



# T660

## 网络存储安全芯片

### 技术手册

#### 修改记录

版本号	描述	日期
v0.1	草稿版	2019/3/2
v1.0	初版发布版本	2019/4/3
v1.1	修改部分细节说明	2019/4/25
v1.2	增加 QSPI 描述	2019/6/4



## Table of Content

<b>1</b>	<b>概述 .....</b>	<b>5</b>
1.1	产品简介 .....	5
1.2	应用产品 .....	5
1.3	芯片架构 .....	6
1.4	芯片特性 .....	6
1.4.1	CPU 资源 .....	6
1.4.2	USB3.0 OTG 接口 .....	6
1.4.3	SATA3.0 (主/从) 接口 .....	7
1.4.4	GMAC 接口 .....	7
1.4.5	安全引擎 .....	7
1.4.6	存储资源 .....	7
1.4.7	其他资源 .....	7
1.4.8	安全特性 .....	8
1.4.9	物理规格 .....	8
1.5	地址映射 .....	9
1.6	中断源 .....	10
<b>2</b>	<b>硬件特性 .....</b>	<b>11</b>
2.1	芯片封装 .....	11
2.2	管脚分布 .....	12
2.3	管脚描述 .....	14
2.4	管脚复用 .....	18
2.5	上电时序 .....	18
2.6	电性能参数 .....	19
2.7	功耗 .....	19
2.8	PCB 设计建议 .....	19
<b>3</b>	<b>CPU 子系统 .....</b>	<b>20</b>
3.1	CK803S 处理器 .....	20
3.1.1	简介 .....	20
3.1.2	特性 .....	20
3.1.3	架构 .....	20
3.1.4	矢量中断控制器 .....	21
3.1.5	系统计时器 .....	22
3.2	存储 .....	22
3.3	DMA .....	23
3.3.1	模块概述 .....	23
3.3.2	模块特性 .....	23
3.4	定时器 .....	24
3.4.1	模块概述 .....	24
3.4.2	模块特性 .....	24



3.5	看门狗	24
3.5.1	模块概述	24
3.5.2	模块特性	25
3.6	SCU	25
3.6.1	模块概述	25
3.6.2	模块特性	26
3.6.3	时钟树	26
3.6.4	复位树	27
<b>4</b>	<b>安全引擎</b>	<b>28</b>
4.1	CRYPTO 引擎	28
4.1.1	模块概述	28
4.1.2	模块特性	28
4.1.3	工作方式	29
4.2	PKE 引擎	31
4.2.1	模块概述	31
4.2.2	模块特性	31
4.2.3	工作方式	31
4.3	TRNG	32
4.3.1	模块概述	32
4.3.2	模块特性	32
<b>5</b>	<b>网络接口</b>	<b>33</b>
5.1	GMAC1 控制器	33
5.1.1	模块概述	33
5.1.2	模块特性	33
5.1.3	工作方式	34
<b>6</b>	<b>USB OTG 接口</b>	<b>36</b>
6.1	模块概述	36
6.2	模块特性	36
<b>7</b>	<b>SATA 接口</b>	<b>38</b>
7.1	SATA Host 控制器	38
7.1.1	模块概述	38
7.1.2	模块特性	39
7.2	SATA Device 控制器	39
7.2.1	模块概述	39
7.2.2	模块特性	40
<b>8</b>	<b>外围设备接口</b>	<b>41</b>
8.1	I2C 控制器	41
8.1.1	模块概述	41
8.1.2	模块特性	41
8.1.3	工作方式	42
8.2	QSPI 控制器	42



---

8.2.1	模块概述	42
8.2.2	模块特性	42
8.3	SPI 控制器	43
8.3.1	模块概述	43
8.3.2	模块特性	43
8.4	UART0 控制器	44
8.4.1	模块概述	44
8.4.2	模块特性	44
8.5	UART1 控制器	45
8.6	GPIO1 控制器	45
8.6.1	模块描述	45
8.6.2	模块特性	45
<b>9</b>	<b>安全特性</b>	<b>47</b>
9.1	电压检测	47
9.1.1	模块概述	47
9.1.2	模块特性	47
9.2	温度检测	48
9.2.1	模块概述	48
9.2.2	模块特性	48
9.2.3	模块时序	49
9.3	物理探测防护	49
9.3.1	金属屏蔽层	49
9.3.2	后端设计防护	49
9.4	芯片 ID	49
9.4.1	模块概述	49
9.4.2	模块特性	49

## 1 概述

### 1.1 产品简介

T660 是由方寸微电子自主开发的新一代 SoC 国密主控安全芯片，具有功能丰富、性能强劲、功耗低、安全性高等特点，可广泛适用于加密移动存储、USB 安全网卡、安全网关、网闸、单向导入导出设备、密码卡、密码机、家庭云存储等众多安全领域产品。

该芯片集成国产 32 位高性能 RISC CPU，可支持 USB3.0、SATA3.0、GMAC 等多种超高速接口，并集成多种国密算法（如 SM2、SM3、SM4），可满足国家信息安全领域对底层核心安全芯片的应用需求。

该芯片提供完整的 SDK 供客户进行定制化开发，尤其针对典型应用场景提供了源码级方案支撑，可帮助客户缩短产品开发周期、降低整体开发成本，提升产品市场竞争力。

### 1.2 应用产品

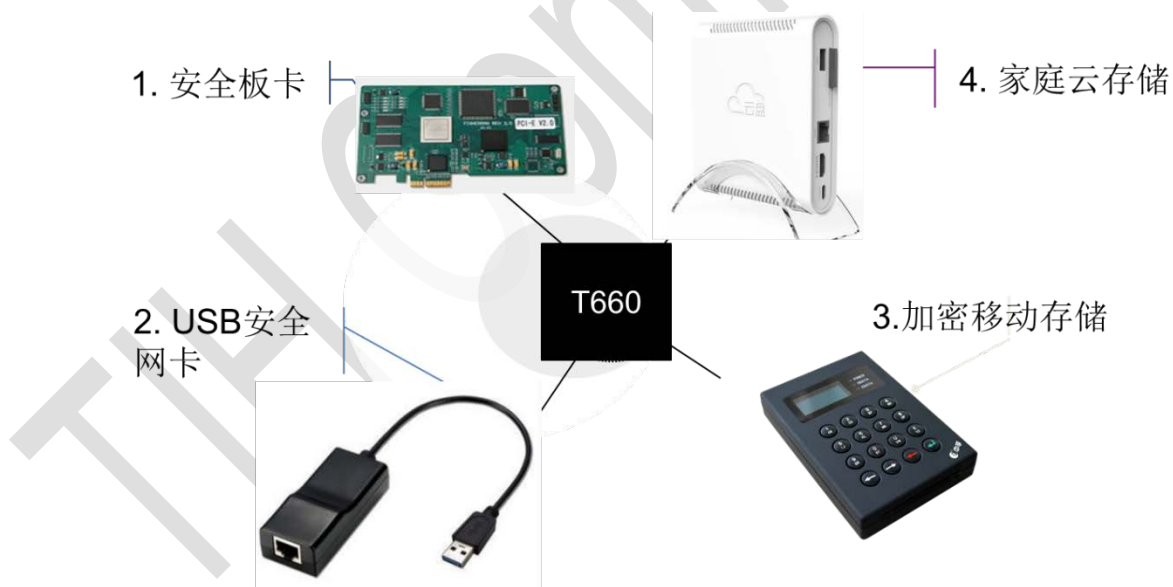


图 1.1 产品应用方案

## 1.3 芯片架构

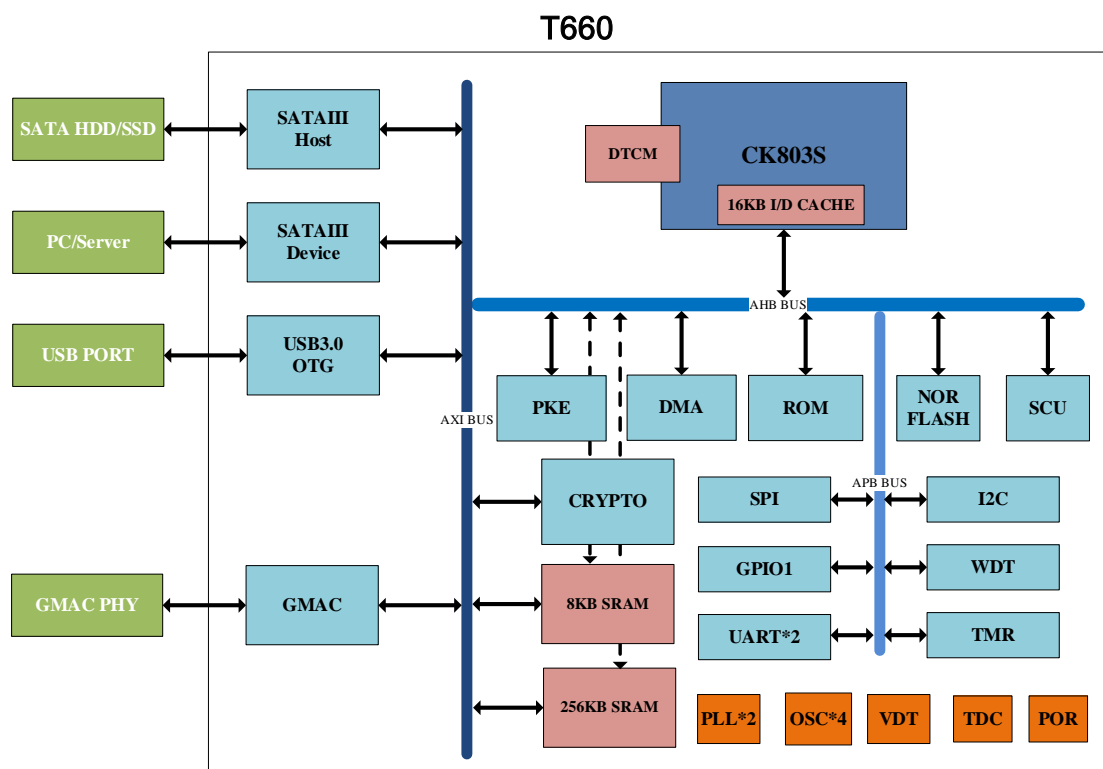


图 1.2 芯片系统架构框图

## 1.4 芯片特性

### 1.4.1 CPU 资源

- 集成 32 位国产 CPU CK803S
- 最高工作频率 260Mhz
- 内置 16KB I/D Cache
- 内置 DTCM 32KB

### 1.4.2 USB3.0 OTG 接口

- 支持一路 USB3.0 OTG 接口速率 5Gbps，向下兼容 USB2.0/USB1.1
- 静态角色转换（主机/设备选择）
- 支持控制/批量/中断/等时传输类型
- 符合 Universal Serial Bus（USB） revision 3.0 标准协议

### 1.4.3 SATA3.0（主/从）接口

- 支持一路 SATAIII device 接口速率 6Gbps，向下兼容 3Gbps/1.5Gbps
- 支持一路 SATAIII host 接口速率 6Gbps，向下兼容 3Gbps/1.5Gbps
- 符合 Serial ATA Revision 3.0 标准协议
- 支持 NCQ 32 命令队列

### 1.4.4 GMAC 接口

- 支持 1 路 GMAC RGMII 接口
- 支持 10/100/1000Mbps 网络接口速率

### 1.4.5 安全引擎

- 支持 SM4 数据硬件加密，加密性能 800MB/s@200Mhz
- 支持 ECB、CBC、OFB、CFB、CTR、XTS 6 种加密模式
- 支持大数模加、模减、模乘运算协处理
- SM2 密钥对生成速度 500 对/s
- 支持 SM2 签名验签，性能 $\geq 1200/600$  次/s@200Mhz
- 支持 SM3 算法
- 支持一路 TRNG 发生器，生成速率 $\geq 30$ Mbps@50Mhz

\*以上为硬件引擎性能，非最终产品性能

### 1.4.6 存储资源

- 32KB ROM
- 256KB SRAM
- 8KB SRAM（系统专用）
- 512KB 片内 flash

### 1.4.7 其他资源

- 内置硬件 DMA
- 内置 POR（Power on reset）电路
- 内置 8 个定时器
- 内置中断控制器
- 内置 1 个看门狗
- 支持 1 路 QSPI 主接口



- 支持 1 路 SPI 主/从接口
- 支持 1 路 I2C 主/从接口
- 支持 2 路 UART 接口
- 支持 16 位 GPIO 接口
- 支持在线调试

#### 1.4.8 安全特性

- 支持电压检测
- 支持温度检测
- 支持物理探测防护
- 每颗芯片具备全球唯一 ID

#### 1.4.9 物理规格

- Core 电压为 1.0V
- IO 电压为 3.3V
- 支持 BGA137 10mm x 10mm x 1.2mm 封装
- 工作温度 0~70°C, -40~85°C



