



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	60KB (60K x 8)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	4K 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/mc9s08dz60mlf

MC9S08DZ60 系列产品的特性

8 位 HCS08 中央处理器（CPU）

- 40-MHz HCS08 CPU（20-MHz 总线）
- HC08 指令集，带附加的 BGND 指令
- 支持最多 32 个中断 / 复位源

片内存储器

- 整个工作电压和温度范围内可读取 / 编程 / 擦除的 Flash 存储器
 - MC9S08DZ60 = 60K
 - MC9S08DZ48 = 48K
 - MC9S08DZ32 = 32K
 - MC9S08DZ16 = 16K
- 最大 2K 的 EEPROM 在线可编程内存；支持 8 字节单页或 4 字节双页擦除分区；执行 Flash 程序的同时可进行编程和擦除操作；支持擦除取消操作
- 最大 4K 的随机存取内存（RAM）

省电模式

- **两种超低功耗停止模式**
- 降低功耗的等待模式
- 超低功耗实时时钟中断，在运行、等待和停止模式下均可操作

时钟源选项

- 振荡器（XOSC）— 闭环控制的皮尔斯（Pierce）振荡器；支持范围 31.25 kHz 至 38.4 kHz 或 1 MHz 至 16MHz 之间的晶体或陶瓷谐振器
- 多功能时钟生成器（MCG）— PLL 和 FLL 模式（在使用内部温度补偿时 FLL 能够达到 1.5% 内的偏差）；带微调功能的内部参考时钟源；带可选择晶体振荡器或陶瓷谐振器的外部参考时钟源

系统保护

- 监视微控制器正常操作的看门狗（COP）复位，支持选择专用的后备 1-kHz 内部时钟源或总线时钟运行
- 带复位和中断的低压检测电路；可选择的电压阈值
- 支持非法指令代码复位
- 支持非法操作地址复位
- 支持 Flash 块保护
- 支持时钟信号丢失保护

开发支持

- 单线背景调试接口
- 片上及在线仿真（ICE），带总线实时捕获功能

外围设备

- ADC — 24 通道，12 位分辨率，2.5 μ S 转换时间，自动比较功能，1.7 mV/ $^{\circ}$ C 温度传感器，包含内部间隙参考源通道
- ACMPx — 两个模拟比较器，支持比较器输出的上升、下降或任意边沿触发的中断；可选择与内部参考电压源进行比较
- MSCAN — CAN 协议 - V2.0 A 和 B；支持标准和扩展数据帧；支持远程帧；5 个带有 FIFO 存储机制的接收缓冲器；灵活的接收识别符过滤器，可编程如下：2 x 32 位、4 x 16 位或 8 x 8 位
- SC1x — 两个 SCI，可支持 LIN 2.0 协议和 SAE J2602 协议；全双工；主节点支持 break 信号生成；从节点支持 break 信号检测；支持激活边沿唤醒
- SPI — 全双工或单线双向；双重缓冲发射和接收；主从模式选择；支持高位优先或低位优先的移位
- IIC — 支持最高 100kbps 的总线波特率；多主节点模式运行；可编程的从地址；通用呼叫地址；逐字节数据传输驱动的中断
- TPMx — 一个 6 通道（TPM1）和一个 2 通道（TPM2）；可支持输入捕捉，输出比较，或每个通道带缓冲的边沿对齐 PWM 输出
- RTC —（实时时钟计数器）8 位模数计数器，带基于二进制或十进制的预分频器；实时时钟功能，使用外部晶体和 RTC 来确保精确时基、时间、日历或任务调度功能；内带低功耗振荡器（1 kHz），用于周期唤醒而不需要外部器件

输入 / 输出

- 53 个通用输入 / 输出（I/O）管脚和 1 个专用输入管脚
- 24 个中断管脚，每个管脚带触发极性选择
- 所有输入管脚上带电压滞后和可配置的上下拉器件
- 所有输入管脚上可配置输出斜率和驱动强度

封装选项

- 64 管脚小尺寸四方扁平封装（LQFP）— 10x10 mm
- 48 管脚小尺寸四方扁平封装（LQFP）— 7x7 mm
- 32 管脚小尺寸四方扁平封装（LQFP）— 7x7 mm

