

# AVP32F335 数字信号处理器

## 数据手册

编号：JXDZ7.381.004 SJSC

**Advancechip**



**Electronics**

**湖南进芯电子科技有限公司**

2021 年 05 月

V1.3

### 历史版本记录

版本号	时间	起草/ 修改人	内容描述	审核人	批准人
V1.0	2019-07-01	刘律辑	首次发布		
V1.1	2019-11-18	张 巍	修正数据, 纠错		
V1.2	2021-03-24	张 巍	参数修正		
V1.3	2021-05-28	张 巍	<ol style="list-style-type: none"> <li>1. 第 1 页, 修改 12/16 位 模 数 转 换 器 (ADC), 16 个通道参数;</li> <li>2. 第 2 页, 修改表 2-1; 12/16 位模数转换器(ADC)参数;</li> <li>3. 第 9 页, 修改表 2-2; 原因: 参数拼写错误。</li> <li>4. 第 29 页, 修改表 ADC 参数;</li> <li>5. 第 94 页, 表格内容修改, 原因: 修改电气参数;</li> <li>6. 第 67 页, 删除内容 (ecan-A)。原因: 此特性是整个 can 的特性而非单独 ecan-A。</li> <li>7. 第 106 页, 修改表 6-9, 原因: 复位时序要求修改;</li> <li>8. 第 127 页, 删除内容 SYSCLKOUT=250MHz, 原因: 表述错误;</li> <li>9. 第 129 页, 修改表 6-37, 原因: 时序参数不准确。</li> <li>10. 第 139 页, 修改表 6-47。原因: 笔误。</li> <li>11. 第 142 页, 删除原表 6-48。</li> <li>12. 第 147 页, 修改表 6-54。原因: 笔误。</li> </ol>		

			13. 第 151 页, 修改表 6-58, 原因: 笔误。		
--	--	--	--------------------------------	--	--

## 目 录

1. 产品特征.....	1
2. AVP32F335 简介.....	2
2.1 器件特性.....	2
2.2 产品编码.....	3
2.3 引脚分配.....	4
2.4 信号说明.....	9
3 功能概述.....	18
3.1 方框图.....	18
3.2 内存映射.....	19
3.3 简要说明.....	23
3.3.1 AVP32F335 DSP.....	23
3.3.2 内存总线（哈弗总线架构）.....	23
3.3.3 外设总线.....	23
3.3.4 实时 JTAG 和分析.....	24
3.3.5 外部接口（XINTF）.....	24
3.3.6 闪存.....	24
3.3.7 M0, M1 SARAM.....	24
3.3.8 L0, L1, L2, L3, L4, L5, L6, L7 SARAM.....	25
3.3.9 引导 ROM.....	25
3.3.9.1 引导加载器使用的外设引脚.....	26
3.3.10 安全性.....	26
3.3.11 外设中断扩展（PIE）块.....	27
3.3.12 外部中断（XINT1-XINT7, XNMI）.....	27
3.3.13 振荡器和锁相环（PLL）.....	27
3.3.14 看门狗.....	28
3.3.15 外设时钟.....	28
3.3.16 低功耗模式.....	28

3.3.17	外设帧 0, 1, 2, 3 (PFn)	28
3.3.18	通用输入/输出 (GPIO) 复用器	29
3.3.19	32 位 CPU 定时器 (0, 1, 2)	29
3.3.20	控制外设	29
3.3.21	串行端口外设	29
3.4	寄存器映射	30
3.5	器件仿真寄存器	32
3.6	中断	33
3.6.1	外部中断	36
3.7	系统控制	38
3.7.1	OSC 和 PLL 模块	39
3.7.1.1	外部基准振荡器时钟选项	41
3.7.1.2	基于 PLL 的时钟模块	41
3.7.1.3	输入时钟丢失	42
3.7.2	看门狗	43
3.8	低功耗模式块	45
4	外设	46
4.1	DMA 概述	46
4.2	32 位 CPU 定时器 0, 定时器 1, 定时器 2	47
4.3	增强型 PWM 模块	49
4.4	高分辨率 PWM (HRPWM)	54
4.5	增强型捕捉模块 (ECAP)	55
4.6	增强型正交编码器脉冲 (EQEP)	57
4.7	ADC 特性	58
4.7.1	如果 ADC 未被使用, ADC 连接方式	63
4.7.2	ADC 寄存器	63
4.7.3	ADC 校准	64
4.8	多通道缓冲串行端口 (McBSP) 模块	64
4.9	增强型控制器局域网 (ECAN) 模块 (ECAN-A 和 ECAN-B)	67

4.10 异步串行通信接口 (SCI) 模块 (SCI-A、SCI-B、SCI-C) .....	72
4.11 串行通信接口 (SPI) 模块 (SPI-A) .....	76
4.12 内部集成电路 (I2C) .....	79
4.13 GPIO MUX .....	82
4.14 外部接口 (XINTF) .....	88
5 开发支持 .....	91
6 电气规范 .....	92
6.1 最大绝对额定值 <sup>(1)(2)</sup> .....	92
6.2 建议的运行条件 .....	93
6.3 电气特性 .....	94
6.4 功耗 .....	94
6.4.1 减少功耗 .....	96
6.4.2 功耗图 .....	98
6.4.3 散热设计考虑 .....	99
6.5 针对 DSP 的无信号缓冲的仿真器连接 .....	99
6.6 时序参数符号 .....	100
6.6.1 定时参数的通用注释 .....	101
6.6.2 测试负载电路 .....	101
6.6.3 器件时钟表 .....	101
6.7 时钟要求和特性 .....	102
6.8 电源排序 .....	104
6.9 通用输入/输出 (GPIO) .....	107
6.9.1 GPIO - 输出时序 .....	107
6.9.2 GPIO - 输入时序 .....	107
6.9.3 针对输入信号的采样窗口宽度 .....	109
6.9.4 低功耗唤醒定时 .....	109
6.10 增强型控制外设 .....	113
6.10.1 增强型脉宽调制器 (ePWM) 时序 .....	113

6.10.2	可编程控制故障区输入时序	114
6.10.3	高分辨率 PWM (HRPWM) 时序	114
6.10.4	增强型捕捉 (eCAP) 时序	115
6.10.5	增强型正交编码脉冲 (eQEP) 时序	115
6.10.6	ADC 转换开始时序	116
6.11	外部中断时序	116
6.12	I <sup>2</sup> C 电气特性和时序	117
6.13	同步串行通信接口(SPI) 主控模式时序	117
6.13.1	主模式时序	117
6.13.2	SPI 受控模式时序	122
6.14	外部接口 (XINTF) 时序	125
6.14.1	USEREADY = 0	125
6.14.2	异步模式(USEREADY=1, READYMODE=0)	125
6.14.3	同步模式(USEREADY=1, READYMODE=1)	126
6.14.4	XINTF 信号与 XCLKOUT 一致	127
6.14.5	外部接口读取时序	128
6.14.6	外部接口写入时序	130
6.14.7	带有一个外部等待状态的外部接口读取准备就绪时序	131
6.14.8	带有一个外部等待状态的外部接口写入准备就绪时序	135
6.14.9	$\overline{\text{XHOLD}}$ 和 $\overline{\text{XHOLDA}}$ 定时	137
6.15	片载模数转换器 ADC	141
6.15.1	ADC 加电控制位定时	142
6.15.2	基本定义	143
6.15.3	ADC 顺序采样模式 (单通道) (SMODE=0)	143
6.15.4	ADC 同步采样模式 (双通道) (SMODE=1)	144
6.15.5	详细说明	145
6.16	多通道缓冲串行端口(McBSP) 模块	146
6.16.1	McBSP 发送和接收时序	146
6.16.2	McBSP 作为 SPI 主控或者受控时序	150

6.17 闪存定时.....	154
7 热性能/机械数据.....	156



## 图 表 目 录

表 2-1 AVP32F335 器件的特性.....	2
图 2-1 AVP32F335 176 引脚 QP (LQFP 顶视图) .....	4
图 2-2 176 焊球 塑封 BGA (左上象限) (底视图) .....	5
图 2-3 176 焊球 塑封 BGA (右上象限) (底视图) .....	6
图 2-4 176 焊球 塑封 BGA (左下象限) (底视图) .....	7
图 2-5 176 焊球 塑封 BGA (右下象限) (底视图) .....	8
表 2-2 信号说明 <sup>(1)</sup> .....	9
图 3-1 功能方框图 .....	18
图 3-2 AVP32F335 内存映射.....	20
表 3-1 AVP32F335 中闪存扇区的地址 .....	21
表 3-2 使用安全代码模块的影响 .....	21
表 3-3 等待状态 .....	22
表 3-4 引导模式选择 .....	25
表 3-5 外设引导加载引脚 .....	26
表 3-6 外设帧 0 寄存器 <sup>(1)</sup> .....	30
表 3-7 外设帧 1 寄存器.....	31
表 3-8 外设帧 2 寄存器.....	31
表 3-9 外设帧 3 寄存器.....	31
表 3-10 新增寄存器 .....	32
表 3-11 器件仿真寄存器 .....	32
图 3-3 外部和 PIE 中断源 .....	33
图 3-4 外部中断 .....	34
图 3-5 使用 PIE 块的中断复用.....	35
表 3-12 PIE 外设中断矢量表 <sup>(1)</sup> .....	35
表 3-13 PIE 配置和控制寄存器.....	36
表 3-14 外部中断寄存器 .....	36
图 3-6 时钟和复位域 .....	38

表 3-15 PLL、时钟、看门狗和低功耗模式寄存器 .....	39
图 3-7 OSC 和 PLL 模块方框图.....	40
图 3-8 3.3V 外部振荡器的使用.....	40
图 3-9 1.9V 外部振荡器的使用.....	40
图 3-10 外部振荡器的使用 .....	41
表 3-16 PLL 设置 <sup>(1)</sup> .....	41
表 3-17 CLKIN 分频选项 .....	42
表 3-18 可能的 PLL 配置模式.....	42
图 3-11 CPU - 看门狗模块 .....	44
表 3-19 低功耗模式 .....	45
图 4-1 DMA 功能方框图.....	47
图 4-3 CPU 定时器中断信号和输出信号.....	48
表 4-1 CPU 定时器 0, 1, 2 配置和控制寄存器 .....	49
图 4-4 时基计数器同步方案 3.....	50
表 4-2 ePWM 控制和状态寄存器 (PF1 中的默认置) .....	51
表 4-3 ePWM 控制和状态寄存器 (PF3 中的默认置重新映射的配置-可由 DMA 访问) .....	52
图 4-6 eCAP 功能方框图 .....	55
表 4-4 eCAP 控制和状态寄存器 .....	56
表 4-5 eQEP 控制和状态寄存器 .....	58
图 4-8 ADC 模块的方框图 .....	60
图 4-9 带有内部基准的 ADC 引脚连接 .....	61
图 4-10 带有外部基准的 ADC 引脚连接 .....	62
表 4-6 ADC 寄存器 .....	63
图 4-11 显示了 McBSP 模块的方框图。 .....	66
表 4-7 McBSP 寄存器汇总 .....	66
图 4-12 eCAN 方框图和接口电路 .....	69
表 4-8 3.3V eCAN 收发器用于 AVP32F335.....	69
图 4-13 eCAN-A 内存映射 .....	70
图 4-14 eCAN-B 内存映射 .....	71

表 4-9 CAN 寄存器映射 <sup>(1)</sup> .....	72
表 4-10 SCI-A 寄存器 <sup>(1)</sup> .....	74
表 4-11 SCI-B 寄存器 <sup>(1) (2)</sup> .....	74
表 4-12 SCI-C 寄存器 <sup>(1) (2)</sup> .....	75
图 4-15 串行通信接口(SCI) 模块方框图 .....	76
表 4-13 SPI-A 寄存器 .....	78
图 4-16 SPI 模块方框图 (受控模式) .....	79
图 4-17 I2C 外设模块接口 .....	80
表 4-14 I2C-A 寄存器 .....	81
图 4-18 GPIO 复用 .....	83
表 4-16 GPIO-A 复用器外设选择矩阵 .....	85
表 4-17 GPIO-B 复用器外设选择矩阵 .....	86
表 4-18 GPIO-C 复用器外设选择矩阵 .....	87
图 4-20 外部接口方框图 .....	89
图 4-21 典型的 16 位数据总线 XINTF 连接 .....	89
图 4-22 典型的 32 位数据总线 XINTF 连接 .....	90
表 4-19 XINTF 配置和控制寄存器映射 .....	90
表 6-1 AVP32F335 在 150MHz SYSCLKOUT 运行条件下的流耗 .....	95
表 6-2 不同外设的典型流耗 (150MHz 上时) <sup>(1)</sup> .....	96
图 6-1 典型运行电流与频率间的关系 (AVP32F335) .....	98
图 6-2 典型运行功率与频率间的关系 (AVP32F335) .....	99
图 6-3 针对 DSP 的无信号缓冲的仿真器连接 .....	100
图 6-4 3.3V 测试负载电路 .....	101
表 6-3 片上设备时钟的周期特征 (150MHz 主时钟) .....	101
表 6-4 片上设备时钟的周期特征 (100MHz 主时钟) .....	102
表 6-5 输入时钟频率 .....	102
表 6-6 XCLKIN 时序要求- PLL 被启用 .....	102
表 6-8 XCLKOUT 开关特性 (PLL 旁路或者被禁用) <sup>(1) (2)</sup> .....	103
图 6-5 时钟时序 .....	103

图 6-6 加电复位 .....	105
图 6-7 热复位 .....	106
图 6-8 写入 PLLCR 寄存器所产生的效果的示例 .....	107
表 6-10 通用输出开关特性 .....	107
图 6-9 通用输出时序 .....	107
表 6-11 通用输入时序要求 .....	107
图 6-10 采样模式 .....	108
图 6-11 通用输入定时 .....	109
表 6-12 IDLE 模式时序要求 <sup>(1)</sup> .....	109
表 6-13 IDLE 模式开关特性 <sup>(1)</sup> .....	110
图 6-12 IDLE 进入和退出时序 .....	110
表 6-14 STANDBY 模式时序要求 .....	111
表 6-15 STANDBY 模式开关特性 .....	111
图 6-13 STANBY 进入和退出时序 .....	112
表 6-16 HALT 模式时序要求 .....	112
表 6-17 HALT 模式开关特性 .....	112
图 6-14 使用 GPIO <sub>n</sub> 的 HALT 唤醒 .....	113
表 6-18 PWM 时序要求 <sup>(1)</sup> .....	114
表 6-19 PWM 开关特性 .....	114
表 6-20 可编程控制故障区输入时序要求 <sup>(1)</sup> .....	114
图 6-15 PWM Hi-Z 特性 .....	114
表 6-21 在 SYSCLKOUT= (60-150MHz) 时, 高分辨率 PWM 开关特性 .....	115
表 6-22 增强型捕捉 (eCAP) 时序要求 <sup>(1)</sup> .....	115
表 6-23 eCAP 开关特性 .....	115
表 6-24 增强型正交编码脉冲 (eQEP) 时序要求 <sup>(1)</sup> .....	115
表 6-25 eQEP 开关特性 .....	116
表 6-26 外部 ADC 转换开始开关特性 .....	116
图 6-16 <b>ADCSOCA0</b> 和 <b>ADCSOCB0</b> 时序图 .....	116
表 6-27 外部中断时序要求 <sup>(1)</sup> .....	116

表 6-28 外部中断开关特性 <sup>(1)</sup> .....	116
图 6-17 外部中断时序 .....	117
表 6-29 I <sup>2</sup> C 时序 .....	117
表 6-30 SPI 主控模式外部时序 (时钟相位= 0) <sup>(1) (2) (3) (4) (5)</sup> .....	118
图 6-18 SPI 主控模式外部时序 (时钟相位= 0) .....	119
表 6-31 SPI 主控模式外部时序 (时钟相位= 1) <sup>(1) (2) (3) (4) (5)</sup> .....	120
图 6-19 SPI 主控模式外部时序 (时钟相位= 1) .....	121
表 6-32 SPI 受控模式外部时序 (时钟相位= 0) <sup>(1) (2) (3) (4) (5)</sup> .....	122
图 6-20 SPI 受控模式外部时序 (时钟相位= 0) .....	123
表 6-33 SPI 受控模式外部时序 (时钟相位= 1) <sup>(1) (2) (3) (4)</sup> .....	124
图 6-21 SPI 受控模式外部时序 (时钟相位= 1) .....	124
表 6-34 XTIMING 中配置的参数和脉冲持续时间之间的关系 .....	125
表 6-35 XINTF 时钟配置 .....	127
图 6-22 XTIMCLK 和 SYSCLKOUT 之间关系 .....	127
表 6-36 外部存储器接口读取时序要求 .....	129
表 6-37 外部内存接口读取开关特性 .....	129
图 6-23 示例读取访问 .....	129
表 6-38 外部存储器接口写入开关特性 .....	130
图 6-24 示例写入访问 .....	130
表 6-39 外部接口读取开关特性 ( (读取准备就绪, 1 个等待状态) ) .....	131
表 6-40 外部接口读取时序要求 (读取就绪, 1 个等待状态) .....	131
表 6-41 同步 XREADY 时序要求 ( (读取准备就绪, 1 个等待状态) <sup>(1)</sup> .....	131
表 6-42 异步 XREADY 时序要求 (读取准备就绪, 1 个等待状态) .....	132
图 6-25 使用同步 XREADY 访问读取的示例 .....	133
图 6-26 使用异步 XREADY 访问读取的示例 .....	134
表 6-43 外部接口写入开关特性 (写入准备就绪, 1 个等待状态) .....	135
表 6-44 同步 XREADY 时序要求 (写入准备就绪, 1 个等待状态) <sup>(1)</sup> .....	135
表 6-45 异步 XREADY 时序要求 (写入准备就绪, 1 个等待状态) <sup>(1)</sup> .....	135
图 6-27 使用同步 XREADY 访问写入的示例 .....	136

图 6-28 使用异步 XREADY 访问写入的示例 .....	137
表 6-46 $\overline{\text{XHOLD}}$ 和 $\overline{\text{XHOLDA}}$ 时序要求 (XCLKOUT=XTIMCLK) <sup>(1)(2)</sup> .....	138
图 6-29 外部接口保持波形 .....	139
表 6-47 $\overline{\text{XHOLD}}$ 和 $\overline{\text{XHOLDA}}$ 时序要求 (XCLKOUT=1/2 XTIMCLK) <sup>(1)(2)(3)</sup> .....	139
图 6-30 $\overline{\text{XHOLD}}$ 和 $\overline{\text{XHOLDA}}$ 时序要求 (XCLKOUT=1/2 XTIMCLK) .....	140
表 6-48 ADC 电气特性 (在推荐的运行条件) <sup>(1)(2)</sup> .....	141
图 6-31 ADC 加电控制位时序 .....	142
表 6-49 ADC 加电延迟 .....	142
表 6-50 不同 ADC 配置的典型电流消耗 (在 7.5MHz ADCCLK 上) <sup>(1)(2)</sup> .....	142
图 6-32 ADC 输入阻抗模型 .....	143
图 6-33 顺序采样模式 (单通道) 时序 .....	144
表 6-51 顺序采样模式时序 .....	144
图 6-34 同步采样模式 (双通道) 时序 .....	145
表 6-52 同步采样模式时序 .....	145
表 6-53 McBSP 时序要求 <sup>(1)(2)</sup> .....	147
表 6-54 McBSP 开关特性 <sup>(1)(2)</sup> .....	147
图 6-35 McBSP 接收时序 .....	149
图 6-36 McBSP 发送时序 .....	149
表 6-55 McBSP 作为 SPI 主控或者受控定时要求 (CLKSTP=10b, CLKXP=0) .....	150
表 6-56 McBSP 作为 SPI 主控或者受控开关特性 (CLKSTP=10b, CLKXP=0) .....	150
图 6-37 作为 SPI 主控或者受控时的 McBSP 时序: CLKSTP=10b, CLKXP=0 .....	150
表 6-57 McBSP 作为 SPI 主控或者受控定时要求 (CLKSTP=11b, CLKXP=0) .....	151
表 6-58 McBSP 作为 SPI 主控或者受控开关特性 (CLKSTP=11b, CLKXP=0) .....	151
图 6-38 作为 SPI 主控或者受控时的 McBSP 时序: CLKSTP=11b, CLKXP=0 .....	151
表 6-59 McBSP 作为 SPI 主控或者受控定时要求 (CLKSTP=10b, CLKXP=1) .....	152
表 6-60 McBSP 作为 SPI 主控或者受控开关特性 (CLKSTP=10b, CLKXP=1) .....	152
图 6-39 作为 SPI 主控或者受控时的 McBSP 时序: CLKSTP=10b, CLKXP=1 .....	152
表 6-61 McBSP 作为 SPI 主控或者受控定时要求 (CLKSTP=11b, CLKXP=1) .....	153
表 6-62 McBSP 作为 SPI 主控或者受控开关特性 (CLKSTP=11b, CLKXP=1) <sup>(1)</sup> .....	153

图 6-40 作为 SPI 主控或者受控时的 McBSP 时序: CLKSTP=11b, CLKXP=1.....	153
表 6-63 闪存对于 (S) 温度材料的耐受度 <sup>(1)</sup> .....	154
表 6-64 150MHz SYSCLKOUT 上的闪存参数.....	154
表 6-65 闪存/ OTP 访问时序.....	154
表 6-66 闪存数据保持持续时间.....	154
表 6-67 不同频率上所需最小的闪存/一次性可编程 (OTP) 等待状态.....	154
表 7-1 散热模型 176 引脚 QP 结果.....	156
表 7-2 散热模型 176 引脚 BA 结果.....	156

## 1. 产品特征

- 高性能静态 CMOS 技术
  - 高达 150MHz ( 6.67ns 周期时间 )
  - 1.9V/1.8V 转 1.5V 内核, 3.3V I/O 设计
- 高性能 32 位 CPU
  - IEEE-754 单精度浮点单元 ( FPU )
  - 16 x 16 和 32 x 32 介质访问控制 (MAC) 运算
  - 16 x 16 双 MAC
  - 哈佛 (Harvard) 总线架构
  - 快速中断响应和处理
  - 统一存储器编程模型
  - 高效代码 ( 使用 C/C++ 和汇编语言 )
- 6 通道 DMA 处理器 ( 用于 ADC , McBSP , ePWM , XINTF 和 SARAM )
- 16 位或 32 位外部接口 (XINTF)
  - 超过 2M × 16 地址范围
- 片载存储器
  - 256K×16 闪存, 34K×16 SARAM
  - 1K x 16 一次性可编程 (OTP) ROM
- 引导 ROM (8K X 16)
  - 支持软件引导模式 ( 通过 SCI,SPI,CAN,I2C,McBSP,XINTF 和并行 I/O )
  - 标准数学表
- 时钟和系统控制
  - 支持动态锁相环 (PLL) 比率变化
  - 片载振荡器
  - 安全装置定时器模块
- GPIO0 到 GPIO63 引脚可以连接到八个外部内核中断 其中的一个
- 可支持全部 58 个外设中断的外设中断扩展(PIE)块
- 128 位安全密钥/锁
  - 保护闪存 / OTP/RAM 模块
  - 防止固件逆向工程
- 低功耗模式和省电模式
  - 支持 IDLE ( 空闲 ) ,STANDBY ( 待机 )、HALT ( 暂停 ) 模式
- 字节序 : 小端序
- 温度选项:
  - A : -40°C 至 85°C
  - S : -40°C 至 125°
- 增强型控制外设
  - 多达 18 个脉宽调制 (PWM) 输出
  - 高达 6 个支持 150ps 微边界定位 (MEP) 分辨率的高分辨率脉宽调制器 (HRPWM) 输出
  - 高达 6 个事件捕捉输入
  - 多达两个正交编码器接口
  - 高达 8 个 32 位定时器 ( 6 个 eCAP 以及 2 个 eQEP )
  - 高达 9 个 32 位定时器 ( 6 个 ePWM 以及 3 个 XINTCTR )
- 三个 32 位 CPU 定时器
- 串行端口外设
  - 多达 2 个控制器局域网 (CAN) 模块
  - 多达 3 个 SCI (UART) 模块
  - 高达 2 个 McBSP 模块 ( 可配置为 SPI )
  - 一个 SPI 模块
  - 一个内部集成电路 (I2C) 总线
- 12/16 位模数转换器 (ADC) , 16 个通道
  - 12 位 — 267ns 转换率
  - 2 x 8 通道输入复用器
  - 两个采样保持
  - 单一/同步转换
  - 内部或者外部基准
- 8 通道 LDC 电感传感器
- 8 通道 CMP(比较器)
- 多达 88 个具有输入滤波功能可单独编程的多路复用通用输入输出 (GPIO) 引脚
- JTAG 边界扫描支持
- 高级仿真特性
  - 分析和断点功能
  - 借助硬件的实时调试
- 开发支持包括
  - ANSI C/C++ 编译器/汇编语言/连接器
  - Code Composer Studio™ IDE
  - DSP/BIOS™
  - 数字电机控制和数字电源软件库
- 封装选项:
  - LQFP176
  - BGA176



## 2. AVP32F335 简介

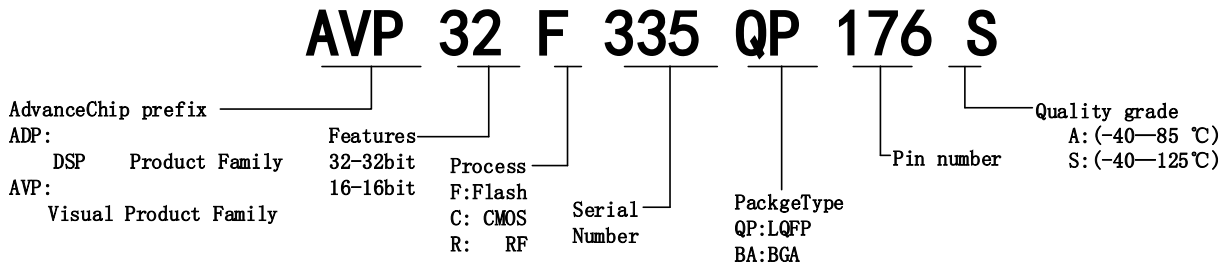
### 2.1 器件特性

**表 2-1 AVP32F335 器件的特性**

特性		AVP32F335 (150MHz)
指令周期		6.67ns
浮点单元		支持
3.3V 片载闪存 ( 16 位字 )		256K
单周期访问RAM (SARAM) ( 16 位字 )		34K
一次性可编程(OTP) ROM ( 16 位字 )		1K
针对片载闪存/ SARAM/OTP 块的代码安全		支持
引导ROM (8K X 16)		支持
16/32 位外部接口 ( XINTF )		支持
6 通道直接内存存取(DMA)		支持
PWM 输出		ePWM1/2/3/4/5/6
HRPWM 通道		ePWM1A/2A/3A/4A/5A/6A
32 位CAPTURE 输入或者辅助PWM 输出		eCAP1/2/3/4/5/6
32 位正交编码器脉冲(QEP) 通道 ( 四个输入/通道 )		eQEP1/2
看门狗定时器		支持
12 位模数转换器 (ADC)	通道的数量	16
	MSPS	3.75
	转换时间   12 位	267ns
32 位CPU 定时器		3
多通道缓冲串行端口(McBSP)/ SPI		2(A/B)
串行外设接口(SPI)		1
串行通信接口(SCI)		3(A/B/C)
增强型控制器局域网(eCAN)		2(A/B)
内部集成电路(I2C)		1
通用I/O 引脚 ( 共享 )		88
外部中断		8
封装	LQFP176	支持
	BGA176	支持
温度选项		S : -40°C 至125°C   LQFP176 , BGA176

## 2.2 产品编码

AVP32F335 产品代号编码规则:



## 2.3 引脚分配

图 2-1 显示了 176 引脚薄型四方扁平封装 (LQFP) 引脚分配。176 球栅阵列 (BGA) 端子分配将显示在图 2-2 至图 2-5。表 2-2 说明了每个引脚的功能。

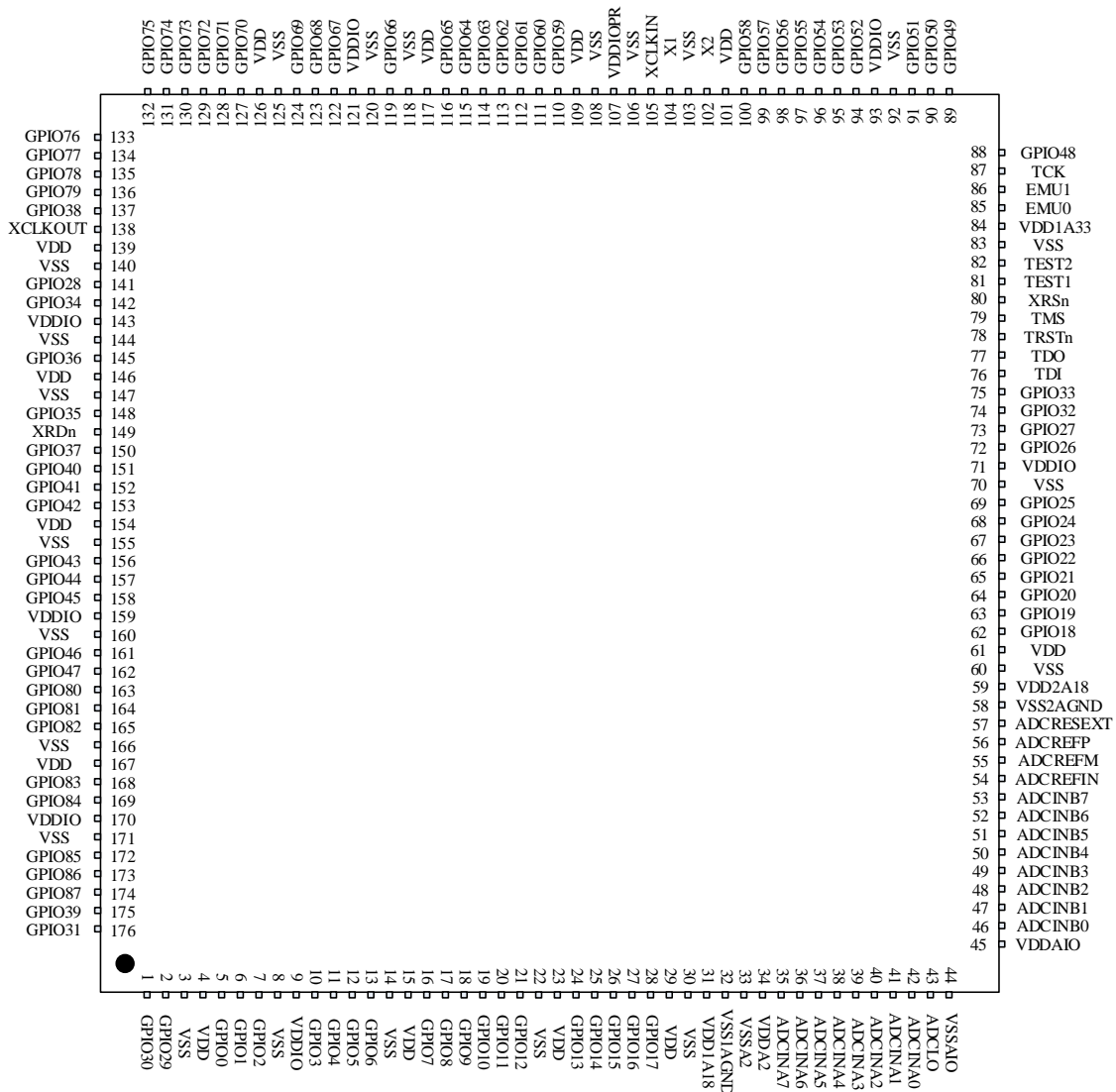


图 2-1 AVP32F335 176 引脚 QP (LQFP 顶视图)

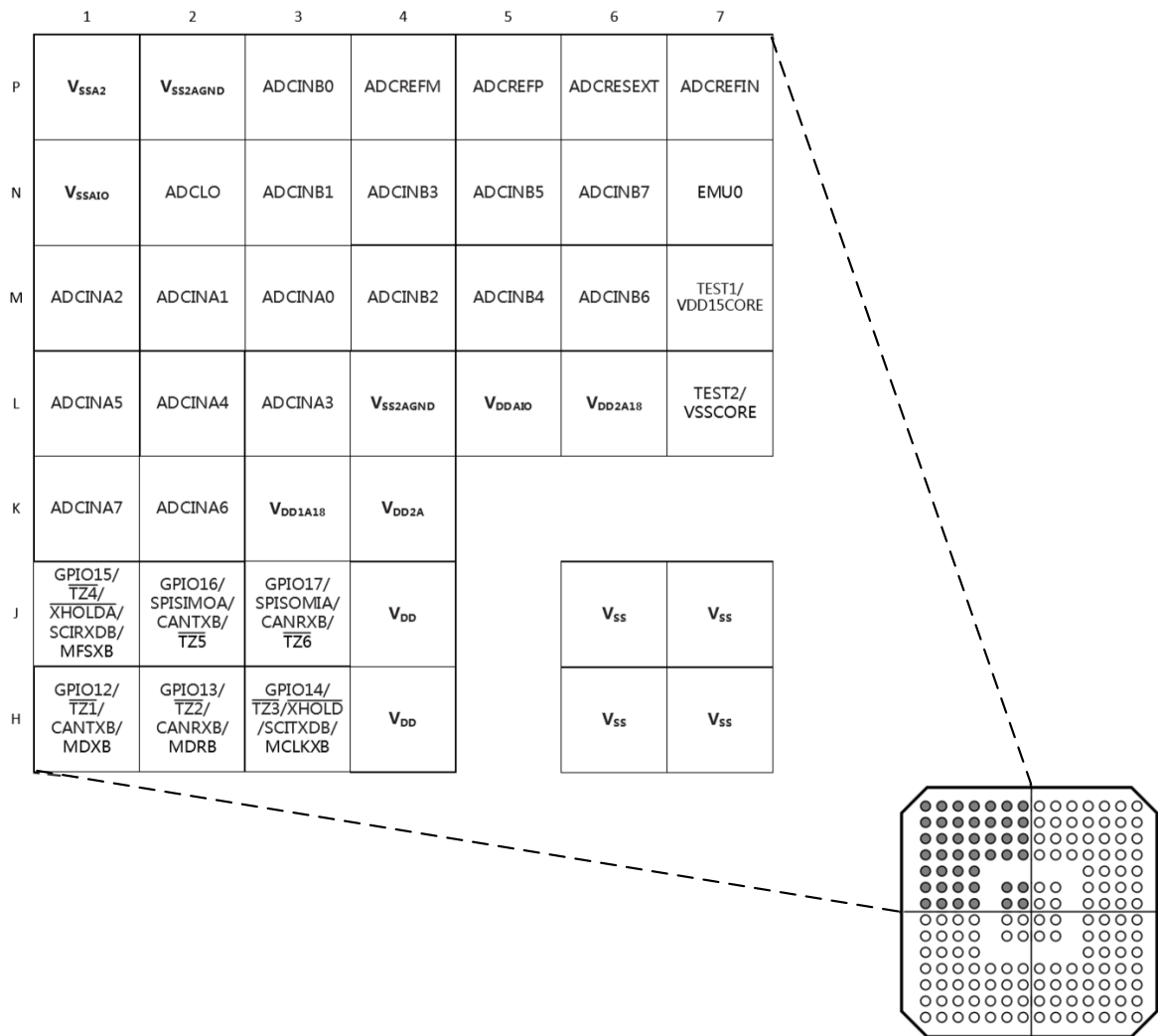


图 2-2 176 焊球 塑封 BGA (左上象限) (底视图)

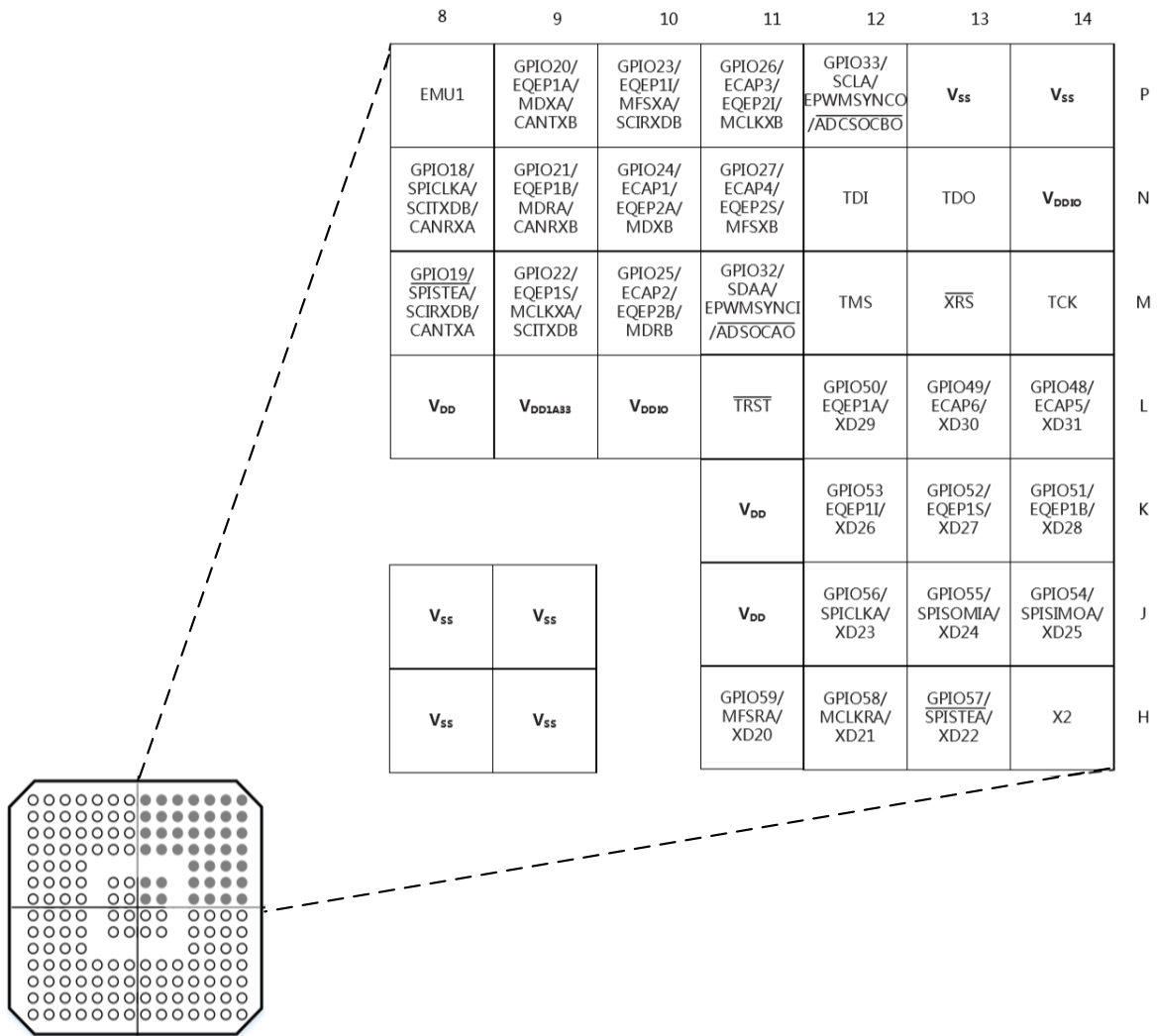


图 2-3 176 焊球 塑封 BGA (右上象限) (底视图)

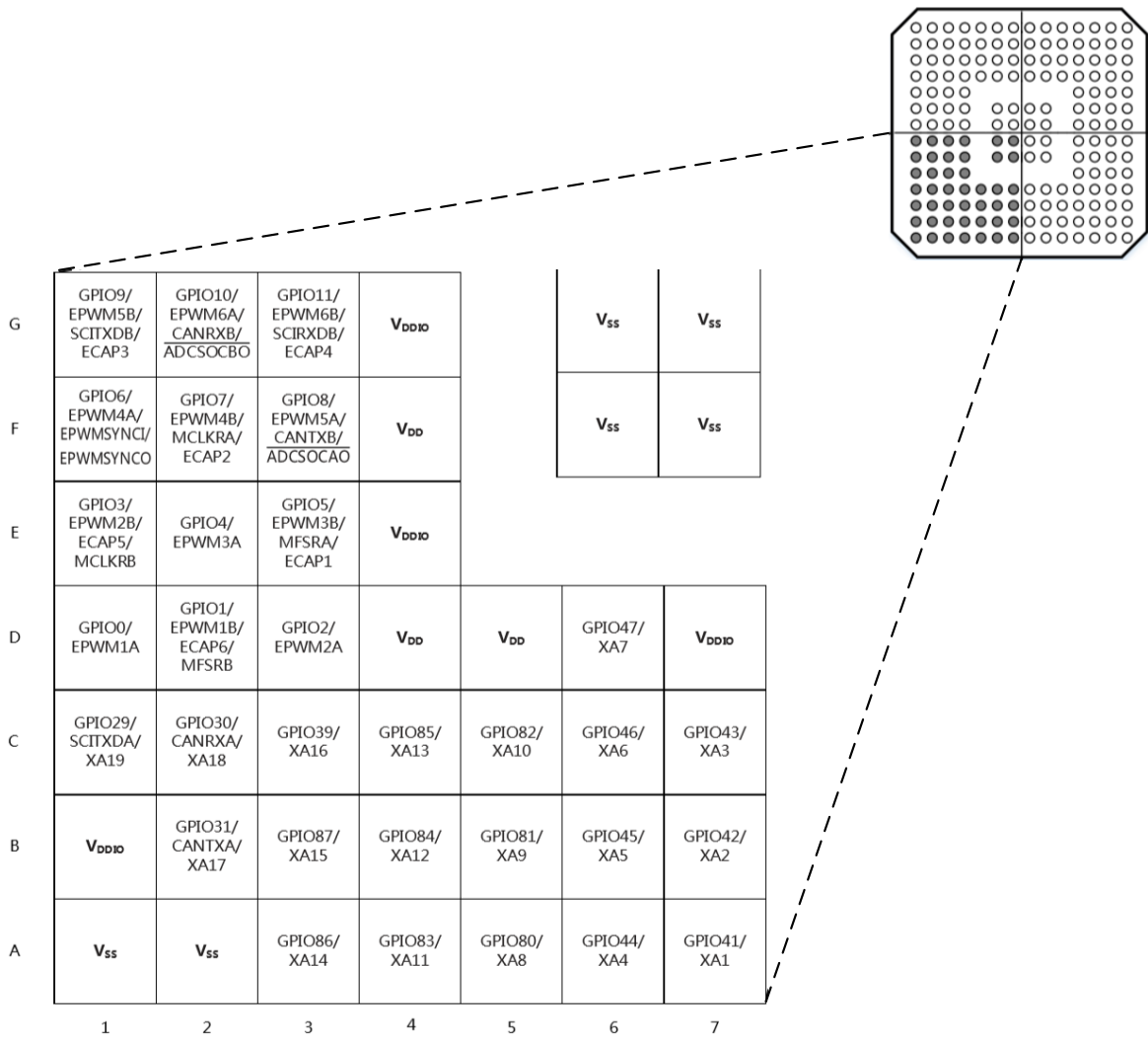


图 2-4 176 焊球 塑封 BGA (左下象限) (底视图)

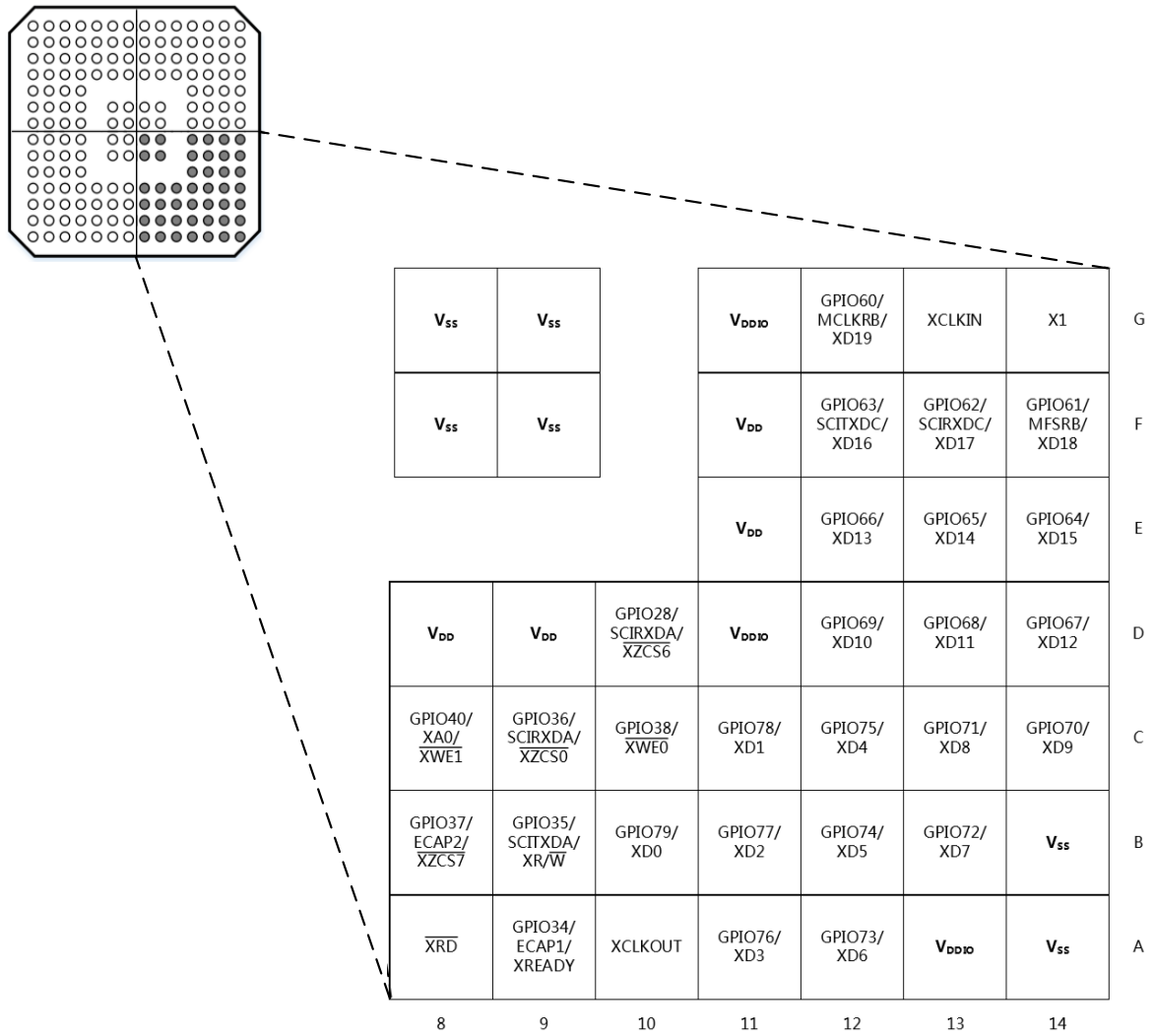


图 2-5 176 焊球 塑封 BGA（右下象限）（底视图）

## 2.4 信号说明

表 2-2 对这些信号进行了说明。GPIO 功能（用粗斜体显示）是复位时的默认值，它们下面列出的外设信号是复用功能。有些外设功能并不在所有器件上提供。详细信息请见表 2-1 和表 2-2。输入不是 5V 耐压。所有能够产生 XINTF 输出功能的引脚有 8mA（典型）的驱动强度。即使引脚没有配置 XINTF 功能，也有此驱动能力。所有其他引脚为 4mA 驱动力的驱动典型值（除另有注明外）。所有 GPIO 引脚为 I/O/Z 且有一个内部上拉电阻，此内部上拉电阻可在每个引脚上有选择性的启用/禁用。这一特性只适用于 GPIO 引脚。GPIO0-GPIO11 引脚上的上拉电阻在复位时并不启用。GPIO12-GPIO87 引脚上的上拉电阻复位时被启用。

**表 2-2 信号说明<sup>(1)</sup>**

名称	引脚编号		说明
	LQFP176 PIN#	BGA176 BALL#	
<b>JTAG</b>			
$\overline{\text{TRST}}$	78	L11	使用内部下拉电阻进行 JTAG 测试复位。当被驱动为高电平时， $\overline{\text{TRST}}$ 使扫描系统获得器件运行的控制权。如果这个信号未连接或者被驱动至低电平，此器件在功能模式下运转，并且测试复位信号被忽略。注释： $\overline{\text{TRST}}$ 是一个高电平有效测试引脚并且必须在正常器件运行期间一直保持低电平。在这个引脚上需要一个外部下拉电阻器。这个电阻器的值应该基于适用于这个设计的调试器推进源代码的驱动强度。通常一个 2.2k $\Omega$ 电阻器可提供足够的保护。由于这是应用专用的，建议针对调试器和应用正确运行对每个目标板进行验证。（I，↓）
TCK	87	M14	带有内部上拉电阻（I，↑）的 JTAG 测试时钟
TMS	79	M12	带有内部上拉电阻的 JTAG 测试模式选择（TMS）。这个串行控制输入在 TCK 上升沿上的 TAP 控制器中计时。（I，↑）
TDI	76	N12	带有内部上拉电阻的 JTAG 测试数据输入（TDI）。TDI 在 TCK 的上升沿上所选择的寄存器（指令或者数据）内计时。（I，↑）
TDO	77	N13	JTAG 扫描输出，测试数据输出（TDO）。所选寄存器（指令或者数据）的内容被从 TCK 下降沿上的 TDO 移出。（O/Z 8mA 驱动）
EMU0	85	N7	仿真器引脚 0。当 $\overline{\text{TRET}}$ 被驱动至高电平时，这个引脚被用作一个到（或者来自）仿真器系统的中断并且在 JTAG 扫面过程中被定义为输入/输出。这个引脚也被用于将器件置于边界扫面模式中。在 EMU0 引脚处于逻辑高电平状态并且 EMU1 引脚处于逻辑低电平状态时， $\overline{\text{TRET}}$ 引脚的上升沿将把器件锁存在边界扫面模式。（I/O/Z，8mA 驱动强度↑）请注意：建议在这个引脚上连接一个外部上拉电阻器。这个电阻器的值应该基于适用于这个设计的调试器推进源代码的驱动强度。通常一个 2.2k $\Omega$ 至 4.7k $\Omega$ 的电阻器已可以满足要求。由于这是应用专用的，建议针对调试器和应用正确运行对每个目标板进行验证。
EMU1	86	P8	仿真器引脚 1。当 $\overline{\text{TRET}}$ 被驱动至高电平时，这个引脚被用作一个到（或者来自）仿真器系统的中断并且在 JTAG 扫面过程中被定义为输入/输出。这个引脚也被用于将器件置于边界扫面模式中。在 EMU0 引脚处于逻辑高电平状态并且 EMU1 引脚处于逻辑低电平状态时， $\overline{\text{TRET}}$ 引脚的上升沿将把器件锁存在边界扫面模式。（I/O/Z，8mA 驱动强度↑）请注意：建议在这个引脚上连接一个外部上拉电阻器。这个电阻器的值应该基于适用于这个设计的调试器推进源代码的驱动强度。通常一个 2.2k $\Omega$ 至 4.7k $\Omega$ 的电阻器已可以满足要求。由于这是应用专用的，建议针对调试器和应用正确运行对每个目标板进行验证。
<b>其他</b>			



名称	引脚编号		说明
	LQFP176 PIN#	BGA176 BALL#	
V <sub>DD1a33</sub>	84	L9	端口电压 3.3V，外接模拟端口电源。
TEST1/ VDD15CORE	81	M7	悬空/推荐接内核供电外接退耦电容正端。
TEST2/ VSSCORE	82	L7	悬空/推荐接内核供电外接退耦电容负端。
<b>时钟</b>			
XCLKOUT	138	A10	取自 SYSCLKOUT 的输出时钟。XCLKOUT 或者 SYSCLKOUT 的频率一样、或者为其一半，或者为其四分之一。这是由位 18 : 16 (XTIMCLK) 和在 XINTCNF2 寄存器中的位 2 (CLKMODE) 控制的。复位时，XCLKOUT=SYSCLKOUT/4。通过将 XINTCNF2 [CLKOFF] 设定为 1，XCLKOUT 信号可被关闭。与其它 GPIO 引脚不同，复位时，不将 XCLKOUT 引脚置于一个高阻抗状态。(O/Z, 8mA 驱动)
XCLKIN	105	G13	外部振荡器输入。这个引脚被用于从一个外部 3.3V 振荡器反馈入一个时钟。在这种情况下，X1 引脚必须连接到 GND。如果使用到了晶振/谐振器 (或 1.9V 外部振荡器被用来把时钟馈入 X1 引脚)，此引脚必须连接到 GND。(I)
X1	104	G14	内部/外部振荡器输入。为了使用这个振荡器，一个石英晶振或者一个陶瓷电容器必须被连接在 X1 和 X2 上。X1 引脚以 1.9V 内核数字电源为基准。一个 1.9V 外部振荡器也可被连接至 X1 引脚。在这种情况下，XCLKIN 引脚必须接地。如果一个 3.3V 外部振荡器与 XCLKIN 引脚一起使用的话，X1 必须接至 GND。(I)
X2	102	H14	内部振荡器输出。可将一个石英晶振或者一个陶瓷电容器连接在 X1 和 X2。如果 X2 未使用，它必须保持在未连接状态。(O)
<b>复位</b>			
$\overline{\text{XRS}}$	80	M13	器件复位 (输入) 和看门狗复位 (输出)。器件复位。 $\overline{\text{XRS}}$ 导致器件终止执行。PC 将指向包含在位置 0x3FFFC0 中的地址。当 XRS 被置为高电平时，在 PC 指向的位置开始执行。当一个看门狗复位发生时，这个引脚被 DSC 驱动至低电平。看门狗复位期间，在 512 个 OSCCLK 周期的看门狗复位持续时间内，XRS 引脚被驱动为低电平。(I/OD, ↑) 这个引脚的输出缓冲器是一个有内部上拉电阻的开漏器件。建议由一个开漏器件驱动这个引脚。
<b>ADC 信号</b>			
ADCINA7	35	K1	ADC 组 A，通道 7 输入 (I)
ADCINA6	36	K2	ADC 组 A，通道 6 输入 (I)
ADCINA5	37	L1	ADC 组 A，通道 5 输入 (I)
ADCINA4	38	L2	ADC 组 A，通道 4 输入 (I)
ADCINA3	39	L3	ADC 组 A，通道 3 输入 (I)
ADCINA2	40	M1	ADC 组 A，通道 2 输入 (I)
ADCINA1	41	M2	ADC 组 A，通道 1 输入 (I)
ADCINA0	42	M3	ADC 组 A，通道 0 输入 (I)
ADCINB7	53	N6	ADC 组 B，通道 7 输入 (I)
ADCINB6	52	M6	ADC 组 B，通道 6 输入 (I)
ADCINB5	51	N5	ADC 组 B，通道 5 输入 (I)
ADCINB4	50	M5	ADC 组 B，通道 4 输入 (I)
ADCINB3	49	N4	ADC 组 B，通道 3 输入 (I)
ADCINB2	48	M4	ADC 组 B，通道 2 输入 (I)
ADCINB1	47	N3	ADC 组 B，通道 1 输入 (I)
ADCINB0	46	P3	ADC 组 B，通道 0 输入 (I)
ADCLO	43	N2	低基准 (连接至模拟接地) (I)
ADCRESEXT	57	P6	ADC 外部电流偏置电阻器。将一个 22kΩ 电阻器接至模拟接地。
ADCREFIN	54	P7	外部基准输入 (I)
ADCREFP	56	P5	内部基准正输出。 要求将一个低等效串联电阻 (ESR) (低于 1.5Ω) 的 2.2μF 陶瓷旁通电容器接至模拟接地。 注释：使用 ADC 时钟速率从系统使用的电容器数据表中提取 ESR 技术规范。

名称	引脚编号		说明
	LQFP176 PIN#	BGA176 BALL#	
ADCREFM	55	P4	内部基准中输出。要求将一个低等效串联电阻 (ESR) ( 低于 1.5Ω ) 的 2.2μF 陶瓷旁通电容器接至模拟接地。 注释：使用 ADC 时钟速率从系统使用的电容器数据表中提取 ESR 技术规范。
<b>CPU 和 I/O 电源引脚</b>			
VDDA2	34	K4	ADC 模拟电源引脚
VSSA2	33	P1	ADC 模拟接地引脚
VDDAIO	45	L5	ADC 模拟 I/O 电源引脚
VSSAIO	44	N1	ADC 模拟 I/O 接地引脚
VDD1A18	31	K3	ADC 模拟电源引脚
VSS1AGND	32	L4	ADC 模拟接地引脚
VDD2A18	59	L6	ADC 模拟电源引脚
VSS2AGND	58	P2	ADC 模拟接地引脚
VDD	4	D4	CPU 和逻辑数字电源引脚
VDD	15	D5	
VDD	23	D8	
VDD	29	D9	
VDD	61	E11	
VDD	101	F4	CPU 和逻辑数字电源引脚
VDD	109	F11	
VDD	117	H4	
VDD	126	J4	
VDD	139	J11	
VDD	146	K11	
VDD	154	L8	
VDD	167		
VDDIO	9	A13	数字 I/O 电源引脚
VDDIO	71	B1	
VDDIO	93	D7	
VDDIO	121	E4	
VDDIO	143	G4	
VDDIO	159	G11	
VDDIO	170	L10	
VDDIO		N14	
VDDIO	107	D11	数字 I/O 预驱动电源
VSS	3	A1	数字接地引脚
VSS	8	A2	
VSS	14	A14	
VSS	22	B14	
VSS	30	F6	
VSS	60	F7	
VSS	70	F8	
VSS	83	F9	
VSS	92	G6	
VSS	103	G7	
VSS	106	G8	
VSS	108	G9	
VSS	118	H6	
VSS	120	H7	

