

# XC6300

# 宽带高速电力线载波通信芯片

## 芯片特性

### ◆ 内核

- Cortex-M0 内核

### ◆ 存储器

- 504KB FLASH 存储空间（252KB 程序运行空间，252KB IAP 在线升级程序存储空间）
- 64KB SRAM

### ◆ 数字外设

- 5 个 UART 接口（其中 UART0，UART1 支持 IR38K 红外调制）
- 2 个硬件 I2C 接口
- 3 个硬件 SPI 主机接口和 1 个硬件 SPI 从机接口
- 4 个定时器（支持定时/PWM 输出/输入捕获/输出比较）
- GPIO 接口可配置上拉/下拉/推挽/开漏
- 支持 DMA
- 独立看门狗定时器
- 支持端口重映射
- 内置载波通信单元
- 支持 CRC32/CRC24 校验
- 支持 AES128/3DES/DES 加密

### ◆ 宽带载波通信单元

- 每帧数据长度为 136Byte 到 2080Byte
- 物理层峰值通信速率高达 23Mbps
- 调制方式为 OFDM
- 载波信号幅度可调，最大幅度为 1V（峰峰值）

### ◆ 时钟源

- 外部晶体输入频率 25MHz
- 内置 PLL 电路，最高输出频率 150MHz
- 主频为 PLL 输出 2 分频，最高 75MHz

### ◆ 供电

- 工作电压 2.7V~3.6V

### ◆ QFN64 封装

### ◆ 工作温度范围-45℃~85℃

## ◆ 编程和调试

- 支持在系统编程（ISP）
- 支持在应用编程（IAP）
- 支持 SWD 在线调试编程

# 目录

XC6300 .....	1
目录 .....	3
1. 概述 .....	8
1.1 简介 .....	8
1.2 结构框图.....	9
1.3 封装 .....	10
1.4 引脚定义.....	11
2. 存储器 .....	14
2.1 存储器组织.....	14
2.2 存储器映射表.....	14
2.3 存储器映射图.....	16
2.4 闪存控制接口.....	17
2.4.1 功能概述.....	17
2.4.2 闪存操作流程.....	18
2.4.3 闪存寄存器描述.....	19
2.5 程序存储区的加载.....	22
2.5.1 功能概述.....	22
2.6 程序错误校验.....	24
2.6.1 功能概述.....	24
2.6.2 功能框图.....	25
3. 时钟单元.....	26
3.1 时钟分类.....	26
3.2 时钟框图.....	27
3.3 时钟说明.....	28
4. 电源及电压检测.....	29
4.1 简介 .....	29
4.2 电源及电压检测寄存器.....	29
4.2.1 电源中断控制寄存器(SYS_PIE).....	29
4.2.1 电源中断状态寄存器(SYS_PIS).....	29
4.2.3 过压检测阈值设置寄存器(SYS_OCVS).....	30
5. 低功耗 .....	30
5.1 简介 .....	30
5.1.1 睡眠模式.....	30
5.1.2 深度睡眠模式.....	30
5.2 特殊功能寄存器.....	31
5.2.1 锁定寄存器(LP_LOCK).....	31
5.2.2 锁定标志位寄存器(LP_LS).....	31
5.2.3 睡眠控制寄存器(LP_SC).....	31
6. 通用/复用功能 IO(GPIO/AFIO).....	32
6.1 GPIO 功能概述 .....	32
6.2 GPIO 寄存器 .....	32

6.2.1 端口输入数据寄存器(GPIOx_IDR).....	33
6.2.2 端口输出数据寄存器(GPIOx_ODR) .....	33
6.2.3 端口位置位寄存器(GPIOx_BSR) .....	33
6.2.4 端口位清除寄存器(GPIOx_BRR).....	34
6.2.5 端口位取反寄存器(GPIOx_BTR) .....	34
6.2.6 端口输出配置寄存器(GPIOx_POSR).....	35
6.2.7 端口上拉配置寄存器(GPIOx_PUR).....	35
6.2.8 端口下拉配置寄存器(GPIOx_PDR).....	36
6.2.9 端口开漏配置寄存器(GPIOx_PODR) .....	36
6.2.10 端口安全锁寄存器(GPIOx_LOCK).....	37
6.2.11 端口位锁定寄存器(GPIOx_BLR) .....	37
6.3 复用功能寄存器.....	38
6.3.1 端口复用功能配置寄存器(GPIOx_PAFR) .....	39
6.3.2 端口复用功能选择寄存器(GPIOx_PAFSR).....	39
6.4 GPIO 中断寄存器 .....	40
6.4.1 外部中断使能寄存器(GPIOx_PIER) .....	40
6.4.2 外部中断类型配置寄存器(GPIOx_PITR) .....	40
6.4.3 外部中断极性配置寄存器(GPIOx_PIPR) .....	41
6.4.4 外部中断边沿触发选择寄存器(GPIOx_PIESR).....	42
6.4.5 外部中断状态寄存器(GPIOx_PISR) .....	42
7. 中断模块.....	43
7.1 中断向量说明.....	43
7.2 中断寄存器.....	43
7.2.1 中断使能寄存器(NVIC_ISER) .....	43
7.2.2 中断禁止寄存器(NVIC_ICER) .....	44
7.2.3 中断标志寄存器(NVIC_ISR) .....	45
7.2.4 中断清标志寄存器(NVIC_ICSR) .....	45
7.2.5 中断优先级 0 寄存器(NVIC_IRPR0).....	46
7.2.6 中断优先级 1 寄存器(NVIC_IRPR1).....	46
7.2.7 中断优先级 2 寄存器(NVIC_IRPR2).....	47
7.2.8 中断优先级 3 寄存器(NVIC_IRPR3).....	47
7.2.9 中断优先级 4 寄存器(NVIC_IRPR4).....	48
7.2.10 中断优先级 5 寄存器(NVIC_IRPR5).....	48
7.2.11 中断优先级 6 寄存器(NVIC_IRPR6).....	49
7.2.12 中断优先级 7 寄存器(NVIC_IRPR7).....	49
8. 复位模块.....	50
8.1 复位说明.....	50
8.2 复位模块寄存器.....	50
8.2.1 复位状态寄存器(SYS_RSTS).....	51
9. 通用异步收发器 (UART) .....	51
9.1 主要特性.....	51
9.2 UART 功能概述.....	51
9.3 IR 功能概述 .....	52
9.4 UART 寄存器.....	52

9.4.1 数据寄存器(UARTx_DR) .....	53
9.4.2 状态寄存器(UARTx_SR).....	53
9.4.3 控制寄存器 1(UARTx_CR1).....	55
9.4.4 中断状态寄存器(UARTx_ISR).....	56
9.4.5 波特率配置寄存器(UARTx_BRR) .....	57
9.4.6 控制寄存器 2(UARTx_CR2).....	58
9.5 IR 寄存器 .....	58
9.5.1 IRO 控制寄存器(IRO_CR).....	59
9.5.2 IRO 调制频率设置寄存器(IRO_BRR).....	59
9.5.3 IR1 控制寄存器(IR1_CR).....	60
9.5.4 IR1 调制频率设置寄存器(IR1_BRR).....	60
10. CRC 计算单元(CRC32/24).....	61
10.1 主要特性.....	61
10.2 功能描述.....	61
10.3 CRC 寄存器 .....	62
10.3.1 数据寄存器(CRC_DR) .....	62
10.3.2 控制寄存器(CRC_CR).....	62
10.3.3 CRC 初始值预置寄存器(CRC_INIT) .....	63
11. AES/3DES.....	64
11.1 主要特性.....	64
11.2 功能描述.....	64
11.3 看门狗寄存器.....	65
11.3.1 数据输入寄存器(AES_PT) .....	65
11.3.2 AES 数据输出寄存器(AES_CT) .....	66
11.3.3 AES 密钥寄存器(AES_KEY) .....	68
11.3.4 AES 控制/状态寄存器(AES_CSR).....	69
12. 独立看门狗(WDG).....	70
12.1 主要特性.....	70
12.2 功能描述.....	70
12.3 看门狗寄存器.....	71
12.3.1 计数值加载寄存器(WDG_LDR) .....	71
12.3.2 计数值寄存器(WDG_CVR) .....	71
12.3.3 控制寄存器(WDG_CR) .....	72
12.3.4 中断清除寄存器(WDG_ICR) .....	72
12.3.5 原中断状态寄存器(WDG_RISR).....	73
12.3.6 中断状态寄存器(WDG_ISR).....	73
12.3.7 安全锁寄存器(WDG_LOCKR) .....	73
13. 通用定时器(TIMx).....	74
13.1 简介 .....	74
13.2 主要特性.....	74
13.3 功能描述.....	74
13.3.1 时基单元.....	74
13.3.2 定时功能.....	75
13.3.3 PWM 功能.....	75

13.4 通用定时器寄存器.....	76
13.4.1 控制寄存器(TIMx_CR).....	76
13.4.2 计数器当前值寄存器(TIMx_CNT).....	76
13.4.3 计数器值设定寄存器(TIMx_CSR).....	77
13.4.4 中断状态寄存器(TIMx_ISR).....	77
13.4.5 中断状态寄存器(TIMx_RST).....	78
13.4.6 预分频值设置寄存器(TIMx_PSV).....	78
13.4.7 预分频值寄存器(TIMx_PV).....	78
13.4.8 匹配控制寄存器(TIMx_MCR).....	79
13.4.9 匹配值设置寄存器(TIMx_MVS).....	79
13.4.10 捕获控制寄存器(TIMx_CAC).....	80
13.4.11 捕获值寄存器(TIMx_CV).....	80
13.4.12 外部匹配控制寄存器(TIMx_EMCC).....	81
13.4.13 计数器/定时器控制寄存器(TIMx_CCR).....	81
13.4.14 PWM 输出控制寄存器(TIMx_PCR).....	82
14. 串行外设接口 (SPI).....	82
14.1 概述.....	82
14.2 详细功能说明.....	82
14.3 接口传输格式.....	83
14.4 SPI 寄存器.....	83
14.4.1 SPI 数据寄存器(SPIx_DR).....	83
14.4.2 SPI 状态寄存器(SPIx_SR).....	84
14.4.3 SPI 控制寄存器(SPIx_CR).....	84
14.4.4 SPI 控制寄存器(SPIx_ISR).....	85
14.4.5 SPI 波特率寄存器(SPIx_BR).....	85
14.4.6 SPI 线路控制寄存器(SPIx_LCR).....	86
15. I <sup>2</sup> C 模块.....	87
15.1 概述.....	87
15.2 功能描述.....	87
15.3 I <sup>2</sup> C 寄存器.....	88
15.3.1 I <sup>2</sup> C 数据寄存器(I <sup>2</sup> Cx_DR).....	88
15.3.2 I <sup>2</sup> C 状态寄存器(I <sup>2</sup> Cx_SR).....	88
15.3.3 I <sup>2</sup> C 控制寄存器(I <sup>2</sup> Cx_CR).....	89
15.3.4 I <sup>2</sup> C 波特率寄存器(I <sup>2</sup> Cx_BD).....	90
15.3.5 I <sup>2</sup> C 超时设置寄存器(I <sup>2</sup> Cx_TR).....	90
15.3.6 I <sup>2</sup> C 停止控制寄存器(I <sup>2</sup> Cx_SCR).....	91
15.3.7 I <sup>2</sup> C 中断状态寄存器(I <sup>2</sup> Cx_ISR).....	91
16. DMA.....	92
16.1 概述.....	92
16.2 功能描述.....	93
16.3 DMA 寄存器.....	94
16.3.1 DMA 写保护寄存器(DMA_LOCK).....	94
16.3.2 DMA 通道软件请求寄存器(DMA_SRR).....	94
16.3.3 DMA 通道使能寄存器(DMA_ER).....	95

