AVP32F08 数字信号处理器

数据手册

编号：JXDZ7.381.022 SJSC

Advancechip Electronics

湖南进芯电子科技有限公司

2023 年 08 月

V2.5

目 次

图表目次

1 产品特征

- 高性能静态 CMOS 技术
  - 高达 150MHz（6.67ns 周期时间）
  - 1.8V 转 1.5V 内核，3.3V I/O 设计
- 高性能 32 位 CPU
  - IEEE-754 单精度浮点单元（FPU）
  - 16 x 16 位和 32 x 32 位乘累加 (MAC) 运算
  - 16 x 16 双 MAC
  - 哈佛 (Harvard) 总线架构
  - 快速中断响应和处理
  - 统一存储器编程模型
  - 高效代码（使用 C/C++ 和汇编语言）
- 6 通道 DMA 处理器（用于 ADC，McBSP，ePWM，和 SARAM）
- 片载存储器
  - 256K×16 位闪存，34K×16 位 SARAM
  - 1K x 16 位 一次性可编程 (OTP) ROM
- 引导 ROM (8K×16 位)
  - 支持软件引导模式（通过 SCI，SPI，CAN，IIC，McBSP 和并行 I/O）
  - 标准数学表
- 时钟和系统控制
  - 支持动态锁相环 (PLL) 比率变化
  - 集成上电和欠压复位源
  - 片载振荡器
  - 看门狗定时器模块
- GPIO0 到 GPIO34 引脚可以连接到八个外部内核中断其中的一个
- 可支持全部 56 个外设中断的外设中断扩展(PIE)块
- 128 位安全密钥/锁
  - 保护闪存 / OTP/RAM 模块
  - 防止固件逆向工程
- 低功耗模式和省电模式
  - 支持 IDLE（空闲）、STANDBY（待机）、HALT（暂停）模式
- 字节序：小端序
- ESD 等级(HBM)：2000V
- MSL 湿敏等级：3
- 增强型控制外设
  - 多达 16 个脉宽调制 (PWM) 输出
  - 高达 6 个支持 150ps 微边界定位 (MEP) 分辨率的高分辨率脉宽调制器 (HRPWM) 输出
  - 高达 6 个事件捕获器输入
  - 多达 2 个正交编码器接口
  - 8 个 32 位定时器（6 个 eCAP，2 个 eQEP）
  - 9 个 32 位定时器（6 个 ePWM，3 个 XINTCTR）
- 三个 32 位 CPU 定时器
- 串行端口外设
  - 2 个控制器局域网 (CAN) 模块
  - 2 个 SCI (UART) 模块
  - 2 个 McBSP 模块（可配置为 SPI）
  - 1 个 SPI 模块
  - 1 个内部集成电路 (IIC) 总线
- 12 位模数转换器 (ADC)，16 个通道
  - 12 位 — 160ns（Max） 转换率
  - 2 x 8 通道输入复用器
  - 两个采样保持
  - 顺序/同步转换
  - 内部或者外部基准
- 多达 35 个具有输入滤波功能可单独编程的多路复用通用输入输出 (GPIO) 引脚
- JTAG 边界扫描支持
- 高级仿真特性
  - 分析和断点功能
  - 硬件实时调试
- 开发支持包括
  - ANSI C/C++ 编译器/汇编语言/连接器
  - 编译器
  - 数字电机控制和数字电源软件库
- 产品型号
  - AVP32F08QP100Q（通过 AEC-Q100 认证）
  - AVP32F08QP100S
  - AVP32F08QP100S1（筛选）

2 AVP32F08 简介

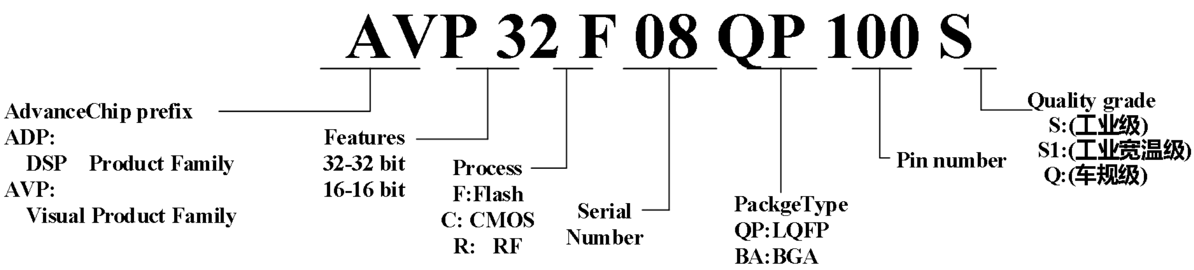
2.1 器件特性

表 2-1 AVP32F08 器件的特性

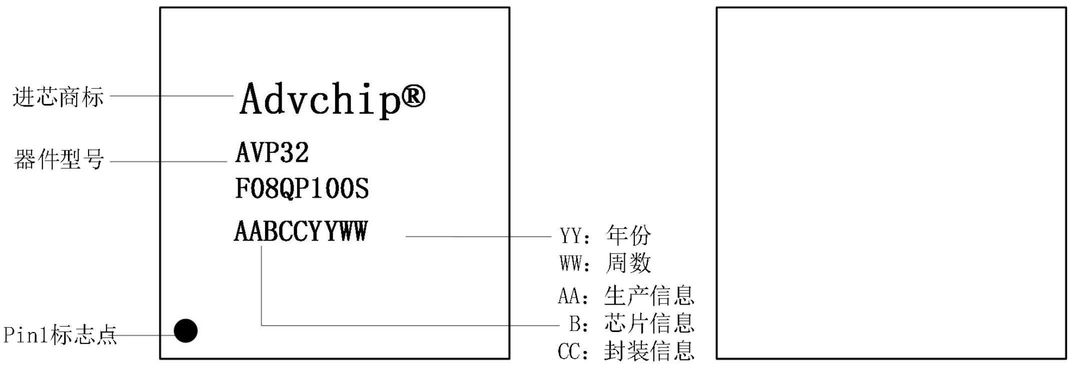
<table>
<tr><th colspan="3">特性</th><th>AVP32F08 (150MHz)</th></tr>
<tr><td colspan="3">指令周期</td><td>6.67ns</td></tr>
<tr><td colspan="3">浮点单元</td><td>支持</td></tr>
<tr><td colspan="3">3.3V 片载闪存（16 位字)</td><td>256K</td></tr>
<tr><td colspan="3">单周期访问RAM (SARAM)（16 位字)</td><td>34K</td></tr>
<tr><td colspan="3">一次性可编程(OTP) ROM（16 位字)</td><td>1K</td></tr>
<tr><td colspan="3">针对片载闪存/ SARAM/OTP 块的代码安全</td><td>支持</td></tr>
<tr><td colspan="3">引导ROM (8K X 16)</td><td>支持</td></tr>
<tr><td colspan="3">6 通道直接内存存取(DMA)</td><td>支持</td></tr>
<tr><td colspan="3">PWM 输出</td><td>ePWM1/2/3/4/5/6</td></tr>
<tr><td colspan="3">HRPWM 通道</td><td>ePWM1A/2A/3A/4A/5A/6A</td></tr>
<tr><td colspan="3">32 位CAPTURE 输入或者辅助PWM 输出</td><td>eCAP1/2/3/4/5/6</td></tr>
<tr><td colspan="3">32 位正交编码器脉冲(QEP) 通道（四个输入/通道)</td><td>eQEP1/2</td></tr>
<tr><td colspan="3">看门狗定时器</td><td>支持</td></tr>
<tr><td rowspan="3">12 位模数转换器
(ADC)</td><td colspan="2">通道的数量</td><td>16</td></tr>
<tr><td colspan="2">MSPS</td><td>6.25</td></tr>
<tr><td>转换时间</td><td>12 位</td><td>160ns</td></tr>
<tr><td colspan="3">32 位CPU 定时器</td><td>3</td></tr>
<tr><td colspan="3">多通道缓冲串行端口(McBSP)/ SPI</td><td>2(A/B)</td></tr>
<tr><td colspan="3">串行外设接口(SPI)</td><td>1</td></tr>
<tr><td colspan="3">串行通信接口(SCI)</td><td>2(A/B)</td></tr>
<tr><td colspan="3">增强型控制器局域网络(eCAN)</td><td>2(A/B)</td></tr>
<tr><td colspan="3">内部集成电路(IIC)</td><td>1</td></tr>
<tr><td colspan="3">通用I/O 引脚（共享)</td><td>35</td></tr>
<tr><td colspan="3">外部中断</td><td>8</td></tr>
<tr><td>封装</td><td colspan="2">LQFP100</td><td>支持</td></tr>
<tr><td rowspan="3">温度选项</td><td colspan="2">S：-40℃至 125℃</td><td>LQFP100</td></tr>
<tr><td colspan="2">S1：-55℃至 125℃</td><td>LQFP100</td></tr>
<tr><td colspan="2">Q：-40℃ 至 125℃（通过 AEC-Q100 认证)</td><td>LQFP100</td></tr>
</table>

2.2 产品编码

AVP32F08 产品代号编码规则:



器件标识:



器件正面　　　　器件背面

2.3 引脚分配

图 2-1 所示为 100 引脚薄型四方扁平封装（LQFP）引脚分配。表 2-2 说明了每个引脚的功能。

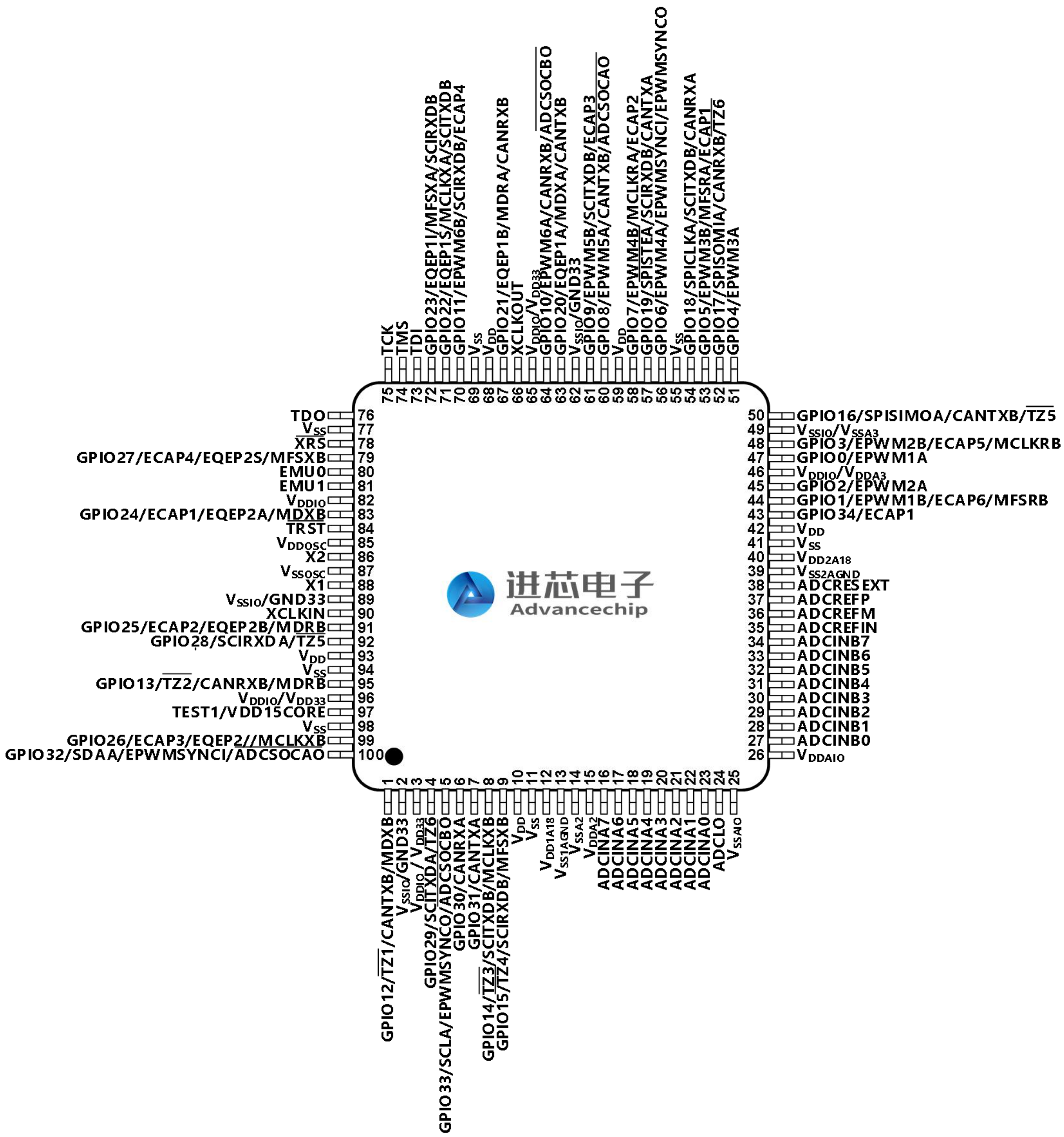


图 2-1 AVP32F08 100 引脚 QP（LQFP 顶视图）

2.4 信号说明

表 2-2 对这些信号进行了说明。GPIO 功能在复位阶段是默认值，其下所列出的外设信号为复用功能。输入不是 5V 耐压。所有 GPIO 引脚为 I/O/Z 且有一个内部上拉电阻，此内部上拉电阻可在每个引脚上有选择性的启用/禁用（**GPIO16 不支持关闭上拉功能**）。这一特性只适用于 GPIO 引脚。器件复位时，GPIO0-GPIO11 引脚上的上拉特性默认禁用，GPIO12-GPIO34 引脚上的上拉特性则默认开启。

表 2-2 信号说明(1)

名称	引脚编号 LQFP100	说明
		JTAG
$\overline{\text{TRST}}$	84	使用内部下拉电阻进行 JTAG 测试复位。当被驱动为高电平时，$\overline{\text{TRST}}$使扫描系统获得器件运行的控制权。如果这个信号未连接或者被驱动至低电平，此器件在功能模式下运转，并且测试复位信号被忽略。注释：$\overline{\text{TRST}}$是一个高电平有效测试引脚并且必须在正常器件运行期间一直保持低电平。在这个引脚上需要一个外部下拉电阻器。这个电阻器的值应该基于适用于这个设计的调试器推进源代码的驱动强度。通常一个 2.2kΩ 电阻器可提供足够的保护。由于这是特定于应用的，建议针对调试器和应用的正确运行对每个目标板进行验证。(I，↓)
TCK	75	带有内部上拉电阻 (I，↑) 的 JTAG 测试时钟
TMS	74	带有内部上拉电阻器的 JTAG 测试模式选择 (TMS)。这个串行控制输入在 TCK 上升沿上的 TAP 控制器中计时。(I，↑)
TDI	73	带有内部上拉电阻的 JTAG 测试数据输入 (TDI)。TDI 在 TCK 的上升沿上所选择的寄存器（指令或者数据）内计时。(I, ↑)
TDO	76	JTAG 扫描输出，测试数据输出 (TDO)。所选寄存器（指令或者数据）的内容被从 TCK 下降沿上的 TDO 移出。（O/Z 8mA 驱动)
EMU0	80	仿真器引脚 0。当$\overline{\text{TRST}}$被驱动至高电平时，这个引脚被用作一个到（或者来自）仿真器系统的中断并且在 JTAG 扫描过程中被定义为输入/输出。这个引脚也被用于将器件置于边界扫描模式中。在 EMU0 引脚处于逻辑高电平状态并且 EMU1 引脚处于逻辑低电平状态时，$\overline{\text{TRST}}$引脚的上升沿将把器件锁存在边界扫描模式。（I/O/Z，8mA 驱动强度↑）请注意：建议在这个引脚上连接一个外部上拉电阻器。这个电阻器的值应该基于适用于这个设计的调试器推进源代码的驱动强度。通常一个 2.2kΩ 至 4.7kΩ 的电阻器已可以满足要求。由于这是特定于应用的，建议针对调试器和应用的正确运行对每个目标板进行验证。
EMU1	81	仿真器引脚 1。当$\overline{\text{TRST}}$被驱动至高电平时，这个引脚被用作一个到（或者来自）仿真器系统的中断并且在 JTAG 扫描过程中被定义为输入/输出。 这个引脚也被用于将器件置于边界扫描模式中。 在 EMU0 引脚处于逻辑高电平状态并且 EMU1 引脚处于逻辑低电平状态时，$\overline{\text{TRST}}$引脚的上升沿将把器件锁存在边界扫描模式。（I/O/Z，8mA 驱动强度↑）请注意：建议在这个引脚上连接一个外部上拉电阻器。 这个电阻器的值应该基于适用于这个设计的调试器推进源代码的驱动强度。通常一个 2.2kΩ 至 4.7kΩ 的电阻器已可以满足要求。由于这是特定于应用的，建议针对调试器和应用的正确运行对每个目标板进行验证。
		其他
TEST1/ VDD15CORE	97	芯片测试端口，内核供电电压输出；需保持悬空
		时钟
XCLKOUT	66	取自 SYSCLKOUT 的输出时钟。XCLKOUT 频率或者与 SYSCLKOUT 的频率相同，或者为后者的一半或四分之一。这是由位 18：16（XTIMCLK）和在 XINTCNF2 寄存器中的位 2（CLKMODE）控制的。复位时，XCLKOUT=SYSCLKOUT/4。通过将 XINTCNF2 [CLKOFF]设定为 1，可关闭 XCLKOUT 信号。与其它 GPIO 引脚不同，复位时，不将 XCLKOUT 引脚置于一个高阻抗状态。（O/Z，8mA 驱动)
XCLKIN	90	外部振荡器输入。这个引脚被用于从一个外部 3.3V 振荡器反馈入一个时钟。在这种情况下，X1 引脚必须连接到 GND。如果使用到了晶振/谐振器（或 1.8V 外部振荡器被用来把时钟馈入 X1 引脚），此引脚必须连接到 GND。(I)

名称	引脚编号 LQFP100	说明
X1	88	内部/外部振荡器输入。为了使用这个振荡器，一个石英晶振或者一个陶瓷电容器必须被连接在 X1 和 X2 上。X1 引脚以 1.8V 内核数字电源为基准。一个 1.8V 外部振荡器也可被连接至 X1 引脚。在这种情况下，XCLKIN 引脚必须接地。如果一个 3.3V 外部振荡器与 XCLKIN 引脚一起使用的话，X1 必须接至 GND。(I)
X2	86	内部振荡器输出。可将一个石英晶振或者一个陶瓷电容器连接在 X1 和 X2。如果 X2 未使用，必须使其保持未连接状态。(O)
复位		
$\overline{XRS}$	78	器件复位（输入）和看门狗复位（输出）。 器件复位。$\overline{XRS}$导致器件终止执行。PC 将指向包含在位置 0x3FFFC0 中的地址。当$\overline{XRS}$被置为高电平时，在 PC 指向的位置开始执行。当一个看门狗复位发生时，这个引脚被 DSP 驱动至低电平。看门狗复位期间，在 512 个 OSCCLK 周期的看门狗复位持续时间内，$\overline{XRS}$引脚被驱动为低电平。(I/OD，↑)这个引脚的输出缓冲器是一个有内部上拉电阻的开漏器件。建议由一个开漏器件驱动这个引脚。在$\overline{XRS}$和 V_{DDIO}之间放置一个值为 2.2 kΩ至 10 kΩ的电阻。如果在$\overline{XRS}$和 V_{SS}之间放置一个电容进行噪声滤波，建议是 100 nF 或更小。
ADC 信号		
ADCINA7	16	ADC A 组，通道 7 输入 (I)
ADCINA6	17	ADC A 组，通道 6 输入 (I)
ADCINA5	18	ADC A 组，通道 5 输入 (I)
ADCINA4	19	ADC A 组，通道 4 输入 (I)
ADCINA3	20	ADC A 组，通道 3 输入 (I)
ADCINA2	21	ADC A 组，通道 2 输入 (I)
ADCINA1	22	ADC A 组，通道 1 输入 (I)
ADCINA0	23	ADC A 组，通道 0 输入 (I)
ADCINB7	34	ADC B 组，通道 7 输入 (I)
ADCINB6	33	ADC B 组，通道 6 输入 (I)
ADCINB5	32	ADC B 组，通道 5 输入 (I)
ADCINB4	31	ADC B 组，通道 4 输入 (I)
ADCINB3	30	ADC B 组，通道 3 输入 (I)
ADCINB2	29	ADC B 组，通道 2 输入 (I)
ADCINB1	28	ADC B 组，通道 1 输入 (I)
ADCINB0	27	ADC B 组，通道 0 输入 (I)
ADCLO	24	ADC 输入信号的参考地（连接至模拟地）(I)
ADCRESEXT	38	ADC 外部电流偏置电阻器。将一个 22kΩ 电阻器接至模拟地。
ADCREFIN	35	外部基准输入 (I)
ADCREFP	37	内部基准参考正端。要求将两个低等效串联电阻 (ESR)（低于 1.5Ω）的陶瓷旁路电容器 10μF 和 0.1μF 并联接至模拟地。(O)
ADCREFM	36	内部基准参考负端。要求将两个低等效串联电阻 (ESR)（低于 1.5Ω）的陶瓷旁路电容器 10μF 和 0.1μF 并联接至模拟地。(O)
供电及接地		
V_{DDA2}	15	ADC 模拟电源引脚
V_{SSA2}	14	ADC 模拟地引脚
V_{DDAIO}	26	ADC 模拟 I/O 电源引脚
V_{SSAIO}	25	ADC 模拟 I/O 接地引脚
V_{DD1A18}	12	ADC 模拟电源引脚
$V_{SS1AGND}$	13	ADC 模拟地引脚
V_{DD2A18}	40	ADC 模拟电源引脚
$V_{SS2AGND}$	39	ADC 模拟地引脚
V_{DD}	10	内部 1.8V 转 1.5V LDO 输入电源（与下页 93 引脚同属性）
V_{DD}	42	
V_{DD}	59	
V_{DD}	68	

名称	引脚编号 LQFP100	说明
V_{DD}	93	
V_{DDIO}/V_{DD33}	3	3.3V 数字 IO 输出驱动电源/3.3V 数字 IO 输入电源
V_{DDIO}/V_{DD33}	65	
V_{DDIO}/V_{DD33}	96	
V_{DDIO}/V_{DDA3}	46	3.3V 数字 IO 输出驱动电源/3.3V ADC 时钟驱动电源
V_{DDOSC}	85	1.8V 晶体振荡器电源
V_{DDIO}	82	3.3V 数字 IO 输出驱动电源
V_{SSIO}/GND33	2	3.3V 数字 IO 输出驱动地/3.3V 数字 IO 输入地
V_{SSIO}/GND33	62	
V_{SSIO}/GND33	89	
V_{SSIO}/V_{SSA3}	49	3.3V 数字 IO 输出驱动地/3.3V ADC 时钟驱动地
V_{SSOSC}	87	1.8V 晶体振荡器地
V_{SS}	11	1.5V 数字内核地
V_{SS}	41	
V_{SS}	55	
V_{SS}	69	
V_{SS}	77	
V_{SS}	94	
V_{SS}	98	
GPIO 和外设信号		
GPIO0 EPWM1A	47	通用输入/输出 0 (I/O/Z) 增强型 PWM1 输出 A 和 HRPWM 通道 (O)
GPIO1 EPWM1B ECAP6 MFSRB	44	通用输入/输出 1 (I/O/Z) 增强型 PWM1 输出 B (O) 增强型捕获器 6 输入/输出 (I/O) McBSP-B 接收帧同步 (I/O)
GPIO2 EPWM2A	45	通用输入/输出 2 (I/O/Z) 增强型 PWM2 输出 A 和 HRPWM 通道 (O)
GPIO3 EPWM2B ECAP5 MCLKRB	48	通用输入/输出 3 (I/O/Z) 增强型 PWM2 输出 B (O) 增强型捕获器 5 输入/输出 (I/O) McBSP-B 接收时钟 (I/O)
GPIO4 EPWM3A	51	通用输入/输出 4 (I/O/Z) 增强型 PWM3 输出 A 和 HRPWM 通道 (O)
GPIO5 EPWM3B MFSRA ECAP1	53	通用输入/输出 5 (I/O/Z) 增强型 PWM3 输出 B (O) McBSP-B 接收帧同步 (I / O) 增强型捕获器输入/输出 1 (I / O)
GPIO6 EPWM4A EPWMSYNCI EPWMSYNCO	56	通用输入/输出 6(I/O/Z) 增强型 PWM4 输出 A 和 HRPWM 通道 (O) 外部 ePWM 同步脉冲输入(I) 外部 ePWM 同步脉冲输出(O)
GPIO7 EPWM4B MCLKRA ECAP2	58	通用输入/输出 7 (I/O/Z) 增强型 PWM4 输出 B (O) McBSP-B 接收时钟 (I / O) 增强型捕获器输入/输出 2 (I / O)
GPIO8 EPWM5A CANTXB $\overline{ADCSOCAO}$	60	通用输入/输出 8 (I/O/Z) 增强型 PWM5 输出 A 和 HRPWM 通道 (O) 增强型 CAN-B 发送 (O) ADC 转换启动 A (O)

名称	引脚编号 LQFP100	说明
GPIO9 EPWM5B SCITXDB ECAP3	61	通用输入/输出 9 (I/O/Z) 增强型 PWM5 输 出 B (O) SCI-B 发送数据 (I/O) 增强型捕获器输入/输出 3 (I/O)
GPIO10 EPWM6A CANRXB $\overline{\text{ADCSOCBO}}$	64	通用输入/输出 10 (I/O/Z) 增强型 PWM6 输出 A 和 HRPWM 通道 (O) 增强型 CAN-B 接收 (I) ADC 转换启动 B (O)
GPIO11 EPWM6B SCIRXDB ECAP4	70	通用输入/输出 11 (I/O/Z) 增强型 PWM6 输出 B (O) SCI-B 接收数据 (I) 增强型 CAP 输入/输出 4 (I/O)
GPIO12 $\overline{\text{TZ1}}$ CANTXB MDXB	1	通用输入/输出 12 (I/O/Z) 触发区输入 1 (I) 增强型 CAN-B 发送 (O) McBSP-B 发送串行数据 (O)
GPIO13 $\overline{\text{TZ2}}$ CANRXB MDRB	95	通用输入/输出 13 (I/O/Z) 触发区输入 2 (I) 增强型 CAN-B 接收 (I) McBSP-B 接收串行数据 (I)
GPIO14 $\overline{\text{TZ3}}$ SCITXDB MCLKXB	8	通用输入/输出 14 (I/O/Z) 触发区输入 3 SCI-B 发送 (O) McBSP-B 发送时钟 (I/O)
GPIO15 $\overline{\text{TZ4}}$ SCIRXDB MFSXB	9	通用输入/输出 15 (I/O/Z) 触发区输入 4。 SCI-B 接收 (I) McBSP-B 发送帧同步 (I/O)
GPIO16 SPISIMOA CANTXB $\overline{\text{TZ5}}$	50	通用输入/输出 16 (I/O) SPI 从器件输入，主器件输出(I/O) 增强型 CAN-B 发送 (O) 触发区输入 5 (I)
GPIO17 SPISOMIA CANRXB $\overline{\text{TZ6}}$	52	通用输入/输出 17 (I/O/Z) SPI-A 从器件输出，主器件输入(I/O) 增强型 CAN-B 接收 (I) 触发区输入 6 (I)
GPIO18 SPICLKA SCITXDB CANRXA	54	通用输入/输出 18 (I/O/Z) SPI-A 时钟输入/输出 (I/O) SCI-B 发送(O) 增强型 CAN-A 接收 (I)
GPIO19 $\overline{\text{SPISTEA}}$ SCIRXDB CANTXA	57	通用输入/输出 19 (I/O/Z) SPI-A 从器件发送使能(I/O) SCI-B 接收 (I) 增强型 CAN-A 发送 (O)
GPIO20 EQEP1A MDXA CANTXB	63	通用输入/输出 20 (I/O/Z) 增 强 型 QEP1 输 入 A (I) McBSP-A 发送串行数据 (O) 增强型 CAN-B 传输 (O)
GPIO21 EQEP1B MDRA CANRXB	67	通用输入/输出 21 (I/O/Z) 增强型 QEP1 输入 B (I) McBSP-A 接收串行数据 (I) 增强型 CAN-B 接收 (I)
GPIO22 EQEP1S MCLKXA SCITXDB	71	通用输入/输出 22 (I/O/Z) 增强型 QEP1 选通脉冲 (I/O) McBSP-A 发送时钟 (I/O) SCI-B 发送 (O)

名称	引脚编号 LQFP100	说明
GPIO23 EQEP1I MFSXA SCIRXDB	72	通用输入/输出 23 (I/O/Z) 增强型 QEP1 索引 (I/O) McBSP-A 发送帧同步 (I/O) SCI-B 接收 (I)
GPIO24 ECAP1 EQEP2A MDXB	83	通用输入/输出 24 (I/O/Z) 增强型捕获 1 (I/O) 增强型 QEP2 输入 A (I) McBSP-B 发送串行数据（O）
GPIO25 ECAP2 EQEP2B MDRB	91	通用输入/输出 25 (I/O/Z) 增强型捕获 2 (I/O) 增强型 QEP2 输入 B (I) McBSP-B 接收串行数据（I）
GPIO26 ECAP3 EQEP2I MCLKXB	99	通用输入/输出 26 (I/O/Z) 增强型捕获 3 (I/O) 增强型 QEP2 索引（I/O) McBSP-B 发送时钟 (I/O)
GPIO27 ECAP4 EQEP2S MFSXB	79	通用输入/输出 27 (I/O/Z) 增强型捕获 4 (I/O) 增强型 QEP2 选通脉冲（I/O) McBSP-B 传输帧同步（I/O)
GPIO28 SCIRXDA $\overline{\text{TZ5}}$	92	通用输入/输出 28 (I/O/Z) SCI 接收数据 (I) 触发区输入 5 (I)
GPIO29 SCITXDA $\overline{\text{TZ6}}$	4	通用输入/输出 29 (I/O/Z) SCI 发送数据 (O) 触发区输入 6 (I)
GPIO30 CANRXA	6	通用输入/输出 30 (I/O/Z) 增强型 CAN-A 接收 (I)
GPIO31 CANTXA	7	通用输入/输出 31 (I/O/Z) 增强型 CAN-A 发送 (O)
GPIO32 SDAA EPWMSYNCI $\overline{\text{ADCSOCAO}}$	100	通用输入/输出 32 (I/O/Z) IIC 数据开漏双向端口 (I/OD) 增强型 PWM 外部同步脉冲输入 (I) ADC 转换启动 A(O)
GPIO33 SCLA EPWMSYNCO $\overline{\text{ADCSOCBO}}$	5	通用输入/输出 33 (I/O/Z) IIC 时钟开漏双向端口 (I/OD) 增强型 PWM 外部同步脉冲输出 (O) ADC 转换启动 B(O)
GPIO34/ECAP1	43	通用输入/输出 34 (I/O/Z) 增强型捕获器输入/输出 1 (I/O)

(1)　I = 输入，O = 输出，Z = 高阻抗，OD = 开漏

3 功能概述

3.1 方框图

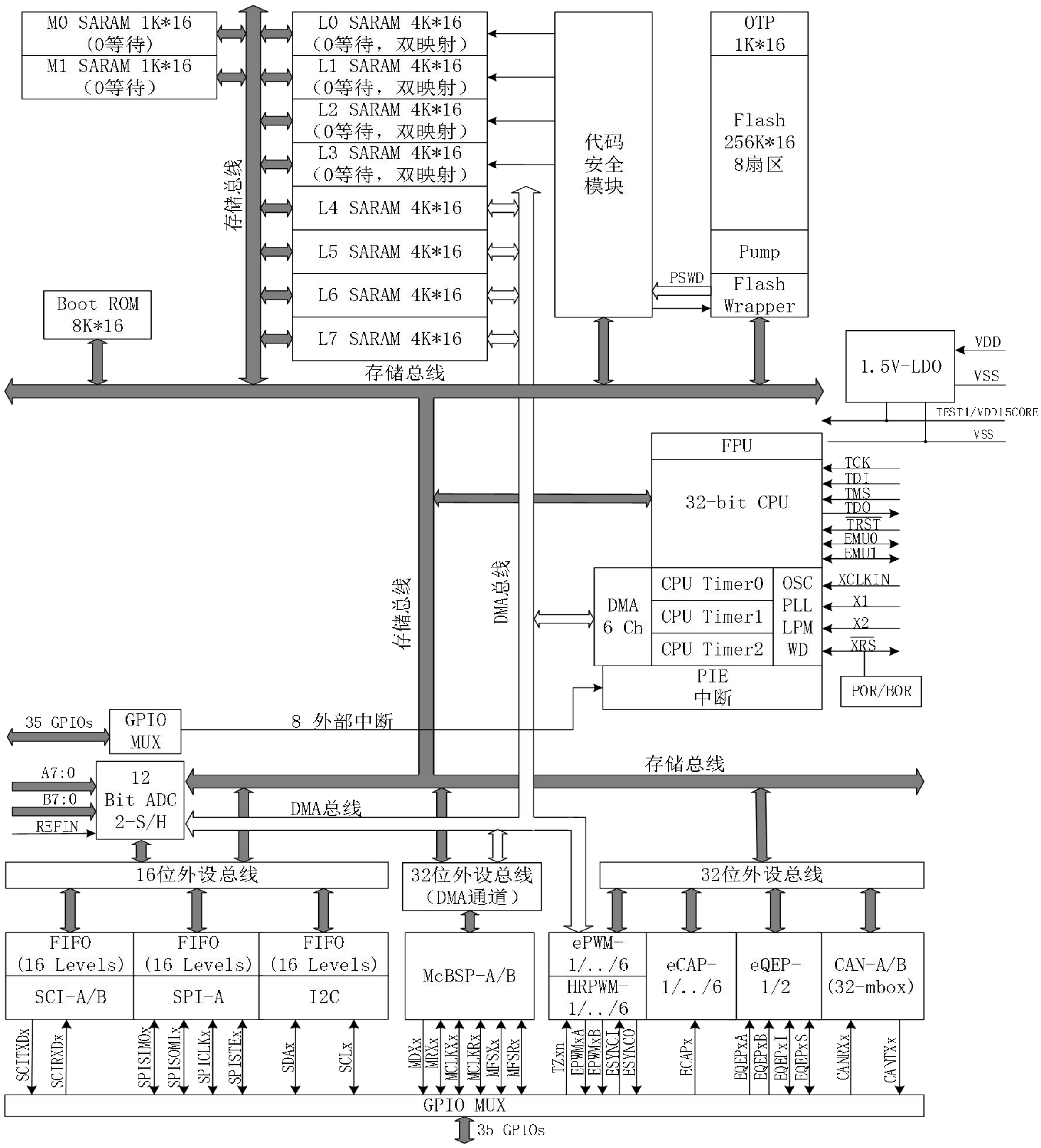


图 3-1 功能方框图

3.2 内存映射

图 3-2 的说明如下：

- 内存块无缩放。
- 外设帧 0、外设帧 1、外设帧 2 和外设帧 3 内存映射只限于数据内存。用户程序不能访问这些处于程序空间内的内存映射。
- “受保护”意思是保护“写后读”的操作顺序，而不是保护流水线顺序。
- 特定内存区域受 EALLOW 保护以防止配置之后的假写入。
- 位置 0x380080-0x38008F 包含 ADC 校准程序，禁止用户编程。
- 如果 eCAN 模块未在应用中使用，提供的 RAM（LAM，MOTS，MOTO 和邮箱 RAM）可被用作通用 RAM。为实现这一功能，应启用 CAN 模块时钟。

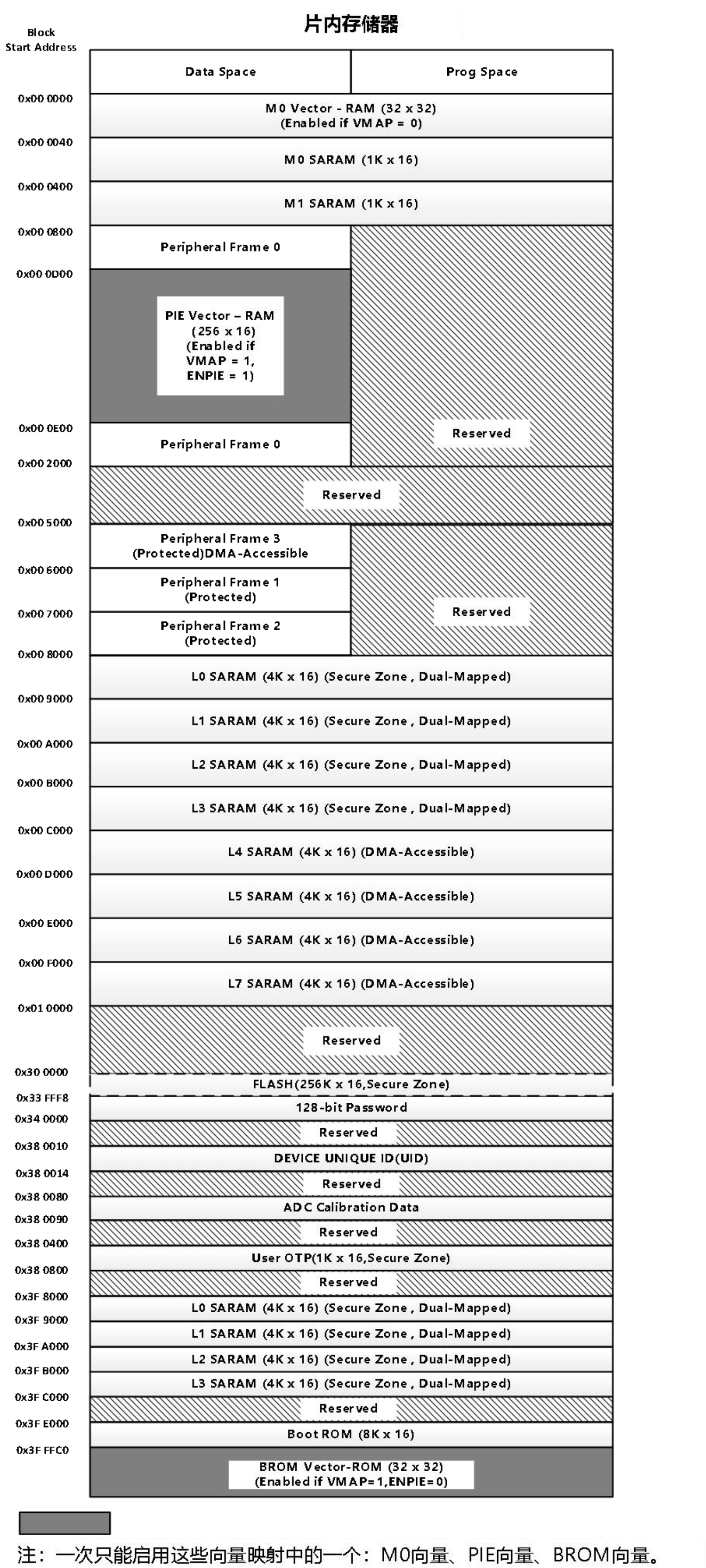


图 3-2 AVP32F08 内存映射

表 3-1 AVP32F08 中闪存扇区的地址

地址范围	程序和数据空间
0x30 0000-0x30 7FFF	扇区 H (32K x 16)
0x30 8000-0x30 FFFF	扇区 G (32K x 16)
0x31 0000-0x31 7FFF	扇区 F (32K x 16)
0x31 8000-0x31 FFFF	扇区 E (32K x 16)
0x32 0000-0x32 7FFF	扇区 D (32K x 16)
0x32 8000-0x32 FFFF	扇区 C (32K x 16)
0x33 0000-0x33 7FFF	扇区 B (32K x 16)
0x33 8000-0x33 FF7F 0x33 FF80-0x33 FFF5 0x33 FFF6-0x33 FFF7 0x33 FFF8-0x33 FFFF	扇区 A (32K x 16) 当使用代码安全模块时，编程至 0x0000 引导至闪存进入点（程序分支指令） 安全密码（128 位）（不要设定为全零）

注

- 当编辑代码安全密码时，0x33FF80 到 0x33FFF5 间的所有地址不能被用作程序代码或者数据。这些位置必须被设定为 0x0000。
- 如果代码安全特性未被使用，地址 0x33FF80 至 0x33FFEF 可被用于代码或者数据。地址 0x33FF0-0x33FF5 为数据保留且不能包含程序代码。

表 3-2 显示了使用安全代码模块的影响及如何处理相应内存地址。

表 3-2 使用安全代码模块的影响

地址范围	闪存	
	启用代码安全	禁用代码安全
0x33FF80-0x33FFEF	用 0x0000 填充	应用代码和数据。
0x33FFF0-0x33FFF5		只为数据保留。

外设帧 1、外设帧 2 以及外设帧 3 组合成为“受保护的写入/读取外设块”。“受保护”模式确保对这些所有的访问与文档中描述的一致。由于 AVP32F08 的流水线，在对不同内存位置读取之前的写入操作，将以倒序的方式出现在 CPU 内存总线上。这会导致特定外设应用中的问题，用户认为写入会首先发生（如文档所描述的那样）。AVP32F08 CPU 支持一个块保护模式，在该模式中，可对一个内存区域进行保护，以确保操作按照本文档所描述的那样进行（代价是会增加额外的周期以校正运行）。可对该模式进行编程，并且默认情况下，它将保护所选的区域。

针对内存映射区域内不同空间的等待状态如表 3-3 所示。

表 3-3 等待状态

<table>
<tr><th>区域 (AREA)</th><th>等待状态(CPU)</th><th>等待状态（DMA）[1]</th><th>注释</th></tr>
<tr><td>M0 和 M1 SARAM</td><td>0 - 等待</td><td></td><td>固定的</td></tr>
<tr><td>外设帧 0</td><td>0 - 等待（写入）
1 – 等待（读取）</td><td>0 - 等待（读取）
无权限（写入）</td><td></td></tr>
<tr><td>外设帧 1</td><td>0 - 等待（写入）
2 - 等待（读取）</td><td>无访问权限</td><td>周期可由已生成的外设扩展。
到外设帧 1 寄存器的背靠背写入操作将生成一个 1 周期停止（1 周期延迟）。</td></tr>
<tr><td>外设帧 2</td><td>0 - 等待（写入）
2 - 写入（读取）</td><td>无访问权限</td><td>固定的周期不可由外设扩展。</td></tr>
<tr><td>外设帧 3</td><td>0 - 等待（写入）
2 - 等待（读取）</td><td>0 - 等待（写入）
1 – 等待（读取）</td><td>假定 CPU 和 CLA 间无冲突。</td></tr>
<tr><td>L0 SARAM</td><td rowspan="4">0 - 等待</td><td rowspan="4">无访问权限</td><td rowspan="4">假定没有 CPU 冲突</td></tr>
<tr><td>L1 SARAM</td></tr>
<tr><td>L2 SARAM</td></tr>
<tr><td>L3 SARAM</td></tr>
<tr><td>L4 SARAM</td><td rowspan="4">0 - 等待数据（读取）
0 - 等待数据（写入）
1 - 等待项目（读取）
1 - 等待项目（写入）</td><td rowspan="4">0 - 等待</td><td rowspan="4">假设 CPU 和 DMA 之间没有冲突。</td></tr>
<tr><td>L5 SARAM</td></tr>
<tr><td>L6 SARAM</td></tr>
<tr><td>L7 SARAM</td></tr>
<tr><td>OTP</td><td>可编程
1 - 等待最小</td><td>无访问权限</td><td>由闪存寄存器设定。
1 - 等待是等待状态所允许的最小数。可在一个减少的 CPU 频率上执行 1 等待状态操作。</td></tr>
<tr><td>闪存</td><td>可编程
0 - 页式等待最小值
1 - 随机等待最小值
随机等待≥页式等待</td><td>无访问权限</td><td>由闪存寄存器设定。
页式访问中不允许 0 - 等待最小</td></tr>
<tr><td>闪存密码</td><td>16=等待固定</td><td>无访问权限</td><td>密码位置的等待状态是固定的。</td></tr>
<tr><td>引导 - ROM</td><td>1 – 等待</td><td>无访问权限</td><td>0 - 等待速度无法实现。</td></tr>
</table>

3.3 简要说明

3.3.1 AVP32F08 DSP

AVP32F08 是基于 ADP32Fx+FPU 的控制器和 Advchip 现有的 ADP32Fx DSP 具有相同的 32 位定点架构，同时还包括一个单精度（32 位）的 IEEE754 浮点单元（FPU）。这是一个非常高效的 C/C++引擎，它能使用户用高层次的语言开发他们的系统控制软件，能够使用 C/C++开发算术算法。此器件同样可以处理原由微控制器处理的系统控制任务，故在处理 DSP 算术任务时和处理系统控制任务时同样有效。此高效率处理可以节省很多系统对第二个处理器的需求。内置的 32x32 位 MAC64 位处理能力使得控制器能够有效地处理更高的数字分辨率运算问题。带有关键寄存器自动环境保存的快速中断响应，能够让一个器件用最小的延迟处理很多异步事件。还内置有一个具有流水线式存储器访问的 8 级深度受保护的流水线。该流水线式操作使得此器件能够在高速执行而无需求助于昂贵的高速存储器。特有的分支超前硬件显著减少了条件不连续而带来的延迟。特别是存储容量的增大进一步提升了性能。

3.3.2 内存总线（哈弗总线架构）

与很多 DSP 类型的器件一样，多总线被用于在内存和外设以及 CPU 之间传输数据。AVP32F08 内存总线架构包含：程序读取总线、数据读取总线和数据写入总线。此程序读取总线由 22 条地址线路和 32 条数据线路组成。数据读取和写入总线由 32 条地址线路和 32 条数据线路组成。32 位宽数据总线可实现单周期 32 位运行。多总线结构通常称为“哈弗总线”，使得 AVP32F08 能够在一个单周期内取一个指令、读取一个数据值和写入一个数据值。所有连接在内存总线上的外设和内存对内存访问进行优先级设定。总的来说，内存总线访问的优先级可概括如下：

最高级： 数据写入　　（内存总线上不能同时进行数据和程序写入。）

程序写入　　（内存总线上不能同时进行数据和程序写入。）

数据读取

程序读取　　（内存总线上不能同时进行程序读取和取指令。）

最低级： 取指令　　（内存总线上不能同时进行程序读取和取指令。）

3.3.3 外设总线

进芯电子（Advchip）AVP32F08 器件采用了一个针对外设互连的外设总线标准。外设总线桥复用了多种总线，此总线将处理器“内存总线”组装进一个由 16 条地址线路和 16 条或者 32 条数据线路以及相关控制信号组成的单总线中。在 AVP32F08 器件上支持三种类型的外设总线。第一种只支持 16 位访问（被称为外设帧 2）并且保留了与 16 位外设的兼容性。第二种支持 16 位和 32 位访问（被称为外设帧 1）。第三种支持 DMA 访问和 16 位以及 32 位访问（被称为外设帧 3）。

3.3.4 实时 JTAG 和分析

AVP32F08 器件使用标准的 IEEE1149.1 JTAG 接口。该器件支持实时运行模式，在处理器正在运行、执行代码并且处理中断时，可修改存储器内容、外设和寄存器位置。用户也可以通过不具有时间紧迫性的代码进行单步操作，同时可在没有干扰的情况下启用即将被处理的实时响应中断。此器件在 CPU 的硬件内执行实时模式。是 AVP32F08 器件独特功能，无需软件监控。此外，还提供了特别分析硬件以实现硬件断点或者数据/地址观察点的设置并当一个匹配发生时生成不同的用户可选中断事件。

3.3.5 闪存

AVP32F08 器件包含 256Kx16 位的嵌入式闪存存储器，分属在 8 个 32Kx16 位扇区内。所有器件还包含一个 1Kx16 位的 OTP 内存，其地址范围为 0x380400-0x3807FF。用户能够在不改变其它扇区的同时单独擦除、编辑和验证一个闪存扇区。但是不能使用闪存的一个扇区或者这个 OTP 来执行擦除/编辑其它扇区的闪存算法。闪存模块内置的特殊内存流水线操作具有更高的性能。闪存/OTP 被映射到程序和数据空间，它可被用于执行代码或者存储数据信息。但是地址 0x33FFF0-0x33FFF5 保留为数据变量，且不应包含程序代码。

注

闪存和 OTP 等待状态可由应用配置。这使得运行在较低频率上的应用能够将闪存配置为使用较少的等待状态。

可通过在闪存选项寄存器中启用闪存流水线操作模式来提升闪存的性能。这个模式被启用时，线性代码执行的性能将显著高于只由等待状态配置所表示的原始性能。使用闪存流水线模式的准确性能增加依应用而定。

3.3.6 M0，M1 SARAM

所有 AVP32F08 器件包含 M0 和 M1 两块单周期访问内存，每一块 SARAM 的大小为 1K x 16。复位时堆栈指针指向块 M1 的开始位置。AVP32F08 中的 M0 和 M1 块的内存块被映射到程序和数据空间。用户能够使用 M0 和 M1 来执行代码或者存储数据变量。分区在连接器内执行。AVP32F08 器件针对用户提供了统一内存映射文件，这使得用高级语言编程更加容易。

3.3.7 L0，L1，L2，L3，L4，L5，L6，L7 SARAM

AVP32F08 器件含有 32K x 16 位的单周期访问 RAM，每个被分为 8 个区块（L0–L7 且每块 4K）。可独立访问每个块，显著减少了 CPU 流水线延迟。这些块被映射到程序和数据空间，L4、L5、L6 和 L7 可由 DMA 访问。

