

ADP32F08 数字信号处理器

数据手册

编号: JXDZ3.590.017SJSC

Advancechip



Electronics

湖南进芯电子科技有限公司

2021 年 10 月

V1.0

历史版本记录

版本号	时间	起草/ 修改人	内容描述	审核人	批准人
V1.0	2021-10-13	刘 杨	首次发布	吴修英	张 巍

目 次

1. 产品特征.....	1
2. ADP32F08 简介.....	2
2.1 产品编码.....	2
2.2 引脚分配.....	2
2.3 信号说明.....	3
3 功能概述.....	11
3.1 方框图.....	11
3.2 内存映射.....	12
3.3 简要说明.....	15
3.3.1 ADP32F08 DSP.....	15
3.3.2 内存总线 (哈弗总线架构)	16
3.3.3 外设总线.....	16
3.3.4 实时 JTAG 和分析.....	16
3.3.5 闪存.....	17
3.3.6 M0, M1 SARAM.....	17
3.3.7 L0, L1, H0 SARAM.....	17
3.3.8 引导 ROM.....	18
3.3.9 安全性.....	18
3.3.10 外设中断扩展 (PIE) 块.....	19
3.3.11 外部中断 (XINT1,XINT2,XNMI).....	19
3.3.12 振荡器和锁相环 (PLL)	20
3.3.13 看门狗.....	20
3.3.14 外设时钟.....	20
3.3.15 低功耗模式.....	20

3.3.16	外设帧 0, 1, 2, 3 (PFn)	20
3.3.17	通用输入/输出(GPIO) 复用器	21
3.3.18	32 位 CPU 定时器 (0, 1, 2)	21
3.3.19	控制外设	22
3.3.20	串行端口外设	22
3.4	寄存器映射	23
3.5	器件仿真寄存器	24
3.6	中断	25
3.7	系统控制	30
3.7.1	OSC 和 PLL 模块	32
3.7.1.1	外部基准振荡器时钟选项	34
3.7.1.2	基于 PLL 的时钟模块	34
3.7.1.3	时钟丢失检测	36
3.7.2	看门狗模块	36
3.8	低功耗模式模块	37
4	外设	38
4.1	32 位 CPU 定时器 0/1/2	38
4.2	增强型模数转换器 (ADC) 模块	40
4.2.1	如果 ADC 未被使用, ADC 连接	45
4.2.2	ADC 寄存器	45
4.3	串行通信接口 (SCI) 模块 (SCI-A, SCI-B)	47
4.4	串行外设接口 (SPI) 模块 (SPI-A, SPI-B, SPI-C, SPI-D)	50
4.5	增强型控制器局域网 (ECAN) 模块 (ECAN-A 和 ECAN-B)	55
4.6	内部集成电路 (I2C)	61
4.7	增强型 PWM 模块 (EPWM1/2/3/4/5/6)	63

4.8 高精度 PWM (HRPWM)	67
4.9 增强型捕捉模块 (ECAP1)	68
4.10 增强型正交编码器脉冲 (EQEP1/2)	70
4.11 GPIO MUX	72
5 电气规范	78
5.1 最大绝对额定值 ⁽¹⁾⁽²⁾	78
5.2 推荐的运行条件	79
5.3 电气特性	80
5.4 功耗	81
5.4.1 减少功耗	82
5.4.2 功耗图	82
5.5 针对 DSP 的无信号缓冲的仿真器连接	83
5.6 时序参数符号	83
5.6.1 定时参数的通用注释	84
5.6.2 测试负载电路	84
5.6.3 器件时钟表	85
5.7 时钟要求和特性	86
5.8 电源排序	87
5.8.1 电源管理和监控电路解决方案	88
5.9 通用输入/输出 (GPIO)	90
5.9.1 GPIO - 输出时序	90
5.9.2 GPIO - 输入时序	91
5.9.3 针对输入信号的采样窗口宽度	91
5.9.4 低功耗唤醒时序	92
5.10 增强型控制外设	96

5.10.1 增强型脉宽调制器 (ePWM) 时序.....	96
5.10.2 可编程控制故障区输入时序.....	96
5.10.3 高分辨率 PWM (HRPWM) 时序.....	97
5.10.4 增强型捕捉 (eCAP) 时序.....	97
5.10.5 增强型正交编码脉冲 (eQEP) 时序.....	98
5.10.6 ADC 转换开始时序.....	98
5.10.7 外部中断时序.....	99
5.10.8 I2C 电气特性和时序.....	99
5.10.9 同步串行通信接口(SPI) 主机模式时序.....	100
5.10.10 SPI 从机模式时序.....	103
5.10.11 片载模数转换器.....	106
5.10.11.1 定义.....	107
5.10.11.2 ADC 加电控制位时序.....	108
5.10.11.3 顺序采样模式 (单通道) (SMODE = 0).....	109
5.10.11.4 同步采样模式 (双通道) (SMODE=1).....	110
5.11 详细说明.....	111
5.12 闪存时序.....	113
6 机械数据.....	115

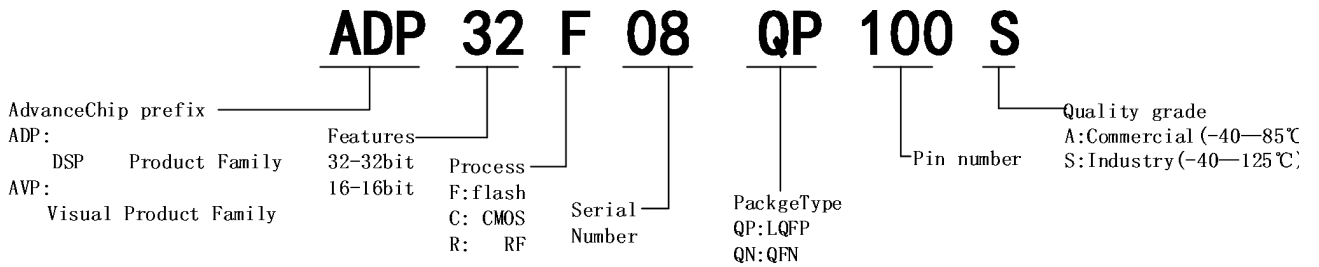
1. 产品特征

- 主频 100MHz (周期 10ns)
- 低功耗 1.5V 内核, 3.3V I/O 电源供电
- **JTAG 边界扫描**
 - 支持 IEEE 标准 1149.1-1990 标准测试端口和边界扫描架构
- **高效 32 位中央处理单元(CPU)**
 - 16 x 16 和 32 x 32 乘累加(MAC) 运算
 - 16 x 16 双 MAC
 - 哈佛(Harvard) 总线架构
 - 连动运算
 - 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)
 - 片上存储器
 - 64K x 16 闪存, 18K x 16 SARAM
 - 1K x 16 一次性可编程 (OTP) ROM
 - 引导 ROM (4K x 16)
 - 带有软件启动模式 (SCI, SPI, CAN, I2C 和并行 I/O)
 - 标准数学表
- **时钟和系统控制**
 - 片载振荡器/外部时钟输入
 - 支持动态锁相环路 (PLL) 比率变化
 - 看门狗模块
- **可支持所有外设中断的外设中断扩展(PIE) 模块**
- **128 位安全密钥/锁**
 - 保护安全内存块
 - 防止硬件逆向工程
- **三个 32 位 CPU 定时器**
- **增强型控制外设**
 - 16 个脉宽调制输出
 - 具有 150ps 微边界定位 (MEP) 的 6 个高分辨率 PWM (HRPWM) 输出
 - 4 个捕捉(CAP)输入
 - 2 个正交编码器 (QEP) 接口
- **串行端口外设**
 - 2 个 SCI(UART) 模块
 - 4个 SPI 模块
 - 2个增强型控制器局域网络(eCAN)总线
 - 1个I2C模块
- **12位模数转换器 (ADC), 16个通道**
 - 2 x 8通道输入
 - 2个采样保持
 - 单一/同步转换
 - 转换率: 160ns – 6.25MSPS
 - 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 GPIO**
- **低功耗模式**
 - 支持空闲 (IDLE)、待机 (STANDBY)、暂停 (HALT) 模式
- **高级仿真特性**
 - 分析和断点功能
 - 借助硬件的实时调试
- **封装选项**
 - 100 引脚 薄型四方扁平 (LQFP)封装

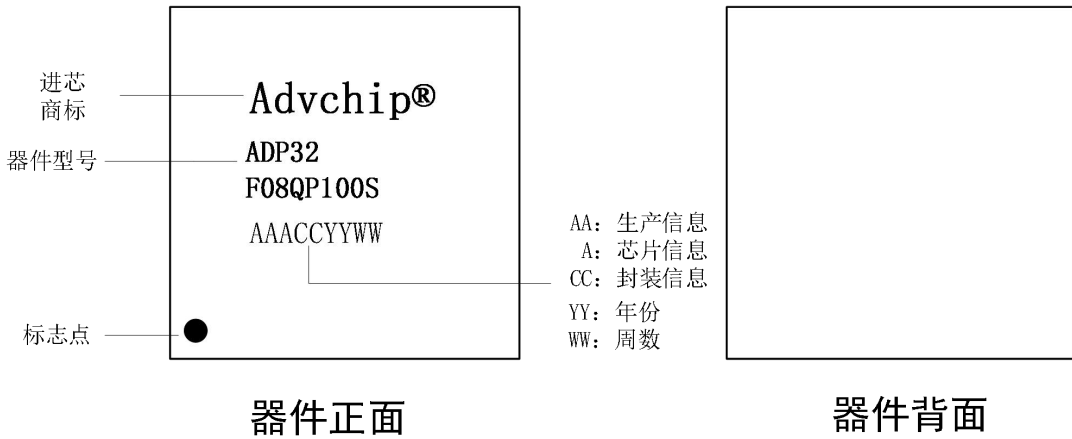
2. ADP32F08 简介

2.1 产品编码

ADP32F08 产品代号编码规则:



器件标识:



2.2 引脚分配

图 2-1 显示了 100 引脚 PN 薄型四方扁平封装 (LQFP) 引脚分配。

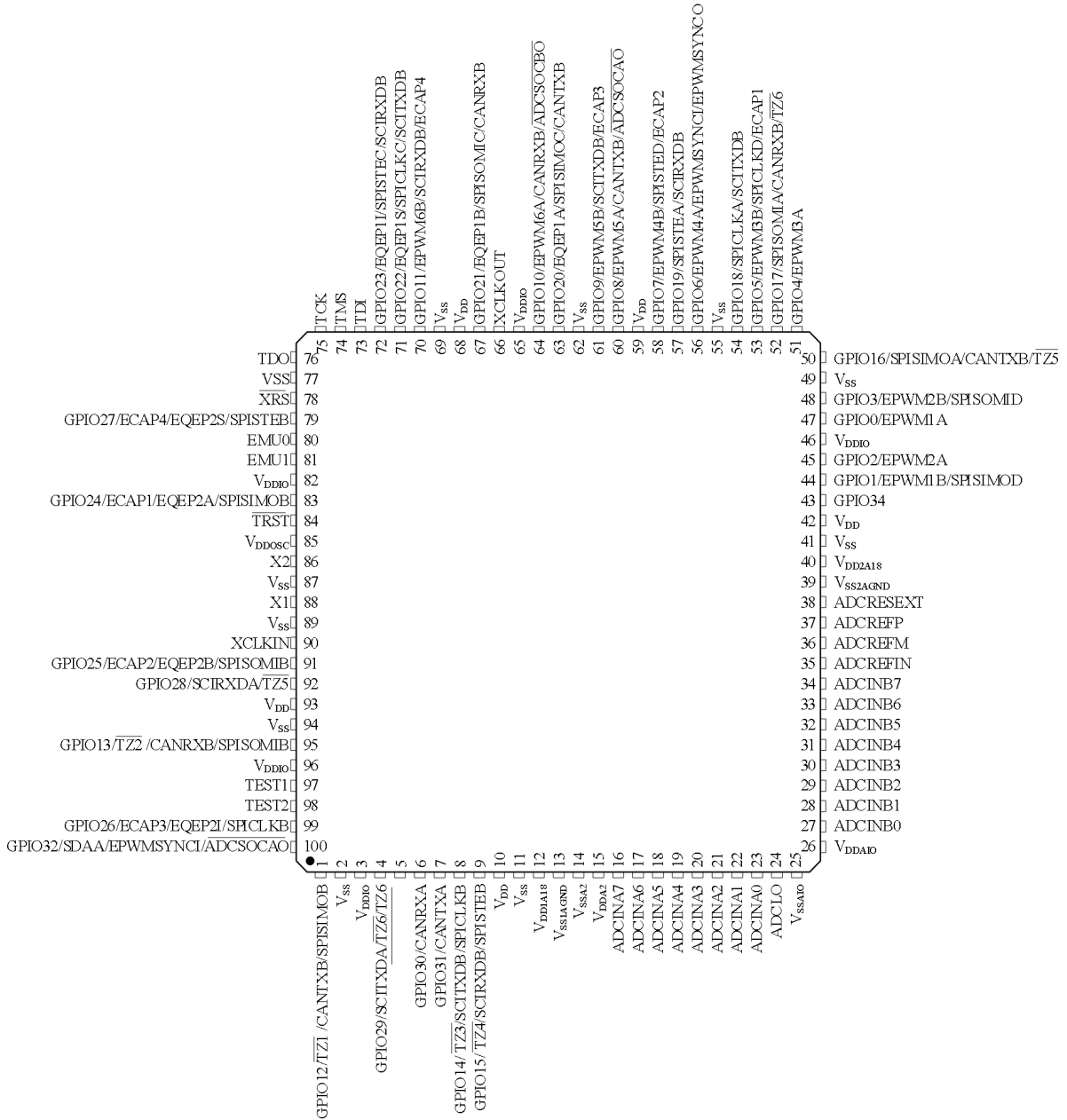


图 2-1 ADP32F08100 引脚 PN LQFP (顶视图)

2.3 信号说明

表 2-1 对这些信号进行了说明。

表 2-1 信号说明⁽¹⁾

名称	引脚编号		说明
	QFP100 引脚 #	I/O/Z	
JTAG信号			
TRST	84	I	<p>使用内部下拉进行JTAG测试复位。TRST,当被驱动为高电平时,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。</p> <p>注释:不要在TRST上使用上拉电阻器;它有一个内部下拉器件。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。强烈建议使用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。一个2.2kΩ电阻器一般提供足够的保护,由于这是应用专用的,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。</p>
TCK	75	I	带有内部上拉电阻器的 JTAG 测试时钟。
TMS	74	I	带有内部上拉电阻的JTAG测试模式选择(TMS)。这个串行控制输入在TCK上升沿锁存到TAP控制器中。
TDI	73	I	带有内部上拉电阻的JTAG测试数据输入(TDI)。TDI在TCK 上升沿锁存到寄存器(指令或者数据)
TDO	76	O/Z	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。
EMU0	80	I/O/Z	<p>仿真器引脚0。当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,TRST引脚的上升沿将把器件锁存在边界扫描模式。</p> <p>注释: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 的电阻器可以满足要求。由于这是应用专用的,建议针对调试器和应用正确运行对每个目标板进行验证。</p>
EMU1	81	I/O/Z	<p>仿真器引脚1。当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,TRST引脚的上升沿将把器件锁存在边界扫描模式。</p> <p>注释: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 的电阻器可以满足要求。由于这是应用专用的,建议针对调试器和应用正确运行对每个目标板进行验证。</p>
电源信号			
VDDIO	96		数字 I/O 电源引脚
VDDOSC	85		内部振荡器电源
TEST1	97		悬空/推荐接内核供电,外接退耦电容负端,退耦电容推荐不小于 2.2uF。
TEST2	98		悬空/推荐接内核供电,外接退耦电容正端,退耦电容推荐不小于 2.2uF。

(1) I = 输入, O = 输出, Z = 高阻抗, OD = 开漏

表 2-1 信号说明⁽¹⁾ (续)

名称	引脚编号	I/O/Z	说明
	QFP100 引脚 #		
时钟信号			
XCLKOUT	66	O/Z	取自 SYSCLKOUT 的输出时钟。XCLKOUT 可以与SYSCLKOUT 的频率一样、也可为其1/2或1/4。这由 XCLK 寄存器内的引脚 1:0 (XCLKOUTDIV) 控制。复位时, XCLKOUT=SYSCLKOUT/4。通过将 XCLKOUTDIV 设定为 3, XCLKOUT 信号可被关闭。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。
XCLKIN	90	I	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 1.9V 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。
X1	88	I	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。
X2	86	O	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, 它必须保持在未连接状态。
复位信号			
\overline{XRS}	78	I/O	器件复位 (输入) 和看门狗复位 (输出)。 \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。当一个安全装置复位发生时, 这个引脚被DSC 驱动至低电平。安全装置复位期间, 在 512个 OSCCLK 周期的安全装置复位持续时间内, \overline{XRS} 引脚被驱动为低电平。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。
模数转换器 (ADC)			
ADCINA7	16	I	ADC A组, 通道 7 输入
ADCINA6	17	I	ADC A组, 通道 6 输入
ADCINA5	13	I	ADC A组, 通道 5 输入
ADCINA4	19	I	ADC A组, 通道 4 输入
ADCINA3	20	I	ADC A组, 通道 3 输入
ADCINA2	21	I	ADC A组, 通道 2 输入
ADCINA1	22	I	ADC A组, 通道 1 输入
ADCINA0	23	I	ADC A组, 通道 0 输入
ADCINB7	34	I	ADC B组, 通道 7 输入
ADCINB6	33	I	ADC B组, 通道 6 输入
ADCINB5	32	I	ADC B组, 通道 5 输入
ADCINB4	31	I	ADC B组, 通道 4 输入

表 2-1 信号说明⁽¹⁾ (续)

名称	引脚编号	I/O/Z	说明
	QFP100 引脚 #		
ADCINB3	30	I	ADC B组, 通道 3 输入
ADCINB2	29	I	ADC B组, 通道 2 输入
ADCINB1	28	I	ADC B组, 通道 1 输入
ADCINB0	27	I	ADC B组, 通道 0 输入
ADCLO	24	I	低基准 (连接至模拟接地)
ADCRESEXT	38		ADC 外部电流偏置电阻器。 将一个 22kΩ 电阻器接至模拟接地。
ADCREFIN	35	I	外部基准输入(I)
ADCREFP	37		内部基准正输出。 要求将一个低等效串联电阻 (ESR) (低于 1.5Ω) 的 2.2μF 陶瓷旁通电容器接至模拟接地。 注释: 使用 ADC 时钟速率从系统使用的电容器数据表中提取 ESR 技术规范。
ADCREFM	36		内部基准中输出。 要求将一个低等效串联电阻 (ESR) (低于 1.5Ω) 的 2.2μF 陶瓷旁通电容器接至模拟接地。 注释: 使用 ADC 时钟速率从系统使用的电容器数据表中提取 ESR 技术规范。
电源信号			
V _D DA2	15		ADC 模拟电源引脚
V _S SA2	14		ADC 模拟接地引脚
V _D DAIO	26		ADC 模拟 I/O 电源引脚
V _S SAIO	25		ADC 模拟 I/O 接地引脚
V _D D1A18	12		ADC 模拟电源引脚 (内部悬空)
V _S S1AGND	13		ADC 模拟接地引脚
V _D D2A18	40		ADC 模拟电源引脚 (内部悬空)
V _S S2AGND	39		ADC 模拟接地引脚
V _D D	10		数字电源引脚 (内部悬空)
V _D D	42		
V _D D	59		
V _D D	68		
V _D D	93		
V _D DDIO	3		数字 I/O 电源引脚
V _D DDIO	46		
V _D DDIO	65		
V _D DDIO	82		
V _S S	2		数字接地引脚
V _S S	11		
V _S S	41		
V _S S	49		
V _S S	55		
V _S S	62		
V _S S	69		
V _S S	77		
V _S S	87		
V _S S	89		
V _S S	94		

表 2-1 信号说明⁽¹⁾ (续)

名称	引脚编号	I/O/Z	说明
	QFP100 引脚 #		
GPIO 和外设信号 ⁽¹⁾			
<i>GPIO0</i> EPWM1A	47	I/O/Z O	通用输入/输出 0 增强型 PWM1 输出 A 和高分辨率脉宽调制 (HRPWM) 通道
<i>GPIO1</i> EPWM1B SPISIMOD	44	I/O/Z O O	通用输入/输出 1 增强型 PWM1 输出 B SPI-D从器件输入, 主器件输出
<i>GPIO2</i> EPWM2A	45	I/O/Z O	通用输入/输出2 增强型 PWM2 输出 A 和 HRPWM 通道
<i>GPIO3</i> EPWM2B SPISOMID	48	I/O/Z O I/O	通用输入/输出3 增强型 PWM2 输出 B SPI-D从器件输出, 主器件输入
<i>GPIO4</i> EPWM3A	51	I/O/Z O	通用输入/输出 4 增强型 PWM3 输出 A 和 HRPWM 通道
<i>GPIO5</i> EPWM3B SPICLKD ECAP1	52	I/O/Z O I/O I/O	通用输入/输出5 增强型 PWM3 输出 B SPI-D时钟 增强型捕捉输入/输出 1
<i>GPIO6</i> EPWM4A EPWMSYNC I EPWMSNC O	56	I/O/Z O I O	通用输入/输出 6 增强型 PWM4 输出 A 和 HRPWM 通道 外部 ePWM同步脉冲输入 外部 ePWM同步脉冲输出
<i>GPIO7</i> EPWM4B SPISTED ECAP2	58	I/O/Z O I/O I/O	通用输入/输出 7 增强型 PWM4 输出 B SPI-D从器件发送启用 增强型捕捉输入/输出 2
<i>GPIO8</i> EPWM5A CANTXB ADCSOCAO	60	I/O/Z O O O	通用输入/输出 8 增强型 PWM5 输出 A 和 HRPWM 通道 增强型 CAN-B 发送 ADC 转换启动 A

(1) GPIO 功能 (用粗斜体显示) 在复位时为默认值。它们下面列出的外设信号是供替换的功能。对于有 GPIO 功能复用的 JTAG 引脚, 到 GPIO

(2) 块的输入路径一直有效。根据 TRST 信号的情况, GPIO 模块的输出路径和引脚到 JTAG 模块的路径被选择启用/禁用。

表 2-1 信号说明⁽¹⁾ (续)

名称	引脚编号	I/O/Z	说明
	QFP100 引脚 #		
GPIO9 EPWM5B SCITXDB ECAP3	61	I/O/Z O O I/O	通用输入/输出 9 增强型PWM5 输出 B SCI-B 发送数据 增强型捕捉输入/输出 3
GPIO10 EPWM6A CANRXB ADCSOCBO	64	I/O/Z O I O	通用输入/输出 10 增强型 PWM6 输出 A 和 HRPWM 通道 增强型 CAN-B 接收 ADC 转换启动 B
GPIO11 EPWM6B SCIRXDB ECAP4	70	I/O/Z O I I/O	通用输入/输出 11 增强型 PWM6 输出 B SCI-B 接收数据 增强型 CAP 输入/输出 4
GPIO12 TZ1 CANTXB SPISIMOB	1	I/O/Z I O I/O	通用输入/输出 12 触发区输入 1 增强型 CAN-B 传输 SPI-B从器件输入, 主器件输入
GPIO13 TZ2 CANRXB SPISOMIB	95	I/O/Z I I I/O	通用输入/输出 13 触发区输入 2 增强型 CAN-B 接收 SPI-B从器件输出, 主器件输入
GPIO14 TZ3 SCITXDB SPICLKB	8	I/O/Z I O I/O	通用输入/输出 14 触发区输入3 SCI-B 发送 SPI-B时钟输入/输出
GPIO15 TZ4 SCIRXDB SPISTEB	9	I/O/Z I I I/O	通用输入/输出 15 触发区输入 4 SCI-B 接收 SPI-B从器件发送启用
GPIO16 SPISIMOA CANTXB TZ5	50	I/O/Z I/O O I	通用输入/输出 16 SPI-A 从器件输入, 主器件输出 增强型 CAN-B 发送 触发区输入5
GPIO17 SPISOMIA CANRXB TZ6	52	I/O/Z I/O I I	通用输入/输出 17 SPI-A 从器件输出, 主器件输入 增强型 CAN-B 接收 触发区输入6

表 2-1 信号说明⁽¹⁾ (续)

名称	引脚编号	I/O/Z	说明
	QFP100 引脚 #		
GPIO18 SPICLKA SCITXDB	54	I/O/Z I/O O	通用输入/输出 18 SPI-A 时钟输入/输出 SCI-B 发送
GPIO19 SPISTEA SCIRXDB	57	I/O/Z I/O I	通用输入/输出 19 SPI-A 从器件发送启用 SCI-B 接收
GPIO20 EQEP1A SPISIMOC CANTXB	63	I/O/Z I I/O O	通用输入/输出 20 增强型 QEP1 输入 A SPI-C从器件输入, 主器件输出 增强型 CAN-B 发送
GPIO21 EQEP1B SPISOMIC CANRXB	67	I/O/Z I I/O O	通用输入/输出 21 增强型 QEP1 输入 B SPI-C主器件输入, 从器件输出 增强型 CAN-B 接收
GPIO22 EQEP1S SPICLKC SCITXDB	71	I/O/Z I/O O O	通用输入/输出 22 增强型 QEP1 选通脉冲 SPI-C时钟 SCI-B 传输
GPIO23 EQEP1I SPISTEC SCIRXDB	72	I/O/Z I/O I/O I	通用输入/输出 23 增强型 QEP1 索引 SPI-C从器件发送启用 SCI-B 接收
GPIO24 ECAP1 EQEP2A SPISIMOB	83	I/O/Z I/O I I/O	通用输入/输出 24 增强型捕获 1 增强型 QEP2 输入 A SPI-B从器件输入, 主器件输出
GPIO25 ECAP2 EQEP2B SPISOMIB	91	I/O/Z I/O I I/O	通用输入/输出 25 增强型捕获 2 增强型 QEP2 输入 B SPI-B主器件输入, 从器件输出

表 2-1 信号说明⁽¹⁾ (续)

名称	引脚编号	I/O/Z	说明
	QFP100 引脚 #		
GPIO26 ECAP3 EQEP2I SPISOMIB	99	I/O/Z I/O I/O O	通用输入/输出 26 增强型捕获 3 增强型 QEP2 索引 (SPI-B时钟
GPIO27 ECAP4 EQEP2S SPISTEB	79	I/O/Z I/O I/O I/O	通用输入/输出 27 增强型捕获 4 增强型 QEP2 选通脉中 SPI-B从器件发送启用
GPIO28 SCIRXDA TZ5	92	I/O/Z I I	通用输入/输出 28 SCI 接收数据 触发区输入5
GPIO29 SCITXDA TZ6	4	I/O/Z O I	通用输入/输出 29 SCI 发送数据 触发区输入6
GPIO30 CANRXA	6	I/O/Z I	通用输入/输出 30 增强型 CAN-A 接收
GPIO31 CANTXA	7	I/O/Z O	通用输入/输出 31 增强型 CAN-A 发送数据
GPIO32 SDAA EPWMSYNC I ADCSOCA0	100	I/O/Z I/OD I O	通用输入/输出 32 I2C 数据开漏双向端口 增强型 PWM 外部同步脉冲输入 ADC 转换启动 A
GPIO33 SCLA EPWMSYNC O ADCSOCB0	5	I/O/Z I/OD O O	通用输入/输出 33 I2C 时钟开漏双向端口 增强型 PWM 外部同步脉冲输出 ADC 转换启动 B
GPIO34	43	I/O/Z	通用输入/输出 34

3 功能概述

3.1 方框图

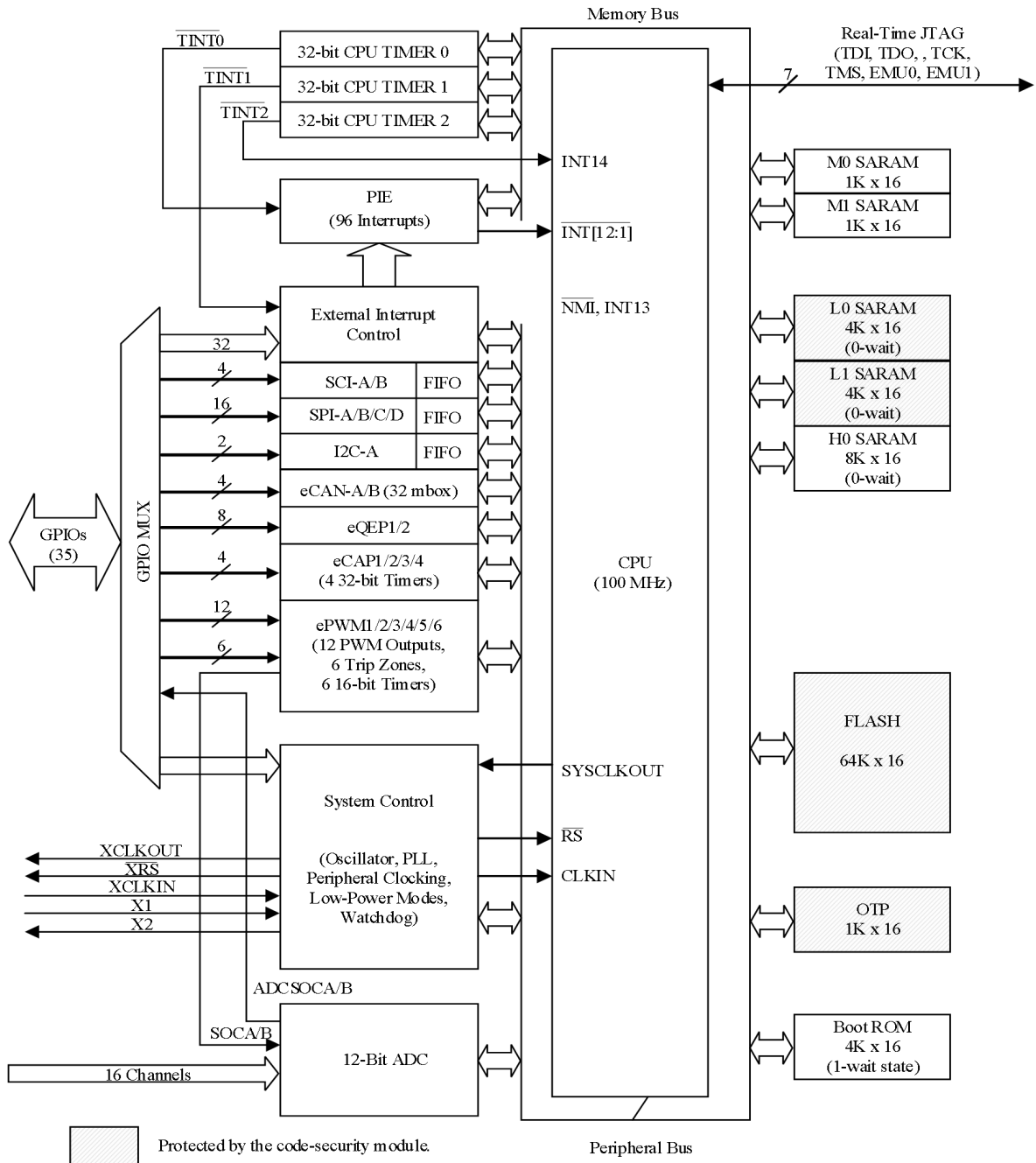


图 3-1 功能方框图

3.2 内存映射

- 内存块不可升级。
- 外设帧 0，外设帧 1，外设帧 2 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。
- 受保护意味着写后读操作的顺序被保存，而不是流水线顺序。
- 特定内存区域受 EALLOW 保护以防止配置之后的假写入。

Block Start Address	Data Space	Prog Space
0x00 0000	M0 Vector – RAM (32 x 32) (Enabled if VMAP = 0)	
0x00 0040	M0 SARAM(1K×16)	
0x00 0400	M1 SARAM(1K×16)	
0x00 0800	Peripheral Frame 0	Reserved
0x00 0D00	PIE Vector-RAM (256×16) (Enabled if ENPIE=1)	
0x00 0E00	Reserved	
0x00 6000	Peripheral Frame 1 (protected)	Reserved
0x00 7000	Peripheral Frame 2 (protected)	
0x00 8000	L0 SARAM (0-wait) (4K×16, Secure Zone, Dual-Mapped)	
0x00 9000	L1 SARAM (0-wait) (4K×16, Secure Zone, Dual-Mapped)	
0x00 A000	H0 SARAM (0-wait) (4K×16, Dual-Mapped)	
0x00 C000	Reserved	
0x3D 7800	OTP (1K×16, Secure Zone)	
0x3D 7C00	Reserved	
0x3D 8000	FLASH (64K×16, Secure Zone)	
0x3F 7FF8	128-bit Password	
0x3F 8000	L0 SARAM (0-wait) (4K×16, Secure Zone, Dual-Mapped)	
0x3F 9000	L1 SARAM (0-wait) (4K×16, Secure Zone, Dual-Mapped)	
0x3F A000	H0 SARAM (0-wait) (8K×16, Dual-Mapped)	
0x3F C000	Reserved	
0x3F F000	Boot ROM (4K×16)	
0x3F FFC0	Vectors (32×32) (enabled if VMAP = 1, ENPIE = 0)	

