

# AVP32F069 数字信号处理器

## 数据手册

编号: JXDZ3.590.018SJSC

**Advancechip**



**Electronics**

**湖南进芯电子科技有限公司**

2022年12月

V1.0

# 目 次

1 产品概述 .....	7
1.1 特性 .....	7
1.2 市场应用 .....	9
1.3 产品描述 .....	10
1.4 功能框图 .....	11
1.5 系统模块图 .....	12
1.6 产品编码 .....	13
2 器件对比表 .....	14
3 终端配置和功能 .....	15
3.1 引脚图 .....	15
3.2 内存映射 .....	16
3.3 信号描述 .....	17
3.3.1 信号说明 .....	17
4 技术规格 .....	26
4.1 最大额定值 .....	26
4.2 推荐操作条件 .....	26
4.3 总体流耗 .....	27
4.4 电气特性 .....	28
4.5 无信号缓冲的调试仿真连接 .....	29
4.6 参数信息 .....	30
4.6.1 时序参数符号 .....	30
4.6.2 时序参数通用说明 .....	30
4.7 负载测试电路 .....	30
4.8 上电时序 .....	31
4.9 时钟规格 .....	34
4.9.1 器件时钟表 .....	34

4.9.2 时钟要求和特性 .....	35
4.10 FLASH 时序 .....	37
4.10.1 FLASH/OTP 耐久度 .....	37
4.10.2 FLASH/OTP 访问时序 .....	37
4.10.3 FLASH 数据保留时间 .....	37
5 详细说明 .....	38
5.1 概述 .....	38
5.1.1 CPU .....	38
5.1.2 控制环加速器 (CLA) .....	38
5.1.3 维特比, 复数, CRC 单元 (VCU) .....	38
5.1.4 内存总线 (哈佛总线架构) .....	40
5.1.5 外设总线 .....	40
5.1.6 实时 JTAG 和分析 .....	40
5.1.7 FLASH .....	41
5.1.8 M0/M1 SARAMS .....	41
5.1.9 L4 SARAM 及 L0/L1/L2/L3/L5/L7/L8 DPSARAMS .....	41
5.1.10 引导 ROM .....	42
5.1.11 安全 .....	43
5.1.12 外设中断扩展块 (PIE) .....	44
5.1.13 外部中断 (XINT1 到 XINT3) .....	44
5.1.14 内部振荡器、振荡器和锁相环 .....	45
5.1.15 看门狗 .....	45
5.1.16 外设时钟 .....	45
5.1.17 低功耗模式 .....	45
5.1.18 外设框架 0/1/2/3 (PFN) .....	46
5.1.19 通用输入/输出复用器 .....	47
5.1.20 32 位 CPU 定时器 (0/1/2) .....	47
5.1.21 控制外设 .....	47
5.1.22 串行端口外设 .....	48
5.2 VREG、BOR、POR .....	50

5.2.1 片上稳压器 (VREG) 与 LDO	50
5.2.2 片上通电复位 (POR) 和掉电复位 (BOR) 电路	50
5.3 系统控制	52
5.3.1 内部振荡器	55
5.3.2 晶体振荡器选项	55
5.3.3 基于 PLL 的时钟模块	56
5.3.4 USB 和 HRCAP 的 PLL 模块 (PLL2)	58
5.3.5 输入时钟丢失 (NMI 看门狗功能)	59
5.3.6 CPU 看门狗模块	60
5.4 低功耗模式	61
5.5 中断	62
5.5.1 外部中断	65
5.6 外围设备	67
5.6.1 CLA 概述	67
5.6.2 模拟模块	71
5.6.3 详细说明	85
5.6.4 串行外围接口模块 (SPI)	87
5.6.5 串行通信接口模块 (SCI)	96
5.6.6 多通道缓冲串口模块 (MCBSP)	100
5.6.7 增强型控制器局域网模块 (ECAN)	110
5.6.8 集成电路 (I2C)	114
5.6.9 增强型脉宽调制器模块 (EPWM1~8)	118
5.6.10 高分辨率 PWM (HRPWM)	124
5.6.11 增强型捕获模块 (ECAP1)	125
5.6.12 高分辨率捕获模块 (HRCAP1~4)	127
5.6.13 增强型正交编码器模块 (EQEP1, EQEP2)	130
5.6.14 JTAG 端口	133
5.6.15 通用输入/输出 (GPIO) 复用器	134
5.6.16 通用串行总线 (USB)	147
6 封装信息	148

---

7 包装及可订购信息 .....	150
8 重要注意事项及声明 .....	151
联系方式 .....	152

## 前 言

本手册是 AVP32F069 数字信号处理器产品的数据手册。

本手册起草单位：湖南进芯电子科技有限公司。

本手册主要起草人：何龙、唐佳。

JXDZ3.590.018SJSC 于 2022 年 12 月首次发布。

# 1 产品概述

## 1.1 特性

- **高性能32位浮点处理器内核(AVP32)**
  - 主频 90MHz(单周期 11.11ns)
  - 支持 16×16 和 32×32 乘累加(MAC)运算
  - 支持 16×16 双 MAC 运算
  - 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)
- **浮点运算单元(FPU)**
  - 支持本机单精度浮点运算
- **可编程控制环路加速器(CLA)**
  - 32 位浮点数学加速器
  - 代码执行独立于主 CPU 之外
- **维特比、复杂数学、CRC单元(VCU)**
  - 以扩展指令集方式支持复杂的乘法、维特比运算和循环冗余校验(CRC)
- **嵌入式存储器**
  - 高达 256KB 的闪存
  - 高达 100KB 的随机存取存储器(RAM)
- **2KB 一次性可编程 ROM(OTP)**
- **6通道直接内存存储(DMA)**
- **低器件和系统成本**
  - 3.3V 单电源供电, 无电源时序要求
  - 集成上电复位和欠压复位监控
  - 低功耗操作模式
  - 无模拟支持引脚
- **字节序: 小端字节序**
- **支持JTAG边界扫描**
  - IEEE 标准 1149.1-1990 标准测试访问端口和边界扫描架构
- **时钟**
  - 两个内部零引脚振荡器
  - 片载晶体振荡器/外部时钟输入
  - 看门狗计时器模块
  - 丢失时钟检测电路
- **可支持所有外设中断扩展(PIE)模块**
- **三个32位CPU计时器**
- **高级控制外设**
- **8个增强型脉冲宽度调制器(ePWM)模块**

- 总共 16 个 PWM 通道(可支持 8 个 HRPWM)
- 每个模块中的独立 16 位计时器
- **3个输入增强型捕捉(eCAP)模块**
- **多达4个高分辨率捕捉(HRCAP)模块**
- **多达2个增强型正交编码器脉冲(eQEP)模块**
- **12位模数转换器(ADC), 具有双路采样与保持(S/H)功能**
  - 高达 3.46MSPS
  - 高达 16 通道
- **片上温度传感器**
- **128 位安全密钥和锁**
  - 保护安全内存块
  - 防止固件逆向工程
- **串行端口外设**
  - 两个串行通信接口(SCI) 模块
  - 两个串行外设接口(SPI) 模块
- 一个内部集成电路(I2C) 总线
- 一个多通道缓冲串行端口(McBSP) 总线
- 一个增强型控制器局域网(eCAN)
- 通用串行总线 USB 2.0 规范
  - 全速器件模式
  - 全速/低速主机模式
- **多达54个具有输入滤波功能的独立可编程、多路复用通用输入/输出(GPIO)引脚**
- **高级仿真特性**
  - 分析和断点功能
  - 通过硬件进行实时调试
- **封装选项**
  - 100 引脚 QP 四方扁平封装(LQFP)
  - 80 引脚 QP 四方扁平封装(LQFP)
- **温度选项**
  - S: -40°C至 125°C



## 1.2 市场应用

- 空调室外机
- 电梯门自动启闭装置驱动控制
- 逆变器和电机控制
- 车载充电器 (OBC) 和无线充电器
- 自动分拣设备
- CNC 控制
- 纺织机
- 焊接机
- 电动汽车充电站电源模块
- 车辆无线充电模块
- 能量存储电源转换系统 (PCS)
- 中央逆变器
- 微型逆变器
- 太阳能电源优化器
- 串式逆变器
- 交流驱动器控制模块
- 交流驱动器功率级模块
- 线性电机功率级
- 伺服驱动器控制模块
- 伺服驱动器功率级模块
- 交流输入 BLDC 电机驱动器
- 直流输入 BLDC 电机驱动器
- 工业交流/直流转换器
- 三相UPS

## 1.3 产品描述

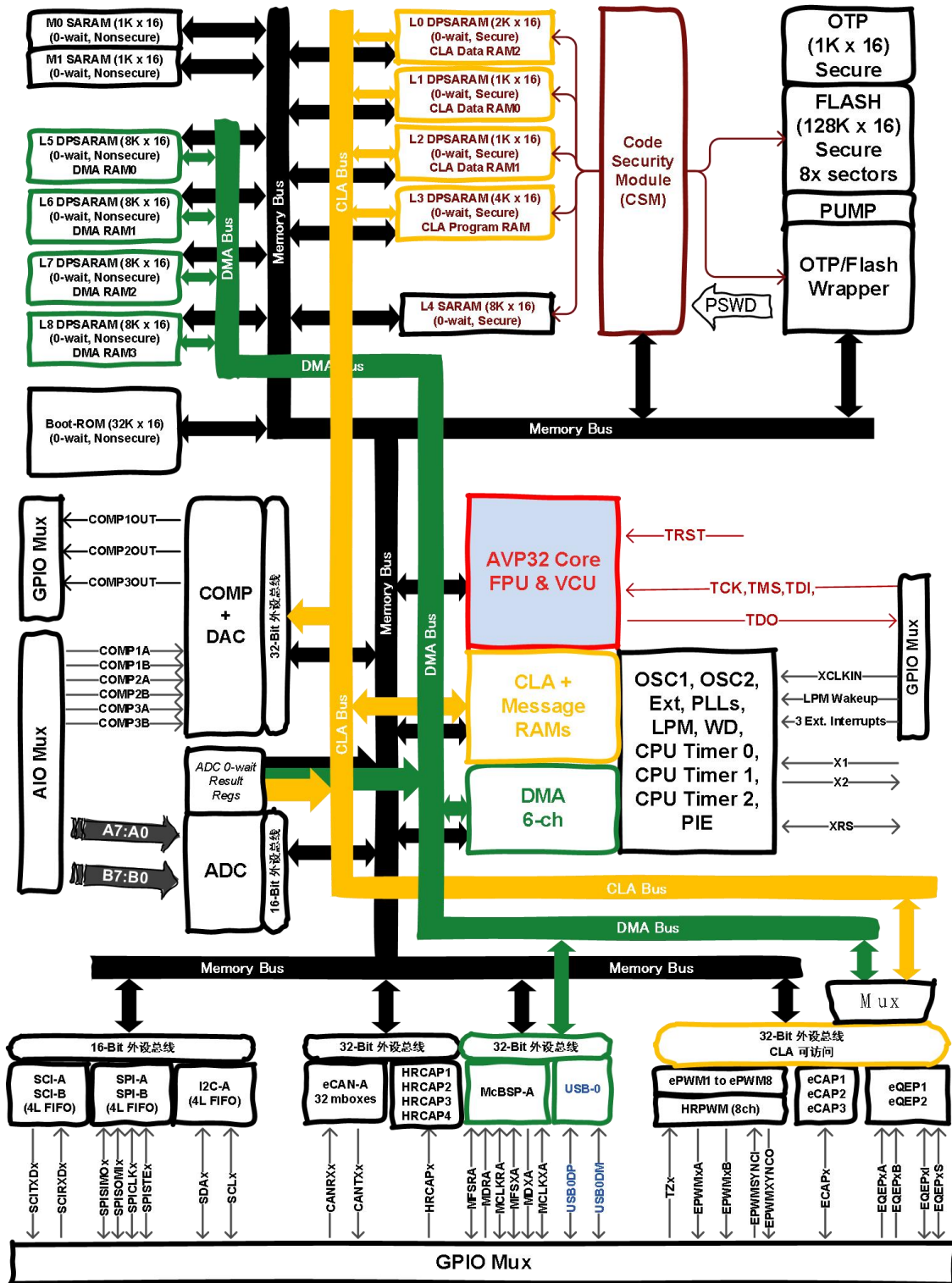
ADP32 和 AVP32 系列数字信号处理器，在处理、传感和驱动方面进行了优化，可提高实时控制应用中的闭环性能，例如工业电机驱动、光伏逆变器和数字电源、电动车辆与运输、电机控制以及传感和信号处理。AVP32F069 是继 ADP32F035 后开发的又一颗 32 位数字信号处理器，在主频提升，存储容量增加的基础上，引入了单精度浮点运算单元以及控制环加速协处理器，进一步增强 32 位系列产品的处理能力，扩展了产品的应用范围。

AVP32F069 数字信号处理器 (DSP) 提供了包含 AVP32 浮点内核以及控制环加速器 (CLA) 的强大处理能力。实现了与 ADP32F035 器件的代码兼容，以及性能升级需求。以较少的引脚输出，高度集成控制外设和高级别的模拟集成。多个内部线性稳压电源实现了单电源供电。对 HRPWM 模块实施了改进，以提供双边缘控制 (调频)。器件内还新增了采用 10 位内部基准的模拟比较器，可通过与其直接相连来控制 ePWM 输出。ADC 可在 0V 至 3.3V 的固定满量程范围内实施转换，支持 VREFHI/VREFLO 基准的比例运算。ADC 接口已针对低开销和延迟进行了优化。

表 1-1 器件信息

型号	封装	尺寸
AVP32F069QP100S	LQFP 100	14.0mm x 14.0mm
AVP32F069QP80S	LQFP 80	12.0mm x 12.0mm

### 1.4 功能框图



注释：因引脚复用原因，并非所有功能外设可同时使用。

图 1-1 功能框图

## 1.5 系统模块图

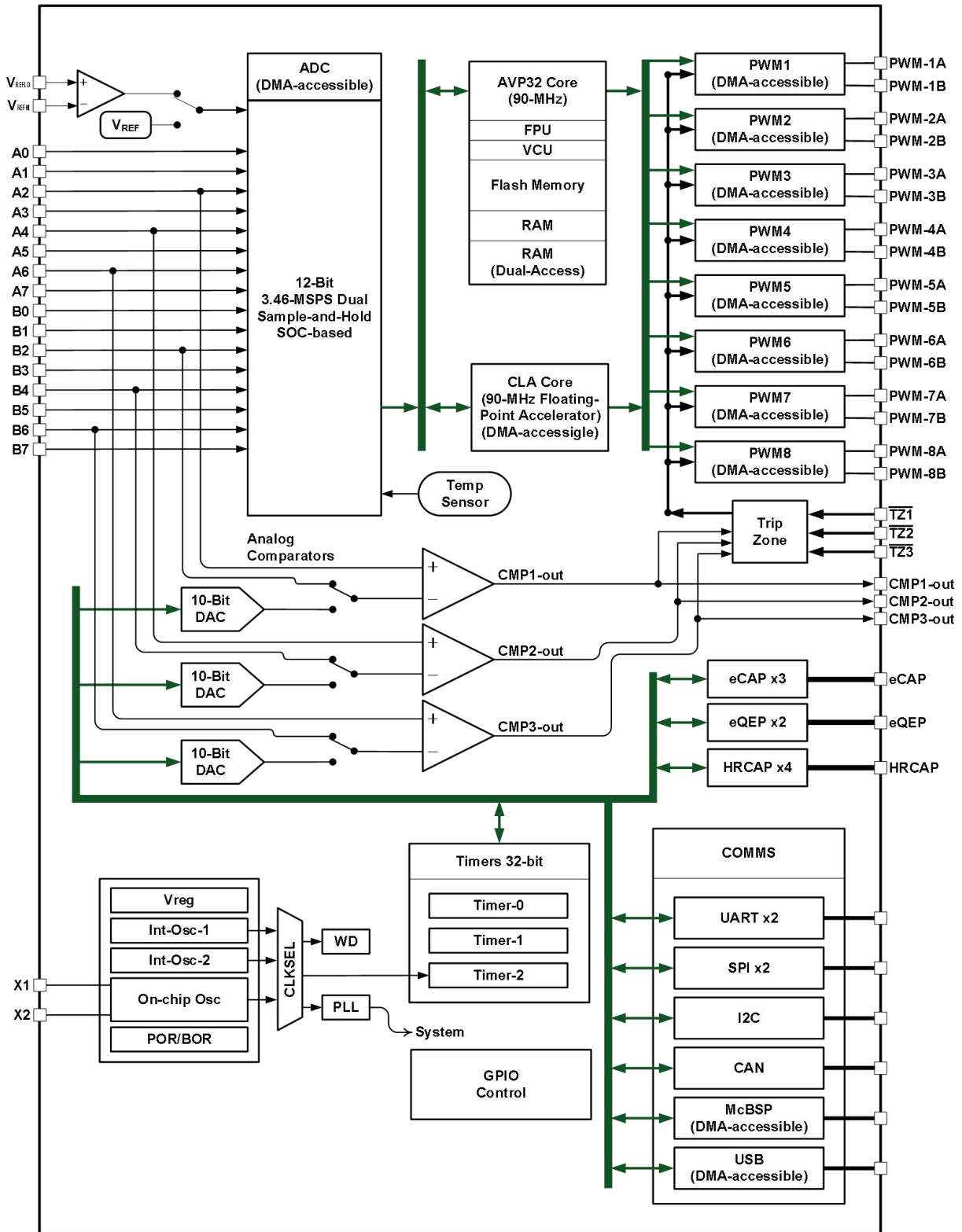
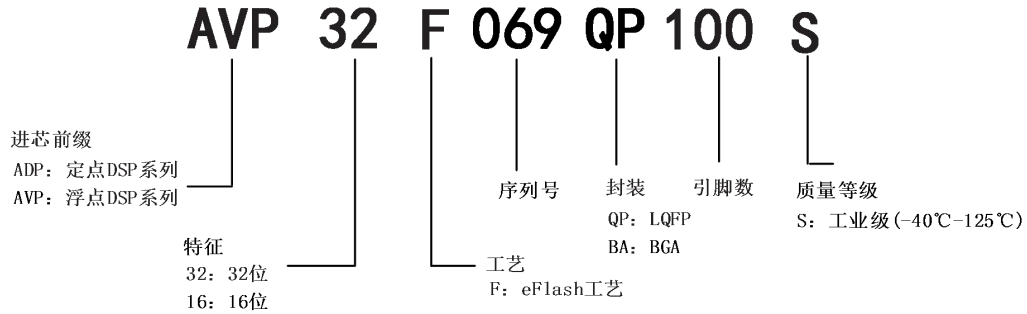


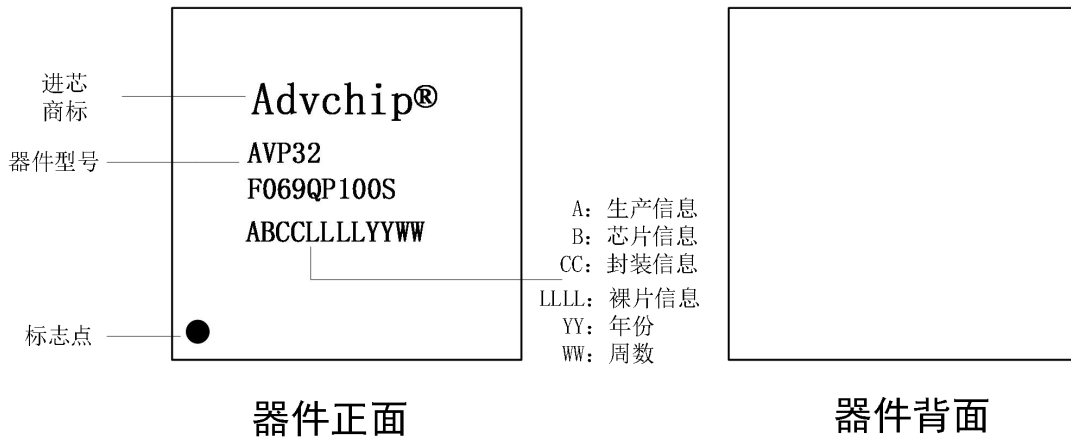
图 1-2 外设模块

## 1.6 产品编码

AVP32F069 产品代号编码规则:



器件标识:



## 2 器件对比表

**表 2-1 器件对比**

特征		类型	AVP32F069QP100S	AVP32F069QP80S
指令周期			11.11ns	11.11ns
浮点运算(FPU)			单精度	单精度
VCU			有	有
CLA		0	有	有
6 通道 DMA		0	有	有
片上 Flash(16 位字)			128K	128K
片上 SARAM(16 位字)			50K	50K
加密模块(CSM)			有	有
引导 ROM(16 位字)			32K	32K
OTP ROM(16 位字)			1K	1K
ePWM 通道		1	16	16
HRPWM 通道		1	8	8
eCAP 输入		0	3	3
HRCAP		0	4	4
eQEP 模块		0	2	2
看门狗			有	有
12 位 ADC	转换速率(MSPS)	3	3.46	3.46
	转换时间		289ns	289ns
	通道数		16	16
	温度传感器		有	有
	采样保持电路		有	有
32 位 CPU 定时器			3	3
集成 DAC 的比较器		0	3	3
I2C		0	1	1
McBSP		1	1	1
eCAN		0	1	1
SPI		1	2	2
SCI/UART		0	2	2
USB		0	1	1
双引脚振荡器			1	1
零引脚振荡器			2	2
I/O 引脚 (共享)	GPIO		54	54
	AIO		6	6
外部中断			3	3
供电电压			3.3 V	3.3 V
温度范围 <sup>(1)</sup>		S	-40°C~125°C	-40°C~125°C

(1) 设计指标为环境温度-40°C~125°C, 结温范围-40°C~150°C。

### 3 终端配置和功能

#### 3.1 引脚图

图 3-1 为 100 脚四方扁平封装 (LQFP) 引脚分布。

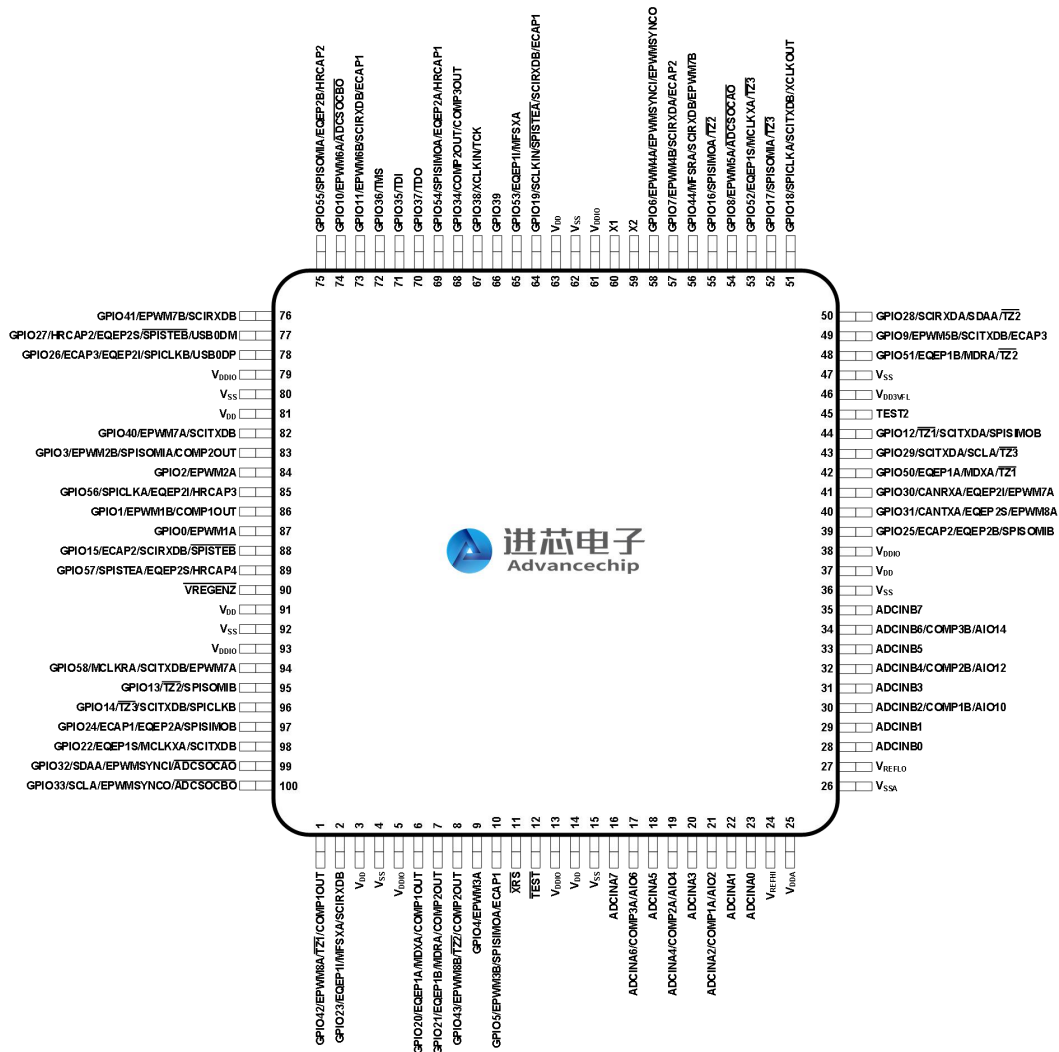


图 3-1 100 脚 LQFP 封装 (顶视图)

图 3-2 为 80 脚四方扁平封装 (LQFP) 引脚分布。

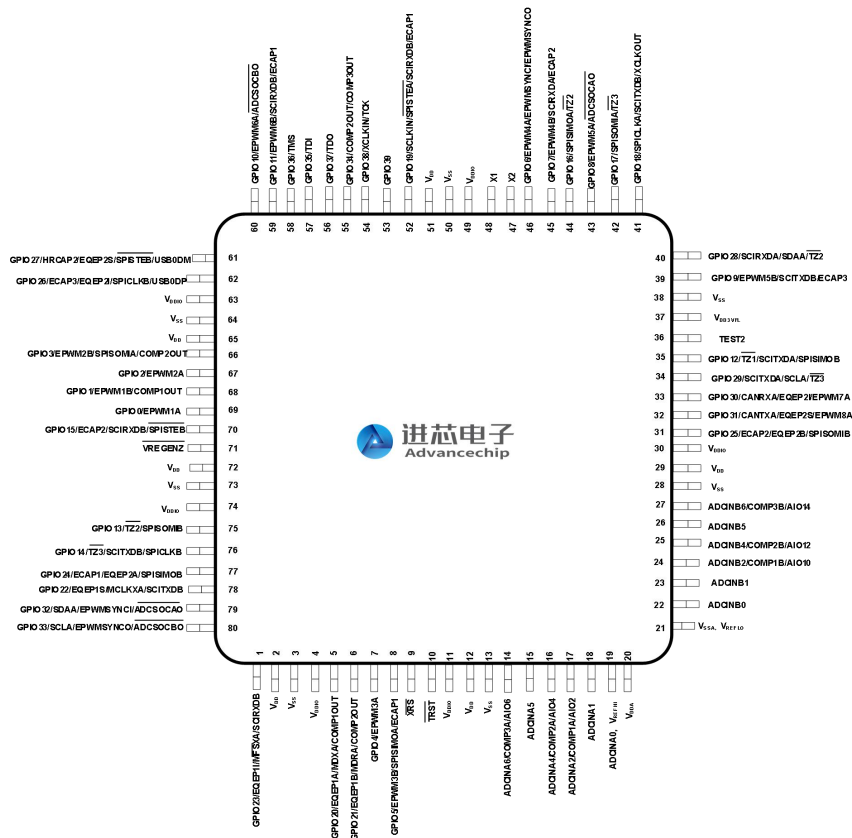


图 3-2 80 脚 LQFP 封装 (顶视图)

### 3.2 内存映射

- 内存块不支持缩放。
- 外围帧 0、外围帧 1、外围帧 2 和外围帧 3 内存映射仅限于数据内存。用户程序无法访问程序空间中的这些内存映射。
- 受保护是指保留写操作后的读取操作顺序，而不是管道顺序。
- 在配置后，某些内存范围可防止虚假写入。
- 位置 0x3D7C80~0x3D7CC0 包含内部振荡器和 ADC 校准例程，属出厂配置内存。用户无法对这些位置进行编程。



### 3.3 信号描述

3.2.1 章节描述了信号。除了 JTAG 引脚外，GPIO 功能是复位时的默认值，除非另有说明。它们下面列出的外设信号是备用功能。某些外设功能可能在所有设备中都不可用。详见 Table3-1。不允许 5V 输入。所有 GPIO 引脚都是 I/O/Z，并且具有内部上拉 (PU)，可以根据每个引脚选择性地启用或禁用。此功能仅适用于 GPIO 引脚。PWM 引脚上的上拉装置在复位时未启用。复位后，其他 GPIO 引脚上的上拉将启用。AIO 引脚没有内部上拉。

#### 注意

使用片上电压调节器 (VREG) 时，GPIO19、GPIO26–27 和 GPIO34–38 引脚可能在通电期间出现故障。此潜在故障将在读取引导模式引脚之前结束，不会影响引导行为。如果应用中不可接受出现此故障，可从外部提供 1.8 V。或者，可以考虑添加一个与这些引脚和任何外部驱动器串联的限流电阻器 (例如 470Ω)，以限制引脚和/或外部电路退化的可能性。使用外部 1.8 V 电源时，无电源排序要求。但是，如果输入/输出引脚的电平移位输出缓冲器中的 3.3-V 晶体管在 1.8-V 晶体管之前通电，则输出缓冲器可能会开启，从而导致通电期间引脚上出现故障。为避免此行为，请在  $V_{DDIO}$  引脚之前或与  $V_{DDIO}$  引脚同时为  $V_{DD}$  引脚通电，确保  $V_{DD}$  引脚在  $V_{DDIO}$  引脚达到 0.7 V 之前已达到 0.7 V。

#### 3.3.1 信号说明

引脚名称	引脚编号		I/O/Z <sup>(1)</sup>	说 明
	QP100	QP80		
JTAG				
TRST	12	10	I	JTAG 测试复位，带有内部下拉功能(PD)。TRST，当驱动高时，提供扫描系统控制设备的操作。如果该信号未连接或驱动低，设备在其功能模式下运行，测试复位信号被忽略。注意：TRST是高有效测试引脚，在正常运行期间必须始终保持低。该引脚上需要一个外部下拉电阻器。该电阻器的值应基于适用于该设计的调试器的驱动强度。2.2kΩ电阻通常提供足够的保护。
TCK	参考 GPIO38		I	带有内部上拉功能的 JTAG 测试时钟。(†)
TMS	参考 GPIO36		I	带有内部上拉功能的 JTAG 测试模式选择(TMS)。该串行控制输入钟进入 TCK 上升边缘的 TAP 控制器。(†)
TDI	参考 GPIO35		I	带有内部上拉功能的 JTAG 测试数据输入(TDI)。TDI 被记录到 TCK 的上升边缘上的选定寄存器 (指令或数据) 中。(†)
TDO	参考 GPIO37		O/Z	JTAG 扫描导出，测试数据输出(TDO)。所选寄存器 (指令或数据) 的内容将离开 TCK 下降边的 TDO。(8mA 驱动器)
FLASH				
$V_{DD3VFL}$	46	37		3.3v 闪存核心电源销。该引脚应始终连接到 3.3V。
TEST2_VCORE	45	36	I/O	内核 1.2V 输出引脚。
CLOCK				
XCLKOUT	参考 GPIO18		O/Z	输出时钟。XCLKOUT 要么是相同的频率，要么是该频率的一半，要么是同步输出频率

引脚名称	引脚编号		I/O/Z <sup>(1)</sup>	说 明
	QP100	QP80		
				的四分之一。这是由 XCLK 寄存器中的位 1: 0(XCLKDV)控制的。重置时, 外部检查=系统检查/4。通过将 XCLKOUTDIV 设置为 3, 可以关闭 XCLKOUT 信号。GPIO18 的 MUX 控制也必须设置为外部输出, 以便这个信号输出到引脚。
XCLKIN	参考 GPIO19、GPIO38		I	外部振荡器输入。时钟的针源由 XCLK 寄存器中的轴位控制, GPIO38 是默认选择。这个引脚从一个外部的 3.3V 振荡器提供一个时钟。在这种情况下, X1 引脚, 如果可用, 必须绑定到 GND, 并且片上的晶体振荡器必须通过 CLKCTL 寄存器中的第 14 位被禁用。如果使用晶体或谐振器, XCLKIN 路径必须被 CLKCTL 寄存器中的第 13 位禁用。
X1	60	48	I	片上 1.8V 晶体振荡器输入。要使用这个振荡器, 石英晶体或陶瓷谐振器必须连接在 X1 和 X2 之间。在这种情况下, XCLKIN 路径必须被 CLKCTL 寄存器中的第 13 位所禁用。如果不使用此引脚, 则必须将其绑定到 GND 上。
X2	59	47	O	片上晶体振荡器的输出。石英晶体或陶瓷谐振器必须连接在 X1 和 X2 之间。如果未使用 X2, 则必须保持不连接。
RESET				
XRS	11	9	I/OD	设备重置 (在) 和看门狗重置 (输出)。这些设备具有内置的电源复位(POR)和紧急断电复位(BOR)电路。在通电或断电条件下, 这个引脚被设备驱动到低。外部电路也可以驱动该引脚, 以维护设备复位。当看门狗复位发生时, 这个销也被 MCU 驱动到低。在看门狗复位期间, XRS销被驱动到较低的看门狗复位持续时间为 512OSCCLK 周期。在 XRS和 VDDIO 之间放置 2.2kΩ到 10kΩ的电阻。如果一个电容器被放置在XRS和 VSS 之间进行噪声滤波, 它应该是 100nF 或更小。当确认看门狗复位时, 这些值将允许看门狗在 512 个 OSCCLK 周期内正确地驱动XRS大头针到 VOL。无论源如何, 设备重置都会导致设备终止执行。程序计数器指向该位置上所包含的地址 0x3F FFC0.停用复位后, 将从程序计数器指定的位置开始执行。这个引脚的输出缓冲器是一个具有内部上拉装置的开漏极装置。(1) 如果此引脚由外部装置驱动, 则应使用开漏装置执行。
ADC, 比较器, 模拟 I/O				
ADCINA7	16	–	I	ADC Group A, 通道 7 输入
ADCINA6	17	14	I	ADC Group A, 通道 6 输入
COMP3A			I	比较器输入 3A
AIO6			I/O	数字 AIO6
ADCINA5	18	15	I	ADC Group A, 通道 5 输入
ADCINA4	19	16	I	ADC Group A, 通道 4 输入
COMP2A			I	比较器输入 2A
AIO4			I/O	数字 AIO4
ADCINA3	20	–	I	ADC Group A, 通道 3 输入
ADCINA2	21	17	I	ADC Group A, 通道 2 输入
COMP1A			I	比较器输入 1A
AIO2			I/O	数字 AIO2
ADCINA1	22	18	I	ADC Group A, 通道 1 输入
ADCINA0	23	19	I	ADC Group A, 通道 0 输入

引脚名称	引脚编号		I/O/Z <sup>(1)</sup>	说 明
	QP100	QP80		
				注意: VREFHI 和 ADCINA0 在 80 脚封装的设备上共享相同的引脚, 并且它们的使用是相互排斥的。
VREFHI	24	19		ADC 外部参考仅在 ADC 外部参考模式下使用, 见第 6.9.2.1 节。注意: VREFHI 和 ADCINA0 在 80 针的 PN 和 PFP 设备上共享相同的引脚, 并且它们使用是相互排斥的。
ADCINB7	35	-	I	ADC Group B, 通道 7 输入
ADCINB6			I	ADC Group B, 通道 6 输入
COMP3B AIO14	34	27	I I/O	比较器输入 3B 数字 AIO14
ADCINB5	33	26	I	ADC Group B, 通道 5 输入
ADCINB4			I	ADC Group B, 通道 4 输入
COMP2B AIO12	32	25	I I/O	比较器输入 2B 数字 AIO12
ADCINB3	31	-	I	ADC Group B, 通道 3 输入
ADCINB2			I	ADC Group B, 通道 2 输入
COMP1B AIO10	30	24	I I/O	比较器输入 1B 数字 AIO10
ADCINB1	29	23	I	ADC Group B, 通道 1 输入
ADCINB0	28	22	I	ADC Group B, 通道 0 输入
VREFLO	27	21		ADC 外部参考低。 注意: VREFLO 始终连接到 80 脚封装的设备的 VSSA 脚上。
VDDA	25	20		模拟功率引脚。接近 2.2 $\mu$ F 电容器 (典型)
VSSA	26	21		模拟接地引脚。 注意: VREFLO 始终连接到 80 脚封装的设备的 VSSA 脚上。
VDD	3	2		CPU 和逻辑数字电源引脚。当使用内部 VREG 时, 在每个 VDD 引脚和接地之间放置一个 1.2 $\mu$ F 的电容器。可以使用更高值的电容器。
	14	12		
	37	29		
	63	51		
	81	65		
VDDIO	91	72		数字 I/O 缓冲器的电源引脚。当启用 VREG 时, 单个电源。在每个引脚上放置一个解耦电容器。确切的值应由系统电压调节的解决方案来确定。
	5	4		
	13	11		
	38	30		
	61	49		
VCC	79	63		数字接地引脚
	93	74		
	4	3		

引脚名称	引脚编号		I/O/Z <sup>(1)</sup>	说 明
	QP100	QP80		
	15	13		
	36	28		
	47	38		
	62	50		
	80	64		
	92	73		
通用输入输出及外设信号				
GPIO0 EPWM1A Reserved Reserved	87	69	I/O/Z O - -	通用输入输出 0 增强的 PWM1 输出 A 和 HRPWM 通道 预留 预留
GPIO1 EPWM1B Reserved COMP1OUT	86	68	I/O/Z O - O	通用输入输出 1 增强型 PWM1 输出 B 预留 比较器 1 的直接输出
GPIO2 EPWM2A Reserved Reserved	84	67	I/O/Z O - -	通用输入输出 2 增强的 PWM2 输出 A 和 HRPWM 通道 预留 预留
GPIO3 EPWM2B SPISOMIA COMP2OUT	83	66	I/O/Z O I/O O	通用输入输出 3 增强型 PWM2 输出 B SPI-A 从机输出, 主机输入 比较器 2 的直接输出
GPIO4 EPWM3A Reserved Reserved	9	7	I/O/Z O - -	通用输入输出 4 增强的 PWM3 输出 A 和 HRPWM 通道 预留 预留
GPIO5 EPWM3B SPISIMOA ECAP1	10	8	I/O/Z O I/O I/O	通用输入输出 5 增强的 PWM3 输出 B SPI-A 从机输入, 主机输出 增强的捕获输入/输出 1
GPIO6 EPWM4A EPWMSYNCI EPWMSYNCO	58	46	I/O/Z O I O	通用输入输出 6 增强的 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
GPIO7 EPWM4B SCIRXDA ECAP2	57	45	I/O/Z O I I/O	通用输入输出 7 增强的 PWM4 输出 B SCI-A 接收数据 增强的捕获输入/输出 2
GPIO8	54	43	I/O/Z	通用输入输出 8

引脚名称	引脚编号		I/O/Z <sup>(1)</sup>	说 明
	QP100	QP80		
EPWM5A Reserved $\overline{\text{ADCSOCAO}}$ GPIO9	49	39	O	增强的 PWM5 输出 A 和 HRPWM 通道
			-	预留
			O	ADC 转换开始 A
EPWM5B SCITXDB ECAP3			I/O/Z	通用输入输出 9
			O	增强的 PWM5 输出 B
			O	SCI-B 传输数据
			I/O	增强的捕获输入/输出 3
GPIO10 EPWM6A Reserved $\overline{\text{ADCSOCBO}}$	74	60	I/O/Z	通用输入输出 10
			O	增强的 PWM6 输出 A 和 HRPWM 通道
			-	预留
			O	ADC 转换开始 B
GPIO11 EPWM6B SCIRXDB ECAP1	73	59	I/O/Z	通用输入输出 11
			O	增强的 PWM6 输出 B
			I	SCI-B 接收数据
			I/O	增强的捕获输入/输出 1
GPIO12 $\overline{\text{TZ1}}$ SCITXDA SPISIMOB	44	35	I/O/Z	通用输入输出 12
			I	跳闸区域输入 1
			O	SCI-A 传输数据
			I/O	SPI-B 从机输入, 主机输出
GPIO13 $\overline{\text{TZ2}}$ Reserved SPISOMIB	95	75	I/O/Z	通用输入输出 13
			I	跳闸区输入 2
			-	预留
			I/O	SPI-B 从机输出, 主机输入
GPIO14 $\overline{\text{TZ3}}$ SCITXDB SPICLKB	96	76	I/O/Z	通用输入输出 14
			I	跳闸区域输入 3
			O	SCI-B 传输数据
			I/O	SPI-B 时钟输入输出
GPIO15 ECAP2 SCIRXDB $\overline{\text{SPISTEB}}$	88	70	I/O/Z	通用输入输出 15
			I/O	增强的捕获输入/输出 2
			I	SCI-B 接收数据
			I/O	SPI-B 从机传输使能输入/输出
GPIO16 SPISIMOA Reserved $\overline{\text{TZ2}}$	55	44	I/O/Z	通用输入输出 16
			I/O	SPI-A 从机输入, 主机输出
			-	预留
			I	跳闸区输入 2
GPIO17 SPISOMIA Reserved $\overline{\text{TZ3}}$	52	42	I/O/Z	通用输入输出 17
			I/O	SPI-A 从机输出, 主机输入
			-	预留
			I	跳闸区域输入 3
GPIO18 SPICLKA	51	41	I/O/Z	通用输入输出 18
			I/O	SPI-A 时钟输入输出

引脚名称	引脚编号		I/O/Z <sup>(1)</sup>	说 明
	QP100	QP80		
SCITXDB			O	SCI-B 传输数据
XCLKOUT			O/Z	输出时钟 XCLKOUT 要么是相同的频率, 要么是 1-频率的一半, 或系统碰撞频率的四分之一。这是由 XCLK 寄存器中的位 1: 0(XCLKUDIV)。重置时, 外部检查=系统检查/4。通过将 XCLKUDIV 设置为 3, 可以关闭 XCLKOUT 信号。MUX 控件对于 GPIO18 也必须设置为外部输出, 以便这个信号突出到引脚。
GPIO19	64	52	I/O/Z	通用输入输出 19
XCLKIN			I	外部振荡器输入从这个大头针到时钟块的路径不是由这个引脚的 MUX 函数。如果是, 必须注意不要启用此路径进行时钟可用于其他外围设备的功能。
SPISTEA			I/O	从 SPI-A 传输使能输入/输出
SCIRXDB			I	SCI-B 接收数据
ECAP1			I/O	增强的捕获输入/输出 1
GPIO20	6	5	I/O/Z	通用输入输出 20
EQEP1A			I	增强的 QEP1 输入 A
MDXA			O	McBSP-A 传输串行数据
COMP1OUT			O	比较器 1 的直接输出
GPIO21	7	6	I/O/Z	通用输入输出 21
EQEP1B			I	增强型 QEP1 输入 B
MDRA			I	McBSP 接收串行数据
COMP2OUT			O	比较器 2 的直接输出
GPIO22	98	78	I/O/Z	通用输入输出 22
EQEP1S			I/O	增强的 QEP1 频闪器
MCLKXA			I/O	McBSP-A 发射时钟
SCITXDB			O	SCI-B 传输数据
GPIO23	2	1	I/O/Z	通用输入输出 23
EQEP1I			I/O	增强的 QEP1 指数
MFSXA			I/O	McBSP-A 传输帧同步
SCIRXDB			I	SCIB 接收数据
GPIO24	97	77	I/O/Z	通用输入输出 24
ECAP1			I/O	增强的捕获输入/输出 1
EQEP2A			I	增强的 QEP2 输入 A。 注意: eQEP2 仅在 PZ 和 PZP 软件包中可用。
SPISIMOB			I/O	SPI-B 从机输入, 主机输出
GPIO25	39	31	I/O/Z	通用输入输出 25
ECAP2			I/O	增强的捕获输入/输出 2
EQEP2B			I	增强的 QEP2 输入 B。注意: eQEP2 仅在 PZ 和 PZP 软件包中可用。
SPISOMIB			I/O	SPI-B 从机输出, 主机输入
GPIO26	78	62	I/O/Z	通用输入输出 26
ECAP3			I/O	增强的捕获输入/输出 3 增强的 QEP2 指数。
EQEP2I			I/O	注意: eQEP2 仅在 PZ 和 PZP 软件包中可用。
SPICKB			I/O	SPI-B 时钟输入输出

引脚名称	引脚编号		I/O/Z <sup>(1)</sup>	说 明
	QP100	QP80		
USB0DP <sup>(3)</sup>			I/O	USB 信号的正差分一半。要在此引脚上启用 USB 功能, 请在 GPACTRL2 寄存器中设置 usbioen 位。
GPIO27 HRCAP2 EQEP2S SPISTEB USB0DM <sup>(3)</sup>	77	61	I/O/Z I I/O I/O I/O	通用输入输出 27 高分辨率输入捕获 2 增强的 QEP2 闪光灯。 注意: eQEP2 仅在 PZ 和 PZP 软件包中可用。 从传输启用输入/输出 USB 信号的负差分的一半。要在此引脚上启用 USB 功能, 请在 GPACTRL2 寄存器中设置 usbioen 位。
GPIO28 SCIRXDA SDAA TZZ	50	40	I/O/Z I I/OD I	通用输入输出 28 SCI-A 接收数据 I2C 数据开漏双向端口 跳闸区输入 2
GPIO29 SCITXDA SCLA TZ3	43	34	I/O/Z O I/OD I	通用输入输出 29 SCIA 传输数据 I2C 时钟开漏双向端口 跳闸区域输入 3
GPIO30 CANRXA EQEP2I EPWM7A	41	33	I/O/Z I I/O O	通用输入输出 30 可以接收增强的 QEP2 指数。 注意: eQEP2 仅在 PZ 和 PZP 软件包中可用。 增强的 PWM7 输出 A 和 HRPWM 通道
GPIO31 CANTXA EQEP2S EPWM8A	40	32	I/O/Z O I/O O	通用输入输出 31 CAN 传输 增强的 QEP2 闪光灯。 注意: eQEP2 仅在 PZ 和 PZP 软件包中可用。 增强的 PWM8 输出 A 和 HRPWM 通道
GPIO32 SDAA EPWMSYNCI ADCSOAO	99	79	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 转换开始 A
GPIO33 SCLA EPWMSYNCO ADCSOAO	100	80	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 转换开始 B
GPIO34 COMP2OUT Reserved COMP3OUT	68	55	I/O/Z O - O	通用输入输出 34 比较器 2 的直接输出 预留 比较器 3 的直接输出
GPIO35 TDI	71	57	I/O/Z I	通用输入输出 35 带有内部上拉功能的 JTAG 测试数据输入(TDI)。TDI 被记录到 TCK 的上升边缘上的选定寄存器 (指令或数据) 中。
Reserved			-	预留
Reserved			-	预留

引脚名称	引脚编号		I/O/Z <sup>(1)</sup>	说 明
	QP100	QP80		
Reserved			-	预留
GPIO36	72	58	I/O/Z	通用输入输出 36
TMS			I	JTAG 测试模式选择(TMS)与内部上拉。该串行控制输入钟进入 TCK 上升边缘的 TAP 控制器。
Reserved			-	预留
Reserved			-	预留
Reserved			-	预留
GPIO37	70	56	I/O/Z	通用输入输出 37
TDO			O/Z	JTAG 扫描导出, 测试数据输出(TDO)。所选寄存器(指令或数据)的内容将离开 TCK (8mA 驱动器) 下降边缘的 TDO。
Reserved			-	预留
Reserved			-	预留
Reserved			-	预留
GPIO38	67	54	I/O/Z	通用输入输出 38
XCLKIN			I	外部振荡器输入。
TCK			I	从这个大头针到时钟块的路径不是由这个引脚的 MUX 函数。如果是, 必须注意不要启用此路径进行时钟被用于其他函数。带有内部上拉功能的 JTAG 测试时钟
Reserved			-	预留
Reserved			-	预留
Reserved			-	预留
GPIO39	66	53	I/O/Z	通用输入输出 39
Reserved			-	预留
Reserved			-	预留
Reserved			-	预留
GPIO40	82	-	I/O/Z	通用输入输出 40
EPWM7A			O	增强的 PWM7 输出 A 和 HRPWM 通道
SCITXDB			O	SCI-B 传输数据
Reserved			-	预留
GPIO41	76	-	I/O/Z	通用输入输出 41
EPWM7B			O	增强的 PWM7 输出 B
SCIRXDB			I	SCI-B 接收数据
Reserved			-	预留
GPIO42	1	-	I/O/Z	通用输入输出 42
EPWM8A			O	增强的 PWM8 输出 A 和 HRPWM 通道
TZI			I	跳闸区域输入 1
COMP1OUT			O	比较器 1 的直接输出
GPIO43	8	-	I/O/Z	通用输入输出 43
EPWM8B			O	增强的 PWM8 输出 B
TZZ			I	跳闸区输入 2
COMP2OUT			O	比较器 2 的直接输出
GPIO44	56	-	I/O/Z	通用输入输出 44



