄存器中的标志。

PR=00：通过 xxPR1x=0、xxPR0x=0 指定优先级 0（高优先级）。

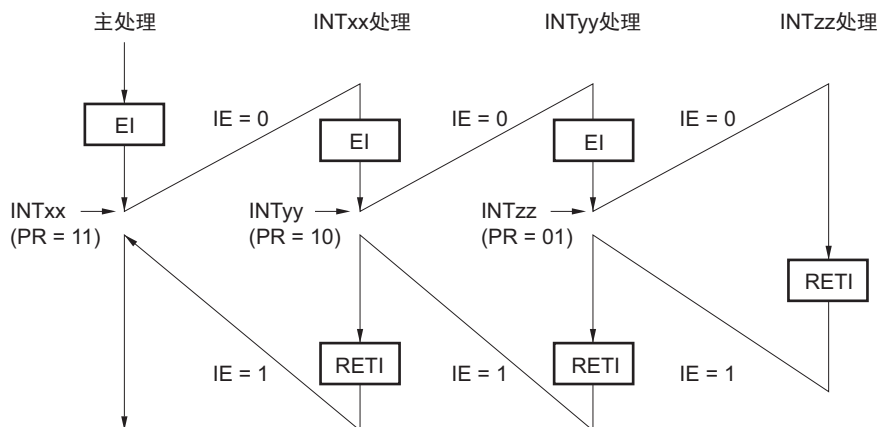
PR=01：通过 xxPR1x=0、xxPR0x=1 指定优先级 1。

PR=10：通过 xxPR1x=1、xxPR0x=0 指定优先级 2。

PR=11：通过 xxPR1x=1、xxPR0x=1 指定优先级 3（低优先级）。

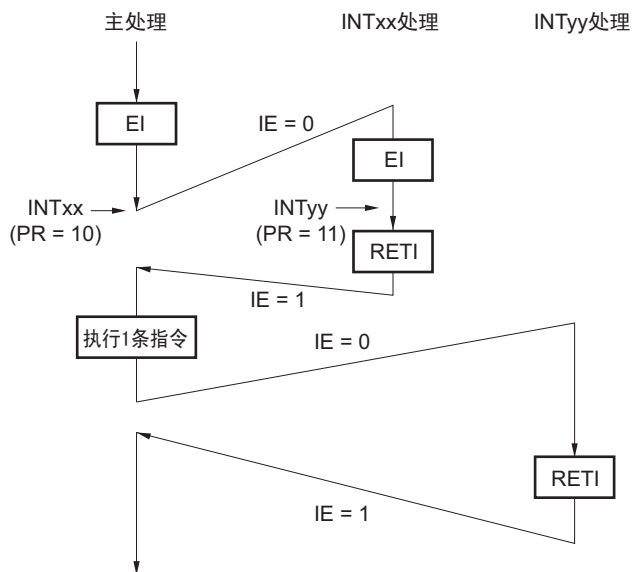
图 20-10 多重中断的例子 (1/2)

例 1. 发生 2 次多重中断的例子



在处理中断 INTxx 的过程中接受 INTyy 和 INTzz 的 2 个中断请求，发生多重中断。必须在接受各中断请求之前发行 EI 指令，使中断请求处于允许接受的状态。

例 2. 因控制优先级而不发生多重中断的例子



在处理中断 INTxx 的过程中，发生的 interrupt 请求 INTyy 因其 interrupt 优先级低于 INTxx 而不被接受，所以不发生多重中断。保留 INTyy 请求，在执行 1 条主处理的指令后接受被保留的 INTyy 请求。

PR=00: 通过 xxPR1x=0、xxPR0x=0 指定优先级 0（高优先级）。

PR=01: 通过 xxPR1x=0、xxPR0x=1 指定优先级 1。

PR=10: 通过 xxPR1x=1、xxPR0x=0 指定优先级 2。

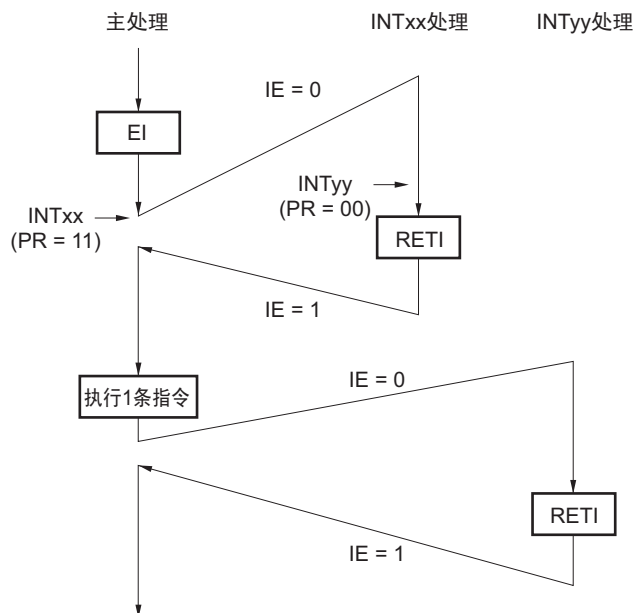
PR=11: 通过 xxPR1x=1、xxPR0x=1 指定优先级 3（低优先级）。

IE=0: 禁止接受中断请求。

IE=1: 允许接受中断请求。

图 20-10 多重中断的例子 (2/2)

例 3. 因不允许中断而不发生多重中断的例子



在处理中断 INTxx 的过程中，因不允许中断（未发行 EI 指令）而不接受中断请求 INTyy，所以不发生多重中断。保留 INTyy 请求，在执行 1 条主处理的指令后接受被保留的 INTyy 请求。

PR=00: 通过 xxPR1x=0、xxPR0x=0 指定优先级 0（高优先级）。

PR=01: 通过 xxPR1x=0、xxPR0x=1 指定优先级 1。

PR=10: 通过 xxPR1x=1、xxPR0x=0 指定优先级 2。

PR=11: 通过 xxPR1x=1、xxPR0x=1 指定优先级 3（低优先级）。

IE=0: 禁止接受中断请求。

IE=1: 允许接受中断请求。

20.4.4 中断请求的保留

有些指令，即使在执行指令的过程中发生了中断请求，也可能将中断请求保留到下一条指令执行结束为止。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
- IF0L、IF0H、IF1L、IF1H、IF2L、IF2H、IF3L、MK0L、MK0H、MK1L、MK1H、MK2L、MK2H、MK3L、PR00L、PR00H、PR01L、PR01H、PR02L、PR02H、PR03L、PR10L、PR10H、PR11L、PR11H、PR12L、PR12H和PR13L的各寄存器的写指令

中断请求的保留时序如图 20-11 所示。

图 20-11 中断请求的保留



- 备注 1. 指令 N：中断请求的保留指令
2. 指令 M：中断请求保留指令以外的指令

第 21 章 待机功能

21.1 待机功能

待机功能是进一步降低系统工作电流的功能，有以下 3 种模式。

(1) HALT 模式

通过执行 HALT 指令进入 HALT 模式。HALT 模式是停止 CPU 运行时钟的模式。在设定 HALT 模式前，如果高速系统时钟振荡电路、高速内部振荡器或者副系统时钟振荡电路正在振荡，各时钟就继续振荡。虽然此模式无法让工作电流降到 STOP 模式的程度，但是在想要通过中断请求立即重新开始处理或者想要频繁地进行间歇运行时是一种有效的模式。

(2) STOP 模式

通过执行 STOP 指令进入 STOP 模式。STOP 模式是停止高速系统时钟振荡电路和高速内部振荡器的振荡并且停止整个系统的模式。能大幅度地降低 CPU 的工作电流。

因为 STOP 模式能通过中断请求来解除，所以也能进行间歇运行。但是，在 X1 时钟的情况下，因为在解除 STOP 模式时需要确保振荡稳定的等待时间，所以如果一定要通过中断请求立即开始处理，就必须选择 HALT 模式。

(3) SNOOZE 模式

通过 CSIp 或者 UARTq 的数据接收以及由定时器触发信号（中断请求信号（INTRTC/INTIT））产生的 A/D 转换请求，解除 STOP 模式，不需要 CPU 运行而进行 CSIp 或者 UARTq 的数据接收，或者进行 A/D 转换。只有在选择高速内部振荡器作为 CPU/ 外围硬件时钟（ f_{CLK} ）时才能设定 SNOOZE 模式。

在任何一种模式中，寄存器、标志和数据存储器全部保持设定为待机模式前的内容，并且还保持输入 / 输出端口的输出锁存器和输出缓冲器的状态。

- 注意 1. 只有在 CPU 以主系统时钟运行时才能使用 STOP 模式。当 CPU 以副系统时钟运行时，不能设定为 STOP 模式。无论 CPU 是以主系统时钟还是以副系统时钟运行，都能使用 HALT 模式。
2. 在转移到 STOP 模式时，必须在停止以主系统时钟运行的外围硬件后执行 STOP 指令（SNOOZE 模式的设定单元除外）。
 3. 在 SNOOZE 模式中使用 CSIp、UARTq 或者 A/D 转换器时，必须在转移到 STOP 模式前设定串行待机控制寄存器 m（SSCm）和 A/D 转换器的模式寄存器 2（ADM2）。详细内容请参照“14.3 控制串行阵列单元的寄存器”和“12.3 控制 A/D 转换器的寄存器”。
 4. 为了降低 A/D 转换器的功耗，必须将 A/D 转换器的模式寄存器 0（ADM0）的 bit7（ADCS）和 bit0（ADCE）清“0”，在停止 A/D 转换运行后执行 STOP 指令。
 5. 能通过选项字节选择在 HALT 模式或者 STOP 模式中是继续还是停止低速内部振荡器的振荡。详细内容请参照“第 27 章 选项字节”。

备注 p=00, q=0、2, m=0、1

21.2 待机功能的控制寄存器

待机功能的控制寄存器如下所示。

- 副系统时钟提供模式控制寄存器 (OSMC)
- 振荡稳定时间计数器的状态寄存器 (OSTC)
- 振荡稳定时间选择寄存器 (OSTS)

备注 上述寄存器的详细内容请参照“第 5 章 时钟发生电路”。另外，有关 SNOOZE 模式功能的控制寄存器，请参照“第 12 章 A/D 转换器”、“第 14 章 串行阵列单元”。

21.3 待机功能的运行

21.3.1 HALT 模式

(1) HALT 模式

通过执行 HALT 指令设定为 HALT 模式。无论设定前的 CPU 时钟是高速系统时钟还是高速内部振荡器时钟或者副系统时钟，都能进行设定。

HALT 模式中的运行状态如下所示。

注意 在中断屏蔽标志为“0”（允许中断处理）并且中断请求标志为“1”（产生中断请求信号）时，中断请求信号用于解除 HALT 模式，因此，在此状态下，即使执行 HALT 指令，也不转移到 HALT 模式。

表 21-1 HALT 模式中的运行状态 (1/2)

HALT 模式的设定 项目			在 CPU 以主系统时钟运行的过程中执行 HALT 指令的情况		
			CPU 以高速内部振荡器时钟（f _{IH} ）运行	CPU 以 X1 时钟 （f _X ）运行	CPU 以外部主系统时钟（f _{EX} ）运行
系统时钟			停止给 CPU 提供时钟。		
	主系统时钟	f _{IH}	继续运行 （不能停止）。	禁止运行	
		f _X	禁止运行	继续运行 （不能停止）。	不能运行
		f _{EX}		不能运行	继续运行 （不能停止）。
	副系统时钟	f _{XT}	保持设定为 HALT 模式前的状态。		
		f _{EXS}			
f _{IL}		通过选项字节（000C0H）的 bit0（WDSTBYON）和 bit4（WDTON）以及副系统时钟提供模式控制寄存器（OSMC）的 WUTMMCK0 位进行设定。 • WUTMMCK0=1：振荡 • WUTMMCK0=0 并且 WDTON=0：停止 • WUTMMCK0=0、WDTON=1 并且 WDSTBYON=1：振荡 • WUTMMCK0=0、WDTON=1 并且 WDSTBYON=0：停止			
CPU			停止运行		
闪存			停止运行		
RAM					
端口（锁存器）			保持设定为 HALT 模式前的状态。		
定时器阵列单元			能运行		
实时时钟 2			能运行	能运行（高精度 1Hz 输出模式中禁止运行）	
副系统时钟频率测量电路			禁止运行	能运行	
12 位间隔定时器			能运行		
看门狗定时器			参照“第 11 章 看门狗定时器”		
时钟输出 / 蜂鸣器输出			能运行		
A/D 转换器					
比较器					
串行阵列单元（SAU）					
串行接口（IICA）					
智能卡接口（SMCI）					
LCD 控制器 / 驱动器					
乘除器和乘加器					
DMA 控制器			能运行		
上电复位功能					
电压检测功能					
外部中断					
CRC 运算功能	高速 CRC				
	通用 CRC				
RAM 奇偶校验错误检测功能			在执行 DMA 时，能运行。		
RAM 保护功能			非法存储器存取检测功能		
SFR 保护功能					

备注 停止运行: 在转移到 HALT 模式时自动停止运行。

禁止运行: 在转移到 HALT 模式前停止运行。

 f_{IH} : 高速内部振荡器时钟 f_{IL} : 低速内部振荡器时钟 f_X : X1 时钟 f_{EX} : 外部主系统时钟 f_{XT} : XT1 时钟 f_{EXS} : 外部副系统时钟

表 21-1 HALT 模式中的运行状态 (2/2)

HALT 模式的设定			在 CPU 以副系统时钟运行的过程中执行 HALT 指令的情况	
项目			CPU 以 XT1 时钟 (f_{XT}) 运行	CPU 以外副系统时钟 (f_{EXS}) 运行
系统时钟			停止给 CPU 提供时钟。	
主系统时钟	f_{IH}	f_X	禁止运行	
		f_{EX}		
		f_{XT}		
	副系统时钟	f_{EXS}	继续运行 (不能停止)。	不能运行。
f_{IL}	f_{XT}	f_{EXS}	不能运行	继续运行 (不能停止)。
		f_{XT}	继续运行 (不能停止)。	不能运行。
		f_{EXS}	不能运行	继续运行 (不能停止)。
	f_{IL}		通过选项字节 (000C0H) 的 bit0 (WDSTBYON) 和 bit4 (WDTON) 以及副系统时钟提供模式控制寄存器 (OSMC) 的 WUTMMCK0 位进行设定。 • WUTMMCK0=1: 振荡 • WUTMMCK0=0 并且 WDTON=0: 停止 • WUTMMCK0=0、WDTON=1 并且 WDSTBYON=1: 振荡 • WUTMMCK0=0、WDTON=1 并且 WDSTBYON=0: 停止	
CPU			停止运行	
闪存				
RAM				
端口 (锁存器)			保持设定为 HALT 模式前的状态。	
定时器阵列单元			当 RTCLPC=0 时, 能运行 (否则禁止运行)。	
实时时钟 2			能运行 (高精度 1Hz 输出模式中禁止运行)	
副系统时钟频率测量电路			禁止运行	
12 位间隔定时器			能运行	
看门狗定时器			参照“第 11 章 看门狗定时器”。	
时钟输出 / 蜂鸣器输出			能运行	
A/D 转换器			禁止运行	
比较器			选择外部输入 (IVREFn) 作为比较器的基准电压时能运行。	
串行阵列单元 (SAU)			当 RTCLPC=0 时, 能运行 (否则禁止运行)。	
串行接口 (IICA)			禁止运行	
智能卡接口 (SMCI)				
LCD 控制器 / 驱动器			能运行 (但是取决于作为 LCD 源时钟选择的时钟的状态 (如果选择的时钟运行, 就能运行, 否则停止运行。))	
乘除器和乘加器			当 RTCLPC=0 时, 能运行 (否则禁止运行)。	
DMA 控制器				
上电复位功能			能运行	
电压检测功能				
外部中断				
CRC 运算	高速 CRC		禁止运行	
功能	通用 CRC		在 RAM 区的运算中执行 DMA 时, 能运行。	
RAM 奇偶校验错误检测功能			在执行 DMA 时, 能运行。	
RAM 保护功能				
SFR 保护功能				
非法存储器存取检测功能				

备注 停止运行: 在转移到 HALT 模式时自动停止运行。

禁止运行: 在转移到 HALT 模式前停止运行。

f_{IH} : 高速内部振荡器时钟

f_{IL} : 低速内部振荡器时钟

f_X : X1 时钟

f_{EX} : 外部主系统时钟

f_{XT} : XT1 时钟

f_{EXS} : 外部副系统时钟

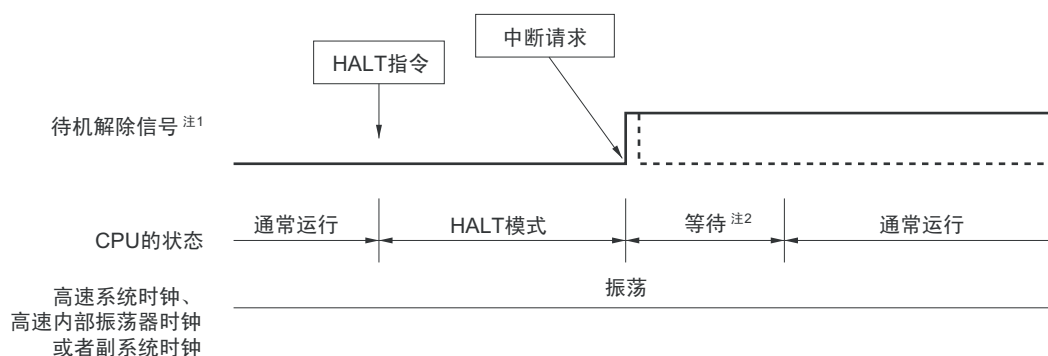
(2) HALT 模式的解除

能通过以下 2 种方法解除 HALT 模式。

(a) 通过未屏蔽的中断请求进行的解除

如果发生未屏蔽的中断请求，就解除 HALT 模式。然后，如果处于允许接受中断的状态，就进行向量中断的处理。如果处于禁止接受中断的状态，就执行下一个地址的指令。

图 21-1 通过中断请求解除 HALT 模式



注 1. 有关待机解除信号的详细内容，请参照图 20-1。

2. HALT 模式解除的等待时间

• 进行向量中断处理的情况

主系统时钟：15 ~ 16 个时钟

副系统时钟 (RTCLPC=0)：10 ~ 11 个时钟

副系统时钟 (RTCLPC=1)：11 ~ 12 个时钟

• 不进行向量中断处理的情况

主系统时钟：9 ~ 10 个时钟

副系统时钟 (RTCLPC=0)：4 ~ 5 个时钟

副系统时钟 (RTCLPC=1)：5 ~ 6 个时钟

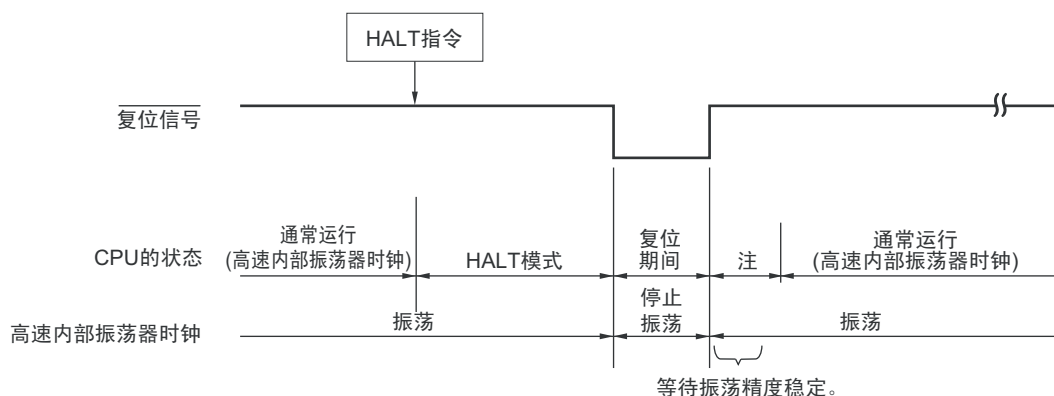
备注 虚线表示接受解除了待机模式的中断请求的情况。

(b) 通过产生复位信号进行的解除

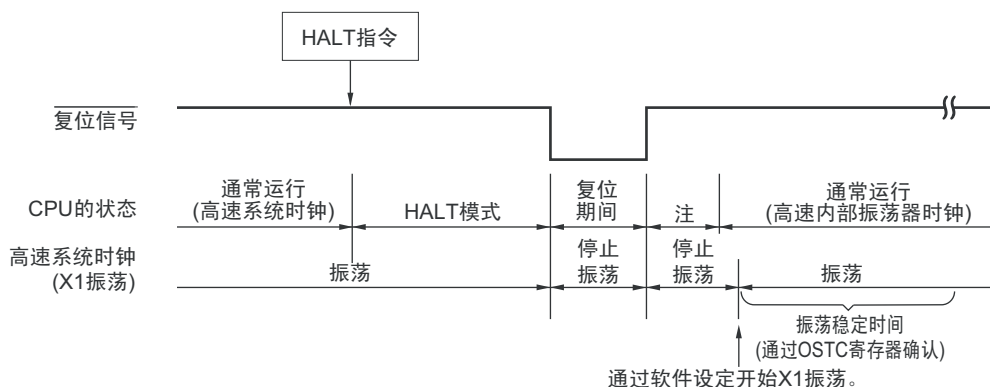
通过产生复位信号来解除 HALT 模式。然后，和通常的复位一样，在转移到复位向量地址后执行程序。

图 21-2 通过复位解除 HALT 模式 (1/2)

(1) CPU时钟为高速内部振荡器时钟的情况



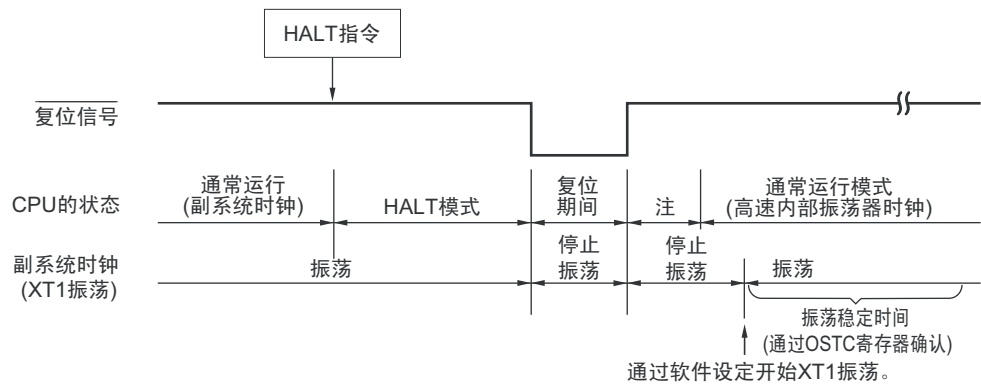
(2) CPU时钟为高速系统时钟的情况



注 有关复位处理时间，请参照“第 22 章 复位功能”。有关上电复位（POR）电路和电压检测（LVD）电路的复位处理时间，请参照“第 23 章 上电复位电路”。

图 21-2 通过复位解除 HALT 模式（2/2）

(3) CPU时钟为副系统时钟的情况



注 有关复位处理时间，请参照“第 22 章 复位功能”。有关上电复位（POR）电路和电压检测（LVD）电路的复位处理时间，请参照“第 23 章 上电复位电路”。

21.3.2 STOP 模式

(1) STOP 模式的设定和运行状态

通过执行 STOP 指令设定为 STOP 模式。只有在设定前的 CPU 时钟为主系统时钟的情况下才能进行设定。

注意 因为在中断屏蔽标志为“0”（允许中断处理）并且中断请求标志为“1”（产生中断请求信号）时，中断请求信号用于解除 STOP 模式，所以在此状态下执行 STOP 指令时，一旦转移到 STOP 模式，然后立即被解除。因此，在执行 STOP 指令后，经过 STOP 模式解除时间，就返回运行模式。

STOP 模式中的运行状态如下所示。

表 21-2 STOP 模式中的运行状态

STOP 模式的设定 项目		在 CPU 以主系统时钟运行的过程中执行 STOP 指令的情况		
		CPU 以高速内部振荡器时钟 (f_{IH}) 运行	CPU 以 X1 时钟 (f_X) 运行	CPU 以外部主系统时钟 (f_{EX}) 运行
系统时钟		停止给 CPU 提供时钟。		
主系统时钟	f_{IH}	停止		
				f_X
				f_{EX}
	副系统时钟	f_{XT}	保持设定为 STOP 模式前的状态。	
		f_{EXS}		
f_{IL}		通过选项字节 (000C0H) 的 bit0 (WDSTBYON) 和 bit4 (WDTON) 以及副系统时钟提供模式控制寄存器 (OSMC) 的 WUTMMCK0 位进行设定。 • WUTMMCK0=1: 振荡 • WUTMMCK0=0 并且 WDTON=0: 停止 • WUTMMCK0=0、WDTON=1 并且 WDSTBYON=1: 振荡 • WUTMMCK0=0、WDTON=1 并且 WDSTBYON=0: 停止		
CPU		停止运行。		
闪存				
RAM				
端口 (锁存器)		保持设定为 STOP 模式前的状态。		
定时器阵列单元		禁止运行。		
实时时钟 2		能运行 (高精度 1Hz 输出模式中禁止运行)。		
副系统时钟频率测量电路		禁止运行		
12 位间隔定时器		能运行		
看门狗定时器		参照“第 11 章 看门狗定时器”。		
时钟输出 / 蜂鸣器输出		在选择副系统时钟作为计数时钟时, 能运行。		
A/D 转换器		能唤醒运行 (转移到 SNOOZE 模式)。		
比较器		能运行 (只在未使用数字滤波器并且选择外部输入 (IVREFn) 作为比较器的基准电压时)		
串行阵列单元 (SAU)		只有 CSIp 和 UARTq 才能唤醒运行 (转移到 SNOOZE 模式)。 除了 CSIp 和 UARTq 以外, 禁止运行。		
串行接口 (IICA)		能通过地址一致进行唤醒。		
智能卡接口 (SMCI)		禁止运行		
LCD 控制器 / 驱动器		能运行 (但是取决于作为 LCD 源时钟选择的时钟的状态 (如果选择的时钟运行, 就能运行, 否则停止运行。))		
乘除器和乘加器		禁止运行。		
DMA 控制器		能运行。		
上电复位功能				
电压检测功能				
外部中断				
键中断功能				
CRC 运算 功能	高速 CRC	停止运行。		
	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				

备注 1. 停止运行: 在转移到 STOP 模式时自动停止运行。

禁止运行: 在转移到 STOP 模式前停止运行。

 f_{IH} : 高速内部振荡器时钟 f_X : X1 时钟 f_{XT} : XT1 时钟 f_{IL} : 低速内部振荡器时钟 f_{EX} : 外部主系统时钟 f_{EXS} : 外部副系统时钟

2. p=00, q=0、2

(2) STOP 模式的解除

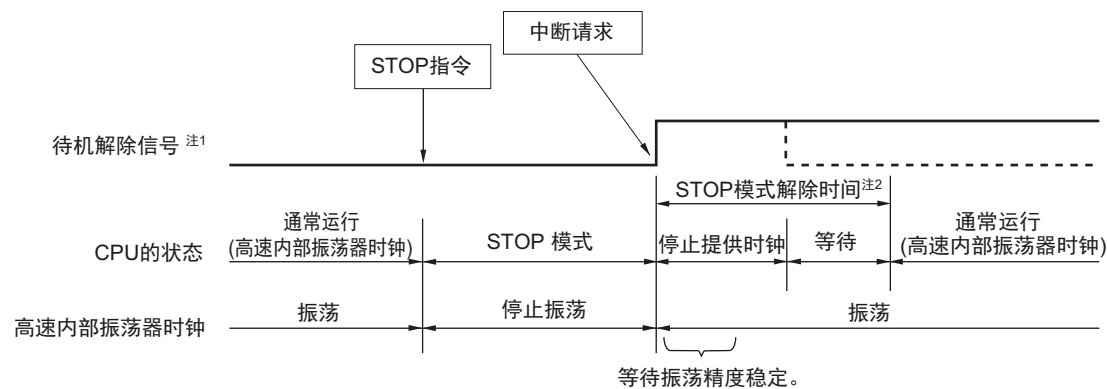
能通过以下 2 种方法解除 STOP 模式。

(a) 通过未屏蔽的中断请求进行的解除

如果发生未屏蔽的中断请求，就解除 STOP 模式。在经过振荡稳定时间后，如果处于允许接受中断的状态，就进行向量中断的处理。如果处于禁止接受中断的状态，就执行下一个地址的指令。

图 21-3 通过中断请求解除 STOP 模式 (1/2)

(1) CPU时钟为高速内部振荡器时钟的情况

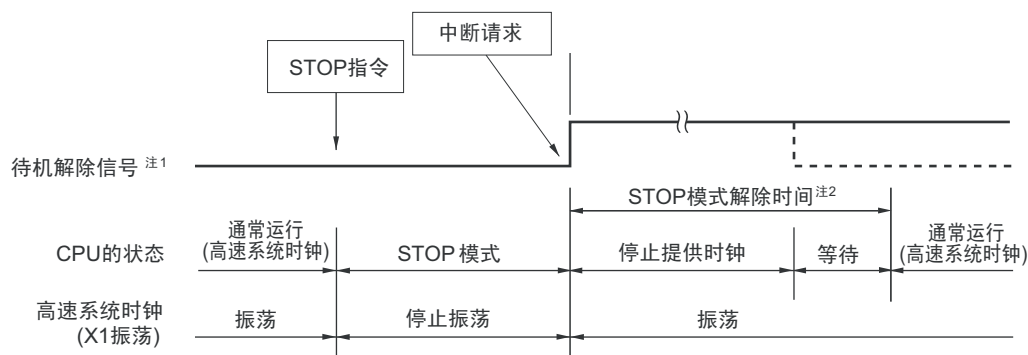


- 注 1. 有关待机解除信号的详细内容，请参照图 20-1。
2. STOP 模式的解除时间
- 停止提供时钟：18μs ~ “65μs 和振荡稳定时间（通过 OSTS 进行设定）中较长的时间”
- 等待
- 进行向量中断处理的情况： 7 个时钟
 - 不进行向量中断处理的情况： 1 个时钟

- 备注 1. 时钟提供的停止时间因温度条件和 STOP 模式期间而变。
2. 虚线表示接受解除了待机模式的中断请求的情况。

图 21-3 通过中断请求解除 STOP 模式 (2/2)

(2) CPU时钟为高速系统时钟 (X1 振荡) 的情况



注 1. 有关待机解除信号的详细内容, 请参照图 20-1。

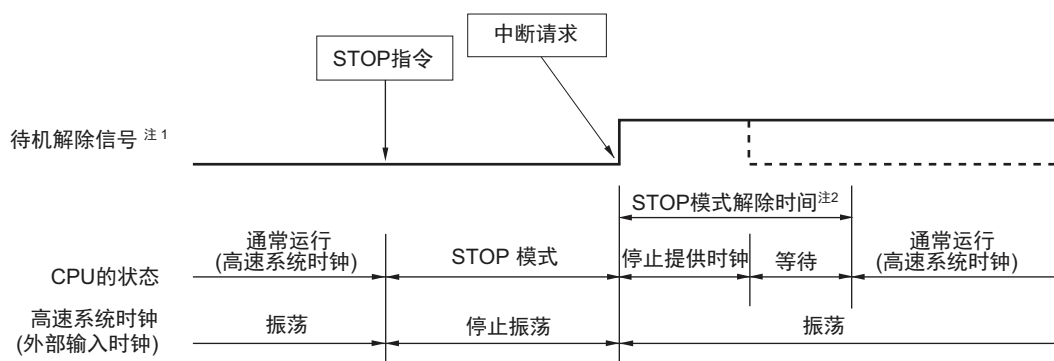
2. STOP 模式的解除时间

停止提供时钟: $18\mu\text{s} \sim$ “ $65\mu\text{s}$ 和 振荡稳定时间 (通过 OSTS 进行设定) 中较长的时间”

等待

- 进行向量中断处理的情况: 10 ~ 11 个时钟
- 不进行向量中断处理的情况: 4 ~ 5 个时钟

(3) CPU时钟为高速系统时钟 (外部时钟输入) 的情况



注 1. 有关待机解除信号的详细内容, 请参照图 20-1。

2. STOP 模式的解除时间

停止提供时钟 : $18\mu\text{s} \sim 65\mu\text{s}$

等待

- 进行向量中断处理的情况: 7 个时钟
- 不进行向量中断处理的情况: 1 个时钟

注意 在 CPU 以高速系统时钟 (X1 振荡) 运行, 并且想要缩短解除 STOP 模式后的振荡稳定时间时, 必须在执行 STOP 指令前, 将 CPU 时钟暂时切换到高速内部振荡器时钟。

备注 1. 时钟提供的停止时间因温度条件和 STOP 模式期间而变。

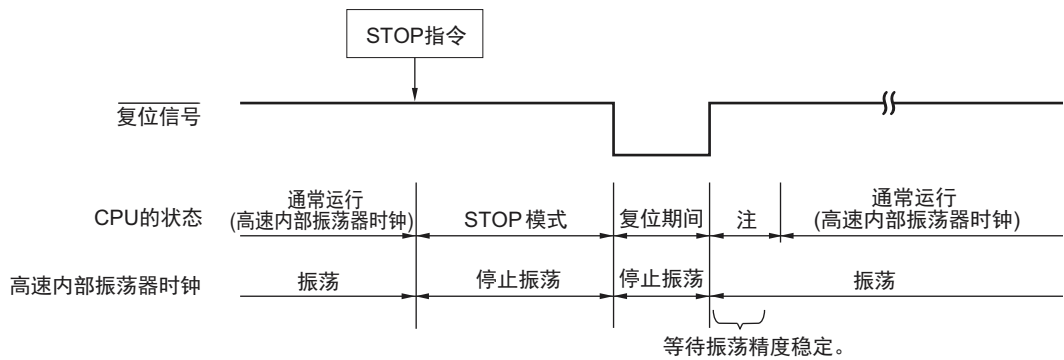
2. 虚线表示接受解除了待机模式的中断请求的情况。

(b) 通过产生复位信号进行的解除

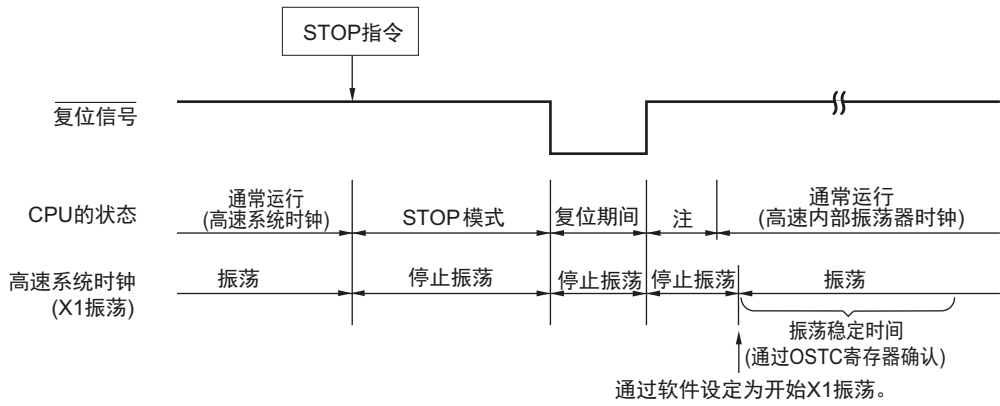
通过产生复位信号来解除 STOP 模式。然后，和通常的复位一样，在转移到复位向量地址后执行程序。

图 21-4 通过复位解除 STOP 模式

(1) CPU时钟为高速内部振荡器时钟的情况



(2) CPU时钟为高速系统时钟的情况



注 有关复位处理时间，请参照“第 22 章 复位功能”。有关上电复位（POR）电路和电压检测（LVD）电路的复位处理时间，请参照“第 23 章 上电复位电路”。

21.3.3 SNOOZE 模式

(1) SNOOZE 模式的设定和运行状态

只能给 CSIp、UARTq 或者 A/D 转换器设定 SNOOZE 模式，并且只有在设定前的 CPU 时钟为高速内部振荡器时钟的情况下才能进行设定。

要在 SNOOZE 模式中使用 CSIp 或者 UARTq 时，必须在转移到 STOP 模式前将串行待机控制寄存器 m (SSCm) 的 SWCm 位置“1”。详细内容请参照“14.3 控制串行阵列单元的寄存器”。

要在 SNOOZE 模式中使用 A/D 转换器时，必须在转移到 STOP 模式前将 A/D 转换器的模式寄存器 2 (ADM2) 的 AWC 位置“1”。详细内容请参照“12.3 控制 A/D 转换器的寄存器”。

备注 p=00, q=0、2, m=0、1

在进行 SNOOZE 模式转移时，只在以下时间内变为等待状态。

STOP 模式 → SNOOZE 模式的转移时间：18μs ~ 65μs

SNOOZE 模式 → 通常运行的转移时间：

- 进行向量中断处理的情况：
 - HS（高速主）模式：“4.99 ~ 9.44μs”+7 个时钟
 - LS（低速主）模式：“1.10 ~ 5.08μs”+7 个时钟
 - LV（低电压主）模式：“16.58 ~ 25.40μs”+7 个时钟
- 不进行向量中断处理的情况：
 - HS（高速主）模式：“4.99 ~ 9.44μs”+1 个时钟
 - LS（低速主）模式：“1.10 ~ 5.08μs”+1 个时钟
 - LV（低电压主）模式：“16.58 ~ 25.40μs”+1 个时钟

SNOOZE 模式中的运行状态如下所示。

表 21-3 SNOOZE 模式中的运行状态

STOP 模式的设定			在 STOP 模式中输入 CSIp、UARTq 的数据接收信号或者 A/D 转换器的定时器触发信号的情况		
项目			CPU 以高速内部振荡器时钟（f _{IH} ）运行		
系统时钟			停止给 CPU 提供时钟。		
			主系统 时钟	f _{IH}	开始运行。
				f _X	停止
				f _{EX}	
			副系统 时钟	f _{XT}	保持 STOP 模式中的状态。
				f _{EXS}	
f _{IL}		通过选项字节（000C0H）的 bit0（WDSTBYON）和 bit4（WDTON）以及副系统时钟提供模式控制寄存器（OSMC）的 WUTMMCK0 位进行设定。 • WUTMMCK0=1：振荡 • WUTMMCK0=0 并且 WDTON=0：停止 • WUTMMCK0=0、WDTON=1 并且 WDSTBYON=1：振荡 • WUTMMCK0=0、WDTON=1 并且 WDSTBYON=0：停止			
CPU			停止运行		
闪存					
RAM					
端口（锁存器）			保持 STOP 模式中的状态。		
定时器阵列单元			禁止运行		
实时时钟 2			能运行		
副系统时钟频率测量电路			禁止运行		
12 位间隔定时器			能运行		
看门狗定时器			参照“第 11 章 看门狗定时器”。		
时钟输出 / 蜂鸣器输出			在选择副系统时钟作为计数时钟并且 RTCLPC=0 时，能运行（否则禁止运行）。		
A/D 转换器			能运行		
比较器			能运行（只在未使用数字滤波器时）		
串行阵列单元（SAU）			只有 CSIp 和 UARTq 才能运行。 除了 CSIp 和 UARTq 以外，禁止运行。		
串行接口（IICA）			禁止运行		
智能卡接口（SMCI）					
LCD 控制器 / 驱动器			能运行（但是取决于作为 LCD 源时钟选择的时钟的状态（如果选择的时钟运行，就能运行，否则停止运行。））		
乘除器和乘加器			禁止运行		
DMA 控制器					
上电复位功能			能运行		
电压检测功能					
外部中断					
CRC 运算功能					
高速 CRC		停止运行			
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					

备注 1. 停止运行: 在转移到 STOP 模式时自动停止运行。

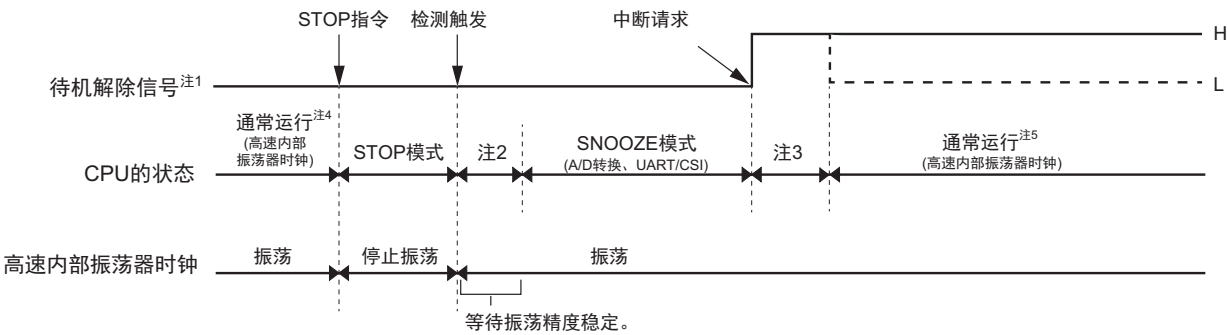
禁止运行: 在转移到 STOP 模式前停止运行。

 f_{IH} : 高速内部振荡器时钟 f_{IL} : 低速内部振荡器时钟 f_X : X1 时钟 f_{EX} : 外部主系统时钟 f_{XT} : XT1 时钟 f_{EXS} : 外部副系统时钟

2. p=00, q=0、2

(2) 在 SNOOZE 模式中产生中断请求信号时的时序图

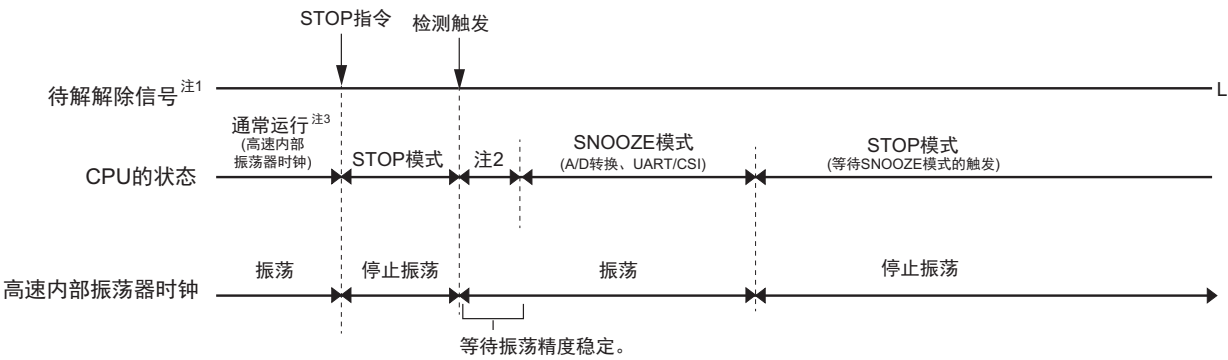
图 21-5 在 SNOOZE 模式中产生中断请求的情况



- 注 1. 有关待机解除信号的详细内容，请参照图 20-1。
2. STOP 模式→ SNOOZE 模式的转移时间
3. SNOOZE 模式→通常运行的转移时间
4. 在即将要转移到 STOP 模式前，必须先设定为允许 SNOOZE 模式（AWC=1/SWC=1）。
5. 在返回到通常运行后，必须立即设定为解除 SNOOZE 模式（AWC=0/SWC=0）。

(3) 没有在 SNOOZE 模式中产生中断请求信号时的时序图

图 21-6 在 SNOOZE 模式中没有产生中断请求的情况



- 注 1. 有关待机解除信号的详细内容，请参照图 20-1。
2. STOP 模式→ SNOOZE 模式的转移时间
3. 在即将要转移到 STOP 模式前，必须先设定为允许 SNOOZE 模式（AWC=1/SWC=1）。

备注 SNOOZE 模式功能的详细内容请参照“第 12 章 A/D 转换器”和“第 14 章 串行阵列单元”。

第 22 章 复位功能

以下 7 种方法产生复位信号。

- (1) 通过 $\overline{\text{RESET}}$ 引脚输入外部复位。
- (2) 通过看门狗定时器的程序失控检测产生内部复位。
- (3) 通过上电复位 (POR) 电路的电源电压和检测电压的比较产生内部复位。
- (4) 通过电压检测电路 (LVD) 的电源电压和检测电压的比较产生内部复位。
- (5) 因执行非法指令而产生内部复位注。
- (6) 因 RAM 奇偶校验错误而产生内部复位。
- (7) 因存取非法存储器而产生内部复位。

内部复位和外部复位相同，在产生复位信号后，从写在地址 0000H 和 0001H 中的地址开始执行程序。

当给 $\overline{\text{RESET}}$ 引脚输入低电平，或者看门狗定时器检测到程序失控，或者检测到 POR 电路和 LVD 电路的电压，或者执行非法指令注，或者发生 RAM 奇偶校验错误，或者存取非法存储器时，产生复位并且各硬件变为如表 22-1 所示的状态。

注 当执行指令码 FFH 时，发生非法指令错误。

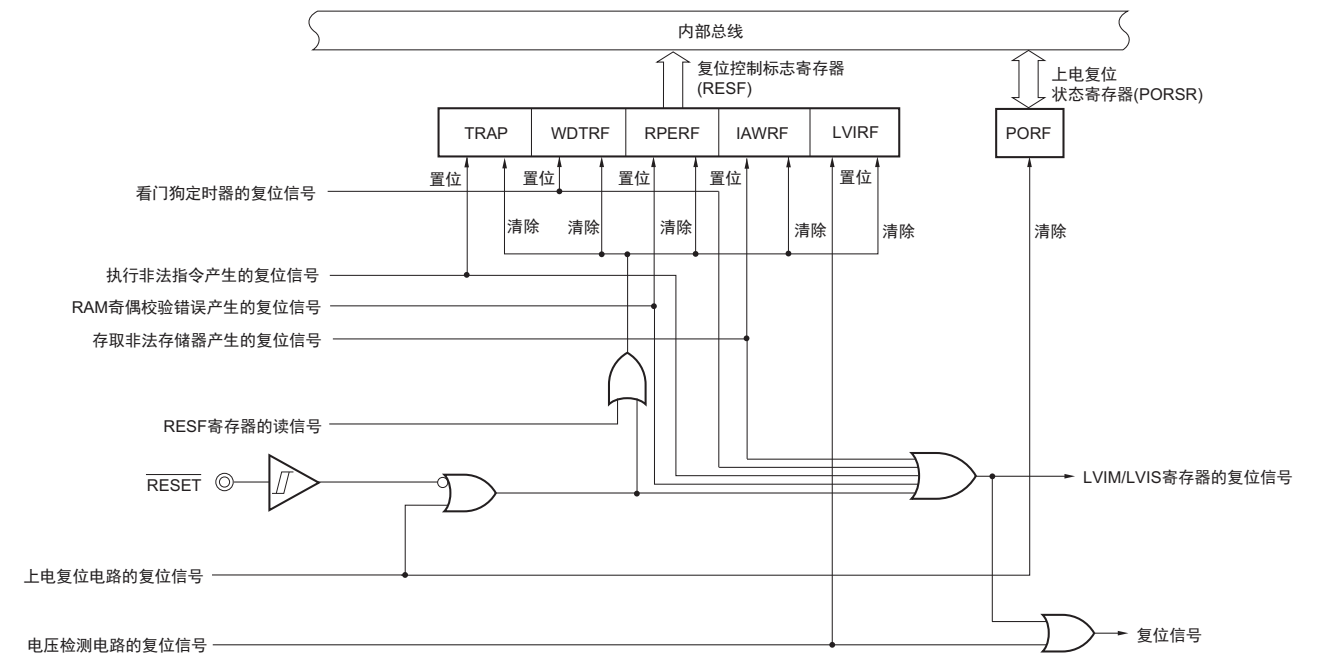
在通过在线仿真器或者片上调试仿真器进行仿真时，不会因执行非法指令而产生复位。

注意 1. 要进行外部复位时，必须至少给 $\overline{\text{RESET}}$ 引脚输入 10 μ s 的低电平。

如果要在接通电源时进行外部复位，就必须在给 $\overline{\text{RESET}}$ 引脚输入低电平后接通电源，并且在“32.4 AC 特性”所示的工作电压范围内的期间至少维持 10 μ s 的低电平，然后输入高电平。

2. 在复位信号有效期间，停止 X1 时钟、高速内部振荡器时钟和低速内部振荡器时钟的振荡。外部主系统时钟和外部副系统时钟的输入无效。
3. 如果发生复位，就对各 SFR 和 2nd SFR 进行初始化，端口引脚的状态如下变化。
 - P40: 在外部复位或者 POR 复位期间为高阻抗。在其他复位期间和接受复位后为高电平（连接内部上拉电阻）。
 - P40 以外的端口：在复位期间和接受复位后为高阻抗。

图 22-1 复位功能的框图



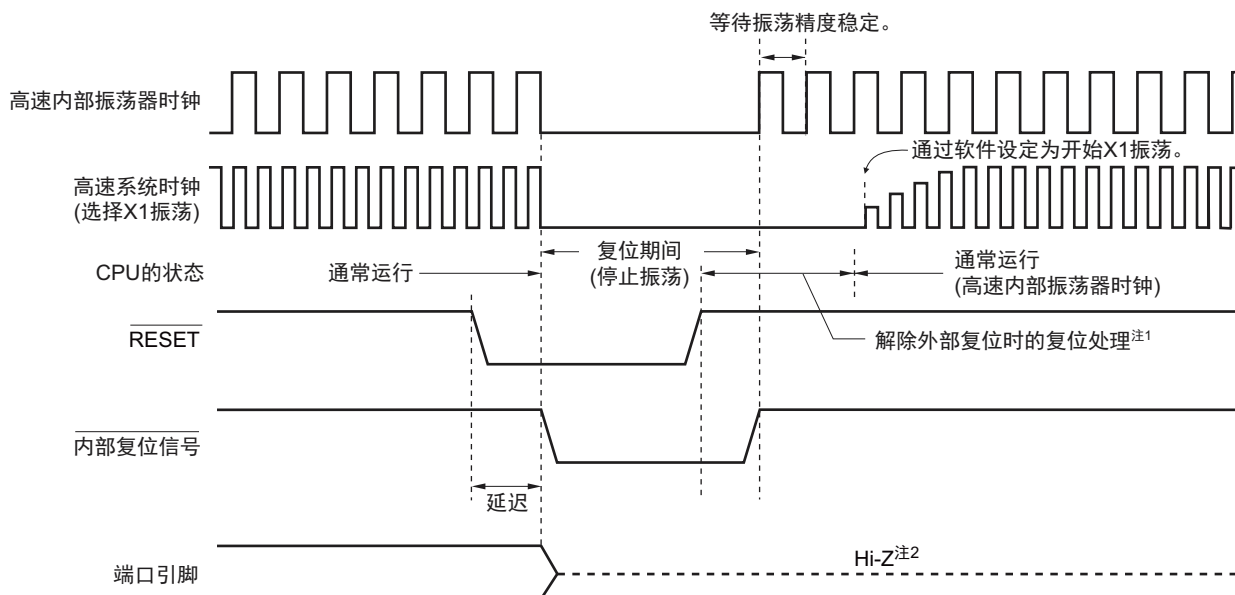
注意 LVD 电路的内部复位不会对 LVD 电路进行复位。

- 备注 1. LVIM: 电压检测寄存器
2. LVIS: 电压检测电平寄存器

22.1 复位运行的时序

如果给 $\overline{\text{RESET}}$ 引脚输入低电平，就发生复位，如果给 $\overline{\text{RESET}}$ 引脚输入高电平，就解除复位。复位处理后，通过高速内部振荡器时钟开始执行程序。

图 22-2 $\overline{\text{RESET}}$ 输入的复位时序



注 1. 外部复位解除时的复位时间:

解除 POR 后第 1 次:	0.672ms(TYP.)、0.832ms(MAX.) (使用 LVD 时)
	0.399ms(TYP.)、0.519ms(MAX.) (LVD 为 OFF 时)
解除 POR 后第 2 次以后:	0.531ms(TYP.)、0.675ms(MAX.) (使用 LVD 时)
	0.259ms(TYP.)、0.362ms(MAX.) (LVD 为 OFF 时)

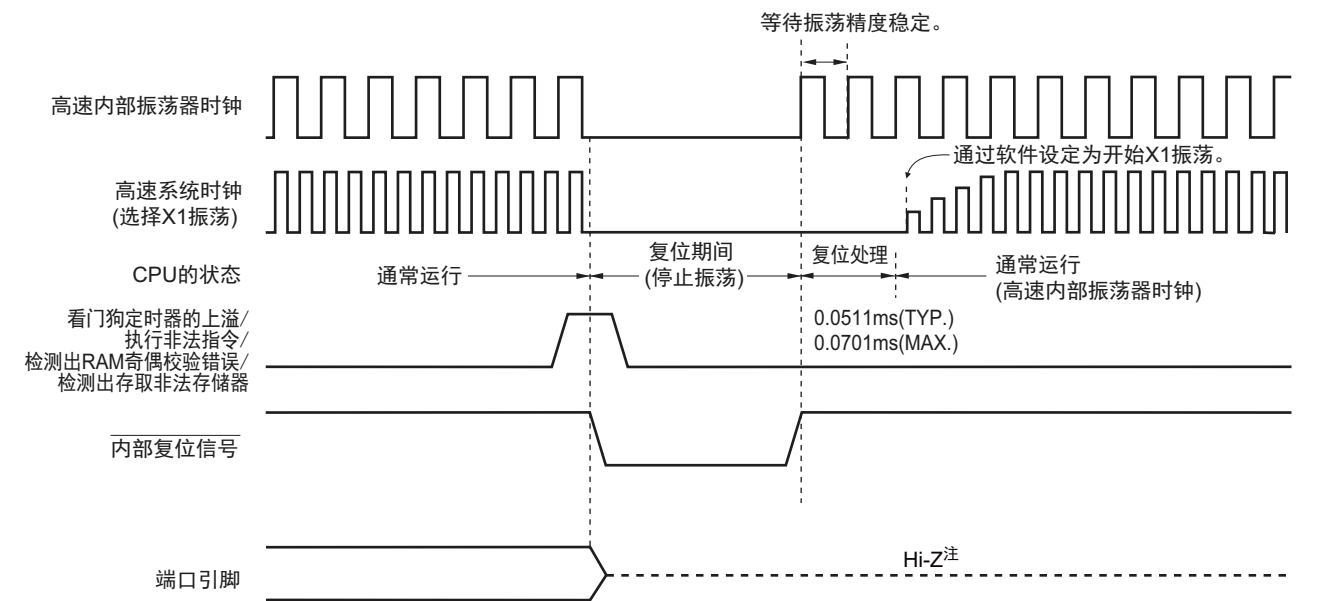
当接通电源时，在外部复位解除时的复位处理时间之前需要电压稳定等待时间 (0.99ms(TYP.)、2.30ms(MAX.))。

2. 端口引脚 P40 的状态如下所示。

- 在外部复位或者 POR 复位期间变为高阻抗。
- 在其他复位期间和接受复位后变为高电平 (连接内部上拉电阻)。

由看门狗定时器的上溢 / 执行非法指令 / RAM 奇偶校验错误 / 存取非法存储器而引起的复位自动被解除，并且在复位处理后，通过高速内部振荡器时钟开始执行程序。

图 22-3 由看门狗定时器的上溢 / 执行非法指令 / RAM 奇偶校验错误 / 存取非法存储器而引起的复位的时序



注 端口引脚 P40 的状态如下所示。

- 在外部复位或者 POR 复位期间变为高阻抗。
- 在其他复位期间和接受复位后变为高电平（连接内部上拉电阻）。

由 POR 电路、LVD 电路的电压检测而引起的复位，在复位后的 $V_{DD} \geq V_{POR}$ 或者 $V_{DD} \geq V_{LVD}$ 时，自动解除复位，并且在复位处理后，通过高速内部振荡器时钟开始执行程序。有关详细内容，请参照“第 23 章 上电复位电路”。

备注 V_{POR} : POR 电源上升沿的检测电压
 V_{LVD} : LVD 检测电压

22.2 复位期间的运行状态

复位期间的运行状态如表 22-1 所示，接受复位后的各硬件的状态如表 22-2 所示。

表 22-1 复位期间的运行状态

项目			复位期间
系统时钟			停止给 CPU 提供时钟。
主系统时钟	f_{IH}		停止运行
		f_X	停止运行（X1 引脚和 X2 引脚处于输入端口模式）。
		f_{EX}	时钟输入无效（引脚处于输入端口模式）。
	副系统时钟	f_{XT}	停止运行（XT1 引脚和 XT2 引脚处于输入端口模式）。
		f_{EXS}	时钟输入无效（引脚处于输入端口模式）。
f_{IL}		停止运行	
CPU			停止运行
闪存			停止运行
RAM			停止运行
端口（锁存器）			高阻抗注
定时器阵列单元			停止运行
实时时钟 2			POR 复位以外：能运行 POR 复位：日历能运行，RTCC0、RTCC1 和 SUBCUD 寄存器停止运行。
副系统时钟频率测量电路			停止运行
12 位间隔定时器			停止运行
看门狗定时器			
时钟输出 / 蜂鸣器输出			
A/D 转换器			
比较器			
串行阵列单元（SAU）			
串行接口（IICA）			
智能卡接口（SMCI）			
LCD 控制器 / 驱动器			
乘除器和乘加器			
DMA 控制器			
上电复位功能			
电压检测功能			在 LVD 复位时可运行，在其他复位时停止运行。
外部中断			停止运行
CRC 运算功能	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			