图 3-2 ADP32F08 内存映射

表 3-1 ADP32F08 中闪存扇区的地址

地址范围	程序和数据空间
0x3E 8000-0x3E BFFF	扇区 D (16K x 16)
0x3E C000-0x3E FFFF	扇区 C (16K x 16)
0x3F 0000-0x3F 3FFF	扇区 B (16K x 16)
0x3F 4000-0x3F 7F7F	扇区 A (16K x 16)
0x3F 7F80-0x3F 7FF5	当使用代码安全模块时，编程为 0x0000
0x3F 7FF6-0x3F 7FF7	引导至闪存进入点（程序分支指令）
0x3F 7FF8-0x3F 7FFF	安全密码（128 位）（不要设定为全零）

注

- 当代码安全密码被编辑时，0x3F7F80 到 0x3F7FF5 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 0x3F 7F80 至 0x3F 7FEF 可被用于代码或者数据。地址 0x3F 7FF0-0x3F 7FF5 为数据保留且不能包含程序代码。

表 3-2 显示如何处理这些内存地址。

表 3-2 使用安全代码模块的影响

地址范围	闪存	
	代码安全被启用	代码安全被禁用
0x3F 7F80-0x3F 7FEF	用 0x0000 填充	应用代码和数据。
0x3F 7FF0-0x3F 7FF5		只为数据保留。

外设帧 1，外设帧 2 被编成一组以使这些块成为“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 ADP32Fx 的流水线，在对不同内存位置读取之前的写入操作将出现在 CPU 内存总线上相反的顺序。这会导致特定外设应用中的问题，在此类应用中，用户认为写入会首先发生（如文档所描述的那样）。ADP32Fx CPU 支持一个块保护模式，在这个模式中，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发生（代价是会增加额外的周期以校正运行）。可对这个模式进行编程，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列在表 3-3 中。

表 3-3 等待状态

区域 (AREA)	等待状态 (CPU)	注释
M0 和 M1 SARAM	0 - 等待	固定的
外设帧 0	0 - 等待	固定的
外设帧 1	0 - 等待 (写入) 2 - 等待 (读取)	固定的 eCAN 外设可以按照需求扩展一个周期。背靠背写入将引进一个周期的延时
外设帧 2	0 - 等待 (写入) 2 - 写入 (读取)	固定的
L0 SARAM	0 - 等待	
L1 SARAM	0 - 等待	
OTP	可编程 最少 1 - 等待	由闪存寄存器设定。
闪存	可编程 最少 0 - 等待	由闪存寄存器设定。
H0 SARAM	0 - 等待	固定的。
引导 - ROM	1- 等待	固定的。

3.3 简要说明

3.3.1 ADP32F08 DSP

ADP32F08 DSP 系列是进芯电子定点 DSP 平台上的最新产品。它是一款非常高效的 C/C++ 引擎，此引擎不但能够让用户能够用高级语言开发他们的控制系统软件，还能够使用 C/C++ 开发数学算法。此器件在处理 DSP 算术任务时与处理系统控制任务时同样有效，而系统控制任务通常由微控制器器件处理。这样的效率在很多系统中省却了对第二个处理器的需要。32x32 位 MAC 功能和它的 64 位处理能力，使得该器件能够有效处理较高数字分辨率问题。添加了带有关键寄存器自动环境保存的快速中断响应，使得器件能够用最小的延迟处理很多异步事件。此器件有一个具有流水线式存储器访问的 8 级深度受保护流水线。这个流水线式操作使得该器件能够高速执行指令而无需求助于昂贵的高速存储器。特别分支超前硬件大大减少了条件不连续而带来的延迟。特殊存储条件操作进一步提升了性能。

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。ADP32Fx 内存总线架构包含一个程序读取总线、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 ADP32Fx 能够在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

ADP32F08 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 ADP32F08 器件上支持两个版本的外设总线。一个版本只支持 16 位访问（被称为外设帧 2）。另外一个版本支持 16 位和 32 位访问（被称为外设帧 1）。

3.3.4 实时 JTAG 和分析

此器件执行标准 IEEE1149.1 JTAG 接口。此外，此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。用户也可以通过非时间关键代码进行单步操作，同时可在没有干扰的情况下启用即将被处理的时间关键中断。此器件在 CPU 的硬件内执行实时模式。这是 ADP32Fx 系列器件所特有的功能，无需软件监控。此外，还提供了特别分析硬件，以使

用户能够设定硬件断电或者数据/地址观察点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

ADP32F08 器件包含 64K x 16 的嵌入式闪存存储器，被分别放置在 4 个 16K x 16 扇区内。还包含一个 1K x 16 OTP 内存，其地址范围为 0x3D 7800-0x3D 7BFF。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。然而，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。提供了特殊内存流水线操作以使闪存模块实现更高的性能。闪存 / OTP 被映射到程序和数据空间；因此，它可被用于执行代码或者存储数据信息。地址 0x3F 7FF0-0x3F 7FF5 为数据变量保留且不能包含程序代码。

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远远快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

器件包含这两块单周期访问内存，每一个的大小为 1K x 16。复位时，堆栈指针指向块 M1 的开始位置。M0 和 M1 块被映射到程序和数据空间。因此，用户能够使用 M0 和 M1 来执行代码或者用于存储数据变量。

3.3.7 L0, L1, H0 SARAM

ADP32F08 有一个 16K x 16 单周期存取的 RAM。分为三个区块 (L0-4K、L1-4K、H0-8K)。为了尽量减少 CPU 阻滞，每个块都可以独立访问。每个块都映射到程序空间和数据空间。

3.3.8 引导 ROM

引导 ROM 在出厂时就完成了固件编程。启动模式信号用于上电时告知引导固件使用什么引导模式。用户可以选择正常启动或从外部连接下载所需的软件代码或从内部的 FLASH 中选择所需的引导程序。引导 ROM 还包含标准向量表，例如 SIN/COS 波形，用来数学相关的算法。

表 3-4 引导模式选择

模式	描述	GPIO18 SPICLKA SCITXDB	GPIO29 SCITXDA	GPIO34
引导至 FLASH	跳转到 FLASH 地址 0x3F7FF6，需要在复位之前安插一个跳转指令来让代码跳转到所需执行的地址	1	1	1
SCI-A 引导	从 SCI-A 加载数据流	1	1	0
SPI-A 引导	从 SPI-A 上的外部串行 SPI EEPROM 加载	1	0	1
I2C 引导	从外部 EEPROM 加载数据在地址为 0x50 的 I2C 总线上	1	0	0
eCAN-A 引导	调用 CAN 引导程序从 eCAN-A 邮箱 1 加载	0	1	1
引导至 MOSARAM	跳转到 MOSARAM 地址为 0x0	0	1	0
引导至 OTP	跳转到 OTP 地址为 0x3D7800	0	0	1
并行 I/O 引导	通过 GPIO0-GPIO5 装载数据	0	0	0

3.3.9 安全性

ADP32Fx 支持高级别安全以保护用户固定不受逆向工程损坏。这个安全性有一个 128 位密码（16 个等待状态的硬编码），此密码由用户编辑入闪存。一个代码安全模块（CSM）被用于保护闪存 /ROM/OTP 和 L0/L1 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密码位置内的值相匹配的正确的 128 位“KEY（密钥）”值。

注

- 128 位密码（位于 0x3F7FF8-0x3F7FFF）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）被设计用于对存储在相关内存（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这个器件的保修期。

但是，原厂不保证或表示CSM不会被损坏或破坏，或不能通过其它方法存取关联的存储器中存储的数据。而且，除了上述内容外，原厂也未对本器件的CSM或操作做任何保证或表示，包括任何隐含的用于特定用途的商用性或适用性保证。

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 个外设中断。在 ADP32F08 上，外设使用 96 个可能中断中的 43 个。96 个中断被分成 8 组，每组被提供 12 个 CPU 中断线（INT1 或者 INT12）中的 1 个。96 个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/禁用。

3.3.11 外部中断 (XINT1,XINT2,XNMI)

ADP32F08支持三个可屏蔽的外部中断(XINT1,XINT2, XNMI)。XNMI可被连接至INT13或者CPU的NMI中断。这些中断可被选择为负边沿、正边沿、或者二者同时触发，并且可被启用或禁用。这些中断还包含一个16 位自由运行递增计数器，当检测到一个有效的中断边沿信号时，此计数器被重置为0。这个计数器可被用于为中断精确计时。没有用于外部中断的专用引脚，外部中断可以配置为组A中的

GPIO触发。

3.3.12 振荡器和锁相环 (PLL)

此器件可由一个外部振荡器或者一个连接至片载振荡器电路的晶振中的任一个计时。PLL 支持 10 个输入时钟缩放比。PLL 比率可利用软件在器件运行时进行更改，这使得用户在没有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 模块可被配置为旁路模式。

3.3.13 看门狗

器件包含一个看门狗模块：用户软件必须在一个特定的时间范围内定期复位看门狗计数器，否则看门狗模块会产生一个复位信号，复位全器件。此模块可关闭。

3.3.14 外设时钟

在外设不需要使用时，每个独立外设的时钟可被禁用以减少功耗。此外，到串行端口（不包括 I2C 和 eCAN）的系统时钟可按照 CPU 时钟进行缩放，根据实际需要平衡串口上数据和 CPU 数据之间的交互。

3.3.15 低功耗模式

ADP32F08 器件是完全静态 CMOS 器件。提供三个低功耗模式：

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。激活外设的中断和看门狗能够把处理器从 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。外部中断事件能够唤醒处理器和外设。在检测到中断事件之后的下一个有效周期后，开始执行。

HALT：关闭内部晶振。该模式会关断器件并将器件置于尽可能低的功耗模式中。复位信号或者外部信号能够将器件从这个模式唤醒。

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

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：GPIO：GPIO MUX 配置和控制寄存器

eCAN：eCAN 邮箱和控制寄存器

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

ePWM：增强型脉冲宽度调制器模块和寄存器

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

I2C：集成电路总线模块和寄存器

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。这个复用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退出特定低功耗模式。

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 位倒计时寄存器，此寄存器在计数器达到 0 时生成一个中断。这个计数器值以预分频的时钟周期递减。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 实时 OS 所预留，并且被连接至 CPU 的 INT14。如果 DSP/BIOS 未被使用，CPU 定时器 2 也可用于普通用途。CPU 定时器 1 为通用定时器并被连接至 CPU 的 INT13。CPU 定时器 0 也为通用计数器并被连接至 PIE 模块。

3.3.19 控制外设

ADP32F08 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/后缘边沿、被锁存的/逐周期机制的独立的/互补的 PWM 生成，可调节死区生成。一些 PWM 引脚支持 HRPWM 高分辨率占空比和周期特性。

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。这个外设也可被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。这个外设有一个安全装置定时器来检测电机停转和输入错误检测逻辑电路来识别 QEP 信号中的同步边沿转换。

ADC：ADC 模块是一个 12 位 16 个单端通道的模数转换器。它包含两个用于同步采样的采样保持单元。

3.3.20 串行端口外设

此器件支持下列的串行通信外设：

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 MCU 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 主机/从机操作支持。SPI 包含一个 16 级接收 FIFO 和一个 16 级发送 FIFO。

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 16 级接收 FIFO 和一个 16 级发送 FIFO。

I2C: 内部集成电路 (I2C) 模块提供一个 DSP 和其它器件 (符合飞利浦半导体内部 IC 总线 (I2C-bus) 规范版本 2.1 并由一个 I2C-bus 相连) 间的接口。通过这个 I2C 模块, 能够发送 8 位数据到 DSP, 或者从 DSP 接收 8 位数据。I2C 包含一个 16 级接收 FIFO 和一个 16 级发送 FIFO。

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 CAN 2.0B 兼容。

3.4 寄存器映射

此器件包含 3 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 3-5。

外设帧 1: 这些是映射到 32 位外设总线的外设。请参阅表 3-6。

外设帧 2: 这些是映射到 16 位外设总线的外设。请参阅表 3-7。

表 3-5 外设帧 0 寄存器^{(1) (2)}

名称	地址范围	大小 (x16)	访问类型 ⁽³⁾
器件仿真寄存器	0x00 0880-0x00 09FF	384	受 EALLOW 保护
闪存寄存器 ⁽⁴⁾	0x00 0A80-0x00 0ADF	96	受 EALLOW 保护; 受 CSM 保护
代码安全模块寄存器	0x00 0AE0-0x00 0AEF	16	受 EALLOW 保护
ADC 结果寄存器 (双映射)	0x00 0B00-0x00 0B0F	16	不受 EALLOW 保护
CPU-定时器 0/1/2 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

- (1) 在帧 0 中的寄存器支持 16 位和 32 位访问。
- (2) 内存空间的丢失段被保留并且不用被用在应用中。
- (3) 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。
- (4) 闪存寄存器也受到代码安全模块(CSM)的保护。

表 3-6 外设帧 1 寄存器^{(1) (2)}

名称	地址范围	大小 (x16)	访问类型
eCANA 寄存器	0x00 6000-0x0060FF	256	部分 eCAN 控制寄存器受 EALLOW 保护
eCANA 邮箱 RAM	0x00 6100-0x00 61FF	256	不受 EALLOW 保护
eCANB 寄存器	0x00 6200-0x00 62FF	256	部分 eCAN 控制寄存器受 EALLOW 保护
eCANB 邮箱 RAM	0x00 6300-0x00 63FF	256	不受 EALLOW 保护
ePWM1 寄存器	0x00 6800-0x00 683F	64	部分寄存器受 EALLOW 保护
ePWM2 寄存器	0x00 6840-0x00 687F	64	
ePWM3 寄存器	0x00 6880-0x00 68BF	64	

ePWM4 寄存器	0x00 68C0-0x00 68FF	64	不受 EALLOW 保护
ePWM5 寄存器	0x00 6900-0x00 693F	64	
ePWM6 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	0x00 6A40-0x00 6A5F	32	
eCAP4 寄存器	0x00 6A60-0x00 6A7F	32	
eQEP1 寄存器	0x00 6B00-0x00 6B3F	64	
eQEP2 寄存器	0x00 6B40-0x00 6B7F	64	
GPIO 控制寄存器	0x00 6F80-0x00 6FBF	128	受 EALLOW 保护
GPIO 数据寄存器	0x00 6FC0-0x00 6FDF	32	不受 EALLOW 保护
GPIO 中断和 LPM 选择寄存器	0x00 6FE0-0x00 6FFF	32	受 EALLOW 保护

- (1) eCAN 控制寄存器只能支持 32 位读取/写入操作。所有 32 位访问与偶数地址边界对齐。
 (2) 内存空间的丢失段被保留并且不能在应用中使用。

表 3-7 外设帧 2 寄存器^{(1) (2)}

名称	地址范围	大小 (x16)	访问类型
系统控制寄存器	0x00 7010-0x00 702F	32	受 EALLOW 保护
SPI-A 寄存器	0x00 7040-0x00 704F	16	不受 EALLOW 保护
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	
SPI-B 寄存器	0x00 7740-0x00 774F	16	
SCI-B 寄存器	0x00 7750-0x00 775F	16	
SPI-C 寄存器	0x00 7760-0x00 776F	16	
SPI-D 寄存器	0x00 7780-0x00 778F	16	
I2C 寄存器	0x00 7900-x00 792F	48	

- (1) 外设帧 2 只允许 16 位访问。所有 32 位访问被忽略。
 (2) 内存空间的丢失段被保留并且不能在应用中使用。

3.5 器件仿真寄存器

表 3-8 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	0x0882	1	部件 ID 寄存器
REVID	0x0883	1	类别 ID 寄存器
PROTSTART	0x0884	1	块保护起始地址
PROTRANGE	0x0885	1	块保护范围

3.6 中断

图 3-3 显示了不同的中断源是如何被复用的。

8 个 PIE 块中断被组合进一个 CPU 中断中。12 个 CPU 中断组，每个 8 个中断，相当于 96 个可能中断。表 3-9 显示了 ADP32F08 器件所用的中断。

TRAP#Vectormumber(矢量号) 指令将程序控制发送至与指定的矢量相对于用的中断处理列程。

TRAP#0 尝试传送程序控制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 被启用时，TRAP#1 至 TRAP#12 将传送程序控制到 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。

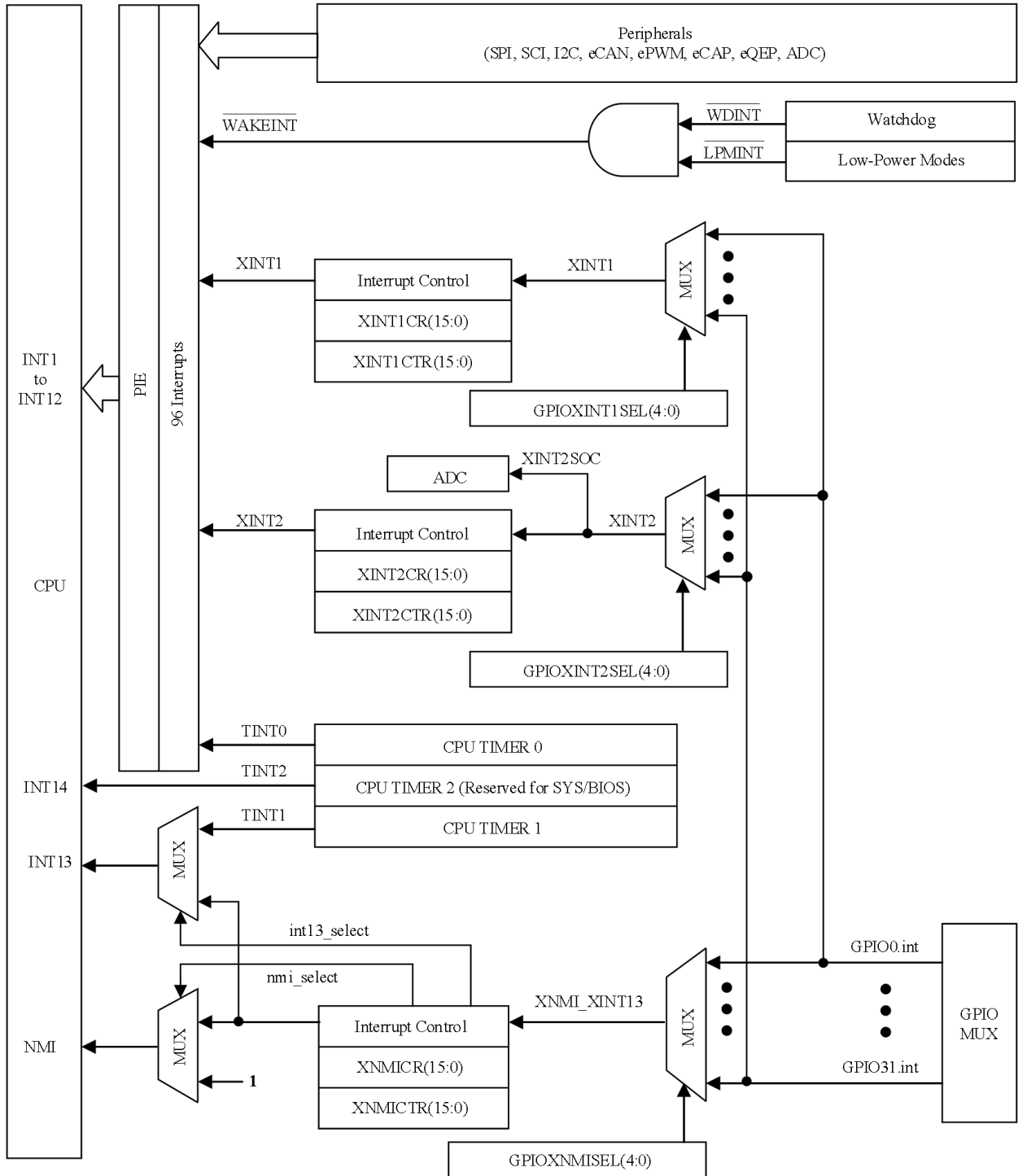


图 3-3 外部和 PIE 中断源

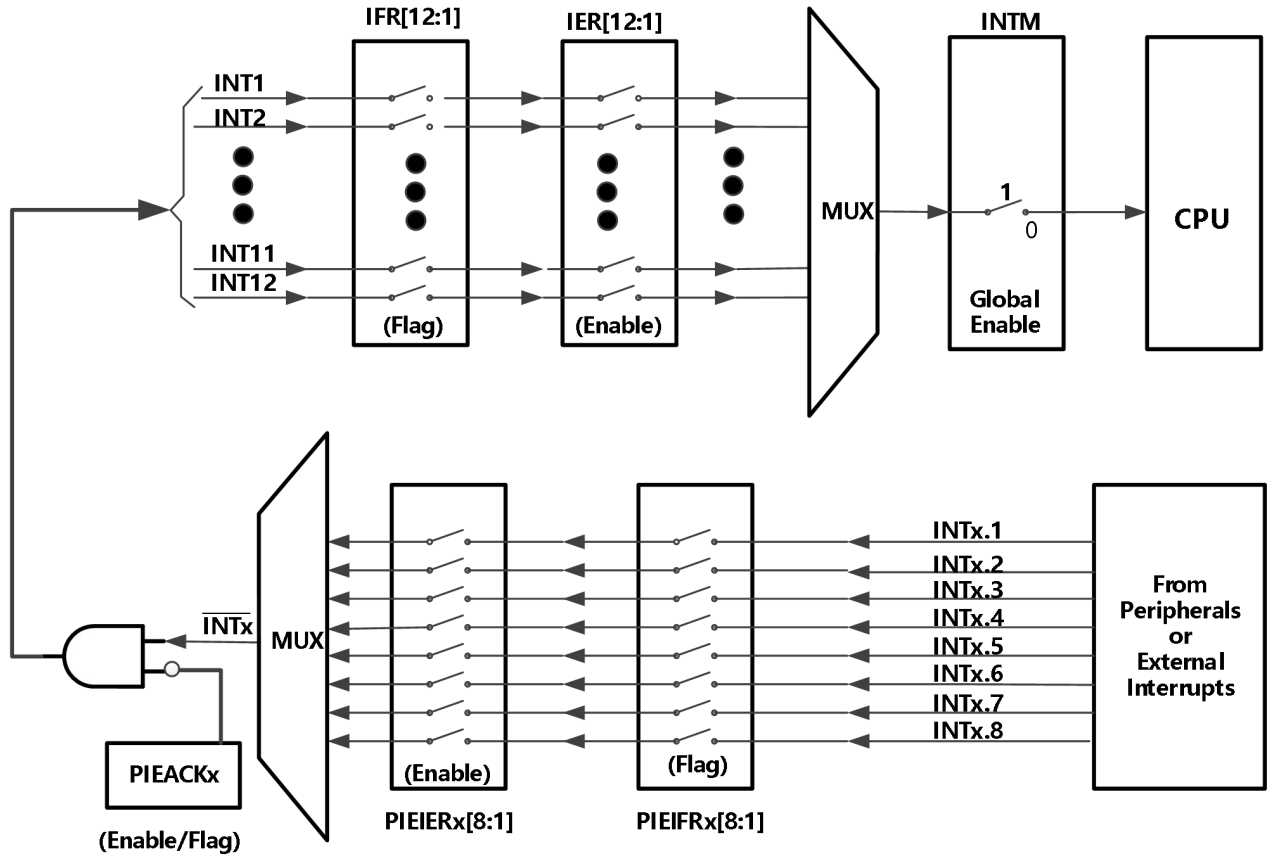


图 3-4 使用 PIE 块的中断复用

表 3-9 PIE 多路复用的外设中断矢量表⁽¹⁾

CPU 中 断	PIE 中断							
	INTx.8	INTx.7	INTx.6	INTx.5	INTx.4	INTx.3	INTx.2	INTx.1
INT1.y	WAKEINT(LPM/WD)	TINT0 (定时器 0)	ADCINT(ADC)	XINT2	XINT1	被保留	SEQ2INT(ADC)	SEQ1INT1(ADC)
INT2.y	被保留	被保留	EPWM6_TZINT(ePWM6)	EPWM5_TZINT(ePWM5)	EPWM4_TZINT(ePWM4)	EPWM3_TZINT(ePWM3)	EPWM2_TZINT(ePWM2)	EPWM1_TZINT(ePWM1)
INT3.y	被保留	被保留	EPWM6_INT(ePWM6)	EPWM5_INT(ePWM5)	EPWM4_INT(ePWM4)	EPWM3_INT(ePWM3)	EPWM2_INT(ePWM2)	EPWM1_INT(ePWM1)
INT4.y	被保留	被保留	被保留	被保留	ECAP4_INT(eCAP4)	ECAP3_INT(eCAP3)	ECAP2_INT(eCAP2)	ECAP1_INT(eCAP1)
INT5.y	被保留	被保留	被保留	被保留	被保留	被保留	EQEP2_INT(eQEP2)	EQEP1_INT(eQEP1)
INT6.y	SPITXINTD(SPI-D)	SPIRXINTD(SPI-D)	SPITXINTC(SPI-C)	SPIRXINTC(SPI-C)	SPITXINTB(SPI-B)	SPIRXINTB(SPI-B)	SPITXINTA(SPI-A)	SPIRXINTA(SPI-A)
INT7.y	被保留	被保留	被保留	被保留	被保留	被保留	被保留	被保留
INT8.y	被保留	被保留	被保留	被保留	被保留	被保留	I2CINT2A(I2C-A)	I2CINT1A(I2C-A)
INT9.y	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.y	被保留	被保留	被保留	被保留	被保留	被保留	被保留	被保留
INT11.y	被保留	被保留	被保留	被保留	被保留	被保留	被保留	被保留
INT12.y	被保留	被保留	被保留	被保留	被保留	被保留	被保留	被保留

(1)在 96 个可能的中断中，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没有一个被外设使用，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，在两个安全情况下，被保留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给组（例如，PIE 组 12）。

表 3-10 PIE 配置和控制寄存器

名称	地址	大小 (X 16)	说明 ⁽¹⁾
PICTRL	0x0000 0CE0	1	PIE,控制寄存器
PIEACK	0x0000 0CE1	1	PIE,确认寄存器
PIEIER1	0x0000 0CE2	1	PIE,INT1 组启用寄存器
PIEIFR1	0x0000 0CE3	1	PIE,INT1 组标志寄存器
PIEIER2	0x0000 0CE4	1	PIE,INT2 组启用寄存器
PIEIFR2	0x0000 0CE5	1	PIE,INT2 组标志寄存器
PIEIER3	0x0000 0CE6	1	PIE,INT3 组启用寄存器
PIEIFR3	0x0000 0CE7	1	PIE,INT3 组标志寄存器
PIEIER4	0x0000 0CE8	1	PIE,INT4 组启用寄存器
PIEIFR4	0x0000 0CE9	1	PIE,INT4 组标志寄存器
PIEIER5	0x0000 0CEA	1	PIE,INT5 组启用寄存器
PIEIFR5	0x0000 0CEB	1	PIE,INT5 组标志寄存器
PIEIER6	0x0000 0CEC	1	PIE,INT6 组启用寄存器
PIEIFR6	0x0000 0CED	1	PIE,INT6 组标志寄存器
PIEIER7	0x0000 0CEE	1	PIE,INT7 组启用寄存器
PIEIFR7	0x0000 0CEF	1	PIE,INT7 组标志寄存器
PIEIER8	0x0000 0CF0	1	PIE,INT8 组启用寄存器
PIEIFR8	0x0000 0CF1	1	PIE,INT8 组标志寄存器
PIEIER9	0x0000 0CF2	1	PIE,INT9 组启用寄存器
PIEIFR9	0x0000 0CF3	1	PIE,INT9 组标志寄存器
PIEIER10	0x0000 0CF4	1	PIE,INT10 组启用寄存器
PIEIFR10	0x0000 0CF5	1	PIE,INT10 组标志寄存器
PIEIER11	0x0000 0CF6	1	PIE,INT11 组启用寄存器
PIEIFR11	0x0000 0CF7	1	PIE,INT11 组标志寄存器
PIEIER12	0x0000 0CF8	1	PIE,INT12 组启用寄存器
PIEIFR12	0x0000 0CF9	1	PIE,INT12 组标志寄存器
被保留	0x0000 0CFA-0x0000 0CFF	6	被保留

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

表 3-11 外部中断寄存器

名称	地址	大小 (X 16)	说明
XINT1CR	0x7070	1	XINT1 控制寄存器
XINT2CR	0x7071	1	XINT2 控制寄存器
被保留	0x7072-0x7076	5	被保留
XNMICR	0x7077	1	XNMI 控制寄存器
XINT1CTR	0x7078	1	XINT1 计数器寄存器
XINT2CTR	0x7079	1	XINT2 计数器寄存器
被保留	0x707A-0x707E	5	被保留
XNMICTR	0x707F	1	XNMI 计数器寄存器

(1) 每个外部中断可被启用/禁用或者使用正边沿以及负边沿来限定中断。

3.7 系统控制

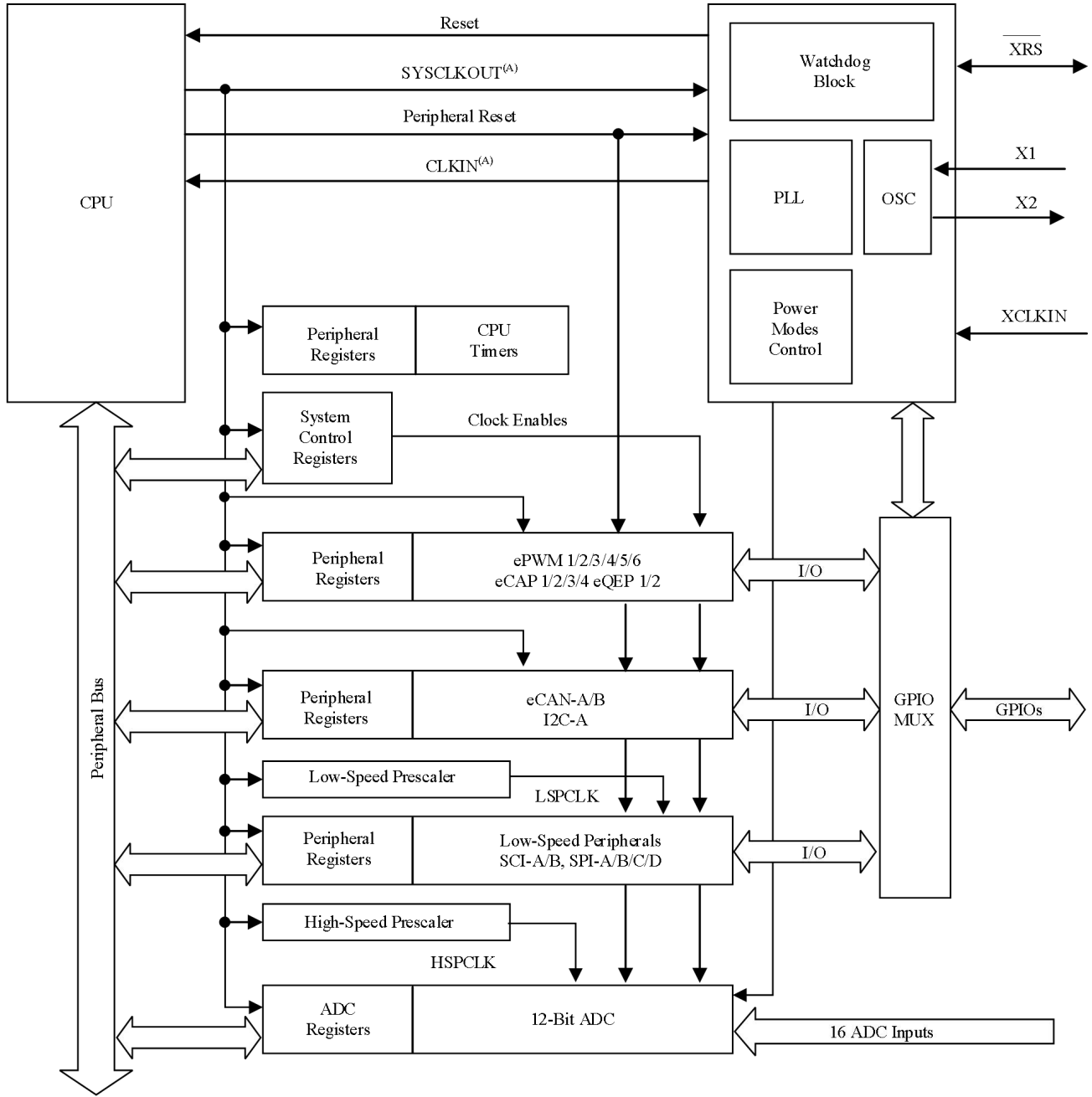
这一部分对振荡器和时钟机制、安全装置功能以及低功耗模式进行了说明。

表 3-12 PLL、时钟、安全装置和低功耗模式寄存器

名称	地址	大小 (X 16)	说明 ⁽¹⁾
XCLK	0x00 7010	1	XCLKOUT 控制、X1 和 XCLKIN 状态寄存器
PLLSTS	0x00 7011	1	PLL 状态寄存器
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
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
WDKEY	0x00 7025	1	看门狗复位密钥寄存器
WDCR	0x00 7029	1	看门狗控制寄存器

