

Haawking_DSC28035_数字信号控制器

数据手册

V1. 6



北京中科昊芯科技有限公司

2025 年 03 月

历史版本记录

| 版本号 | 内容描述 | 时间 |
|--------|--------------------------|------------|
| V1.0.0 | 初版, 手册初版 | 2023.03.09 |
| V1.0.1 | 修正封装形式 | 2023.03.16 |
| V1.0.2 | ADC 参数修正 | 2023.03.29 |
| V1.0.3 | PWM8 添加 | 2023.04.13 |
| V1.0.4 | 更新文档 | 2023.05.02 |
| V1.0.5 | 更新文档封面和页眉以满足质量基线 | 2023.07.18 |
| V1.1 | 更新 6.5 功耗 | 2023.09.18 |
| V1.2 | 校对更新 | 2023.11.06 |
| V1.3 | 增加 56PIN 封装说明 | 2023.12.20 |
| V1.4 | HRPWM 最小捕获 7 个周期, 更新产品编码 | 2024.03.27 |
| V1.5 | 更新 VDD 内核电压, 订正错误, 优化格式 | 2024.12.15 |
| V1.6 | 公司联系信息变更 | 2025.3.18 |

目录

| | |
|--------------------------------------|----|
| 1. 器件概述 | 6 |
| 1. 1 产品特性 | 6 |
| 1. 2 应用 | 7 |
| 2. DSC28035 简介 | 8 |
| 2. 1 器件特性 | 8 |
| 2. 2 产品编码 | 9 |
| 2. 3 引脚分配 | 10 |
| 2. 4 信号说明 | 13 |
| 3. 功能概述 | 21 |
| 3. 1 系统架构 | 21 |
| 3. 2 内存映射 | 22 |
| 3. 3 简要说明 | 23 |
| 3. 3. 1 HXS320F28035 DSC | 23 |
| 3. 3. 2 内存总线 | 23 |
| 3. 3. 3 外设总线 | 23 |
| 3. 3. 4 实时 JTAG 和分析 | 24 |
| 3. 3. 5 闪存 | 24 |
| 3. 3. 6 M0、M1、L0、L1 SARAM | 24 |
| 3. 3. 7 引导 ROM | 25 |
| 3. 3. 7. 1 JTAG 引导模式 | 25 |
| 3. 3. 7. 2 GetMode 获取模式 | 25 |
| 3. 3. 7. 3 引导加载器使用的外设引脚 | 26 |
| 3. 3. 8 安全性 | 26 |
| 3. 3. 9 外设中断扩展（PIE）模块 | 27 |
| 3. 3. 10 外部中断（XINT1-XINT3） | 27 |
| 3. 3. 11 内部振荡器、外部振荡器和 PLL | 28 |
| 3. 3. 12 看门狗 | 28 |
| 3. 3. 13 外设时钟 | 28 |
| 3. 3. 14 低功耗模式 | 28 |
| 3. 3. 15 通用输入/输出（GPIO）复用器 | 29 |
| 3. 3. 16 32 位 CPU 定时器（0, 1, 2） | 29 |
| 3. 3. 17 控制外设 | 30 |
| 3. 3. 18 串行端口外设 | 30 |
| 3. 4 寄存器映射 | 31 |

| | |
|--|----|
| 3.5 器件仿真寄存器 | 32 |
| 3.6 中断系统 | 32 |
| 3.7 LDO/VREG/BOR/POR | 36 |
| 3.7.1 VREG | 36 |
| 3.7.1.1 使用片上 VREG | 36 |
| 3.7.1.2 不使用片上 VREG | 37 |
| 3.7.2 片上上电复位 (POR) 和掉电复位 (BOR) 电路 | 37 |
| 3.8 系统控制 | 38 |
| 3.8.1 内部零引脚振荡器 | 40 |
| 3.8.2 晶体振荡器选项 | 41 |
| 3.8.3 基于 PLL 的时钟模块 | 42 |
| 3.9 低功耗模块 | 42 |
| 4. 外设 | 44 |
| 4.1 模拟器件 | 44 |
| 4.1.1 特性 | 44 |
| 4.1.2 ADC MUX 多路复选 | 48 |
| 4.1.3 比较器模块 | 48 |
| 4.2 同步串行通信接口 (SPI) 模块 | 49 |
| 4.3 异步串行通信接口 (SCI) 模块 | 52 |
| 4.4 本地互联网络 (Lin) 模块 | 53 |
| 4.5 增强型控制器局域网络 (eCAN) 模块 | 56 |
| 4.6 内部集成电路接口 (I2C) 模块 | 59 |
| 4.7 增强型 PWM 模块 (ePWM1/2/3/4/5/6/7/8) | 62 |
| 4.8 高分辨率 PWM 模块 (HRPWM) | 64 |
| 4.9 增强型脉冲捕获模块 (eCAP) | 65 |
| 4.10 高分辨率脉冲捕获模块 (HRCAP) | 67 |
| 高分辨率捕捉 (HRCAP) 时序要求 | 68 |
| 4.11 增强型正交编码器模块 (eQEP) | 68 |
| 4.12 JTAG 端口 | 73 |
| 4.13 GPIO 控制寄存器 | 74 |
| 4.14 直接存储访问存储单元 (DMA) | 78 |
| 4.14.1 DMA | 78 |
| 4.15 可编程控制律加速器 (CLA) | 83 |
| 5. 开发支持 | 86 |
| 5.1 集成开发环境 Haawking IDE | 86 |
| 5.2 仿真器 | 87 |
| 5.2.1 简介 | 87 |

| | |
|-----------------------|----|
| 5.2.2 概览 | 87 |
| 5.2.2.1 特性 | 88 |
| 5.4 离线下载器 | 89 |
| 6. 电气规范 | 90 |
| 6.1 绝对最大额定值 | 90 |
| 6.2 推荐工作条件 | 90 |
| 6.3 ESD 额定值 | 91 |
| 6.4 电气特性 | 92 |
| 6.5 功耗 | 93 |
| 6.6 散热设计考虑 | 93 |
| 7. 封装外形尺寸 | 94 |
| 7.1 LQFP 80 PIN | 94 |
| 7.2 LQFP 64 PIN | 95 |
| 7.3 VQFN 56 PIN | 96 |
| 8. 订货信息 | 97 |
| 8.1 封装信息 | 97 |
| 8.2 托盘信息 | 97 |
| 8.3 包装盒尺寸 | 98 |
| 8.4 封装概览 | 98 |
| 关于中科昊芯 | 99 |

1. 器件概述

1.1 产品特性

高能效 32 位处理器 (H28x 内核)

- ✓ 主频 120MHz (周期 8.33ns)
- ✓ 哈佛 (Harvard) 总线架构
- ✓ 硬件乘法/除法单元
- ✓ 4/6 通道高速 DMA
- ✓ 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)

可编程控制律加器 (CLA)

- ✓ 32 位浮点数学加速器
- ✓ 独立于主 CPU 之外的代码执行

增强型控制外设

- ✓ 增强型脉宽调制器 (ePWM)
- ✓ 高分辨率 PWM (HRPWM)
- ✓ 增强型捕获 (eCAP)
- ✓ 高分辨率输入捕获 (HRCAP)
- ✓ 增强型正交编码器脉冲 (eQEP)
- ✓ 模数转换器 (ADC*2)
- ✓ 片载温度传感器
- ✓ 比较器

芯片及系统开销

- ✓ 3.3V 单电源供电
- ✓ 无上电顺序要求
- ✓ 集成上电和掉电复位
- ✓ 代码安全模块
- ✓ 无模拟支持引脚

时钟和系统控制

- ✓ 两个内部 12MHz 零引脚振荡器
- ✓ 片载振荡器/外部时钟输入

- ✓ 看门狗定时器模组功能
- ✓ 支持 PLL 倍频系数动态调整

多达 45 个具有输入滤波功能、可单独编程的多路复用 GPIO

- ✓ 可支持所有外设中断的外设中断扩展 (PIE) 模块
- ✓ 三个 32 位 CPU 定时器
- ✓ 每个 ePWM 模块中包含独立的 32 位定时器
- ✓ 片载存储器
- ✓ 闪存, SRAM, OTP, 引导 ROM 可用

128 位安全密钥/锁

- ✓ 保护安全内存块
- ✓ 防止固件逆向工程
- ✓ 串行端口外设
- ✓ 一个 SCI (UART) 模块
- ✓ 两个 SPI 模块
- ✓ 一个内部集成电路 (I2C) 总线
- ✓ 一个本地互连网络 (LIN) 总线
- ✓ 一个增强型控制器局域网络 (eCAN) 总线

高级仿真特性

- ✓ 分析和断点功能
- ✓ 封装选项
- ✓ 80 引脚薄型四方扁平 (LQFP) 封装
- ✓ 64 引脚薄型四方扁平 (LQFP) 封装
- ✓ 56 引脚极细四方扁平封装 (VQFN)

温度选项

- ✓ T: -40° 至 105°
- ✓ S: -40° 至 125°

1.2 应用

- 电器
- 楼宇自动化
- 电动汽车、混合动力电动汽车（EV/HEV）动力传动
- 工厂自动化
- 电网基础设施
- 医疗、保健与健身
- 电机驱动器
- 电力传送
- 电信基础设施
-

中科昊芯

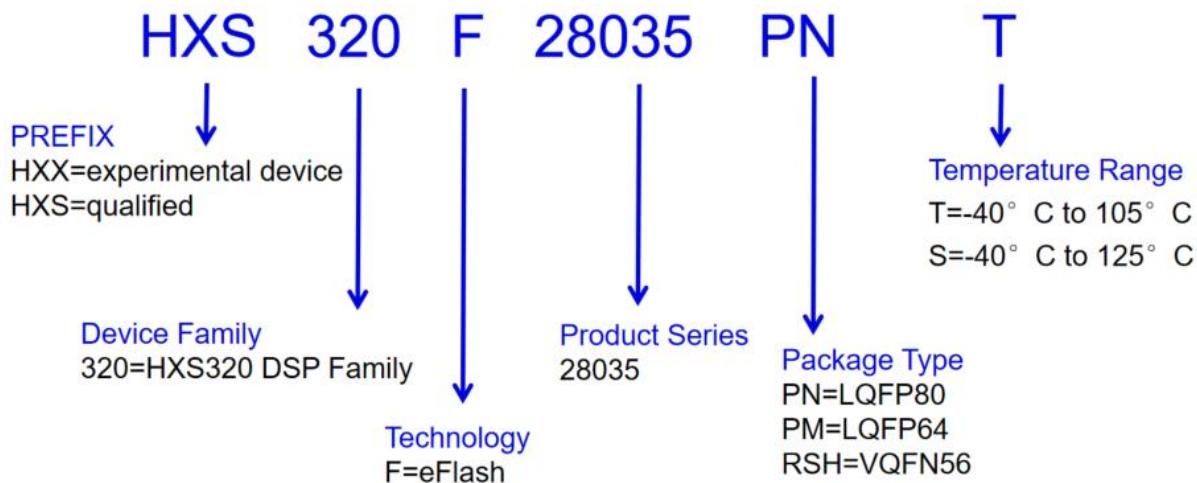
2. DSC28035 简介

2.1 器件特性

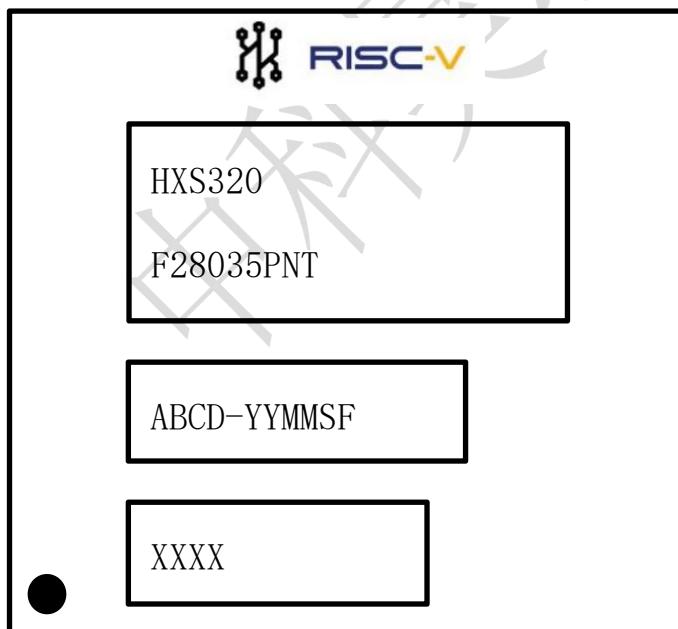
表 2-1 DSC28035 产品特性

| 功能 | HXS320F28035 | HXS320F28035 | HXS320F28035 |
|----------------|---------------|---------------|----------------|
| 封装类型 | 80 引脚 PN LQFP | 64 引脚 PM LQFP | 56 引脚 RSH VQFN |
| 时钟主频 (MHz) | 120 | 120 | 120 |
| 指令周期 (ns) | 8.33ns | 8.33ns | 8.33ns |
| 硬件乘法/除法单元 | 32-bit | 32-bit | 32-bit |
| 片内闪存 (Byte) | 256K | 256K | 256K |
| 片载 SRAM (Byte) | 36K | 36K | 36K |
| 代码安全 CSM 模块 | 支持 | 支持 | 支持 |
| 引导 ROM | 支持 | 支持 | 支持 |
| 一次可编程 OTP | 8K | 8K | 8K |
| 高速 DMA 模块通道 | 6 | 6 | 6 |
| ePWM 模块 | 8 | 6 | 4 |
| ePWM 通道 | 8×2 | 6×2 | 4×2 |
| eCAP 模块 | 1 | 1 | 1 |
| eQEP 模块 | 2 | 1 | 1 |
| 看门狗定时器 | 支持 | 支持 | 支持 |
| 12 位 ADC | MspS | 1.25MspS*2 | 1.25MspS*2 |
| | 转换时间 | 800ns | 800ns |
| | 通道 | 16 | 14 |
| | 温度传感器 | 支持 | 支持 |
| 32 位 CPU 定时器 | 3 | 3 | 3 |
| 高分辨率 HRPWM 模块 | 8 | 6 | 4 |
| 高分辨率捕获模块 | 2 | 2 | - |
| 集成 DAC 的比较器 | 3 | 3 | 3 |
| 内部集成电路 I2C | 1 | 1 | 1 |
| 增强型控制器局域网 eCAN | 1 | 1 | 1 |
| 本地互联网络 Lin | 1 | 1 | 1 |
| 串行外设接口 SPI | 2 | 1 | 1 |
| 串行通信接口 SCI | 1 | 1 | 1 |
| IO 引脚 | GPIO | 45 | 33 |
| | AI0 | 6 | 6 |
| 外部中断 | 3 | 3 | 3 |
| 电源电压 (标称值) | 3.3v | 3.3v | 3.3v |
| 温度选项 | -40° 至 105° | 支持 | 支持 |
| | -40° 至 125° | 支持 | 支持 |

2.2 产品编码



丝印 Marking 说明



ABCD-YYWWSF : ABC 表示硅片的版本
 YY 表示生产年份编号
 WW 表示生产周份编号

2.3 引脚分配

下图所示为 DSC28035 80-pin LQFP 封装，引脚分配示意图。

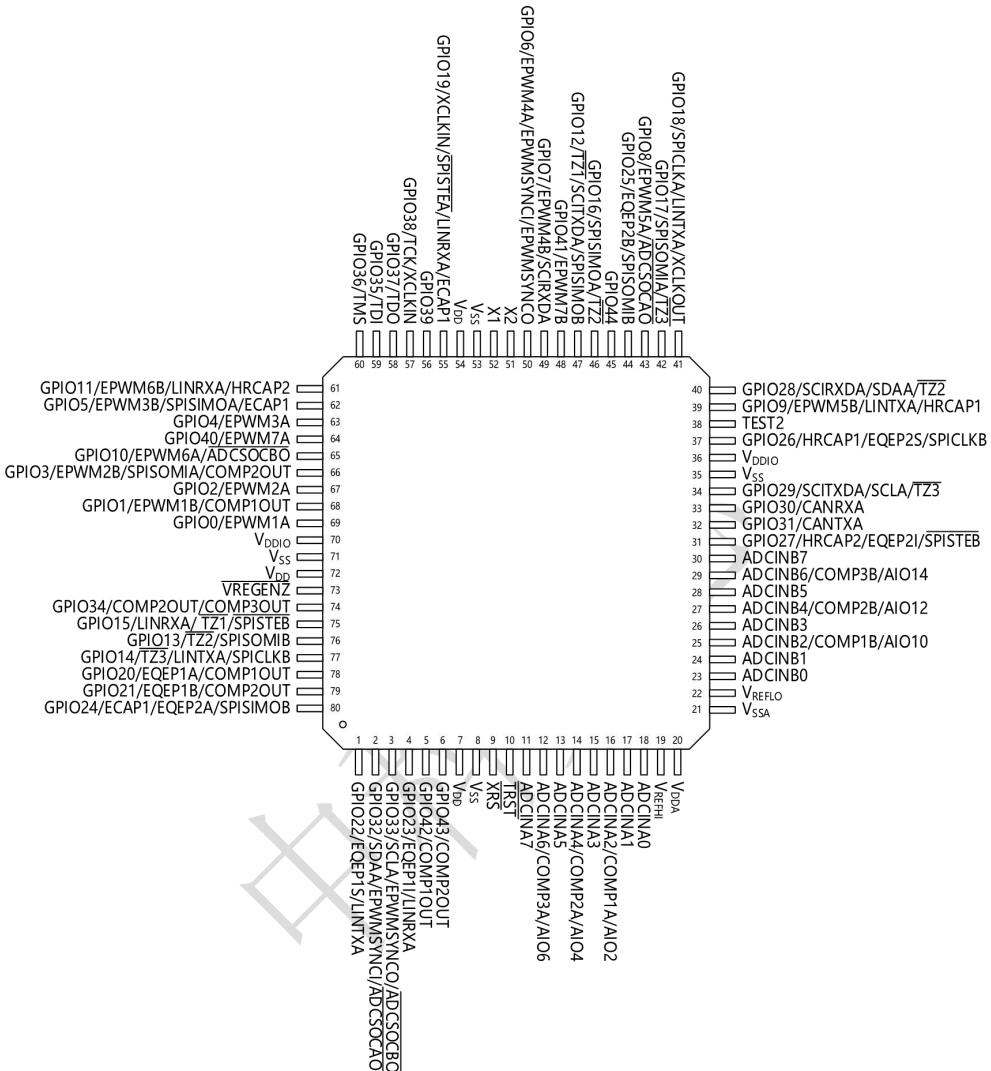
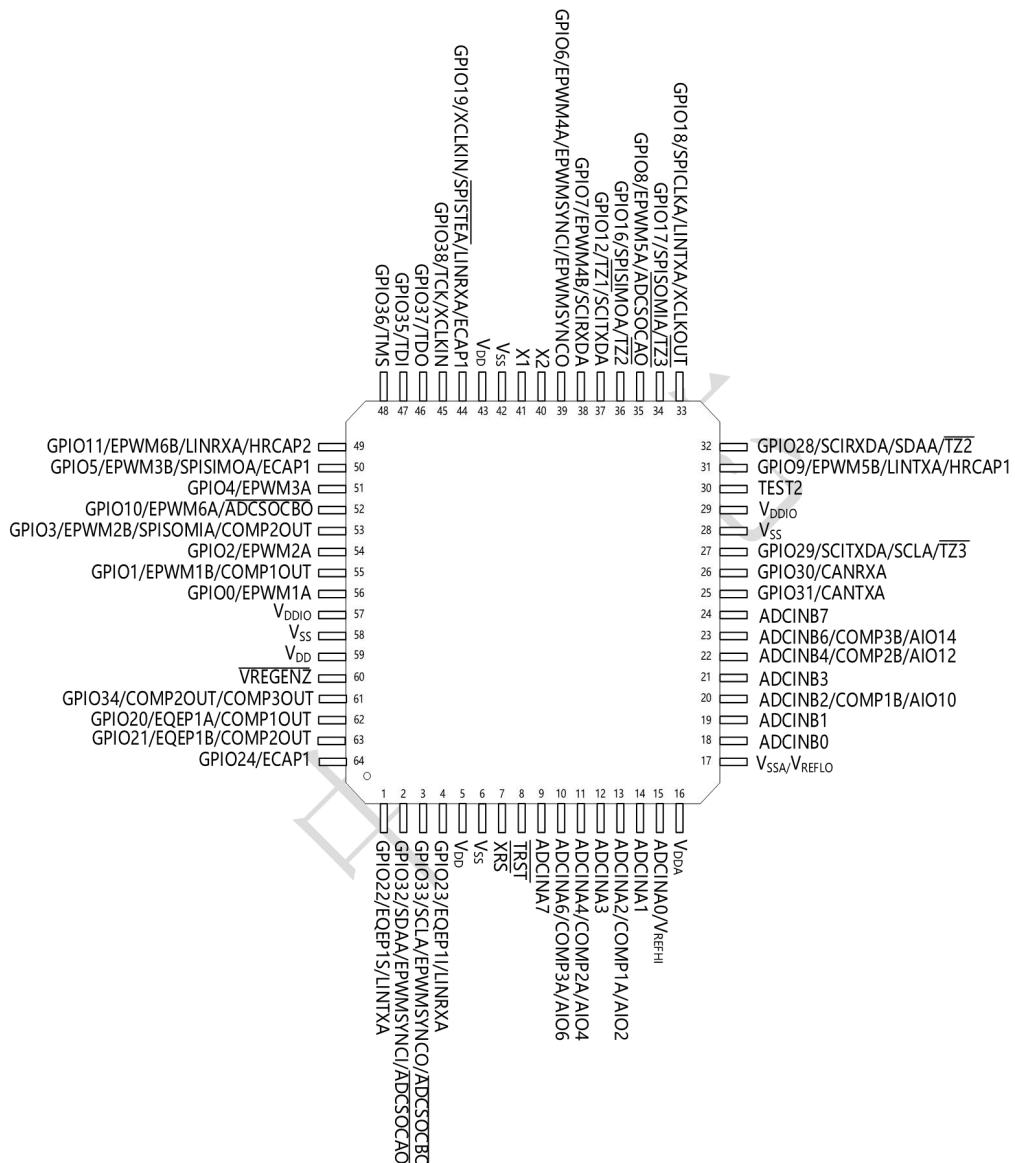


图 2-1 DSC28035 80 引脚 PN LQFP (顶视图)

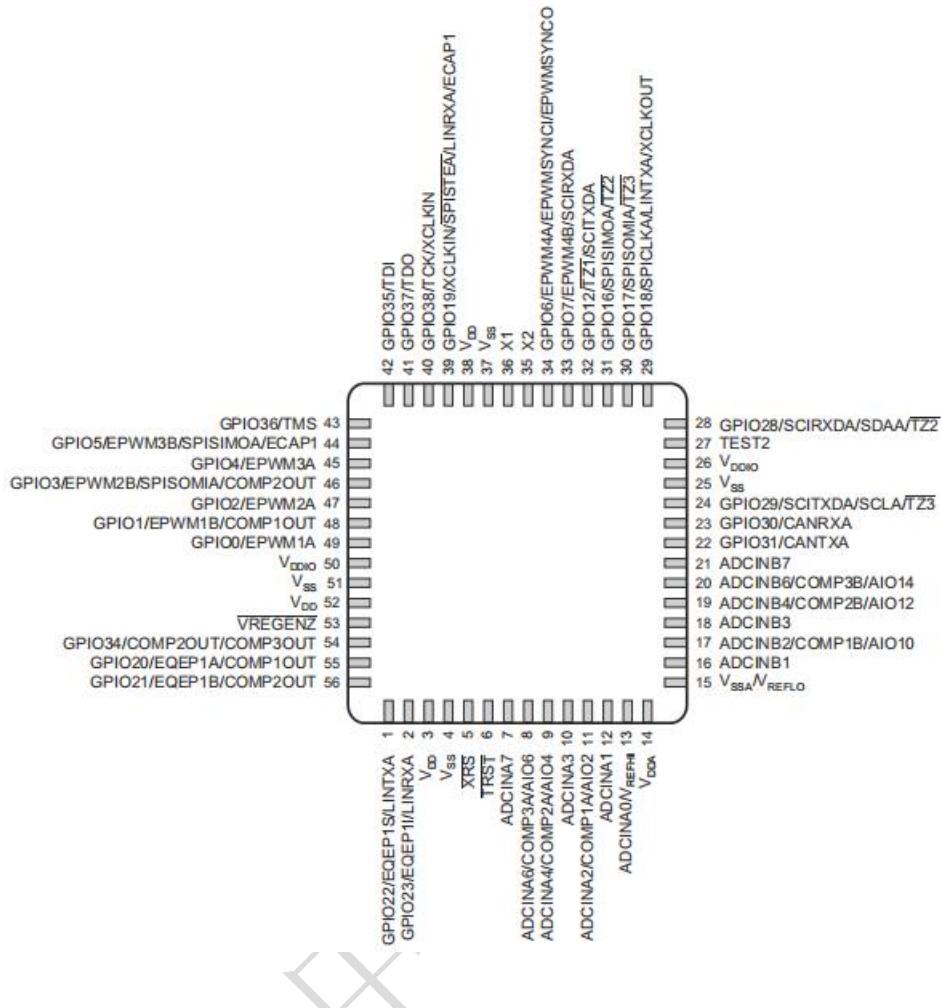
下图所示为 DSC28035 64-pin LQFP 封装，引脚分配示意图。



- A. 引脚 15:VREFHI 和 ADCINA0 在 64 引脚 PM 设备上共享相同的引脚，它们的使用是相互排斥的。
 - B. 引脚 17:VREFLO 总是连接到 64 引脚 PM 设备上的 VSSA。

图 2-2 DSC28035 64 引脚 PM LQFP (顶视图)

下图所示为 DSC28035 56-pin VQFN 封装，引脚分配示意图。



A. 引脚 13:VREFHI 和 ADCINA0 在 56 引脚 PM 设备上共享相同的引脚，它们的使用是相互排斥的。

B. 引脚 15:VREFLO 总是连接到 64 引脚 PM 设备上的 VSSA。

图 2- 3 DSC28035 56 引脚 RSH VQFN (顶视图)

2.4 信号说明

表 2-2 对这些信号进行了说明。除 JTAG 引脚以外，引脚的 GPIO 功能是复位时的默认功能，除非另外注明。下面列出的外设信号是供替换的功能。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。输入是 3.63V 耐压。所有 GPIO 引脚为 I/O/Z 且有一个内部上拉电阻器，此内部上拉电阻器可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。PWM 引脚上的上拉电阻在复位时并不启用。其它 GPIO 引脚上的上拉电阻复位时被启用。AI0 引脚没有内部上拉电阻。

表 2-2 信号说明

| 引脚名称 | 引脚编号 | | | I/O/Z | 描述 |
|--------------|----------|-------|-------|-------|---|
| | 80PIN | 64PIN | 56PIN | | |
| JTAG | | | | | |
| TRST | 10 | 8 | 6 | I | <p>JTAG 测试复位，内部下拉。 TRST 在高电平时进行扫描设备操作的系统控制。如果此信号未连接或驱动为低电平，设备在其功能模式下运行，测试复位信号被忽略。 注：TRST 是一个在高电平有效的测试引脚，在设备正常运行期间的任何时候必须为低电平。此引脚上需要一个外部下拉电阻。该电阻的值应该基于适用于设计的调试器盒的驱动长度。 2.2K 电阻通常可提供足够的保护。由于这是特定于应用程序的，因此昊芯建议验证每个目标板，以使调试器和应用程序正常运行</p> |
| TCK | 见 GPIO38 | | | I | 见 GPIO38。内部上拉的 JTAG 测试时钟 |
| TMS | 见 GPIO36 | | | I | 见 GPIO36。内部上拉的 JTAG 测试模式选择（TMS）。该串行控制输入在 TCK 的上升沿输入 TAP 控制器 |
| TDI | 见 GPIO35 | | | I | 见 GPIO35。内部上拉的 JTAG 测试数据输入（TDI）。TDI 在 TCK 的上升沿输入选定的寄存器（指令或数据） |
| TDO | 见 GPIO37 | | | O/Z | 见 GPIO37。JTAG 扫描输出，测试数据输出（TDO）。所选寄存器的内容（指令或数据）在 TCK 的下降沿移出 TDO |
| FLASH | | | | | |
| TEST2 | 38 | 30 | 27 | I/O | 测试引脚。为 HX 保留。必须保持未连接 |
| CLOCK | | | | | |
| XCLKOUT | 见 GPIO18 | | | - | <p>见 GPIO18. 来自 SYSCLKOUT 的输出时钟。 XCLKOUT 也有 SYSCLKOUT 相同的频率，二分频，四分频。这是由 XCLK 寄存器中的 1:0 (XCLKOUTDIV) 位控制的。复位后，XCLKOUT=SYSCLKOUT/4。XCLKOUT 信号可以通过设置 XCLKOUTDIV 为 3 来关闭。GPIO18 的复用控制也必须设置为 XCLKOUT，此信号才能传播到</p> |

| 引脚名称 | 引脚编号 | | | I/O/Z | 描述 |
|------------------------------|-------------------|-------|-------|---------------|--|
| | 80PIN | 64PIN | 56PIN | | |
| | | | | | 引脚。 |
| XCLKIN | 见 GPIO19 和 GPIO18 | | | I | <p>见 GPIO19 和 GPIO38. 外部振荡器输入。时钟的引脚源由 XCLK 寄存器中的 XCLKINSEL 位控制，GPIO38 是默认选择。该引脚从外部 3.3V 振荡器馈入时钟。在这种情况下，必须将 X1 引脚（如果可用）连接到 GND，并且必须通过 CLKCTL 寄存器中的第 14 位禁用片上晶体振荡器。如果使用了晶体 / 谐振器，则必须通过 CLKCTL 寄存器中的第 13 位禁用 XCLKIN 路径。</p> <p>注意：使用 GPIO38 / TCK / XCLKIN 引脚为正常器件操作提供外部时钟的设计可能需要结合一些挂钩，以在使用 JTAG 连接器进行调试的过程中禁用该路径。这是为了防止与 TCK 信号争用该信号在 JTAG 调试会话期间处于活动状态。在这段时间内，可以使用零引脚内部振荡器为器件提供时钟。</p> |
| X1 | 52 | 41 | 36 | I | <p>片上 1.8V 晶体振荡器输入。</p> <p>要使用此振荡器，必须在 X1 和 X2 之间连接一个石英晶体或一个陶瓷谐振器。在这种情况下，必须通过 CLKCTL 寄存器中的第 13 位禁用 XCLKIN 路径。如果不使用该引脚，则必须将其连接至 GND。</p> |
| X2 | 51 | 40 | 35 | 0 | <p>片上晶体振荡器输出。</p> <p>要使用此振荡器，必须在 X1 和 X2 之间连接一个石英晶体或一个陶瓷谐振器。如果 X2 没有使用，必须保持未连接。</p> |
| XRS | 9 | 7 | 5 | I/O | <p>设备复位和看门狗复位。昊芯设备具有内置的上电复位 (POR) 和掉电复位 (BOR)。</p> <p>在上电或掉电情况下，该引脚被器件驱动为低电平。外部电路也可以驱动该引脚以使器件复位。当看门狗复位发生时，该引脚也被 MCU 驱动为低电平。在看门狗复位期间，在 512 个 OSCCLK 周期的看门狗复位期间，XRS 引脚被驱动为低电平。在 XRS 和 V_{DDIO} 之间放置一个电阻值在 2.2K 和 10K 之间的电阻。如果在 XRS 和 V_{SS} 之间放置一个电容器以进行噪声过滤，则该电容器应为 100nF 或更小。这些值将使看门狗在断言看门狗复位后在 512 个 OSCCLK 周期内将 XRS 引脚正确驱动到 V_{OL}。不管源是什么，设备复位都会导致设备终止执行。程序计数器指向地址 0x7fb600 包含的地址。禁用复位后，将在程序计数器指定的位置开始执行。该引脚的输出缓冲器是带有内部上拉电阻的漏极开路器件。如果该引脚由外部设备驱动，则应使用漏极开路设备来完成。</p> |
| ADC, COMP ARATOR, ANALOG I/O | | | | | |
| ADCINA7 | 11 | 9 | 7 | I | ADC 组 A, 输入通道 7 |
| ADCINA6 COMP3A AI06 | 12 | 10 | 8 | I I I/O | ADC 组 A, 输入通道 6 比较器输入 3A 数字 AI06 |

