

厦门国科安芯科技有限公司

AS32A601 数据手册

目 录

1 芯片介绍.....	2
1.1 特色 (Features)	2
1.2 典型应用场景 (Target Application)	4
2 芯片概述.....	5
2.1 内核.....	5
2.2 总线架构.....	5
2.3 时钟管理.....	6
2.4 安全概述.....	7
2.5 存储系统.....	8
2.5.1 内置 SRAM.....	8
2.5.2 片内 Flash 存储器.....	8
2.6 DMA.....	9
2.7 MPPU	9
2.8 PMU.....	10
2.9 PMC.....	10
2.10 DSE	11
2.11 PLIC.....	11
2.12 CRC.....	12
2.13 ADC	13
2.14 DAC	13
2.15 TIMER.....	14
2.16 FlexCAN	15
2.17 USART.....	16
2.18 FlexSPI.....	16
2.19 FlexI2C	17
2.20 MAC.....	18
2.21 GPIO.....	18
2.22 DEBUG.....	19
3 附录	20
3.1 地址映射.....	20

1 芯片介绍

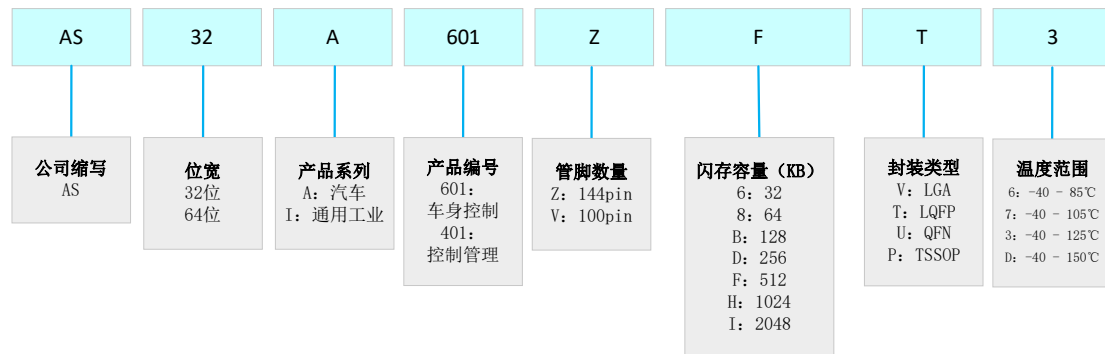
AS32A601 是国科环宇公司研制的一款基于 32 位 RISC-V 指令集、具有通用性的车规级 MCU 产品。产品具有丰富的 Flash 容量、支持 ASIL-B 等级的功能安全 ISO26262，同时具有高安全、低失效、多 IO、低成本等特点。

- ◆ 工作频率高达 180MHz
- ◆ 工作温度范围支持：-40° ~125° （符合 AEC-Q100 grade1 认证标准）
- ◆ 工作输入电压支持：2.7V~5.5V
- ◆ 休眠电流：≤200uA(可唤醒)
- ◆ 典型工作电流：≤50mA
- ◆ 封装工艺：LQFP144

芯片型号如下：

芯片类型	芯片型号	等级	封装	备注
MCU	AS32A601ZFT3	汽车级	LQFP144	
MCU	AS32I601ZFT6	工业级	LQFP144	

芯片命名规则



1.1 特色 (Features)

模块	说明
内核	自研 E7 内核，带有 FPU 与 L1Cache: 16KiB 数据缓存，16KiB 指令缓存，允许零等待访问嵌入式 Flash 与外部内存，最高频率 180MHz，带有 MPU。

时钟	<ul style="list-style-type: none"> ● 外部晶振 (OSC): 范围在 8MHz~40MHz ● 内部高频振荡器 (FIRC): 16MHz ● 内部低频振荡器 (FIRC): 32KHz ● 系统锁相环 (PLL): 最大支持 500MHz 输出
存储	<ul style="list-style-type: none"> ● 512KiB 内部 SRAM (带 ECC) ● 16KiB ICache 和 16KiB DCache (带 ECC) ● 512KiB D-Flash (带 ECC) ● 2MiB P-Flash (带 ECC)
系统	<ul style="list-style-type: none"> ● 2 个 16 通道的 DMA 模块 ● 1 个内存保护模块 (MPU) ● 1 个时钟监测模块 (CMU) ● 1 个错误控制模块 (FCU) ● 1 个电源管理模块 (PMU) ● 1 个其他控制模块 (CFU) ● 1 个实时计数器模块 (RTC) ● 1 个硬件加密模块 (DSU), 支持 AES、SM2/3/4 和 TRNG ● 1 个中断控制模块 (PLIC), ● 1 个 CRC 校验模块
电源管理	<ul style="list-style-type: none"> ● 5 种电源管理模式: RUN, SRUN, SLEEP, STOP, STANDBY ● 低电压检测和复位功能 (LVD/LVR) ● 高电压检测功能 (HVD)
模拟接口	<ul style="list-style-type: none"> ● 3 个 12 位的模数转换器 (ADC), 最多支持 48 通道模拟通路 ● 2 个模拟比较器 (ACMP) ● 2 个 8 位的数模转换器 (DAC) ● 1 个温度传感器
定时器	<ul style="list-style-type: none"> ● 4 个 32 位高级定时器 ● 4 个 16 位通用定时器