16.3.4 DMA 总线错误寄存器(DMA_BER).....	95
16.3.5 DMA 中断状态寄存器(DMA_IS).....	96
16.3.6 DMA 错误中断使能寄存器(DMA_EIE).....	96
16.3.7 DMA 完成中断使能寄存器(DMA_CIE).....	96
16.3.8 DMA 通道模式选择寄存器(DMA_CMSR).....	97
16.3.9 DMA 源数据起始地址寄存器(DMA_SDSA).....	98
16.3.10 DMA 目的数据起始地址寄存器(DMA_DDSA).....	98
16.3.11 DMA 数据长度配置寄存器(DMA_LCR).....	98
17. SSPI 从机模块.....	100
17.1 概述.....	100
17.2 功能说明.....	100
17.3 SSPI 寄存器.....	100
17.3.1 SSPI 数据寄存器(SSPIx_DR).....	100
17.3.2 SSPI 控制寄存器(SSPIx_CR).....	101
17.3.3 SSPI 状态寄存器(SSPIx_SR).....	102
17.3.4 SSPI 线路控制寄存器(SSPIx_LCR).....	102
18. 宽带载波通信模块(BPL).....	103
18.1 主要特性.....	103
18.2 功能概述.....	103
19. 系统控制模块.....	104
19.1 简介.....	104
19.2 功能概述.....	104
19.3 系统控制寄存器.....	104
19.3.1 SYS 复位状态寄存器(SYS_RSTS).....	104
19.3.2 SYS 锁定寄存器(SYS_ULOCK).....	105
19.3.3 SYS 程序存储器校验控制寄存器(SYS_PMCC).....	105
19.3.4 SYS 校验错误地址寄存器(SYS_MFCA).....	105
19.3.5 SYS FLASH 程序加载控制寄存器(SYS_FLC).....	106
19.3.6 SYS FLASH 程序加载错误寄存器(SYS_FLE).....	106
19.3.7 SYS FLASH 程序加载错误地址寄存器(SYS_FLEA).....	106
19.3.8 SYS CPU 时钟分频寄存器(SYS_CCD).....	107
19.3.9 SYS 电源中断控制寄存器(SYS_PIE).....	107
19.3.10 SYS 电源中断状态寄存器(SYS_PIS).....	108
19.3.11 SYS 过压检测阈值设置寄存器(SYS_OCVS).....	108
19.3.12 SYS 外部晶振微调控制寄存器(SYS_XTC).....	109
19.3.13 SYS 外部晶振输出分频寄存器(SYS_XTD).....	109
20. 参考电路.....	110
20.1 最小系统参考.....	110
20.2 载波电路参考.....	111

# 1. 概述

## 1.1 简介

XC6300 的 MCU 采用 ARM 的 32 位 Cortex-M0 处理器内核，提供了一种低成本的平台旨在满足少引脚数和低功耗单片机的需求，同时提供卓越的代码效率，出色的计算性能和先进的系统响应中断。支持大容量的程序存储空间和数据存储空间，提供了 SWD 在线调试接口便于程序的开发和调试。

载波通信单元发送和接收端采用全新的数字接收信号处理算法，在达到大幅度提高抗干扰性能和稳定性的同时，通信速率也得到大幅度的提升，将很容易组建自适应性强的电力线通信网络，应用于自动抄表及工业控制等应用中。

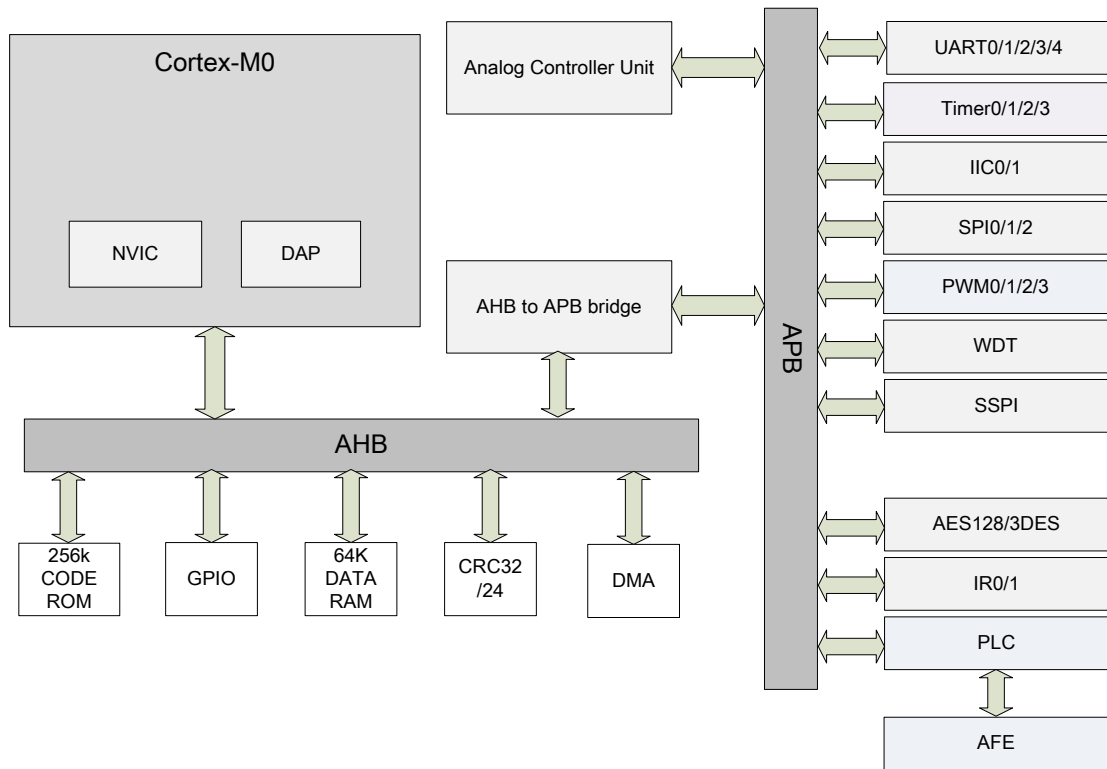
主要功能及指标参数：

- 工作电压范围：2.7V~3.6V
- 工作温度范围：-45° C~85° C
- ARM Cortex-M0 内核，系统时钟频率 37.5Mhz/75MHz 可配置
- 内置 PLL，输入频率为 25M，最高输出频率为 150MHz
- 252K Bytes 程序运行空间，504K Bytes 程序存储空间（分为 BANK0 和 BANK1，各 252K Bbytes），系统每次加载其中一个 BANK 空间的数据到程序运行空间。
- 64K Bytes SRAM
- 提供 5 个 UART 接口（其中 2 个通道支持 IR38K 红外调制）
- 支持 2 个 IIC 接口
- 支持 3 个 SPI 主机接口和 1 个 SPI 从机接口
- 4 个定时器（支持 PWM/输入捕获/输出匹配）
- 支持端口重映射功能
- 提供 32/24 位 CRC 校验功能
- 支持 AES128/3DES/DES 加密算法
- 支持 DMA 功能
- 独立看门狗定时器
- 支持在系统编程（ISP）
- 支持在应用编程（IAP）



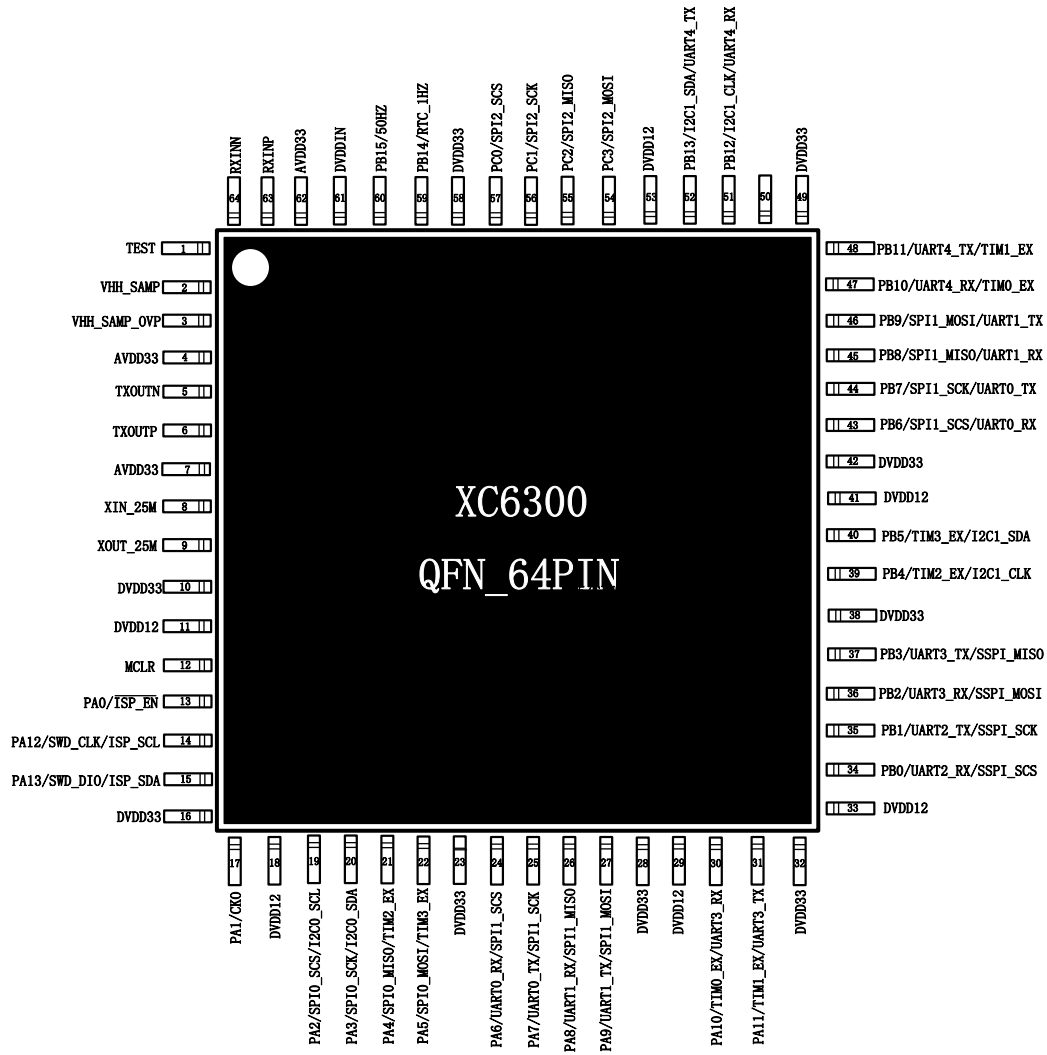
- 支持串行线调试（SWD）功能
- 提供 GPIO 接口，上拉/下拉/漏极开路/推挽功能可配置
- 内置载波通信单元

## 1.2 结构框图



系统结构框图

### 1.3 封装



引脚排列图

## 1.4 引脚定义

管脚名称	QFN64	类型	描述
GND	0	PWR	模拟地和数字地
TEST	1	0	芯片测试引脚, 不对用户开放。使用时悬空
VHH_SAMP	2	I	12V 电源采样引脚。用户通过合适的电阻分压比例对 12V 电源进行采样, 当 12V 电源低于设置阈值时, 发出欠压报警信号。(内部比较器阈值为 0.6V)
VHH_SAMP_OVP	3	I	12V 电源采样引脚。用户通过合适的电阻分压比例对 12V 电源进行采样, 当 12V 电源高于设置阈值时, 发出过压报警信号。(内部比较器阈值可编程, 0.3V~1.0V, 0.1V 步长)。此功能引脚可用于电力线倒灌监测
AVDD33	4	PWR	芯片的 3.3V 模拟电源, 该引脚建议使用 10uF 电容并联 100nF 电容进行去耦
TXOUTN	5	0	载波通信发射电路差分输出引脚
TXOUTP	6	0	载波通信发射电路差分输出引脚
AVDD33	7	PWR	芯片的 3.3V 模拟电源, 该引脚建议使用 10uF 电容并联 100nF 电容进行去耦
XIN_25M	8	I	外部晶振输入端, 或是外部系统时钟输入。晶体典型频率为 25MHZ。同时需接对应的匹配电容
XOUT_25M	9	0	外部晶振输出端, 同时需接对应的匹配电容。外部系统时钟输入时此引脚悬空。
DVDD33	10	PWR	芯片的 3.3V 数字电源, 该引脚建议使用 10uF 电容并联 100nF 电容进行去耦
PC13	11	I/O	通用 IO 口
MCLR	12	I	外部手动复位端, 低有效
PA0	13	I/O	通用 IO 口
		I	ISP 选通使能, 低有效, 当系统上电或手动复位完成后, 硬件会在 4 个时钟后检查 PA0 的状态为低, 将进入到 ISP 编程模式
PA12	14	I/O	通用 IO 口,
		I	SWD 串行调试时钟输入端(SWCLK)
		I	ISP 串行编程时钟输入端(ISPSCK)
PA13	15	I/O	通用 IO 口
		I/O	SWD 串行调试数据输入输出端(SWDIO)
		I/O	ISP 串行编程数据输入输出端(ISPSDA)
DVDD33	16	PWR	芯片的 3.3V 数字电源, 该引脚建议使用 10uF 电容并联 100nF 电容进行去耦
PA1	17	I/O	通用 IO 口
		0	提供外部 25MHz 分频的时钟或者内部 PLL 分频的时钟输出端。
DVDD33	18	PWR	芯片的 3.3V 数字电源, 该引脚建议使用 10uF 电容并联 100nF 电容进行去耦
PA2	19	I/O	通用 IO 口
		0	SPI0 主机片选输出端(SCS)
		0	IIC0 主机时钟输出端(SCL)
PA3	20	I/O	通用 IO 口
		0	SPI0 主机时钟信号输出端(SCK)
		I/O	IIC0 主机数据输入输出端(SDA)