引脚名称	引脚编号		I/O/Z <sup>(1)</sup>	说 明
	QP100	QP80		
MFSRA			I/O	McBSP-A 接收帧同步
SCIRXDB			I	SCIB 接收数据
EPWM7B			O	增强的 PWM7 输出 B
GPIO50	42	-	I/O/Z	通用输入输出 50
EQEP1A			I	增强的 QEP1 输入 A
MDXA			O	McBSP-A 传输串行数据
TZ1			I	跳闸区域输入 1
GPIO51	48	-	I/O/Z	通用输入输出 51
EQEP1B			I	增强型 QEP1 输入 B
MDRA			I	McBSP-A 接收串行数据
TZ2			I	跳闸区输入 2
GPIO52	53	-	I/O/Z	通用输入输出 52
EQEP1S			I/O	增强的 QEP1 频闪器
MCLKXA			I/O	McBSP-A 发射时钟
TZ3			I	跳闸区域输入 3
GPIO53	65	-	I/O/Z	通用输入输出 53
EQEP1I			I/O	增强的 QEP1 指数
MFSXA			I/O	McBSP-A 传输帧同步
Reserved			-	预留
GPIO54	69	-	I/O/Z	通用输入输出 54
SPISIMOA			I/O	SPI-A 从机输入, 主机输出
EQEP2A			I	增强的 QEP2 输入 A
HRCAP1			I	高分辨率输入捕获 1
GPIO55	75	-	I/O/Z	通用输入输出 55
SPISOMIA			I/O	SPI-A 从机输入, 主机输出
EQEP2B			I	增强的 QEP2 输入 B
HRCAP2			I	高分辨率输入捕获 2
GPIO56	85	-	I/O/Z	通用输入输出 56
SPICLKA			I/O	SPI-A 时钟输入/输出
EQEP2I			I/O	增强的 QEP2 索引
HRCAP3			I	高分辨率输入捕获 3
GPIO57	89	-	I/O/Z	通用输入输出 57
SPISTEA			I/O	从发射启用输入/输出增强的
EQEP2S			I/O	QEP2 频闪器
HRCAP4			I	高分辨率输入捕获 4
GPIO58	94	-	I/O/Z	通用输入输出 58
MCLKRA			I/O	McBSP-A 接收时钟
SCITXDB			O	SCI-B 传输数据
EPWM7A			O	增强的 PWM7 输出 A 和 HRPWM 通道

## 4 技术规格

### 4.1 最大额定值

超出下列应力最大额定值可能导致芯片的永久性损坏，长时间工作在最大额定值下也会影响芯片的可靠性。限值仅反映应力等级，并非暗示器件功能可超出推荐操作值，在下列限制条件下正常运行。

下列所有电压值均基于  $V_{SS}$ 。

		最小值	最大值	单位
供电电压	$V_{DDIO}$ (I/O 和 Flash)相对于 $V_{SS}$	-0.3	4.6	V
	$V_{DD}$ 相对于 $V_{SS}$	-0.3	2.5	
模拟电压	$V_{DDA}$ 相对于 $V_{SSA}$	-0.3	4.6	V
输入电压	$V_{IN}$ (3.3V)	-0.3	4.6	V
	$V_{IN}$ (X1)	-0.3	2.5	
输出电压	$V_O$	-0.3	4.6	V
输入钳位电流	数字输入(每引脚), $I_{IK}(V_{IN} < V_{SS}$ 或 $V_{IN} > V_{DDIO})^{(1)}$	-20	20	mA
	模拟输入(每引脚), $I_{IKANALOG}(V_{IN} < V_{SSA}$ 或 $V_{IN} > V_{DDA})$	-20	20	
	所有输入总计, $V_{IKTOTAL}(V_{IN} < V_{SS}/V_{SSA}$ 或 $V_{IN} > V_{DDIO}/V_{DDA})$	-20	20	
输出钳位电流	$I_{OK}(V_O < 0$ 或 $V_O > V_{DDIO})$	-20	20	mA
结温 <sup>(2)</sup>	$T_J$	-40	150	°C
存储温度 <sup>(2)</sup>	$T_{STG}$	-65	150	°C

(1) 单个引脚钳位电流持续 $\pm 2$ mA。

(2) 长时间高温存储或最大温度条件下的扩展使用可能减少器件使用寿命。

### 4.2 推荐操作条件

	最小值	典型值	最大值	单位
I/O 供电电压: $V_{DDIO}$		3.3		V
CPU 及时钟模块供电电压: $V_{DD}$ (禁用内部 VREG 使用外部 1.8V 供电时)		1.8		V
数字地: $V_{SS}$		0		V
模拟供电电压: $V_{DDA}$		3.3		V
模拟地: $V_{SSA}$		0		V
器件时钟频率 (系统时钟)		90		MHz
结温: $T_J$	-40		150	°C
环境温度: $T_A$	-40		125	

(1) 第二组 GPIO 包括: GPIO16、GPIO17、GPIO18、GPIO19、GPIO28、GPIO29、GPIO36、GPIO37。

## 4.3 总体流耗

待测试

## 4.4 电气特性

待测试

## 4.5 无信号缓冲的调试仿真连接

下图演示了单 DSP 芯片和 JTAG 接头之间的连接。如果 DSP 和 JTAG 之间的距离大于 6 英寸 (15.24 厘米)，调试信号就必须要缓冲。如果距离小于 6 英寸，通常情况下不需要缓冲。下图简单展示了无缓冲的情况。

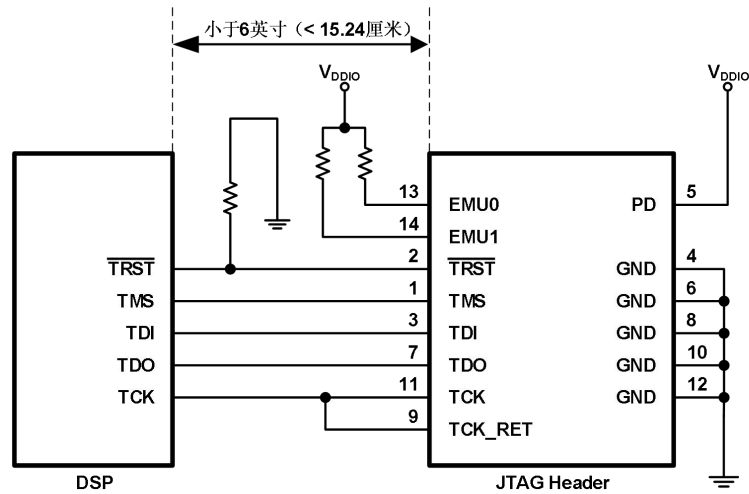


图 4-1 无信号缓冲的调试仿真连接

### 注意

AVP32F069 没有 EMU0/EMU1 引脚，板上 JTAG 接口的 EMU0/EMU1 必须通过典型值为 4.7kΩ 的电阻上拉到 V<sub>DDIO</sub>。

## 4.6 参数信息

### 4.6.1 时序参数符号

使用的时序参数符号是根据 JEDEC 标准 100 创建的。为了缩短符号，一些引脚名称和其他相关术语已缩写如下：

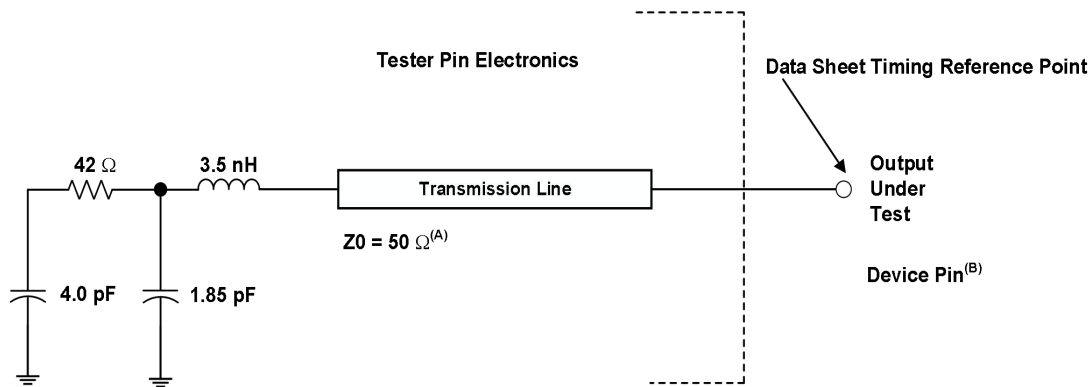
小写下标及其含义：		字母、符号及其含义：	
a	存取时间	H	高
c	周期时间	L	低
d	延迟时间	V	有效
f	下降时间	X	未知，变化，或无关紧要
h	维持时间	Z	高阻
r	上升时间		
su	建立时间		
t	转换时间		
v	有效时间		
w	脉冲宽度		

### 4.6.2 时序参数通用说明

设备的所有输出信号（包括 XCLKOUT）均来自内部时钟，因此相互之间具有最小的相位差。以下时序图中所示的信号组合可能不一定代表实际周期。有关实际周期示例，请参阅本文档中相应周期描述部分。

## 4.7 负载测试电路

此测试电路用于测量本文档中提及的所有开关特性。

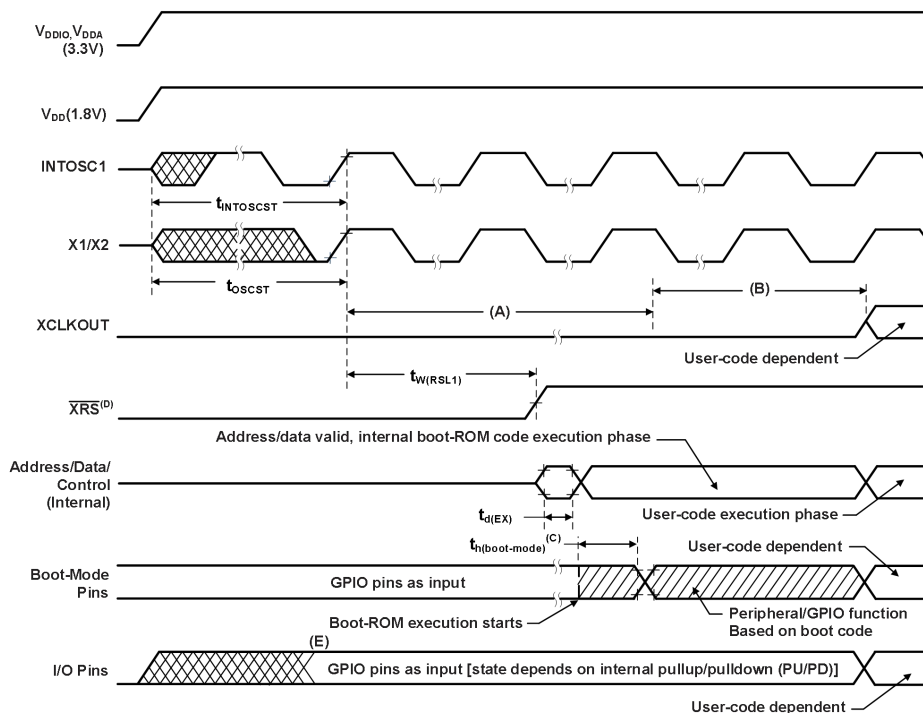


- A. 本数据表中的输入，要求器件在引脚处以小于 4 伏/纳秒 (4 V/ns) 的输入转换速率进行测试。
- B. 数据表提供了设备引脚的时序，对于输出时序分析，必须考虑测试脚电子器件和其传输线的影响。传输线延时效果在 2ns 或更长，传输线仅用作负载，没必要从手册中时序上加入或减少 2ns 或更长的传输线延时。

图 4-2 3.3V 负载测试电路

## 4.8 上电时序

在给器件通电之前，不得向任何数字引脚施加高于 VDDIO 的二极管压降 (0.7 V) 电压 (模拟引脚以 VDDA 为基准，不得高于 0.7V)。因为，施加在无源设备引脚上的电压会意外的偏置内部 p-n 结，并产生不可预料的结果。确保设备重置后处于正确状态，或防止 I/O 在通电、断电期间出现故障，在上电时序方面并无其他要求。



- A. 上电后, PLLCR 寄存器重置为 0, 锁相环旁路, PLLSTS 寄存器的 DIVSEL 位重置为 0, OSCCLK 被 4 分频后, SYSCLKOUT=OSCCLK/4。又因 XCLK 寄存器的 XCLKOUTDIV 位复位后被重置为 0, SYSCLKOUT 到 XCLKOUT 被 4 分频, 因此, 这个阶段 XCLKOUT=OSCCLK/16。
- B. Boot ROM 配置 DIVSEL=3, 为 1 分频操作。这个阶段 XCLKOUT=OSCCLK/4, 在用户代码配置前, XCLKOUT 在引脚端是不可见的 (XCLKOUTDIV=3 时, XCLKOUT 无输出)。
- C. 复位后, 引导 ROM 代码对引导模式引脚进行采样。根据引导模式引脚的状态, 引导代码分岔到目标内存或引导代码功能。在调试器环境中, 如果 Boot ROM 代码上电后执行, 则启动代码执行时间是基于当前 SYSCLKOUT 速度。SYSCLKOUT 将基于用户环境启用或不启用 PLL。
- D. 由于片上 POR 电路产生的 PBRs 信号会从芯片内部拉低 XRS 引脚, 因此 XRS 引脚可用可不用。
- E. 当 BOR 处于高位时, 内部上拉或下拉将有效。

图 4-3 上电复位

### 4.8.1 复位 ( $\overline{XRS}$ ) 时序要求

		最小值	最大值	单位
$t_{h(boot-mode)}$	引导模式引脚维持时间	$1000t_c(SCO)$		周期
$t_{w(RSL2)}$	热复位下 $\overline{XRS}$ 低电平脉宽持续时间	$32t_c(OSCCLK)$		周期

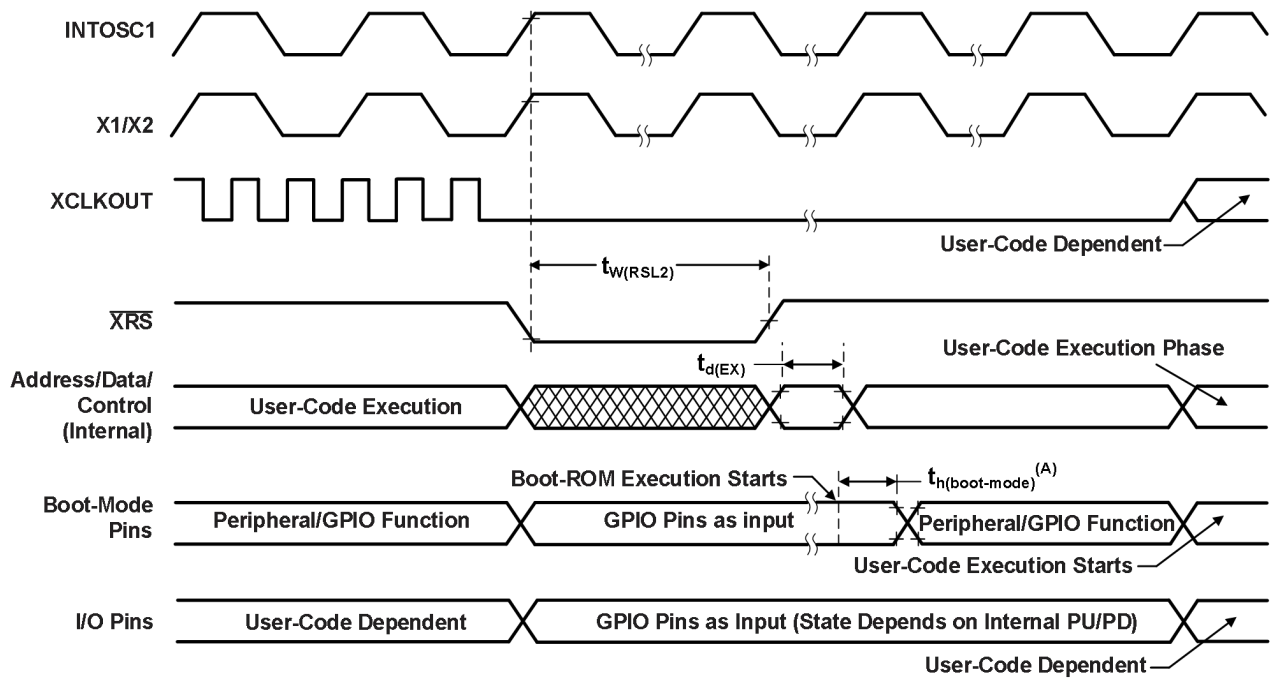
### 4.8.2 复位 ( $\overline{XRS}$ ) 开关特性



在推荐操作条件范围内（除非另有说明）

参数	最小值	典型值	最大值	单位
$t_{w(RSL1)}$		600		$\mu\text{s}$
$t_{w(WDRS)}$		$512t_{c(SSCCLK)}$		周期
$t_{d(EX)}$		$32t_{c(SSCCLK)}$		周期
$t_{INTOSCST}$		3		$\mu\text{s}$
$t_{OSCST}^{(1)}$	1	10		ms

(1) 依赖于晶体/振荡器和电路板设计



A. 复位后，BOOTROM引导代码对引导模式引脚采样。基于引导模式引脚的状态，引导代码分支到目标内存或引导代码函数。如果引导代码在上电之后执行(在调试器环境中)，则引导代码执行时间基于当前SYSCLKOUT速度。SYSCLKOUT将基于用户环境，可以启用或不启用PLL。

图 4-4 热复位

下图示例展示了写入 PLLCR 寄存器的效果。在第一阶段， $PLLCR = 0x0004$ ， $SYSCLKOUT = OSCCLK \times 2$ 。然后用  $0x0008$  写入 PLLCR。就在 PLLCR 寄存器写入之后，PLL 锁定阶段开始。在此阶段， $SYSCLKOUT = OSCCLK/2$ 。锁相环锁定完成后， $SYSCLKOUT$  反映新的工作频率  $OSCCLK \times 4$ 。

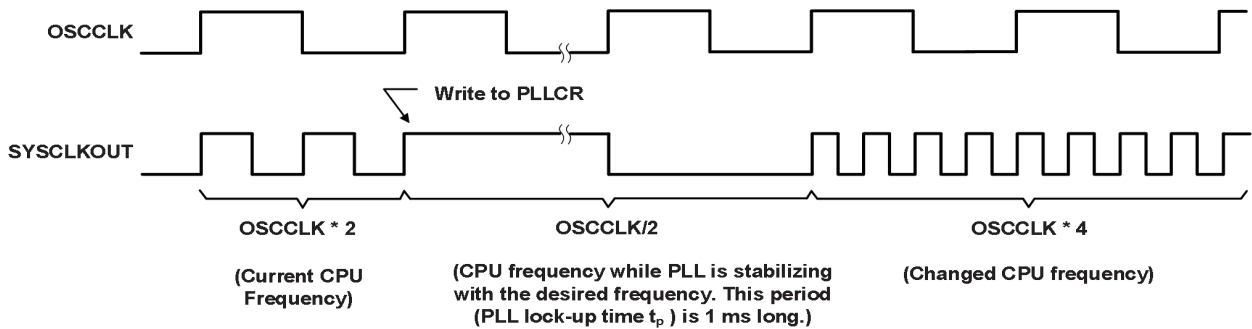


图 4-5 写入 PLLCR 寄存器的效果示例

## 4.9 时钟规格

### 4.9.1 器件时钟表

本节介绍 AVP32F069 微系统控制单元各种时钟选项的时序要求和开关特性。列出了各种时钟周期。

#### 4.9.1.1 时钟表和术语 (90MHz 设备)

		最小值	典型值	最大值	单位
SYSCLKOUT	$t_c(\text{SCO})$ 系统周期时间	11.11		500	ns
	频率	2		90	MHz
LSPCLK <sup>(1)</sup>	$t_c(\text{LCO})$ 低速时钟周期	11.11	44.4(2)		ns
	频率		22.5(2)		MHz
ADC时钟	$t_c(\text{ADCCLK})$ ADC时钟周期	22.22			ns
	频率			45	MHz

- (1) 更低的低速时钟 (LSPCLK) 可以减少器件功耗。
- (2) 当系统时钟为90MHz时, 这个是复位后默认值。

#### 4.9.1.2 器件时钟要求/特性

		最小值	典型值	最大值	单位
片上振荡器(X1/X2脚)	$t_c(\text{SCO})$ 系统周期时间	50		200	ns
	频率	5		20	MHz
外部振荡器/时钟源(XCLKIN脚)—PLL使能	$t_c(\text{CI})$ 时钟周期C8	33.3		200	ns
	频率	5		30	MHz
外部振荡器/时钟源(XCLKIN脚)—PLL禁用	$t_c(\text{CI})$ 时钟周期C8	11.11		250	ns
	频率	4		90	MHz
LIMP模式SYSCLKOUT(带2分频使能)	频率范围		1 到 5		MHz
XCLKOUT	$t_c(\text{XCO})$ 时钟周期C1	44.44		2000	ns
	频率	0.5		22.5	MHz
PLL锁定时间 <sup>(1)</sup>	$t_p$			1	ms

- (1) 寄存器PLLLOCKPRD的值必需是周期OSCCLK的倍数。如果使用内部振荡器(10MHz)作为时钟源, 则寄存器PLLLOCKPRD必须写入最小值10,000。

### 4.9.1.3 内部 OSC 特性 (INTOSC1/INTOSC2)

参数	最小值	典型值	最大值	单位
30°C下的内部振荡器1 <sup>(1)(2)</sup>	频率	10.000		MHz
30°C下的内部振荡器2 <sup>(1)(2)</sup>	频率	10.000		MHz
粗调步长		55		kHz
微调步长		14		kHz
温漂 <sup>(3)</sup>		3.03	4.85	kHz/°C
V <sub>DD</sub> 电压漂移 <sup>(3)</sup>		175		Hz/mV

- (1) 振荡器频率会随温度变化, 需补偿振荡器温度漂移。  
 (2) 仅当启用 VREG (VREGENZ = VSS)时, 才能确保频率范围。  
 (3) 内部振荡器的输出频率与温度和电压 (V<sub>DD</sub>) 相关。

## 4.9.2 时钟要求和特性

### 4.9.2.1 XCLKIN 时序要求 (启用 PLL)

No.	参数	最小值	最大值	单位
C9	t <sub>f(CI)</sub> XCLKIN下降时间		6	ns
C10	t <sub>r(CI)</sub> XCLKIN上升时间		6	ns
C11	t <sub>w(CIL)</sub> XCLKIN低电平脉宽占空比(t <sub>c(OSCCLK)</sub> )	45%	55%	
C12	t <sub>w(CIH)</sub> XCLKIN高电平脉宽占空比(t <sub>c(OSCCLK)</sub> )	45%	55%	

### 4.9.2.2 XCLKIN 时序要求 (禁用 PLL)

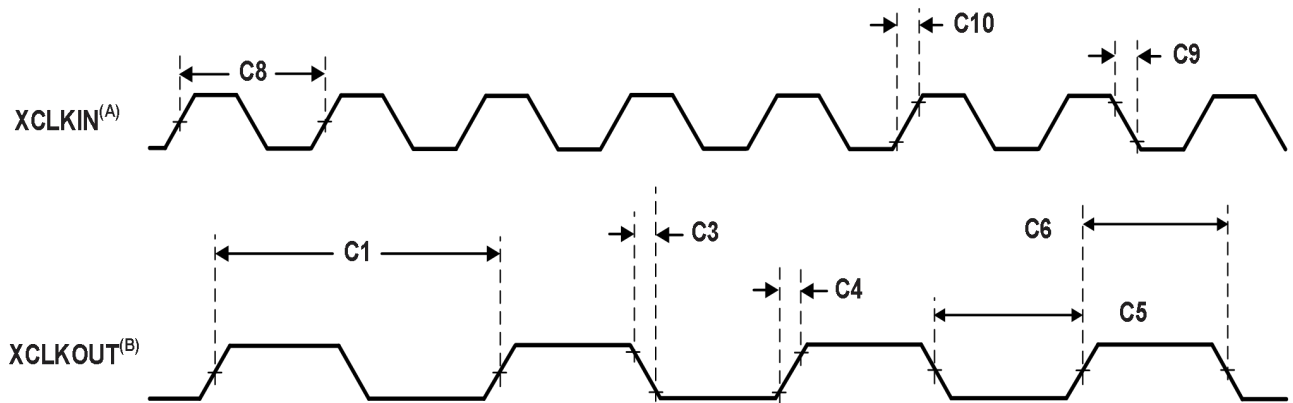
No.	参数	最小值	最大值	单位
C9	t <sub>f(CI)</sub> XCLKIN下降时间	最高20MHz	6	ns
		20MHz到90MHz	2	ns
C10	t <sub>r(CI)</sub> XCLKIN上升时间	最高20MHz	6	ns
		20MHz到90MHz	2	ns
C11	t <sub>w(CIL)</sub> XCLKIN低电平脉宽占空比(t <sub>c(OSCCLK)</sub> )	45%	55%	
C12	t <sub>w(CIH)</sub> XCLKIN高电平脉宽占空比(t <sub>c(OSCCLK)</sub> )	45%	55%	

### 4.9.2.3 XCLKOUT 开关特性 (PLL 旁路或启用)

在推荐操作条件范围内 (除非另有说明) (1)(2)

No.	参数	最小值	最大值	单位
C3	t <sub>f(XCO)</sub> XCLKOUT下降时间		5	ns
C4	t <sub>r(XCO)</sub> XCLKOUT上升时间		5	ns
C5	t <sub>w(XCOL)</sub> XCLKOUT低电平脉宽持续时间	H - 2	H + 2	ns
C6	t <sub>w(XCOH)</sub> XCLKOUT高电平脉宽持续时间	H - 2	H + 2	ns

- (1) 假定这些参数是在40pF负载的情形下。  
 (2) H = 0.5t<sub>c(XCO)</sub>



- A. XCLKIN与XCLKOUT的关系取决于所选的分频因子。所示波形关系仅用于说明时序参数，可能因实际配置而不同。  
 B. 配置XCLKOUT是为了反映SYSCLKOUT。

图 4-6 时钟时序图

## 4.10 Flash 时序

### 4.10.1 Flash/OTP 耐久度

		擦/写时温度(1)	最小值	典型值	最大值	单位
N <sub>f</sub>	Flash阵列耐久度(擦后写循环)	0°C至105°C环境温度	20000	50000		次数
N <sub>OTP</sub>	OTP阵列耐久度(写循环)	0°C至30°C环境温度			1	次数

(1) 在所示温度范围之外的擦/写操作虽无规定，但可能影响耐久次数。

### 4.10.2 Flash/OTP 访问时序

参数		最小值	最大值	单位
t <sub>a</sub> (fp)	Flash 页面访问时间	36		ns
t <sub>a</sub> (fr)	Flash 随机访问时间	36		ns
t <sub>a</sub> (OTP)	OTP 访问时间	60		ns

### 4.10.3 Flash 数据保留时间

参数	测试条件	最小值	最大值	单位
t <sub>retention</sub>	数据保留时间	T <sub>J</sub> =55°C	10	年

表 4-1 不同主频下最小 Flash/OTP 等待周期

SYSCLOCKOUT(MHz)	SYSCLOCKOUT(ns)	页面等待状态	随机等待状态	OTP等待状态
90	11.11	3	3	5
80	12.50	2	2	4
70	14.29	2	2	4
60	16.67	2	2	3
55	18.18	1	1	3
50	20.00	1	1	2
45	22.22	1	1	2
40	25.00	1	1	2
35	28.57	1	1	2
30	33.33	1	1	1

(1) 页面和随机等待状态必须≥1。

Flash 页面和随机等待状态的计算公式如下：

Flash 页面等待状态 =  $\lceil \frac{t_{a(f-p)}}{t_{c(SCO)}} - 1 \rceil$  向上舍入到下一个最大整数或 1，以较大者为准；

Flash 随机等待状态 =  $\lceil \frac{t_{a(f-r)}}{t_{c(SCO)}} - 1 \rceil$  向上舍入到下一个最大整数或 1，以较大者为准；

OTP 等待状态的公式如下：

OTP 等待状态 =  $\lceil \frac{t_{a(OTP)}}{t_{c(SCO)}} - 1 \rceil$  向上舍入到下一个最大整数或 1，以较大者为准。

## 5 详细说明

### 5.1 概述

#### 5.1.1 CPU

AVP32F069 是 AVP32 浮点 DSP 平台的成员之一，采用与现有 AVP32 系列产品相同的 32 位浮点架构。每个 AVP32 和 ADP32 的控制器，包括 AVP32F069 产品，都是一个非常高效的 C/C++ 引擎，使用户不仅能够用高级语言开发他们的系统控制软件，而且还能够使用 C/C++ 开发数学算法。该设备在单片机的数学任务上的效率与在通常由微控制器设备处理的系统控制任务上的效率一样高。这种效率消除了在许多系统中需要第二个处理器。32×32 位 MAC 64 位处理能力使控制器能够有效地处理更高的数值分辨率问题。再加上快速中断响应和自动保存关键寄存器的上下文，从而使设备能够以最小的延迟服务于许多异步事件。该设备有一个 8 级深度保护的管道与管道内存访问。这种流水线使它能够高速执行，而不诉诸于昂贵的高速内存。特殊的分支展望硬件最小化了条件不连续性的延迟。特殊的存储区、有条件的操作进一步提高了性能。

#### 5.1.2 控制环加速器 (CLA)

CLA 是一个单精度 (32 位) 浮点协处理单元，通过并行处理方式，扩展了 AVP32 的 CPU 处理能力。CLA 是一个独立的处理器，具有自己的总线结构、取码机制和流水线。可以指定 8 个独立的 CLA 任务或例程。每个任务均可由软件或外围设备启动，如 ADC、ePWM、eCAP、eQEP 或 CPU-Timer0。CLA 在每次只执行一项任务。当一个任务完成时，主 CPU 通过一个中断通知给 PIE，CLA 自动开始下一个最高优先级的待定任务。CLA 可以直接访问 ADC 结果寄存器、ePWM、RPWM、eCAP 和 eQEP 寄存器。专用消息 RAM 提供在主 CPU 和 CLA 之间传递额外数据的方法。

#### 5.1.3 维特比，复数，CRC 单元 (VCU)

VCU 通过为目标复杂数学、维特比解码和 CRC 计算添加额外的装配指令，提高了设备的处理能力。VCU 指令加速了许多应用程序，包括以下内容：

- 用于电力线载波PRIME和G3标准的正交频分复用标准(OFDM)
- 短程雷达复杂数学计算
- 功率计算
- 存储器和数据通信数据包检查(CRC)

VCU 功能包括:

- 支持循环冗余检查(CRCs)的指令, 这是一个多项式代码校验和。
  - CRC8
  - CRC16
  - CRC32
- 支持维特比解码器的灵活软件实现的指令
  - 对于1/2或1/3的代码率的分支度量计算
  - 加法/比较选择或维特比蝴蝶在五个周期的每个蝴蝶
  - 每个阶段有三个周期的跟踪
  - 很容易支持在PRIME和G3标准中使用的K=7的约束长度
- 复杂数学算术单元
  - 单个循环加法或减法
  - 2周期乘法
  - 2周期乘法累积(MAC)
  - 单周期重复MAC
- 独立寄存器空间

### 5.1.4 内存总线（哈佛总线架构）

与许多 MCU 类型的设备一样，多个总线用于在存储器和外设和 CPU 之间移动数据。内存总线体系结构包含一个程序读总线、数据读总线和数据写总线。该程序读取总线由 22 条地址线和 32 条数据线组成。数据读写总线分别由 32 条地址线和 32 条数据线组成。32 位宽的数据总线支持单个周期的 32 位操作。多总线架构，通常称为哈佛总线，使内核能够在在一个周期内获取指令、读取数据值和写入数据值。连接到内存总线的所有外设和内存都会优先考虑内存访问。通常，内存总线访问的优先级可以概括如下：

最高：	数据写入	（不能同时发生数据和程序写入操作存储总线）
	程序写入	（在内存总线上不能同时发生数据和程序写入操作）
	数据读取	
	程序读取	（在内存总线上不能同时读取程序和获取）
最低：	取指令	（同时进行的程序读取和获取不能发生在程序上存储总线）

### 5.1.5 外设总线

为了使外设能够在 ADP32 和 AVP32 系列设备之间迁移，这些设备采用外围总线标准进行外围互连。外围总线桥将构成处理器内存总线的各种总线复用成由 16 条地址线、16 或 32 条数据线和相关控制信号组成的单个总线。支持三种版本的外围设备总线。一个版本仅支持 16 位访问（称为外围帧 2）。另一个版本同时支持 16 位和 32 位访问（称为外围帧 1）。

### 5.1.6 实时 JTAG 和分析

该设备实现了标准的 IEEE 1149.1（IEEE 标准 1149.1-1990 标准测试访问端口和边界扫描架构）JTAG 接口进行电路调试。此外，该设备支持实时操作模式，允许在处理器运行、执行代码和服务中断时修改内存、外设和寄存器位置的内容。用户还可以通过非时间关键代码，同时在不受干扰的情况下实现对时间关键中断的服务。该设备在 CPU 内的硬件中实现了实时模式，不需要软件监视器。此外，



还提供了特殊的分析硬件，允许设置硬件断点或数据/地址监视点，并在发生匹配时生成各种用户可选择的中断事件。

### 5.1.7 Flash

AVP32F069 包含 128K×16 的嵌入式 FLASH 内存，分成 8 个 16K×16 扇区。还包含一个 1K×16 的 OTP 内存，地址范围为 0x3D7800 到 0x3D7BF9。用户可以单独删除、编程和验证 FLASH 扇区，同时保留其他扇区。但是，不支持在 FLASH 或 OTP 的一个扇区内执行擦除或编程其他扇区的 FLASH 算法。提供了特殊的内存流水线，使 FLASH 模块能够实现更高的性能。FLASH/OTP 被映射到程序和数据空间；因此，它可以用于执行代码或存储数据信息。地址 0x3F7FF0 到 0x3F7FF5 是为数据变量保留的，不应包含程序代码。

---

#### 注意

应用程序可以配置 Flash 和 OTP 等待状态。通过将 FLASH 配置为更长等待时间，允许应用程序在较慢的频率下运行。

通过在 FLASH 选项寄存器中启用 FLASH 管道模式，可以提高 FLASH 的有效性能。启用这种模式后，线性代码执行的有效性能将比仅通过等待状态配置所显示的原始性能要快得多。当使用 FLASH 管道模式时的准确性能增益取决于应用程序。

---

### 5.1.8 M0/M1 SARAMs

所有设备都包含这两个单存取内存块，每个内存大小为 1K×16。重置时，堆栈指针指向 M1 块的开始地址。M0 和 M1 块被映射到程序和数据空间。因此，用户可以使用 M0 和 M1 来执行代码或数据变量。该分区将在链接器中执行。设备向程序员提供了一个统一的内存映射，这使得在高级语言中进行编程变得更加容易。

### 5.1.9 L4 SARAM 及 L0/L1/L2/L3/L5/L7/L8 DPSARAMs

该设备包含多达 48K×16 的单访问 RAM，这块内存被映射到程序和数据空间。L0 的大小是 2K；L1 和 L2 大小各为 1K；L3 大小是 4K；L4、L5、L6、L7 和 L8 各为 8K 大小。L0、L1 和 L2 与 CLA 共享；CLA 可以使用这些内存作为其数据空间；L3 与 CLA 共享，CLA 可以使用它作为其程序空间；L5、L6、L7 和 L8 与 DMA 共享，DMA 可以使用它们作为其数据空间。DPSARAM 是指这些块的双端口配

置。

### 5.1.10 引导 ROM

引导 ROM 是用引导加载软件进行工厂编程的。提供了启动模式信号来告诉引导程序在开机时使用哪种启动模式。用户可以选择正常引导或从外部连接下载新软件, 或选择 Flash/ROM 内部的引导程序。

BootROM 还包含标准的表, 如 SIN/COS 波形, 用于与数学相关的算法。

表 5-1 引导模式选择

MODE	GPIO37/TDO	GPIO34/COMP2OUT/COMP3OUT	$\overline{\text{TRST}}$	MODE
3	1	1	0	GetMode
2	1	0	0	Wait
1	0	1	0	SCI
0	0	0	0	并行IO
EMU	X	X	1	仿真引导

#### 5.1.10.1 调试引导

当连接了调试探测器时, GPIO37/TDO 引脚不能用于引导模式的选择。在这种情况下, 引导 ROM 检测到调试探测器已连接, 并使用 PIE 向量表中两个保留的 SARAM 位置的内容来确定引导模式。如果任一个位置的内容无效, 则使用“Wait”引导选项。所有的引导模式选项都可以在仿真调试引导中被访问。

#### 5.1.10.2 Get Mode

GetMode 选项的默认行为是引导到 flash。通过对 OTP 中的两个位置进行编程, 可以将此行为更改为另一个引导选项。如果 OTP 两个位置中任何一个的内容无效, 则引导到 flash。可以指定以下加载器之一: SCI、SPI、I2C、CAN 或 OTP。

#### 5.1.10.3 Bootloader 使用的外设引脚

下表显示了每个外围引导加载程序使用哪些 GPIO 引脚。应用中需确定这些引脚是否与您希望在本应用程序中使用的任何外设发生冲突。

**表 5-2 引导加载外围引脚**

引导方式	加载外围引脚
SCI	SCIRXDA (GPIO28) SCITXDA (GPIO29)
并行IO引导	Data (GPIO31,30,5:0) DSP Control (AIO6) Host Control (AIO12)
SPI	SPISIMOA (GPIO16) SPISOMIA (GPIO17) SPICLKA (GPIO18) $\overline{\text{SPISTEA}}$ (GPIO19)
I2C	SDAA (GPIO28) SCLA (GPIO29)
CAN	CANRXA (GPIO30) CANTXA (GPIO31)

### 5.1.11 安全

器件支持很高安全性，保护用户固件不被逆向工程。该安全功能是一个 128 位的密码（硬编码为 16 个等待状态），用户可将其编程到 FLASH 中。一个代码安全模块(CSM)用于保护 FLASH/OTP 和 L0/L1 SARAM 内存块。该安全功能防止未经授权的用户通过 JTAG 端口查看内存内容，或尝试加载破解程序以导出安全内存的数据。要启用对安全地址的访问，用户必须编写正确的 128 位 KEY 值，且确保该值与 FLASH 中密码位置存储的值相匹配。

除 CSM 外，还有调试代码安全逻辑(ECSL)防止未经授权的用户进行单步操作安全代码。在调试探针连接时，对 CSM 安全内存的任何代码或数据访问都将触发 ECSL 并断开调试探针连接。为了允许安全代码的调试，同时保持 CSM 对安全内存读取的保护，用户必须将正确的值写入密钥寄存器的低 64 位 (KEY0-KEY3)，该值与 FLASH 中密码位置低 64 位相匹配，且必须执行对 FLASH 中所有 128 位密码的伪读取。如果密码位置 (PWL0-PWL3) 的低 64 位均为 1（未编程），则不需要匹配密钥值。在调试安全代码期间，可以进行单步操作。但是，在编辑器窗口中仍无法看到安全存储器的实际内容。

当向已连接 JTAG 调试探针的加密器件供电时，CPU 开始运行，并执行一条指令以访问受保护区，从而触发 ECSL 导致 JTAG 电路停用。在这种情况下，主机（比如运行调试软件或 FLASH 编程软

件的计算机) 将无法与芯片建立连接。

解决方案是使用“Wait”引导选项。这个模式下，器件围绕软件断点循环，以允许调试探针连接而不触发安全保护。这些器件不支持复位模式下的硬件等待。

---

**注意**

- 当编程了代码安全密码时，从0x3F 7F80到0x3F 7FF5的所有地址都不能用作程序代码或数据。这些位置必须被编程为0x0000。
  - 如果将来可能需要通过JTAG对安全设备进行重新编程，则重要的是以这样一种方式设计电路板，即设备在通电时（当重新编程得到保证时）可以处于等待引导模式。否则，如前所述，ECSL可能停用JTAG电路并阻止与设备的连接。如果在现场重新配置设备以等待引导模式不可行，则必须在固件中实现某种机制，以检测何时需要固件更新。然后，代码可以分支到BootROM中所需的引导加载程序。它还可以分支到等待引导模式，此时可以连接JTAG调试探针，设备不安全，并通过JTAG本身完成编程。
  - 如果不使用代码安全特性，则可以将地址0x3F 7F80到0x3F 7FEF用于代码或数据。地址0x3F 7FF0到0x3F 7FF5是为数据保留，不应包含程序代码。
  - 128位密码（在0x3F 7FF8至0x3F 7FFF）不能被编程为零。这样做将永久锁定该设备。
- 

**CSM 免责声明**

本器件上包含的代码安全模块（CSM）旨在密码保护存储在相关内存（ROM 或 FLASH）中的数据，并由进芯电子根据其标准条款和条件，符合适用的保修规范。

然而，进芯电子不保证或声明 CSM 不能被破坏或破坏存储在相关内存中的数据不能通过其他方式访问。此外，除上文规定外，进芯电子对 CSM 或本器件的操作不作任何保证或陈述，包括任何对适销性或适合于特定用途的隐含保证。

在任何情况下，进芯电子均不对因您使用 CSM 而产生的任何后果性、特殊性、间接性、附带性或惩罚性损害承担责任，无论您是否告知进芯电子此类损害的可能。损害包括且不限于数据损失、商誉损失、使用损失或业务中断等其他经济损失。

---

### 5.1.12 外设中断扩展块 (PIE)

PIE 用于将多个中断源多路复用成一组较小的中断输入，模块最多可支持 96 个外设中断。在 AVP32F069 上，其中的 72 个被外设使用，96 个中断被分为 8 组，每组被送到 12 条 CPU 中断线 (INT1-INT12) 中的 1 条)。96 个中断有其自身向量表，存储在专用 RAM 块中，用户可以覆盖 RAM 以重定位该向量表。中断服务时，CPU 会自动获取向量。需要 8 个 CPU 时钟周期来获取向量并保存关键 CPU 寄存器。因此，CPU 可以快速响应中断事件。中断的优先级是由硬件和软件来控制的，每个单独的中断都可以在 PIE 块内启用或禁用。

### 5.1.13 外部中断 (XINT1 到 XINT3)

器件支持三个可屏蔽外部中断 (XINT1-XINT3)。每个中断皆可选择上升沿、下降沿或双沿触发,也可以启用或禁用。这些中断还包含一个 16 位的自由递增计数器,当检测到一个有效的中断边沿时,计数器被重置为零。此计数器可用于准确标记中断时间。外部中断没有专用引脚。XINT1、XINT2 和 XINT3 中断可以接受任何来自 GPIO0-GPIO31 引脚的输入。

#### 5.1.14 内部振荡器、振荡器和锁相环

该器件可以由两个内部振荡器中的任何一个、一个外部振荡器或连接到芯片内起振电路上的晶体提供计时。提供一个 PLL 支持 16 种输入时钟倍频。PLL 倍频系数可以在程序中动态更改,在用户需要减小功耗时,降低工作频率。PLL 可以设置为旁路模式。第二个 PLL(PLL2)提供给了 HRCAP 模块。

#### 5.1.15 看门狗

器件包含两个看门狗:一个 CPU 看门狗用来监视内核执行;一个 NMI 看门狗用来检测时钟丢失。用户程序必须在一定时间内定期重置 CPU 看门狗计数器;否则,CPU 看门狗将产生一个复位信号给处理器。如果有必要,可以禁用 CPU 看门狗。NMI 看门狗仅在时钟故障时启动,并产生中断或器件复位。

#### 5.1.16 外设时钟

每个外设的时钟都可以启用或禁用,可关闭外设时钟以减少功耗。此外,串行端口(I2C 除外)的系统时钟可以从 CPU 时钟分频。

#### 5.1.17 低功耗模式

AVP32F069 芯片是全静态 CMOS 器件。提供三种低功率模式:

(1) IDLE: 将CPU置于低功耗模式。可以选择性的关闭设时钟,并且只有那些在IDLE期间工作的外设才处于工作状态。启用来自活动外设或看门狗定时器的中断将使处理器从IDLE模式中唤醒。

(2) STANDBY: 关闭CPU和外设时钟。此模式让振荡器和PLL保持正常工作。一个外部中断事件

将唤醒处理器和外设，并在检测到中断事件后的下一个有效周期开始执行。

- (3) HALT: 这种模式基本上关闭器件，并将其置于最低功耗模式。如果使用内部振荡器作为时钟源，默认情况下，HALT模式会将其关闭。为了防止关闭内部振荡器，需要用到寄存器CLKCTL中的INTSOCnHALTI位。因此，内部振荡器可用于该模式下CPU看门狗的计时。如果使用片上晶体振荡器作为时钟源，该模式下将其关闭。复位、外部信号(通过GPIO引脚)或CPU看门狗可以将芯片从HALT模式中唤醒。

在尝试将 CPU 置于 HALT(暂停)或 STANDBY(待机)状态之前，需确保 CPU 时钟(OSCCLK)和看门狗时钟来自同一时钟源。

### 5.1.18 外设框架 0/1/2/3 (PFn)

该设备将外围设备分为四个部分。外围设备的映射如下：

	PIE:	PIE中断使能和控制寄存器再加PIE向量表
	Flash:	FLASH等待状态寄存器
PF0:	Timers:	CPU定时器0、1、2的寄存器
	CSM:	代码安全模块KEY寄存器
	ADC:	ADC结果寄存器
PF1:	CLA:	控制环加速器寄存器和消息RAMs空间
	GPIO:	GPIO复用器配置和控制寄存器
	eCAN:	eCAN配置和控制寄存器
	SYS:	系统控制寄存器
PF2:	SCI:	串行通信接口(SCI)控制和RX/TX寄存器
	SPI:	串口端口接口(SPI)控制和RX/TX寄存器
	ADC:	ADC状态、控制和配置寄存器
	I2C:	内部集成电路模块(I2C)寄存器
	XINT:	外部中断寄存器
PF3:	McBSP:	多通道缓冲串行端口(McBSP)寄存器
	ePWM:	增强型脉宽调制器模块寄存器
	eCAP:	增强型捕获模块寄存器
	eQEP:	增强型正交编码器脉冲模块寄存器
	Comparators:	比较器模块
	USB:	通用串行总线模块寄存器

### 5.1.19 通用输入/输出复用器

大多数外设复用了通用输入/输出(GPIO)引脚。这使得用户如果不使用某外设功能,就可以将其相应引脚用作 GPIO。复位时,GPIO 引脚被配置为输入,用户可独立设置每个引脚为 GPIO 模式或外设模式。对于特定输入,用户还可以选择输入确认周期数。这是为了过滤不必要的噪声毛刺。GPIO 信号也可用于使 CPU 脱离特定的低功耗模式。

### 5.1.20 32 位 CPU 定时器 (0/1/2)

CPU 定时器 0、1 和 2 是相同的 32 位定时器,具有可预置周期和 16 位时钟预分频。定时器有一个 32 位倒数计数寄存器,当计数器达到零时,它会产生一个中断。计数器以 CPU 时钟除以设置的预分频值递减。当计数器达到零时,它将自动重新加载 32 位周期值。

CPU 定时器 0 连接到 PIE,用于一般用途。CPU 定时器 1 也是通用的,可以连接到 CPU 的 INT13。CPU 定时器 2 是为实时操作系统保留的。CPU 定时器 2 已连接到 CPU 的 INT14。如果不使用实时操作系统,CPU 定时器 2 也可通用。

CPU 定时器 2 可由以下任一项计时:

- SYSCLKOUT(默认)
- 内部振荡器1(INTOSC1)
- 内部振荡器2(INTSOC2)
- 外部时钟源

### 5.1.21 控制外设

器件支持用于嵌入式控制和通信的以下外设:

ePWM: 增强型 PWM 外设支持独立、互补 PWM 生成,前沿、后沿可调死区,锁存、逐周期跳闸机制。一些 PWM 引脚支持 HRPWM 高分辨率占空比和周期特性。器件上的 1 型模块还支持增加死区分辨率、增强 SOC 和中断生成以及高级触发,包括基于比较器输出的跳闸功能。

eCAP: 增强型捕获外设使用 32 位时基, 并在连续/单次捕获模式下最多可注册四个可编程事件。

该外设还可以配置产生一个辅助 PWM 信号。

eQEP: 增强型 QEP 外设使用 32 位位置计数器, 支持使用捕获单元的低速测量和使用 32 位单元计时器的高速测量。该外设有一个看门狗定时器, 用来检测电机失速, 并输入错误检测逻辑, 用于识别 QEP 信号中的同时边缘转换。

ADC: ADC 是一个 12 位转换器。最多有 16 个单端通道, 具体取决于器件封装。ADC 还包含两个用于同时采样的采保单元。

比较器: 每个比较器包括一个模拟比较器以及一个内部 10 位基准, 用于提供比较器的一个输入。

HRCAP: 高分辨率捕获外设通过关闭时钟 HCCAPCLK 的 16 位计数器以正常捕获模式运行, 或通过使用内置校准逻辑和校准库以高分辨率捕获模运行。

### 5.1.22 串行端口外设

器件支持以下串行通信外设:

SPI: SPI 是一个高速、同步串行 I/O 端口, 允许编程 1 至 16 位长度的串行比特流以可编程比特传输速率移入和移出器件。通常 SPI 用于 MCU 与外部器件或其他处理器之间的通信。典型应用包括通过移位寄存器、显示驱动器和 ADC 等器件进行外部 I/O 或外围扩展。SPI 的主/从操作支持多设备通信。SPI 包含一个 4 级接收和传输 FIFO, 用于减少中断服务开销。