通信接口	<ul style="list-style-type: none"> ● 6 个 SPI 模块，支持主从模式标准 SPI 协议，速率最高可达 30MHz。同时支持 I2S 协议 ● 4 个 CAN 模块，都支持 CANFD ● 8 个 USART 模块，都具有 LIN 模式、同步串口模式和 SPI 模式 ● 1 个以太网（MAC）模块，支持 10/100M 模式、全/半双工模式 ● 4 路 I2C 模块，支持主从模式标准 IIC 协议。 ● 8 个通用 IO 模块（GPIO），GPIOH 支持唤醒输入
调试接口	<p>满足 RISC-V Debug Spec 0.13.2 标准的带有 JTAG 接口的调试器</p>

1.2 典型应用场景（Target Application）

- 工业领域：机器人控制（关节控制、通信管理控制）、工业通用控制系统、自动化控制系统（平台升降机控制）
- 汽车领域：ECU 控制系统（防抱死）、BCM 车身控制系统（内外灯光控制、中控锁控制、车窗控制、喇叭控制、后视镜控制等）、电机驱动系统（引擎散热风扇、水泵、高低压压缩机等）
- 高铁领域：运动控制、信号系统

2 芯片概述

2.1 内核

带有硬件 FPU 的 E7 内核是自行研发的最新一代的嵌入式处理器。它的设计目标是低成本、低功耗、高实时性、高安全性，在这基础之上同时提供优秀的运算性能。

E7 处理器是一款高效的高性能处理器：

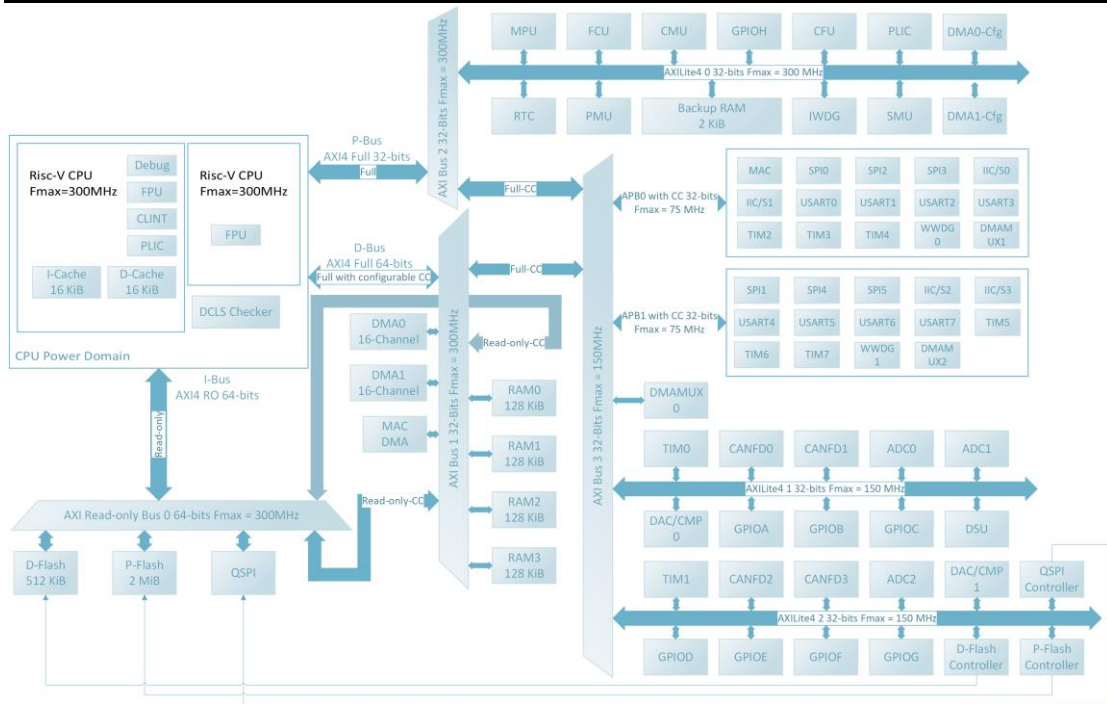
- 8 级双发射流水线
- 动态分支预测
- 哈弗架构的缓存（16 KiB 的 I-cache 和 16KiB 的 D-cache）
- 64 位 AXI4 总线接口

