



Welcome to [E-XFL.COM](https://www.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	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 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/product-detail/nxp-semiconductors/s9s08dz60f2clh

表 3-2. 停止模式

外围设备	模式	
	Stop2	Stop3
CPU	关闭	待机
RAM	待机	待机
Flash/EEPROM	关闭	待机
并行端口寄存器	关闭	待机
ACMP	关闭	关闭
ADC	关闭	可选择开启 ¹
IIC	关闭	待机
MCG	关闭	可选择开启 ²
MSCAN	关闭	待机
RTC	可选择开启 ³	可选择开启 ³
SCI	关闭	待机
SPI	关闭	待机
TPM	关闭	待机
电压调节器	关闭	可选择开启 ⁴
XOSC	关闭	可选择开启 ⁵
I/O 管脚	状态被保持	状态被保持
BDM	关闭 ⁶	可选择开启
LVD/LVW	关闭 ⁷	可选择开启

¹ 要求启用 DC 时钟和 LVD，否则为待机。

² MCGC1 中设置 IRCLKEN 和 IREFSTEN，否则为待机。

³ 要求启用 RTC，否则为待机。

⁴ 要求启用 LVD 或 BDC。

⁵ MCGC2 中设置 ERCLKEN 和 EREFSTEN，否则为待机。在高频率范围（MCGC2 中设置 RANGE）还要求在 Stop3 中启用 LVD。

⁶ 如果进入 Stop2 模式时设置了 ENBDM，MCU 实际上会进入 Stop3 模式。

⁷ 如果进入 Stop2 模式时设置了 LVDSE，MCU 实际上会进入 Stop3 模式。

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。

表 5-11. SPMSC2 寄存器字段描述

字段	描述
2 PPDACK	局部断电确认 — 向 PPDACK 写入 1 清除 PPDF 位。
0 PPDC	局部断电控制 — 这个单次写入有效的位控制着是选择 STOP2 还是选择 STOP3 模式。 0 Stop3 模式启动。 1 Stop2、局部断电、模式启动。

表 5-12. LVD 和 LVW 跳变点典型值¹

LVDV:LVWV	LVW 跳变点	LVD 跳变点
0:0	$V_{LVW0} = 2.74 \text{ V}$	$V_{LVD0} = 2.56 \text{ V}$
0:1	$V_{LVW1} = 2.92 \text{ V}$	
1:0	$V_{LVW2} = 4.3 \text{ V}$	$V_{LVD1} = 4.0 \text{ V}$
1:1	$V_{LVW3} = 4.6 \text{ V}$	

¹ 最小最大值请参见附录电气特性。

6.5.5.2 E 端口数据方向寄存器 (PTEDD)

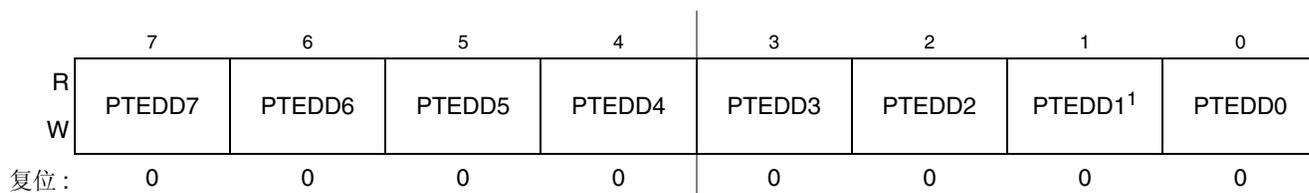


图 6-33. E 端口数据方向寄存器 (PTEDD)

¹ PTEDD1 对输入 PTE1 管脚没有影响。

表 6-31. PTEDD 寄存器字段描述

字段	描述
7:0 PTEDD[7:0]	E 端口位的数据方向 — 这些读 / 写位控制着 E 端口管脚的方向以及为 PTED 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 E 端口位 - 输出驱动使能, PTED 读取返回 PTEDn 内容。

6.5.5.3 E 端口上拉使能寄存器 (PTEPE)



图 6-34. E 端口寄存器内部上拉使能 (PTEPE)