名称	引脚编号		说明
	LQFP176 PIN#	BGA176 BALL#	
VSS	125	H8	
VSS	140	H9	
VSS	144	J6	
VSS	147	J7	
VSS	155	J8	
VSS	160	J9	
VSS	166	P13	
VSS	171	P14	
<b>GPIO 和外设信号</b>			
GPI00 EPWM1A	5	D1	通用输入/输出 0 (I/O/Z) 增强型 PWM1 输出 A 和 HRPWM 通道 (O)
GPI01 EPWM1B ECAP6 MFSRB	6	D2	通用输入/输出 1 (I/O/Z) 增强型 PWM1 输出 B (O) 增强型捕捉 6 输入/输出 (I / O) McBSP-B 接收帧同步 (I / O)
GPI02 EPWM2A	7	D3	通用输入/输出 2 (I/O/Z) 增强型 PWM2 输出 A 和 HRPWM 通道 (O)
GPI03 EPWM2B ECAP5 MCLKRB	10	E1	通用输入/输出 3 (I/O/Z) 增强型 PWM2 输出 B (O) 增强型捕捉 5 输入/输出 (I / O) McBSP-B 接收帧同步 (I / O)
GPI04 EPWM3A	11	E2	通用输入/输出 4 (I/O/Z) 增强型 PWM3 输出 A 和 HRPWM 通道 (O)
GPI05 EPWM3B MFSRA ECAP1	12	E3	通用输入/输出 5 (I/O/Z) 增强型 PWM3 输出 B (O) McBSP-B 接收帧同步 (I / O) 增强型捕捉输入/输出 1 (I / O)
GPI06 EPWM4A EPWMSYNCI EPWMSNCO	13	F1	通用输入/输出 6 (I/O/Z) 增强型 PWM4 输出 A 和 HRPWM 通道 (O) 外部 ePWM 同步脉冲输入 (I) 外部 ePWM 同步脉冲输出 (O)
GPI07 EPWM4B MCLKRA ECAP2	16	F2	通用输入/输出 7 (I/O/Z) 增强型 PWM4 输出 B (O) McBSP-B 接收时钟 (I / O) 增强型捕捉输入/输出 2 (I / O)
GPI08 EPWM5A CANTXB <b>ADC SOCA0</b>	17	F3	通用输入/输出 8 (I/O/Z) 增强型 PWM5 输出 A 和 HRPWM 通道 (O) 增强型 CAN-B 传输 (O) ADC 转换启动 A (O)
GPI09 EPWM5B SCITXDB ECAP3	18	G1	通用输入/输出 9 (I/O/Z) 增强型 PWM5 输出 B (O) SCI-B 发送数据 (I/O) 增强型捕捉输入/输出 3 (I/O)
GPI010 EPWM6A CANRXB <b>ADC SOCB0</b>	19	G2	通用输入/输出 10 (I/O/Z) 增强型 PWM6 输出 A 和 HRPWM 通道 (O) 增强型 CAN-B 接收 (O) ADC 转换启动 B (O)
GPI011 EPWM6B SCIRXDB ECAP4	20	G3	通用输入/输出 11 (I/O/Z) 增强型 PWM6 输出 B (O) SCI-B 接收数据 (I) 增强型 CAP 输入/输出 4 (I/O)

名称	引脚编号		说明
	LQFP176 PIN#	BGA176 BALL#	
GPIO12 <u>TZ1</u> CANTXB MDXB	21	H1	通用输入/输出 12 (I/O/Z) 触发区输入 1 (I) 增强型 CAN-B 传输 (O) McBSP-B 串行数据传输 (O)
GPIO13 <u>TZ2</u> CANRXB MDRB	24	H2	通用输入/输出 13 (I/O/Z) 触发区输入 2 (I) 增强型 CAN-B 接收 (O) McBSP-B 串行数据接收 (O)
GPIO14  <u>TZ3/XHOLD</u>  SCITXDB MCLKXB	25	H3	通用输入/输出 14 (I/O/Z) 触发区输入 3/外部保持请求XHOLD，当有效时（低电平），请求外部接口 XINIF 释放外部总线并将所有总线和选通脉冲置于一个高阻抗状态。为阻止该事件的发生，当TZ3信号变为有效，通过写入 XINTCNF2[HOLD] = 1 来禁用此功能。如果没有这样做，XINTF 总线将在TZ3变为低电平时随时进入高阻抗状态。在 ePWM 端，TZn信号在默认情况下被忽略，除非它们由代码启用。当任一当前的访问完成并且在 XINIF 上没有等待的访问时，XINIF 将释放总线。(I) SCI-B 传输 (O) McBSP-B 传输时钟 (I/O)
GPIO15  <u>TZ4/XHOLDA</u>  SCIRXDB MFSXB	26	J1	通用输入/输出 15 (I/O/Z) 触发区输入 4/ 外部保持确认。在 GPADIR 寄存器中，此选项的引脚功能基于所选择的方向。如果此引脚被配置为输入，则TZ4功能就会被选择。如果此引脚被配置为输出，则XHOLDA功能就会被选择。当 XININ 已经准予一个XHOLD请求时，XHOLDA被驱动至有效（低电平）。所有 XINIF 总线和选通闸门将处于高阻抗状态。当XHOLD信号被释放时，XHOLDA被释放。当XHOLDA为有效（低电平）时，外部器件应该只驱动外部总线。(I/O) SCI-B 接收 (I) McBSP-B 传输帧同步 (I/O)
GPIO16 SPISIMOA CANTXB TZ5	27	J2	通用输入/输出 16 (I/O/Z) SPI 从器件输入，主器件输出(I/O) 增强型 CAN-B 发送 (O) 触发区输入 5 (I)
GPIO17 SPISOMIA CANRXB TZ6	28	J3	通用输入/输出 17 (I/O/Z) SPI-A 从器件输出，主器件输入(I/O) 增强型 CAN-B 接收 (I) 触发区输入 6 (I)
GPIO18 SPICLKA SCITXDB CANRXA	62	N8	通用输入/输出 18 (I/O/Z) SPI-A 时钟输入/输出 (I/O) SCI-B 传输(O) 增强型 CAN-A 接收 (I)
GPIO19 SPISTEA SCIRXDB CANTXA	63	M8	通用输入/输出 19 (I/O/Z) SPI-A 从器件发送使能(I/O) SCI-B 接收 (I) 增强型 CAN-A 传输 (O)
GPIO20 EQEP1A MDXA CANTXB	64	P9	通用输入/输出 20 (I/O/Z) 增强型 QEP1 输入 A (I) McBSP-A 串行数据传输 (O) 增强型 CAN-B 传输 (O)
GPIO21 EQEP1B MDRA CANRXB	65	N9	通用输入/输出 21 (I/O/Z) 增强型 QEP1 输入 B (I) McBSP-A 串行数据接收 (I) 增强型 CAN-B 接收 (I)
GPIO22 EQEP1S MCLKXA SCITXDB	66	M9	通用输入/输出 22 (I/O/Z) 增强型 QEP1 选通脉冲 (I/O) McBSP-A 传输时钟 (I/O) SCI-B 传输 (O)
GPIO23 EQEP1I MFSXA CIRXDB	67	P10	通用输入/输出 23 (I/O/Z) 增强型 QEP1 索引 (I/O) McBSP-A 传输帧同步 (I/O) SCI-B 接收 (I)

名称	引脚编号		说明
	LQFP176 PIN#	BGA176 BALL#	
GPIO24 ECAP1 EQEP2A MDXB	68	N10	通用输入/输出 24 (I/O/Z) 增强型捕获 1 (I/O) 增强型 QEP2 输入 A (I) McBSP-B 串行数据传输 (O)
GPIO25 ECAP2 EQEP2B MDRB	69	M10	通用输入/输出 25 (I/O/Z) 增强型捕获 2 (I/O) 增强型 QEP2 输入 B (I) McBSP-B 串行数据接收 (I)
GPIO26 ECAP3 EQEP2I MCLKXB	72	P11	通用输入/输出 26 (I/O/Z) 增强型捕获 3 (I/O) 增强型 QEP2 索引 (I/O) McBSP-B 传输时钟 (I/O)
GPIO27 ECAP4 EQEP2S MFSXB	73	N11	通用输入/输出 27 (I/O/Z) 增强型捕获 4 (I/O) 增强型 QEP2 选通脉中 (I/O) McBSP-B 传输帧同步 (I/O)
GPIO28 SCIRXDA XZCS6	141	D10	通用输入/输出 28 (I/O/Z) SCI 接收数据 (I) 外部接口区域 6 芯片选择 (O)
GPIO29 SCITXDA XA19	2	C1	通用输入/输出 29。(I/O/Z) SCI 传输数据 (O) 外部接口地址线路 19 (O)
GPIO30 CANRXA XA18	1	C2	通用输入/输出 30 (I/O/Z) 增强型 CAN-A 接收 (I) 外部接口地址线路 18 (O)
GPIO31 CANTXA XA17	176	B2	通用输入/输出 31 (I/O/Z) 增强型 CAN-A 传输 (I) 外部接口地址线路 17 (O)
GPIO32 SDAA EPWMSYNCI ADCSOCAO	74	M11	通用输入/输出 32 (I/O/Z) I2C 数据开漏双向端口 (I/OD) 增强型 PWM 外部同步脉冲输入 (I) ADC 转换启动 A(O)
GPIO33 SCLA EPWMSYNCO ADCSOCBO	75	P12	通用输入/输出 33 (I/O/Z) I2C 时钟开漏双向端口 (I/OD) 增强型 PWM 外部同步脉冲输出 (O) ADC 转换启动 B(O)
GPIO34 ECAP1 XREADY	142	A9	通用输入/输出 34 (I/O/Z) 增强型捕捉输入/输出 1 (I/O) 外部接口就绪信号。请注意，此引脚始终是（直接）连接到 XINTF 的。如果一个应用程序使用引脚作为 GPIO，同时还使用了 XINTF，则应配置 XINTF 来忽略就绪。
GPIO35 SCITXDA XR/W	148	B9	通用输入/输出 35 (I/O/Z) SCI 传输数据 (O) 外部接口读取，不能写入选通脉冲
GPIO36 SCIRXDA XZCS0	145	C9	通用输入/输出 36 (I/O/Z) SCI 接收数据 (I) 外部接口 0 区芯片选择 (O)
GPIO37 ECAP2 XZCS7	150	B8	通用输入/输出 37 (I/O/Z) 增强型捕获输入/输出 2 (I/O) 外部接口 7 区芯片选择 (O)
GPIO38 - XWE0	137	C10	通用输入/输出 38 (I/O/Z) - 外部接口写入时能 0 (O)
GPIO39 - XA16	175	C3	通用输入/输出 39 (I/O/Z) - 外部接口地址线路 16 (O)

名称	引脚编号		说明
	LQFP176 PIN#	BGA176 BALL#	
GPIO40 - XA0/XWE1	151	C8	通用输入/输出 40 (I/O/Z) - 外部接口地址线路 0 / 外部接口写入时能 1(O)
GPIO41 - XA1	152	A7	通用输入/输出 41 (I/O/Z) - 外部接口地址线路 1 (O)
GPIO42 - XA2	153	B7	通用输入/输出 42 (I/O/Z) - 外部接口地址线路 2 (O)
GPIO43 - XA3	156	C7	通用输入/输出 43 (I/O/Z) - 外部接口地址线路 3 (O)
GPIO44 - XA4	157	A6	通用输入/输出 44 (I/O/Z) - 外部接口地址线路 4 (O)
GPIO45 - XA5	158	B6	通用输入/输出 45 (I/O/Z) - 外部接口地址线路 5 (O)
GPIO46 - XA6	161	C6	通用输入/输出 46 (I/O/Z) - 外部接口地址线路 6 (O)
GPIO47 - XA7	162	D6	通用输入/输出 47 (I/O/Z) - 外部接口地址线路 7 (O)
GPIO48 ECAP5 XD31	88	L14	通用输入/输出 48 (I/O/Z) 增强型捕捉输入/输出 5 (I/O) 外部接口地址线路 31 (I/O/Z)
GPIO49 ECAP6 XD30	89	L13	通用输入/输出 49 (I/O/Z) 增强型捕捉输入/输出 6 (I/O) 外部接口地址线路 30 (I/O/Z)
GPIO50 EQEP1A XD29	90	L12	通用输入/输出 50 (I/O/Z) 增强型 QEP1 输入 A (I/O) 外部接口地址线路 29 (I/O/Z)
GPIO51 EQEP1B XD28	91	K14	通用输入/输出 51 (I/O/Z) 增强型 QEP1 输入 B (I) 外部接口地址线路 28 (I/O/Z)
GPIO52 EQEP1S XD27	94	K13	通用输入/输出 52 (I/O/Z) 增强型 QEP1 选通脉冲 (I/O) 外部接口地址线路 27 (I/O/Z)
GPIO53 EQEP1I XD26	95	K12	通用输入/输出 53 (I/O/Z) 增强型 QEP1 索引 (I/O) 外部接口地址线路 26 (I/O/Z)
GPIO54 SPISIMOA XD25	96	J14	通用输入/输出 54 (I/O/Z) SPI-A 从器件输入, 主器件输出 (I/O) 外部接口地址线路 25 (I/O/Z)
GPIO55 SPISOMIA XD24	97	J13	通用输入/输出 55 (I/O/Z) SPI-A 从器件输出, 主器件输入 (I/O) 外部接口地址线路 24 (I/O/Z)
GPIO56 SPICKLA XD23	98	J12	通用输入/输出 56 (I/O/Z) SPI-A 时钟 (I/O) 外部接口地址线路 23 (I/O/Z)
GPIO57 SPISTEA XD22	99	H13	通用输入/输出 57 (I/O/Z) SPI-A 从器件发送时能 (I/O) 外部接口地址线路 22 (I/O/Z)

名称	引脚编号		说明
	LQFP176 PIN#	BGA176 BALL#	
GPIO58 MCLKRA XD21	100	H12	通用输入/输出 58 (I/O/Z) McBSP-A 接收时钟 (I/O) 外部接口地址线路 21 (I/O/Z)
GPIO59 MFSRA XD20	110	H11	通用输入/输出 59 (I/O/Z) McBSP-A 接收帧同步 (I/O) 外部接口地址线路 20 (I/O/Z)
GPIO60 MCLKRB XD19	111	G12	通用输入/输出 60 (I/O/Z) McBSP-B 接收时钟 (I/O) 外部接口地址线路 19 (I/O/Z)
GPIO61 MFSRB XD18	112	F14	通用输入/输出 61 (I/O/Z) McBSP-B 接收帧同步 (I/O) 外部接口地址线路 18 (I/O/Z)
GPIO62 SCIRXDC XD17	113	F13	通用输入/输出 62 (I/O/Z) SCI-C 接收数据 (I/O) 外部接口地址线路 17 (I/O/Z)
GPIO63 SCITXDC XD16	114	F12	通用输入/输出 63 (I/O/Z) SCI-C 发送数据 (O) 外部接口地址线路 16 (I/O/Z)
GPIO64 - XD15	115	E14	通用输入/输出 64 (I/O/Z) - 外部接口地址线路 15 (O)
GPIO65 - XD14	116	E13	通用输入/输出 65 (I/O/Z) - 外部接口地址线路 14 (I/O/Z)
GPIO66 - XD13	119	E12	通用输入/输出 66 (I/O/Z) - 外部接口地址线路 13 (I/O/Z)
GPIO67 - XD12	122	D14	通用输入/输出 67 (I/O/Z) - 外部接口地址线路 12 (I/O/Z)
GPIO68 - XD11	123	D13	通用输入/输出 68 (I/O/Z) - 外部接口地址线路 11 (I/O/Z)
GPIO69 - XD10	124	D12	通用输入/输出 69 (I/O/Z) - 外部接口地址线路 10 (I/O/Z)
GPIO70 - XD9	127	C14	通用输入/输出 70 (I/O/Z) - 外部接口地址线路 9 (I/O/Z)
GPIO71 - XD8	128	C13	通用输入/输出 71 (I/O/Z) - 外部接口地址线路 8 (I/O/Z)
GPIO72 - XD7	129	B13	通用输入/输出 72 (I/O/Z) - 外部接口地址线路 7 (I/O/Z)
GPIO73 - XD6	130	A12	通用输入/输出 73 (I/O/Z) - 外部接口地址线路 6 (I/O/Z)
GPIO74 - XD5	131	B12	通用输入/输出 74 (I/O/Z) - 外部接口地址线路 5 (I/O/Z)
GPIO75 - XD4	132	C12	通用输入/输出 75 (I/O/Z) - 外部接口地址线路 4 (I/O/Z)

名称	引脚编号		说明
	LQFP176 PIN#	BGA176 BALL#	
GPIO76 - XD3	133	A11	通用输入/输出 76 (I/O/Z) - 外部接口地址线路 3 (I/O/Z)
GPIO77 - XD2	134	B11	通用输入/输出 77 (I/O/Z) - 外部接口地址线路 2 (I/O/Z)
GPIO78 - XD1	135	C11	通用输入/输出 78 (I/O/Z) - 外部接口地址线路 1 (I/O/Z)
GPIO79 - XD0	136	B10	通用输入/输出 79 (I/O/Z) - 外部接口地址线路 0 (I/O/Z)
GPIO80 - XA8	163	A5	通用输入/输出 80 (I/O/Z) - 外部接口地址线 8 (I/O/Z)
GPIO81 - XA9	164	B5	通用输入/输出 81 (I/O/Z) - 外部接口地址线 9 (I/O/Z)
GPIO82 - XA10	165	C5	通用输入/输出 82 (I/O/Z) - 外部接口地址线 10 (I/O/Z)
GPIO83 - XA11	168	A4	通用输入/输出 83 (I/O/Z) - 外部接口地址线 11 (I/O/Z)
GPIO84 - XA12	169	B4	通用输入/输出 84 (I/O/Z) - 外部接口地址线路 12 (I/O/Z)
GPIO85 - XA13	172	C4	通用输入/输出 85 (I/O/Z) - 外部接口地址线路 13 (O)
GPIO86 - XA14	173	A3	通用输入/输出 86 (I/O/Z) - 外部接口地址线路 14 (O)
GPIO87 - XA15	174	B3	通用输入/输出 87 (I/O/Z) - 外部接口地址线路 15 (O)
XDR	149	A8	外部接口读取时能

(1) I = 输入, O = 输出, Z = 高阻抗, OD = 开漏



### 3 功能概述

#### 3.1 方框图

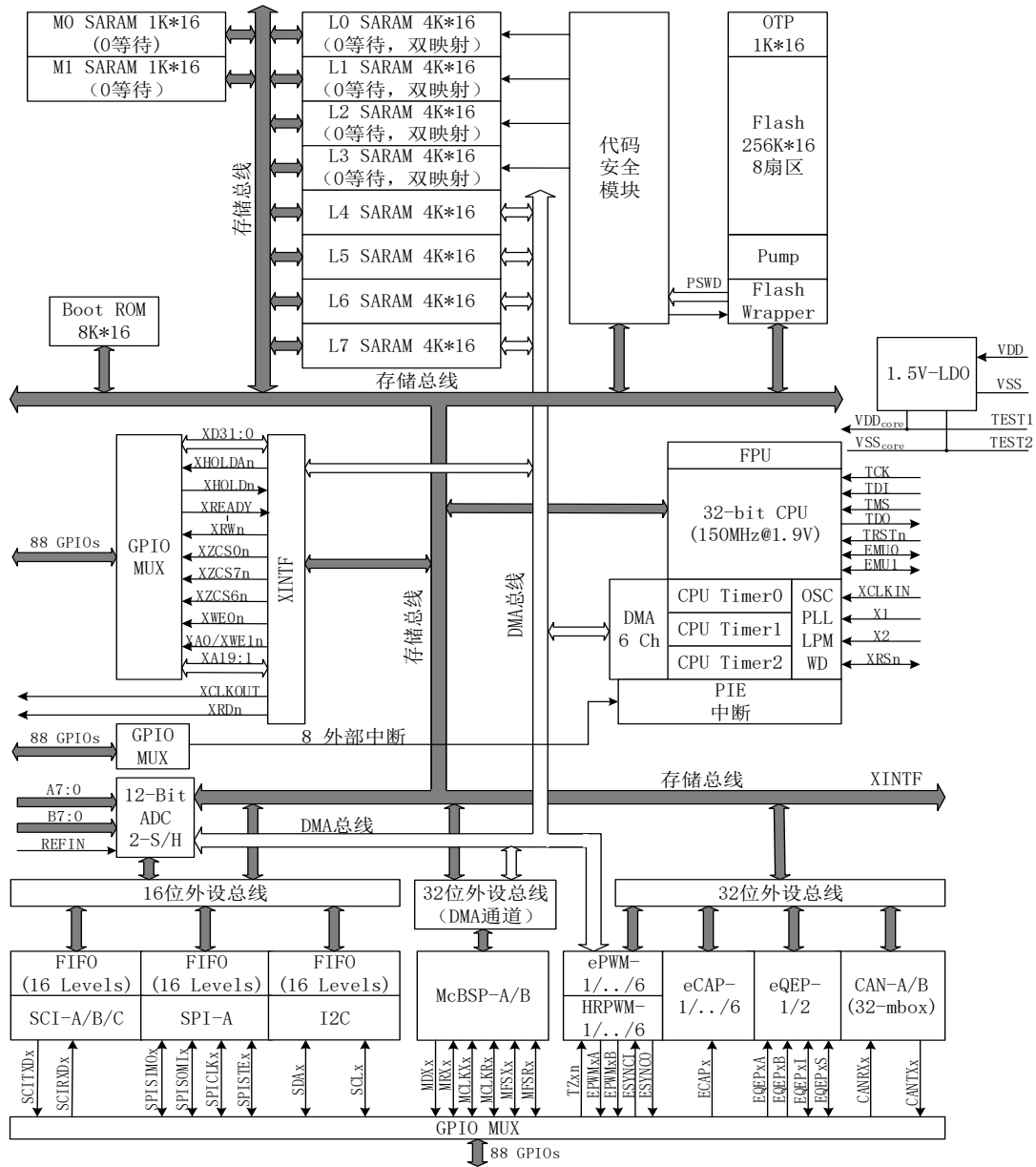


图 3-1 功能方框图

## 3.2 内存映射

从图 3-2，采用的命名规则如下：

- 内存块不可升级。
- 外设帧 0，外设帧 1，外设帧 2 和外设帧 3 内存映射只限于数据内存。用户程序不能访问这些处于程序空间内的内存映射。
- 受保护意味着“写后读操作”的顺序被保存，而不是保存流水线顺序。
- 特定内存区域受 EALLOW 保护以防止配置之后的假写入。
- 位置 0x380080-0x38008F 包含 ADC 校准程序。它不由用户编程。
- 如果 eCAN 模块未在应用中使用，提供的 RAM（LAM，MOTS，MOTO 和邮箱 RAM）可被用作通用 RAM。为实现这一功能 CAN 模块时钟应被启用。

片内存储器		片外存储器 (XINTF)	
Data Space	Prog Space	Data Space	Prog Space
0x00 0000	M0 Vector RAM (32 x 32) (Enabled if VMAP = 0)	M0 Vector RAM (Enabled if VMAP = 0)	
0x00 0040	M0 SARAM (1K x 16, 0-Wait)	M0 SARAM (1K x 16, 0-Wait)	
0x00 0400	M1 SARAM (1K x 16, 0-Wait)	M1 SARAM (1K x 16, 0-Wait)	
0x00 0800	Peripheral Frame 0	Reserved	
0x00 0D00	PIE Vector - RAM (256 x 16) (Enabled if VMAP = 1, ENPIE = 1)		
0x00 0E00	Peripheral Frame 0		
0x00 2000	Reserved	XINTF Zone 0 (4K x 16, XZCS0) (Protected) DMA-Accessible	
0x00 5000	Peripheral Frame 1 (1K x 16, Protected)	Reserved	
0x00 6000	Peripheral Frame 3 (1.5K x 16, Protected)		
0x00 7000	Peripheral Frame 1 (1.5K x 16, Protected)		
0x00 8000	L0 SARAM (4K x 16) (Secure Zone, Dual-Mapped)	Reserved	
0x00 9000	L1 SARAM (4K x 16) (Secure Zone, Dual-Mapped)		
0x00 A000	L2 SARAM (4K x 16) (Secure Zone, Dual-Mapped)		
0x00 B000	L3 SARAM (4K x 16) (Secure Zone, Dual-Mapped)		
0x00 C000	L4 SARAM (4K x 16) (DMA-Accessible)		
0x00 D000	L5 SARAM (4K x 16) (DMA-Accessible)		
0x00 E000	L6 SARAM (4K x 16) (DMA-Accessible)		
0x00 F000	L7 SARAM (4K x 16) (DMA-Accessible)		
0x01 0000	Reserved	XINTF Zone 6 (1M x 16, XZCS6) DMA-Accessible	
0x30 0000	FLASH(256K x 16,Secure Zone)	XINTF Zone 7 (1M x 16, XZCS7) DMA-Accessible	
0x33 FFF8	128-bit Password	Reserved	
0x34 0000	Reserved		
0x38 0080	ADC Calibration Data		
0x38 0090	Reserved		
0x38 0400	User OTP(1K x 16,Secure Zone)		
0x38 0800	Reserved		
0x3F 8000	L0 SARAM (4K x 16) (Secure Zone, Dual-Mapped)		
0x3F 9000	L1 SARAM (4K x 16) (Secure Zone, Dual-Mapped)		
0x3F A000	L2 SARAM (4K x 16) (Secure Zone, Dual-Mapped)		
0x3F B000	L3 SARAM (4K x 16) (Secure Zone, Dual-Mapped)		
0x3F C000	Reserved		
0x3F E000	Boot ROM (8K x 16)		
0x3F FFC0	BROM Vector-ROM (32 x 32) (Enabled if VMAP=1,ENPIE=0)		

注：一次只能启用这些向量映射中的一个：M0向量、PIE向量、BPOM向量。

图 3-2 AVP32F335 内存映射

**表 3-1 AVP32F335 中闪存扇区的地址**

地址范围	程序和数据空间
0x30 0000-0x30 7FFF	扇区 H (32K x 16)
0x30 8000-0x30 FFFF	扇区 G (32K x 16)
0x31 0000-0x31 7FFF	扇区 F (32K x 16)
0x31 8000-0x31 FFFF	扇区 E (32K x 16)
0x32 0000-0x32 7FFF	扇区 D (32K x 16)
0x32 8000-0x32 FFFF	扇区 C (32K x 16)
0x33 0000-0x33 7FFF	扇区 B (32K x 16)
0x33 8000-0x33 FF7F	扇区 A (32K x 16)
0x33 FF80-0x33 FFF5	当使用代码安全模块时，编程至 0x0000
0x33 FFF6-0x33 FFF7	引导至闪存进入点（程序分支指令）
0x33 FFF8-0x33 FFFF	安全密码（128 位）（不要设定为零）

**注**

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

[表 3-2](#) 显示如何处理这些内存地址。

**表 3-2 使用安全代码模块的影响**

地址范围	闪存	
	代码安全被启用	代码安全被禁用
0x33FF80-0x33FFEf	用 0x0000 填充	应用代码和数据。
0x33FFF0-0x33FFF5		只为数据保留。

外设帧 1，外设帧 2 以及 XINTF 区域 1 为一组，成为“受保护的写入/读取外设块”。“受保护”模式确保对这些所有的访问与文档中描述的一致。由于 ADP32Fx 的流水线，在对不同内存位置读取之前的写入操作将以倒序的方式出现在 CPU 内存总线上。这会导致特定外设应用中用户认为写入会首先发生（如文档所描述的那样）。ADP32Fx CPU 支持一个块保护模式，在这个模式中，可对一个内存区域进行保护，以确保操作按照本文档所描述的那样进行（代价是会增加额外的周期以校正运行）。可对这个模式进行编程，并且默认情况下，它将保护所选的区域。

针对内存映射区域内不同空间的等待状态列在[表 3-3](#)中。

**表 3-3 等待状态**

区域 (AREA)	等待状态(CPU)	等待状态 ( DMA ) (1)	注释
<b>M0 和 M1 SARAM</b>	0 - 等待		固定的
<b>外设帧 0</b>	0 - 等待 ( 写入 ) 1 - 等待 ( 读取 )	0 - 等待 ( 读取 ) 无权限 ( 写入 )	
<b>外设帧 1</b>	0 - 等待 ( 写入 ) 2 - 等待 ( 读取 )	无访问	周期可由已生成的外设扩展。 到外设帧 1 寄存器的背靠背写入操作将生成一个 1 周期停止 ( 1 周期延迟 )。
<b>外设帧 2</b>	0 - 等待 ( 写入 ) 2 - 写入 ( 读取 )	无访问	固定的 周期不可由外设扩展。
<b>外设帧 3</b>	0 - 等待 ( 写入 ) 2 - 等待 ( 读取 )	0 - 等待 ( 写入 ) 1 - 等待 ( 读取 )	假定 CPU 和 CLA 间无冲突。
<b>L0 SARAM</b>	0 - 等待	无访问	假定没有 CPU 冲突
<b>L1 SARAM</b>			
<b>L2 SARAM</b>			
<b>L3 SARAM</b>			
<b>L4 SARAM</b>	0 - 等待数据 ( 读取 )	0 - 等待	假设 CPU 和 DMA 之间没有冲突。
<b>L5 SARAM</b>	0 - 等待数据 ( 写入 )		
<b>L6 SARAM</b>	1 - 等待项目 ( 读取 )		
<b>L7 SARAM</b>	1 - 等待项目 ( 写入 )		
<b>XINTF</b>	可编程  0 - 写入缓冲器启用的 最小写入等待	可编程  0 - 写入缓冲器启用的 最小写入等待	通过由 XTIMING 寄存器设定或通过外部 XREADY 信号扩展, 来满足系统的时序要求。 对于在 XINTF 上的读取和写入, 1-等待是在外部波形上的最小等待状态。 0 - 假定写入缓冲器启用并且不满针对写入的最小等待。 假设 CPU 和 DMA 之间没有冲突。当 DMA 和 CPU 尝试同步冲突时, 加入 1 周期延迟用于仲裁。
<b>OTP</b>	可编程 1 - 等待最小	无访问	由闪存寄存器设定。 1 - 等待是等待状态所允许的最小数。可在一个减少的 CPU 频率上执行 1 等待状态操作。
<b>闪存</b>	可编程 0 - 页式等待最小值 1 - 随机等待最小值 随机等待 ≥ 页式等待	无访问	由闪存寄存器设定。 页式访问中不允许 0 - 等待最小
<b>闪存密码</b>	16=等待固定	无访问	密码位置的等待状态是固定的。
<b>引导 - ROM</b>	1 - 等待	无访问	0 - 等待速度无法实现。

## 3.3 简要说明

### 3.3.1 AVP32F335 DSP

此 AVP32F335 (ADP32x+FPU) 系列都属于 ADP32Fx 数字信号控制器 (DSP) 平台。基于 ADP32x+FPU 的控制器和 Advchip 现有的 ADP32Fx DSP 具有相同的 32 位定点架构，但是还包括一个单精度 (32 位) 的 IEEE754 浮点单元 (FPU)。这是一个非常高效的 C/C++ 引擎，它能使用户用高层次的语言开发他们的系统控制软件，能够使用 C/C++ 开发算术算法。此器件同样可以处理原由微控制器处理的系统控制任务，故在处理 DSP 算术任务时和处理系统控制任务时同样有效。此高效率处理可以节省很多系统对第二个处理器的需求。内置的 32x32 位 MAC64 位处理能力使得控制器能够有效地处理更高的数字分辨率运算问题。带有关键寄存器自动环境保存的快速中断响应，能够让一个器件用最小的延迟处理很多异步事件。还内置有一个具有流水线式存储器访问的 8 级深度受保护的流水线。该流水线式操作使得此器件能够在高速执行而无需求助于昂贵的高速存储器。特别分支超前硬件大大减少了条件不连续而带来的延迟。特别存储条件操作进一步提升了性能。

### 3.3.2 内存总线 (哈佛总线架构)

与很多 DSP 类型器件一样，多总线被用于在内存和外设以及 CPU 之间传输数据。ADP32Fx 内存总线架构包含：程序读取总线、数据读取总线和数据写入总线。此程序读取总线由 22 条地址线路和 32 条数据线路组成。数据读取和写入总线由 32 条地址线路和 32 条数据线路组成。32 位宽数据总线可实现单周期 32 位运行。多总线结构通常称为“哈佛总线”，使得 ADP32Fx 能够在单周期内取一个指令、读取一个数据值和写入一个数据值。所有连接在内存总线上的外设和内存对内存访问进行优先级设定。总的来说，内存总线访问的优先级可概括如下：

**最高级：**数据写入 (内存总线上不能同时进行数据和程序写入。)

程序写入 (内存总线上不能同时进行数据和程序写入。)

数据读取

程序读取 (内存总线上不能同时进行程序读取和取指令。)

**最低级：**取指令 (内存总线上不能同时进行程序读取和取指令。)

### 3.3.3 外设总线

为了实现不同进芯电子 (Advchip) ADP32Fx 系列器件采用了一个针对外设互连的外设总线标准。外设总线桥复用了多种总线，此总线将处理器“内存总线”组装进一个由 16 条地址线路和 16 条或者 32 条数据线路以及相关控制信号组成的单总线中。在 ADP32Fx 器件上支持三个版本的外设总线。一个版本只支持 16 位访问 (被称为外设帧 2) 并且这个版本保留了与 ADP16 外设的兼容性。另外一个版本支持 16 位和 32 位访问 (被称为外设帧 1)。第三个版本支持 DMA 访问和 16 位以及 32 位访问 (被称为外设

帧 3 )。

### 3.3.4 实时 JTAG 和分析

AVP32F335 器件使用标准的 IEEE1149.1 JTAG 接口。该器件支持实时运行模式，在处理器正在运行、执行代码并且处理中断时，可修改存储器内容、外设和寄存器位置。用户也可以通过非时间关键代码进行单步操作，同时可在没有干扰的情况下启用即将被处理的时间关键中断。此器件在 CPU 的硬件内执行实时模式。这是 AVP32F335 器件的独特功能，无需软件监控。此外，还提供了特别分析硬件以实现硬件断点或者数据/地址观察点的设置并当一个匹配发生时生成不同的用户可选中断事件。

### 3.3.5 外部接口 ( XINTF )

该异步接口由 20 位地址总线，32 位数据总线，和 3 个芯片选信号组成。此芯片选择线路被映射到 3 个外部区域，即区域 0、6 和 7。3 个区域中的每个区域可被设定为不同的等待状态数量、选通信号设置和保持时序，并且每个区域可被外部设定为扩展等待状态或者没有扩展等待状态。设定的等待状态、芯片选择和可编程选通时序可实现到外部存储器和外设的无缝对接。

### 3.3.6 闪存

AVP32F335 器件包含 256Kx16 的嵌入式闪存存储器，被分别放置在 8 个 32Kx16 扇区内。所有器件还包含一个单 1Kx16OTP 内存，其地址范围为 0x380400-0x3807FF。用户能够在不改变其它扇区的同时单独擦除、编辑和验证一个闪存扇区。但是不能使用闪存的一个扇区或者这个 OTP 来执行擦除/编辑其它扇区的闪存算法。闪存模块内置的特殊内存流水线操作具有更高的性能。闪存/OTP 被映射到程序和数据空间，它可被用于执行代码或者存储数据信息。但是地址 0x33FFF0-0x33FFF5 保留为数据变量，且不应包含程序代码。

---

#### 注

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

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

---

### 3.3.7 M0 , M1 SARAM

所有 AVP32F335 器件包含这两块单周期访问内存，每一块 SRAM 的大小为 1K x 16。复位时堆栈指针指向块 M1 的开始位置。M0 和 M1 块与所有其它 ADP32Fx 器件上的内存块一样被映射到程序和数据空间。用户能够使用 M0 和 M1 来执行代码或者存储数据变量。分区在连接器内执行。ADP32Fx 器件提供了用高级语言编程变得更加容易的到编程器的统一内存映射。

### 3.3.8 L0 , L1 , L2 , L3 , L4 , L5 , L6 , L7 SARAM

AVP32F335 器件含有 32K x 16 的单周期访问 RAM。每个被分为 8 个区块 ( L0–L7 且每块 4K )。每个块可被独立访问大大减少了 CPU 管线延迟。这些块被映射到程序和数据空间。L4 , L5 , L6 , L7 可由 DMA 访问。

### 3.3.9 引导 ROM

引导 ROM 由厂家使用引导载入软件进行设定。引导 ROM 程序在器件复位后，检查四个 GPIO 引脚状态后确认芯片进入相应的引导模式。例如，用户可以选择执行已经出现在内部闪存中的代码或者通过几个串行端口中的一个将全新的程序下载至内部 RAM。引导 ROM 还包含用于数学相关算法中的标准表，例如 SIN/COS 波形。

**表 3-4 引导模式选择**

模式	GPIO87/XA15	GPIO86/XA14	GPIO85/XA13	GPIO84/XA12	模式 <sup>(1)</sup>
F	1	1	1	1	跳转到闪存
E	1	1	1	0	SCI-A boot
D	1	1	0	1	SPI-A 引导
C	1	1	0	0	I2C-A 引导
B	1	0	1	1	eCAN-A 引导
A	1	0	1	0	McBSP-A 引导
9	1	0	0	1	跳转到 XINTF x16
8	1	0	0	0	跳转到 XINTF x32
7	0	1	1	1	跳转到 OTP
6	0	1	1	0	并行 GPIO I/O 引导
5	0	1	0	1	并行 XINTF 引导
4	0	1	0	0	跳转至 SARAM
3	0	0	1	1	分支到检查引导模式
2	0	0	1	0	跳转到闪存, 跳过 ADC 校准
1	0	0	0	1	跳转至 SARAM, 跳过 ADC 校准
0	0	0	0	0	跳转至 SCI, 跳过 ADC 校准

(1) 所有的 4 个 GPIO 引脚都有内部上拉电阻

**注**

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



### 3.3.9.1 引导加载器使用的外设引脚

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

**表 3-5 外设引导加载引脚**

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

### 3.3.10 安全性

ADP32Fx 支持高级别安全以保护用户固件不受逆向工程损坏。该加密装置有一个 128 位密码（针对 16 个等待状态的硬编码），此密码由用户编辑写入闪存。一个代码安全模块（CSM）被用于保护闪存/OTP 和 L0/L1/L2/L3 SARAM 块。这个安全特性防止未经授权的用户通过 JTAG 端口检查内存内容，从外部内存执行代码或者试图引导加载一些将会输出安全内存内容的恶意软件。为了启用到安全块的访问，用户必须写入与存储在闪存/ROM 密码位置内的值相匹配的正确的 128 位“KEY（密钥）”值。

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

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

1. 首先是使用在等待中复位的仿真模式，该模式将保持器件在复位状态直到仿真器获得控制权。仿真器必须支持此选项的这种模式。

2. 第二种选择是使用“分支至检查引导模式”引导选项。这将进入一个环路，并不断轮询引导模式选择引脚。通过重新映射PC到另一个地址，或通过把引导模式选择引脚更改为所需的引导模式，用户可以选择此引导模式，然后在仿真器被连接时退出这种模式。

---

**注**

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

**代码安全模块免责声明**

- 此器件所包含的代码安全模块（CSM）被设计用于对存储在相关内存（ROM 或者闪存）中的数据进行密码保护并且由原厂提供质量保证，与其标准条款和条件相一致，符合原厂发布的规范以获得适用于这个器件的保修期。
  - 但是，原厂不保证或表示CSM不会被损坏或破坏，或不能通过其它方法存取关联的存储器中存储的数据。而且，除了上述内容外，原厂也未对本器件的CSM或操作做任何保证或表示，包括任何隐含的用于特定用途的商用性或适用性保证。
  - 在任何情况下，原厂对以任何方法使用CSM或本器件产生的任何必然、特殊、间接、偶然或严重伤害不负任何责任，无论原厂是否被告知存在这种伤害的可能性。排除的损害包括但不限于数据丢失、信誉损失、无法使用、业务中断或其它经济损失。
- 

### 3.3.11 外设中断扩展 (PIE) 块

PIE 块将许多中断源复用至中断输入的较小的集合中。PIE 块能够支持多达 96 个外设中断。在 AVP32F335 中，96 中断中的 58 个被外设使用。96 个中断被分成 8 组，每组被提供 12 个 CPU 中断线（INT1 或者 INT12）中的 1 个。96 个中断中的每一个中断由其存储在一个可被用户写覆盖的专用 RAM 块中的矢量支持。在处理这个中断时，这个矢量由 CPU 自动抽取。抽取这个矢量以及保存关键 CPU 寄存器将花费 8 个 CPU 时钟周期。因此 CPU 能够对中断事件作出快速响应。可以通过硬件和软件控制中断的优先级。每个中断都可以在 PIE 块内启用或禁用。

### 3.3.12 外部中断 (XINT1-XINT7, XNMI)

AVP32F335支持8个可屏蔽的外部中断(XINT1-XINT7, XNMI)。XNMI可被连接至INT13或者CPU的NMI中断。这些中断中的每一个可被选择用于负边沿、正边沿或者正负边沿触发，并且可被启用或禁用（包括XNMI在内）。XINT1,XINT2,和XNMI还包含一个16位自由运行的上数计数器，当检测到一个有效的中断边沿时，该计数器复位为0。这个计数器可被用于为中断精确计时。XINT1, XINT2和XINT中断可接受来自GPIO0-GPIO31引脚的输入。XINT3-XINT7中断可接受来自GPIO32-GPIO63引脚的输入。

### 3.3.13 振荡器和锁相环 (PLL)

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

### 3.3.14 看门狗

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

### 3.3.15 外设时钟

在外设闲置时，到每一个独立外设的时钟可被启用/禁用以减少功耗。此外，到串行端口（除了 I<sup>2</sup>C和 eCAN）和ADC的系统时钟可按照 CPU 时钟进行缩放。

### 3.3.16 低功耗模式

AVP32F335 器件是完全静态 CMOS 器件。提供三个低功耗模式：

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

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

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

### 3.3.17 外设帧 0, 1, 2, 3 (PFn)

此器件将外设分成四个部分。外设映射如下：

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

- XINT : 外部中断寄存器
- PF3 : McBSP : 多通道缓冲串行端口寄存器
- ePWM : 增强型脉冲宽度调制器模块和寄存器

### 3.3.18 通用输入/输出(GPIO)复用器

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

### 3.3.19 32 位 CPU 定时器 (0, 1, 2)

CPU 定时器 0、1 和 2 是完全一样的 32 位定时器，这些定时器带有可预先设定的周期和 16 位时钟预分频。此定时器有一个 32 位倒计时寄存器，此寄存器在计数器达到 0 时生成一个中断。这个计数器的减量为被预分频值设置所分频的 CPU 时钟速度的值。当此计数器达到 0 时，它自动重新载入一个 32 位的周期值。CPU 定时器 2 为 DSP/BIOS 预留，并且连接到 CPU 的 INT14。如果 DSP/BIOS 未被使用，CPU 定时器 2 也可作为通用定时器使用。CPU 定时器 0 也为通用定时器并被连接至 PIE 模块。

### 3.3.20 控制外设

AVP32F335 支持以下用于嵌入式控制和通信的外设：

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

### 3.3.21 串行端口外设

此器件支持下列的串行通信外设：

<b>eCAN :</b>	这是 CAN 外设的增强型版本。它支持 32 个邮箱、消息时间戳并与 CAN2.0B 兼容。
<b>McBSP:</b>	多通道缓冲串行端口(McBSP)连接到 E1/T1 线路、语音质量编解码器以实现最新应用或者高质量立体声音频 DAC 器件。McBSP 接收和发送寄存器由 DMA 支持

	以大大减少处理这个外设所用的资源。如果需要，每一个 McBSP 模块可被配置为一个 SPI。
<b>SPI :</b>	SPI 是一个高速、同步串行 I/O 端口，此端口可在设定的位传输速率上将一个设定长度（1 至 16 位）的串行比特流移入和移出器件。通常，SPI 用于 DSP 和外部外设或者其它处理器之间的通信。典型应用包括外部 I/O 或者从诸如移位寄存器、显示驱动器、和 ADC 等器件的外设扩展。多器件通信由 SPI 的主控/受控操作支持。在 AVP32F335 上，SPI 包含一个 16 级接收和发送 FIFO 来减少中断处理开销。
<b>SCI :</b>	串行通信接口是一个两线制异步串行端口，通常被称为 UART。SCI 包含一个用于减少中断处理开销的 16 级接收和发送 FIFO。
<b>I2C :</b>	内部集成电路 (I2C) 模块提供一个 MCU 和其它器件（符合飞利浦半导体内部 IC 总线(I2C-bus)规范版本 2.1 并由一个 I2C-bus 相连）间的接口。通过这个 I2C 模块，连接在这个两线制总线上的外部组件能够发送 8 位数据到 MCU 或者从 MCU 接收 8 位数据。I2C 包含一个用于减少中断处理开销的 16 级接收和发送 FIFO。

### 3.4 寄存器映射

此器件包含 4 个外设寄存器空间。这些空间分类如下：

外设帧 0：这些是直接映射到 CPU 内存总线的外设。请参阅[表 3-6](#)。

外设帧 1：这些是映射到 32 位外设总线的外设。请参阅[表 3-7](#)。

外设帧 2：这些是映射到 16 位外设总线的外设。请参阅[表 3-8](#)。

外设帧 3：这些是映射到 32 位外设总线并可由 DMA 访问的外设。请参阅[表 3-9](#)。

**表 3-6 外设帧 0 寄存器<sup>(1)</sup>**

名称	地址范围	大小 (x16)	受 EALLOW 保护 <sup>(2)</sup>
器件仿真寄存器	0x00 0880-0x00 09FF	384	受 EALLOW 保护
闪存寄存器 <sup>(3)</sup>	0x00 0A80-0x00 0ADF	96	受 EALLOW 保护
代码安全模块寄存器	0x00 0AE0-0x00 0AEF	16	受 EALLOW 保护
ADC 寄存器（双映射） 0 等待(DMA)，1 个等待(CPU)，只读	0x00 0B00-0x00 0B0F	16	不受 EALLOW 保护
XINTF 寄存器	0x00 0B20-0x00 0B3F	32	受 EALLOW 保护
CPU 定时器0，CPU 定时器1，CPU 定时器2 寄存器	0x00 0C00-0x00 0C3F	64	不受 EALLOW 保护
PIE 寄存器	0x00 0CE0-0x00 0CFF	32	不受 EALLOW 保护
PIE 向量表	0x00 0D00-0x00 0DFF	256	受 EALLOW 保护
DMA 寄存器	0x00 1000-0x00 11FF	512	受 EALLOW 保护

(1) 在帧 0 中的寄存器支持 16 位和 32 位访问。

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

(3) 闪存寄存器也受到代码安全模块(CSM)的保护。

**表 3-7 外设帧 1 寄存器**

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

**表 3-8 外设帧 2 寄存器**

名称	地址范围	大小 (x16)
系统控制寄存器	0x00 7010-0x00 702F	32
SPI-A 寄存器	0x00 7040-0x00 704F	16
SCI-A 寄存器	0x00 7050-0x00 705F	16
外部中断寄存器	0x00 7070-0x00 707F	16
ADC 寄存器	0x00 7100-0x00 711F	32
SCI-B 寄存器	0x00 7750-0x00 775F	16
SCI-C 寄存器	0x00 7770-0x00 777F	16
I2C-A 寄存器	0x00 7900-0x00 793F	64

**表 3-9 外设帧 3 寄存器**

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

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

**表 3-10 新增寄存器**

名称	地址范围	大小 (x16)
LDC+比较器寄存器	0x7012-0x7015	4
负压选择+内部 LDO 寄存器	0x7016	1
LDC 寄存器	0x7018	1

### 3.5 器件仿真寄存器

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

**表 3-11 器件仿真寄存器**

名称	地址范围	大小 (x16)	说明
<b>DEVICECNF</b>	0x0880 0x0881	2	器件配置寄存器
<b>PARTID</b>	0x380090	1	部件 ID 寄存器 AVP32F335 0x00EF
<b>CLASSID</b>	0x0882	1	AVP32F335 浮点类设备 AVP32F335 0x00EF
<b>REVID</b>	0x0883	1	修订版本 ID 寄存器
<b>PROTSTART</b>	0x0884	1	区块保护起始地址寄存器
<b>PROTRANGE</b>	0x0885	1	区块保护范围地址寄存器

### 3.6 中断

图 3-3 显示了不同的中断源是如何被复用的。

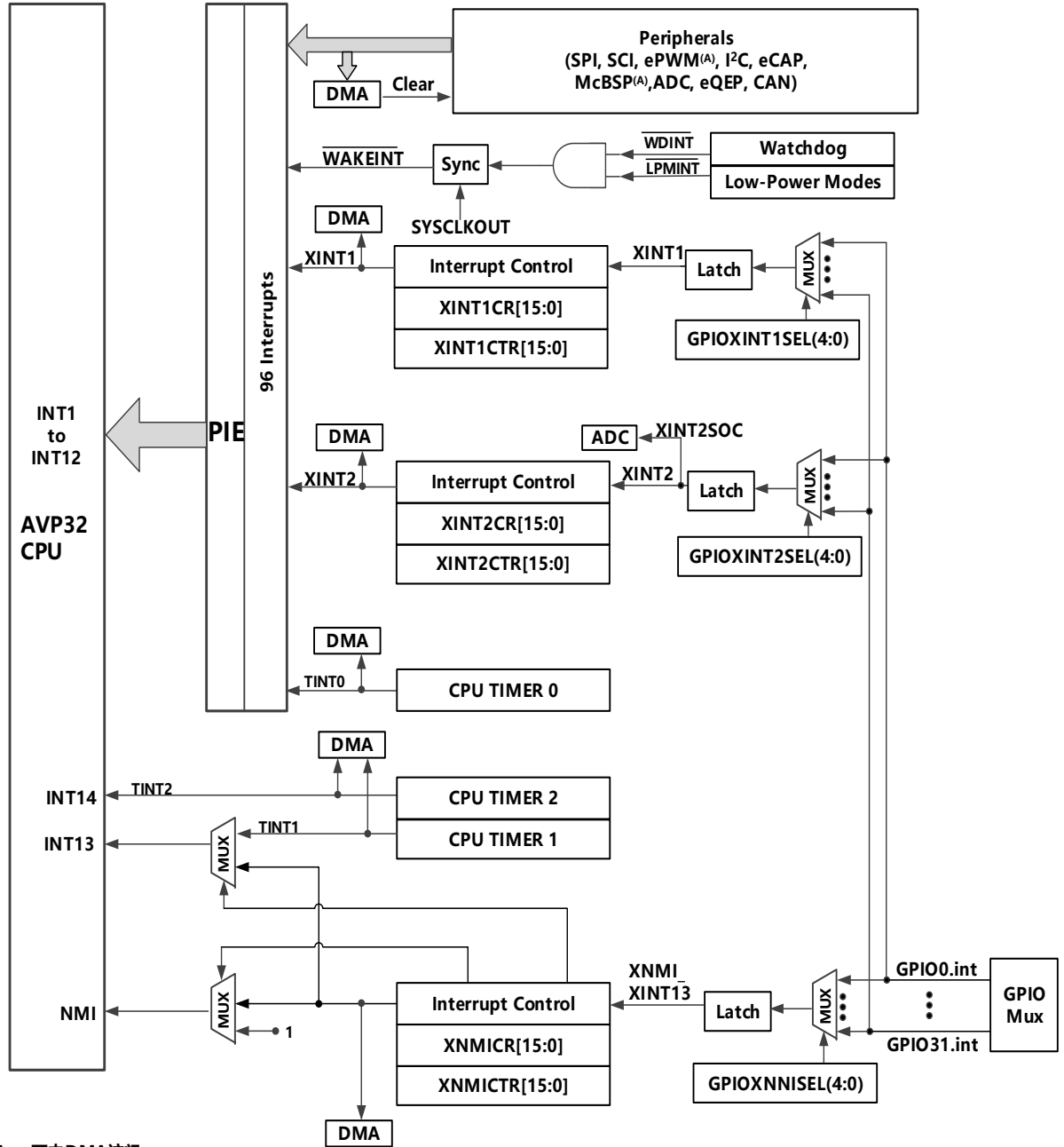


图 3-3 外部和 PIE 中断源



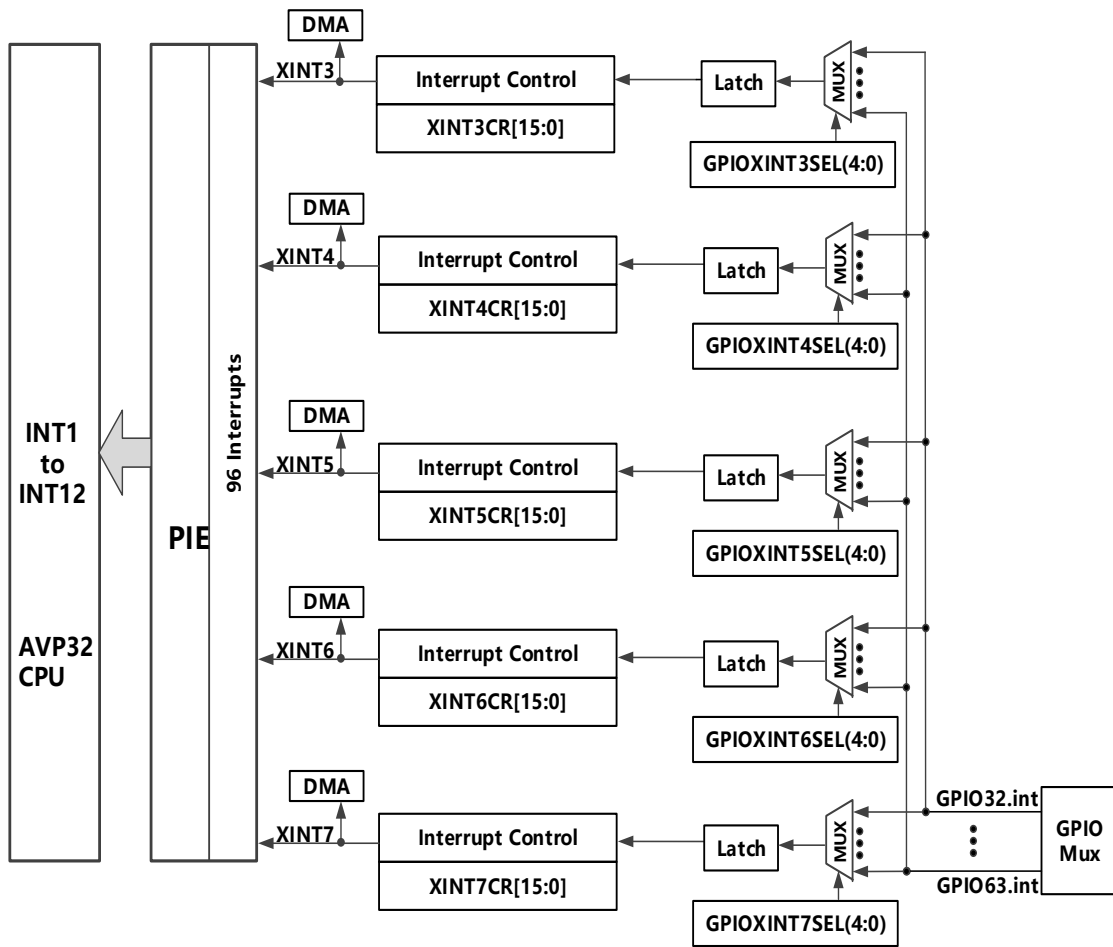


图 3-4 外部中断

8 个 PIE 块中断被组合进一个 CPU 中断中。12 个 CPU 中断组每个有 8 个中断相当于 96 个可能中断。表 3-11 显示了 AVP32F335 器件所用的中断。

