



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	Active
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	53
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 24x12b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	64-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/pro/item?MUrl=&PartUrl=mc9s08dz48clh

4.5.8 向量重定向

在 Flash 受到块保护的同时，复位和中断向量也将受到有效保护。向量重定向使用户可以修改中断向量信息而不必影响对 bootloader 和复位向量空间的保护。

。要使重定向正常进行，至少一部分 Flash 必须通过对地址 0xFFBD 上的 NVPROT 寄存器进行编程来进行块保护。所有中断向量（内存位置为 0xFFC0–0xFFFFD）都被重定向，虽然复位向量（0xFFFFE:0xFFFFF）没有被重定向。

例如，如果 Flash 的 1536 个字节受到保护，那么受保护的地址区域为从 0xFA00 到 0xFFFF。中断向量 (0xFFC0–0xFFFFD) 被重定向到位置 0xF9C0–0xF9FD。如果启用了向量重定向并发生了中断，那么 0xF9E0:0xF9E1 上的值被用于向量而不是 0xFFE0:0xFFE1 上的值。这样，用户就可以利用新的程序代码对 Flash 的未保护部分进行重新编程，增加新的中断向量值，同时保留受保护的区域（这包括不发生任何变化的缺省向量）。

4.5.9 安全性

MC9S08DZ60 系列包含用于防止非法访问 Flash、EEPROM 和 RAM 存储器内容的电路。启用了安全性功能后，Flash、EEPROM 和 RAM 被看作是安全的资源。直接页面寄存器、高端页面寄存器和后台调试控制器被看作是不安全的资源。安全存储器中执行的程序可以正常访问 MCU 位置和资源。通过不安全存储器空间内运行的程序或通过后台调试接口访问安全存储器位置的任何尝试都将被阻止（写入操作被忽略，而读取操作则全部返回 0）。

安全性的启用和关闭由 FOPT 寄存器中的两个寄存器位 (SEC[1:0]) 的状态确定。在复位过程中，非易失性位置 NVOPT 的内容从 Flash 中拷贝到高端页面寄存器空间内的工作 FOPT 寄存器上。用户可以通过编程 NVOPT 位置来启用安全性。这一操作可以在对 Flash 进行编程的同时进行。1:0 状态会关闭安全性；而另外 3 种组合会启用安全性。请注意，擦除状态 (1:1) 会使 MCU 处于安全状态。在开发过程中，不管 Flash 什么时候被擦除，立即将 NVOPT 中的 SEC0 设置为 0 以便使 SEC = 1:0 是一种有效的方法。这将使 MCU 在后来的复位完成后继续处于不安全状态。

MCU 处于安全状态时不能启用片上调试模块。您可以为后台存储器访问命令使用独立的后台调试控制器，但 MCU 不能进入主动后台模式，除非您在复位的上升边将 BKGD 保持在较低位置。

用户可以选择通过一个 8 字节后门安全密钥来设置允许或不允许安全解锁机制。如果 NVOPT/FOPT 中的非易失性 KEYEN 位为 0，那么说明后门密钥被禁用，在没有完全擦除所有 Flash 位置的情况下不能启用安全性。如果 KEYEN 为 1，那么安全的用户程序可以通过以下方式暂时关闭安全性：

1. 向 FCNFG 寄存器中的 KEYACC 写入 1。这将使 Flash 模块将后门对比密钥位置 (NVBACKKEY through NVBACKKEY+7) 的写入解释为将与密钥进行对比的值而不是 Flash 编程或擦除命令的第一步。
2. 将用户输入的密钥值写入到位置 NVBACKKEY 到 NVBACKKEY+7 上。这些写入操作必须按顺序进行，以 NVBACKKEY 值开始，以 NVBACKKEY+7 值结束。这些写入操作中必须使用 STHX，因为这些写入不能在相邻的总线循环上完成。用户软件一般通过通信接口（如串行 I/O）从 MCU 系统外部获取密钥代码。
3. 向 FCNFG 寄存器中的 KEYACC 写入 0。如果写入的 8 字节密钥与 Flash 位置上保存的密钥相匹配，那么 SEC 位被自动修改为 1:0，同时安全性将关闭，直到下一次复位。

表 7-2. 指令集小结 (第 8 页, 共 9 页)

Source Form	Operation	Address Mode	Object Code	Cycles	Cyc-by-Cyc Details	Affect on CCR	
						V I 1 H	I N Z C
SUB #opr8i SUB opr8a SUB opr16a SUB oprx16,X SUB oprx8,X SUB ,X SUB oprx16,SP SUB oprx8,SP	减 A'' (A) - (M)	IMM DIR EXT IX2 IX1 IX SP2 SP1	A0 ii B0 dd C0 hh ll D0 ee ff E0 ff F0 9E D0 ee ff 9E E0 ff	2 3 4 4 3 3 5 4	pp rpp prpp prpp rpp rfp pprpp prpp	↑ 1 1 -	- ↓ ↓ ↓
SWI	软件中断 PC'' (PC) + \$0001 推 (PCL); SP'' (SP) - \$0001 推 (PCH); SP'' (SP) - \$0001 推 (X); SP'' (SP) - \$0001 推 (A); SP'' (SP) - \$0001 推 (CCR); SP'' (SP) - \$0001 I'' 1; PCH'' 中断向量高字节 PCL'' 中断向量低字节	INH	83	11	sssssvvfppp	- 1 1 -	1 - - -
TAP	将累加器转移到 CCR CCR'' (A)	INH	84	1	p	↑ 1 1 ↑	↓ ↓ ↓ ↓
TAX	将累加器转移到 X (索引寄存器低) X'' (A)	INH	97	1	p	- 1 1 -	- - - -
TPA	将 CCR 转移到累加器 A'' (CCR)	INH	85	1	p	- 1 1 -	- - - -
TST opr8a TSTA TSTX TST oprx8,X TST ,X TST oprx8,SP	负数或 0(M) 测试 (M) - \$00 (A) - \$00 (X) - \$00 (M) - \$00 (M) - \$00 (M) - \$00	DIR INH INH IX1 IX SP1	3D dd 4D 5D 6D ff 7D 9E 6D ff	4 1 1 4 3 5	rfpp p p rfpp rfp prfpp	0 1 1 -	- ↓ ↓ -
TSX	将 SP 转移到索引寄存器。 H:X'' (SP) + \$0001	INH	95	2	fp	- 1 1 -	- - - -
TXA	将 X(索引寄存器低) 转移到累加器上 A'' (X)	INH	9F	1	p	- 1 1 -	- - - -