(1) 这个表中的所有寄存器是受 EALLOW 保护的。

图 3-5 显示了讨论的多种时钟域。



A. CLKIN 是到 CPU 的时钟。它作为 SYSCLKOUT 从 CPU 传出（也就是说，CLKIN 与 SYSCLKOUT 频率相同）。

图 3-5 时钟和复位域

3.7.1 OSC 和 PLL 模块

图 3-6 显示了 ADP32F08 上的 OSC 和 PLL 模块。

名称地址大小(x 16) 说明:

表 3-13 PLL、计时、安全装置和低功率模式寄存器(1)

名称	地址	大小(x16)	说明
XCLK	0x7010	1	XCLKOUT引脚控制、X1和 XCLKIN状态寄存器
PLLSTS	0x7011	1	PLL状态寄存器
被保留	0x7012-0x7019	8	被保留
HISPCP	0x701A	1	高速外设时钟预分频器寄存器 (用于 HSPCLK)
LOSPCP	0x701B	1	低速外设时钟预分频器寄存器 (用于 LSPCLK)
PCLKCR0	0x701C	1	外设时钟控制寄存器0
PCLKCR1	0x701D	1	外设时钟控制寄存器1
LPMCR0	0x701E	1	低功耗模式控制寄存器0
被保留	0x701F-0x7020	1	被保留
PLLCR	0x7021	1	PLL控制寄存器
SCSR	0x7022	1	系统控制与状态寄存器
WDCNTR	0x7023	1	安全装置计数器寄存器
被保留	0x7024	1	被保留
WDKEY	0x7025	1	安全装置复位密钥寄存器
被保留	0x7026-0x7028	3	被保留
WDCR	0x7029	1	安全装置控制寄存器
被保留	0x702A-0x702F	6	被保留

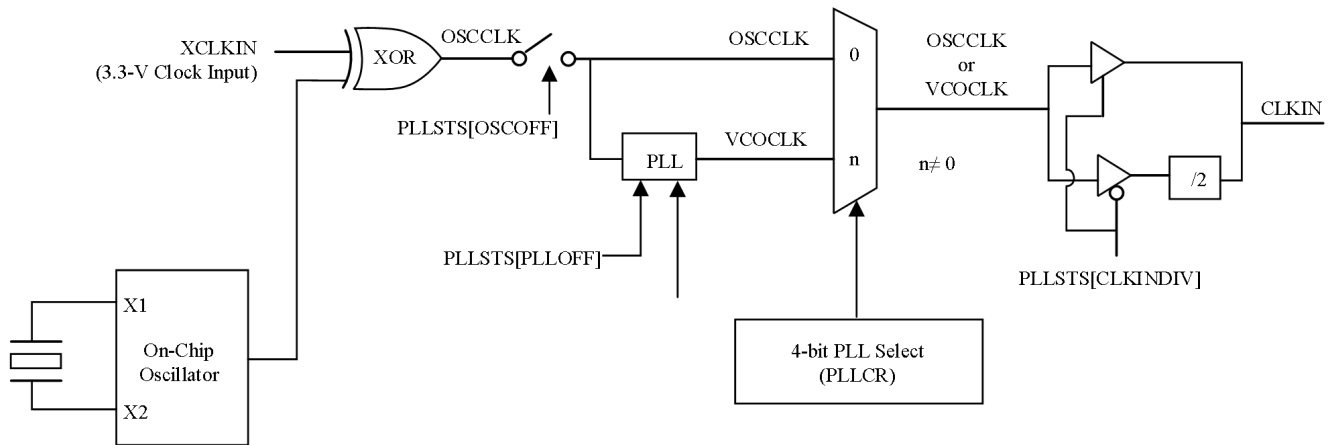


图 3-6 OSC 和 PLL 模块

片载振荡器电路启用一个使用 X1 和 X2 引脚连接至 ADP32F08 器件的晶振。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. 一个 3.3V 外部振荡器可被直接接至 XCLKIN 引脚。X2 引脚应保持未连接，而 X1 引脚接至低电平。这个情况下的逻辑高电平不用超过 V_{DDIO} 。
2. 一个 1.8V 外部振荡器也可被直接连接至 X1 引脚。X2 引脚应保持未连接，而 XCLKIN 引脚接至低电平。这个情况下的逻辑高电平不用超过 V_{DD} 。

图 3-7 至图 3-9 显示了这三个可能的输入时钟配置

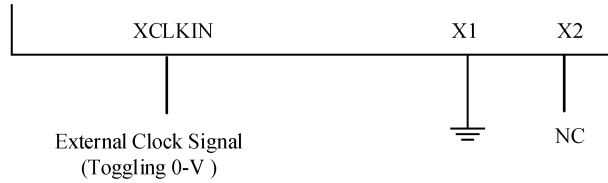


图 3-7 使用一个 3.3V 外部振荡器

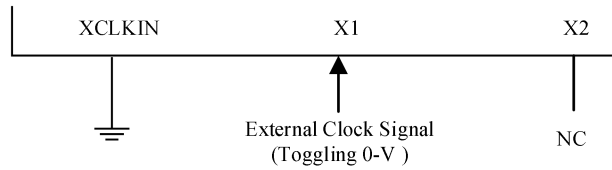


图 3-8 使用一个 1.8V 外部振荡器

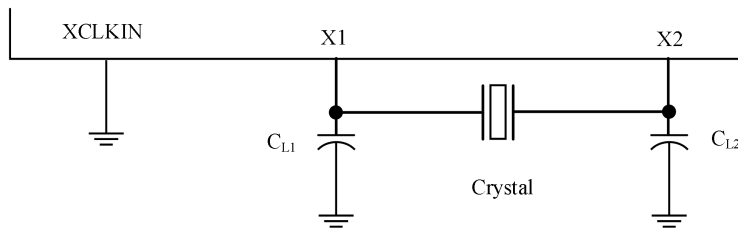


图 3-9 使用内部振荡器

3.7.1.1 外部基准振荡器时钟选项

20MHz 外部石英晶振的典型技术规范如下:

- 基本模式、并联谐振
- C_L (负载电容) = 12pF
- $C_{L1}=C_{L2}=24\text{pF}$
- $C_{\text{并联}}=6\text{pF}$
- ESR 范围 = 30 至 60 Ω

建议客户让谐振器/晶体供应商提供他们的器件与 DSP 芯片一起工作的信息。谐振器/晶体供应商具有调谐谐振电路的设备和专业技术。销售商也可建议客户考虑适当的谐振组件值, 这个值将在整个运行范围内产生合适的启动和稳定性。

3.7.1.2 基于 PLL 的时钟模块

ADP32F08 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, 以及对低功耗模式进入的控制。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, 安全装置模块应该被禁用。在 PLL 模式稳定后, 它可被重新启用 (如果需要的话), 重新启用的时间为 131072 个 OSCCLK 周期。

表 3-14 PLL 寄存器位设置

PLLCR[DIV] ⁽¹⁾	SYSCLKOUT(CLKIN) ⁽²⁾
0000 (PLL旁路)	OSCCLK/n
0001	(OSCCLK*1)/n
0010	(OSCCLK*2)/n
0011	(OSCCLK*3)/n
0100	(OSCCLK*4)/n
0101	(OSCCLK*5)/n
0110	(OSCCLK*6)/n
0111	(OSCCLK*7)/n
1000	(OSCCLK*8)/n
1001	(OSCCLK*9)/n
1010	(OSCCLK*10)/n
1011-1111	被保留

此寄存器是受 EALLOW 保护的。

- (1) CLKIN 是到 CPU 的输入时钟。SYSCLKOUT 为来自 CPU 的输出时钟。SYSCLKOUT 的频率与 CLKIN 一样。如果 CLKINDIV = 0, n = 2; 如果 CLKINDIV = 1, n = 1。

注

在时钟被馈入输入内核前, PLLSTS[CLKINDIV] 启用或者旁通此二分频块。这个位在写入 PLLCR 前必须为 0, 在 PLLSTS[PLLLOCKS] = 1 后必须被配置。

基于 PLL 的时钟模块提供两种操作模式:

- 晶振操作-这个模式允许使用一个外部晶振/谐振器来提供到器件的时基。
- 外部时钟源操作-这个模式允许内部振荡器被旁通。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 3-15 PLL 配置模式

PLL 模式	说明	PLLSTS[CLKINDIV]	SYSCCLKOUT (CLKIN)
PLL 关闭	由在 PLLSTS 寄存器中设置 PLLOFF 位的用户调用。在此模式中，PLL 块被禁用。这对降低系统噪声和低功率操作非常有用。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 (CLKIN) 直接源自 X1/X2, X1 或者 XCLKIN 上的输入时钟。	0	OSCCLK/2
		1	OSCCLK
PLL 旁路	PLL 旁路是加电或外部复位 (XRS) 时的默认 PLL 配置。当 PLLCR 寄存器设置为 0x0000 时或在修改 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	OSCCLK/2
		1	OSCCLK
PLL 启用	通过将非零值 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	OSCCLK*n/2

3.7.1.3 时钟丢失检测

在 PLL 启用或者 PLL 旁通模式中，如果输入时钟 OSCCLK 被去除或者缺失，PLL 仍将产生一个跛行模式时钟。这个跛行模式时钟持续为 CPU 和典型频率为 1-5MHz 的外设计时。额定情况下，跛行模式加电时并不运行，只在输入时钟已经首次出现时才运行。在 PLL 旁通模式中，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。

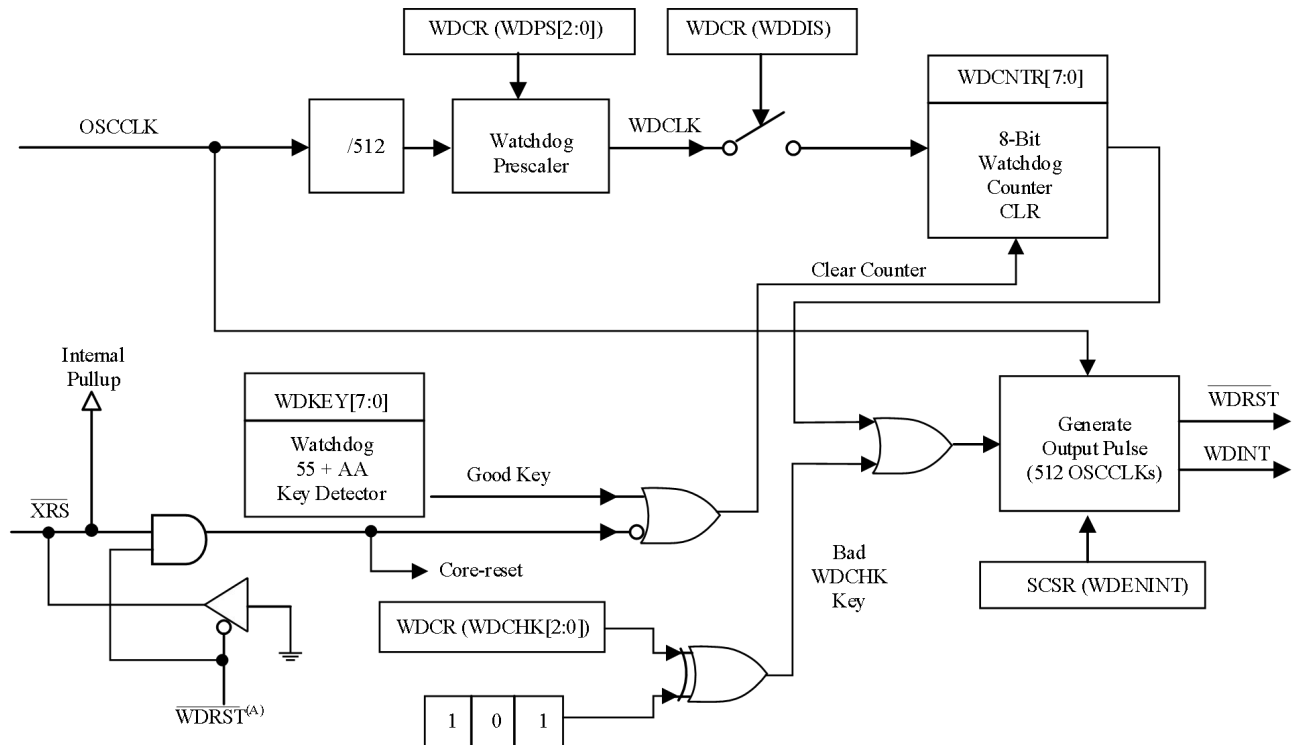
通常情况下，当输入时钟出现时，安全装置计数器减量来启动一个安全装置复位或者 WDINT 中断。然而，当外部输入时钟发生故障时，安全装置计数器停止减量（也就是说，安全装置计数器不会随着跛行模式时钟而改变）。除此之外，器件将被复位并且设置“丢失的时钟状态” (MCLKSTS) 位。这些条件可被应用固件用来检测输入时钟故障并启动系统所需的关闭程序。

注

正确的 CPU 运行频率非常重要，应用中应该执行一个机制，通过这个机制，只要输入时钟出现故障，DSP 就被保持在复位状态。

3.7.2 看门狗模块

看门狗模块只要 8 位递增计数器达到了它的最大值，这个模块就生成一个输出脉冲，512 个振荡器时钟宽度 (OSCCLK)。为了防止这一情况，用户必须禁用此计数器或者软件必须定期地向复位此看门狗模块计数器的安全装置密钥寄存器写入一个 0x55+0xAA 序列。图 3-10 显示了看门狗模块内的各种功能块。



A. WDRST信号在 512 个 OSCCLK 周期内被驱动为低电平。

图 3-10 看门狗模块

WDINT信号使看门狗可用于唤醒 IDLE/STANDY 模式。

在 STANDBY 模式中，器件上的所有外设关闭。看门狗是唯一继续工作的外设。看门狗模块将关闭 OSCCLK。WDINT信号被发送到 LPM 块以便将器件从 STANDBY 模式唤醒（如已启用）。更多细节，请见 3.8 节低功耗模式模块。

在 IDLE 模式中，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，不能使用此功能，因为振荡器（和 PLL）关闭，所以安全狗也关闭。

3.8 低功耗模式模块

表 3-16 总结了多种模式。

表 3-16 低功耗模式

模式	LPM(1:0)	OSCCLK	CLKIN	SYSCLKOUT	退出 ⁽¹⁾
IDLE	0,0	打开	打开	打开 ⁽²⁾	\overline{XRS} ,看门狗中断, 任何启动中断, XNMI
STANDBY	0,1	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	1,X	关闭 (振荡器和 PLL 关闭, 看门狗模块 不工作)	关闭	关闭	\overline{XRS} ,GPIO 端口 A 信号, XNMI, 调试器

(1) “退出列”列出了哪些信号或在哪些情况下会退出低功耗模式。任意信号中的一个低电平信号, 都将退出低功耗状态。此信号低电平保持时间需要足够长, 以便器件能够识别中断。否则, 将不会退出 IDLE 模式, 而器件将返回到所示低功耗模式。

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。LPM 块在这个模式期间, 在 LPMCR0(LPM)位被设定为 0, 0 时, LPM 块不执行任何任务。

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。所选的信号在经过 LPMCR0 寄存器中设定 OSCCLK 的数量时钟后, 在合格时间内被取值。

HALT 模式 \overline{XRS} 和任何一个 GPIO 端口 A 信号 (GPIO[31:0])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 IDLE 被指令执行时, 它们将保持在代码指定的状态中。

4 外设

4.1 32 位 CPU 定时器 0/1/2

ADP32F08 上有 3 个 32 位 CPU 定时器 (CPU-TIMER0/1/2)。

用户可以使用 CPU 定时器 0 和 CPU 定时器 1。定时器 2 为 DSP/BIOS 预留。这些定时器与 ePWM 模块中的定时器不同。

注

如果应用没有使用 DSP/BIOS, 那么用户可以使用 CPU 定时器 2.

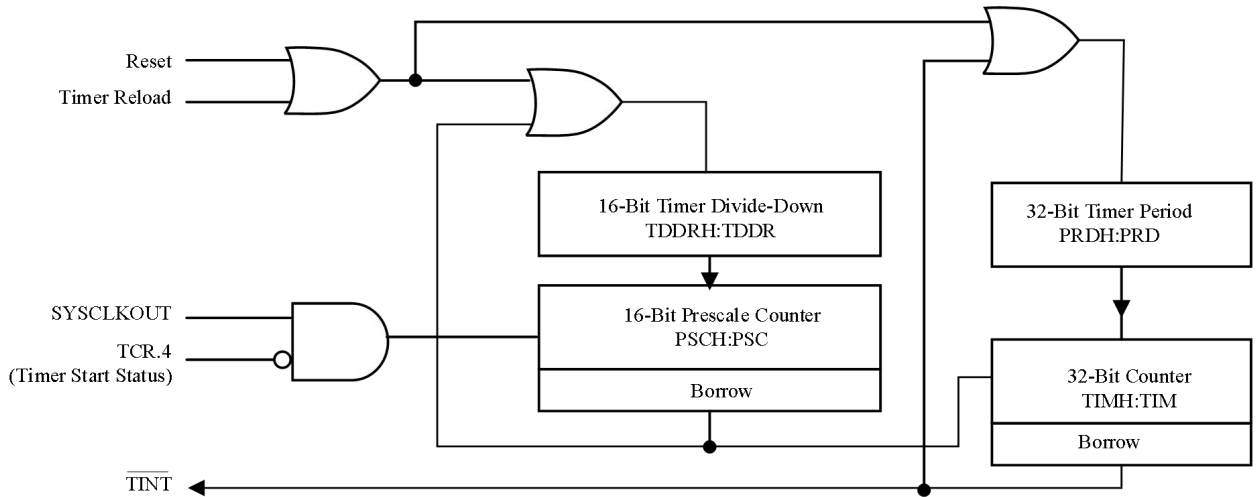


图 4-1 CPU 定时器

定时器中断信号的连接如图 4-2 所示。

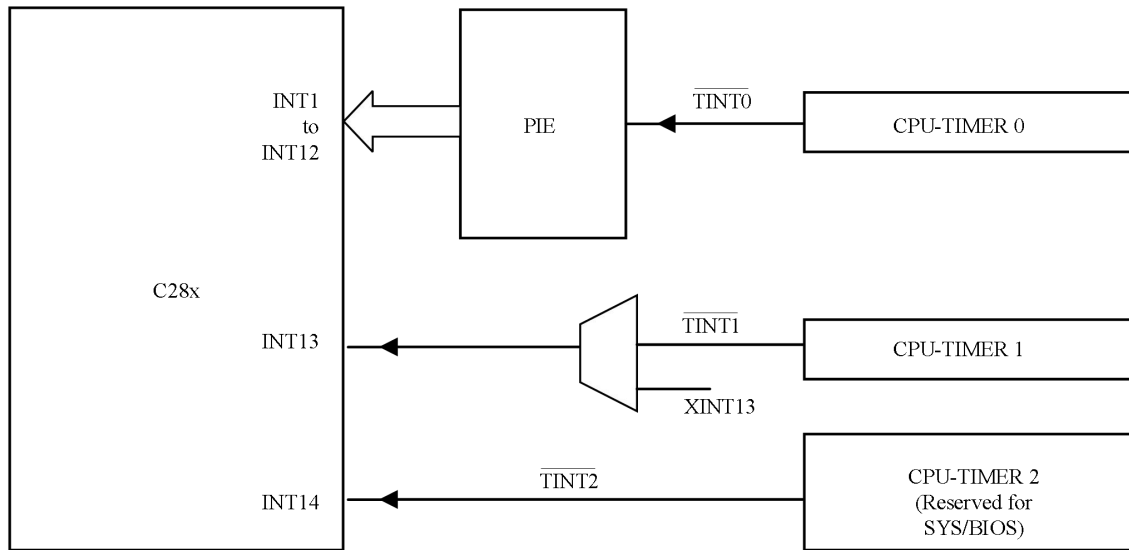


图 4-2 CPU 定时器中断信号和输出信号

定时器的通常操作如下：32 位计数器寄存器 “TIMH:TIM” 被装入周期寄存器 PRDH:PRD 中的值。计数器寄存器按 SYSCLKOUT 速率递减。当计数器到达 0 时，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

名称	地址	大小(x16)	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
被保留	0x0C05	1	被保留
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
被保留	0x0C0D	1	被保留
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
被保留	0x0C15	1	被保留
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

4.2 增强型模数转换器 (ADC) 模块

图 4-3 显示了 ADC 模块的一个简化功能方框图。这个 ADC 模块由一个带有内置采样保持 (S/H) 电路的 12 位 ADC 组成。ADC 模块的功能包括：

- 具有内置 S/H 的 12 位 ADC 内核
- 模拟输入：0.0V 至 3.0V (高于 3.0V 的电压产生满刻度转换结果)。

- 快速转换率：在 12.5MHzADC 时钟 6.25MSPS 上时高达 160ns
- 16 通道，复用输入
- 自动定序功能在单次会话中可提供多达 16 次“自动转换”。每次转换可被设定为选择 16 个输入信道中的任何一个。
- 序列发生器可运行作为 2 个独立的 8 通道序列发生器，或作为 1 个较大的 16 通道序列发生器使用（即 2 个级联的 8 通道序列发生器）。
- 用于存储转换值的 16 个结果寄存器（可单独寻址）

– 输入模拟电压的数值源自：

$$\text{Digital Value} = 0 \quad \text{when input} \leq 0V$$

$$\text{Digital Value} = 4096 \times \frac{\text{Input Analog Voltage} - \text{ADCL0}}{3} \quad \text{when } 0V < \text{input} < 3V$$

$$\text{Digital Value} = 4095 \quad \text{when input} \geq 3V$$

- A. 所有分数值均为截断值。
- 作为转换开始序列 (SOC) 源的多个触发器
 - S/W - 软件立即启动
 - ePWMM 转换开始
 - XINT2 ADC 转换开始
 - 灵活的中断控制允许每个序列结束 (EOS) 或每个其它 EOS 上的中断请求。
 - 序列发生器可运行于“启/停”模式，从而实现多个“时序触发器”同步转换。
 - SOCA 和 SOCB 触发器可独立运行在双序列发生器模式中。
 - 采样保持 (S/H) 采集时间窗口具有独立的预分频控制。

ADP32F08 增强了 ADC 模块以提供 ePWM 外设的灵活接口。ADC 接口被建立在一个快速，12 位 ADC 模块上，此模块在 12.5MHzADC 时钟上的快速转换率高达 160ns。ADC 模块有一个 16 通道序列发生器，此发生器可配置为两个独立的 8 通道序列发生器。这两个独立的 8 通道序列发生器可被级联成一个 16 通道序列发生器。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 4-7 显示了 ADC 模块的方框图。

这两个 8 信道模块可对一系列转换自动定序，每个模块可以选择通过模拟 MUX 从可用的 8 个通道中选取任何一个。在级联模式中，自动序列发生器将作为一个单个 16 信道序列发生器使用。一旦每个序列发生器上的转换完成，所选的信道值将存储在其各自的 RESULT 寄存器中。自动定序功能使得系统

可多次转换同一信道，从而使用户可执行过采样算法。这种过采样算法可提供比传统的单一采样转换结果更高的分辨率。

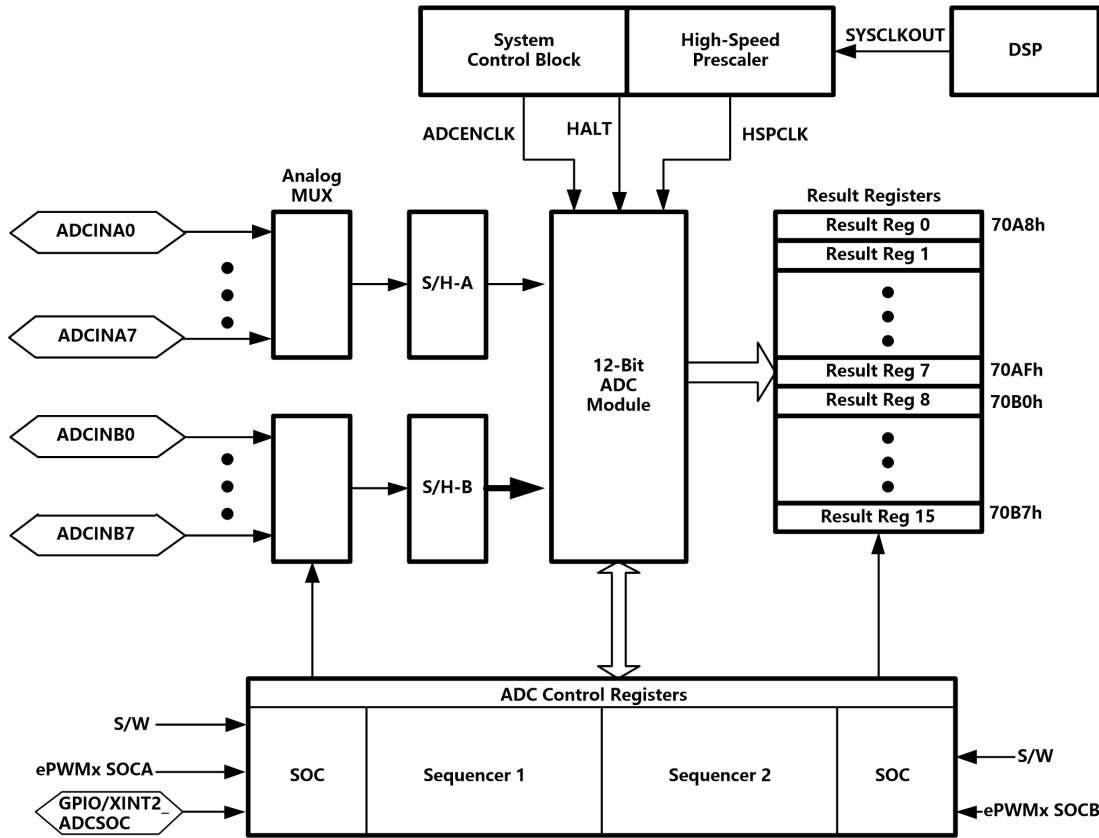


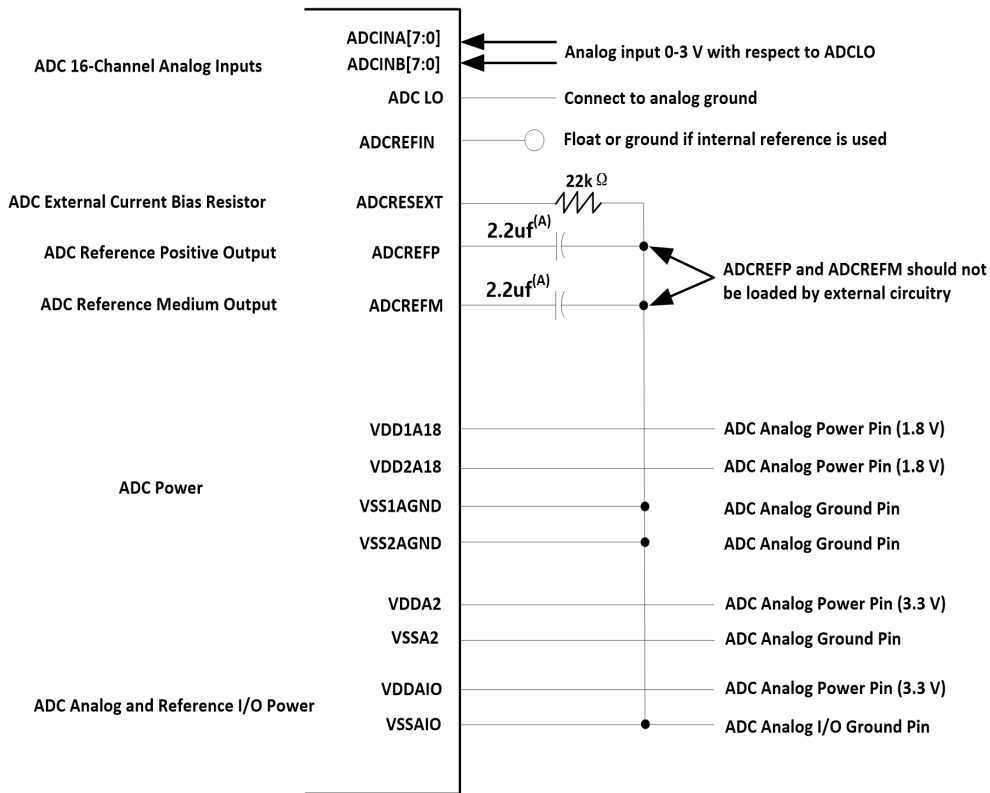
图 4-3 ADC 模块方框图

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进 ADC 输入的开关噪声。而且，适当的隔离技术必须被用来将数字电源从 ADC 模块电源引脚 (V_{DD1A18} , V_{DD2A18} , V_{DDA2} , V_{DDA10}) 上隔离。图 4-4 和图 4-5 显示了对于 F08 器件的 ADC 引脚连接。

注

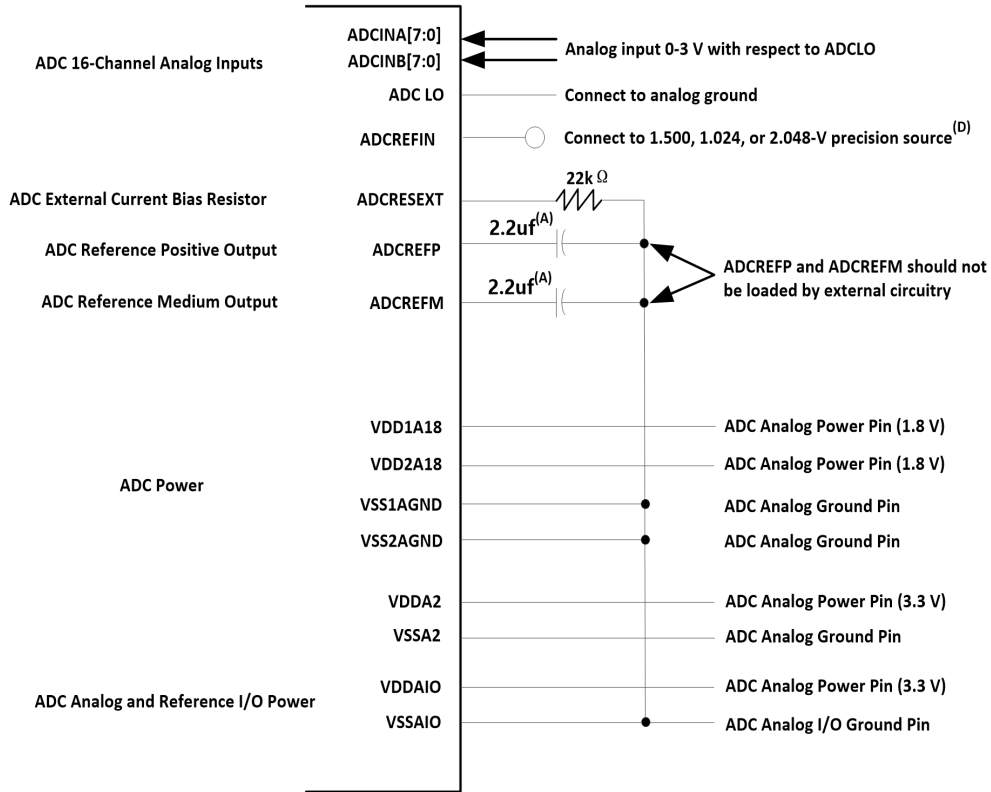
1. 在 SYSCLKOUT 速率上对 ADCC 寄存器进行访问。ADC 模块的内部时序由高速外设时钟 (HSPCLK) 控制。
2. 基于 ADCENCLK 和 HALT 信号的 ADC 模块的运行方式如下：
 - *ADCENCLK*: 复位时，这个信号将为低电平。虽然复位为低电平有效 (XRS)，到寄存器的时钟仍将运行。有必要确保所有寄存器和模式进入它们的复位状态。然而，模拟模块将处于一个低功耗非激活状态。一旦复位变成高电平，那么到寄存器的时钟将被禁用。当用户将 ADCENCLK 信号设定为高电平时，那么到寄存器的时钟将被启用并且模拟模块将被启用。在 ADC 稳定并且可被使用之前，将有一个特定的时间延迟 (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 4-4 显示了针对内部基准的 ADC 引脚偏置而图 4-5 显示了针对外部基准的 ADC 引脚偏置。



- A. TAIYO YUDEN LMK212BJ225MG-T 或等效器件
- B. 建议在所有电源引脚上使用外部去耦合电容器。
- C. 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。

图 4-4 带有内部基准的 ADC 引脚连接



- A. TAIYO YUDEN LMK212BJ225MG-T 或者等效器件
- B. 建议在所有电源引脚上使用外部去耦合电容器。
- C. 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。
- 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。