| 引脚名称 | 引脚编号 | | | I/O/Z | 描述 |
|--------------------|-------|-------|-------|-------|---|
| | 80PIN | 64PIN | 56PIN | | |
| ADCINA5 | 13 | - | - | I | ADC 组 A, 输入通道 5 |
| ADCINA4 | | | | I | ADC 组 A, 输入通道 4 |
| COMP2A | 14 | 11 | 9 | I | 比较器输入 2A |
| AI04 | | | | I/O | 数字 AI04 |
| ADCINA3 | 15 | 12 | 10 | I | ADC 组 A, 输入通道 3 |
| ADCINA2 | | | | I | ADC 组 A, 通道 2 输入 |
| COMP1A | 16 | 13 | 11 | I | 比较器输入 1A |
| AI02 | | | | I/O | 数字 AI02 |
| ADCINA1 | 17 | 14 | 12 | I | ADC 组 A, 输入通道 1 |
| ADCINA0 | 18 | 15 | 13 | I | ADC 组 A, 输入通道 0 注: VREFHI 和 ADCINA0 在 64 引脚 PM 设备上共享相同的管脚, 它们的使用相互排斥。 注: VREFHI 和 ADCINA0 在 56 引脚 PM 设备上共享同一管脚, 它们的使用相互排斥。 |
| V _{REFHI} | 19 | 15 | 13 | I | DC 外部参考高仅在处于 ADC 外部参考模式时使用 注: VREFHI 和 ADCINA0 在 64 引脚 PM 设备上共享相同的引脚, 它们的使用相互排斥。 注: VREFHI 和 ADCINA0 在 56 引脚 PM 设备上共享同一引脚, 它们的使用相互排斥。 |
| ADCINB7 | 30 | 24 | 21 | I | ADC 组 B, 输入通道 7 |
| ADCINB6 | | | | I | ADC 组 B, 输入通道 6 |
| COMP3B | 29 | 23 | 20 | I | 比较器输入 3B |
| AI014 | | | | I/O | 数字 AI0 14 |
| ADCINB5 | 28 | - | - | I | ADC 组 B, 输入通道 5 |
| ADCINB4 | | | | I | ADC 组 B, 输入通道 4 |
| COMP2B | 27 | 22 | 19 | I | 比较器输入 2B |
| AI012 | | | | I/O | 数字 AI012 |
| ADCINB3 | 26 | 21 | 18 | I | ADC 组 B, 输入通道 3 |
| ADCINB2 | | | | I | ADC 组 B, 输入通道 2 |
| COMP1B | 25 | 20 | 17 | I | 比较器输入 1B |
| AI010 | | | | I/O | 数字 AI0 10 |
| ADCINB1 | 24 | 19 | 16 | I | ADC 组 B, 输入通道 1 |
| ADCINB0 | 23 | 18 | - | I | ADC 组 B, 输入通道 0 |
| V _{REFLO} | 22 | 17 | 15 | I | ADC 外部参考低 注: VREFLO 为始终连接到 64 引脚 PM 设备上和 56 引脚 RSH 设备上的 VSSA。 |
| CPU 和 I/O 电源 | | | | | |
| V _{DPA} | 20 | 16 | 14 | | 模拟电源引脚。与接近 pi 的 2.2Uf 电容器 (典型) 连接 |
| V _{SSA} | 21 | 17 | 15 | | 模拟接地引脚 注意: VREFLO 始终连接到 64 引脚 PM 设备和 56 引脚 RSH 设备上的 VSSA。 |
| V _{DD} | 7 | 5 | 3 | | CPU 和逻辑数字电源引脚。使用内部 VREG 时, 可在每个 VDD 引脚和接地高值电容器之间放置一个 1.2uf 电容器。 |
| | 54 | 43 | 38 | | |
| | 72 | 59 | 52 | | |
| V _{DDIO} | 36 | 29 | 26 | | 数字 I/O 缓冲器和闪存电源引脚。启用 VREG 时, 单 |

| 引脚名称 | 引脚编号 | | | I/O/Z | 描述 |
|---|-------|-------|-------|--------------------------|---|
| | 80PIN | 64PIN | 56PIN | | |
| | 70 | 57 | 50 | | 一电源。在每个引脚上放置一个解耦电容器。准确值应通过系统电压调节溶液确定。 |
| V_{SS} | 8 | 6 | 4 | | 数字接地引脚 |
| | 35 | 28 | 25 | | |
| | 53 | 42 | 37 | | |
| | 71 | 58 | 51 | | |
| 电压调节器控制信号 | | | | | |
| VREGENZ | 73 | 60 | 53 | I | 内部电压(VREG)通过内部下拉使能。直接连接到VSS(低)以使能内部1.8V VREG。直接连接到VDDIO(高)以禁用VREG，并使用外部1.57V电源。 |
| GPIO 和外部信号 | | | | | |
| GPI00 EPWM1A — — | 69 | 56 | 49 | I/O/Z 0 — — | 通用输入/输出0 增强型 PWM1 输出 A 和 HRPWM 通道- |
| GPI01 EPWM1B — COMP1OUT | | 55 | 48 | I/O/Z 0 — 0 | 通用输入/输出1 增强型 PWM1 输出 B — 比较器1的直接输出 |
| GPI02 EPWM2A — — | | 54 | 47 | I/O/Z 0 — — | 通用输入/输出2 增强的 PWM2 输出 A 和 HRPWM 通道 — — |
| GPI03 EPWM2B SPISOM1A COMP2OUT | 66 | 53 | 46 | I/O/Z 0 I/O 0 | 通用输入/输出3 增强型 PWM2 输出 B SPI—从机输出，主机输入 比较器2的直接输出 |
| GPI04 EPWM3A — — | | 51 | 45 | I/O/Z 0 — — | 通用输入/输出4 增强型 PWM3 输出 A 和 HRPWM 通道 — — |
| GPI05 EPWM3B SPISIMOA ECAP1 | | 50 | 44 | I/O/Z 0 I/O I/O | 一般输入/输出5 增强型 PWM3 输出 B SPI—从机输入，主机输出 增强捕获输入/输出1 |
| GPI06 EPWM4A EPWMSYNCI EPWMSYNCO | 50 | 39 | 34 | I/O/Z 0 I 0 | 通用输入/输出6 增强型 PWM4 输出 A 和 HRPWM 通道 外部 ePWM 同步脉冲输入 外部 Epwm 同步脉冲输出 |
| GPI07 EPWM4B SCIRXDA — | | 38 | 33 | I/O/Z 0 I — | 通用输入/输出7 增强型 PWM4 输出 B SCI-A 接收数据 — |

| 引脚名称 | 引脚编号 | | | I/O/Z | 描述 |
|--|-------|-------|-------|------------------------|--|
| | 80PIN | 64PIN | 56PIN | | |
| GPI08 EPWM5A — ADCSOC _A O | 43 | 35 | — | I/O/Z 0 0 | 通用输入/输出 8 增强型 PWM5 输出 A 和 HRPWM 通道 — ADC 开始转换 A |
| GPI09 EPWM5B LINTXA HRCAP1 | 39 | 31 | — | I/O/Z 0 0 I | 通用输入/输出 9 增强型 PWM5 输出 B LIN 传输 A 高分辨率输入捕获 1 |
| GPI010 EPWM6A — ADCSOC _B O | 65 | 52 | — | I/O/Z 0 0 | 通用输入/输出 10 增强型 PWM6 输出 A 和 HRPWM 通道 — ADC 开始转换 B |
| GPI011 EPWM6B LINRXA HRCAP2 | 61 | 49 | — | I/O/Z 0 I I | 通用输入/输出 11 增强型 PWM6 输出 B LIN 接收 A 高分辨率输入捕获 2 |
| GPI012 TZ1 SCITXDA SPISIMOB | 47 | 37 | 32 | I/O/Z I 0 I/O | 通用输入/输出 12 行程区域输入 1 SCI-A 传输数据 SPI-B 从机输入，主机输出。 注： SPI-B 仅在 PN 包中提供。 |
| GPI013 TZ2 — SPISOM1B | 76 | — | — | I/O/Z I I/O | 通用输入/输出 13 行程区域输入 2 — SPI-B 从机输出，主机输入 |
| GPI014 TZ3 LINTXA SPICLKB | 77 | — | — | I/O/Z I 0 I/O | 通用输入/输出 14 行程区域输入 3 LIN 传输 A SPI-B 时钟输入/输出 |
| GPI015 TZ1 LINRXA SPISTEB | 75 | — | — | I/O/Z I I I/O | 通用输入/输出 15 行程区域输入 1 LIN 接收 SPI-B 从机传输使能输入/输出 |
| GPI016 SPISIMO _A — TZ2 | 46 | 36 | 31 | I/O/Z I/O I | 通用输入/输出 16 SPI—从机输入，主机输出 — 行程区域输入 2 |
| GPI017 SPISOMIA — TZ3 | 42 | 34 | 30 | I/O/Z I/O I | 通用输入/输出 17 SPI—从机输出，主机输入 — 行程区域输入 3 |

| 引脚名称 | 引脚编号 | | | I/O/Z | 描述 |
|---|-------|-------|-------------------------------|------------------------------|---|
| | 80PIN | 64PIN | 56PIN | | |
| GPIO18 SPICLKA LINTXA XCLKOUT | 41 | 33 | 29 | I/O/Z I/O 0 0/Z | 通用输入/输出 18 SPI-A 时钟输入/输出 LIN 传输 由 SYSCLKOUT 导出的输出时钟。XCLKOUT 要么是和 SYSCLKOUT 相同的频率，要么是 SYSCLKOUT 频率的一半，要么是 SYSCLKOUT 频率的四分之一。这由 XCLK 寄存器中的 [1:0] XCLKOUTDIV 位控制。复位时，XCLKOUT=SYSCLKOUT/4。通过将 XCLKOUTDIV 设置为 3，可以关闭 XCLKOUT 信号。GPIO18 的 mux 控制也必须设置为 XCLKOUT，以便该信号传播到引脚。 |
| GPIO19 XCLKIN SPISTEA LINRXA ECAP1 | 55 | 44 | 39 | I/O/Z I/O I I/O | 通用输入/输出 19 外部振荡器输入从该引脚到时钟块的路径不由该引脚的 mux 功能选通。如果此路径用于其他外设功能，则必须小心不要启用此路径进行计时 SPI-A 从机传输使能输入/输出 LIN 接收 增强捕获输入/输出 1 |
| GPIO20 EQEP1A - COMP1OUT | 78 | 62 | 55 | I/O/Z I - 0 | 通用输入/输出 20 增强型 QEP1 输入 A - 比较器 1 的直接输出 |
| GPIO21 EQEP1B - COMP2OUT | 79 | 63 | 56 | I/O/Z I - 0 | 通用输入/输出 21 增强型 QEP1 输入 B - 比较器 2 的直接输出 |
| GPIO22 EQEP1S - LINTXA | 1 | 1 | 1 | I/O/Z I/O - 0 | 通用输入/输出 22 增强 QEP1 选通 - LIN 传输 |
| GPIO23 EQEP1I - LINRXA | 4 | 4 | 2 | I/O/Z I/O - I | 通用输入/输出 23 增强型 QEP1 指数 - LIN 接收 |
| GPIO24 ECAP1 EQEP2A SPISIMOB | 80 | 64 | See GPIO5 And GPIO19 | I/O/Z I/O I I/O | 通用输入/输出 24 增强捕获输入/输出 1 增强型 QEP2 输入 A SPI-B 从机输入，主机输出。 注：SPI-B 仅在 PN 和 RSH 包中提供。 |
| GPIO25 - EQEP2B SPISOMIB | 44 | - | - | I/O/Z I I/O | 通用输入/输出 25 - 增强型 QEP2 输入 B SPI-B 从机输出，主机输入 |

| 引脚名称 | 引脚编号 | | | I/O/Z | 描述 |
|--|-------|-------|-------|--------------------------|--|
| | 80PIN | 64PIN | 56PIN | | |
| GPI026 HRCAP1 EQEP2S SPICLKB | 37 | — | — | I/O/Z I I/O I/O | 通用输入/输出 26 高分辨率输入捕获 1 增强型 QEP2 选通 SPI-B 时钟输入/输出 |
| GPI027 HRCAP2 EQEP2I SPISTEB | 31 | — | — | I/O/Z I I/O I/O | 通用输入/输出 27 高分辨率输入捕获 2 增强型 QEP2 指数 SPI-B 从机传输使能输入/输出 |
| GPI028 SCIRXDA SDAA TZ2 | 40 | 32 | 28 | I/O/Z I I/OD I | 通用输入/输出 28 SCI 接收数据 I2C 数据双向开漏端口 行程区域输入 2 |
| GPI029 SCITXDA SCLA TZ3 | 34 | 27 | 24 | I/O/Z 0 I/OD I | 通用输入/输出 29 SCI 传输数据 I2C 时钟双向开漏端口 行程区域输入 3 |
| GPI030 CANRXA — — | 33 | 26 | 23 | I/O/Z I — — | 通用输入/输出 30 CAN 接收 — — |
| GPI031 CANTXA — — | 32 | 25 | 22 | I/O/Z 0 — — | 通用输入/输出 31 CAN 传输 — — |
| GPI032 SDAA EPWMSYNCI ADCSOCAO | 2 | 2 | — | I/O/Z I/OD I 0 | 通用输入/输出 32 I2C 数据双向开漏端口 增强型 PWM 外部同步脉冲输入 ADC 开始转换 A |
| GPI033 SCLA EPWMSYNCO ADCSOCBO | 3 | 3 | — | I/O/Z I/OD 0 0 | 通用输入/输出 33 I2C 时钟开漏双向端口 增强型 PWM 外部同步脉冲输出 ADC 开始转换 B |
| GPI034 COMP2OUT — COMP3OUT | 74 | 61 | 54 | I/O/Z 0 — 0 | 通用输入/输出 34 比较器 2 的直接输出 — 比较器 3 的直接输出 |
| GPI035 TDI | 59 | 47 | 42 | I/O/Z I | 通用输入/输出 35 内部上拉的 JTAG 测试数据输入 (TDI)。TDI 被计时到 TCLK 上升沿上的所选寄存器 (指令或数据) 中 |

| 引脚名称 | 引脚编号 | | | I/O/Z | 描述 |
|-------------------------------------|-------|-------|-------|-------|--|
| | 80PIN | 64PIN | 56PIN | | |
| GPIO36 TMS | 60 | 48 | 43 | I/O/Z | 通用输入/输出 36 |
| | | | | I | 内部上拉的 JTAG 测试模式选择 (TMS)。此串行控制输入被时钟输入 TCK 上升边缘的 TAP 控制器 |
| GPIO37 TDO | 58 | 46 | 41 | I/O/Z | 通用输入/输出 37 |
| | | | | O/Z | JTAG 扫描输出，测试数据输出 (TDO)。所选寄存器 (指令或数据) 的内容从 TCK (8mA 驱动器) 下降沿的 TDO 中移出 |
| GPIO38 TCK XCKIN — | 57 | 45 | 40 | I/O/Z | 通用输入/输出 38 |
| | | | | I | 内部上拉的 JTAG 测试时钟 |
| | | | | I | 外部振荡器输入。从该引脚到时钟块的路径不受该引脚的 mux 功能选通。如果此路径用于其他功能，则必须小心不要启用此路径进行计时。 |
| GPIO39 EPWM8A — — | 56 | — | — | I/O/Z | 通用输入/输出 39 增强型 PWM8 输出 A — — |
| | | | | — | |
| | | | | — | |
| GPIO40 EPWM7A — — | 64 | — | — | I/O/Z | 通用输入/输出 40 增强型 PWM7 输出 A 和 HRPWM 通道 — — |
| | | | | 0 | |
| | | | | — | |
| GPIO41 EPWM7B — — | 48 | — | — | I/O/Z | 通用输入/输出 41 增强型 PWM7 输出 B — — |
| | | | | 0 | |
| | | | | — | |
| GPIO42 — — COMP1OUT | 5 | — | — | I/O/Z | 通用输入/输出 42 — — 比较器 1 的直接输出 |
| | | | | 0 | |
| | | | | — | |
| GPIO43 — — COMP2OUT | 6 | — | — | I/O/Z | 通用输入/输出 43 — — 比较器 2 的直接输出 |
| | | | | 0 | |
| | | | | — | |
| GPIO44 EPWM8B — — | 45 | — | — | I/O/Z | 通用输入/输出 44 增强型 PWM8 输出 B — — |
| | | | | — | |
| | | | | — | |

3. 功能概述

3.1 系统架构

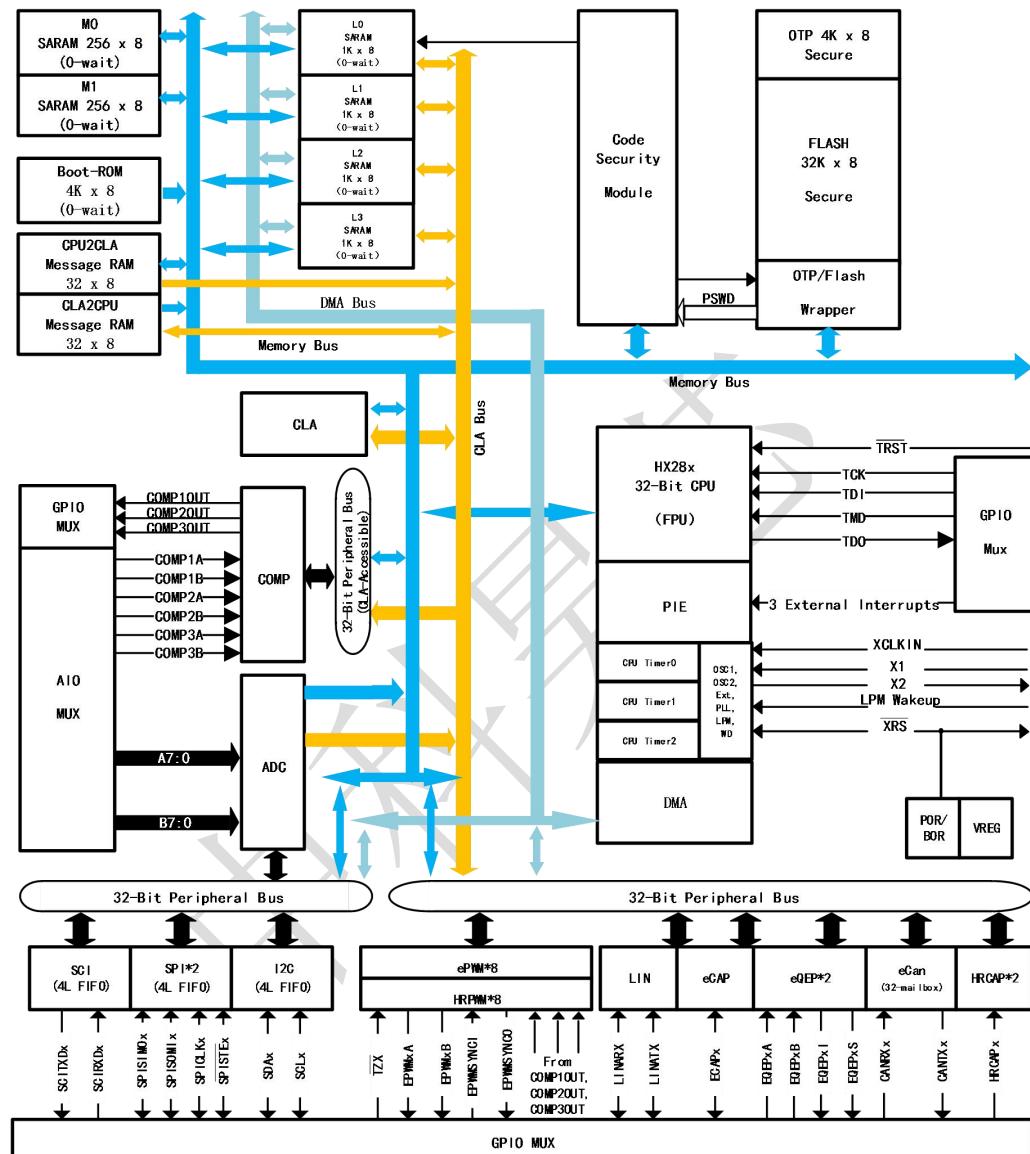


图 3-1 HXS320F28035 芯片架构图

3.2 内存映射

表 3-1 Peripheral frame 0 模块内存映射

| 序号 | 设备 | 地址范围（不含最高地址） | 大小 (Bytes) |
|-----|--------------------------------|---------------------|------------|
| 1. | M0 | 0x00 0000~0x00 0800 | 2K |
| 2. | M1 | 0x00 0800~0x00 1000 | 2K |
| | Res. | 0x00 1000~0x00 1400 | |
| 3. | ADC | 0x00 1400~0x00 1800 | 1K |
| 4. | Timer/PIE | 0x00 1800~0x00 1C00 | 1K |
| 5. | DMA | 0x00 1C00~0x00 2000 | 1K |
| 6. | CLA2CPU | 0x00 2000~0x00 2100 | 256 |
| 7. | CPU2CLA | 0x00 2100~0x00 2200 | 256 |
| | Res. | 0x00 2000~0x00 9000 | |
| 8. | CAN | 0x00 9000~0x00 9400 | 1K |
| 9. | L0 | 0x01 0000~0x01 2000 | 8K |
| 10. | L1 | 0x01 2000~0x01 4000 | 8K |
| 11. | L2 | 0x01 4000~0x01 6000 | 8K |
| 12. | L3 | 0x01 6000~0x01 8000 | 8K |
| | Res. | 0x01 8000~0x70 0000 | |
| 13. | Flash OTP Flash Register | 0x70 0000~0x74 0000 | 256K |
| | | 0x7A 0000~0x7A 8000 | 32K |
| | | 0x7A F800~0x7A F900 | 256 |
| | Res. | 0x7F 4000~0x7F C000 | |
| 14. | Bootrom | 0x7F 8000~0x80 0000 | 32K |

表 3-2 peripheral frame 1 模块内存映射

| 序号 | 设备 | 地址范围（不含最高地址） | 大小 (Bytes) |
|----|---------------|---------------------|------------|
| 1. | Debug | 0x00 A000~0x00 B000 | 4K |
| 2. | PWMx8 | 0x00 B000~0x00 B800 | 2K |
| | Res. | 0x00 B800~0x00 C000 | |
| 3. | Comparatormx3 | 0x00 C000~0x00 C400 | 1K |
| 4. | LIN | 0x00 C400~0x00 C800 | 1K |
| | Res. | 0x00 C800~0x00 CC00 | |
| 5. | HRCAPx2 | 0x00 CC00~0x00 D000 | 1K |
| 6. | CAP | 0x00 D000~0x00 D400 | 1K |
| 7. | QEPx2 | 0x00 D400~0x00 D800 | 1K |
| 8. | GPIO | 0x00 D800~0x00 DC00 | 1K |
| 9. | SYSCtrl | 0x00 DC00~0x00 E000 | 1K |

表 3-3 peripheral frame 2 模块内存映射

| 序号 | 设备 | 地址范围（不含最高地址） | 大小（Bytes） |
|----|----------|---------------------|-----------|
| 1. | SCI | 0x00 E000~0x00 E400 | 1K |
| 2. | I2C | 0x00 E400~0x00 E800 | 1K |
| 3. | SPIx2 | 0x00 E800~0x00 EC00 | 1K |
| | Reserved | 0x00 EC00~0x01 0000 | |

3.3 简要说明

3.3.1 HXS320F28035 DSC

HXS320F28035 DSC 系列是中科昊芯定点 DSC 平台上的最新产品，基于 RISC-V 开放指令集架构 H28x。它是一款非常高效的 C/C++ 引擎，此引擎不但能够让用户能够用高级语言开发他们的控制系统软件，还能够使用 C/C++ 开发数学算法。此器件在处理 DSP 算术任务时与处理系统控制任务时同样有效，而系统控制任务通常由微控制器器件处理。这样的效率在很多系统中省却了对第二个处理器的需要。

3.3.2 内存总线

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 之间移动数据。HXS320F28035 内存总线架构包含一个程序总线、数据总线和 DMA 总线。程序总线由 32 条地址线路和 32 条数据线路组成。多总线结构，通常称为哈佛总线，使得 HXS320F28035 能够在一个单周期内取一个指令、读写一个数据值。对所有连接在内存总线上的外设和内存访问时具有一定的优先级。总的来说，内存总线访问的优先级可概括如下：

- 1、数据总线具有最高优先级；
- 2、程序总线和 DMA 总线具有较低的平等优先级。

3.3.3 外设总线

为了实现不同中科昊芯 DSP 系列器件间的外设迁移，HXS320F28035 器件采用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。

3.3.4 实时 JTAG 和分析

HXS320F28035 器件使用标准的 IEEE1149.1 JTAG 接口。此外，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、外设寄存器内容。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 内硬件实现实时模式。这是 HXS320F28035 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断点事件。

这些设备不支持边界扫描；但是，如果有 IDCODE 和旁路功能，则可用考虑到以下因素。默认情况下 IDCODE 不提供。用户必须经过一系列的移位 IR 和移位 JTAG 的 DR 状态以获得 IDCODE。用于旁路指令，第一个移位的 DR 值将为 1。

3.3.5 闪存

HXS320F28035 器件包含 256KB 的嵌入式闪存存储器，其地址范围为 0x70 0000–0x74 0000，被分别放置在 128 个 2KB 扇区内。该器件还包含一个 8KB OTP 内存，其地址范围为 0x7A 0000–0x7A 2000。用户能够在不改变其它扇区的同时单独擦除、编辑和验证一个闪存扇区。然而，不能使用闪存的一个扇区或者这个 OTP 来执行擦除/编辑其它扇区的闪存算法。

HXS320F28035 器件提供了特殊内存流水线操作以使闪存模块实现更高的性能。闪存/OTP 被映射到程序和数据空间；因此它可被用于执行代码或者存储数据信息。

地址 0x73 FFF0–0x73 FFFF 为数据变量保留且不能包含程序代码，此处为密码区。

3.3.6 M0、M1、L0、L1 SARAM

所有器件包含这四块单周期访问内存 M0、M1、L0、L1。M0 和 M1 块与所有其它 HXS320F28035 器件上的内存块一样被映射到程序和数据空间。因此，用户能够使用 M0 和 M1 来执行代码或者存储数据变量。分区在连接器内执行。HXS320F28035 器件提供了一个到编程器的统一内存映射。这使得用高级语言编程变得更加容易。

HXS320F28035 内置四个 SARAM 存储器：

- M0: SARAM 2K x 8 bit (0-wait): 单端 SARAM, 2Kx8bit 大小, 映射到程序和数据地址空间, 既可用于执行程序又可用于存储器数据。
- M1: SARAM 2K x 8 bit (0-wait): 单端 SARAM, 2Kx8bit 大小, 映射到程序和数据地址空间, 既可用于执行程序又可用于存储器数据。
- SARAM (L0) : 16K x 8 bit Secure (0-wait): SARAM, 同样映射到程序和数据存储器。
- SARAM (L1) : 16K x 8 bit (0-wait): SARAM, 同样映射到程序和数据存储器。

3.3.7 引导 ROM

引导 ROM 由厂家使用引导载入软件进行设定。引导 ROM 程序在器件复位并检查了几个 GPIO 引脚后执行以确定应该进入哪一个引导模式。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软件下载至内部 RAM。还有其它的引导模式。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。

表格 3-4 引导模式选择

| 模式 | GPIO37/TDO | GPIO34/COMP2OUT/COMP3OUT | TRSTn | 模式 |
|------|------------|--------------------------|-------|------------------|
| 3 | 1 | 1 | 0 | 取模式 (GetMode) |
| 2 | 1 | 0 | 0 | 等待模式 |
| 1 | 0 | 1 | 0 | SCI 模式 |
| 0 | 0 | 0 | 0 | FLASH 启动 |
| JTAG | x | x | 1 | JTAG 模式 |

3.3.7.1 JTAG 引导模式

当仿真器被连接时, GPIO37/TDO 引脚不能被用于引导模式选择。在这种情况下, 引导 ROM 检测一个被连接的仿真器并使用 PIE 矢量表中两个被保留的 SARAM 位置内的内容来确定引导模式。如果两个位置内的内容均无效, 那么使用等待引导选项。可在仿真引导中访问所有引导模式选项。

3.3.7.2 GetMode 获取模式

GetMode 的默认运行状态选项为引导至闪存。通过在 OTP 中设定两个位置, 这个运行状态能够被改变为其它的引导选项。如果两个 OTP 位置的内容均为无效, 那么引导至闪存。可指定下列加载器中的一个: SCI、SPI、I2C、CAN 或者 OTP。

3.3.7.3 引导加载器使用的外设引脚

表格 3-5 显示了每一个外设引导加载器所使用的 GPIO 引脚。

表格 3-5 外设引导加载引脚使用情况

| 引导加载器 | 外设加载器引脚 |
|-------|--|
| SCI | SCIRXDA (GPIO28) SCITXDA (GPIO29) |
| 并行引导 | 数据(GPIO31, 30, 5:0) HXS320F28035 控制(AI06) 主机控制(AI012) |
| SPI | SPISIMOA (GPIO16) SPISOMIA (GPIO17) SPICLKA (GPIO18) SPISTEA (GPIO19) |
| IIC | SDAA (GPIO28) SCLA (GPIO29) |
| CAN | CANRXA (GPIO30) CANTXA (GPIO31) |

3.3.8 安全性

此器件支持高级安全性以保护用户固件不受反向工程的损坏。这个安全性特有一个 128 位密码（针对 16 个等待状态的硬编码），此密码由用户编辑入闪存。一个代码安全模块(CSM)用于保护闪存/OTP 和 LOSARAM 块。这个安全特性防止未经授权的用户通过 JTAG 端口检查内存内容，从外部内存执行代码或者试图引导加载一些将会输出安全内存内容的恶意软件。为了启用到安全块的访问，用户必须写入与存储在闪存密码位置内的值相匹配的正确的 128 位 KEY（密钥）值。除了 CSM，仿真代码安全逻辑电路(ECSL)也已经被实现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 内存的代码或者数据访问将触发 ECSL 错误并返回 0 值。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 位密码的读

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。当使用闪存内设定的密码位置开始调试一个器件时（也就是说，调试是安全的），仿真器需一些时间来控制 CPU。在此期间，CPU 将开始运行，并可能执行一个访问 受保护 ECSL 区的指令。如果这一情况发生，ECSL 将发生错误并使仿真器读取返回 0。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：

此处有关于 CSM 当代码安全密码被编辑时，0x73FFF0 到 0x73FFFC 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 0xFFFFFFFF。

地址 0x73FFF0 – 0x73FFFC 为数据保留且不能包含程序代码。128 位密码（位于 0x73FFF0 – 0x73FFFF）不能被设定为全 0，否则将永久锁住此器件。

3.3.9 外设中断扩展 (PIE) 模块

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 HXS320F28035 中，96 个中断中的 67 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。

3.3.10 外部中断 (XINT1–XINT3)

器件 HXS320F28035 支持三个屏蔽的外部中断 (XINT1 – XINT3)。选择用于下降，上升或下降和上升边沿触发，也可以选择使能/禁用这些中断，它还包含一个 16 位自由运行计数器，该计数器重置为零，可用于为中断准确地打上时间戳。没有用于外部中断的专用引脚，XINT1，XINT2 和 XINT3 中断可以配置为 GPIO0 – GPIO31 引脚的输入。

3.3.11 内部振荡器、外部振荡器和 PLL

该器件可以由两个内部零引脚振荡器，一个外部振荡器或一个晶体连接到片上振荡器电路，并提供了一个 PLL，最多可支持 12 个输入时钟缩放比例，PLL 比例可以在软件中即时更改，从而使用户能够如果需要低功率运行，则按比例缩小工作频率，可以在旁路模式下设置 PLL 模块。

3.3.12 看门狗

每个 HXS320F28035 器件包含一个看门狗电路——CPU 看门狗监视内核。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。

3.3.13 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或禁用以减少功耗。此外，到串行端口（除了 I2C 之外）和 ADC 时钟的系统时钟可相对于 CPU 时钟进行缩放。这样可去除外设时序到逐渐增加的 CPU 时钟速度的耦合。

3.3.14 低功耗模式

HXS320F28035 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:** 该模式会关断器件并将器件置于尽可能低的功耗模式中。HALT 模式在内部零引脚振荡器被用作时钟源时默认关闭。为了防止这些振荡器被关闭，可配置 CLKCTL 寄存器内的

INTOSCnHALTI 位。这样零引脚振荡器可在这个模式下被用于为 CPU 看门狗计时。如果片载晶体振荡器被用作时钟源，在这个模式中它将被关闭。在此模式下可由一个复位信号或者外部信号（通过一个 GPIO 引脚）以及 CPU 看门狗唤醒。