## 1.5 地址映射

表 1.1 地址映射表

基地址	大小	模块名称	说明
0x0000_0000	1MB	ROM	
0x1000_0000	1MB	SATA Device 寄存器端口	
0x1010_0000	1MB	CRYPTO 寄存器端口	
0x1020_0000	1MB	DMA 寄存器端口	
0x1040_0000	1MB	QSPI 寄存器端口	
0x1050_0000	1MB	SCU 寄存器端口	
0x1090_0000	1MB	GMAC1 寄存器端口	
0x10B0_0000	1MB	INTC 寄存器端口	
0x10C0_0000	1MB	TRNG 寄存器端口	
0x10D0_0000	1MB	PKE 寄存器端口	
0x1110_0000	1MB	8KB SRAM	8KB SRAM 在 AHB 总线地址
0x1120_0000	1MB	256KB SRAM	256KB SRAM 在 AHB 总线地址
0x1210_0000	1MB	SPI 寄存器端口	
0x1220_0000	1MB	UART0 寄存器端口	
0x1230_0000	1MB	UART1 寄存器端口	
0x1240_0000	1MB	TIMER 寄存器端口	
0x1250_0000	1MB	WDT 寄存器端口	
0x1280_0000	1MB	GPIO1 寄存器端口	
0x12A0_0000	1MB	IIC 寄存器端口	
0x2200_0000	1MB	USB OTG 寄存器端口	
0x2210_0000	1MB	SATA Host 寄存器端口	
0x2220_0000	1MB	8KB SRAM	8KB SRAM 在 AXI 总线地址
0x2230_0000	1MB	256KB SRAM	256KB SRAM 在 AXI 总线地址
0x2300_0000	8MB	CRYPTO_S1 数据端口	
0x2380_0000	8MB	CRYPTO_S2 数据端口	

## 1.6 中断源

CK803S 的中断源映射如下：

表 1.2 CK803S 中断源

No.	中断源	说明
26	Core_Timer	
25	VDT	
24	Reserved	
23	TRNG	
22	PKE	
21	AXIC	
20	AHBC	
19	GPIO1	
18	Reserved	
17	WDT	
16	IIC	
15	UART_1	
14	UART_0	
13	QSPI	
12	Nor-flash	
11	SPI	
10	TIMER	
9	DMA	
8	Reserved	
7	GMAC_1	
6	Reserved	
5	Reserved	
4	Reserved	
3	CRYPTO	
2	SATA3_Host	
1	SATA3_Slave	
0	USB3_OTG	



## 2 硬件特性

### 2.1 芯片封装

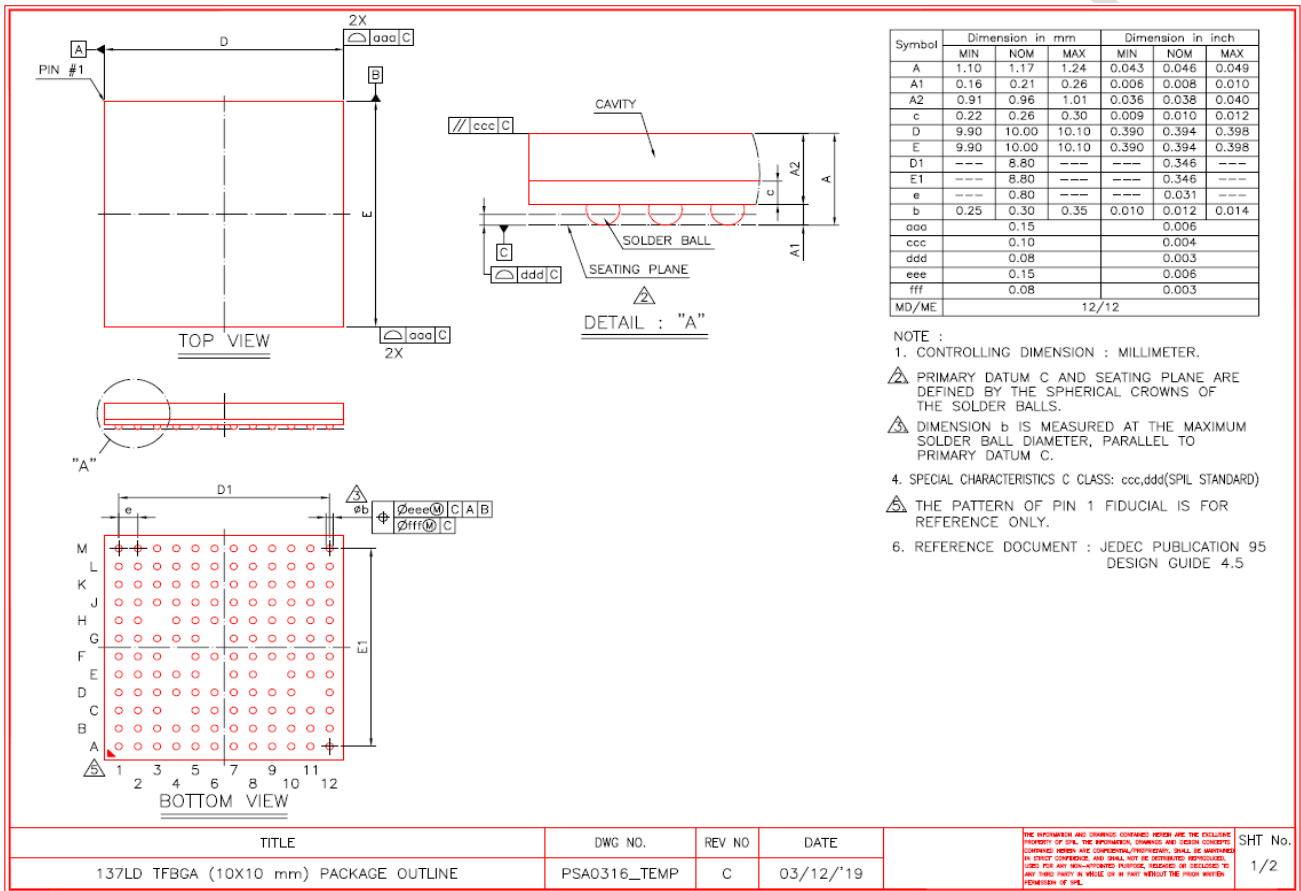


图 2.1 芯片封装尺寸图

## 2.2 管脚分布

Top view

	1	2	3	4	5	6
A	GNDA_USB	GNDA_SD	X_SD_TXN	GNDA_SD	X_SD_XSCI	GNDA_SD
B	X_DP	VCC33IO	X_SD_TXP	GNDA_SD	X_SH_XSCI	GNDA_SD
C	X_DM	GNDA_USB	X_SD_REXT	NC	GNDA_SD	X_SH_REXT
D	GNDA_USB	X_JTG_TCK	X_TM	X_JTG_TMS	VCCCK	VCC33A_SD
E	X_SSTXA	GNDA_USB	X_GPIO1_15	VCCCK	VCC33A_USB	NC
F	X_SSTXB	GNDA_USB	X_GPIO1_14	NC	GNDA_USB	GNDK

	7	8	9	10	11	12
A	X_SD_RXN	GNDA_SH	X_SH_TXP	GNDA_SH	X_SH_RXN	GNDA_SH
B	X_SD_RXP	GNDA_SH	X_SH_TXN	GNDA_SH	X_SH_RXP	GNDA_SH
C	VCCCK	VCC33A_SH	GNDA_SH	VCC33A_SH	VCC10A_SH	X_XSCI
D	GNDK	VCC10A_SH	VCCCK	VCC33_OS	NC	X_XSCO
E	GNDK	GNDK	NC	NC	NC	NC
F	GNDK	GNDK	GNDK	VCCCK	VCC33IO	NC



	1	2	3	4	5	6
<b>G</b>	GNDA_US B	VCC33A_U SB	X_GPIO1_ 13	X_GPIO1_ 2	GNDA_US B	NC
<b>H</b>	X_SSRXA	GNDA_US B	NC	VCC33A_U SB	VCC10A_U SB	GNDK
<b>J</b>	X_SSRXB	GNDA_US B	VCC33IO	GNDK	X_GPIO1_ 0	GNDK
<b>K</b>	GNDA_US B	X_GPIO1_ 1	GNDK	GNDK	GNDK	GNDK
<b>L</b>	X_I2C_SC L	GNDK	VCC3318_ ETH	ETH1_RXD 2	GNDK	ETH1_RCK
<b>M</b>	X_I2C_SD A	GNDK	VCC3318_ ETH	ETH1_RXD 3	GNDK	ETH1_MD C

	7	8	9	10	11	12
<b>G</b>	GNDK	GNDK	GNDK	X_GPIO1_ 8	X_GPIO1_ 6	X_GPIO1_ 5
<b>H</b>	GNDK	GNDK	GNDK	X_GPIO1_ 9	X_GPIO1_ 7	X_GPIO1_ 4
<b>J</b>	GNDK	GNDK	GNDK	ETH1_TCK	X_GPIO1_ 0	X_GPIO1_ 3
<b>K</b>	VCCCK	GNDK	ETH1_TXC TL	ETH1_TXD 2	ETH1_TXD 3	X_GPIO1_ 1
<b>L</b>	ETH1_RX D1	ETH1_RXC TL	VCC33IO	X_UART0_ TXD	X_UART0_ RXD	X_GPIO1_ 2
<b>M</b>	ETH1_MDI O	ETH1_RXD 0	VCC33IO	ETH1_TXD 0	ETH1_TXD 1	X_RST_N

## 2.3 管脚描述

表 2.1 管脚描述

位置	名称	I/O	功能描述
A			
A1	GNDA_USB		模拟地
A2	GNDA_SD		模拟地
A3	X_SD_TXN	O	SATA Device 接口差分输出 N
A4	GNDA_SD		模拟地
A5	X_SD_XSCI	I	SATA Device 参考输入时钟, 30MHz
A6	GNDA_SD		模拟地
A7	X_SD_RXN	I	SATA device 接口高速差分输入 N
A8	GNDA_SH		模拟地
A9	X_SH_TXP	O	SATA Host 接口高速差分输出 P
A10	GNDA_SH		模拟地
A11	X_SH_RXN	I	SATA Host 接口高速差分输入 N
A12	GNDA_SH		模拟地
B			
B1	X_DP	I	USB2.0 高速差分输入输出
B2	VCC33IO		3.3v 数字电源
B3	X_SD_TXP	O	SATA Device 接口差分输出 P
B4	GNDA_SD		模拟地
B5	X_SH_XSCI	I	SATA Host 参考输入时钟
B6	GNDA_SD		模拟地
B7	X_SD_RXP	I	SATA device 接口高速差分输入 P
B8	GNDA_SH		模拟地
B9	X_SH_TXN	O	SATA Host 接口高速差分输出 N
B10	GNDA_SH		模拟地
B11	X_SH_RXP	I	SATA Host 接口高速差分输入 P
B12	GNDA_SH		模拟地
C			
C1	X_DM	I	USB2.0 高速差分输入输出
C2	GNDA_USB		模拟地
C3	X_SD_REXT	I	SATA Device 匹配电阻, 需下拉 18K 欧姆至地信号
C4	NC	I	悬空
C5	GNDA_SD		模拟地
C6	X_SH_REXT	I	SATA Host 匹配电阻, 需下拉 18K 欧姆至地信号
C7	VCCK		1.0v 数字电源
C8	VCC33A_SH		3.3v 模拟电源



C9	GNDA_SH		模拟地
C10	VCC33A_SD		3.3v 模拟电源
C11	VCC10A_SH		1.0v 模拟电源
C12	X_XSCI	I	系统输入时钟, 30MHz
D			
D1	GNDA_USB		模拟地
D2	X_JTG_TCK	I	CK803S JTAG 测试参考时钟输入
D3	X_TM	IO	测试模式使能信号: 0: 正常功能模式 1: 测试模式
D4	X_JTG_TMS	IO	CK803S JTAG 测试控制信号输入输出接口
D5	VCCCK		1.0v 数字电源
D6	VCC33A_SD		SATA Device 3.3v 模拟电源
D7	GNDK		数字地
D8	VCC10A_SD		SATA Device 1.0v 模拟电源
D9	VCCCK		1.0v 数字电源
D10	VCC33_OSC		OSC 3.3v 数字电源
D11	NC		悬空
D12	X_XSCO	O	系统晶振输出, 30Mhz
E			
E1	X_SSTXA	O	USB Device 3.0 接口差分输出 A
E2	GNDA_USB		模拟地
E3	X_GPIO1_15	IO	GPIO1 通用输入输出端口 15
E4	VCCCK		1.0v 数字电源
E5	VCC33A_USB		USB 3.3v 模拟电源
E6	NC		悬空
E7	GNDK		数字地
E8	GNDK		数字地
E9	NC		悬空
E10	NC		NC
E11	NC		NC
E12	NC		NC
F			
F1	X_SSTXB	O	USB Device 3.0 接口差分输出 B
F2	GNDA_USB		模拟地
F3	X_GPIO1_14	IO	GPIO1 通用输入输出端口 14
F4	NC		悬空
F5	GNDA_USB		模拟地
F6	GNDK		数字地
F7	GNDK		数字地
F8	GNDK		数字地
F9	GNDK		数字地