图 4-5 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。

4.2.1 如果 ADC 未被使用，ADC 连接

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 未在实际应用中使用，应该如何连接 ADC 引脚：

- V_{DD1A18}/V_{DD2A18} - 连接至 V_{DD}
- V_{DDA2} , V_{DDAIO} - 连接至 V_{DDIO}
- $V_{SS1AGND}/V_{SS2AGND}$, V_{SSA2} , V_{SSAIO} - 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 20k Ω 电阻器（非常松散的耐受）至 V_{SS} 。
- ADCIN $_n$, ADCIN $_Bn$ - 连接至 V_{SS}

当 ADC 未被使用时，为了达到节能的目的，请确保 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接地 ($V_{SS1AGND}/V_{SS2AGND}$)。

4.2.2 ADC 寄存器

表 4-2 中所列寄存器对 ADC 操作进行配置、控制、和监视。

表 4-2ADC 寄存器 (1)

名称	地址 ⁽¹⁾	地址 ⁽²⁾	大小(x16)	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最大转换信道数寄存器
ADCCHSELSEQ1	0x7103		1	ADC信道选择排序控制寄存器 1
ADCCHSELSEQ2	0x7104		1	ADC信道选择排序控制寄存器 2
ADCCHSELSEQ3	0x7105		1	ADC信道选择排序控制寄存器 3
ADCCHSELSEQ4	0x7106		1	ADC信道选择排序控制寄存器 4
ADCASEQSR	0x7107		1	ADC自动排序状态寄存器
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
被保留	0x711A- 0x711B		2	被保留
ADCREFSEL	0x711C		1	ADC基准选择寄存器
ADCOFFTRIM	0x711D		1	ADC偏移调整寄存器
被保留	0x711E 0x711F		2	被保留

(1) 本列中的寄存器为外设帧 2 寄存器。

(2) ADC 结果寄存器在ADP32F08 DSP中进行了双映射。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)中的位置为 0等待状态, 且为右对齐。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。

4.3 串行通信接口 (SCI) 模块 (SCI-A, SCI-B)

ADP32F08器件包括2个串行通信接口 (SCI)模块。SCI 模块支持 CPU 与其它异步使用标准非归零 (NRZ) 码格式外设之间的数字通信。SCI 接收器和发射器是双缓冲的, 并且它们中的每一个有其自身独立的启用和中断位。两个器件都可独立或者同时地运行在全双工模式。为了确保数据完整性, SCI 在中断检测、奇偶校验、超载、和组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 个不同的速度。

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:
- $$\text{Baudrate} = \frac{LSPCLK}{(BRR+1)*8} \text{ when } BRR \neq 0$$
- $$\text{Baudrate} = \frac{LSPCLK}{16} \text{ when } BRR = 0$$
- 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、超载、组帧、和中断检测
 - 两个唤醒多处理器模式: 空闲线路和地址位
 - 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射器和接收器操作。
 - 发射器: TXRDY标志 (发射器缓冲寄存器已经准备好接收另外字符) 和 TXEMPTY (TX空) 标志 (发射器移位寄存器已空)
 - 接收器: RXRDY标志 (接收器缓冲寄存器已经准备好接收另外的字符), BRKDT标志 (发生了中断条件), 和 RXERROR (错误) 标志 (监控四个中断条件)
 - 用于发射器和接收器中断的独立启用位 (除了 BRKDT)
 - $\text{Max bit rate} = \frac{100\text{MHz}}{16} = 6.25 \times 10^6 \text{b/s}$ (for 100-MHz 器件)

- Max bit rate = $\frac{60MHz}{16} = 3.75 \times 10^6 b/s$ (for 60-MHz 器件)
- NRZ (非归零) 格式
- 在开始地址为 7050h 控制寄存器帧中, 有 10 个 SCI 模块控制寄存器。

注

这个模块中的所有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。对高字节的写入没有效果。

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 4-3 至表 4-4 中列出的寄存器配置和控制。

表 4-3 SCI-A 寄存器⁽¹⁾

名称	地址	大小 (x16)	说明
SCICCRA	0x7050	1	SCI-A 通信控制寄存器
SCICTL1A	0x7051	1	SCI-A 控制寄存器 1
SCIHBAUDA	0x7052	1	SCI-A 波特率寄存器, 高位
SCILBAUDA	0x7053	1	SCI-A 波特率寄存器, 低位
SCICTL2A	0x7054	1	SCI-A 控制寄存器 2
SCIRXSTA	0x7055	1	SCI-A 接收状态寄存器
SCIRXEM UA	0x7056	1	SCI-A 接收仿真数据缓冲寄存器
SCIRXBUFA	0x7057	1	SCI-A 接收数据缓冲寄存器
SCITXBUFA	0x7059	1	SCI-A 发送数据缓冲寄存器
SCIFFTXA ⁽²⁾	0x705A	1	SCI-A FIFO 发送寄存器
SCIFFRXA ⁽²⁾	0x705B	1	SCI-A FIFO 接收寄存器
SCIFFCTA ⁽²⁾	0x705C	1	SCI-A FIFO 控制寄存器
SCIPRIA	0x705F	1	SCI-A 优先级控制寄存器

(1) 上表的寄存器被映射到外设帧 2 空间。这空间只允许 16 位访问。32 位访问会产生未定义的后果。

(2) 这些寄存器是用于 FIFO 模式的全新寄存器。

表 4-4 SCI-B 寄存器⁽¹⁾⁽²⁾

名称	地址	大小 (x16)	说明
SCICCRB	0x7750	1	SCI-B通信控制寄存器
SCICTL1B	0x7751	1	SCI-B控制寄存器1
SCIHBAU DB	0x7752	1	SCI-B波特率寄存器, 高位
SCILBAUD B	0x7753	1	SCI-B波特率寄存器, 低位
SCICTL2B	0x7754	1	SCI-B控制寄存器2
SCIRXSTB	0x7755	1	SCI-B接收状态寄存器
SCIRXEM UB	0x7756	1	SCI-B接收仿真数据缓冲寄存器
SCIRXBUF B	0x7757	1	SCI-B接收数据缓冲寄存器
SCITXBUF B	0x7759	1	SCI-B发送数据缓冲寄存器
SCIFFTXB ⁽²⁾	0x775A	1	SCI-BFIFO发送寄存器
SCIFFRXB ⁽²⁾	0x775B	1	SCI-BFIFO接收寄存器
SCIFFCTB ⁽²⁾	0x775C	1	SCI-BFIFO控制寄存器
SCIPRIB	0x775F	1	SCI-B优先级控制寄存器

(1) 上表中的寄存器被映射到外设总线 16空间。这空间只允许 16位访问。32位访问会产生未定义的后果。

(2) 这些寄存器是用于 FIFO模式的全新寄存器。

图 4-6 为 SCI 模块方框图。

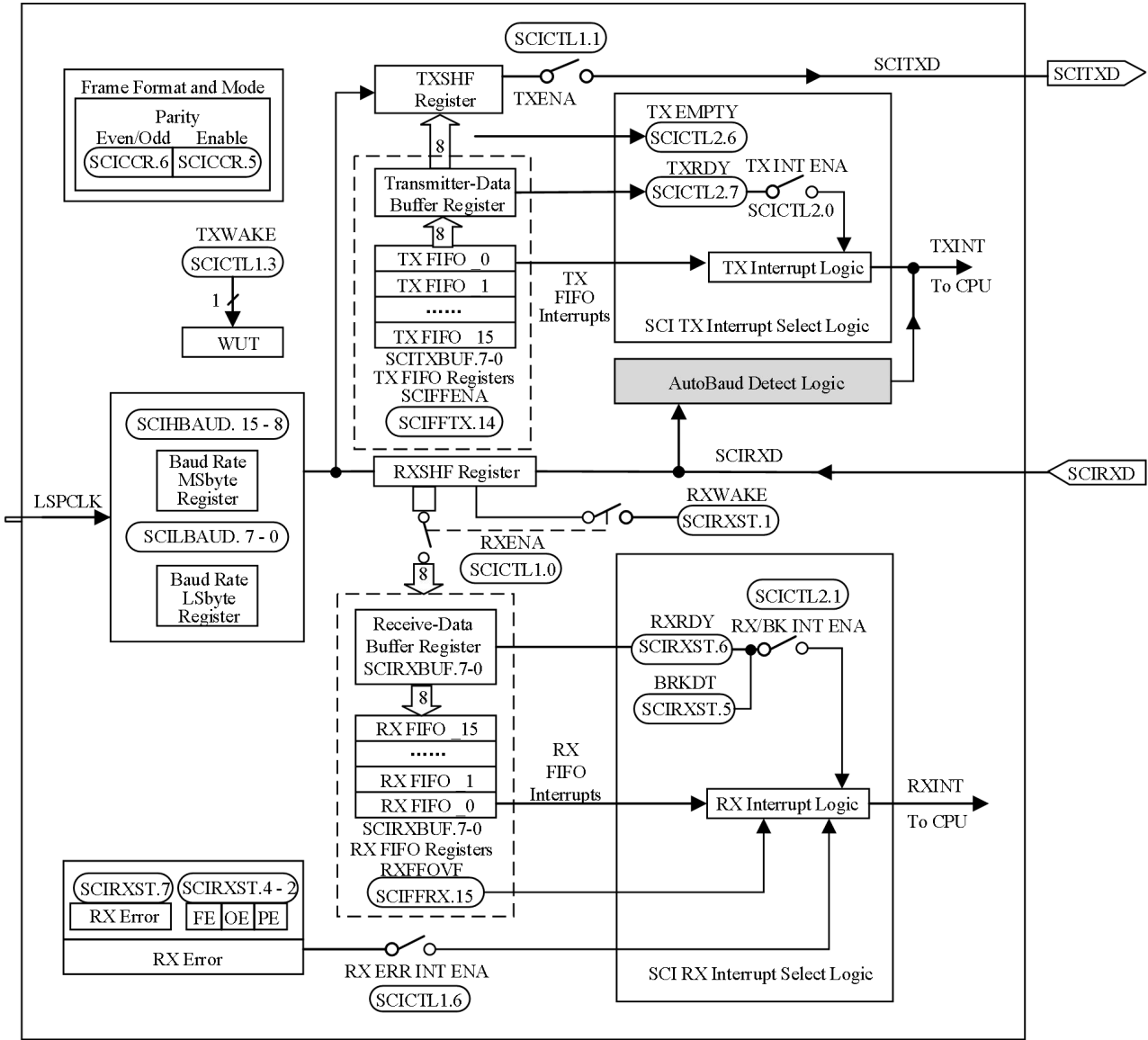


图 4-6 串行通信接口 (SCI)模块方框图

4.4 串行外设接口 (SPI)模块(SPI-A, SPI-B, SPI-C, SPI-D)

ADP32F08 包括四引脚串行外设接口 (SPI)模块。提供多达四个 SPI 模块 (SPI-A, SPI-B, SPI-C, 和 SPI-D)。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。通常情况下，SPI 用于 DSP 控制器和外部外设或者其他处理器之间的通信。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 的主机/从机操作支持。

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从器件发送启用引脚
 - 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: 主机和从机
- 波特率: 125个不同的可编辑速率。
- Baud rate = $\frac{LSPCLK}{(SPIBRR+1)}$ when SPIBRR=3 to 127
- Baud rate = $\frac{LSPCLK}{4}$ when SPIBRR=0,1,2
- 数据字长度: 一到十六数据位
- 包括四个计时机制 (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信号的下降沿上传送数据, 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射器和接收器运行。
- 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。对高字节的写入没有效果。

增强型特性：

- 16级发送/接收 FIFO
- 经延迟的发射控制

SPI 端口运行由表 4-5 至表 4-8 中列出的寄存器配置和控制。

表 4-5 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
SPIRXEM U	0x7046	1	SPI-A接收仿真缓冲寄存器
SPIRXBUF	0x7047	1	SPI-A串行输入缓冲寄存器
SPITXBUF	0x7048	1	SPI-A串行输出缓冲寄存器
SPIDAT	0x7049	1	SPI-A串行数据寄存器
SPIFFTX	0x704A	1	SPI-AFIFO发送寄存器
SPIFFRX	0x704B	1	SPI-AFIFO接收寄存器
SPIFFCT	0x704C	1	SPI-AFIFO控制寄存器
SPIPRI	0x704F	1	SPI-A优先级控制寄存器

(1)这个表中寄存器被映射到外设帧 2。这空间只允许 16 位访问。32 位访问会生成未定义的后果。

表 4-6 SPI-B 寄存器

名称	地址	大小 (x16)	说明 (1)
SPICCR	0x7740	1	SPI-B配置控制寄存器
SPICTL	0x7741	1	SPI-B运行控制寄存器
SPISTS	0x7742	1	SPI-B状态寄存器
SPIBRR	0x7744	1	SPI-B波特率寄存器
SPIRXEM U	0x7746	1	SPI-B接收仿真缓冲寄存器
SPIRXBUF	0x7747	1	SPI-B串行输入缓冲寄存器
SPITXBUF	0x7748	1	SPI-B串行输出缓冲寄存器
SPIDAT	0x7749	1	SPI-B串行数据寄存器
SPIFFTX	0x774A	1	SPI-BFIFO发送寄存器
SPIFFRX	0x774B	1	SPI-BFIFO接收寄存器
SPIFFCT	0x774C	1	SPI-BFIFO控制寄存器
SPIPRI	0x774F	1	SPI-B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。这空间只允许 16 位访问。32 位访问会生成未定义的后果。

表 4-7 SPI-C 寄存器

名称	地址	大小 (x16)	说明 (1)
SPICCR	0x7760	1	SPI-C配置控制寄存器
SPICTL	0x7761	1	SPI-C运行控制寄存器
SPISTS	0x7762	1	SPI-C状态寄存器
SPIBRR	0x7764	1	SPI-C波特率寄存器
SPIRXEM U	0x7766	1	SPI-C接收仿真缓冲寄存器
SPIRXBUF	0x7767	1	SPI-C串行输入缓冲寄存器
SPITXBUF	0x7768	1	SPI-C串行输出缓冲寄存器
SPIDAT	0x7769	1	SPI-C串行数据寄存器
SPIFFTX	0x776A	1	SPI-CFIFO发送寄存器
SPIFFRX	0x776B	1	SPI-CFIFO接收寄存器
SPIFFCT	0x776C	1	SPI-CFIFO控制寄存器
SPIPRI	0x776F	1	SCI-C优先级控制寄存器

(1)这个表中寄存器被映射到外设帧 2。这空间只允许 16 位访问。32 位访问会生成未定义的后果。

表 4-8 SPI-D 寄存器

名称	地址	大小 (x16)	说明 (1)
SPICCR	0x7780	1	SPI-D配置控制寄存器
SPICTL	0x7781	1	SPI-D运行控制寄存器
SPISTS	0x7782	1	SPI-D状态寄存器
SPIBRR	0x7784	1	SPI-D波特率寄存器
SPIRXEM U	0x7786	1	SPI-D接收仿真缓冲寄存器
SPIRXBUF	0x7787	1	SPI-D串行输入缓冲寄存器
SPITXBUF	0x7788	1	SPI-D串行输出缓冲寄存器
SPIDAT	0x7789	1	SPI-D串行数据寄存器
SPIFFTX	0x778A	1	SPI-DFIFO发送寄存器
SPIFFRX	0x778B	1	SPI-DFIFO接收寄存器
SPIFFCT	0x778C	1	SPI-DFIFO控制寄存器
SPIPRI	0x778F	1	SPI-D优先级控制寄存器

(1)这个表中寄存器被映射到外设帧 2 这空间只允许 16 位访问。32 位访问会产生未定义的后果。

图 4-7 是一个处于从机模式下的 SPI 方框图。

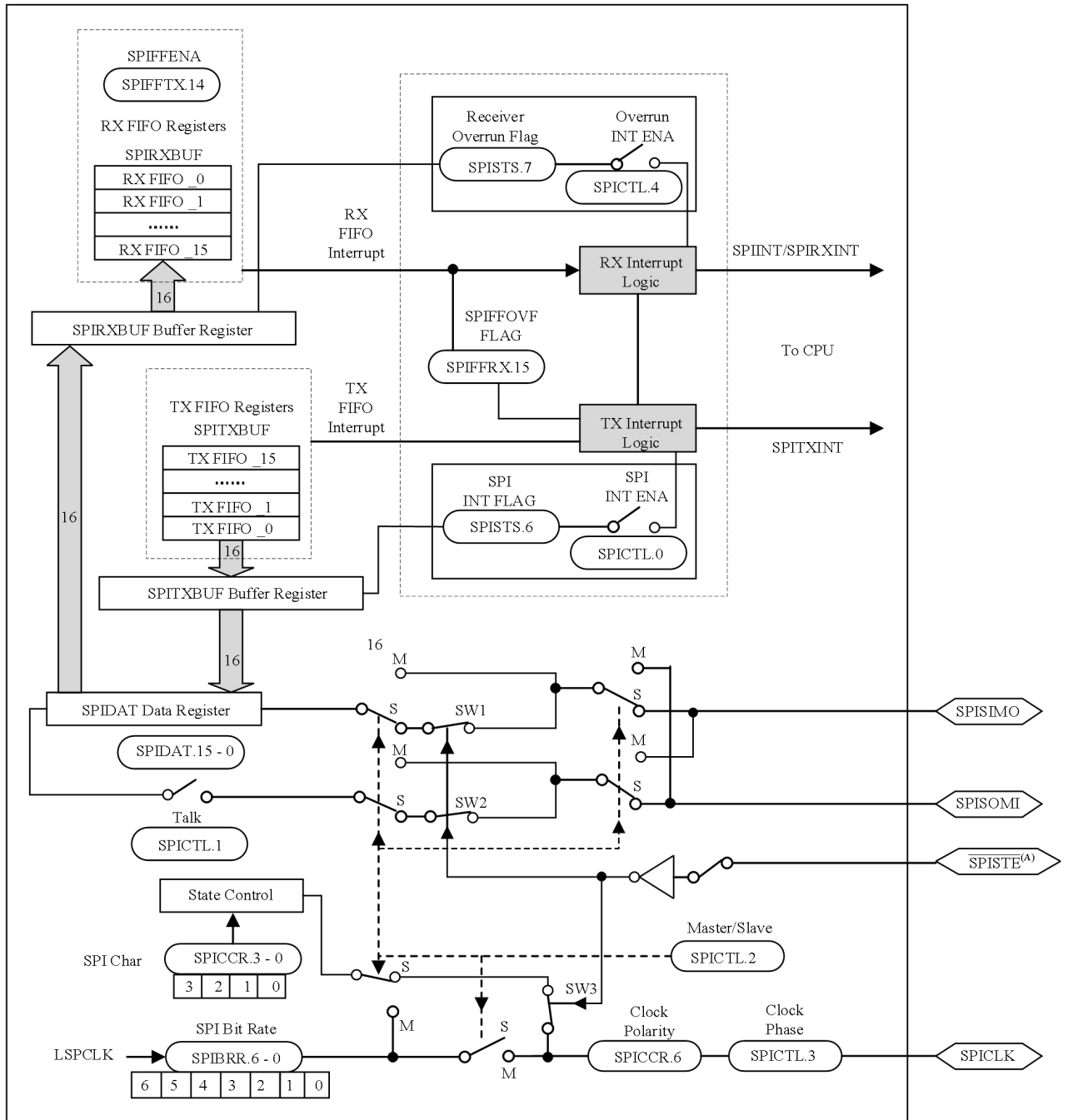


图 4-7 SPI 模块方框图 (从机模式)

4.5 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1Mbps 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告
- 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6 协同通信）
- 自测模式
 - 运行在接收自身消息的回路模式。用来提供一个虚拟的确认，从而不需要由其他节点来提供确认位的需要。

注

对于 100MHz 的 SYSCLKOUT，最小比特率为 15.625kbps。

对于 60MHz 的 SYSCLKOUT，最小比特率为 9.375kbps。

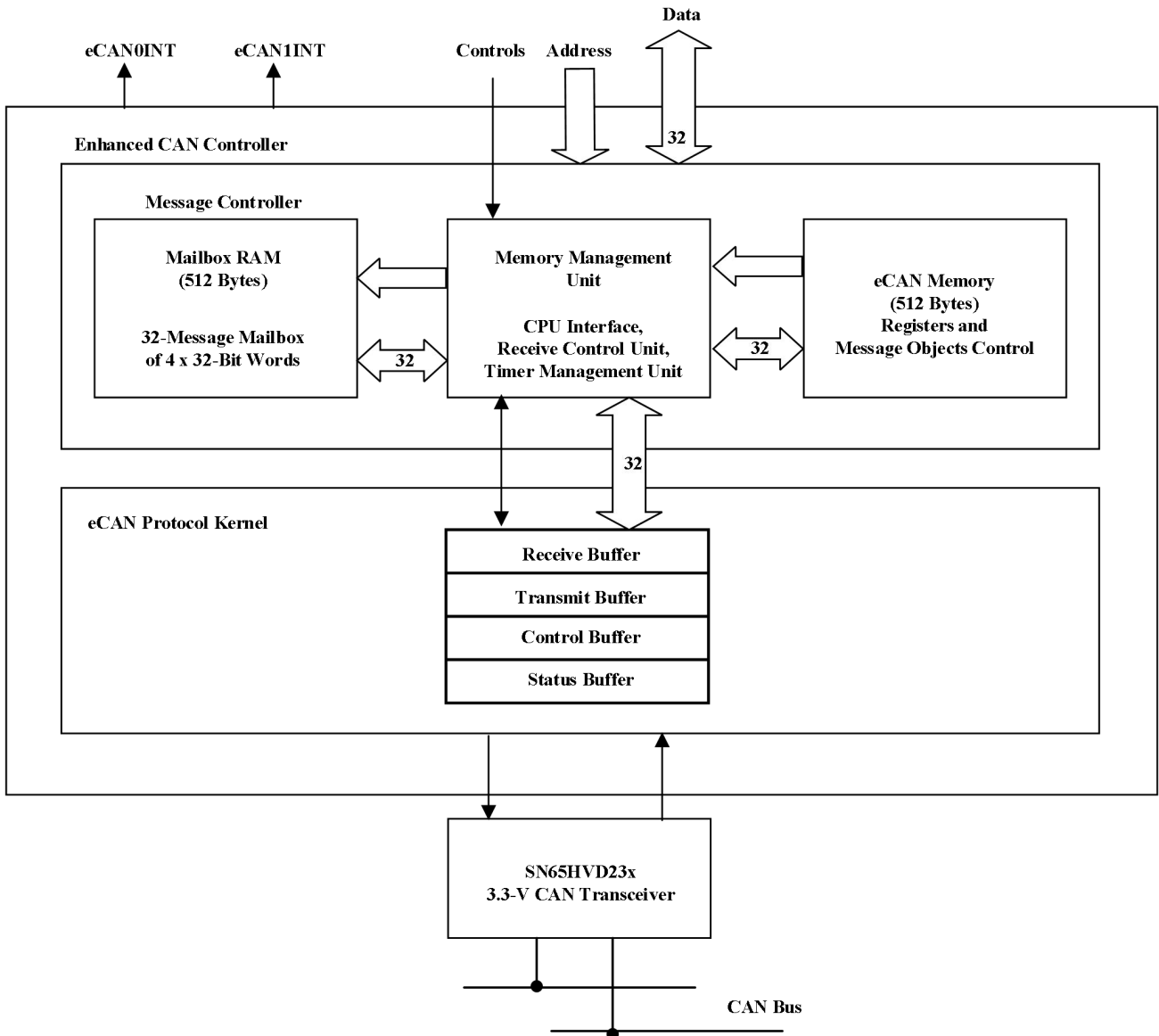


图 4-8 eCAN 方框图和接口电路

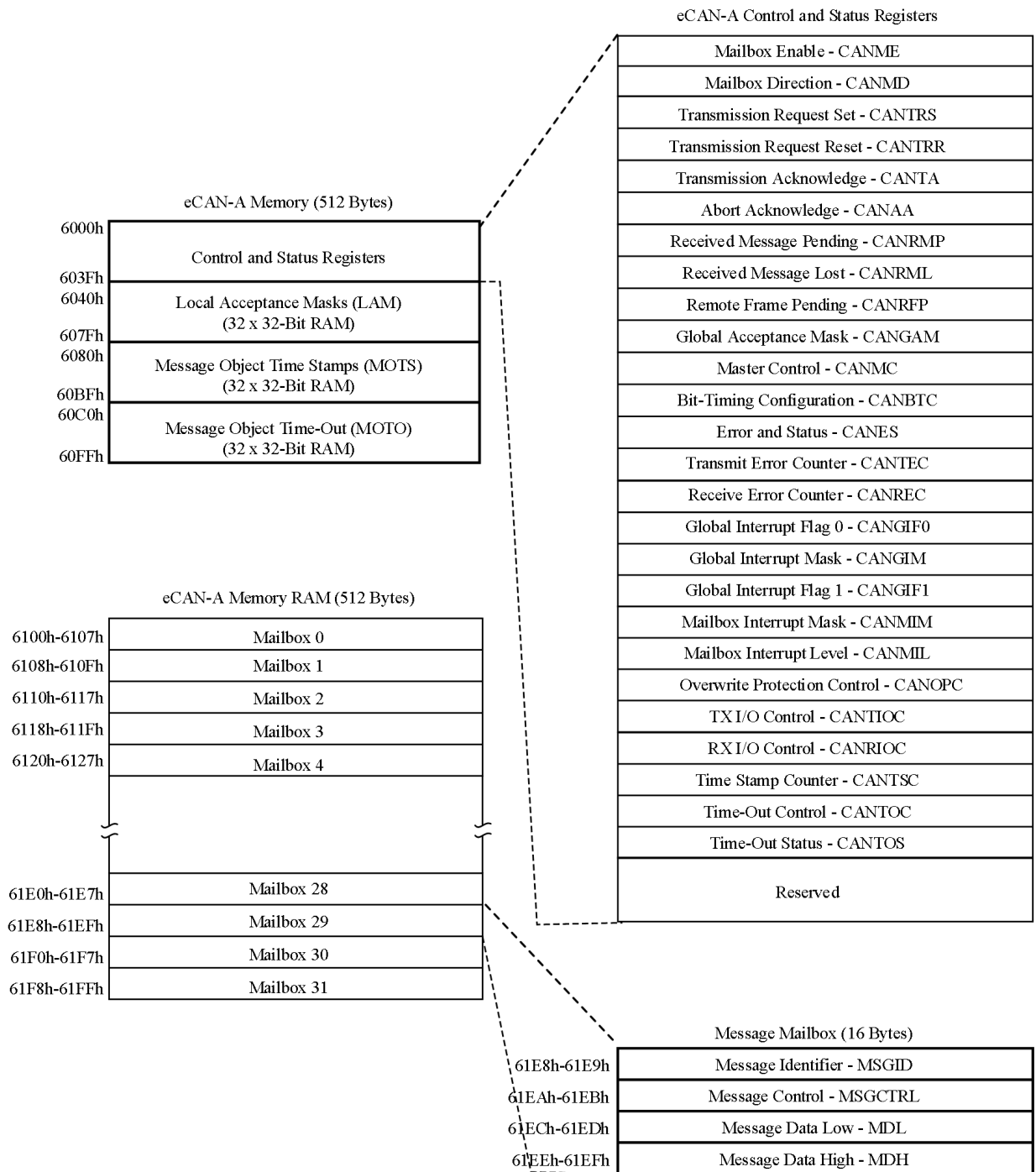


图4-9 eCAN-A内存映射

注

如果未在应用中使用 eCAN 模块，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。

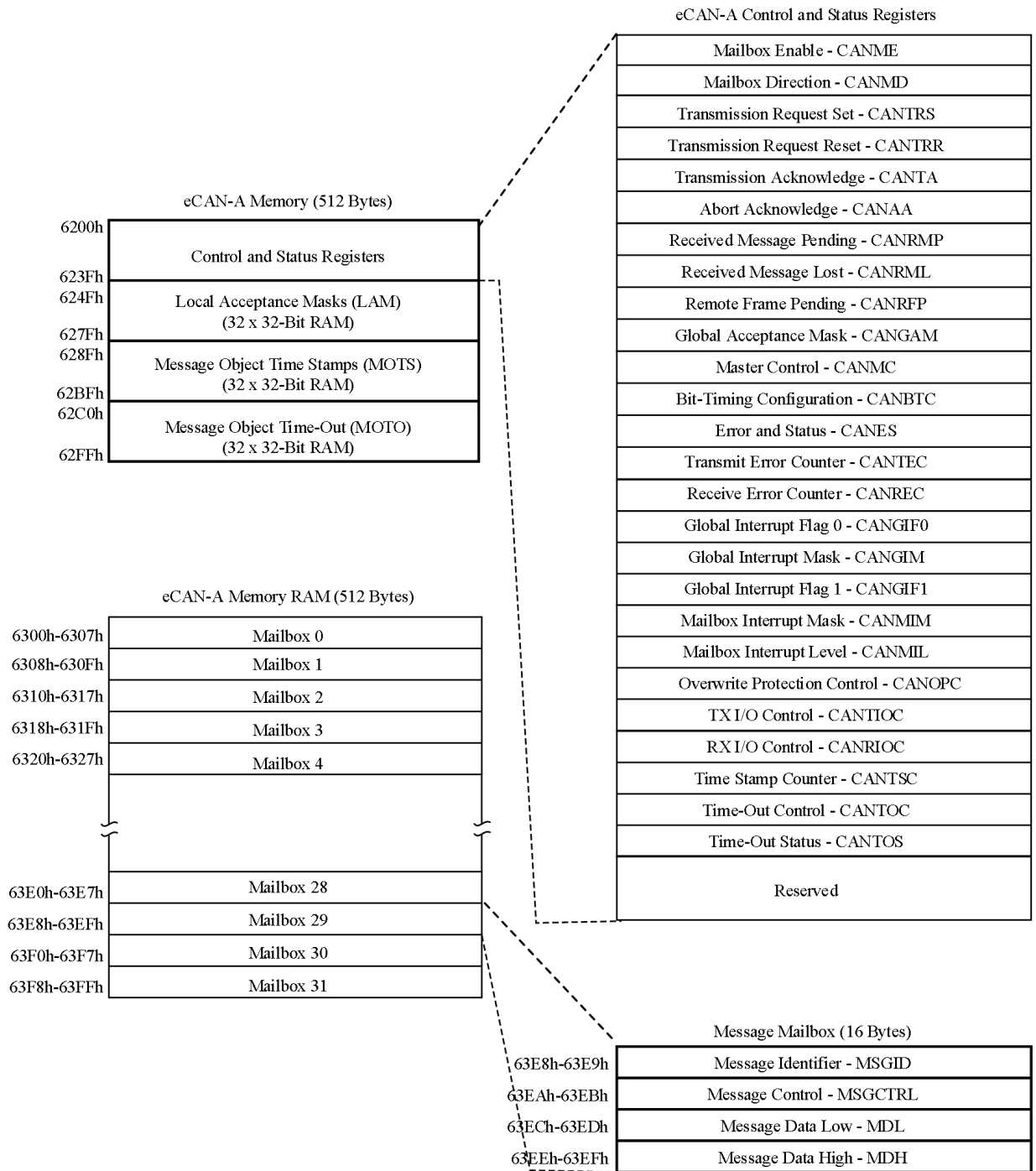


图4-10 eCAN-B内存映射

表 4-10 中列出的 CAN 寄存器，被 CPU 用来配置和控制 CAN 控制器以及消息目标。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 位访问与一个偶边界对齐。