在试图将器件置于 HALT 或者 STANDBY 模式前，CPU 时钟(OSCCLK)和 WDCLK 信号应来自同一个时钟源。

3.3.15 通用输入/输出 (GPIO) 复用器

HXS320F28035 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。

3.3.16 32 位 CPU 定时器 (0,1,2)

CPU 定时器 0、1 和 2 是完全一样的 32 位定时器，这些定时器带有可预先设定的周期和 16 位时钟预分频。此定时器有一个 32 位倒计数寄存器，此寄存器在计数器达到 0 时生成一个中断。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 0 为通用定时器，并连接到 PIE 模块。CPU 定时器 1 同样为通用定时器，并可以被连接到 CPU 的 INT13。CPU 定时器 2 为 DSC 预留，并且连接到 CPU 的 INT14。如果 DSC 未被使用，CPU 定时器 2 也可作为通用定时器使用。

CPU 定时器 2 可由下列任何一种方式进行计时：

- SYSCLKOUT (默认)
- 内部零引脚振荡器 1 (INTOSC1)
- 内部零引脚振荡器 2 (INTOSC2)
- 外部时钟源

3.3.17 控制外设

HXS320F28035 支持以下用于嵌入式控制和通信的外设：

- **ePWM:** 增强型 PWM 外设支持针对前缘/后缘边沿、被锁存的/逐周期机制的独立的/互补的 PWM 生成，可调节死区生成。一些 PWM 引脚支持 HRPWM 高分辨率占空比和周期特性。HXS320F28035 器件上的类型 1 模块也支持增强型片上系统(SOC) 和中断生成和包括基于比较器输出的触发功能的高级触发。
- **eCAP:** 此增强型捕获外设使用一个 32 位时基并在连续/单次捕获模式中记录多达四个可编程事件。这个外设也被可被配置为生成一个辅助 PWM 信号。
- **eQEP:** 增强型 QEP 外设使用一个 32 位位置计数器，使用捕获单元和一个 32 位单元定时器分别支持低速测量和高速测量。这个外设有一个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。
- **ADC:** ADC 模块是一个 12 位转换器。根据器件的不同它有多达 16 个输入引脚。它包含两个用于同步采样的采样保持单元。
- **比较器:** 每个比较器块由一个模拟比较器连同一个为比较器的输入供电的内部 10 位基准组成。
- **HRCAP:** 这个高分辨率捕捉外设通过一个为 HCCAPCLK 计时的 16 位计数器运行在正常捕捉模式或者通过采用一个与校准库协同工作的内置校准逻辑运行在高性能捕获模式下。

3.3.18 串行端口外设

HXS320F28035 器件支持下列的串行通信外设：

- **SPI:** SPI 是一个高速、同步串行 I/O 端口，此端口可在设定的位传输速率上将一个设定长度（1 至 16 位）的串行比特流移入和移出器件。SPI 常用于 MCU 和外部外设或者其它处理器之间的通信。典型应用包括外部 I/O 或者从诸如移位寄存器、显示驱动器和 ADC 等器件的外设扩展。多器件通信由 SPI 主控/受控操作支持。SPI 包含一个用于减少中断处理开销的 4 级接收和发送 FIFO。
- **SCI:** 串行通信接口是一个两线制异步串行端口，通常被称为 UART。SCI 包含一个用于减少中断处理开销的 4 级接收和发送 FIFO。

- I2C:** 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 4 级接收和发送 FIFO。
- eCAN:** 这是 CAN 外设的增强型版本。它支持 32 个邮箱、消息时间戳并与 CAN 2.0B 兼容。
- LIN:** LIN 1.3 或者 2.0 兼容外设可被配置为额外的 SCI 端口。

3.4 寄存器映射

表 3-6 HXS320F28035 寄存器映射表

| 名称 | 地址范围 | 大小 (字节) | EALLOW 保护 |
|---------------------|-------------------|---------|----------------|
| Flash 控制寄存器 | 0x7AF800—0x7AF8FF | 256 | 是 |
| CSM 寄存器 | 0x7AF800—0x7AF813 | 20 | 是 |
| ADC 寄存器 | 0x1400—0x1498 | 152 | 是 ^① |
| ADC 结果寄存器 (只读) | 0x149C—0x14D8 | 60 | 否 |
| PIE 和 CPU 定时器 0/1/2 | 0x1800—0x1BFF | 1024 | 否 ^② |
| DMA 寄存器 | 0x1C00—0x1FFF | 1024 | 是 |
| eCAN-A 寄存器 | 0x9000—0x93FF | 1024 | 是 ^① |
| ePWM1+HRPWM1 | 0xB000—0xB0FF | 256 | 是 ^① |
| ePWM2+HRPWM2 | 0xB100—0xB1FF | 256 | 是 ^① |
| ePWM3+HRPWM3 | 0xB200—0xB2FF | 256 | 是 ^① |
| ePWM4+HRPWM4 | 0xB300—0x3FFF | 256 | 是 ^① |
| ePWM5+HRPWM5 | 0xB400—0xB4FF | 256 | 是 ^① |
| ePWM6+HRPWM6 | 0xB500—0xB5FF | 256 | 是 ^① |
| ePWM7+HRPWM7 | 0xB600—0xB6FF | 256 | 是 ^① |
| 比较器 1 | 0xC000—0xC07F | 128 | 是 ^① |
| 比较器 2 | 0xC080—0xC0FF | 128 | 是 ^① |
| 比较器 3 | 0xC100—0xC17F | 128 | 是 ^① |
| LIN-A 寄存器 | 0xC400—0xC7FF | 1024 | 是 ^① |
| HRCAP1 | 0xCC00—0xCC7F | 128 | 是 ^① |
| HRCAP2 | 0xCC80—0xCCFF | 128 | 是 ^① |
| eCAP1 寄存器 | 0xD000—0xD3FF | 1024 | 否 |
| eQEP1 寄存器 | 0xD400—0xD47F | 128 | 否 |
| eQEP2 寄存器 | 0xD480—0xD800 | 896 | 否 |
| GPIO 寄存器 | 0xD800—0xDBFF | 1024 | 是 ^① |
| 系统控制寄存器 | 0xDC00—0xDFF | 1024 | 是 |
| SCI-A 寄存器 | 0xE000—0xE3FF | 1024 | 否 |
| IIC-A 寄存器 | 0xE400—0xE7FF | 1024 | 否 |
| SPI-A 寄存器 | 0xE800—0xE87F | 128 | 否 |
| SPI-B 寄存器 | 0xE880—0xEBFF | 896 | 否 |

注：

- ① 此寄存器为部分受 EALLOW 保护，即给部分位赋值需要提前开启 EALLOW。
- ② CPU 和 PIE 整体的存储空间为 0x1800~0x1BFF 1K，其中只有 PIE 的中断向量入口地址是受 EALLOW 保护的。

说明：

1. 如果寄存器受 EALLOW 保护，则在执行 EALLOW 指令之前，无法执行写操作。EDIS 指令禁用写操作以防止杂散代码或指针损坏寄存器内容。
2. 闪存寄存器也受到代码安全模块（CSM）的保护。

3.5 器件仿真寄存器

表 3-7 器件仿真寄存器映射

| 名称 | 地址范围 | 大小 | 说明 |
|-----------|----------|------|---------|
| DEVICECNF | 0xDC80 | 4 字节 | 设备配置寄存器 |
| PARTID | 0x7A3BFC | 4 字节 | 芯片型号 |
| CLASSID | 0xDC84 | 4 字节 | 芯片系列号 |
| REVID | 0xDC88 | 4 字节 | 芯片版本号 |

3.6 中断系统

下图显示了各种中断源是如何复用的。

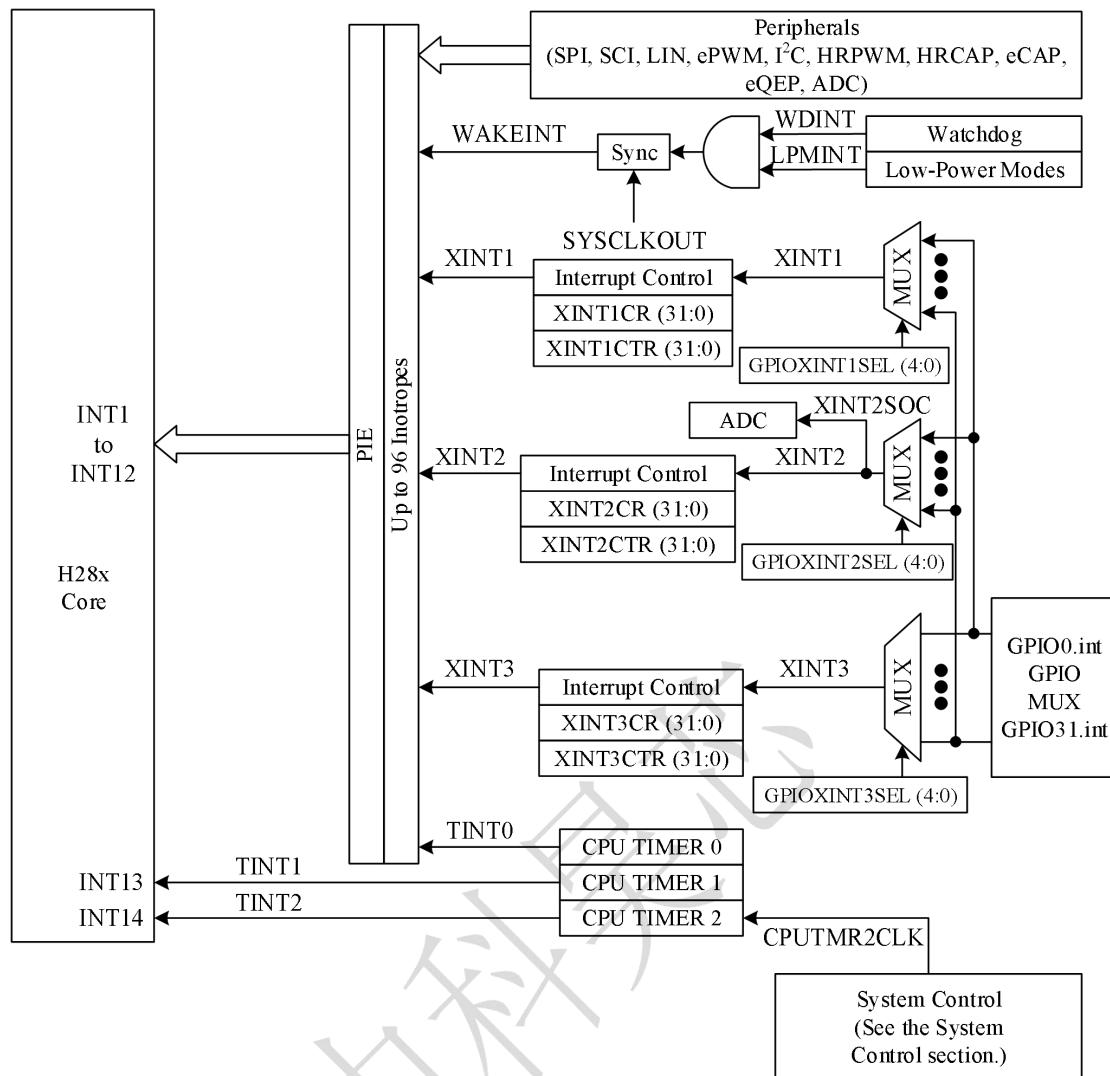


图 3-2 PIE 模块互连示意图

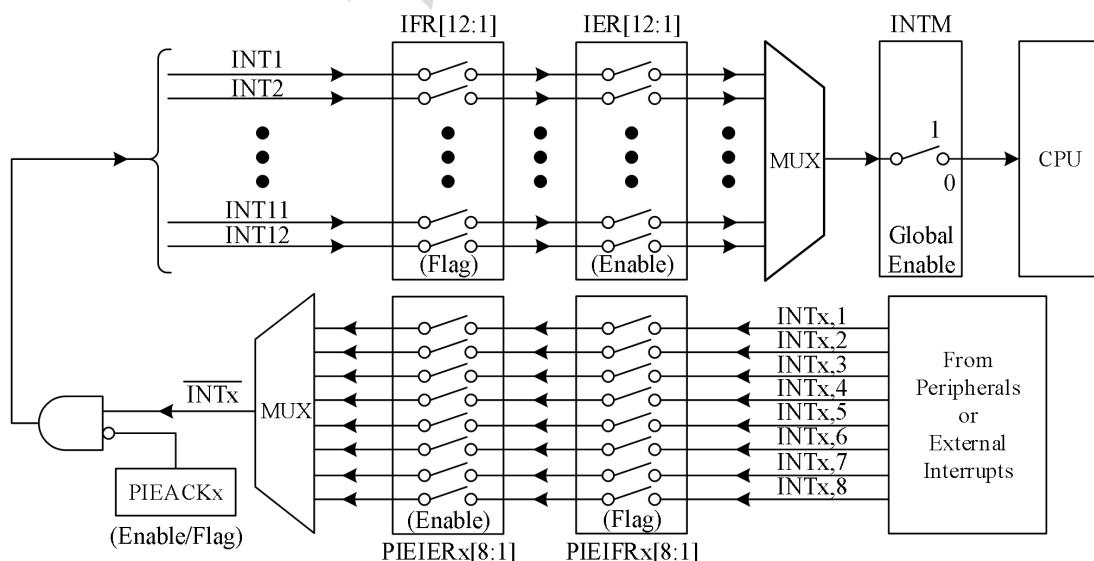


图 3-3 PIE 中断映射示意图

表 3-8 PIE 外设中断向量表

| 中断向量入口 0x1800 | INTx. 8 | INTx. 7 | INTx. 6 | INTx. 5 | INTx. 4 | INTx. 3 | INTx. 2 | INTx. 1 |
|------------------|----------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| INT1.y | WAKE INT (LPM/WD) 0x21C | TINT0 (TIMERO) 0x218 | ADCINT9 (ADC) 0x214 | XINT2 Ext int 2 0x210 | XINT1 Ext int 1 0x20C | Reserved - 0x208 | ADCINT2 (ADC) 0x204 | ADCINT1 (ADC) 0x200 |
| INT2.y | EPWM8_TZIN T 0x23C | EPWM7_TZIN T (ePWM7) 0x238 | EPWM6_TZIN T (ePWM6) 0x234 | EPWM5_TZIN T (ePWM5) 0x230 | EPWM4_TZIN T (ePWM4) 0x22C | EPWM3_TZIN T (ePWM3) 0x228 | EPWM2_TZIN T (ePWM2) 0x224 | EPWM1_TZIN T (ePWM1) 0x220 |
| INT3.y | EPWM7_INT (ePWM7) 0x25C | EPWM7_INT (ePWM7) 0x258 | EPWM6_INT (ePWM6) 0x254 | EPWM5_INT (ePWM5) 0x250 | EPWM4_INT (ePWM4) 0x24C | EPWM3_INT (ePWM3) 0x248 | EPWM2_INT (ePWM2) 0x244 | EPWM1_INT (ePWM1) 0x240 |
| INT4.y | HRCAP2_INT (HRCAP2) 0x27C | HRCAP1_INT (HRCAP1) 0x278 | Reserved - 0x274 | Reserved - 0x270 | Reserved - 0x26C | Reserved - 0x268 | Reserved - 0x264 | ECAP1_INT (ECAP1) 0x260 |
| INT5.y | Reserved - 0x29C | Reserved - 0x298 | Reserved - 0x294 | Reserved - 0x290 | Reserved - 0x28C | Reserved - 0x288 | EQEP2_INT (eQEP2) 0x284 | EQEP1_INT (eQEP1) 0x280 |
| INT6.y | Reserved - 0x2BC | Reserved - 0x2B8 | Reserved - 0x2B4 | Reserved - 0x2B0 | SPITXINTB (SPI-B) 0x2AC | SPIRXINTB (SPI-B) 0x2A8 | SPITXINTA (SPI-A) 0x2A4 | SPIRXINTA (SPI-A) 0x2A0 |
| INT7.y | Reserved - 0x2DC | Reserved - 0x2D8 | DINCH6 (DMA) 0x2D4 | DINCH5 (DMA) 0x2D0 | DINCH4 (DMA) 0x2CC | DINTCH3 (DMA) 0x2C8 | DINTCH2 (DMA) 0x2C4 | DINTCH1 (DMA) 0x2C0 |
| INT8.y | Reserved - 0x2FC | Reserved - 0x2F8 | Reserved - 0x2F4 | Reserved - 0x2F0 | Reserved - 0x2EC | Reserved - 0x2E8 | I2CINT2A (I2C-A) 0x2E4 | I2CINT1A (I2C-A) 0x2E0 |
| INT9.y | Reserved - 0x31C | Reserved - 0x318 | ECAN1_INTA 0x314 | ECANO_INTA 0x310 | LIN1_INTA 0x30C | LINO_INTA 0x308 | SCITXINTA 0x304 | SCIRXINTA 0x300 |
| INT10.y | ADCINT8 (ADC) 0x33C | ADCINT7 (ADC) 0x338 | ADCINT6 (ADC) 0x334 | ADCINT5 (ADC) 0x330 | ADCINT4 (ADC) 0x32C | ADCINT3 (ADC) 0x328 | ADCINT2 (ADC) 0x324 | ADCINT1 (ADC) 0x320 |
| INT11.y | CLA1_INT8 (CLA) - 0x35C | CLA1_INT7 (CLA) - 0x358 | CLA1_INT6 (CLA) - 0x354 | CLA1_INT5 (CLA) - 0x350 | CLA1_INT4 (CLA) - 0x34C | CLA1_INT3 (CLA) - 0x348 | CLA1_INT2 (CLA) - 0x344 | CLA1_INT1 (CLA) - 0x340 |
| INT12.y | CLA1_LUF (CLA) - 0x37C | CLA1_LVF (CLA) - 0x378 | CPU_CUF (CPU) -- 0x374 | CPU_LVF (CPU) - 0x370 | Reserved - 0x36C | Reserved - 0x368 | Reserved - 0x364 | XINT3 Ext int3 0x360 |
| INT13.y | Reserved - 0x39C | Reserved - 0x398 | Reserved - 0x394 | Reserved - 0x390 | Reserved - 0x38C | Reserved - 0x388 | Reserved - 0x384 | TIMER1 (TIMER) 0x380 |
| INT14.y | Reserved - 0x3BC | Reserved - 0x3B8 | Reserved - 0x3B4 | Reserved - 0x3B0 | Reserved - 0x3AC | Reserved - 0x3A8 | Reserved - 0x3A4 | TIMER2 (TIMER) 0x3A0 |

表 3-9 PIE 配置和控制寄存器（基地址为 0x1800）

| 名称 | 偏移地址 | 描述 |
|----------|-------|-------------------|
| PIECTRL | 0x100 | PIE 控制寄存器 |
| PIEACK | 0x104 | PIE 确认寄存器 |
| PIEIER1 | 0x108 | PIE, 中断第 1 组使能寄存器 |
| PIEIFR1 | 0x10C | PIE 中断第 1 组标志寄存器 |
| PIEIER2 | 0x110 | PIE 中断第 2 组使能寄存器 |
| PIEIFR2 | 0x114 | PIE 中断第 2 组标志寄存器 |
| PIEIER3 | 0x118 | PIE 中断第 3 组使能寄存器 |
| PIEIFR3 | 0x11C | PIE 中断第 3 组标志寄存器 |
| PIEIER4 | 0x120 | PIE 中断第 4 组使能寄存器 |
| PIEIFR4 | 0x124 | PIE 中断第 4 组标志寄存器 |
| PIEIER5 | 0x128 | PIE 中断第 5 组使能寄存器 |
| PIEIFR5 | 0x12C | PIE 中断第 5 组标志寄存器 |
| PIEIER6 | 0x130 | PIE 中断第 6 组使能寄存器 |
| PIEIFR6 | 0x134 | PIE 中断第 6 组标志寄存器 |
| PIEIER7 | 0x138 | PIE 中断第 7 组使能寄存器 |
| PIEIFR7 | 0x13C | PIE 中断第 7 组标志寄存器 |
| PIEIER8 | 0x140 | PIE 中断第 8 组使能寄存器 |
| PIEIFR8 | 0x144 | PIE 中断第 8 组标志寄存器 |
| PIEIER9 | 0x148 | PIE 中断第 9 组使能寄存器 |
| PIEIFR9 | 0x14C | PIE 中断第 9 组标志寄存器 |
| PIEIER10 | 0x150 | PIE 中断第 10 组使能寄存器 |
| PIEIFR10 | 0x154 | PIE 中断第 10 组标志寄存器 |
| PIEIER11 | 0x158 | PIE 中断第 11 组使能寄存器 |
| PIEIFR11 | 0x15C | PIE 中断第 11 组标志寄存器 |

| 名称 | 偏移地址 | 描述 |
|----------|-------|-------------------|
| PIEIER12 | 0x160 | PIE 中断第 12 组使能寄存器 |
| PIEIFR12 | 0x164 | PIE 中断第 12 组标志寄存器 |
| XINT1CR | 0x168 | 外部中断 1 配置寄存器 |
| XINT2CR | 0x16C | 外部中断 2 配置寄存器 |
| XINT3CR | 0x170 | 外部中断 3 配置寄存器 |
| XINT1CTR | 0x174 | 外部中断 1 计数器寄存器 |
| XINT2CTR | 0x178 | 外部中断 2 计数器寄存器 |
| XINT3CTR | 0x17C | 外部中断 3 计数器寄存器 |

3.7 LDO/VREG/BOR/POR

虽然核心和 I/O 电路在两种不同的电压下工作，但这些设备具有片上稳压器（VREG）从 V_{DDIO} 电源产生 V_{DD} 电压。这消除了在应用板上使用第二个外部稳压器的成本和空间。此外，内部上电复位(POR) 在通电和运行模式下，掉电复位(BOR) 电路同时监视 V_{DD} 和 V_{DDIO} 轨道。

3.7.1 VREG

线性稳压器从 V_{DDIO} 电源产生核心电压 (V_{DD})。因此，尽管每个 V_{DD} 引脚上都需要电容器来稳定所产生的的电压，但并不需要向这些引脚供电来操作设备。相反，如果电源或冗余成为应用程序的主要考虑因素，则可以禁用 VREG。

3.7.1.1 使用片上 VREG

为了采用片载 VREG，VREGNZ 引脚应该被接至低电平并且适当的建议运行电压应该被用于 V_{DDA} 和 V_{DDIO} 引脚。在这个情况下，内核逻辑所需的 V_{DD} 电压将由 VREG 生成。为了实现 VREG 正确的调节，每一个 V_{DD} 引脚需要电容值为 $1.2 \mu F$ 的电容器。这些电容应该被放置在尽可能接近 V_{DD} 引脚的位置。

3.7.1.2 不使用片上 VREG

为了节约能耗，也可禁用片载 VREG 并使用一个效率更高的外部稳压器将内核逻辑电压提供给 V_{DD} 引脚。为了启用这个选项，VREGNZ 引脚必须被接至高电平。

3.7.2 片上上电复位 (POR) 和掉电复位 (BOR) 电路

两个片上监控电路，上电复位 (POR) 和掉电复位 (BOR) 从应用板上移除了监控 V_{DD} 和 V_{DDIO} 电源的负担。POR 的目的是在整个上电过程中，在整个设备中创建一个干净的复位。此触发点是一个非固定的，比 BOR 更低的触发点，将在器件运行期间监视 V_{DD} 或 V_{DDIO} 电源轨道的骤减操作。POR 功能总是出现在 V_{DD} 和 V_{DDIO} 电源轨道上。设备首次上电后 BOR 功能总是出现在 V_{DDIO} 上，并且当内部 VREG 启用时，出现在 V_{DD} 上（VREGNZ 引脚被接至低电平）。当其中一个电压低于它们各自的触发点时两个功能连接至 XRS 引脚低电平。此外，当内部电压稳压器被启用时，一个过压保护电路将连接至 XRS 低电平，此时 V_{DD} 电源轨道上升至高于其触发点。图 3-4 显示 VREG、POR 和 BOR。

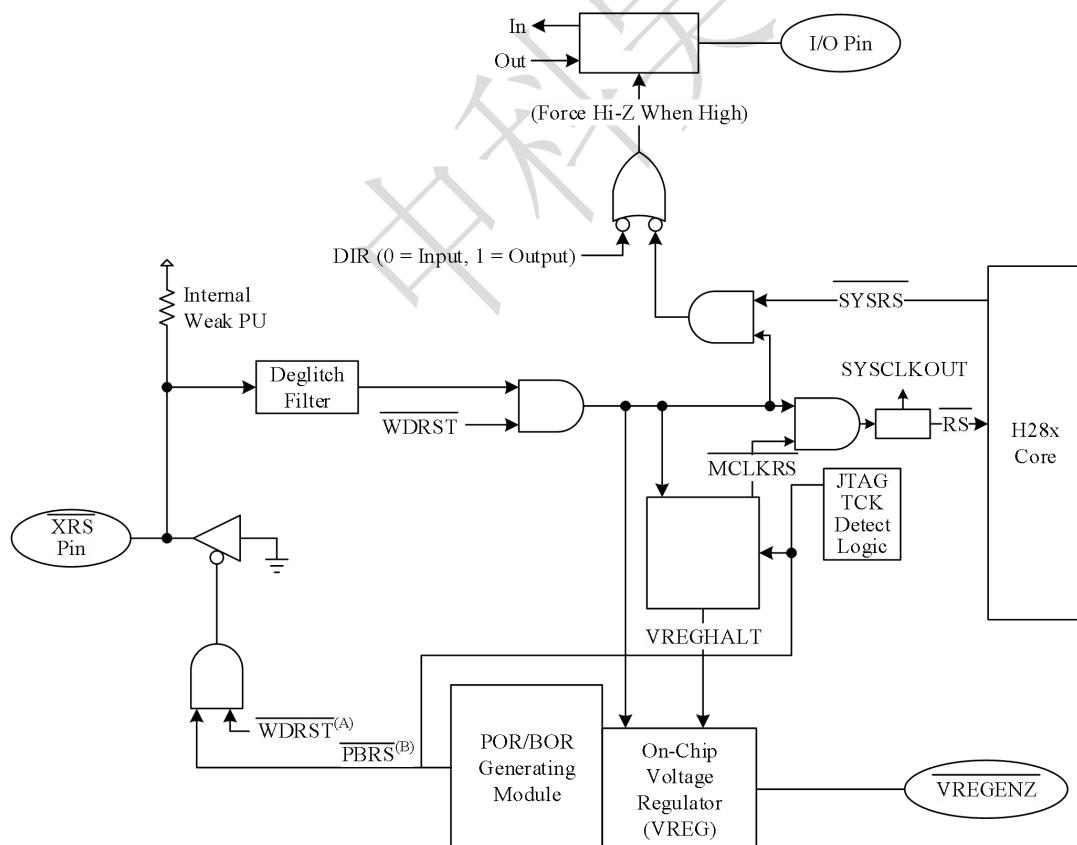


图 3-4 VREG+POR+BOR+复位信号连接

A. WDRST 为从 CPU-看门狗发出的复位信号

B. PBRS 是从 POR/BOR 发出的复位信号

3.8 系统控制

这一部分对振荡器和时钟机制、看门狗功能以及低功耗模式进行了说明。

所有系统控制寄存器都是 EALLOW-protected 寄存器，系统控制寄存器的基址地址为：

0x0000DC00

表 3-10 系统控制寄存器映射

| 地址偏移 | 寄存器 | 描述 |
|------|-------------|---------------------|
| 0h | XCLK | XCLKOUT/XCLKIN 时钟控制 |
| 4h | PLLSTS | PLL 状态寄存器 |
| 8h | PCLKCR0 | 外设时钟控制寄存器 0 |
| ch | PCLKCR1 | 外设时钟控制寄存器 1 |
| 10h | PCLKCR2 | 外设时钟控制寄存器 2 |
| 14h | PCLKCR3 | 外设时钟控制寄存器 3 |
| 18h | LOSPCP | 低速外设时钟预分频寄存器 |
| 1ch | INTOSC1TRIM | 内部晶振校准寄存器 1 |
| 20h | INTOSC2TRIM | 内部晶振校准寄存器 2 |
| 24h | CLKCTL | 时钟控制寄存器 |
| 28h | PLLCR | PLL 控制寄存器 |
| 2ch | PLLLOCKPRD | PLLLOCKPRD 寄存器 |
| 30h | LPMCR0 | 低功耗模式控制寄存器 |
| 34h | SCSR | 系统控制状态寄存器 |
| 38h | WDCNTR | 看门狗计数器寄存器 |
| 3ch | WDKEY | 看门狗复位秘钥寄存器 |
| 40h | WDCR | 看门狗控制寄存器 |
| 44h | JTAGDEBUG | JTAG 控制寄存器 |
| 60h | BORCFG | BOR 配置寄存器 |
| 80h | DEVICECNF | 器件配置寄存器 |
| 84h | CLASSID | CLASSID 寄存器 |
| 88h | REVID | 芯片版本寄存器 |