处理器支持以下的内存接口：

- 指令 AXI 接口（AXII）
- 数据 AXI 接口（AXID）
- 低延迟外设 AXI 接口（AXIP）

它内建的双浮点 FPU（浮点单元）可以加速浮点相关的软件运行。

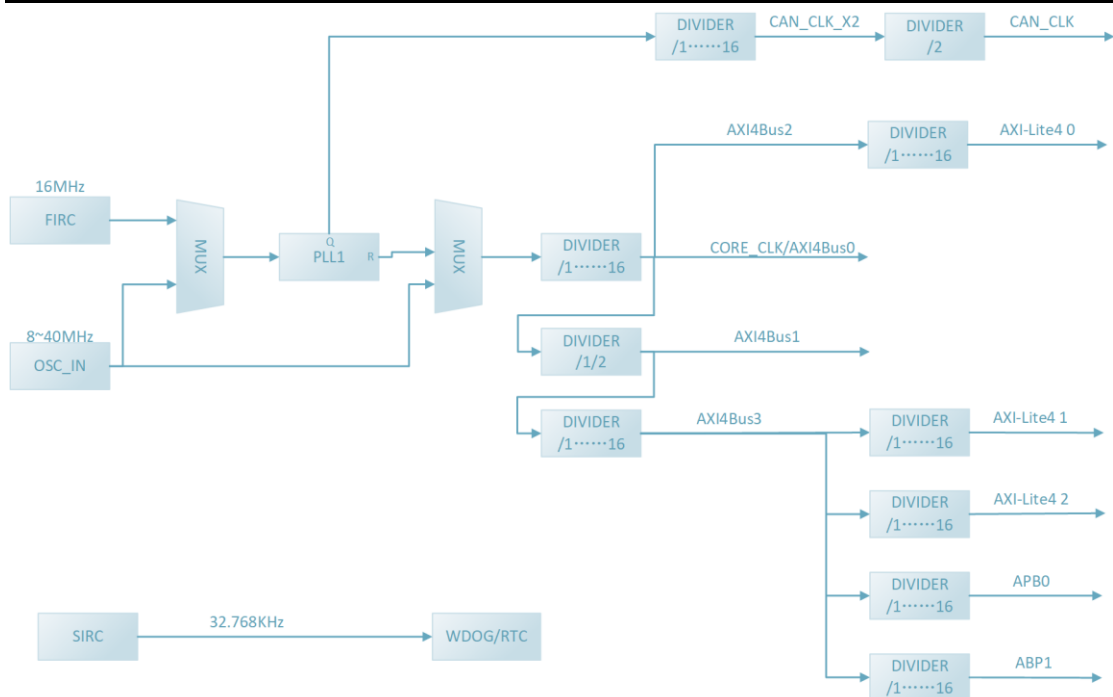
2.2 总线架构



MCU 的总线的 AXI Crossbar 是一个总线矩阵，用于互联 CPU 内核与系统存储器及外设模块的访问（读写）。

- Crossbar 的主机可以主动发起数据访问请求，而从机则只能被动接受访问
- Crossbar 的每个主机/从机与总线之间都有 ECC 编解码模块
- 每个端口都有独立的控制总线、地址总线和数据总线
- 不同的主机可以同时访问不同的从机，从而保证 MCU 系统工作带宽
- 主机对任意从机的保护均受 MPU 的保护

2.3 时钟管理



时钟管理模块为 MCU 提供时钟选择。包括快速内部时钟（FIRC）、内部慢速时钟（SIRC）、外部晶振（OSC）和 PLL 作为时钟源。其中 PLL 的参考时钟可以来源于 FIRC，也可以来源于 OSC。

在复位后 CPU 工作的默认时钟是来自 FIRC。FIRC 是 16MHz 的内部 RC 振荡器，其误差精度在 1%左右。然后应用程序可以选择 FIRC 或者外部的 8-40MHz 晶振作为时钟源。如果时钟监测模块检测到 OSC 时钟不稳定，则自动切换到 FIRC，并通过中断的方式通知应用程序。PLL 的倍频时钟可以高达 500MHz。时钟监测模块检测到 PLL 输出的时钟出现错误时，也会通过中断告知应用程序。

2.4 安全概述

系统针对需要高安全完整性的相关系统，进行了安全设计，来保证 MCU 的功能安全。对于不同的模块，保护措施也有所不同：

- 对于内核类的作为主机对系统进行操作的设备，采用延迟锁步方法保证安全
- 存储器及存储器和外设数据路径的安全由端到端 ECC 保护。对于外设，

端到端保护在 APB 转接桥处结束

- 时钟由多个分立的 CMU 进行监控
- 电源由 PMU 与 ADC 进行配合进行监控
- 外设的安全由应用级措施确保（多外设同时访问同一单元），硬件支持此应用级冗余，方法是提供连接至不同外设桥的 IO 模块，以最大化被监控和监控资源之间的独立性
- MBIST 和 LBIST 用于避免功能逻辑和安全机制中的潜在故障累积。专用机制用于检查安全机制的可用性及每个错误反应路径的功能
- 故障收集单元负责收集故障并作出反应
- 对于错误事件（包括 ECC 校正与检测、软件配置的验证错误）FDU（Fault Detect Unit）负责收集并向 FCU（Fault Control Unit）报告系统的错误事件
- 共因故障（Common Cause Failure, CCF）通过一组措施处理，以控制和避免跨系统级方法（温度及非功能信号监控）和后端技术（物理隔离硅区域、路由限制）的 CCF
- 通过分层处理器保护方法确保运行干扰保护，以允许不同 ASIL 标准的软件并行运行

MCU 集成的安全设计能够在高诊断范围内检测单点故障和潜在故障。但不是所有的共模故障都可以检测到。为了实现安全要求，需要在系统层面上对硬件环境与软件环境进行一定的要求，此项要求请参考安全手册。

2.5 存储系统

2.5.1 内置 SRAM

- 多达 512KiB 内部 SRAM；
- 提供 16KiB ICache；
- 提供 16KiB DCache；
- 支持误码校正（ECC）。

2.5.2 片内 Flash 存储器

高性能片内 Flash 的主要特性：

- Flash 最大支持 512KiB。

Flash 控制器特性：

- 支持 Flash 编程/擦除操作；
- 支持 Flash 读取和编程/擦除保护机制；
- 支持误码校正（ECC）：64 位 Flash 字 8 个 ECC 位；
- 多个存储区构成，支持同时操作：可在两个存储区中并行执行读取和编程/擦除操作；
- 内置预取缓冲区（Cache）以提升读效率，最快可实现单周期读取数据。