第 2 章

管脚和连接

本章描述连接到各封装管脚的信号，内容包括管脚布局图、建议的系统连接并对信号进行了详细地描述。

2.1 器件管脚分配

本节介绍了 MC9S08DZ60 系列 MCU 各种封装的管脚分配状况。

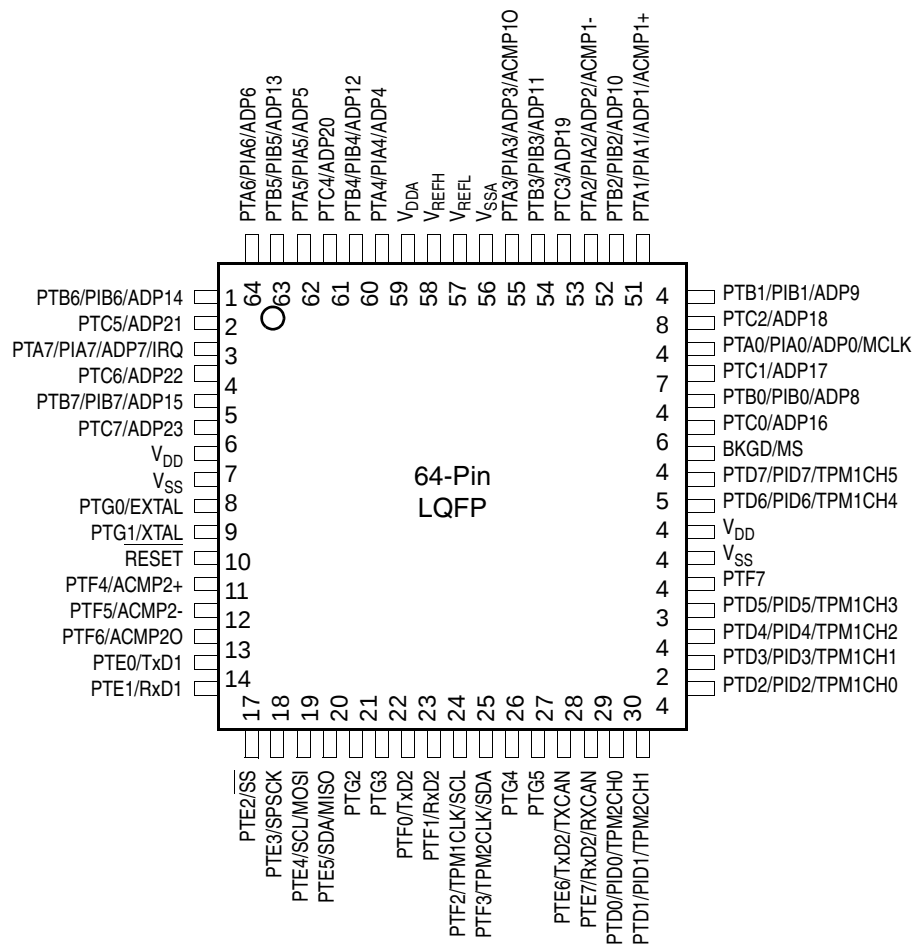


图 2-1. 64 管脚 LQFP

表 4-12. FPROT 寄存器字段描述

字段	描述
7:6 EPS	EEPROM 保护选择位 — 这个 2 位字段决定不能被擦除或编程的受保护 EEPROM 位置。参见表 4-13.
5:0 FPS	Flash 保护选择位 — 这个 6 位字段决定不能被擦除或编程的受保护 Flash 位置。参见表 4-14.

表 4-13. EEPROM 块保护

EPS	受保护的地址域	受保护的内存大小（字节）	受保护的扇区数量
0x3	N/A	0	0
0x2	0x17F0 - 0x17FF	32	4
0x1	0x17E0 - 0x17FF	64	8
0x0	0x17C0-0x17FF	128	16

表 4-14. Flash 块保护

FPS	受保护的地址域	受保护的内存大小（字节）	受保护的扇区数量
0x3F	N/A	0	0
0x3E	0xFA00-0xFFFF	1.5K	2
0x3D	0xF400-0xFFFF	3K	4
0x3C	0xEE00-0xFFFF	4.5K	6
0x3B	0xE800-0xFFFF	6K	8
...	...	...	...
0x37	0xD000-0xFFFF	12K	16
0x36	0xCA00-0xFFFF	13.5K	18
0x35	0xC400-0xFFFF	15K	20
0x34	0xBE00-0xFFFF	16.5K	22
...	...	...	...
0x2C	0x8E00-0xFFFF	28.5K	38
0x2B	0x8800-0xFFFF	30K	40
0x2A	0x8200-0xFFFF	31.5K	42
0x29	0x7C00-0xFFFF	33K	44
...	...	...	...
0x22	0x5200-0xFFFF	43.5K	58
0x21	0x4C00-0xFFFF	45K	60
0x20	0x4600-0xFFFF	46.5K	62
0x19	0x4000-0xFFFF	48K	64
...	...	...	...

6.5.2.1 B 端口数据寄存器 (PTBD)

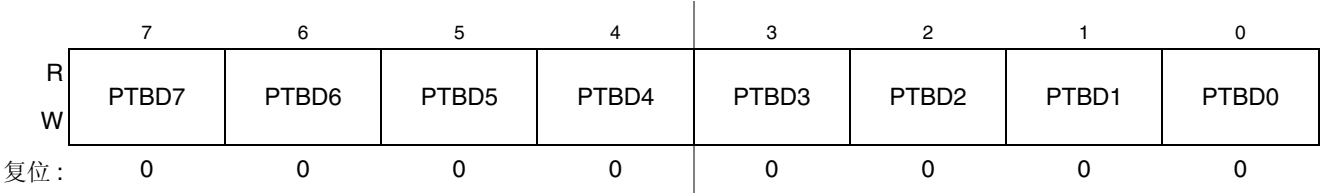


图 6-11. B 端口数据寄存器 (PTBD)

表 6-9. PTBD 寄存器字段描述

字段	描述
7:0 PTBD[7:0]	B 端口数据寄存器位 — 对于配置为输入的 B 端口管脚，读数返回管脚上的逻辑电平。对于配置为输出的 B 端口管脚，读数返回写入寄存器的最后一个值。 写入值被锁定在本寄存器的所有位中。对于配置为输出的 B 端口管脚，逻辑电平被输出到相应的 MCU 管脚。 复位强制 PTBD 都为 0，但是这些 0 未被输出到相应的管脚，因为复位还会将所有端口管脚配置为上拉 / 下拉被禁止的高阻抗输入。

6.5.2.2 B 端口数据方向寄存器 (PTBDD)

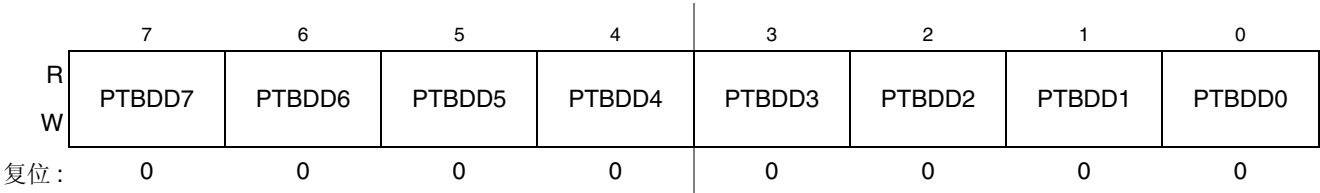


图 6-12. B 端口数据方向寄存器 (PTBDD)