表 6-32. PTEPE 寄存器字段描述

字段	描述
7:0 PTEPE[7:0]	E 端口内部上拉使能位 — 这些控制位决定是否相关的 PTE 管脚使能内部上拉器件。对于配置为输出的 E 端口管脚, 这些位不会产生影响, 同时内部拉器件被禁止。 0 E 端口位 - 内部上拉器件禁止。 1 E 端口位 - 内部上拉器件使能。

注意

只有当使用管脚中断功能且配置了相应的边沿选择和管脚选择功能时, 才能使用下拉器件。

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 pwwpp pwwpp	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 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 -	- - - -

- d) BLPE: 如果通过 BLPE 模式转换, 将 MCGC2 的 LP (位 3) 转换到 0, 切换到 PBE 模式。
 - e) PBE: 循环检测, 直到 MCGSC 中的 PLLST (位 5) 已经设置, 表明 PLLS 时钟的当前源是 PLL。
 - f) PBE: 循环检测, 直到 MCGSC 中的 LOCK (位 6) 已经设置, 表明 PLL 已经获得锁定。
3. 最后, PBE 模式转换到 PEE 模式:
- a) MCGC1 = 0x10 (%00010000)
 - MCGSC1 中的 CLKS (位 7 和 6) 设置为 %00, 以便将 PLL 输出选择为系统时钟源。
 - b) 循环检测, 直到 MCGSC 中的 CLKST (位 3 和 2) 是 %11, 表明已经选择 PLL 输出为当前时钟模式的 MCGOUT 馈电。
 - 这样, RDIV 除以 4、BDIV 除以 1、VDIV 乘以 16, $MCGOUT = [(4 \text{ MHz} / 4) * 16] / 1 = 16 \text{ MHz}$, 总线频率是 MCGOUT / 2 或 8 MHz

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。

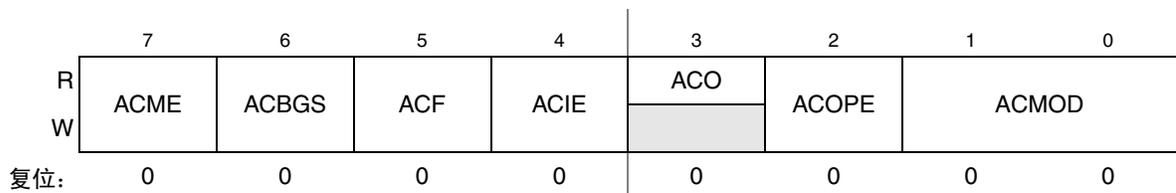


图 9-3. ACMPx 状态和控制寄存器 (ACMPxSC)

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。

10.3 外部信号描述

ADC 模块最多可支持 28 个独立模拟输入。它还需要 4 个电源 / 参考 / 接地连接。

表 10-2. 信号属性

名称	功能
AD27-AD0	模拟通道输入
V_{REFH}	高参考电压
V_{REFL}	低参考电压
V_{DDAD}	模拟电源
V_{SSAD}	模拟接地

10.3.1 模拟电源 (V_{DDAD})

ADC 模拟部分使用 V_{DDAD} 作为其电源连接。在有些封装中, V_{DDAD} 与 V_{DD} 是内部连接。如果是外部连接, 将 V_{DDAD} 管脚连到与 V_{DD} 相同的电平。为了确保干净的 V_{DDAD} 信号, 可能还需要外部滤波。

10.3.2 模拟接地 (V_{SSAD})

ADC 模拟部分使用 V_{SSAD} 作为其接地连接。在有些封装中, V_{SSAD} 与 V_{SS} 是内部连接。如果是外部连接, 将 V_{SSAD} 管脚连到与 V_{SS} 相同的电平。

10.3.3 参考电压高 (V_{REFH})

V_{REFH} 是转换器的高参考电压。在有些封装中, V_{REFH} 与 V_{DDAD} 是内部连接。如果是外部连接, V_{REFH} 可以连接到与 V_{DDAD} 相同的电平, 或者由介于 V_{DDAD} 最低限值和 V_{DDAD} 电平之间的外部源驱动 (V_{REFH} 必须不能超过 V_{DDAD})。

10.3.4 参考电压低 (V_{REFL})