2.6 DMA

直接存储器访问（DMA）控制器用于在外设和存储器之间，或存储器与存储器之间的高速数据传输，无需 CPU 干预，数据可以通过 DMA 快速移动。

MCU 上集成有两组通用 DMA，每组 DMA 预设 16 个通道。

DMA 通过统一的 RMUX 单元进行请求信号分配，这使得每个 DMA 都可以通过软件配置访问任意外设。

DMA 支持以下外设：

- SPI
- I2C
- USART
- TIMER
- DAC
- ADC
- CMP
- QSPI（仅支持从 QSPI 到外设）

此外，Ethernet MAC 使用自己独立的 DMA。

2.7 MPPU

存储与外设保护单元（MPPU）作为 MCU 的安全机制之一，提供区域保护功能。MPPU 可以限定总线主机对选定地址区域的内存/外设的访问权限。可以对每个总线上的主机独立配置保护存储器区域的不同访问权限。

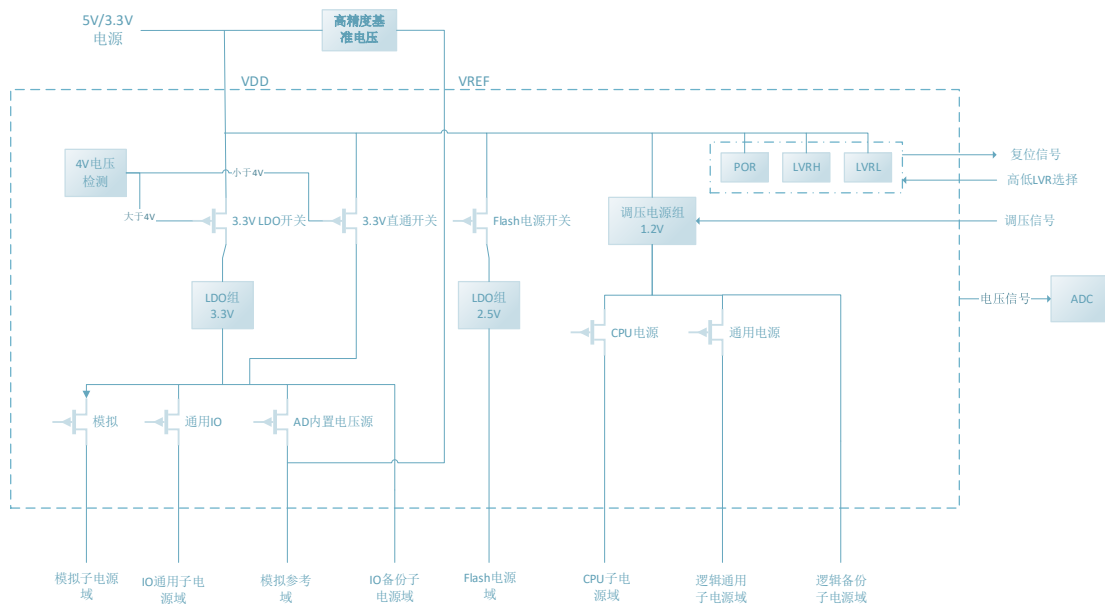
MPPU 特别有助于那些必须保护的某些关键或认证代码免受其他任务不当行为影响的应用程序。它通常由实时操作系统（RTOS）管理。如果程序访问 MPPU 禁止的存储位置，RTOS 可以检测到它并采取行动。在 RTOS 环境中，内核可以根据要执行的进程动态更新 MPPU 区域设置。

MPPU 具有以下特性

- 4 个主保护域，各有 8 个可编程的子保护区域
- 每个区域的大小在 32Bytes 到 4GBytes 之间
- 每个主机对每个区域可分别配置读、写、执行的属性
- 保护区域允许重叠与叠加
- 发生访问错误，信息会被保存至 MPPU

2.8 PMU

PMU 为一整体的电源管理模块，可进行电源开关的控制与 1.2V 电源域的调压，并有上电复位（POR）与低压复位（LVR）功能，整体框图如下：



2.9 PMC

电源管理控制器（PMC）负责切换 MCU 的各种电源模式及在电源模式下的功能。

- 运行模式（RUN）-CPU 全速运行，系统默认的模式
- 低速运行模式（SRUN）-CPU 在低速下运行，此时 PLL 与外部晶振被关闭，只使用内部高速晶振运行
- 停止模式（STOP）-关闭 CPU 时钟与系统时钟，开启部分外设与备份域设备
- 待机模式（STANDBY）-关闭除备份域外的绝大多数电源与时钟，备份域通过内部低速时钟驱动

2.10 DSE

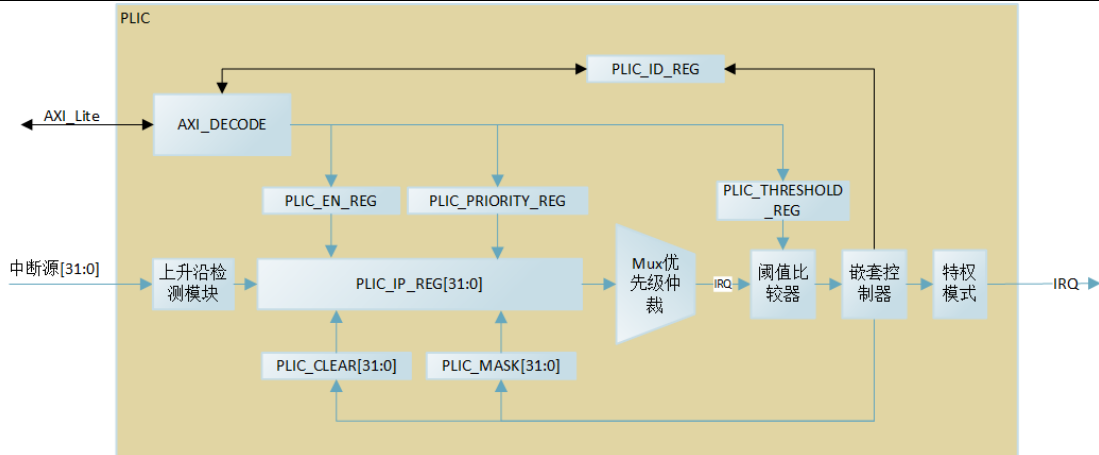
DSU（Device Security Engine）是硬件加解密模块，符合 HIS-SHE 安全规范标准。