表 6-10. PTBDD 寄存器字段描述

字段	描述
7:0 PTBDD[7:0]	B 端口位的数据方向 — 这些读 / 写位控制着 B 端口管脚的方向以及为 PTBD 读数读取的内容。 0 输入（输出驱动被禁止），读数返回管脚值。 1 B 端口位 - 输出驱动使能，PTBD 读数返回 PTBDn 内容。

6.5.3.1 C 端口数据寄存器 (PTCD)

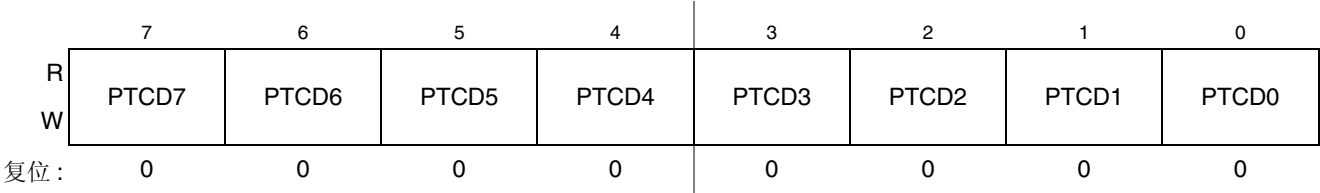


图 6-19. C 端口数据寄存器 (PTCD)

表 6-17. PTCD 寄存器字段描述

字段	描述
7:0 PTCD[7:0]	C 端口数据寄存器位 — 对于配置为输入的 C 端口管脚，读数返回管脚上的逻辑电平。对于配置为输出的 C 端口管脚，读数返回写入寄存器的最后一个值。 写入值被锁定在本寄存器的所有位中。对于配置为输出的 C 端口管脚，逻辑电平被输出到相应的 MCU 管脚。 复位强制 PTCD 都为 0，但是这些 0 未被输出到驱出相应的管脚，因为复位还会将所有端口管脚配置为上拉 / 下拉禁止的高抗阻输入。

6.5.3.2 C 端口数据方向寄存器 (PTCDD)

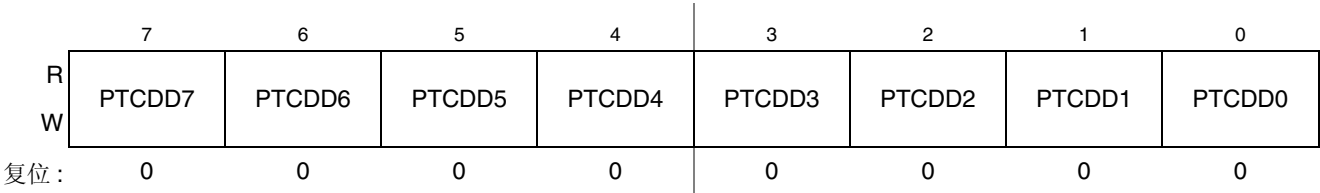


图 6-20. C 端口数据方向寄存器 (PTCDD)

表 6-18. PTCDD 寄存器字段描述

字段	描述
7:0 PTCDD[7:0]	C 端口位的数据方向 — 这些读 / 写位控制着 C 端口管脚的方向以及为 PTCD 读数读取的内容。 0 输入（输出驱动禁止），读数返回管脚值。 1 C 端口位 - 输出驱动使能，PTCD 读数返回 PTCDDn 内容。

表 7-2. 指令集小结 (第 7 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
RSP	复位堆栈指针 (低字节) SPL `` \$FF (高字节未受影响)	INH	9C	1	p	- 1 1 -	- - - -
RTI	从中断返回 SP `` (SP) + \$0001; Pull (CCR) SP `` (SP) + \$0001; Pull (A) SP `` (SP) + \$0001; Pull (X) SP `` (SP) + \$0001; Pull (PCH) SP `` (SP) + \$0001; Pull (PCL)	INH	80	9	uuuuufppp	↑ 1 1 ↑	↑↑↑↑
RTS	从子程序返回 SP `` SP + \$0001; Pull (PCH) SP `` SP + \$0001; Pull (PCL)	INH	81	5	ufppp	- 1 1 -	- - - -
SBC #opr8i SBC opr8a SBC opr16a SBC oprx16,X SBC oprx8,X SBC ,X SBC oprx16,SP SBC oprx8,SP	减去进位 A `` (A) - (M) - (C)	IMM DIR EXT IX2 IX1 IX SP2 SP1	A2 ii B2 dd C2 hh ll D2 ee ff E2 ff F2 9E D2 ee ff 9E E2 ff	2 3 4 4 3 3 5 4	pp rpp prpp prpp rpp rfp pprpp prpp	↑ 1 1 -	-↑↑↑
SEC	设置进位 (C `` 1)	INH	99	1	p	- 1 1 -	- - - 1
SEI	设置中断屏蔽位 (I `` 1)	INH	9B	1	p	- 1 1 -	1 - - -
STA opr8a STA opr16a STA oprx16,X STA oprx8,X STA ,X STA oprx16,SP STA oprx8,SP	将累加器保存在存储器中 M `` (A)	DIR EXT IX2 IX1 IX SP2 SP1	B7 dd C7 hh ll D7 ee ff E7 ff F7 9E D7 ee ff 9E E7 ff	3 4 4 3 2 5 4	wpp pwpp pwpp wpp wp ppwpp pwpp	0 1 1 -	-↑↑-
STHX opr8a STHX opr16a STHX oprx8,SP	保存 H:X (索引寄存器) (M:M + \$0001) `` (H:X)	DIR EXT SP1	35 dd 96 hh ll 9E FF ff	4 5 5	wwpp pwpp pwpp	0 1 1 -	-↑↑-
STOP	使能中断: 停止处理 参见 MCU 文档 I 位 `` 0; 停止处理	INH	8E	2	fp...	- 1 1 -	0 - - -
STX opr8a STX opr16a STX oprx16,X STX oprx8,X STX ,X STX oprx16,SP STX oprx8,SP	将 X (索引寄存器的低 8 位) 保存在存储器中 M `` (X)	DIR EXT IX2 IX1 IX SP2 SP1	BF dd CF hh ll DF ee ff EF ff FF 9E DF ee ff 9E EF ff	3 4 4 3 2 5 4	wpp pwpp pwpp wpp wp ppwpp pwpp	0 1 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)