F10	VCCK		1.0v 数字电源
F11	VCC33IO		3.3v 数字电源
F12	NC		NC
G			
G1	GNDA_USB		USB 模拟地
G2	VCC33A_USB		USB 3.3v 模拟电源
G3	X_GPIO1_13	IO	GPIO1 通用输入输出端口 13
G4	X_GPIO1_12	IO	GPIO1 通用输入输出端口 12
G5	GNDA_USB		模拟地
G6	NC		悬空
G7	GNDK		数字地
G8	GNDK		数字地
G9	GNDK		数字地
G10	X_GPIO1_8	IO	GPIO1 通用输入输出端口 8
G11	X_GPIO1_6	IO	GPIO1 通用输入输出端口 6
G12	X_GPIO1_5	IO	GPIO1 通用输入输出端口 5
H			
H1	X_SSRXA	IO	USB 3.0 超速差分输入 A
H2	GNDA_USB		模拟地
H3	NC		悬空
H4	VCC33A_USB		3.3v 模拟电源
H5	VCC10A_USB		1.0v 模拟电源
H6	GNDK		数字地
H7	GNDK		数字地
H8	GNDK		数字地
H9	GNDK		数字地
H10	X_GPIO1_9	IO	GPIO1 通用输入输出端口 9
H11	X_GPIO1_7	IO	GPIO1 通用输入输出端口 7
H12	X_GPIO1_4	IO	GPIO1 通用输入输出端口 4
J			
J1	X_SSRXB	IO	USB 3.0 超速差分输入 B
J2	GNDA_USB		模拟地
J3	VCC33IO		3.3v 数字电源
J4	GNDK		数字地
J5	X_GPIO1_10	IO	GPIO1 通用输入输出端口 10
J6	GNDK		数字地
J7	GNDK		数字地
J8	GNDK		数字地
J9	GNDK		数字地
J10	ETH1_TCK	I	GMAC1 TX 端时钟
J11	X_GPIO1_0	IO	GPIO1 通用输入输出端口 0, 默认上拉
J12	X_GPIO1_3	IO	GPIO1 通用输入输出端口 3





K			
K1	GND_USB		模拟地
K2	X_GPIO1_11	IO	GPIO1 通用输入输出端口 11
K3	GNDK		数字地
K4	GNDK		数字地
K5	GNDK		数字地
K6	GNDK		数字地
K7	VCCK		1.0v 数字电源
K8	GNDK		数字地
K9	ETH1_TXCTL	O	GMAC1 TX 端控制信号输出
K10	ETH1_TXD2	O	GMAC1 TX 端数据输出端口 2
K11	ETH1_TXD3	O	GMAC1 TX 端数据输出端口 3
K12	X_GPIO1_1	IO	GPIO1 通用输入输出端口 1
L			
L1	X_I2C_SCL	IO	I2C SCL 信号
L2	GNDK		数字地
L3	VCC3318_ETH		GMAC 3.3v/1.8v 数字电源
L4	ETH1_RXD2		GMAC1 RX 端数据输入端口 2
L5	GNDK		数字地
L6	ETH1_RCK	I	GMAC1 RX 端时钟输入端口
L7	ETH1_RXD1	I	GMAC1 RX 端数据输入端口 1
L8	ETH1_RXCTL	I	GMAC1 RX 端控制信号输入端口
L9	VCC33IO		3.3v 数字电源
L10	X_UART0_TXD	O	UART0 TXD 信号
L11	X_UART0_RXD	I	UART0 RXD 信号
L12	X_GPIO_2	IO	GPIO1 通用输入输出端口 2
M			
M1	X_I2C_SDA	IO	I2C SDA 信号
M2	GNDK		数字地
M3	VCC3318_ETH		GMAC 3.3v/1.8v 数字电源
M4	ETH1_RXD3	I	GMAC1 RX 端数据输入端口 3
M5	GNDK		数字地
M6	ETH1_MDC	O	GMAC1 PHY 时钟输出
M7	ETH1_MDIO	IO	GMAC1 PHY 数据输入输出
M8	ETH1_RXD0	I	GMAC1 RX 端数据输入端口 0
M9	VCC33IO		3.3v 数字电源
M10	ETH1_TXD0	O	GMAC1 TX 端数据输出端口 0
M11	ETH1_TXD1	O	GMAC1 TX 端数据输出端口 1
M12	X_RST_N	I	系统复位信号, 低电平有效

## 2.4 管脚复用

在芯片内部，GPIO1、UART1、QSPI 等模块复用 16 根 IO 线，复用模式如下表 2.2 所示：FUNC\_MODE 通过 SCU 寄存器，可配置为 0、1 两种模式。

GPIO1[5:0]由 SCU 寄存器控制，在任何 FUNC\_MODE 下均可进行模式选择。

表 2.2 管脚复用表

FUNC_MODE[1:0]	0	1
接口信号	功能模式 0	功能模式 1
GPIO1[15]	GPIO1[15]	QSPI_SCK
GPIO1[14]	GPIO1[14]	QSPI_CS#
GPIO1[13]	GPIO1[13]	QSPI_TX (IO0)
GPIO1[12]	GPIO1[12]	QSPI_RX (IO1)
GPIO1[11]	GPIO1[11]	QSPI_WP# (IO2)
GPIO1[10]	GPIO1[10]	QSPI_HOLD# (IO3)
GPIO1[9]	GPIO1[9]	PWM0
GPIO1[8]	GPIO1[8]	GPIO1[8]
GPIO1[7]	GPIO1[7]	GPIO1[7]
GPIO1[6]	GPIO1[6]	GPIO1[6]
GPIO1[5]	GPIO1[5]/SPI_CS	GPIO1[5]/SPI_CS
GPIO1[4]	GPIO1[4]/SPI_CK	GPIO1[4]/SPI_CK
GPIO1[3]	GPIO1[3]/SPI_TXD	GPIO1[3]/SPI_TXD
GPIO1[2]	GPIO1[2]/SPI_RXD	GPIO1[2]/SPI_RXD
GPIO1[1]	GPIO1[1]/UART1_TXD	GPIO1[1]/UART1_TXD
GPIO1[0]	GPIO1[0]/UART1_RXD	GPIO1[0]/UART1_RXD

## 2.5 上电时序

为确保芯片内部逻辑与外部器件通讯正常，上电时应先供应 VCCK 类(1.0V)引脚电压，再对 IO 类 (3.3/1.8V) 进行供电，最差情况也要保证 VCCK 类电压和 IO 类电压同时上电。如下图所示：

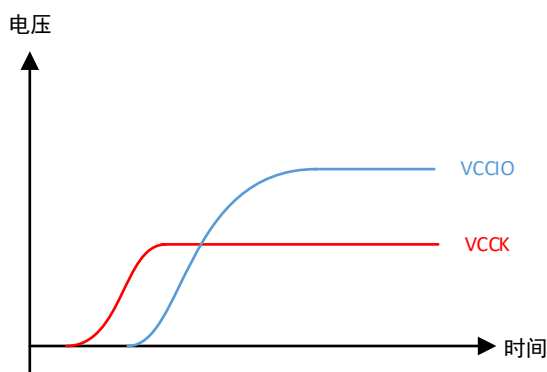


图 2.3 芯片上电时序要求图

## 2.6 电性能参数

表 2.3 电气特性参数

符号	描述	Min	Typ	Max	单位
VCC33IO	普通 IO 口电源	3.0	3.3	3.6	V
VCCCK	Core 电源	0.9	1.0	1.1	V
VCC10_PLL0	PLL 模拟电源	0.9	1.0	1.1	V
VCC33A_USB	USB 接口 IO 模拟电源	3.0	3.3	3.6	V
VCC10A_USB	USB 接口 core 模拟电源	0.9	1.0	1.1	V
VCC3318_ETH	网络接口 IO 数字电源	1.62/3.0	1.8/3.3	1.98/3.6	V
VCC33_OSC	系统晶振数字电源	3.0	3.3	3.6	V
VCC10A_SH	SATA HOST PHY 内核模拟电源	0.9	1.0	1.1	V
VCC11A_SH	SATA HOST PHY 内核模拟电源输出	1.0	1.1	1.2	V
VCC33A_SH	SATA HOST PHY IO 模拟电源	3.0	3.3	3.6	V

## 2.7 功耗

- 静态功耗 < 0.1W
- 动态功耗 < 1.5W

## 2.8 PCB 设计建议

请参考《T660 PCB 设计指南》

## 3 CPU 子系统

### 3.1 CK803S 处理器

#### 3.1.1 简介

CK803S 是面向控制领域的 32 位高效嵌入式 CPU 核，具有低成本、低功耗、高代码密度等多种特点。CK803S 采用 16/32 位混合编码指令系统，设计了精简高效的 3 级流水线。

CK803S 提供多总线接口，支持系统总线、指令总线、数据总线的灵活配置。CK803S 针对内存拷贝应用做了特殊优化，可以获得极致的内存拷贝性能。此外，CK803S 对中断响应做了特殊加速，中断响应延时仅需 13 个周期。

#### 3.1.2 特性

- 精简指令集（RISC）处理器架构
- 32 位数据，16 位/32 位混合编码指令
- 16 个 32 位通用寄存器
- 3 级流水线
- 最高工作频率 260Mhz
- 单位性能 1.5DMIPS/MHz
- 按序发射、按序执行、按序退出
- 支持 AHB 系统总线和 AHB Databus 总线接口
- 内置 16KB 高速缓存
- 32KB DTCM
- 内置 8 个内存保护单元
- 内置紧耦合矢量中断控制器与计时器
- 支持 1:1 和 2:1 处理器与系统时钟比
- 中断响应延时仅为 13 个处理器周期
- 静态分支预测
- 支持硬件乘除法
- 支持连续内存访问
- 仅支持 little endian

#### 3.1.3 架构



### 3.1.5 系统计时器

系统计时器 Core Timer 是 CK803S 内部集成的一个紧耦合模块，主要用于计时。Core Timer 提供了一个简单易用的 24 位循环递减的计数器，当 Core Timer 使能时，计数器开始工作，当计数器递减到 0 时，会向矢量中断控制器发起中断请求，申请获得处理器响应并处理 Core Timer 的事务。

Core Timer 的结构框图如图所示：

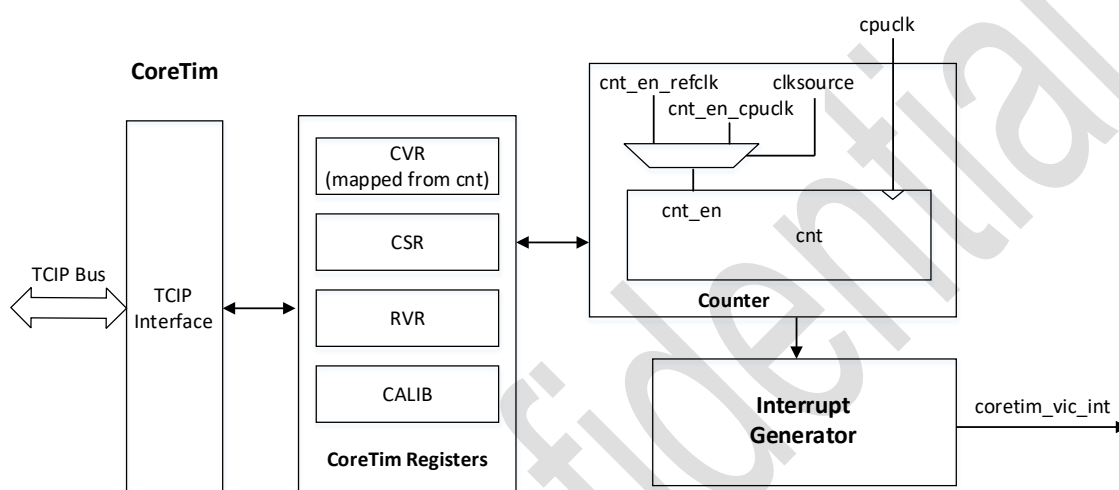


图 3.3 CoreTimer 结构框图

## 3.2 存储

芯片内部包含 4 块存储单元：ROM、Nor flash 和 2 块 SRAM。

内置 32KB ROM 固化了 Bootrom 程序，用于上电固件引导及固件下载，用户无法修改；

内置 512KB Nor flash，可用于存储固件代码及用户敏感信息；

Nor flash 主要参数如下：

- 页大小：512B
- 8/16/32bit 读、32bit 写
- 擦写次数：10 万次

内置 1 片 8KB SRAM 和 1 片 256KB SRAM，8KB SRAM 用于存储 SATA Host、SATA Device 和 GMAC 等模块的命令链表，用户不能使用该存储空间；256KB SRAM 可供用户使用，用于高速固件代码执行、临时数据存储和算法运算等。8KB SRAM 和 256KB SRAM 各自拥有独立 AHB 和 AXI 接口访问通道，可大大提升 AHB 和 AXI 间数据搬运效率。

## 3.3 DMA

### 3.3.1 模块概述

DMA (Direct Memory Access) 是为了降低 CPU 负担专门用来进行数据搬运的模块。在 T660 中，单纯的 DMA 模块只在 AHB 总线上集成了一个，如果 AXI 总线上需要进行数据搬运，可以通过 CRYPTO 模块中的 DMA 实现。