- AES 的密钥长度支持 128bit、192bit 和 256bit
- AES 的加密方式支持 ECB、CBC 和 OFB
- 支持 AES CMAC 生成和验证
- 支持 SM2、SM3 和 SM4
- 支持真随机数（TRNG）的生成

2.11 PLIC

系统平台中断控制器（Platform Level Interrupt Controller, PLIC）是 CPU 的中断控制器，主要对中断源进行采样，优先级仲裁和分发。外设、CSR 软件中断和 CSR 定时器中断统一连到 PLIC，PLIC 统一管理并输出中断请求 IRQ_REQ 到内核。

PLIC 每次派发一个优先级最高的中断请求，并且支持中断嵌套。



- 支持 32 个中断源
- 每个中断源分配一个 ID，从 1 开始，0 代表无中断
- 支持设置每个中断源优先级，共计 32 级，可以设置为相同优先级，31 级最高，0 级最低，优先级一样时，ID 号小的先执行
- 支持设置中断使能位，注意区别于接口私有的中断使能，这两种需要全部开启
- 支持设置中断优先级阈值
- 输出单周期中断请求到内核，相同时间只有一个最高优先级中断输出，执行中断函数时不响应相同中断
- 中断源上升沿触发
- 中断入口为向量模式
- 支持中断嵌套（保存现场和恢复现场时不支持中断嵌套）
- 保存恢复现场固定为软件方式
- 支持 RISC-V 标准 CSR 定时器和 CSR 软件中断
- 支持中断委托机制
- 支持机器模式和管理模式

2.12 CRC

循环冗余校验（Cyclic Redundancy Check, CRC）是一种根据网络数据包或计算机文件等数据产生简短固定位数校验码的一种信道编码技术，主要用来检测或校验数据传输或者保存后可能出现的错误。它是利用除法及余数的原

理来作错误侦测的。

CRC（循环冗余校验）计算单元使用一个多项式发生器从 8 位/16 位/32 位的数据字中产生 CRC 码。

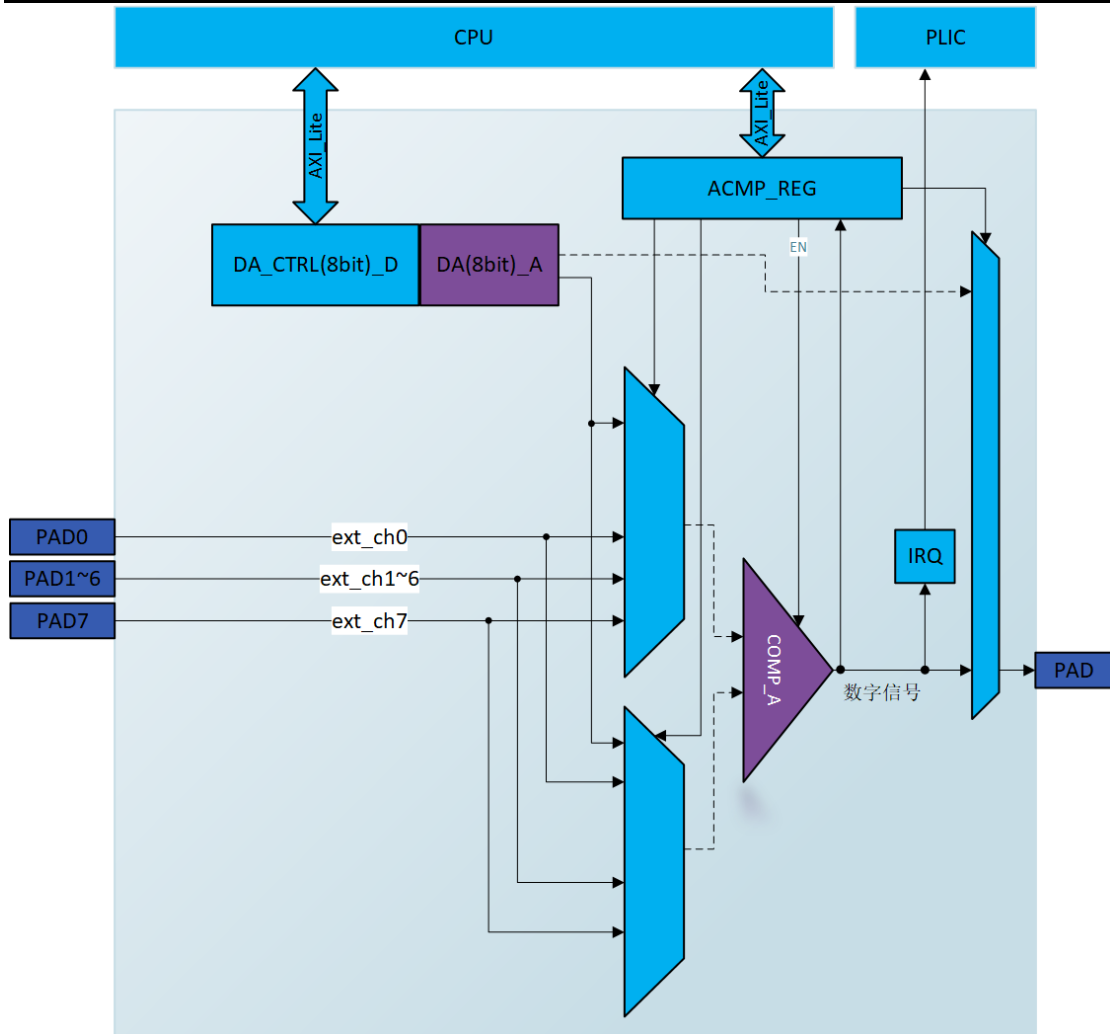
在众多的应用中，基于 CRC 的技术还常用来验证数据传输或存储的完整性。根据功能安全标准的规定，这些技术提供了验证 Flash 完整性的方法。CRC 计算单元有助于在运行期间计算软件的签名，并将该签名与链接时生成并存储在指定存储单元的参考签名加以比较。