第 8 章

多功能时钟发生器 (S08MCGV1)

8.1 介绍

多功能时钟发生器 (MCG) 模块为 MCU 提供了几个时钟源选项。MCG 模块中包含 1 个锁频环 (FLL) 和 1 个锁相环 (PLL)，可以由内部或外部参考时钟控制。模块可以选择 FLL 或 PLL 时钟作为 MCU 系统时钟，也可以选择内部或外部参考时钟作为 MCU 系统时钟。无论选择哪个时钟源，它都要通过降阶总线分频器，该分频器允许生成更低的输出时钟频率。MCG 还控制一个外部振荡器 (XOSC)，以便把晶体或共鸣器用作外部参考时钟。

MC9S08DZ60 系列的所有器件都含有 MCG 模块。

注意

如需了解整个芯片的分配时钟源的更多信息，请参见 1.3，“系统时钟分配”。

8.5.5 内部参考时钟

当设置了 IRCLKEN 时，内部参考时钟信号将作为 MCGIRCLK 出现，作为另外一个时钟源使用。通过调整内部参考时钟时段，MCGIRCLK 的目标频率可以重新设定。在 MCGTRM 寄存器的 FRIM 位中写入一个新值就可以完成该操作。向 MCGTRM 寄存器写入一个更大的值将降低 MCGIRCLK 频率，写入一个更小的值将提高 MCGIRCLK 频率。如果 MCG 处于 FLL Engaged Internal (FEI)、FLL Bypassed Internal (FBI) 或 Bypassed Low Power Internal (BLPI) 模式，TRIM 位会影响 MCGOUT 频率。TRIM 和 FTRIM 值由 POR 初始化，但不会受其他复位影响。

如果 MCGIRCLK 未调整，编程低参考分频器 (RDIV) 因子可能导致 MCGOUT 频率超过芯片级最高频率，且违反芯片级时钟定时技术规范 (参见 Device Overview)

如果 IREFSTEN 和 IRCLKEN 位均已设置，内部参考时钟将在停止模式期间保持运行，以便在退出停止模式时快速恢复。

8.5.6 外部参考时钟

MCG 模块可以支持 FEE 和 FBE 模式中频率在 31.25 kHz-5 MHz，PEE 和 PBE 模式中频率在 1 MHz -16 MHz 之间，BLPE 模式中频率在 0 - 40 MHz 之间的外部参考时钟。当设置了 ERCLKEN 时，内部参考时钟信号将作为 MCGERCLK 出现，作为另外一个时钟源使用。当 IREFS = 1 时，FLL 或 PLL 不使用外部参考时钟，外部参考时钟只能用作 MCGERCLK。在这些模式中，该频率可能等于芯片级定时规范支持的最大频率 (参见 Device Overview)

如果 EREFSTEN 和 ERCLKEN 位均已设置，或者 MCG 处于 FEE、FBE、PEE、PBE 或 BLPE 模式，外部参考时钟将在停止模式期间保持运行，以便在退出停止模式时快速恢复。

如果 CME 位写入 1，时钟监控器使能。如果外部参考降到某一频率 (根据 MCGC2 中的 RANGE 位可以是 floc_high 或 floc_low) 以下，MCU 将复位。系统复位状态 (SRS) 寄存器中的 LOC 位将用来标志错误。

8.5.7 固定频率时钟

MCG 将分频参考时钟作为 MCGFFCLK，作为另外一个时钟源使用。MCGFFCLK 频率小于或等于 MCGOUT 频率的 1/4 才有效。正是因为这个要求，MCGFFCLK 在旁路模式中的如下 BDIV 和 RDIV 值组合中无效：

- BDIV=00 (除以 1)，RDIV
- BDIV=01 (除以 2)，RDIV

8.6 初始化 / 应用报文

本节描述了如何在应用中初始化和配置的 MCG 模块。后面几节给出了几个如何对 MCG 进行初始化、如何在不同模式间进行适当切换的例子。

8.6.2.2 示例 2: 从 PEE 切换到 BLPI 模式: 外部晶体 = 4 MHz、总线频率 = 16 kHz

本例中, MCG 将通过适当的运行模式, 从晶体频率为 4MHz、总线频率为 8MHz 的 PEE 模式 (参见前一示例) 切换到总线频率为 16kHz 的 BLPI 模式。示例中首先介绍了代码序列, 然后提供了一个演示该顺序的流程图。

1. 首先, PEE 必须转换到 PBE 模式:
 - a) MCGC1 = 0x90 (%10010000)
 - CLKS (位 7 和 6) 设置为 %10, 以便把系统时钟源切换到外部参考时钟。
 - b) 循环检测, 直到 MCGSC 中 CLKST (位 3 和 2) 是 %10, 表明已经选择外部参考时钟为 MCGOUT 馈电。
2. 然后, PBE 必须要么直接转换到 FBE 模式, 要么先转换到 BLPE 模式, 然后再转换到 FBE 模式:
 - a) BLPE: 如果需要从 BLPE 模式转换, 首先把 MCGC2 中的 LP (位 3) 设置为 1。
 - b) BLPE/FBE: MCGC1 = 0xB8 (%10111000)
 - RDIV (位 5-3) 设置为 %111 或除以 128, 因为 $4 \text{ MHz} / 128 = 31.25 \text{ kHz}$, 这在 FLL 要求的 31.25 kHz -- 39.0625 kHz 频率范围内。在 BLPE 模式中, RDIV 的配置不重要, 因为 FLL 和 PLL 都被禁止。更改它们只会建立供 FLL 在 FBE 模式中使用的分频器。
 - c) BLPE/FBE: MCGC3 = 0x04 (%00000100)
 - PLLS (位 6) 清除至 0, 选择 FLL。在 BLPE 模式中, 更改该位只会让 MCG 准备在 FBE 模式中的 FLL 使用。如果 PLLS = 0, VDIV 值不重要。
 - d) BLPE: 如果通过 BLPE 模式转换, 将 MCGC2LP (位 3) 中的 LP 清除至 0, 切换到 FBE 模式。
 - e) FBE: 循环检测, 直到 MCGSC 中的 PLLST (位 5) 已经清除, 表明 PLLS 时钟的当前源是 FLL。
 - f) FBE: 循环检测, 直到在 MCGSC 中的 LOCK (位 6) 已经置位, 表明 FLL 已经获得锁定。尽管在 FBE 模式中 FLL 被旁通, 但它仍使能且在运行。
3. 接下来, FBE 模式转换到 FBI 模式:
 - a) MCGC1 = 0x44 (%01000100)
 - MCGSC1 中的 CLKS (位 7 和 6) 设置为 %01, 以便将系统时钟切换到内部参考时钟。
 - IREFS (位 2) 设置为 1, 选择内部参考时钟为参考时钟源。
 - RDIV (位 5-3) 设置为 %000 或除以 1, 因为调整后的内部参考应在 FLL 要求的 31.25 kHz--39.0625 kHz 频率范围内。
 - b) 循环检测, 直到 MCGSC 中的 IREFST (位 4) 是 1, 表明已经选择内部参考时钟为参考时钟源。
 - c) 循环检测, 直到 MCGSC 中的 CLKST (位 3 和 2) 是 %01, 表明已经选择内部参考时钟为 MCGOUT 馈电。