注 在外部复位或者 POR 复位期间为高阻抗。在其他复位期间为高电平（连接内部上拉电阻）。

备注 f_{IH} : 高速内部振荡器时钟
 f_X : X1 振荡时钟
 f_{EX} : 外部主系统时钟
 f_{XT} : XT1 振荡时钟
 f_{EXS} : 外部副系统时钟
 f_{IL} : 低速内部振荡器时钟

表 22-2 接受复位后的各硬件状态

硬件		接受复位后的状态注
程序计数器（PC）		设定复位向量表（0000H、0001H）的内容。
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

注 在产生复位信号期间和等待振荡稳定期间的各硬件状态中，只有 PC 的内容不定，而其他的硬件状态保持复位后的状态。

备注 有关特殊功能寄存器（SFR: Special Function Register）在接受复位后的状态，请参照“3.1.4 特殊功能寄存器（SFR: Special Function Register）的区域 ”和“3.1.5 扩展特殊功能寄存器（2nd SFR: 2nd Special Function Register）的区域”。

22.3 确认复位源的寄存器

22.3.1 复位控制标志寄存器（RESF）

RL78 微控制器存在多种内部复位发生源。复位控制标志寄存器（RESF）保存发生复位请求的复位源。能通过 8 位存储器操作指令读 RESF 寄存器。

通过 RESET 的输入、上电复位（POR）电路的复位和 RESF 寄存器的读取，清除 TRAP、WDTRF、RPERF、IAWRF、LVIRF 标志。

图 22-4 复位控制标志寄存器（RESF）的格式

地址: FFFA8H

复位后: 不定注¹

R

符号	7	6	5	4	3	2	1	0
RESF	TRAP	0	0	WDTRF	0	RPERF	IAWRF	LVIRF

TRAP	因执行非法指令产生的内部复位请求注 ²
0	没有产生内部复位请求或者清除了 RESF 寄存器。
1	产生内部复位请求。

WDTRF	因看门狗定时器（WDT）产生的内部复位请求
0	没有产生内部复位请求或者清除了 RESF 寄存器。
1	产生内部复位请求。

RPERF	因 RAM 奇偶校验错误产生的内部复位请求
0	没有产生内部复位请求或者清除了 RESF 寄存器。
1	产生内部复位请求。

IAWRF	因存取非法存储器产生的内部复位请求
0	没有产生内部复位请求或者清除了 RESF 寄存器。
1	产生内部复位请求。

LVIRF	因电压检测电路（LVD）产生的内部复位请求
0	没有产生内部复位请求或者清除了 RESF 寄存器。
1	产生内部复位请求。

注 1. 因复位源而不同。请参照表 22-3。
2. 当执行指令码 FFH 时，发生非法指令错误。
在通过在线仿真器或者片上调试仿真器进行仿真时，不会因执行非法指令而产生复位。

注意 1. 不能通过 1 位存储器操作指令读数据。
2. 要允许产生 RAM 奇偶校验错误复位（RPERDIS=0）时，必须在存取数据时对“所用 RAM 区”的区域进行初始化，在执行 RAM 区的指令时对“所用 RAM 区 +10 字节”的区域进行初始化。
通过产生复位，允许产生 RAM 奇偶校验错误复位（RPERDIS=0）。详细内容请参照“25.3.3 RAM 奇偶校验错误检测功能”。

产生复位请求时的 RESF 寄存器状态如表 22-3 所示。

表 22-3 产生复位请求时的 RESF 寄存器的状态

复位源 标志	$\overline{\text{RESET}}$ 输入	因 POR 产 生的复位	因执行非法 指令产生的 复位	因 WDT 产 生的复位	因 RAM 奇 偶校验错误 产生的复位	因存取非法 存储器产生 的复位	因 LVD 产生 的复位
TRAP	清“0”	清“0”	置“1”	保持	保持	保持	保持
WDTRF			保持	置“1”			
RPERF				保持	置“1”		
IAWRF					保持	置“1”	
LVIRF						保持	置“1”

如果通过 8 位存储器操作指令读 RESF 寄存器，就自动清除 RESF 寄存器。
复位源的确认步骤如图 22-6 所示。

22.3.2 上电复位状态寄存器（PORSR）

PORSR 寄存器是用于确认发生上电复位的寄存器。
给 PORSR 寄存器的 bit0（PORF）写“1”有效，写“0”无效。
如果要确认有无发生上电复位，必须事先给 PORF 位写“1”。
通过 8 位存储器操作指令设定 PORSR 寄存器。
在产生上电复位信号后，此寄存器的值变为“00H”。

- 注意 1. 只在产生上电复位时对 PORSR 寄存器进行复位，在其他复位源引起的复位时保持原值不变。
2. 当 PORF 位为“1”时，可以保证没有发生上电复位，但是不能保证 RAM 的值保持不变。

图 22-5 上电复位状态寄存器（PORSR）的格式

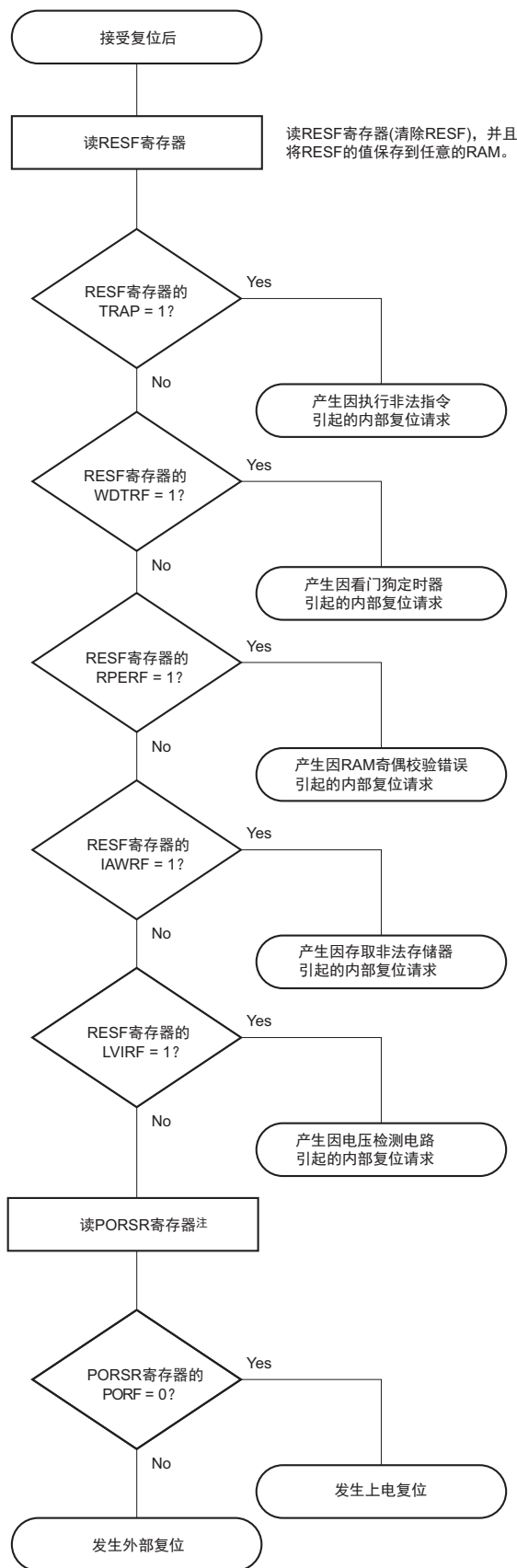
地址：F00F9H 复位后：00H R/W

符号	7	6	5	4	3	2	1	0
PORSR	0	0	0	0	0	0	0	PORF

PORF	确认有无发生上电复位
0	还未写“1”或者发生上电复位。
1	没有发生上电复位。

复位源的确认步骤如图 22-6 所示。

图 22-6 复位源的确认步骤



注 在接受复位前, 必须给 PORSR 寄存器的 bit0 (PORF) 写 “1”。

第 23 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）有以下功能。

- 在接通电源时产生内部复位信号。
如果电源电压（ V_{DD} ）大于检测电压（ V_{POR} ），就解除复位。但是，在达到“32.4 AC 特性”所示的工作电压范围前，必须通过电压检测电路或者外部复位保持复位状态。
- 将电源电压（ V_{DD} ）和检测电压（ V_{PDR} ）进行比较，当 $V_{DD} < V_{PDR}$ 时，产生内部复位信号。但是，在电源下降时，必须在小于“32.4 AC 特性”所示的工作电压范围前，转移到 STOP 模式，或者通过电压检测电路或外部复位设定为复位状态。如果要重新开始运行，必须确认电源电压已恢复到工作电压范围内。

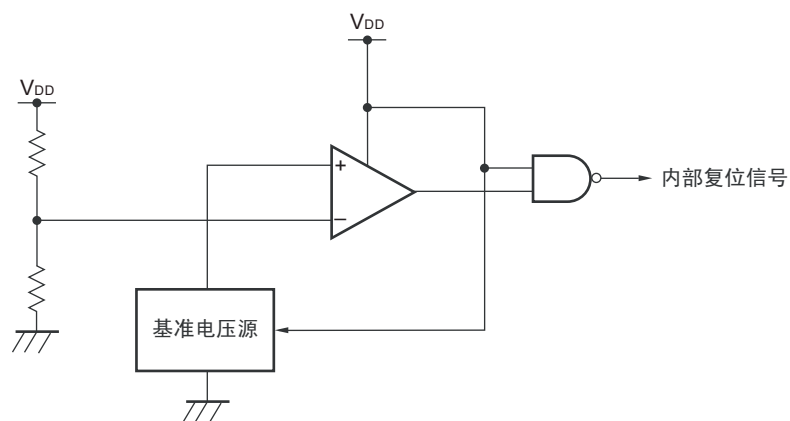
注意 当产生由上电复位电路引起的内部复位信号时，将复位控制标志寄存器（RESF）和上电复位状态寄存器（PORSR）清“00H”。

- 备注 1. R7F0C003、R7F0C004 内置多个产生内部复位信号的硬件。当由看门狗定时器（WDT）、电压检测（LVD）电路、执行非法指令的执行、RAM 奇偶校验错误或者非法存储器的存取而产生内部复位信号时，用于表示复位源的标志分配在复位控制标志寄存器（RESF）；当由 WDT、LVD、非法指令的执行、RAM 奇偶校验错误或者非法存储器的存取而产生内部复位信号时，不将 RESF 寄存器清“00H”而将标志置“1”。有关 RESF 寄存器的详细内容，请参照“第 22 章 复位功能”。
2. 可通过上电复位状态寄存器（PORSR）确认因上电复位电路产生的内部复位。有关 PORSR 寄存器的详细内容，请参照“第 22 章 复位功能”。

23.2 上电复位电路的结构

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

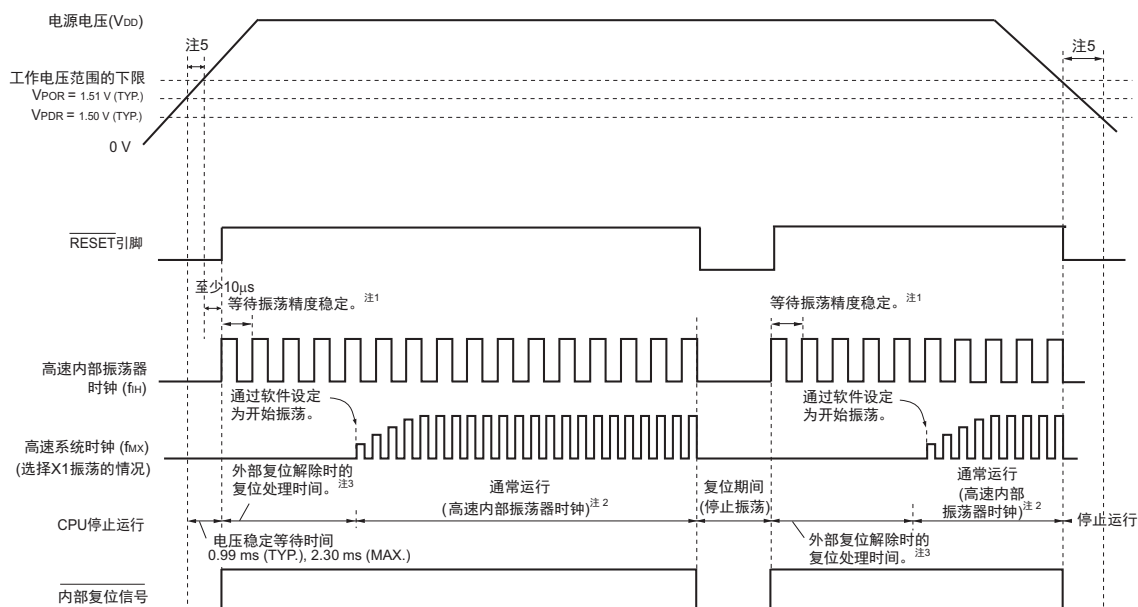


23.3 上电复位电路的运行

上电复位电路和电压检测电路的内部复位信号的产生时序如下所示。

图 23-2 上电复位电路和电压检测电路的内部复位信号的产生时序 (1/3)

(1) 使用由 RESET 引脚引起的外部复位时



- 注 1. 内部的复位处理时间包含高速内部振荡器时钟的振荡精度稳定等待时间。
2. 能将 CPU 时钟从高速内部振荡器时钟切换为高速系统时钟或者副系统时钟。在使用 X1 时钟的情况下，必须在通过振荡稳定时间计数器的状态寄存器（OSTC）确认振荡稳定时间后进行切换；在使用 XT1 时钟的情况下，必须在利用定时器功能等确认振荡稳定时间后进行切换。
3. 到开始通常运行的时间为除了达到 $V_{POR}(1.51V(TYP.))$ 后的“电压稳定等待时间”以外，还需要 $\overline{RESET}$ 信号变为高电平（1）后的如下的“外部复位解除时的复位处理时间”（POR 解除后的第 1 次）。外部复位解除时的复位处理时间如下所示。
- POR 解除后的第 1 次：0.672ms(TYP.)、0.832ms(MAX.)（使用 LVD 时）
0.399ms(TYP.)、0.519ms(MAX.)（LVD OFF 时）
4. POR 解除后的第 2 次以后（包括第 2 次）的外部复位解除时的复位处理时间如下所示。
- POR 解除后的第 2 次以后（包括第 2 次）的处理时间：0.531ms(TYP.)、0.675ms(MAX.)（使用 LVD 时）
0.259ms(TYP.)、0.362ms(MAX.)（LVD OFF 时）
5. 当电源上升时，在达到“32.4 AC 特性”所示的工作电压范围前，必须通过外部复位保持复位状态。当电源下降时，必须在小于工作电压范围前，转移到 STOP 模式，或者通过电压检测电路或外部复位设定为复位状态。如果要重新开始运行，必须确认电源电压已恢复到工作电压范围内。

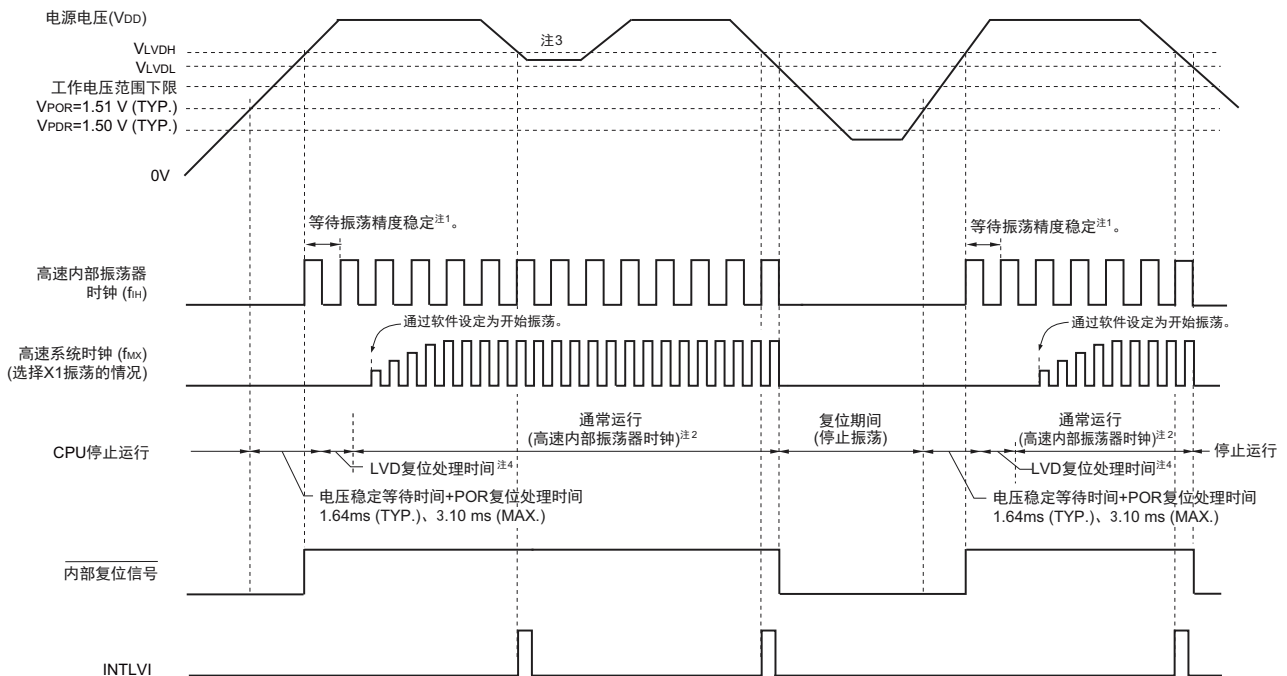
备注 V_{POR} ：POR 电源上升检测电压

V_{PDR} ：POR 电源下降检测电压

注意 当 LVD OFF 时，必须使用由 $\overline{RESET}$ 引脚引起的外部复位。详细内容请参照“第 24 章 电压检测电路”。

图 23-2 上电复位电路和电压检测电路的内部复位信号的产生时序 (2/3)

(2) LVD 为中断 & 复位模式的情况 (选项字节 000C1H 的 LVIMDS1、LVIMDS0=1、0)



注 1. 内部的复位处理时间包含高速内部振荡器时钟的振荡精度稳定等待时间。

2. 能将 CPU 时钟从高速内部振荡器时钟切换为高速系统时钟或者副系统时钟。在使用 X1 时钟的情况下，必须通过振荡稳定时间计数器的状态寄存器（OSTC）确认振荡稳定时间后进行切换；在使用 XT1 时钟的情况下，必须在利用定时器功能等确认振荡稳定时间后进行切换。

3. 在产生中断请求信号（INTLVI）后，自动将电压检测电平寄存器（LVIS）的LVIL位和LVIMD位置“1”。因此，考虑到电源电压在不低于低电压检测电压（ V_{LVDL} ）的情况下恢复到高电压检测电压（ V_{LVDH} ）或者以上时，必须在产生INTLVI后，根据“图24-8 工作电压的确认/复位的设定步骤”。

4. 到开始通常运行的时间除了达到 V_{POR} (1.51V(TYP.)) 后的“电压稳定等待时间 + POR 复位处理时间”以外, 还需要达到 LVD 检测电平 (V_{LVDH}) 后的如下的“LVD 复位处理时间”。

LVD 复位处理时间: 0ms ~ 0.0701ms(MAX.)

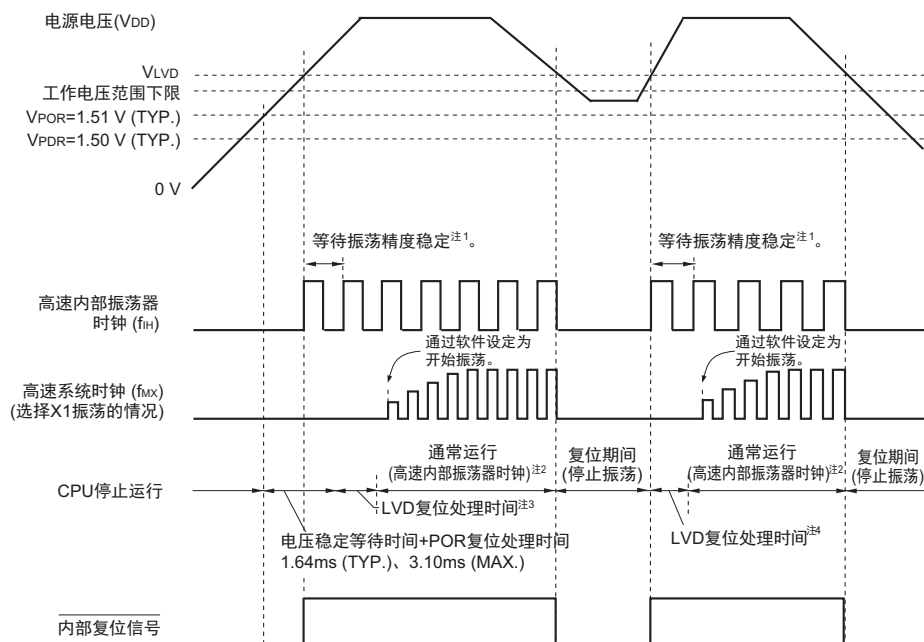
备注 V_{LVDH} 、 V_{LVDL} : LVD 检测电压

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的内部复位信号的产生时序 (3/3)

(3) LVD 复位模式的情况 (选项字节 000C1H 的 LVIMDS1、LVIMDS0=1、1)



注 1. 内部的复位处理时间包含高速内部振荡器时钟的振荡精度稳定等待时间。

2. 能将 CPU 时钟从高速内部振荡器时钟切换为高速系统时钟或者副系统时钟。在使用 X1 时钟的情况下，必须在通过振荡稳定时间计数器的状态寄存器 (OSTC) 确认振荡稳定时间后进行切换；在使用 XT1 时钟的情况下，必须在利用定时器功能等确认振荡稳定时间后进行切换。

3. 到开始通常运行的时间除了达到 V_{POR} (1.51V(TYP.)) 后的“电压稳定等待时间 + POR 复位处理时间”以外，还需要达到 LVD 检测电平 (V_{LVD}) 后的如下的“LVD 复位处理时间”。

LVD 复位处理时间: 0ms ~ 0.0701ms(MAX.)

4. 在电源电压下降时，如果只在发生电压检测电路 (LVD) 的内部复位后恢复电源电压，就需要达到 LVD 检测电平 (V_{LVD}) 后的如下的“LVD 复位处理时间”。

LVD 复位处理时间: 0.0511ms(TYP.)、0.0701ms(MAX.)

备注 1. V_{LVDH}、V_{LVDL}: LVD 检测电压

V_{POR}: POR 电源上升检测电压

V_{PDR}: POR 电源下降检测电压

2. 在选择 LVD 中断模式 (选项字节 000C1H 的 LVIMD1、LVIMD0=0、1) 时，从接通电源到开始通常运行所需要的时间与图 23-2(3) 的注 3 的时间相同。

第 24 章 电压检测电路

24.1 电压检测电路的功能

电压检测电路通过选项字节（000C1H）设定运行模式和检测电压（ V_{LVDH} 、 V_{LVDL} 、 V_{LVD} ）。电压检测（LVD）电路有以下功能。

- 将电源电压（ V_{DD} ）和检测电压（ V_{LVDH} 、 V_{LVDL} 、 V_{LVD} ）进行比较，产生内部复位或者内部中断信号。
- 电源电压的检测电压（ V_{LVDH} 、 V_{LVDL} 、 V_{LVD} ）能通过选项字节选择 14 种的检测电平（详细内容请参照“第 27 章 选项字节”）。
- 也能在 STOP 模式中运行。
- 当电源上升时，在达到“32.4 AC 特性”所示的工作电压范围前，必须通过电压检测电路或者外部复位保持复位状态。当电源下降时，必须在小于工作电压范围前，转移到 STOP 模式，或者通过电压检测电路或外部复位设定为复位状态。

工作电压范围根据用户选项字节（000C2H/010C2H）的设定而变。

(a) 中断 & 复位模式（选项字节的 LVIMDS1、LVIMDS0=1、0）

通过选项字节 000C1H 选择 2 个检测电压（ V_{LVDH} 、 V_{LVDL} ）。高电压检测电平（ V_{LVDH} ）用于解除复位 / 产生中断，低电压检测电平（ V_{LVDL} ）用于产生复位。

(b) 复位模式（选项字节的 LVIMDS1、LVIMDS0=1、1）

将选项字节 000C1H 选择的 1 个检测电压（ V_{LVD} ）用于产生 / 解除复位。

(c) 中断模式（选项字节的 LVIMDS1、LVIMDS0=0、1）

将选项字节 000C1H 选择的 1 个检测电压（ V_{LVD} ）用于解除复位 / 产生中断。

各模式中产生的中断请求信号和内部复位信号如下所示。

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)
在工作电压下降时，如果检测到 $V_{DD} < V_{LVDH}$ ，就产生中断请求信号；如果检测到 $V_{DD} < V_{LVDL}$ ，就产生内部复位。 如果检测到 $V_{DD} \geq V_{LVDH}$ ，就解除内部复位。	如果检测到 $V_{DD} \geq V_{LVD}$ ，就解除内部复位。如果检测到 $V_{DD} < V_{LVD}$ ，就产生中断请求信号。	在 POR 解除后的第 1 次工作电压上升时，如果检测到 $V_{DD} \geq V_{LVD}$ ，就解除内部复位。 从 POR 解除后的第 2 次工作电压上升开始，如果检测到 $V_{DD} < V_{LVD}$ 或者 $V_{DD} \geq V_{LVD}$ ，就产生中断请求信号。

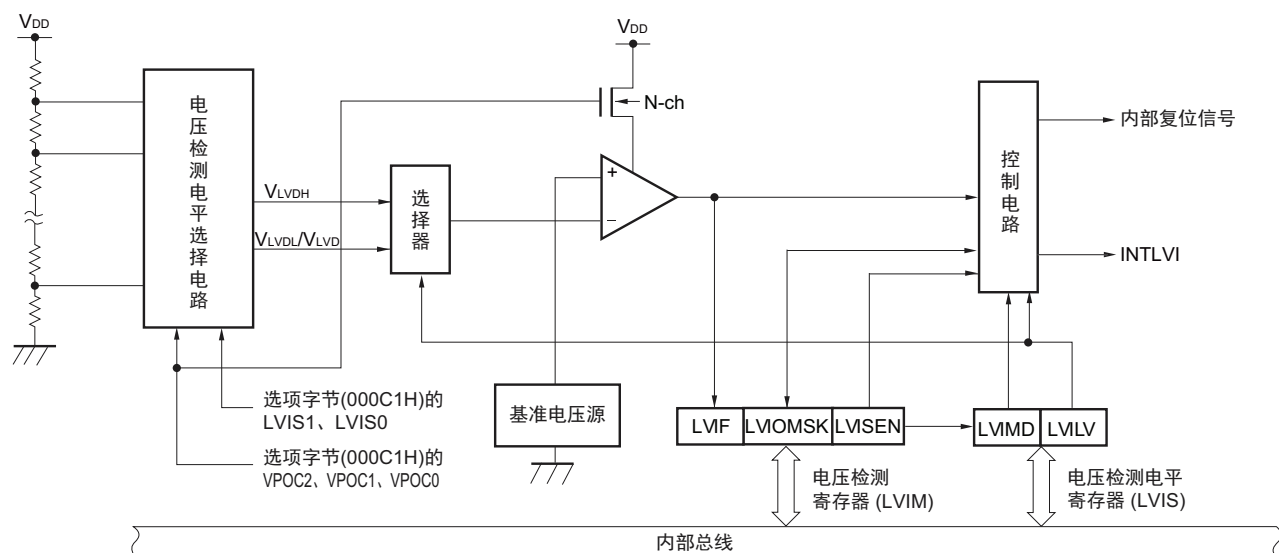
在电压检测电路运行时，能通过读电压检测标志（LVIF：电压检测寄存器（LVIM）的 bit0）来确认电源电压是大于等于检测电压还是小于检测电压。

如果发生复位，就将复位控制标志寄存器（RESF）的 bit0（LVIRF）置“1”。有关 RESF 寄存器的详细内容，请参照“第 22 章 复位功能”。

24.2 电压检测电路的结构

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 电压检测电路的控制寄存器

通过以下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）

24.3.1 电压检测寄存器（LVIM）

LVIM 寄存器设定允许或者禁止改写电压检测电平寄存器（LVIS），并且确认 LVD 输出的屏蔽状态。通过 1 位或者 8 位的存储器操作指令设定 LVIM 寄存器。
在产生复位信号后，此寄存器的值变为“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 注 3	0	0	0	0	0	LVIOMSK	LVIF

LVISEN 注 3	允许 / 禁止改写电压检测电平寄存器（LVIS）的设定
0	禁止改写 LVIS 寄存器（LVIOMSK 变为“0”（LVD 输出的屏蔽无效））。
1	允许改写 LVIS 寄存器（LVIOMSK 变为“1”（LVD 输出的屏蔽有效））。

LVIOMSK	LVD 输出的屏蔽状态标志
0	LVD 输出的屏蔽无效。
1	LVD 输出的屏蔽有效注 4。

LVIF	电压检测标志
0	电源电压（V _{DD} ）≥检测电压（V _{LVD} ）或者 LVD OFF 时。
1	电源电压（V _{DD} ）<检测电压（V _{LVD} ）

- 注 1. 复位值因复位源而变。
在 LVD 发生复位时，不对 LVIM 寄存器的值进行复位而保持原来的值；在其他复位时，将 LVISEN 位清“0”。
2. bit0 和 bit1 是只读位。
3. 只有在选择中断 & 复位模式（选项字节的 LVIMDS1 位和 LVIMDS0 位分别为“1”和“0”）时才能使用。在其他模式中，不能更改初始值。
4. 只有在选择中断 & 复位模式（选项字节的 LVIMDS1 位和 LVIMDS0 位分别为“1”和“0”）时，LVIOMSK 位才能在以下期间自动变为“1”，并且屏蔽由 LVD 引起的复位或者中断。
- 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
 - 从更改 LVILV 位的值到 LVD 检测电压稳定为止的等待时间

24.3.2 电压检测电平选择寄存器（LVIS）

这是设定电压检测电平的寄存器。
通过 1 位或者 8 位的存储器操作指令设定 LVIS 寄存器。
在产生复位信号后，此寄存器的值变为“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

电压检测的运行模式

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（V_{LVDH}）

1

低电压检测电平（V_{LVDL} 或者 V_{LVD}）

- 注 1. 复位值因复位源和选项字节的设定而变。
在发生 LVD 复位时，不将此寄存器清“00H”。
在发生 LVD 以外的复位时，此寄存器的值如下：
- 选项字节的 LVIMDS1、LVIMDS0=1、0 时：00H
 - 选项字节的 LVIMDS1、LVIMDS0=1、1 时：81H
 - 选项字节的 LVIMDS1、LVIMDS0=0、1 时：01H
2. 只有在选择中断 & 复位模式（选项字节的 LVIMDS1 位和 LVIMDS0 位分别为“1”和“0”）时才能写“0”。在其他情况下禁止设定。通过在中断 & 复位模式发生复位或者中断，自动进行值的替换。
- 注意 1. 根据图 24-8 的步骤改写 LVIS 寄存器。
2. 通过选项字节（000C1H）选择 LVD 的运行模式和各模式的检测电压（V_{LVDH}、V_{LVDL}、V_{LVD}）。选项字节（000C1H/010C1H）的格式如图 24-4 所示。有关选项字节的详细内容，请参照“第 27 章 选项字节”。

图 24-4 通过用户选项字节（000C1H）进行 LVD 运行模式和检测电压的设定（1/2）

地址：000C1H/010C1H 注

7	6	5	4	3	2	1	0
VPOC2	VPOC1	VPOC0	1	LVIS1	LVIS0	LVIMDS1	LVIMDS0

- LVD 的设定（中断 & 复位模式）

检测电压			选项字节的设定值						
V _{LVDH}		V _{LVDL}	VPOC2	VPOC1	VPOC0	LVIS1	LVIS0	模式设定	
上升	下降	下降						LVIMDS1	LVIMDS0
1.77V	1.73V	1.63V	0	0	0	1	0	1	0
1.88V	1.84V					0	1		
2.92V	2.86V					0	0		
1.98V	1.94V	1.84V		0	1	1	0		
2.09V	2.04V					0	1		
3.13V	3.06V					0	0		
2.61V	2.55V	2.45V		1	0	1	0		
2.71V	2.65V					0	1		
3.75V	3.67V					0	0		
2.92V	2.86V	2.75V	1	1	1	0			
3.02V	2.96V				0	1			
4.06V	3.98V				0	0			
—			上述以外，禁止设定。						

- LVD 的设定（复位模式）

检测电压		选项字节的设定值						
V _{LVD}		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0	模式设定	
上升	下降						LVIMDS1	LVIMDS0
1.67V	1.63V	0	0	0	1	1	1	1
1.77V	1.73V		0	0	1	0		
1.88V	1.84V		0	1	1	1		
1.98V	1.94V		0	1	1	0		
2.09V	2.04V		0	1	0	1		
2.50V	2.45V		1	0	1	1		
2.61V	2.55V		1	0	1	0		
2.71V	2.65V		1	0	0	1		
2.81V	2.75V		1	1	1	1		
2.92V	2.86V		1	1	1	0		
3.02V	2.96V		1	1	0	1		
3.13V	3.06V		0	1	0	0		
3.75V	3.67V		1	0	0	0		
4.06V	3.98V		1	1	0	0		
—		上述以外，禁止设定。						

注 在引导交换时，000C1H 被 010C1H 替换，因此必须给 010C1H 设定和 000C1H 相同的值。

备注 检测电压为 TYP. 值。详细内容请参照“32.6.5 LVD 电路特性”。

图 24-4 通过用户选项字节（000C1H）进行 LVD 运行模式和检测电压的设定（2/2）

地址：000C1H/010C1H 注

7	6	5	4	3	2	1	0
VPOC2	VPOC1	VPOC0	1	LVIS1	LVIS0	LVIMDS1	LVIMDS0

- LVD 的设定（中断模式）

检测电压		选项字节的设定值						
V _{LVD}		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0	模式设定	
上升	下降						LVIMDS1	LVIMDS0
1.67V	1.63V	0	0	0	1	1	0	1
1.77V	1.73V		0	0	1	0		
1.88V	1.84V		0	1	1	1		
1.98V	1.94V		0	1	1	0		
2.09V	2.04V		0	1	0	1		
2.50V	2.45V		1	0	1	1		
2.61V	2.55V		1	0	1	0		
2.71V	2.65V		1	0	0	1		
2.81V	2.75V		1	1	1	1		
2.92V	2.86V		1	1	1	0		
3.02V	2.96V		1	1	0	1		
3.13V	3.06V		0	1	0	0		
3.75V	3.67V		1	0	0	0		
4.06V	3.98V		1	1	0	0		
—		上述以外，禁止设定。						

- LVD OFF（使用由 RESET 引脚引起的外部复位）

检测电压		选项字节的设定值						
V_{LVD}		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0	模式设定	
上升	下降						LVIMDS1	LVIMDS0
—	—	1	×	×	×	×	×	1
—		上述以外，禁止设定。						

注 在引导交换时，000C1H 被 010C1H 替换，因此必须给 010C1H 设定和 000C1H 相同的值。

注意 1. 必须给 bit4 写“1”。

- 当电源上升时，在达到“32.4 AC 特性”所示的工作电压范围前，必须通过电压检测电路或者外部复位保持复位状态。当电源下降时，必须在小于工作电压范围前，转移到 STOP 模式，或者通过电压检测电路或外部复位设定为复位状态。

工作电压范围根据用户选项字节（000C2H/010C2H）的设定而变。

备注 1. ×：忽略

- 检测电压为 TYP. 值。详细内容请参照“32.6.5 LVD 电路特性”。

24.4 电压检测电路的运行

24.4.1 用作复位模式时的设定

通过选项字节 000C1H 设定运行模式（复位模式（LVIMDS1、LVIMDS0=1、1））和检测电压（ V_{LVD} ）。

当设定为复位模式时，在以下的初始设定状态下开始运行。

- 将电压检测寄存器（LVIM）的bit7（LVISEN）置“0”（禁止改写电压检测电平寄存器（LVIS））。
- 将电压检测电平寄存器（LVIS）的初始值置“81H”。
bit7（LVIMD）为“1”（复位模式）。
bit0（LVILV）为“1”（低电压检测电平： V_{LVD} ）。

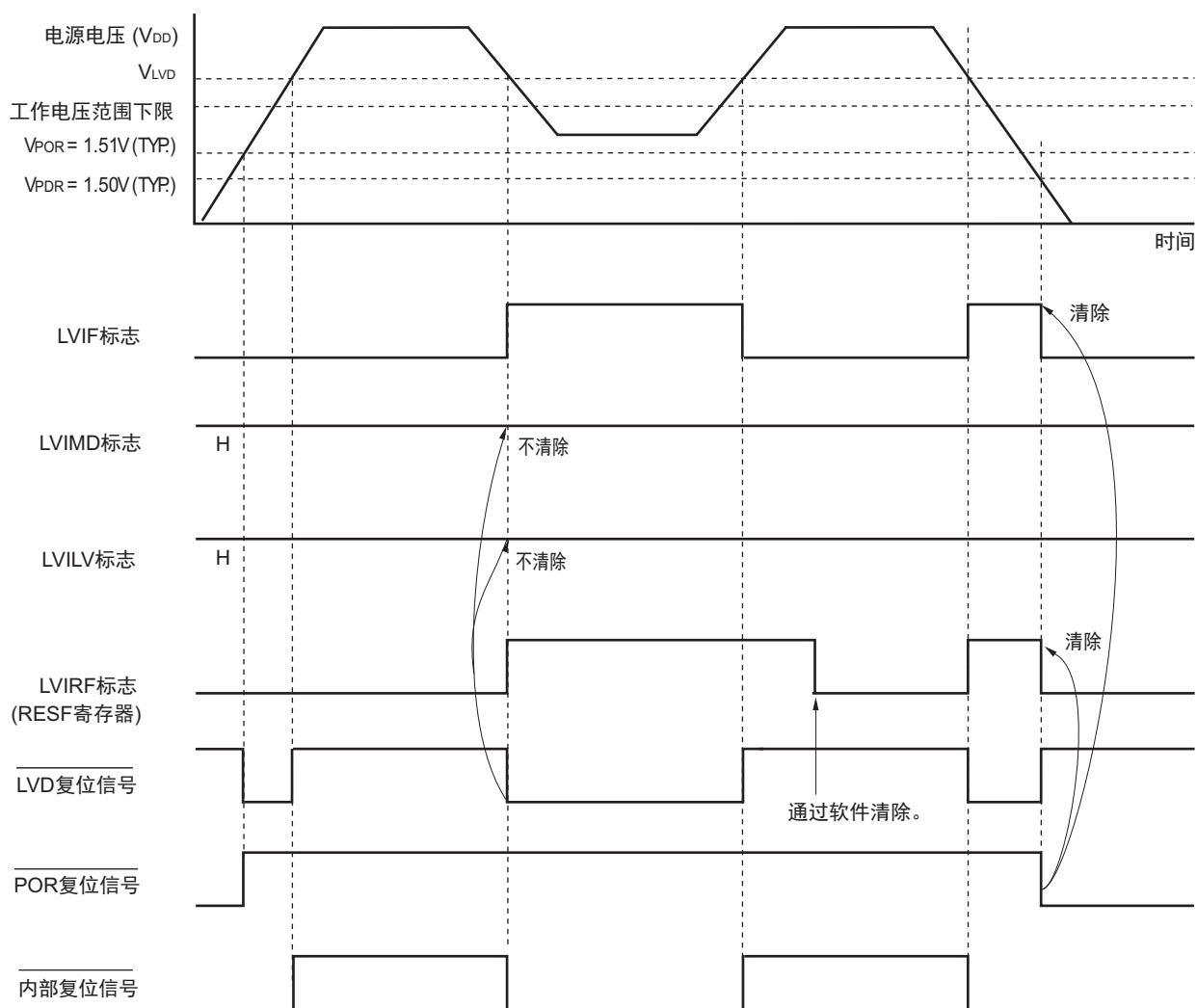
● LVD 复位模式的运行

当接通电源后，在电源电压（ V_{DD} ）大于电压检测电平（ V_{LVD} ）前，复位模式（选项字节的 LVIMDS1、LVIMDS0=1、1）保持由LVD引起的内部复位。一旦电源电压（ V_{DD} ）大于电压检测电平（ V_{LVD} ），就解除内部复位。

当工作电压下降时，一旦电源电压（ V_{DD} ）小于电压检测电平（ V_{LVD} ），就发生由LVD引起的内部复位。

LVD 复位模式的内部复位信号的产生时序如图 24-5 所示。

图 24-5 内部复位信号的产生时序（选项字节的 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 用作中断模式时的设定

通过选项字节 000C1H 设定运行模式（中断模式（LVIMDS1、LVIMDS0=0、1））和检测电压（ V_{LVD} ）。

当设定为中断模式时，在以下的初始设定状态下开始运行。

- 将电压检测寄存器（LVIM）的bit7（LVISEN）置“0”（禁止改写电压检测电平寄存器（LVIS））。
- 将电压检测电平寄存器（LVIS）的初始值置“01H”。
bit7（LVIMD）为“0”（中断模式）。
bit0（LVILV）为“1”（低电压检测电平： V_{LVD} ）。

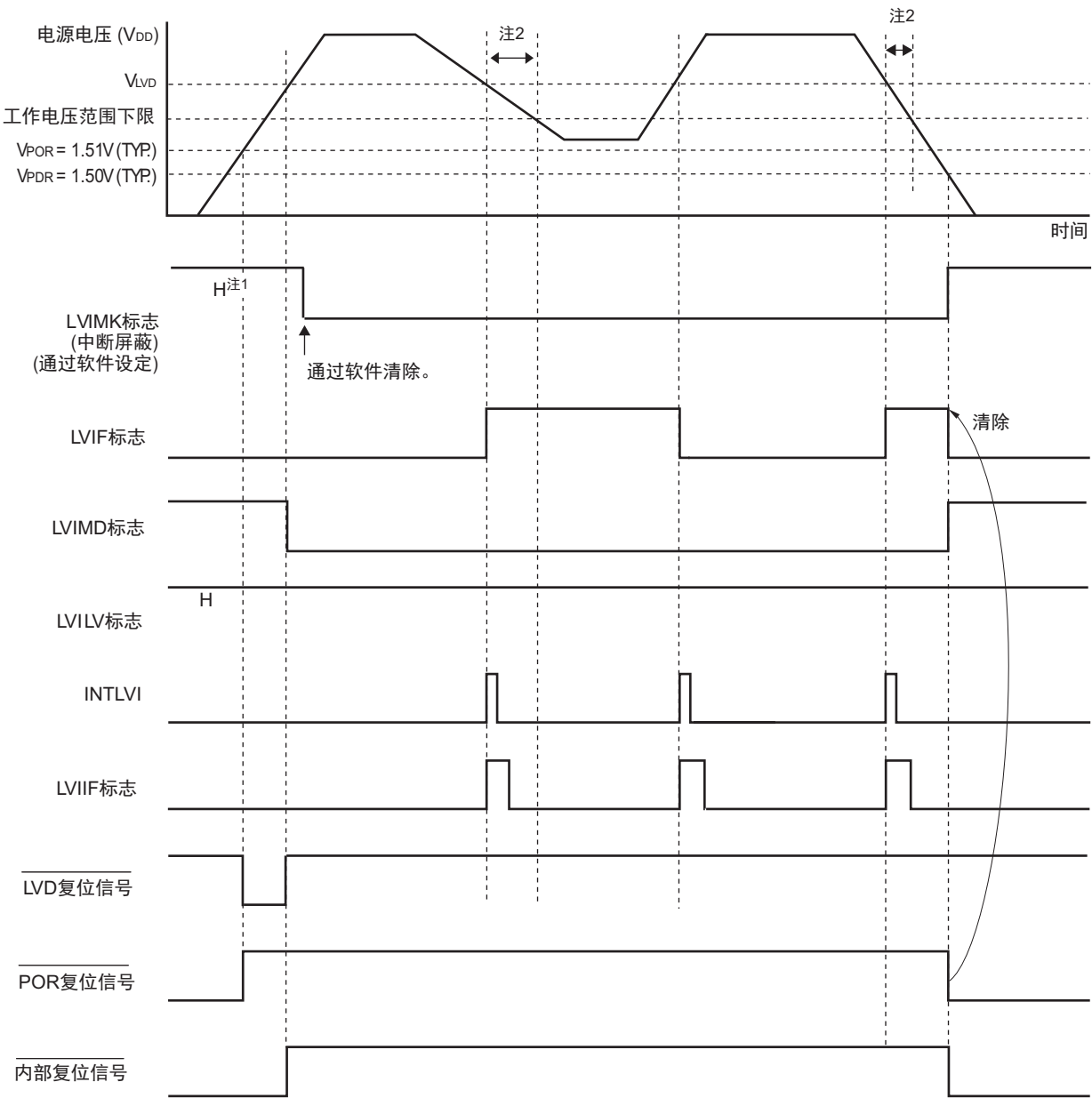
● LVD 中断模式的运行

在发生复位后，LVD 的内部复位保持复位状态直到 $V_{DD} \geq V_{LVD}$ 前。一旦检测到 $V_{DD} \geq V_{LVD}$ ，就解除 LVD 的内部复位。

此后，如果检测到 $V_{DD} < V_{LVD}$ 或者 $V_{DD} \geq V_{LVD}$ ，就产生中断请求信号（INTLVI）。

LVD 中断模式的中断请求信号的产生时序如图 24-6 所示。

图 24-6 中断信号的产生时序（选项字节的 LVIMDS1、LVIMDS0=0、1）



- 注 1. 在产生复位信号后，LVIMK 标志变为“1”。
2. 当工作电压下降时，必须在小于“32.4 AC 特性”所示的工作电压范围前，转移到 STOP 模式，或者通过外部复位设定为复位状态。如果要重新开始运行，必须确认电源电压已恢复到工作电压范围内。

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 用作中断 & 复位模式时的设定

通过选项字节 000C1H 设定运行模式（中断 & 复位模式（LVIMDS1、LVIMDS0=1、0））和检测电压（ V_{LVDH} 、 V_{LVDL} ）。

当设定为中断 & 复位模式时，在以下的初始设定状态下开始运行。

- 将电压检测寄存器（LVIM）的 bit7（LVISEN）置“0”（禁止改写电压检测电平寄存器（LVIS））。
- 将电压检测电平寄存器（LVIS）的初始值置“00H”。
bit7（LVIMD）为“0”（中断模式）。
bit0（LVILV）为“0”（高电压检测电平： V_{LVDH} ）。

● LVD 中断 & 复位模式的运行

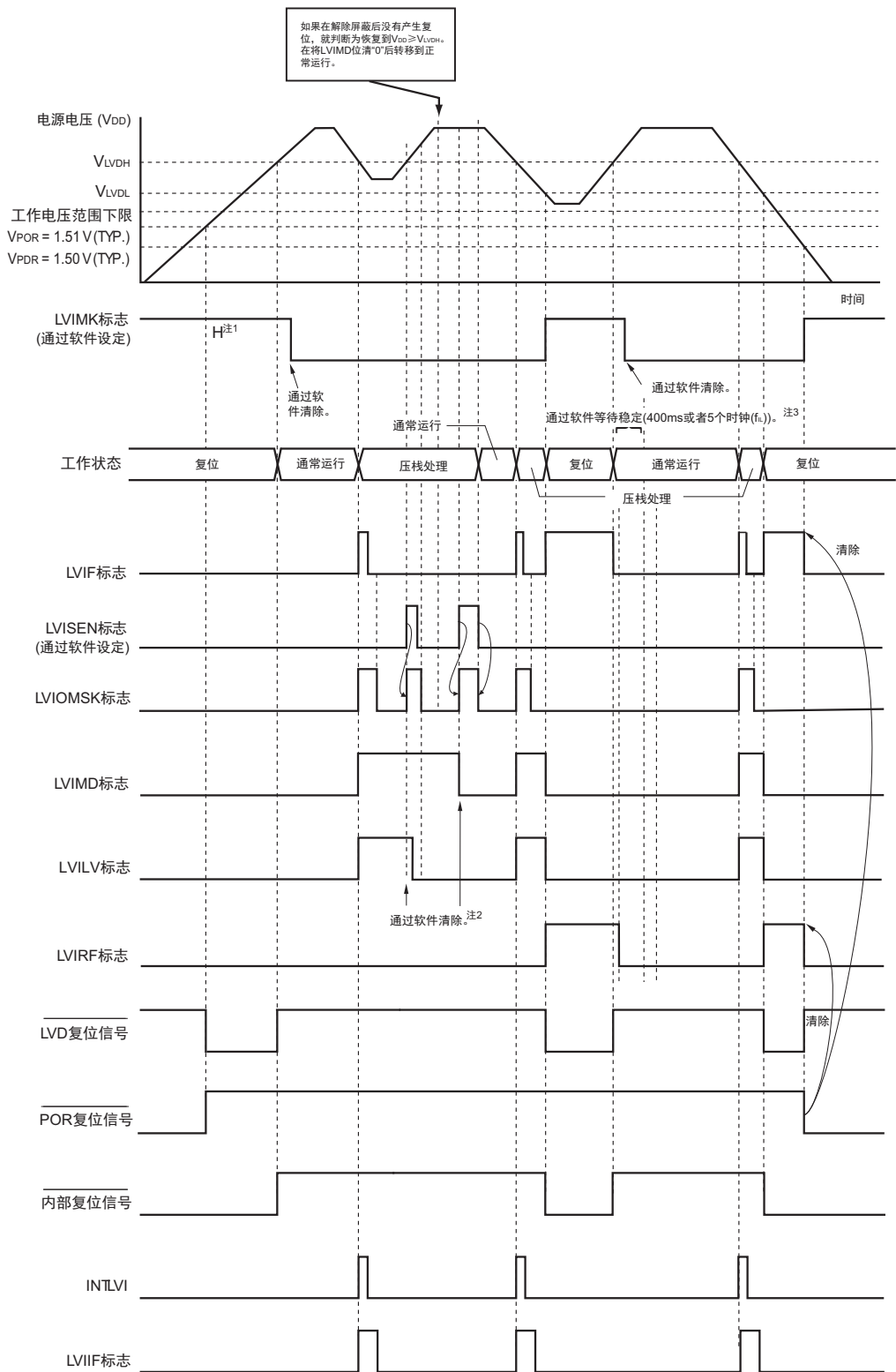
当接通电源后，在电源电压（ V_{DD} ）大于高电压检测电平（ V_{LVDH} ）前，中断 & 复位模式（选项字节的 LVIMDS1、LVIMDS0=1、0）保持由 LVD 引起的内部复位状态。一旦电源电压（ V_{DD} ）大于高电压检测电平（ V_{LVDH} ），就解除内部复位。

当工作电压下降时，一旦电源电压（ V_{DD} ）小于高电压检测电平（ V_{LVDH} ），就产生由 LVD 引起的中断请求信号（INTLVI），并且能执行任意的压栈处理。此后，当电源电压（ V_{DD} ）小于低电压检测电平（ V_{LVDL} ），就发生由 LVD 引起的内部复位。但是，如果在产生 INTLVI 后，电源电压（ V_{DD} ）不小于低电压检测电平（ V_{LVDL} ），并且又恢复到大于高电压检测电平（ V_{LVDH} ），也不产生中断请求信号。

如果要使用 LVD 中断 & 复位模式，必须按照“图 24-8 工作电压的确认/复位的设定步骤”所示的流程图的步骤进行设定。

LVD 中断 & 复位模式的内部复位信号和中断信号的产生时序如图 24-7 所示。

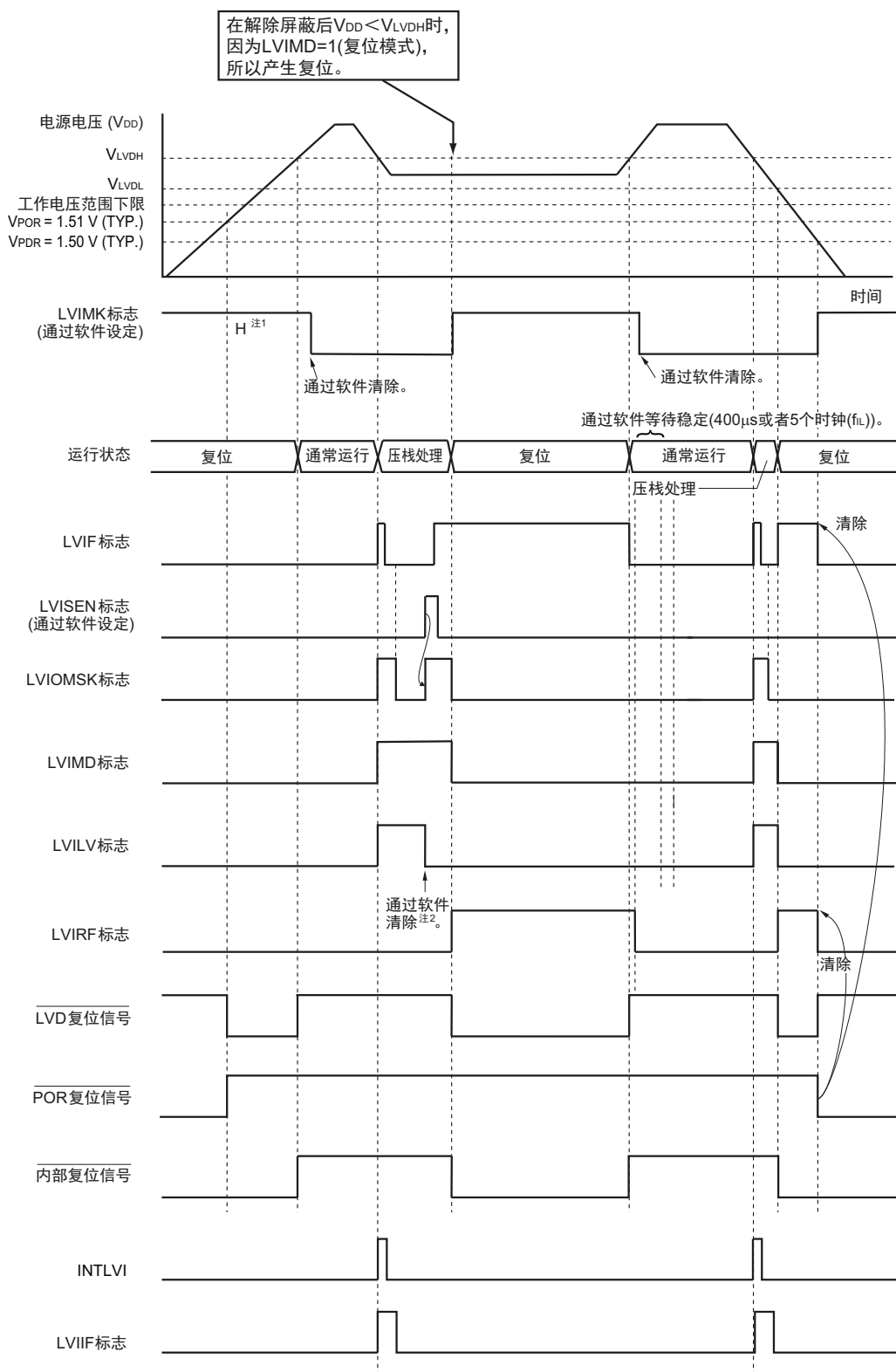
图 24-7 复位 & 中断信号的产生时序（选项字节的 LVIMDS1、 LVIMDS0=1、 0）（1/2）



注 1. 在产生复位信号后， LVIMK 标志变为“1”。
2. 当使用中 断 & 复位模式时， 必须在发生中断后按照“ 图 24-8 工作电压的确认 / 复位的设定步骤” 进行设定。

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 中断 & 复位信号的产生时序 (选项字节的 LVIMDS1、LVIMDS0=1、0) (2/2)



