

致尊敬的顾客

关于产品目录等资料中的旧公司名称

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

瑞萨电子公司网址：<http://www.renesas.com>

2010年4月1日
瑞萨电子公司

【发行】瑞萨电子公司（<http://www.renesas.com>）

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

Notice

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

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

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

概要

4571 群是采用 CMOS 工艺开发的独创 4 位单片机，以具有简单、高速指令体系的 4500 系列 CPU 为核心，内置 3 个 8 位定时器（带重加载寄存器）、中断功能和低电压检测电路。

4571 群有多种不同的机种。

详细内容请参照下表。

特点

- 最小指令执行时间.....0.5μs
(在振荡频率为 6MHz、through-mode 时)
- 电源电压.....1.8~5.5V
(根据运行模式和振荡频率而不同)
- 定时器
 - 定时器 1.....8 位
(带重加载寄存器和载波输出自动控制功能)
 - 定时器 2.....8 位
(带重加载寄存器)
 - 定时器 3.....8 位
(带 2 个重加载寄存器和载波发生功能)
- 中断功能.....6 个源
- 键唤醒功能.....12 个引脚
- 输入/输出端口.....17 个
- 输出端口.....1 个
- 输入端口.....1 个
- 低电压检测电路
 - 复位发生.....典型值 1.65V(在 Ta=25°C 时)
 - 复位解除.....典型值 1.75V(在 Ta=25°C 时)
 - 中断发生.....典型值 1.85V(在 Ta=25°C 时)
- 上电复位电路
- 看门狗定时器
- 时钟发生电路（陶瓷谐振电路）

应用

遥控器

表 1. 支持产品一览表

产品型号	ROM 容量 (×10 位)	RAM 容量 (×4 位)	封装	ROM 种类
M34571G4FP (注 1)	4096 字	128 字	PRSP0024GA-A	QzROM
M34571G4-XXXFP	4096 字	128 字	PRSP0024GA-A	QzROM
M34571G6FP (注 1)	6144 字	128 字	PRSP0024GA-A	QzROM
M34571G6-XXXFP	6144 字	128 字	PRSP0024GA-A	QzROM
M34571GDFF (注 1)	16384 字	128 字	PRSP0024GA-A	QzROM
M34571GD-XXXFP	16384 字	128 字	PRSP0024GA-A	QzROM

注 1. 空白出货产品

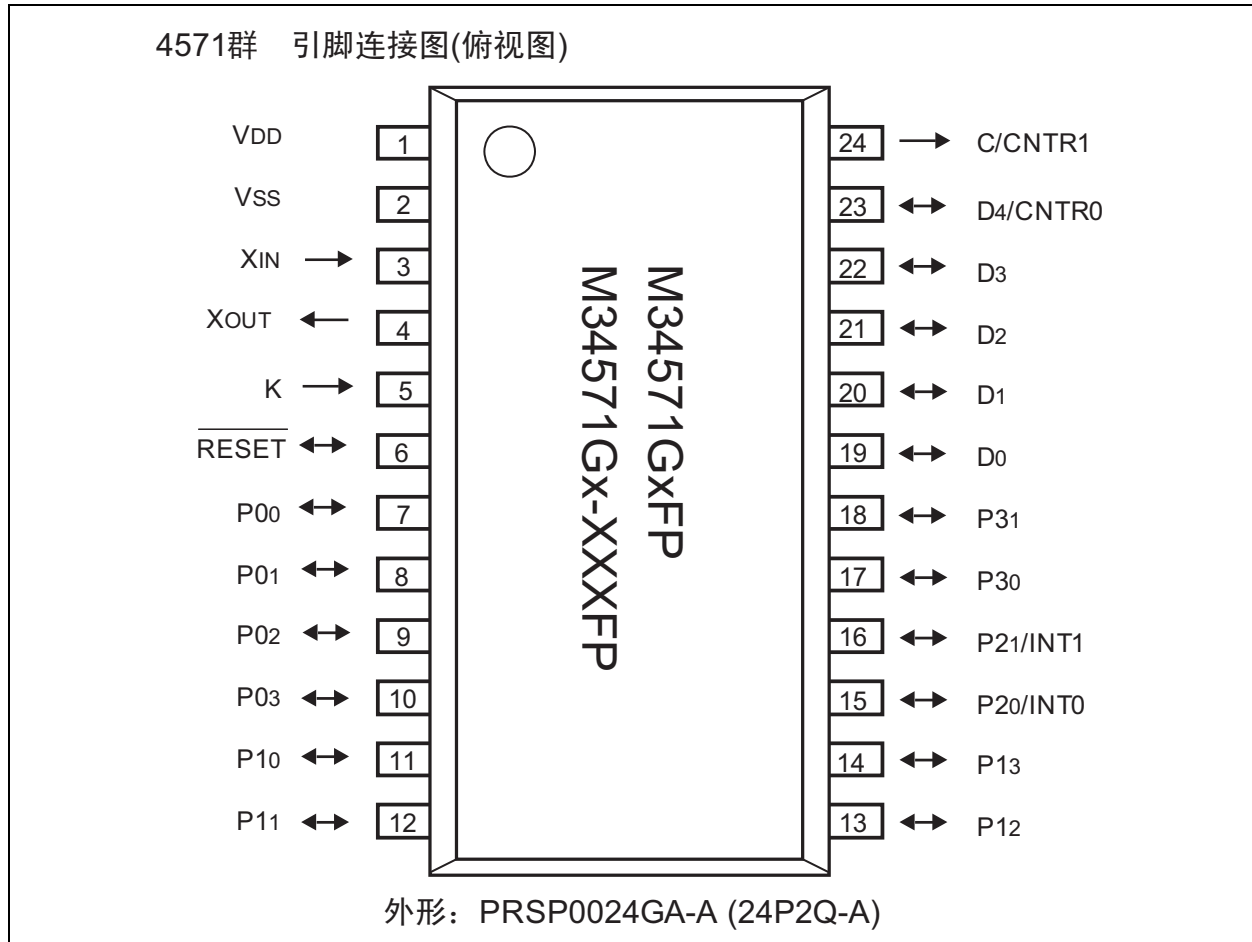


图 1. 引脚连接图 (PRSP0024GA-A 封装型)

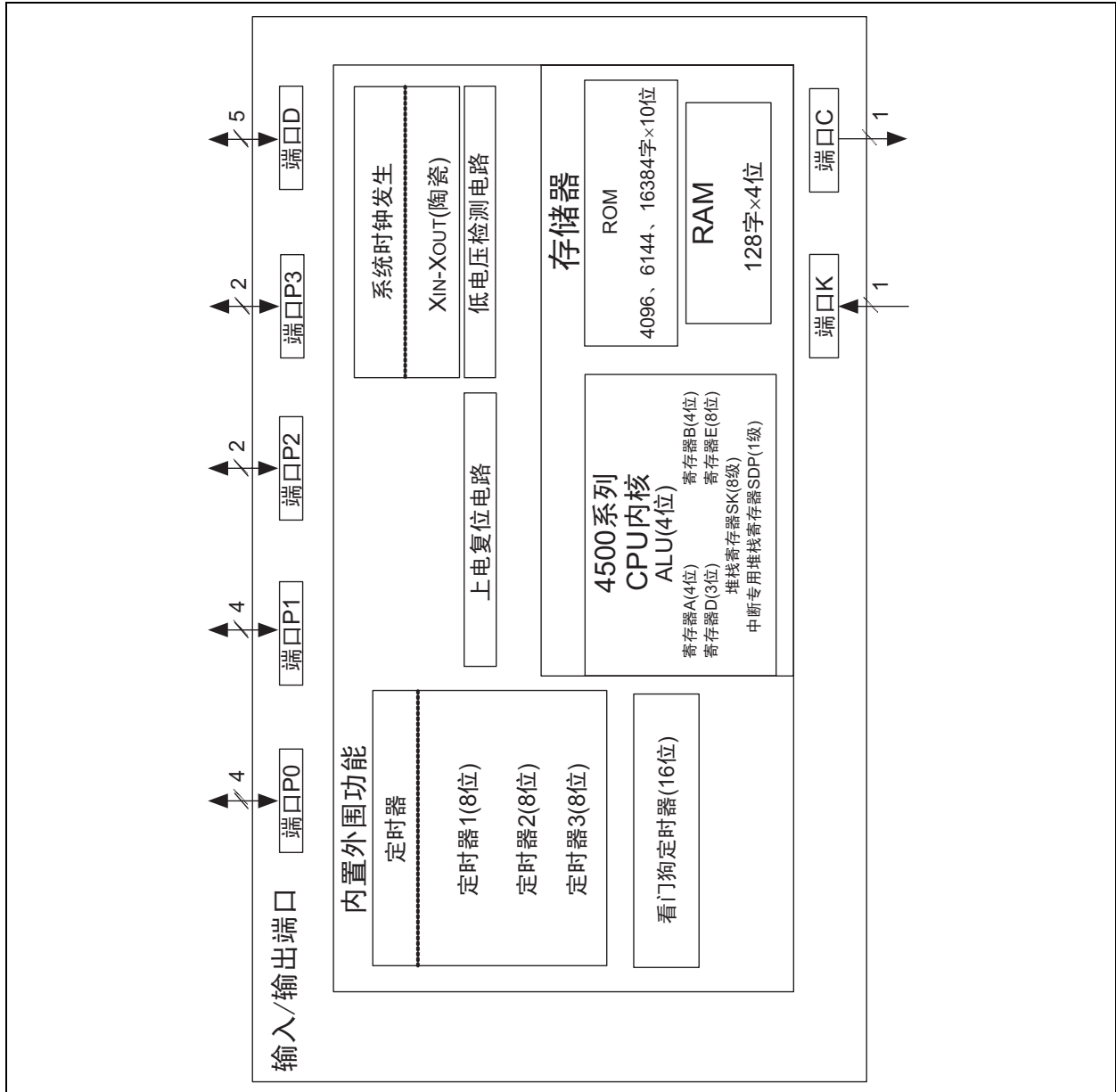


图 2. 4571 群功能框图 (PRSP0024GA-A 封装型)

性能概要

表 2. 性能概要

项 目		性 能	
基本指令数	M34571G4/G6	126	
	M34571GD	128	
最小指令执行时间		0.5 μ s (振荡频率 6MHz: through-mode 时)	
存储器容量	ROM	M34571G4	4096 字 \times 10 位
		M34571G6	6144 字 \times 10 位
		M34571GD	16384 字 \times 10 位
	RAM		128 字 \times 4 位
输入/输出端口	D0~D4	输入/输出 (输入为跳越判别)	1 位 \times 5 可通过软件转换端口 D0~D3 的输出形式 端口 D4 可复用为 CNTR0 引脚
	P00~P03	输入/输出	4 位 \times 1 可通过软件转换上拉功能、键唤醒功能
	P10~P13	输入/输出	4 位 \times 1 可通过软件转换上拉功能、键唤醒功能
	P20、P21	输入/输出	2 位 \times 1 可通过软件转换上拉功能、键唤醒功能 端口 P20、P21 分别可复用为 INT0、INT1 引脚
	P30、P31	输入/输出	2 位 \times 1 可通过软件转换输出形式
	C	输出	1 位 \times 1 (CMOS 输出专用) 可复用为 CNTR1 引脚
	K	输入	1 位 \times 1 (输入专用) 可通过软件转换键唤醒功能
	CNTR0	定时器输入/输出	1 位 \times 1, 可复用为端口 D4
	CNTR1	定时器输出	1 位 \times 1, 可复用为端口 C
	INT0、INT1	中断输入	1 位 \times 2, INT0、INT1 引脚分别可复用为端口 P20、P21
定时器	定时器 1	8 位定时器/事件计数器、带重加载寄存器和载波输出自动控制功能	
	定时器 2	8 位定时器、带重加载寄存器	
	定时器 3	8 位定时器、带 2 个重加载寄存器和载波发生功能	
	看门狗定时器	16 位定时器、固定分频 (用于监视的定时器)	
上电复位		内置	
低电压检测电路	复位发生	典型值 1.65V (当 Ta=25°C 时)	
	复位解除	典型值 1.75V (当 Ta=25°C 时)	
	中断发生	典型值 1.85V (当 Ta=25°C 时)	
中断	源	6 个源 (外部 \times 2、定时器 \times 3、低电压检测电路)	
	嵌套	1 级	
子程序嵌套		8 级	
元件结构		CMOS 硅栅	
封装		24 引脚塑模 SSOP (PRSP0024GA-A)	
工作环境温度		-20~85°C	
电源电压		1.8~5.5V (根据运行模式和振荡频率而不同)	
消耗电流	在 CPU 运行时	0.3mA (Ta=25°C、V _{DD} =3.0V、f(XIN)=4MHz、f(STCK)=f(XIN)/8)	
	在 RAM 备份时	0.1 μ A (Ta=25°C、输出晶体管为截止状态)	

引脚的功能说明

表 3. 引脚的功能说明

引脚名	名称	输入/输出	功能
VDD	电源	—	是供给正电源电压的引脚。
VSS	接地	—	是 GND 引脚。
XIN	主时钟输入	输入	是主时钟发生电路的输入/输出引脚。在 XIN 引脚和 XOUT 引脚之间连接陶瓷谐振器，在 XIN 引脚和 XOUT 引脚之内置了反馈电阻。
XOUT	主时钟输出	输出	
RESET	复位输入/输出	输入/输出	是复位脉冲的输入/输出引脚。在通过 SRST 指令、看门狗定时器以及内置上电复位电路进行系统复位时，输出“L”电平。输出形式为 N 沟道漏极开路。
D0~D4	输入/输出端口 D (输入为跳越判别)	输入/输出	每个引脚具有 1 位输入/输出功能。端口 D0~D3 的输出形式可通过软件转换 N 沟道漏极开路或者 CMOS。如果选择 N 沟道漏极开路的输出形式并将输出锁存器置“1”，就变为可输入状态。端口 D4 的输出形式是 N 沟道漏极开路。如果将输出锁存器置“1”，就变为可输入状态。端口 D4 可复用为 CNTR0 引脚。
P00~P03	输入/输出端口 P0	输入/输出	具有作为端口的 4 位输入/输出功能。输出形式是 N 沟道漏极开路。如果将输出锁存器置“1”，就变为可输入状态。内置了可通过软件转换的键唤醒功能和上拉功能。
P10~P13	输入/输出端口 P1	输入/输出	具有作为端口的 4 位输入/输出功能。输出形式是 N 沟道漏极开路。如果将输出锁存器置“1”，就变为可输入状态。内置了可通过软件转换的键唤醒功能和上拉功能。
P20、P21	输入/输出端口 P2	输入/输出	具有作为端口的 2 位输入/输出功能。输出形式是 N 沟道漏极开路。如果将输出锁存器置“1”，就变为可输入状态。内置了可通过软件转换的键唤醒功能和上拉功能。端口 P20、P21 分别可复用为 INT0、INT1 引脚。
P30、P31	输入/输出端口 P3	输入/输出	具有作为端口的 2 位输入/输出功能。输出形式可通过软件选择为 N 沟道漏极开路或者 CMOS。如果选择 N 沟道漏极开路的输出形式，并将输出锁存器置“1”，就变为可输入状态。
C	输出端口 C	输出	具有作为端口的 1 位输出功能。输出形式是 CMOS。 端口 C 可复用为 CNTR1 引脚。
K	输入端口 K	输入	具有作为端口的 1 位输入功能。 具有可通过软件转换的键唤醒功能。 使用键矩阵输入时，请在外部连接上拉电阻。
CNTR0	定时器输入/输出	输入/输出	具有用于定时器 1 事件计数的时钟输入功能、定时器 1 或者定时器 2 下溢的 2 分频信号输出功能。 CNTR0 引脚可复用为端口 D4。
CNTR1	定时器输入/输出	输入/输出	具有定时器 3 生成的 PWM 信号输出功能。CNTR1 引脚可复用为端口 C。
INT0 INT1	中断输入	输入	具有外部中断接受功能以及可通过软件转换的键唤醒功能。 INT0、INT1 引脚分别可复用为端口 P20、P21。

表 4. 多功能一览表

引脚名	多功能	引脚名	多功能	引脚名	多功能	引脚名	多功能
C	CNTR1	P20	INT0	CNTR1	C	INT0	P20
D4	CNTR0	P21	INT1	CNTR0	D4	INT1	P21

注 1. 上述以外的引脚为单功能。

注 2. 即使在使用 CNTR0 引脚输出功能的情况下，端口 D4 的输入功能也有效。即使在使用 CNTR0 引脚输入功能的情况下，端口 D4 的输入/输出功能也有效。因为 CNTR0 引脚和端口 D4 的输入阈值不同，所以在使用两者的输入时必须注意。

注 3. 即使在使用 CNTR1 引脚输出功能的情况下，端口 C 的“H”输出功能也有效。

注 4. 即使在使用 INT0 引脚的情况下，端口 P20 的输入/输出功能也有效。因为 INT0 引脚和端口 P20 的输入阈值不同，所以在使用两者的输入时必须注意。

注 5. 即使在使用 INT1 引脚的情况下，端口 P21 的输入/输出功能也有效。因为 INT1 引脚和端口 P21 的输入阈值不同，所以在使用两者的输入时必须注意。

表 5. 端口功能一览表

端口名	引脚名	输入/输出	输出形式	输入/输出单位	控制指令	控制寄存器	特 记 事 项
端口 D	D0~D3	输入/输出 (5 个)	N 沟道漏极开路 /CMOS	1 位	SD、RD SZD CLD	FR1	带输出形式选择功能 (可通过软件转换)
	D4/CNTR0		N 沟道漏极开路			W1 W2 W5	—
端口 P0	P00 P01 P02 P03	输入/输出 (4 个)	N 沟道漏极开路	4 位	OP0A IAP0	PU0 K0	带上拉和键唤醒功能 (可通过软件转换)
端口 P1	P10 P11 P12 P13	输入/输出 (4 个)	N 沟道漏极开路	4 位	OP1A IAP1	PU1 K1	带上拉和键唤醒功能 (可通过软件转换)
端口 P2	P20/INT0 P21/INT1	输入/输出 (2 个)	N 沟道漏极开路	2 位	OP2A IAP2	PU2 K2、I1、I2、L1	带上拉和键唤醒功能 (可通过软件转换)
端口 P3	P30 P31	输入/输出 (2 个)	N 沟道漏极开路 /CMOS	2 位	OP3A IAP3	FR0	带输出形式选择功能 (可通过软件转换)
端口 C	C/CNTR1	输出 (1 个)	CMOS	1 位	RCP SCP	W1、W3、W5	—
端口 K	K	输入 (1 个)	—	1 位	IAK	K2	带键唤醒功能 (可通过软件转换)

时钟和周期的定义

● 运行源时钟

它是运行源的时钟，在本产品中能使用以下时钟：

- 外接陶瓷谐振器的时钟（f(XIN)）
- 外部输入的时钟（f(XIN)）

● 系统时钟（STCK）

系统时钟是控制本产品的基本时钟。

能通过时钟控制寄存器 MR 的设定，选择如表 6 所示的系统时钟（STCK）。

表 6. 系统时钟的选择

时钟控制寄存器 MR		系统时钟	运行模式名
MR ₃	MR ₂		
1	1	$f(\text{STCK})=f(\text{XIN})/8$	8 分频模式
1	0	$f(\text{STCK})=f(\text{XIN})/4$	4 分频模式
0	1	$f(\text{STCK})=f(\text{XIN})/2$	2 分频模式
0	0	$f(\text{STCK})=f(\text{XIN})$	through-mode

注. 在复位解除后，选择 8 分频模式。

● 机器周期

机器周期是执行指令所需的基准周期。

● 指令时钟（INSTCK）

指令时钟是控制 CPU 的基准时钟。

指令时钟（INSTCK）是对系统时钟（STCK）进行 3 分频后的信号，1 个周期生成 1 个机器周期。

表 7. 不用引脚的处理

引脚名	处理方法	使用条件				
		输出形式	上拉晶体管	键唤醒	输出锁存器的值	其它
D0~D3、 P30、P31	开路	N 沟道漏极开路	—	—	0/1	(注 1)
		CMOS	—	—	0/1	—
	连接到 Vss	N 沟道漏极开路	—	—	0/1	—
		CMOS	—	—	0	—
	连接到 VDD	N 沟道漏极开路	—	—	1	—
		CMOS	—	—	1	—
D4/CNTR0	开路	N 沟道漏极开路	—	—	0/1	(注 1、2)
	连接到 Vss	N 沟道漏极开路	—	—	0/1	(注 2)
	连接到 VDD	N 沟道漏极开路	—	—	1	(注 2)
P00~P03、 P10~P13	开路	N 沟道漏极开路	OFF	无效	0/1	(注 1)
			ON	无效	1	—
	连接到 Vss	N 沟道漏极开路	OFF	无效	0/1	—
	连接到 VDD	N 沟道漏极开路	ON/OFF	有效/无效	1	—
P20/INT0、 P21/INT1	开路	N 沟道漏极开路	OFF	无效	0/1	(注 1、3)
			ON	无效	1	(注 3)
	连接到 Vss	N 沟道漏极开路	OFF	无效	0/1	(注 3)
	连接到 VDD	N 沟道漏极开路	ON/OFF	有效/无效	1	(注 3)
C/CNTR1	开路	CMOS	—	—	0/1	—
	连接到 Vss	CMOS	—	—	0	(注 4)
K	连接到 Vss	—	—	无效	—	—
	连接到 VDD	—	—	有效/无效	—	—

注 1. 在输出锁存器的值为“1”的情况下，如果执行端口输入指令（SZD、IAP0、IAP1、IAP2、IAP3），那么在指令执行周期内就会因穿透电流而引起电源电流增加。

注 2. 不要给定定时器 1 的计数源选择 CNTR0 输入（W1₁ 和 W1₀ ≠ “11”）。

注 3. 设置 INT0 或 INT1 引脚为输入禁止（I1₃ = “0”、I2₃ = “0”）。

注 4. 设置 CNTR1 引脚为输出无效（W3₃ = “0”）。

（连接到 VDD 引脚和 Vss 引脚时的注意事项）

- 为了避免噪声的传播，请尽量以短距离粗线处理未使用的引脚。

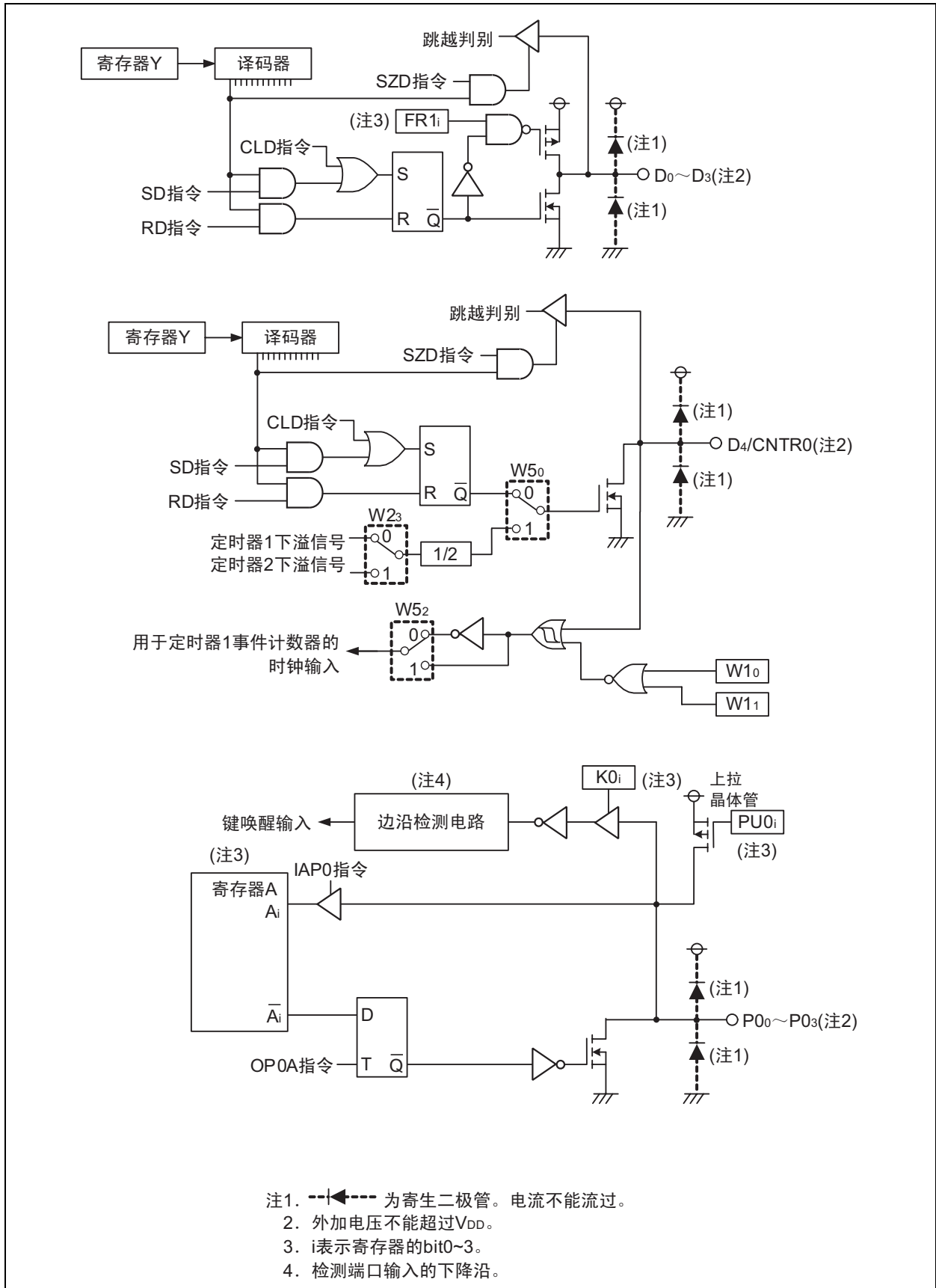


图 3. 端口框图 (1)

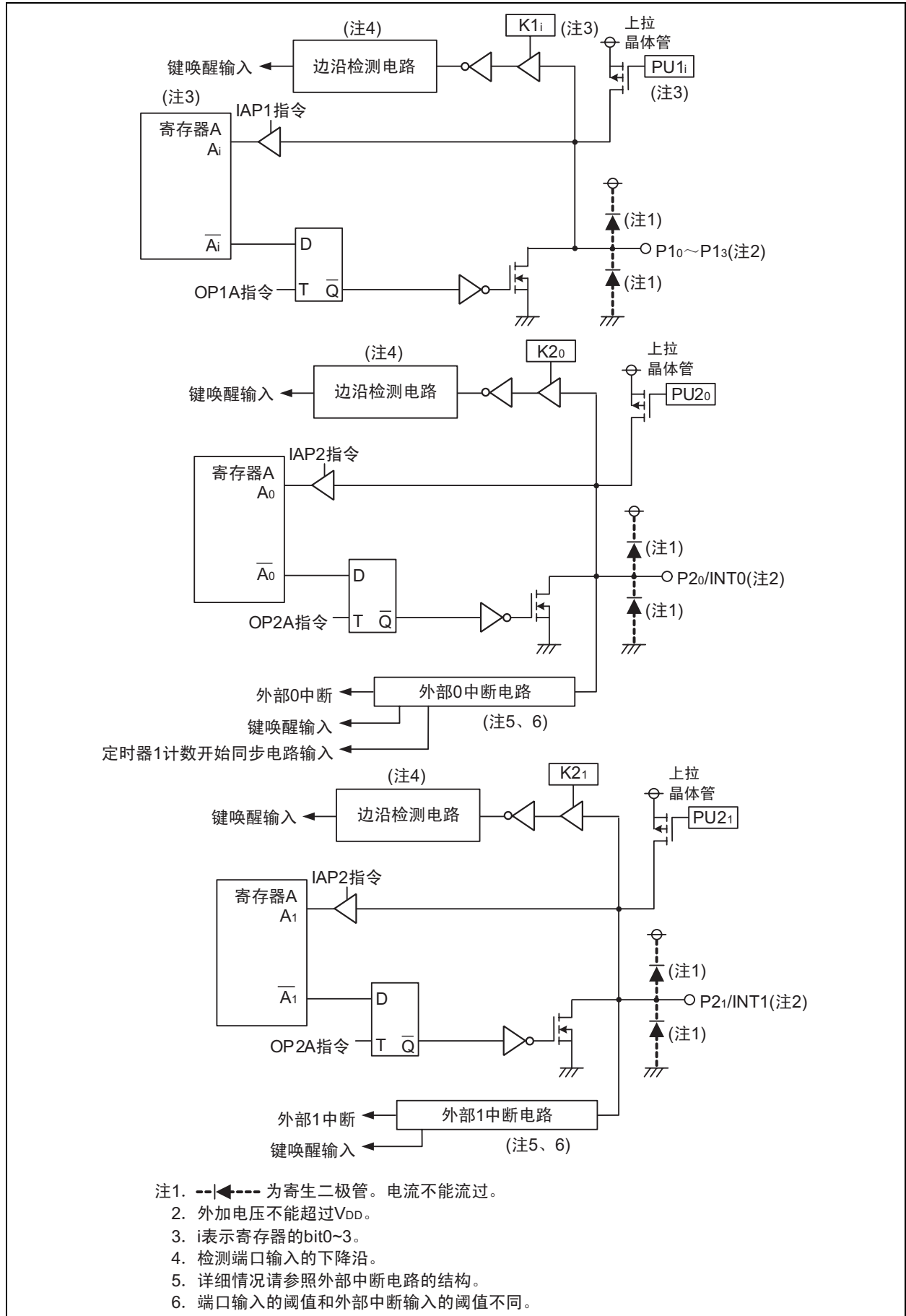


图4. 端口框图(2)

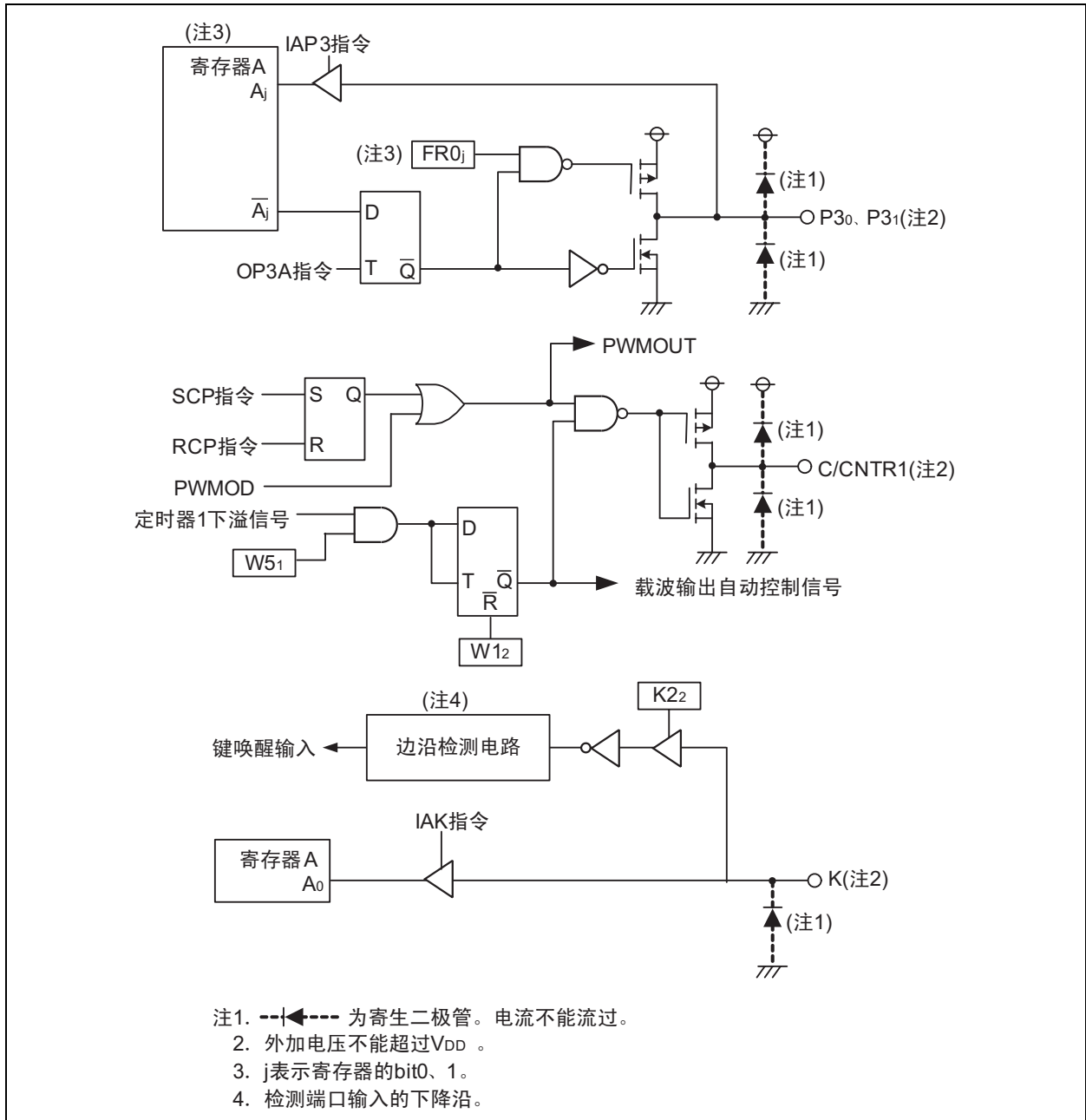


图 5. 端口框图 (3)

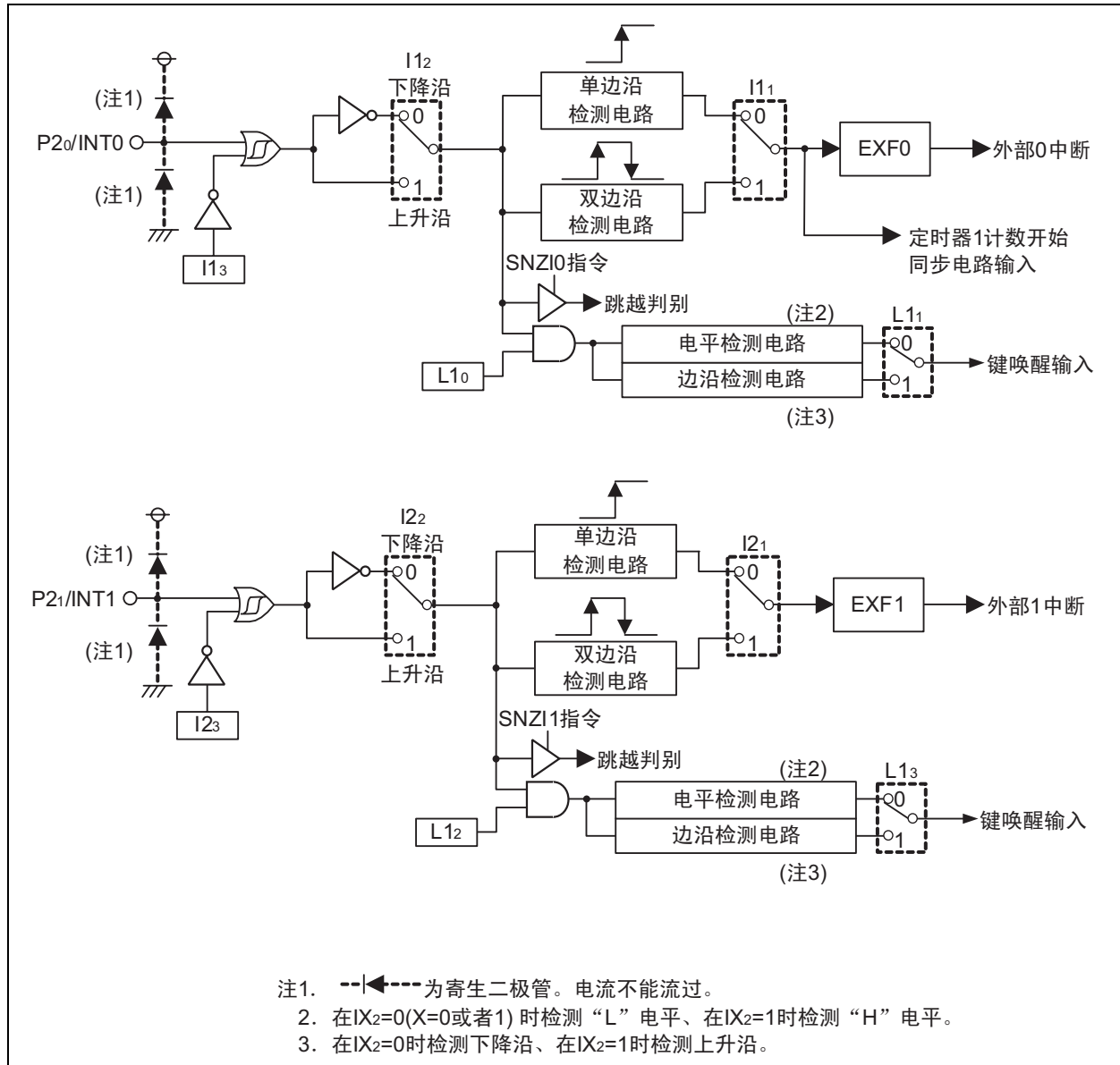


图 6. 端口框图 (4)

功能块的运行说明

CPU

1. 4 位逻辑运算单元 (ALU)

ALU 是进行 4 位加法、比较、逻辑与、逻辑或和位处理等运算的单元。

2. 寄存器 A 和进位标志 (CY)

寄存器 A 是以运算、传送、交换和输入/输出等数据处理为核心的 4 位寄存器。

如果在执行 AMC 指令时发生进位，就将标志 CY 置“1”（图 7）。

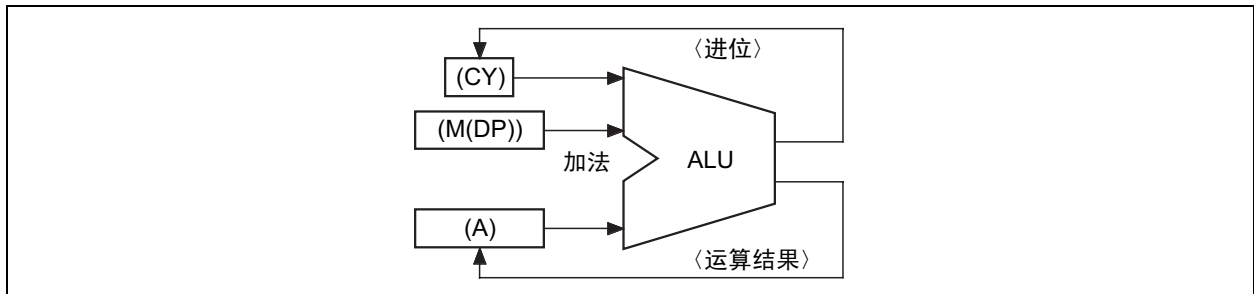


图 7. 执行 AMC 指令的例子

即使执行 A_n 指令和 AM 指令，标志 CY 的内容也不变。另外，通过执行 RAR 指令， A_0 的值被保存到标志 CY（图 8）。

标志 CY 通过 SC 指令置“1”，通过 RC 指令清“0”。

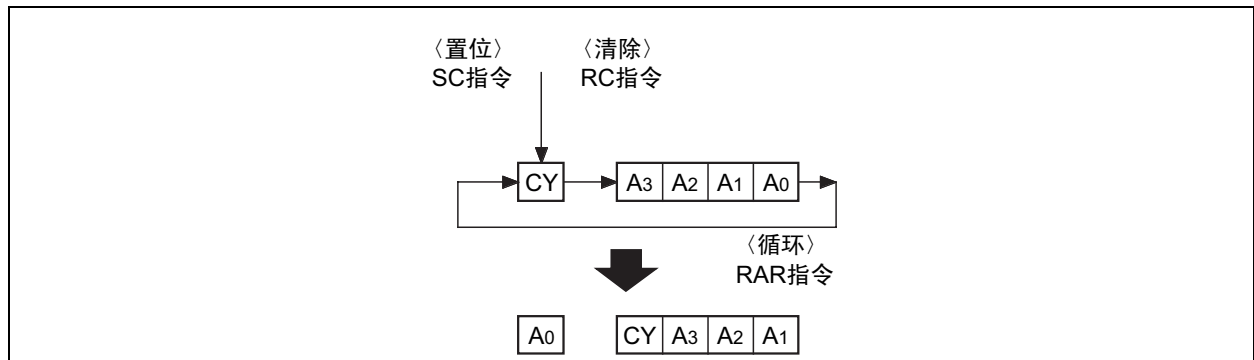


图 8. 执行 RAR 指令的例子

3. 寄存器 B 和 E

寄存器 B 由 4 位构成，用于暂时保存 4 位数据或者和寄存器 A 组合后进行 8 位数据的传送。

寄存器 E 由 8 位构成，用于将寄存器 B 作为高 4 位、寄存器 A 作为低 4 位的 8 位数据的传送（图 9）。

因为寄存器 E 在复位解除后或者从 RAM 备份返回后其值不定，所以必须进行初始设定。

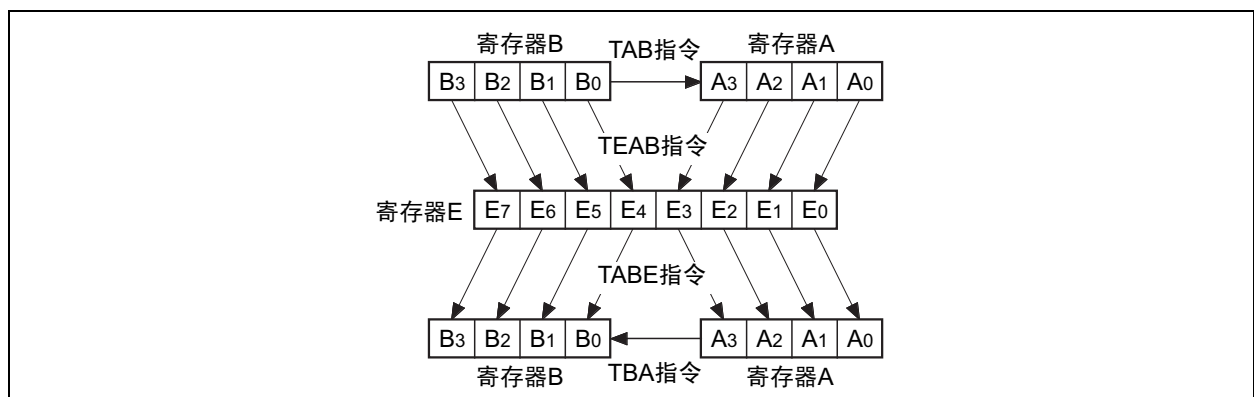


图 9. 寄存器 A、B 和寄存器 E

4. 寄存器 D

寄存器 D 由 3 位构成，和寄存器 A 组合后保存 7 位地址，在执行 TABP p 指令、BLA p 指令和 BMLA p 指令时作为指定页内的指针使用（图 10）。

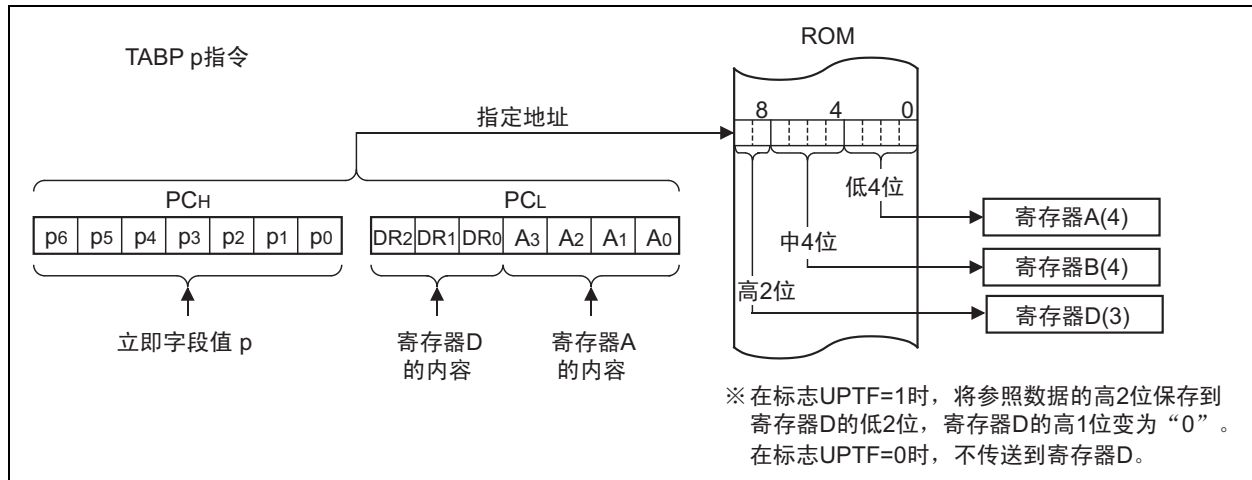


图 10. 执行 TABP p 指令的例子

另外，如果在标志 UPTF 为“1”时执行 TABP p 指令，就将 ROM 内参照数据的高 2 位保存到寄存器 D 的低 2 位，寄存器 D 的高 1 位变为“0”。在标志 UPTF 为“0”时，即使执行 TABP p 指令，寄存器 D 的内容也不变。

标志 UPTF 通过 SUPT 指令置“1”，通过 RUPT 指令清“0”。标志 UPTF 的初始值为“0”。

因为寄存器 D 在复位解除后或者从 RAM 备份返回后其值不定，所以必须进行初始设定。

5. 堆栈寄存器 SK 和堆栈指针 (SP)

寄存器 SK 是在执行中断处理程序的转移、子程序调用或者表参照指令 (TABP p) 时使用的 8 段 14 位寄存器。在返回原程序前，暂时保存转移前程序计数器的内容。

因为寄存器 SK 由 8 段构成，所以最多能使用 8 级子程序。但是，在使用中断处理程序以及执行表参照指令时，因为分别使用 1 段寄存器 SK，所以必须注意：在同时使用这些处理时，其总计不能超过 8 级。如果超过 8 级，将破坏寄存器 SK 的内容。

另外，由 3 位构成的堆栈指针 (SP) 自动指定寄存器 SK 的嵌套，堆栈指针的内容能通过 TASP 指令传送到寄存器 A。寄存器 SK 的结构如图 11、子程序调用时的运行例子如图 12 所示。