9.3 存储器映射 / 寄存器定义

ACMP 包括一个寄存器:

- 一个 8 位状态和控制寄存器

如需了解 ACMP 寄存器的绝对地址分配, 请参见本文档的存储器节“直接页面寄存器概述”。本节仅按寄存器和控制位的名称及相关地址偏移进行参考。

有些 MCU 的 ACMP 可能不止一个, 因此寄存器名称包括占位符 (x), 以明确正在参考哪个 ACMP。

表 9-2. ACMP 寄存器摘要

名称		7	6	5	4	3	2	1	0
ACMPxSC	R	ACME	ACBGS	ACF	ACIE	ACO	ACOPE	ACMOD	
	W								

9.3.1 ACMPx 状态和控制寄存器 (ACMPxSC)

ACMPxSC 包括状态标记和使能和配置 ACMP 所需的控制位。

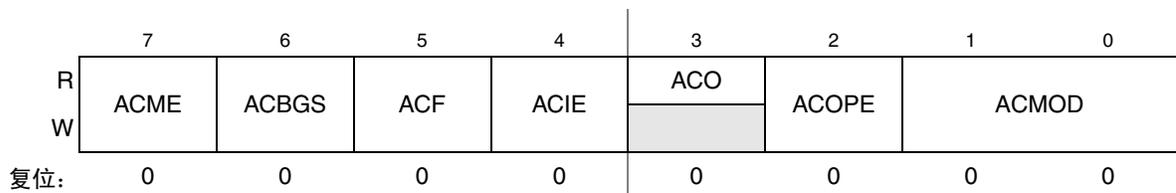


图 9-3. ACMPx 状态和控制寄存器 (ACMPxSC)

表 9-3. ACMPxSC 字段描述

字段	描述
7 ACME	模拟比较器模块使能。使能 ACMP 模块。 0 ACMP 关闭 1 ACMP 使能
6 ACBGS	模拟比较器死区选择。选择带死区参考电压或 ACMPx+ 管脚作为模拟比较器同相输入的输入。 0 外部管脚 ACMPx+ 选择为比较器的同相输入 1 内部参考选择为比较器的同相输入
5 ACF	模拟比较器标记。每次发生比较事件时都设置 ACF。比较事件由 ACMOD 定义。通过在 ACF 上写入 1 来清除 ACF。 0 未发生比较事件。 1 已发生比较事件。
4 ACIE	模拟比较器中断使能。从 ACMP 那里使能中断。设置了 ACIE 后, 在 ACF 置位时中断被触发。 0 中断禁止 1 中断使能
3 ACO	模拟比较器输出。ACO 读数返回模拟比较器输出的当前值。ACO 复位为 0, 在 ACMP 禁止时 (ACME = 0) 读数为 0。

11.5.1.1 启动信号

当总线空闲时，没有主机占用总线（SCL 和 SDA 线处于逻辑高电平），主机可以通过发送启动信号发起通信。如图 11-9 所示，启动信号定义为，当 SCL 在高电平时，SDA 从高到低的跳变。该信号表示开始新的数据传输（每次数据传输可能包含几个字节的数据），并使所有从机退出空闲状态。

11.5.1.2 从机地址发送

启动信号发出后传输的第一个字节数据是主机发送的从机地址。这是 R/W 位之前的 7 位主叫地址。R/W 位告知从机数据传输的方向。

1 = 读取传输，从机向主机传输数据

0 = 写入传输，主机向从机传输数据

当主机发送的地址与某从机地址匹配时，此从机发回应答位进行响应。通过将 SDA 第 9 个时钟拉低实现（见图 11-9）。

系统中不能有两个地址相同的从机。如果 IIC 模块是主机，它就不能发送与其自己的从机地址相同的地址。IIC 不能同时既是主机又是从机。但是，如果仲裁在寻址周期中丢失，IIC 就重新返回到从机模式并正确运行，即便它正被另一个主机寻址。

11.5.1.3 数据传输

成功实现从机寻址之前，就可以按照主叫主机发送的 $\overline{R/W}$ 位指定的方向逐字节地进行数据传输。

地址周期后的所有传输都被称为数据传输，即使它们包含从机的子地址报文。

每个数据字节的长度均为 8 位。只有当 SCL 处于低时数据才可以更改，如果 SCL 处于高位，那么它必须保持稳定，如图 11-9 所示。SCL 的一个时钟脉冲传输一个数据位，最高位被首先传输。每个数据字节后面都有一个第 9（确认）位，该位由从机发出信号，通过把 SDA 拉低到第 9 个时钟实现。总之，一个完整数据传输需要 9 个时钟脉冲。

