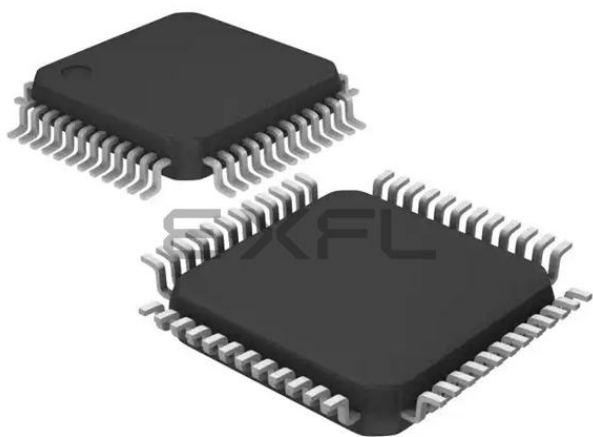




Welcome to [E-XFL.COM](#)

What is "[Embedded - Microcontrollers](#)"?

"[Embedded - Microcontrollers](#)" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "[Embedded - Microcontrollers](#)"

Details

Product Status	Obsolete
Core Processor	S08
Core Size	8-Bit
Speed	40MHz
Connectivity	CANbus, I ² C, LINbus, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	39
Program Memory Size	48KB (48K x 8)
Program Memory Type	FLASH
EEPROM Size	1.5K x 8
RAM Size	3K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 16x12b
Oscillator Type	External
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	48-LQFP
Supplier Device Package	48-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/s9s08dz48f1mlf

4.3 寄存器地址和位分配

MC9S08DZ60 系列产品中的寄存器可分为以下几组：

- 直接页面寄存器，位于存储器映象的前 128 个位置上。这些寄存器可以通过高效的直接寻址模式指令访问。
- 高端页面（High-page）寄存器，不经常使用，因此位于存储器映象中 0x1800 以上。在直接页面寄存器中为经常使用的寄存器和 RAM 留出了更多空间。
- 非易失性寄存器，由 Flash 中 0xFFB0 ~ 0xFFBF 之间 16 个位置组成的位置段组成。非易失性寄存器位置包括：
 - NVPROT 和 NVOPT，在复位时上载到工作寄存器中。
 - 一个 8 字节后门对比密钥，可选择为用户分配有控制的安全内存访问权限。

由于非易失性寄存器的位置是在 Flash 中，所以必须像其他位置 Flash 一样擦除和编程。

直接页面寄存器可以通过高效的直接寻址模式指令访问。位操作指令可用于访问任何直接页面寄存器中的任何位。表 4-2 总结了所有用户可访问的直接页面寄存器和控制位。

表 4-2 所列的直接页面寄存器可以使用更高效的直接寻址模式（这种模式只需要地址的较低字节）。因此，第 1 栏中地址的较低字节用粗体显示。在表 4-3 和表 4-5 中，第 1 栏中的整个地址都用粗体显示。在表 4-2, 表 4-3, 和表 4-5 中，第 2 栏中的寄存器名称用粗体显示以便与右侧的位名称区分。与所列出的位不相关的单元在阴影中显示。带有 0 的阴影单元表示这个未使用的位始终应为 0。带有破折号的阴影单元表示未使用的或预留的位，可以是 1 或 0。

6.5.2.5 B 端口驱动强度选择寄存器 (PTBDS)

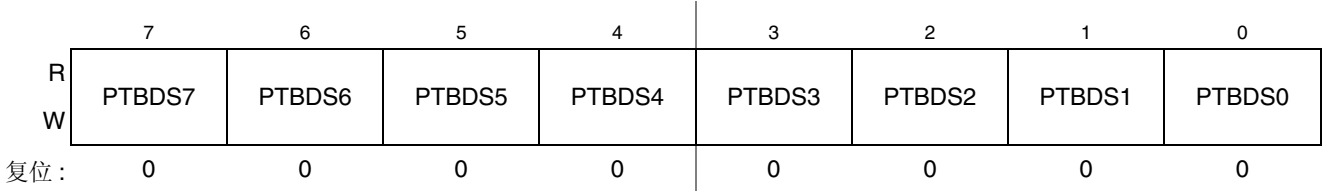


图 6-15. B 端口寄存器驱动强度选择 (PTBDS)

表 6-13. PTBDS 寄存器字段描述

字段	描述
7:0 PTBDS[7:0]	B 端口位的输出驱动强度选择 — 这些控制位为相关 PTB 管脚选择低输出驱动和高输出驱动。对于配置为输入的 B 端口管脚，这些位不会产生任何影响。 0 B 端口位 - 选择的低输出驱动强度。 1 B 端口位 - 选择的高输出驱动强度。

6.5.2.6 B 端口中断状态和控制寄存器 (PTBSC)

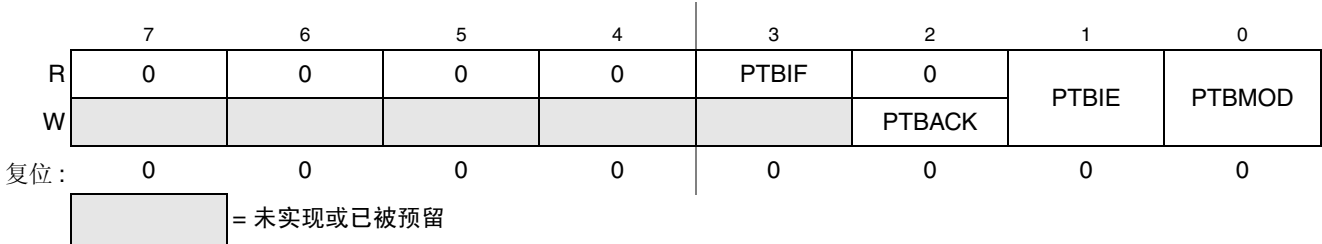


图 6-16. B 端口中断状态和控制寄存器 (PTBSC)

表 6-14. PTBSC 寄存器字段描述

字段	描述
3 PTBIF	B 端口中断标志 — PTBIF 显示是否检测到 B 端口中断。写入对 PTBIF 没有任何影响。 0 未检测到 B 端口中断。 1 检测到 B 端口中断。
2 PTBACK	B 端口中断确认 — 向 PTBACK 写入 1 是标记清除机制的一部分。PTBACK 的读数总为 0。
1 PTBIE	B 端口中断使能 — PTBIE 决定是否请求 B 端口中断。 0 B 端口中断请求禁止。 1 B 端口中断请求使能。
0 PTBMOD	B 端口检测模式 — PTBMOD (同 PTBES 位一起) 控制着 B 端口中断管脚的检测模式。 0 B 端口管脚只检测边沿。 1 B 端口管脚同时检测边沿和电平。