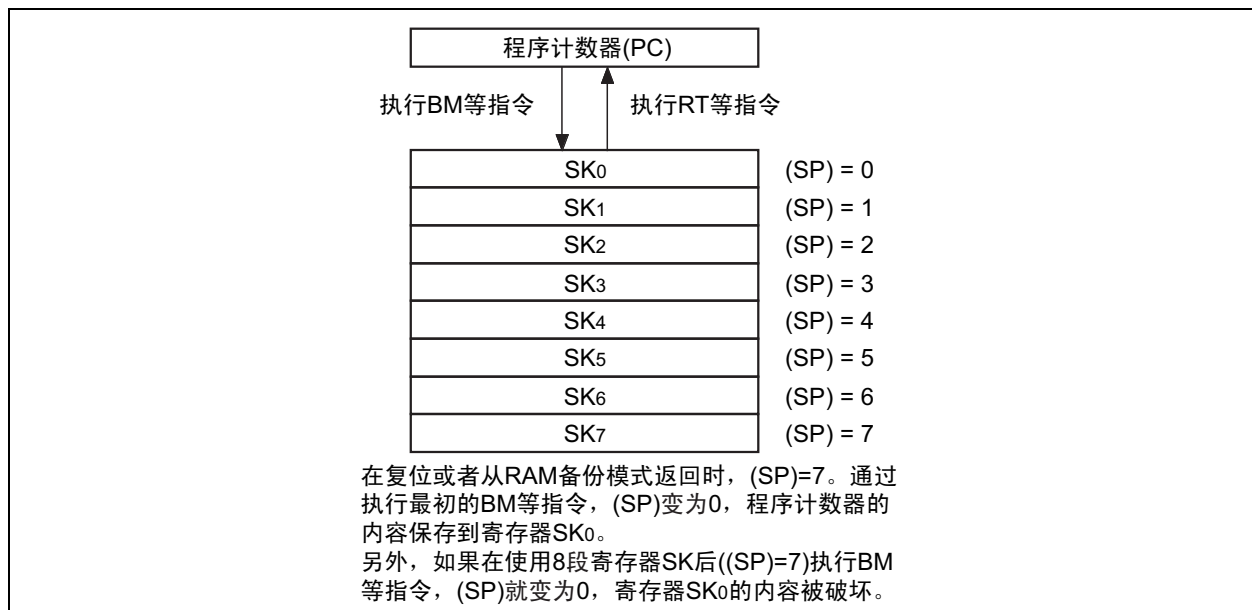


图 11. 堆栈寄存器 SK 的结构

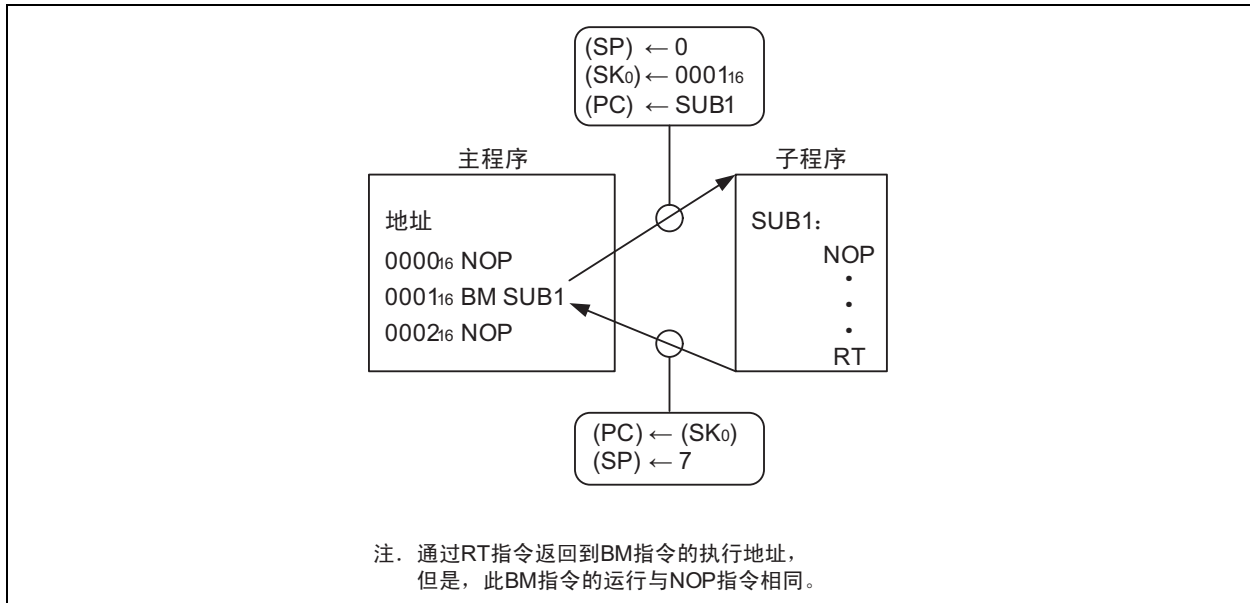


图 12. 子程序调用时的运行例子

6. 中断专用堆栈寄存器 SDP

寄存器 SDP 是在发生中断时将发生中断前的数据指针、进位标志 (CY)、跳越标志和寄存器 A、B 的内容暂时保存到返回原程序为止。寄存器 SDP 由 1 段构成。

寄存器 SDP 和上述的寄存器 SK 不同, 不能在执行子程序调用指令和表参照指令时使用。

7. 跳越标志

跳越标志是对条件跳越指令以及连续记述跳越指令的跳越判定进行控制的标志。如果发生中断, 跳越标志的内容就自动保存到寄存器 SDP, 并保持跳越条件。

8. 程序计数器 (PC)

程序计数器是指定 ROM 地址 (页和地址) 的计数器, 用于决定保存在 ROM 中的指令的读顺序。

程序计数器是 2 进制计数器, 每执行一条指令就将指令字节数+1。

但是, 在执行转移指令、子程序调用指令、返回指令以及表参照指令 (TABP p) 时, 程序计数器的值为被指定的地址值。

程序计数器分为指定 ROM 页的 PCH (最高位~bit7) 和指定页内地址的 PCL (bit6~bit0), 当到达各页的最后地址 (地址 127) 时, 就指定下一页的地址 0 (图 13)。

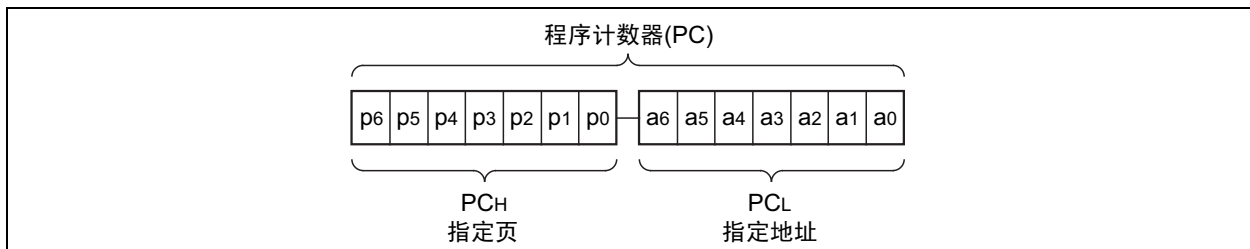


图 13. 程序计数器 (PC) 的结构

另外, 必须注意: PCH 指定的页不能超过内部 ROM 的最后页。

9. 数据指针 (DP)

数据指针是指定 RAM 地址的指针，由寄存器 Z、X、Y 构成 (图 14)。其中，寄存器 Z 指定 RAM 的文件组、寄存器 X 指定 RAM 的文件、寄存器 Y 指定 RAM 的位数。

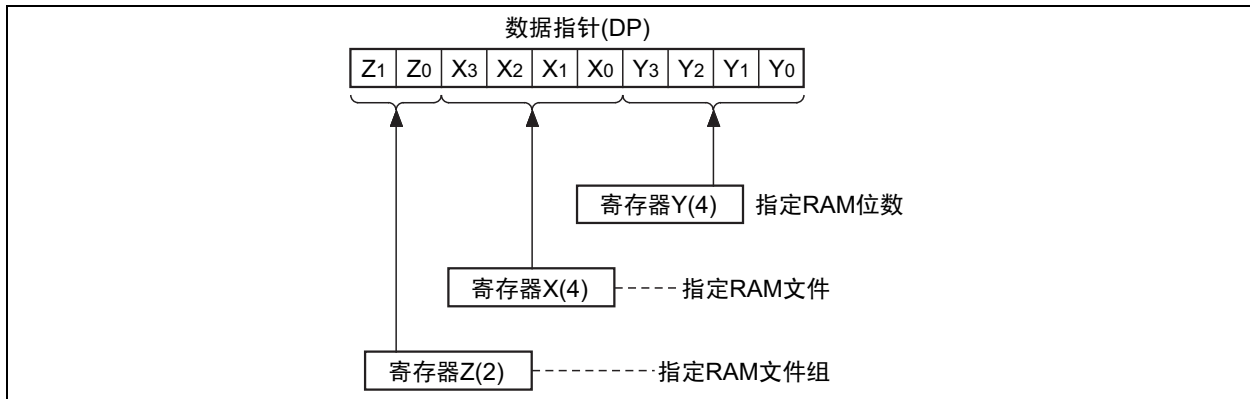


图 14. 数据指针 (DP) 的结构

另外，寄存器 Y 也用于指定端口 D 的位的位置。在使用端口 D 时，必须给寄存器 Y 设定端口 D 的位（引脚位置），并执行 SD、RD 和 SZD 指令。

执行 SD 指令的例子如图 15 所示。

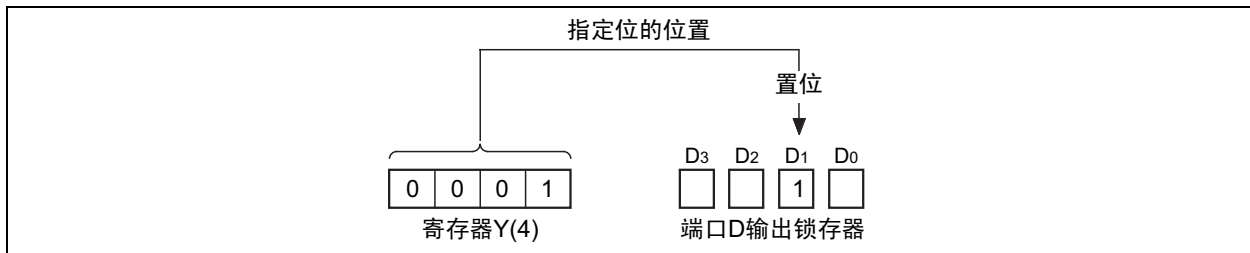


图 15. 执行 SD 指令的例子

● 注意事项

因为数据指针的寄存器 Z 在复位解除后其值不定，所以必须进行初始设定。

另外，寄存器 Z、X、Y 在 RAM 备份时处于不定，必须在从 RAM 备份返回后重新设定这些寄存器。

程序存储器 (ROM)

程序存储器的 1 字由 10 位构成，以每 128 字（地址 0~127）为页单位进行划分。

在页 1（0080₁₆~00FF₁₆）的起始部分分配了中断地址（图 17）。

如果发生中断，就将对应各中断的地址（中断地址）设定到程序计数器（PC），执行中断地址的指令。在使用中断处理程序时，必须将转移到该程序的指令写入中断地址。

页 2（0100₁₆~017F₁₆）是用于子程序调用的特殊页（图 16）。能通过字指令（BM 指令）从任意页中调用被写在此页中的子程序。另外，即使是从页 2 跨越到其他页的子程序，只要其起始部分在页 2 中，也能通过 BM 指令调用。

另外，能通过 TABP p 指令将全部地址的 ROM 模式区（bit9~0）用作数据区。

表 8. ROM 容量和页数

型 号	ROM (PROM) 容量 (×10 位)	页数
M34571G4	4096 字	32 (0~31)
M34571G6	6144 字	48 (0~47)
M34571GD (注 1)	16384 字	128 (0~127)

注 1. 初始状态可使用 TABP 指令参照页 0~63 的数据。使用执行 SBK 指令后的 TABP 指令可参照页 64~127 的数据。使用执行 RBK 指令后的 TABP 指令可参照页 0~63 的数据。

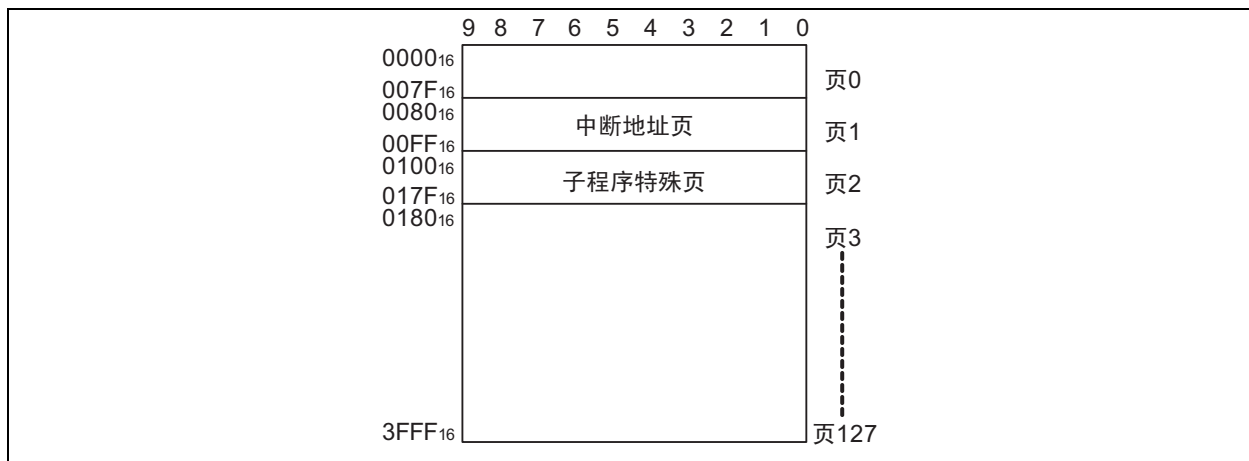
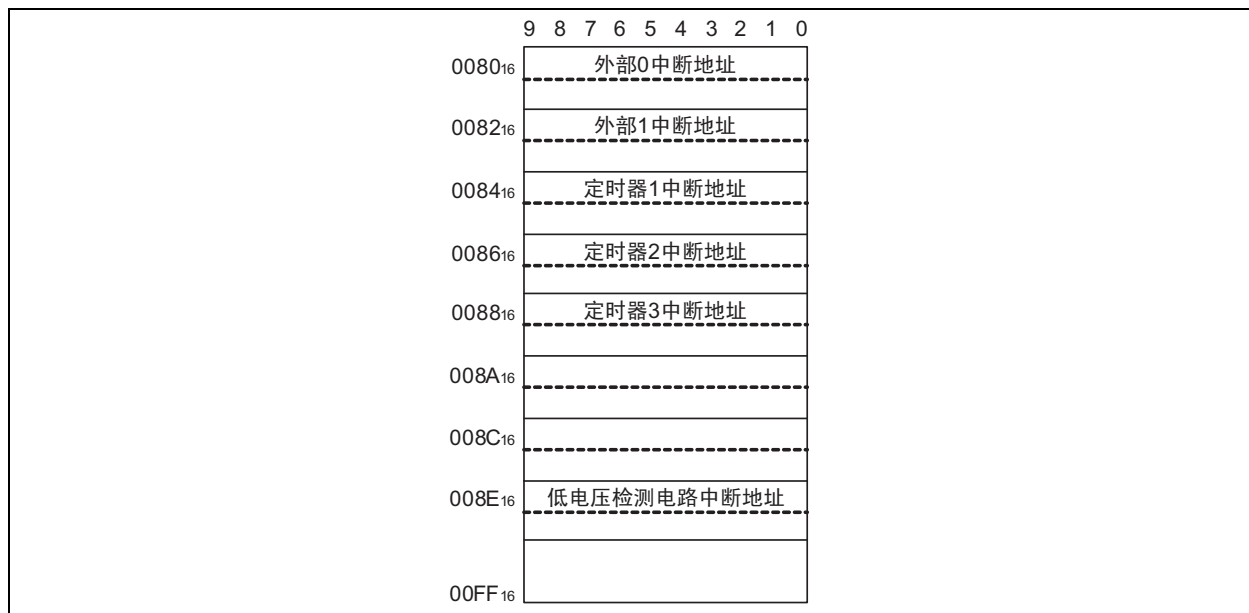


图 16. M34571GD 的 ROM 映像

图 17. 中断地址页 (0080₁₆~00FF₁₆) 的结构

●ROM 代码保护功能

如果在串行编程器中选择了“保护位写”，或者对本公司的编程后出货产品选择了“有保护”，就不能由串行编程器读写 ROM。

在由串行编程器对 QzROM 空白产品进行 ROM 写操作时，通过选择“保护位写”来保护 ROM 代码。

可在订货时用 ROM 选项（在掩模文件转换实用程序中记为“掩模选项”）选择 QzROM 编程后出货产品是否有 ROM 代码保护。详细内容请确认 QzROM 编程确认书。

数据存储器（RAM）

RAM 的 1 字由 4 位构成，但是，能通过 SBj、RBj、SZBj 指令以 1 位为单位对全部的存储区进行处理。

通过由寄存器 Z、X、Y 构成的数据指针指定 RAM 的地址。在执行存取 RAM 的指令时，必须给数据指针设定值（从 RAM 备份返回后也必须设定）。

RAM 容量如表 9、RAM 映像如图 18 所示。

表 9. RAM 容量

型 号	RAM 容量
M34571G4	128 字×4 位（512 位）
M34571G6	
M34571GD	

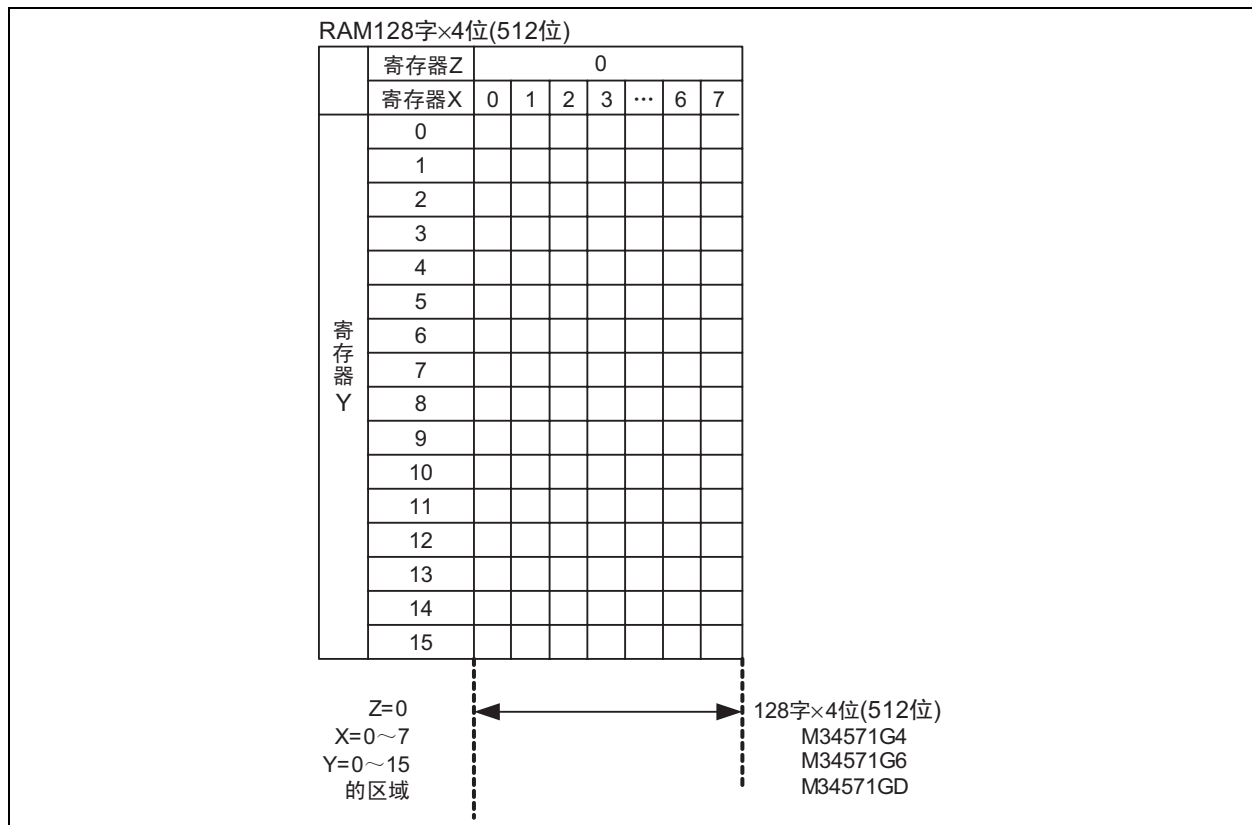


图 18. RAM 映像

●注意事项

因为数据指针的寄存器 Z 在复位解除后值不定，所以必须进行初始设定。

另外，寄存器 Z、X、Y 在 RAM 备份时处于不定，必须在从 RAM 备份返回后重新设定这些寄存器。

中断功能

中断类型是转移到各中断源不同地址（中断地址）的向量中断。在满足以下 3 个条件时发生中断：

- 中断允许标志处于允许状态（INTE=“1”）
- 中断允许位处于允许状态（“1”）
- 中断启动条件成立（请求标志=“1”）

各中断源的启动条件、中断地址与中断优先级的对应如表 10 所示。

有关启动条件的详细内容请参照各中断请求标志项。

表 10. 中断源、中断地址和优先级

优先级	中断源		中断地址
	中断名	启动条件	
1	低电压检测电路中断	低电源电压	页 1 的地址 E
2	外部 0 中断	INT0 引脚的电平变化	页 1 的地址 0
3	外部 1 中断	INT1 引脚的电平变化	页 1 的地址 2
4	定时器 1 中断	定时器 1 下溢	页 1 的地址 4
5	定时器 2 中断	定时器 2 下溢	页 1 的地址 6
6	定时器 3 中断	定时器 3 下溢	页 1 的地址 8

1. 中断允许标志（INTE）

标志 INTE 是控制所有中断的允许、禁止的标志。通过执行 EI 指令，将标志 INTE 置“1”，允许中断。另外，通过执行 DI 指令，将标志 INTE 清“0”，禁止中断。如果发生某个中断，标志 INTE 就自动清“0”。然后，在执行 EI 指令前，单片机内部保持为中断禁止状态。

2. 中断允许位（V10~V13、V20、V23）

该位用于控制各个中断源的中断请求是否有效或者跳越指令是否有效。各中断源的请求标志、跳越指令和中断控制寄存器的中断允许位的关系如表 11、中断允许位的功能如表 12 所示。

表 11. 中断请求标志、跳越指令和中断控制寄存器位

中断源	中断请求标志	中断跳越指令	中断允许位
低电压检测电路中断	VDF	SNZVD	V23
外部 0 中断	EXF0	SNZ0	V10
外部 1 中断	EXF1	SNZ1	V11
定时器 1 中断	T1F	SNZT1	V12
定时器 2 中断	T2F	SNZT2	V13
定时器 3 中断	T3F	SNZT3	V20

表 12. 中断允许位的功能

中断允许位的状态	中断的发生	跳越指令
1	允许	无效
0	禁止	有效

3. 中断请求标志

如果各中断的启动条件成立，就将对应该中断的中断请求标志置“1”。

除低电压检测电路中断请求标志外，当发生中断或者执行跳越指令时，就将与其对应的中断请求标志清“0”。低电压检测电路中断请求标志在启动条件成立的状态下不能清“0”。即使通过标志 INTE 或者中断允许位将各中断请求标志设定为中断禁止状态，如果启动条件成立，各中断请求标志也被置位。一旦置位后，在清除条件成立之前，中断请求标志始终保持该状态。

因此，如果在保持中断请求的状态下解除中断禁止状态，就在此时产生中断。在解除中断禁止状态时，如果有 2 个或者 2 个以上的中断请求标志被置位，就按照表 10 所示的优先级产生中断。

4. 发生中断时的内部状态

在发生中断时，单片机的内部状态如下（参照图 20）：

- 程序计数器 (PC)

设定中断地址。返回主程序时的执行地址自动保存到堆栈寄存器 SK。

- 中断允许标志 (INTE)

标志 INTE 被清“0”，变为中断禁止状态。

- 中断请求标志

只有对应中断源的请求标志被清“0”（除低电压检测电路中断请求标志外）。

- 数据指针、进位标志 (CY)、跳越标志以及寄存器 A、B

这些寄存器和标志的内容被自动保存到中断专用的堆栈寄存器 SDP。

5. 中断的处理方法

如果发生中断，就在处理寄存器 SK 的数据压栈顺序后从中断地址开始执行程序。必须将中断处理程序的转移指令写到中断地址，另外，必须使用 RTI 指令返回主程序。

另外，在经过 1 条指令后（在下一条指令的执行刚结束后）进行 EI 指令执行的中断允许。因此，如果在 RTI 指令前执行 EI 指令，就能在返回主程序后立即发生中断（参照图 19）。

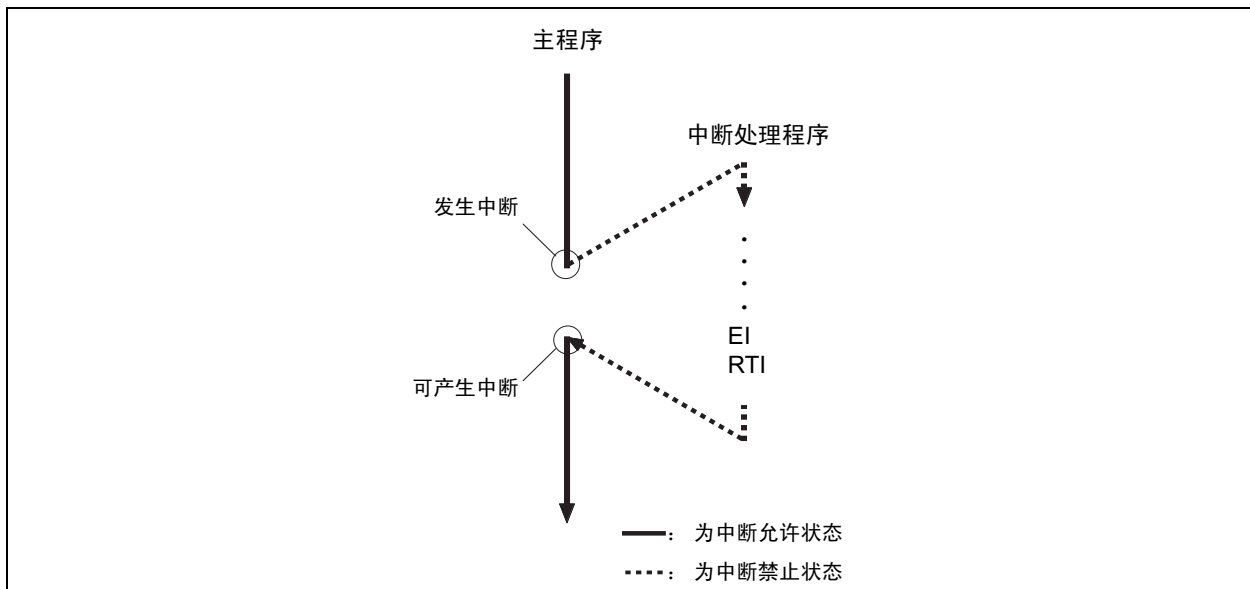


图 19. 中断处理程序的例子

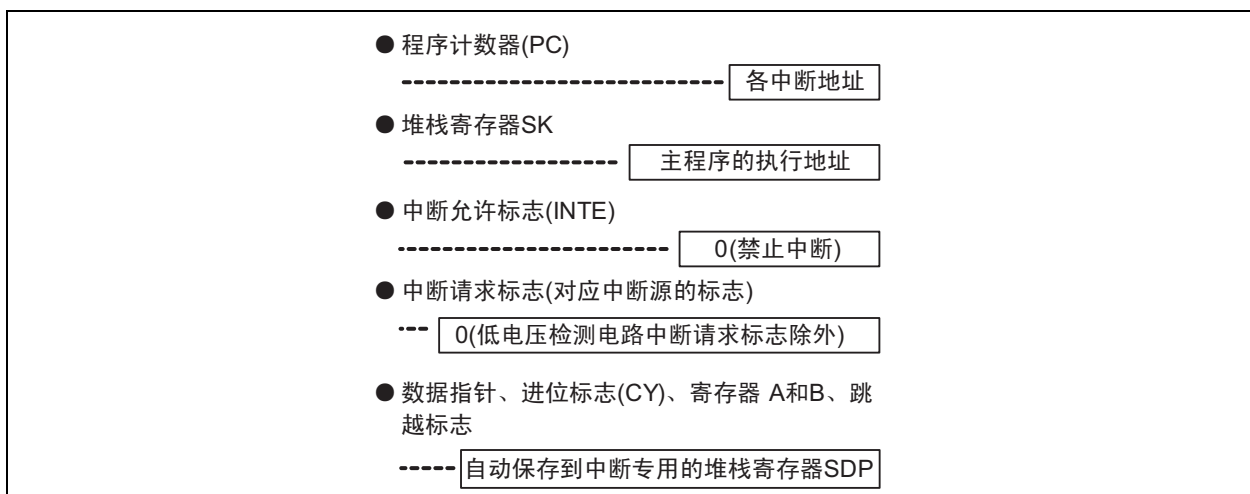


图 20. 发生中断时的内部状态

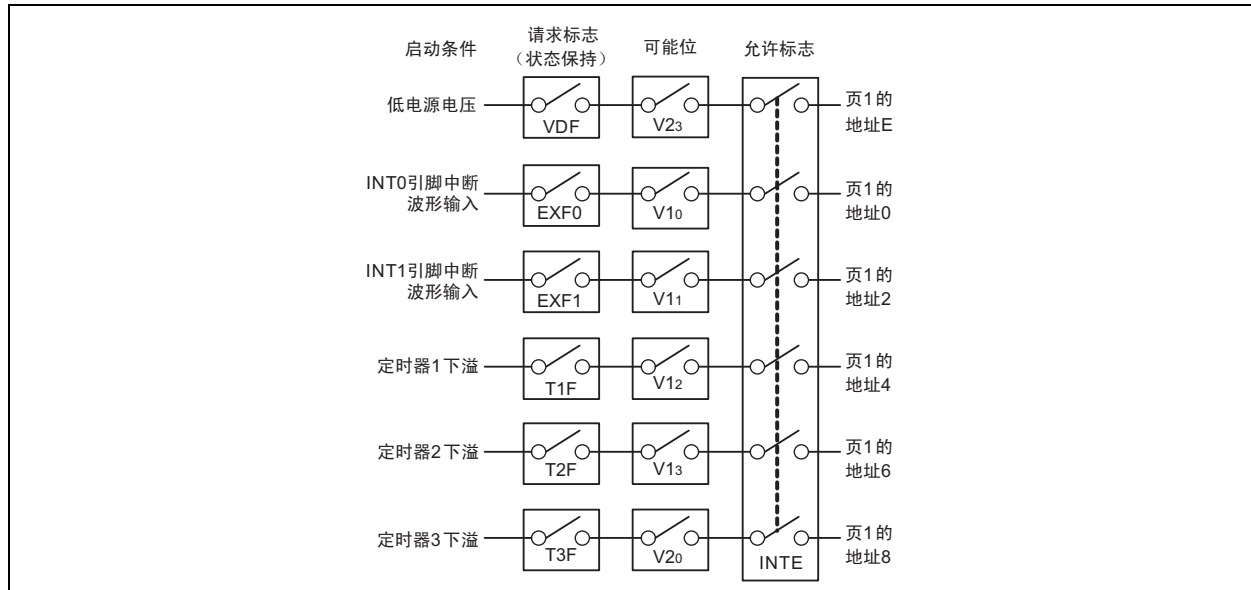


图 21. 中断系统图

6. 中断控制寄存器

● 中断控制寄存器 V1

在寄存器 V1 中分配了外部 0、定时器 1 和定时器 2 的中断允许位。必须由 TV1A 指令通过寄存器 A 设定寄存器 V1 的内容。

另外，能通过 TAV1 指令将寄存器 V1 的内容传送到寄存器 A。

● 中断控制寄存器 V2

在寄存器 V2 中分配了低电压检测电路中断允许位和定时器 3 中断允许位。必须由 TV2A 指令通过寄存器 A 设定寄存器 V2 的内容。另外，能通过 TAV2 指令将寄存器 V2 的内容传送到寄存器 A。

表 13. 中断控制寄存器

中断控制寄存器 V1		复位时: 0000 ₂		RAM 备份时: 0000 ₂	R/W TAV1/TV1A
V13	定时器 2 中断允许位	0	禁止发生 (SNZT2 指令有效)		
		1	允许发生 (SNZT2 指令无效)		
V12	定时器 1 中断允许位	0	禁止发生 (SNZT1 指令有效)		
		1	允许发生 (SNZT1 指令无效)		
V11	外部 1 中断允许位	0	禁止发生 (SNZ1 指令有效)		
		1	允许发生 (SNZ1 指令无效)		
V10	外部 0 中断允许位	0	禁止发生 (SNZ0 指令有效)		
		1	允许发生 (SNZ0 指令无效)		

中断控制寄存器 V2		复位时: 0000 ₂		RAM 备份时: 0000 ₂	R/W TAV2/TV2A
V23	低电压检测电路中断允许位	0	禁止发生 (SNZVD 指令有效)		
		1	允许发生 (SNZVD 指令无效)		
V22	不使用	0	此位无功能, 可进行读写操作。		
		1			
V21	不使用	0	此位无功能, 可进行读写操作。		
		1			
V20	定时器 3 中断允许位	0	禁止发生 (SNZT3 指令有效)		
		1	允许发生 (SNZT3 指令无效)		

注: “R”表示可读、“W”表示可写。

7. 中断顺序

当标志 INTE、中断允许位 (V10~V13、V20、V23) 和各中断请求标志为“1”时, 启动各中断。中断发生的时序是以上述 3 个条件全部成立的周期为起点的 2~3 个机器周期后。

中断发生在 3 个机器周期后的原因是中断条件成立时的指令不是 1 个周期的指令时 (参照图 22)。

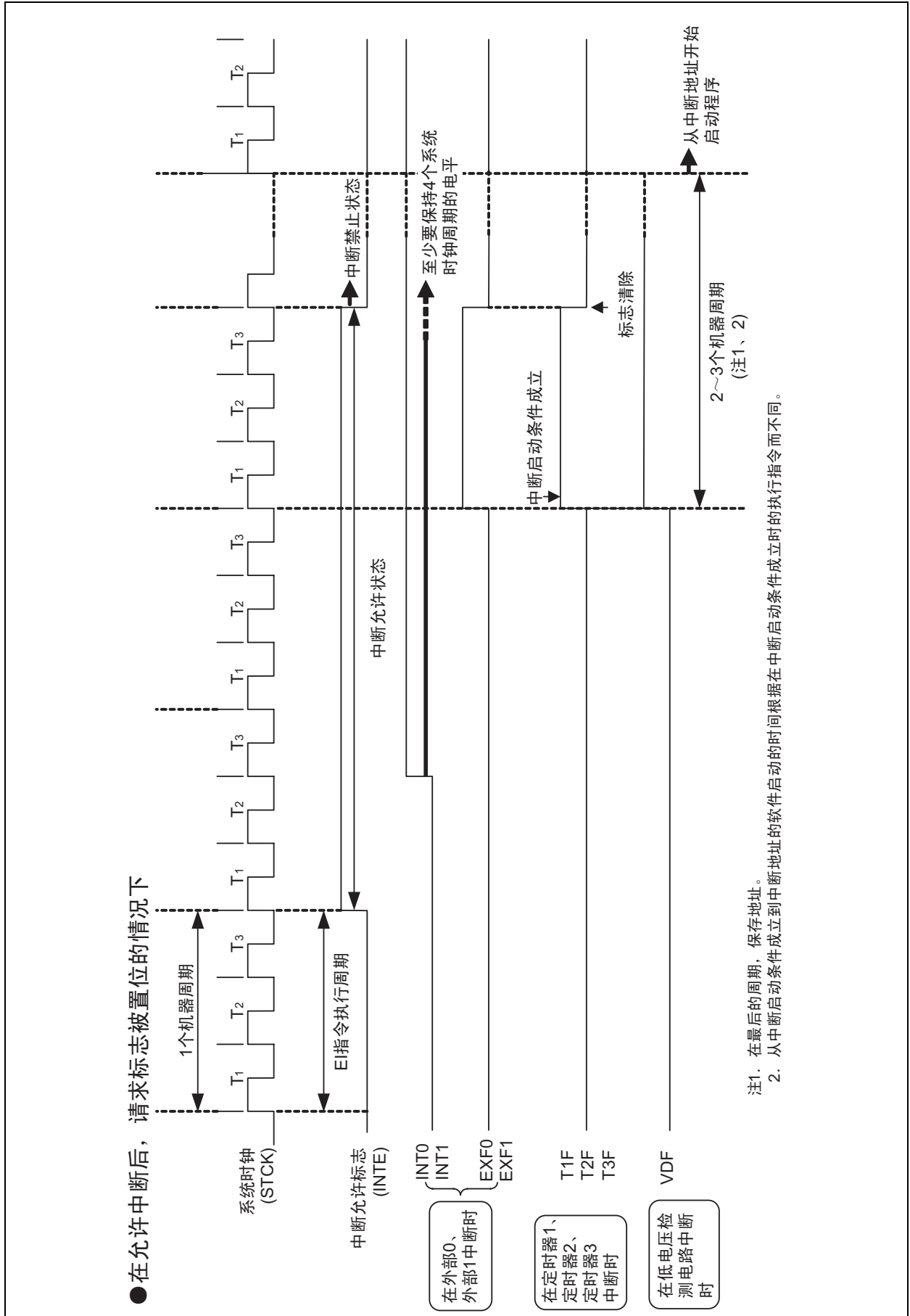


图 22. 中断顺序

外部中断

如果将有效波形输入到中断输入引脚，外部中断就发生中断请求（边沿检测）。

本产品具有 2 个外部中断功能（外部 0、外部 1）。它们能由中断控制寄存器 I1、I2 控制。

表 14. 外部中断的启动条件

中断名	输入引脚	有效波形	有效波形的选择位
外部 0 中断	P20/INT0	在下一个波形输入到 P20/INT0 引脚时 • 下降波形（“H” → “L”） • 上升波形（“L” → “H”） • 下降波形和上升波形两波形	I11、I12
外部 1 中断	P21/INT1	在下一个波形输入到 P21/INT1 引脚时 • 下降波形（“H” → “L”） • 上升波形（“L” → “H”） • 下降波形和上升波形两波形	I21、I22

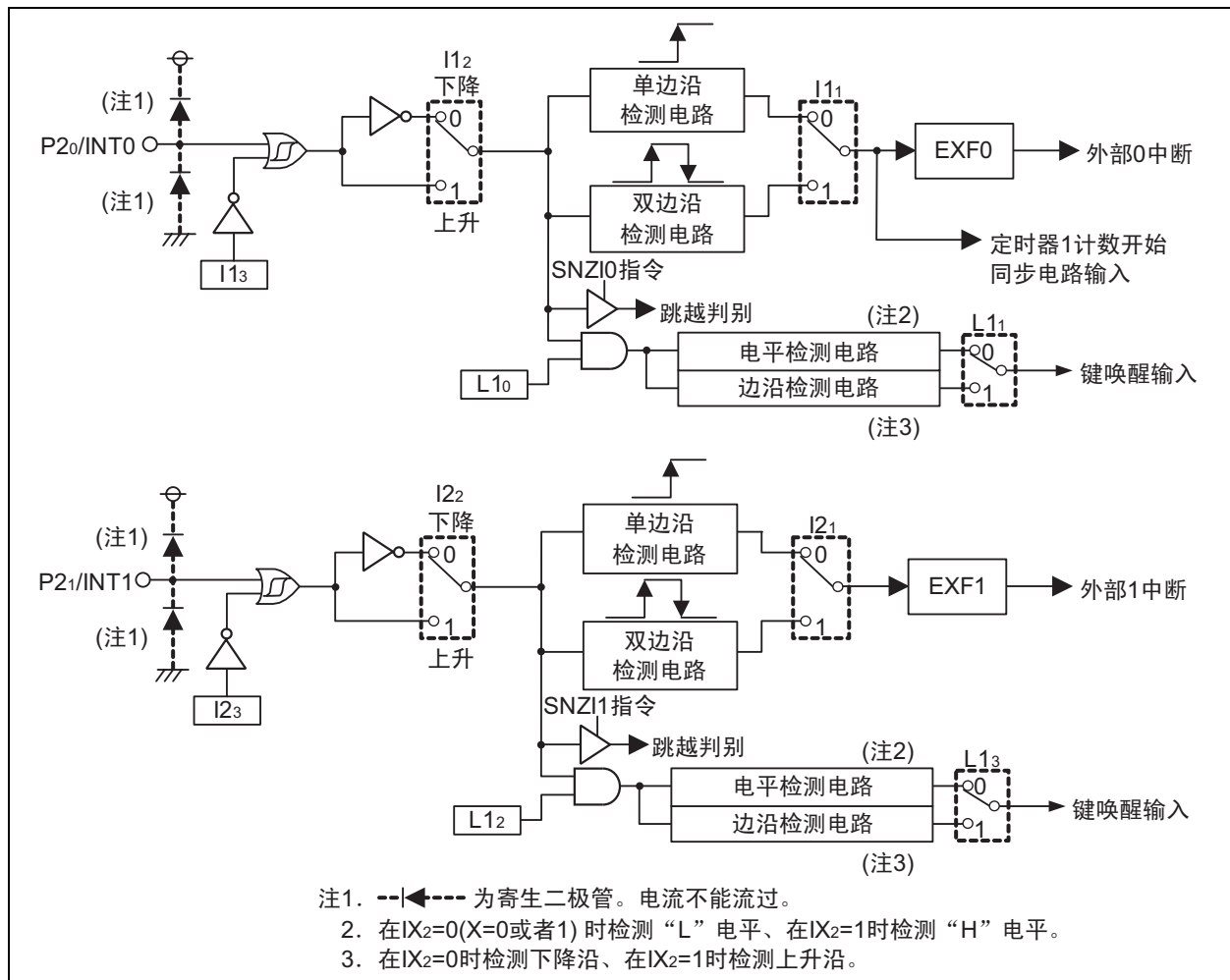


图 23. 外部中断电路的结构

1. 外部 0 中断的请求标志 (EXF0)

在有效波形输入到 P20/INT0 引脚时，将标志 EXF0 置“1”。

作为外部 0 中断启动条件的有效波形，需要将变化前后的电平至少保持 4 个系统时钟周期（参照图 22）。

能通过执行跳越指令（SNZ0 指令）确认标志 EXF0 的状态。

必须通过中断控制寄存器 V1 选择是使用中断还是使用跳越指令。

在发生中断或者执行跳越指令时，将标志 EXF0 清“0”。

●外部 0 中断的启动条件

在有效波形输入到 P20/INT0 引脚时外部 0 中断的启动条件成立。有效波形能从下降波形、上升波形、或者上升和下降两波形的 3 种中选择。外部 0 中断的使用方法的一个例子如下所示：

- (1) 将中断控制寄存器 I1 的 bit3 (I13) 置“1”，设定 INT0 引脚为可输入状态
- (2) 通过中断控制寄存器 I1 的 bit1 (I11)、bit2 (I12) 选择有效波形
- (3) 使用 SNZ0 指令，将标志 EXF0 清“0”
- (4) 考虑到因 SNZ0 指令发生跳越的情况，插入 NOP 指令
- (5) 将外部 0 中断允许位 (V10) 和中断允许标志 (INTE) 都置“1”

通过以上操作，变为外部 0 中断发生允许状态。如果在此状态下将有效波形输入到 P20/INT0 引脚，就将标志 EXF0 置“1”，产生外部 0 中断。

2. 外部 1 中断的请求标志 (EXF1)

在有效波形输入到 P21/INT1 引脚时，将标志 EXF1 置“1”。

作为外部 1 中断启动条件的有效波形，需要将变化前后的电平至少保持 4 个系统时钟周期（参照图 22）。

能通过执行跳越指令（SNZ1 指令）确认标志 EXF1 的状态。

必须通过中断控制寄存器 V1 选择是使用中断还是使用跳越指令。

在发生中断或者执行跳越指令时，将标志 EXF1 清“0”。

●外部 1 中断的启动条件

在有效波形输入到 P21/INT1 引脚时外部 1 中断的启动条件成立。有效波形能从下降波形、上升波形、或者上升和下降两波形的 3 种中选择。外部 1 中断的使用方法的一个例子如下所示：

- (1) 将中断控制寄存器 I2 的 bit3 (I23) 置“1”，设定 INT1 引脚为可输入状态
- (2) 通过中断控制寄存器 I2 的 bit1 (I21)、bit2 (I22) 选择有效波形
- (3) 使用 SNZ1 指令，将标志 EXF1 清“0”
- (4) 考虑到因 SNZ1 指令发生跳越的情况，插入 NOP 指令
- (5) 将外部 1 中断允许位 (V11) 和中断允许标志 (INTE) 都置“1”

通过以上操作，成为外部 1 中断发生允许状态。如果在此状态下将有效波形输入到 P21/INT1 引脚，就将标志 EXF1 置“1”，产生外部 1 中断。

3. 外部中断控制寄存器

(1) 中断控制寄存器 I1

寄存器 I1 控制外部 0 中断的有效波形，必须由 TI1A 指令通过寄存器 A 设定此寄存器的内容。另外，能通过 TAI1 指令将寄存器 I1 的内容传送到寄存器 A。

(2) 中断控制寄存器 I2

寄存器 I2 控制外部 1 中断的有效波形，必须由 TI2A 指令通过寄存器 A 设定此寄存器的内容。另外，能通过 TAI2 指令将寄存器 I2 的内容传送到寄存器 A。

表 15. 外部中断控制寄存器

中断控制寄存器 I1		复位时: 0000 ₂		RAM 备份时: 保持状态	R/W TAI1/TI1A
I13	INT0 引脚输入控制位 (注 2)	0	禁止输入		
		1	允许输入		
I12	INT0 引脚中断有效波形/ 返回电平选择位 (注 2)	0	下降波形/“L”电平 (SNZIO 指令识别 “L” 电平)		
		1	上升波形/“H”电平 (SNZIO 指令识别 “H” 电平)		
I11	INT0 引脚边沿检测电路控制位	0	检测单边沿		
		1	检测双边沿		
I10	INT0 引脚定时器 1 控制允许位	0	禁止定时器 1 控制		
		1	允许定时器 1 控制		

中断控制寄存器 I2		复位时: 0000 ₂		RAM 备份时: 保持状态	R/W TAI2/TI2A
I23	INT1 引脚输入控制位 (注 3)	0	禁止输入		
		1	允许输入		
I22	INT1 引脚中断有效波形/ 返回电平选择位 (注 3)	0	下降波形/“L”电平 (SNZI1 指令识别 “L” 电平)		
		1	上升波形/“H”电平 (SNZI1 指令识别 “H” 电平)		
I21	INT1 引脚边沿检测电路控制位	0	检测单边沿		
		1	检测双边沿		
I20	不使用	0	此位没有功能, 可进行读写操作		
		1			

注 1. “R” 表示可读, “W” 表示可写。

注 2. 在更改这些位 (I12、I13) 的内容时, 有可能将外部中断请求标志 (EXF0) 置位。

注 3. 在更改这些位 (I22、I23) 的内容时, 有可能将外部中断请求标志 (EXF1) 置位。

4. 注意事项

(1) 有关寄存器 I1 的 bit3 的注意事项 1

在软件中通过中断控制寄存器 I1 的 bit3 控制 INT0 引脚的输入时，必须注意：

- 在更改寄存器 I1 的 bit3 的内容时，有时需要根据 P20/INT0 引脚的输入状态将外部 0 中断请求标志 (EXF0) 置“1”。为了防止发生意外的中断，必须在将中断控制寄存器 V1 的 bit0 清“0” (图 24①) 后更改寄存器 I1 的 bit3 的内容。而且，必须在至少间隔一条指令后执行 (图 24②) SNZ0 指令，将标志 EXF0 清“0”。另外，考虑到因 SNZ0 指令发生跳越的情况，必须在 SNZ0 指令后插入 NOP 指令 (图 24③)。

⋮	
LA 4	; (×××02)
TV1A	; SNZ0指令有效⋯⋯①
LA 8	; (1×××2)
TI1A	; 改变INT0引脚输入控制
NOP	; ⋯⋯⋯⋯⋯⋯②
SNZ0	; 执行SNZ0指令
	(清除标志EXF0)
NOP	; ⋯⋯⋯⋯⋯⋯③
⋮	
	×：此位与本例无关。

图 24. 外部 0 中断程序例 1

(2) 有关寄存器 I1 的 bit3 的注意事项 2

在将中断控制寄存器 I1 的 bit3 清“0”并且在禁止 INT0 引脚输入的状态下使用 RAM 备份时，必须注意：

- 在禁止INT0引脚输入时 (寄存器I13=“0”)，必须在转移到RAM备份模式前将INT0引脚的键唤醒设定为无效 (寄存器L10=“0”) (图25①)。

⋮	
LA 0	; (×××02)
TI1A	; INT0键唤醒无效⋯⋯①
DI	
EPOF	
POF	; RAM备份
⋮	
	×：此位与本例无关。

图 25. 外部 0 中断程序例 2

(3) 有关寄存器 I1 的 bit2 的注意事项

在软件中通过中断控制寄存器 I1 的 bit2 更改 P20/INT0 引脚的中断有效波形时，必须注意：

- 在更改寄存器 I1 的 bit2 的内容时，有时需要根据 P20/INT0 引脚的输入状态将外部 0 中断请求标志 (EXF0) 置“1”。为了防止发生意外的中断，必须在将中断控制寄存器 V1 的 bit0 清“0” (图 26①) 后更改寄存器 I1 的 bit2 的内容。而且，必须在至少间隔一条指令后执行 (图 26②) SNZ0 指令，将标志 EXF0 清“0”。另外，考虑到因 SNZ0 指令发生跳越的情况，必须在 SNZ0 指令后插入 NOP 指令 (图 26③)。

⋮	
LA 4	; (×××02)
TV1A	; SNZ0指令有效⋯⋯①
LA 12	; (1×××2)
TI1A	; 改变中断有效波形
NOP	; ⋯⋯⋯⋯⋯⋯②
SNZ0	; 执行SNZ0指令 (清除标志EXF0)
NOP	; ⋯⋯⋯⋯⋯⋯③
⋮	
	×: 此位与本例无关。