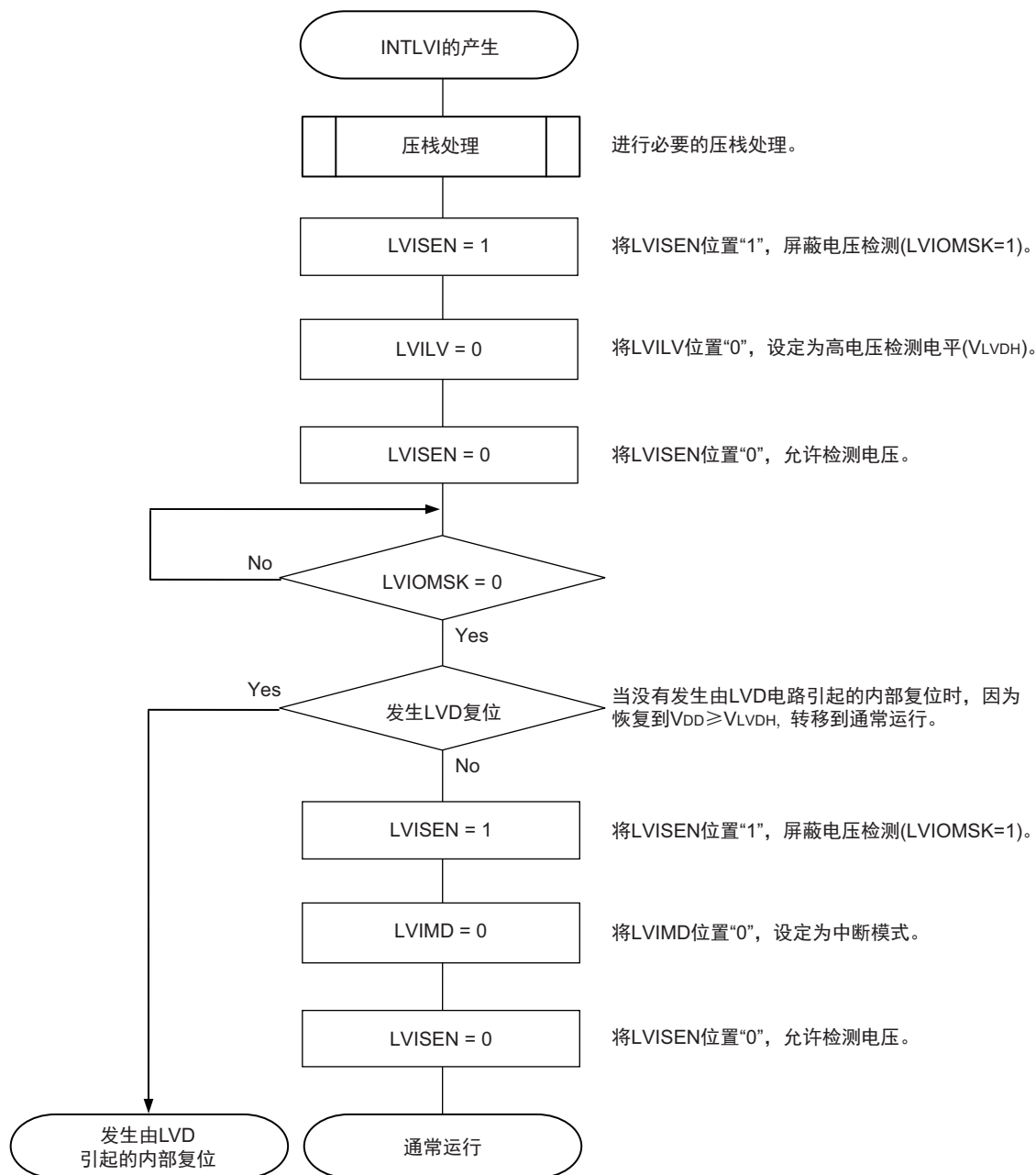
注 1. 在产生复位信号后, LVIMK 标志变为“1”。

2. 当使用中断 & 复位模式时, 必须在发生中断后按照“图 24-8 工作电压的确认 / 复位的设定步骤”进行设定。

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 24-8 工作电压的确认 / 复位的设定步骤



24.5 电压检测电路的注意事项

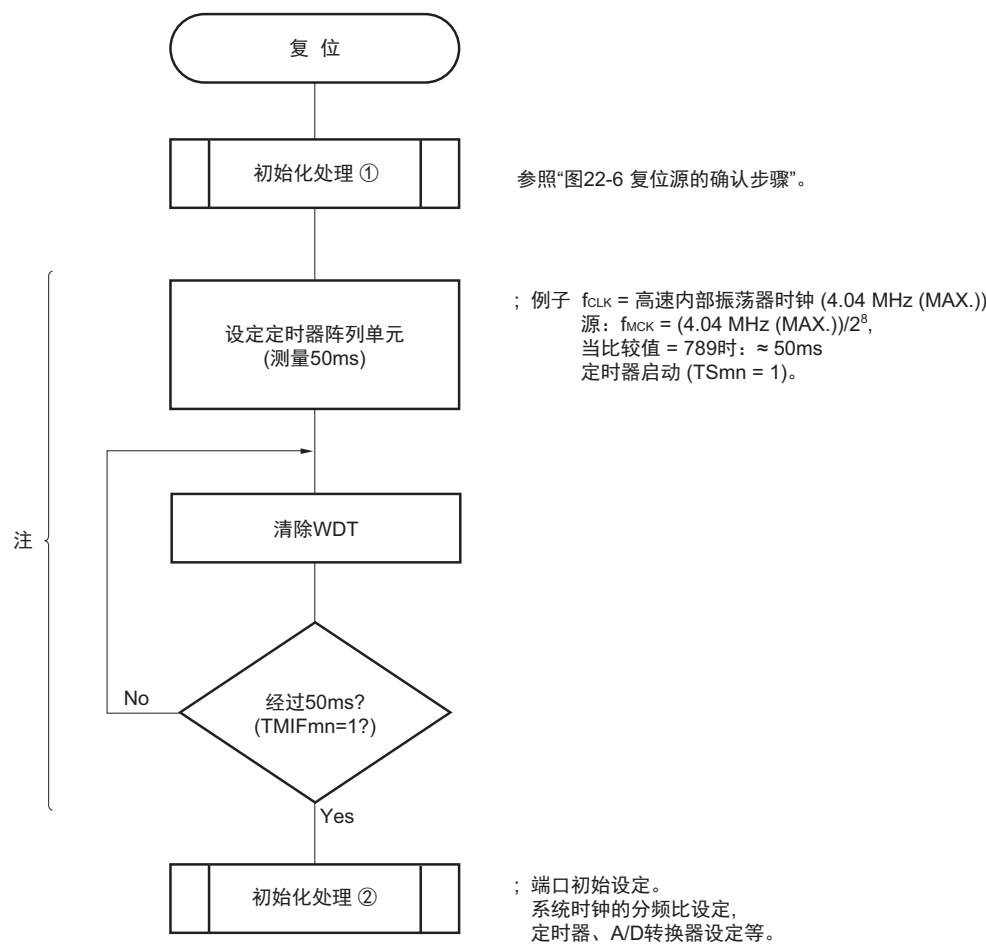
(1) 有关接通电源时的电压波动

某期间的电源电压（V_{DD}）在 LVD 检测电压附近波动的系统中，单片机可能会重复地进入复位状态 / 复位解除状态。通过以下方法，可以任意设定从复位解除到单片机开始工作的时间。

< 处理 >

复位解除后，通过使用定时器等软件计数器，等待因系统而不同的电源电压波动期间过后，对端口等进行初始设定。

图 24-9 当 LVD 检测电压附近的电源电压波动不超过 50ms 时的软件处理例子



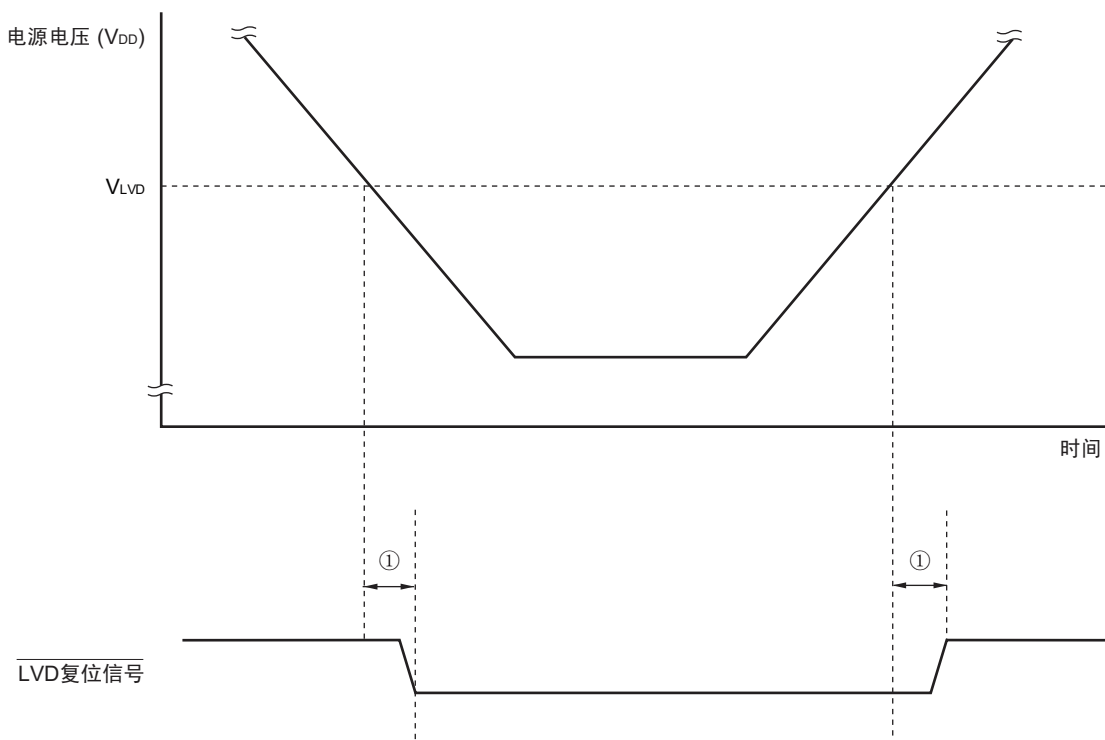
注 在此期间再次发生复位时，不转移到初始化处理②。

备注 m=0、n=0 ~ 7

(2) 从产生 LVD 复位源到产生或者解除 LVD 复位的延迟

从满足电源电压 (V_{DD}) < LVD 检测电压 (V_{LVD}) 到产生 LVD 复位为止会发生延迟。同样, 从 LVD 检测电压 (V_{LVD}) ≤ 电源电压 (V_{DD}) 到解除 LVD 复位为止也会发生延迟 (参照图 24-10)。

图 24-10 从产生 LVD 复位源到产生或者解除 LVD 复位的延迟



①: 检测延迟 (300 μ s (MAX.))

(3) 有关设定为 LVD OFF 时的电源上升

在设定为 LVD OFF 时, 必须使用由 RESET 引脚引起的外部复位。

如果要执行外部复位, 必须给 RESET 引脚输入至少 10 μ s 的低电平。如果要在电源上升时执行外部复位, 必须在给 RESET 引脚输入低电平后接通电源, 然后在“32.4 AC 特性”所示的工作电压范围内继续输入至少 10 μ s 的低电平后再输入高电平。

(4) 有关设定为 LVD OFF 以及 LVD 中断模式时的工作电压下降的情况

当设定为 LVD OFF 以及 LVD 中断模式时, 在工作电压下降至小于“32.4 AC 特性”所示的工作电压范围前, 转移到 STOP 模式, 或者通过外部复位设定为复位状态。如果要重新开始运行, 必须确认电源电压已恢复到工作电压范围内。

第 25 章 安全功能

25.1 安全功能的概要

为了对应 IEC60730 和 IEC61508 安全标准，R7F0C003、R7F0C004 内置以下安全功能。
此安全功能的目的是通过单片机的自诊断，在检测到故障时安全地停止工作。

(1) 闪存 CRC 运算功能（高速 CRC、通用 CRC）

通过 CRC 运算检测闪存的数据错误。

能根据不同的用途和使用条件，分别使用以下 2 个 CRC。

- “高速 CRC”...在初始化程序中，能停止 CPU 的运行并且高速检查整个代码闪存区。
- “通用 CRC”...在 CPU 运行中，不限于代码闪存区而能用于多用途的检查。

(2) RAM 奇偶校验错误检测功能

在读 RAM 数据时，检测奇偶校验错误。

(3) RAM 保护功能

防止因 CPU 失控而改写 RAM 数据。

(4) SFR 保护功能

防止因 CPU 失控而改写 SFR。

(5) 非法存储器存取检测功能

检测对非法存储器区域（没有存储器的区域或者存取受限的区域）的非法存取。

(6) 频率检测功能

能使用定时器阵列单元自检测 CPU/ 外围硬件时钟频率。

(7) A/D 测试功能

通过对 A/D 转换器的正（+）基准电压、负（-）基准电压、模拟输入通道（ANI）、温度传感器输出电压以及内部基准电压进行 A/D 转换来对 A/D 转换器进行自检测。

(8) 输入 / 输出引脚的数字输出信号电平检测功能

在输出模式中，能读输入 / 输出引脚的输出电平。

备注 有关符合安全标准 IEC60730 的安全功能的使用例子，请参照《RL78 Family VDE Certified IEC60730/60335 Self Test Library》Application Note（R01AN0749）。

25.2 安全功能使用的寄存器

安全功能的各功能使用以下寄存器。

寄存器	安全功能的各功能
- 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)
- CRC 输入寄存器 (CRCIN) - CRC 数据寄存器 (CRCD)	CRC 运算功能 (通用 CRC)
RAM 奇偶校验错误控制寄存器 (RPECTL)	RAM 奇偶校验错误检测功能
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
定时器输入选择寄存器 0 (TIS0)	频率检测功能
A/D 测试寄存器 (ADTES)	A/D 测试功能
端口模式选择寄存器 (PMS)	输入 / 输出引脚的数字输出信号电平检测功能

有关各寄存器的内容，在“25.3 安全功能的运行”中进行说明。

25.3 安全功能的运行

25.3.1 闪存 CRC 运算功能（高速 CRC）

IEC60730 标准要求确认闪存内的数据，并且建议 CRC 为确认手段。此高速 CRC 能在初始设定（初始化）程序中检查整个代码闪存区。只能在通过 RAM 内的程序以主系统时钟运行的 HALT 模式中执行高速 CRC。

高速 CRC 停止 CPU 的运行并且以 1 个时钟从闪存读取 32 位数据进行运算。因此，其特点是完成检查的时间短（例如，32KB 闪存：341μs@24MHz）。

CRC 生成多项式对应 CRC-16-CCITT 的“ $X^{16}+X^{12}+X^5+1$ ”。

以 bit31→bit0 的 MSB 优先进行运算。

注意 在片上调试时，因为配置监视程序，所以会出现不同的 CRC 运算结果。

备注 因为通用 CRC 为 LSB 优先，所以运算结果不同。

25.3.1.1 闪存 CRC 控制寄存器 (CRC0CTL)

这是设定高速 CRC 运算器的运行控制和运算范围的寄存器。
 能通过 1 位或者 8 位存储器操作指令设定 CRC0CTL 寄存器。
 在产生复位信号后，此寄存器的值变为“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	0	0	0	FEA2	FEA1	FEA0

CRC0EN	高速 CRC 运算器的运行控制
0	停止运行。
1	通过执行 HALT 指令开始运算。

FEA2	FEA1	FEA0	高速 CRC 的运算范围
0	0	0	00000H ~ 3FFBH (16K-4 字节)
0	0	1	00000H ~ 7FFBH (32K-4 字节)
0	1	0	00000H ~ BFFBH (48K-4 字节)
0	1	1	00000H ~ FFFBH (64K-4 字节)
1	0	0	00000H ~ 13FFBH (80K-4 字节)
1	0	1	00000H ~ 17FFBH (96K-4 字节)
1	1	0	00000H ~ 1BFFBH (112K-4 字节)
1	1	1	00000H ~ 1FFFH (128K-4 字节)

备注 必须事先将用于比较的 CRC 运算结果期待值存入闪存的最后 4 字节，因此运算范围为减去 4 字节的范围。

25.3.1.2 闪存 CRC 运算结果寄存器 (PGCRCL)

这是保存高速 CRC 运算结果的寄存器。
 通过 16 位存储器操作指令设定 PGCRCL 寄存器。
 在产生复位信号后，此寄存器的值变为“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

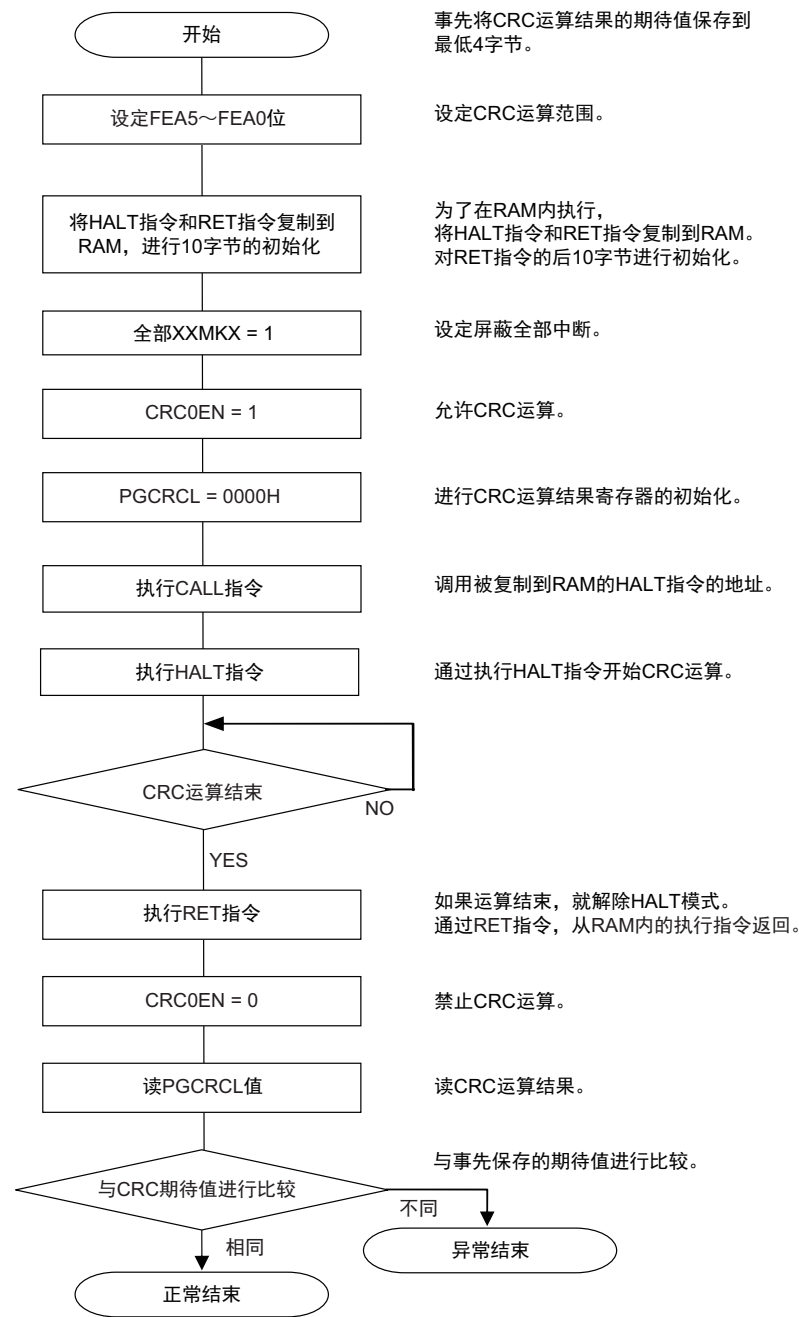
PGCRC15 ~ 0	高速 CRC 的运算结果
0000H ~ FFFFH	保存高速 CRC 的运算结果。

注意 只有在 CRC0EN (CRC0CTL 寄存器的 bit7) 位为“1”时才能写 PGCRCL 寄存器。

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 1. 只以代码闪存为 CRC 运算的对象。
2. 必须将 CRC 运算的期待值保存到代码闪存内运算范围后的区域。
3. 通过在 RAM 区执行 HALT 指令，CRC 运算变为有效。
- 必须在 RAM 区执行 HALT 指令。

能通过使用综合开发环境“CubeSuite+”计算 CRC 运算的期待值。有关详细内容，请参照《Integrated Development Environment CubeSuite+ User's Manual》。

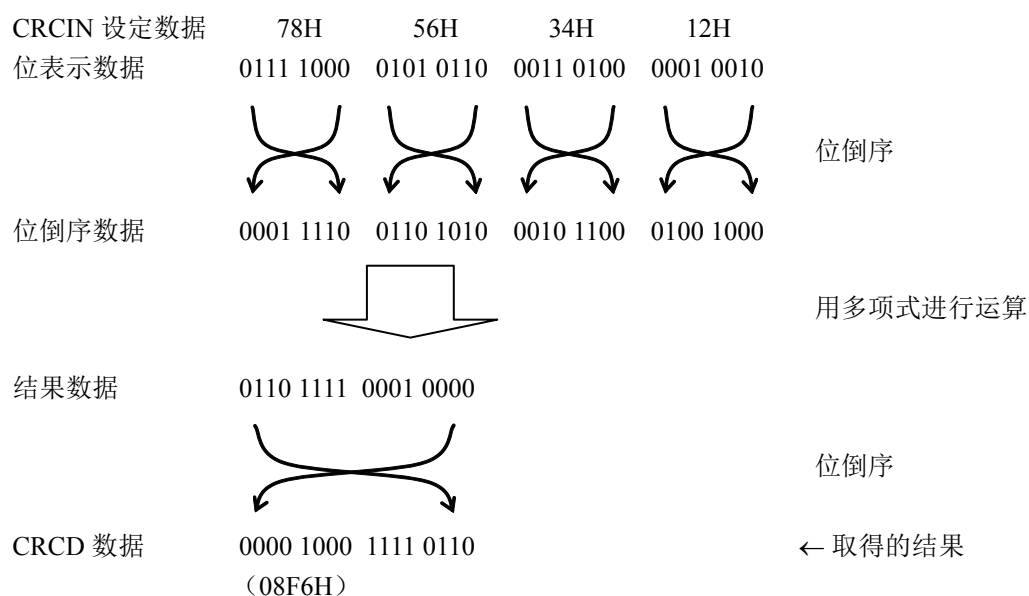
25.3.2 CRC 运算功能（通用 CRC）

为了必须保证运行过程中的安全，IEC61508 标准要求即使在 CPU 运行中也需要确认数据。

此通用 CRC 能在 CPU 运行中作为外围功能进行 CRC 运算。通用 CRC 不限于代码闪存区而能用于多用途的检查。通过软件（用户程序）指定要确认的数据。HALT 模式中的 CRC 运算功能只能在 DMA 传送过程中使用。

也能在主系统时钟运行模式和副系统时钟运行模式中使用通用 CRC。

CRC 生成多项式使用 CRC-16-CCITT 的“ $X^{16}+X^{12}+X^5+1$ ”。因为考虑到是以 LSB 优先进行的通信，所以在将输入数据的位序颠倒后进行计算。例如，从 LSB 发送数据“12345678H”的情况，按照“78H”、“56H”、“34H”、“12H”的顺序给 CRCIN 寄存器写值，从 CRCD 寄存器得到“08F6H”的值。这是针对颠倒了数据“12345678H”的位序后的以下位序进行 CRC 运算的结果。



注意 在执行程序的过程中, 因为调试程序将软件断点的设定行改写为 BREAK 指令, 所以如果在 CRC 运算的对象区设定软件断点, 就会出现不同的 CRC 运算结果。

25.3.2.1 CRC 输入寄存器 (CRCIN)

这是设定通用 CRC 的 CRC 计算数据的 8 位寄存器。

能设定的范围为“00H ~ FFH”。

通过 8 位存储器操作指令设定 CRCIN 寄存器。

在产生复位信号后，此寄存器的值变为“00H”。

图 25-4 CRC 输入寄存器 (CRCIN) 的格式

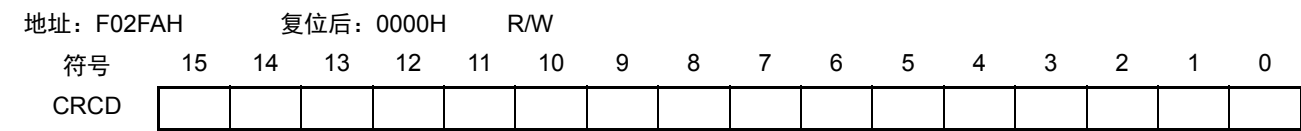
地址: FFFACH	复位后: 00H	R/W						
符号	7	6	5	4	3	2	1	0
CRCIN								

bit7 ~ 0	功能
00H ~ FFH	数据输入

25.3.2.2 CRC 数据寄存器 (CRCD)

这是保存通用 CRC 运算结果的寄存器。
能设定的范围为“0000H ~ FFFFH”。
在写 CRCIN 寄存器开始经过 1 个 CPU/ 外围硬件时钟 (f_{CLK}) 后, 将 CRC 运算结果保存到 CRCD 寄存器。
通过 16 位存储器操作指令设定 CRCD 寄存器。
在产生复位信号后, 此寄存器的值变为“0000H”。

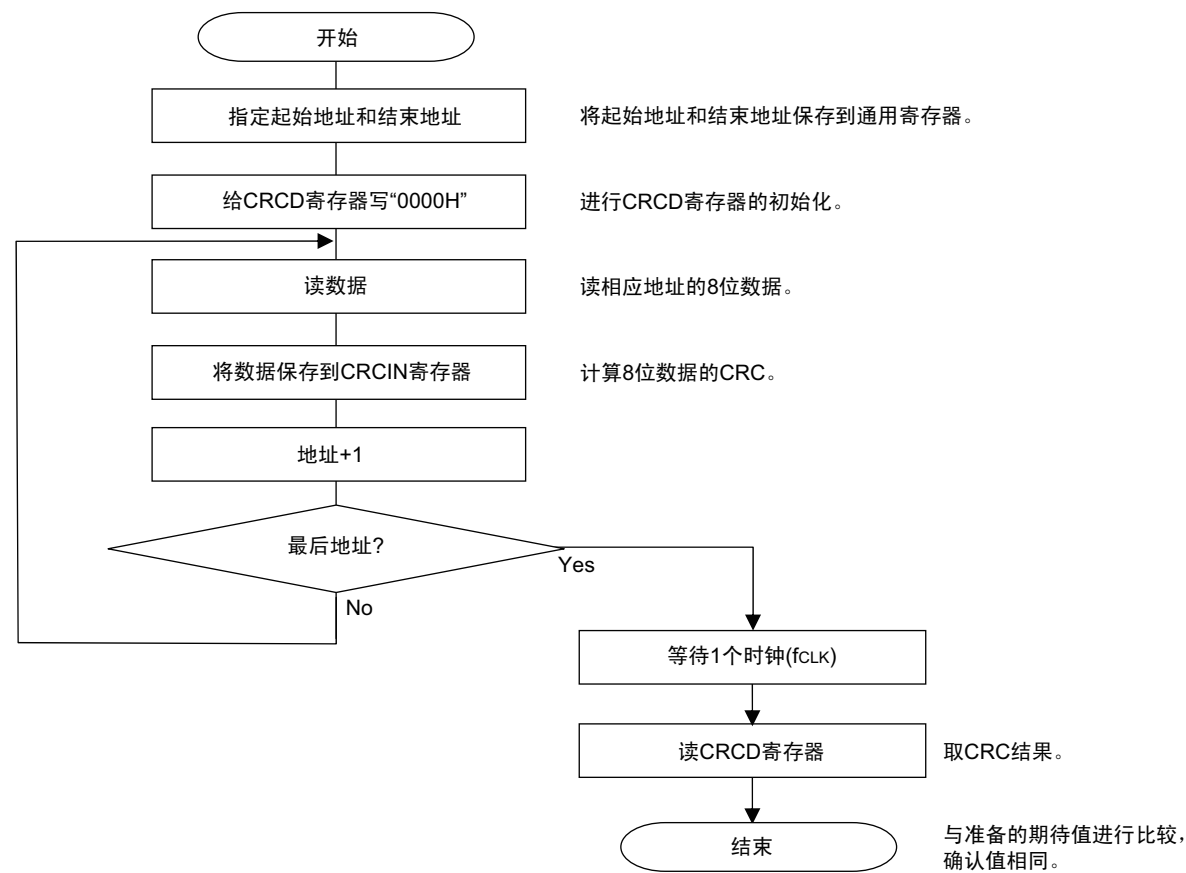
图 25-5 CRC 数据寄存器 (CRCD) 的格式



- 注意 1. 要读 CRCD 寄存器的写入值时, 必须在写 CRCIN 寄存器前读 CRCD 寄存器。
2. 如果 CRCD 寄存器的写操作与运算结果的保存发生竞争, 就忽视写操作。

<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

IEC60730 标准要求确认 RAM 数据。因此，R7F0C003、R7F0C004 的 RAM 每 8 位附加 1 位奇偶校验位。RAM 奇偶校验错误检测功能在写数据时附加奇偶校验位，而在读数据时检查奇偶校验位，并且能在发生奇偶校验错误时产生复位。

25.3.3.1 RAM 奇偶校验错误控制寄存器（RPECTL）

此寄存器控制奇偶校验的错误确认位和因奇偶校验错误而产生复位。
通过 1 位或者 8 位存储器操作指令设定 RPECTL 寄存器。
在产生复位信号后，此寄存器的值变为“00H”。

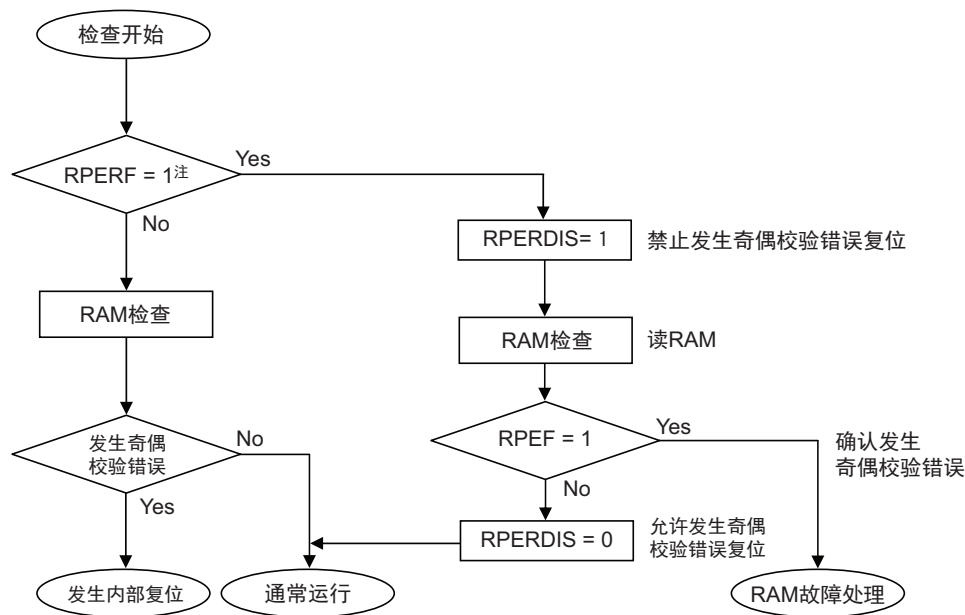
图 25-7 RAM 奇偶校验错误控制寄存器（RPECTL）的格式

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	奇偶校验错误复位的屏蔽标志							
	0	允许产生奇偶校验错误复位。							
	1	禁止产生奇偶校验错误复位。							
	RPEF	奇偶校验错误状态标志							
	0	没有发生奇偶校验错误。							
	1	发生奇偶校验错误。							

注意 在写数据时附加奇偶校验位，而在读数据时检查奇偶校验位。
因此，当允许产生 RAM 奇偶校验错误复位（RPERDIS=0）时，必须在存取数据时并且在读数据前对“所用 RAM 区”进行初始化。
因为 RL78 为流水线运行，所以 CPU 进行预读，有可能因读取所用 RAM 区前的未初始化 RAM 区而发生 RAM 奇偶校验错误。因此，要允许发生 RAM 奇偶校验错误复位（RPERDIS=0）时，必须在执行 RAM 区的指令时对“所用 RAM 区+10 字节”的区域进行初始化。

- 备注 1. 在初始状态下为允许发生奇偶校验错误复位（RPERDIS=0）。
2. 在设定为禁止发生奇偶校验错误复位（RPERDIS=1）时，如果发生奇偶校验错误，RPEF 标志就被置“1”。在 RPEF 标志为“1”的状态下，如果设定为允许发生奇偶校验错误复位（RPERDIS=0），就在 RPERDIS 清“0”时发生奇偶校验错误复位。
3. RPECTL 寄存器的 RPEF 标志在发生奇偶校验错误时被置“1”，通过写“0”或者全部的复位源清“0”。在 RPEF 标志为“1”时，即使读未发生奇偶校验错误的 RAM，RPEF 标志也保持“1”的状态。
4. RAM 奇偶校验错误检测的范围不包括通用寄存器。

图 25-8 RAM 奇偶校验检查的流程图



注 由 RAM 奇偶校验错误引起的内部复位的确认方法请参照“第 22 章 复位功能”。

25.3.4 RAM 保护功能

为了必须保证运行过程中的安全，IEC61508 标准要求即使 CPU 失控也需要保护存储在 RAM 的重要数据。此 RAM 保护功能用于保护被指定的 RAM 空间的数据。
如果设定为 RAM 保护功能，指定空间的 RAM 写操作就无效，但是能正常读取。

25.3.4.1 非法存储器存取检测控制寄存器（IAWCTL）

此寄存器控制是否允许检测非法存储器的存取以及控制 RAM/SFR 保护功能。
RAM 保护功能使用 GRAM1 位和 GRAM0 位。
通过 8 位存储器操作指令设定 IAWCTL 寄存器。
在产生复位信号后，此寄存器的值变为“00H”。

图 25-9 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。能写 RAM。
0	1	RAM 起始地址开始的 128 字节
1	0	RAM 起始地址开始的 256 字节
1	1	RAM 起始地址开始的 512 字节

注 RAM 的起始地址因产品内置的 RAM 容量而变。

25.3.5 SFR 保护功能

为了必须保证运行过程中的安全，IEC61508标准要求即使CPU失控也需要保护重要的SFR，使其免遭改写。
SFR 保护功能用于保护端口功能，中断功能，时钟控制功能，电压检测电路和 RAM 奇偶校验错误功能的数据。
如果设定为 SFR 保护功能，被保护的 SFR 的写操作就无效，但是能正常读取。

25.3.5.1 非法存储器存取检测控制寄存器（IAWCTL）

此寄存器控制是否允许检测非法存储器的存取以及控制 RAM/SFR 保护功能。
SFR 保护功能使用 GPORT 位、GINT 位和 GCSC 位。
通过 8 位存储器操作指令设定 IAWCTL 寄存器。
在产生复位信号后，此寄存器的值变为“00H”。

图 25-10 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

GPORT	端口功能的控制寄存器的保护
0	无效。能读写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操作无效，能读。 [被保护的SFR] PMxx、PUxx、PIMxx、POMxx、PMCxx、ADPC、PIOR、PFSEGxx、ISCLCD注

GINT	中断功能的寄存器的保护
0	无效。能读写中断功能的控制寄存器。
1	有效。中断功能的控制寄存器的写操作无效，能读。 [被保护的SFR] IFxx、MKxx、PRxx、EGPx、EGNx

GCSC	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。能读写时钟控制功能、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的写操作无效，能读。 [被保护的SFR] CMC、CSC、OSTS、CKC、PERx、OSMC、LVIM、LVIS、RPECTL

注 不保护 Pxx（端口寄存器）。

25.3.6 非法存储器存取检测功能

IEC60730 标准要求需要确认 CPU 和中断是否正常运行。
非法存储器存取检测功能在存取被规定的非法存取检测空间时产生复位。
非法存取检测空间为图 25-11 中记载为“NG”的范围。

图 25-11 非法存取检测空间

		能否存取		取指令 (执行)
		读	写	
FFFFFH	特殊功能寄存器 (SFR) 256 字节			NG
FFF00H FFEFFH	通用寄存器 32 字节		OK	
FFEE0H FFEDFH				OK
	RAM 注			
zzzzzH				
	镜像区	OK	NG	NG
	数据闪存			
F1000H F0FFFH				OK
F0800H F07FFH	保留区		OK	NG
	特殊功能寄存器 (2nd SFR) 2K 字节			
F0000H EFFFFH				OK
EF000H EEFFFFH				
	保留区	NG	NG	NG
yyyyyH				
xxxxxH				
	代码闪存 注	OK		OK
00000H				

注 各产品的代码闪存、RAM 地址和检测最低位地址如下所示：

产品	代码闪存 (00000H ~ xxxxxH)	RAM (zzzzH ~ FFEFFH)	读 / 取指令 (执行) 时的检测 最低位地址 (yyyyyH)
R7F0C003	98304×8 位 (00000H ~ 17FFFFH)	8192×8 位 (FDF00H ~ FFEFFH)	20000H
R7F0C004	131072×8 位 (00000H ~ 1FFFFH)		20000H

25.3.6.1 非法存储器存取检测控制寄存器（IAWCTL）

此寄存器控制是否允许检测非法存储器的存取以及控制 RAM/SFR 保护功能。
非法存储器存取检测功能使用 IAWEN 位。
通过 8 位存储器操作指令设定 IAWCTL 寄存器。
在产生复位信号后，此寄存器的值变为“00H”。

图 25-12 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

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 IAWEN 位只有写“1”的操作有效，IAWEN 位为“1”后的写“0”的操作无效。

备注 在选项字节（000C0H）的 WDTON 位为“1”（允许看门狗定时器运行）时，即使 IAWEN 位为“0”，非法存储器存取检测功能也有效。

25.3.7 频率检测功能

IEC60730 标准要求确认振荡频率是否正常。

频率检测功能可通过使用 CPU/ 外围硬件时钟频率 (f_{CLK}) 测量定时器阵列单元 0 (TAU0) 的通道 1 的输入脉冲, 判断 2 个时钟的比率关系是否正确。但是, 在其中一个或者两个时钟完全停止时, 就无法判断时钟的比率关系。

< 比较时钟 >

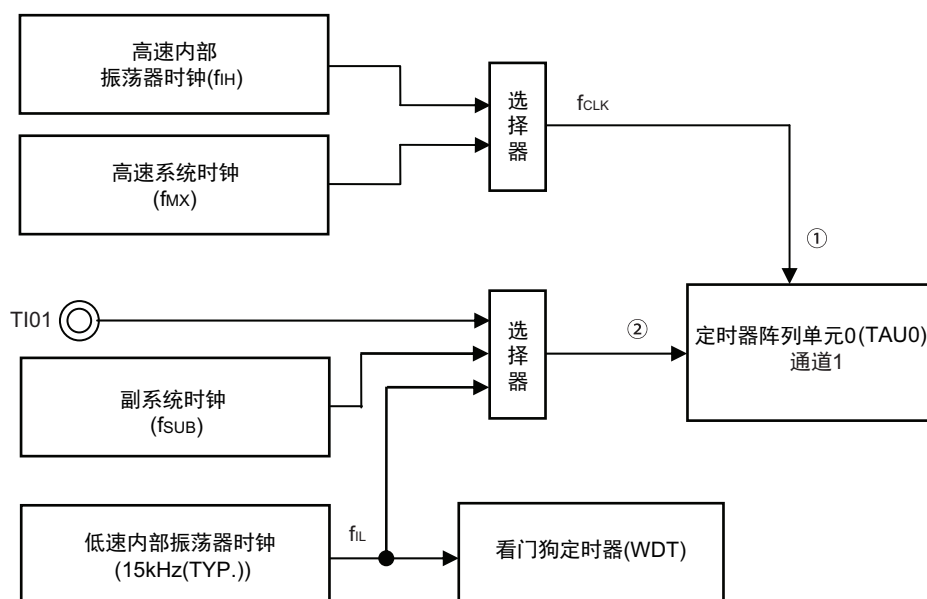
① CPU/ 外围硬件时钟频率 (f_{CLK}):

- 高速内部振荡器时钟 (f_{IH})
- 高速系统时钟 (f_{MX})

② 定时器阵列单元的通道 1 输入:

- 通道 1 的定时器输入 (TI01)
- 低速内部振荡器时钟 (f_{IL} : 15kHz(TYP.))
- 副系统时钟 (f_{SUB})

图 25-13 频率检测功能的结构



当输入脉冲间隔的测量结果为异常值时, 能判断为“时钟频率异常”。

有关输入脉冲间隔的测量方法, 请参照“6.8.3 作为输入脉冲间隔测量的运行”。

25.3.7.1 定时器输入选择寄存器 0（TIS0）

此寄存器选择定时器阵列单元 0（TAU0）的通道 1 的定时器输入。
通过 8 位存储器操作指令设定 TIS0 寄存器。
在产生复位信号后，此寄存器的值变为“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	通道 1 使用的定时器输入的选择
0	0	0	定时器输入引脚（TI01）的输入信号
0	0	1	
0	1	0	
0	1	1	
1	0	0	低速内部振荡器时钟（f _{IL} ）
1	0	1	副系统时钟（f _{SUB} ）
上述以外			禁止设定。

25.3.8 A/D 测试功能

IEC60730 标准要求进行 A/D 转换器的测试。此 A/D 测试功能通过对 A/D 转换器的正 (+) 基准电压、负 (-) 基准电压、模拟输入通道 (ANl)、温度传感器输出电压以及内部基准电压进行 A/D 转换来确认 A/D 转换器的正常运行。确认方法的详细内容请参照《RL78/G13 Safety Function (A/D Test)》Application Note (R01AN0955)。

能通过以上步骤确认模拟多路转换器：

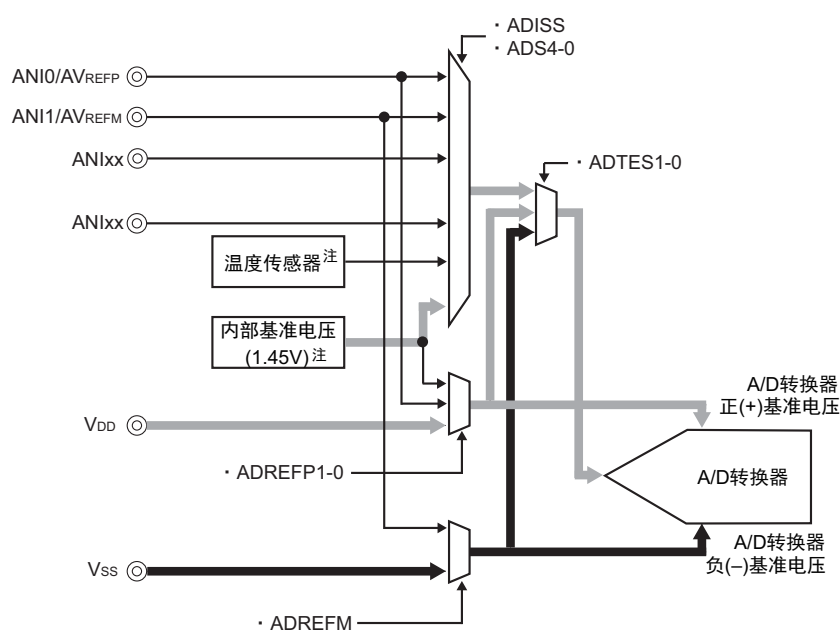
1. 通过 ADTES 寄存器选择 ANl_x 引脚作为 A/D 转换对象 (ADTES1、ADTES0=0、0)。
2. 对 ANl_x 引脚进行 A/D 转换 (转换结果 1-1)。
3. 通过 ADTES 寄存器选择 A/D 转换器的负 (-) 基准电压作为 A/D 转换对象 (ADTES1、ADTES0=1、0)。
4. 对 A/D 转换器的负 (-) 基准电压进行 A/D 转换 (转换结果 2-1)。
5. 通过 ADTES 寄存器选择 ANl_x 引脚作为 A/D 转换对象 (ADTES1、ADTES0=0、0)。
6. 对 ANl_x 引脚进行 A/D 转换 (转换结果 1-2)。
7. 通过 ADTES 寄存器选择 A/D 转换器的正 (+) 基准电压作为 A/D 转换对象 (ADTES1、ADTES0=1、1)。
8. 对 A/D 转换器的正 (+) 基准电压进行 A/D 转换 (转换结果 2-2)。
9. 通过 ADTES 寄存器选择 ANl_x 引脚作为 A/D 转换对象 (ADTES1、ADTES0=0、0)。
10. 对 ANl_x 引脚进行 A/D 转换 (转换结果 1-3)。
11. 确认“转换结果 1-1”=“转换结果 1-2”=“转换结果 1-3”。
12. 确认“转换结果 2-1”的 A/D 转换结果全部为“0”，“转换结果 2-2”的 A/D 转换结果全部为“1”。

通过以上步骤，能确认已选择模拟多路转换器以及布线没有断线。

备注 1. 在步骤 1～步骤 10 的转换运行过程中，如果模拟输入电压可变，就必须采用其他方法来确认模拟多路转换器。

2. 转换结果含有误差，因此必须在比较转换结果时要考虑到恰当的误差。

图 25-15 A/D 测试功能的结构



注 只有在 HS（高速主）模式中才能选择。

25.3.8.1 A/D 测试寄存器（ADTES）

此寄存器选择 A/D 转换器的正（+）基准电压、负（-）基准电压、模拟输入通道（ANIdx）、温度传感器输出电压以及内部基准电压（1.45V）作为 A/D 转换对象。

当用作 A/D 测试功能时，进行以下设定：

- 在测量零刻度时，选择负（-）基准电压作为 A/D 转换对象。
- 在测量满刻度时，选择正（+）基准电压作为 A/D 转换对象。

通过 8 位存储器操作指令设定 ADTES 寄存器。

在产生复位信号后，此寄存器的值变为“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 转换对象
0	0	ANIdx/ 温度传感器输出电压注 / 内部基准电压（1.45V）注 （由模拟输入通道指定寄存器（ADS）进行设定）
1	0	负（-）基准电压（通过 ADM2 寄存器的 ADREFM 位选择）
1	1	正（+）基准电压（通过 ADM2 寄存器的 ADREFP1、ADREFP0 位选择）注
上述以外		禁止设定。

注 只有在 HS（高速主）模式中才能选择温度传感器输出电压和内部基准电压（1.45V）。

25.3.8.2 模拟输入通道指定寄存器 (ADS)

此寄存器指定 A/D 转换的模拟电压的输入通道。

要通过 A/D 测试功能测量 ANIxx/ 温度传感器输出电压 / 内部基准电压 (1.45V) 时, 必须将 A/D 测试寄存器 (ADTES) 置 “00H”。

通过 1 位或者 8 位存储器操作指令设定 ADS 寄存器。

在产生复位信号后, 此寄存器的值变为 “00H”。

图 25-17 模拟输入通道指定寄存器 (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

ADISS	ADS4	ADS3	ADS2	ADS1	ADS0	模拟输入通道	输入源
0	0	0	0	0	0	ANI0	P21/ANI0/AV _{REFP} 引脚
0	0	0	0	0	1	ANI1	P20/ANI1/AV _{REFM} 引脚
0	1	0	0	0	0	ANI16	P22/ANI16 引脚
0	1	0	0	0	1	ANI17	P23/ANI17 引脚
1	0	0	0	0	0	—	温度传感器输出电压注
1	0	0	0	0	1	—	内部基准电压 (1.45V) 注
上述以外						禁止设定。	

注 只有在 HS (高速主) 模式中才能选择。

注意 1. 必须将 bit5 和 bit6 位置 “0”。

- 通过端口模式寄存器 2 (PM2) 将 ADPC、PMC 寄存器设定为模拟输入的端口设定为输入模式。
- 不能通过 ADS 寄存器对 A/D 端口配置寄存器 (ADPC) 设定的数字输入 / 输出引脚进行设定。
- 不能通过 ADS 寄存器对端口模式控制寄存器 2 (PMC2) 设定的数字输入 / 输出引脚进行设定。
- 要改写 ADISS 位时, 必须在转换停止的状态下 (ADCS=0、ADCE=0) 进行。
- 当将 AV_{REFP} 用作 A/D 转换器的正 (+) 基准电压时, 不能选择 ANI0 作为 A/D 转换通道。
- 当将 AV_{REFM} 用作 A/D 转换器的负 (-) 基准电压时, 不能选择 ANI1 作为 A/D 转换通道。
- 在将 ADISS 位置 “1” 的情况下, 不能将内部基准电压 (1.45V) 用作正 (+) 基准电压。

另外, 在将 ADISS 位置 “1” 后, 不能使用第 1 次的转换结果。详细设定流程请参照 “12.7.4 使用温度传感器时的设定 (以软件触发模式、单次转换模式为例)”。

- 要转移到 STOP 模式或者在以副系统时钟进行 CPU 运行时转移到 HALT 模式时, 不能将 ADISS 位置 “1”。如果将 ADISS 位置 “1”, 增加 “32.3.2 电源电流特性” 所示的 A/D 转换器基准电压电流 (I_{ADREF}) 的电流值。

25.3.9 输入 / 输出引脚的数字输出信号电平检测功能

IEC60730 标准要求确认 I/O 功能是否正确。
在引脚为输出模式时，可通过输入 / 输出引脚的数字输出信号电平检测功能读引脚的数字输出电平。

25.3.9.1 端口模式选择寄存器（PMS）

在引脚为输出模式（端口模式寄存器（PMm）的 PMmn 位为“0”）时，此寄存器选择读端口的输出锁存器的值或者读引脚的输出电平。

通过 1 位或者 8 位存储器操作指令设定 PMS 寄存器。
在产生复位信号后，此寄存器的值变为“00H”。

图 25-18 端口模式选择寄存器（PMS）的格式

地址: F0078H	复位后: 00H	R/W						
符号	7	6	5	4	3	2	1	0
PMS	0	0	0	0	0	0	0	PMS0

PMS0	选择端口为输出模式时（PMmn=0）的读取数据
0	读 Pmn 寄存器的值。
1	读引脚的数字输出电平。

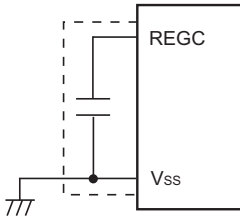
备注 m=0 ~ 7、12
 n=0 ~ 7

- 注意 1. 在 PMS 寄存器的 PMS0 位置“1”时，不能通过读修改指令改写 Px 寄存器。如图要改写 Px 寄存器，必须使用 8 位存储器操作指令。
2. PMS 寄存器控制对 LCD 专用引脚和输入专用引脚（P121 ~ P124、P137）无效。
3. 将 SEG 复用引脚用作 SEG 输出时，PMS 寄存器控制无效（读取时固定为低电平）。
4. 当 IICA0EN（PER0 寄存器的 bit4）=0 时，PMS 寄存器控制对 P60、P61 引脚无效。

第 26 章 稳压器

26.1 稳压器的概述

R7F0C003 和 R7F0C004 内置了使器件内部恒压工作的电路。为了使稳压器的输出电压稳定，必须通过电容器（0.47 ~ 1μF）将 REGC 引脚连接 V_{SS}。另外，为了稳定内部电压，必须使用特性好的电容器。



注意 必须尽量缩短上图虚线部分的布线。

稳压器的输出电压如表 26-1 所示。

表 26-1 稳压器输出电压条件

模式	输出电压	条件
LV（低电压主）模式	1.8V	—
LS（低速主）模式		
HS（高速主）模式	1.8V	STOP 模式
		在 CPU 以副系统时钟（f _{SUB} ）运行的过程中，高速系统时钟（f _{MX} ）和高速内部振荡器时钟（f _{IH} ）都停止振荡。
	2.1V	在设定为 CPU 以副系统时钟（f _{SUB} ）运行时的 HALT 模式中，高速系统时钟（f _{MX} ）和高速内部振荡器时钟（f _{IH} ）都停止振荡。
		上述以外（包含片上调试期间）注

注 在片上调试期间转移到副系统时钟运行模式或者 STOP 模式时，稳压器的输出电压保持 2.1V（不变为 1.8V）。

第 27 章 选项字节

27.1 选项字节的功能

R7F0C003 和 R7F0C004 的闪存地址 000C0H ~ 000C3H 为选项字节区。

选项字节由用户选项字节（000C0H ~ 000C2H）和片上调试选项字节（000C3H）构成。

在接通电源或者复位启动时，自动参照选项字节进行指定功能的设定。在使用本产品时，必须通过选项字节进行以下功能的设定。

对于没有配置功能的位，必须设定为初始值，不能更改。

要在自编程过程中使用引导交换功能时，因为 000C0H ~ 000C3H 被 010C0H ~ 010C3H 替换，所以必须给 010C0H ~ 010C3H 设定和 000C0H ~ 000C3H 相同的值。

注意 与是否使用各功能无关，必须设定选项字节。

27.1.1 用户选项字节（000C0H ~ 000C2H/010C0H ~ 010C2H）

(1) 000C0H/010C0H

- 看门狗定时器的运行
 - 禁止或者允许计数器的运行。
 - 在 HALT/STOP 模式中停止或者允许计数器的运行。
- 看门狗定时器的上溢时间的设定
- 看门狗定时器的窗口打开期间的设定
- 看门狗定时器的间隔中断的设定
 - 使用或者不使用间隔中断

注意 在引导交换时，000C0H 被 010C0H 替换，因此必须给 010C0H 设定和 000C0H 相同的值。

(2) 000C1H/010C1H

- LVD 运行模式的设定
 - 中断&复位模式
 - 复位模式
 - 中断模式
 - LVD OFF（使用 $\overline{\text{RESET}}$ 引脚引起的外部复位）
- LVD 检测电平（ V_{LVDH} 、 V_{LVDL} 、 V_{LVD} ）的设定

注意 1. 当电源上升时，在达到“32.4 AC 特性”所示的工作电压范围前，必须通过电压检测电路或者外部复位保持复位状态。当电源下降时，必须在小于工作电压范围前，转移到 STOP 模式，或者通过电压检测电路或外部复位设定为复位状态。

工作电压范围因用户选项字节（000C2H/010C2H）的设定而变。

2. 在引导交换时，000C1H 被 010C1H 替换，因此必须给 010C1H 设定和 000C1H 相同的值。

(3) 000C2H/010C2H

- 闪存运行模式的设定
 - LV（低电压主）模式
 - LS（低速主）模式
 - HS（高速主）模式
- 高速内部振荡器的频率设定
 - 从24MHz/16MHz/12MHz/8MHz/6MHz/4MHz/3MHz/2MHz/1MHz(TYP.)中选择。

注意 在引导交换时，000C2H 被 010C2H 替换，因此必须给 010C2H 设定和 000C2H 相同的值。

27.1.2 片上调试选项字节（000C3H/010C3H）

- 片上调试运行的控制
 - 禁止或者允许片上调试运行
- 安全ID验证失败时闪存数据的处理
 - 是否在片上调试安全ID验证失败时擦除闪存的数据。

注意 在引导交换时，000C3H 被 010C3H 替换，因此必须给 010C3H 设定和 000C3H 相同的值。

27.2 用户选项字节的格式

图 27-1 用户选项字节（000C0H/010C0H）的格式

地址：000C0H/010C0H 注 1

7	6	5	4	3	2	1	0
WDTINT	WINDOW1	WINDOW0	WDTON	WDCS2	WDCS1	WDCS0	WDSTBYON

WDTINT	看门狗定时器的间隔中断的使用 / 不使用
0	不使用间隔中断。
1	当达到上溢时间的 75%+1/2f _{IL} 时，产生间隔中断。

WINDOW1	WINDOW0	看门狗定时器的窗口打开期间注 2
0	0	禁止设定。
0	1	50%
1	0	75%
1	1	100%

WDTON	看门狗定时器的计数器运行控制
0	禁止计数器运行（解除复位后停止计数）。
1	允许计数器运行（解除复位后开始计数）。

WDCS2	WDCS1	WDCS0	看门狗定时器的上溢时间 (f _{IL} =17.25kHz(MAX.))
0	0	0	2 ⁶ /f _{IL} (3.71ms)
0	0	1	2 ⁷ /f _{IL} (7.42ms)
0	1	0	2 ⁸ /f _{IL} (14.84ms)
0	1	1	2 ⁹ /f _{IL} (29.68ms)
1	0	0	2 ¹¹ /f _{IL} (118.72ms)
1	0	1	2 ¹³ /f _{IL} (474.90ms)
1	1	0	2 ¹⁴ /f _{IL} (949.80ms)
1	1	1	2 ¹⁶ /f _{IL} (3799.19ms)

WDSTBYON	看门狗定时器的计数器运行控制（HALT/STOP 模式）
0	在 HALT/STOP 模式中，停止计数器运行注 2。
1	在 HALT/STOP 模式中，允许计数器运行。

注 1. 在引导交换时，000C0H 被 010C0H 替换，因此必须给 010C0H 设定和 000C0H 相同的值。

2. 当 WDSTBYON 位为“0”时，与 WINDOW1 位和 WINDOW0 位的值无关，窗口打开期间为 100%。

备注 f_{IL}：低速内部振荡器时钟频率

图 27-2 用户选项字节（000C1H/010C1H）的格式（1/2）

地址：000C1H/010C1H 注

7	6	5	4	3	2	1	0
VPOC2	VPOC1	VPOC0	1	LVIS1	LVIS0	LVIMDS1	LVIMDS0

- LVD 设定（中断&复位模式）

检测电压			选项字节设定值						
V _{LVDH}		V _{LVDL}	模式设定		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0
上升沿	下降沿	下降沿	LVIMDS1	LVIMDS0					
1.77 V	1.73 V	1.63 V	1	0	0	0	0	1	0
1.88 V	1.84 V							0	1
2.92 V	2.86 V							0	0
1.98 V	1.94 V	1.84 V				0	1	1	0
2.09 V	2.04 V							0	1
3.13 V	3.06 V							0	0
2.61 V	2.55 V	2.45 V			1		0	1	0
2.71 V	2.65 V							0	1
3.75 V	3.67 V							0	0
2.92 V	2.86 V	2.75 V			1	1	1	0	0
3.02 V	2.96 V							1	1
4.06 V	3.98 V							0	0
上述以外		禁止设定							