如果从机在第 9 个位时间时未应答主机，从机必须保留 SDA 线在高电平。主机将未接收应答信号解释为不成功的数据传输。

如果主机接收器在一个数据字节传输后未应答从机发送器，从机把这种情况理解为数据传输结束，并释放 SDA 线。

对于这两种情况，数据传输都被中止，主机会进行以下两种操作之一：

- 发送停止信号，放弃总线
- 发送重复启动信号，开始新呼叫

表 12-30. IDR1 寄存器字段描述

字段	描述
7:5 ID[2:0]	标准格式标识符 — 标准格式标识符 — 该标识符由 11 个扩展格式位 (ID[10:0]) 组成。ID10 是最高位，仲裁流程期间最先在 CAN 总线上发送。标识符的优先级定义为处于最高位的最小二进制数。也可参见表 12-29 中的 ID 位。
4 RTR	远程发送请求 — 该标志反应 CAN 帧中远程发送请求的状态。在接收缓冲器中，它显示已接收帧的状态，并在软件中支持应答帧的发送。在发送缓冲器中，该标志定义将要发送的 RTR 位的设置。 0 数据帧 1 远程帧
3 IDE	ID 扩展 — 该标志显示扩展或标准标识符格式是否应用于该缓冲器。在接收缓冲器中，该标志设置为已接收，并向 CPU 显示如何处理缓冲器标识符寄存器。在发送缓冲器中，该标志向 MSCAN 显示将发送的标识符类型。 0 标准格式 (11 位) 1 扩展格式 (29 位)

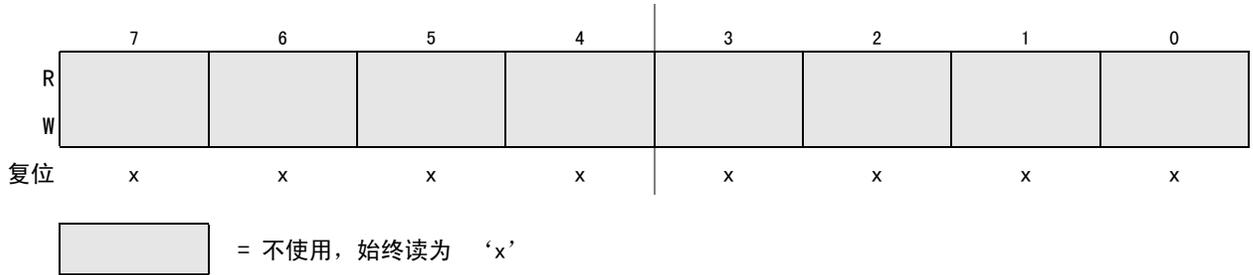


图 12-31. 标识符寄存器 2 — 标准映射

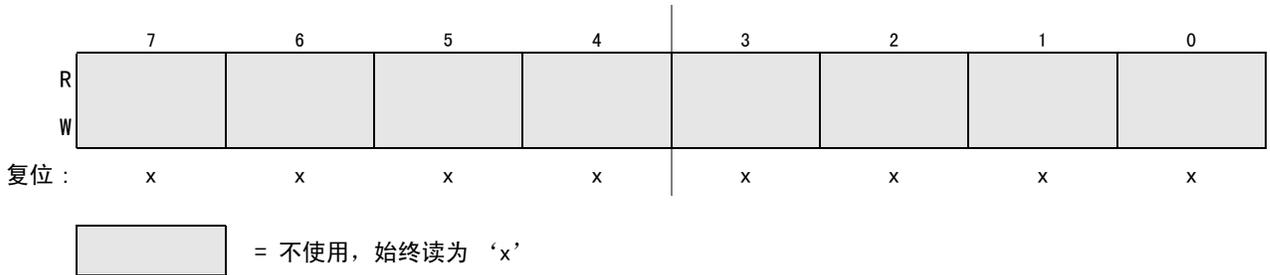


图 12-32. 标识符寄存器 3 — 标准映射

12.4.3 数据段寄存器 (DSR0-7)

8 个数据段寄存器 (每个都带有位 DB[7:0]) 包含将要发送或接收的数据。将要发送或接收的字节数由相应 DLR 寄存器中的数据长度代码决定。

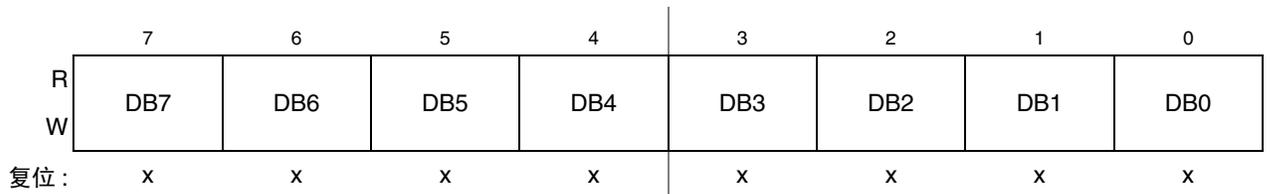


图 12-33. 数据段寄存器 (DSR0 - DSR7) — 扩展标识符映射

可编程预分频器从 CANCLK 生成时间冲量 (Tq) 时钟。时间冲量是 MSCAN 所处理时间的原子单位。

等式 12-2

$$f_{Tq} = \frac{f_{CANCLK}}{(\text{Prescaler value})}$$

位时间再分成三段，如 Bosch CAN 规范所述 (见图 12-43):

- SYNC_SEG: 该段有一个长度固定的时间冲量，信号边沿预计出现在本段。
- 时段 1: 本段包括 CAN 标准的 PROP_SEG 和 PHASE_SEG1。通过设置参数 TSEG1，使之包含 4-16 个时间冲量，可以对其进行编程。
- 时段 2: 本段表示 CAN 标准的 PHASE_SEG2。通过设置 TSEG2 参数，使之具有 2-8 个时间冲量长，可以对其进行编程。