PA4	21	I/O	通用 I/O 口
		I	SPI0 主机数据信号输入端 (SDI)
		I/O	TIMER2 外部输入输出端
PA5	22	I/O	通用 I/O 口
		0	SPI0 主机数据信号输出端 (SDO)
		I/O	TIMER3 外部输入输出端
DVDD33	23	PWR	芯片的 3.3V 数字电源, 该引脚建议使用 10uF 电容并联 100nF 电容进行去耦
PA6	24	I/O	通用 I/O 口
		I	串口 0 数据接收端 (RXD)
		0	SPI1 主机片选输出端 (SCS)
PA7	25	I/O	通用 I/O 口
		0	串口 0 数据发送端 (TXD)
		0	SPI1 主机时钟信号输出端 (SCK)
PA8	26	I/O	通用 I/O 口
		I	串口 1 数据接收端 (RXD)
		I	SPI1 主机数据信号输入端 (SDI)
PA9	27	I/O	通用 I/O 口
		0	串口 1 数据发送端 (TXD)
		0	SPI1 主机数据信号输出端 (SDO)
DVDD33	28	PWR	芯片的 3.3V 数字电源, 该引脚建议使用 10uF 电容并联 100nF 电容进行去耦
PC11	29	I/O	通用 I/O 口
PA10	30	I/O	通用 I/O 口
		I/O	TIMER0 外部输入输出端
		I	串口 3 数据接收端 (RXD)
PA11	31	I/O	通用 I/O 口
		I/O	TIMER1 外部输入输出端
		0	串口 3 数据发送端 (TXD)
DVDD33	32	PWR	芯片的 3.3V 数字电源, 该引脚建议使用 10uF 电容并联 100nF 电容进行去耦
PC9	33	I/O	通用 I/O 口
PB0	34	I/O	通用 I/O 口
		I	串口 2 数据接收端 (RXD)
		I	SPI 从机片选输入端 (SCS)
PB1	35	I/O	通用 I/O 口
		0	串口 2 数据发送端 (TXD)
		I	SPI 从机时钟输入端 (SCK)
PB2	36	I/O	通用 I/O 口
		I	串口 3 数据接收端 (RXD)
		I	SPI 从机数据输入端 (SDI)
PB3	37	I/O	通用 I/O 口
		0	串口 3 数据发送端 (TXD)
		0	SPI 从机数据输出端 (SDO)
DVDD33	38	PWR	芯片的 3.3V 数字电源, 该引脚建议使用 10uF 电容并联 100nF 电容进行去耦
PB4	39	I/O	通用 I/O 口

		0	TIMER2 外部输入输出端
		0	IIC1 主机时钟输出端 (SCL)
PB5	40	I/O	通用 IO 口
		0	TIMER3 外部输入输出端
		I/O	IIC1 主机数据输入输出端 (SDA)
PC8	41	I/O	通用 IO 口
DVDD33	42	PWR	芯片的 3.3V 数字电源, 该引脚建议使用 10uF 电容并联 100nF 电容进行去耦
PB6	43	I/O	通用 IO 口
		0	SPI1 主机片选输出端 (SCS)
		I	串口 0 数据接收端 (RXD)
PB7	44	I/O	通用 IO 口
		0	SPI1 主机时钟信号输出端 (SCK)
		0	串口 0 数据发送端 (TXD)
PB8	45	I/O	通用 IO 口
		I	SPI1 主机数据信号输入端 (SDI)
		I	串口 1 数据接收端 (RXD)
PB9	46	I/O	通用 IO 口
		0	SPI1 主机数据信号输出端 (SDO)
		0	串口 1 数据发送端 (TXD)
PB10	47	I/O	通用 IO 口
		I	串口 4 数据接收端 (RXD)
		0	TIMER0 外部输入输出端
PB11	48	I/O	通用 IO 口
		0	串口 4 数据发送端 (TXD)
		I/O	TIMER1 外部输入输出端
DVDD33	49	PWR	芯片的 3.3V 数字电源, 该引脚建议使用 10uF 电容并联 100nF 电容进行去耦
Reserved	50	-	保留
PB12	51	I/O	通用 IO 口
		0	IIC1 主机时钟输出端 (SCL)
		I	串口 4 数据发送端 (TXD)
PB13	52	I/O	通用 IO 口
		I/O	IIC1 主机数据输入输出端 (SDA)
		0	串口 4 数据发送端 (TXD)
PC4	53	I/O	通用 IO 口
PC3	54	I/O	通用 IO 口
		0	SPI2 主机数据信号输出端 (SDO)
PC2	55	I/O	通用 IO 口
		I	SPI2 主机数据信号输入端 (SDI)
PC1	56	I/O	通用 IO 口
		0	SPI2 主机时钟信号输出端 (SCK)
PC0	57	I/O	通用 IO 口
		0	SPI2 主机片选输出端 (SCS)
DVDD33	58	PWR	芯片的 3.3V 数字电源, 该引脚建议使用 10uF 电容并联 100nF 电容进行去耦

PB14	59	I/O	通用 I/O 口
PB15	60	I/O	通用 I/O 口
		I	50HZ 工频信号输入
DVDDIN	61	PWR	1.5V 数字 core 电源, 该引脚建议使用 10uF 电容并联 100nF 电容进行去耦
AVDD33	62	PWR	芯片的 3.3V 模拟电源, 该引脚建议使用 10uF 电容并联 100nF 电容进行去耦
RXINP	63	I	载波通信接收电路差分输入引脚
RXINN	64	I	载波通信接收电路差分输入引脚

## 2. 存储器

### 2.1 存储器组织

程序存储器、数据存储器、寄存器和输入输出端口被组织在同一线性地址空间内。数据字节以小端格式存放在存储器中, 一个字里的最低地址字节被认为是该字的最低有效字节, 而最高地址字节是最高有效字节。

具体请参考存储器映射表及映射图。

### 2.2 存储器映射表

表 1 寄存器组地址映射

起始地址	外设
0x5000 0000 - 0x5000 0018	FLASH 操作
0x4002 0000 - 0x4FFF FFFF	保留
0x4001 F800 - 0x4001 FFFF	系统控制器 2
0x4001 F000 - 0x4001 F7FF	系统控制器 1
0x4001 3000 - 0x4001 3008	CRC
0x4001 2000 - 0x4001 2044	GPIO 端口 C
0x4001 1000 - 0x4001 1044	GPIO 端口 B
0x4001 0000 - 0x4001 0044	GPIO 端口 A
0x4000 A800 - 0x4000 B7FF	AES 加密
0x4000 A000 - 0x4000 A7FF	SSPI
0x4000 9800 - 0x4000 9FFF	ANA 模拟控制
0x4000 9000 - 0x4000 97FF	载波控制
0x4000 8800 - 0x4000 8FFF	DMA
0x4000 8000 - 0x4000 87FF	IR 红外控制
0x4000 7000 - 0x4000 77FF	I2C1
0x4000 6800 - 0x4000 6FFF	I2C0

0x4000 6000 - 0x4000 67FF	SPI2
0x4000 5800 - 0x4000 5FFF	SPI1
0x4000 5000 - 0x4000 57FF	SPI0
0x4000 4800 - 0x4000 4FFF	看门狗定时
0x4000 4000 - 0x4000 47FF	UART4
0x4000 3800 - 0x4000 3FFF	UART3
0x4000 3000 - 0x4000 37FF	UART2
0x4000 2800 - 0x4000 2FFFF	UART1
0x4000 2000 - 0x4000 27FF	UART0
0x4000 1800 - 0x4000 1FFF	TIMER3
0x4000 1000 - 0x4000 17FF	TIMER2
0x4000 0800 - 0x4000 0FFF	TIMER1
0x4000 0000 - 0x4000 07FF	TIMER0

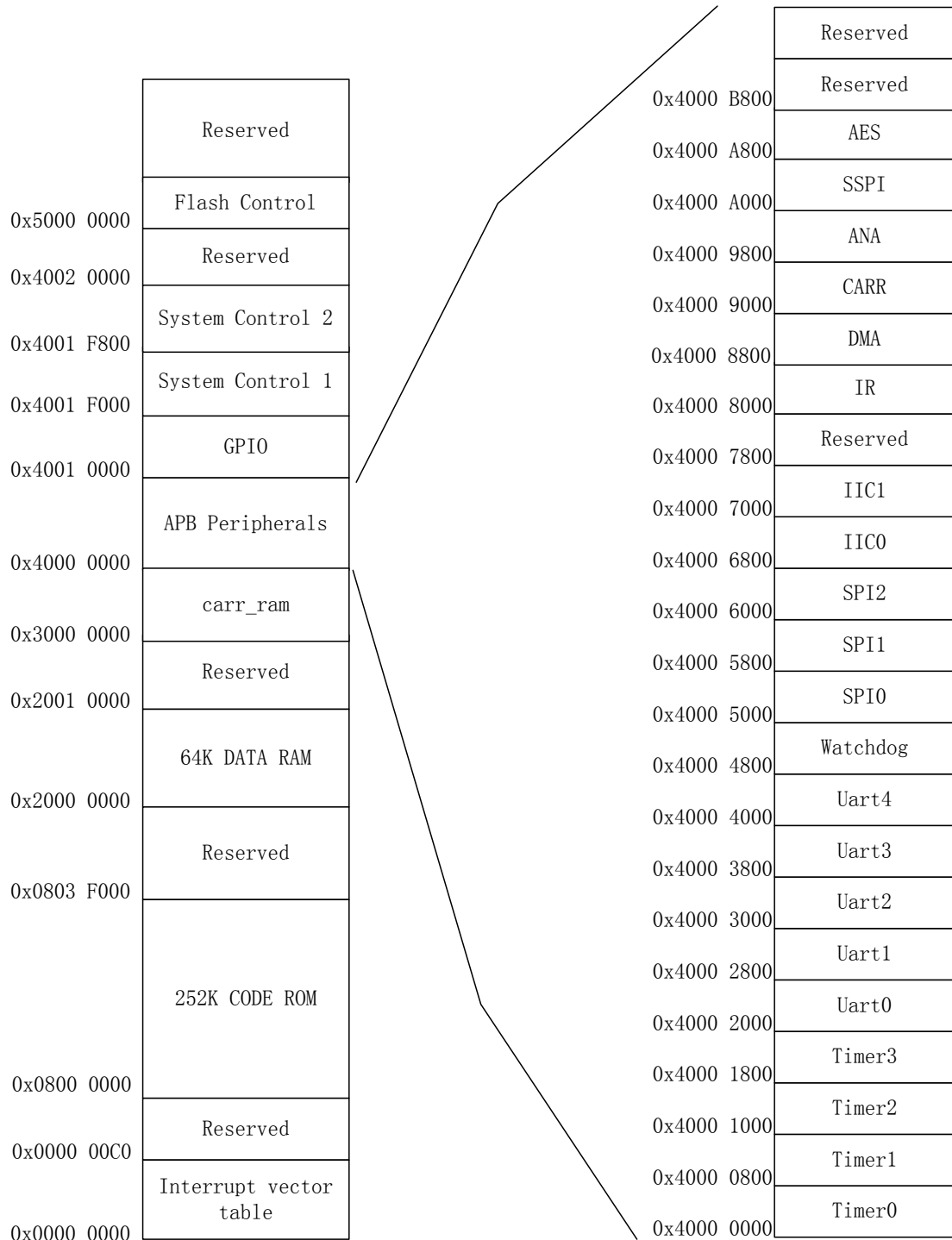
表 2 SRAM 地址映射

起始地址	功能
0x3000 0000 - 0x3000 07FF	载波载荷数据存储
0x3000 0800 - 0x3000 09FF	子载波信噪比参数
0x3000 0A00 - 0x3000 0BFF	子载波信号衰减参数
0x3000 0C00 - 0x3000 0DFF	载波发送预加重值
0x3000 0E00 - 0x3000 0FFF	载波接收预加重值
0x2001 0000 - 0x2FFF FFFF	保留
0x2000 0000 - 0x2000 FFFF	64KB SRAM

表 3 闪存地址映射

模块	起始地址	大小(字节)	Sector(4K)	名称
主存储块	0x0800 0000 - 0x0800 00FF	256	Sector 0	Page 0
	0x0800 0100 - 0x0800 01FF	256		Page 1
	...	...		...
	0x0800 0F00 - 0x0800 0FFF	256		Page 15
	0x0800 1000 - 0x0800 1FFF	4K	Sector 1	Page 16 ~ Page 31
	...	...	...	...
	0x0803 D000 - 0x0803 DFFF	4K	Sector 61	Page 976~Page 991
	0x0803 E000 - 0x0803 EFFF	4K	Sector 62	Page 992~Page 1007
闪存存储器接口寄存器	0x5000 0000 - 0x5000 0003	4	FMC_STATUS	
	0x5000 0004 - 0x5000 0007	4	FMC_START_ADDR	
	0x5000 0008 - 0x5000 000B	4	FMC_SRAM_ADDR	
	0x5000 000C - 0x5000 000F	4	FMC_DATA_LEN	
	0x5000 0010 - 0x5000 0013	4	FMC_COMMAND	
	0x5000 0014 - 0x5000 0017	4	FMC_FLASHLOCK	
	0x5000 0018 - 0x5000 001B	4	FLASH_INFOLOCK	