第 7 章

中央处理器（S08CPUV3）

7.1 介绍

本节简要地介绍了 HCS08 系列的寄存器、寻址模式和 CPU 指令集。如需了解更多信息，请参见 HCS08 系列参考手册第 1 卷，飞思卡尔半导体文档订单号 HCS08RMV1/D。

HCS08 CPU 和 M68HC08 CPU 的源和目标代码完全兼容。在 HCS08 的 CPU 中增加了几个指令和增强型寻址模式来提高 C 编译器效率、支持取代了早期 M68HC08 微控制器（MCU）监控模式的新背景调试系统。

7.1.1 特性

HCS08 CPU 的特性包括：

- 目标代码完全向上兼容 M68HC05 和 M68HC08 系列
- 所有寄存器和存储器都被映射到一个 64Kb 地址空间
- 16 位堆栈指针（64Kb 地址空间内任意规模的堆栈）
- 16 位索引寄存器（H:X），具有强大的索引寻址模式
- 8 位累加器（A）
- 很多指令将 X 当作备用的通用 8 位寄存器
- 7 个寻址模式：
 - Inherent — 内部寄存器里的操作数
 - Relative — 分支目的地的 8 位带符号偏移
 - Immediate — 下一个目标代码字节里的操作数
 - Direct — 0x0000 - 0x00FF 存储器的操作数
 - Extended — 64-Kb 地址空间里的操作数
 - Indexed relative to H:X — 5 个子模式，包括自动累加
 - Indexed relative to SP — 大幅提高 C 效率
- 存储器至存储器数据移动指令，具有 4 个地址模式组合
- 溢出、半进位、负数、零和进位条件代码支持在带符号、不带符号和十进制计数法（BCD）运算结果上的有条件转移
- 有效的位操控指令
- 快速的 8-bit by 8-bit 乘法和 16-bit by 8-bit 除法指令
- 调用低功率运行模式的 STOP 和 WAIT 指令

7.3.6.1 有索引、无偏移 (IX)

这个索引寻址变种将 H:X 索引寄存器对的 16 位地址作为完成指令所需的操作数地址。

7.3.6.2 有索引、无带后增量的偏移 (IX+)

这个索引寻址变种将 H:X 索引寄存器对的 16 位值作为完成指令所需的操作数地址。在获得操作数后，索引寄存器对然后被增加 ($H:X = H:X + 0x0001$)。这种寻址模式只用于 MOV 和 CBEQ 指令。

7.3.6.3 有索引、8 位偏移 (IX1)

这个索引寻址变种将 H:X 索引寄存器对和指令中不带符号的 8 位偏移作为完成指令所需的操作数地址。

7.3.6.4 有索引、带后增量的 8 位偏移 (IX1+)

这个索引寻址变种将 H:X 索引寄存器对和指令中不带符号的 8 位偏移作为完成指令所需的操作数地址。在获得操作数后，索引寄存器对然后被增加 ($H:X = H:X + 0x0001$)。这种寻址模式只用于 CBEQ 指令。

7.3.6.5 有索引、16 位偏移 (IX2)

这个索引寻址变种将 H:X 索引寄存器对和指令中的 16 位偏移作为完成指令所需的操作数地址。

7.3.6.6 SP 相关、8 位偏移 (SP1)

这个索引寻址变种将堆栈指针 (SP) 和指令中不带符号的 8 位偏移作为完成指令所需的操作数地址。

7.3.6.7 SP 相关、16 位偏移 (SP2)

这个索引寻址变种将堆栈指针 (SP) 和指令中的 16 位偏移作为完成指令所需的操作数地址。

表 7-2. 指令集小结 (第 3 页, 共 9 页)

Source Form	Operation	Address Mode	Object Code	Cycles	Cyc-by-Cyc Details	Affect on CCR	
						V 1 1 H	I N Z C
BRA <i>rel</i>	总是分支 (如果 I = 1)	REL	20 rr	3	ppp	- 1 1 -	- - - -
BRCLR <i>n,opr8a,rel</i>	如果存储器的位 <i>n</i> 清除, 分支 (如果 (Mn) = 0)	DIR (b0) DIR (b1) DIR (b2) DIR (b3) DIR (b4) DIR (b5) DIR (b6) DIR (b7)	01 dd rr 03 dd rr 05 dd rr 07 dd rr 09 dd rr 0B dd rr 0D dd rr 0F dd rr	5 5 5 5 5 5 5 5	rpppp rpppp rpppp rpppp rpppp rpppp rpppp rpppp	- 1 1 -	- - - ↑
BRN <i>rel</i>	从不分支 (如果 I = 0)	REL	21 rr	3	ppp	- 1 1 -	- - - -
BRSET <i>n,opr8a,rel</i>	如果存储器的位 <i>n</i> 设置, 分支 (如果 (Mn) = 1)	DIR (b0) DIR (b1) DIR (b2) DIR (b3) DIR (b4) DIR (b5) DIR (b6) DIR (b7)	00 dd rr 02 dd rr 04 dd rr 06 dd rr 08 dd rr 0A dd rr 0C dd rr 0E dd rr	5 5 5 5 5 5 5 5	rpppp rpppp rpppp rpppp rpppp rpppp rpppp rpppp	- 1 1 -	- - - ↑
BSET <i>n,opr8a</i>	在存储器里设置位 <i>n</i> (Mn ~ 1)	DIR (b0) DIR (b1) DIR (b2) DIR (b3) DIR (b4) DIR (b5) DIR (b6) DIR (b7)	10 dd 12 dd 14 dd 16 dd 18 dd 1A dd 1C dd 1E dd	5 5 5 5 5 5 5 5	rffwpp rffwpp rffwpp rffwpp rffwpp rffwpp rffwpp rffwpp	- 1 1 -	- - - -
BSR <i>rel</i>	分支到子程序 PC ~ (PC) + \$0002 推 (PCL); SP ~ (SP) - \$0001 推 (PCH); SP ~ (SP) - \$0001 PC ~ (PC) + <i>rel</i>	REL	AD rr	5	ssppp	- 1 1 -	- - - -
CBEQ <i>opr8a,rel</i> CBEQA # <i>opr8i,rel</i> CBEQX # <i>opr8i,rel</i> CBEQ <i>opr8,X+,rel</i> CBEQ <i>,X+,rel</i> CBEQ <i>opr8,SP,rel</i>	比较 ... 分支, 如果 (A) = (M) 分支, 如果 (A) = (M) 分支, 如果 (X) = (M) 分支, 如果 (A) = (M) 分支, 如果 (A) = (M) 分支, 如果 (A) = (M)	DIR IMM IMM IX1+ IX+ SP1	31 dd rr 41 ii rr 51 ii rr 61 ff rr 71 rr 9E 61 ff rr	5 4 4 5 5 6	rpppp pppp pppp rpppp rffppp prpppp	- 1 1 -	- - - -
CLC	清除进位 (C ~ 0)	INH	98	1	p	- 1 1 -	- - - 0
CLI	清除中断屏蔽位 (I ~ 0)	INH	9A	1	p	- 1 1 -	0 - - -
CLR <i>opr8a</i> CLRA CLR X CLR H CLR <i>opr8,X</i> CLR <i>,X</i> CLR <i>opr8,SP</i>	清除 M ~ \$00 A ~ \$00 X ~ \$00 H ~ \$00 M ~ \$00 M ~ \$00 M ~ \$00	DIR INH INH INH IX1 IX SP1	3F dd 4F 5F 8C 6F ff 7F 9E 6F ff	5 1 1 1 5 4 6	rffwpp p p p rffwpp rffw prffwpp	0 1 1 -	- 0 1 -