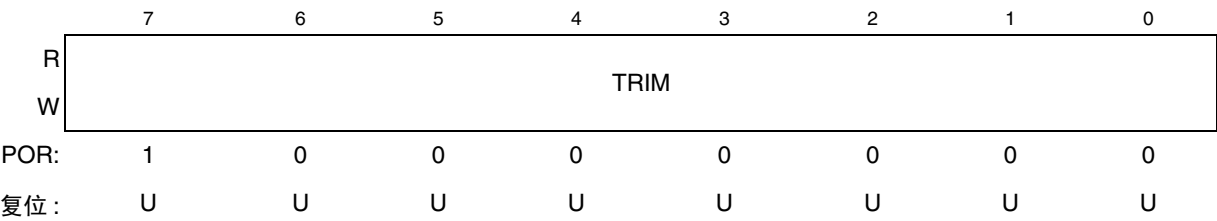


图 8-5. MCG 修正寄存器 (MCGTRM)

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.

8.6.2.3 示例 3: 从 BLPI 转换到 FEE 模式: 外部晶体 = 4 MHz、总线频率 = 16 MHz

本例中, MCG 将选择适当的运行模式, 从以基于内部参考时钟, 运行于 16 kHz 总线频率的 BLPI 模式 (参见前例) 转换到 4MHz 晶体频率、16 MHz 总线频率的 FEE 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, BLPI 必须转换到 FBI 模式。
 - a) MCGC2 = 0x00 (%00000000)
 - MCGSC 中的 LP (位 3) 是 0
 - b) 循环检测, 直到 MCGSC 中的 LOCK (位 6) 置位, 表明 FLL 已经获得锁定。尽管在 FBI 模式中 FLL 被旁通, 但它仍使能并运行。
2. 接下来, FBI 将转换到 FEE 模式。
 - a) MCGC2 = 0x36 (%00110110)
 - RANGE (位 5) 设置为 1, 因为 4 MHz 频率在高频范围内。
 - HGO (位 4) 设置为 1, 为高增益运行配置外部振荡器。
 - EREFS (位 2) 设置为 1, 因为正在使用晶体。
 - ERCLKEN (位 1) 设置为 1, 确保外部参考时钟处于活动状态。
 - b) 循环检测, 直到 MCGSC 中的 OSCINIT (位 1) 是 1, 表明 EREFS 位选择的晶体已经完成初始化。
 - c) MCGC1 = 0x38 (%00111000)
 - CLKS (位 7 和 6) 设置为 %00, 以便将 FLL 输出选为系统时钟源。
 - 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。
 - IREFS (位 1) 清除至 0, 选择外部参考时钟
 - d) 循环检测, 直到 MCGSC 中的 IREFST (位 4) 是 0, 表明外部参考时钟是参考时钟的当前源。
 - e) 循环检, 直到 MCGSC 中的 LOCK (位 6) 置位, 表明 FLL 重新获得了锁定。
 - f) 循环检测, 直到 MCGSC 中的 CLKST (位 3 和 2) 是 %00, 表明已经选择 FLL 输出为 MCGOUT 馈电。

下列代码顺序描述了如何从 FEI 模式转换到 PEE 模式，直到设置 8 MHz 晶体参考频率来获得 8 MHz 总线频率。因为 MCG 复位后处于 FEI 模式，本例还显示了如何初始化 MCG，以实现在复位后进入 PEE 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，FEI 必须转换到 FBE 模式：

a) MCGC2 = 0x36 (%00110110)

- BDIV 位 7 和 6) 设置为 %00 或除以 1。
 - RANGE (位 5) 设置为 1，因为 8 MHz 的频率属于高频范围。
 - HGO (位 4) 设置为 1，为高增益运行配置外部振荡器。
 - EREFS (位 2) 设置为 1，因为正在使用晶体。
 - ERCLKEN (位 1) 设置为 1，确保外部参考时钟处于活动状态。
- b) 循环检测，直到 MCGSC 中的 OSCINIT (位 1) 是 1，表明 EREFS 位选择的晶体已经完成初始化。
- c) 禁止中断 (如果适用，在 CCR 中设置中断位)。
- d) MCGC1 = 0xB8 (%10111000)
- CLKS (位 7 和 6) 设置为 %10，以便将外部参考时钟选择为系统时钟源。
 - RDIV (位 5-3) 设置为 %111 或除以 128。

注意

$8 \text{ MHz} / 128 = 62.5 \text{ kHz}$ ，这大于 FLL 要求的 $31.25 \text{ kHz} \text{ -- } 39.0625 \text{ kHz}$ 频率范围。因此，当 FBE 的转换完成后，必须通过在软件中设置 MCGC2 中 LP 位，让 MCG 立即进入 BLPE 模式。

- IREFS 位 2) 清除至 0，选择外部参考时钟。
- e) 循环检测，直到 MCGSC 中的 IREFST (位 4) 是 0，表明外部参考是参考时钟的当前源。
- f) 循环检测，直到 MCGSC 中的 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，从 FBE 模式转换到 BLPE 模式：

a) MCGC2 = 0x3E (%00111110)

- MCGC2 中的 LP (位 3) 为 1 (已进入 BLPE 模式)