3.3.8 引导 ROM

Bootloader 程序出厂时固化在 ROM 存储器中。引导 ROM 程序在器件复位后，检查三个 GPIO 引脚状态后确认芯片进入相应的引导模式。例如，用户可以选择引导到 Flash 执行程序或者通过某个串行端口将程序下载至内部 RAM。引导 ROM 还包含用于数学相关算法中的标准表，例如 SIN/COS 函数。

表 3-4 引导模式选择

模式	GPIO18/SPICLKA	GPIO29/SCITXDA	GPIO34	模式 (1)
7	1	1	1	跳转到闪存
6	1	1	0	SCI-A boot
5	1	0	1	SPI-A boot
4	1	0	0	IIC-A boot
3	0	1	1	ECANA boot
2	0	1	0	Boot to M0-SARAM
1	0	0	1	Boot to OTP
0	0	0	0	并行引导 GPIO0~GPIO15

(1) 所有的 3 个 GPIO 引脚都有内部上拉电阻。

注

表 3-4 中的 0、1、2 模式只用于 Advchip 调试。应用程序中跳过 ADC 校准功能将导致 ADC 在规定的技术规格之外运行。

3.3.8.1 引导加载器使用的外设引脚

表 3-5 显示了每一个外设引导加载器所使用的 GPIO 引脚。参考 GPIO 复用表以避免与任一外设冲突使用。

表 3-5 外设引导加载引脚

引导加载器	外设加载器引脚
SCI-A	SCIRXDA (GPIO28)　SCITXDA (GPIO29)
SPI-A	SPISIMOA (GPIO16)　SPISOMIA (GPIO17) SPICLKA (GPIO18)　SPISTEA(GPIO19)
IIC	SDAA (GPIO32)　SCLA (GPIO33)
CAN	CANRXA (GPIO30)　CANTXA (GPIO31)
McBSP	MDXA(GPIO20)　MDRA(GPIO21) MCLKXA(GPIO22)　MFSXA(GPIO23) MCLKRA(GPIO7)　MFSRA(GPIO5)

3.3.9 安全性

AVP32F08 支持高级别安全以保护用户固件不受逆向工程损坏。该加密装置有一个 128 位密码

（针对 16 个等待状态的硬编码），此密码由用户编辑写入闪存。一个代码安全模块（CSM）被用于保护闪存/OTP 和 L0/L1/L2/L3 SARAM 块。这个安全特性防止未经授权的用户通过 JTAG 端口检查内存内容，从外部内存执行代码或者试图引导加载一些将会输出安全内存内容的恶意软件。为了启用到安全块的访问，用户必须写入与存储在闪存/ROM 密码位置内的值相匹配的正确的 128 位“KEY（密钥）”值。

除了CSM，还实现了仿真代码安全逻辑电路(ECSL)，以防止未经授权的用户安全代码。在仿真器连接时，任何对于闪存、用户OTP、L0、L1、L2或L3内存的代码或者数据访问将触发ECSL并断开仿真连接。为了实现安全代码仿真，同时保持CSM 安全内存读取，用户必须向KEY 寄存器的低64 位写入正确的值，这个值与存储在闪存密码位置的低64位的值相符合。请注意仍须执行闪存内所有128 位密钥的伪读取。如果密码位置的低64位为全1（未被编辑），则无须符合KEY值。

当对闪存内被编辑的密码区（即安全的）进行最初调试时，CPU 将开始运行并可执行一个指令来访问一个受保护的ECSL区域。如果这一情况发生，ECSL 将发生错误并使仿真器连接被断开。这个问题有两个解决方案：

1. 首先是使用在等待中复位的仿真模式，该模式将保持器件在复位状态直到仿真器获得控制权。仿真器必须支持此选项的这种模式。
2. 第二种选择是使用“分支检查引导模式”引导选项。这将进入一个环路，并不断轮询引导模式选择引脚。通过重新映射PC到另一个地址，或通过把引导模式选择引脚更改为所需的引导模式，用户可以选择此引导模式，然后在连接仿真器探针时退出该模式。

注

- 当代码安全密钥被编辑时，0xFF3380 到 0x33FFF5 间的所有地址不能被用作程序代码或者数据。这些位置必须被设定为 0x0000
- 如果代码安全特性未被使用，地址 0x33FF80 至 0x33FFEF 可被用于代码或者数据。地址 0x33FFF0-0x33FFF5 为数据保留且不能包含程序代码。
- 128 位密码（位于 0x33FFF8-0x33FFFF）不能写入全零。一旦写入全零将永久锁住此器件，请务必慎重。

代码安全模块免责声明

- 此器件所包含的代码安全模块（CSM）旨在对存储在相关内存（ROM 或者闪存）中的数据进行密码保护并且由原厂提供质量保证，与其标准条款和条件相一致，符合原厂发布的规范以获得适用于本器件的保修期规范。
- 但是，原厂不保证或承诺CSM不会被损坏或破坏，也不保证或承诺在相关存储器中存储的数据不能通过其它方法存取。此外，除上述内容外，原厂也未对本器件的CSM或操作做任何保证或承诺，包括对适销性或特定用途适用性的任何暗示保证。
- 在任何情况下，原厂对以任何方法使用CSM或本器件产生的任何必然、特殊、间接、偶然或严重伤害不负任何责任，无论原厂是否被告知存在这种伤害的可能性。排除的损害包括但不限于数据丢失、信誉损失、无法使用、业务中断或其它经济损失。

3.3.10 外设中断扩展 (PIE) 块

PIE 块将许多中断源复用至中断输入的较小的集合中。PIE 块能够支持多达 96 个可用的外设中断。在 AVP32F08 中，96 个可用的中断中的每一个中断由其存储在一个可被用户写覆盖的专用 RAM 块中的矢量支持。在处理这个中断时，这个矢量由 CPU 自动抽取。抽取这个矢量以及保存关键 CPU 寄存

器将花费 8 个 CPU 时钟周期。因此 CPU 能够对中断事件作出快速响应。可以通过硬件和软件控制中断的优先级。每个中断都可以在 PIE 块内启用或禁用。

3.3.11 外部中断 (XINT1-XINT7，XNMI)

AVP32F08支持8个可屏蔽的外部中断(XINT1-XINT7，XNMI)。XNMI可被连接至INT13或者CPU的NMI中断。这些中断中的每一个都可被选择用于负边沿、正边沿或双沿触发，并且可被启用或禁用（包括XNMI在内）。XINT1、XINT2和XNMI还包含一个16位自由运行的向上计数器，当检测到一个有效的中断边沿时，该计数器复位为0。这个计数器可被用于为中断精确计时。XINT1、XINT2和XINT中断可接受来自GPIO0-GPIO31引脚的输入。XINT3–XINT7中断可接受来自GPIO32-GPIO34引脚的输入。

3.3.12 振荡器和锁相环（PLL）

此器件可由一个外部振荡器计时或者由一个连接到片载振荡器电路的晶振计时。提供的一个 PLL 支持高达 1031 个输入时钟缩放比。PLL 比率可用软件在器件运行时更改，这使得用户在需要低功耗运行时能够按比例降低运行频率。时序细节，请参考电气规范部分。PLL 块可被设定为旁路模式。

3.3.13 看门狗

用户软件必须在特定的周期内定期复位看门狗计数器；否则看门狗将产生一个复位信号复位处理器。如果需要可将看门狗禁用。

3.3.14 外设时钟

在外设闲置时，可以启用/禁用到每一个独立外设的时钟，以减少功耗。此外，串行端口（除了IIC和eCAN）和ADC的系统时钟可按照 CPU 时钟进行缩放。

3.3.15 低功耗模式

AVP32F08 器件是完全静态 CMOS 器件。提供三种低功耗模式：

IDLE： 将 CPU 置于低功耗模式。可有选择性地关闭外设时钟并且只有那些在 IDLE 期间必须运行的外设保持运行状态。来自使能外设的已启用的中断或者看门狗将处理器从 IDLE 模式中唤醒。

STANDBY： 关闭到 CPU 和外设的时钟。在这个模式下振荡器和 PLL 仍然运行。一个外部中断事件将唤醒处理器和外设。唤醒操作在检测到中断事件之后的下一个时钟周期执行。

HALT： 关断内部振荡器的基础上，将器件置于尽可能低的功耗模式中。在此模式下可由一个复位信号或者外部信号将器件从这个模式中唤醒。

3.3.16 外设帧 0，1，2，3 (PFn)

此器件将外设分成四个部分。外设映射如下：

PF0：	PIE：	PIE 中断启用和控制寄存器加上 PIE 矢量表
	闪存：	闪存控制、编程、擦除、验证寄存器
	DMA：	DMA 寄存器
	定时器：	CPU-定时器 0，1，2 寄存器
	CSM：	代码安全模块 KEY 寄存器
	ADC：	ADC 结果寄存器
PF1:	eCAN：	eCAN 邮箱和控制寄存器
	GPIO：	GPIO MUX 配置和控制寄存器
	ePWM：	增强型脉冲宽度调制器模块和寄存器
	eCAP：	增强型捕获模块和寄存器
	eQEP:	增强型正交编码脉冲模块和寄存器
PF2：	SYS：	系统控制寄存器
	SCI：	异步串行通信接口(SCI)控制和 RX/TX 寄存器
	SPI：	同步串行通信接口(SPI)控制和 RX/TX 寄存器
	ADC：	ADC 状态、控制和配置寄存器
	IIC：	集成电路总线模块和寄存器
	XINT：	外部中断寄存器
PF3：	McBSP：	多通道缓冲串行端口寄存器
	ePWM：	增强型脉冲宽度调制器模块和寄存器

3.3.17 通用输入/输出(GPIO)复用器

大多数的外设信号与通用输入/输出(GPIO)信号复用。这使得用户能够在外设信号或者功能不使用时将一个引脚用作 GPIO。复位时，所有 GPIO 引脚被配置为输入。针对 GPIO 模式或者外设信号模式，用户能够独立设定每一个引脚。对于特定的输入引脚，用户也可以选择输入限定周期的数量。这是为了过滤掉有害的噪音毛刺脉冲。GPIO 信号也可被用于使器件脱离特定低功耗模式。

3.3.18 32 位 CPU 定时器 (0，1，2)

CPU 定时器 0、1 和 2 是完全一样的 32 位定时器，这些定时器带有可预先设定的周期和 16 位时钟预分频。此定时器有一个 32 位倒计数寄存器，此寄存器在计数器达到 0 时生成一个中断。定时器的计数器在一个时钟频率下递减计数，此时钟频率是 CPU 时钟经过分频而来（此分频的值是通过寄存器设置的）。当此计数器达到 0 时，它自动重新载入一个 32 位的周期值。CPU 定时器 2 为 DSP/BIOS 预留，并且连接到 CPU 的 INT14。如果 DSP/BIOS 未被使用，CPU 定时器 2 也可作为通用定时器使用。CPU 定时器 0 也可作为通用定时器并被连接至 PIE 模块。

3.3.19 控制外设

AVP32F08 支持以下用于嵌入式控制和通信的外设：

ePWM：	增强型 PWM 外设支持针对前缘和后缘边沿、被锁存的以及逐周期触发机制的独立的和互补的 PWM 生成，可调节死区生成。某些 PWM 引脚支持 HRPWM 特性。ePWM 寄存器由 DMA 支持以便减少处理该外设的开销。
eCAP：	此增强型捕获外设使用一个 32 位时基，并在连续/单次捕获模式中记录多达四个可编程事件。 这个外设还可以配置为生成一个辅助 PWM 信号。
eQEP：	增强型 QEP 外设使用一个 32 位位置计数器，使用捕获单元和一个 32 位单元定时器分别支持低速测量和高速测量。 这个外设有一个看门狗定时器来检测电机停转和输入错误检测逻辑电路，以识别 QEP 信号中的同步边沿转换。
ADC：	ADC 模块是一个 12 位、16 通道转换器。它包含两个用于同步采样的采样保持单元。ADC 寄存器支持 DMA 操作，减少处理器的硬件开销。

3.3.20 串行端口外设

此器件支持下列的串行通信外设：

eCAN:	这是 CAN 外设的增强型版本。它支持 32 个邮箱、消息时间戳并与 CAN2.0B 兼容。
McBSP:	多通道缓冲串行端口(McBSP)应用到 E1/T1 线路、语音质量编解码器或高质量立体声音频 DAC 器件。McBSP 接收和发送寄存器由 DMA 支持以显著地减少处理这个外设所用的资源。如果需要，每一个 McBSP 模块可被配置为一个 SPI。
SPI:	SPI 是一个高速、同步串行 I/O 端口，此端口可在设定的位传输速率上将一个设定长度（1 至 16 位）的串行比特流移入和移出器件。通常，SPI 用于 DSP 和外部外设或者其它处理器之间的通信。典型应用包含通过移位寄存器、显示驱动器和 ADC 等器件进行外部 I/O 或外设扩展。多器件通信由 SPI 的主模式/从模式操作支持。在 AVP32F08 上，SPI 包含一个 16 级接收和发送 FIFO 来减少中断处理开销。
SCI:	串行通信接口是一个两线制异步串行端口，通常被称为 UART。SCI 包含一个用于减少中断处理开销的 16 级接收和发送 FIFO。
IIC:	内部集成电路 (IIC) 模块在 MCU 和其它器件（符合飞利浦半导体内部 IC 总线(IIC-bus) 规范版本 2.1 并由一个 IIC-bus 相连）之间提供一个接口。通过这个 IIC 模块，连接在这个两线制总线上的外部组件能够发送 8 位数据到 MCU 或者从 MCU 接收 8 位数据。IIC 包含一个用于减少中断处理开销的 16 级接收和发送 FIFO。

3.4 寄存器映射

此器件包含 4 个外设寄存器空间。这些空间分类如下：

外设帧 0： 这些是直接映射到 CPU 内存总线的外设。请参阅表 3-6。

外设帧 1： 这些是映射到 32 位外设总线的外设。请参阅表 3-7。

外设帧 2： 这些是映射到 16 位外设总线的外设。请参阅表 3-8。

外设帧 3： 这些是映射到 32 位外设总线并可由 DMA 访问的外设。请参阅表 3-9。

表 3-6 外设帧 0 寄存器(1)

名称	地址范围	大小（x16)	受 EALLOW 保护(2)
器件仿真寄存器	0x00 0880-0x00 09FF	384	受EALLOW 保护
闪存寄存器(3)	0x00 0A80-0x00 0ADF	96	受EALLOW 保护
代码安全模块寄存器	0x00 0AE0-0x00 0AEF	16	受EALLOW 保护
ADC 寄存器（双映射） 0 等待(DMA)，1 个等待(CPU)，只读	0x00 0B00-0x00 0B0F	16	不受 EALLOW 保护
CPU 定时器0，CPU 定时器1，CPU 定时器2 寄存器	0x00 0C00-0x00 0C3F	64	不受 EALLOW 保护
PIE 寄存器	0x00 0CE0-0x00 0CFF	32	不受 EALLOW 保护
PIE 矢量表	0x00 0D00-0x00 0DFF	256	受EALLOW 保护
DMA 寄存器	0x00 1000-0x00 11FF	512	受EALLOW 保护

(1) 在帧 0 中的寄存器支持 16 位和 32 位访问。

(2) 如果寄存器是 EALLOW 受保护的，那么在 EALLOW 指令被执行前写入不能被执行。EDIS 指令禁用写入以防止杂散代码或指针破坏寄存器内容。

(3) 闪存寄存器也受到代码安全模块(CSM)的保护。

表 3-7 外设帧 1 寄存器

名称	地址范围	大小（x16)
eCAN-A 寄存器	0x00 6000-0x00 61FF	512
eCAN-B 寄存器	0x00 6200–0x00 63FF	512
ePWM1 + HRPWM1 寄存器	0x00 6800-0x00 683F	64
ePWM2 + HRPWM2 寄存器	0x00 6840-0x00 687F	64
ePWM3 + HRPWM3 寄存器	0x00 6880-0x00 68BF	64
ePWM4 + HRPWM4 寄存器	0x00 68C0-0x00 68FF	64
ePWM5 + HRPWM5 寄存器	0x00 6900-0x00 693F	64
ePWM6 + HRPWM6 寄存器	0x00 6940-0x00 697F	64
eCAP1 寄存器	0x00 6A00-0x00 6A1F	32
eCAP2 寄存器	0x00 6A20-0x00 6A3F	32
eCAP3 寄存器	6x40 6A00-0x00 0A5F	32
eCAP4 寄存器	6x60 6A00-0x00 0A7F	32
eCAP5 寄存器	6x80 6A00-0x00 0A9F	32
eCAP6 寄存器	0x00 6AA0–0x00 6ABF	32
eQEP1 寄存器	0x00 6B00-0x00 6B3F	64
eQEP2 寄存器	0x00 6B40-0x00 6B7F	64
GPIO 寄存器	0x00 6F80-0x00 6FFF	128

表 3-8 外设帧 2 寄存器

名称	地址范围	大小（x16）
系统控制寄存器	0x00 7010-0x00 702F	32
SPI-A 寄存器	0x00 7040-0x00 704F	16
SCI-A 寄存器	0x00 7050-0x00 705F	16
外部中断寄存器	0x00 7070-0x00 707F	16
ADC 寄存器	0x00 7100-0x00 711F	32
SCI-B 寄存器	0x00 7750-0x00 775F	16
IIC-A 寄存器	0x00 7900-0x00 793F	64

表 3-9 外设帧 3 寄存器

名称	地址范围	大小（x16）
McBSP-A 寄存器 (DMA)	0x5000 -0x503F	64
McBSP-B 寄存器 (DMA)	0x5040 -0x507F	64
ePWM1 + HRPWM1 (DMA)(1)	0x5800 -0x583F	64
ePWM2 + HRPWM2 (DMA)	0x5840 -0x587F	64
ePWM3 + HRPWM3 (DMA)	0x5880-0x58BF	64
ePWM4 + HRPWM4 (DMA)	0x58C0-0x58FF	64
ePWM5 + HRPWM5 (DMA)	0x5900 -0x593F	64
ePWM6 + HRPWM6 (DMA)	0x5940 -0x597F	64

(1) EPWM 和 HRPWM 模块可以重新映射到可由 DMA 模块访问的外设帧 3。要做到这点，MAPCNF 寄存器（地址 0x702E）的位 0（MAPEPWM）必须被设置为 1。此寄存器受 EALLOW 保护。当此位为 0 时，ePWM 和 HRPWM 模块映射到外设帧 1。

3.5 器件仿真寄存器

这些寄存器用于控制 AVP32Fx CPU 的保护模式并监控某些关键器件信号。表 3-10 中对这些寄存器进行了定义。

表 3-10 器件仿真寄存器

名称	地址范围	大小（x16）	说明
DEVICECNF	0x0880 0x0881	2	器件配置寄存器
PARTID	0x380090	1	部件 ID 寄存器 0x00EF
CLASSID	0x0882	1	AVP32F08 浮点类设备 0x00EF
REVID	0x0883	1	修订版本 ID 寄存器
PROTSTART	0x0884	1	区块保护起始地址寄存器
PROTRANGE	0x0885	1	区块保护范围地址寄存器
UID	0x380010 0x380011 0x380012 0x380013	4	芯片唯一码（64 位随机值）

3.6 中断

图 3-3 显示了不同的中断源是如何被复用的。

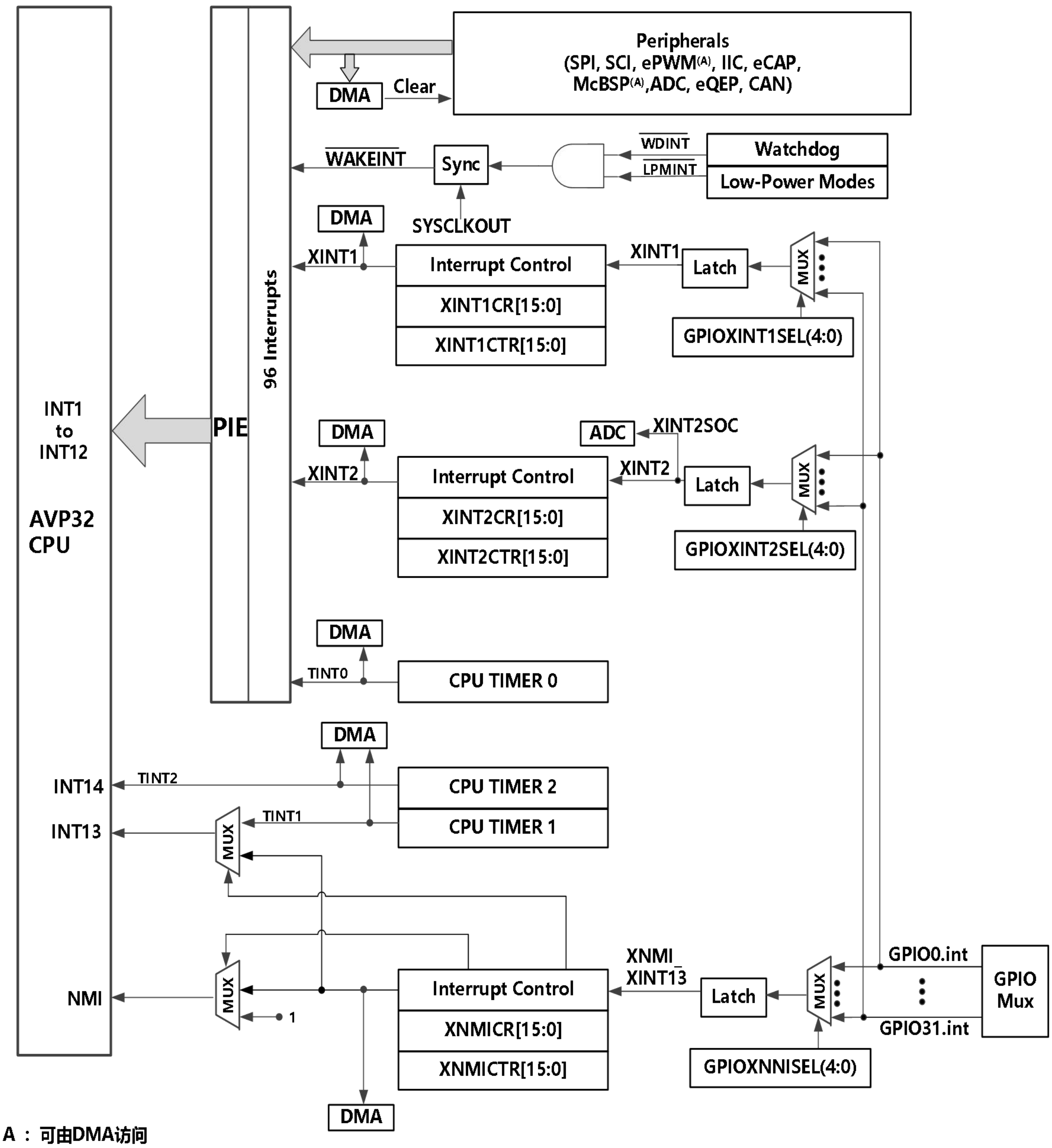


图 3-3 外部和 PIE 中断源

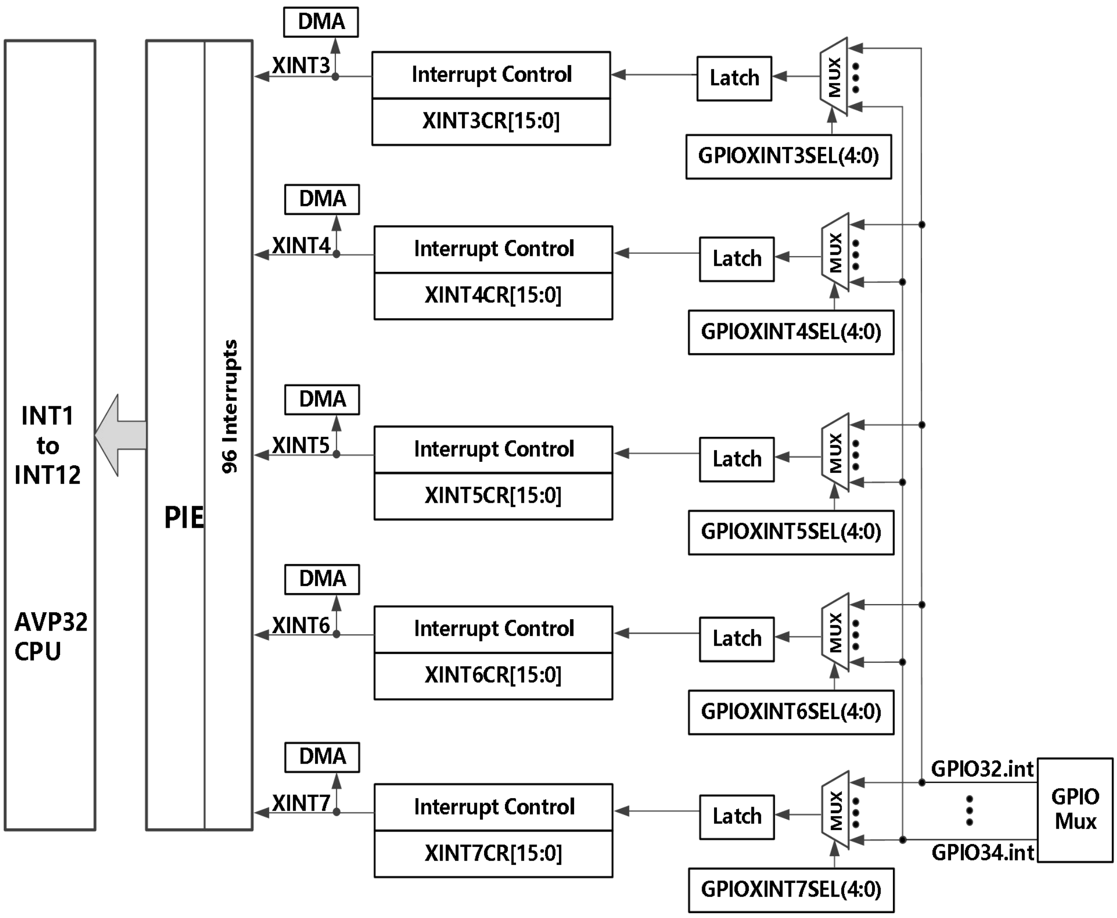


图 3-4 外部中断

8 个 PIE 块中断组合成一个 CPU 中断。总共 12 个 CPU 中断组，每组有 8 个中断，等于 96 个可用的中断。表 3-11 显示了 AVP32F08 器件所用的中断。

TRAP #Vectornumber(矢量号）指令将程序控制发送至与指定的矢量相对应的中断处理例程。TRAP#0 尝试将程序控制传送到复位矢量所指向的地址。然而，PIE 矢量表不含复位矢量。因此，当 PIE 被启用时，TRAP#0 不应被使用。这样做将导致未定义的运行状态。

当 PIE 启用时，TRAP#1 至 TRAP#12 将程序控制传送到与 PIE 组内第一个矢量相对应的中断处理例程。例如：TRAP #1 从 INT1.1 取矢量，TRAP #2 从 INT2.1 取矢量，以此类推。

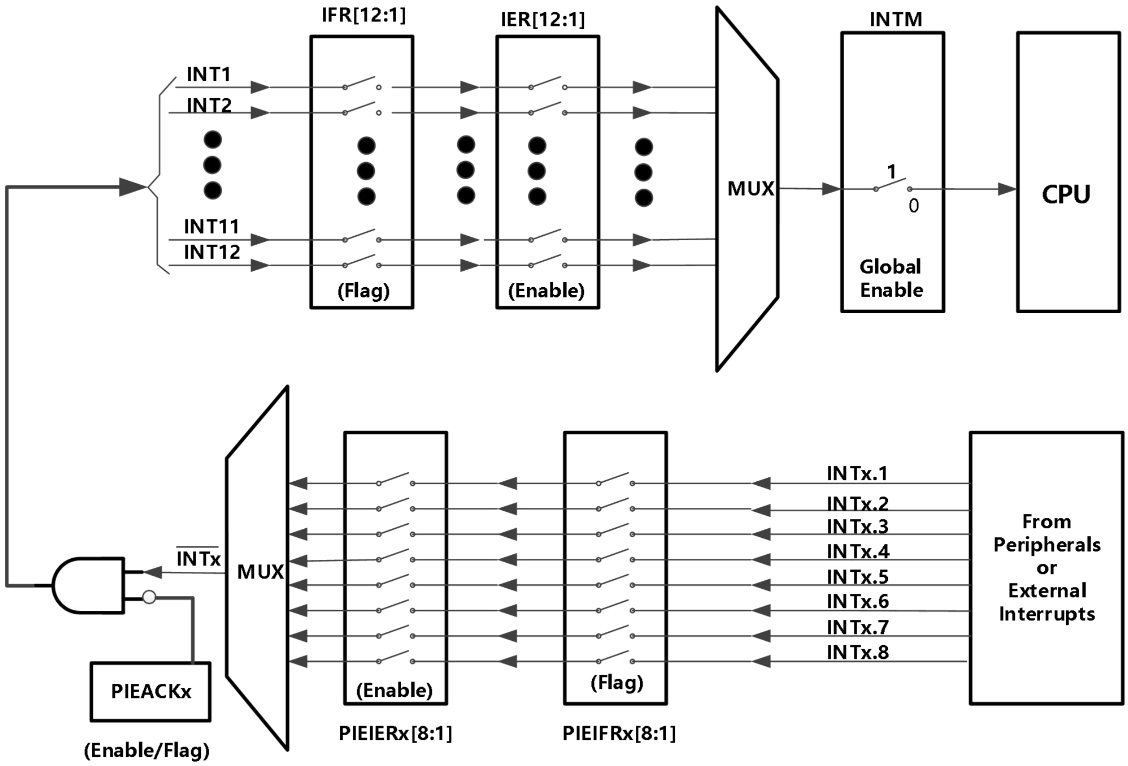


图 3-5 使用 PIE 块的中断复用

表 3-11 PIE 外设中断矢量表(1)

	INTx.8	INTx.7	INTx.6	INTx.5	INTx.4	INTx.3	INTx.2	INTx.1
INT1	WAKEINT (LPM/WD)	TINT0 (定时器0)	ADCINT (ADC)	XINT2	XINT1	保留区	SEQ2INT (ADC)	SEQ1INT (ADC)
INT2	保留区	保留区	EPWM6_TZINT (ePWM6)	EPWM5_TZINT (ePWM5)	EPWM4_TZINT (ePWM4)	EPWM3_TZINT (ePWM3)	EPWM2_TZINT (ePWM2)	EPWM1_TZINT (ePWM1)
INT3	保留区	保留区	EPWM6_INT (ePWM6)	EPWM5_INT (ePWM5)	EPWM4_INT (ePWM4)	EPWM3_INT (ePWM3)	EPWM2_INT (ePWM2)	EPWM1_INT (ePWM1)
INT4	保留区	保留区	ECAP6_INT (eCAP6)	ECAP5_INT (eCAP5)	ECAP4_INT (eCAP4)	ECAP3_INT (eCAP3)	ECAP2_INT (eCAP2)	ECAP1_INT (eCAP1)
INT5	保留区	保留区	保留区	保留区	保留区	保留区	EQEP2_INT (eQEP2)	EQEP1_INT (eQEP1)
INT6	保留区	保留区	MXINTA (McBSP-A)	MRINTA (McBSP-A)	MXINTB (McBSP-B)	MRINTB (McBSP-B)	SPITXINTA (SPI-A)	SPIRXINTA (SPI-A)
INT7	保留区	保留区	DINTCH6 (DMA)	DINTCH5 (DMA)	DINTCH4 (DMA)	DINTCH3 (DMA)	DINTCH2 (DMA)	DINTCH1 (DMA)
INT8	保留区	保留区	禁用	禁用	保留区	保留区	IICINT2A (IIC-A)	IICINT1A (IIC-A)
INT9	ECAN1_INTB (CAN-B)	ECAN0_INTB (CAN-B)	ECAN1_INTA (CAN-A)	ECAN0_INTA (CAN-A)	SCITXINTB (SCI-B)	SCIRXINTB (SCI-B)	SCITXINTA (SCI-A)	SCIRXINTA (SCI-A)
INT10	保留区	保留区	保留区	保留区	保留区	保留区	保留区	保留区
INT11	保留区	保留区	保留区	保留区	保留区	保留区	保留区	保留区
INT12	LUF (FPU)	LVF (FPU)	保留区	INT7	XINT6	XINT5	XINT4	XINT3

(1) 在 96 个可用中断中，有一些是不使用的。这些中断是为以后的器件所保留的。如果它们在 PIEIFRx 级被启用并且这个组中的中断均未被外设使用，则这些中断可被用作软件中断。否则，在意外地清除它们的标志同时修改 PIEIFR 的情况下，来自外设的中断也许会丢失。总的来说，在两种安全情况下，保留区的中断可被用作软件中断：

- 组内没有外设使中断有效。
- 没有外设中断被分配到组（例如，PIE 组 11）。

表 3-12 PIE 配置和控制寄存器

名称	地址	大小（X 16)	说明 (1)
PIECTRL	0x 0CE0	1	PIE,控制寄存器
PIEACK	0x 0CE1	1	PIE,应答寄存器
PIEIER1	0x 0CE2	1	PIE,INT1 组启用寄存器
PIEIFR1	0x 0CE3	1	PIE,INT1 组标志寄存器
PIEIER2	0x 0CE4	1	PIE,INT2 组启用寄存器
PIEIFR2	0x 0CE5	1	PIE,INT2 组标志寄存器
PIEIER3	0x 0CE6	1	PIE,INT3 组启用寄存器
PIEIFR3	0x 0CE7	1	PIE,INT3 组标志寄存器
PIEIER4	0x 0CE8	1	PIE,INT4 组启用寄存器
PIEIFR4	0x 0CE9	1	PIE,INT4 组标志寄存器
PIEIER5	0x 0CEA	1	PIE,INT5 组启用寄存器
PIEIFR5	0x 0CEB	1	PIE,INT5 组标志寄存器
PIEIER6	0x 0CEC	1	PIE,INT6 组启用寄存器
PIEIFR6	0x 0CED	1	PIE,INT6 组标志寄存器
PIEIER7	0x 0CEE	1	PIE,INT7 组启用寄存器
PIEIFR7	0x 0CEF	1	PIE,INT7 组标志寄存器
PIEIER8	0x 0CF0	1	PIE,INT8 组启用寄存器
PIEIFR8	0x 0CF1	1	PIE,INT8 组标志寄存器
PIEIER9	0x 0CF2	1	PIE,INT9 组启用寄存器
PIEIFR9	0x 0CF3	1	PIE,INT9 组标志寄存器
PIEIER10	0x 0CF4	1	PIE,INT10 组启用寄存器
PIEIFR10	0x 0CF5	1	PIE,INT10 组标志寄存器
PIEIER11	0x 0CF6	1	PIE,INT11 组启用寄存器
PIEIFR11	0x 0CF7	1	PIE,INT11 组标志寄存器
PIEIER12	0x 0CF8	1	PIE,INT12 组启用寄存器
PIEIFR12	0x 0CF9	1	PIE,INT12 组标志寄存器
保留区	0x 0CFA-0x 0CFF	6	保留区

(1) PIE 配置和控制寄存器未受 EALLOW 模式保护。PIE 矢量表受保护。

3.6.1 外部中断

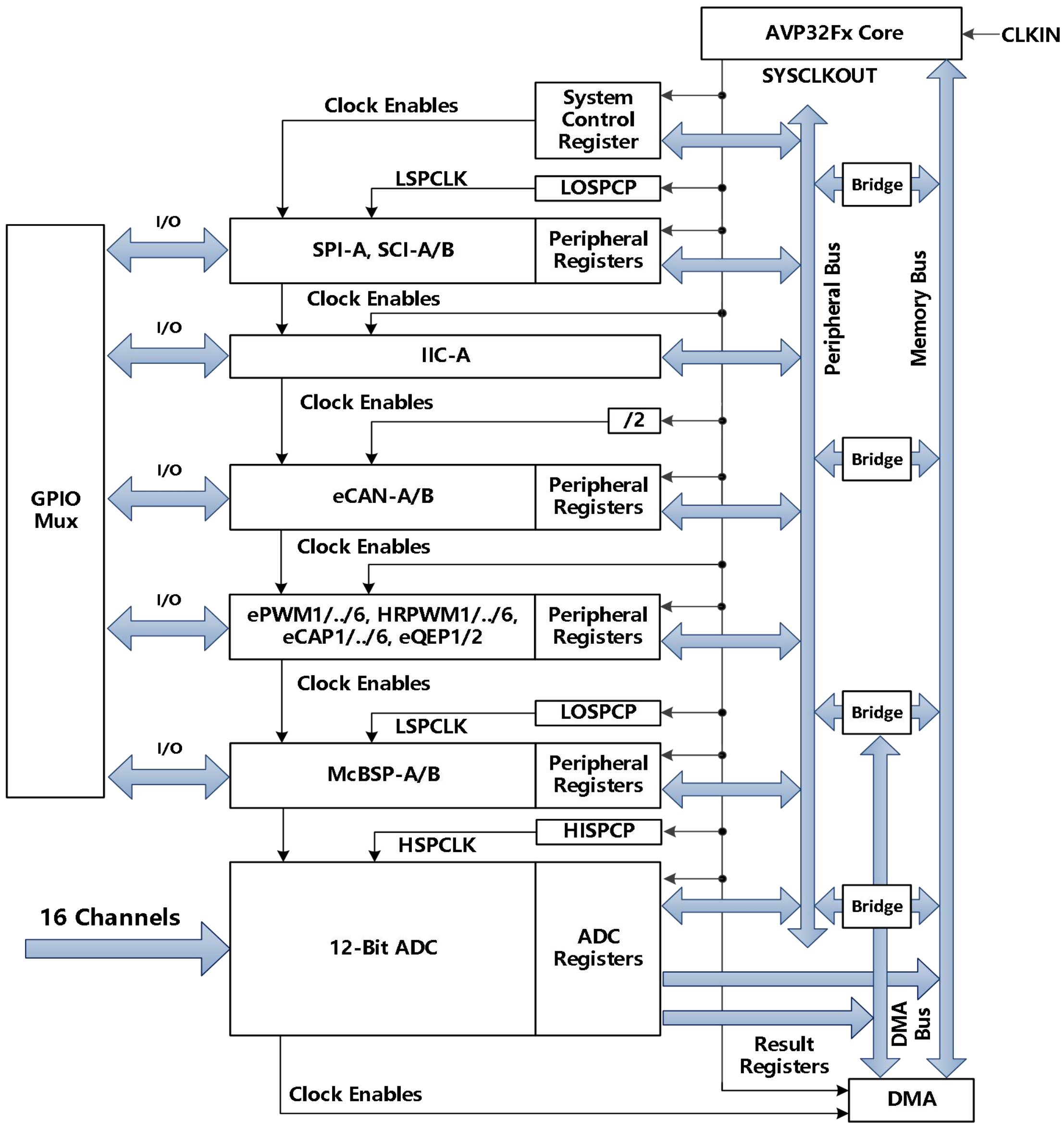
表 3-13 外部中断寄存器

名称	地址	大小（X 16)	说明
XINT1CR	0x00 7070	1	XINT1 配置寄存器
XINT2CR	0x00 7071	1	XINT2 配置寄存器
XINT3CR	0x00 7072	1	XINT3 配置寄存器
XINT4CR	0x00 7073	1	XINT4 配置寄存器
XINT5CR	0x00 7074	1	XINT5 配置寄存器
XINT6CR	0x00 7075	1	XINT6 配置寄存器
XINT7CR	0x00 7076	1	XINT7 配置寄存器
XNMICR	0x00 7077	1	XNMI 配置寄存器
XINT1CTR	0x00 7078	1	XINT1 计数器寄存器
XINT2CTR	0x00 7079	1	XINT2 计数器寄存器
保留区	0x707A-0x707E	5	保留区
XNMICTR	0x00 707A	1	XNMI 计数器寄存器

(1) 每个外部中断可由正边沿、负边沿或双沿启用、禁用或限定。

3.7 系统控制

本节对振荡器、PLL、时钟机制、看门狗功能以及低功耗模式进行了说明。图 3-6 显示了多个时钟域。



A. CLKIN为 CPU 提供时钟。它作为 SYSCLKOUT 从 CPU 传出（也就是说，CLKIN 与 SYSCLKOUT 频率相同）。如何为 CLKIN 供源的图解请参阅图 3-7。

图 3-6 时钟和复位域

注

从写入 PCLKCR0、PCLKCR1 和 PCLKCR2 寄存器（启用外设时钟）发生到操作有效，有两个 SYSCLKOUT 周期延迟。在尝试访问外设配置寄存器前，必须将该延迟考虑在内。

PLL、时钟、看门狗和低功率模式由表 3-14 中列出的寄存器控制。

表 3-14 PLL、时钟、看门狗和低功耗模式寄存器

名称	地址	大小（X 16）	说明
PLLSTS	0x00 7011	1	PLL 状态寄存器
保留区	0x00 7013-0x00 7019	7	保留区
HISPCP	0x00 701A	1	高速外设时钟预分频寄存器
LOSPCP	0x00 701B	1	低速外设时钟预分频寄存器
PCLKCR0	0x00 701C	1	外设时钟控制寄存器0
PCLKCR1	0x00 701D	1	外设时钟控制寄存器1
LPMCR0	0x00 701E	1	低功率模式控制寄存器0
保留区	0x00 701F	1	保留区
PCLKCR3	0x00 7020	1	外设时钟控制寄存器3
PLLCR	0x00 7021	1	PLL 控制寄存器
SCSR	0x00 7022	1	系统控制与状态寄存器
WDCNTR	0x00 7023	1	看门狗计数器寄存器
保留区	0x00 7024	1	保留区
WDKEY	0x00 7025	1	看门狗复位密钥寄存器
保留区	0x00 7026-0x00 7028	3	保留区
WDCR	0x00 7029	1	看门狗控制寄存器
保留区	0x00 702A–0x00 702D	4	保留区
MAPCNF	0x00 702E	1	EPWM/HRPWM 重新映射寄存器

3.7.1 OSC 和 PLL 模块

图 3-7 所示为 OSC 和 PLL 模块框图。

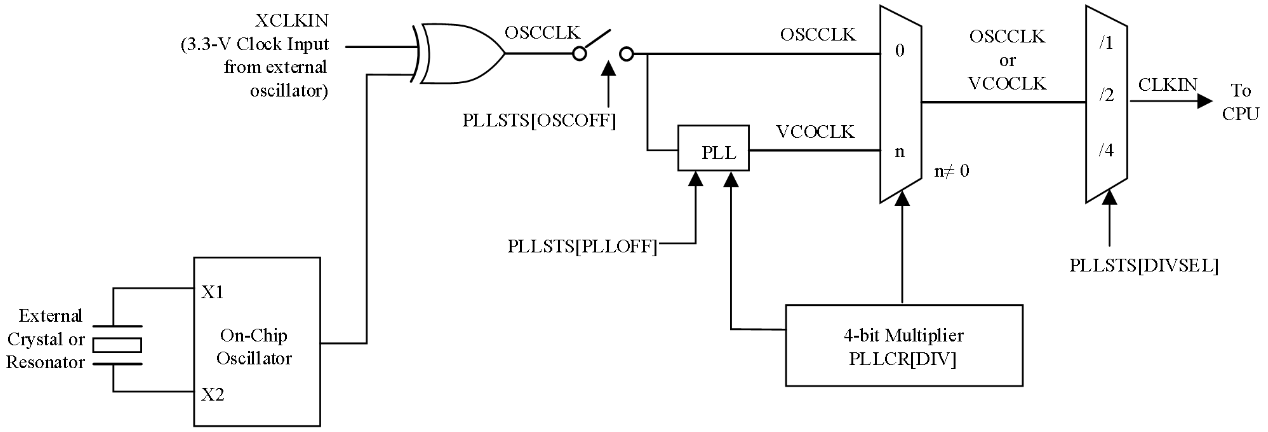


图 3-7 OSC 和 PLL 模块框图

片载振荡器电路启用一个使用 X1 和 X2 引脚连接至 AVP32F08 器件的晶振/谐振器。如果片载振荡器未被使用，那么一个外部振荡器可被用在下列配置中的任何一个：

1. 3.3V 外部振荡器可直接连接至 XCLKIN 引脚，此时 X2 引脚保持悬空，X1 引脚连接到地，该逻辑电平不可以超过 V_{DDIO}。

2. 一个 1.8V 外部振荡器可以直接连接到 X1 引脚，此时 X2 引脚保持悬空，而 XCLKIN 引脚连接到地。该逻辑电平不可以超过 V_{DD}。

图 3-8 至图 3-10 显示了三种模式的输入时钟配置。

图 3-8 3.3V 外部振荡器的使用

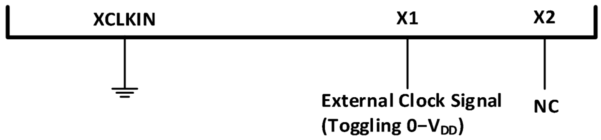


图 3-9 1.8V 外部振荡器的使用

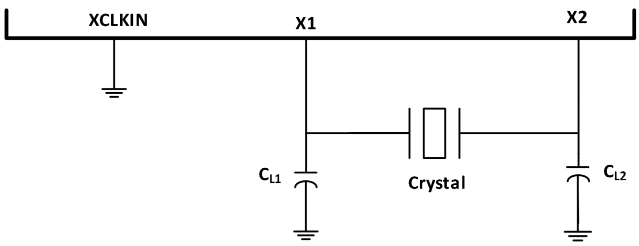


图 3-10 外部振荡器的使用

3.7.1.1 外部基准振荡器时钟选项

30MHz 外部石英晶振的典型技术规范如下:

- 基本模式、并联谐振
- CL（负载电容）=12pF
- $C_{L1}=C_{L2}$=24pF
- C_{shunt}=6pF
- ESR 范围 = 25 至 40Ω

3.7.1.2 基于 PLL 的时钟模块

此器件有一个片载、基于 PLL 的时钟模块。这个模块为器件提供所有需要的时钟信号，以及对进入低功耗模式的控制。PLL 有一个 4 位比率控制 PLLCR[DIV]来选择不同的 CPU 时钟速率。在写入 PLLCR 寄存器之前，应该禁用看门狗模块。在 PLL 模式稳定后，可重新启用看门狗模块（如有需要），重新启用的时间为 131072 个 OSCCLK 周期。输入时钟和 PLLCR[DIV]位应该在 PLL(VCOCLK)的输出频率不超过 300MHz 时选择。

表 3-15 PLL 设置

PLLCR[DIV]的值[2][3]	SYSCLKOUT (CLKIN)		
	PLLSTS[DIVSEL]=0 或 1[1]	PLLSTS[DIVSEL]=2[1]	PLLSTS[DIVSEL]=3[1][4]
0000（PLL 旁路）	OSCCLK/4（默认）[1]	OSCCLK/2	OSCCLK
0001	(OSCCLK * 1)/4	(OSCCLK * 1)/2	-
0010	(OSCCLK * 2)/4	(OSCCLK * 2)/2	-
0011	(OSCCLK * 3)/4	(OSCCLK * 3)/2	-
0100	(OSCCLK * 4)/4	(OSCCLK * 4)/2	-
0101	(OSCCLK * 5)/4	(OSCCLK * 5)/2	-
0110	(OSCCLK * 6)/4	OSCCLK * 6)/2	-
0111	(OSCCLK * 7)/4	(OSCCLK * 7)/2	-
1000	(OSCCLK * 8)/4	(OSCCLK * 8)/2	-
1001	(OSCCLK * 9)/4	(OSCCLK * 9)/2	-
1010	(OSCCLK * 10)/4	(OSCCLK * 10)/2	-
1011-1111	保留区	保留区	保留区

(1) 默认情况下，PLLSTS[DIVSEL]被配置为/4。（引导 ROM 将这个配置改为/2。）在写入 PLLCR 前，PLLSTS[DIVSEL]必须为 0，而只有当 PLLSTS[PLLOCKS]=1 时才应被改变。

(2) PLL 控制寄存器(PLLCR)和 PLL 状态寄存器(PLLSTS)只能通过 $\overline{XRS}$信号或者一个看门狗复位被复位至它们的缺省值。调试器或丢失时钟检测逻辑发出的复位信号无效。

(3) 此寄存器受 EALLOW 保护。

(4) PLL 输出上的分频器是必须的，以确保馈入内核中的时钟正确占空比。出于这个原因，当 PLL 处于激活状态时，DIVSEL 值不允许为 3。

表 3-16 CLKIN 分频选项

PLLSTS [DIVSEL]	CLKIN 分频
0	/4
1	/4
2	/2
3	/1 [1]

（1）仅当 PLL 旁路或关闭时，才能使用该模式。

基于 PLL 的时钟模块提供两种运行模式：

• 晶振操作-这个模式允许使用一个外部晶振/谐振器来提供器件的时基。

• 外部时钟源操作-这个模式允许内部振荡器被旁路。此器件时钟由一个 X1 或者 XCLKIN 引脚上的外部时钟源输入生成。

表 3-17 PLL 配置模式

PLL 模式	注释	PLLSTS[DIVSEL]	CLKIN 和 SYSCLKOUT
PLL 禁用	由 PLLSTS 寄存器中的 PLLOFF 位控制。在这个模式中，PLL 块被禁用。这对降低系统噪声和功耗非常有用。在进入此模式之前，必须先将 PLLCR 寄存器设置为 0x0000（PLL 旁路）。CPU 时钟源 (CLKIN) 来自 X1/X2、X1 或者 XCLKIN 上的输入时钟。	0，1 2 3	OSCCLK/4 OSCCLK/2 OSCCLK/1
PLL 旁路	PLL 旁路是上电或外部复位（$\overline{XRS}$）后的默认 PLL 配置。当 PLLCR 寄存器设置为 0x0000 时或在 PLLCR 寄存器已经被修改之后，PLL 锁定至新频率时选择此模式。在此模式中 PLL 本身被旁路，但未关闭。	0，1 2 3	OSCCLK/4 OSCCLK/2 OSCCLK/1
PLL 启用	通过将一个非零值"n"写入 PLLCR 寄存器实现。在写入 PLLCR 时，此器件将在 PLL 锁之前切换至 PLL 旁路模式。	0，1 2	OSCCLK * n/4 OSCCLK * n/2