等式 12-3

$$\text{Bit Rate} = \frac{f_{Tq}}{(\text{number of Time Quanta})}$$

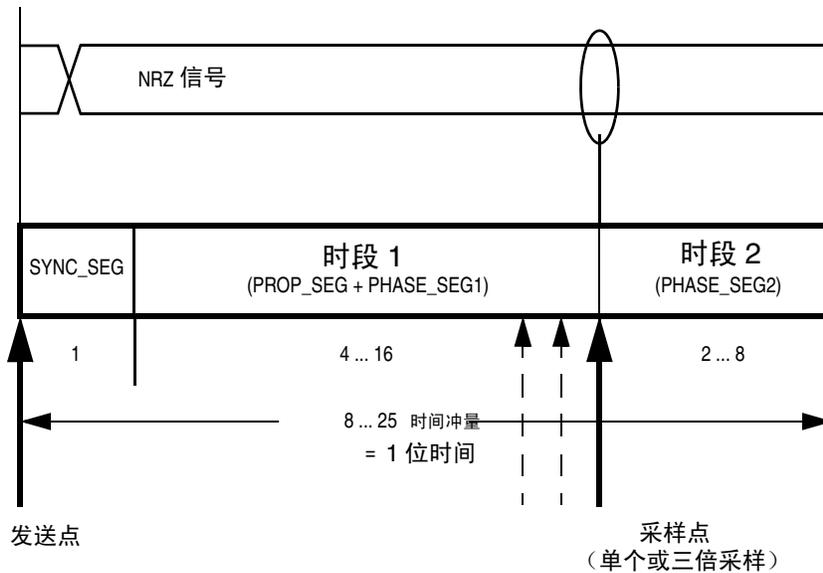


图 12-43. 位时间内的段

12.5.4.4 监听模式

在可选的 CAN 总线监控模式（监听）中，CAN 节点能够接收有效数据帧和有效远程帧，但它只发送 CAN 总线上的“隐性”位。此外，它不能启动发送。如果 MAC 层需要发送“显性”位（ACK 位、超载标志或有效错误标志），该位将在内部传输，这样 MAC 层就监控该“显性”位，此时 CAN 总线在外部仍保持隐性状态。

12.5.4.5 保密模式

MSCAN 模块没有保密功能。

12.5.4.6 环回自测模式

环回自测模式独立于外部系统的连接，有时用于检查软件，以帮助隔离系统问题。在该模式中，发送器输出内部连接到接收器输入。RXCAN 输入管脚被忽略，TXCAN 输出进入隐性状态（逻辑 1）。发送时，MSCAN 的运行如常，把它自己发送的报文作为从远程节点接收的报文。在该状态中，MSCAN 将忽略 CAN 帧应答场中 ACK 间隙中发送的位，以确保正确接受它自己的报文。同时生成发送和接收中断。

12.5.5 低功耗选项

如果 MSCAN 禁止 (CANE = 0)，MSCAN 时钟停止，以节省功率。

如果 MSCAN 使能 (CANE = 1)，MSCAN 还有两种与正常模式相比功耗更低的模式：睡眠模式和断电模式，在睡眠模式中，可以通过停止所有时钟（除了 CPU 端访问寄存器的时钟外）来降低功耗。在节电模式中，所有时钟停止，没有功率消耗。

表 12-36 总结了 MSCAN 和 CPU 模式的各种组合。模式的特殊组合通过 CSWAI 和 SLPRQ/SLPAK 位上的设置决定。

对所有模式来说，只有当 MSCAN 处于睡眠模式（SLPRQ = 1，SLPAK = 1）、唤醒功能使能（WUPE = 1）且唤醒中断使能（WUPIE = 1）时，MSCAN 唤醒中断才可能发生。

只有当出现以下情形时，MSCAN 才能够退出睡眠模式（唤醒）：

- 出现 CAN 总线有效和 $\overline{\text{WUPE}} = 1$
或
- CPU 清除 SLPRQ 位

注意

在使能睡眠模式（ $\text{SLPRQ} = 1$ ， $\text{SLPAK} = 1$ ）前，CPU 不能清除 SLPRQ 位。

唤醒之后，MSCAN 等待 11 个连续隐性位与 CAN 总线同步。因此，如果 MSCAN 被 CAN 帧唤醒，就不会收到该帧。

如果在进入睡眠模式前已经收到报文，接收报文缓冲器（RxFG 和 RxBG）就包含该报文。所有挂起操作在唤醒后执行，复制 RxBG 至 RxFG，报文中止和报文发送。如果在退出睡眠模式后 MSCAN 仍处于总线脱离状态，它将继续计数 128 次 11 个连续隐性位的出现。

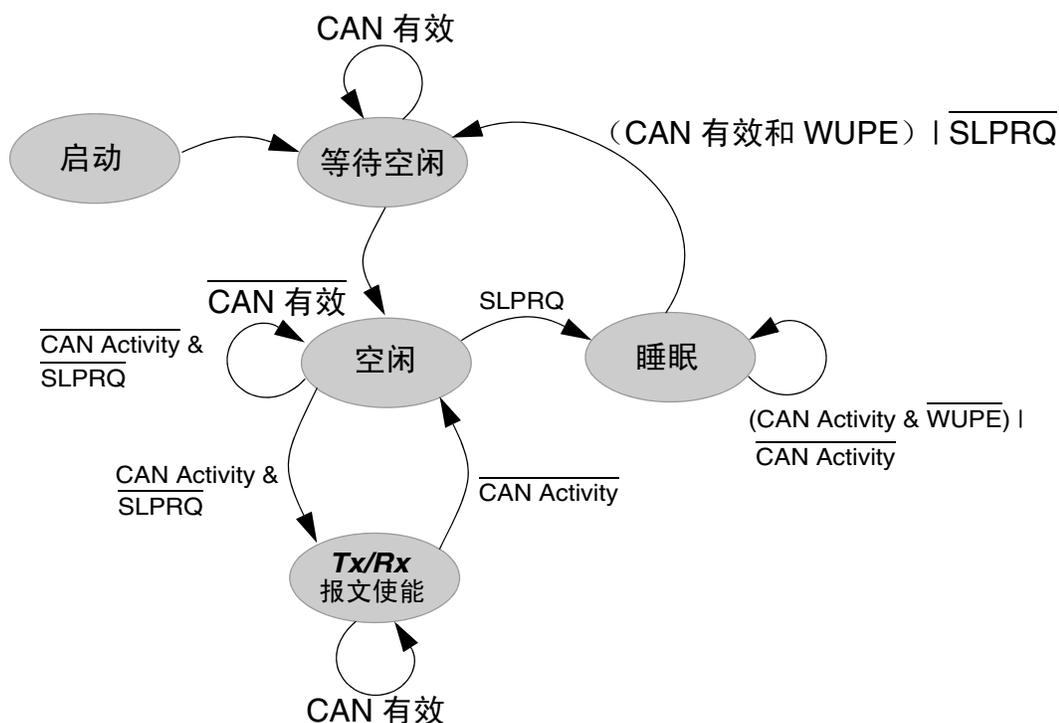


图 12-45. 进入 / 退出睡眠模式的简单状态 转换

13.4.5 SPI 数据寄存器 (SPID)

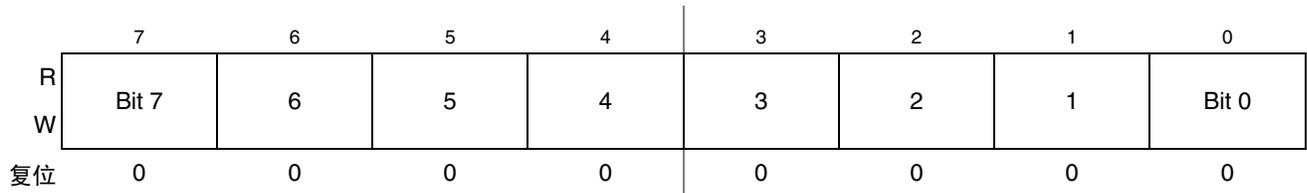


图 13-9. SPI 数据寄存器 (SPID)