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

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

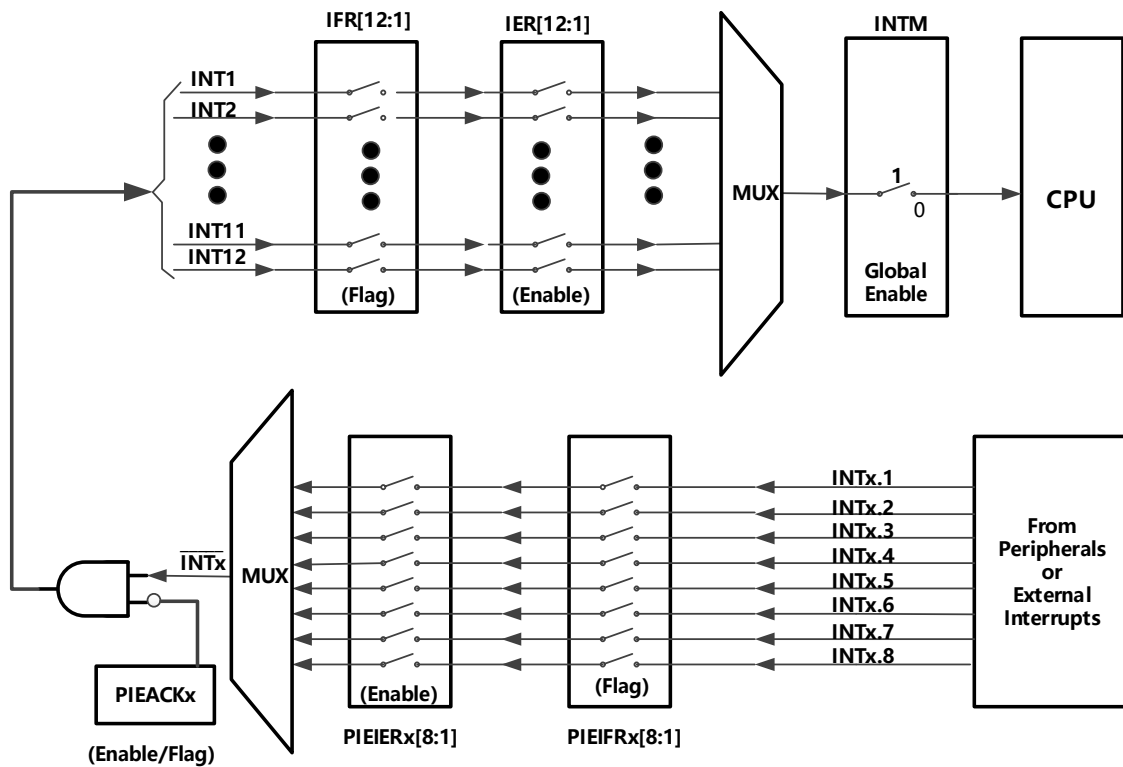


图 3-5 使用 PIE 块的中断复用

表 3-12 PIE 外设中断矢量表<sup>(1)</sup>

	INTx.8	INTx.7	INTx.6	INTx.5	INTx.4	INTx.3	INTx.2	INTx.1
INT1	WAKEINT (LPM/WD)	TINT0 (定时器0)	ADCINT (ADC)	XINT2	XINT1	被保留	SEQ2INT (ADC)	SEQ1INT (ADC)
INT2	被保留	被保留	EPWM6_TZINT (ePWM6)	EPWM5_TZINT (ePWM5)	EPWM4_TZINT (ePWM4)	EPWM3_TZINT (ePWM3)	EPWM2_TZINT (ePWM2)	EPWM1_TZINT (ePWM1)
INT3	被保留	被保留	EPWM6_INT (ePWM6)	EPWM5_INT (ePWM5)	EPWM4_INT (ePWM4)	EPWM3_INT (ePWM3)	EPWM2_INT (ePWM2)	EPWM1_INT (ePWM1)
INT4	被保留	被保留	ECAP6_INT (eCAP6)	ECAP5_INT (eCAP5)	ECAP4_INT (eCAP4)	ECAP3_INT (eCAP3)	ECAP2_INT (eCAP2)	ECAP1_INT (eCAP1)
INT5	被保留	被保留	被保留	被保留	被保留	被保留	EQEP2_INT (eQEP2)	EQEP1_INT (eQEP1)
INT6	被保留	被保留	MXINTA (McBSP-A)	MRINTA (McBSP-A)	MXINTB (McBSP-B)	MRINTB (McBSP-B)	SPITXINTA (SPI-A)	SPIRXINTA (SPI-A)
INT7	被保留	被保留	DINTCH6 (DMA)	DINTCH5 (DMA)	DINTCH4 (DMA)	DINTCH3 (DMA)	DINTCH2 (DMA)	DINTCH1 (DMA)
INT8	被保留	被保留	SCITXINTC (SCI-C)	SCIRXINTC (SCI-C)	被保留	被保留	I2CINT2A (I2C-A)	I2CINT1A (I2C-A)
INT9	ECAN1_INTB (CAN-B)	ECAN0_INTB (CAN-B)	ECAN1_INTA (CAN-A)	ECAN0_INTA (CAN-A)	SCITXINTB (SCI-B)	SCIRXINTB (SCI-B)	SCITXINTA (SCI-A)	SCIRXINTA (SCI-A)
INT10	被保留	被保留	被保留	被保留	被保留	被保留	被保留	被保留
INT11	被保留	被保留	被保留	被保留	被保留	被保留	被保留	被保留
INT12	LUF (FPU)	LVF (FPU)	被保留	INT7	XINT6	XINT5	XINT4	XINT3

(1) 在 96 个可能的中断中，有一些是不使用的。这些中断是为以后的器件所保留的。如果它们在 PIEIFR<sub>x</sub> 级被启用并且这个组中的中断没有一个被外设使用，这些中断可被用作软件中断。否则，在意外地清除它们的标志同时修改 PIEIFR 的情况下，来自外设的中断也许会丢失。总的来说，在两个安全情况下，被保留的中断可被用作软件中断：

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

**表 3-13 PIE 配置和控制寄存器**

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

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

### 3.6.1 外部中断

**表 3-14 外部中断寄存器**

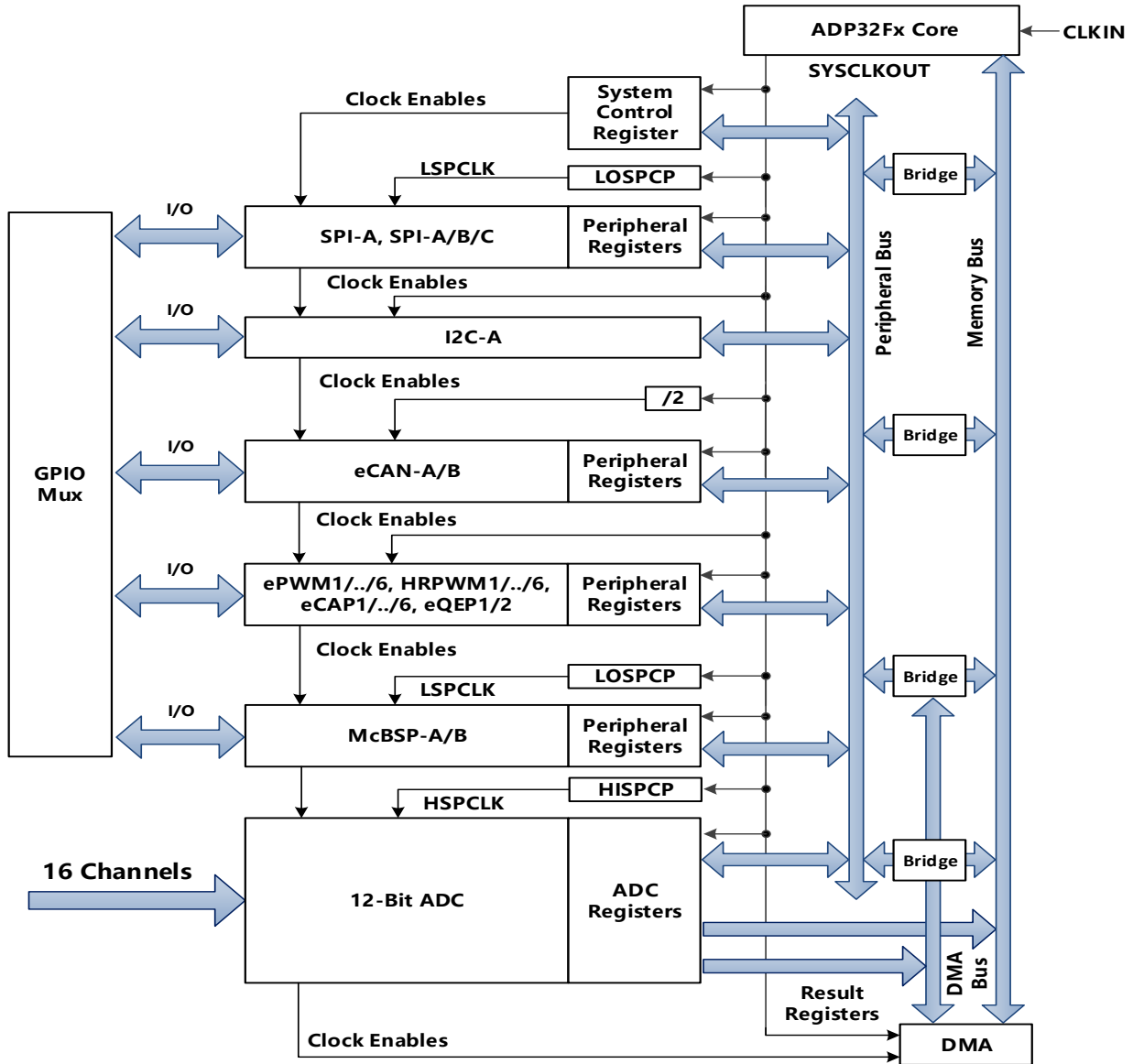
名称	地址	大小 (X 16)	说明
XINT1CR	0x00 7070	1	XINT1 配置寄存器
XINT2CR	0x00 7071	1	XINT2 配置寄存器
XINT3CR	0x00 7072	1	XINT3 配置寄存器
XINT4CR	0x00 7073	1	XINT4 配置寄存器

<b>XINT5CR</b>	0x00 7074	1	XINT5 配置寄存器
<b>XINT6CR</b>	0x00 7075	1	XINT6 配置寄存器
<b>XINT7CR</b>	0x00 7076	1	XINT7 配置寄存器
<b>XNMICR</b>	0x00 7077	1	XNMI 配置寄存器
<b>XINT1CTR</b>	0x00 7078	1	XINT1 计数器寄存器
<b>XINT2CTR</b>	0x00 7079	1	XINT2 计数器寄存器
<b>被保留</b>	0x707A-0x707E	5	
<b>XNMICTR</b>	0x00 707A	1	XNMI 计数器寄存器

(1) 每个外部中断可被启用/禁用或者使用正边沿以及负边沿来限定中断。

### 3.7 系统控制

这一部分对振荡器、PLL、时钟机制、看门狗功能以及低功耗模式进行了说明。图 3-6 显示了讨论的多种时钟域。



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

图 3-6 时钟和复位域

注

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

PLL, 时钟, 看门狗和低功耗模式由表 3-15 中列出的寄存器控制。

表 3-15 PLL、时钟、看门狗和低功耗模式寄存器

名称	地址	大小 (X 16)	说明 <sup>(1)</sup>
PLLSTS	0x00 7011	1	PLL 状态寄存器
被保留	0x00 7012-0x00 7018	7	被保留
被保留	0x00 7019	1	被保留
HISPCP	0x00 701A	1	高速外设时钟预分频寄存器
LOSPCP	0x00 701B	1	低速外设时钟预分频寄存器
PCLKCR0	0x00 701C	1	外设时钟控制寄存器0
PCLKCR1	0x00 701D	1	外设时钟控制寄存器1
LPMCR0	0x00 701E	1	低功耗模式控制寄存器0
被保留	0x00 701F	1	被保留
PCLKCR3	0x00 7020	1	外设时钟控制寄存器3
PLLCR	0x00 7021	1	PLL 控制寄存器
SCSR	0x00 7022	1	系统控制与状态寄存器
WDCNTR	0x00 7023	1	看门狗计数器寄存器
被保留	0x00 7024	1	被保留
WDKEY	0x00 7025	1	看门狗复位密钥寄存器
被保留	0x00 7026-0x00 7028	3	被保留
WDCR	0x00 7029	1	看门狗控制寄存器
被保留	0x00 702A-0x00 702D	4	被保留
MAPCNF	0x00 702E	1	EPWM/HRPWM 重新映射寄存器

### 3.7.1 OSC 和 PLL 模块

图 3-7 显示了 OSC 和 PLL 块。

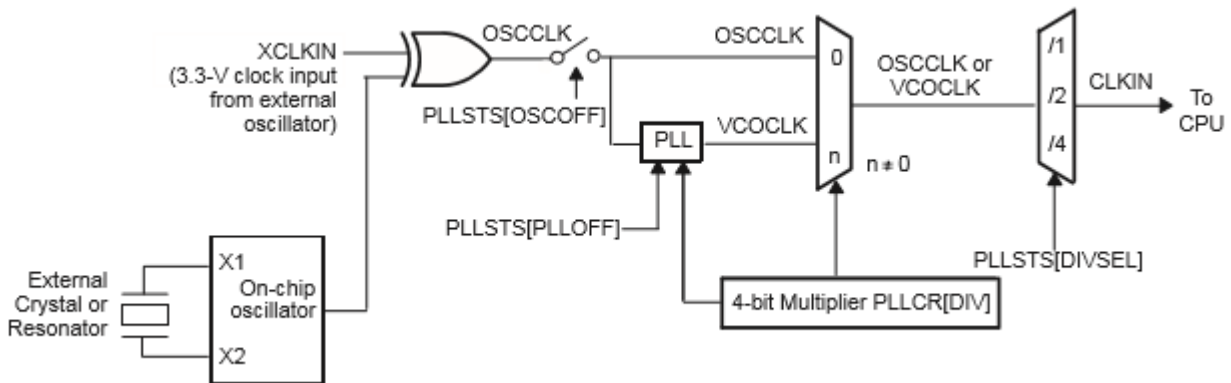


图 3-7 OSC 和 PLL 模块方框图

片载振荡器电路启用一个使用 X1 和 X2 引脚连接至 AVP32F335 器件的晶振/谐振器。如果片载振荡器未被使用，那么一个外部振荡器可被用在下列配置中的任何一个：

1. 一个 3.3V 外部振荡器可被直接接至 XCLKIN 引脚。X2 引脚应被悬空，而 X1 引脚应在低电平时。这个情况下的逻辑高电平不用超过 VDDIO。
2. 一个 1.9V (100MHz 器件时为 1.8V，或在 VDD 幅值范围内 90%以上的振幅) 外部振荡器可以直接连接到 X1 引脚。X2 引脚应被悬空，而 XCLKIN 引脚应在低电平时。这个情况下的逻辑高电平不应超过 VDD。

图 3-8 至图 3-10 显示了三个可能的输入时钟配置。



图 3-8 3.3V 外部振荡器的使用

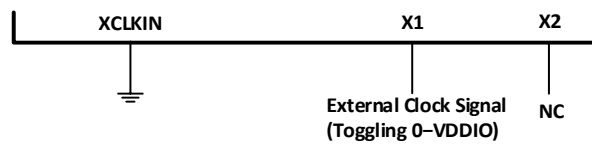


图 3-9 1.9V 外部振荡器的使用

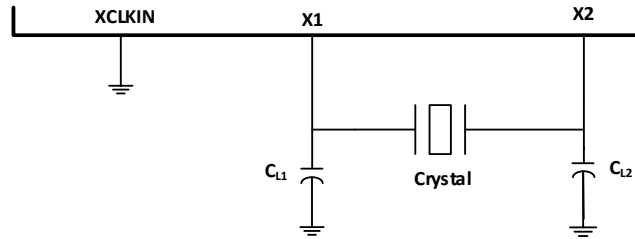


图 3-10 外部振荡器的使用

### 3.7.1.1 外部基准振荡器时钟选项

30MHz 外部石英晶振的典型技术规范如下：

- 基本模式、并联谐振
- CL ( 负载电容 ) = 12pF
- CL1=CL2=24pF
- C 并联=6pF
- ESR 范围 = 25 至 40Ω

### 3.7.1.2 基于 PLL 的时钟模块

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

表 3-16 PLL 设置<sup>(1)</sup>

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



1011-1111	被保留	被保留	被保留
-----------	-----	-----	-----

- (1) 缺省情况下，PLLSTS[DIVSEL]被配置为/4。（引导 ROM 将这个配置改为/2。）在写入 PLLCR 前，PLLSTS[DIVSEL]必须为 0，而只有当 PLLSTS[PLLOCKS]=1 时才应被改变。
- (2) PLL 控制寄存器(PLLCR)和 PLL 状态寄存器(PLLSTS)只能通过 XRS 信号或者一个看门狗复位被复位至它们的缺省值。调试器或丢失时钟检测逻辑发出的复位信号无效。
- (3) 此寄存器受 EALLOW 保护。
- (4) 在 PLL 输出上的分频器是必须的以确保馈入内核的时钟的正确占空比。出于这个原因，当 PLL 处于激活状态时，DIVSEL 值不许为 3。

**表 3-17 CLKIN 分频选项**

PLLSTS [DIVSEL]	CLKIN 分频
0	/4
1	/4
2	/2
3	/1 <sup>(1)</sup>

基于 PLL 的时钟模块提供两种运行模式：

- 晶振操作-这个模式允许使用一个外部晶振/谐振器来提供到器件的时基。
- 外部时钟源操作-这个模式允许内部振荡器被旁通。此器件时钟由一个 X1 或者 XCLKIN 引脚上的外部时钟源输入生成。

**表 3-18 可能的 PLL 配置模式**

PLL 模式	注释	PLLSTS[DIVSEL]	CLKIN 和 SYSCLKOUT
PLL 被禁用	由 PLLSTS 寄存器中的 PLLOFF 位控制。在这个模式中，PLL 块被禁用。这对降低系统噪声和功耗非常有用。在进入此模式之前，必须先将 PLLCR 寄存器设置为 0x0000 (PLL 旁路)。CPU 时钟 (CLKIN) 直接源自 X1/X2、X1 或者 XCLKIN 中任何一个上的时钟输入。	0, 1	OSCCLK/4
		2	OSCCLK/2
		3	OSCCLK/1
PLL 被旁路	PLL 旁路是加电或外部复位 XRS 时的默认 PLL 配置。当 PLLCR 寄存器设置为 0x0000 时或在 PLLCR 寄存器已经被修改之后，PLL 锁定至新频率时选择此模式。在此模式中 PLL 本身被旁路，但未关闭。	0, 1	OSCCLK/4
		2	OSCCLK/2
		3	OSCCLK/1
PLL 被启用	通过将一个非零值“n”写入 PLLCR 寄存器实现。在写入 PLLCR 时，此器件将在 PLL 锁之前切换至 PLL 旁路模式。	0, 1	OSCCLK * n/4
		2	OSCCLK * n/2

### 3.7.1.3 输入时钟丢失

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

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

---

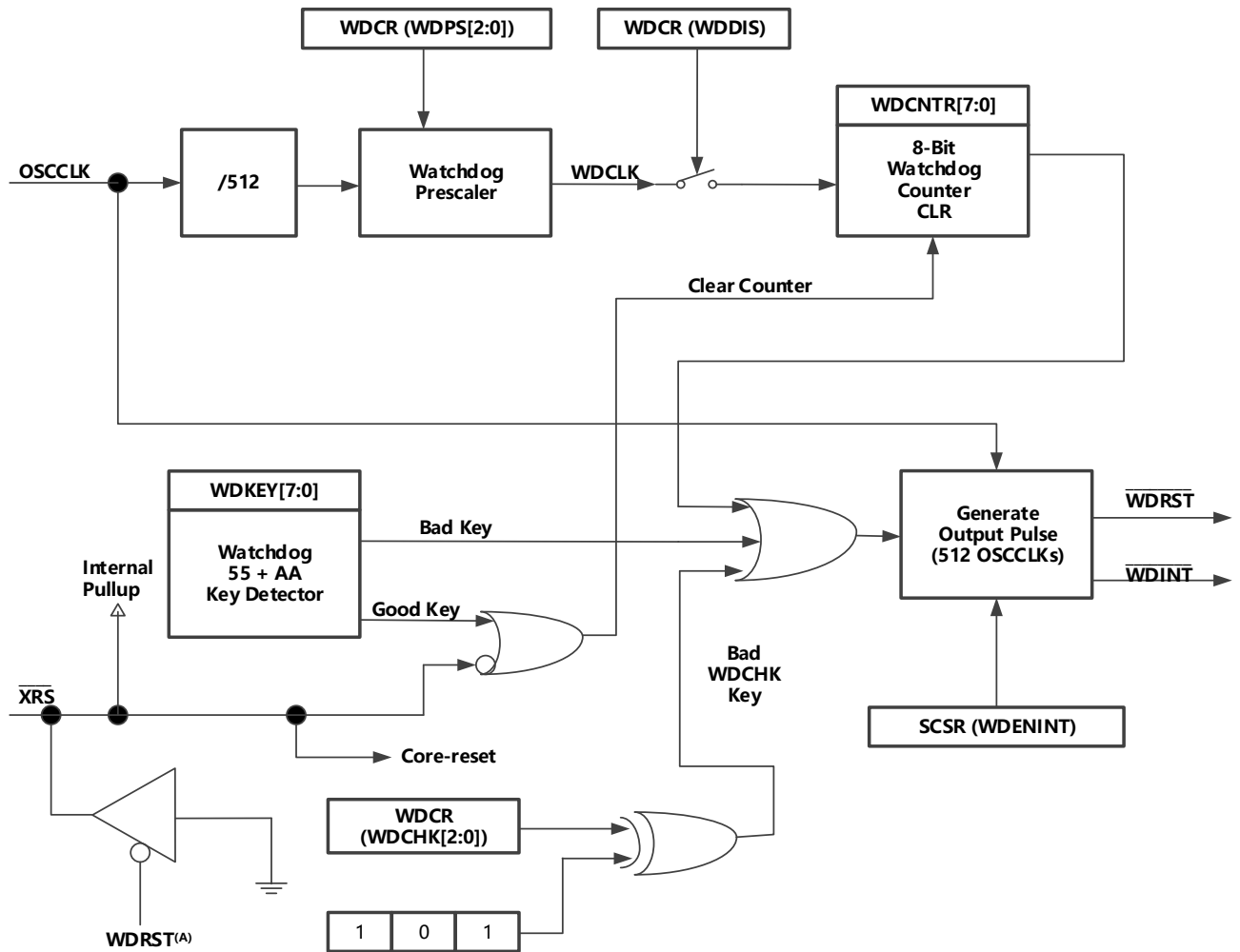
#### 注

在正确 CPU 运行频率绝对关键的应用中应该执行一个机制，通过这个机制，只要输入时钟出现故障，DSC 就被保持在复位状态。例如，只要电容器充满电，一个 R-C 电路可被用于触发 DSC 的 XRS 引脚。一个 I/O 引脚可被用于定期为电容器放电以防止其被完全充满。这样一个电路也将有助于检测 VDD3VFL 电源轨的故障。

---

### 3.7.2 看门狗

AVP32F335 器件上的 CPU 看门狗与 ADP32F1x/AVP32F335 器件上所使用的模块相类似。只要 8 位看门狗上的数计数器在 512 个振荡器时钟宽度(OSCCLK)后达到了它的最大值，这个模块就生成一个输出脉冲。为了防止这一情况，用户必须禁用此计数器或者通过软件定期地往看门狗密钥寄存器写入一个 0x55+0xAA 序列复位此看门狗计数器。图 3-11 显示了看门狗模块内的各种功能块。



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

图 3-11 CPU - 看门狗模块

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

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

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

在 HALT 模式中，CPU 看门狗可被用于通过一个器件复位来唤醒器件。

## 3.8 低功耗模式块

AVP32F335 器件的低功率模式和 ADP16x 器件的模式相似。表 3-19 总结了各种模式。

**表 3-19 低功耗模式**

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

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

(2) 在 AVP32F335 上，内核时钟 (CLKIN) 被关闭，JTAG 仍然可以工作。

三种不同的低功耗模式运行状态如下：

### **IDLE 模式**

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

### **STANDBY 模式**

任何一个 GPIO 端口 A 信号 (GPIO[31:0]) 能够将器件从 STANDBY 模式中唤醒。用户必须通过 GPIOLPMSEL 寄存器选择哪一个信号将器件唤醒。所选的信号在经过 LPMCR0 寄存器中设定 OSCCLK 的数量时钟后，在合格时间内被取值。

### **HALT 模式**

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

---

**注**

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

---

## 4 外设

AVP32F335 器件的集成外设有以下部分进行了说明：

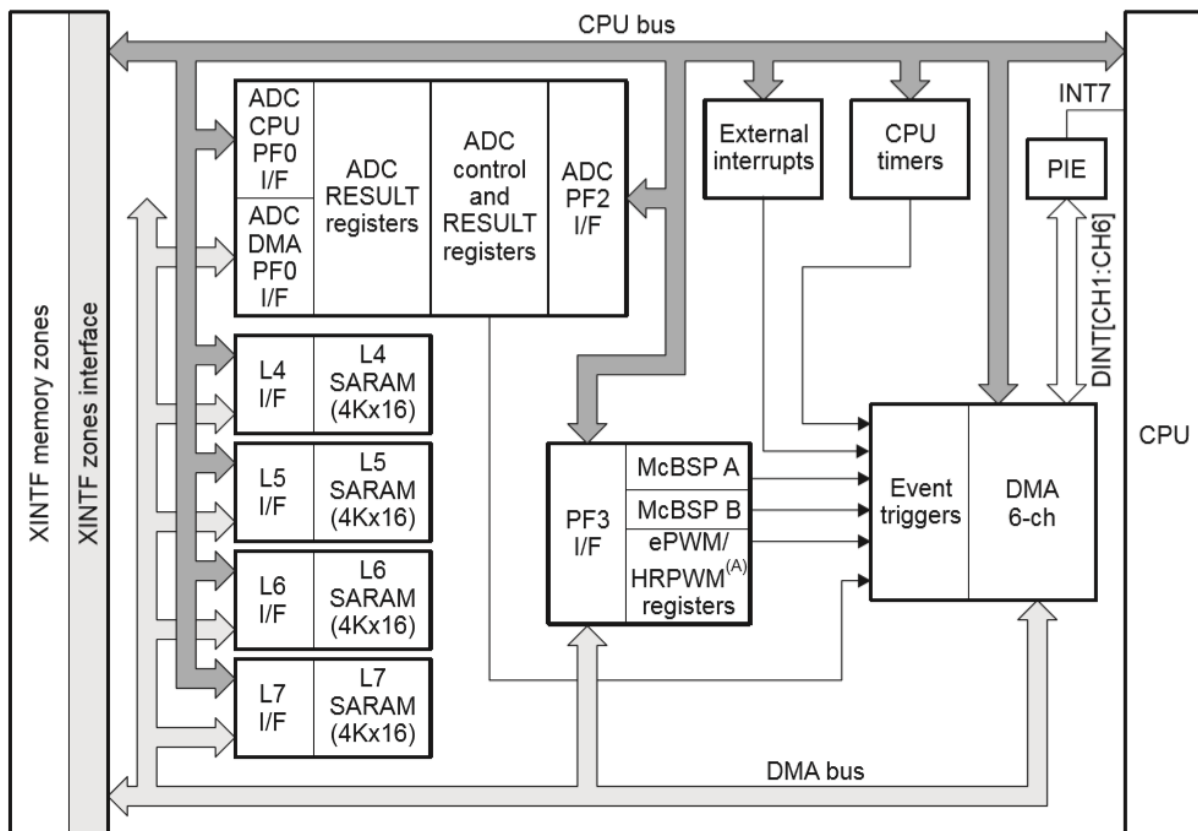
- 6 通道直接内存存取(DMA)
- 三个 32 位 CPU 定时器
- 高达 6 个增强型 PWM 模块(ePWM1 , ePWM2 , ePWM3 , ePWM4 , ePWM5 , ePWM6)
- 高达 6 个增强型捕获模块(eCAP1 , eCAP2 , eCAP3 , eCAP4 , eCAP5 , eCAP6)
- 高达 2 个增强型 QEP 模块(eQEP1 , eQEP2)
- 增强型模数转换器(ADC)模块
- 多达 2 个增强型控制器局域网(eCAN)模块(eCAN-A , eCAN-B)
- 多达 3 个串行通信接口模块(SCI-A , SCI-B , SCI-C)
- 1 个串行外设接口(SPI)模块(SPI-A)
- 内部集成电路模块(I2C)
- 高达两个多通道缓冲串口(McBSP-A , McBSP-B)模块
- 数字 I/O 和共用引脚功能
- 外部接口(XINTF)

### 4.1 DMA 概述

特性:

- 6 个具有独立 PIE 中断的通道
- 触发源：
  - ePWMSOCA/SOCB-ADC 序列发生器 1 和序列发生器 2
  - McBSP-A 和 McBSP-B 传输和接收逻辑
  - XINT1-7 和 XINT13
  - CPU 定时器
  - 软件
- 数据源/目的地：
  - L4-L716K×16SARAM
  - 所有 XINTF 区域
  - ADC 内存总线映射结果寄存器
  - McBSP-A 和 McBSP-B 发送和接收缓冲区
  - ePWM 寄存器

- 字大小：16 位或 32 位 ( McBSPs 限制到 16 位 )
- 吞吐量：4 周期/字 ( McBSP 读取时为 5 周期/字 )



A. ePWM 和 HRPWM 寄存器必须重新映射到 PF3 (通过 MAPCNF 寄存器的位 0) 之后才可以由 DMA 访问。

图 4-1 DMA 功能方框图

## 4.2 32 位 CPU 定时器 0 , 定时器 1 , 定时器 2

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

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

注

请注意：如果应用没有在使用 DSP/BIOS，那么 CPU 定时器 2 可被用在应用中。

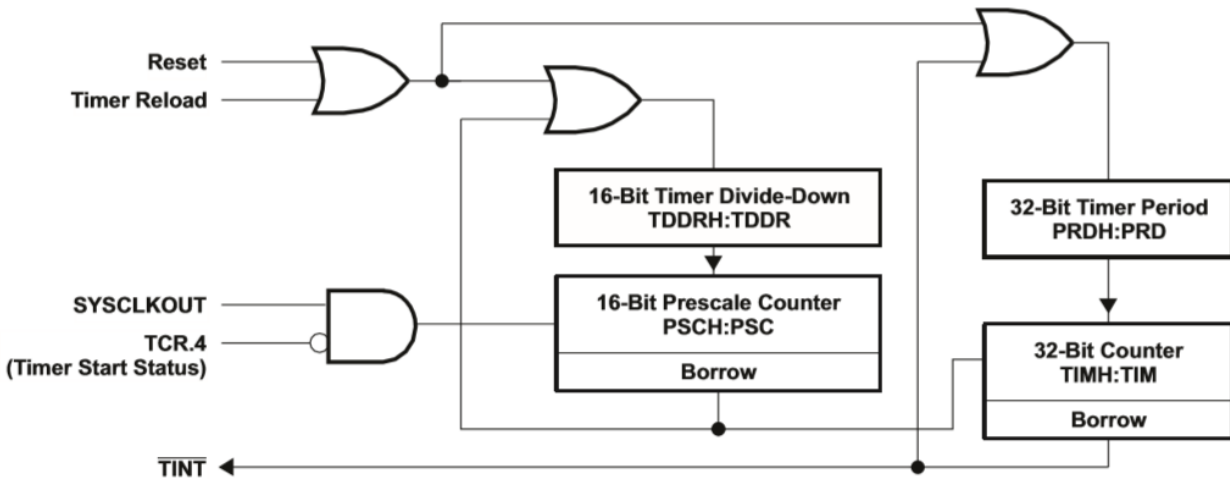
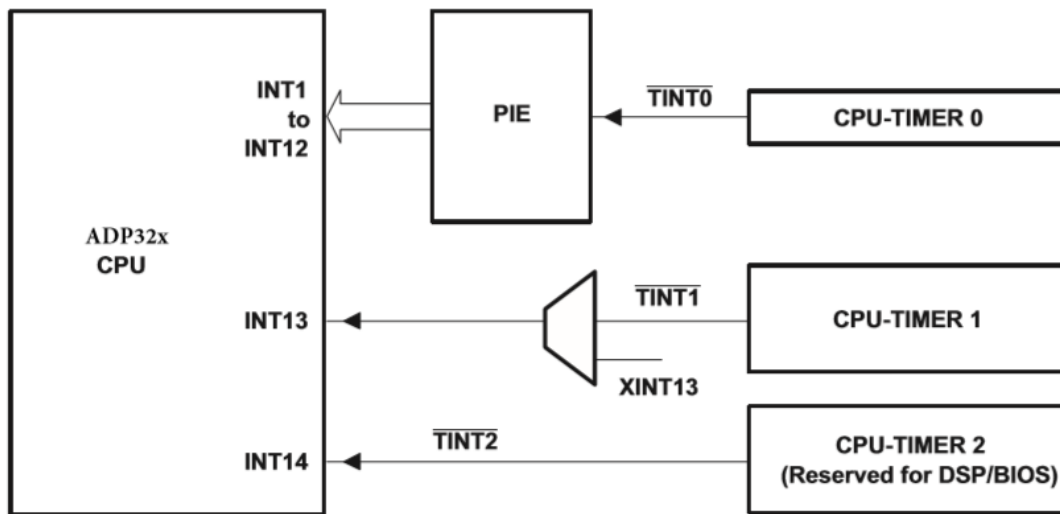


图 4-2 CPU 定时器

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



- A. 定时器寄存器连接到 ADP32x 处理器的存储器总线。
- B. 定时器的时序与处理器时钟的 SYSCLKOUT 同步。

图 4-3 CPU 定时器中断信号和输出信号

定时器的通常操作如下：32 位计数器寄存器“TIMH:TIM”被装入周期寄存器 PRDH:PRD 中的值。计数器寄存器按 ADP32x 的 SYSCLKOUT 速率递减。当计数器到达 0 时，一个定时器中断输出信号生成一个中断脉冲。表 4-1 中列出的寄存器用于配置定时器。

**表 4-1 CPU 定时器 0,1,2 配置和控制寄存器**

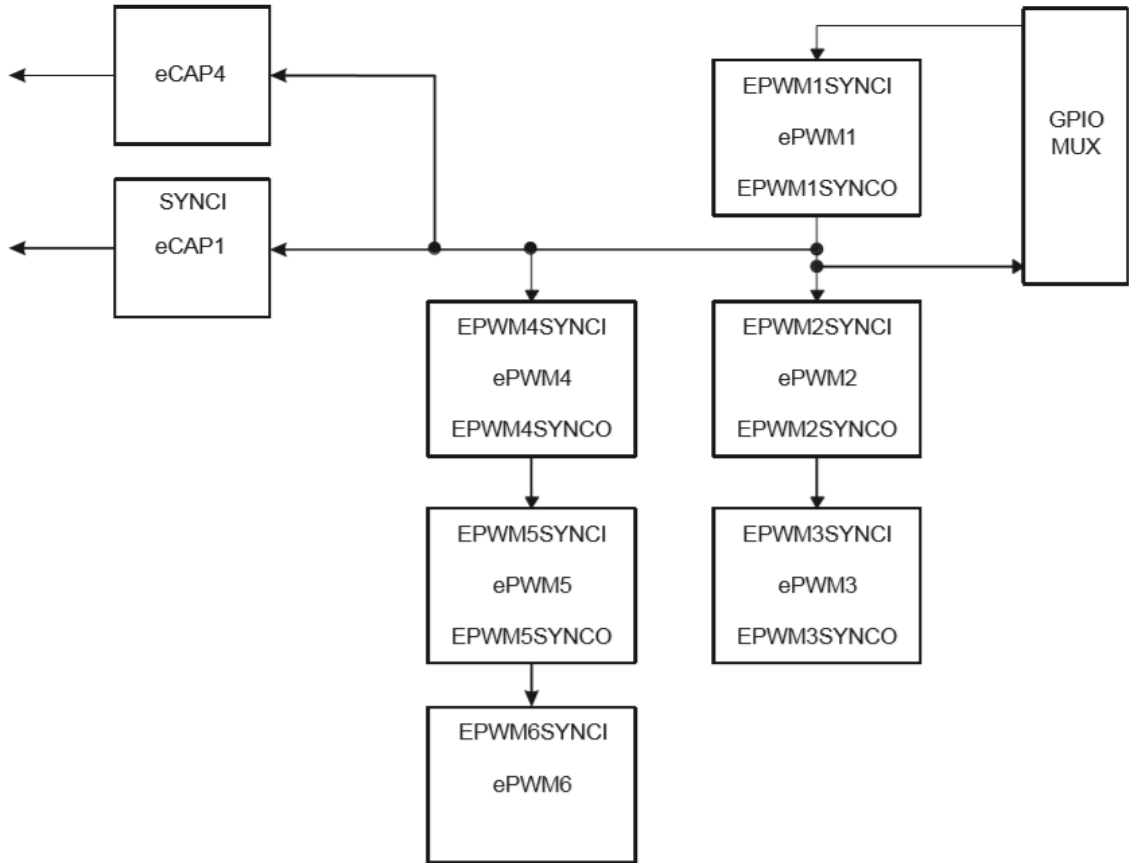
寄存器名称	地址	大小(x 16)	说明
TIMER0TIM	0x0C00	1	CPU 定时器 0, 计数器寄存器
TIMER0TIMH	0x0C01	1	CPU 定时器 0, 计数器寄存器高电平
TIMER0PRD	0x0C02	1	CPU 定时器 0, 周期寄存器
TIMER0PRDH	0x0C03	1	CPU 定时器 0, 周期寄存器高电平
TIMER0TCR	0x0C04	1	CPU 定时器 0, 控制寄存器
被保留	0x0C05	1	
TIMER0TPR	0x0C06	1	CPU 定时器 0, 预分频寄存器
TIMER0TPRH	0x0C07	1	CPU 定时器 0, 预分频寄存器高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

### 4.3 增强型 PWM 模块

AVP32F335 器件包含高达 6 个增强型 PWM 模块 (ePWM1, ePWM2, ePWM3, ePWM4, ePWM5, ePWM6)。图 4-4 显示时基计数器同步方案 3。图 4-5 显示了与 ePWM 互连的信号。

表 4-2 显示 每个模块的完整 ePWM 寄存器设置和表 4-3 显示重新映射的寄存器配置。





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

图 4-4 时基计数器同步方案 3

**表 4-2 ePWM 控制和状态寄存器(PF1 中的默认置)**

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

(1) 寄存器受 EALLOW 保护。

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

名称	ePWM1	ePWM2	ePWM3	ePWM4	ePWM5	ePWM6	大小(x16)/ #SHADOW	说明
TBCTL	0x5800	0x5840	0x5880	0x58C0	0x5900	0x5940	1/0	时基控制寄存器
TBSTS	0x5801	0x5841	0x5881	0x58C1	0x5901	0x5941	1/0	时基状态寄存器
TBPHSHR	0x5802	0x5842	0x5882	0x58C2	0x5902	0x5942	1/0	时基相位 HRPWM 寄存器
TBPHS	0x5803	0x5843	0x5883	0x58C3	0x5903	0x5943	1/0	时基相位寄存器
TBCTR	0x5804	0x5844	0x5884	0x58C4	0x5904	0x5944	1/0	时基计数器寄存器
TBPRD	0x5805	0x5845	0x5885	0x58C5	0x5905	0x5945	1/1	时基周期寄存器集
CMPCTL	0x5807	0x5847	0x5887	0x58C7	0x5907	0x5947	1/0	计数器比较控制寄存器
CMPAHR	0x5808	0x5848	0x5888	0x58C8	0x5908	0x5948	1/1	时基比较 A HRPWM 寄存器
CMPA	0x5809	0x5849	0x5889	0x58C9	0x5909	0x5949	1/1	计数器比较 A 寄存器设置
CMPB	0x580A	0x584A	0x588A	0x58CA	0x590A	0x594A	1/1	计数器比较 B 寄存器设置
AQCTLA	0x580B	0x584B	0x588B	0x58CB	0x590B	0x594B	1/0	用于输出 A 的操作限定器控制寄存器
AQCTLB	0x580C	0x584C	0x588C	0x58CC	0x590C	0x594C	1/0	用于输出 B 的操作限定器控制寄存器
AQSFR	0x580D	0x584D	0x588D	0x58CD	0x590D	0x594D	1/0	操作限定器软件强制寄存器
AQCSFR	0x580E	0x584E	0x588E	0x58CE	0x590E	0x594E	1/1	操作限定器连续 S/W 强制寄存器设置
DBCTL	0x580F	0x584F	0x588F	0x58CF	0x590F	0x594F	1/1	死区生成器控制寄存器
DBRED	0x5810	0x5850	0x5890	0x58D0	0x5910	0x5950	1/0	死区生成器上升沿延迟计数寄存器
DBFED	0x5811	0x5851	0x5891	0x58D1	0x5911	0x5951	1/0	死区生成器下降沿延迟计数寄存器
TZSEL	0x5812	0x5852	0x5892	0x58D2	0x5912	0x5952	1/0	触发区选择寄存器 <sup>(1)</sup>
TZCTL	0x5814	0x5854	0x5894	0x58D4	0x5914	0x5954	1/0	触发区控制寄存器 <sup>(1)</sup>
TZEINT	0x5815	0x5855	0x5895	0x58D5	0x5915	0x5955	1/0	触发区启用中断寄存器 <sup>(1)</sup>
TZFLG	0x5816	0x5856	0x5896	0x58D6	0x5916	0x5956	1/0	触发区标志寄存器
TZCLR	0x5817	0x5857	0x5897	0x58D7	0x5917	0x5957	1/0	触发区清除寄存器 <sup>(1)</sup>
TZFRC	0x5818	0x5858	0x5898	0x58D8	0x5918	0x5958	1/0	触发区强制寄存器 <sup>(1)</sup>
ETSEL	0x5819	0x5859	0x5899	0x58D9	0x5919	0x5959	1/0	事件触发器选择寄存器
ETPS	0x581A	0x585A	0x589A	0x58DA	0x591A	0x595A	1/0	事件触发器预分频寄存器
ETFLG	0x581B	0x585B	0x589B	0x58DB	0x591B	0x595B	1/0	事件触发器标志寄存器
ETCLR	0x581C	0x585C	0x589C	0x58DC	0x591C	0x595C	1/0	事件触发器清除寄存器
ETFRC	0x581D	0x585D	0x589D	0x58DD	0x591D	0x595D	1/0	事件触发器强制寄存器
PCCTL	0x581E	0x585E	0x589E	0x58DE	0x591E	0x595E	1/0	PWM 斩波器控制寄存器
HRCNFG	0x5820	0x5860	0x58A0	0x58E0	0x5920	0x5960	1/0	HRPWM 配置寄存器 <sup>(1)</sup>

(1) 寄存器受 EALLOW 保护。

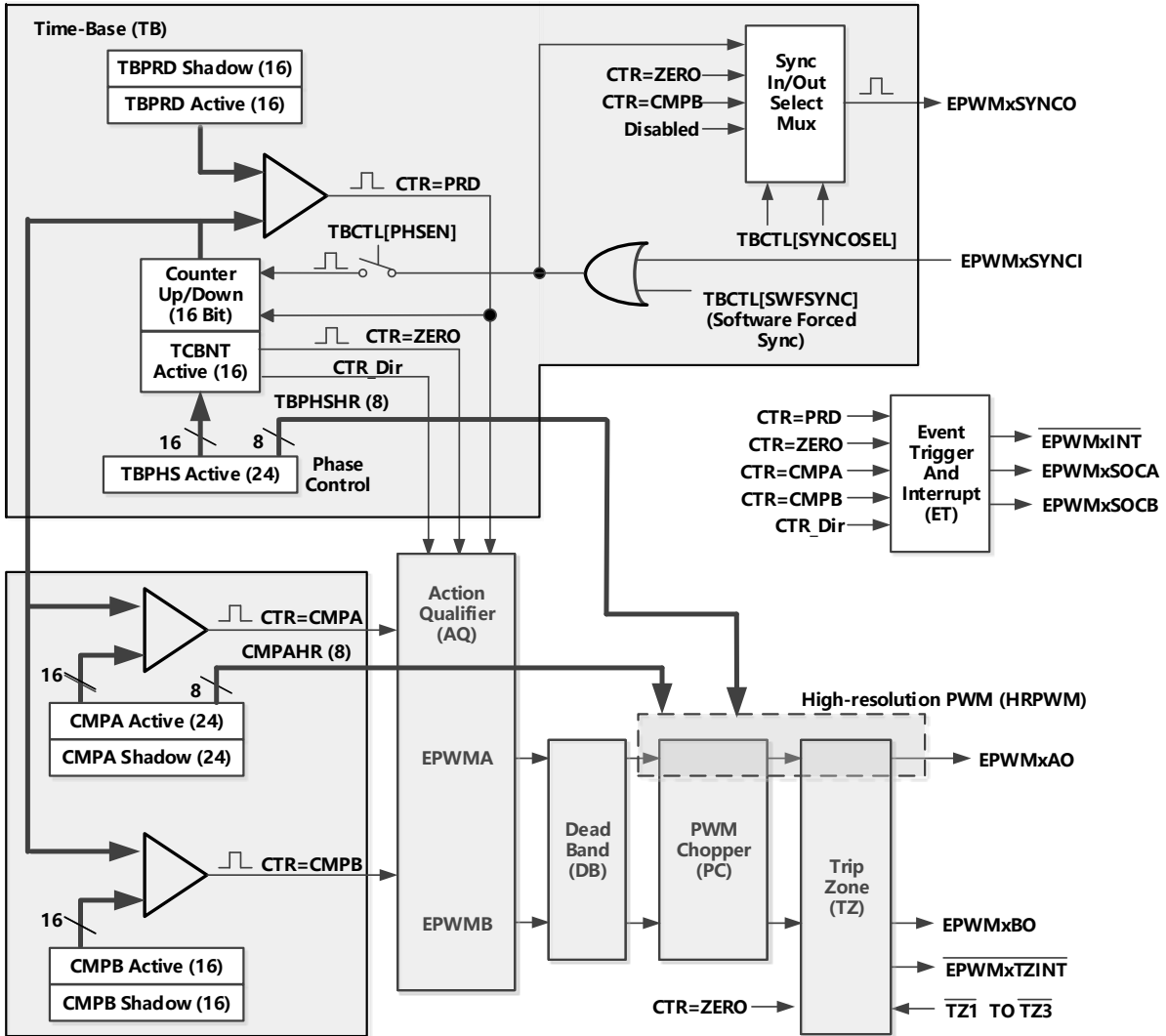


图 4-5 ePWM 子模块显示关键内部信号互连

## 4.4 高分辨率 PWM (HRPWM)

HRPWM 模块提供 PWM 分辨率（时间粒度），此分辨率大大好于使用传统数字 PWM 方法所能导出的分辨率。HRPWM 模块的关键点为：

- 大大扩展了传统导出数字 PWM 的时间分辨率功能
- 通常在有效 PWM 分辨率下降到低于大约 9-10 位时使用。当用一个 100MHz 的 CPU/系统时钟时，PWM 频率大于大约 200kHz 时会发生这种情况。
- 这个功能可被用在占空比和相移控制方法中。
- 通过对 ePWM 模块的比较 A 和相位寄存器的扩展来控制更加精细的时间粒度控制或者边沿定位。
- HRPWM 功能，只在 ePWM 模块的 A 信号路径上提供（也就是说，在 EPWMxA 输出上提供）。EPWMxB 输出具有传统 PWM 功能。

## 4.5 增强型捕捉模块 (eCAP)

AVP32F335 器件包括 6 个增强型捕获 (eCAP) 模块 (eCAP1, eCAP2, eCAP3, eCAP4, eCAP5 和 eCAP6)。图 4-6 显示了一个模块的功能方框图。

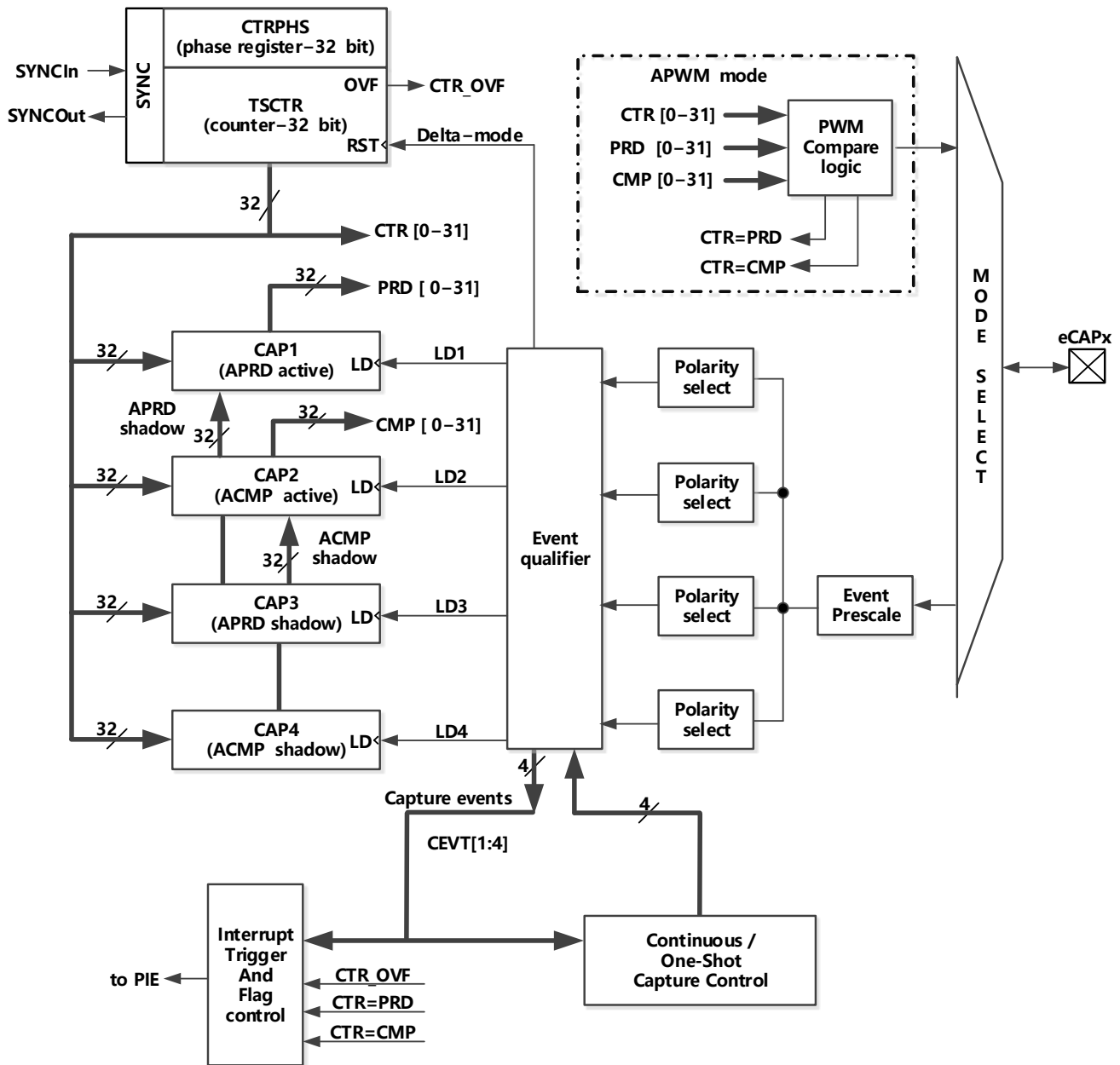


图 4-6 eCAP 功能方框图

eCAP 模块以 SYSCLKOUT 速率计时。

PCLKCR1 寄存器中的时钟使能位(ECAP1ENCLK, ECAP2ENCLK, ECAP3ENCLK, ECAP4ENCLK,

ECAP5ENCLK,ECAP6ENCLK)被单独用来关闭 eCAP 模块 ( 针对低功耗运行 )。复位后 , ECAP1ENCLK,ECAP2ENCLK,ECAP3ENCLK,ECAP4ENCLK,ECAP5ENCLK,ECAP6ENCLK 被设为低电平 , 表明外设时钟关闭。

**表 4-4 eCAP 控制和状态寄存器**

名称	eCAP1	eCAP2	eCAP3	eCAP4	eCAP5	eCAP6	大小 (x 16)	说明
TSCTR	0x6A00	0x6A20	0x6A40	0x6A60	0x6A80	0x6AA0	2	时间戳计数器
CTRPHS	0x6A02	0x6A22	0x6A42	0x6A62	0x6A82	0x6AA2	2	计数器相位偏移值寄存器
CAP1	0x6A04	0x6A24	0x6A44	0x6A64	0x6A84	0x6AA4	2	捕捉 1 寄存器
CAP2	0x6A06	0x6A26	0x6A46	0x6A66	0x6A86	0x6AA6	2	捕捉 2 寄存器
CAP3	0x6A08	0x6A28	0x6A48	0x6A68	0x6A88	0x6AA8	2	捕捉 3 寄存器
CAP4	0x6A0A	0x6A2A	0x6A4A	0x6A6A	0x6A8A	0x6AAA	2	捕捉 4 寄存器
被保留	0x6A0C- 0x6A12	0x6A2C- 0x6A32	0x6A4C- 0x6A52	0x6A6C- 0x6A72	0x6A8C- 0x6A92	0x6AAC- 0x6AB2	8	被保留
ECCTL1	0x6A14	0x6A34	0x6A54	0x6A74	0x6A94	0x6AB4	1	捕捉控制寄存器 1
ECCTL2	0x6A15	0x6A35	0x6A55	0x6A75	0x6A95	0x6AB5	1	捕捉控制寄存器 2
ECEINT	0x6A16	0x6A36	0x6A56	0x6A76	0x6A96	0x6AB6	1	捕捉中断使能寄存器
ECFLG	0x6A17	0x6A37	0x6A57	0x6A77	0x6A97	0x6AB7	1	捕捉中断标志寄存器
ECCLR	0x6A18	0x6A38	0x6A58	0x6A78	0x6A98	0x6AB8	1	捕捉中断清除寄存器
ECFRC	0x6A19	0x6A39	0x6A59	0x6A79	0x6A99	0x6AB9	1	捕捉中断强制寄存器
被保留	0x6A1A- 0x6A1F	0x6A3A- 0x6A3F	0x6A5A- 0x6A5F	0x6A7A- 0x6A7F	0x6A9A- 0x6A9F	0x6ABA- 0x6ABF	6	被保留