表 8-2. MCG 控制寄存器 2 字段描述

字段	描述
7:6 BDIV	总线分频器 — 选择被 MCGC1 寄存器中的 CLKS 位所选的时钟源除的数字。这样可以控制总线频率。 00 Encoding 0 — 用所选时钟除以 1 01 Encoding 1 — 用所选时钟除以 2 (复位默认) 10 Encoding 2 — 用所选时钟除以 4 11 Encoding 3 — 用所选时钟除以 8
5 RANGE	频率范围选择 — 选择外部振荡器或外部时钟源的频率范围。 1 为外部振荡器选择 1 MHz --16 MHz 的高频率范围 (外部时钟源为 1 MHz-- 40 MHz) 0 为外部振荡器选择 32 kHz --100 kHz 的低频率范围 (外部时钟源为 32 kHz --1 MHz)
4 HGO	高增益振荡器选择 — 控制外部振荡器的运行模式 1 配置外部振荡器的高增益运行 0 配置外部振荡器的低增益运行
3 LP	低功率选择 — 控制是否在旁路模式中禁止 FLL (或 PLL)。 1 旁路模式中禁止 FLL (或 PLL) (低功率)。 0 旁路模式中激活旁路模式中激活 FLL (或 PLL)。
2 EREFS	外部参考选择 选择外部参考源。 1 选择振荡器 0 选择外部时钟源
1 ERCLKEN	外部参考使能 — 使能外部参考时钟, 用作 MCGERCLK。 1 MCGERCLK 活动 0 MCGERCLK 未活动
0 EREFSTEN	外部参考停止使能 — 控制着当 MCG 进入停止模式时, 内部参考时钟是否仍保持使能状态。 1 如果设置了 IRCLKEN 或者在 MCG 进入停止状态前已处于 FEI、FBI 或 BLPI 模式, 那么外部参考时钟在停止状态中保持使能。 0 外部参考时钟在停止状态中禁用。

8.4.3 MCG 修正寄存器 (MCGTRM)

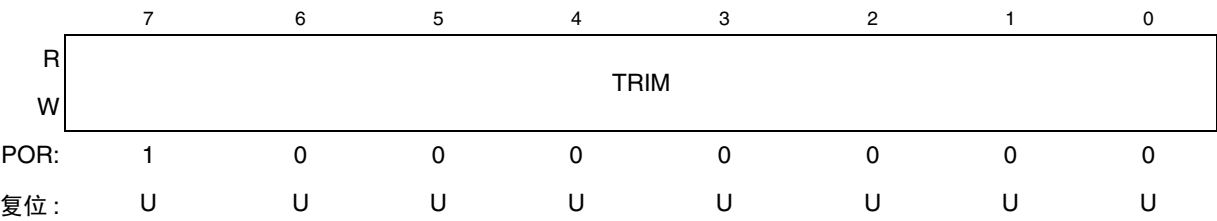


图 8-5. MCG 修正寄存器 (MCGTRM)

表 8-3. MCG 修正寄存器字段描

字段	描述
7:0 TRIM	MCG 修正设置 — 通过控制内部参考时钟时段来控制内部参考时钟频率。TRIM 位是加权二进位 (即位 1 的调整次数是位 0 调整次数的两倍)。增大 TRIM 中的二进位值将延长这个时段, 减小 TRIM 值会缩短该时段。 MCGSC 中还提供了另外一个微调位, 即 FTRIM 位 t. 如果使用保存在非易失性存储器中的 TRIM[7:0] 值, 用户就有责任将这个值从非易失性存储器位置复制到该寄存器。r.

第 11 章

IIC 模块 (S08IICV2)

11.1 介绍

IIC 模块 (IIC) 提供了不同器件间的通信方法。接口可以在最大的总线负载和时序下，支持最高 100kbps 的传输速率。器件可以在较低总线负载下、以更高的波特率（最高时钟 /20）运行。最大通信长度和可以连接的器件数量受 400 pF 的最高总线电容限制。