注 1：这个表中的所有寄存器是受 EALLOW 保护的。

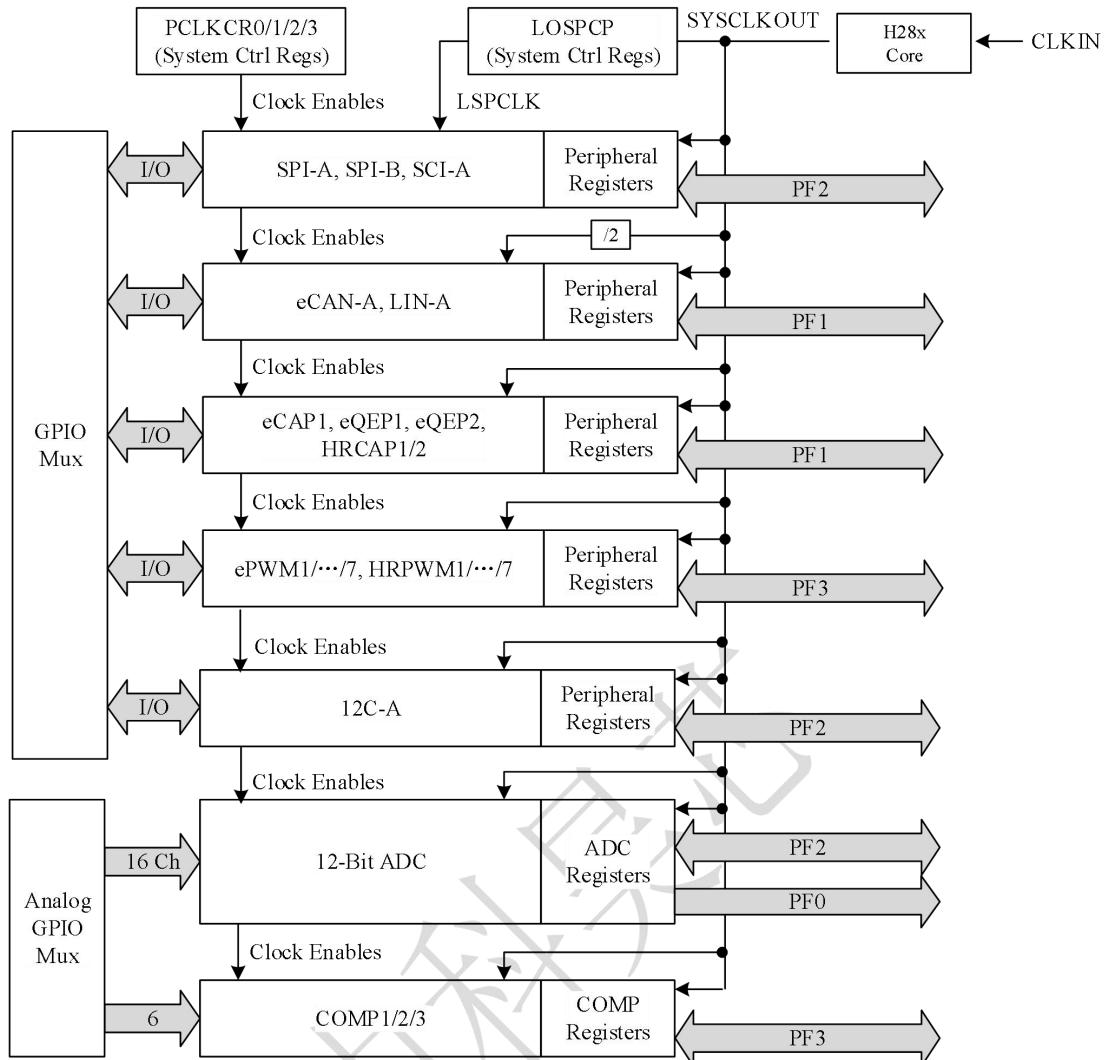


图 3-3 时钟及复位域

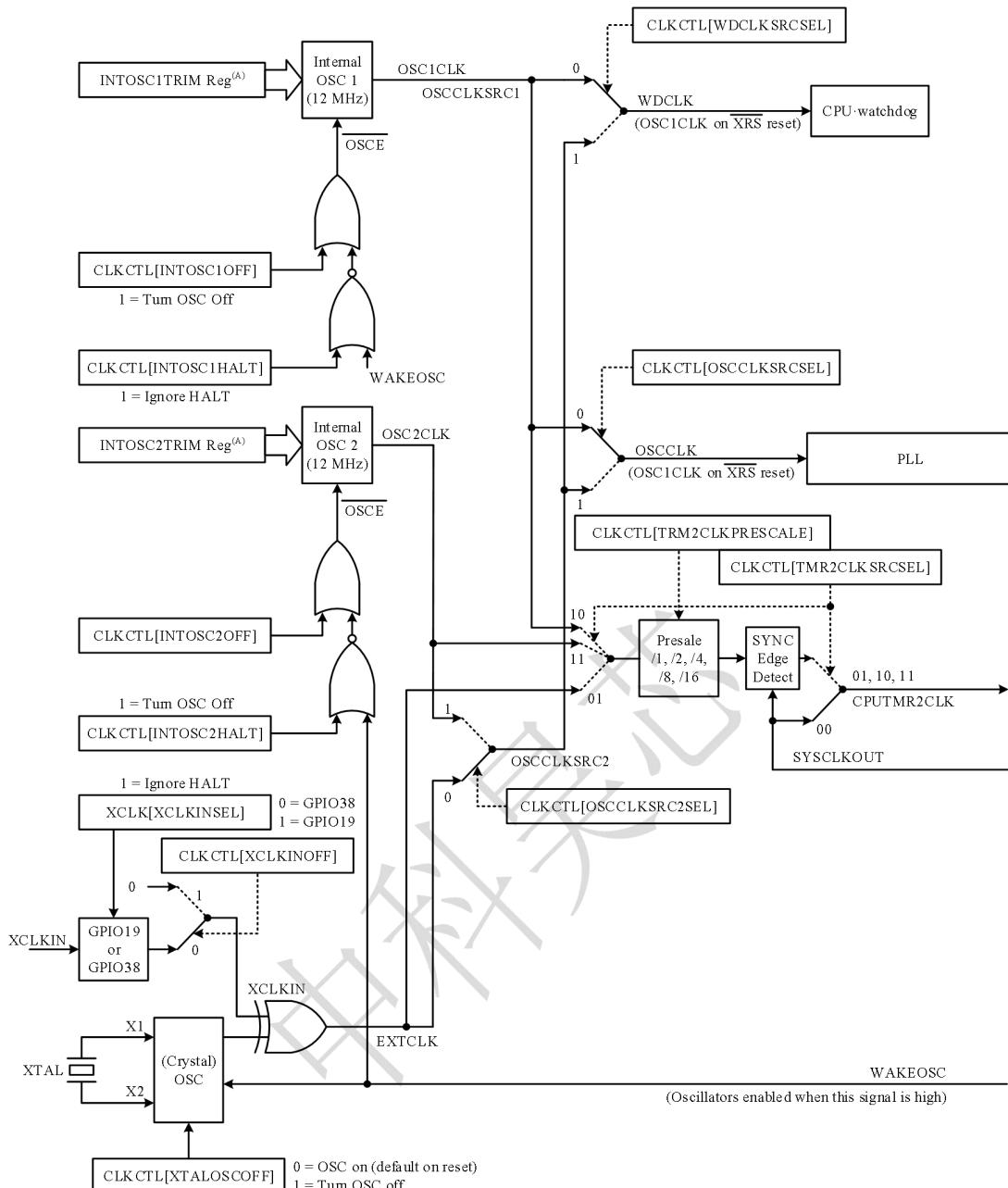


图 3-4 时钟树

A. 由 OTP 中的校正程序载入寄存器

3.8.1 内部零引脚振荡器

HXS320F28035 器件包含两个独立的内部零引脚振荡器。默认情况下两个振荡器在上电时全都打开，此时内部振荡器 1 是默认时钟源。为了降低功耗，用户可将不使用的振荡器关断。这些振荡器的中心频率由它们各自的振荡器调整寄存器决定，此寄存器在校准例程中被写入作为引导 ROM 执行的一部分。

3.8.2 晶体振荡器选项

片上晶体振荡器 X1 和 X2 引脚为 1.8V 电平信号，并且绝对不能为 3.3V 电平。如果将系统 3.3V 外部振荡器用作时钟源，则应将 XCLKIN 用作时钟源。X1 引脚不打算用作单端时钟输入，它是应与 X2 和晶体一起使用。

外部石英晶体的典型规格（基本模式，并联谐振）在以下列表中列出。此外，ESR 范围= 30 至 150 Ω 。

表格 3-11 外部石英晶体的典型规格

| 频率 (MHz) | $R_d (\Omega)$ | $C_{L1} (pF)$ | $C_{L2} (pF)$ |
|----------|----------------|---------------|---------------|
| 5 | 2200 | 18 | 18 |
| 10 | 470 | 15 | 15 |
| 15 | 0 | 15 | 15 |
| 20 | 0 | 12 | 12 |

注： C_{shunt} 应该小于或等于 5pF。

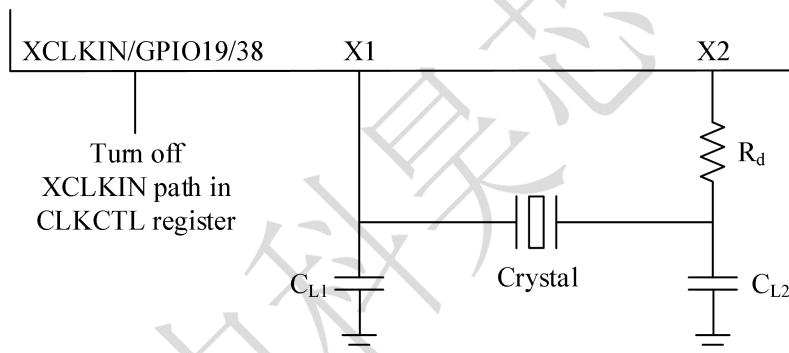


图 3-5 使用片上晶体振荡器

- C_{L1} 和 C_{L2} 是电路板和不包括电容的组件的总电容，该值通常约为晶体负载值的两倍电容。
- 晶体的负载电容在晶振制造商的晶体规格中有所描述。
- 中科昊芯建议客户让谐振器/晶体供应商来表征他们的设备使用 MCU 芯片进行操作。谐振器/晶体供应商拥有调整震荡电路的设备和专业知识。供应商还可以建议 客户关于适当的元器件值将产生正确的启动整个工作范围内的稳定性。

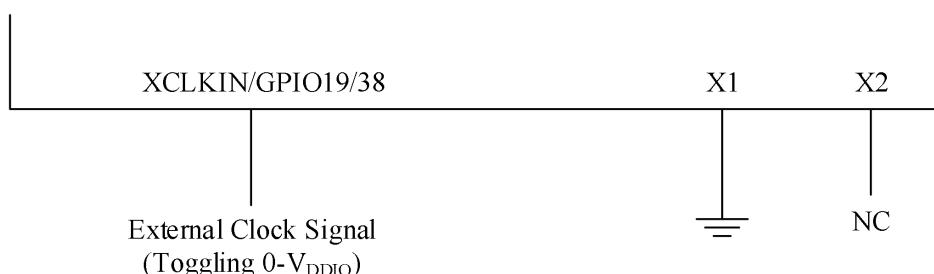


图 3-6 使用 3.3v 片外晶体振荡器

3.8.3 基于 PLL 的时钟模块

HXS320F28035 有一个基于片载 PLL 的时钟模块。这个模块为器件提供所有需要的时钟信号以及对进入低功耗模式的控制。PLL 有一个 4 位比率控制 PLLCR[DIV]来选择不同的 CPU 时钟速率。在写入 PLLCR 寄存器之前，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 1ms。输入时钟和 PLLCR[DIV]位的选择方法应该是在 PLL(VCOCLK) 的输出频率至少为 50MHz 的时候再选择。

3.9 低功耗模块

HXS320F28035 处理器的低功耗模式如下：

表 3-12 低功耗模式列表

| 模式 | LPMCR0[1:0] | OSCCLK | CLKIN | SYSCLKOUT | Exit |
|---------|-------------|------------------------------------|-------|-----------|---------------------------------|
| IDLE | 00 | On | On | On | XRS, 看门狗中断， 任何中断 |
| STANDBY | 01 | On (看门狗还 在运行) | Off | Off | XRS, 看门狗中断， GPIO 端口 A 信号， |
| HALT | 1X | Off (OSC 和 PLL 关断，看门 狗停止) | Off | Off | XRS， GPIO 端口 A 信号， |

在各种低功耗模式下，处理器进行的操作如下表所示：

- 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 GPIOLOPMSEL 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：**CPU 看门狗，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 GPIOLOPMSEL 寄存器中选择信号。

HXS320F28035 处理器提供了两种选项在没有外部激励的情况下，可自动从 HALT 和 STANDBY 模式唤醒

- 从 HALT 模式唤醒: 将 CLKCTL 寄存器中 WDHALTI 位置 1, 通过处理器看门狗复位将处理器从 HALT 模式唤醒。WDCR 寄存器中的 WDFLAG 位用来区别处理器的看门狗复位和设备复位。
- 从 STANDBY 模式唤醒: 将寄存器 LPMCR0 中的 WDINTE 位置 1, 通过 WAKEINT 中断将处理器从 STANDBY 模式唤醒。

中科昊芯

4. 外设

4.1 模拟器件

图 4-1 显示了模拟器件与 HXS320F28035 系统其余部分的交互。

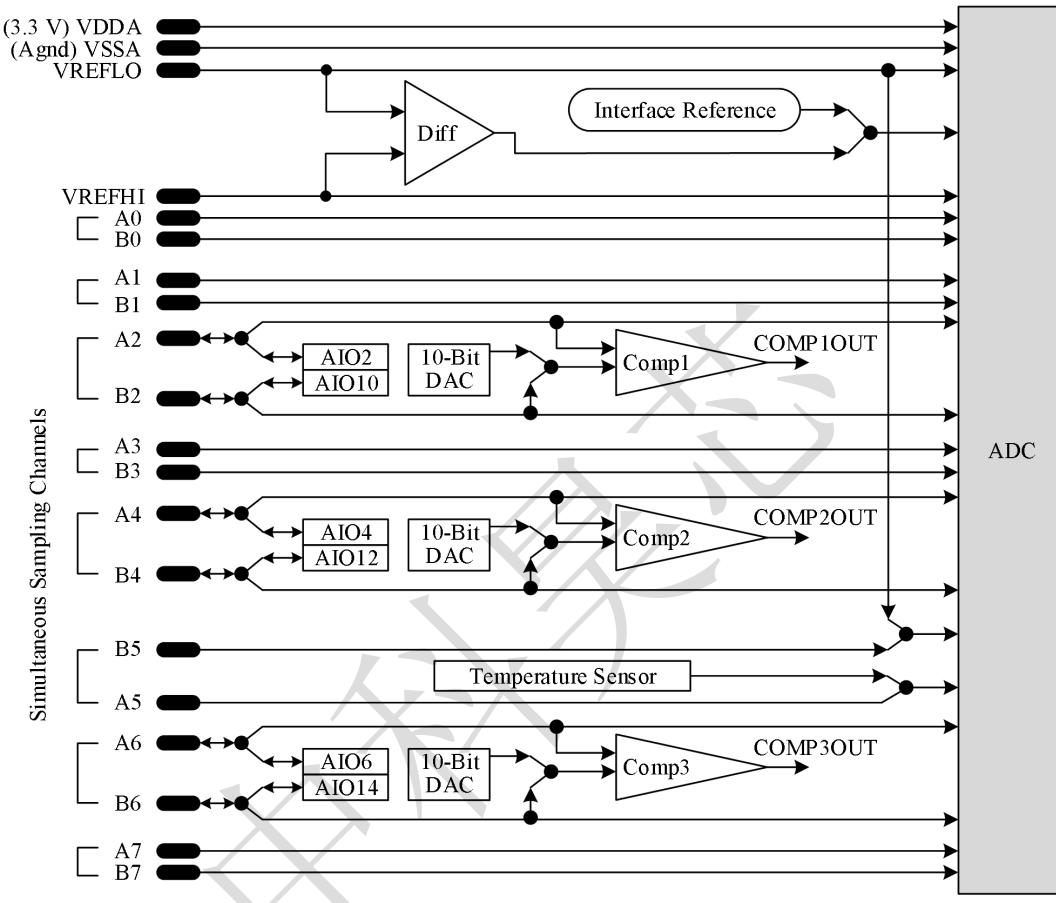


图 4-1 模拟引脚配置

4.1.1 特性

ADC 的核包含一个 12 位转换器, 转换器由两个采样保持电路供电。采样保持电路可以同时采样, 也可以按顺序采样。ADC 总共有多达 16 个模拟输入通道。转换器可被配置为使用一个内部带隙基准来创建真正的基于电压的转换。

与之前的 ADC 类型不同此 ADC 不是基于序列器的。用户很容易从单个触发器创建一系列转换。然而, 操作的基本原理是围绕配置的单个转换的开始, 称为 SOC 或转换的开始。

ADC 模块的功能包括：

内置双采样保持 (S/H) 的 12 位 ADC 核心

同时采样或顺序采样模式全量程模拟输入：0V 至 3.3V 固定。模拟输入的数字值

电压由以下公式得出：

当 输入 $\leq 0V$, 数字值 = 0

当 $0V < \text{输入} < 3.3V$, 数字值 = $4096 \times \frac{\text{模拟输入电压} - V_{REFLO}}{3.3}$

当 输入 $\geq 3.3V$, 数字值 = 4095

- 最多 16 通道，多路输入
- 16 个 SOC，可配置触发器、采样窗口和通道
- 16 个结果寄存器（可单独寻址），用于存储转换值
- 多触发源
 - 软件 - 软件立即启动
 - ePWM 1 - 7
 - GPIO XINT2
 - CPU 计时器 0/1/2
 - ADC 中断 1/2
- 9 个灵活的 PIE 中断，可配置任意转换后的中断请求

表 4-1 ADC 配置和控制寄存器

| 名称 | 地址 | EALLOW 保护 | 描述 |
|--------------|--------|-----------|----------------|
| ADCCTL1 | 0x1400 | 是的 | 控制 1 寄存器 |
| ADCCTL2 | 0x1404 | 是的 | 控制 2 寄存器 |
| ADCINTFLG | 0x1408 | 不 | 中断标志寄存器 |
| ADCINTFLGCLR | 0x140C | 不 | 中断标志清除寄存器 |
| ADCINTOVF | 0x1410 | 不 | 中断溢出寄存器 |
| ADCINTOVFCLR | 0x1414 | 不 | 中断溢出清除寄存器 |
| INTSEL1N2 | 0x1418 | 是的 | 中断 1 和 2 选择寄存器 |
| INTSEL3N4 | 0x141C | 是的 | 中断 3 和 4 选择寄存器 |
| INTSEL5N6 | 0x1420 | 是的 | 中断 5 和 6 选择寄存器 |

| 名称 | 地址 | EALLOW 保护 | 描述 |
|----------------------------|--------------------|-----------|--------------------------|
| INTSEL7N8 | 0x1424 | 是的 | 中断 7 和 8 选择寄存器 |
| INTSEL9N10 | 0x1428 | 是的 | 中断 9 选择寄存器(保留中断 10 选择) |
| SOCPRICCTL | 0x142C | 是的 | SOC 优先级控制寄存器 |
| ADCSAMPLEMODE | 0x1430 | 是的 | 采样模式寄存器 |
| ADCINTSOCSEL1 | 0x1434 | 是的 | 中断 SOC 选择 1 寄存器 (8 通道) |
| ADCINTSOCSEL2 | 0x1438 | 是的 | 中断 SOC 选择 2 寄存器 (8 通道) |
| ADCSOCFLG1 | 0x143C | 不 | SOC 标志 1 寄存器 (16 通道) |
| ADCSOCFRC1 | 0x1440 | 不 | SOC 力 1 寄存器 (16 通道) |
| ADCSOCOVF1 | 0x1444 | 不 | SOC 溢出 1 寄存器 (用于 16 通道) |
| ADCSOCOVFCLR1 | 0x1448 | 不 | SOC 溢出清除 1 寄存器(用于 16 通道) |
| ADCSOCOCTL ~ADCSOC15CTL | 0x144C - 0x1488 | 是的 | SOC0 控制寄存器到 SOC15 控制寄存器 |
| ADCREFTRIM | 0x148c | 是的 | 无效 |
| ADCOFFTRIM | 0x1490 | 是的 | 偏移微调寄存器 |
| COMPHYSTCTL | 0x1494 | 是的 | 比较器滞后控制寄存器 |
| ADCREV | 0x1498 | 不 | ADC 版本号 |

ADC 结果寄存器寄存器

| 名称 | 地址 | EALLOW 保护 | 描述 |
|----------------------------|-----------------|-----------|-----------|
| ADCRESULT0~ ADCRESULT15 | 0x149C 至 0x14D8 | 否 | ADC 结果寄存器 |

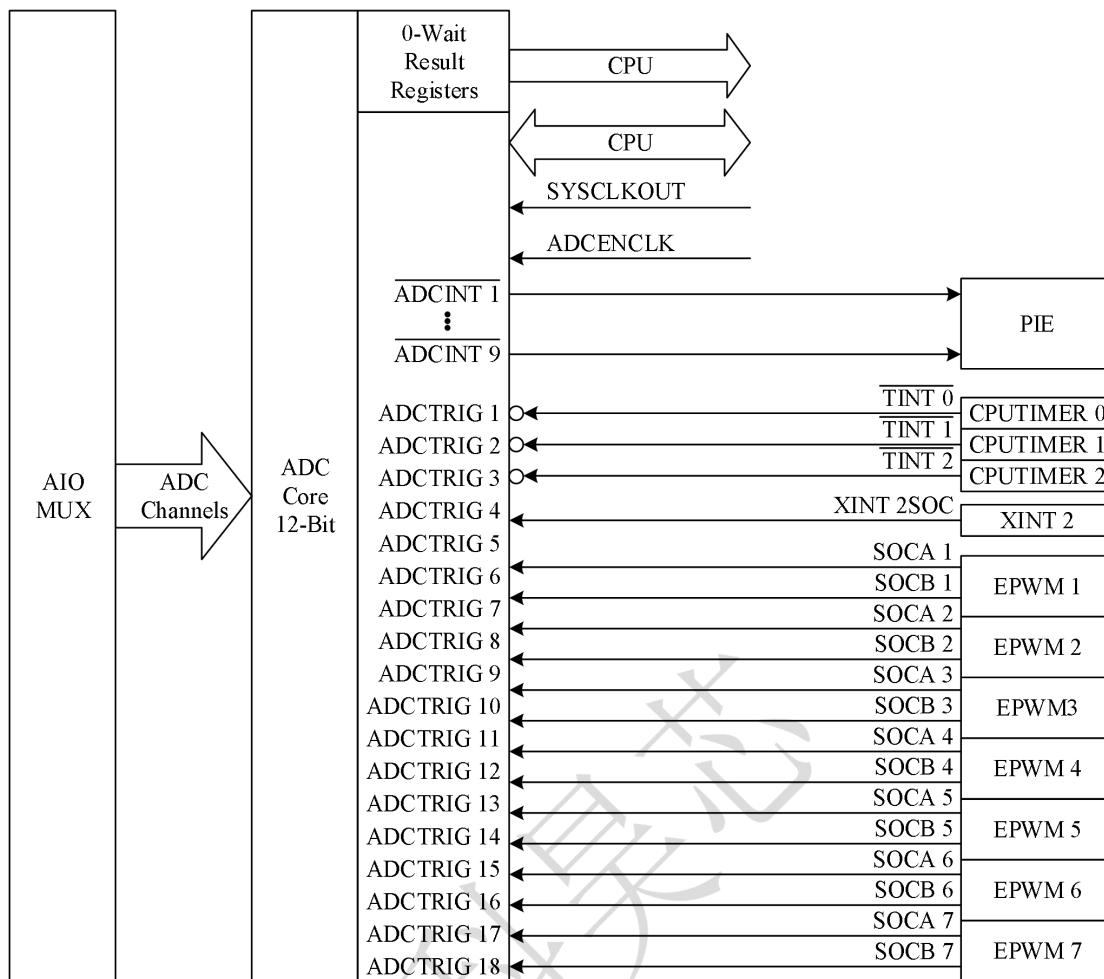


图 4-2 ADC 模块连接图

ADC 连接（如果未使用 ADC）

中科昊芯建议保持模拟电源引脚的连接，即使未使用 ADC。下面是一个如果应用程序中未使用 ADC，则应如何连接 ADC 引脚的摘要：

- VDDA - 连接到 VDDIO
- VSSA - 连接到 VSS
- VREFLO - 连接到 VSS
- ADCINAn、ADCINBn、VREFHI - 连接到 VSSA

在应用中使用 ADC 模块时，未使用的 ADC 输入引脚应连接到模拟地（VSSA）。

注意：

与 AIO 功能多路复用的未使用的 ADCIN 引脚不应直接连接到模拟地。它们应通过 $1-k\Omega$ 电阻器接地。这是为了防止错误代码将这些引脚配置为 AIO 输出，并将接地引脚驱动到逻辑高状态。

当不使用 ADC 时，确保 ADC 模块的时钟未打开以实现节能。

4.1.2 ADC MUX 多路复选

ADC 通道和比较器功能始终可用。数字 I/O 功能仅在以下情况下可用 AIOMUX1 寄存器中的相应位为 0。在这种模式下，读取 AIODAT 寄存器反映实际的 pin 状态。

当 AIOMUX1 寄存器中的相应位为 1 时，数字 I/O 功能被禁用。在此模式下，读取 AIODAT 寄存器反映 AIODAT 寄存器的输出锁存，并且输入数字 I/O 缓冲区被禁用防止模拟信号产生噪音。

复位时，数字功能被禁用。如果引脚用作模拟输入，用户应保留 AIO 该 pin 的功能已禁用。

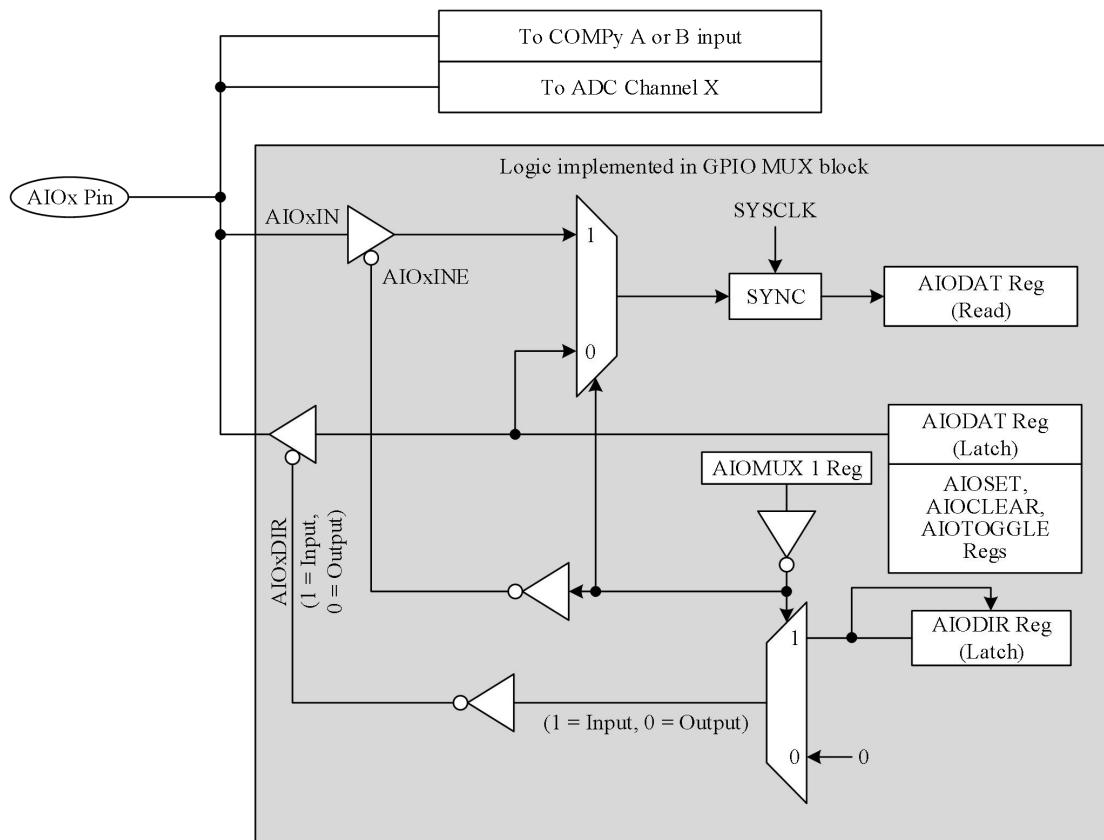


图 4-8 AIOx 引脚复用

4.1.3 比较器模块

下图显示了比较器模块与系统其余部分的交互。

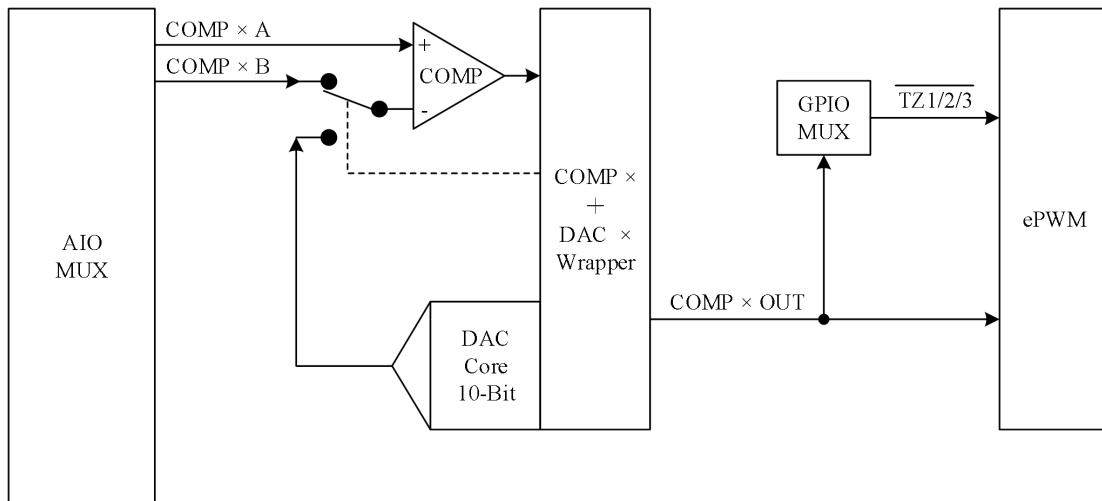


图 4-9 比较器模块结构框图

表 4-3 比较器模块寄存器映射

| 名称 | 比较器 1 | 比较器 2 | 比较器 3 | EALLOW 保护 | 描述 |
|-------------------|--------|--------|--------|-----------|-----------|
| COMPCTL | 0xC000 | 0xC080 | 0xC100 | 是的 | 比较器控制寄存器 |
| COMPSTS | 0xC004 | 0xC084 | 0xC104 | 不 | 比较器状态寄存器 |
| DACCTL | 0xC008 | 0xC088 | 0xC108 | 不 | DAC 控制寄存器 |
| DACVAL | 0xC00C | 0xC08C | 0xC10C | 不 | DAC 值寄存器 |
| RAMPMAXREF_ACTIVE | 0xC010 | 0xC090 | 0xC110 | 不 | 斜坡发生器最大值 |
| RAMPMAXREF_SHDW | 0xC014 | 0xC094 | 0xC114 | 不 | 参考（活动）寄存器 |
| RAMPDECVL_ACTIVE | 0xC018 | 0xC098 | 0xC118 | 不 | 斜坡发生器最大值 |
| RAMPDECVL_SHDW | 0xC01C | 0xC09C | 0xC11C | 不 | 参考（阴影）寄存器 |
| RAMPSTS | 0xC020 | 0xC0A0 | 0xC110 | 不 | 斜坡发生器衰减值 |
| DACEX | 0xC024 | 0xC0A4 | 0xC114 | 不 | （活动）寄存器 |

4.2 同步串行通信接口 (SPI) 模块

HXS320F28035 器件包括 2 个四引脚的同步串行通信接口 (SPI) 模块。SPI 是一个高速、同步串行 I/O 端口，此端口可在设定的位传输速率上将一个设定长度（1 至 16 位）的串行比特流移入和移出器件。通常，SPI 用于 DSP 和外部外设或者其它处理器之间的通信。典型应用包括外部 I/O 或者从诸如移位寄存器、显示驱动器和 ADC 等器件的外设扩展。多器件通信由 SPI 的主控/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
- 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: 主机模式和从机模式
- 波特率: 共 125 个可编程的不同波特率以供使用。计算公式为:

当 SPIBRR = 3 到 127 时:

$$\text{SPI 波特率} = \frac{\text{LSPCLK}}{(\text{SPIBRR} + 1)}$$

当 SPIBRR = 0, 1, 或 2 时:

$$\text{SPI 波特率} = \frac{\text{LSPCLK}}{4}$$

其中:

LSPCLK = 设备低速外设时钟频率

SPIBRR = SPI 主机的 SPIBRR 的值

- 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 E800h。
- 增强型特性:
- 4 级发送/接收 FIFO

- 经延迟的发送控制
- 支持双向 3 线 SPI 模式
- 借助 SPISTE 翻转的音频数据接收支持

表格 4-4 SPI-A 寄存器

| 寄存器 | 地址 | 寄存器描述 |
|----------|---------|------------------|
| SPICCR | 0xE800h | SPI-A 配置控制寄存器 |
| SPICTL | 0xE804h | SPI-A 运行控制寄存器 |
| SPISTS | 0xE808h | SPI-A 状态寄存器 |
| SPIBRR | 0xE80Ch | SPI-A 波特率寄存器 |
| SPIRXEMU | 0xE810h | SPI-A 接收仿真缓存寄存器 |
| SPIRXBUF | 0xE814h | SPI-A 串行输入缓存寄存器 |
| SPITXBUF | 0xE818h | SPI-A 串行输出缓存寄存器 |
| SPIDAT | 0xE81Ch | SPI-A 串行数据寄存器 |
| SPIFFTX | 0xE820h | SPI-A FIFO 发送寄存器 |
| SPIFFRX | 0xE824h | SPI-A FIFO 接收寄存器 |
| SPIFFCT | 0xE828h | SPI-A FIFO 控制寄存器 |
| SPIPRI | 0xE82Ch | SPI-A 优先级控制寄存器 |

表格 4-5 SPI-B 寄存器

| 寄存器 | 地址 | 寄存器描述 |
|----------|---------|------------------|
| SPICCR | 0xE880h | SPI-B 配置控制寄存器 |
| SPICTL | 0xE884h | SPI-B 运行控制寄存器 |
| SPISTS | 0xE888h | SPI-B 状态寄存器 |
| SPIBRR | 0xE88Ch | SPI-B 波特率寄存器 |
| SPIRXEMU | 0xE890h | SPI-B 接收仿真缓存寄存器 |
| SPIRXBUF | 0xE894h | SPI-B 串行输入缓存寄存器 |
| SPITXBUF | 0xE898h | SPI-B 串行输出缓存寄存器 |
| SPIDAT | 0xE89Ch | SPI-B 串行数据寄存器 |
| SPIFFTX | 0xE8A0h | SPI-B FIFO 发送寄存器 |
| SPIFFRX | 0xE8A4h | SPI-B FIFO 接收寄存器 |
| SPIFFCT | 0xE8A8h | SPI-B FIFO 控制寄存器 |
| SPIPRI | 0xE8ACh | SPI-B 优先级控制寄存器 |

4.3 异步串行通信接口（SCI）模块

SCI 的基本功能如下：

- 对外两个数据信号与 GPIO 复用：SCITXD (Output)；SCIRXD (Input)；
- 波特率通过 16bit 寄存器可配置；
- 数据格式：start (1bit) +data (1~8bit 可配置) +地址数据标示 (1bit, 仅 address mode)
奇偶位 (奇/偶/无) +结束位 (1/2 bit)
- 四种错误检测：parity, overrun, framing, 和 break detection
- 全双工和半双工
- 接收发送双缓存
- 接收发送可中断触发或者轮询状态寄存器触发
- 接收发送中断分别使能 (BRKDT)
- NRZ 格式传输