## 4.6 增强型正交编码器脉冲 (eQEP)

该器件包括高达 2 个增强型正交编码器(eQEP)模块(eQEP1 , eQEP2)。图 4-7 显示了 eQEP 模块的方框。

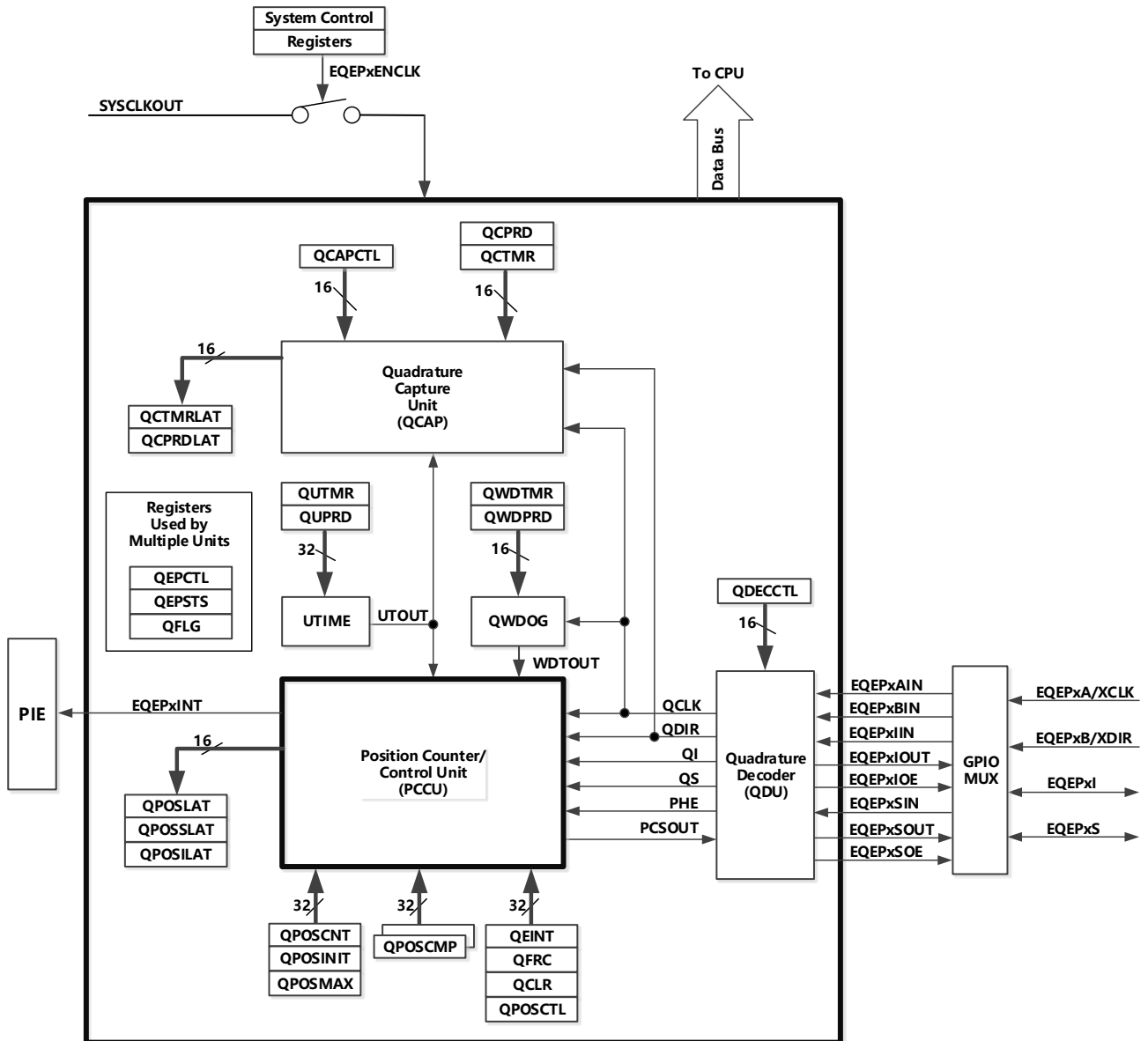


图 4-7 eQEP 功能方框图

表 4-5 显示了 eQEP 寄存器的汇总。



**表 4-5 eQEP 控制和状态寄存器**

名称	eQEP1 地址	eQEP2 地址	eQEP1 大小(x16)/ #SHADOW	说明
QPOSCNT	0x6B00	0x6B40	2/0	eQEP 位置计数器
QPOSINIT	0x6B02	0x6B42	2/0	eQEP 初始化位置计数
QPOSMAX	0x6B04	0x6B44	2/0	eQEP 最大位置计数
QPOSCMP	0x6B06	0x6B46	2/1	eQEP 位置比较
QPOSILAT	0x6B08	0x6B48	2/0	eQEP 索引位置锁存
QPOSSLAT	0x6B0A	0x6B4A	2/0	eQEP 选通脉冲位置锁存
QPOSLAT	0x6B0C	0x6B4C	2/0	eQEP 位置锁存
QUTMR	0x6B0E	0x6B4E	2/0	eQEP 单位定时器
QUPRD	0x6B10	0x6B50	2/0	eQEP 单位周期寄存器
QWDTMR	0x6B12	0x6B52	1/0	eQEP 看门狗定时器
QWDPRD	0x6B13	0x6B53	1/0	eQEP 看门狗周期寄存器
QDECCTL	0x6B14	0x6B54	1/0	eQEP 解码器控制寄存器
QEPCTL	0x6B15	0x6B55	1/0	eQEP 控制寄存器
QCAPCTL	0x6B16	0x6B56	1/0	eQEP 捕捉控制寄存器
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
被保留	0x6B21-0x6B3F	0x6B61-0x6B7F	31/0	

## 4.7 ADC 特性

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

ADC 模块的功能包括：

- 具有内置 S/H 的12位ADC内核
- 模拟输入：0.0 V至3.0 V ( 高于3.0 V的电压会产生满量程转换结果。 )
- 快速转换速率：在7.5 MHz ADC时钟，3.75百万次采样每秒钟 ( MSPS ) 时高达267 ns
- 16 个输入通道，多路复用输入

- 自动排序功能可在单个会话中提供多达16个“自动转换”。可以对每个转换进行编程，以选择16个输入通道中的任何一个
- 可将排序器作为 2 个独立的 8 态排序器，或作为 1 个较大的 16 态排序器使用（即 2 个级联的 8 态排序器）。
- 16个结果寄存器（可单独寻址）存储转换值
  - 输入模拟电压的数字值由以下公式得出：

$$\text{Digital Value} = 0$$

$$\text{when } input \leq 0V$$

$$\text{Digital Value} = 4096 \times \frac{\text{Input Analog Voltage} - \text{ADCL0}}{3}$$

$$\text{when } 0V < input < 3V$$

$$\text{Digital Value} = 4095$$

$$\text{when } input \geq 3V$$

A 所有分数值均截断（小数位都舍去）。

- 作为序列转换开始的多个触发源
  - S/W -软件立即启动
  - ePWM 触发ADC开始转换
  - XINT2 ADC 触发ADC开始转换
- 灵活的中断控制允许对每个序列结束（EOS）或每隔一个EOS的中断请求。
- 排序器可运行于“启动/停止”，以便多个“时序触发器”进行同步转换
- SOCA和SOCB触发器可以在双排序器模式下独立运行。
- 采样保持（S/H）采集时间窗口具有单独的预分频控制。

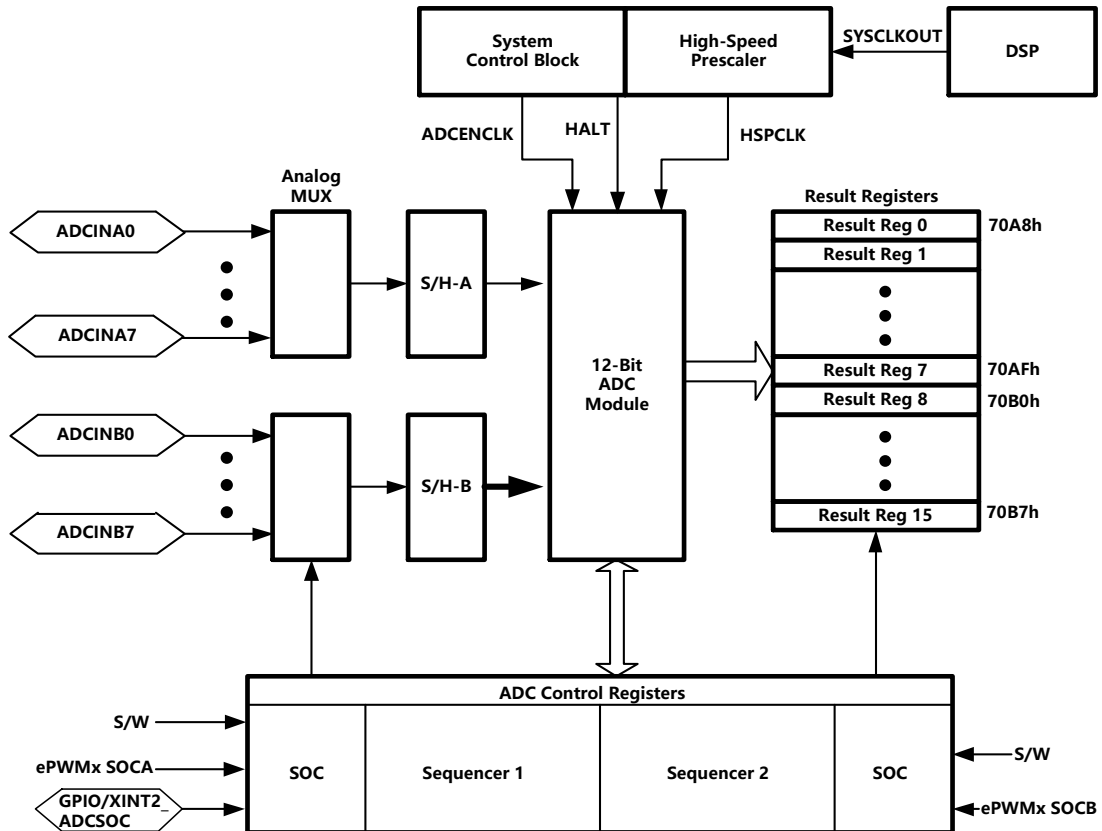


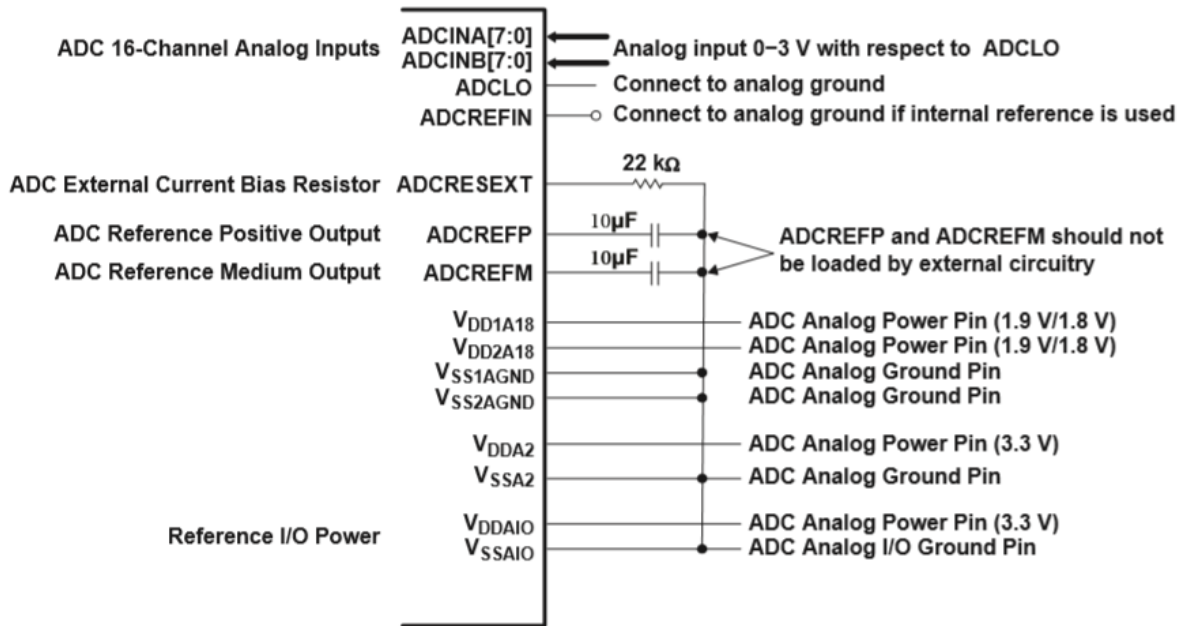
图 4-8 ADC 模块的方框图

要获得指定的 ADC 精度，正确的电路板布局非常关键。为尽可能达到最佳效果，引入 ADCIN 引脚的走线不应太靠近数字信号通道。这是为了最大程度地减少数字线路上因 ADC 输入耦合而产生的开关噪声。而且，适当的隔离技术必须被用来将数字电源从 ADC 模块电源引脚(VDD1A18, VDD2A18, VDDA2, VDDAIO)上隔离。图 4-9 显示针对器件的 ADC 引脚。

注

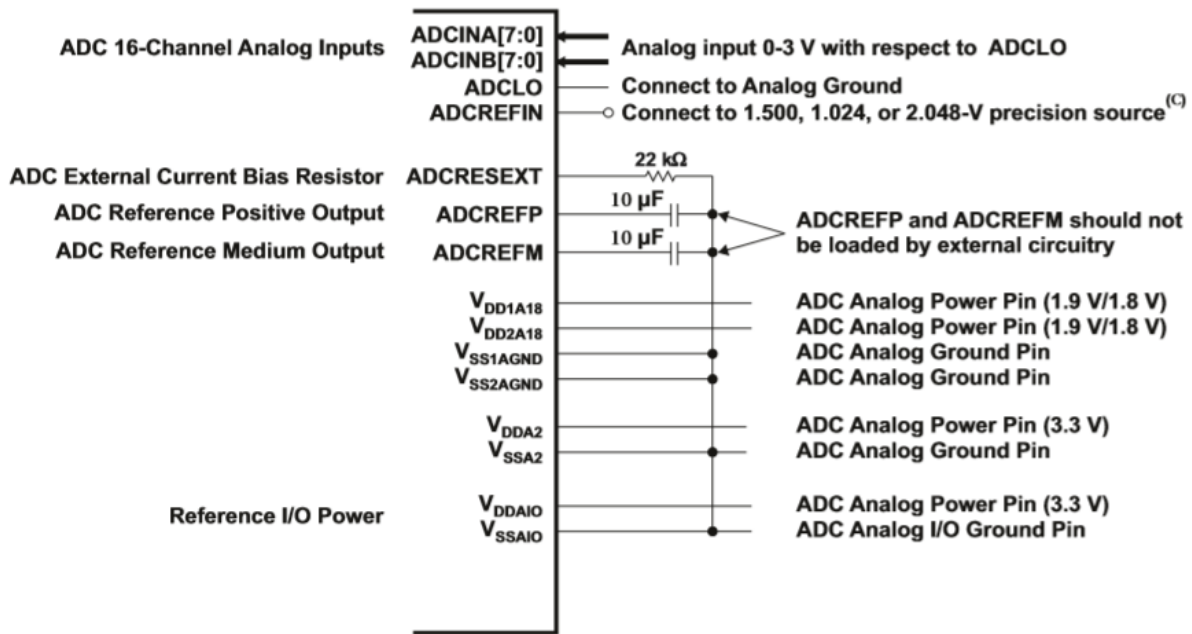
1. 用 SYSCLKOUT 速率对 ADC 寄存器进行访问。ADC 模块的内部时序由高速外设时钟(HSPCLK)控制。
2. 基于 ADCENCLK 和 HALT 信号的 ADC 模块的运行方式如下：
  - ADCENCLK：复位时，这个信号必须为低电平。虽然复位为低电平有效(XRS)，到寄存器的时钟将仍然起作用。有必要确保所有寄存器和模式进入它们的复位状态。然而，模拟模块将处于一个低功耗非激活状态。一旦复位变成高电平，那么到寄存器的时钟将被禁用。当用户将 ADCENCLK 信号设定为高电平时，那么到寄存器的时钟将被启用并且模拟模块将被启用。在 ADC 稳定并且可被使用之前，将有一个特定的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

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



- A. 建议在所有电源引脚上使用外部去耦合电容器。
- B. 必须从不会降低 ADC 性能的运算放大器上驱动模拟输入。

图 4-9 带有内部基准的 ADC 引脚连接



- A. 建议在所有电源引脚上使用外部去耦合电容器。
- B. 模拟输入必须由一个运算放大器驱动，此运算放大器不会降低 ADC 性能。
- C. 根据这个引脚上的电压，通过改变 ADC 基准选择寄存器中的位 15:14 可启用 ADCREFIN 上的外部电压。总体增益精度将由这个电压源的精度确定。

图 4-10 带有外部基准的 ADC 引脚连接

注

任何推荐组件的额定温度必须与最终产品的额定值相匹配。

### 4.7.1 如果 ADC 未被使用，ADC 连接方式

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

- $V_{DD1A18}/V_{DD2A18}$  - 连接至  $V_{DD}$
- $V_{DDA2}$  ,  $V_{DDAIO}$  - 连接至  $V_{DDIO}$
- $V_{SS1AGND}/V_{SS2AGND}$  ,  $V_{SSA2}$  ,  $V_{SSAIO}$  - 连接至  $V_{SS}$
- ADCLO - 连接至  $V_{SS}$
- ADCREFIN - 连接至  $V_{SS}$
- ADCREFP/ADCREFM - 连接一个 100nF 电容器至  $V_{SS}$
- ADCRESEXT - 连接一个 20kΩ 电阻器（非常松散的耐受）至  $V_{SS}$ 。
- ADCINAn , ADCINBn - 连接至  $V_{SS}$

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

---

**注**

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

---

### 4.7.2 ADC 寄存器

[表 4-6](#) 中所列寄存器对 ADC 操作进行配置、控制和监视。

**表 4-6 ADC 寄存器**

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

ADCRESLT5	0x710D	0x0B05	1	ADC 转换结果缓冲寄存器 5
ADCRESLT6	0x710E	0x0B06	1	ADC 转换结果缓冲寄存器 6
ADCRESLT7	0x710F	0x0B07	1	ADC 转换结果缓冲寄存器 7
ADCRESLT8	0x7110	0x0B08	1	ADC 转换结果缓冲寄存器 8
ADCRESLT9	0x7111	0x0B09	1	ADC 转换结果缓冲寄存器 9
ADCRESLT10	0x7112	0x0B0A	1	ADC 转换结果缓冲寄存器 10
ADCRESLT11	0x7113	0x0B0B	1	ADC 转换结果缓冲寄存器 11
ADCRESLT12	0x7114	0x0B0C	1	ADC 转换结果缓冲寄存器 12
ADCRESLT13	0x7115	0x0B0D	1	ADC 转换结果缓冲寄存器 13
ADCRESLT14	0x7116	0x0B0E	1	ADC 转换结果缓冲寄存器 14
ADCRESLT15	0x7117	0x0B0F	1	ADC 转换结果缓冲寄存器 15
ADCTRL3	0x7118		1	ADC 控制寄存器 3
ADCTRL4	0x711A		1	ADC 控制寄存器 4
ADCST	0x7119		1	ADC 状态寄存器
被保留	0x711B		1	
ADCREFSEL	0x711C		1	ADC 基准选择寄存器
ADCOFFTRIM	0x711D		1	ADC 偏移调整寄存器
被保留	0x711E 0x711F		2	

(1)本列中的寄存器为外设帧 2 寄存器。

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

### 4.7.3 ADC 校准

ADC\_cal()例程被工厂编程到 Advchip 预留的 OTP 存储器中。引导 ROM 自动调用 ADC\_cal()例程来使用特定器件的校准数据来初始化 ADCREFSEL 和 ADCOFFTRIM 寄存器。正常运行期间，这个过程中会自动发生，无需用户进行任何操作。

如果在开发过程中，引导 ROM 被 CodeComposerStudio 绕过，那么 ADCREFSEL 和 ADCOFFTRIM 就必须由应用进行初始化。

---

#### 注

这些寄存器初始化失败将导致 ADC 的功能不能达到技术规格。

如果系统复位或 ADC 模块使用 ADC 控制寄存器 1 中的 14 位 ( 复位 ) 复位，则必须重复例程。

---

## 4.8 多通道缓冲串行端口 (McBSP) 模块

McBSP 模块有以下特性：

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

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

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

---

**注**

最大 I/O 引脚切换速度请见[第 6 章](#)。

---



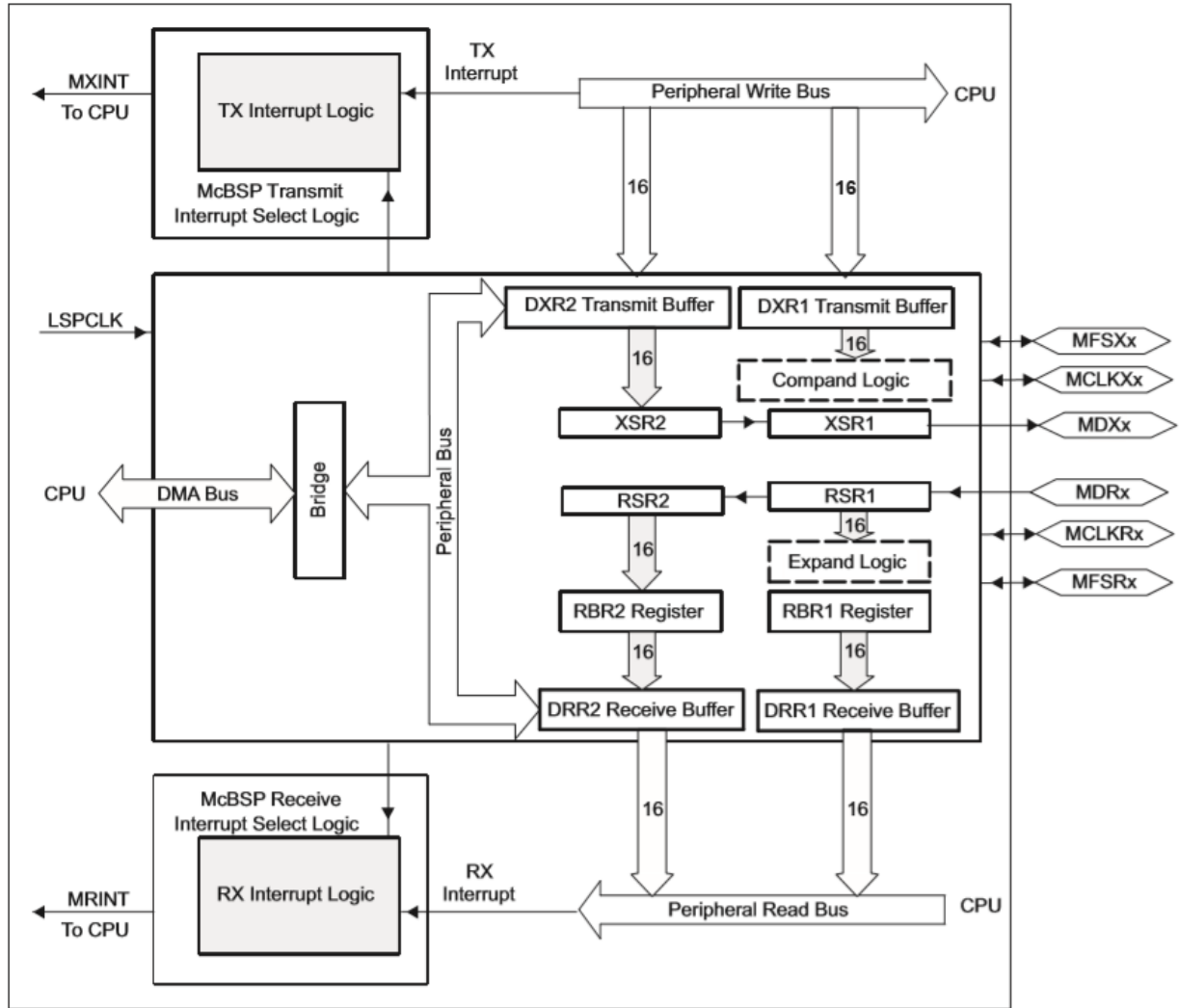


图 4-11 显示了 McBSP 模块的方框图。

表 4-7 提供了 McBSP 寄存器的汇总。

表 4-7 McBSP 寄存器汇总

名称	McBSP-A地址	McBSP-B地址	类型	复位值	说明
数据寄存器，接收、发送					
DRR2	0x5000	0x5040	R	0x0000	McBSP 数据接收寄存器 2
DRR1	0x5001	0x5041	R	0x0000	McBSP 数据接收寄存器 1
DXR2	0x5002	0x5042	W	0x0000	McBSP 数据发送寄存器 2
DXR1	0x5003	0x5043	W	0x0000	McBSP 数据发送寄存器 1
McBSP 控制寄存器					
SPCR2	0x5004	0x5044	读/写	0x0000	McBSP 串行端口控制寄存器 2
SPCR1	0x5005	0x5045	读/写	0x0000	McBSP 串行端口控制寄存器 1

RCR2	0x5006	0x5046	读/写	0x0000	McBSP 接收控制寄存器 2
RCR1	0x5007	0x5047	读/写	0x0000	McBSP 接收控制寄存器 1
XCR2	0x5008	0x5048	读/写	0x0000	McBSP 发送控制寄存器 2
XCR1	0x5009	0x5049	读/写	0x0000	McBSP 发送控制寄存器 1
SRGR2	0x500A	0x504A	读/写	0x0000	McBSP 采样率发生器寄存器 2
SRGR1	0x500B	0x504B	读/写	0x0000	McBSP 采样率发生器寄存器 1
多通道控制寄存器					
MCR2	0x500C	0x504C	读/写	0x0000	McBSP 多通道寄存器 2
MCR1	0x500D	0x504D	读/写	0x0000	McBSP 多通道寄存器 1
RCERA	0x500E	0x504E	读/写	0x0000	McBSP 接收通道使能寄存器分区 A
RCERB	0x500F	0x504F	读/写	0x0000	McBSP 接收通道使能寄存器分区 B
XCERA	0x5010	0x5050	读/写	0x0000	McBSP 发送通道使能寄存器分区 A
XCERB	0x5011	0x5051	读/写	0x0000	McBSP 发送通道使能寄存器分区 B
PCR	0x5012	0x5052	读/写	0x0000	McBSP 引脚控制寄存器
RCERC	0x5013	0x5053	读/写	0x0000	McBSP 接收通道使能寄存器分区 C
RCERD	0x5014	0x5054	读/写	0x0000	McBSP 接收通道使能寄存器分区 D
XCERC	0x5015	0x5055	读/写	0x0000	McBSP 发送通道使能寄存器分区 C
XCERD	0x5016	0x5056	读/写	0x0000	McBSP 发送通道使能寄存器分区 D
RCERE	0x5017	0x5057	读/写	0x0000	McBSP 接收通道使能寄存器分区 E
RCERF	0x5018	0x5058	读/写	0x0000	McBSP 接收通道使能寄存器分区 F
XCERE	0x5019	0x5059	读/写	0x0000	McBSP 发送通道使能寄存器分区 E
XCERF	0x501A	0x505A	读/写	0x0000	McBSP 发送通道使能寄存器分区 F
RCERG	0x501B	0x505B	读/写	0x0000	McBSP 接收通道使能寄存器分区 G
RCERH	0x501C	0x505C	读/写	0x0000	McBSP 接收通道使能寄存器分区 H
XCERG	0x501D	0x505D	读/写	0x0000	McBSP 发送通道使能寄存器分区 G
XCERH	0x501E	0x505E	读/写	0x0000	McBSP 发送通道使能寄存器分区 H
MFFINT	0x5023	0x5063	读/写	0x0000	McBSP 中断使能寄存器

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

CAN 模块有下列特性：

- 与CAN 协议，版本2.0B 完全兼容
- 支持高达1Mbps 的数据速率
- 32 个邮箱，每一个邮箱有下列属性：
  - 可配置为接收或者发送
  - 可使用标准或者扩展标识符进行配置
  - 有一个可编辑接收屏蔽
  - 支持数据和远程帧

- 由0 至8 字节数据组成
- 在接收和发送消息上使用一个32 位时间戳
- 防止接收新消息
- 保持发送消息的动态可编辑优先级
- 采用一个具有两个中断级别的可编辑中断机制
- 在发送或者接收超时采用一个可编辑报警
- 低功耗模式
- 总线活动上的可编辑唤醒
- 对远程请求消息的自动答复
- 丢失仲裁或者错误情况下的帧自动重传
- 由一个特定消息同步的32 位本地网络时间计数器（与邮箱16 协同通信）
- 自测模式
  - 运行在接收自身消息的回路模式。用来提供一个虚拟的确认，从而不需要由其他节点来提供确认位的需要。

---

**注**

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

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

---

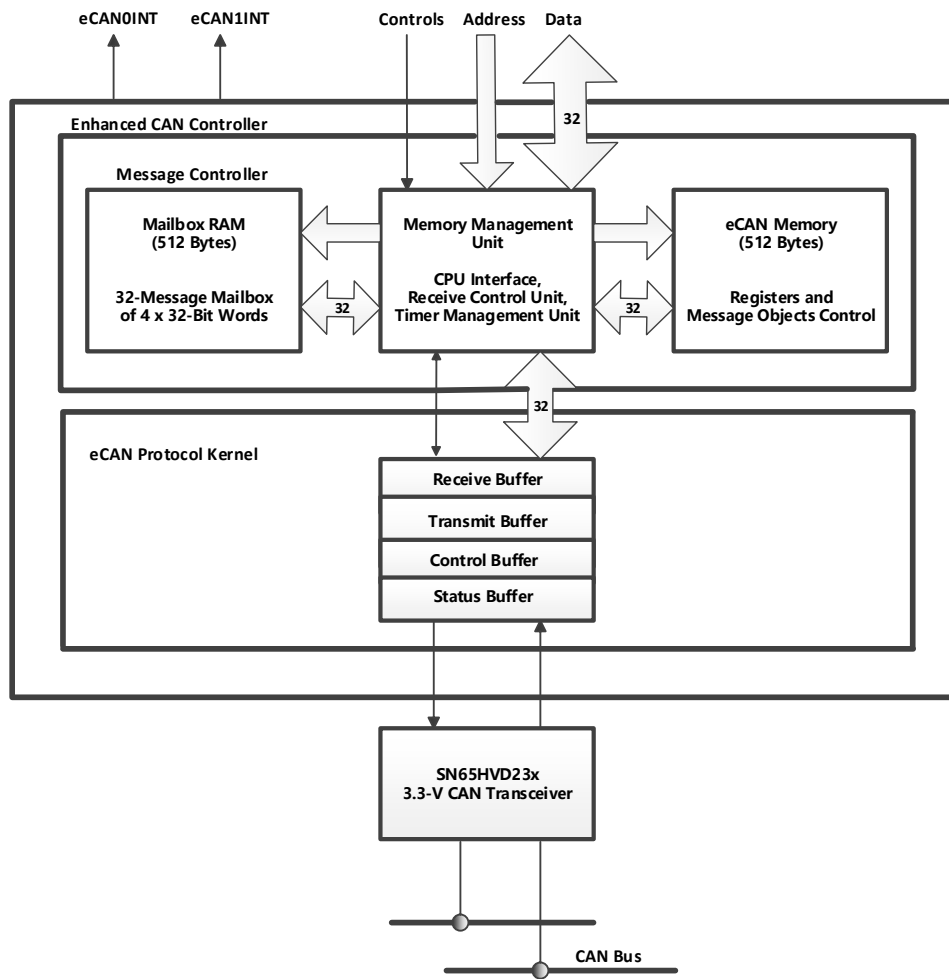


图 4-12 eCAN 方框图和接口电路

表 4-8 3.3V eCAN 收发器用于 AVP32F335

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

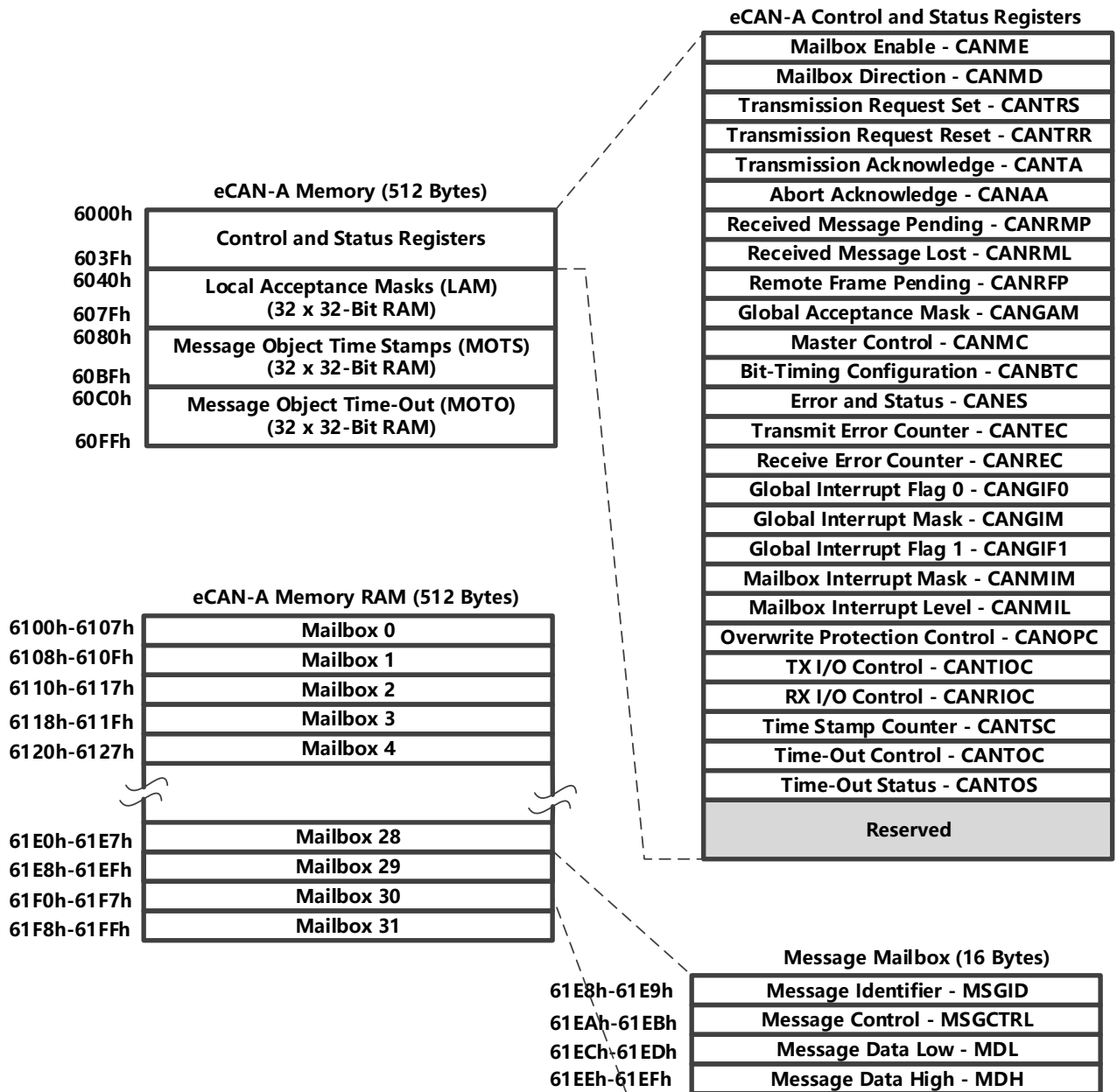


图 4-13 eCAN-A 内存映射

注

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

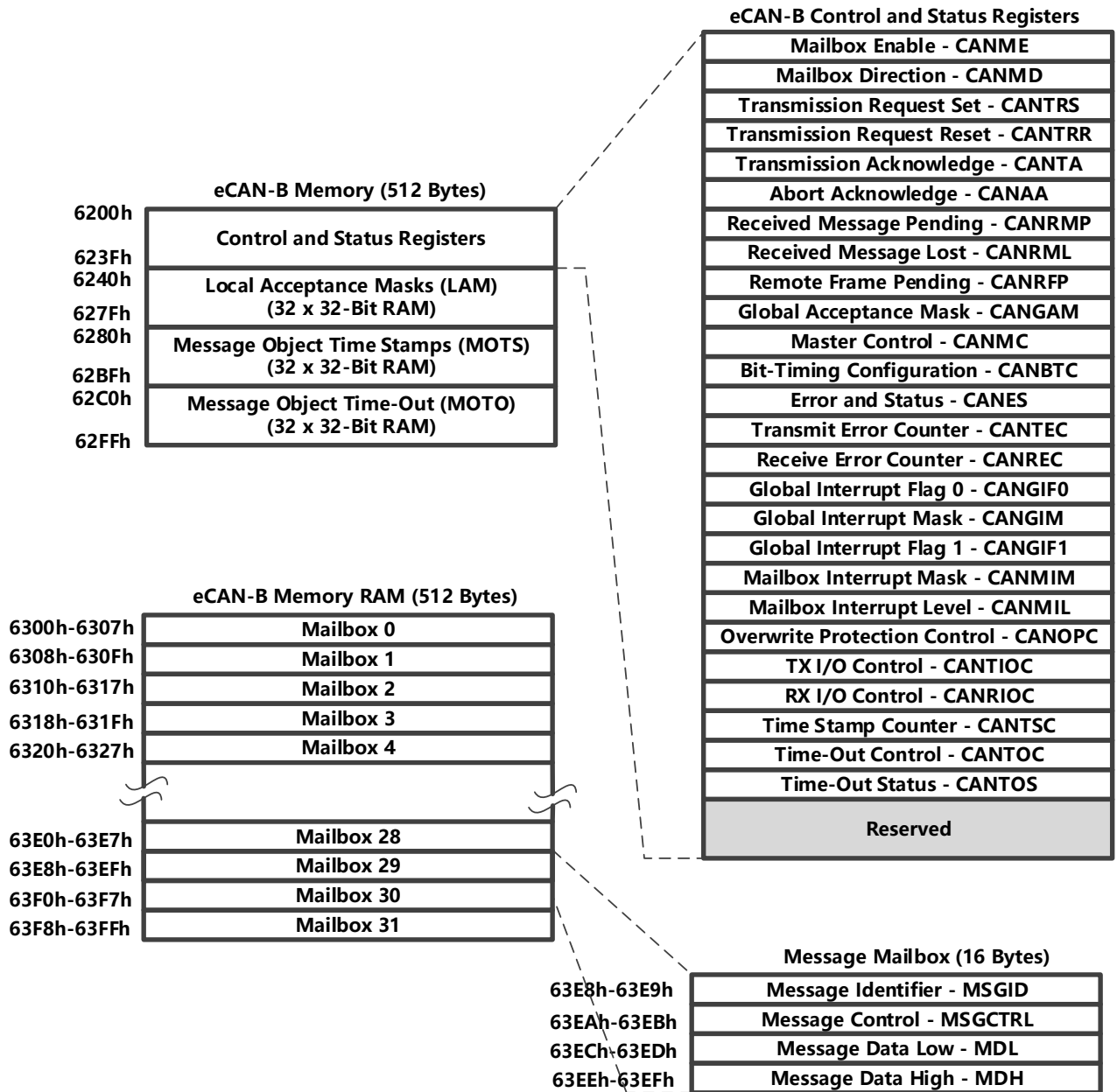


图 4-14 eCAN-B 内存映射

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

**表 4-9 CAN 寄存器映射<sup>(1)</sup>**

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

(1) 这些寄存器被映射至外设帧 1。

## 4.10 异步串行通信接口 (SCI) 模块 ( SCI-A、SCI-B、SCI-C )

AVP32F335器件包括三个异步串行通信接口(SCI) 模块。SCI 模块支持CPU 与其它异步外设之间的使用标准非归零码(NRZ) 格式的数字通信。SCI 接收器和发送器是双缓冲的，并且它们中的每一个有其自身独立的使能和中断位。两个器件都可独立或者同时地运行在全双工模式。为了确保数据完整性，SCI 在中断检测、奇偶性、超载和组帧错误方面对接收到的数据进行检查。通过一个16 位波特率选择寄存器，可将比特率设定为超过65000 种不同的波特率。

每个SCI 模块的特性包括：

• 两个外部引脚：

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

**注释：**两个引脚如果不被用于SCI的话，可被用作GPIO。

• 波特率被设定为64K 个不同速率

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

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

---

**注**

最大 I/O 引脚切换速度请见[第 6 章](#)。

---

• 数据-字格式

- 一个开始位
- 数据-字长度可被设定为1 至8 位
- 可选偶/奇/无奇偶校验位
- 一个或者两个停止位
- 四个错误检测标志：奇偶、超载、组帧和中断检测
- 两个唤醒多处理器模式：空闲线路和地址位
- 半双工或者全双工运行
- 双缓冲接收和发送功能
- 可通过带有状态标志的中断驱动或者轮询算法来完成发射器和接收器操作。
  - 发送器：TXRDY 标志（发射器缓冲寄存器已经准备好接收另外字符）和TX EMPTY（TX 空）标志（发送器移位寄存器已空）
  - 接收器：RXRDY 标志（接收器缓冲寄存器已经准备好接收另外的字符），BRKDT 标志（发生了中断条件）和RX ERROR 错误标志（监控四个中断条件）
- 用于发送器和接收器中断的独立使能位（除了BRKDT）
- NRZ（非归零）码格式

---

**注**

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

---

增强型特性：

- 自动波特率检测硬件逻辑电路



- 16 级发送/接收FIFO

SCI 端口运行由表 4-10, 表 4-11, 表 4-12 中列出的寄存器配置和控制

**表 4-10 SCI-A 寄存器<sup>(1)</sup>**

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

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

(2) 这些寄存器是用于 FIFO 模式的全新寄存器。

**表 4-11 SCI-B 寄存器<sup>(1) (2)</sup>**

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

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

(2) 这些寄存器是用于 FIFO 模式的全新寄存器。

表 4-12 SCI-C 寄存器<sup>(1) (2)</sup>

名称	地址	大小 (x16)	受 EALLOW 保护	说明
SCICCRC	0x00 7770	1	否	SCI-C 通信控制寄存器
SCICTL1C	0x00 7771	1	否	SCI-C 控制寄存器
SCIHBAUDC	0x00 7772	1	否	SCI-C 波特率寄存器, 高位
SCILBAUDC	0x00 7773	1	否	SCI-C 波特率寄存器, 低位
SCICTL2C	0x00 7774	1	否	SCI-C 控制寄存器2
SCIRXSTC	0x00 7775	1	否	SCI-C 接收状态寄存器
SCIRXEMUC	0x00 7776	1	否	SCI-C 接收仿真数据缓冲寄存器
SCIRXBUFC	0x00 7777	1	否	SCI-C 接收数据缓冲寄存器
SCITXBUFC	0x00 7779	1	否	SCI-C 发送数据缓冲寄存器
SCIFFTXC <sup>(2)</sup>	0x00 777A	1	否	SCI-C FIFO 发送寄存器
SCIFFRXC <sup>(2)</sup>	0x00 777B	1	否	SCI-C FIFO 接收寄存器
SCIFFCTC <sup>(2)</sup>	0x00 777C	1	否	SCI-C FIFO 控制寄存器
SCIPRIC	0x00 777F	1	否	SCI-C 优先级控制寄存器

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

(2) 这些寄存器是用于 FIFO 模式的全新寄存器。

[图4-15](#) 显示了SCI 模块方框图。

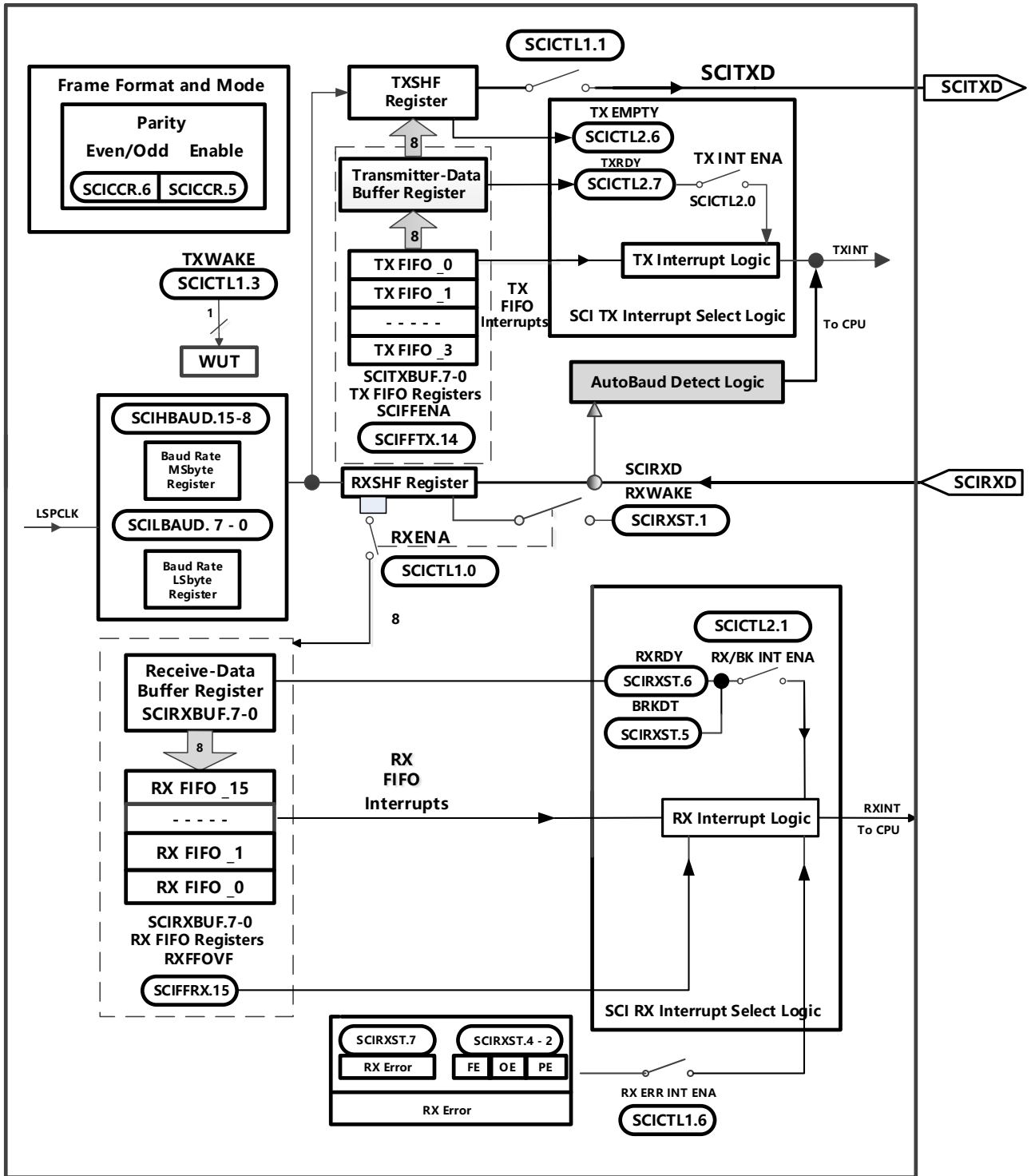


图 4-15 串行通信接口(SCI) 模块方框图

## 4.11 串行通信接口 (SPI) 模块(SPI-A)

AVP32F335 器件包括四引脚的串行通信接口(SPI) 模块。一个SPI模块 ( SPI-A ) 可用。SPI 是一个高

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

SPI 模块的特性包括：

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

**注释：**如果SPI 模块未被使用，所有四个引脚可被用作GPIO。

- 两个运行模式：主控和受控
- 波特率：125 个不同的可编辑速率

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

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

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

---

**注**

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

---

增强型特性：

- 16 级发送/接收FIFO
- 经延迟的发送控制

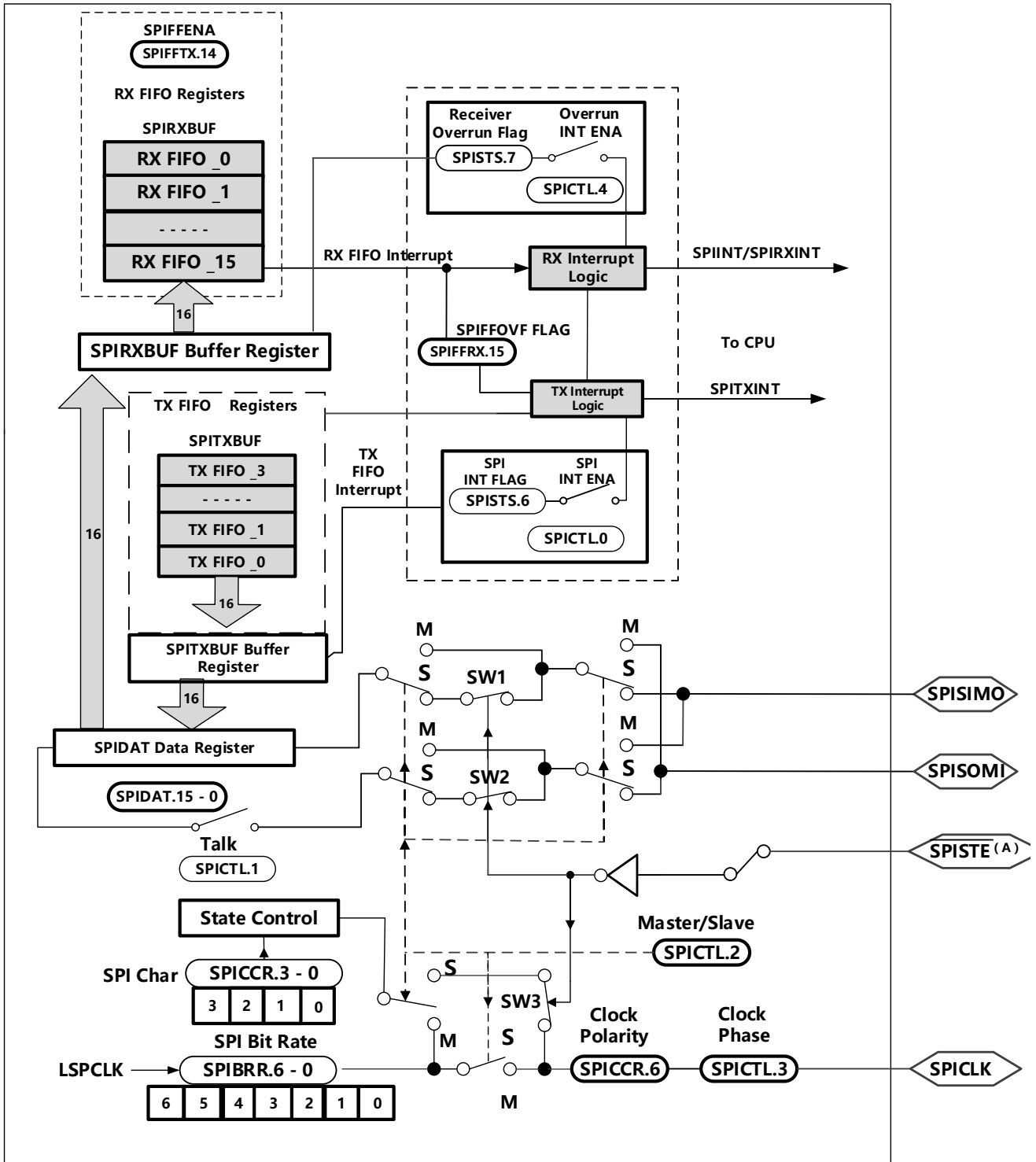
SPI 端口运行由[表4-13](#)中列出的寄存器配置和控制。

**表 4-13 SPI-A 寄存器**

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

(1) 这些寄存器被映射至外设帧 2。这空间只允许 16 位访问。32 位访问会产生未定义的后果。

[图4-16](#)是一个处于受控模式下的SPI 的方框图。

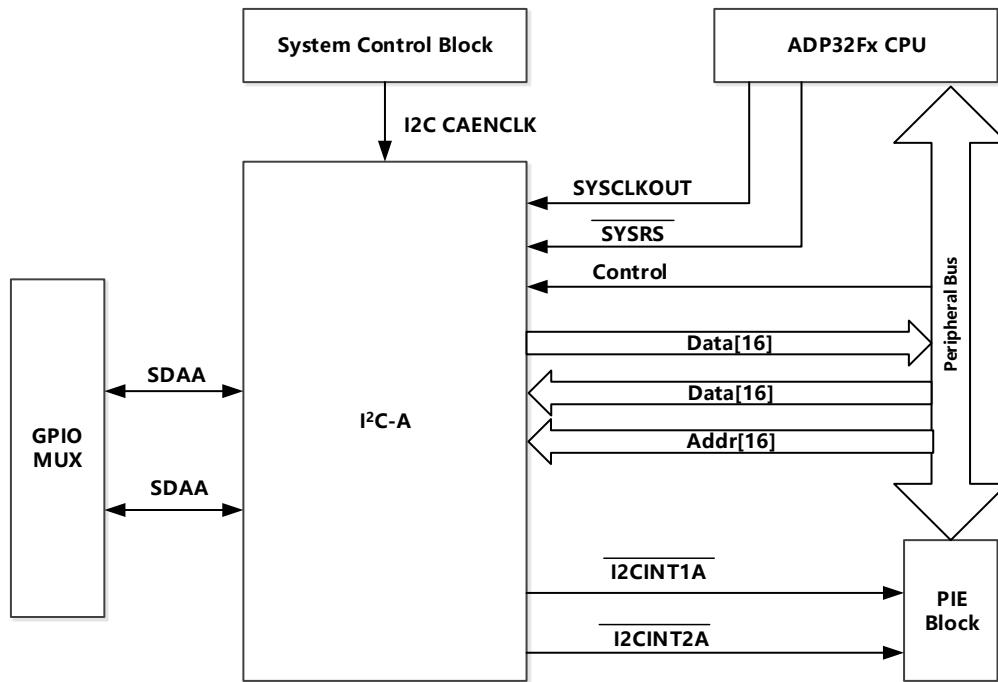


A.  $\overline{SPISTE}$ 被主控器件驱动为用于受控器件的低电平。

图 4-16 SPI 模块方框图 (受控模式)

## 4.12 内部集成电路(I2C)

AVP32F335 包含一个 I2C 串行端口。图 4-17 显示了此器件内的 I2C 外设模块接口。



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

图 4-17 I2C 外设模块接口

I2C 模块具有以下特性：

- 符合飞利浦半导体 I2C 总线规格 (版本 2.1)：
  - 支持 1 位至 8 位格式传输
  - 7 位和 10 位寻址模式
  - 常规调用
  - START 字节模式
  - 支持多个主发送器和从接收器
  - 支持多个从发送器和主接收器
  - 组合主器件发送/接收和接收/发送模式
  - 数据传输速率从 10kbps 到高达 400kbps (I2C 快速模式速率)
- 一个 16 字接收 FIFO 和一个 16 字发送 FIFO
- 可以由 CPU 使用的一个中断。这个中断可由下列条件中的一个生成：
  - 发送数据准备好
  - 接收数据准备好
  - 寄存器访问准备好

- 没有接收到确认
- 仲裁丢失
- 检测到停止条件
- 被寻址为从器件
- 在 FIFO 模式下，CPU 可以使用附加的中断
- 模块启用/禁用能力
- 自由数据格式模式

表 4-14 中的寄存器配置并且扩展 I2C 端口操作。

**表 4-14 I2C-A 寄存器**

名称	地址	说明
I2COAR	0x7900	I2C 自身的地址寄存器
I2CIER	0x7901	I2C 中断使能寄存器
I2CSTR	0x7902	I2C 状态寄存器
I2CCLKL	0x7903	I2C 时钟低电平时间分频器寄存器
I2CCLKH	0x7904	I2C 时钟高电平时间分频器寄存器
I2CCNT	0x7905	I2C 数据计数寄存器
I2CDRR	0x7906	I2C 数据接收寄存器
I2CSAR	0x7907	I2C 从器件地址寄存器
I2CDXR	0x7908	I2C 数据发送寄存器
I2CMDR	0x7909	I2C 模式寄存器
I2CISRC	0x790A	I2C 中断源寄存器
I2CPSC	0x790C	I2C 预分频器寄存器
I2CFFTX	0x7920	I2C FIFO 发送寄存器
I2CFFRX	0x7921	I2C FIFO 接收寄存器
I2CRSR	-	I2C 接收移位寄存器 (不可访问 CPU)
I2CXSR	-	I2C 发送移位寄存器 (不可访问 CPU)



## 4.13 GPIO MUX

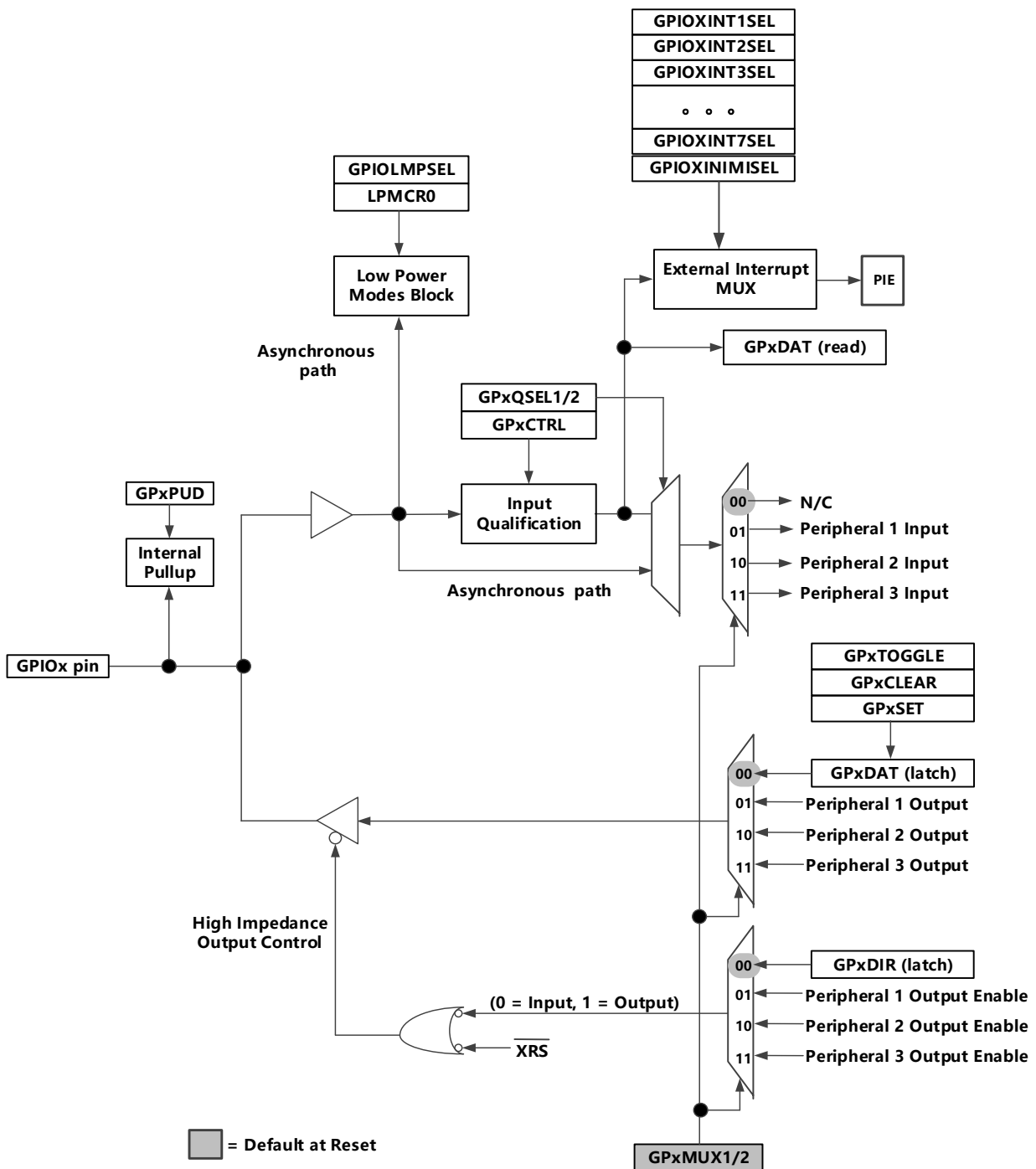
在 AVP32F335 器件上，除了提供独立的引脚位分裂 I/O 功能外，GPIO MUX 还可以将最多 3 个独立的外设信号复用在单一的一个 GPIO 引脚上。每个引脚的 GPIO MUX 方框图显示在图 4-18 中。由于 I2C 引脚的开漏功能，这些引脚的 GPIO MUX 方框图是不同的。