V_{REFL} 是转换器的低参考电压。在有些封装中, V_{REFL} 与 V_{SSAD} 是内部连接。如果是外部连接, 将 V_{REFL} 管脚连到和 V_{SSAD} 相同的电平。

10.3.5 模拟通道输入 (ADx)

ADC 模块最多可支持 28 个独立的模拟输入。通过 $ADCH$ 通道选择位选择转换。

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 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。

11.2.1 特性

IIC 包括以下明显的特性：

- 兼容 IIC 总线标准
- 多主操作
- 可以软件编程为 64 种不同串行时钟频率的任意一种
- 软件可选应答位
- 中断驱动的逐字节数据传输
- 仲裁丢失中断，可以自动地从主模式切换到从模式
- 主叫地址识别中断
- 开始和停止信号生成 / 检测
- 重复生成启动信号
- 应答位生成 / 检测
- 总线忙检测
- 通用呼叫识别
- 10 位地址扩展

11.2.2 运行模式

不同 MCU 模式中的 IIC 的简要描述如下：

- 运行模式 — 这是基本运行模式。要降低这种模式下的功耗，只需禁止模块。
- 等待模式 — 当 MCU 处于等待模式时，模块继续运行并能够提供唤醒中断。
- 停止模式 — IIC 在 STOP3 模式中是不停止的，以降低功耗。停止指令不会影响 IIC 寄存器状态。STOP2 复位寄存器内容。

表 12-25. IDR0 寄存器字段描述 — 扩展

字段	描述
7:0 ID[28:21]	扩展格式标识符—该标识符由 29 个扩展格式位 (ID[28:0]) 组成。ID28 是最高的位，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。

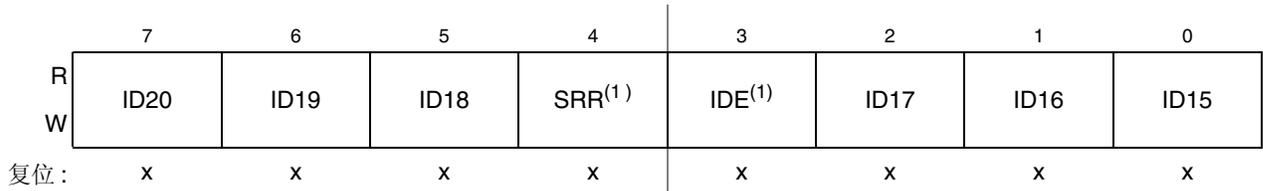


图 12-26. I 标识符寄存器 3 (IDR3) — 扩展标识符映射

¹ 1 SRR 和 IDE 都为 1。

表 12-26. 标识符寄存器 1 (IDR1) — 扩展标识符映射

字段	描述
7:5 ID[20:18]	扩展格式标识符—该标识符由 29 个扩展格式位 (ID[28:0]) 组成。ID28 是最高的位，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。
4 SRR	替代远程请求—该固定隐性位仅用于扩展格式。它必须由用户为传输缓冲器置位为 1，并为接收缓冲器在 CAN 总线上置位为接收。
3 IDE	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) 2:0
2:0 ID[17:15]	扩展格式标识符—该标识符由 29 个扩展格式位 (ID[28:0]) 组成。ID28 是最高的位，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。

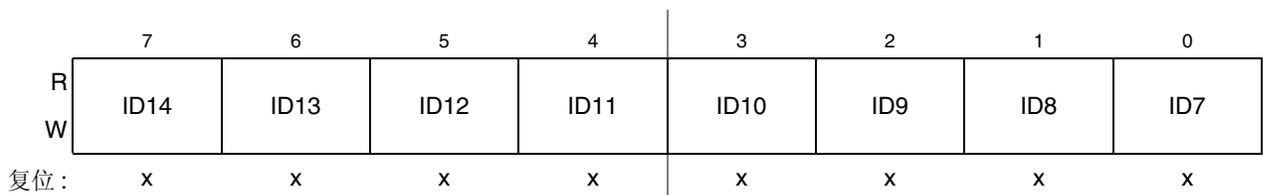


图 12-27. 标识符寄存器 2 (IDR2) — 扩展标识符映射

表 12-27. IDR2 寄存器字段说明—扩展

字段	描述
7:0 ID[14:7]	扩展格式标识符—该标识符由 29 个扩展格式位 (ID[28:0]) 组成。ID28 是最高的位，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。