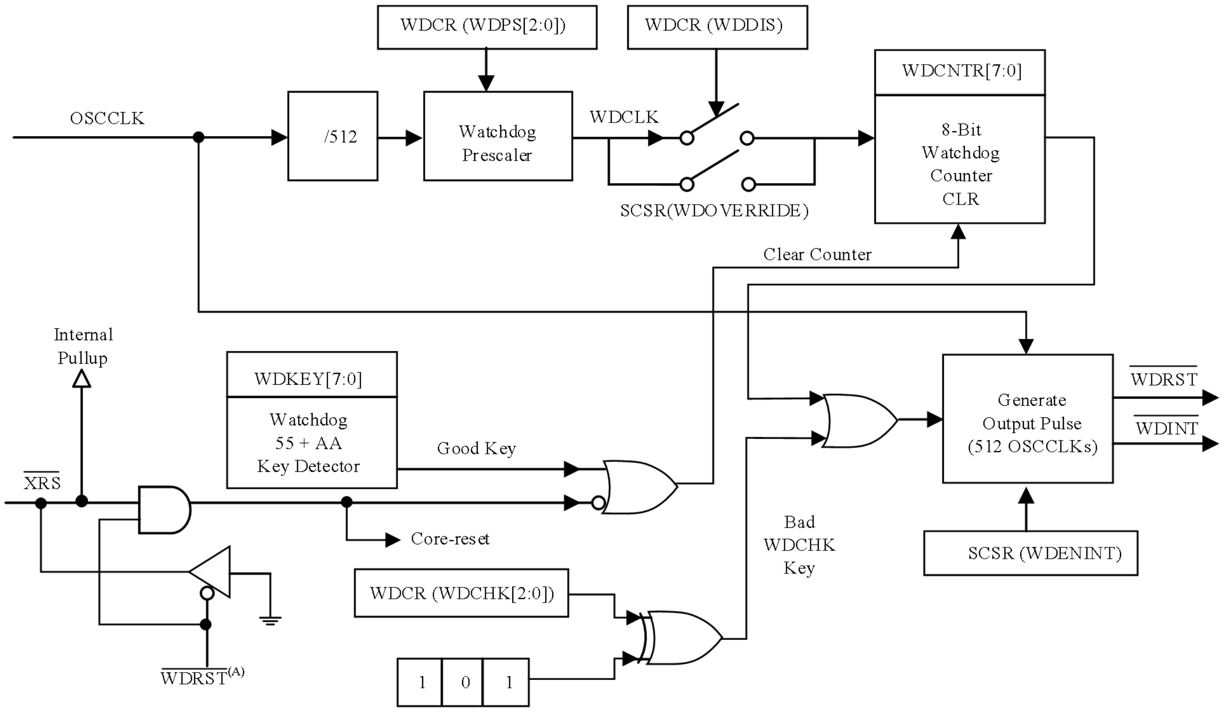
3.7.1.3 输入时钟丢失

在 PLL 启用或者 PLL 旁路模式中，如果输入时钟 OSCCLK 去除或者缺失，PLL 仍将输出一个跛行模式时钟。这个跛行模式时钟持续为 CPU 和典型频率为 200kHz-500kHz 的外设计时。额定情况下，跛行模式上电时并不运行，只在输入时钟首次出现时才运行。在 PLL 旁路模式中，如果输入时钟被移除或者缺失，来自 PLL 的跛行模式时钟被自动引至 CPU。

通常情况下，当输入时钟出现时，看门狗计数器递减来启动一个看门狗复位或者 WDINT 中断。然而，当外部输入时钟发生故障时，看门狗计数器停止递减（也就是说，看门狗计数器不会随着跛行模式时钟而改变）。除此之外，器件将被复位并且“丢失时钟状态”(MCLKSTS)位将被设定。这个条件可被应用固件用来检测输入时钟故障并启动系统所需的关断过程。

3.7.2 看门狗

AVP32F08 器件上的 8 位看门狗计数器计数达到最大值后，就会输出一个宽度为 512 个 OSC 时钟周期的脉冲。为了防止这一情况，用户必须禁用此计数器或者通过软件定期地往看门狗密钥寄存器写入一个 0x55+0xAA 序列复位此看门狗计数器。图 3-11 所示为看门狗模块功能框图。



A. $\overline{\text{WDRST}}$信号在 512 个 OSCCLK 周期内被驱动为低电平。

图 3-11 看门狗模块

$\overline{\text{WDINT}}$信号使得看门狗可被用作一个从 IDLE/STANDY 模式唤醒的激励源。

在 STANDBY 模式中，器件上的所有外设关闭。仍然可用的唯一外设是看门狗，这个模块将关闭 OSCCLK。$\overline{\text{WDINT}}$信号被馈送到 LPM 块以便它可以将器件从 STANDBY 唤醒（如已启用）。更多细节，请见第 3.8 节低功耗模式块。

在 IDLE 模式中，$\overline{\text{WDINT}}$信号可通过 PIE 来生成一个到 CPU 的中断来将 CPU 从 IDLE 模式中唤醒。

在 HALT 模式中，该功能无法使用，因为振荡器（及 PLL）关闭，因此看门狗也处于关闭状态。

3.8 低功耗模式

表 3-18 总结了各种低功耗模式。

表 3-18 低功耗模式

模式	LPMCR0(1:0)	OSCCLK	CLKIN	SYSCLKOUT	退出(1)
IDLE	00	打开	打开	打开(2)	$\overline{XRS}$，看门狗中断，任何被启动的中断、XNMI
STANDBY	01	打开 (看门狗仍然运行)	关闭	关闭	$\overline{XRS}$，看门狗中断，GPIO 端口 A 信号，调试器、XNMI
HALT	1X	关闭 (片载振荡器和 PLL 关闭，看门狗不工作)	关闭	关闭	$\overline{XRS}$，GPIO 端口 A 信号，调试器、XNMI

(1) “退出”列说明了哪些信号或在哪些情况下会退出低功耗模式，在这些信号中的任何一个低电平信号，都将退出低功耗状态。此信号必须保持低电平足够长时间以便器件识别中断。否则，将不会从 IDLE 模式中退出，而器件将返回指示的低功耗模式。

(2) 在 AVP32F08 上，即使内核时钟（CLKIN）被关闭，JTAG 端口仍可以工作。

三种不同的低功耗模式运行状态如下：

IDLE 模式 通过任一被启用的中断或者一个被处理器识别的 XNMI 来退出此模式。LPMCR0(LPM)位被设定为 0 时，LPM 块在该模式期间不执行任何任务。

STANDBY 模式 任何一个 GPIO 端口 A 信号 (GPIO[31:0]) 能够将器件从 STANDBY 模式中唤醒。用户必须通过 GPIOLPMSEL 寄存器选择哪一个信号将器件唤醒。在唤醒器件前，所选的信号会被一定数量的 OSCCLK 限定，OSCCLK 的数量由 LPMCR0 寄存器相应的比特位设置。

HALT 模式 $\overline{XRS}$和任何一个 GPIO 端口 A 信号 (GPIO[31:0])都可将器件从 HALT 模式中唤醒。用户在 GPIOLPMSEL 寄存器中选择信号。

注

低功耗模式并不会影响输出引脚的状态（包括 PWM 引脚在内）。当执行 IDLE 指令时，它们将保持在代码指定的状态中。

4 外设

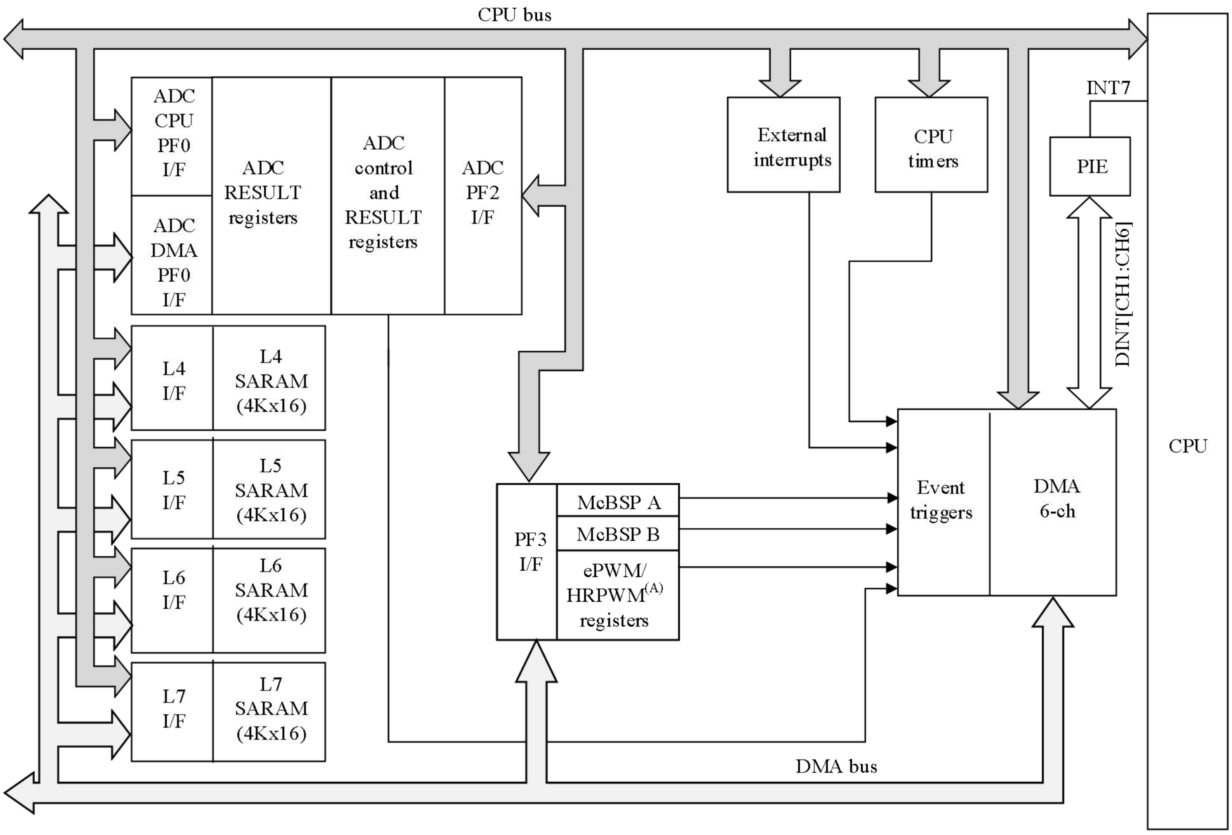
AVP32F08 器件的集成外设在以下部分进行了说明：

- 6 通道直接内存存取(DMA)
- 三个 32 位 CPU 定时器
- 高达 6 个增强型 PWM 模块(ePWM1，ePWM2，ePWM3，ePWM4，ePWM5，ePWM6)
- 高达 6 个增强型捕获模块(eCAP1，eCAP2，eCAP3，eCAP4，eCAP5，eCAP6)
- 高达 2 个增强型 QEP 模块(eQEP1，eQEP2)
- 增强型模数转换器(ADC)模块
- 多达 2 个增强型控制器局域网(eCAN)模块(eCAN-A，eCAN-B)
- 多达 2 个串行通信接口模块(SCI-A，SCI-B)
- 1 个串行外设接口(SPI)模块(SPI-A)
- 1 个 IIC 模块
- 高达两个多通道缓冲串口(McBSP-A，McBSP-B)模块
- 数字 I/O 和共用引脚功能

4.1 DMA 概述

特性:

- 6 个具有独立 PIE 中断的通道
- 触发源：
  - ePWM SOCA/SOCB
  - ADC 序列发生器 1 和序列发生器 2
  - McBSP-A 和 McBSP-B 发送和接收逻辑
  - XINT1-7 和 XINT13
  - CPU 定时器
  - 软件
- 数据源/目的地：
  - L4–L7 16K×16 Bit SARAM
  - ADC 内存总线映射结果寄存器
  - McBSP-A 和 McBSP-B 发送和接收缓冲区
  - ePWM 寄存器
- 字大小：16 位或 32 位（McBSP 限制到 16 位）
- 吞吐量：4 个周期/字（McBSP 读取时为 5 个周期/字）



A.　ePWM和HRPWM寄存器必须重新映射到外设帧3（通过MAPCNF寄存器的位 0）之后才可以由DMA访问。

图 4-1 DMA 功能方框图

4.2 32 位 CPU 定时器 0，定时器 1，定时器 2

在器件上有 3 个 32 位 CPU 定时器(CPU 定时器 0，CPU 定时器 1，CPU 定时器 2)。

定时器 2 为 DSP 系统预留。可以在用户应用程序中使用 CPU 定时器 0 和定时器 1。这些定时器与 ePWM 模块中的定时器不同。

注

请注意：如果应用不使用 DSP/BIOS，则 CPU 定时器 2 可用在应用中。

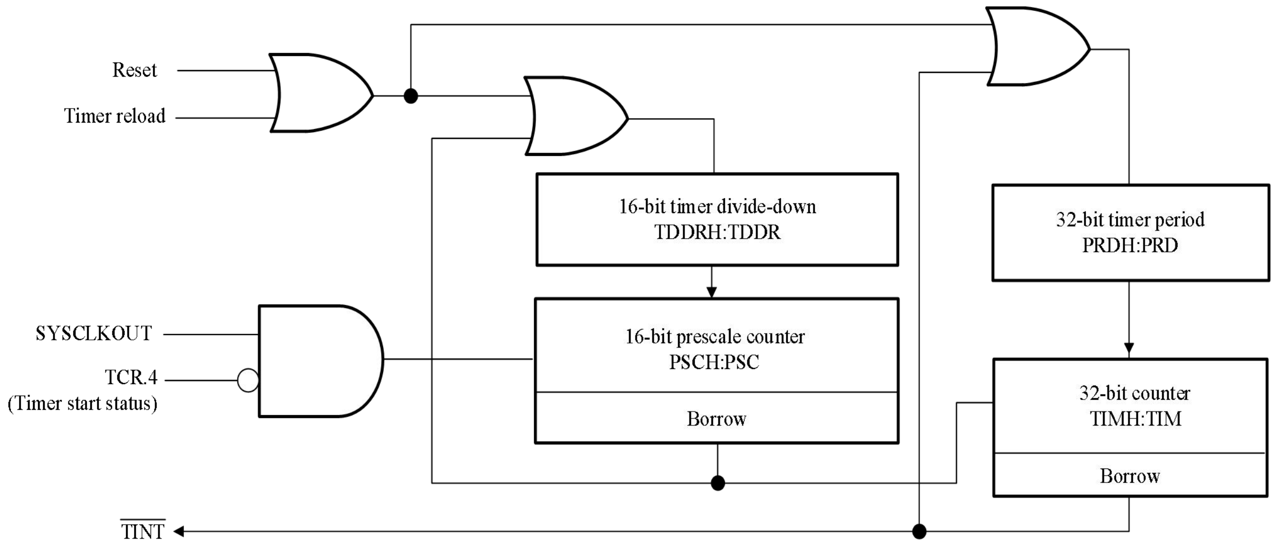
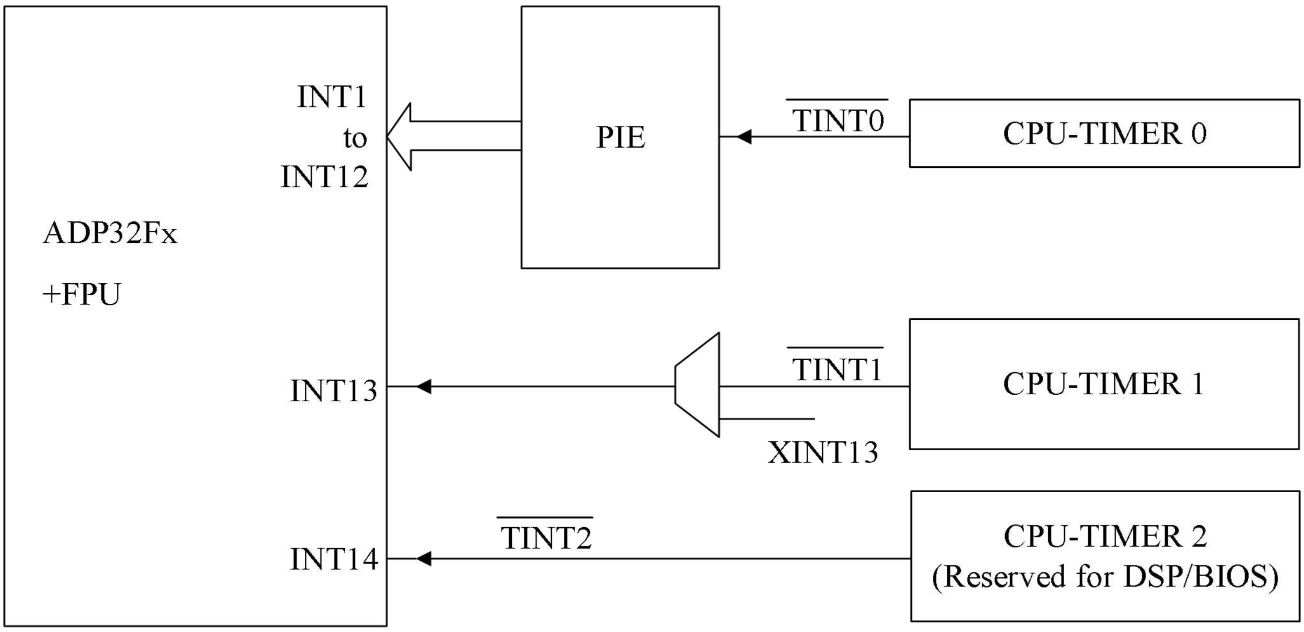


图 4-2 CPU 定时器

定时器中断信号 ($\overline{\text{TINT0}}$，$\overline{\text{TINT1}}$，$\overline{\text{TINT2}}$)的连接如图 4-3 所示。



A 定时器寄存器连接到 AVP32F08 处理器的存储器总线。

B 定时器的时序与处理器时钟的 SYSCLKOUT 同步。

图 4-3 CPU 定时器中断信号和输出信号

定时器的通常操作如下：32 位计数器寄存器“TIMH:TIM”会加载周期寄存器“PRDH:PRD”中的值。计数器寄存器按 AVP32F08 的 SYSCLKOUT 速率递减。当计数器到达 0 时，一个定时器中断输出信号生成一个中断脉冲。表 4-1 中列出的寄存器用于配置定时器。

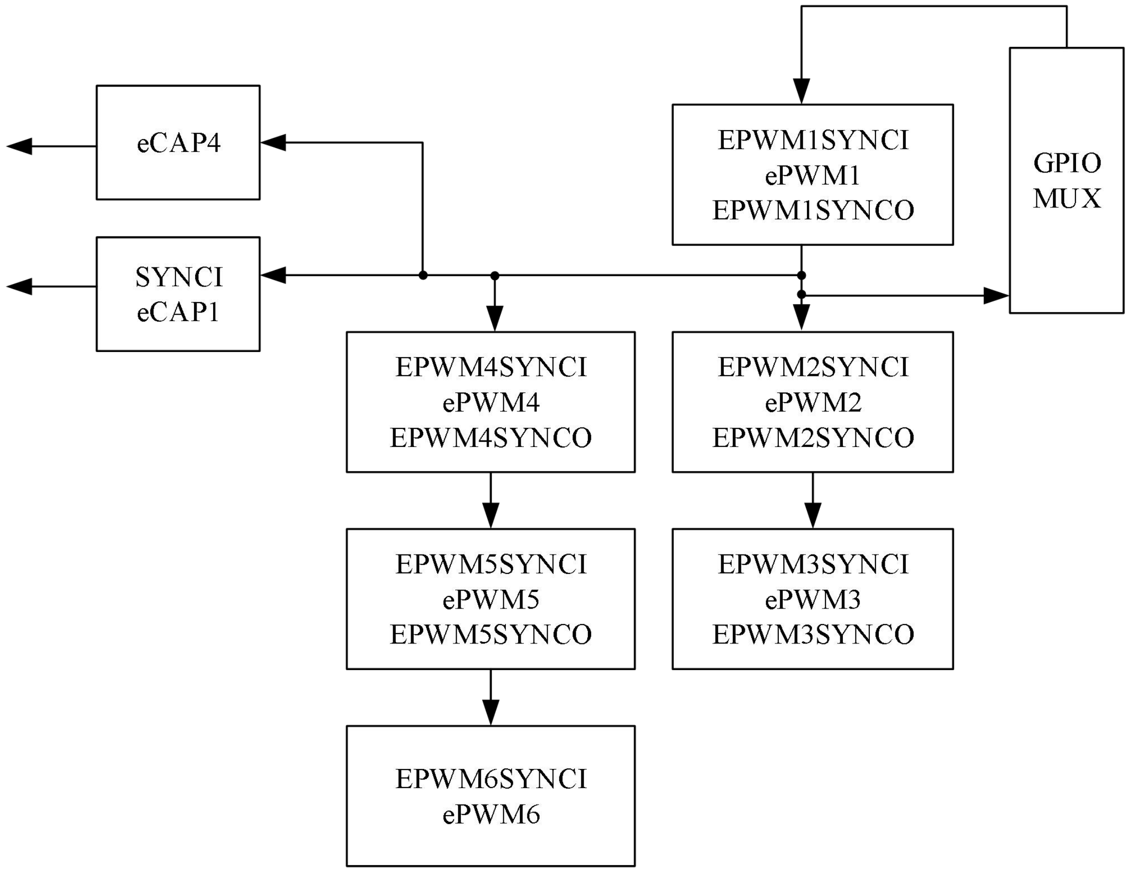
表 4-1 CPU 定时器 0, 1, 2 配置和控制寄存器

寄存器名称	地址	大小(x 16)	说明
TIMER0TIM	0x0C00	1	CPU 定时器 0，计数器寄存器
TIMER0TIMH	0x0C01	1	CPU 定时器 0，计数器寄存器高电平
TIMER0PRD	0x0C02	1	CPU 定时器 0，周期寄存器
TIMER0PRDH	0x0C03	1	CPU 定时器 0，周期寄存器高电平
TIMER0TCR	0x0C04	1	CPU 定时器 0，控制寄存器
保留区	0x0C05	1	保留区
TIMER0TPR	0x0C06	1	CPU 定时器 0，预分频寄存器
TIMER0TPRH	0x0C07	1	CPU 定时器 0，预分频寄存器高电平
TIMER1TIM	0x0C08	1	CPU 定时器 1，计数器寄存器
TIMER1TIMH	0x0C09	1	CPU 定时器 1，计数器寄存器高电平
TIMER1PRD	0x0C0A	1	CPU 定时器 1，周期寄存器
TIMER1PRDH	0x0C0B	1	CPU 定时器 1，周期寄存器高电平
TIMER1TCR	0x0C0C	1	CPU 定时器 1，控制寄存器
保留区	0x0C0D	1	保留区
TIMER1TPR	0x0C0E	1	CPU 定时器 1，预分频寄存器
TIMER1TPRH	0x0C0F	1	CPU 定时器 1，预分频寄存器高电平
TIMER2TIM	0x0C10	1	CPU 定时器 2，计数器寄存器
TIMER2TIMH	0x0C11	1	CPU 定时器 2，计数器寄存器高电平
TIMER2PRD	0x0C12	1	CPU 定时器 2，周期寄存器
TIMER2PRDH	0x0C13	1	CPU 定时器 2，周期寄存器高电平
TIMER2TCR	0x0C14	1	CPU 定时器 2，控制寄存器
保留区	0x0C15	1	保留区
TIMER2TPR	0x0C16	1	CPU 定时器 2，预分频寄存器
TIMER2TPRH	0x0C17	1	CPU 定时器 2，预分频寄存器高电平
保留区	x0 C18-0x0 0C3F	40	保留区

4.3 增强型 PWM 模块

AVP32F08 器件包含高达 6 个增强型 PWM 模块 (ePWM1, ePWM2, ePWM3, ePWM4, ePWM5, ePWM6)。图 4-4 显示了时基计数器同步方案。图 4-5 显示了与 ePWM 互连的信号。

表 4-2 显示了每个模块的完整 ePWM 寄存器设置，表 4-3 显示了重新映射的寄存器配置。



(1) 默认情况下，ePWM 和 HRPWM 寄存器被映射到外设帧 1（PF1）。表 4-2 显示该配置。重新映射寄存器至外设帧 3 (PF3) 来启用 DMA 访问，MAPCNF 寄存器（地址 0x702E）的位 0 (MAPEPWM) 必须被设置为 1。表 4-3 显示了重新映射的配置。

图 4-4 时基计数器同步方案

表 4-2 ePWM 控制和状态寄存器(PF1 中的默认值)

名称	ePWM1	ePWM2	ePWM3	ePWM4	ePWM5	ePWM6	大小(x16)/ #SHADOW	说明
TBCTL	0x6800	0x6840	0x6880	0x68C0	0x6900	0x6940	1/0	时基控制寄存器
TBSTS	0x6801	0x6841	0x6881	0x68C1	0x6901	0x6941	1/0	时基状态寄存器
TBPHSHR	0x6802	0x6842	0x6882	0x68C2	0x6902	0x6942	1/0	时基相位HRPWM 寄存器
TBPHS	0x6803	0x6843	0x6883	0x68C3	0x6903	0x6943	1/0	时基相位寄存器
TBCTR	0x6804	0x6844	0x6884	0x68C4	0x6904	0x6944	1/0	时基计数器寄存器
TBPRD	0x6805	0x6845	0x6885	0x68C5	0x6905	0x6945	1/1	时基周期寄存器集
CMPCTL	0x6807	0x6847	0x6887	0x68C7	0x6907	0x6947	1/0	计数器比较控制寄存器
CMPAHR	0x6808	0x6848	0x6888	0x68C8	0x6908	0x6948	1/1	时基比较 A HRPWM寄存器
CMPA	0x6809	0x6849	0x6889	0x68C9	0x6909	0x6949	1/1	计数器比较A 寄存器集
CMPB	0x680A	0x684A	0x688A	0x68CA	0x690A	0x694A	1/1	计数器比较B 寄存器集
AQCTLA	0x680B	0x684B	0x688B	0x68CB	0x690B	0x694B	1/0	用于输出A 的操作限定器控制寄存器
AQCTLB	0x680C	0x684C	0x688C	0x68CC	0x690C	0x694C	1/0	用于输出B 的操作限定器控制寄存器
AQSFRC	0x680D	0x684D	0x688D	0x68CD	0x690D	0x694D	1/0	操作限定器软件强制寄存器
AQCSFRC	0x680E	0x684E	0x688E	0x68CE	0x690E	0x694E	1/1	操作限定器连续S/W 强制寄存器集
DBCTL	0x680F	0x684F	0x688F	0x68CF	0x690F	0x694F	1/1	死区生成器控制寄存器
DBRED	0x6810	0x6850	0x6890	0x68D0	0x6910	0x6950	1/0	死区生成器上升沿延迟计数寄存器
DBFED	0x6811	0x6851	0x6891	0x68D1	0x6911	0x6951	1/0	死区生成器下降沿延迟计数寄存器
TZSEL	0x6812	0x6852	0x6892	0x68D2	0x6912	0x6952	1/0	触发区选择寄存器(1)
TZCTL	0x6814	0x6854	0x6894	0x68D4	0x6914	0x6954	1/0	触发区控制寄存器(1)
TZEINT	0x6815	0x6855	0x6895	0x68D5	0x6915	0x6955	1/0	触发区启用中断寄存器(1)
TZFLG	0x6816	0x6856	0x6896	0x68D6	0x6916	0x6956	1/0	触发区标志寄存器
TZCLR	0x6817	0x6857	0x6897	0x68D7	0x6917	0x6957	1/0	触发区清除寄存器(1)
TZFRC	0x6818	0x6858	0x6898	0x68D8	0x6918	0x6958	1/0	触发区强制寄存器(1)
ETSEL	0x6819	0x6859	0x6899	0x68D9	0x6919	0x6959	1/0	事件触发器选择寄存器
ETPS	0x681A	0x685A	0x689A	0x68DA	0x691A	0x695A	1/0	事件触发器预分频寄存器
ETFLG	0x681B	0x685B	0x689B	0x68DB	0x691B	0x695B	1/0	事件触发器标志寄存器
ETCLR	0x681C	0x685C	0x689C	0x68DC	0x691C	0x695C	1/0	事件触发器清除寄存器
ETFRC	0x681D	0x685D	0x689D	0x68DD	0x691D	0x695D	1/0	事件触发器强制寄存器
PCCTL	0x681E	0x685E	0x689E	0x68DE	0x691E	0x695E	1/0	PWM 斩波器控制寄存器
HRCNFG	0x6820	0x6860	0x68A0	0x68E0	0x6920	0x6960	1/0	HRPWM 配置寄存器(1)
TBCTL	0x6800	0x6840	0x6880	0x68C0	0x6900	0x6940	1/0	时基控制寄存器
TBSTS	0x6801	0x6841	0x6881	0x68C1	0x6901	0x6941	1/0	时基状态寄存器

(1) 寄存器受 EALLOW 保护。

表 4-3 ePWM 控制和状态寄存器(PF3 中的默认值重新映射的配置-可由 DMA 访问)

名称	ePWM1	ePWM2	ePWM3	ePWM4	ePWM5	ePWM6	大小(x16)/ #SHADOW	说明
TBCTL	0x5800	0x5840	0x5880	0x58C0	0x5900	0x5940	1/0	时基控制寄存器
TBSTS	0x5801	0x5841	0x5881	0x58C1	0x5901	0x5941	1/0	时基状态寄存器
TBPHSHR	0x5802	0x5842	0x5882	0x58C2	0x5902	0x5942	1/0	时基相位HRPWM 寄存器
TBPHS	0x5803	0x5843	0x5883	0x58C3	0x5903	0x5943	1/0	时基相位寄存器
TBCTR	0x5804	0x5844	0x5884	0x58C4	0x5904	0x5944	1/0	时基计数器寄存器
TBPRD	0x5805	0x5845	0x5885	0x58C5	0x5905	0x5945	1/1	时基周期寄存器集
CMPCTL	0x5807	0x5847	0x5887	0x58C7	0x5907	0x5947	1/0	计数器比较控制寄存器
CMPAHR	0x5808	0x5848	0x5888	0x58C8	0x5908	0x5948	1/1	时基比较 A HRPWM寄存器
CMPA	0x5809	0x5849	0x5889	0x58C9	0x5909	0x5949	1/1	计数器比较A 寄存器设置
CMPB	0x580A	0x584A	0x588A	0x58CA	0x590A	0x594A	1/1	计数器比较B 寄存器设置
AQCTLA	0x580B	0x584B	0x588B	0x58CB	0x590B	0x594B	1/0	用于输出A 的操作限定器控制寄存器
AQCTLB	0x580C	0x584C	0x588C	0x58CC	0x590C	0x594C	1/0	用于输出B 的操作限定器控制寄存器
AQSFRC	0x580D	0x584D	0x588D	0x58CD	0x590D	0x594D	1/0	操作限定器软件强制寄存器
AQCSFRC	0x580E	0x584E	0x588E	0x58CE	0x590E	0x594E	1/1	操作限定器连续S/W 强制寄存器设置
DBCTL	0x580F	0x584F	0x588F	0x58CF	0x590F	0x594F	1/1	死区生成器控制寄存器
DBRED	0x5810	0x5850	0x5890	0x58D0	0x5910	0x5950	1/0	死区生成器上升沿延迟计数寄存器
DBFED	0x5811	0x5851	0x5891	0x58D1	0x5911	0x5951	1/0	死区生成器下降沿延迟计数寄存器
TZSEL	0x5812	0x5852	0x5892	0x58D2	0x5912	0x5952	1/0	触发区选择寄存器(1)
TZCTL	0x5814	0x5854	0x5894	0x58D4	0x5914	0x5954	1/0	触发区控制寄存器(1)
TZEINT	0x5815	0x5855	0x5895	0x58D5	0x5915	0x5955	1/0	触发区启用中断寄存器(1)
TZFLG	0x5816	0x5856	0x5896	0x58D6	0x5916	0x5956	1/0	触发区标志寄存器
TZCLR	0x5817	0x5857	0x5897	0x58D7	0x5917	0x5957	1/0	触发区清除寄存器(1)
TZFRC	0x5818	0x5858	0x5898	0x58D8	0x5918	0x5958	1/0	触发区强制寄存器(1)
ETSEL	0x5819	0x5859	0x5899	0x58D9	0x5919	0x5959	1/0	事件触发器选择寄存器
ETPS	0x581A	0x585A	0x589A	0x58DA	0x591A	0x595A	1/0	事件触发器预分频寄存器
ETFLG	0x581B	0x585B	0x589B	0x58DB	0x591B	0x595B	1/0	事件触发器标志寄存器
ETCLR	0x581C	0x585C	0x589C	0x58DC	0x591C	0x595C	1/0	事件触发器清除寄存器
ETFRC	0x581D	0x585D	0x589D	0x58DD	0x591D	0x595D	1/0	事件触发器强制寄存器
PCCTL	0x581E	0x585E	0x589E	0x58DE	0x591E	0x595E	1/0	PWM 斩波器控制寄存器
HRCNFG	0x5820	0x5860	0x58A0	058E0	0x5920	0x5960	1/0	HRPWM 配置寄存器(1)

(1) 寄存器受 EALLOW 保护。

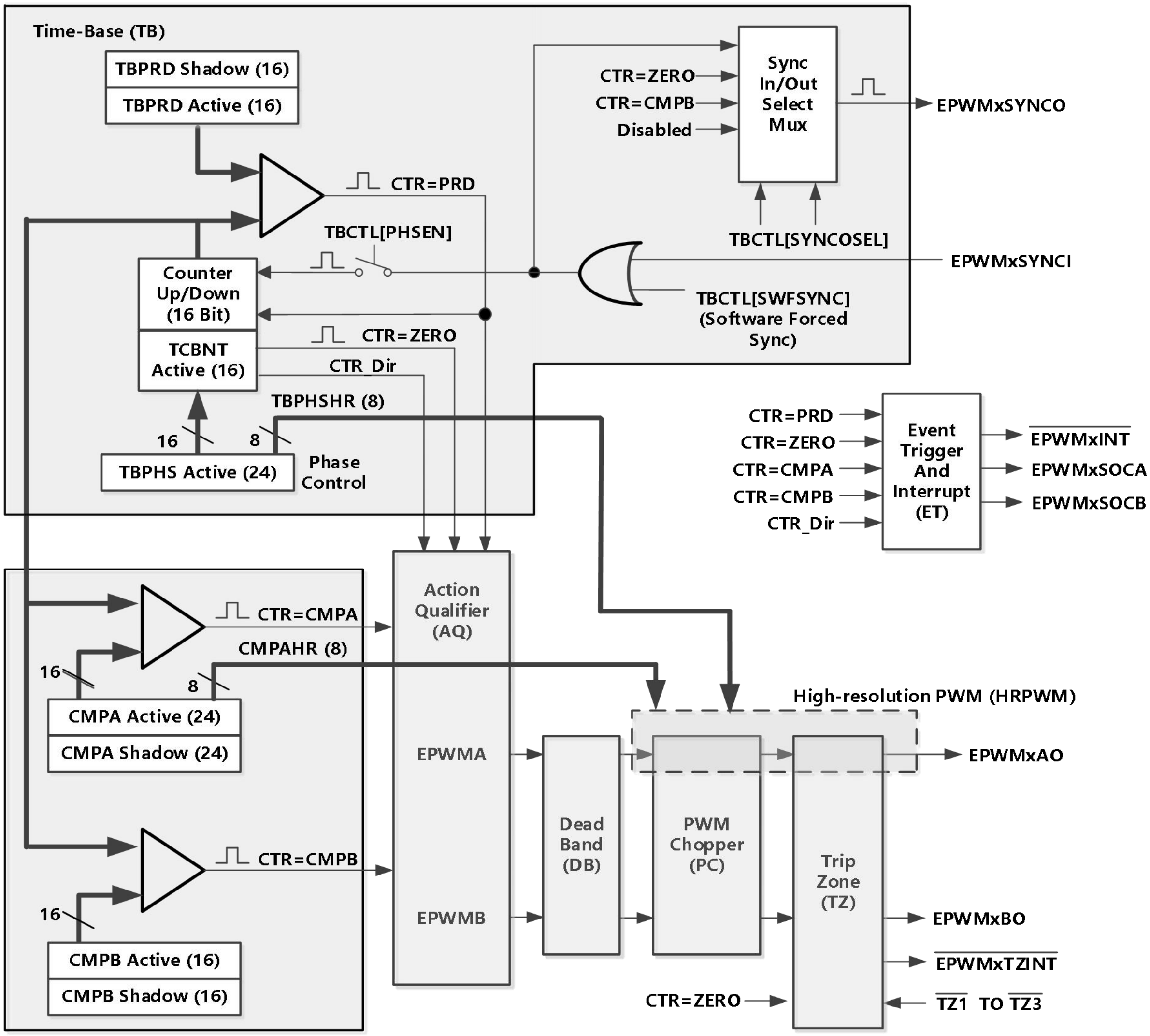


图 4-5 ePWM 子模块显示关键内部信号互连

4.4 高分辨率 PWM (HRPWM)

HRPWM 模块所提供的 PWM 分辨率（时间粒度）显著强于使用传统数字 PWM 方法所能输出的分辨率。HRPWM 模块的关键点为:

- 显著扩展了传统导出数字 PWM 的时间分辨率功能。
- 通常在有效 PWM 分辨率下降到低于大约 9 位时使用。当用一个 150MHz 的 CPU/系统时钟时，PWM 频率大于大约 300kHz 时会发生这种情况。
- 这个功能可被用在占空比和相移控制方法中。
- 通过对 ePWM 模块的比较器 A 和相位寄存器的扩展来控制更加精细的时间粒度控制或者边沿定位。
- HRPWM 功能，只在 ePWM 模块的 A 信号路径上提供（也就是说，在 EPWMxA 输出上提供），EPWMxB 输出具有传统 PWM 功能。

4.5 增强型捕获模块 (eCAP)

AVP32F08 器件包括 6 个增强型捕获 (eCAP) 模块 (eCAP1，eCAP2，eCAP3，eCAP4，eCAP5 和 eCAP6)。图 4-6 显示了一个模块的功能框图。

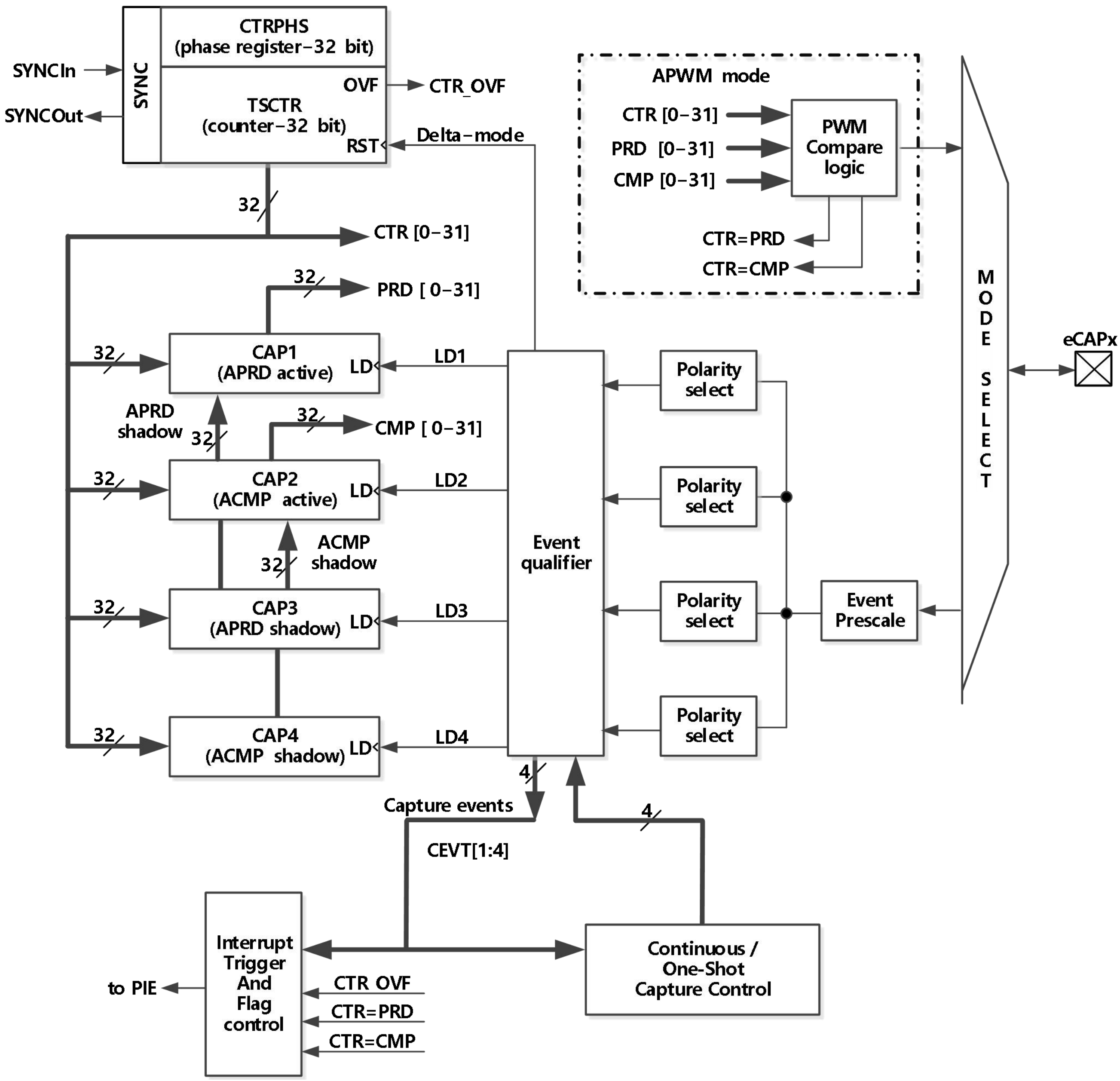


图 4-6 eCAP 功能框图

eCAP 模块以 SYSCLKOUT 速率计时。

PCLKCR1 寄存器中的时钟使能位(ECAP1ENCLK、ECAP2ENCLK、ECAP3ENCLK、ECAP4ENCLK、ECAP5ENCLK 和 ECAP6ENCLK)被单独用来关闭 eCAP 模块（针对低功耗运行）。复位后，ECAP1ENCLK、ECAP2ENCLK、ECAP3ENCLK、ECAP4ENCLK、 ECAP5ENCLK 和 ECAP6ENCLK 被设为低电平，表明外设时钟关闭。

表 4-4 eCAP 控制和状态寄存器

名称	eCAP1	eCAP2	eCAP3	eCAP4	eCAP5	eCAP6	大小 (x 16)	说明
TSCTR	0x6A00	0x6A20	0x6A40	0x6A60	0x6A80	0x6AA0	2	时间戳计数器
CTRPHS	0x6A02	0x6A22	0x6A42	0x6A62	0x6A82	0x6AA2	2	计数器相位偏移值寄存器
CAP1	0x6A04	0x6A24	0x6A44	0x6A64	0x6A84	0x6AA4	2	捕获器1 寄存器
CAP2	0x6A06	0x6A26	0x6A46	0x6A66	0x6A86	0x6AA6	2	捕获器2 寄存器
CAP3	0x6A08	0x6A28	0x6A48	0x6A68	0x6A88	0x6AA8	2	捕获器3 寄存器
CAP4	0x6A0A	0x6A2A	0x6A4A	0x6A6A	0x8A6A	0x6AAA	2	捕获器4 寄存器
保留区	0x6A0C-0x6A12	0x6A2C-0x6A32	0x6A4C-0x6A52	0x6A6C-0x6A72	0x6A8C-0x6A92	0x6AAC-0x6AB2	8	保留区
ECCTL1	0x6A14	0x6A34	0x6A54	0x6A74	0x6A94	0x6AB4	1	捕获控制寄存器1
ECCTL2	0x6A15	0x6A35	0x6A55	0x6A75	0x6A95	0x6AB5	1	捕获控制寄存器2
ECEINT	0x6A16	0x6A36	0x6A56	0x6A76	0x6A96	0x6AB6	1	捕获中断使能寄存器
ECFLG	0x6A17	0x6A37	0x6A57	0x6A77	0x6A97	0x6AB7	1	捕获中断标志寄存器
ECCLR	0x6A18	0x6A38	0x6A58	0x6A78	0x6A98	0x6AB8	1	捕获中断清除寄存器
ECFRC	0x6A19	0x6A39	0x6A59	0x6A79	0x6A99	0x6AB9	1	捕获中断强制寄存器
保留区	0x6A1A-0x6A1F	0x6A3A-0x6A3F	0x6A5A-0x6A5F	0x6A7A-0x6A7F	0x6A9A-0x6A9F	0x6ABA-0x6ABF	6	保留区

4.6 增强型正交编码器 (eQEP)

该器件包括高达 2 个增强型正交编码器(eQEP)模块(eQEP1，eQEP2)。图 4-7 所示为 eQEP 模块框图。

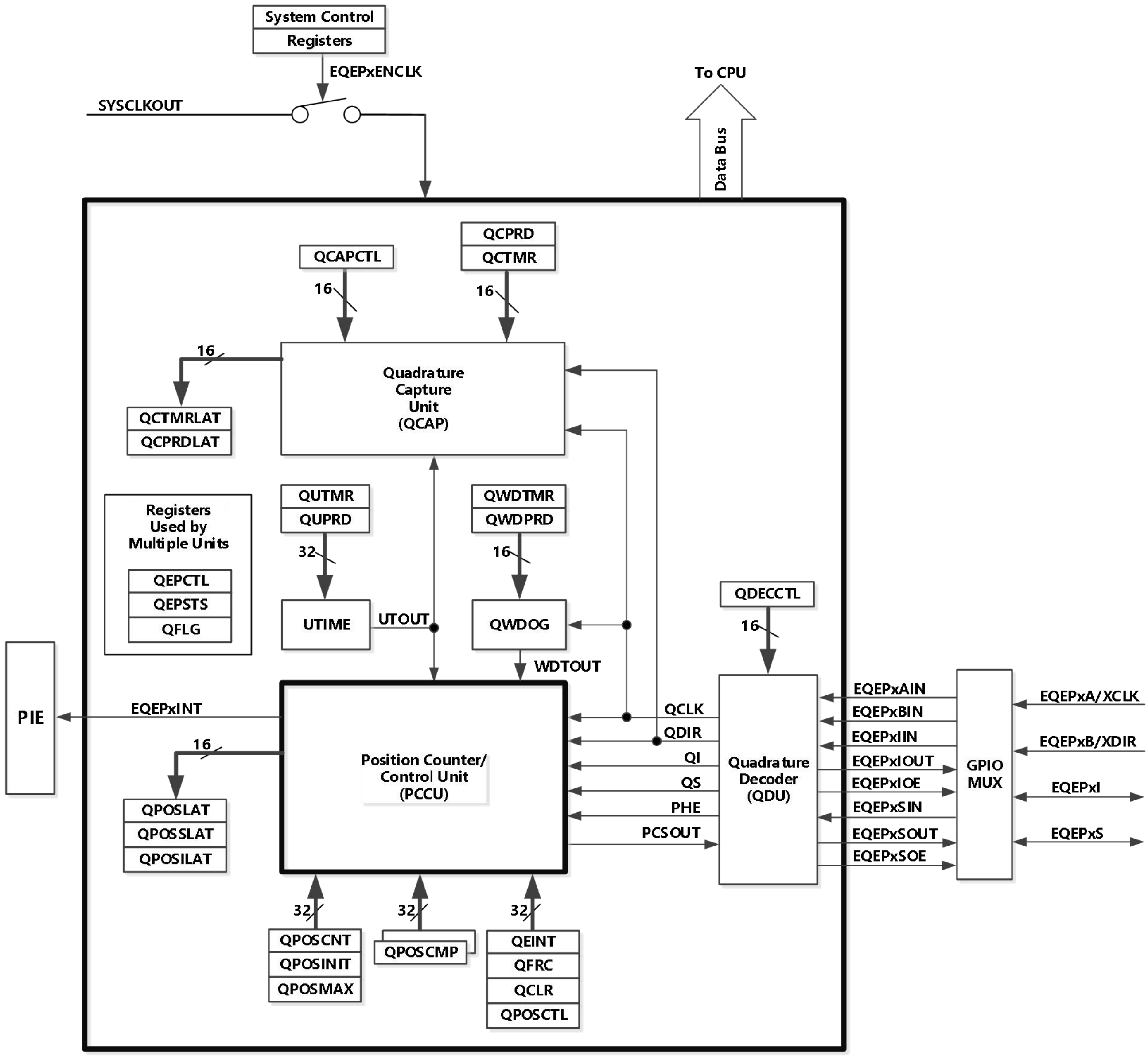


图 4-7 eQEP 功能框图

表 4-5 为 eQEP 寄存器的汇总表。

表 4-5 eQEP 控制和状态寄存器

名称	eQEP1 地址	eQEP2 地址	eQEP1 大小(x16)/ #SHADOW	说明
QPOSCNT	0x6B00	0x6B40	2/0	eQEP 位置计数器
QPOSINIT	0x6B02	0x6B42	2/0	eQEP 初始化位置计数
QPOSMAX	0x6B04	0x6B44	2/0	eQEP 最大位置计数
QPOSCMP	0x6B06	0x6B46	2/1	eQEP 位置比较
QPOSILAT	0x6B08	0x6B48	2/0	eQEP 索引位置锁存
QPOSSLAT	0x6B0A	0x6B4A	2/0	eQEP 选通脉冲位置锁存
QPOSLAT	0x6B0C	0x6B4C	2/0	eQEP 位置锁存
QUTMR	0x6B0E	0x6B4E	2/0	eQEP 单位定时器
QUPRD	0x6B10	0x6B50	2/0	eQEP 单位周期寄存器
QWDTMR	0x6B12	0x6B52	1/0	eQEP 看门狗定时器
QWDPRD	0x6B13	0x6B53	1/0	eQEP 看门狗周期寄存器
QDECCTL	0x6B14	0x6B54	1/0	eQEP 解码器控制寄存器
QEPCTL	0x6B15	0x6B55	1/0	eQEP 控制寄存器
QCAPCTL	0x6B16	0x6B56	1/0	eQEP 捕获控制寄存器
QPOSCTL	0x6B17	0x6B57	1/0	eQEP 位置比较控制寄存器
QEINT	0x6B18	0x6B58	1/0	eQEP 中断使能寄存器
QFLG	0x6B19	0x6B59	1/0	eQEP 中断标志寄存器
QCLR	0x6B1A	0x6B5A	1/0	eQEP 中断清除寄存器
QFRC	0x6B1B	0x6B5B	1/0	eQEP 中断强制寄存器
QEPSTS	0x6B1C	0x6B5C	1/0	eQEP 状态寄存器
QCTMR	0x6B1D	0x6B5D	1/0	eQEP 捕获定时器
QCPRD	0x6B1E	0x6B5E	1/0	eQEP 捕获周期寄存器
QCTMRLAT	0x6B1F	0x6B5F	1/0	eQEP 捕获定时器锁存
QCPRDLAT	0x6B20	0x6B60	1/0	eQEP 捕获周期锁存
保留区	0x6B21-0x6B3F	0x6B61-0x6B7F	31/0	保留区