MC9S08DZ60 系列的所有 MCU 都具有 IIC 功能，如以下结构图所示。

注意

在使用 IIC 模块时，禁止为 IIC 管脚使用驱动强度（DSE=0），以实现正确操作。

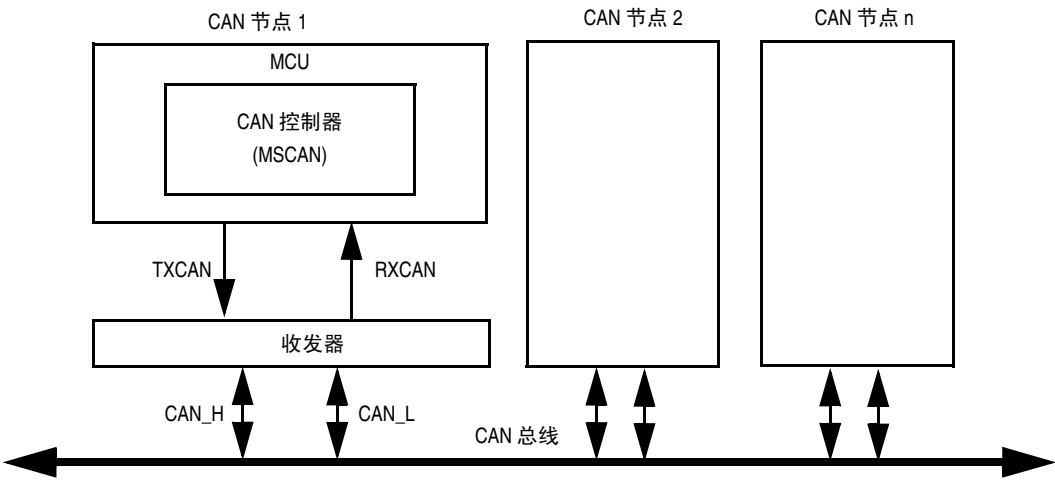


图 12-3. CAN 系统

12.3 寄存器定义

本节详细描述 MSCAN 模块中的所有寄存器和寄存器位。每个描述都包括带有相关图形编号的标准寄存器示意图。寄存器位和字段功能的详细说明在寄存器图后面，按位顺序。该模块中所有寄存器的所有位在寄存器读取过程中都与内部时钟完全同步。

12.3.1 MSCAN 控制寄存器 0 (CANCTL0)

The CANCTL0 寄存器提供了如下所述的 MSCAN 模块的各种位控制。

.

	7	6	5	4	3	2	1	0
R	RXFRM	RXACT	CSWAI	SYNCH	TIME	WUPE	SLPRQ	INITRQ
W								
复位：	0	0	0	0	0	0	0	1
	= Unimplemented							

图 12-4. 控制寄存器 0 (CANCTL0)

注意

当初始化模式处于有效 (INITRQ = 1 and INITAK = 1). 时，除 WUPE、INITRQ 和 SLPRQ 外的所有 CANCTL0 寄存器位都处于复位状态。只要退出初始化模式 (INITRQ = 0, INITAK = 0)，该寄存器可以再次写入。

读取：任何时间

写入：退出初始化模式的任何时间；例外是只读 RXACT、SYNCH、RXFRM （只由该模块设置）和 INITRQ （也可以在初始化模式中写入）。.

表 12-1. CANCTL0 寄存器字段描述

字段	描述
7 RXFRM ¹	已收到帧标记— 该位是只读和只清除位。当接收器正确收到有效报文（独立于滤波器配置）时，设置该位。设置后，该位一直保持设置，直到通过软件或复位将其清除。清除通过写入 1 完成。写 0 被忽略。该位在环回模式中无效。 0 自上次清除该标记以来未收到有效报文 1 自上次清除该标记以来收到有效报文
6 RXACT	接收器使能状态— 该只读标记表示 MSCAN 正在接收报文。该标记由接收器前端控制。该位在环回模式中无效。 0 MSCAN 正在发送或空闲 ² 1 MSCAN 正在接收报文（包括仲裁丢失时） ²
5 CSWAI ²	在等待模式中 CAN 停止— 设置此位，可以在等待模式中通过禁止 MSCAN 模块与 CPU 总线接口的所有时钟而降低功耗。 0 在等待模式中，CAN 模块不受影响 1 等待模式中，CAN 模块停止计时
4 SYNCH	同步状态— 该只读标记显示 MSCAN 是否与 CAN 总线同步，是否能够参与通信流程。其设置和清除通过 MSCAN 进行。 0 MSCAN 与 CAN 总线不同步 1 MSCAN 与 CAN 总线同步
3 TIME	计时器使能— 该位使能内部 16 位字宽自由运行计时器，由位时钟速率计时。如果计时器被使能，16 位时间标签将分配给有效 TX/RX 缓冲器内的每条发送 / 接收报文。一旦报文在 CAN 总线上确认，时间标签将被写入适当缓冲器（参见 12.4，“报文存储模式”）的最高字节（0x000E, 0x000F）。禁止时，内部计时器复位（所有位都设置为 0）。该位在初始化模式中保持低。 0 禁止内部 MSCAN 计时器 1 使能内部 MSCAN 计时器
2 WUPE ³	唤醒使能— 当检测到 CAN 上有流量时（参见 12.5.5.4，“MSCAN 睡眠模式”），该配置位能够让 MSCAN 从睡眠模式中重启。为了让所选功能发挥作用，在该位进入睡眠模式前必须进行配置。 0 唤醒禁止 — MSCAN 忽略 CAN 上的流量 1 唤醒使能 — MSCAN 能够重启

Register Name		Bit 7	6	5	4	3	2	1	Bit0
IDR0	R W	ID28	ID27	ID26	ID25	ID24	ID23	ID22	ID21
IDR1	R W	ID20	ID19	ID18	SRR ⁽¹⁾	IDE ⁽¹⁾	ID17	ID16	ID15
IDR2	R W	ID14	ID13	ID12	ID11	ID10	ID9	ID8	ID7
IDR3	R W	ID6	ID5	ID4	ID3	ID2	ID1	ID0	RTR ²
DSR0	R W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
DSR1	R W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
DSR2	R W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
DSR3	R W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
DSR4	R W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
DSR5	R W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
DSR6	R W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
DSR7	R W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
DLR	R W					DLC3	DLC2	DLC1	DLC0

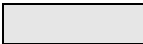
 = Unused, always read 0

图 12-23. 接收 / 发送报文缓冲器 — 扩展标识符映射

¹ SRR 和 IDE 都为 1。

² RTR 的位置在扩展和标准标识符映射间存在差异。

12.4.5 发送缓冲器优先寄存器 (TBPR)

表 12-33. 数据长度代码

数据长度代码				数据字节计数 Count
DLC3	DLC2	DLC1	DLC0	
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8

该寄存器定义相关报文发送缓冲器的本地优先级。本地优先级用于 MSCAN 的内部优先级排队程序，优先级定义为最小二进制数字取得最高优先级。MSCAN 执行下列内部优先级排队机制：