SCI: 串行通信接口是一个 2 线异步串口, 通常称为 UART。SCI 包含一个 4 级接收和发送 FIFO, 用于减少中断服务开销。

I2C: 内部集成电路(I2C)模块提供 MCU 与符合飞利浦半导体 IC 总线(I2C 总线®)规范 2.1 版并通过 I2C 总线连接的其他设备之间的接口。连接到该 2 线串行总线的外部组件可以通过 I2C 模块向 MCU 发送或从 MCU 接收最多 8 位数据。I2C 包含 4 级接收和发送 FIFO, 用于减少中断服务开销。



eCAN: 这是 CAN 外设的增强版本, eCAN 支持 32 个邮箱、消息时间戳, 并符合 ISO

11898-1(CAN2.0B)规范。

McBSP: 多通道缓冲串行端口(McBSP)连接到 E1/T1 线路、用于调制解调器应用的电话质量编解

码器或高质量立体声音频 DAC 设备。DMA 支持 McBSP 接收和发送寄存器, 以显著减

少为该外设提供服务的开销。McBSP 模块可以根据需要配置为 SPI。

USB: 符合 USB2.0 规范的 USB 外设可用作全速(12Mbps)设备控制器, 或者用作全速(12Mbps)

或低速(1.5Mbps)主机控制器。该控制器共支持六个用户可配置端点, 所有这些端点都可以

通过 DMA 访问, 此外还有一个用于端点零的专用控制端点。所有发送或接收的数据包都缓

冲在 4KB 的专用端点存储器中。USB 外设支持所有三种传输类型: 控制、中断和批量传输。

## 5.2 VREG、BOR、POR

因为内核和 I/O 电路工作在不同的电压下，所以芯片内部集成了一个片上稳压器(VREG)和一个线性稳压器(LDO),VREG 用于从 VDDIO 电源产生 VDD 电压，而 LDO 用于从 VDD 电压上产生 1.1V 的内核电压(VDD11)。在芯片内部集成稳压器避免了在应用板上再使用外部稳压器所增加的成本和 PCB 面积。此外，内部上电复位 (POR) 与掉电复位 (BOR) 电路在上电及运行模式下监控 VDDIO、VDD 与 VDD11 电压。

### 5.2.1 片上稳压器 (VREG) 与 LDO

片上 VREG 从 VDDIO 电源上产生 VDD 电压，因此，尽管每个 VDD 引脚上都需要电容器来稳定产生的电压，但在使用过程中不需要向这些引脚提供工作电压。相反，如果应用中从外部给 VDD 引脚供电，则需要禁用 VREG。内部 LDO 从 VDD 电压上产生内核电压 VDD11，在芯片运行时始终工作，不能被关闭。

#### 5.2.1.1 使用片上稳压器 (VREG)

若使用片上 VREG，需将  $\overline{\text{VREGENZ}}$  引脚置低，并向 VDDIO 和 VDDA 引脚提供合适的工作电压。在这种情况下，VDD 电压将由 VREG 产生，每个 VDD 引脚最小需要 1.2 $\mu\text{F}$  电容，以正确调节 VREG。这些电容应尽可能靠近 VDD 引脚。不允许使用内部 VREG 驱动外部负载。

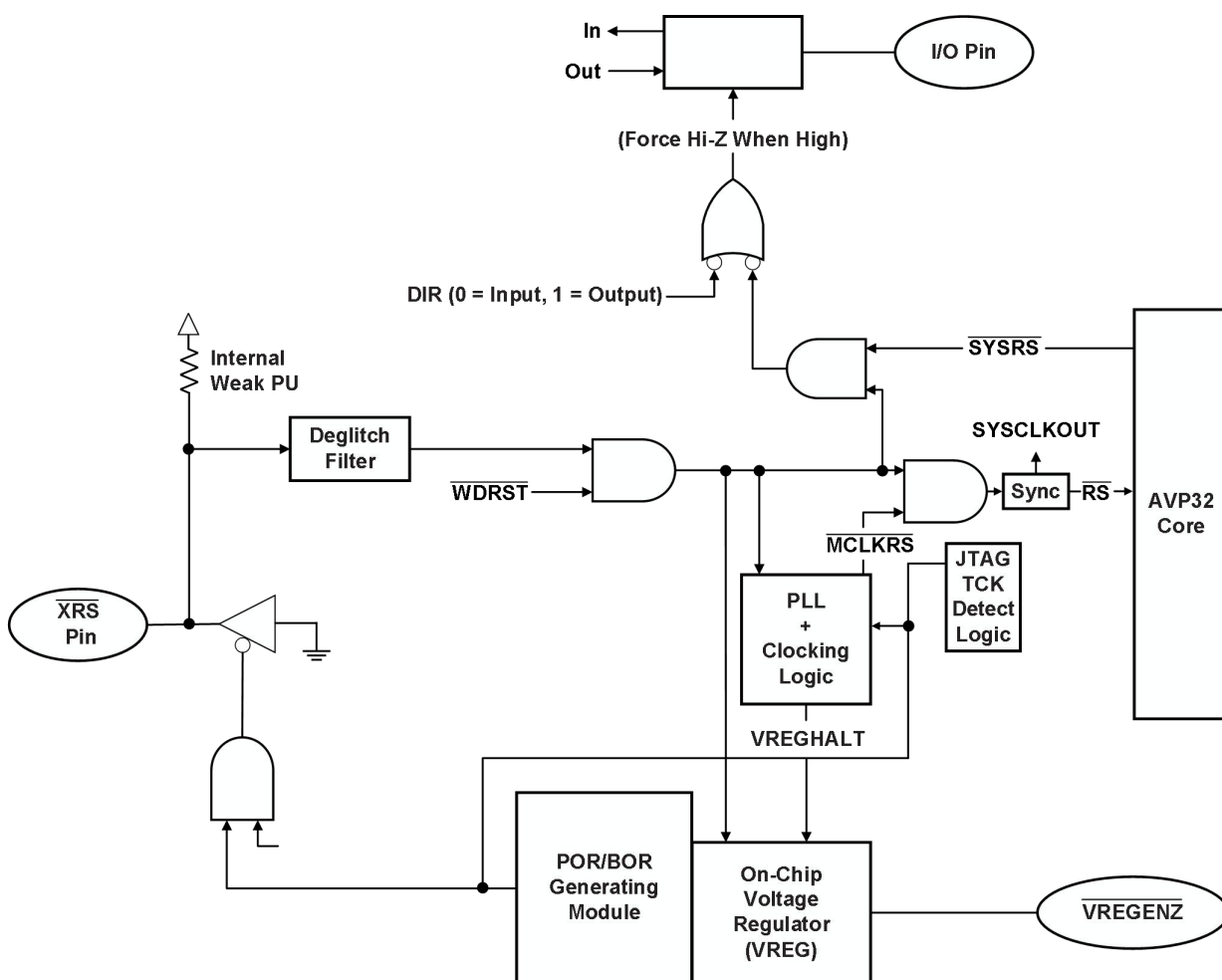
#### 5.2.1.2 禁用片上稳压器 (VREG)

为了省电，还可以禁用片上 VREG。将  $\overline{\text{VREGENZ}}$  引脚置高，禁用片上 VREG，然后使用更高效的外部稳压器向 VDD 引脚提供 VDD 电压，从而使内部 LDO 正常工作，产生稳定可靠的内核逻辑工作电压 VDD11。

### 5.2.2 片上通电复位 (POR) 和掉电复位 (BOR) 电路

通电复位(POR)和掉电复位(BOR)两个片上监控电路，消除了从应用板上监控供电源轨的负担。POR 的目的是在芯片整个通电过程中建立一个干净的复位。相比在设备运行期间用于监测 VDD 和

VDDIO 电压的 BOR 电路, POR 具有更宽松、更低的触发点。POR 功能始终存在于 VDDIO、VDD 和 VDD11 电压轨上。而 BOR 功能则是在器件上电初始化后,始终出现在 VDDIO 上;且在启用内部 VREG 时( $\overline{\text{VREGENZ}}$ 引脚拉低),始终作用于 VDD 与 VDD11 上。当其中任何一个电压低于各自的触发点时, POR 和 BOR 都会将  $\overline{\text{XRS}}$  引脚拉低。VDD 的 BOR 触发点和过压触发点在推荐的正常工作电压范围之外。如果应用关注过压或欠压条件对系统的影响,应添加外部电压监控器。下图展示了 VREG、POR 和 BOR 以及复位信号的连接关系。寄存器 BORCFG 中提供了一个控制位,可以禁用 VDDIO、VDD 和 VDD11 的 BOR 功能。



- A.  $\overline{\text{WDRST}}$ 是来自 CPU 看门狗的复位信号。
- B.  $\overline{\text{PBRST}}$ 是来自 POR/BOR 模块的复位信号。

图 5-1 VREG+POR+BOR+复位信号连接

## 5.3 系统控制

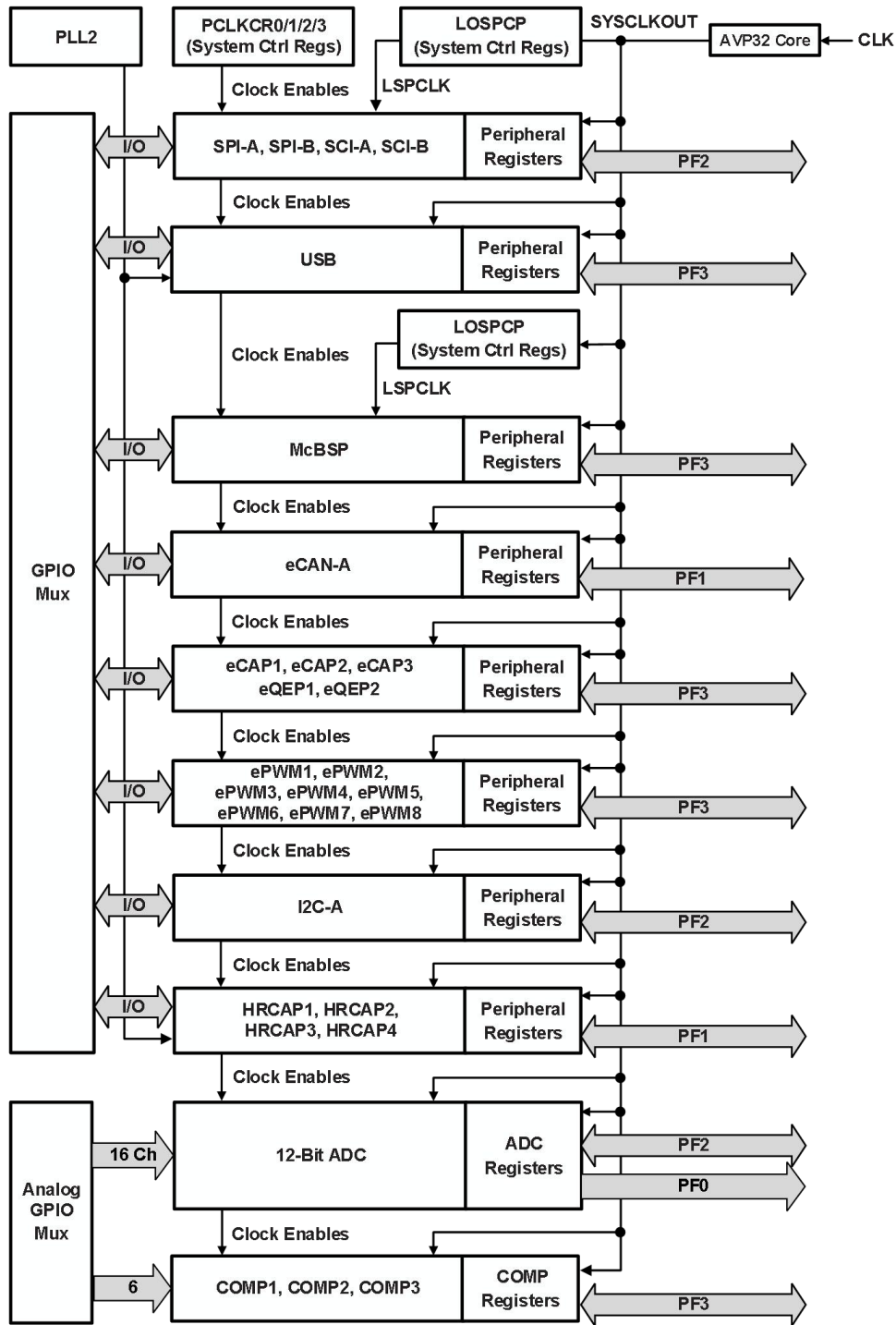
本节介绍了振荡器和时钟机制、看门狗的功能和低功耗模式。

表 5-3 PLL、时钟、看门狗和低功耗模式寄存器

名称	地址	大小(×16)	寄存器说明 <sup>(1)</sup>
BORCFG	0x00 0985	1	BOR Configuration Register
XCLK	0x00 7010	1	XCLKOUT Control
PLLSTS	0x00 7011	1	PLL Status Register
CLKCTL	0x00 7012	1	Clock Control Register
PLLLOCKPRD	0x00 7013	1	PLL Lock Period
INTOSC1TRIM	0x00 7014	1	Internal Oscillator 1 Trim Register
INTOSC2TRIM	0x00 7016	1	Internal Oscillator 2 Trim Register
PCLKCR2	0x00 7019	1	Peripheral Clock Control Register 2
LOSPCP	0x00 701B	1	Low-Speed Peripheral Clock Prescaler Register
PCLKCR0	0x00 701C	1	Peripheral Clock Control Register 0
PCLKCR1	0x00 701D	1	Peripheral Clock Control Register 1
LPMCR0	0x00 701E	1	Low-Power Mode Control Register 0
PCLKCR3	0x00 7020	1	Peripheral Clock Control Register 3
PLLCR	0x00 7021	1	PLL Control Register
SCSR	0x00 7022	1	System Control and Status Register
WDCNTR	0x00 7023	1	Watchdog Counter Register
WDKEY	0x00 7025	1	Watchdog Reset Key Register
WDCR	0x00 7029	1	Watchdog Control Register
JTAGDEBUG	0x00 702A	1	JTAG Port Debug Register
PLL2CTL	0x00 7030	1	PLL2 Configuration Register
PLL2MULT	0x00 7032	1	PLL2 Multiplier Register
PLL2STS	0x00 7034	1	PLL2 Lock Status Register
SYSCLK2CNTR	0x00 7036	1	SYSCLK2 Clock Counter Register
EPWMCFG	0x00 703A	1	ePWM DMA/CLA Configuration Register

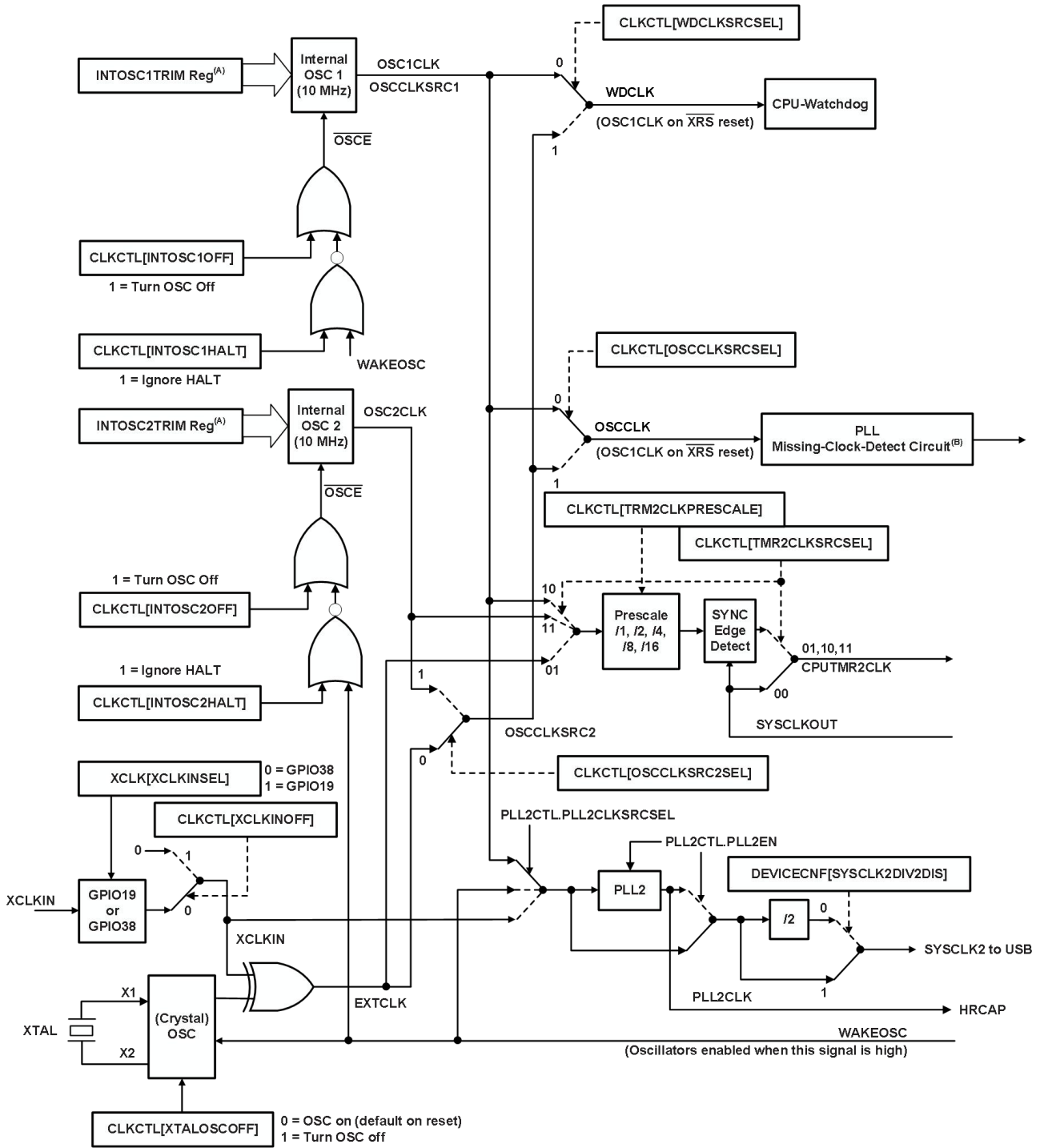
(1) 此表中的所有寄存器都受到 EALLOW 的保护。

下图展示多种时间域，图 5-3 显示多种时钟源(包括内部和外部)可提供给设备操作的时钟。



A. CLKIN 是输入 CPU 的时钟。CLKIN 通过 CPU 被输出到 SYSCLKOUT (也就是 CLKIN 和 SYSCLKOUT 同频率)。

图 5-2 时钟和复位域



- A. 从基于 OTP 校验函数加载寄存器。
- B. 详细的时钟检测逻辑参考。

图 5-3 时钟树

### 5.3.1 内部振荡器

AVP32F069 内部包含两个独立的零引脚振荡器。默认情况下，两个振荡器在上电时都处于打开状态，此时内部振荡器 1 是默认时钟源。为了节省功耗，用户可能会关闭未使用的振荡器。这些振荡器的中心频率由其各自的振荡器调整寄存器决定，作为启动 ROM 执行的一部分，伴随校准例程执行写入。

### 5.3.2 晶体振荡器选项

片上晶体振荡器 X1 和 X2 引脚是 1.8V 电平信号，绝不能施加 3.3V 电平信号。如果要将系统 3.3V 外部振荡器用作时钟源，则应仅将其连接到 XCLKIN 引脚。X1 引脚不能用作单端时钟输入，应配合 X2 一起连接晶体使用。

表 5-3-2 列出了外部石英晶体的典型规格（基波模式，并联谐振）。此外，ESR 范围 = 30Ω-150 Ω。对于表 5-3-2，CSHUNT 应小于或等于 5pF。

表 5-4 外部石英晶体的典型规格参数

频率(MHz)	R <sub>d</sub> (Ω)	C <sub>L1</sub> (pF)	C <sub>L2</sub> (pF)
5	2200	18	18
10	470	15	15
15	0	15	15
20	0	12	12

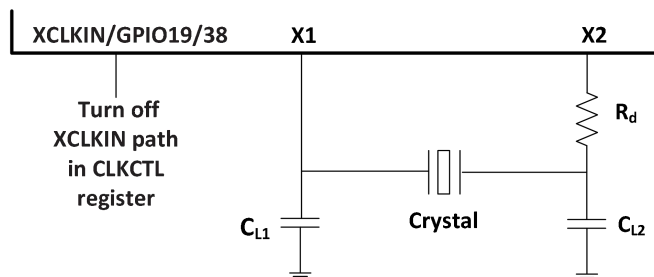


图 5-4 使用片内晶体振荡器

注意

1.  $C_{L1}$  和  $C_{L2}$  为电路板和元件的总电容，不包括 IC 和晶体。这个值通常大约是晶体负载电容值的两倍。
2. 晶体的负载电容在厂家的晶体规格中有说明。
3. 推荐客户其设备与 MCU 芯片关于谐振器/晶体供应商的业务。谐振器/晶体供应商有设备和专业知识来调整振荡电路。供应商还可以向客户提供适当的元件值，以便在整个操作范围内产生适当的启动和稳定性。

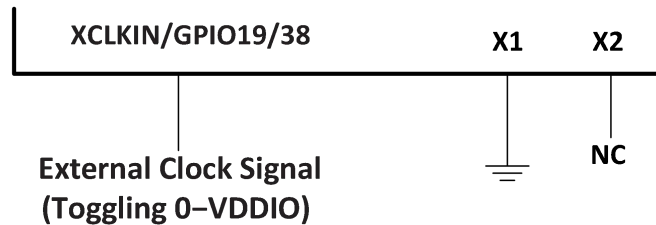


图 5-5 使用一个 3.3v 外部振荡器

### 5.3.3 基于 PLL 的时钟模块

这些设备有一个芯片上的、基于 PLL 的时钟模块。该模块为设备提供所有必要的时钟信号，以及对低功率模式进入的控制。PLL 有一个 5 位比率控制 PLLCR[DIV]来选择不同的 CPU 时钟速率。在写入 PLLCR 寄存器之前，应该禁用监视器模块。在 PLL 模块稳定后，看门狗模块可以重新启用（如果需要），这需要 1ms。输入时钟和 PLLCR[DIV]位的选择应使 PLL(VCOCLK)的输出频率至少为 50MHz。

表 5-5 PLL 设置

PLLCR[DIV]值 <sup>(1)(2)</sup>	SYSCLKOUT(CLKIN)		
	PLLSTS[DIVSEL] = 0或1 <sup>(3)</sup>	PLLSTS[DIVSEL] = 2	PLLSTS[DIVSEL] = 3
00000(PLL旁路)	OSCCLK/4(缺省值) <sup>(1)</sup>	OSCCLK/2	OSCCLK
00001	(OSCCLK×1)/4	(OSCCLK×1)/2	(OSCCLK×1)/1
00010	(OSCCLK×2)/4	(OSCCLK×2)/2	(OSCCLK×2)/1
00011	(OSCCLK×3)/4	(OSCCLK×3)/2	(OSCCLK×3)/1
00100	(OSCCLK×4)/4	(OSCCLK×4)/2	(OSCCLK×4)/1
00101	(OSCCLK×5)/4	(OSCCLK×5)/2	(OSCCLK×5)/1
00110	(OSCCLK×6)/4	(OSCCLK×6)/2	(OSCCLK×6)/1
00111	(OSCCLK×7)/4	(OSCCLK×7)/2	(OSCCLK×7)/1
01000	(OSCCLK×8)/4	(OSCCLK×8)/2	(OSCCLK×8)/1
01001	(OSCCLK×9)/4	(OSCCLK×9)/2	(OSCCLK×9)/1
01010	(OSCCLK×10)/4	(OSCCLK×10)/2	(OSCCLK×10)/1
01011	(OSCCLK×11)/4	(OSCCLK×11)/2	(OSCCLK×11)/1



PLLCR[DIV]值 <sup>(1)(2)</sup>	SYSCLKOUT(CLKIN)		
	PLLSTS[DIVSEL] = 0或1 <sup>(3)</sup>	PLLSTS[DIVSEL] = 2	PLLSTS[DIVSEL] = 3
01100	(OSCCLK×12)/4	(OSCCLK×12)/2	(OSCCLK×12)/1
01101	(OSCCLK×13)/4	(OSCCLK×13)/2	(OSCCLK×13)/1
01110	(OSCCLK×14)/4	(OSCCLK×14)/2	(OSCCLK×14)/1
01111	(OSCCLK×15)/4	(OSCCLK×15)/2	(OSCCLK×15)/1
10000	(OSCCLK×16)/4	(OSCCLK×16)/2	(OSCCLK×16)/1
10001	(OSCCLK×17)/4	(OSCCLK×17)/2	(OSCCLK×17)/1
10010	(OSCCLK×18)/4	(OSCCLK×18)/2	(OSCCLK×18)/1

- (1) PLL控制寄存器(PLLCR)和PLL状态寄存器(PLLSTS)通过XRS信号或看门狗重置为其默认状态仅重置。由调试器发出的重置或丢失的时钟检测逻辑没有作用。
- (2) 此寄存器受EALLOW保护。
- (3) 默认情况下，PLLSTS[DIVSEL]配置为/4(引导ROM会将此操作更改为/1)。PLLSTS[DIVSEL]在写入PLLCR之前必须是0，并且只有在PLLSTS[请]=1之后才应该更改。

表 5-6 CLKIN 分频选项

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

基于 PLL 的时钟模块提供了四种操作模式：

- INTOSC1(内部振荡器1)：这是片上的内部振荡器1。这可以为看门狗、内核和CPU定时器2提供计时。
- INTOSC2(内部振荡器2)：这是片上的内部振荡器2。也可以为看门狗、内核和CPU定时器2提供计时。INTOSC1和INTOSC2均可独立选择用于看门狗、内核和CPU定时器2。
- 晶体/谐振器：片上(晶体)振荡器使能连到芯片上的外部晶体/谐振器来提供时基。晶体/谐振器连接到X1/X2引脚，有些器件可能没有X1/X2引脚。
- 外部时钟源：如果未使用片上(晶体)振荡器，此模式下允许它被旁路。设备时钟由XCLKIN引脚上的外部时钟源输入产生。XCLKIN与GPIO19或GPIO38引脚复用。可通过寄存器XCLK中的XCLKINSEL位选择GPIO19或GPIO38作为XCLKIN输入。通过设置CLKCTL[XCLKINOFF]位

以禁用此时钟输入。如果不使用时钟源或将各个引脚用作GPIO，用户应该在引导时禁用。

更改时钟源之前，需确保目标时钟存在。如果时钟不存在，则在切换时钟之前必须使用 CLKSCCTL 寄存器禁用该时钟源。

表 5-7 PLL 配置模式

PLL模式	备注	PLLSTS[DIVSEL]	CLKIN和SYSCLKOUT
PLL关闭	通过设置PLLSTS寄存器中PLLOFF位，禁用PLL进入此模式。对于降低系统噪声和低功率运行非常有用。进入此模式前，必须先将PLLCR寄存器设置为0x0000(PLL旁路)。CPU时钟(CLKIN)直接从X1/X2、X1或XCLKIN输入。	0,1 2 3	OSCCLK/4 OSCCLK/2 OSCCLK/1
PLL旁路	PLL旁路是上电或外部复位( $\overline{XRS}$ )后的默认PLL配置。当PLLCR寄存器设置为0x0000或修改PLLCR寄存器后直到锁定到新频率时，选择此模式。在此模式下，PLL被旁路，但PLL并未关闭。	0,1 2 3	OSCCLK/4 OSCCLK/2 OSCCLK/1
PLL启用	通过向PLLCR寄存器中写入一个非零值n来实现的。写入PLLCR后，设备将切换到PLL旁路模式，直到PLL锁定。	0,1 2 3	OSCCLK $\times$ n/4 OSCCLK $\times$ n/2 OSCCLK $\times$ n/1

### 5.3.4 USB 和 HRCAP 的 PLL 模块(PLL2)

除主系统 PLL 外，器件还包含第二个 PLL(PLL2),用于 USB 和 HRCAP 外设计时。PLL 支持 1 到 15 倍频，并且在其输出上具有固定的 2 分频。

通过适当修改 PLL2CTL 寄存器中的 PLL2CLKSRCSEL 位,可以从以下三个时钟源对 PLL2 进行计时:

- INTOSC1(内部振荡器1): 内部振荡器1提供一个10MHz的时钟。如果用作HRCAP的时钟源，需频繁调用振荡器补偿程序。由于精度的要求，INTOSC1不能用作USB的时钟源。
- 晶体/谐振器: 晶体振荡器使能连到芯片上的外部晶体/谐振器来提供时基。晶体/谐振器连接到X1/X2引脚。
- 外部时钟源: 这种模式允许从连接到GPIO19或GPIO38的外部单端时钟源导出参考时钟。应适当设置XCLK寄存器中的XCLKINSEL位，以启用所选GPIO驱动XCLKIN。

#### 注意

为了 USB 模块的正常操作，应配置 PLL2 生成 120Mhz 时钟，然后除以 2 产生 USB 外设所需的 60MHz 时钟。HRCAP 支持 120MHz 的最大时钟输入频率。

### 5.3.5 输入时钟丢失(NMI 看门狗功能)

AVP32F069 可以从内部振荡器(INTOSC1\INTOSC2)、片上晶体振荡器或外部时钟输入中的任意一个进行计时。无论时钟源如何，在 PLL 启用和 PLL 旁路模式下，如果 PLL 的输入时钟消失，PLL 将输出跛行模式时钟。此跛行模式时钟继续以 1-5MHz 的典型频率为 CPU 和外设计时。

当跛行模式被激活时，会产生 CLOCKFAIL 信号，该信号被锁存为 NMI 中断。

根据 NMIRESETSEL 位的配置方式，可以立即触发对器件的复位，或者当设备溢出时，NMI 看门狗计数器可以发出复位。除此之外，还设置了丢失时钟状态(MCLKSTS)位。应用程序可以使用 NMI 中断来检测输入时钟故障并启动必要的纠正措施，例如切换到替代时钟源(如果可用)或启动系统的关闭程序。

如果软件未响应时钟故障条件，则 NMI 看门狗会在预编程的时间间隔后触发复位。下图显示了所涉及的中断机制。

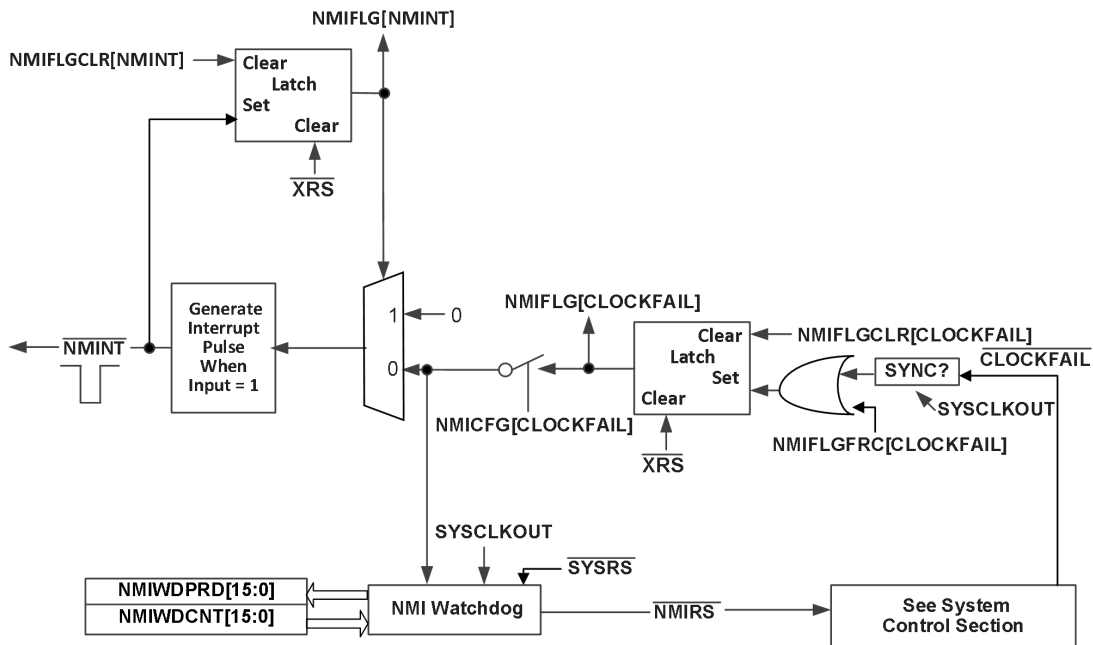


图 5-6 NMI 看门狗

### 5.3.6 CPU 看门狗模块

AVP32F069 拥有和 ADP32F12 以及 AVP32F335 一样的 CPU 看门狗模块。当 8 位看门狗递增计数器达到最大值时，该模块产生一个 512 振荡器时钟(OSCCLK)宽度的低脉冲输出。为了防止此情况，用户必须禁用看门狗或程序周期性向看门狗 KEY 寄存器写入 0x55+0xAA 序列，以重置看门狗计数器。看门狗模块内部的各个功能如下图所示。

正常情况下，当输入时钟存在时，CPU 看门狗计数器递减直到启动看门狗复位或 WDINT 中断。

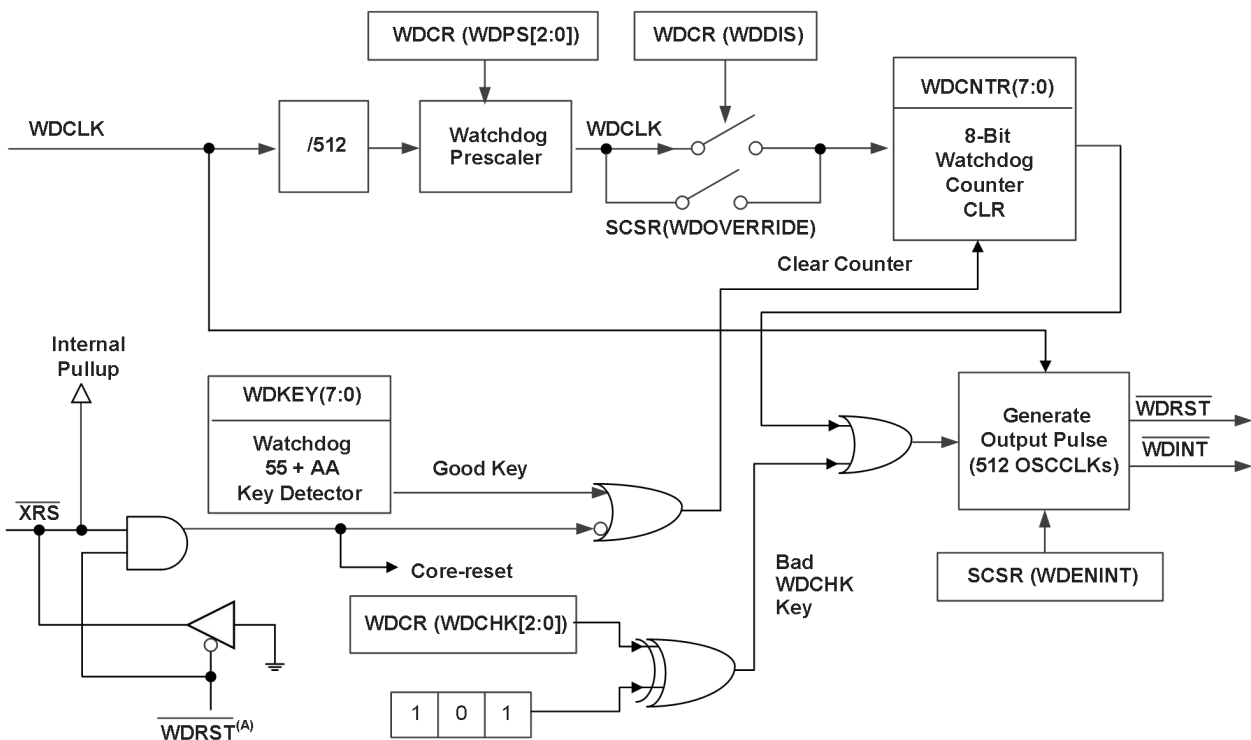
但是，当外部输入时钟失效时，CPU 看门狗将停止递减计数(即：PLL 跛行时钟无法给看门狗计时)。

**注意**

CPU 看门狗区别于 NMI 看门狗，CPU 看门狗是 ADP32、AVP32 系列产品中都存在的传统看门狗。

**注意**

如果应用程序中，正确的 CPU 工作频率起到绝对关键的作用。那么应用程序必须实现一种机制，在输入时钟出现故障时，将 MCU 保持在复位状态。例如：一个 R-C 电路组成的外部看门狗功能，在电容充满电时，给 XRS 引脚复位信号触发 MCU 复位重启。然后利用 MCU 的 I/O 引脚输出周期性脉冲，不间断的给电容放电，以防止电容充满后给出的 MCU 复位信号。这种电路有助于检测 FLASH 的故障和时钟缺失，在程序无法正常执行时，I/O 停止给外部电容器放电，直至 MCU 复位。



A. WDRST信号被驱动为低电平 512 个 OSCCLK 周期。

图 5-7 CPU 看门狗模块

$\overline{\text{WDINT}}$ 信号使看门狗可用来从 IDLE 或 STANDBY 模式唤醒 CPU。STANDBY 模式下，器件所有外设关闭，仅留下看门狗外设功能。这个模块在 OSCCLK 下运行， $\overline{\text{WDINT}}$ 信号被送到 LPM 模块，以便它可以将 CPU 从 STANDBY 模式唤醒。IDLE 模式下， $\overline{\text{WDINT}}$ 信号可以通过 PIE 向 CPU 产生中断，使其退出 IDLE 模式。HALT 模式下，CPU 看门狗只能通过设备重置来唤醒 CPU。

## 5.4 低功耗模式

下表汇总了不同的模式。

表 5-8 低功耗模式

模式	LPMCR0(1:0)	OSCCLK	CLKIN	SYSCCLKOUT	退出信号及条件 <sup>(1)</sup>
IDLE	00	打开	打开	打开	$\overline{\text{XRS}}$ , CPU看门狗中断, 任何启用的中断
STANDBY	01	打开(CPU看门狗仍在运行)	关闭	关闭	$\overline{\text{XRS}}$ , CPU看门狗中断, GPIO端口A信号, 调试器 <sup>(2)</sup>
HALT <sup>(3)</sup>	1X	关闭(片上晶体振荡器和PLL关闭, 内部振荡器和CPU看门狗的状态取决于用户代码)	关闭	关闭	$\overline{\text{XRS}}$ , GPIO端口A信号, 调试器 <sup>(2)</sup> , CPU看门狗

- (1) 任何信号上的低电平都退出低功耗状态，信号必须保持低电平足够长时间，以便器件识别中断。否则，低功耗模式将不会退出，且设备将返回到指定的低功耗模式。
- (2) 即使CPU时钟(CLKIN)被关闭，JTAG端口仍然可以工作。
- (3) WDCLK必须有效，器件才能进入HALT模式。

## 5.5 中断

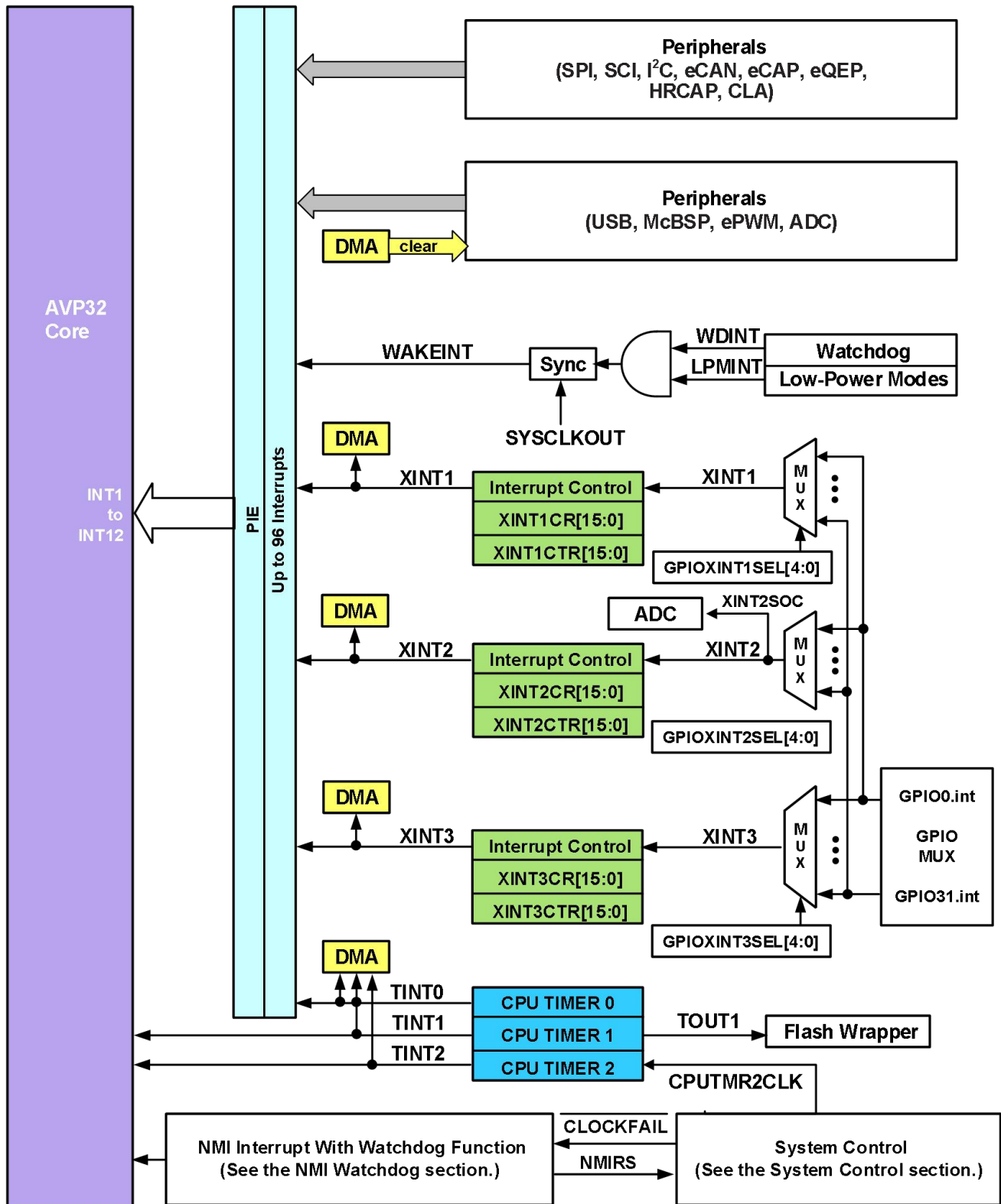


图 5-8 外部 PIE 中断源

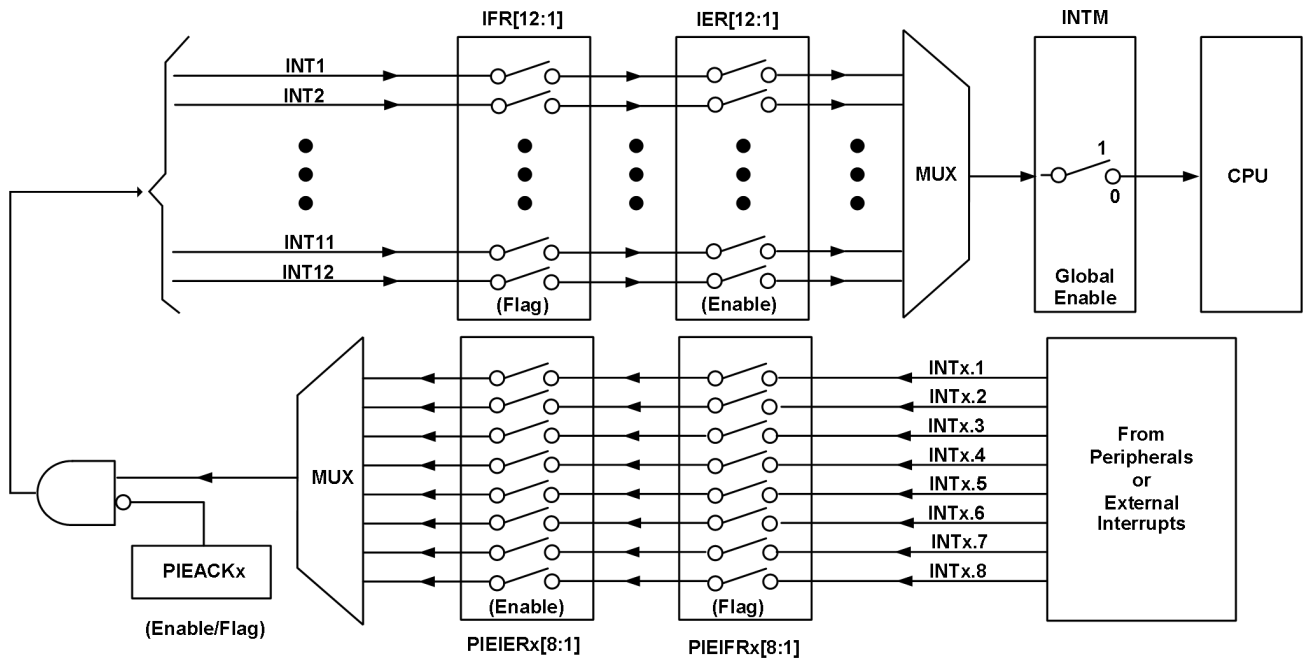


图 5-9 使用 PIE 块的中断复用

**表 5-9 PIE 混合式外设中断向量表(1)**

	INTx.8	INTx.7	INTx.6	INTx.5	INTx.4	INTx.3	INTx.2	INTx.1
INT1.y	WAKEINT (LPM/WD) 0xD4E	TINT0 (定时器0) 0xD4C	ADCINT9 (ADC) 0xD4A	XINT2 外部中断2 0xD48	XINT1 外部中断1 0xD46	Reserved — 0xD44	ADCINT2 (ADC) 0xD42	ADCINT1 (ADC) 0xD40
INT2.y	EPWM8_TZ INT (ePWM8) 0xD5E	EPWM7_TZ INT (ePWM7) 0xD5C	EPWM6_TZ INT (ePWM6) 0xD5A	EPWM5_TZ INT (ePWM5) 0xD58	EPWM4_TZ INT (ePWM4) 0xD56	EPWM3_TZ INT (ePWM3) 0xD54	EPWM2_TZ INT (ePWM2) 0xD52	EPWM1_TZ INT (ePWM1) 0xD50
INT3.y	EPWM8_IN T (ePWM8) 0xD6E	EPWM7_IN T (ePWM7) 0xD6C	EPWM6_IN T (ePWM6) 0xD6A	EPWM5_IN T (ePWM5) 0xD68	EPWM4_IN T (ePWM4) 0xD66	EPWM3_IN T (ePWM3) 0xD64	EPWM2_IN T (ePWM2) 0xD62	EPWM1_IN T (ePWM1) 0xD60
INT4.y	HRCAP2_I NT (HRCAP2) 0xD7E	HRCAP1_I NT (HRCAP1) 0xD7C	Reserved — 0xD7A	Reserved — 0xD78	Reserved — 0xD76	ECAP3_INT (eCAP3) 0xD74	ECAP2_INT (eCAP2) 0xD72	ECAP1_INT (eCAP1) 0xD70
INT5.y	USB0_INT (HRCAP2) 0xD8E	Reserved — 0xD8C	Reserved — 0xD8A	HRCAP4_I NT (HRCAP4) 0xD88	HRCAP3_I NT (HRCAP3) 0xD86	Reserved — 0xD84	EQEP2_INT (eQEP2) 0xD82	EQEP1_INT (eQEP1) 0xD80
INT6.y	Reserved — 0xD9E	Reserved — 0xD9C	MXINTA (McBSP-A) 0xD9A	MRINTA (McBSP-A) 0xD98	SPITXINTB (SPI-B) 0xD96	SPIRXINTB (SPI-B) 0xD94	SPITXINTA (SPI-A) 0xD92	SPIRXINTA (SPI-A) 0xD90
INT7.y	Reserved — 0xDAE	Reserved — 0xDAC	DINTCH6 (DMA) 0xDAA	DINTCH5 (DMA) 0xDA8	DINTCH4 (DMA) 0xDA6	DINTCH3 (DMA) 0xDA4	DINTCH2 (DMA) 0xDA2	DINTCH1 (DMA) 0xDA0
INT8.y	Reserved — 0xDBE	Reserved — 0xDBC	Reserved — 0xDBA	Reserved — 0xDB8	Reserved — 0xDB6	Reserved — 0xDB4	I2CINT2A (I2C-A) 0xDB2	I2CINT1A (I2C-A) 0xDB0
INT9.y	Reserved — 0xDCE	Reserved — 0xDCC	ECAN1_INT A (CAN-A) 0xDCA	ECAN0_INT A (CAN-A) 0xDC8	SCITXINTB (SCI-B) 0xDC6	SCIRXINTB (SCI-B) 0xDC4	SCITXINTA (SCI-A) 0xDC2	SCIRXINTA (SCI-A) 0xDC0
INT10.y	ADCINT8 (ADC) 0xDDE	ADCINT7 (ADC) 0xDDC	ADCINT6 (ADC) 0xDDA	ADCINT5 (ADC) 0xDD8	ADCINT4 (ADC) 0xDD6	ADCINT3 (ADC) 0xDD4	ADCINT2 (ADC) 0xDD2	ADCINT1 (ADC) 0xDD0
INT11.y	CLA1_INT8 (CLA) 0xDEE	CLA1_INT7 (CLA) 0xDEC	CLA1_INT6 (CLA) 0xDEA	CLA1_INT5 (CLA) 0xDE8	CLA1_INT4 (CLA) 0xDE6	CLA1_INT3 (CLA) 0xDE4	CLA1_INT2 (CLA) 0xDE2	CLA1_INT1 (CLA) 0xDE0
INT12.y	LUF (CLA) 0xDFE	LVF (CLA) 0xDFC	Reserved — 0xDFA	Reserved — 0xDF8	Reserved — 0xDF6	Reserved — 0xDF4	Reserved — 0xDF2	XINT3 外部中断3 0xDF0