图 26. 外部 0 中断程序例 3

(4) 有关寄存器 I2 的 bit3 的注意事项 1

在软件中通过中断控制寄存器 I2 的 bit3 控制 INT1 引脚的输入时，必须注意：

- 在更改寄存器 I2 的 bit3 的内容时，有时需要根据 P21/INT1 引脚的输入状态将外部 1 中断请求标志 (EXF0) 置“1”。为了防止发生意外的中断，必须在将中断控制寄存器 V1 的 bit1 清“0” (图 27①) 后更改寄存器 I2 的 bit3 的内容。而且，必须在至少间隔一条指令后执行 (图 27②) SNZ1 指令，将标志 EXF1 清“0”。另外，考虑到因 SNZ1 指令发生跳越的情况，必须在 SNZ1 指令后插入 NOP 指令 (图 27③)。

⋮	
LA 4	; (××0×2)
TV1A	; SNZ1指令有效⋯⋯①
LA 8	; (1×××2)
TI2A	; 改变INT1引脚输入控制
NOP	; ⋯⋯⋯⋯⋯⋯②
SNZ1	; 执行SNZ1指令 (清除标志EXF1)
NOP	; ⋯⋯⋯⋯⋯⋯③
⋮	
	×: 此位与本例无关。

图 27. 外部 1 中断程序例 1

(5) 有关寄存器 I2 的 bit3 的注意事项 2

在将中断控制寄存器 I2 的 bit3 清“0”并且在禁止 INT1 引脚输入的状态下使用 RAM 备份时，必须注意：

- 在禁止INT1引脚输入时 (寄存器I23=“0”)，必须在转移到RAM备份模式前将INT1引脚的键唤醒设定为无效 (寄存器L12=“0”) (图28①)。

```

      ⋮
      LA 0    ;(×0××2)
      TL1A   ;INT1键唤醒无效⋯⋯①
      DI
      EPOF
      POF    ;RAM备份
      ⋮
      ×: 此位与本例无关。

```

图 28. 外部 1 中断程序例 2

(6) 有关寄存器 I2 的 bit2 的注意事项

在软件中通过中断控制寄存器 I2 的 bit2 更改 P2i/INT1 引脚的中断有效波形时，必须注意：

- 在更改寄存器 I2 的 bit2 的内容时，有时需要根据 P2i/INT1 引脚的输入状态将外部 1 中断请求标志 (EXF1) 置“1”。为了防止发生意外的中断，必须在将中断控制寄存器 V1 的 bit1 清“0” (图 29①) 后更改寄存器 I2 的内容。而且，必须在至少间隔一条指令后执行 (图 29②) SNZ1 指令，将标志 EXF1 清“0”。另外，考虑到因 SNZ1 指令发生跳越的情况，必须在 SNZ1 指令后插入 NOP 指令 (图 29③)。

```

      ⋮
      LA 4    ;(××0×2)
      TV1A   ;SNZ1指令有效⋯⋯①
      LA 12   ;(1×××2)
      TI2A   ;改变中断有效波形
      NOP    ;⋯⋯⋯⋯⋯⋯②
      SNZ1   ;执行SNZ1指令
            (清除标志EXF1)
      NOP    ;⋯⋯⋯⋯⋯⋯③
      ⋮
      ×: 此位与本例无关。

```

图 29. 外部 1 中断程序例 3

定时器

本产品内置的定时器有以下几种：

● 可编程定时器

可编程定时器是能设定分频比的定时器，有重新加载寄存器。如果从设定值 n 开始递减计数，并且发生下溢（进行 $n+1$ 计数），就从重新加载寄存器重新装入数据，继续计数（自动重新加载功能）。

● 固定分频定时器

固定分频定时器是固定分频比（ n ）的定时器，在对计数脉冲每进行 n 次计数后发生下溢。

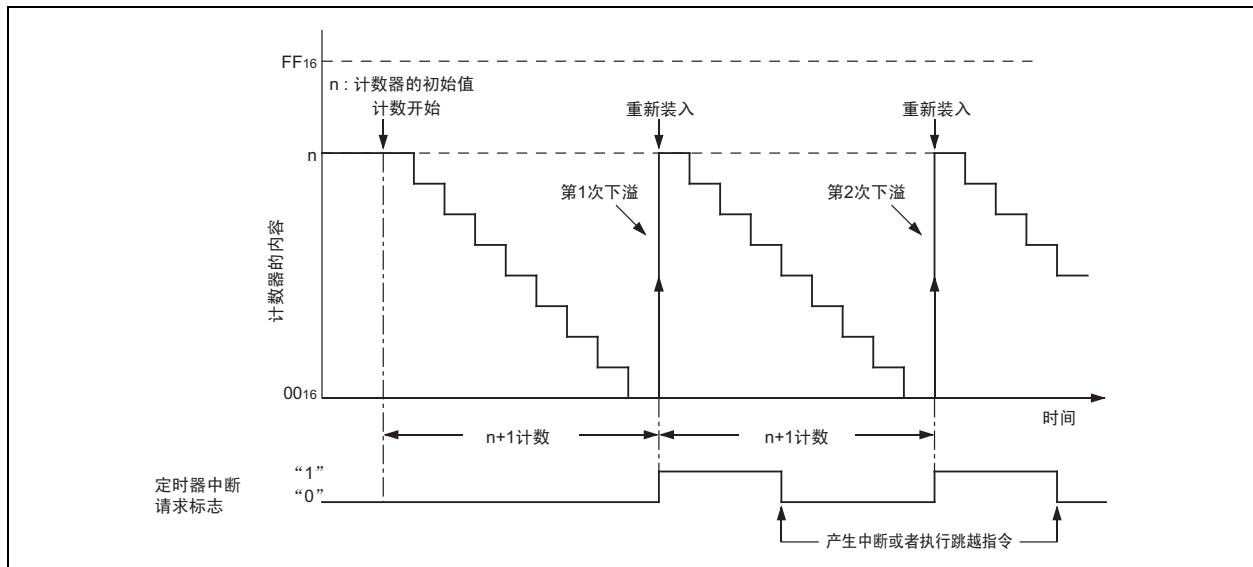


图 30. 自动重新加载功能

本产品的定时器由以下电路构成：

- 预分频器：8 位可编程定时器
- 定时器 1：8 位可编程定时器
- 定时器 2：8 位可编程定时器
- 定时器 3：8 位可编程定时器
- 看门狗定时器：16 位固定分频定时器（定时器 1、2、3 带中断功能）

预分频器、定时器 1、2、3 能由定时器控制寄存器 PA、W1、W2、W3、W5 控制。

看门狗定时器是没有控制寄存器的自由计数器。

以下说明各功能：

表 16. 定时器的功能一览表

电路名	结构	计数源	分频比	输出信号的用途	控制寄存器
预分频器	8 位可编程二进制递减计数器	<ul style="list-style-type: none"> • 指令时钟 (INSTCK) • 指令时钟的 4 分频信号 (INSTCK/4) 	1~256	<ul style="list-style-type: none"> • 定时器 1 计数源 • 定时器 2 计数源 • 定时器 3 计数源 	PA
定时器 1	8 位可编程二进制递减计数器 (带 INTO 输入连动功能、载波输出自动控制功能)	<ul style="list-style-type: none"> • PWM 信号 (PWMOUT) • 预分频器输出 (ORCLK) • CNTR0 输入 (CNTR0IN) • 系统时钟 (STCK) 	1~256	<ul style="list-style-type: none"> • 定时器 2 计数源 • CNTR0 输出 • 载波输出自动控制 • 定时器 1 中断 	W1 W5
定时器 2	8 位可编程二进制递减计数器	<ul style="list-style-type: none"> • PWM 信号 (PWMOUT) • 定时器 1 下溢 (T1UDF) • 预分频器输出 (ORCLK) • 系统时钟 (STCK) 	1~256	<ul style="list-style-type: none"> • CNTR0 输出 • 定时器 2 中断 	W2 W5
定时器 3	8 位可编程二进制递减计数器 (带载波发生功能)	<ul style="list-style-type: none"> • XIN 输入 • 预分频器输出的 2 分频信号 (ORCLK/2) 	1~256	<ul style="list-style-type: none"> • 定时器 1 计数源 • 定时器 2 计数源 • CNTR1 输出 • 定时器 3 中断 	W1 W3 W5
看门狗定时器	16 位固定分频	<ul style="list-style-type: none"> • 指令时钟 (INSTCK) 	65536	<ul style="list-style-type: none"> • 系统复位 (2 次计数) • WDF1 标志判断 	—

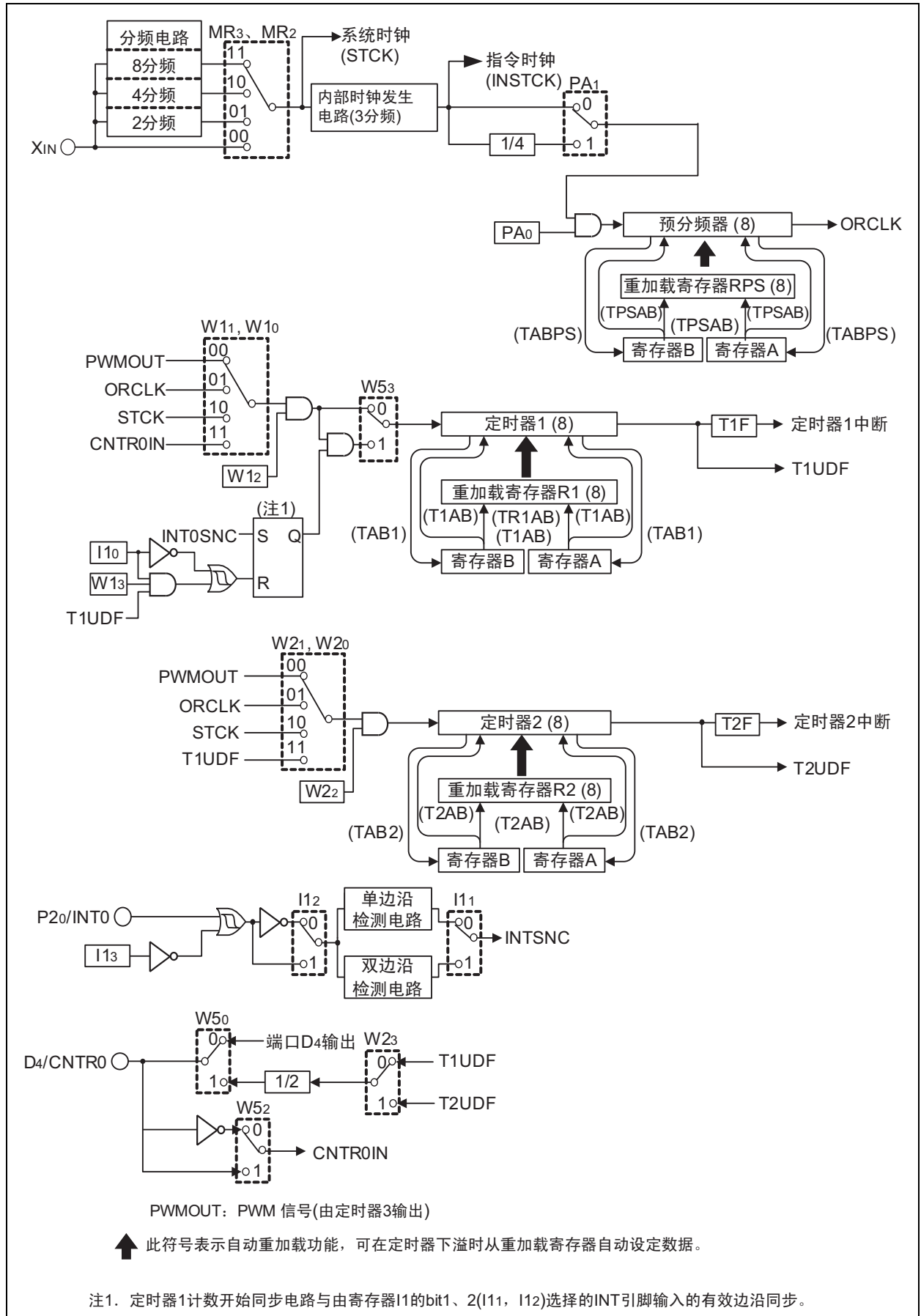


图 31. 定时器的结构 (1)

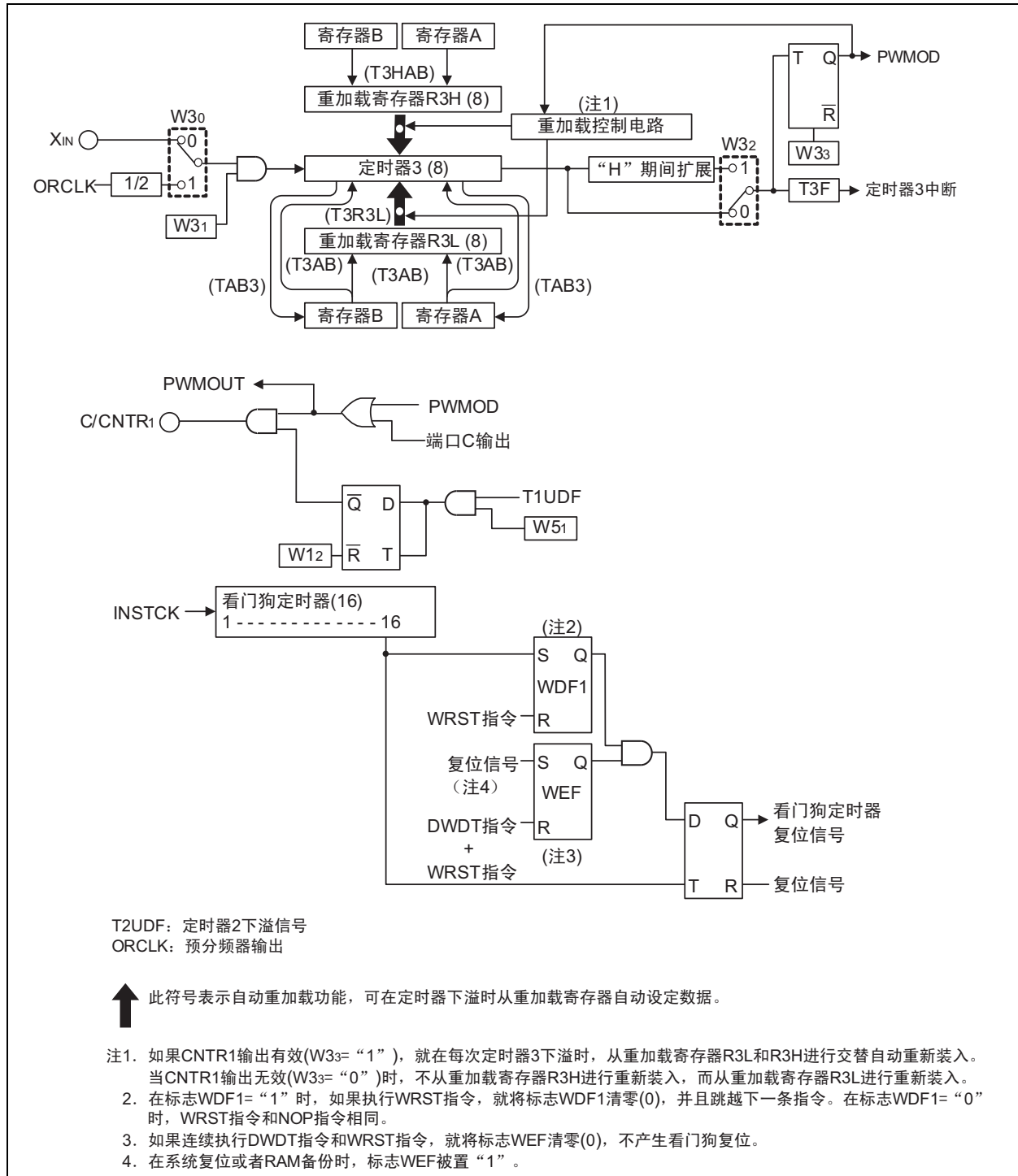


图 32. 定时器的结构 (2)

表 17. 定时器控制寄存器

定时器控制寄存器 PA		复位时: 00 ₂		RAM 备份时: 00 ₂	W TPAA
PA1	预分频器计数源选择位	0	指令时钟 (INSTCK)		
		1	指令时钟 (INSTCK) 的 4 分频信号		
PA0	预分频器控制位	0	停止 (保持状态)		
		1	运行		

定时器控制寄存器 W1		复位时: 0000 ₂		RAM 备份时: 保持状态	R/W TAW1/TW1A
W13	定时器 1 计数自动停止电路选择位 (注 2)	0	不选择定时器 1 计数自动停止电路		
		1	选择定时器 1 计数自动停止电路		
W12	定时器 1 控制位	0	停止 (保持状态)		
		1	运行		
W11	定时器 1 计数源选择位	W11	W10	计数源	
		0	0	PWM 信号 (PWMOUT)	
W10	定时器 1 计数源选择位	0	1	预分频器输出 (ORCLK)	
		1	0	系统时钟 (STCK)	
		1	1	CNTR0 输入	

定时器控制寄存器 W2		复位时: 0000 ₂		RAM 备份时: 保持状态	R/W TAW2/TW2A
W23	CNTR0 引脚功能选择位	0	定时器 1 下溢信号的 2 分频输出		
		1	定时器 2 下溢信号的 2 分频输出		
W22	定时器 2 控制位	0	停止 (保持状态)		
		1	运行		
W21	定时器 2 计数源选择位	W21	W20	计数源	
		0	0	PWM 信号 (PWMOUT)	
W20	定时器 2 计数源选择位	0	1	预分频器输出 (ORCLK)	
		1	0	系统时钟 (STCK)	
		1	1	定时器 1 的下溢信号 (T1UDF)	

定时器控制寄存器 W3		复位时: 0000 ₂		RAM 备份时: 0000 ₂	R/W TAW3/TW3A
W33	CNTR1 引脚输出控制位	0	CNTR1 引脚输出无效		
		1	CNTR1 引脚输出有效		
W32	PWM 信号 为高电平期间的扩展功能控制位	0	PWM 信号为高电平期间的扩展功能无效		
		1	PWM 信号为高电平期间的扩展功能有效		
W31	定时器 3 控制位	0	停止 (保持状态)		
		1	运行		
W30	定时器 3 计数源选择位	0	XIN 输入		
		1	预分频器输出 (ORCLK) 的 2 分频信号		

定时器控制寄存器 W5		复位时: 0000 ₂		RAM 备份时: 保持状态	R/W TAW5/TW5A
W53	定时器 1 计数开始同步电路选择位 (注 3)	0	不选择定时器 1 计数开始同步电路		
		1	选择定时器 1 计数开始同步电路		
W52	CNTR0 引脚输入计数边沿选择位	0	下降沿		
		1	上升沿		
W51	CNTR1 引脚输出自动控制电路选择位	0	不选择 CNTR1 引脚输出自动控制电路		
		1	选择 CNTR1 引脚输出自动控制电路		
W50	D4/CNTR0 引脚功能选择位	0	D4 输入/输出或者 CNTR0 输入		
		1	D4 输入或者 CNTR0 输入/输出		

注 1. “R”表示可读, “W”表示可写。

注 2. 此功能只在 INTO 引脚定时器 1 控制允许 (I10=“1”) 以及选择定时器 1 的计数开始同步电路 (W53=“1”) 时有效。

注 3. 此功能只在 INTO 引脚定时器 1 控制允许 (I10=“1”) 时有效。

1. 定时器相关的控制寄存器

● 定时器控制寄存器 PA

寄存器 PA 控制预分频器的计数源选择及计数运行。必须由 TPAA 指令通过寄存器 A 设定此寄存器。

● 定时器控制寄存器 W1

寄存器 W1 控制定时器 1 的计数源选择、计数运行和定时器 1 计数自动停止电路。必须由 TW1A 指令通过寄存器 A 设定此寄存器的内容。另外，能通过 TAW1 指令将寄存器 W1 的内容传送到寄存器 A。

● 定时器控制寄存器 W2

寄存器 W2 控制定时器 2 的计数源选择、计数运行和 CNTR0 引脚输出信号。必须由 TW2A 指令通过寄存器 A 设定此寄存器的内容。另外，能通过 TAW2 指令将寄存器 W2 的内容传送到寄存器 A。

● 定时器控制寄存器 W3

寄存器 W3 控制定时器 3 的计数源选择、计数运行、CNTR1 引脚输出以及在 PWM 信号为高电平“H”时的扩展功能。必须由 TW3A 指令通过寄存器 A 设定此寄存器的内容。另外，能通过 TAW3 指令将寄存器 W3 的内容传送到寄存器 A。

● 定时器控制寄存器 W5

寄存器 W5 控制定时器 1 计数开始同步电路、CNTR0 引脚输入计数边沿、CNTR1 引脚输出自动控制电路以及 D4/CNTR0 引脚功能。必须由 TW5A 指令通过寄存器 A 设定此寄存器的内容。另外，能通过 TAW5 指令将寄存器 W5 的内容传送到寄存器 A。

2. 预分频器

预分频器是 8 位二进制计数器，带有预分频器重加载寄存器 RPS。能通过 TPSAB 指令同时给预分频器和重加载寄存器 RPS 设定数据，也能通过 TABPS 指令从预分频器读数据。

在设定或者读预分频器的数据时，必须在停止计数后执行 TPSAB 指令或者 TABPS 指令。

如果在给预分频器设定数据后将寄存器 PA 的 bit0 置“1”，预分频器就开始计数。

如果重加载寄存器 RPS 的设定值为 n，预分频器就对计数源的信号进行 n+1 分频（n=0~255）。预分频器的计数源可选择是指令时钟（INSTCK）还是指令时钟的 4 分频信号。

如果在开始计数后预分频器发生下溢（在预分频器的内容变为“0”后，输入下一个计数脉冲），就从重加载寄存器 RPS 重新装入数据，继续计数（自动重加载功能）。预分频器的输出信号（ORCLK）能用于定时器 1、2 的计数源。

3. 定时器 1（带中断功能）

定时器 1 是 8 位二进制计数器，带有定时器 1 重加载寄存器 R1。能通过 T1AB 指令同时给定时器 1 和重加载寄存器 R1 设定数据，通过 TR1AB 指令给重加载寄存器 R1 设定数据。通过 TAB1 指令从定时器 1 读数据。

在设定或者读定时器 1 的数据时，必须在停止计数后执行 T1AB 指令或者 TAB1 指令。

在定时器 1 运行过程中给重加载寄存器 R1 设定数据时，必须在不与下溢重叠的时序执行 TR1AB 指令。

在给定时器 1 设定数据后，如果通过寄存器 W1 的 bit0、1 设定计数源，并将寄存器 W1 的 bit2 置“1”，定时器 1 就开始计数。

如果重加载寄存器 R1 的设定值为 n，定时器 1 就对计数源的信号进行 n+1 分频（n=0~255）。

如果在开始计数后定时器 1 下溢（在定时器 1 的内容变为“0”后，输入下一个计数脉冲），就将定时器 1 的中断请求标志（T1F）置“1”，从重加载寄存器 R1 重新装入数据，并继续计数（自动重加载功能）。

如果给寄存器 I1 的 bit0 置“1”，将 INTO 引脚的定时器 1 控制设定为允许状态，并将寄存器 W5 的 bit3 置“1”，INT0 引脚的输入就能用于定时器 1 计数运行的开始触发。另外，此时如果将寄存器 W1 的 bit3 置“1”，就能通过定时器 1 的下溢自动停止。

如果将寄存器 W5 的 bit0 置“1”、寄存器 W2 的 bit3 清“0”，就能从 CNTR0 引脚输出定时器 1 下溢信号的 2 分频信号。

4. 定时器 2（带中断功能）

定时器 2 是 8 位二进制计数器，带有定时器 2 重加载寄存器 R2。能通过 T1AB 指令同时给定时器 2 和重加载寄存器 R2 设定数据，并能通过 TAB2 指令从定时器 2 读数据。

在设定或者读定时器 2 的数据时，必须在停止计数后执行 T2AB 指令或者 TAB2 指令。

在给定时器 2 设定数据后，如果通过寄存器 W2 的 bit0、1 设定计数源，并将寄存器 W2 的 bit2 置“1”，定时器 2 就开始计数。

如果重加载寄存器 R2 的设定值为 n，定时器 2 就对计数源的信号进行 n+1 分频（n=0~255）。

如果在计数开始后定时器 2 下溢（在定时器 2 的内容变为“0”后，输入下一个计数脉冲），就将定时器 2 的中断请求标志（T2F）置“1”，从重加载寄存器 R2 重新装入数据，并继续计数（自动重加载功能）。

如果将寄存器 W5 的 bit0 置“1”、寄存器 W2 的 bit3 置“1”，就能从 CNTR0 引脚输出定时器 2 下溢信号的 2 分频信号。

5. 定时器 3（带中断功能）

定时器 3 是 8 位二进制计数器，带有 2 个定时器 3 重加载寄存器 R3L、R3H。能通过 T3AB 指令同时给定时器 3 和重加载寄存器 R3L 设定数据。能通过 T3HAB 指令给重加载寄存器 R3H 设定数据。通过 T3AB 指令设定的重加载寄存器 R3L 的内容可通过 T3R3L 指令重新设定给定时器 3。通过 TAB3 指令可从定时器 3 读数据。

在设定或者读出定时器 3 的数据时，必须在停止计数后执行 T3AB 指令或者 TAB3 指令。

在定时器 3 运行过程中给重加载寄存器 R3H 设定数据时，必须在不与下溢重叠的时序执行 T3HAB 指令。

在给定时器 3 设定数据后，如果通过寄存器 W3 的 bit0 设定计数源，并将寄存器 W3 的 bit1 置“1”，定时器 3 就开始计数。如果重加载寄存器 R3L 的设定值为 n、重加载寄存器 R3H 的设定值为 m，定时器 3 就对计数源的信号进行 n+1 或者 m+1 分频（n=0~255、m=0~255）。

寄存器 W3 的 bit3 为“0”时，如果在计数开始后定时器 3 下溢（在定时器 3 的内容变为“0”后，输入下一个计数脉冲），就将定时器 3 的中断请求标志（T3F）置“1”，从重加载寄存器 R3L 重新装入数据，并继续计数（自动重加载功能）。

在寄存器 W3 的 bit3 为“1”时，定时器 3 就产生“L”期间和“H”期间的 PWM（PWMOD）信号（“L”期间是在重加载寄存器 R3L 中设定的期间，“H”期间是在重加载寄存器 R3H 中设定的期间），从 CNTR1 引脚输出。

此时如果将寄存器 W3 的 bit2 置“1”，那么定时器 3 在重加载寄存器 R3H 中设定的期间就扩展计数源的半个周期。如果重加载寄存器 R3H 的设定值为 m，定时器 3 就对计数源的信号进行 m+1.5 分频（m=1~255）。使用此功能时，重加载寄存器 R3H 的设定值必须至少为“1”。

如果将寄存器 W5 的 bit1 置“1”，那么，每当定时器 1 下溢时，CNTR1 引脚的 PWM 信号输入就在有效/无效状态间不断切换。但是如果定时器 1 停止，此功能就解除。

在 PWM 信号为高电平“H”期间，即使寄存器 W3 的 bit1 清“0”，定时器 3 就在发生下次下溢之前保持运行状态。

在使用 PWM 输出功能时，如果在停止定时器 3 的情况下，寄存器 W3 的 bit1 清“0”的时序和定时器 3 的下溢重叠，CNTR1 引脚的输出波形就会出现险态。

6. 计数开始同步电路（定时器 1）

定时器 1 具有能与 INTO 引脚的输入同步开始定时器计数运行的计数同步电路。

如果将寄存器 I1 的 bit0 置“1”、INT0 引脚的定时器 1 控制设定为允许状态，并将寄存器 W5 的 bit3 置“1”，就可选择定时器 1 的计数开始同步电路的功能，并可通过 INTO 引脚的输入进行控制。

在使用定时器 1 的计数开始同步电路的情况下，当有效波形输入到 INTO 引脚时，计数开始同步电路就被置位，并输入计数源。

用于计数开始同步电路置位的 INTO 引脚输入的有效波形和外部中断的启动条件相同。

另外，计数开始同步电路如果被置位，可通过将寄存器 I1 的 bit0 清“0”或者通过系统复位来清除。

但是，在选择计数自动停止电路时，如果定时器 1 下溢，就清除计数开始同步电路（自动停止）。

7. 计数自动停止电路（定时器 1）

在使用计数开始同步电路时，定时器 1 具有计数自动停止电路，可通过定时器 1 的下溢来自动停止计数。

如果将寄存器 W1 的 bit3 置“1”，定时器 1 的计数自动停止电路就有效；如果定时器 1 下溢，就清除计数开始同步电路，停止定时器 1 的计数源输入。此功能只在选择定时器 1 计数开始同步电路时有效。

8. 定时器输入/输出引脚 (D4/CNTR0)

CNTR0 引脚具有定时器 1 的计数源输入功能和定时器 1 或者定时器 2 下溢信号的 2 分频信号输出功能。

可以通过寄存器 W5 的 bit0 控制 D4/CNTR0 引脚的功能选择, 通过寄存器 W2 的 bit3 控制输出信号选择。

如果选择 CNTR0 输入作为定时器 1 的计数源, 定时器 1 对 CNTR0 输入的下降波形或者上升波形计数。通过寄存器 W5 的 bit2 选择计数沿。

9. PWM 信号输出功能 (C/CNTR1、定时器 1、定时器 3)

C/CNTR1 引脚具有输出定时器 3 生成的 PWM 信号的输出功能。

如果将寄存器 W3 的 bit3 置“1”, PWM 信号就从 C/CNTR1 引脚输出。此时必须设置端口 C 的输出锁存器为“0”。

10. 定时器中断请求标志 (T1F、T2F、T3F)

定时器中断请求标志在各定时器发生下溢时被置“1”。能通过执行跳越指令 (SNZT1、SNZT2、SNZT3) 确认这些标志的状态。

必须通过寄存器 V1、V2 选择是使用中断还是使用跳越指令。

中断请求标志在发生中断时或者在执行了跳越指令时被清“0”。

11. 注意事项

在使用定时器时, 必须注意以下几点:

●有关预分频器的注意事项

在从预分频器读数据时, 必须首先停止预分频器的计数, 然后执行数据的读指令 (TABPS)。

在将数据写到预分频器时, 必须首先停止预分频器的计数, 然后执行数据的写指令 (TPSAB)。

在切换预分频器的计数源时, 必须先停止预分频器的计数, 然后切换计数源。

●有关计数源的注意事项

在转换定时器 1、2、3 的计数源时, 必须先停止各定时器的计数, 然后转换计数源。

●有关读计数值的注意事项

在从定时器 1、2、3 读数据时, 必须先停止各定时器的计数, 然后执行数据的读指令 (TAB1、TAB2、TAB3)。

●有关给定时器写数据的注意事项

在给定时器 1、2、3 写数据时, 必须先停止各定时器的计数, 然后执行数据的写指令 (T1AB、T2AB、T3AB、T3R3L)。

●有关给重加载寄存器写数据的注意事项

在定时器 1 运行过程中给重加载寄存器 R1 写数据时, 必须在不与定时器 1 下溢重叠的时序执行数据的写指令 (TR1AB)。

在定时器 3 运行过程中给重加载寄存器 R3H 写数据时, 必须在不与定时器 3 下溢重叠的时序执行数据的写指令 (T3HAB)。

●有关 PWM 信号的注意事项

如果在输出 PWM 信号时定时器 3 的计数停止时序和定时器 3 的下溢时序重叠, PWM 输出波形就有可能出现险态。

在 PWM 信号为高电平“H”期间使用扩展功能时, 必须给重加载寄存器 R3H 设定“1”或“1”以上的值。

在从 C/CNTR1 引脚输出 PWM 信号时, 必须将端口 C 的输出锁存器设置为“0”。

●有关预分频器、定时器 1、2、3 的计数开始时序和运行时的计数时间的注意事项

预分频器、定时器在开始运行 (图 33①) 后, 从计数源最初的上升沿 (图 33②) 开始计数。

根据定时器和计数源的运行开始时序, 开始计数后到最初下溢的时间 (图 33③) 短于以后的下溢时间 (图 33④) (最大为计数源的一个周期)。

另外, 如果选择 CNTR0 输入为定时器 1 的计数源, 定时器 1 就与通过软件选择的 CNTR0 输入计数边沿 (下降沿或者上升沿) 同步运行。

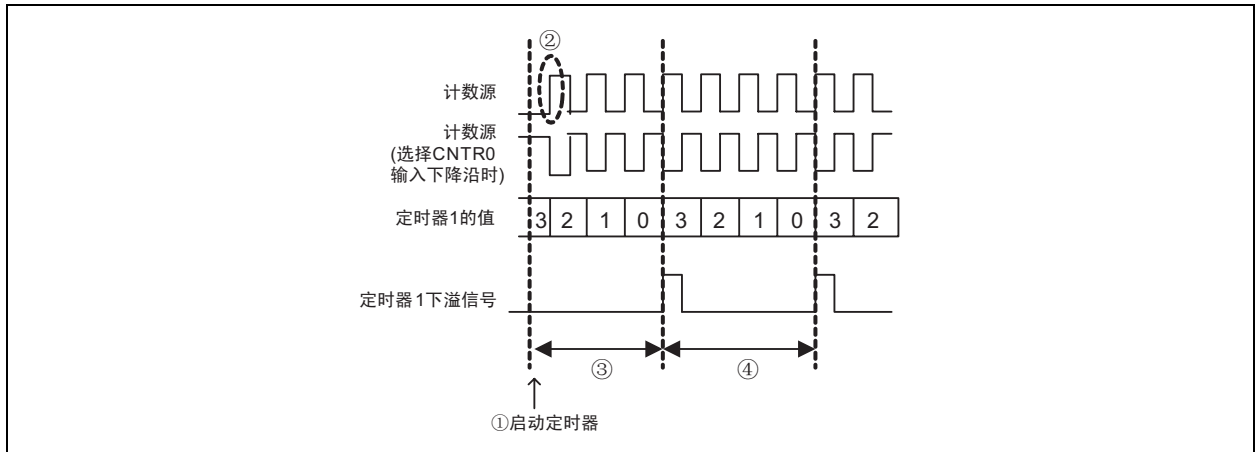


图 33. 定时器的计数开始时序和开始运行时的计数时间

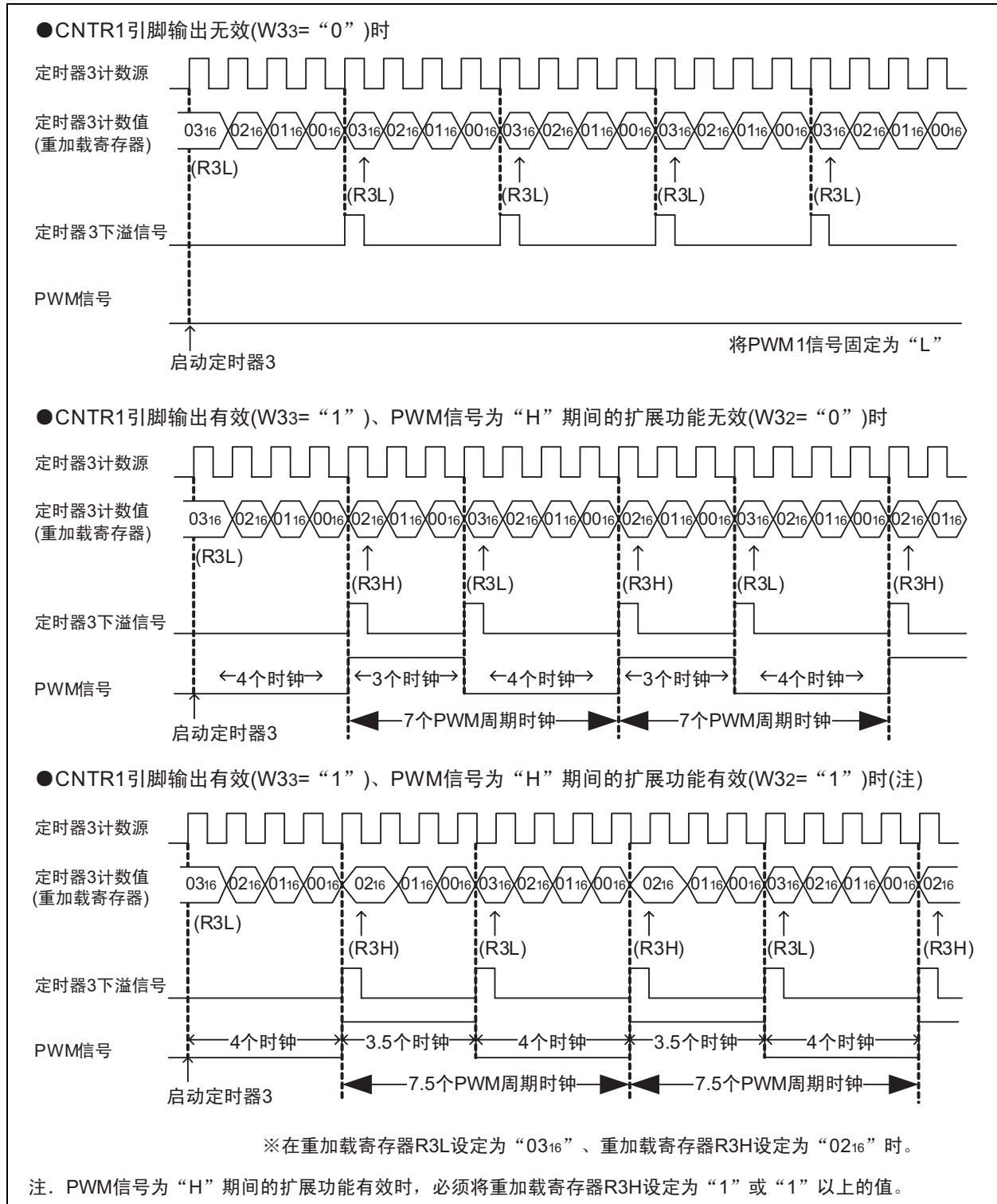


图 34. 定时器 3 的运行例子

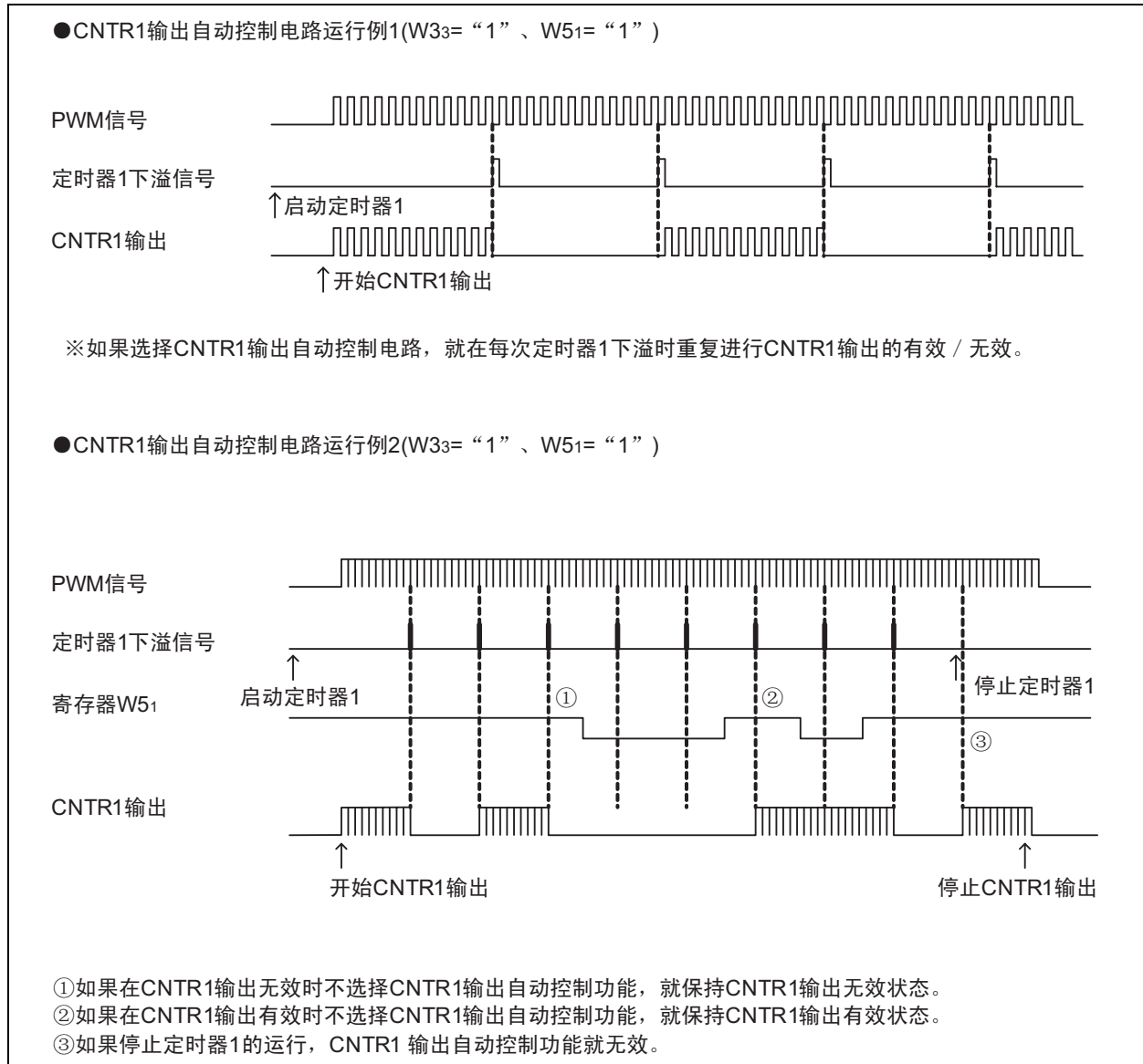


图 35. 定时器 1 的 CNTR1 输出自动控制功能

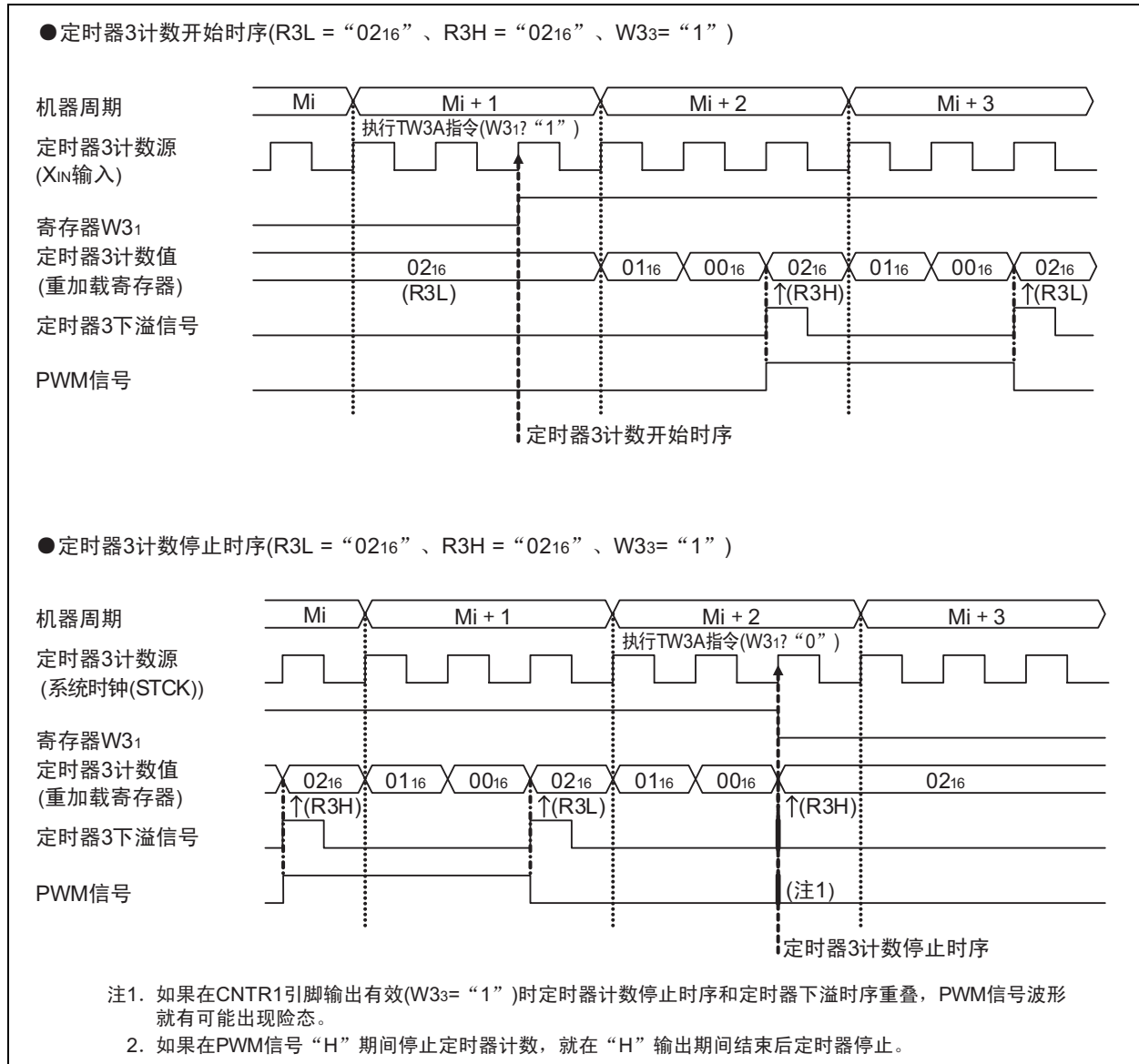


图 36. 定时器计数的开始/停止时序

看门狗定时器

看门狗定时器是在因失控等而无法执行程序的情况下将单片机设定为复位状态并使单片机重新启动的定时器。看门狗定时器由定时器 WDT (16 位二进制计数器)、看门狗定时器允许标志 (WEF) 和看门狗定时器标志 (WDF1、WDF2) 构成。

定时器 WDT 在复位解除后立即将指令时钟作为计数源, 从“ $FFFF_{16}$ ”的值开始递减计数。

在开始计数后, 如果定时器 WDT 下溢 (在定时器 WDT 的内容为“ 0000_{16} ”后, 输入下一个计数脉冲), 就首先将标志 WDF1 置“1”, 然后在下一次定时器 WDT 发生下溢 (定时器 WDT 计数到 65534) 前, 如果不执行 WRST 指令, 就将标志 WDF2 置“1”, 从 $\overline{\text{RESET}}$ 引脚输出“L”电平, 将单片机设定为复位状态。

为了保持单片机正常运行, 必须通过使用看门狗定时器的软件在少于 65534 个的机器周期内执行 WRST 指令进行处理。

在复位解除后, 标志 WEF 被置“1”, 看门狗定时器的功能变为有效。如果连续执行 DWDT 指令和 WRST 指令, 标志 WEF 就被清“0”, 看门狗定时器的功能变为无效。在单片机复位或 RAM 备份模式时, WEF 标志被置“1”。

WRST 指令有跳越功能, 如果在标志 WDF1 的内容为“1”时执行 WRST 指令, 就将标志 WDF1 清“0”, 并跳越下一条指令。在标志 WDF1 为“0”时, 即使执行 WRST 指令也不发生跳越。即使在将看门狗定时器的功能设定为无效的情况下, 也能使用 WRST 指令的跳越功能。