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)

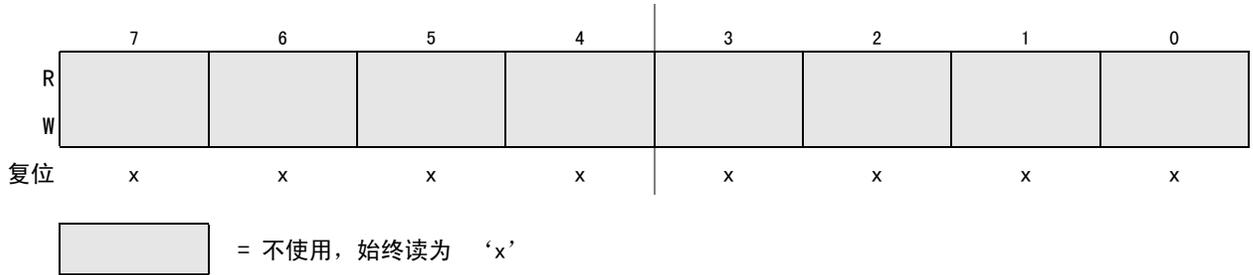


图 12-31. 标识符寄存器 2 — 标准映射

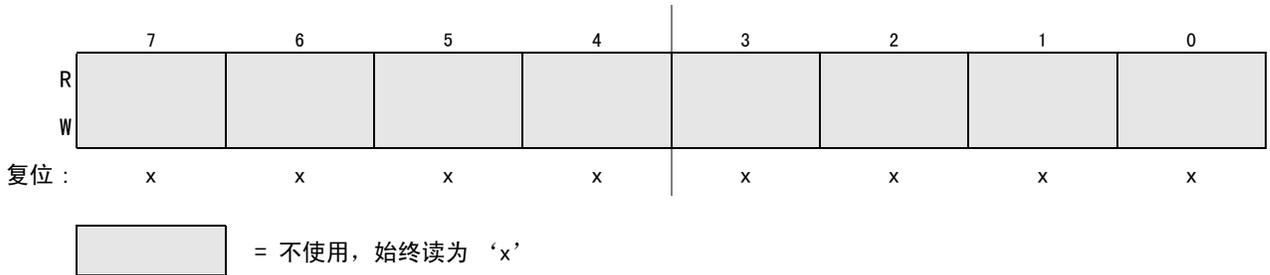


图 12-32. 标识符寄存器 3 — 标准映射

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。

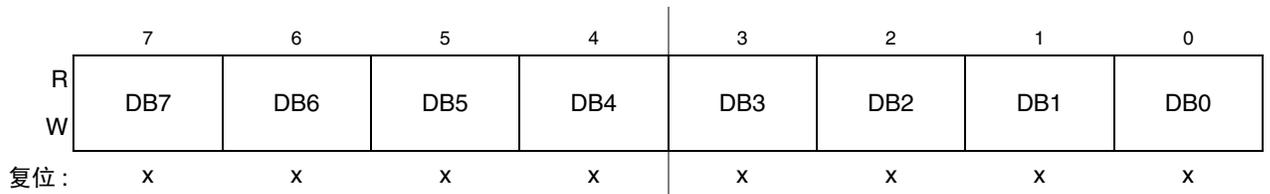


图 12-33. 数据段寄存器 (DSR0 - DSR7) — 扩展标识符映射

12.5.2 报文存储

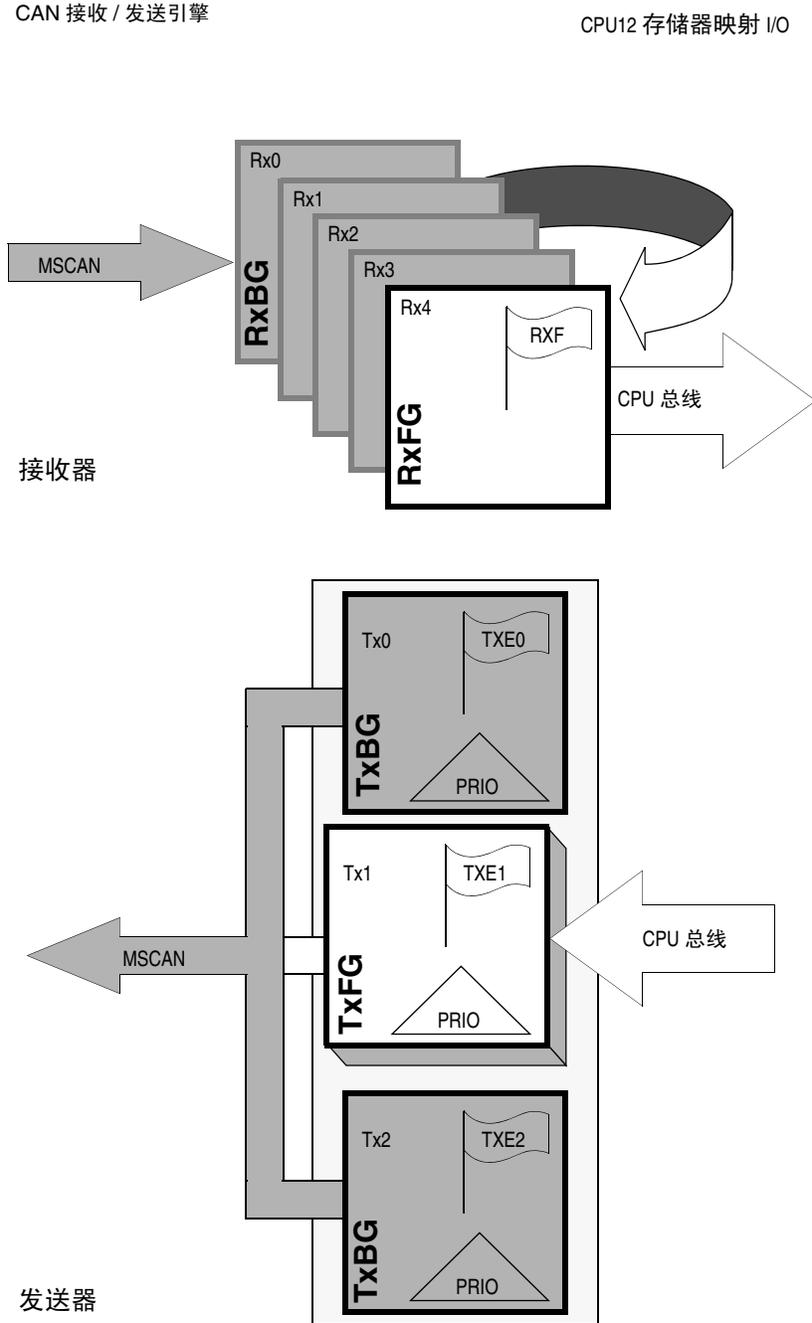


图 12-38. 报文缓冲器结构的用户模型

MSCAN 促进了一个能够满足一系列网络应用需求的先进报文存储系统。

表 14-3. SCiXBDL 字段描述

字段	描述
7:0 SBR[7:0]	波特率模数系数 — SBR[12:0] 中的这 13 个位统称为 BR，它们为 SCI 波特率发生器设置模数除数系数。当 BR = 0，SCI 波特率发生器禁止，以降低电源电流。当 BR = 1~8191 时，SCI 波特率 = BUSCLK / (16xBR)。也可参见表 14-2. 中的 BR 位。

14.2.2 SCI 控制寄存器 1(SCiXC1)

该读 / 写寄存器用于控制 SCI 系统的各种可选功能。

	7	6	5	4	3	2	1	0
R								
W								
复位	0	0	0	0	0	0	0	0

图 14-6. SCI 控制寄存器 1 (SCiXC1)