4.7 ADC 特性

ADC 模块具有 16 个通道，并可配置为用于 ePWM 模块的 2 个独立的 8 通道模块。可将 2 个独立的 8 通道模块级联成 1 个 16 通道模块。尽管有多个输入通道和 2 个序列发生器，但在 ADC 模块中只有一个转换器。图 4–8 所示为 ADC 模块框图。

ADC 模块的功能包括：

- 具有内置 S/H 的12位ADC内核
- 模拟输入：0V - 3V（高于3.0V的电压会产生满量程转换结果）
- 快速转换速率：

  12位输出数据，在12.5 MHz ADC时钟，6.25百万次采样每秒钟（MSPS）时最快转换速度高达160 ns
- 16 个输入通道，多路复用输入
- 自动排序功能可在单个会话中提供多达16个“自动转换”。可以对每个转换进行编程，以选择16个输入通道中的任何一个
- 可将排序器作为 2 个独立的 8 态排序器，或作为 1 个较大的 16 态排序器使用（即 2 个级联的 8态排序器）
- 16个结果寄存器（可独立寻址）存储转换值

  - 输入模拟电压的数字值由以下公式得出：

$$\text{Digital Value} = 0 \qquad when\ input \le 0V$$

$$\text{Digital Value} = 4096 \times \frac{\text{Input Analog Voltage} - \text{ADCLO}}{3} \qquad when\ 0V < input < 3V$$

$$\text{Digital Value} = 4095 \qquad when\ input \ge 3V$$

  - 所有分数值均截断（小数位都舍去）。
- 作为序列转换开始的多个触发源

  - S/W -软件立即启动
  - ePWM 触发ADC开始转换
  - XINT2 ADC 触发ADC开始转换
- 灵活的中断控制方式允许每个EOS或每隔一个EOS产生中断请求
- 排序器可运行于“启动/停止”模式，以便多个“时序触发器”进行同步转换
- SOCA和SOCB触发源可以在双排序器模式下独立运行
- 采样保持（S/H）的采样时间窗口具有单独的预分频控制

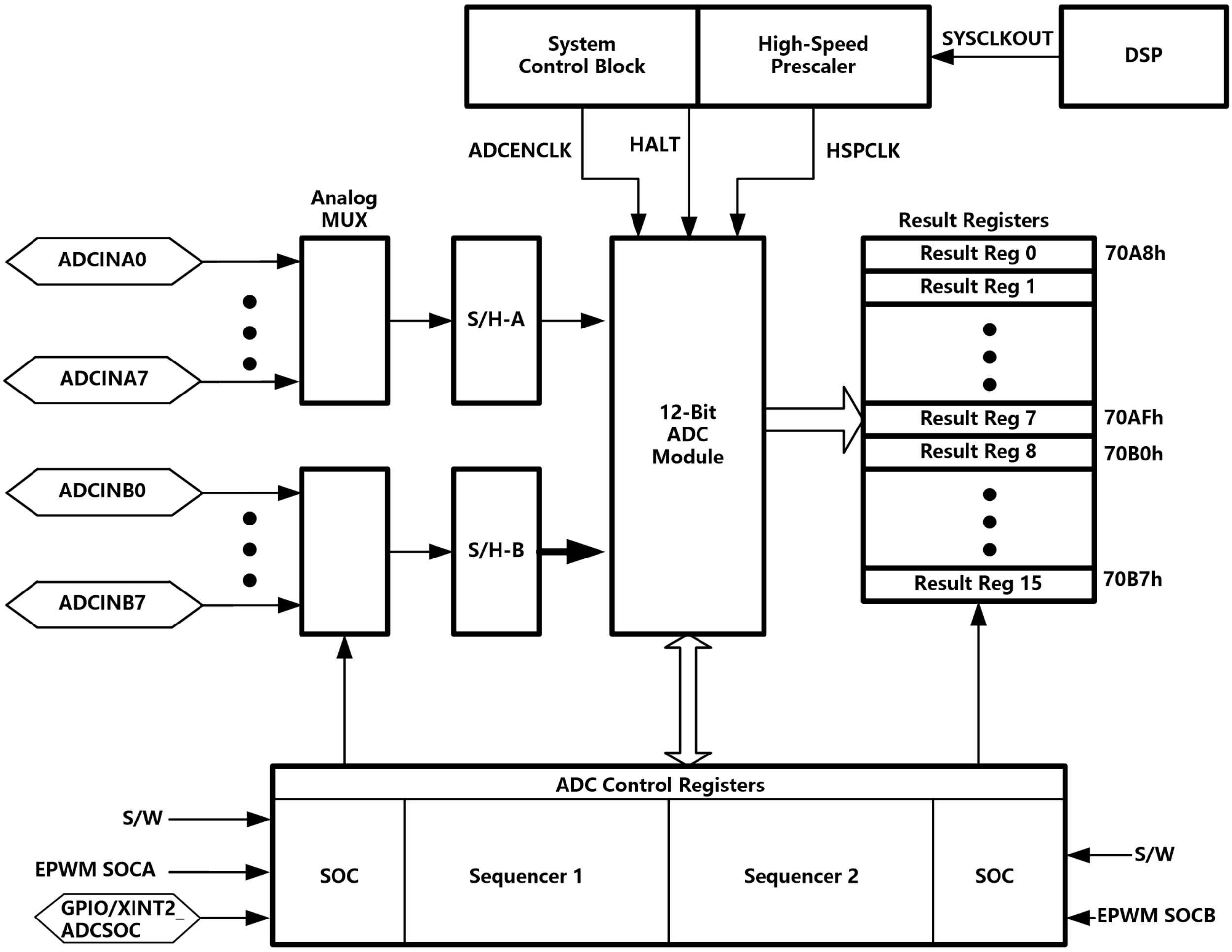


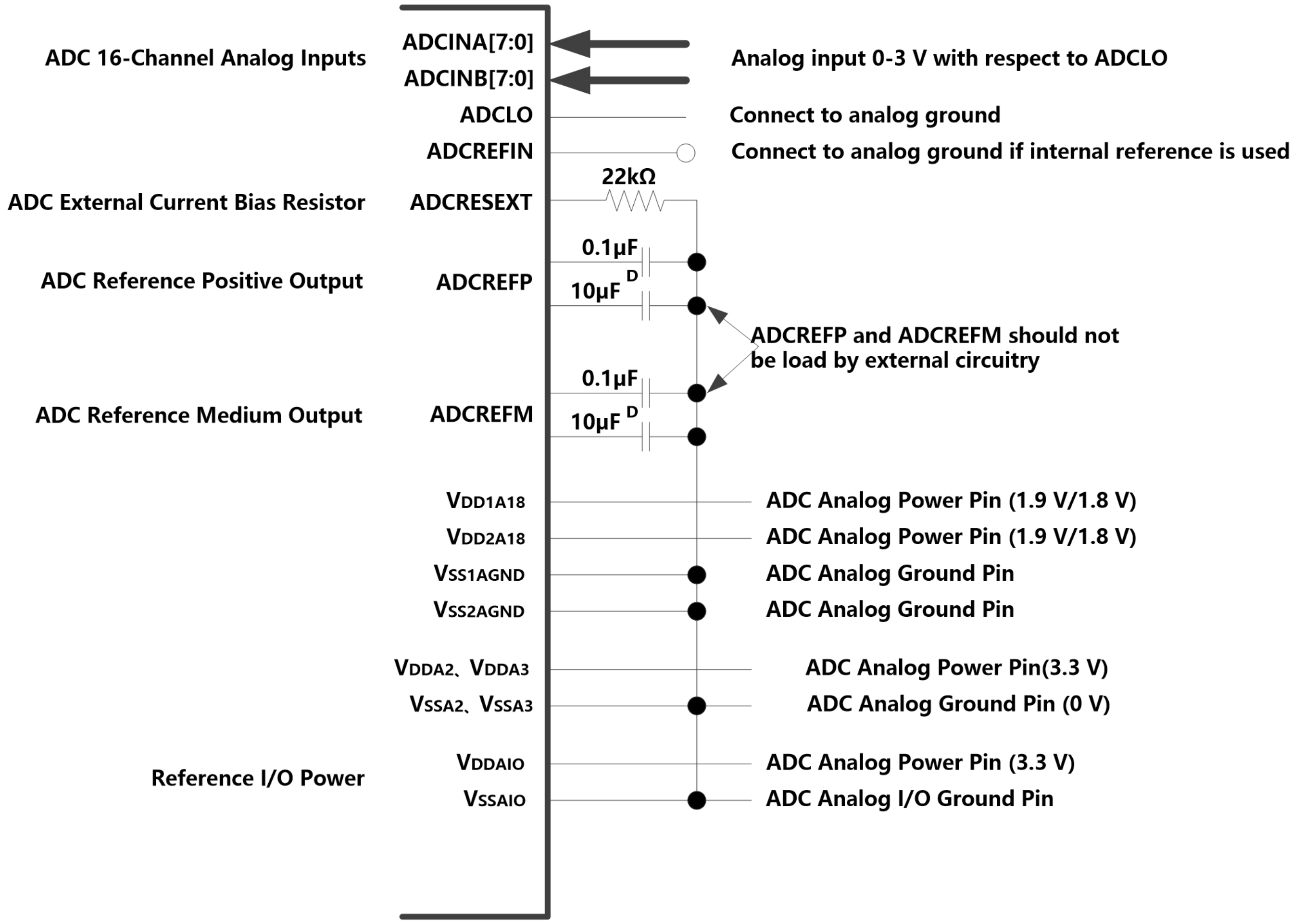
图 4-8 ADC 模块的方框图

要获得指定的 ADC 精度，正确的电路板布局非常关键。为尽可能达到最佳效果，引入 ADCIN 引脚的走线不应太靠近数字信号通道。这是为了最大程度地减少数字线路上耦合到 ADC 输入的开关噪声。而且，必须使用适当的隔离技术将 ADC 模块电源引脚(V_{DD1A18}，V_{DD2A18}，V_{DDA2}，V_{DDA3}，V_{DDAIO})从数字电源上隔离。图 4-9 所示为 ADC 的引脚连接。

注

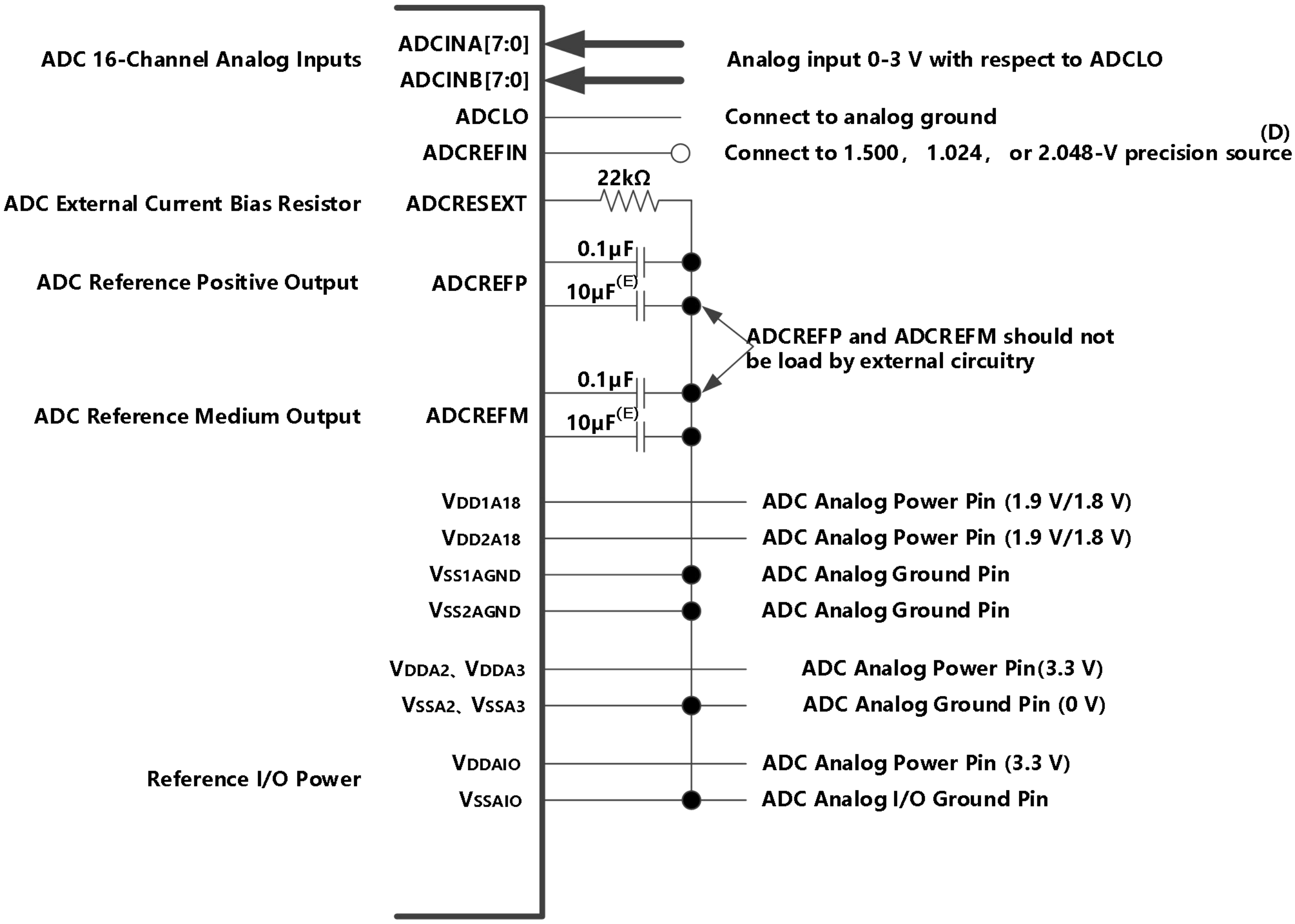
1. 用 SYSCLKOUT 速率对 ADC 寄存器进行访问。ADC 模块的内部时序由高速外设时钟(HSPCLK)控制。
2. 基于 ADCENCLK 和 HALT 信号的 ADC 模块的运行方式如下：
   - ADCENCLK：复位时，这个信号必须为低电平。虽然复位为低电平有效($\overline{XRS}$)，到寄存器的时钟将仍然起作用。有必要确保所有寄存器和模式进入它们的复位状态。然而，模拟模块将处于一个低功耗非激活状态。一旦复位变成高电平，那么到寄存器的时钟将被禁用。当用户将 ADCENCLK 信号设定为高电平时，那么到寄存器的时钟将被启用并且模拟模块将被启用。在 ADC 稳定并且可被使用之前，将有一个特定的时间延迟（毫秒范围内）。
   - HALT：这个模式只影响模拟模块。它不影响寄存器。在这个模式下，ADC 模块进入低功耗模式。这个模式将停止到 CPU 的时钟，即 HSPCLK；因此，将间接的关闭 ADC 逻辑。

图 4-9 显示了针对内部基准的 ADC 引脚偏置，而图 4-10 显示了针对外部基准的 ADC 引脚偏置。



A. 建议在所有电源引脚上使用外部去耦合电容器。
B. 必须从不会降低 ADC 性能的运算放大器上驱动模拟输入。
C. 所有电容都必须布局在距离芯片对应引脚最近的地方。
D. ADC时钟≤12.5MHz时需接10μF电容。

图 4-9 带有内部基准的 ADC 引脚连接



A. 建议在所有电源引脚上使用外部去耦合电容器。
B. 模拟输入必须由一个运算放大器驱动，此运算放大器不会降低 ADC 性能。
C. 所有电容都必须布局在距离芯片对应引脚最近的地方。
D. 根据这个引脚上的电压，通过改变 ADC 基准选择寄存器中的位 15:14 可启用 ADCREFIN 上的外部电压。总体增益精度将由这个电压源的精度确定。
E. ADC时钟≤12.5MHz时需接10μF电容。

图 4-10 带有外部基准的 ADC 引脚连接

注

任何推荐组件的额定温度必须与最终产品的额定值相匹配。

4.7.1 如果 ADC 未被使用，ADC 连接方式

建议保持针对模拟电源引脚的连接，即便在 ADC 未被使用时也是如此。下面总结了如果 ADC 未在应用中使用，应该如何连接 ADC 引脚：

- V_{DD1A18}/V_{DD2A18}- 连接至 V_{DD}
- V_{DDA2}，V_{DDA3}，V_{DDAIO}- 连接至 V_{DDIO}
- $V_{SS1AGND}/V_{SS2AGND}$，V_{SSA2}，V_{SSAIO}，V_{SSA3}，- 连接至 V_{SS}
- ADCLO - 连接至 V_{SS}
- ADCREFIN - 连接至 V_{SS}
- ADCREFP/ADCREFM - 连接一个 100nF 电容器至 V_{SS}
- ADCRESEXT - 连接一个 20kΩ 电阻器（阻值偏差容忍度很大）至 V_{SS}。
- ADCINAn，ADCINBn - 连接至 V_{SS}

当 ADC 未被使用时，为了达到节能的目的，请确保到 ADC 模块的时钟未被打开。当在一个应用中使用 ADC 模块时，未使用的 ADC 输入引脚应被连接至模拟地 ($V_{SS1AGND}/V_{SS2AGND}$)

注

只有当 ADC 的校准程序从引导 ROM 执行时，ADC 的增益误差和偏移误差参数才为额定值。 更多信息请查阅第 4.7.3 节。

4.7.2 ADC 寄存器

表 4-6 中所列寄存器为 ADC 操作相关的寄存器。

表 4-6 ADC 寄存器

名称	地址(1)	地址(2)	大小(x 16)	说明
ADCTRL1	0x7100		1	ADC 控制寄存器1
ADCTRL2	0x7101		1	ADC 控制寄存器2
ADCMAXCONV	0x7102		1	ADC 最大转换通道数寄存器
ADCCHSELSEQ1	0x7103		1	ADC 通道选择定序控制寄存器1
ADCCHSELSEQ2	0x7104		1	ADC 通道选择定序控制寄存器2
ADCCHSELSEQ3	0x7105		1	ADC 通道选择定序控制寄存器3
ADCCHSELSEQ4	0x7106		1	ADC 通道选择定序控制寄存器4
ADCASEQSR	0x7107		1	ADC 自动定序状态寄存器
ADCRESULT0	0x7108	0x0B00	1	ADC 转换结果缓冲寄存器0
ADCRESULT1	0x7109	0x0B01	1	ADC 转换结果缓冲寄存器1
ADCRESULT2	0x710A	0x0B02	1	ADC 转换结果缓冲寄存器2
ADCRESULT3	0x710B	0x0B03	1	ADC 转换结果缓冲寄存器3
ADCRESULT4	0x710C	0x0B04	1	ADC 转换结果缓冲寄存器4
ADCRESULT5	0x710D	0x0B05	1	ADC 转换结果缓冲寄存器5
ADCRESULT6	0x710E	0x0B06	1	ADC 转换结果缓冲寄存器6

ADCRESULT7	0x710F	0x0B07	1	ADC 转换结果缓冲寄存器7
ADCRESULT8	0x7110	0x0B08	1	ADC 转换结果缓冲寄存器8
ADCRESULT9	0x7111	0x0B09	1	ADC 转换结果缓冲寄存器9
ADCRESULT10	0x7112	0x0B0A	1	ADC 转换结果缓冲寄存器10
ADCRESULT11	0x7113	0x0B0B	1	ADC 转换结果缓冲寄存器11
ADCRESULT12	0x7114	0x0B0C	1	ADC 转换结果缓冲寄存器12
ADCRESULT13	0x7115	0x0B0D	1	ADC 转换结果缓冲寄存器13
ADCRESULT14	0x7116	0x0B0E	1	ADC 转换结果缓冲寄存器14
ADCRESULT15	0x7117	0x0B0F	1	ADC 转换结果缓冲寄存器15
ADCTRL3	0x7118		1	ADC 控制寄存器3
ADCST	0x7119		1	ADC 状态寄存器
保留	0x711A- 0x711B		2	
ADCREFSEL	0x711C		1	ADC 基准选择寄存器
ADCOFFTRIM	0x711D		1	ADC 偏移调整寄存器
保留	0x711E 0x711F		2	

(1) 本列中的寄存器为外设帧 2 寄存器。

(2) ADC 结果寄存器是双映射。外设帧 2(0x7108-0x7117)中的位置为 2 等待状态，且为左对齐。外设帧 0 空间（0x0B00-0x0B0F）的位置对 CPU 访问是 1 等待状态和对于 DMA 访问是 0 等待状态，右对齐。在 ADC 的高速/连续转换使用期间，使用 0 等待状态位置进行 ADC 结果到用户内存的快速转换。

4.7.3 ADC 校准

ADC_cal()函数在出厂时固化到 OTP 存储器中。引导 ROM 自动调用 ADC_cal()函数来使用特定器件的校准数据来初始化 ADCREFSEL 和 ADCOFFTRIM 寄存器。正常运行期间，这个过程中会自动发生，无需用户进行任何操作。

如果在开发过程中，引导 ROM 被 编译器 绕过，那么 ADCREFSEL 和ADCOFFTRIM 就必须由用户进行初始化。

注

这些寄存器初始化失败将导致 ADC 的功能不符合技术规格。

如果系统复位或 ADC 模块使用 ADC 控制寄存器 1 中的 14 位复位，则必须重新调用 ADC_cal()函数。

4.8 多通道缓冲串行端口 (McBSP) 模块

McBSP 模块有以下特性:

- 全双工通信
- 允许连续数据流的双缓冲数据寄存器
- 用于接收和发送的独立成帧和时钟
- 外部移位时钟生成或者一个内部可设定频率移位时钟
- 包括 8、12、16、20、24 或者 32 位在内的广泛数据大小选择
- 以 LSB 或者 MSB 开头的 8 位数据传输
- 用于帧同步和数据时钟的可编程极性
- 高度可编程内部时钟和帧生成
- 到工业标准 CODEG、模拟接口芯片(AIC)、和其它串行连接的 A/D 和 D/A 器件的直接接口
- 与 SPI 兼容器件一起工作
- McBSP 支持下列应用接口:
  - T1/E1 帧
  - 符合 IOM-2 的器件
  - AC97 - 兼容器件（提供所需的多相位帧同步功能。）
  - IIS - 兼容器件
  - 串行外设接口(SPI)
- McBSP 时钟速率:

$$\mathrm{CLKG} = \frac{\mathrm{CLKSRG}}{1+\mathrm{CLKGDV}}$$

CLKSRG 时钟源可以是 LSPCLK、CLKX 或者 CLKR。串行端口性能受到 I/O 缓冲器开关速度的影响。内部预分频器必须根据使用的时钟源进行调整，外设速度将低于 I/O 缓冲器速度。

注

最大 I/O 引脚切换速度请见第 6 章。

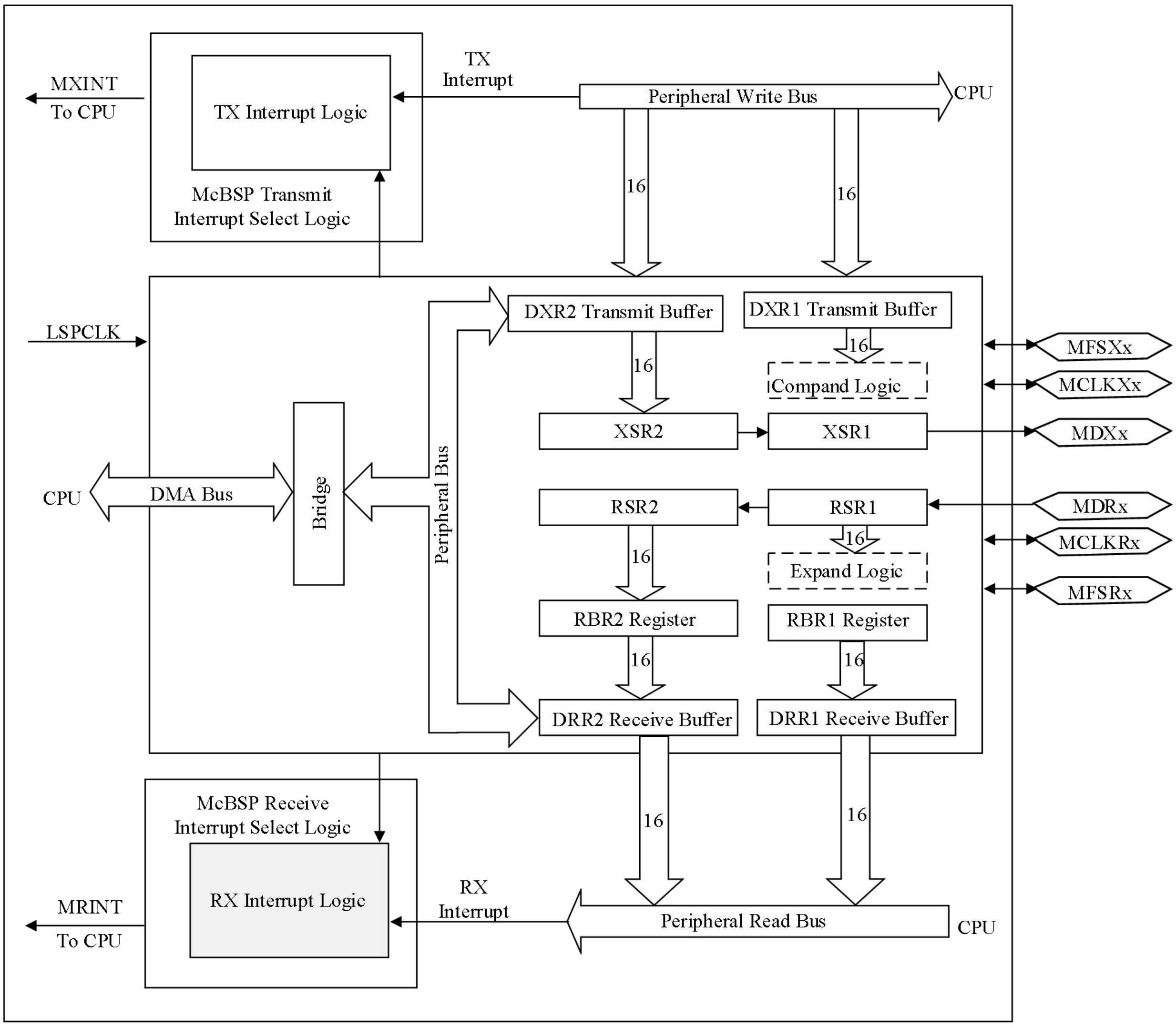


图 4-11 McBSP 模块框图

表 4-7 所示为 McBSP 寄存器列表。

表 4-7 McBSP 寄存器列表

名称	McBSP-A地址	McBSP-B地址	类型	复位值	说明
数据寄存器，接收、发送					
DRR2	0x5000	0x5040	读	0x0000	McBSP 数据接收寄存器2
DRR1	0x5001	0x5041	读	0x0000	McBSP 数据接收寄存器1
DXR2	0x5002	0x5042	写	0x0000	McBSP 数据发送寄存器2
DXR1	0x5003	0x5043	写	0x0000	McBSP 数据发送寄存器1
McBSP 控制寄存器					
SPCR2	0x5004	0x5044	读/写	0x0000	McBSP 串行端口控制寄存器2
SPCR1	0x5005	0x5045	读/写	0x0000	McBSP 串行端口控制寄存器1
RCR2	0x5006	0x5046	读/写	0x0000	McBSP 接收控制寄存器2
RCR1	0x5007	0x5047	读/写	0x0000	McBSP 接收控制寄存器1
XCR2	0x5008	0x5048	读/写	0x0000	McBSP 发送控制寄存器2
XCR1	0x5009	0x5049	读/写	0x0000	McBSP 发送控制寄存器1
SRGR2	0x500A	0x504A	读/写	0x0000	McBSP 采样率发生器寄存器2
SRGR1	0x500B	0x504B	读/写	0x0000	McBSP 采样率发生器寄存器1
多通道控制寄存器					
MCR2	0x500C	0x504C	读/写	0x0000	McBSP 多通道寄存器2
MCR1	0x500D	0x504D	读/写	0x0000	McBSP 多通道寄存器1
RCERA	0x500E	0x504E	读/写	0x0000	McBSP 接收通道使能寄存器分区A
RCERB	0x500F	0x504F	读/写	0x0000	McBSP 接收通道使能寄存器分区B
XCERA	0x5010	0x5050	读/写	0x0000	McBSP 发送通道使能寄存器分区A
XCERB	0x5011	0x5051	读/写	0x0000	McBSP 发送通道使能寄存器分区B
PCR	0x5012	0x5052	读/写	0x0000	McBSP 引脚控制寄存器
RCERC	0x5013	0x5053	读/写	0x0000	McBSP 接收通道使能寄存器分区C
RCERD	0x5014	0x5054	读/写	0x0000	McBSP 接收通道使能寄存器分区D
XCERC	0x5015	0x5055	读/写	0x0000	McBSP 发送通道使能寄存器分区C
XCERD	0x5016	0x5056	读/写	0x0000	McBSP 发送通道使能寄存器分区D
RCERE	0x5017	0x5057	读/写	0x0000	McBSP 接收通道使能寄存器分区E
RCERF	0x5018	0x5058	读/写	0x0000	McBSP 接收通道使能寄存器分区F
XCERE	0x5019	0x5059	读/写	0x0000	McBSP 发送通道使能寄存器分区E
XCERF	0x501A	0x505A	读/写	0x0000	McBSP 发送通道使能寄存器分区F
RCERG	0x501B	0x505B	读/写	0x0000	McBSP 接收通道使能寄存器分区G
RCERH	0x501C	0x505C	读/写	0x0000	McBSP 接收通道使能寄存器分区H
XCERG	0x501D	0x505D	读/写	0x0000	McBSP 发送通道使能寄存器分区G
XCERH	0x501E	0x505E	读/写	0x0000	McBSP 发送通道使能寄存器分区H
MFFINT	0x5023	0x5063	读/写	0x0000	McBSP 中断使能寄存器

4.9 增强型控制器局域网 (eCAN) 模块（eCAN-A 和 eCAN-B）

CAN 模块有下列特性：

- 与CAN协议，版本2.0B完全兼容
- 支持高达1 Mbps的数据速率
- 32个邮箱，每一个邮箱有下列属性：
  - 可配置为接收或者发送
  - 可使用标准或者扩展标识符进行配置
  - 有一个可编辑接收屏蔽
  - 支持数据和远程帧
  - 由0至8字节数据组成
  - 在接收和发送消息上使用一个32位时间戳
  - 防止接收新消息
  - 保持发送消息的动态可编辑优先级
  - 采用一个具有两个中断级别的可编辑中断机制
  - 在发送或者接收超时采用一个可编辑警报
- 低功耗模式
- 总线活动上的可编辑唤醒
- 对远程请求消息的自动答复
- 丢失仲裁或者错误情况下的帧自动重传
- 由一个特定消息同步的32 位本地网络时间计数器（与邮箱16 协同通信）
- 自测模式
  - 在接收其自身消息的回路模式下运行。用来提供一个“假”应答，从而无需由其他节点来提供响应位。

注

对于 100MHz 的 SYSCLKOUT，最小的比特率为 7.812 kbps；
对于 150MHz 的 SYSCLKOUT，最小的比特率为 11.719 kbps。

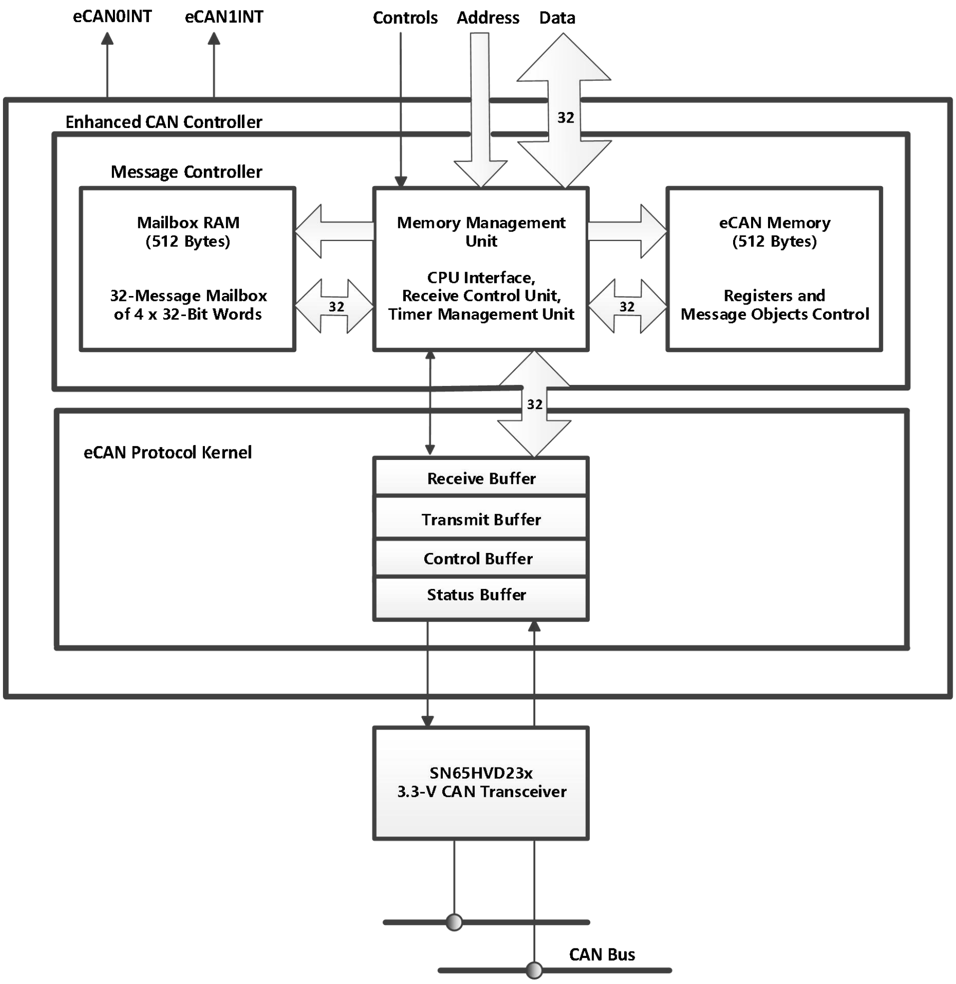


图 4-12 eCAN 框图和接口电路

表 4-8 3.3V eCAN 收发器

部件型号	电源电压	低功耗模式	斜坡控制	VREF	其它	T_A
SN65HVD230	3.3V	待机	可调节	支持	-	-40°C 至 85°C
SN65HVD230Q	3.3V	待机	可调节	支持	-	-40°C 至 125°C
SN65HVD231	3.3V	睡眠	可调节	支持	-	-40°C 至 85°C
SN65HVD231Q	3.3V	睡眠	可调节	支持	-	-40°C 至 125°C
SN65HVD232	3.3V	无	无	无	-	-40°C 至 85°C
SN65HVD232Q	3.3V	无	无	无	-	-40°C 至 125°C
SN65HVD233	3.3V	待机	可调节	无	诊断回路	-40°C 至 125°C
SN65HVD234	3.3V	待机和睡眠	可调节	无	-	-40°C 至 125°C
SN65HVD235	3.3V	待机	可调节	无	自动波特率回路	-40°C 至 125°C

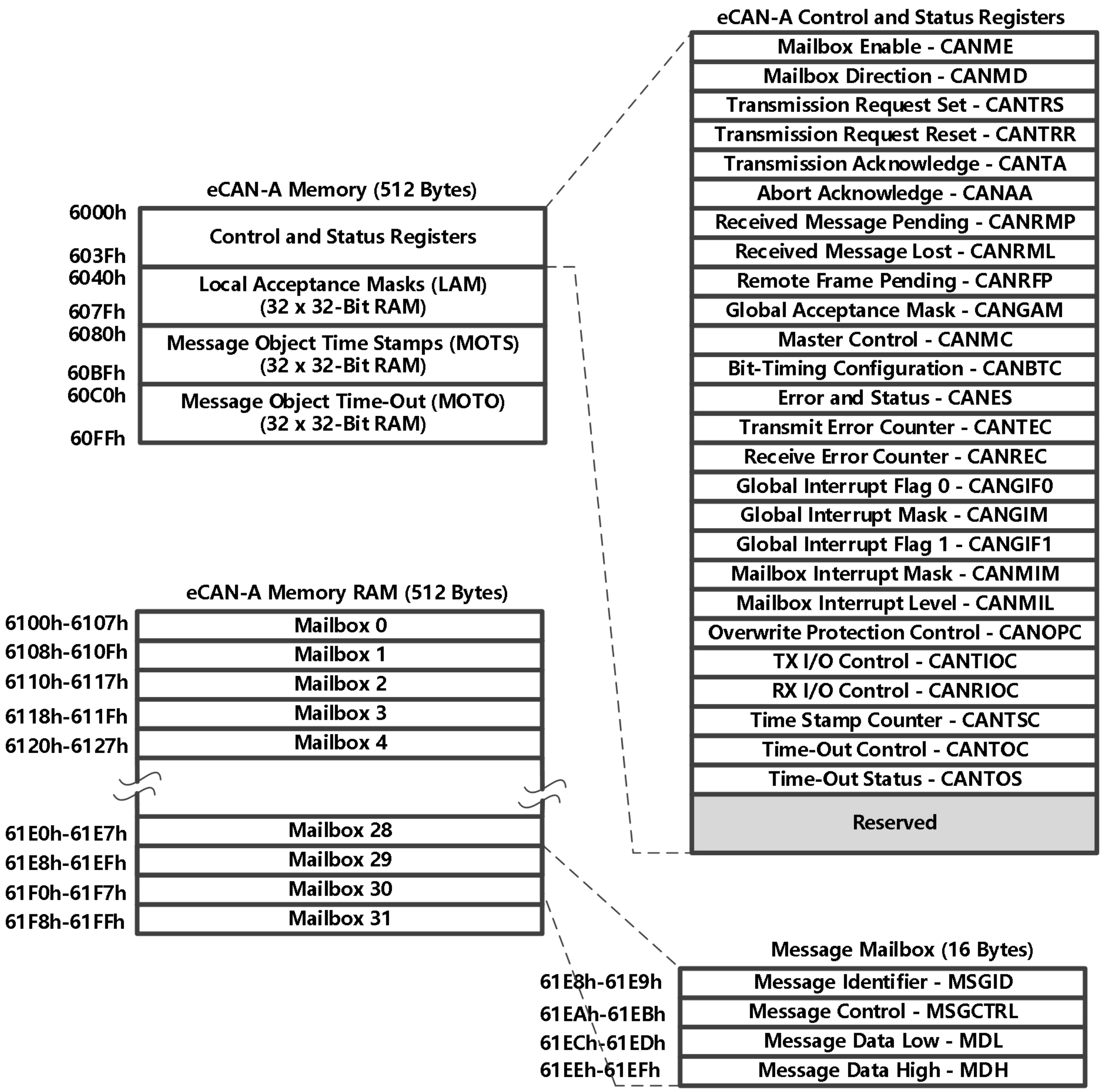


图 4-13 eCAN-A 内存映射

注

如果 eCAN 模块未在应用中使用，提供的 RAM （LAM，MOTS，MOTO，和邮箱 RAM）可被用作通用 RAM。为实现这一功能 CAN 模块时钟应被启用。

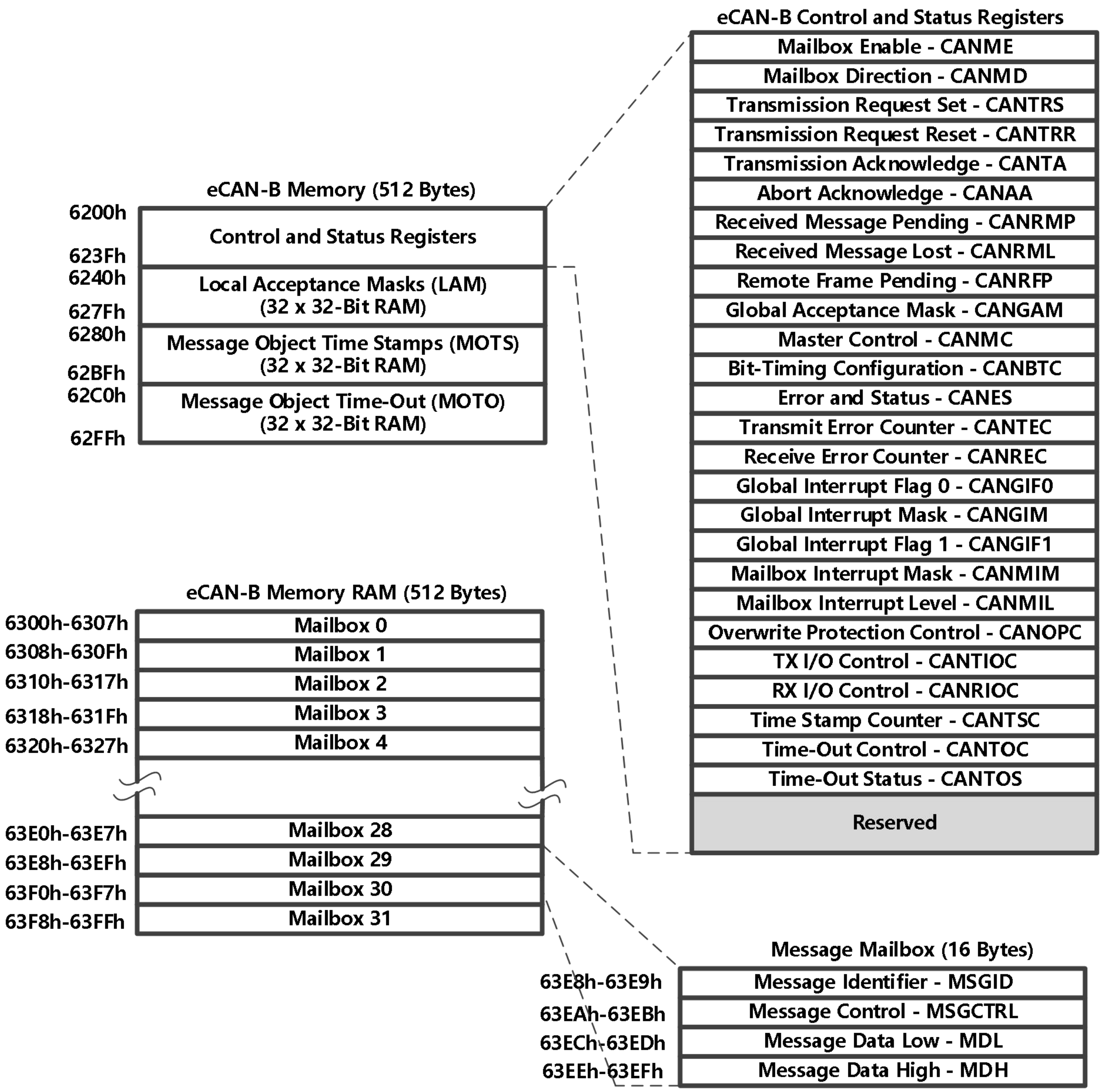


图 4-14 eCAN-B 内存映射

表4-9中列出的CAN 寄存器由CPU 用于配置和控制CAN 控制器和消息目标。eCAN 控制寄存器只支持32位读取/写入操作。邮箱RAM 可进行16 位或者32 位访问。32 位访问被对齐到一个偶数边界。

表 4-9 eCAN 寄存器映射[1]

名称	eCAN-A 地址	eCAN-B 地址	大小(x32)	说明
CANME	0x6000	0x6200	1	邮箱启用
CANMD	0x6002	0x6202	1	邮箱方向
CANTRS	0x6004	0x6204	1	发送请求设定
CANTRR	0x6006	0x6206	1	发送请求复位
CANTA	0x6008	0x6208	1	传输确认
CANAA	0x600A	0x620A	1	中止确认
CANRMP	0x600C	0x620C	1	接收消息等待
CANRML	0x600E	0x620E	1	接收消息丢失
CANRFP	0x6010	0x6210	1	远程帧等待
CANGAM	0x6012	0x6212	1	全局接收屏蔽
CANMC	0x6014	0x6214	1	主器件控制
CANBTC	0x6016	0x6216	1	位时序配置
CANES	0x6018	0x6218	1	错误和状态
CANTEC	0x601A	0x621A	1	发送错误计数器
CANREC	0x601C	0x621C	1	接收错误计数器
CANGIF0	0x601E	0x621E	1	全局中断标志0
CANGIM	0x6020	0x6220	1	全局中断屏蔽
CANGIF1	0x6022	0x6222	1	全局中断标志1
CANMIM	0x6024	0x6224	1	邮箱中断屏蔽
CANMIL	0x6026	0x6226	1	邮箱中断级别
CANOPC	0x6028	0x6228	1	写覆盖保护控制
CANTIOC	0x602A	0x622A	1	TX I/O 控制
CANRIOC	0x602C	0x622C	1	RX I/O 控制
CANTSC	0x602E	0x622E	1	时间戳计数器（保留在SCC 模式中）
CANTOC	0x6030	0x6230	1	超时控制（保留在SCC 模式中）
CANTOS	0x6032	0x6232	1	超时状态（保留在SCC 模式中）

(1) 这些寄存器被映射至外设帧 1。

4.10 异步串行通信接口 (SCI) 模块

AVP32F08器件包括两个异步串行通信接口(SCI)模块。SCI 模块支持CPU与其它异步外设之间的使用标准非归零码(NRZ) 格式的数字通信。SCI 接收器和发送器是双缓冲的，且各自有其独立的使能位和中断位。两者可独立运行或在全双工模式下同时运行。为了确保数据完整性，SCI在中断检测、奇偶校验、超限和成帧错误方面对接收到的数据进行检查。通过一个16 位波特率选择寄存器，可将比特率设定为超过65000种不同的波特率。

每个SCI模块的特性包括：

• 两个外部引脚：

- SCITXD：SCI发送-输出引脚
- SCIRXD：SCI接收-输入引脚

注释：两个引脚如果不被用于SCI 的话，可被用作GPIO。

• 波特率被设定为64K 个不同速率

$$\text{Baudrate}=\frac{\text{LSPCLK}}{(\text{BRR}+1)*8} \quad \text{when BRR}\neq 0$$

$$\text{Baudrate}=\frac{\text{LSPCLK}}{16} \quad \text{when BRR}=0$$

注

最大 I/O 引脚切换速度请见第 6 章。

•数据-字格式

– 一个开始位
– 数据-字长度可被设定为1 至8 位
– 可选偶/奇/无奇偶校验位
– 一个或者两个停止位

• 四个错误检测标志：奇偶、超限、成帧和中断检测
• 两个唤醒多处理器模式：空闲线和地址位
• 半双工或者全双工运行
• 双缓冲接收和发送功能
•可通过带有状态标志的中断驱动或者轮询算法来完成发送器和接收器操作

– 发送器：TXRDY 标志（发送器缓冲寄存器已经准备好接收另一个字符）和TX EMPTY标志（发送器移位寄存器已空）
– 接收器：RXRDY 标志（接收器缓冲寄存器已经准备好接收另一个字符），BRKDT 标志（发生了中断条件）和RX ERROR 错误标志（监测四个中断条件）

• 发送器和接收器中断的独立使能位（除了BRKDT）
• NRZ（非归零）码格式

注

所有处于这个模式中的寄存器是被连接至外设帧2 的8 位寄存器。当一个寄存器被访问时，低字节(7-0)，和高字节(15-8) 内的寄存器数据被读作零。对高字节的写入不起作用。

增强型特性:

- 自动波特率检测硬件逻辑电路
- 16级发送/接收FIFO

SCI 端口运行由表 4-10，表 4-11 中列出的寄存器配置和控制。

表 4-10 SCI-A 寄存器(1)