## 2.3 存储器映射图



存储器映射图



## 2.4 闪存控制接口

### 2.4.1 功能概述

通过配置相应闪存控制寄存器，用户可以对闪存进行擦除、读写和加密操作。配置闪存寄存器前，必须先进行解除锁定操作，然后等待直到寄存器 FMC\_FLASHLOCK 的 bit0 为 0，表示闪存解锁成功。

对闪存的操作，实际上是闪存与 SRAM 间的数据读写过程，将预操作闪存起始地址、SRAM 起始地址以及读写长度分别写入对应寄存器中，再写入相应操作指令即可实现对闪存的相应操作。

对闪存进行擦除和写入操作前需要先置闪存写使能，每次擦除和写入后，写使能位硬件自动清零，因此在下一次进行擦除和写入操作时要重新置使能。

#### 2.4.1.1 解锁和锁定

对闪存进行任何操作前需要向锁定寄存器 FMC\_FLASHLOCK 写入 0x8AA55679 解除锁定，解锁后闪存主区域可以进行操作，向锁定寄存器写入解锁命令以外的任意其它数据可以对闪存进行锁定。

#### 2.4.1.2 置写使能

解锁闪存后，向寄存器 FMC\_COMMAND 写入闪存置使能指令码 0x0000 0003，等待操作完成后，即可对闪存进行写操作。

#### 2.4.1.3 闪存指令码

向 FLASH\_COMMAND 寄存器写入相应操作指令可以对闪存进行相应操作，具体指令如下表所示。

操作指令	功能说明
0x0000 0001	从闪存读取数据到 SRAM
0x0000 0002	将 SRAM 中数据写入闪存
0x0000 0003	闪存置使能
0x0000 0004	闪存 SECTOR 擦除
0x0000 0005	闪存 32K BLOCK 擦除
0x0000 0006	闪存 64K BLOCK 擦除
0x0000 0007	闪存主区域全擦除
0x0000 0008	闪存加密

## 2.4.2 闪存操作流程

### 2.4.2.1 闪存读取数据

闪存读取数据流程:

- 闪存解锁
- 写入预读取闪存的起始地址
- 写入 SRAM 的起始地址
- 写入读取数据长度 (单位: 字)
- 写入读取指令码 0x0000 0001
- 等待读操作完成

### 2.4.2.2 闪存写入数据

闪存写入数据流程:

- 解除锁定
- 闪存置写使能
- 写入预写入闪存的起始地址
- 写入数据所在 SRAM 的起始地址
- 写入操作数据长度 (单位: 字)
- 写入操作指令码 0x0000 0002
- 等待操作完成

### 2.4.2.3 闪存擦除

闪存擦除流程:

- 解除锁定
- 闪存置写使能
- 写入预擦除闪存区域的页首地址
- 写入 SRAM 的起始地址
- 写入擦除指令码 (主区域全擦除/Sector 擦除/Block 擦除)
- 等待操作完成

### 2.4.2.4 闪存加密操作

闪存加密流程:

- 闪存主区域解锁
- 信息页解锁
- 闪存置写使能
- 写入加密页首地址, 固定为 0x0000 1000

- 写入 Sector 擦除指令码 0x0000 0004
- 等待擦除操作完成
- 闪存置写使能
- 写入加密指令码 0x0000 0008
- 等待加密操作完成

### 2.4.3 闪存寄存器描述

闪存寄存器基地址为 0x5000 0000.

#### 2.4.3.1 状态寄存器 (FLASH\_SR)

地址偏移: 0x00

复位值: 0x0000 0000

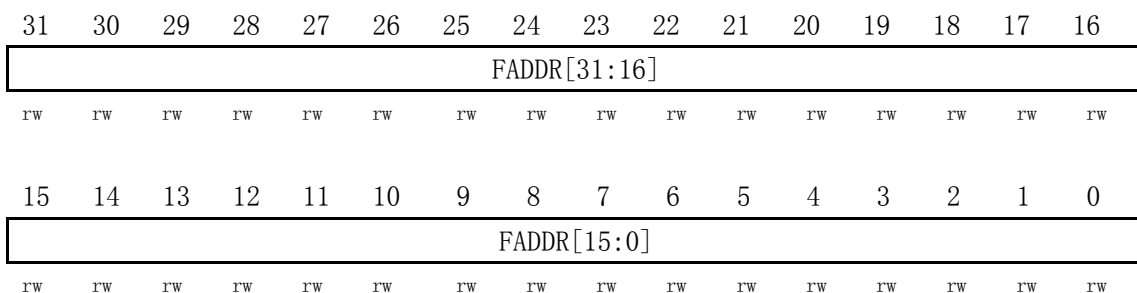


Bit31:Bit1	保留位, 硬件强制为 0.
Bit0	0C: 闪存操作完成标志位 0: 闪存操作完成 1: 闪存操作未完成 对闪存进行读写操作时, 该位由硬件自动置 1; 当操作完成后, 该位由硬件自动清零。

#### 2.4.3.2 闪存地址寄存器 (FLASH\_FADDR)

地址偏移: 0x04

复位值: 0x0000 0000

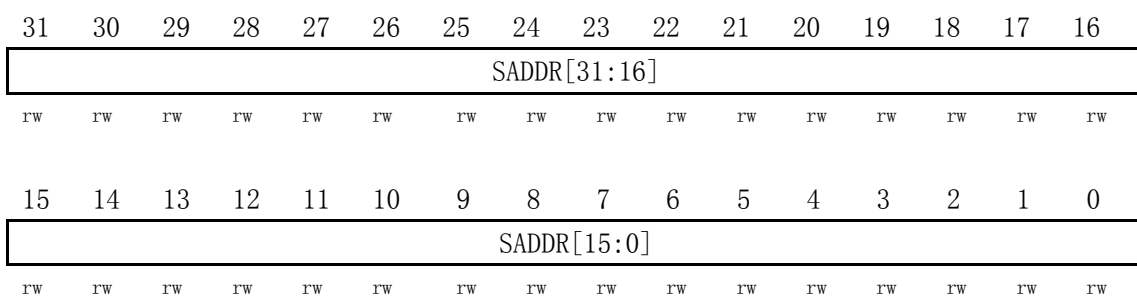


Bit31:Bit0	FADDR[31:0]: 闪存起始地址 读写闪存时, 该寄存器写入预读写闪存数据所在页的首地址
------------	--

### 2.4.3.3 SRAM 地址寄存器 (FLASH\_SADDR)

地址偏移: 0x08

复位值: 0x0000 0000

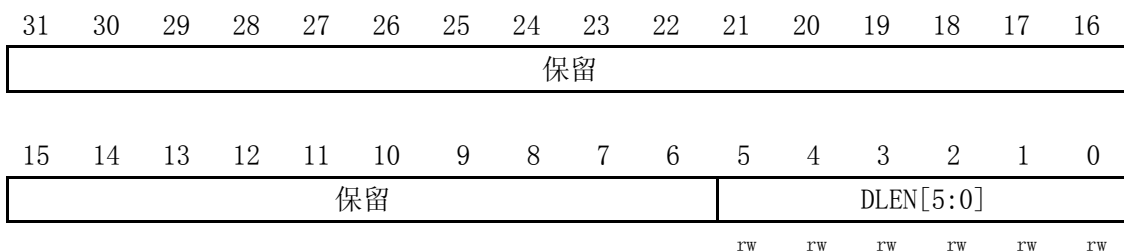


Bit31:Bit0	SADDR[31:0]: SRAM 起始地址 读写闪存数据时, 实质上是闪存与 SRAM 间的数据交互。该寄存器写入预读写的数据在 SRAM 中存储的首地址。
------------	--

### 2.4.3.4 数据长度寄存器 (FLASH\_DLEN)

地址偏移: 0x0C

复位值: 0x0000 0000



Bit31:Bit6	保留位, 硬件强制为 0.
Bit5: Bit0	DLEN[5:0]: 闪存读写的数据长度 读写闪存时, 该寄存器写入预读写闪存数据的长度, 单位为字。

### 2.4.3.5 操作指令寄存器 (FLASH\_CMD)

地址偏移: 0x10

复位值: 0x0000 0000

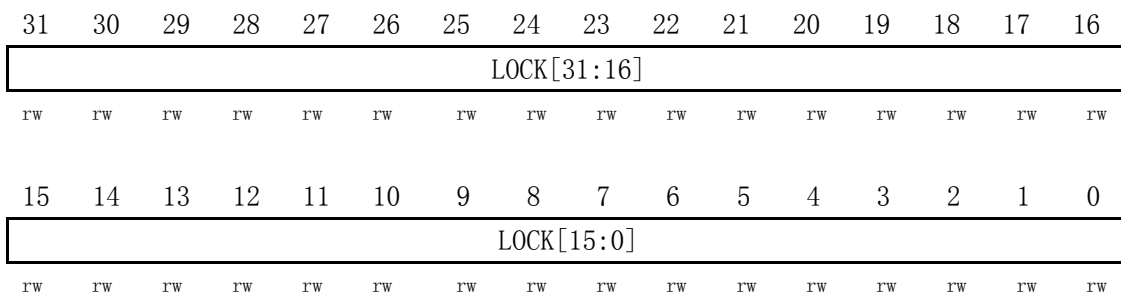


Bit31:Bit4	保留位, 硬件强制为 0.			
Bit3 :Bit0	CMD[3:0]: 闪存操作指令 读写闪存时, 该寄存器写入具体对应的操作指令码, 有效指令码范围为 0x01~0x08, 其他值无效。			
	CMD	功能说明	CMD	功能说明
	0x01	从闪存读取数据	0x05	闪存 32K Block 擦除
	0x02	向闪存写入数据	0x06	闪存 64K Block 擦除
	0x03	置闪存写使能	0x07	闪存主区域全擦除
	0x04	闪存 Sector 擦除	0x08	闪存加密

### 2.4.3.6 闪存锁定寄存器 (FLASH\_LOCK)

地址偏移: 0x14

复位值: 0x0000 0001

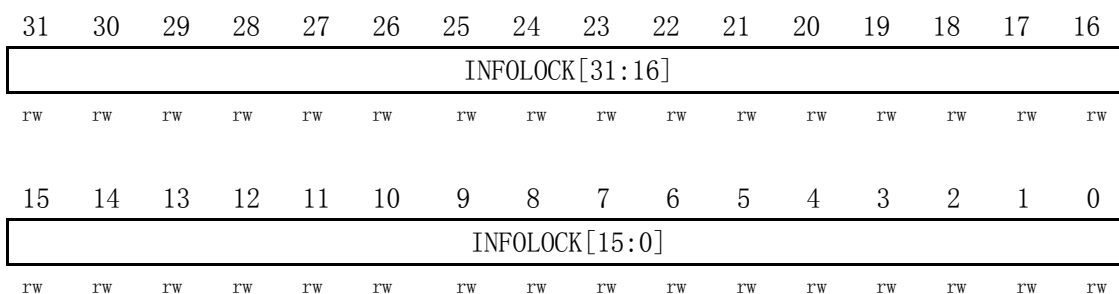


Bit31:Bit0	LOCK[31:0]: 闪存锁定 向该寄存器写入 0x8AA55679, 闪存解锁; 向该寄存器写入其他数据, 则闪存锁定。		
Bit0 (R)	仅在读取该寄存器时, Bit0 作为闪存锁定状态标志位。 0: 解锁状态 1: 锁定状态		