增强的功能：

- 硬件自动检测波特率
- 4 级接收发送 FIFO

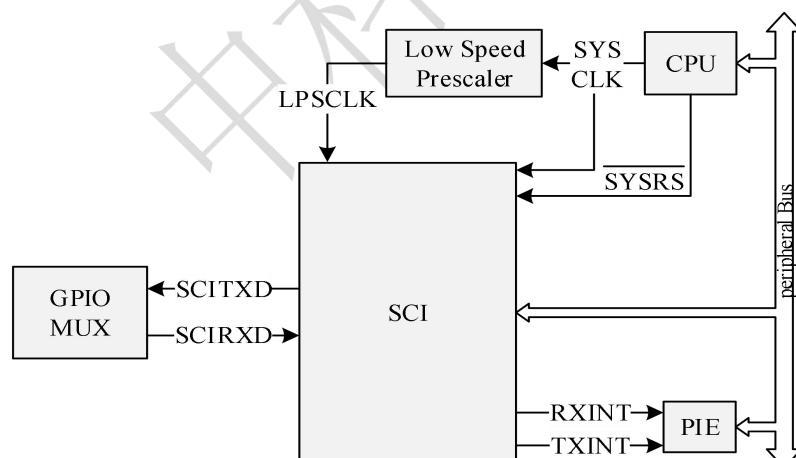


图 4-10 SCI 连接关系图

| 理想波特率 | LSPCLK 时钟频率, 100MHz | | |
|-------|---------------------|-------|------|
| | BRR | 实际波特率 | 误差% |
| 2400 | 5207(1457h) | 2400 | 0 |
| 4800 | 2603(A2Bh) | 4800 | 0 |
| 9600 | 1301(515h) | 9601 | 0.01 |
| 19200 | 650(28Ah) | 19201 | 0.01 |
| 38400 | 324(144h) | 38462 | 0.16 |

表格 4-6 波特率配置

串行传输的时钟由 LSPCLK 根据 16bit 配置寄存器生成。最大比达特为 LSPCLK/16.

$$BRR = (SCIHBAUD \ll 8) + (SCILBAUD)$$

$$SCI \text{ Asynchronous Baud} = LSPCLK / ((BRR + 1) * 8)$$

$$BRR = LSPCLK / (SCI \text{ Asynchronous Baud} * 8) - 1, \quad 0 < BRR < 65536$$

$$BRR=0 \text{ 时, } SCI \text{ Asynchronous Baud} = LSPCLK / 16$$

表 4-7 SCI-A 寄存器

| 名称 | 地址 | 说明 |
|----------|--------|------------------|
| SCICCR | 0XE000 | SCI-A 通信控制寄存器 |
| SCICTL1 | 0XE004 | SCI-A 控制寄存器 1 |
| SCIHBAUD | 0XE008 | SCI-A 波特率寄存器高位 |
| SCILBAUD | 0XE00C | SCI-A 波特率寄存器低位 |
| SCICTL2 | 0XE010 | SCI-A 控制寄存器 2 |
| SCIRXST | 0XE014 | SCI-A 接收状态寄存器 |
| SCITXEMU | 0XE018 | SCI-A 模拟缓冲寄存器 |
| SCIRXBUF | 0XE01C | SCI-A 数据接收缓冲器 |
| SCITXBUF | 0XE020 | SCI-A 数据传输寄存器 |
| SCIFFTX | 0XE024 | SCI-A FIFO 发送寄存器 |
| SCIFFRX | 0XE028 | SCI-A FIFO 接收寄存器 |
| SCIFFCT | 0XE02C | SCI-A FIFO 控制寄存器 |
| SCIPRI | 0XE030 | SCI-A 优先级寄存器 |

4.4 本地互联网络 (Lin) 模块

该设备包含一个 LIN 控制器。LIN 标准基于 SCI (UART) 串行数据链路格式。LIN 模块也可以配置为 SCI。

LIN 模块具有以下功能：

- 兼容 LIN 1.3 或 2.0 协议
- 两个外部引脚：LINRX 和 LINTX
- 多缓冲接收和发送单元
- 用于信息过滤的识别掩码

- 自动生成主机帧头
 - 可编程同步中段字段
 - 同步字段
 - 标识符字段
- 从机自动同步
 - 同步中断检测
 - 可选波特率更新
 - 同步验证
- 2^{31} 个可编程传输速率，7个分频比特位
- 从收发器接收唤醒 LINRX 的主电平
- 自动唤醒支持
 - 唤醒信号生成
 - 唤醒信号的超时时间
- 自动总线空闲检测
- 错误检测
 - 位错误
 - 总线错误
 - 无响应错误
 - 校验和错误
 - 同步字段错误
 - 奇偶校验错误
- 2条中断线，优先编码：
 - 接收
 - 传输
 - ID、错误和状态

表 4-8 LIN 寄存器

| 寄存器 | 地址 | 寄存器描述 |
|---------|--------|-----------|
| SCIGCR0 | 0xC400 | 全局控制寄存器 0 |
| SCIGCR1 | 0xC404 | 全局控制寄存器 1 |
| SCIGCR2 | 0xC408 | 全局控制寄存器 2 |

| | | |
|----------------|--------|---------------------------------|
| SCISETINT | 0xC40C | 中断使能寄存器 |
| SCICLEARINT | 0xC410 | 中断禁用寄存器 |
| SCISETINTLVL | 0xC414 | 设置中断级别寄存器 |
| SCICLEARINTLVL | 0xC418 | 清除中断级别寄存器 |
| SCIFLR | 0xC41C | 标志寄存器 |
| SCIINTVECT0 | 0xC420 | 中断矢量偏移寄存器 0 |
| SCIINTVECT1 | 0xC424 | 中断矢量偏移寄存器 1 |
| SCI FORMAT | 0xC428 | 长度控制寄存器 |
| BRSR | 0xC42C | 波特率选择寄存器 |
| SCI EID | 0xC430 | 仿真缓冲寄存器 |
| SCI RD | 0xC434 | 接收器数据缓冲寄存器 |
| SCI TD | 0xC438 | 发送数据缓冲寄存器 |
| SPI PO2 | 0xC43C | 引脚控制寄存器 2 |
| LIN COMP | 0xC440 | 比较寄存器 |
| LIN RD0 | 0xC444 | 接收数据寄存器 0 |
| LIN RD1 | 0xC448 | 接收数据寄存器 1 |
| LIN MASK | 0xC44C | 接收屏蔽寄存器 |
| LIN ID | 0xC450 | 寄存器包含 ID 字节，ID 从任务字节和接收到的 ID 域。 |
| LIN TD0 | 0xC454 | 发送数据寄存器 0 |
| LIN TD1 | 0xC458 | 发送数据寄存器 1 |
| MBRSR | 0xC45C | 波特率选择寄存器 |
| IODFTCTRL | 0xC460 | 用于 BLIN 的 IODFT |

注：(1) 一些寄存器和其它寄存器中的一些位是受 EALLOW 保护的。

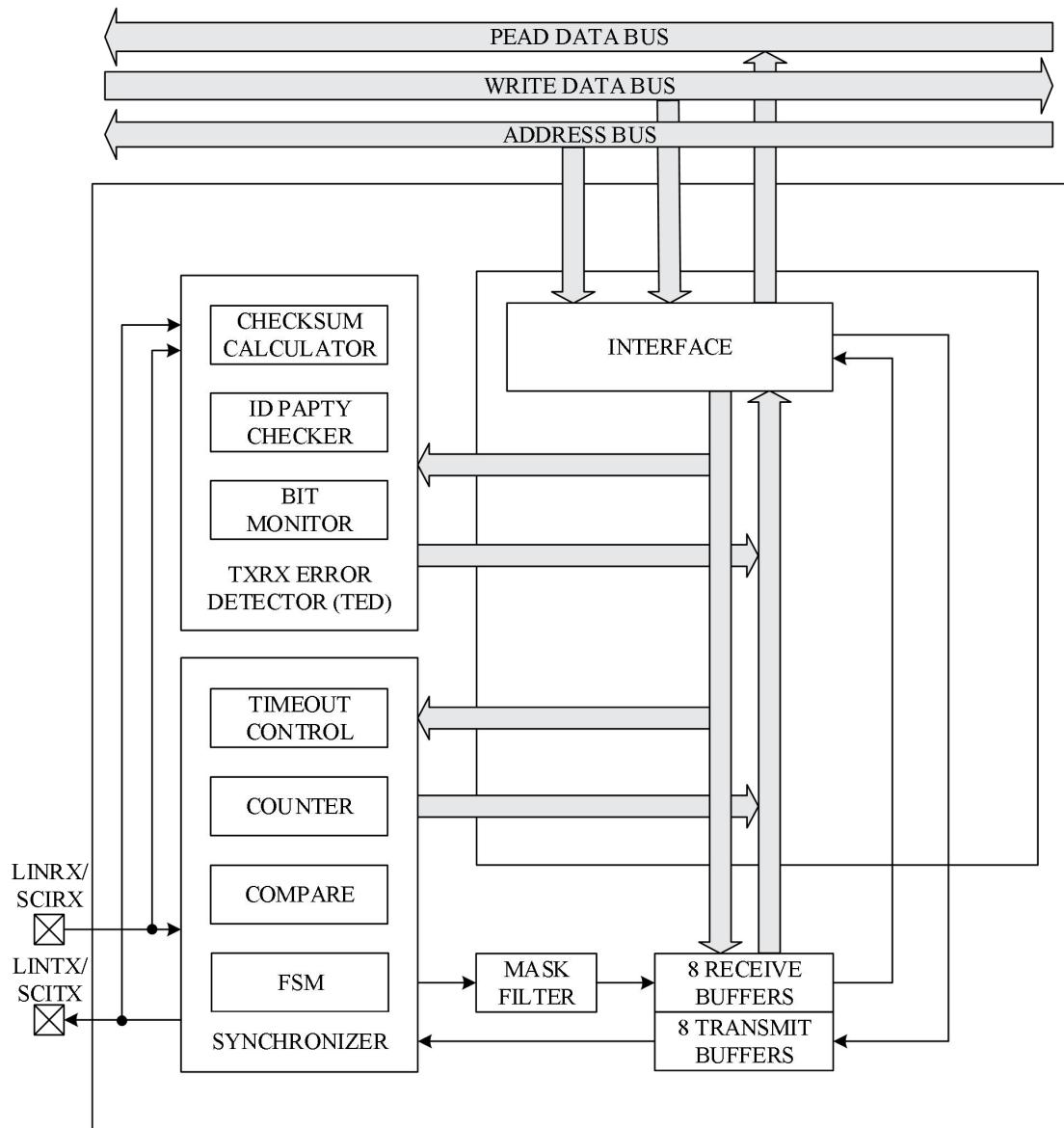


图 4-11 LIN 模块的结构图

4.5 增强型控制器局域网络 (eCAN) 模块

CAN 模块 (eCAN-A) 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
 - 有一个可编辑接收掩码

- 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
- 对接收到的新消息进行保护
- 允许动态调整发送优先级
- 具有两级中断
- 具有发送或接收超时中断
- 低功耗模式
- 总线活跃时可自动唤醒
- 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
- 32位时间戳与特定消息同步（与邮箱16 联结）
- 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。

表 4-9 eCAN 模块寄存器

| 寄存器 | 地址 | 寄存器描述 |
|--------|--------|------------|
| CANME | 0x9000 | 邮箱启用寄存器 |
| CANMD | 0x9004 | 邮箱方向寄存器 |
| CANTRS | 0x9008 | 发送请求设置存器 |
| CANTRR | 0x900C | 发送请求复位寄存器 |
| CANTA | 0x9010 | 传输确认寄存器 |
| CANAA | 0x9014 | 中止确认寄存器 |
| CANRMP | 0x9018 | 接收消息等待寄存器 |
| CANRML | 0x901C | 接收消息丢失寄存器 |
| CANRFP | 0x9020 | 远程帧等待寄存器 |
| CANGAM | 0x9024 | 全局接收屏蔽寄存器 |
| CANMC | 0x9028 | 主控制寄存器 |
| CANBTC | 0x902C | 位时序配置寄存器 |
| CANES | 0x9030 | 错误和状态寄存器 |
| CANTEC | 0x9034 | 发送错误计数器寄存器 |

| 寄存器 | 地址 | 寄存器描述 |
|---------|--------|--|
| CANREC | 0x9038 | 接收错误计数器寄存器 |
| CANGIFO | 0x903C | 全局中断标志0寄存器 |
| CANGIM | 0x9040 | 全局中断屏蔽寄存器 |
| CANGIF1 | 0x9044 | 全局中断标志1寄存器 |
| CANMIM | 0x9048 | 邮箱中断屏蔽寄存器 |
| CANMIL | 0x904C | 邮箱中断级别寄存器 |
| CANOPC | 0x9050 | 写覆盖保护控制寄存器 |
| CANTIOC | 0x9054 | TX I/O 控制寄存器 |
| CANRIOC | 0x9058 | RX I/O 控制寄存器 |
| CANTSC | 0x905C | 时间戳计数器寄存器 (保留在SCC 模式中) |
| CANTOC | 0x9060 | 超时控制寄存器 (保留在SCC 模式中) |
| CANTOS | 0x9064 | 超时状态寄存器 (保留在SCC 模式中) |
| LAM | 0x9080 | 局部接收屏蔽 (Local-Acceptance Masks) |
| MOTS | 0x9100 | 消息对象时间戳寄存器 (Message Object Time Stamp Registers) |
| MOTO | 0x9180 | 消息对象超时寄存器 (Message-Object Time-Out Registers) |
| Mailbox | 0x9200 | 信箱RAM (Mailbox RAM) |

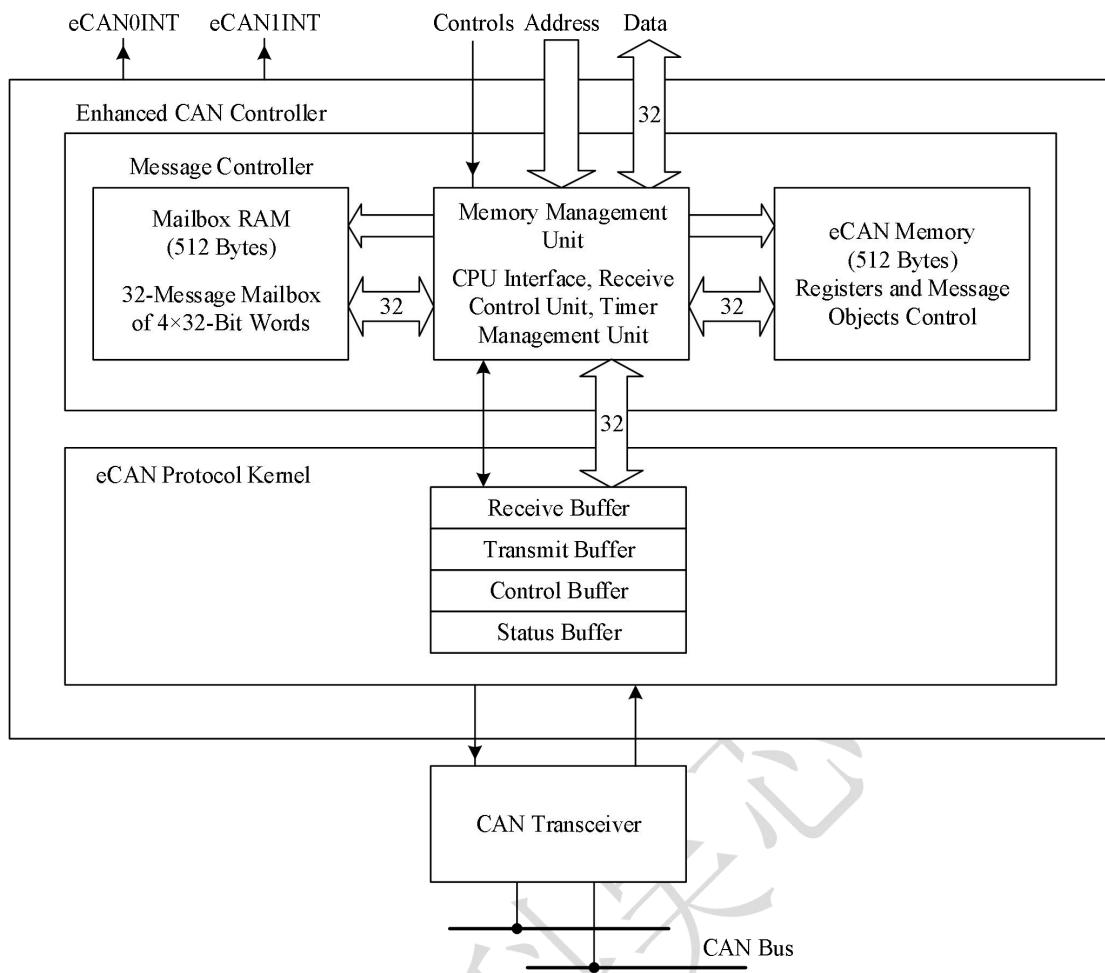


图 4-12 eCAN 模块的架构图和互连示意

4.6 内部集成电路接口 (I2C) 模块

I2C 主要功能如下：

- 兼容 NXP I2C 标准规范 v2.1;
- 支持 8bit (1~7) 字节传输
- 支持 7 位寻址和 10 位寻址
- 支持广播
- 支持 START 模式
- 支持多主机-发送器和从机-接收器模式
- 支持多从机-发送器和主机-接收器模式
- 支持主机-发送/接受和从机-接受/发送的组合
- 速率 10kbps~400kbps (快速)
- 支持 4 X 8bit 接收和发送 FIFO

- 支持两个中断：
 - I2C 中断（接收数据 ready; 发送数据 ready; 寄存器访问 ready; 未收到 ACK; 仲裁失败; 检测到停止位; 从机被寻址）
 - I2C FIFO 中断（发送 FIFO 中断; 接收 FIFO 中断）
- 模块使能和禁止
- 支持自由数据传输模式
- 不支持的功能有：
 - 高速模式
 - CBUS 模式

I2C 模块结构：

I2C 模块结构上包含以下部分：

- 串行数据接口：SDA 和 SCL;
- 数据寄存器和数据 FIFO，用于缓存 SDA 与 CPU 之间的数据;
- 控制和状态寄存器;
- 配置总线接口，用于 CPU 配置 I2C 寄存器和 FIFO;
- 噪音滤波器，过滤 SDA 和 SCL 噪声;
- 仲裁器;
- FIFO 中断产生逻辑;

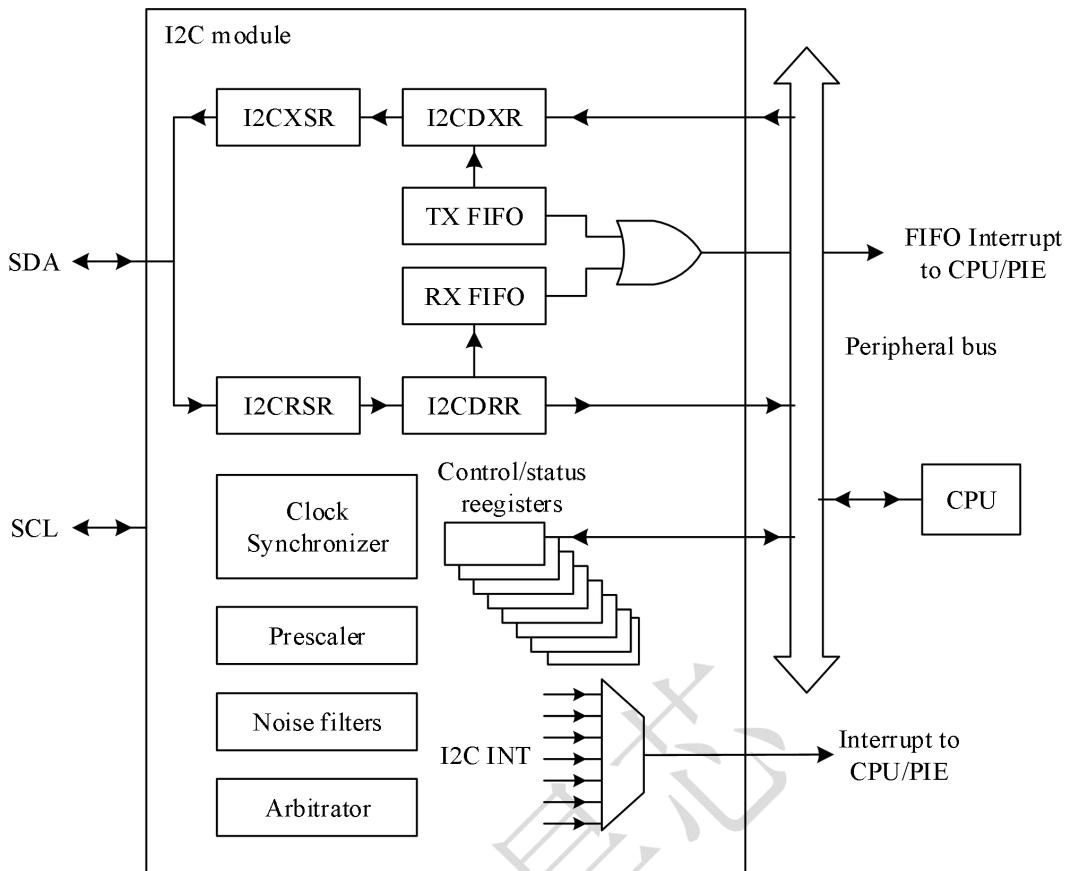


图 4-13 I2C 模块结构图

表 4-10 IIC 模块寄存器映射

| 寄存器 | 地址 | 寄存器描述 |
|---------|--------|--------------------|
| I2COAR | 0xE400 | I2C 自有地址寄存器 |
| I2CIER | 0xE404 | I2c 中断寄存器 |
| I2CSTR | 0xE408 | I2c 状态寄存器 |
| I2CCLKL | 0xE40C | I2C 时钟分频器低 32 位寄存器 |
| I2CCLKH | 0xE410 | I2C 时钟分频器高 32 位寄存器 |
| I2CCNT | 0xE414 | I2C 数据计数器寄存器 |
| I2CDRR | 0xE418 | I2C 数据接收寄存器 |
| I2CSAR | 0xE41C | I2C 从机地址寄存器 |
| I2CDXR | 0xE420 | I2C 数据传输寄存器 |
| I2CMDR | 0xE424 | I2C 模式寄存器 |
| I2CISRC | 0xE428 | I2C 中断源寄存器 |
| I2CEMDR | 0xE42C | I2C 扩展模式寄存器 |
| I2CPSC | 0xE430 | I2C 预分频寄存器 |
| I2CFFTX | 0xE434 | I2C FIFO 传输寄存器 |
| I2CFFRX | 0xE438 | I2C FIFO 接收寄存器 |

4.7 增强型 PWM 模块 (ePWM1/2/3/4/5/6/7/8)

器件包含高达 8 个增强型 PWM 模块 (ePWM)。图 4-展示了 ePWM 模块的方框图。DSC28035 器件包括 8 路增强型 PWM 模块每个 ePWM 模块支持：

- 专用 16 位 time-base 计数器，周期和频率可控；
- 每个 ePWM 模块有两个 PWM 输出 (EPWMxA, EPWMxB)，可配置为：
 - (1) 两个独立的单沿 PWM
 - (2) 两个独立的双沿对称 PWM
 - (3) 一个独立的双沿非对称 PWM
- 通过软件异步改写 PWM 信号。
- 相对于其他 ePWM 模块，可以编程对相位进行控制，从而支持滞后或超前操作。
- 逐周期地进行相位同步。
- 对上升沿和下降沿可以独立控制延迟，产生死区 (Dead Band)。
- 故障发生时，可以编程控制进行逐周期 (cycle-by-cycle) 或一次触发 (one-shot) 跳闸 (trip)。
- 跳闸能够强制 PWM 输出为高，低或高阻状态。
- 比较器模块的输出和跳闸区域的输入可以生成事件，已过滤的事件或跳闸条件。
- 所有事件都可以触发 CPU 中断和 ADC 开始转换 Start Of Conversion, SOC
- 可编程的事件预分频可最大程度地减少中断对 CPU 开销。
- 通过高频载波信号对 PWM 进行斩波。

多个 ePWM 模块通过时钟信号连接起来，通过同步机制使得多个 ePWM 模块在需要时，能够作为一个系统协同工作。另外，这种同步机制可以扩展到 eCAP 模块。模块也可以单独 (stand-alone) 运行。图 4-14 显示了与 ePWM 互连的信号。

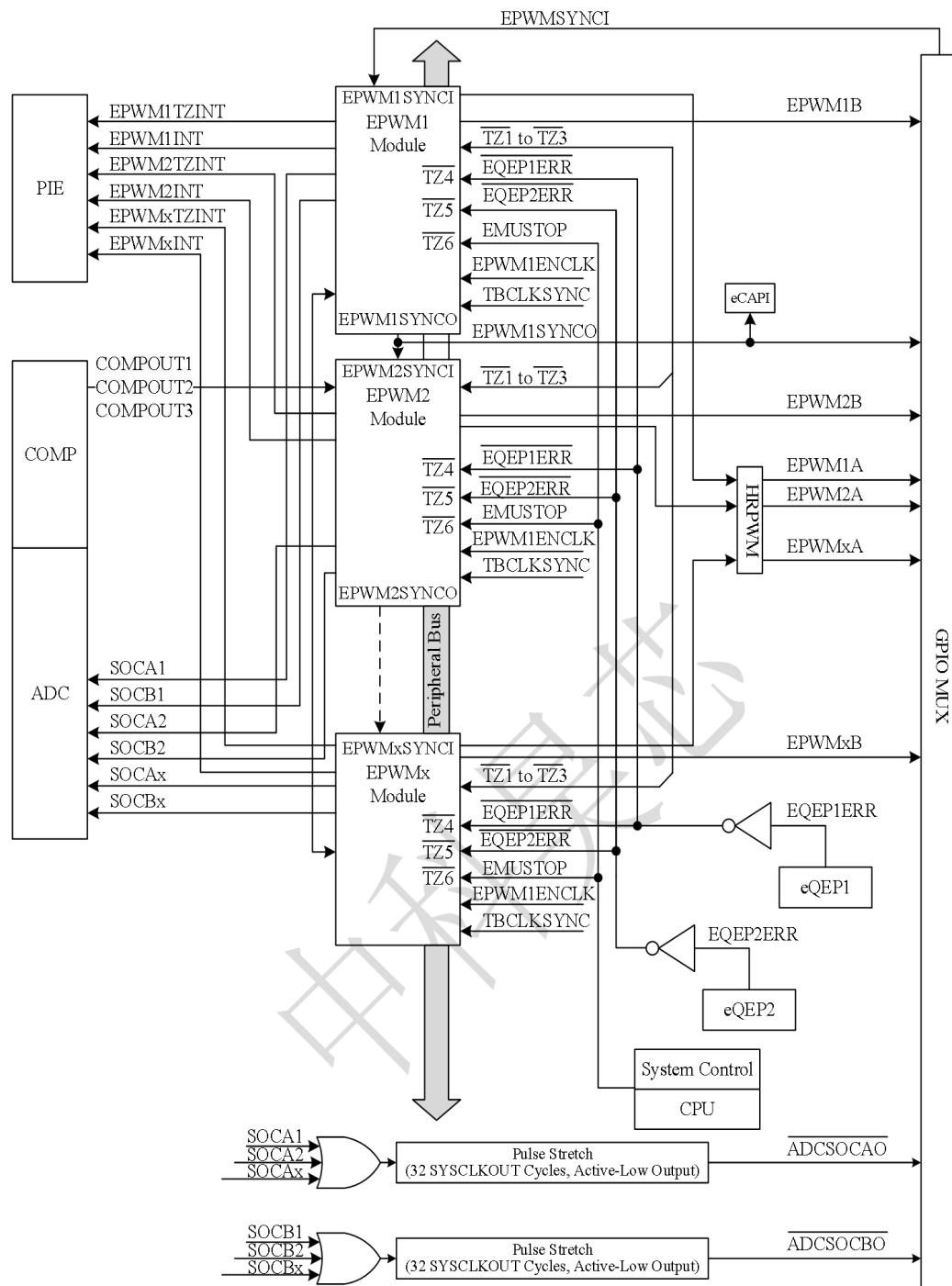


图 4-14 ePWM 模块结构图

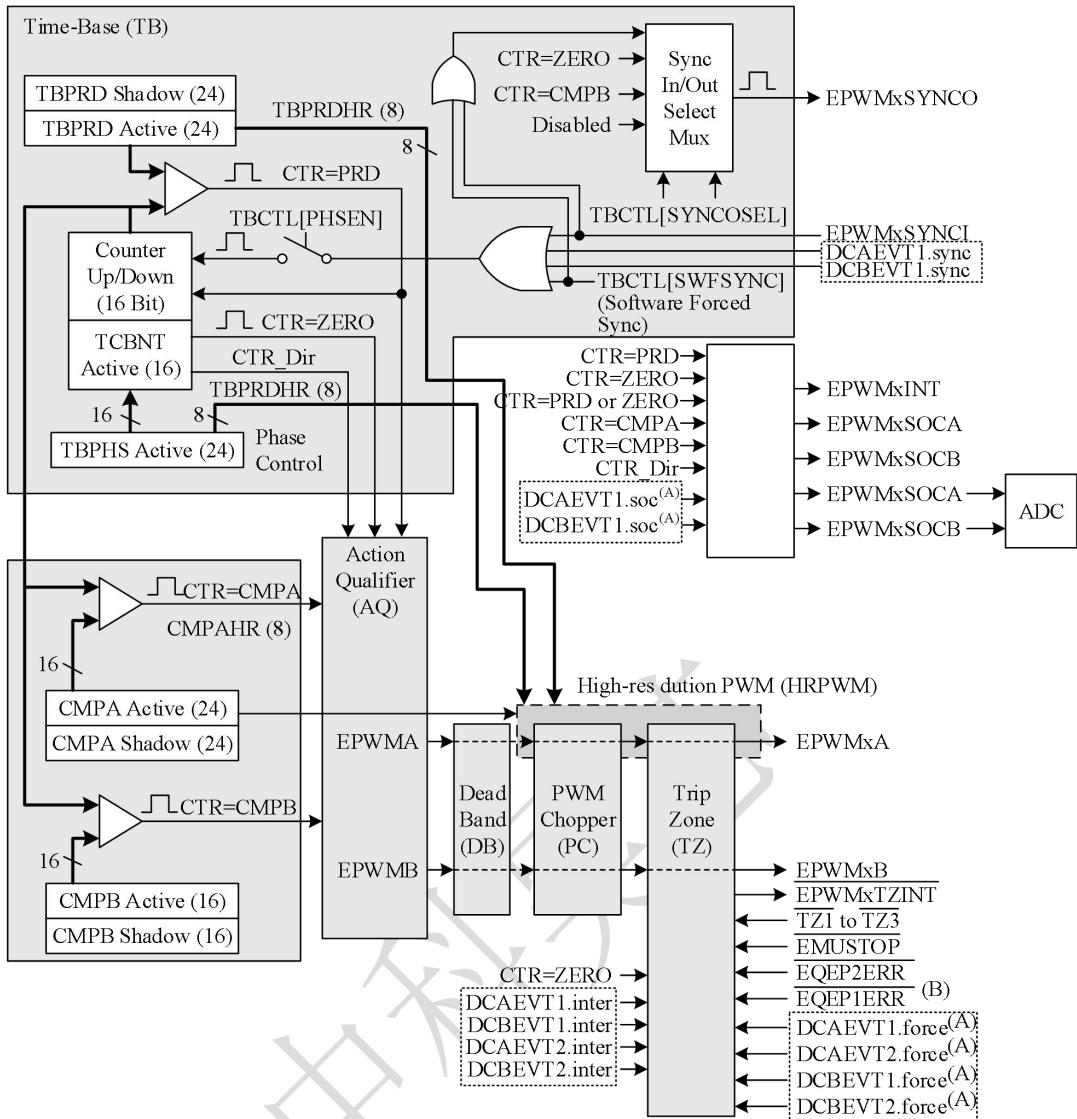


图 4-15 ePWM 子模块关键内部信号互连示意图

4.8 高分辨率 PWM 模块 (HRPWM)