名称	地址	大小 (x16)	说明
SCICCRA	0x00 7050	1	SCI-A 通信控制寄存器
SCICTL1A	0x00 7051	1	SCI-A 控制寄存器1
SCIHBAUDA	0x00 7052	1	SCI-A 波特率寄存器，高位
SCILBAUDA	0x00 7053	1	SCI-A 波特率寄存器，低位
SCICTL2A	0x00 7054	1	SCI-A 控制寄存器2
SCIRXSTA	0x00 7055	1	SCI-A 接收状态寄存器
SCIRXEMUA	0x00 7056	1	SCI-A 接收仿真数据缓冲寄存器
SCIRXBUFA	0x00 7057	1	SCI-A 接收数据缓冲寄存器
SCITXBUFA	0x00 7059	1	SCI-A 发送数据缓冲寄存器
SCIFFTXA(2)	0x00 705A	1	SCI-A FIFO 发送寄存器
SCIFFRXA(2)	0x00 705B	1	SCI-A FIFO 接收寄存器
SCIFFCTA(2)	0x00 705C	1	SCI-A FIFO 控制寄存器
SCIPRIA	0x00 705F	1	SCI-A 优先级控制寄存器

(1) 这个表中的寄存器被映射到外设帧 2 空间。该空间只允许 16 位访问。32 位访问会产生未知结果。

(2) 这些寄存器是用于 FIFO 模式的全新寄存器。

表 4-11 SCI-B 寄存器(1) (2)

名称	地址	大小 (x16)	说明
SCICCRB	0x00 7750	1	SCI-B 通信控制寄存器
SCICTL1B	0x00 7751	1	SCI-B 控制寄存器1
SCIHBAUDB	0x00 7752	1	SCI-B 波特率寄存器，高位
SCILBAUDB	0x00 7753	1	SCI-B 波特率寄存器，低位
SCICTL2B	0x00 7754	1	SCI-B 控制寄存器2
SCIRXSTB	0x00 7755	1	SCI-B 接收状态寄存器
SCIRXEMUB	0x00 7756	1	SCI-B 接收仿真数据缓冲寄存器
SCIRXBUFB	0x00 7757	1	SCI-B 接收数据缓冲寄存器
SCITXBUFB	0x00 7759	1	SCI-B 发送数据缓冲寄存器
SCIFFTXB(2)	0x00 775A	1	SCI-B FIFO 发送寄存器
SCIFFRXB(2)	0x00 775B	1	SCI-B FIFO 接收寄存器
SCIFFCTB(2)	0x00 775C	1	SCI-B FIFO 控制寄存器
SCIPRIB	0x00 775F	1	SCI-B 优先级控制寄存器

(1) 此表中的寄存器被映射到外设帧 2 空间。此空间只允许 16 位访问。32 位访问会产生未知结果。

(2) 这些寄存器是用于 FIFO 模式的全新寄存器。

图4-15 所示为SCI模块框图。

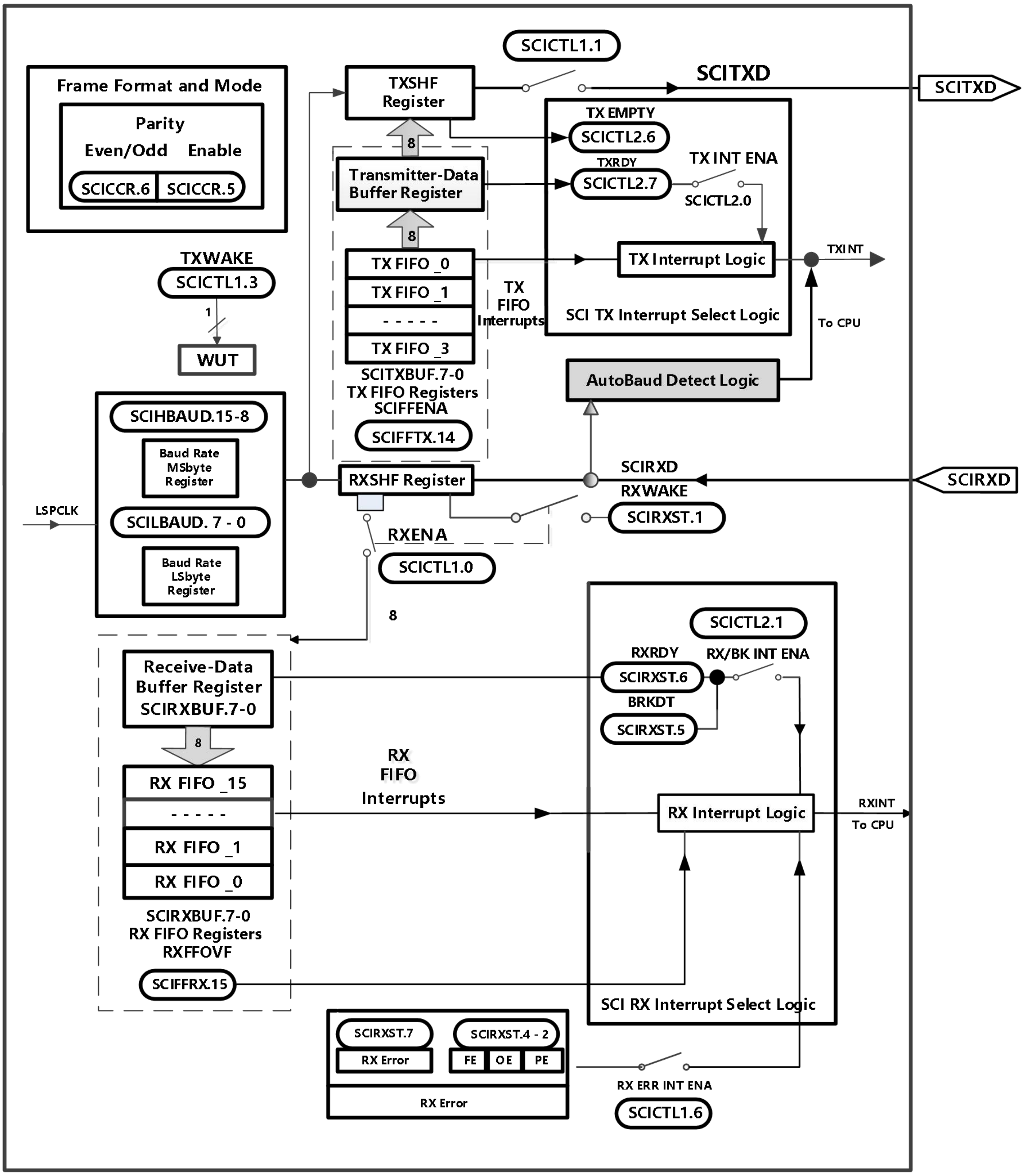


图 4-15 串行通信接口(SCI) 模块框图

4.11 串行通信接口 (SPI) 模块(SPI-A)

AVP32F08 器件包括四引脚的串行通信接口（SPI）模块。SPI是一个高速、同步串行I/O端口，此端口可在设定的位传输速率上将一个设定长度（1至16位）的串行比特流移入和移出器件。通常，SPI用于DSP 和外部外设或者其它处理器之间的通信。典型应用包括外部I/O 或者从诸如移位寄存器、显示驱动器和ADC 等器件的外设扩展。多器件通信由SPI的主模式/从模式操作支持。

SPI 模块的特性包括：

- 四个外部引脚：
  - SPISOMI：SPI 从器件输出/主器件输入引脚
  - SPISIMO：SPI 从器件输入/主器件输出引脚
  - $\overline{\text{SPISTE}}$：SPI 从器件发送使能引脚
  - SPICLK：SPI 串行时钟引脚

注释：如果未使用SPI模块，所有四个引脚可被用作GPIO。

- 两个运行模式：主模式和从模式
- 波特率：125个不同的可编辑速率

$$\text{Baud Rate} = \frac{\text{LSPCLK}}{(\text{SPIBRR}+1)} \qquad \text{when SPIBRR=3} \sim 127$$

$$\text{Baud Rate} = \frac{\text{LSPCLK}}{4} \qquad \text{when SPIBRR=0,1,2}$$

- 数据字长度：1~16数据位
- 包括4种时钟模式（由时钟极性和时钟相位的位控制）：
  - 无相位延迟的下降沿：SPICLK高电平有效。SPI 在SPICLK信号的下降沿上发送数据，而在SPICLK信号的上升沿上接收数据。
  - 有相位延迟的下降沿：SPICLK高电平有效。SPI 在SPICLK信号下降沿的一半周期之前发送数据，而在SPICLK信号的下降沿上接收数据。
  - 无相位延迟的上升沿：SPICLK低电平无效。SPI 在SPICLK信号的上升沿上发送数据，而在SPICLK信号的下降沿上接收数据。
  - 有相位延迟的上升沿：SPICLK低电平无效。SPI 在SPICLK信号下降沿之前的半个周期发送数据，而在SPICLK信号的上升沿上接收数据。
- 同时接收和发送操作（发送功能可在软件中被禁用）
- 通过中断驱动或者轮询算法来完成发送器和接收器运行。
- 9 个SPI模块控制寄存器：位于控制寄存器内，帧开始地址7040h。

注

这个模块中的所有寄存器是被连接至外设帧2 的16 位寄存器。当一个寄存器被访问时，低字节(7-0)，和高字节(15-8) 内的寄存器数据被读作零。对高字节的写入无效。

增强型特性:

- 16 级发送/接收FIFO
- 经延迟的发送控制

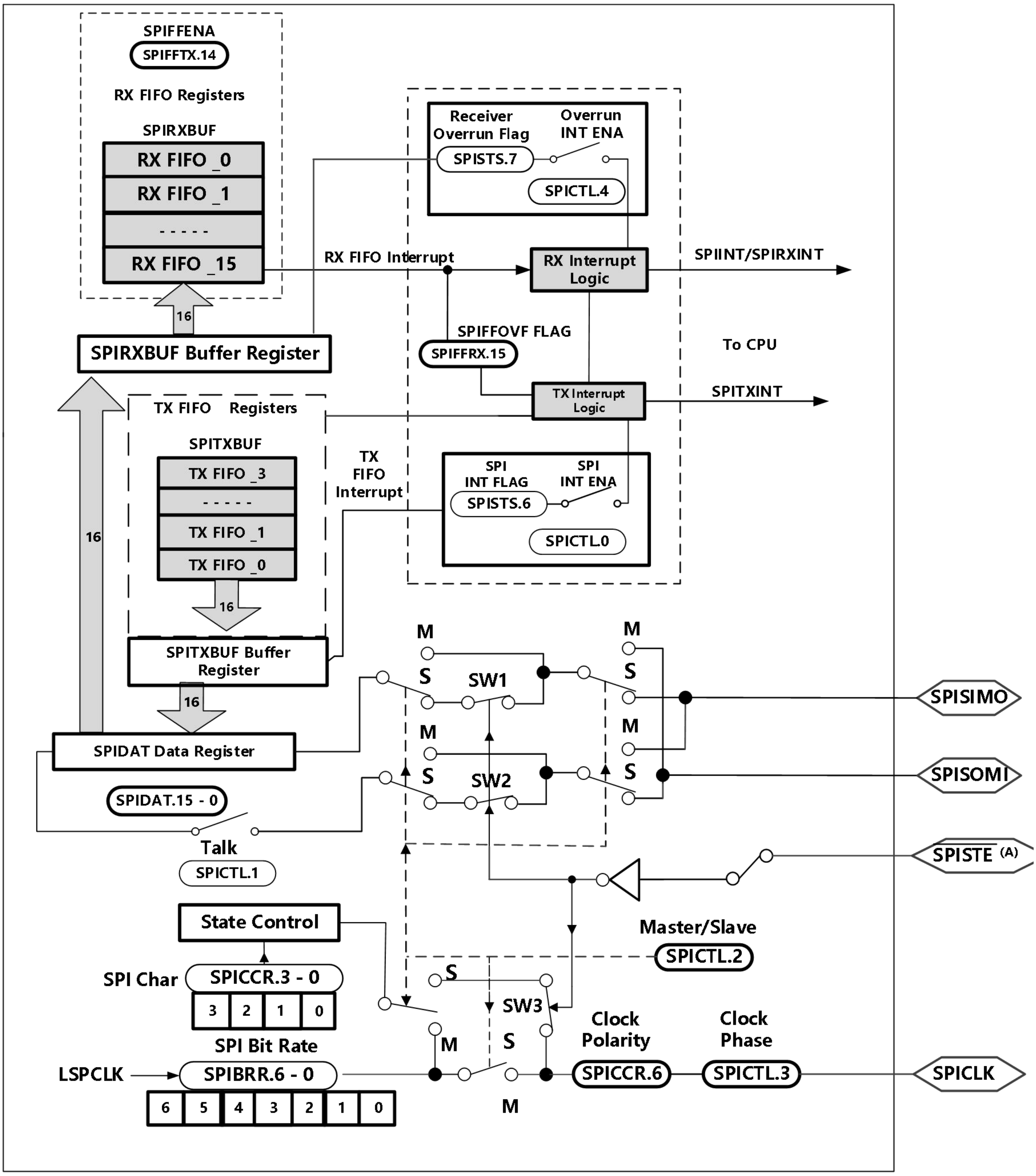
SPI 端口运行由表4-12中列出的寄存器配置和控制。

表 4-12 SPI-A 寄存器

名称	地址	大小（x16)	说明(1)
SPICCR	0x7040	1	SPI-A 配置控制寄存器
SPICTL	0x7041	1	SPI-A 运行控制寄存器
SPISTS	0x7042	1	SPI-A 状态寄存器
SPIBRR	0x7044	1	SPI-A 波特率寄存器
SPIRXEMU	0x7046	1	SPI-A 接收仿真缓冲器寄存器
SPIRXBUF	0x7047	1	SPI-A 串行输入缓冲器寄存器
SPITXBUF	0x7048	1	SPI-A串行输出缓冲器寄存器
SPIDAT	0x7049	1	SPI-A 串行数据寄存器
SPIFFTX	0x704A	1	SCI-A FIFO 发送寄存器
SPIFFRX	0x704B	1	SCI-A FIFO 接收寄存器
SPIFFCT	0x704C	1	SCI-A FIFO 控制寄存器
SPIPRI	0x704F	1	SCI-A 优先级控制寄存器

(1) 这些寄存器被映射至外设帧 2。该空间只允许 16 位访问。32 位访问会产生未知结果。

图4-16所示为从模式下的SPI框图。

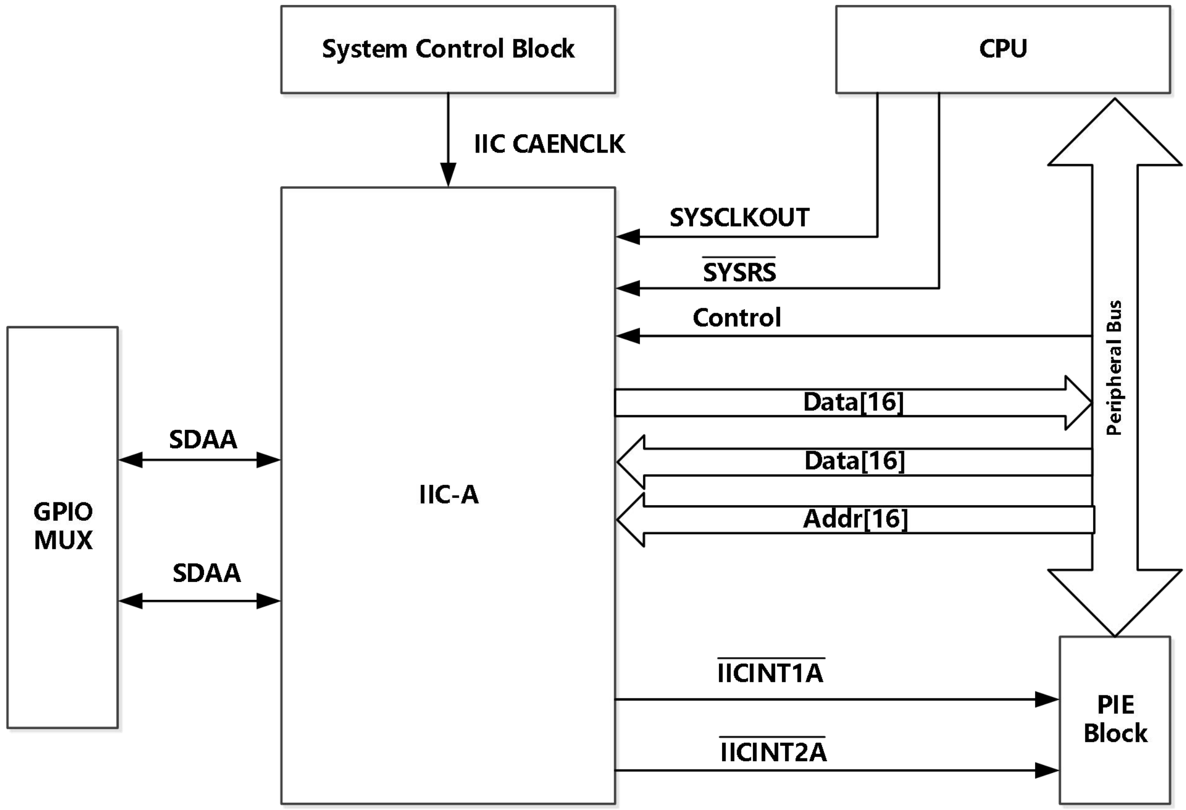


A $\overline{\text{SPISTE}}$低电平有效，作为从机的片选信号。

图 4-16 SPI 模块框图（从模式）

4.12 内部集成电路(IIC)

AVP32F08 包含一个 IIC 串行端口。图 4-17 显示了此器件内的 IIC 外设模块接口。



A.以 SYSCLKOUT 速率对 IIC 寄存器进行访问。IIC 端口的内部时序和信号波形也为 SYSCLKOUT 速率。
B.PCLKCR0寄存器内的时钟使能位 (IICAENCLK) 关闭到 IIC 端口的时钟以实现低功耗运行。复位时，IICAENCLK 被清除，这表明外设内部时钟被关闭。

图 4-17 IIC 外设模块接口

IIC 模块具有以下特性:

a) 符合飞利浦半导体 IIC 总线规格（版本 2.1）：
   1) 支持 1 位至 8 位格式传输
   2) 7 位和 10 位寻址模式
   3) 常规调用
   4) START 字节模式
   5) 支持多个主发送器和从接收器
   6) 支持多个从发送器和主接收器
   7) 组合主器件发送/接收和接收/发送模式
   8) 数据传输速率从 10kbps 到高达 400kbps（IIC 快速模式速率）

b) 一个 16 字接收 FIFO 和 一个 16 字发送 FIFO

c) 可以由 CPU 使用的一个中断。 这个中断可由下列条件中的一个生成：
    1) 发送数据准备好
    2) 接收数据准备好
    3) 寄存器访问准备好
    4) 接收 NACK
    5) 仲裁丢失
    6) 检测到停止条件
    7) 被寻址为从器件
d) 在 FIFO 模式下，CPU 可以使用附加的中断
e) 模块启用/禁用能力
f) 自由数据格式模式

表 4-13 中的寄存器配置并且扩展 IIC 端口操作。

表 4-13 IIC-A 寄存器

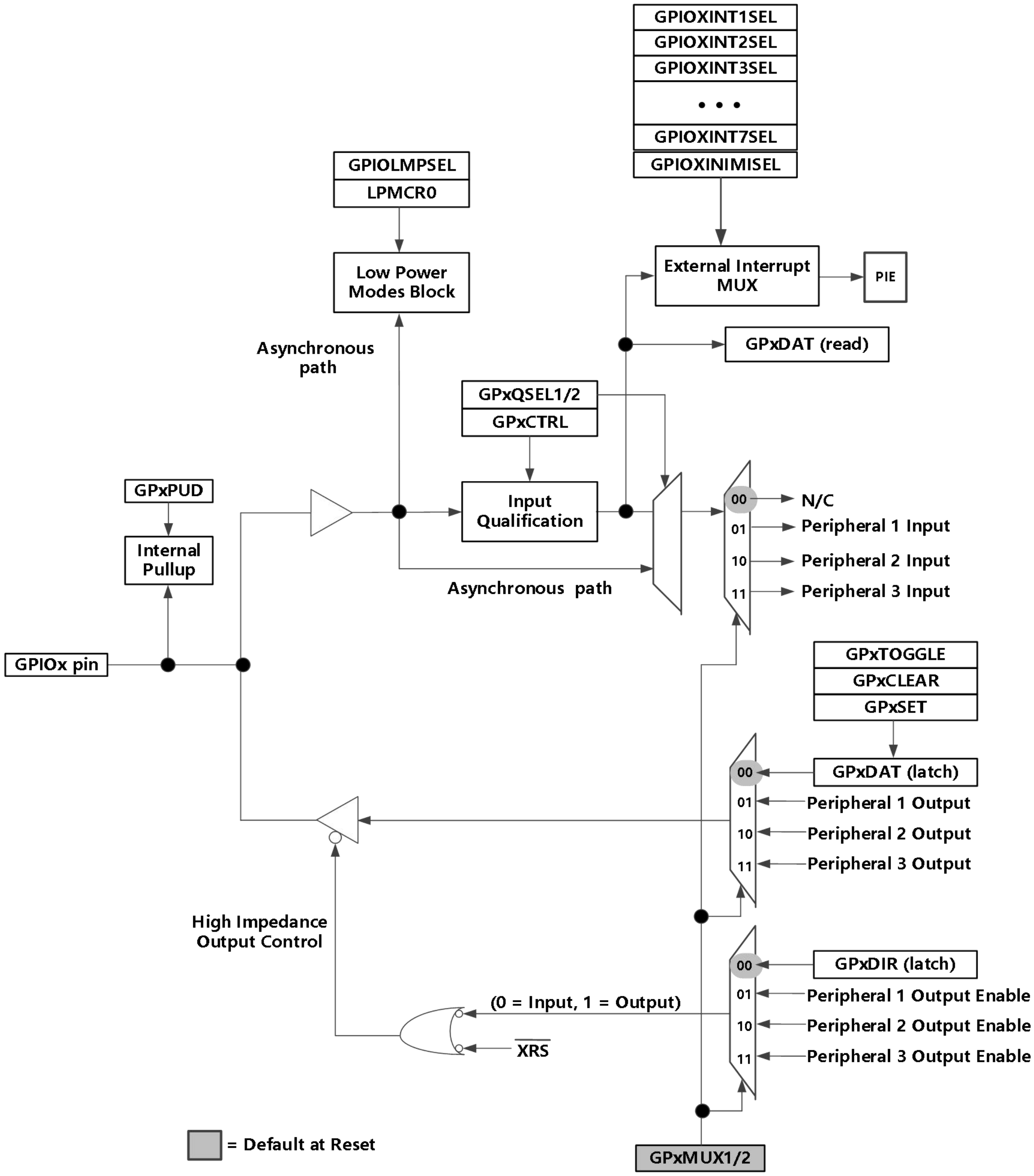
名称	地址	说明
IICOAR	0x7900	IIC 自身的地址寄存器
IICIER	0x7901	IIC 中断使能寄存器
IICSTR	0x7902	IIC 状态寄存器
IICCLKL	0x7903	IIC 时钟低电平时间分频器寄存器
IICCLKH	0x7904	IIC 时钟高电平时间分频器寄存器
IICCNT	0x7905	IIC 数据计数寄存器
IICDRR	0x7906	IIC 数据接收寄存器
IICSAR	0x7907	IIC 从器件地址寄存器
IICDXR	0x7908	IIC 数据发送寄存器
IICMDR	0x7909	IIC 模式寄存器
IICISRC	0x790A	IIC 中断源寄存器
IICPSC	0x790C	IIC 预分频器寄存器
IICFFTX	0x7920	IIC FIFO 发送寄存器
IICFFRX	0x7921	IIC FIFO 接收寄存器
IICRSR	-	IIC 接收移位寄存器（CPU不可访问）
IICXSR	-	IIC 发送移位寄存器（CPU不可访问）

4.13 GPIO MUX

在 AVP32F08 器件上，除了提供独立的 I/O 功能外，GPIO MUX 还可以将最多 3 个独立的外设信号多路复用到单个 GPIO 引脚上。每个引脚的 GPIO MUX 框图如图 4-18 所示。由于 IIC 引脚的开漏功能，这些引脚的 GPIO MUX 框图是不同的。

注

从写入 GPxMUXn 和 GPxQSELn 寄存器发生到动作有效有两个 SYSCLKOUT 周期延迟。



A. x 代表端口，A 或 B。例如，GPxDIR 是指 GPADIR 和 GPBDIR 寄存器，这取决于所选择的特定 GPIO 引脚。

B. 在相同的存储器位置存取 GPxDAT 锁定/读取。

C. 这是一个常用的 GPIO MUX 框图。并不是所有选项都可用于所有 GPIO 引脚。

图 4-18 GPIO 复用

AVP32F08 器件支持 35 个 GPIO 引脚。GPIO 控制和数据寄存器被映射到外设帧 1 以在寄存器上启用 32 位运行（连同 16 位运行）。表 4-14 显示了 GPIO 寄存器映射。

表 4-14 GPIO 寄存器

名称	地址	大小（x16）	说明
GPIO 控制寄存器（受 EALLOW 保护）			
GPACTRL	0x6F80	2	GPIO A 控制寄存器 (GPIO0 至 31)
GPAQSEL1	0x6F82	2	GPIO A 限定器选择 1 寄存器 (GPIO0 至 15)
GPAQSEL2	0x6F84	2	GPIO A 限定器选择 2 寄存器 (GPIO16 至 31)
GPAMUX1	0x6F86	2	GPIO A MUX 1 寄存器（GPIO0 至 15)
GPAMUX2	0x6F88	2	GPIO A MUX 2 寄存器（GPIO16 至 31)
GPADIR	0x6F8A	2	GPIO A 方向寄存器 (GPIO0 至 31)
GPAPUD	0x6F8C	2	GPIO A 上拉电阻器禁用寄存器 (GPIO0 至 GPIO31)
GPBCTRL	0x6F90	2	GPIO B 控制寄存器 (GPIO32 至 34)
GPBQSEL1	0x6F92	2	GPIO B 限定器选择 1 寄存器 (GPIO32 至 34)
GPBMUX1	0x6F96	2	GPIO B MUX 1 寄存器 (GPIO32 至 34)
GPBDIR	0x6F9A	2	GPIO B 方向寄存器 (GPIO32 至 34)
GPBBUD	0x6F9C	2	GPIO B 上拉禁用寄存器（GPIO32 至 34)
GPIO 数据寄存器（不受 EALLOW 保护）			
GPADAT	0x6FC0	2	GPIO A 数据寄存器（GPIO0 至 31)
GPASET	0x6FC2	2	GPIO A 数据设定寄存器（GPIO0 至 31)
GPACLEAR	0x6FC4	2	GPIO A 数据清除寄存器（GPIO0 至 31)
GPATOGGLE	0x6FC6	2	GPIO A 数据取反寄存器（GPIO0 至 31)
GPBDAT	0x6FC8	2	GPIO B 数据寄存器 (GPIO32 至 34)
GPBSET	0x6FCA	2	GPIO B 数据设定寄存器（GPIO32 至 34)
GPBCLEAR	0x6FCC	2	GPIO B 数据清除寄存器（GPIO32 至 34)
GPBTOGGLE	0x6FCE	2	GPIO B 数据取反寄存器（GPIO32 至 34)
GPIO 中断和低功耗模式选择寄存器（受 EALLOW 保护）			
GPIOXINT1SEL	0x6FE0	1	XINT1 GPIO 输入选择寄存器 (GPIO0 至 31)
GPIOXINT2SEL	0x6FE1	1	XINT2 GPIO 输入选择寄存器 (GPIO0 至 GPIO31)
GPIOXNIMISEL	0x6FE2	1	XNMI GPIO 输入选择寄存器 (GPIO0 至 GPIO31)
GPIOXINT3SEL	0x6FE3	1	XINT3 GPIO 输入选择寄存器 (GPIO32 至 GPIO34)
GPIOXINT4SEL	0x6FE4	1	XINT4 GPIO 输入选择寄存器 (GPIO32 至 GPIO34)
GPIOXINT5SEL	0x6FE5	1	XINT5 GPIO 输入选择寄存器 (GPIO32 至 GPIO34)
GPIOXINT6SEL	0x6FE6	1	XINT6 GPIO 输入选择寄存器 (GPIO32 至 GPIO34)
GPIOXINT7SEL	0x6FE7	1	XINT7 GPIO 输入选择寄存器 (GPIO32 至 GPIO34)
GPIOLPMSEL	0x6FE8	2	LPM GPIO 选择寄存器 (GPIO0 至 GPIO31)

表 4-15 GPIO-A 复用器外设选择矩阵

寄存器位			外设选择			
GPADIR GPADAT GPASET GPACLR GPATOGGLE		GPAMUX1 GPAQSEL1	GPIOx GPAMUX1 = 0,0	PER1 GPAMUX1 = 0,1	PER2 GPAMUX1 = 1,0	PER3 GPAMUX1 = 1,1
QUALPRD0	0	1, 0	GPIO0 (I/O)	EPWM1A (O)	-	-
	1	3, 2	GPIO1 (I/O)	EPWM1B (O)	ECAP6 (I/O)	MFSRB (I/O)
	2	5, 4	GPIO2 (I/O)	EPWM2A (O)	-	-
	3	7, 6	GPIO3 (I/O)	EPWM2B (O)	ECAP5 (I/O)	MCLKRB (I/O)
	4	9, 8	GPIO4 (I/O)	EPWM3A (O)	-	-
	5	11, 10	GPIO5 (I/O)	EPWM3B (O)	MFSRA (I/O)	ECAP1 (I/O)
	6	13, 12	GPIO6 (I/O)	EPWM4A (O)	EPWMSYNCI (I)	EPWMSYNCO (O)
	7	15, 14	GPIO7 (I/O)	EPWM4B (O)	MCLKRA (I/O)	ECAP2 (I/O)
QUALPRD1	8	17, 16	GPIO8 (I/O)	EPWM5A (O)	CANTXB (O)	$\overline{\text{ADCSOCA0}}$(O)
	9	19, 18	GPIO9 (I/O)	EPWM5B (O)	SCITXDB (O)	ECAP3 (I/O)
	10	21, 20	GPIO10 (I/O)	EPWM6A (O)	CANRXB (I)	$\overline{\text{ADCSOCB0}}$(O)
	11	23, 22	GPIO11 (I/O)	EPWM6B (O)	SCIRXDB (I)	ECAP4 (I/O)
	12	25, 24	GPIO12 (I/O)	$\overline{\text{TZ1}}$(I)	CANTXB (O)	MDXB (O)
	13	27, 26	GPIO13 (I/O)	$\overline{\text{TZ2}}$(I)	CANRXB (I)	MDRB (I)
	14	29, 28	GPIO14 (I/O)	$\overline{\text{TZ3}}$(I)	SCITXDB (O)	MCLKXB (I/O)
	15	31, 30	GPIO15 (I/O)	$\overline{\text{TZ4}}$(I)	SCIRXDB (I)	MFSXB (I/O)
		GPAMUX2 GPAQSEL2	GPAMUX2 = 0, 0	GPAMUX2 = 0, 1	GPAMUX2 = 1, 0	GPAMUX2 = 1, 1
QUALPRD2	16	1, 0	GPIO16 (I/O)	SPISIMOA (I/O)	CANTXB (O)	$\overline{\text{TZ5}}$(I)
	17	3, 2	GPIO17 (I/O)	SPISOMIA (I/O)	CANRXB (I)	$\overline{\text{TZ6}}$(I)
	18	5, 4	GPIO18 (I/O)	SPICLKA (I/O)	SCITXDB (O)	CANRXA (I)
	19	7, 6	GPIO19 (I/O)	$\overline{\text{SPISTEA}}$(I/O)	SCIRXDB (I)	CANTXA (O)
	20	9, 8	GPIO20 (I/O)	EQEP1A (I)	MDXA (O)	CANTXB (O)
	21	11, 10	GPIO21 (I/O)	EQEP1B (I)	MDRA (I)	CANRXB (I)
	22	13, 12	GPIO22 (I/O)	EQEP1S (I/O)	MCLKXA (I/O)	SCITXDB (O)
	23	15, 14	GPIO23 (I/O)	EQEP1I (I/O)	MFSXA (I/O)	SCIRXDB (I)
QUALPRD3	24	17, 16	GPIO24 (I/O)	ECAP1 (I/O)	EQEP2A (I)	MDXB (O)
	25	19, 18	GPIO25 (I/O)	ECAP2 (I/O)	EQEP2B (I)	MDRB (I)
	26	21, 20	GPIO26 (I/O)	ECAP3 (I/O)	EQEP2I (I/O)	MCLKXB (I/O)
	27	23, 22	GPIO27 (I/O)	ECAP4 (I/O)	EQEP2S (I/O)	MFSXB (I/O)
	28	25, 24	GPIO28 (I/O)	SCIRXDA (I)	$\overline{\text{TZ5}}$	
	29	27, 26	GPIO29 (I/O)	SCITXDA (O)	$\overline{\text{TZ6}}$	
	30	29, 28	GPIO30 (I/O)	CANRXA (I)	-	
	31	31, 30	GPIO31 (I/O)	CANTXA (O)	-	

表 4-16 GPIO-B 复用器外设选择矩阵

寄存器位			外设选择			
GPBDIR GPBDAT GPBSET GPBCLR GPBTOGGLE		GPBMUX1 GPBQSEL1	GPIOx GPBMUX1 = 0,0	PER1 GPBMUX1 = 0,1	PER2 GPBMUX1 = 1,0	PER3 GPBMUX1 = 1,1
QUALPRD0	0	1, 0	GPIO32(I/O)	SDAA (I/OC)(1)	EPWMSYNCI (I)	$\overline{\text{ADCSOCAO}}$(O)
	1	3, 2	GPIO33(I/O)	SCLA (I/OC)(1)	EPWMSYNCO (O)	$\overline{\text{ADCSOCBO}}$(O)
	2	5, 4	GPIO34 (I/O)	ECAP1(I/O)	-	

(1) I = 输入，O = 输出，OD = 开漏

通过配置 GPxQSEL1/2 寄存器，用户可为每一个 GPIO 引脚选择输入限定的类型：

a) 只同步至 SYSCLKOUT (GPxQSEL1/2=0,0)：这是复位时所有 GPIO 引脚的默认模式并且它只是将输入信号同步至系统时钟 (SYSCLKOUT)。

b) 使用采样窗口的限定条件（GPxQSEL1/2=0，1 和 1，0）：这个模式中，在与系统时钟 (SYSCLKOUT)同步后，输入信号在输入被允许改变前，被一定数量的周期所限定。

c) 采样周期由 GPxCTRL 寄存器内的 QUALPRD 位所指定并且可在每组 8 个信号中进行配置。它为输入信号采样指定了多个 SYSCLKOUT 周期。 一个采样窗口为可包含 3 次或 6 次采样，并且只有当所有采样与图 4-19（对于 6 次采样）中所显示的一样时（全 0 或者全 1），输出才会改变。

d) 无同步 (GPxQSEL1/2=1，1)：这个模式用于无需同步的外设（同步不在外设内执行）。

由于器件上所要求的多级复用，有可能会有一个外设输入信号被映射到多于一个 GPIO 引脚的情况。此外，当一个输入信号未被选择时，输入信号将默认为一个 0 或者 1 状态，这由外设而定。

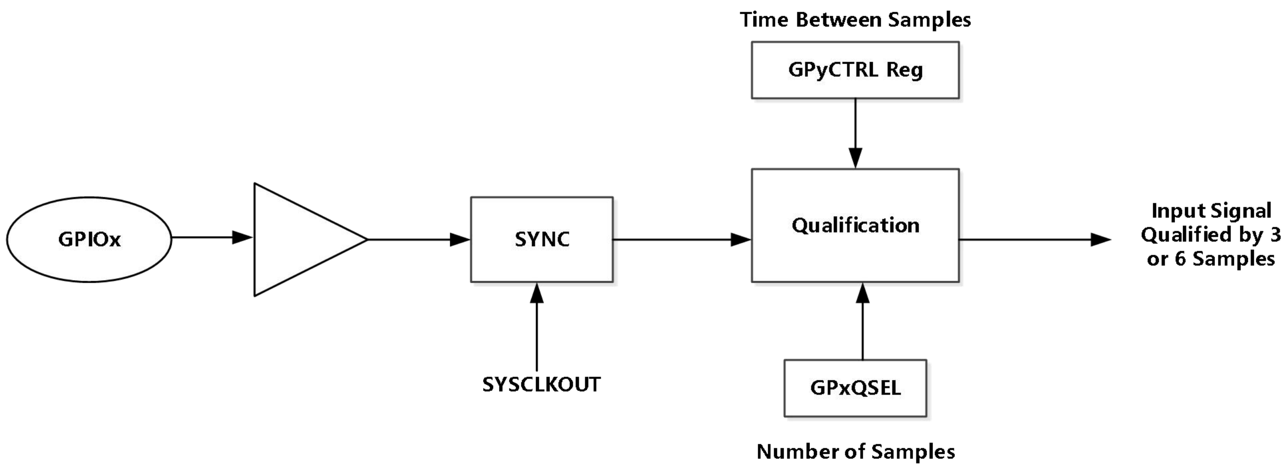


图 4-19 使用采样窗口的限定

5 开发支持

AVP32F08 产品支持多种开发工具，其中包括评估处理器性能、生成代码、开发算法执行的工具，且完全集成以及调试软件和硬件模块。

下面的软件硬件工具支持基于 AVP32F08 的应用开发：

软件开发工具

a) 编译器集成开发环境(IDE)

1) C/C++编译器

2) 代码生成工具

3) 汇编器/连接器

4) 周期精确模拟器

b) 应用算法

c) 示例应用代码

硬件开发工具

a) AVP32F08 开发板

b) 支持基于JTAG 的仿真器 - XDS510，XDS110V1，XDS100V1

c) 通用5V 直流电源

d) 文档和仿真器连接线

6 电气规范

6.1 最大绝对额定值(1)(2)

电源电压范围 V_{DDIO}	相对于 V_{SSIO}	-0.3V 至 4.6V
电源电压范围 V_{DDA2}，V_{DDA3}，V_{DDAIO}	相对于 V_{SSAIO}	-0.3 V 至 4.6V
电源电压范围 V_{DD}, V_{DDOSC}	相对于 V_{SS}	-0.3 V 至 2.5V
电源电压范围 V_{DD1A18}，V_{DD2A18}	相对于 V_{SSAIO}	-0.3V 至 2.5V
电源电压范围 V_{DDOSC}	相对于 V_{SSOSC}	-0.3 V 至 2.5V
电源电压范围 V_{SSA2}，V_{SSA3}，V_{SSAIO}, $V_{SS1AGND}$，$V_{SS2AGND}$, V_{SSIO}, V_{SSOSC}	相对于 V_{SS}	-0.3V 至 0.3V
输入电压范围，V_{IN}		-0.3V 至 4.6V
输出电压范围，V_{O}		-0.3V 至 4.6V
输入钳制电流，I_{IK}(V_{IN}<0 或者 V_{IN}>V_{DDIO})(3)		±20mA
输出钳制电流，I_{OK}(V_{O}< 0 或者 V_{O}> V_{DDIO})		±20mA
结温范围，T_{J}(4)		-40°C 至 150°C
贮存温度范围，T_{stg}(4)		-65°C 至 150°C

(1) 除非另外说明，绝对最大额定值的列表在运行温度范围内指定。在超过那些绝对最大额定值下列出的条件下运行有可能会对器件造成永久损坏。这些仅为在应力额定值下的工作情况，对于额定值下的器件的功能性操作以及在超出第 6.2 节推荐的操作条件下的任何其它操作，在此并未说明。长时间运行在最大绝对额定条件下会影响设备的可靠性。所有电压值都是以 V_{SS} 为参考。

(2) 所有电压值都是相对于 V_{SS} 的值，除非额外注明。

(3) 每个引脚上的持续钳制电流为±2mA。

(4) 长期高温存储/或者在最大温度条件下长时间使用，会使器件总体使用寿命缩短。

6.2 建议的运行条件

在自然通风条件下的工作温度范围内（除非另有说明）

		最小值	典型值	最大值	单位
器件电源电压，I/O，V_{DDIO}		3.135	3.3	3.465	V
器件电源电压CPU，V_{DD}	150MHz	1.805	1.9	1.98	V
电源接地，V_{SS}, V_{SSIO}, V_{SSAIO}, V_{SSA2}, V_{SSA3}，$V_{SS1AGND}$, $V_{SS2AGND}$			0		V
ADC 电源电压(3.3V)，V_{DDA2}，V_{DDA3}，V_{DDAIO}		3.135	3.3	3.465	V
ADC 电源电压(1.8V)，V_{DD1A18}，V_{DD2A18}	150MHz	1.805	1.9	1.98	V
器件时钟频率（系统时钟），$f_{SYSCLKOUT}$	AVP32F08	2		150	MHz
高电平输入电压，V_{IH}	除X1 之外的所有输入	2		V_{DDIO}	V
	X1	0.7 * V_{DD}		V_{DD}	
低电平输入电压，V_{IL}	除X1 之外的所有输入			0.8	V
	X1			0.3 * V_{DD}+0.05	
高电平输出源电流，V_{OH}=2.4V，I_{OH}	除组 2 之外的所有I/O			-4	mA
	组2(1)			-8	
低电平输出灌电流，V_{OL}=V_{OL}（最大值），I_{OL}	除组 2 之外的所有I/O			4	mA
	组2(1)			8	
环境温度，T_A	S版本	-40		125	°C
	S1（筛选）	-55		125	°C
	Q 版本 (通过AEC-Q100认证)	-40		125	°C
结温，T_J		-40		150	°C

(1) 组 2 引脚如下：GPIO4, GPIO17, GPIO5, GPIO18, GPIO6, GPIO19, GPIO7, GPIO8, GPIO9, GPIO20, GPIO10, GPIO21, GPIO11, GPIO22, GPIO23, GPIO27, GPIO24, GPIO25, GPIO28, GPIO13, GPIO26, GPIO32, TDO, XCLKOUT, EMU0, EMU1。

6.3 电气特性

在推荐的运行条件下（除非额外注明）

<table>
<tr><th colspan="2">参数</th><th colspan="2">测试条件</th><th>最小值</th><th>典型值</th><th>最大值</th><th>单位</th></tr>
<tr><td colspan="2" rowspan="2">V_{OH} 高电平输出电压</td><td colspan="2">I_{OH}=I_{OH} 最大值</td><td>2.4</td><td></td><td></td><td rowspan="2">V</td></tr>
<tr><td colspan="2">I_{OH}=50μA</td><td>V_{DDIO}-0.2</td><td></td><td></td></tr>
<tr><td colspan="2">V_{OL} 低电平输出电压</td><td colspan="2">I_{OL}=I_{OL} 最大值</td><td></td><td></td><td>0.4</td><td>V</td></tr>
<tr><td rowspan="3">I_{IL}输入电流（低电平）</td><td rowspan="2">上拉电阻器被启用</td><td rowspan="2">V_{DDIO}=3.3V ，V_{IN}=0V</td><td>所有带上拉 IO（除$\overline{XRS}$引脚）</td><td>-120</td><td>-150</td><td>-200</td><td rowspan="3">μA</td></tr>
<tr><td>$\overline{XRS}$引脚</td><td></td><td>-160</td><td></td></tr>
<tr><td>下拉电阻器被启用</td><td colspan="2">V_{DDIO}=3.3V，V_{IN}=0V</td><td></td><td></td><td>±5</td></tr>
<tr><td rowspan="2">I_{IH}输入电流（高电平）</td><td>上拉电阻器被启用</td><td colspan="2">V_{DDIO}=3.3V，V_{IN}= V_{DDIO}</td><td></td><td></td><td>±5</td><td rowspan="2">μA</td></tr>
<tr><td>下拉电阻器被启用</td><td colspan="2">V_{DDIO}=3.3V，V_{IN}= V_{DDIO}</td><td>40</td><td>55</td><td>80</td></tr>
<tr><td colspan="2">I_{OZ} 输出电流，上拉电阻器或者下拉电阻器被禁用</td><td colspan="2">V_O=V_{DDIO} 或者 0V</td><td></td><td></td><td>±5</td><td>μA</td></tr>
<tr><td colspan="2">C_i 输入电容</td><td colspan="2"></td><td></td><td>2</td><td></td><td>pF</td></tr>
</table>

（1）除 $\overline{TRST}$，TDO，X1，X2 外其余数字 IO 口均带上拉电阻，$\overline{TRST}$内部带下拉电阻。

6.4 流耗

表 6-1 AVP32F08 在 150MHz SYSCLKOUT 运行条件下的流耗

模式	测试条件	I_{DD}		I_{DDIO}[1]		I_{DDA18}[2]		I_{DDA33}[3]	
		典型值[4]	最大值	典型值[4]	最大值	典型值[4]	最大值	典型值[4]	最大值
可用（闪存）[5]	下列的外设时钟被启用: a) ePWM1, ePWM2, ePWM3, ePWM4, ePWM5, ePWM6 b) eCAP1, eCAP2, eCAP3, eCAP4, eCAP5, eCAP6 c) eQEP1, eQEP2 d) eCAN-A e) SCI-A, SCI-B（FIFO 模式) f) SPI-A（FIFO 模式) g) ADC h) IIC i) CPU 定时器0，CPU 定时器1，CPU 定时器2 所有PWM 引脚被切换至150kHz。 所有I/O 引脚保持未连接状态[6]	130mA	145mA	14 mA	35 mA	45 mA	60 mA	9 mA	15 mA
IDLE（空闲)	闪存被断电。 XCLKOUT 被关闭。下列的外设时钟被启用: a) eCAN-A b) SCI-A c) SPI-A d) IIC	55mA	70mA	2.5mA	4mA	50uA	80uA	450uA	700uA
STANDBY（待机)	闪存被断电。 外设时钟被关闭。	7mA	10mA	2.5mA	4mA	50uA	80uA	450uA	700uA
HALT[7]	闪存被断电。 外设时钟被关闭。 输入时钟被禁用。[8]	300uA		2.5mA	4mA	50uA	80uA	450uA	700uA

(1) I_{DDIO}电流取决于 I/O 引脚上的电气负载。

(2) I_{DDA18}包括进入V_{DD1A18}和V_{DD2A18}引脚的电流。为了实现所显示的用于IDLE，STANDBY，和HALT 的I_{DDA18}电流，必须通过写入PCLKCR0 寄存器来明确关闭到 ADC模块的时钟。

(3) I_{DDA33}包括进入V_{DDA2}和V_{DDAIO} 和V_{DDA3}引脚的电流。

(4) TYP 数适用于常温和标称电压。125°C 时的最大值，和最大电压(V_{DD}= 2.0V；V_{DDIO}，V_{DDA3}，V_{DDA}=3.6V)。

(5) 当SARAM 运行相同的代码时，I_{DD} 会随着代码从0 等待状态运行而增加。

(6) 下面的操作在环路内完成:

a) 数据从SCI-A，SCI-B，SPI-A，McBSP-A，和eCAN-A 端口连续发出。
b) 执行乘法/加法运算。
c) 看门狗被复位。
d) ADC 正在执行持续转换。ADC 中的数据通过DMA 传送到SARAM。
e) GPIO19 被接通。

(7) HALT 模式I_{DD}电流将随温度非线性增加。

(8) 如果一个石英晶振或者陶瓷谐振器被用作时钟源，HALT 模式将关闭内部振荡器。

注

外设 - 器件中执行的 I/O 复用防止同时使用所有可用外设。这是因为不止一个外设功能可共用一个 I/O 引脚。然而，可同时打开到所有外设的时钟，虽然这一配置并无实际用途。如果这一操作完成，器件汲取的电流将大于流耗表中的额定值。

6.4.1 降低流耗

通过关闭外设模块的时钟可降低流耗。此外，可利用这三个低功耗模式的任意一个来进一步降低流耗。表6-2表明了通过关闭不同外设的时钟所实现的流耗降低的典型值。

表 6-2 不同外设的典型流耗（150MHz 上时）(1)