- 带有 TXEx 清除标志的所有发送缓冲器在发送 SOF（帧开始）前立即参与优先级排队。
- 带有最低本地优先级字段的发送缓冲器优先。

当出现一个以上的缓冲器具有相同最低优先级情况时，索引编号较小的报文缓冲器优先。

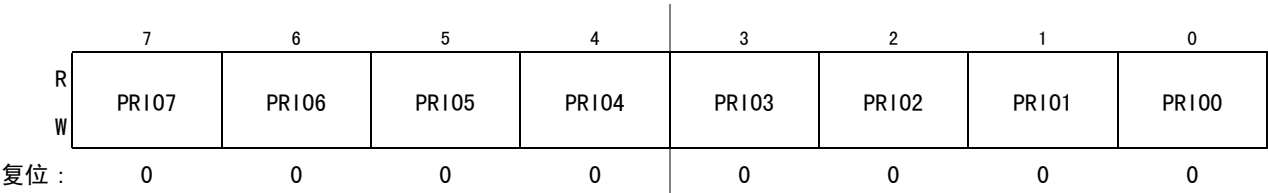


图 12-35. 发送缓冲器优先寄存器 (TBPR)

读取：当设置了 TXEx 标志（参见 12.3.6，“MSCAN 发送器标志寄存器 (CANTFLG)”）且在 CANTBSEL 中选择了相应发送缓冲器（见 12.3.10，“MSCAN 发送缓冲器选择寄存器 (CANTBSEL)”）的任何时间。

写入：当设置了 TXEx 标志（参见 12.3.6，“MSCAN 发送器标志寄存器 (CANTFLG)”）且在 CANTBSEL 中选择了相应发送缓冲器（参见 12.3.10，“MSCAN 发送缓冲器选择寄存器 (CANTBSEL)”）的任何时间。

12.4.6 时间标签寄存器 (TSRH - TSRL)

如果使能了 TIME 位，只要报文已经在 CAN 总线上得到确认，MSCAN 就把时间标签写入有效发送或接收缓冲器中的各自寄存器（参见 12.3.1，“MSCAN 控制寄存器 0 (CANCTL0)”）。发送时，CPU 只有在各自发送缓冲器标志空后才可以读取时间标签。

表 13-3. SPIC2 寄存器字段描述 (continued)

字段	描述
1 SPISWAI	SPI 停止在等待模式中 0 在等待模式中 SPI 时钟继续运行 1 当 MCU 进入等待模式时 SPI 时钟停止
0 SPC0	管脚控制 0 — SPC0 位选择单线双向模式。如果 MSTR = 0 (辅模式), SPI 将 MISO (SISO) 管脚用于双向 SPI 数据传输。如果 MSTR = 1 (主模式), SPI 将 MOSI (MOMI) 管脚用于双向 SPI 数据传输。当 SPC0 = 1, BIDIROE 用于使能或禁止单个双向 SPI I/O 管脚的输出驱动器。 0 SPI 为数据输入和数据输出使用独立管脚 1 SPI 配置用于单线双向运行

13.4.3 SPI 波特率寄存器 (SPIBR)

该寄存器用来为 SPI 主器件设置预分频器和波特率系数。该寄存器可以随时读取或写入。

	7	6	5	4	3	2	1	0
R	0	SPPR2	SPPR1	SPPR0		SPR2	SPR1	SPR0
W								
复位	0	0	0	0	0	0	0	0
	= 不执行或预留							

图 13-7. SPI 波特率寄存器 (SPIBR)

表 13-4. SPIBR 寄存器字段描述

字段	描述
6:4 SPPR[2:0]	SPI 波特率预分频系数 — 该 3 位字段为 SPI 波特率预分频器选择 8 个系数中的一个, 如表 13-5 所示。该预分频器的输入是总线速率时钟 (BUSCLK)。该预分频器的输出驱动 SPI 波特率系数的输入 (图 13-4)。
2:0 SPR[:0]	SPI 波特率系数 — 该 3 位字段为 SPI 波特率系数选择 8 个系数中的一个, 如表 13-6 所示。该被除数的输入来自 SPI 波特率预分频器 (见图 13-4)。该被除数的输出是主模式的 SPI 波特率时钟。

表 13-5. SPI Baud 波特率预分频器系数

SPPR2:SPPR1:SPPR0	预分频器系数
0:0:0	1
0:0:1	2
0:1:0	3
0:1:1	4
1:0:0	5
1:0:1	6
1:1:0	7
1:1:1	8

15.4.1 操作实例

这一部分显示了计数器达到模数寄存器中的匹配值时 RTC 的运行情况。

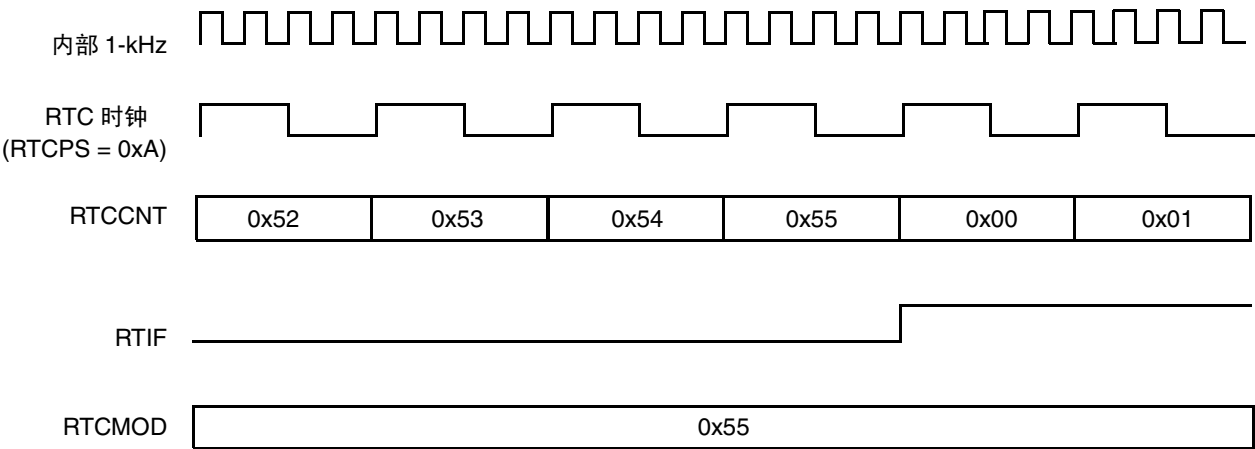


图 15-6. RTC 计数器溢出实例

在图 15-6 所示的例子中，所选时钟源为内部时钟源。预分频器 (RTCPs) 设置为 0xA 或 divide-by-4。RTCMOD 寄存器中的模数值设置为 0x55。当计数器 RTCCNT 达到模数值 0x55 时，计数器溢出为 0x00 并继续计数。实时中断标记 RTIF 会在计数器值从 0x55 变为 0x00 时设置。如果 RTIE = 1，RTIE 被设置时会生成一个实时中断。