该模块将多条延迟线组合在一个模块中，并使用一个简化的校准系统专用校准延迟线。每个 ePWM 模块有一条 HR 延迟线。

HRPWM 模块提供的 PWM 分辨率（时间粒度）明显优于现有的分辨率采用传统的数字 PWM 方法实现。HRPWM 模块的要点是：

- 显著扩展了传统衍生数字 PWM 的时间分辨率能力
- 此功能既可用于单边（占空比和相移控制）也可用于双边频率/周期调制的边缘控制。
- 更精细的时间粒度控制或边缘定位是通过对 CMPA 以及 ePWM 模块的相位寄存器的扩展来控制的。

在特定设备上可用时，HRPWM 功能仅在设备的 a 信号路径上提供（即，在 EPWMxA 输出上）。EPWMxB 输出具有传统的 PWM 功能。

4.9 增强型脉冲捕获模块（eCAP）

eCAP 包含以下功能：

- 4 个 32bit 事件捕获寄存器；
- 可选择 4 个沿触发采集事件；
- 每个采集事件可触发中断；
- 最大支持 4 个事件的单发射捕获；
- 支持 4 深度缓存的循环模式捕获；
- 绝对时间捕获；
- 差分模式捕获；
- 1 个信号输入；
- 不工作在捕获模式时，可配置为单通道 PWM 输出；

eCAP 主要包括以下资源：

- 专用信号输入；
- 32bit 计数器；
- 4X32bit 捕获器；
- 4 级序列发生器（mod4），与外部序列沿同步；
- 4 个捕获事件均可独立选择上下沿触发；
- 信号输入可做 $2^{~62}$ 预分频；
- 通过单发射寄存器在 1~4 捕获事件后停止捕获；
- 通过 4 个 CAP 寄存器实现连续捕获；
- 四个捕获事件均可触发中断；

eCAP 可配置为捕获模式或者 APWM 模式，APWM 模式下，cap1、cap2 分配为周期寄存器和比较寄存器，cap3 和 cap4 是其 shadow 寄存器。

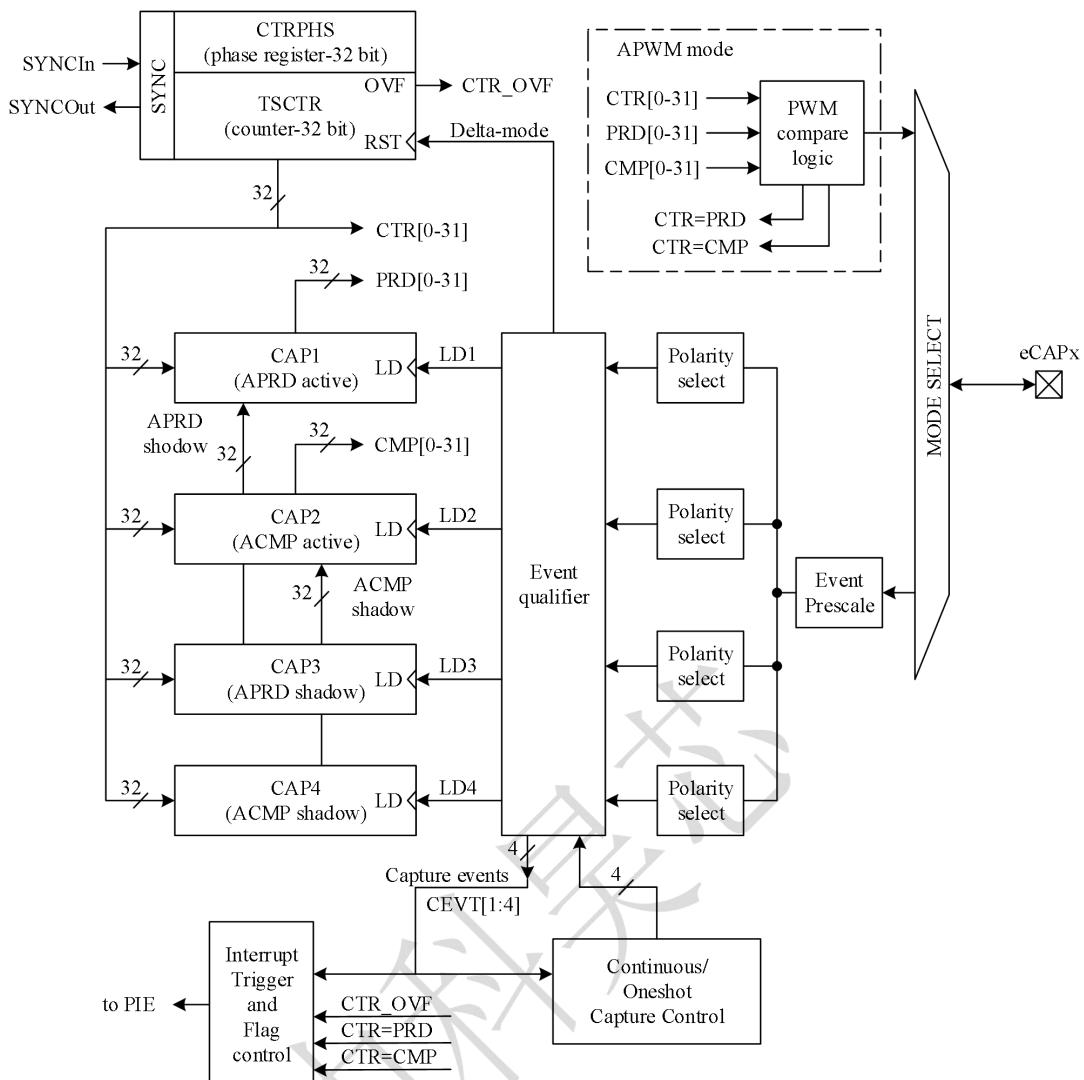


图 4-16 eCAP 功能结构图

表 4-11 eCAP 模块寄存器映射

| 寄存器 | 地址 | 寄存器描述 |
|--------|--------|-------------|
| TSCTR | 0xD000 | ECAP 时基计数器 |
| CTRPHS | 0xD004 | 计数相位寄存器 |
| CAP1 | 0xD008 | CAP 寄存器 1 |
| CAP2 | 0xD00C | CAP 寄存器 2 |
| CAP3 | 0xD010 | CAP 寄存器 3 |
| CAP4 | 0xD014 | CAP 寄存器 4 |
| ECCTL1 | 0xD018 | CAP 控制寄存器 1 |
| ECCTL2 | 0xD01C | CAP 控制寄存器 2 |
| ECEINT | 0xD020 | CAP 中断使能寄存器 |
| ECFLG | 0xD024 | CAP 中断标志寄存器 |
| ECCLR | 0xD028 | CAP 中断清除器 |
| ECFRC | 0xD02C | CAP 中断强制寄存器 |

4. 10 高分辨率脉冲捕获模块 (HRCAP)

高性能捕获 (HRCAP) 模块测量外部脉冲与一个 300ps 典型分辨率之间的差异。

HRCAP 的用途包括：

- 电容触摸应用
- 脉冲序列周期的高分辨率周期和占空比测量
- 瞬时速度测量
- 瞬时频率测量
- 在一个隔离边界上的电压测量
- 距离/回声定位测量和扫描

HRCAP 模块的特性包括：

- 在非高分辨率或者高分辨率模式中的脉宽捕获
- 差分 (Delta) 模式脉宽捕获
- 在每个边沿上 300ps 分辨率的典型高性能捕获
- 下降或者上升边沿上的中断
- 2 深度缓冲器中脉冲宽度的持续模式捕获
- 针对精准高性能捕获的校准逻辑
- 所有上述资源只用于一个单输入引脚

除了一个高分辨率校准时钟，HRCAP 模块还包括一个捕获通道，校准时，将内部连接至最后一个可用的 ePWMxA HRPWM 通道。

每一个 HRCAP 通道有下列独立的关键资源：

- 专用输入捕捉引脚
- 16 位 HRCAP 时钟等于 PLL 输出频率（与 SYSCLK 异步）或者等于 SYSCLK 频率（与 SYSCLK 同步）
- 在一个 2 级深度缓冲器中的高分辨率脉宽捕捉

表 4-12 HRCAP 寄存器

| 名称 | HRCAP1 | HRCAP2 | 寄存器描述 |
|--------|--------|--------|----------------------------|
| HCCTL | 0xCC00 | 0xCC80 | HRCAP 控制寄存器 ⁽¹⁾ |
| HCIFR | 0xCC04 | 0xCC84 | HRCAP 中断标志寄存器 |
| HCICLR | 0xCC08 | 0xCC88 | HRCAP 中断清除寄存器 |

| 名称 | HRCAP1 | HRCAP2 | 寄存器描述 |
|---------------|--------|--------|---------------------------|
| HCFIFRC | 0xCC0C | 0xCC8C | HRCAP 中断强制寄存器 |
| HCCOUNTER | 0xCC10 | 0xCC90 | HRCAP 16 位计数器寄存器 |
| HCCAL | 0xCC14 | 0xCC94 | HRCAP 校准控制寄存器 |
| HCCALMEP | 0xCC18 | 0xCC98 | HRCAP MEP 寄存器 |
| HCMEPSTATUS | 0xCC1C | 0xCC9C | HRCAP MEP 状态寄存器 |
| HCCAPCNTRISE0 | 0xCC40 | 0xCCCO | 在上升边沿 0 寄存器上的 HRCAP 捕捉计数器 |
| HCCAPCNTFALL0 | 0xCC48 | 0xCC8 | 在下降边沿 0 寄存器上的 HRCAP 捕捉计数器 |
| HCCAPCNTRISE1 | 0xCC60 | 0xCCE0 | 在上升边沿 1 寄存器上的 HRCAP 捕捉计数器 |
| HCCAPCNTFALL1 | 0xCC68 | 0xCCE8 | 在下降边沿 1 寄存器上的 HRCAP 捕捉计数器 |

注：（1）寄存器是 EALLOW 受保护的。

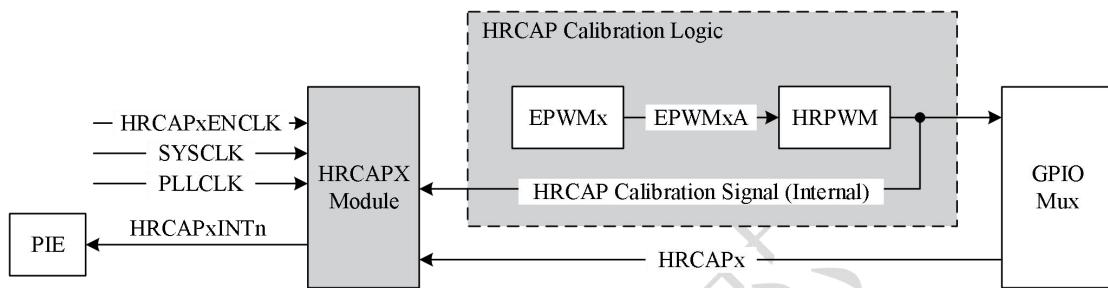


图 4-17 HRCAP 模块结构图

高分辨率捕捉 (HRCAP) 时序要求

| | | 最小值 | 标称值 | 最大值 | 单位 |
|--------------|------------------|------------------|-----|-----|----|
| $t_w(HRCAP)$ | 脉冲宽度, HRCAP捕捉的时间 | $7t_c(HCCAPCLK)$ | | | ns |

4.11 增强型正交编码器模块 (eQEP)

eQEP 主要功能单元有：

- GPIO MUX：选择端口信号（由 GPIO 实现）；
- QDU：正交解码器；
- PCCU：用于位置测量的位置计数器和控制单元；
- QCAP：用于低速测量的正交边沿捕获单元；
- UTIME：用于测量速度或者频率的单位时间单元；
- QWDG：用于停止检测的看门狗；

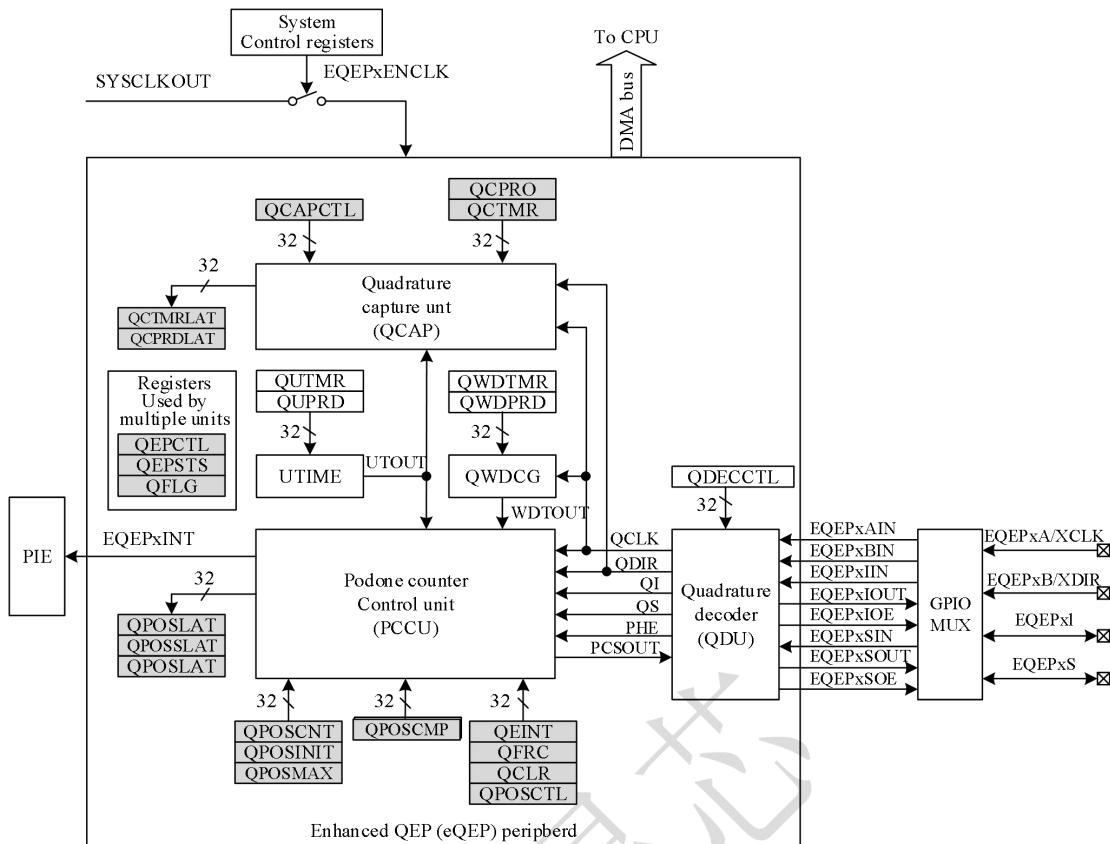


图 4-18 eQEP 结构图

表 4-13 eQEP1 模块寄存器映射

eQEP1 寄存器基址为 0x0000D400:

| 偏移 | 名称 | 描述 |
|-----|----------|--|
| 0h | QPOSCNT | 位置计数器, 32bit, 可读写, SYSRSn 下复位位 0h, 在 eqep 脉冲沿根据方向进行增计数/减计数, 表示运动的位置, 计数时该寄存器只读, 推荐在初始化时写该寄存器, QEPCTL 内的 QOPEN=0 时, 位置寄存器 Disable, 反之位置寄存器使能, 使能时向该寄存器写可能引发不可预知的问题。 |
| 4h | QPOSINIT | 位置计数器初始化寄存器, 32bit, 可读写, SYSRSn 下复位位 0h, 用于存放选通 (strobe)、索引 (index)、软件初始化位置寄存器的值。 |
| 8h | QPOSMAX | 最大位置计数寄存器, 32bit, 可读写, SYSRSn 下复位位 0h, 用于存放计数寄存器的最大值。 |
| Ch | QPOSCMP | 位置比较寄存器, 32bit, 可读写, SYSRSn 下复位位 0h, 与位置计数器的值进行比较, 匹配时产生同步输出或者中断 |
| 10h | QPOSILAT | 索引位置锁存寄存器, 32bit, 只读, SYSRSn 下复位位 0h, 根据 QEPCTL 内 IEL 定义, 在索引 (index) 事件发生时将位置计数器的值锁存于此寄存器。 |

| 偏移 | 名称 | 描述 |
|-----|----------|---|
| 14h | QPOSSLAT | 选通位置锁存寄存器, 32bit, 只读, SYSRSn 下复位位 0h, 根据 QEPCTL 内 SEL 定义, 在选通 (strobe) 事件发生时, 将位置计数器的值锁存到该寄存器。 |
| 18h | QPOSLAT | 位置计数器锁存寄存器, 32bit, 只读, SYSRSn 下复位位 0h, 在单位事件超时事件发生时, 将位置计数器的值锁存于此寄存器。 |
| 1Ch | QUTMR | 单位定时器寄存器, 32bit, 可读写, SYSRSn 下复位位 0h, 为单位事件产生提供基准, 当其值与单位周期值匹配时, 产生单位时间事件。 |
| 20h | QUprd | 单位周期寄存器, 32bit, 可读写, SYSRSn 下复位位 0h, 为单位定时器提供周期值, 用于周期性的产生单位事件, 周期性的锁存位置信息, 并产生中断。 |
| 24h | QWDTMR | 看门狗定时器寄存器, 16bit, 可读写, SYSRSn 下复位位 0h, 为看门狗提供时间基准检测电机是否停转, 当其值与看门狗周期值匹配时, 产生看门狗超时中断, 该寄存器在正交时钟边沿跳变时复位。 |
| 28h | QWDPRD | 看门狗周期寄存器, 16bit, 可读写, SYSRSn 下复位位 0h, 存放看门狗周期值, 用于产生看门狗超时中断。 |
| 2Ch | QDECCTL | 编码控制器 |
| 30h | QEPCCTL | 控制寄存器 |
| 34h | QCAPCTL | 捕获控制寄存器 |
| 38h | QPOSCTL | 位置比较控制寄存器 |
| 3Ch | QEINT | 中断使能寄存器, Bit15:12, bit0 只读为 0, Bit11-1 分别为 UTO (单元事件超时)、IEL (index 事件锁存)、SEL (strobe 事件锁存)、PCM (位置比较匹配中断)、PCR (位置比较备好)、PCO (位置计数器上溢)、PCU (位置计数器下溢出)、WTO (看门狗超时)、QDC (正交方向转向)、PHE (正交相位错误)、PCE (位置计数器错误) 等中断使能 |
| 40h | QFLG | 中断标志寄存器, Bit15:12 只读为 0, Bit11:0 分别为 UTO (单元事件超时)、IEL (index 事件锁存)、SEL (strobe 事件锁存)、PCM (位置比较匹配中断)、PCR (位置比较备好)、PCO (位置计数器上溢)、PCU (位置计数器下溢出)、WTO (看门狗超时)、QDC (正交方向转向)、PHE (正交相位错误)、PCE (位置计数器错误)、INT 等中断使能 |
| 44h | QCLR | 中断清除寄存器, 对应中断清零 |

| 偏移 | 名称 | 描述 |
|-----|----------|---|
| 48h | QFRC | 中断强制寄存器，对应中断强制为1 |
| 4Ch | QEPSTS | 状态寄存器，详见本章第五节 |
| 50h | QCTMR | 捕获计时器，16bit，可读写，SYSRSn复位为0，为沿捕获单元提供时间基准 |
| 54h | QCPRD | 捕获周期寄存器，16bit，可读写，SYSRSn复位为0，在两个连续的eqep位置事件之间，保存捕获周期计数值 |
| 58h | QCTMRLAT | 捕获定时器锁存寄存器，16bit，可读写，SYSRSn复位为0，发生单位事件超时事件或者读取位置计数器时，将捕获计数器的值锁存于此寄存器中 |
| 5Ch | QCPRDLAT | 捕获周期锁存寄存器，16bit，可读写，SYSRSn复位为0，发生单位时间超时事件或者读取位置计数器时，将捕获周期锁存于此寄存器中 |

表 4-14 eQEP2 模块寄存器映射

eQEP2 寄存器基地址为 0x0000D480:

| 偏移 | 名称 | 描述 |
|-----|----------|---|
| 0h | QPOSCNT | 位置计数器，32bit，可读写，SYSRSn下复位位0h，在eqep脉冲沿根据方向进行增计数/减计数，表示运动的位置，计数时该寄存器只读，推荐在初始化时写该寄存器，QEPCTL内的QOPEN=0时，位置寄存器Disable，反之位置寄存器使能，使能时向该寄存器写可能引发不可预知的问题。 |
| 4h | QPOSINIT | 位置计数器初始化寄存器，32bit，可读写，SYSRSn下复位位0h，用于存放选通(strobe)、索引(index)、软件初始化位置寄存器的值。 |
| 8h | QPOSMAX | 最大位置计数寄存器，32bit，可读写，SYSRSn下复位位0h，用于存放计数寄存器的最大值。 |
| Ch | QPOSCMP | 位置比较寄存器，32bit，可读写，SYSRSn下复位位0h，与位置计数器的值进行比较，匹配时产生同步输出或者中断 |
| 10h | QPOSILAT | 索引位置锁存寄存器，32bit，只读，SYSRSn下复位位0h，根据QEPCTL内IEL定义，在索引(index)事件发生时将位置计数器的值锁存于此寄存器。 |
| 14h | QPOSSLAT | 选通位置锁存寄存器，32bit，只读，SYSRSn下复位位0h，根据QEPCTL内SEL定义，在选通(strobe)事件发生时，将位置计数器的值锁存到该寄存器。 |
| 18h | QPOSLAT | 位置计数器锁存寄存器，32bit，只读，SYSRSn下复位位0h，在单位事件 |

| 偏移 | 名称 | 描述 |
|-----|---------|--|
| | | 超时事件发生时，将位置计数器的值锁存于此寄存器。 |
| 1Ch | QUTMR | 单位定时器寄存器，32bit，可读写，SYSRSn 下复位位 0h，为单位事件产生提供基准，当其值与单位周期值匹配时，产生单位时间事件。 |
| 20h | QUPRD | 单位周期寄存器，32bit，可读写，SYSRSn 下复位位 0h，为单位定时器提供周期值，用于周期性的产生单位事件，周期性的锁存位置信息，并产生中断。 |
| 24h | QWDTMR | 看门狗定时器寄存器，16bit，可读写，SYSRSn 下复位位 0h，为看门狗提供时间基准检测电机是否停转，当其值与看门狗周期值匹配时，产生看门狗超时中断，该寄存器在正交时钟边沿跳变时复位。 |
| 28h | QWDPRD | 看门狗周期寄存器，16bit，可读写，SYSRSn 下复位位 0h，存放看门狗周期值，用于产生看门狗超时中断。 |
| 2Ch | QDECCTL | 编码控制器 |
| 30h | QEPCCTL | 控制寄存器 |
| 34h | QCAPCTL | 捕获控制寄存器 |
| 38h | QPOSCTL | 位置比较控制寄存器 |
| 3Ch | QEINT | 中断使能寄存器，Bit15:12，bit0 只读为 0，Bit11-1 分别为 UTO（单元事件超时）、IEL（index 事件锁存）、SEL（strobe 事件锁存）、PCM（位置比较匹配中断）、PCR（位置比较备好）、PCO（位置计数器上溢）、PCU（位置计数器下溢出）、WTO（看门狗超时）、QDC（正交方向转向）、PHE（正交相位错误）、PCE（位置计数器错误）等中断使能 |
| 40h | QFLG | 中断标志寄存器，Bit15:12 只读为 0，Bit11:0 分别为 UTO（单元事件超时）、IEL（index 事件锁存）、SEL（strobe 事件锁存）、PCM（位置比较匹配中断）、PCR（位置比较备好）、PCO（位置计数器上溢）、PCU（位置计数器下溢出）、WTO（看门狗超时）、QDC（正交方向转向）、PHE（正交相位错误）、PCE（位置计数器错误）、INT 等中断使能 |
| 44h | QCLR | 中断清除寄存器，对应中断清零 |
| 48h | QFRC | 中断强制寄存器，对应中断强制为 1 |
| 4Ch | QEPCSTS | 状态寄存器 |
| 50h | QCTMR | 捕获计时器，16bit，可读写，SYSRSn 复位为 0，为沿捕获单元提供时间基准 |

| 偏移 | 名称 | 描述 |
|-----|----------|--|
| 54h | QCPRD | 捕获周期寄存器, 16bit, 可读写, SYSRSn 复位为 0, 在两个连续的 eqep 位置事件之间, 保存捕获周期计数值 |
| 58h | QCTMRLAT | 捕获定时器锁存寄存器, 16bit, 可读写, SYSRSn 复位为 0, 发生单位事件超时事件或者读取位置计数器时, 将捕获计数器的值锁存于此寄存器中 |
| 5Ch | QCPRDLAT | 捕获周期锁存寄存器, 16bit, 可读写, SYSRSn 复位为 0, 发生单位时间超时事件或者读取位置计数器时, 将捕获周期锁存于此寄存器中 |

4. 12JTAG 端口

JTAG 有 5 个端口: TRST, TCK, TDI, TMS, TDO, 其中 TCK, TDI, TMS 及 TDO 同时作为 GPIO 端口, $\overline{\text{TRST}}$ 选择要么作为 JTAG 端口要么作为 GPIO 端口, 如下图所示。

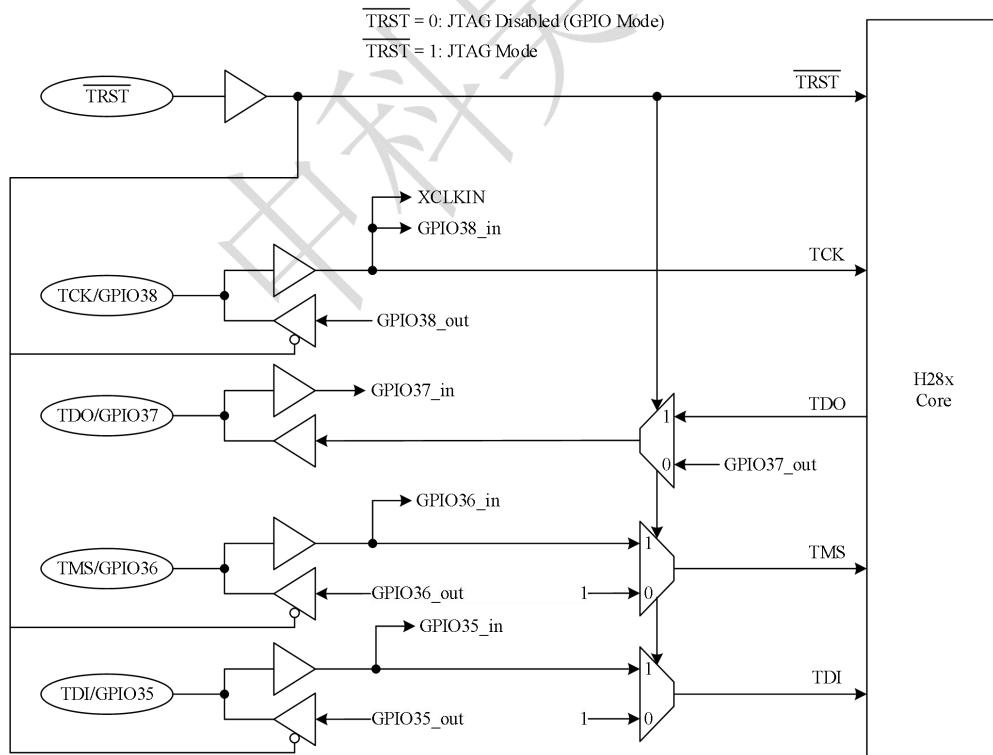


图 4-19 JTAG 端口多路复用示意图

注意: JTAG 端口也可以作为 GPIO 端口, 在板级设计时, 需注意连接这些端口的电路不能影响 JTAG 仿真, 不能影响仿真器驱动这些端口进行调试。

TRST 信号结合 JTAG DEBUG 寄存器内的 JTADIS 位选择 JTAG 的功能。

| TRST | JTAGDIS bit | JTAG Port Mode |
|------|-------------|---|
| 0 | X | GPIO mode enabled, JTAG port disabled |
| 1 | 0 | JTAG port enabled(GPIOs should be configured as inputs) |
| 1 | 1 | GPIO mode enabled, JTAG port disabled |

其中 JTAG DEBUG 寄存器地址为 0xDC44，受 EALLOW 保护。

JTAG DEBUG 寄存器位宽 32bits, bit[15:1]保留只读为 0, bit0 为 JTADIS- JTAG disable bit, 0- JTAG enable, 1-JTAG disable, 当 JTAG disable 时, JATG 端口可用于 GPIO 功能, 当 TRST=0 时, JTADIS 被复位为 0, 当 TRST=1 时, 可以通过 CPU 配置 JTADIS。

4.13 GPIO 控制寄存器

GPIO 可以复用于片内的外设，并具有上下拉控制、输入信号滤波等功能。

GPIO MUX 寄存器

表 4-15 GPIO 寄存器映射及说明

| 名称 | 地址 | 描述 |
|------------------------|--------|----------------------------------|
| GPIO 控制寄存器 (EALLOW 保护) | | |
| GPACTRL | 0xD810 | GPIO A 控制寄存器 (GPIO0 到 31) |
| GPAQSEL1 | 0xD818 | GPIO A 限定符选择 1 寄存器 (GPIO0 到 15) |
| GPAQSEL2 | 0xD81C | GPIO A 限定符选择 2 寄存器 (GPIO16 到 31) |
| GPAMUX1 | 0xD800 | GPIO A MUX 1 寄存器 (GPIO0 到 15) |
| GPAMUX2 | 0xD804 | GPIO A MUX 2 寄存器 (GPIO0 到 15) |
| GPADIR | 0xD824 | GPIOA 方向寄存器 (GPIO0 到 31) |
| GPAPUD | 0xD830 | GPIO A 上拉禁用寄存器 (GPIO0 到 31) |
| GPBCTRL | 0xD814 | GPIO B 控制寄存器 (GPIO32 到 44) |
| GPBQSEL1 | 0xD820 | GPIO B 限定符选择 1 寄存器 (GPIO32 到 44) |
| GPBMUX1 | 0xD808 | GPIO B MUX 1 寄存器 (GPIO32 到 44) |

| 名称 | 地址 | 描述 |
|--------------------------------|--------|----------------------------------|
| GPBDIR | 0xD828 | GPIO B 方向寄存器 (GPIO32 到 44) |
| GPBPUD | 0xD834 | GPIO B 上拉禁用寄存器 (GPIO32 到 44) |
| AIOMUX1 | 0xD80C | 模拟, I/O mux 1 寄存器 (AI00 到 AI015) |
| AIODIR | 0xD82C | 模拟, I/O 方向寄存器 (AI00 到 AI015) |
| GPIO 数据寄存器 (非 EALLOW 保护) | | |
| GPADAT | 0xD838 | GPIO A 数据寄存器 (GPIO0 到 31) |
| GPASET | 0xD844 | GPIO A 数据置位寄存器 (GPIO0 到 31) |
| GPACLEAR | 0xD848 | GPIO A 数据清除寄存器 (GPIO0 到 31) |
| GPATOGGLE | 0xD84C | GPIO A 数据翻转寄存器 (GPIO0 到 31) |
| GPBDAT | 0xD83C | GPIO B 数据寄存器 (GPIO32 到 44) |
| GPBSET | 0xD850 | GPIO B 数据置位寄存器 (GPIO32 到 44) |
| GPBCLEAR | 0xD854 | GPIO B 数据清除寄存器 (GPIO32 到 44) |
| GPBToggle | 0xD858 | GPIO B 数据翻转寄存器 (GPIO32 到 44) |
| AIODAT | 0xD840 | 模拟 I/O 数据寄存器 (AI00 到 AI015) |
| AIOSET | 0xD85C | 模拟 I/O 数据置位寄存器 (AI00 到 AI015) |
| AIOCLEAR | 0xD860 | 模拟 I/O 数据清除寄存器 (AI00 到 AI015) |
| AIOTOGGLE | 0xD864 | 模拟 I/O 数据翻转寄存器 (AI00 到 AI015) |
| GPIO 中断和低功耗模式选择寄存器 (EALLOW 保护) | | |
| GPIOXINT1SEL | 0xD868 | XINT1 GPIO 输入选择寄存器 (GPIO0 到 31) |
| GPIOXINT2SEL | 0xD86C | XINT2 GPIO 输入选择寄存器 (GPIO0 到 31) |
| GPIOXINT3SEL | 0xD870 | XINT3 GPIO 输入选择寄存器 (GPIO0 到 31) |
| GPIOLPMSEL | 0xD874 | LPM GPIO 选择寄存器 (GPIO0 到 31) |