### 2.4.3.7 信息页锁定寄存器 (FLASH\_INFOLOCK)

地址偏移: 0x18

复位值: 0x0000 0001

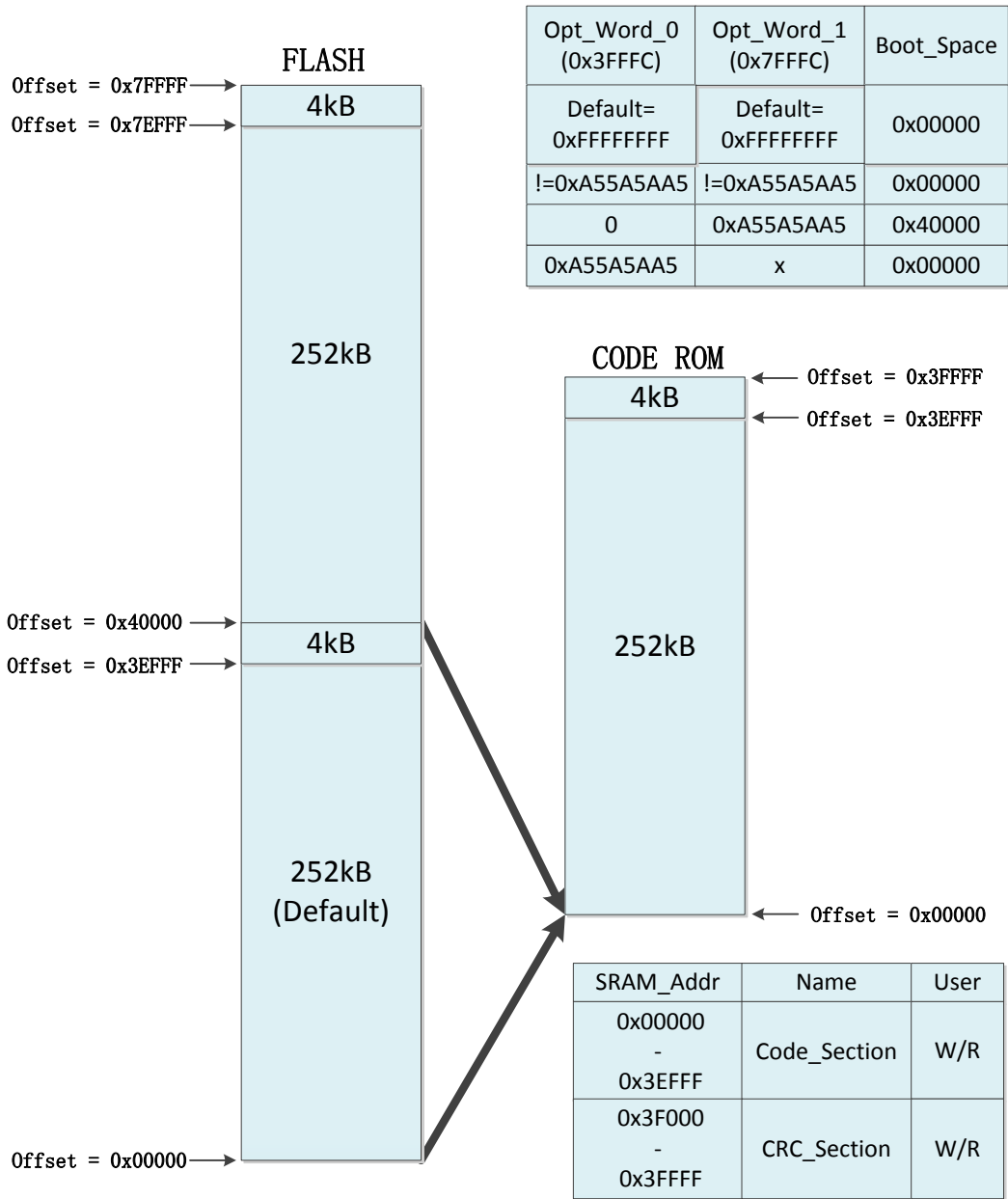


Bit31:Bit0	INFOLOCK[31:0]: 闪存锁定 向该寄存器写入 0x966A4268, 信息页解锁; 向该寄存器写入其他数据, 则信息页锁定。
Bit0 (R)	仅在读取该寄存器时, Bit0 作为信息页锁定状态标志位。 0: 解锁状态 1: 锁定状态

## 2.5 程序存储区的加载

### 2.5.1 功能概述

XC6300 采用外置 FLASH, 容量为 512KB, 分为 BANK0 和 BANK1 各 252KB, 系统上电或者外部手动复位或者看门狗复位 (FLASH\_LOAD ==1) 会将程序加载到内部的 CODE ROM 中, 同时对每个页面进行 CRC 校验, 校验结果存放在 CODE ROM 的最后 4KB 中。



外置 FLASH 同 CODE ROM 地址对应关系

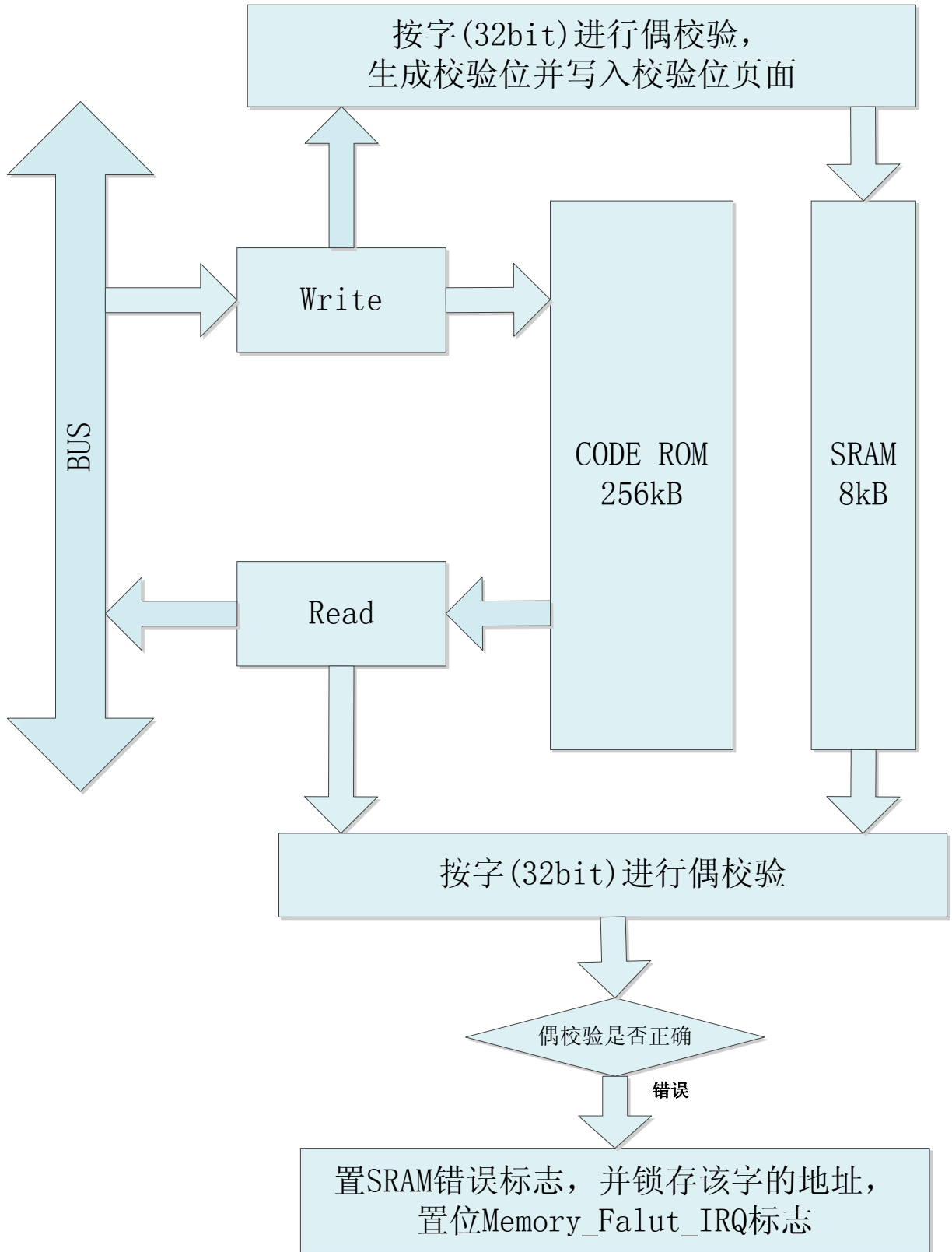
## 2.6 程序错误校验

### 2.6.1 功能概述

由于应用程序运作在 SRAM 中，SRAM 比较容易受到外界的干扰导致程序被修改，导致运行错误，ARM 内核自带了一些错误检测机制，但是主要针对不合法的指令和程序跑飞的情况，当指令是合法的，但是指令后面的操作数是不正确(被篡改)的，就无能为力了，基于这个原因，设计了偶校验检测电路，对 CODE ROM 每次进行读写操作时都进行偶校验，在写操作的情况下，会将偶校验值保存起来；在读操作的情况下，会将保存起来的偶校验值和计算值进行比较，判断是否一致，若不一致，将会给出错误中断和相应的错误地址。



### 2.6.2 功能框图



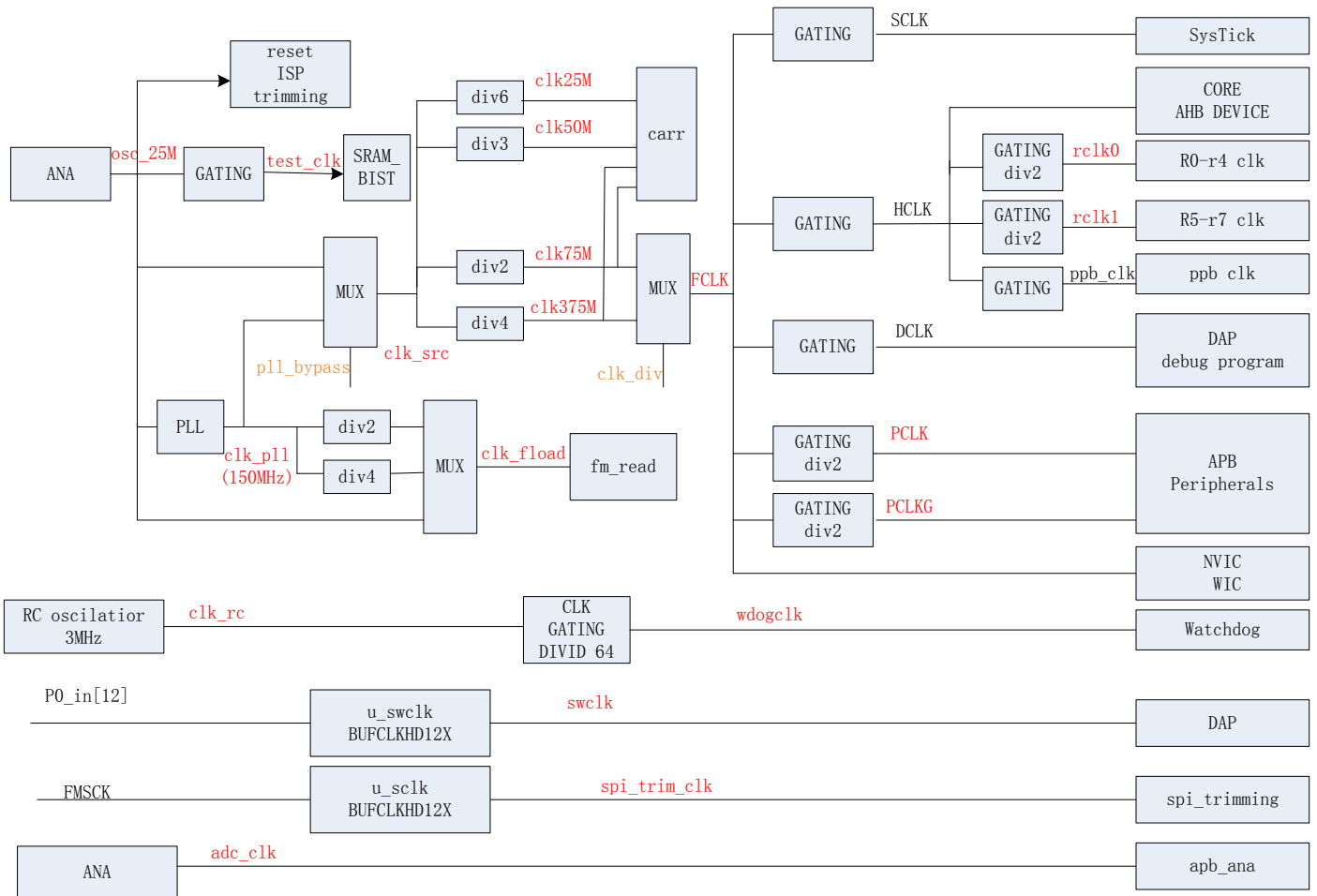
## 3. 时钟单元

### 3.1 时钟分类

时钟	频率	源头	说明
osc_25M	25MHz	外置晶振通过模拟模块的 OSC buffer 输入的时钟	用于 ISP 编程, flash_trim 以及 reset 模块, 在 PLL_BYPASS 为 1 时, 用于替代 clk_pll
clk_rc	3MHz	模拟模块 RC 振荡器生成	用于生成看门狗时钟
wdog_clk	46.8KHz	clk_rc 的 64 分频	用于看门狗定时
swclk		SWD 调试端口 SWCLK 输入	用于 DAP 调试模块中 SWD 数据的锁存
spi_trim_sclk		FLASH_ENA 为 0 时由 FMSCK 端口输入	用于 spi_trimming_regs
adc_clk	37.5MHz	模拟模块 ADC 的采样时钟	用于 apb_ana 模块中 CALIC_RXADC 功能
test_clk	25MHz	由 osc_25M 时钟经过 TESTMODE 门控后生成	用于 sram_bist 模块以及 SRAM 测试时所有 SRAM 的时钟输入
clk_pll	150MHz	模拟模块中 PLL 生成	生成 clk_src 和 clk_fload
clk_fload	25MHz/ 75MHz/ 37.5MHz	在 reset_module 中由 osc_25M, clk_pll/2, clk_pll/4 选择生成	用于 fm_read 模块, 即 flash 程序的加载
clk_src	25MHz/ 150MHz	在 cortex_m0_mcu_clkctrl 模块中由 osc_25M 和 clk_pll 选择生成	分频后用于系统各主要模块
clk75M	12.5MHz/ 75MHz	clk_src 二分频	用于载波计算 cpu_clk_div 为 1 时, 作为 cpu 工作时钟
clk375M	6.25MHz/ 37.5MHz	clk_src 四分频	用于 ADC 数据的锁存和 AGC 模块 cpu_clk_div 为 0 时, 作为 cpu 工作时钟
clk50M	8.3MHz/ 50MHz	clk_src 三分频	反相并经 dac_ena 门控后用于模拟模块 DAC; 门控后用于载波发送