15.5 初始化 / 应用信息

本小节提供示例代码，为用户提供如何初始化和配置 RTC 模块方面的一些基本指导。示例软件采用 C 语言实施。

下面的示例介绍了如何通过使用 1-kHz 时钟源的 RTC 实现时间计时，以实现尽可能低的功耗。由于 1-kHz 时钟源不如晶振准确，我们可以添加软件来进行必要的调整。为了在没有以额外的功耗为代价进行调整的情况下实现精确性，可选择具有适当预分频器和模数值的外部时钟 (ERCLK) 或内部时钟 (IRCLK)。

```
/* Initialize the elapsed time counters */
Seconds = 0;
Minutes = 0;
Hours = 0;
Days=0;

/* Configure RTC to interrupt every 1 second from 1-kHz clock source */
RTCMOD.byte = 0x00;
RTCSC.byte = 0x1F;
```

16.3 寄存器定义

本小节包括按地址顺序排列的寄存器描述。

16.3.1 TPM 状态和控制寄存器（TPMxSC）

TPMxSC 包含用于配置中断使能、TPM 配置、时钟源和预分频器等因素的溢出状态标志和控制位。这些控制与本定时器模块中的所有通道相关。

	7	6	5	4	3	2	1	0
R	TOF	TOIE	CPWMS	CLKSB	CLKSA	PS2	PS1	PS0
W	0							
复位	0	0	0	0	0	0	0	0

图 16-7. TPM 状态和控制寄存器（TPMxSC）

表 16-2. TPMxSC 字段描述

字段	描述
7 TOF	定时器溢出标志。这个读 / 写标记在 TPM 定时器达到 TPM 计数器模数寄存器中设置的模数值后复位为 0x0000 时被设置。设置了 TOF 的情况下，读取 TPM 状态和控制寄存器，然后将逻辑 0 写入 TOF 可清除 TOF。如果清除序列完成前发生另一个 TPM 溢出，则序列被复位，因此为较早 TOF 完成清除序列后 TOF 仍将保持设置状态。这样做的目的是确保 TOF 中断请求在为前一个 TOF 完成清除序列期间不会丢失。复位可清除 TOF。向 TOF 写入逻辑数 1 是无效的 t。 0 TPM 计数器未达到模数值或未溢出 1 TPM 计数器已溢出。
6 TOIE	定时器溢出中断使能。这个读 / 写位使能 TPM 溢出中断。如果 TOIE 被设置，那么在 TOF 等于 1 时会生成中断。复位可清除 TOIE。 0 TOF 中断关闭（用于软件轮询） 1 TOF 中断允许
5 CPWMS	中央对齐 PWM 选择。如果存在，这个读 / 写位选择 CPWM 运行模式。默认情况下，TPM 在执行输入捕捉、输出比较和边缘对齐 PWM 功能时以向上计数模式运行。设置 CPWMS 可重新配置 TPM，以便在执行 CPWM 功能时以向上 / 向下计数模式运行。复位可清除 CPWMS。 0 所有通道以输入捕捉、输出比较或边缘对齐 PWM 模式运行，即按每个通道的状态和控制寄存器中 MSnB:MSnA 控制位所选的模式运行。 1 所有通道以中央对齐 PWM 模式运行。
4-3 CLKS[B:A]	时钟源选择。表 16-3 所示，这个 2 位字段用于关闭 TPM 系统或选择三个时钟源之一来驱动计数器预分频器。固定系统时钟源仅在采用基于 PLL 的系统时钟的系统中有意义。没有 PLL 时，固定系统时钟源与总线速率时钟相同。TPM 模块使外部源与总线时钟保持同步，而片上同步电路使固定系统时钟源（PLL 存在时）与总线时钟保持同步。当 PLL 存在但未使能时，固定系统时钟源与总线速率时钟相同。
2-0 PS[2:0]	预分频器因子选择。这个 3 位字段表 16-4 所示为 TPM 时钟输入选择 8 个系数之一。任何时钟源同步或时钟源选择后，这个预分频器被定位，以便影响所选的驱动 TPM 系统的时钟源。新值被更新到寄存器位上之后，这个新的预分频器因子将在下一个系统时钟周期内影响时钟源。

表 16-7. TPM 时钟源选择

CLKSB:CLKSA	预分频器输入的 TPM 时钟源
00	没有选择时钟 (TPM 计数器被关闭)
01	总线速率时钟
10	固定系统时钟
11	外部源

总线速率时钟是 MCU 的主系统总线时钟。这个时钟源不要求同步，因为它是用于所有内部 MCU 活动（包括 CPU 和总线运行）的时钟。

在没有 PLL 或没有使用 PLL 的 MCU 中，固定系统时钟源与总线速率时钟源相同，不需要经过同步器。当存在并使用了 PLL 时，在晶振 2 分时钟源和定时器计数器之间要求同步器，以确保计数器过渡与总线时钟过渡同步。同步器将用于芯片一级，以便将晶振相关源时钟与总线时钟同步。

外部时钟源可与任何 TPM 通道管脚连接。这种时钟源必须始终通过同步器，确保计数器过渡与总线时钟过渡保持同步。总线速率时钟驱动同步器；因此，要满足 Nyquist 标准甚至抖动，外部时钟源的频率必须不能高于总线速率的四分之一。使用适当时钟时，外部时钟可与总线时钟的四分之一一样快。

当外部时钟源共享 TPM 通道管脚时，该管脚不应用于其他通道计数功能。例如，当 TPM 通道 0 管脚同时也用作定时器外部时钟源时，将通道 0 配置用于输入捕捉会造成不明确。（用户应负责避免这种设置。）TPM 通道仍可用于输出比较模式以支持软件计时功能（管脚控制位不影响 TPM 通道管脚）。