- LVD 设定（复位模式）

检测电压		选项字节设定值						
V _{LVD}		模式设定		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0
上升沿	下降沿	LVIMDS1	LVIMDS0					
1.67 V	1.63 V	1	1	0	0	0	1	1
1.77 V	1.73 V				0	0	1	0
1.88 V	1.84 V				0	1	1	1
1.98 V	1.94 V				0	1	1	0
2.09 V	2.04 V				0	1	0	1
2.50 V	2.45 V				1	0	1	1
2.61 V	2.55 V				1	0	1	0
2.71 V	2.65 V				1	0	0	1
2.81 V	2.75 V				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
上述以外		禁止设定						

注 在引导交换时，000C1H 被 010C1H 替换，因此必须给 010C1H 设定和 000C1H 相同的值。

备注 1. 有关 LVD 电路的详细内容，请参照“第 24 章 电压检测电路”。

2. 检测电压为 TYP. 值。详细内容请参照“32.6.5 LVD 电路特性”。

图 27-2 用户选项字节（000C1H/010C1H）的格式（2/2）

地址：000C1H/010C1H 注

7	6	5	4	3	2	1	0
VPOC2	VPOC1	VPOC0	1	LVIS1	LVIS0	LVIMDS1	LVIMDS0

- LVD 设定（中断模式）

检测电压		选项字节设定值						
V _{LVD}		模式设定		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0
上升沿	下降沿	LVIMDS1	LVIMDS0					
1.67 V	1.63 V	0	1	0	0	0	1	1
1.77 V	1.73 V				0	0	1	0
1.88 V	1.84 V				0	1	1	1
1.98 V	1.94 V				0	1	1	0
2.09 V	2.04 V				0	1	0	1
2.50 V	2.45 V				1	0	1	1
2.61 V	2.55 V				1	0	1	0
2.71 V	2.65 V				1	0	0	1
2.81 V	2.75 V				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
上述以外		禁止设定						

- LVD 设定（LVD OFF）

检测电压		选项字节设定值						
V_{LVD}		模式设定		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0
上升沿	下降沿	LVIMDS1	LVIMDS0					
—	—	×	1	1	×	×	×	×
上述以外		禁止设定						

注 在引导交换时，000C1H 被 010C1H 替换，因此必须给 010C1H 设定和 000C1H 相同的值。

注意 1. 必须给 bit4 写“1”。

- 当电源上升时，在达到“32.4 AC 特性”所示的工作电压范围前，必须通过电压检测电路或者外部复位保持复位状态。当电源下降时，必须在小于工作电压范围前，转移到 STOP 模式，或者通过电压检测电路或外部复位设定为复位状态。

工作电压范围因用户选项字节（000C2H/010C2H）的设定而变。

备注 1. ×：忽略

- 有关 LVD 电路的详细内容，请参照“第 24 章 电压检测电路”。
- 检测电压为 TYP. 值。详细内容请参照“32.6.5 LVD 电路特性”。

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	闪存运行模式的设定		
			工作频率范围	工作电压范围
0	0	LV（低电压主）模式	1 ～ 4MHz	1.6 ～ 5.5V
1	0	LS（低速主）模式	1 ～ 8MHz	1.8 ～ 5.5V
1	1	HS（高速主）模式	1 ～ 16MHz	2.4 ～ 5.5V
			1 ～ 24MHz	2.7 ～ 5.5V
上述以外		禁止设定		

FRQSEL3	FRQSEL2	FRQSEL1	FRQSEL0	高速内部振荡器时钟的频率（f _{IH} ）
0	0	0	0	24MHz
1	0	0	1	16MHz
0	0	0	1	12MHz
1	0	1	0	8MHz
0	0	1	0	6MHz
1	0	1	1	4MHz
0	0	1	1	3MHz
1	1	0	0	2MHz
1	1	0	1	1MHz
上述以外				禁止设定

注 在引导交换时，000C2H 被 010C2H 替换，因此必须给 010C2H 设定和 000C2H 相同的值。

注意 1. 必须给 bit5 写“1”，给 bit4 写“0”。

2. 工作频率范围和工作电压范围因闪存的各运行模式而不同。详细内容请参照“32.4 AC 特性”。

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	片上调试运行的控制
0	0	禁止片上调试运行。
0	1	禁止设定。
1	0	允许片上调试运行。 在片上调试安全 ID 验证失败时，擦除闪存的数据。
1	1	允许片上调试运行。 在片上调试安全 ID 验证失败时，不擦除闪存的数据。

注 在引导交换时，000C3H 被 010C3H 替换，因此必须给 010C3H 设定和 000C3H 相同的值。

注意 只有 bit7 和 bit0（OCDENSET 和 OCDERSD）才能指定值。
必须给 bit6～1 写“000010B”。

备注 bit3～1 的值在使用片上调试功能时被改写，因此在设定后变为不定值。
但是，在设定时必须给 bit3～1 设定初始值（0、1、0）。

27.4 选项字节的设定

用户选项字节和片上调试选项字节除了在源程序中记述以外，还能通过链接程序选项进行设定。此时，即使在如下的源程序中有记述也优先链接程序选项的设定。

选项字节设定的软件记述例子如下所示。

OPT	CSEG	OPT_BYTE	
	DB	36H	； 不使用看门狗定时器的间隔中断。 ； 允许看门狗定时器运行。 ； 看门狗定时器的窗口打开期间为 50%。 ； 看门狗定时器的上溢时间为 $2^9/f_{IL}$ 。 ； 在 HALT/STOP 模式中，停止看门狗定时器的运行。
	DB	1AH	； V_{LVDL} 选择 1.63V。 ； V_{LVDH} 选择上升 1.77V、下降 1.73V。 ； 选择中断 & 复位模式作为 LVD 运行模式。
	DB	2DH	； 选择 LV（低电压主）模式作为闪存运行模式。 ； 选择高速内部振荡器的时钟频率 1MHz。
	DB	85H	； 允许片上调试运行，在安全 ID 验证失败时不擦除闪存的数据。

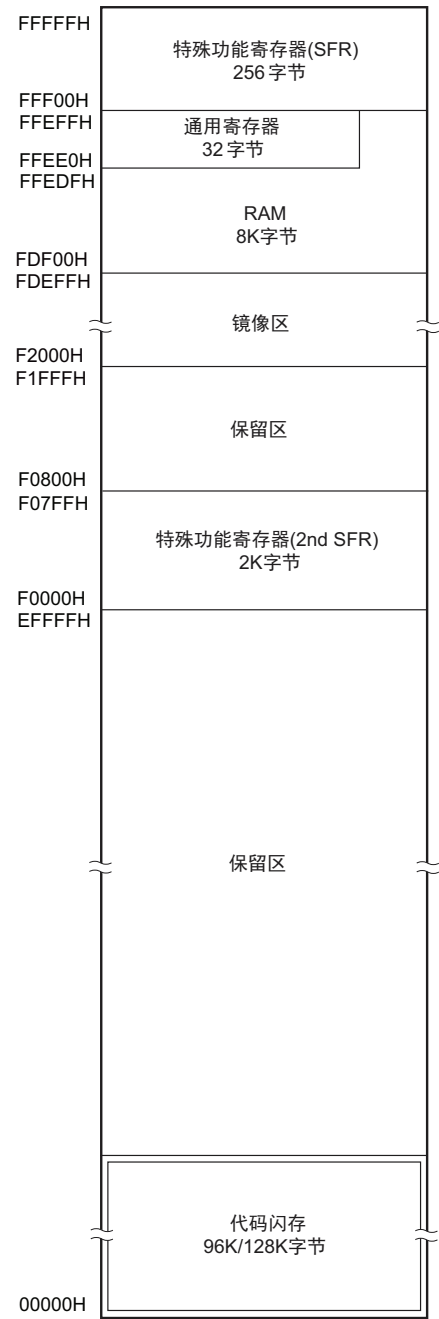
要在自编程过程中使用引导交换功能时，因为 000C0H ~ 000C3H 被 010C0H ~ 010C3H 替换，所以必须按照以下方式给 010C0H ~ 010C3H 设定和 000C0H ~ 000C3H 相同的值。

OPT2	CSEG	AT	010C0H	
	DB		36H	； 不使用看门狗定时器的间隔中断。 ； 允许看门狗定时器运行。 ； 看门狗定时器的窗口打开期间为 50%。 ； 看门狗定时器的上溢时间为 $2^9/f_{IL}$ 。 ； 在 HALT/STOP 模式中，停止看门狗定时器的运行。
	DB		1AH	； V_{LVDL} 选择 1.63V。 ； V_{LVDH} 选择上升 1.77V、下降 1.73V。 ； 选择中断 & 复位模式作为 LVD 运行模式。
	DB		2DH	； 选择 LV（低电压主）模式作为闪存运行模式。 ； 选择高速内部振荡器的时钟频率 1MHz。
	DB		85H	； 允许片上调试运行，在安全 ID 验证失败时不擦除闪存的数据。

注意 当通过汇编语言指定选项字节时，CSEG 伪指令的重定位属性名必须使用 OPT_BYTE。为了使用引导交换功能而给 010C0H ~ 010C3H 指定选项字节时，必须使用重定位属性 AT 来指定绝对地址。

第 28 章 闪存

RL78 微控制器内置了可进行编程、擦除和重写的闪存。闪存有可执行程序的“代码闪存”和数据保存区的“数据闪存”。



闪存的编程方法有以下几种。

能通过闪存编程器或者外部器件（UART 通信）进行的串行编程或者通过自编程改写代码闪存。

- 使用闪存编程器进行的串行编程（参照 28.1）
能使用专用闪存编程器进行板上或者板外编程。
- 使用外部器件（内置 UART）进行的串行编程（参照 28.2）
能使用外部器件（单片机和 ASIC）的 UART 通信进行板上编程。
- 自编程（参照 28.5）
能使用闪存自编程库在用户应用上自改写代码闪存。

28.1 通过闪存编程器进行的串行编程

能使用以下专用的闪存编程器对 RL78 微控制器的内部闪存进行数据编程。

- PG-FP5、FL-PR5
- E1 片上调试仿真器

能使用专用闪存编程器进行板上或者板外编程。

(1) 板上编程

在将 RL78 微控制器安装到目标系统后改写闪存的内容。必须在目标系统上安装连接专用闪存编程器的连接器。

(2) 板外编程

能在将 RL78 微控制器安装到目标系统前使用专用编程适配器（FA 系列）等进行闪存编程。

备注 FL-PR5 和 FA 系列是 Naito Densei Machida Mfg. Co., Ltd 的产品。

表 28-1 R78 微控制器和专用闪存编程器的连线表

专用闪存编程器的连接引脚				引脚名	引脚号
信号名		输入 / 输出	引脚功能		LFQFP (12×12)
PG-FP5、FL-PR5	E1 片上调试仿真器				
—	TOOL0	输入 / 输出	发送 / 接收信号	TOOL0/P40	9
SI/RxD	—	输入 / 输出	发送 / 接收信号		
—	RESET	输出	复位信号	RESET	10
/RESET	—	输出			
V _{DD}		输入 / 输出	V _{DD} 电压生成 / 电源监视	V _{DD}	18
GND		—	接地	V _{SS}	17
				EV _{SS}	—
				REGC 注	16
EMV _{DD}		—	TOOL0 引脚驱动电源	V _{DD}	18

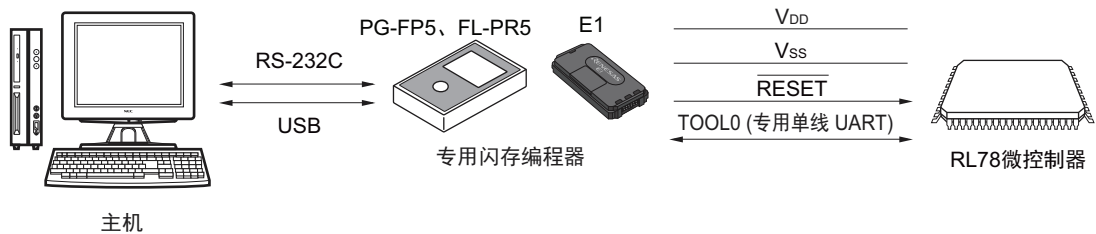
注 必须通过电容器（0.47 ~ 1μF）将 REGC 引脚接地。

备注 在使用闪存编程器进行编程时，此表中未记载的引脚可以置为开路。

28.1.1 编程环境

RL78 微控制器的闪存的编程环境如下所示。

图 28-1 闪存的编程环境



需要控制专用闪存编程器的主机。

专用闪存编程器和 RL78 微控制器的接口使用 TOOL0 引脚，通过专用的单线 UART 进行编程和擦除。

28.1.2 通信方式

专用闪存编程器和 RL78 微控制器的通信使用 RL78 微控制器的 TOOL0 引脚，通过专用的单线 UART 进行串行通信。

传送速率：1M、500k、250k、115.2kbps

图 28-2 和专用闪存编程器的通信



- 注 1. 这是使用 E1 片上调试仿真器的情况。
- 注 2. 这是使用 PG-FP5 或者 FL-PR5 的情况。
- 注 3. 必须通过电容器（0.47 ~ 1μF）将 REGC 引脚接地。

专用闪存编程器对 RL78 微控制器生成以下信号。详细内容请参照 PG-FP5、FL-PR5 或者 E1 片上调试仿真器的各手册。

表 28-2 引脚连接一览表

专用闪存编程器				RL78 微控制器
信号名		输入 / 输出	引脚功能	引脚名
PG-FP5、 FL-PR5	E1 片上调试仿真器			
V _{DD}		输入 / 输出	V _{DD} 电压生成 / 电压监视	V _{DD}
GND		—	接地	V _{SS} 、REGC 注
EMV _{DD}		—	TOOL0 引脚驱动电源	V _{DD}
/RESET	—	输出	复位信号	$\overline{\text{RESET}}$
—	$\overline{\text{RESET}}$	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	

注 必须通过电容器（0.47 ~ 1μF）将 REGC 引脚接地。

28.2 通过外部器件（内置 UART）进行的串行编程

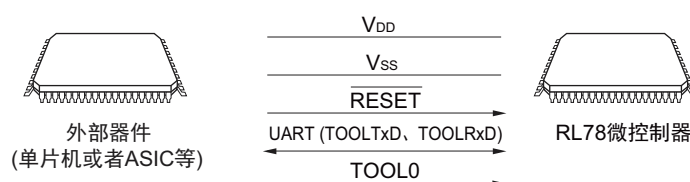
能使用 RL78 微控制器和与 UART 连接的外部器件（单片机或者 ASIC）对内部闪存进行数据的板上编程。

有关用户闪存编程器的开发，请参照《RL78 Microcontrollers (RL78 Protocol A) Programmer Edition》Application Note（R01AN0815）。

28.2.1 编程环境

RL78 微控制器的闪存的编程环境如下所示。

图 28-3 闪存的编程环境



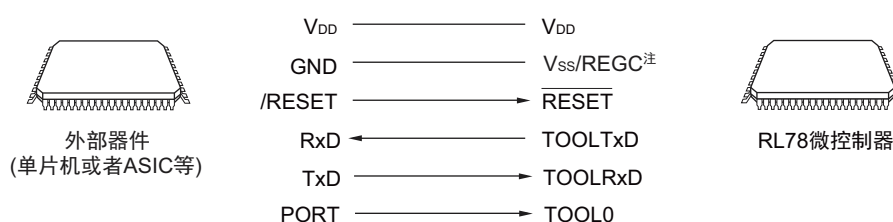
外部器件对 RL78 微控制器进行板上编程和擦除，但是不能进行板外编程。

28.2.2 通信方式

外部器件和 RL78 微控制器的通信使用 RL78 微控制器的 TOOLTxD 引脚和 TOOLRxD 引脚，通过专用的 UART 进行串行通信。

传送速率：1M、500k、250k、115.2kbps

图 28-4 和外部器件的通信



注 必须通过电容器（0.47 ~ 1μF）将 REGC 引脚接地。

外部器件对 RL78 微控制器生成以下信号。

表 28-3 引脚连接一览表

外部器件			RL78 微控制器
信号名	输入 / 输出	引脚功能	引脚名
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
RESETOUT	输出	复位信号输出	RESET
RxD	输入	接收信号	TOOLTxD
TxD	输出	发送信号	TOOLRxD
PORT	输出	模式信号	TOOL0

注 必须通过电容器（0.47 ~ 1μF）将 REGC 引脚接地。

28.3 电路板上的引脚处理

要通过闪存编程器进行板上编程时，必须在目标系统上设置连接专用闪存编程器的连接器，并且在电路板上设置从通常运行模式到闪存编程模式的切换功能。

如果转移到闪存编程模式，闪存编程时未使用的全部引脚就保持刚复位后的相同状态。因此，如果外部器件不认可刚复位后的状态，就需要进行引脚处理。

备注 有关闪存编程模式，请参照“28.4.2 闪存编程模式”。

28.3.1 P40/TOOL0 引脚

在闪存编程模式中，必须在外部通过 1kΩ 电阻将此引脚上拉并且连接到专用的闪存编程器。

当此引脚用作端口引脚时，必须按照以下方法使用此引脚。

用作输入引脚：在解除外部复位时的 t_{HD} 期间，不能输入低电平。另外，在通过下拉使用此引脚时，使用的电阻必须至少为 500kΩ。

用作输出引脚：在通过下拉使用此引脚时，使用的电阻必须至少为 500kΩ。

备注 1. t_{HD} ：在转移到闪存编程模式时，从解除外部 / 内部复位后 TOOL0 引脚保持低电平的时间。参照“32.11 闪存编程模式的转移时序”。

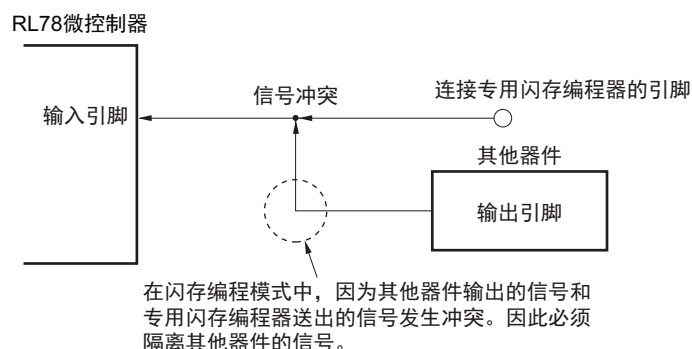
2. 因为 RL78 微控制器和专用闪存编程器的通信使用单线 UART（TOOL0 引脚），所以不使用 SAU 引脚和 IICA 引脚。

28.3.2 $\overline{\text{RESET}}$ 引脚

如果在电路板上将专用闪存编程器和外部器件的复位信号连接到与复位信号生成电路相连的 $\overline{\text{RESET}}$ 引脚，就会发生信号冲突。为了避免此信号冲突，必须隔离与复位信号生成电路的连接。

在闪存编程模式的期间，如果从用户系统输入复位信号，就不能进行正常的编程，因此除了专用闪存编程器和外部器件的复位信号以外，不能输入其他复位信号。

图 28-5 信号冲突 ($\overline{\text{RESET}}$ 引脚)



28.3.3 端口引脚

如果转移到闪存编程模式，闪存编程时未使用的全部引脚就保持刚复位后的相同状态。因此，如果连接各端口的外部器件不允许刚复位后的端口状态，就需要通过电阻将引脚连接 V_{DD} ，或者通过电阻将引脚连接 V_{SS} 进行引脚处理。

28.3.4 REGC 引脚

和通常运行模式相同，必须通过特性良好的电容器 ($0.47 \sim 1\mu\text{F}$) 将 REGC 引脚连接 GND。另外，为了稳定内部电压，必须使用特性好的电容器。

28.3.5 X1 引脚和 X2 引脚

X1、X2 的连接方法必须和通常运行模式相同。

备注 在闪存编程模式中，使用高速内部振荡器时钟 (f_{IH})。

28.3.6 电源

当使用闪存编程器输出的电源时，必须将 V_{DD} 引脚连接闪存编程器的 V_{DD} ，并且将 V_{SS} 引脚连接闪存编程器的 GND。

当使用电路板上的电源时，必须按照通常运行模式进行连接。

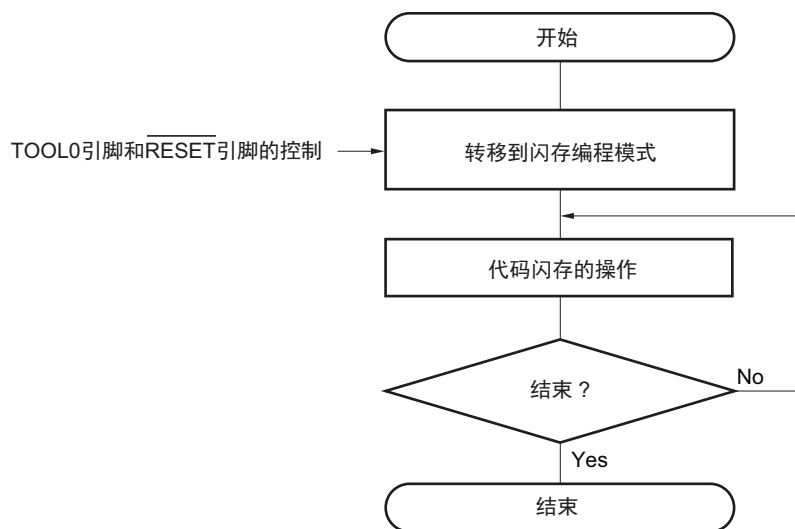
但是，在通过闪存编程器进行编程时，即使使用电路板上的电源电压，也必须将 V_{DD} 引脚和 V_{SS} 引脚分别连接闪存编程器的 V_{DD} 和 GND，以便通过闪存编程器监视电压。

28.4 串行编程方法

28.4.1 串行编程步骤

通过串行编程执行代码闪存编程的步骤如图 28-6 所示。

图 28-6 代码闪存的操作步骤



28.4.2 闪存编程模式

通过串行编程改写代码闪存的内容时，必须设定为闪存编程模式。要转移到闪存编程模式时，必须按照以下方法进行。

＜使用专用闪存编程器进行串行编程的情况＞

将 RL78 微控制器与专用闪存编程器连接。通过与专用闪存编程器的通信，自动转移到闪存编程模式。

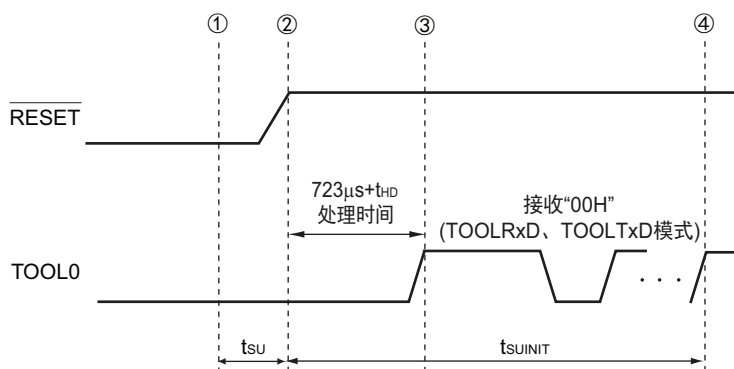
＜使用外部器件（UART 通信）进行串行编程的情况＞

在将 TOOL0 引脚置为低电平后解除复位（参照表 28-4）。此后，按照图 28-7 所示的步骤①～④转移到闪存编程模式。详细内容请参照《RL78 Microcontrollers (RL78 Protocol A) Programmer Edition》Application Note (R01AN0815)。

表 28-4 复位解除时的 TOOL0 引脚与运行模式的关系

TOOL0	运行模式
V_{DD}	通常运行模式
0V	闪存编程模式

图 28-7 向闪存编程模式的转移



- ① 给TOOL0引脚输入低电位。
- ② 解除外部复位（在此之前需要解除POR和LVD复位）。
- ③ 解除TOOL0引脚的低电平。
- ④ 通过UART接收结束波特率的设定。

备注 t_{SUNIT} : 在此区间，必须在解除复位后的 100ms 之内完成初始设定的通信。

t_{SU} : 这是从将TOOL0引脚置低电平到解除外部复位为止的时间。

t_{HD} : 这是从解除外部/内部复位后TOOL0引脚保持低电平的时间（闪存固件处理时间除外）。

详细内容请参照“32.11 闪存编程模式的转移时序”。

闪存编程模式包括 2 个模式，分别是宽电压模式和全速模式。模式的选择取决于编程时向单片机提供的电源电压值和转移到闪存编程模式时的用户选项字节的设定信息。

另外，在使用专用闪存编程器进行串行编程时，能通过在 GUI 上设定电压自动选择模式。

表 28-5 编程模式和可进行编程、擦除或者校验的电压

电源电压 (V_{DD})	转移到闪存编程模式时的选项字节的设定		闪存编程模式
	闪存运行模式	工作频率	
$2.7V \leq V_{DD} \leq 5.5V$	空白状态		全速模式
	HS（高速主）模式	1MHz ~ 24MHz	全速模式
	LS（低速主）模式	1MHz ~ 8MHz	宽电压模式
	LV（低电压主）模式	1MHz ~ 4MHz	宽电压模式
$2.4V \leq V_{DD} < 2.7V$	空白状态		全速模式
	HS（高速主）模式	1MHz ~ 16MHz	全速模式
	LS（低速主）模式	1MHz ~ 8MHz	宽电压模式
	LV（低电压主）模式	1MHz ~ 4MHz	宽电压模式
$1.8V \leq V_{DD} < 2.4V$	空白状态		宽电压模式
	LS（低速主）模式	1MHz ~ 8MHz	宽电压模式
	LV（低电压主）模式	1MHz ~ 4MHz	宽电压模式

备注 1. 在宽电压模式和全速模式并用的情况下进行编程、擦除或者校验时，没有任何限制事项。

2. 有关通信命令的详细内容，请参照“28.4.4 通信命令”。

28.4.3 通信模式

RL78 微控制器的通信模式如下所示。

表 28-6 通信模式

通信方式	标准设定注 1				使用的引脚
	端口	速率注 2	频率	乘率	
单线 UART (使用闪存编程器或者外部器件时)	UART	115200bps、 250000bps、 500000bps、 1Mbps	—	—	TOOL0
专用 UART (使用外部器件时)	UART	115200bps、 250000bps、 500000bps、 1Mbps	—	—	TOOLTxD、 TOOLRxD

注 1. 这是与闪存编程器 GUI 标准设定相关的设定项目。

2. 除了波特率误差以外，信号波形的失真等也会影响 UART 通信，因此必须在充分评估后使用。

28.4.4 通信命令

RL78 微控制器通过表 28-7 所示的命令执行串行编程。

从专用闪存编程器或者外部器件发送到 RL78 微控制器的信号称为“命令”，并且执行与该命令对应的各功能处理。详细内容请参照《RL78 Microcontrollers (RL78 Protocol A) Programmer Edition》Application Note (R01AN0815)。

表 28-7 闪存控制命令

分类	命令名称	功能
校验	Verify	将指定闪存区的内容和从编程器送来的数据进行比较。
擦除	Block Erase	擦除指定的闪存区。
空白检查	Block Blank Check	检查指定块的闪存擦除状态。
编程	Programming	将数据写到指定的闪存区注。
信息取得	Silicon Signature	取得 RL78 微控制器的信息（例如：产品名、闪存结构、用于编程的固件版本等）。
	Checksum	取得指定区域的校验和。
安全	Security Set	设定安全信息。
	Security Get	取得安全信息。
	Security Release	解除禁止编程的设定。
其他	Reset	用于通信的同步检测。
	Baud Rate Set	设定选择 UART 时的波特率。

注 必须确认在编程区域中没有写入数据。因为在设定为禁止块擦除后就无法进行擦除，所以在没有擦除数据时，不能写数据。

如果执行“Silicon Signature”命令，就能取得产品信息（产品名、固件版本等）。

识别标志数据一览表和识别标志数据的例子分别如表 28-8 和表 28-9 所示。

表 28-8 识别标志数据一览表

字段名	内容	发送字节数
器件代码	分配给器件的序列号	3 字节
器件名	器件名（ASCII 码）	10 字节
闪存区的结束地址	闪存区的结束地址 （从地址的低位开始发送。例：00000H ~ 0FFFFH（64KB） →FFH、FFH、00H）	3 字节
固件版本	用于编程的固件版本信息 （从版本的高位开始发送。例：Ver.1.23→01H、02H、03H）	3 字节

表 28-9 识别标志数据的例子

字段名	内容	发送字节数	数据（十六进制）
器件代码	RL78 协议 A	3 字节	10 00 06
器件名	R7F0C004	10 字节	52 = “R” 37 = “7” 46 = “F” 30 = “0” 43 = “C” 30 = “0” 30 = “0” 34 = “4” 4D = “M” 20 = “ ”
代码闪存区的结束地址	代码闪存区 00000H ~ 1FFFFH（128KB）	3 字节	FF FF 01
固件版本	Ver.1.23	3 字节	01 02 03

28.5 自编程

RL78 微控制器支持自编程功能，能通过用户程序改写代码闪存。由于此功能可让用户应用程序利用闪存自编程库来改写代码闪存，因此能在现场进行程序升级等。

注意 1. 当 CPU 以副系统时钟运行时，不能使用自编程功能。

2. 为了在自编程过程中禁止中断，必须和通常运行模式中一样，在通过 DI 指令将 IE 标志清“0”的状态下执行闪存自编程库。
在允许中断的情况下，必须在通过 EI 指令将 IE 标志置“1”的状态下将接受中断的中断屏蔽标志清“0”，然后执行闪存自编程库。
3. 在自编程过程中，需要使高速内部振荡器振荡。当高速内部振荡器处于停止状态时，必须使高速内部振荡器时钟运行（HIOSTOP=0），并且在用户选项字节（000C2H）的 FRQSEL4 为“0”时经过 30 μ s 或者在 FRQSEL4 为“1”时经过 80 μ s 后执行闪存自编程库。

备注 1. 有关自编程功能的详细内容，请参照《RL78 Family Flash Self Programming Library Type01》User's Manual (R01US0050)。

2. 有关自编程的执行处理时间，请参照闪存自编程库工具附属的使用时的注意点。

另外，自编程功能包括 2 种模式，根据闪存编程模式分为宽电压模式和全速模式。

根据选项字节 000C2H 的 CMODE1 和 CMODE0 设定的闪存运行模式，设定其中一种模式。

在设定为 HS（高速主）模式时，必须设定为全速模式；在设定为 LS（低速主）模式和 LV（低电压主）模式时，必须设定为宽电压模式。

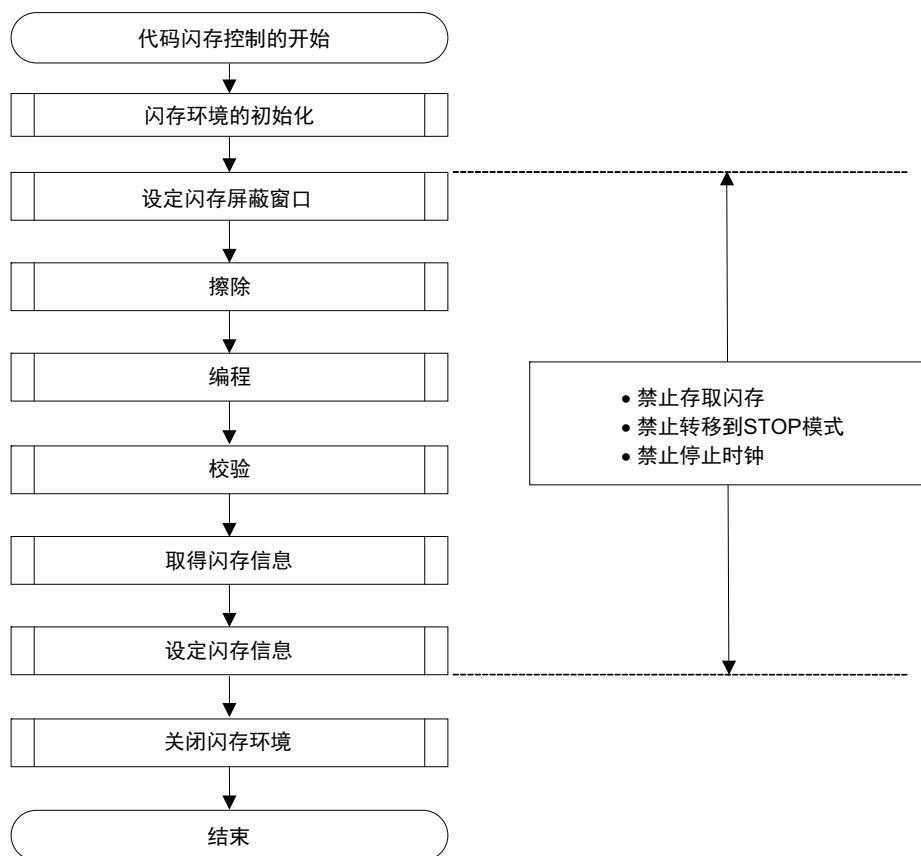
在执行本公司提供的闪存自编程库的“FSL_Init”函数时，如果参数“fsl_flash_voltage_u08”为“00H”，就设定为全速模式，不为“00H”，就设定为宽电压模式。

备注 在宽电压模式和全速模式并用的情况下进行编程、删除或者校验时，没有任何限制事项。

28.5.1 自编程步骤

使用闪存自编程库改写代码闪存的步骤如图 28-8 所示。

图 28-8 自编程（改写代码闪存）流程



28.5.2 引导交换功能

在通过自编程改写引导区的过程中，如果因瞬间断电等导致改写失败，引导区的数据就会遭到破坏，以致复位时无法重新启动程序并且无法重新编程。

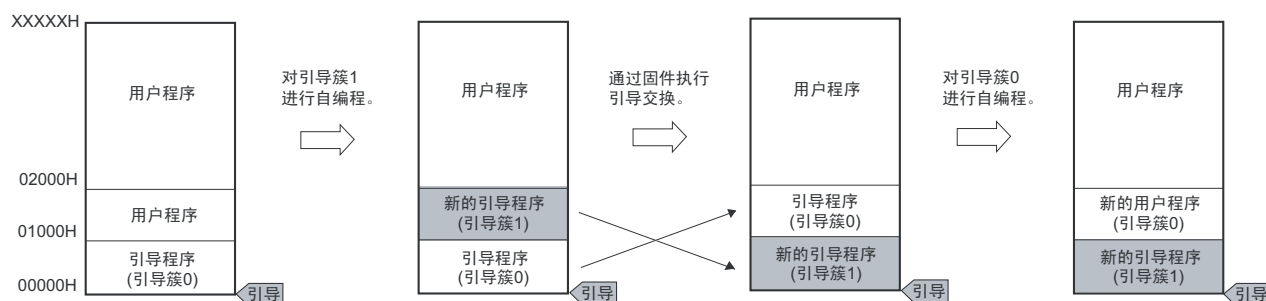
为了避免此问题，提供了引导交换功能。

在擦除引导区的引导簇 0 前，能事先通过自编程对引导簇 1 进行新的引导程序的编程。如果对引导簇 1 的编程正常结束，就能通过 RL78 微控制器内部固件的设定信息功能进行引导簇 1 和引导簇 0 的交换，将引导簇 1 设定为引导区。此后，对原区域的引导簇 0 进行擦除和编程。

从而，即使在改写区域的过程中发生瞬间断电，也能在下次的复位启动时从交换对象的引导簇 1 进行引导，正常启动程序。

注 引导簇是 4K 字节的区域，通过引导交换功能进行引导簇 0 和引导簇 1 的置换。

图 28-9 引导交换功能

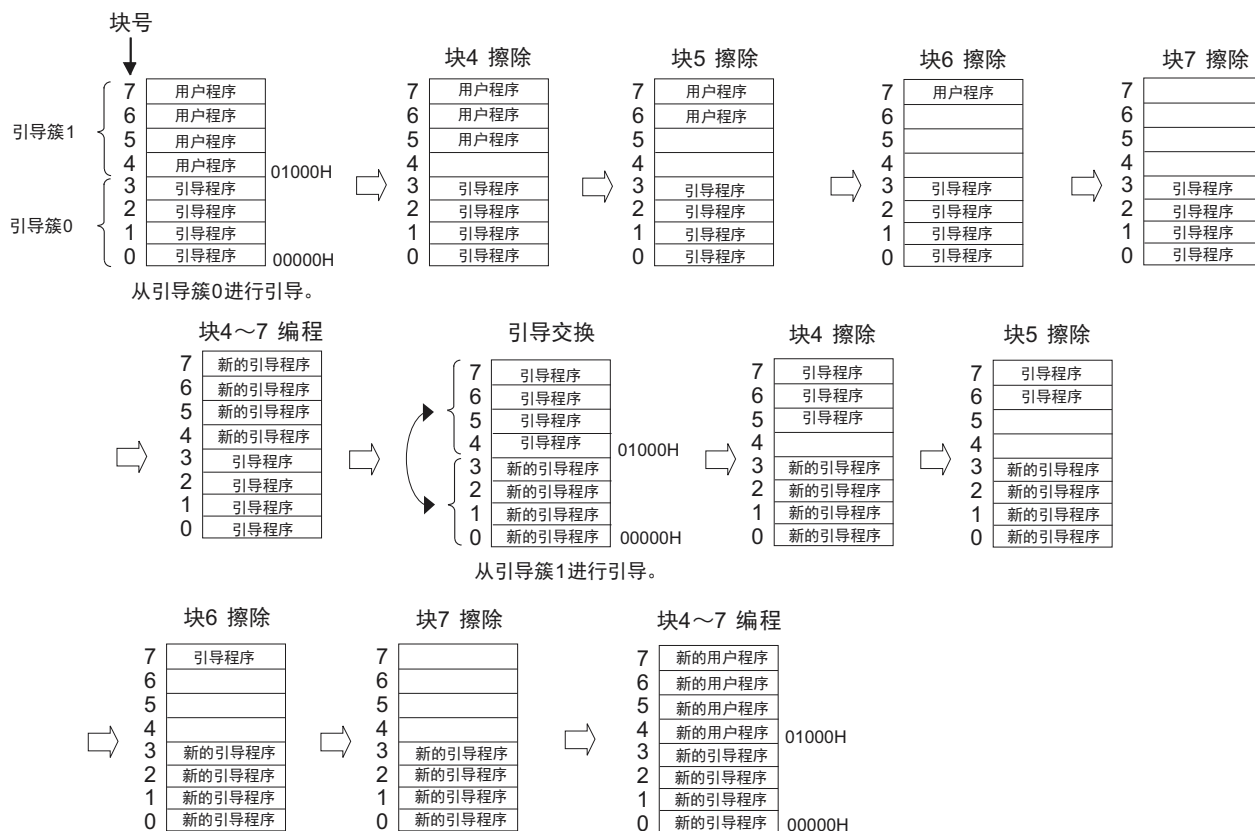


在此图的例子中，

引导簇 0：引导交换前的引导区

引导簇 1：引导交换后的引导区

图 28-10 引导交换的执行例子



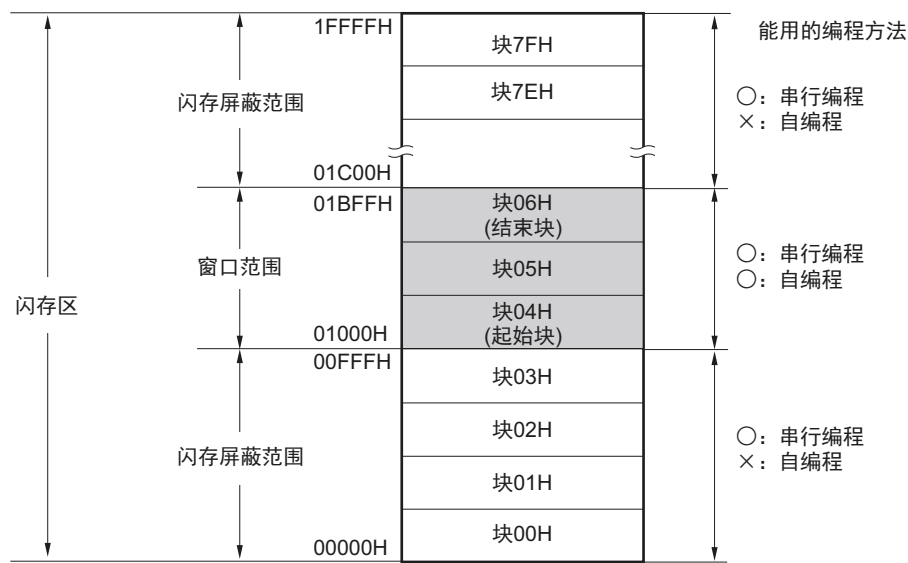
28.5.3 闪存屏蔽窗口功能

闪存屏蔽窗口功能作为自编程的一个安全功能，只在自编程时禁止对指定的窗口范围以外的区域进行编程和擦除。

能通过指定起始块和结束块来设定窗口范围。能在串行编程以及自编程时设定或者更改窗口范围的指定。

在自编程时，禁止对窗口范围以外的区域进行编程和擦除。但是，在串行编程时，也能对指定窗口范围以外的区域进行编程和擦除。

图 28-11 闪存屏蔽窗口的设定例子
(对象器件：R7F0C004，起始块：04H，结束块：06H)



注意 如果引导簇 0 的禁止改写区和闪存屏蔽窗口范围重叠，就优先禁止改写引导簇 0。

表 28-10 闪存屏蔽窗口功能的设定 / 更改方法和命令的关系

编程条件	窗口范围的设定 / 更改方法	执行的命令	
		块擦除	编程
自编程	通过闪存自编程库指定窗口的起始块和结束块。	只能擦除窗口范围内的块。	只能对窗口范围以内的区域进行编程。
串行编程	使用专用闪存编程器的 GUI 等指定窗口的起始块和结束块。	也能擦除窗口范围以外的块。	也能对窗口范围以外的区域进行编程。

备注 要禁止串行编程时的编程或者擦除时，请参照“28.6 安全功能”。

28.6 安全功能

RL78 微控制器支持安全功能，此安全功能禁止改写闪存中的用户程序，并且能防止第三方篡改程序。能通过使用 Security Set 命令进行以下的操作。

- 禁止块擦除
在进行串行编程时，禁止执行闪存中的块擦除命令。但是，在自编程时能进行块擦除。
- 禁止编程
在进行串行编程时，禁止对代码闪存中的全部块执行编程命令。但是，在自编程时能进行编程。在设定为禁止编程后，通过 Security Release 命令进行的解除在复位后有效。
- 禁止改写引导簇 0
禁止对代码闪存中的引导簇 0（00000H～00FFFH）执行块擦除命令和编程命令。

出厂时的初始状态为允许块擦除、编程以及改写引导簇 0。能在串行编程和自编程时设定安全功能，能同时组合使用各种安全设定。

RL78 微控制器的安全功能有效时与擦除、编程命令的关系如表 28-11 所示。

注意 专用闪存编程器的安全功能不支持自编程。

备注 如果要禁止自编程时的编程和擦除，使用闪存屏蔽窗口功能（详细内容请参照 28.5.3）。

表 28-11 安全功能有效时与命令的关系

(1) 串行编程的情况

有效的安全	执行的命令	
	块擦除	编程
禁止块擦除	不能进行块擦除	能进行编程 ^注
禁止编程	能进行块擦除	不能进行编程
禁止改写引导簇 0	不能擦除引导簇 0	不能进行引导簇 0 的编程

注 必须确认编程区中没有数据。由于在禁止块擦除后无法进行擦除，所以当数据没有被擦除时，不能进行数据编程。

(2) 自编程的情况

有效的安全	执行的命令	
	块擦除	编程
禁止块擦除	能进行擦除块	能进行编程
禁止编程		
禁止改写引导簇 0	不能擦除引导簇 0	不能进行引导簇 0 的编程

备注 如果要禁止自编程时的编程和擦除，使用闪存屏蔽窗口功能（详细内容请参照 28.5.3）。

表 28-12 各编程模式中的安全设定方法

(1) 串行编程的情况

安全	安全设定方法	如何使安全设定无效
禁止块擦除	使用专用闪存编程器的 GUI 等进行设定	在设定后无法置为无效
禁止编程		使用专用闪存编程器的 GUI 等进行设定
禁止改写引导簇 0		在设定后无法置为无效

注意 只有在没有设定为“禁止块擦除”或者“禁止改写引导簇 0”并且代码闪存区和数据闪存区为空白时，能解除“禁止编程”的设定。

(2) 自编程的情况

安全	安全设定方法	如何使安全设定无效
禁止块擦除	通过闪存自编程库进行设定	在设定后无法置为无效。
禁止编程		在自编程时无法置为无效（在串行编程时，使用专用闪存编程器的 GUI 等进行设定）。
禁止改写引导簇 0		在设定后无法置为无效。

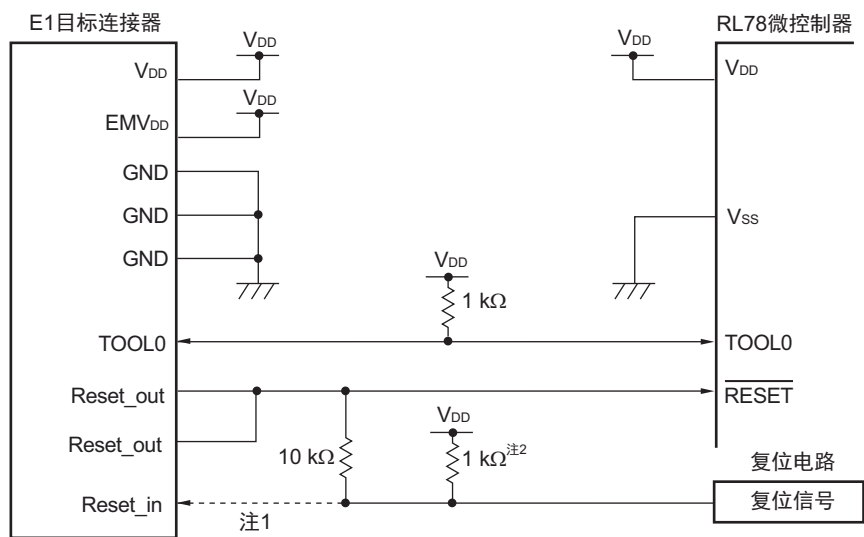
第 29 章 片上调试功能

29.1 与 E1 片上调试仿真器的连接

当 RL78 微控制器通过对应片上调试的 E1 片上调试仿真器与主机进行通信时，使用 V_{DD} 、 $\overline{RESET}$ 、 $TOOL0$ 、 V_{SS} 引脚。通过使用 $TOOL0$ 引脚的单线 UART 进行串行通信。

注意 RL78 微控制器配置了用于开发和评估的片上调试功能。如果使用片上调试功能，就可能超过闪存保证的改写次数而无法保证产品的可靠性，因此不能对量产的产品使用此功能。对于使用片上调试功能的产品，瑞萨电子不承担任何责任。

图 29-1 与 E1 片上调试仿真器的连接例子



- 注 1. 在串行编程时，虚线部分不需要连接。
2. 如果目标系统的复位电路没有缓冲器而只通过电阻或者电容器生成复位信号，就不需要此上拉电阻。

注意 这是假设复位信号的输出为 N 沟道漏极开路缓冲器（输出电阻不大于 100Ω）的电路例子。

29.2 片上调试安全 ID

为了防止他人读取存储器的内容，RL78 微控制器在闪存的 000C3H 中提供了片上调试运行控制位（参照“第 27 章 选项字节”），并且在 000C4H ~ 000CDH 中提供了片上调试安全 ID 设定区。

在自编程时使用引导交换功能的情况下，因为 000C3H 和 000C4H ~ 000CDH 分别被 010C3H 和 010C4H ~ 010CDH 替换，所以必须事先给 010C3H 和 010C4H ~ 010CDH 设定相同的值。

表 29-1 片上调试安全 ID

地址	片上调试安全 ID 码
000C4H ~ 000CDH	任意 10 字节的 ID 码
010C4H ~ 010CDH	

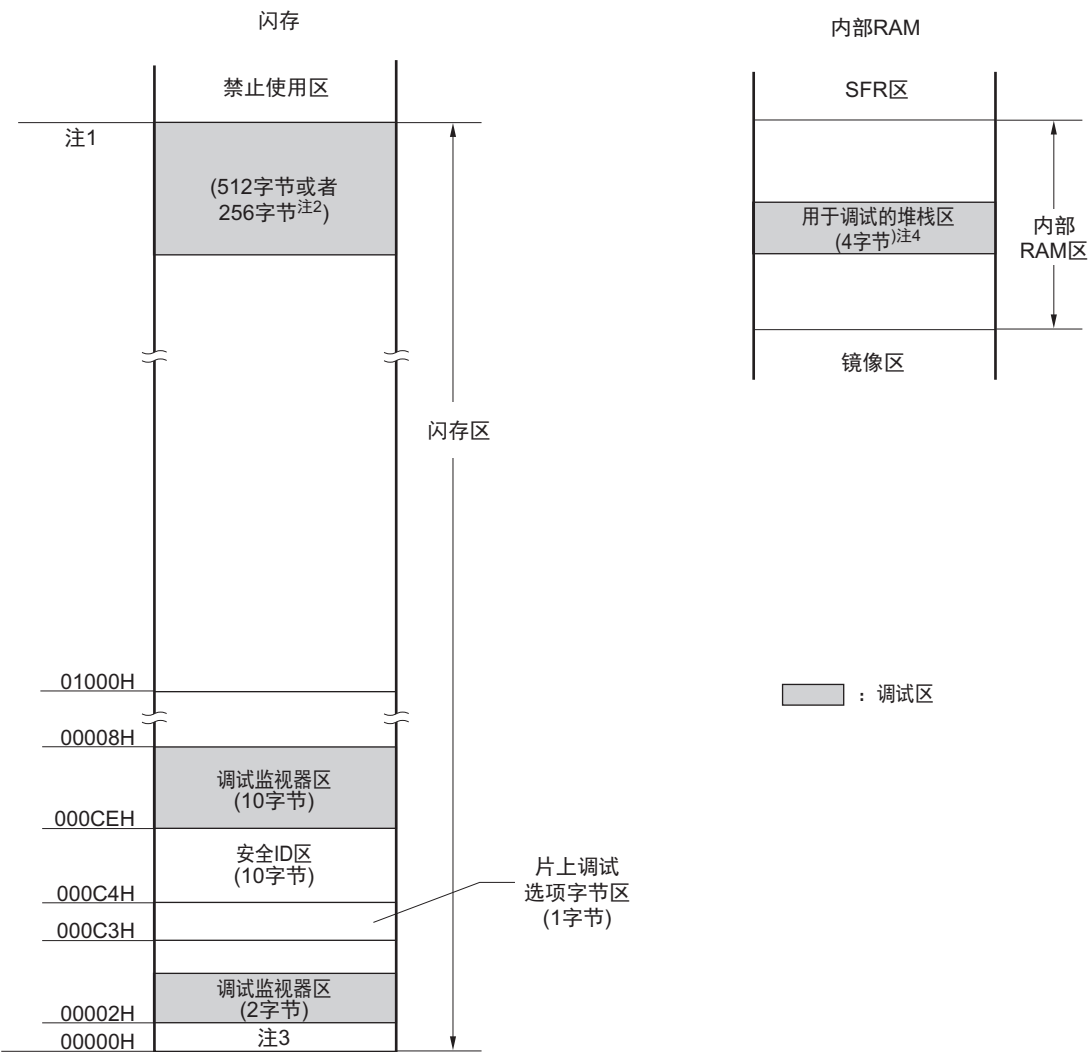
29.3 用户资源的确保

为了实现 RL78 微控制器与 E1 片上调试仿真器的通信和各种调试功能，需提前确保存储空间。当使用本公司的汇编程序和编译程序时，能通过链接程序的选项进行设定。

(1) 存储空间的确保

图 29-2 的灰色部分表示的区域保存用于调试的监控程序，是不能保存用户程序或者数据的空间。要使用片上调试功能时，需要确保不使用此空间的区域，而且不能在用户程序中改写此空间。

图 29-2 保存调试监控程序的存储空间



注 1. 如下所示，地址因产品而不同。

产品	注 1 的地址
R7F0C003	17FFFH
R7F0C004	1FFFFH

- 2. 在不使用实时 RAM 监控（RRM）功能和 Dynamic Memory Modification（DMM）功能时，为 256 字节。
- 3. 在调试时，复位向量被改写为监控程序的分配地址。
- 4. 此区域分配在堆栈区之后，因此用于调试的堆栈区地址随着堆栈的增减而变。即，对于使用的堆栈区，额外占用 4 字节。在自编程时，额外占用 12 字节。

第 30 章 十进制校正（BCD）电路

30.1 十进制校正电路的功能

能以 BCD 码（二进制编码的十进制）计算 BCD 码和 BCD 码相加减的结果。
在执行以 A 寄存器为操作数的加减运算指令后，通过加减 BCD 校正结果寄存器（BCDADJ）的值，计算十进制的校正运算结果。

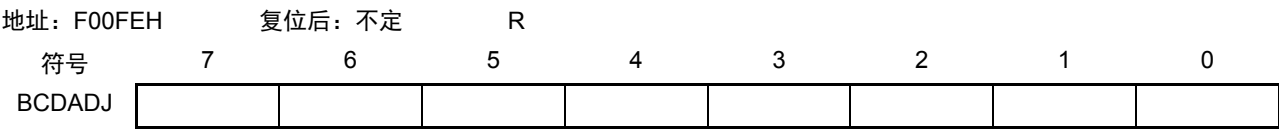
30.2 十进制校正电路使用的寄存器

- 十进制校正电路使用以下的寄存器。
- BCD 校正结果寄存器（BCDADJ）

30.2.1 BCD 校正结果寄存器（BCDADJ）

通过以 A 寄存器为操作数的加减运算指令，将用于计算 BCD 码加减运算结果的校正值保存到 BCDADJ 寄存器。
BCDADJ 寄存器的读取值因读时的 A 寄存器、CY 标志和 AC 标志的值而变。
通过 8 位存储器操作指令读 BCDADJ 寄存器。
在产生复位信号后，此寄存器的值变为不定值。

图 30-1 BCD 校正结果寄存器（BCDADJ）的格式



30.3 十进制校正电路的运行

十进制校正电路的基本运行如下所示。

(1) 加法 以 BCD 码值计算 BCD 码值加上 BCD 码值的结果

- ① 将要相加的BCD码值（被加数）保存到A寄存器。
- ② 以二进制将A寄存器的值和第2个操作数（另一个要相加的BCD码值，加数）相加，二进制的运算结果保存在A寄存器，校正值保存在BCD校正结果寄存器（BCDADJ）。
- ③ 以二进制将A寄存器的值（二进制的加法运算结果）和BCDADJ寄存器（校正值）相加进行十进制校正运算，校正结果保存在A寄存器和CY标志。

注意 BCDADJ寄存器的读取值因读时的A寄存器、CY标志和AC标志的值而变。因此，必须在②的指令后不执行其他指令而执行③的指令。要在允许中断的状态下进行BCD校正时，需要在中断函数中进行A寄存器的压栈和退栈。通过RETI指令恢复PSW（CY标志、AC标志）。

例子如下所示。

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	—

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	—

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 BCD 码值减去 BCD 码值的结果

- ① 将要减去的BCD码值（被减数）保存到A寄存器。
- ② 以二进制将A寄存器的值减去第2个操作数（要减去的BCD码值，减数），二进制的运算结果保存在A寄存器，校正值保存在BCD校正结果寄存器（BCDADJ）。
- ③ 以二进制将A寄存器的值（二进制的减法运算结果）减去BCDADJ寄存器的值（校正值）进行十进制校正运算，校正结果保存在A寄存器和CY标志。

注意 BCDADJ寄存器的读取值因读时的A寄存器、CY标志和AC标志的值而变。因此，必须在②的指令后不执行其他指令而执行③的指令。要在允许中断的状态下进行BCD校正时，需要在中断函数中进行A寄存器的压栈和退栈。通过RETI指令恢复PSW（CY标志、AC标志）。

例子如下所示。

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 31 章 指令集的概要

RL78 单片机的指令集表示在一览表中。有关各指令的详细动作和机器码（指令码），请参照《RL78 family User's Manual: Software》（R01US0015）。

31.1 凡例

31.1.1 操作数的表现形式和记述方法

根据指令操作数的表现形式所对应的记述方法，在各指令的操作数栏中记述了操作数（详细内容请参照汇编程序规格）。当在记述方法中存在多个内容时，选择其中一个。大写字母以及 #、!、!!、\$、\$!、[] 和 ES: 的符号为关键字，应按照原样记述。符号的说明如下：

- #: 指定立即数。
- !: 指定 16 位绝对地址。
- !!: 指定 20 位绝对地址。
- \$: 指定 8 位相对地址。
- \$!: 指定 16 位相对地址。
- []: 指定间接地址。
- ES:: 指定扩展地址。

对于立即数，记述适当的数值或者标号。在使用标号时，必须记述 #、!、!!、\$、\$!、[] 和 ES: 的符号。

对于操作数的寄存器记述形式 r 和 rp，能使用功能名（X、A、C 等）或者绝对名（表 31-1 中括号内的名称，如 R0、R1、R2 等）进行记述。

表 31-1 操作数的表现形式和记述方法

表现形式	记述方法
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	特殊功能寄存器的符号（SFR 符号）FFF00H ~ FFFFFH
sfrp	特殊功能寄存器的符号（可进行 16 位操作的 SFR 符号，只限于偶数地址注）FFF00H ~ FFFFFH
saddr	FFE20H ~ FFF1FH 立即数或者标号
saddrp	FFE20H ~ FFF1FH 立即数或者标号（只限于偶数地址注）
addr20	00000H ~ FFFFFH 立即数或者标号
addr16	0000H ~ FFFFH 立即数或者标号（在 16 位数据时，只限于偶数地址注）
addr5	0080H ~ 00BFH 立即数或者标号（只限于偶数地址注）
word	16 位立即数或者标号
byte	8 位立即数或者标号
bit	3 位立即数或者标号
RBn	RB0 ~ RB3