读该寄存器会返回从接收数据缓冲器读取的数据。写该寄存器会把数据写入发送数据缓冲器。当 SPI 配置为主 SPI 时，向发送数据缓冲器写入数据将发起一个 SPI 传输。

数据不应写入发送数据缓冲器，除非设置了 SPI 发送缓冲器空标记（SPTEF），显示发送缓冲器中有一定的空间来排队新的发送字节。

在设置了 SPRF 后、完成另外一个传输前，可以随时从 SPID 中读取数据。如果在新传输结束前未能从接收数据缓冲器读取数据，就会导致接收溢出，新传输的数据丢失。

13.5 功能描述

检查 SPI 发送缓冲器空标记（SPTEF = 1）然后将数据字节写入主 SPI 器件中的 SPI 数据寄存器，这样就发起 SPI 传输。当 SPI 移位寄存器可用时，该数据字节从发送数据缓冲器移到移位器，设置 SPTEF，显示缓冲器中有一定的空间来排队另外一个发送字符（如需要的话），SPI 串行传输开始。

在 SPI 传输过程中，数据在一个 SPSCCK 边沿从 MOSI 管脚上进行采样（读取）和转移，半个 SPSCCK 周期后改变 MOSI 管脚上的位值。在 8 个 SPSCCK 周期后，主 SPI 的移位寄存器中的数据已经从 MOSI 管脚转移到从 SPI，同时 MISO 管脚里的 8 位数据被转移到主 SPI 的移位寄存器中。传输结束时，收到的数据字节从移位器转移到接收数据缓冲器，并设置 SPRF，显示数据可以通过读取 SPID 进行读取。如果传输结束时发送缓冲器中有另外一个数据字节在等待，它就被移到移位器，设置 SPTEF，启动新传输。

在正常情况下，SPI 数据首先传输最高位（MSB）。如果设置了最低位先发使能位（LSBFE），SPI 数据首先移位 LSB。

当 SPI 配置为从 SPI 时，传输开始前其 SS 管脚必须驱动到低态，整个传输过程 SS 必须保持低态。如果选择了 CPHA = 0 的时钟格式，SS 必须在两次连续传输间被驱动到逻辑 1。如果 CPHA = 1，SS 在两次连续传输间必须保持低态。如需了解更多详细信息，参见 13.5.1，“SPI 时钟格式”。

因为发送器和接收器被双缓冲，因此第二个字节（除了当前正在移出的字节外）可以排队进入发送数据缓冲器，原来接收的字符可以放在接收数据缓冲器中，同时新字符被转移进来。SPTEF 标记显示发送缓冲器何时有新字符空间。SPRF 标记显示已接收的字符何时出现在接收数据缓冲器。在下一次传输完成或导致接收溢出错误前，已接收字符必须从接收缓冲器读出（读 SPID）。

当出现接收溢出时，新数据丢失，因为接收缓冲器仍留有以前的字符，不准备接受新数据。这种溢出没有任何显示，因此应用系统设计人员必须确保在发起新传输前以前的数据已经从接收缓冲器中读出。