注意

在 1d 和 2a 步骤间没有额外步骤 (包括中断)。

- b) 使能中断 (如果适用，清除 CCR 中的中断位)
- c) MCGC1 = 0x98 (%10011000)
- RDIV (位 5-3) 设置为 %011 或除以 8，因为 $8 \text{ MHz} / 8 = 1 \text{ MHz}$ ，这在 PLL 要求的 $1 \text{ MHz} \text{ -- } 2 \text{ MHz}$ 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。更改它们只会为 PLL 建立在 PBE 模式中使用的分频器。
- d) MCGC3 = 0x44 (%01000100)
- PLLS (位 6) 设置为 1，选择 PLL。在 BLPE 模式中，更改该位只会使 MCG 准备在 PBE 模式中的 PLL 使用。

图 10-4. 输入通道选择

ADCH	输入选择
00000–01111	AD0–15
10000–11011	AD16–27
11100	预留
11101	V_{REFH}
11110	V_{REFL}
11111	模块禁止

10.4.2 状态和控制寄存器 2 (ADCSC2)

ADCSC2 寄存器用来控制 ADC 模块的比较功能、转换触发和转换状态。

	7	6	5	4	3	2	1	0
复位:	0	0	0	0	0	0	0	0

图 10-5. 状态和控制寄存器 2 (ADCSC2)

表 10-4. ADCSC2 寄存器字段描述

字段	描述
7 ADACT	转换状态 — ADACT 显示正在进行一个转换。当开始转换时就设置 ADACT，当转换完成或中止时就清除 ADACT。 0 转换未进行 1 转换正在进行
6 ADTRG	转换触发选择 — ADTRG 用来选择转换的触发类型。 有 2 种触发可供选择：软件触发和硬件触发。当选择软件触发时，写入 ADCSC1 就能发起转换。当选择硬件触发时，ADHWT 触发后就能发起转换。 0 选择软件触发 1 选择硬件触发
5 ACFE	比较功能使能 — ACFE 用来使能比较功能。 0 比较功能禁止 1 比较功能使能
4 ACFGT	比较功能大于使能 — ACFGT 用来配置使当前的转换结果大于或等于比较值时触发，默认的比较功能为在当前的转换结果小于比较值时触发。 0 当输入小于比较值时触发 1 当输入大于或等于比较值时触发

10.5 功能描述

复位期间或当 ADCH 位都高时，ADC 模块禁止。当已经完成当前转换，而下一次转换还未发起时，模块进入空闲状态。空闲时，模块处于最低功耗状态。

ADC 可以对软件选择的任意通道实施模数转换。在 12 位和 10 位模式中，所选的通道电压通过逐次逼近算法被转换成 12 位数字结果。在 8 位模式中，所选的通道电压通过逐次逼近算法被转换成 9 位数字结果。

转换完成后，结果保存在数据寄存器（ADCRH 和 ADCRL）中。在 10 位模式中，结果被圆整到 10 位并保存在数据寄存器（ADCRH 和 ADCRL）中。在 8 位模式中，结果被圆整到 8 位并保存在 ADCRL 中。然后设置转换完成标记（COCO），如果已经使能了转换完成中断（AIEN = 1），则触发中断。

ADC 模块能够自动地把转换结果与比较寄存器的内容进行比较。通过设置 ACFE 位并结合任意一种转换模式和配置一起运行，就使能了比较功能。

10.5.1 时钟选择和分频控制

ADC 模块可选择 4 个时钟源之一，然后由可配置值进行分频，生成转换器的输入时钟（ADCK）。时钟源通过 ADICLK 位设置从以下源中选择。

- 总线时钟，等于软件运行的频率。这是复位后的默认选择。
- 总线时钟除以 2，如果总线时钟很高，允许总线时钟最大除以 16。
- ALTCLK，由此 MCU 定义（参见本章节的概述部分）。
- 异步时钟（ADACK）- 该时钟从 ADC 模块内部时钟源产生。当 MCU 则处于等待或 STOP3 模式时，此时钟源仍然有效，从而实现此模式下的低噪音转换。

无论选择哪种时钟，其频率必须在 ADCK 的指定频率范围内。如果可用时钟太慢，ADC 将无法保证正常运行。如果可用时钟太快，那么时钟必须分频为适当的频率。除数由 ADIV 位指定，可以是 1、2、4 或 8。