**表 5-10 PIE 配置和控制寄存器**

名称	地址	大小(×16)	寄存器描述 <sup>(1)</sup>
PIECTRL	0x0CE0	1	PIE, Control Register
PIEACK	0x0CE1	1	PIE, Acknowledge Register
PIEIER1	0x0CE2	1	PIE, INT1 Group Enable Register
PIEIFR1	0x0CE3	1	PIE, INT1 Group Flag Register
PIEIER2	0x0CE4	1	PIE, INT2 Group Enable Register
PIEIFR2	0x0CE5	1	PIE, INT2 Group Flag Register
PIEIER3	0x0CE6	1	PIE, INT3 Group Enable Register
PIEIFR3	0x0CE7	1	PIE, INT3 Group Flag Register
PIEIER4	0x0CE8	1	PIE, INT4 Group Enable Register
PIEIFR4	0x0CE9	1	PIE, INT4 Group Flag Register
PIEIER5	0x0CEA	1	PIE, INT5 Group Enable Register
PIEIFR5	0x0CEB	1	PIE, INT5 Group Flag Register
PIEIER6	0x0CEC	1	PIE, INT6 Group Enable Register
PIEIFR6	0x0CED	1	PIE, INT6 Group Flag Register
PIEIER7	0x0CEE	1	PIE, INT7 Group Enable Register
PIEIFR7	0x0CEF	1	PIE, INT7 Group Flag Register
PIEIER8	0x0CF0	1	PIE, INT8 Group Enable Register
PIEIFR8	0x0CF1	1	PIE, INT8 Group Flag Register
PIEIER9	0x0CF2	1	PIE, INT9 Group Enable Register
PIEIFR9	0x0CF3	1	PIE, INT9 Group Flag Register
PIEIER10	0x0CF4	1	PIE, INT10 Group Enable Register
PIEIFR10	0x0CF5	1	PIE, INT10 Group Flag Register
PIEIER11	0x0CF6	1	PIE, INT11 Group Enable Register
PIEIFR11	0x0CF7	1	PIE, INT11 Group Flag Register
PIEIER12	0x0CF8	1	PIE, INT12 Group Enable Register
PIEIFR12	0x0CF9	1	PIE, INT12 Group Flag Register
Reserved	0x0CFA–0x0CFF	6	Reserved

### 5.5.1 外部中断

**表 5-11 外部中断寄存器**

名称	地址	大小(×16)	寄存器描述
XINT1CR	0x00 7070	1	XINT1 配置寄存器
XINT2CR	0x00 7071	1	XINT2 配置寄存器
XINT3CR	0x00 7072	1	XINT3 配置寄存器
XINT1CTR	0x00 7078	1	XINT1 计数寄存器
XINT2CTR	0x00 7079	1	XINT2 计数寄存器
XINT3CTR	0x00 707A	1	XINT3 计数寄存器

每个外部中断都可以使用正、负或正负边启用或禁用或限定。

### 5.5.1.1 外部中断电气数据/时序

表 5-12 外部中断时序要求

参数		最小值	最大值	单位
$t_w(\text{INT})^{(2)}$	INT脉冲持续时间			
	同步	$1t_c(\text{SCO})$		周期
	带输入限定器	$1t_c(\text{SCO}) + t_w(\text{IQSW})$		周期

表 5-13 外部中断开关特性

在推荐的操作条件范围内(除非另有说明)

参数	最小值	最大值	单位
$t_d(\text{INT})$	$12t_c(\text{SCO}) + t_w(\text{IQSW})$		周期

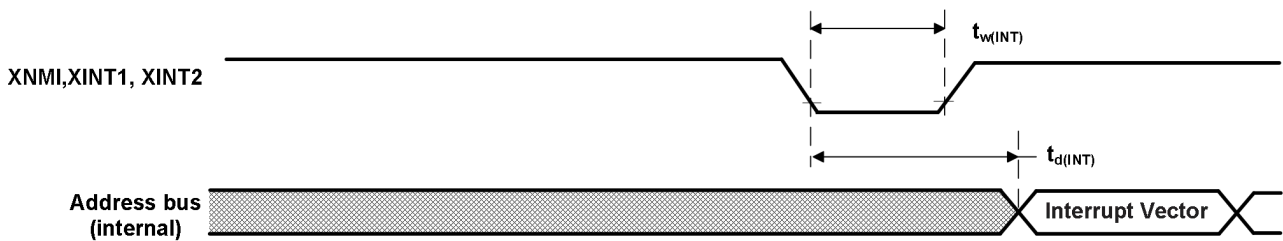


图 5-10 外部中断时序

## 5.6 外围设备

### 5.6.1 CLA 概述

CLA 通过添加并行处理，扩展了 CPU 的功能。由 CLA 服务的时间敏感控制回路可以实现低 ADC 采样到输出延迟。因此，CLA 能够实现更快的系统响应和更高频率控制环。使用 CLA 来执行时间关键的任务可以使主 CPU 能够一致地执行其他系统和通信功能。以下是 CLA 的主要特征列表。

- 与主CPU相同的速率（系统输出）
- 一个独立的架构，允许独立执行CLA算法的主CPU
  - 完整的总线架构：
    - 程序地址总线和程序数据总线
    - 数据地址总线、数据读取总线和数据写入总线
  - 独立的八级管道
  - 12位程序计数器(MPC)
  - 4个32位结果寄存器（MR0到MR3）
  - 两个16位辅助寄存器（MAR0、MAR1）
  - 状态寄存器(MSTF)
- 指令集包括：
  - IEEE单精度（32位）浮点数学运算
  - 浮点数学与并行负载或存储
  - 浮点乘
  - $1/X$ 和 $1/\sqrt{X}$ 估计
  - 数据类型转换
  - 条件分支和调用

- 数据加载和存储操作
- CLA程序代码最多可以包含8个任务或ISRs。
  - 每个任务的起始地址都由MVECT寄存器指定。
  - 只要任务符合CLA程序内存空间，任务大小就没有限制。
  - 一次只完成一个任务才能完成。任务未嵌套。
  - 在任务完成后，在PIE中标记一个特定于任务的中断。
  - 当任务完成时，下一个最高优先级的未决任务将自动启动。
- 任务触发机制：
  - AVP32 CPU通过IACK指令
  - 任务1到任务7：对应的ADC、ePWM、eQEP或eCAP模块中断。例如：
    - 任务1：ADCINT1或EPWM1\_INT
    - 任务2：ADCINT2或EPWM2\_INT
    - 任务4：ADCINT4或EPWM4\_INT或EQEPx\_INT或ECAPx\_INT
    - 任务7：ADCINT7或EPWM7\_INT或EQEPx\_INT或ECAPx\_INT
  - 任务8：ADCINT8或由CPU计时器0或EQEPx\_INT或ECAPx\_INT。
- 内存和共享外围设备：
  - 两个专用的消息ram，用于CLA和主CPU之间的通信
  - AVP32 CPU可以将CLA程序和数据存储器映射到主CPU空间或CLA空间。
  - CLA可以直接访问ADC结果寄存器、比较器寄存器，以及eCAP、eQEP和ePWM+HRPWM寄存器。

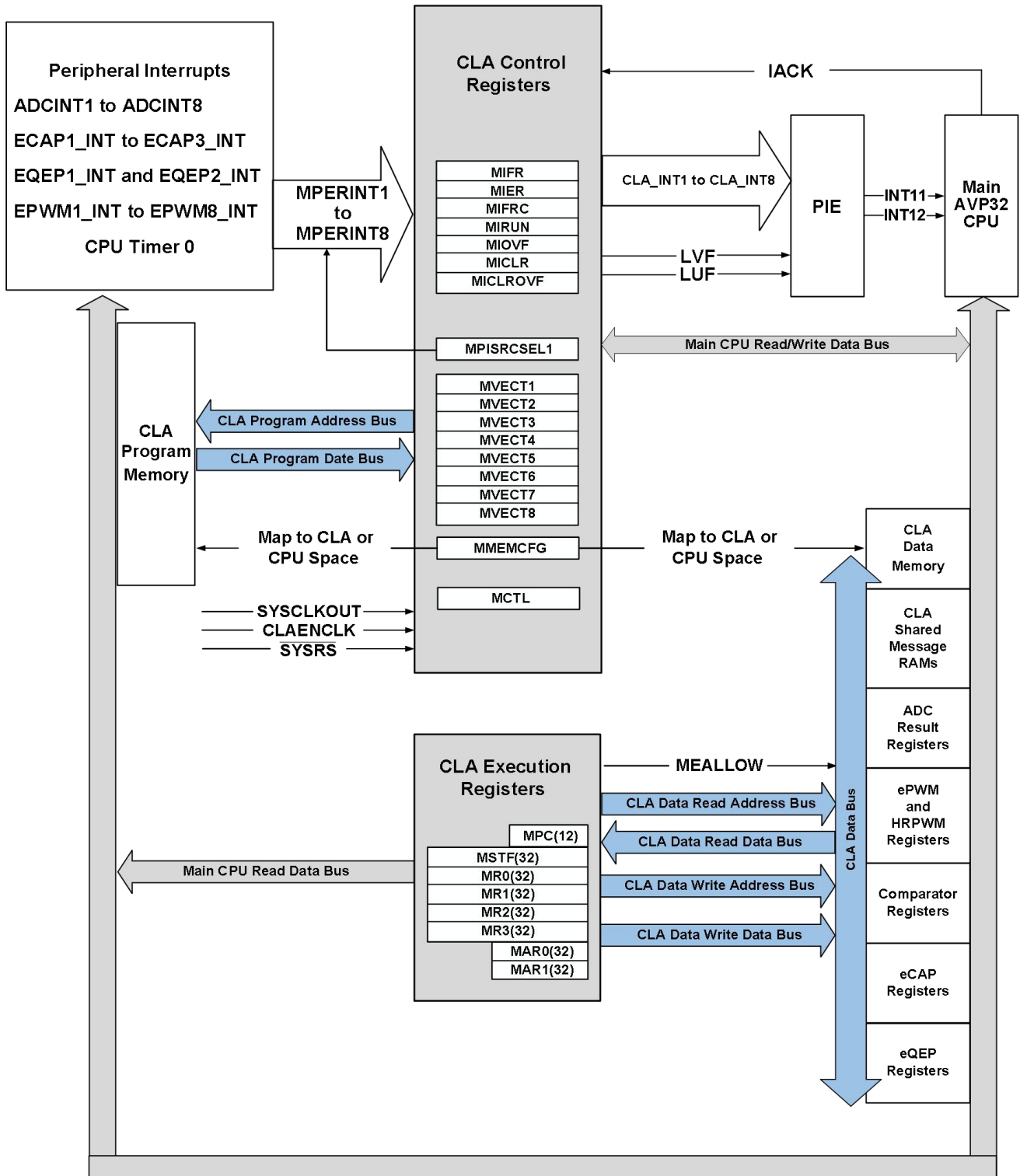


图 5-11 CLA 框图

**表 5-14 CLA 控制寄存器**

寄存器	CLA1地址	大小(×16)	允许保护	寄存器描述
MVECT1	0x1400	1	Yes	CLA中断/任务1开始处理
MVECT2	0x1401	1	Yes	CLA Interrupt/Task 2 Start Address
MVECT3	0x1402	1	Yes	CLA Interrupt/Task 3 Start Address
MVECT4	0x1403	1	Yes	CLA Interrupt/Task 4 Start Address
MVECT5	0x1404	1	Yes	CLA Interrupt/Task 5 Start Address
MVECT6	0x1405	1	Yes	CLA Interrupt/Task 6 Start Address
MVECT7	0x1406	1	Yes	CLA Interrupt/Task 7 Start Address
MVECT8	0x1407	1	Yes	CLA Interrupt/Task 8 Start Address
MCTL	0x1410	1	Yes	CLA控制寄存器
MMEMCFG	0x1411	1	Yes	CLA内存配置寄存器
MPISRCSEL1	0x1414	2	Yes	Peripheral Interrupt Source Select Register 1
MIFR	0x1420	1	Yes	Interrupt Flag Register
MIOVF	0x1421	1	Yes	Interrupt Overflow Register
MIFRC	0x1422	1	Yes	Interrupt Force Register
MICLR	0x1423	1	Yes	Interrupt Clear Register
MICLROVF	0x1424	1	Yes	Interrupt Overflow Clear Register
MIER	0x1425	1	Yes	Interrupt Enable Register
MIRUN	0x1426	1	Yes	Interrupt RUN Register
MIPCTL	0x1427	1	Yes	Interrupt Priority Control Register
MPC <sup>(2)</sup>	0x1428	1	-	CLA Program Counter
MAR0 <sup>(2)</sup>	0x142A	1	-	CLA Aux Register 0
MAR1 <sup>(2)</sup>	0x142B	1	-	CLA Aux Register 1
MSTF <sup>(2)</sup>	0x142E	2	-	CLA STF Register
MRO <sup>(2)</sup>	0x1430	2	-	CLA R0H Register
MR1 <sup>(2)</sup>	0x1434	2	-	CLA R1H Register
MR2 <sup>(2)</sup>	0x1438	2	-	CLA R2H Register
MR3 <sup>(2)</sup>	0x143C	2	-	CLA R3H Register

(1) 此表格所有寄存器受 CSM 保护。

(2) 主 CPU 调试目的下仅有只读访问权限。主 CPU 不能执行 CPU 或调试器写这些寄存器操作。

**表 5-15 CLA Message RAM**

地址范围	大小(×16)	描述
0x1480 - 0x14FF	128	CLA to CPU Message RAM
0x1500 - 0x157F	128	CPU to CLA Message RAM

## 5.6.2 模拟模块

实现了一个 12 位 ADC 核心，其时序与 ADP32F12 和 AVP32F335 设备上使用的 12 位 ADC 不同。ADC 包装器被修改以合并新的定时和其他增强，以改进转换开始的定时控制。图 5-12 显示了模拟模块与 AVP32F069 系统其余部分的交互作用。

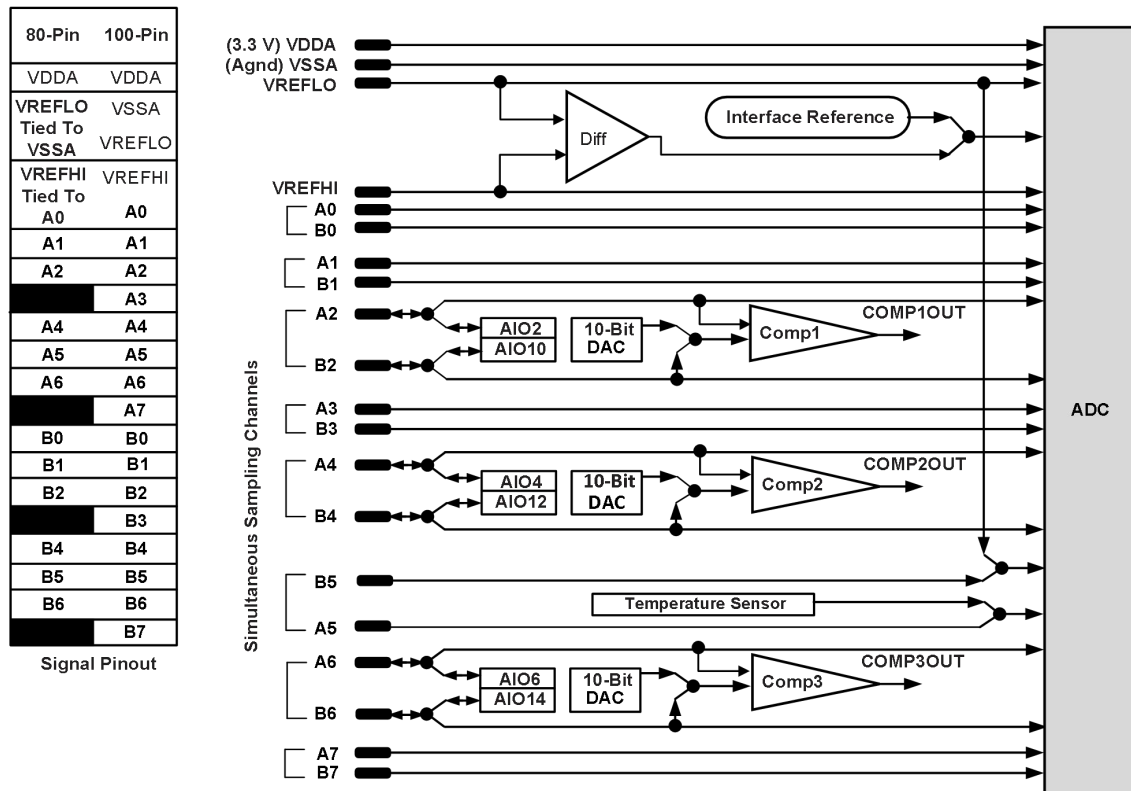


图 5-12 模拟引脚配置

### 5.6.2.1 模数转换器(ADC)

#### 5.6.2.1.1 特征

ADC 的核心包含由两个采样保持电路馈电的单个 12 位转换器。采样和保持电路可以同时或顺序采样。这些又由总共 16 个模拟输入通道提供。转换器可配置为使用内部带隙基准运行，以创建基于真实电压的转换，或使用一对外部电压基准 (VREFHI/VREFLO) 运行，以生成基于比率的转换。

与以前的 ADC 类型相反，这种 ADC 不是基于序列器的。用户可以轻松地从一个触发器创建一系列转换。然而，操作的基本原则是围绕单个转换的配置，称为 SOC 或转换开始。

ADC 模块功能包括:

- 内置双采样保持(S/H)电路的 12 位 ADC 核心。
- 同步采样和顺序采样模式。
- 全量程模拟输入: 0V 至 3.3V 固定, 或  $V_{REFHI}/V_{REFLO}$  比率测量。输入模拟电压和数字值由下式导出:
  - 内部基准 ( $V_{REFLO} = V_{SSA}$ , 使用内部或外部参考模式时,  $V_{REFHI}$  不得超过  $V_{DDA}$ 。 )
 

数值 = 0,	当输入 $\leq 0V$
数值 = $4096 * \frac{\text{模拟输入电压} - V_{REFLO}}{3.3}$	当 $0V < \text{输入} < 3.3V$
数值 = 4095,	当输入 $\geq 3.3V$
  - 外部基准 ( $V_{REFHI}/V_{REFLO}$  连接到外部基准。使用内部或外部参考模式时,  $V_{REFHI}$  不得超过  $V_{DDA}$ 。 )
 

数值 = 0,	当输入 $\leq 0V$
数值 = $4096 * \frac{\text{模拟输入电压} - V_{REFLO}}{V_{REFHI} - V_{REFLO}}$	当 $0V < \text{输入} < V_{REFHI}$
数值 = 4095,	当输入 $\geq V_{REFHI}$
- 最多 16 通道多路输入。
- 16 个 SOC, 可配置触发、采样窗口和通道选择。
- 16 个结果寄存器 (可单独寻址) 用于存储转换值。
- 多个触发源
  - S/W (软件立即启动)
  - ePWM1-8
  - GPIO XINT2
  - CPU 定时器 0\1\2
  - ADCINT1,ADCINT2
- 9 个灵活的 PIE 中断, 可配置任何转换后的中断请求。



**表 5-16 ADC 配置和控制寄存器**

寄存器名字ME	地址	大小(×16)	EALLOW保护	寄存器描述
ADCCTL1	0x7100	1	Yes	Control 1 Register
ADCCTL2	0x7101	1	Yes	Control 2 Register
ADCINTFLG	0x7104	1	No	Interrupt Flag Register
ADCINTFLGCLR	0x7105	1	No	Interrupt Flag Clear Register
ADCINTOVF	0x7106	1	No	Interrupt Overflow Register
ADCINTOVFCLR	0x7107	1	No	Interrupt Overflow Clear Register
INTSEL1N2	0x7108	1	Yes	Interrupt 1 and 2 Selection Register
INTSEL3N4	0x7109	1	Yes	Interrupt 3 and 4 Selection Register
INTSEL5N6	0x710A	1	Yes	Interrupt 5 and 6 Selection Register
INTSEL7N8	0x710B	1	Yes	Interrupt 7 and 8 Selection Register
INTSEL9N10	0x710C	1	Yes	Interrupt 9 Selection Register (reserved Interrupt 10 Selection)
SOCPRCTL	0x7110	1	Yes	SOC Priority Control Register
ADCSAMPLEMODE	0x7112	1	Yes	Sampling Mode Register
ADCINTSOCSEL1	0x7114	1	Yes	Interrupt SOC Selection 1 Register (for 8 channels)
ADCINTSOCSEL2	0x7115	1	Yes	Interrupt SOC Selection 2 Register (for 8 channels)
ADCSOCFLG1	0x7118	1	No	SOC Flag 1 Register (for 16 channels)
ADCSOCFRC1	0x711A	1	No	SOC Force 1 Register (for 16 channels)
ADCSOCOVF1	0x711C	1	No	SOC Overflow 1 Register (for 16 channels)
ADCSOCOVFCLR1	0x711E	1	No	SOC Overflow Clear 1 Register (for 16 channels)
ADCSOC0CTL to ADCSOC15CTL	0x7120~0x712F	1	Yes	SOC0 Control Register to SOC15 Control Register
ADCREFTRIM	0x7140	1	Yes	Reference Trim Register
ADCOFFTRIM	0x7141	1	Yes	Offset Trim Register
COMPHYSTCTL	0x714C	1	Yes	Comparator Hysteresis Control Register
ADCREV	0x714F	1	No	Revision Register

**表 5-17 ADC 结果寄存器 (已映射到 PF0)**

名称	地址	大小(×16)	EALLOW保护	寄存器描述
ADCRESULT0 to ADCRESULT15	0xB00 to 0xB0F	1	No	ADC Result 0 Register to ADC Result 15 Register

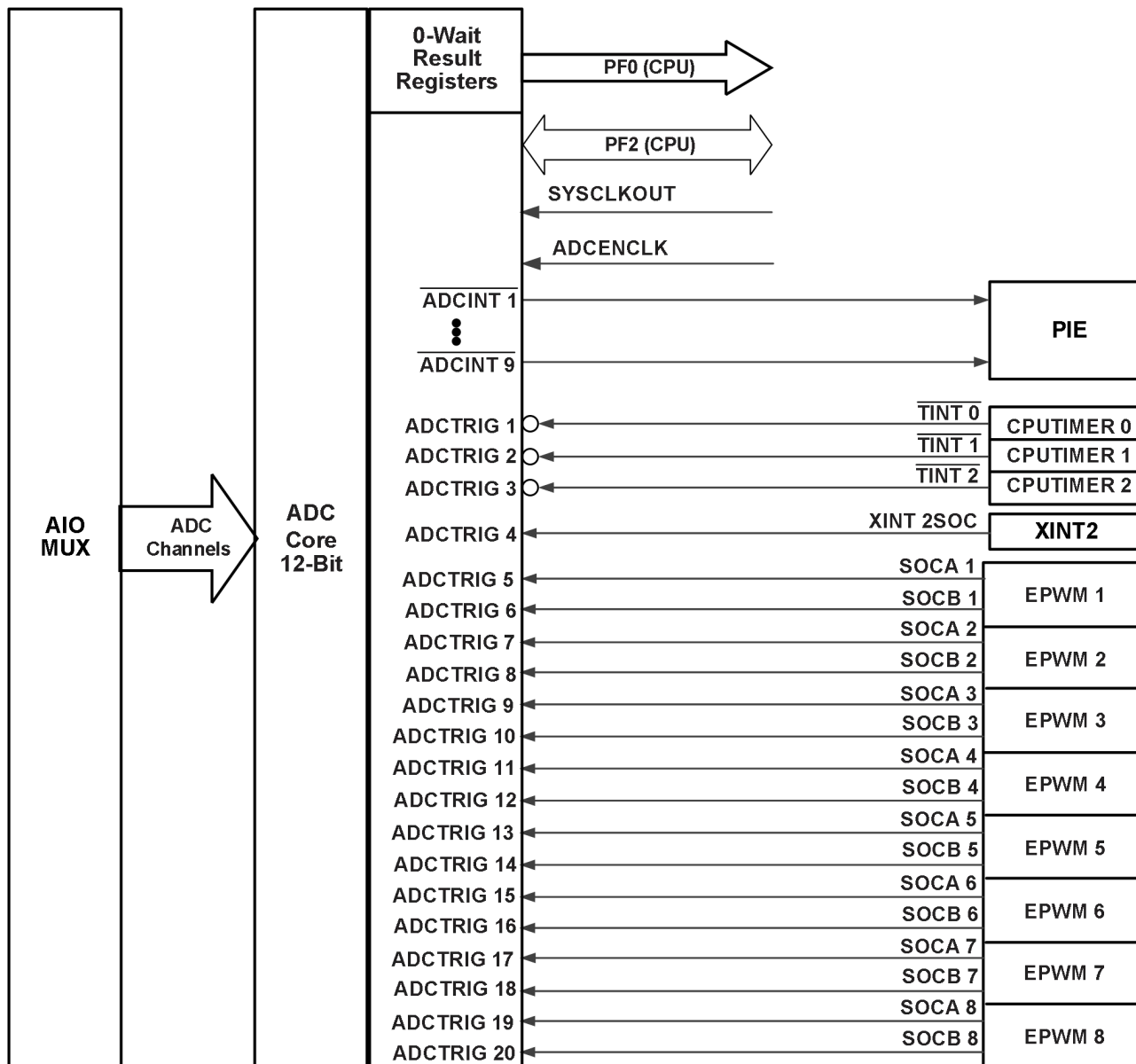


图 5-13 ADC 连接

### 如果 ADC 不使用时的 ADC 连接

原厂推荐保持模拟电源引脚的连接，尽管 ADC 不使用。如果 ADC 在应用中不使用，以下是 ADC 引脚应该如何连接的汇总：

- $V_{DDA}$  连接到  $V_{DDIO}$
- $V_{SSA}$  连接到  $V_{SS}$
- $V_{REFLO}$  连接到  $V_{SS}$

- $ADCINA_n, ADCINB_n, V_{REFHI}$  连接到  $V_{SSA}$

当在应用中使用 ADC 模块时，未使用的 ADC 输入引脚需连接到模拟地 ( $V_{SSA}$ )。

**注意**

建议通过 1kΩ 电阻器将与 AIO 功能多路复用的未使用 ADCIN 引脚接地。本建议旨在防止 AIO 输出逻辑高电平直接接地的任何意外软件激活；这种情况可能会因超过绝对最大值而导致永久性设备损坏。

当不使用 ADC 时，确保 ADC 模块的时钟未开启，以实现节能。

### 5.6.2.1.2 ADC开始转换电气数据/时间

#### 5.6.2.1.2.1 外部ADC转换启动开关特性

在推荐的操作条件范围(除非另有说明)

参数	最小值	最大值	单位
$t_w(ADCSOCL)$ ADCSOCxO低电平脉冲持续时间	$32t_c(HCO)$		周期

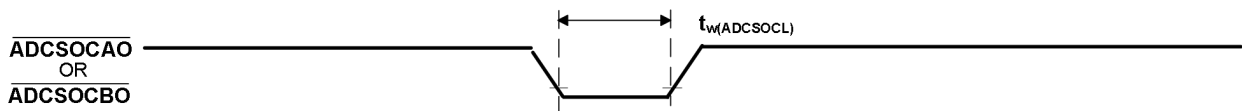


图 5-14 ADCSOCAO或ADCSOCBO时序

### 5.6.2.1.3 片上模数转换器(ADC)电气数据/时间

#### 5.6.2.1.3.1 ADC电气特性

参数	最小值	典型值	最大值	单位
直流特性				
分辨率	12			Bits
ADC时钟	0.001		45	MHz
采样窗口	7		64	ADC Clocks
精度				
INL (积分非线性) <sup>(1)</sup>	-4		4	LSB
DNL (微分非线性), 无失码	-1		1.5	LSB
偏移误差 <sup>(2)</sup>	执行一次自校准 <sup>(3)</sup>	-20	20	LSB
	执行定期自校准 <sup>(4)</sup>	-4	4	
外部参考的总偏移误差	-60		60	LSB
内部参考的总增益误差	-40		40	LSB
通道间偏移变化	-4		4	LSB
通道间增益变化	-4		4	LSB
带内部参考的ADC温度系数		-50		ppm/°C
带外部参考的ADC温度系数		-20		ppm/°C

参数		最小值	典型值	最大值	单位
$V_{REFLO}$			-100		$\mu A$
$V_{REFHI}$			100		$\mu A$
模拟输入					
带内部参考的模拟输入电压		0	3.3	3.3	V
带外部参考的模拟输入电压		$V_{REFLO}$		$V_{REFHI}$	V
$V_{REFLO}$ 输入电压 <sup>(5)</sup>		$V_{SSA}$		0.66	V
$V_{REFHI}$ 输入电压 <sup>(6)</sup>		2.64		$V_{DDA}$	V
	$V_{REFLO} = V_{SSA}$	1.98		$V_{DDA}$	
输入电容			5		pF
输入漏电流			$\pm 2$		$\mu A$

- (1) 当 ADC 输入电压高于  $V_{DDA}$  时, INL 将降级。
- (2) 1 LSB 的加权值为满标度范围 (FSR) / 4096。内部参考的 FSR 为 3.3V, 外部参考的是  $V_{REFHI} - V_{REFLO}$ 。
- (3) 定期自校准能够消除 ADC 零偏移误差对系统和温度的依赖。
- (4) 80 脚的封装中,  $V_{REFLO}$  一直连接在  $V_{SSA}$  上。
- (5) 不管使用内部还是外部参考模式,  $V_{REFHI}$  一定不能超过  $V_{DDA}$ 。因为 80 脚的封装中,  $V_{REFHI}$  被连接至 ADCINA0, 因此 ADCINA0 的输入信号一定不能超过  $V_{DDA}$ 。

### 5.6.2.1.3.2 ADC电源模式

ADC工作模式	条件	$I_{DDA}$	UNIT
模式A: 操作模式	ADC时钟已启用 带隙开启 (ADCBGPWD=1) 参考开启 (ADCREFPWD=1) ADC通电 (ADCPWDN=1)	22	mA
模式B: 快速唤醒模式	ADC时钟已启用 带隙开启 (ADCBGPWD=1) 参考开启 (ADCREFPWD=1) ADC掉电 (ADCPWDN=0)	4	mA
模式C: 仅比较器模式	ADC时钟已启用 带隙开启 (ADCBGPWD=1) 参考关闭 (ADCREFPWD=0) ADC掉电 (ADCPWDN=0)	1.5	mA
模式D: 关闭模式	ADC时钟已启用 带隙关闭 (ADCBGPWD=0) 参考关闭 (ADCREFPWD=0) ADC掉电 (ADCPWDN=0)	0.075	mA

### 5.6.2.1.3.3 内部温度传感器

#### 5.6.2.1.3.3.1 温度传感器系数

参数 <sup>(1)</sup>		最小值	典型值	最大值	单位
$T$ 斜率	测量温度传感器的ADC每一个LSB对应的摄氏温度量的变化		0.18 <sup>(2)</sup>		$^{\circ}C/LSB$
$T$ 偏移	温度传感器 $0^{\circ}C$ 时的ADC输出		1750		LSB

- (1) 使用 ADC 内部参考时, 以 ADC 的 LSBs 形式给出的温度传感器的斜率和偏移。在外部参考模式下, 必须根据外部参考电压调整相应值。
- (2) 温度传感器的输出(以 LSBs 形式)与温度变化量方向一致。相对于初始值, 升温将增加 ADC 值, 降温将减少 ADC 值。

### 5.6.2.1.3.4 ADC通电控制位时间

#### 5.6.2.1.3.4.1 ADC上电延迟

参数(1)	最小值	最大值	单位
$t_d(\text{PWD})$ 通电后ADC保持稳定的延迟时间		1	ms

(1) 保持与 ADC 模块时序兼容性，在第一次转换前的  $t_d(\text{PWD})$  时间内，ADC 支持同时驱动所有 3 位。

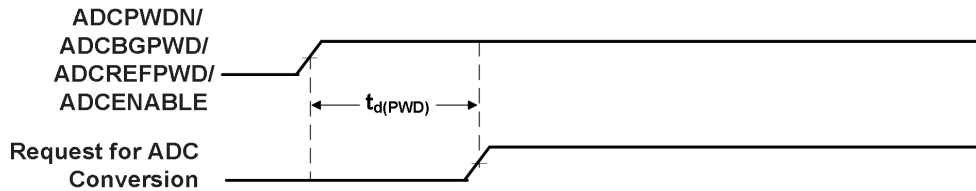
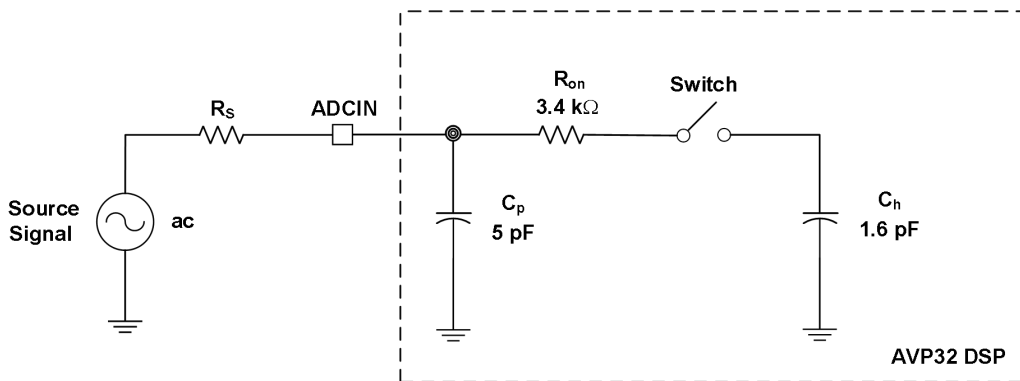


图 5-15 ADC 转换时序



输入电路部件的典型值：

开关电阻 ( $R_{on}$ ) : 3.4 k $\Omega$

采样电容 ( $C_H$ ) : 1.6 pF

寄生电容 ( $C_p$ ) : 5 pF

信号源电阻 ( $R_s$ ) : 50  $\Omega$

图 5-16 ADC 输入阻抗模型

5.6.2.1.3.5 ADC顺序和同步时序

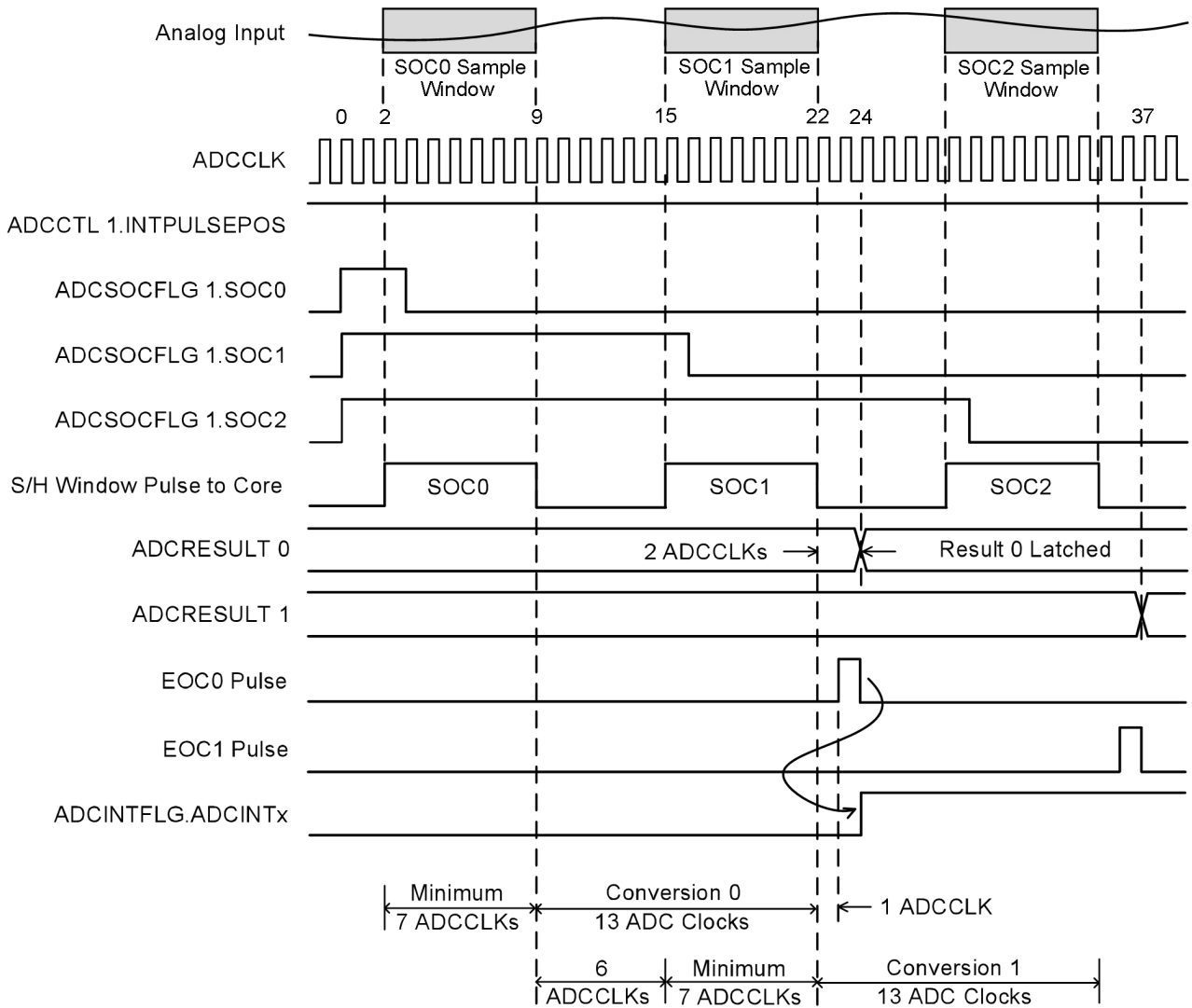


图 5-17 顺序模式/延迟中断脉冲的时序示例

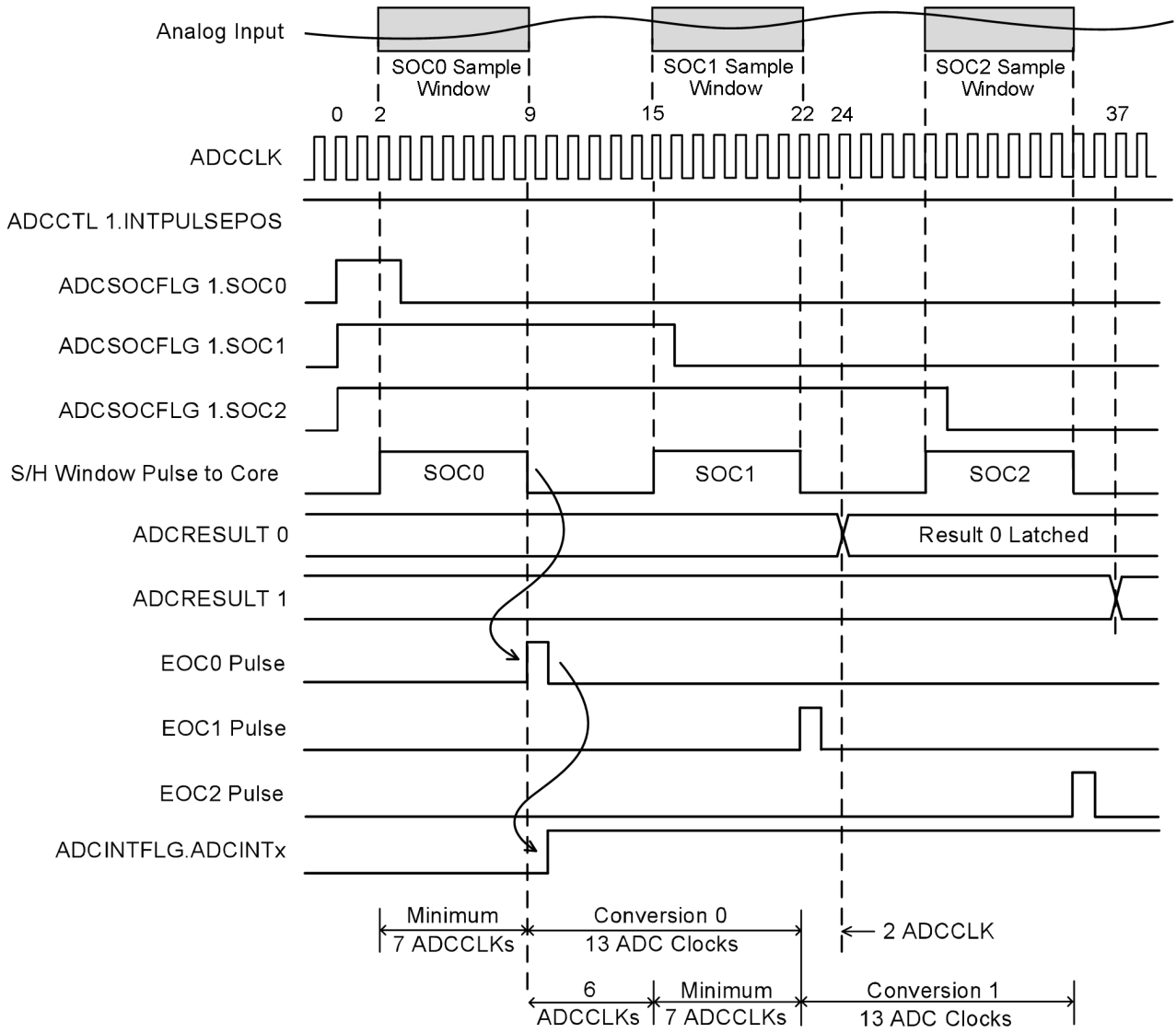


图 5-18 顺序模式/早期中断脉冲的时序示例

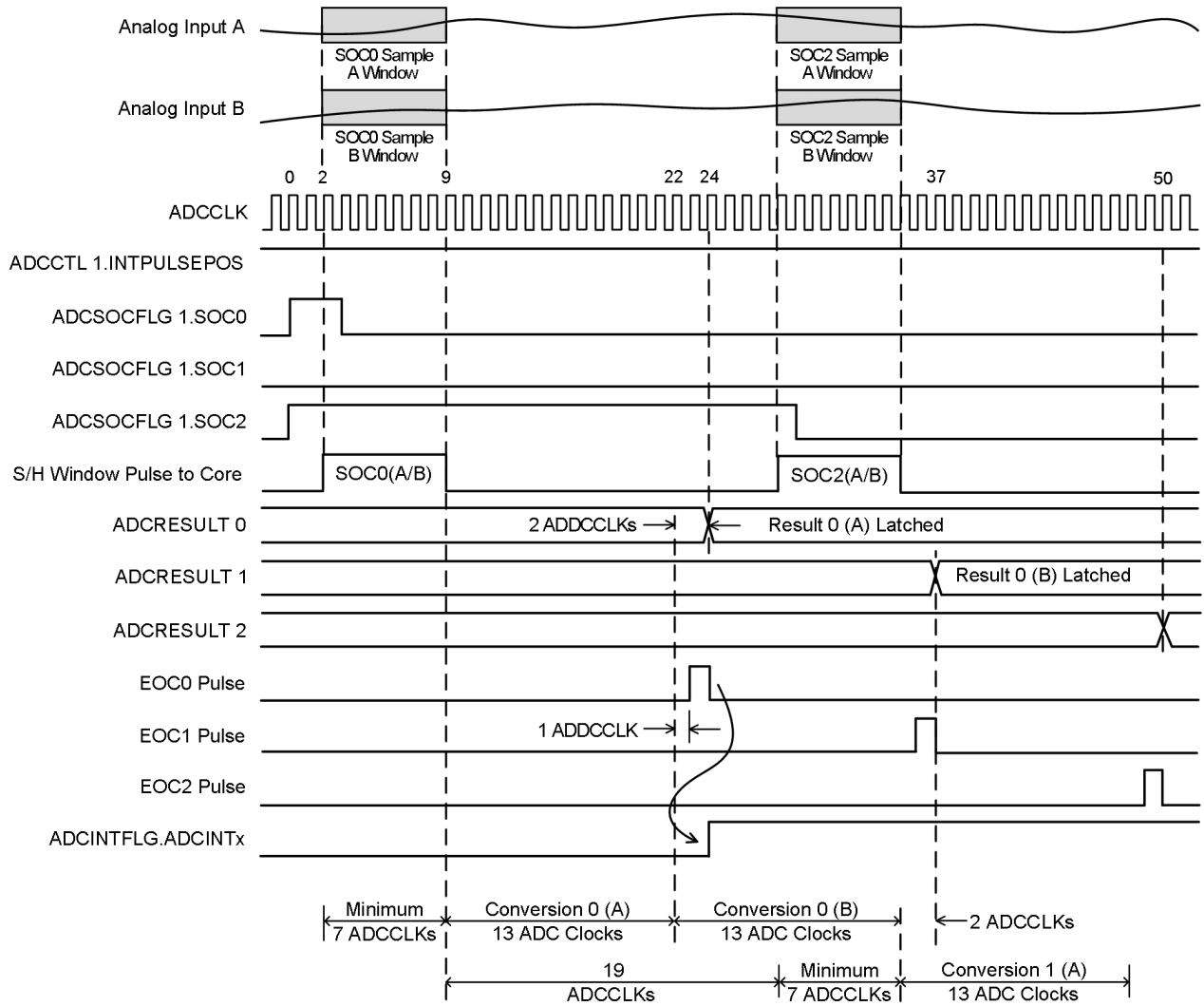


图 5-19 同步模式/延迟中断脉冲的时序示例



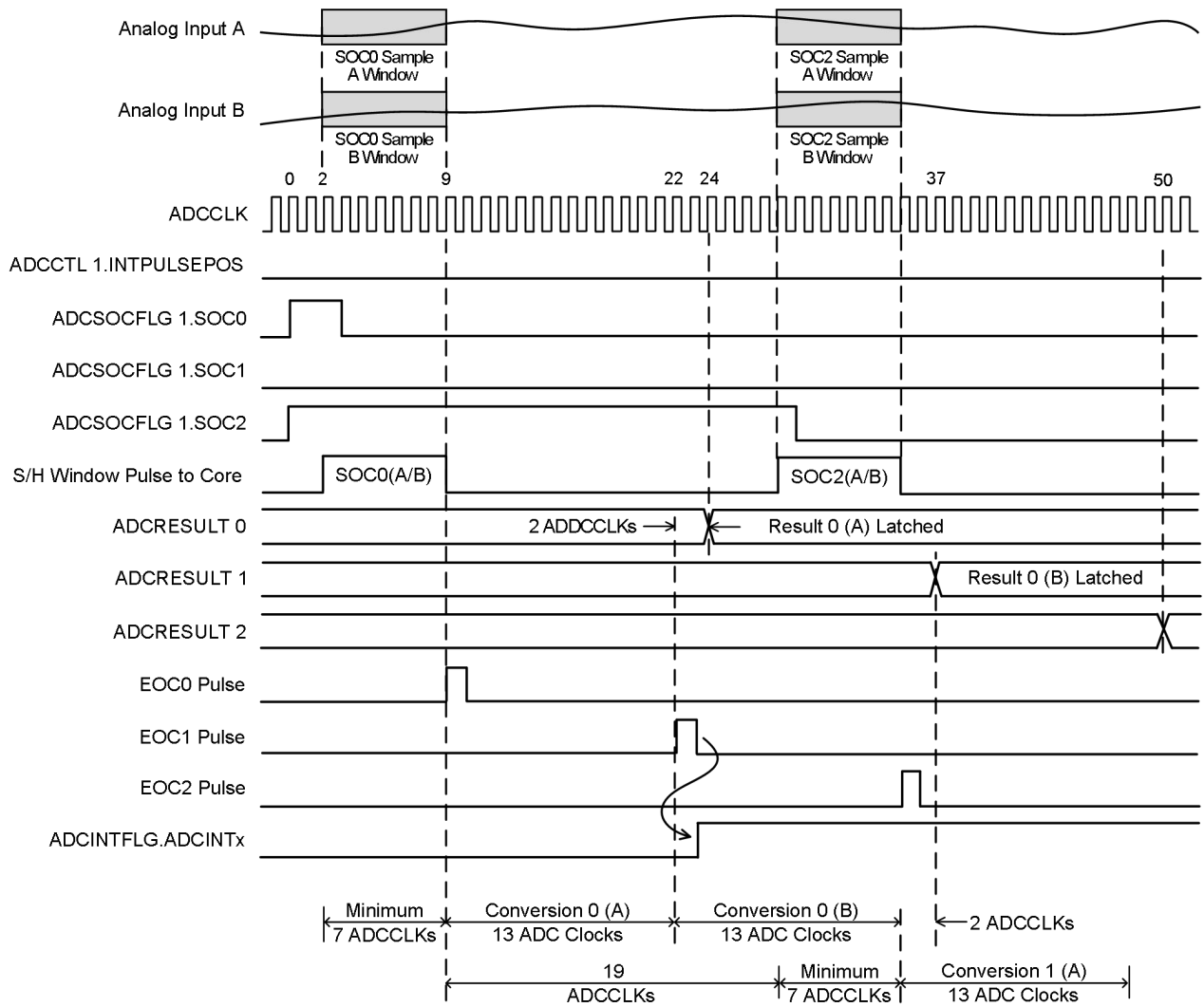


图 5-20 同步模式/早期中断脉冲的时序示例

### 5.6.2.2 ADC 复用

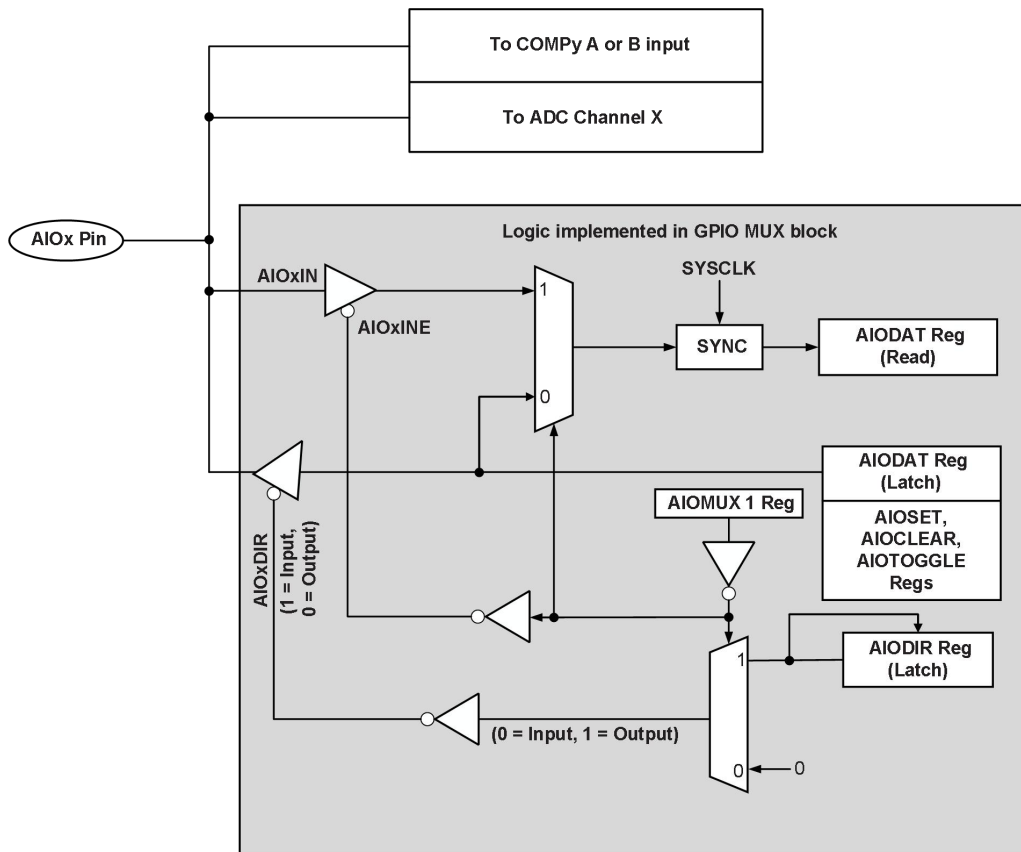


图 5-21 AIOx 引脚复用

ADC 通道和比较器功能始终可用。只有当 AIOMUX1 寄存器中的相应位为 0 时，数字 I/O 功能才可用。在这种模式下，读取 AIODAT 寄存器反映实际引脚状态。