---

### 注

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

---



- A. 代表端口，A 或 B。例如，GPxDIR 是指 GPADIR 和 GPBDIR 寄存器，这取决于所选择的特定 GPIO 引脚。
- B. 在相同的存储器位置存取 GPxDAT 锁定/读取。
- C. 这是一个常用的 GPIO MUX 方框图。并不是所有选项都可用于所有 GPIO 引脚。

图 4-18 GPIO 复用

器件支持 88 个 GPIO 引脚。GPIO 控制和数据寄存器被映射到外设帧 1 以在寄存器上启用 32 位运行（连同 16 位运行）。表 4-15 显示了 GPIO 寄存器映射。

**表 4-15 GPIO 寄存器**

名称	地址	大小 (x16)	说明
<b>GPIO 控制寄存器 (受 EALLOW 保护)</b>			
GPACTRL	0x6F80	2	GPIO A 控制寄存器 (GPIO0 至 31)
GPAQSEL1	0x6F82	2	GPIO A 限定器选择 1 寄存器 (GPIO0 至 15)
GPAQSEL2	0x6F84	2	GPIO A 限定器选择 2 寄存器 (GPIO16 至 31)
GPAMUX1	0x6F86	2	GPIO A MUX 1 寄存器 ( GPIO0 至 15 )
GPAMUX2	0x6F88	2	GPIO A MUX 2 寄存器 ( GPIO16 至 31 )
GPADIR	0x6F8A	2	GPIO A 方向寄存器 (GPIO0 至 31)
GPAPUD	0x6F8C	2	GPIO A 上拉电阻器禁用寄存器 (GPIO0 至 GPIO31)
被保留	0x6F8E-0x6F8F	2	
GPBCTRL	0x6F90	2	GPIO B 控制寄存器 (GPIO32 至 63)
GPBQSEL1	0x6F92	2	GPIO B 限定器选择 1 寄存器 (GPIO32 至 47)
GPBQSEL2	0x6F94	2	GPIO B 限定器选择 2 寄存器 (GPIO48 至 63)
GPBMUX1	0x6F96	2	GPIO B MUX 1 寄存器 (GPIO32 至 47)
GPBMUX2	0x6F98	2	GPIO B MUX 2 寄存器 (GPIO48 至 63)
GPBDIR	0x6F9A	2	GPIO B 方向寄存器 (GPIO32 至 44)
GPBPUD	0x6F9C	2	GPIO B 上拉电阻器禁用寄存器 (GPIO38 至 44)
被保留	0x6F9E-0x6FA5	8	
GPCMUX1	0x6FA6	2	GPIO C MUX 1 寄存器 (GPIO64 至 79)
GPCMUX2	0x6FA8	2	GPIO C MUX 2 寄存器 (GPIO80 至 87)
GPCDIR	0x6FAA	2	GPIO C 方向寄存器 (GPIO64 至 87)
GPCPUD	0x6FAC	2	GPIO C 上拉电阻器禁用寄存器 (GPIO64 至 87)
被保留	0x6FAE-0x6FBF	18	
<b>GPIO 数据寄存器 (不受 EALLOW 保护)</b>			
GPADAT	0x6FC0	2	GPIO A 数据寄存器 ( GPIO0 至 31 )
GPASET	0x6FC2	2	GPIO A 数据设定寄存器 ( GPIO0 至 31 )
GPACLEAR	0x6FC4	2	GPIO A 数据清除寄存器 ( GPIO0 至 31 )
GPATOGGLE	0x6FC6	2	GPIO A 数据取反寄存器 ( GPIO0 至 31 )
GPBDAT	0x6FC8	2	GPIO B 数据寄存器 (GPIO32 至 63)
GPBSET	0x6FCA	2	GPIO B 数据设定寄存器 ( GPIO32 至 63 )
GPBCLEAR	0x6FCC	2	GPIO B 数据清除寄存器 ( GPIO32 至 63 )
GPBTOGGLE	0x6FCE	2	GPIO B 数据取反寄存器 ( GPIO32 至 63 )
GPCDAT	0x6FD0	2	GPIO C 数据寄存器 (GPIO64 至 87)
GPCSET	0x6FD2	2	GPIO C 数据设定寄存器 ( GPIO64 至 87 )
GPCCLEAR	0x6FD4	2	GPIO C 数据清除寄存器 ( GPIO64 至 87 )
GPCTOGGLE	0x6FD6	2	GPIO C 数据取反寄存器 ( GPIO64 至 87 )
被保留	0x6FD8-0x6FDF	8	
<b>GPIO 中断和低功耗模式选择寄存器 (受 EALLOW 保护)</b>			
GPIOXINT1SEL	0x6FE0	1	XINT1 GPIO 输入选择寄存器 (GPIO0 至 31)
GPIOXINT2SEL	0x6FE1	1	XINT2 GPIO 输入选择寄存器 (GPIO0 至 GPIO31)
GPIOXNIMISEL	0x6FE2	1	XNMI GPIO 输入选择寄存器 (GPIO0 至 GPIO31)
GPIOXINT3SEL	0x6FE3	1	XINT3 GPIO 输入选择寄存器 (GPIO32 至 GPIO63)
GPIOXINT4SEL	0x6FE4	1	XINT4 GPIO 输入选择寄存器 (GPIO32 至 GPIO63)
GPIOXINT5SEL	0x6FE5	1	XINT5 GPIO 输入选择寄存器 (GPIO32 至 GPIO63)
GPIOXINT6SEL	0x6FE6	1	XINT6 GPIO 输入选择寄存器 (GPIO32 至 GPIO63)

GPIOXINT7SEL	0x6FE7	1	XINT7 GPIO 输入选择寄存器 (GPIO32 至 GPIO63)
GPIO_LPMSEL	0x6FE8	2	LPM GPIO 选择寄存器 (GPIO0 至 GPIO32)
被保留	0x6FEA-0x6FFF	22	

**表 4-16 GPIO-A 复用器外设选择矩阵**

寄存器位		外设选择				
GPADIR GPADAT GPASET GPACLR GPATOGGLE	GPAMUX1 GPAQSEL1	GPIOx	PER1	PER2	PER3	
		GPAMUX1 = 0, 0	GPAMUX1 = 0, 1	GPAMUX1 = 1, 0	GPAMUX1 = 1, 1	
QUALPRD0	0	1, 0	GPIO0 (I/O)	EPWM1A (O)	被保留	被保留
	1	3, 2	GPIO1 (I/O)	EPWM1B (O)	ECAP6 (I/O)	MFSRB (I/O)
	2	5, 4	GPIO2 (I/O)	EPWM2A (O)	被保留	被保留
	3	7, 6	GPIO3 (I/O)	EPWM2B (O)	ECAP5 (I/O)	MCLKRB (I/O)
	4	9, 8	GPIO4 (I/O)	EPWM3A (O)	被保留	被保留
	5	11, 10	GPIO5 (I/O)	EPWM3B (O)	MFSRA (I/O)	ECAP1 (I/O)
	6	13, 12	GPIO6 (I/O)	EPWM4A (O)	EPWMSYNCI (I)	EPWMSYNCO (O)
QUALPRD1	7	15, 14	GPIO7 (I/O)	EPWM4B (O)	MCLKRA (I/O)	ECAP2 (I/O)
	8	17, 16	GPIO8 (I/O)	EPWM5A (O)	CANTXB (O)	ADCSOCAO(O)
	9	19, 18	GPIO9 (I/O)	EPWM5B (O)	SCITXDB (O)	ECAP3 (I/O)
	10	21, 20	GPIO10 (I/O)	EPWM6A (O)	CANRXB (I)	ADCSOCBO(O)
	11	23, 22	GPIO11 (I/O)	EPWM6B (O)	SCIRXDB (I)	ECAP4 (I/O)
	12	25, 24	GPIO12 (I/O)	TZ1(I)	CANTXB (O)	MDXB (O)
	13	27, 26	GPIO13 (I/O)	TZ2(I)	CANRXB (I)	MDRB (I)
QUALPRD2	14	29, 28	GPIO14 (I/O)	TZ3(I)/XHOLD(I)	SCITXDB (O)	MCLKXB (I/O)
	15	31, 30	GPIO15 (I/O)	TZ4(I)/XHOLDA(O)	SCIRXDB (I)	MFSXB (I/O)
		<b>GPAMUX2 GPAQSEL2</b>	<b>GPAMUX2 = 0, 0</b>	<b>GPAMUX2 = 0, 1</b>	<b>GPAMUX2 = 1, 0</b>	<b>GPAMUX2 = 1, 1</b>
	16	1, 0	GPIO16 (I/O)	SPISIMOA (I/O)	CANTXB (O)	TZ5(I)
	17	3, 2	GPIO17 (I/O)	SPISOMIA (I/O)	CANRXB (I)	TZ6(I)
	18	5, 4	GPIO18 (I/O)	SPICLKA (I/O)	SCITXDB (O)	CANRXA (I)
	19	7, 6	GPIO19 (I/O)	SPISTEA(I/O)	SCIRXDB (I)	CANTXA (O)
QUALPRD3	20	9, 8	GPIO20 (I/O)	EQEP1A (I)	MDXA (O)	CANTXB (O)
	21	11, 10	GPIO21 (I/O)	EQEP1B (I)	MDRA (I)	CANRXB (I)
	22	13, 12	GPIO22 (I/O)	EQEP1S (I/O)	MCLKXA (I/O)	SCITXDB (O)
	23	15, 14	GPIO23 (I/O)	EQEP1I (I/O)	MFSXA (I/O)	SCIRXDB (I)
	24	17, 16	GPIO24 (I/O)	ECAP1 (I/O)	EQEP2A (I)	MDXB (O)
	25	19, 18	GPIO25 (I/O)	ECAP2 (I/O)	EQEP2B (I)	MDRB (I)
	26	21, 20	GPIO26 (I/O)	ECAP3 (I/O)	EQEP2I (I/O)	MCLKXB (I/O)
QUALPRD3	27	23, 22	GPIO27 (I/O)	ECAP4 (I/O)	EQEP2S (I/O)	MFSXB (I/O)
	28	25, 24	GPIO28 (I/O)	SCIRXDA (I)	XZCS6(O)	
	29	27, 26	GPIO29 (I/O)	SCITXDA (O)	XA19(O)	
	30	29, 28	GPIO30 (I/O)	CANRXA (I)	XA18(O)	
	31	31, 30	GPIO31 (I/O)	CANTXA (O)	XA17(O)	

**表 4-17 GPIO-B 复用器外设选择矩阵**

寄存器位		外设选择				
GPBDIR GPBDAT GPBSET GPBCLR GPBTOGGLE	GPBMUX1 GPBQSEL1	GPIOx GPBMUX1 = 0,0	PER1 GPBMUX1 = 0,1	PER2 GPBMUX1 = 1,0	PER3 GPBMUX1 = 1,1	
QUALPRD0	0	1, 0	GPIO32(I/O)	SDAA (I/OC)(1)	EPWMSYNCI (I)	ADCSOCAO(O)
	1	3, 2	GPIO33(I/O)	SCLA (I/OC)(1)	EPWMSYNCO (O)	ADCSOCBO(O)
	2	5, 4	GPIO34 (I/O)	ECAP1 (I/O)	XREADY (I)	
	3	7, 6	GPIO35(I/O)	SCITXDA (O)	XR/W(O)	
	4	9, 8	GPIO36(I/O)	SCIRXDA (I)	XZCS0(O)	
	5	11, 10	GPIO37(I/O)	ECAP2 (I/O)	XZCS7(O)	
	6	13, 12	GPIO38(I/O)		XWE0(O)	
QUALPRD1	7	15, 14	GPIO39(I/O)		XA16(O)	
	8	17, 16	GPIO40(I/O)		XA0/XWE1(O)	
	9	19, 18	GPIO41 (I/O)		XA1(O)	
	10	21, 20	GPIO42(I/O)		XA2(O)	
	11	23, 22	GPIO43(I/O)		XA3(O)	
	12	25, 24	GPIO44(I/O)		XA4(O)	
	13	27, 26	GPIO45 (I/O)		XA5(O)	
	14	29, 28	GPIO46(I/O)		XA6(O)	
	15	31, 30	GPIO47(I/O)		XA7(O)	
		<b>GPBMUX2 GPBQSEL2</b>	<b>GPBMUX2 = 0, 0</b>	<b>GPBMUX2 = 0, 1</b>	<b>GPBMUX2 = 1, 0</b>	<b>GPBMUX2 = 1, 1</b>
QUALPRD2	16	1, 0	GPIO48 (I/O)	ECAP5 (I/O)	XD31(I/O)	
	17	3, 2	GPIO49 (I/O)	ECAP6 (I/O)	XD30 (I/O)	
	18	5, 4	GPIO50 (I/O)	EQEP1A (I)	XD29 (I/O)	
	19	7, 6	GPIO51 (I/O)	EQEP1B (I)	XD28 (I/O)	
	20	9, 8	GPIO52 (I/O)	EQEP1S (I/O)	XD27 (I/O)	
	21	11, 10	GPIO53 (I/O)	EQEP1I (I/O)	XD26 (I/O)	
	22	13, 12	GPIO54 (I/O)	SPISIMOA (I/O)	XD25 (I/O)	
	23	15, 14	GPIO55 (I/O)	SPISOMIA (I/O)	XD24 (I/O)	
QUALPRD3	24	17, 16	GPIO56 (I/O)	SPICLKA (I/O)	XD23 (I/O)	
	25	19, 18	GPIO57 (I/O)	SPISTEA(I/O)	XD22(I/O)	
	26	21, 20	GPIO58 (I/O)	MCLKRA (I/O)	XD21 (I/O)	
	27	23, 22	GPIO59 (I/O)	MFSRA (I/O)	XD20 (I/O)	
	28	25, 24	GPIO60(I/O)	MCLKRB (I/O)	XD19(I/O)	
	29	27, 26	GPIO61 (I/O)	MFSRB (I/O)	XD18 (I/O)	
	30	29, 28	GPIO62 (I/O)	SCIRXDC (I)	XD17 (I/O)	
	31	31, 30	GPIO63 (I/O)	SCITXDC (O)	XD16 (I/O)	

(1) I = 输入, O = 输出, OC = 开漏

表 4-18 GPIO-C 复用器外设选择矩阵

寄存器位		外设选择		
		GPCMUX1	GPIOx 或 PER1 GPCMUX1 = 0, 0 or 0, 1	PER2 或 PER3 GPCMUX1 = 1, 0 or 1, 1
在其中	0	1, 0	GPIO64 (I/O)	XD15 (I/O)
	1	3, 2	GPIO65 (I/O)	XD14 (I/O)
	2	5, 4	GPIO66 (I/O)	XD13 (I/O)
	3	7, 6	GPIO67 (I/O)	XD12 (I/O)
	4	9, 8	GPIO68 (I/O)	XD11 (I/O)
	5	11, 10	GPIO69 (I/O)	XD10 (I/O)
	6	13, 12	GPIO70 (I/O)	XD9 (I/O)
	7	15, 14	GPIO71 (I/O)	XD8 (I/O)
在其中	8	17, 16	GPIO72 (I/O)	XD7 (I/O)
	9	19, 18	GPIO73 (I/O)	XD6 (I/O)
	10	21, 20	GPIO74 (I/O)	XD5 (I/O)
	11	23, 22	GPIO75 (I/O)	XD4 (I/O)
	12	25, 24	GPIO76 (I/O)	XD3 (I/O)
	13	27, 26	GPIO77 (I/O)	XD2 (I/O)
	14	29, 28	GPIO78 (I/O)	XD1 (I/O)
	15	31, 30	GPIO79 (I/O)	XD0 (I/O)
		<b>GPCMUX2</b>	<b>GPCMUX2 = 0, 0 或 0, 1</b>	<b>GPCMUX2 = 1, 0 或 1, 1</b>
在其中	16	1, 0	GPIO80 (I/O)	XA8 (O)
	17	3, 2	GPIO81 (I/O)	XA9 (O)
	18	5, 4	GPIO82 (I/O)	XA10 (O)
	19	7, 6	GPIO83 (I/O)	XA11(O)
	20	9, 8	GPIO84 (I/O)	XA12 (O)
	21	11, 10	GPIO85 (I/O)	XA13 (O)
	22	13, 12	GPIO86 (I/O)	XA14 (O)
	23	15, 14	GPIO87 (I/O)	XA15 (O)

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

- 只同步至 SYSCLKOUT (GPxQSEL1/2=0,0)：这是复位时所有 GPIO 引脚的默认模式并且它只是将输入信号同步至系统时钟 (SYSCLKOUT)。
- 使用采样窗口的限定条件 (GPxQSEL1/2=0, 1 和 1, 0)：这个模式中，在与系统时钟 (SYSCLKOUT)同步后，输入信号在输入被允许改变前，被一定数量的周期所限定。
- 采样周期由 GPxCTRL 寄存器内的 QUALPRD 位所指定并且可在每组 8 个信号中进行配置。它为输入信号采样指定了多个 SYSCLKOUT 周期。一个采样窗口为可包含 3 次或 6 次采样，并且只有当所有采样与图 4-19 (对于 6 次采样) 中所显示的一样时 (全 0 或者全 1)，输出才会改变。

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

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

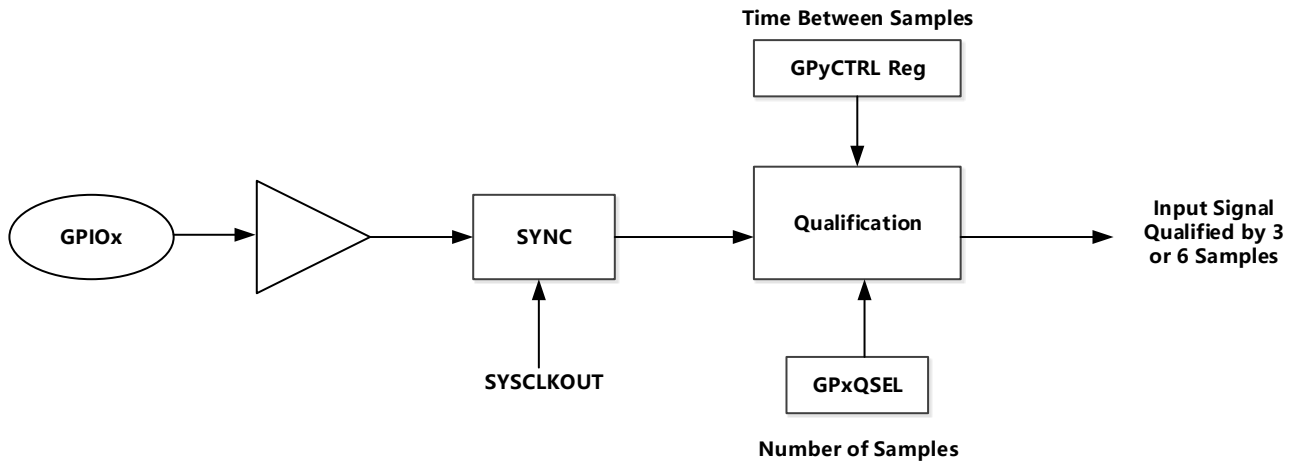
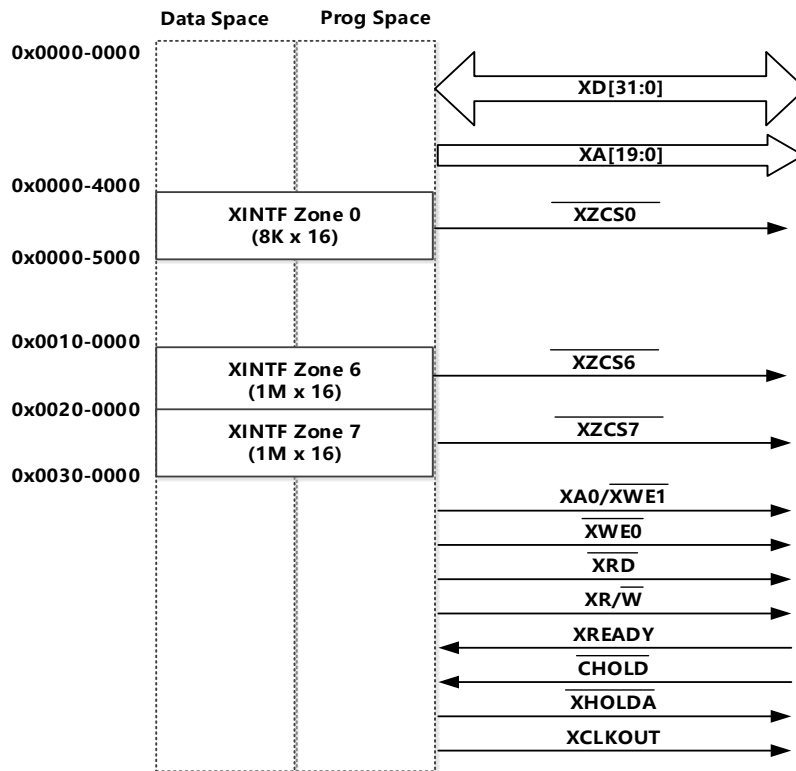


图 4-19 使用采样窗口的限定

## 4.14 外部接口 (XINTF)

本节给出了在 AVP32F335 器件上执行的外部接口 (XINTF) 的方框图。XINTF 是一个非复用的异步总线, 它类似于 ADP32F12 XINTF。XINTF 被映射到图 4-20 中所示的三个固定区域。



- A. 每个区域可被设定为具有不同的等待状态、建立和保持时序，并且当执行到一个特定区域的访问时，由区域芯片选择切换支持。这些特性可实现到很多外部存储器和外设的无缝连接。
- B. 1-5 区被保留用于将来的扩展。
- C. 区域 0, 6, 和 7 一直被启用。

图 4-20 外部接口方框图

图 4-21 和图 4-22 展示了典型 16 位和 32 位数据总线 XINTF 连接，说明了 XA0 和 XWE1 的信号功能是如何根据具体的配置变化的。表 4-19 定义了 XINTF 配置和控制寄存器。

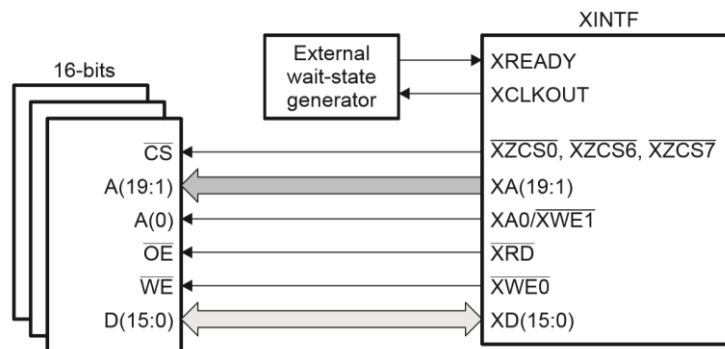


图 4-21 典型的 16 位数据总线 XINTF 连接



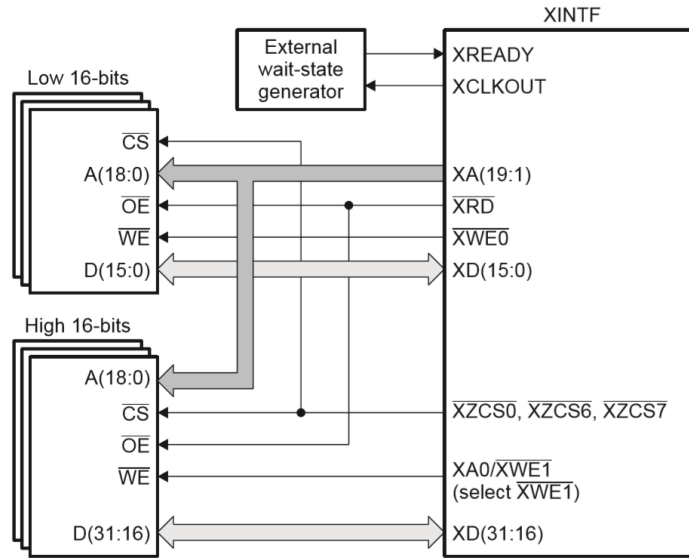


图 4-22 典型的32 位数据总线XINTF 连接

表 4-19 XINTF 配置和控制寄存器映射

名称	地址	大小(x 16)	说明
XTIMING0	0x00-0B20	2	XINTF 定时寄存器, 区域0
XTIMING6 <sup>(1)</sup>	0x00-0B2C	2	XINTF 定时寄存器, 区域6
XTIMING7	0x00-0B2E	2	XINTF 定时寄存器, 区域7
XINTCNF2 <sup>(2)</sup>	0x00-0B34	2	XINTF 配置寄存器
XBANK	0x00-0B38	1	XINTF 组控制寄存器
XREVISION	0x00-0B3A	1	XINTF 修订版本寄存器
XRESET	0x00-0B3D	1	XINTF 复位寄存器

(1) XTIMING1-XTIMING5 为将来的扩展所保留, 目前没有使用。

(2) XINTCNF1 被保留, 目前没有使用。

## 5 开发支持

进芯电子(Advchip) 为 ADP32Fx 产品提供了大量的开发工具，其中包括评估处理器性能、生成代码、开发算法执行的工具，且完全集成以及调试软件和硬件模块。

下面的软件硬件工具支持基于 AVP32F335 的应用开发：

### 软件开发工具

- Code Composer Studio™ 集成开发环境(IDE)
  - C/C++ 编译器
  - 代码生成工具
  - 汇编器/连接器
  - 周期精确模拟器
- 应用算法
- 示例应用代码

### 硬件开发工具

- AVP32F335 eZdsp
- 支持基于JTAG 的仿真器 - XDS510 , XDS110V1 , XDS100V1
- 通用5V 直流电源，推荐5V/1A以上
- 文档和仿真器连接线

## 6 电气规范

### 6.1 最大绝对额定值<sup>(1)(2)</sup>

电源电压范围 $V_{DDIO}$ , $V_{DD1A33}$	相对于 $V_{SS}$	-0.3V 至 4.6V
电源电压范围 $V_{DDA2}$ , $V_{DDAIO}$	相对于 $V_{SSA}$	-0.3 V 至 4.6V
电源电压范围 $V_{DD}$	相对于 $V_{SS}$	-0.3 V 至 2.5V
电源电压范围 $V_{DD1A18}$ , $V_{DD2A18}$	相对于 $V_{SSA}$	-0.3V 至 4.6V
电源电压范围 $V_{SSA2}$ , $V_{SSAIO}$ , $V_{SSA1AGND}$ , $V_{SS2AGND}$	相对于 $V_{SS}$	-0.3V 至 0.3V
输入电压范围, $V_{IN}$		-0.3V 至 4.6V
输出电压范围, $V_O$		-0.3V 至 4.6V
输入钳制电流, $I_{IK}(V_{IN} < 0$ 或者 $V_{IN} > V_{DDIO})$ <sup>(3)</sup>		$\pm 20\text{mA}$
输出钳制电流, $I_{OK}(V_O < 0$ 或者 $V_O > V_{DDIO})$		$\pm 20\text{mA}$
结温范围, $T_J$ <sup>(4)</sup>		-40°C 至 150°C
贮存温度范围, $T_{stg}$ <sup>(4)</sup>		-65°C 至 150°C

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

(2) 所有电压值都是相对于  $V_{SS}$  的值，除非额外注明。

(3) 每个引脚上的持续钳制电流为  $\pm 2\text{mA}$ 。

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

## 6.2 建议的运行条件

在自然通风条件下的工作温度范围内（除非另有说明）

		最小值	标称值	最大值	单位
器件电源电压, I/O, VDDIO		3.135	3.3	3.465	V
器件电源电压CPU, VDD	器件操作@ 150MHz	1.71	1.8	1.89	V
	器件操作@ 100MHz	1.53	1.7	1.87	
电源接地, VSS, VSSIO, VSSAIO, VSSA2, VSS1AGND, VSS2AGND			0		V
ADC 电源电压(3.3V), VDDA2, VDDAIO		3.135	3.3	3.465	V
ADC 电源电压, VDD1A18, VDD2A18	器件操作@ 150MHz	1.71	1.8	1.89	V
	器件操作@ 100MHz	1.53	1.7	1.87	
VDD1A33		3.135	3.3	3.465	V
器件时钟频率(系统时钟), fSYSCLKOUT	AVP32F335	2		150	MHz
高电平输入电压, VIH	除X1 之外的所有输入	2		VDDIO	V
	X1	0.7 * VDD		VDD	
低电平输入电压, VIL	除X1 之外的所有输入			0.8	V
	X1			0.3 * VDD+0.05	
高电平输出源电流, VOH=2.4V, IOH	除组 2 之外的所有I/O			-4	mA
	组 2(1)			-8	
低电平输出灌电流, VOL=VOL(最大值), IOL	除组 2 之外的所有I/O			4	mA
	组 2(1)			8	
环境温度, TA	A版本	-40		85	°C
	S版本	-40		125	°C
结温, Tj				125	°C

(1) 组 2 引脚如下: GPIO28, GPIO29, GPIO30, GPIO31, TDO, XCLKOUT, EMU0, EMU1, XINTF pins, GPIO35-87,XRD。

## 6.3 电气特性

在推荐的运行条件下（除非额外注明）

参数		测试条件		最小值	典型值	最大值	单位
VoH 高电平输出电压		IOH=IOH 最大值		2.4			V
		IOH=50μA		VDDIO-0.2			
VoL 低电平输出电压		IOL=IOL 最大值				0.4	V
IIL 输入电流 (低电平)	上拉电阻器被启用	VDDIO=3.3V , VIN=0V	所有带上拉 IO (除 XRS 引脚)	-80	-145	-150	μA
			XRS 引脚		-220		
	下拉电阻器被启用	VDDIO=3.3V , VIN=0V				±5	
IIH 输入电流 (高电平)	上拉电阻器被启用	VDDIO=3.3V , VIN= VDDIO				±5	μA
	下拉电阻器被启用	VDDIO=3.3V , VIN= VDDIO		28	37	80	
IoZ 输出电流, 上拉电阻器或者下拉电阻器被禁用		VO=VDDIO 或者 0V				±5	μA
Ci 输入电容					2		pF

(1) 除 TRST, TDO, XCLKOUT, XCLKIN, X1, X2 外其余数字 IO 口均带上拉电阻, TRST, TDO, XCLKOUT, XCLKIN 内部带下拉电阻。

## 6.4 流耗

**表 6-1 AVP32F335 在 150MHz SYSCLKOUT 运行条件下的流耗**

模式	测试条件	I <sub>DD</sub>		I <sub>DDIO</sub> (1)		I <sub>DDA18</sub> (3)		I <sub>DDA33</sub> (4)	
		典型值(5)	最大值	典型值(5)	最大值	典型值(5)	最大值	典型值(5)	最大值
可用 (闪存) (6)	下列的外设时钟被启用： <ul style="list-style-type: none"> <li>ePWM1, ePWM2, ePWM3, ePWM4, ePWM5, ePWM6</li> <li>eCAP1, eCAP2, eCAP3, eCAP4, eCAP5, eCAP6</li> <li>eQEP1, eQEP2</li> <li>eCAN-A</li> <li>SCI-A, SCI-B (FIFO 模式)</li> <li>SPI-A (FIFO 模式)</li> <li>ADC</li> <li>IC2</li> <li>CPU 定时器 0, CPU 定时器 1, CPU 定时器 2</li> </ul> 所有 PWM 引脚被切换至 150KHz。 所有 I/O 引脚保持未连接状态(7)	140mA	160mA	20mA	50mA	25mA	30mA	5mA	8mA
IDLE (闲置)	闪存被断电。 XCLKOUT 被关闭。下列的外设时钟被启用： <ul style="list-style-type: none"> <li>eCAN-A</li> <li>SCI-A</li> <li>SPI-A</li> </ul>	20mA	30mA	60μA	120μA	180μA	190μA	3μA	10μA
STANDBY (待机)	闪存被断电。 外设时钟被关闭。	4mA	8mA	60μA	120μA	75μA	90μA	3μA	10μA
HALT(8)	闪存被断电。 外设时钟被关闭。 输入时钟被禁用。(9)	200μA		60μA	120μA	3μA	30μA	3μA	10μA

(1) IDDIO电流取决于 I/O 引脚上的电气负载。

(2) IDDA18包括进入VDD1A18和VDD2A18引脚的电流。为了实现所显示的用于IDLE, STANDBY, 和HALT 的IDDA18电流, 必须通过写入 PCLKCR0 寄存器来明确关闭到 ADC 模块的时钟。

(3) IDDA33包括进入VDDA2和VDDAIO 和VDD1A33引脚的电流。

(4) TYP 数适用于常温和标称电压。125°C 时的最大值, 和最大电压(V<sub>DD</sub> = 2.0V; V<sub>DDIO</sub>, V<sub>DD1A33</sub>, V<sub>DDA</sub>=3.6V)。

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

(6) 下面的操作在环路内完成：

- 数据从SCI-A, SCI-B, SPI-A, McBSP-A, 和eCAN-A 端口连续发出。
- 执行乘法/加法运算。

- 看门狗被复位。
  - ADC 正在执行持续转换。ADC 中的数据通过DMA 传送到SARAM。
  - 执行XINTF 的32 位读/写。
  - GPIO19 被接通。
- (7) HALT 模式IDD电流将随温度非线性增加。
- (8) 如果一个石英晶振或者陶瓷谐振器被用作时钟源，HALF 模式将关闭内部振荡器。

### 注

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

## 6.4.1 减少流耗

AVP32F335 包含一个减少器件流耗的方法。通过关闭到任意不在一个指定应用中使用的外设模块的时钟可实现流耗减少。此外，可利用这三个低功耗模式的任意一个来进一步减少流耗。[表6-2](#)表明了通过关闭到不同外设的时钟所实现的流耗减少的典型值。

**表 6-2 不同外设的典型流耗 ( 150MHz 上时 ) (1)**

外设 模块	IDD 电流 减少/模块(mA) (2)
ADC	6.5 (3)
I <sup>2</sup> C	1
eQEP	1
ePWM	2
eCAP	2.5
SCI	1.5
SPI	2
eCAN	4
McBSP	2
CPU - 定时器	0.5
XINTF	2.5 (4)
DMA	3
FPU	6.5

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

(2) 对于具有多个实例的外设，依照模块引用电流。例如，为 ePWM 所引出的 5mA 电流是用于一个 ePWM 模块。

(3) 这个数字代表了取自ADC 模块数字部分的电流。关闭ADC 模块的时钟也将消除取自ADC (IDD<sub>DA18</sub>) 模拟部分汲取的电流。

(4) 运行 XINTF 总线对 IDDIO 电流有明显的影。基于以下原因，这将大大增加此电流：

- 多少个地址/数据引脚从一个周期切换到另一个
- 它们切换的速度有多快
- 使用的接口是 16 位还是 32 位以及
- 这些引脚上的负载。

下面是进一步减少流耗的其它方法：

- 当 XCLKOUT 被关闭时， $I_{DDIO}$  功耗减少了 3mA (典型值)。
- 基础  $I_{DD}$  电流 (此电流是指当内核在无外设被启用的情况下执行一个仿真环路时的电流) 为 55mA (典型值)。
- 为了达到一个指定应用所需的  $I_{DD}$  电流，外设 (由应用启用) 汲取的电流必须被添加到基础  $I_{DD}$  电流上。



### 6.4.2 流耗图

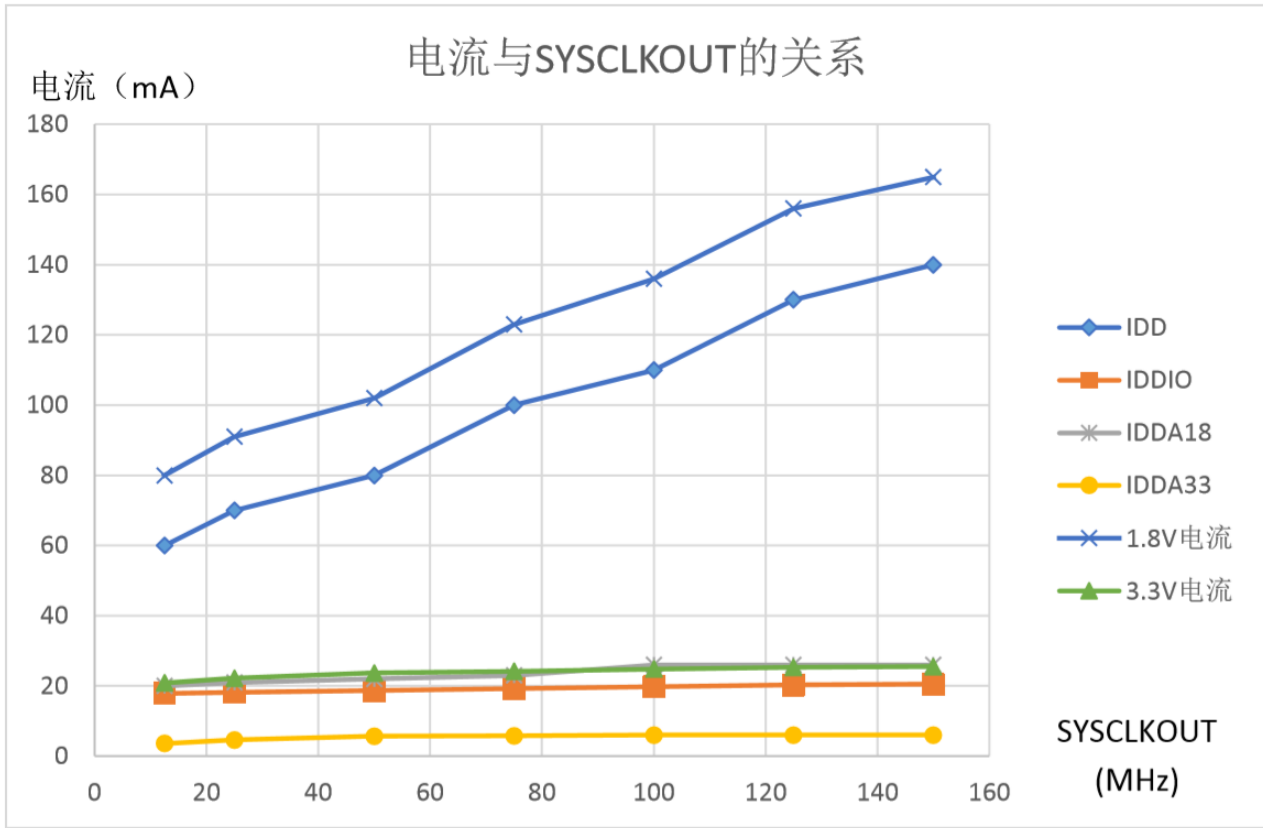


图 6-1 典型运行电流与频率间的关系 (AVP32F335)

- A.  $I_{DD}$  代表数字 1.8V 的电流。
- B.  $I_{DDIO}$  代表数字 3.3V 的电流。
- C.  $I_{DDA18}$  代表模拟 1.8V 的电流，它包括  $V_{DD1A18}$ 和  $V_{DD2A18}$ 的电流。
- D.  $I_{DDA33}$  代表模拟 1.8V 的电流，它包括  $V_{DDA10}$ 和  $V_{DDA2}$ 和  $V_{DD1A33}$ 的电流。
- E. 1.8V 电流：代表数字 1.8V 和模拟 1.8V 的总电流。
- F. 3.3V 电流：代表数字 3.3V 和模拟 3.3V 的总电流。

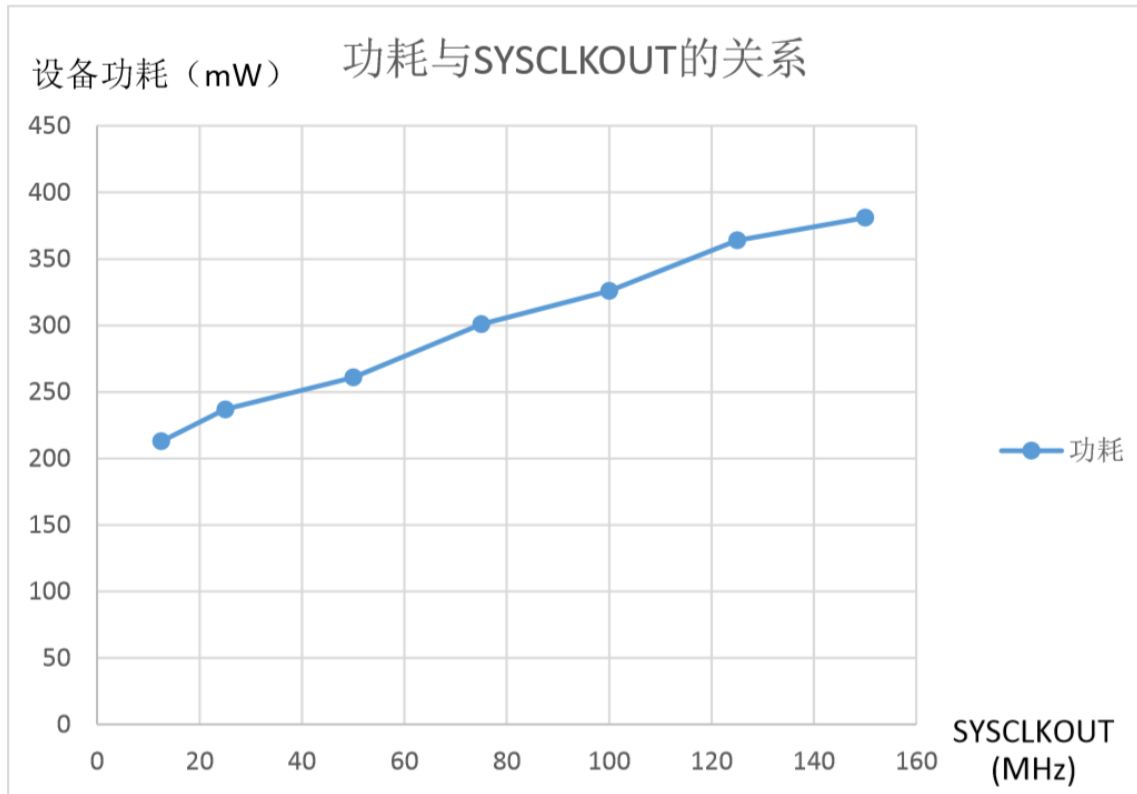


图 6-2 典型运行功率与频率间的关系 (AVP32F335)

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

### 6.4.3 散热设计考虑

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

## 6.5 针对 DSP 的无信号缓冲的仿真器连接

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

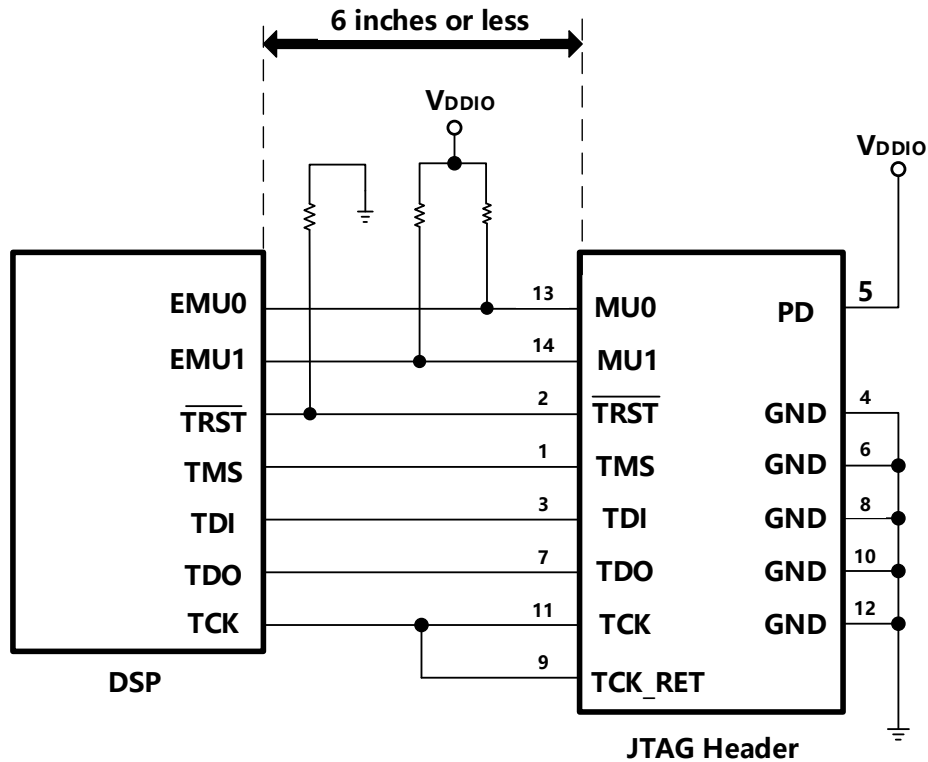


图 6-3 针对 DSP 的无信号缓冲的仿真器连接

## 6.6 时序参数符号

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

### 小写下标和它们的含意：

- a 访问时间
- c 周期时间（周期）
- d 延迟时间
- f 下降时间
- h 保持时间
- r 上升时间
- su 建立时间
- t 转换时间
- v 有效时间
- w 脉冲持续时间（宽度）

### 字母和它们的含意：

- H 高
- L 低
- V 有效
- X 未知、改变、或者无关电平
- Z 高阻抗

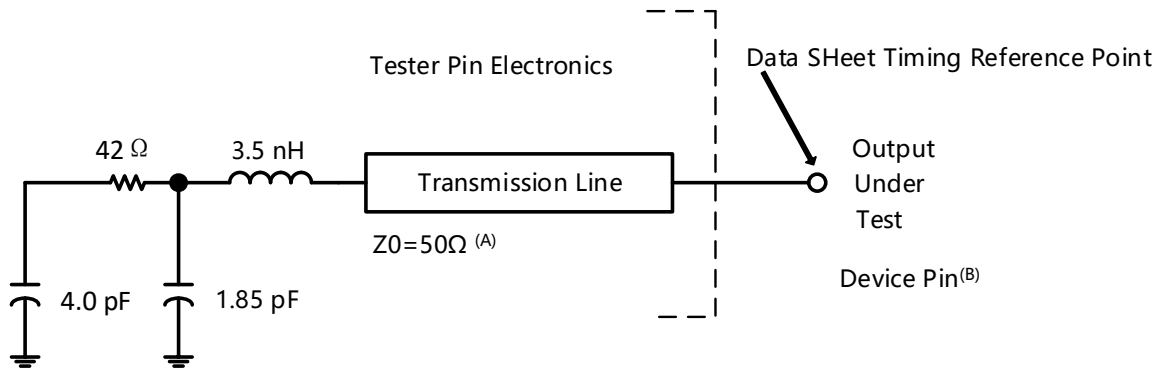
### 6.6.1 定时参数的通用注释

所有 AVP32F335 器件的输出信号（包括 XCLKOUT）取自一个内部时钟，这样，对于一个指定半周期的所有输出转换在一个互相之间相对最小转换率时发生。

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

### 6.6.2 测试负载电路

这个测试负载电路用于测量这个文档中提供的所有开关特性。



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

图 6-4 3.3V 测试负载电路

### 6.6.3 器件时钟表

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

表 6-3 片上设备时钟的周期特征（150MHz 主时钟）

		最小值	标称值	最大值	单位
片载振荡器时钟	tc(OSC), 周期时间	28.6		50	ns
	频率	20		35	MHz
XCLKIN(1)	tc(CI), 周期时间	6.67		250	ns
	频率	4		150	MHz
SYSCLKOUT	tc(SCO), 周期时间	6.67		500	ns
	频率	2		150	MHz
XCLKOUT	t(XCO), 周期时间	6.67		2000	ns
	频率	0.5		150	MHz
HSPCLK(2)	tc(LCO), 周期时间	6.67	13.3 (3)		ns
	频率		75(3)	150	MHz
LSPCLK(2)	tc(LCO), 周期时间	13.3	26.7(3)		ns

ADC 时钟	频率	37.5(3) 75 (4)	MHz
	t <sub>c</sub> (ADCCLK), 周期时间	133	ns
	频率	7.5	MHz

- (1) 如果使用一个 1.9V 振荡器，这也适用于 X1 引脚。
- (2) 更低的 LSPCLK 和 HSPCLK 将减少器件功耗。
- (3) 如果 SYSCLKOUT=150MHz，此值为缺省值。
- (4) 尽管 LSPCLK 能够达到 100MHz，但由于对于 150MHz 器件，最小有效“低速外设时钟预分频寄存器”的值是“2”，所以它的额定值为 75MHz。

**表 6-4 片上设备时钟的周期特征 (100MHz 主时钟)**

		最小值	标称值	最大值	单位
片载振荡器时钟	t <sub>c</sub> (OSC), 周期时间	28.6	50		ns
	频率	20		35	MHz
XCLKIN(1)	t <sub>c</sub> (CI), 周期时间	10		250	ns
	频率	4		100	MHz
SYSCLKOUT	t <sub>c</sub> (SCO), 周期时间	10		500	ns
	频率	2		100	MHz
XCLKOUT	t <sub>c</sub> (XCO), 周期时间	10		2000	ns
	频率	0.5		100	MHz
HSPCLK(2)	t <sub>c</sub> (HCO), 周期时间	10	20 (3)		ns
	频率	50(3)		100	MHz
LSPCLK(2)	t <sub>c</sub> (LCO), 周期时间	10	40(3)		ns
	频率	25(3)		100	MHz
ADC 时钟	t <sub>c</sub> (ADCCLK), 周期时间		133		ns
	频率		7.5		MHz

- (1) 如果使用一个 1.8V 振荡器，这也应用于 X1 引脚。
- (2) 更低的 LSPCLK 和 HSPCLK 将减少器件功耗。
- (3) 如果 SYSCLKOUT=100MHz，此值为缺省值。

## 6.7 时钟要求和特性

**表 6-5 输入时钟频率**

参数		最小值	标称值	最大值	单位	
f <sub>x</sub> 输入时钟频率	谐振器 (X1/X2)	20		35	MHz	
	晶振 (X1/X2)	20		35	MHz	
	外部振荡器/时钟源 (XCLKIN 或者 X1 引脚)	150MHz	4		150	MHz
		100MHz	4		100	MHz
f <sub>i</sub> 跛行模式 SYSCLKOUT 频率范围 (/2 启用时)			1-5		MHz	

**表 6-6 XCLKIN 时序要求- PLL 被启用**

编号		最小值	最大值	单位
<b>C8</b>	t <sub>c(CI)</sub> 周期时间, XCLKIN	33.3	200	ns
<b>C9</b>	t <sub>f(CI)</sub> 下降时间, XCLKIN <sup>(1)</sup>		6	ns

<b>C10</b>	$t_{r(CI)}$ 上升时间, XCLKIN <sup>(1)</sup>		6	ns
<b>C11</b>	$t_{w(CIL)}$ 脉冲持续时间, XCLKIN 低电平作为 $t_{c(CI)}$ <sup>(1)</sup> 的一部分的时间	45	55	%
<b>C12</b>	$t_{w(CIH)}$ 脉冲持续时间, XCLKIN 高电平作为 $t_{c(CI)}$ <sup>(1)</sup> 的一部分的时间	45	55	%

(1) 这也被应用到 X1 引脚。

**表 6-7 XCLKIN 时序需求-- PLL 被禁用**

编号	参数	最小值	最大值	单位	
<b>C8</b>	$t_{c(CI)}$ 周期时间, XCLKIN	150MHz 器件	6.67	250	ns
		100MHz 器件	10	250	
<b>C9</b>	$t_{f(CI)}$ 下降时间, XCLKIN	高达 30 MHz		6	ns
		30MHz 至 150MHz		2	
<b>C10</b>	$t_{r(CI)}$ 上升时间, XCLKIN	高达 30 MHz		6	ns
		30MHz 至 150MHz		2	
<b>C11</b>	$t_{w(CIL)}$ 脉冲持续时间, XCLKIN 低电平作为 $t_{c(CI)}$ <sup>(1)</sup> 的一部分的时间	45	55	%	
<b>C12</b>	$t_{w(CIH)}$ 脉冲持续时间, XCLKIN 高电平作为 $t_{c(CI)}$ <sup>(1)</sup> 的一部分的时间	45	55	%	

(1) 这也被应用到 X1 引脚。

表 3-18中显示了可能的配置模式

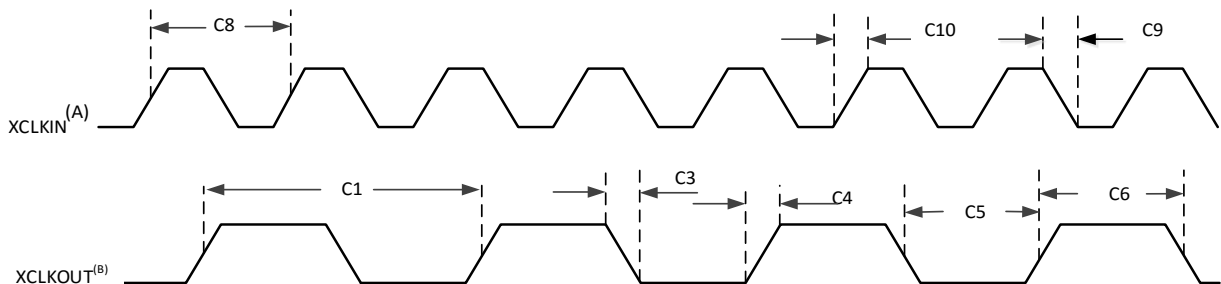
**表 6-8 XCLKOUT 开关特性 ( PLL 旁路或者被禁用 )<sup>(1) (2)</sup>**

编号	参数	最小值	典型值	最大值	单位
<b>C1</b>	$t_{c(XCO)}$ 周期时间, XCLKOUT	150MHz 器件	6.67		ns
		100MHz 器件	10		ns
<b>C3</b>	$t_{f(XCO)}$ 下降时间, XCLKOUT		2	5	ns
<b>C4</b>	$t_{r(XCO)}$ 上升时间, XCLKOUT		2	5	ns
<b>C5</b>	$t_{w(XCO_L)}$ 脉冲持续时间, XCLKOUT 低电平的时间	H-2		H+2	ns
<b>C6</b>	$t_{w(XCO_H)}$ 脉冲持续时间, XCLKOUT 高电平的时间	H-2		H+2	ns
	$t_p$ PLL 锁定时间	131072 $t_{c(OSCCLK)}$ <sup>(3)</sup>			周期