			模块
clk25M	4.16MHz/ 25MHz	clk_src 六分频	用于 apb_carr 模块中载波基准时钟
FCLK	37.5MHz/ 75MHz	clk75M 和 clk375M 选择生成	作为 CPU 的自由工作时钟
HCLK	37.5MHz/ 75MHz	FCLK 门控时钟	AHB 外设时钟以及 CPU CORE 工作时钟
SCLK	37.5MHz/ 75MHz	FCLK 门控时钟	
DCLK	37.5MHz/ 75MHz	FCLK 门控时钟	DAP 调试模块时钟 debug 编程模块时钟
PCLK	18.75MHz/ 37.5MHz	FCLK 二分频门控时钟	APB 外设工作时钟
PCLKG	18.75MHz/ 37.5MHz	FCLK 二分频门控时钟	APB 外设写寄存器时钟
rclk0	18.75MHz/ 37.5MHz	HCLK 二分频门控时钟	寄存器 r0~r4 控制时钟
rclk1	18.75MHz/ 37.5MHz	HCLK 二分频门控时钟	寄存器 r5~r7 控制时钟
ppb_clk	37.5MHz/ 75MHz	HCLK 门控时钟	cpu core 中 PPB 模块控制时钟

### 3.2 时钟框图



### 3.3 时钟说明

osc\_25M, swclk, spi\_tr\_sclk, clk\_rc, adc\_clk, clk\_pll 时钟都是外部输入，互为异步时钟。

clk\_fload 的三种频率由 pll\_byass 和 CLK\_SEL 引脚控制，默认状态，PLL\_BYPASS 和 CLK\_SEL 都为 0，clk\_fload 为 37.5MHz。

clk\_src 由 pll\_bypass 控制，默认为 clk\_pll，即 150MHz。

FCLK 及其门控分频时钟由 cpu\_clk\_div 控制，默认 FCLK 为 37.5MHz。

## 4. 电源及电压检测

### 4.1 简介

- 监测电源欠压输入端口 VHH\_SAMP, 当电压低于设定阈值时, 产生中断信号。
- 监测电源过压输入端口 VHH\_SAMP\_OVP, 当电压高于设定阈值时, 产生中断信号。
- 监测系统电源 AVDD, 根据设定的阈值产生 POR 复位信号。

### 4.2 电源及电压检测寄存器

#### 4.2.1 电源中断控制寄存器 (SYS\_PIE)

地址: 0x4001F81C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留													OVIE	OVDE	PDIE
													rW	rW	rW
位 31:3		保留													
位 2		OVIE: 过压检测中断使能, 高有效, 默认为 0													
位 1		OVDE: 过压检测使能, 高有效, 默认为 0													
位 0		PDIE: 掉电检测中断使能, 高有效, 默认为 0													

#### 4.2.1 电源中断状态寄存器 (SYS\_PIS)

地址: 0x4001F820

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
保留																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
保留													OVIS	OVS	PDIS	PDS
													rW	rW	rW	rW
位 31:4		保留														

位 3	OVIS: 过压检测中断标志位, 写 1 清零
位 2	OVS: 过压标志位, 1: 12V 电压升高超过设定阈值
位 1	PDIS: 掉电检测中断标志位, 写 1 清零
位 0	PDS: 掉电标志位, 1: 12V 电压降至设定阈值

### 4.2.3 过压检测阈值设置寄存器(SYS\_OCVS)

地址: 0x4001F824

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留													OCVS		
													rw	rw	rw

位 31:3	保留
位 2:0	OCDS: 过压比较器阈值控制信号 000~111 : 0.3V~1V 0.1V /step 默认设置: 000 (0.3V)

## 5. 低功耗

### 5.1 简介

#### 5.1.1 睡眠模式

通过 CortexM0 的系统自带指令 WFI 可以进入睡眠模式: `__WFI()`;

进入睡眠模式后, 系统 core 时钟 HCLK 停止。

在睡眠模式下, 通过任一中断都可以唤醒 core 继续工作。

#### 5.1.2 深度睡眠模式

将系统控制寄存器中的深度睡眠控制位置 1, 然后通过 WFI 指令可以使芯片进入深度睡眠模式, 此时 core 时钟 HCLK 和外设时钟 PCLK 都会停止, 只能通过看门狗中断和外部引脚中断唤醒系统继续工作。

## 5.2 特殊功能寄存器

### 5.2.1 锁定寄存器 (LP\_LOCK)

地址: 0x4000880C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ULOCK [31:16]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ULOCK [15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

位 31:0	ULOCK: 写入 0x5500CC01 解除锁定, 写入其他数值锁定
--------	-------------------------------------

### 5.2.2 锁定标志位寄存器 (LP\_LS)

地址: 0x4000880C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留															LS

r

位 31:1	保留
位 0	LS: 锁定标志位, 只读

### 5.2.3 睡眠控制寄存器 (LP\_SC)

地址: 0xE000ED10

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留													DS	SOE	保留
													rw	rw	

位 31:5	保留
位 2	DS: 深度睡眠设置位
位 1	SOE: SLEEP_ON_EXIT
位 0	保留

解除锁定后才能对 LP\_SC 写数据，其他寄存器写不需要解锁。

## 6. 通用/复用功能 IO(GPIO/AFIO)

### 6.1 GPIO 功能概述

该芯片有 PA, PB, PC 三个 16 位端口，包含 48 个双向 IO 引脚。每个 GPIO 端口有十个配置寄存器，两个数据寄存器，一个置位寄存器，一个复位寄存器，一个位取反寄存器，一个锁定寄存器。GPIO 的每位可以由软件分别配置成多种模式：输入浮空，输入上拉，输入下拉，开漏输出，推挽输出，复用功能。

端口可配置为复用功能端口和普通 GPIO，默认状态下，除 PA[13]外都为普通 GPIO，PA[13]默认为串行调试接口数据输入/输出 SWDIO。端口如果需要作为复用功能端口使用，比如 SPI, I2C 等接口，需要将功能设置寄存器 PAFS 相应位置 1；若要将复用功能 IO 重新配置为 GPIO 时，需先将 PAFS 相应位清 0。如果将端口配置为第二复用功能，同时需要将 PAFR 相应位置 1。

GPIO 默认状态为浮空输入，若需要配置为输出端口时，要将 POENS 寄存器相应位置 1；若要重新配置为输入端口时，需将 POENS 相应位清 0。

POD 相应位置 1，即可将端口作为开漏输出端口使用。将 POD 相应位清 0，即可重置为推挽方式。

BSR, BRR 和 BTR 相应位置 1，由硬件对 ODR\_x 位进行置位，清 0 或者取反，大幅度提升了 IO 口的响应速度。

GPIO 每个端口都可以作为外部中断输入，此时需要配置相应端口的中断使能，并配置中断类型和极性，中断响应后，通过中断状态寄存器可查询是哪个端口发出中断请求，向中断状态寄存器相应位写 1 可清除该标志位。

### 6.2 GPIO 寄存器

GPIOA 基址: 0x4001 0000

GPIOB 基址: 0x4001 1000

GPIOC 基址: 0x4001 2000



## 6.2.1 端口输入数据寄存器 (GPIOx\_IDR)

偏移地址: 0x00

复位值: 0x0000 XXXX



Bit31:Bit16	保留位, 硬件强制为 0.
Bit15:Bit0	IDRy[15:0]: 端口(y=0..15)输入数据 该寄存器为只读寄存器, 并只能以字(16 位)的形式读出, 读出的值为对应 I/O 口的状态。

## 6.2.2 端口输出数据寄存器 (GPIOx\_ODR)

偏移地址: 0x04

复位值: 0x0000 0000



Bit31:Bit16	保留位, 硬件强制为 0.
Bit15:Bit0	ODRy[15:0]: 端口(y=0..15)输出数据 该寄存器为可读可写寄存器, 并只能以字(16 位)的形式操作, 读出的值为当前 I/O 口的输出状态。

## 6.2.3 端口位置寄存器 (GPIOx\_BSR)

偏移地址: 0x08

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BSR1	BSR1	BSR1	BSR1	BSR1	BSR1	BSR9	BSR8	BSR7	BSR6	BSR5	BSR4	BSR3	BSR2	BSR1	BSR0
5	4	3	2	1	0										
W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit31:Bit16	保留位, 硬件强制为 0.
Bit15:Bit0	BSRy: 将端口 (y=0..15) 的相应位置 1 该寄存器为只写寄存器, 可以按位操作。 0: 对对应的 ODRy 位不产生影响 1: 置对应的 ODRy 位为 1

### 6.2.4 端口位清除寄存器 (GPIOx\_BRR)

偏移地址: 0x0C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BRR1	BRR1	BRR1	BRR1	BRR1	BRR1	BRR9	BRR8	BRR7	BRR6	BRR5	BRR4	BRR3	BRR2	BRR1	BRR0
5	4	3	2	1	0										
W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit31:Bit16	保留位, 硬件强制为 0.
Bit15:Bit0	BRRy: 将端口 (y=0..15) 的位清 0 该寄存器为只写寄存器, 可以按位操作。 0: 对对应的 ODRy 位不产生影响 1: 清除对应的 ODRy 位为 0

### 6.2.5 端口位取反寄存器 (GPIOx\_BTR)

偏移地址: 0x10

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

保留

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BTR	BTR	BTR	BTR	BTR	BTR	BTR	BTR	BTR	BTR	BTR	BTR	BTR	BTR	BTR	BTR
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit31:Bit16	保留位，硬件强制为 0。
Bit15:Bit0	BTRy: 将端口 (y=0..15) 的位取反 该寄存器为只写寄存器，可以按位操作。 0: 对对应的 ODRy 位不产生影响 1: 将对应的 ODRy 位取反

### 6.2.6 端口输出配置寄存器 (GPIOx\_POSR)

偏移地址: 0x14

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
POSR	POSR	POSR	POSR	POSR	POSR	POSR	POSR	POSR	POSR	POSR	POSR	POSR	POSR	POSR	POSR0
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit31:Bit16	保留位，硬件强制为 0。
Bit15:Bit0	POSRy: 端口 (y=0..15) 位输入输出配置 该寄存器为可读可写寄存器，并只能以字 (16 位) 的形式操作。 0: 输入模式 1: 输出模式

### 6.2.7 端口上拉配置寄存器 (GPIOx\_PUR)

偏移地址: 0x18

复位值: 0x2000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PUR	PUR	PUR	PUR	PUR	PUR	PUR	PUR	PUR	PUR	PUR	PUR	PUR	PUR	PUR	PUR
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

Bit31:Bit16	保留位，硬件强制为 0.
Bit15:Bit0	PURy: 端口 (y=0..15) 位上拉配置 该寄存器为可读可写寄存器，并只能以字 (16 位) 的形式操作。 0: 禁止上拉 1: 使能上拉

### 6.2.8 端口下拉配置寄存器 (GPIOx\_PDR)

偏移地址: 0x1C

复位值: 0x1000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PDR	PDR	PDR	PDR	PDR	PDR	PDR	PDR	PDR	PDR	PDR	PDR	PDR	PDR	PDR	PDR
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

Bit31:Bit16	保留位，硬件强制为 0.
Bit15:Bit0	PDRy: 端口 (y=0..15) 位下拉配置 该寄存器为可读可写寄存器，并只能以字 (16 位) 的形式操作。 0: 禁止下拉 1: 使能下拉

### 6.2.9 端口开漏配置寄存器 (GPIOx\_PODR)

偏移地址: 0x20

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
POD R15	POD R14	POD R13	POD R12	POD R11	POD R10	POD R9	POD R8	POD R7	POD R6	POD R5	POD R4	POD R3	POD R2	POD R1	POD R0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit31:Bit16	保留位，硬件强制为 0.
Bit15:Bit0	PODRy：端口 (y=0..15) 位开漏/推挽配置 该寄存器为可读可写寄存器，并只能以字 (16 位) 的形式操作。 0：推挽输出 1：开漏输出