当 AIOMUX1 寄存器中的相应位为 1 时，数字 I/O 功能被禁用。这种模式下，读取 AIODAT 寄存器反映了 AIODAT 的输出锁存器，输入数字 I/O 缓冲器被禁用，以防止模拟信号产生噪声。

复位时，数字功能被禁用，如果该引脚用作模拟输入，用户应保持禁用该引脚的 AIO 功能。

### 5.6.2.3 比较器块

图 5-22 显示了比较器模块与系统其他部分的交互作用。

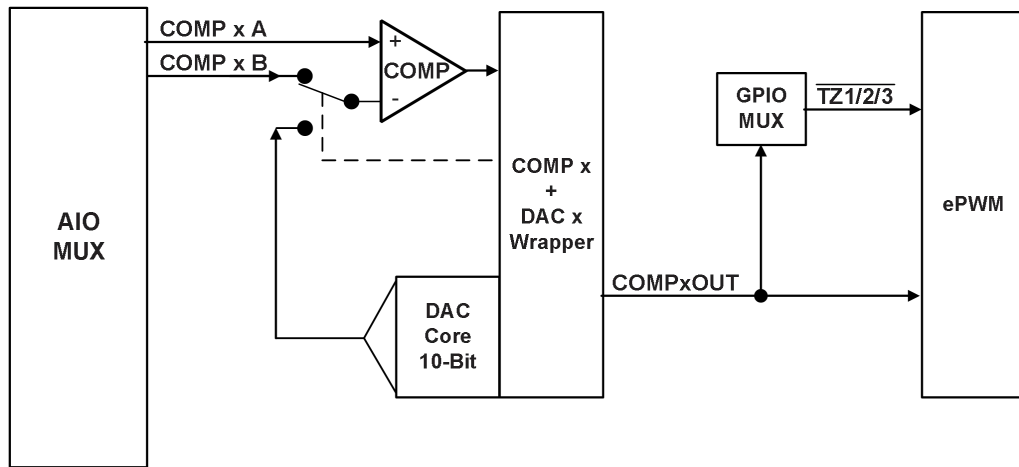


图 5-22 比较器方框图

表 5-18 比较器控制寄存器

名称	COMP1地址	COMP2地址	COMP3地址	大小(×16)	EALLOW保护	寄存器描述
COMPCTL	0x6400	0x6420	0x6440	1	Yes	比较器控制寄存器
COMPSTS	0x6402	0x6422	0x6442	1	No	比较器状态寄存器
DACCTL	0x6404	0x6424	0x6444	1	Yes	DAC 控制寄存器
DACVAL	0x6406	0x6426	0x6446	1	No	DAC 值寄存器
RAMPMAXREF_ACTIVE	0x6408	0x6428	0x6448	1	No	斜坡发生器最大参考(有效)寄存器
RAMPMAXREF_SHDW	0x640A	0x642A	0x644A	1	No	斜坡发生器最大参考(影子)寄存器
RAMPDECVAL_ACTIVE	0x640C	0x642C	0x644C	1	No	斜坡发生器减少值(有效)寄存器
RAMPDECVAL_SHDW	0x640E	0x642E	0x644E	1	No	斜坡发生器减少值(影子)寄存器
RAMPSTS	0x6410	0x6430	0x6450	1	No	斜坡发生器状态寄存器

**5.6.2.3.1 片上比较器/DAC电气数据/时序**
**5.6.2.3.1.1 比较器/DAC的电气特性**

特性	最小值	典型值	最大值	单位
比较器				
比较器输入范围	$V_{SSA}-V_{DDA}$			V
PWM 跳闸区到比较器响应时间（异步）	30			ns
输入偏移	$\pm 5$			mV
输入迟滞 <sup>(1)</sup>	35			mV
DAC				
DAC 输出范围	$V_{SSA}-V_{DDA}$			V
DAC 分辨率	10			bits
DAC 建立时间	待测			
DAC 增益	-1.5%			
DAC 偏移	10			mV
单调性	Yes			
INL	$\pm 3$			LSB

(1) 比较器输入端的滞后通过施密特触发器配置实现。这导致比较器输出和非反相输入之间的 100kΩ有效反馈电阻。

### 5.6.3 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，

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

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。

## 5.6.4 串行外围接口模块 (SPI)

该器件包括 4 个引脚串行外设接口 (SPI) 模块。最多可提供两个 SPI 模块。SPI 是一个高速、同步串行 I/O 端口，允许以可编程位传输速率将编程长度 (1 至 16 位) 的串行位流移入和移出器件。通常，SPI 用于 MCU 与外部外设或其他处理器之间的通信。典型应用包括外部 I/O 或通过移位寄存器、显示驱动器和 ADC 等器件进行外设扩展。SPI 的主/从操作支持多器件通信。

SPI 模块特性包括：

- 四个外部引脚：
  - SPISOMI: SPI 从输出/主输入引脚
  - SPISIMO: SPI 从输入/主输出引脚
  - $\overline{\text{SPISTE}}$ : SPI 从发送使能引脚
  - SPICLK: SPI 串行时钟引脚

---

**注意**

如果 SPI 不使用，所有四个引脚都可以用作 GPIO。

---

- 两种操作模式：主模式和从模式

波特率：125 种不同的可编程速率。

$$\text{波特率} = \frac{\text{LSPCLK}}{(\text{SPIBRR}+1)} \quad \text{当 SPIBRR} = 3 \text{ 到 } 127$$

$$\text{波特率} = \frac{\text{LSPCLK}}{4} \quad \text{当 SPIBRR} = 0, 1, 2$$

- 数据字长度：1 到 16 位数据
- 四种时钟方案（由时钟极性和时钟相位控制）包括：
  - 无相位延迟下降沿：SPICLK 高有效，SPI 在 SPICLK 信号下降沿发送数据，并在 SPICLK 信号上升沿接收数据。
  - 有相位延迟下降沿：SPICLK 高有效，SPI 在 SPICLK 信号下降沿前半个周期发送数据，并在

SPICLK 信号的下降沿接收数据。

- 无相位延迟上升沿：SPICLK 低有效，SPI 在 SPICLK 信号上升沿发送数据，并在 SPICLK 信号下降沿接收数据。
- 有相位延迟上升沿：SPICLK 低有效，SPI 在 SPICLK 信号上升沿前半个周期发送数据，并在 SPICLK 信号上升沿接收数据。
- 接收和发送同时（可在软件中禁用发送功能）。
- 可通过中断或轮询完成收发。
- 九个模块控制寄存器：从 0x7040 地址开始的寄存器框架中。

---

#### 注意

这个模块内所有寄存器都是 16 位寄存器，被连接到外设帧 2。当寄存器被访问时，寄存器数据是低字节(7-0)数据，读高字节是 0，写高字节无作用。

---

扩展特性包括：

- 4 级收发 FIFO
- 延迟发送控制
- 双向三线制 SPI 模式支持
- 通过  $\overline{\text{SPISTE}}$  反转实现音频数据接收支持

下列表格列出了 SPI 端口操作配置和控制寄存器。



**表 5-19 SPI-A 寄存器**

名称	地址	大小(×16)	EALLOW保护	寄存器描述(1)
SPICCR	0x7040	1	No	SPI-A Configuration Control Register
SPICTL	0x7041	1	No	SPI-A Operation Control Register
SPISTS	0x7042	1	No	SPI-A Status Register
SPIBRR	0x7044	1	No	SPI-A Baud Rate Register
SPIRXEMU	0x7046	1	No	SPI-A Receive Emulation Buffer Register
SPIRXBUF	0x7047	1	No	SPI-A Serial Input Buffer Register
SPITXBUF	0x7048	1	No	SPI-A Serial Output Buffer Register
SPIDAT	0x7049	1	No	SPI-A Serial Data Register
SPIFFTX	0x704A	1	No	SPI-A FIFO Transmit Register
SPIFFRX	0x704B	1	No	SPI-A FIFO Receive Register
SPIFFCT	0x704C	1	No	SPI-A FIFO Control Register
SPIPRI	0x704F	1	No	SPI-A Priority Control Register

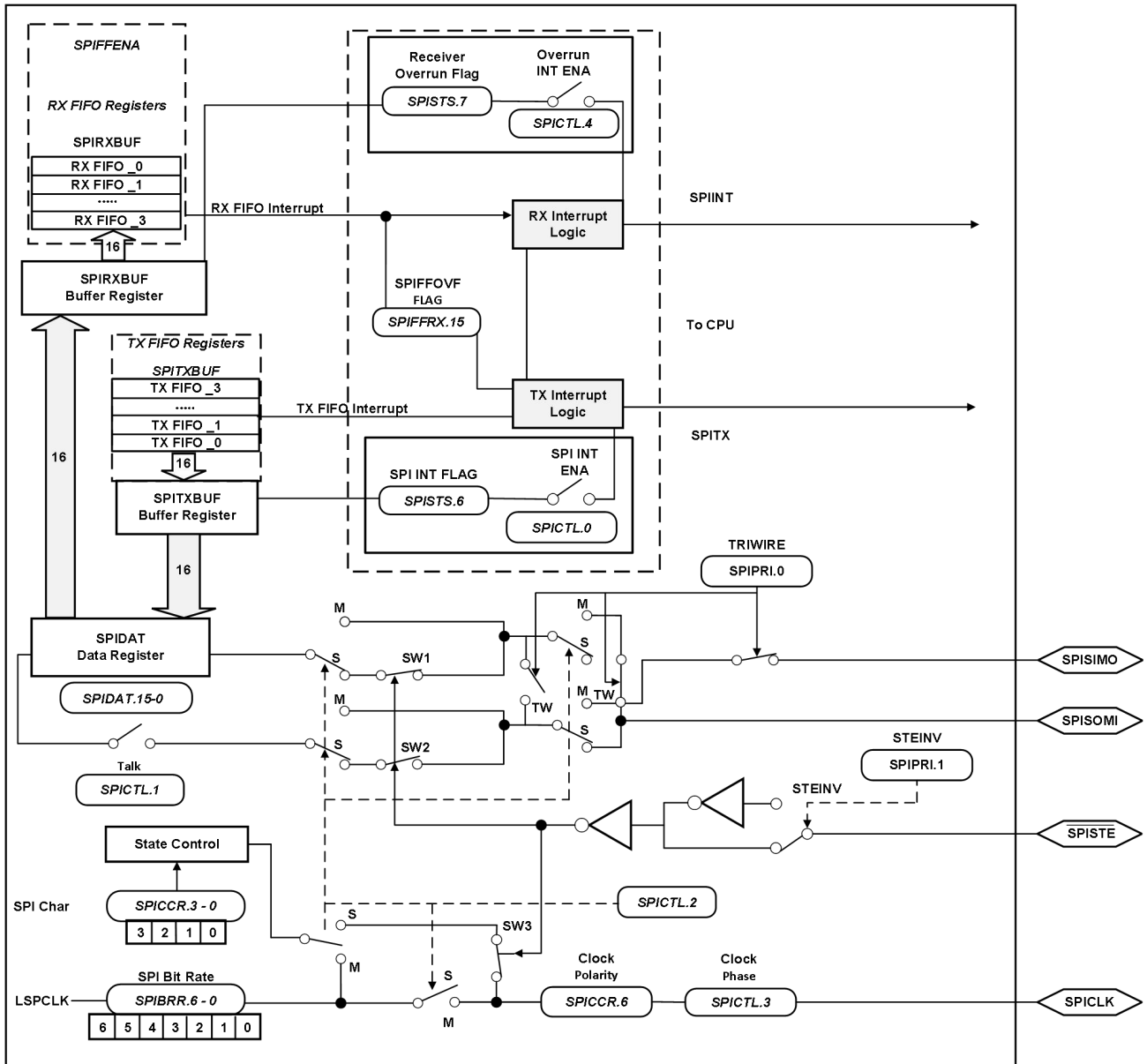
(1) 此表中的寄存器映射到外设帧 2，此空间仅允许 16 位访问，32 位的访问将产生未定义的结果。

**表 5-20 SPI-B 寄存器**

名称	地址	大小(×16)	EALLOW保护	寄存器描述(1)
SPICCR	0x7740	1	No	SPI-B Configuration Control Register
SPICTL	0x7741	1	No	SPI-B Operation Control Register
SPISTS	0x7742	1	No	SPI-B Status Register
SPIBRR	0x7744	1	No	SPI-B Baud Rate Register
SPIRXEMU	0x7746	1	No	SPI-B Receive Emulation Buffer Register
SPIRXBUF	0x7747	1	No	SPI-B Serial Input Buffer Register
SPITXBUF	0x7748	1	No	SPI-B Serial Output Buffer Register
SPIDAT	0x7749	1	No	SPI-B Serial Data Register
SPIFFTX	0x774A	1	No	SPI-B FIFO Transmit Register
SPIFFRX	0x774B	1	No	SPI-B FIFO Receive Register
SPIFFCT	0x774C	1	No	SPI-B FIFO Control Register
SPIPRI	0x774F	1	No	SPI-B Priority Control Register

(1) 此表中的寄存器映射到外设帧 2，此空间仅允许 16 位访问，32 位的访问将产生未定义的结果。

下图是 SPI 从模式下的框图。



A.  $\overline{\text{SPISTE}}$ 由主机驱动到低电平。

图 5-23 SPI 模块框图 (从模式)

5.6.4.1 SPI 主模式电气数据/时序

5.6.4.1.1 SPI 主模式外部时序(Clock Phase = 0)

No.	参数 (1) (2) (3) (4) (5)		波特率配置寄存器偶数		波特率配置寄存器奇数		单位
			最小值	最大值	最小值	最大值	
1	$t_c$ (SPC)M	SPICLK 时钟周期	$4t_c$ (LSPCLK)	$128t_c$ (LSPCLK)	$5t_c$ (LSPCLK)	$127t_c$ (LSPCLK)	ns
2	$t_w$ (SPC1)M	SPICLK 首脉宽	$0.5t_c$ (SPC)M <sup>-10</sup>	$0.5t_c$ (SPC)M <sup>+10</sup>	$0.5t_c$ (SPC)M + $0.5t_c$ (LSPCLK) <sup>-10</sup>	$0.5t_c$ (SPC)M + $0.5t_c$ (LSPCLK) <sup>+10</sup>	ns
3	$t_w$ (SPC2)M	SPICLK 次脉宽	$0.5t_c$ (SPC)M <sup>-10</sup>	$0.5t_c$ (SPC)M <sup>+10</sup>	$0.5t_c$ (SPC)M - $0.5t_c$ (LSPCLK) <sup>-10</sup>	$0.5t_c$ (SPC)M - $0.5t_c$ (LSPCLK) <sup>+10</sup>	ns
4	$t_d$ (SIMO)M	主机输出数据延迟		10		10	ns
5	$t_v$ (SIMO)M	输出数据有效时间	$0.5t_c$ (SPC)M <sup>-10</sup>		$0.5t_c$ (SPC)M - $0.5t_c$ (LSPCLK) <sup>-10</sup>		ns
8	$t_{su}$ (SOMI) M	输入数据建立时间	26		26		ns
9	$t_h$ (SOMI)M	输入数据保持时间	0		0		ns
23	$t_d$ (SPC)M	片选有效延时	$1.5t_c$ (SPC)M - $3t_c$ (SYSCLK) <sup>-10</sup>		$0.5t_c$ (SPC)M - $3t_c$ (SYSCLK) <sup>-10</sup>		ns
24	$t_d$ (STE)M	片选失效延时	$0.5t_c$ (SPC)M <sup>-10</sup>		$0.5t_c$ (SPC)M - $0.5t_c$ (LSPCLK) <sup>-10</sup>		ns

- (1) 主模式/从模式 (SPICTL. 2) 寄存器位置 1, 时钟相位 (SPICTL. 3) 被清零。
- (2)  $t_c$ (SPC) = SPI 时钟周期 = LSPCLK/4 或 LSPCLK/(SPIBRR+1)。
- (3)  $t_c$ (LCO) = LSPCLK 时钟周期。
- (4) 必须调整内部时钟预分频器, 使 SPI 时钟限制在以下 SPI 时钟速率: 主模式最大 25MHz 发送, 最大 12.5MHz 接收; 从模式最大 12.5MHz 发送, 最大 12.5MHz 接收。
- (5) 参考的 SPICLK 信号的有效边沿由时钟极性 (SPICCR. 6) 控制。

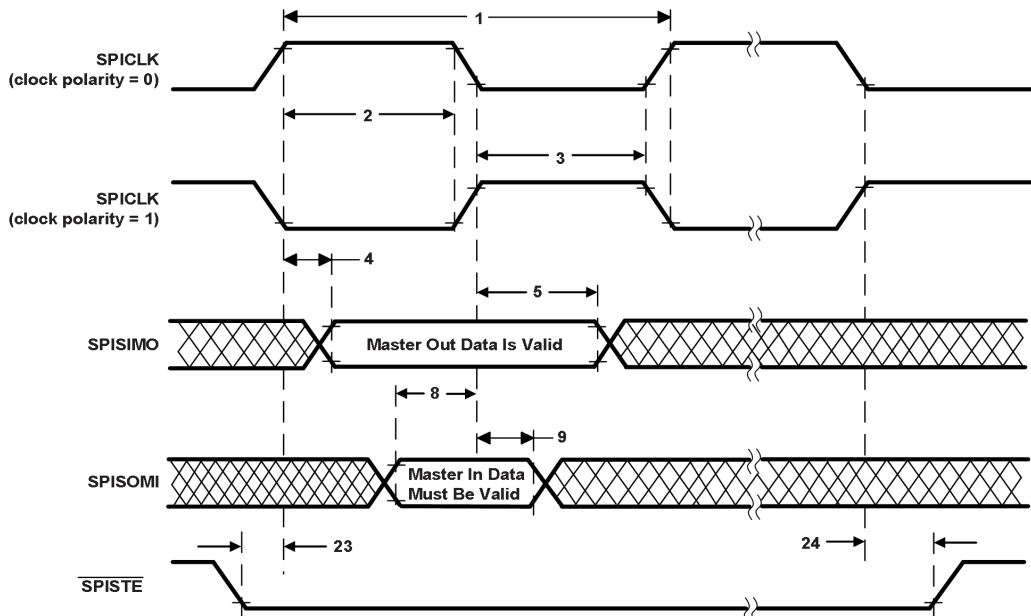


图 5-24 SPI 主模式外部时序(Clock Phase = 0)

**5.6.4.1.2 SPI 主模式外部时序(Clock Phase = 1)**

No.	参数(1)(2)(3)(4)(5)		波特率配置寄存器偶数		波特率配置寄存器奇数		单位
			最小值	最大值	最小值	最大值	
1	$t_c(\text{SPC})_M$	SPICLK 时钟周期	$4t_c(\text{LSPCLK})$	$128t_c(\text{LSPCLK})$	$5t_c(\text{LSPCLK})$	$127t_c(\text{LSPCLK})$	ns
2	$t_w(\text{SPC1})_M$	SPICLK 首脉宽	$0.5t_c(\text{SPC})_M - 10$	$0.5t_c(\text{SPC})_M + 10$	$0.5t_c(\text{SPC})_M - 0.5t_c(\text{LSPCLK}) - 10$	$0.5t_c(\text{SPC})_M - 0.5t_c(\text{LSPCLK}) + 10$	ns
3	$t_w(\text{SPC2})_M$	SPICLK 次脉宽	$0.5t_c(\text{SPC})_M - 10$	$0.5t_c(\text{SPC})_M + 10$	$0.5t_c(\text{SPC})_M + 0.5t_c(\text{LSPCLK}) - 10$	$0.5t_c(\text{SPC})_M + 0.5t_c(\text{LSPCLK}) + 10$	ns
6	$t_d(\text{SIMO})_M$	主机输出数据延迟	$0.5t_c(\text{SPC})_M - 10$		$0.5t_c(\text{SPC})_M + 0.5t_c(\text{LSPCLK}) - 10$		ns
7	$t_v(\text{SIMO})_M$	输出数据有效时间	$0.5t_c(\text{SPC})_M - 10$		$0.5t_c(\text{SPC})_M - 0.5t_c(\text{LSPCLK}) - 10$		ns
10	$t_{su}(\text{SOMI})_M$	输入数据建立时间	26		26		ns
11	$t_h(\text{SOMI})_M$	输入数据保持时间	0		0		ns
23	$t_d(\text{SPC})_M$	片选有效延时	$2t_c(\text{SPC})_M - 3t_c(\text{SYSCLK}) - 10$		$2t_c(\text{SPC})_M - 3t_c(\text{SYSCLK}) - 10$		ns
24	$t_d(\text{STE})_M$	片选失效延时	$0.5t_c(\text{SPC})_M - 10$		$0.5t_c(\text{SPC})_M - 0.5t_c(\text{LSPCLK}) - 10$		ns

- (1) 主模式/从模式(SPICTL. 2)寄存器位置1,且时钟相位(SPICTL. 3)置1。
- (2)  $t_c(\text{SPC}) = \text{SPI 时钟周期} = \text{LSPCLK}/4$  或  $\text{LSPCLK}/(\text{SPIBRR}+1)$ 。
- (3) 必须调整内部时钟预分频器,使SPI时钟限制在以下SPI时钟速率:主模式最大25MHz发送,最大12.5MHz接收;从模式最大12.5MHz发送,最大12.5MHz接收。
- (4)  $t_c(\text{LCO}) = \text{LSPCLK 时钟周期}$ 。
- (5) 参考的SPICLK信号的有效边沿由时钟极性(SPICCR. 6)控制。

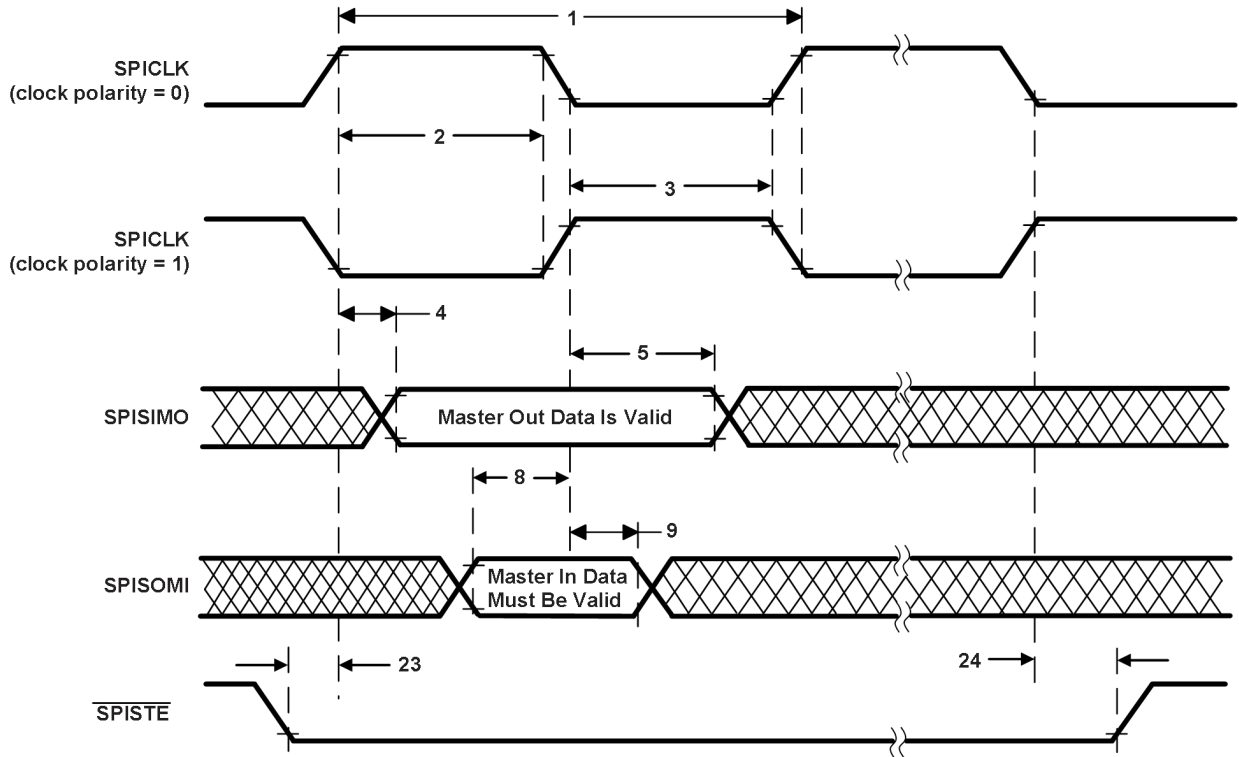


图 5-25 SPI 主模式外部时序(Clock Phase = 1)

### 5.6.4.2 SPI 从模式电气数据/时序

#### 5.6.4.2.1 SPI 从模式外部时序(Clock Phase = 0)

以下列出 SPI 从模式下，时钟相位不含半周期延时和含半周期延时的时序和波形。

No.	参数 (1) (2) (3) (4) (5)	最小值	最大值	单位
12	$t_c$ (SPC) S	SPICLK 周期		ns
13	$t_w$ (SPC1) S	SPICLK 首脉宽		ns
14	$t_w$ (SPC2) S	SPICLK 次脉宽		ns
15	$t_d$ (SOMI) S	SPICLK 到 SPISOMI 输出最大延迟时间		ns
16	$t_v$ (SOMI) S	SPICLK 到 SPISOMI 数据失效最短时间		ns
19	$t_{su}$ (SIMO) S	SPICLK 前，从机 SPISIMO 有效建立最短时间		ns
20	$t_h$ (SIMO) S	SPICLK 后，从机 SPISIMO 数据保持最短时间		ns
25	$t_{su}$ (STE) S	SPICLK 前， $\overline{\text{SPISITE}}$ 最小建立时间		ns
26	$t_h$ (STE) S	SPICLK 后， $\overline{\text{SPISITE}}$ 最小保持时间		ns

- (1) 主模式/从模式 (SPICLK) 和时钟相位 (SPICLK) 都被清零。
- (2)  $t_c$ (SPC) = SPI 时钟周期 = LSPCLK/4 或 LSPCLK/(SPIBRR+1)。
- (3)  $t_c$ (LCO) = LSPCLK 时钟周期。
- (4) 必须调整内部时钟预分频器，使 SPI 时钟限制在以下 SPI 时钟速率：主模式最大 25MHz 发送，最大 12.5MHz 接收；从模式最大 12.5MHz 发送，最大 12.5MHz 接收。
- (5) 参考的 SPICLK 信号的有效边沿由时钟极性 (SPICCR. 6) 控制。

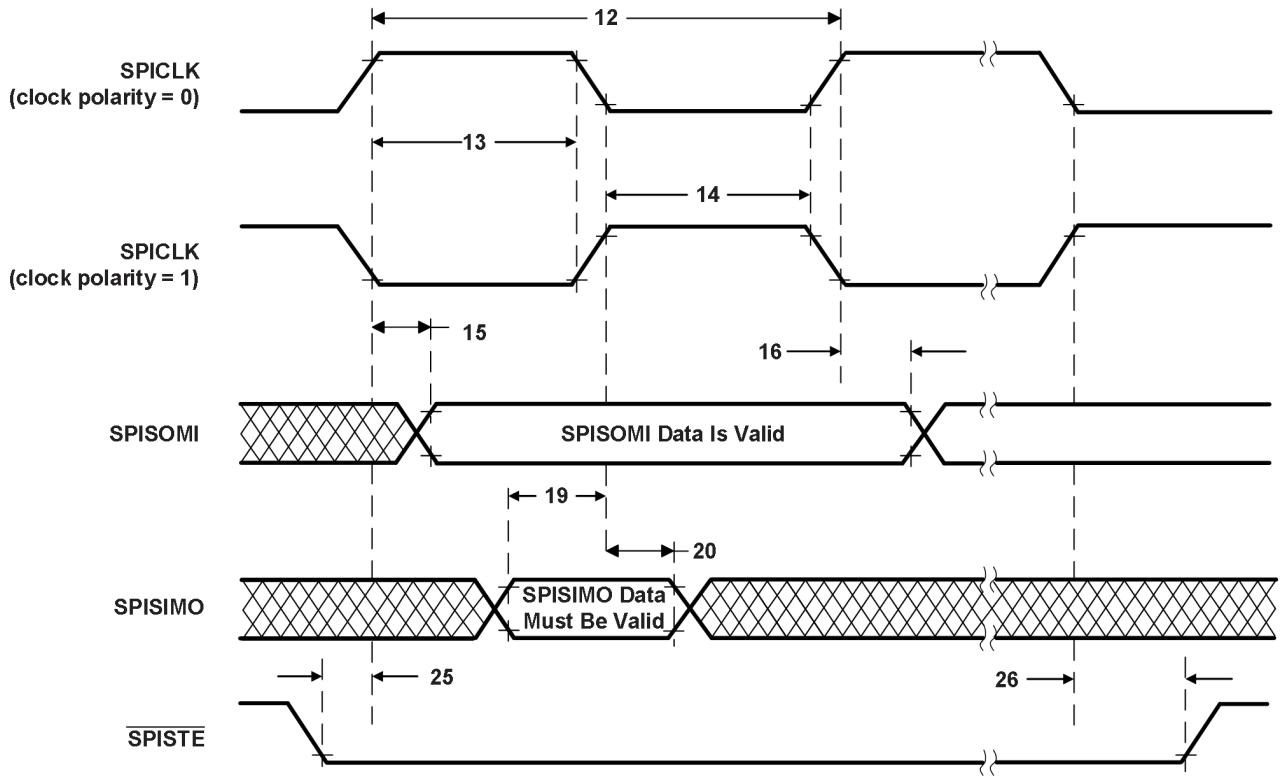


图 5-26 SPI 从模式外部时序(Clock Phase = 0)

5.6.4.2.2 SPI 从模式外部时序(Clock Phase = 1)

No.	参数(1) (2) (3) (4)	最小值	最大值	单位
12	$t_c$ (SPC) S SPICLK周期	$4t_c$ (SYSCLK)		ns
13	$t_w$ (SPC1) S SPICLK首脉宽	$2t_c$ (SYSCLK) <sup>-1</sup>		ns
14	$t_w$ (SPC2) S SPICLK次脉宽	$2t_c$ (SYSCLK) <sup>-1</sup>		ns
17	$t_d$ (SOMI) S SPICLK到SPISOMI输出最大延迟时间		21	ns
18	$t_v$ (SOMI) S SPICLK到SPISOMI数据失效最短时间	0		ns
21	$t_{su}$ (SIMO) S SPICLK前, 从机SPISIMO有效建立最短时间	$1.5t_c$ (SYSCLK)		ns
22	$t_h$ (SIMO) S SPICLK后, 从机SPISIMO数据保持最短时间	$1.5t_c$ (SYSCLK)		ns
25	$t_{su}$ (STE) S SPICLK前, $\overline{SPISTE}$ 最小建立时间	$1.5t_c$ (SYSCLK)		ns
26	$t_h$ (STE) S SPICLK后, $\overline{SPISTE}$ 最小保持时间	$1.5t_c$ (SYSCLK)		ns

- (1) 主模式/从模式 (SPICTL. 2) 和时钟相位 (SPICTL. 3) 都被清零。
- (2)  $t_c$ (SPC) = SPI 时钟周期 = LSPCLK/4 或 LSPCLK/(SPIBRR+1)。
- (3) 必须调整内部时钟预分频器, 使 SPI 时钟限制在以下 SPI 时钟速率: 主模式最大 25MHz 发送, 最大 12.5MHz 接收; 从模式最大 12.5MHz 发送, 最大 12.5MHz 接收。
- (4) 参考的 SPICLK 信号的有效边沿由时钟极性 (SPICCR. 6) 控制。

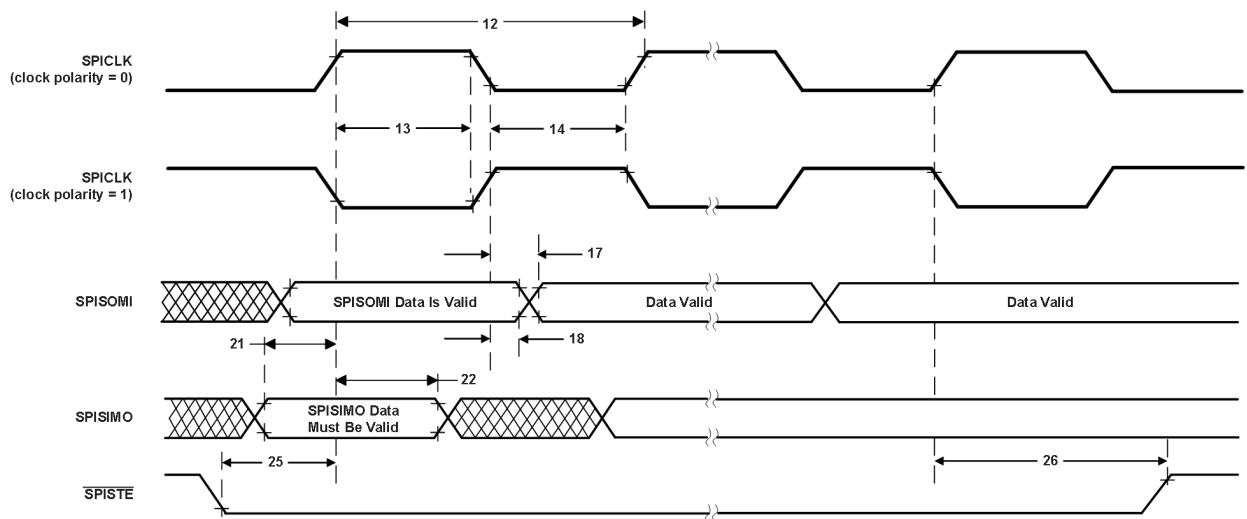


图 5-27 SPI 从模式外部时序(Clock Phase = 1)

### 5.6.5 串行通信接口模块 (SCI)

设备包括两个 SCI 串行通信接口模块 (SCI-A、SCI-B)。SCI 模块支持 CPU 和其他异步外设之间的数字通信，这些外设使用标准的非归零 (NRZ) 格式。SCI 接收端和发送端是双缓冲的，每个都有自己独立的使能位和中断位。两者都可以独立运行，也可以在全双工模式下同时运行。为了确保数据的完整性，SCI 检查接收数据的中断检测、奇偶校验、溢出和组帧错误。通过 16 位波特选择寄存器，比特率可编程超过 65000 个不同的速率。

每个 SCI 模块的特点包括：

- 两个外部引脚
  - SCITXD: SCI 发送输出引脚
  - SCIRXD: SCI 接收输入引脚

---

**注意**

如果不用于 SCI，两个引脚都可以用作 GPIO。

---

- 波特率可编程为 64K 不同速率

$$\text{波特率} = \frac{\text{LSPCLK}}{(\text{BRR}+1) \times 8} \quad \text{当 BRR} \neq 0$$

$$\text{波特率} = \frac{\text{LSPCLK}}{16} \quad \text{当 BRR} = 0$$

- 数据字格式
  - 一个起始位
  - 数据字长度可从 1 到 8 位编程
  - 可选偶/奇/无奇偶校验位
  - 1 或 2 个停止位
- 四个错误检测标志：奇偶校验、溢出、组帧和中断检测
- 两种唤醒多处理器模式：空闲线和地址位



- 半双工或全双工操作
- 双缓冲接收和发送功能
- 发送和接收操作可以通过中断驱动或带状态标志的轮询算法完成。
  - 发送端：TXRDY 标志（发送缓冲寄存器准备接受另一个字符）和 TX EMPTY 标志（发送移位寄存器为空）。
  - 接收端：RXRDY 标志（接收缓冲寄存器组合准备接收另一个字符），BRKDT 标志（中断条件发生）和 RX ERROR 标志（监视四种中断条件）。
- 发送端和接收端中断使能位分开（BRKDT 除外）。
- NRZ（不归零制）格式。

**注意**

这个模块中的所有寄存器都是连接到外围帧 2 的 8 位寄存器。当访问寄存器时，寄存器数据位于下字节（7-0），上字节（15-8）被读为 0。写入上字节没有效果。

增强功能：

- 自动波特率检测硬件逻辑
- 4 级发送/接收 FIFO

下表列出了 SCI 端口配置和控制寄存器。

表 5-21 SCI-A 模块

名称(1)	地址	大小(×16)	EALLOW保护	寄存器描述
SCICCRA	0x7050	1	No	SCI-A Communications Control Register
SCICTL1A	0x7051	1	No	SCI-A Control Register 1
SCIHBAUDA	0x7052	1	No	SCI-A Baud Register, High Bits
SCILBAUDA	0x7053	1	No	SCI-A Baud Register, Low Bits
SCICTL2A	0x7054	1	No	SCI-A Control Register 2
SCIRXSTA	0x7055	1	No	SCI-A Receive Status Register
SCIRXEMUA	0x7056	1	No	SCI-A Receive Emulation Data Buffer Register
SCIRXBUFA	0x7057	1	No	SCI-A Receive Data Buffer Register
SCITXBUFA	0x7059	1	No	SCI-A Transmit Data Buffer Register

名称(1)	地址	大小(×16)	EALLOW保护	寄存器描述
SCIFFTXA(2)	0x705A	1	No	SCI-A FIFO Transmit Register
SCIFFRXA(2)	0x705B	1	No	SCI-A FIFO Receive Register
SCIFFCTA(2)	0x705C	1	No	SCI-A FIFO Control Register
SCIPRIA	0x705F	1	No	SCI-A Priority Control Register

- (1) 该表中的寄存器映射到外围帧 2 空间。这个空间只允许 16 位访问。32 位访问产生未定义的结果。
- (2) 这些寄存器是 FIFO 模式的新寄存器。

**表 5-22 SCI-B 模块**

名称(1)	地址	大小(×16)	寄存器描述
SCICCRB	0x7750	1	SCI-B Communications Control Register
SCICTL1B	0x7751	1	SCI-B Control Register 1
SCIHBAUDB	0x7752	1	SCI-B Baud Register, High Bits
SCILBAUDB	0x7753	1	SCI-B Baud Register, Low Bits
SCICTL2B	0x7754	1	SCI-B Control Register 2
SCIRXSTB	0x7755	1	SCI-B Receive Status Register
SCIRXEMUB	0x7756	1	SCI-B Receive Emulation Data Buffer Register
SCIRXBUFB	0x7757	1	SCI-B Receive Data Buffer Register
SCITXBUFB	0x7759	1	SCI-B Transmit Data Buffer Register
SCIFFTXB(2)	0x775A	1	SCI-B FIFO Transmit Register
SCIFFRXB(2)	0x775B	1	SCI-B FIFO Receive Register
SCIFFCTB(2)	0x775C	1	SCI-B FIFO Control Register
SCIPRIB	0x775F	1	SCI-B Priority Control Register

- (1) 该表中的寄存器映射到外围帧 2 空间。这个空间只允许 16 位访问。32 位访问产生未定义的结果。
- (2) 这些寄存器是 FIFO 模式的新寄存器。

SCI 模块框图如图 5-28 所示。

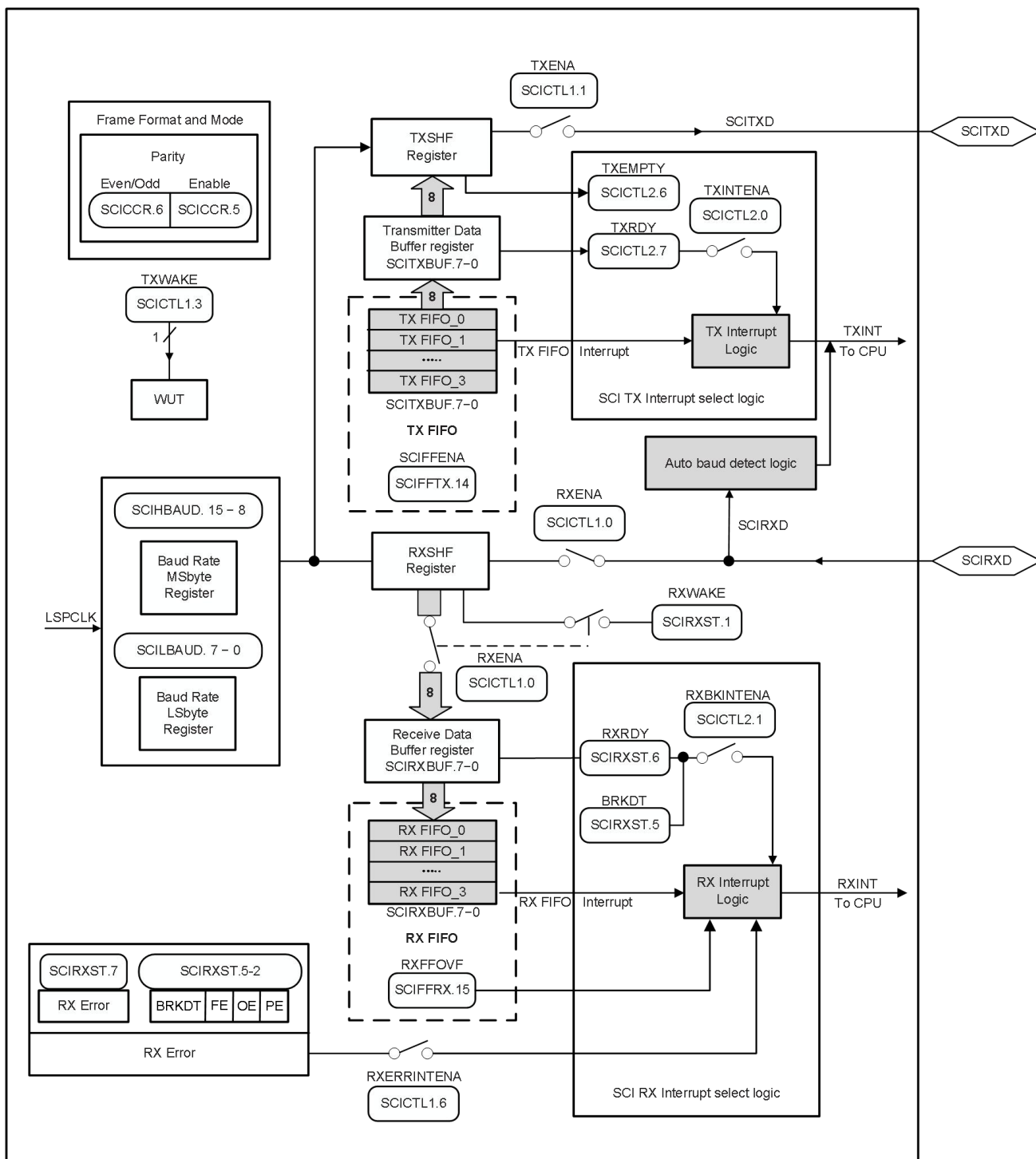


图 5-28 串行通讯接口 (SCI) 模块框图

## 5.6.6 多通道缓冲串口模块 (McBSP)

McBSP 模块具有以下特性:

- 兼容 AVP32F335 和 ADP32F12 产品中的 McBSP 模块。
- 全双工通信
- 双缓冲数据寄存器, 允许连续的数据流。
- 发送和接收均有独立的帧和时钟。
- 外部移位时钟生成或内部可编程频移时钟。
- 广泛的数据大小选择, 包括 8 位、12 位、16 位、20 位、24 位或 32 位。
- LSB 或 MSB 优先的 8 位数据传输。
- 帧同步和数据时钟均带可编程极性。
- 高度可编程的内部时钟和帧生成。
- 直接接口到工业标准编解码器, 模拟接口芯片 (AICs), 和其他串行连接的模数 (A/D) 和数模 (D/A) 设备。
- 与 SPI 兼容的设备工作。
- 在 McBSP 上可以支持以下应用程序接口:
  - T1/E1 制定者
  - IOM-2 兼容设备
  - AC97 兼容设备 (提供了必要的多相帧同步能力。)
  - IIS 兼容设备
  - SPI
- McBSP 时钟频率,

$$\text{CLKG} = \frac{\text{CLKSRG}}{(1 + \text{CLKGDV})}$$

其中 CLKSRG 源可以是 LSPCLK、CLKX 或 CLKR。串口性能受 I/O 缓冲区切换速度的限制。

内部预调器必须进行调整，使外围速度小于 I/O 缓冲区速度限制。

**注意**

参见 8.9 节了解最大 I/O 引脚切换速度。

**注意**

在 80 脚封装上，只支持 McBSP 的时钟停止模式 SPI (clock-stop mode)。

McBSP 模块框图如下图所示。

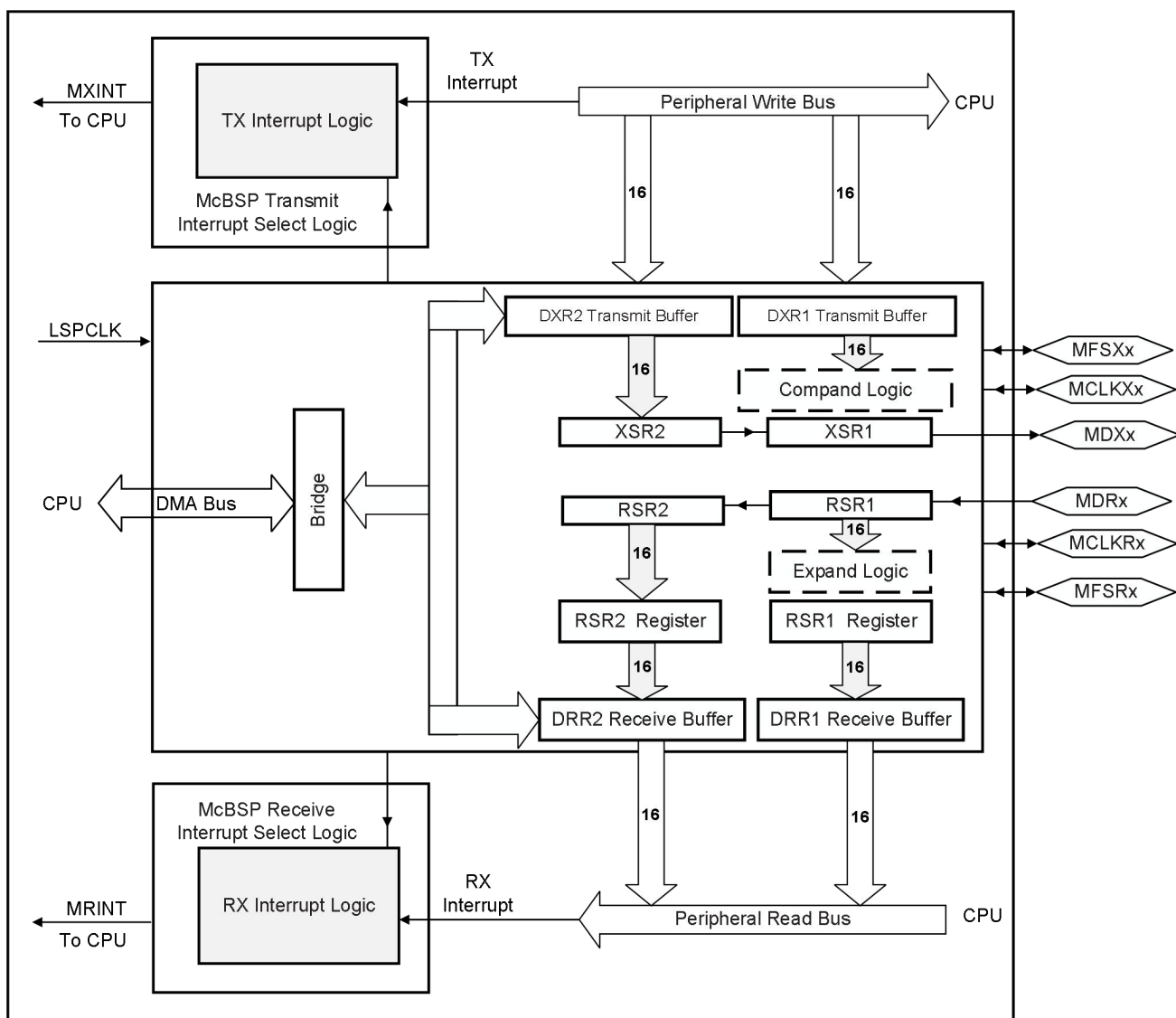


图 5-29 McBSP 模块框图

表 5-23 提供了 McBSP 寄存器的摘要。

表 5-23 McBSP 寄存器汇总

名称	地址	类型	默认值	寄存器描述
数据收发寄存器				
DRR2	0x5000	R	0x0000	McBSP Data Receive Register 2
DRR1	0x5001	R	0x0000	McBSP Data Receive Register 1
DXR2	0x5002	W	0x0000	McBSP Data Transmit Register 2
DXR1	0x5003	W	0x0000	McBSP Data Transmit Register 1
McBSP控制寄存器				
SPCR2	0x5004	R/W	0x0000	McBSP Serial Port Control Register 2
SPCR1	0x5005	R/W	0x0000	McBSP Serial Port Control Register 1
RCR2	0x5006	R/W	0x0000	McBSP Receive Control Register 2
RCR1	0x5007	R/W	0x0000	McBSP Receive Control Register 1
XCR2	0x5008	R/W	0x0000	McBSP Transmit Control Register 2
XCR1	0x5009	R/W	0x0000	McBSP Transmit Control Register 1
SRGR2	0x500A	R/W	0x0000	McBSP Sample Rate Generator Register 2
SRGR1	0x500B	R/W	0x0000	McBSP Sample Rate Generator Register 1
多通道控制寄存器				
MCR2	0x500C	R/W	0x0000	McBSP Multichannel Register 2
MCR1	0x500D	R/W	0x0000	McBSP Multichannel Register 1
RCERA	0x500E	R/W	0x0000	McBSP Receive Channel Enable Register Partition A
RCERB	0x500F	R/W	0x0000	McBSP Receive Channel Enable Register Partition B
XCERA	0x5010	R/W	0x0000	McBSP Transmit Channel Enable Register Partition A
XCERB	0x5011	R/W	0x0000	McBSP Transmit Channel Enable Register Partition B
PCR	0x5012	R/W	0x0000	McBSP Pin Control Register
RCERC	0x5013	R/W	0x0000	McBSP Receive Channel Enable Register Partition C
RCERD	0x5014	R/W	0x0000	McBSP Receive Channel Enable Register Partition D
XCERC	0x5015	R/W	0x0000	McBSP Transmit Channel Enable Register Partition C
XCERD	0x5016	R/W	0x0000	McBSP Transmit Channel Enable Register Partition D
RCERE	0x5017	R/W	0x0000	McBSP Receive Channel Enable Register Partition E
RCERF	0x5018	R/W	0x0000	McBSP Receive Channel Enable Register Partition F
XCERE	0x5019	R/W	0x0000	McBSP Transmit Channel Enable Register Partition E
XCERF	0x501A	R/W	0x0000	McBSP Transmit Channel Enable Register Partition F
RCERG	0x501B	R/W	0x0000	McBSP Receive Channel Enable Register Partition G
RCERH	0x501C	R/W	0x0000	McBSP Receive Channel Enable Register Partition H
XCERG	0x501D	R/W	0x0000	McBSP Transmit Channel Enable Register Partition G
XCERH	0x501E	R/W	0x0000	McBSP Transmit Channel Enable Register Partition H
MFFINT	0x5023	R/W	0x0000	McBSP Interrupt Enable Register

### 5.6.6.1 McBSP 电气数据/时序

#### 5.6.6.1.1 McBSP发送和接收时序

##### 5.6.6.1.1.1 McBSP时序要求

No. (1) (2)			最小值	最大值	单位
	McBSP模块时钟范围 (CLKG, CLKX, CLKR)		1		kHz
				20 <sup>(3)</sup> (4)	MHz
	McBSP模块周期范围 (CLKG, CLKX, CLKR)		50 <sup>(4)</sup>		ns
				1	ms
M11	$t_c$ (CLKR/X)	CLKR/X周期	外部CLKR/X	2P	ns
M12	$t_w$ (CLKR/X)	CLKR/X高或低脉宽	外部CLKR/X	P - 7	ns
M13	$t_r$ (CLKR/X)	CLKR/X上升时间	外部CLKR/X		7 ns
M14	$t_f$ (CLKR/X)	CLKR/X下降时间	外部CLKR/X		7 ns
M15	$t_{su}$ (FSRH-CLKRL)	外部接收帧同步信号建立时间	内部CLKR	18	ns
			外部CLKR	2	
M16	$t_h$ (CLKRL-FSRH)	外部接收帧同步信号保持时间	内部CLKR	0	ns
			外部CLKR	6	
M17	$t_{su}$ (DRV-CLKRL)	有效接收数据信号建立时间	内部CLKR	18	ns
			外部CLKR	2	
M18	$t_h$ (CLKRL-DRV)	有效接收数据信号保持时间	内部CLKR	0	ns
			外部CLKR	6	
M19	$t_{su}$ (FSXH-CLKXL)	外部发送帧同步信号建立时间	内部CLKR	18	ns
			外部CLKR	2	
M20	$t_h$ (CLKXL-FSXH)	外部发送帧同步信号保持时间	内部CLKR	0	ns
			外部CLKR	6	

- (1) 极性位 CLKRP=CLKXP=FSRP=FSXP=0。如果任意信号的极性反转，则该信号的时序参考也反转。
- (2)  $2P=1/CLKG$ ，单位 ns。CLKG 是采样率发生器 mux 输出。CLKG=clkserg/(1+clkdiv)。CLKSERG 可以是 LSPCLK, CLKX, CLKR 作为源。CLKSERG  $\leq$  (SYSCLKOUT/2)。McBSP 的性能受到 I/O 缓冲区切换速度的限制。
- (3) 内部时钟预调整必须调整，使 McBSP 时钟 (CLKG, CLKX, CLKR) 速度不大于 I/O 缓冲区速度限制 (20MHz)。
- (4) 对于内部 CLKR，最大 McBSP 模块时钟频率降低到 10MHz。

**5.6.6.1.1.2 McBSP开关特性**

在推荐操作条件范围内(除非另有说明)。

No.	参数 (1) (2)		最小值	最大值	单位	
M1	$t_c$ (CLKR/X)	CLKR/X周期	内部CLKR/X	2P	ns	
M2	$t_w$ (CLKR/XH)	CLKR/X高脉宽	内部CLKR/X	$D - 5^{(3)}$ $D + 5^{(3)}$	ns	
M3	$t_w$ (CLKR/XL)	CLKR/X低脉宽	内部CLKR/X	$C - 5^{(3)}$ $C + 5^{(3)}$	ns	
M4	$t_d$ (CLKRH-FSRV)	内部接收帧同步信号延迟范围 (输出)	内部CLKR	0    4	ns	
			外部CLKR	3    27		
M5	$t_d$ (CLKXH-FSXV)	内部发送帧同步信号延迟范围 (输出)	内部CLKX	0    4	ns	
			外部CLKX	3    27		
M6	$t_{dis}$ (CLKXH-DXHZ)	发送最后一位数据完, DX禁用延迟	内部CLKX	8	ns	
			外部CLKX	14		
M7	$t_d$ (CLKXH-DXV)	有效发送数据位延迟 (发送数据的第一位除外)	内部CLKX	9	ns	
			外部CLKX	28		
		有效发送数据位延迟 (发送数据的第一位)	DXENA=0	内部CLKX		8
			外部CLKX	14		
		仅适用于数据延迟为1或2模式 (XDATDLY=01b或10b)	DXENA=1	内部CLKX		P + 8
			外部CLKX	P + 14		
M8	$t_{en}$ (CLKXH-DX)	数据发送使能时间  仅适用于数据延迟为0模式 (XDATDLY=00b)的首位发送	DXENA=0	内部CLKX	0	ns
			外部CLKX	6		
			DXENA=1	内部CLKX	P	
			外部CLKX	P + 6		
M9	$t_d$ (FXH-DXV)	有效发送数据延迟  仅适用于数据延迟为0模式 (XDATDLY=00b)的首位发送	DXENA=0	内部FSX	8	ns
			外部FSX	14		
			DXENA=1	内部FSX	P + 8	
			外部FSX	P + 14		
M10	$t_{en}$ (FXH-DX)	数据发送使能时间  仅适用于数据延迟为0模式 (XDATDLY=00b)的首位发送	DXENA=0	内部FSX	0	ns
			外部FSX	6		
			DXENA=1	内部FSX	P	
			外部FSX	P + 6		

(1) 极性位 CLKRP=CLKXP=FSRP=FSXP=0。如果任意信号的极性反转, 则该信号的时序参考也反转。

(2)  $2P=1/CLKG$ , 单位 ns。

(3)  $C = CLKR/X$  低脉宽 = P;  $D = CLKR/X$  高脉宽 = P



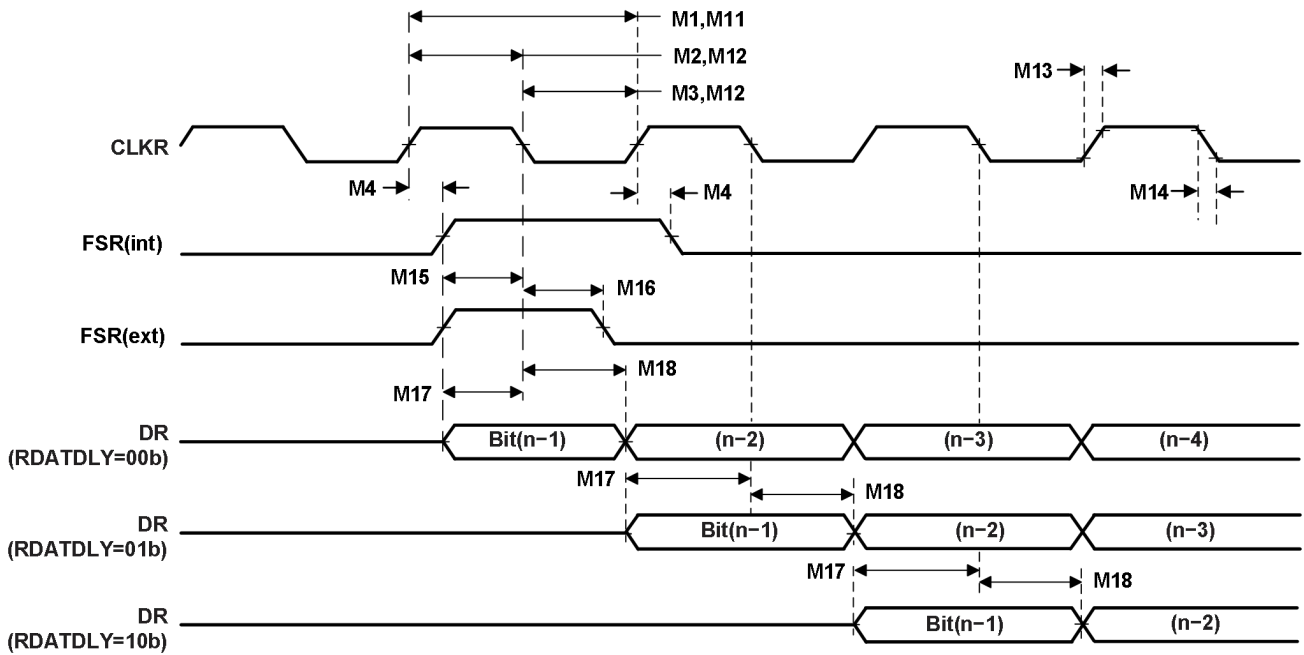


图 5-30 McBSP 接收时序

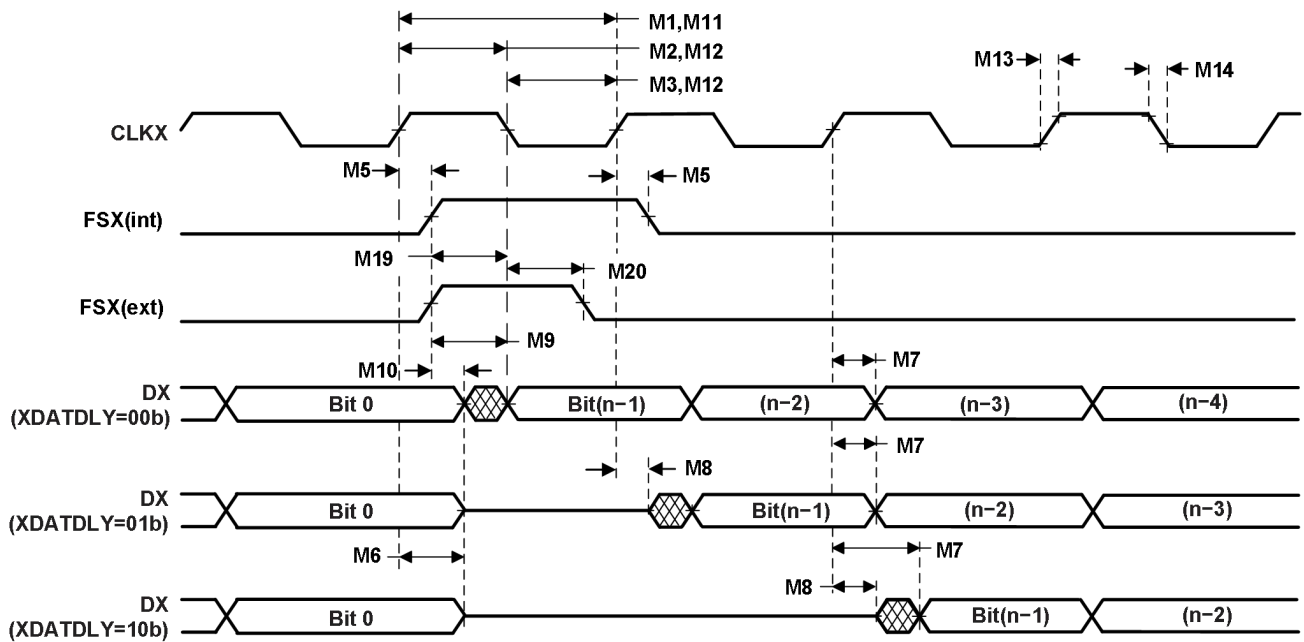


图 5-31 McBSP 发送时序

### 5.6.6.1.2 McBSP作为SPI用的时序

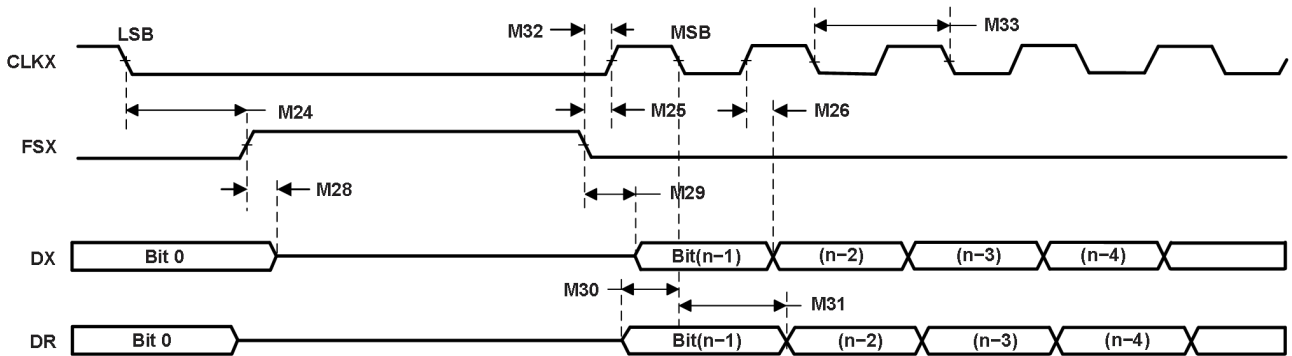


图 5-32 McBSP 作为 SPI 主或从模式的时序: CLKSTP=10b, CLKXP=0

#### 5.6.6.1.2.1 McBSP作为SPI主或从模式的时序要求(CLKSTP=10b, CLKXP=0)

No. (1)	参数	主模式		从模式		单位
		最小值	最大值	最小值	最大值	
M30	$t_{su}(DRV-CKXL)$ DR 建立时间: DR 有效开始至 CLKX 下降沿前	30		8P - 10		ns
M31	$t_h(CKXL-DRV)$ DR 保持时间: CLKX 下降沿后至 DR 有效结束	1		8P - 10		ns
M32	$t_{su}(BFXL-CKXH)$ FSX 建立时间: 作 SPI 从时, FSX 下降沿开始至 CLKX 上升沿 (对比 M25)			8P + 10		ns
M33	$t_c(CKX)$ CLKX 周期	2P <sup>(2)</sup>		16P		ns

(1) 对于所有的SPI从模式, CLKX必须至少是8个CLKG周期。此外, 通过设置CLKSM=CLKGDV=1, CLKG应该为LSPCLK/2。

(2) 2P = 1/CLKG

#### 5.6.6.1.2.2 McBSP作为SPI主或从模式的开关特性(CLKSTP=10b, CLKXP=0)

在推荐操作条件范围内, 除非另有说明。

No.	参数	主模式		从模式		单位
		最小值	最大值	最小值	最大值	
M24	$t_h(CKXL-FXL)$ FSX 保持时间: CLKX 最后一个时钟置低后 FSX 维持低有效的时间。	2P <sup>(1)</sup>				ns
M25	$t_d(FXL-CKXH)$ CLKX 延迟时间: 作 SPI 主时, FSX 低有效后到 CLKX 第一个上升沿 (对比 M32)	P				ns
M26	$t_d(CKXH-DXV)$ 延迟时间, CLKX 高到 DX 有效	-2	0	3P + 6	5P + 20	ns
M28	$t_{dis}(FXH-DXHZ)$ 禁用时间, 从 FSX 高到 DX 最后一个数据位后的高阻抗输出	6		6P + 6		ns
M29	$t_d(FXL-DXV)$ 延迟时间, FSX 低到 DX 有效	6		4P + 6		ns

(1) 2P=1/CLKG

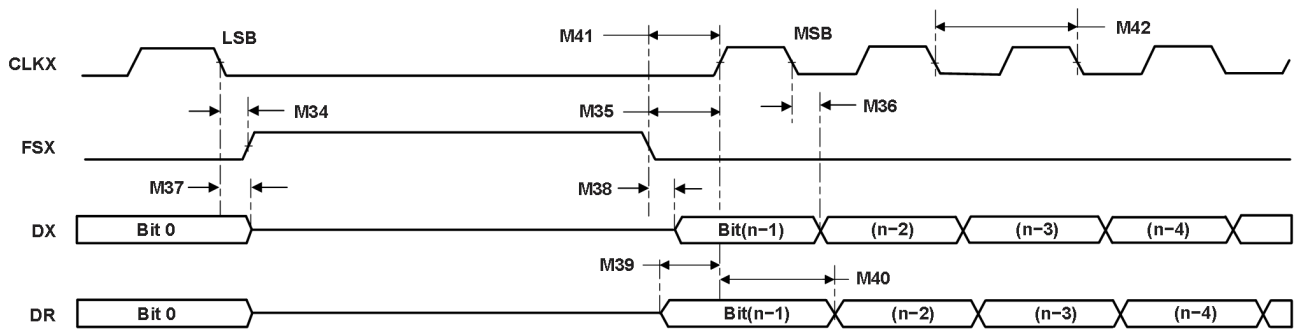


图 5-33 McBSP 作为 SPI 主或从模式的时序: CLKSTP=11b, CLKXP=0

### 5.6.6.1.2.3 McBSP作为SPI主或从模式的时序要求(CLKSTP=11b, CLKXP=0)

No. (1)	参数	主模式		从模式		单位
		最小值	最大值	最小值	最大值	
M39	$t_{su}(DRV-CKXH)$ 建立时间, DR 在 CLKX 高之前有效	30		8P - 10		ns
M40	$t_h(CKXH-DRV)$ 保持时间, DR 在 CLKX 高后有效	1		8P - 10		ns
M41	$t_{su}(FXL-CKXH)$ 建立时间, FSX 低, CLKX 高			16P + 10		ns
M42	$t_c(CKX)$ 周期时间, CLKX	$2P^{(2)}$		16P		ns

(1) 对于所有的 SPI 从模式, CLKX 必须至少是 8 个 CLKG 周期。此外, 通过设置 CLKSM=CLKGDV=1, CLKG 应该为 LSPCLK/2。

(2)  $2P=1/CLKG$

### 5.6.6.1.2.4 McBSP作为SPI主或从模式的开关特性(CLKSTP=11b, CLKXP=0)

在推荐操作条件范围内, 除非另有说明。

No.	参数	主模式		从模式		单位
		最小值	最大值	最小值	最大值	
M34	$t_h(CKXL-FXL)$ 保持时间, CLKX 低后 FSX 低	P				ns
M35	$t_d(FXL-CKXH)$ 延迟时间, FSX 低到 CLKX 高	$2P^{(1)}$				ns
M36	$t_d(CKXL-DXV)$ 延迟时间, CLKX 低到 DX 有效	- 2	0	3P + 6	5P + 20	ns
M37	$t_{dis}(CKXL-DXHZ)$ 禁用时间, DX 高阻抗后的最后一个数据位从 CLKX 低	P + 6		7P + 6		ns
M38	$t_d(FXL-DXV)$ 延迟时间, FSX 低到 DX 有效	6		4P + 6		ns

(1)  $2P=1/CLKG$

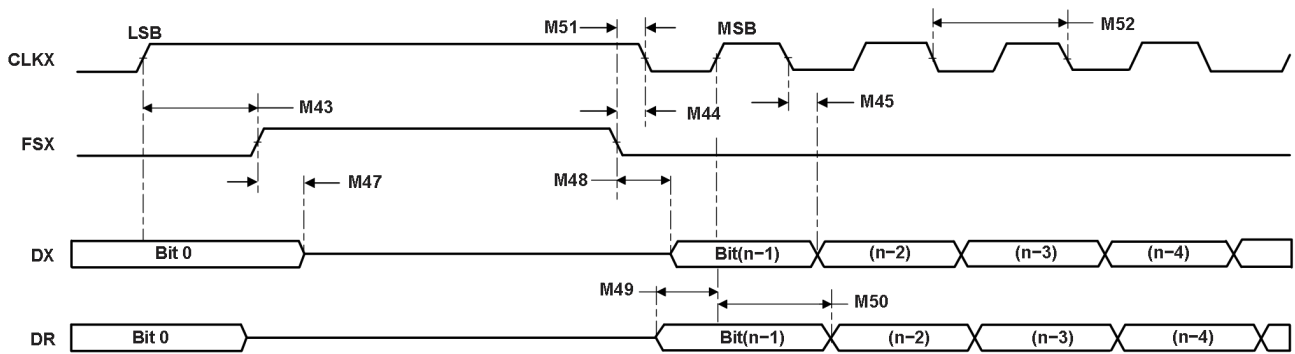


图 5-34 McBSP 作为 SPI 主或从时序: CLKSTP=10b, CLKXP=1

### 5.6.6.1.2.5 McBSP作为SPI主或从模式的时序要求(CLKSTP=10b, CLKXP=1)

No. (1)	参数	主模式		从模式		单位
		最小值	最大值	最小值	最大值	
M49	$t_{su}(DRV-CKXH)$ 建立时间, DR 在 CLKX 高之前有效	30		8P - 10		ns
M50	$t_h(CKXH-DRV)$ 保持时间, DR 在 CLKX 高后有效	1		8P - 10		ns
M51	$t_{su}(FXL-CKXL)$ 建立时间, FSX 低, CLKX 低			8P + 10		ns
M52	$t_c(CKX)$ 周期时间, CLKX	$2P^{(2)}$		16P		ns

(1) 对于所有的 SPI 从模式, CLKX 必须至少是 8 个 CLKG 周期。此外, 通过设置 CLKSM=CLKGDV=1, CLKG 应该为 LSPCLK/2。

(2)  $2P=1/CLKG$

### 5.6.6.1.2.6 McBSP作为SPI主或从模式的开关特性(CLKSTP=10b, CLKXP=1)

在推荐操作条件范围内, 除非另有说明。

No.	参数	主模式		从模式		单位
		最小值	最大值	最小值	最大值	
M43	$t_h(CKXH-FXL)$ 保持时间, 在 CLKX 高后, FSX 较低	$2P^{(1)}$				ns
M44	$t_d(FXL-CKXL)$ 延迟时间, FSX 低至 CLKX 低	P				ns
M45	$t_d(CKXL-DXV)$ 延迟时间, CLKX 低到 DX 有效	-2	0	3P + 6	5P + 20	ns
M47	$t_{dis}(FXH-DXHZ)$ 禁用时间, DX 高阻抗后的最后一个数据位从 CLKX 低	6		6P + 6		ns
M48	$t_d(FXL-DXV)$ 延迟时间, FSX 低到 DX 有效	6		4P + 6		ns

(1)  $2P=1/CLKG$

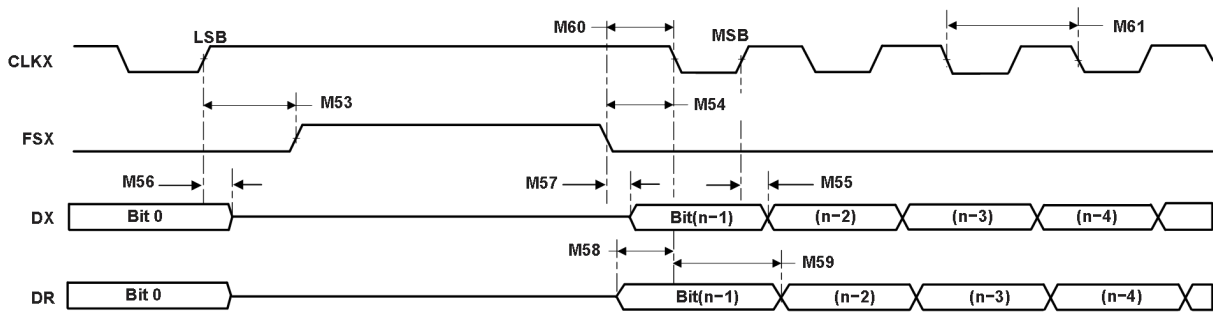


图 5-35 McBSP 作为 SPI 主或从时序: CLKSTP=11b, CLKXP=1

#### 5.6.6.1.2.7 McBSP作为SPI主或从模式的时序要求(CLKSTP=11b, CLKXP=1)

No.	参数	主模式		从模式		单位
		最小值	最大值	最小值	最大值	
M58	$t_{su}(DRV-CKXL)$ 建立时间, DR 在 CLKX 低之前有效	30		8P - 10		ns
M59	$t_h(CKXL-DRV)$ 保持时间, DR 在 CLKX 低后有效	1		8P - 10		ns
M60	$t_{su}(FXL-CKXL)$ 建立时间, FSX 低, CLKX 低			16P + 10		ns
M61	$t_c(CKX)$ 周期时间, CLKX	2P <sup>(2)</sup>		16P		ns

- (1) 对于所有的 SPI 从模式, CLKX 必须至少是 8 个 CLKG 周期。此外, 通过设置 CLKSM=CLKGDV=1, CLKG 应该为 LSPCLK/2。  
 (2) 2P=1/CLKG

#### 5.6.6.1.2.8 McBSP作为SPI主或从模式的开关特性(CLKSTP=11b, CLKXP=1)

在推荐操作条件范围内, 除非另有说明。

No.	参数	主模式		从模式		单位
		最小值	最大值	最小值	最大值	
M53	$t_h(CKXH-FXL)$ 保持时间, 在 CLKX 高后, FSX 较低	P				ns
M54	$t_d(FXL-CKXL)$ 延迟时间, FSX 低至 CLKX 低	2P <sup>(1)</sup>				ns
M55	$t_d(CKXH-DXV)$ 延迟时间, CLKX 高到 DX 有效	-2	0	3P + 6	5P + 20	ns
M56	$t_{dis}(CKXH-DXHZ)$ 禁用时间, DX 高阻抗后的最后一个数据位从 CLKX 高	P + 6		7P + 6		ns
M57	$t_d(FXL-DXV)$ 延迟时间, FSX 低到 DX 有效	6		4P + 6		ns

- (1) 2P=1/CLKG

### 5.6.7 增强型控制器局域网模块 (eCAN)

CAN 模块 (eCAN-A) 有以下特性:

- 全兼容 CAN 2.0B 版本协议。
- 最高支持 1Mbps 速率。
- 32 个邮箱, 每一个都有以下属性:
  - 可配置为接收或发送
  - 可配置为标准或扩展标识符
  - 具有可编程的接收掩码
  - 支持数据帧和远程帧
  - 由 0 到 8 字节的数据组成
  - 在接收和发送消息上使用 32 位时间戳
  - 防止接收新消息
  - 保持发送消息的动态可编程优先级
  - 采用具有两个中断级别的可编程中断方案
  - 在发送或接收超时采用可编程报警
- 低功耗模式
- 可编程总线唤醒
- 远程请求消息自动回复
- 在丢失仲裁或错误的情况下自动重传帧。
- 由特定消息同步的 32 位本地网络时间计数器(与邮箱 16 一起通信)。
- 自测试模式
  - 以回环模式运行, 接收自己的消息。提供了一个“虚拟”确认, 从而消除了另一个节点提

供确认位的需要。

注意

对于 90MHz 的 SYSCLKOUT, 可能的最小比特率是 6.25kbps。

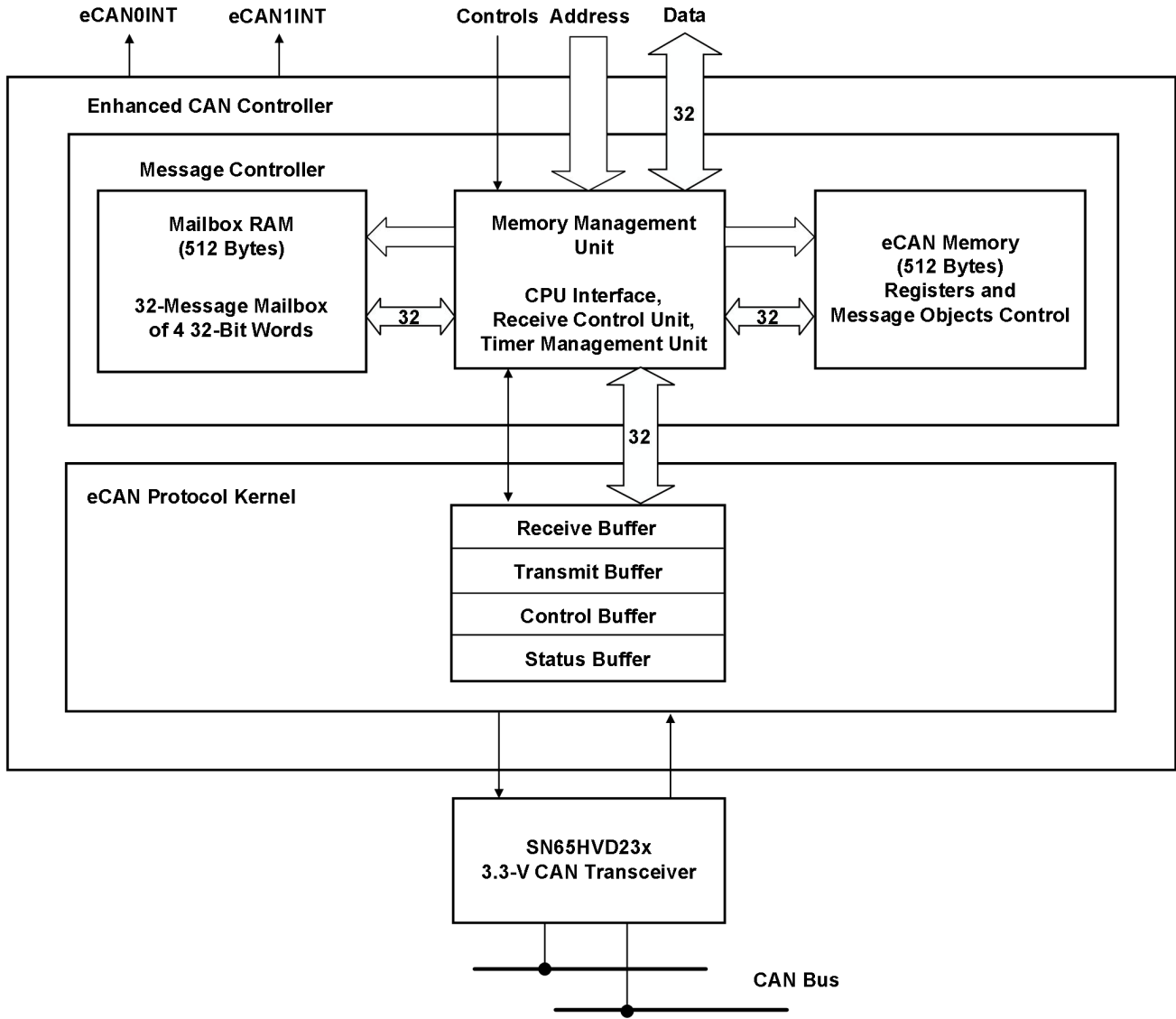


图 5-36 eCAN 框图和接口电路

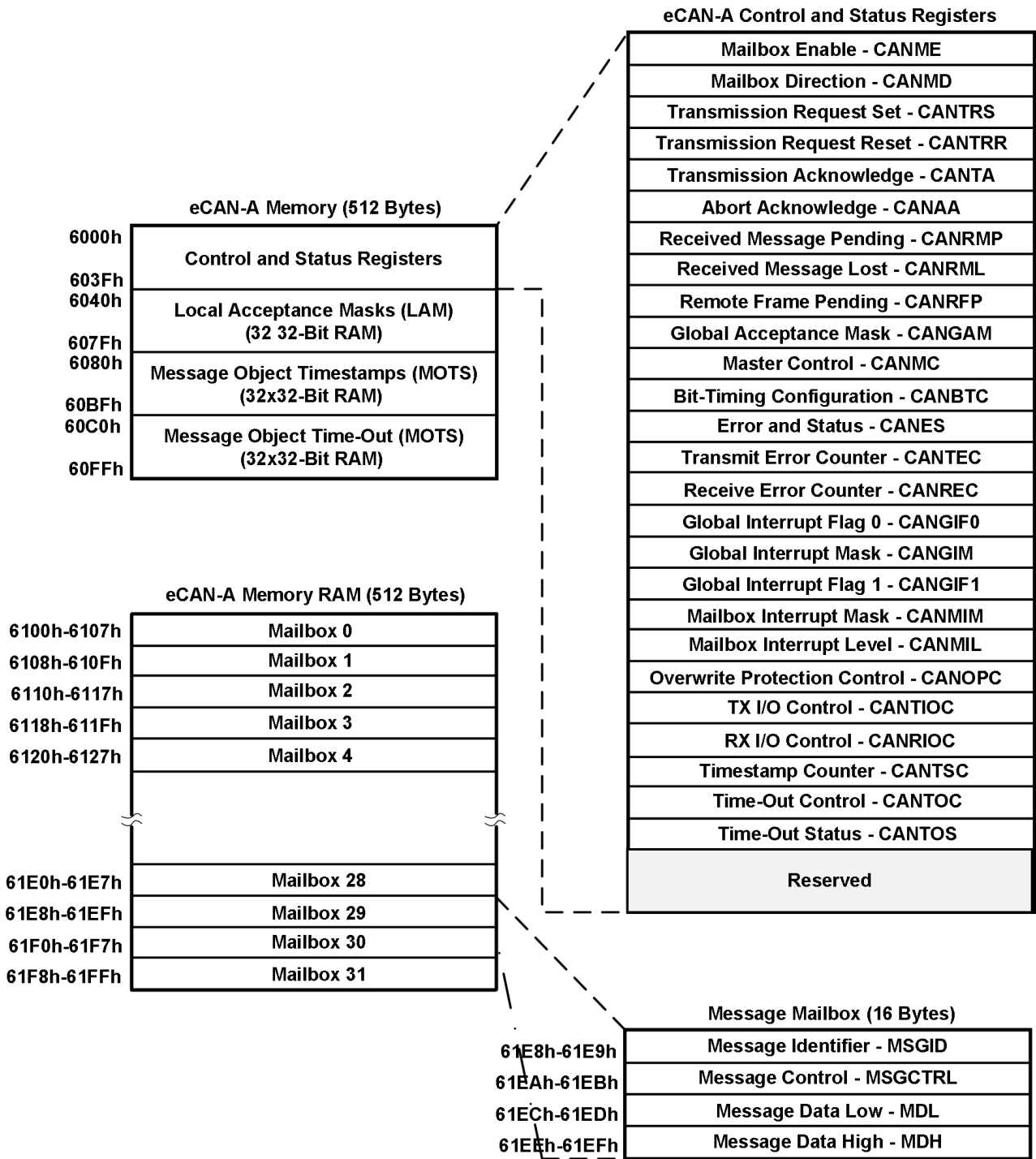


图 5-37 eCAN-A 内存映射

**注意**

如果在应用程序中没有使用 eCAN 模块，可用的 RAM(LAM、MOTS、MOTO 和邮箱 RAM)可以用作通用 RAM。CAN 模块时钟应该为此启用。



在下表中列出的 CAN 寄存器被 CPU 用来配置和控制 CAN 控制器和消息对象。eCAN 控制寄存器只支持 32 位的读写操作。邮箱 RAM 可以 16 位或 32 位方式被访问。所有 32 位访问都对齐到一个偶边界。

**表 5-24 eCAN-A 寄存器**

名称(1)	地址	大小(×32)	寄存器描述
CANME	0x6000	1	Mailbox enable
CANMD	0x6002	1	Mailbox direction
CANTRS	0x6004	1	Transmit request set
CANTRR	0x6006	1	Transmit request reset
CANTA	0x6008	1	Transmission acknowledge
CANAA	0x600A	1	Abort acknowledge
CANRMP	0x600C	1	Receive message pending
CANRML	0x600E	1	Receive message lost
CANRFP	0x6010	1	Remote frame pending
CANGAM	0x6012	1	Global acceptance mask
CANMC	0x6014	1	Master control
CANBTC	0x6016	1	Bit-timing configuration
CANES	0x6018	1	Error and status
CANTEC	0x601A	1	Transmit error counter
CANREC	0x601C	1	Receive error counter
CANGIF0	0x601E	1	Global interrupt flag 0
CANGIM	0x6020	1	Global interrupt mask
CANGIF1	0x6022	1	Global interrupt flag 1
CANMIM	0x6024	1	Mailbox interrupt mask
CANMIL	0x6026	1	Mailbox interrupt level
CANOPC	0x6028	1	Overwrite protection control
CANTIOC	0x602A	1	TX I/O control
CANRIOC	0x602C	1	RX I/O control

(1) 这些寄存器被映射到外设帧 1。

## 5.6.8 集成电路 (I2C)

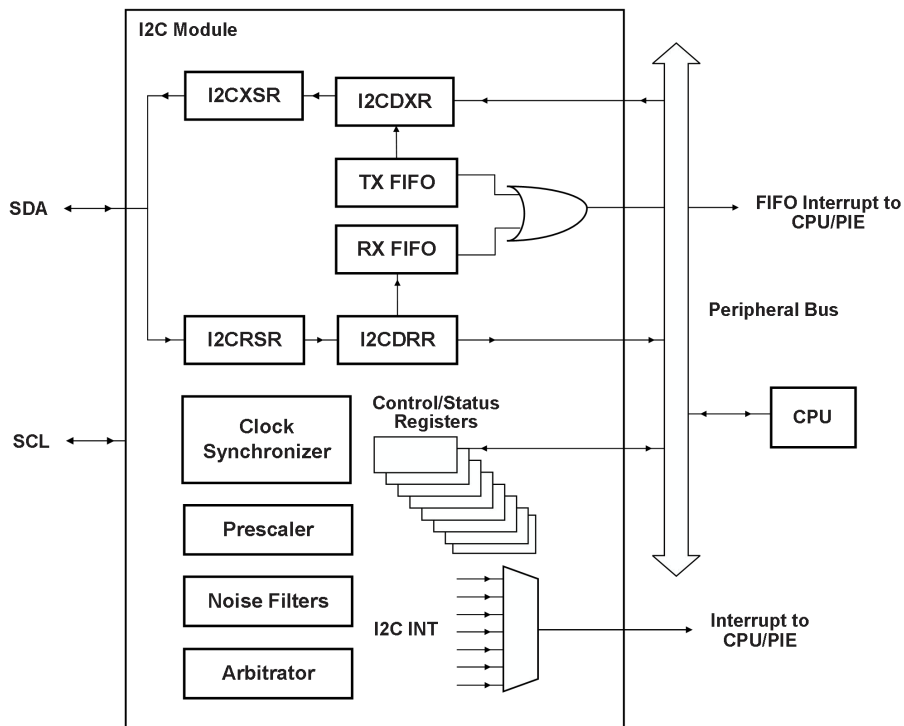
设备包含一个 I2C 串行端口，模块具有以下特性：

- 符合飞利浦半导体 I2C 总线规范 2.1 版本。
  - 支持 1 位到 8 到格式传输。
  - 7 位和 10 位寻址模式。
  - 通用呼叫功能。
  - 启动或起始 (START) 字节模式。
  - 支持多个主发送器和从接收器。
  - 支持多个主接收器和从发送器。
  - 具有主发送/接收和接收/发送模式。
  - 数据传输速率从 10 kbit/s 到 400 kbit/s (I2C 快速模式)。
- 一个 4 级接收 FIFO 和一个 4 级发送 FIFO。
- CPU 具有一个专用中断，该中断可以由以下条件产生：
  - 发送数据准备好
  - 接收数据准备好
  - 寄存器访问准备好
  - 无应答接收
  - 仲裁丢失
  - 检测到停止条件
  - 作为从器件寻址
- 当工作在 FIFO 模式，CPU 可以使用一个附加中断。

- 可以使能/禁止 I2C 模块。
- 自由数据格式模式。

I2C 总线不支持以下特性：

- 高速模式 (HS 模式)
- CBUS 兼容模式。



- A. I2C 寄存器以 SYSCLKOUT 速率访问。I2C 端口的内部定时和信号波形也是 SYSCLKOUT 速率。
- B. PCLKCRO 寄存器中的时钟使能位 (I2CAENCLK) 关闭 I2C 端口的时钟，用于低功耗操作。复位时 I2CAENCLK 为 clear，表示外部内部时钟关闭。

图 5-38 I2C 外设模块接口

**表 5-25 I2C-A 寄存器**

名称	地址	EALLOW保护	寄存器描述
I2COAR	0x7900	No	I2C own address register
I2CIER	0x7901	No	I2C interrupt enable register
I2CSTR	0x7902	No	I2C status register
I2CCLKL	0x7903	No	I2C clock low-time divider register
I2CCLKH	0x7904	No	I2C clock high-time divider register
I2CCNT	0x7905	No	I2C data count register
I2CDRR	0x7906	No	I2C data receive register
I2CSAR	0x7907	No	I2C slave address register
I2CDXR	0x7908	No	I2C data transmit register
I2CMDR	0x7909	No	I2C mode register
I2CISRC	0x790A	No	I2C interrupt source register
I2CPSC	0x790C	No	I2C prescaler register
I2CFFTX	0x7920	No	I2C FIFO transmit register
I2CFFRX	0x7921	No	I2C FIFO receive register
I2CRSR	-	No	I2C receive shift register (not accessible to the CPU)
I2CXSR	-	No	I2C transmit shift register (not accessible to the CPU)

### 5.6.8.1 I2C 电气数据/时序

#### 5.6.8.1.1 I2C 时序要求

		最小值	最大值	单位
$t_h$ (SDA-SCL) START	START 保持时间 (SDA 下降沿到 SCL 下降沿)	0.6		$\mu s$
$t_{su}$ (SCL-SDA) START	重复 START 建立时间 (SCL 上升沿到 SDA 下降沿)	0.6		$\mu s$
$t_h$ (SCL-DAT)	SCL 下降沿后的数据保持时间	0		$\mu s$
$t_{su}$ (DAT-SCL)	SCL 上升沿前的数据建立时间	100		ns
$t_r$ (SDA)	SDA 上升时间	20	300	ns
$t_r$ (SCL)	SCL 上升时间	20	300	ns
$t_f$ (SDA)	SDA 下降时间	11.4	300	ns
$t_f$ (SCL)	SCL 下降时间	11.4	300	ns
$t_{su}$ (SCL-SDA) STOP	STOP 建立时间 (SDA 上升沿到 SCL 上升沿)	0.6		$\mu s$

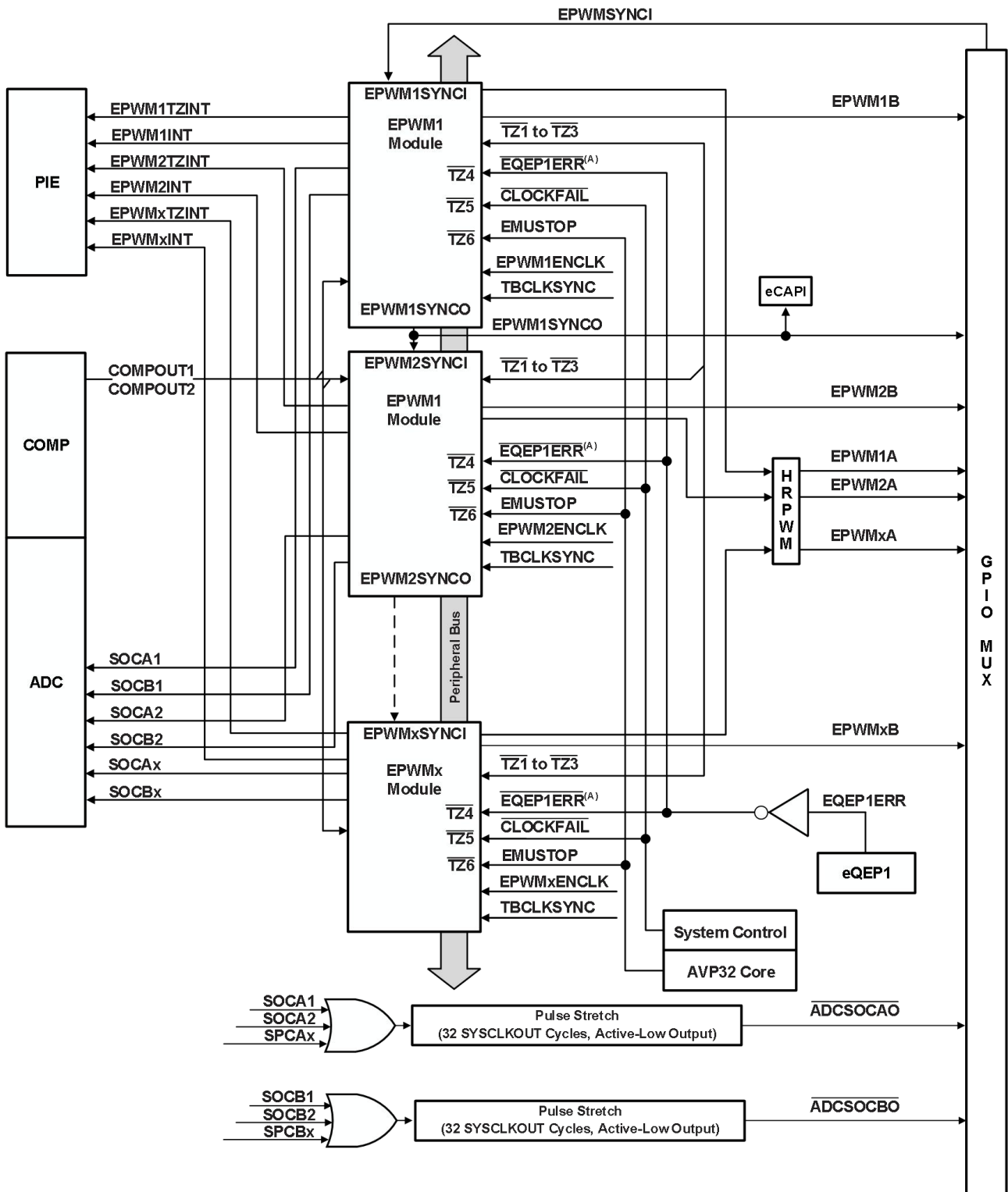
#### 5.6.8.1.2 I2C 开关特性

在推荐操作条件范围内，除非另有说明。

参数	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 7-12MHz，其预调频器和时钟分频器寄存器配置适当。		400	kHz
$V_{il}$ 低电平输入电压			$0.3 V_{DDIO}$	V
$V_{ih}$ 高电平输入电压		$0.7 V_{DDIO}$		V
$V_{hys}$ 输入迟滞		$0.05 V_{DDIO}$		V
$V_{ol}$ 低电平输出电压	3mA 吸收电流	0	0.4	V
$t_{LOW}$ SCL 时钟周期低电平	I2C 时钟模块频率 7-12MHz，其预分频器和时钟分频寄存器配置适当。	1.3		$\mu s$
$t_{HIGH}$ SCL 时钟周期高电平	I2C 时钟模块频率 7-12MHz，其预调频器和时钟分频寄存器配置适当。	0.6		$\mu s$
$I_I$ 从 $0.1 V_{DDIO}$ 到 $0.9 V_{DDIO}$ 最大输入电压下的输入电流		-10	10	$\mu A$