- 使用位数可预设定的（7 位、8 位、16 位和 32 位）完全可编程多项式；
- 处理 8 位、16 位、32 位数据大小
- 可编程 CRC 初始值
- 单输入/输出 32 位数据寄存器
- 输入缓冲器可避免计算期间发生总线阻塞
- 对于 32 位数据大小，CRC 计算在 4 个 APB 时钟周期（PCLK）内完成
- 8 位通用寄存器（可用于临时存储）
- I/O 数据的可逆性选项
- 支持异或计算

2.13 ADC

ADC

2.14 DAC



蓝色部分为数字电路，紫色部分为模拟电路

2.15 TIMER

芯片内共包含 8 个定时器，其中 4 个高级定时器、4 个通用定时器。

高级定时器特性：

- 32 位递增、递减、递增/递减自动重载计数器；
- 32 位可编程预分频器；
- 重复计数器；
- 6 个独立通道，可用于输入捕获、输出比较、PWM 生成、单脉冲模式输出；
- 可生成中断和 DMA 请求；
- 支持可编程死区的互补输出；

- 支持正交编码器和霍尔传感器电路；
- 支持外部信号控制定时器，可实现多个定时器互连的同步电路；
- 支持 2 个断路输入，保证定时器输出信号处于安全状态。

通用定时器特性：

- 16 位递增、递减、递增/递减自动重装载计数器；
- 16 位可编程预分频器；
- 4 个独立通道，可用于输入捕获、输出比较、PWM 生成、单脉冲模式输出；
- 可生成中断和 DMA 请求；
- 支持正交编码器和霍尔传感器电路；
- 支持外部信号控制定时器，可实现多个定时器互连的同步电路

2.16 FlexCAN

控制器局域网络（CAN）根据 ISO11898-1: 2015 和 Bosch CAN FD 规范进行通信。连接到物理层需要额外的收发器硬件。

所有有关处理消息的函数都由接收处理程序和发送处理程序实现。接收处理程序管理消息接受过滤，将接收到的消息从 CAN 核心发送到消息 RAM，以及提供接收消息状态信息。发送处理程序负责将发送消息从消息 RAM 发送到 CAN 核心，并提供发送状态信息。

接受过滤通过最多 32 个过滤器元素的组合来实现，其中每个元素可以配置为范围、位掩码或专用 ID 过滤器。

- 设计符合 ISO11898-1/2015 规范；
- 支持 CAN 和 CAN FD 帧；
- 支持 ISO11899: 2015 规范中指定的 CAN FD 帧格式；
- 支持 64 字节 CAN FD 帧；
- 支持可变数据速率高达 8Mb/s；
- 支持正常数据速率 1Mb/s；
- 支持高达三个数据位的发射器延迟补偿；
- 支持可配置的发送和接受邮箱缓冲区；

- 支持两个深度为 64 帧消息的缓冲与 32 个 ID 过滤掩码；
- 支持低 ID 的消息优先发送；
- 支持待发送消息消除；
- 支持快速数据速率时的单独错误记录。

2.17 USART

USART 能够灵活地与外部设备进行全双工数据交换，满足外部设备对工业标准 NRZ 异步串行数据格式的要求。USART 可实现多种波特率。

USART 不仅支持同步单向通信和半双工单线通信，以及 LIN（局域互连网络）和调制解调器操作 (CTS/RTS)。通过配置多个缓冲区使用 DMA（直接存储器访问）可实现高速数据通信。

- 支持全双工/半双工异步通信
- 支持 NRZ 标准格式
- 可配置 16 倍过采样和 8 倍过采样
- 自动波特率检测
- 两个可用于发送和接收数据的内部 FIFO，可通过软件使能/禁止
- 数据字长度可编程，可以实现 6 位、7 位、8 位和 9 位数据字
- 可配置停止位，支持 1 个、1.5 个或 2 个停止位
- 用于同步通信的时钟输入/输出
- 支持 SPI 模式
- 支持 DMA 连续数据传输
- 支持发送器和接收器单独使能
- 支持发送和接收的单独信号极性控制
- 支持 TX/RX 引脚交换
- 支持调制解调 RS-485 发送器的硬件流控制
- 支持奇偶校验控制
- 支持 LIN 主从模式，可配置中断符