(1) 假定这些参数有一个40pF 的负载。

(2)  $H=0.5t_{c(XCO)}$

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



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

B. XCLKOUT 被配置成反映SYSCLKOUT。

**图 6-5 时钟时序**

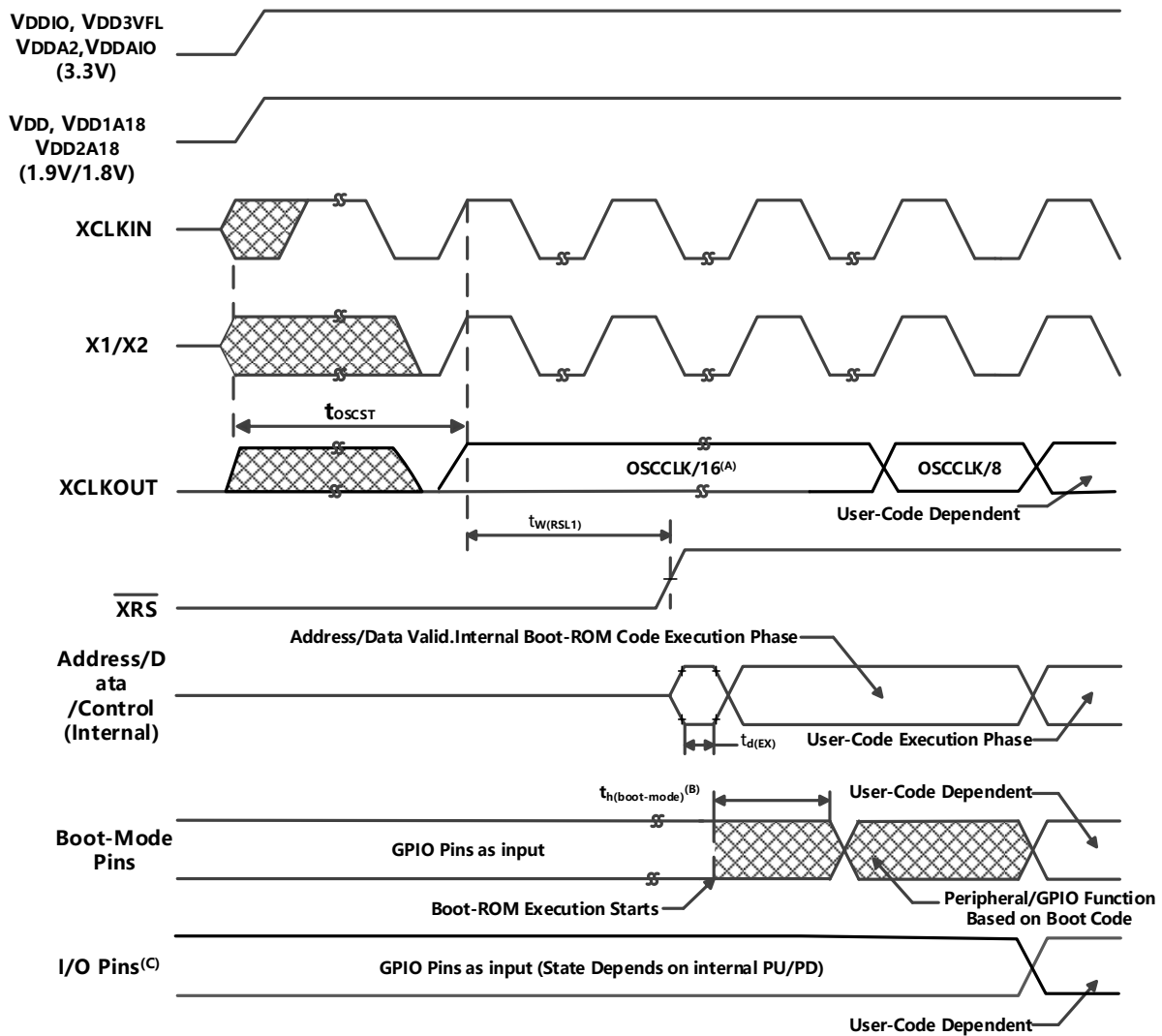
## 6.8 电源排序

对于不同电源引脚的加电/断电序列无特别要求以确保针对所有模块的正确复位。然而，如果 I/O 引脚的电平转移输出缓冲器中的 3.3V 晶体管在 1.9V 晶体管之前加电，输出缓冲器有可能打开，这会在加电期间导致引脚上的毛刺脉冲。为了避免这一运行状态，给  $V_{DD}$  引脚加电应早于对  $V_{DDIO}$  引脚供电，或者与之同时，以确保  $V_{DD}$  引脚在  $V_{DDIO}$  引脚达到 0.7V 之前达到 0.7V。

有一些对于  $\overline{XRS}$  引脚的要求：

1. 加电期间， $\overline{XRS}$  引脚必须在输入时钟稳定之后的  $t_{w(RSL1)}$  内保持低电平（请见表 6-11）。这使得整个器件从一个已知的条件启动。
2. 断电期间， $\overline{XRS}$  引脚必须至少在  $V_{DD}$  达到 1.5V 之前的  $8\mu s$  内被下拉至低电平。这样做提高了闪存可靠性。

在为器件加电之前，不应将  $V_{DDIO}$  之上大于二极管压降 (0.7V) 的电压应用于任何数字引脚上（对于模拟引脚，这个值是比  $V_{DDA}$  高 0.7V 的电压值）。此外， $V_{DDIO}$  和  $V_{DDA}$  之间的差距应一直在 0.3V 之内。施加在未加电器件的引脚上的电压会以一种无意的的方式偏置内部 p-n 结并产生无法预料的结果。



- A. 加电时，SYSCLKOUT为OSCCLK/4。由于XINTCNF2寄存器内的XTIMCLK和CLKMODE位出现时的状态为复位状态1，SYSCLKOUT在它出现在XCLKOUT上之前被进一步4频。这就是在这个阶段XCLKOUT=OSCCLK/16的原因。随后，引导ROM要把SYSCLKOUT改为OSCCLK/2。因为XTIMCLK寄存器不能被引导ROM改变，所以在此阶段，XCLKOUT为OSCCLK/8。
- B. 复位后，引导ROM代码采样引导模式引脚。基于引导模式引脚的状态，引导代码向目的内存或者引导代码函数下达分支指令。如果引导ROM代码在加电条件后（在调试器环境中）执行代码，引导代码执行时间由当前的SYSCLKOUT的速度而定。SYSCLKOUT将基于用户环境并可在PLL启用或者不启用时使用。
- C. 对于加电期间，确保一个GPIO引脚为高阻抗状态的要求，请见[节6.8](#)。

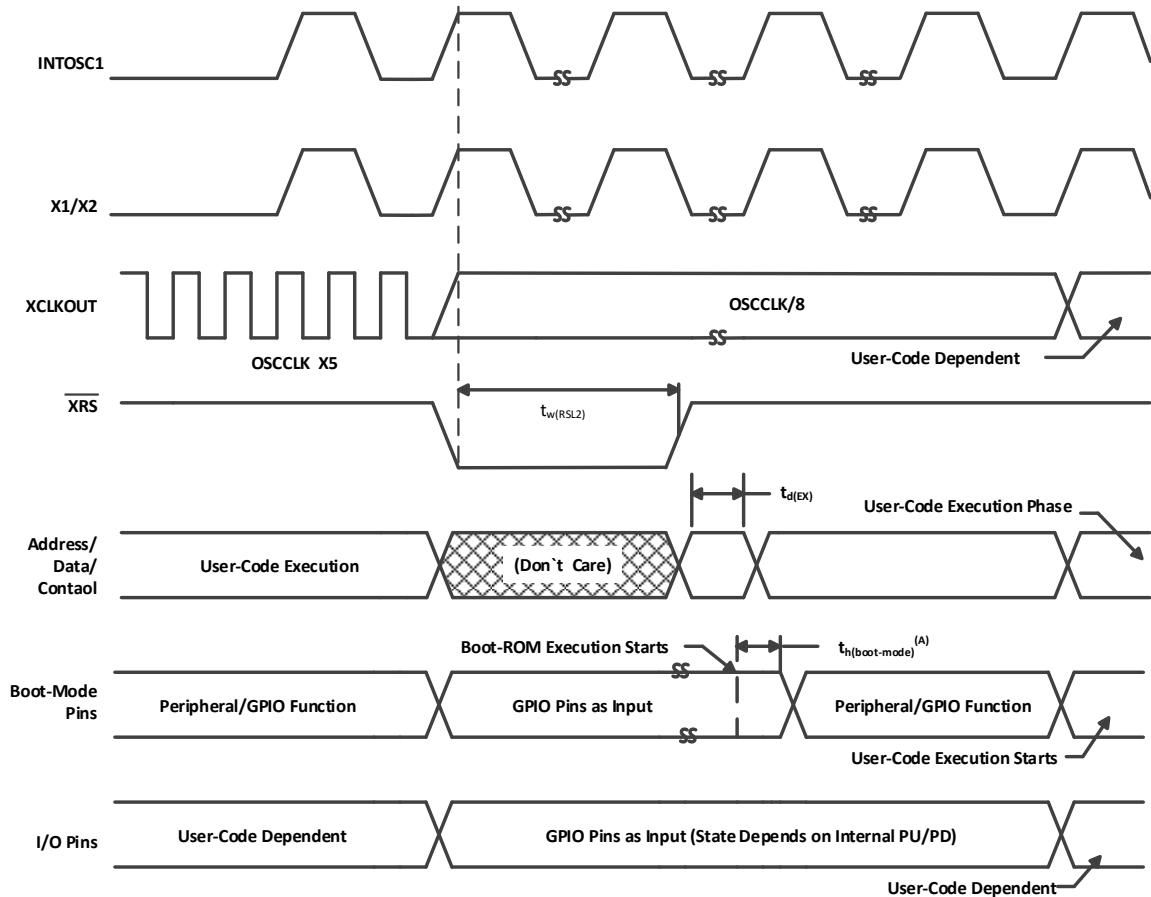
图 6-6 加电复位



表 6-9 复位XRS时序要求

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

- (1) 另外,  $t_{w(RSL1)}$ 要求,  $\overline{XRS}$ 必须在 VDD 达到 1.5V 后的 10ms 内为低电平。  
 (2) 取决于晶振/谐振器和电路板设计。



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

图 6-7 热复位

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

率, OSCCLK x 4。

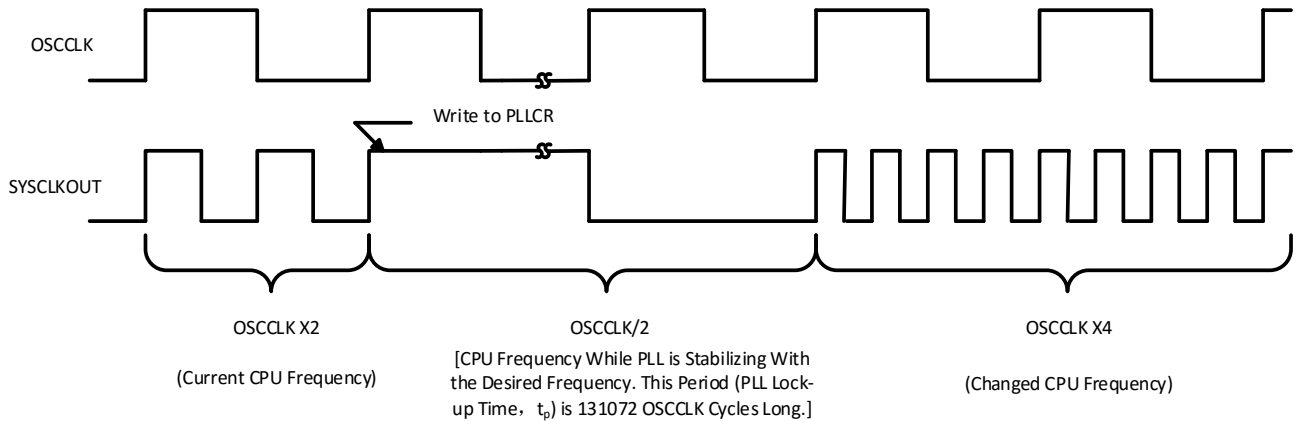


图 6-8 写入 PLLCR 寄存器所产生的效果的示例

## 6.9 通用输入/输出 (GPIO)

### 6.9.1 GPIO - 输出时序

表 6-10 通用输出开关特性

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

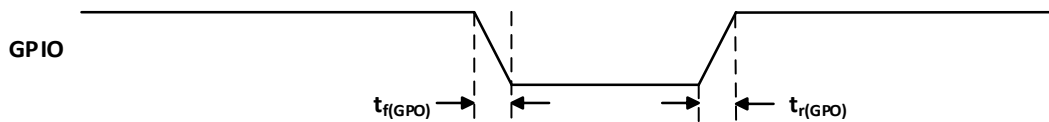


图 6-9 通用输出时序

### 6.9.2 GPIO - 输入时序

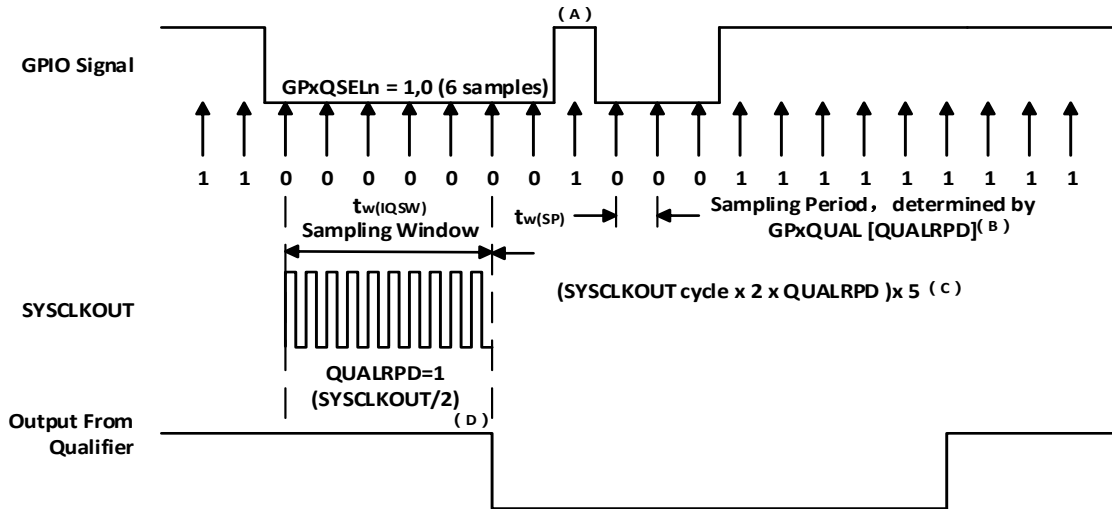
表 6-11 通用输入时序要求

		最小值	最大值	单位
$t_w(SP)$	采样周期	QUALPRD=0	$1t_{c(SCO)}$	周期
		QUALPRD≠0	$2t_{c(SCO)}*QUALPRD$	周期
$t_w(IQSW)$	输入限定器采样窗口		$t_w(SP)*(n^{(1)}-1)$	周期

$t_{w(GPI)}$ (2) 脉冲持续时间, GPIO 低电平/高电平的时间	同步模式	$2t_{c(SCO)}$	周期
	带有输入限定器	$t_w(IQSW) + t_w(SP) + 1t_{c(SCO)}$	周期

(1) "n" 代表由 GPxQSELn 寄存器定义的限定采样的数量。

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



- A. 这个毛刺脉冲将被输入限定器所忽略。QUALRPD位字段指定了限定采样周期。它可在0x00至0xFF间变化。如果QUALRPD=00, 那么采样周期为1个SYSCLKOUT周期。对于任何其它的"n"值, 限定采样周期为2nSYSCLKOUT周期 (也就是说, 在每一个SYSCLKOUT周期上, GPIO引脚将被采样)。
- B. 通过GPxCTRL寄存器选择的限定周期应用于一组8个GPIO引脚上。
- C. 此限定块可采样3个或者6个样本。GPxQSELn寄存器选择使用的采样模式。
- D. 在所示的示例中, 为了使限定器检测到变化, 输入应该在10个SYSCLKOUT周期或者更长的时间内保持稳定。换句话说, 输入应该在  $(5 \times \text{QUALRPD} \times 2) \times \text{SYSCLKOUT}$  周期内保持稳定。这将确保发生5个用于检测的采样周期。由于外部时钟被异步驱动, 一个13SYSCLKOUT宽的脉冲将确保可靠识别。

图 6-10 采样模式

### 6.9.3 针对输入信号的采样窗口宽度

下面的部分总结了不同的输入限定器配置下用于输入信号的采样窗口宽度。

采样频率表明相对于 SYSCLKOUT 的信号采样频率。

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

如果 QUALPRD=0 的话，采样频率 = SYSCLKOUT

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

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

如果 QUALPRD=0 的话，采样周期 = SYSCLKOUT 周期

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

#### 情况 1：

使用 3 个样本的限定

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

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

#### 情况 2：

使用 6 个样本的限定

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

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

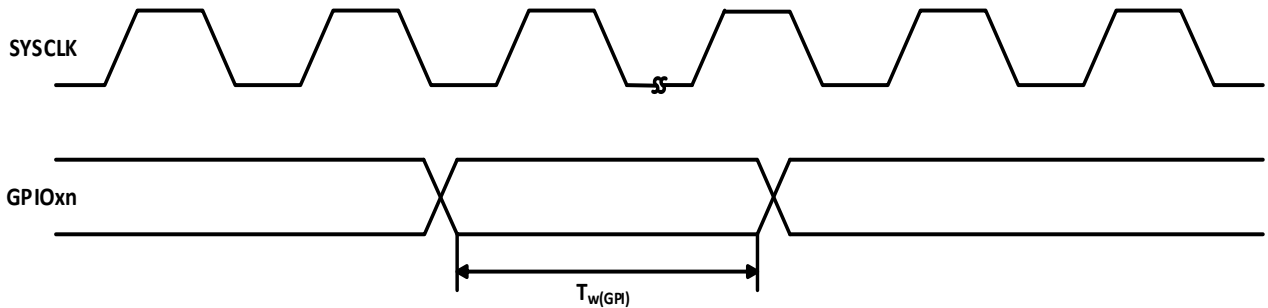


图 6-11 通用输入定时

### 6.9.4 低功耗唤醒定时

表 6-12 显示时序要求，表 6-13 显示了开关特性，而图 6-12 显示了 IDLE 模式下的时序图

表 6-12 IDLE 模式时序要求<sup>(1)</sup>

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

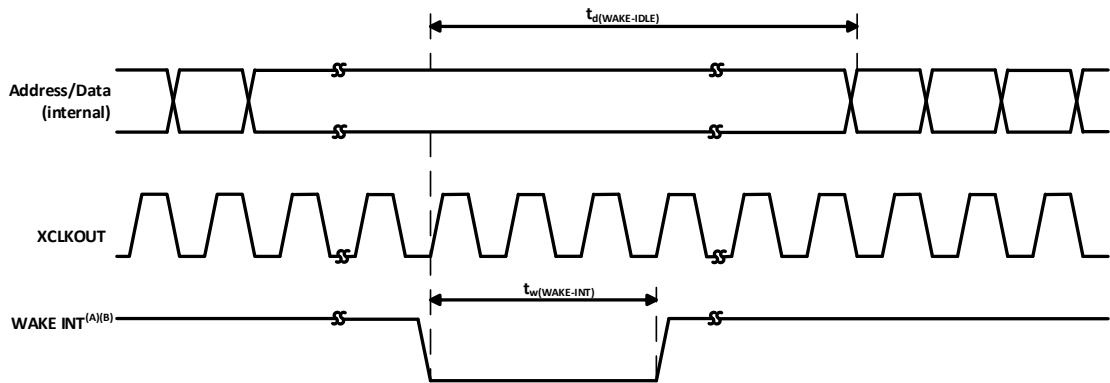
(1) 对于输入限定器参数的说明，请见表 6-11。

表 6-13 IDLE 模式开关特性<sup>(1)</sup>

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

(1) 对于输入限定器器参数的说明，请见表 6-11。

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



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

图 6-12 IDLE 进入和退出时序

**表 6-14 STANDBY 模式时序要求**

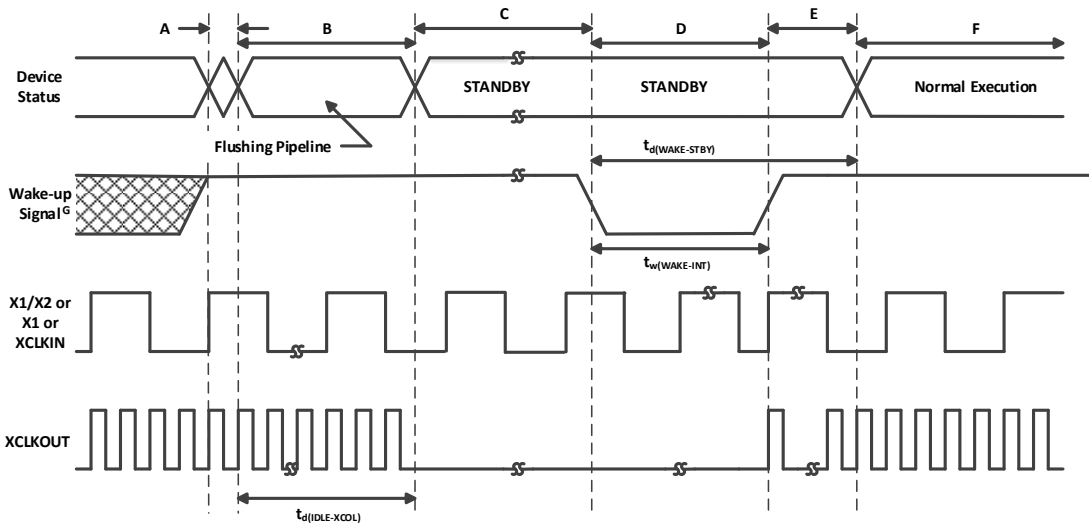
		最小值	标称值	最大值	单位
$t_w(\text{WAKE-INT})$ 脉冲持续时间, 外部唤醒信号的时间	无输入限定器	$3t_{c(\text{OSCCLK})}$			周期
	带有输入限定器 <sup>(1)</sup>	$(2 + \text{QUALSTDBY}) * t_{c(\text{OSCCLK})}$			

(1) QUALSTDBY 是一个 LPMCR0 寄存器内的 6 位字段。

**表 6-15 STANDBY 模式开关特性**

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

(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 (由唤醒触发) 信号的执行会涉及额外的延迟。



- A. 被执行的IDLE 指令将器件置于STANDBY 模式。
- B. PLL 块响应STANDBY 信号。在被关闭前，SYSCLOCKOUT 在下面标明的一定数量的周期内被保持：
  - 当DIVSEL=00 或11时，16 个周期
  - 当DIVSEL=10 时，32 个周期
  - 当DIVSEL=11 时，64 个周期

这个延迟使得CPU 管线和其它等待的操作被适当清空。到外设的时钟被关闭。如果一个到 XINTF 的访问正在进行中并且它的访问时间大于这个值，那么这个访问将发生故障。建议在没有一个 XINTF 访问进行时从 SARAM 进入 STANDBY 模式。
- C. 到外设的时钟被关闭。PLL 和看门狗并未关闭。此器件现在处于STANDBY 模式。
- D. 外部唤醒信号被驱动为有效。
- E. 在一个延迟周期内，退出STANDBY 模式。
- F. 正常执行重新开始。此器件将响应中断（如果被启用的话）。
- G. 从将器件置于低功耗模式 (LPM) 的 IDLE 指令被执行开始，在至少 4 个 OSCCLK 周期之前，唤醒不应被启动。

图 6-13 STANDBY 进入和退出时序

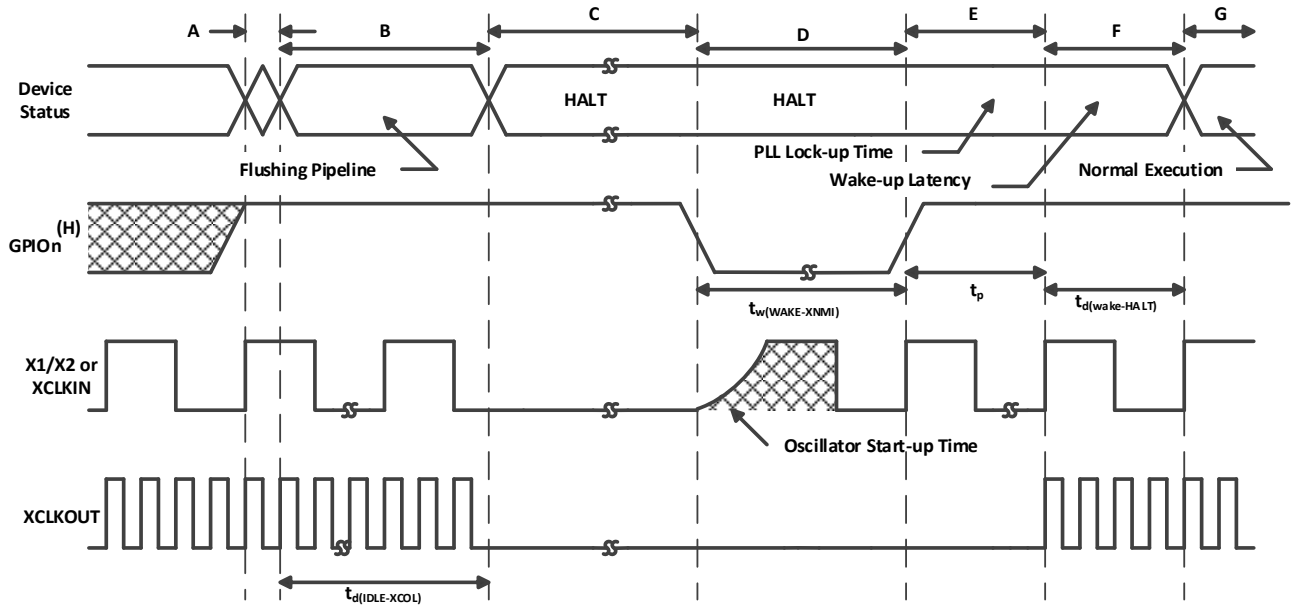
表 6-16 HALT 模式时序要求

	最小值	标称值	最大值	单位
$t_{w(WAKE-GPIO)}$ 脉冲持续时间，GPIO 唤醒信号的时间	$t_{oscst} + 2t_c(OSCCLK)^{(1)}$			周期
$t_{w(WAKE-XRS)}$ 脉冲持续时间，XRS唤醒信号的时间	$t_{oscst} + 8t_c(OSCCLK)$			周期

(1) oscst 的解释请见表 6-9

表 6-17 HALT 模式开关特性

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



- A. IDLE 指令被执行以将器件置于 HALT 模式。
- B. PLL 块响应 HALT 信号。在振荡器被关闭并且到内核的 CLKIN 被停止前 SYSCLKOUT 在下面所示的一定数量的周期内保持：
- 当 DIVSEL=00 或 11 时，16 周期
  - 当 DIVSEL=10 时，32 个周期
  - 当 DIVSEL=11 时，64 个周期
- 这个延迟使得 CPU 流水线和它等待的操作被适当清空。如果一个到 XINTF 的访问正在进行中并且它的访问时间大于这个值，那么这个访问将发生故障。建议在没有一个 XINTF 访问进行时从 SARAM 进入 HALT 模式
- C. 到外设的时钟被关闭并且 PLL 被关断。如果一个石英晶振或者陶瓷谐振器被用作时钟源，内部振荡器也被关断。器件现在处于 HALT 模式，消耗绝对最小功率。
- D. 当 GPIO<sub>n</sub> 引脚（用于使器件脱离 HALT 模式）被驱动为低电平时，振荡器被打开并且振荡器唤醒序列被启动。只有当振荡器稳定时，GPIO 才应被驱动为高电平。这样可在 PLL 锁序期间提供一个干净的时钟信号。由于 GPIO 引脚的下降边沿异步开始唤醒序列，请注意在进入和处于 HALT 模式期间保持一个低噪声环境。
- E. 一旦振荡器已经稳定，PLL 锁序列被启动（耗时 1ms），这将花费 131072 个 OSCCLK (X1/X2 或者 X1 或者 XCLKIN) 周期。请注意，即使当 PLL 被禁用（也就是说，即使当 PLL 被禁用时，代码执行也将被这个持续时间推迟），131072 个时钟周期也适用。
- F. 当到内核的时钟和外设被启用时，现在退出 HALT 模式。在一个延迟后，此器件响应此中断（如果被启用）。
- G. 正常运行重新开始。
- H. 从将器件置于低功耗模式 (LPM) 的 IDLE 指令被执行开始，在至少 4 个 OSCCLK 周期之前，唤醒不应被启动。

图 6-14 使用 GPIO<sub>n</sub> 的 HALT 唤醒

## 6.10 增强型控制外设

### 6.10.1 增强型脉宽调制器 (ePWM) 时序

PWM 是指 ePWM1-6 上的 PWM 输出。[表 6-18](#) 显示了 PWM 时序要求，[表 6-19](#) 显示了其开关特性。



表 6-18 PWM 时序要求<sup>(1)</sup>

参数		测试条件	最小值	最大值	单位
t <sub>w</sub> (SYCIN)	同步输入脉冲宽度	异步	2t <sub>c</sub> (SCO)		周期
		同步	2t <sub>c</sub> (SCO)		周期
		带有输入限定器	1t <sub>c</sub> (SCO)+t <sub>w</sub> (IQSW)		周期

(1) 要获得输入限定符参数的解释说明, 请见表 6-11

表 6-19 PWM 开关特性

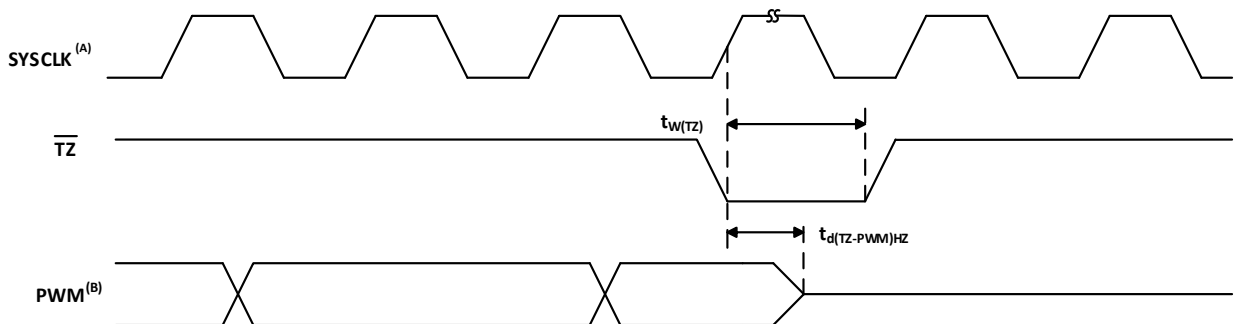
参数		测试条件	最小值	最大值	单位
t <sub>w</sub> (PWM)	脉冲持续时间, PWMx 输出高电平/低电平的时间		33.33		ns
t <sub>w</sub> (SYNCOUT)	同步输出脉冲宽度		8t <sub>c</sub> (SCO)		周期
t <sub>d</sub> (PWM)tza	延迟时间, 触发输入有效到 PWM 强制高电平的时间; 延迟时间, 触发输入有效到 PWM 强制低电平的时间;	无引脚负载		25	ns
t <sub>d</sub> (TZ-PWM)HZ	延迟时间, 触发输入有效至 PWM 高阻抗 (Hi-Z) 的时间			20	ns

### 6.10.2 可编程控制故障区输入时序

表 6-20 可编程控制故障区输入时序要求<sup>(1)</sup>

		测试条件	最小值	最大值	单位
t <sub>w</sub> (TZ) 脉冲持续时间, $\overline{TZ}$ 输入低电平的时间	异步		2t <sub>c</sub> (TBCLK)		周期
	同步		2t <sub>c</sub> (TBCLK)		周期
	带有输入限定器		2t <sub>c</sub> (TBCLK)+t <sub>w</sub> (IQSW)		周期

(1) 要获得输入限定符参数的解释说明, 请见表 6-11.



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

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

图 6-15 PWM Hi-Z 特性

### 6.10.3 高分辨率 PWM (HRPWM) 时序

表 6-21 显示了高分辨率 PWM 的开关特性。

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

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

(1) 最大 MEP 步长基于最差情况、最高温度。MEP 步长将随着电压的升高而增大，随着电压的降低而减小。

使用 HRPWM 特性的应用应该使用 MEP 缩放因子优化器 (SFO) 近似软件函数。SFO 函数有助于在 HRPWM 运行时动态地估计每个 SYSCLKOUT 周期内的 MEP 步数量。

### 6.10.4 增强型捕捉 (eCAP) 时序

[表 6-22](#)显示了 eCAP 时序要求，而[表 6-23](#)显示了 eCAP 开关特性。

**表 6-22 增强型捕捉 (eCAP) 时序要求<sup>(1)</sup>**

参数		测试条件	最小值	最大值	单位
t <sub>w</sub> (CAP)	捕捉输入脉冲宽度	异步	2t <sub>c(SCO)</sub>		周期
		同步	2t <sub>c(SCO)</sub>		周期
		带有输入限定器	1t <sub>c(SCO)</sub> +t <sub>w</sub> (IQSW)		周期

(1) 对于输入限定器参数的说明，请见[表 6-11](#)。

**表 6-23 eCAP 开关特性**

参数		测试条件	最小值	最大值	单位
t <sub>w</sub> (APWM)	脉冲持续时间，APWMx 输出高电平/低电平的时间		20		ns

### 6.10.5 增强型正交编码脉冲 (eQEP) 时序

[表 6-24](#)显示了 eQEP 时序要求，而[表 6-25](#)显示了 eQEP 开关特性。

**表 6-24 增强型正交编码脉冲 (eQEP) 时序要求<sup>(1)</sup>**

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

(1) 对于输入限定器参数的说明，请见[表 6-11](#)。

表 6-25 eQEP 开关特性

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

### 6.10.6 ADC 转换开始时序

表 6-26 外部 ADC 转换开始开关特性

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

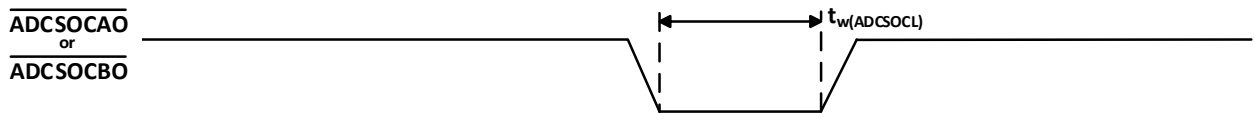


图 6-16 ADCSOCAO和ADCSOCBO时序图

### 6.11 外部中断时序

表 6-27 外部中断时序要求<sup>(1)</sup>

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

- (1) 对于输入限定器参数的说明, 请见表 6-11。  
 (2) 这个时序适用于为 ADCSOC 功能性所配置的任一 GPIO 引脚

表 6-28 外部中断开关特性<sup>(1)</sup>

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

- (1) 对于输入限定器参数的说明, 请见表 6-11。

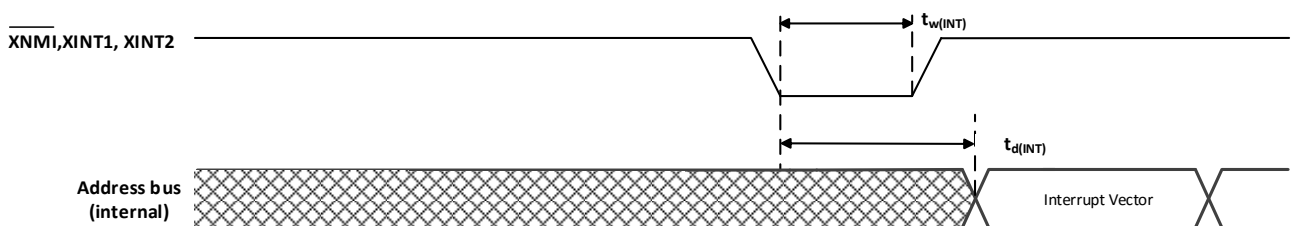


图 6-17 外部中断时序

## 6.12 I<sup>2</sup>C 电气特性和时序

表 6-29 I<sup>2</sup>C 时序

	测试条件	最小值	最大值	单位
f <sub>SCL</sub> SCL 时钟频率	I2C 时钟模块频率介于 7MHz 和 12MHz 之间并且 I2C 预分频器和时钟分频器寄存器被适当配置		400	kHz
V <sub>il</sub> 低电平输入电压			0.3V <sub>DDIO</sub>	V
V <sub>ih</sub> 高电平输入电压		0.7V <sub>DDIO</sub>		V
V <sub>hys</sub> 输入滞后		0.05V <sub>DDIO</sub>		V
V <sub>ol</sub> 低电平输出电流	3mA 吸收电流	0	0.4	V
t <sub>LOW</sub> SCL 时钟的低周期	I2C 时钟模块频率介于 7MHz 和 12MHz 之间并且 I2C 预分频器和时钟分频器寄存器被适当配置	1.3		μs
t <sub>HIGH</sub> SCL 时钟的高周期	I2C 时钟模块频率介于 7MHz 和 12MHz 之间并且 I2C 预分频器和时钟分频器寄存器被适当配置	0.6		μs
I <sub>i</sub> 输入电压介于 0.1V <sub>DDIO</sub> 和 0.9V <sub>DDIO</sub> (最大值) 的输入电流		-10	10	μA

## 6.13 同步串行通信接口(SPI) 主控模式时序

本节包含两个主模式和从模式时序数据

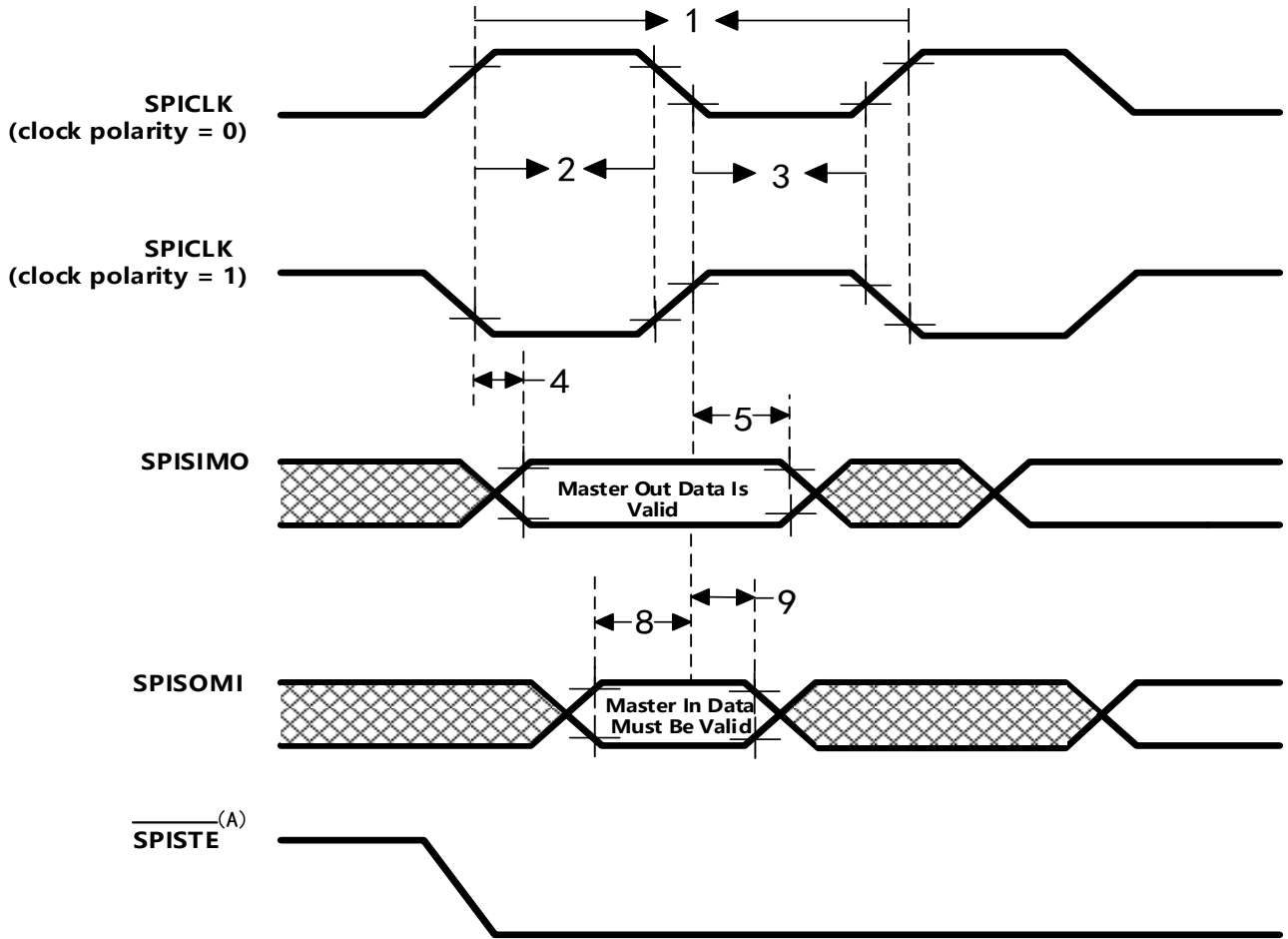
### 6.13.1 主模式时序

表6-30列出了主控模式时序（时钟相位= 0）而表6-31列出了时序（时钟相位=1）。图6-18和图6-19显示了时序波形。

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

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

- (1) 主控/受控位(SPICTL.2) 被设定, 而时钟相位的位(SPICTL.3) 被清除。
- (2)  $t_{c(SPC)} = \text{SPI 时钟周期时间} = \text{LSPCLK}/4$  或者  $\text{LSPCLK}/(\text{SPIBRR} + 1)$
- (3)  $t_{c(LCO)} = \text{LSPCLK 周期时间}$
- (4) 内部时钟预分频器必须被调整, 这样的话, SPI 时钟速度被限制在下列 SPI 时钟速率上:  
 主控模式发射最大值 25MHz, 主控模式接收最大值 12.5MHz  
 受控模式发送最大值 12.5MHz, 受控模式接收最大值 12.5MHz。
- (5) 作为基准的 SPICLK 信号的有效边沿由 CLOCK POLARITY ( 时钟极性 ) 位(SPICCR 6) 控制。



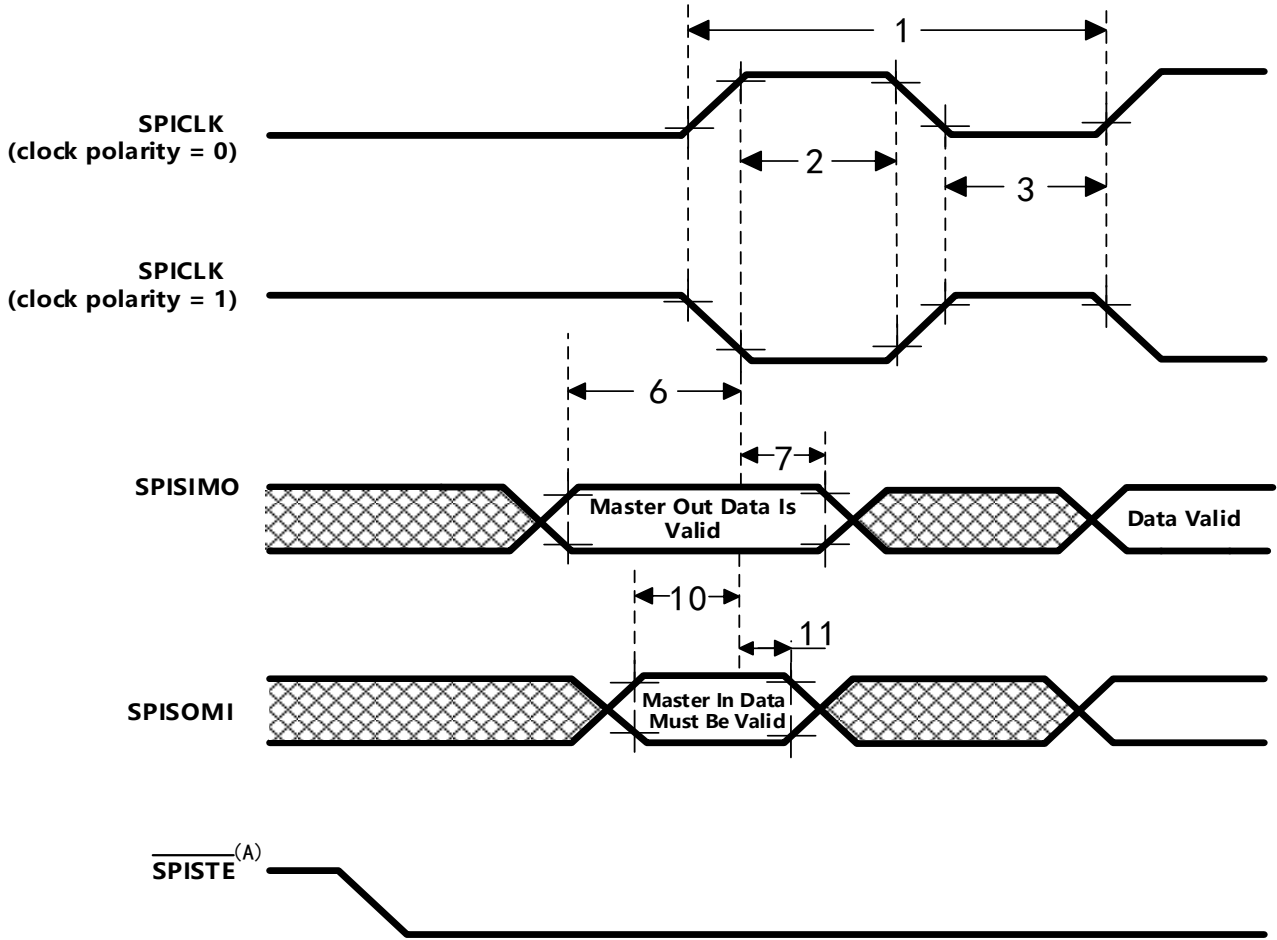
- A. 在主机模式下，在有效的SPI 时钟边沿之前 $0.5t_{c(SPC)}$ （最小值）， $\overline{SPISTE}$ 变为有效。在字的尾端， $\overline{SPISTE}$ 在接收到最后一个数据位的边沿(SPICLK) 之后 $0.5t_{c(SPC)}$ 将变为无效，除非 $\overline{SPISTE}$ 在FIFO 和非FIFO 模式中的背靠背传送字间保持有效。

图 6-18 SPI 主机模式外部时序 ( 时钟相位 = 0 )

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

编号			当(SPIBRR+1) 为偶数或者 SPIBRR=0 或者2 时的SPI		当(SPIBRR+1) 为奇数并且 SPIBRR>3 时的 SPI		单位
			最小值	最大值	最小值	最大值	
1	$t_{c(SPC)M}$	周期时间, SPICLK	$4t_{c(LCO)}$	$128t_{c(LCO)}$	$5t_{c(LCO)}$	$127t_{c(LCO)}$	ns
2	$t_{w(SPCH)M}$	脉冲持续时间, SPICLK 高电平的时间 ( 时钟极性= 0 )	$0.5t_{c(SPC)M}-10$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}-0.5t_{c(LCO)-10}$	$0.5t_{c(SPC)M}-0.5t_{c(LCO)}$	ns
	$t_{w(SPCL)M}$	脉冲持续时间, SPICLK 低电平的时间 ( 时钟极性= 1 )	$0.5t_{c(SPC)M}-10$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}-0.5t_{c(LCO)-10}$	$0.5t_{c(SPC)M}-0.5t_{c(LCO)}$	
3	$t_{w(SPCL)M}$	脉冲持续时间, SPICLK 低电平的时间 ( 时钟极性= 0 )	$0.5t_{c(SPC)M}-10$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}+0.5t_{c(LCO)-10}$	$0.5t_{c(SPC)M}+0.5t_{c(LCO)}$	ns
	$t_{w(SPCH)M}$	脉冲持续时间, SPICLK 高电平的时间 ( 时钟极性= 1 )	$0.5t_{c(SPC)M}-10$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}+0.5t_{c(LCO)-10}$	$0.5t_{c(SPC)M}+0.5t_{c(LCO)}$	
6	$t_{su(SIMO-SPCH)M}$	建立时间, 在 SPICLK 高电平之前 SPISIMO 数据有效的的时间 ( 时钟极性= 0 )	$0.5t_{c(SPC)M}-10$		$0.5t_{c(SPC)M}-10$		ns
	$t_{su(SIMO-SPCL)M}$	建立时间, 在 SPICLK 低电平之前 SPISIMO 数据有效的的时间 ( 时钟极性= 1 )	$0.5t_{c(SPC)M}-10$		$0.5t_{c(SPC)M}-10$		
7	$t_{v(SPCH-SIMO)M}$	有效时间, SPICLK 高电平之后 SPISIMO 数据有效的的时间 ( 时钟极性= 0 )	$0.5t_{c(SPC)M}-10$		$0.5t_{c(SPC)M}-10$		ns
	$t_{v(SPCL-SIMO)M}$	有效时间, SPICLK 低电平后, SPISIMO 数据有效的的时间 ( 时钟极性= 1 )	$0.5t_{c(SPC)M}-10$		$0.5t_{c(SPC)M}-10$		
10	$t_{su(SOMI-SPCH)M}$	建立时间, SPISOMI 在 SPICLK 高电平之前的时间 ( 时钟极性= 0 )	35		35		ns
	$t_{su(SOMI-SPCL)M}$	建立时间, SPISOMI 在 SPICLK 低电平之前的时间 ( 时钟极性= 1 )	35		35		
11	$t_{v(SPCH-SOMI)M}$	有效时间, SPICLK 高电平之后 SPISOMI 数据有效的的时间 ( 时钟极性= 0 )	$0.25t_{c(SPC)M}-10$		$0.5t_{c(SPC)M}-10$		ns
	$t_{v(SPCL-SOMI)M}$	有效时间, SPICLK 低电平之后 SPISOMI 数据有效的的时间 ( 时钟极性= 1 )	$0.25t_{c(SPC)M}-10$		$0.5t_{c(SPC)M}-10$		

- (1) 主控/受控位(SPICTL.2) 被设定, 而时钟相位的位(SPICTL.3) 被清除。
- (2)  $t_{c(SPC)} = \text{SPI 时钟周期时间} = \text{LSPCLK}/4$  或者  $\text{LSPCLK}/(\text{SPIBRR} + 1)$
- (3) 内部时钟预分频器必须被调整, 这样的话, SPI 时钟速度被限制在下列 SPI 时钟速率上:  
主控模式发射最大值 25MHz, 主控模式接收最大值 12.5MHz  
受控模式发送最大值 12.5MHz, 受控模式接收最大值 12.5MHz。
- (4)  $t_{c(LCO)} = \text{LSPCLK 周期时间}$
- (5) 作为基准的 SPICLK 信号的有效边沿由 CLOCK POLARITY ( 时钟极性 ) 位(SPICCR 6) 控制。



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

图 6-19 SPI 主控模式外部时序 ( 时钟相位= 1 )



### 6.13.2 SPI 受控模式时序

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

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

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

- (1) 主控/受控位(SPICTL.2) 被设定, 而时钟相位的位(SPICTL.3) 被清除。
- (2)  $t_{c(SPC)} = \text{SPI 时钟周期时间} = \text{LSPCLK}/4$  或者  $\text{LSPCLK}/(\text{SPIBRR} + 1)$
- (3) 内部时钟预分频器必须被调整, 这样的话, SPI 时钟速度被限制在下列 SPI 时钟速率上:  
 主控模式发射最大值 25MHz, 主控模式接收最大值 12.5MHz  
 受控模式发送最大值 12.5MHz, 受控模式接收最大值 12.5MHz。
- (4)  $t_{c(LCO)} = \text{LSPCLK 周期时间}$
- (5) 作为基准的 SPICLK 信号的有效边沿由 CLOCK POLARITY (时钟极性) 位(SPICCR. 6) 控制。

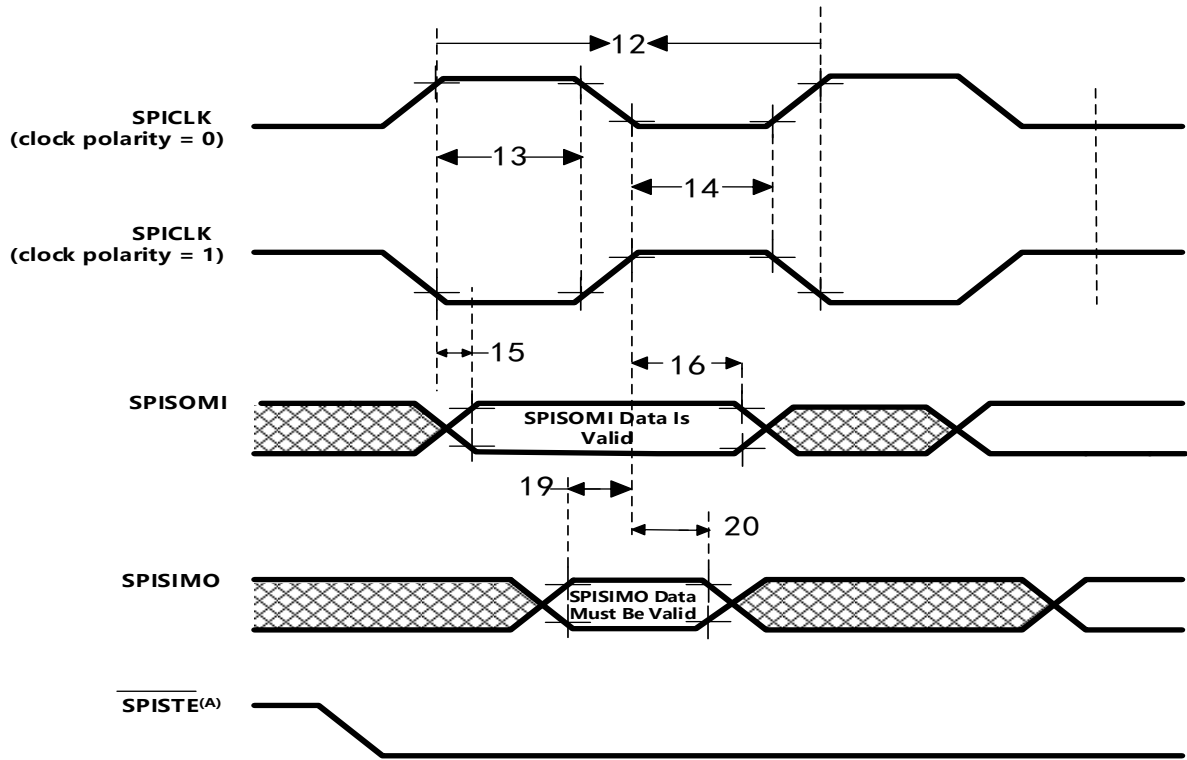


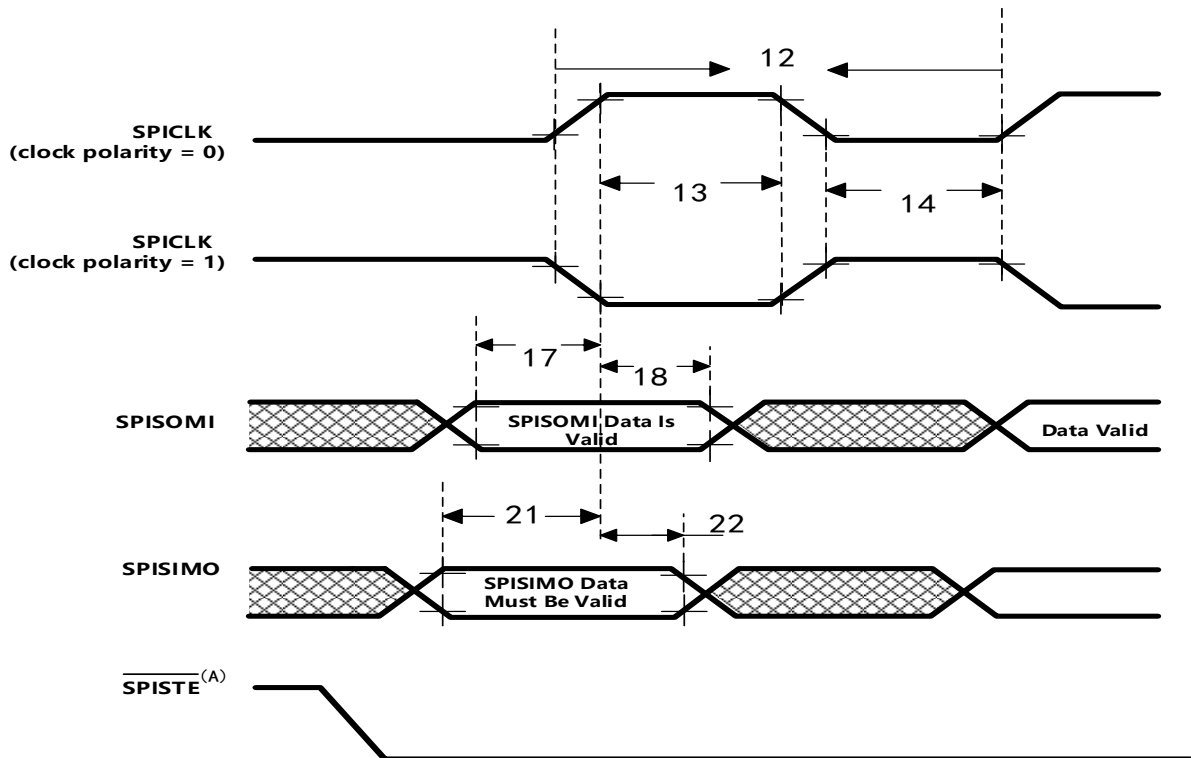
图 6-20 SPI 受控模式外部时序 ( 时钟相位 = 0 )