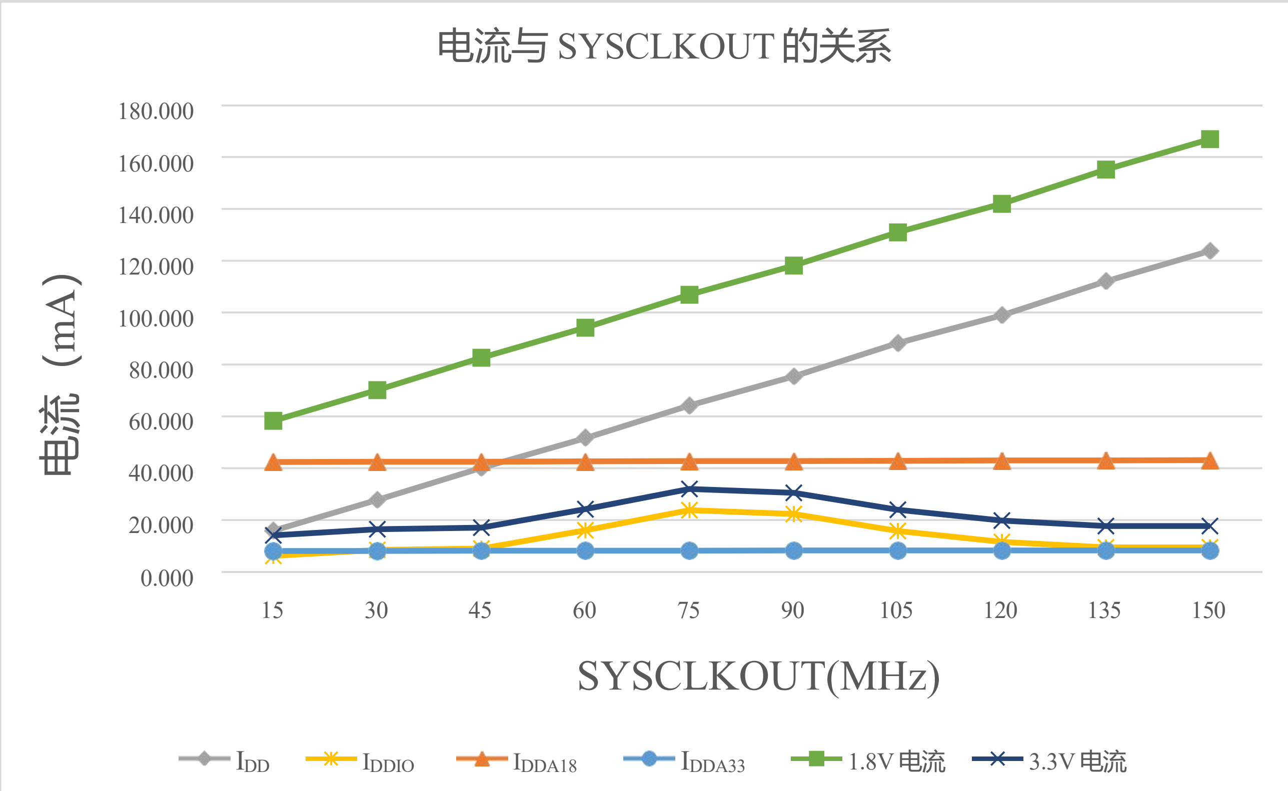
外设 模块	I_{DD} 电流 减少/模块(mA) (2)
ADC	2.9 (3)
IIC	1.3
eQEP	1.2
ePWM	1.9
eCAP	0.5
SCI	2.8
SPI	2.5
eCAN	3.8
McBSP	2.9
CPU - 定时器	0.6
DMA	4.7

(1) 复位时，所有外设时钟被禁用(除了 CPU 定时器时钟）。只有在外设时钟被打开后，才可对外设寄存器进行写入/读取操作。

(2) 对于具有多个实例的外设，每个实例化模块的电流如上表所示。例如，ePWM 减小的 1.9mA 电流只针对其中一个 ePWM 模块。

(3) 这个数字代表了取自ADC 模块数字部分的电流。关闭ADC 模块的时钟也将消除取自ADC (I_{DDA18}) 模拟部分吸收的电流。

6.4.2 流耗图



(1) I_{DD} 代表数字 1.8V 的电流。

(2) I_{DDIO} 代表数字 3.3V 的电流 。

(3) I_{DDA18} 代表模拟 1.8V 的电流，它包括 V_{DD1A18}和 V_{DD2A18}的电流。

(4) I_{DDA33} 代表模拟 3.3V 的电流，它包括 V_{DDAIO}和 V_{DDA2}和 V_{DDA3}的电流。

(5) 1.8V 电流：代表数字 1.8V 和模拟 1.8V 的总电流。

(6) 3.3V 电流：代表数字 3.3V 和模拟 3.3V 的总电流。

图 6-1 典型运行电流与频率间的关系 (AVP32F08)

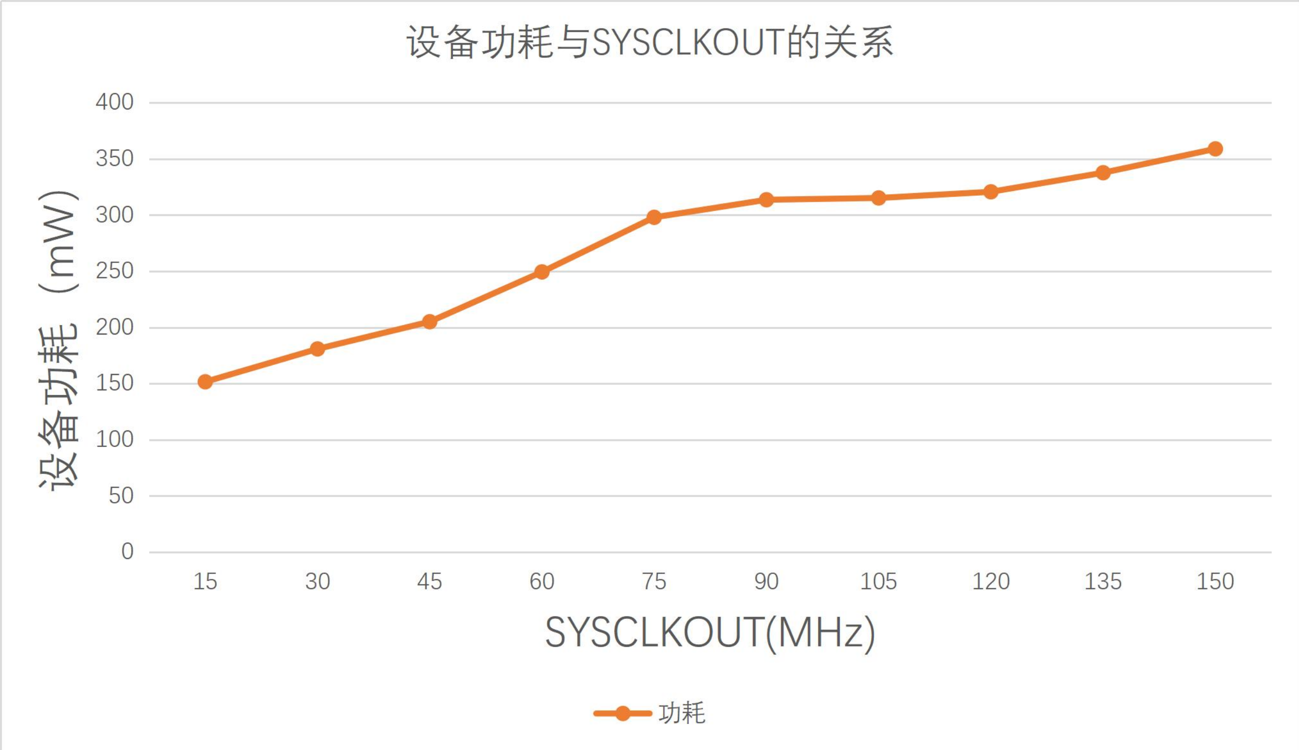


图 6-2 典型运行功率与频率间的关系 (AVP32F08)

器件的典型工作电流可通过图 6-1 估计。由图可知模拟的电流几乎保持不变。然而，可以预计到 I_{DDIO} 电流的少量下降，这是由外设引脚的外部活动减少造成的，电流的减少主要在 I_{DD} 中。

6.4.3 散热设计考虑

根据最终应用设计和运行情况，I_{DD} 和 I_{DDIO} 电流会不同。超过 1W功耗的系统可能需要一种产品级别的散热设计。因此，应该注意将T_J保持在额定限值内。在终端应用中，应当测量$T_{外壳温度}$，以此来估算工作结温T_J。$T_{外壳温度}$通常在封装顶部表面的中央进行测量。

6.5 针对 DSP 的无信号缓冲的仿真器连接

图 6-3 显示了DSP 和JTAG 接头之间针对单处理器配置的连接。如果JTAG接头和DSP之间的距离大于6英寸，那么仿真信号必须加缓冲器。如果距离小于6英寸，通常无需加缓冲器。图6-3显示了较简单、无缓冲的情况。对于上拉/下拉电阻器的值，请见表2-2引脚说明部分。

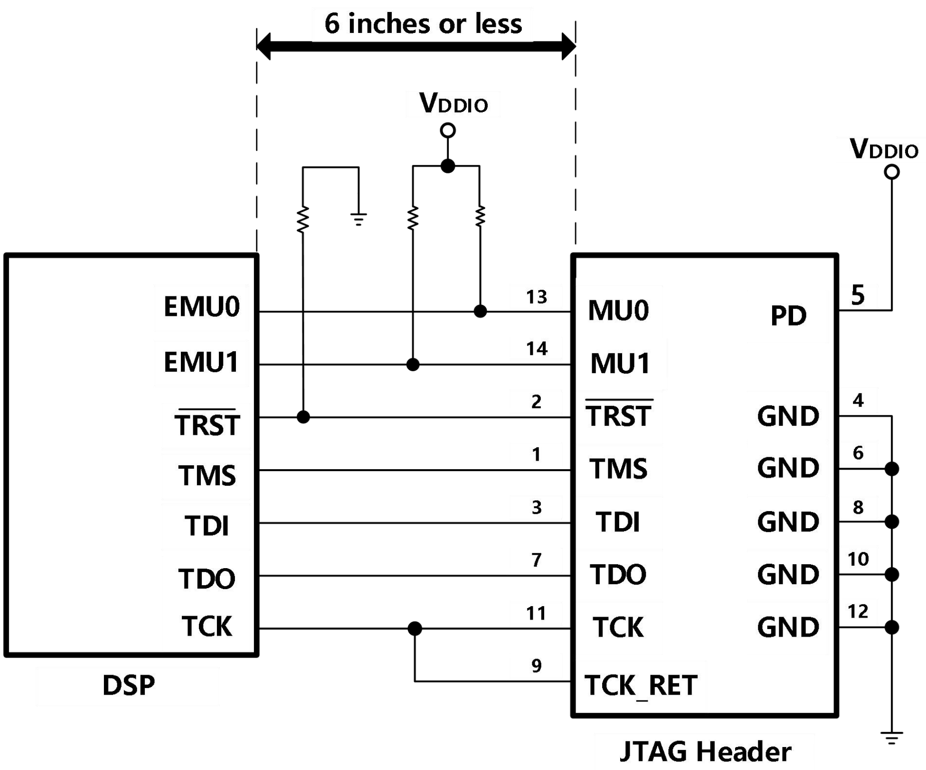


图 6-3 针对 DSP 的无信号缓冲的仿真器连接

6.6 时序参数符号

所用的时序参数符号按照 JEDEC-100 创建。为了缩短符号，有些引脚的名称和其它相关的术语名已经按如下方式缩写：

小写下标及其含意：	字母及其含意：
a 访问时间	H 高
c 周期时间（周期）	L 低
d 延迟时间	V 有效
f 下降时间	X 未知、改变、或者无关电平
h 保持时间	Z 高阻抗
r 上升时间	
su 建立时间	
t 转换时间	
v 有效时间	
w 脉冲持续时间（宽度）	

6.6.1 时序参数的通用说明

所有 AVP32F08 器件的输出信号（包括 XCLKOUT）由内部时钟进行同步。

6.6.2 测试负载电路

这个测试负载电路用于测量本文档中提供的所有开关特性。

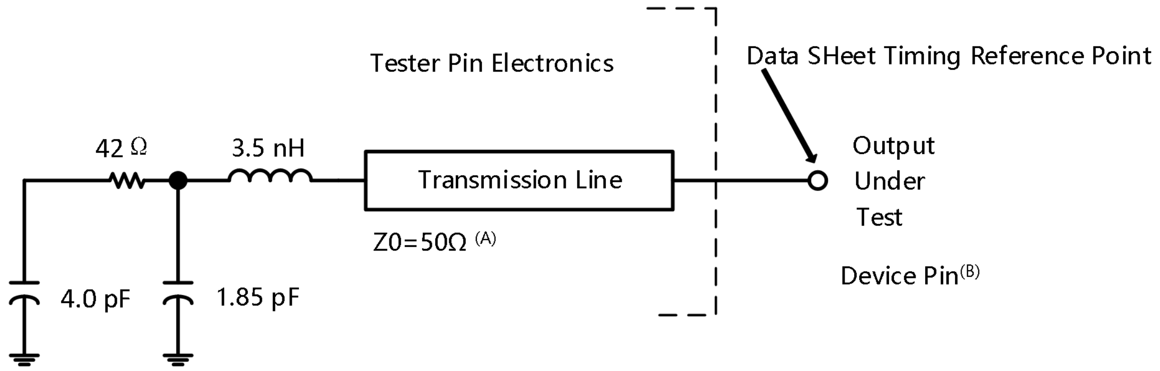


图 6-4 3.3V 测试负载电路

6.6.3 器件时钟表

这个部分提供针对DSP上可用的多种时钟选项的时序要求和开关特性。表6-3和表6-4列出了多种时钟的周期时间。

表 6-3 片上设备时钟的周期特征（150MHz 主时钟）

		最小值	典型值	最大值	单位
片载振荡器时钟	$t_{c(OSC)}$，周期时间	28.6		50	ns
	频率	20		35	MHz
XCLKIN(1)	$t_{c(CI)}$，周期时间	6.67		250	ns
	频率	4		150	MHz
SYSCLKOUT	$t_{c(SCO)}$，周期时间	6.67		500	ns
	频率	2		150	MHz
XCLKOUT	$t_{(XCO)}$，周期时间	6.67		2000	ns
	频率	0.5		150	MHz
HSPCLK(2)	$t_{c(LCO)}$，周期时间	6.67	13.3(3)		ns
	频率		75(3)	150	MHz
LSPCLK(2)	$t_{c(LCO)}$，周期时间	13.3	26.7(3)		ns
	频率		37.5(3)	75 (4)	MHz
ADC 时钟	$t_{c(ADCCLK)}$，周期时间	80			ns
	频率			12.5	MHz

(1) 如果使用一个 1.8V 振荡器，这也适用于 X1 引脚。

(2) 更低的 LSPCLK 和 HSPCLK 将减少器件功耗。

(3) 如果 SYSCLKOUT=150MHz，此值为缺省值。

(4) 尽管 LSPCLK 能够达到 100MHz，但由于对于 150MHz 器件，最小有效"低速外设时钟预分频寄存器"的值是"2"，所以它的额定值为 75MHz。

表 6-4 片上设备时钟的周期特征（100MHz 主时钟）

		最小值	典型值	最大值	单位
片载振荡器时钟	$t_{c(OSC)}$，周期时间	28.6		50	ns
	频率	20		35	MHz
XCLKIN(1)	$t_{c(CI)}$，周期时间	10		250	ns
	频率	4		100	MHz
SYSCLKOUT	$t_{c(SCO)}$，周期时间	10		500	ns
	频率	2		100	MHz
XCLKOUT	$t_{(XCO)}$，周期时间	10		2000	ns
	频率	0.5		100	MHz
HSPCLK(2)	$t_{c(HCO)}$，周期时间	10	20(3)		ns
	频率		50(3)	100	MHz
LSPCLK(2)	$t_{c(LCO)}$，周期时间	10	40(3)		ns
	频率		25(3)	100	MHz
ADC 时钟	$t_{c(ADCCLK)}$，周期时间	80			ns
	频率			12.5	MHz

(1) 如果使用一个 1.8V 振荡器，这也应用于 X1 引脚。

(2) 更低的 LSPCLK 和 HSPCLK 将减少器件功耗。

(3) 如果 SYSCLKOUT=100MHz，此值为默认值。

6.7 时钟要求和特性

表 6-5 输入时钟频率

参数		最小值	典型值	最大值	单位
f_X 输入时钟频率	谐振器（X1/X2）	20		35	MHz
	晶振（X1/X2）	20		35	MHz
	外部振荡器/时钟源（XCLKIN 或者 X1 引脚）	4		150	MHz
f_l 跛行模式 SYSCLKOUT 频率范围（/2 启用时）			200-500		kHz

表 6-6 XCLKIN 时序要求- 启用 PLL

编号		最小值	最大值	单位
C8	$t_{C(CI)}$ 周期时间，XCLKIN	33.3	200	ns
C9	$t_{f(CI)}$ 下降时间，XCLKIN (1)		6	ns
C10	$t_{r(CI)}$ 上升时间，XCLKIN (1)		6	ns
C11	$t_{w(CIL)}$ 脉冲持续时间，XCLKIN 低电平作为 $t_{c(CI)}$ (1) 的一部分的时间	45	55	%
C12	$t_{w(CIH)}$ 脉冲持续时间，XCLKIN 高电平作为 $t_{c(CI)}$ (1) 的一部分的时间	45	55	%

(1) 这也被应用到 X1 引脚。

表 6-7 XCLKIN 时序需求-- 禁用 PLL

编号			最小值	最大值	单位
C8	$t_{C(CI)}$周期时间，XCLKIN		6.67	250	ns
C9	$t_{f(CI)}$下降时间，XCLKIN	高达 30 MHz		6	ns
		30MHz 至 150MHz		2	
C10	$t_{r(CI)}$上升时间，XCLKIN	高达 30 MHz		6	ns
		30MHz 至 150MHz		2	
C11	$t_{w(CIL)}$脉冲持续时间，XCLKIN 低电平作为 $t_{c(CI)}$ (1) 的一部分的时间		45	55	%
C12	$t_{w(CIH)}$脉冲持续时间，XCLKIN 高电平作为 $t_{c(CI)}$ (1) 的一部分的时间		45	55	%

(1) 这也被应用到 X1 引脚。

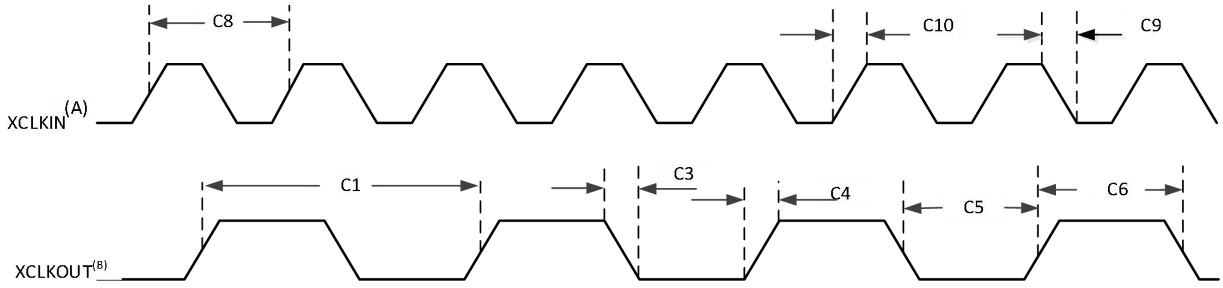
表 6-8 XCLKOUT 开关特性（旁路或禁用 PLL） (1) (2)

编号	参数	最小值	典型值	最大值	单位
C1	$t_{C(XCO)}$周期时间，XCLKOUT	6.67			ns
C3	$t_{f(XCO)}$下降时间，XCLKOUT		2	5	ns
C4	$t_{r(XCO)}$上升时间，XCLKOUT		2	5	ns
C5	$t_{w(XCOL)}$脉冲持续时间，XCLKOUT 低电平的时间	H-2		H+2	ns
C6	$t_{w(XCOH)}$ 脉冲持续时间，XCLKOUT 高电平的时间	H-2		H+2	ns
	t_p PLL 锁定时间	131072$t_{C\,(OSCCLK)}$ (3)			周期

(1) 假定这些参数有一个40pF 的负载。

(2) H=0.5$t_{c(XCO)}$。

(3) OSCCLK 或者为片载振荡器的输出，或者是来自一个外部振荡器的输出。



A. XCLKIN 与 XCLKOUT 的关系取决于所选择的分频因子。图6-5中显示的波形关系只用于解释时序参数并且根据实际配置会有所不同。
B. XCLKOUT 可配置为SYSCLKOUT的比例输出。

图 6-5 时钟时序

6.8 电源时序

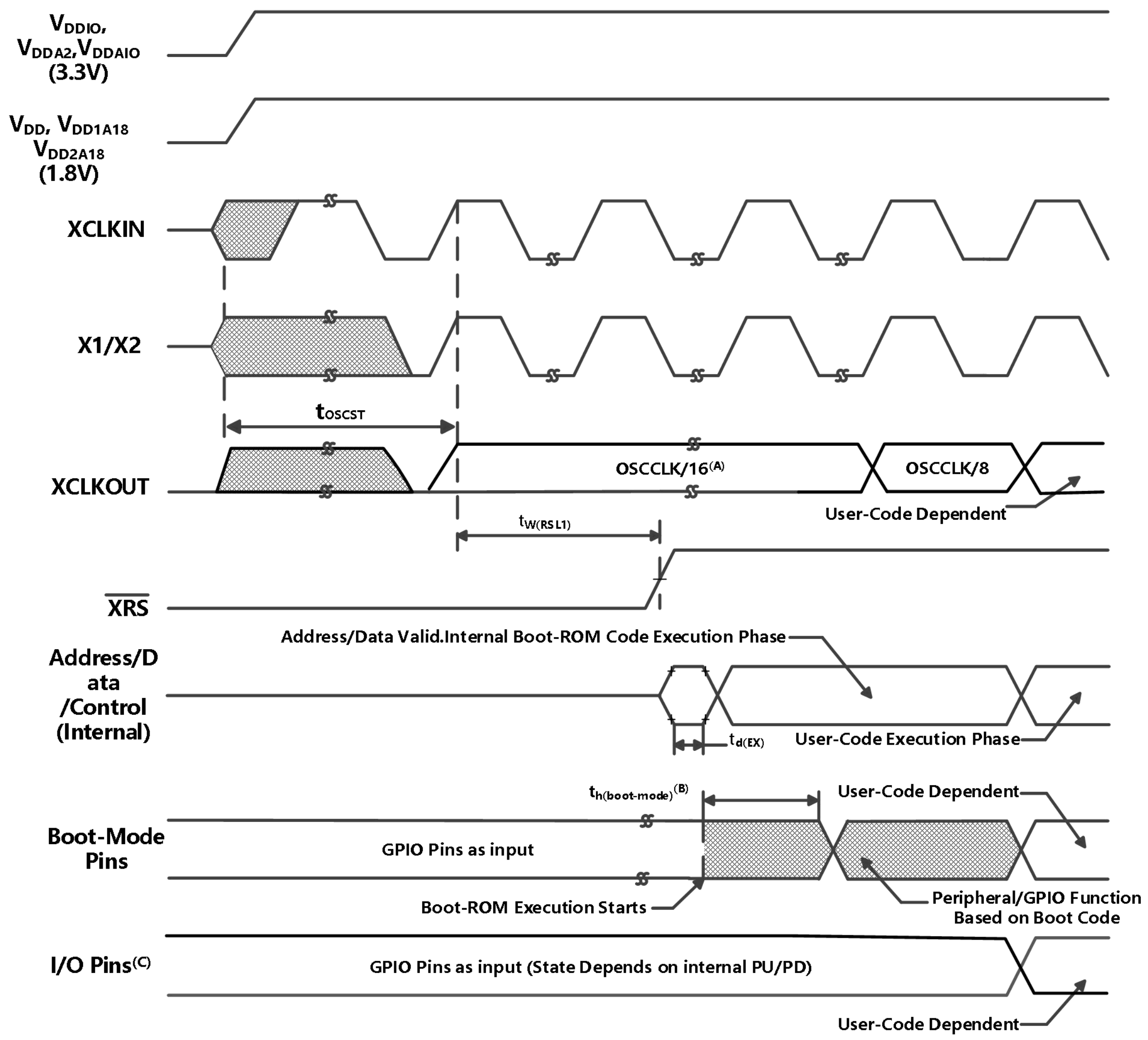
对于不同电源引脚的上电/断电序列无特别要求，须确保所有模块正确复位。

对于$\overline{XRS}$引脚的要求：

a) 上电期间，$\overline{XRS}$引脚必须在输入时钟稳定之后的 $t_{w(RSL1)}$内保持低电平（请见表 6-11）。这使得整个器件从一个已知的条件启动。

b) 断电期间，$\overline{XRS}$引脚必须至少在 V_{DD} 达到 1.5V 之前的 8μs 内被下拉至低电平。这样做提高了闪存可靠性。

在器件上电之前，不应将 V_{DDIO} 之上大于二极管压降 (0.7V) 的电压应用于任何数字引脚上（对于模拟引脚，这个值是比 V_{DDA} 高 0.7V 的电压值）。此外，V_{DDIO} 和 V_{DDA} 之间的差距应一直在 0.3V 之内。施加在未上电器件引脚上的电压会以非法方式正偏内部 PN 节并产生无法预料的结果。



A. 上电时，SYSCLKOUT为OSCCLK/4。由于XINTCNF2寄存器内的XTIMCLK和CLKMODE位出现时的状态为复位状态1，SYSCLKOUT在出现在XCLKOUT上之前会进一步进行4分频。这就是在这个阶段XCLKOUT=OSCCLK/16的原因。随后，引导ROM会将SYSCLKOUT改为OSCCLK/2。因为XTIMCLK存器不能被引导ROM改变，所以在此阶段，XCLKOUT为OSCCLK/8。

B. 复位后，引导ROM代码采样引导模式引脚。基于引导模式引脚的状态，引导代码向目的内存或者引导代码函数下达分支指令。如果引导ROM代码在上电条件后（在调试器环境中）执行代码，引导代码执行时间由当前的SYSCLKOUT的速度而定。SYSCLKOUT将基于用户环境并可在PLL启用或者不启用时使用。

C. 对于上电期间，确保GPIO引脚为高阻抗状态的要求，请见第6.8节。

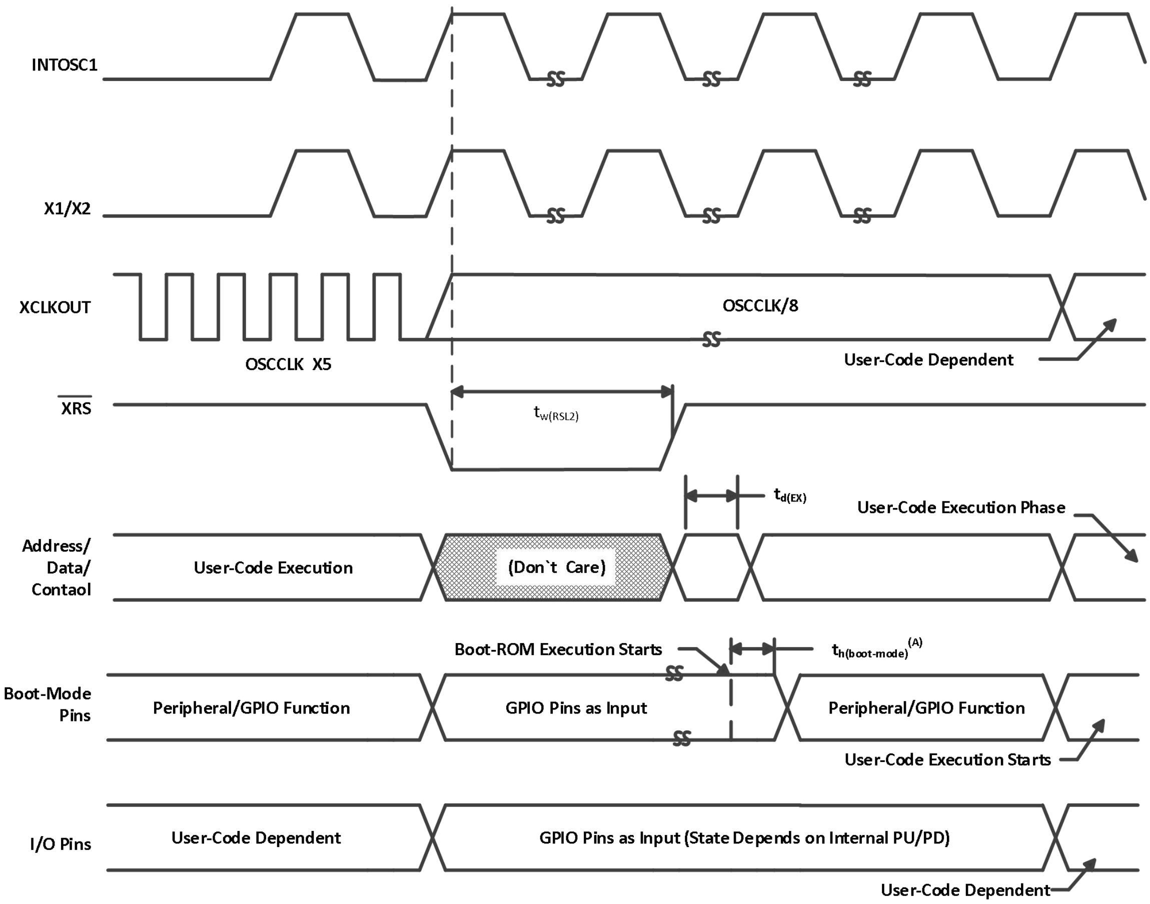
图 6-6 上电复位

表 6-9 复位$\overline{XRS}$时序要求

			最小值	典型值	最大值	单位
$t_{w(RSL1)}$ (1)	脉冲持续时间，稳定输入时钟$\overline{XRS}$高电平的时间		$32t_{c(OSCCLK)}$			周期
$t_{w(RSL2)}$	脉冲持续时间，$\overline{XRS}$低电平的时间	热复位	$32t_{c(OSCCLK)}$			周期
$t_{W(WDRS)}$	脉冲持续时间，由看门狗生成复位脉冲的时间			$512t_{c(OSCCLK)}$		周期
$t_{d(EX)}$	延迟时间，$\overline{XRS}$高电平后，地址/数据有效时间			$32t_{c(OSCCLK)}$		周期
T_{OSCST} (2)	振荡器启动时间			10		ms
t_h（引导模式）	引导模式引脚的保持时间		$200t_{c(SCO)}$			周期

(1) 另外，$t_{W(RSL1)}$要求, $\overline{XRS}$必须在 V_{DD} 达到 1.5V 后的 10ms 内为低电平。

(2) 取决于晶振/谐振器和电路板设计。



A. 复位后，引导ROM代码采样BOOT模式 引脚。基于引导模式引脚的状态，引导代码向目的内存或者引导代码函数下达分支指令。如果引导 ROM 代码在上电条件后（在调试器环境中）执行代码，引导代码执行时间由当前的SYSCLKOUT 的速度而定。SYSCLKOUT 将基于用户环境并可在PLL启用或者不启用时使用。

图 6-7 热复位

图6-8显示了写入PLLCR寄存器所产生的效果的一个示例。在第一个阶段，PLLCR=0x0004 并且 SYSCLKOUT=OSCCLK x 2。然后将 0x0008 写入 PLLCR。在PLLCR寄存器被写入后，PLL 锁存阶段开始。在这个阶段期间，SYSCLKOUT=OSCCLK/2。在PLL锁存完成后，SYSCLKOUT表示新的运行频率，即OSCCLK x 4。

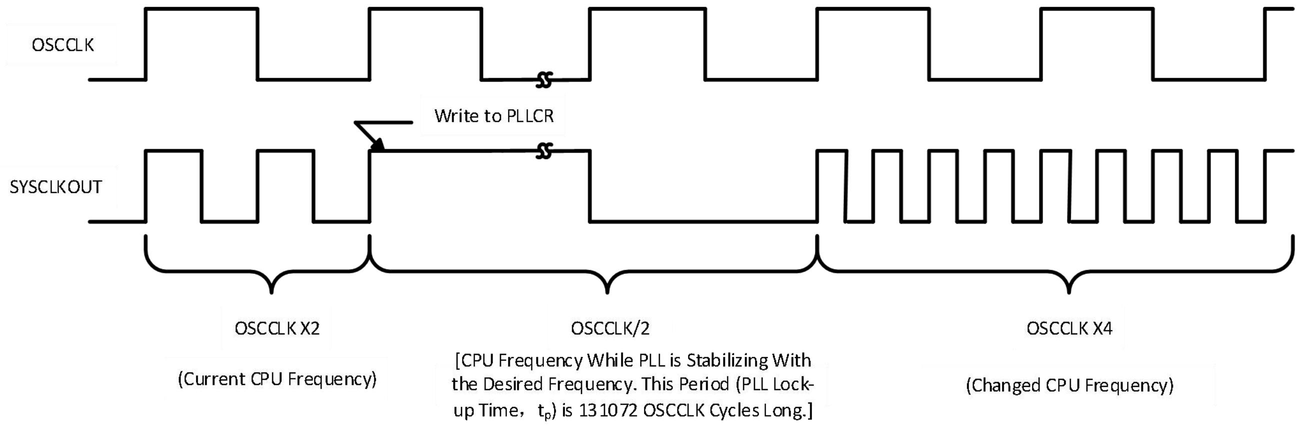


图 6-8 写入 PLLCR 寄存器所产生的效果示例

6.9 通用输入/输出(GPIO)

6.9.1 GPIO - 输出时序

表 6-10 通用输出开关特性

参数			最小值	最大值	单位
$t_{r(GPO)}$	上升时间，GPIO 从低电平切换至高电平的时间	所有 GPIO		8	ns
$t_{f(GPO)}$	下降时间，GPIO 从高电平切换至低电平的时间	所有 GPIO		8	ns
t_{fGPO}	切换频率，GPO 引脚			25	MHz

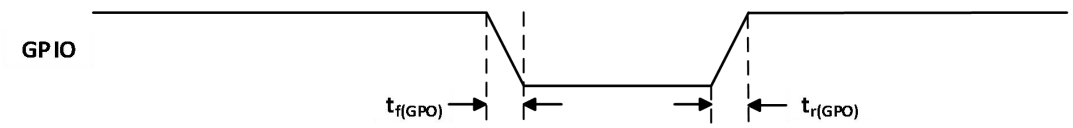


图 6-9 通用输出时序

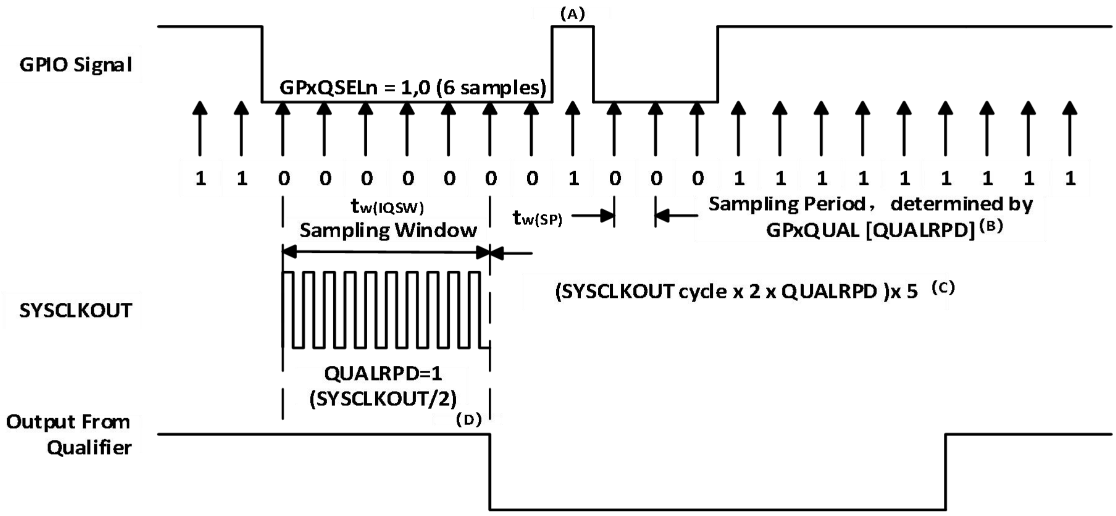
6.9.2 GPIO - 输入时序

表 6-11 通用输入时序要求

			最小值	最大值	单位
$t_{w(SP)}$	采样周期	QUALPRD=0	$1t_{c(SCO)}$		周期
		QUALPRD≠0	$2t_{c(SCO)}$*QUALPRD		周期
$t_{w(IQSW)}$	输入限定器采样窗口		$t_{w(SP)}$*(n[1]-1)		周期
$t_{w(GPI)}$ [2]	脉冲持续时间，GPIO 低电平/高电平的时间	同步模式	$2t_{c(SCO)}$		周期
		带有输入限定器	$t_{w(IQSW)} + t_{w(SP)} + 1t_{c(SCO)}$		周期

（1）"n" 代表由 GPxQSELn 寄存器定义的限定采样的数量。

（2）对于 $t_{w(GPI)}$，对于一个低电平有效信号，脉宽在 V_{IL} 至 V_{IL} 之间进行测量，而对于一个高电平有效信号脉宽在 V_{IH} 至 V_{IH} 之间进行测量。



A. 这个毛刺脉冲将被输入限定器所忽略。QUALPRD位字段指定了限定采样周期。它可在0x00至0xFF间变化。如果QUALPRD=00，那么采样周期为1个SYSCLKOUT周期。对于任何其它的"n"值，限定采样周期为2nSYSCLKOUT周期（也就是说，在每一个SYSCLKOUT周期上，GPIO引脚将被采样）。

B. 通过GPxCTRL寄存器选择的限定周期应用于一组8个GPIO引脚上。

C. 此限定块可采样3个或者6个样本。GPxQSELn寄存器选择使用的采样模式。

D. 在所示的示例中，为了使限定器检测到变化，输入应该在10个SYSCLKOUT周期或者更长的时间内保持稳定。换句话说，输入应该在(5xQUALPRDx2)SYSCLKOUT周期内保持稳定。这将确保发生5个用于检测的采样周期。由于外部时钟被异步驱动，一个宽度为13 SYSCLKOUT的脉冲将确保可靠识别。

图 6-10 采样模式

6.9.3 针对输入信号的采样窗口宽度

下文总结了不同的输入限定器配置下用于输入信号的采样窗口宽度。

采样频率表明相对于 SYSCLKOUT 的信号采样频率。

如果 QUALPRD≠0 的话，采样频率 = SYSCLKOUT/(2 x QUALPRD)

如果 QUALPRD=0 的话，采样频率 = SYSCLKOUT

如果 QUALPRD≠0 的话，采样周期 = SYSCLKOUT 周期 x 2 x QUALPRD

在上面的等式中，SYSCLKOUT 周期表明 SYSCLKOUT 的时间周期。

如果 QUALPRD=0 的话，采样周期 = SYSCLKOUT 周期。

在一个指定的采样窗口中，输入信号的 3 个样本或者 6 个样本被采样以确定信号的有效性。由写入到GPxQSELn 寄存器的值确定。

情况 1：

使用 3 个采样的限定

如果 QUALPRD≠0，采样窗口宽度 = （SYSCLKOUT 周期 x 2 x QUALPRD） x 2

如果 QUALPRD=0，采样窗口宽度 = （SYSCLKOUT 周期） x 2

情况2：

使用 6 个采样的限定

如果 QUALPRD≠0，采样窗口宽度 = （SYSCLKOUT 周期 x 2 x QUALPRD） x 5

如果 QUALPRD=0，采样窗口宽度 = （SYSCLKOUT 周期） x 5

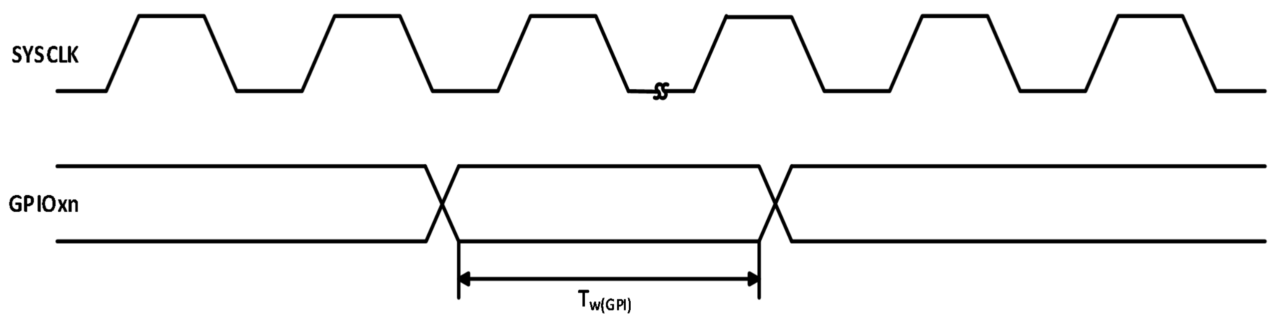


图 6-11 通用输入时序

6.9.4 低功耗唤醒时序

表 6-12 为时序要求，表 6-13 为开关特性，而图 6-12 为 IDLE 模式下的时序图。

表 6-12 IDLE 模式时序要求(1)

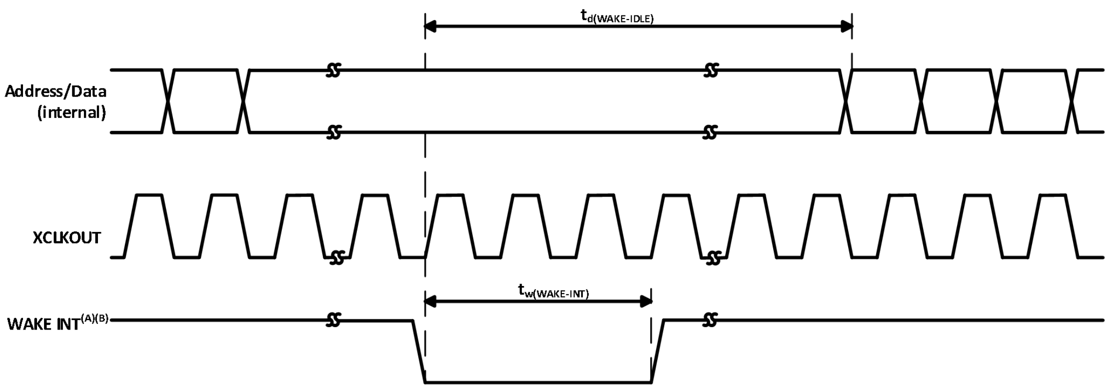
		最小值	典型值	最大值	单位
$t_{w(WAKE-INT)}$ **脉冲持续时间，外部唤醒信号的时间**	无输入限定器	$2t_{c(SCO)}$			周期
	带有输入限定器	$5t_{c(SCO)}+t_{w(IQSW)}$			

（1）对于输入限定器参数的说明，请见表 6-11。

表 6-13 IDLE 模式开关特性(1)

参数		测试条件	最小值	典型值	最大值	单位
$t_{d(WAKE-IDLE)}$	延迟时间，外部唤醒信号到程序执行重新开始的时间(2) 从闪存唤醒- 激活状态中的闪存模块	无输入限定器			$20t_{c(SCO)}$	周期
		带有输入限定器			$20t_{c(SCO)}+ t_{w(IQSW)}$	周期
	从闪存唤醒- 睡眠状态中的闪存模块	无输入限定器			$1050t_{c(SCO)}$	周期
		带有输入限定器			$1050t_{c(SCO)}+ t_{w(IQSW)}$	
	从 SARAM 中唤醒	无输入限定器			$20t_{c(SCO)}$	周期
		带有输入限定器			$20t_{c(SCO)}+ t_{w(IQSW)}$	

（1）对于输入限定器器参数的说明，请见表 6-11。
（2）这个时间是在 IDLE 指令之后立即开始指令执行的时间。一个 ISR（由唤醒触发）信号的执行会涉及额外的延迟。



A.WAKE INT 可以是任一被启用的中断，$\overline{\text{WDINT}}$或者$\overline{\text{XRS}}$。
B. 从将器件置于低功耗模式 (LPM) 的 IDLE 指令被执行开始，在至少 4 个 OSCCLK 周期之前，唤醒不应被启动。

图 6-12 IDLE 进入和退出时序

表 6-14 STANDBY 模式时序要求

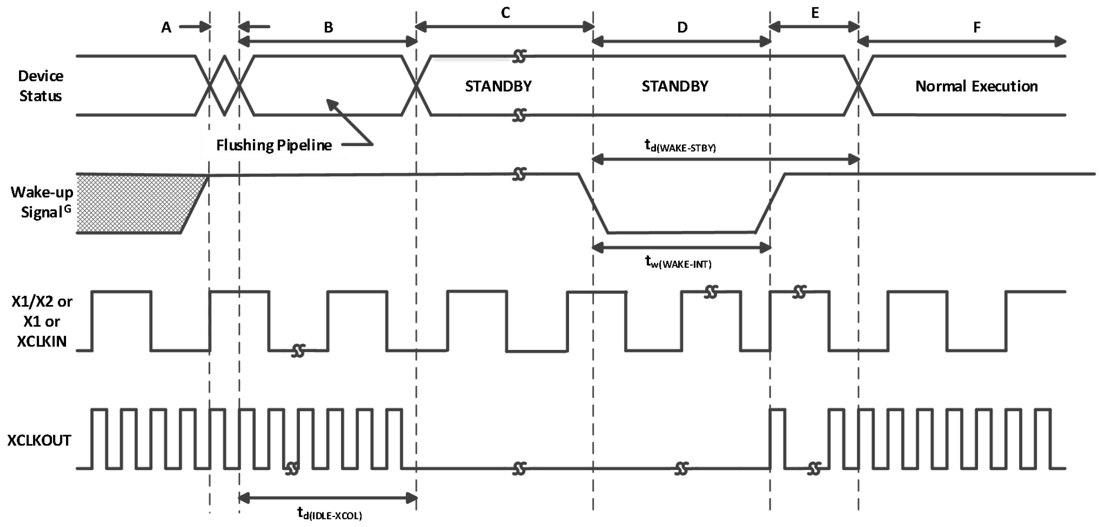
			最小值	典型值	最大值	单位
$t_{w(WAKE-INT)}$	**脉冲持续时间，外部唤醒信号的时间**	无输入限定器	$3t_{c(OSCCLK)}$			周期
		带有输入限定器[1]	(2 + QUALSTDBY) * $t_{c(OSCCLK)}$			

(1) 在 LPMCR0 寄存器中，QUALSTDBY 是一个 6 位宽字段。

表 6-15 STANDBY 模式开关特性

参数		测试条件	最小值	典型值	最大值	单位
$t_{d(IDLE-XCOL)}$	延迟时间，IDLE 指令执行至 XCLKOUT 为低电平的时间		$32t_{c(SCO)}$		$45t_{c(SCO)}$	周期
$t_{d(WAKE-STBY)}$	延迟时间，外部唤醒信号到程序执行重新开始的时间[1]					周期
	从闪存唤醒– 激活状态中的闪存模块	无输入限定器			$100t_{c(SCO)}$	周期
		带有输入限定器			$100t_{c(SCO)}+t_{w(WAKE-INT)}$	
	从闪存唤醒– 睡眠状态中的闪存模块	无输入限定器			$1125t_{c(SCO)}$	周期
		带有输入限定器			$1125t_{c(SCO)}+t_{w(WAKE-INT)}$	
	从 SARAM 中唤醒	无输入限定器			$100t_{c(SCO)}$	周期
		带有输入限定器			$100t_{c(SCO)}+t_{w(WAKE-INT)}$	

(1) 这个时间是在 IDLE 指令之后立即开始指令执行的时间。一个 ISR（由唤醒触发）信号的执行会涉及额外的延迟。



A. 被执行的IDLE 指令将器件置于STANDBY 模式。

B. PLL 块响应STANDBY 信号。在被关闭前，SYSCLKOUT 在下面标明的一定数量的周期内被保持：

当 DIVSEL=00 或 11 时，16 个周期

当 DIVSEL=10 时，32 个周期

当 DIVSEL=11 时，64 个周期

这个延迟使得CPU 管线和其它等待的操作被适当清空。到外设的时钟被关闭。

C. 到外设的时钟被关闭。PLL和看门狗并未关闭。此器件现在处于STANDBY 模式。

D. 外部唤醒信号被驱动为有效。

E. 在一个延迟周期内，退出STANDBY 模式。

F. 正常执行重新开始。此器件将响应中断（如果被启用的话）。

G. 从将器件置于低功耗模式 (LPM) 的 IDLE 指令被执行开始，在至少 4 个 OSCCLK 周期之前，唤醒不应被启动。

图 6-13 STANDBY 进入和退出时序

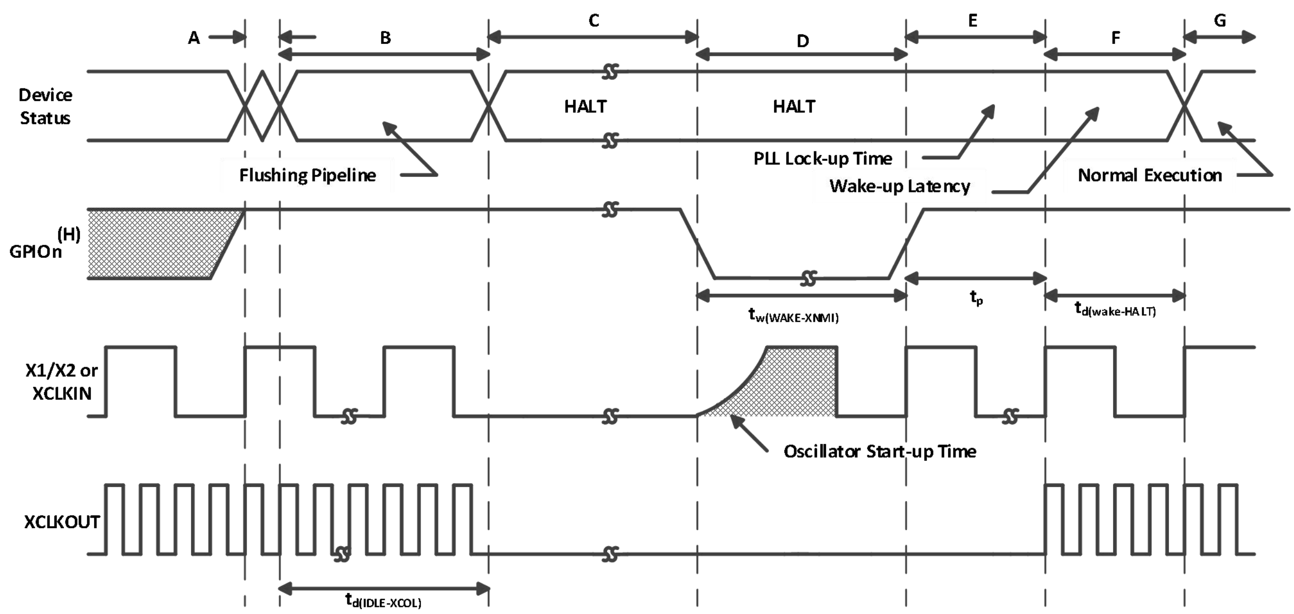
表 6-16 HALT 模式时序要求

	最小值	典型值	最大值	单位
$t_{w(WAKE-GPIO)}$ 脉冲持续时间，GPIO 唤醒信号的时间	$t_{oscst}+2t_{c(OSCCLK)}$(1)			周期
$t_{w(WAKE-XRS)}$ 脉冲持续时间，$\overline{XRS}$唤醒信号的时间	$t_{oscst}+8t_{c(OSCCLK)}$			周期