注 当指定奇数地址时，bit0 为“0”。

备注 能用符号将特殊功能寄存器记述为操作数 sfr。有关特殊功能寄存器的符号，请参照“表 3-5 SFR 一览表”。
能用符号将扩展特殊功能寄存器记述为操作数 laddr16。有关扩展特殊功能寄存器的符号，请参照“表 3-6 扩展 SFR（2nd SFR）一览表”。

31.1.2 操作栏的说明

在各指令的操作栏中，用以下符号表示指令执行时的动作。

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	中断请求允许标志
()	() 内的地址或者寄存器的内容所示的存储器内容
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 (bit19 ~ 16)、 X_H (bit15 ~ 8)、 X_L (bit7 ~ 0)
$\wedge$	逻辑与 (AND)
$\vee$	逻辑或者 (OR)
∇	异或 (exclusive OR)
—	数据取反
addr5	16 位立即数 (只限于偶数地址 0080H ~ 00BFH)
addr16	16 位立即数
addr20	20 位立即数
jdisp8	带符号的 8 位数据 (位移量)
jdisp16	带符号的 16 位数据 (位移量)

31.1.3 标志栏的说明

在各指令的标志栏中，用以下符号表示指令执行时的标志值变化。

表 31-3 标志栏的符号

符号	标志的变化
(Blank)	无变化。
0	清“0”。
1	置“1”。
×	根据结果置位或者复位。
R	恢复以前保存的值。

31.1.4 PREFIX 指令

ES: 所示的指令以 PREFIX 指令码为前缀，将能存取的数据区从 F0000H ~ FFFFFH 的 64K 字节空间扩展为附加 ES 寄存器值的 00000H ~ FFFFFH 的 1M 字节空间。通过将 PREFIX 指令码置于对象指令的前头，只有紧接在 PREFIX 指令码后的 1 条指令作为附加 ES 寄存器值的地址被执行。

在 PREFIX 指令码和紧随其后的 1 条指令之间，不接受中断或者 DMA 传送。

表 31-4 PREFIX 指令码的使用例子

指令	指令码				
	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 在执行 PREFIX 指令前，必须事先通过 MOV ES, A 等设定 ES 寄存器的值。

31.2 操作一览表

表 31-5 操作一览表 (1/17)

指令集	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	r, #byte	2	1	—	r←byte			
		PSW, #byte	3	3	—	PSW←byte	×	×	×
		CS, #byte	3	1	—	CS←byte			
		ES, #byte	2	1	—	ES←byte			
		!addr16, #byte	4	1	—	(addr16)←byte			
		ES:!addr16, #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 注 3	1	1	—	A←r			
		r, A 注 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, !addr16	3	1	4	A←(addr16)			
		A, ES:!addr16	4	2	5	A←(ES, addr16)			
		!addr16, A	3	1	—	(addr16)←A			
		ES:!addr16, A	4	2	—	(ES, addr16)←A			
		A, saddr	2	1	—	A←(saddr)			
		saddr, A	2	1	—	(saddr)←A			

注 1. 这是存取内部 RAM 区、SFR 区或者扩展 SFR 区时或者不存取数据时的 CPU 时钟 (f_{CLK}) 数。

2. 这是存取闪存区时的 CPU 时钟 (f_{CLK}) 数。

3. r=A 除外。

备注 时钟数是在内部 ROM (闪存) 区有程序的情况。当从内部 RAM 区取指令时, 最大为 2 倍 +3 个时钟。

表 31-5 操作一览表 (2/17)

指令集	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, sfr	2	1	—	$A \leftarrow \text{sfr}$			
		sfr, A	2	1	—	$\text{sfr} \leftarrow A$			
		A, [DE]	1	1	4	$A \leftarrow ([DE])$			
		[DE], A	1	1	—	$([DE]) \leftarrow A$			
		A, ES:[DE]	2	2	5	$A \leftarrow (ES, [DE])$			
		ES:[DE], A	2	2	—	$(ES, [DE]) \leftarrow A$			
		A, [HL]	1	1	4	$A \leftarrow ([HL])$			
		[HL], A	1	1	—	$([HL]) \leftarrow A$			
		A, ES:[HL]	2	2	5	$A \leftarrow (ES, [HL])$			
		ES:[HL], A	2	2	—	$(ES, [HL]) \leftarrow A$			
		A, [DE+byte]	2	1	4	$A \leftarrow ([DE+byte])$			
		[DE+byte], A	2	1	—	$([DE+byte]) \leftarrow A$			
		A, ES:[DE+byte]	3	2	5	$A \leftarrow ((ES, [DE+byte]))$			
		ES:[DE+byte], A	3	2	—	$((ES, [DE+byte])) \leftarrow A$			
		A, [HL+byte]	2	1	4	$A \leftarrow ([HL+byte])$			
		[HL+byte], A	2	1	—	$([HL+byte]) \leftarrow A$			
		A, ES:[HL+byte]	3	2	5	$A \leftarrow ((ES, [HL+byte]))$			
		ES:[HL+byte], A	3	2	—	$((ES, [HL+byte])) \leftarrow A$			
		A, [SP+byte]	2	1	—	$A \leftarrow ([SP+byte])$			
		[SP+byte], A	2	1	—	$([SP+byte]) \leftarrow A$			
		A, word[B]	3	1	4	$A \leftarrow (B+word)$			
		word[B], A	3	1	—	$(B+word) \leftarrow A$			
		A, ES:word[B]	4	2	5	$A \leftarrow ((ES, B)+word)$			
		ES:word[B], A	4	2	—	$((ES, B)+word) \leftarrow A$			
		A, word[C]	3	1	4	$A \leftarrow (C+word)$			
		word[C], A	3	1	—	$(C+word) \leftarrow A$			
		A, ES:word[C]	4	2	5	$A \leftarrow ((ES, C)+word)$			
		ES:word[C], A	4	2	—	$((ES, C)+word) \leftarrow A$			
		A, word[BC]	3	1	4	$A \leftarrow (BC+word)$			
		word[BC], A	3	1	—	$(BC+word) \leftarrow A$			
		A, ES:word[BC]	4	2	5	$A \leftarrow ((ES, BC)+word)$			
		ES:word[BC], A	4	2	—	$((ES, BC)+word) \leftarrow A$			

注 1. 这是存取内部 RAM 区、SFR 区或者扩展 SFR 区时或者不存取数据时的 CPU 时钟 (f_{CLK}) 数。

2. 这是存取闪存区时的 CPU 时钟 (f_{CLK}) 数。

备注 时钟数是在内部 ROM (闪存) 区有程序的情况。当从内部 RAM 区取指令时, 最大为 2 倍 +3 个时钟。

表 31-5 操作一览表 (3/17)

指令集	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, !addr16	3	1	4	$X \leftarrow (addr16)$			
		X, ES:!addr16	4	2	5	$X \leftarrow (ES, addr16)$			
		X, saddr	2	1	—	$X \leftarrow (saddr)$			
		B, !addr16	3	1	4	$B \leftarrow (addr16)$			
		B, ES:!addr16	4	2	5	$B \leftarrow (ES, addr16)$			
		B, saddr	2	1	—	$B \leftarrow (saddr)$			
		C, !addr16	3	1	4	$C \leftarrow (addr16)$			
		C, ES:!addr16	4	2	5	$C \leftarrow (ES, addr16)$			
		C, saddr	2	1	—	$C \leftarrow (saddr)$			
		ES, saddr	3	1	—	$ES \leftarrow (saddr)$			
	XCH	A, r 注 3	1 (r=X) 2 (r=X 以外)	1	—	$A \leftrightarrow r$			
		A, !addr16	4	2	—	$A \leftrightarrow (addr16)$			
		A, ES:!addr16	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. 这是存取内部 RAM 区、SFR 区或者扩展 SFR 区时或者不存取数据时的 CPU 时钟 (f_{CLK}) 数。

2. 这是存取闪存区时的 CPU 时钟 (f_{CLK}) 数。

3. r=A 除外。

备注 时钟数是在内部 ROM (闪存) 区有程序的情况。当从内部 RAM 区取指令时, 最大为 2 倍 +3 个时钟。

表 31-5 操作一览表 (4/17)

指令集	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	XCH	A, [HL+B]	2	2	—	$A \leftrightarrow (HL+B)$			
		A, ES:[HL+B]	3	3	—	$A \leftrightarrow ((ES, HL)+B)$			
		A, [HL+C]	2	2	—	$A \leftrightarrow (HL+C)$			
		A, ES:[HL+C]	3	3	—	$A \leftrightarrow ((ES, HL)+C)$			
	ONEB	A	1	1	—	$A \leftarrow 01H$			
		X	1	1	—	$X \leftarrow 01H$			
		B	1	1	—	$B \leftarrow 01H$			
		C	1	1	—	$C \leftarrow 01H$			
		!addr16	3	1	—	$(addr16) \leftarrow 01H$			
		ES:!addr16	4	2	—	$(ES, addr16) \leftarrow 01H$			
		saddr	2	1	—	$(saddr) \leftarrow 01H$			
	CLRB	A	1	1	—	$A \leftarrow 00H$			
		X	1	1	—	$X \leftarrow 00H$			
		B	1	1	—	$B \leftarrow 00H$			
		C	1	1	—	$C \leftarrow 00H$			
		!addr16	3	1	—	$(addr16) \leftarrow 00H$			
		ES:!addr16	4	2	—	$(ES, addr16) \leftarrow 00H$			
		saddr	2	1	—	$(saddr) \leftarrow 00H$			
	MOVS	[HL+byte], X	3	1	—	$(HL+byte) \leftarrow X$	×		×
		ES:[HL+byte], X	4	2	—	$(ES, HL+byte) \leftarrow X$	×		×
16 位数据传送	MOVW	rp, #word	3	1	—	$rp \leftarrow word$			
		saddrp, #word	4	1	—	$(saddrp) \leftarrow word$			
		sfrp, #word	4	1	—	$sfrp \leftarrow word$			
		AX, rp 注 3	1	1	—	$AX \leftarrow rp$			
		rp, AX 注 3	1	1	—	$rp \leftarrow AX$			
		AX, !addr16	3	1	4	$AX \leftarrow (addr16)$			
		!addr16, AX	3	1	—	$(addr16) \leftarrow AX$			
		AX, ES:!addr16	4	2	5	$AX \leftarrow (ES, addr16)$			
		ES:!addr16, AX	4	2	—	$(ES, addr16) \leftarrow AX$			
		AX, saddrp	2	1	—	$AX \leftarrow (saddrp)$			
		saddrp, AX	2	1	—	$(saddrp) \leftarrow AX$			
		AX, sfrp	2	1	—	$AX \leftarrow sfrp$			
		sfrp, AX	2	1	—	$sfrp \leftarrow AX$			

注 1. 这是存取内部 RAM 区、SFR 区或者扩展 SFR 区时或者不存取数据时的 CPU 时钟 (f_{CLK}) 数。

2. 这是存取闪存区时的 CPU 时钟 (f_{CLK}) 数。

3. $rp=AX$ 除外。

备注 时钟数是在内部 ROM (闪存) 区有程序的情况。当从内部 RAM 区取指令时, 最大为 2 倍 +3 个时钟。

表 31-5 操作一览表 (5/17)

指令集	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
16 位 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. 这是存取内部 RAM 区、SFR 区或者扩展 SFR 区时或者不存取数据时的 CPU 时钟 (f_{CLK}) 数。

2. 这是存取闪存区时的 CPU 时钟 (f_{CLK}) 数。

备注 时钟数是在内部 ROM (闪存) 区有程序的情况。当从内部 RAM 区取指令时, 最大为 2 倍 +3 个时钟。

表 31-5 操作一览表 (6/17)

指令集	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
16 位 数 据 传 送	MOVW	BC, !addr16	3	1	4	BC←(addr16)			
		BC, ES:!addr16	4	2	5	BC←(ES, addr16)			
		DE, !addr16	3	1	4	DE←(addr16)			
		DE, ES:!addr16	4	2	5	DE←(ES, addr16)			
		HL, !addr16	3	1	4	HL←(addr16)			
		HL, ES:!addr16	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 运 算	ADD	A, #byte	2	1	—	A, CY←A+byte	×	×	×
		saddr, #byte	3	2	—	(saddr), CY←(saddr)+byte	×	×	×
		A, r 注 4	2	1	—	A, CY←A+r	×	×	×
		r, A	2	1	—	r, CY←r+A	×	×	×
		A, !addr16	3	1	4	A, CY←A+(addr16)	×	×	×
		A, ES:!addr16	4	2	5	A, CY←A+(ES, addr16)	×	×	×
		A, saddr	2	1	—	A, CY←A+(saddr)	×	×	×
		A, [HL]	1	1	4	A, CY←A+(HL)	×	×	×
		A, ES:[HL]	2	2	5	A, CY←A+(ES, HL)	×	×	×
		A, [HL+byte]	2	1	4	A, CY←A+(HL+byte)	×	×	×
		A, ES:[HL+byte]	3	2	5	A, CY←A+((ES, HL)+byte)	×	×	×
		A, [HL+B]	2	1	4	A, CY←A+(HL+B)	×	×	×
		A, ES:[HL+B]	3	2	5	A, CY←A+((ES, HL)+B)	×	×	×
		A, [HL+C]	2	1	4	A, CY←A+(HL+C)	×	×	×
		A, ES:[HL+C]	3	2	5	A, CY←A+((ES, HL)+C)	×	×	×

注 1. 这是存取内部 RAM 区、SFR 区或者扩展 SFR 区时或者不存取数据时的 CPU 时钟 (f_{CLK}) 数。

2. 这是存取闪存区时的 CPU 时钟 (f_{CLK}) 数。

3. rp=AX 除外。

4. r=A 除外。

备注 时钟数是在内部 ROM (闪存) 区有程序的情况。当从内部 RAM 区取指令时, 最大为 2 倍 +3 个时钟。

表 31-5 操作一览表 (7/17)

指令集	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
8 位运算	ADDC	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, !addr16	3	1	4	A, CY←A+(addr16)+CY	×	×	×
		A, ES:!addr16	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	×	×	×
	SUB	A, #byte	2	1	—	A, CY←A-byte	×	×	×
		saddr, #byte	3	2	—	(saddr), CY←(saddr)-byte	×	×	×
		A, r 注 3	2	1	—	A, CY←A-r	×	×	×
		r, A	2	1	—	r, CY←r-A	×	×	×
		A, !addr16	3	1	4	A, CY←A-(addr16)	×	×	×
		A, ES:!addr16	4	2	5	A, CY←A-(ES, addr16)	×	×	×
		A, saddr	2	1	—	A, CY←A-(saddr)	×	×	×
		A, [HL]	1	1	4	A, CY←A-(HL)	×	×	×
		A, ES:[HL]	2	2	5	A, CY←A-(ES, HL)	×	×	×
		A, [HL+byte]	2	1	4	A, CY←A-(HL+byte)	×	×	×
		A, ES:[HL+byte]	3	2	5	A, CY←A-((ES, HL)+byte)	×	×	×
		A, [HL+B]	2	1	4	A, CY←A-(HL+B)	×	×	×
		A, ES:[HL+B]	3	2	5	A, CY←A-((ES, HL)+B)	×	×	×
		A, [HL+C]	2	1	4	A, CY←A-(HL+C)	×	×	×
		A, ES:[HL+C]	3	2	5	A, CY←A-((ES, HL)+C)	×	×	×

注 1. 这是存取内部 RAM 区、SFR 区或者扩展 SFR 区时或者不存取数据时的 CPU 时钟 (f_{CLK}) 数。

2. 这是存取闪存区时的 CPU 时钟 (f_{CLK}) 数。

3. r=A 除外。

备注 时钟数是在内部 ROM (闪存) 区有程序的情况。当从内部 RAM 区取指令时, 最大为 2 倍 +3 个时钟。

表 31-5 操作一览表 (8/17)

指令集	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
8 位运算	SUBC	A, #byte	2	1	—	$A, CY \leftarrow A - \text{byte} - CY$	×	×	×
		saddr, #byte	3	2	—	$(saddr), CY \leftarrow (saddr) - \text{byte} - CY$	×	×	×
		A, r 注 3	2	1	—	$A, CY \leftarrow A - r - CY$	×	×	×
		r, A	2	1	—	$r, CY \leftarrow r - A - CY$	×	×	×
		A, !addr16	3	1	4	$A, CY \leftarrow A - (\text{addr16}) - CY$	×	×	×
		A, ES:!addr16	4	2	5	$A, CY \leftarrow A - (ES, \text{addr16}) - CY$	×	×	×
		A, saddr	2	1	—	$A, CY \leftarrow A - (saddr) - CY$	×	×	×
		A, [HL]	1	1	4	$A, CY \leftarrow A - (HL) - CY$	×	×	×
		A, ES:[HL]	2	2	5	$A, CY \leftarrow A - (ES, HL) - CY$	×	×	×
		A, [HL+byte]	2	1	4	$A, CY \leftarrow A - (HL + \text{byte}) - CY$	×	×	×
		A, ES:[HL+byte]	3	2	5	$A, CY \leftarrow A - ((ES, HL) + \text{byte}) - CY$	×	×	×
		A, [HL+B]	2	1	4	$A, CY \leftarrow A - (HL + B) - CY$	×	×	×
		A, ES:[HL+B]	3	2	5	$A, CY \leftarrow A - ((ES, HL) + B) - CY$	×	×	×
		A, [HL+C]	2	1	4	$A, CY \leftarrow A - (HL + C) - CY$	×	×	×
		A, ES:[HL+C]	3	2	5	$A, CY \leftarrow A - ((ES, HL) + C) - CY$	×	×	×
	AND	A, #byte	2	1	—	$A \leftarrow A \wedge \text{byte}$	×		
		saddr, #byte	3	2	—	$(saddr) \leftarrow (saddr) \wedge \text{byte}$	×		
		A, r 注 3	2	1	—	$A \leftarrow A \wedge r$	×		
		r, A	2	1	—	$r \leftarrow r \wedge A$	×		
		A, !addr16	3	1	4	$A \leftarrow A \wedge (\text{addr16})$	×		
		A, ES:!addr16	4	2	5	$A \leftarrow A \wedge (ES: \text{addr16})$	×		
		A, saddr	2	1	—	$A \leftarrow A \wedge (saddr)$	×		
		A, [HL]	1	1	4	$A \leftarrow A \wedge (HL)$	×		
		A, ES:[HL]	2	2	5	$A \leftarrow A \wedge (ES: HL)$	×		
		A, [HL+byte]	2	1	4	$A \leftarrow A \wedge (HL + \text{byte})$	×		
		A, ES:[HL+byte]	3	2	5	$A \leftarrow A \wedge ((ES: HL) + \text{byte})$	×		
		A, [HL+B]	2	1	4	$A \leftarrow A \wedge (HL + B)$	×		
		A, ES:[HL+B]	3	2	5	$A \leftarrow A \wedge ((ES: HL) + B)$	×		
		A, [HL+C]	2	1	4	$A \leftarrow A \wedge (HL + C)$	×		
		A, ES:[HL+C]	3	2	5	$A \leftarrow A \wedge ((ES: HL) + C)$	×		

注 1. 这是存取内部 RAM 区、SFR 区或者扩展 SFR 区时或者不存取数据时的 CPU 时钟 (f_{CLK}) 数。

2. 这是存取闪存区时的 CPU 时钟 (f_{CLK}) 数。

3. $r=A$ 除外。

备注 时钟数是在内部 ROM (闪存) 区有程序的情况。当从内部 RAM 区取指令时, 最大为 2 倍 +3 个时钟。

表 31-5 操作一览表 (9/17)

指令集	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
8 位运算	OR	A, #byte	2	1	—	$A \leftarrow A \vee \text{byte}$	×		
		saddr, #byte	3	2	—	$(\text{saddr}) \leftarrow (\text{saddr}) \vee \text{byte}$	×		
		A, r 注 3	2	1	—	$A \leftarrow A \vee r$	×		
		r, A	2	1	—	$r \leftarrow r \vee A$	×		
		A, !addr16	3	1	4	$A \leftarrow A \vee (\text{addr16})$	×		
		A, ES:!addr16	4	2	5	$A \leftarrow A \vee (\text{ES:addr16})$	×		
		A, saddr	2	1	—	$A \leftarrow A \vee (\text{saddr})$	×		
		A, [HL]	1	1	4	$A \leftarrow A \vee (\text{HL})$	×		
		A, ES:[HL]	2	2	5	$A \leftarrow A \vee (\text{ES:HL})$	×		
		A, [HL+byte]	2	1	4	$A \leftarrow A \vee (\text{HL} + \text{byte})$	×		
		A, ES:[HL+byte]	3	2	5	$A \leftarrow A \vee ((\text{ES:HL}) + \text{byte})$	×		
		A, [HL+B]	2	1	4	$A \leftarrow A \vee (\text{HL} + B)$	×		
		A, ES:[HL+B]	3	2	5	$A \leftarrow A \vee ((\text{ES:HL}) + B)$	×		
		A, [HL+C]	2	1	4	$A \leftarrow A \vee (\text{HL} + C)$	×		
		A, ES:[HL+C]	3	2	5	$A \leftarrow A \vee ((\text{ES:HL}) + C)$	×		
	XOR	A, #byte	2	1	—	$A \leftarrow A \oplus \text{byte}$	×		
		saddr, #byte	3	2	—	$(\text{saddr}) \leftarrow (\text{saddr}) \oplus \text{byte}$	×		
		A, r 注 3	2	1	—	$A \leftarrow A \oplus r$	×		
		r, A	2	1	—	$r \leftarrow r \oplus A$	×		
		A, !addr16	3	1	4	$A \leftarrow A \oplus (\text{addr16})$	×		
		A, ES:!addr16	4	2	5	$A \leftarrow A \oplus (\text{ES:addr16})$	×		
		A, saddr	2	1	—	$A \leftarrow A \oplus (\text{saddr})$	×		
		A, [HL]	1	1	4	$A \leftarrow A \oplus (\text{HL})$	×		
		A, ES:[HL]	2	2	5	$A \leftarrow A \oplus (\text{ES:HL})$	×		
		A, [HL+byte]	2	1	4	$A \leftarrow A \oplus (\text{HL} + \text{byte})$	×		
		A, ES:[HL+byte]	3	2	5	$A \leftarrow A \oplus ((\text{ES:HL}) + \text{byte})$	×		
		A, [HL+B]	2	1	4	$A \leftarrow A \oplus (\text{HL} + B)$	×		
		A, ES:[HL+B]	3	2	5	$A \leftarrow A \oplus ((\text{ES:HL}) + B)$	×		
		A, [HL+C]	2	1	4	$A \leftarrow A \oplus (\text{HL} + C)$	×		
		A, ES:[HL+C]	3	2	5	$A \leftarrow A \oplus ((\text{ES:HL}) + C)$	×		

注 1. 这是存取内部 RAM 区、SFR 区或者扩展 SFR 区时或者不存取数据时的 CPU 时钟 (f_{CLK}) 数。

2. 这是存取闪存区时的 CPU 时钟 (f_{CLK}) 数。

3. $r=A$ 除外。

备注 时钟数是在内部 ROM (闪存) 区有程序的情况。当从内部 RAM 区取指令时, 最大为 2 倍 +3 个时钟。

表 31-5 操作一览表 (10/17)

指令集	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
8 位运算	CMP	A, #byte	2	1	—	A-byte	×	×	×
		!addr16, #byte	4	1	4	(addr16)-byte	×	×	×
		ES:!addr16, #byte	5	2	5	(ES:addr16)-byte	×	×	×
		saddr, #byte	3	1	—	(saddr)-byte	×	×	×
		A, r 注 3	2	1	—	A-r	×	×	×
		r, A	2	1	—	r-A	×	×	×
		A, !addr16	3	1	4	A-(addr16)	×	×	×
		A, ES:!addr16	4	2	5	A-(ES:addr16)	×	×	×
		A, saddr	2	1	—	A-(saddr)	×	×	×
		A, [HL]	1	1	4	A-(HL)	×	×	×
		A, ES:[HL]	2	2	5	A-(ES:HL)	×	×	×
		A, [HL+byte]	2	1	4	A-(HL+byte)	×	×	×
		A, ES:[HL+byte]	3	2	5	A-((ES:HL)+byte)	×	×	×
		A, [HL+B]	2	1	4	A-(HL+B)	×	×	×
		A, ES:[HL+B]	3	2	5	A-((ES:HL)+B)	×	×	×
		A, [HL+C]	2	1	4	A-(HL+C)	×	×	×
		A, ES:[HL+C]	3	2	5	A-((ES:HL)+C)	×	×	×
	CMP0	A	1	1	—	A-00H	×	0	0
		X	1	1	—	X-00H	×	0	0
		B	1	1	—	B-00H	×	0	0
		C	1	1	—	C-00H	×	0	0
		!addr16	3	1	4	(addr16)-00H	×	0	0
		ES:!addr16	4	2	5	(ES:addr16)-00H	×	0	0
		saddr	2	1	—	(saddr)-00H	×	0	0
	CMPS	X, [HL+byte]	3	1	4	X-(HL+byte)	×	×	×
		X, ES:[HL+byte]	4	2	5	X-((ES:HL)+byte)	×	×	×

注 1. 这是存取内部 RAM 区、SFR 区或者扩展 SFR 区时或者不存取数据时的 CPU 时钟 (f_{CLK}) 数。

2. 这是存取闪存区时的 CPU 时钟 (f_{CLK}) 数。

3. r=A 除外。

备注 时钟数是在内部 ROM (闪存) 区有程序的情况。当从内部 RAM 区取指令时, 最大为 2 倍 +3 个时钟。

表 31-5 操作一览表 (11/17)

指令集	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
16 位运算	ADDW	AX, #word	3	1	—	AX, CY←AX+word	×	×	×
		AX, AX	1	1	—	AX, CY←AX+AX	×	×	×
		AX, BC	1	1	—	AX, CY←AX+BC	×	×	×
		AX, DE	1	1	—	AX, CY←AX+DE	×	×	×
		AX, HL	1	1	—	AX, CY←AX+HL	×	×	×
		AX, !addr16	3	1	4	AX, CY←AX+(addr16)	×	×	×
		AX, ES:!addr16	4	2	5	AX, CY←AX+(ES:addr16)	×	×	×
		AX, saddrp	2	1	—	AX, CY←AX+(saddrp)	×	×	×
		AX, [HL+byte]	3	1	4	AX, CY←AX+(HL+byte)	×	×	×
		AX, ES: [HL+byte]	4	2	5	AX, CY←AX+((ES:HL)+byte)	×	×	×
	SUBW	AX, #word	3	1	—	AX, CY←AX-word	×	×	×
		AX, BC	1	1	—	AX, CY←AX-BC	×	×	×
		AX, DE	1	1	—	AX, CY←AX-DE	×	×	×
		AX, HL	1	1	—	AX, CY←AX-HL	×	×	×
		AX, !addr16	3	1	4	AX, CY←AX-(addr16)	×	×	×
		AX, ES:!addr16	4	2	5	AX, CY←AX-(ES:addr16)	×	×	×
		AX, saddrp	2	1	—	AX, CY←AX-(saddrp)	×	×	×
		AX, [HL+byte]	3	1	4	AX, CY←AX-(HL+byte)	×	×	×
		AX, ES: [HL+byte]	4	2	5	AX, CY←AX-((ES:HL)+byte)	×	×	×
	CMPW	AX, #word	3	1	—	AX-word	×	×	×
		AX, BC	1	1	—	AX-BC	×	×	×
		AX, DE	1	1	—	AX-DE	×	×	×
		AX, HL	1	1	—	AX-HL	×	×	×
		AX, !addr16	3	1	4	AX-(addr16)	×	×	×
		AX, ES:!addr16	4	2	5	AX-(ES:addr16)	×	×	×
		AX, saddrp	2	1	—	AX-(saddrp)	×	×	×
		AX, [HL+byte]	3	1	4	AX-(HL+byte)	×	×	×
		AX, ES: [HL+byte]	4	2	5	AX-((ES:HL)+byte)	×	×	×
乘	MULU	X	1	1	—	AX←A×X			

注 1. 这是存取内部 RAM 区、SFR 区或者扩展 SFR 区时或者不存取数据时的 CPU 时钟 (f_{CLK}) 数。

2. 这是存取闪存区时的 CPU 时钟 (f_{CLK}) 数。

备注 时钟数是在内部 ROM (闪存) 区有程序的情况。当从内部 RAM 区取指令时, 最大为 2 倍 +3 个时钟。

表 31-5 操作一览表 (12/17)

指令集	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
递增 / 递减	INC	r	1	1	—	$r \leftarrow r+1$	×	×	
		!addr16	3	2	—	$(addr16) \leftarrow (addr16)+1$	×	×	
		ES:!addr16	4	3	—	$(ES, addr16) \leftarrow (ES, addr16)+1$	×	×	
		saddr	2	2	—	$(saddr) \leftarrow (saddr)+1$	×	×	
		[HL+byte]	3	2	—	$(HL+byte) \leftarrow (HL+byte)+1$	×	×	
		ES: [HL+byte]	4	3	—	$((ES:HL)+byte) \leftarrow ((ES:HL)+byte)+1$	×	×	
	DEC	r	1	1	—	$r \leftarrow r-1$	×	×	
		!addr16	3	2	—	$(addr16) \leftarrow (addr16)-1$	×	×	
		ES:!addr16	4	3	—	$(ES, addr16) \leftarrow (ES, addr16)-1$	×	×	
		saddr	2	2	—	$(saddr) \leftarrow (saddr)-1$	×	×	
		[HL+byte]	3	2	—	$(HL+byte) \leftarrow (HL+byte)-1$	×	×	
		ES: [HL+byte]	4	3	—	$((ES:HL)+byte) \leftarrow ((ES:HL)+byte)-1$	×	×	
	INCW	rp	1	1	—	$rp \leftarrow rp+1$			
		!addr16	3	2	—	$(addr16) \leftarrow (addr16)+1$			
		ES:!addr16	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	—	$rp \leftarrow rp-1$			
		!addr16	3	2	—	$(addr16) \leftarrow (addr16)-1$			
		ES:!addr16	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	—	$(CY \leftarrow A_0, A_{m-1} \leftarrow A_m, A_7 \leftarrow 0) \times cnt$			×
	SHRW	AX, cnt	2	1	—	$(CY \leftarrow AX_0, AX_{m-1} \leftarrow AX_m, AX_{15} \leftarrow 0) \times cnt$			×
	SHL	A, cnt	2	1	—	$(CY \leftarrow A_7, A_m \leftarrow A_{m-1}, A_0 \leftarrow 0) \times cnt$			×
		B, cnt	2	1	—	$(CY \leftarrow B_7, B_m \leftarrow B_{m-1}, B_0 \leftarrow 0) \times cnt$			×
		C, cnt	2	1	—	$(CY \leftarrow C_7, C_m \leftarrow C_{m-1}, C_0 \leftarrow 0) \times cnt$			×
	SHLW	AX, cnt	2	1	—	$(CY \leftarrow AX_{15}, AX_m \leftarrow AX_{m-1}, AX_0 \leftarrow 0) \times cnt$			×
		BC, cnt	2	1	—	$(CY \leftarrow BC_{15}, BC_m \leftarrow BC_{m-1}, BC_0 \leftarrow 0) \times cnt$			×
	SAR	A, cnt	2	1	—	$(CY \leftarrow A_0, A_{m-1} \leftarrow A_m, A_7 \leftarrow A_7) \times cnt$			×
	SARW	AX, cnt	2	1	—	$(CY \leftarrow AX_0, AX_{m-1} \leftarrow AX_m, AX_{15} \leftarrow AX_{15}) \times cnt$			×

注 1. 这是存取内部 RAM 区、SFR 区或者扩展 SFR 区时或者不存取数据时的 CPU 时钟 (f_{CLK}) 数。

2. 这是存取闪存区时的 CPU 时钟 (f_{CLK}) 数。

备注 1. 时钟数是在内部 ROM (闪存) 区有程序的情况。当从内部 RAM 区取指令时, 最大为 2 倍 +3 个时钟。

2. cnt 是位的移位个数。

表 31-5 操作一览表 (13/17)

指令集	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
循环	ROR	A, 1	2	1	—	$(CY, A_7 \leftarrow A_0, A_{m-1} \leftarrow A_m) \times 1$			×
	ROL	A, 1	2	1	—	$(CY, A_0 \leftarrow A_7, A_{m+1} \leftarrow A_m) \times 1$			×
	RORC	A, 1	2	1	—	$(CY \leftarrow A_0, A_7 \leftarrow CY, A_{m-1} \leftarrow A_m) \times 1$			×
	ROLC	A, 1	2	1	—	$(CY \leftarrow A_7, A_0 \leftarrow CY, A_{m+1} \leftarrow A_m) \times 1$			×
	ROLWC	AX, 1	2	1	—	$(CY \leftarrow AX_{15}, AX_0 \leftarrow CY, AX_{m+1} \leftarrow AX_m) \times 1$			×
		BC, 1	2	1	—	$(CY \leftarrow BC_{15}, BC_0 \leftarrow CY, BC_{m+1} \leftarrow BC_m) \times 1$			×
位操作	MOV1	CY, A.bit	2	1	—	$CY \leftarrow A.bit$			×
		A.bit, CY	2	1	—	$A.bit \leftarrow CY$			
		CY, PSW.bit	3	1	—	$CY \leftarrow PSW.bit$			×
		PSW.bit, CY	3	4	—	$PSW.bit \leftarrow CY$	×	×	
		CY, saddr.bit	3	1	—	$CY \leftarrow (saddr).bit$			×
		saddr.bit, CY	3	2	—	$(saddr).bit \leftarrow CY$			
		CY, sfr.bit	3	1	—	$CY \leftarrow sfr.bit$			×
		sfr.bit, CY	3	2	—	$sfr.bit \leftarrow CY$			
		CY, [HL].bit	2	1	4	$CY \leftarrow (HL).bit$			×
		[HL].bit, CY	2	2	—	$(HL).bit \leftarrow CY$			
		CY, ES:[HL].bit	3	2	5	$CY \leftarrow (ES, HL).bit$			×
		ES:[HL].bit, CY	3	3	—	$(ES, HL).bit \leftarrow CY$			
	AND1	CY, A.bit	2	1	—	$CY \leftarrow CY \wedge A.bit$			×
		CY, PSW.bit	3	1	—	$CY \leftarrow CY \wedge PSW.bit$			×
		CY, saddr.bit	3	1	—	$CY \leftarrow CY \wedge (saddr).bit$			×
		CY, sfr.bit	3	1	—	$CY \leftarrow CY \wedge sfr.bit$			×
		CY, [HL].bit	2	1	4	$CY \leftarrow CY \wedge (HL).bit$			×
		CY, ES:[HL].bit	3	2	5	$CY \leftarrow CY \wedge (ES, HL).bit$			×
	OR1	CY, A.bit	2	1	—	$CY \leftarrow CY \vee A.bit$			×
		CY, PSW.bit	3	1	—	$CY \leftarrow CY \vee PSW.bit$			×
		CY, saddr.bit	3	1	—	$CY \leftarrow CY \vee (saddr).bit$			×
		CY, sfr.bit	3	1	—	$CY \leftarrow CY \vee sfr.bit$			×
		CY, [HL].bit	2	1	4	$CY \leftarrow CY \vee (HL).bit$			×
		CY, ES:[HL].bit	3	2	5	$CY \leftarrow CY \vee (ES, HL).bit$			×

注 1. 这是存取内部 RAM 区、SFR 区或者扩展 SFR 区时或者不存取数据时的 CPU 时钟 (f_{CLK}) 数。

2. 这是存取闪存区时的 CPU 时钟 (f_{CLK}) 数。

备注 时钟数是在内部 ROM (闪存) 区有程序的情况。当从内部 RAM 区取指令时, 最大为 2 倍 +3 个时钟。

表 31-5 操作一览表 (14/17)

指令集	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
位操作	XOR1	CY, A.bit	2	1	—	$CY \leftarrow CY \oplus A.bit$			×
		CY, PSW.bit	3	1	—	$CY \leftarrow CY \oplus PSW.bit$			×
		CY, saddr.bit	3	1	—	$CY \leftarrow CY \oplus (saddr).bit$			×
		CY, sfr.bit	3	1	—	$CY \leftarrow CY \oplus sfr.bit$			×
		CY, [HL].bit	2	1	4	$CY \leftarrow CY \oplus (HL).bit$			×
		CY, ES:[HL].bit	3	2	5	$CY \leftarrow CY \oplus (ES, HL).bit$			×
	SET1	A.bit	2	1	—	$A.bit \leftarrow 1$			
		PSW.bit	3	4	—	$PSW.bit \leftarrow 1$	×	×	×
		laddr16.bit	4	2	—	$(addr16).bit \leftarrow 1$			
		ES:laddr16.bit	5	3	—	$(ES, addr16).bit \leftarrow 1$			
		saddr.bit	3	2	—	$(saddr).bit \leftarrow 1$			
		sfr.bit	3	2	—	$sfr.bit \leftarrow 1$			
		[HL].bit	2	2	—	$(HL).bit \leftarrow 1$			
		ES:[HL].bit	3	3	—	$(ES, HL).bit \leftarrow 1$			
	CLR1	A.bit	2	1	—	$A.bit \leftarrow 0$			
		PSW.bit	3	4	—	$PSW.bit \leftarrow 0$	×	×	×
		laddr16.bit	4	2	—	$(addr16).bit \leftarrow 0$			
		ES:laddr16.bit	5	3	—	$(ES, addr16).bit \leftarrow 0$			
		saddr.bit	3	2	—	$(saddr).bit \leftarrow 0$			
		sfr.bit	3	2	—	$sfr.bit \leftarrow 0$			
		[HL].bit	2	2	—	$(HL).bit \leftarrow 0$			
		ES:[HL].bit	3	3	—	$(ES, HL).bit \leftarrow 0$			
	SET1	CY	2	1	—	$CY \leftarrow 1$			1
	CLR1	CY	2	1	—	$CY \leftarrow 0$			0
	NOT1	CY	2	1	—	$CY \leftarrow \overline{CY}$			×

注 1. 这是存取内部 RAM 区、SFR 区或者扩展 SFR 区时或者不存取数据时的 CPU 时钟 (f_{CLK}) 数。

2. 这是存取闪存区时的 CPU 时钟 (f_{CLK}) 数。

备注 时钟数是在内部 ROM (闪存) 区有程序的情况。当从内部 RAM 区取指令时, 最大为 2 倍 +3 个时钟。

表 31-5 操作一览表 (15/17)

指令集	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
调用 / 返回	CALL	rp	2	3	—	(SP-2) $\leftarrow$ (PC+2) _S , (SP-3) $\leftarrow$ (PC+2) _H , (SP-4) $\leftarrow$ (PC+2) _L , PC $\leftarrow$ CS, rp, SP $\leftarrow$ SP-4			
		\$!addr20	3	3	—	(SP-2) $\leftarrow$ (PC+3) _S , (SP-3) $\leftarrow$ (PC+3) _H , (SP-4) $\leftarrow$ (PC+3) _L , PC $\leftarrow$ PC+3+jdisp16, SP $\leftarrow$ SP-4			
		!addr16	3	3	—	(SP-2) $\leftarrow$ (PC+3) _S , (SP-3) $\leftarrow$ (PC+3) _H , (SP-4) $\leftarrow$ (PC+3) _L , PC $\leftarrow$ 0000, addr16, SP $\leftarrow$ SP-4			
		!!addr20	4	3	—	(SP-2) $\leftarrow$ (PC+4) _S , (SP-3) $\leftarrow$ (PC+4) _H , (SP-4) $\leftarrow$ (PC+4) _L , PC $\leftarrow$ addr20, SP $\leftarrow$ SP-4			
	CALLT	[addr5]	2	5	—	(SP-2) $\leftarrow$ (PC+2) _S , (SP-3) $\leftarrow$ (PC+2) _H , (SP-4) $\leftarrow$ (PC+2) _L , PC _S $\leftarrow$ 0000, PC _H $\leftarrow$ (0000, addr5+1), PC _L $\leftarrow$ (0000, addr5), SP $\leftarrow$ SP-4			
	BRK	—	2	5	—	(SP-1) $\leftarrow$ PSW, (SP-2) $\leftarrow$ (PC+2) _S , (SP-3) $\leftarrow$ (PC+2) _H , (SP-4) $\leftarrow$ (PC+2) _L , PC _S $\leftarrow$ 0000, PC _H $\leftarrow$ (0007FH), PC _L $\leftarrow$ (0007EH), SP $\leftarrow$ SP-4, IE $\leftarrow$ 0			
	RET	—	1	6	—	PC _L $\leftarrow$ (SP), PC _H $\leftarrow$ (SP+1), PC _S $\leftarrow$ (SP+2), SP $\leftarrow$ SP+4			
	RETI	—	2	6	—	PC _L $\leftarrow$ (SP), PC _H $\leftarrow$ (SP+1), PC _S $\leftarrow$ (SP+2), PSW $\leftarrow$ (SP+3), SP $\leftarrow$ SP+4	R	R	R
	RETB	—	2	6	—	PC _L $\leftarrow$ (SP), PC _H $\leftarrow$ (SP+1), PC _S $\leftarrow$ (SP+2), PSW $\leftarrow$ (SP+3), SP $\leftarrow$ SP+4	R	R	R

注 1. 这是存取内部 RAM 区、SFR 区或者扩展 SFR 区时或者不存取数据时的 CPU 时钟 (f_{CLK}) 数。

2. 这是存取闪存区时的 CPU 时钟 (f_{CLK}) 数。

备注 时钟数是在内部 ROM (闪存) 区有程序的情况。当从内部 RAM 区取指令时, 最大为 2 倍 +3 个时钟。

表 31-5 操作一览表 (16/17)

指令集	助记符	操作数	字节	时钟		操作	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)←rp _H , (SP-2)←rp _L , SP←SP-2			
	POP	PSW	2	3	—	PSW←(SP+1), SP←SP+2	R	R	R
		rp	1	1	—	rp _L ←(SP), rp _H ←(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			
无条件转移	BR	AX	2	3	—	PC←CS, AX			
		\$addr20	2	3	—	PC←PC+2+jdisp8			
		\$!addr20	3	3	—	PC←PC+3+jdisp16			
		!addr16	3	3	—	PC←0000, addr16			
		!!addr20	4	3	—	PC←addr20			
条件转移	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+disp8 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. 这是存取内部 RAM 区、SFR 区或者扩展 SFR 区时或者不存取数据时的 CPU 时钟 (f_{CLK}) 数。
2. 这是存取闪存区时的 CPU 时钟 (f_{CLK}) 数。
3. 表示“条件不成立时 / 条件成立时”的时钟数。

备注 时钟数是在内部 ROM (闪存) 区有程序的情况。当从内部 RAM 区取指令时, 最大为 2 倍 +3 个时钟。

表 31-5 操作一览表 (17/17)

指令集	助记符	操作数	字节	时钟		操作	标志		
				注 1	注 2		Z	AC	CY
条件转移	BF	saddr.bit, \$addr20	4	3/5 注3	—	PC←PC+4+jdisp8 if (saddr).bit=0			
		sfr.bit, \$addr20	4	3/5 注3	—	PC←PC+4+jdisp8 if sfr.bit=0			
		A.bit, \$addr20	3	3/5 注3	—	PC←PC+3+jdisp8 if A.bit=0			
		PSW.bit, \$addr20	4	3/5 注3	—	PC←PC+4+jdisp8 if PSW.bit=0			
		[HL].bit, \$addr20	3	3/5 注3	6/7	PC←PC+3+jdisp8 if (HL).bit=0			
		ES:[HL].bit, \$addr20	4	4/6 注3	7/8	PC←PC+4+jdisp8 if (ES, HL).bit=0			
	BTCLR	saddr.bit, \$addr20	4	3/5 注3	—	PC←PC+4+jdisp8 if (saddr).bit=1 then reset (saddr).bit			
		sfr.bit, \$addr20	4	3/5 注3	—	PC←PC+4+jdisp8 if sfr.bit=1 then reset sfr.bit			
		A.bit, \$addr20	3	3/5 注3	—	PC←PC+3+jdisp8 if A.bit=1 then reset A.bit			
		PSW.bit, \$addr20	4	3/5 注3	—	PC←PC+4+jdisp8 if PSW.bit=1 then reset PSW.bit	×	×	×
		[HL].bit, \$addr20	3	3/5 注3	—	PC←PC+3+jdisp8 if (HL).bit=1 then reset (HL).bit			
		ES:[HL].bit, \$addr20	4	4/6 注3	—	PC←PC+4+jdisp8 if (ES, HL).bit=1 then reset (ES, HL).bit			
条件跳转	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			
CPU 控制	SEL 注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. 这是存取内部 RAM 区、SFR 区或者扩展 SFR 区时或者不存取数据时的 CPU 时钟 (f_{CLK}) 数。

2. 这是存取闪存区时的 CPU 时钟 (f_{CLK}) 数。

3. 表示“条件不成立时/条件成立时”的时钟数。

4. n 为寄存器组号 (n=0 ~ 3)。

备注 时钟数是在内部 ROM (闪存) 区有程序的情况。当从内部 RAM 区取指令时, 最大为 2 倍 +3 个时钟。

第 32 章 电特性

注意 RL78 微控制器内置用于开发和评估的片上调试功能。如果使用片上调试功能，就可能会超过闪存的保证改写次数而无法保证产品的可靠性，因此批量生产的产品不能使用片上调试功能。对于使用片上调试功能的产品，不作为投诉受理对象。

32.1 绝对最大额定值

绝对最大额定值（1/3）

项目	符号	条件	额定值	单位
电源电压	V_{DD}		$-0.5 \sim +6.5$	V
REGC 引脚输入电压	V_{IREGC}	REGC	$-0.3 \sim +2.8$ 并且 $-0.3 \sim V_{DD} + 0.3$ 注 1	V
输入电压	V_{I1}	P00 ~ P07、P10 ~ P17、P20 ~ P27、 P30 ~ P35、P40 ~ P47、P50 ~ P57、 P60、P61、P70 ~ P77、P121 ~ P127、 P130、P137	$-0.3 \sim V_{DD} + 0.3$ 注 2	V
	V_{I2}	P60、P61（N 沟道漏极开路）	$-0.3 \sim +6.5$	V
	V_{I3}	EXCLK、EXCLKS、 $\overline{RESET}$	$-0.3 \sim V_{DD} + 0.3$ 注 2	V
输出电压	V_{O1}	P00 ~ P07、P10 ~ P17、P20 ~ P27、 P30 ~ P35、P40 ~ P47、P50 ~ P57、 P60、P61、P70 ~ P77、P121 ~ P127、 P130、P137	$-0.3 \sim V_{DD} + 0.3$ 注 2	V
模拟输入电压	V_{AI1}	ANI0、ANI1、ANI16、ANI17	$-0.3 \sim V_{DD} + 0.3$ 并且 $-0.3 \sim AV_{REF(+)} + 0.3$ 注 2、注 3	V

- 注 1. 必须通过电容器（ $0.47 \sim 1\mu\text{F}$ ）将 REGC 引脚连接 V_{SS} 。此值规定 REGC 引脚的绝对最大额定值，在使用时不能外加电压。
2. 不超过 6.5V。
3. A/D 转换对象的引脚不能超过 $AV_{REF(+)} + 0.3$ 。

注意 即使是各项目中的 1 个项目一瞬间超过绝对最大额定值，也可能降低产品的质量。绝对最大额定值是可能给产品带来物理性损伤的额定值，必须在不超过额定值的状态下使用产品。

备注 1. 在没有特别指定的情况下，复用引脚的特性和端口引脚的特性相同。

2. $AV_{REF(+)}$: A/D 转换器的正（+）基准电压
3. 将 V_{SS} 作为基准电压。

绝对最大额定值 (2/3)

项目	符号	条件		额定值	单位
LCD 电压	V _{L1}	V _{L1} 电压 ^{注 1}		-0.3 ~ +2.8 并且 -0.3 ~ V _{L4} +0.3	V
	V _{L2}	V _{L2} 电压 ^{注 1}		-0.3 ~ V _{L4} +0.3 ^{注 2}	V
	V _{L3}	V _{L3} 电压 ^{注 1}		-0.3 ~ V _{L4} +0.3 ^{注 2}	V
	V _{L4}	V _{L4} 电压 ^{注 1}		-0.3 ~ +6.5	V
	V _{LCAP}	CAPL、CAPH 电压 ^{注 1}		-0.3 ~ V _{L4} +0.3 ^{注 2}	V
	V _{OUT}	COM0 ~ COM7、 SEG0 ~ SEG50、 输出电压	外部电阻分压方式	-0.3 ~ V _{DD} +0.3 ^{注 2}	V
内部升压方式			-0.3 ~ V _{L4} +0.3 ^{注 2}	V	

注 1. 此值规定向 V_{L1} 、 V_{L2} 、 V_{L3} 、 V_{L4} 引脚外加电压时的绝对最大额定值，并不表示推荐使用外加电压。在内部升压方式中，必须通过电容器 ($0.47\mu\text{F} \pm 30\%$) 将 V_{L1} 、 V_{L2} 、 V_{L3} 、 V_{L4} 引脚连接 V_{SS} ，并且在 CAPL 引脚和 CAPH 引脚之间连接电容器 ($0.47\mu\text{F} \pm 30\%$)。

2. 不超过 6.5 V。

注意 即使是各项目中的 1 个项目一瞬间超过绝对最大额定值，也可能降低产品的质量。绝对最大额定值是可能给产品带来物理性损伤的额定值，必须在不超过额定值的状态下使用产品。

备注 将 V_{SS} 作为基准电压。

绝对最大额定值 (3/3)

参数	符号	条件		额定值	单位
高电平输出电流	I _{OH1}	1 个引脚	P00 ~ P07、P10 ~ P17、P22 ~ P27、 P30 ~ P35、P40 ~ P47、P50 ~ P57、 P60、P61、P70 ~ P77、 P125 ~ P127、P130	-40	mA
		引脚合计 -170mA	P00 ~ P07、P14 ~ P17、P30 ~ P35、 P40 ~ P47、P50 ~ P57、P60、P61、 P70 ~ P77、P125 ~ P127、P130	-170	mA
	I _{OH2}	1 个引脚	P20、P21	-0.5	mA
		引脚合计		-2	mA
低电平输出电流	I _{OL1}	1 个引脚	P00 ~ P07、P10 ~ P17、P22 ~ P27、 P30 ~ P35、P40 ~ P47、P50 ~ P57、 P60、P61、P70 ~ P77、 P125 ~ P127、P130	40	mA
		引脚合计 170mA	P40 ~ P47、P130	70	mA
			P00 ~ P07、P14 ~ P17、P30 ~ P35、 P50 ~ P57、P60、P61、P70 ~ P77、 P125 ~ P127	100	mA
	I _{OL2}	1 个引脚	P20、P21	1	mA
		引脚合计		5	mA
工作环境温度	T _A	通常运行时		-40 ~ +85	°C
		闪存编程时			
保存温度	T _{stg}			-65 ~ +150	°C

注意 即使是各项目中的 1 个项目一瞬间超过绝对最大额定值, 也可能降低产品的质量。绝对最大额定值是可能给产品带来物理性损伤的额定值, 必须在不超过额定值的状态下使用产品。

备注 在没有特别指定的情况下, 复用引脚的特性和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

($T_A = -40 \sim +85^\circ\text{C}$ 、 $1.6\text{V} \leq V_{DD} \leq 5.5\text{V}$ 、 $V_{SS} = 0\text{V}$)

参数	谐振器	条件	最小值	典型值	最大值	单位
X1 时钟振荡频率 (f_X) 注	陶瓷谐振器 / 晶体谐振器	$2.7\text{V} \leq V_{DD} \leq 5.5\text{V}$	1.0		20.0	MHz
		$2.4\text{V} \leq V_{DD} < 2.7\text{V}$	1.0		16.0	
		$1.8\text{V} \leq V_{DD} < 2.4\text{V}$	1.0		8.0	
		$1.6\text{V} \leq V_{DD} < 1.8\text{V}$	1.0		4.0	
XT1 时钟振荡 频率 (f_{XT}) 注	晶体谐振器		32	32.768	35	kHz

注 只表示振荡电路的频率容许范围，指令执行时间请参照 AC 特性。

另外，委托谐振器厂商进行在安装电路中的评估，并且在确认振荡特性后再使用。

注意 在解除复位后，通过高速内部振荡器时钟启动 CPU，因此用户必须通过振荡稳定时间计数器的状态寄存器（OSTC）确认 X1 时钟的振荡稳定时间，并且必须充分对所用谐振器进行振荡稳定时间的评估，然后决定 OSTC 寄存器和振荡稳定时间选择寄存器（OSTS）的振荡稳定时间。

备注 在使用 X1、XT1 振荡电路时，请参照“5.4 系统时钟振荡电路”。

32.2.2 内部振荡器特性

($T_A = -40 \sim +85^\circ\text{C}$ 、 $1.6\text{V} \leq V_{DD} \leq 5.5\text{V}$ 、 $V_{SS} = 0\text{V}$)

振荡电路	符号	条件		MIN.	TYP.	MAX.	单位
高速内部振荡器时钟频率 注 1、注 2	f_{IH}			1		24	MHz
高速内部振荡器时钟频率 精度		$-20 \sim +85^\circ\text{C}$	$1.8\text{V} \leq V_{DD} \leq 5.5\text{V}$	-1.0		+1.0	%
			$1.6\text{V} \leq V_{DD} < 1.8\text{V}$	-5.0		+5.0	%
		$-40 \sim -20^\circ\text{C}$	$1.8\text{V} \leq V_{DD} \leq 5.5\text{V}$	-1.5		+1.5	%
			$1.6\text{V} \leq V_{DD} < 1.8\text{V}$	-5.5		+5.5	%
低速内部振荡器时钟频率	f_{IL}				15		kHz
低速内部振荡器时钟频率 精度				-15		+15	%

注 1. 通过选项字节（000C2H/010C2H）的 bit0 ~ 4 和 HOCODIV 寄存器的 bit0 ~ 2 选择高速内部振荡器的频率。

2. 只表示振荡电路的特性，指令执行时间请参照 AC 特性。

32.3 DC 特性

32.3.1 引脚特性

(T_A=−40 ~ +85°C、1.6V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件	MIN.	TYP.	MAX.	单位
高电平输出电流 注 1	I _{OH1}	P00 ~ P07、P10 ~ P17、 P22 ~ P27、P30 ~ P35、 P40 ~ P47、P50 ~ P57、 P70 ~ P77、P125 ~ P127、 P130 1 个引脚	1.6V ≤ V _{DD} ≤ 5.5V		−10.0 注 2	mA
		P00 ~ P07、P14 ~ P17、 P30 ~ P35、P40 ~ P47、 P50 ~ P57、P70 ~ P77、 P125 ~ P127、P130 合计 (占空比 =70% 时注 3)	4.0V ≤ V _{DD} ≤ 5.5V		−90.0	mA
			2.7V ≤ V _{DD} < 4.0V		−15.0	mA
			1.8V ≤ V _{DD} < 2.7V		−7.0	mA
			1.6V ≤ V _{DD} < 1.8V		−3.0	mA
	I _{OH2}	P20、P21 1 个引脚	1.6V ≤ V _{DD} ≤ 5.5V		−0.1 注 2	mA
		全部引脚合计 (占空比 =70% 时注 3)	1.6V ≤ V _{DD} ≤ 5.5V		−0.2	mA

注 1. 这是即使电流从 V_{DD} 引脚流到输出引脚也保证器件运行的电流值。

2. 不能超过合计的电流值。

3. 这是在占空比 ≤ 70% 条件下的输出电流值。

更改为占空比 > 70% 时的输出电流值能用以下的计算式进行计算 (将占空比更改为 n% 的情况)。

$$\bullet \text{ 引脚合计的输出电流} = (I_{OH} \times 0.7) / (n \times 0.01)$$

$$< \text{计算例子} > I_{OH} = -90.0\text{mA}、n = 80\%$$

$$\text{引脚合计的输出电流} = (-90.0 \times 0.7) / (80 \times 0.01) \approx -78.75\text{mA}$$

各引脚的电流不会因占空比而变，而且不会流过绝对最大额定值以上的电流。

注意 在 N 沟道漏极开路模式中，P00、P04 ~ P07、P16、P17、P35、P42 ~ P44、P46、P47、P53 ~ P56、P130 不输出高电平。