表4-10 CAN 寄存器映射⁽¹⁾

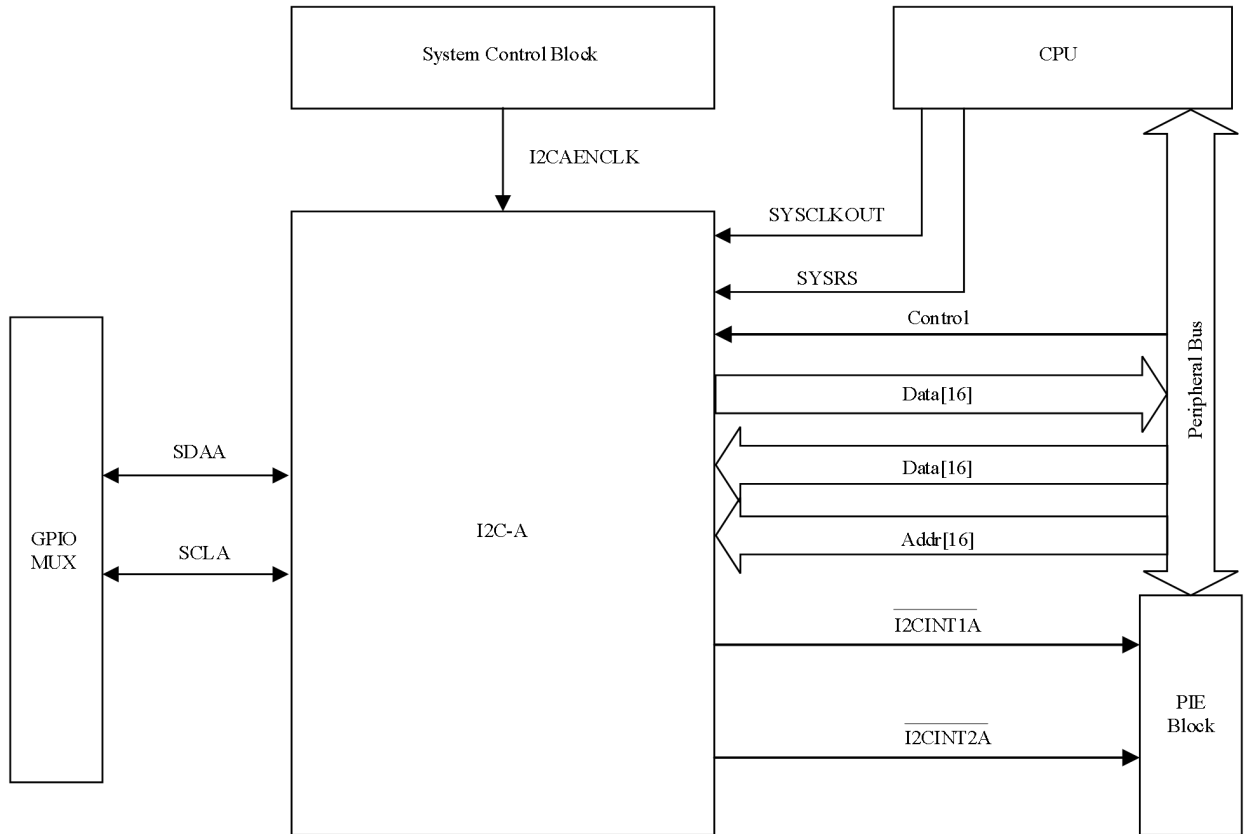
寄存器名称	eCAN-A 地址	eCAN-B 地址	大小 (x32)	说明
CANME	0x6000	0x6200	1	邮箱启用
CANMD	0x6002	0x6202	1	邮箱方向
CANTRS	0x6004	0x6204	1	发送请求设定
CANTRR	0x6006	0x6206	1	发送请求复位
CANTA	0x6008	0x6208	1	传输确认
CANAA	0x600A	0x620A	1	中止确认
CANRMP	0x600C	0x620C	1	接收消息等待
CANRML	0x600E	0x620E	1	接收消息丢失
CANRFP	0x6010	0x6210	1	远程帧等待
CANGAM	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	TXI/O控制
CANRIOC	0x602C	0x622C	1	RXI/O控制
CANTSC	0x602E	0x622E	1	时间戳控制 (在 SCC模式中被保留)
CANTOC	0x6030	0x6230	1	超时控制 (在SOC模式中被保留)
CANTOS	0x6032	0x6232	1	超时状态 (在 SCC模式中被保留)

(1) 这些寄存器被映射至外设帧 1。

4.6 内部集成电路(I2C)

ADP32F08 包含一个 I2C 串行端口。[图 4-11](#) 显示了此器件内的 I2C 外设模块接口。I2C 模块具有以下特性：

- 符合飞利浦半导体 I2C 总线规格 (版本 2.1):
 - 支持 1 位至 8 位格式传输
 - 7 位和 10 位寻址模式
 - 常规调用
 - START 字节模式
 - 支持多个主发送器和从接收器
 - 支持多个从发送器和主接收器
 - 组合主器件发送/接收和接收/发送模式
 - 数据传输速率从 10kbps 到高达 400kbps (I2C 快速模式速率)
- 一个 16 字接收 FIFO 和一个 16 字发送 FIFO
- 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：
 - 发送数据准备好
 - 接收数据准备好
 - 寄存器访问准备好
 - 没有接收到确认
 - 仲裁丢失
 - 检测到停止条件
 - 被寻址为从器件
- 在 FIFO 模式下, CPU 可以使用附加中断
- 模块启用/禁用能力
- 自由数据格式模式



- A. 在 SYSCLKOUT 速率上对 I2C 寄存器进行访问。I2C 端口的内部定时和信号波形也为 SYSCLKOUT 速率。
- B. PCLKCR0 寄存器内的时钟启用位 (I2CAENCLK) 关闭到 I2C 端口的时钟以实现低功耗运行。复位时, I2CAENCLK 被清除, 这表明外设内部时钟被关闭。

图 4-11 I2C 外设模块接口

表 4-11 中的寄存器配置并且扩展 I2C 端口操作。

表 4-11 I2C-A 寄存器

名称	地址	说明
I2COA	0x7900	I2C自身的地址寄存器
I2CIER	0x7901	I2C中断启用寄存器
I2CST	0x7902	I2C状态寄存器
I2CCLKL	0x7903	I2C时钟低电平时间分频器寄存器
I2CCLKH	0x7904	I2C时钟高电平时间分频器寄存器
I2CCNT	0x7905	I2C数据数量寄存器
I2CDRR	0x7906	I2C数据接收寄存器
I2CSAR	0x7907	I2C从器件地址寄存器
I2CDXR	0x7908	I2C数据发送寄存器
I2CMDR	0x7909	I2C模式寄存器
I2CISRC	0x790A	I2C中断源寄存器
I2CPSC	0x790C	I2C预分频器寄存器
I2CFFTX	0x7920	I2CFIFO发送寄存器
I2CFFRX	0x7921	I2CFIFO接收寄存器
I2CRSR	-	I2C接收移位寄存器 (CPU不可访问)
I2CXSR	-	I2C发送移位寄存器 (CPU不可访问)

4.7 增强型 PWM 模块 (ePWM1/2/3/4/5/6)

器件包含高达 6 个增强型 PWM 模块 (ePWM)。图 4-12 为 ePWM 模块图。图 4-13 为与 ePWM 互连的信号。

表 4-12 显示了每个模块的完整的 ePWM 寄存器集。

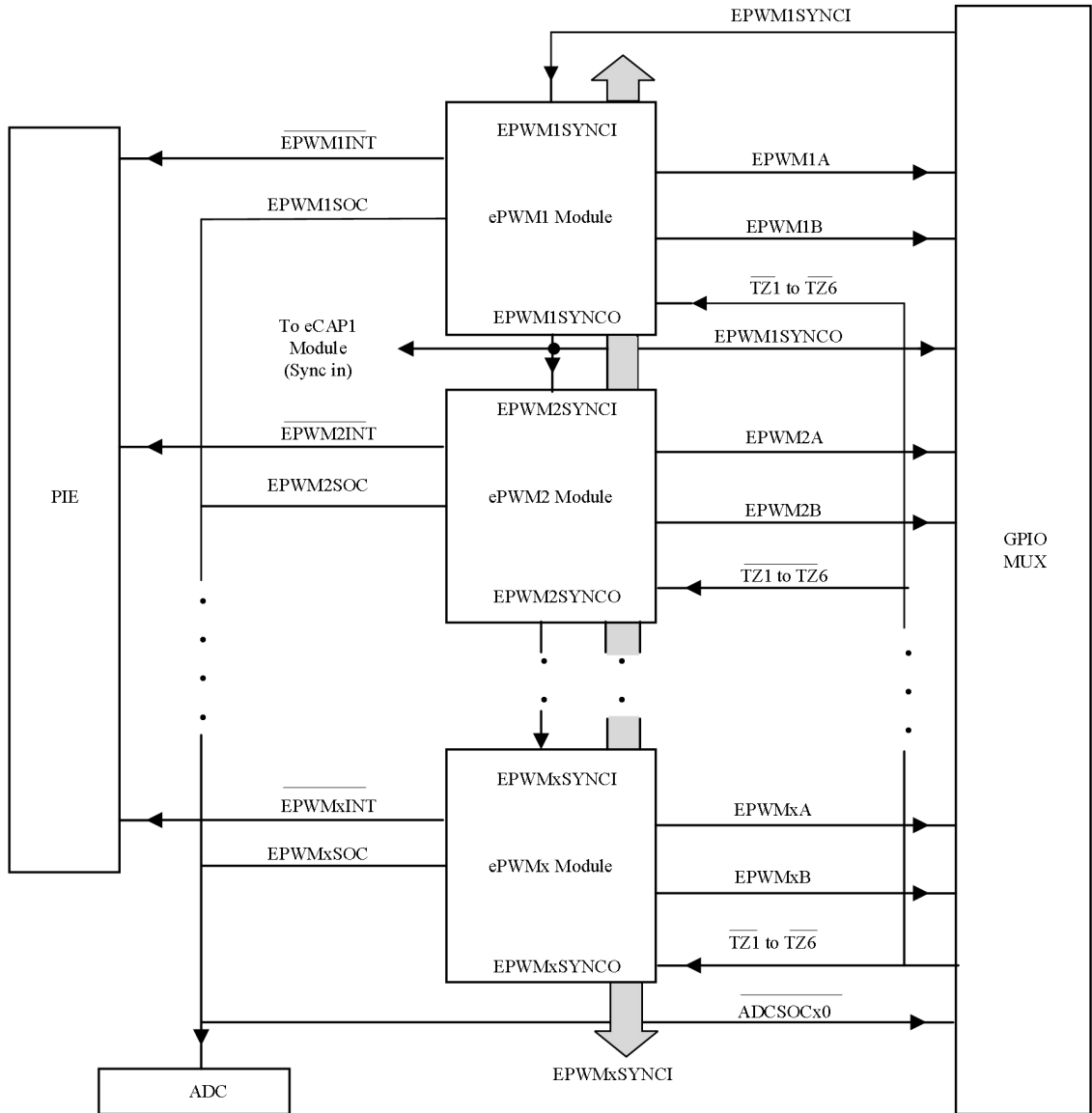


图 4-12 ePWM 模块

表 4-12 ePWM1-ePWM4 控制和状态寄存器

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	不可用	不可用	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	不可用	不可用	1/1	时基比较 AHRPWM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	针对输出 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
AQSFRCD	0x680D	0x684D	0x688D	0x68CD	0x690D	0x694D	1/0	操作限定器软件强制寄存器
AQCSFRCE	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 ⁽²⁾	0x6960 ⁽²⁾	1/0	HRPWM配置寄存器 ⁽¹⁾

(1) 寄存器受 EALLOW 保护。

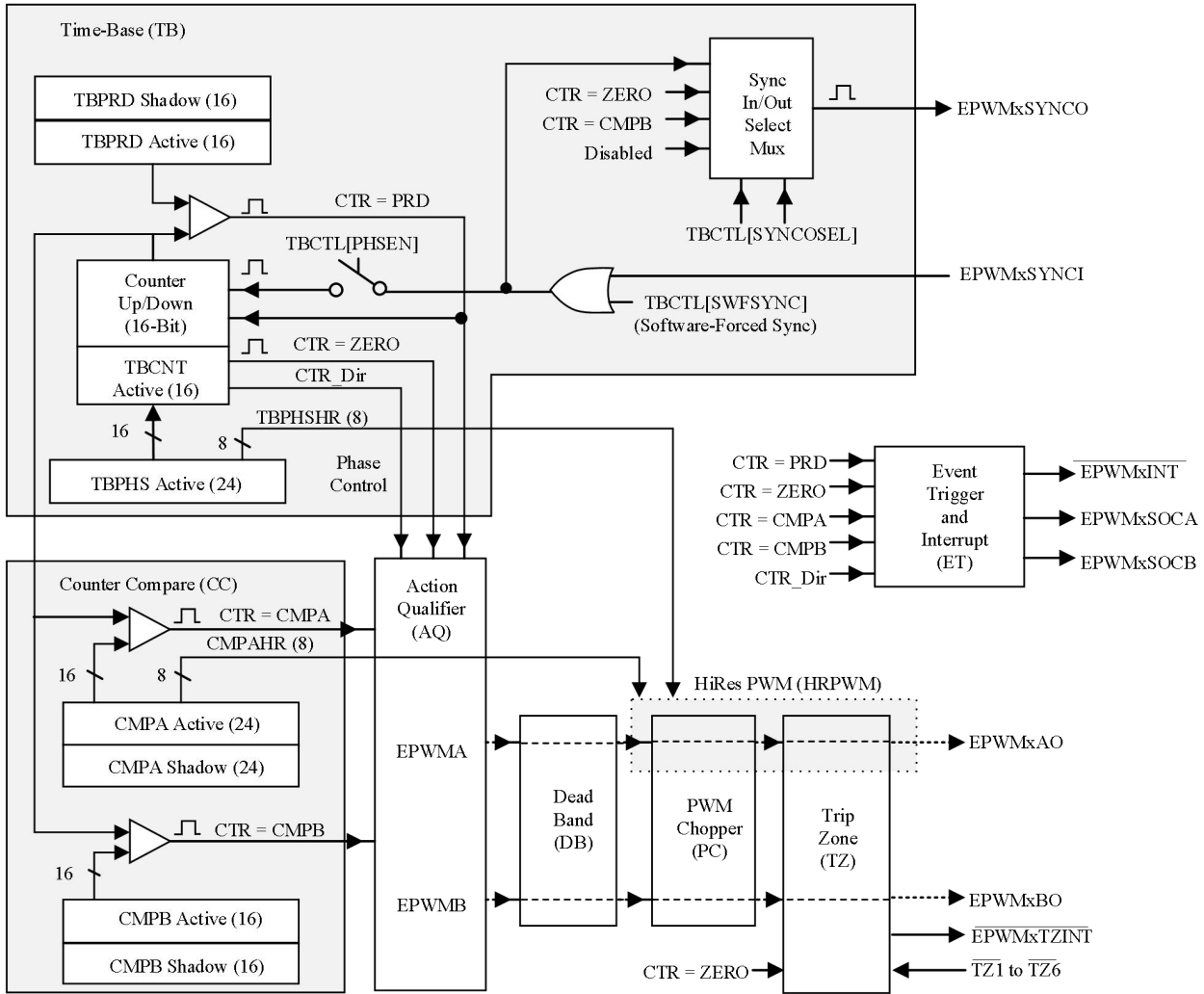


图 4-13ePWM 信号互连

4.8 高精度 PWM (HRPWM)

HRPWM 模块配置的 PWM 分辨率 (时间粒度), 此分辨率显著优于使用传统数字 PWM 方法所导出的分辨率。HRPWM 模块的关键点为:

- 极大地扩展了传统 PWM 的分辨率
- 通常在有效 PWM 精度下降到低于约 9-10 位时使用。当使用 100MHz 的 CPU/系统时钟时, 这种情况在 PWM 频率大于约 200kHz 时发生。
- 该功能可用于占空比和相移控制方法。
- 通过对 ePWM 模块的比较 A 和相位寄存器的扩展, 来进行更加精细的时间粒度控制或边沿

定位。

- 仅在 ePWM 模块的 A 信号路径上配置 HRPWM 功能（也就是说，在 EPWMxA 输出上配置）。EPWMxB 输出传统 PWM 功能。

4.9 增强型捕捉模块 (eCAP1)

器件包含多达四个增强型捕捉 (eCAP) 模块。图 4-14 显示了一个模块的功能块图。eCAP 模块以 SYSCLKOUT 速率计时。

PCLKCR1 寄存器中的时钟启用位 (ECAP1/2/3/4ENCLK) 用于独立关闭 eCAP 模块。复位时，ECAP1ENCLK, ECAP2ENCLK, ECAP3ENCLK, 和 ECAP4ENCLK 被设为低电平，表明外设时钟关闭。

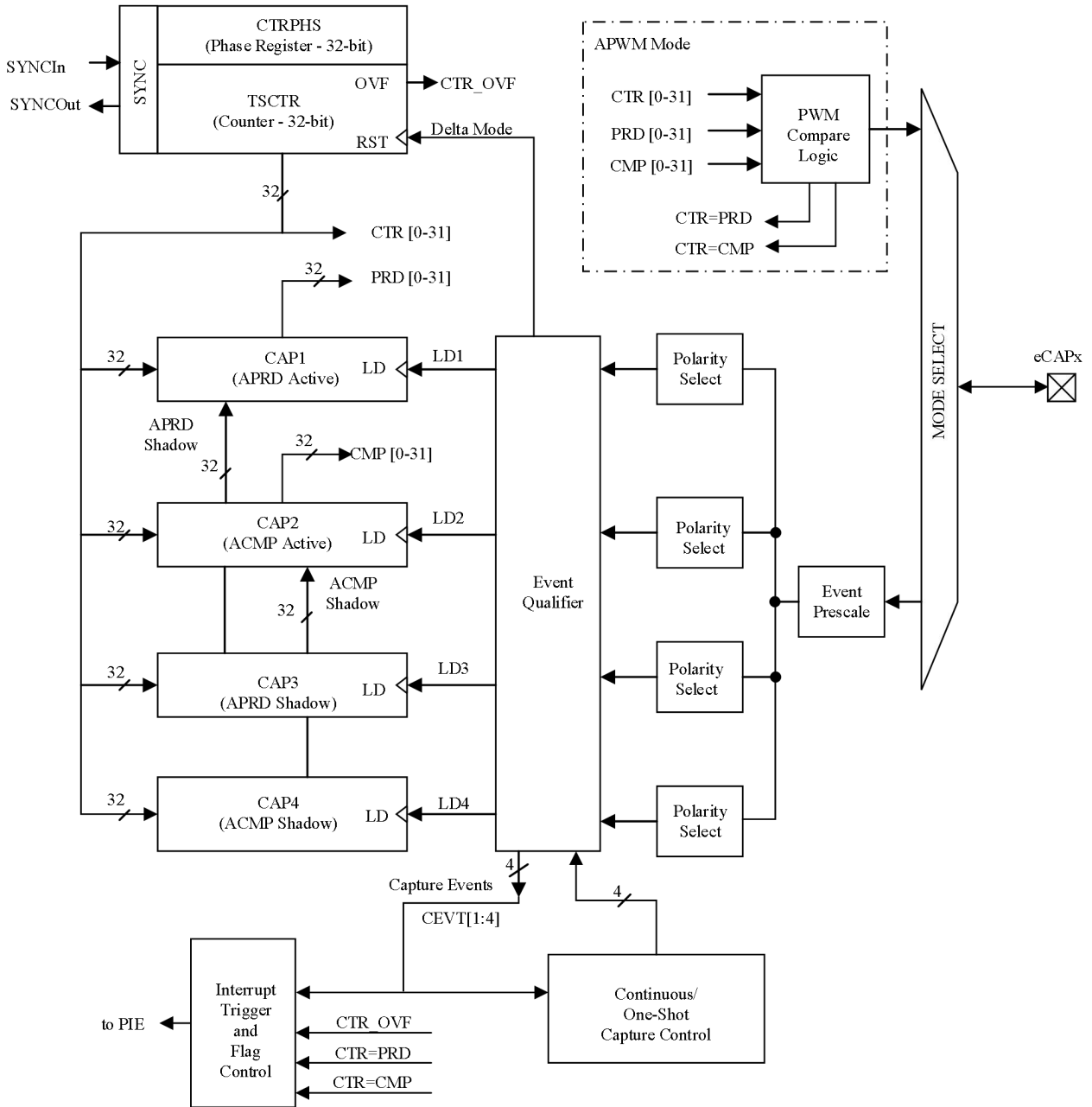


图 4-14 eCAP 功能方框图

表 4-13 eCAP 控制和状态寄存器

名称	eCAP1	eCAP2	eCAP3	eCAP4	大小 (x 16)	说明
TSCTR	0x6A00	0x6A20	0x6A40	0x6A60	2	时间戳计数器
CTRPHS	0x6A02	0x6A22	0x6A42	0x6A62	2	计数器相位偏移值寄存器
CAP1	0x6A04	0x6A24	0x6A44	0x6A64	2	捕捉 1 寄存器
CAP2	0x6A06	0x6A26	0x6A46	0x6A66	2	捕捉 2 寄存器
CAP3	0x6A08	0x6A28	0x6A48	0x6A68	2	捕捉 3 寄存器
CAP4	0x6A0A	0x6A2A	0x6A4A	0x6A6A	2	捕捉 4 寄存器
被保留	0x6A0C- 0x6A12	0x6A2C- 0x6A32	0x6A4C- 0x6A52	0x6A6C- 0x6A72	8	被保留
ECCTL1	0x6A14	0x6A34	0x6A54	0x6A74	1	捕捉控制寄存器 1
ECCTL2	0x6A15	0x6A35	0x6A55	0x6A75	1	捕捉控制寄存器 2
ECEINT	0x6A16	0x6A36	0x6A56	0x6A76	1	捕捉中断启用寄存器
ECFLG	0x6A17	0x6A37	0x6A57	0x6A77	1	捕捉中断标志寄存器
ECCLR	0x6A18	0x6A38	0x6A58	0x6A78	1	捕捉中断清除寄存器
ECFRC	0x6A19	0x6A39	0x6A59	0x6A79	1	捕捉中断强制寄存器
被保留	0x6A1A- 0x6A1F	0x6A3A- 0x6A3F	0x6A5A- 0x6A5F	0x6A7A- 0x6A7F	6	被保留

4.10 增强型正交编码器脉冲 (eQEP1/2)

此器件包含两个增强型正交编码器脉冲 (eQEP) 模块。

[图 4-15](#) 显示了 eQEP 功能块图。

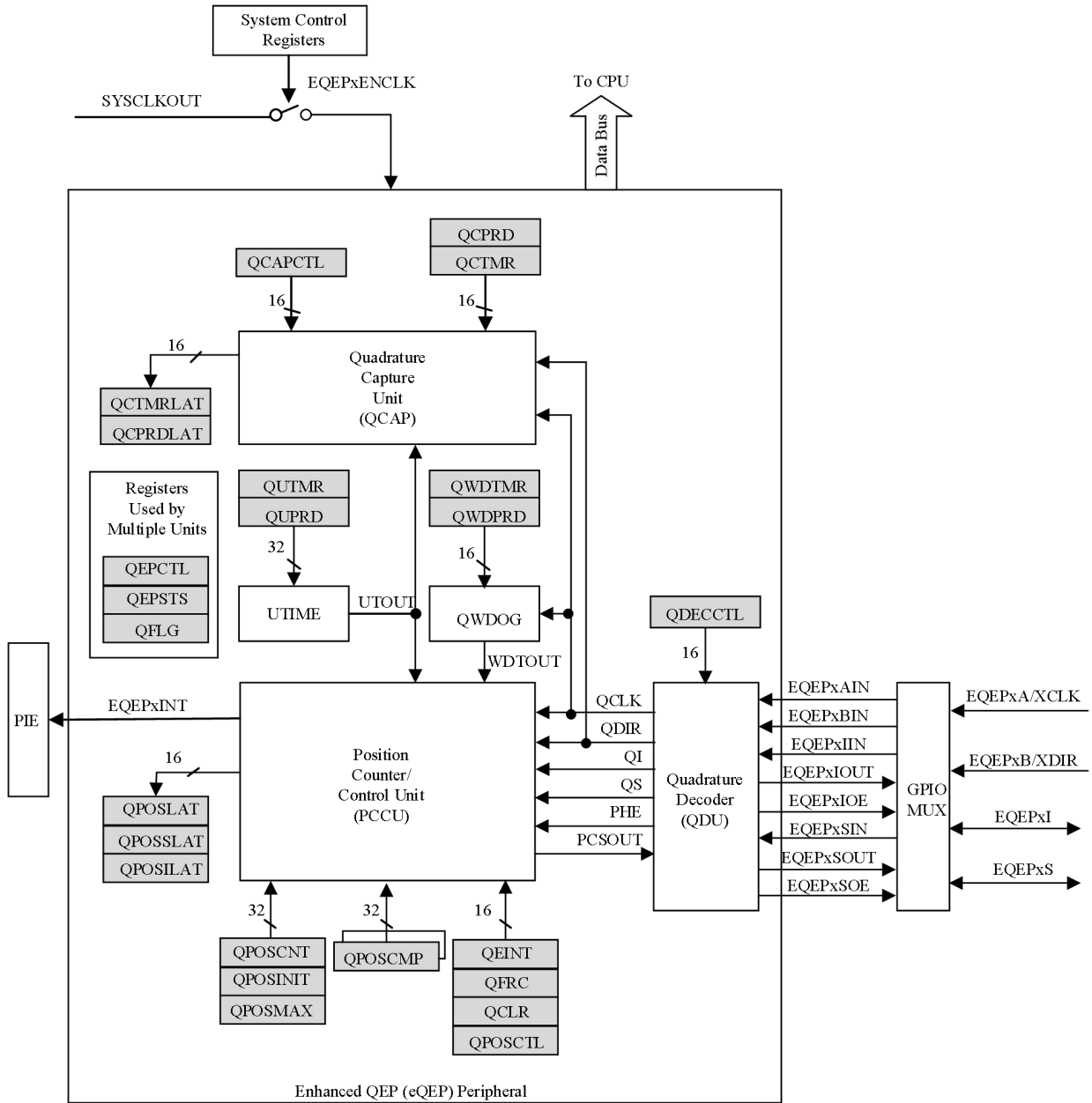


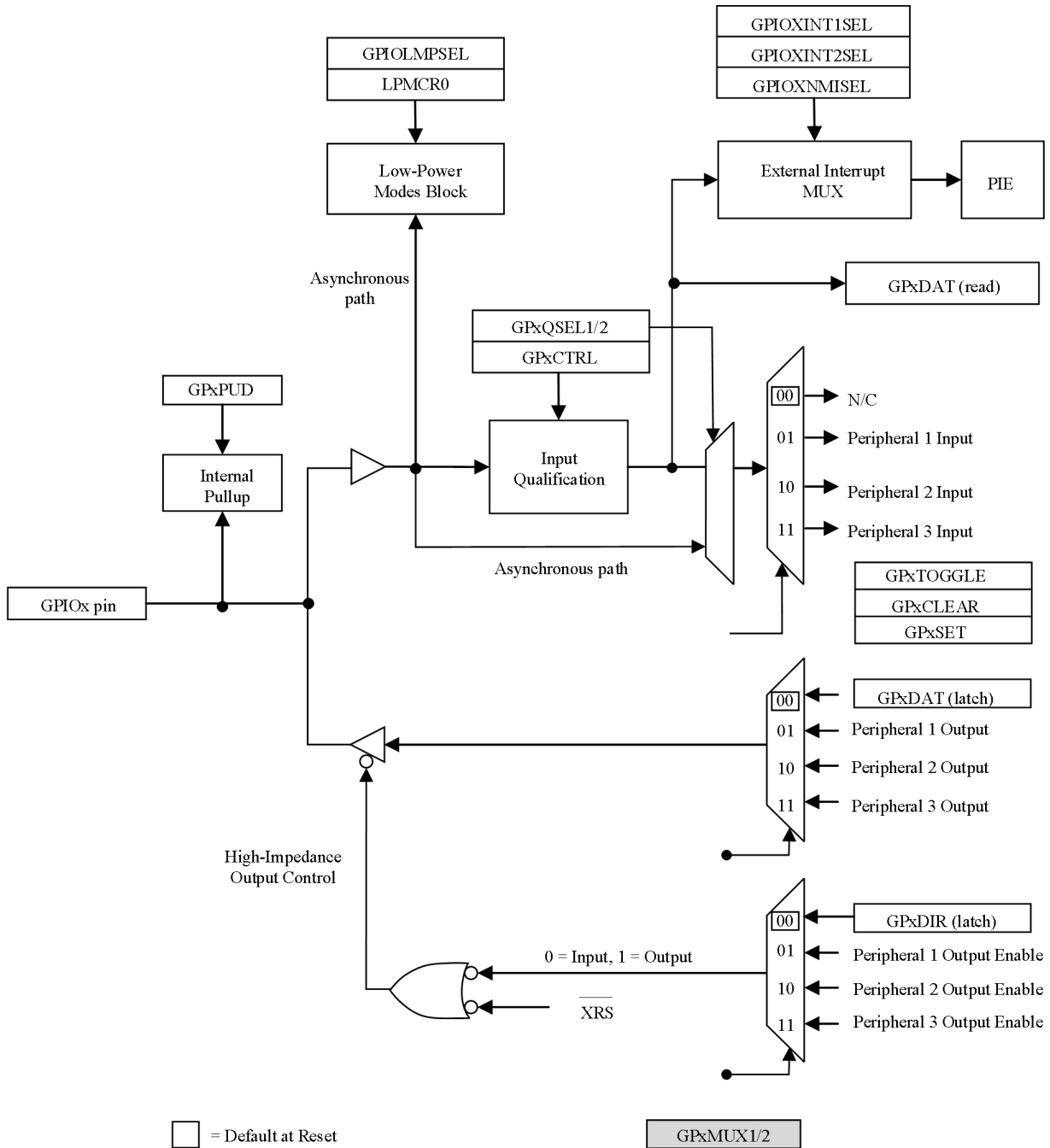
图 4-15 eQEP 功能块图

表 4-14 eQEP 控制和状态寄存器

名称	eQEP1 地址	eQEP2 地址	eQEP1 大小 (x16)/ #SHADOW	寄存器说明
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 单元定时器
QUPRD	0x6B10	0x6B50	2/0	eQEP 单元周期寄存器
QWDTMR	0x6B12	0x6B52	1/0	eQEP 安全装置定时器
QWDPRD	0x6B13	0x6B53	1/0	eQEP 安全装置周期寄存器
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 捕捉控制寄存器
QPOSCTL	0x6B17	0x6B57	1/0	eQEP 位置比较控制寄存器
QEINT	0x6B18	0x6B58	1/0	eQEP 中断启用寄存器
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 捕捉定时器
QCPRD	0x6B1E	0x6B5E	1/0	eQEP 捕捉周期寄存器
QCTMRLAT	0x6B1F	0x6B5F	1/0	eQEP 捕捉定时器锁存
QCPRDLAT	0x6B20	0x6B60	1/0	eQEP 捕捉周期锁存
被保留	0x6B21- 0x6B3F	0x6B61- 0x6B7F	31/0	被保留

4.11 GPIO MUX

在 ADP32F08 上，除了提供独立的引脚 I/O 功能外，GPIOMUX 还可以将最多 3 个独立的外设信号复用在单一的一个 GPIO 引脚上。图 4-16 为每个引脚的 GPIOMUX 方框图。由于 I2C 引脚的开漏功能，这些引脚的 GPIOMUX 方框图是不同的。



- A. x 代表端口, A 或 B。例如, GPxDIR是指 GPADIR 或者 GPBDIR寄存器, 至于是哪一个是哪一个寄存器, 则取决于所选择的特定 GPIO引脚。
- B. 在相同的存储器位置访问 GPxDAT锁定/读取。

图 4-16GPIOMUX方框图

ADP32F08 支持 34 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 4-15 显示了 GPIO 寄存器映射。

表 4-15 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	GPIOA控制寄存器(GPIO0至 31)
GPAQSEL1	0x6F82	2	GPIOA限定器选择1寄存器 (GPIO0至 15)
GPAQSEL2	0x6F84	2	GPIOA限定器选择2寄存器 (GPIO16至 31)
GPAMUX1	0x6F86	2	GPIOAMUX1寄存器 (GPIO0至 15)
GPAMUX2	0x6F88	2	GPIOAMUX2寄存器 (GPIO16至 31)
GPAD IR	0x6F8A	2	GPIOA方向寄存器(GPIO0至 31)
GPAPUD	0x6F8C	2	GPIOA上拉禁用寄存器(GPIO0至 31)
被保留	0x6F8E 0x6F8F	2	被保留
GPBCTRL	0x6F90	2	GPIOB控制寄存器(GPIO32至 35)
GPBQSEL1	0x6F92	2	GPIOB限定器选择1寄存器 (GPIO32至 35)
GPBQSEL2	0x6F94	2	被保留
GPBMUX1	0x6F96	2	GPIOBMUX1寄存器 (GPIO32至 35)
GPBMUX2	0x6F98	2	被保留
GPBDI R	0x6F9A	2	GPIOB方向寄存器(GPIO32至 35)
GPBPUD	0x6F9C	2	GPIOB上拉禁用寄存器(GPIO32至 35)
被保留	0x6F9E 0x6F9F	2	被保留
被保留	0x6FA0 0x6FBF	32	被保留
GPIO数据寄存器 (不受 EALLOW保护)			
GPAD AT	0x6FC0	2	GPIO数据寄存器 (GPIO0至 31)
GPAS ET	0x6FC2	2	GPIO数据设定寄存器 (GPIO0至 31)
GPACLEAR	0x6FC4	2	GPIO数据清除寄存器 (GPIO0至 31)
GPATOGGLE	0x6FC6	2	GPIO数据切换寄存器 (GPIO0至 31)
GPBD AT	0x6FC8	2	GPIO数据寄存器 (GPIO32至 35)
GPBS ET	0x6FCA	2	GPIO数据设定寄存器 (GPIO32至 35)
GPBCLEAR	0x6FCC	2	GPIO数据清除寄存器 (GPIO32至 35)
GPBTOGGLE	0x6FCE	2	GPIO数据切换寄存器 (GPIO32至 35)
被保留	0x6FD0 0x6FDF	16	被保留
GPIO中断和低功耗模式选择寄存器 (受 EALLOW保护)			
GPIOXINT1SEL	0x6FE0	1	XINT1GPIO输入选择寄存器 (GPIO0至 31)
GPIOXINT2SEL	0x6FE1	1	XINT2GPIO输入选择寄存器 (GPIO0至 31)
GPIOXNMISEL	0x6FE2	1	XNMI输入选择寄存器 (GPIO0至 31)