DMA 模块架构如下：

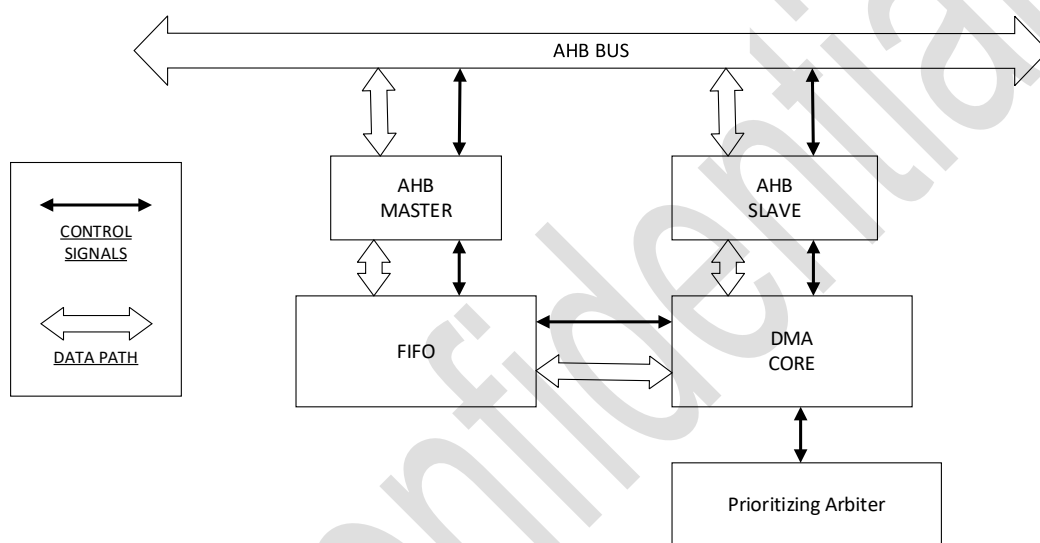


图 3.4 DMA 结构框图

### 3.3.2 模块特性

- 支持 8 路可配 DMA 通道
- 通道共享 16 个字节 buffer
- 支持链表模式传输
- 可在 AHB、AXI、APB bus 间进行数据搬运
- 支持 8/16/32 位数据传输
- 仅支持 little-endian 传输
- 支持 INCR 和 FIXED 地址传输模式

## 3.4 定时器

### 3.4.1 模块概述

定时器模块挂载于 APB 总线上，可提供 8 个独立的计数器，用于生成定时中断给 CPU 进行定时任务处理。同时定时器模块可生成并输出一路 PWM 信号，用于芯片外设时钟或者电机类设备的控制。

模块框图如下：

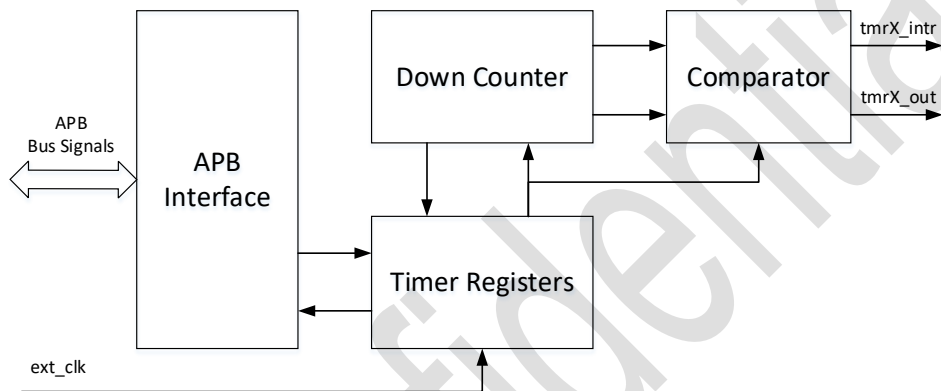


图 3.5 定时器结构框图

### 3.4.2 模块特性

- 支持 8 个独立的 32 位计数器
- 支持一路 PWM 输出，最高频率 20Mhz
- PWM 极性和占空比可配
- 支持自动加载模式

## 3.5 看门狗

### 3.5.1 模块概述

看门狗模块用于防止芯片固件跑飞或部分硬件造成的系统卡死情况，一旦发生上述情况，看门狗可以产生硬件复位，让整个芯片重新复位启动。

看门狗模块结构如下：



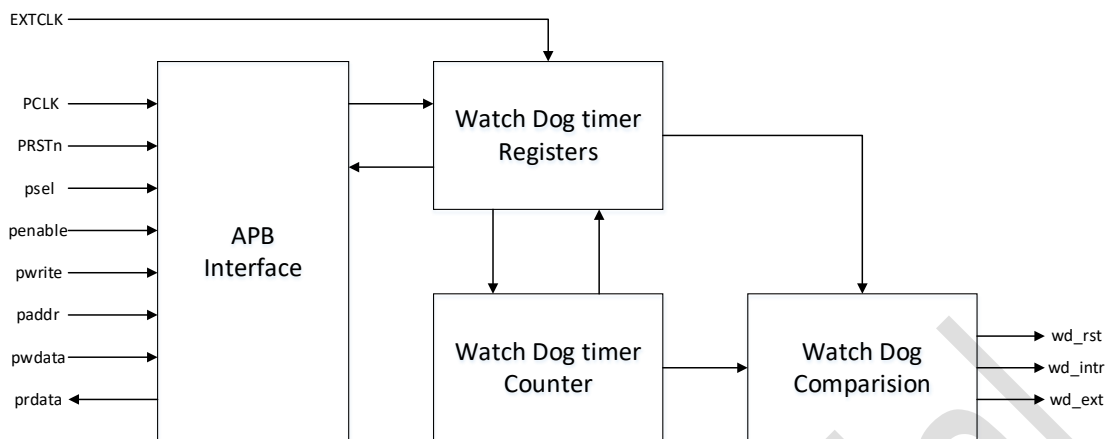


图 3.6 看门狗结构框图

### 3.5.2 模块特性

- 支持一路系统复位输出
- 复位输出时间可配置
- 支持一路 CPU 中断输出
- 内置 32 位递减计数器

## 3.6 SCU

### 3.6.1 模块概述

SCU 模块是系统控制单元，主要对芯片时钟、复位、功耗等芯片级配置进行控制。SCU 模块架构如下：

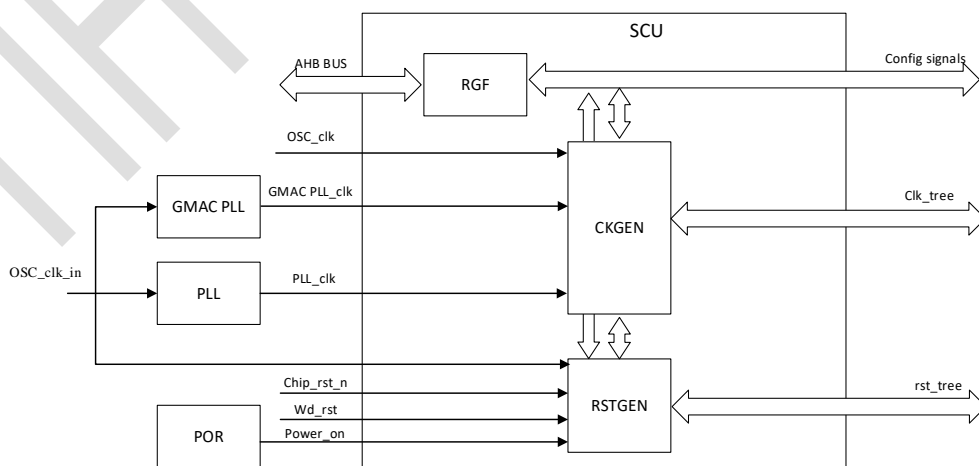


图 3.7 SCU 结构框图

### 3.6.2 模块特性

- 支持各模块时钟分频及门控
- 支持各模块复位控制
- 支持 PLL 输出频率可配
- 支持 PLL、OSC 时钟切换
- 支持管脚复用配置
- 内置看门狗复位状态寄存器