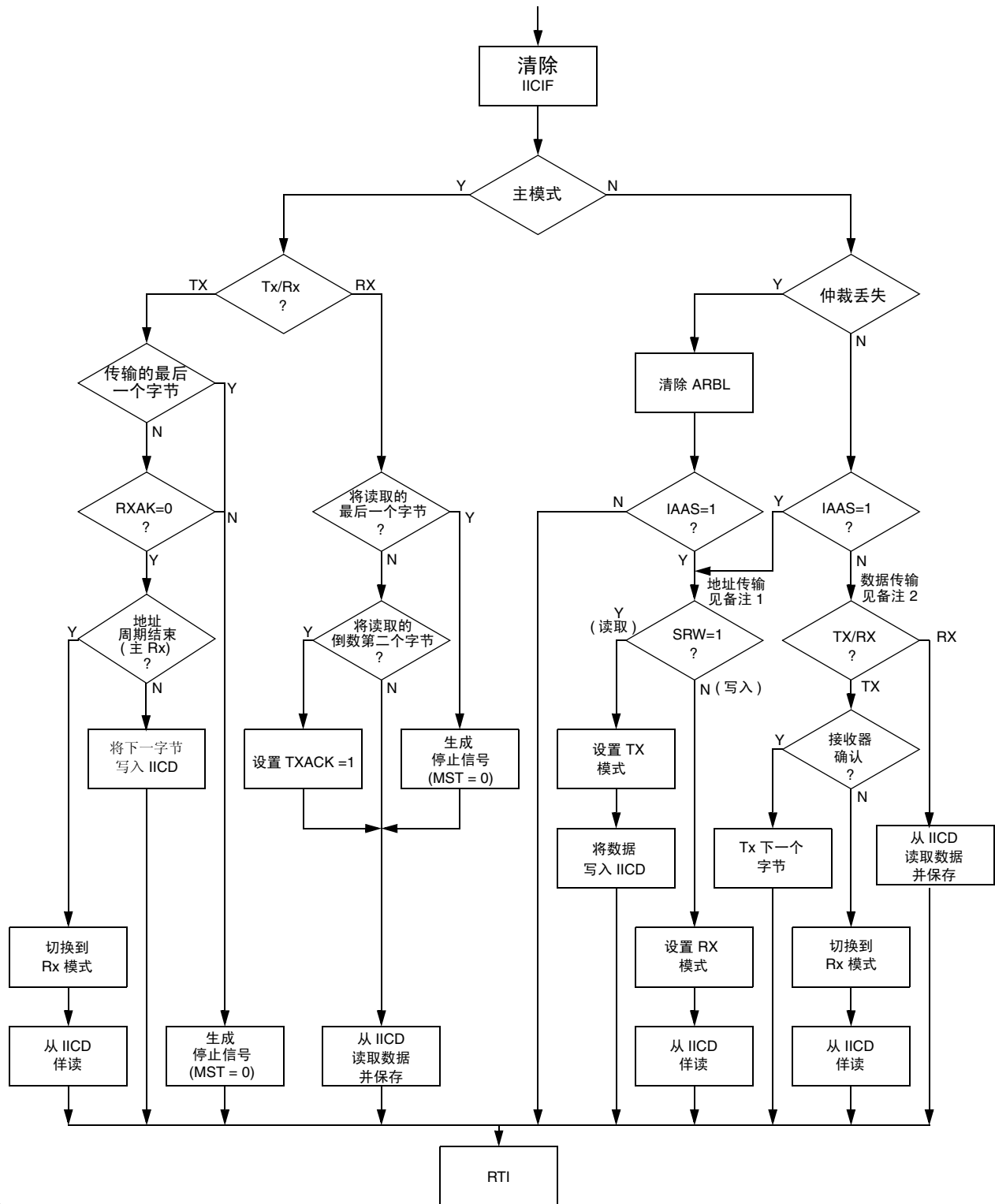
10.5.2 输入选择和管脚控制

管脚控制寄存器（APCTL3，APCTL2 和 APCTL1）用来禁止对作为模拟输入的管脚的 I/O 控制功能。当置位管脚控制寄存器相应位时，对应的 MCU 管脚进入以下状态：

- 输出缓冲器进入高阻抗状态。
- 输入缓冲器禁止。对于其输入缓冲器被禁止的任何管脚，I/O 端口读数均返回 0。
- 上拉禁止。

10.5.3 硬件触发

ADC 模块有一个可选的异步硬件转换触发 ADHWT，当设置了 ADTRG 位时，ADHWT 使能。并不是所有 MCU 上都有这个源。如需了解该 MCU 的特定 ADHWT 源的更多报文，请参见本章概述部分。



备注：

1. 如果使能了通用呼叫，必须进行检查，以确定收到的地址是否为通用呼叫地址（0x00）。如果收到的地址是通用呼叫地址，那么通用呼叫必须由用户软件处理。
2. 当使用 10 位寻址来寻址从器件时，从器件在扩展地址的首字节后发现中断。用户软件必须为该中断确保这一点，那就是忽略 IICD 的内容，且不把它作为有效数据传输对待。

图 11-12. 典型的 IIC 中断程序

表 12-6. CANBTR1 寄存器字段描述

字段	描述
7 SAMP	采样 — 该位确定每位时间所采集的 CAN 总线样本数量 0 每位 1 个样本 1 每位 3 个样本 1。 如果 SAMP = 0，得到的位值等于采样点上定位的单个位的值。如果 SAMP = 1，得到的位值通过在总共三个采样点上使用多数规则来决定。要实现更高比特速率，建议每个位时间只采集一个样本（SAMP = 0）。).
6:4 TSEG2[2:0]	TSEG2[2:0] 时间段 2—位时间内的时间段固定每个位时间的时钟周期数和采样点的位置 (参见图 12-43). 时间段 2（TSEG2）值可以如表 12-7 所示进行编程。
3:0 TSEG1[3:0]	时间段 1—位时间内的时间段固定每个位时间的时钟周期数和采样点的位置 (参见图 12-43). 时间段 1（TSEG1）值可以如表 12-8 所示进行编程。.

表 12-7. 时间段 2 值

TSEG22	TSEG21	TSEG20	时间段 2
0	0	0	1 Tq 时钟周期 ¹
0	0	1	2 Tq 时钟周期
:	:	:	:
1	1	0	7 Tq 时钟周期
1	1	1	8 Tq 时钟周期

¹ This setting is not valid. Please refer to 表 12-35 for valid settings.

表 12-8. 时间段 1 值

TSEG13	TSEG12	TSEG11	TSEG10	时间 段 1
0	0	0	0	1 Tq c 时钟周期 ¹
0	0	0	1	2 Tq 时钟周期 ¹
0	0	1	0	3 Tq 时钟周期 ¹
0	0	1	1	4 Tq 时钟周期
:	:	:	:	:
1	1	1	0	15 Tq 时钟周期
1	1	1	1	16 Tq 时钟周期

¹ 该设置无效，请参见表 12-35 查看有效设置。

位时间由振荡器频率、波特率预分频器和每位的时间冲量（Tq）数量确定（如表 12-7 和表 12-8).