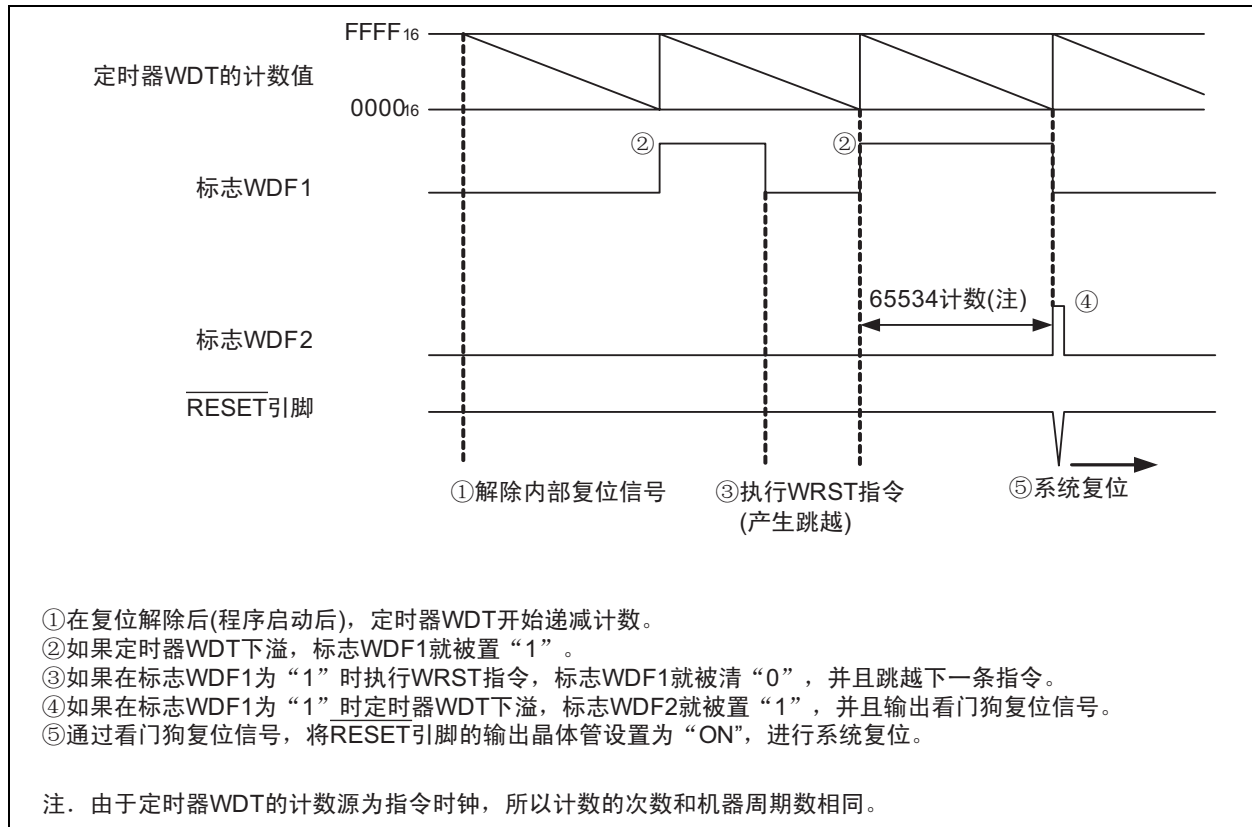


图 37. 看门狗定时器功能的运行

在使用看门狗定时器功能时，必须通过 WRST 指令在少于 65534 个机器周期内清除标志 WDF1。在不使用看门狗定时器功能时，必须连续执行 DWDT 指令和 WRST 指令（参照图 38）。如果只使用 DWDT 指令，就不能停止看门狗定时器功能。

```

      :
      :
      : WRST ;清除标志WDF1
      :
      :
      :
      : DI
      : DWDT ;禁止或者允许看门狗定时器功能
      : WRST ;清除标志WEF和WDF1
      :
      :
  
```

图 38. 使用以及停止看门狗定时器时的程序例子

在 RAM 备份时，标志 WDF1 和定时器 WDT 的值被初始化。另外，在同时使用看门狗定时器功能和 RAM 备份时，必须在转移到 RAM 备份模式前执行 WRST 指令，初始化标志 WDF1。

另外，考虑到因 WRST 指令发生跳越的情况，必须在 WRST 指令后插入 NOP 指令（参照图 39）。

```

      :
      :
      :
      :
      :
      : WRST ;清除标志WDF1
      : NOP
      : DI ;禁止中断
      : EPOF ;允许POF指令
      : POF ;RAM备份
      : ↓
      : 振荡停止
  
```

图 39. 使用看门狗定时器时的程序例子

复位功能

本产品通过以下的操作或者功能进行系统复位：

- 从外部给 $\overline{\text{RESET}}$ 引脚外加“L”电平
- 执行系统复位指令（SRST 指令）
- 通过看门狗定时器产生复位
- 通过内部上电复位电路产生复位
- 通过低电压检测电路产生复位

在系统复位解除后，从页 0 的地址 0 开始执行软件。

1. $\overline{\text{RESET}}$ 引脚输入

当给本产品的 $\overline{\text{RESET}}$ 引脚外加“L”电平时，无论单片机处于任何状态，都进行系统复位。

给 $\overline{\text{RESET}}$ 引脚外加的“L”电平需要电源电压不低于推荐运行条件的最小规格值，并且至少需要 1 个机器周期。

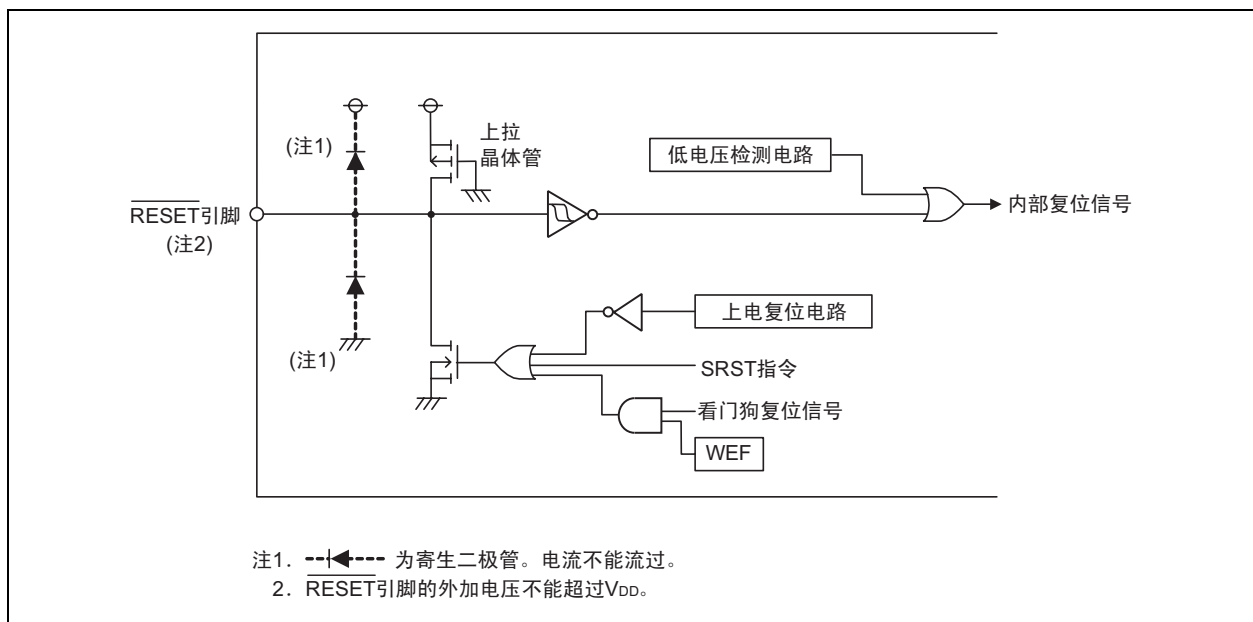


图 40. $\overline{\text{RESET}}$ 引脚外围的结构

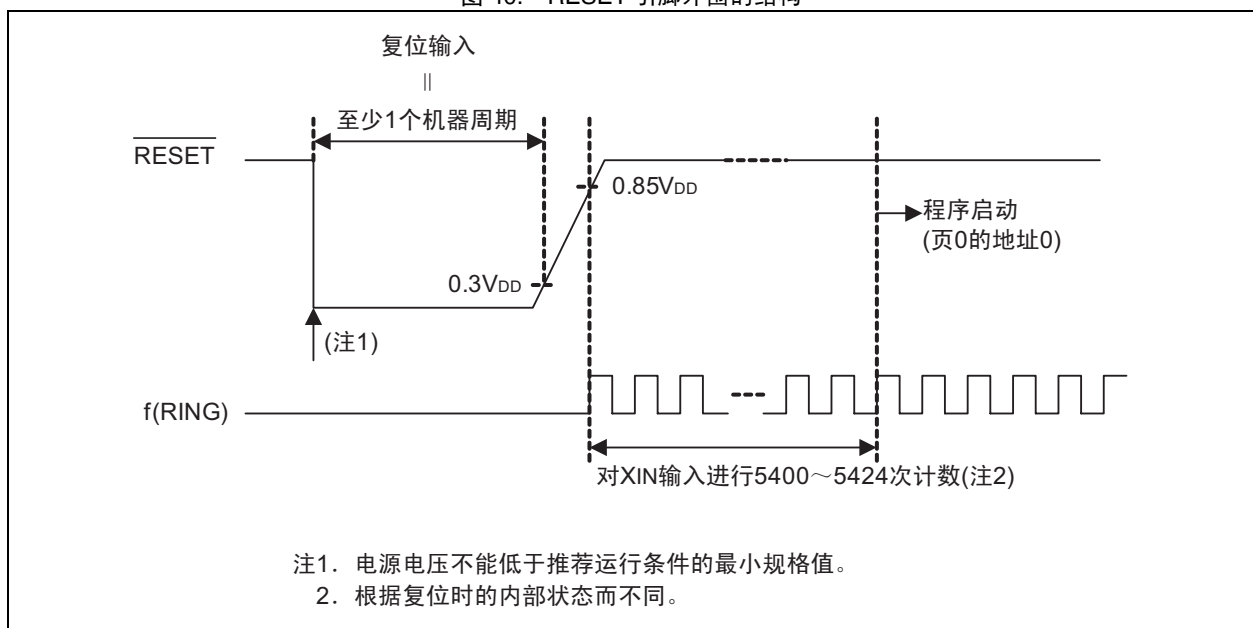


图 41. $\overline{\text{RESET}}$ 引脚的输入波形和复位解除的时序

2. 上电复位

本产品内置有上电复位电路，可在接通电源时自动复位（上电复位）。

在使用内部上电复位电路时，电源电压从 0V 上升到超过推荐运行条件的最小规格值为止的时间不能超过 100 μ s。

在上升时间超过 100 μ s 的情况下，必须在 $\overline{\text{RESET}}$ 引脚和 Vss 之间以最短的距离连接电容，并在电源电压超过推荐运行条件的最小规格值之前将“L”电平输入到 $\overline{\text{RESET}}$ 引脚。

3. 系统复位指令（SRST 指令）

如果执行 SRST 指令，就将“L”电平输出到 $\overline{\text{RESET}}$ 引脚，进行系统复位。

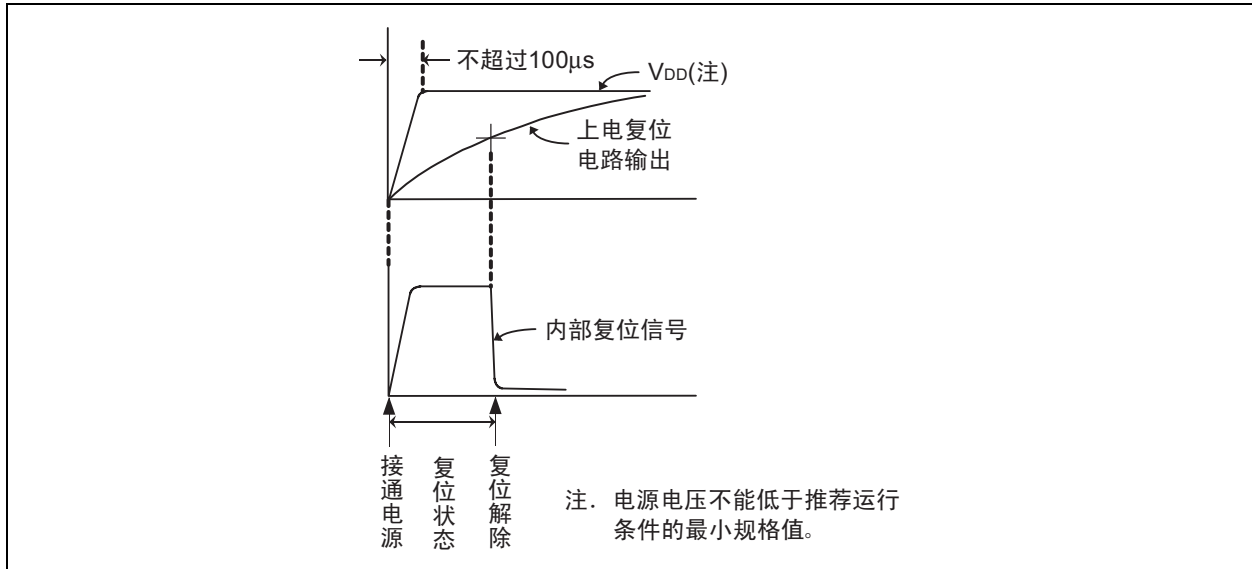


图 42. 上电复位运行

表 18. 复位时的端口状态

端口名	复位时的功能	复位时的状态
D0~D3	D0~D3	高阻抗状态（注 1、注 2）
D4/CNTR0	D4	高阻抗状态（注 1）
P00~P03	P00~P03	高阻抗状态（注 1、注 3）
P10~P13	P10~P13	高阻抗状态（注 1、注 3）
P20/INT0、P21/INT1	P20、P21	高阻抗状态（注 1、注 3）
P30、P31	P30、P31	高阻抗状态（注 1、注 2）
C/CNTR1	C/CNTR1	“L”（Vss）电平
K	K	高阻抗状态

注 1. 输出锁存器被置“1”。

注 2. 输出形式为 N 沟道漏极开路。

注 3. 上拉晶体管为 OFF 状态。

4. 复位时的内部状态

复位时的内部状态如下所示（复位刚解除后也处于相同状态）。因为图 43 以外的定时器、寄存器、标志、RAM 等的内容不定，所以需要进行初始设定。

● 程序计数器(PC) -----	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	
页0的地址0被设定。		
● 中断允许标志(INTE) -----	0	(禁止中断)
● 掉电标志(P) -----	0	
● 外部0中断请求标志(EXF0) -----	0	
● 外部1中断请求标志(EXF1) -----	0	
● 中断控制寄存器V1 -----	0 0 0 0	(禁止中断)
● 中断控制寄存器V2 -----	0 0 0 0	(禁止中断)
● 中断控制寄存器I1 -----	0 0 0 0	
● 中断控制寄存器I2 -----	0 0 0 0	
● 定时器1中断请求标志(T1F) -----	0	
● 定时器2中断请求标志(T2F) -----	0	
● 定时器3中断请求标志(T3F) -----	0	
● 看门狗定时器标志(WDF1、WDF2) -----	0	
● 看门狗定时器允许标志(WEF) -----	1	
● 定时器控制寄存器PA -----	0	(预分频器停止)
● 定时器控制寄存器W1 -----	0 0 0 0	(定时器1停止)
● 定时器控制寄存器W2 -----	0 0 0 0	(定时器2停止)
● 定时器控制寄存器W3 -----	0 0 0 0	(定时器3停止)
● 定时器控制寄存器W5 -----	0 0 0 0	
● 时钟控制寄存器MR -----	1 1 1 1	
● 键唤醒控制寄存器K0 -----	0 0 0 0	
● 键唤醒控制寄存器K1 -----	0 0 0 0	
● 键唤醒控制寄存器K2 -----	0 0 0 0	
● 键唤醒控制寄存器L1 -----	0 0 0 0	
● 上拉控制寄存器PU0 -----	0 0 0 0	
● 上拉控制寄存器PU1 -----	0 0 0 0	
● 上拉控制寄存器PU2 -----	0 0 0 0	
● 端口输出形式控制寄存器FR0 -----	0 0 0 0	
● 端口输出形式控制寄存器FR1 -----	0 0 0 0	
● 进位标志(CY) -----	0	
● 寄存器 A -----	0 0 0 0	
● 寄存器 B -----	0 0 0 0	
● 寄存器 D -----	x x x	
● 寄存器 E -----	x x x x x x x x	
● 寄存器 X -----	0 0 0 0	
● 寄存器 Y -----	0 0 0 0	
● 寄存器 Z -----	x x	
● 堆栈指针(SP) -----	1 1 1	

“x”表示不定。

图 43. 复位时的内部状态

低电压检测电路（带中断功能）

本产品带有低电压检测电路，用于检测电源电压的下降，并且将低电压检测电路中断请求标志（VDF）置位（1）以及进行系统复位。

RAM 备份模式时，低电压检测电路处于停止状态。

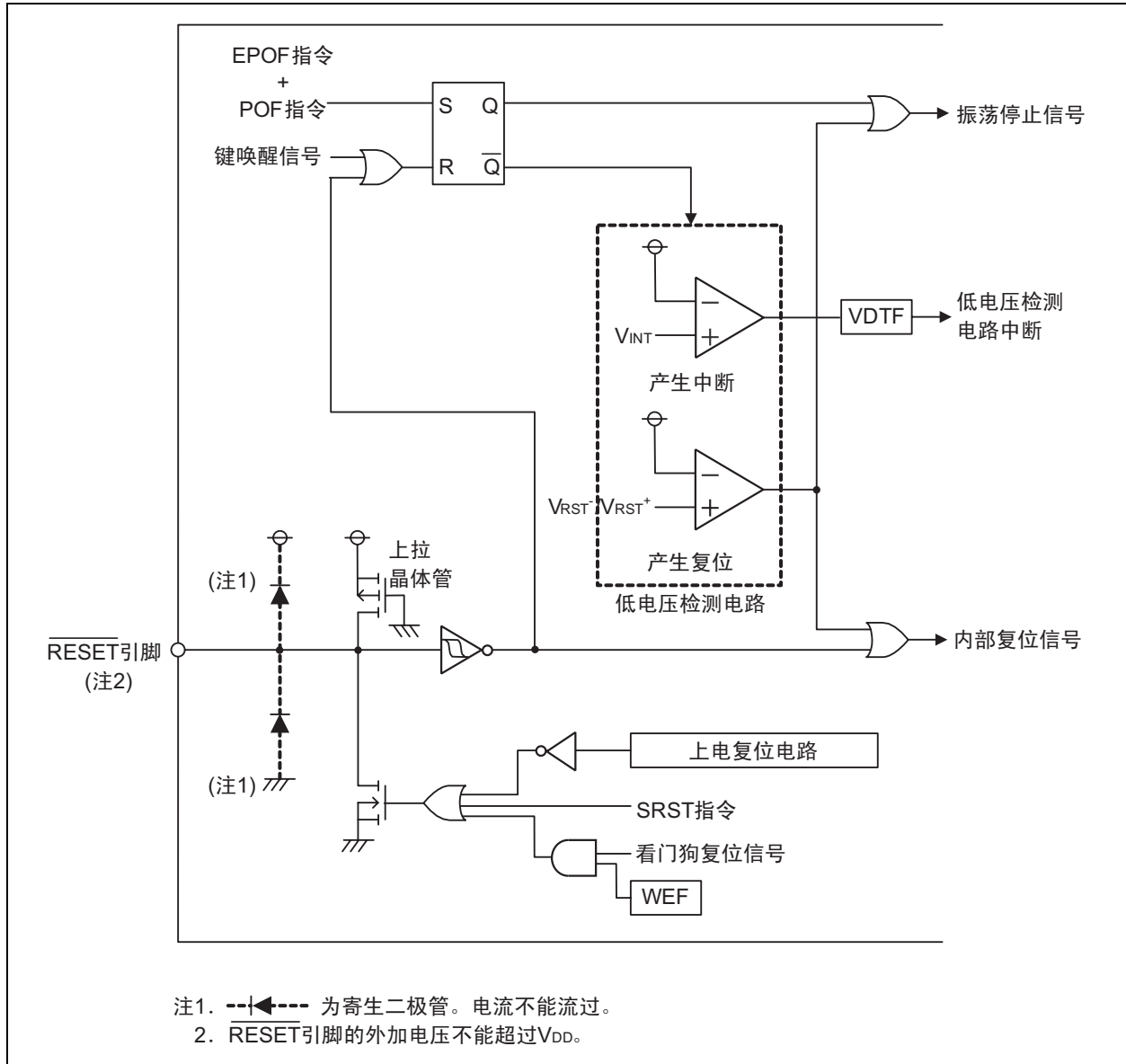


图 44. 低电压检测电路

1. 低电压检测电路中中断请求标志 (VDF)

如果电源电压低于规定值 (V_{INT})，就将低电压检测电路中中断请求标志 (VDF) 置“1”。当电源电压不低于规定值 (V_{INT}) 时，就将低电压检测电路中中断请求标志 (VDF) 清“0”。

通过执行跳越指令 (SNZVD) 可确认中断请求标志的状态。

必须用寄存器 V2 选择是使用中断还是使用跳越指令。

低电压检测电路中中断请求标志与其它中断请求标志不同，即使发生中断请求或者执行跳越指令，该标志也不清“0”。

2. 低电压检测电路复位

如果电源电压低于规定值 (V_{RST^-})，就进行系统复位 (\overline{RESET} 引脚不输出“L”电平)。但是与通常的系统复位不同，振荡电路处于停止状态。

如果电源电压不低于规定值 (V_{RST^+})，振荡电路就处于运行状态并且系统复位被解除。

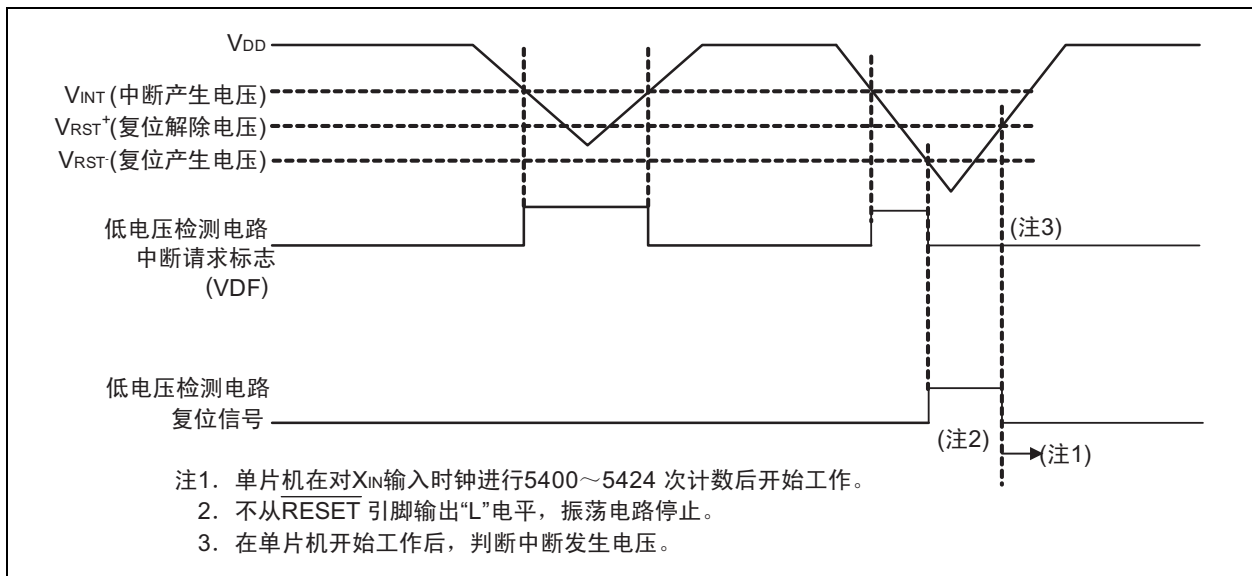


图 45. 低电压检测电路的运行波形

3. 低电压检测电路的注意事项

本产品的低电压检测电路的检测电压设定为低于单片机的电源电压推荐运行条件的下限值。

在应用产品进行电池交换等情况下，当单片机的电源电压降到低于推荐运行条件的下限值后再上升时，根据电源引脚外加的旁路电容的电容值，电源电压没降到复位发生电压 (V_{RST^-}) 以下，所以就不发生复位而继续上升，并且单片机可能会失控 (图 46)。

在这种情况下，必须设计成先使电源电压降到复位发生电压 (V_{RST^-}) 以下再上升的系统。

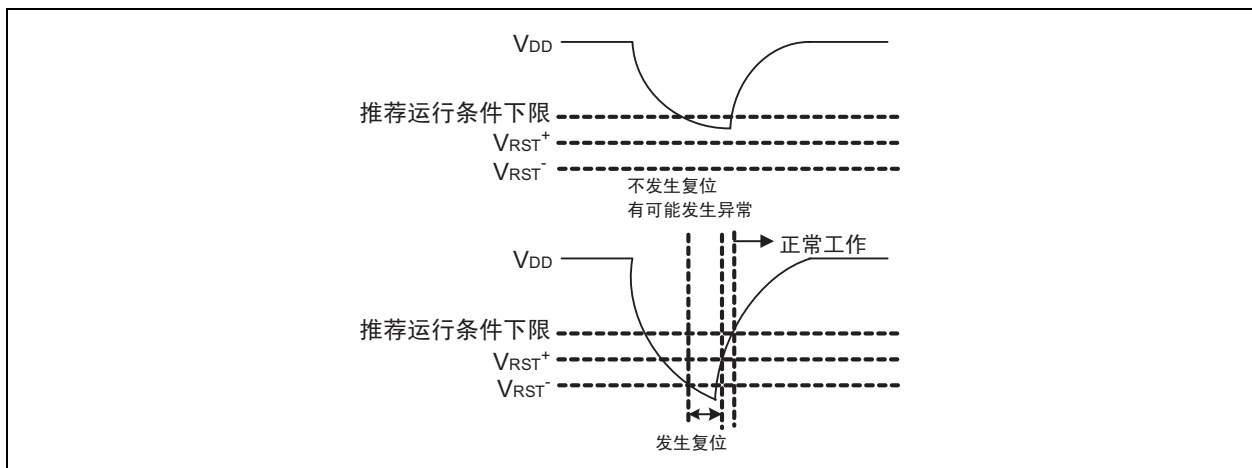


图 46. V_{DD} 和 V_{RST^-}

RAM 备份模式

本产品具有 RAM 备份模式，通过连续执行 EPOF 指令和 POF 指令进入 RAM 备份状态。

如果在执行 POF 指令前不执行 EPOF 指令，POF 指令就和 NOP 指令相同。

在 RAM 备份时，因为在保持 RAM、复位电路的功能和状态下停止振荡，所以 RAM 的数据不会丢失并能减少消耗电流。

RAM 备份时的内部状态如表 19、状态转移图如图 47 所示。

表 19. RAM 备份时保持的功能和状态

功 能	RAM 备份
程序计数器 (PC)	
堆栈指针 (SP) (注 2)	×
进位标志 (CY)	
寄存器 A、B	
RAM 的内容	○
中断控制寄存器 V1、V2	×
中断控制寄存器 I1、I2	○
时钟控制寄存器 MR	×
定时器 1、2、3 功能	(注 3)
看门狗定时器功能	× (注 4)
定时器控制寄存器 PA、W3	×
定时器控制寄存器 W1、W2、W5	○
低电压检测电路	(注 5)
端口的电平	(注 6)
键唤醒控制寄存器 K0~K2、L1	○
上拉控制寄存器 PU0~PU2	○
端口输出形式控制寄存器 FR0~FR1	○
外部中断请求标志 (EXF0、EXF1)	×
定时器中断请求标志 (T1F、T2F、T3F)	(注 3)
低电压检测电路中断请求标志 (VDF)	×
中断允许标志 (INTE)	×
看门狗定时器标志 (WDF1、WDF2)	× (注 4)
看门狗定时器允许标志 (WEF)	× (注 4)

注 1. 表中的“○”表示可保持，“×”表示初始化。上述以外的寄存器和标志的内容在 RAM 备份时不定，必须在返回后设定初始值。

注 2. 因为堆栈指针指向堆栈寄存器的位置，所以在 RAM 备份时被初始化为“7”。

注 3. 定时器的状态不定。

注 4. 必须通过 WRST 指令在初始化看门狗定时器标志 WDF1 后设定为 RAM 备份状态。

注 5. 低电压检测电路变为无效状态。

注 6. C/CNTR1 引脚输出“L”电平。其他端口保持输出电平。

1. 冷启动和热启动的识别

能通过 SNZP 指令检查掉电标志 (P) 的状态，识别热启动 (从 RAM 备份状态的返回) 和冷启动 (从通常复位状态的返回) 的启动条件。

2. 热启动条件

如果在连续执行 EPOF 指令和 POF 指令进入 RAM 备份状态后输入外部唤醒信号，CPU 就从页 0 的地址 0 开始执行程序。此时，掉电标志 (P) 为“1”。

3. 冷启动条件

- 从外部给 $\overline{\text{RESET}}$ 引脚外加“L”电平
- 执行系统复位指令 (SRST 指令)
- 通过看门狗定时器产生复位
- 通过内部上电复位电路产生复位
- 通过低电压检测电路产生复位

满足以上任意条件时，CPU 从页 0 的地址 0 开始执行程序。此时，掉电标志 (P) 为“0”。

4. 返回信号

因为振荡停止，所以通过外部唤醒信号进行从 RAM 备份模式的返回。各返回源的返回条件如表 20 所示。

5. 与键唤醒相关的寄存器

●键唤醒控制寄存器 K0

寄存器 K0 控制端口 P0 的键唤醒功能。

必须由 TK0A 指令通过寄存器 A 设定此寄存器的内容。另外，能通过 TAK0 指令将寄存器 K0 的内容传送到寄存器 A。

●键唤醒控制寄存器 K1

寄存器 K1 控制端口 P1 的键唤醒功能。

必须由 TK1A 指令通过寄存器 A 设定此寄存器的内容。另外，能通过 TAK1 指令将寄存器 K1 的内容传送到寄存器 A。

●键唤醒控制寄存器 K2

寄存器 K2 控制端口 K、P2 的键唤醒功能。

必须由 TK2A 指令通过寄存器 A 设定此寄存器的内容。另外，能通过 TAK2 指令将寄存器 K2 的内容传送到寄存器 A。

●键唤醒控制寄存器 L1

寄存器 L1 控制 INT0 引脚的键唤醒功能和返回条件的选择，以及控制 INT1 引脚的键唤醒功能和返回条件的选择。

必须由 TL1A 指令通过寄存器 A 设定此寄存器的内容。另外，能通过 TAL1 指令将寄存器 L1 的内容传送到寄存器 A。

●上拉控制寄存器 PU0

寄存器 PU0 控制端口 P0 的上拉晶体管的 ON/OFF。

必须由 TPU0A 指令通过寄存器 A 设定此寄存器的内容。另外，能通过 TAPU0 指令将寄存器 PU0 的内容传送到寄存器 A。

●上拉控制寄存器 PU1

寄存器 PU1 控制端口 P1 的上拉晶体管的 ON/OFF。

必须由 TPU1A 指令通过寄存器 A 设定此寄存器的内容。另外，能通过 TAPU1 指令将寄存器 PU1 的内容传送到寄存器 A。

●上拉控制寄存器 PU2

寄存器 PU2 控制端口 P2 的上拉晶体管的 ON/OFF。

必须由 TPU2A 指令通过寄存器 A 设定此寄存器的内容。另外，能通过 TAPU2 指令将寄存器 PU2 的内容传送到寄存器 A。

●中断控制寄存器 I1

寄存器 I1 控制 INT0 引脚的输入、有效波形/电平的选择。

必须由 TI1A 指令通过寄存器 A 设定此寄存器的内容。另外，能通过 TAI1 指令将寄存器 I1 的内容传送到寄存器 A。

表 20. 返回源和返回条件

返回源		返回条件	备注
外部唤醒信号	端口 P0 ₀ ~P0 ₃ 端口 P1 ₀ ~P1 ₃ 端口 P2 ₀ 、P2 ₁ 端口 K	通过外部输入的下降沿 (“H” → “L” 电平) 返回。	能以 1 个端口为单位选择键唤醒功能。 必须在转移到 RAM 备份状态前, 将使用键唤醒的端口设定为 “H” 电平。
	INT 引脚	通过外部输入的 “L” 电平或者 “H” 电平、或者下降沿 (“H” → “L” 电平) 或者上升沿 (“L” → “H” 电平) 返回。 在返回输入时, 中断请求标志 (EXF0) 不被置位。	能以 1 个端口为单位选择键唤醒功能。 必须在转移到 RAM 备份状态前, 根据外部的状态通过中断控制寄存器 I1 选择返回电平 (“L” 电平或者 “H” 电平), 通过键唤醒控制寄存器 L1 选择返回条件 (电平返回或者边沿返回)。

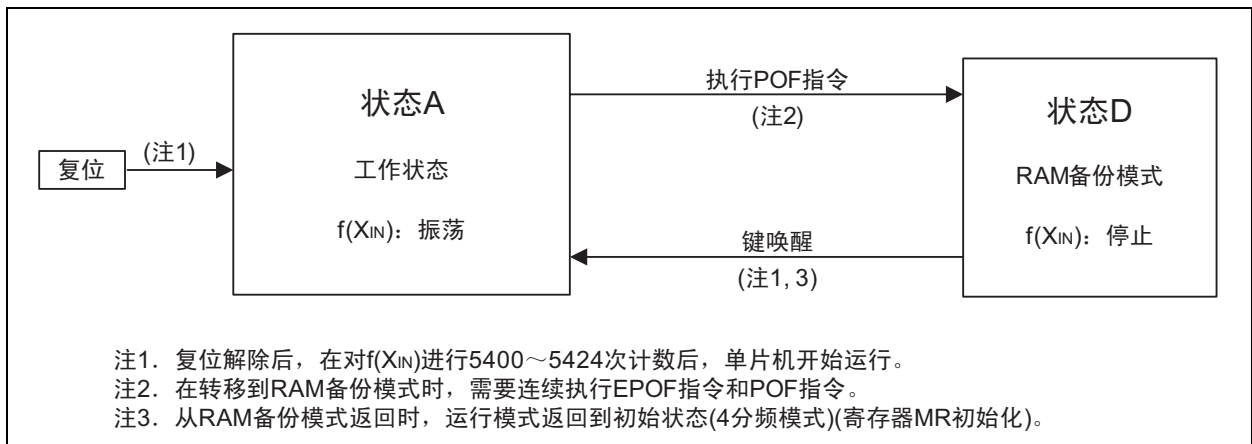


图 47. 状态转移图

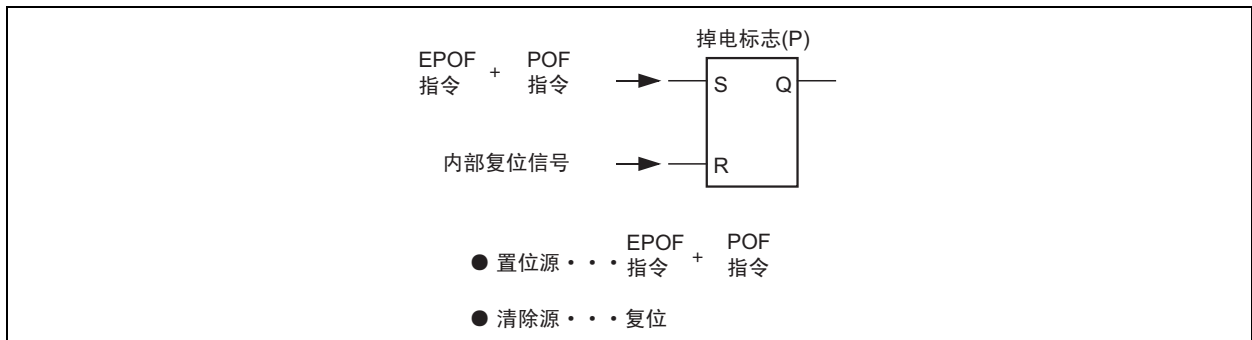


图 48. 掉电标志 (P) 的置位源和清除源

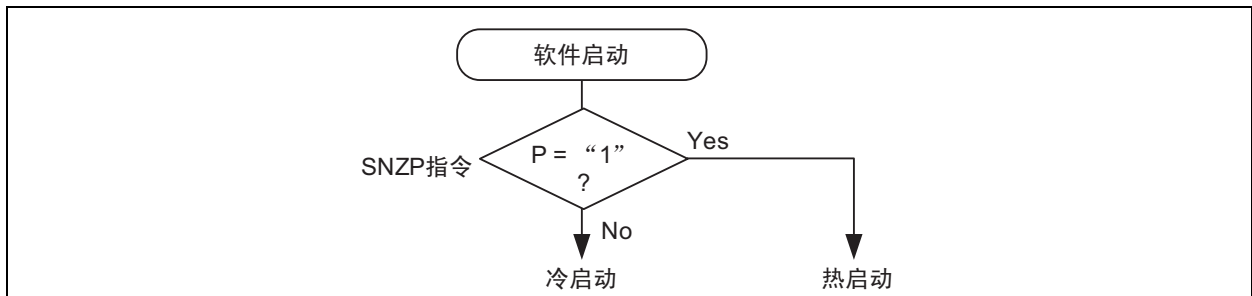


图 49. SNZP 指令的启动识别例子

表 21. 键唤醒控制寄存器

键唤醒控制寄存器 K0		复位时: 0000 ₂		RAM 备份时: 保持状态	R/W TAK0/TK0A
K0 ₃	端口 P0 ₃ 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		
K0 ₂	端口 P0 ₂ 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		
K0 ₁	端口 P0 ₁ 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		
K0 ₀	端口 P0 ₀ 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		

键唤醒控制寄存器 K1		复位时: 0000 ₂		RAM 备份时: 保持状态	R/W TAK1/TK1A
K1 ₃	端口 P1 ₃ 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		
K1 ₂	端口 P1 ₂ 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		
K1 ₁	端口 P1 ₁ 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		
K1 ₀	端口 P1 ₀ 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		

键唤醒控制寄存器 K2		复位时: 0000 ₂		RAM 备份时: 保持状态	R/W TAK2/TK2A
K2 ₃	不使用	0	此位没有功能、可进行读写操作		
		1			
K2 ₂	端口 K 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		
K2 ₁	端口 P2 ₁ 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		
K2 ₀	端口 P2 ₀ 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		

键唤醒控制寄存器 L1		复位时: 0000 ₂		RAM 备份时: 保持状态	R/W TAL1/TL1A
L1 ₃	INT1 引脚返回条件选择位	0	电平返回		
		1	边沿返回		
L1 ₂	INT1 引脚有效波形/电平选择位	0	键唤醒无效		
		1	键唤醒有效		
L1 ₁	INT0 引脚返回条件选择位	0	电平返回		
		1	边沿返回		
L1 ₀	INT0 引脚键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		

注 1. “R”表示可读, “W”表示可写。

表 22. 上拉控制寄存器

上拉控制寄存器 PU0		复位时: 00002		RAM 备份时: 保持状态	R/W TAPU0/TPU0A
PU03	端口 P03 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		
PU02	端口 P02 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		
PU01	端口 P01 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		
PU00	端口 P00 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		

上拉控制寄存器 PU1		复位时: 00002		RAM 备份时: 保持状态	R/W TAPU1/TPU1A
PU13	端口 P13 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		
PU12	端口 P12 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		
PU11	端口 P11 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		
PU10	端口 P10 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		

上拉控制寄存器 PU2		复位时: 00002		RAM 备份时: 保持状态	R/W TAPU2/TPU2A
PU23	不使用	0	此位没有功能、可进行读写操作		
		1			
PU22	不使用	0	此位没有功能、可进行读写操作		
		1			
PU21	端口 P21 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		
PU20	端口 P20 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		

注 1. “R” 表示可读, “W” 表示可写。

时钟控制

本产品的时钟控制电路由以下电路构成：

- 陶瓷振荡电路
- 分频电路
- 内部时钟发生电路

由这些电路生成本产品运行源的系统时钟和指令时钟。

时钟控制电路的结构如图 50 所示。

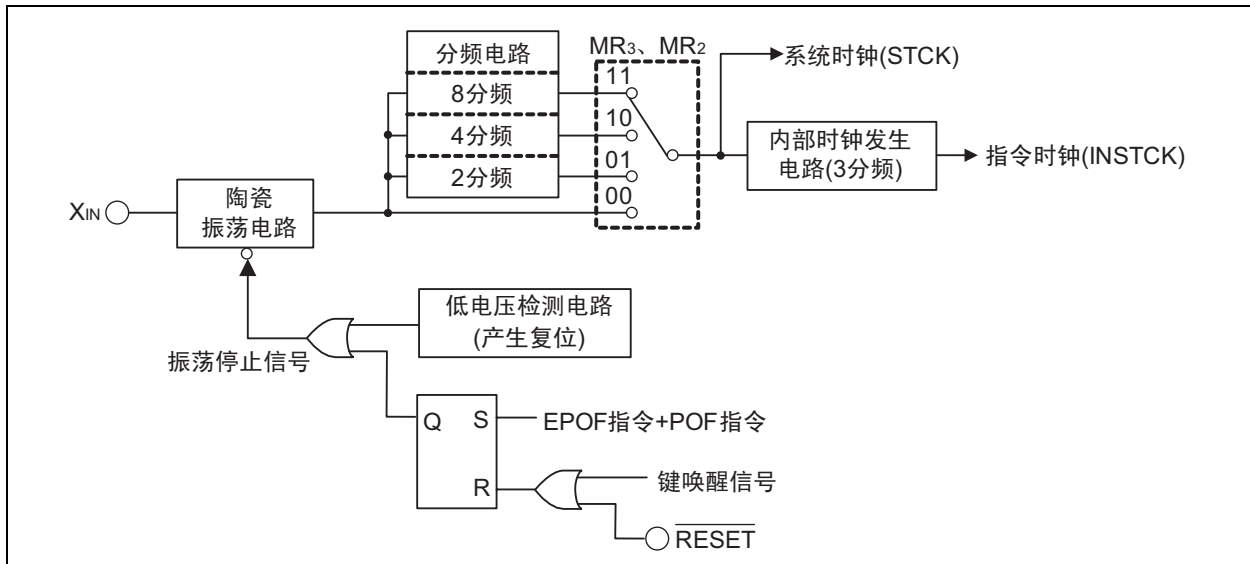


图 50. 时钟控制电路的结构

1. 使用陶瓷谐振器时

在主时钟 ($f(XIN)$) 使用陶瓷谐振器时，必须以最短的距离将陶瓷谐振器和外部电路连接到 XIN 引脚和 $XOUT$ 引脚。在 XIN 引脚和 $XOUT$ 引脚之间内置了反馈电阻 (图 51)。

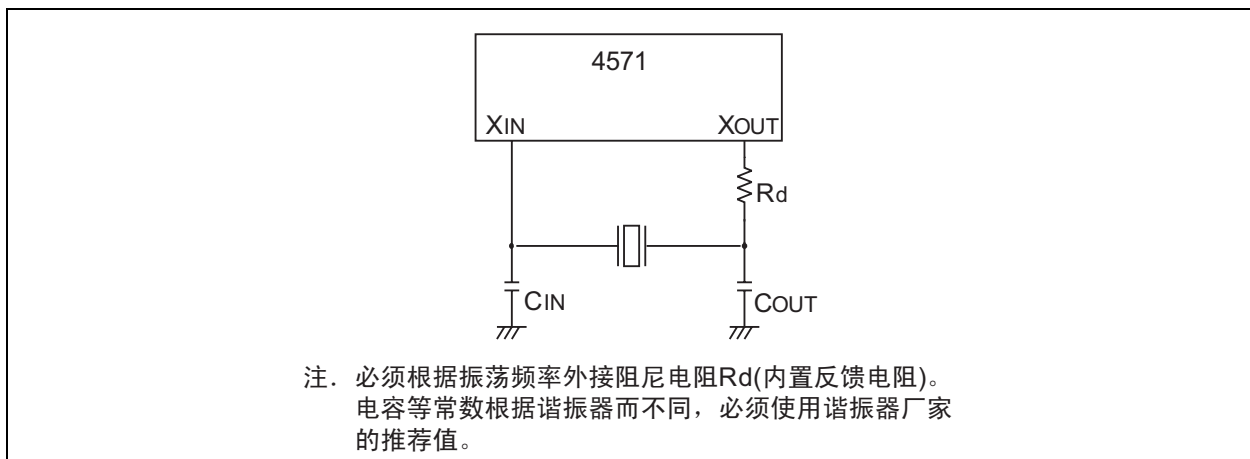


图 51. 陶瓷谐振器外接电路

2. 使用外部时钟时

在主时钟 ($f(XIN)$) 使用外部时钟时，必须将时钟发生源连接到 XIN 引脚，并将 $XOUT$ 引脚置为开路。

使用外部时钟时的振荡频率最大值和使用陶瓷谐振器时不同，必须注意 (参照推荐运行条件)。

另外，在使用外部时钟时，不能使用 RAM 备份模式 (POF 指令)。

3. 时钟控制寄存器 MR

寄存器 MR 控制运行模式（系统时钟分频）的选择。

必须由 TMRA 指令通过寄存器 A 设定此寄存器的内容。另外，能通过 TAMR 指令将寄存器 MR 的内容传送到寄存器 A。

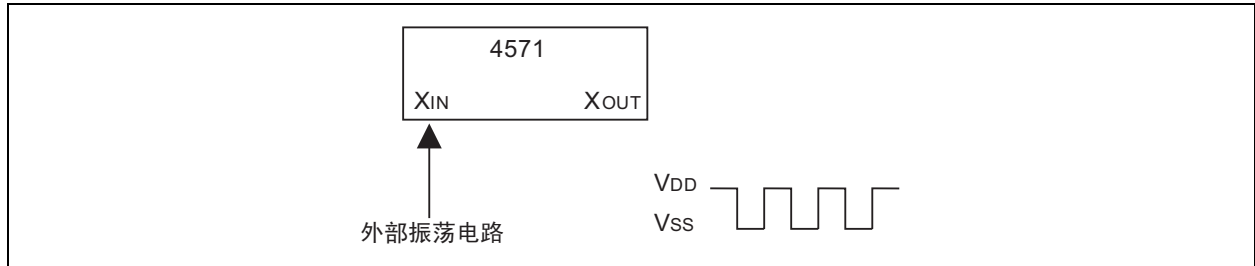


图 52. 外部时钟输入电路

表 23. 时钟控制寄存器

时钟控制寄存器 MR		复位时: 11112		RAM 备份时: 11112	R/W TAMR/TMRA
MR3	运行模式选择位	MR3	MR2	运行模式	
		0	0	through-mode (无分频)	
		0	1	2 分频模式	
		1	0	4 分频模式	
MR2		1	1	8 分频模式	
		0		此位没有功能、可进行读写操作	
MR1	不使用	1			
		0			
MR0	不使用	0		此位没有功能、可进行读写操作	
		1			

注 1. “R”表示可读，“W”表示可写。

QzROM 编程模式

在 QzROM 编程模式中，能使用与本单片机对应的串行编程器，在单片机安装在电路板的状态下对用户 ROM 区进行编程。

引脚的功能说明（QzROM 编程模式）如表 24 所示，引脚连接图如图 53 所示。

与串行编程器连接的例子请参照电路板上引脚处理的例子（图 54）。有关串行编程器请向各厂家询问，有关串行编程器的操作方法请参照串行编程器的用户手册。

表 24 引脚的功能说明（QzROM 编程模式）

引脚名	名称	输入/输出	功能
VDD	电源	—	供给电源电压的引脚。
VSS	GND	—	GND 引脚。
K	VPP 输入	—	QzROM 的电源输入引脚。
P01	SDA 输入/输出	输入/输出	QzROM 的串行数据的输入/输出引脚。
P00	SCLK 输入	输入	QzROM 的串行时钟的输入引脚。
P10	PGM 输入	输入	QzROM 的读/编程脉冲信号的输入引脚。
RESET	复位输入	输入	是复位输入引脚，必须输入“L”电平。
XIN	时钟输入	—	必须连接振荡电路，或者将 XIN 引脚连接到 VSS 并且将 XOUT 引脚置为开路。
XOUT	时钟输出	—	
P02、P03、P11~P13、 P20/INT0、P21/INT1、 P30、P31、D0~D3、 D4/CNTR0、C/CNTR1	输入/输出端口	—	必须输入“H”电平或者“L”电平，或者置为开路。