### 3.6.3 时钟树

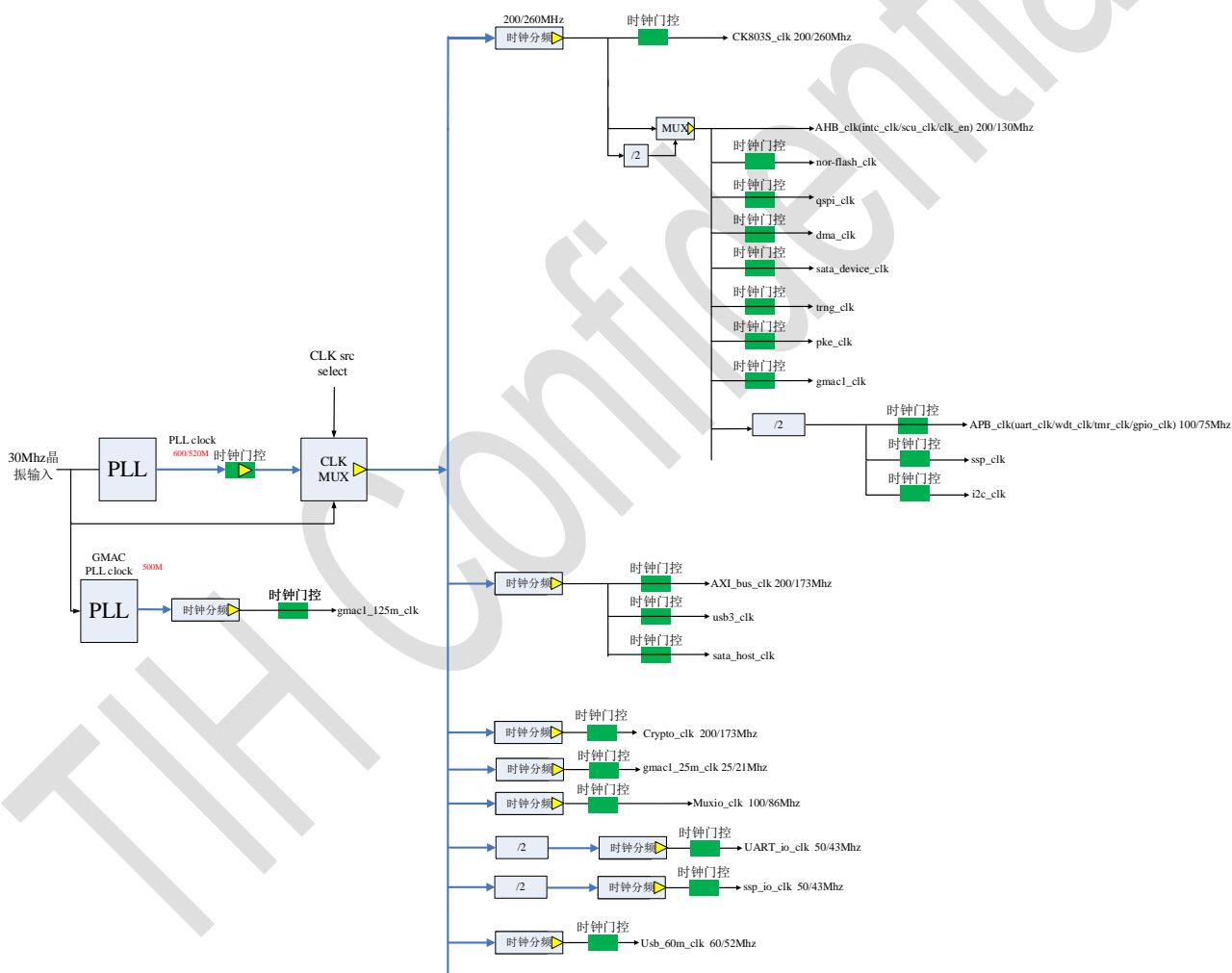


图 3.8 系统时钟树

### 3.6.4 复位树

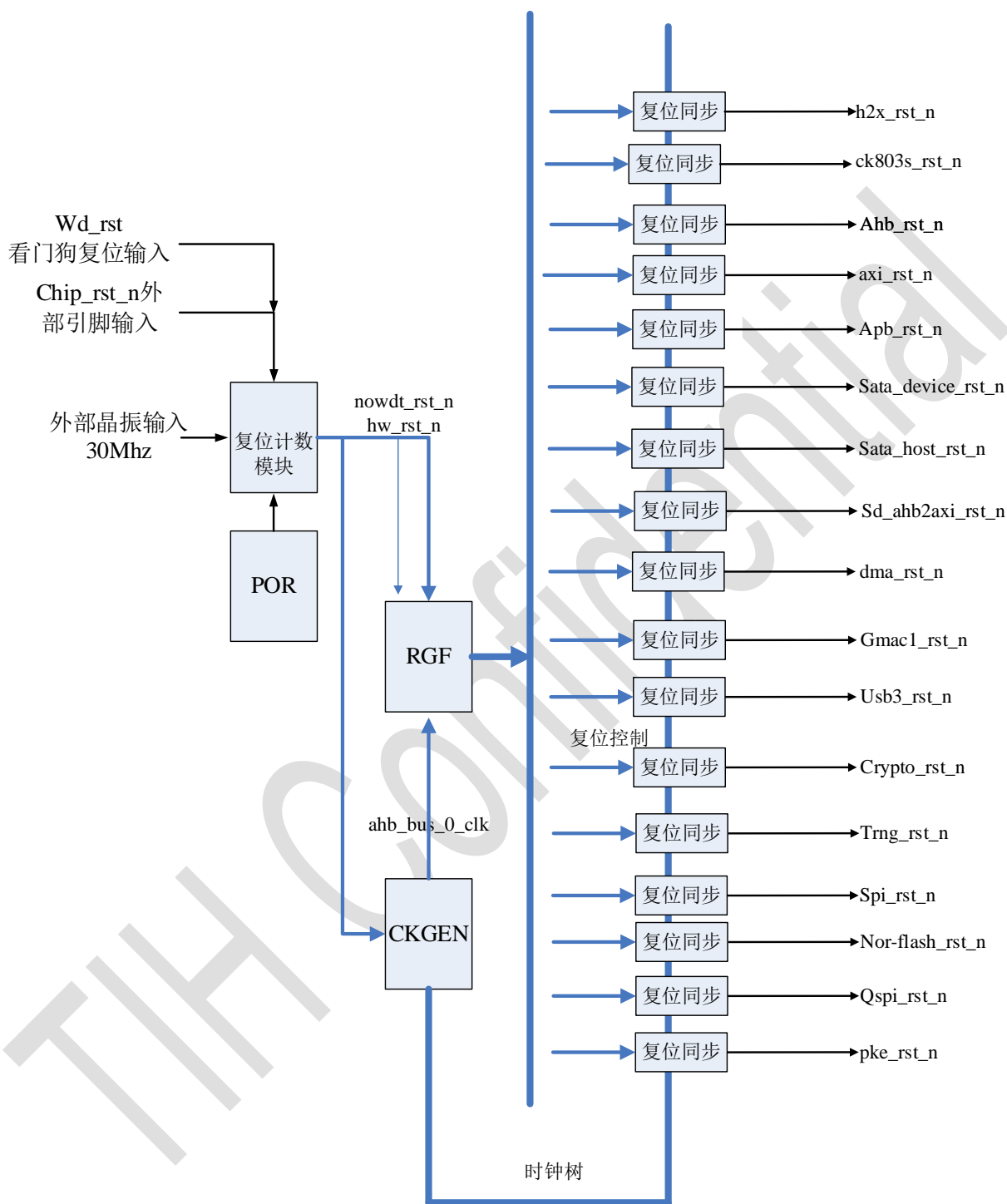


图 3.9 系统复位树

## 4 安全引擎

### 4.1 CRYPTO 引擎

#### 4.1.1 模块概述

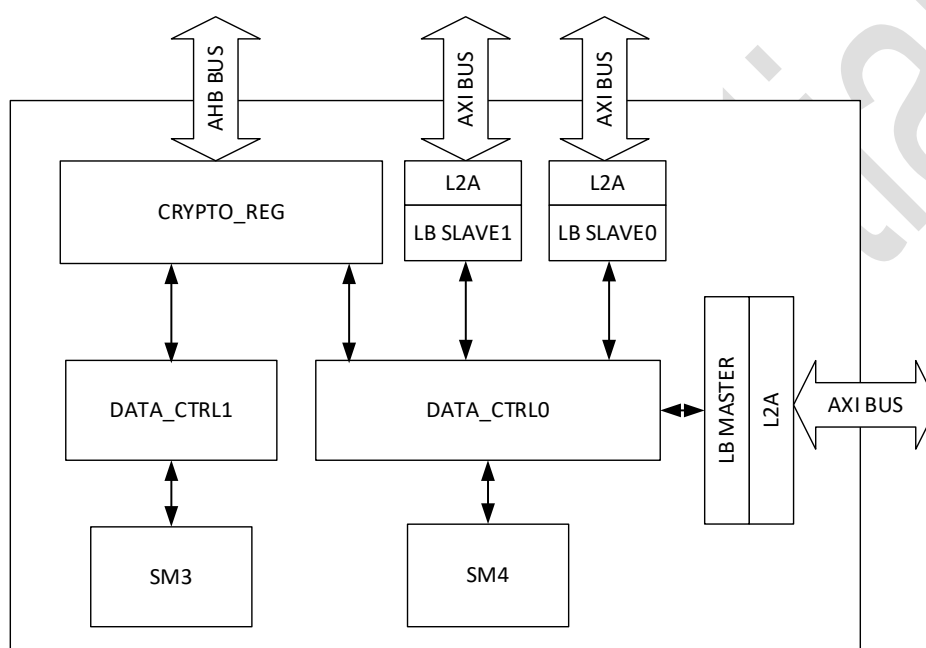


图 4.1 CRYPTO 引擎结构框图

CRYPTO 模块的作用主要是对进入模块的数据进行加解密操作。目前支持 SM4 和 SM3 算法。两者之间相互独立，SM4 数据走 AXI 总线，SM3 数据走 AHB 总线。当数据从 AXI SLAVE 端口、AXIMASTER 端口或者 AHB SLAVE 端口进入 CRYPTO 模块之后，CPU 通过 AHB 总线配置 CRYPTO 模块寄存器，选择对应功能，直至加解密结束。

#### 4.1.2 模块特性

- 支持一路 AHB SLAVE 配置端口
- 支持一路 AXI MASTER 端口
- 支持两路 AXI SLAVE 数据端口
- 支持 SM4 KEY 128 bit
- 支持 BYPASS 模式
- 支持数据填充和舍弃功能
- 对称加密模式支持数据输入输出端口选择



- 支持 ECB/CBC/CFB/OFB/CTR/XTS 六种操作模式
- 支持 SM4  $\geq 800\text{MB/s}@200\text{MHz}$  (ECB/CTR/XTS)
- 支持 SM3  $\geq 80\text{MB/s}@200\text{MHz}$

### 4.1.3 工作方式

#### 1) BYPASS 模式

数据从 CRYPTO 模块流过，不做任何处理，输入输出相同。

- a) 配置控制寄存器选择 bypass 模式
- b) 配置中断使能寄存器（根据需求）
- c) 配置数据流向寄存器选择输入输出端口
- d) 配置数据长度寄存器
- e) 配置开始寄存器开启数据传输
- f) 等待数据传输完成

#### 2) FIFO 模式

将 CRYPTO 模块看成一个带有加解密功能的 FIFO，数据从一个 slave 端口写入，另一个 slave 端口取走（对应 FIFO 的读写端口）。上层 MASTER 将数据写入 CRYPTO 模块之后，对数据进行加解密操作，然后再由上层 MASTER 将已加解密完数据取走。

- a) 配置控制寄存器选择密码算法、加/解密、算法模式、数据大小端、密钥类型，如果选择 CTR 模式还需配置步长寄存器
- b) 配置中断使能寄存器（根据需求）
- c) 配置密钥以及初始值寄存器
- d) 配置数据流向寄存器，选择一个 AXI SLAVE 端口或两个 AXI SLAVE 端口（推荐使用两个，便于理解）
- e) 配置数据长度寄存器
- f) 配置开始寄存器，开始进行密钥扩展
- g) 等待密钥扩展完成，配置开始寄存器开启数据传输
- h) 等待数据传输完成

#### 3) BRIDGE 模式

由一个 MASTER 端口和一个 SLAVE 端口组成。可分为两种模式，一种是正常模式，另一种是 LLI 模式。正常模式下，只需要配置一次源地址或者目的地址，LLI 模式下可以将不同的源地址或者目的地址写入命令 FIFO 中，模块会自动根据 FIFO 中的命令去执行操作（MASTER 读是配置源地址，写配置目的地址）。两种模式下的数据长度寄存器是有区别的，正常模式下按照已配置好的数据长度操作，LLI 模式下按照写入 FIFO 中的命令数据长度操作（FIFO 中命令的数据总长度等于已配置的数据长度）。

- a) 配置控制寄存器选择密码算法、加/解密、算法模式、数据大小端、密钥类型以及是否使用 LLI 模式，如果选择 CTR 模式，则还需配置步长寄存器
- b) 配置中断使能寄存器（根据需求）



- c) 配置密钥以及初始值寄存器
- d) 如果配置了 LLI 模式，则需要向 LLI 寄存器中写入命令，如果没有则跳过
- e) 配置数据流向寄存器，选择一个 AXI SLAVE 端口和一个 AXI MASTER 端口
- f) 根据需求配置源地址或目的地址寄存器以及 MASTER 控制寄存器
- g) 配置数据长度寄存器
- h) 配置开始寄存器，开始进行密钥扩展
- i) 等待密钥扩展完成，配置开始寄存器开启数据传输
- j) 等待数据传输完成

#### 4) DMA 模式

由一个 MASTER 端口来完成读写操作。CRYPTO 模块会根据已配置的源地址及数据长度取数据进行加解密，然后将加解密后的数据写入对应目的地址。

- a) 配置控制寄存器选择密码算法、加/解密、算法模式、数据大小端、密钥类型，如果选择 CTR 模式，则还需配置步长寄存器
- b) 配置中断使能寄存器（根据需求）
- c) 配置密钥以及初始值寄存器
- d) 配置数据流向寄存器，选择一个 AXI MASTER 端口
- e) 根据需求配置源地址或者目的地址寄存器以及 MASTER 控制寄存器
- f) 配置数据长度寄存器
- g) 配置开始寄存器，开始进行密钥扩展
- h) 等待密钥扩展完成，配置开始寄存器开启数据传输
- i) 等待数据传输完成

#### 5) 哈希算法模式

- a) 配置控制寄存器选择加密模式以及大小端
- b) 配置中断使能寄存器（根据需求）
- c) CPU 向数据寄存器写入数据（512bit）
- d) 检测加密核的状态是否处于忙状态
- e) 如果检测到加密核的状态处于空闲状态，则继续向数据寄存器写入数据（如果是最后一笔则将控制寄存器的第 5 位使能，再继续向数据寄存器写入数据），重复该步骤直到数据输入完成
- f) 等待传输结束，将最终结果从数据寄存器取走

## 4.2 PKE 引擎

### 4.2.1 模块概述

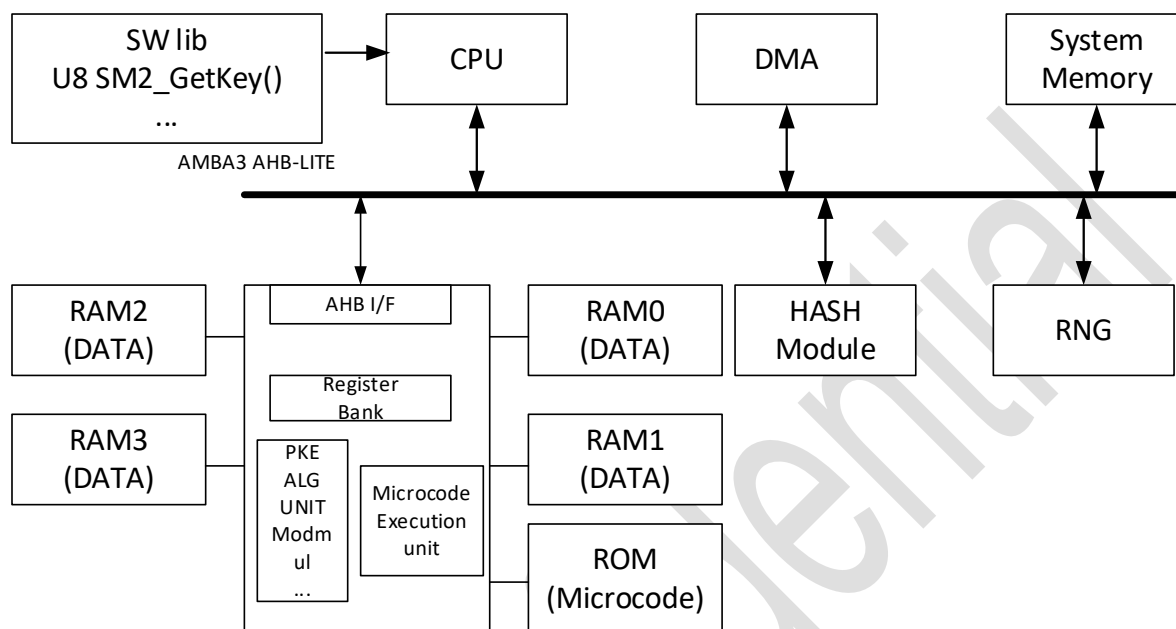


图 4.2 PKE 引擎结构框图

Public Key Engine (PKE) 用来加速公钥密码运算中的大数模运算。公钥密码的运算过程中，存在大量的大数模运算，对于普通的嵌入式 CPU 而言，完成这些大数模运算将会花费大量 CPU 指令，效率极低，因此在大多数支持公钥密码运算的芯片中都会加入公钥密码加速模块来完成公钥密码的运算。PKE 用来加速公钥密码中 SM2 所涉及到的大数模运。CPU 可以通过轮询或者中断方式来查询 PKE 的工作情况。

PKE 包含 AHB 接口模块 (AHB I/F)、寄存器组模块、大数运算单元、微码运行单元 (MEU)。另外，PKE 模块需要四块 RAM 和一块 ROM，可根据不同寄存器配置完成不同精度的运算。

### 4.2.2 模块特性

- 支持 SM2 签名验签，性能 $\geq 1200/600$  次/s@200Mhz
- 支持一路 AMBA 3 AHB-Lite 接口

### 4.2.3 工作方式

PKE 的运算通过微码 (Microcode) 形式完成，微码存储在程序存储单元中。因此通过向程序存储单元中灌入不同微码来实现不同要求的公钥密码运算。例如，在一个安全性要求较

高的 SoC 中，可以向 PKE 模块中的程序存储单元灌入高安全性的公钥算法指令。在一些性能优先的设计中，可以向 PKE 模块中的程序存储单元灌入性能优化的公钥算法指令，实现性能优先的目的。在程序存储单元容量较大的设计中，可以将这些运算指令都写入 ROM，由 CPU 根据不同的使用场景进行实时调用，完整的微码大小大约为 2KB。

PKE 接口被映射到 7KB 地址空间内。这一块地址映射空间内主要包含 CPU 可以访问的所有操作数，这些操作数包含了模数、幂指数、部分中间变量等。除此之外，该地址映射空间内也包含控制和状态寄存器。CPU 可以通过控制寄存器和状态寄存器来配置、监控 PKE 模块。