表 14-4. SCiXC1 字段描述

字段	描述
7 LOOPS	循环模式选择 — 在环回模式和正常的 2 管脚全双工模式之间选择。当 LOOPS = 1，发射器输出内部连接到接收器输入。 0 正常运行 — RxD 和 TxD 使用独立管脚。 1 循环模式或单线模式，发射器输出内部连接到接收器输入（见 <st-blue>RSRC 位）。SCI 不使用 RxD 管脚。
6 SCISWAI	等待模式中的 SCI 停止 0 SCI 时钟继续在等待模式中运行，因此 SCI 可以是唤醒 CPU 的中断源。 1 SCI 时钟在 CPU 处于等待模式时冻结。
5 RSRC	接收器源选择 — 该位没有任何意义或影响，除非 LOOPS 位设置为 1。当 LOOPS = 1 时，接收器输入内部连接到 TxD 管脚，RSRC 决定该连接是否也连接到发射器输出。 0 假设 LOOPS = 1，RSRC = 0 选择内部环回模式，SCI 不使用 RxD 管脚。 1 单线 SCI 模式，其中 TxD 管脚连接到发射器输出和接收器输入。
4 M	9 位或 8 位模式选择 0 正常 — 启动 + 8 个数据位（LSB 先发）+ 停止 1 接收器和发射器使用 9 位数据字符 启动 + 8 个数据位（LSB 先发）+ 第 9 个数据位 + 停止
3 WAKE	接收器唤醒方法选择 — 详情请 14.3.3.2，“接收器唤醒操作” 0 闲置线路唤醒 1 地址标记唤醒
2 ILT	闲置线路类型选择 — 将该位设置为 1，确保字符末端的停止位和逻辑 1 位不会计数闲置线路检测逻辑所需的逻辑高电平的 10 或 11 个位时间。如需了解更多信息，14.3.3.2.1，“闲置线路唤醒” 0 开始位后闲置字符位计数开始。 1 停止位后闲置字符位计数开始。

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。 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。

范围内 ($A \leq \text{地址} \leq B$) — 当地址大于或等于比较器 A 的值, 且小于等于比较器 B 的值时, 触发发生。

范围外 ($\text{地址} < A$ 或 $\text{地址} > B$) — 当地址小于比较器 A 的值, 或大于比较器 B 的值时, 触发发生。

17.3.6 硬件断点

DBG 寄存器中的 BRKEN 控制位可以设置为 1, 来允许使用 17.3.5, “触发模式” 所描述的任何触发条件, 向 CPU 生成硬件断点请求。DBG 中的 TAG 控制断点请求是否处理为标记类断点或强制类断点。标记断点使当前的操作码进入指令队列时被标记。如果标记的操作码达到队列的末端, CPU 执行 BGND 指令, 进入激活背景调试模式, 而不是执行被标记的操作码。强制类断点使 CPU 完成当前指令, 然后进入激活背景调试模式。

如果背景调试模式没有被通过 BKGD 管脚的串行 WRITE_CONTROL 命令激活 ($ENBDM = 1$), CPU 将执行 SWI 指令, 而不是进入激活背景调试模式。

17.4 寄存器定义

本小节描述了 BDC 和 DBG 寄存器及控制位。

参见本文的器件概述章节中的 high-page 寄存器一览, 了解所有 DBG 寄存器的绝对地址分配。本小节只按名字参考了寄存器和控制位。使用飞思卡尔提供的等式或头文件, 将这些名称翻译为相应的绝对地址。

17.4.1 BDC 寄存器和控制位

BDC 有两个寄存器:

- 状态和控制寄存器 (BDCSCR) 是一个包含背景调试控制器控制和状态位的 8- 位寄存器。
- BDC 断点匹配寄存器 (BDCBKPT) 拥有一个 16- 位断点匹配地址。

这些寄存器通过专门的串行 BDC 命令接入, 没有位于目标 MCU 的存储器空间中 (因此, 它们没有地址, 用户程序不能接入)。

BDCSCR 中的一些位有写限制, 否则这些寄存器可以随时被读或写。例如, 当 MCU 处于激活背景调试模式中时, ENBDM 控制位不能被写。(这防止了在 MCU 已经处于激活后台模式时, 禁止激活后台模式的控制位的模糊条件) 而且, 有四个状态位 (BDMACT, WS, WSF, 和 DVF) 是只读状态指示符, 永远也不能被 WRITE_CONTROL 串行 BDC 命令写。时钟开关 (CLKSW) 控制位随时都可读或写。

17.4.1.1 BDC 状态和控制寄存器 (BDCSCR)

这个寄存器可以被串行 BDC 命令 (READ_STATUS 和 WRITE_CONTROL) 读或写, 但是用户程序不能存取它, 因为它不位于 MCU 的正常的存储器映射空间中。