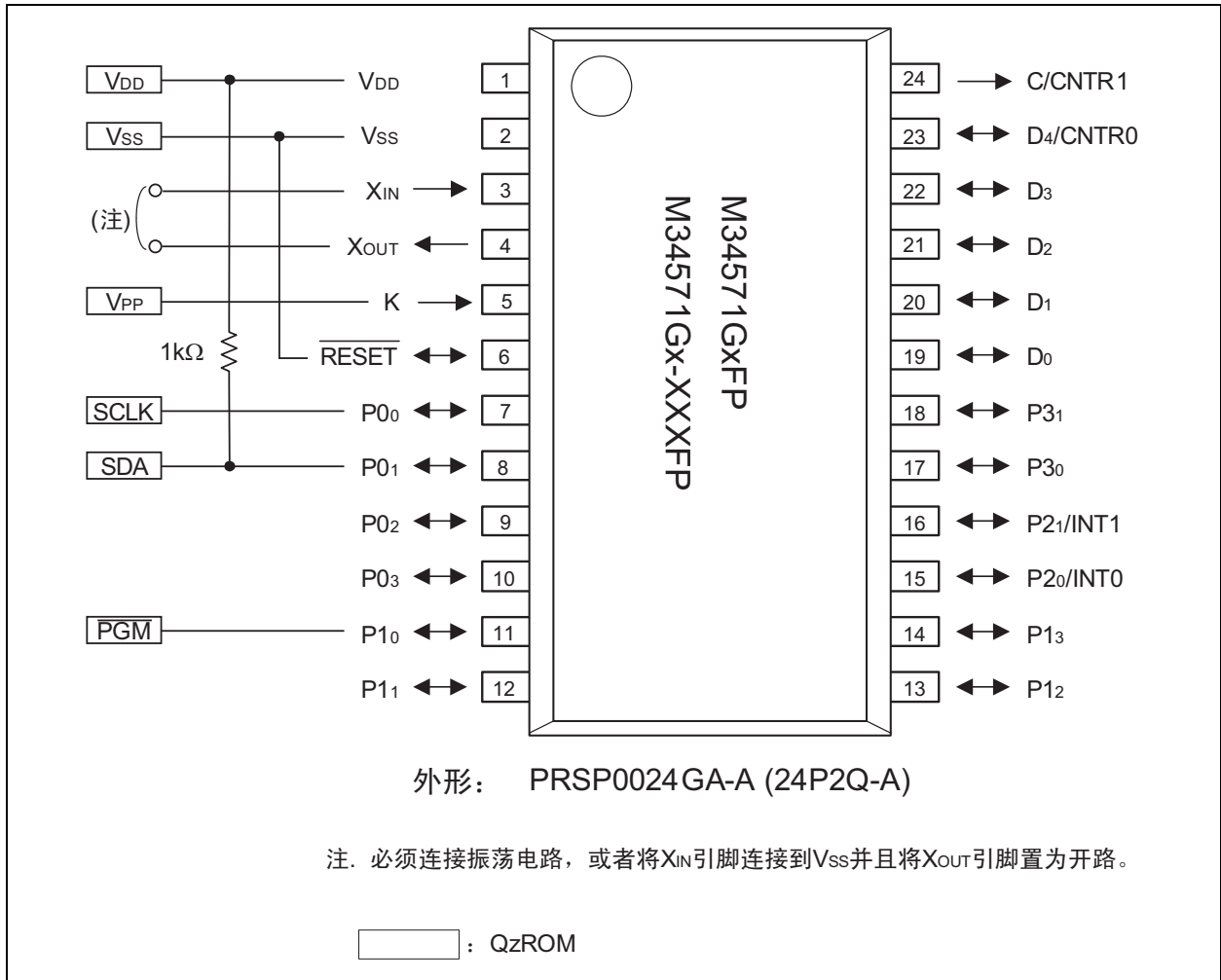


图 53 引脚连接图

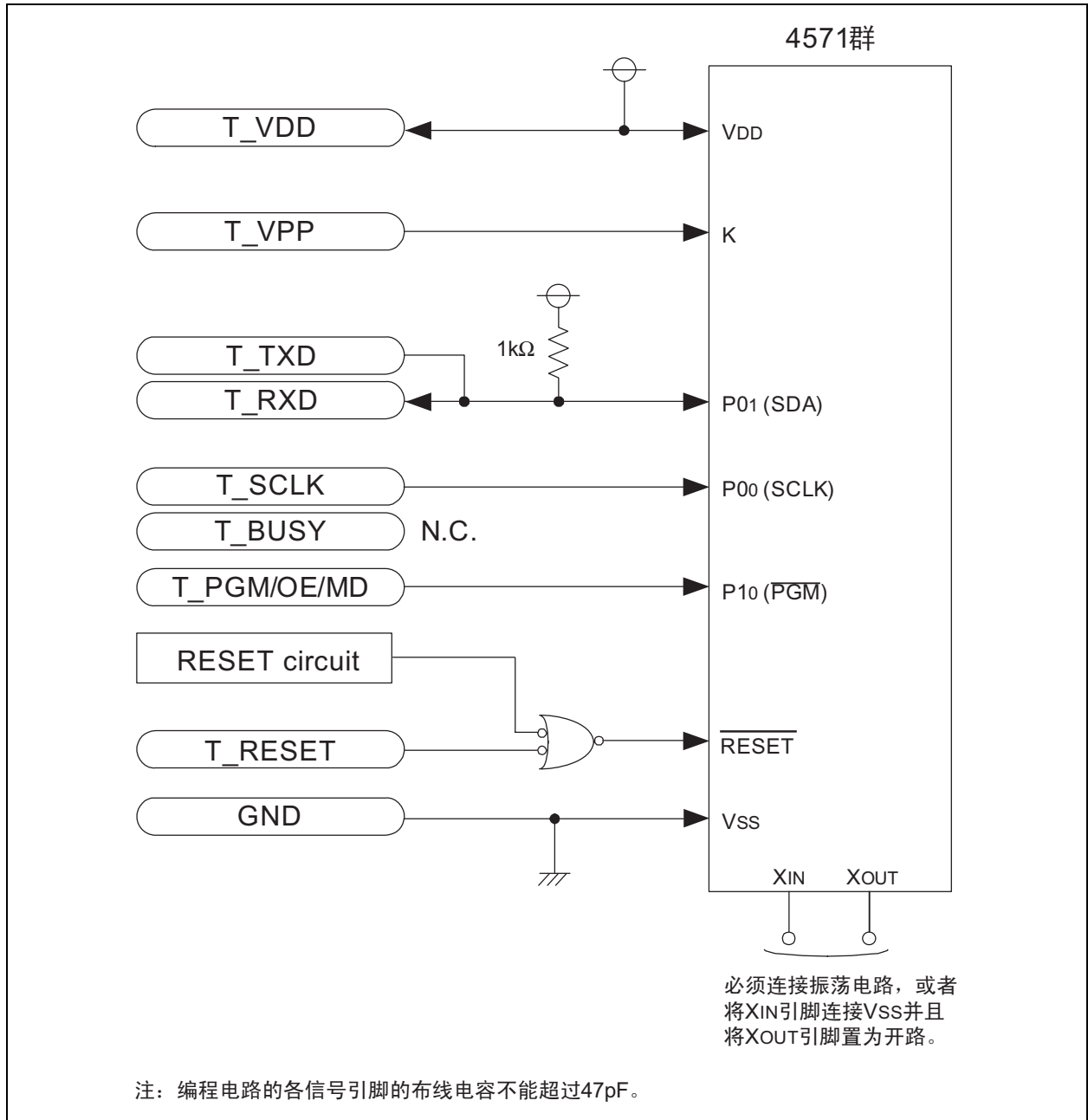


图 54 使用彗星电子系统公司产的编程器时的电路板上引脚处理的例子

订购 QzROM 编程后产品时的提交资料

必须在订购 QzROM 编程后出货产品时提交以下资料：

- QzROM 编程确认书*
- 标记指定书*
- ROM 的数据 . . . 掩模文件

*有关 QzROM 编程确认书和标记指定书，请参照瑞萨科技公司主页 (<http://www.renesas.com/homepage.jsp>)。另外，QzROM 单片机不对应特殊字体标记（贵公司商标等）。

使用时的注意事项（总结）

1. 噪声和门锁对策

作为噪声和门锁对策，必须在 VDD 引脚和 VSS 引脚之间以最短距离、同等宽度、同等布线长度并且尽可能使用粗的布线连接电容（ $\approx 0.1\mu\text{F}$ ）。

端口 K 可复用为 VPP 引脚。端口 K 不能开路，必须接到 VDD 引脚或者 VSS 引脚。使用键矩阵时，必须通过上拉电阻连接到 VDD 引脚。

2. 电源电压

在单片机的电源电压低于推荐运行条件的最小规格值时，单片机可能无法正常运行，处于不稳定的运行状态。

对于在电源电压下降和切断电源时电源电压缓慢下降的系统，系统设计时必须考虑即使在电源电压低于推荐运行条件的最小规格值时的不稳定运行状态下也能保证系统正常的单片机复位等对策。

3. 寄存器的初始值 1

以下寄存器在复位解除后初始值不定，必须在复位解除后进行初始设定：

- 寄存器 Z（2 位）
- 寄存器 D（3 位）
- 寄存器 E（8 位）

4. 寄存器的初始值 2

以下寄存器在 RAM 备份时值不定，必须在从 RAM 备份返回后重新设定：

- 寄存器 Z（2 位）
- 寄存器 X（4 位）
- 寄存器 Y（4 位）
- 寄存器 D（3 位）
- 寄存器 E（8 位）

5. 程序计数器

必须注意：程序计数器指定的页不能超过内部 ROM 的最后页。

6. 堆栈寄存器（SK）

因为堆栈寄存器（SK）由 8 段构成，所以最多能使用 8 级子程序。但是，在使用中断处理程序以及执行表参照指令（TABPp）时，因为分别使用 1 段寄存器 SK，所以必须注意：在同时使用这些处理时，总计不能超过 8 级。

7. 多功能

- 即使在使用 INT0 引脚的情况下，端口 P20 的输入/输出功能也有效。因为 INT0 引脚和端口 P20 的输入阈值不同，所以在同时使用两者的输入时必须注意。
- 即使在使用 INT1 引脚的情况下，端口 P21 的输入/输出功能也有效。因为 INT1 引脚和端口 P21 的输入阈值不同，所以在同时使用两者的输入时必须注意。
- 即使在使用 CNTR0 引脚的输出功能的情况下，端口 D4 的输入功能也有效；即使在使用 CNTR0 引脚的输入功能的情况下，端口 D4 的输入/输出功能也有效。CNTR0 引脚和端口 D4 的输入阈值不同，所以在同时使用两者的输入时必须注意。
- 即使在使用 CNTR1 引脚的输出功能的情况下，端口 C 的“H”输出功能也有效。

8. 上电复位

使用内置上电复位电路时，电源电压从 0V 上升到超过推荐运行条件的最小规格值为止的时间不能超过 100 μs 。

在上升时间超过 100 μs 的情况下，必须在 $\overline{\text{RESET}}$ 引脚和 VSS 之间以最短的距离连接电容，并在电源电压超过推荐运行条件的最小规格值之前将“L”电平输入到 $\overline{\text{RESET}}$ 引脚。

9. POF 指令

如果在执行 EPOF 指令后立即执行 POF 指令，就进入 RAM 备份状态。

必须注意：只用 POF 指令不能进入 RAM 备份状态。

另外，在连续执行 EPOF 指令和 POF 指令前，必须设定为中断禁止状态（执行 DI 指令）。

10. P20/INT0 引脚

(1) 有关寄存器 I1 的 bit3 的注意事项 1

在软件中通过中断控制寄存器 I1 的 bit3 控制 INT0 引脚的输入时，必须注意：

- 在更改寄存器 I1 的 bit3 的内容时，有时需要根据 P20/INT0 引脚的输入状态将外部 0 中断请求标志（EXF0）置“1”。为了防止发生意外的中断，必须在将中断控制寄存器 V1 的 bit0 清“0”（图 55①）后更改寄存器 I1 的 bit3 的内容。而且，必须至少间隔一条指令后（图 55②）执行 SNZ0 指令，将标志 EXF0 清“0”。另外，考虑到因 SNZ0 指令发生跳越的情况，必须在 SNZ0 指令后插入 NOP 指令（图 55③）。

⋮	
LA 4	; (×××0 ₂)
TV1A	; SNZ0指令有效⋯⋯①
LA 8	; (1×××2)
TI1A	; 改变INT0引脚输入控制
NOP	; ⋯⋯⋯⋯⋯⋯②
SNZ0	; 执行SNZ0指令 (清除标志EXF0)
NOP	; ⋯⋯⋯⋯⋯⋯③
⋮	
	×：此位与本例无关。

图 55. 外部 0 中断程序例 1

(2) 有关寄存器 I1 的 bit3 的注意事项 2

在将中断控制寄存器 I1 的 bit3 清“0”并且在禁止 INT0 引脚输入的状态下使用 RAM 备份时，必须注意：

- 在禁止 INT0 引脚输入时（寄存器 I13=“0”），必须在转移到 RAM 备份模式前将 INT0 引脚的键唤醒设定为无效（寄存器 L10=“0”）（图 56①）。

⋮	
LA 0	; (×0××2)
TL1A	; INT0键唤醒无效⋯⋯①
DI	
EPOF	
POF	; RAM备份
⋮	
	×：此位与本例无关。

图 56. 外部 0 中断程序例 2

(3) 有关寄存器 I1 的 bit2 的注意事项

在软件中通过中断控制寄存器 I1 的 bit2 更改 P20/INT0 引脚的中断有效波形时，必须注意：

- 在更改寄存器 I1 的 bit2 的内容时，有时需要根据 P20/INT0 引脚的输入状态将外部 0 中断请求标志（EXF0）置“1”。为了防止发生意外的中断，必须在将中断控制寄存器 V1 的 bit0 清“0”（图 57①）后更改寄存器 I1 的 bit2 的内容。而且，必须至少间隔一条指令后执行（图 57②）SNZ0 指令，将标志 EXF0 清“0”。另外，考虑到因 SNZ0 指令发生跳越的情况，必须在 SNZ0 指令后插入 NOP 指令（图 57③）。

⋮	
LA 4	; (×××0 ₂)
TV1A	; SNZ0指令有效···①
LA 12	; (1××× ₂)
TI1A	; 改变中断有效波形
NOP	; ·········②
SNZ0	; 执行SNZ0指令 (清除标志EXF0)
NOP	; ·········③
⋮	
	×: 此位与本例无关。

图 57. 外部 0 中断程序例 3

11. P21/INT1 引脚

(1) 有关寄存器 I2 的 bit3 的注意事项 1

在软件中通过中断控制寄存器 I2 的 bit3 控制 INT1 引脚的输入时，必须注意：

- 在更改寄存器 I2 的 bit3 的内容时，有时需要根据 P21/INT1 引脚的输入状态将外部 1 中断请求标志 (EXF1) 置“1”。为了防止发生意外的中断，必须在将中断控制寄存器 V1 的 bit1 清“0” (图 58①) 后更改寄存器 I2 的 bit3 的内容。而且，必须至少间隔一条指令后执行 (图 58②) SNZ1 指令，将标志 EXF1 清“0”。另外，考虑到因 SNZ1 指令发生跳越的情况，必须在 SNZ1 指令后插入 NOP 指令 (图 58③)。

⋮	
LA 4	; (××0× ₂)
TV1A	; SNZ1指令有效···①
LA 8	; (1××× ₂)
TI2A	; 改变INT1引脚输入控制
NOP	; ·········②
SNZ1	; 执行SNZ1指令 (清除标志EXF1)
NOP	; ·········③
⋮	
	×: 此位与本例无关。

图 58. 外部 1 中断程序例 1

(2) 有关寄存器 I2 的 bit3 的注意事项 2

在将中断控制寄存器 I2 的 bit3 清“0”并且在禁止 INT1 引脚输入的状态下使用 RAM 备份时，必须注意：

- 在禁止INT1引脚输入时 (寄存器I23=“0”)，必须在转移到RAM备份模式前将INT1引脚的键唤醒设定为无效 (寄存器L12=“0”) (图59①)。

```

      ⋮
LA 0  ;(××0×2)
TL1A  ;INT1键唤醒无效...①
DI
EPOF
POF   ;RAM备份
      ⋮
      ×：此位与本例无关。

```

图 59. 外部 1 中断程序例 2

(3) 有关寄存器 I2 的 bit2 的注意事项

在软件中通过中断控制寄存器 I2 的 bit2 更改 P2i/INT1 引脚的中断有效波形时，必须注意：

- 在更改寄存器 I2 的 bit2 的内容时，有时需要根据 P2i/INT1 引脚的输入状态将外部 1 中断请求标志 (EXF1) 置“1”。为了防止发生意外的中断，必须在将中断控制寄存器 V1 的 bit1 清“0” (图 60①) 后更改寄存器 I2 的 bit2 的内容。而且，必须至少间隔一条指令后执行 (图 60②) SNZ1 指令，将标志 EXF1 清“0”。另外，考虑到因 SNZ1 指令发生跳越的情况，必须在 SNZ1 指令后插入 NOP 指令 (图 60③)。

```

      ⋮
LA 4  ;(××0×2)
TV1A  ;SNZ1指令有效...①
LA 12 ;(1×××2)
TI2A  ;改变中断有效波形
NOP   ;.....②
SNZ1  ;执行SNZ1指令
      (清除标志EXF1)
NOP   ;.....③
      ⋮
      ×：此位与本例无关。

```

图 60. 外部 1 中断程序例 3

12. 预分频器

在从预分频器读数据时，必须首先停止预分频器的计数，然后执行数据的读指令 (TABPS)。
在将数据写到预分频器时，必须首先停止预分频器的计数，然后执行数据的写指令 (TPSAB)。

13. 计数源

在转换定时器 1、2、3 的计数源时，必须首先停止各定时器的计数，然后转换计数源。

14. 读计数值

在从定时器 1、2、3 读数据时，必须首先停止各定时器的计数，然后执行数据的读指令 (TAB1、TAB2、TAB3)。

15. 给定时器写数据

在给定时器 1、2、3 写数据时，必须首先停止各定时器的计数，然后执行数据的写指令 (T1AB、T2AB、T3AB、T3R3L)。

16. 给重加载寄存器写数据

在定时器 1 运行过程中给重加载寄存器 R1 写数据时，必须在不与定时器 1 下溢重叠的时序执行数据的写指令 (TR1AB)。

在定时器 3 运行过程中给重加载寄存器 R3H 写数据时，必须在不与定时器 3 下溢重叠的时序执行数据的写指令 (T3HAB)。

17. PWM 信号

如果在输出 PWM 信号时定时器 3 的计数停止时序和定时器 3 的下溢时序重叠，PWM 输出波形就有可能发生险态。在 PWM 信号为高电平“H”期间使用扩展功能时，必须给重加载寄存器 R3H 设定“1”或“1”以上的值。从 C/CNTR1 引脚输出 PWM 信号时，必须把端口 C 的输出锁存器清“0”。

18. 预分频器、定时器 1、2、3 的计数开始时序和开始运行时的计数时间

预分频器、定时器在开始运行 (①) 后，从计数源的最初的上升沿 (②) 开始计数。

根据定时器和计数源的运行开始时序，开始计数后到最初下溢的时间 (③) 短于以后的下溢时间 (④) (最大为计数源的一个周期)。

另外，如果选择 CNTR0 输入为定时器 1 的计数源，定时器 1 就与通过软件选择的 CNTR0 输入计数边沿 (下降沿或者上升沿) 同步运行。

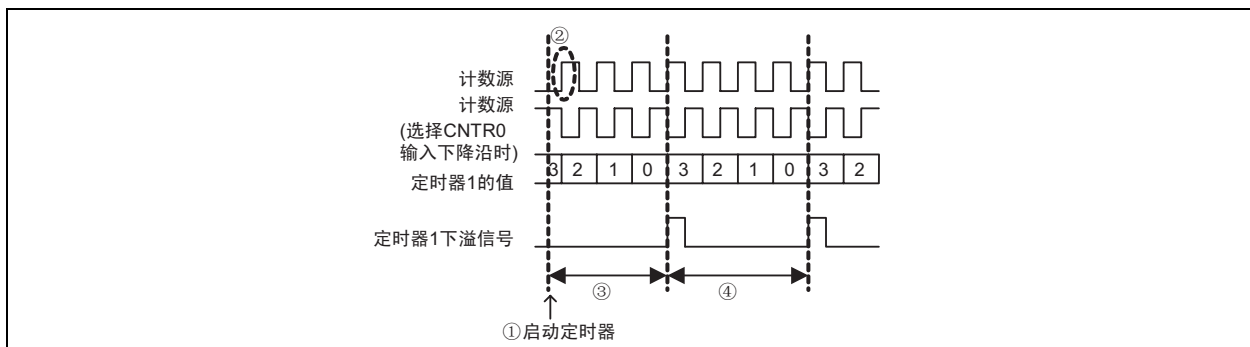


图 61. 定时器的计数开始时序和开始运行时的计数时间

19. 看门狗定时器

- 看门狗定时器功能在复位解除后有效。如果不使用看门狗定时器功能，就必须连续执行 DWDT 指令和 WRST 指令，将标志 WEF 清“0”，停止看门狗定时器功能。
- 在 RAM 备份时，标志 WDF1 和定时器 WDT 的值被初始化。
- 在同时使用看门狗定时器功能和 RAM 备份功能时，必须在转移到 RAM 备份模式前执行 WRST 指令，初始化标志 WDF1。另外，考虑到因 WRST 指令发生跳越的情况，必须在 WRST 指令后插入 NOP 指令。

20. 外部时钟

必须注意：使用外部时钟时的振荡频率最大值和使用陶瓷谐振时不同（参照推荐运行条件）。

另外，在使用外部时钟时，不能使用 RAM 备份模式（POF 指令）。

21. QzROM

- (1) 必须注意不要过电压。过电压可能会改写 QzROM 的内容，尤其是在接通电源时必须注意。
- (2) 虽然在组装工序前对空白出货产品进行了充分的 QzROM 写测试，但是在组装工序后并没有对用户 ROM 区进行写测试。因此，有可能出现 0.1%左右的编程故障。另外，编程环境也是造成编程故障的原因，所以必须在使用时充分注意电缆的接触和插座上的异物等。

注. 空白出货产品：在工厂出货时，不写 QzROM 的内容

22. 有关 ROM 代码保护的注意事项（编程后出货产品）

QzROM 编程后出货产品的 ROM 代码保护，由订货时提交的建立掩模文件时的 ROM 选项数据决定。

建立掩模文件时的 ROM 选项数据为有保护时设定“0016”，为无保护时设定“FF16”。

必须注意：在没有设定 ROM 选项数据或者设定了“0016”和“FF16”以外的数据时，不能接受该掩模文件。

使用注意事项

有关噪声的注意事项

有关噪声的注意事项和对策例子如下所示。本对策例子在理论上对噪声是有效的，但在实际使用时即使实施了本对策也需要进行充分的系统评价。

1. 缩短布线长度

(1) 复位引脚的布线

必须缩短连接复位引脚的布线，尤其是在复位引脚和 Vss 引脚之间连接电容时，必须尽量使用短的布线来连接。

●理由

为了对单片机进行正常的复位，输入到复位引脚的脉宽至少需要 1 个机器周期。如果小于 1 个机器周期的脉冲噪声输入到复位引脚，单片机内部就会在完全进入初始化状态前解除复位，从而导致程序失控。

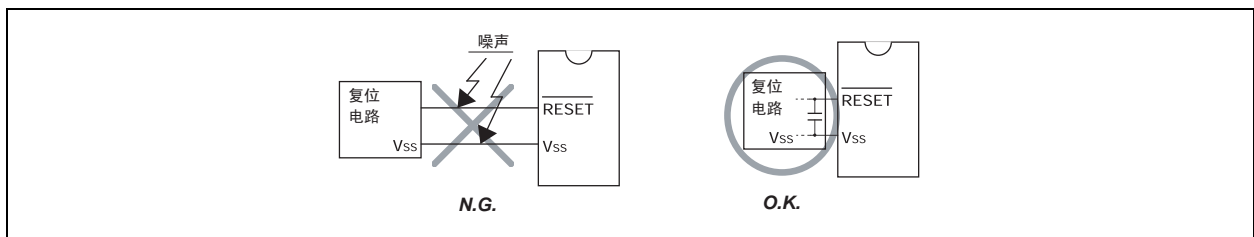


图 62 复位输入引脚的布线

(2) 时钟输入/输出引脚的布线

- 必须缩短连接时钟输入/输出引脚的布线。
- 必须用最短的布线连接谐振器的电容接地端引线和单片机的 Vss 引脚。
- 必须将用于振荡的 Vss 布线作为振荡电路专用布线，并和其他 Vss 布线分离。

●理由

如果有噪声侵入时钟输入/输出引脚，时钟的波形就会发生紊乱，导致误动作和失控。另外，如果因噪声而引起单片机 Vss 电平和谐振器 Vss 电平之间的电位差，就不能将正确的时钟输入到单片机。

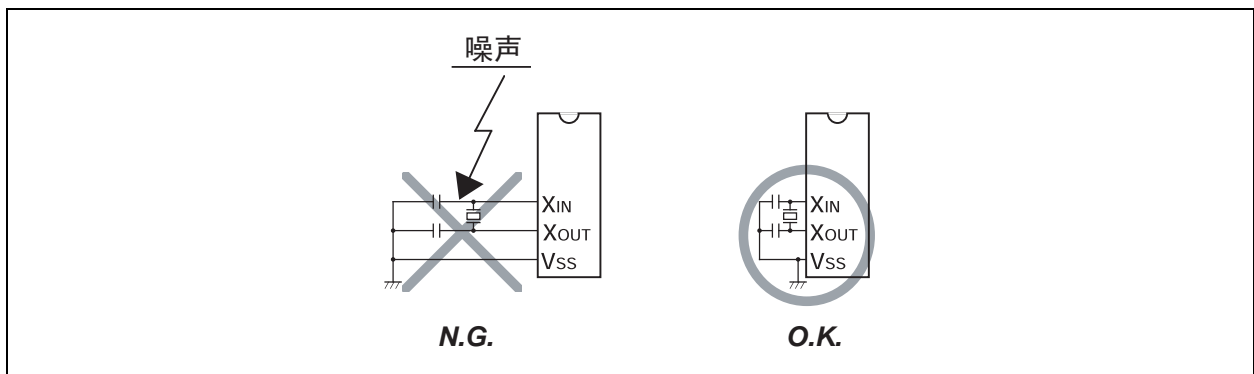


图 63 时钟输入/输出引脚的布线

(3) 端口 K 的布线

不能将端口 K 置为开路，必须尽量用粗的和短的布线将其连接到 VDD 引脚或者 Vss 引脚。

在用于键矩阵时，必须通过上拉电阻连接 VDD 引脚。此时，上拉电阻必须配置在端口 K 的附近，并且尽量用粗的和短的布线连接端口 K 和 VDD 引脚。

●理由

端口 K 可复用为内部 QzROM 的电源输入引脚 (Vpp 引脚)。

在进行 QzROM 编程时，为了使编程电流流过，降低了端口 K 的阻抗，所以噪声容易侵入。如果噪声从端口 K 侵入，就不能正常执行 QzROM 的指令码和数据的读操作，从而导致程序失控。

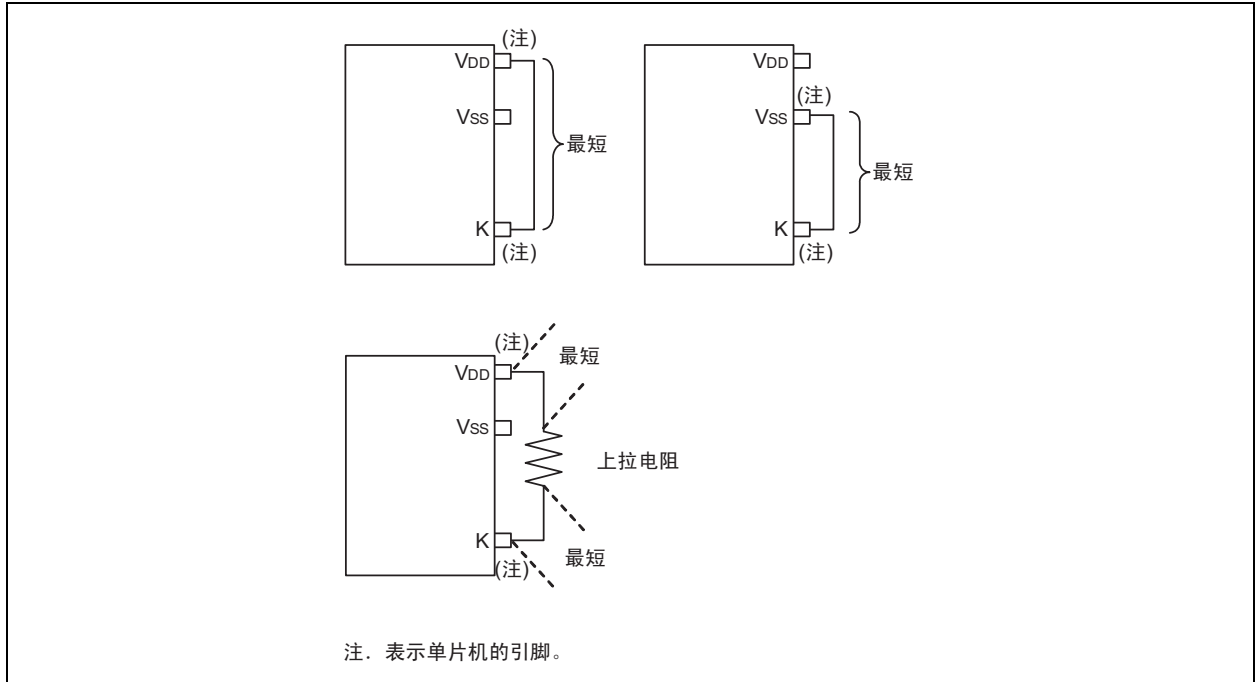


图 64 端口 K 的布线

2. 在 Vss-VDD 的布线之间插入旁路电容

必须用以下的方法在 Vss-VDD 的布线之间插入 0.1 μ F 左右的旁路电容。

- Vss 引脚—旁路电容间的布线长度和 VDD 引脚—旁路电容间的布线长度相等
- 尽量缩短 Vss 引脚—旁路电容之间的布线长度和 VDD 引脚—旁路电容之间的布线长度
- Vss 布线和 VDD 布线使用比其他信号线粗的布线
- 电源布线经由旁路电容连接到 Vss 引脚和 VDD 引脚

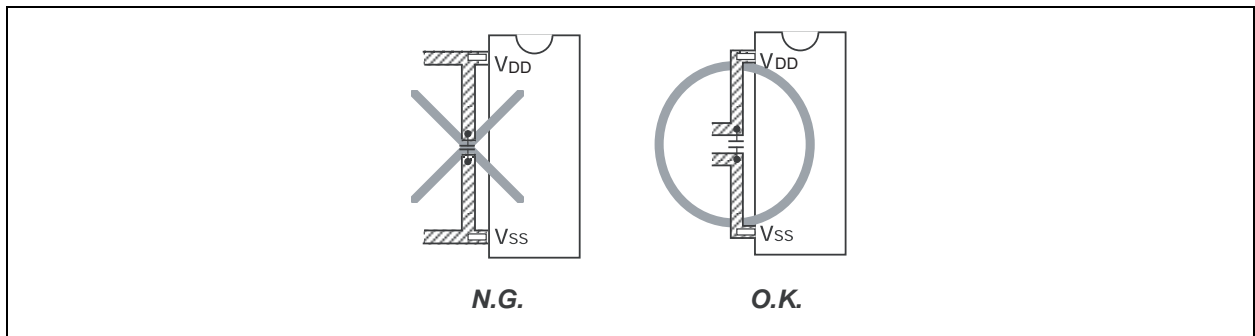


图 65 Vss-VDD 布线间的旁路电容

3. 对谐振器的考虑

必须考虑不能让其他信号影响谐振器，该谐振器用于产生单片机运行的基本时钟。

(1) 大电流信号线的回避

请尽量使大电流（超过单片机规定电流值范围的）信号线远离单片机（特别是谐振器）。

● 理由

在使用单片机的系统中有马达、LED 和热敏头等的控制信号线。当大电流在这些信号线中流过时，由于互感会产生噪声。

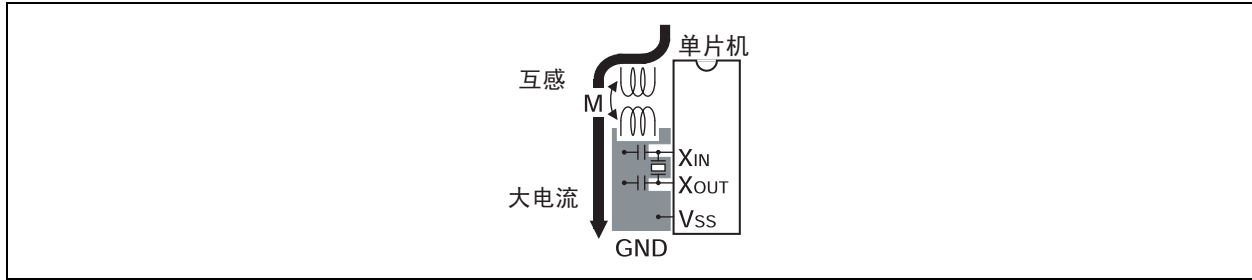


图 66 大电流信号线的布线

(2) 电平高速变化信号线的回避

请尽量使电平高速变化的信号线远离谐振器和谐振器的布线。

另外，电平高速变化的信号线不能和与时钟相关的信号线及其他易受噪声影响的信号线交叉。

●理由

电平高速变化的 CNTR 引脚等信号因上升或者下降时的电平变化而容易影响其他信号线。尤其是在和与时钟相关的信号线交叉时，时钟的波形会发生紊乱，导致误动作和失控。

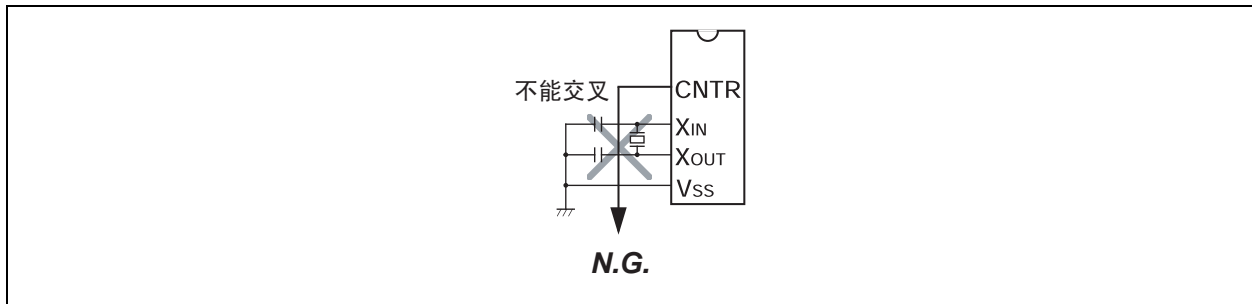


图 67 电平高速变化信号线的布线

(3) 用 Vss 布线进行保护

在双面电路板的情况下，必须在谐振器安装面的背面（焊接面）进行 Vss 布线，其位置与谐振器的位置相同。此 Vss 布线必须用最短的布线与单片机的 Vss 引脚连接，并且独立于其它 Vss 布线。

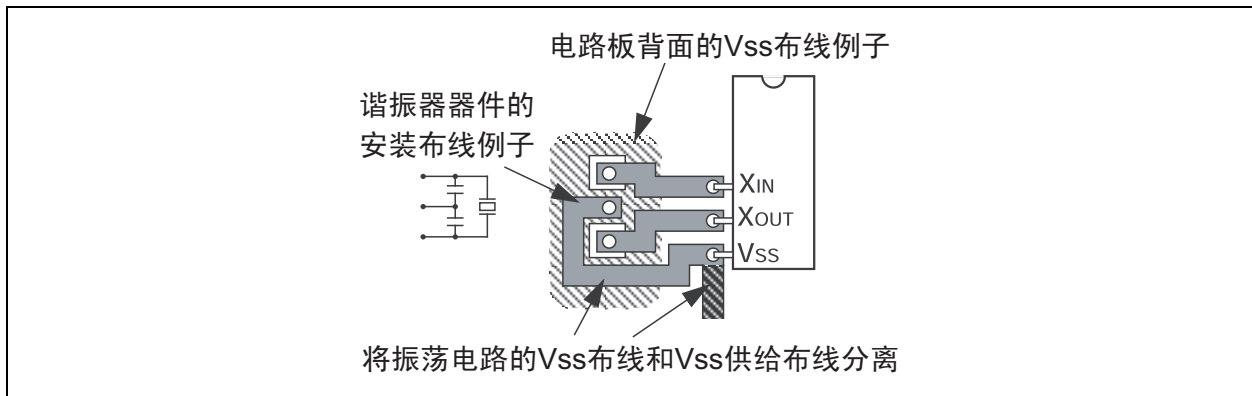


图 68 谐振器背面的 Vss 布线

4. 输入/输出端口的处理

输入/输出端口必须按下列要领用硬件和软件对应：

<硬件方面>

- 必须在输入/输出端口串联 1 个最小为 100Ω 的电阻。

<软件方面>

- 对于输入端口，必须用程序多次读取输入端口，确认电平的一致。
- 对于输出端口和输入/输出端口，可能因噪声而引起输出数据反转，所以必须以固定周期重写端口锁存器。
- 必须以固定周期重写上拉控制寄存器。

5. 通过软件实现看门狗定时器功能

在由噪声等引起单片机失控时，能用软件看门狗定时器检测失控，并且让其返回到正常运行。此方法与用硬件看门狗定时器检测失控的方法具有同等效果或更好的效果。软件看门狗定时器的例子如下所示。

在此例中，主程序监视中断处理程序的运行，中断处理程序监视主程序的运行，在检测到异常时让单片机恢复到正常运行状态。但是，此例的前提是在主程序的 1 个周期中进行多次中断处理。

<主程序>

- 将 1 字的 RAM 分配给软件看门狗定时器（SWDT），在主程序的每个周期给 SWDT 写一次初始值 N。初始值 N 满足以下条件：

$$N+1 \geq \text{在主程序的 1 个周期中进行的中断处理次数}$$

注：由于主程序的周期根据中断处理等发生变化，因此必须给初始值 N 设定充裕的值。

- 通过将 SWDT 的内容和设定初始值 N 后的中断处理次数进行比较，监视中断处理程序的运行。
- 如果在中断处理中 SWDT 的内容不发生变化，就将中断处理程序判断为异常运行，并进行向程序初始化转移等恢复处理。

<中断处理程序>

- 每进行 1 次中断处理，SWDT 的内容就减 1。
- 以几乎固定的周期（固定的中断处理次数）将 SWDT 的内容返回到初始值 N，来确认主程序的正常动作。
- SWDT 的内容不被设定成初始值 N 而继续减 1，如果 SWDT 的内容减少为 0 或 0 以下，就将主程序判断为异常运行，进行向程序初始化转移等恢复处理。

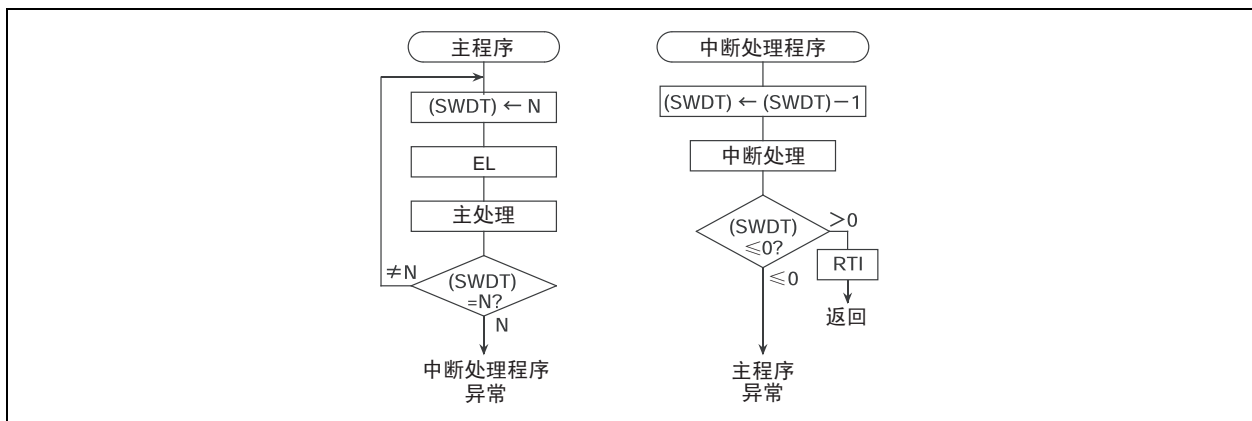


图 69 软件看门狗定时器

控制寄存器一览

中断控制寄存器 V1		复位时: 0000 ₂	RAM 备份时: 0000 ₂	R/W TAV1/TV1A
V13	定时器 2 中断允许位	0	禁止发生 (SNZT2 指令有效)	
		1	允许发生 (SNZT2 指令无效)	
V12	定时器 1 中断允许位	0	禁止发生 (SNZT1 指令有效)	
		1	允许发生 (SNZT1 指令无效)	
V11	外部 1 中断允许位	0	禁止发生 (SNZ1 指令有效)	
		1	允许发生 (SNZ1 指令无效)	
V10	外部 0 中断允许位	0	禁止发生 (SNZ0 指令有效)	
		1	允许发生 (SNZ0 指令无效)	

中断控制寄存器 V2		复位时: 0000 ₂	RAM 备份时: 0000 ₂	R/W TAV2/TV2A
V23	低电压检测电路中断允许位	0	禁止发生 (SNZVD 指令有效)	
		1	允许发生 (SNZVD 指令无效)	
V22	不使用	0	此位没有功能, 可进行读写操作	
		1		
V21	不使用	0	此位没有功能, 可进行读写操作	
		1		
V20	定时器 3 中断允许位	0	禁止发生 (SNZT3 指令有效)	
		1	允许发生 (SNZT3 指令无效)	

中断控制寄存器 I1		复位时: 0000 ₂	RAM 备份时: 保持状态	R/W TAI1/TI1A
I13	INT0 引脚输入控制位 (注 2)	0	禁止输入	
		1	允许输入	
I12	INT0 引脚中断有效波形/ 返回电平选择位 (注 2)	0	下降波形/“L”电平 (SNZI0 指令识别的“L”电平)	
		1	上升波形/“H”电平 (SNZI0 指令识别的“H”电平)	
I11	INT0 引脚边沿检测电路控制位	0	检测单边沿	
		1	检测双边沿	
I10	INT0 引脚定时器 1 控制允许位	0	禁止定时器 1 控制	
		1	允许定时器 1 控制	

中断控制寄存器 I2		复位时: 0000 ₂	RAM 备份时: 保持状态	R/W TAI2/TI2A
I23	INT1 引脚输入控制位 (注 3)	0	禁止输入	
		1	允许输入	
I22	INT1 引脚中断有效波形/ 返回电平选择位 (注 3)	0	下降波形/“L”电平 (SNZI1 指令识别的“L”电平)	
		1	上升波形/“H”电平 (SNZI1 指令识别的“H”电平)	
I21	INT1 引脚边沿检测电路控制位	0	检测单边沿	
		1	检测双边沿	
I20	不使用	0	此位没有功能, 可进行读写操作	
		1		

注 1. “R”表示可读, “W”表示可写。

注 2. 在更改这些位 (I12、I13) 的内容时, 可能会将外部中断请求标志 (EXF0) 置位。

注 3. 在更改这些位 (I22、I23) 的内容时, 可能会将外部中断请求标志 (EXF1) 置位。

定时器控制寄存器 PA		复位时: 00 ₂		RAM 备份时: 00 ₂	W TPAA
PA1	预分频器计数源选择位	0	指令时钟 (INSTCK)		
		1	指令时钟 (INSTCK) 的 4 分频信号		
PA0	预分频器控制位	0	停止 (保持状态)		
		1	运行		

定时器控制寄存器 W1		复位时: 0000 ₂		RAM 备份时: 保持状态	R/W TAW1/TW1A
W13	定时器 1 计数自动停止电路选择位 (注 2)	0	不选择定时器 1 计数自动停止电路		
		1	选择定时器 1 计数自动停止电路		
W12	定时器 1 控制位	0	停止 (保持状态)		
		1	运行		
W11	定时器 1 计数源选择位	W11	W10	计数源	
W10		0	0	PWM 信号 (PWMOUT)	
		0	1	预分频器输出 (ORCLK)	
		1	0	系统时钟 (STCK)	
		1	1	CNTR0 输入	

定时器控制寄存器 W2		复位时: 0000 ₂		RAM 备份时: 保持状态	R/W TAW2/TW2A
W23	CNTR0 引脚功能选择位	0	定时器 1 下溢信号的 2 分频输出		
		1	定时器 2 下溢信号的 2 分频输出		
W22	定时器 2 控制位	0	停止 (保持状态)		
		1	运行		
W21	定时器 2 计数源选择位	W21	W20	计数源	
W20		0	0	PWM 信号 (PWMOUT)	
		0	1	预分频器输出 (ORCLK)	
		1	0	系统时钟 (STCK)	
		1	1	定时器 1 下溢信号 (T1UDF)	

定时器控制寄存器 W3		复位时: 0000 ₂		RAM 备份时: 0000 ₂	R/W TAW3/TW3A
W33	CNTR1 引脚输出控制位	0	CNTR1 引脚输出无效		
		1	CNTR1 引脚输出有效		
W32	PWM 信号 高电平“H”期间扩展功能控制位	0	PWM 信号高电平“H”期间扩展功能无效		
		1	PWM 信号高电平“H”期间扩展功能有效		
W31	定时器 3 控制位	0	停止 (保持状态)		
		1	运行		
W30	定时器 3 计数源选择位	0	XIN 输入		
		1	预分频器输出 (ORCLK) 的 2 分频信号		

定时器控制寄存器 W5		复位时: 0000 ₂		RAM 备份时: 保持状态	R/W TAW5/TW5A
W53	定时器 1 计数开始同步电路选择位 (注 3)	0	不选择定时器 1 计数开始同步电路		
		1	选择定时器 1 计数开始同步电路		
W52	CNTR0 引脚输入计数边沿选择位	0	下降沿		
		1	上升沿		
W51	CNTR1 引脚 输出自动控制电路选择位	0	不选择 CNTR1 引脚输出自动控制电路		
		1	选择 CNTR1 引脚输出自动控制电路		
W50	D4/CNTR0 引脚功能选择位	0	D4 输入/输出或 CNTR0 输入		
		1	D4 输入或 CNTR0 输入/输出		

注 1. “R”表示可读, “W”表示可写。

注 2. 此功能只在 INTO 引脚定时器 1 控制允许 (I10=“1”) 以及选择定时器 1 的计数开始同步 (W53=“1”) 时有效。

注 3. 此功能只在 INTO 引脚定时器 1 控制允许 (I10=“1”) 时有效。

时钟控制寄存器 MR		复位时: 11112		RAM 备份时: 11112	R/W TAMR/TMRA
MR3	运行模式选择位	MR3	MR2	运行模式	
		0	0	through-mode (不分频)	
		0	1	2 分频模式	
		1	0	4 分频模式	
MR2		1	1	8 分频模式	
MR1	不使用	0	此位没有功能, 可进行读写操作		
		1			
MR0	不使用	0	此位没有功能, 可进行读写操作		
		1			

键唤醒控制寄存器 K0		复位时: 00002		RAM 备份时: 保持状态	R/W TAK0/TK0A
K03	端口 P03 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		
K02	端口 P02 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		
K01	端口 P01 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		
K00	端口 P00 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		

键唤醒控制寄存器 K1		复位时: 00002		RAM 备份时: 保持状态	R/W TAK1/TK1A
K13	端口 P13 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		
K12	端口 P12 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		
K11	端口 P11 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		
K10	端口 P10 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		

键唤醒控制寄存器 K2		复位时: 00002		RAM 备份时: 保持状态	R/W TAK2/TK2A
K23	不使用	0	此位没有功能, 可进行读写操作		
		1			
K22	端口 K 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		
K21	端口 P21 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		
K20	端口 P20 键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		

键唤醒控制寄存器 L1		复位时: 00002		RAM 备份时: 保持状态	R/W TAL1/TL1A
L13	INT1 引脚返回条件选择位	0	电平返回		
		1	边沿返回		
L12	INT1 引脚有效波形电平选择位	0	键唤醒无效		
		1	键唤醒有效		
L11	INT0 引脚返回条件选择位	0	电平返回		
		1	边沿返回		
L10	INT0 引脚键唤醒控制位	0	键唤醒无效		
		1	键唤醒有效		

注 1. “R” 表示可读, “W” 表示可写。

上拉控制寄存器 PU0		复位时: 0000 ₂		RAM 备份时: 保持状态	R/W TAPU0/TPU0A
PU0 ₃	端口 P0 ₃ 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		
PU0 ₂	端口 P0 ₂ 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		
PU0 ₁	端口 P0 ₁ 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		
PU0 ₀	端口 P0 ₀ 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		

上拉控制寄存器 PU1		复位时: 0000 ₂		RAM 备份时: 保持状态	R/W TAPU1/TPU1A
PU1 ₃	端口 P1 ₃ 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		
PU1 ₂	端口 P1 ₂ 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		
PU1 ₁	端口 P1 ₁ 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		
PU1 ₀	端口 P1 ₀ 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		