PKE 支持的运算中，运算数最小为 192 比特，因此，CPU 或 DMA 将数据放入数据 RAM 中会遇到字间大小端问题。在 PKE 模块中，字与字之间都是按照小端进行排列的，下一个部分会给出具体的例子。

PKE 中，最小的操作数为 256 比特（4 个双字），因为目前 ALU 的输入位宽为 256 比特，如果操作数不是字对齐的，需要将高位补零。

PKE 接到开始命令后，开始进行运算，运算过程中，上位机可以通过状态寄存器查询目前的运行状态，也可以通过控制寄存器来中断目前的运行。另外，通过访问数据 RAM 地址可以获得部分中间运算结果。

上位机可以通过轮询或中断的方式来获取 PKE 是否完成目标运算的结果。数据 RAM 都是双字（64-bit）对齐，不支持字节对齐。

## 4.3 TRNG

### 4.3.1 模块概述

TRNG 模块通过物理随机源产生随机序列，后经 SM4 均衡处理，生成真随机数，为 SM2 等非对称算法提供密钥对。

### 4.3.2 模块特性

- 符合 GM/T 0005-2012《随机性检测规范》
- 符合 NIST SP800-90 a/b/c 的要求
- 集成 4 路物理随机源
- 具有在线健康检测功能
- 随机数生成速率 $\geq 30$ Mbps



## 5 网络接口

### 5.1 GMAC1 控制器

#### 5.1.1 模块概述

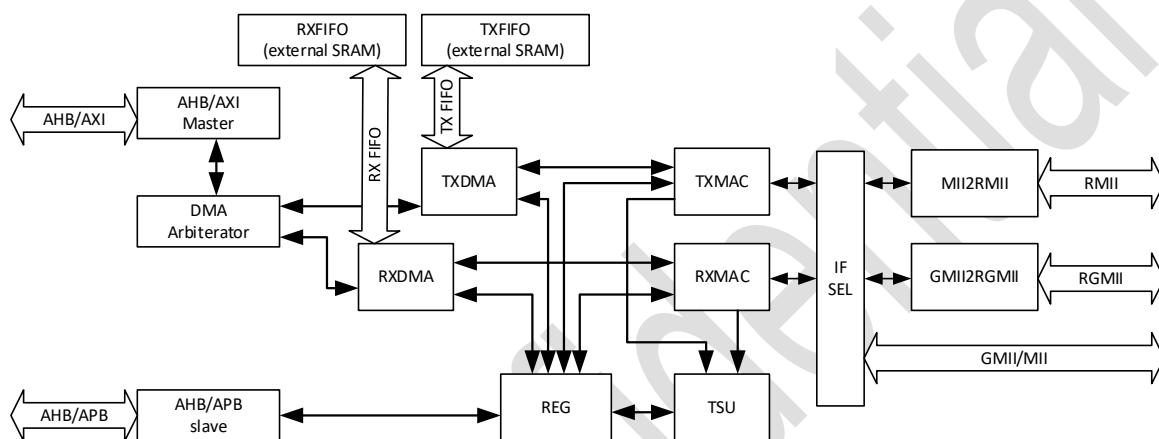


图 5.1 GMAC 控制器结构框图

\*注：本芯片中仅支持 RGMII 接口

GMAC 控制器是一个高效以太网控制器，完全兼容 IEEE 802.3 规范，包括 AHB/AXI 接口、DMA 引擎、数据缓冲区(TXFIFO 和 RXFIFO)、MAC 和接口逻辑等模块。

- 1) DMA 引擎负责系统内存和数据缓冲区之间的所有数据传输，可以减少 CPU 负载，使芯片整体性能更优；
- 2) 数据缓冲区(TXFIFO 和 RXFIFO)用于帧的发送和接收，可降低对系统内存的响应时间要求；
- 3) RGMII 接口基于 HP RGMII 规范 2.0 版本，支持 10Mbps、100Mbps 和 1000Mbps 三种数据速率；
- 4) GMAC 控制器实现了 TCP、UDP、IPv4 和 IPv6 checksum 生成和验证，并支持 VLAN 标记,可有效减少 CPU 运算负载；GMAC 控制器同时支持局域网唤醒功能，支持三个唤醒事件：连接状态更改，魔术包和自定义掩码帧；

#### 5.1.2 模块特性

- 支持 DMA 数据传输
- 支持全双工、半双工操作



- 支持全双工、半双工流控
- 内置 4KB TX FIFO 和 8KB RX FIFO
- 支持最大 4KB 的巨型帧
- 支持 TCP/UDP/IPV4/IPv6 checksum offloads
- 支持 IEEE 802.1Q VLAN 标签植入和移除
- 支持 Wake-On-LAN 功能和三种唤醒方式
- RGMII 接口支持 10Mbps/100Mbps/1000Mbps 传输速率

### 5.1.3 工作方式

#### (1) 发送描述符和数据缓冲区

GMAC 控制器使用环形描述符队列来进行发送管理，发送描述符和数据包都存放于系统内存中。GMAC 控制器根据发送描述符内容，将发送数据包从系统内存中转移到 GMAC 控制器的 TX FIFO 中，然后将数据包发送到以太网。

GMAC 控制器支持两个用于发送的描述符环，分别是普通优先级传输环和高优先级传输环，普通优先级传输环用于普通分组传输，高优先级传输环用于高优先级分组传输，具有高优先级的包可以放入高优先级传输环中，以便更快地传输。

#### (2) 接收描述符和数据缓冲区

GMAC 控制器使用环形描述符队列来进行接收管理，接收描述符和数据接收区都位于系统内存中。GMAC 控制器首先将接收到的数据包存储在 RXFIFO 中，然后根据接收描述符内容，将 RXFIFO 中的数据包移动到系统内存中。

#### (3) Wake-On-LAN

GMAC 控制器支持三种方式唤醒：

##### 1) 连接状态改变

连接状态改变是指以太网的连接状态发生变化。如果 GMAC 控制器启用了连接状态更改模式，连接状态更改将被视为唤醒事件。

##### 2) 魔术包

一个魔术包包含一个特定序列和 16 个连续的本地 MAC 地址，特定的序列由 6 字节的 0xFF 组成。魔术包的格式如下：

DA + SA + 6 \* (0xFFh) + 16 \* (network adaptor's node address) + ...

如果 GMAC 启用了魔术包模式，魔术包会被看做唤醒事件。

##### 3) 自定义掩码帧

GMAC 控制器进入自定义掩码帧模式之前，系统应该将自定义掩码帧列表传递给驱动程序，将对应位的字节掩码设置为 1，每个字节掩码决定了将要传入帧的哪些字节应该与相应的自定义掩码帧进行比较，以确定是否为唤醒事件。

此外，自定义掩码帧的数据包长度必须至少为 128 字节。

#### (4) 流控

GMAC 控制器实现了流控功能，支持 IEEE802.3x 全双工流控模式和半双工流控模式。



全双工模式采用 IEEE802.3x 流控。当 A 和 B 在全双工模式下相互发送和接收数据包时，如果 B 中 RX FIFO 接近满，B 会向 A 发送一个含有暂停时间的帧，以避免数据丢失。然后 A 会在指定时间段内暂停发送数据，B 在指定的时间段内处理已接收数据，暂停时间结束后，A 继续向 B 发送数据包。

半双工操作采用 Back Pressure 方式，当 A 以半双工模式发送和接收数据包时，如果 A 中的 RX FIFO 接近满，A 会发送一个 jam 模式来产生冲突，以避免传入的数据包被保存到 RX FIFO 中，A 需在一段时间内尽快处理接收到的数据，当 RX FIFO 不接近满可以继续接收数据包时 A 不会发送 jam 包。

## 6 USB OTG 接口

### 6.1 模块概述

USB OTG 接口是通用串行总线双功能设备控制器，支持 USB2.0/USB3.0 及可扩展主机控制器接口 (XHCI) 协议，可以通过寄存器配置选择来切换不同的功能。模块的功能框图如下：

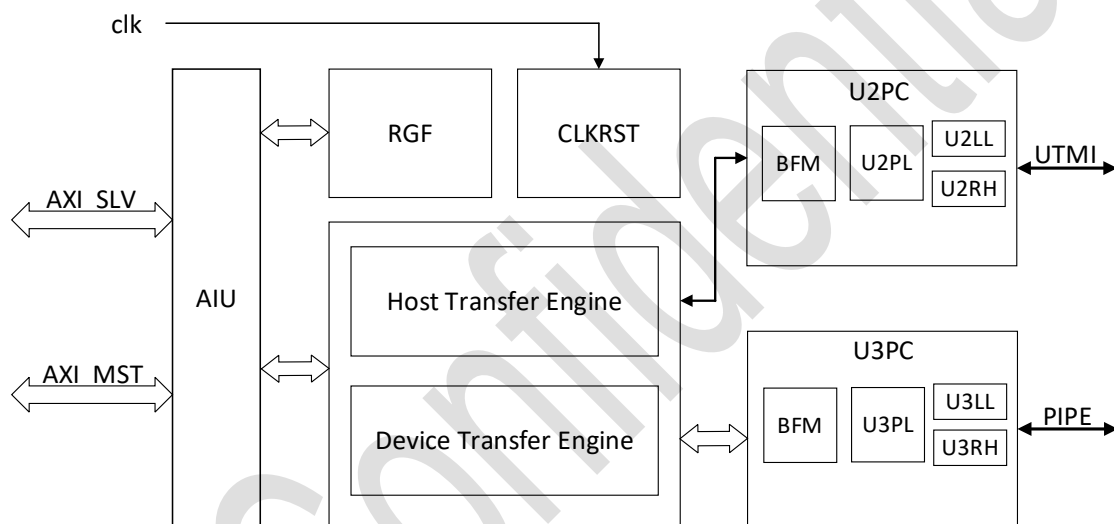


图 6.1 USB 内部结构框图

当 USB OTG 接口作为主机时，控制器根据系统软件准备的数据结构向 USB 设备发出传输请求，控制器支持 XHCI1.0 作为系统软件开发的接口。

当 USB OTG 接口作为设备时，控制器响应来自主机的传输请求，内置 9 个端点，端点使用数目可由控制器中的专用缓冲空间配置，端点缓冲空间与系统内存之间的数据传输可以通过内部 DMA 或外部 DMA 完成。

### 6.2 模块特性

- 支持可扩展主机控制器接口协议 1.0(XHCI1.0)
- 静态角色转换（主机/设备选择）
- 主机和设备模式下支持所有的 USB 传输类型，包括控制/批量/中断/等时传输
- 支持优异的功耗管理，USB3.0 模式下支持 U0/U1/U2/U3，USB2.0 模式下支持 LPM
- 主机和设备模式都支持 DMA 传输
- 主机模式下支持上下文缓存以减少等待时间



- 
- 设备模式下支持 9 个端点
  - 设备模式下每个端点的 FIFO 深度可配置
  - 设备模式下支持大批量数据流协议

TIH Confidential

## 7 SATA 接口

### 7.1 SATA Host 控制器

#### 7.1.1 模块概述

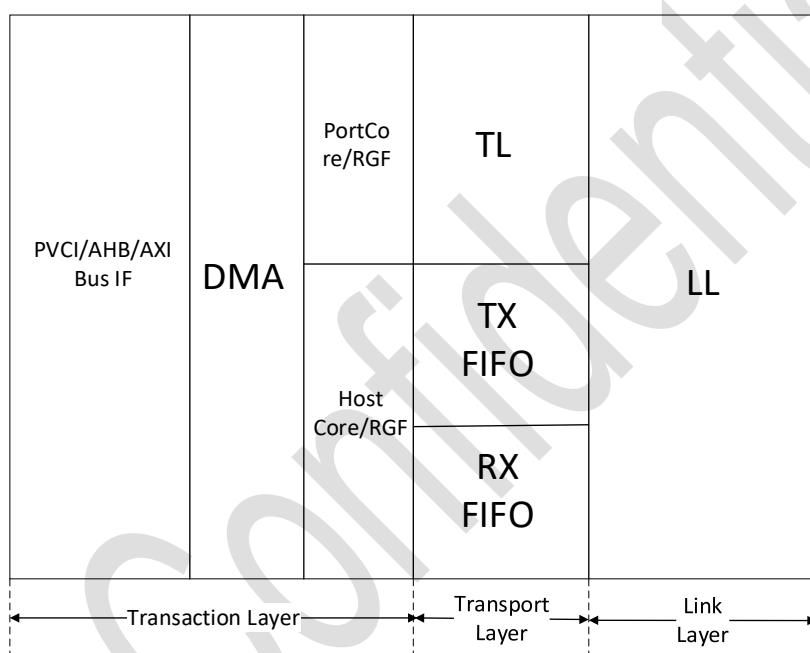


图 7.1 SATA Host 控制器结构框图

SATA host 控制器包含三个协议层：事务层，传输层和链路层。

事务层包含用户总线接口（AHB, AXI, PVCi），DMA 主控，端口/主机寄存器文件，以及端口/主机控制器：

- 1) DMA 控制器负责帧信息结构在系统内存和传输层 TXFIFO/RXFIFO 之间的发送和接收；
- 2) 主机核（Host core）负责全局复位，全局中断以及如 AHCI v1.1 协议中定义的每个端口命令完成合并，无论多少个端口被定义都只有一个主机核；
- 3) 主机寄存器文件（Host register file）是按照 AHCI v1.1 协议实现的，主机寄存器文件会提供实现的特性，比如对 HBA 的控制以及收集所有端口的状态反馈；
- 4) 端口核（Port core）负责处理 HBA 和已连接设备之间的事务。端口核请求 DMA 控制器从设备中获取一条命令或者数据 FIS 并将从设备中获取的 FIS 存储在系统内存中；