16.4.1.2 计数器溢出和模数复位

中断标记和使能与 16 位主计数器相关。标记 (TOF) 是显示定时器计数器溢出的软件可接入指标。不论何时 TOF 标记等于 1，使能信号都在软件轮询 (TOIE=0)（无硬件中断被生成）或中断驱动操作 (TOIE=1)（生成静态硬件中断）间选择。

导致 TOF 被设置的条件取决于 TPM i 是否被配置为中央对齐 PWM (CPWMS=1)。在最简单的模式下没有模数限制，TPM 也不在 CPWMS=1 模式中。在这种情况下，16 位定时器计数器从 0x0000 计数到 0xFFFF，然后在下一个计数时钟周期内溢出为 0x0000。在从 0xFFFF 过渡到 0x0000 时，TOF 被设置。设置了模数限制时，TOF 在从模数寄存器中设置的值向 0x0000 过渡时设置。当 TPM 处于中央对齐 PWM 模式时 (CPWMS=1)，TOF 标记在计数器到达模数寄存器中设置的计数值结束改变方向时被设置（也就是说从模数寄存器中设置的值向下一个更低计数值过渡时）。这与 PWM 周期结束对应（0x0000 计数值与周期中央对应）。

16.4.1.3 计数模式

主定时器的计数器有两种计数模式。选择中央对齐 PWM 时，计数器以向上 / 向下计数模式运行。否则，计数器作为简单的向上计数器运行。作为向上计数器时，定时器计数器从 0x0000 计数到其终端计数，然后从 0x0000 重新开始。终端计数为 0xFFFF 或 TPMxMODH:TPMxMODL 中的模数值。

17.3.5 触发模式

触发模式控制调试器运行的整体行为。DBGT 寄存器中的 4- 位 TRG 字段选择九个触发模块中的一个。当 DBGT 寄存器中的 TRGSEL = 1, 比较器的输出必须在触发 FIFO 操作前通过操作码跟踪电路传播。DBGT 中的 BEGIN 位选择当检测到合格的触发时 FIFO 是否开始存储数据（开始跟踪），或 FIFO 从其打开之时开始循环存储数据，直到检测到合格的触发（结束触发）。

将 1 写入到寄存器中的 ARM 位便可启动调试运行，它设置 DBGS 中的 ARMF 标记，并清除 AF 和 BF 标记及 CNT 位。开始跟踪调试运行当 FIFO 满时结束。结束跟踪运行则在所选触发事件发生时结束。任何调试运行均可通过将 0 写入到 DBGC 中的 ARM 或 DBGEN 位停止。

除纯事件模式外的所有触发模式中，FIFO 都存储流变化地址。在纯事件触发模式中，FIFO 将数据存储在 FIFO 的低八位。

控制位在纯事件触发模式中被忽略，而且所有这样的调试运行都是开始类型跟踪。当 TRGSEL = 1 选择操作码获取触发器，没有必要在比较中使用 R/W，因为操作码标签只应用于操作码获取，而这一直都是读周期。在采用全模式触发器时，规定 TRGSEL = 1 也是不正常的，因为操作码的值通常在特定的地址可以知道。

下面的触发模式描述只说明了导致触发的主要比较器条件。比较器 A 或 B 通常都可以被 R/W 进一步鉴定，通过将 RWAEN (RWBEN) 和相应的 RWA (RWB) 值设置为与 R/W 相匹配。如果 BRKEN = 1, 来自比较器的带可选 R/W 鉴定的信号，用来请求 CPU 断点，TAG 决定 CPU 请求是标记请求还是强制请求。

只 A — 当地址匹配比较器 A 的值时触发

A 或 B — 当地址匹配比较器 A 或 B 的值时触发

A 然后 B — 当地址匹配比较器 B 但只能在另一个周期的地址匹配比较器 A 的值以后，触发。可能在 A 匹配后 B 匹配前有许多周期。

A 和 B 数据（全模式）— 这称为全模式，因为地址，数据和 R/W（可选）必须在同一个总线周期内匹配，才能产生触发事件。比较器 A 检查地址，比较器的低阶字节检查数据，如果 RWAEN = 1, R/W 对照 RWA 进行检查。比较器 B 的高半部分没有使用。

在全触发模式中，规定标签类 CPU 断点 (BRKEN = TAG = 1) 没有用，但是如果你这样做了，就会忽略比较器 B 数据匹配，以例向 CPU 发送标签请求，当比较器 A 地址匹配时发送 CPU 断点。

A 但非 B 数据（全模式）— 地址必须匹配比较器 A, 数据必须不能匹配比较器 B 的低阶部分，如果 RWAEN = 1, R/W 必须匹配 RWA。所有三个条件必须在同一个总线周期中达到才能引起触发。

在全触发模式中，规定标签类 CPU 断点 (BRKEN = TAG = 1) 没有用，但是如果你这样做了，就会忽略比较器 B 数据匹配，以例向 CPU 发送标签请求，当比较器 A 地址匹配时发送 CPU 断点。

纯事件 B（存储数据）— 当地址每次匹配比较器 B 的值时，触发事件发生。触发事件导致数据被捕获到 FIFO 中。当 FIFO 满时调试运行结束。

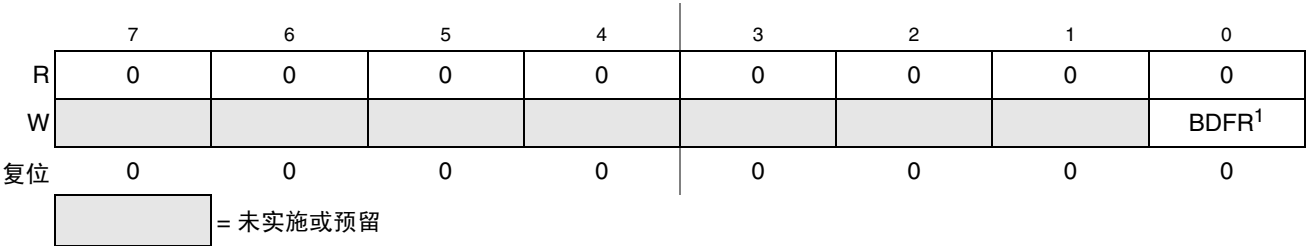
A 然后纯事件 B（存储数据）— 当地址匹配比较器 A 中的值后，每次地址匹配比较器 B 中的值时，触发事件发生。触发事件导致数据被捕获到 FIFO 中。当 FIFO 满时调试运行结束。