2.18 FlexSPI

SPI 全称是 serial peripheral interface ， 串行外围设备接口。AS32A601 提供了 4 个 SPI 接口，支持半双工/全双工的同步串行通信。该接口可配置为主机或从机模式，配置为主机模式时，它可为外部从器件提供通信时钟（SCK），SPI2 和 SPI3 接口支持 4 个从机，拥有 4 个从机片选，SPI0 和 SPI1 支持 1 个从机。

- 支持 Motorola 模式串行外设接口（SPI）
- 4 线全双工同步传输（3 线半双工）
- 4~32 位传输帧格式选择
- SPI 时钟速率可配置
- 主模式或从模式操作
- 主模式时最高频率 PCLK/2
- 从模式时最高频率 PCLK/8
- 数据发送顺序 MSB 在前
- 可编程的时钟极性和相位
- 可触发中断的专用发送和接收标志
- 支持 DMA 功能
- 支持 IIS 协议，支持 IIS 数据格式，左对齐数据格式，右对齐数据格式和 DSP 数据格式

2.19 FlexI2C

I2C（Inter-Integrated Circuit）总线是一种简单、双向二线制同步串行总线，通过时钟线（SCL）与数据线（SDA）进行数据传输。SCL 是由主机驱动产生的时钟信号，SDA 是双向数据信号，既可由主设备（主机）产生，也可由从设备（从机）产生。

- 支持主从一体模式
- 支持 7bit 范围地址和 10bit 范围地址
- 支持从机监测功能
- 支持多主机仲裁

- 支持可编程毛刺过滤
- 支持 SDA/SCL 低电平超时中断
- 支持 DMA 发送和接收

2.20 MAC

MAC 接口它实现了由 IEEE 802.3 定义的媒体访问控制（MAC）通过以太网连接的碰撞检测（CSMA/CD）算法。与外部主机的通信是通过一组控制和状态寄存器和针对外部共享 RAM 的直接内存访问（DMA）控制器来实现的。MAC 接口通过 DMA 方式与主机实现数据发送。它自动获取发送缓冲区数据和存储接收缓冲区数据到外部 RAM。它通过对接收和发送描述符链表的管理，实现多种内存分配方案。MAC 接口内部 RAM 被用作可配置的 FIFO 存储器块，并且有单独的用于发送和接收进程的存储器块。

- 支持外部 PHY 接口实现 10/100Mbit/s 数据传输速率；
- 通过符合 IEEE 802.3 的 MII 接口与外部快速以太网 PHY 进行通信；
- 支持全双工和半双工操作；
- 报头和帧起始数据（SFD）在发送路径中插入、在接收路径中删除；
- 可逐帧控制 CRC 和 PAD 自动生成；
- 接收帧时可自动去除 PAD/CRC；
- 支持多种灵活的地址过滤模式；
- 两组缓存 FIFO（接收 FIFO 和发送 FIFO）；
- DMA 在内存和 MAC 的缓存区进行数据搬移。

2.21 GPIO

每个 GPIO 引脚都可以通过软件配置为输出模式（推挽输出、开漏输出）、输入模式（浮空输入、上拉输入、下拉输入）或外设复用模式。绝大多数 GPIO 复用有多个数字模块/模拟模块的功能，最多复用功能可达 6 组。所有的 GPIO 都有高电流输出（20mA）的能力。每个 IO 都有独立的中断控制使能，并可配置为上升沿、下降沿或双沿触发。每个 GPIO 都配置有可选择的数字滤

波器，滤波的时钟周期可配置。

2.22 DEBUG

调试器基于 RISC-V Foundation 的 Debug Spec 0.13.2 制作，外部接口可以访问内部的寄存器与存储器，控制程序的运行/停止/复位等操作。

- JTAG 调试接口
- 16 个硬件断点

3 附录

3.1 地址映射

地址映射表

空间	起始地址	结束地址	空间大小	说明
AXI Bus 0	0x0000_0000	0x1FFF_FFFF	512 MiB	指令总线
	0x0000_0000	0x0000_0FFF	4 KiB	BOOT-ROM
	0x0100_0000	0x0107_FFFF	512 KiB	D-FLASH
	0x0200_0000	0x023F_FFFF	2 MiB	P-FLASH
	0x1000_0000	0x1FFF_FFFF	256 MiB	QSPI-FLASH
AXI Bus 1	0x2000_0000	0x2FFF_2FFF	256 MiB	数据总线
	0x2000_0000	0x2001_FFFF	128KiB	SRAM0
	0x2002_0000	0x2003_FFFF	128KiB	SRAM1
	0x2004_0000	0x2005_FFFF	128KiB	SRAM2
	0x2006_8000	0x2007_FFFF	128KiB	SRAM3
AXi Bus 2	0x3000_0000	0x7FFF_FFFF	1.25 GiB	配置/外设总线
AxiLite4 Bus0	0x3000_0000	0x3FFF_FFFF	256MiB	设置总线
	0x3000_0000	0x3000_03FF	1 KiB	MPU0-IBUS
	0x3000_0400	0x3000_07FF	1 KiB	MPU1-DBUS
	0x3000_0800	0x3000_0BFF	1 KiB	MPU2-PBUS
	0x3000_0C00	0x3000_0FFF	1 KiB	MPU3-DMA0
	0x3000_1000	0x3000_13FF	1 KiB	MPU4-DMA1
	0x3000_2000	0x3000_23FF	1 KiB	FCU
	0x3000_3000	0x3000_33FF	1 KiB	CMU
	0x3000_4000	0x3000_43FF	1 KiB	CFU
	0x3000_5000	0x3000_53FF	1 KiB	DMA0-CFG
	0x3000_5400	0x3000_57FF	1 KiB	DMA1-CFG
	0x3000_6000	0x3000_63FF	1 KiB	SMU