备注 在没有特别指定的情况下，复用引脚的特性和端口引脚的特性相同。

(T_A=−40 ~ +85°C、1.6V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件	MIN.	TYP.	MAX.	单位
低电平输出电流 注 1	I _{OL1}	P00 ~ P07、P10 ~ P17、 P22 ~ P27、P30 ~ P35、 P40 ~ P47、P50 ~ P57、 P70 ~ P77、P125 ~ P127、 P130 1 个引脚			20.0 注 2	mA
		P60、P61 1 个引脚			15.0 注 2	mA
		P40 ~ P47、P130 合计 (占空比 =70% 时注 3)	4.0V ≤ V _{DD} ≤ 5.5V		70.0	mA
			2.7V ≤ V _{DD} < 4.0V		15.0	mA
			1.8V ≤ V _{DD} < 2.7V		9.0	mA
			1.6V ≤ V _{DD} < 1.8V		4.5	mA
		P00 ~ P07、P14 ~ P17、 P30 ~ P35、P50 ~ P57、 P70 ~ P77、P125 ~ P127 合计 (占空比 =70% 时注 3)	4.0V ≤ V _{DD} ≤ 5.5V		90.0	mA
			2.7V ≤ V _{DD} < 4.0V		35.0	mA
			1.8V ≤ V _{DD} < 2.7V		20.0	mA
			1.6V ≤ V _{DD} < 1.8V		10.0	mA
		全部引脚合计 (占空比 =70% 时注 3)			160.0	mA
	I _{OL2}	P20、P21 1 个引脚			0.4 注 2	mA
		全部引脚合计 (占空比 =70% 时注 3)	1.6V ≤ V _{DD} ≤ 5.5V		0.8	mA

注 1. 这是电流即使从输出引脚流到 V_{SS} 引脚也保证器件运行的电流值。

2. 不能超过合计的电流值。

3. 这是在占空比 ≤ 70% 条件下的输出电流值。

更改为占空比 > 70% 时的输出电流值能用以下的计算式进行计算（将占空比更改为 n% 的情况）。

• 引脚合计的输出电流 = (I_{OL} × 0.7) / (n × 0.01)

<计算例子> I_{OL} = 70.0mA、n = 80%

引脚合计的输出电流 = (70.0 × 0.7) / (80 × 0.01) ≈ 61.25mA

各引脚的电流不会因占空比而变，而且不会流过绝对最大额定值以上的电流。

备注 在没有特别指定的情况下，复用引脚的特性和端口引脚的特性相同。

(T_A=−40 ~ +85°C、1.6V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件	MIN.	TYP.	MAX.	单位
高电平输入电压	V _{IH1}	P00 ~ P07、P10 ~ P17、 P22 ~ P27、P30 ~ P35、 P40 ~ P47、P50 ~ P57、 P70 ~ P77、P125 ~ P127、 P130、P137	通常输入缓冲器 0.8V _{DD}		V _{DD}	V
	V _{IH2}	P03、P05、P06、P16、 P17、P34、P43、P44、 P46、P47、P53、P55	TTL 输入缓冲器 4.0V ≤ V _{DD} ≤ 5.5V	2.2	V _{DD}	V
			TTL 输入缓冲器 3.3V ≤ V _{DD} ≤ 4.0V	2.0	V _{DD}	V
			TTL 输入缓冲器 1.6V ≤ V _{DD} ≤ 3.3V	1.5	V _{DD}	V
	V _{IH3}	P20、P21	0.7V _{DD}		V _{DD}	V
	V _{IH4}	P60、P61	0.7V _{DD}		6.0	V
	V _{IH5}	P121 ~ P124、P137、EXCLK、EXCLKS、RESET	0.8V _{DD}		V _{DD}	V
低电平输入电压	V _{IL1}	P00 ~ P07、P10 ~ P17、 P22 ~ P27、P30 ~ P35、 P40 ~ P47、P50 ~ P57、 P70 ~ P77、P125 ~ P127、 P130、P137	通常输入缓冲器 0		0.2V _{DD}	V
	V _{IL2}	P03、P05、P06、P16、 P17、P34、P43、P44、 P46、P47、P53、P55	TTL 输入缓冲器 4.0 V ≤ V _{DD} ≤ 5.5V	0	0.8	V
			TTL 输入缓冲器 3.3 V ≤ V _{DD} < 4.0V	0	0.5	V
			TTL 输入缓冲器 1.6V ≤ V _{DD} < 3.3V	0	0.32	V
	V _{IL3}	P20、P21	0		0.3V _{DD}	V
	V _{IL4}	P60、P61	0		0.3V _{DD}	V
	V _{IL5}	P121 ~ P124、P137、EXCLK、EXCLKS、RESET	0		0.2V _{DD}	V

注意 即使在 N 沟道漏极开路模式中，P00、P04 ~ P07、P16、P17、P35、P42 ~ P44、P46、P47、P53 ~ P56、P130 的 V_{IH} 最大值 (MAX.) 也为 V_{DD}。

备注 在没有特别指定的情况下，复用引脚的特性和端口引脚的特性相同。

(T_A=−40 ~ +85°C、1.6V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件	MIN.	TYP.	MAX.	单位
高电平输出电压	V _{OH1}	P00 ~ P07、P10 ~ P17、 P22 ~ P27、P30 ~ P35、 P40 ~ P47、P50 ~ P57、 P70 ~ P77、P125 ~ P127、 P130	4.0V ≤ V _{DD} ≤ 5.5V、 I _{OH1} =−10.0mA	V _{DD} −1.5		V
			4.0V ≤ V _{DD} ≤ 5.5V、 I _{OH1} =−3.0mA	V _{DD} −0.7		V
			2.7V ≤ V _{DD} ≤ 5.5V、 I _{OH1} =−2.0mA	V _{DD} −0.6		V
			1.8V ≤ V _{DD} ≤ 5.5V、 I _{OH1} =−1.5mA	V _{DD} −0.5		V
			1.6V ≤ V _{DD} ≤ 5.5V、 I _{OH1} =−1.0mA	V _{DD} −0.5		V
	V _{OH2}	P20、P21	1.6V ≤ V _{DD} ≤ 5.5V、 I _{OH2} =−100μA	V _{DD} −0.5		V
低电平输出电压	V _{OL1}	P00 ~ P07、P10 ~ P17、 P22 ~ P27、P30 ~ P35、 P40 ~ P47、P50 ~ P57、 P70 ~ P77、P125 ~ P127、 P130	4.0V ≤ V _{DD} ≤ 5.5V、 I _{OL1} =20.0mA		1.3	V
			4.0V ≤ V _{DD} ≤ 5.5V、 I _{OL1} =8.5mA		0.7	V
			2.7V ≤ V _{DD} ≤ 5.5V、 I _{OL1} =3.0mA		0.6	V
			2.7V ≤ V _{DD} ≤ 5.5V、 I _{OL1} =1.5mA		0.4	V
			1.8V ≤ V _{DD} ≤ 5.5V、 I _{OL1} =0.6mA		0.4	V
			1.6V ≤ V _{DD} < 1.8V、 I _{OL1} =0.3mA		0.4	V
	V _{OL2}	P20 和 P21	1.6V ≤ V _{DD} ≤ 5.5V、 I _{OL2} =400μA		0.4	V
	V _{OL3}	P60 和 P61	4.0V ≤ V _{DD} ≤ 5.5V、 I _{OL3} =15.0mA		2.0	V
			4.0V ≤ V _{DD} ≤ 5.5V、 I _{OL3} =5.0mA		0.4	V
			2.7V ≤ V _{DD} ≤ 5.5V、 I _{OL3} =3.0mA		0.4	V
			1.8V ≤ V _{DD} ≤ 5.5V、 I _{OL3} =2.0mA		0.4	V
			1.6V ≤ V _{DD} < 1.8V、 I _{OL3} =1.0mA		0.4	V

注意 在 N 沟道漏极开路模式中，P00、P04 ~ P07、P16、P17、P35、P42 ~ P44、P46、P47、P53 ~ P56、P130 不输出高电平。

备注 在没有特别指定的情况下，复用引脚的特性和端口引脚的特性相同。

(T_A=−40 ~ +85°C、1.6V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件		MIN.	TYP.	MAX.	单位	
高电平输入漏电流	I _{LIH1}	P00 ~ P07、P10 ~ P17、 P22 ~ P27、P30 ~ P35、 P40 ~ P47、P50 ~ P57、 P70 ~ P77、P125 ~ P127、 P130、P137	V _I =V _{DD}				1	μA
	I _{LIH2}	P20、P21、 $\overline{\text{RESET}}$	V _I =V _{DD}				1	μA
	I _{LIH3}	P121 ~ P124 (X1、X2、XT1、XT2、 EXCLK、EXCLKS)	V _I =V _{DD}	输入端口时 外部时钟输入时			1	μA
				连接谐振器时			10	μA
低电平输入漏电流	I _{LIL1}	P00 ~ P07、P10 ~ P17、 P22 ~ P27、P30 ~ P35、 P40 ~ P47、P50 ~ P57、 P70 ~ P77、P125 ~ P127、 P130、P137	V _I =V _{SS}				-1	μA
	I _{LIL2}	P20、P21、 $\overline{\text{RESET}}$	V _I =V _{SS}				-1	μA
	I _{LIL3}	P121 ~ P124 (X1、X2、XT1、XT2、 EXCLK、EXCLKS)	V _I =V _{SS}	输入端口时 外部时钟输入时			-1	μA
				连接谐振器时			-10	μA
内部上拉电阻	R _{U1}	P00 ~ P07、P10 ~ P17、 P22 ~ P27、P30 ~ P35、 P45 ~ P47、P50 ~ P57、 P70 ~ P77、P125 ~ P127、 P130	V _I =V _{SS}	2.4V ≤ V _{DD} < 5.5V	10	20	100	kΩ
				1.6V ≤ V _{DD} < 2.4V	10	30	100	kΩ
	R _{U2}	P40 ~ P44	V _I =V _{SS}		10	20	100	kΩ

备注 在没有特别指定的情况下，复用引脚的特性和端口引脚的特性相同。

32.3.2 电源电流特性

(T_A=−40 ~ +85°C、1.6V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V) (1/2)

项目	符号	条件				MIN.	TYP.	MAX.	单位
电源电流 注 1	I _{DD1}	运行 模式	HS (高速主) 模式 注 5	f _{IH} =24MHz 注 3	基本运行	V _{DD} =5.0V		2.0	mA
						V _{DD} =3.0V		2.0	
					通常运行	V _{DD} =5.0V		3.8	mA
						V _{DD} =3.0V		3.8	
				f _{IH} =24MHz 注 3	基本运行	V _{DD} =5.0V		1.7	mA
						V _{DD} =3.0V		1.7	
					通常运行	V _{DD} =5.0V		3.6	mA
						V _{DD} =3.0V		3.6	
				f _{IH} =16MHz 注 3	通常运行	V _{DD} =5.0V		2.7	mA
						V _{DD} =3.0V		2.7	
			LS (低速主) 模式 注 5	f _{IH} =8MHz 注 3	通常运行	V _{DD} =3.0V		1.2	mA
						V _{DD} =2.0V		1.2	
			LV (低电压主) 模式 注 5	f _{IH} =4MHz 注 3	通常运行	V _{DD} =3.0V		1.2	mA
						V _{DD} =2.0V		1.2	
			HS (高速主) 模式 注 5	f _{MX} =20MHz 注 2、 V _{DD} =5.0 V	通常运行	方波输入		3.0	mA
						连接谐振器		3.2	
				f _{MX} =20MHz 注 2、 V _{DD} =3.0 V	通常运行	方波输入		2.9	mA
						连接谐振器		3.2	
				f _{MX} =16MHz 注 2、 V _{DD} =5.0 V	通常运行	方波输入		2.5	mA
						连接谐振器		2.7	
				f _{MX} =16MHz 注 2、 V _{DD} =3.0 V	通常运行	方波输入		2.5	mA
						连接谐振器		2.7	
				f _{MX} =10MHz 注 2、 V _{DD} =5.0 V	通常运行	方波输入		1.9	mA
						连接谐振器		1.9	
				f _{MX} =10MHz 注 2、 V _{DD} =3.0 V	通常运行	方波输入		1.9	mA
						连接谐振器		1.9	
			LS (低速主) 模式 注 5	f _{MX} =8MHz 注 2、 V _{DD} =3.0 V	通常运行	方波输入		1.1	mA
						连接谐振器		1.1	
				f _{MX} =8MHz 注 2、 V _{DD} =2.0 V	通常运行	方波输入		1.1	mA
						连接谐振器		1.1	
			副系统时钟运行	f _{SUB} =32.768kHz 注 4、 T _A =−40°C	通常运行	方波输入		4.0	μA
						连接谐振器		4.3	
				f _{SUB} =32.768kHz 注 4、 T _A =+25°C	通常运行	方波输入		4.0	μA
						连接谐振器		4.3	
				f _{SUB} =32.768kHz 注 4、 T _A =+50°C	通常运行	方波输入		4.1	μA
						连接谐振器		4.4	
				f _{SUB} =32.768kHz 注 4、 T _A =+70°C	通常运行	方波输入		4.3	μA
						连接谐振器		4.7	
				f _{SUB} =32.768kHz 注 4、 T _A =+85°C	通常运行	方波输入		4.7	μA
						连接谐振器		5.2	

- 注 1. 这是流过 V_{DD} 的总电流，包含输入引脚固定为 V_{DD} 或者 V_{SS} 状态的输入漏电流。MAX. 值包含外围工作电流。但是，不包含流到 LCD 控制器 / 驱动器、A/D 转换器、LVD 电路、比较器、I/O 端口、内部上拉或者下拉电阻的电流。
2. 这是高速内部振荡器和副系统时钟停止振荡的情况。
3. 这是高速系统时钟和副系统时钟停止振荡的情况。
4. 这是高速内部振荡器和高速系统时钟停止振荡的情况，或者是设定为超低功耗振荡（AMPHS1=1）的情况。不包含流到 LCD 控制器 / 驱动器、实时时钟 2、12 位间隔定时器、看门狗定时器的电流。
5. 工作电压范围、CPU 工作频率和运行模式的关系如下所示：
- HS（高速主）模式： $2.7V \leq V_{DD} \leq 5.5V @ 1MHz \sim 24MHz$
 $2.4V \leq V_{DD} \leq 5.5V @ 1MHz \sim 16MHz$
- LS（低速主）模式： $1.8V \leq V_{DD} \leq 5.5V @ 1MHz \sim 8MHz$
- LV（低电压主）模式： $1.6V \leq V_{DD} \leq 5.5V @ 1MHz \sim 4MHz$

- 备注 1. f_{MX} : 高速系统时钟频率（X1 时钟振荡频率或者外部主系统时钟频率）
2. f_{IH} : 高速内部振荡器的时钟频率
3. f_{SUB} : 副系统时钟频率（XT1 时钟振荡频率）
4. “副系统时钟运行”以外的 TYP. 值的温度条件是 $T_A=25^\circ C$ 。

(T_A=−40 ~ +85°C、1.6 V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V) (2/2)

项目	符号	条件				MIN.	TYP.	MAX.	单位	
电源电流 注 1	I _{DD2} 注 2	HALT 模式	HS（高速主） 模式 注 7	f _{IH} =24MHz 注 4	V _{DD} =5.0V		0.71	1.95	mA	
					V _{DD} =3.0V		0.71	1.95		
				f _{IH} =24MHz 注 4	V _{DD} =5.0V		0.49	1.64	mA	
					V _{DD} =3.0V		0.49	1.64		
				f _{IH} =16MHz 注 4	V _{DD} =5.0V		0.43	1.11	mA	
					V _{DD} =3.0V		0.43	1.11		
			LS（低速主） 模式 注 7	f _{IH} =8MHz 注 4	V _{DD} =3.0V		280	770	mA	
					V _{DD} =2.0V		280	770		
			LV（低电压主） 模式 注 7	f _{IH} =4MHz 注 4	V _{DD} =3.0V		430	700	mA	
					V _{DD} =2.0V		430	700		
			HS（高速主） 模式 注 7	f _{MX} =20MHz 注 3、 V _{DD} =5.0V	方波输入		0.31	1.42	mA	
					连接谐振器		0.48	1.42		
				f _{MX} =20MHz 注 3、 V _{DD} =3.0V	方波输入		0.29	1.42	mA	
					连接谐振器		0.48	1.42		
				f _{MX} =16MHz 注 3、 V _{DD} =5.0V	方波输入		0.26	0.86	mA	
					连接谐振器		0.45	1.15		
				f _{MX} =16MHz 注 3、 V _{DD} =3.0V	方波输入		0.25	0.86	mA	
					连接谐振器		0.44	1.15		
				f _{MX} =10MHz 注 3、 V _{DD} =5.0V	方波输入		0.20	0.63	mA	
					连接谐振器		0.28	0.71		
				f _{MX} =10MHz 注 3、 V _{DD} =3.0V	方波输入		0.19	0.63	mA	
					连接谐振器		0.28	0.71		
			LS（低速主） 模式 注 7	f _{MX} =8MHz 注 3、 V _{DD} =3.0V	方波输入		100	560	mA	
					连接谐振器		160	560		
				f _{MX} =8MHz 注 3、 V _{DD} = 2.0V	方波输入		100	560	mA	
					连接谐振器		160	560		
			副系统时钟运行	f _{SUB} =32.768kHz 注 5、 T _A =−40°C	方波输入		0.34	0.62	μA	
					连接谐振器		0.51	0.80		
				f _{SUB} =32.768kHz 注 5、 T _A =+25°C	方波输入		0.38	0.62	μA	
					连接谐振器		0.57	0.80		
				f _{SUB} =32.768kHz 注 5、 T _A =+50°C	方波输入		0.46	2.30	μA	
					连接谐振器		0.67	2.49		
				f _{SUB} =32.768kHz 注 5、 T _A =+70°C	方波输入		0.65	4.03	μA	
					连接谐振器		0.91	4.22		
				f _{SUB} =32.768kHz 注 5、 T _A =+85°C	方波输入		1.00	8.04	μA	
					连接谐振器		1.31	8.23		
	I _{DD3} 注 6	STOP 模式注 8	T _A =−40°C					0.18	0.52	μA
			T _A =+25°C					0.24	0.52	
			T _A =+50°C					0.33	2.21	
			T _A =+70°C					0.53	3.94	
			T _A =+85°C					0.93	7.95	

- 注 1. 这是流过 V_{DD} 的总电流，包含输入引脚固定为 V_{DD} 或者 V_{SS} 状态的输入漏电流。MAX. 值包含外围工作电流。但是，不包含流到 LCD 控制器 / 驱动器、A/D 转换器、LVD 电路、比较器、I/O 端口、内部上拉或者下拉电阻的电流。
2. 这是执行闪存的 HALT 指令的情况。
3. 这是高速内部振荡器和副系统时钟停止振荡的情况。
4. 这是高速系统时钟和副系统时钟停止振荡的情况。
5. 这是高速内部振荡器和高速系统时钟停止振荡的情况，或者是 $RTCLPC=1$ 并且设定为超低功耗振荡 ($AMPHS1=1$) 的情况。包含流到实时时钟 2 的电流。但是，不包含流到时钟输出 / 蜂鸣器输出、12 位间隔定时器和看门狗定时器的电流。
6. 不包含流到实时时钟 2、时钟输出 / 蜂鸣器输出、12 位间隔定时器、看门狗定时器的电流。
7. 工作电压范围、CPU 工作频率和运行模式的关系如下所示：
- HS (高速主) 模式: $2.7V \leq V_{DD} \leq 5.5V@1MHz \sim 24MHz$
 $2.4V \leq V_{DD} \leq 5.5V@1MHz \sim 16MHz$
- LS (低速主) 模式: $1.8V \leq V_{DD} \leq 5.5V@1MHz \sim 8MHz$
- LV (低电压主) 模式: $1.6V \leq V_{DD} \leq 5.5V@1MHz \sim 4MHz$
8. STOP 模式中副系统时钟运行时的电流值，请参照 HALT 模式中副系统时钟运行时的电流值。

- 备注 1. f_{MX} : 高速系统时钟频率 (X1 时钟振荡频率或者外部主系统时钟频率)
2. f_{IH} : 高速内部振荡器的时钟频率
3. f_{SUB} : 副系统时钟频率 (XT1 时钟振荡频率)
4. “副系统时钟运行”和“STOP 模式”以外的 TYP. 值的温度条件是 $T_A=25^\circ C$ 。

(T_A=−40 ~ +85°C、1.6V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件				MIN.	TYP.	MAX.	单位
低速内部振荡器工作电流	I _{FIL} 注 1						0.20		μA
RTC2 工作电流	I _{RTC} 注 1、 2、 3	f _{SUB} =32.768kHz					0.02		μA
12 位间隔定时器工作电流	I _{TMKA} 注 1、 2、 4						0.04		μA
看门狗定时器工作电流	I _{WDT} 注 1、 2、 5	f _{IL} =15kHz					0.22		μA
A/D 转换器工作电流	I _{ADC} 注 1、 6	最高速转换时	标准模式、 AV _{REFP} =V _{DD} =5.0V				1.3	1.7	mA
			低电压模式、 AV _{REFP} =V _{DD} =3.0V				0.5	0.7	mA
A/D 转换器基准电压电流	I _{ADREF} 注 1						75.0		μA
温度传感器工作电流	I _{TMPS} 注 1						75.0		μA
LVD 工作电流	I _{LVD} 注 1、 7						0.08		μA
比较器工作电流	I _{CMP} 注 1、 10	V _{DD} =5.0V、 稳压器输出电压=2.1V	窗口模式				12.5		μA
			比较器高速模式				6.5		
			比较器低速模式				1.7		
		V _{DD} =5.0V、 稳压器输出电压=1.8V	窗口模式				8.0		μA
			比较器高速模式				4.0		
			比较器低速模式				1.3		
自编程工作电流	I _{FSP} 注 1、 8						2.00	12.20	mA
SNOOZE 工作电流	I _{SNOZ} 注 1	ADC 运行	正在模式转移注 9				0.50	0.60	mA
			正在转换、低电压模式、 AV _{REFP} =V _{DD} =3.0 V				1.20	1.44	
		CSI/UART 操作					0.70	0.84	mA
LCD 工作电流	I _{LCD1} 注 1、 11、 12	外部电阻分压方式	f _{LCD} =f _{SUB} LCD 时钟=128Hz	1/3 偏压 4 个时间片	V _{DD} =5.0V、 V _{L4} =5.0V		0.04	0.20	μA
	I _{LCD2} 注 1、 11	内部升压方式	f _{LCD} =f _{SUB} LCD 时钟=128Hz	1/3 偏压 4 个时间片	V _{DD} =3.0V、 V _{L4} =3.0V (VLCD=04H)		0.85	2.20	μA
					V _{DD} =5.0V、 V _{L4} =5.1V (VLCD=12H)		1.55	3.70	

- 注 1. 这是流过 V_{DD} 的电流。
2. 这是高速内部振荡器和高速系统时钟停止振荡的情况。
3. 这是只流到实时时钟 2 的电流（不包含低速内部振荡器、XT1 振荡电路的工作电流）。在运行模式或者 HALT 模式中实时时钟 2 运行的情况下，RL78 微控制器的电流值为 I_{DD1} 或者 I_{DD2} 加上 I_{RTC} 的值。另外，在选择低速内部振荡器时，必须加上 I_{FIL} 。副系统时钟运行时的 I_{DD2} 包含实时时钟 2 的工作电流。
4. 这是只流到 12 位间隔定时器的电流（不包含低速内部振荡器、XT1 振荡电路的工作电流）。在运行模式或者 HALT 模式中 12 位间隔定时器运行的情况下，RL78 微控制器的电流值为 I_{DD1} 或者 I_{DD2} 加上 I_{TMKA} 的值。另外，在选择低速内部振荡器时，必须加上 I_{FIL} 。
5. 这是只流到看门狗定时器的电流（包含低速内部振荡器的工作电流）。在看门狗定时器运行的情况下，RL78 微控制器的电流值为 I_{DD1} 、 I_{DD2} 或者 I_{DD3} 加上 I_{WDT} 的值。
6. 这是只流到 A/D 转换器的电流。在运行模式或者 HALT 模式中 A/D 转换器运行的情况下，RL78 微控制器的电流值为 I_{DD1} 或者 I_{DD2} 加上 I_{ADC} 的值。
7. 这是只流到 LVD 电路的电流。在 LVD 电路运行的情况下，RL78 微控制器的电流值为 I_{DD1} 、 I_{DD2} 或者 I_{DD3} 加上 I_{LVD} 的值。
8. 这是自编程时流过的电流。
9. 有关 SNOOZE 模式的转移时间，请参照“21.3.3 SNOOZE 模式”。
10. 这是只流到比较器电路的电流。在比较器电路运行的情况下，RL78 微控制器的电流值为 I_{DD1} 、 I_{DD2} 或者 I_{DD3} 加上 I_{CMP} 的值。
11. 这是只流到 LCD 控制器 / 驱动器的电流。在运行模式或者 HALT 模式中 LCD 控制器 / 驱动器运行的情况下，RL78 微控制器的电流值为电源电流（ I_{DD1} 或者 I_{DD2} ）加上 LCD 工作电流（ I_{LCD1} 、 I_{LCD2} 或者 I_{LCD3} ）的值。不包含流到 LCD 显示屏的电流。TYP. 值和 MAX. 值变为以下条件下的值。
- 将 20 引脚设定为段功能并且全部亮灯
 - 选择 f_{SUB} 为系统时钟并且 LCD 时钟 = 128Hz 时（LCDC0=07H）
 - 设定为 4 个时间片和 1/3 偏压
12. 不包含流到使用外部电阻分压方式时的外部电阻的电流。

- 备注 1. f_{IL} : 低速内部振荡器时钟频率
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. TYP. 值的温度条件是 $T_A=25^{\circ}\text{C}$ 。

32.4 AC 特性

(T_A=−40 ~ +85°C、1.6V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

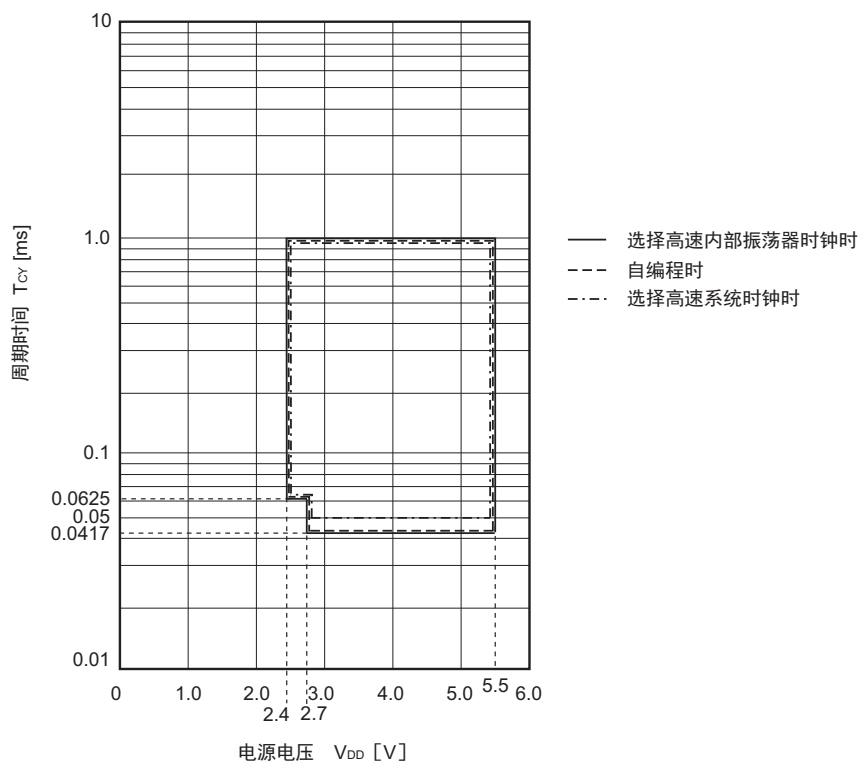
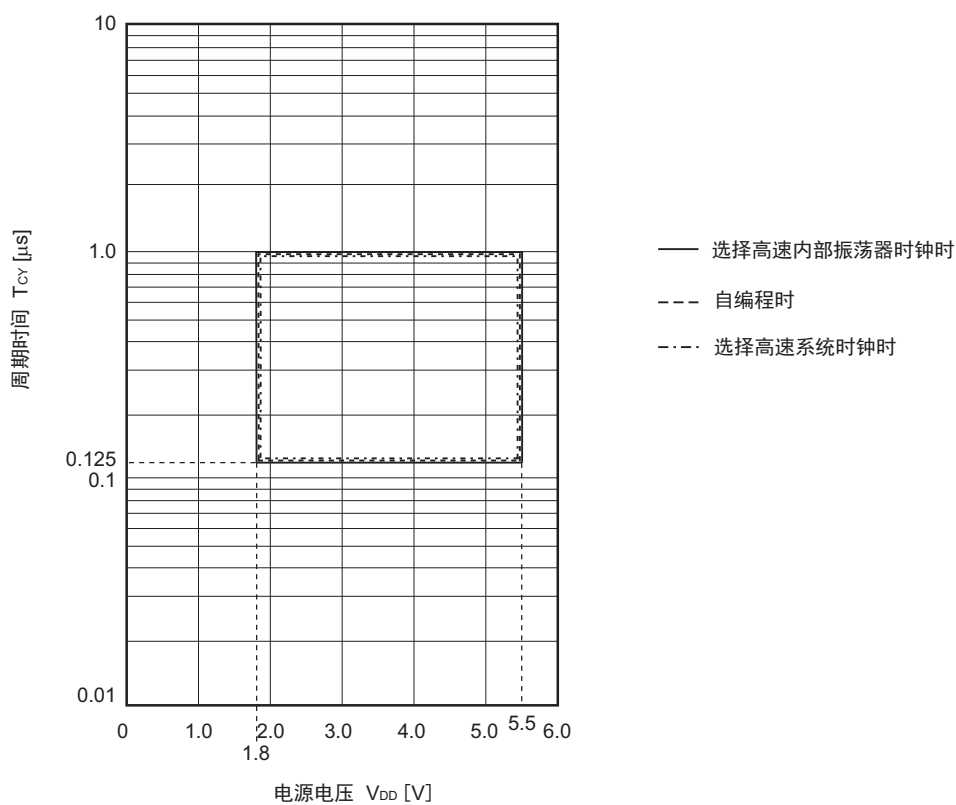
项目	符号	条件		MIN.	TYP.	MAX.	单位
指令周期 (最短指令执行 时间)	T _{CY}	主系统时钟 (f _{MAIN}) 运行	HS (高速主) 模式	2.7V ≤ V _{DD} ≤ 5.5V	0.0417		1 μs
				2.4V ≤ V _{DD} < 2.7V	0.0625		1 μs
			LS (低速主) 模式	1.8V ≤ V _{DD} ≤ 5.5V	0.125		1 μs
			LV (低电压主) 模式	1.6V ≤ V _{DD} ≤ 5.5V	0.25		1 μs
		副系统时钟 (f _{SUB}) 运行注		1.8V ≤ V _{DD} ≤ 5.5V	28.5	30.5	31.3 μs
		自编程模式	HS (高速主) 模式	2.7V ≤ V _{DD} ≤ 5.5V	0.0417		1 μs
				2.4V ≤ V _{DD} < 2.7V	0.0625		1 μs
			LS (低速主) 模式	1.8V ≤ V _{DD} ≤ 5.5V	0.125		1 μs
			LV (低电压主) 模式	1.8V ≤ V _{DD} ≤ 5.5V	0.25		1 μs
外部系统时钟频率	f _{EX}	2.7V ≤ V _{DD} ≤ 5.5V		1.0		20.0	MHz
		2.4V ≤ V _{DD} < 2.7V		1.0		16.0	MHz
		1.8V ≤ V _{DD} < 2.4V		1.0		8.0	MHz
		1.6V ≤ V _{DD} < 1.8V		1.0		4.0	MHz
	f _{EXT}			32		35	kHz
外部系统时钟输入 高电平 / 低电平宽度	t _{EXH} 、 t _{EXL}	2.7V ≤ V _{DD} ≤ 5.5V		24			ns
		2.4V ≤ V _{DD} < 2.7V		30			ns
		1.8V ≤ V _{DD} < 2.4V		60			ns
		1.6V ≤ V _{DD} < 1.8V		120			ns
	t _{EXHS} 、 t _{EXLS}			13.7			μs
TI00 ~ TI07 输入 高电平 / 低电平宽度	t _{TIH} 、 t _{TIL}			1/f _{MCK} +10			ns
TO00 ~ TO07 输出 频率	f _{TO}	HS (高速主) 模式	4.0V ≤ V _{DD} ≤ 5.5V			12	MHz
			2.7V ≤ V _{DD} < 4.0V			8	MHz
			1.8V ≤ V _{DD} < 2.7V			4	MHz
		LV (低电压主) 模式	1.6V ≤ V _{DD} ≤ 5.5V			2	MHz
		LS (低速主) 模式	1.8V ≤ V _{DD} ≤ 5.5V			4	MHz
PCLBUZ0、 PCLBUZ1 输出频率	f _{PCL}	HS (高速主) 模式	4.0V ≤ V _{DD} ≤ 5.5V			16	MHz
			2.7V ≤ V _{DD} < 4.0V			8	MHz
			1.8V ≤ V _{DD} < 2.7V			4	MHz
		LV (低电压主) 模式	1.8V ≤ V _{DD} ≤ 5.5V			4	MHz
			1.6V ≤ V _{DD} < 1.8V			2	MHz
		LS (低速主) 模式	1.8V ≤ V _{DD} ≤ 5.5V			4	MHz
中断输入高电平 / 低电平宽度	t _{INTH} 、 t _{INTL}	INTP0 ~ INTP7	1.6V ≤ V _{DD} ≤ 5.5V	1			μs
RESET 低电平宽度	t _{RSL}			10			μs

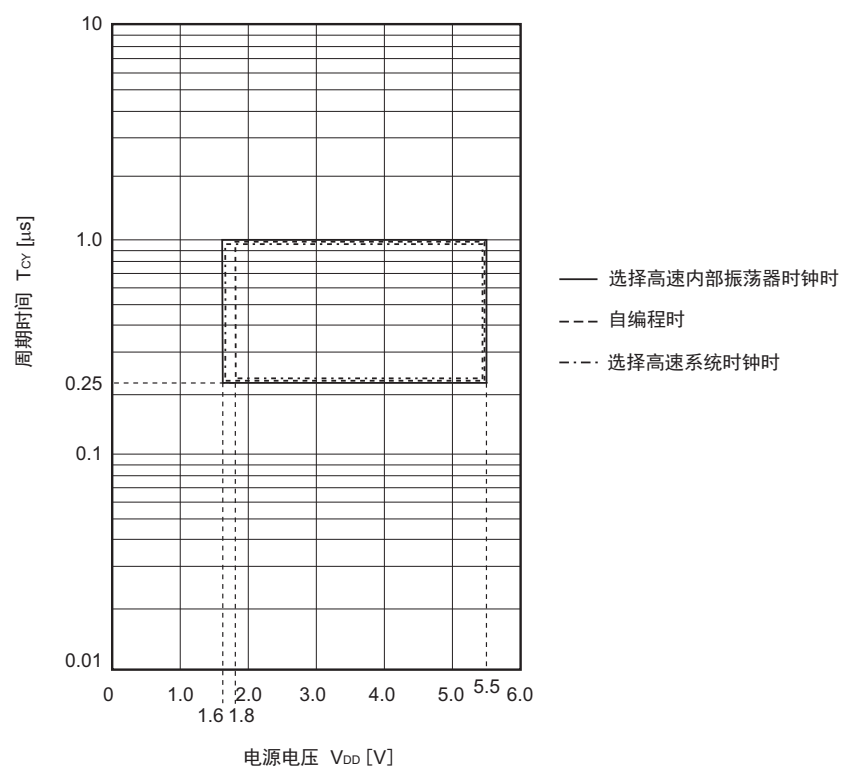
注 在 LV (低电压主) 模式中的 1.6V ≤ V_{DD} < 1.8V 时, 副系统时钟不可运行。备注 f_{MCK}: 定时器阵列单元的运行时钟频率

(这是定时器模式寄存器 mn (TMRmn) 的 CKSMn0 位和 CKSMn1 位设定的运行时钟。m: 单元号 (m=0)

n: 通道号 (n=0 ~ 7))

主系统时钟运行时的最短指令执行时间

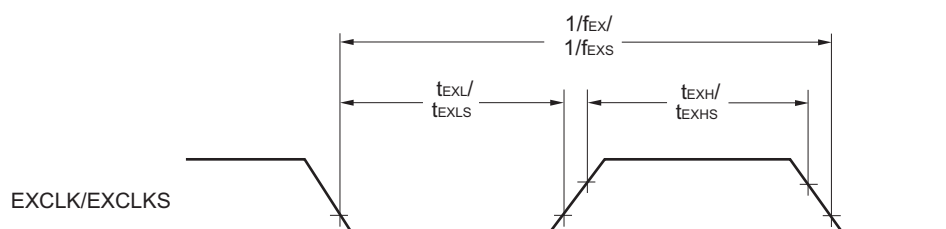
 T_{CY} vs V_{DD} (HS (高速主) 模式) T_{CY} vs V_{DD} (LS (低速主) 模式)

T_{CY} vs V_{DD} (LV (低电压主) 模式)

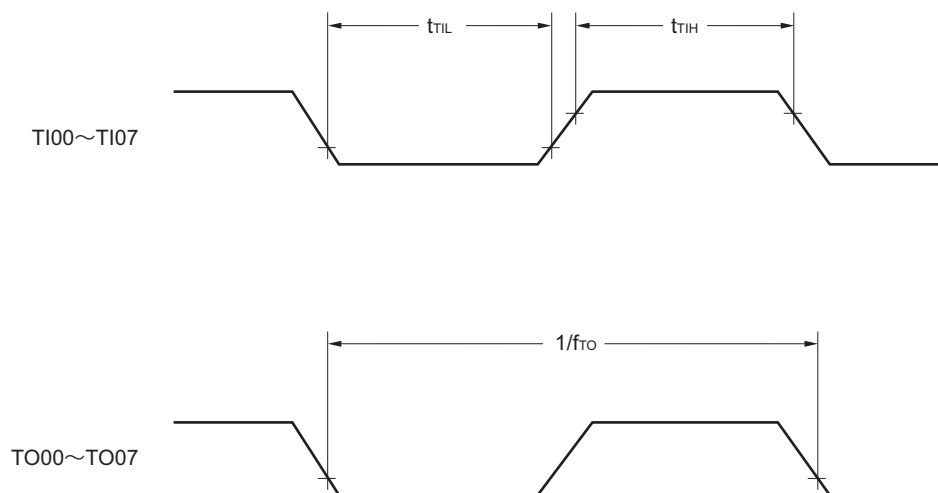
AC 时序测量点



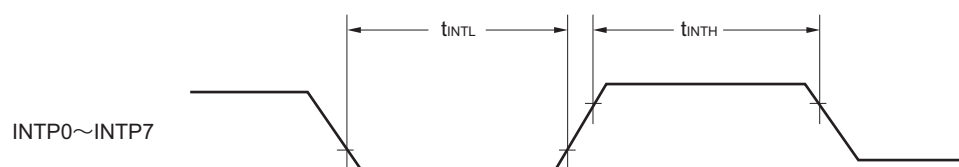
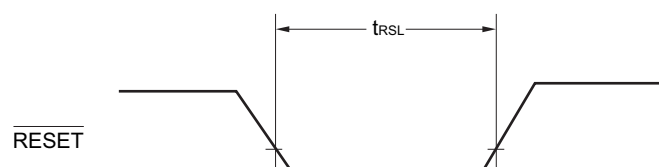
外部系统时钟时序



TI/TO 时序



中断请求的输入时序

 $\overline{\text{RESET}}$ 输入时序

32.5 外围功能特性

AC 时序测量点



32.5.1 串行阵列单元

(1) 同电位的通信 (UART 模式)

($T_A = -40 \sim +85^\circ\text{C}$ 、 $1.6\text{V} \leq V_{DD} \leq 5.5\text{V}$ 、 $V_{SS}=0\text{V}$)

项目	符号	条件	HS (高速主) 模式		LS (低速主) 模式		LV (低电压主) 模式		单位
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
传送速率 注 1		$2.4\text{V} \leq V_{DD} \leq 5.5\text{V}$		$f_{MCK}/6$		$f_{MCK}/6$		$f_{MCK}/6$	bps
		最大传送速率的理论值 $f_{MCK}=f_{CLK}$ 注 2		4.0		1.3		0.6	Mbps
		$1.8\text{V} \leq V_{DD} \leq 5.5\text{V}$		—		$f_{MCK}/6$		$f_{MCK}/6$	bps
		最大传送速率的理论值 $f_{MCK}=f_{CLK}$ 注 2		—		1.3		0.6	Mbps
		$1.6\text{V} \leq V_{DD} \leq 5.5\text{V}$		—		—		$f_{MCK}/6$	bps
		最大传送速率的理论值 $f_{MCK}=f_{CLK}$ 注 2		—		—		0.6	Mbps

注 1. SNOOZE 模式中的传送速率只有 4800bps。

2. CPU/ 外围硬件时钟 (f_{CLK}) 的最大工作频率如下所示。

HS (高速主) 模式: 24MHz ($2.7\text{V} \leq V_{DD} \leq 5.5\text{V}$)

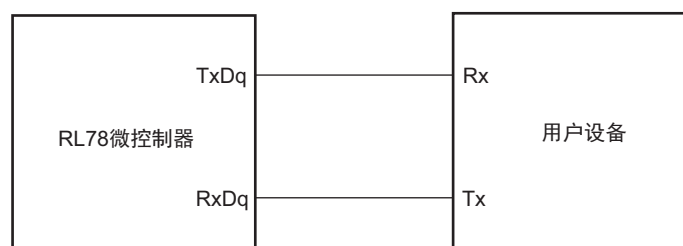
16MHz ($2.4\text{V} \leq V_{DD} \leq 5.5\text{V}$)

LS (低速主) 模式: 8MHz ($1.8\text{V} \leq V_{DD} \leq 5.5\text{V}$)

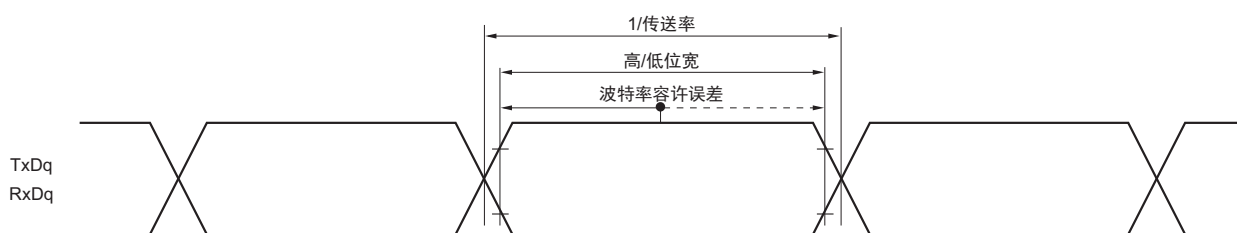
LV (低电压主) 模式: 4MHz ($1.6\text{V} \leq V_{DD} \leq 5.5\text{V}$)

注意 通过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 引脚选择为通常输入缓冲器并且将 TxDq 引脚选择为通常输出模式。

UART 模式连接图 (同电位的通信)



UART 模式的位宽（同电位的通信）（参考）



备注 1. q: UART 号 (q=0 ~ 3) g: PIM、POM 号 (g=0、1、3)

2. f_{MCK} : 串行阵列单元的运行时钟频率

（这是串行模式寄存器 mn (SMRmn) 的 CKS_{mn} 位设定的运行时钟。m: 单元号、n: 通道号 (mn=00 ~ 03、10 ~ 13)）

(2) 同电位的通信 (CSI 模式) (主控模式、SCKp... 内部时钟输出)

(T_A=−40 ~ +85°C、1.6V ≤ V_{DD} ≤ 5.5V、V_{SS} = 0 V)

项目	符号	条件	HS (高速主) 模式		LS (低速主) 模式		LV (低电压主) 模式		单位
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCKp 周期时间	t _{KCY1}	2.7V ≤ V _{DD} ≤ 5.5V	167 注 1		500 注 1		1000 注 1		ns
		2.4V ≤ V _{DD} ≤ 5.5V	250 注 1		500 注 1		1000 注 1		ns
		1.8V ≤ V _{DD} ≤ 5.5V	—		500 注 1		1000 注 1		ns
		1.6V ≤ V _{DD} ≤ 5.5V	—		—		1000 注 1		ns
SCKp 高 / 低电平 宽度	t _{KH1} 、 t _{KL1}	4.0V ≤ V _{DD} ≤ 5.5V	t _{KCY1} /2–12		t _{KCY1} /2–50		t _{KCY1} /2–50		ns
		2.7V ≤ V _{DD} ≤ 5.5V	t _{KCY1} /2–18		t _{KCY1} /2–50		t _{KCY1} /2–50		ns
		2.4V ≤ V _{DD} ≤ 5.5V	t _{KCY1} /2–38		t _{KCY1} /2–50		t _{KCY1} /2–50		ns
		1.8V ≤ V _{DD} ≤ 5.5V	—		t _{KCY1} /2–50		t _{KCY1} /2–50		ns
		1.6V ≤ V _{DD} ≤ 5.5V	—		—		t _{KCY1} /2–100		ns
Slp 准备时间 (对 SCKp↑) 注 2	t _{SIK1}	2.7V ≤ V _{DD} ≤ 5.5V	44		110		110		ns
		2.4V ≤ V _{DD} ≤ 5.5V	75		110		110		ns
		1.8V ≤ V _{DD} ≤ 5.5V	—		110		110		ns
		1.6V ≤ V _{DD} ≤ 5.5V	—		—		220		ns
Slp 保持时间 (对 SCKp↑) 注 2	t _{KSI1}	2.4V ≤ V _{DD} ≤ 5.5V	19		19		19		ns
		1.8V ≤ V _{DD} ≤ 5.5V	—		19		19		ns
		1.6V ≤ V _{DD} ≤ 5.5V	—		—		19		ns
SCKp↓→SOp 输出延迟时间注 3	t _{KSO1}	C=30pF 注 4	2.4V ≤ V _{DD} ≤ 5.5V	25		25		25	ns
			1.8V ≤ V _{DD} ≤ 5.5V	—		25		25	ns
			1.6V ≤ V _{DD} ≤ 5.5V	—		—		25	ns

注 1. 并且设定为 CSI00 ≥ 2/f_{CLK}。

2. 这是 DAPmn=0、CKPmn=0 或者 DAPmn=1、CKPmn=1 的情况。在 DAPmn=0、CKPmn=1 或者 DAPmn=1、CKPmn=0 时, 为“对 SCKp↓”。

3. 这是 DAPmn=0、CKPmn=0 或者 DAPmn=1、CKPmn=1 的情况。在 DAPmn=0、CKPmn=1 或者 DAPmn=1、CKPmn=0 时, 为“对 SCKp↑”。

4. C 是 SCKp、SO_p 输出线的负载电容。

注意 通过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 引脚选择为通常输入缓冲器并且将 SO_p 引脚和 SCKp 引脚选择为通常输出模式。

备注 1. p: CSI 号 (p=00) m: 单元号 (m=0) n: 通道号 (n=0) g: PIM、POM 号 (g=0、1)

2. f_{MCK}: 串行阵列单元的运行时钟频率

(这是串行模式寄存器 mn (SMRmn) 的 CKSmn 位设定的运行时钟。m: 单元号、n: 通道号 (mn=00))

(3) 同电位的通信 (CSI 模式) (从属模式、SCKp... 外部时钟输入)

(T_A=−40 ~ +85°C、1.6V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

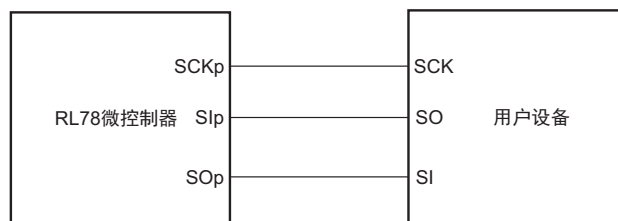
项目	符号	条件		HS (高速主) 模式		LS (低速主) 模式		LV (低电压主) 模式		单位
				MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCKp 周期 时间注 4	t _{KCY2}	4.0V ≤ V _{DD} ≤ 5.5V	f _{MCK} > 20MHz	8/f _{MCK}		—		—		ns
			f _{MCK} ≤ 20MHz	6/f _{MCK}		6/f _{MCK}		6/f _{MCK}		ns
		2.7V ≤ V _{DD} ≤ 5.5V	f _{MCK} > 16MHz	8/f _{MCK}		—		—		ns
			f _{MCK} ≤ 16MHz	6/f _{MCK}		6/f _{MCK}		6/f _{MCK}		ns
		2.4V ≤ V _{DD} ≤ 5.5V		6/f _{MCK} 并且 500		6/f _{MCK}		6/f _{MCK}		ns
		1.8V ≤ V _{DD} ≤ 5.5V		—		6/f _{MCK}		6/f _{MCK}		ns
		1.6V ≤ V _{DD} ≤ 5.5V		—		—		6/f _{MCK}		ns
SCKp 高/低电平 宽度	t _{KH2} 、 t _{KL2}	4.0V ≤ V _{DD} ≤ 5.5V		t _{KCY2} /2-7		t _{KCY2} /2-7		t _{KCY2} /2-7		ns
		2.7V ≤ V _{DD} ≤ 5.5V		t _{KCY2} /2-8		t _{KCY2} /2-8		t _{KCY2} /2-8		ns
		2.4V ≤ V _{DD} ≤ 5.5V		t _{KCY2} /2-18		t _{KCY2} /2-18		t _{KCY2} /2-18		ns
		1.8V ≤ V _{DD} ≤ 5.5V		—		t _{KCY2} /2-18		t _{KCY2} /2-18		ns
		1.6V ≤ V _{DD} ≤ 5.5V		—		—		t _{KCY2} /2-66		ns
Slp 准备时间 (对 SCKp↑) 注 1	t _{SIK2}	2.7V ≤ V _{DD} ≤ 5.5V		1/f _{MCK} +20		1/f _{MCK} +30		1/f _{MCK} +30		ns
		2.4V ≤ V _{DD} ≤ 5.5V		1/f _{MCK} +30		1/f _{MCK} +30		1/f _{MCK} +30		ns
		1.8V ≤ V _{DD} ≤ 5.5V		—		1/f _{MCK} +30		1/f _{MCK} +30		ns
		1.6V ≤ V _{DD} ≤ 5.5V		—		—		1/f _{MCK} +40		ns
Slp 保持时间 (对 SCKp↑) 注 1	t _{KSI2}	2.4V ≤ V _{DD} ≤ 5.5V		1/f _{MCK} +31		1/f _{MCK} +31		1/f _{MCK} +31		ns
		1.8V ≤ V _{DD} ≤ 5.5V		—		1/f _{MCK} +31		1/f _{MCK} +31		ns
		1.6V ≤ V _{DD} ≤ 5.5V		—		—		1/f _{MCK} +250		ns
SCKp↓→SOp 输出延迟时间 注 2	t _{KSO2}	C=30pF 注 3	2.7V ≤ V _{DD} ≤ 5.5V		2/f _{MCK} +44		2/f _{MCK} +110		2/f _{MCK} +110	ns
			2.4V ≤ V _{DD} ≤ 5.5V		2/f _{MCK} +75		2/f _{MCK} +110		2/f _{MCK} +110	ns
			1.8V ≤ V _{DD} ≤ 5.5V		—		2/f _{MCK} +110		2/f _{MCK} +110	ns
			1.6V ≤ V _{DD} ≤ 5.5V		—		—		2/f _{MCK} +220	ns

- 注 1. 这是 DAPmn=0、CKPmn=0 或者 DAPmn=1、CKPmn=1 的情况。在 DAPmn=0、CKPmn=1 或者 DAPmn=1、CKPmn=0 时, 为“对 SCKp↓”。
2. 这是 DAPmn=0、CKPmn=0 或者 DAPmn=1、CKPmn=1 的情况。在 DAPmn=0、CKPmn=1 或者 DAPmn=1、CKPmn=0 时, 为“对 SCKp↑”。
3. C 是 SOp 输出线的负载电容。
4. SNOOZE 模式中的传送速率为 MAX.1Mbps。

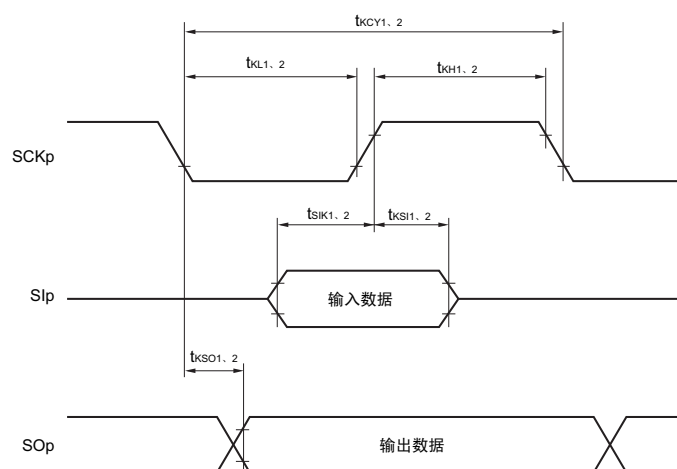
注意 通过端口输入模式寄存器 g (PIMg) 和端口输出模式寄存器 g (POMg), 将 Slp 引脚和 SCKp 引脚选择为通常输入缓冲器并且将 SOp 引脚选择为通常输出模式。

- 备注 1. p: CSI 号 (p=00) m: 单元号 (m=0) n: 通道号 (n=0) g: PIM、POM 号 (g=0、1)
2. f_{MCK}: 串行阵列单元的运行时钟频率
(这是串行模式寄存器 mn (SMRmn) 的 CKSmn 位设定的运行时钟。m: 单元号、n: 通道号 (mn=00))

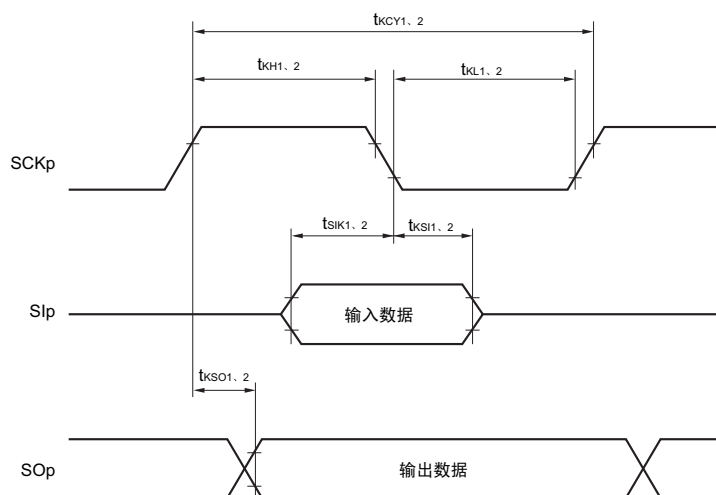
CSI 模式连接图（同电位的通信）



CSI 模式串行传送时序（同电位的通信）
 （DAPmn=0、CKPmn=0 或者 DAPmn=1、CKPmn=1 时）



CSI 模式串行传送时序（同电位的通信）
 （DAPmn=0、CKPmn=1 或者 DAPmn=1、CKPmn=0 时）



备注 1. p: CSI 编 (p=00)

2. m: 单元号、n: 通道号 (mn=00)