17.4.1.2 BDC 断点匹配寄存器 (BDCBKPT)

6- 位 寄存器保留 BDC 中的硬件断点的地址。BDCSCR 中的 BKPTEN 和 FTS 控制位用来使能和配置断点逻辑。专门的串行 BDC 命令 (READ_BKPT 和 WRITE_BKPT) 用来读和写 BDCBKPT 寄存器，但是用户程序不能存取它，因为它不位于 MCU 的普通存储器映射空间中。当目标 MCU 处于激活背景调试模式时，断点一般在运行用户应用程序前设置。关于建立和使用 BDC 中的硬件断点逻辑的更多信息，请参见 17.2.4，“BDC 硬件断点”。

17.4.2 系统背景调试强制复位寄存器 (SBDFR)

这个寄存器包含单个只写控制位。必须要用一个串行后台模式命令，如 WRITE_BYTE，来写 SBDFR。从用户程序写该寄存器的尝试被忽略。读总是返回 0x00。



¹ BDFR 只有通过串行后台模式调试命令才可写，不能通过用户程序来写。

图 17-6. 系统背景调试强制复位寄存器 (SBDFR)

表 17-3. 寄存器字段描述

字段	描述
0 BDFR	背景调试强制复位 — 一系列激活后台模式命令，如 WRITE_BYTE 等，允许外部调试主机强制目标系统复位。将 1 写到这个位，强制 MCU 复位。这个位 不能从用户程序写。

17.4.3 DBG 寄存器和控制位

这个调试模块包括 9 个字节的寄存器空间，用于三个 16- 位寄存器和三个 8- 位控制和状态寄存器。这些寄存器位于存储器空间的高地址空间中，这样它们可以存取正常的应用程序。普通用户应用程序几乎从不接入这些寄存器，除了使用断点逻辑的 ROM patching 机制。

17.4.3.1 调试比较器 A 高寄存器 (DBGCAH)

这个寄存器包含比较器 A 的高 8 位的比较值位。在复位时，这个寄存器被强制设置为 0x00，可以随时被读或写，除非 ARM = 1。

17.4.3.2 调试比较器 A 低寄存器 (DBGCAL)

这个寄存器包含比较器 A 的低 8 位的比较值位。在复位时，这个寄存器被强制设置为 0x00，可以随时被读或写，除非 ARM = 1。

附录 A

电气特征

A.1 简介

本小节包含本文档出版时 MC9S08DZ60 系列微控制器最精确的电气和时序信息。

A.2 参数分类

本附件中显示的电气参数通过各种方法得到了保证。为了让客户更好地理解，我们进行了如下分类，并在合适的地方对表中的参数进行了相应地标注：

表 A-1. 参数分类

P	在各个器件上进行生产测试时这些参数都得到了保证。
C	这些参数通过设计特性表征来实现的，方法是从统计上测量整个工艺参数变化的相关样本尺寸。
T	这些参数通过对典型条件下的典型设备（除非另有说明）进行小规模采样进行设计特性表征而获得。典型列中显示的所有值均属本类别。
D	这些参数主要源自于仿真。

注意

参数表“C”栏中显示了相应的分类。

A.3 绝对最大额定值

绝对最大额定值仅仅强调额定值，不保证最大值的功能操作。超过表 A-2 中指定限度的压力可能会影响设备可靠性或造成设备的永久损坏。有关功能操作条件的信息，请参考本小节中的其它表格。

本芯片含有防高静电压或电场的电路，保护设备免遭损坏。然而，我们还是建议采取一些正常的防范措施，以避免任何高于最大额定电压的电压应用到该高阻电路。如果未使用的输入固定到一个适当的逻辑电压水平（例如， V_{SS} 或 V_{DD} ）。

表 A-6. DC 特性 (续)

编号	C	特性	符号	条件	最小值	典型值 ¹	最大值	单位
10	P	Hi-Z (关态) 漏电流 (每管脚) 所有输入 / 输出	I_{OZ}	$V_{IN} = V_{DD} \text{ or } V_{SS}$	—	0.1	1	μA
11	P	上拉电阻 (或下拉电阻 ² , 如果启用的话)	R_{PU}, R_{PD}	5 V	20	45	65	$k\Omega$
	C			3 V	20	45	65	
12	T	输入电容、所有管脚	C_{In}		—	—	8	pF
13	D	RAM 保持电压	V_{RAM}		0.9	1.4	2.0	V
14	D	POR re-arm 电压 ³	V_{POR}		0.9	1.4	2.0	V
15	D	POR re-arm 时间 ⁴	t_{POR}		10	—	—	μs
16	P	低压探测阈值 — 高量程 V_{DD} 下降 V_{DD} 上升	V_{LVD1}		3.9 4.0	4.0 4.1	4.1 4.2	V
17	P	低压探测阈值 — 低量程 V_{DD} 下降 V_{DD} 上升	V_{LVD0}		2.48 2.54	2.56 2.62	2.64 2.70	V
18	C	低压报警阈值 — 高量程 1 V_{DD} 下降 V_{DD} 上升	V_{LVW3}		4.5 4.6	4.6 4.7	4.7 4.8	V
19	P	低压报警阈值 — 高量程 0 V_{DD} 下降 V_{DD} 上升	V_{LVW2}		4.2 4.3	4.3 4.4	4.4 4.5	V
20	P	低压报警阈值 — 低量程 1 V_{DD} 下降 V_{DD} 上升	V_{LVW1}		2.84 2.90	2.92 2.98	3.00 3.06	V
21	C	低压报警阈值 — 低量程 0 V_{DD} 下降 V_{DD} 上升	V_{LVW0}		2.66 2.72	2.74 2.80	2.82 2.88	V
22	T	低压禁止复位 / 恢复滞后	V_{hys}	5 V	—	100	—	mV
				3 V	—	60	—	
23	D	dc 注入电流 ^{5, 6, 7, 8} 单管脚限制 总 MCU 限制, 包括所有加应用的管脚	I_{IC}	$V_{IN} > V_{DD}$	0	—	2	mA
				$V_{IN} < V_{SS}$	0	—	-0.2	
				$V_{IN} > V_{DD}$	0	—	25	
				$V_{IN} < V_{SS}$	0	—	-5	