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。

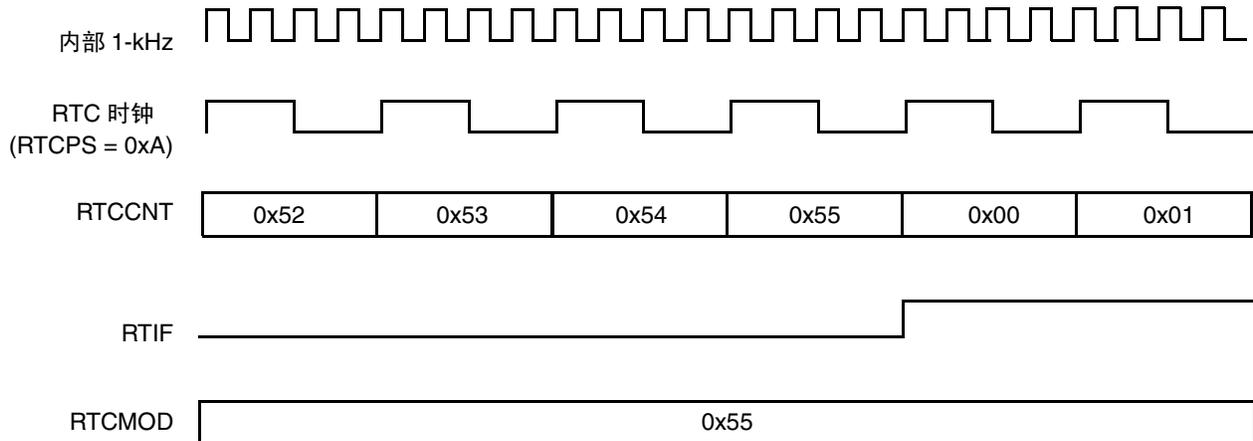


图 15-6. RTC 计数器溢出实例

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;

```

```
/******  
Function Name : RTC_ISR  
Notes : Interrupt service routine for RTC module.  
*****/  
#pragma TRAP_PROC  
void RTC_ISR(void)  
{  
    /* Clear the interrupt flag */  
    RTCSC.byte = RTCSC.byte | 0x80;  
    /* RTC interrupts every 1 Second */  
    Seconds++;  
    /* 60 seconds in a minute */  
    if (Seconds > 59){  
        Minutes++;  
        Seconds = 0;  
    }  
    /* 60 minutes in an hour */  
    if (Minutes > 59){  
        Hours++;  
        Minutes = 0;  
    }  
    /* 24 hours in a day */  
    if (Hours > 23){  
        Days ++;  
        Hours = 0;  
    }  
}
```

第 16 章

定时器脉冲宽度调节器 (S08TPMV3)

注意

本章的描述适用于 S08TPM 版本 3，该版本适用于 0M74K 及本器件的更新掩码集。3M05C 和更早版本的掩码集器件使用 S08TPM 版本 2。如果您的器件使用掩码 3M05C 或更早版本，请参考 364 页上的附录 B，“定时器脉宽调制器 (TPMV2)”，以了解该模块的相关信息。

16.1 简介

TPM 为每个通道使用一个输入 / 输出 (I/O) 管脚，即 TPMxCHn，其中 x 为 TPM 编号（如 1 或 2）而 n 为通道编号（如 0–5）。TPM 与通用 I/O 端口管脚共享其 I/O 管脚（请参考 Pins and Connections 章节了解更多信息）。

C9S08DZ60 系列 MCU 有两种 TPM 模块。在所有封装中，TPM2 有 2 个通道。TPM1 中外部管脚上的可用通道数取决于封装：

- 64 管脚和 48 管脚封装中有 6 个通道
- 32 管脚封装中有 4 个通道。

当 TPM 被配置用于中央对齐 PWM (and ELSnB:ELSnA not = 0:0) 时, 该 TPM 中所有通道的数据方向被改变; TPMxCHn 管脚被强制用作受 TPM 控制的输出, 而 ELSnA 位控制每个 TPMxCHn 输出的极性。如果 ELSnB:ELSnA=1:0, 那么在定时器计数器向上计数、通道值寄存器与定时器计数器匹配时, 相应的 TPMxCHn 管脚被清除; 当定时器计数器向下计数、通道值寄存器与定时器的计数器匹配时, TPMxCHn 管脚被设置。如果 ELSnA=1, 在定时器计数器向上计数、通道值寄存器与定时器的计数器匹配时, 相应的 TPMxCHn 管脚被设置; 当定时器计数器向下计数、通道值寄存器与定时器计数器匹配时, TPMxCHn 管脚被清除。

TPMxMODH:TPMxMODL = 0x0008
 TPMxCnVH:TPMxCnVL = 0x0005

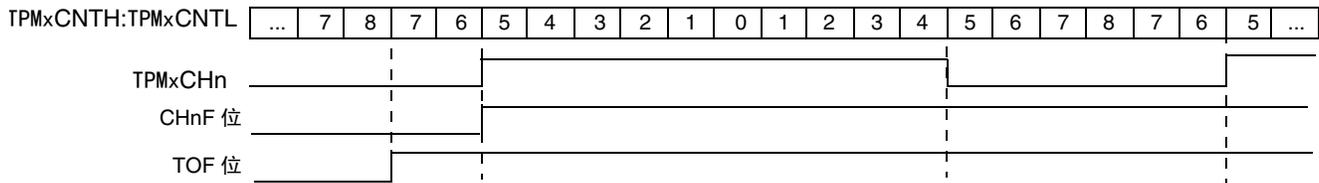


图 16-5. 中央对齐 PWM 的 High-True 脉冲

TPMxMODH:TPMxMODL = 0x0008
 TPMxCnVH:TPMxCnVL = 0x0005

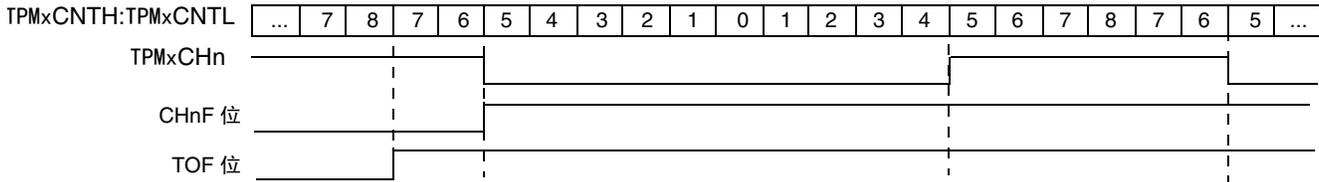


图 16-6. 中央对齐 PWM 的 Low-True 脉冲

TPM 中断标志通过两步清除：一是标志位被设置（为 1）时读取，二是向其中写入 0。如果这两步之间检测到新事件，序列被复位，并且在完成第二步后中断标志仍被设置以避免丢失新事件。

16.6.2.1 定时器溢出中断 (TOF) 介绍

根据 TPM 系统的运行模式（通用定时功能和中央对齐 PWM 操作），TOF 中断操作的意义和细节会有细微变化。标志通过上述两步序列清除。

16.6.2.1.1 常见情况

当定时器计数器从 0xFFFF 变为 0x0000 时，TOF 通常被设置。当 TPM 没有被配置为中央对齐 PWM 时 (CPWMS=0)，TOF 在定时器计数器从终端计数（模数寄存器中的值）变为 0x0000 时被设置。这种情况与计数器溢出的正常意义相对应。

16.6.2.1.2 中央对齐 PWM 情况

当 CPWMS=1 时，TOF 在定时器计数器方向在终端计数（模数寄存器中的值）结束时从向上计数变为向下计数时被设置。在这种情况下，TOF 与 PWM 周期结束相对应。

16.6.2.2 通道事件中断描述

通道中断的意义取决于通道的当前模式（输入捕捉、输出比较、边缘对齐 PWM 或中央对齐 PWM）。

16.6.2.2.1 输入捕捉事件

当通道被配置为输入捕捉通道时，ELSnB:ELSnA 控制位选择无边沿（关）、上升边沿、下降边沿或任何边沿作为触发输入捕捉事件的边沿。检测到选定边沿时，中断标志被设置。标志通过 16.6.2，“中断操作描述”中所述的两步序列清除。

16.6.2.2.2 输出比较事件

如果通道被配置为输出比较通道，中断标志在每次主定时器计数器与通道值寄存器中的 16 位值匹配时被设置。标志通过 16.6.2，“中断操作描述”中所述的两步序列清除。

16.6.2.2.3 PWM 占空比结束事件