等式 12-1

Bit Time= $\frac{\text{Prescaler value}}{f_{\text{CANCLK}}} \cdot (1 + \text{TimeSegment1} + \text{TimeSegment2})$

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。

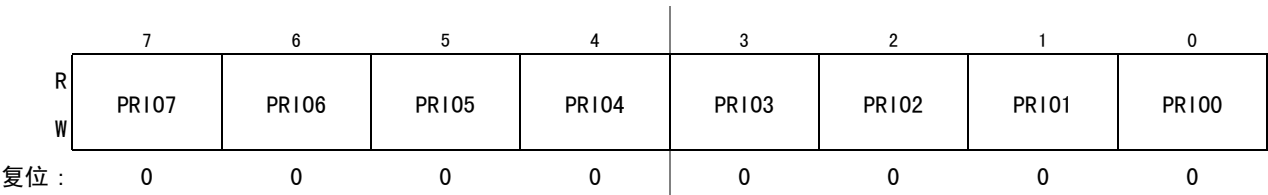


图 12-35. 发送缓冲器优先寄存器 (TBPR)

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。

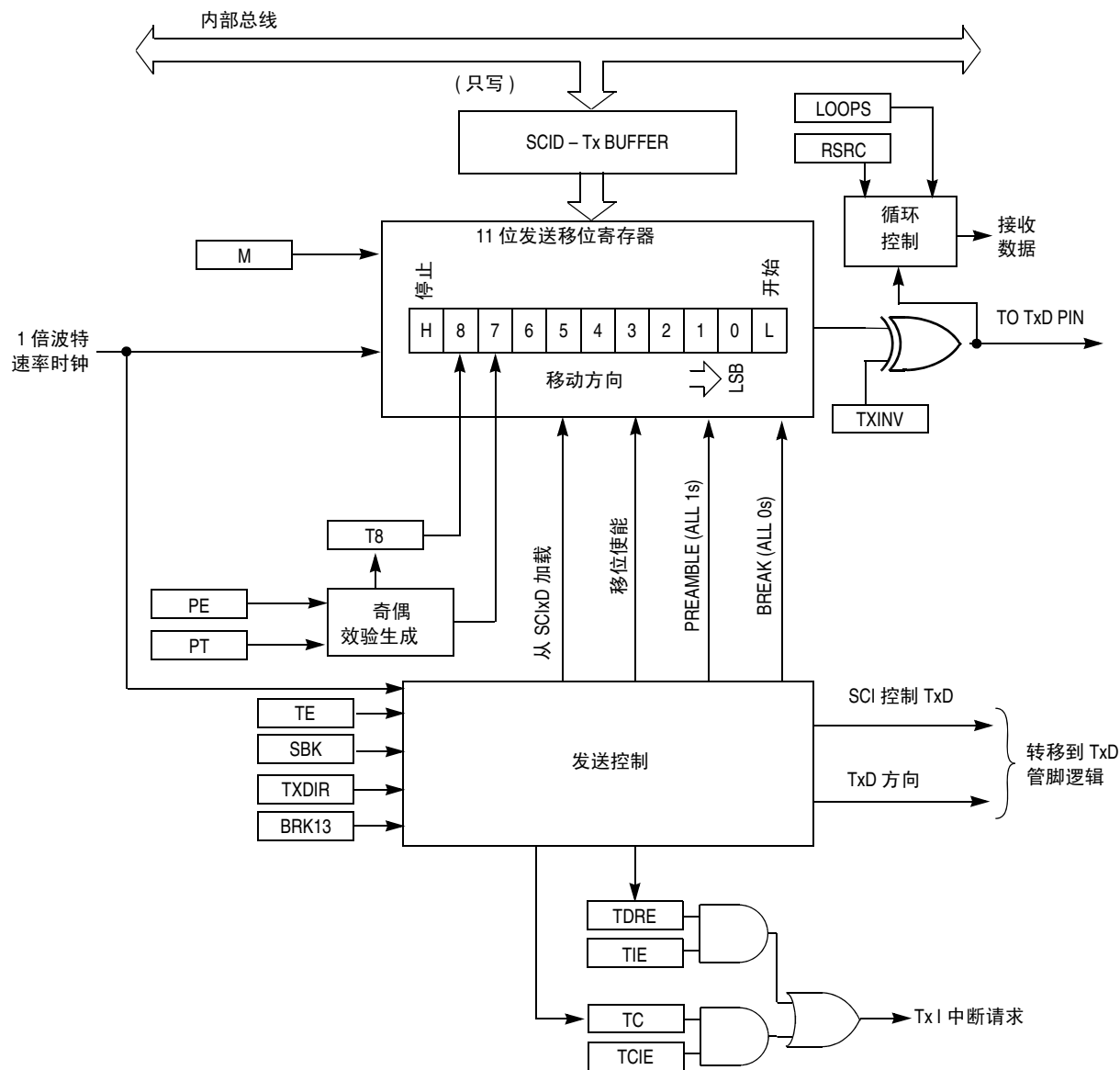


图 14-2. SCI 发射器结构图

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。

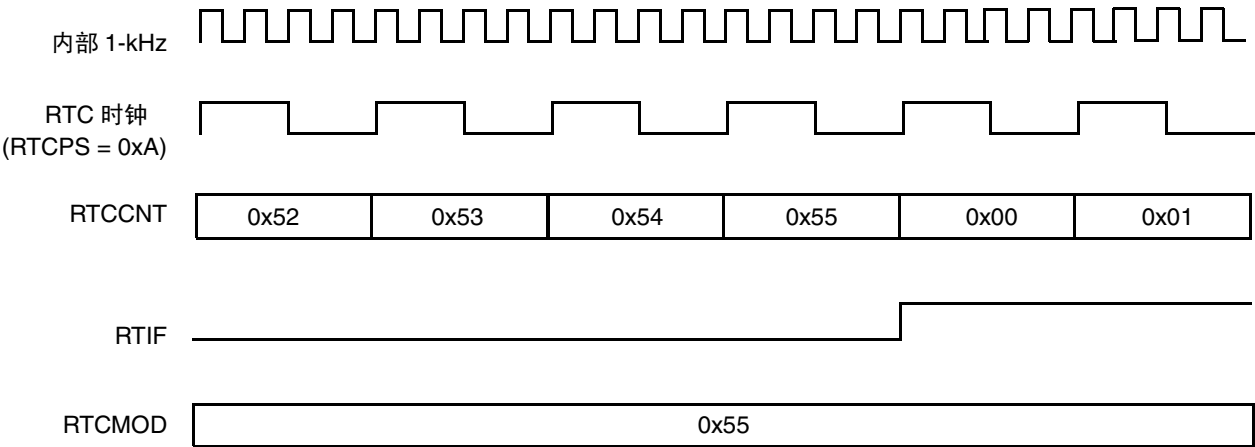


图 15-6. RTC 计数器溢出实例

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;
```


表 A-10. 12- 位 ADC 特性 ($V_{REFH} = V_{DDAD}$, $V_{REFL} = V_{SSAD}$) (续)

特性	条件	C	符号	最小值	典型值 ¹	最大值	单位	注释
转换时间（包括采样时间）	短采样（ADLSMP=0）	D	t _{ADC}	—	20	—	ADCK 周期	参见表 A-12， 查看转换时间 变量
	长采样（ADLSMP=1）			—	40	—		
采样时间	短采样（ADLSMP=0）	D	t _{ADS}	—	3.5	—	ADCK 周期	
	长采样（ADLSMP=1）			—	23.5	—		
未调整误差总数	12 位模式	T	E _{TUE}	—	±3.0	±10	LSB ²	包括基本倍数
	10 位模式	P		—	±1	±2.5		
	8 位模式	T		—	±0.5	±1.0		
差分非线性误差	12 位模式	T	DNL	—	±1.75	±4.0	LSB ²	
	10 位模式 ³	P		—	±0.5	±1.0		
	8 位模式 ³	T		—	±0.3	±0.5		
积分非线性误差	12 位模式	T	INL	—	±1.5	±4.0	LSB ²	
	10 位模式	T		—	±0.5	±1.0		
	8 位模式	T		—	±0.3	±0.5		
零刻度误差	12 位模式	T	E _{ZS}	—	±1.5	±6.0	LSB ²	V _{ADIN} = V _{SSAD}
	10 位模式	P		—	±0.5	±1.5		
	8 位模式	T		—	±0.5	±0.5		
满刻度误差	12 位模式	T	E _{FS}	—	±1	±4.0	LSB ²	V _{ADIN} = V _{DDAD}
	10 位模式	T		—	±0.5	±1		
	8 位模式	T		—	±0.5	±0.5		
量化误差	12 位模式	D	E _Q	—	-1 to 0	-1 to 0	LSB ²	
	10 位模式			—	—	±0.5		
	8 位模式			—	—	±0.5		