### 5.6.9 增强型脉宽调制器模块 (ePWM1~8)

器件包含 8 个增强型 PWM (ePWM) 模块。



A. 仅带有 eQEP1 模块的器件上才有此信号。

图 5-39 ePWM

**表 5-26 ePWM1–ePWM4 Control and Status Registers**

名称	ePWM1 地址	ePWM2 地址	ePWM3 地址	ePWM4 地址	大小(×16)/ #SHADOW	寄存器描述
TBCTL	0x6800	0x6840	0x6880	0x68C0	1/0	Time Base Control Register
TBSTS	0x6801	0x6841	0x6881	0x68C1	1/0	Time Base Status Register
TBPHSHR	0x6802	0x6842	0x6882	0x68C2	1/0	Time Base Phase HRPWM
TBPHS	0x6803	0x6843	0x6883	0x68C3	1/0	Time Base Phase Register
TBCTR	0x6804	0x6844	0x6884	0x68C4	1/0	Time Base Counter Register
TBPRD	0x6805	0x6845	0x6885	0x68C5	1/1	Time Base Period Register Set
TBPRDHR	0x6806	0x6846	0x6886	0x68C6	1/1	Time Base Period
CMPCTL	0x6807	0x6847	0x6887	0x68C7	1/0	Counter Compare Control
CMPAHR	0x6808	0x6848	0x6888	0x68C8	1/1	Time Base Compare A HRPWM
CMPA	0x6809	0x6849	0x6889	0x68C9	1/1	Counter Compare A Register Set
CMPB	0x680A	0x684A	0x688A	0x68CA	1/1	Counter Compare B Register Set
AQCTLA	0x680B	0x684B	0x688B	0x68CB	1/0	Action Qualifier Control Register
AQCTLB	0x680C	0x684C	0x688C	0x68CC	1/0	Action Qualifier Control Register
AQSFRC	0x680D	0x684D	0x688D	0x68CD	1/0	Action Qualifier Software Force
AQCSFRC	0x680E	0x684E	0x688E	0x68CE	1/1	Action Qualifier Continuous S/W
DBCTL	0x680F	0x684F	0x688F	0x68CF	1/1	Dead-Band Generator Control
DBRED	0x6810	0x6850	0x6890	0x68D0	1/0	Dead-Band Generator Rising
DBFED	0x6811	0x6851	0x6891	0x68D1	1/0	Dead-Band Generator Falling
TZSEL	0x6812	0x6852	0x6892	0x68D2	1/0	
TZDCSEL	0x6813	0x6853	0x6893	0x68D3	1/0	Trip Zone Digital Compare
TZCTL	0x6814	0x6854	0x6894	0x68D4	1/0	
TZEINT	0x6815	0x6855	0x6895	0x68D5	1/0	Trip Zone Enable Interrupt
TZFLG	0x6816	0x6856	0x6896	0x68D6	1/0	
TZCLR	0x6817	0x6857	0x6897	0x68D7	1/0	
TZFRC	0x6818	0x6858	0x6898	0x68D8	1/0	
ETSEL	0x6819	0x6859	0x6899	0x68D9	1/0	Event Trigger Selection Register
ETPS	0x681A	0x685A	0x689A	0x68DA	1/0	Event Trigger Prescale Register
ETFLG	0x681B	0x685B	0x689B	0x68DB	1/0	Event Trigger Flag Register
ETCLR	0x681C	0x685C	0x689C	0x68DC	1/0	Event Trigger Clear Register
ETFRC	0x681D	0x685D	0x689D	0x68DD	1/0	Event Trigger Force Register
PCCTL	0x681E	0x685E	0x689E	0x68DE	1/0	PWM Chopper Control Register
HRCNFG	0x6820	0x6860	0x68A0	0x68E0	1/0	HRPWM Configuration
HRMSTEP	0x6826	-	-	-	1/0	HRPWM MEP Step Register
HRPCTL	0x68 28	0x6868	0x68A8	0x68E8	1/0	High-resolution Period Control
TBPRDHRM	0x682A	0x686A	0x68AA	0x68EA	1/W(2)	Time Base Period HRPWM
TBPRDM	0x682B	0x686B	0x68AB	0x68EB	1/W(2)	Time Base Period Register Mirror
CMPAHRM	0x682C	0x686C	0x68AC	0x68EC	1/W(2)	Compare A HRPWM Register
CMPAM	0x682D	0x686D	0x68AD	0x68ED	1/W(2)	Compare A Register Mirror
DCTRIPSEL	0x6830	0x6870	0x68B0	0x68F0	1/0	Digital Compare Trip Select
DCACTL	0x6831	0x6871	0x68B1	0x68F1	1/0	Digital Compare A Control
DCBCTL	0x6832	0x6872	0x68B2	0x68F2	1/0	Digital Compare B Control
DCFCTL	0x6833	0x6873	0x68B3	0x68F3	1/0	Digital Compare Filter Control
DCCAPCT	0x6834	0x6874	0x68B4	0x68F4	1/0	Digital Compare Capture Control
DCOFFSET	0x6835	0x6875	0x68B5	0x68F5	1/1	Digital Compare Filter Offset
DCOFFSETCNT	0x6836	0x6876	0x68B6	0x68F6	1/0	Digital Compare Filter Offset
DCFWINDOW	0x6837	0x6877	0x68B7	0x68F7	1/0	Digital Compare Filter Window
DCFWINDOWCNT	0x6838	0x6878	0x68B8	0x68F8	1/0	Digital Compare Filter Window
DCCAP	0x6839	0x6879	0x68B9	0x68F9	1/1	Digital Compare Counter

- (1) 寄存器受 EALLOW 保护。  
 (2) W = 写入影子寄存器。

**表 5-27 ePWM5–ePWM8 Control and Status Registers**

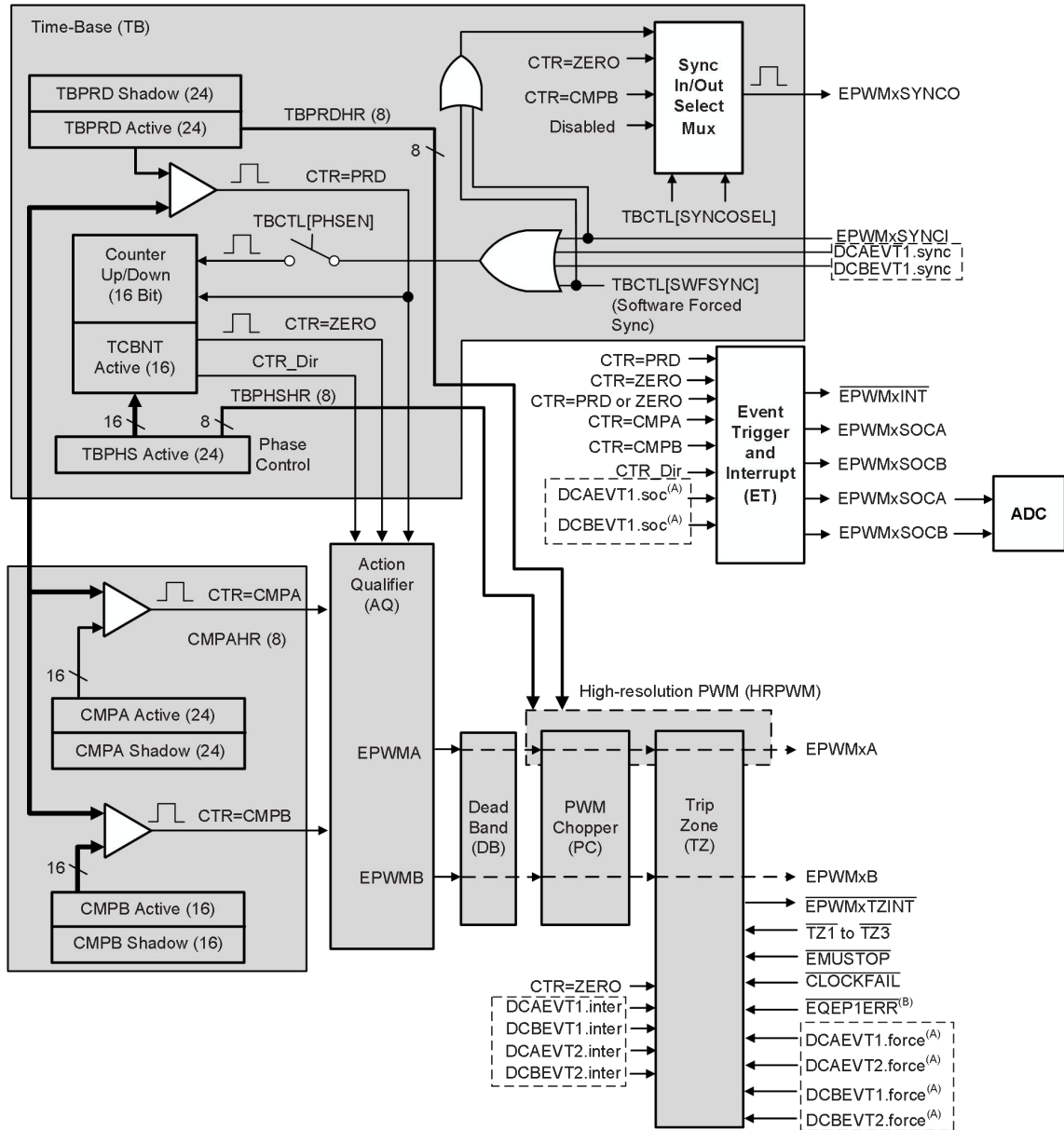
名称	ePWM1 地址	ePWM2 地址	ePWM3 地址	ePWM4 地址	大小(×16)/ #SHADOW	寄存器描述
TBCTL	0x6900	0x6940	0x6980	0x69C0	1/0	Time Base Control Register
TBSTS	0x6901	0x6941	0x6981	0x69C1	1/0	Time Base Status Register
TBPHSHR	0x6902	0x6942	0x6982	0x69C2	1/0	Time Base Phase HRPWM
TBPHS	0x6903	0x6943	0x6983	0x69C3	1/0	Time Base Phase Register
TBCTR	0x6904	0x6944	0x6984	0x69C4	1/0	Time Base Counter Register
TBPRD	0x6905	0x6945	0x6985	0x69C5	1/1	Time Base Period Register Set
TBPRDHR	0x6906	0x6946	0x6986	0x69C6	1/1	Time Base Period
CMPCTL	0x6907	0x6947	0x6987	0x69C7	1/0	Counter Compare Control
CMPAHR	0x6908	0x6948	0x6988	0x69C8	1/1	Time Base Compare A HRPWM
CMPA	0x6909	0x6949	0x6989	0x69C9	1/1	Counter Compare A Register Set
CMPB	0x690A	0x694A	0x698A	0x69CA	1/1	Counter Compare B Register Set
AQCTLA	0x690B	0x694B	0x698B	0x69CB	1/0	Action Qualifier Control Register
AQCTLB	0x690C	0x694C	0x698C	0x69CC	1/0	Action Qualifier Control Register
AQSFRC	0x690D	0x694D	0x698D	0x69CD	1/0	Action Qualifier Software Force
AQCSFRC	0x690E	0x694E	0x698E	0x69CE	1/1	Action Qualifier Continuous S/W
DBCTL	0x690F	0x694F	0x698F	0x69CF	1/1	Dead-Band Generator Control
DBRED	0x6910	0x6950	0x6990	0x69D0	1/0	Dead-Band Generator Rising
DBFED	0x6911	0x6951	0x6991	0x69D1	1/0	Dead-Band Generator Falling
TZSEL	0x6912	0x6952	0x6992	0x69D2	1/0	
TZDCSEL	0x6913	0x6953	0x6993	0x69D3	1/0	Trip Zone Digital Compare
TZCTL	0x6914	0x6954	0x6994	0x69D4	1/0	
TZEINT	0x6915	0x6955	0x6995	0x69D5	1/0	Trip Zone Enable Interrupt
TZFLG	0x6916	0x6956	0x6996	0x69D6	1/0	
TZCLR	0x6917	0x6957	0x6997	0x69D7	1/0	
TZFRC	0x6918	0x6958	0x6998	0x69D8	1/0	
ETSEL	0x6919	0x6959	0x6999	0x69D9	1/0	Event Trigger Selection Register
ETPS	0x691A	0x695A	0x699A	0x69DA	1/0	Event Trigger Prescale Register
ETFLG	0x691B	0x695B	0x699B	0x69DB	1/0	Event Trigger Flag Register
ETCLR	0x691C	0x695C	0x699C	0x69DC	1/0	Event Trigger Clear Register
ETFRC	0x691D	0x695D	0x699D	0x69DD	1/0	Event Trigger Force Register
PCCTL	0x691E	0x695E	0x699E	0x69DE	1/0	PWM Chopper Control Register
HRCNFG	0x6920	0x6960	0x69A0	0x69E0	1/0	HRPWM Configuration
HRMSTEP	0x6926	-	-	-	1/0	HRPWM MEP Step Register
HRPCTL	0x6928	0x6968	0x69A8	0x69E8	1/0	High-resolution Period Control
TBPRDHRM	0x692A	0x696A	0x69AA	0x69EA	1/W(2)	Time Base Period HRPWM Register Mirror
TBPRDM	0x692B	0x696B	0x69AB	0x69EB	1/W(2)	Time Base Period Register Mirror
CMPAHRM	0x692C	0x696C	0x69AC	0x69EC	1/W(2)	Compare A HRPWM Register Mirror
CMPAM	0x692D	0x696D	0x69AD	0x69ED	1/W(2)	Compare A Register Mirror
DCTRIPSEL	0x6930	0x6970	0x69B0	0x69F0	1/0	Digital Compare Trip Select
DCACTL	0x6931	0x6971	0x69B1	0x69F1	1/0	Digital Compare A Control



名称	ePWM1 地址	ePWM2 地址	ePWM3 地址	ePWM4 地址	大小(×16)/ #SHADOW	寄存器描述
DCBCTL	0x6932	0x6972	0x69B2	0x69F2	1/0	Digital Compare B Control
DCFCTL	0x6933	0x6973	0x69B3	0x69F3	1/0	Digital Compare Filter Control
DCCAPCT	0x6934	0x6974	0x69B4	0x69F4	1/0	Digital Compare Capture Control
DCFOFFSET	0x6935	0x6975	0x69B5	0x69F5	1/1	Digital Compare Filter Offset
DCFOFFSETCNT	0x6936	0x6976	0x69B6	0x69F6	1/0	Digital Compare Filter Offset
DCFWINDOW	0x6937	0x6977	0x69B7	0x69F7	1/0	Digital Compare Filter Window
DCFWINDOWCNT	0x6938	0x6978	0x69B8	0x69F8	1/0	Digital Compare Filter Window
DCCAP	0x6939	0x6979	0x69B9	0x69F9	1/1	Digital Compare Counter

(1) 寄存器受 EALLOW 保护。

(2) W = 写入影子寄存器。



- A. 这些事件是由类型 1 的 ePWM 数字比较 (DC) 子模块基于 COMPxOUT 和 TZ 信号的电平生成的。
- B. 仅带有 eQEP1 模块的器件上才有此信号。

表 5-40 ePWM 子模块及关键内部信号互连

### 5.6.9.1 ePWM 电气数据/时序

#### 5.6.9.1.1 ePWM 时序要求

		最小值	最大值	单位
$t_w(\text{SYCIN})$	同步输入脉宽持续时间			
	异步	$2t_c(\text{SCO})$		周期
	同步	$2t_c(\text{SCO})$		周期
	带输入限定器	$1t_c(\text{SCO}) + t_w(\text{IQSW})$		周期

#### 5.6.9.1.2 ePWM 开关特性

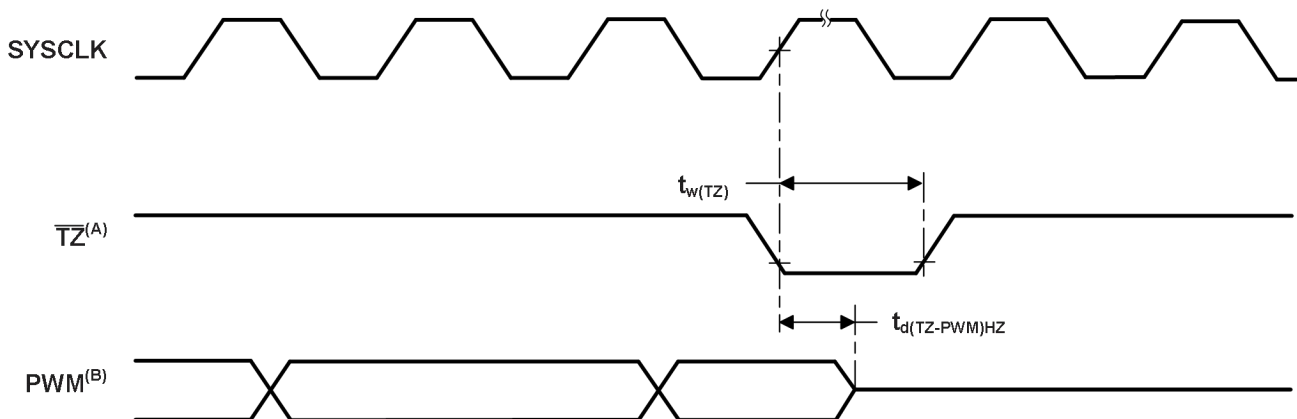
在推荐的操作条件范围内（除非另有说明）

参数	测试条件	最小值	最大值	单位
$t_w(\text{PWM})$	PWMx输出高/低脉宽持续时间	33.33		ns
$t_w(\text{SYNCOUT})$	同步输出脉冲宽度	$8t_c(\text{SCO})$		周期
$t_d(\text{PWM}) t_{za}$	跳闸输入有效到PWM强制高电平的延迟时间		25	ns
	跳闸输入有效到PWM强制低电平的延迟时间	无引脚负载		
$t_d(\text{TZ-PWM})\text{HZ}$	跳闸输入有效到PWM高阻延迟时间		20	ns

### 5.6.9.2 跳闸区输入时序

#### 5.6.9.2.1 跳闸区输入时序要求

		最小值	最大值	单位
$t_w(\text{TZ})$	$\overline{\text{TZx}}$ 低电平输入持续时间			
	异步	$2t_c(\text{TBCLK})$		周期
	同步	$2t_c(\text{TBCLK})$		周期
	带输入限定器	$2t_c(\text{TBCLK}) + t_w(\text{IQSW})$		周期



- A.  $\overline{\text{TZ}}$ :  $\overline{\text{TZ1}}$ 、 $\overline{\text{TZ2}}$ 、 $\overline{\text{TZ3}}$ 、 $\overline{\text{TZ4}}$ 、 $\overline{\text{TZ5}}$ 、 $\overline{\text{TZ6}}$
- B. PWM 是指设备中所有 PWM 引脚。 $\overline{\text{TZ}}$ 变高后的 PWM 引脚状态取决于 PWM 的软件配置。

图 5-41 PWM 高阻抗特性

## 5.6.10 高分辨率 PWM (HRPWM)

该模块将多条延迟线组合在一个模块中，并使用专用校准延迟线简化校准系统。每个 ePWM 模块都有一条高分辨率 (HR) 延迟线。

HRPWM 模块提供的 PWM 分辨率 (时间粒度) 明显优于使用传统数字 PWM 方法所能实现的分辨率。HRPWM 模块的关键点包括：

- 显著扩展了传统数字 PWM 的时间分辨率
- 此功能可用于占空比和相移控制的单边沿以及频率/周期调制的双边沿控制。
- 通过扩展 ePWM 模块的 Compare A 和 Phase 寄存器来实现更精细的时间粒度控制或边沿定位。
- HRPWM 功能在特定器件上有效时，仅在 ePWM 模块的 A 信号路径 (即：EPWMxA 输出) 上提供。EPWMxB 输出仅有传统的 PWM 功能。

---

**注意**

HRPWM 允许的最小 SYSCLKOUT 频率是 60MHz。

---



---

**注意**

当启用双边沿高分辨率 (高分辨率周期模式) 时，PWMxB 通道的输出将有  $\pm 1-2$  个 TBCLK 周期的抖动

---

### 5.6.10.1 HRPWM 电气数据/时序

#### 5.6.10.1.1 HRPWM 特性

参数 <sup>(1)</sup>	最小值	典型值	最大值	单位
微边沿定位 (MEP) 步长 <sup>(2)</sup>		150	310	ps

(1) HRPWM 以 60MHz 的最小 SYSCLKOUT 频率工作。

(2) MEP 步长在高温和最小 V<sub>DD</sub> 电压条件下最大。MEP 步长将随着温度升高和电压降低而增加；并随温度降低和电压提高而减小。使用

HRPWM 功能的应用程序应使用 MEP 比例因子优化器 (SFO) 估算软件功能。SFO 函数有助于在 HRPWM 运行时动态估算每个 SYSCLKOUT 周期的 MEP 步数。

### 5.6.11 增强型捕获模块 (eCAP1)

器件包含一个增强型捕获 (eCAP) 模块，以下是该模块的功能框图。

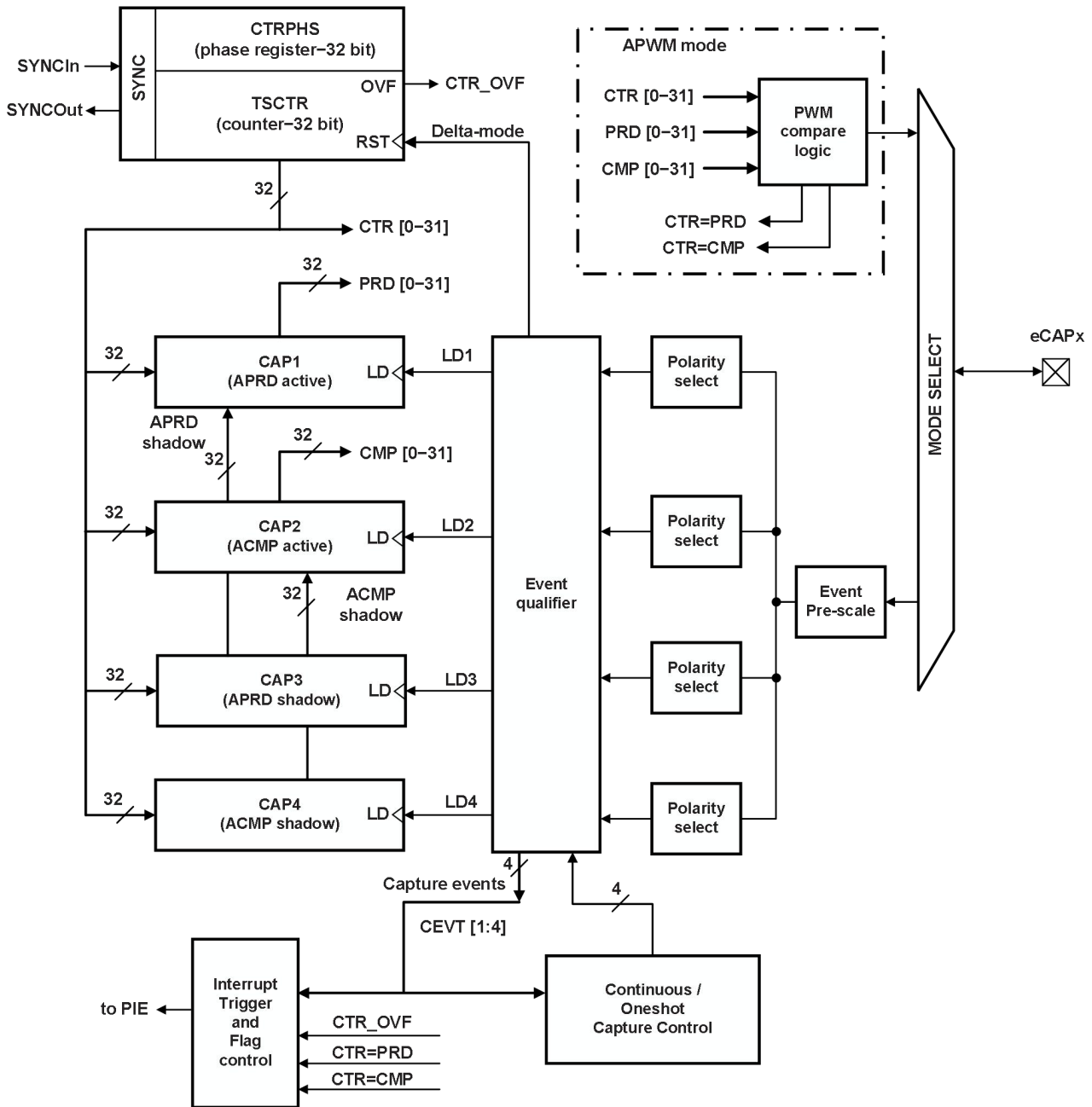


图 5-42 eCAP 功能框图

eCAP 模块使用 SYSCLKOUT 时钟。PCLKCR1 寄存器中的时钟使能位 (ECAP1ENCLK) 用于独立关闭 eCAP 模块 (为了低功耗运行)。复位时, ECAP1ENCLK 被清 0, 外设时钟关闭。

**表 5-28 eCAP 控制和状态寄存器**

名称	eCAP1 地址	eCAP2 地址	eCAP3 地址	SIZE (×16)	EALLOW PROTECTED	寄存器说明
TSCTR	0x6A00	0x6A20	0x6A40	2	No	Timestamp Counter
CTRPHS	0x6A02	0x6A22	0x6A42	2	No	Counter Phase Offset Value Register
CAP1	0x6A04	0x6A24	0x6A44	2	No	Capture 1 Register
CAP2	0x6A06	0x6A26	0x6A46	2	No	Capture 2 Register
CAP3	0x6A08	0x6A28	0x6A48	2	No	Capture 3 Register
CAP4	0x6A0A	0x6A2A	0x6A4A	2	No	Capture 4 Register
Reserved	0x6A0C ~0x6A12	0x6A2C ~0x6A32	0x6A4C ~0x6A52	8	No	Reserved
ECCTL1	0x6A14	0x6A34	0x6A54	1	No	Capture Control Register 1
ECCTL2	0x6A15	0x6A35	0x6A55	1	No	Capture Control Register 2
ECEINT	0x6A16	0x6A36	0x6A56	1	No	Capture Interrupt Enable Register
ECFLG	0x6A17	0x6A37	0x6A57	1	No	Capture Interrupt Flag Register
ECCLR	0x6A18	0x6A38	0x6A58	1	No	Capture Interrupt Clear Register
ECFRC	0x6A19	0x6A39	0x6A59	1	No	Capture Interrupt Force Register
Reserved	0x6A1A ~0x6A1F	0x6A3A ~0x6A3F	0x6A5A ~0x6A5F	6	No	Reserved

### 5.6.11.1 eCAP 电气数据/时序

#### 5.6.11.1.1 eCAP 时序要求

		最小值	最大值	单位
$t_w(\text{CAP})$	异步	$2t_c(\text{SCO})$		周期
	同步	$2t_c(\text{SCO})$		周期
	带输入限定器	$1t_c(\text{SCO}) + t_w(\text{IQSW})$		周期

#### 5.6.11.1.2 eCAP 开关特性

在推荐的操作条件范围内（除非另有说明）

参数	最小值	最大值	单位
$t_w(\text{APWM})$	20		ns

## 5.6.12 高分辨率捕获模块 (HRCAP1~4)

该器件有四个分辨率捕获(HRCAP)模块。高分辨率捕获 (HRCAP) 模块以 300ps 的典型分辨率测量外部脉冲之间的差异。

HRCAP 用途包括:

- 电容式触摸应用
- 脉冲序列周期的高分辨率周期和占空比测量
- 瞬时速度测量
- 瞬时频率测量
- 跨越隔离边界的电压测量
- 距离测量 (声纳) 和扫描

HRCAP 模块的特征包括:

- 在非高分辨率或高分辨率模式下捕获脉冲宽度
- 差分 (增量) 模式脉冲宽度捕获
- 每个边缘上 300ps 分辨率的典型高分辨率捕获
- 下降沿或上升沿中断
- 在两级深度的缓冲区内, 脉冲宽度的连续捕获模式
- 用于精确高分辨率捕获的校准逻辑
- 上述所有资源都专用于单个输入引脚
- HRCAP 校准软件库用于校准和计算分数脉冲宽度

除了高分辨率校准模块外, HRCAP 模块还包括一个捕获通道, 在校准时, 内部连接到最后一个可用的 ePWMxA HRPWM 通道(即: 如果有八个具有 HRPWM 功能的 ePWMs, 则为 HRPWM8A)。

每个 HRCAP 通道都有以下独立的关键资源:

- 专用输入捕获引脚
- 等于 PLL2 输出频率 (与 SYSCLK2 异步) 或 SYSCLKOUT 的 16 位 HRCAP 时钟
- 将高分辨率脉冲宽度捕获在两级深度的缓冲区中

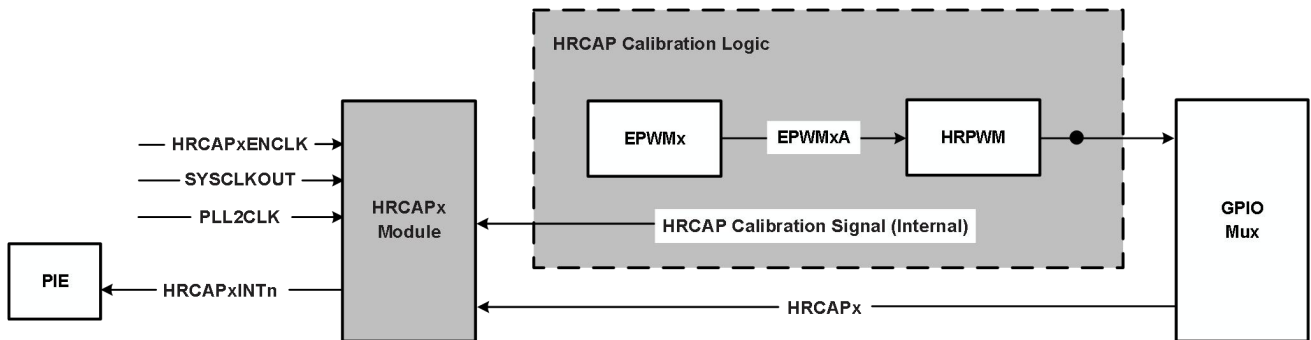


图 5-43 HRCAP 功能框图



**表 5-29 HRCAP 寄存器**

名称	HRCAP1 地址	HRCAP2 地址	HRCAP3 地址	HRCAP4 地址	大小 (×16)	寄存器描述
HCCTL	0x6AC0	0x6AE0	0x6C80	0x6CA0	1	HRCAP Control Register <sup>(1)</sup>
HCIFR	0x6AC1	0x6AE1	0x6C81	0x6CA1	1	HRCAP Interrupt Flag Register
HCICLR	0x6AC2	0x6AE2	0x6C82	0x6CA2	1	HRCAP Interrupt Clear Register <sup>(1)</sup>
HCIFRC	0x6AC3	0x6AE3	0x6C83	0x6CA3	1	HRCAP Interrupt Force Register <sup>(1)</sup>
HCCOUNTER	0x6AC4	0x6AE4	0x6C84	0x6CA4	1	HRCAP 16-bit Counter Register
HCCAPCNTRISE0	0x6AD0	0x6AF0	0x6C90	0x6CB0	1	HRCAP Capture Counter on Rising Edge 0 Register
HCCAPCNTFALLO	0x6AD2	0x6AF2	0x6C92	0x6CB2	1	HRCAP Capture Counter on Falling Edge 0 Register
HCCAPCNTRISE1	0x6AD8	0x6AF8	0x6C98	0x6CB8	1	HRCAP Capture Counter on Rising Edge 1 Register
HCCAPCNTFALL1	0x6ADA	0x6AFA	0x6C9A	0x6CBA	1	HRCAP Capture Counter on Falling Edge 1 Register

(1) 所有寄存器都受 EALLOW 保护。

### 5.6.12.1 HRCAP 电气数据/时序

#### 5.6.12.1.1 高分辨率捕获(HRCAP)时序要求

		最小值	典型值	最大值	单位
$t_c$ (HCCAPCLK)	HRCAP捕获时钟周期	8.333		10.204	ns
$t_w$ (HRCAP)	HRCAP捕获脉冲宽度	$7t_c$ (HCCAPCLK) <sup>(1)</sup>			ns
	HRCAP步长 <sup>(2)</sup>	300			ps

(1) 列出的最小脉冲宽度没有考虑到必须读取所有相关 HCCAP 寄存器以及在脉冲宽度内清除 RISE/FALL 事件标志以确保有效捕获数据的限制。

(2) HRCAP 步长会随着电压降低或温度升高而增加，随着电压升高或温度降低而减小。针对不同的工作条件，在高分辨率模式下使用 HRCAP 的应用程序应使用 HRCAP 校准功能进行动态校准。

### 5.6.13 增强型正交编码器模块 (eQEP1, eQEP2)

该设备最多包含两个增强型正交编码器 (eQEP) 模块。下表列出 eQEP 寄存器的描述。

表 5-30 eQEP 控制和状态寄存器

名称	eQEP1 地址	eQEP2 地址	eQEP1大小(×16) /#SHADOW	寄存器描述
QPOSCNT	0x6B00	0x6B40	2/0	eQEP位置计数器
QPOSINIT	0x6B02	0x6B42	2/0	eQEP初始化位置计数
QPOSMAX	0x6B04	0x6B44	2/0	eQEP最大位置计数
QPOSCMP	0x6B06	0x6B46	2/1	eQEP位置比较
QPOSILAT	0x6B08	0x6B48	2/0	eQEP索引位置锁存
QPOSSLAT	0x6B0A	0x6B4A	2/0	eQEP选通脉冲位置锁存
QPOSLAT	0x6B0C	0x6B4C	2/0	eQEP位置锁存
QUTMR	0x6B0E	0x6B4E	2/0	eQEP单位定时器
QUPRD	0x6B10	0x6B50	2/0	eQEP单位周期寄存器
QWDTMR	0x6B12	0x6B52	1/0	eQEP看门狗定时器
QWDPRD	0x6B13	0x6B53	1/0	eQEP看门狗周期寄存器
QDECCTL	0x6B14	0x6B54	1/0	eQEP解码器控制寄存器
QEPCTL	0x6B15	0x6B55	1/0	eQEP控制寄存器
QCAPCTL	0x6B16	0x6B56	1/0	eQEP捕捉控制寄存器
QPOSCTL	0x6B17	0x6B57	1/0	eQEP位置比较控制寄存器
QEINT	0x6B18	0x6B58	1/0	eQEP中断使能寄存器
QFLG	0x6B19	0x6B59	1/0	eQEP中断标志寄存器
QCLR	0x6B1A	0x6B5A	1/0	eQEP中断清除寄存器
QFRC	0x6B1B	0x6B5B	1/0	eQEP中断强制寄存器
QEPSTS	0x6B1C	0x6B5C	1/0	eQEP状态寄存器
QCTMR	0x6B1D	0x6B5D	1/0	eQEP捕捉定时器
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
Reserved	0x6B21 to 0x6B3F	0x6B61 to 0x6B7F	31/0	保留寄存器

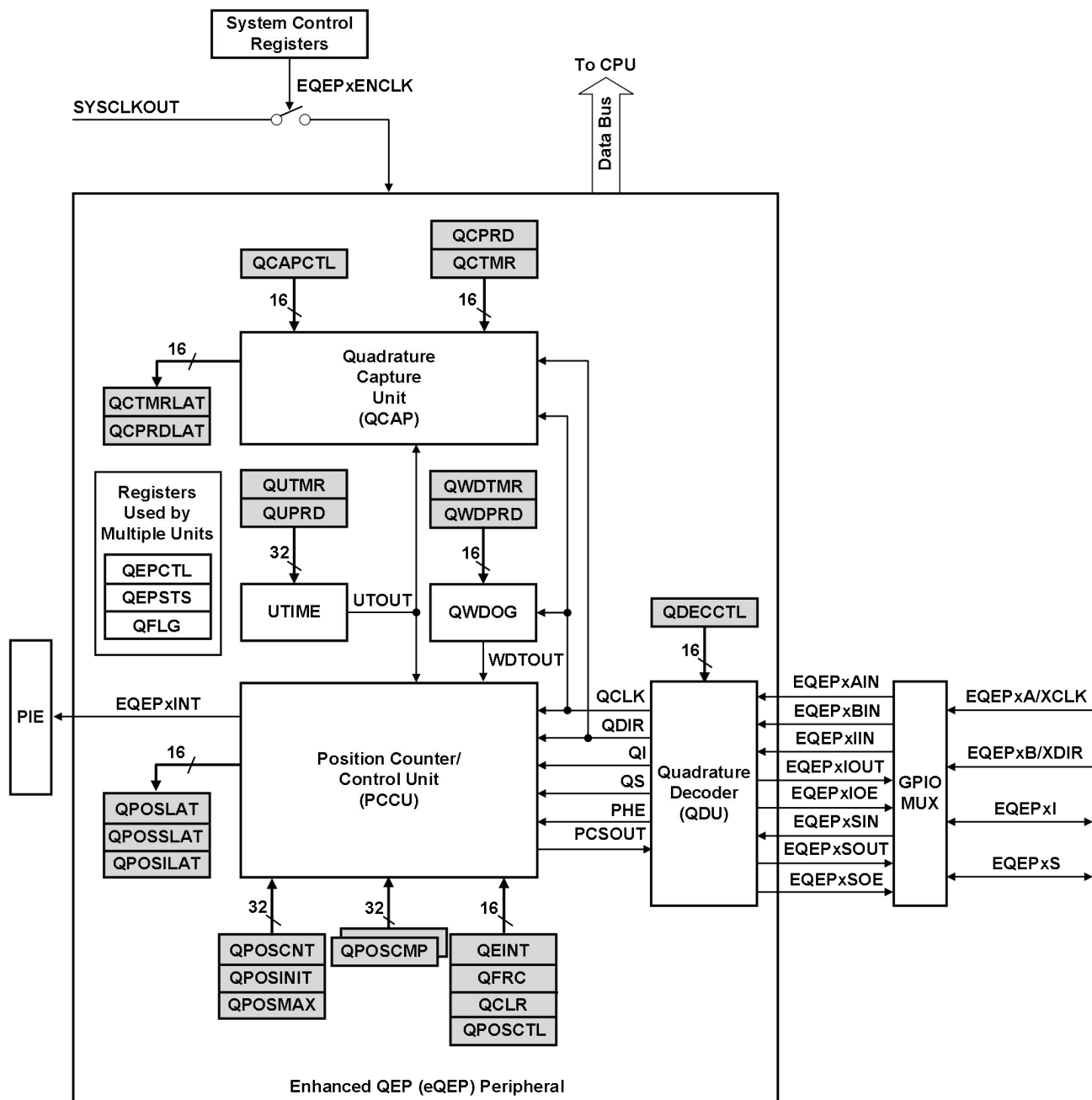


图 5-44 eQEP 功能框图

### 5.6.13.1 eQEP 电气数据/时序

#### 5.6.13.1.1 eQEP 时序要求

		异步/同步	最小值	最大值	单位
$t_w(QEPP)$	QEP 输入周期	异步/同步		$2t_c(SCO)$	周期
		带输入限定		$2[1t_c(SCO) + t_w(IQSW)]$	
$t_w(INDEXH)$	QEP index 输入高电平时间	异步/同步		$2t_c(SCO)$	周期
		带输入限定		$2t_c(SCO) + t_w(IQSW)$	
$t_w(INDEXL)$	QEP index 低电平时间	异步/同步		$2t_c(SCO)$	周期
		带输入限定		$2t_c(SCO) + t_w(IQSW)$	
$t_w(STROBH)$	QEP Strobe 高电平时间	异步/同步		$2t_c(SCO)$	周期
		带输入限定		$2t_c(SCO) + t_w(IQSW)$	
$t_w(STROBL)$	QEP Strobe 低电平时间	异步/同步		$2t_c(SCO)$	周期
		带输入限定		$2t_c(SCO) + t_w(IQSW)$	

#### 5.6.13.1.2 eQEP 开关特性

在推荐的操作条件范围内（除非另有说明）。

参数	描述	最小值	最大值	单位
$t_d(CNTR)_{xin}$	外部时钟来计数增的延迟时间		$4t_c(SCO)$	周期
$t_d(PCS-OUT)_{QEP}$	QEP输入沿到位置比较同步输出延迟时间		$6t_c(SCO)$	周期

### 5.6.14 JTAG 端口

器件上 JTAG 端口被减少到 5 个引脚 ( $\overline{\text{TRST}}$ 、TCK、TDI、TMS、TDO)。TCK、TDI、TMS 和 TDO 引脚同时也是 GPIO 引脚。 $\overline{\text{TRST}}$ 信号在图 4-17 中为引脚选择 JTAG 或者 GPIO 运行模式。在仿真/调试期间, 这些引脚的 GPIO 功能并不可用。如果 GPIO38/TCK/XCLKIN 引脚被用于提供一个外部时钟, 一个替代的内部时钟源应该被用于在仿真/调试期间为器件计时, 这是因为 TCK 功能需要这个引脚。

**注意**

器件中, JTAG 引脚也可被用作 GPIO 引脚。在电路板设计时应小心以确保连接到这些引脚的电路不会影响 JTAG 引脚功能的仿真能力。任一连接到这些引脚的电路不应防止仿真器驱动 JTAG 引脚 (或者被 JTAG 引脚驱动) 进行成功的调试。

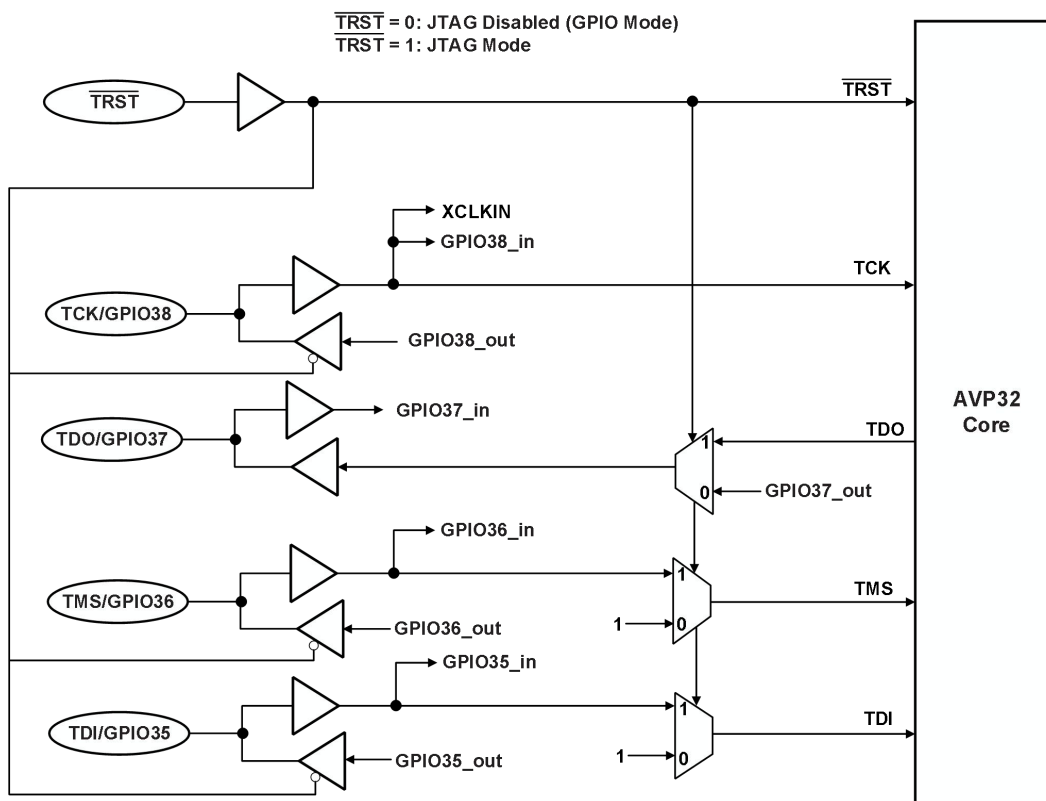


图 5-45 JTAG/GPIO 复用

### 5.6.15 通用输入/输出 (GPIO) 复用器

GPIO 复用器除了按位提供独立的引脚 I/O 功能外, 还可以在单个 GPIO 引脚上复用三个独立的外设信号。该器件支持 45 个 GPIO 引脚, GPIO 控制和数据寄存器被映射到外设帧 1, 以启用对寄存器的 32 位操作 (以及 16 位操作)。以下是 GPIO 寄存器映射表。

表 5-31 GPIO 寄存器

名称	地址	大小 (×16)	说明
GPIO控制寄存器 (受EALLOW保护)			
GPACTRL	0x6F80	2	GPIO A 控制寄存器 (GPIO0 至 31)
GPAQSEL1	0x6F82	2	GPIO A 限定器选择 1 寄存器 (GPIO0 至 15)
GPAQSEL2	0x6F84	2	GPIO A 限定器选择 2 寄存器 (GPIO16 至 31)
GPAMUX1	0x6F86	2	GPIO A MUX 1 寄存器 (GPIO0 至 15)
GPAMUX2	0x6F88	2	GPIO A MUX 2 寄存器 (GPIO16 至 31)
GPADIR	0x6F8A	2	GPIO A 方向寄存器 (GPIO0 至 31)
GPAPUD	0x6F8C	2	GPIO A 上拉禁用寄存器 (GPIO0 至 31)
GPBCTRL	0x6F90	2	GPIO B 控制寄存器 (GPIO32 至 44)
GPBQSEL1	0x6F92	2	GPIO B 限定器选择 1 寄存器 (GPIO32 至 44)
GPBQSEL2	0x6F94	2	GPIO B 限定器选择 2 寄存器
GPBMUX1	0x6F96	2	GPIO B MUX 1 寄存器 (GPIO32 至 44)
GPBMUX2	0x6F98	2	GPIO B MUX 2 寄存器 (GPIO50 至 58)
GPBDIR	0x6F9A	2	GPIO B 方向寄存器 (GPIO32 至 44)
GPBPUD	0x6F9C	2	GPIO B 上拉禁用寄存器 (GPIO32 至 44)
AIOMUX1	0x6FB6	2	模拟, I/O 复用 1 寄存器 (AI00 至 AI015)
AIODIR	0x6FBA	2	模拟, I/O 方向寄存器 (AI00 至 AI015)
GPIO数据寄存器 (不受EALLOW保护)			
GPADAT	0x6FC0	2	GPIO A 数据寄存器 (GPIO0 至 31)
GPASET	0x6FC2	2	GPIO A 数据设定寄存器 (GPIO0 至 31)
GPACLEAR	0x6FC4	2	GPIO A 数据清除寄存器 (GPIO0 至 31)
GPATOGGLE	0x6FC6	2	GPIO A 数据切换寄存器 (GPIO0 至 31)
GPBDAT	0x6FC8	2	GPIO B 数据寄存器 (GPIO32 至 44)
GPBSET	0x6FCA	2	GPIO B 数据设定寄存器 (GPIO32 至 44)
GPBCLEAR	0x6FCC	2	GPIO B 数据清除寄存器 (GPIO32 至 44)
GPBTOGGLE	0x6FCE	2	GPIO B 数据切换寄存器 (GPIO32 至 44)
AIODAT	0x6FD8	2	模拟 I/O 数据寄存器 (AI00 至 AI015)
AIOSET	0x6FDA	2	模拟 I/O 数据设定寄存器 (AI00 至 AI015)
AIOCLEAR	0x6FDC	2	模拟 I/O 数据清除寄存器 (AI00 至 AI015)
AIOTOGGLE	0x6FDE	2	模拟 I/O 数据切换寄存器 (AI00 至 AI015)
GPIO中断和低功耗模式选择寄存器 (受EALLOW保护)			
GPIOXINT1SEL	0x6FE0	1	XINT1 GPIO 输入选择寄存器 (GPIO0 至 31)

名称	地址	大小 (×16)	说明
GPIOXINT2SEL	0x6FE1	1	XINT2 GPIO 输入选择寄存器 (GPIO0 至 31)
GPIOXINT3SEL	0x6FE2	1	XINT3 GPIO 输入选择寄存器 (GPIO0 至 31)
GPIO_LPMSEL	0x6FE8	2	LPM GPIO 选择寄存器 (GPIO0 至 31)