对于配置用于 PWM 操作的通道，有两种可能性。当通道配置用于边缘对齐 PWM 时，通道标志在定时器计数器与标志占空比周期结束的通道值寄存器匹配时被设置。当通道配置为中央对齐 PWM 时，定时器计数在每个 PWM 周期内两次与通道值寄存器相匹配。在这种 CPWM 情况中，通道标志在占空比周期时间开始和结束时（定时器计数器与通道值寄存器匹配时）被设置。标志通过 16.6.2，“中断操作描述”中所述的两步序列清除。

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} ）。

中断标记和启动与 16 位主计数器相关。定时器溢出标记 (TOF) 是一种显示定时器计数器溢出的软件可接入指示。TOF 标记等于 1 时自动生成静态硬件中断的情况下, 启动信号都在软件轮询 (TOIE=0) (无硬件中断生成) 或中断驱动操作 (TOIE=1) 之间选择。

导致 TOF 被设置的条件取决于 (向上或向上 / 向下) 计数模式。在向上计数模式中, 16 位主计数器从 0x0000 计数到 0xFFFF, 并且在下一次计数时钟中溢出到 0x0000。在从 0xFFFF 向 0x0000 过渡时, TOF 被设置。设置了模数限制时, TOF 会在从模数寄存器中设置的值向 0x0000 过渡时设置。当 16 位主计数器以向上 / 向下模式运行的情况下, 计数器在从模数寄存器中设置的值和下一个更低的计数值过渡而改变方向时 TOF 标记被设置。这对应 PWM 周期的结束。(0x0000 计算值对应周期的中央。)

因为 HCS08 MCU 是一种 8 位的架构, 所以一致性机制被设计到定时器计数器中以进行读取操作。当计数器的任何一个字节 (TPMxCNTH or TPMxCNTL) 被读取时, 两个字节都被捕获到缓冲器中, 这样当另一个字节被读取时, 值会显示读取第一个字节时计数的另一个字节。计数器继续正常计数, 但是没有新的值从任意字节中读取, 直到旧的计数的两个字节都被读取。

主定时器的计数器可随时通过将任何值写入 TPMxCNTH 或 TPMxCNTL 计数的任一比特来手动复位。如果在复位计数前只有计数器的一个字节被读取, 那么这种计数器复位方式还会复位一致性机制。

B.6.2 通道模式选择

如果 CPWMS=0 (未规定中央对齐的 PWM 操作), 那么通道 n 状态和控制寄存器中 MSnB 和 MSnA 控制位为相应通道确定基本运行模式。选择包括输入捕获、输出对比或缓冲的边缘对齐 PWM。

B.6.2.1 输入捕获模式

利用输入捕获功能, TPM 可捕获外部事件发生的时间。当输入捕获通道的管脚上发生激活边时, TPM 会将 TPM 计数器的内容锁入到通道值寄存器中 (TPMxCnVH:TPMxCnVL)。上升边、下降边或任何边可被选为触发输入捕获的活动边。

当 16 位捕获寄存器的任何一个字节被读取时, 两个字节都被锁入到缓冲器中, 以支持连贯的 16 位接入而不受顺序的影响。一致性序列可通过向通道状态 / 控制寄存器 (TPMxCnSC) 中写入值来手动复位。

输入捕获事件会设置标记位 (CHnF), 该标记可选择生成一个 CPU 中断请求。

B.6.2.2 输出比较模式

通过输出比较功能, TPM 可生成具有可编程位置、极性、持续时间和频率的定时脉冲。当定时器达到输出对比通道的通道值寄存器中的值时, TPM 可以置 1, 置 0, 翻转引脚状态通道。

在输出比较模式中, 值只有在 16 位寄存器的两个 8 位字节都被写入后被传输到相应定时器的通道寄存器中。这种一致性序列可通过向通道状态 / 控制寄存器 (TPMxCnSC) 中写入值来手动复位。

输出比较事件会设置标记位 (CHnF), 该标记可选择产生 CPU 中断请求。

B.7 TPM 中断

TPM 为主计数器溢出生成可选的中断，为每个通道生成一个中断。通道中断的意义取决于每个通道的运行模式。如果通道被配置用于输入捕获，所选的输入捕获边每次被识别时中断标记被设置。如果通道配置用于输出比较或 PWM 模式，中断标记会在每次主定时器计数器与 16 位通道值寄存器中的值匹配时被设置。参见复位、中断和系统配置一章了解绝对中断向量地址、优先级和本地中断掩码控制位。

对于 TPM 中的每个中断源，会在识别到中断条件（如定时器溢出、通道输入捕获或输出比较事件等）后设置标记位。这个标记可被软件读取（轮询）以确定操作已经发生，或者也可设置相关的启动位（TOIE 或 CHnIE）以启动硬件中断生成。中断启动位被设置时，相关中断标记等于 1 时会生成静态中断。从中断服务程序中返回前，用户软件必须执行一系列步骤来清除中断标记。

B.7.1 清除定时器中断标记

TPM 中断标记通过两个步骤来清除：标记位被设置（1）时被读取，然后是向该位中写入一个 0。如果在这两步间检测到新事件，序列被复位，并且在第二步后中断标记仍被设置以避免错过新事件的可能性。

B.7.2 定时器溢出中断描述

导致 TOF 被设置的条件取决于计数模式（向上或向上 / 向下）。在向上计数模式中，16 位定时器计数器从 0x0000 计数到 0xFFFF，然后在下一个计数时钟上溢出到 0x0000。在从 0xFFFF 过渡到 0x0000 时 TOF 被设置。设置了模数限制的情况下，TOF 标记会在从模数寄存器中设置的值过渡到 0x0000 时被设置。当计数器以向上 / 向下模式运行时，TOF 标记会在计数器从模数寄存器中设置的计数值和下一个更低计数值过渡而改变方向时被设置。这与 PWM 周期的结束对应（0x0000 计数值与周期中央对应）。

B.7.3 通道事件中断描述

通道中断的含义取决于通道的当前模式（输入捕获、输出比较、边缘对齐 PWM 或中央对齐 PWM）。

当通道被配置为输入捕获通道时，ELSnB:ELSnA 控制位选择上升边、下降边、任何边或无边（关）作为触发输入捕获事件的边。检测到选定的边之后，中断标记被设置。标记通过 B.7.1，“清除定时器中断标记”中所述的两步序列清除。

如果通道被配置为输出比较通道，每次主定时器计数器与通道值寄存器中的 16 位值匹配时会设置中断标记。标记通过 B.7.1，“清除定时器中断标记”中所述的两步序列清除。