被保留	0x6FE3 0x6FE7	5	被保留
GPIOLPMSEL	0x6FE8	2	LPMGPIO选择寄存器(GPIO0至 31)
被保留	0x6FEA 0x6FFF	22	被保留

表 4-16 GPIO MUX

GPAMUX1/2 ⁽¹⁾ 寄存器位	复位时默认主 I/O 功能 (GPxMUX1/2 位= 0, 0)	外设选择 1 ⁽²⁾ (GPxMUX1/2 位= 0, 1)	外设选择 2 (GPxMUX1/2 位= 1, 0)	外设选择 3 (GPxMUX1/2 位= 1, 1)
GPAMUX1				
1-0	GPIO0	EPWM1A(O)	被保留 ⁽³⁾	被保留 ⁽³⁾
3-2	GPIO1	EPWM1B(O)	SPISIMOD(I/O)	被保留 ⁽³⁾
5-4	GPIO2	EPWM2A(O)	被保留 ⁽³⁾	被保留 ⁽³⁾
7-6	GPIO3	EPWM2B(O)	SPISOMID(I/O)	被保留 ⁽³⁾
9-8	GPIO4	EPWM3A(O)	被保留 ⁽³⁾	被保留 ⁽³⁾
11-10	GPIO5	EPWM3B(O)	SPICLKD(I/O)	ECAP1(I/O)
13-12	GPIO6	EPWM4A(O)	EPWMSYNCl(I)	EPWMSYNCO(O)
15-14	GPIO7	EPWM4B(O)	SPISTED(I/O)	ECAP2(I/O)
17-16	GPIO8	EPWM5A(O)	CANTXB(O)	ADCSOCAO(O)
19-18	GPIO9	EPWM5B(O)	SCITXDB(O)	ECAP3(I/O)
21-20	GPIO10	EPWM6A(O)	CANRXB(I)	ADCSOCBO(O)
23-22	GPIO11	EPWM6B(O)	SCIRXDB(I)	ECAP4(I/O)
25-24	GPIO12	TZ1(I)	CANTXB(O)	SPISIMOB(I/O)
27-26	GPIO13	TZ2(I)	CANRXB(I)	SPISOMIB(I/O)
29-28	GPIO14	TZ3(I)	SCITXDB(O)	SPICLKB(I/O)
31-30	GPIO15	TZ4(I)	SCIRXDB(I)	SPISTEB(I/O)
GPAMUX2				
1-0	GPIO16	SPISIMOA(I/O)	CANTXB(O)	TZ5(I)
3-2	GPIO17	SPISOMIA(I/O)	CANRXB(I)	TZ6(I)
5-4	GPIO18	SPICLKA(I/O)	SCITXDB(O)	被保留 ⁽⁴⁾
7-6	GPIO19	SPISTEA(I/O)	SCIRXDB(I)	被保留 ⁽⁴⁾
9-8	GPIO20	EQEP1A(I)	SPISIMOC(I/O)	CANTXB(O)
11-10	GPIO21	EQEP1B(I)	SPISOMIC(I/O)	CANRXB(I)
13-12	GPIO22	EQEP1S(I/O)	SPICLKC(I/O)	SCITXDB(O)
15-14	GPIO23	EQEP1I(I/O)	SPISTEC(I/O)	SCIRXDB(I)
17-16	GPIO24	ECAP1(I/O)	EQEP2A(I)	SPISIMOB(I/O)
19-18	GPIO25	ECAP2(I/O)	EQEP2B(I)	SPISOMIB(I/O)
21-20	GPIO26	ECAP3(I/O)	EQEP2I(I/O)	SPICLKB(I/O)
23-22	GPIO27	ECAP4(I/O)	EQEP2S(I/O)	SPISTEB(I/O)
25-24	GPIO28	SCIRXDA(I)	被保留 ⁽⁴⁾	TZ5(I)

27-26	GPIO29	SCITXDA(O)	被保留 ⁽⁴⁾	TZ6(I)
29-28	GPIO30	CANRXA(I)	被保留 ⁽⁴⁾	被保留 ⁽⁴⁾
31-30	GPIO31	CANTXA(O)	被保留 ⁽⁴⁾	被保留 ⁽⁴⁾
GPBMUX1				
1-0	GPIO32	SDAA(I/OC)	EPWMSYNCI(I)	ADCSOCAO(O)
3-2	GPIO33	SCLA(I/OC)	EPWMSYNCO(O)	ADCSOCBO(O)
5-4	GPIO34	被保留 ⁽⁴⁾	被保留 ⁽⁴⁾	被保留 ⁽⁴⁾

- (1) GPxMUX1/2指适合引脚的 MUX寄存器; GPAMUX1, GPAMUX2或 GPBMUX1。
- (2) 这个表属于 ADP32F08器件。
- (3) “保留”字意味着没有外设被指定到这个 GPxMUX1/2寄存器设置。如果它被选择,那么引脚的状态将为未定义并且此引脚可被驱动。这个选择是为以后扩展预留的保留配置。
- (4) “保留”字意味着没有外设被指定到这个 GPxMUX1/2寄存器设置。如果它被选择,那么引脚的状态将为未定义并且此引脚可被驱动。这个选择是为以后扩展预留的保留配置。

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:

- 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)。
- 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。

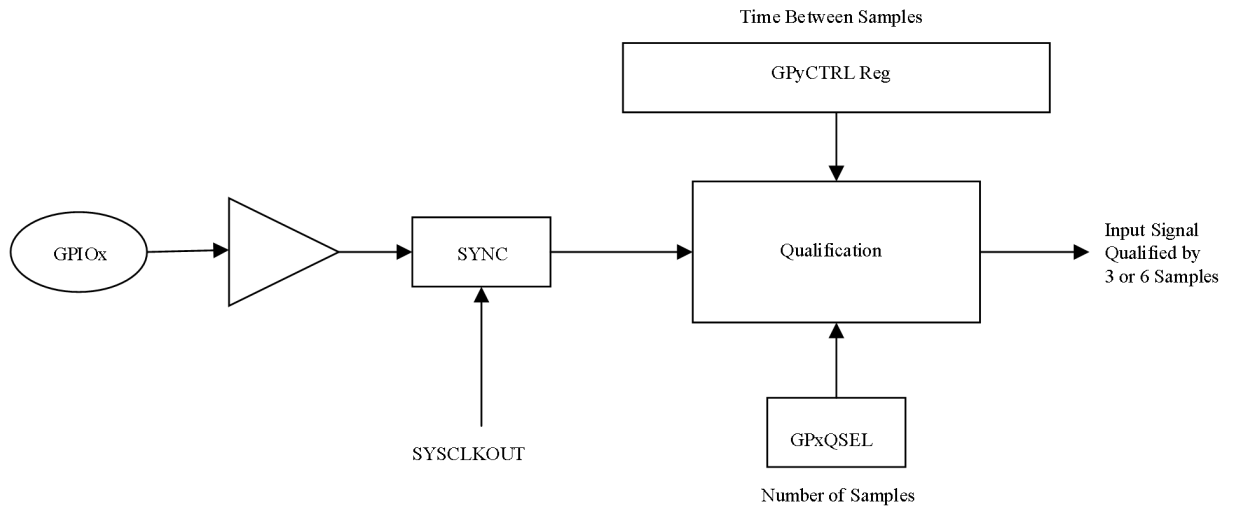


图 4-17 使用采样窗口的限定

- 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 5-12 (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。

- 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。

5 电气规范

5.1 最大绝对额定值⁽¹⁾⁽²⁾

电源电压范围, V_{DDIO}	以 V_{SS} 为参考	-0.3V至 4.6V
电源电压范围, V_{DDA2} , V_{DDAIO}	以 V_{SSA} 为参考	-0.3V至 4.6V
电源电压范围, V_{DDOSC}	以 V_{SS} 为参考	-0.3V至 2.5V
电源电压范围, V_{SSA2} , V_{SSAIO} , $V_{SS1AGND}$, $V_{SS2AGND}$	以 V_{SS} 为参考	-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 $V_{IN} > V_{DDIO})^{(3)}$		± 20 mA
输出钳制电流, $I_{OK}(V_O < 0$ 或者 $V_O > V_{DDIO})$		± 20 mA
工作环境温度范围		-40°C至 125°C ⁽⁵⁾
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 5.2 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。

(2) 所有电压值都是以 V_{SS} 为参考, 除非另有说明。

(3) 每个引脚上的持续钳制电流为 ± 2 mA。

(4) 长期高阻抗存储并/或在最大温度条件下长时间使用会使器件总体使用寿命的缩短。

(5) flash 的擦除与编程操作在 -40°C 至 85°C

5.2 推荐的运行条件

在自然通风条件下的工作温度范围内（除非另有说明）

参数		最小值	典型值	最大值	单位
器件电源电压, I/O, V_{DDIO}		3.14	3.3	3.47	V
器件电源电压 OSC, V_{DDOSC}		1.71	1.8	1.89	V
电源接地, V_{SS} , V_{SSIO}			0		V
ADC电源电压 (3.3V), V_{DDA2} , V_{DDAIO}		3.14	3.3	3.47	V
器件时钟频率 (系统时钟), $f_{SYSCLKOUT}$	100MHz器件	2		100	MHz
	60MHz器件	2		60	MHz
高电平输入电压, V_{IH}	除 X1 之外的所有输入	2		$V_{DDIO}+0.3$	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	$V_{SS}-0.3$		0.8	V
	X1			$0.3*V_{DD}$	
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 ⁽²⁾		-40		125	°C

(1) 组 2 引脚如下: GPIO28, GPIO29, GPIO30, GPIO31, TDO, XCLKOUT, EMU0 和 EMU1

(2) flash 擦除编程最高温度为 85°C

5.3 电气特性

在推荐的运行条件下（除非另有说明）

参数		测试条件		最小值	典型值	最大值	单位		
V _{OH}	高电平输出电压	I _{OH} =I _{OH} 最大值		2.4			V		
		I _{OH} =50μA		V _{DDIO} -0.2					
V _{OL}	低电平输出电压	I _{OL} =I _{OL} 最大值				0.4	V		
I _{IL}	输入电流 (低电平)	带有上拉启用的引脚	V _{DDIO} =3.3V, V _{IN} =0V	所有I/Os (包括XRS)		-80	-140	-190	μA
		具有下拉启用的引脚	V _{DDIO} =3.3V, V _{IN} =0V				±2		
I _{IH}	输入电流 (高电平)	带有上拉启用的引脚	V _{DDIO} =3.3V, V _{IN} =V _{DDIO}				±2	μA	
		具有下拉启用的引脚	V _{DDIO} =3.3V, V _{IN} =V _{DDIO}		28	50	80		
		具有下拉启用的引脚	V _{DDIO} =3.3V, V _{IN} =V _{DDIO}		80	140	190		
I _{OZ}	输出电流, 上拉或者下拉被禁用	V _O =V _{DDIO} 或者0V				±2	μA		

5.4 流耗

表 5-1 ADP32F08 在 100MHz SYSCLKOUT 运行条件下的流耗

模式	测试条件	$I_{DDIO}^{(2)}$		$I_{DDA2}^{(2)}$		$I_{DDAIO}^{(2)}$	
		典型值	最大值	典型值	最大值	典型值	最大值
可用 (闪存)	下列的外设时钟被启用： • ePWM1/2/3/4/5/6 • eCAP1/2/3/4 • eQEP1/2 • eCAN-A • SCI-A/B • SPI-A • ADC • I2C 所有 PWM 引脚被切换至 100MHz 所有 I/O 引脚保持未连接状态。 数据在 SCI-A, SCI-B, 和 eCAN-A 端口上持续发出。硬件复用器被使用。正在闪存之外运行的代码具有 3 个等待状态。 XCLKOUT 被关闭。	94mA	104mA	6mA	8mA	21mA	25mA
IDLE	闪存被断电。 XCLKOUT 被关闭。 下列的外设时钟被启用： • eCAN-A • SCI-A • SPI-A • I2C	30mA	40mA	30μA	50μA	46μA	50μA
STANDBY	闪存被断电。 外设时钟被关闭。	5mA	7mA	22μA	30μA	46μA	50μA
HALT	闪存被断电。 外设时钟被关闭。输入时钟被禁用。	4mA	5mA	19μA	21μA	46μA	50μA

(1) I_{DDIO} 电流取决于 I/O 引脚上的电力负载。

(2) 数适用于常温和标称电压。

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。

5.4.1 减少流耗

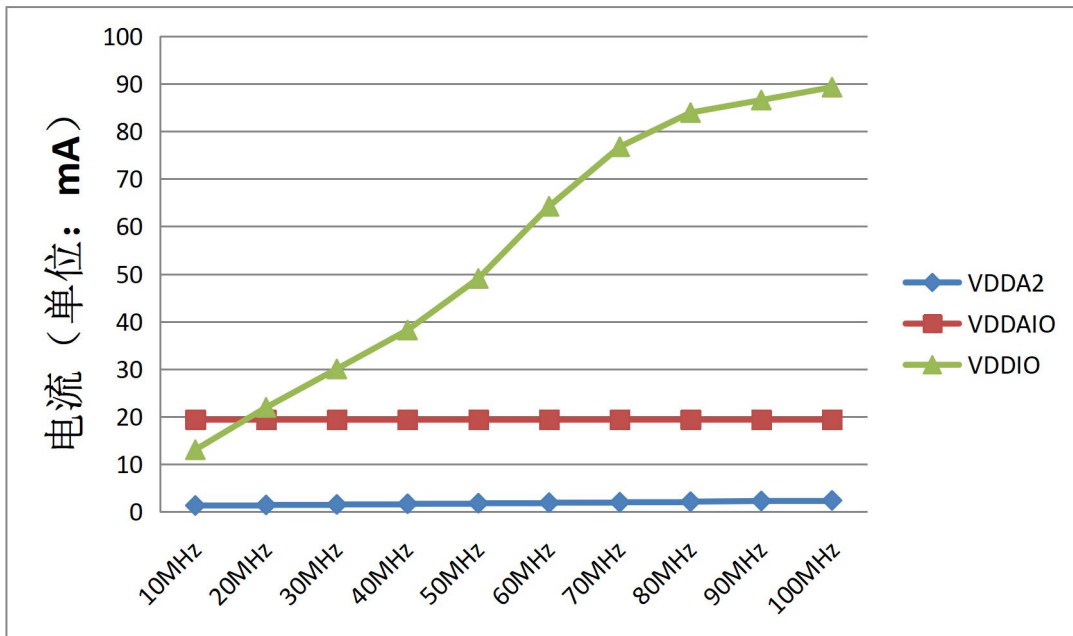
ADP32F08包含一个减少器件流耗的方法。由于每一个外设单元有一个独立的时钟启用位，通过关闭不在指定应用中使用的任一外设模块时钟可显著减少流耗。此外，可利用这三种低功耗模式中的任意一种来进一步减少流耗。[表5-2](#)表明了通过关闭到不同外设的时钟所实现的流耗减少的典型值。

表 5-2 不同外设的典型流耗 (100MHz) ⁽¹⁾

外设模块	I _{DD} 电流减少(mA) ⁽²⁾
ADC	2.3 ⁽³⁾
I ² C	1.5
ePWM	1.9
eCAP	0.8
eQEP	1.2
SCI	1.4
SPI	1.4
eCAN	4.0

- (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，依照模块引用电流。例如，为 ePWM 所引出的 1.9mA 电流是用于一个 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_{DDA}) 模拟部分的电流。

5.4.2 流耗图



- (1) 横坐标标志 SYSCLKOUT
- (2) 操作条件: 温度 27°C, VDDAIO、VDDA2 和 VDDAIO 3.3V; 所有外设时钟都打开

图 5-1 典型运行电流与频率间的关系 (ADP32F08)

5.5 针对 DSP 的无信号缓冲的仿真器连接

图 5-2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，那么仿真信号必须被缓冲。如果距离小于6英寸，通常无需缓冲。图5-2显示了较简单、无缓冲的情况。对于上拉/下拉电阻器的值，请见引脚说明部分。

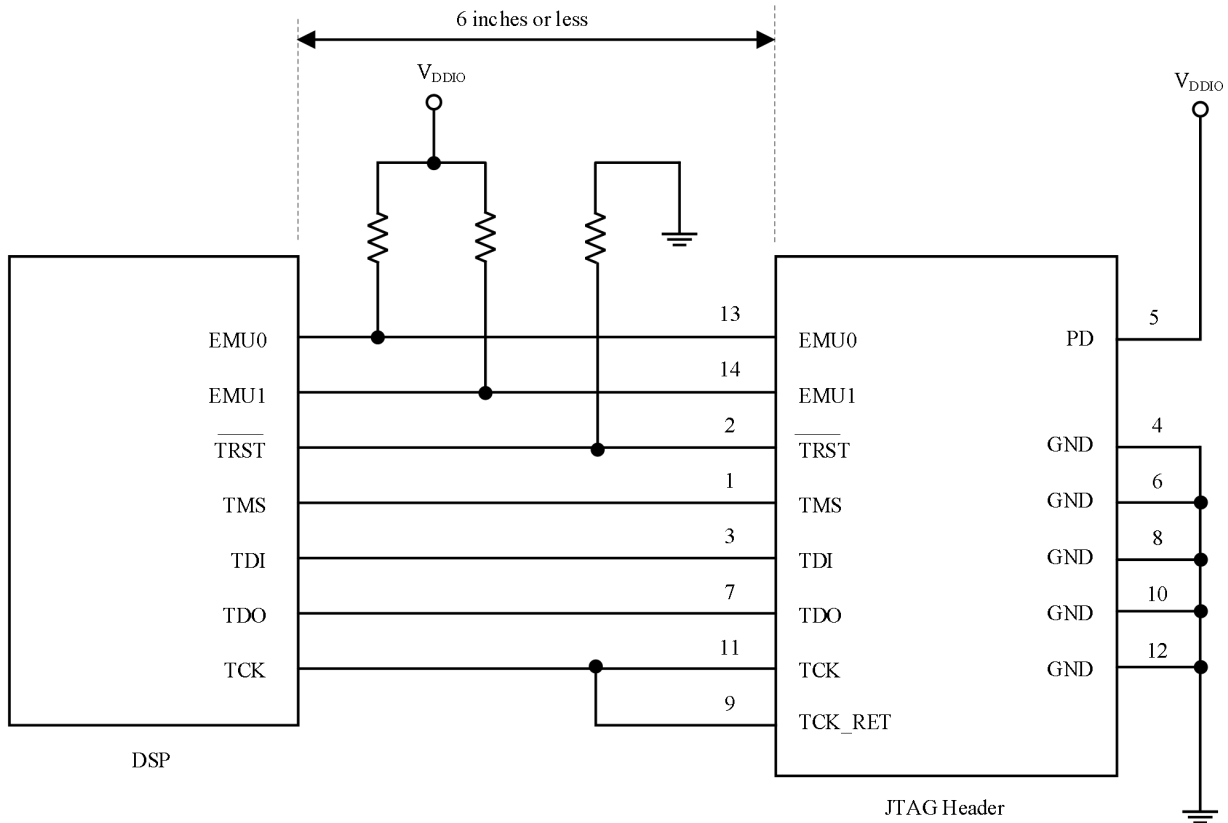


图 5-2 针对 DSP 的无信号缓冲的仿真器连接

5.6 时序参数符号

所用的时序参数符号按照 JEDEC 标准 100 创建。为了缩短符号，一些引脚的名称和其它相关的术语名已经按如下方法缩减：

小写下标的含意:
字母和符号的含意:

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)	

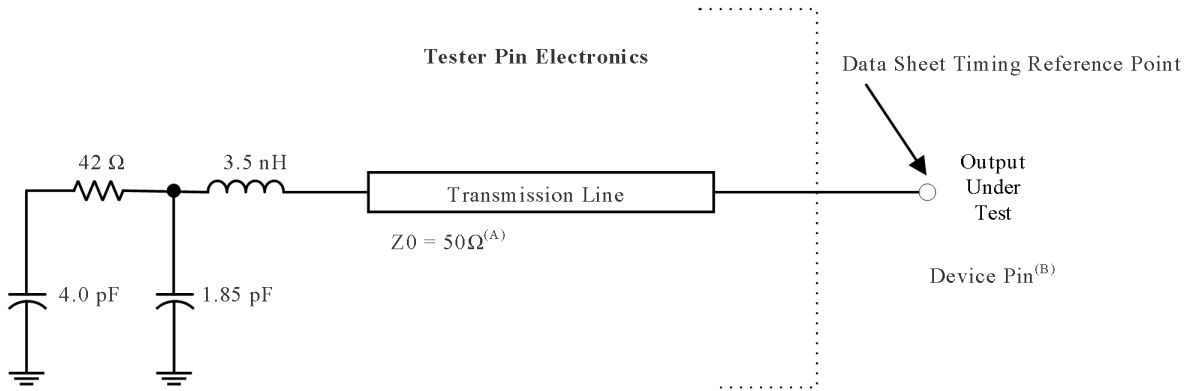
5.6.1 定时参数的通用注释

所有 ADP32F08 器件的输出信号 (包括 XCLKOUT) 取自一个内部时钟, 这样, 对于一个指定半周期的所有输出转换在一个互相之间相对最小转换率时发生。

这个显示在下面时序图中的信号组合也许不一定代表真实的周期。对于真实周期范例, 请参见本文档的合适周期说明部分。

5.6.2 测试负载电路

这个测试负载电路用于测量这个文档中提供的所有开关特性。



- A. 使用一个器件引脚上 < 4伏特每纳秒(4V/ns) 的输入转换率对这个数据表中的输入要求进行测试。
- B. 此数据表在器件引脚上提供时序。对于输出时序分析，必须将测试器引脚电子特性和其传输线路效应考虑在内。带有2ns或者更长时间延迟的传输线路可被用于生成所需的传输线路效应。传输线路只用于一个负载。无需从数据表时序中增加或者减少传输线路延迟 (2ns 或者更长)。

图 5-3 3.3V 测试负载电路

5.6.3 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5-3](#)列出了多种时钟的周期时间。

表 5-3 ADP32F08 时钟表和命名规则 (100MHz 器件)

		最小值	标称值	最大值	单位
片载振荡器时钟	$t_c(\text{OSC})$, 周期时间	28.6		50	ns
	频率	20		35	MHz
XCLKIN ⁽¹⁾	$t_c(\text{CI})$, 周期时间	10		250	ns
	频率	4		100	MHz
SYSCLKOUT	$t_c(\text{SCO})$, 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⁾	$t_c(\text{HCO})$, 周期时间	10	20 ⁽³⁾		ns
	频率		50 ⁽³⁾	100	MHz
LSPCLK ⁽²⁾	$t_c(\text{LCO})$, 周期时间	10	40 ⁽³⁾		ns
	频率		25 ⁽³⁾	100	MHz
ADC 时钟	$t_c(\text{ADCCLK})$, 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，这个值为默认复位值。

5.7 时钟要求和特性

表 5-4 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				MHz
	谐振器(X1/X2)	20		35	
	晶振(X1/X2)	20		35	
	外部振荡器/时钟源 (XCLKIN或者 X1引脚)	4		100	
f _i	跛行模式 SYSCLKOUT频率范围 (/2启用时)			1-5	MHz

表 5-5 XCLKIN ⁽¹⁾ 时序需求—启用 PLL

编号	参数	描述	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		6	ns
C10	t _r (CI)	上升时间, XCLKIN		6	ns
C11	t _w (CIL)	脉冲持续时间, XCLKIN 低电平是 t _c (OSCLK)的一部分的时间	45	55	%
C12	t _w (CIH)	脉冲持续时间, XCLKIN 高电平是 t _c (OSCLK)的一部分的时间	45	55	%

(1)这也被应用到 X1 引脚。

表 5-6 XCLKIN(1)时序需求 - PLL 被禁用

编号	参数	描述	最大值	最小值	单位	
C8	t _c (CI)	周期时间, XCLKIN	100MHz 器件	10	250	ns
			60MHz 器件	16.67		
C9	t _f (CI)	下降时间, XCLKIN	高达 20MHz		6	ns
			20MHz 至 100MHz		2	ns
C10	t _r (CI)	上升时间, XCLKIN	高达 20MHz		6	ns
			20MHz 至 100MHz		2	ns
C11	t _w (CIL)	脉冲持续时间, XCLKIN 低电平为 t _c (OSCLK)的一部分的时间	45	55	%	
C12	t _w (CIH)	脉冲持续时间, XCLKIN 高电平为 t _c (OSCLK)的一部分	45	55	%	

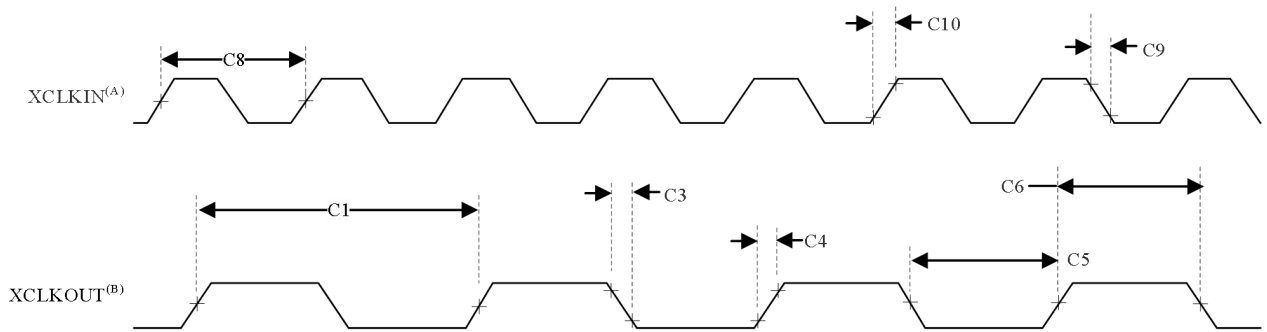
(1)这也被应用到 X1 引脚。

表 3-15 为可能的配置模式。

表 5-7 XCLKOUT 开关特性 (PLL 旁通或者被禁用) ⁽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	100MHz器件	10	2	2
			60MHz器件			
C3	$t_f(XCO)$	下降时间, XCLKOUT		2		ns
C4	$t_r(XCO)$	上升时间, XCLKOUT		2		ns
C5	$t_w(XCOL)$	脉冲持续时间, XCLKOUT低电平的时间	H-2	H+2	H+2	ns
C6	$t_w(XCOH)$	脉冲持续时间, XCLKOUT高电平的时间	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 40pF 的负载。
- (2) $H=0.5t_c(XCO)$
- (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。所显示的波形只用于说明时序参数并且根据实际配置会有所不同。
- B. XCLKOUT 被配置成反映 SYSCLKOUT。

图 5-4 时钟时序

5.8 电源排序

对于不同电源引脚的加电/断电序列无特别要求以确保针对所有模块的正确复位。然而, 如果 I/O 引脚的电平移动输出缓冲器中的 3.3V 晶体管在 1.8V 晶体管之前加电, 输出缓冲器有可能打开, 这会在加电期间导致引脚上的毛刺脉冲。为了避免这一运行状态, 给 VDD (内核电压) 引脚加电应早于对 VDDIO (输入/输出电压) 引脚供电, 或者与之同时, 以确保 VDD 引脚在 VDDIO 引脚达到 0.7V 之前达到 0.7V。

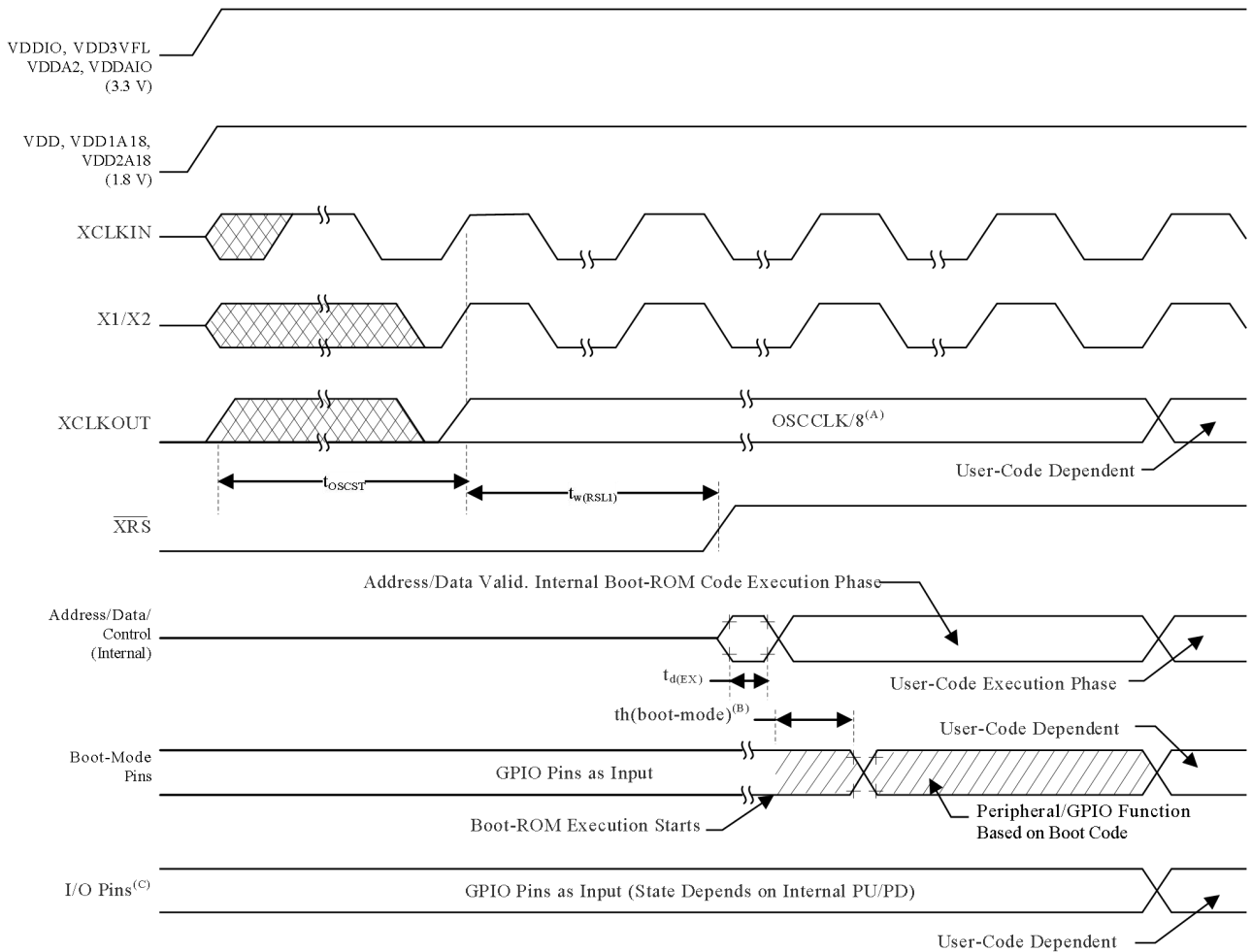
对于 XRS 引脚的要求:

1. 加电期间, 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 5-8)。这使得整个器件从一个已知的条件启动。
2. 断电期间, XRS 引脚必须至少在 VDD 达到 1.5V 之前的 $8\mu s$ 内被下拉至低电平。这样提高了闪

存的可靠性。

在为器件加电之前，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 之内应用于未加电器件的引脚上的电压会以一种无意的方式偏置内部 p-n 接头并产生无法预料的结果。