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

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

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

- (1) 主控/受控位(SPICTL.2) 被设定, 而时钟相位的位(SPICTL.3) 被清除。
- (2)  $t_{c(SPC)} = \text{SPI 时钟周期时间} = \text{LSPCLK}/4$  或者  $\text{LSPCLK}/(\text{SPIBRR} + 1)$
- (3) 内部时钟预分频器必须被调整, 这样的话, SPI 时钟速度被限制在下列 SPI 时钟速率上:  
 主控模式发射最大值 25MHz, 主控模式接收最大值 12.5MHz  
 受控模式发送最大值 12.5MHz, 受控模式接收最大值 12.5MHz。
- (4) 作为基准的 SPICLK 信号的有效边沿由 CLOCK POLARITY ( 时钟极性 ) 位(SPICCR. 6) 控制。



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

图 6-21 SPI 受控模式外部时序 ( 时钟相位= 1 )

## 6.14 外部接口 ( XINTF ) 时序

每个 XINTF 访问由三部分组成：建立、有效和跟踪。用户在 XTIMING 寄存器中配置建立/有效/跟踪等待状态。有一个 XTIMING 寄存器用于每个 XINTF 区域。表 6-34 显示了 XTIMING 寄存器中配置的参数和以 XTIMING 周期为单位的脉冲持续时间之间的关系。

**表 6-34 XTIMING 中配置的参数和脉冲持续时间之间的关系**

说明	持续时间 ( ns ) <sup>(1)(2)</sup>	
	X2TIMING=0	X2TIMING=1
<b>LR</b> 建立周期, 读取访问	$XRDLEAD \times t_{c(XTIM)}$	$(XRDLEAD \times 2) \times t_{c(XTIM)}$
<b>AR</b> 激活周期, 读取访问	$(XRDACTIVE + WS + 1) \times t_{c(XTIM)}$	$(XRDACTIVE \times 2 + WS + 1) \times t_{c(XTIM)}$
<b>TR</b> 跟踪周期, 读取访问	$XRDTRAIL \times t_{c(XTIM)}$	$(XRDTRAIL \times 2) \times t_{c(XTIM)}$
<b>LW</b> 建立周期, 写入访问	$XWRLEAD \times t_{c(XTIM)}$	$(XWRLEAD \times 2) \times t_{c(XTIM)}$
<b>AW</b> 激活周期, 写入访问	$(XWRACTIVE + WS + 1) \times t_{c(XTIM)}$	$(XWRACTIVE \times 2 + WS + 1) \times t_{c(XTIM)}$
<b>TW</b> 跟踪周期, 写入访问	$XWRTRAIL \times t_{c(XTIM)}$	$(XWRTRAIL \times 2) \times t_{c(XTIM)}$

(1)  $t_{c(XTIM)}$ -周期时间, XTIMCLK

(2) WS 是指当使用 XREADY 时, 由硬件插入的等待状态的数量。如果此区域被配置成忽略 XREADY (USEREADY=0), 那么 WS=0。

当配置每个区域的 XTIMING 寄存器时, 必须满足最小等待状态要求。这些要求是器件数据表中指定的任一时序要求之外的要求。没有任何内部器件硬件来检测非法设置。

### 6.14.1 USEREADY = 0

如果 XREADY 信号被忽略 (USEREADY=0), 那么：

前置： $LR \geq t_{c(XTIM)}$

$LW \geq t_{c(XTIM)}$

这些要求导致了下列 XTIMING 寄存器的配置限制：

XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
$\geq 1$	$\geq 0$	$\geq 0$	$\geq 1$	$\geq 0$	$\geq 0$	0, 1

当不采样 XREADY 时的有效和无效示例：

	XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
无效 <sup>(1)</sup>	0	0	0	0	0	0	0,1
有效	1	0	0	1	0	0	0,1

(1) 没有硬件检测非 XTIMING 配置

### 6.14.2 异步模式(USEREADY=1, READYMODE=0)

如果 XREADY 信号在同步模式中被采样(USEREADY=1, READYMODE=1), 那么：

1 前置： $LR \geq t_{c(XTIM)}$

$LW \geq t_{c(XTIM)}$

2 有效： $AR \geq 2 \times t_{c(XTIM)}$

$AW \geq 2 \times t_{c(XTIM)}$

**注**

限制条件不包括外部硬件等待状态。

这些要求导致了下列 XTIMING 寄存器的配置限制：

XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
≥ 1	≥ 1	≥ 0	≥ 1	≥ 1	≥ 0	0, 1

使用同步 XREADY 时，有效和无效时序示例：

	XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
无效 <sup>(1)</sup>	0	0	0	0	0	0	0,1
无效 <sup>(1)</sup>	1	0	0	1	0	0	0,1
有效	1	1	0	1	1	0	0,1

(1) 没有硬件检测非 XTIMING 配置

### 6.14.3 同步模式(USEREADY=1, READYMODE=1)

如果 XREADY 信号在同步模式中被采样(USEREADY=1, READYMODE=0)，那么：

- 1 前置：  
 $LR \geq t_{c(XTIM)}$   
 $LW \geq t_{c(XTIM)}$
- 2 有效：  
 $AR \geq 2 \times t_{c(XTIM)}$   
 $AW \geq 2 \times t_{c(XTIM)}$
- 3 前置+有效：  
 $LR + AR \geq 4 \times t_{c(XTIM)}$   
 $LW + AW \geq 4 \times t_{c(XTIM)}$

**注**

限制条件不包括外部硬件等待状态。

这些要求导致了下列 XTIMING 寄存器的配置限制：

XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
≥ 1	≥ 2	0	≥ 1	≥ 2	0	0, 1

或者

XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
≥ 2	≥ 1	0	≥ 2	≥ 1	0	0, 1

使用异步 XREADY 时，有效和无效时序示例：

	XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
无效 <sup>(1)</sup>	0	0	0	0	0	0	0,1
无效 <sup>(1)</sup>	1	0	0	1	0	0	0,1

无效 <sup>(1)</sup>	1	1	0	1	1	0	0
有效	1	1	0	1	1	0	1
有效	1	2	0	1	2	0	0,1
有效	2	1	0	2	1	0	0,1

(1) 没有硬件检测非 XTIMING 配置

除非另外注明，否则所有 XINTF 时序适用于表 6-35 中显示的时钟配置。

表 6-35 XINTF 时钟配置

模式	SYSCALLKOUT	XTIMCLK	XCLKOUT
1		SYSCALLKOUT	SYSCALLKOUT
示例：	150MHz	150MHz	150MHz
2		SYSCALLKOUT	1/2 SYSCALLKOUT
示例：	150MHz	150MHz	75MHz
3		1/2SYSCALLKOUT	1/2 SYSCALLKOUT
示例：	150MHz	75MHz	75MHz
4		1/2SYSCALLKOUT	1/4 SYSCALLKOUT
示例：	150MHz	75MHz	37.5MHz

SYSCALLKOUT 和 XTIMCLK 之间的关系显示在图 6-22 中。

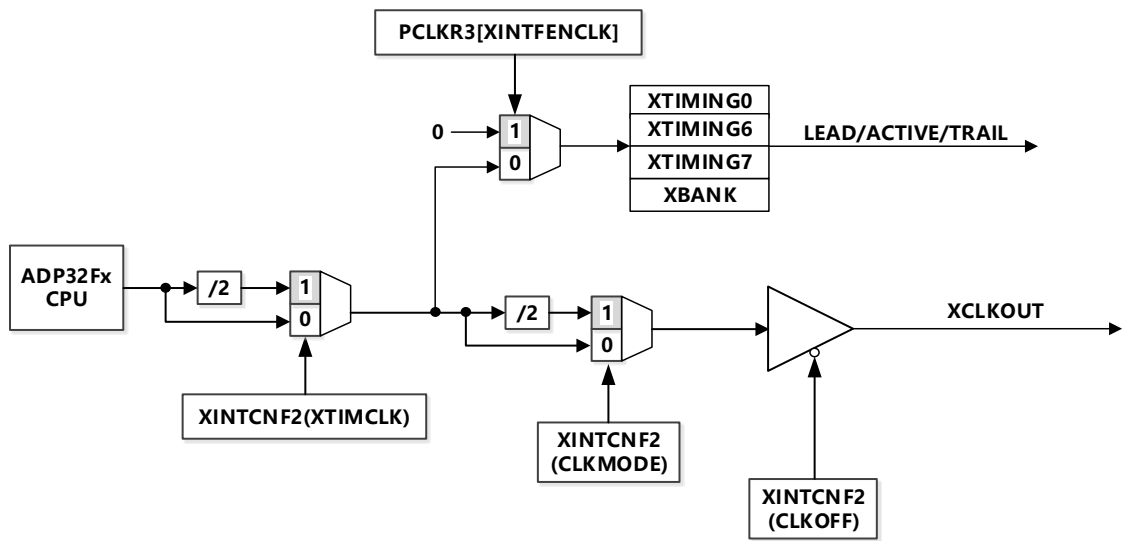


图 6-22 XTIMCLK 和 SYSCALLKOUT 之间关系

### 6.14.4 XINTF 信号与 XCLKOUT 一致

对于每个 XINTF 访问，前置、有效、后置周期的数量基于内部时钟 XTIMCLK。诸如  $\overline{XRDR}$ ， $\overline{XWE0}$ ， $\overline{XWE1}$  选通脉冲，和区域芯片选择 (XZCS) 状态改变与 XTIMCLK 的上升边沿有关。外部时钟 XCLKOUT，可被配置成等于 XTIMCLK 周期或者为 XTIMCLK 周期的一半。

对于 XCLKOUT=XTIMCLK 的情况，所有 XINTF 选通时钟将相对于 XCLKOUT 的上升边沿改变状态。对于 XCLKOUT 为 XTIMCLK 的一半的情况，一些选通脉冲将在 XCLKOUT 的上升边沿或者 XCLKOUT 的下降边沿上改变状态。在 XINTF 时序表中，符号 XCOHL 被用于表示相对于任何一种情况的参数；XCLKOUT 上升边沿（高电平）或者 XCLKOUT 下降边沿（低电平）。如果参数一直相对于 XCLKOUT 的上升边沿的话，符号 XCOH 被使用。

对于 XCLKOUT 为 XTIMCLK 的一半的情况，基于从访问开始到信号变化发生点的 XTIMCLK 周期的数量，可确定与变化对齐的 XCLKOUT 边沿。如果这个 XTIMCLK 周期的数量为偶数，对齐将相对于 XCLKOUT 的上升边沿。如果这个 XTIMCLK 周期的数量为奇数，那么信号将相对于 XCLKOUT 的下降边沿发生变化。示例包括如下：

- 在一个访问开始时发生变化的选通脉冲一直与 XCLKOUT 的上升边沿对齐。这是因为所有 XINTF 方位相对于 XCLKOUT 的上升边沿开始。

示例： XZCSL 区域选低电平  
 XRNWL XR/ $\overline{W}$ 低电平有效

- 如果用于访问的建立XTIMCLK周期为偶数，在一个有效周期开始时发生变化的选通脉冲将与 XCLKOUT 的上升边沿对齐。如果建立XTIMCLK周期的数量为偶数，那么对齐将相对于 XCLKOUT 的下降边沿。

示例： XRDL  $\overline{XRD}$ 低电平有效  
 XWEL  $\overline{XWE1}$ 或 $\overline{XWE0}$ 低电平有效

- 如果用于访问的建立和有效 XTIMCLK 周期总数（包括硬件等待状态）为偶数，在一个跟踪周期开始时发生变化的选通脉冲将与 XCLKOUT 的上升边沿对齐。如果建立和有效 XTIMCLK 周期的数量（包括硬件等待状态）为奇数，那么对齐将相对于 XCLKOUT 的下降边沿。

示例： XRDH  $\overline{XRD}$ 高电平无效  
 XWEH  $\overline{XWE1}$ 或 $\overline{XWE0}$ 高电平无效

- 如果建立和有效加上跟踪 XTIMCLK 周期总数（包括硬件等待状态）为偶数，在一个访问末尾发生变化的选通脉冲将与 XCLKOUT 的上升边沿对齐。如果建立和有效加上跟踪 XTIMCLK 周期的数量（包括硬件等待状态）为奇数，那么对齐将相对于 XCLKOUT 的下降边沿。

示例： XZCSH 区片选高电平无效  
 XRNWH XR/ $\overline{W}$ 高电平无效

### 6.14.5 外部接口读取时序

表 6-36 外部存储器接口读取时序要求

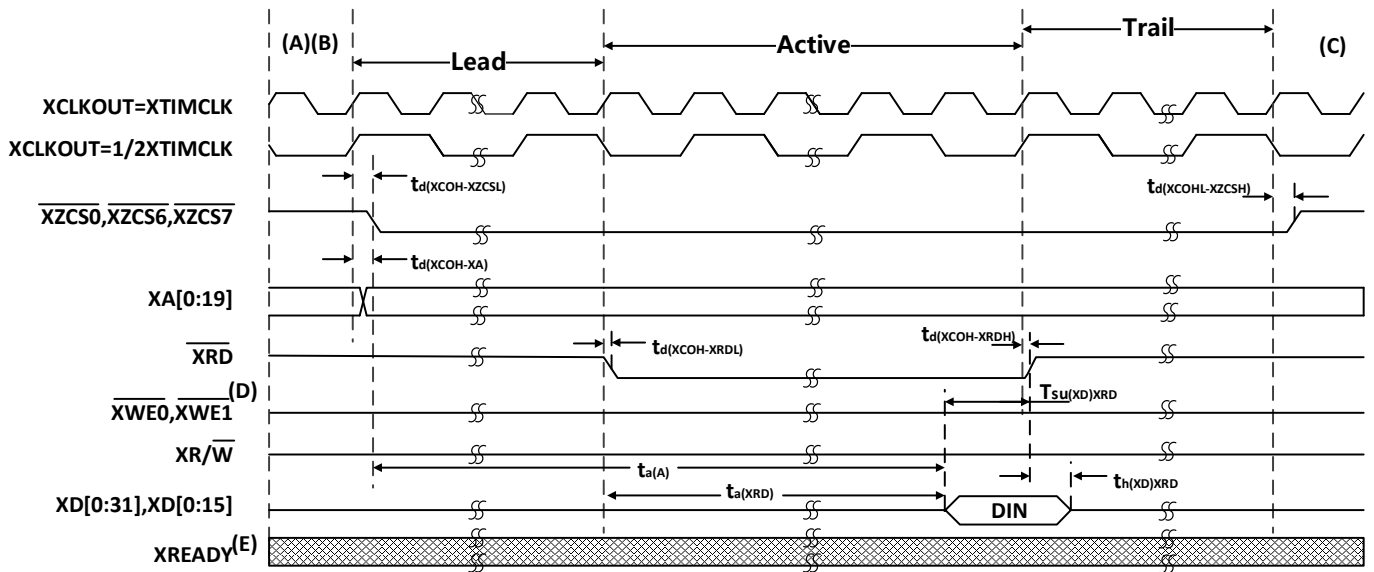
		最小值	最大值	单位
$t_{a(A)}$	访问时间，从有效地址读取数据的时间		$(LR + AR) - 16^{(1)}$	ns
$t_{a(XRD)}$	访问时间，从 $\overline{XRD}$ 低电平有效读取有效数据的时间		$AR - 14^{(1)}$	ns
$t_{su(XD)XRD}$	建立时间，在 $\overline{XRD}$ 选通脉冲高电平无效之前，读取有效数据的时间	14		ns
$t_{h(XD)XRD}$	保持时间， $\overline{XRD}$ 高电平无效之后读取数据有效的时间	0		ns

(1) LR = 建立周期，读取访问。AR = 有效周期，读取访问。请参考表6-34。

表 6-37 外部内存接口读取开关特性

参数	描述	最小值	最大值	单位
$t_{d(XCOH-XZCSL)}$	延迟时间，XCLKOUT 高电平到区域芯片选择低电平有效的的时间		0.5	ns
$t_{d(XCOHL-XZCSH)}$	延迟时间，XCLKOUT 高电平/低电平到芯片选择高电平无效的时间	-0.5	1	ns
$t_{d(XCOH-XA)}$	延迟时间，XCLKOUT 高电平到地址有效的的时间		1	ns
$t_{d(XCOHL-XRDL)}$	延迟时间，XCLKOUT 高电平/低电平到 $\overline{XRD}$ 低电平有效的的时间		0.5	ns
$t_{d(XCOHL-XRDH)}$	延迟时间，XCLKOUT 高电平/低电平到 $\overline{XRD}$ 高电平无效的时间	-1.8	0.5	ns
$t_{h(XA)XZCSH}$	保持时间，区域芯片选择高电平无效之后地址有效时间	(1)		ns
$t_{h(XA)XRD}$	保持时间， $\overline{XRD}$ 高电平无效后的地址有效时间	(1)		ns

(1) 在未激活周期期间，XINTF 地址总线将一直保持总线上产生的最后一个地址除 XA0 之外，它将一直保持高电平。这个包括对齐周期。



- A. 所有 XINTF 访问（前置周期）在 XCLKOUT 的上升沿上开始。当需要时，器件将在一个满足这个要求的访问之前插入一个对准周期。
- B. 在对准周期期间，所有信号将被转换为它们的未激活状态。
- C. XA[0:19] 在非活动周期保持总线上的最后一个地址，包括校准周期，除 XA0 之外，它一直保持高电平状态。
- D.  $\overline{XWE1}$ 用于32位数据总线模式。在16位模式中，该信号是XAO。
- E. 因为USEREADY=0，外部XREADY输入信号被忽略。

图 6-23 示例读取访问

在本例中使用XTIMING 寄存器参数：

XRDLEAD	XRDACTIVE	XRDTRAIL	USEREADY	X2TIMING	XWRLEAD	XWRACTIV	XWRTRAIL	READYMOD
---------	-----------	----------	----------	----------	---------	----------	----------	----------



≥ 1	≥ 0	≥ 0	0	0	N/A <sup>(1)</sup>	N/A <sup>(1)</sup>	N/A <sup>(1)</sup>	N/A <sup>(1)</sup>
-----	-----	-----	---	---	--------------------	--------------------	--------------------	--------------------

(1) 这个例子中 N/A = 不适用 ( 或 “无关” )

### 6.14.6 外部接口写入时序

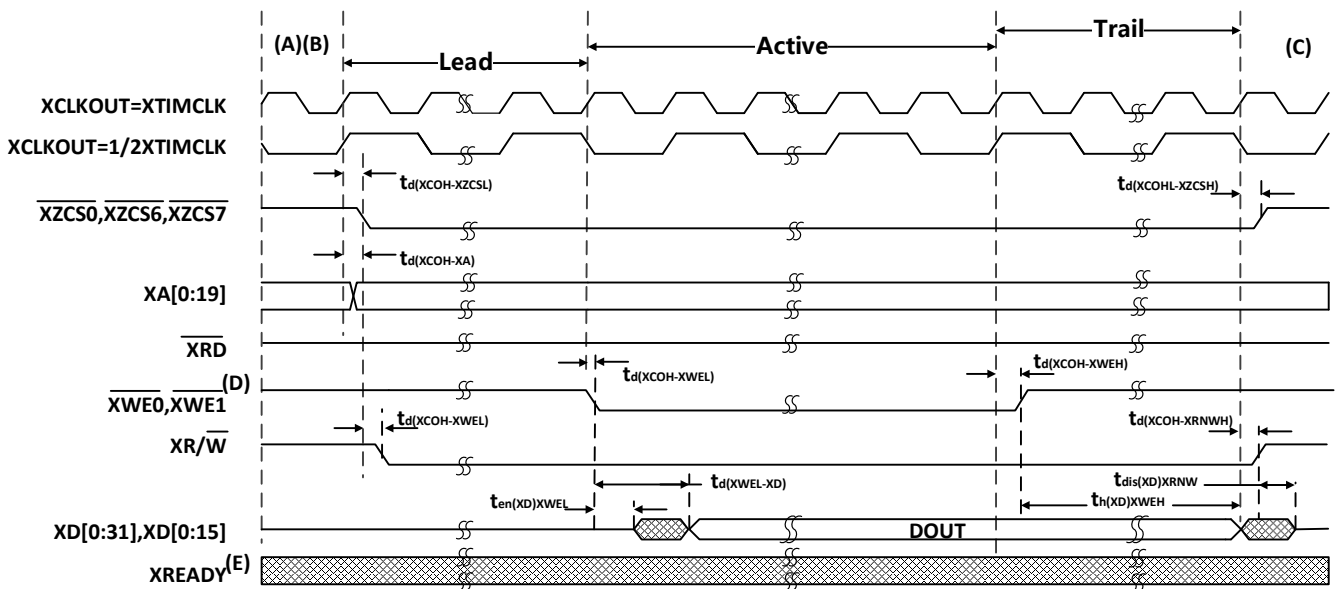
表 6-38 外部存储器接口写入开关特性

参数	描述	最小值	最大值	单位
$t_{d(XCOH-XZCSL)}$	延迟时间, XCLKOUT 高电平到区域芯片选择低电平有效的的时间		1	ns
$t_{d(XCOHL-XZCSH)}$	延迟时间, XCLKOUT 高电平/低电平到芯片选择高电平无效的时间	-0.5	1	ns
$t_{d(XCOH-XA)}$	延迟时间, XCLKOUT 高电平到地址有效的的时间		1	ns
$t_{d(XCOHL-XWEL)}$	延迟时间, XCLKOUT 高电平/低电平到 $\overline{XWE0}$ , $\overline{XWE1}$ <sup>(1)</sup> 低电平的时间		3	ns
$t_{d(XCOHL-XWEH)}$	延迟时间, XCLKOUT 高电平/低电平到 $\overline{XWE0}$ , $\overline{XWE1}$ 高电平的时间		3	ns
$t_{d(XCOH-XRNWL)}$	延迟时间, XCLKOUT 高电平到 XR/ $\overline{W}$ 低电平的时间		1	ns
$t_{d(XCOHL-XRNWH)}$	延迟时间, XCLKOUT 高电平/低电平到 XR/ $\overline{W}$ 高电平的时间	-1	1	ns
$t_{en(XD)XWEL}$	使能时间, 从 $\overline{XWE0}$ , $\overline{XWE1}$ 低电平驱动数据总线的的时间	0		ns
$t_{d(XWEL-XD)}$	延迟时间, $\overline{XWE0}$ , $\overline{XWE1}$ 低电平有效后的数据有效时间		1	ns
$t_{h(XA)XZCSH}$	保持时间, 区域芯片选择高电平无效之后地址有效时间	(2)		ns
$t_{h(XD)XWE}$	保持时间, $\overline{XWE0}$ , $\overline{XWE1}$ 高电平无效之后写入数据有效时间	TW-2 <sup>(3)</sup>		ns
$t_{dis(XD)XRNW}$	XR/ $\overline{W}$ 高电平无效之后 DSP 释放数据总线的的最长时间		4	ns

(1)  $\overline{XWE1}$ 只用于32 位数据总线模式。在16 位模式中, 该信号是 XAO。

(2) 在未激活周期期间, XINTF 地址总线将一直保持总线上产生的最后一个地址, 除 XAO 之外, 它将一直保持高电平。这包括对准周期。

(3) TW = 跟踪周期, 写入访问。请参考 表 6-34。



- A. 所有 XINTF 访问 (前置周期) 在 XCLKOUT 的上升沿上开始。当需要时, 器件将在一个满足这个要求的访问之前插入一个对准周期。
- B. 在对准周期期间, 所有信号将被转换为它们的未激活状态。
- C. XA[0:19] 在非活动周期保持总线上产生的最后一个地址, 包括校准周期, 除了 XAO 一直保持高电平状态。
- D.  $\overline{XWE1}$ 用于 32 位数据总线模式。在 16 位模式中, 该信号是 XAO。因为 USEREADY=0, 外部 XREADY 输入信号被忽略。

图 6-24 示例写入访问

这个例子中使用 XTIMING 寄存器参数：

XRDLEAD	XRDACTIVE	XRDTRAIL	USEREADY	X2TIMING	XWRLEAD	XWRACTIVE	XWRTRAIL	READYMODE
N/A <sup>(1)</sup>	N/A <sup>(1)</sup>	N/A <sup>(1)</sup>	0	0	≥ 1	≥ 0	≥ 0	N/A <sup>(1)</sup>

(1) 这个例子中 N/A=不适用 (或“无关”)

### 6.14.7 带有一个外部等待状态的外部接口读取准备就绪时序

**表 6-39 外部接口读取开关特性 ( ( 读取准备就绪, 1 个等待状态 ) )**

参数	描述	最小值	最大值	单位
t <sub>d</sub> (XCOH-XZCSL)	延迟时间, XCLKOUT 高电平到区域芯片选择低电平有效的的时间		0.5	ns
t <sub>d</sub> (XCOHL-XZCSH)	延迟时间, XCLKOUT 高电平/低电平到芯片选择高电平无效的时间	-0.5	1	ns
t <sub>d</sub> (XCOH-XA)	延迟时间, XCLKOUT 高电平到地址有效的的时间		1	ns
t <sub>d</sub> (XCOHL-XRDL)	延迟时间, XCLKOUT 高电平/低电平到 $\overline{XRD}$ 低电平有效的的时间		0.5	ns
t <sub>d</sub> (XCOHL-XRDH)	延迟时间, XCLKOUT 高电平/低电平到 $\overline{XRD}$ 高电平无效的时间	-1.8	0.5	ns
t <sub>h</sub> (XA)XZCSH	保持时间, 区域芯片选择高电平无效之后地址有效时间	(1)		ns
t <sub>h</sub> (XA)XRD	保持时间, $\overline{XRD}$ 高电平无效后的地址有效时间	(1)		ns

(1) 在未激活周期期间, XINTF 地址总线将一直保持总线上产生的最后一个地址除了 XAO 一直处于高电平。这个包括对准周期。

**表 6-40 外部接口读取时序要求 ( 读取就绪, 1 个等待状态 )**

参数	描述	最小值	最大值	单位
t <sub>a</sub> (A)	访问时间, 从有效地址读取数据的时间		(LR + AR)-16 <sup>(1)</sup>	ns
t <sub>a</sub> (XRD)	访问时间, 从 $\overline{XRD}$ 低电平有效到读取有效数据的时间		AR-14 <sup>(1)</sup>	ns
t <sub>su</sub> (XD)XRD	建立时间, 在 $\overline{XRD}$ 选通脉冲高电平无效之前, 读取有效数据的时间	14		ns
t <sub>h</sub> (XD)XRD	保持时间, $\overline{XRD}$ 高电平无效之后读取数据有效的的时间	0		ns

(1) LR = 建立周期, 读取访问。 AR = 有效周期, 读取访问。 请参考 [表 6-34](#)。

**表 6-41 同步 XREADY 时序要求 ( ( 读取准备就绪, 1 个等待状态 ) )<sup>(1)</sup>**

参数	描述	最小值	最大值	单位
t <sub>su</sub> (XRDYsynchL)XCOHL	建立时间, XCLKOUT 高电平/低电平之前 XREADY (同步) 低电平的时间	12		ns
t <sub>h</sub> (XRDYsynchL)	保持时间, XREADY (同步) 低电平的时间	6		ns
t <sub>e</sub> (XRDYsynchH)	采样 XCLKOUT 边沿之前 XREADY (同步) 能够变为高电平的最早时间		3	ns
t <sub>su</sub> (XRDYsynchH)XCOHL	建立时间, XCLKOUT 高电平/低电平之前 XREADY (同步) 高电平的时间	12		ns
t <sub>h</sub> (XRDYsynchH)XZCSH	保持时间, 区域芯片选择高电平之后 XREADY (同步) 保持高电平的时间	0		ns

(1) 第一个 XREADY (同步) 样本相对于 [图 6-25](#) 中的 E 发生：

$$E = (XRDLEAD + XRDACTIVE) t_c(XTIM)$$

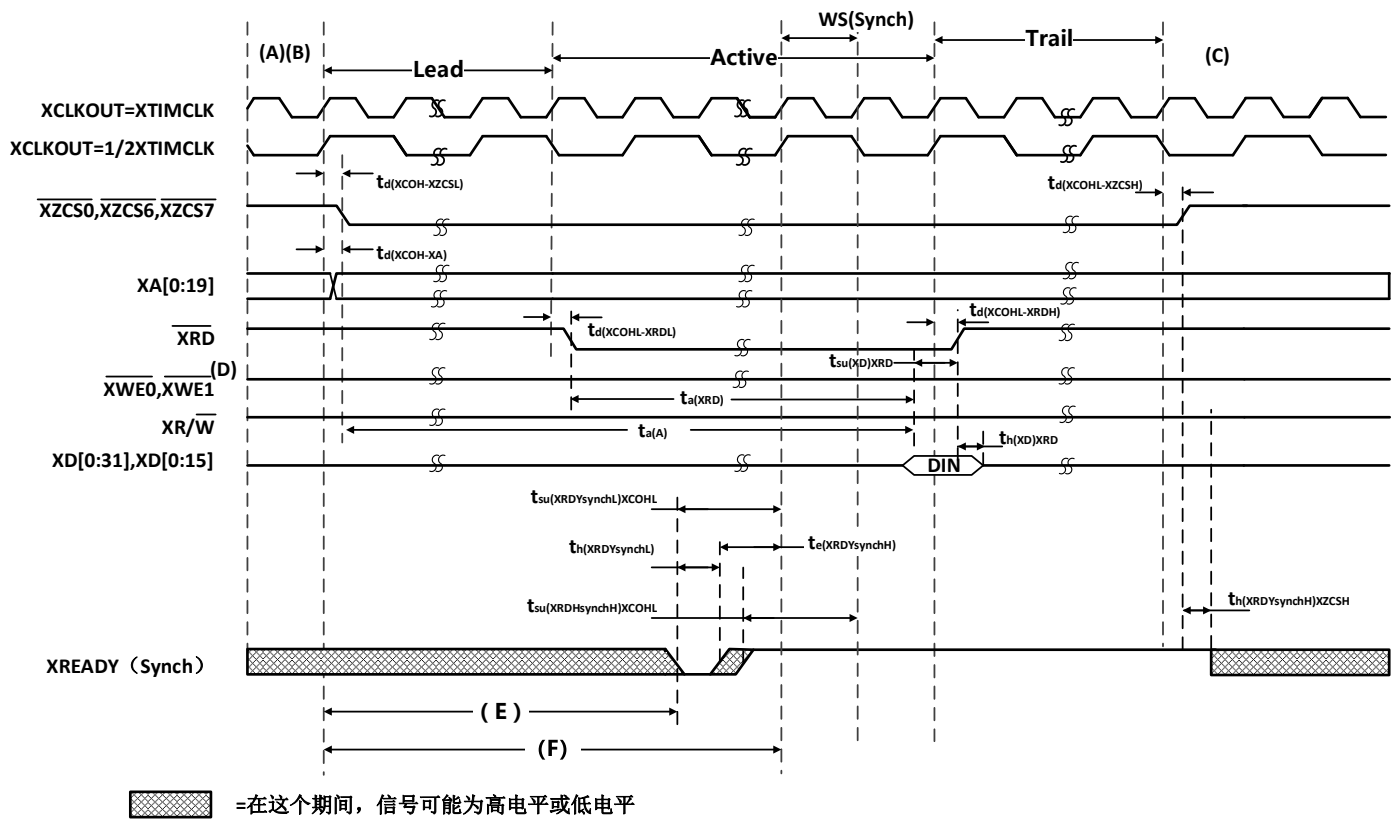
当首次采样时, 如果 XREADY (同步) 被发现为高电平, 那么访问将完成。如果发现 XREADY (同步) 为低电平, 它将在每个 t<sub>c</sub>(XTIM)内被重新采样直到它为高电平。

对于每个样本, 相对于访问开始的建立时间 (F) 可计算为：

$$F = (XRDLEAD + XRDACTIVE + n - 1) t_c(XTIM) - t_{su}(XRDYsynchL)XCOHL, \text{ 在这里, } n \text{ 为样本数量: } n = 1, 2, 3, \text{ 以此类推。}$$

**表 6-42 异步 XREADY 时序要求 ( 读取准备就绪, 1 个等待状态 )**

		最小值	最大值	单位
$t_{su}(XRDYAsynchL)XCOHL$	建立时间, 在 XCLKOUT 高电平/低电平之前 XREADY ( 异步 ) 低电平的时间	11		ns
$t_{h}(XRDYAsynchL)$	保持时间, XREADY ( 异步 ) 低电平的时间	6		ns
$t_{e}(XRDYAsynchH)$	采样 XCLKOUT 边沿之前, XREADY ( 异步 ) 能够变为高电平的最早时间		3	ns
$t_{su}(XRDYAsynchH)XCOHL$	建立时间, 在 XCLKOUT 高电平/低电平之前 XREADY ( 异步 ) 高电平的时间	11		ns
$t_{h}(XRDYAsynchH)XZCSH$	保持时间, 区域芯片选择高电平之后 XREADY ( 异步 ) 保持高电平的时间	0		ns



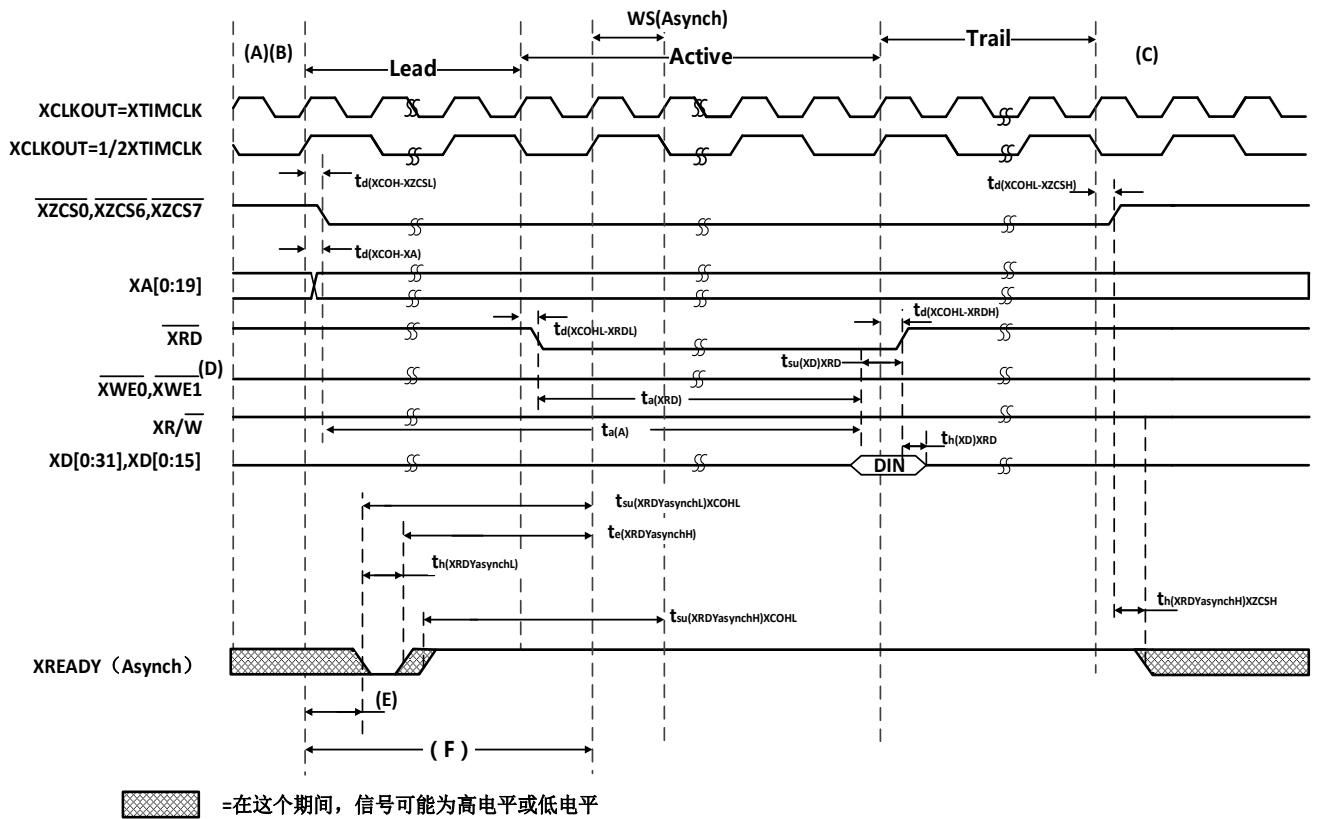
- A. 所有 XINTF 访问（建立周期）在 XCLKOUT 的上升沿上开始。当需要时,器件将在一个满足这个要求的访问之前插入一个对准周期。
- B. 在对准周期期间，所有信号将都被转换为它们的未激活状态。
- C. 在未激活周期期间，XINTF 地址总线将一直保持总线上产生的最后一个地址，除了 XAO,它将一直处于高电平。这个包 括对齐周期。
- D.  $\overline{XWE1}$ 仅在 32 位数据模式下有效。在 16 位模式中，该信号是 XAO。
- E. 对于每一个样本，从访问开始的建立时间可计算如下：  
 $D = (XRDLEAD + XRDACTIVE + n - 1) t_{c(XTIM)} - t_{su(XRDYsynchL)XCOHL}$  在这里，n 为样本数量：n = 1, 2, 3，以此类推。
- F. 关于相对于这个点的第一个样本： $F = (XRDLEAD + XRDACTIVE) t_{c(XTIM)}$

图 6-25 使用同步 XREADY 访问读取的示例

这个例子中使用 XTIMING 寄存器参数：

XRDLEAD	XRDACTIVE	XRDTRAIL	USEREADY	X2TIMING	XWRLEAD	XWRACTIVE	XWRTRAIL	READYMODE
≥1	3	≥1	1	0	N/A <sup>(1)</sup>	N/A <sup>(1)</sup>	N/A <sup>(1)</sup>	0=XREADY(Synch)

(1) 这个例子中 N/A=不适用（或“无关”）



- A. 所有 XINTF 访问（建立周期）在 XCLKOUT 的上升沿上开始。当需要时,器件将在一个满足这个要求的访问之前插入一个对准周期。
- B. 在对准周期期间，所有信号将被转换为它们的未激活状态。
- C. 在未激活周期期间，XINTF 地址总线将一直保持总线上产生的最后一个地址，除了 XAO,它将一直处于高电平。这个包括对齐周期。
- D.  $\overline{XWE1}$ 仅在 32 位数据模式下有效。在 16 位模式中，该信号是 XAO。
- E. 对于每一个样本，从访问开始的建立时间可计算如下：  
 $D = (XRDLEAD + XRDACTIVE - 3 + n)t_{c(XTIM)} - t_{su(XRDYasynchL)XCOHL}$  在这里，n 为样本数量： $n = 1, 2, 3$ ，以此类推。
- F. 关于相对于这个点的第一个样本： $F = (XRDLEAD + XRDACTIVE - 2)t_{c(XTIM)}$

图 6-26 使用异步 XREADY 访问读取的示例

这个例子中使用 XTIMING 寄存器参数：

XRDLEAD	XRDACTIVE	XRDTRAIL	USEREADY	X2TIMING	XWRLEAD	XWRACTIVE	XWRTRAIL	READYMODE
≥1	3	≥1	1	0	N/A <sup>(1)</sup>	N/A <sup>(1)</sup>	N/A <sup>(1)</sup>	1=XREADY(异步)

(1) 这个例子中 N/A=不适用（或“无关”）

### 6.14.8 带有一个外部等待状态的外部接口写入准备就绪时序

**表 6-43 外部接口写入开关特性（写入准备就绪，1 个等待状态）**

参数		最小值	最大值	单位
$t_{d(XCOH-XZCSL)}$	延迟时间, XCLKOUT高电平到区域芯片选择低电平有效的的时间		1	ns
$t_{d(XCOHL-XZCSH)}$	延迟时间, XCLKOUT高电平或者低电平到区域芯片选择高电平无效的时间	-0.5	1	ns
$t_{d(XCOH-XA)}$	延迟时间, XCLKOUT高电平到地址有效的的时间		1	ns
$t_{d(XCOHL-XWEL)}$	延迟时间, XCLKOUT高电平/低电平到 $\overline{XWE0}$ , $\overline{XWE1}$ 低电平的时间 <sup>(1)</sup>		3	ns
$t_{d(XCOHL-XWEH)}$	延迟时间, XCLKOUT高电平/低电平到 $\overline{XWE0}$ , $\overline{XWE1}$ 高电平的时间 <sup>(1)</sup>		3	ns
$t_{d(XCOH-XRNWL)}$	延迟时间, XCLKOUT高电平到XR/W低电平的时间		1	ns
$t_{d(XCOHL-XRNWH)}$	延迟时间, XCLKOUT高电平/低电平到XR/ $\overline{W}$ 高电平的时间	-1	1	ns
$t_{en(XD)XWEL}$	使能时间, 从 $\overline{XWE0}$ , $\overline{XWE1}$ 低电平驱动数据总线的时间 <sup>(1)</sup>	0		ns
$t_{d(XWEL-XD)}$	延迟时间, $\overline{XWE0}$ , $\overline{XWE1}$ 低电平有效后的数据有效时间 <sup>(1)</sup>		1	ns
$t_{h(XA)XZCSH}$	保持时间, 区域芯片选择高电平无效之后地址有效时间	<sup>(2)</sup>		ns
$t_{h(XD)XWE}$	保持时间, $\overline{XWE0}$ , $\overline{XWE1}$ 高电平无效之后写入数据有效时间 <sup>(1)</sup>	TW-2 <sup>(3)</sup>		ns
$t_{dis(XD)XRNW}$	XR/ $\overline{W}$ 高电平无效之后DSP释放数据总线的最长时间		4	ns

(1)  $\overline{XWE1}$ XWE1 只用于32 位数据总线模式。在16 位模式中, 该信号是XAO。

(2) 在未激活周期期间, XINTF 地址总线将一直保持总线上产生的最后一个地址除了 XAO, 仍为高电平。这个包括对齐周期。

(3) TW = 后置周期, 写入访问 (请见表 6-34)

**表 6-44 同步 XREADY 时序要求（写入准备就绪，1 个等待状态）<sup>(1)</sup>**

参数		最小值	最大值	单位
$t_{su(XRDYsynchL)XCOH}$	建立时间, XCLKOUT 高电平/低电平之前 XREADY (同步) 低电平的时间	12		ns
$t_{h(XRDYsynchL)}$	保持时间, XREADY (同步) 低电平的时间	6		ns
$t_{e(XRDYsynchH)}$	采样 XCLKOUT 边沿之前 XREADY (同步) 能够变为高电平的最早时间		3	ns
$t_{su(XRDYsynchH)XCOH}$	建立时间, XCLKOUT 高电平/低电平之前 XREADY (同步) 高电平的时间	12		ns
$t_{h(XRDYsynchH)XZCSH}$	保持时间, 区域芯片选择高电平之后 XREADY (同步) 保持高电平的时间		0	ns

(1) 第一个XREADY (同步) 采样相对于图 6-27 中的E 发生:

$E = (XWRLEAD + XWRACTIVE) t_{c(XTIM)}$ , 当首次采样时, 如果XREADY (同步) 为高电平, 那么访问将完成。如果XREADY (异步) 是低电平, 它在每个 $t_{c(XTIM)}$ 内再次采样, 直到它是高电平。

对于每个样本, 从访问开始的建立时间可计算为:

$D = (XWRLEAD + XWRACTIVE + n - 1) t_{c(XTIM)} - t_{su(XRDYsynchL)XCOHL}$ , 在这里, n 为样本数量:  $n = 1, 2, 3$ , 以此类推。

**表 6-45 异步 XREADY 时序要求（写入准备就绪，1 个等待状态）<sup>(1)</sup>**

参数		最小值	最大值	单位
$t_{su(XRDYsynchL)XCOH}$	建立时间, XCLKOUT 高电平/低电平之前 XREADY (异步) 低电平的时间	12		ns
$t_{h(XRDYsynchL)}$	保持时间, XREADY (异步) 低电平的时间	6		ns
$t_{e(XRDYsynchH)}$	采样 XCLKOUT 边沿之前 XREADY (异步) 能够变为高电平的最早时间		3	ns

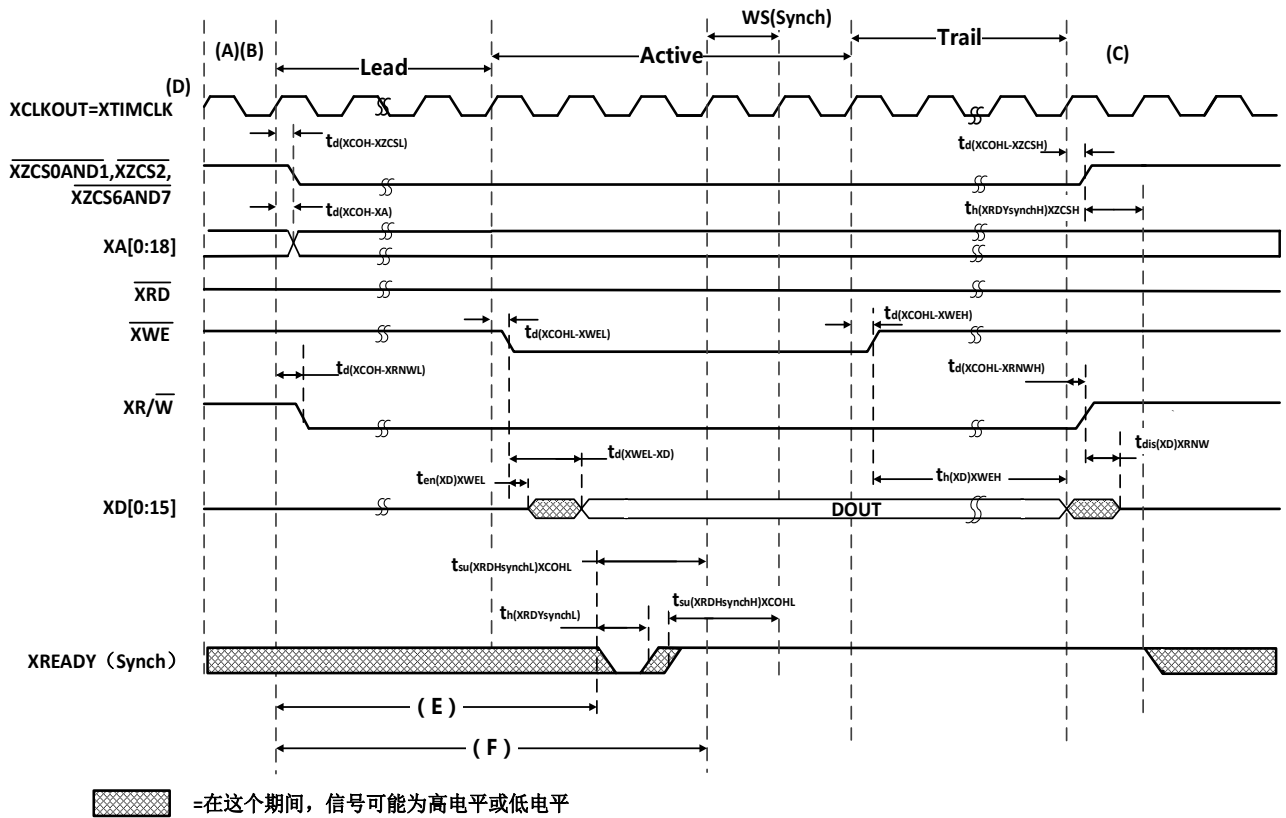
$t_{su(XRDYsynchH)XCOHL}$	建立时间, XCLKOUT 高电平/低电平之前 XREADY (异步) 高电平的时间	11		ns
$t_{h(XRDYsynchH)XZCSH}$	保持时间, 区域芯片选择高电平之后 XREADY (异步) 保持高电平的时间		0	ns

(1) 第一个XREADY (同步) 采样相对于图 6-27 中的E 发生:

$E = (XWRLEAD+XWRACTIVE-2) t_{c(XTIM)}$ 。在第一次取样时, 如果XREADY (异步) 是高电平, 则访问将完成。如果XREADY (异步) 是低电平, 它在每个 $t_{c(XTIM)}$ 内再次采样, 直到它是高电平。

对于每个样本, 访问开始的建立时间可计算为:

$D=(XWRLEAD+XWRACTIVE-3+n) t_{c(XTIM)}-t_{su(XRDYsynchL)XCOHL}$ , 在这里, n 为样本数量:  $n = 1, 2, 3$ , 以此类推。



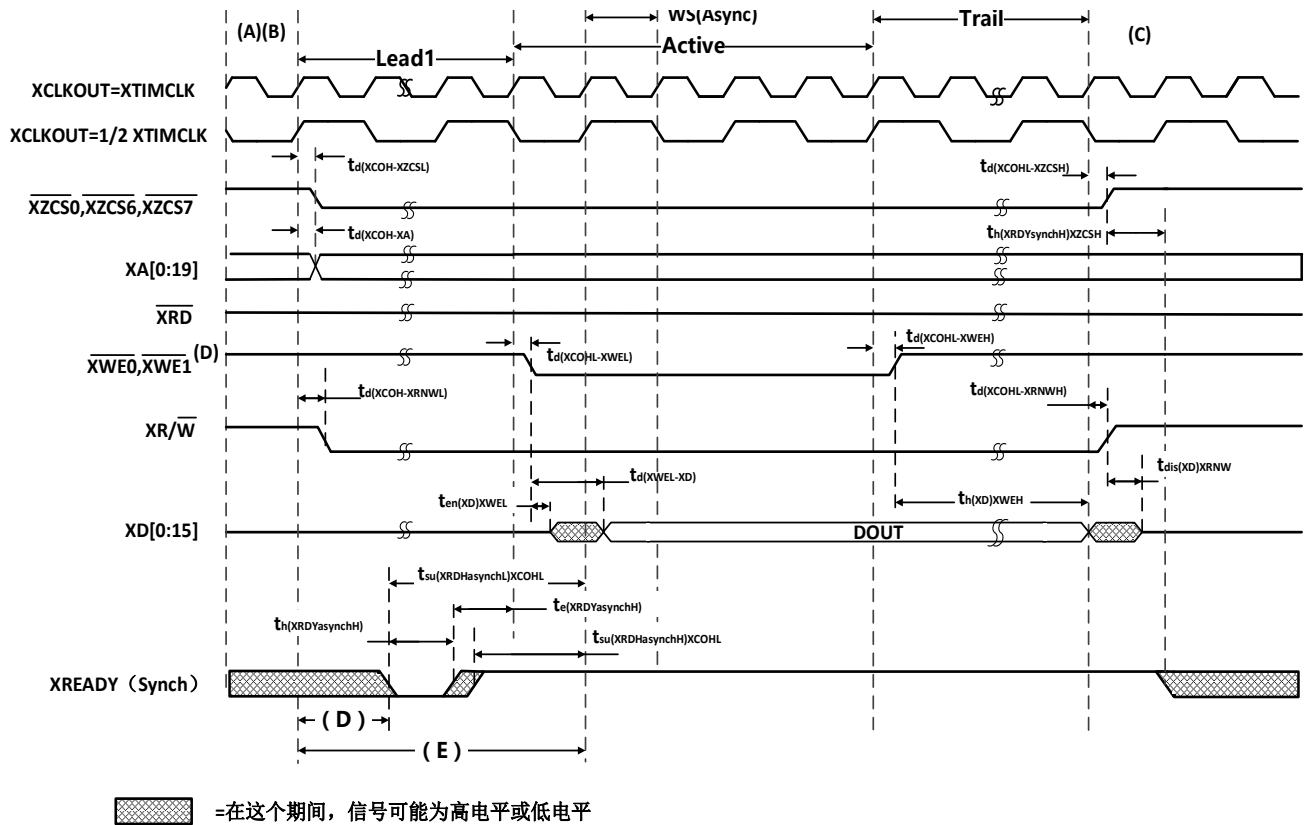
- A. 所有 XINTF 访问 (建立周期) 在 XCLKOUT 的上升沿上开始。当需要时, 器件将在一个满足这个要求的访问之前插入一个对准周期。
- B. 在对准周期期间, 所有信号将都被转换为它们的未激活状态。
- C. 在未激活周期期间, XINTF 地址总线将一直保持总线上产生的最后一个地址, 除了 XAO, 它将一直处于高电平。这个包括对齐周期。
- D.  $\overline{XWE1}$  仅在 32 位数据模式下有效。在 16 位模式中, 该信号是 XAO。
- E. 对于每一个样本, 从访问开始的建立时间可计算如下:  
 $D = (XRDLEAD+XRDACTIVE+n-1)t_{c(XTIM)}-t_{su(XRDYsynchL)XCOHL}$  在这里, n 为样本数量:  $n = 1, 2, 3$ , 以此类推。
- F. 关于相对于这个点的第一个样本:  $F=(XRDLEAD+XRDACTIVE)t_{c(XTIM)}$

图 6-27 使用同步 XREADY 访问写入的示例

这个例子中使用 XTIMING 寄存器参数:

XRDLEAD	XRDACTIVE	XRDTRAIL	USEREADY	X2TIMING	XWRLEAD	XWRACTIVE	XWRTRAIL	READYMODE
N/A <sup>(1)</sup>	N/A <sup>(1)</sup>	N/A <sup>(1)</sup>	1	0	≥1	3	≥1	0=XREADY(同步)

(1) 这个例子中 N/A=不适用 (或 “无关” )



- A. 所有 XINTF 访问（建立周期）在 XCLKOUT 的上升沿上开始。当需要时,器件将在一个满足这个要求的访问之前插入一个对准周期。
- B. 在对准周期期间,所有信号将被转换为它们的未激活状态。
- C. 在未激活周期期间, XINTF 地址总线将一直保持总线上产生的最后一个地址,除了 XAO,它将一直处于高电平。这个包括对齐周期。
- D.  $\overline{XWE1}$ 仅在 32 位数据模式下有效。在 16 位模式中,该信号是 XAO。
- E. 对于每一个样本,从访问开始的建立时间可计算如下:  
 $D = (XRDLEAD + XRDACTIVE - 3 + n) t_{c(XTIM)} - t_{su(XRDYsynchL)XCOH}$  在这里, n 为样本数量:  $n = 1, 2, 3$ , 以此类推。
- F. 关于相对于这个点的第一个样本:  $F = (XRDLEAD + XRDACTIVE - 2) t_{c(XTIM)}$

图 6-28 使用异步 XREADY 访问写入的示例

这个例子中使用 XTIMING 寄存器参数:

XRDLEAD	XRDACTIVE	XRDTRAIL	USEREADY	X2TIMING	XWRLEAD	XWRACTIVE	XWRTRAIL	READYMODE
N/A <sup>(1)</sup>	N/A <sup>(1)</sup>	N/A <sup>(1)</sup>	1	0	≥1	3	≥1	1=XREADY(异步)

(1) 这个例子中 N/A=不适用 (或“无关”)

### 6.14.9 XHOLD 和 XHOLDA 定时

如果在 XHOLD 和 XHOLDA 同时为低电平时 (授权外部总线访问) HOLD 模式位被设定, XHOLDA 信号被强制为高电平 (在当前周期的末尾) 并且外部接口不再为高阻抗模式。

复位时 ( $\overline{XRS}$ ), HOLD 模式位被设定为 0。如果 XHOLD 信号在系统复位时为低电平有效, 总线和所有信号选通必须为高阻抗模式, 并且 XHOLDA 信号也被驱动为低电平有效。



当 HOLD 模式被启用并且 $\overline{\text{XHOLDA}}$ 为低电平有效（外部总线置为有效），CPU 仍然可以从内部存储器执行代码。如果对外部接口进行访问，在 $\overline{\text{XHOLD}}$ 信号被去除前，CPU 暂停。