5) 端口寄存器文件 (Port register file) 与主机寄存器文件都是按照 AHCI v1.1 协议实现, 端口控制器文件负责控制端口并反映端口上的事务和连接状态。

传输层负责构建需要发送的帧信息结构 (FIS) 并解析接收到的帧信息结构, 传输层包含三个主要的部分: FIS 发送器, FIS 接收器和 BIST 控制器模块。

链路层负责发送和接收帧, 发送原语是基于传输层的控制信号, 而接收原语来自物理层并且会转化成控制信号到传输层:

- 1) 数据的发送先经过 CRC 校验, 加扰, 8B/10B 转码, 速度调整, 然后发给物理层;
- 2) 数据的接收则经过数据调整, 10B/8B 转码, 解扰和 CRC 校验然后到传输层。

### 7.1.2 模块特性

- 符合 Serial ATA Revision 3.0 标准协议
- 符合 AHCI1.1 协议
- 支持数据传输速率 1.5Gbps, 3.0Gbps 和 6.0Gbps
- 支持原生命令队列 (NCQ)
- 支持命令列表覆盖特性
- 支持 PIO 针对多 DRQ 块
- 支持自动局部休眠功能
- 集成 TX FIFO 深度为 256 words
- 集成 RX FIFO 深度为 256 words
- 支持 PHY 数据位宽 20bit 或 40bit 可配

## 7.2 SATA Device 控制器

### 7.2.1 模块概述

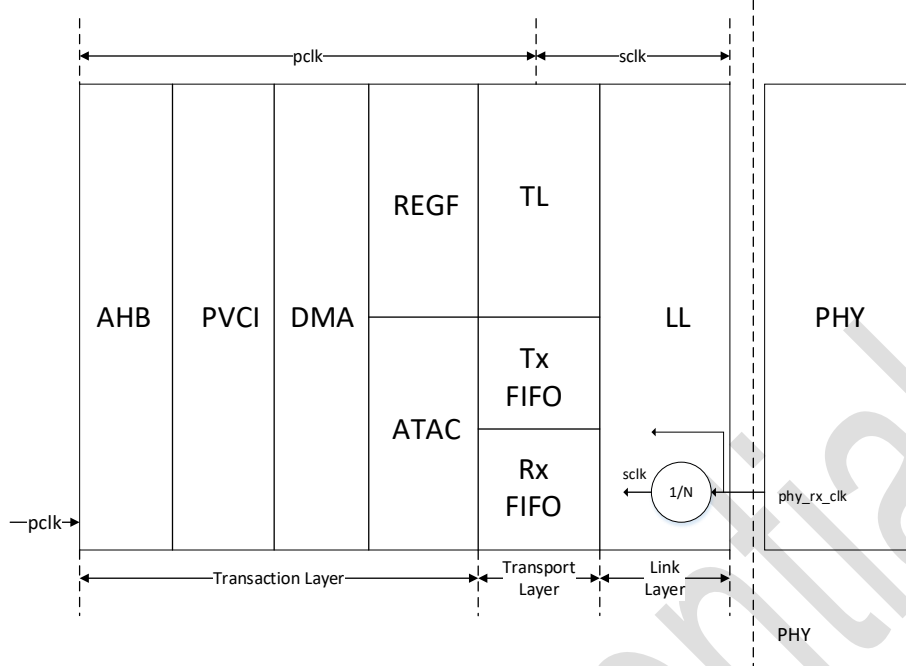


图 7.2 SATA Device 控制器结构框图

SATA device 控制器是一个串行 ATA 从设备控制器，符合 Serial ATA Revision 3.0 标准协议。SATA device 控制器支持 1.5Gbps/3.0Gbps/6.0Gbps 三种数据传输速率，系统总线为 32 位 AHB 总线。SATA device 控制器包含三层：事物层、传输层和链路层，事物层包含 AHB 接口，ATA 控制器，内置 DMA 控制器以及从设备控制寄存器，传输层包含 TX 和 RX FIFO,通过自定义接口与外部 PHY 相连接。

## 7.2.2 模块特性

- 符合 Serial ATA Revision 3.0 标准协议
- 支持 1.5Gbps,3.0Gbps,6.0Gbps 数据传输速率
- 支持 EXECUTE DEVICE DIAGNOSTIC, Non-data, DMA, PIO, NCQ 命令协议
- 支持 PIO multiple-DRQ 块
- 集成 TX FIFO 深度为 256 words
- 集成 RX FIFO 深度为 256 words
- 支持 PHY 数据位宽 20bit 或 40bit 可配
- 支持 NCQ 命令协议下 FDMA 声明单个 setup FIS
- 支持远端重定时环路
- 支持局部自动睡眠切换



## 8 外围设备接口

### 8.1 I2C 控制器

#### 8.1.1 模块概述

I2C 控制器挂载于 APB bus 上，可作为 I2C master 外接 I2C 接口设备，或作为 I2C slave 外接 MCU。

I2C 控制器框图如下：

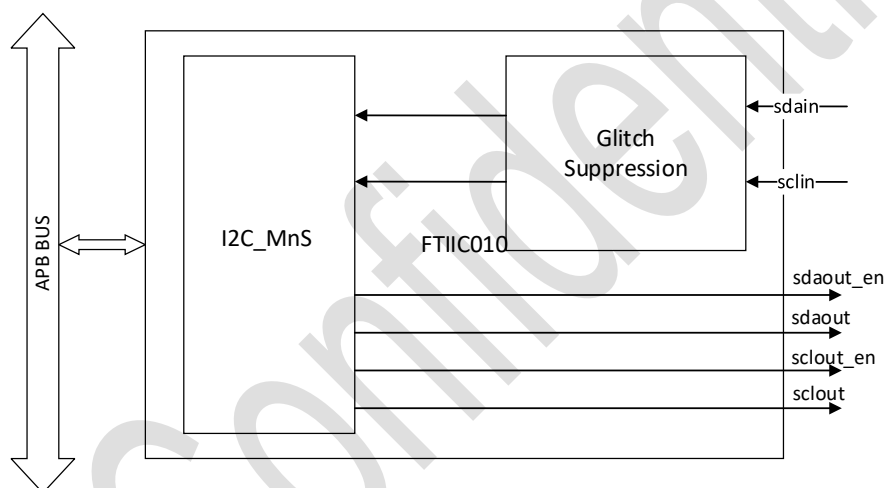


图 8.1 I2C 控制器结构框图

#### 8.1.2 模块特性

- 支持 Standard, Fast and Fast+ modes
- 支持 HS-mode
- 支持 7/10-bit 地址模式
- 支持总线毛刺过滤
- 支持主从模式
- 从模式地址可配
- 支持 master-TX, master-RX, slave-TX, slave-RX 模式
- 集成 32 byte 数据 buffer
- 支持 General Call 和 Start Byte 功能

### 8.1.3 工作方式

I2C 控制器可工作在如下模式：

- TX/RX in Slave Mode
- RX in Slave Mode with Repeat-Start
- TX in Master Mode
- RX in Master Mode
- TX in Master Mode with HS-Mode (or START Byte)
- RX in Master Mode with HS-Mode (or START Byte)
- TX/RX in Slave Mode with HS-Mode (or START Byte)
- Master TX Burst Mode
- Master RX Burst Mode

## 8.2 QSPI 控制器

### 8.2.1 模块概述

QSPI 控制器主要用于外扩 SPI SRAM、SPI flash 外设等。

QSPI 控制器架构如下：

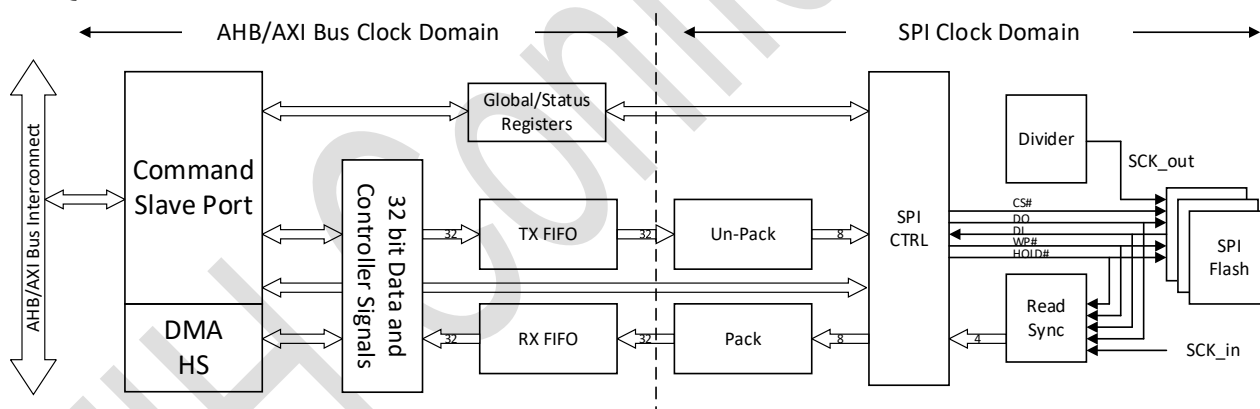


图 8.2 QSPI 控制器结构框图

### 8.2.2 模块特性

- 支持 Motorola SPI 协议标准
- 控制器时钟和接口时钟异步可调
- 支持 SPI 单线/双线/四线模式
- 最高接口工作频率 100 MHz

## 8.3 SPI 控制器

### 8.3.1 模块概述

SPI控制器挂载于APB总线上，符合Motorola总线协议，可作为SPI 主从设备进行外设扩展，操作简单、可扩展性强。

SPI模块结构如下：

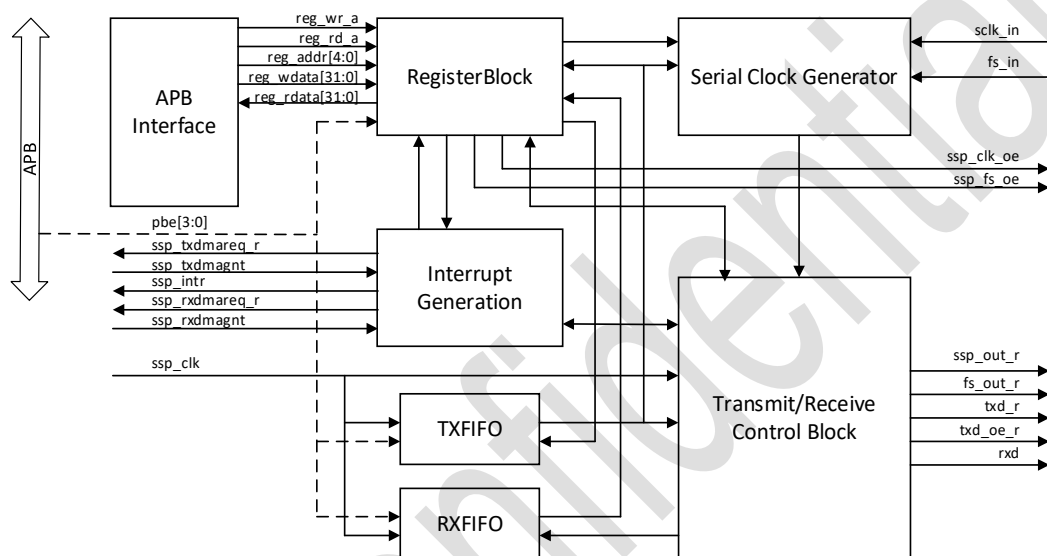


图 8.3 SPI 控制器结构框图

### 8.3.2 模块特性

- 支持 Motorola SPI 协议标准
- 最高接口工作频率 20 MHz
- 支持主从模式
- 输出时钟的极性、相位、频率可配
- 串行数据支持 MSB 或者 LSB first 模式
- 集成 32bytes TXFIFO
- 集成 32bytes RXFIFO
- TXFIFO/RXFIFO 阈值中断可配
- 支持中断和查询模式
- 独立的 SPI 工作时钟
- 独立可配置的中断使能