输入漏电流误差	12 位模式	D	E _{IL}	—	±1	±10.0	LSB ²	Pad leakage ⁴ * R _{AS}
	10 位模式			—	±0.2	±2.5		
	8 位模式			—	±0.1	±1		
温度传感器范围	-40°C– 25°C	D	m	—	3.266	—	mV/ °C	
	25°C– 125°C			—	3.638	—		
温度传感器电压	25°C	D	V _{TEMP25}	—	1.396	—	V	

¹ 典型值假设 $V_{DDAD} = 5.0V$ 、温度 = 25°C, $f_{ADCK} = 1.0MHz$, 除非另有其他说明。典型值仅用于参考, 并在生产中测试。

² $1 \text{ LSB} = (V_{REFH} - V_{REFL}) / 2^N$

³ 10 位和 8 位模式中保证单调和无丢码。

⁴ 基于输入引脚 (input pad) 漏电流。请参考板电气 (pad electricals)。

所有 TPM 通道都可以独立编程为输入捕捉、输出比较或缓冲边沿 PWM 通道。

B.4 外部信号描述

与定时器关联的任意管脚配置为定时器输入时，被动上拉功能使能。复位后，TPM 模块禁止，所有管脚默认设置为被动上拉功能禁止的通用输入。

B.4.1 外部 TPM 时钟源

当定时器状态和控制器寄存器里的控制位 `CLKSB:CLKSA` 设置为 1:1 时，预分频器及 TPMx 的 16 位计数器由外部时钟源 `TPMxCLK` 驱动。该时钟源与 I/O 管脚连接。外部时钟和 TPM 的其余部分之间需要一个同步装置。该同步装置采用总线时钟，因而外部源的频率必须低于总线速率时钟频率的 1.5 倍。这个外部时钟源的频率上限明确规定为总线频率的 1/4，以便能适应负载循环和相位锁定环路（PLL）/ 频率锁定环路（FLL）频率抖动产生的影响。

在部分设备上，外部时钟输入由其中一个 TPM 通道共享。当 TPM 通道作为外部时钟输入共享时，关联 TRM 通道不能使用该管脚。（该通道仍然可以作为软件定时器用于输出比较模式）。此外，如果其中一个 TPM 通道用作外部时钟输入，对应的 `ELSnB:ELSnA` 控制位必须设置为 0:0，因此该通道不会使用同一个管脚。

B.4.2 TPMxCHn — TPMx 通道 n I/O 管脚

所有 TPM 通道都与 MCU 上的一个 I/O 管脚关联。这个管脚的功能与通道配置有关。部分情况不需要管脚功能，因此该管脚由通用 I/O 控制装置进行控制。当定时器拥有某个端口管脚的控制时，端口数据和数据方向寄存器不会对关联管脚产生影响。如需了解共享管脚功能的相关信息，请参见 *Pins and Connections* 章节。

B.5 寄存器定义

TPM 包括：

- 8 位状态和控制寄存器 (TPMxSC)
- 16 位计数器 (TPMxCNTH:TPMxCNTL)
- 16 位模量寄存器 (TPMxMODH:TPMxMODL)

每个定时器通道都包括：

- 8 位状态和控制寄存器寄存器 (TPMxCnSC)
- 16 位通道值寄存器 (TPMxCnVH:TPMxCnVL)

如需了解所有 TPM 寄存器的绝对地址分配信息，请参见本文 *Memory* 章节的直接地址页寄存器概况。