一个内部 DMA 请求，当被批准时，将以下信号置为高阻抗模式：

A[19:0]                     $\overline{\text{XZCS0}}$

XD[31:0],XD[15:0]       $\overline{\text{XZCS6}}$

XWE0,XWE1,             $\overline{\text{XZCS7}}$

$\overline{\text{XRD}}$

$\overline{\text{XR/W}}$

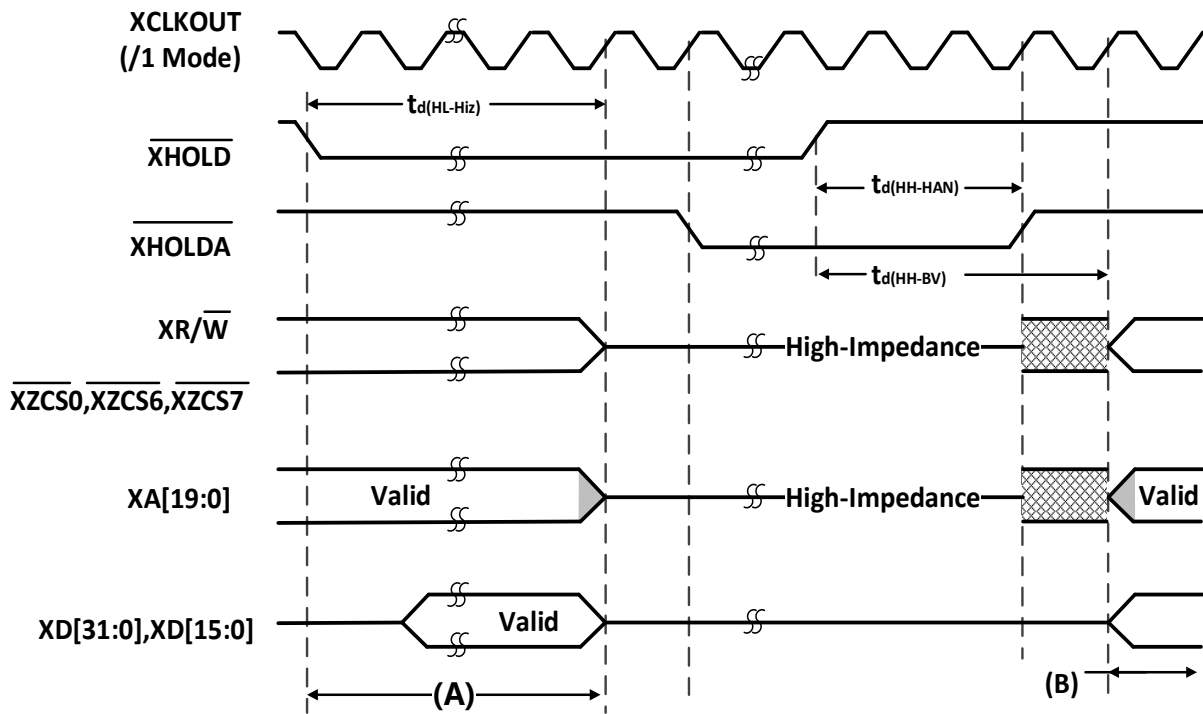
在这些信号事件期间，所有在这个组中未列出的其它信号保持在它们的缺省值或者功能运行模式。

**表 6-46  $\overline{\text{XHOLD}}$  和  $\overline{\text{XHOLDA}}$ 时序要求 ( XCLKOUT=XTIMCLK ) (1)(2)**

参数		最小值	最大值	单位
$t_{d(\text{HL-HiZ})}$	延迟时间， $\overline{\text{XHOLD}}$ 低电平到所有地址、数据、和控制上的低阻抗到高阻抗的时间		$4t_{c(\text{XTIM})} + 30$	ns
$t_{d(\text{HL-HAL})}$	延迟时间， $\overline{\text{XHOLD}}$ 低电平至 $\overline{\text{XHOLDA}}$ 低电平的时间		$5t_{c(\text{XTIM})} + 30$	ns
$t_{d(\text{HH-HAH})}$	延迟时间， $\overline{\text{XHOLD}}$ 高电平至 $\overline{\text{XHOLDA}}$ 高电平的时间		$3t_{c(\text{XTIM})} + 30$	ns
$t_{d(\text{HH-BV})}$	延迟时间， $\overline{\text{XHOLD}}$ 高电平到总线有效的的时间		$4t_{c(\text{XTIM})} + 30$	ns
$t_{d(\text{HL-HAL})}$	延迟时间， $\overline{\text{XHOLD}}$ 低电平至 $\overline{\text{XHOLDA}}$ 低电平的时间		$4t_{c(\text{XTIM})} + 2t_{c(\text{XCO})} + 30$	ns

(1) 当在 $\overline{\text{XHOLD}}$ 上检测到一个低电平信号时，所有等待的 XINTF 访问将在总线被置为一个高阻抗状态前完成。

(2)  $\overline{\text{XHOLD}}$ 的状态被锁存在 XTIMCLK 的上升边沿上。



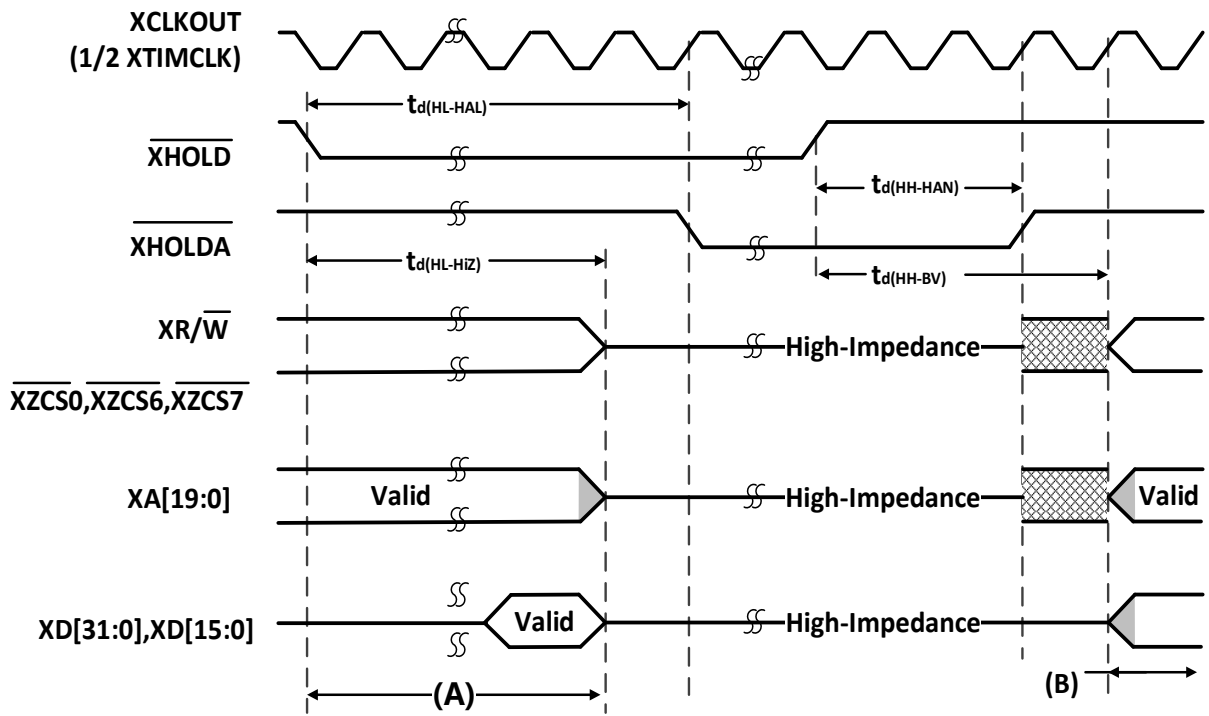
- A. 所有等待中的 XINTF 访问被完成。
- B. 正常 XINTF 运行重新开始。

图 6-29 外部接口保持波形

表 6-47 XHOLD 和 XHOLDA 时序要求 ( XCLKOUT = 1/2 XTIMCLK ) (1)(2)(3)

参数	描述	最小值	最大值	单位
$t_{d(HL-HIZ)}$	延迟时间, XHOLD 低电平到所有地址、数据、和控制上的低阻抗到高阻抗的时间		$4t_{c(XTIM)} + t_{c(XCO)} + 30$	ns
$t_{d(HL-HAL)}$	延迟时间, XHOLD 低电平至 XHOLDA 低电平的时间		$4t_{c(XTIM)} + 2t_{c(XCO)} + 30$	ns
$t_{d(HH-HAH)}$	延迟时间, XHOLD 高电平至 XHOLDA 高电平的时间		$4t_{c(XTIM)} + 30$	ns
$t_{d(HH-BV)}$	延迟时间, XHOLD 高电平到总线有效的的时间		$6t_{c(XTIM)} + 30$	ns

- (1) 当在 XHOLD 上检测到一个低电平信号时, 所有等待的 XINTF 访问将在总线被置为一个高阻抗状态前完成。
- (2) XHOLD 的状态被锁存在 XTIMCLK 的上升边沿上。
- (3) 在 XHOLD 被检测为高电平或者低电平后, 所有总线转换和 XHOLDA 转换将相对于 XCLKOUT 的上升边沿发生。因此, 对于这个 XCLKOUT = 1/2 XTIMCLK 的模式, 转换最多可以早于最大额定值 1 XTIMCLK 周期发生。



- A. 所有等待中的 XINTF 访问被完成。
- B. 正常 XINTF 运行重新开始。

图 6-30 XHOLD 和 XHOLDA 时序要求 (XCLKOUT = 1/2 XTIMCLK)

## 6.15 片载模数转换器 ADC

**表 6-48 ADC 电气特性 (在推荐的运行条件) (1)(2)**

参数	最小值	典型值	最大值	单位	
<b>DC 技术规范<sup>(3)</sup></b>					
分辨率	12			位	
ADC 时钟	0.001		7.5	MHZ	
<b>精度</b>					
INL (积分非线性)	1-7.5MHz ADC 时钟 (3.75MSPS)		-7	7	最低有效位 (LSB)
DNL (微分非线性) <sup>(4)</sup>			-2	2	LSB
偏移误差 <sup>(5)(3)</sup>			-15	15	最低有效位 (LSB)
带有内部基准的总增益误差 <sup>(6)(3)</sup>			-30	30	最低有效位 (LSB)
带有外部基准的总增益误差 <sup>(3)</sup>			-30	30	LSB
通道到通道偏移变化			-4	4	LSB
通道到通道增益变化			-4	4	LSB
<b>模拟输入</b>					
模拟输入电压 (ADCINx 至 ADCLO) <sup>(7)</sup>	0		3	V	
ADCLO	-5		5	mV	
输入电容值		10		pF	
输入漏电流	-5		5	μA	
<b>内部参考基准<sup>(6)</sup></b>					
V <sub>ADCREFP</sub> -在基于内部基准的引脚上的 ADCREFP 输出电压		1.275		V	
V <sub>ADCREFM</sub> -在基于内部基准的引脚上的 ADCREFM 输出电压		0.525		V	
电压差异, ADCREFP-ADCREFM		0.75		V	
温度系数		50		PPM/°C	
<b>外部基准电压<sup>(6)(8)</sup></b>					
V <sub>ADCREFIN</sub> -在推荐的 ADCREFIN 引脚 0.2%或者更好的精确基准上的外部基准电压输入	ADCREFSEL[15:14]=11b		1.024	V	
	ADCREFSEL[15:14]=10b		1.500	V	
	ADCREFSEL[15:14]=01b		2.048	V	
<b>AC 技术规范</b>					
SINAD (100kHz) 信噪比+失真		56.1		dB	
SNR (100kHz) 信噪比		58		dB	
THD (100kHz) 总谐波失真		-61		dB	
ENOB (100kHz) 有效位数		9		位	
SFDR (100kHz) 无杂散动态范围		62		dB	

(1) 在 7.5 MHz ADCCLK 上测得。

(2) 这个表中的所有电压相对 V<sub>SSA2</sub>。

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

(4) Advchip 指定 ADC 将无丢码。

(5) 1 个 LSB 有 3.0/4096=0.732 mV 的加权值。

- (6) 一个单一内部/外部带隙基准为 ADCREFP 和 ADCREFM 信号供源，因此，这些电压可一起跟踪。ADC 转换器使用这两个之间的差异作为它的基准。这里列出的针对内部基准的总体增益误差包括温度范围内内部带隙的运动。针对外部基准选项的温度范围内的增益误差将取决于所使用源的温度参数。
- (7) 应用到一个模拟输入引脚上的高于  $V_{DDA} + 0.3V$  或者低于  $V_{SS} - 0.3V$  的电压有可能暂时影响另外引脚的转换。为了避免这种情况，模拟输入应该被保持在这些限值内。
- (8) Advchip 建议使用高精度外部基准部件 REF3020/3120 或者针对 2.048V 基准的等效器件。

### 6.15.1 ADC 加电控制位定时

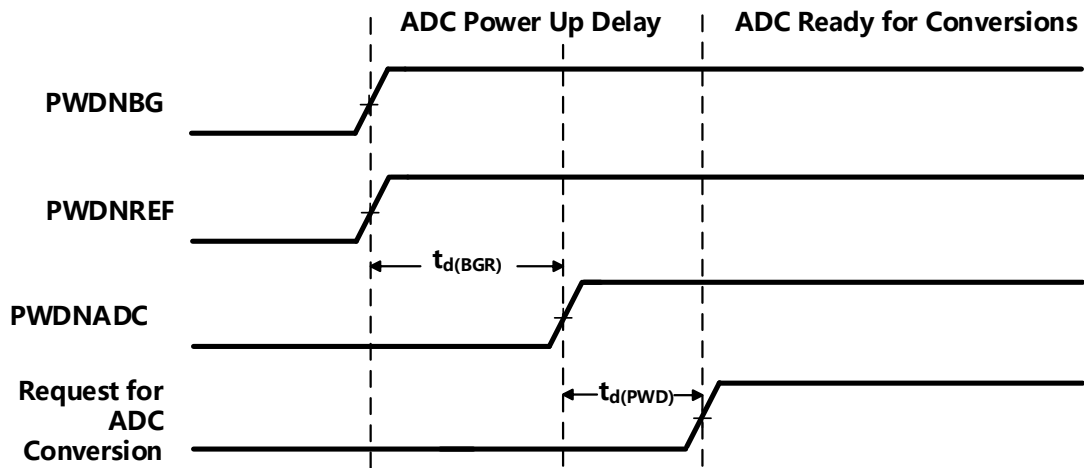


图 6-31 ADC 加电控制位时序

表 6-49 ADC 加电延迟

参数 <sup>(1)</sup>	条件	最小值	典型值	最大值	单位
$t_{d(BGR)}$	带隙基准稳定所需的延迟时间。ADCTRL3 寄存器的位 7 和 6(ADCBGRFDN1/0)在 PWDNADC 位被启用前被设定为 1。ADCREFP/ADCREFM 外接 2.2 $\mu$ f 电容时			5	ms
	ADCREFP/ADCREFM 外接 10 $\mu$ f 电容时			10	
$t_{d(PWD)}$	断电控制稳定所需的延迟时间。带隙基准稳定所需的位延迟时间。ADCTRL3 寄存器的位 7 和 6(ADCBGRFDN1/0)在 PWDNADC 位被启用前被设定为 1。ADCTRL3 寄存器的位 5(PWDNADC)在任何 ADC 转换启动前被设定为 1。	20	50		$\mu$ s
				1	ms

(1) 时序保持与 ADP32F1x ADC 模块的兼容性。AVP32F335 ADC 也同时支持驱动所有 3 位，并在首次转换前等待  $t_{d(BGR)}$ ms。

表 6-50 不同 ADC 配置的典型电流消耗 (在 7.5MHz ADCCLK 上) <sup>(1)(2)</sup>

ADC 运行模式	条件	VDDA18	VDDA3.3	单位
模式 A - 运行, 模式	<ul style="list-style-type: none"> <li>● BG REF 被启用</li> <li>● PWD 被禁用</li> </ul>	23	5	mA
模式 B - 快速唤醒模式	<ul style="list-style-type: none"> <li>● ADC 时钟被启用</li> <li>● BG 和 REF 被启用</li> <li>● PWD 被启用</li> </ul>	5	0.5	mA

模式 C - 只比较器可用模式	<ul style="list-style-type: none"> <li>● ADC 时钟被启用</li> <li>● BG 和 REF 被禁用</li> <li>● PWD 被启用</li> </ul>	5	20	μA
模式 D - 关闭模式	<ul style="list-style-type: none"> <li>● ADC 时钟被禁用</li> <li>● BG 和 REF 被禁用</li> <li>● PWD 被启用</li> </ul>	5	15	μA

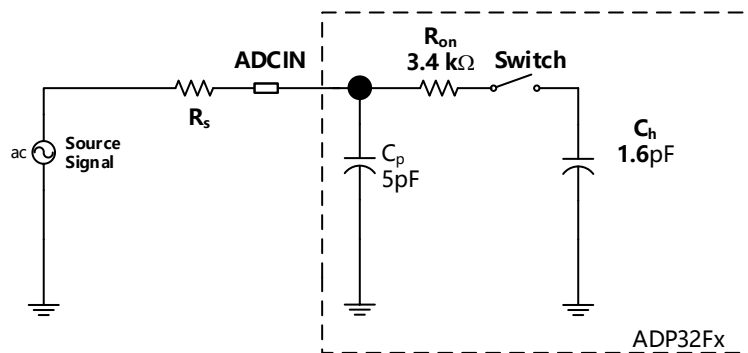
(1) 测试条件：

SYSCLKOUT= 150MHz

ADC 模块时钟 = 7.5MHz

ADC 在模式 A 中执行一个所有 16 通道的连续转换

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



Typical Values of the Input Circuit Components:

Switch Resistance ( $R_{on}$ ): 3.4 kΩ

Sampling Capacitor ( $C_h$ ): 1.6 pF

Parasitic Capacitance ( $C_p$ ): 5 pF

Source Resistance ( $R_s$ ): 50 Ω

图 6-32 ADC 输入阻抗模型

### 6.15.2 基本定义

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

模拟输入：片载 ADC 由 16 个模拟输入组成，这些通道或者同时采样，或者每次两个通道采样。这些输入为软件可选。

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

转换时间：转换可以在两个不同的转换模式中执行：

- 顺序采样模式 (SMODE = 0)
- 同步采样模式 (SMODE= 1)

### 6.15.3 ADC 顺序采样模式 (单通道) (SMODE=0)

在顺序采样模式下，ADC 能够持续在任一通道 ( $A_x$  至  $B_x$ ) 上转换输入信号。ADC 能够在来自

ePWM，软件触发器，或者来自一个外部ADCSOC信号的事件触发上启动转换。如果SMODE位为0，ADC将在每个采样/保持脉冲上的所选通道上进行转换。下面对转换时间和结果寄存器更新的延迟进行解释说明。ADC中断标志在结果寄存器更新之后的几个SYSCLKOUT周期内被设定。所选通道将在采样/保持脉冲的每个下降边沿上被采样。采样/保持脉冲宽度可被设定为1个ADC时钟宽（最小值）或者16个ADC时钟宽（最大值）。

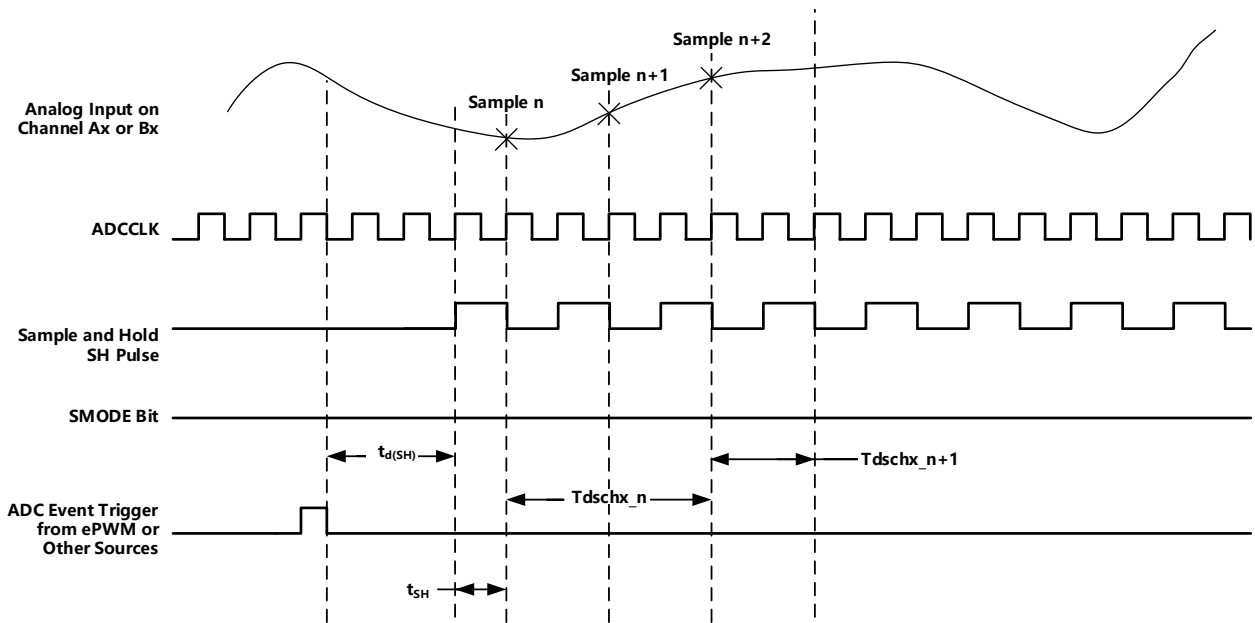


图 6-33 顺序采样模式（单通道）时序

表 6-51 顺序采样模式时序

参数		SAMPLE (样本)n	SAMPLE n+1	在 7.5MHz ADC 时钟上, $t_c(ADCCLK) = 133ns$	注释
$t_{d(SH)}$	从事件触发器到采样的延迟时间	$2.5t_c(ADCCLK)$			
$t_{SH}$	采样/保持宽度/采集宽度	$(1 + Acqps) * t_c(ADCCLK)$		Acqps=0 时为 133ns	Acqps 值=0-15 ADCTRL1[8:11]
$t_{d(schx_n)}$	结果寄存器出现第一个结果的延迟时间	$4t_c(ADCCLK)$		532ns	
$t_{d(schx_{n+1})}$	结果寄存器中出现连续结果的延迟时间		$(2 + Acqps) * t_c(ADCCLK)$	266ns	

### 6.15.4 ADC 同步采样模式（双通道）（SMODE=1）

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

设定。所选通道将在采样/保持脉冲的下降边沿上被同时采样。采样/保持脉冲宽度可被设定为1个ADC时钟宽（最小值）或者16个ADC时钟宽（最大值）。

注

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

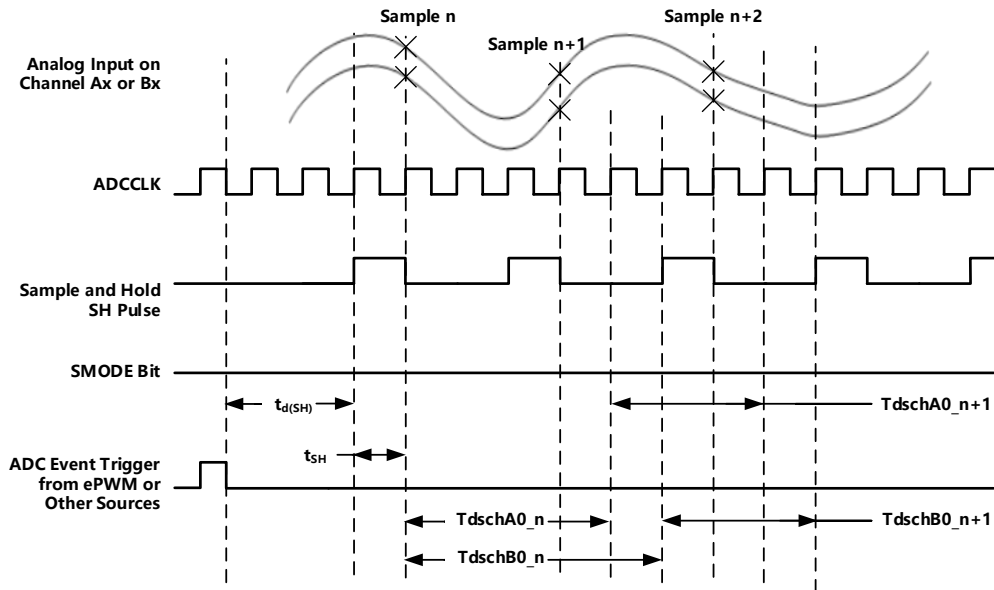


图 6-34 同步采样模式（双通道）时序

表 6-52 同步采样模式时序

参数		SAMPLE (样本)n	SAMPLE n+1	在 7.5MHz ADC 时钟上, $t_c(ADCCLK)=133ns$	注释
$t_{d(SH)}$	从事件触发器到采样的延迟时间	$2.5t_c(ADCCLK)$			
$t_{SH}$	采样/保持宽度/采集宽度	$(1+Acqps)^* t_c(ADCCLK)$		Acqps=0 时为 133ns	Acqps 值=0-15 ADCTRL1[8:11]
$t_{d(schA0_n)}$	结果寄存器出现第一个结果的延迟时间	$4t_c(ADCCLK)$		532ns	
$t_{d(schB0_n)}$	结果寄存器出现第一个结果的延迟时间	$5t_c(ADCCLK)$		665ns	
$t_{d(schA0_{n+1})}$	结果寄存器中出现连续结果的延迟时间		$(3+Acqps)^* t_c(ADCCLK)$	399ns	
$t_{d(schB0_{n+1})}$	结果寄存器中出现连续结果的延迟时间		$(3+Acqps)^* t_c(ADCCLK)$	399ns	

### 6.15.5 详细说明

#### 积分非线性

积分非线性是指每个独立代码从零至满刻度所画的一条直线上的偏离。在首次代码转换前，作为



零点的点出现1/2 LSB。满刻度点被定义为超过最后一次代码转换的级别1/2 LSB。这个偏离为每一个特定代码的中心到这两个点之间的精确直线的距离。

### 微分非线性

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

### 零偏移

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

### 增益误差

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

### 信噪比+失真(SINAD)

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

### 有效位数(ENOB)

对于一个正弦波，SINAD 可用位的数量表示。使用下面的公式， $N = \frac{\text{SINAD}-1.76}{6.02}$ 有可能获得一个用 N（位的有效数）表达的性能测量值。因此，对于在给定输入频率上用于正弦波输入的器件的有效位数可从这个测得的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。

## 6.16 多通道缓冲串行端口(McBSP) 模块

### 6.16.1 McBSP 发送和接收时序

**表 6-53 McBSP 时序要求<sup>(1)(2)</sup>**

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

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

(2)  $2P=1/CLKG$ , 单位为 ns。CLKG 是采样率发生器复用的输出。  

$$CLKG = \frac{CLKSRG}{(1+CLKGDV)} CLKSRG$$
 可由 LSPCLK, CLKX, CLKR 供源  
 $CLKSRG \leq (SYSCLKOUT/2)$ 。McBSP 的性能受到 I/O 缓冲器开关速度的限制。

(3) 内部时钟预分频器必须被调整, 这样的话, McBSP 时钟 (CLKG, CLKX, CLKR) 速度不会大于 I/O 缓冲器速度限制 (25MHz)。

**表 6-54 McBSP 开关特性<sup>(1)(2)</sup>**

编号	参数		最小值	最大值	单位	
M1	$t_{c(CKRX)}$	周期时间, CLKR/X	CLKR/X 内部	2P	ns	
M2	$t_{w(CKRXH)}$	脉冲持续时间, CLKR/X 高电平的时间	CLKR/X 内部	D-5 <sup>(3)</sup>	D+5 <sup>(3)</sup>	ns
M3	$t_{r(CKRXL)}$	脉冲持续时间, CLKR/X 低电平的时间	CLKR/X 内部	C-5 <sup>(3)</sup>	C+5 <sup>(3)</sup>	ns
M4	$t_{f(CKRXH-FRV)}$	延迟时间, CLKR 高电平到内部 FSR 有效的时间	CLKR 内部	0	4	ns
			CLKR 外部	3	27	
M5	$t_{su(CKXH-FXV)}$	延迟时间, CLKX 高电平到内部 FSR 有效的时间	CLKR 内部	0	4	ns
			CLKR 外部	3	27	
M6	$t_{dis(CKXH-DXHZ)}$	禁用时间, CLKX 高电平到 DX 在最后一个数据位后为高阻抗的时间	CLKR 内部		8	ns

				CLKR 外部		14	
M7	$t_{d(CLKH-DX)}$	延迟时间, CLKX 高电平到 DX 有效的时间。 这应用于除了第一个位之外的所有被发送的位。		CLKR 内部		9	ns
				CLKR 外部		28	
		延迟时间, CLKX 高电平到 DX 有效的时间	DXENA=0	CLKR 内部		8	
				CLKR 外部		14	
		当处于数据延迟 1 或者 2 ( XDATDLY=01b 或者 10b ) 模式时, 只应用于发送的第一个位。	DXENA=1	CLKR 内部		P+8	
				CLKR 外部		P+14	
M8	$t_{en(CLKH-DX)}$	使能时间, CLKX 高电平待 DX 被驱动的时间	DXENA=0	CLKR 内部		0	ns
				CLKR 外部		6	
		当处于数据延迟 1 或者 2 ( XDATDLY=01b 或者 10b ) 模式时, 只应用于发送的第一个位。	DXENA=1	CLKR 内部		P	
				CLKR 外部		P+6	
M9	$t_{d(FXH-DX)}$	延迟时间, FSX 高电平到 DX 有效的时间	DXENA=0	FSX 内部		8	ns
				FSX 外部		14	
		当处于数据延迟 0 ( XDATDLY=00b ) 模式时, 只应用于发送的第一个位。	DXENA=1	FSX 内部		P+8	
				FSX 外部		P+14	
M10	$t_{d(FXH-DX)}$	使能时间, FSX 高电平到 DX 驱动的时间	DXENA=1	FSX 内部		0	ns
				FSX 外部		6	
		当处于数据延迟 0 ( XDATDLY=00b ) 模式时, 只应用于发送的第一个位。		FSX 内部		P	
				FSX 外部		P+6	

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

(2)  $2P=1/CLKG$ , 单位为 ns。

(3) C=CLKRX 低脉冲宽度 = P

D=CLKRX 高脉冲宽度 = P

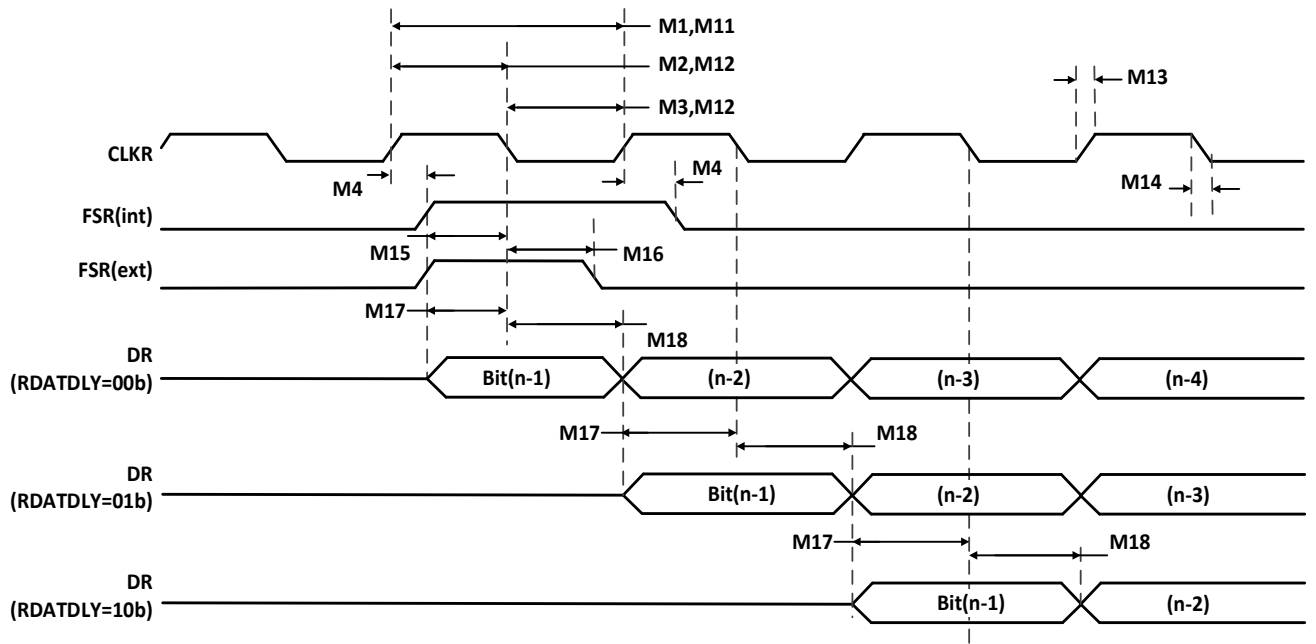


图 6-35 McBSP 接收时序

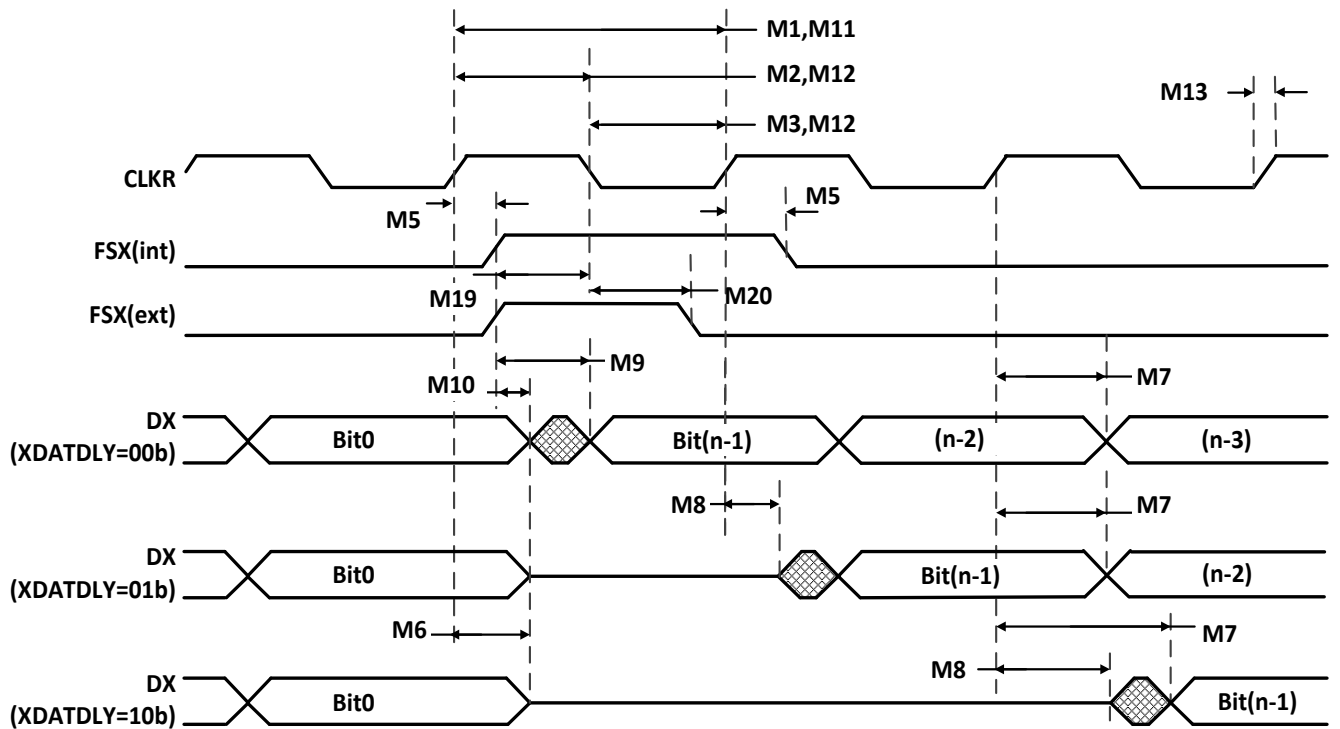


图 6-36 McBSP 发送时序

### 6.16.2 McBSP 作为 SPI 主控或者受控时序

表 6-55 McBSP 作为 SPI 主控或者受控定时要求 (CLKSTP=10b, CLKXP=0)

编号	参数		主控		受控		单位
			最小值	最大值	最小值	最大值	
M30	$t_{su}(DRV-CKXL)$	建立时间, 在 CLKX 低电平之前, DR 的有效时间	30		8P-10		ns
M31	$t_h(CKXL-DRV)$	保持时间, 在 CLKX 低电平之后, DR 的有效时间	1		8P-10		ns
M32	$t_{su}(BFXL-CKXH)$	建立时间, 在 CLKX 高电平之前, FSX 为低电平的时间			8P+10		ns
M33	$t_c(CKX)$	周期时间, CLKX	2P <sup>(1)</sup>		16P		ns

(1) 2P=1/CLKG

表 6-56 McBSP 作为 SPI 主控或者受控开关特性 (CLKSTP=10b, CLKXP=0)

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

(1) 2P=1/CLKG

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

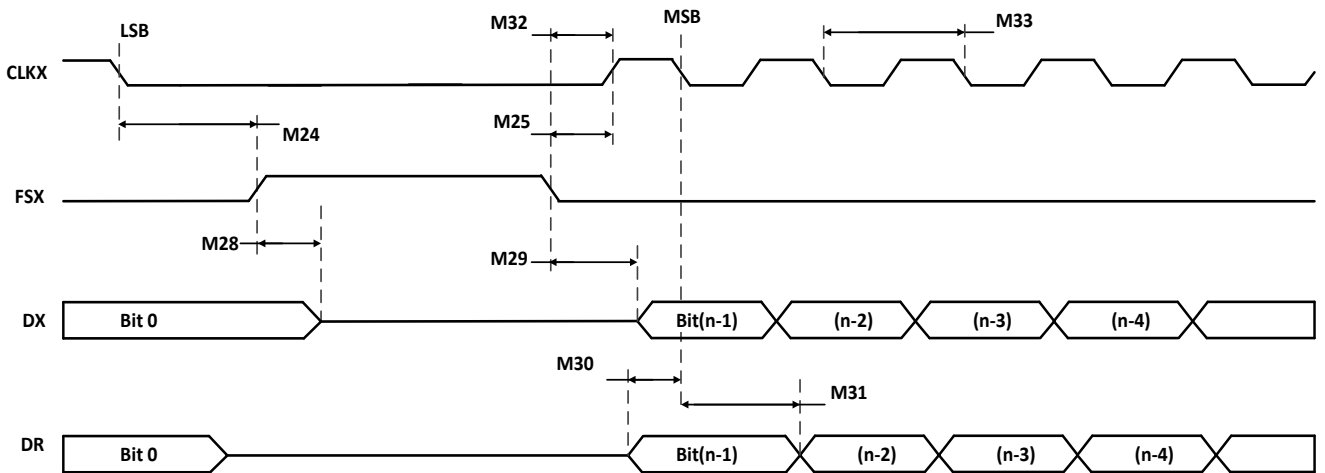


图 6-37 作为 SPI 主控或者受控时的 McBSP 时序 : CLKSTP=10b, CLKXP=0

表 6-57 McBSP 作为 SPI 主控或者受控定时要求 (CLKSTP=11b, CLKXP=0)

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

(1) 2P=1/CLKG

表 6-58 McBSP 作为 SPI 主控或者受控开关特性 (CLKSTP=11b, CLKXP=0)

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

(1) 2P=1/CLKG

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

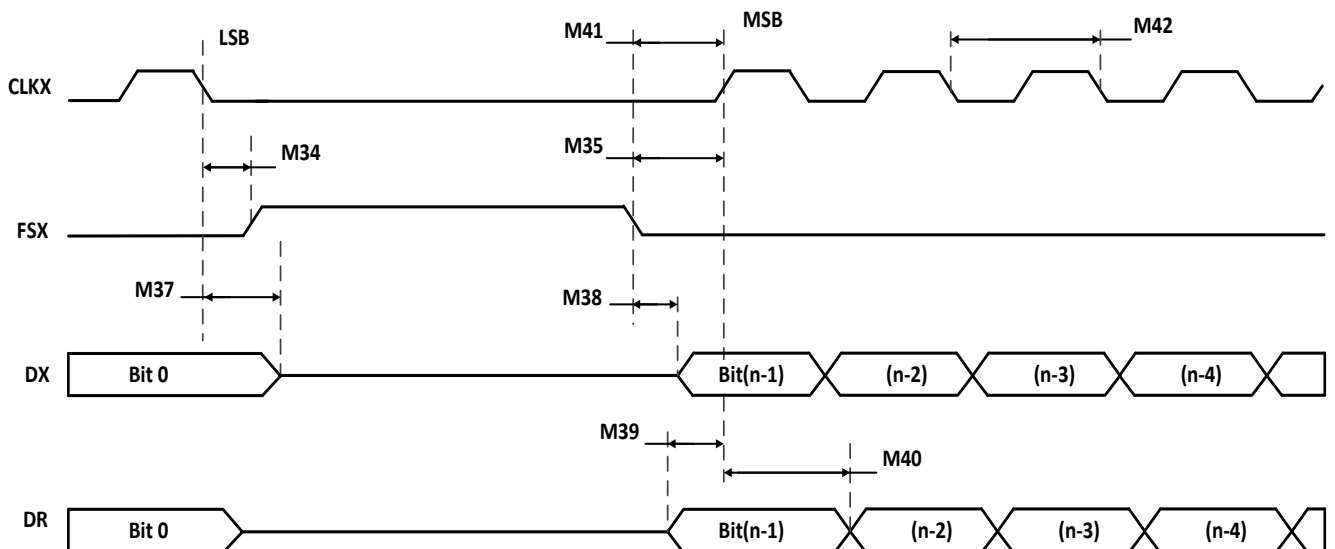


图 6-38 作为 SPI 主控或者受控时的 McBSP 时序 : CLKSTP=11b, CLKXP=0

表 6-59 McBSP 作为 SPI 主控或者受控定时要求 (CLKSTP=10b, CLKXP=1)

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

(1) 2P=1/CLKG

表 6-60 McBSP 作为 SPI 主控或者受控开关特性 (CLKSTP=10b, CLKXP=1)

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

(1) 2P=1/CLKG

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

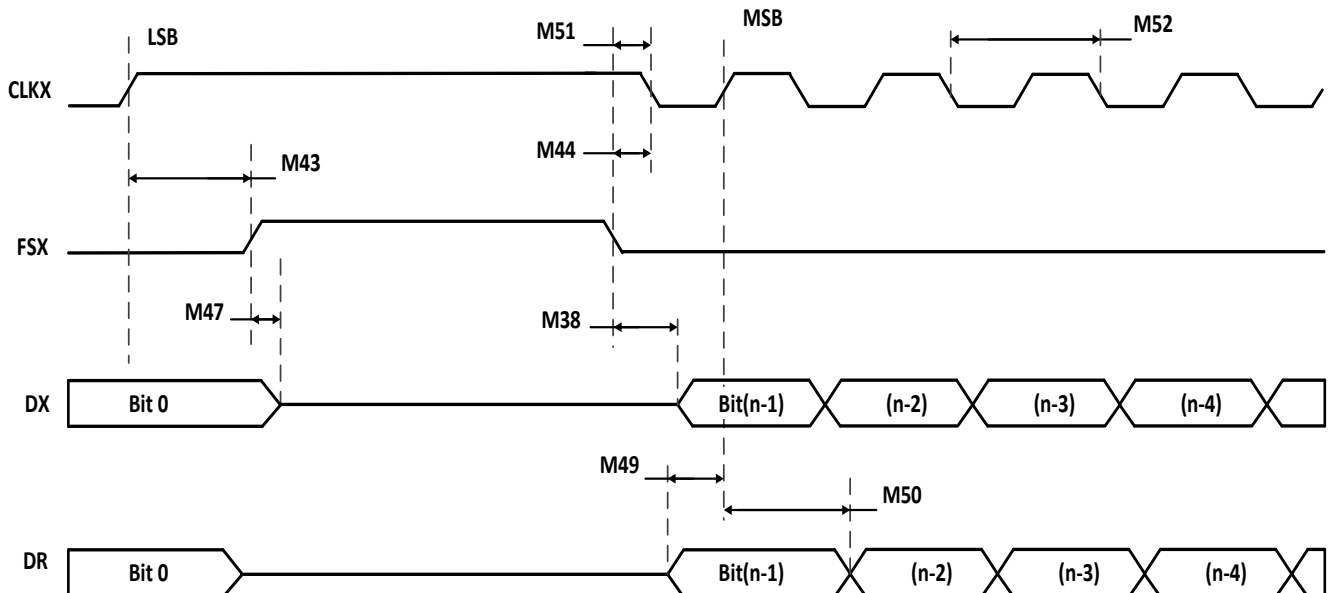


图 6-39 作为 SPI 主控或者受控时的 McBSP 时序 : CLKSTP=10b, CLKXP=1

表 6-61 McBSP 作为 SPI 主控或者受控定时要求 (CLKSTP=11b, CLKXP=1)

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

2P=1/CLKG

表 6-62 McBSP 作为 SPI 主控或者受控开关特性 (CLKSTP=11b, CLKXP=1)<sup>(1)</sup>

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

(1) 2P=1/CLKG

(2) C=CLKRX 低脉冲宽度 = P

D=CLKRX 高脉冲宽度 = P

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

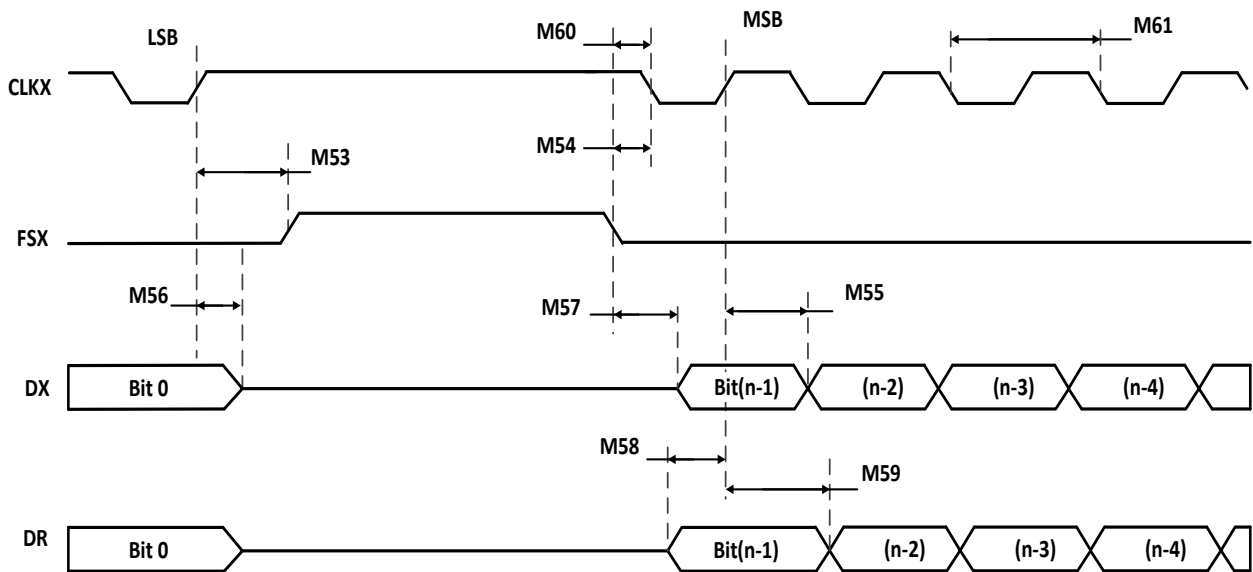


图 6-40 作为 SPI 主控或者受控时的 McBSP 时序: CLKSTP=11b, CLKXP=1



## 6.17 闪存定时

**表 6-63 闪存对于 (S) 温度材料的耐受度<sup>(1)</sup>**

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

(1) 所示温度范围之外的写入/擦除操作并未说明, 有可能影响耐受数。

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

**表 6-64 150MHz SYSCLKOUT 上的闪存参数**

参数		测试条件	最小值	典型值	最大值	单位
编程时间 <sup>(1)</sup>	16 位字			50		μs
	32K 扇区			1600		ms
擦除时间 <sup>(1)</sup>	32K 扇区			900		ms
I <sub>DDP</sub> <sup>(2)</sup>	擦除/编程周期期间的 V <sub>DD</sub> 流耗			60		mA
I <sub>DDOP</sub> <sup>(2)</sup>	擦除/编程周期期间的 V <sub>DDIO</sub> 流耗			5		mA

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

(2) 室温下包括函数调用开销在内的典型参数, 是在所有外设关闭时的参数。

(3) AVP32F335 中 FLASH 总共 256K×16 位, 分为 8 个扇区, 每个扇区扇区空间为 32K×16 位。

**表 6-65 闪存/ OTP 访问时序**

参数		最小值	最大值	单位
t <sub>a</sub> (fp)	页式闪存访问时间	30		ns
t <sub>a</sub> (fr)	随机闪存访问时间	30		ns
t <sub>a</sub> (OTP)	OTP 访问时间	30		ns

**表 6-66 闪存数据保持持续时间**

参数	测试条件	最小值	最大值	单位
t <sub>保持</sub>	数据保持持续时间	T <sub>J</sub> =55°C	15	年

**表 6-67 不同频率上所需最小的闪存/一次性可编程 (OTP) 等待状态**

SYSCLKOUT (MHz)	SYSCLKOUT (ns)	页等待状态 <sup>(1)</sup>	随机等待状态 <sup>(1)</sup>	OTP
150	6.67	5	5	8
120	8.33	4	4	7
100	10	3	3	5
75	13.33	2	2	4
50	20	1	1	2

30	33.33	1	1	1
25	40	1	1	1
15	66.67	1	1	1
4	250	1	1	1

(1) 随机等待状态必须大于或者等于 1，用于计算表 6-67 中页等待状态和随机等待状态的公式如下：

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

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

计算表 6-67 中 OTP 等待状态的公式如下：

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

## 7 热性能/机械数据

[表 7-1](#) 和 [表 7-2](#) 显示了散热数据。有关散热设计考虑的更多信息请见 [章节 6.4.3 散热设计考虑](#)。

表格之后的机械封装图反映了指定器件的机械数据。

**表 7-1 散热模型 176 引脚 QP 结果**

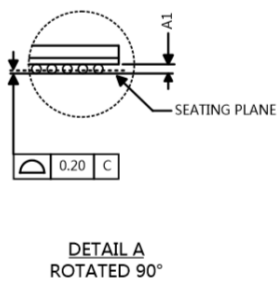
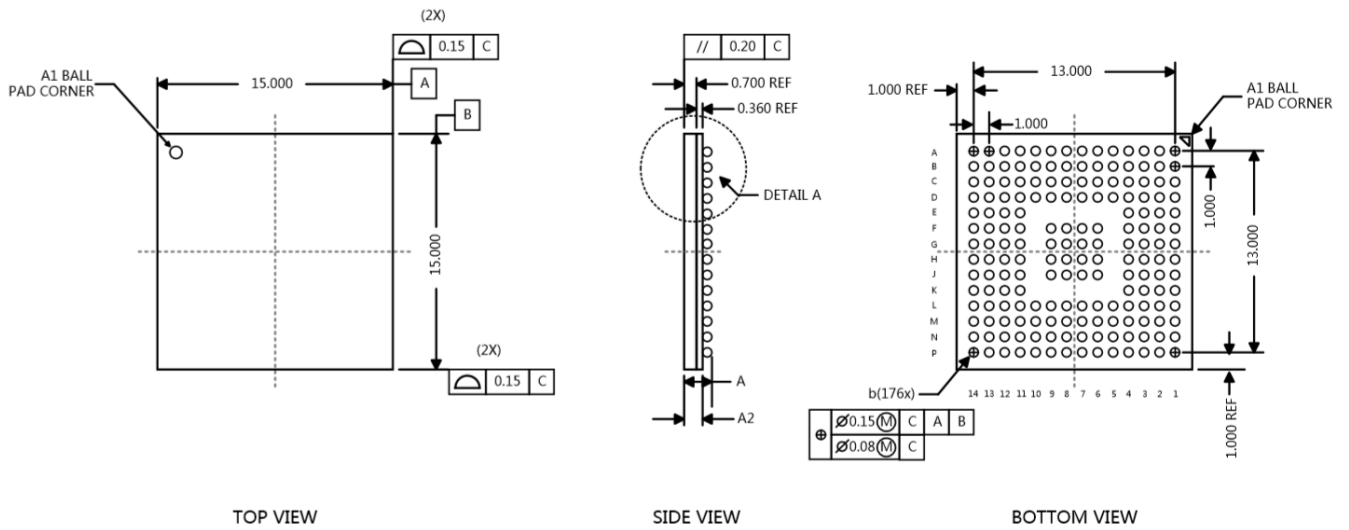
参数	0lfm	150lfm	250lfm	500lfm
$\theta_{JA}$ [°C/W] 高 k PCB	44	34.5	33	31
$\Psi_{JT}$ [°C/W]	0.12	0.48	0.57	0.74
$\Psi_{JB}$	28.1	26.3	25.9	25.2
$\theta_{JC}$	8.2			
$\theta_{JB}$	28.1			

**表 7-2 散热模型 176 引脚 BA 结果**

参数	0lfm	150lfm	250lfm	500lfm
$\theta_{JA}$ [°C/W] 高 k PCB	17.4	11.7	10.1	8.8
$\Psi_{JT}$ [°C/W]	0.2	0.3	0.4	0.5
$\Psi_{JB}$	5.0	4.7	4.7	4.6
$\theta_{JC}$	12.1			
$\theta_{JB}$	5.1			

BGA176

单位：mm

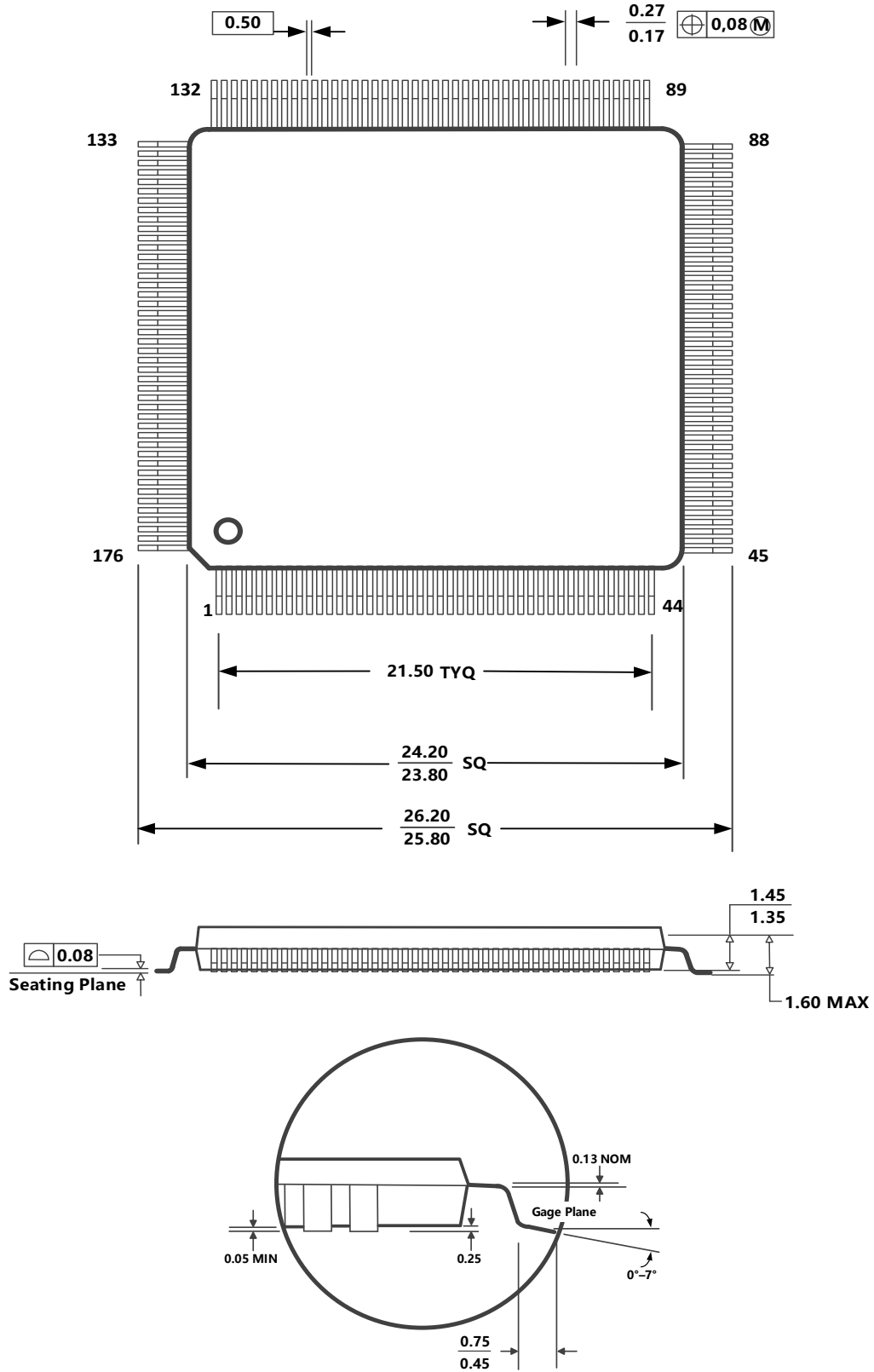


单位：MM

DIMENSION	MINIMUM	NOMINAL	MAXIMUM
A	1.448	1.560	1.672
A1	0.400	0.500	0.600
A2	1.010	1.060	1.110
b	0.500	0.600	0.700
NUMBER OF BALL 176			

LQFP176

单位 : mm



## 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

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