(1) t_{oscst}的解释请见表 6-9。

表 6-17 HALT 模式开关特性

	参数	最小值	典型值	最大值	单位
$t_{d(IDLE-XCOL)}$	延迟时间，IDLE 指令被执行至 XCLKOUT 为低电平的时间	$32t_{c(SCO)}$		$45t_{c(SCO)}$	周期
t_p	PLL 锁存时间			$131072t_{c(OSCCLK)}$	周期
$t_{d(WAKE-HALT)}$	延迟时间，PLL 锁存到程序执行重新开始的时间 从闪存唤醒 – 处于睡眠状态的闪存模块			$1125t_{c(SCO)}$	周期
	从 SARAM 中唤醒			$35t_{c(SCO)}$	周期



A. IDLE 指令被执行以将器件置于HALT 模式。

B. PLL 块响应 HALT 信号。 在振荡器被关闭并且到内核的 CLKIN 被停止前 SYSCLKOUT 在下面所示的一定数量的周期内保持：

当 DIVSEL=00 或 11 时，16个 周期

当 DIVSEL=10 时，32 个周期

当 DIVSEL=11 时，64 个周期

这个延迟使得 CPU 流水线和其它等待的操作被适当清空。

C. 到外设的时钟被关闭并且 PLL 被关断。如果一个石英晶振或者陶瓷谐振器被用作时钟源，内部振荡器也被关断。 器件现在处于 HALT 模式，消耗绝对最小功率。

D.当 GPIOn 引脚（用于使器件脱离 HALT 模式）被驱动为低电平时，振荡器被打开并且振荡器唤醒序列被启动。只有当振荡器稳定时，GPIO 才应被驱动为高电平。这样可在 PLL 锁序列期间提供一个干净的时钟信号。由于 GPIO 引脚的下降边沿异步开始唤醒序列，请注意在进入和处于 HALT 模式期间保持一个低噪声环境。

E. 一旦振荡器已经稳定，PLL 锁序列被启动（耗时 1ms），这将花费131072个OSCCLK（X1/X2或者X1或者XCLKIN）周期。请注意，即使当PLL被禁用（也就是说，即使当PLL被禁用时，代码执行也将被这个持续时间推迟），131072 个时钟周期也适用。

F. 当启用到内核的时钟和外设时，已经退出 HALT 模式。经过一段延迟后，器件响应此中断（如果被启用）。

G. 恢复正常工作。

H. 从将器件置于低功耗模式 (LPM) 的 IDLE 指令被执行开始，在至少 4 个 OSCCLK 周期之后，才可启动唤醒。

图 6-14 使用 GPIOn 的 HALT 唤醒

6.10 增强型控制外设

6.10.1 增强型脉宽调制器 (ePWM) 时序

PWM 是指ePWM1-6上的 PWM 输出。表 6-18为 PWM 时序要求，表 6-19为其开关特性。

表 6-18 PWM 时序要求

参数		测试条件	最小值	最大值	单位
$t_{w(SYCIN)}$	同步输入脉冲宽度	异步	$2t_{c(SCO)}$		周期
		同步	$2t_{c(SCO)}$		周期
		带有输入限定器(1)	$1t_{c(SCO)}+t_{w(IQSW)}$		周期

(1) 有关输入限定器参数的解释说明，请见表 6-11。

表 6-19 PWM 开关特性

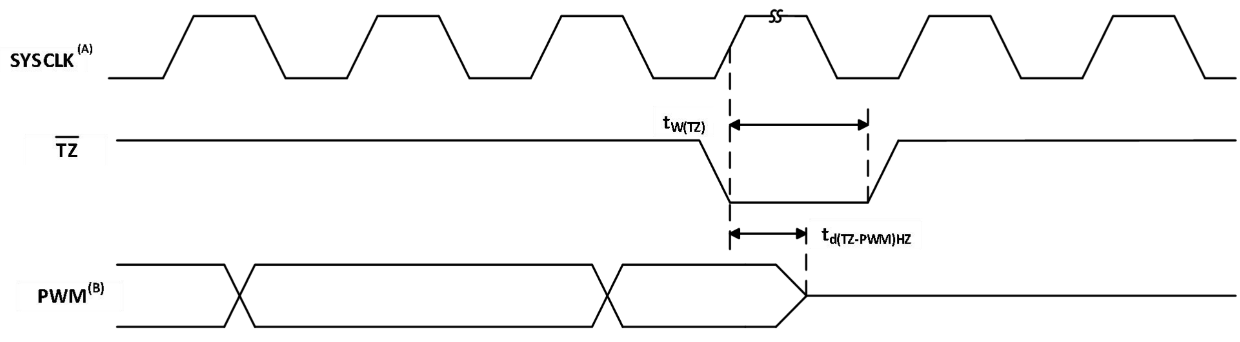
参数		测试条件	最小值	最大值	单位
$t_{w(PWM)}$	脉冲持续时间，PWMx 输出高电平/低电平的时间		33.33		ns
$t_{w(SYNCOUT)}$	同步输出脉冲宽度		$8t_{c(SCO)}$		周期
$t_{d(PWM)tza}$	延迟时间，触发输入有效到 PWM 强制高电平的时间； 延迟时间，触发输入有效到 PWM 强制低电平的时间；	无引脚负载		25	ns
$t_{d(TZ-PWM)HZ}$	延迟时间，触发输入有效至 PWM 高阻抗 (Hi-Z)的时间			20	ns

6.10.2 可编程控制故障区输入时序

表 6-20 可编程控制故障区输入时序要求

		最小值	最大值	单位
$t_{w(TZ)}$ 脉冲持续时间，$\overline{TZ}$x输入低电平的时间	异步	$2t_{c(TBCLK)}$		周期
	同步	$2t_{c(TBCLK)}$		周期
	带有输入限定器(1)	$2t_{c(TBCLK)}+t_{w(IQSW)}$		周期

(1) 有关输入限定器参数的解释说明，请见表 6-11。



A. TZ - TZ1，TZ2，TZ3，TZ4，TZ5，TZ6。

B. PWM 是指器件内的所有 PWM 引脚。TZ为高电平之后的 PWM 引脚的状态，取决于 PWM模块寄存器的配置。

图 6-15 PWM Hi-Z 特性

6.10.3 高分辨率 PWM (HRPWM) 时序

表6-21显示了高分辨率 PWM 的开关特性。

表 6-21 在 SYSCLKOUT=（60-150MHz）时，高分辨率 PWM 开关特性

	最小值	典型值	最大值	单位
微边沿定位 (MEP) 步长(1)		150	310	ps

(1)最大 MEP 步长基于最差情况、最高温度。MEP 步长将随温度的升高和电压的降低而增加，随着温度的下降和电压的升高而减小。使用 HRPWM 特性的应用应该使用 MEP 缩放因子优化器 (SFO) 近似软件函数。SFO 函数有助于在 HRPWM 运行时动态地估计每个 SYSCLKOUT 周期内的 MEP 步数。

6.10.4 增强型捕获器(eCAP)时序

表 6-22为eCAP时序要求，而表 6-23为eCAP开关特性。

表 6-22 增强型捕获器 (eCAP)时序要求

参数		测试条件	最小值	最大值	单位
$t_{w(CAP)}$	捕获器输入脉冲宽度	异步	$2t_{c(SCO)}$		周期
		同步	$2t_{c(SCO)}$		周期
		带有输入限定器(1)	$1t_{c(SCO)}+t_{w(IQSW)}$		周期

(1) 对于输入限定器参数的说明，请见表 6-11。

表 6-23 eCAP 开关特性

参数		测试条件	最小值	最大值	单位
$t_{w(APWM)}$	脉冲持续时间，APWMx 输出高电平/低电平的时间		20		ns

6.10.5 增强型正交编码脉冲 (eQEP) 时序

表 6-24为 eQEP 时序要求，而表 6-25为 eQEP 开关特性。

表 6-24 增强型正交编码脉冲 (eQEP) 时序要求(1)

参数		测试条件	最小值	最大值	单位
$t_{w(QEPP)}$	QEP 输入周期	异步/同步	$2t_{c(SCO)}$		周期
		带有输入限定器	$2[1t_{c(SCO)}+t_{w(IQSW)}]$		周期
$t_{w(INDEXH)}$	QEP 索引输入高电平时间	异步/同步	$2t_{c(SCO)}$		周期
		带有输入限定器	$2t_{c(SCO)}+t_{w(IQSW)}$		周期
$t_{w(INDEXL)}$	QEP 索引输入低电平时间	异步/同步	$2t_{c(SCO)}$		周期
		带有输入限定器	$2t_{c(SCO)}+t_{w(IQSW)}$		周期
$t_{w(STROBH)}$	QEP 选通脉冲高电平时间	异步/同步	$2t_{c(SCO)}$		周期
		带有输入限定器	$2t_{c(SCO)}+t_{w(IQSW)}$		周期
$t_{w(STROBL)}$	QEP 选通脉冲输入低电平时间	异步/同步	$2t_{c(SCO)}$		周期
		带有输入限定器	$2t_{c(SCO)}+t_{w(IQSW)}$		周期

(1) 对于输入限定器参数的说明，请见表 6-11。

表 6-25 eQEP 开关特性

参数	测试条件	最小值	最大值	单位
$t_{d(CNTR)xin}$ 延迟时间，外部时钟到计数器增量的时间			$4t_{c(SCO)}$	周期
$t_{d(PCS-OUT)QEP}$ 延迟时间，QEP 输入边沿到位置比较同步输出的时间			$6t_{c(SCO)}$	周期

6.10.6 ADC 转换开始时序

表 6-26 外部 ADC 转换开始开关特性

参数	最小值	最大值	单位
$t_{w(ADCSOCL)}$ 脉冲持续时间，$\overline{\mathrm{ADCSOCxO}}$ 低电平的时间	$32t_{c(HCO)}$		周期

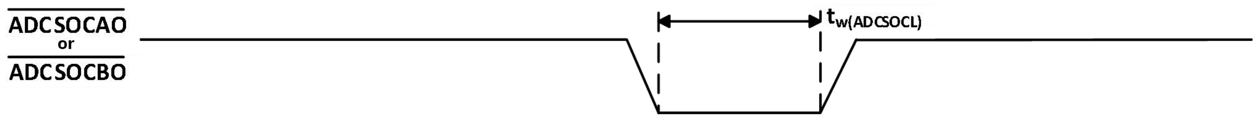


图 6-16 $\overline{\mathrm{ADCSOCAO}}$和$\overline{\mathrm{ADCSOCBO}}$时序图

6.11 外部中断时序

表 6-27 外部中断时序要求

参数	测试条件	最小值	最大值	单位
$t_{w(INT)}$(2) 脉冲持续时间，INT 输入低电平/高电平的时间	同步	$1t_{c(SCO)}$		周期
	带有输入限定器(1)	$1t_{c(SCO)}+t_{w(IQSW)}$		周期

(1)对于输入限定器参数的说明，请见表 6-11。

(2)这个时序适用于为 ADCSOC 功能性所配置的任一 GPIO 引脚。

表 6-28 外部中断开关特性 (1)

参数	最小值	最大值	单位
$T_{d(INT)}$ 延迟时间，INT 低电平/高电平到中断矢量提取的时间		$t_{w(IQSW)} + 12t_{c(SCO)}$	周期

(1) 对于输入限定器参数的说明，请见表 6-11。

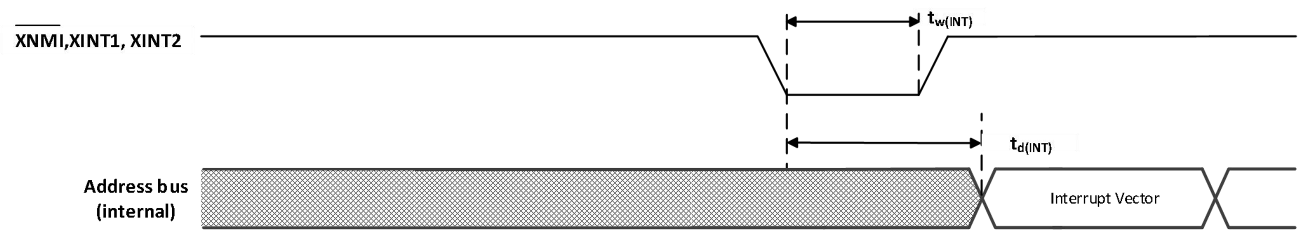


图 6-17 外部中断时序

6.12 IIC 电气特性和时序

表 6-29 IIC 时序

	测试条件	最小值	最大值	单位
f_{SCL} **SCL 时钟频率**	IIC 时钟模块频率介于 7MHz 和 12MHz 之间并且 IIC 预分频器和时钟分频器寄存器被适当配置		400	kHz
V_{il} **低电平输入电压**			$0.3V_{DDIO}$	V
V_{ih} **高电平输入电压**		$0.7V_{DDIO}$		V
V_{hys} **输入滞后**		$0.05V_{DDIO}$		V
V_{ol} **低电平输出电流**	3mA 灌电流	0	0.4	V
t_{LOW} **SCL 时钟的低周期**	IIC 时钟模块频率介于 7MHz 和 12MHz 之间并且 IIC 预分频器和时钟分频器寄存器被适当配置	1.3		µs
t_{HIGH} **SCL 时钟的高周期**	IIC 时钟模块频率介于 7MHz 和 12MHz 之间并且 IIC 预分频器和时钟分频器寄存器被适当配置	0.6		µs
I_I **输入电压介于 $0.1V_{DDIO}$ 和 $0.9V_{DDIO}$（最大值）的输入电流**		-10	10	µA

6.13 同步串行通信接口(SPI)时序

本节包含主模式和从模式时序数据。

6.13.1 主模式时序

表6-30所示为主模式时序（时钟相位= 0），表6-31所示为主模式时序（时钟相位=1）。图6-18和图6-19所示为时序波形。

表 6-30 SPI 主模式外部时序（时钟相位= 0） (1) (2) (3) (4) (5)

编号			当 (SPIBRR+1) 为偶数或者SPIBRR=0 或者2 时的SPI		当(SPIBRR+1) 为奇数并且SPIBRR>3 时的SPI		单位
			最小值	最大值	最小值	最大值	
1	$t_{c(SPC)M}$	周期时间，SPICLK	$4t_{c(LCO)}$	$128t_{c(LCO)}$	$5t_{c(LCO)}$	$127t_{c(LCO)}$	ns
2	$t_{w(SPCH)M}$	脉冲持续时间，SPICLK 高电平的时间（时钟极性= 0）	$0.5t_{c(SPC)M}-10$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}-0.5t_{c(LCO)}-10$	$0.5t_{c(SPC)M}-0.5t_{c(LCO)}$	ns
	$t_{w(SPCL)M}$	脉冲持续时间，SPICLK 低电平的时间（时钟极性= 1）	$0.5t_{c(SPC)M}-10$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}-0.5t_{c(LCO)}-10$	$0.5t_{c(SPC)M}-0.5t_{c(LCO)}$	
3	$t_{w(SPCL)M}$	脉冲持续时间，SPICLK 低电平的时间（时钟极性= 0）	$0.5t_{c(SPC)M}-10$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}+0.5t_{c(LCO)}-10$	$0.5t_{c(SPC)M}+0.5t_{c(LCO)}$	ns
	$t_{w(SPCH)M}$	脉冲持续时间，SPICLK 高电平的时间（时钟极性= 1）	$0.5t_{c(SPC)M}-10$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}+0.5t_{c(LCO)}-10$	$0.5t_{c(SPC)M}+ 0.5t_{c(LCO)}$	
4	$t_{d(SPCH-SIMO)M}$	延迟时间，SPICLK 高电平至 SPISIMO 有效的时间（时钟极性= 0）		10		10	ns
	$t_{d(SPCL-SIMO)M}$	延迟时间，SPICLK 低电平至 SPISIMO 有效的时间（时钟极性= 1）		10		10	
5	$t_{v(SPCL-SIMO)M}$	有效时间，SPICLK 低电平后，SPISIMO 数据有效的时间（时钟极性= 0）	$0.5t_{c(SPC)M}-10$		$0.5t_{c(SPC)M}+0.5t_{c(LCO)}-10$		ns
	$t_{v(SPCH-SIMO)M}$	有效时间，SPICLK 高电平之后，SPISIMO 数据有效的时间（时钟极性= 1）	$0.5t_{c(SPC)M}-10$		$0.5t_{c(SPC)M}+0.5t_{c(LCO)}-10$		
8	$t_{su(SOMI-SPCL)M}$	建立时间，SPISOMI 在 SPICLK 低电平之前的时间（时钟极性= 0）	35		35		ns
	$t_{su(SOMI-SPCH)M}$	建立时间，SPISOMI 在 SPICLK 高电平之前的时间（时钟极性= 1）	35		35		
9	$t_{v(SPCL-SOMI)M}$	有效时间，SPICLK 低电平之后 SPISOMI 数据有效的时间（时钟极性= 0）	$0.25t_{c(SPC)M}-10$		$0.5t_{c(SPC)M}-0.5t_{c(LCO)}-10$		ns
	$t_{v(SPCH-SOMI)M}$	有效时间，SPICLK 高电平之后 SPISOMI 数据有效的时间（时钟极性= 1）	$0.25t_{c(SPC)M}-10$		$0.5t_{c(SPC)M}-0.5t_{c(LCO)}-10$		

(1) 主模式/从模式(SPICTL.2) 被置位，而时钟相位寄存器(SPICTL.3) 被清除;

(2) $t_{c(SPC)}$ = SPI 时钟周期时间= LSPCLK/4 或者 LSPCLK/(SPIBRR + 1);

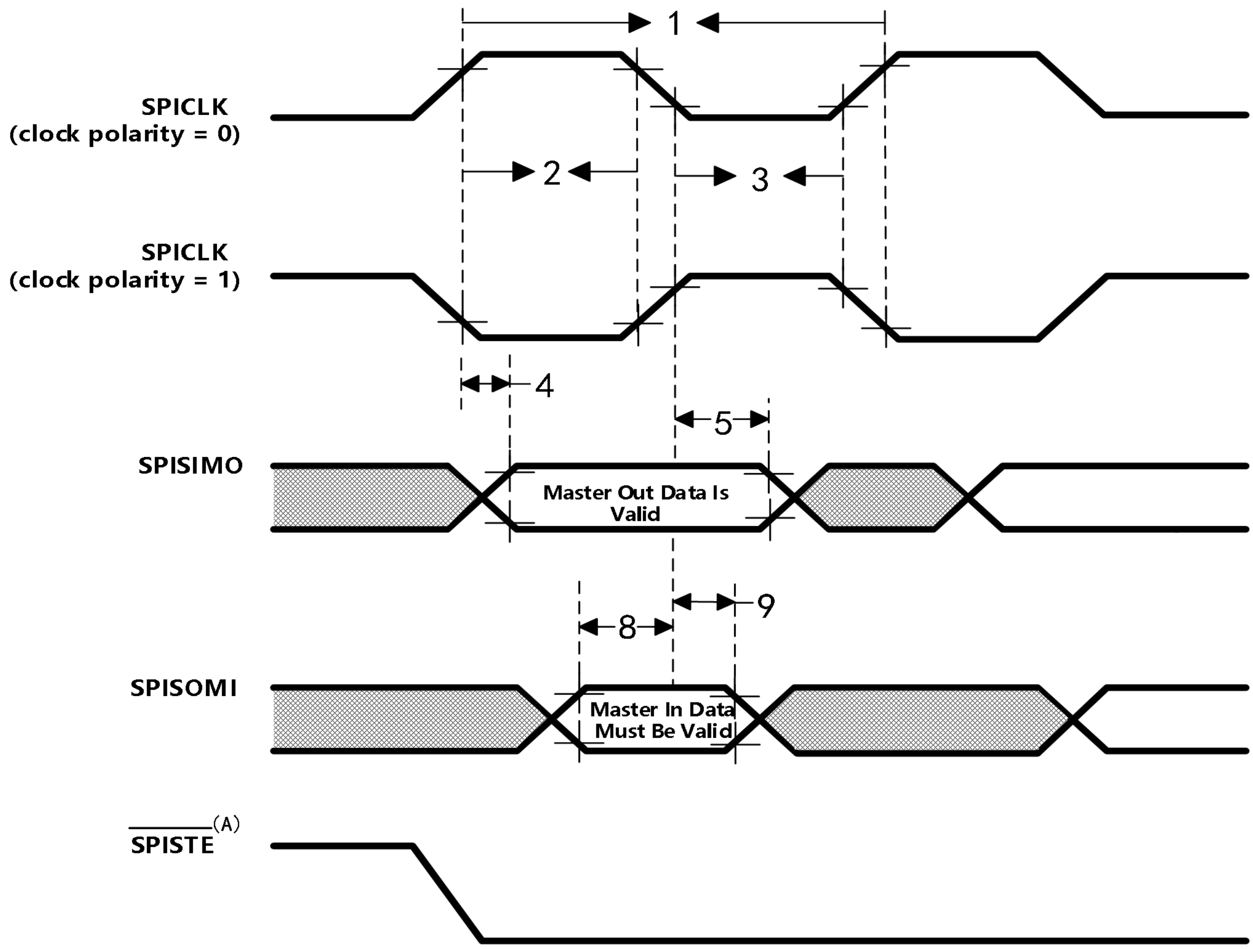
(3) $t_{c(LCO)}$ = LSPCLK 周期时间;

(4) 必须调节内部时钟预分频器，以将 SPI 时钟速度限制在以下 SPI 时钟速率：

主模式发送频率最大值 25MHz，主模式接收频率最大值 12.5MHz;

从模式发送频率最大值 12.5MHz，从模式接收频率最大值 12.5MHz;

(5) 作为基准的 SPICLK 信号的有效边沿由 CLOCK POLARITY（时钟极性）位(SPICCR 6) 控制。



A. 在主模式下，在有效的SPI 时钟边沿之前$0.5t_{c(SPC)}$（最小值），$\overline{SPISTE}$变为有效。在字的末端，$\overline{SPISTE}$在接收到最后一个数据位的边沿(SPICLK) 之后$0.5t_{c(SPC)}$将变为无效，除非$\overline{SPISTE}$在FIFO 和非FIFO 模式中的背靠背传送字间保持有效。

图 6-18 SPI 主模式外部时序（时钟相位= 0）

表 6-31 SPI 主模式外部时序（时钟相位= 1） (1) (2) (3) (4) (5)

编号			当(SPIBRR+1) 为偶数或者 SPIBRR=0 或者2 时的SPI		当(SPIBRR+1) 为奇数并且 SPIBRR>3 时的 SPI		单位
			最小值	最大值	最小值	最大值	
1	$t_{c(SPC)M}$	周期时间，SPICLK	$4t_{c(LCO)}$	$128t_{c(LCO)}$	$5t_{c(LCO)}$	$127t_{c(LCO)}$	ns
2	$t_{w(SPCH)M}$	脉冲持续时间，SPICLK 高电平的时间（时钟极性= 0）	$0.5t_{c(SPC)M}-10$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}-0.5t_{c(LCO)}-10$	$0.5t_{c(SPC)M}-0.5t_{c(LCO)}$	ns
	$t_{w(SPCL)M}$	脉冲持续时间，SPICLK 低电平的时间（时钟极性= 1）	$0.5t_{c(SPC)M}-10$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}-0.5t_{c(LCO)}-10$	$0.5t_{c(SPC)M}-0.5t_{c(LCO)}$	
3	$t_{w(SPCL)M}$	脉冲持续时间，SPICLK 低电平的时间（时钟极性= 0）	$0.5t_{c(SPC)M}-10$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}+0.5t_{c(LCO)}-10$	$0.5t_{c(SPC)M}+0.5t_{c(LCO)}$	ns
	$t_{w(SPCH)M}$	脉冲持续时间，SPICLK 高电平的时间（时钟极性= 1）	$0.5t_{c(SPC)M}-10$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}+0.5t_{c(LCO)}-10$	$0.5t_{c(SPC)M}+0.5t_{c(LCO)}$	

6	$t_{su(SIMO-SPCH)M}$	建立时间，在SPICLK 高电平之前 SPISIMO 数据有效的时间（时钟极性= 0)	$0.5t_{c(SPC)M}-10$		$0.5t_{c(SPC)M}-10$		ns
	$t_{su(SIMO-SPCL)M}$	建立时间，在 SPICLK 低电平之前 SPISIMO 数据有效的时间（时钟极性 =1)	$0.5t_{c(SPC)M}-10$		$0.5t_{c(SPC)M}-10$		
7	$t_{v(SPCH-SIMO)M}$	有效时间， SPICLK 高电平之后 SPISIMO 数据有效的时间（时钟极性= 0)	$0.5t_{c(SPC)M}-10$		$0.5t_{c(SPC)M}-10$		ns
	$t_{v(SPCL-SIMO)M}$	有效时间， SPICLK 低电平后， SPISIMO 数据有效的时间（时钟极性= 1)	$0.5t_{c(SPC)M}-10$		$0.5t_{c(SPC)M}-10$		
10	$t_{su(SOMI-SPCH)M}$	建立时间，SPISOMI 在 SPICLK 高电平之前的时间（时钟极性= 0)	35		35		ns
	$t_{su(SOMI-SPCL)M}$	建立时间，SPISOMI 在 SPICLK 低电平之前的时间（时钟极性= 1)	35		35		
11	$t_{v(SPCH-SOMI)M}$	有效时间， SPICLK 高电平之后 SPISOMI 数据有效的时间（时钟极性= 0)	$0.25t_{c(SPC)M}-10$		$0.5t_{c(SPC)M}-10$		ns
	$t_{v(SPCL-SOMI)M}$	有效时间， SPICLK 低电平之后 SPISOMI 数据有效的时间（时钟极性= 1)	$0.25t_{c(SPC)M}-10$		$0.5t_{c(SPC)M}-10$		

(1) 主模式/从模式(SPICTL.2) 被置位，而时钟相位寄存器(SPICTL.3) 被置位。

(2) $t_{c(SPC)}$ = SPI 时钟周期时间= LSPCLK/4 或者 LSPCLK/(SPIBRR + 1)。

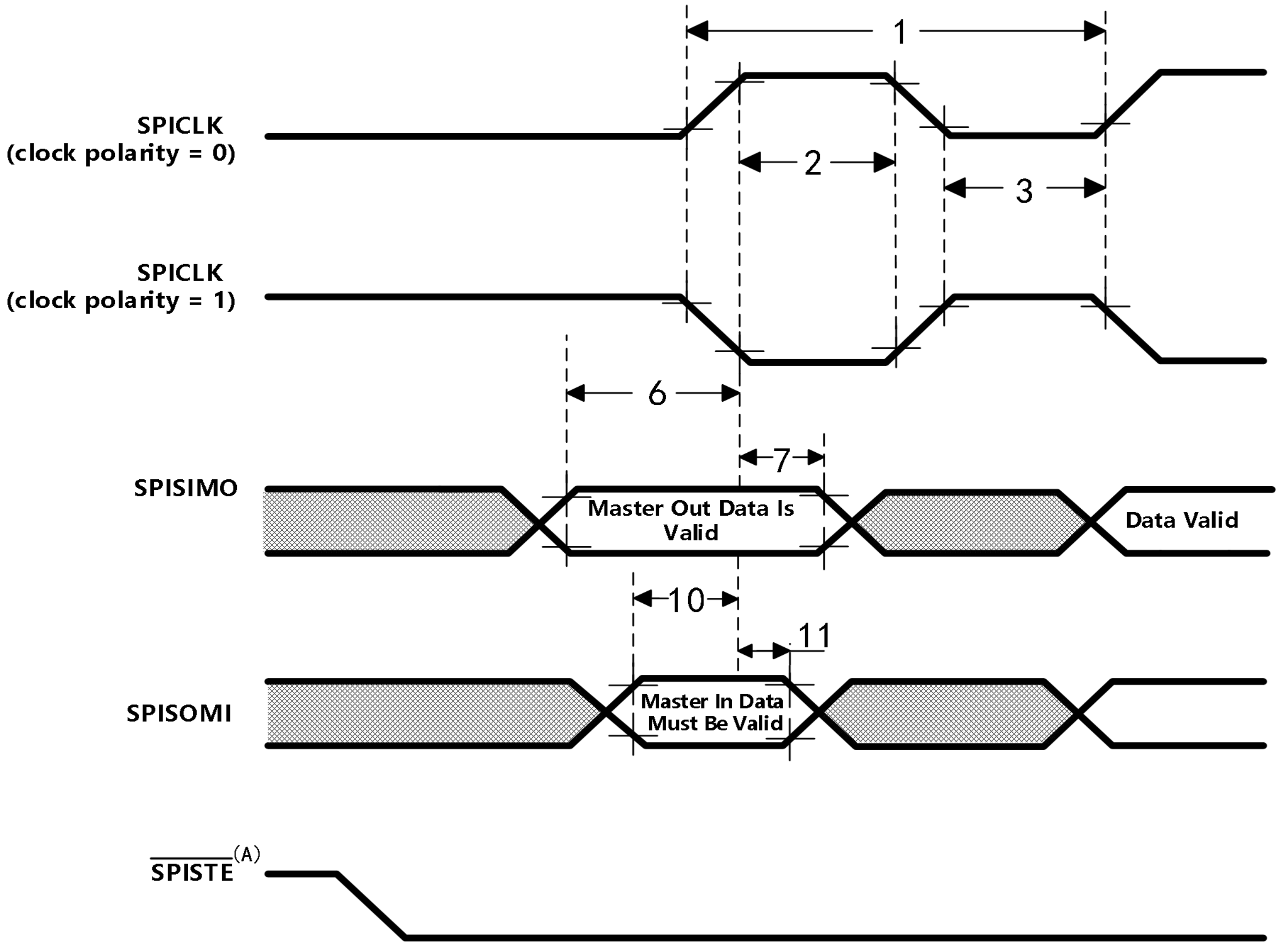
(3) 内部时钟预分频器必须被调整，这样的话，SPI 时钟速度被限制在下列 SPI 时钟速率上：

主模式发送最大值 25MHz，主模式接收最大值 12.5MHz；

从模式发送最大值 12.5MHz，从模式接收最大值 12.5MHz。

(4) $t_{c(LCO)}$ = LSPCLK 周期时间。

(5) 作为基准的 SPICLK 信号的有效边沿由 CLOCK POLARITY（时钟极性）位(SPICCR 6) 控制。



A.主模式下，在有效的SPI 时钟边沿之前0.5t$_{c(SPC)}$（最小值），$\overline{\text{SPISTE}}$变为有效。在字的末端，$\overline{\text{SPISTE}}$在接收到最后一个数据位的边沿(SPICLK) 之后0.5t$_{c(SPC)}$将变为无效，除非$\overline{\text{SPISTE}}$在FIFO 和非FIFO 模式中的背靠背传送字间保持有效。

图 6-19 SPI 主模式外部时序（时钟相位= 1）

6.13.2 SPI 从模式时序

表6-32列出了从模式外部时序（时钟相位= 0），而表6-33（时钟相位= 1）、图6-20和图6-21显示了时序波形。

表 6-32 SPI 从模式外部时序（时钟相位= 0）(1) (2) (3) (4) (5)

编号		最小值	最大值	单位
12	$t_{c(SPC)S}$ 周期时间，SPICLK	$4t_{c(LCO)}$		ns
13	$t_{w(SPCH)S}$ 脉冲持续时间，SPICLK 高电平的时间（时钟极性= 0)	$0.5t_{c(SPC)S}$-10	$0.5t_{c(SPC)S}$	ns
	$t_{w(SPCL)S}$ 脉冲持续时间，SPICLK 低电平的时间（时钟极性= 1)	$0.5t_{c(SPC)S}$-10	$0.5t_{c(SPC)S}$	
14	$t_{w(SPCL)S}$ 脉冲持续时间，SPICLK 低电平的时间（时钟极性= 0)	$0.5t_{c(SPC)S}$-10	$0.5t_{c(SPC)S}$	ns
	$t_{w(SPCH)S}$ 脉冲持续时间，SPICLK 高电平的时间（时钟极性= 1)	$0.5t_{c(SPC)S}$-10	$0.5t_{c(SPC)S}$	
15	$t_{d(SPCH-SOMI)S}$ 延迟时间，SPICLK 高电平至 SPISOMI 有效的时间（时钟极性= 0)		35	ns
	$t_{d(SPCL-SOMI)S}$ 延迟时间，SPICLK 低电平至 SPISOMI 有效的时间（时钟极性= 1)		35	
16	$t_{v(SPCL-SOMI)S}$ 有效时间，SPICLK 低电平之后 SPISOMI 数据有效的时间（时钟极性= 0)	$0.75t_{c(SPC)S}$		ns
	$t_{v(SPCH-SOMI)S}$ 有效时间，SPICLK 高电平之后 SPISOMI 数据有效的时间（时钟极性= 1)	$0.75t_{c(SPC)S}$		
19	$t_{su(SIMO-SPCL)S}$ 建立时间，SPISIMO 在 SPICLK 低电平之前的时间（时钟极性= 0)	35		ns
	$t_{su(SIMO-SPCH)S}$ 建立时间，SPISIMO 在 SPICLK 高电平之前的时间（时钟极性= 1)	35		
20	$t_{v(SPCL-SIMO)S}$ 有效时间，SPICLK 低电平后，SPISIMO 数据有效的时间（时钟极性= 0)	$0.5t_{c(SPC)S}$-10		ns
	$t_{v(SPCH-SIMO)S}$ 有效时间，SPICLK 高电平之后 SPISIMO 数据有效的时间（时钟极性= 1)	$0.5t_{c(SPC)S}$-10		

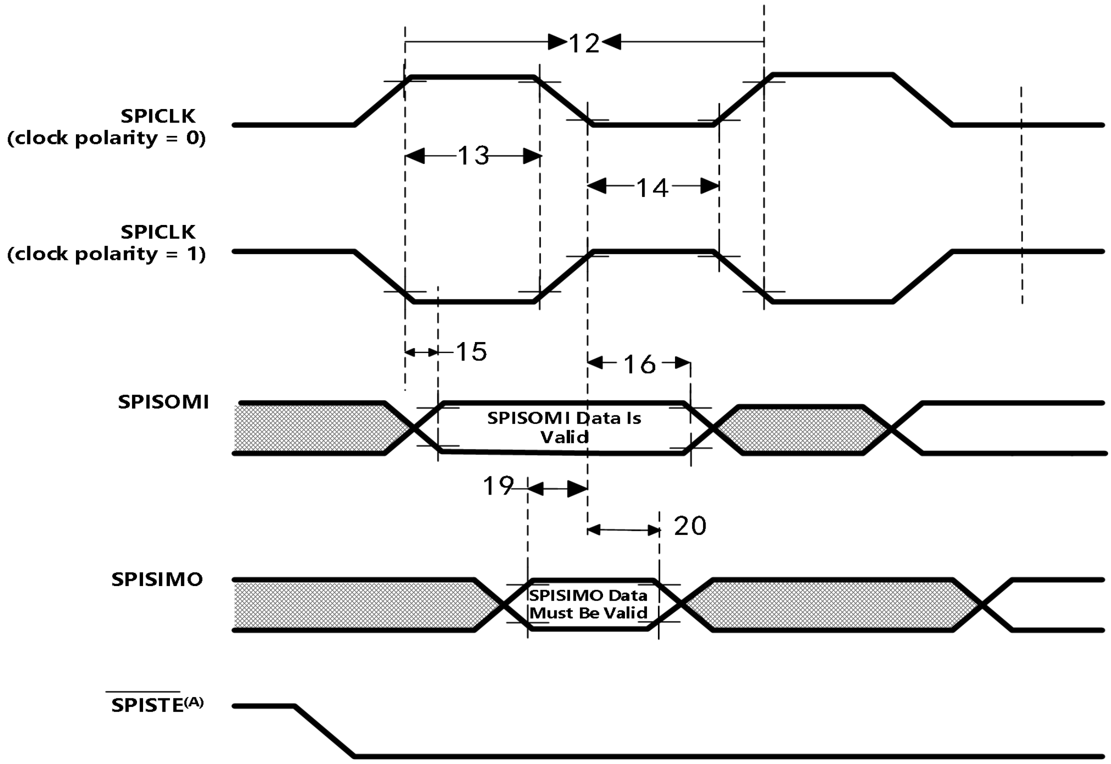
(1) 主模式/从模式(SPICTL.2) 被清除，而时钟相位寄存器(SPICTL.3) 被清除。

(2) $t_{c(SPC)}$ = SPI 时钟周期时间= LSPCLK/4 或者 LSPCLK/(SPIBRR + 1)。

(3) 内部时钟预分频器必须被调整，这样的话，SPI 时钟速度被限制在下列 SPI 时钟速率上：
主模式发送频率最大值 25MHz，主模式接收频率最大值 12.5MHz；
从模式发送频率最大值 12.5MHz，从模式接收频率最大值 12.5MHz。

(4) $t_{c(LCO)}$ = LSPCLK 周期时间。

(5) 作为基准的 SPICLK 信号的有效边沿由 CLOCK POLARITY（时钟极性）位(SPICCR. 6) 控制。



A. 在从模式下，$\overline{\text{SPISTE}}$信号至少应该在有效SPI 时钟边沿前0.5$t_{c(SPC)}$（最小值）被置为低电平有效并且在接收到最后一个数据位的 边沿(SPICLK) 之后保持至少0.5$t_{c(SPC)}$。

图 6-20 SPI 从模式外部时序（时钟相位= 0）

表 6-33 SPI 从模式外部时序（时钟相位= 1）(1) (2) (3) (4)

编号		最小值	最大值	单位
12	$t_{c(SPC)S}$周期时间，SPICLK	$8t_{c(LCO)}$		ns
13	$t_{w(SPCH)S}$脉冲持续时间，SPICLK 高电平的时间（时钟极性=0）	$0.5t_{c(SPC)S}$-10	$0.5t_{c(SPC)S}$	ns
	$t_{w(SPCL)S}$脉冲持续时间，SPICLK 低电平的时间（时钟极性=1）	$0.5t_{c(SPC)S}$-10	$0.5t_{c(SPC)S}$	
14	$t_{w(SPCL)S}$脉冲持续时间，SPICLK 低电平的时间（时钟极性=0）	$0.5t_{c(SPC)S}$-10	$0.5t_{c(SPC)S}$	ns
	$t_{w(SPCH)S}$脉冲持续时间，SPICLK 高电平的时间（时钟极性=1）	$0.5t_{c(SPC)S}$-10	$0.5t_{c(SPC)S}$	
17	$t_{su(SOMI-SPCH)S}$建立时间，SPISOMI 在 SPICLK 高电平之前的时间（时钟极性=0）	$0.125t_{c(SPC)S}$		ns
	$t_{su(SOMI-SPCL)S}$建立时间，SPISOMI 在 SPICLK 低电平之前的时间（时钟极性=1）	$0.125t_{c(SPC)S}$		
18	$t_{v(SPCL-SOMI)S}$有效时间，SPICLK 低电平后 SPISOMI 数据有效的时间（时钟极性=1）	$0.75t_{c(SPC)S}$		ns
	$t_{v(SPCH-SOMI)S}$有效时间，SPICLK 高电平后 SPISOMI 数据有效时间（时钟极性= 0）	$0.75t_{c(SPC)S}$		
21	$t_{su(SIMO-SPCH)S}$建立时间，SPISIMO 在 SPICLK 高电平之前的时间（时钟极性=0）	35		ns
	$t_{su(SIMO-SPCL)S}$建立时间，SPISIMO 在 SPICLK 低电平之前的时间（时钟极性=1）	35		
22	$t_{v(SPCH-SIMO)S}$有效时间，SPICLK 高电平后 SPISIMO 数据有效的时间（时钟极性=0）	$0.5t_{c(SPC)S}$-10		ns
	$t_{v(SPCL-SIMO)S}$有效时间，SPICLK 低电平后，SPISIMO 数据有效的时间（时钟极性=1）	$0.5t_{c(SPC)S}$-10		

(1) 主模式/从模式位(SPICTL.2) 被清除，而时钟相位寄存器(SPICTL.3) 被清除。

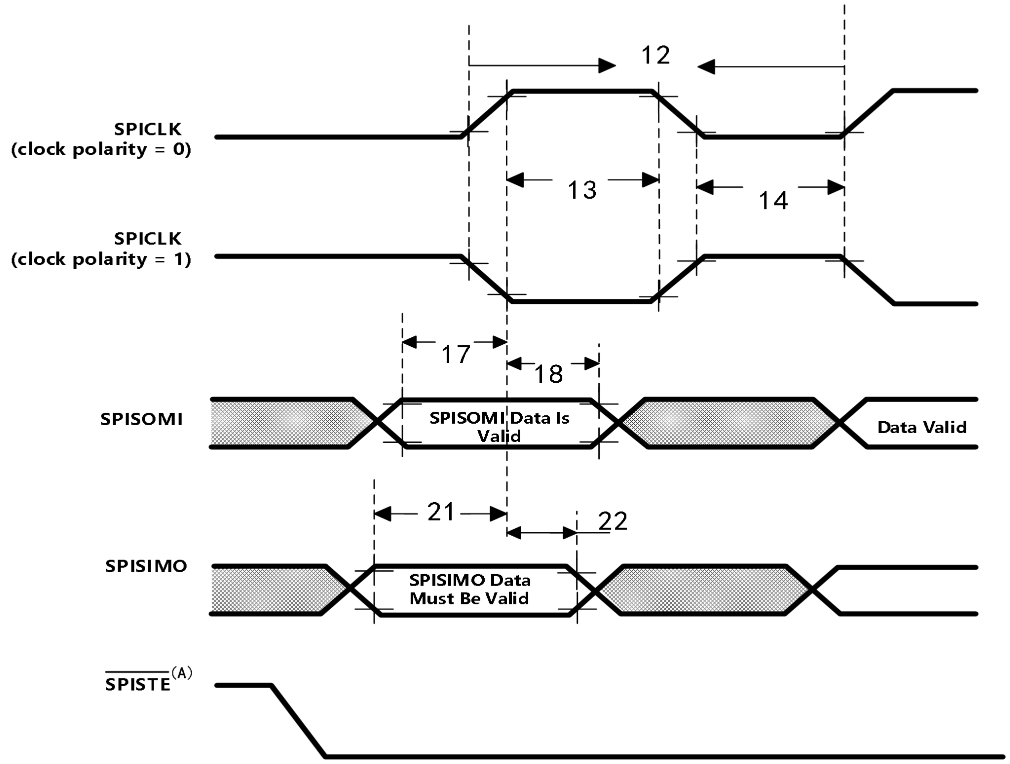
(2) $t_{c(SPC)}$ = SPI 时钟周期时间= LSPCLK/4 或者 LSPCLK/(SPIBRR + 1)。

(3) 内部时钟预分频器必须被调整，这样的话，SPI 时钟速度被限制在下列 SPI 时钟速率上：

主模式发送最大值 25MHz，主模式接收最大值 12.5MHz；

从模式发送最大值 12.5MHz，从模式接收最大值 12.5MHz。

(4) 作为基准的 SPICLK 信号的有效边沿由 CLOCK POLARITY（时钟极性）位(SPICCR. 6) 控制。



A.在从模式下，$\overline{\text{SPISTE}}$信号至少应该在有效SPI 时钟边沿前$0.5t_{c(SPC)}$被置为低电平有效并且在接收到最后一个数据位的边沿(SPICLK) 之后保持至少$0.5t_{c(SPC)}$。

图 6-21 SPI 从模式外部时序（时钟相位= 1）

6.14 片载模数转换器 ADC

表 6-34 ADC 电气特性（在推荐的运行条件）[1][2]

参数		最小值	典型值	最大值	单位
DC 技术规范[3]					
分辨率			12		位
ADC 时钟		0.001		12.5	MHz
精度					
INL（积分非线性）	1-12.5MHz ADC 时钟			±4	最低有效位 (LSB)
DNL（微分非线性）[4]				±2	LSB
偏移误差[5][3]		-15		+15	最低有效位 (LSB)
带有内部基准的总增益误差[6][3]		-60		+60	最低有效位 (LSB)
带有外部基准的总增益误差[3]		-60		+60	LSB
通道到通道偏移变化			±4		LSB
通道到通道增益变化			±4		LSB
模拟输入					
模拟输入电压（ADCINx 至 ADCLO）[7]		0		3	V
ADCLO		-5		5	mV
输入电容值			10		pF
输入漏电流		-5		5	μA
内部基准电压[6]					
$V_{ADCREFP}$-在基于内部基准的引脚上的 ADCREFP 输出电压			1.275		V
$V_{ADCREFM}$-在基于内部基准的引脚上的 ADCREFM 输出电压			0.525		V
ADCREFP-ADCREFM 的电压差值			0.75		V
温度系数			50		PPM/°C
外部基准电压[6][8]					
$V_{ADCREFIN}$-推荐在外部基准电压 ADCREFIN 引脚输入 0.2%或者更高精度的基准电压	ADCREFSEL[15:14]=11b		1.024		V
	ADCREFSEL[15:14]=10b		1.500		V
	ADCREFSEL[15:14]=01b		2.048		V
AC 技术规范					
SINAD (100kHz) 信噪比+失真			63.3		dB
SNR (100kHz) 信噪比			66.4		dB
THD (100kHz) 总谐波失真			-66.3		dB
ENOB (100kHz) 有效位数			10.2		位
SFDR (100kHz) 无杂散动态范围			68.5		dB

(1)在 12.5 MHz ADCCLK 上测得。

(2)这个表中的所有电压相对 V_{SSA2}。

(3)如果 ADC 的校准例程被从引导 ROM 执行，ADC 增益误差和偏移误差参数只为额定值。更多信息请查阅第 4.7.3 节。

(4)指定 ADC 将无丢码。

(5)1 个 LSB 等于 3.0/4096=0.732 mV 的加权值。

(6)一个单一内部/外部带隙基准为 ADCREFP 和 ADCREFM 信号供源，因此，这些电压可一起跟踪。 ADC 转换器使用这两个电压之间的 差值作为它的基准电压。这里列出的基于内部基准模式的总增益误差包括内部带隙在全温范围内的温漂引起的增益误差。基于外部基准模式的全温范围内的增益误差将取决于所使用基准源的温度系数。

(7)在模拟输入引脚上应用一个高于 V_{DDA} +0.3V 或者低于 V_{SS}-0.3V 的电压可能会暂时影响其它引脚的转换。为了避免这种情况，模拟输入应该被保持在这些限值内。

(8)建议使用高精度外部基准部件 REF3020/3120 或者针对 2.048V 基准的等效器件。

6.14.1 ADC 上电控制位时序

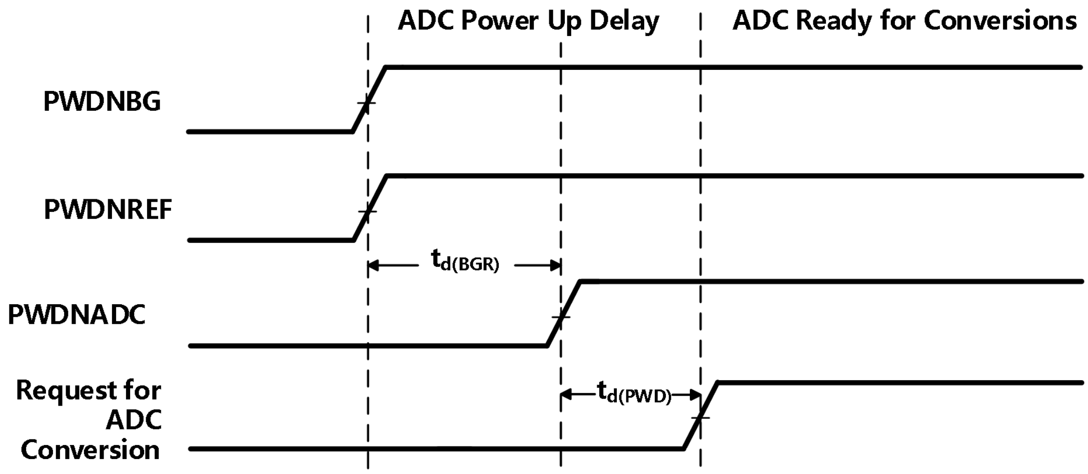


图 6-22 ADC 上电控制位时序

表 6-35 ADC 上电延迟

	参数(1)	条件	最小值	典型值	最大值	单位
$t_{d(BGR)}$	带隙基准稳定所需的延迟时间。ADCTRL3 寄存器的位 7 和 6(ADCBGRFDN1/0)在 PWDNADC 位被启用前被设定为 1。	ADCREFP/ADCREFM 外接 10μF 电容时			10	ms
$t_{d(PWD)}$	断电控制稳定所需的延迟时间。带隙基准稳定所需的位延迟时间。ADCTRL3 寄存器的位 7 和 6(ADCBGRFDN1/0)在 PWDNADC 位被启用前被设定为 1。ADCTRL3 寄存器的位 5(PWDNADC)在任何 ADC 转换启动前被设为 1。	ADCREFP/ADCREFM 外接 10μF 电容时	10	12	20	ms