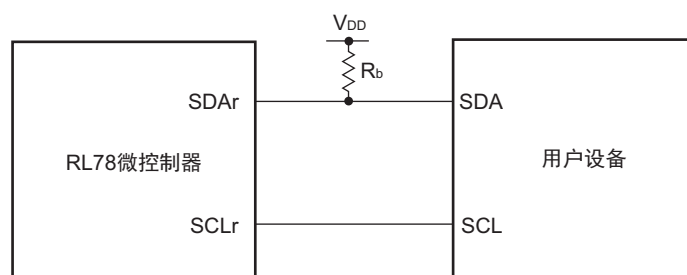
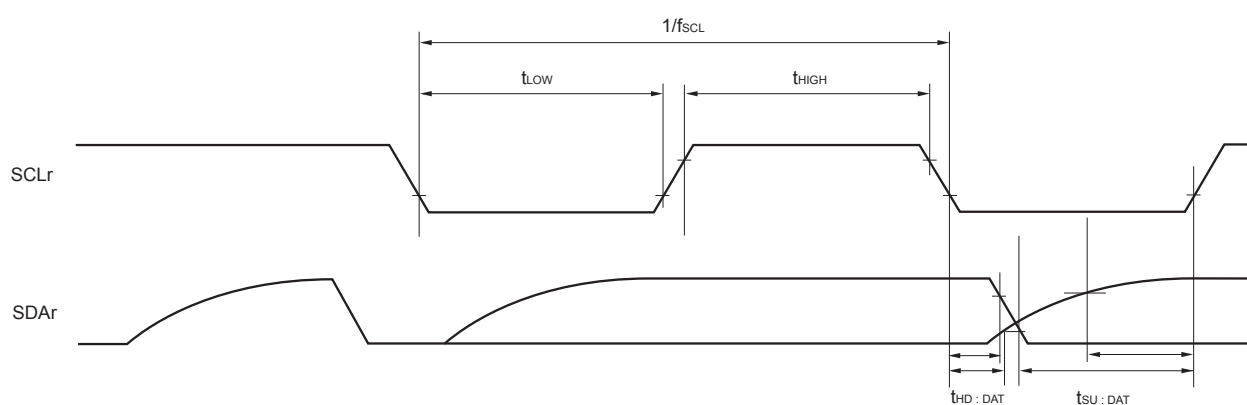
(4) 同电位的通信 (简易 I²C 模式)(T_A=−40 ~ +85°C、1.6V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件	HS (高速主) 模式		LS (低速主) 模式		LV (低电压主) 模式		单位
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCLr 时钟频率	f _{SCL}	2.7V ≤ V _{DD} ≤ 5.5V、 C _b =50pF、R _b =2.7kΩ		1000 注 1		400 注 1		400 注 1	kHz
		1.8V(2.4V 注 3) ≤ V _{DD} ≤ 5.5V、 C _b =100pF、R _b =3kΩ		400 注 1		400 注 1		400 注 1	kHz
		1.8V(2.4V 注 3) ≤ V _{DD} < 2.7V、 C _b =100pF、R _b =5kΩ		300 注 1		300 注 1		300 注 1	kHz
		1.6V ≤ V _{DD} < 1.8V、 C _b =100pF、R _b =5kΩ		—		—		250 注 1	kHz
SCLr="L" 的 保持时间	t _{LOW}	2.7V ≤ V _{DD} ≤ 5.5V、 C _b =50pF、R _b =2.7kΩ	475		1150		1150		ns
		1.8V(2.4V 注 3) ≤ V _{DD} ≤ 5.5V、 C _b =100pF、R _b =3kΩ	1150		1150		1150		ns
		1.8V(2.4V 注 3) ≤ V _{DD} < 2.7V、 C _b =100pF、R _b =5kΩ	1550		1550		1550		ns
		1.6V ≤ V _{DD} < 1.8V、 C _b =100pF、R _b =5kΩ	—		—		1850		ns
SCLr="H" 的 保持时间	t _{HIGH}	2.7V ≤ V _{DD} ≤ 5.5V、 C _b =50pF、R _b =2.7kΩ	475		1150		1150		ns
		1.8V(2.4V 注 3) ≤ V _{DD} ≤ 5.5V、 C _b =100pF、R _b =3kΩ	1150		1150		1150		ns
		1.8V(2.4V 注 3) ≤ V _{DD} < 2.7V、 C _b =100pF、R _b =5kΩ	1550		1550		1550		ns
		1.6V ≤ V _{DD} < 1.8V、 C _b =100pF、R _b =5kΩ	—		—		1850		ns
数据准备时间 (接收时)	t _{SU:} DAT	2.7V ≤ V _{DD} ≤ 5.5V、 C _b =50pF、R _b =2.7kΩ	1/f _{MCK} +85 注 2		1/f _{MCK} +145 注 2		1/f _{MCK} +145 注 2		ns
		1.8V(2.4V 注 3) ≤ V _{DD} ≤ 5.5V、 C _b =100pF、R _b =3kΩ	1/f _{MCK} +145 注 2		1/f _{MCK} +145 注 2		1/f _{MCK} +145 注 2		ns
		1.8V(2.4V 注 3) ≤ V _{DD} < 2.7V、 C _b =100pF、R _b =5kΩ	1/f _{MCK} +230 注 2		1/f _{MCK} +230 注 2		1/f _{MCK} +230 注 2		ns
		1.6V ≤ V _{DD} < 1.8V、 C _b =100pF、R _b =5kΩ	—		—		1/f _{MCK} +290 注 2		ns
数据保持时间 (发送时)	t _{HD:} DAT	2.7V ≤ V _{DD} ≤ 5.5V、 C _b =50pF、R _b =2.7kΩ	0	305	0	305	0	305	ns
		1.8V(2.4V 注 3) ≤ V _{DD} ≤ 5.5V、 C _b =100pF、R _b =3kΩ	0	355	0	355	0	355	ns
		1.8V(2.4V 注 3) ≤ V _{DD} < 2.7V、 C _b =100pF、R _b =5kΩ	0	405	0	405	0	405	ns
		1.6V ≤ V _{DD} < 1.8V、 C _b =100pF、R _b =5kΩ	—	—	—	—	0	405	ns

注 1. 并且必须设定为“≤ f_{MCK}/4”。2. f_{MCK} 的设定值不能超过 SCLr="L" 和 SCLr="H" 的保持时间的值。

3. 这是 HS (高速主) 模式时的条件。

注意 通过端口输入模式寄存器 g (PIMg) 和端口输出模式寄存器 g (POMg)，将 SDAr 引脚选择为通常输入缓冲器和 N 沟道漏极开路输出 (V_{DD} 耐压) 模式，并且将 SCLr 引脚选择为通常输出模式。

简易 I²C 模式的连接图（同电位的通信）简易 I²C 模式的串行传送时序（同电位的通信）

备注 1. $R_b[\Omega]$: 通信线 (SDAr) 的上拉电阻值 $C_b[F]$: 通信线 (SCLr、SDAr) 的负载电容值

2. r: IIC 号 (r=00、10) g: PIM 号、POM 号 (g=0、1)

3. f_{MCK} : 串行阵列单元的运行时钟频率

(这是 SMRmn 寄存器的 CKSmn 位设定的运行时钟。m: 单元号 (0) n: 通道号 (n=0、2) mn=00、02)

(5) 不同电位 (1.8V、2.5V、3V) 的通信 (UART 模式) (1/2)

(T_A=−40 ~ +85°C、1.8V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件	HS (高速主) 模式		LS (低速主) 模式		LV (低电压主) 模式		单位
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
传送速率	接收	4.0V ≤ V _{DD} ≤ 5.5V、 2.7V ≤ V _b ≤ 4.0V		f _{MCK} /6 注 1		f _{MCK} /6 注 1		f _{MCK} /6 注 1	bps
		最大传送速率的理论值 f _{MCK} =f _{CLK} 注 3		4.0		1.3		0.6	Mbps
		2.7V ≤ V _{DD} < 4.0V、 2.3V ≤ V _b ≤ 2.7V		f _{MCK} /6 注 1		f _{MCK} /6 注 1		f _{MCK} /6 注 1	bps
		最大传送速率的理论值 f _{MCK} =f _{CLK} 注 3		4.0		1.3		0.6	Mbps
		1.8V(2.4V 注 4) ≤ V _{DD} < 3.3V、 1.6V ≤ V _b ≤ 2.0V		f _{MCK} /6 注 1、2		f _{MCK} /6 注 1、2		f _{MCK} /6 注 1、2	bps
		最大传送速率的理论值 f _{MCK} =f _{CLK} 注 3		4.0		1.3		0.6	Mbps

注 1. SNOOZE 模式中的传送速率只有 4800bps。

2. 必须在 V_{DD} ≥ V_b 的情况下使用。3. CPU/ 外围硬件时钟 (f_{CLK}) 的最大工作频率如下所示。HS (高速主) 模式: 24MHz (2.7V ≤ V_{DD} ≤ 5.5V)16MHz (2.4V ≤ V_{DD} ≤ 5.5V)LS (低速主) 模式: 8MHz (1.8V ≤ V_{DD} ≤ 5.5V)LV (低电压主) 模式: 4MHz (1.6V ≤ V_{DD} ≤ 5.5V)

4. 这是 HS (高速主) 模式时的条件。

注意 通过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 输入缓冲器并且将 TxDq 引脚选择为 N 沟道漏极开路输出 (V_{DD} 耐压) 模式。另外，有关 V_{IH}、V_{IL}，请参照选择 TTL 输入缓冲器时的 DC 特性。

备注 1. V_b[V]: 通信线的电压

2. q: UART 号 (q=0 ~ 3) g: PIM、POM 号 (g=0、1、3)

3. f_{MCK}: 串行阵列单元的运行时钟频率

(这是串行模式寄存器 mn (SMRmn) 的 CKSmn 位设定的运行时钟。m: 单元号、n: 通道号 (mn=00 ~ 03、10 ~ 13))

(5) 不同电位 (1.8V、2.5V、3V) 的通信 (UART 模式) (2/2)

(T_A=−40 ~ +85°C、1.8V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件	HS (高速主) 模式		LS (低速主) 模式		LV (低电压主) 模式		单位
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
传送速率	发送	4.0V ≤ V _{DD} ≤ 5.5V、 2.7V ≤ V _b ≤ 4.0V		注 1		注 1		注 1	bps
		最大传送速率的理论值 C _b =50pF、R _b =1.4kΩ、V _b =2.7V		2.8 注 2		2.8 注 2		2.8 注 2	Mbps
		2.7V ≤ V _{DD} < 4.0V、 2.3V ≤ V _b ≤ 2.7V		注 3		注 3		注 3	bps
		最大传送速率的理论值 C _b =50pF、R _b =2.7kΩ、V _b =2.3V		1.2 注 4		1.2 注 4		1.2 注 4	Mbps
		1.8V(2.4 注 8) ≤ V _{DD} < 3.3V、 1.6V ≤ V _b ≤ 2.0V		注 5、6		注 5、6		注 5、6	bps
		最大传送速率的理论值 C _b =50pF、R _b =5.5kΩ、V _b =1.6V		0.43 注 7		0.43 注 7		0.43 注 7	Mbps

注 1. f_{MCK}/6 和用以下计算式算出的最大传送速率中的较小的值为有效的最大传送速率。4.0V ≤ V_{DD} ≤ 5.5V、2.7V ≤ V_b ≤ 4.0V 时的传送速率计算式：

$$\text{最大传送速率} = \frac{1}{\{-C_b \times R_b \times \ln(1 - \frac{2.2}{V_b})\} \times 3} \text{ [bps]}$$

$$\text{波特率容许误差 (理论值)} = \frac{\frac{1}{\text{传送速率} \times 2} - \{-C_b \times R_b \times \ln(1 - \frac{2.2}{V_b})\}}{(\frac{1}{\text{传送速率}}) \times \text{传送位数}} \times 100 \quad [\%]$$

※ 此值为发送方和接收方的相对误差的理论值。

2. 作为例子，此值是满足条件栏中条件的计算值。有关用户条件下的最大传送速率，必须根据注 1 进行计算。

3. f_{MCK}/6 和用以下计算式算出的最大传送速率中的较小的值为有效的最大传送速率。2.7V ≤ V_{DD} < 4.0V、2.3V ≤ V_b ≤ 2.7V 时的传送速率计算式：

$$\text{最大传送速率} = \frac{1}{\{-C_b \times R_b \times \ln(1 - \frac{2.0}{V_b})\} \times 3} \text{ [bps]}$$

$$\text{波特率容许误差 (理论值)} = \frac{\frac{1}{\text{传送速率} \times 2} - \{-C_b \times R_b \times \ln(1 - \frac{2.0}{V_b})\}}{(\frac{1}{\text{传送速率}}) \times \text{传送位数}} \times 100 \quad [\%]$$

※ 此值为发送方和接收方的相对误差的理论值。

4. 作为例子，此值是满足条件栏中条件的计算值。有关用户条件下的最大传送速率，必须根据注 3 进行计算。

5. 必须在 V_{DD} ≥ V_b 的情况下使用。

注 6. $f_{MCK}/6$ 和用以下计算式算出的最大传送速率中的较小的值为有效的最大传送速率。

1.8V(2.4V^{注8}) $\leq V_{DD} < 3.3V$ 、 $1.6V \leq V_b \leq 2.0V$ 时的传送速率计算式：

$$\text{最大传送速率} = \frac{1}{\{-C_b \times R_b \times \ln(1 - \frac{1.5}{V_b})\} \times 3} \text{ [bps]}$$

$$\text{波特率容许误差 (理论值)} = \frac{\frac{1}{\text{传送速率} \times 2} - \{-C_b \times R_b \times \ln(1 - \frac{1.5}{V_b})\}}{(\frac{1}{\text{传送速率}}) \times \text{传送位数}} \times 100 \quad [\%]$$

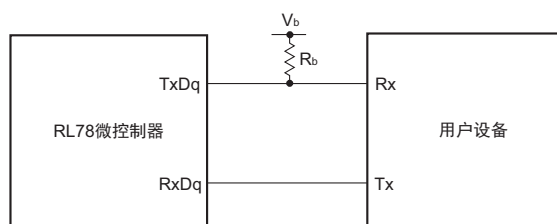
※ 此值为发送方和接收方的相对误差的理论值。

7. 作为例子，此值是满足条件栏中条件的计算值。有关用户条件下的最大传送速率，必须根据注 6 进行计算。

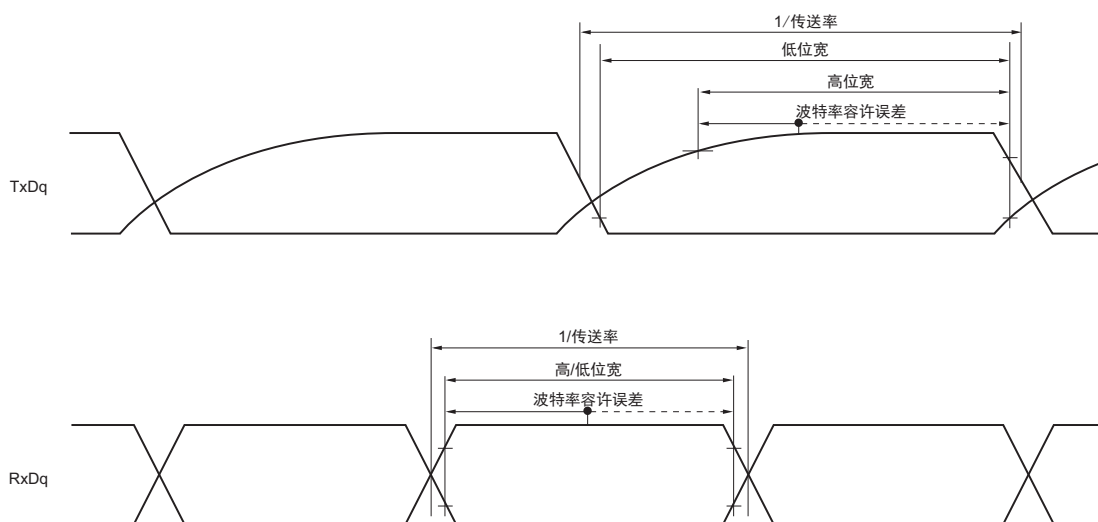
8. 这是 HS（高速主）模式时的条件。

注意 通过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 输入缓冲器并且将 TxDq 引脚选择为 N 沟道漏极开路输出（ V_{DD} 耐压）模式。另外，有关 V_{IH} 、 V_{IL} ，请参照选择 TTL 输入缓冲器时的 DC 特性。

UART 模式的连接图（不同电位的通信）



UART 模式的位宽（不同电位的通信）（参考）



备注 1. $R_b[\Omega]$: 通信线（TxDq）的上拉电阻值 $C_b[F]$: 通信线（TxDq）的负载电容值 $V_b[V]$: 通信线的电压

2. q: UART 号（q=0 ~ 3） g: PIM、POM 号（g=0、1、3）

3. f_{MCK} : 串行阵列单元的运行时钟频率

（这是串行模式寄存器 mn（SMRmn）的 CKSmn 位设定的运行时钟。m: 单元号、n: 通道号（mn=00 ~ 03、10 ~ 13））

(6) 不同电位 (2.5V、3V) 的通信 (CSI 模式) (主控模式、SCKp... 内部时钟输出、只对应 CSI00)

(T_A=−40 ~ +85°C、2.7V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件	HS (高速主) 模式		LS (低速主) 模式		LV (低电压主) 模式		单位
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCKp 周期时间	t _{KCY1}	t _{KCY1} ≥ 2/f _{CLK} 4.0V ≤ V _{DD} ≤ 5.5V 2.7V ≤ V _b ≤ 4.0V C _b =20pF、R _b =1.4kΩ	200		1150		1150		ns
		2.7V ≤ V _{DD} < 4.0V 2.3V ≤ V _b ≤ 2.7V C _b =20pF、R _b =2.7kΩ	300		1150		1150		ns
SCKp 高电平宽度	t _{KH1}	4.0V ≤ V _{DD} ≤ 5.5V、2.7V ≤ V _b ≤ 4.0V C _b =20pF、R _b =1.4kΩ	t _{KCY1} /2−50		t _{KCY1} /2−50		t _{KCY1} /2−50		ns
		2.7V ≤ V _{DD} < 4.0V、2.3V ≤ V _b ≤ 2.7V C _b =20pF、R _b =2.7kΩ	t _{KCY1} /2−120		t _{KCY1} /2−120		t _{KCY1} /2−120		ns
SCKp 低电平宽度	t _{KL1}	4.0V ≤ V _{DD} ≤ 5.5V、2.7V ≤ V _b ≤ 4.0V C _b =20pF、R _b =1.4kΩ	t _{KCY1} /2−7		t _{KCY1} /2−50		t _{KCY1} /2−50		ns
		2.7V ≤ V _{DD} < 4.0V、2.3V ≤ V _b ≤ 2.7V C _b =20pF、R _b =2.7kΩ	t _{KCY1} /2−10		t _{KCY1} /2−50		t _{KCY1} /2−50		ns
Slp 准备时间 (对 SCKp↑) 注 1	t _{SIK1}	4.0V ≤ V _{DD} ≤ 5.5V、2.7V ≤ V _b ≤ 4.0V C _b =20pF、R _b =1.4kΩ	58		479		479		ns
		2.7V ≤ V _{DD} < 4.0V、2.3V ≤ V _b ≤ 2.7V C _b =20pF、R _b =2.7kΩ	121		479		479		ns
Slp 保持时间 (对 SCKp↑) 注 1	t _{KSI1}	4.0V ≤ V _{DD} ≤ 5.5V、2.7V ≤ V _b ≤ 4.0V C _b =20pF、R _b =1.4kΩ	10		10		10		ns
		2.7V ≤ V _{DD} < 4.0V、2.3V ≤ V _b ≤ 2.7V C _b =20pF、R _b =2.7kΩ	10		10		10		ns
SCKp↓→Sop 输出延迟时间注 1	t _{KSO1}	4.0V ≤ V _{DD} ≤ 5.5V、2.7V ≤ V _b ≤ 4.0V C _b =20pF、R _b =1.4kΩ		60		60		60	ns
		2.7V ≤ V _{DD} < 4.0V、2.3V ≤ V _b ≤ 2.7V C _b =20pF、R _b =2.7kΩ		130		130		130	ns
Slp 准备时间 (对 SCKp↓) 注 2	t _{SIK1}	4.0V ≤ V _{DD} ≤ 5.5V、2.7V ≤ V _b ≤ 4.0V C _b =20pF、R _b =1.4kΩ	23		110		110		ns
		2.7V ≤ V _{DD} < 4.0V、2.3V ≤ V _b ≤ 2.7V C _b =20pF、R _b =2.7kΩ	33		110		110		ns
Slp 保持时间 (对 SCKp↓) 注 2	t _{KSI1}	4.0V ≤ V _{DD} ≤ 5.5V、2.7V ≤ V _b ≤ 4.0V C _b =20pF、R _b =1.4kΩ	10		10		10		ns
		2.7V ≤ V _{DD} < 4.0V、2.3V ≤ V _b ≤ 2.7V C _b =20pF、R _b =2.7kΩ	10		10		10		ns
SCKp↑→Sop 输出延迟时间注 2	t _{KSO1}	4.0V ≤ V _{DD} ≤ 5.5V、2.7V ≤ V _b ≤ 4.0V C _b =20pF、R _b =1.4kΩ		10		10		10	ns
		2.7V ≤ V _{DD} < 4.0V、2.3V ≤ V _b ≤ 2.7V C _b =20pF、R _b =2.7kΩ		10		10		10	ns

注 1. 这是 DAPmn=0、CKPmn=0 或者 DAPmn=1、CKPmn=1 的情况。

2. 这是 DAPmn=0、CKPmn=1 或者 DAPmn=1、CKPmn=0 的情况。

注意 通过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 输入缓冲器并且将 Sop 引脚和 SCKp 引脚选择为 N 沟道漏极开路输出 (V_{DD} 耐压) 模式。有关 V_{IH}、V_{IL}，请参照选择 TTL 输入缓冲器时的 DC 特性。

备注 1. R_b[Ω]: 通信线 (SCKp、Sop) 的上拉电阻值 C_b[F]: 通信线 (SCKp、Sop) 的负载电容值 V_b[V]: 通信线的电压

2. p: CSI 号 (p=00) m: 单元号 (m=0) n: 通道号 (n=0) g: PIM、POM 号 (g=0、1)

3. f_{MCK}: 串行阵列单元的运行时钟频率

(这是串行模式寄存器 mn (SMRmn) 的 CKSmn 位设定的运行时钟。m: 单元号、n: 通道号 (mn=00))

4. 此值只对应不使用 CSI00 的外围 I/O 重定向功能的情况。

(7) 不同电位 (1.8V、2.5V、3 V) 的通信 (CSI 模式) (主控模式、SCKp... 内部时钟输出) (1/2)

(T_A=−40 ~ +85°C、1.8V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件	HS (高速主) 模式		LS (低速主) 模式		LV (低电压主) 模式		单位
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCKp 周期时间	t _{KCY1}	t _{KCY1} ≥ 4/f _{CLK} 4.0V ≤ V _{DD} ≤ 5.5V 2.7V ≤ V _b ≤ 4.0V C _b =30pF、R _b =1.4kΩ	300		1150		1150		ns
		2.7V ≤ V _{DD} < 4.0V 2.3V ≤ V _b ≤ 2.7V C _b =30pF、R _b =2.7kΩ	500		1150		1150		ns
		1.8V(2.4V注1) ≤ V _{DD} < 3.3V 1.6V ≤ V _b ≤ 1.8V注2 C _b =30pF、R _b =5.5kΩ	1150		1150		1150		ns
SCKp 高电平宽度	t _{KH1}	4.0V ≤ V _{DD} ≤ 5.5V 2.7V ≤ V _b ≤ 4.0V C _b =30pF、R _b =1.4kΩ	t _{KCY1} /2−75		t _{KCY1} /2−75		t _{KCY1} /2−75		ns
		2.7V ≤ V _{DD} < 4.0V 2.3V ≤ V _b ≤ 2.7V C _b =30pF、R _b =2.7kΩ	t _{KCY1} /2−170		t _{KCY1} /2−170		t _{KCY1} /2−170		ns
		1.8V(2.4V注1) ≤ V _{DD} < 3.3V 1.6V ≤ V _b ≤ 2.0V注2 C _b =30pF、R _b =5.5kΩ	t _{KCY1} /2−458		t _{KCY1} /2−458		t _{KCY1} /2−458		ns
SCKp 低电平宽度	t _{KL1}	4.0V ≤ V _{DD} ≤ 5.5V 2.7V ≤ V _b ≤ 4.0V C _b =30pF、R _b =1.4kΩ	t _{KCY1} /2−12		t _{KCY1} /2−50		t _{KCY1} /2−50		ns
		2.7V ≤ V _{DD} < 4.0V 2.3V ≤ V _b ≤ 2.7V C _b =30pF、R _b =2.7kΩ	t _{KCY1} /2−18		t _{KCY1} /2−50		t _{KCY1} /2−50		ns
		1.8V(2.4V注1) ≤ V _{DD} < 3.3V 1.6V ≤ V _b ≤ 2.0V注2 C _b =30pF、R _b =5.5kΩ	t _{KCY1} /2−50		t _{KCY1} /2−50		t _{KCY1} /2−50		ns
Slp 准备时间 (对 SCKp↑) 注3	t _{SIK1}	4.0V ≤ V _{DD} ≤ 5.5V 2.7V ≤ V _b ≤ 4.0V C _b =30pF、R _b =1.4kΩ	81		479		479		ns
		2.7V ≤ V _{DD} < 4.0V 2.3V ≤ V _b ≤ 2.7V C _b =30pF、R _b =2.7kΩ	177		479		479		ns
		1.8V(2.4V注1) ≤ V _{DD} < 3.3V 1.6V ≤ V _b ≤ 2.0V注2 C _b =30pF、R _b =5.5kΩ	479		479		479		ns
Slp 保持时间 (对 SCKp↑) 注3	t _{KSI1}	4.0V ≤ V _{DD} ≤ 5.5V 2.7V ≤ V _b ≤ 4.0V C _b =30pF、R _b =1.4kΩ	19		19		19		ns
		2.7V ≤ V _{DD} < 4.0V 2.3V ≤ V _b ≤ 2.7V C _b =30pF、R _b =2.7kΩ	19		19		19		ns
		1.8V(2.4V注1) ≤ V _{DD} < 3.3V 1.6V ≤ V _b ≤ 2.0V注2 C _b =30pF、R _b =5.5kΩ	19		19		19		ns
SCKp↓→SOp 输出延迟时间注3	t _{KSO1}	4.0V ≤ V _{DD} ≤ 5.5V 2.7V ≤ V _b ≤ 4.0V C _b =30pF、R _b =1.4kΩ		100		100		100	ns
		2.7V ≤ V _{DD} < 4.0V 2.3V ≤ V _b ≤ 2.7V C _b =30pF、R _b =2.7kΩ		195		195		195	ns
		1.8V(2.4V注1) ≤ V _{DD} < 3.3V 1.6V ≤ V _b ≤ 2.0V注2 C _b =30pF、R _b =5.5kΩ		483		483		483	ns

(7) 不同电位 (1.8V、2.5V、3 V) 的通信 (CSI 模式) (主控模式、SCKp... 内部时钟输出) (2/2)

(T_A=−40 ~ +85°C、1.8V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件	HS (高速主) 模式		LS (低速主) 模式		LV (低电压主) 模式		单位
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
Slp 准备时间 (对 SCKp↓) 注 4	t _{SIK1}	4.0V ≤ V _{DD} ≤ 5.5V 2.7V ≤ V _b ≤ 4.0V C _b =30pF、R _b =1.4kΩ	44		110		110		ns
		2.7V ≤ V _{DD} < 4.0V 2.3V ≤ V _b ≤ 2.7V C _b =30pF、R _b =2.7kΩ	44		110		110		ns
		1.8V(2.4V 注 1) ≤ V _{DD} < 3.3V 1.6V ≤ V _b ≤ 2.0V 注 2 C _b =30pF、R _b =5.5kΩ	110		110		110		ns
Slp 保持时间 (对 SCKp↓) 注 4	t _{KS11}	4.0V ≤ V _{DD} ≤ 5.5V 2.7V ≤ V _b ≤ 4.0V C _b =30pF、R _b =1.4kΩ	19		19		19		ns
		2.7V ≤ V _{DD} < 4.0V 2.3V ≤ V _b ≤ 2.7V C _b =30pF、R _b =2.7kΩ	19		19		19		ns
		1.8V(2.4V 注 1) ≤ V _{DD} < 3.3V 1.6V ≤ V _b ≤ 2.0V 注 2 C _b =30pF、R _b =5.5kΩ	19		19		19		ns
SCKp↑→SO _p 输出延迟时间注 4	t _{KSO1}	4.0V ≤ V _{DD} ≤ 5.5V 2.7V ≤ V _b ≤ 4.0V C _b =30pF、R _b =1.4kΩ		25		25		25	ns
		2.7V ≤ V _{DD} < 4.0V 2.3V ≤ V _b ≤ 2.7V C _b =30pF、R _b =2.7kΩ		25		25		25	ns
		1.8V(2.4V 注 1) ≤ V _{DD} < 3.3V 1.6V ≤ V _b ≤ 2.0V 注 2 C _b =30pF、R _b =5.5kΩ		25		25		25	ns

注 1. 这是 HS (高速主) 模式时的条件。

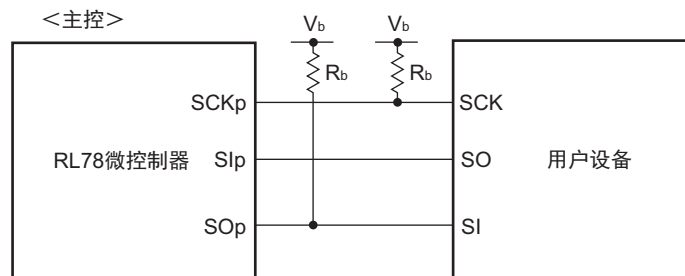
2. 必须在 V_{DD} ≥ V_b 的情况下使用。

3. 这是 DAPmn=0、CKPmn=0 或者 DAPmn=1、CKPmn=1 的情况。

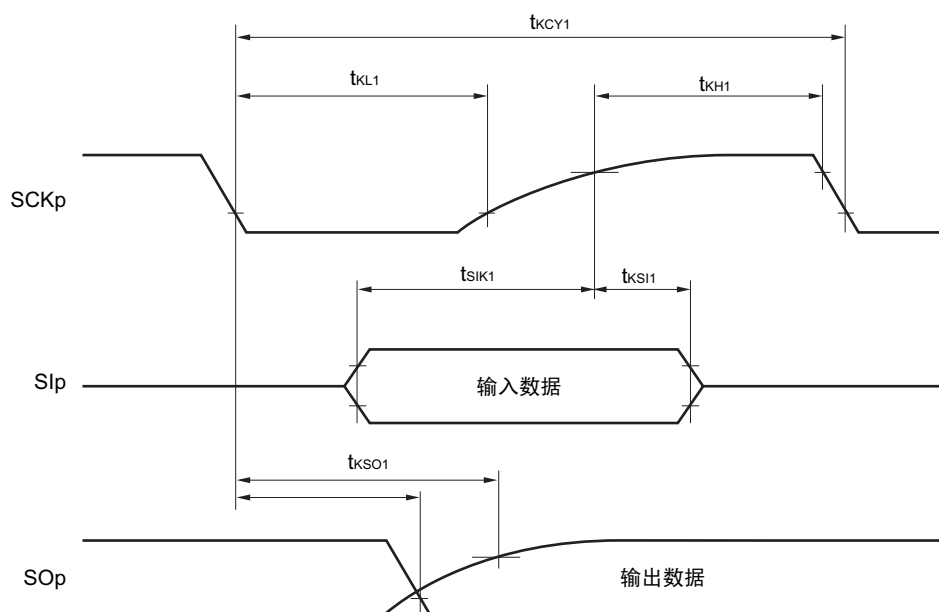
4. 这是 DAPmn=0、CKPmn=1 或者 DAPmn=1、CKPmn=0 的情况。

注意 通过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 输入缓冲器并且将 SO_p 引脚和 SCKp 引脚选择为 N 沟道漏极开路输出 (V_{DD} 耐压) 模式。有关 V_{IH}、V_{IL}，请参照选择 TTL 输入缓冲器时的 DC 特性。

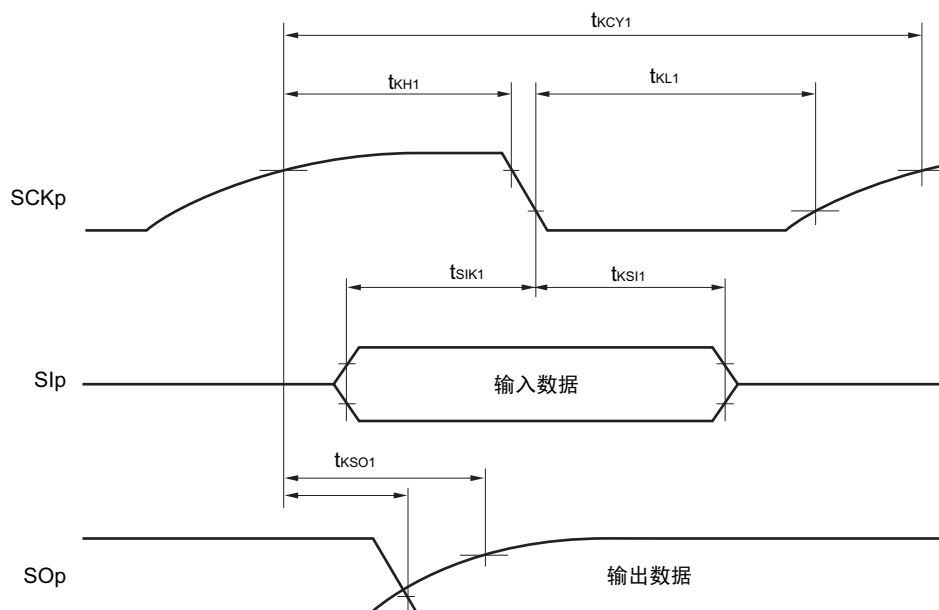
CSI 模式的连接图 (不同电位的通信)



CSI模式的串行传送时序：主控模式（不同电位的通信）
 （DAPmn=0、CKPmn=0 或者 DAPmn=1、CKPmn=1 的情况）



CSI模式的串行传送时序：主控模式（不同电位的通信）
 （DAPmn=0、CKPmn=1 或者 DAPmn=1、CKPmn=0 的情况）



备注 1. $R_b[\Omega]$: 通信线 (SCKp、SOp) 的上拉电阻值 $C_b[F]$: 通信线 (SCKp、SOp) 的负载电容值 $V_b[V]$: 通信线的电压

2. p: CSI 号 (p=00) m: 单元号 n: 通道号 (mn=00) g: PIM、POM 号 (g=0、1)

3. f_{MCK} : 串行阵列单元的运行时钟频率

（这是串行模式寄存器 mn (SMRmn) 的 CKSmn 位设定的运行时钟。m: 单元号、n: 通道号 (mn=00)）

(8) 不同电位 (1.8 V、2.5 V、3 V) 的通信 (CSI 模式) (从属模式、SCKp... 外部时钟输入)

(T_A=−40 ~ +85°C、1.8V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件		HS (高速主) 模式		LS (低速主) 模式		LV (低电压主) 模式		单位
				MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCKp 周期 时间注 1	t _{KCY2}	4.0V ≤ V _{DD} ≤ 5.5V、 2.7V ≤ V _b ≤ 4.0V	20MHz < f _{MCK}	12/f _{MCK}		—		—		ns
			8MHz < f _{MCK} ≤ 20MHz	10/f _{MCK}		—		—		ns
			4MHz < f _{MCK} ≤ 8MHz	8/f _{MCK}		16/f _{MCK}		—		ns
			f _{MCK} ≤ 4MHz	6/f _{MCK}		10/f _{MCK}		10/f _{MCK}		ns
		2.7V ≤ V _{DD} < 4.0V、 2.3V ≤ V _b ≤ 2.7V	20MHz < f _{MCK}	16/f _{MCK}		—		—		ns
			16MHz < f _{MCK} ≤ 20MHz	14/f _{MCK}		—		—		ns
			8MHz < f _{MCK} ≤ 16MHz	12/f _{MCK}		—		—		ns
			4MHz < f _{MCK} ≤ 8MHz	8/f _{MCK}		16/f _{MCK}		—		ns
			f _{MCK} ≤ 4MHz	6/f _{MCK}		10/f _{MCK}		10/f _{MCK}		ns
		1.8V(2.4V 注 2) ≤ V _{DD} < 3.3V、 1.6V ≤ V _b ≤ 2.0V 注 3	20MHz < f _{MCK}	36/f _{MCK}		—		—		ns
			16MHz < f _{MCK} ≤ 20MHz	32/f _{MCK}		—		—		ns
			8MHz < f _{MCK} ≤ 16MHz	26/f _{MCK}		—		—		ns
			4MHz < f _{MCK} ≤ 8MHz	16/f _{MCK}		16/f _{MCK}		—		ns
			f _{MCK} ≤ 4MHz	10/f _{MCK}		10/f _{MCK}		10/f _{MCK}		ns
SCKp 高 / 低 电平宽度	t _{KH2} 、 t _{KL2}	4.0V ≤ V _{DD} ≤ 5.5V、2.7V ≤ V _b ≤ 4.0V		t _{KCY2} /2–12		t _{KCY2} /2–50		t _{KCY2} /2–50		ns
		2.7V ≤ V _{DD} < 4.0V、2.3V ≤ V _b ≤ 2.7V		t _{KCY2} /2–18		t _{KCY2} /2–50		t _{KCY2} /2–50		ns
		1.8V(2.4V 注 2) ≤ V _{DD} < 3.3V、1.6V ≤ V _b ≤ 2.0V 注 3		t _{KCY2} /2–50		t _{KCY2} /2–50		t _{KCY2} /2–50		ns
Slp 准备时间 (对 SCKp↑) 注 4	t _{SIK2}	4.0V ≤ V _{DD} ≤ 5.5V、2.7V ≤ V _b ≤ 4.0V		1/f _{MCK} +20		1/f _{MCK} +30		1/f _{MCK} +30		ns
		2.7V ≤ V _{DD} < 4.0V、2.3V ≤ V _b ≤ 2.7V		1/f _{MCK} +20		1/f _{MCK} +30		1/f _{MCK} +30		ns
		1.8V(2.4V 注 2) ≤ V _{DD} < 3.3V、1.6V ≤ V _b ≤ 2.0V 注 3		1/f _{MCK} +30		1/f _{MCK} +30		1/f _{MCK} +30		ns
Slp 保持时间 (对 SCKp↑) 注 4	t _{SIK2}	4.0V ≤ V _{DD} ≤ 5.5V、2.7V ≤ V _b ≤ 4.0V		1/f _{MCK} +31		1/f _{MCK} +31		1/f _{MCK} +31		ns
		2.7V ≤ V _{DD} < 4.0V、2.3V ≤ V _b ≤ 2.7V		1/f _{MCK} +31		1/f _{MCK} +31		1/f _{MCK} +31		ns
		1.8V(2.4V 注 2) ≤ V _{DD} < 3.3V、1.6V ≤ V _b ≤ 2.0V 注 3		1/f _{MCK} +31		1/f _{MCK} +31		1/f _{MCK} +31		ns
SCKp↓→SOp 输出延迟时间 注 5	t _{KSO2}	4.0V ≤ V _{DD} ≤ 5.5V、2.7V ≤ V _b ≤ 4.0V、 C _b =30pF、R _b =1.4kΩ		2/f _{MCK} +120		2/f _{MCK} +573		2/f _{MCK} +573		ns
		2.7V ≤ V _{DD} < 4.0V、2.3V ≤ V _b ≤ 2.7V、 C _b =30pF、R _b =2.7kΩ		2/f _{MCK} +214		2/f _{MCK} +573		2/f _{MCK} +573		ns
		1.8V(2.4V 注 2) ≤ V _{DD} < 3.3V、1.6V ≤ V _b ≤ 2.0V 注 3 C _b =30pF、R _b =5.5kΩ		2/f _{MCK} +573		2/f _{MCK} +573		2/f _{MCK} +573		ns

注 1. SNOOZE 模式中的传送速率为 MAX.1Mbps。

2. 这是 HS (高速主) 模式时的条件。

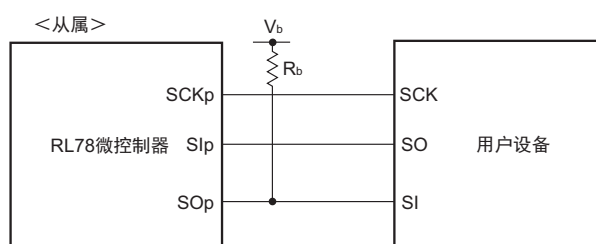
3. 必须在 V_{DD} ≥ V_b 的情况下使用。

4. 这是 DAPmn=0、CKPmn=0 或者 DAPmn=1、CKPmn=1 的情况。在 DAPmn=0、CKPmn=1 或者 DAPmn=1、CKPmn=0 时, 为“对 SCKp↓”。

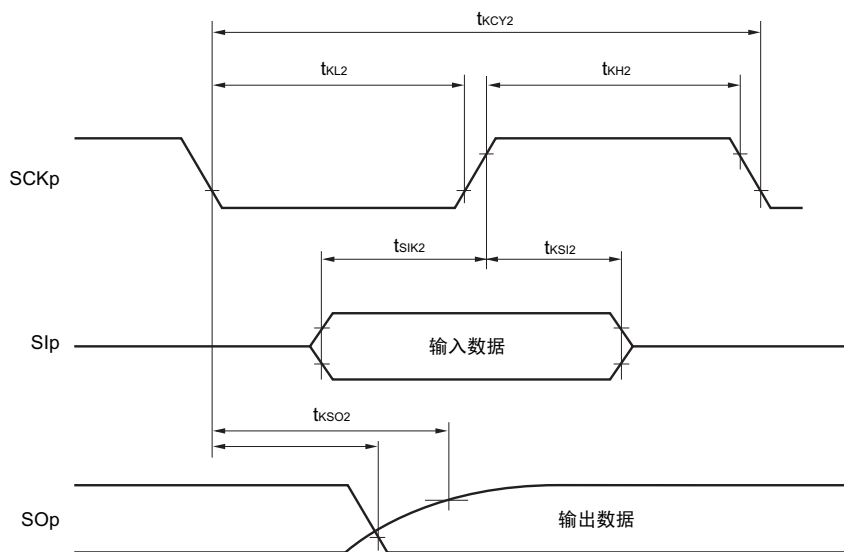
5. 这是 DAPmn=0、CKPmn=0 或者 DAPmn=1、CKPmn=1 的情况。在 DAPmn=0、CKPmn=1 或者 DAPmn=1、CKPmn=0 时, 为“对 SCKp↑”。

注意 通过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 引脚和 SCKp 引脚选择为 TTL 输入缓冲器并且将 SOp 引脚选择为 N 沟道漏极开路输出 (V_{DD} 耐压) 模式。有关 V_{IH}、V_{IL}, 请参照选择 TTL 输入缓冲器时的 DC 特性。

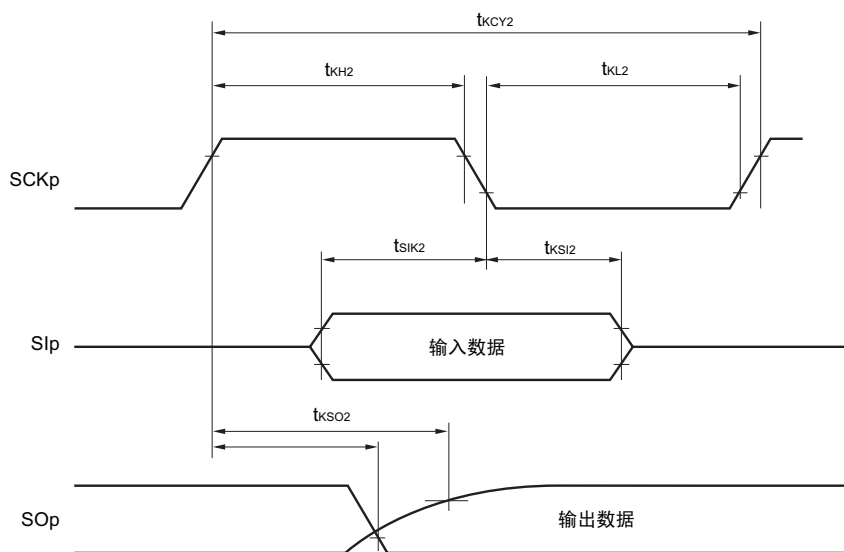
CSI 模式的连接图 (不同电位的通信)



CSI模式的串行传送时序：从属模式（不同电位的通信）
 （DAPmn=0、CKPmn=0或者DAPmn=1、CKPmn=1的情况）



CSI模式的串行传送时序：从属模式（不同电位的通信）
 （DAPmn=0、CKPmn=1或者DAPmn=1、CKPmn=0的情况）



备注 1. $R_b[\Omega]$: 通信线 (SO_p) 的上拉电阻值 $C_b[F]$: 通信线 (SO_p) 的负载电容值 $V_b[V]$: 通信线的电压

2. p: CSI 号 (p=00)、m: 单元号、n: 通道号 (mn=00)、g: PIM、POM 号 (g=0、1)

3. f_{MCK} : 串行阵列单元的运行时钟频率

（这是串行模式寄存器 mn (SMRmn) 的 CKSmn 位设定的运行时钟。m: 单元号、n: 通道号 (mn=00)）

(9) 不同电位 (1.8V、2.5V、3V) 的通信 (简易 I²C 模式) (1/2)(T_A=−40 ~ +85°C、1.8V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件	HS (高速主) 模式		LS (低速主) 模式		LV (低电压主) 模式		单位
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCLr 时钟频率	f _{SCL}	4.0V ≤ V _{DD} ≤ 5.5V、 2.7V ≤ V _b ≤ 4.0V、 C _b =50pF、R _b =2.7kΩ		1000 注 1		300 注 1		300 注 1	kHz
		2.7V ≤ V _{DD} < 4.0V、 2.3V ≤ V _b ≤ 2.7V、 C _b =50pF、R _b =2.7kΩ		1000 注 1		300 注 1		300 注 1	kHz
		4.0V ≤ V _{DD} ≤ 5.5V、 2.7V ≤ V _b ≤ 4.0V、 C _b =100pF、R _b =2.8kΩ		400 注 1		300 注 1		300 注 1	kHz
		2.7V ≤ V _{DD} < 4.0V、 2.3V ≤ V _b ≤ 2.7V、 C _b =100pF、R _b =2.7kΩ		400 注 1		300 注 1		300 注 1	kHz
		1.8V(2.4V 注 2) ≤ V _{DD} < 3.3V、 1.6V ≤ V _b ≤ 2.0V 注 3、 C _b =100pF、R _b =5.5kΩ		300 注 1		300 注 1		300 注 1	kHz
SCLr="L" 的保持 时间	t _{LOW}	4.0V ≤ V _{DD} ≤ 5.5V、 2.7V ≤ V _b ≤ 4.0V、 C _b =50pF、R _b =2.7kΩ	475		1550		1550		ns
		2.7V ≤ V _{DD} < 4.0V、 2.3V ≤ V _b ≤ 2.7V、 C _b =50pF、R _b =2.7kΩ	475		1550		1550		ns
		4.0V ≤ V _{DD} ≤ 5.5V、 2.7V ≤ V _b ≤ 4.0V、 C _b =100pF、R _b =2.8kΩ	1150		1550		1550		ns
		2.7V ≤ V _{DD} < 4.0V、 2.3V ≤ V _b ≤ 2.7V、 C _b =100pF、R _b =2.7kΩ	1150		1550		1550		ns
		1.8V(2.4V 注 2) ≤ V _{DD} < 3.3V、 1.6V ≤ V _b ≤ 2.0V 注 3、 C _b =100pF、R _b =5.5kΩ	1550		1550		1550		ns
SCLr="H" 的保持 时间	t _{HIGH}	4.0V ≤ V _{DD} ≤ 5.5V、 2.7V ≤ V _b ≤ 4.0V、 C _b =50pF、R _b =2.7kΩ	245		610		610		ns
		2.7V ≤ V _{DD} < 4.0V、 2.3V ≤ V _b ≤ 2.7V、 C _b =50pF、R _b =2.7kΩ	200		610		610		ns
		4.0V ≤ V _{DD} ≤ 5.5V、 2.7V ≤ V _b ≤ 4.0V、 C _b =100pF、R _b =2.8kΩ	675		610		610		ns
		2.7V ≤ V _{DD} < 4.0V、 2.3V ≤ V _b ≤ 2.7V、 C _b =100pF、R _b =2.7kΩ	600		610		610		ns
		1.8V(2.4V 注 2) ≤ V _{DD} < 3.3V、 1.6V ≤ V _b ≤ 2.0V 注 3、 C _b =100pF、R _b =5.5kΩ	610		610		610		ns

(9) 不同电位 (1.8V、2.5V、3V) 的通信 (简易 I²C 模式) (2/2)(T_A=−40 ~ +85°C、1.8V ≤ V_{DD} ≤ 5.5V、V_{SS}=0 V)

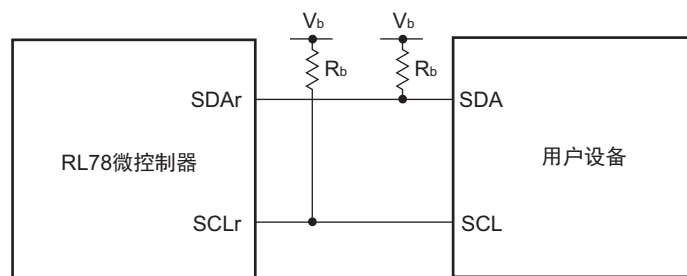
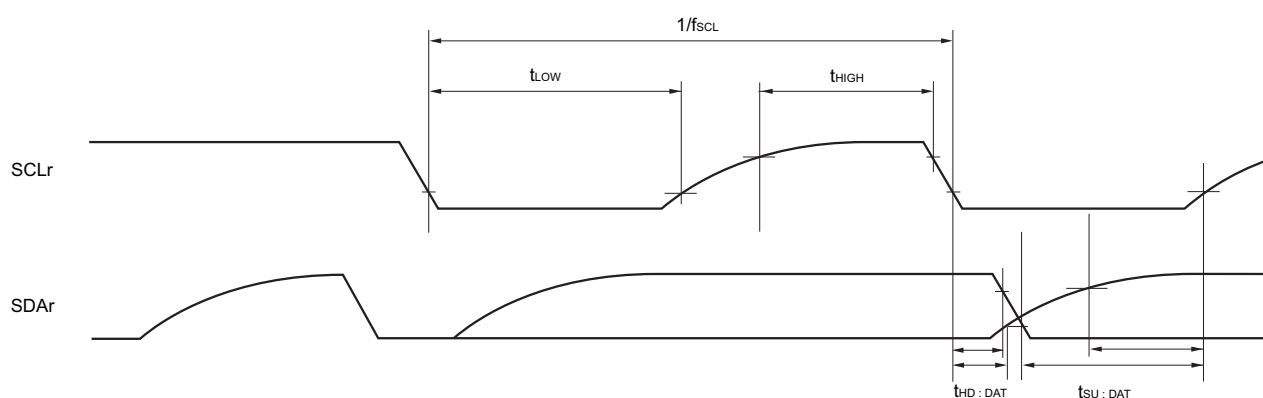
项目	符号	条件	HS (高速主) 模式		LS (低速主) 模式		LV (低电压主) 模式		单位
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
数据准备时间 (接收时)	t _{SU} : DAT	4.0V ≤ V _{DD} ≤ 5.5V、 2.7V ≤ V _b ≤ 4.0V、 C _b =50pF、R _b =2.7kΩ	1/f _{MCK} +135 注 4		1/f _{MCK} +190 注 4		1/f _{MCK} +190 注 4		ns
		2.7V ≤ V _{DD} < 4.0V、 2.3V ≤ V _b ≤ 2.7V、 C _b =50pF、R _b =2.7kΩ	1/f _{MCK} +135 注 4		1/f _{MCK} +190 注 4		1/f _{MCK} +190 注 4		ns
		4.0V ≤ V _{DD} ≤ 5.5V、 2.7V ≤ V _b ≤ 4.0V、 C _b =100pF、R _b =2.8kΩ	1/f _{MCK} +190 注 4		1/f _{MCK} +190 注 4		1/f _{MCK} +190 注 4		ns
		2.7V ≤ V _{DD} < 4.0V、 2.3V ≤ V _b ≤ 2.7V、 C _b =100pF、R _b =2.7kΩ	1/f _{MCK} +190 注 4		1/f _{MCK} +190 注 4		1/f _{MCK} +190 注 4		ns
		1.8V(2.4V 注 2) ≤ V _{DD} < 3.3V、 1.6V ≤ V _b ≤ 2.0V 注 3、 C _b =100pF、R _b =5.5kΩ	1/f _{MCK} +190 注 4		1/f _{MCK} +190 注 4		1/f _{MCK} +190 注 4		ns
数据保持时间 (发送时)	t _{HD} : DAT	4.0V ≤ V _{DD} ≤ 5.5V、 2.7V ≤ V _b ≤ 4.0V、 C _b =50pF、R _b =2.7kΩ	0	305	0	305	0	305	ns
		2.7V ≤ V _{DD} < 4.0V、 2.3V ≤ V _b ≤ 2.7V、 C _b =50pF、R _b =2.7kΩ	0	305	0	305	0	305	ns
		4.0V ≤ V _{DD} ≤ 5.5V、 2.7V ≤ V _b ≤ 4.0V、 C _b =100pF、R _b =2.8kΩ	0	355	0	355	0	355	ns
		2.7V ≤ V _{DD} < 4.0V、 2.3V ≤ V _b ≤ 2.7V、 C _b =100pF、R _b =2.7kΩ	0	355	0	355	0	355	ns
		1.8V(2.4V 注 2) ≤ V _{DD} < 3.3V、 1.6V ≤ V _b ≤ 2.0V 注 3、 C _b =100pF、R _b =5.5kΩ	0	405	0	405	0	405	ns

注 1. 并且必须设定为“≤ f_{MCK}/4”。

2. 这是 HS (高速主) 模式时的条件。

2. 必须在 V_{DD} ≥ V_b 的情况下使用。3. f_{MCK} 的设定值不能超过 SCLr=“L”和 SCLr=“H”的保持时间。

注意 通过端口输入模式寄存器 g (PIMg) 和端口输出模式寄存器 g (POMg)，将 SDAr 引脚选择为 TTL 输入缓冲器和 N 沟道漏极开路输出 (V_{DD} 耐压) 模式，并且将 SCLr 引脚选择为 N 沟道漏极开路输出 (V_{DD} 耐压) 模式。有关 V_{IH}、V_{IL}，请参照选择 TTL 输入缓冲器时的 DC 特性。

简易 I²C 模式连接图（不同电位的通信）简易 I²C 模式串行传送时序（不同电位的通信）

备注 1. $R_b[\Omega]$: 通信线 (SDAr、SCLr) 的上拉电阻值 $C_b[F]$: 通信线 (SDAr、SCLr) 的负载电容值 $V_b[V]$: 通信线的电压

2. r: IIC 号 (r=00、10) g: PIM、POM 号 (g=0、1)

3. f_{MCK} : 串行阵列单元的运行时钟频率

(这是 SMRmn 寄存器的 CKSmn 位设定的运行时钟。m: 单元号、n: 通道号 (mn=00、02))

32.5.2 串行接口 IICA

(1) I²C 标准模式(T_A=-40 ~ +85°C、1.6V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件		HS（高速主） 模式		LS（低速主） 模式		LV（低电压主） 模式		单位
				MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCLA0 时钟频率	f _{SCL}	标准模式： f _{CLK} ≥ 1MHz	2.7V ≤ V _{DD} ≤ 5.5V	0	100	0	100	0	100	kHz
			1.8V(2.4V 注 3) ≤ V _{DD} ≤ 5.5V	0	100	0	100	0	100	kHz
			1.6V ≤ V _{DD} ≤ 5.5V	—	—	—	—	0	100	kHz
重新开始条件的准备时间	t _{SU: STA}	2.7V ≤ V _{DD} ≤ 5.5V	4.7		4.7		4.7		μs	
		1.8V(2.4V 注 3) ≤ V _{DD} ≤ 5.5V	4.7		4.7		4.7		μs	
		1.6V ≤ V _{DD} ≤ 5.5V	—	—	—	—	4.7		μs	
保持时间注 1	t _{HD: STA}	2.7V ≤ V _{DD} ≤ 5.5V	4.0		4.0		4.0		μs	
		1.8V(2.4V 注 3) ≤ V _{DD} ≤ 5.5V	4.0		4.0		4.0		μs	
		1.6V ≤ V _{DD} ≤ 5.5V	—	—	—	—	4.0		μs	
SCLA0=“L” 的保持时间	t _{LOW}	2.7V ≤ V _{DD} ≤ 5.5V	4.7		4.7		4.7		μs	
		1.8V(2.4V 注 3) ≤ V _{DD} ≤ 5.5V	4.7		4.7		4.7		μs	
		1.6V ≤ V _{DD} ≤ 5.5V	—	—	—	—	4.7		μs	
SCLA0=“H” 的保持时间	t _{HIGH}	2.7V ≤ V _{DD} ≤ 5.5V	4.0		4.0		4.0		μs	
		1.8V(2.4V 注 3) ≤ V _{DD} ≤ 5.5V	4.0		4.0		4.0		μs	
		1.6V ≤ V _{DD} ≤ 5.5V	—	—	—	—	4.0		μs	
数据准备时间（接收时）	t _{SU: DAT}	2.7V ≤ V _{DD} ≤ 5.5V	250		250		250		ns	
		1.8V(2.4V 注 3) ≤ V _{DD} ≤ 5.5V	250		250		250		ns	
		1.6V ≤ V _{DD} ≤ 5.5V	—	—	—	—	250		ns	
数据保持时间（发送时）注 2	t _{HD: DAT}	2.7V ≤ V _{DD} ≤ 5.5V	0	3.45	0	3.45	0	3.45	μs	
		1.8V(2.4V 注 3) ≤ V _{DD} ≤ 5.5V	0	3.45	0	3.45	0	3.45	μs	
		1.6V ≤ V _{DD} ≤ 5.5V	—	—	—	—	0	3.45	μs	
停止条件的准备时间	t _{SU: STO}	2.7V ≤ V _{DD} ≤ 5.5V	4.0		4.0		4.0		μs	
		1.8V(2.4V 注 3) ≤ V _{DD} ≤ 5.5V	4.0		4.0		4.0		μs	
		1.6V ≤ V _{DD} ≤ 5.5V	—	—	—	—	4.0		μs	
总线空闲时间	t _{BUF}	2.7V ≤ V _{DD} ≤ 5.5V	4.7		4.7		4.7		μs	
		1.8V(2.4V 注 3) ≤ V _{DD} ≤ 5.5V	4.7		4.7		4.7		μs	
		1.6V ≤ V _{DD} ≤ 5.5V	—	—	—	—	4.7		μs	

- 注 1. 在开始条件和重新开始条件的情况下, 在此期间之后生成第一个时钟脉冲。
2. t_{HD: DAT} 的最大值 (MAX.) 是通常传送时的数值, 在进行应答 ACK (应答) 时需要等待。
3. 这是 HS (高速主) 模式时的条件。