表 4-16 GPIOA MUX 选择

| | 复位默认功能 | 外设 1 | 外设 2 | 外设 3 |
|-----------|------------|------------|---------------|---------------|
| GPAMUX1 位 | GPAMUX1=00 | GPAMUX1=01 | GPAMUX1=10 | GPAMUX1=11 |
| 1-0 | GPIO0 | EPWM1A (O) | Reserved | Reserved |
| 3-2 | GPIO1 | EPWM1B (O) | Reserved | COMP1OUT (O) |
| 5-4 | GPIO2 | EPWM2A (O) | Reserved | Reserved |
| 7-6 | GPIO3 | EPWM2B (O) | SPISOMIA(I/O) | COMP2OUT (O) |
| 9-8 | GPIO4 | EPWM3A (O) | Reserved | Reserved |
| 11-10 | GPIO5 | EPWM3B (O) | SPISIMOA(I/O) | ECAP1 (I/O) |
| 13-12 | GPIO6 | EPWM4A (O) | EPWMSYNCI (I) | EPWMSYNCO (O) |
| 15-14 | GPIO7 | EPWM4B (O) | SCIRXDA (I) | Reserved |

| | 复位默认功能 | 外设 1 | 外设 2 | 外设 3 |
|-----------|---------------|----------------|--------------|----------------|
| 17–16 | GPIO8 | EPWM5A (O) | Reserved | ADCSOCAO (O) |
| 19–18 | GPIO9 | EPWM5B (O) | LINTXA (O) | HRCAP1 (I) |
| 21–20 | GPIO10 | EPWM6A (O) | Reserved | ADCSOCBO (O) |
| 23–22 | GPIO11 | EPWM6B (O) | LINRXA (I) | HRCAP2 |
| 25–24 | GPIO12 | TZ1 (I) | SCITXDA (O) | SPISIMOB (I/O) |
| 27–26 | GPIO13 | TZ2 (I) | Reserved | SPISOMIB (I/O) |
| 29–28 | GPIO14 | TZ3 (I) | LINTXA (O) | SPICLKB (I/O) |
| 31–30 | GPIO15 | TZ1 (I) | LINRXA (I) | SPISTEB (I/O) |
| GPAMUX2 位 | GPAMUX2=00 | GPAMUX2=01 | GPAMUX2=10 | GPAMUX2=11 |
| 1–0 | GPIO16 | SPISIMOA (I/O) | Reserved | TZ2 (I) |
| 3–2 | GPIO17 | SPISOMIA (I/O) | Reserved | TZ3 (I) |
| 5–4 | GPIO18 | SPICLKA (I/O) | LINTXA (O) | XCLKOUT (O) |
| 7–6 | GPIO19/XCLKIN | SPISTEA (I/O) | LINRXA (I) | ECAP1 (I/O) |
| 9–8 | GPIO20 | EQEP1A (I) | Reserved | CMP1OUT (O) |
| 11–10 | GPIO21 | EQEP1B (I) | Reserved | CMP2OUT (O) |
| 13–12 | GPIO22 | EQEP1S (I/O) | Reserved | LINTXA (O) |
| 15–14 | GPIO23 | EQEP11 (I/O) | Reserved | LINRXA (I) |
| 17–16 | GPIO24 | ECAP1 (I/O) | EQEP2A (I) | SPISIMOB (I/O) |
| 19–18 | GPIO25 | Reserved | EQEP2B (I) | SPISOMIB (I/O) |
| 21–20 | GPIO26 | HRCAP1 (I) | EQEP2S (I/O) | SPICLKB (I/O) |
| 23–22 | GPIO27 | HRCAP2 (I) | EQEP2I (I/O) | SPISTEB (I/O) |
| 25–24 | GPIO28 | SCIRXDA (I) | SDAA (I/OD) | TZ2 (I) |
| 27–26 | GPIO29 | SCITXDA (O) | SCLA (I/OD) | TZ3 (I) |
| 29–28 | GPIO30 | CANRXA (I) | Reserved | Reserved |
| 31–30 | GPIO31 | CANTXA (O) | Reserved | Reserved |

表 4-17 GPIOB MUX 选择

| | 复位默认功能 | 外设 1 | 外设 2 | 外设 3 |
|-----------|---------------------|--------------|---------------|--------------|
| GPBMUX1 位 | GPBMUX1=00 | GPBMUX1=01 | GPBMUX1=10 | GPBMUX1=11 |
| 1–0 | GPIO32 | SDAA (I/OD) | EPWMSYNCI (I) | ADCSOCAO (O) |
| 3–2 | GPIO33 | SCLA ((I/OD) | EPWMSYNC0 (O) | ADCSOCBO (O) |
| 5–4 | GPIO34 | COMP2OUT (O) | Reserved | COMP3OUT (O) |
| 7–6 | GPIO35 (TDI) | Reserved | Reserved | Reserved |
| 9–8 | GPIO36 (TMS) | Reserved | Reserved | Reserved |
| 11–10 | GPIO37 (TDO) | Reserved | Reserved | Reserved |
| 13–12 | GPIO38/XCLKIN (TCK) | Reserved | Reserved | Reserved |
| 15–14 | GPIO39 | EPWM8A (O) | Reserved | Reserved |
| 17–16 | GPIO40 | EPWM7A (O) | Reserved | Reserved |
| 18–19 | GPIO41 | EPWM7B (O) | Reserved | Reserved |
| 21–20 | GPIO42 | Reserved | Reserved | COMP1OUT (O) |
| 23–22 | GPIO43 | Reserved | Reserved | COMP2OUT (O) |

| | 复位默认功能 | 外设 1 | 外设 2 | 外设 3 |
|-------|----------|-----------|----------|----------|
| 25–24 | GPIO44 | EPWM8B(0) | Reserved | Reserved |
| 27–26 | Reserved | Reserved | Reserved | Reserved |
| 29–28 | Reserved | Reserved | Reserved | Reserved |
| 31–30 | Reserved | Reserved | Reserved | Reserved |

表 4-18 模拟 IO MUX 选择

| AIOMUX1 位 | AIOMUX1=0, x | AIOMUX1=1, x |
|-----------|----------------------|-------------------------|
| 1–0 | ADCINA0 (I) (VREFHI) | ADCINA0 (I) (VREFHI) |
| 3–2 | ADCINA1 (I) | ADCINA1 (I) |
| 5–4 | AI02 (I/O) | ADCINA2 (I), COMP1A (I) |
| 7–6 | ADCINA3 (I) | ADCINA3 (I) |
| 9–8 | AI04 (I/O) | ADCINA4 (I), COMP2A (I) |
| 11–10 | ADCINA5 (I) | ADCINA5 (I) |
| 13–12 | AI06 (I/O) | ADCINA6 (I), COMP3A (I) |
| 15–14 | ADCINA7 (I) | ADCINA7 (I) |
| 17–16 | ADCINB0 (I) | ADCINB0 (I) |
| 19–18 | ADCINB1 (I) | ADCINB1 (I) |
| 21–20 | AI010 (I/O) | ADCINB2 (I), COMP1B (I) |
| 23–22 | ADCINB3 (I) | ADCINB3 (I) |
| 25–24 | AI012 (I/O) | ADCINB4 (I), COMP2B (I) |
| 27–26 | ADCINB5 (I) | ADCINB5 (I) |
| 29–28 | AI014 (I/O) | ADCINB6 (I), COMP3B (I) |
| 31–30 | ADCINB7 (I) | ADCINB7 (I) |

用户可以通过 GPxQSEL1/2 寄存器从四个选项中选择每个 GPIO 引脚的输入限定类型：

仅同步到 SYSCLKOUT (GPxQSEL1/2=0, 0)：这是复位时所有 GPIO 引脚的默认模式，它只是将输入信号同步到系统时钟 (SYSCLKOUT)。

- 使用采样窗口 (GPxQSEL1/2=0, 1 和 1, 0) 进行确认：在这种模式下，输入信号在与系统时钟 (SYSCLKOUT) 同步后，在允许更改输入之前，通过指定的周期数进行确认。
- 采样周期由 GPxCTRL 寄存器中的 QUALPRD 位指定，可配置为 8 个信号组。它指定用于对输入信号进行采样的 SYSCLKOUT 周期的倍数。采样窗口为 3 个采样或 6 个采样宽，只有当所有采样相同（所有 0 或所有 1）时，输出才会更改，如图 6-42 所示（对于 6 采样模式）。
- 无同步 (GPxQSEL1/2=1, 1)：此模式用于不需要同步的外设（同步在外设内执行）。

由于设备上需要多级复用，可能存在外围输入信号可以映射到多个 GPIO 管脚的情况。另外，当没有选择输入信号时，根据外围设备的不同，输入信号将默认为 0 或 1 状态。

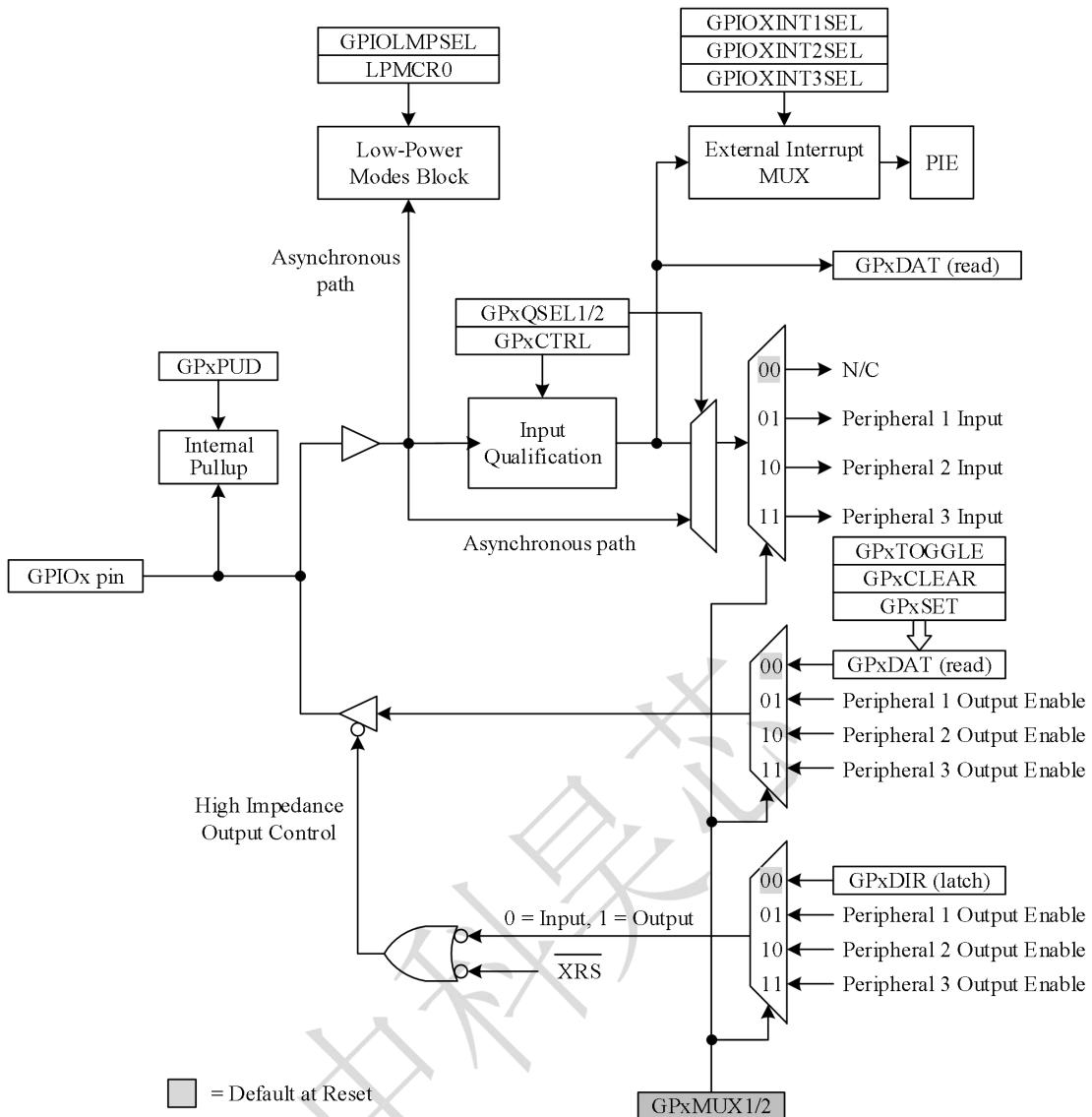


图 4-20 GPIO 复用模块连接示意图

- x 代表端口 A 或 B。例如, GPxDIR 指的是 GPADIR 和 GPBDIR 寄存器取决于所选的特定 GPIO 引脚。
- 在同一内存位置访问 GPxDAT 锁存/读取。
- 这是一个通用的 GPIO MUX 框图。并非所有选项都适用于所有 GPIO 引脚。

4.14 直接存储访问存储单元 (DMA)

4.14.1 DMA

28035 内 DMA 基本功能，其触发源包括：

- 6 通道，每个通道有独立的 PIE；
- 外设中断触发源有：

- ADC sequencer1 和 sequencer2;
- XINT1–XINT3;
- CPU timers
- ePWM1–8 ADCSOCA 和 ADCSOCB
- SPI/SCI/I2C
- 软件

数据源/目的可以是下列模块:

- L0/L1 SRAM
- M0/M1 SRAM
- ADC 结果寄存器
- ePWM1–4/HRPWM1–4 寄存器
- SPI
- SCI
- I2C
- 数据宽度: 32bit
- 吞吐率: 4 cycle/word

表 4-13 DMA 寄存器列表

| 偏移地址 | 寄存器名称 | 功能描述 |
|-------|---------------------|-------------|
| 0000h | DMACTRL | DMA控制寄存器 |
| 0004h | DEBUGCTRL | 调试控制寄存器 |
| 0008h | REVISION | 外设修改寄存器 |
| 000Ch | PRIORITYCTRL1 | 优先级控制寄存器1 |
| 0010h | PRIORITYSTAT | 优先级状态寄存器 |
| 0080h | MODE_0 | 模式寄存器 |
| 0084h | CONTROL_0 | 控制寄存器 |
| 0088h | BURST_SIZE_0 | 突发大小寄存器 |
| 008Ch | BURST_COUNT_0 | 突发计数寄存器 |
| 0090h | SRC_BURST_STEP_0 | 源突发步长大小寄存器 |
| 0094h | DST_BURST_STEP_0 | 目的突发步长大小寄存器 |
| 0098h | TRANSFER_SIZE_0 | 传输大小寄存器 |
| 009Ch | TRANSFER_COUNT_0 | 传输计数寄存器 |
| 00A0h | SRC_TRANSFER_STEP_0 | 源传输步长寄存器 |
| 00A4h | DST_TRANSFER_STEP_0 | 目的传输步长寄存器 |
| 00A8h | SRC_WRAP_SIZE_0 | 源换行步长寄存器 |
| 00ACh | SRC_WRAP_COUNT_0 | 源换行计数寄存器 |

| 偏移地址 | 寄存器名称 | 功能描述 |
|-------|-----------------------|------------------|
| 00B0h | SRC_WRAP_STEP_0 | 源换行步长大小寄存器 |
| 00B4h | DST_WRAP_SIZE_0 | 目标换行步长大小寄存器 |
| 00B8h | DST_WRAP_COUNT_0 | 目标换行计数寄存器 |
| 00BCh | DST_WRAP_STEP_0 | 目标换行步长寄存器 |
| 00C0h | SRC_BEG_ADDR_SHADOW_0 | 影子源开始与当前地址指针寄存器 |
| 00C4h | SRC_ADDR_SHADOW_0 | 影子目标开始与当前地址指针寄存器 |
| 00C8h | SRC_BEG_ADDR_0 | 活动源开始与当前地址指针寄存器 |
| 00CCh | SRC_ADDR_0 | 活动目标开始与当前地址指针寄存器 |
| 00D0h | DST_BEG_ADDR_SHADOW_0 | 影子源开始与当前地址指针寄存器 |
| 00D4h | DST_ADDR_SHADOW_0 | 影子目标开始与当前地址指针寄存器 |
| 00D8h | DST_BEG_ADDR_0 | 活动目标开始与当前地址指针寄存器 |
| 00DCh | DST_ADDR_0 | 活动目标开始与当前地址指针寄存器 |
| 0100h | MODE_1 | 模式寄存器1 |
| 0104h | CONTROL_1 | 控制寄存器2 |
| 0108h | BURST_SIZE_1 | 突发大小寄存器 |
| 010Ch | BURST_COUNT_1 | 突发计数寄存器 |
| 0110h | SRC_BURST_STEP_1 | 源突发步长大小寄存器 |
| 0114h | DST_BURST_STEP_1 | 目标突发步长大小寄存器 |
| 0118h | TRANSFER_SIZE_1 | 传输大小寄存器 |
| 011Ch | TRANSFER_COUNT_1 | 传输计数寄存器 |
| 0120h | SRC_TRANSFER_STEP_1 | 源传输步长大小寄存器 |
| 0124h | DST_TRANSFER_STEP_1 | 目标传输步长大小寄存器 |
| 0128h | SRC_WRAP_SIZE_1 | 源换行大小寄存器 |
| 012Ch | SRC_WRAP_COUNT_1 | 源换行计数寄存器 |
| 0130h | SRC_WRAP_STEP_1 | 源换行步长大小寄存器 |
| 0134h | DST_WRAP_SIZE_1 | 目标换行步长大小寄存器 |
| 0138h | DST_WRAP_COUNT_1 | 目标换行计数寄存器 |
| 013Ch | DST_WRAP_STEP_1 | 目标换行步长寄存器 |
| 0140h | SRC_BEG_ADDR_SHADOW_1 | 影子源开始与当前地址指针寄存器 |
| 0144h | SRC_ADDR_SHADOW_1 | 影子目标开始与当前地址指针寄存器 |
| 0148h | SRC_BEG_ADDR_1 | 活动源开始与当前地址指针寄存器 |
| 014Ch | SRC_ADDR_1 | 活动目标开始与当前地址指针寄存器 |
| 0150h | DST_BEG_ADDR_SHADOW_1 | 影子源开始与当前地址指针寄存器 |
| 0154h | DST_ADDR_SHADOW_1 | 影子目标开始与当前地址指针寄存器 |
| 0158h | DST_BEG_ADDR_1 | 活动目标开始与当前地址指针寄存器 |
| 015Ch | DST_ADDR_1 | 活动目标开始与当前地址指针寄存器 |
| 0180h | MODE_2 | 模式寄存器 |
| 0184h | CONTROL_2 | 控制寄存器 |

| 偏移地址 | 寄存器名称 | 功能描述 |
|-------|-----------------------|------------------|
| 0188h | BURST_SIZE_2 | 突发大小寄存器 |
| 018Ch | BURST_COUNT_2 | 突发计数寄存器 |
| 0190h | SRC_BURST_STEP_2 | 源突发步长大小寄存器 |
| 0194h | DST_BURST_STEP_2 | 目标突发步长大小寄存器 |
| 0198h | TRANSFER_SIZE_2 | 传输大小寄存器 |
| 019Ch | TRANSFER_COUNT_2 | 传输计数寄存器 |
| 01A0h | SRC_TRANSFER_STEP_2 | 源传输步长大小寄存器 |
| 01A4h | DST_TRANSFER_STEP_2 | 目标传输步长大小寄存器 |
| 01A8h | SRC_WRAP_SIZE_2 | 源换行大小寄存器 |
| 01ACh | SRC_WRAP_COUNT_2 | 源换行计数寄存器 |
| 01B0h | SRC_WRAP_STEP_2 | 源换行步长大小寄存器 |
| 01B4h | DST_WRAP_SIZE_2 | 目标换行步长大小寄存器 |
| 01B8h | DST_WRAP_COUNT_2 | 目的换行计数寄存器 |
| 01BCh | DST_WRAP_STEP_2 | 目标换行步长大小寄存器 |
| 01C0h | SRC_BEG_ADDR_SHADOW_2 | 影子源开始与当前地址指针寄存器 |
| 01C4h | SRC_ADDR_SHADOW_2 | 影子目标开始与当前地址指针寄存器 |
| 01C8h | SRC_BEG_ADDR_2 | 活动源开始与当前地址指针寄存器 |
| 01CCh | SRC_ADDR_2 | 活动目标开始与当前地址指针寄存器 |
| 01D0h | DST_BEG_ADDR_SHADOW_2 | 影子目标开始与当前地址指针寄存器 |
| 01D4h | DST_ADDR_SHADOW_2 | 影子目标开始与当前地址指针寄存器 |
| 01D8h | DST_BEG_ADDR_2 | 活动目标开始与当前地址指针寄存器 |
| 01DCh | DST_ADDR_2 | 活动目标开始与当前地址指针寄存器 |
| 0200h | MODE_3 | 模式寄存器 |
| 0204h | CONTROL_3 | 控制寄存器 |
| 0208h | BURST_SIZE_3 | 突发大小寄存器 |
| 020Ch | BURST_COUNT_3 | 突发计数寄存器 |
| 0210h | SRC_BURST_STEP_3 | 源突发步长大小寄存器 |
| 0214h | DST_BURST_STEP_3 | 目标突发步长大小寄存器 |
| 0218h | TRANSFER_SIZE_3 | 传输大小寄存器 |
| 021Ch | TRANSFER_COUNT_3 | 传输计数寄存器 |
| 0220h | SRC_TRANSFER_STEP_3 | 源传输步长大小寄存器 |
| 0224h | DST_TRANSFER_STEP_3 | 目标传输步长大小寄存器 |
| 0228h | SRC_WRAP_SIZE_3 | 源换行大小寄存器 |
| 022Ch | SRC_WRAP_COUNT_3 | 源换行计数寄存器 |
| 0230h | SRC_WRAP_STEP_3 | 源换行步长大小寄存器 |
| 0234h | DST_WRAP_SIZE_3 | 目标换行步长大小寄存器 |
| 0238h | DST_WRAP_COUNT_3 | 目标换行计数寄存器 |
| 023Ch | DST_WRAP_STEP_3 | 目标换行步长寄存器 |

| 偏移地址 | 寄存器名称 | 功能描述 |
|-------|-----------------------|------------------|
| 0240h | SRC_BEG_ADDR_SHADOW_3 | 影子源开始与当前地址指针寄存器 |
| 0244h | SRC_ADDR_SHADOW_3 | 影子目标开始与当前地址指针寄存器 |
| 0248h | SRC_BEG_ADDR_3 | 活动源开始与当前地址指针寄存器 |
| 024Ch | SRC_ADDR_3 | 活动目标开始与当前地址指针寄存器 |
| 0250h | DST_BEG_ADDR_SHADOW_3 | 影子目标开始与当前地址指针寄存器 |
| 0254h | DST_ADDR_SHADOW_3 | 影子目标开始与当前地址指针寄存器 |
| 0258h | DST_BEG_ADDR_3 | 活动目标开始与当前地址指针寄存器 |
| 025Ch | DST_ADDR_3 | 活动目标开始与当前地址指针寄存器 |
| 0280h | MODE_4 | 模式寄存器 |
| 0284h | CONTROL_4 | 控制寄存器 |
| 0288h | BURST_SIZE_4 | 突发大小寄存器 |
| 028Ch | BURST_COUNT_4 | 突发计数寄存器 |
| 0290h | SRC_BURST_STEP_4 | 源突发步长大小寄存器 |
| 0294h | DST_BURST_STEP_4 | 目标突发步长大小寄存器 |
| 0298h | TRANSFER_SIZE_4 | 传输大小寄存器 |
| 029Ch | TRANSFER_COUNT_4 | 传输计数寄存器 |
| 02A0h | SRC_TRANSFER_STEP_4 | 源突发步长大小寄存器 |
| 02A4h | DST_TRANSFER_STEP_4 | 目标突发步长大小寄存器 |
| 02A8h | SRC_WRAP_SIZE_4 | 源换行大小寄存器 |
| 0ACh | SRC_WRAP_COUNT_4 | 源换行计数寄存器 |
| 02B0h | SRC_WRAP_STEP_4 | 源换行步长大小寄存器 |
| 02B4h | DST_WRAP_SIZE_4 | 目标换行步长大小寄存器 |
| 02B8h | DST_WRAP_COUNT_4 | 目标换行计数寄存器 |
| 02BCh | DST_WRAP_STEP_4 | 目标换行步长寄存器 |
| 02C0h | SRC_BEG_ADDR_SHADOW_4 | 影子源开始与当前地址指针寄存器 |
| 02C4h | SRC_ADDR_SHADOW_4 | 影子目标开始与当前地址指针寄存器 |
| 02C8h | SRC_BEG_ADDR_4 | 活动源开始与当前地址指针寄存器 |
| 02CCh | SRC_ADDR_4 | 活动目标开始与当前地址指针寄存器 |
| 02D0h | DST_BEG_ADDR_SHADOW_4 | 影子目标开始与当前地址指针寄存器 |
| 02D4h | DST_ADDR_SHADOW_4 | 影子目标开始与当前地址指针寄存器 |
| 02D8h | DST_BEG_ADDR_4 | 活动目标开始与当前地址指针寄存器 |
| 02DCh | DST_ADDR_4 | 活动目标开始与当前地址指针寄存器 |
| 0300h | MODE_5 | 模式寄存器 |
| 0304h | CONTROL_5 | 控制寄存器 |
| 0308h | BURST_SIZE_5 | 突发大小寄存器 |
| 030Ch | BURST_COUNT_5 | 突发计数寄存器 |
| 0310h | SRC_BURST_STEP_5 | 源突发步长大小寄存器 |
| 0314h | DST_BURST_STEP_5 | 目标突发步长大小寄存器 |

| 偏移地址 | 寄存器名称 | 功能描述 |
|-------|-----------------------|------------------|
| 0318h | TRANSFER_SIZE_5 | 传输大小寄存器 |
| 031Ch | TRANSFER_COUNT_5 | 传输计数寄存器 |
| 0320h | SRC_TRANSFER_STEP_5 | 源突发步长大小寄存器 |
| 0324h | DST_TRANSFER_STEP_5 | 目标突发步长大小寄存器 |
| 0328h | SRC_WRAP_SIZE_5 | 源换行大小寄存器 |
| 032Ch | SRC_WRAP_COUNT_5 | 源换行计数寄存器 |
| 0330h | SRC_WRAP_STEP_5 | 源换行步长大小寄存器 |
| 0334h | DST_WRAP_SIZE_5 | 目标换行步长大小寄存器 |
| 0338h | DST_WRAP_COUNT_5 | 目标换行计数寄存器 |
| 033Ch | DST_WRAP_STEP_5 | 目标换行步长寄存器 |
| 0340h | SRC_BEG_ADDR_SHADOW_5 | 影子目标开始与当前地址指针寄存器 |
| 0344h | SRC_ADDR_SHADOW_5 | 影子目标开始与当前地址指针寄存器 |
| 0348h | SRC_BEG_ADDR_5 | 活动目标开始与当前地址指针寄存器 |
| 034Ch | SRC_ADDR_5 | 活动目标开始与当前地址指针寄存器 |
| 0350h | DST_BEG_ADDR_SHADOW_5 | 活动目标开始与当前地址指针寄存器 |
| 0354h | DST_ADDR_SHADOW_5 | 活动目标开始与当前地址指针寄存器 |
| 0358h | DST_BEG_ADDR_5 | 活动目标开始与当前地址指针寄存器 |
| 035Ch | DST_ADDR_5 | 活动目标开始与当前地址指针寄存器 |

4.15 可编程控制律加速器 (CLA)

CLA 是一个独立的、完全可编程的 32 位浮点数学处理器, H28x 系列的并发控制循环执行。CLA 的低中断延迟允许其读取 ADC 采样 “及时” 。这显著减少了 ADC 采样到输出的延迟, 以实现更快的系统响应和更高 MHz 的控制回路。通过使用 CLA 服务时间关键型控制回路, 主 CPU 是可以自由执行其他系统任务, 例如通信和诊断。

控制律加速器通过添加并行处理扩展了 H28xCPU 的功能。时间关键型 CLA 服务的控制环路可以实现低 ADC 采样到输出延迟。因此, CLA 能够更快地实现系统响应和更高频率的控制回路。将 CLA 用于时间关键型任务 CPU 同时执行其他系统和通信功能。

以下是 CLA 的主要特征列表:

- 以与主 CPU (SYSCLKOUT) 相同的速率计时。
- 独立架构, 允许独立于主 H28x CPU 执行 CLA 算法。
 - 完整的总线架构:

- IFU 和 LSU 总线
- 完整的一套数据总线
 - 独立的 8 级流水线。
 - 16 位程序计数器 (MPC)
 - 通用的 32 位定点和 32 位浮点寄存器各 32 个
 - 通用的 32 位定点和 32 位浮点寄存器各 32 个
 - 状态寄存器 (MSTF)
 - 返回程序寄存器 RPC (Return Program Counter)

• 指令集包括：

- IEEE 单精度 (32 位) 浮点数学运算
- $1/X$ 和 $1/\sqrt{X}$ 估计值
- 数据类型转换
- 条件分支和调用
- 数据加载/存储操作

• CLA 程序代码可包含多达八个任务或中断服务例程，或七个任务和一个主要后台任务。

- 每个任务的起始地址由 MVECT 寄存器指定。
- 只要任务符合可配置 CLA 程序内存空间，就不限制任务大小。
- 一次完成一项任务。没有任务嵌套。
- 任务完成后，PIE 中会标记任务特定中断。
- 当任务完成时，下一个最高优先级的待定任务将自动启动。
- 2 类 CLA 可以有一个在后台连续运行的主任务，而其他高优先级任务

事件触发前台任务。

- 任务触发机制:

- 任务 1 至任务 8: 来自连接到共享总线的外围设备的多达 256 个可能的触发源 CLA 承担次要所有权。
- 任务 8 可以设置为后台任务, 而任务 1 至 7 采用外围触发。

- 内存和共享外设:

- 两个专用消息 RAM, 用于 CLA 和主 CPU 之间的通信。
- H28x CPU 可以将 CLA 程序和数据内存映射到主 CPU 空间或 CLA 空间。

5. 开发支持

所有软件均可从中科昊芯官方网站 (<http://www.haawking.com/download>) 下载到最新版本。

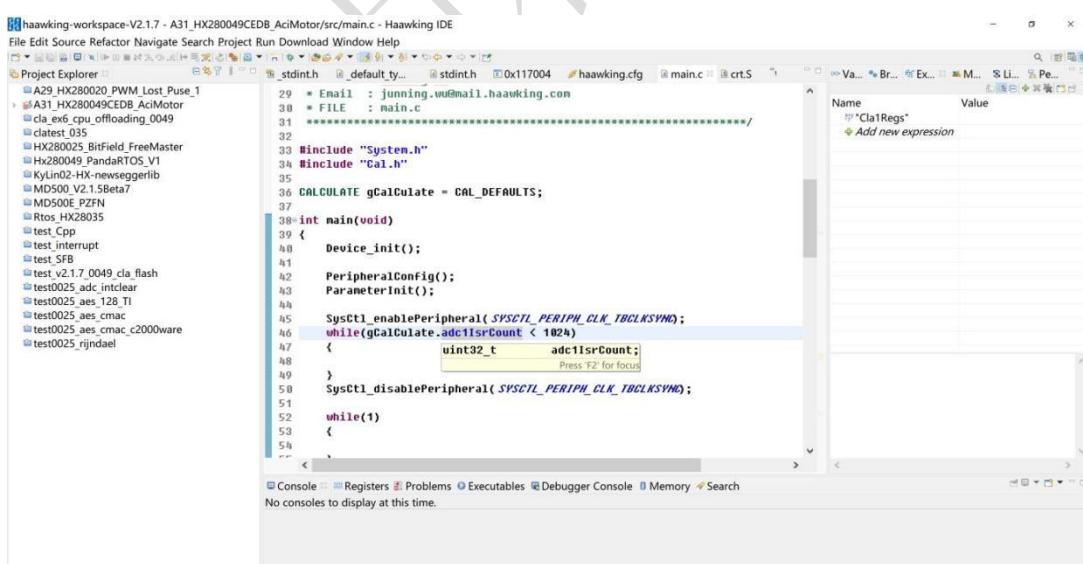
5.1 集成开发环境 Haawking IDE

用户可以从中科昊芯官方网站 (<http://www.haawking.cn/download-soft>) 下载到最新版本的 Haawking IDE 软件和相关的驱动库。首次使用，请下载完整包（Haawking-IDE-win64-Vx.x.x.exe），包括编译器、JRE 环境、HAAWING LINK 驱动以及 OpenOCD 等工具。