上拉控制寄存器 PU2		复位时: 0000 ₂		RAM 备份时: 保持状态	R/W TAPU2/TPU2A
PU2 ₃	不使用	0	此位没有功能, 可进行读写操作		
		1			
PU2 ₂	不使用	0	此位没有功能, 可进行读写操作		
		1			
PU2 ₁	端口 P2 ₁ 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		
PU2 ₀	端口 P2 ₀ 上拉晶体管控制位	0	上拉晶体管 OFF		
		1	上拉晶体管 ON		

端口输出形式控制寄存器 FR0		复位时: 0000 ₂		RAM 备份时: 保持状态	W TFR0A
FR0 ₃	不使用	0	此位没有功能, 可进行读写操作		
		1			
FR0 ₂	不使用	0	此位没有功能, 可进行读写操作		
		1			
FR0 ₁	端口 P3 ₁ 输出形式选择位	0	N 沟道漏极开路输出		
		1	CMOS 输出		
FR0 ₀	端口 P3 ₀ 输出形式选择位	0	N 沟道漏极开路输出		
		1	CMOS 输出		

端口输出形式控制寄存器 FR1		复位时: 0000 ₂		RAM 备份时: 保持状态	W TFR1A
FR1 ₃	端口 D ₃ 输出形式选择位	0	N 沟道漏极开路输出		
		1	CMOS 输出		
FR1 ₂	端口 D ₂ 输出形式选择位	0	N 沟道漏极开路输出		
		1	CMOS 输出		
FR1 ₁	端口 D ₁ 输出形式选择位	0	N 沟道漏极开路输出		
		1	CMOS 输出		
FR1 ₀	端口 D ₀ 输出形式选择位	0	N 沟道漏极开路输出		
		1	CMOS 输出		

注 1. “R”表示可读, “W”表示可写。

指令

通过以下的结构说明各指令：

1. 按指令功能索引
2. 英文字母顺序的机器指令一览表
3. 按功能分类的机器指令一览表（用双联页记载）
4. 指令码对应表

按指令功能索引、机器指令一览表使用以下符号：

指令符号一览表

符号	内 容	符号	内 容
A	寄存器 A (4 位)	T1F	定时器 1 中断请求标志
B	寄存器 B (4 位)	T2F	定时器 2 中断请求标志
DR	寄存器 DR (3 位)	T3F	定时器 3 中断请求标志
E	寄存器 E (8 位)	WDF1	看门狗定时器标志
V1	中断控制寄存器 V1 (4 位)	WEF	看门狗定时器允许标志
V2	中断控制寄存器 V2 (4 位)	INTE	中断允许标志
I1	中断控制寄存器 I1 (4 位)	EXF0	外部 0 中断请求标志
I2	中断控制寄存器 I2 (4 位)	EXF1	外部 1 中断请求标志
PA	定时器控制寄存器 PA (2 位)	VDF	低电压检测电路中断请求标志
W1	定时器控制寄存器 W1 (4 位)	P	掉电标志
W2	定时器控制寄存器 W2 (4 位)	D	端口 D (5 位)
W3	定时器控制寄存器 W3 (4 位)	P0	端口 P0 (4 位)
W5	定时器控制寄存器 W5 (4 位)	P1	端口 P1 (4 位)
MR	时钟控制寄存器 MR (4 位)	P2	端口 P2 (2 位)
K0	键唤醒控制寄存器 K0 (4 位)	P3	端口 P3 (2 位)
K1	键唤醒控制寄存器 K1 (4 位)	x	16 进制变量
K2	键唤醒控制寄存器 K2 (4 位)	y	16 进制变量
L1	键唤醒控制寄存器 L1 (4 位)	z	16 进制变量
PU0	上拉控制寄存器 PU0 (4 位)	p	16 进制变量
PU1	上拉控制寄存器 PU1 (4 位)	n	16 进制常量
PU2	上拉控制寄存器 PU2 (4 位)	i	16 进制常量
FR0	端口输出形式控制寄存器 FR0 (4 位)	j	16 进制常量
FR1	端口输出形式控制寄存器 FR1 (4 位)	A3 A2 A1 A0	16 进制变量 A 的 2 进制表示 (其它也同样)
X	寄存器 X (4 位)	←	数据的移动方向
Y	寄存器 Y (4 位)	()	寄存器、存储器等的内容
Z	寄存器 Z (2 位)	—	否定, 即使在执行指令后标志也不变
DP	数据指针 (10 位) (由寄存器 X、Y、Z 构成)	M(DP)	用数据指针指定的 RAM 地址
PC	程序计数器 (14 位)	a	表示地址 a6 a5 a4 a3 a2 a1 a0 的标志
PC _H	程序计数器的高 7 位	p、a	表示 P6 P5 P4 P3 P2 P1 P0 页的 a6 a5 a4 a3 a2 a1 地址的标志
PCL	程序计数器的低 7 位	C	16 进制 C+16 进制 x (其它也同样)
SK	堆栈寄存器 (14 位×8)	+	
SP	堆栈指针 (3 位)	×	
CY	进位标志	?	判断被表示在“?”前的状态
RPS	预分频器重加载寄存器 (8 位)	↔	寄存器和存储器之间的数据交换
R1L	定时器 1 重加载寄存器 (8 位)	AND	逻辑与
R2	定时器 2 重加载寄存器 (8 位)	OR	逻辑或
R3L	定时器 3 重加载寄存器 (8 位)		
R3H	定时器 3 重加载寄存器 (8 位)		
PS	预分频器		
T1	定时器 1		
T2	定时器 2		
T3	定时器 3		

[指令跳越的方法] 在发生跳越时, 只是下一条指令无效, 并不执行程序计数器的内容+2。因此, 即使不发生跳越, 周期数也不变。但是, 如果跳越 TABP p、RT、RTS 指令, 周期数就为“1”。

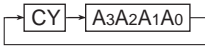
指令功能索引

分类	指令符号	功能	登载页
寄存器间的传送指令	TAB	$(A) \leftarrow (B)$	94、109
	TBA	$(B) \leftarrow (A)$	101、109
	TAY	$(A) \leftarrow (Y)$	101、109
	TYA	$(Y) \leftarrow (A)$	107、109
	TEAB	$(E7 \sim E4) \leftarrow (B)$ $(E3 \sim E0) \leftarrow (A)$	102、109
	TABE	$(B) \leftarrow (E7 \sim E4)$ $(A) \leftarrow (E3 \sim E0)$	95、109
	TDA	$(DR2 \sim DR0) \leftarrow (A2 \sim A0)$	101、109
	TAD	$(A2 \sim A0) \leftarrow (DR2 \sim DR0)$ $(A3) \leftarrow 0$	96、109
	TAZ	$(A1, A0) \leftarrow (Z1, Z0)$ $(A3, A2) \leftarrow 0$	101、109
	TAX	$(A) \leftarrow (X)$	100、109
	TASP	$(A2 \sim A0) \leftarrow (SP2 \sim SP0)$ $(A3) \leftarrow 0$	99、109
RAM地址指令	LXY x, y	$(X) \leftarrow x, x=0 \sim 15$ $(Y) \leftarrow y, y=0 \sim 15$	83、109
	LZ z	$(Z) \leftarrow z, z=0 \sim 3$	83、109
	INY	$(Y) \leftarrow (Y)+1$	82、109
	DEY	$(Y) \leftarrow (Y)-1$	80、109
RAM和寄存器间的传送指令	TAM j	$(A) \leftarrow (M(DP))$ $(X) \leftarrow (X) \text{EXOR}(j), j=0 \sim 15$	97、109
	XAM j	$(A) \leftrightarrow (M(DP))$ $(X) \leftarrow (X) \text{EXOR}(j), j=0 \sim 15$	108、109
	XAMD j	$(A) \leftrightarrow (M(DP))$ $(X) \leftarrow (X) \text{EXOR}(j), j=0 \sim 15$ $(Y) \leftarrow (Y)-1$	108、109
	XAMI j	$(A) \leftrightarrow (M(DP))$ $(X) \leftarrow (X) \text{EXOR}(j), j=0 \sim 15$ $(Y) \leftarrow (Y)+1$	108、109
	TMA j	$(M(DP)) \leftarrow (A)$ $(X) \leftarrow (X) \text{EXOR}(j), j=0 \sim 15$	104、109

M34571G4: p=0~31

M34571G6: p=0~47

M34571GD: p=0~127

分类	指令符号	功能	登载页
运算指令	LA n	$(A) \leftarrow n, n=0 \sim 15$	82、108
	TABP p	$(SP) \leftarrow (SP)+1$ $(SK(SP)) \leftarrow (PC)$ $(PCH) \leftarrow p$ $(PCL) \leftarrow (DR_2 \sim DR_0, A_3 \sim A_0)$ 在(UPTF)=1 时, $(DR_2) \leftarrow 0$ $(DR_1, DR_0) \leftarrow (ROM(PC))_{9,8}$ $(B) \leftarrow (ROM(PC))_{7 \sim 4}$ $(A) \leftarrow (ROM(PC))_{3 \sim 0}$ $(PC) \leftarrow (SK(SP))$ $(SP) \leftarrow (SP)-1$	95、108
	AM	$(A) \leftarrow (A)+(M(DP))$	77、111
	AMC	$(A) \leftarrow (A)+(M(DP))+(CY)$ $(CY) \leftarrow$ 进位	77、111
	A n	$(A) \leftarrow (A)+n, n=0 \sim 15$	77、111
	AND	$(A) \leftarrow (A) \text{ AND } (M(DP))$	77、111
	OR	$(A) \leftarrow (A) \text{ OR } (M(DP))$	84、111
	SC	$(CY) \leftarrow 1$	88、111
	RC	$(CY) \leftarrow 0$	86、111
	SZC	$(CY)=0?$	92、111
	CMA	$(A) \leftarrow (\bar{A})$	79、111
RAR		85、111	
位操作指令	SB j	$(M_j(DP)) \leftarrow 1, j=0 \sim 3$	87、111
	RB j	$(M_j(DP)) \leftarrow 0, j=0 \sim 3$	85、111
	SZB j	$(M_j(DP))=0?, j=0 \sim 3$	92、111
比较指令	SEAM	$(A)=(M(DP))?$	89、113
	SEA n	$(A)=n, n=0 \sim 15$	89、113
转移指令	B a	$(PCL) \leftarrow a_6 \sim a_0$	78、113
	BLp, a	$(PCH) \leftarrow p$ $(PCL) \leftarrow a_6 \sim a_0$	78、113
	BLA p	$(PCH) \leftarrow p$ $(PCL) \leftarrow (DR_2 \sim DR_0, A_3 \sim A_0)$	78、113

按指令功能索引 (续)

分类	指令符号	功能	登载页
子程序调用指令	BM a	(SP) ← (SP)+1 (SK(SP)) ← (PC) (PCH) ← 2 (PCL) ← a6~a0	78、113
	BMLp, a	(SP) ← (SP)+1 (SK(SP)) ← (PC) (PCH) ← p (PCL) ← a6~a0	79、113
	BMLA p	(SP) ← (SP)+1 (SK(SP)) ← (PC) (PCH) ← p (PCL) ← (DR2~DR0, A3~A0)	79、113
返回指令	RTI	(PC) ← (SK(SP)) (SP) ← (SP)-1	87、113
	RT	(PC) ← (SK(SP)) (SP) ← (SP)-1	86、113
	RTS	(PC) ← (SK(SP)) (SP) ← (SP)-1	87、113
中断控制指令	DI	(INTE) ← 0	80、115
	EI	(INTE) ← 1	80、115
	SNZ0	V10=0: (EXF0)=1? (EXF0) ← 0 V10=1: SNZ0: NOP	89、115
	SNZI0	I12=0: (INT0)=“L” ? I12=1: (INT0)=“H” ?	90、115
	SNZ1	V11=0: (EXF1)=1? (EXF1) ← 0 V11=1: SNZ1: NOP	89、115
	SNZI1	I22=0: (INT1)=“L” ? I22=1: (INT1)=“H” ?	90、115
	TAV1	(A) ← (V1)	99、115
	TV1A	(V1) ← (A)	106、115
	TAV2	(A) ← (V2)	99、115
	TV2A	(V2) ← (A)	106、115
	TAI1	(A) ← (I1)	96、115
	TI1A	(I1) ← (A)	102、115
	TAI2	(A) ← (I2)	96、115
TI2A	(I2) ← (A)	103、115	

分类	指令符号	功能	登载页
定时器操作指令	TPAA	(PA0) ← (A0)	104、115
	TAW1	(A) ← (W1)	99、115
	TW1A	(W1) ← (A)	106、115
	TAW2	(A) ← (W2)	100、115
	TW2A	(W2) ← (A)	107、115
	TAW3	(A) ← (W3)	100、115
	TW3A	(W3) ← (A)	107、115
	TAW5	(A) ← (W5)	100、115
	TW5A	(W5) ← (A)	107、115
	TABPS	(B) ← (TPS7~TPS4) (A) ← (TPS3~TPS0)	95、117
	TPSAB	(RPS7~RPS4) ← (B) (TPS7~TPS4) ← (B) (RPS3~RPS0) ← (A) (TPS3~TPS0) ← (A)	105、117
	TAB1	(B) ← (T17~T14) (A) ← (T13~T10)	94、117
	T1AB	(R17~R14) ← (B) (T17~T14) ← (B) (R13~R10) ← (A) (T13~T10) ← (A)	93、117
	TR1AB	(R17~R14) ← (B) (R13~R10) ← (A)	106、117
	TAB2	(B) ← (T27~T24) (A) ← (T23~T20)	94、117
	T2AB	(R27~R24) ← (B) (T27~T24) ← (B) (R23~R20) ← (A) (T23~T20) ← (A)	93、117
	TAB3	(B) ← (T37~T34) (A) ← (T33~T30)	95、117
	T3AB	(R3L7~R3L4) ← (B) (T37~T34) ← (B) (R3L3~R3L0) ← (A) (T33~T30) ← (A)	93、117
	T3R3L	(T37~T30) ← (R3L7~R3L0)	94、117
	T3HAB	(R3H7~R3H4) ← (B) (R3H3~R3H0) ← (A)	93、117

M34571G4: p=0~31

M34571G6: p=0~47

M34571GD: p=0~127

按指令功能索引 (续)

分类	指令符号	功能	登载页
定时器操作指令	SNZT1	V12=0: (T1F)=1? (T1F) ← 0 V12=1: SNZT1=NOP	90、117
	SNZT2	V13=0: (T2F)=1? (T2F) ← 0 V13=1: SNZT2=NOP	91、117
	SNZT3	V20=0: (T3F)=1? (T3F) ← 0 V20=1: SNZT3=NOP	91、117
输入 / 输出指令	IAP0	(A) ← (P0)	81、119
	OP0A	(P0) ← (A)	83、119
	IAP1	(A) ← (P1)	81、119
	OP1A	(P1) ← (A)	84、119
	IAP2	(A1、A0) ← (P21、P20) (A3、A2) ← 0	81、119
	OP2A	(P21、P20) ← (A1、A0)	84、119
	IAP3	(A1、A0) ← (P31、P30) (A3、A2) ← 0	82、119
	OP3A	(P31、P30) ← (A1、A0)	84、119
	CLD	(D) ← 1	79、119
	RD	(D(Y)) ← 0、(Y)=0~4	86、119
	SD	(D(Y)) ← 1、(Y)=0~4	88、119
	SZD	(D(Y))=0?、(Y)=0~4	92、119
	RCP	(C) ← (0)	86、119
	SCP	(C) ← (1)	88、119
	IAK	(A0) ← (K) (A3~A1) ← 0	81、119
	TFR0A	(FR0) ← (A)	102、119
	TFR1A	(FR1) ← (A)	102、119
	TAPU0	(A) ← (PU0)	98、119
	TPU0A	(PU0) ← (A)	105、119
	TAPU1	(A) ← (PU1)	98、119
TPU1A	(PU1) ← (A)	105、119	
TAPU2	(A) ← (PU2)	98、119	
TPU2A	(PU2) ← (A)	105、119	

分类	指令符号	功能	登载页
输入 / 输出指令	TAK0	(A) ← (K0)	96、121
	TK0A	(K0) ← (A)	103、121
	TAK1	(A) ← (K1)	97、121
	TK1A	(K1) ← (A)	103、121
	TAK2	(A) ← (K2)	97、121
	TK2A	(K2) ← (A)	103、121
	TAL1	(A) ← (L1)	97、121
	TL1A	(L1) ← (A)	104、121
其它	TAMR	(A) ← (MR)	98、121
	TMRA	(MR) ← (A)	104、121
	NOP	(PC) ← (PC)+1	83、121
	POF	转移到RAM备份模式	85、121
	EPOF	POF指令有效	81、121
	SNZP	(P)=1?	90、121
	SNZVD	V23=0: (VDF)=1? V23=0: NOP	91、121
	WRST	(WDF1)=1? (WDF1) ← 0	108、121
	DWDT	允许看门狗定时器功能停止	80、121
	SRST	系统复位	91、121
	RUPT	(UPTF) ← 0	87、121
	SUPT	(UPTF) ← 1	92、121
	RBK(注2)	TABP p指令执行时: p6 ← 0	85、121
	SBK(注2)	TABP p指令执行时: p6 ← 1	88、121

注1. M34571G4/G6中, RBK、SBK指令不能使用。

[英文字母顺序] 机器指令一览表

An (Add n and accumulator)				
机器码:	D ₉	D ₀	字数	周期数
	0 0 0 1 1 0 n n n n	2 0 6 n	1	1
				标志CY
				—
				跳越条件
				上溢=0
功能: (A) ← (A) + n n=0~15			分类: 运算指令	
			详细说明: 给寄存器A的内容加上立即字段的值n。其结果保存到寄存器A。进位标志 (CY) 的内容不变。 如果加法运算的结果上溢, 就继续执行下一条指令。否则就跳越下一条指令。	
AM (Add accumulator and Memory)				
机器码:	D ₉	D ₀	字数	周期数
	0 0 0 0 0 0 1 0 1 0	2 0 0 A	1	1
				标志CY
				—
				跳越条件
				—
功能: (A) ← (A) + (M(DP))			分类: 运算指令	
			详细说明: 给寄存器A的内容加上M(DP)的内容。其结果保存到寄存器A。进位标志 (CY) 的内容不变。	
AMC (Add accumulator, Memory and Carry)				
机器码:	D ₉	D ₀	字数	周期数
	0 0 0 0 0 0 1 0 1 1	2 0 0 B	1	1
				标志CY
				0/1
				跳越条件
				—
功能: (A) ← (A) + (M(DP)) + (CY) (CY) ← 进位			分类: 运算指令	
			详细说明: 给寄存器A的内容加上M(DP)的内容和进位标志 (CY) 的内容。其结果保存到寄存器A和标志CY。	
AND (logical AND between accumulator and memory)				
机器码:	D ₉	D ₀	字数	周期数
	0 0 0 0 0 1 1 0 0 0	2 0 1 8	1	1
				标志CY
				—
				跳越条件
				—
功能: (A) ← (A) AND (M(DP))			分类: 运算指令	
			详细说明: 计算寄存器A的内容和M(DP)的内容的逻辑与。其结果保存到寄存器A。	

[英文字母顺序] 机器指令一览表 (续)

B a (Branch to address a)				
机器码: D ₉ D₀ <div style="display: flex; justify-content: space-between; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">a₆</div> <div style="border: 1px solid black; padding: 2px;">a₅</div> <div style="border: 1px solid black; padding: 2px;">a₄</div> <div style="border: 1px solid black; padding: 2px;">a₃</div> <div style="border: 1px solid black; padding: 2px;">a₂</div> <div style="border: 1px solid black; padding: 2px;">a₁</div> <div style="border: 1px solid black; padding: 2px;">a₀</div> <div style="border: 1px solid black; padding: 2px;">2</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">8 +a</div> <div style="border: 1px solid black; padding: 2px;">a</div> <div style="border: 1px solid black; padding: 2px;">16</div> </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (PCL) ← a ₆ ~a ₀	分类: 转移指令 详细说明: 页内转移, 转移到同一页的地址a。 注意点: 必须在该指令的所在页内指定转移目标。			
BL p,a (Branch Long to address a in page p)				
机器码: D ₉ D₀ <div style="display: flex; justify-content: space-between; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">p₄</div> <div style="border: 1px solid black; padding: 2px;">p₃</div> <div style="border: 1px solid black; padding: 2px;">p₂</div> <div style="border: 1px solid black; padding: 2px;">p₁</div> <div style="border: 1px solid black; padding: 2px;">p₀</div> <div style="border: 1px solid black; padding: 2px;">2</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">E +p</div> <div style="border: 1px solid black; padding: 2px;">p</div> <div style="border: 1px solid black; padding: 2px;">16</div> </div> <div style="display: flex; justify-content: space-between; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">p₆</div> <div style="border: 1px solid black; padding: 2px;">p₅</div> <div style="border: 1px solid black; padding: 2px;">a₆</div> <div style="border: 1px solid black; padding: 2px;">a₅</div> <div style="border: 1px solid black; padding: 2px;">a₄</div> <div style="border: 1px solid black; padding: 2px;">a₃</div> <div style="border: 1px solid black; padding: 2px;">a₂</div> <div style="border: 1px solid black; padding: 2px;">a₁</div> <div style="border: 1px solid black; padding: 2px;">a₀</div> <div style="border: 1px solid black; padding: 2px;">2</div> <div style="border: 1px solid black; padding: 2px;">2</div> <div style="border: 1px solid black; padding: 2px;">a</div> <div style="border: 1px solid black; padding: 2px;">a</div> <div style="border: 1px solid black; padding: 2px;">16</div> </div>	字数	周期数	标志CY	跳越条件
	2	2	—	—
功能: (PCH) ← p (PCL) ← a ₆ ~a ₀	分类: 转移指令 详细说明: 页外转移, 转移到页p的地址a。 注意点: M34571G4时、p=0~31 M34571G6时、p=0~47 M34571GD时、p=0~127			
BLA p (Branch Long to address (D)+(A) in page p)				
机器码: D ₉ D₀ <div style="display: flex; justify-content: space-between; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">2</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">16</div> </div> <div style="display: flex; justify-content: space-between; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">p₆</div> <div style="border: 1px solid black; padding: 2px;">p₅</div> <div style="border: 1px solid black; padding: 2px;">p₄</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">p₃</div> <div style="border: 1px solid black; padding: 2px;">p₂</div> <div style="border: 1px solid black; padding: 2px;">p₁</div> <div style="border: 1px solid black; padding: 2px;">p₀</div> <div style="border: 1px solid black; padding: 2px;">2</div> <div style="border: 1px solid black; padding: 2px;">2</div> <div style="border: 1px solid black; padding: 2px;">p</div> <div style="border: 1px solid black; padding: 2px;">p</div> <div style="border: 1px solid black; padding: 2px;">16</div> </div>	字数	周期数	标志CY	跳越条件
	2	2	—	—
功能: (PCH) ← p (PCL) ← (DR ₂ ~DR ₀ , A ₃ ~A ₀)	分类: 转移指令 详细说明: 页外转移, 转移到由页p的寄存器D和寄存器A的内容表示的地址(DR ₂ DR ₁ DR ₀ A ₃ A ₂ A ₁ A ₀) ₂ 。 注意点: M34571G4时、p=0~31 M34571G6时、p=0~47 M34571GD时、p=0~127			
BM a (Branch and Mark to address a in page 2)				
机器码: D ₉ D₀ <div style="display: flex; justify-content: space-between; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">a₆</div> <div style="border: 1px solid black; padding: 2px;">a₅</div> <div style="border: 1px solid black; padding: 2px;">a₄</div> <div style="border: 1px solid black; padding: 2px;">a₃</div> <div style="border: 1px solid black; padding: 2px;">a₂</div> <div style="border: 1px solid black; padding: 2px;">a₁</div> <div style="border: 1px solid black; padding: 2px;">a₀</div> <div style="border: 1px solid black; padding: 2px;">2</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">a</div> <div style="border: 1px solid black; padding: 2px;">a</div> <div style="border: 1px solid black; padding: 2px;">16</div> </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (SP) ← (SP) + 1 (SK(SP)) ← (PC) (PCH) ← 2 (PCL) ← a ₆ ~a ₀	分类: 子程序调用指令 详细说明: 页2的子程序调用, 调用页2的地址a的子程序。 注意点: 即使是 从页2跨页写的子程序, 只要其起始部分在页2内, 就能调用。子程序嵌套最大8层, 必须注意不能超出堆栈。			

[英文字母顺序] 机器指令一览表 (续)

BML p,a (Branch and Mark Long to address a in page p)				
机器码: D ₉ D₀ <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">p₄</div> <div style="border: 1px solid black; padding: 2px;">p₃</div> <div style="border: 1px solid black; padding: 2px;">p₂</div> <div style="border: 1px solid black; padding: 2px;">p₁</div> <div style="border: 1px solid black; padding: 2px;">p₀</div> <div style="border: 1px solid black; padding: 2px;">2</div> </div> <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 5px;"> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">C_{4p}</div> <div style="border: 1px solid black; padding: 2px;">p</div> <div style="border: 1px solid black; padding: 2px;">16</div> </div>	字数	周期数	标志CY	跳越条件
	2	2	—	—
<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">p₆</div> <div style="border: 1px solid black; padding: 2px;">p₅</div> <div style="border: 1px solid black; padding: 2px;">a₆</div> <div style="border: 1px solid black; padding: 2px;">a₅</div> <div style="border: 1px solid black; padding: 2px;">a₄</div> <div style="border: 1px solid black; padding: 2px;">a₃</div> <div style="border: 1px solid black; padding: 2px;">a₂</div> <div style="border: 1px solid black; padding: 2px;">a₁</div> <div style="border: 1px solid black; padding: 2px;">a₀</div> <div style="border: 1px solid black; padding: 2px;">2</div> </div> <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 5px;"> <div style="border: 1px solid black; padding: 2px;">2</div> <div style="border: 1px solid black; padding: 2px;">a</div> <div style="border: 1px solid black; padding: 2px;">a</div> <div style="border: 1px solid black; padding: 2px;">16</div> </div>	分类: 子程序调用指令 详细说明: 子程序调用, 调用页p的地址a的子程序。 注意点: M34571G4时、p=0~31 M34571G6时、p=0~47 M34571GD时、p=0~127 子程序嵌套最大8层, 必须注意不能超出堆栈。			
功能: (SP) ← (SP) + 1 (SK(SP)) ← (PC) (PCH) ← p (PCL) ← a ₆ ~a ₀				
BMLA p (Branch and Mark Long to address (D)+(A) in page p)				
机器码: D ₉ D₀ <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">2</div> </div> <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 5px;"> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">3</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">16</div> </div>	字数	周期数	标志CY	跳越条件
	2	2	—	—
<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">p₆</div> <div style="border: 1px solid black; padding: 2px;">p₅</div> <div style="border: 1px solid black; padding: 2px;">p₄</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">p₃</div> <div style="border: 1px solid black; padding: 2px;">p₂</div> <div style="border: 1px solid black; padding: 2px;">p₁</div> <div style="border: 1px solid black; padding: 2px;">p₀</div> <div style="border: 1px solid black; padding: 2px;">2</div> </div> <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 5px;"> <div style="border: 1px solid black; padding: 2px;">2</div> <div style="border: 1px solid black; padding: 2px;">p</div> <div style="border: 1px solid black; padding: 2px;">p</div> <div style="border: 1px solid black; padding: 2px;">16</div> </div>	分类: 子程序调用指令 详细说明: 子程序调用, 调用由页p的寄存器D和寄存器A的内容指定的地址(DR ₂ DR ₁ DR ₀ A ₃ A ₂ A ₁ A ₀) ₂ 的子程序。 注意点: M34571G4时、p=0~31 M34571G6时、p=0~47 M34571GD时、p=0~127 子程序嵌套最大8层, 必须注意不能超出堆栈。			
功能: (SP) ← (SP) + 1 (SK(SP)) ← (PC) (PCH) ← p (PCL) ← (DR ₂ ~DR ₀ 、A ₃ ~A ₀)				
CLD (Clear port D)				
机器码: D ₉ D₀ <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">2</div> </div> <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 5px;"> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">16</div> </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (D) ← 1	分类: 输入/输出指令 详细说明: 将端口D全部置位(1)。			
CMA (CoMplement of Accumulator)				
机器码: D ₉ D₀ <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">2</div> </div> <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 5px;"> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">C</div> <div style="border: 1px solid black; padding: 2px;">16</div> </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (A) ← (\bar{A})	分类: 运算指令 详细说明: 将寄存器A的内容的1的补码保存到寄存器A。			

[英文字母顺序] 机器指令一览表 (续)

DEY (DEcrement register Y)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
0 0 0 0 0 1 0 1 1 1	0 1 7	1	1	—	(Y)=15
功能: (Y) ← (Y) - 1		分类: RAM地址指令			
		详细说明: 将寄存器Y的内容减1。作为其结果, 如果寄存器Y的内容为“15”, 就跳越下一条指令。否则就继续执行下一条指令。			
DI (Disable Interrupt)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
0 0 0 0 0 0 0 1 0 0	0 0 4	1	1	—	—
功能: (INTE) ← 0		分类: 中断控制指令			
		详细说明: 将中断允许标志 (INTE) 清零, 并且设定成中断发生禁止状态。 注意点: 通过DI指令设定的中断禁止在从DI指令执行开始经过1个机器周期后有效。			
DWDT (Disable WatchDog Timer)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
1 0 1 0 0 1 1 1 0 0	2 9 C	1	1	—	—
功能: 看门狗定时器功能停止允许		分类: 其它			
		详细说明: 在执行DWDT指令后, 能立即通过WRST指令停止看门狗定时器功能。			
EI (Enable Interrupt)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
0 0 0 0 0 0 0 1 0 1	0 0 5	1	1	—	—
功能: (INTE) ← 1		分类: 中断控制指令			
		详细说明: 将中断允许标志 (INTE) 置位 (1), 并且设定成中断发生允许状态。 注意点: 通过EI指令设定的中断允许在从EI指令执行开始经过1个机器周期后有效。			

[英文字母顺序] 机器指令一览表 (续)

EPOF (Enable POF instruction)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div> 2 <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">5</div><div style="border: 1px solid black; padding: 2px;">B</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: POF指令有效	分类: 其它 详细说明: 在执行EPOF指令后, 紧接的POF指令变为有效。			
IAK (Input Accumulator from port K)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div> 2 <div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">6</div><div style="border: 1px solid black; padding: 2px;">F</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (A ₀) ← (K) (A ₃ ~A ₁) ← 0	分类: 输入/输出指令 详细说明: 将端口K的输入传送到寄存器A的最低位 (A ₀)。寄存器A的高3位 (A ₃ ~A ₁) 的内容为“0”。			
IAP0 (Input Accumulator from port P0)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div> 2 <div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">6</div><div style="border: 1px solid black; padding: 2px;">0</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (A) ← (P0)	分类: 输入/输出指令 详细说明: 将端口P0的输入传送到寄存器A。			
IAP1 (Input Accumulator from port P1)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div> 2 <div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">6</div><div style="border: 1px solid black; padding: 2px;">1</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (A) ← (P1)	分类: 输入/输出指令 详细说明: 将端口P1的输入传送到寄存器A。			

[英文字母顺序] 机器指令一览表 (续)

IAP2 (Input Accumulator from port P2)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
1 0 0 1 1 0 0 0 1 0	2 2 6 2	1	1	—	—
功能: (A ₁ 、A ₀) ← (P ₂₁ 、P ₂₀) (A ₃ 、A ₂) ← 0		分类: 输入/输出指令 详细说明: 将端口P2的输入传送到寄存器A的低2位 (A ₁ 、A ₀)。寄存器A的高2位 (A ₃ 、A ₂) 的内容为“0”。			
IAP3 (Input Accumulator from port P3)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
1 0 0 1 1 0 0 0 1 1	2 2 6 3	1	1	—	—
功能: (A ₁ 、A ₀) ← (P ₃₁ 、P ₃₀) (A ₃ 、A ₂) ← 0		分类: 输入/输出指令 详细说明: 将端口P3的输入传送到寄存器A的低2位 (A ₁ 、A ₀)。寄存器A的高2位 (A ₃ 、A ₂) 的内容为“0”。			
INY (INcrement register Y)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
0 0 0 0 0 1 0 0 1 1	2 0 1 3	1	1	—	(Y)=0
功能: (Y) ← (Y) + 1		分类: RAM地址指令 详细说明: 将寄存器Y的内容加1。作为其结果, 如果寄存器Y的内容为“0”, 就跳越下一条指令。否则就继续执行下一条指令。			
LA n (Load n in Accumulator)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
0 0 0 1 1 1 n n n n	2 0 7 n	1	1	—	连续描述
功能: (A) ← n n=0~15		分类: 运算指令 详细说明: 将立即字段的值n装入寄存器A。在连续描述和执行LA指令时, 除了最初执行的LA指令外, 跳越后面连续描述的LA指令。			

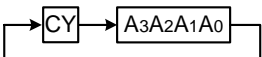
[英文字母顺序] 机器指令一览表 (续)

LXY x,y (Load register X and Y with x and y)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">x₃</div> <div style="border: 1px solid black; padding: 2px;">x₂</div> <div style="border: 1px solid black; padding: 2px;">x₁</div> <div style="border: 1px solid black; padding: 2px;">x₀</div> <div style="border: 1px solid black; padding: 2px;">y₃</div> <div style="border: 1px solid black; padding: 2px;">y₂</div> <div style="border: 1px solid black; padding: 2px;">y₁</div> <div style="border: 1px solid black; padding: 2px;">y₀</div> <div style="border: 1px solid black; padding: 2px;">2</div> <div style="border: 1px solid black; padding: 2px;">3</div> <div style="border: 1px solid black; padding: 2px;">x</div> <div style="border: 1px solid black; padding: 2px;">y</div> <div style="border: 1px solid black; padding: 2px;">16</div> </div>	字数	周期数	标志CY	跳越条件
	1	1	—	连续描述
功能: (X) ← x x=0~15 (Y) ← y y=0~15	分类: RAM地址指令 详细说明: 将立即字段的值x装入寄存器X、立即字段的值y装入寄存器Y。在连续描述和执行LXY指令时,除了最初执行的LXY指令外,跳越后面连续描述的LXY指令。			
LZ z (Load register Z with z)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">z₁</div> <div style="border: 1px solid black; padding: 2px;">z₀</div> <div style="border: 1px solid black; padding: 2px;">2</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">4</div> <div style="border: 1px solid black; padding: 2px;">8</div> <div style="border: 1px solid black; padding: 2px;">z</div> <div style="border: 1px solid black; padding: 2px;">16</div> </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (Z) ← z z=0~3	分类: RAM地址指令 详细说明: 将立即字段的值z装入寄存器Z。			
NOP (No Operation)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">2</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">16</div> </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (PC) ← (PC) + 1	分类: 其它 详细说明: 空操作,程序计数器的值加1,其它不变化。			
OP0A (Output port P0 from Accumulator)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">1</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">2</div> <div style="border: 1px solid black; padding: 2px;">2</div> <div style="border: 1px solid black; padding: 2px;">0</div> <div style="border: 1px solid black; padding: 2px;">16</div> </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (P0) ← (A)	分类: 输入/输出指令 详细说明: 将寄存器A的内容输出到端口P0。			

[英文字母顺序] 机器指令一览表 (续)

OP1A (Output port P1 from Accumulator)				
机器码: D9 D0 <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div> 2 <div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">1</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (P1) ← (A)	分类: 输入/输出指令 详细说明: 将寄存器A的内容输出到端口P1。			
OP2A (Output port P2 from Accumulator)				
机器码: D9 D0 <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div> 2 <div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">2</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (P21、P20) ← (A1、A0)	分类: 输入/输出指令 详细说明: 将寄存器A的低2位 (A1、A0) 的内容输出到端口P2。			
OP3A (Output port P3 from Accumulator)				
机器码: D9 D0 <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div> 2 <div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">3</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (P31、P30) ← (A1、A0)	分类: 输入/输出指令 详细说明: 将寄存器A低2位 (A1、A0) 的内容输出到端口P3。			
OR (logical OR between accumulator and memory)				
机器码: D9 D0 <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div> 2 <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">9</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (A) ← (A) OR (M(DP))	分类: 运算指令 详细说明: 计算寄存器A内容和M(DP)内容的逻辑或。其结果保存到寄存器A。			

[英文字母顺序] 机器指令一览表 (续)

POF (Power OFF)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
0 0 0 0 0 0 0 0 1 0	2	0 0 2	16	—	—
功能: 转移到RAM备份模式		分类: 其它			
		详细说明: 如果EPOF指令执行后接着执行POF指令, 本产品就进入RAM备份模式。			
		注意点: 如果在执行此指令前没有执行EPOF指令, 此指令就等价于NOP指令。			
RAR (Rotate Accumulator Right)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
0 0 0 0 0 1 1 1 0 1	2	0 1 D	16	0/1	—
功能: 		分类: 运算指令			
		详细说明: 包含进位标志 (CY), 将寄存器A的内容向右循环1位。			
RB j (Reset Bit)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
0 0 0 1 0 0 1 1 j j	2	0 4	C _j	16	—
功能: (M _j (DP)) ← 0 j=0~3		分类: 位操作指令			
		详细说明: 将M(DP)的第j位 (由立即字段的值j指定的位) 的内容清零。			
RBK (Reset Bank flag)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
0 0 0 1 0 0 1 1 0 0	2	0 4 0	16	—	—
功能: 执行TABP p指令时: p ₆ ← 0		分类: 其他			
		详细说明: 把执行TABP p指令时参照的数据区设定为页0~63。该指令仅对TABP p指令有效。			
		注意点: M34571G4/G6中, 该指令不能使用。			

[英文字母顺序] 机器指令一览表 (续)

RC (Reset Carry flag)				
机器码: D9 D0 <div style="display: flex; align-items: center; gap: 10px;"> 00000000110 ₂ 006 ₁₆ </div>	字数	周期数	标志CY	跳越条件
	1	1	0	—
功能: (CY) ← 0	分类: 运算指令 详细说明: 将进位标志 (CY) 清零。			
RCP (Reset Port C)				
机器码: D9 D0 <div style="display: flex; align-items: center; gap: 10px;"> 1010001100 ₂ 28C ₁₆ </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (C) ← 0	分类: 输入/输出指令 详细说明: 将端口C清零。			
RD (Reset port D specified by register Y)				
机器码: D9 D0 <div style="display: flex; align-items: center; gap: 10px;"> 0000010100 ₂ 014 ₁₆ </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (D(Y)) ← 0 (Y)=0~4	分类: 输入/输出指令 详细说明: 将由端口D的寄存器Y的内容把指定的端口清零。 注意点: (Y)=0~4 寄存器Y在指定范围外时, 不执行此指令。			
RT (ReTurn from subroutine)				
机器码: D9 D0 <div style="display: flex; align-items: center; gap: 10px;"> 00010000100 ₂ 044 ₁₆ </div>	字数	周期数	标志CY	跳越条件
	1	2	—	—
功能: (PC) ← (SK(SP)) (SP) ← (SP) - 1	分类: 返回指令 详细说明: 从子程序返回到调用此子程序的程序。			

[英文字母顺序] 机器指令一览表 (续)

RTI (ReTurn from Interrupt)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
0 0 0 1 0 0 0 1 1 0 ₂	0 4 6 ₁₆	1	2	—	—
功能: (PC) ← (SK(SP)) (SP) ← (SP) - 1		分类: 返回指令 详细说明: 从中断处理程序返回到主程序。将数据指针 (寄存器Z、X、Y)、进位标志 (CY)、跳越状态、由LA/LXY连续描述的NOP状态、寄存器A、寄存器B的各值返回到中断前的状态。			
RTS (ReTurn from subroutine and Skip)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
0 0 0 1 0 0 0 1 0 1 ₂	0 4 5 ₁₆	1	2	—	无条件跳越
功能: (PC) ← (SK(SP)) (SP) ← (SP) - 1		分类: 返回指令 详细说明: 从子程序返回到调用此子程序的程序, 无条件跳越下一条指令。			
RUPT (Reset UPT flag)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
0 0 0 1 0 1 1 0 0 0 ₂	0 5 8 ₁₆	1	1	—	—
功能: (UPTF) ← 0		分类: 其它 详细说明: 将高位参照允许标志UPTF清零。 注意点: 即使执行表参照指令 (TABP p指令), ROM内参照数据的高2位也不传送到寄存器D.			
SB j (Set Bit)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
0 0 0 1 0 1 1 1 j j ₂	0 5 $\frac{C}{j}$ ₁₆	1	1	—	—
功能: (Mj(DP)) ← 1 j=0~3		分类: 位操作指令 详细说明: 将M(DP)的第j位 (由立即字段的值j指定的位) 的内容置位 (1)。			

[英文字母顺序] 机器指令一览表 (续)

SBK (Set Bank flag)							
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件		
0 0 0 1 0 0 0 0 0 1	2	0 4 1	16	1	1	—	—
功能: 执行TABP p指令时: p ₆ ← 1		分类: 其他 详细说明: 把执行TABP p指令时参照的数据区设定为页64~127。该指令仅对TABP p指令有效。 注意点: 在M34571G4/G6中, 该指令不能使用。					
SC (Set Carry flag)							
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件		
0 0 0 0 0 0 0 1 1 1	2	0 0 7	16	1	1	1	—
功能: (CY) ← 1		分类: 运算指令 详细说明: 将进位标志 (CY) 置位 (1)。					
SCP (Set Port C)							
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件		
1 0 1 0 0 0 1 1 0 1	2	2 8 D	16	1	1	—	—
功能: (C) ← 1		分类: 输入/输出指令 详细说明: 将端口C置 (1)。					
SD (Set port D specified by register Y)							
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件		
0 0 0 0 0 1 0 1 0 1	2	0 1 5	16	1	1	—	—
功能: (D(Y)) ← 1 (Y)=0~4		分类: 输入/输出指令 详细说明: 将由端口D的寄存器Y的内容指定的端口置位 (1)。 注意点: (Y)=0~4 寄存器Y在指定范围外时, 不执行此指令。					

[英文字母顺序] 机器指令一览表 (续)

SEA n (Skip Equal, Accumulator with immediate data n)				
机器码: D ₉ D₀ <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div> 2 </div> <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 5px;"> <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">5</div> 16 </div>	字数	周期数	标志CY	跳越条件
	2	2	—	(A)=n n=0~15
机器码: D ₉ D₀ <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">n</div><div style="border: 1px solid black; padding: 2px;">n</div><div style="border: 1px solid black; padding: 2px;">n</div><div style="border: 1px solid black; padding: 2px;">n</div> 2 </div> <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 5px;"> <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">7</div><div style="border: 1px solid black; padding: 2px;">n</div> 16 </div>	分类: 比较指令			
功能: (A)=n ? n=0~15	详细说明: 如果寄存器A的内容和立即字段的值n相同, 就跳越下一条指令。否则就继续执行下一条指令。			
SEAM (Skip Equal, Accumulator with Memory)				
机器码: D ₉ D₀ <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div> 2 </div> <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 5px;"> <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">6</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	(A)=(M(DP))
功能: (A)=(M(DP)) ?	分类: 比较指令			
	详细说明: 如果寄存器A的内容和M(DP)的内容相同, 就跳越下一条指令。否则就继续执行下一条指令。			
SNZ0 (Skip if Non Zero condition of external interrupt 0 request flag)				
机器码: D ₉ D₀ <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div> 2 </div> <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 5px;"> <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">3</div><div style="border: 1px solid black; padding: 2px;">8</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	V1 ₀ =0 : (EXF0)=1
功能: V1 ₀ =0 : (EXF0)=1 ? (EXF0) ← 0 V1 ₀ =1 : SNZ0=NOP	分类: 中断控制指令			
	详细说明: 在中断控制寄存器V1的bit0 (V1 ₀) 的内容为“0”时, 如果外部0中断请求标志 (EXF0) 为“1”, 就将标志EXF0清零, 然后跳越下一条指令; 如果标志EXF0为“0”, 就继续执行下一条指令。 在中断控制寄存器V1的bit0 (V1 ₀) 的内容为“1”时, 此指令等价于NOP指令。			
SNZ1 (Skip if Non Zero condition of external interrupt 1 request flag)				
机器码: D ₉ D₀ <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div> 2 </div> <div style="display: flex; justify-content: space-around; align-items: center; margin-top: 5px;"> <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">3</div><div style="border: 1px solid black; padding: 2px;">9</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	V1 ₁ =0 : (EXF1)=1
功能: V1 ₁ =0 : (EXF1)=1 ? (EXF1) ← 0 V1 ₁ =1 : SNZ1=NOP	分类: 中断控制指令			
	详细说明: 在中断控制寄存器V1的bit1 (V1 ₁) 的内容为“0”时, 如果外部1中断请求标志 (EXF1) 为“1”, 就将标志EXF1清零, 然后跳越下一条指令; 如果标志EXF1为“0”, 就继续执行下一条指令。 在中断控制寄存器V1的bit1 (V1 ₁) 的内容为“1”时, 此指令等价于NOP指令。			

[英文字母顺序] 机器指令一览表 (续)

SNZIO (Skip if Non Zero condition of external Interrupt 0 input pin)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
0 0 0 0 1 1 1 0 1 0	0 3 A	1	1	—	I12=0: (INT0)=“L” I12=1: (INT0)=“H”
功能: I12=0: (INT0)=“L”? I12=1: (INT0)=“H”?		分类: 中断控制指令 详细说明: 在中断控制寄存器I1的bit2 (I12) 的内容为“0”时, 如果INT0引脚的电平为“L”, 就跳越下一条指令; 如果INT0引脚的电平为“H”, 就继续执行下一条指令。 在中断控制寄存器I1的bit2 (I12) 的内容为“1”时, 如果INT0引脚的电平为“H”, 就跳越下一条指令; 如果INT0引脚的电平为“L”, 就继续执行下一条指令。			
SNZI1 (Skip if Non Zero condition of external Interrupt 1 input pin)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
0 0 0 0 1 1 1 0 1 1	0 3 B	1	1	—	I22=0: (INT1)=“L” I22=1: (INT1)=“H”
功能: I22=0: (INT1)=“L”? I22=1: (INT1)=“H”?		分类: 中断控制指令 详细说明: 在中断控制寄存器I2的bit2 (I22) 的内容为“0”时, 如果INT1引脚的电平为“L”, 就跳越下一条指令; 如果INT1引脚的电平为“H”, 就继续执行下一条指令。 在中断控制寄存器I2的bit2 (I22) 的内容为“1”时, 如果INT1引脚的电平为“H”, 就跳越下一条指令; 如果INT1引脚的电平为“L”, 就继续执行下一条指令。			
SNZP (Skip if Non Zero condition of Power down flag)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
0 0 0 0 0 0 0 0 1 1	0 0 3	1	1	—	(P)=1
功能: (P)=1?		分类: 其它 详细说明: 如果掉电标志 (P) 的内容为“1”, 就跳越下一条指令, 如果为“0”, 就继续执行下一条指令。即使在跳越后, 标志P也不变。			
SNZT1 (Skip if Non Zero condition of Timer 1 interrupt request flag)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
1 0 1 0 0 0 0 0 0 0	2 8 0	1	1	—	V12=0: (T1F)=1
功能: V12=0: (T1F)=1? (T1F) ← 0 V12=1: SNZT1=NOP		分类: 定时器操作指令 详细说明: 在中断控制寄存器V1的bit2 (V12) 的内容为“0”时, 如果定时器1中断请求标志 (T1F) 为“1”, 就将标志T1F清零, 然后跳越下一条指令; 如果定时器1中断请求标志 (T1F) 为“0”, 就继续执行下一条指令。 在中断控制寄存器V1的bit2 (V12) 的内容为“1”时, 此指令等价于NOP指令。			

[英文字母顺序] 机器指令一览表 (续)