注意 在外围 I/O 重定向寄存器 (PIOR) 的 bit2 (PIOR2) 为 "1" 时, 也适用上述值。但是, 引脚特性 (I_{OH1}、I_{OL1}、V_{OH1}、V_{OL1}) 必须满足重定向目标的值。

备注 各模式的 C_b (通信线电容) 的 MAX. 值和此时的 R_b (通信线的上拉电阻值) 的值如下所示。
标准模式: C_b=400pF、R_b=2.7kΩ

(2) I²C 快速模式(T_A=-40 ~ +85°C、1.6V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件	HS (高速主) 模式		LS (低速主) 模式		LV (低电压主) 模式		单位
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCLA0 时钟频率	f _{SCL}	快速模式:							
		f _{CLK} ≥ 3.5MHz							
		2.7V ≤ V _{DD} ≤ 5.5V	0	400	0	400	0	400	kHz
重新开始条件的准备时间	t _{SU: STA}	1.8V(2.4V注3) ≤ V _{DD} ≤ 5.5V	0	400	0	400	0	400	kHz
		2.7V ≤ V _{DD} ≤ 5.5V	0.6		0.6	400	0.6		μs
保持时间注1	t _{HD: STA}	1.8V(2.4V注3) ≤ V _{DD} ≤ 5.5V	0.6		0.6		0.6		μs
		2.7V ≤ V _{DD} ≤ 5.5V	0.6		0.6		0.6		μs
SCLA0="L" 的保持时间	t _{LOW}	1.8V(2.4V注3) ≤ V _{DD} ≤ 5.5V	0.6		0.6		0.6		μs
		2.7V ≤ V _{DD} ≤ 5.5V	1.3		1.3		1.3		μs
SCLA0="H" 的保持时间	t _{HIGH}	1.8V(2.4V注3) ≤ V _{DD} ≤ 5.5V	0.6		0.6		0.6		μs
		2.7V ≤ V _{DD} ≤ 5.5V	0.6		0.6		0.6		μs
数据准备时间 (接收时)	t _{SU: DAT}	1.8V(2.4V注3) ≤ V _{DD} ≤ 5.5V	100		100		100		ns
		2.7V ≤ V _{DD} ≤ 5.5V	100		100		100		ns
数据保持时间 (发送时)注2	t _{HD: DAT}	1.8V(2.4V注3) ≤ V _{DD} ≤ 5.5V	0	0.9	0	0.9	0	0.9	μs
		2.7V ≤ V _{DD} ≤ 5.5V	0	0.9	0	0.9	0	0.9	μs
停止条件的准备时间	t _{SU: STO}	1.8V(2.4V注3) ≤ V _{DD} ≤ 5.5V	0.6		0.6		0.6		μs
		2.7V ≤ V _{DD} ≤ 5.5V	0.6		0.6		0.6		μs
总线空闲时间	t _{BUF}	1.8V(2.4V注3) ≤ V _{DD} ≤ 5.5V	1.3		1.3		1.3		μs
		2.7V ≤ V _{DD} ≤ 5.5V	1.3		1.3		1.3		μs

注 1. 在开始条件和重新开始条件的情况下, 在此期间之后生成第一个时钟脉冲。

2. t_{HD: DAT} 的最大值 (MAX.) 是通常传送时的数值, 在进行应答 ACK (应答) 时需要等待。

3. 这是 HS (高速主) 模式时的条件。

注意 在外围 I/O 重定向寄存器 (PIOR) 的 bit2 (PIOR2) 为 "1" 时, 也适用上述值。但是, 引脚特性 (I_{OH1}、I_{OL1}、V_{OH1}、V_{OL1}) 必须满足重定向目标的值。

备注 各模式的 C_b (通信线电容) 的 MAX. 值和此时的 R_b (通信线的上拉电阻值) 的值如下所示。

快速模式: C_b=320pF、R_b=1.1kΩ

(3) I²C 增强型快速模式(T_A=−40 ~ +85°C、1.6V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件	HS (高速主) 模式		LS (低速主) 模式		LV (低电压主) 模式		单位
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCLA0 时钟频率	f _{SCL}	增强型快速模式: f _{CLK} ≥ 10MHz	0	1000	—	—	—	—	kHz
重新开始条件的准备时间	t _{SU: STA}	2.7V ≤ V _{DD} ≤ 5.5V	0.26		—	—	—	—	μs
保持时间 ^{注 1}	t _{HD: STA}	2.7V ≤ V _{DD} ≤ 5.5V	0.26		—	—	—	—	μs
SCLA0="L" 的保持时间	t _{LOW}	2.7V ≤ V _{DD} ≤ 5.5V	0.5		—	—	—	—	μs
SCLA0="H" 的保持时间	t _{HIGH}	2.7V ≤ V _{DD} ≤ 5.5V	0.26		—	—	—	—	μs
数据准备时间 (接收时)	t _{SU: DAT}	2.7V ≤ V _{DD} ≤ 5.5V	50		—	—	—	—	ns
数据保持时间 (发送时) ^{注 2}	t _{HD: DAT}	2.7V ≤ V _{DD} ≤ 5.5V	0	0.45	—	—	—	—	μs
停止条件的准备时间	t _{SU: STO}	2.7V ≤ V _{DD} ≤ 5.5V	0.26		—	—	—	—	μs
总线空闲时间	t _{BUF}	2.7V ≤ V _{DD} ≤ 5.5V	0.5		—	—	—	—	μs

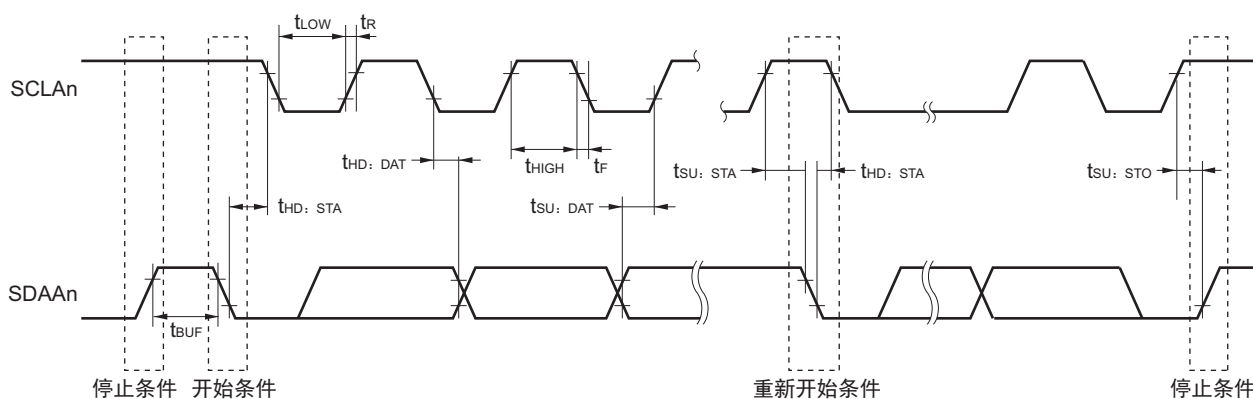
注 1. 在开始条件和重新开始条件的情况下, 在此期间之后生成第一个时钟脉冲。

2. t_{HD: DAT} 的最大值 (MAX.) 是通常传送时的数值, 在进行应答 ACK (应答) 时需要等待。

注意 在外围 I/O 重定向寄存器 (PIOR) 的 bit2 (PIOR2) 为 "1" 时, 也适用上述值。但是, 引脚特性 (I_{OH1}、I_{OL1}、V_{OH1}、V_{OL1}) 必须满足重定向目标的值。

备注 各模式的 C_b (通信线电容) 的 MAX. 值和此时的 R_b (通信线的上拉电阻值) 的值如下所示。
增强型模式: C_b=120pF、R_b=1.1kΩ

IICA 串行传送时序



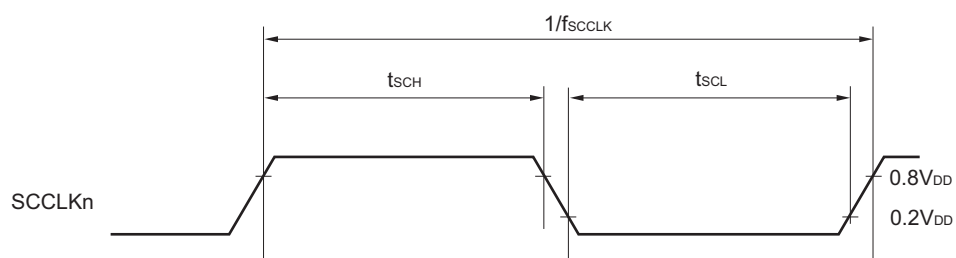
32.5.3 智能卡接口 (SMCI)

(T_A=−40 ~ +85°C、1.6 V ≤ V_{DD} ≤ 5.5V、V_{SS}=0 V)

参数	符号	条件	增强快速模式		单位
			最小值	最大值	
SCCLKn 时钟频率	f _{SCCLK}	2.7V ≤ V _{DD} ≤ 5.5V	符合 ISO/IEC7816-3 标准	6	MHz
		2.4V ≤ V _{DD} ≤ 5.5V		4	
		1.8V ≤ V _{DD} ≤ 5.5V		2	
		1.6V ≤ V _{DD} ≤ 5.5V		1	
SCCLKn 高电平 / 低电平宽度	t _{SCH} 、t _{SCL}	4.0V ≤ V _{DD} ≤ 5.5V	1/(f _{SCCLK} × 2) − 12		ns
		2.7V ≤ V _{DD} ≤ 5.5V	1/(f _{SCCLK} × 2) − 18		
		2.4V ≤ V _{DD} ≤ 5.5V	1/(f _{SCCLK} × 2) − 38		
		1.8V ≤ V _{DD} ≤ 5.5V	1/(f _{SCCLK} × 2) − 50		
		1.6V ≤ V _{DD} ≤ 5.5V	1/(f _{SCCLK} × 2) − 100		

备注 n: 通道号 (n = 0、1)

SMCI 时序



32.6 模拟特性

32.6.1 A/D 转换器特性

A/D 转换器特性的区分

输入通道 \ 基准电压	基准电压 (+) = AV_{REFP} 基准电压 (-) = AV_{REFM}	基准电压 (+) = V_{DD} 基准电压 (-) = V_{SS}	基准电压 (+) = V_{BGR} 基准电压 (-) = AV_{REFM}
ANI0、ANI1	—	参照 32.6.1(2)	参照 32.6.1(3)
ANI16、ANI17	参照 32.6.1(1)		—
内部基准电压 温度传感器输出电压	参照 32.6.1(1)		

(1) 选择基准电压 (+) = $AV_{REFP}/ANI0$ ($ADREFP1=0$ 、 $ADREFP0=1$)、基准电压 (-) = $AV_{REFM}/ANI1$ ($ADREFM=1$) 时, 转换对象为: ANI16、ANI17、内部基准电压、温度传感器输出电压

($T_A=-40 \sim +85^{\circ}\text{C}$ 、 $1.6\text{V} \leq V_{DD} \leq 5.5\text{V}$ 、 $V_{SS}=0\text{V}$ 、基准电压 (+) = AV_{REFP} 、基准电压 (-) = $AV_{REFM}=0\text{V}$)

项目	符号	条件	MIN.	TYP.	MAX.	单位
分辨率	RES		8		10	bit
综合误差注 1	AINL	10 位分辨率 $AV_{REFP}=V_{DD}$ 注 3	$1.8\text{V} \leq AV_{REFP} \leq 5.5\text{V}$	1.2	± 5.0	LSB
			$1.6\text{V} \leq AV_{REFP} \leq 5.5\text{V}$ 注 4	1.2	± 8.5	LSB
转换时间	t_{CONV}	10 位分辨率 转换对象: ANI16、ANI17	$3.6\text{V} \leq V_{DD} \leq 5.5\text{V}$	2.125	39	μs
			$2.7\text{V} \leq V_{DD} \leq 5.5\text{V}$	3.1875	39	μs
			$1.8\text{V} \leq V_{DD} \leq 5.5\text{V}$	17	39	μs
			$1.6\text{V} \leq V_{DD} \leq 5.5\text{V}$	57	95	μs
		10 位分辨率 转换对象: 内部基准电压、温度传 感器输出电压 (HS (高速主) 模式)	$3.6\text{V} \leq V_{DD} \leq 5.5\text{V}$	2.375	39	μs
			$2.7\text{V} \leq V_{DD} \leq 5.5\text{V}$	3.5625	39	μs
			$2.4\text{V} \leq V_{DD} \leq 5.5\text{V}$	17	39	μs
零刻度误差注 1、2	E_{ZS}	10 位分辨率 $AV_{REFP}=V_{DD}$ 注 3	$1.8\text{V} \leq AV_{REFP} \leq 5.5\text{V}$		± 0.35	%FSR
			$1.6\text{V} \leq AV_{REFP} \leq 5.5\text{V}$ 注 4		± 0.60	%FSR
满刻度误差注 1、2	E_{FS}	10 位分辨率 $AV_{REFP}=V_{DD}$ 注 3	$1.8\text{V} \leq AV_{REFP} \leq 5.5\text{V}$		± 0.35	%FSR
			$1.6\text{V} \leq AV_{REFP} \leq 5.5\text{V}$ 注 4		± 0.60	%FSR
积分线性误差注 1	ILE	10 位分辨率 $AV_{REFP}=V_{DD}$ 注 3	$1.8\text{V} \leq AV_{REFP} \leq 5.5\text{V}$		± 3.5	LSB
			$1.6\text{V} \leq AV_{REFP} \leq 5.5\text{V}$ 注 4		± 6.0	LSB
微分线性误差注 1	DLE	10 位分辨率 $AV_{REFP}=V_{DD}$ 注 3	$1.8\text{V} \leq AV_{REFP} \leq 5.5\text{V}$		± 2.0	LSB
			$1.6\text{V} \leq AV_{REFP} \leq 5.5\text{V}$ 注 4		± 2.5	LSB
模拟输入电压	V_{AIN}	ANI16、ANI17	0		AV_{REFP}	V
		内部基准电压 ($2.4\text{V} \leq V_{DD} \leq 5.5\text{V}$ 、HS (高速主) 模式)		V_{BGR} 注 5		V
		温度传感器输出电压 ($2.4\text{V} \leq V_{DD} \leq 5.5\text{V}$ 、HS (高速主) 模式)		V_{TMPS25} 注 5		V

注 1. 不包含量化误差 ($\pm 1/2$ LSB)。

2. 用对满刻度值的比率 (%FSR) 表示。

3. 在 $AV_{REFP} < V_{DD}$ 时, MAX. 值如下变化。

综合误差: 在 $AV_{REFP}=V_{DD}$ 时的 MAX. 值上加 $\pm 4\text{LSB}$

零刻度误差 / 满刻度误差: 在 $AV_{REFP}=V_{DD}$ 时的 MAX. 值上加 $\pm 0.2\%$ FSR

积分线性误差 / 微分线性误差: 在 $AV_{REFP}=V_{DD}$ 时的 MAX. 值上加 $\pm 2\text{LSB}$

4. 这是将转换时间设定为 MIN. $57\mu\text{s}$ 、MAX. $95\mu\text{s}$ 时的值。

5. 请参照“32.6.2 温度传感器 / 内部基准电压特性”。

(2) 选择基准电压 (+) = V_{DD} (ADREFP1=0、ADREFP0=0)、基准电压 (-) = V_{SS} (ADREFM=0) 时,
转换对象为: ANI0、ANI1、ANI16、ANI17、内部基准电压、温度传感器输出电压

($T_A = -40 \sim +85^{\circ}\text{C}$ 、 $1.6\text{V} \leq V_{DD} \leq 5.5\text{V}$ 、 $V_{SS}=0\text{V}$ 、基准电压 (+) = V_{DD} 、基准电压 (-) = V_{SS})

项目	符号	条件	MIN.	TYP.	MAX.	单位
分辨率	RES		8		10	bit
综合误差注 1、2	AINL	10 位分辨率	$1.8\text{V} \leq V_{DD} \leq 5.5\text{V}$	1.2	± 7.0	LSB
			$1.6\text{V} \leq V_{DD} \leq 5.5\text{V}$ 注 3	1.2	± 10.5	LSB
转换时间	t_{CONV}	10 位分辨率 转换对象: ANI0、ANI1、ANI16、 ANI17 注 3	$3.6\text{V} \leq V_{DD} \leq 5.5\text{V}$	2.125	39	μs
			$2.7\text{V} \leq V_{DD} \leq 5.5\text{V}$	3.1875	39	μs
			$1.8\text{V} \leq V_{DD} \leq 5.5\text{V}$	17	39	μs
			$1.6\text{V} \leq V_{DD} \leq 5.5\text{V}$	57	95	μs
		10 位分辨率 转换对象: 内部基准电 压、温度传感器输出电 压 (HS (高速主) 模式)	$3.6\text{V} \leq V_{DD} \leq 5.5\text{V}$	2.375	39	μs
			$2.7\text{V} \leq V_{DD} \leq 5.5\text{V}$	3.5625	39	μs
			$2.4\text{V} \leq V_{DD} \leq 5.5\text{V}$	17	39	μs
零刻度误差注 1、2	E_{ZS}	10 位分辨率	$1.8\text{V} \leq V_{DD} \leq 5.5\text{V}$		± 0.60	%FSR
			$1.6\text{V} \leq V_{DD} \leq 5.5\text{V}$ 注 3		± 0.85	%FSR
满刻度误差注 1、2	E_{FS}	10 位分辨率	$1.8\text{V} \leq V_{DD} \leq 5.5\text{V}$		± 0.60	%FSR
			$1.6\text{V} \leq V_{DD} \leq 5.5\text{V}$ 注 3		± 0.85	%FSR
积分线性误差注 1	ILE	10 位分辨率	$1.8\text{V} \leq V_{DD} \leq 5.5\text{V}$		± 4.0	LSB
			$1.6\text{V} \leq V_{DD} \leq 5.5\text{V}$ 注 3		± 6.5	LSB
微分线性误差注 1	DLE	10 位分辨率	$1.8\text{V} \leq V_{DD} \leq 5.5\text{V}$		± 2.0	LSB
			$1.6\text{V} \leq V_{DD} \leq 5.5\text{V}$ 注 3		± 2.5	LSB
模拟输入电压	V_{AIN}	ANI0、ANI1、ANI16、ANI17	0		V_{DD}	V
		内部基准电压 ($2.4\text{V} \leq V_{DD} \leq 5.5\text{V}$ 、HS (高速主) 模式)	V_{BGR} 注 4			V
		温度传感器输出电压 ($2.4\text{V} \leq V_{DD} \leq 5.5\text{V}$ 、HS (高速主) 模式)	V_{TMPS25} 注 4			V

注 1. 不包含量化误差 ($\pm 1/2$ LSB)。

2. 用对满刻度值的比率 (%FSR) 表示。

3. 这是将转换时间设定为 MIN.57 μs 、MAX.95 μs 时的值。

4. 请参照“32.6.2 温度传感器 / 内部基准电压特性”。

(3) 选择基准电压 (+) = 内部基准电压 (ADREFP1=1、ADREFP0=0)、基准电压 (-) = $AV_{REFM}/ANI1$ (ADREFM=1) 时, 转换对象为: ANI0、ANI16、ANI17

($T_A = -40 \sim +85^{\circ}\text{C}$ 、 $2.4\text{V} \leq V_{DD} \leq 5.5\text{V}$ 、 $V_{SS}=0\text{V}$ 、基准电压 (+) = V_{BGR} 注 3、基准电压 (-) = AV_{REFM} 注 4 = 0V 、HS (高速主) 模式)

项目	符号	条件	MIN.	TYP.	MAX.	单位
分辨率	RES		8			bit
转换时间	t_{CONV}	8 位分辨率 $2.4\text{V} \leq V_{DD} \leq 5.5\text{V}$	17		39	μs
零刻度误差注 1、2	E_{ZS}	8 位分辨率 $2.4\text{V} \leq V_{DD} \leq 5.5\text{V}$			± 0.60	%FSR
积分线性误差注 1	ILE	8 位分辨率 $2.4\text{V} \leq V_{DD} \leq 5.5\text{V}$			± 2.0	LSB
微分线性误差注 1	DLE	8 位分辨率 $2.4\text{V} \leq V_{DD} \leq 5.5\text{V}$			± 1.0	LSB
模拟输入电压	V_{AIN}		0		V_{BGR} 注 3	V

注 1. 不包含量化误差 ($\pm 1/2$ LSB)。

2. 用对满刻度值的比率 (%FSR) 表示。

3. 请参照“32.6.2 温度传感器 / 内部基准电压特性”。

4. 在基准电压 (-) = V_{SS} 时, MAX. 值如下变化。

综合误差: 在基准电压 (-) = AV_{REFM} 时的 MAX. 值上加 $\pm 0.35\%$ FSR

零刻度误差 / 满刻度误差: 在基准电压 (-) = AV_{REFM} 时的 MAX. 值上加 ± 0.5 LSB

积分线性误差 / 微分线性误差: 在基准电压 (-) = AV_{REFM} 时的 MAX. 值上加 ± 0.2 LSB

32.6.2 温度传感器 / 内部基准电压特性

($T_A = -40 \sim +85^{\circ}\text{C}$ 、 $2.4\text{V} \leq V_{DD} \leq 5.5\text{V}$ 、 $V_{SS}=0\text{V}$ 、HS (高速主) 模式)

项目	符号	条件	MIN.	TYP.	MAX.	单位
温度传感器输出电压	V_{TMPS25}	设定 ADS 寄存器 = 80H、 $T_A = +25^{\circ}\text{C}$		1.05		V
内部基准电压	V_{BGR}	设定 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		$\text{mV}/^{\circ}\text{C}$
工作稳定等待时间	t_{AMP}		5			μs

32.6.3 比较器

(T_A=−40 ~ +85°C、1.6V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件	MIN.	TYP.	MAX.	单位
输入电压范围	Ivref		0		V _{DD} −1.4	V
	Ivcmp		−0.3		V _{DD} +0.3	V
输出迟延	td	V _{DD} = 3.0 V 输入转换速率 > 50 mV/μs 比较器高速模式、 基本模式			1.2	μs
		比较器高速模式、 窗口模式			2.0	μs
		比较器低速模式、 基本模式		3.0	5.0	μs
高电位的基准电压	VTW+	比较器高速模式、窗口模式	0.66V _{DD}	0.76V _{DD}	0.86V _{DD}	V
低电位的基准电压	VTW−	比较器高速模式、窗口模式	0.14V _{DD}	0.24V _{DD}	0.34V _{DD}	V
工作稳定等待时间	t _{CMP}		100			μs
内部基准电压注	V _{BGR}	2.4V ≤ V _{DD} ≤ 5.5V、HS（高速主）模式	1.38	1.45	1.50	V

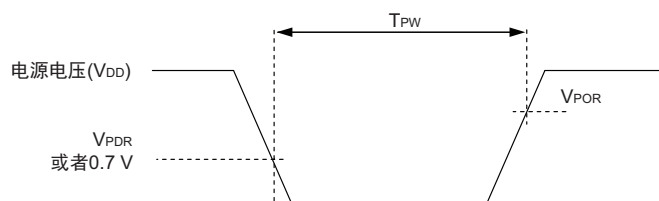
注 在 LS（低速主）模式、LV（低电压主）模式、副系统时钟运行以及 STOP 模式时不能使用。

32.6.4 POR 电路特性

(T_A=−40 ~ +85°C、V_{SS}=0V)

项目	符号	条件	MIN.	TYP.	MAX.	单位
检测电压	V _{POR}	电源电压上升时	1.47	1.51	1.55	V
	V _{PDR}	电源电压下降时	1.46	1.50	1.54	V
最小脉宽注	T _{PW}		300			μs

注 这是在 V_{DD} 低于 V_{PDR} 时 POR 复位所需的时间。另外，也是在 STOP 模式和通过设定时钟运行状态控制寄存器（CSC）的 bit0（HIOSTOP）和 bit7（MSTOP）停止主系统时钟（f_{MAIN}）运行时，从 V_{DD} 低于 0.7V 到回升超过 V_{POR} 为止 POR 复位所需的时间。



32.6.5 LVD 电路特性

复位模式和中断模式的 LVD 检测电压

(T_A=−40 ~ +85°C、V_{PDR} ≤ V_{DD} ≤ 5.5V、V_{SS} = 0V)

项目		符号	条件	MIIN.	TYP.	MAX.	单位
检测电压	电源电压电位	V _{LVD0}	电源上升时	3.98	4.06	4.14	V
			电源下降时	3.90	3.98	4.06	V
		V _{LVD1}	电源上升时	3.68	3.75	3.82	V
			电源下降时	3.60	3.67	3.74	V
		V _{LVD2}	电源上升时	3.07	3.13	3.19	V
			电源下降时	3.00	3.06	3.12	V
		V _{LVD3}	电源上升时	2.96	3.02	3.08	V
			电源下降时	2.90	2.96	3.02	V
		V _{LVD4}	电源上升时	2.86	2.92	2.97	V
			电源下降时	2.80	2.86	2.91	V
		V _{LVD5}	电源上升时	2.76	2.81	2.87	V
			电源下降时	2.70	2.75	2.81	V
		V _{LVD6}	电源上升时	2.66	2.71	2.76	V
			电源下降时	2.60	2.65	2.70	V
		V _{LVD7}	电源上升时	2.56	2.61	2.66	V
			电源下降时	2.50	2.55	2.60	V
		V _{LVD8}	电源上升时	2.45	2.50	2.55	V
			电源下降时	2.40	2.45	2.50	V
		V _{LVD9}	电源上升时	2.05	2.09	2.13	V
			电源下降时	2.00	2.04	2.08	V
		V _{LVD10}	电源上升时	1.94	1.98	2.02	V
			电源下降时	1.90	1.94	1.98	V
		V _{LVD11}	电源上升时	1.84	1.88	1.91	V
			电源下降时	1.80	1.84	1.87	V
		V _{LVD12}	电源上升时	1.74	1.77	1.81	V
			电源下降时	1.70	1.73	1.77	V
		V _{LVD13}	电源上升时	1.64	1.67	1.70	V
			电源下降时	1.60	1.63	1.66	V
最小脉宽		t _{LW}		300			μs
检测迟延						300	μs

中断和复位模式的 LVD 检测电压

(T_A=−40 ~ +85°C、V_{PDR} ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件		MIN.	TYP.	MAX.	单位		
中断 & 复位模式	V _{LVD13}	V _{POC2} 、	V _{POC1} 、 V _{POC0} =0、 0、 0， 下降复位电压		1.60	1.63	1.66	V	
	V _{LVD12}		LVIS1、 LVIS0=1、 0	上升复位解除电压	1.74	1.77	1.81	V	
				下降中断电压	1.70	1.73	1.77	V	
	V _{LVD11}		LVIS1、 LVIS0=0、 1	上升复位解除电压	1.84	1.88	1.91	V	
				下降中断电压	1.80	1.84	1.87	V	
	V _{LVD4}		LVIS1、 LVIS0=0、 0	上升复位解除电压	2.86	2.92	2.97	V	
		下降中断电压		2.80	2.86	2.91	V		
	V _{LVD11}	V _{POC2} 、	V _{POC1} 、 V _{POC0} =0、 0、 1， 下降复位电压		1.80	1.84	1.87	V	
	V _{LVD10}		LVIS1、 LVIS0=1、 0	上升复位解除电压	1.94	1.98	2.02	V	
				下降中断电压	1.90	1.94	1.98	V	
	V _{LVD9}		LVIS1、 LVIS0=0、 1	上升复位解除电压	2.05	2.09	2.13	V	
				下降中断电压	2.00	2.04	2.08	V	
	V _{LVD2}		LVIS1、 LVIS0=0、 0	上升复位解除电压	3.07	3.13	3.19	V	
				下降中断电压	3.00	3.06	3.12	V	
	V _{LVD8}		V _{POC2} 、	V _{POC1} 、 V _{POC0} =0、 1、 0， 下降复位电压		2.40	2.45	2.50	V
	V _{LVD7}			LVIS1、 LVIS0=1、 0	上升复位解除电压	2.56	2.61	2.66	V
					下降中断电压	2.50	2.55	2.60	V
	V _{LVD6}	LVIS1、 LVIS0=0、 1		上升复位解除电压	2.66	2.71	2.76	V	
				下降中断电压	2.60	2.65	2.70	V	
	V _{LVD1}	LVIS1、 LVIS0=0、 0		上升复位解除电压	3.68	3.75	3.82	V	
				下降中断电压	3.60	3.67	3.74	V	
	V _{LVD5}	V _{POC2} 、		V _{POC1} 、 V _{POC0} =0、 1、 1， 下降复位电压		2.70	2.75	2.81	V
	V _{LVD4}		LVIS1、 LVIS0=1、 0	上升复位解除电压	2.86	2.92	2.97	V	
				下降中断电压	2.80	2.86	2.91	V	
	V _{LVD3}		LVIS1、 LVIS0=0、 1	上升复位解除电压	2.96	3.02	3.08	V	
				下降中断电压	2.90	2.96	3.02	V	
	V _{LVD0}		LVIS1、 LVIS0=0、 0	上升复位解除电压	3.98	4.06	4.14	V	
				下降中断电压	3.90	3.98	4.06	V	

32.6.6 电源电压上升斜率特性

(T_A=−40 ~ +85°C、V_{SS}=0V)

项目	符号	条件	MIN.	TYP.	MAX.	单位
电源电压上升斜率	SV _{DD}				54	V/ms

注意 通过 LVD 电路或者外部复位保持内部复位状态，直到 V_{DD} 达到“32.4 AC 特性”所示的工作电压范围内。

32.7 LCD 特性

32.7.1 外部电阻分压方式

(1) 静态显示模式

 $(T_A = -40 \sim +85^{\circ}\text{C}, V_{L4}(\text{MIN.}) \leq V_{DD} \leq 5.5\text{V}, V_{SS} = 0\text{V})$

项目	符号	条件	MIN.	TYP.	MAX.	单位
LCD 驱动电压	V_{L4}		2.0		V_{DD}	V

(2) 1/2 偏压、1/4 偏压

 $(T_A = -40 \sim +85^{\circ}\text{C}, V_{L4}(\text{MIN.}) \leq V_{DD} \leq 5.5\text{V}, V_{SS} = 0\text{V})$

项目	符号	条件	MIN.	TYP.	MAX.	单位
LCD 驱动电压	V_{L4}		2.7		V_{DD}	V

(3) 1/3 偏压

 $(T_A = -40 \sim +85^{\circ}\text{C}, V_{L4}(\text{MIN.}) \leq V_{DD} \leq 5.5\text{V}, V_{SS} = 0\text{V})$

参数	符号	条件	MIN.	TYP.	MAX.	单位
LCD 驱动电压	V_{L4}		2.5		V_{DD}	V

32.7.2 内部升压方式

(1) 1/3 偏压

(T_A=−40 ~ +85°C、1.8V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件		MIN.	TYP.	MAX.	单位
LCD 输出电压可变范围	V _{L1}	C1 ~ C4 注 1 = 0.47μF 注 2	VLCD=04H	0.90	1.00	1.08	V
			VLCD=05H	0.95	1.05	1.13	V
			VLCD=06H	1.00	1.10	1.18	V
			VLCD=07H	1.05	1.15	1.23	V
			VLCD=08H	1.10	1.20	1.28	V
			VLCD=09H	1.15	1.25	1.33	V
			VLCD=0AH	1.20	1.30	1.38	V
			VLCD=0BH	1.25	1.35	1.43	V
			VLCD=0CH	1.30	1.40	1.48	V
			VLCD=0DH	1.35	1.45	1.53	V
			VLCD=0EH	1.40	1.50	1.58	V
			VLCD=0FH	1.45	1.55	1.63	V
			VLCD=10H	1.50	1.60	1.68	V
			VLCD=11H	1.55	1.65	1.73	V
			VLCD=12H	1.60	1.70	1.78	V
			VLCD=13H	1.65	1.75	1.83	V
倍频器输出电压	V _{L2}	C1 ~ C4 注 1=0.47μF	2V _{L1} -0.10	2V _{L1}	2V _{L1}	V	
三倍频器输出电压	V _{L3}	C1 ~ C4 注 1=0.47μF	2V _{L1} -0.15	3V _{L1}	3V _{L1}	V	
基准电压准备时间 注 2	t _{VWAIT1}		5			ms	
升压等待时间 注 3	t _{VWAIT2}	C1 ~ C4 注 1=0.47μF	500			ms	

注 1. 这是连接在用于驱动 LCD 的电压引脚之间的电容器。

C1: 连接在 CAPH 和 CAPL 之间的电容器。

C2: 连接在 V_{L1} 和 GND 之间的电容器。C3: 连接在 V_{L2} 和 GND 之间的电容器。C4: 连接在 V_{L4} 和 GND 之间的电容器。

C1=C2=C3=C4=0.47μF±30 %

2. 这是从使用 VLCD 寄存器设定基准电压后（当使用默认值作为基准电压时，在选择内部升压方式（LCDM0 寄存器的 MDSET1、MDSET0=01B）后），到开始升压（VLCON=1）前所需要的等待时间。

3. 这是从开始升压后（VLCON=1）到可以显示（LCDON=1）前的等待时间。

(2) 1/4 偏压

(T_A=−40 ~ +85°C、1.8V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件		MIN.	TYP.	MAX.	单位
LCD 输出电压可变范围	V _{L1}	C1 ~ C5 注 1 = 0.47 μF 注 2	VLCD=04H	0.90	1.00	1.08	V
			VLCD=05H	0.95	1.05	1.13	V
			VLCD=06H	1.00	1.10	1.18	V
			VLCD=07H	1.05	1.15	1.23	V
			VLCD=08H	1.10	1.20	1.28	V
			VLCD=09H	1.15	1.25	1.33	V
			VLCD=0AH	1.20	1.30	1.38	V
倍频器输出电压	V _{L2}	C1 ~ C5 注 1=0.47μF		2V _{L1} -0.08	2V _{L1}	2V _{L1}	V
三倍频器输出电压	V _{L3}	C1 ~ C5 注 1=0.47μF		3V _{L1} -0.12	3V _{L1}	3V _{L1}	V
四倍频器输出电压	V _{L4}	C1 ~ C5 注 1=0.47μF		4V _{L1} -0.16	4V _{L1}	4V _{L1}	V
基准电压准备时间注 2	t _{VWAIT1}			5			ms
升压等待时间 注 3	t _{VWAIT2}	C1 ~ C5 注 1=0.47μF		500			ms

注 1. 这是连接在用于驱动 LCD 的电压引脚之间的电容器。

C1: 连接在 CAPH 和 CAPL 之间的电容器。

C2: 连接在 V_{L1} 和 GND 之间的电容器。C3: 连接在 V_{L2} 和 GND 之间的电容器。C4: 连接在 V_{L3} 和 GND 之间的电容器。C5: 连接在 V_{L4} 和 GND 之间的电容器。

C1=C2=C3=C4=C5=0.47 μF±30 %

2. 这是从使用 VLCD 寄存器设定基准电压后（当使用默认值作为基准电压时，在选择内部升压方式（LCDM0 寄存器的 MDSET1、MDSET0=01B）后），到开始升压（VLCON=1）前所需要的等待时间。

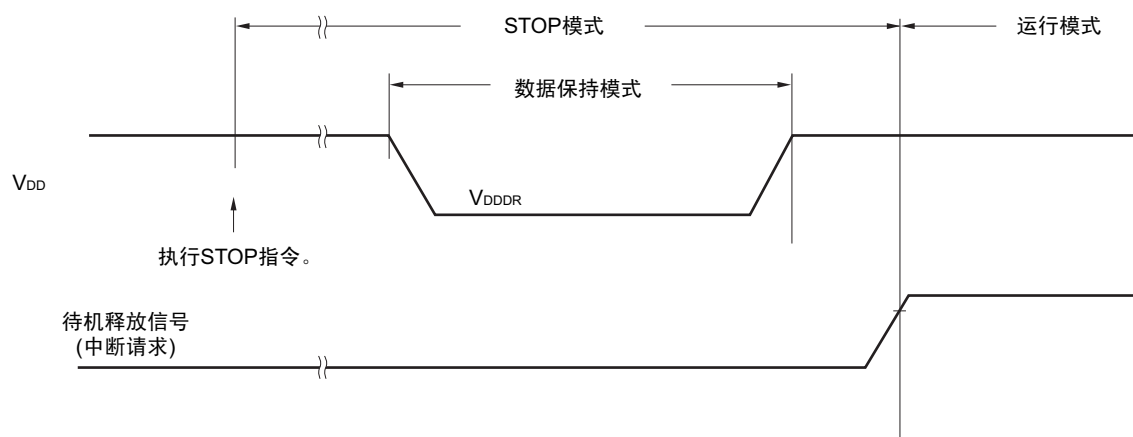
3. 这是从开始升压后（VLCON=1）到可以显示（LCDON=1）前的等待时间。

32.8 STOP 模式中的数据存储器的低电源电压数据保持特性

(T_A=−40 ~ +85°C)

项目	符号	条件	MIN.	TYP.	MAX.	单位
数据保持电源电压	V _{DDDR}		1.46 注		5.5	V

注 取决于POR检测电压。当电压降下时，保持数据，直到发生POR复位为止。但是，当发生POR复位时，不保持数据。



32.9 闪存编程特性

(T_A=−40 ~ +85°C、1.8V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件	MIN.	TYP.	MAX.	单位
系统时钟频率	f _{CLK}	1.8V ≤ V _{DD} ≤ 5.5V	1		24	MHz
闪存的改写次数注 1、2、3	C _{erwr}	保持年数：20 年 T _A =85°C	1000			次

注 1. 1 次改写是指 1 次擦除 + 擦除后的 1 次编程。保持年数是指从 1 次改写到下次改写为止的期间。

2. 这是使用闪存编程器和本公司提供的库程序的情况。

3. 此特性表示闪存的特性，是本公司的可靠性试验的结果。

32.10 专用闪存编程器通信 (UART)

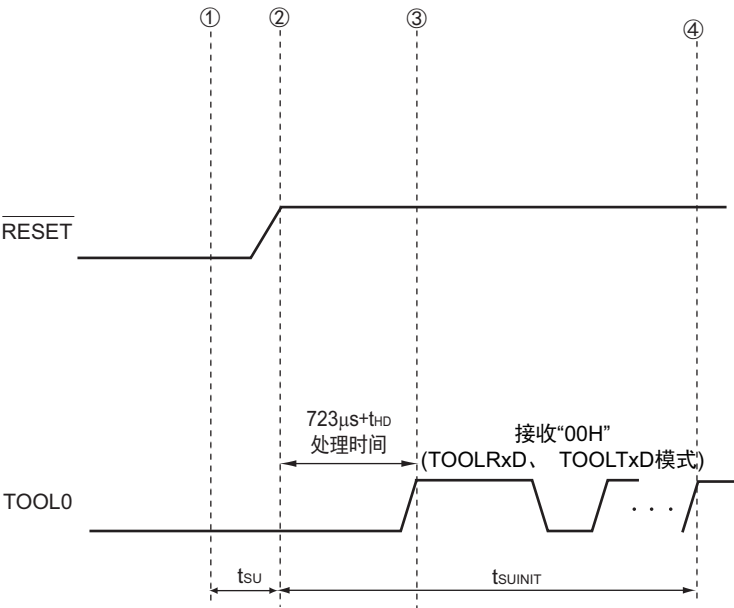
(T_A=−40 ~ +85°C、1.8V ≤ V_{DD} ≤ 5.5V、V_{SS}=0V)

项目	符号	条件	MIN.	TYP.	MAX.	单位
传送速率		串行编程时	115200		1000000	bps

32.11 闪存编程模式的转移时序

($T_A=-40 \sim +85^{\circ}\text{C}$ 、 $1.8\text{V} \leq V_{DD} \leq 5.5\text{V}$ 、 $V_{SS}=0\text{V}$)

项目	符号	条件	MIN.	TYP.	MAX.	单位
从解除外部复位到结束初始设定的通信为止的时间	t_{SUINIT}	在解除外部复位前，解除 POR 和 LVD 的复位。			100	ms
从将 TOOL0 引脚置为低电平到解除外部复位为止的时间	t_{SU}	在解除外部复位前，解除 POR 和 LVD 的复位。	10			μs
在解除外部复位后保持 TOOL0 引脚为低电平的时间 (闪存固件处理时间除外)	t_{HD}	在解除外部复位前，解除 POR 和 LVD 的复位。	1			ms



- ① 给 TOOL0 引脚输入低电平。
- ② 解除外部复位（在此之前需要解除 POR 和 LVD 的复位）。
- ③ 解除 TOOL0 引脚的低电平。
- ④ 通过 UART 接收，完成波特率的设定。

备注 t_{SUINIT} : 在此区间，必须在解除复位后的 100ms 之内完成通信的初始设定。
 t_{SU} : 这是从将 TOOL0 引脚置为低电平到解除外部复位为止的时间。
 t_{HD} : 这是在解除外部复位后保持 TOOL0 引脚为低电平的时间（闪存固件处理时间除外）。

第 33 章 封装尺寸图

R7F0C003M2DFB、 R7F0C004M2DFB

[illegible]

附录 A 修订记录

Rev.	发行日	修订内容	
		页	修订处
1.00	2013.03.26	—	初版发行
2.00	2015.01.23	全文	删除“SCKxx”和“ACK”的上划线。
		全文	将“f _{EXT} ”更能“f _{EXS} ”、“高精度 RTC”更改为“RTC2”、“高精度实时时钟”更改为“实时时钟 2”。
		全文	将“操作速度模式控制寄存器”更改为“副系统时钟提供模式控制寄存器”。
		1、2	替换 1.1。
		3	替换 1.2。
		5	更改 1.4
		6	替换 1.5
		7、8	更改 1.6。
		11	更改 2.1 表 (3/3)
		12	将 2.2 表 (1/2) 中“SCCLK0、SCCLK1”和“SCIO0、SCIO1”的“串行接口”更改为“智能卡接口”。
		21	更改图 3-1、注 1 和注意。
		22	更改图 3-2 的注 1 和注意。
		27	删除表 3-3(2/2) 中的“004EH”和“INTLCD0”。
		29	更改图 3-3 的注意、更改表 3-4 的注意。
		30	更改 3.1.6。
		31	更改图 3-4。
		33	更改 3.2.1(3)。
		34	更改 3.2.2。
		35	更改 3.2.3。
		36	更改 3.2.4 中“1 位操作”说明。
		38 ~ 41	更改表 3-5(2/5) ~ (4/5)。
		43	更改 3.2.5 中“1 位操作”说明。
		44	在表 3-6(1/8) 中追加“F0061H”以及相关说明，更改表 3-6(2/9)、(7/9)。
		52 ~ 54	在图 3-11 ~ 图 3-13、图 3-15、图 3-16 中追加“指令码”，在图 3-17 中追加“寄存器组区”。
		55	更改 3.4.3。
		56、57	在图 3-20、图 3-21 中追加“指令码”。
		58	更改图 3-22、图 3-23。
		59 ~ 67	更改 3.4.7 ~ 3.4.9。
		68	追加 4.2.1 中第四段说明。
		75	更改 4.2.2 说明。
		80	更改 4.2.3 说明。
		84	更改 4.2.4 说明。
		89	更改 4.2.5 说明。
		97	更改 4.2.6 说明。
		98	更改表 4-9 的注 2。
		106	在 4.2.8 中追加第二段说明。
		108	更改 4.2.9 说明。
		113	在 4.2.10 中追加第三段说明。

Rev.	发行日	修订内容	
		页	修订处
2.00	2015.01.23	116	更改 4.3 说明中的注意。
		120	更改图 4-35 的注意。
		121	在图 4-36 中追加注 2，更改注意。
		122	在 4.3.3 说明中追加“通常输出模式”，更改图 4-37 的注意。
		123	更改图 4-38 的注意。
		124	更改 4.3.5。
		125	更改图 4-40 的注意 1 和注意 5。
		126	更改图 4-41 的注意 3。
		128	在图 4-43 中追加注意。
		130	更改 4.3.10 第一段说明和图 4-44。
		132、133	替换 4.4.4。
		134、135	追加 4.5.1、4.5.2 和 4.5.3 说明。
		149、150	更改 5.1(1) 的②、(2) 和 (3)。
		151	更改表 5-1。
		152	替换图 5-1。
		154	更改 5.3 说明，追加注意。
		155	更改图 5-2 的注意。
		158	在 5.3.3 说明下追加注意。
		159	更改图 5-5。
		160 ~ 165	更改 5.3.5 ~ 5.3.8。
		166	追加 5.3.9。
		168	更改图 5-13 的注意。
		171	更改 5.4.3、5.4.4。
		172	更改图 5-15。
		173	更改 5.6.1。
		175	更改 5.6.3。
		176	替换图 5-16。
		178	更改表 5-3(2/5) 中的 (5)。
		179	将表 5-3(3/5) 中 (6) 和 (8) 的“30 μ s”更改为“18 μ s ~ 65 μ s”，追加备注 2。
		180	更改表 5-3(4/5) 的 (9)。
		186 ~ 189	追加 5.7。
		190	更改第六章首页说明。
		193	更改 6.1.2(1) 中的图。
		195	更改表 6-1 中“定时器输入”的结构，更改注。
		196	在表 6-2 下追加备注 1。
		197	更改图 6-1。
		202	更改 6.2.2 第四段说明。
		203	删除 6.3 说明中的注，更改注意。
		204	删除图 6-10 的注，更改注意 1。
		205	更改 6.3.2 第一段说明。
		206、207	更改图 6-11。
		216	更改图 6-16。
		218	更改图 6-18。
		223	更改 6.3.14。
		224	替换 6.3.15。
		231	更改图 6-27 的备注。

Rev.	发行日	修订内容	
		页	修订处
2.00	2015.01.23	232	更改图 6-28 和备注。
		235	在图 6-31 下的说明中追加备注。
		242、243	追加 6.7。
		246	在图 6-44 中追加“0: 16 位定时器”。
		247	在图 6-45(1/2) 中追加“禁止更改 TMRmn 寄存器、TOMmn 位和 TOLmn 位的设定值”。
		249	更改图 6-46。
		251	在图 6-18 中追加“0: 16 位定时器”。
		252	更改图 6-49 中“通道初始设定”的软件操作说明。
		253	更改 6.8.3 第一段说明和图 6-50。
		256	更改图 6-53 中“通道初始设定”的软件操作说明。
		257	更改图 6-54。
		260	更改图 6-57 中“通道初始设定”的软件操作说明。。
		261	更改图 6-58。
		263	在图 6-60 中追加“0: 16 位定时器”。
		264	更改图 6-61 中“通道初始设定”的软件操作说明和“开始运行”的硬件状态说明。
		266	更改图 6-62。
		268	更改图 6-64 和注。
		270、271	更改图 6-66。
		275	更改图 6-69 和注。
		276	更改图 6-70 和注。
		282	更改图 6-74 和注。
		283	更改图 6-75。
		287	更改 7.1。
		288	在表 7-1 中追加“上电复位状态寄存器 (PORSR)”。
		289	更改图 7-1, 追加注。
		590	在 7.3 中追加“上电复位状态寄存器 (PORSR)”、“端口模式寄存器 3 (PM3)”和“端口寄存器 3 (P3)”。
		291	更改图 7-2 的注意 1 和注意 2。
		294	追加 7.3.4。
		295、296	更改图 7-6 和表 7-2。
		297、298	更改图 7-7。
		299	在图 7-8 中追加备注。
		299 ~ 304	更改 7.3.7 ~ 7.3.13 中的“2 个时钟 (f _{RTC})”更改为“2 个 f _{RTC} 时钟”。
		305	更改 7.3.14 第二段说明。
		306	更改表 7-5 下计算式。
		308	替换 7.3.18。
		309	追加 7.4 和 7.4.1 标题, 更改图 7-19 的注 1。
		310	将 7.4.2 中的“副系统时钟”更改为“计数时钟”, 将图 7-20 中的“HALT/STOP 模式”更改为“执行 HALT/STOP 指令”。
		311	在图 7-21 中追加注 1。
		312	在图 7-22 中追加注 1。
		313	更改图 7-23。
		314	更改图 7-24 和注意。
		315	更改 7.4.7。

Rev.	发行日	修订内容	
		页	修订处
2.00	2015.01.23	316	将 7.4.8 中的“F15 为“1””更改为“F15 为“0””，更改“目标校正值的计算方法 1”的计算式。
		318	更改 7.4.8 中“时钟误差校正寄存器（SBUCUD）的 F8～F0 值的计算方法”的计算式，更改 7.4.9 的注。
		326	更改图 9-1。
		327	更改图 9-2 的注意 1 和注意 2。
		330	追加 9.4.1 标题。
		331	追加 9.4.2。
		332	更改 10.1 说明。
		334	更改图 10-2 的注 1，删除注意 3。
		335	在 10.3.2 第三段说明中追加“副系统时钟频率测量电路”，替换 10.3.3。
		336	更改 10.4.1。
		337	在 11.1 中追加第一段说明，更改表 11-1。
		338	更改图 11-1，追加备注。
		340	更改 11.4.1 中的注意 2。
		341	更改表 11-3。
		342	更改表 11-4 的备注。
		343	更改 11.4.4 第一段说明。
		344	更改 12.1 说明。
		345	替换图 12-1。
		347	更改 12.2(8)。
		349	删除图 12-2 的注，更改注意 1。
		351	更改表 12-1。
		352	更改图 12-4。
		353～356	更改表 12-3。
		357	更改图 12-5。
		358	更改图 12-6 的注意 2。
		359、360	更改图 12-7。
		361	更改图 12-8。
		362	更改图 12-9 的注意 1。
		363	更改图 12-11 的注意 6～8，追加注意 10。
		364	在图 12-13 中追加注意 2 和注意 3。
		365	更改 12.3.10 和 12.3.11。
		367	更改图 12-15 和图下第一、第二段说明。
		369～379	更改图 12-17～图 12-27。
		382	追加图 12-31。
		384	更改图 12-34 和图 12-35。
		386、387	更改 12.10(2)、(4)、(5) 和 (6)。
		390	更改 13.1，更改 13.2。
		391	更改 13.3 标题和表 13-1，更改图 13-2 的注意 1。
		396	替换 13.3.5。
		397	更改表 13-2。
		399	更改 13.4.3。
		402	更改 14.1.1。
		405	删除表 14-1 中的“输出控制电路”。
		411	删除图 14-5 的注，更改注意 1。

Rev.	发行日	修订内容	
		页	修订处
2.00	2015.01.23	416	在图 14-8(2/2) 中追加“在 UART 发送时, 设定为 1 个 (SLCmn1、SLCmn0=0、1) 或者 2 个 (SLCmn1、SLCmn0=1、0) 停止位”。
		417	在 14.3.5 中追加第四段说明。
		420	删除图 14-11 的注 1。
		428	追加图 14-18。
		429	更改图 14-19 和图 14-20。
		431	更改 14.3.16 第四段说明。
		432	替换 14.3.17。
		435	更改 14.5。
		436	更改 14.5.1 说明和表中“传送速率”说明。
		437	更改图 14-25。
		439 ~ 443	更改图 14-28 ~ 图 14-32。
		444	更改 14.5.2 说明和表中“传送速率”说明。
		445	更改图 14-33。
		447 ~ 451	更改图 14-36 ~ 图 14-40。
		452	更改 14.5.3 说明和表中“传送速率”说明。
		453	更改图 14-41。
		454	更改图 14-43。
		456 ~ 459	更改图 14-45 ~ 图 14-48。
		460	更改 14.5.4 说明和表中“传送速率”说明。
		461	更改图 14-49。
		462 ~ 467	更改图 14-51 ~ 图 14-56。
		458	更改 14.5.5 说明和注 1。
		469	更改图 14-57。
		470	更改图 14-59。
		472	更改图 14-61、图 14-62。
		474	更改 14.5.6 说明和注 1。
		475	更改图 14-63。
		476	更改图 14-65。
		478 ~ 481	更改图 14-67 ~ 图 14-70。
		482	更改 14.5.7 说明, 删除图 14-71 的注 2, 追加注意 2。
		483 ~ 485	更改图 14-72 ~ 图 14-74。
		494 ~ 499	更改图 14-77 ~ 图 14-83。
		500	将图 14.6.2 表中的“检测 1 位”更改为“附加 1 位”。
		503 ~ 506	更改图 14-86 ~ 图 14-89。
		507	更改 14.6.3。
		520	更改图 14-98 的注 1 和注 3。
		527	在 14.8.1 说明中追加注 2。
		532	在 14.8.2 说明中追加注 2。
		535	在 14.8.3 说明中追加注 2。
		542	追加图 14-115。
		544	更改图 15-1。
		549	删除图 15-5 的注, 更改注意 1。
		563	在图 15-6(3/4) 中追加注 2。
		554	在图 15-6(4/4) 中追加注。
		559、560	更改图 15-9。
		561	更改图 15-11 的备注 1。

Rev.	发行日	修订内容	
		页	修订处
2.00	2015.01.23	564、565	更改 15.4.2。
		578、579、580	在图 15-22、图 15-23 和图 15-24 中追加“f _{MCK} ”。
		579	更改图 15-23 下说明。
		581、583	更改 15.5.14(1) 和 (2) 说明。
		584	更改 15.5.15(3) 中的④。
		588	更改图 15-29(2/3)。
		589	更改图 15-29(3/3)。
		591	更改图 15-30。
		663	更改 17.1 说明和表 17-2。
		667	更改图 17-2。
		668	更改表 17-4。
		669	更改图 17-3。
		671	更改 17.3.3。
		672	更改图 17-5。
		673	更改图 17-6 的注 5。
		675	更改图 17-9。
		677	更改图 17-10。
		686	更改 17.5 说明和注意。
		688	更改 17.6。
		694	删除 17.9.1。
		702	更改图 17-27。
		722	替换图 18-1。
		727	追加 18.3.1 第三段说明。
		728	在图 18-5 中追加注 1。
		729	更改 18.4.1 中的⑦。
		730	更改 18.4.2 中的⑦。
		731	更改 18.4.3 中的⑪。
		732	更改 18.4.4 中的⑬。
		734	更改 18.4.5 中的⑬。
		735	更改 19.1。
		748	更改图 19-9。
		753	更改 19.6(4)。
		760	更改表 20-2(2/3) 的注。
		768	更改 20.3.5 第二段说明。
		771	更改图 20-9。
		772	更改表 20-5。
		777	更改 21.2, 在 21.3.1(1) 中追加注意。
		778、779	更改表 21-1。
		781、782	更改图 21-2。
		782 ~ 786	更改 21.3.2。
		787 ~ 789	更改 21.3.3。
		790	更改第 22 章首页说明。
		792、793	更改 22.1。
		794	更改 22.2。
		796	更改图 22-4 复位后的值和注意 2。
		797	在表 22-3 下追加说明。

Rev.	发行日	修订内容	
		页	修订处
2.00	2015.01.23	796	追加图 22-6。
		799	更改 23.1。
		801	更改图 23-2。
		804	更改 24.1。
		805	更改图 24-1。
		806	更改图 24-2。
		807	更改图 24-3 的注 2 和注意。
		808、809	更改图 24-4。
		810 ~ 817	更改 24.4.1 ~ 24.4.3。
		818、819	更改 24.5(1) 和 (3)。
		820	更改 25.1(6) ~ (8) 和备注。
		823	更改图 25-3。
		826	更改图 25-7 的注意和备注。
		827	追加图 25-8, 更改图 25-9。
		828	更改图 25-10。
		829	更改图 25-11。
		830	更改图 25-12 的备注。
		831 ~ 836	更改 25.3.7 ~ 25.3.9。
		838	更改 27.1 说明和 27.1.1。
		841、842	更改图 27-2。
		843	更改图 27-3。
		847	更改第 28 章首页图下的说明。
		848	更改表 28-1。
		850	更改表 28-2。
		851	更改 28.2 标题。
		852	更改表 28-3, 更改 28.3.1。
		854	更改 28.4.2。
		856	更改表 28-6。
		856、857	更改 28.4.4。
		858	更改 28.5 说明。
		861	更改图 28-10。
		862	更改 28.5.3。
		863	追加 28.6。
		865	更改 29.1。
		872	更改表 31-2。
		874 ~ 890	更改表 31-5 的注 2。
		891	在 32.1 表 (1/3) 下追加备注 3。
		892、893	更改 32.1 表 (2/3)、(3/3)。更改 32.2.1。
		894	更改 32.2.2。
		895 ~ 899	更改 32.3.1。
		900 ~ 905	更改 32.3.2。
		906 ~ 909	更改 32.4。
		910 ~ 928	更改 32.5.1。
		929 ~ 931	更改 32.5.2。
		933 ~ 938	更改 32.6。
		942、943	更改 32.8 ~ 32.11。

R7F0C003M2DFB、R7F0C004M2DFB
用户手册 硬件篇

Publication Date: Rev.2.00 Jan 23, 2015

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.

2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited

9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

Renesas Electronics Europe Limited

Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K.
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH

Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.

Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.

Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited

Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.

13F, No. 363, Fu Shing North Road, Taipei 10543, 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 1207, 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 India Pvt. Ltd.

No.777C, 100 Feet Road, HAL II Stage, Indiranagar, Bangalore, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.

12F., 234 Teheran-ro, Gangnam-Gu, Seoul, 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

R7F0C003M2DFB、 R7F0C004M2DFB



瑞萨电子株式会社

R01UH0393CJ0200