### 6.2.10 端口安全锁寄存器 (GPIOx\_LOCK)

偏移地址：0x40

复位值：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LOCK_KEY[15:8]								保留							LEN
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit31:Bit16	保留位，硬件强制为 0.
Bit15:Bit8	LOCK_KEY[15:8]：安全锁关键字 仅当输入字节为 0xAC 时，对该寄存器的 Bit0 位操作才有效。
Bit7:Bit1	保留，硬件强制为 0
Bit0	LEN：位锁定寄存器 (GPIOx_BLR) 操作使能位 0：对位锁定寄存器 (GPIOx_BLR) 操作有效 1：对未锁定寄存器 (GPIOx_BLR) 操作无效

### 6.2.11 端口位锁定寄存器 (GPIOx\_BLR)

偏移地址：0x44

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

BLR1	BLR1	BLR1	BLR1	BLR1	BLR1	BLR9	BLR8	BLR7	BLR6	BLR5	BLR4	BLR3	BLR2	BLR1	BLR0
5	4	3	2	1	0										
I <sup>W</sup>	I <sup>W</sup>	I <sup>W</sup>	I <sup>W</sup>	I <sup>W</sup>	I <sup>W</sup>	I <sup>W</sup>	I <sup>W</sup>	I <sup>W</sup>	I <sup>W</sup>	I <sup>W</sup>	I <sup>W</sup>	I <sup>W</sup>	I <sup>W</sup>	I <sup>W</sup>	I <sup>W</sup>

Bit31:Bit16	保留位，硬件强制为 0。
Bit15:Bit0	BLRy[15:0]：端口位状态锁定 0：不锁定端口位操作 1：锁定端口位操作 锁定端口位操作后，对该位的操作均无效。

## 6.3 复用功能寄存器

复用功能端口映射表如下表所示。

GPIO	复用功能 1	复用功能 2
PA0	ISP 引脚选择	
PA1	CK0	
PA2	SPI0_SCS	IIC0_SCL
PA3	SPI0_SCK	IIC0_DIO
PA4	SPI0_SDI	TIMER2_EXTOUT
PA5	SPI0_SDO	TIMER3_EXTOUT
PA6	UART0_RXD	SPI1_SCS
PA7	UART0_TXD	SPI1_SCK
PA8	UART1_RXD	SPI1_SDI
PA9	UART1_TXD	SPI1_SDO
PA10	TRIMERO_EXTOUT	UART3_RXD
PA11	TRIMER1_EXTOUT	UART3_TXD
PA12	SWD_CLK	ISP_SCL①
PA13	SWD_DIO	ISP_SDA①
PA14	IIC0_SCL	
PA15	IIC0_DIO	
PB0	UART2_RXD	SSPI_SCS
PB1	UART2_TXD	SSPI_SCK
PB2	UART3_RXD	SSPI_SDI
PB3	UART3_TXD	SSPI_SDO
PB4	TIMER2_EXTOUT	IIC1_CLK
PB5	TIMER3_EXTOUT	IIC1_SDIO
PB6	SPI1_SCS	UART0_RXD
PB7	SPI1_SCK	UART0_TXD
PB8	SPI1_SDI	UART1_RXD

PB9	SPI1_SDO	UART1_TXD
PB10	UART4_RXD	TRIMERO_EXTOUT
PB11	UART4_TXD	TRIMER1_EXTOUT
PB12	IIC11_CLK	UART4_RXD
PB13	IIC1_SDI0	UART4_TXD
PB14	RTC 1Hz 输出	
PB15	50Hz 工频信号输入	
PC0	SPI2_SCS	
PC1	SPI2_SCK	
PC2	SPI2_SDI	
PC3	SPI2_SDO	

### 6.3.1 端口复用功能配置寄存器 (GPIOx\_PAFR)

偏移地址: 0x24

复位值: 0x2000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PAFR	PAFR	PAFR	PAFR	PAFR	PAFR	PAFR	PAFR	PAFR	PAFR	PAFR	PAFR	PAFR	PAFR	PAFR	PAFR0
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit31:Bit16	保留位，硬件强制为 0。
Bit15:Bit0	PAFR[15:0]: 端口位复用功能使能 0: 该端口位作普通 IO 用 1: 该端口位作复用功能用 IO 作复用功能用时，端口由硬件自动配置上拉或下拉，开漏或推挽模式，此时用户对 IO 的配置无效。

### 6.3.2 端口复用功能选择寄存器 (GPIOx\_PAFSR)

偏移地址: 0x28

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

保留

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PAFS	PAFS	PAFS	PAFS	PAFS	PAFS	PAFS	PAFS	PAFS	PAFS	PAFS	PAFS	PAFS	PAFS	PAFS	PAFSR
R15	R14	R13	R12	R11	R10	R9	R8	R7	R6	R5	R4	R3	R2	R1	0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit31:Bit16	保留位，硬件强制为 0.
Bit15:Bit0	PAFSR[15:0]：端口位复用功能选择 0：端口复用功能 1 1：端口复用功能 2 当 IO 作复用功能时，可通过配置该寄存器来选择端口位作为复用功能 1 还是复用功能 2.

## 6.4 GPIO 中断寄存器

### 6.4.1 外部中断使能寄存器 (GPIOx\_PIER)

偏移地址：0x2C

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PIER	PIER	PIER	PIER	PIER	PIER	PIER	PIER	PIER	PIER	PIER	PIER	PIER	PIER	PIER	PIERO
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit31:Bit16	保留位，硬件强制为 0.
Bit15:Bit0	PIER[15:0]：端口外部中断使能 0：禁止中断 1：使能中断

### 6.4.2 外部中断类型配置寄存器 (GPIOx\_PITR)

偏移地址：0x30

复位值：0x0000 0000



31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16

保留															
----	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

PI TR	PI TR	PI TR	PI TR	PI TR	PI TR	PI TR	PI TR	PI TR	PI TR	PI TR	PI TR	PI TR	PI TR	PI TR	PI TR
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
I W	I W	I W	I W	I W	I W	I W	I W	I W	I W	I W	I W	I W	I W	I W	I W

Bit31:Bit16	保留位，硬件强制为 0.
Bit15:Bit0	PI TR[15:0]：端口外部中断触发类型选择 0：电平触发 1：边沿触发

中断类型寄存器	中断极性寄存器	中断触发类型
0	0	低电平触发
0	1	高电平触发
1	0	下降沿触发
1	1	上升沿触发

### 6.4.3 外部中断极性配置寄存器 (GPIOx\_PIPR)

偏移地址：0x34

复位值：0x0000 0000

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16

保留															
----	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

PI PR	PI PR	PI PR	PI PR	PI PR	PI PR	PI PR	PI PR	PI PR	PI PR	PI PR	PI PR	PI PR	PI PR	PI PR	PI PR
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
I W	I W	I W	I W	I W	I W	I W	I W	I W	I W	I W	I W	I W	I W	I W	I W

Bit31:Bit16	保留位，硬件强制为 0.
Bit15:Bit0	PI PR[15:0]：端口外部中断触发极性选择 0：低电平触发(电平触发)/下降沿触发(边沿触发) 1：高电平触发(电平触发)/上升沿触发(边沿触发)

中断类型寄存器	中断极性寄存器	中断触发类型
0	0	低电平触发
0	1	高电平触发
1	0	下降沿触发

1	1	上升沿触发
---	---	-------

### 6.4.4 外部中断边沿触发选择寄存器 (GPIOx\_PIESR)

偏移地址: 0x38

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PIES	PIES	PIES	PIES	PIES	PIES	PIES	PIES	PIES	PIES	PIES	PIES	PIES	PIES	PIES	PIESR
R15	R14	R13	R12	R11	R10	R9	R8	R7	R6	R5	R4	R3	R2	R1	0
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

Bit31:Bit16	保留位, 硬件强制为 0.
Bit15:Bit0	PIESR[15:0]: 端口外部中断边沿触发选择 0: 单边沿触发 (由寄存器 GPIOx_PIPR 决定上升沿或下降沿触发) 1: 双边沿触发 (上升沿和下降沿均触发中断)

### 6.4.5 外部中断状态寄存器 (GPIOx\_PISR)

偏移地址: 0x3C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PISR	PISR	PISR	PISR	PISR	PISR	PISR	PISR	PISR	PISR	PISR	PISR	PISR	PISR	PISR	PISR0
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

Bit31:Bit16	保留位, 硬件强制为 0.
Bit15:Bit0	PISR[15:0]: 端口外部中断状态标志 0: 无中断请求 1: 对应位产生中断请求 该寄存器可读可写, 写 1 清除中断标志位, 写 0 无效。

## 7. 中断模块

### 7.1 中断向量说明

系统中断	中断号	功能描述
NMI	X	不可屏蔽中断
GPIOA	0	PA 任一端口中断
GPIOB	1	PB 任一端口中断
GPIOC	2	PC 任一端口中断
PW	4	电压检测中断
DMA	5	DMA 中断
CARR	6	载波中断
UART0	8	串口 0 中断
UART1	9	串口 1 中断
UART2	10	串口 2 中断
UART3	11	串口 3 中断
UART4	12	串口 4 中断
SPI0	13	SPI0 中断
SPI1	14	SPI1 中断
SPI2	15	SPI2 中断
SSPI	16	SPI3 中断
TIMERO	17	定时器 0 中断
TIMER1	18	定时器 1 中断
IIC0	19	IIC0 中断
IIC1	20	IIC1 中断
WDT	22	看门狗中断
AES/3DES	25	AES 高级加密中断
TIMER2	27	定时器 2 中断
TIMER3	28	定时器 3 中断
GPIOA0	29	PA0 口中断
GPIOA1	30	PA1 口中断
GPIOA2	31	PA2 口中断

### 7.2 中断寄存器

#### 7.2.1 中断使能寄存器(NVIC\_ISER)

地址: 0xE000E100

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ISE3	ISE3	ISE2	ISE2	ISE2	ISE2	ISE2	ISE2	ISE2	ISE2	ISE2	ISE2	ISE1	ISE1	ISE1	ISE1
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ISE1	ISE1	ISE1	ISE1	ISE1	ISE1	ISE9	ISE8	ISE7	ISE6	ISE5	ISE4	ISE3	ISE2	ISE1	ISE0
5	4	3	2	1	0										
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

Bit 31:0	中断使能配置寄存器：ISE <sub>x</sub> (0..31) 中断源 x 使能。写 1 置 1，使能中断，写 0 无效，每一位的对应关系见中断向量说明。推荐使用 CortexM0 库函数中断 NVIC_EnableIRQ 来使能中断 0：无效 1：使能中断
----------	--

### 7.2.2 中断禁止寄存器 (NVIC\_ICER)

地址：0xE000E180

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ICE3	ICE	ICE	ICE	ICE	ICE	ICE	ICE	ISE2	ICE2	ICE2	ICE2	ICE1	ICE1	ICE1	ICE1
1	30	29	28	27	26	25	24	3	2	1	0	9	8	7	6
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ICE1	ICE1	ICE1	ICE1	ICE1	ICE1	ICE9	ICE8	ICE7	ICE6	ICE5	ICE4	ICE3	ICE2	ICE1	ICE0
5	4	3	2	1	0										
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

Bit 31:0	中断禁止配置寄存器：ICE <sub>x</sub> (0..31) 中断源 x 失能。写 1 清 0，禁止中断，写 0 无效，每一位的对应关系见中断向量说明。该寄存器和 ISER 联动，如果将 ICER 的 bit0 置 1，则 ICER 和 ISER 的 bit0 都会被清 0。推荐使用库函数中的 NVIC_DisableIRQ 来禁止中断。 0：无效 1：禁止中断
----------	--

### 7.2.3 中断标志寄存器 (NVIC\_ISR)

地址: 0xE000E200

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IS31	IS30	IS29	IS28	IS27	IS26	IS25	IS24	IS23	IS22	IS21	IS20	IS19	IS18	IS17	IS16
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IS15	IS14	IS13	IS12	IS11	IS10	IS9	IS8	IS7	IS6	IS5	IS4	IS3	IS2	IS1	IS0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit 31:0	中断标志置位寄存器: ISx (0..31) 中断源 x 中断标志。写 1 置 1, 写 0 无效, 可通过查询该寄存器来查询内核中断标志, 每一位的对应关系见中断向量说明。推荐使用库函数中的 NVIC_SetPendingIRQ 来置位中断标志 0: 无效 1: 置位中断
----------	---

### 7.2.4 中断清标志寄存器 (NVIC\_ICSR)

地址: 0xE000E280

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ICS3	ICS3	ICS2	ICS2	ICS2	ICS2	ICS2	ICS2	ICS2	ICS2	ICS2	ICS2	ICS1	ICS1	ICS1	ICS1
1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ICS1	ICS1	ICS1	ICS1	ICS1	ICS1	ICS9	ICS8	ICS7	ICS6	ICS5	ICS4	ICS3	ICS2	ICS1	ICS0
5	4	3	2	1	0										
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit 31:0	中断标志清零寄存器: ISx (0..31) 清中断源 x 中断标志。写 1 清 0, 清除中断标志, 写 0 无效, 该寄存器和 ISER 联动, 如果将 ICPR 的 bit0 置 1, 则 ICPR 和 ISPR 的 bit0 都会被清 0, 每一位的对应关系见中断向量说明。推荐使用库函数中的 NVIC_ClearPendingIRQ 来清中断标志 0: 无效 1: 清中断标志
----------	---