**注意**

从写入 GPxMUXn/AIOMUXn 和 GPxQSELn 寄存器到动作有效有两个 SYSCLKOUT 周期延迟。

**表 5-32 GPIOA 复用器**

	复位默认值	外设选择1	外设选择2	外设选择3
GPAMUX1 REGISTER BITS	(GPAMUX1 BITS = 00)	(GPAMUX1 BITS = 01)	(GPAMUX1 BITS = 10)	(GPAMUX1 BITS = 11)
1-0	GPIO0	EPWM1A (0)	Reserved	Reserved
3-2	GPIO1	EPWM1B (0)	Reserved	COMP1OUT (0)
5-4	GPIO2	EPWM2A (0)	Reserved	Reserved
7-6	GPIO3	EPWM2B (0)	SPISOMIA (I/O)	COMP2OUT (0)
9-8	GPIO4	EPWM3A (0)	Reserved	Reserved
11-10	GPIO5	EPWM3B (0)	SPISIMOA (I/O)	ECAP1 (I/O)
13-12	GPIO6	EPWM4A (0)	EPWMSYNCI (I)	EPWMSYNCO (0)
15-14	GPIO7	EPWM4B (0)	SCIRXDA (I)	ECAP2 (I/O)
17-16	GPIO8	EPWM5A (0)	Reserved	ADCSOCAO (0)
19-18	GPIO9	EPWM5B (0)	SCITXDB (0)	ECAP3 (I/O)
21-20	GPIO10	EPWM6A (0)	Reserved	ADCSOCBO (0)
23-22	GPIO11	EPWM6B (0)	SCIRXDB (I)	ECAP1 (I/O)
25-24	GPIO12	TZ1 (I)	SCITXDA (0)	SPISIMOB (I/O)
27-26	GPIO13	TZ2 (I)	Reserved	SPISOMIB (I/O)
29-28	GPIO14	TZ3 (I)	SCITXDB (0)	SPICLKB (I/O)
31-30	GPIO15	ECAP2 (I/O)	SCIRXDB (I)	SPISTEB (I/O)
GPAMUX2 REGISTER BITS	(GPAMUX2 BITS = 00)	(GPAMUX2 BITS = 01)	(GPAMUX2 BITS = 10)	(GPAMUX2 BITS = 11)
1-0	GPIO16	SPISIMOA (I/O)	Reserved	TZ2 (I)
3-2	GPIO17	SPISOMIA (I/O)	Reserved	TZ3 (I)
5-4	GPIO18	SPICLKA (I/O)	SCITXDB (0)	XCLKOUT (0)
7-6	GPIO19/XCLKIN	SPISTEA (I/O)	SCIRXDB (I)	ECAP1 (I/O)
9-8	GPIO20	EQEP1A (I)	MDXA (0)	COMP1OUT (0)
11-10	GPIO21	EQEP1B (I)	MDRA (I)	COMP2OUT (0)
13-12	GPIO22	EQEP1S (I/O)	MCLKXA (I/O)	SCITXDB (0)
15-14	GPIO23	EQEP1I (I/O)	MFSXA (I/O)	SCIRXDB (I)
17-16	GPIO24	ECAP1 (I/O)	EQEP2A <sup>(3)</sup> (I)	SPISIMOB (I/O)
19-18	GPIO25	ECAP2 (I/O)	EQEP2B <sup>(3)</sup> (I)	SPISOMIB (I/O)
21-20	GPIO26 <sup>(4)</sup>	ECAP3 (I/O)	EQEP2I <sup>(3)</sup> (I/O)	SPICLKB (I/O)
23-22	GPIO27 <sup>(4)</sup>	HRCAP2 (I)	EQEP2S <sup>(3)</sup> (I/O)	SPISTEB (I/O)
25-24	GPIO28	SCIRXDA (I)	SDAA (I/OD)	TZ2 (I)

	复位默认值	外设选择1	外设选择2	外设选择3
27-26	GPI029	SCITXDA (O)	SCLA (I/OD)	TZ3 (I)
29-28	GPI030	CANRXA (I)	EQEP2I <sup>(3)</sup> (I/O)	EPWM7A (O)
31-30	GPI031	CANTXA (O)	EQEP2S <sup>(3)</sup> (I/O)	EPWM8A (O)

(1) “保留”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。

(2) I = 输入，O = 输出，OD = 开漏。

(3) 在 80 脚封装芯片上没有引出 eQEP2 外设引脚。

(4) 要启用 GPIO26 (USB0DP) 和 GPIO27 (USB0DM) 上的 USB 功能，设置 GPACTRL2 寄存器中的 USBIOEN 位。根据您的 USB 应用，可能需要额外的引脚来保持符合 USB 2.0 规范。

**表 5-33 GPIOB 复用器**

	复位默认值	外设选择1	外设选择2	外设选择3
GPBMUX1 REGISTER BITS	(GPBMUX1 BITS = 00)	(GPBMUX1 BITS = 01)	(GPBMUX1 BITS = 10)	(GPBMUX1 BITS = 11)
1-0	GPI032	SDAA (I/OD)	EPWMSYNCI (I)	ADCSOCAA (O)
3-2	GPI033	SCLA (I/OD)	EPWMSYNCO (O)	ADCSOCBA (O)
5-4	GPI034	COMP2OUT (O)	Reserved	COMP3OUT (O)
7-6	GPI035 (TDI)	Reserved	Reserved	Reserved
9-8	GPI036 (TMS)	Reserved	Reserved	Reserved
11-10	GPI037 (TDO)	Reserved	Reserved	Reserved
13-12	GPI038/XCLKIN (TCK)	Reserved	Reserved	Reserved
15-14	GPI039	Reserved	Reserved	Reserved
17-16	GPI040 (3)	EPWM7A (O)	SCITXDB (O)	Reserved
19-18	GPI041 (3)	EPWM7B (O)	SCIRXDB (I)	Reserved
21-20	GPI042 (3)	EPWM8A (O)	TZ1 (I)	COMP1OUT (O)
23-22	GPI043 (3)	EPWM8B (O)	TZ2 (I)	COMP2OUT (O)
25-24	GPI044 (3)	MFSRA (I/O)	SCIRXDB (I)	EPWM7B (O)
27-26	Reserved	Reserved	Reserved	Reserved
29-28	Reserved	Reserved	Reserved	Reserved
31-30	Reserved	Reserved	Reserved	Reserved
GPBMUX2 REGISTER BITS	(GPBMUX2 BITS = 00)	(GPBMUX2 BITS = 01)	(GPBMUX2 BITS = 10)	(GPBMUX2 BITS = 11)
1-0	Reserved	Reserved	Reserved	Reserved
3-2	Reserved	Reserved	Reserved	Reserved
5-4	GPI050 (3)	EQEP1A (I)	MDXA (O)	TZ1 (I)
7-6	GPI051 (3)	EQEP1B (I)	MDRA (I)	TZ2 (I)
9-8	GPI052 (3)	EQEP1S (I/O)	MCLKXA (I/O)	TZ3 (I)
11-10	GPI053 (3)	EQEP1I (I/O)	MFSXA (I/O)	Reserved
13-12	GPI054 (3)	SPISIMOA (I/O)	EQEP2A (I)	HRCAP1 (I)
15-14	GPI055 (3)	SPISOMIA (I/O)	EQEP2B (I)	HRCAP2 (I)
17-16	GPI056 (3)	SPICLKA (I/O)	EQEP2I (I/O)	HRCAP3 (I)
19-18	GPI057 (3)	SPISTEA (I/O)	EQEP2S (I/O)	HRCAP4 (I)



	复位默认值	外设选择1	外设选择2	外设选择3
21-20	GPI058 (3)	MCLKRA (I/O)	SCITXDB (O)	EPWM7A (O)
23-22	Reserved	Reserved	Reserved	Reserved
25-24	Reserved	Reserved	Reserved	Reserved
27-26	Reserved	Reserved	Reserved	Reserved
29-28	Reserved	Reserved	Reserved	Reserved
31-30	Reserved	Reserved	Reserved	Reserved

(1) “Reserved” 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。

(2) I = 输入, O = 输出, OD = 开漏。

(3) 在 80 引脚 LQFP 封装内这个引脚不可用。

**表 5-34 100 脚封装的 Analog 复用器**

	AIOx或外设选择1	外设选择2或3 (缺省值)
AIOMUX1 REGISTER BITS	AIOMUX1 BITS = 0, x	AIOMUX1 BITS = 1, x
1-0	ADCINA0 (I)	ADCINA0 (I)
3-2	ADCINA1 (I)	ADCINA1 (I)
5-4	AIO2 (I/O)	ADCINA2 (I), COMP1A (I)
7-6	ADCINA3 (I)	ADCINA3 (I)
9-8	AIO4 (I/O)	ADCINA4 (I), COMP2A (I)
11-10	ADCINA5 (I)	ADCINA5 (I)
13-12	AIO6 (I/O)	ADCINA6 (I), COMP3A (I)
15-14	ADCINA7 (I)	ADCINA7 (I)
17-16	ADCINB0 (I)	ADCINB0 (I)
19-18	ADCINB1 (I)	ADCINB1 (I)
21-20	AIO10 (I/O)	ADCINB2 (I), COMP1B (I)
23-22	ADCINB3 (I)	ADCINB3 (I)
25-24	AIO12 (I/O)	ADCINB4 (I), COMP2B (I)
27-26	ADCINB5 (I)	ADCINB5 (I)
29-28	AIO14 (I/O)	ADCINB6 (I), COMP3B (I)
31-30	ADCINB7 (I)	ADCINB7 (I)

(1) I = 输入, O = 输出。

**表 5-35 80 脚封装的 Analog 复用器**

	AIOx或外设选择1	外设选择2或3 (缺省值)
AIOMUX1 REGISTER BITS	AIOMUX1 BITS = 0, x	AIOMUX1 BITS = 1, x
1-0	ADCINA0 (I), VREFHI (I)	ADCINA0 (I), VREFHI (I)
3-2	ADCINA1 (I)	ADCINA1 (I)
5-4	AIO2 (I/O)	ADCINA2 (I), COMP1A (I)
7-6	-	-
9-8	AIO4 (I/O)	ADCINA4 (I), COMP2A (I)

	AIO <sub>x</sub> 或外设选择1	外设选择2或3 (缺省值)
AIOMUX1 REGISTER BITS	AIOMUX1 BITS = 0, x	AIOMUX1 BITS = 1, x
11-10	ADCINA5 (I)	ADCINA5 (I)
13-12	AIO6 (I/O)	ADCINA6 (I), COMP3A (I)
15-14	-	-
17-16	ADCINB0 (I)	ADCINB0 (I)
19-18	ADCINB1 (I)	ADCINB1 (I)
21-20	AIO10 (I/O)	ADCINB2 (I), COMP1B (I)
23-22	-	-
25-24	AIO12 (I/O)	ADCINB4 (I), COMP2B (I)
27-26	ADCINB5 (I)	ADCINB5 (I)
29-28	AIO14 (I/O)	ADCINB6 (I), COMP3B (I)
31-30	-	-

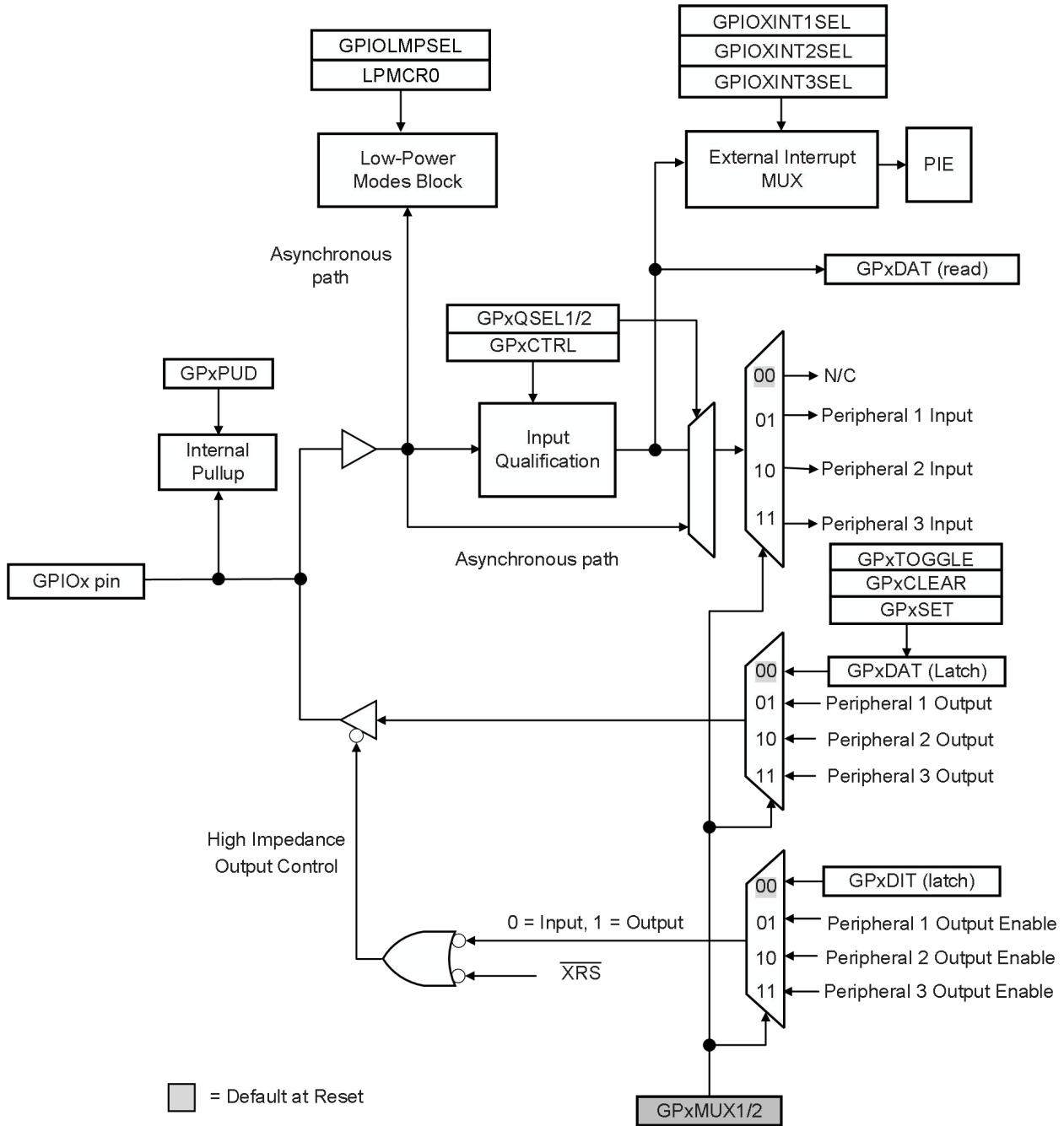
(1) I = 输入, O = 输出。

用户可以通过 GPxQSEL1/2 寄存器从以下四个选项中选择每个 GPIO 引脚的输入限定类型：

- (GPxQSEL1/2 = 00) 仅同步到 SYSCLKOUT：这是所有 GPIO 引脚在复位时的默认值，它只是将输入信号同步到系统时钟 (SYSCLKOUT)。
- (GPxQSEL1/2 = 01 或 10) 使用采样窗口的限定条件：在此模式下，输入信号在与系统时钟 (SYSCLKOUT) 同步后，通过指定的周期数进行鉴定，然后才允许输入改变。
- 采样周期由 GPxCTRL 寄存器中的 QUALPRD 位指定，对一组 8 个信号进行配置。采样周期指定用于采样输入信号的 SYSCLKOUT 周期的倍数。采样窗口是 3 或 6 个样本宽，只有当所有样本都相同 (全 0 或全 1) 时，输出才会改变。
- (GPxQSEL1/2 = 11) 无同步：该模式用于不需要同步的外设 (在外设内执行同步)。

由于器件上所要求的多级复用，有可能会出现一个外设输入信号被映射到多个 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 0 或 1 状态，具体取决于外设。



- A. x代表端口A或B。例如，GPxDIR是指GPADIR和GPBDIR寄存器，这取决于所选择的特定GPIO引脚。
- B. GPxDAT的锁存/读取访问存储器同一位置。
- C. 这是一个通用的GPIO复用器框图。并不是所有选项都可用于所有GPIO引脚。

图 5-46 GPIO 复用

### 5.6.15.1 GPIO 电气数据/时序

#### 5.6.15.1.1 GPIO 输出时序

##### 5.6.15.1.1.1 通用输出开关特性

在推荐的操作条件范围内（除非另有说明）。

参数		最小值	最大值	单位
$t_r$ (GPO)	GPIO从低电平切换至高电平的上升时间		13(1)	ns
$t_f$ (GPO)	GPIO 从高电平切换至低电平的下降时间		13(1)	
$f_{GPO}$	翻转频率		22.5	MHz

(1) 上升时间和下降时间随着 I/O 引脚上的电气负载变化，给出的值适用于 I/O 引脚上 40pF 负载。

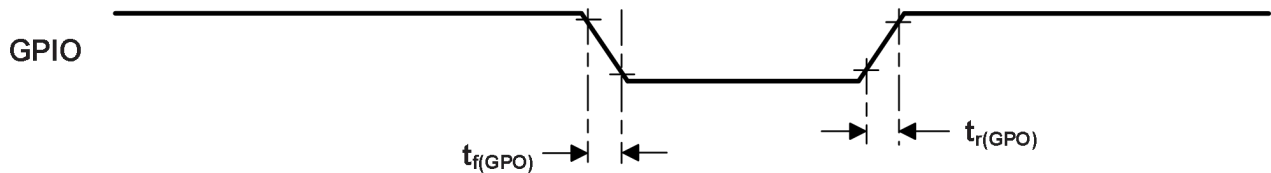


图 5-47 通用输出时序

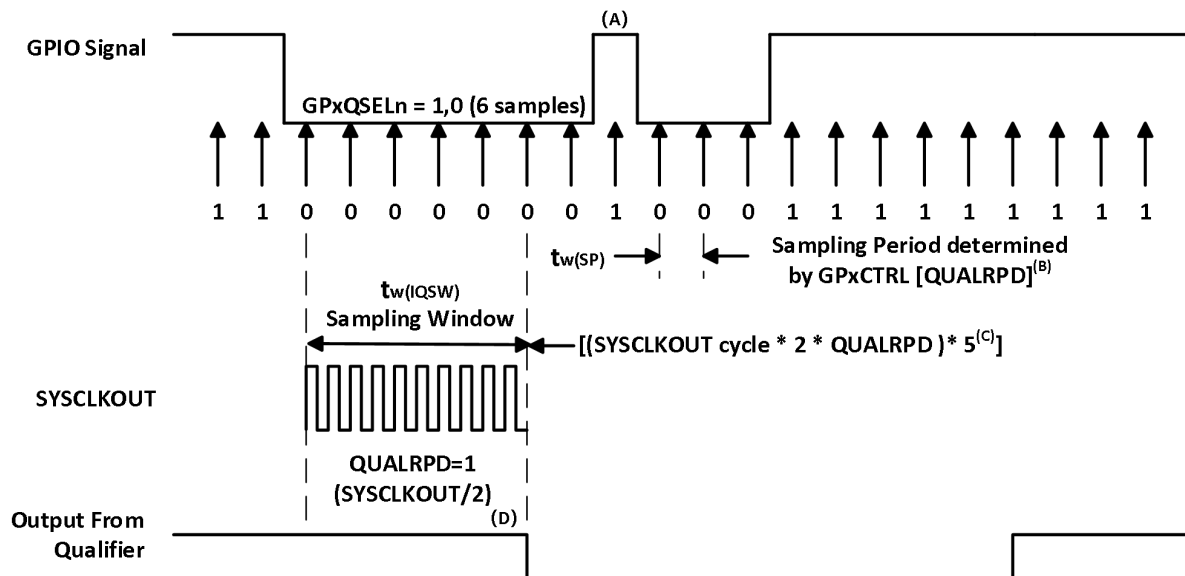
#### 5.6.15.1.2 GPIO 输入时序

##### 5.6.15.1.2.1 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	$2t_c(SCO)*QUALPRD$	
$t_w$ (IQSW)	输入限定器采样窗口		$t_w(SP)*(n^{(1)}-1)$	周期
$t_w$ (GPI) <sup>(2)</sup>	GPIO 低电平/高电平脉冲持续时间	同步模式	$2t_c(SCO)$	周期
		有输入限定器	$t_w(IQSW)+t_w(SP)+1t_c(SCO)$	

(1) "n"代表由 GPxQSELn 寄存器定义的限定采样数量。

(2) 对于  $t_w$ (GPI)，低电平有效信号的脉宽测量是从  $V_{IL}$  至  $V_{IL}$ ，高电平有效信号脉宽则为  $V_{IH}$  至  $V_{IH}$ 。



- A. 这个毛刺将被输入限定器所忽略。QUALPRD 位字段指定了限定采样周期，值可在 0x00 至 0xFF 间变化。如果 QUALPRD=00，那么采样周期为 1 个 SYSCLKOUT 周期。对于任何其它的“n”值，限定采样周期为 2n SYSCLKOUT 周期（即每 2n SYSCLKOUT 周期对 GPIO 引脚进行采样）。
- 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  $(5 \times \text{QUALPRD} \times 2)$  SYSCLKOUT 周期内保持稳定，这将确保出现 5 个采样周期用于检测。由于外部信号是异步驱动的，13 个 SYSCLKOUT 宽的脉冲可确保可靠识别。

图 5-48 采样模式

### 5.6.15.1.2.2 输入信号的采样窗口宽度

以下内容总结了：各种输入限定器配置下，输入信号的采样窗口宽度。

Sampling frequency 是指相对于 SYSCLKOUT 的信号采样频率。

如果  $\text{QUALPRD} \neq 0$ , Sampling frequency =  $\text{SYSCLKOUT} / (2 \times \text{QUALPRD})$

如果  $\text{QUALPRD} = 0$ , Sampling frequency = SYSCLKOUT

如果  $\text{QUALPRD} \neq 0$ , Sampling period =  $\text{SYSCLKOUT cycle} \times 2 \times \text{QUALPRD}$

在上面的等式中，SYSCLKOUT cycle 是指 SYSCLKOUT 的周期。

如果  $\text{QUALPRD} = 0$ , Sampling period = SYSCLKOUT cycle

在一个指定的采样窗口中，采样输入信号的 3 或 6 个样本以确定信号的有效性，这个值由 GPxQSELn 寄存器的值确定。

**例 1:**

使用 3 个样本的限定

如果 QUALPRD ≠ 0, 采样窗口宽度 = (SYSCLKOUT cycle × 2 × QUALPRD) × 2

如果 QUALPRD = 0, 采样窗口宽度 = (SYSCLKOUT cycle) × 2

**例 2:**

使用 6 个样本的限定

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

如果 QUALPRD = 0, 采样窗口宽度 = (SYSCLKOUT cycle) × 5

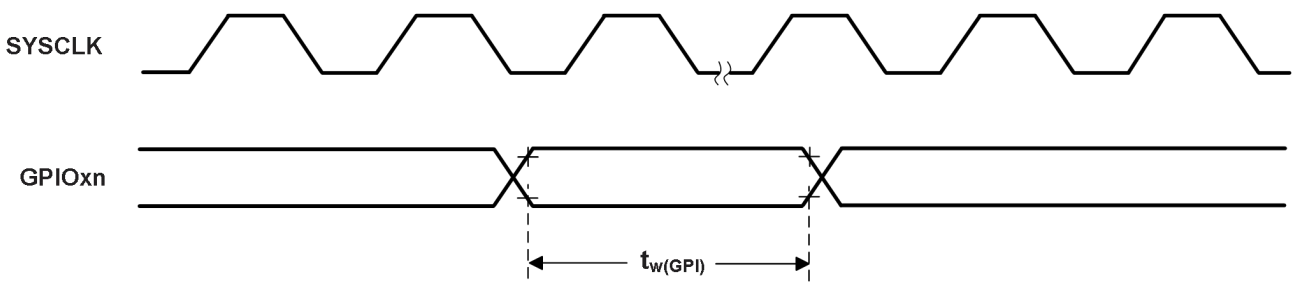


图 5-49 通用输入时序

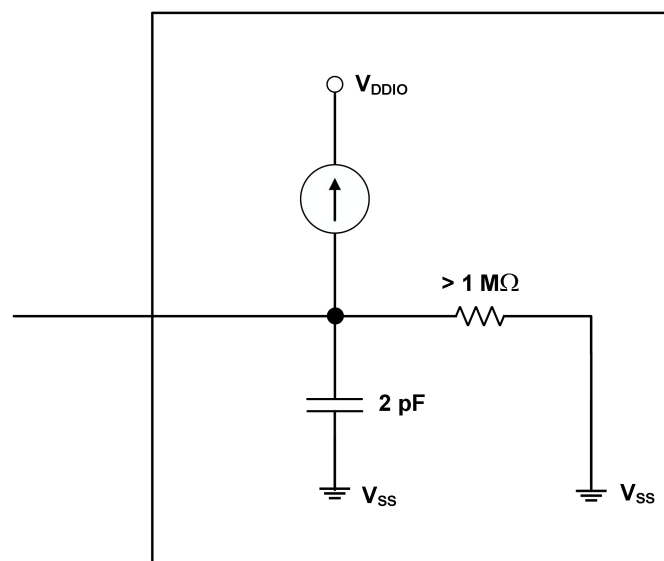


图 5-50 内部带上拉的 GPIO 引脚输入阻抗模型

### 5.6.15.1.3 低功耗模式唤醒时间

#### 5.6.15.1.3.1 IDLE模式时序要求

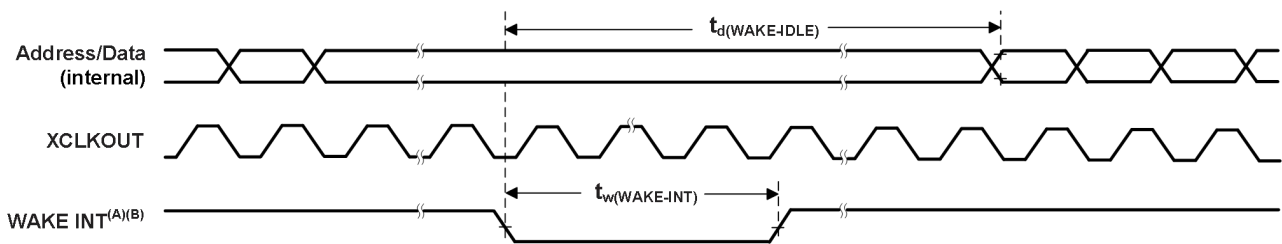
		最小值	最大值	单位
$t_w(\text{WAKE-INT})$	脉冲持续时间, 外部唤醒信号的时间	无输入限定器	$2t_c(\text{SCO})$	周期
		有输入限定器	$5t_c(\text{SCO}) + t_w(\text{IQSW})$	

#### 5.6.15.1.3.2 IDLE模式开关特性

在推荐的操作条件范围内 (除非另有说明)。

参数	测试条件	最小值	最大值	单位
$t_d(\text{WAKE-IDLE})$	延迟时间, 外部唤醒信号到程序恢复执行(1)			周期
	<ul style="list-style-type: none"> <li>从Flash唤醒</li> <li>—Flash处于active状态</li> </ul>	无输入限定器	$20t_c(\text{SCO})$	周期
		有输入限定器	$20t_c(\text{SCO}) + t_w(\text{IQSW})$	
	<ul style="list-style-type: none"> <li>从Flash唤醒</li> <li>—Flash处于sleep状态</li> </ul>	无输入限定器	$1050t_c(\text{SCO})$	周期
		有输入限定器	$1050t_c(\text{SCO}) + t_w(\text{IQSW})$	
	<ul style="list-style-type: none"> <li>从SARAM中唤醒</li> </ul>	无输入限定器	$20t_c(\text{SCO})$	周期
有输入限定器		$20t_c(\text{SCO}) + t_w(\text{IQSW})$		

(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 (由唤醒信号触发) 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 $\text{XRS}$ 。IDLE 指令被执行后, 在唤醒信号生效前需要 5 个 OSCCLK 周期 (最小值) 的延迟。

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

图 5-51 进入和退出 IDLE 模式的时序

#### 5.6.15.1.3.3 STANDBY模式时序要求

		最小值	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
		有输入限定器 (1)	$(2+\text{QUALSTDBY}) * t_c(\text{OSCCLK})$	

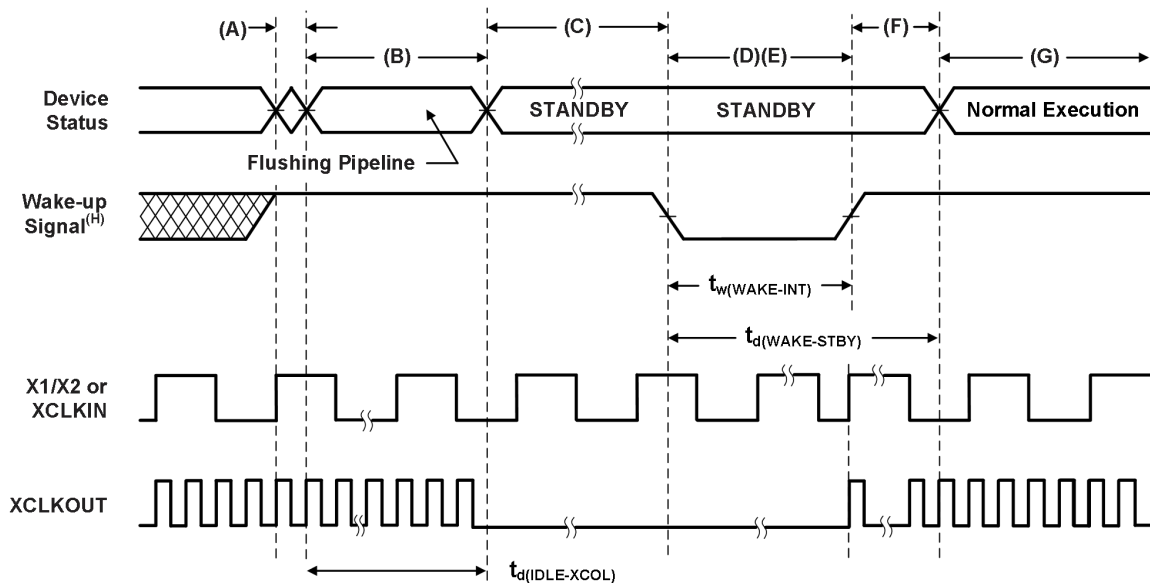
(1) QUALSTDBY 是 LPMCR0 寄存器内的 6 位字段。

#### 5.6.15.1.3.4 STANDBY模式开关特性

在推荐的操作条件范围内（除非另有说明）。

参数	测试条件	最小值	最大值	单位
$t_d(\text{IDLE-XCOL})$	延时时间，IDLE指令执行到XCLKOUT低电平	$32t_c(\text{SCO})$	$45t_c(\text{SCO})$	周期
$t_d(\text{WAKE-STBY})$	延迟时间，外部唤醒信号到程序恢复执行 <sup>(1)</sup>			周期
	● 从Flash唤醒 —Flash处于active状态	无输入限定器	$100t_c(\text{SCO})$	周期
		有输入限定器	$100t_c(\text{SCO}) + t_w(\text{WAKE-INT})$	
	● 从Flash唤醒 —Flash处于sleep状态	无输入限定器	$1125t_c(\text{SCO})$	周期
		有输入限定器	$1125t_c(\text{SCO}) + t_w(\text{WAKE-INT})$	
● 从SARAM中唤醒	无输入限定器	$100t_c(\text{SCO})$	周期	
	有输入限定器	$100t_c(\text{SCO}) + t_w(\text{WAKE-INT})$		

(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（由唤醒信号触发）的执行会涉及额外的延迟。



- A. 器件执行 IDLE 指令进入 STANDBY 模式。
- B. PLL 响应 STANDBY 信号。SYSCLKOUT 在关闭之前保持输出如下所示的周期数：
  - 当 DIVSEL = 00 或 01 时，16 个周期
  - 当 DIVSEL = 10 时，32 个周期
  - 当 DIVSEL = 11 时，64 个周期
 这个延时使得 CPU 流水线和其它等待的操作被适当清空。
- C. 外设时钟关闭，但 PLL 和看门狗并未关闭。器件处于 STANDBY 模式。IDLE 指令被执行后，在唤醒信号生效前最小需要 5 个 OSCCLK 周期的延迟。
- D. 外部唤醒信号被激活。
- E. 给到 GPIO 引脚的唤醒信号必须满足最小脉宽要求。此外，该信号必须无毛刺。如果噪声信号被送到 GPIO 引脚，器件的唤醒行为将不确定，并且在随后的唤醒脉冲中，也可能不会退出低功耗模式。
- F. 在延迟一个周期后，退出 STANDBY 模式。
- G. 恢复正常执行，器件将响应中断（如果被启用）。
- H. 执行 IDLE 指令，器件进入低功耗模式（LPM）后，至少四个 OSCCLK 周期内，不应启动唤醒。

图 5-52 进入和退出 STANDBY 模式的时序



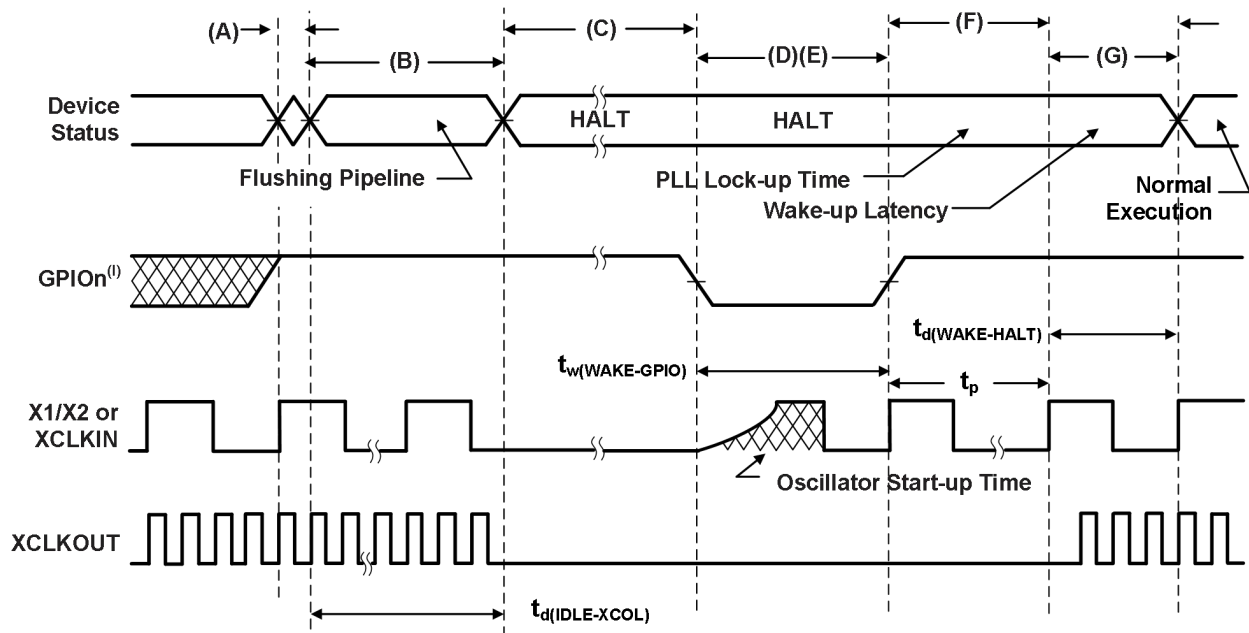
**5.6.15.1.3.5 HALT模式时序要求**

		最小值	最大值	单位
$t_w(\text{WAKE-GPIO})$	GPIO 唤醒脉冲持续时间	$t_{\text{OSCST}} + 2t_c(\text{OSCCLK})$		周期
$t_w(\text{WAKE-XRS})$	$\overline{\text{XRS}}$ 唤醒脉冲持续时间	$t_{\text{OSCST}} + 8t_c(\text{OSCCLK})$		

**5.6.15.1.3.6 HALT模式开关特性**

在推荐的操作条件范围内（除非另有说明）。

参数		最小值	最大值	单位
$t_d(\text{IDLE-XCOL})$	延时时间，IDLE指令执行到XCLKOUT低电平	$32t_c(\text{SCO})$	$45t_c(\text{SCO})$	周期
$t_p$	PLL锁定时间	1		ms
$t_d(\text{WAKE-HALT})$	延迟时间，PLL锁定到程序恢复执行	$1125t_c(\text{SCO})$		周期
	<ul style="list-style-type: none"> <li>● 从Flash唤醒</li> <li>—Flash处于sleep状态</li> </ul>			
	<ul style="list-style-type: none"> <li>● 从SARAM中唤醒</li> </ul>	$35t_c(\text{SCO})$		周期



- A. 器件执行 IDLE 指令进入 HALT 模式。
- B. PLL 响应 HALT 信号。SYSCLKOUT 在关闭振荡器且停止内核输入时钟 CLKIN 之前保持输出如下所示的周期数：
  - 当 DIVSEL = 00 或 01 时，16 个周期
  - 当 DIVSEL = 10 时，32 个周期
  - 当 DIVSEL = 11 时，64 个周期
 这个延时使得 CPU 流水线和其它等待的操作被适当清空。
- C. 关闭外设时钟和 PLL。如果使用石英晶体或陶瓷谐振器作为时钟源，内部振荡器也会关闭。该设备现在处于 HALT 模式，绝对是最小功耗。通过写入 CLKCTL 寄存器中的适当位，可以在 HALT 模式下保持零引脚内部振荡器 (INTOSC1 和 INTOSC2) 和看门狗活动。IDLE 指令被执行后，在唤醒信号生效前最小需要 5 个 OSCCLK 周期的延迟。
- D. 当 GPIO<sup>n</sup> 引脚 (用于使设备脱离 HALT) 被驱动为低电平时，振荡器被打开，振荡器唤醒序列被启动。只有在振荡器稳定后，GPIO 引脚才应被驱动为高电平。这使得能够在 PLL 锁定序列期间提供干净的时钟信号。由于 GPIO 引脚的下降沿异步启动唤醒程序，因此在进入 HALT 模式之前和期间应注意保持低噪声环境。
- E. 给到 GPIO 引脚的唤醒信号必须满足最小脉宽要求。此外，该信号必须无毛刺。如果噪声信号被送到 GPIO 引脚，器件的唤醒行为将不确定，并且在随后的唤醒脉冲中，也可能不会退出低功耗模式。
- F. 一旦振荡器稳定，PLL 锁定序列启动需要 1 毫秒。
- G. 当输入内核的 CLKIN 被启用时，设备将在延迟后响应中断 (如果启用)，现在退出 HALT 模式。
- H. 恢复正常执行。
- I. 执行 IDLE 指令，器件进入低功耗模式 (LPM) 后，至少四个 OSCCLK 周期内，不应启动唤醒。

图 5-53 使用 GPIO<sup>n</sup> 唤醒 HALT 模式

## 5.6.16 通用串行总线 (USB)

### 5.6.16.1 USB 电气数据/时序

#### 5.6.16.1.1 USB 输入端口 DP 和 DM 时序要求

		V <sub>CC</sub>	最小值	最大值	单位
V <sub>(CM)</sub>	差分输入共模范围		0.8	2.5	V
Z <sub>(IN)</sub>	输入阻抗		300		kΩ
VC <sub>RS</sub>	交叉电压		1.3	2.0	V
V <sub>IL</sub>	静态 SE 低电平输入逻辑		0.8		V
V <sub>IH</sub>	静态 SE 高电平输入逻辑			2.0	V
V <sub>DI</sub>	差分输入电压			0.2	V

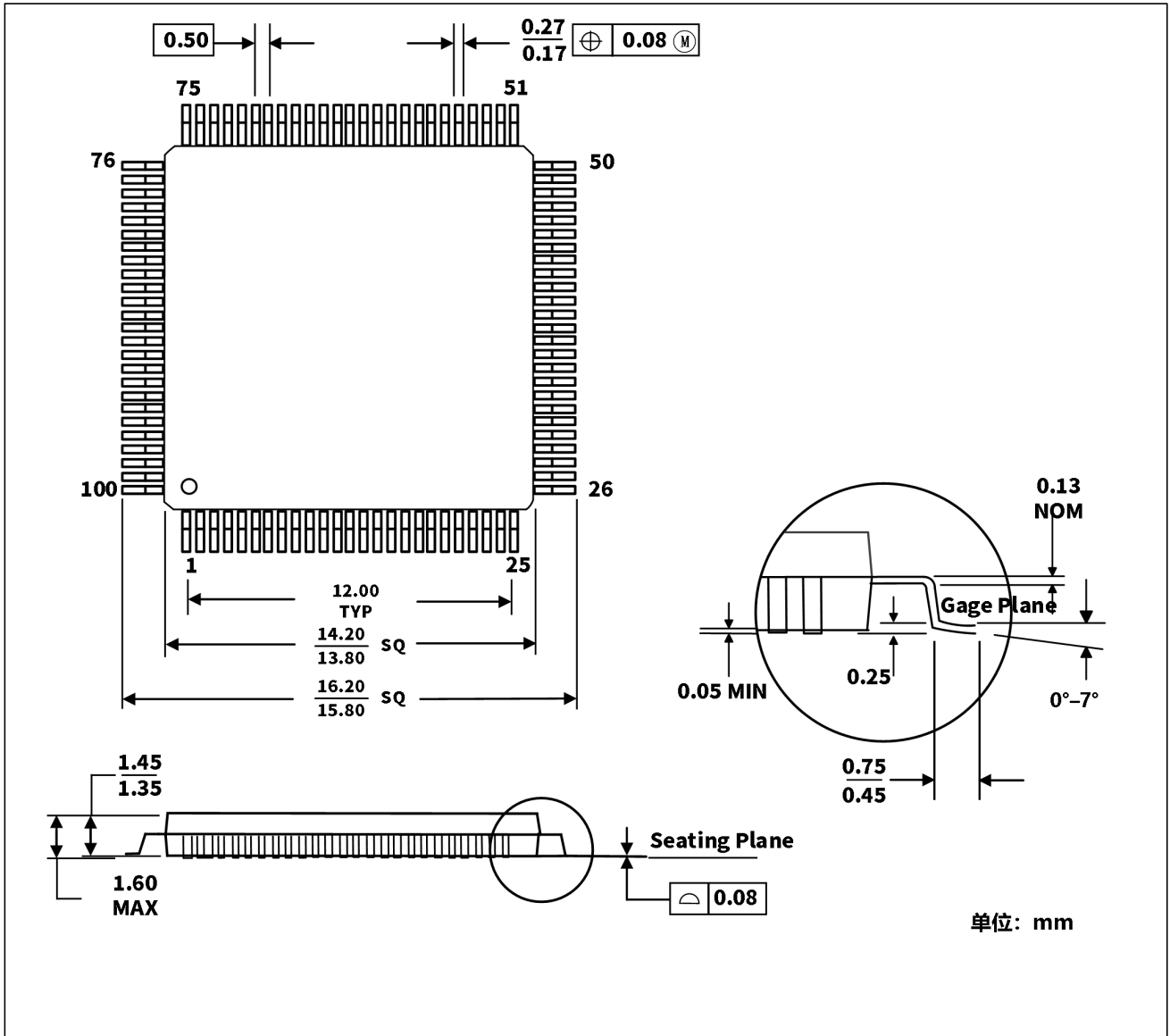
#### 5.6.16.1.2 USB 输出端口 DP 和 DM 开关特性

在推荐的操作条件范围内（除非另有说明）。

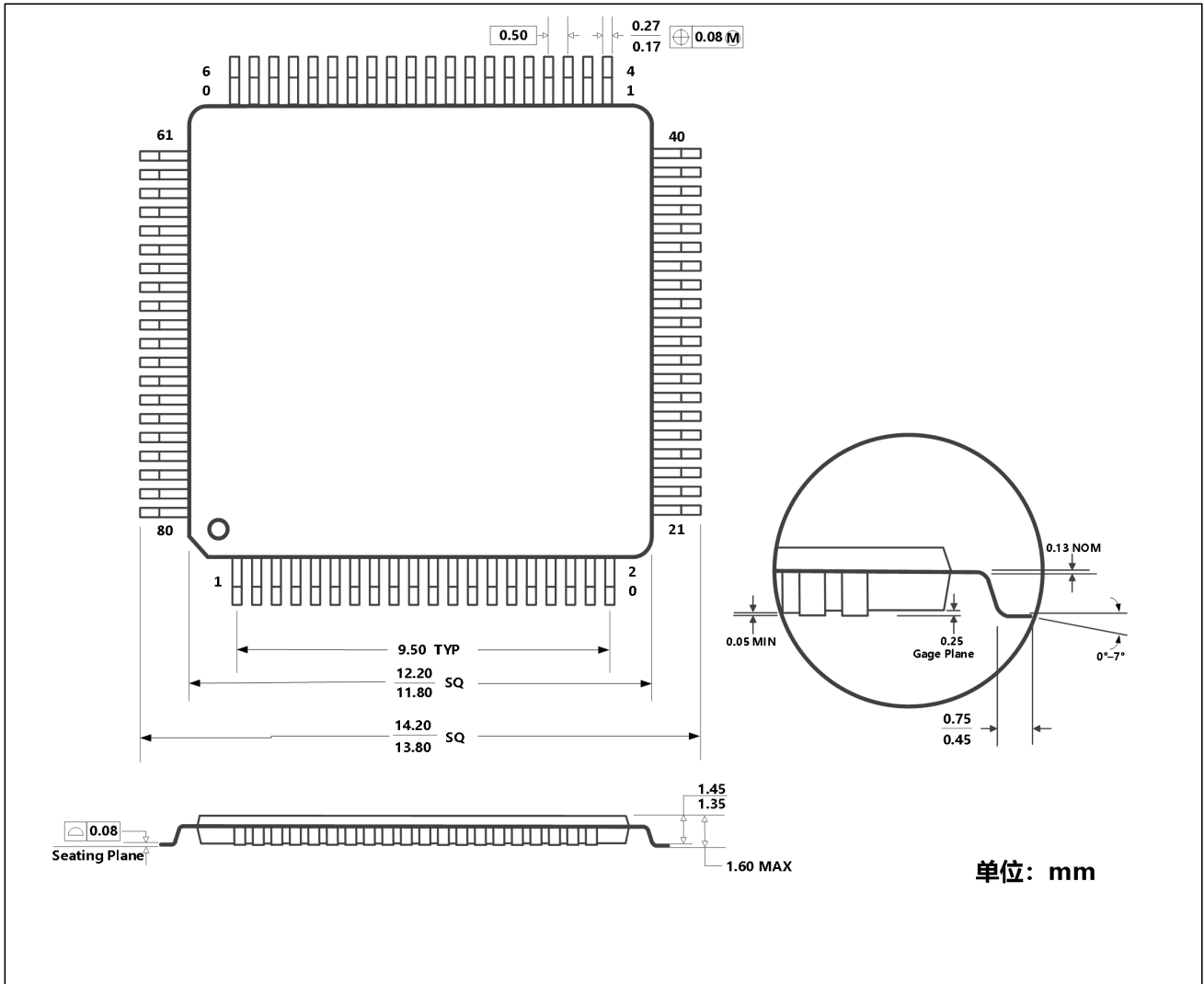
参数		测试条件	V <sub>CC</sub>	最小值	最大值	单位
V <sub>OH</sub>	D+, D- 单端	USB 2.0负载条件		2.8	3.6	V
V <sub>OL</sub>	D+, D- 单端	USB 2.0负载条件		0	0.3	V
Z <sub>(DRV)</sub>	D+, D- 阻抗			28	50	Ω
t <sub>r</sub>	上升时间	全速, 差分, C <sub>L</sub> =50pF, 10%/90%, R <sub>pu</sub> on D+		4	20	ns
t <sub>f</sub>	下降时间	全速, 差分, C <sub>L</sub> =50pF, 10%/90%, R <sub>pu</sub> on D+		4	20	ns

## 6 封装信息

### LQFP100



# LQFP80



## 7 包装及可订购信息

待定

## 8 重要注意事项及声明

Advancechip 均以“原样”提供技术性及其可靠性数据（包括数据表）、设计资源（包括参考设计）、应用或其他设计建议、网络工具、安全信息和其他资源，不保证其中不含任何瑕疵，且不做任何明示或暗示的担保，包括但不限于对适销性、适合某特定用途或不侵犯任何第三方知识产权的暗示担保。

所述资源可供专业开发人员应用 Advancechip 产品进行设计使用。您将对以下行为独自承担全部责任：(1)针对您的应用选择合适的 Advancechip 产品；(2)设计、验证并测试您的应用；(3)确保您的应用满足相应标准以及任何其他安全、安保或其他要求。所述资源如有变更，恕不另行通知。进芯电子对您使用所述资源的授权仅限于开发资源所涉及 Advancechip 产品的相关应用。除此之外不得复制或展示所述资源，也不提供其它进芯电子或任何第三方的知识产权授权许可。如因使用所述资源而产生任何索赔、赔偿、成本、损失及债务等，进芯电子对此概不负责，并且您须赔偿由此对进芯电子及其代表造成的损害。

## 联系方式

公司网址: [www.advancechip.com](http://www.advancechip.com)

联系邮箱: [sales@advancechip.com](mailto:sales@advancechip.com)

销售联系电话: 0731-88731027 (长沙)

025-66051670 (南京)

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

南京销售中心: 南京市雨花台区软件大道 106 号 2 号楼 802 室