## 8.4 UART0 控制器

### 8.4.1 模块概述

UART0 控制器与通用的 16C550 UART 完全兼容。

UART0 控制器架构如下：

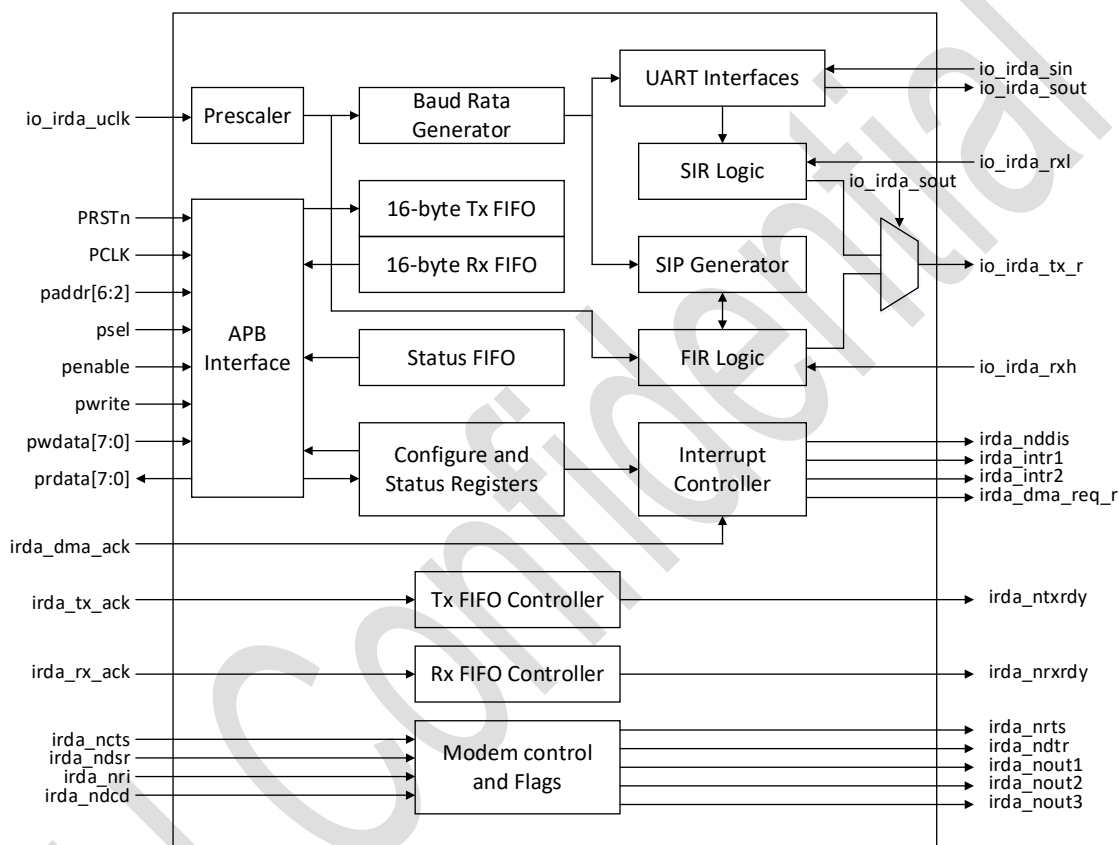


图 8.4 UART 控制器结构框图

\*注：上图中 Irda 功能本芯片中未支持

### 8.4.2 模块特性

- 完全兼容高速 NS 16C550A UART
- 最高波特率为 3Mbit/s
- 集成 32bytes TX FIFO
- 集成 32bytes RX FIFO
- 支持奇偶校验方式或无校验
- 支持帧错误检测

- 支持 FIFO 溢出报警
- 波特率可配置
- 支持数据位和停止位的位宽配置，数据位宽可配置为 5/6/7/8bits，停止位可配置为 1/1.5/2bits

## 8.5 UART1 控制器

UART1 控制器与 UART0 控制器内部结构及逻辑完全相同，只是基地址不同。

## 8.6 GPIO1 控制器

### 8.6.1 模块描述

GPIO1 提供 32 位可编程的输入输出管脚。每个管脚可配置为输入或输出。管脚用于生成特定应用的输出信号或采集特定应用的输入信号。输入管脚，GPIO 可作为中断源；输出管脚，每个 GPIO 都可以独立地清 0 或置 1。

GPIO1 的 32 个管脚输入状态下也可以根据电平或跳变值产生可屏蔽中断。

GPIO1 模块结构图如下：

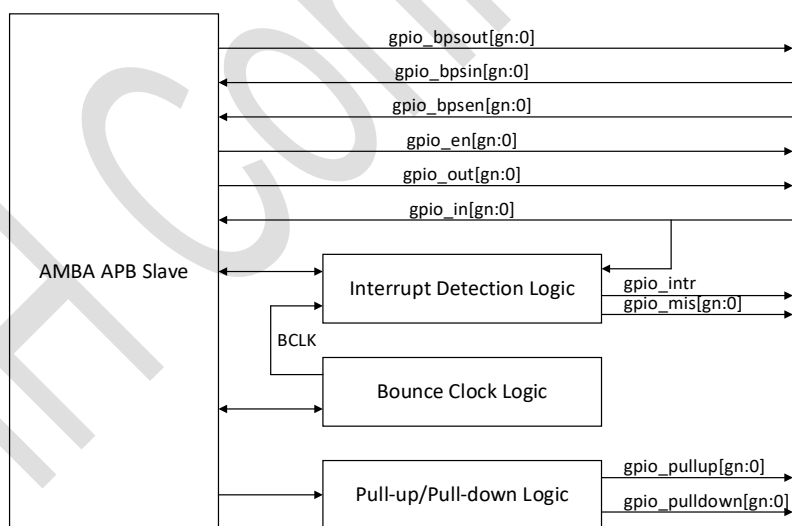


图 8.5 GPIO 控制器结构框图

### 8.6.2 模块特性

- 32 个管脚可独立设置为输入或输出
- 每个管脚均可以设置为 bypass 模式



- 
- 每个管脚输入状态下可作为中断源
  - 输入中断源可以设置为电平触发或边沿触发
  - 每个端口可通过 SCU 配置为上拉或下拉
  - 输出状态下每个 bit 都可单独设置 0 或 1
  - 所有管脚上电复位后默认为输入

TIH Confidential

## 9 安全特性

### 9.1 电压检测

#### 9.1.1 模块概述

电压检测模块 VDT 用于检测当前 IO 电压是否正常，当 IO 电压低于配置电压时，电压检测模块将触发 CPU 中断，可有效防止各种电压攻击手段。

电压检测模块结构如下图所示：

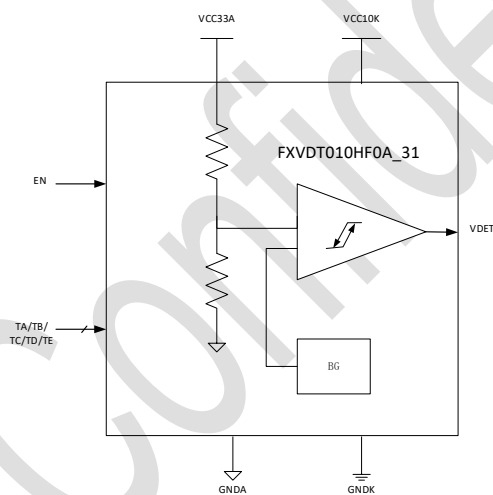


图 9.1 电压检测模块框图

#### 9.1.2 模块特性

- 节点工作温度范围-40~125°C
- 支持低功耗模式
- 支持检测电压阈值微调

## 9.2 温度检测

### 9.2.1 模块概述

温度检测 TDC 是一个高速温度转数字信号模块，可以帮助 CPU 进行实时温度监控及报警，当芯片处于极端环境或者瞬时温差较大的情况下，CPU 可以调整各模块配置让芯片进入更加安全的工作模式，以保持芯片工作的稳定性，同时也可以防止外部温度环境的攻击。

TDC 模块结构如下：

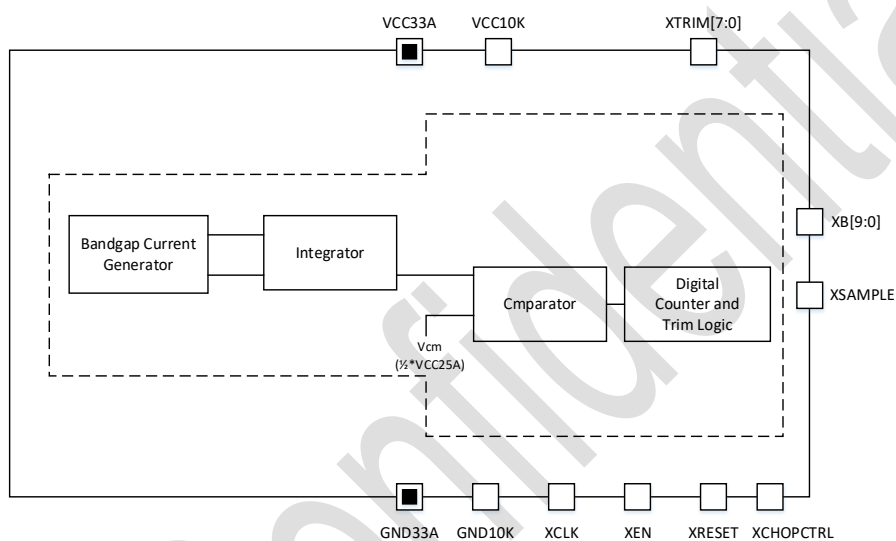


图 9.2 温度检测模块结构框图

### 9.2.2 模块特性

- 工作温度范围：-40~125°C
- 转换精度 10bit
- 转换速率 1.0KSPS
- 支持低功耗模式



### 9.2.3 模块时序

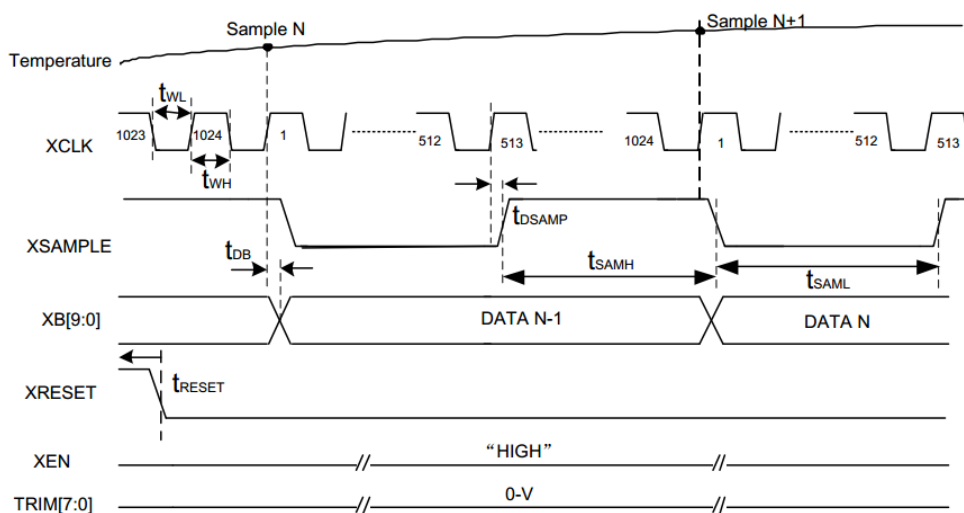


图 9.3 TDC 工作时序图

## 9.3 物理探测防护

### 9.3.1 金属屏蔽层

芯片采用 Power mesh 方法增加了金属屏蔽层，可有效防止芯片外部的电磁攻击。

### 9.3.2 后端设计防护

采用 Chip Level 层 Flatten 的方法，将接口电路、功能电路、密码算法电路和随机电路等完全进行混合布线，可有效防止后端电路反向分析等外部攻击。

## 9.4 芯片 ID

### 9.4.1 模块概述

芯片内置 OTP (One Time Programmable) 电路，提供一次性编程机会，可作为芯片全球唯一识别号。

### 9.4.2 模块特性

- 有效数据位宽 64bits



- 
- 可支持出厂烧写和用户烧写 2 种模式
  - 用户可自定义烧写内容
  - 支持低功耗模式

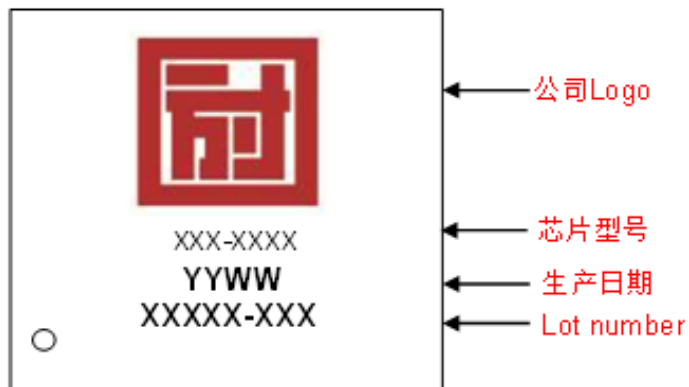
TIH Confidential

## 订购信息

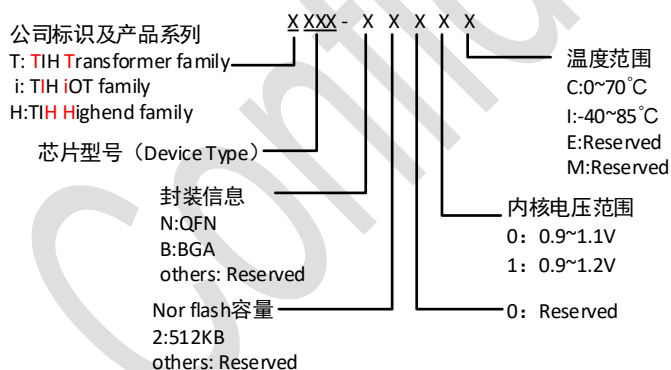
芯片名称	flash 容量	封装信息	温度范围	Package Qty
T660-B200C	512KB	BGA137	0~70°C	2400
T660-B201C	512KB	BGA137	0~70°C	2400

\*注: Package Qty 表示单箱芯片数量。单箱中含 10 个托盘, 每个托盘数量为 24\*10PCS, 托盘尺寸: 32.2\*13.6cm。

芯片外部丝印



芯片型号命名规则



举例