	0x3000_7000	0x3000_73FF	1 KiB	IWDG
	0x3000_8000	0x3000_83FF	1 KiB	PMU
	0x3000_9000	0x3000_93FF	1 KiB	GPIOH
	0x3000_A000	0x3000_A3FF	1 KiB	RTC
	0x3000_B000	0x3000_C3FF	2 KiB	BACKUP-RAM
	0x3001_0000	0x3001_FFFF	64 KiB	CLINT
	0x3002_0000	0x3002_FFFF	64 KiB	PLIC
AXI Bus 3	0x4000_0000	0x7FFF_FFFF	1 GiB	外设总线
	0x4000_0000	0x4000_03FF	1 KiB	DMAMUX0
AxiLite4 Bus1	0x4100_0000	0x41FF_FFFF	16 MiB	外设 Lite 总线 1
	0x4100_0000	0x4100_03FF	1 KiB	TIM0
	0x4101_0000	0x4101_FFFF	1 KiB	CANFD0
	0x4102_0000	0x4102_FFFF	1 KiB	CANFD1
	0x4100_1000	0x4100_13FF	1 KiB	ADC0
	0x4100_2000	0x4100_23FF	1 KiB	ADC1
	0x4100_3000	0x4100_33FF	1 KiB	DAC/CMP0
	0x4100_4000	0x4100_43FF	1 KiB	GPIOA
	0x4100_5000	0x4100_53FF	1 KiB	GPIOB
	0x4100_6000	0x4100_63FF	1 KiB	GPIOC
	0x4100_7000	0x4100_A3FF	4 KiB	DSU
AxiLite4 Bus2	0x4200_0000	0x42FF_FFFF	16 MiB	外设 Lite 总线 2
	0x4200_0000	0x4200_03FF	1 KiB	TIM1
	0x4201_0000	0x4201_FFFF	16 KiB	CANFD2
	0x4202_0000	0x4202_FFFF	16 KiB	CANFD3
	0x4200_1000	0x4200_13FF	1 KiB	ADC2
	0x4200_2000	0x4200_23FF	1 KiB	DAC/CMP1
	0x4200_3000	0x4200_33FF	1 KiB	GPIOD
	0x4200_4000	0x4200_43FF	1 KiB	GPIOE

	0x4200_5000	0x4200_53FF	1 KiB	GPIOF
	0x4200_6000	0x4200_63FF	1 KiB	GPIOG
	0x4210_0000	0x4210_03FF	1 KiB	D-FLASH CTRL
	0x4210_0400	0x4210_07FF	1 KiB	P-FLASH CTRL
	0x4210_0800	0x4210_0BFF	1 KiB	QSPI CTRL
APBBus0	0x5000_0000	0x50FF_FFFF	16 MiB	外设 APB 总线 0
	0x5000_0000	0x5000_03FF	1 KiB	MAC
	0x5000_1000	0x5000_13FF	1 KiB	SPI0
	0x5000_2000	0x5000_23FF	1 KiB	SPI1
	0x5000_3000	0x5000_33FF	1 KiB	SPI2
	0x5000_4000	0x5000_43FF	1 KiB	IIC/IIS0
	0x5000_5000	0x5000_53FF	1 KiB	IIC/IIS1
	0x5000_6000	0x5000_63FF	1 KiB	USART0
	0x5000_7000	0x5000_73FF	1 KiB	USART1
	0x5000_8000	0x5000_83FF	1 KiB	USART2
	0x5000_9000	0x5000_93FF	1 KiB	USART3
	0x5000_A000	0x5000_A3FF	1 KiB	TIM2
	0x5000_B000	0x5000_B3FF	1 KiB	TIM3
	0x5000_C000	0x5000_C3FF	1 KiB	TIM4
	0x5000_D000	0x5000_D3FF	1 KiB	WWDG0
0x5000_E000	0x5000_E3FF	1 KiB	DMAMUX1	
APBBus1	0x5100_0000	0x51FF_FFFF	16 MiB	外设 APB 总线 1
	0x5100_0000	0x5100_03FF	1 KiB	SPI3
	0x5100_1000	0x5100_13FF	1 KiB	SPI4
	0x5100_2000	0x5100_23FF	1 KiB	SPI5
	0x5100_3000	0x5100_33FF	1 KiB	IIC/IIS2
	0x5100_4000	0x5100_43FF	1 KiB	IIC/IIS3
	0x5100_5000	0x5100_53FF	1 KiB	USART4

	0x5100_6000	0x5100_63FF	1 KiB	USART5
	0x5100_7000	0x5100_73FF	1 KiB	USART6
	0x5100_8000	0x5100_83FF	1 KiB	USART7
	0x5100_9000	0x5100_93FF	1 KiB	TIM5
	0x5100_A000	0x5100_A3FF	1 KiB	TIM6
	0x5100_B000	0x5100_B3FF	1 KiB	TIM7
	0x5100_C000	0x5100_C3FF	1 KiB	WWDG1
	0x5100_D000	0x5100_D3FF	1 KiB	DMAMUX2