5.8.1 电源管理和监控电路解决方案



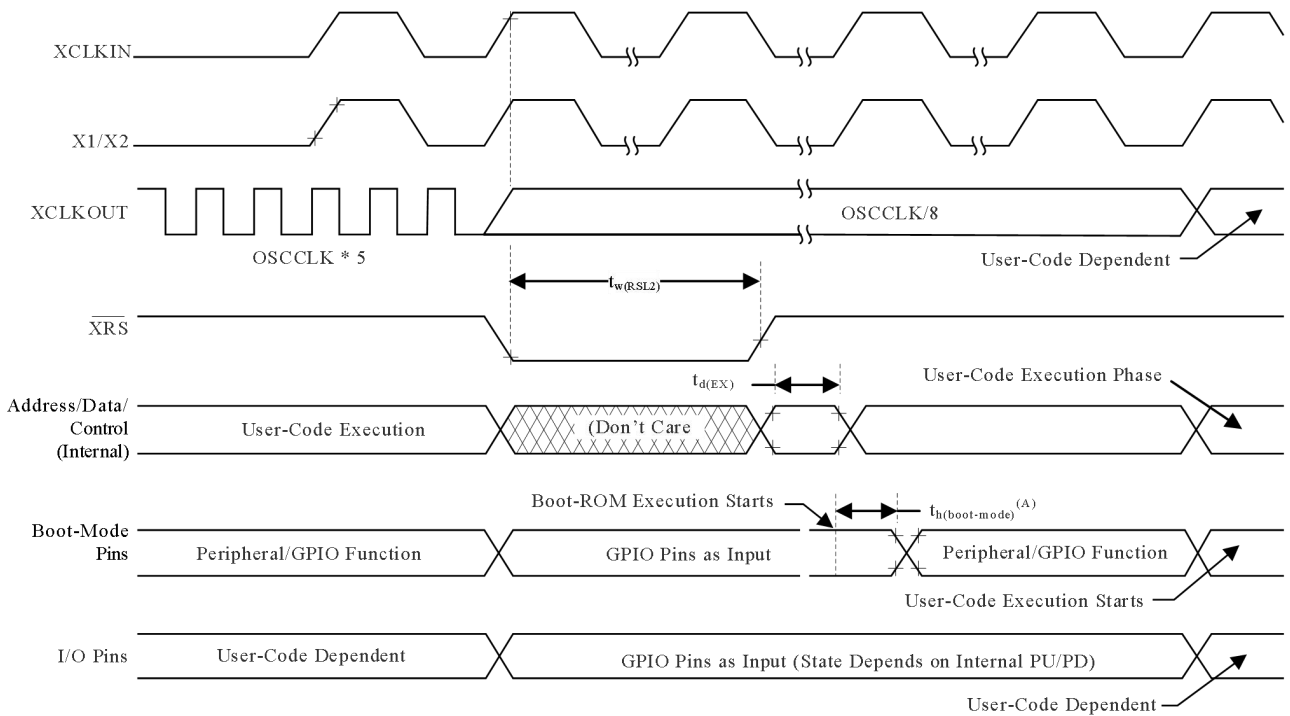
- 加电时，SYSCLKOUT 为 $OSCCLK/2$ 。由于 XCLK寄存器内的 XCLKOUTDIV位出现时的状态为复位状态0，SYSCLKOUT在出现在 XCLKOUT上之前被进一步 4分频。这就是在这个阶段 $XCLKOUT=OSCCLK/8$ 的原因。
- 复位后，引导 ROM代码采样引导模式引脚。基于引导模式引脚的状态，引导代码向目的内存或者引导代码函数下达分支指令。如果引导 ROM代码在加电条件后（在调试器环境中）执行代码，引导代码执行时间由当前的 SYSCLKOUT的速度而定。SYSCLKOUT将基于用户环境并可在 PLL启用或者不启用时使用。
- 对于加电期间，确保一个 GPIO引脚为高阻抗状态的要求。

图5-5 上电复位

表 5-8 复位 $\overline{\text{XRS}}$ 时序要求

		最小值	标称值	最大值	单位
$t_w(\text{RSL1})^{(1)}$	脉冲持续时间, 稳定 XCLKIN到 $\overline{\text{XRS}}$ 高电平的时间	$8t_{c(\text{OSCCLK})}$			周期
$t_w(\text{RSL2})$	脉冲持续时间, $\overline{\text{XRS}}$ 低电平的时间	热复位	$8t_{c(\text{OSCCLK})}$		周期
$t_w(\text{WDRS})$	脉冲持续时间, 由安全装置生成复位脉冲的时间		$512t_{c(\text{OSCCLK})}$		周期
$t_d(\text{EX})$	延迟时间, $\overline{\text{XRS}}$ 高电平后, 地址/数据有效的时间		$32t_{c(\text{OSCCLK})}$		周期
$t_{\text{OSCST}}^{(2)}$	振荡器启动时间	1	10		ms
t_h (引导模式)	引导模式引脚的保持时间	$200t_{c(\text{OSCCLK})}$			周期

- (1) 除了 $t_w(\text{RSL1})$ 环境, $\overline{\text{XRS}}$ 必须在 V_{DD} 达到 1.5V 之后的至少 1ms 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代码在加电条件后 (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5-6 热复位

图5-8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。然后写入 0x0008 到 PLLCR。就在 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，OSCCLKx4。

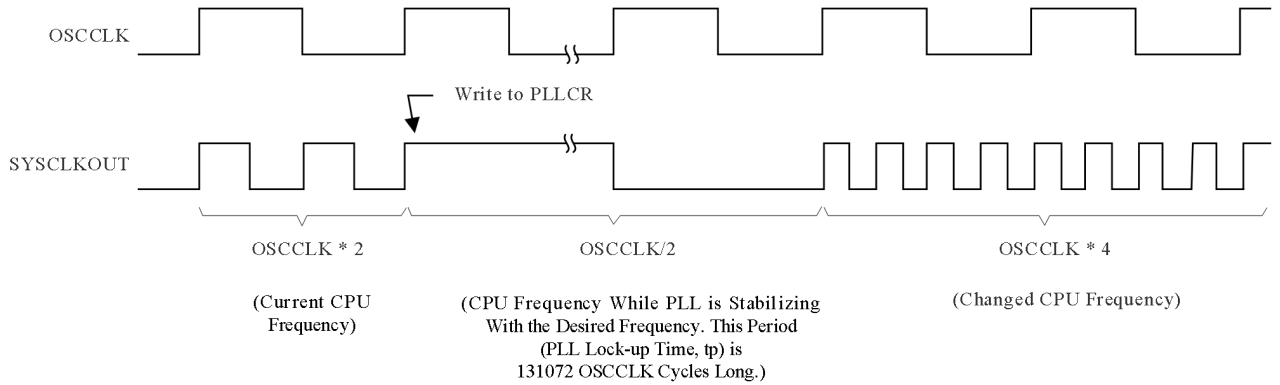


图5-7 写入 PLLCR 寄存器所产生的效果的示例

5.9 通用输入/输出 (GPIO)

5.9.1 GPIO - 输出时序

表5-9 通用输出开关特性

在推荐的运行条件下（除非额外注明）

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

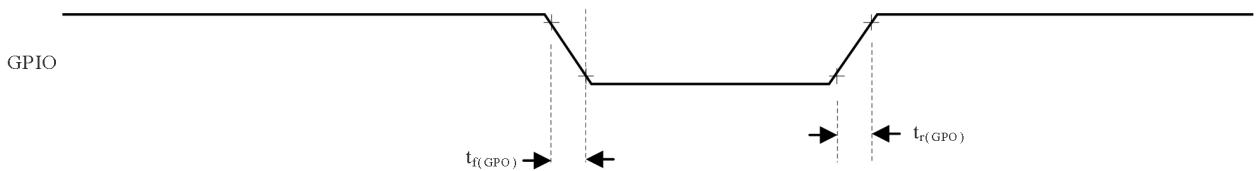
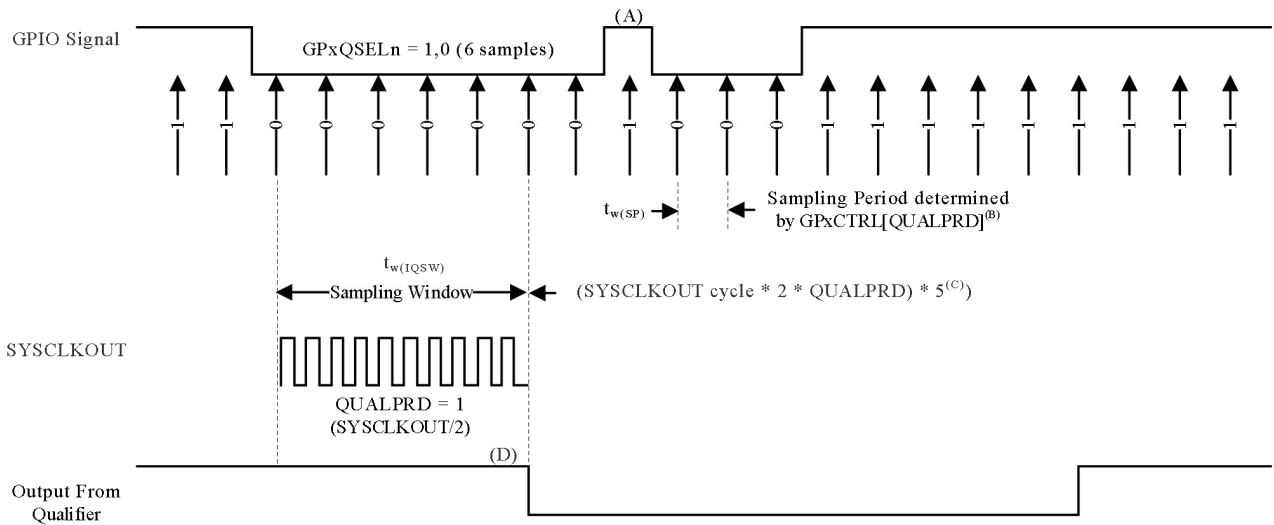


图 5-8 通用输出时序

5.9.2 GPIO - 输入时序



- 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 00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 13SYSCLKOUT宽的脉冲将确保可靠识别。

图 5-9 采样模式

表 5-10 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) $t_{w(GPI)}$, 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。

5.9.3 针对输入信号的采样窗口宽度

下面的部分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 = $SYSCLKOUT/(2*QUALPRD)$

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 个样本的限定

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 个样本的限定

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

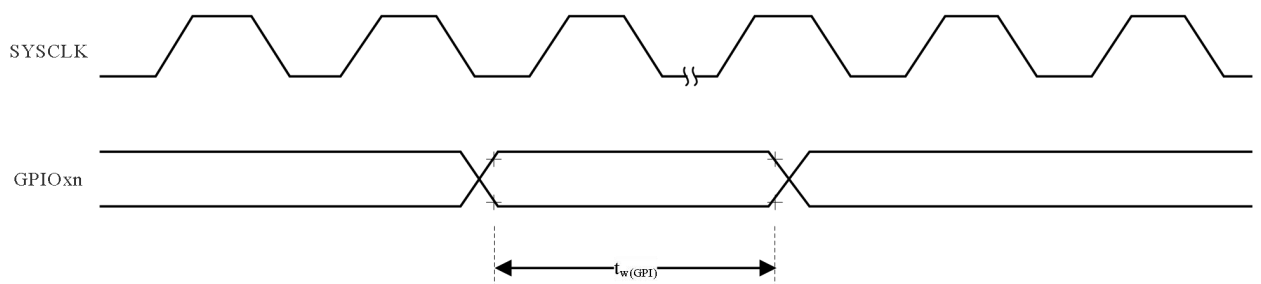


图5-10 通用输入时序

5.9.4 低功耗唤醒时序

表 5-11 显示时序要求，表 5-12 显示了开关特性，而图 5-11 显示了 IDLE 模式下的时序图。

表 5-11 IDLE 模式时序要求⁽¹⁾

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

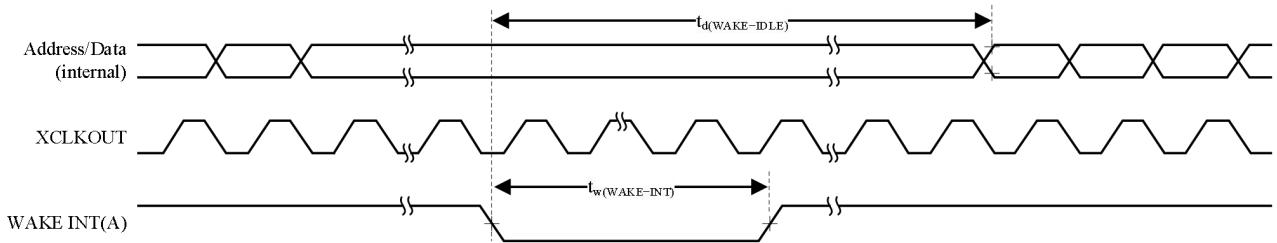
(1) 对于输入限定器参数的说明, 请见表 5-15。

表 5-12 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	
$t_d(\text{WAKE-IDLE})$	延迟时间, 外部唤醒信号到程序执行重新开始的时间 ⁽²⁾				周期	
	• 从闪存唤醒 – 激活状态中的闪存模块	无输入限定器			$20t_{c(\text{SCO})}$	周期
		带有输入限定器			$20t_{c(\text{SCO})} + t_w(\text{IQSW})$	
	• 从闪存唤醒 – 睡眠状态中的闪存模块	无输入限定器			$1050t_{c(\text{SCO})}$	周期
		带有输入限定器			$1050t_{c(\text{SCO})} + t_w(\text{IQSW})$	
	• 从 SARAM 中唤醒	无输入限定器			$20t_{c(\text{SCO})}$	周期
带有输入限定器				$20t_{c(\text{SCO})} + t_w(\text{IQSW})$		

(1) 对于输入限定器器参数的说明, 请见表 5-15。

(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 $\overline{\text{XRS}}$ 。

图 5-11 IDLE 推入和推出时序

表5-13 STANDBY 模式时序要求

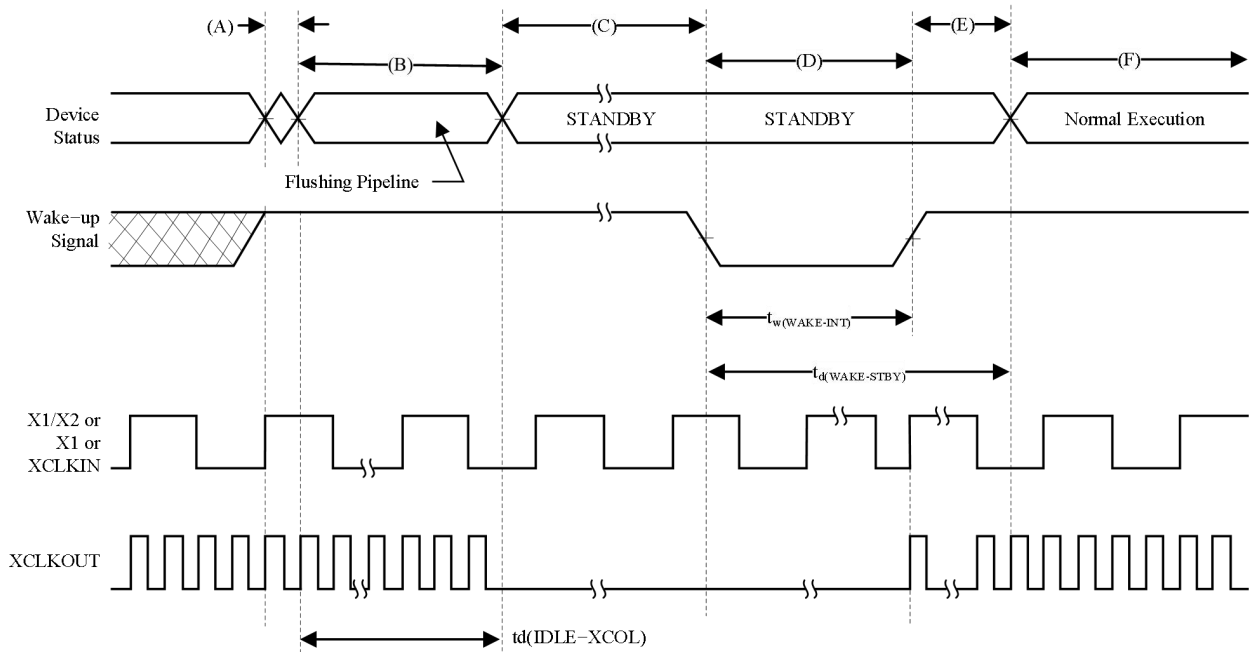
		最小值	标称值	最大值	单位
$t_w(\text{WAKE-INT})$ 脉冲持续时间, 外部唤醒信号的时间	无输入限定器	$3t_{c(\text{OSCCLK})}$			周期
	带有输入限定器 ⁽¹⁾	$(2 + \text{QUALSTDBY}) * t_{c(\text{OSCCLK})}$			

(1) QUALSTDBY 是一个 LPMCRO 寄存器内的 6 位字段。

表 5-14 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 被保持大约 32 个周期 (如果 CLKINDIV=0) 或者 64 个周期 (如果 CLKINDIV=1)。这个延迟使得 CPU 管线和其它等待的操作被适当清空。
- C. 到外设的时钟被关闭。然而, PLL 和安全装置并未关闭。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)。

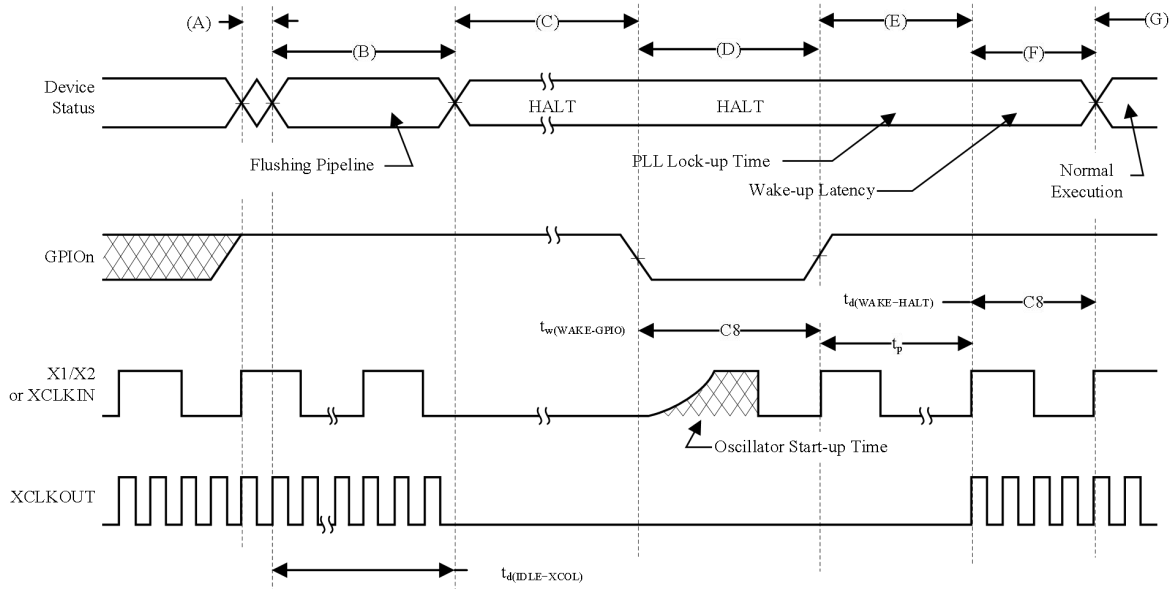
图 5-12 STANDBY 推入和推出时序图

表 5-15 HALT 模式时序要求

	最小值	标称值	最大值	单位
$t_{w(WAKE-GPIO)}$ 脉冲持续时间, GPIO 唤醒信号的时间	$t_{oscst} + 2t_c(OSCCLK)$			周期
$t_{w(WAKE-XRS)}$ 脉冲持续时间, XRS唤醒信号的时间	$t_{oscst} + 8t_c(OSCCLK)$			周期

表 5-16 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



- 执行IDLE 指令以将器件置于 HALT模式。
- PLL块响应 HALT信号。在振荡器被关闭并且到内的 CLKIN的被停止前, SYSCLKOUT被保持大约 32个周期 (如果CLKINDIV=0) 或者 64个周期 (如果 CLKINDIV=1)。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。
- 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洁净的时钟信号。由于 GPIO引脚的下降边沿异步开始唤醒过程, 应该注意在进入和处于 HALT模式期间保持一个低噪声环境。
- 一旦振荡器已经稳定, PLL 锁序列被启动, 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。
- 到内核的时钟和外设被启用。现在退出 HALT模式。一个延迟后, 这个器件将相应此中断 (如果被启用的话)。
- 正常运行重新开始。

图 5-13 使用 GPIOn 的 HALT 唤醒

5.10 增强型控制外设

5.10.1 增强型脉宽调制器 (ePWM) 时序

PWM 是指ePWM1-7上的 PWM 输出。[表 5-17](#)显示了 PWM 时序要求,[表 5-18](#)显示了其开关特性。

表 5-17ePWM 时序要求⁽¹⁾

参数		测试条件	最小值	最大值	单位
$t_{w(SYCNIN)}$	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	$1t_{c(SCO)}+t_{w(IQSW)}$		周期

表 5-18ePWM 开关特性

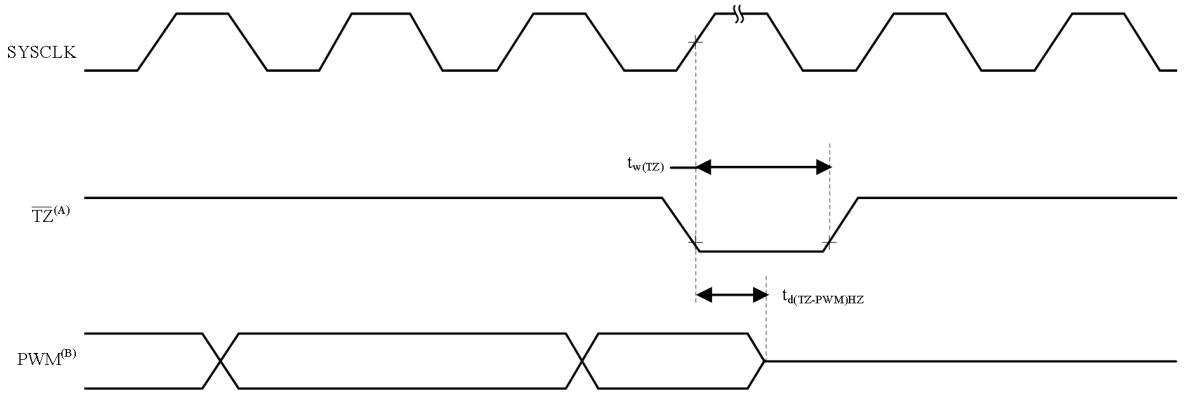
参数		测试条件	最小值	最大值	单位
$t_w(PWM)$	脉冲持续时间, PWMx输出高电平/低电平的时间		20		ns
$t_w(SYNCOUT)$	同步输出脉冲宽度		$8t_{c(SCO)}$		周期
$t_d(PWM)tza$	延迟时间, 触发输入有效到 PWM 强制高电平的时间; 延迟时间, 触发输入有效到 PWM 强制低电平的时间;	无引脚负载		25	ns
$t_d(TZ-PWM)HZ$	延迟时间, 触发输入有效至 PWM 高阻抗 (Hi-Z) 的时间			20	ns

5.10.2 可编程控制故障区输入时序

表 5-19 可编程控制故障区输入时序要求⁽¹⁾

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

(1) 要获得输入限定符参数的解释说明, 请见表 5-15。



A. \overline{TZ} -TZ1, TZ2, TZ3, TZ4, TZ5, TZ6

B. PWM 是指器件内的所有 PWM 引脚。 \overline{TZ} 为高电平之后的 PWM 引脚的状态，取决于 PWM 恢复软件。

图5-14 PWM Hi-Z 特性

5.10.3 高分辨率 PWM (HRPWM) 时序

表 5-20 显示了高分辨率 PWM 的开关特性。

表 5-20 SYSCLKOUT=60-100MHz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 步长基于最差情况过程、最大温度和最大电压。MEP 步长将随着低电压和高温度而增加，随着电压和冷却温度而降低。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 步数量。

5.10.4 增强型捕捉 (eCAP) 时序

表 5-21 显示了 eCAP 时序要求，而表 5-22 显示了 eCAP 开关特性。

表 5-21 增强型捕捉 (eCAP) 时序要求⁽¹⁾

参数		测试条件	最小值	最大值	单位
$t_w(\text{CAP})$	捕捉输入脉冲宽度	异步	$2t_{c(\text{SCO})}$		周期
		同步	$2t_{c(\text{SCO})}$		周期
		带有输入限定器	$1t_{c(\text{SCO})} + t_w(\text{IQSW})$		周期

(1) 对于输入限定器参数的说明，请见表 5-15。

表 5-22 eCAP 开关特性

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

5.10.5 增强型正交编码脉冲 (eQEP) 时序

表 5-23显示了 eQEP 时序要求, 而表 5-24显示了 eQEP 开关特性。

表5-23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
$t_w(QEPP)$	异步 ⁽²⁾ /同步	$2t_{c(SCO)}$		周期
	带有输入限定器	$2[1t_{c(SCO)}+t_w(IQSW)]$		周期
$t_w(INDEXH)$	异步 ⁽²⁾ /同步	$2t_{c(SCO)}$		周期
	带有输入限定器	$2t_{c(SCO)}+t_w(IQSW)$		周期
$t_w(INDEXL)$	异步 ⁽²⁾ /同步	$2t_{c(SCO)}$		周期
	带有输入限定器	$2t_{c(SCO)}+t_w(IQSW)$		周期
$t_w(STROBH)$	异步 ⁽²⁾ /同步	$2t_{c(SCO)}$		周期
	带有输入限定器	$2t_{c(SCO)}+t_w(IQSW)$		周期
$t_w(STROBL)$	异步 ⁽²⁾ /同步	$2t_{c(SCO)}$		周期
	带有输入限定器	$2t_{c(SCO)}+t_w(IQSW)$		周期

(1) 要获得输入限定符参数的解释说明, 请见表 5-10。

表5-24eQEP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

5.10.6 ADC 转换开始时序

表5-25 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, ADCSOCA0低电平的时间	$32t_{c(HCO)}$		周期

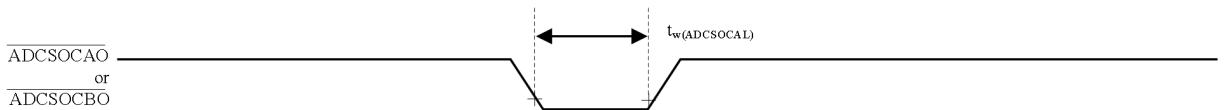


图 5-15 ADCSOCA0或 ADCSOCB0 时序

5.10.7 外部中断时序

表5-26外部中断时序要求⁽¹⁾

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	$1t_{c(SCO)} + t_{w(IQSW)}$		周期

(1)要获得输入限定符参数的解释说明, 请见表 5-10。

(2)这个时序适用于为 ADCSOC 功能性所配置的任一 GPIO 引脚

表5-27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) 要获得输入限定符参数的解释说明, 请见表 5-10。

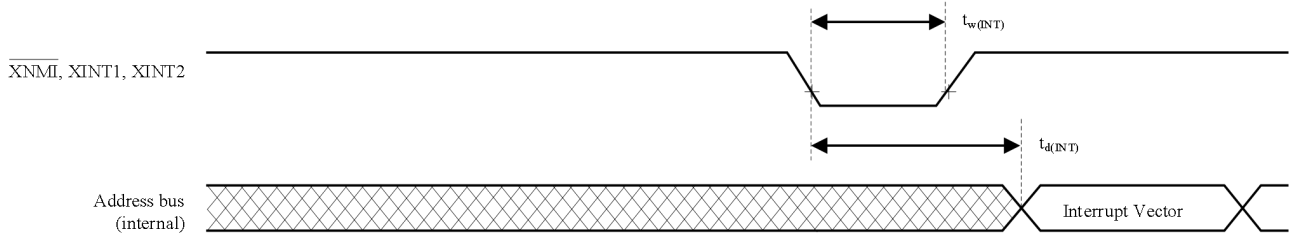


图5-16 外部中断时序

5.10.8 I2C 电气特性和时序

表5-28 I2C时序

	测试条件	最小值	最大值	单位
f_{SCL} SCL 时钟频率	I2C 时钟模块频率介于 7MHz 和 12MHz 之间并且 I2C 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 吸收电流	0	0.4	V
t_{LOWSCL} 时钟的低周期	I2C 时钟模块频率介于 7MHz 和 12MHz 之间并且 I2C 预分频器和时钟分频器寄存器被适当配置	1.3		μs
$t_{HIGHSCL}$ 时钟的高周期	I2C 时钟模块频率介于 7MHz 和 12MHz 之间并且 I2C 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

5.10.9 同步串行通信接口(SPI) 主机模式时序

表5-29列出了主机模式时序（时钟相位= 0）而表5-30列出了时序（时钟相位=1）。图5-18和图5-19显示了时序波形。

表5-29 SPI 主机模式外部时序（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)	26		26		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)	26		26		
9	$t_{v(SPCL-SOMI)M}$	有效时间, SPICLK 低电平之后 SPISOMI 数据有效的的时间 (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) 主机/从机位(SPICTL.2) 被设定, 而时钟相位的位(SPICTL.3) 被清除。

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

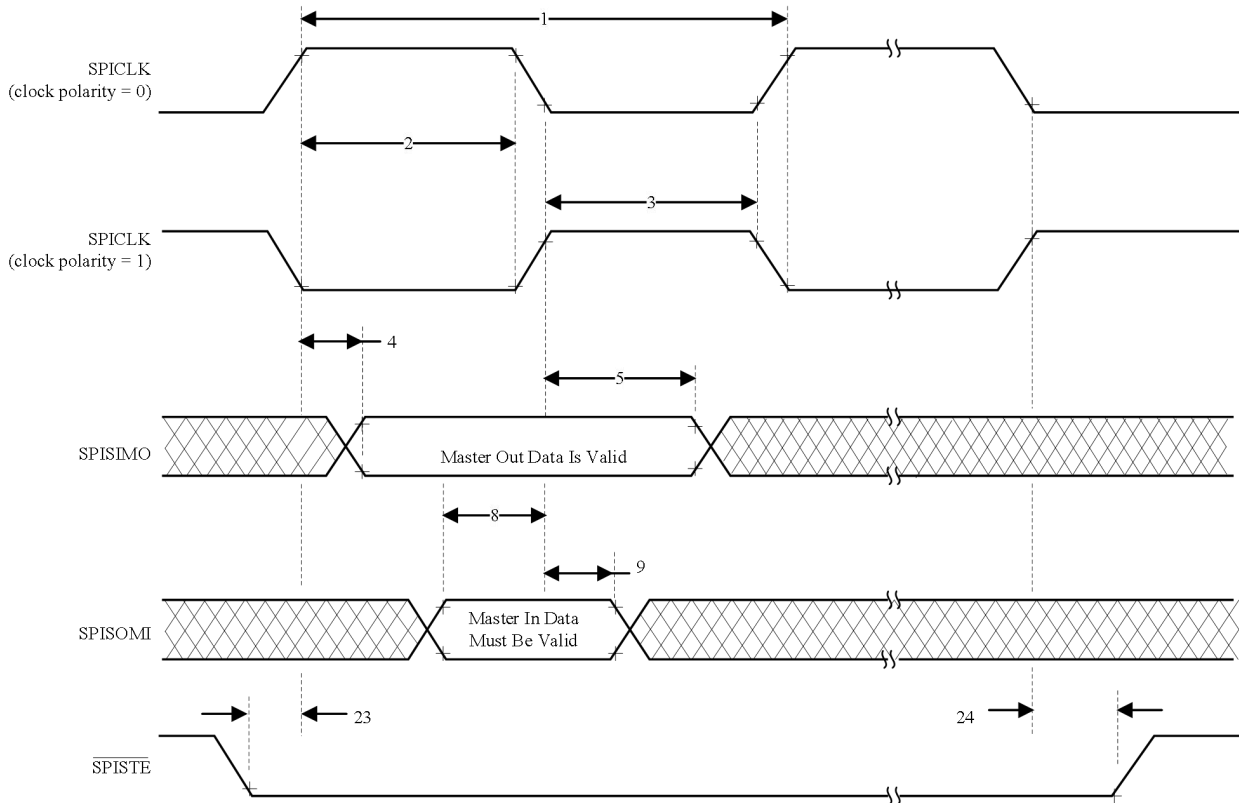
(3) 内部时钟预分频器必须被调整, 这样的话, SPI 时钟速度被限制在下列 SPI 时钟速率上:

主机模式发射最大值 25MHz, 主机模式接收最大值 12.5MHz

从机模式发送最大值 12.5MHz, 从机模式接收最大值12.5MHz。

(4) $t_{c(LCO)} = \text{LSPCLK 周期时间}$

(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。

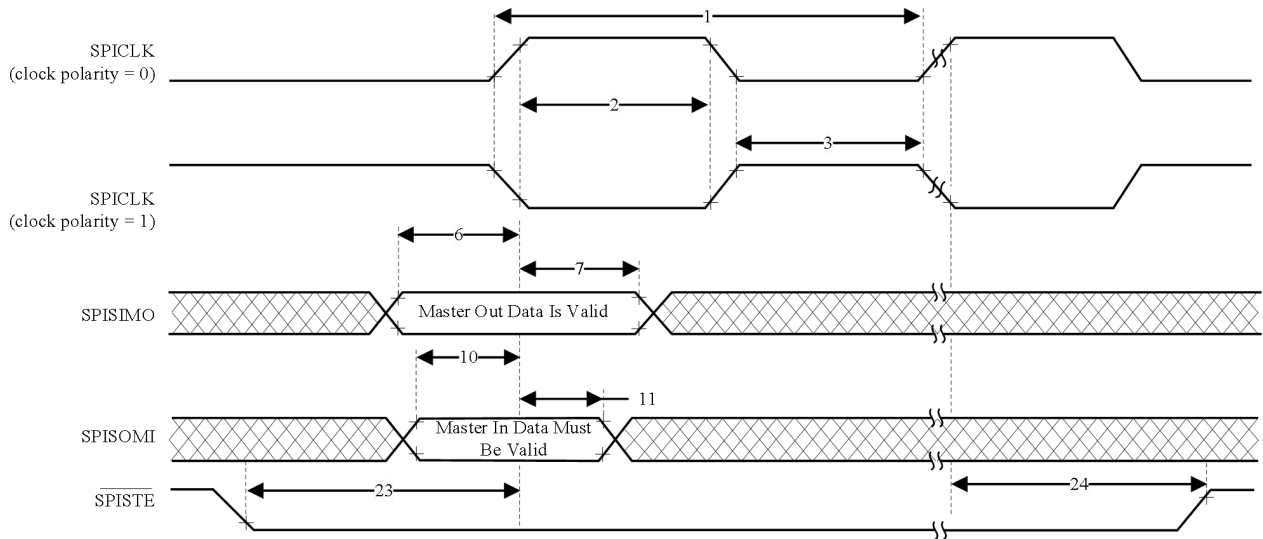


A. 在主机模式下，在有效的 SPI 时钟边沿之前 $0.5t_{c(SPC)}$ （最小值）， \overline{SPISTE} 变为有效。在字的尾端， \overline{SPISTE} 在接收到最后一个数据位的边沿(SPICLK)之后 $0.5t_{c(SPC)}$ 将变为无效，除非 \overline{SPISTE} 在 FIFO 和非 FIFO 模式中的背靠背传送字间保持有效。