SNZT2 (Skip if Non Zero condition of Timer 2 interrupt request flag)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
1 0 1 0 0 0 0 0 0 1	2	2 8 1	1	—	V13=0: (T2F)=1
功能: V13=0: (T2F)=1? (T2F) ← 0 V13=1: SNZT2=NOP		分类: 定时器操作指令 详细说明: 在中断控制寄存器V1的bit3 (V13) 的内容为“0”时, 如果定时器2中断请求标志 (T2F) 为“1”, 就将标志T2F清零, 然后跳越下一条指令; 如果定时器2中断请求标志 (T2F) 为“0”, 就继续执行下一条指令。 在中断控制寄存器V1的bit3 (V13) 的内容为“1”时, 此指令等价于NOP指令。			
SNZT3 (Skip if Non Zero condition of Timer 3 interrupt request flag)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
1 0 1 0 0 0 0 0 1 0	2	2 8 2	1	—	V20=0: (T3F)=1
功能: V20=0: (T3F)=1? (T3F) ← 0 V20=1: SNZT3=NOP		分类: 定时器操作指令 详细说明: 在中断控制寄存器V2的bit0 (V20) 的内容为“0”时, 如果定时器3中断请求标志 (T3F) 为“1”, 就将标志T3F清零, 然后跳越下一条指令; 如果定时器3中断请求标志 (T3F) 为“0”, 就继续执行下一条指令。 在中断控制寄存器V2的bit0 (V20) 的内容为“1”时, 此指令等价于NOP指令。			
SNZVD (Skip if Non Zero condition of Voltage Detector interrupt request flag)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
1 0 1 0 0 0 1 0 1 0	2	2 8 A	1	—	V23=0: (VDF)=1
功能: V23=0: (VDF)=1? V23=1: SNZVD=NOP		分类: 其他 详细说明: 在中断控制寄存器V2的bit3 (V23) 的内容为“0”时, 如果低电压检测电路中断请求标志 (VDF) 为“1”, 就跳越下一条指令。标志VDF不清零。如果低电压检测电路中断请求标志 (VDF) 为“0”, 就继续执行下一条指令。 在中断控制寄存器V2的bit3 (V23) 的内容为“1”时, 此指令等价于NOP指令。			
SRST (System ReSeT)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
0 0 0 0 0 0 0 0 0 1	2	0 0 1	1	—	—
功能: 系统复位		分类: 其它 详细说明: 产生系统复位。			

[英文字母顺序] 机器指令一览表 (续)

SUPT (Set UPT flag)							
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件		
0 0 0 1 0 1 1 0 0 1	2	0 5 9	16	1	1	—	—
功能: (UPTF) ← 1		分类: 其它					
		详细说明: 将高位参照允许标志UPTF置位(1)。 注意点: 如果执行表参照指令(TABP p指令), 就将ROM内参照数据的高2位传送到寄存器D的低2位。					
SZB j (Skip if Zero, Bit)							
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件		
0 0 0 0 1 0 0 0 j j	2	0 2 j	16	1	1	—	(Mj(DP))=0 j=0~3
功能: (Mj(DP))=0? j=0~3		分类: 位操作指令					
		详细说明: 如果M(DP)的第j位(由立即字段的值j指定的位)的内容为“0”, 就跳越下一条指令, 为“1”时, 就继续执行下一条指令。					
SZC (Skip if Zero, Carry flag)							
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件		
0 0 0 0 1 0 1 1 1 1	2	0 2 F	16	1	1	—	(CY)=0
功能: (CY)=0?		分类: 运算指令					
		详细说明: 在进位标志(CY)的内容为“0”时, 就跳越下一条指令, 为“1”时, 就继续执行下一条指令。即使在跳越后, 标志CY也不变。					
SZD (Skip if Zero, port D specified by register Y)							
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件		
0 0 0 0 1 0 0 1 0 0	2	0 2 4	16	2	2	—	(D(Y))=0
功能: (D(Y))=0? (Y)=0~4		分类: 输入/输出指令					
		详细说明: 如果由端口D的寄存器Y的内容指定的端口的内容为“0”, 就跳越下一条指令, 为“1”时, 就继续执行下一条指令。 注意点: (Y)=0~4 寄存器Y在指定范围外时, 不执行此指令。					

[英文字母顺序] 机器指令一览表 (续)

T1AB (Transfer data to timer 1 and register R1 from Accumulator and register B)			
机器码: D9	D0	字数	周期数
1 0 0 0 1 1 0 0 0 0	2 3 0	1	1
功能: (T17~T14) ← (B) (R17~R14) ← (B) (T13~T10) ← (A) (R13~R10) ← (A)		分类: 定时器操作指令 详细说明: 将寄存器B的内容传送到定时器1的高4位 (T17~T14) 和定时器1的重加载寄存器R1的高4位 (R17~R14)、寄存器A的内容传送到定时器1的低4位 (T13~T10) 和重加载寄存器R1的低4位 (R13~R10)。	
T2AB (Transfer data to timer2 and register R2 from Accumulator and register B)			
机器码: D9	D0	字数	周期数
1 0 0 0 1 1 0 0 0 1	2 3 1	1	1
功能: (T27~T24) ← (B) (R27~R24) ← (B) (T23~T20) ← (A) (R23~R20) ← (A)		分类: 定时器操作指令 详细说明: 将寄存器B的内容传送到定时器2的高4位 (T27~T24) 和定时器2的重加载寄存器R2的高4位 (R27~R24)、寄存器A的内容传送到定时器2的低4位 (T23~T20) 和重加载寄存器R2的低4位 (R23~R20)。	
T3AB (Transfer data to timer 3 and register R3L from Accumulator and register B)			
机器码: D9	D0	字数	周期数
1 0 0 0 1 1 0 0 1 0	2 3 2	1	1
功能: (T37~T34) ← (B) (R3L7~R3L4) ← (B) (T33~T30) ← (A) (R3L3~R3L0) ← (A)		分类: 定时器操作指令 详细说明: 将寄存器B的内容传送到定时器3的高4位 (T37~T34) 和定时器3的重加载寄存器R3L的高4位 (R3L7~R3L4)、寄存器A的内容传送到定时器3的低4位 (T33~T30) 和重加载寄存器R3L的低4位 (R3L3~R3L0)。	
T3HAB (Transfer data to register R3H from Accumulator and register B)			
机器码: D9	D0	字数	周期数
1 0 0 0 1 1 1 1 0 1	2 3 D	1	1
功能: (R3H7~R3H4) ← (B) (R3H3~R3H0) ← (A)		分类: 定时器操作指令 详细说明: 将寄存器B的内容传送到定时器3的重加载寄存器R3H的高4位 (R3H7~R3H4)、寄存器A的内容传送到定时器3的重加载寄存器R3H的低4位 (R3H3~R3H0)。	

[英文字母顺序] 机器指令一览表 (续)

T3R3L (Transfer data to timer 3 from register R3L)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
1 0 0 0 1 1 0 1 0 0	2	2 3 4	16	—	—
功能: (T3 ₇ ~T3 ₀) ← (R3L ₇ ~R3L ₀)		分类: 定时器操作指令			
		详细说明: 将重加载寄存器R3L的内容传送到定时器3。			
TAB (Transfer data to Accumulator from register B)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
0 0 0 0 0 1 1 1 1 0	2	0 1 E	16	—	—
功能: (A) ← (B)		分类: 寄存器间传送指令			
		详细说明: 将寄存器B的内容传送到寄存器A。			
TAB1 (Transfer data to Accumulator and register B from timer 1)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
1 0 0 1 1 1 0 0 0 0	2	2 7 0	16	—	—
功能: (B) ← (T1 ₇ ~T1 ₄) (A) ← (T1 ₃ ~T1 ₀)		分类: 定时器操作指令			
		详细说明: 将定时器1的高4位 (T1 ₇ ~T1 ₄) 的内容传送到寄存器B、 定时器1的低4位 (T1 ₃ ~T1 ₀) 的内容传送到寄存器A。			
TAB2 (Transfer data to Accumulator and register B from timer 2)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
1 0 0 1 1 1 0 0 0 1	2	2 7 1	16	—	—
功能: (B) ← (T2 ₇ ~T2 ₄) (A) ← (T2 ₃ ~T2 ₀)		分类: 定时器操作指令			
		详细说明: 将定时器2的高4位 (T2 ₇ ~T2 ₄) 的内容传送到寄存器B、 定时器2的低4位 (T2 ₃ ~T2 ₀) 的内容传送到寄存器A。			

[英文字母顺序] 机器指令一览表 (续)

TAB3 (Transfer data to Accumulator and register B from timer 3)				
机器码: D ₉ D ₀	字数	周期数	标志CY	跳越条件
$\begin{array}{ c c c c c c c c c c } \hline 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ \hline \end{array}_2 \quad \begin{array}{ c c c } \hline 2 & 7 & 2 \\ \hline \end{array}_{16}$	1	1	—	—
功能: (B) ← (T37~T34) (A) ← (T33~T30)	分类: 定时器操作指令 详细说明: 将定时器3的高4位 (T37~T34) 的内容传送到寄存器B、定时器3的低4位 (T33~T30) 的内容传送到寄存器A。			
TABE (Transfer data to Accumulator and register B from register E)				
机器码: D ₉ D ₀	字数	周期数	标志CY	跳越条件
$\begin{array}{ c c c c c c c c c c } \hline 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ \hline \end{array}_2 \quad \begin{array}{ c c c } \hline 0 & 2 & A \\ \hline \end{array}_{16}$	1	1	—	—
功能: (B) ← (E7~E4) (A) ← (E3~E0)	分类: 寄存器间传送指令 详细说明: 将寄存器E的高4位 (E7~E4) 的内容传送到寄存器B、寄存器E的低4位 (E3~E0) 的内容传送到寄存器A。			
TABP p (Transfer data to Accumulator and register B from Program memory in page p)				
机器码: D ₉ D ₀	字数	周期数	标志CY	跳越条件
$\begin{array}{ c c c c c c c c c c } \hline 0 & 0 & 1 & 0 & p_5 & p_4 & p_3 & p_2 & p_1 & p_0 \\ \hline \end{array}_2 \quad \begin{array}{ c c c } \hline 0 & \frac{8}{+p} & p \\ \hline \end{array}_{16}$	1	3	—	—
功能: (SP) ← (SP) + 1 (SK(SP)) ← (PC) (PCH) ← p (PCL) ← (DR2~DR0, A3~A0) (B) ← (ROM(PC)) _{7~4} (A) ← (ROM(PC)) _{3~0} 在(UPTF)=1时, (DR1, DR0) ← (ROM(PC)) _{9, 8} (DR2) ← 0 (PC) ← (SK(SP)) (SP) ← (SP) - 1	分类: 运算指令 详细说明: 将由页p的寄存器D和寄存器A的内容指定的地址 (DR2DR1DR0A3A2A1A0) ₂ 的ROM模式区的bit7~4、bit3~0分别传送到寄存器B、寄存器A。在高位参照允许标志UPTF的内容为“1”时, 将ROM模式区的bit9、8传送到寄存器D的低2位 (DR1, DR0), 寄存器D的最高位 (DR2) 为“0”。在执行此指令时, 使用1段堆栈寄存器 (SK)。 注意点: M34571G4时, p=0~31 M34571G6时, p=0~47 M34571GD时, p=0~127 在执行TABP p指令时, 由于使用1段堆栈寄存器 (SK), 所以必须注意不要超出堆栈。			
TABPS (Transfer data to Accumulator and register B from Pre-Scaler)				
机器码: D ₉ D ₀	字数	周期数	标志CY	跳越条件
$\begin{array}{ c c c c c c c c c c } \hline 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 \\ \hline \end{array}_2 \quad \begin{array}{ c c c } \hline 2 & 7 & 5 \\ \hline \end{array}_{16}$	1	1	—	—
功能: (B) ← (TPS7~TPS4) (A) ← (TPS3~TPS0)	分类: 定时器操作指令 详细说明: 将预分频器的高4位的内容传送到寄存器B、预分频器的低4位的内容传送到寄存器A。			

[英文字母顺序] 机器指令一览表 (续)

TAD (Transfer data to Accumulator from register D)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div> 2 <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">5</div><div style="border: 1px solid black; padding: 2px;">1</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (A ₂ ~A ₀) ← (DR ₂ ~DR ₀) (A ₃) ← 0	分类: 寄存器间传送指令 详细说明: 将寄存器D的内容传送到寄存器A的低3位(A ₂ ~A ₀), 寄存器A的最高位(A ₃)为“0”。			
TAI1 (Transfer data to Accumulator from register I1)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div> 2 <div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">5</div><div style="border: 1px solid black; padding: 2px;">3</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (A) ← (I1)	分类: 中断控制指令 详细说明: 将中断控制寄存器I1的内容传送到寄存器A。			
TAI2 (Transfer data to Accumulator from register I2)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div> 2 <div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">5</div><div style="border: 1px solid black; padding: 2px;">4</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (A) ← (I2)	分类: 中断控制指令 详细说明: 将中断控制寄存器I2的内容传送到寄存器A。			
TAK0 (Transfer data to Accumulator from register K0)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div> 2 <div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">5</div><div style="border: 1px solid black; padding: 2px;">6</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (A) ← (K0)	分类: 输入/输出指令 详细说明: 将键唤醒控制寄存器K0的内容传送到寄存器A。			

[英文字母顺序] 机器指令一览表 (续)

TAK1 (Transfer data to Accumulator from register K1)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 0 1 0 1 1 0 0 1	2 2 5 9	1	1	—	—
功能: (A) ← (K1)		分类: 输入/输出指令 详细说明: 将键唤醒控制寄存器K1的内容传送到寄存器A。			
TAK2 (Transfer data to Accumulator from register K2)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 0 1 0 1 1 0 1 0	2 2 5 A	1	1	—	—
功能: (A) ← (K2)		分类: 输入/输出指令 详细说明: 将键唤醒控制寄存器K2的内容传送到寄存器A。			
TAL1 (Transfer data to Accumulator from register L1)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 0 1 0 0 1 0 1 0	2 2 4 A	1	1	—	—
功能: (A) ← (L1)		分类: 输入/输出指令 详细说明: 将键唤醒控制寄存器L1的内容传送到寄存器A。			
TAM j (Transfer data to Accumulator from Memory)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 1 1 0 0 j j j j	2 2 C j	1	1	—	—
功能: (A) ← (M(DP)) (X) ← (X)EXOR(j) j=0~15		分类: RAM • 寄存器间传送指令 详细说明: 在将M(DP)的内容传送到寄存器A后, 计算寄存器X的内容和立即字段的值j的逻辑异或, 其结果保存到寄存器X。			

[英文字母顺序] 机器指令一览表 (续)

TAMR (Transfer data to Accumulator from register MR)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
1 0 0 1 0 1 0 0 1 0	2	2 5 2	16	—	—
功能: (A) ← (MR)		分类: 时钟控制指令 详细说明: 将时钟控制寄存器MR的内容传送到寄存器A。			
TAPU0 (Transfer data to Accumulator from register PU0)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
1 0 0 1 0 1 0 1 1 1	2	2 5 7	16	—	—
功能: (A) ← (PU0)		分类: 输入/输出指令 详细说明: 将上拉控制寄存器PU0的内容传送到寄存器A。			
TAPU1 (Transfer data to Accumulator from register PU1)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
1 0 0 1 0 1 1 1 1 0	2	2 5 E	16	—	—
功能: (A) ← (PU1)		分类: 输入/输出指令 详细说明: 将上拉控制寄存器PU1的内容传送到寄存器A。			
TAPU2 (Transfer data to Accumulator from register PU2)					
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件
1 0 0 1 0 1 1 1 1 1	2	2 5 F	16	—	—
功能: (A) ← (PU2)		分类: 输入/输出指令 详细说明: 将上拉控制寄存器PU2的内容传送到寄存器A。			

[英文字母顺序] 机器指令一览表 (续)

TASP (Transfer data to Accumulator from Stack Pointer)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
0 0 0 1 0 1 0 0 0 0 ₂	0 5 0 ₁₆	1	1	—	—
功能: (A2 ~ A0) ← (SP2~SP0) (A3) ← 0		分类: 寄存器间传送指令 详细说明: 将堆栈指针(SP)的内容传送到寄存器A的低3位(A2~A0), 寄存器A的最高位(A3)为“0”。			
TAV1 (Transfer data to Accumulator from register V1)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
0 0 0 1 0 1 0 1 0 0 ₂	0 5 4 ₁₆	1	1	—	—
功能: (A) ← (V1)		分类: 中断控制指令 详细说明: 将中断控制寄存器V1的内容传送到寄存器A。			
TAV2 (Transfer data to Accumulator from register V2)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
0 0 0 1 0 1 0 1 0 1 ₂	0 5 5 ₁₆	1	1	—	—
功能: (A) ← (V2)		分类: 中断控制指令 详细说明: 将中断控制寄存器V2的内容传送到寄存器A。			
TAW1 (Transfer data to Accumulator from register W1)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 0 1 0 0 1 0 1 1 ₂	2 4 B ₁₆	1	1	—	—
功能: (A) ← (W1)		分类: 定时器操作指令 详细说明: 将定时器控制寄存器W1的内容传送到寄存器A。			

[英文字母顺序] 机器指令一览表 (续)

TAW2 (Transfer data to Accumulator from register W2)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div> 2 <div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">4</div><div style="border: 1px solid black; padding: 2px;">C</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (A) ← (W2)	分类: 定时器操作指令 详细说明: 将定时器控制寄存器W2的内容传送到寄存器A。			
TAW3 (Transfer data to Accumulator from register W3)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div> 2 <div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">4</div><div style="border: 1px solid black; padding: 2px;">D</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (A) ← (W3)	分类: 定时器操作指令 详细说明: 将定时器控制寄存器W3的内容传送到寄存器A。			
TAW5 (Transfer data to Accumulator from register W5)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div> 2 <div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">4</div><div style="border: 1px solid black; padding: 2px;">F</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (A) ← (W5)	分类: 定时器操作指令 详细说明: 将定时器控制寄存器W5的内容传送到寄存器A。			
TAX (Transfer data to Accumulator from register X)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div> 2 <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">5</div><div style="border: 1px solid black; padding: 2px;">2</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (A) ← (X)	分类: 寄存器间传送指令 详细说明: 将寄存器X的内容传送到寄存器A。			

[英文字母顺序] 机器指令一览表 (续)

TAY (Transfer data to Accumulator from register Y)				
机器码: D9 D0 <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div> 2 <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">F</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (A) ← (Y)	分类: 寄存器间传送指令 详细说明: 将寄存器Y的内容传送到寄存器A。			
TAZ (Transfer data to Accumulator from register Z)				
机器码: D9 D0 <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div> 2 <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">5</div><div style="border: 1px solid black; padding: 2px;">3</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (A1、A0) ← (Z1、Z0) (A3、A2) ← 0	分类: 寄存器间传送指令 详细说明: 将寄存器Z的内容传送到寄存器A的低2位 (A1、A0), 寄存器A的高2位 (A3、A2) 为“0”。			
TBA (Transfer data to register B from Accumulator)				
机器码: D9 D0 <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div> 2 <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">E</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (B) ← (A)	分类: 寄存器间传送指令 详细说明: 将寄存器A的内容传送到寄存器B。			
TDA (Transfer data to register D from Accumulator)				
机器码: D9 D0 <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div> 2 <div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">9</div> 16 </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (DR2~DR0) ← (A2~A0)	分类: 寄存器间传送指令 详细说明: 将寄存器A的低3位 (A2~A0) 的内容传送到寄存器D。			

[英文字母顺序] 机器指令一览表 (续)

TEAB (Transfer data to register E from Accumulator and register B)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
0 0 0 0 0 1 1 0 1 0 ₂	0 1 A ₁₆	1	1	—	—
功能: (E7~E4) ← (B) (E3~E0) ← (A)		分类: 寄存器间传送指令 详细说明: 将寄存器B的内容传送到寄存器E的高4位 (E7~E4)、寄存器A的内容传送到寄存器E的低4位 (E3~E0)。			
TFR0A (Transfer data to register FR0 from Accumulator)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 0 0 1 0 1 0 0 0 ₂	2 2 8 ₁₆	1	1	—	—
功能: (FR0) ← (A)		分类: 输入/输出指令 详细说明: 将寄存器A的内容传送到端口输出形式控制寄存器FR0。			
TFR1A (Transfer data to register FR1 from Accumulator)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 0 0 1 0 1 0 0 1 ₂	2 2 9 ₁₆	1	1	—	—
功能: (FR1) ← (A)		分类: 输入/输出指令 详细说明: 将寄存器A的内容传送到端口输出形式控制寄存器FR1。			
TI1A (Transfer data to register I1 from Accumulator)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 0 0 0 1 0 1 1 1 ₂	2 1 7 ₁₆	1	1	—	—
功能: (I1) ← (A)		分类: 中断控制指令 详细说明: 将寄存器A的内容传送到中断控制寄存器I1。			

[英文字母顺序] 机器指令一览表 (续)

T12A (Transfer data to register I2 from Accumulator)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 0 0 0 1 1 0 0 0	2 2 1 8	1	1	—	—
功能: (I2) ← (A)		分类: 中断控制指令			
		详细说明: 将寄存器A的内容传送到中断控制寄存器I2。			
TK0A (Transfer data to register K0 from Accumulator)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 0 0 0 1 1 0 1 1	2 2 1 B	1	1	—	—
功能: (K0) ← (A)		分类: 输入/输出指令			
		详细说明: 将寄存器A的内容传送到键唤醒控制寄存器K0。			
TK1A (Transfer data to register K1 from Accumulator)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 0 0 0 1 0 1 0 0	2 2 1 4	1	1	—	—
功能: (K1) ← (A)		分类: 输入/输出指令			
		详细说明: 将寄存器A的内容传送到键唤醒控制寄存器K1。			
TK2A (Transfer data to register K2 from Accumulator)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 0 0 0 1 0 1 0 1	2 2 1 5	1	1	—	—
功能: (K2) ← (A)		分类: 输入/输出指令			
		详细说明: 将寄存器A的内容传送到键唤醒控制寄存器K2。			

[英文字母顺序] 机器指令一览表 (续)

TL1A (Transfer data to register L1 from Accumulator)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 0 0 0 0 1 0 1 0	2 0 A	1	1	—	—
功能: (L1) ← (A)		分类: 输入/输出指令 详细说明: 将寄存器A的内容传送到键唤醒控制寄存器L1。			
TMA j (Transfer data to Memory from Accumulator)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 1 0 1 1 j j j j	2 B j	1	1	—	—
功能: (M(DP)) ← (A) (X) ← (X)EXOR(j) j=0~15		分类: RAM・寄存器间传送指令 详细说明: 在将寄存器A的内容传送到M(DP)后, 计算寄存器X的内容和立即字段的值j的逻辑异或, 其结果保存到寄存器X。			
TMRA (Transfer data to register MR from Accumulator)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 0 0 0 1 0 1 1 0	2 1 6	1	1	—	—
功能: (MR) ← (A)		分类: 时钟控制指令 详细说明: 将寄存器A的内容传送到时钟控制寄存器MR。			
TPAA (Transfer data to register PA from Accumulator)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 1 0 1 0 1 0 1 0	2 A A	1	1	—	—
功能: (PA0) ← (A0)		分类: 定时器操作指令 详细说明: 将寄存器A的最低位(A0)内容传送到定时器控制寄存器PA。			

[英文字母顺序] 机器指令一览表 (续)

TPSAB (Transfer data to Pre-Scaler and register RPS from Accumulator and register B)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 0 0 1 1 0 1 0 1	2 2 3 5	1	1	—	—
功能: (RPS7~RPS4) ← (B) (TPS7~TPS4) ← (B) (RPS3~RPS0) ← (A) (TPS3~TPS0) ← (A)		分类: 定时器操作指令 详细说明: 将寄存器B的内容传送到预分频器和预分频器重加载寄存器RPS的高4位、寄存器A的内容传送到预分频器和预分频器重加载寄存器RPS的低4位。			
TPU0A (Transfer data to register PU0 from Accumulator)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 0 0 1 0 1 1 0 1	2 2 2 D	1	1	—	—
功能: (PU0) ← (A)		分类: 输入/输出指令 详细说明: 将寄存器A的内容传送到上拉控制寄存器PU0。			
TPU1A (Transfer data to register PU1 from Accumulator)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 0 0 1 0 1 1 1 0	2 2 2 E	1	1	—	—
功能: (PU1) ← (A)		分类: 输入/输出指令 详细说明: 将寄存器A的内容传送到上拉控制寄存器PU1。			
TPU2A (Transfer data to register PU2 from Accumulator)					
机器码: D9	D0	字数	周期数	标志CY	跳越条件
1 0 0 0 1 0 1 1 1 1	2 2 2 F	1	1	—	—
功能: (PU2) ← (A)		分类: 输入/输出指令 详细说明: 将寄存器A的内容传送到上拉控制寄存器PU2。			

[英文字母顺序] 机器指令一览表 (续)

TR1AB (Transfer data to register R1 from Accumulator and register B)																				
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件															
1	0	0	0	1	1	1	1	1	1	1	1	2	2	3	F	16	1	1	—	—
功能: (R17~R14) ← (B) (R13~R10) ← (A)		分类: 输入/输出指令 详细说明: 将寄存器B的内容传送到重加载寄存器R1的高4位 (R17~R14), 寄存器A的内容传送到重加载寄存器R1的低4位 (R13~R10)。																		
TV1A (Transfer data to register V1 from Accumulator)																				
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件															
0	0	0	0	1	1	1	1	1	1	1	1	2	0	3	F	16	1	1	—	—
功能: (V1) ← (A)		分类: 中断控制指令 详细说明: 将寄存器A的内容传送到中断控制寄存器V1。																		
TV2A (Transfer data to register V2 from Accumulator)																				
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件															
0	0	0	0	1	1	1	1	1	1	0	2	0	3	E	16	1	1	—	—	
功能: (V2) ← (A)		分类: 中断控制指令 详细说明: 将寄存器A的内容传送到中断控制寄存器V2。																		
TW1A (Transfer data to register W1 from Accumulator)																				
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件															
1	0	0	0	0	0	1	1	1	0	2	2	0	E	16	1	1	—	—		
功能: (W1) ← (A)		分类: 定时器操作指令 详细说明: 将寄存器A的内容传送到定时器控制寄存器W1。																		

[英文字母顺序] 机器指令一览表 (续)

TW2A (Transfer data to register W2 from Accumulator)						
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件	
1 0 0 0 0 0 1 1 1 1	2	2 0 F	1	1	—	—
功能: (W2) ← (A)		分类: 定时器操作指令				
		详细说明: 将寄存器A的内容传送到定时器控制寄存器W2。				
TW3A (Transfer data to register W3 from Accumulator)						
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件	
1 0 0 0 0 1 0 0 0 0	2	2 1 0	1	1	—	—
功能: (W3) ← (A)		分类: 定时器操作指令				
		详细说明: 将寄存器A的内容传送到定时器控制寄存器W3。				
TW5A (Transfer data to register W5 from Accumulator)						
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件	
1 0 0 0 0 1 0 0 1 0	2	2 1 2	1	1	—	—
功能: (W5) ← (A)		分类: 定时器操作指令				
		详细说明: 将寄存器A的内容传送到定时器控制寄存器W5。				
TYA (Transfer data to register Y from Accumulator)						
机器码: D ₉	D ₀	字数	周期数	标志CY	跳越条件	
0 0 0 0 0 0 1 1 0 0	2	0 0 C	1	1	—	—
功能: (Y) ← (A)		分类: 寄存器间传送指令				
		详细说明: 将寄存器A的内容传送到寄存器Y。				

[英文字母顺序] 机器指令一览表 (续)

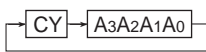
WRST (Watchdog timer ReSeT)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">0</div> ₂ <div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">A</div><div style="border: 1px solid black; padding: 2px;">0</div> ₁₆ </div>	字数	周期数	标志CY	跳越条件
	1	1	—	(WDF1)=1
功能: (WDF1)=1 ? (WDF1) ← 0	分类: 其它 详细说明: 如果看门狗定时器标志 (WDF1) 为 “1”, 就将标志WDF1 清零, 然后跳越下一条指令, 如果为 “0”, 就继续执行下一条指令。 另外, 如果在执行DWDWT指令后立即执行WRST指令, 就停止看门狗定时器的复位产生功能。			
XAM j (eXchange Accumulator and Memory data)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">j</div><div style="border: 1px solid black; padding: 2px;">j</div><div style="border: 1px solid black; padding: 2px;">j</div><div style="border: 1px solid black; padding: 2px;">j</div><div style="border: 1px solid black; padding: 2px;">j</div> ₂ <div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">D</div><div style="border: 1px solid black; padding: 2px;">j</div> ₁₆ </div>	字数	周期数	标志CY	跳越条件
	1	1	—	—
功能: (A) ← → (M(DP)) (X) ← (X)EXOR(j) j=0~15	分类: RAM和寄存器间传送指令 详细说明: 在将M(DP)的内容和寄存器A的内容交换后, 计算寄存器X的内容和立即字段的值j的逻辑异或, 其结果保存到寄存器X。			
XAMD j (eXchange Accumulator and Memory data and Decrement register Y and skip)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">j</div><div style="border: 1px solid black; padding: 2px;">j</div><div style="border: 1px solid black; padding: 2px;">j</div><div style="border: 1px solid black; padding: 2px;">j</div><div style="border: 1px solid black; padding: 2px;">j</div> ₂ <div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">F</div><div style="border: 1px solid black; padding: 2px;">j</div> ₁₆ </div>	字数	周期数	标志CY	跳越条件
	1	1	—	(Y)=15
功能: (A) ← → (M(DP)) (X) ← (X)EXOR(j) j=0~15 (Y) ← (Y) - 1	分类: RAM和寄存器间传送指令 详细说明: 在将M(DP)的内容和寄存器A的内容交换后, 计算寄存器X的内容和立即字段的值j的逻辑异或, 其结果保存到寄存器X。 另外, 将寄存器Y的内容减1, 如果其结果为 “15”, 就跳越下一条指令, 否则就继续执行下一条指令。			
XAMI j (eXchange Accumulator and Memory data and Increment register Y and skip)				
机器码: D ₉ D₀ <div style="display: flex; align-items: center; gap: 10px;"> <div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">1</div><div style="border: 1px solid black; padding: 2px;">0</div><div style="border: 1px solid black; padding: 2px;">j</div><div style="border: 1px solid black; padding: 2px;">j</div><div style="border: 1px solid black; padding: 2px;">j</div><div style="border: 1px solid black; padding: 2px;">j</div><div style="border: 1px solid black; padding: 2px;">j</div> ₂ <div style="border: 1px solid black; padding: 2px;">2</div><div style="border: 1px solid black; padding: 2px;">E</div><div style="border: 1px solid black; padding: 2px;">j</div> ₁₆ </div>	字数	周期数	标志CY	跳越条件
	1	1	—	(Y)=0
功能: (A) ← → (M(DP)) (X) ← (X)EXOR(j) j=0~15 (Y) ← (Y) + 1	分类: RAM和寄存器间传送指令 详细说明: 在将M(DP)的内容和寄存器A的内容交换后, 计算寄存器X的内容和立即字段的值j的逻辑异或, 其结果保存到寄存器X。 另外, 将寄存器Y的内容加1, 如果其结果为 “0”, 就跳越下一条指令, 否则就继续执行下一条指令。			

[按功能分类] 机器指令一览表

分类	指令符号	指令码		字数	周期数	功能
		D9 D8 D7 D6 D5 D4 D3 D2 D1 D0	16进制码			
寄存器间传送指令	TAB	0 0 0 0 0 1 1 1 1 0	0 1 E	1	1	(A) ← (B)
	TBA	0 0 0 0 0 0 1 1 1 0	0 0 E	1	1	(B) ← (A)
	TAY	0 0 0 0 0 1 1 1 1 1	0 1 F	1	1	(A) ← (Y)
	TYA	0 0 0 0 0 0 1 1 0 0	0 0 C	1	1	(Y) ← (A)
	TEAB	0 0 0 0 0 1 1 0 1 0	0 1 A	1	1	(E7~E4) ← (B) (E3~E0) ← (A)
	TABE	0 0 0 0 1 0 1 0 1 0	0 2 A	1	1	(B) ← (E7~E4) (A) ← (E3~E0)
	TDA	0 0 0 0 1 0 1 0 0 1	0 2 9	1	1	(DR2~DR0) ← (A2~A0)
	TAD	0 0 0 1 0 1 0 0 0 1	0 5 1	1	1	(A2~A0) ← (DR2~DR0) (A3) ← 0
	TAZ	0 0 0 1 0 1 0 0 1 1	0 5 3	1	1	(A1, A0) ← (Z1, Z0) (A3, A2) ← 0
	TAX	0 0 0 1 0 1 0 0 1 0	0 5 2	1	1	(A) ← (X)
	TASP	0 0 0 1 0 1 0 0 0 0	0 5 0	1	1	(A2~A0) ← (SP2~SP0) (A3) ← 0
RAM地址指令	LXY x, y	1 1 x3 x2 x1 x0 y3 y2 y1 y0	3 x y	1	1	(X) ← x, x=0~15 (Y) ← y, y=0~15
	LZ z	0 0 0 1 0 0 1 0 z1 z0	0 4 8 +z	1	1	(Z) ← z, z=0~3
	INY	0 0 0 0 0 1 0 0 1 1	0 1 3	1	1	(Y) ← (Y) + 1
	DEY	0 0 0 0 0 1 0 1 1 1	0 1 7	1	1	(Y) ← (Y) - 1
RAM和寄存器间传送指令	TAM j	1 0 1 1 0 0 j j j j	2 C j	1	1	(A) ← (M(DP)) (X) ← (X) EXOR (j), j=0~15
	XAM j	1 0 1 1 0 1 j j j j	2 D j	1	1	(A) ↔ (M(DP)) (X) ← (X) EXOR (j), j=0~15
	XAMD j	1 0 1 1 1 1 j j j j	2 F j	1	1	(A) ↔ (M(DP)) (X) ← (X) EXOR (j), j=0~15 (Y) ← (Y) - 1
	XAMI j	1 0 1 1 1 0 j j j j	2 E j	1	1	(A) ↔ (M(DP)) (X) ← (X) EXOR (j), j=0~15 (Y) ← (Y) + 1
	TMA j	1 0 1 0 1 1 j j j j	2 B j	1	1	(M(DP)) ← (A) (X) ← (X) EXOR (j), j=0~15

跳越条件	标志 CY	详细说明
—	—	将寄存器B的内容传送到寄存器A。
—	—	将寄存器A的内容传送到寄存器B。
—	—	将寄存器Y的内容传送到寄存器A。
—	—	将寄存器A的内容传送到寄存器Y。
—	—	将寄存器B的内容传送到寄存器E的高4位 (E7~E4)、寄存器A的内容传送到寄存器E的低4位 (E3~E0)。
—	—	将寄存器E的高4位 (E7~E4) 内容传送到寄存器B、寄存器E的低4位 (E3~E0) 内容传送到寄存器A。
—	—	将寄存器A的低3位 (A2~A0) 内容传送到寄存器D。
—	—	将寄存器D的内容传送到寄存器A的低3位 (A2~A0)、寄存器A的最高位 (A3) 为 “0”。
—	—	将寄存器Z的内容传送到寄存器A的低2位 (A1、A0)、寄存器A的高2位 (A3、A2) 为 “0”。
—	—	将寄存器X的内容传送到寄存器A。
—	—	将堆栈指针 (SP) 的内容传送到寄存器A的低3位 (A2~A0)、寄存器A的最高位 (A3) 为 “0”。
连续描述	—	将立即字段的值x装入寄存器X、立即字段的值y装入寄存器Y。 在连续描述并执行LXY指令时，除了最初执行的LXY指令外，跳越后面连续描述的LXY指令。
—	—	将立即字段的值z装入寄存器Z。
(Y)=0	—	将寄存器Y的内容加1。作为其结果，如果寄存器Y的内容为 “0”，就跳越下一条指令；如果不为 “0”，就继续执行下一条指令。
(Y)=15	—	将寄存器Y的内容减1。作为其结果，如果寄存器Y的内容为 “15”，就跳越下一条指令；如果不为 “15”，就继续执行下一条指令。
—	—	在将M(DP)的内容传送到寄存器A后，计算寄存器X的内容和立即字段的值j的逻辑异或，其结果保存到寄存器X。
—	—	在将M(DP)和寄存器A的内容交换后，计算寄存器X和立即字段的值j的逻辑异或，其结果保存到寄存器X。
(Y)=15	—	在将M(DP)和寄存器A的内容交换后，计算寄存器X的内容和立即字段的值j的逻辑异或，其结果保存到寄存器X。另外，将寄存器Y的内容减1，如果其结果为 “15”，就跳越下一条指令；如果不为 “15”，就继续执行下一条指令。
(Y)=0	—	在将M(DP)和寄存器A的内容交换后，计算寄存器X的内容和立即字段的值j的逻辑异或，其结果保存到寄存器X。另外，将寄存器Y的内容加1，如果其结果为 “0”，就跳越下一条指令。如果其结果不为 “0”，就继续执行下一条指令。
—	—	在将寄存器A的内容传送到M(DP)后，计算寄存器X的内容和立即字段的值j的逻辑异或，其结果保存到寄存器X。

[按功能分类] 机器指令一览表 (续)

分类	指令符号	指令码		字 数	周 期 数	功能
		D ₉ D ₈ D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ D ₀	16进制码			
运算 指令	LA n	0 0 0 1 1 1 n n n n	0 7 n	1	1	(A) ← n, n=0~15
	TABP p	0 0 1 0 p ₅ p ₄ p ₃ p ₂ p ₁ p ₀	0 8 p +p	1	3	(SP) ← (SP)+1 (SK(SP)) ← (PC) (PCH) ← p (注) (PCL) ← (DR ₂ ~ ₀ , A ₃ ~ ₀) (B) ← (ROM(PC)) ₇ ~ ₄ (A) ← (ROM(PC)) ₃ ~ ₀ 在(UPTF)=1时, (DR ₁ , DR ₀) ← (ROM(PC)) ₉ , ₈ (DR ₂) ← 0 (PC) ← (SK(SP)) (SP) ← (SP) - 1
	AM	0 0 0 0 0 0 1 0 1 0	0 0 A	1	1	(A) ← (A)+(M(DP))
	AMC	0 0 0 0 0 0 1 0 1 1	0 0 B	1	1	(A) ← (A)+(M(DP))+(CY) (CY) ← 进位
	A n	0 0 0 1 1 0 n n n n	0 6 n	1	1	(A) ← (A)+n, n=0~15
	AND	0 0 0 0 0 1 1 0 0 0	0 1 8	1	1	(A) ← (A) AND (M(DP))
	OR	0 0 0 0 0 1 1 0 0 1	0 1 9	1	1	(A) ← (A) OR (M(DP))
	SC	0 0 0 0 0 0 0 1 1 1	0 0 7	1	1	(CY) ← 1
	RC	0 0 0 0 0 0 0 1 1 0	0 0 6	1	1	(CY) ← 0
	SZC	0 0 0 0 1 0 1 1 1 1	0 2 F	1	1	(CY)=0?
	CMA	0 0 0 0 0 1 1 1 0 0	0 1 C	1	1	(A) ← (\bar{A})
RAR	0 0 0 0 0 1 1 1 0 1	0 1 D	1	1		
位 操 作 指 令	SB j	0 0 0 1 0 1 1 1 j j	0 5 C +j	1	1	(M _j (DP)) ← 1, j=0~3
	RB j	0 0 0 1 0 0 1 1 j j	0 4 C +j	1	1	(M _j (DP)) ← 0, j=0~3
	SZB j	0 0 0 0 1 0 0 0 j j	0 2 j	1	1	(M _j (DP))=0?, j=0~3

M34571G4: p=0~31

M34571G6: p=0~47

M34571GD: p=0~127

跳越条件	标志 CY	详细说明
连续描述	—	将立即字段的值n装入寄存器A。 在连续描述并执行LA指令时，除了最初执行的LA指令外，跳越后面连续描述的LA指令。
—	—	将由页p的寄存器D和寄存器A的内容指定的地址(DR2 DR1 DR0 A3 A2 A1 A0) ₂ 的ROM模式区的bit7~4、bit3~0分别传送到寄存器B、寄存器A。 在高位参照允许标志UPTF的内容为“1”时，将ROM模式区的bit9~8传送到寄存器D的低2位(DR1、DR0)，寄存器D的最高位(DR2)变为“0”。 在执行此指令时，使用1段堆栈(SK)。
—	—	寄存器A加上M(DP)的内容，结果保存到寄存器A。 进位的内容不变。
—	0/1	寄存器A加上M(DP)的内容和进位标志CY的内容，结果保存到寄存器A和标志CY。
上溢=0	—	寄存器A的内容加上立即字段的值n，结果保存到寄存器A。 进位标志(CY)的内容不变。如果运算结果上溢，就继续执行下一条指令。 如果不上溢，就跳越下一条指令。
—	—	计算寄存器A的内容和M(DP)的内容的逻辑与，结果保存到寄存器A。
—	—	计算寄存器A的内容和M(DP)的内容的逻辑或，结果保存到寄存器A。
—	1	将进位标志(CY)置位(1)。
—	0	将进位标志(CY)清零。
(CY)=0	—	在进位标志(CY)的内容为“0”时，跳越下一条指令。如果为“1”，就继续执行下一条指令。 跳越后标志(CY)也不变化。
—	—	将寄存器A的内容的1的补码保存到寄存器A。
—	0/1	包含进位标志(CY)，将寄存器A的内容向右循环1位。
—	—	将M(DP)的第j位(由立即字段的值j指定的位)的内容置位(1)。
—	—	将M(DP)的第j位(由立即字段的值j指定的位)的内容清零。
(Mj(DP))=0 其中，j=0~3	—	如果M(DP)的第j位(由立即字段的值j指定的位)的内容为“0”，就跳越下一条指令。 如果为“1”，就继续执行下一条指令。

[按功能分类] 机器指令一览表 (续)

分类	指令符号	指令码		字 数	周 期 数	功能
		D ₉ D ₈ D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ D ₀	16进制码			
比较指令	SEAM	0 0 0 0 1 0 0 1 1 0	0 2 6	1	1	(A)=(M(DP))?
	SEAn	0 0 0 0 1 0 0 1 0 1	0 2 5	2	2	(A)=n, n=0~15
		0 0 0 1 1 1 n n n n	0 7 n			
转移指令	Ba	0 1 1 a ₆ a ₅ a ₄ a ₃ a ₂ a ₁ a ₀	1 8 a +a	1	1	(PCL) ← a ₆ ~a ₀
	BL p, a	0 0 1 1 1 p ₄ p ₃ p ₂ p ₁ p ₀	0 E p +p	2	2	(PCH) ← p (注) (PCL) ← a ₆ ~a ₀
		1 p ₆ p ₅ a ₆ a ₅ a ₄ a ₃ a ₂ a ₁ a ₀	2 a a			
	BLA p	0 0 0 0 0 1 0 0 0 0	0 1 0	2	2	(PCH) ← p (注) (PCL) ← (DR ₂ ~DR ₀ , A ₃ ~A ₀)
		1 p ₆ p ₅ p ₄ 0 0 p ₃ p ₂ p ₁ p ₀	2 p p			
	子程序调用指令	BM a	0 1 0 a ₆ a ₅ a ₄ a ₃ a ₂ a ₁ a ₀	1 a a	1	1
BML p, a		0 0 1 1 0 p ₄ p ₃ p ₂ p ₁ p ₀	0 C p +p	2	2	(SP) ← (SP)+1 (SK(SP)) ← (PC) (PCH) ← p (注) (PCL) ← a ₆ ~a ₀
		1 p ₆ p ₅ a ₆ a ₅ a ₄ a ₃ a ₂ a ₁ a ₀	2 a a			
BMLA p		0 0 0 0 1 1 0 0 0 0	0 3 0	2	2	(SP) ← (SP)+1 (SK(SP)) ← (PC) (PCH) ← p (注) (PCL) ← (DR ₂ ~DR ₀ , A ₃ ~A ₀)
		1 p ₆ p ₅ p ₄ 0 0 p ₃ p ₂ p ₁ p ₀	2 p p			
返回指令		RTI	0 0 0 1 0 0 0 1 1 0	0 4 6	1	1
	RT	0 0 0 1 0 0 0 1 0 0	0 4 4	1	2	(PC) ← (SK(SP)) (SP) ← (SP) - 1
	RTS	0 0 0 1 0 0 0 1 0 1	0 4 5	1	2	(PC) ← (SK(SP)) (SP) ← (SP) - 1

M34571G4:p=0~31

M34571G6:p=0~47

M34571GD:p=0~127

跳越条件	标志 CY	详细说明
(A)=(M(DP))	—	如果寄存器A的内容和M(DP)的内容相同, 就跳越下一条指令。如果不相同, 就继续执行下一条指令。
(A)=n 其中, n=0~15	—	如果寄存器A的内容和立即字段的值n相同, 就跳越下一条指令。如果不相同, 就继续执行下一条指令。
—	—	页内转移: 转移到同一页的地址a。
—	—	页外转移: 转移到页p的地址a。
—	—	页外转移: 转移到由页p的寄存器D和寄存器A的内容指定的地址(DR2 DR1 DR0 A3 A2 A1 A0) ₂ 。
—	—	页2的子程序调用: 调用页2的地址a的子程序。
—	—	子程序调用: 调用页p的地址a的子程序。
—	—	子程序调用: 调用由页p的寄存器D和寄存器A的内容指定的地址(DR2 DR1 DR0 A3 A2 A1 A0) ₂ 的子程序。
—	—	从中断处理程序返回到主程序。 将数据指针(X、Y、Z)、进位标志CY、跳越状态、由LA/LXY连续描述的NOP状态、寄存器A、寄存器B的各值恢复到中断前的状态。
—	—	从子程序返回到调用此子程序的程序。
无条件跳越	—	从子程序返回到调用此子程序的程序, 无条件跳越下一条指令。

[按功能分类] 机器指令一览表 (续)

分类	指令符号	指令码		字数	周期数	功能
		D9 D8 D7 D6 D5 D4 D3 D2 D1 D0	16进制码			
中断控制指令	DI	0 0 0 0 0 0 0 1 0 0	0 0 4	1	1	(INTE) ← 0
	EI	0 0 0 0 0 0 0 1 0 1	0 0 5	1	1	(INTE) ← 1
	SNZ0	0 0 0 0 1 1 1 0 0 0	0 3 8	1	1	V10=0: (EXF0)=1? (EXF0) ← 0 V10=1: SNZ0=NOP
	SNZI0	0 0 0 0 1 1 1 0 1 0	0 3 A	1	1	I12=1: (INT0)=“L”? I12=0: (INT0)=“H”?
	SNZI	0 0 0 0 1 1 1 0 0 1	0 3 9	1	1	V11=0: (EXF1)=1? (EXF1) ← 0 V11=1: SNZI=NOP
	SNZI1	0 0 0 0 1 1 1 0 1 1	0 3 B	1	1	I22=0: (INT1)=“L”? I22=1: (INT1)=“H”?
	TAV1	0 0 0 1 0 1 0 1 0 0	0 5 4	1	1	(A) ← (V1)
	TV1A	0 0 0 0 1 1 1 1 1 1	0 3 F	1	1	(V1) ← (A)
	TAV2	0 0 0 1 0 1 0 1 0 1	0 5 5	1	1	(A) ← (V2)
	TV2A	0 0 0 0 1 1 1 1 1 0	0 3 E	1	1	(V2) ← (A)
	TAI1	1 0 0 1 0 1 0 0 1 1	2 5 3	1	1	(A) ← (I1)
	TI1A	1 0 0 0 0 1 0 1 1 1	2 1 7	1	1	(I1) ← (A)
	TAI2	1 0 0 1 0 1 0 1 0 0	2 5 4	1	1	(A) ← (I2)
	TI2A	1 0 0 0 0 1 1 0 0 0	2 1 8	1	1	(I2) ← (A)
定时器操作指令	TPAA	1 0 1 0 1 0 1 0 1 0	2 A A	1	1	(PA1、PA0) ← (A1、A0)
	TAW1	1 0 0 1 0 0 1 0 1 1	2 4 B	1	1	(A) ← (W1)
	TW1A	1 0 0 0 0 0 1 1 1 0	2 0 E	1	1	(W1) ← (A)
	TAW2	1 0 0 1 0 0 1 1 0 0	2 4 C	1	1	(A) ← (W2)
	TW2A	1 0 0 0 0 0 1 1 1 1	2 0 F	1	1	(W2) ← (A)
	TAW3	1 0 0 1 0 0 1 1 0 1	2 4 D	1	1	(A) ← (W3)
	TW3A	1 0 0 0 0 1 0 0 0 0	2 1 0	1	1	(W3) ← (A)
	TAW5	1 0 0 1 0 0 1 1 1 1	2 4 F	1	1	(A) ← (W5)
TW5A	1 0 0 0 0 1 0 0 1 0	2 1 2	1	1	(W5) ← (A)	