Haawking IDE 启动界面如下图所示：



Haawking IDE 运行界面如下图所示：



具体使用和操作指南，欢迎查看 Haawking IDE 安装目录下的说明文档《Haawking-IDE User's Guide》。

5.2 仿真器

5.2.1 简介

HAAWING LINK 110V1-ISO 是一款小巧便携、安全可靠、低成本的仿真器，支持 HX 中科昊芯的 HXS320F28035 系列 DSP 产品。

5.2.2 概览

HAAWING LINK 110V1-ISO 仿真器调试框图如下图所示。

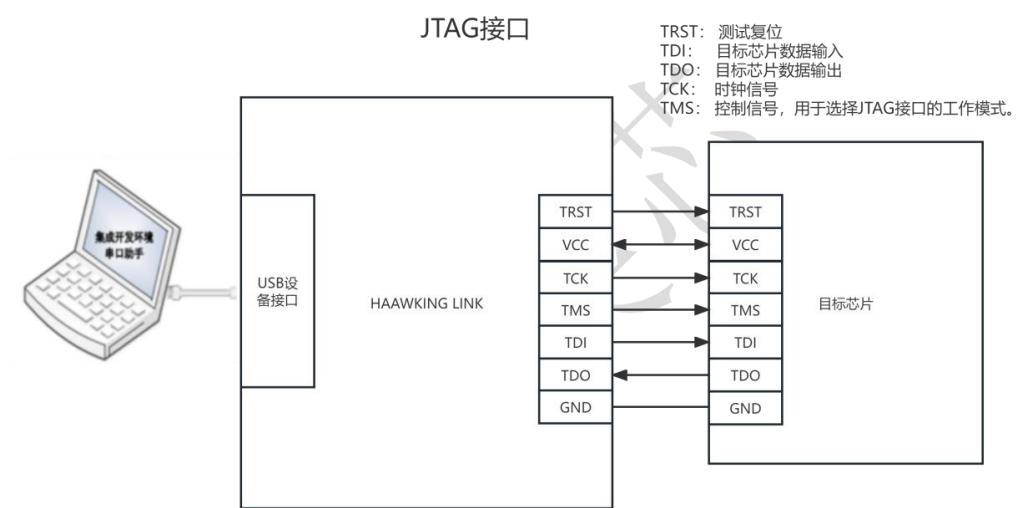


图 8-4. Haawking Link 连接示意图 (JTAG 接口)

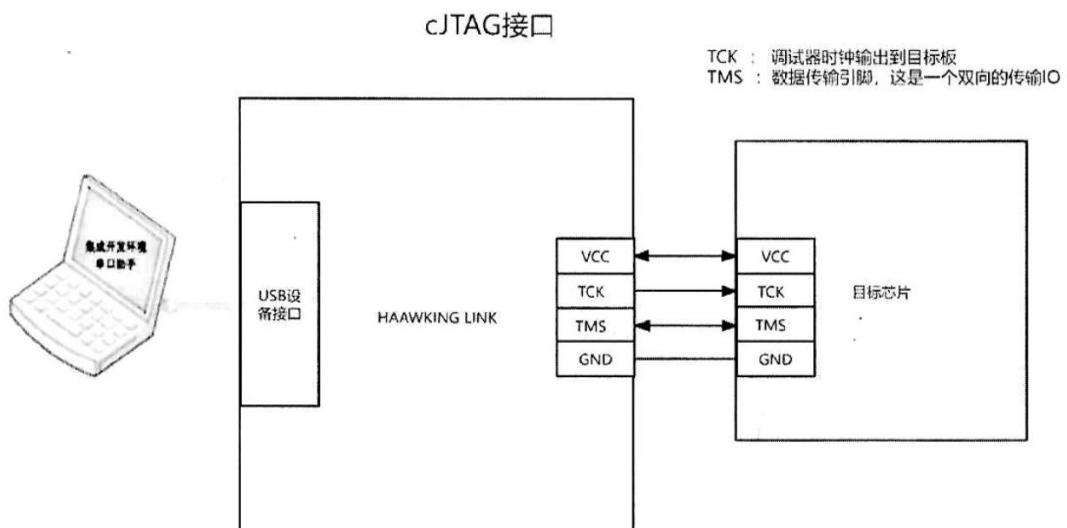


图 8-5. Haawking Link连接示意图 (cJTAG接口)

5.2.2.1 特性

HAAWING LINK 110V1-ISO仿真器具有如下特点：在线调试的功能 — 即插即用，在Windows10及以上系统不需安装驱动，可直接在Haawking IDE集成开发环境下调试，调试信号接口和USB侧电气隔离。

HAAWING LINK 110V1-ISO仿真器正面如图所示，内含状态指示灯。



表 8-1. Haawking Link 引脚说明

| 引脚号 | 引脚名 | 功能 | 备注 |
|-----|------|------------|--|
| 1 | TMS | 控制信号 | 用于选择 JTAG 接口的工作模式。通过改变 TMS 的状态，可以切换 JTAG 接口的状态和操作。此引脚默认上拉。 |
| 2 | TRST | Test Reset | 测试复位信号，此引脚默认下拉。 |
| 3 | TDI | 目标板信号输入 | 仿真器连接至目标芯片的数据输入信号，此引脚默认上拉。 |
| 4 | GND | 地 | 连接目标芯片接地引脚。 |
| 5 | VCC | 电源输入 | 为芯片供电。 |
| 6 | NC | 悬空 | |
| 7 | TDO | 目标板数据输出 | 芯片返回给仿真器的数据信号，此引脚默认上拉。 |
| 8 | GND | 地 | 连接目标芯片接地引脚。 |
| 9 | TCK | 时钟信号 | 时钟信号，此引脚默认上拉。 |
| 10 | GND | 地 | 连接目标芯片接地引脚。 |
| 11 | TCK | 时钟信号 | 时钟信号，此引脚默认上拉。 |
| 12 | GND | 地 | 连接目标芯片接地引脚。 |
| 13 | NC | 悬空 | |
| 14 | NC | 悬空 | |

5.3 串口下载器

Haawking-Downloader 是中科昊芯自研串口下载工具，可以用来烧写片内闪存。用户可以从中科昊芯官方网站（<http://www.haawking.cn/download-soft>）下载到最新版本的软件。串口下载工具支持中科昊芯自研 HX2000 系列的产品程序烧录，用户只需要提供 Haawking IDE 编译生成的 elf 或 hex 文件即可。

支持的环境有：WIN XP、WIN7(win7 64 位版本如果提示错误，需要补充安装 Windows6.1-KB3033929-x64.msu 补丁，可通过百度等进行下载)、WIN8、WIN10、WIN11。支持 JTAG、cJTAG 和 SCI 三种连接方式进行下载，用户只需要提供 Haawking IDE 编译生成的 elf 或 hex 文件即可。

具体使用和操作指南，欢迎查看安装目录下的说明文档《Haawking_Downloader 用户手册》。

5.4 离线下载器

Haawking-Writer 是中科昊芯自研离线下载工具，用户可以用来升级芯片固件。用户可以从中科昊芯官方网站（<http://www.haawking.cn/download-soft>）下载到最新版本的软件。

支持的环境有：WIN XP、WIN7（win7 64 位版本如果提示错误，需要补充安装 Windows6.1-KB3033929-x64.msu 补丁，可通过百度等进行下载）、WIN8、WIN10 都可使用。支持 cJTAG 模式和 JTAG 模式下载。

具体使用和操作指南，欢迎查看安装目录下的说明文档《Haawking_Writer 用户手册》。

6. 电气规范

6.1 绝对最大额定值

表 6-1 HXS320F28035 芯片绝对最大额定值

| | |
|--|-----------------|
| 电源电压范围, V_{DDIO} , (I/O 和闪存) | -0.3V 至 3.63V |
| 电源电压范围, V_{DD} | -0.3V 至 2.5V |
| 模拟电压范围, V_{DDA} | -0.3V 至 3.63V |
| 输入电压范围, V_{IN} (3.3V) | -0.3V 至 3.63V |
| 输出电压范围, V_O | -0.3V 至 3.63V |
| 输入钳制电流, I_{IK} ($V_{IN} < 0$ 或者 $V_{IN} > V_{DDIO}$) | ±20mA |
| 输出钳制电流, I_{OK} ($V_O < 0$ 或者 $V_O > V_{DDIO}$) | ±20mA |
| 贮存温度范围, T_{stg} | -65° C 至 150° C |
| 结温, T_J | -40° C 至 150° C |

注: 1. 上述数据仅供参考。

(1) 长期高温存储或在最高温度条件下长期使用可能会导致设备寿命的缩短。

6.2 推荐工作条件

表 6-2 HXS320F28035 芯片建议运行条件

| | | 最小值 | 典型值 | 最大值 | 单位 |
|---|----------------|----------------|------|------------------|-----|
| 器件电源电压, I/O, V_{DDIO} | | 2.97 | 3.3 | 3.63 | V |
| 器件电源电压 CPU, VDD 当内部 VREG 被禁用, 使用外部 1.5V 电压) | | 1.55 | 1.57 | 1.582 | V |
| 电源接地, V_{SS} | | | 0 | | V |
| 模拟电源电压, V_{DDA} | | 2.97 | 3.3 | 3.63 | V |
| 模拟接地, V_{SSA} | | | 0 | | V |
| $f_{SYSCLKOUT}$ 器件时钟频率 (系统时钟) | | | | 120 | MHz |
| 高电平输入电压, V_{IH} (3.3V) | | 2 | | $V_{DDIO} + 0.3$ | V |
| 低电平输入电压, V_{IL} (3.3V) | | $V_{SS} - 0.3$ | | 0.8 | V |
| 高电平输出拉电流, $V_{OH} = V_{IH}$ (最小 值), I_{OH} | 所有 GPIO/AIO 引脚 | | | -4 | mA |
| | 组 2 | | | -8 | |

| | | | | | |
|--------------------------------------|----------------|-----|--|-----|-----|
| 低电平输出灌电流 $V_{OL}=V_{OL}$ 最大值, IOL | 所有 GPIO/AIO 引脚 | | | 4 | mA |
| | 组 2 | | | 8 | |
| 环境温度, T_A | | -40 | | 105 | ° C |
| | | -40 | | 125 | |
| 结温, T_J | | -40 | | 150 | ° C |

注:

1. 上述数据仅供参考。
2. 组 2: GPIO16, GPIO17, GPIO18, GPIO19, GPIO28, GPIO29, GPIO36, GPIO37

6.3 ESD 额定值

表 6-3 HXS320F28035 芯片 ESD 额定值

| HXS320F28035 80 引脚 PN 封装 | | 数值 | 单位 |
|----------------------------------|--|-------|----|
| HXS320F28035 | | | |
| $V_{(ESD)}$ 静电放电 | 人体模型 (HBM) , 符合 ESDA/JEDEC JS-001-2017 ⁽¹⁾ | ±2000 | V |
| | 充电装置型号 (CDM) , 符合 ESDA/JEDEC JS-002-2018 ⁽²⁾ | ±500 | V |
| HXS320F28035 64 引脚 PM 封装 | | | |
| $V_{(ESD)}$ 静电放电 | 人体模型 (HBM) , 符合 ESDA/JEDEC JS-001-2017 ⁽¹⁾ | ±2000 | V |
| | 充电装置型号 (CDM) , 符合 ESDA/JEDEC JS-002-2018 ⁽²⁾ | ±500 | V |
| HXS320F28035 56 引脚 RSH 封装 | | | |
| $V_{(ESD)}$ 静电放电 | 人体模型 (HBM) , 符合 ESDA/JEDEC JS-001-2017 ⁽¹⁾ | ±2000 | V |
| | 充电装置型号 (CDM) , 符合 ESDA/JEDEC JS-002-2018 ⁽²⁾ | ±500 | V |

注:

1. JEDEC 文件 JEP155 规定, 500-V HBM 允许通过标准 ESD 控制工艺进行安全制造。
2. JEDEC 文件 JEP157 指出, 250-V CDM 允许采用标准 ESD 控制工艺进行安全制造。
3. 上述数据仅供参考。

6.4 电气特性

表 6-4 HXS320F28035 芯片电气特性

| 参数 | 测试条件 | | 最小值 | 典型值 | 最大值 | 单位 | |
|---------------------------|-------------------------------------|----------------------------------|--------------------|------|---------|---------|---------|
| V_{OH} 高电平输出电压 | $I_{OH}=I_{OL}$ 最大值 | | 2.4 | | | V | |
| | $I_{OH}=50 \mu A$ | | $V_{DDIO}=0.$ 2 | | | | |
| V_{OL} 低电平输出电压 | $I_{OL}=I_{OH}$ 最大值 | | | | 0.4 | V | |
| I_{IL} 输入电流 (低电平) | 带有上拉电 阻器 | $V_{DDIO}=3.3V,$ $V_{IN}=0V$ | 所有 GPIO | -70 | -150 | -225 | μA |
| | | XRS 引脚 | | -120 | -220 | -325 | |
| I_{IH} 输入电流 (高电平) | 带有下拉电 阻器 | $V_{DDIO}=3.3V, V_{IN}=0V$ | | | | ± 2 | μA |
| | | $V_{DDIO}=3.3V, V_{IN}=V_{DDIO}$ | | 20 | 34 | 60 | |
| IOZ 输出电流, 高阻抗状态 (关闭状态) | $V_O=V_{DDIO}$ 或者 0V | | | | ± 2 | μA | |
| C_i 输入电容 | | | | 2 | | pF | |
| $V_{DDIOBOR}$ 触发点 | 下降的 V_{DDIO} | | | 2.78 | | V | |
| $V_{DDIOBOR}$ 滞后 | | | | 35 | | mV | |
| 监视器复位延迟时间 | 延迟时间过后, BOR/POR/OVR 事件被移除以释放 XRS | | | 500 | | μs | |
| VREG V_{DD} 输出 | 内部 VREG 打开 | | | 1.9 | | V | |
| 片载 LDO 输出 | | | | 1.51 | | V | |

注:

- 上述数据仅供参考。

6.5 功耗

表 6-5 HXS320F28035 芯片功耗

| 模式 | 测试条件 | VREG 被启用 (vregenz=0) | | | | | | VREG 被禁用 (vregenz=1) | | | |
|--------------|-------------------------------------|----------------------|-----|------------|-----|-----------|-----|----------------------|-----|-----------|-----|
| | | I_VDD(2) | | I_VDDIO(1) | | I_VDDA(2) | | I_VDDIO(1) | | I_VDDA(2) | |
| | | 典型值(3) | 最大值 | 典型值(3) | 最大值 | 典型值(3) | 最大值 | 典型值(3) | 最大值 | 典型值(3) | 最大值 |
| IDLE 空闲模式 | 闪存休眠; XCLKOUT 被关闭; 所有外设时钟被关闭。 | 8mA | | 11mA | | 200uA | | 18mA | | 200uA | |
| HALT 停机模式 | 闪存休眠; 外设时钟被关闭; 输入时钟被禁用。 | 2mA | | 4mA | | 200uA | | 5mA | | 200uA | |

注:

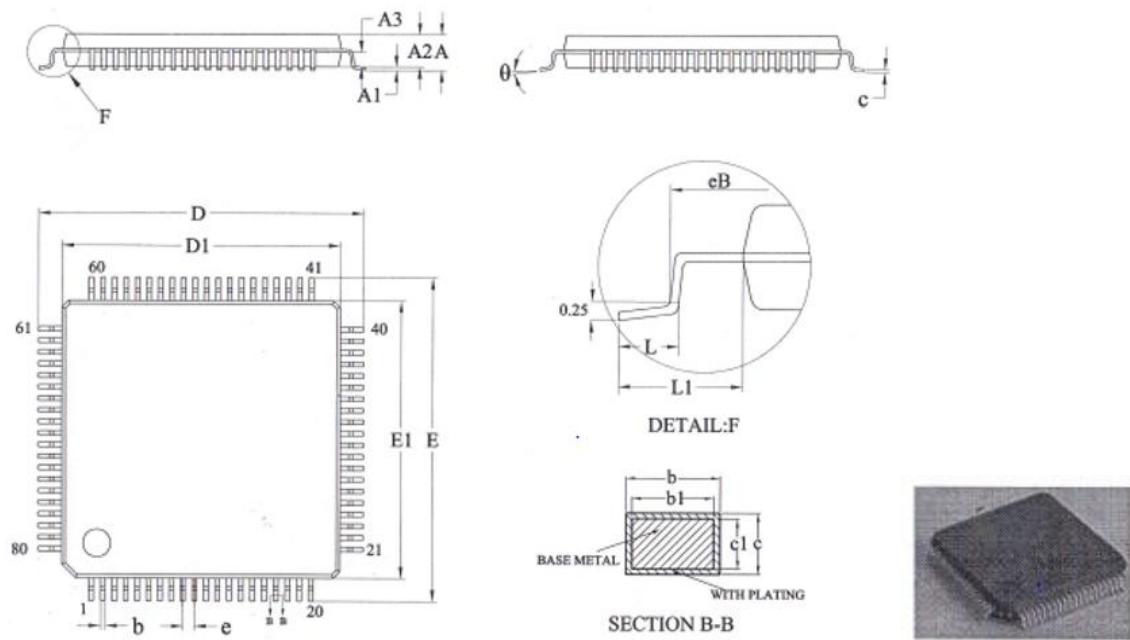
1. I_VDDIO 电流取决于 I/O 引脚上的电气负载;
2. 为了实现所显示的用于 IDLE 和 HALT 的 I_VDDA 电流, 必须通过写入 PCLKCR0 寄存器来明确关闭到 ADC 模块的时钟。
3. 典型值数据适用于常温和表压。
4. 上述数据仅供参考。

6.6 散热设计考虑

根据最终应用设计和运行情况, I_{DD} 和 I_{DDIO} 电流不能相同。最终应用中超过建议最大功率耗散的系统也许需要额外的散热增强。环境温度 (T_A) 随着最终应用和产品设计而变化。影响可靠性和功能性的关键参数是 T_J , 结温, 而非环境温度。因此, 应该注意将 T_J 保持在额定限值内。应该测量 T 外壳温度以估计运行结温 T_J 。

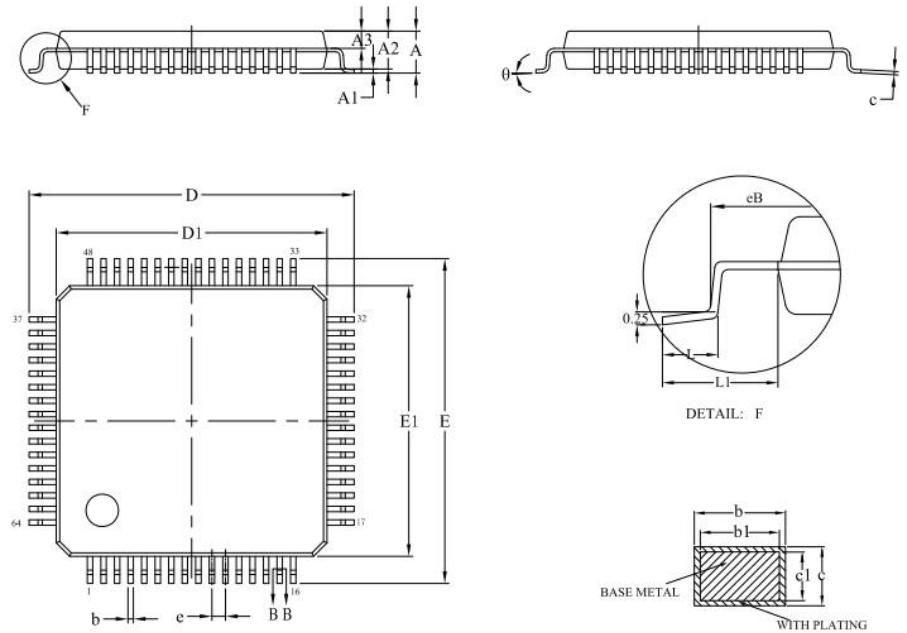
7. 封装外形尺寸

7.1 LQFP 80 PIN



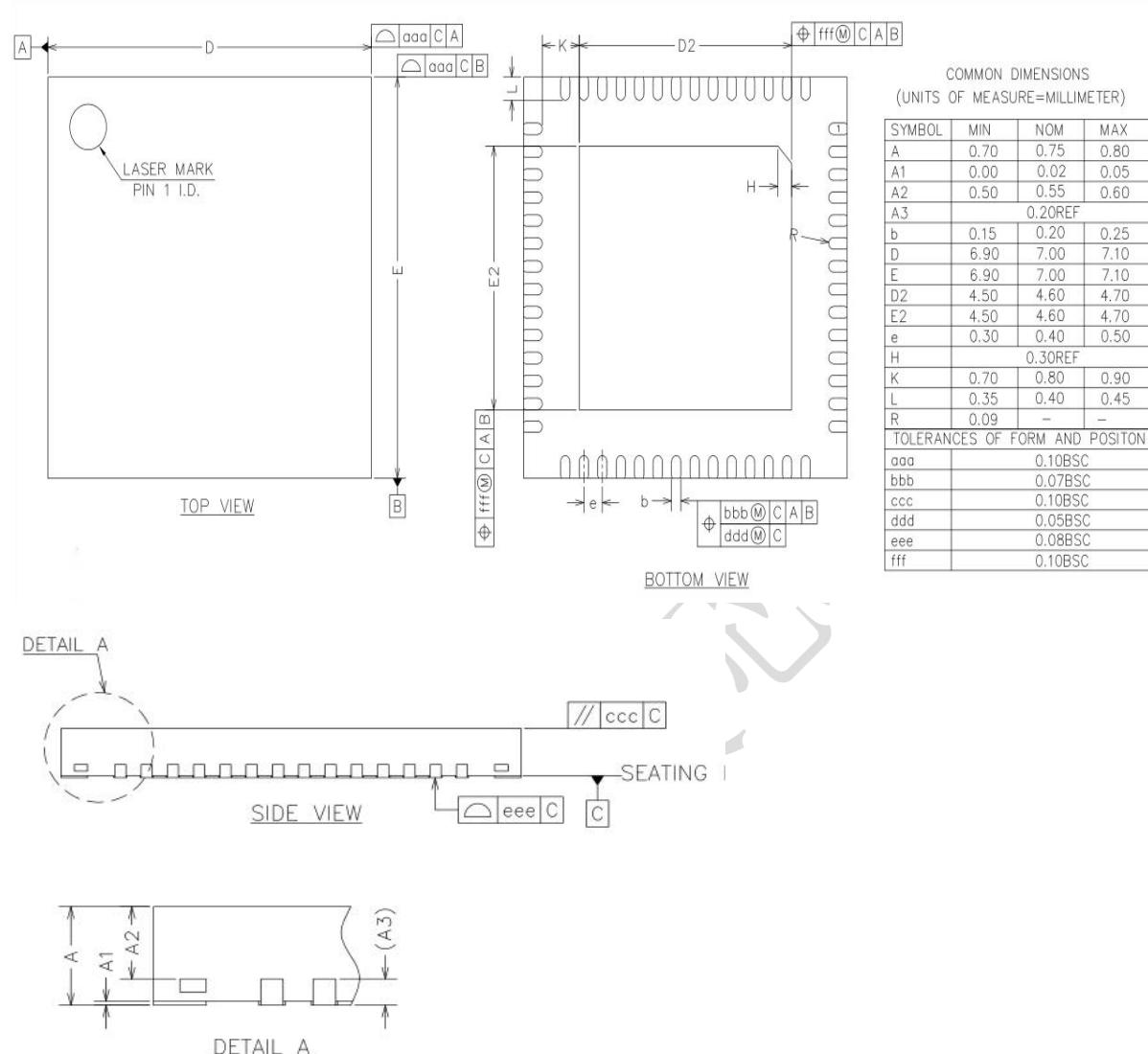
| SYMBOL | MILLIMETER | | |
|--------|------------|-------|-------|
| | MIN | NOM | MAX |
| A | — | — | 1.60 |
| A1 | 0.05 | — | 0.15 |
| A2 | 1.35 | 1.40 | 1.45 |
| A3 | 0.59 | 0.64 | 0.69 |
| b | 0.18 | — | 0.26 |
| b1 | 0.17 | 0.20 | 0.23 |
| c | 0.13 | — | 0.17 |
| c1 | 0.12 | 0.13 | 0.14 |
| D | 13.80 | 14.00 | 14.20 |
| D1 | 11.90 | 12.00 | 12.10 |
| E | 13.80 | 14.00 | 14.20 |
| E1 | 11.90 | 12.00 | 12.10 |
| eB | 13.05 | — | 13.25 |
| e | 0.50BSC | | |
| L | 0.45 | 0.60 | 0.75 |
| L1 | 1.00REF | | |
| θ | 0 | — | 7 |

7.2 LQFP 64 PIN



| SYMBOL | MILLIMETER | | |
|--------|------------|-------|-------|
| | MIN | NOM | MAX |
| A | — | — | 1.60 |
| A1 | 0.05 | — | 0.15 |
| A2 | 1.35 | 1.40 | 1.45 |
| A3 | 0.59 | 0.64 | 0.69 |
| b | 0.18 | — | 0.26 |
| b1 | 0.17 | 0.20 | 0.23 |
| c | 0.13 | — | 0.17 |
| c1 | 0.12 | 0.13 | 0.14 |
| D | 11.80 | 12.00 | 12.20 |
| D1 | 9.90 | 10.00 | 10.10 |
| E | 11.80 | 12.00 | 12.20 |
| E1 | 9.90 | 10.00 | 10.10 |
| e | 0.50BSC | | |
| eB | 11.05 | — | 11.25 |
| L | 0.45 | — | 0.75 |
| L1 | 1.00REF | | |
| θ | 0 | — | 7° |

7.3 VQFN 56 PIN

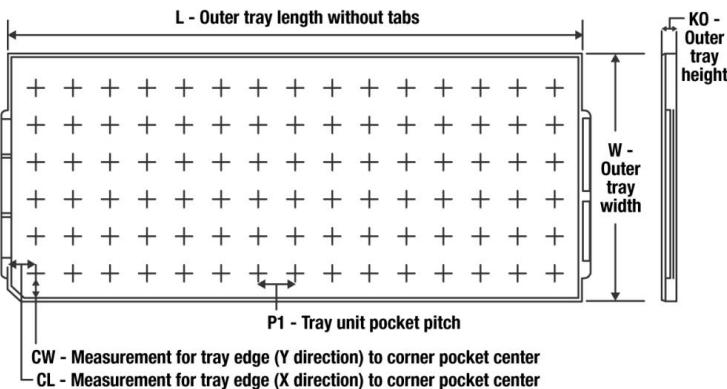


8. 订货信息

8.1 封装信息

| 产品型号 | 封装类型 | 封装图 | 管脚数 | 每托盘的芯片数量 | 生态计划 | 引线完成/球面处理 | 湿气等级-峰值温度 | 工作温度 | 芯片表面丝印 |
|------------------|------|-----|-----|----------|-----------------|-----------|--|-----------------|------------------|
| HXS320F28035PNT | LQFP | PN | 80 | 119 | Green (RoHS&无卤) | Sn | Level-3-260C-168HR (3 级-小于或等于 260° C-168 小时车间寿命) | -40° C 至 105° C | HXS320F28035PNT |
| HXS320F28035PMT | LQFP | PM | 64 | 160 | Green (RoHS&无卤) | Sn | Level-3-260C-168HR (3 级-小于或等于 260° C-168 小时车间寿命) | -40° C 至 105° C | HXS320F28035PMT |
| HXS320F28035PNS | LQFP | PN | 80 | 119 | Green (RoHS&无卤) | Sn | Level-3-260C-168HR (3 级-小于或等于 260° C-168 小时车间寿命) | -40° C 至 125° C | HXS320F28035PNS |
| HXS320F28035PMS | LQFP | PM | 64 | 160 | Green (RoHS&无卤) | Sn | Level-3-260C-168HR (3 级-小于或等于 260° C-168 小时车间寿命) | -40° C 至 125° C | HXS320F28035PMS |
| HXS320F28035RSHS | VQFN | RSH | 56 | 260 | Green (RoHS&无卤) | Sn | Level-3-260C-168HR (3 级-小于或等于 260° C-168 小时车间寿命) | -40° C 至 125° C | HXS320F28035RSHS |

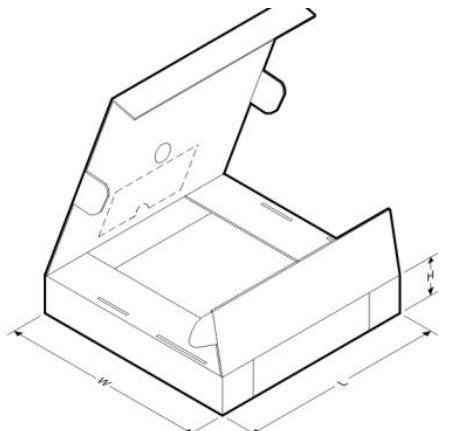
8.2 托盘信息



Chamfer on Tray corner indicates Pin 1 orientation of packed units.

| 订货产品型号 | 封装类型 | 封装图 | 管脚数 | 最小的小包装数量 | 托盘行x列 | 最高温度 | L (mm) 托盘长度 | W (mm) 托盘宽度 | KO (mm) 托盘高度 | P1 (mm) | CL (mm) | CW (mm) | 芯片第1脚标识点 |
|------------------|------|-----|-----|----------|-------|------|----------------|----------------|-----------------|---------|---------|---------|----------|
| HXS320F28035PNT | LQFP | PN | 80 | 119 | 7x17 | 150 | 315 | 135.9 | 7.620 | 17.9 | 14.30 | 13.95 | 左下角 |
| HXS320F28035PMT | LQFP | PM | 64 | 160 | 8x20 | 150 | 315 | 135.9 | 7.620 | 15.2 | 13.1 | 13.0 | 左下角 |
| HXS320F28035PNS | LQFP | PN | 80 | 119 | 7x17 | 150 | 315 | 135.9 | 7.620 | 17.9 | 14.30 | 13.95 | 左下角 |
| HXS320F28035PMS | LQFP | PM | 64 | 160 | 8x20 | 150 | 315 | 135.9 | 7.620 | 15.2 | 13.1 | 13.0 | 左下角 |
| HXS320F28035RSHS | VQFN | RSH | 56 | 260 | 26x10 | 150 | 315 | 135.9 | 7.620 | 11.8 | 10.0 | 10.35 | 左下角 |

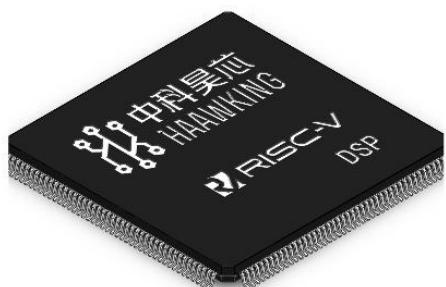
8.3 包装盒尺寸



包装盒尺寸

| 产品型号 | 封装系列 | 封装类型 | 管脚数 | 最小包装数量 | 包装盒长度(mm) | 包装盒宽度(mm) | 包装盒高度(mm) |
|------------------|------|------|-----|--------|------------|------------|------------|
| HXS320F28035PNT | LQFP | PN | 80 | 1190 | 370 | 150 | 88 |
| HXS320F28035PMT | LQFP | PM | 64 | 1600 | 370 | 150 | 88 |
| HXS320F28035PNS | LQFP | PN | 80 | 1190 | 370 | 150 | 88 |
| HXS320F28035PMS | LQFP | PM | 64 | 1600 | 370 | 150 | 88 |
| HXS320F28035RSHS | VQFN | RSH | 56 | 2600 | 370 | 150 | 88 |

8.4 封装概览



关于中科昊芯

“智由芯生 创享未来”的中科昊芯作为中国科学院科技成果转化企业，是国产数字信号处理器专业供应商，践行集成电路完全自主可控的国家战略，依托科研国家队的雄厚实力和深厚底蕴，以一种开放包容的积极心态，基于开源指令集架构 RISC-V，打造多个系列数字信号处理器产品，并构建完善的处理器产品生态系统，产品可广泛应用于图形图像处理、数字信号处理、工业控制及电机驱动、数字电源、运动控制等领域。



微信公众号



哔哩哔哩

公司网址: www.haawking.com

联系邮箱: haawking@mail.haawking.com

销售联系电话: 010-61934561

北京总部地址: 北京市海淀区中关村街道南四街 18 号紫金数码园 3 号楼 806 室

深圳办公室: 深圳市南山区高新南九道 45 号三航科技大厦 2218

联系电话: 0755-86573406

上海办公室: 上海市闵行区宜山路 2016 号合川大厦 1 号楼 7 楼 J 座

联系电话: 021-34687850

杭州办公室: 杭州市西湖区文三路 477 号华星科技大厦 701 室

联系电话: 19357651802