图5-17 SPI 主机模式外部时序 (时钟相位= 0)

表 5-30 SPI 主机模式外部时序 (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	$t_{c(SPCM)}$	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)	$0.5t_{c(SPCM)}-10$	$0.5t_{c(SPCM)}$	$0.5t_{c(SPCM)}-0.5t_{c(LCO)}-10$	$0.5t_{c(SPCM)}-0.5t_{c(LCO)}$	ns
	$t_{w(SPCL)M}$	脉冲持续时间, SPICLK 低电平的时间 (时钟极性= 1)	$0.5t_{c(SPCM)}-10$	$0.5t_{c(SPCM)}$	$0.5t_{c(SPCM)}-0.5t_{c(LCO)}-10$	$0.5t_{c(SPCM)}-0.5t_{c(LCO)}$	
3	$t_{w(SPCL)M}$	脉冲持续时间, SPICLK 低电平的时间 (时钟极性= 0)	$0.5t_{c(SPCM)}-10$	$0.5t_{c(SPCM)}$	$0.5t_{c(SPCM)}+0.5t_{c(LCO)}-10$	$0.5t_{c(SPCM)}+0.5t_{c(LCO)}$	ns
	$t_{w(SPCH)M}$	脉冲持续时间, SPICLK 高电平的时间 (时钟极性= 1)	$0.5t_{c(SPCM)}-10$	$0.5t_{c(SPCM)}$	$0.5t_{c(SPCM)}+0.5t_{c(LCO)}-10$	$0.5t_{c(SPCM)}+0.5t_{c(LCO)}$	
6	$t_{su(SIMO-SPCH)M}$	建立时间, 在SPICLK 高电平之前 SPISIMO 数据有效的的时间 (时钟极性= 0)	$0.5t_{c(SPCM)}-10$		$0.5t_{c(SPCM)}-10$		ns
	$t_{su(SIMO-SPCL)M}$	建立时间, 在 SPICLK 低电平之前 SPISIMO 数据有效的的时间 (时钟极性= 0)	$0.5t_{c(SPCM)}-10$		$0.5t_{c(SPCM)}-10$		
7	$t_{v(SPCH-SIMO)M}$	有效时间, SPICLK 高电平之后 SPISIMO 数据有效的的时间 (时钟极性= 0)	$0.5t_{c(SPCM)}-10$		$0.5t_{c(SPCM)}-10$		ns
	$t_{v(SPCL-SIMO)M}$	有效时间, SPICLK 低电平后, SPISIMO 数据有效的的时间 (时钟极性= 1)	$0.5t_{c(SPCM)}-10$		$0.5t_{c(SPCM)}-10$		
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	$t_{su(SPCH-SOMI)M}$	有效时间, SPICLK 高电平之后 SPISOMI 数据有效的的时间 (时钟极性= 0)	$0.25t_{c(SPCM)}-10$		$0.5t_{c(SPCM)}-10$		ns
	$t_{su(SPCH-SOMI)M}$	有效时间, SPICLK 低电平之后 SPISOMI 数据有效的的时间 (时钟极性= 1)	$0.25t_{c(SPCM)}-10$		$0.5t_{c(SPCM)}-10$		



A. 主机模式下，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 5-18 SPI 主机模式外部时序 (时钟相位= 1)

5.10.10 SPI 从机模式时序

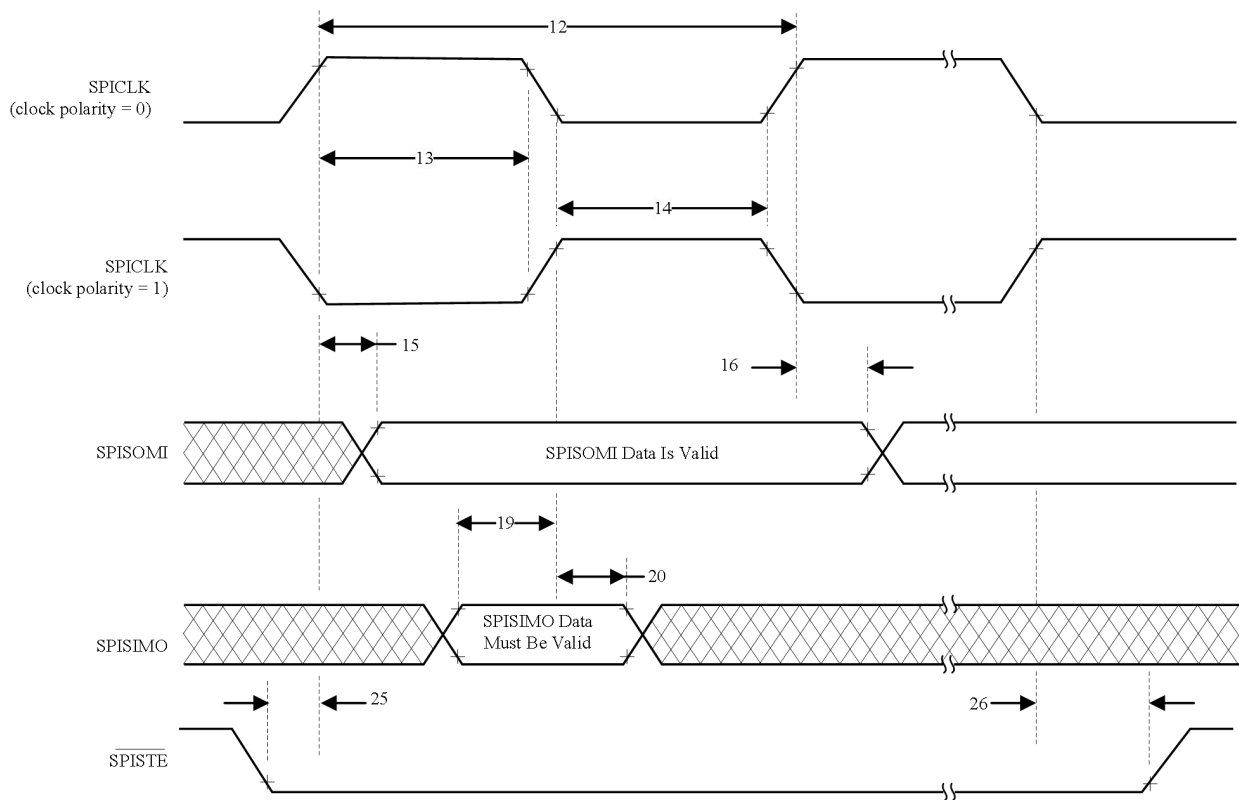
表5-31列出了从机模式外部时序 (时钟相位= 0)，而表5-32 (时钟相位= 1)、图5-20和图5-21显示了时序波形。

表5-31 SPI 从机模式外部时序 (时钟相位= 0) (1) (2) (3) (4) (5)

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

	钟极性= 1)			
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) 主机/从机位(SPICTL.2) 被设定, 而时钟相位的位(SPICTL.3) 被清除。
- (2) $t_{C(SPC)} = \text{SPI 时钟周期时间} = \text{LSPCLK}/4$ 或者 $\text{LSPCLK}/(\text{SPIBRR} + 1)$
- (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:
主机模式发射最大值 25MHz, 主机模式接收最大值 12.5MHz
从机模式发送最大值 12.5MHz, 从机模式接收最大值 12.5MHz。
- (4) $t_{C(LCO)} = \text{LSPCLK 周期时间}$
- (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。



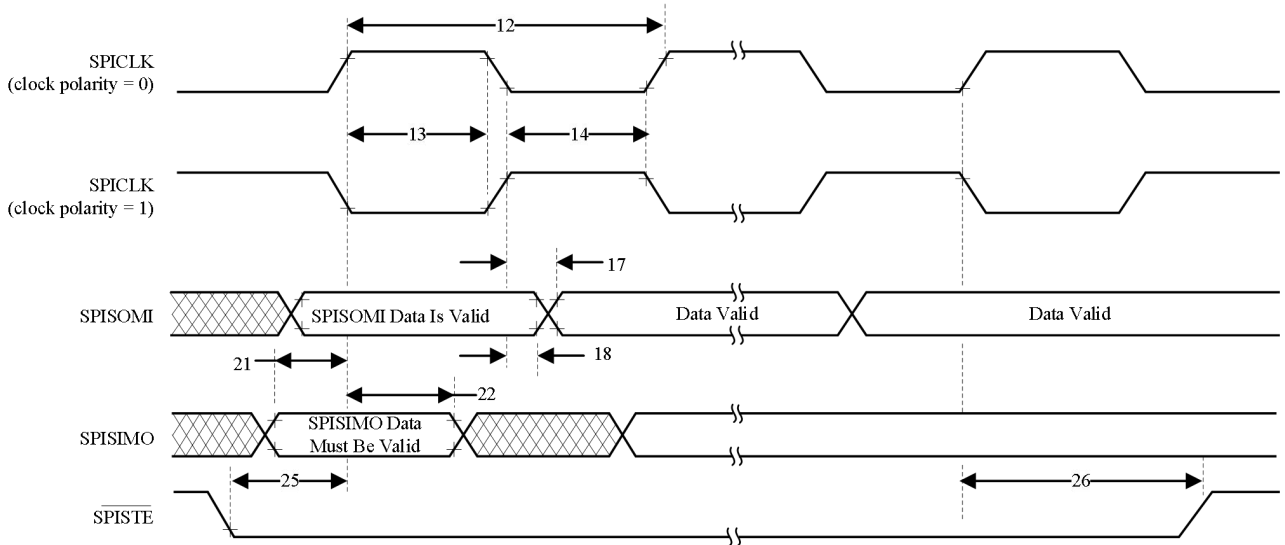
- A. 在从机模式下, $\overline{\text{SPISTE}}$ 信号至少应该在有效SPI 时钟边沿前 $0.5t_{C(SPC)}$ (最小值) 被置为低电平有效并且在接收到最后一个数据位的边沿(SPICLK) 之后保持至少 $0.5t_{C(SPC)}$ 。

图5-19 SPI 从机模式外部时序 (时钟相位= 0)

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

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

- (1) 主机/从机位(SPICTL.2) 被设定, 而时钟相位的位(SPICTL.3) 被清除。
 (2) $t_{c(SPC)} = \text{SPI 时钟周期时间} = \text{LSPCLK}/4$ 或者 $\text{LSPCLK}/(\text{SPIBRR} + 1)$
 (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:
 主机模式发射最大值25Hz, 主机模式接收最大值12.5
 从机模式发送最大值12.5, 从机模式接收最大值12.5。
 (3) 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. 在从机模式下, $\overline{\text{SPISTE}}$ 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5-20 SPI 从机模式外部时序 (时钟相位= 1)

5.10.11 片载模数转换器
表5-33 ADC电气特性 (在推荐的运行条件下) ⁽¹⁾⁽²⁾

参数		最小值	典型值	最大值	单位
DC技术规范					
分辨率		12			位
ADC时钟		0.001		12.5	MHz
精度					
INL (积分非线性)	1-12.5MHzADC时钟 (6.25MSPS)			±7	最低有效位(LSB)
DNL (微分非线性) ⁽³⁾				±2	LSB
偏移误差 ⁽⁴⁾		-60		+60	LSB
带有硬件调整的偏移误差			±8		LSB
带有内部基准的总增益误差 ⁽⁵⁾		-60		+60	LSB
带有外部基准的总增益误差		-60		+60	LSB
通道到通道偏移变化			±8		LSB
通道到通道增益变化			±8		LSB
模拟输入					
模拟输入电压 (ADCIN _x 至 ADCLO) ⁽⁶⁾		0		3	V
ADCLO		-5	0	5	mV
输入电容			10		pF
输入漏电流				±5	μA
内部参考基准⁽⁵⁾					
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
电压差异, ADCREFP- ADCREFM			0.75		V
温度系数			50		PPM/°C
外部电压基准⁽⁷⁾					
V _{ADCREFIN} - 推荐在外部基准	ADCREFSSEL[15:14]=11b		1.024		V

电压输入引脚-ADCREFIN 上提供 0.2%或者更高精度 的基准电压源					
	ADCREFSSEL[15:14]=10b		1.500		V
	ADCREFSSEL[15:14]=01b		2.048		V
AC 技术规格					
SINAD(100kHz)信噪比+ 失真			62.5		dB
SNR(100kHz)信噪比			63.2		dB
THD(100kHz)总谐波失真			-70.3		dB
ENOB(100kHz)有效位数			10.1		位
SFDR(100kHz)无杂散动态 动态范围			73		dB

- (1) 在 12.5MHzADCCLK上测得。
- (2) 这个表中的所有电压相对 VSSA2。
- (3) ADC无丢码。
- (4) 1个 LSB有 $3.0/4096=0.732\text{mV}$ 的加权值。
- (5) 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转换器使用这两个之间的差值作为它的基准。这里列出的针对内部基准的总体增益误差包括温度范围内内部带隙的运动。针对外部基准选项的温度范围内的增益误差将取决于所使用源的温度参数。
- (6) 应用到一个模拟输入引脚上的高于 $VDDA+0.3\text{V}$ 或者低于 $VSS-0.3\text{V}$ 的电压有可能暂时影响另外引脚的转换。为了避免这种情况，模拟输入应该被保持在这些限值内。
- (7) 推荐使用高精度外部基准部件：REF3020/3120或者等效的2.048V基准部件。

5.10.11.1 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，这些通道或者同时采样，或者每次两个通道采样。这些输入为软件可选。

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一个高采样率。转换时间

转换可以在两个不同的转换模式中执行：

顺序采样模式 (SMODE = 0)

同步采样模式 (SMODE=1)

5.10.11.2 ADC 加电控制位时序

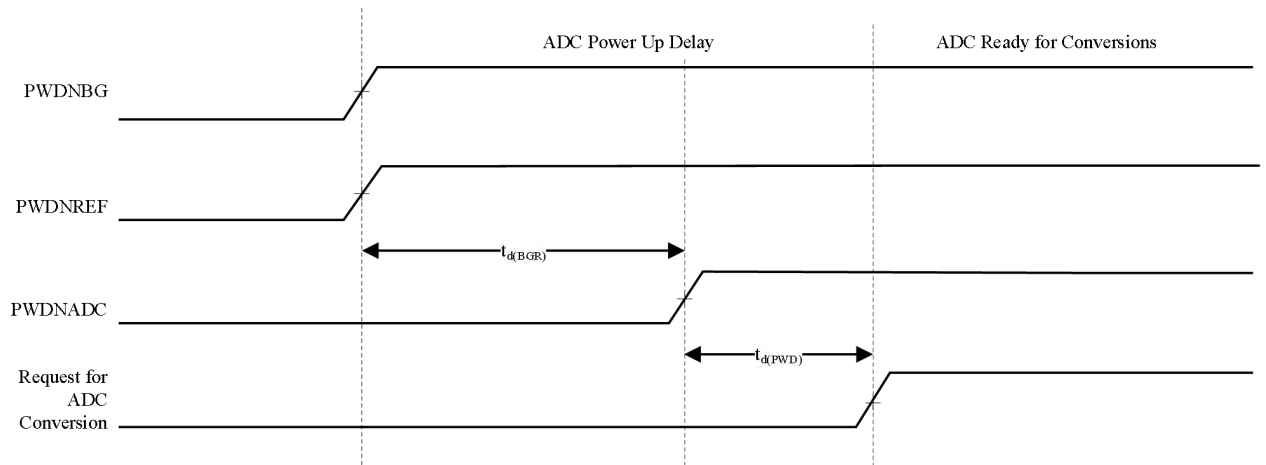


图 5-21 ADC 加电控制位时序

表5-34 ADC 加电延迟

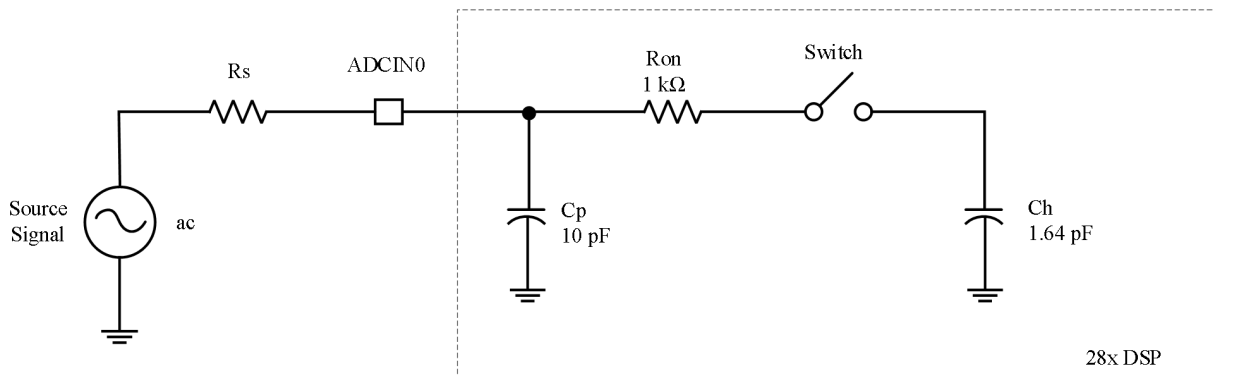
参数 ⁽¹⁾	最小值	典型值	最大值	单位
$t_{d(BGR)}$ 带隙基准稳定所需的延迟时间。ADCTRL3 寄存器的位 7 和 6 (ADCBGRFDN1/0) 在 PWDNADC 位被启用前被设定为 1。			5	ms
$t_{d(PWD)}$ 断电控制稳定所需的延迟时间。带隙基准稳定所需的位延迟时间。ADCTRL3 寄存器的位 7 和 6 (ADCBGRFDN1/0) 在 PWDNADC 位被启用前被设定为 1。ADCTRL3 寄存器的位 5 (PWDNADC) 在任何 ADC 转换启动前被设定为 1。	20	50		μ s
			1	ms

(1) 时序保持与 ADC 模块的兼容性。在首次转换前的 $t_{d(PWD)}$ ms, F08xADC 支持同时驱动所有 3 个位。

表 5-35 针对不同 ADC 配置的流耗 (在 12.5MHzADCCLK 上) (1) (2)

ADC运行模式	条件	V _{DDA18}	V _{DDA3.3}	单位
模式A (运行模式)	<ul style="list-style-type: none"> BG和REF被启用 PWD被禁用 	0.01	23.5	mA
模式B:	<ul style="list-style-type: none"> ADC时钟被启用 BG和REF被启用 PWD被启用 	0.01	4.5	mA
模式C:	<ul style="list-style-type: none"> ADC时钟被启用 BG和REF被启用 PWD被启用 	10	40	μA
模式D:	<ul style="list-style-type: none"> ADC时钟被禁用 BG和REF被启用 PWD被启用 	10	40	μA

- (1) 测试条件:
- (2) SYSCLKOUT=100MHz
- (3) ADC模块时钟 = 12.5MHz
- (4) ADC在模式 A中执行一个所有16通道的连续转换。
- (5) V_{DDA18}包括进入 V_{DD1A18}和 V_{DD2A18}的电流。 V_{DDA3.3}包括进入 V_{DDA2}和 V_{DDA10}的电流。



Typical Values of the Input Circuit Components:

- Switch Resistance (Ron): 1 kΩ
- Sampling Capacitor (Ch): 1.64 pF
- Parasitic Capacitance (Cp): 10 pF
- Source Resistance (Rs): 50 Ω

图 5-22 ADC 模拟输入阻抗模型

5.10.11.3 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，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 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 周期内被设定。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 (最小值) 或者 16 个 ADC 时钟宽 (最大值)。

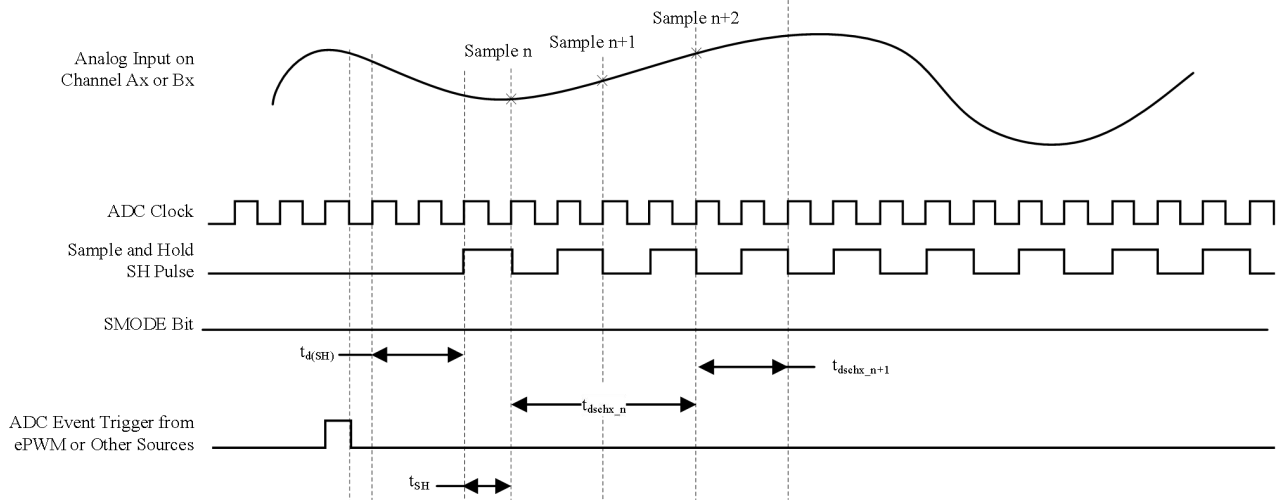


图5-23 顺序采样模式 (单通道) 时序

表 5-36 顺序采样模式时序

		SAMPLE (样本) n	SAMPLEn+1	在 12.5MHz ADCCLK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	$2.5t_{c(ADCCLK)}$			
t_{SH}	采样/保持宽度/采集宽度	$(1 + Acqps) * t_{c(ADCCLK)}$		Acqps=0时为80ns	Acqps值= 0-15ADCTRL1[8:11]
$t_{d(schx_n)}$	结果寄存器出现第一个结果的延迟时间	$4t_{c(ADCCLK)}$		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	

5.11.10.4 同步采样模式 (双通道) (SMODE=1)

在同步模式中, ADC 可在任何一对通道 (A0/B0 至 A7/B7) 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 (最小值) 或者 16 个 ADC 时钟宽 (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，并且不是任何其它组合（例如 A1/B3，等等）。

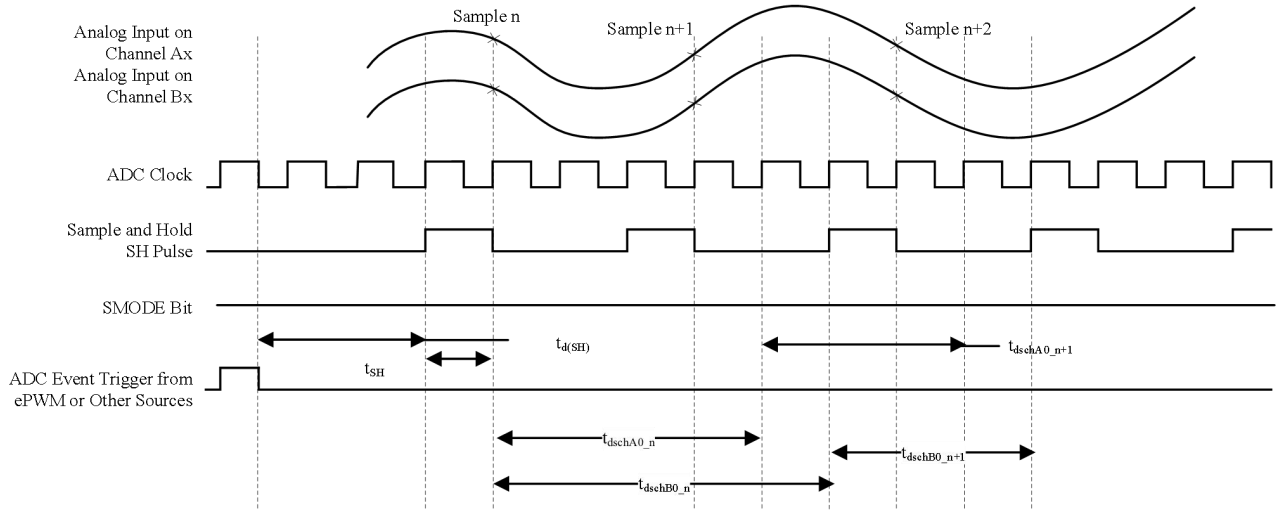


图 5-24 同步采样模式时序

表 5-37 同步采样模式时序

		SAMPLEn	SAMPLEn+1	在 12.5MHzADCCL OCK时钟 上, tc(ADCCLK)=80ns	注 释
td(SH)	从事件触发到采样的延迟时间	2.5tc(ADCCLK)			
tSH	采样/保持宽度/采集宽度	(1 + Acqps) * tc(ADCCLK)		Acqps=0时为}80ns	Acqps值= 0-15ADCTR L1[8:11]
td(schA0_n)	结果寄存器出现第一个结果的延迟时间	4tc(ADCCLK)		320ns	
td(schB0_n)	结果寄存器出现第一个结果的延迟时间	5tc(ADCCLK)		400ns	
td(schA0_n+1)	结果寄存器中出现连续结果的延迟时间		(3+Acqps)*tc(ADCC LK)	240ns	
td(schB0_n+1)	结果寄存器中出现连续结果的延迟时间		(3+Acqps)*tc(ADCC LK)	240ns	

5.11 详细说明

积分非线性

积分非线性是指每个独立代码从零至满刻度所画的一条直线上的偏离。在首次代码转换前，作为零点的点出现1/2 LSB。满刻度点被定义为超过最后一次代码转换的级别1/2 LSB。这个偏离为每一个特定代码的中心到这两个点之间的精确直线的距离。

微分非线性

一个理想ADC 显示分开距离恰好为1 个LSB 的代码转换。DNL 是从这个理想值的偏离。少于±1 LSB 的微分非线性误差可确保无丢码。

零偏移

当模拟输入为零伏时，应当发生主进位转换。零误差被定义为实际转换到那个点的偏离。

增益误差

第一个代码转换应该出现在高于负满刻度的一个模拟值1/2 LSB 上。最后一次转换应该出现在低于标称满刻度的一个模拟值1.5 LSB 上。增益误差是首次和末次代码转换间的实际差异以及它们之间的理想差异。

信噪比+失真(SINAD)

SINAD 是测得的输入信号的均方根值与所有其它低于那奎斯特频率的频谱分量（包括谐波但不包括dc）的均方根总和的比。SINAD 的值用分贝表示。

有效位数(ENOB)

对于一个正弦波，SINAD 可用位的数量表示。使用下面的公式，

$$N = \frac{(\text{SINAD} - 1.76)}{6.02}$$

有可能获得一个用N（位的有效数）表达的性能测量值。因此，对于在给定输入频率上用于正弦波输入的器件的有效位数量可从这个测得的SINAD 直接计算。

总谐波失真(THD)

THD 是前9个谐波分量的均方根总和与测得的输入信号的均方根值的比并表达为一个百分比或者分贝值。

无杂散动态范围(SFDR)

SFDR 是输入信号均方根振幅与峰值寄生信号间以分贝为单位的差异。

5.12 闪存时序

表 5-38 闪存/ OTP 闪存耐受度⁽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)	0°C 至 85°C (环境温度)		20000		周期
N _{OTP} OTP 对于阵列的耐受度 (写入周期)	0°C 至 85°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, 有可能影响耐受数。

表 5-39 100MHz SYSCLKOUT 上的闪存参数

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			38		μs
	32K 扇区			302		ms
擦除时间 ⁽¹⁾	32K 扇区			584		ms
	全片			2.3		s
I _{DD} 擦除/编程周期期间的流耗	擦除			5		mA
	编程			2		mA

(1) 当器件出货时, 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。

表 5-40 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	36		ns
t _{a(fr)} 随机闪存访问时间	36		ns
t _{a(OTP)} OTP 访问时间	60		ns

(1) 随机等待状态必须大于或者等于1, 用于计算表5-42中页等待状态和随机等待状态的公式如下:

$$\text{FlashPage Wait State} = \left\lceil \left(\frac{t_{a(fp)}}{t_{c(SCO)}} - 1 \right) \right\rceil \text{ (四舍五入到下一个最大的整数, 或0, 以最大的数为准)}$$

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

计算表5-42中OTP等待状态的公式如下:

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

表 5-41 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)	闪存页等待状态	闪存随机等待状态 ⁽¹⁾	OTP等待状态
100	10	3	3	5
75	13.33	2	2	4
60	16.67	2	2	3
50	20	1	1	2
30	33.33	1	1	1
25	40	0	1	1
15	66.67	0	1	1
4	250	0	1	1

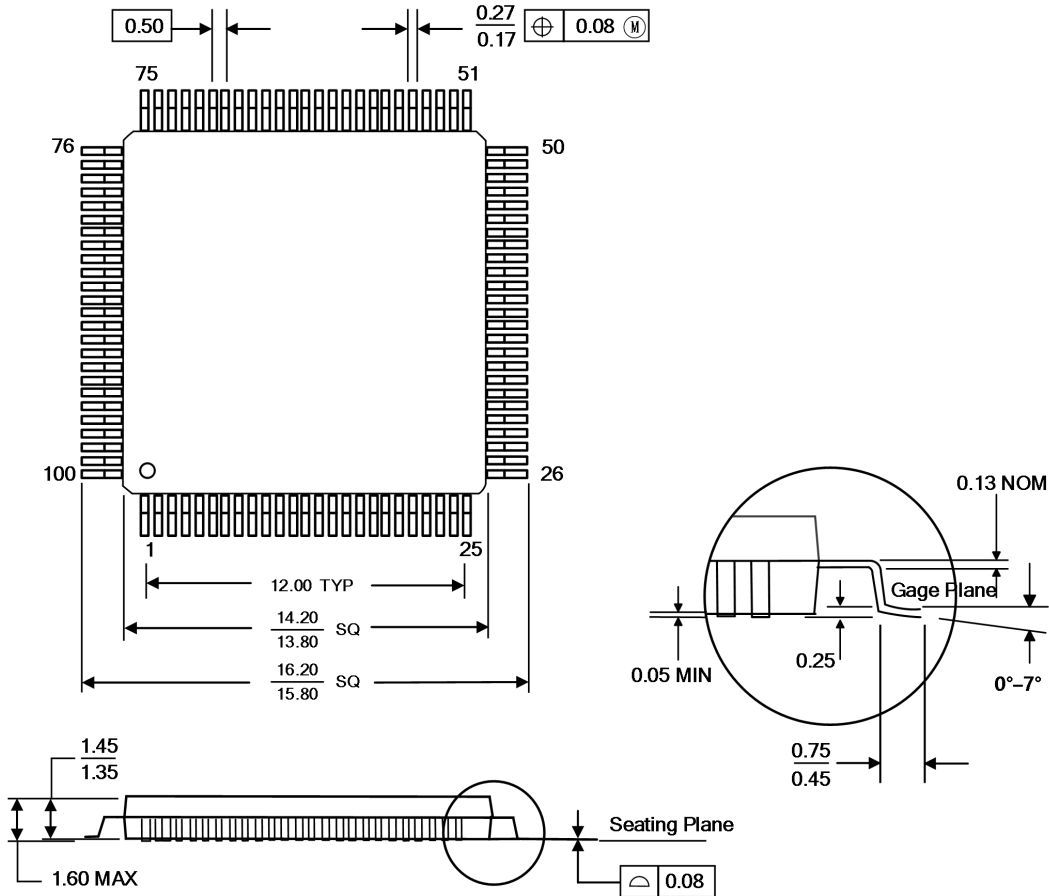
(1) 随机等待状态必须大于或者等于1。

6 机械数据

机械封装图反映了最新发布的用于指定器件的机械数据。

LQFP100

单位: mm



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

公司总部地址: 长沙市高新开发区尖山路 39 号中电软件园总部大楼 10 楼