(1) AVP32F08 ADC 支持同时驱动所有 3 位，并在首次转换前等待 $t_{d(BGR)}$ms。

表 6-36 不同 ADC 配置的典型电流消耗（在 12.5MHz ADCCLK 上）(1)(2)

ADC 运行模式	条件	V_{DDA18}	$V_{DDA3.3}$	单位
模式 A - 运行模式	• BG REF 启用 • PWD 禁用	50	9	mA
模式 B	• ADC 时钟启用 • BG 和 REF 启用 • PWD 启用	60	500	μA
模式 C	• ADC 时钟启用 • BG 和 REF 禁用 • PWD 启用	50	500	μA
模式 D	• ADC 时钟禁用 • BG 和 REF 禁用 • PWD 启用	40	500	μA

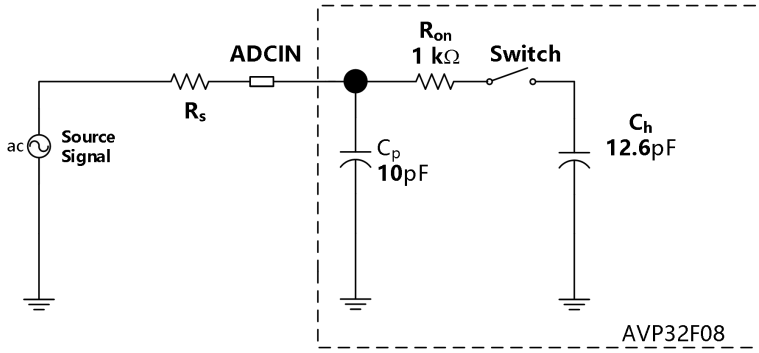
(1) 测试条件：

SYSCLKOUT= 150MHz

ADC 模块时钟 = 12.5MHz

ADC 在模式 A 中执行一个所有 16 通道的连续转换。

(2) V_{DDA18} 包括进入 V_{DD1A18} 和 V_{DD2A18} 的电流。$V_{DDA3.3}$ 包括进入 V_{DDA2}、V_{DDAIO} 和 V_{DDA3} 的电流。



Typical Values of the Input Circuit Components:

Switch Resistance (R_{on}): 1 kΩ

Sampling Capacitor (C_h): 12.6 pF

Parasitic Capacitance (C_p): 10 pF

Source Resistance (R_s): 50 Ω

图 6-23 ADC 输入阻抗模型

6.14.2 基本定义

基准电压：片载 ADC 有一个内置基准，这个基准为 ADC 提供了基准电压。

模拟输入：片载 ADC 由 16 个模拟输入组成，一次采样一个通道，或者一次采样两个通道。这些输入可选。

转换器：片载 ADC 使用一个 12 位四级流水线架构，此架构可在低功耗时实现高采样率。

转换时间：转换可以在两个不同的转换模式中执行：

a) 顺序采样模式 (SMODE = 0)

b) 同步采样模式 (SMODE= 1)

6.14.3 ADC 顺序采样模式（单通道）（SMODE=0）

在顺序采样模式下，ADC能够持续在任一通道（Ax至Bx）上转换输入信号。ADC能够在来自ePWM、软件触发器、外部ADCSOC信号的事件触发上启动转换。如果SMODE位为0，ADC将在每个采样/保持脉冲上的所选通道上进行转换。下面对转换时间和结果寄存器更新的延迟进行解释说明。ADC中断标志在结果寄存器更新之后的几个SYSCLKOUT周期内被置位。所选通道将在采样/保持脉冲的每个下降边沿上被采样。采样/保持脉冲宽度可被设定为1个ADC时钟宽度（最小值）或者16个ADC时钟宽度（最大值）。

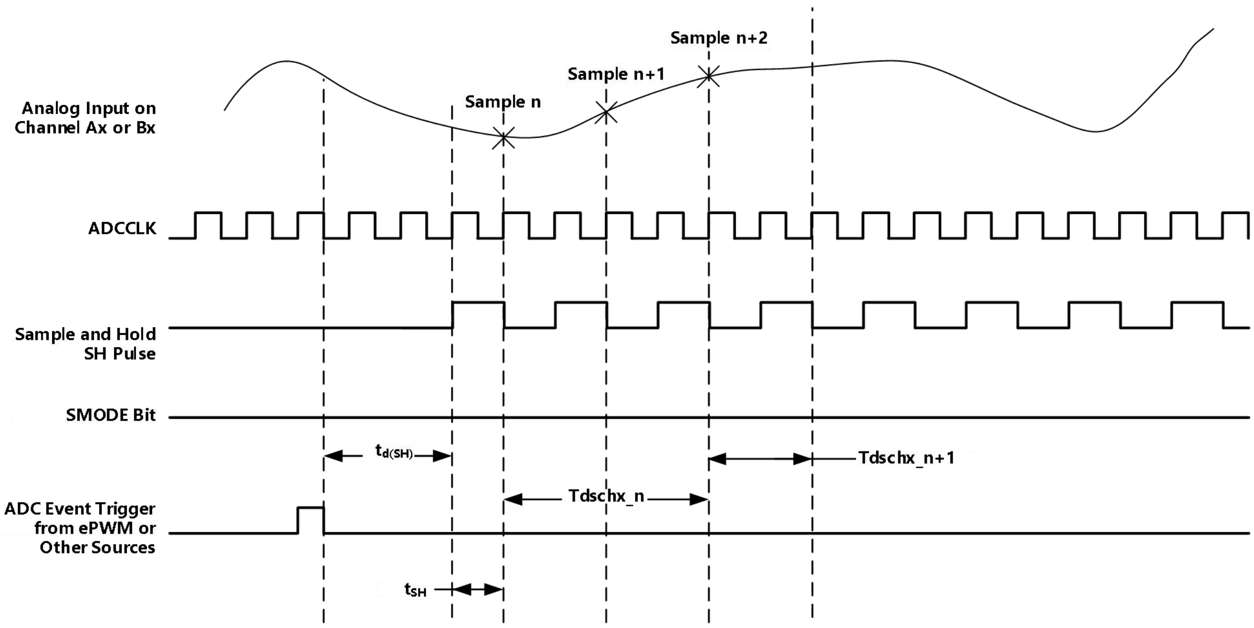


图 6-24 顺序采样模式（单通道）时序

表 6-37 顺序采样模式时序

参数		SAMPLE n	SAMPLE n+1	在 12.5MHz ADC 时钟上，$t_{c(ADCCLK)}$=80ns	注释
$t_{d(SH)}$	从事件触发器到采样的延迟时间	2.5$t_{c(ADCCLK)}$			
t_{SH}	采样/保持宽度或采样宽度	(1+Acqps)* $t_{c(ADCCLK)}$		Acqps=0 时为 80ns	Acqps 值=0-15 ADCTRL1[8:11]
$t_{d(schx\_n)}$	结果寄存器出现第一个结果的延迟时间	4$t_{c(ADCCLK)}$		320ns	
$t_{d(schx\_n+1)}$	结果寄存器中出现连续结果的延迟时间		(2+Acqps)*$t_{c(ADCCLK)}$	160ns	

6.14.4 ADC 同步采样模式（双通道）（SMODE=1）

在同步采样模式下，ADC能够在任何一对通道（A0/B0至A7/B7）上持续转换输入信号。ADC能够在来自ePWM，软件触发器，或者来自一个外部ADCSOC信号的事件触发上启动转换。如果SMODE位为1，ADC将在每个采样/保持脉冲上的两个所选通道上进行转换。下面对转换时间和结果寄存器更新的延迟进行解释说明。ADC中断标志在结果寄存器更新之后的几个SYSCLKOUT周期内被设定。所选通道将在采样/保持脉冲的下降边沿上被同时采样。采样/保持脉冲宽度可被设定为1个ADC时钟宽度（最小值）或者16个ADC时钟宽度（最大值）。

注

在同步模式中，ADCIN 通道对选择必须为 A0/B0，A1/B1，...，A7/B7，并且不能进行任何其它组合（例如 A1/B3，等等）。

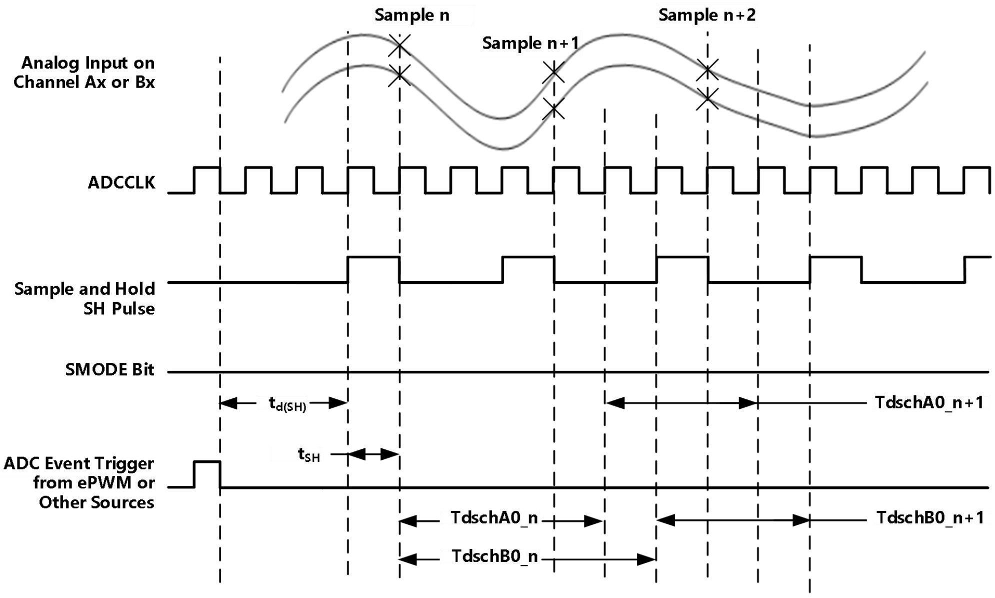


图 6-25 同步采样模式（双通道）时序

表 6-38 同步采样模式时序

参数		**SAMPLE n**	**SAMPLE n+1**	**在 12.5MHz ADC 时钟上，$t_{c(ADCCLK)}$=80ns**	**注释**
$t_{d(SH)}$	从事件触发器到采样的延迟时间	2.5$t_{c(ADCCLK)}$			
t_{SH}	采样/保持宽度或采样宽度	(1+Acqps)* $t_{c(ADCCLK)}$		Acqps=0 时为 80ns	Acqps 值=0-15 ADCTRL1[8:11]
$t_{d(schA0\_n)}$	结果寄存器出现第一个结果的延迟时间	4$t_{c(ADCCLK)}$		320ns	
$t_{d(schB0\_n)}$	结果寄存器出现第一个结果的延迟时间	5$t_{c(ADCCLK)}$		400ns	
$t_{d(schA0\_n+1)}$	结果寄存器中出现连续结果的延迟时间		(3+Acqps)*$t_{c(ADCCLK)}$	240ns	
$t_{d(schB0\_n+1)}$	结果寄存器中出现连续结果的延迟时间		(3+Acqps)*$t_{c(ADCCLK)}$	240ns	

6.14.5 详细说明

积分非线性

积分非线性是指每个单独的转换码与从零点到满量程点绘制直线的偏差。零点出现在第一个转换码之前1/2LSB的位置处。满量程点定义为低于最后转换码1/2 LSB的位置处。偏差是指测量每个特定转换码的中心到这两个点之间的真实直线的距离。

微分非线性

理想的ADC展示出转换码间的间距恰好相距1 LSB。DNL是与此理想值的偏差。小于 ±1 LSB 的差分非线性误差可确保无失码。

零偏移

当模拟输入为0V时，应当发生主进位转换。零误差被定义为实际转换与该点的偏差。

增益误差

第一个转换码应出现在比负满量程高1/2LSB的模拟值处。最后一个转换码应出现在比标称满量程低1/2 LSB 的模拟值处。增益误差是指实际第一个和最后一个转换码之间的差值与理想第一个和最后一个转换码之间的差值的偏差

信噪比+失真(SINAD)

SINAD是输入信号测量值的均方根值与其他所有低于奈奎斯特频率的频谱分量的均方根总和之比，包括谐波，但不包括直流。SINAD 的值以dB表示。

有效位数(ENOB)

对于正弦波，SINAD可以用位数来表示。使用以下公式，$N = \frac{(SINAD-1.76)}{6.02}$可以得到以N表示的性能度量，即有效位数。因此，在给定输入频率下，正弦波输入器件的有效位数可以直接从其测量的SINAD计算出来。

总谐波失真(THD)

THD是前九个谐波分量的均方根总和与输入信号测量值的均方根值的比值，以百分比或分贝表示。

无杂散动态范围(SFDR)

SFDR是输入信号的均方根幅度与峰值杂散信号的均方根幅度之间的差值，以dB为单位。

6.15 多通道缓冲串行端口(McBSP)模块

6.15.1 McBSP 发送和接收时序

表 6-39 McBSP 时序要求[(1)(2)]

编号	参数			最小值	最大值	单位
		McBSP 模块时钟 (CLKG，CLKX，CLKR) 范围		1		kHz
					25 [(3)]	MHz
		McBSP 模块周期时间 (CLKG，CLKX，CLKR) 范围		40		ns
					1	ms
M11	$t_{c(CKRX)}$	周期时间，CLKR/X	CLKR/X 外部	2P		ns
M12	$t_{w(CKRX)}$	脉冲持续时间，CLKR/X 高电平或者 CLKR/X 低电平的时间	CLKR/X 外部	P–7		ns
M13	$t_{r(CKRX)}$	上升时间，CLKR/X	CLKR/X 外部		7	ns
M14	$t_{f(CKRX)}$	下降时间，CLKR/X	CLKR/X 外部		7	ns
M15	$t_{su(FRH-CKRL)}$	建立时间，在 CLKR 低电平之前外部 FSR 为高电平的时间	CLKR 内部	18		ns
			CLKR 外部	2		
M16	$t_{h(CKRL-FRH)}$	保持时间，CLKR 低电平之后，外部 FSR 为高电平的时间	CLKR 内部	0		ns
			CLKR 外部	6		
M17	$t_{su(DRV-CKRL)}$	建立时间，在 CLKR 低电平之前，DR 有效的时间	CLKR 内部	18		ns
			CLKR 外部	2		
M18	$t_{h(CKRL-DRV)}$	保持时间，在 CLKR 低电平之后，DR 有效的时间	CLKR 内部	0		ns
			CLKR 外部	6		
M19	$t_{su(FXH-CKXL)}$	建立时间，在 CLKX 低电平之前，外部 FSX 为高电平的时间	CLKX 内部	18		ns
			CLKX 外部	2		
M20	$t_{h(CKXL-FXH)}$	保持时间，CLKX 低电平之后，外部 FSX 为高电平的时间	CLKX 内部	0		ns
			CLKX 外部	6		

(1) 极性位 CLKRP=CLKXP=FSRP=FSXP=0。如果任一信号的极性被反转，则相应信号的时序基准也被反转。

(2) 2P=1/CLKG， 单位为 ns。CLKG 是采样率发生器复用的输出。 $CLKG = \frac{CLKSRG}{(1+CLKGDV)}$，CLKSRG 可由 LSPCLK、CLKX 和 CLKR 供源。CLKSRG≤(SYSCLKOUT/2)。McBSP 的性能受到 I/O 缓冲器开关速度的限制。

(3) 必须调节内部时钟预分频器，以使 McBSP 时钟 (CLKG、CLKX、CLKR) 速度不大于 I/O 缓冲器速度限制 (25MHz)。

表 6-40 McBSP 开关特性

编号	参数[(1)]				最小值	最大值	单位
M1	$t_{c(CKRX)}$	周期时间，CLKR/X		CLKR/X 内部	2P[(2)]		ns
M2	$t_{w(CKRXH)}$	脉冲持续时间，CLKR/X 高电平的时间		CLKR/X 内部	D-5 [(3)]	D+5 [(3)]	ns
M3	$t_{r(CKRXL)}$	脉冲持续时间，CLKR/X 低电平的时间		CLKR/X 内部	C-5 [(3)]	C+5 [(3)]	ns
M4	$t_{f(CKRH-FRV)}$	延迟时间,CLKR 高电平到内部 FSR 有效的时间		CLKR 内部	0	4	ns
				CLKR 外部	3	27	
M5	$t_{su(CKXH-FXV)}$	延迟时间,CLKX 高电平到内部 FSR 有效的时间		CLKR 内部	0	4	ns
				CLKR 外部	3	27	
M6	$t_{dis(CKXH-DXHZ)}$	禁用时间，CLKX 高电平到 DX 在最后一个数据位后为高阻抗的时间		CLKR 内部		8	ns
				CLKR 外部		14	
M7	$t_{d(CKXH-DXV)}$	延迟时间，CLKX 高电平到 DX 有效的时间。 这应用于除了第一个位之外的所有被发送的位。		CLKR 内部		9	ns
				CLKR 外部		28	
		延迟时间，CLKX 高电平到 DX 有效的时间 当处于数据延迟 1 或者2（XDATDLY=01b 或者 10b）模式时，只应用于发送的第一个位。	DXENA=0	CLKR 内部		8	
				CLKR 外部		14	
			DXENA=1	CLKR 内部		P+8	
				CLKR 外部		P+14	
M8	$t_{en(CKXH-DX)}$	使能时间，CLKX 高电平待 DX 被驱动的时间 当处于数据延迟 1或者2（XDATDLY=01b 或者 10b）模式时，只应用于发送的第一个位。	DXENA=0	CLKR 内部	0		ns
				CLKR 外部	6		
			DXENA=1	CLKR 内部	P		
				CLKR 外部	P+6		
M9	$t_{d(FXH-DXV)}$	延迟时间，FSX 高电平到 DX 有效的时间 当处于数据延迟 0（XDATDLY=00b）模式时，只应用于发送的第一个位。	DXENA=0	FSX 内部		8	ns
				FSX 外部		14	
			DXENA=1	FSX 内部		P+8	
				FSX 外部		P+14	
M10	$t_{d(FXH-DXV)}$	使能时间，FSX 高电平到 DX 驱动的时间 当处于数据延迟 0（XDATDLY=00b）模式时，只应用于发送的第一个位。	DXENA=1	FSX 内部	0		ns
				FSX 外部	6		
				FSX 内部	P		
				FSX 外部	P+6		

(1) 极性位 CLKRP=CLKXP=FSRP=FSXP=0。如果任一信号的极性被反转，则相应信号的时序基准也被反转。

(2) 2P=1/CLKG，单位为 ns。

(3) C=CLKRX 低脉冲宽度 = P；
D=CLKRX 高脉冲宽度 = P。

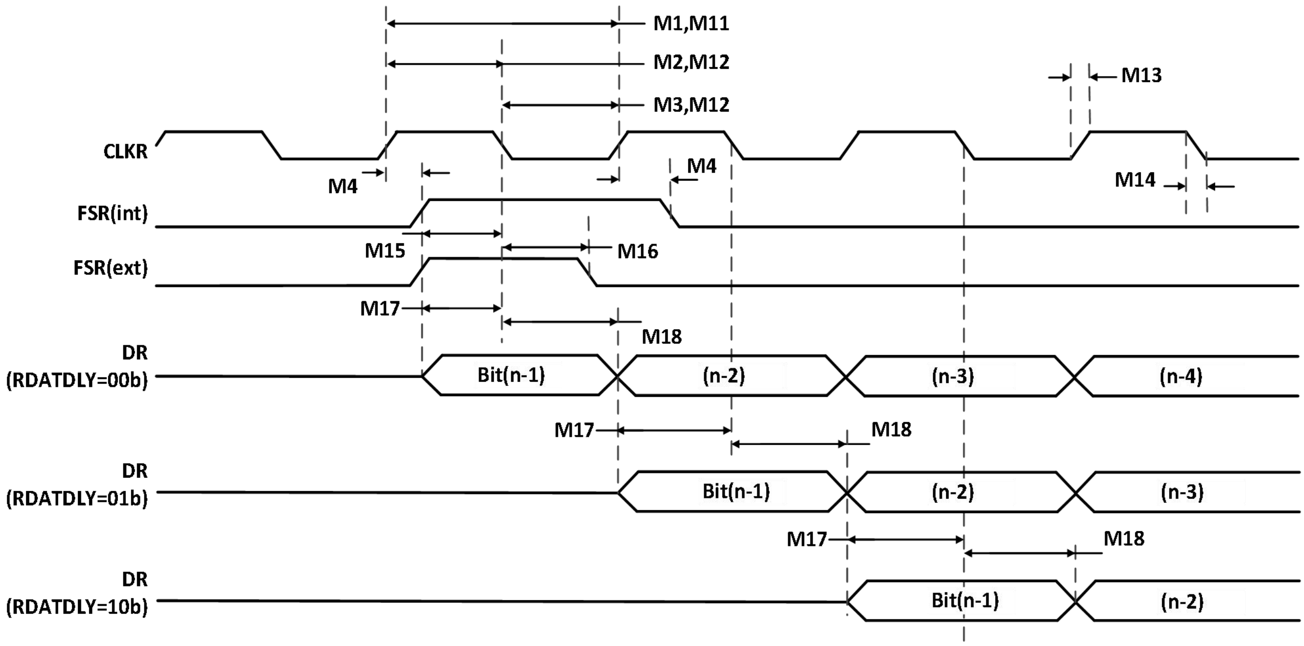


图 6-26 McBSP 接收时序

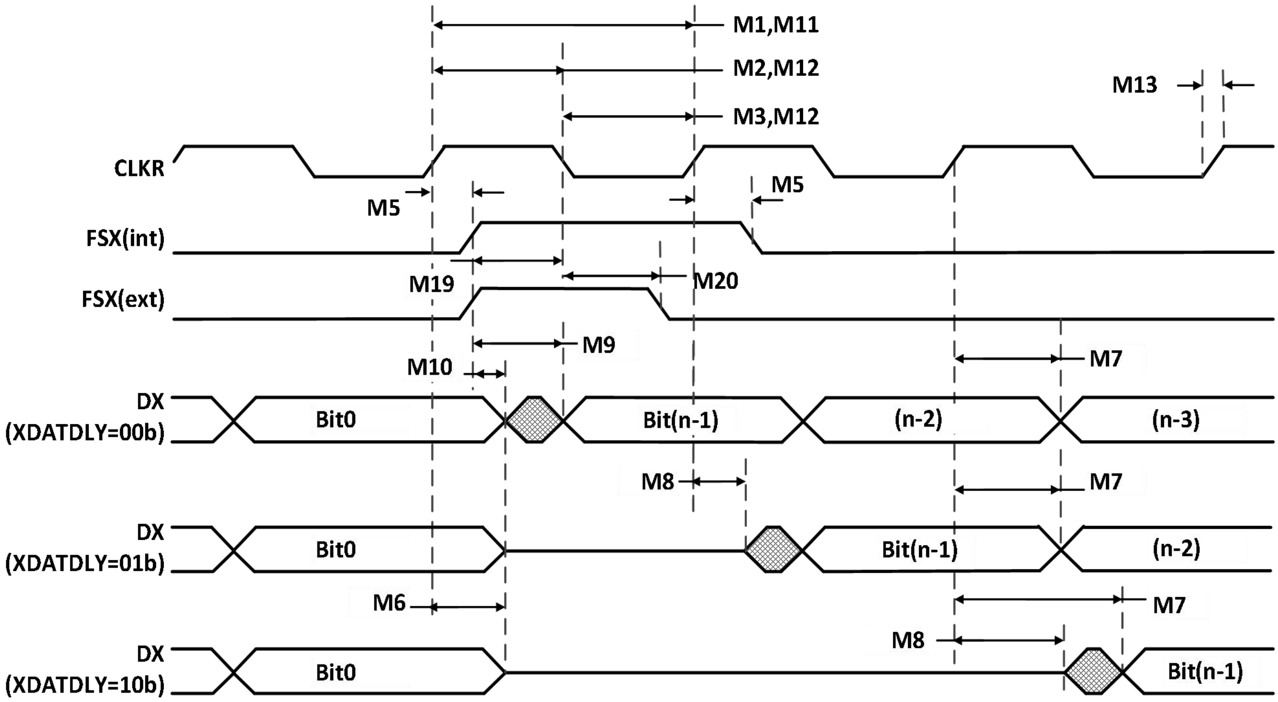


图 6-27 McBSP 发送时序

6.15.2 McBSP 作为 SPI 主模式或从模式时序

表 6-41 McBSP 作为 SPI 主模式或从模式时序要求 (CLKSTP=10b，CLKXP=0)

编号	参数		主模式		从模式		单位
			最小值	最大值	最小值	最大值	
M30	$t_{su(DRV\text{-}CKXL)}$	建立时间，在 CLKX 低电平之前，DR 的有效时间	30		8P-10		ns
M31	$t_{h(CKXL\text{-}DRV)}$	保持时间，在 CLKX 低电平之后，DR 的有效时间	1		8P-10		ns
M32	$t_{su(BFXL\text{-}CKXH)}$	建立时间，在 CLKX 高电平之前，FSX 为低电平的时间			8P+10		ns
M33	$t_{c(CKX)}$	周期时间，CLKX	2P(1)		16P		ns

(1) 2P=1/CLKG

表 6-42 McBSP 作为 SPI 主模式或从模式开关特性 (CLKSTP=10b，CLKXP=0)

编号	参数		主模式		从模式		单位
			最小值	最大值	最小值	最大值	
M24	$t_{h(CKXL\text{-}FXL)}$	保持时间，CLKX 低电平之后，FSX 为低电平的时间	2P(1)				ns
M25	$t_{d(FXL\text{-}CKXH)}$	延迟时间，FSX 低电平到 CLKX 变为高电平的时间	P				ns
M28	$t_{dis(FXH\text{-}DXHZ)}$	禁用时间，从 FSX 高电平到最后一个数据位后 DX 高阻抗的时间	6		6P+6		ns
M29	$t_{d(FXL\text{-}DXV)}$	延迟时间，FSX 低电平到 DX 有效时间	6		4P+6		ns

(1) 2P=1/CLKG

对于所有 SPI 从模式，CLKX 必须至少为 8 个 CLKG 周期。此外，通过设置 CLKSM=CLKGDV=1，CLKG 应该为 LSPCLK/2。在 LSPCLK 速度为 75MHz 的最大值时，CLKX 最大频率将达到 LSPCLK/16，即 4.6875 MHz 且 P=13.3ns。

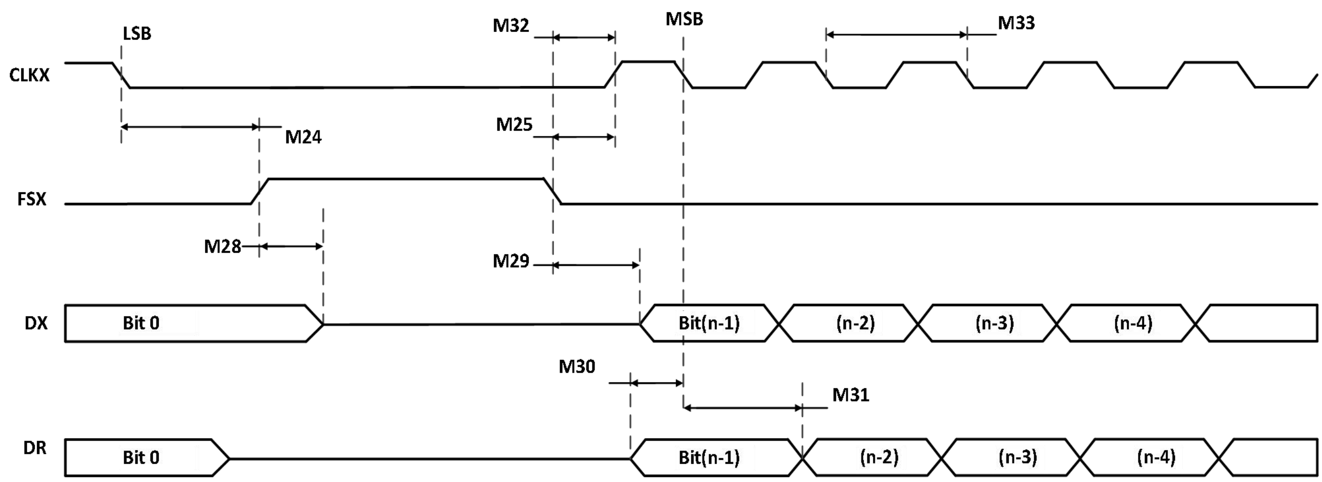


图 6-28 作为 SPI 主模式或者从模式时的 McBSP 时序：CLKSTP=10b，CLKXP=0

表 6-43 McBSP 作为 SPI 主模式或者从模式时序要求 (CLKSTP=11b，CLKXP=0)

编号	参数		主模式		从模式		单位
			最小值	最大值	最小值	最大值	
M39	$t_{su(DRV-CKXH)}$	建立时间，在 CLKX 高电平之前，DR 的有效时间	30		8P-10		ns
M40	$t_{h(CKXH-DRV)}$	保持时间，在 CLKX 高电平之后，DR 的有效时间	1		8P-10		ns
M41	$t_{su(FXL-CKXH)}$	建立时间，在 CLKX 高电平之前，FSX 为低电平的时间			16P+10		ns
M42	$t_{c(CKX)}$	周期时间，CLKX	2P(1)		16P		ns

(1) 2P=1/CLKG

表 6-44 McBSP 作为 SPI 主模式或者从模式开关特性 (CLKSTP=11b，CLKXP=0)

编号	参数		主模式		从模式		单位
			最小值	最大值	最小值	最大值	
M34	$t_{h(CKXL-FXL)}$	保持时间，CLKX 低电平之后，FSX 为低电平的时间	P				ns
M35	$t_{d(FXL-CKXH)}$	延迟时间，FSX 低电平到 CLKX 变为高电平的时间	2P(1)				ns
M37	$t_{dis(CKXL-DXHZ)}$	禁用时间，从 CLKX 低电平到最后一个数据位后 DX 高阻抗的时间	P+6		7P+6		ns
M38	$t_{d(FXL-DXV)}$	延迟时间，FSX 低电平到 DX 有效时间	6		4P+6		ns

(1) 2P=1/CLKG

对于所有 SPI 从模式，CLKX 必须至少为 8 个 CLKG 周期。此外，通过设置 CLKSM=CLKGDV=1，CLKG 应该为 LSPCLK/2。在 LSPCLK 速度为 75MHz 的最大值时，CLKX 最大频率将达到 LSPCLK/16，即 4.6875 MHz 且 P=13.3ns。

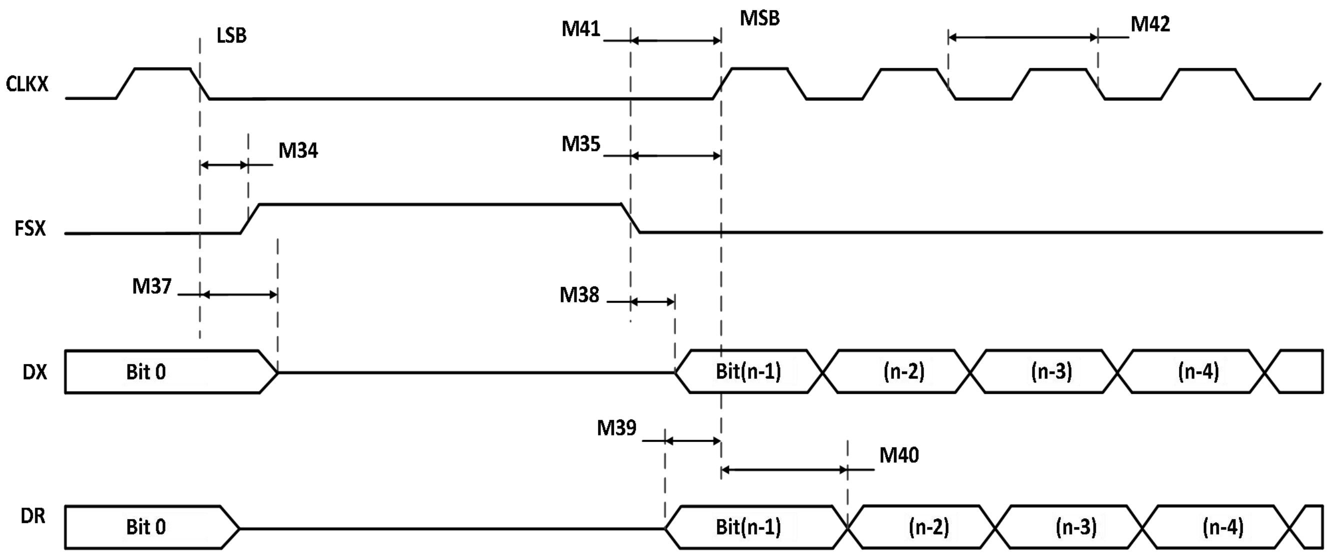


图 6-29 作为 SPI 主模式或从模式时的 McBSP 时序：CLKSTP=11b，CLKXP=0

表 6-45 McBSP 作为 SPI 主模式或者从模式时序要求 (CLKSTP=10b，CLKXP=1)

编号	参数		主模式		从模式		单位
			最小值	最大值	最小值	最大值	
M49	$t_{su(DRV-CKXH)}$	建立时间，在 CLKX 高电平之前，DR 的有效时间	30		8P-10		ns
M50	$t_{h(CKXH-DRV)}$	保持时间，在 CLKX 高电平之后，DR 的有效时间	1		8P-10		ns
M51	$t_{su(FXL-CKXL)}$	建立时间，在 CLKX 低电平之前，FSX 为低电平的时间			8P+10		ns
M52	$t_{c(CKX)}$	周期时间，CLKX	2P(1)		16P		ns

(1) 2P=1/CLKG

表 6-46 McBSP 作为 SPI 主模式或者从模式开关特性 (CLKSTP=10b，CLKXP=1)

编号	参数		主模式		从模式		单位
			最小值	最大值	最小值	最大值	
M43	$t_{h(CKXH-FXL)}$	保持时间，CLKX 高电平之后，FSX 为低电平的时间	2P(1)				ns
M44	$t_{d(FXL-CKXL)}$	延迟时间，FSX 低电平到 CLKX 变为低电平的时间	P				ns
M47	$t_{dis(FXH-DXHZ)}$	禁用时间，从 FSX 高电平到最后一个数据位后 DX 高阻抗的时间	6		6P+6		ns
M48	$t_{d(FXL-DXV)}$	延迟时间，FSX 低电平到 DX 有效时间	6		4P+6		ns

(1) 2P=1/CLKG

对于所有 SPI 从模式，CLKX 必须至少为 8 个 CLKG 周期。此外，通过设置 CLKSM=CLKGDV=1，CLKG 应该为 LSPCLK/2。在 LSPCLK 速度为 75MHz 的最大值时，CLKX 最大频率将达到 LSPCLK/16，即 4.6875 MHz 且 P=13.3ns。

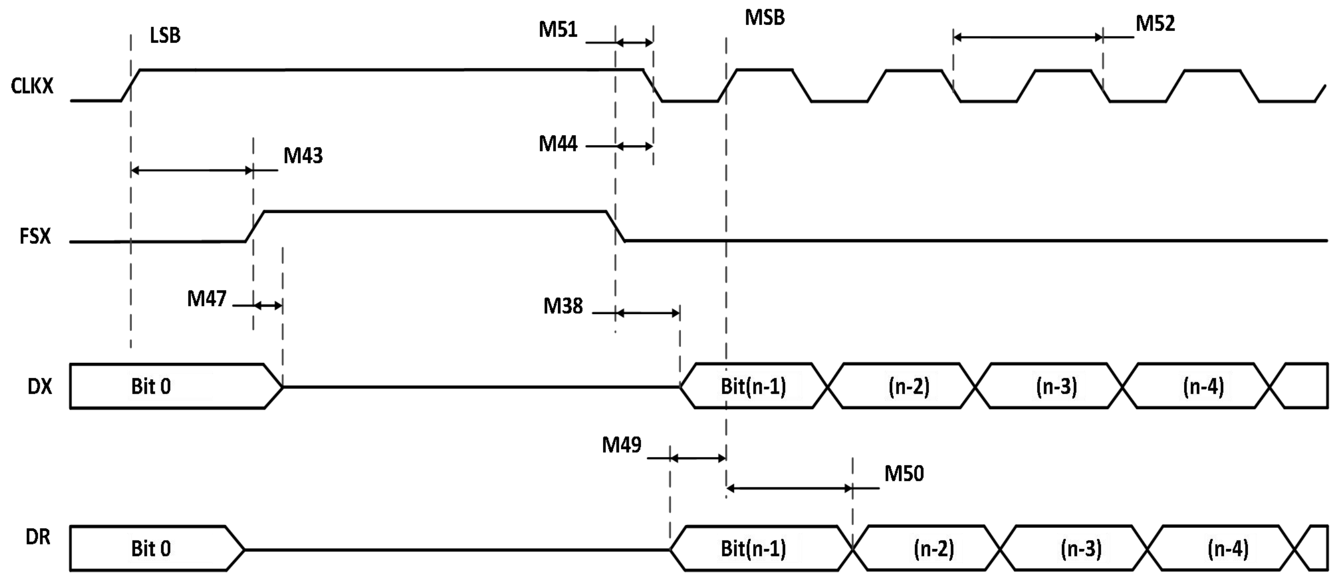


图 6-30 作为 SPI 主模式或者从模式时的 McBSP 时序：CLKSTP=10b，CLKXP=1

表 6-47 McBSP 作为 SPI 主模式或者从模式时序要求 (CLKSTP=11b，CLKXP=1)

编号	参数		主模式		从模式		单位
			最小值	最大值	最小值	最大值	
M58	$t_{su(DRV-CKXL)}$	建立时间，在 CLKX 低电平之前，DR 的有效时间	30		8P-10		ns
M59	$t_{h(CKXL-DRV)}$	保持时间，在 CLKX 低电平之后，DR 的有效时间	1		8P-10		ns
M60	$t_{su(FXL-CKXL)}$	建立时间，在 CLKX 低电平之前，FSX 为低电平的时间			16P+10		ns
M61	$t_{c(CKX)}$	周期时间，CLKX	2P(1)		16P		ns

(1) 2P=1/CLKG

表 6-48 McBSP 作为 SPI 主模式或从模式开关特性 (CLKSTP=11b，CLKXP=1) (1)

编号	参数		主模式(2)		从模式		单位
			最小值	最大值	最小值	最大值	
M53	$t_{h(CKXH-FXL)}$	保持时间，CLKX 高电平之后，FSX 为低电平的时间	P				ns
M54	$t_{d(FXL-CKXL)}$	延迟时间，FSX 低电平到 CLKX 变为低电平的时间	2P(1)				ns
M55	$t_{d(CLKXH-DXV)}$	延迟时间，CLKX 高电平到 DX 有效时间	-2	0	3P+6	5P+20	ns
M56	$t_{dis(CKXH-DXHZ)}$	禁用时间，从 CLKX 高电平到最后一个数据位后 DX 高阻抗的时间	P+6		7P+6		ns
M57	$t_{d(FXL-DXV)}$	延迟时间，FSX 低电平到 DX 有效时间	6		4P+6		ns

(1) 2P=1/CLKG

(2) C=CLKRX 低脉冲宽度 = P

D=CLKRX 高脉冲宽度 = P

对于所有 SPI 从模式，CLKX 必须至少为 8 个 CLKG 周期。此外，通过设置 CLKSM=CLKGDV=1，CLKG 应该为 LSPCLK/2。在 LSPCLK 速度为 75MHz 的最大值时，CLKX 最大频率将达到 LSPCLK/16，即 4.6875 MHz 且 P=13.3ns。

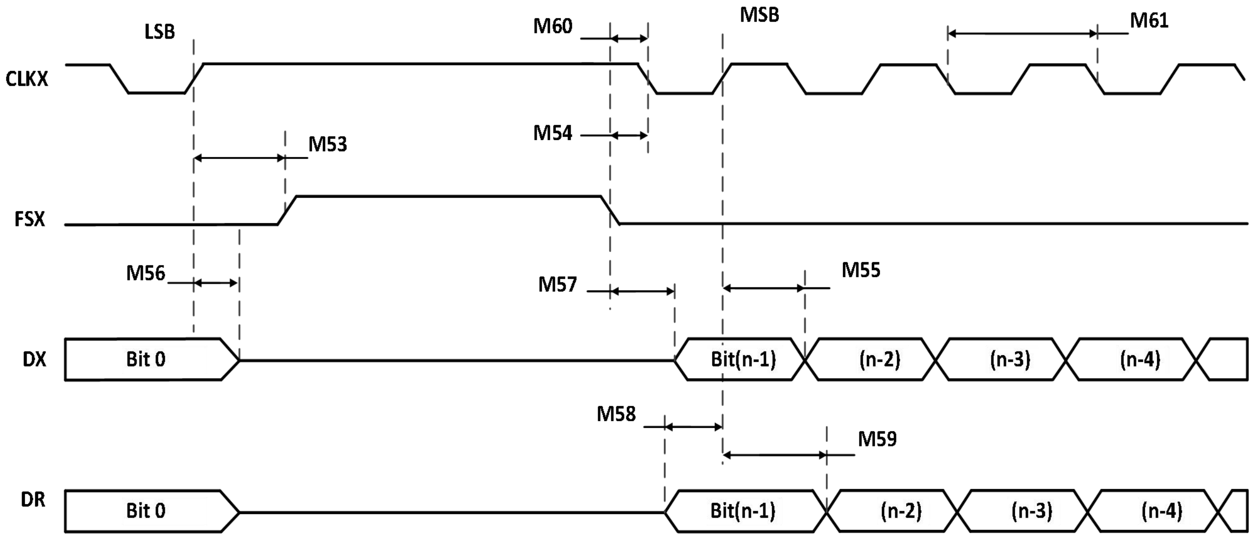


图 6-31 作为 SPI 主模式或者从模式时的 McBSP 时序：CLKSTP=11b，CLKXP=1

6.16 闪存时序

表 6-49 闪存对于（S）温度材料的耐受度(1)

	擦除/编程温度	最小值	典型值	最大值	单位
N_f 闪存对于阵列的耐受度（写入/擦除周期）	-40°C 至 85°C（环境温度）(2)	20000	50000		周期
N_{OTP} OTP 对于阵列的耐受度（写入周期）	-40°C 至 125°C（环境温度）			1	写入

(1) 所示温度范围之外的写入/擦除操作并未说明，有可能影响耐受次数。

(2) 建议 Flash 擦除和写入数据的温度范围-40°C至 85°C，读工作温度范围-40°C至 125°C。

表 6-50 150 MHz SYSCLKOUT 上的闪存参数 (3)

参数		测试条件	最小值	典型值	最大值	单位
编程时间(1)	16 位字			50		µs
	32K 扇区			1600		ms
擦除时间(1)	32K 扇区			900		ms
I_{DDP}(2)	擦除/编程周期期间的 V_{DD} 流耗			60		mA
I_{DDIOP}(2)	擦除/编程周期期间的 V_{DDIO} 流耗			5		mA

(1) 当器件出厂时，片载闪存存储器处于一个被擦除状态。这样，当首次编辑器件时，在编程前无需擦除闪存存储器。然而，对于所有随后的编程操作，需要执行擦除操作。

(2) 室温下包括函数调用开销在内的典型参数，是在所有外设关闭时的参数。

(3) AVP32F08 中 FLASH 总共 256K×16 位，分为 8 个扇区，每个扇区的空间为 32K×16 位。

表 6-51 闪存/ OTP 访问时序

参数		最小值	最大值	单位
$t_{a(fp)}$	页式闪存访问时间	37		ns
$t_{a(fr)}$	随机闪存访问时间	37		ns
$t_{a(OTP)}$	OTP 访问时间	60		ns

表 6-52 闪存数据保持时间

参数		测试条件	最小值	最大值	单位
$t_{retention}$	数据保持时间	T_J=55°C	15		年

表 6-53 不同频率上所需最小的闪存/一次性可编程 (OTP) 等待状态

SYSCLKOUT (MHz)	SYSCLKOUT (ns)	页等待状态 (1)	随机等待状态 (1)	OTP
150	6.67	5	5	8
120	8.33	4	4	7
100	10	3	3	5
75	13.33	2	2	4
50	20	1	1	2
30	33.33	1	1	1
25	40	1	1	1
15	66.67	1	1	1
4	250	1	1	1

(1) 随机等待状态必须大于或等于 1，用于计算表 6-53 中页等待状态和随机等待状态的公式如下：

$$\text{FlashPage Wait State} = \left[\left(\frac{t_{a(f\bullet p)}}{t_{c(SCO)}}\right) - 1\right]$$(四舍五入到下一个最大的整数)

$$\text{FlashRandom Wait State} = \left[\left(\frac{t_{a(f\bullet r)}}{t_{c(SCO)}}\right) - 1\right]$$(四舍五入到下一个最大的整数或 1，以最大的数为准)

计算表6-53中OTP等待状态的公式如下：

$$\text{OTP Wait State} = \left[\left(\frac{t_{a(OTP)}}{t_{c(SCO)}}\right) - 1\right]$$(四舍五入到下一个最大的整数或 1，以最大的数为准)

7 热性能/机械数据

表 7-1 概括了该型器件的热数据。有关散热设计考虑的更多信息请见第 6.4.3 节散热设计考虑。

表格之后的机械封装图反映了指定器件的机械数据。

表 7-1 散热模型 100 引脚 QP 结果

参数	0lfm
θ_{JA} [°C/W] 高 k PCB	46.78
Ψ_{JT} [°C/W]	0.28
θ_{JC}	11.74
θ_{JB}	28.75

LQFP100 单位：mm

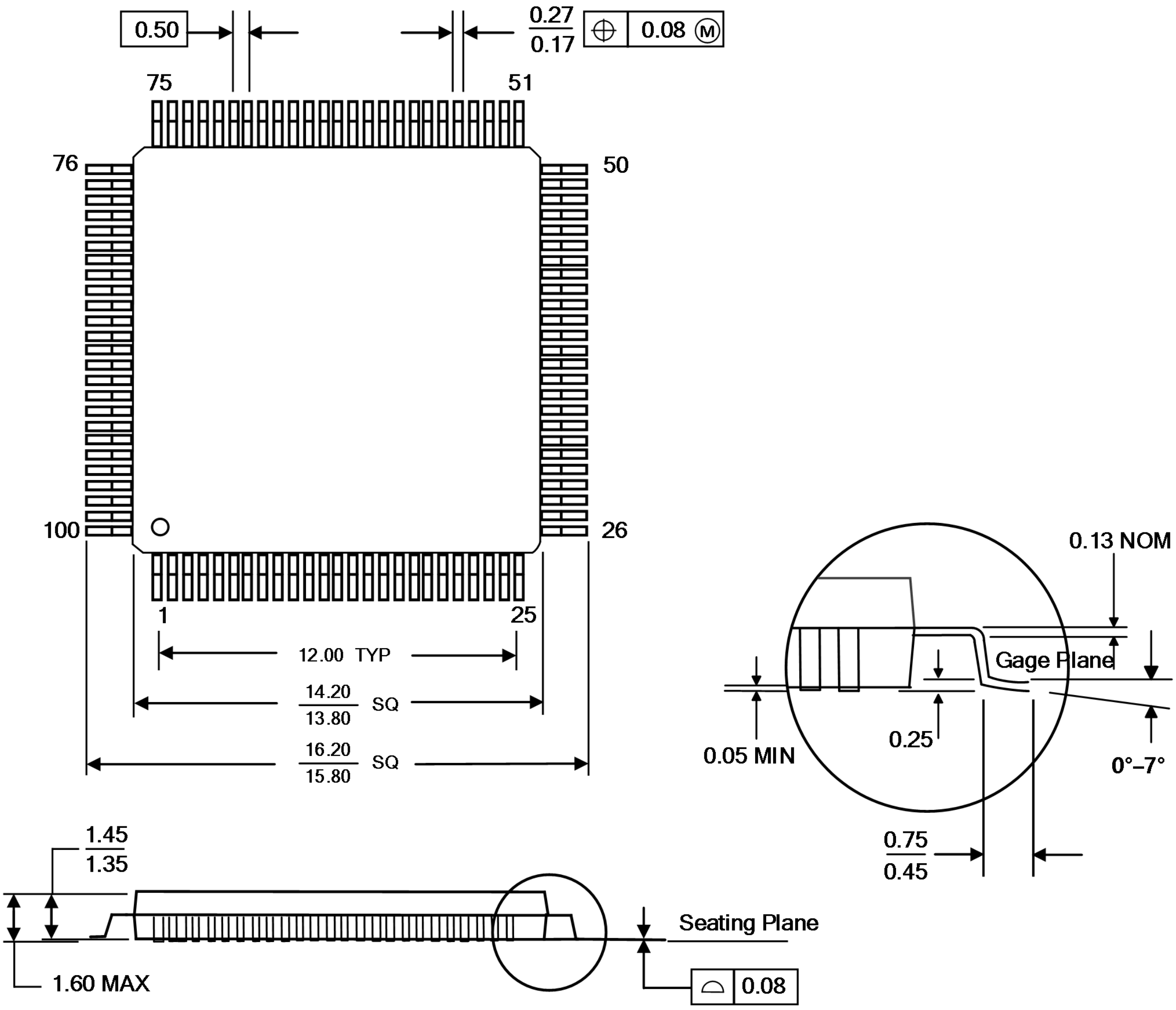


图 7 封装引脚尺寸图

联系方式

公司网址：www.advancechip.com

联系邮箱：sales@advancechip.com

销售联系电话：0731-88731027（长沙）

025-66051670（南京）

公司总部地址：长沙市湘江新区北斗产业园黄金园 A5 栋

南京销售中心：南京市雨花台区软件大道 106 号 2 号楼 802 室