跳越条件	标志 CY	详细说明
—	—	将中断允许标志 (INTE) 清零, 设定成中断发生禁止状态。
—	—	将中断允许标志 (INTE) 置位 (1), 设定成中断发生允许状态。
V10=0:(EXF0)=1	—	在中断控制寄存器V1的bit0 (V10) 的内容为“0”时, 如果外部0中断请求标志 (EXF0) 为“1”, 就将外部0中断请求标志EXF0清零, 然后跳越下一条指令。 如果标志EXF0为“0”, 就继续执行下一条指令。 在中断控制寄存器V1的bit0 (V10) 的内容为“1”时, 此指令与NOP指令等价。
I12=0: (INT0)=“L”	—	在中断控制寄存器I1的bit2 (I12) 的内容为“0”时, 如果INT0引脚的电平为“L”, 就跳越下一条指令; 如果为“H”, 就继续执行下一条指令。
I12=1: (INT0)=“H”	—	在中断控制寄存器I1的bit2 (I12) 的内容为“1”时, 如果INT0引脚的电平为“H”, 就跳越下一条指令; 如果为“L”, 就继续执行下一条指令。
V11=0:(EXF1)=1	—	在中断控制寄存器V1的bit1 (V11) 的内容为“0”时, 如果外部1中断请求标志 (EXF1) 为“1”, 就将外部1中断请求标志EXF1清零, 然后跳越下一条指令。 如果标志EXF1为“0”, 就继续执行下一条指令。 在中断控制寄存器V1的bit1 (V11) 的内容为“1”时, 此指令与NOP指令等价。
I22=0: (INT1)=“L”	—	在中断控制寄存器I2的bit2 (I22) 的内容为“0”时, 如果INT1引脚的电平为“L”, 就跳越下一条指令; 如果为“H”, 就继续执行下一条指令。
I22=1: (INT1)=“H”	—	在中断控制寄存器I2的bit2 (I22) 的内容为“1”时, 如果INT1引脚的电平为“H”, 就跳越下一条指令; 如果为“L”, 就继续执行下一条指令。
—	—	将中断控制寄存器V1的内容传送到寄存器A。
—	—	将寄存器A的内容传送到中断控制寄存器V1。
—	—	将中断控制寄存器V2的内容传送到寄存器A。
—	—	将寄存器A的内容传送到中断控制寄存器V2。
—	—	将中断控制寄存器I1的内容传送到寄存器A。
—	—	将寄存器A的内容传送到中断控制寄存器I1。
—	—	将中断控制寄存器I2的内容传送到寄存器A。
—	—	将寄存器A的内容传送到中断控制寄存器I2。
—	—	将寄存器A的低2位 (A1、A0) 的内容传送到定时器控制寄存器PA。
—	—	将定时器控制寄存器W1的内容传送到寄存器A。
—	—	将寄存器A的内容传送到定时器控制寄存器W1。
—	—	将定时器控制寄存器W2的内容传送到寄存器A。
—	—	将寄存器A的内容传送到定时器控制寄存器W2。
—	—	将定时器控制寄存器W3的内容传送到寄存器A。
—	—	将寄存器A的内容传送到定时器控制寄存器W3。
—	—	将定时器控制寄存器W5的内容传送到寄存器A。
—	—	将寄存器A的内容传送到定时器控制寄存器W5。

[按功能分类] 机器指令一览表 (续)

分类	指令符号	指令码		字 数	周 期 数	功 能
		D9 D8 D7 D6 D5 D4 D3 D2 D1 D0	16 进制码			
定时器操作指令	TABPS	1 0 0 1 1 1 0 1 0 1	2 7 5	1	1	(B) ← (TPS7~TPS4) (A) ← (TPS3~TPS0)
	TPSAB	1 0 0 0 1 1 0 1 0 1	2 3 5	1	1	(RPS7~RPS4) ← (B) (TPS7~TPS4) ← (B) (RPS3~RPS0) ← (A) (TPS3~TPS0) ← (A)
	TAB1	1 0 0 1 1 1 0 0 0 0	2 7 0	1	1	(B) ← (T17~T14) (A) ← (T13~T10)
	T1AB	1 0 0 0 1 1 0 0 0 0	2 3 0	1	1	(R17~R14) ← (B) (T17~T14) ← (B) (R13~R10) ← (A) (T13~T10) ← (A)
	TR1AB	1 0 0 0 1 1 1 1 1 1	2 3 F	1	1	(R17~R14) ← (B) (R13~R10) ← (A)
	TAB2	1 0 0 1 1 1 0 0 0 1	2 7 1	1	1	(B) ← (T27~T24) (A) ← (T23~T20)
	T2AB	1 0 0 0 1 1 0 0 0 1	2 3 1	1	1	(R27~R24) ← (B) (T27~T24) ← (B) (R23~R20) ← (A) (T23~T20) ← (A)
	TAB3	1 0 0 1 1 1 0 0 1 0	2 7 2	1	1	(B) ← (T37~T34) (A) ← (T33~T30)
	T3AB	1 0 0 0 1 1 0 0 1 0	2 3 2	1	1	(R3L7~R3L4) ← (B) (T37~T34) ← (B) (R3L3~R3L0) ← (A) (T33~T30) ← (A)
	T3HAB	1 0 0 0 1 1 1 1 0 1	2 3 D	1	1	(R3H7~R3H4) ← (B) (R3H3~R3H0) ← (A)
	T3R3L	1 0 0 0 1 1 0 1 0 0	2 3 4	1	1	(T37) ← (R3L)
	SNZT1	1 0 1 0 0 0 0 0 0 0	2 8 0	1	1	V12=0: (T1F)=1? (T1F) ← 0 V12=1: SNZT1=NOP
	SNZT2	1 0 1 0 0 0 0 0 0 1	2 8 1	1	1	V13=0: (T2F)=1? (T2F) ← 0 V13=1: SNZT2=NOP
	SNZT3	1 0 1 0 0 0 0 0 1 0	2 8 2	1	1	V20=0: (T3F)=1? (T3F) ← 0 V20=1: SNZT3=NOP

跳越条件	标志 CY	详细说明
—	—	将预分频器的高4位 (TPS7~TPS4) 的内容传送到寄存器B。 将预分频器的低4位 (TPS3~TPS0) 的内容传送到寄存器A。
—	—	将寄存器B的内容传送到预分频器的高4位 (TPS7~TPS4) 和预分频器的重加载寄存器RPS的高4位 (RPS7~RPS4), 并且将寄存器A的内容传送到预分频器的低4位 (TPS3~TPS0) 和预分频器的重加载寄存器RPS的低4位 (RPS3~RPS0)。
—	—	将定时器1的高4位 (T17~T14) 的内容传送到寄存器B。 将定时器1的低4位 (T13~T10) 的内容传送到寄存器A。
—	—	将寄存器B的内容传送到定时器1的高4位 (T17~T14) 和定时器1的重加载寄存器R1L的高4位 (R17~R14), 并且将寄存器A的内容传送到定时器1的低4位 (T13~T10) 和定时器1的重加载寄存器R1L的低4位 (R13~R10)。
—	—	将寄存器B的内容传送到定时器1的重加载寄存器R1的高4位 (R17~R14)。 将寄存器A的内容传送到定时器1的重加载寄存器R1的低4位 (R13~R10)。
—	—	将定时器2的高4位 (T27~T24) 的内容传送到寄存器B。 将定时器2的低4位 (T23~T20) 的内容传送到寄存器A。
—	—	将寄存器B的内容传送到定时器2的高4位 (T27~T24) 和定时器2的重加载寄存器R2的高4位 (R27~R24), 并且将寄存器A的内容传送到定时器2的低4位 (T23~T20) 和定时器2的重加载寄存器R2L的低4位 (R23~R20)。
—	—	将定时器3的高4位 (T37~T34) 的内容传送到寄存器B。 将定时器3的低4位 (T33~T30) 的内容传送到寄存器A。
—	—	将寄存器B的内容传送到定时器3的高4位 (T37~T34) 和定时器3的重加载寄存器R3L的高4位 (R3L7~R3L4)、将寄存器A的内容传送到定时器3的低4位 (T33~T30) 和定时器3的重加载寄存器R3L的低4位、(R3L3~R3L0)。
—	—	将寄存器B的内容传送到定时器3的重加载寄存器R3H的高4位 (R3H7~R3H4)。 将寄存器A的内容传送到定时器3的重加载寄存器R3H的低4位 (R3H3~R3H0)。
—	—	定时器3的重加载寄存器R3L的内容传送到定时器3
V1 ₂ =0: (T1F)=1	—	在中断控制寄存器V1的bit2 (V1 ₂) 的内容为“0”时, 如果定时器1中断请求标志T1F为“1”, 就将定时器1中断请求标志T1F清零, 然后跳越下一条指令。 如果标志T1F为“0”, 就继续执行下一条指令。 在中断控制寄存器V1的bit2 (V1 ₂) 的内容为“1”时, 此指令与NOP指令等价。
V1 ₃ =0: (T2F)=1	—	在中断控制寄存器V1的bit3 (V1 ₃) 的内容为“0”时, 如果定时器2中断请求标志T2F为“1”, 就将定时器2中断请求标志T2F清零, 然后跳越下一条指令。 如果标志T2F为“0”, 就继续执行下一条指令。 在中断控制寄存器V1的bit3 (V1 ₃) 的内容为“1”时, 此指令与NOP指令等价。
V2 ₀ =0: (T3F)=1	—	在中断控制寄存器V2的bit0 (V2 ₀) 的内容为“0”时, 如果定时器3中断请求标志 (T3F) 为“1”, 就将定时器3中断请求标志T3F清零, 然后跳越下一条指令。如果标志T3F为“0”, 就继续执行下一条指令。在中断控制寄存器V2的bit0 (V2 ₀) 的内容为“1”时, 此指令与NOP指令等价。

[按功能分类] 机器指令一览表 (续)

分类	指令符号	指令码		字 数	周 期 数	功能	
		D9 D8 D7 D6 D5 D4 D3 D2 D1 D0	16进制码				
输入 /输出 指令	IAP0	1 0 0 1 1 0 0 0 0 0	2 6 0	1	1	(A) ← (P0)	
	OP0A	1 0 0 0 1 0 0 0 0 0	2 2 0	1	1	(P0) ← (A)	
	IAP1	1 0 0 1 1 0 0 0 0 1	2 6 1	1	1	(A) ← (P1)	
	OP1A	1 0 0 0 1 0 0 0 0 1	2 2 1	1	1	(P1) ← (A)	
	IAP2	1 0 0 1 1 0 0 0 1 0	2 6 2	1	1	(A ₁ 、A ₀) ← (P ₂₁ 、P ₂₀) (A ₃ 、A ₂) ← 0	
	OP2A	1 0 0 0 1 0 0 0 1 0	2 2 2	1	1	(P ₂₁ 、P ₂₀) ← (A ₁ 、A ₀)	
	IAP3	1 0 0 1 1 0 0 0 1 1	2 6 3	1	1	(A ₁ 、A ₀) ← (P ₃₁ 、P ₃₀) (A ₃ 、A ₂) ← 0	
	OP3A	1 0 0 0 1 0 0 0 1 1	2 2 3	1	1	(P ₃₁ 、P ₃₀) ← (A ₁ 、A ₀)	
	CLD	0 0 0 0 0 1 0 0 0 1	0 1 1	1	1	(D) ← 1	
	RD	0 0 0 0 0 1 0 1 0 0	0 1 4	1	1	(D(Y)) ← 0、(Y)=0~4	
	SD	0 0 0 0 0 1 0 1 0 1	0 1 5	1	1	(D(Y)) ← 1、(Y)=0~4	
	SZD	0 0 0 0 1 0 0 1 0 0	0 2 4	2	2	(D(Y))=0?、(Y)=0~4	
			0 0 0 0 1 0 1 0 1 1	0 2 B			
	RCP	1 0 1 0 0 0 1 1 0 0	2 8 C	1	1	(C) ← (0)	
	SCP	1 0 1 0 0 0 1 1 0 1	2 8 D	1	1	(C) ← (1)	
	TFR0A	1 0 0 0 1 0 1 0 0 0	2 2 8	1	1	(FR0) ← (A)	
	TFR1A	1 0 0 0 1 0 1 0 0 1	2 2 9	1	1	(FR1) ← (A)	
	TAPU0	1 0 0 1 0 1 0 1 1 1	2 5 7	1	1	(A) ← (PU0)	
	TPU0A	1 0 0 0 1 0 1 1 0 1	2 2 D	1	1	(PU0) ← (A)	
	TAPU1	1 0 0 1 0 1 1 1 1 0	2 5 E	1	1	(A) ← (PU1)	
	TPU1A	1 0 0 0 1 0 1 1 1 0	2 2 E	1	1	(PU1) ← (A)	
	TAPU2	1 0 0 1 0 1 1 1 1 1	2 5 F	1	1	(A) ← (PU2)	
	TPU2A	1 0 0 0 1 0 1 1 1 1	2 2 F	1	1	(PU2) ← (A)	
	IAK	1 0 0 1 1 0 1 1 1 1	2 6 F	1	1	(A0) ← (K) (A ₃ ~A ₁) ← 0	

跳越条件	标志 CY	详细说明
—	—	将端口P0的输入传送到寄存器A。
—	—	将寄存器A的内容输出到端口P0。
—	—	将端口P1的输入传送到寄存器A。
—	—	将寄存器A的内容输出到端口P1。
—	—	将端口P2的输入传送到寄存器A的低2位 (A ₁ 、A ₀)。 寄存器A的高2位 (A ₃ 、A ₂) 的内容为“0”。
—	—	将寄存器A的低2位 (A ₁ 、A ₀) 的内容输出到端口P2。
—	—	将端口P3的输入传送到寄存器A的低2位 (A ₁ 、A ₀)。 寄存器A的高2位 (A ₃ 、A ₂) 的内容为“0”。
—	—	将寄存器A的低2位 (A ₁ 、A ₀) 的内容输出到端口P3。
—	—	将端口D全部置位 (1)。
—	—	将由端口D的寄存器Y的内容指定的端口清零。
—	—	将由端口D的寄存器Y的内容指定的端口置位 (1)。
(D(Y))=0 Y=0~4	—	如果由端口D的寄存器Y的内容指定的端口的内容为“0”，跳越下一条指令。 如果为“1”，就继续执行下一条指令。
—	—	端口C清零。
—	—	端口C置位 (1)。
—	—	将寄存器A的内容传送到端口输出形式控制寄存器FR0。
—	—	将寄存器A的内容传送到端口输出形式控制寄存器FR1。
—	—	将上拉控制寄存器PU0的内容传送到寄存器A。
—	—	将寄存器A的内容传送到上拉控制寄存器PU0。
—	—	将上拉控制寄存器PU1的内容传送到寄存器A。
—	—	将寄存器A的内容传送到上拉控制寄存器PU1。
—	—	将上拉控制寄存器PU2的内容传送到寄存器A。
—	—	将寄存器A的内容传送到上拉控制寄存器PU2。
—	—	将端口K的输入传送到寄存器A的最低位 (A ₀)。 寄存器A的高3位 (A ₃ ~A ₁) 的内容为“0”。

[按功能分类] 机器指令一览表 (续)

分类	指令符号	指令码		字 数	周 期 数	功能
		D ₉ D ₈ D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ D ₀	16进制码			
输入/输出指令	TAK0	1 0 0 1 0 1 0 1 1 0	2 5 6	1	1	(A) ← (K0)
	TK0A	1 0 0 0 0 1 1 0 1 1	2 1 B	1	1	(K0) ← (A)
	TAK1	1 0 0 1 0 1 1 0 0 1	2 5 9	1	1	(A) ← (K1)
	TK1A	1 0 0 0 0 1 0 1 0 0	2 1 4	1	1	(K1) ← (A)
	TAK2	1 0 0 1 0 1 1 0 1 0	2 5 A	1	1	(A) ← (K2)
	TK2A	1 0 0 0 0 1 0 1 0 1	2 1 5	1	1	(K2) ← (A)
	TAL1	1 0 0 1 0 0 1 0 1 0	2 4 A	1	1	(A) ← (L1)
	TL1A	1 0 0 0 0 0 1 0 1 0	2 0 A	1	1	(L1) ← (A)
其它	TAMR	1 0 0 1 0 1 0 0 1 0	2 5 2	1	1	(A) ← (MR)
	TMRA	1 0 0 0 0 1 0 1 1 0	2 1 6	1	1	(MR) ← (A)
	NOP	0 0 0 0 0 0 0 0 0 0	0 0 0	1	1	(PC) ← (PC)+1
	POF	0 0 0 0 0 0 0 0 1 0	0 0 2	1	1	转移到RAM备份模式
	EPOF	0 0 0 1 0 1 1 0 1 1	0 5 B	1	1	POF指令有效
	SNZP	0 0 0 0 0 0 0 0 1 1	0 0 3	1	1	(P)=1?
	WRST	1 0 1 0 1 0 0 0 0 0	2 A 0	1	1	(WDF1)=1? (WDF1) ← 0
	DWDT	1 0 1 0 0 1 1 1 0 0	2 9 C	1	1	允许看门狗定时器功能停止
	SRST	0 0 0 0 0 0 0 0 0 1	0 0 1	1	1	系统复位
	RUPT	0 0 0 1 0 1 1 0 0 0	0 5 8	1	1	(UPTF) ← 0
	SUPT	0 0 0 1 0 1 1 0 0 1	0 5 9	1	1	(UPTF) ← 1
	SNZVD	1 0 1 0 0 0 1 0 1 0	2 8 A	1	1	V23=0: (VDF)=1? V23=1: SNZVD=NOP
RBK (注)	0 0 0 1 0 0 0 0 0 0	0 4 0	1	1	TABP p指令执行时: p ₆ ← 0	
SBK (注)	0 0 0 1 0 0 0 0 0 1	0 4 1	1	1	TABP p指令执行时: p ₆ ← 1	

注. M34571G4/G6 中, RBK 和 SBK 指令不能使用。

跳越条件	标志 CY	详细说明
—	—	将键唤醒控制寄存器K0的内容传送到寄存器A。
—	—	将寄存器A的内容传送到键唤醒控制寄存器K0。
—	—	将键唤醒控制寄存器K1的内容传送到寄存器A。
—	—	将寄存器A的内容传送到键唤醒控制寄存器K1。
—	—	将键唤醒控制寄存器K2的内容传送到寄存器A。
—	—	将寄存器A的内容传送到键唤醒控制寄存器K2。
—	—	将键唤醒控制寄存器L1的内容传送到寄存器A。
—	—	将寄存器A的内容传送到键唤醒控制寄存器L1。
—	—	将时钟控制寄存器MR的内容传送到寄存器A。
—	—	将寄存器A的内容传送到时钟控制寄存器MR。
—	—	空操作：程序计数器的值加“1”，其它不变。
—	—	如果在执行EPOF指令后执行POF指令，本产品将变为RAM备份模式。
—	—	如果执行EPOF指令，紧接的POF指令就有效。
(P)=1	—	如果掉电标志(P)的内容为“1”，就跳越下一条指令；如果为“0”，就继续执行下一条指令。即使在跳越后，掉电标志P的内容也不变。
(WDF1)=1	—	如果看门狗定时器标志(WDF1)为“1”，就将标志WDF1清(0)，然后跳越下一条指令。如果为“0”，就继续执行下一条指令。另外，如果在执行DWDWT指令后紧接着执行WRST指令，看门狗定时器的复位产生功能就无效。
—	—	如果执行DWDWT指令，就能通过紧接的WRST指令停止看门狗定时器功能。
—	—	产生系统复位。
—	—	将高位参照允许标志UPTF清(0)。
—	—	将高位参照允许标志UPTF置位(1)。
V2≠0:(VDF)=1	—	在中断控制寄存器V2的bit3(V2 ₃)的内容为“0”时，如果低电压检测电路中断请求标志(VDF)为“1”，就跳越下一条指令，标志VDF不清零。如果标志VDF为“0”，就继续执行下一条指令。 在中断控制寄存器V2的bit3(V2 ₃)的内容为“1”时，此指令与NOP指令等价。
—	—	TABP p指令执行时，把参照数据区设定为页0~63。 此指令仅对TABP p指令有效。
—	—	TABP p指令执行时，把参照数据区设定为页64~127。 此指令仅对TABP p指令有效。

指令码对应表

D3~D0	16进制码	D9~D4																010000	011000
		000000	000001	000010	000011	000100	000101	000110	000111	001000	001001	001010	001011	001100	001101	001110	001111	010111	011111
		00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10~17	18~1F
0000	0	NOP	BLA	SZB 0	BMLA	RBK ***	TASP	A 0	LA 0	TABP 0	TABP 16	TABP 32*	TABP 48**	BML	BML	BL	BL	BM	B
0001	1	SRST	CLD	SZB 1	—	SBK ***	TAD	A 1	LA 1	TABP 1	TABP 17	TABP 33*	TABP 49**	BML	BML	BL	BL	BM	B
0010	2	POF	—	SZB 2	—	—	TAX	A 2	LA 2	TABP 2	TABP 18	TABP 34*	TABP 50**	BML	BML	BL	BL	BM	B
0011	3	SNZP	INY	SZB 3	—	—	TAZ	A 3	LA 3	TABP 3	TABP 19	TABP 35*	TABP 51**	BML	BML	BL	BL	BM	B
0100	4	DI	RD	SZD	—	RT	TAV1	A 4	LA 4	TABP 4	TABP 20	TABP 36*	TABP 52**	BML	BML	BL	BL	BM	B
0101	5	EI	SD	SEAn	—	RTS	TAV2	A 5	LA 5	TABP 5	TABP 21	TABP 37*	TABP 53**	BML	BML	BL	BL	BM	B
0110	6	RC	—	SEAM	—	RTI	—	A 6	LA 6	TABP 6	TABP 22	TABP 38*	TABP 54**	BML	BML	BL	BL	BM	B
0111	7	SC	DEY	—	—	—	—	A 7	LA 7	TABP 7	TABP 23	TABP 39*	TABP 55**	BML	BML	BL	BL	BM	B
1000	8	—	AND	—	SNZ0	LZ 0	RUPT	A 8	LA 8	TABP 8	TABP 24	TABP 40*	TABP 56**	BML	BML	BL	BL	BM	B
1001	9	—	OR	TDA	SNZ1	LZ 1	SUPT	A 9	LA 9	TABP 9	TABP 25	TABP 41*	TABP 57**	BML	BML	BL	BL	BM	B
1010	A	AM	TEAB	TABE	SNZI0	LZ 2	—	A 10	LA 10	TABP 10	TABP 26	TABP 42*	TABP 58**	BML	BML	BL	BL	BM	B
1011	B	AMC	—	—	SNZI1	LZ 3	EPOF	A 11	LA 11	TABP 11	TABP 27	TABP 43*	TABP 59**	BML	BML	BL	BL	BM	B
1100	C	TYA	CMA	—	—	RB 0	SB 0	A 12	LA 12	TABP 12	TABP 28	TABP 44*	TABP 60**	BML	BML	BL	BL	BM	B
1101	D	—	RAR	—	—	RB 1	SB 1	A 13	LA 13	TABP 13	TABP 29	TABP 45*	TABP 61**	BML	BML	BL	BL	BM	B
1110	E	TBA	TAB	—	TV2A	RB 2	SB 2	A 14	LA 14	TABP 14	TABP 30	TABP 46*	TABP 62**	BML	BML	BL	BL	BM	B
1111	F	—	TAY	SZC	TV1A	RB 3	SB 3	A 15	LA 15	TABP 15	TABP 31	TABP 47*	TABP 63**	BML	BML	BL	BL	BM	B

上表为机器码和机器指令的对应表。D3~D0表示机器码的低4位，D9~D4表示机器码的高6位。同时，还表示了这些码的16进制码。有1字指令和2字指令2种，各种指令的第1字码如上表、2字指令的第2字码如下表所示。

注. 用“—”表示的码不能使用。

	第2字
BL	10 0aaa aaaa
BML	10 0aaa aaaa
BLA	10 0p00 pppp
BMLA	10 0p00 pppp
SEA	00 0111 nnnn
SZD	00 0010 1011

- M34571G4 中，不使用*、**、***。
- M34571G6，不使用**、***。
- M34571GD 的 TABP 指令在***（RBK，SBK 指令）中参照的页可切换。
 执行 RBK 指令后，TABP 指令中可参照的页变为 0~63。
 执行 SBK 指令后，TABP 指令中可参照的页变为 64~127（例如：TABP0 → TABP64）。
 如果不执行 SBK 指令，TABP 指令中可参照页通常为 0~63。

指令码对应表

D3~D0	16进制码	D9~D4	100000	100001	100010	100011	100100	100101	100110	100111	101000	101001	101010	101011	101100	101101	101110	101111	110000 111111
		20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F	30~3F	
0000	0	—	TW3A	OP0A	T1AB	—	—	IAP0	TAB1	SNZT1	—	WRST	TMA0	TAM0	XAM0	XAMI0	XAMD0	LXY	
0001	1	—	—	OP1A	T2AB	—	—	IAP1	TAB2	SNZT2	—	—	TMA1	TAM1	XAM1	XAMI1	XAMD1	LXY	
0010	2	—	TW5A	OP2A	T3AB	—	TAMR	IAP2	TAB3	SNZT3	—	—	TMA2	TAM2	XAM2	XAMI2	XAMD2	LXY	
0011	3	—	—	OP3A	—	—	TAI1	IAP3	—	—	—	—	TMA3	TAM3	XAM3	XAMI3	XAMD3	LXY	
0100	4	—	TK1A	—	T3R3L	—	TAI2	—	—	—	—	—	TMA4	TAM4	XAM4	XAMI4	XAMD4	LXY	
0101	5	—	TK2A	—	TPSAB	—	—	—	TABPS	—	—	—	TMA5	TAM5	XAM5	XAMI5	XAMD5	LXY	
0110	6	—	TMRA	—	—	—	TAK0	—	—	—	—	—	TMA6	TAM6	XAM6	XAMI6	XAMD6	LXY	
0111	7	—	TI1A	—	—	—	TAPU0	—	—	—	—	—	TMA7	TAM7	XAM7	XAMI7	XAMD7	LXY	
1000	8	—	TI2A	TFR0A	—	—	—	—	—	—	—	—	TMA8	TAM8	XAM8	XAMI8	XAMD8	LXY	
1001	9	—	—	TFR1A	—	—	TAK1	—	—	—	—	—	TMA9	TAM9	XAM9	XAMI9	XAMD9	LXY	
1010	A	TL1A	—	—	—	TAL1	TAK2	—	—	SNZVD	—	TPAA	TMA10	TAM10	XAM10	XAMI10	XAMD10	LXY	
1011	B	—	TK0A	—	—	TAW1	—	—	—	—	—	—	TMA11	TAM11	XAM11	XAMI11	XAMD11	LXY	
1100	C	—	—	—	—	TAW2	—	—	—	RCP	DWDT	—	TMA12	TAM12	XAM12	XAMI12	XAMD12	LXY	
1101	D	—	—	TPU0A	T3HAB	TAW3	—	—	—	SCP	—	—	TMA13	TAM13	XAM13	XAMI13	XAMD13	LXY	
1110	E	TW1A	—	TPU1A	—	—	TAPU1	—	—	—	—	—	TMA14	TAM14	XAM14	XAMI14	XAMD14	LXY	
1111	F	TW2A	—	TPU2A	TR1AB	TAW5	TAPU2	IAK	—	—	—	—	TMA15	TAM15	XAM15	XAMI15	XAMD15	LXY	

上表为机器码和机器指令的对应表。D3~D0表示机器码的低4位，D9~D4表示机器码的高6位。同时，还表示了这些码的16进制码。有1字指令和2字指令2种，各种指令的第1字码如上表、2字指令的第2字码如下表所示。

注. 用“—”表示的码不能使用。

	第2字
BL	10 0aaa aaaa
BML	10 0aaa aaaa
BLA	10 0p00 pppp
BMLA	10 0p00 pppp
SEA	00 0111 nnnn
SZD	00 0010 1011

电特性

表 25. 绝对最大额定值

符号	项 目	条 件	额定值	单位
VDD	电源电压	—	-0.3~6.5	V
VI	输入电压 P0、P1、P20/INT0、P21/INT1、P3、D0~D3、D4/CNTR0、RESET、XIN	—	-0.3~VDD+0.3	V
VO	输出电压 P0、P1、P2、P3、D0~D3、D4/CNTR0、RESET	输出晶体管为截止状态	-0.3~VDD+0.3	V
VO	输出电压C、XOUT	—	-0.3~VDD+0.3	V
Pd	功耗	Ta=25°C	300	mW
Topr	工作环境温度	—	-20~85	°C
Tstg	保存温度	—	-40~125	°C

表 26. 推荐运行条件 1 (在没有指定时, Ta=-20~85°C、VDD=1.8~5.5V)

符号	项 目	条 件	规格值			单位
			最小	典型	最大	
VDD	电源电压 (在使用陶瓷谐振器时)	f(STCK)≤6MHz	4		5.5	V
		f(STCK)≤4.4MHz	2.7		5.5	
		f(STCK)≤2.2MHz	2		5.5	
		f(STCK)≤1.1MHz	1.8		5.5	
VDD	电源电压 (使用外部时钟时)	f(STCK)≤4.8MHz	4		5.5	V
		f(STCK)≤3.2MHz	2.7		5.5	
		f(STCK)≤1.6MHz	2		5.5	
		f(STCK)≤0.8MHz	1.8		5.5	
VRAM	RAM保持电压	RAM备份模式	1.6		5.5	V
VSS	电源电压			0		V
VIH	“H”电平输入电压	P0、P1、P2、P3、D0~D4、K	0.8VDD		VDD	V
		XIN	0.7VDD		VDD	
		RESET、INT0、INT1	0.85VDD		VDD	
		CNTR0	0.85VDD		VDD	
VIL	“L”电平输入电压	P0、P1、P2、P3、D0~D4、K	0		0.3VDD	mA
		XIN	0		0.3VDD	
		RESET、INT0、INT1	0		0.3VDD	
		CNTR0	0		0.15VDD	
IOH(peak)	“H”电平峰值输出电流	P3、D0~D3	VDD=5V		-20	mA
			VDD=3V		-10	
		C、CNTR1	VDD=5V		-30	
			VDD=3V		-15	
IOH(avg)	“H”电平平均输出电流 (注1)	P3、D0~D3	VDD=5V		-10	mA
			VDD=3V		-5	
		C、CNTR1	VDD=5V		-15	
			VDD=3V		-7	
IOL(peak)	“L”电平峰值输出电流	P0、P1、P2、P3、D0~D4、C、 RESET、CNTR0、CNTR1	VDD=5V		24	mA
			VDD=3V		12	
IOL(avg)	“L”电平平均输出电流 (注1)	P0、P1、P2、P3、D0~D4、C、 RESET、CNTR0、CNTR1	VDD=5V		12	mA
			VDD=3V		6	
ΣIOH(avg)	“H”电平总电流	P3、D0~D3、C、CNTR1			-30	mA
ΣIOL(avg)	“L”电平总电流	P0、P10、P11、RESET			30	mA
		P10、P11、P2、P3、D0~D4、C、CNTR0、CNTR1、			30	

注1. 平均输出电流为100ms期间的平均值。

表 27. 推荐运行条件 1 (在没有指定时, $T_a = -20 \sim 85^\circ\text{C}$ 、 $V_{DD} = 1.8 \sim 5.5\text{V}$)

符号	项 目	条 件	规格值			单位			
			最小	典型	最大				
f(XIN)	振荡频率 (在使用陶瓷谐振器时)	through-mode	$V_{DD} = 4 \sim 5.5\text{V}$		6	MHz			
			$V_{DD} = 2.7 \sim 5.5\text{V}$		4.4				
			$V_{DD} = 2 \sim 5.5\text{V}$		2.2				
			$V_{DD} = 1.8 \sim 5.5\text{V}$		1.1				
		2分频模式	$V_{DD} = 2.7 \sim 5.5\text{V}$		6				
			$V_{DD} = 2 \sim 5.5\text{V}$		4.4				
			$V_{DD} = 1.8 \sim 5.5\text{V}$		2.2				
		4、8分频模式	$V_{DD} = 2 \sim 5.5\text{V}$		6				
$V_{DD} = 1.8 \sim 5.5\text{V}$			4.4						
f(XIN)	振荡频率 (使用外部时钟时)	through-mode	$V_{DD} = 4 \sim 5.5\text{V}$		4.8	MHz			
			$V_{DD} = 2.7 \sim 5.5\text{V}$		3.2				
			$V_{DD} = 2 \sim 5.5\text{V}$		1.6				
			$V_{DD} = 1.8 \sim 5.5\text{V}$		0.8				
		2分频模式	$V_{DD} = 2.7 \sim 5.5\text{V}$		4.8				
			$V_{DD} = 2 \sim 5.5\text{V}$		3.2				
			$V_{DD} = 1.8 \sim 5.5\text{V}$		1.6				
		4、8分频模式	$V_{DD} = 2 \sim 5.5\text{V}$		4.8				
			$V_{DD} = 1.8 \sim 5.5\text{V}$		3.2				
		f(CNTR)	定时器外部输入频率	CNTR0、CNTR1				$f(\text{STCK})/6$	Hz
		tw(CNTR)	定时器外部输入周期 (“H”和“L”脉宽)	CNTR0、CNTR1	$3/f(\text{STCK})$				s
		TPON	上电复位电路有效电源上升 时间(注1)	$V_{DD} = 0 \rightarrow 1.8\text{V}$				100	μs

注1. 当电源上升时间超过最大规格值时, 必须在RESET引脚和Vss引脚之间以最短距离连接电容, 并且在电源电压达到推荐运行电压的最小规格值之前对RESET引脚输入“L”电平。

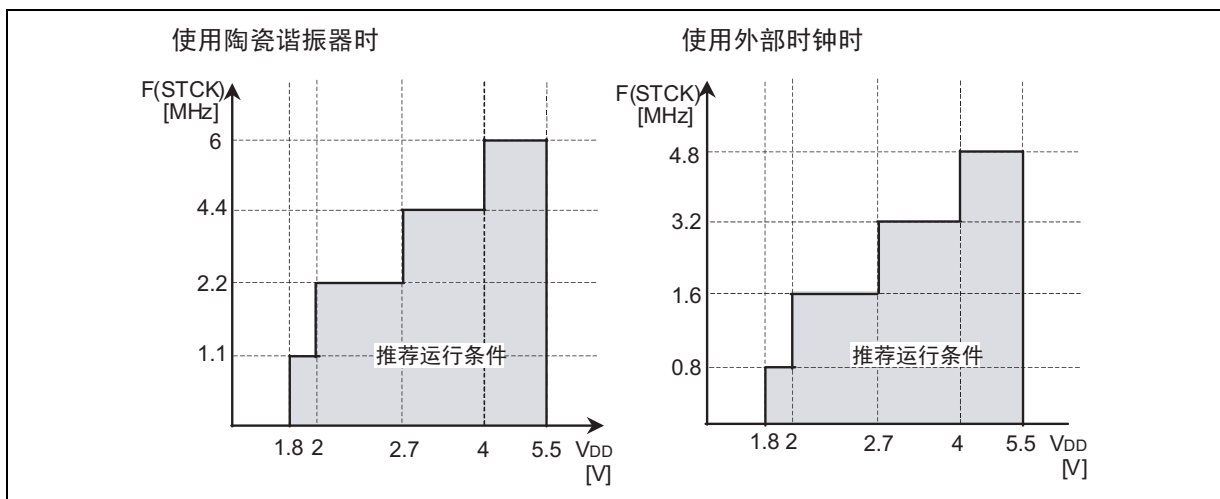


图 70. 系统时钟 (STCK) 运行条件映射

表 28. 电特性（在没有指定时， $T_a = -20 \sim 85^\circ\text{C}$ 、 $V_{DD} = 1.8 \sim 5.5\text{V}$ ）

符号	项目		测定条件		规格值			单位	
					最小	典型	最大		
VOH	“H”输出电压	P3、D0~D4 CNTR0	VDD=5V	IOH=-10mA	3			V	
				IOH=-3mA	4.1				
			VDD=3V	IOH=-5mA	2.1				
				IOH=-1mA	2.4				
VOH	“H”输出电压	C CNTR1	VDD=5V	IOH=-20mA	3			V	
				IOH=-6mA	4.1				
			VDD=3V	IOH=-10mA	2.1				
				IOH=-3mA	2.4				
VOL	“L”输出电压	P0、P1、P2、P3、D0~D4 $\overline{\text{RESET}}$ 、C、CNTR0、CNTR1	VDD=5V	IOH=15mA			2	V	
				IOH=5mA			0.9		
			VDD=3V	IOH=9mA			1.4		
				IOH=3mA			0.9		
IiH	“H”输入电流	P0、P1、P2、P3、D0~D4 K、 $\overline{\text{RESET}}$ 、INT0、INT1 CNTR0	Vi=VDD				2	μA	
IiL	“L”输入电流	P0、P1、P2、P3、D0~D4 K、 $\overline{\text{RESET}}$ 、INT0、INT1 CNTR0	Vi=0V P0、P1、P2 不选择上拉				-2	μA	
RPU	上拉电阻	P0、P1、P2 $\overline{\text{RESET}}$	Vi=0V	VDD=5V	30	60	125	k Ω	
				VDD=3V	50	120	250		
VT+ - VT-	滞后	$\overline{\text{RESET}}$ 、INT0、INT1	VDD=5V			1		V	
			VDD=3V			0.4			
VT+ - VT-	滞后	CNTR0	VDD=5V			0.2		V	
			VDD=3V			0.2			
IDD	电源电流	CPU运行时 (在使用陶瓷谐振器时) (注1)	VDD=5V f(XIN)=6MHz f(RING)=停止	f(STCK)=f(XIN)/8		1.2	2.4	mA	
				f(STCK)=f(XIN)/4		1.3	2.6		
				f(STCK)=f(XIN)/2		1.6	3.2		
				f(STCK)=f(XIN)		2.2	4.4		
			VDD=3V f(XIN)=2MHz f(RING)=停止	f(STCK)=f(XIN)/8		0.3	0.6	mA	
				f(STCK)=f(XIN)/4		0.4	0.8		
				f(STCK)=f(XIN)/2		0.6	1.2		
				f(STCK)=f(XIN)		0.8	1.6		
		RAM备份模式时 (在执行POF指令时)	Ta=25°C				0.1	3	μA
			VDD=5V					10	
VDD=3V						6			

注 1. 加上低电压检测电路运行电流 (IRST)。

表 29. 低电压检测电路特性（在没有指定时， $T_a = -20 \sim 85^\circ\text{C}$ ）

符号	项 目	测定条件	规格值			单位
			最小	典型	最大	
VRST ⁻	检测电压（产生复位）（注1）	Ta=25°C		1.65		V
		-20°C ≤ Ta < 0°C	1.6		2.2	
		0°C ≤ Ta < 50°C	1.3		2.1	
		50°C ≤ Ta ≤ 85°C	1.1		1.8	
VRST ⁺	检测电压（解除复位）（注2）	Ta=25°C		1.75		V
		-20°C ≤ Ta < 0°C	1.7		2.3	
		0°C ≤ Ta < 50°C	1.4		2.2	
		50°C ≤ Ta ≤ 85°C	1.2		1.9	
VINT	检测电压（产生中断）（注3）	Ta=25°C		1.85		V
		-20°C ≤ Ta < 0°C	1.8		2.4	
		0°C ≤ Ta < 50°C	1.5		2.3	
		50°C ≤ Ta ≤ 85°C	1.3		2.2	
VRST ⁺ - VRST ⁻	检测电压滞后			0.1		V
IRST	工作电流（注4）	VDD=5V		40	80	μA
		VDD=3V		20	40	
		VDD=1.65V		7	15	
TRST	判定时间（注5）	VDD → (VRST ⁻ - 0.1V)		0.2	1.2	ms

注1. 检测电压（VRST⁻）是在低电压检测电路有效时并且在电源电压（VDD）从高侧下降时产生复位的电压。

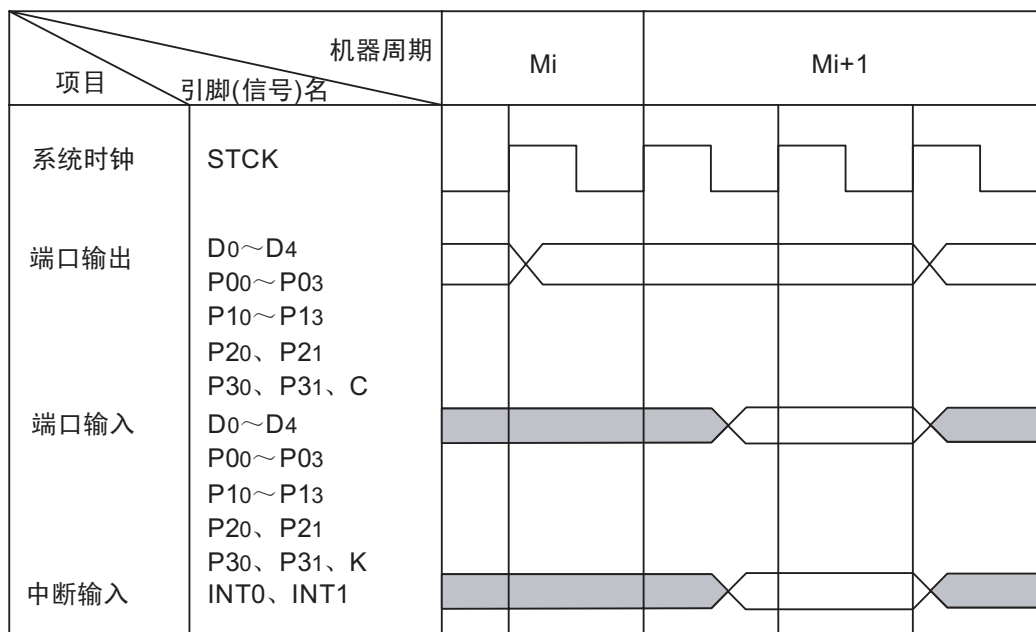
注2. 检测电压（VRST⁺）是在低电压检测电路有效时并且在电源电压（VDD）从高侧下降时产生复位的电压。

注3. 当电源电压（VDD）降低到低于检测电压（VINT）时，低电压检测电路中断请求标志（VDTF）被置位（1）。

注4. 把低电压检测电路运行电流（IRST）加到电源电流（IDD）。

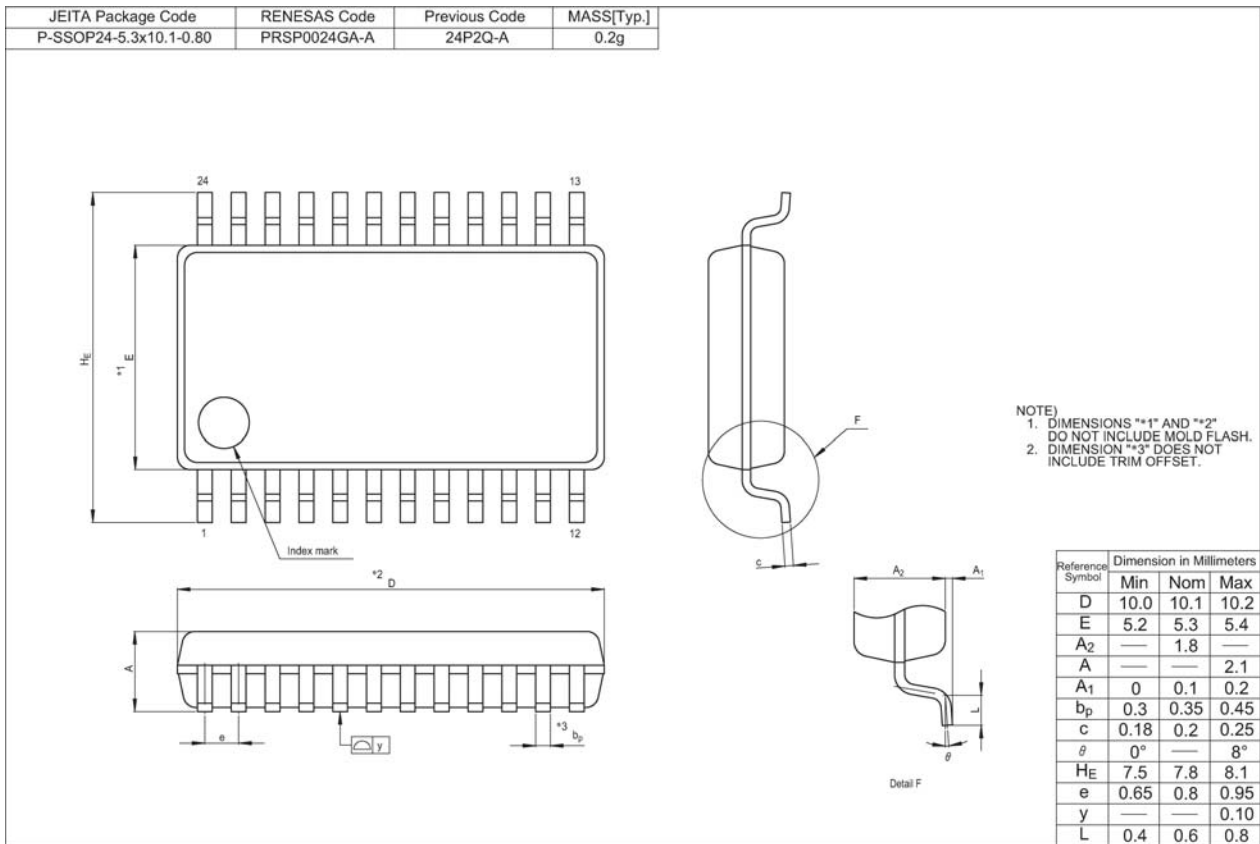
注5. 判定时间（TRST）是在电源电压（VDD）在从高侧下降到 [VRST⁻ - 0.1V] 时产生复位为止的时间。

基本时序图



封装尺寸图

封装尺寸图的最新版本和安装信息登载于瑞萨科技公司主页的“封装”项目中。



Rev.	发行日	修订内容	
		页	修订处
1.00	2006.09.13	—	初版发行
1.02	2008.03.07	全文	删除了“开发中”。
		4	在表 2 中追加了“上电复位电路”。
		10	将图 4 的“外部 0 中断”改为“外部 1 中断”。
		27~29	修改了图 24~图 29。
		32	将表 17 的“定时器控制寄存器 W1”的“CNTR1 输入”改为“CNTR0 输入”。
		40	修改了“看门狗定时器”的说明文。
		46	将“VDTF”改为“VDF”。
		52	将图 50 的“陶瓷谐振电路”改为“陶瓷振荡电路”。
		54~56	追加了“QzROM 编程模式”。
		58~60	修改了图 55~图 60。
		62~65	追加了“有关噪声的注意事项”。
		67	将“定时器控制寄存器 W1”的“CNTR1 输入”改为“CNTR0 输入”。
		73	将“SNZ1”的“V10”改为“V11”。
		111	修改了“TABP p”的功能。
		127、128	将表 26 的“在使用陶瓷谐振时”改为“在使用陶瓷谐振器时”。

Notes:

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

株式会社 瑞萨科技

下面所记中文只作为参考译文，英文具有正式效力。

请遵循安全第一进行电路设计:

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



RENESAS SALES OFFICES

<http://www.renesas.com>

Refer to "<http://www.renesas.com/en/network>" for the latest and detailed information.

Renesas Technology America, Inc.

450 Holger Way, San Jose, CA 95134-1368, U.S.A
Tel: <1> (408) 382-7500, Fax: <1> (408) 382-7501

Renesas Technology Europe Limited

Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K.
Tel: <44> (1628) 585-100, Fax: <44> (1628) 585-900

Renesas Technology (Shanghai) Co., Ltd.

Unit 204, 205, AZIACenter, No.1233 Lujiazui Ring Rd, Pudong District, Shanghai, China 200120
Tel: <86> (21) 5877-1818, Fax: <86> (21) 6887-7858/7898

Renesas Technology Hong Kong Ltd.

7th Floor, North Tower, World Finance Centre, Harbour City, Canton Road, Tsimshatsui, Kowloon, Hong Kong
Tel: <852> 2265-6688, Fax: <852> 2377-3473

Renesas Technology Taiwan Co., Ltd.

10th Floor, No.99, Fushing North Road, Taipei, Taiwan
Tel: <886> (2) 2715-2888, Fax: <886> (2) 3518-3399

Renesas Technology Singapore Pte. Ltd.

1 Harbour Front Avenue, #06-10, Keppel Bay Tower, Singapore 098632
Tel: <65> 6213-0200, Fax: <65> 6278-8001

Renesas Technology Korea Co., Ltd.

Kukje Center Bldg. 18th Fl., 191, 2-ka, Hangang-ro, Yongsan-ku, Seoul 140-702, Korea
Tel: <82> (2) 796-3115, Fax: <82> (2) 796-2145

Renesas Technology Malaysia Sdn. Bhd

Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No.18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: <603> 7955-9390, Fax: <603> 7955-9510