## 7.2.5 中断优先级 0 寄存器 (NVIC\_IRPR0)

地址: 0xE000E400

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRP3		保留						IRP2		保留					
rW		rW						rW		rW					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IRP1		保留						IRP0		保留					
rW		rW						rW		rW					

Bit 31:30	IRP3: 中断 3 优先级寄存器
Bit 29:24	保留
Bit 23:22	IRP2: 中断 2 优先级寄存器
Bit 21:16	保留
Bit 15:14	IRP1: 中断 1 优先级寄存器
Bit 13:8	保留
Bit 7:6	IRP0: 中断 0 优先级寄存器
Bit 5:0	保留

## 7.2.6 中断优先级 1 寄存器 (NVIC\_IRPR1)

地址: 0xE000E404

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRP7		保留						IRP6		保留					
rW		rW						rW		rW					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IRP5		保留						IRP4		保留					
rW		rW						rW		rW					

Bit 31:30	IRP7: 中断 7 优先级寄存器
Bit 29:24	保留
Bit 23:22	IRP6: 中断 6 优先级寄存器
Bit 21:16	保留
Bit 15:14	IRP5: 中断 5 优先级寄存器

Bit 13:8	保留
Bit 7:6	IRP4: 中断 4 优先级寄存器
Bit 5:0	保留

### 7.2.7 中断优先级 2 寄存器 (NVIC\_IRPR2)

地址: 0xE000E408

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRP11		保留						IRP10		保留					
rw rw								rw rw							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IRP9		保留						IRP8		保留					
rw rw								rw rw							

Bit 31:30	IRP11: 中断 11 优先级寄存器
Bit 29:24	保留
Bit 23:22	IRP10: 中断 10 优先级寄存器
Bit 21:16	保留
Bit 15:14	IRP9: 中断 9 优先级寄存器
Bit 13:8	保留
Bit 7:6	IRP8: 中断 8 优先级寄存器
Bit 5:0	保留

### 7.2.8 中断优先级 3 寄存器 (NVIC\_IRPR3)

地址: 0xE000E40C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRP15		保留						IRP14		保留					
rw rw								rw rw							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IRP13		保留						IRP12		保留					
rw rw								rw rw							
Bit 31:30	IRP15: 中断 15 优先级寄存器														

Bit 29:24	保留
Bit 23:22	IRP14: 中断 14 优先级寄存器
Bit 21:16	保留
Bit 15:14	IRP13: 中断 13 优先级寄存器
Bit 13:8	保留
Bit 7:6	IRP12: 中断 12 优先级寄存器
Bit 5:0	保留

### 7.2.9 中断优先级 4 寄存器(NVIC\_IRPR4)

地址: 0xE000E410

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRP19		保留						IRP18		保留					
rw								rw							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IRP17		保留						IRP16		保留					
rw								rw							

Bit 31:30	IRP19: 中断 19 优先级寄存器
Bit 29:24	保留
Bit 23:22	IRP18: 中断 18 优先级寄存器
Bit 21:16	保留
Bit 15:14	IRP17: 中断 17 优先级寄存器
Bit 13:8	保留
Bit 7:6	IRP16: 中断 16 优先级寄存器
Bit 5:0	保留

### 7.2.10 中断优先级 5 寄存器(NVIC\_IRPR5)

地址: 0xE000E414

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRP23		保留						IRP22		保留					
rw								rw							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0



IRP21	保留	IRP20	保留
rW	rW	rW	rW

Bit 31:30	IRP23: 中断 23 优先级寄存器
Bit 29:24	保留
Bit 23:22	IRP22: 中断 22 优先级寄存器
Bit 21:16	保留
Bit 15:14	IRP21: 中断 21 优先级寄存器
Bit 13:8	保留
Bit 7:6	IRP20: 中断 20 优先级寄存器
Bit 5:0	保留

### 7.2.11 中断优先级 6 寄存器(NVIC\_IRPR6)

地址: 0xE000E418

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IRP27	保留				IRP26	保留									
rW	rW				rW	rW									
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IRP25	保留				IRP24	保留									
rW	rW				rW	rW									

Bit 31:30	IRP27: 中断 27 优先级寄存器
Bit 29:24	保留
Bit 23:22	IRP26: 中断 26 优先级寄存器
Bit 21:16	保留
Bit 15:14	IRP25: 中断 25 优先级寄存器
Bit 13:8	保留
Bit 7:6	IRP24: 中断 24 优先级寄存器
Bit 5:0	保留

### 7.2.12 中断优先级 7 寄存器(NVIC\_IRPR7)

地址: 0xE000E41C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

IRP31	保留	IRP30	保留
-------	----	-------	----

rW rW

rW rW

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

IRP29	保留	IRP28	保留
-------	----	-------	----

rW rW

rW rW

Bit 31:30	IRP31: 中断 31 优先级寄存器
Bit 29:24	保留
Bit 23:22	IRP30: 中断 30 优先级寄存器
Bit 21:16	保留
Bit 15:14	IRP29: 中断 29 优先级寄存器
Bit 13:8	保留
Bit 7:6	IRP28: 中断 28 优先级寄存器
Bit 5:0	保留

推荐使用库函数中的 NVIC\_SetPriority 来设置中断优先级

## 8. 复位模块

### 8.1 复位说明

系统共有 4 种复位方式。

序号	复位原	复位方式	说明
1	上电复位 (POR)	低电平复位，上电时，外部电源电压大于 2.4V 时 POR 变为高电平。	上电复位会延续 10ms，用于外部 flash 的稳定。 上电复位能够复位系统所有寄存器
2	外部引脚/MCLR 复位	低电平复位	能够滤除 100us 以内的毛刺。 能够复位除 RTC 外其它所有寄存器
3	看门狗复位	由看门狗模块产生高脉冲引发复位	不能引起 PORESETn 和 DBGRESETn 复位信号，即部分系统控制寄存器和 DEBUG 模块不会复位
4	系统软复位	通过设置系统寄存器的复位位为 1，产生高脉冲引发复位	能够复位 CPU 核和外设，包括看门狗和载波

### 8.2 复位模块寄存器

## 8.2.1 复位状态寄存器 (SYS\_RSTS)

地址: 0x4001F010

复位值: 0x0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留												ER	LS	WR	SR
												r w	r w	r w	r w

位 31:4	保留
位 3	ER: 外部引脚复位标志, 外部引脚复位自动置 1, 软件清 0
位 2	LS: 锁定复位标志, 锁定复位自动置 1, 软件清 0
位 1	WR: 看门狗复位标志, 看门狗复位自动置 1, 软件清 0
位 0	SR: 软复位标志, 软复位自动置 1, 软件清 0

## 9. 通用异步收发器 (UART)

### 9.1 主要特性

- 全双工, 异步通信
- 波特率设定: [31:0]
- 接收和发送可以通过查询或中断方式实现
- 支持极性反转配置
- 支持停止位 1 位, 2 位
- 支持校验: 无校验, 奇校验, 偶校验和强制 1/0 校验
- UART0, UART1 支持红外调制发送, 输出极性、频率可配置

### 9.2 UART 功能概述

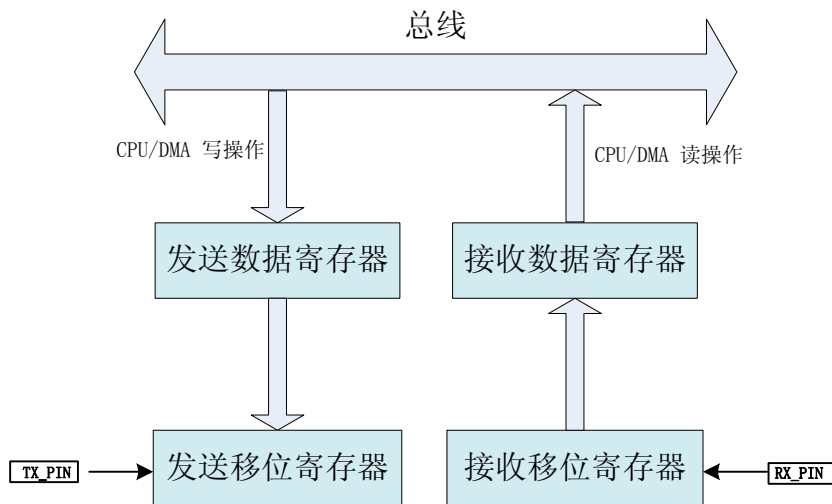
接口通过三个引脚预其他设备连接。任何 UART 双向通信至少需要两个脚: 接收数据输入 (UART\_RXD) 和发送数据输出 (TX)。

使能 UART 功能后, 硬件自动将 IO 端口配置为 UART 端口模式, 外部配置上/下拉或开漏/推挽无效。不发送数据时, UART\_TXD 处于高电平状态。

- 总线在发送或接收前应处于空闲状态

- 一个起始位
- 数据位 8 位
- 停止位 1 位或 2 位
- 波特率由 20Bit 有效位分频产生
- 状态寄存器 (UART\_SR)
- 控制寄存器 (UART\_CR1/CR2)
- 数据寄存器 (UART\_DR)
- 波特率设置寄存器, 20Bit (UART\_BRR)
- 中断状态寄存器 (UART\_ISR)
- 红外控制寄存器 0/1
- 红外频率设置寄存器 0/1

通用异步收/发器 (UART) 对从外设收到的数据执行串到并的转换, 对从 CPU 发送的数据执行并到串的转换。该串口同时支持红外调制, 输出极性、脉冲宽度可配。



### 9.3 IR 功能概述

UART0、UART1 均支持红外调制发送功能, 其中 UART0 可使能为 IR0, UART1 可使能为 IR1。

### 9.4 UART 寄存器

UART0 基址: 0x4000 2000

UART1 基址: 0x4000 2800

UART2 基址: 0x4000 3000

UART3 基址: 0x4000 3800

UART4 基址: 0x4000 4000

**注: 配置寄存器时需要先使能再填写数据。**

### 9.4.1 数据寄存器 (UARTx\_DR)

偏移地址: 0x00

复位值: 0x0000 0000



Bit31:Bit8	保留位, 硬件强制为 0.
Bit7:Bit0	DR[7:0]:UART 数据值 包含了发送和接收的数据。实际它是由两个寄存器组成的, 一个发送寄存器, 一个接收寄存器, 该寄存器兼具读和写的功能。发送寄存器提供了内部总线和输出移位寄存器之间的并行接口, 接收寄存器提供和输入寄存器和内部总线之间的并行接口。

### 9.4.2 状态寄存器 (UARTx\_SR)

偏移地址: 0x04

复位值: 0x0000 0000



Bit31:Bit7	保留位, 硬件强制为 0.
Bit6	TC:发送完成 0:发送还未完成 1:发送完成 当包含有数据的一帧发送完成后, 由硬件将该位置 1。如果 UARTx_CR1 寄存器的 TCIE 位为 1, 则 UARTx_ISR 寄存器的 TCIR 位由硬件置 1, 并产生中断。在向寄存器 DR 写入数据时, 硬件将该位清零。
Bit5	PE:校验错误 0:没有校验错误 1:校验错误

	<p>在接收模式下，如果出现数据校验错误，硬件将该位置 1。如果 UARTx_CR1 寄存器的 PEIE 位为 1，则 UARTx_ISR 寄存器的 PEIR 位由硬件置 1，并产生中断。</p> <p>写 1 清零，写 0 无效。</p>
Bit4	<p>FE:帧错误</p> <p>0:没有检测到帧错误</p> <p>1:检测到帧错误</p> <p>在接收模式下，如果检测到同步错位，硬件将该位置 1。如果 UARTx_CR1 寄存器的 FEIE 位为 1，则 UARTx_ISR 寄存器的 FEIR 位由硬件置 1，并产生中断。</p> <p>写 1 清零，写 0 无效。</p>
Bit3	<p>RXOR:接收寄存器溢出错误</p> <p>0:没有溢出</p> <p>1:检测到溢出错误</p> <p>在接收模式下，当 RXNE 为 1 时，当前移位寄存器中的数据要往 RDR 寄存器中传送时，硬件将该位置 1。如果 UARTx_CR1 寄存器的 RXORIE 位为 1，则 UARTx_ISR 寄存器的 RXORIR 位由硬件置 1，并产生中断。</p> <p>写 1 清零，写 0 无效。</p>
Bit2	<p>TXOR:发送寄存器溢出错误</p> <p>0:没有溢出</p> <p>1:检测到溢出错误</p> <p>在发送模式下，当 TXNE 为 1 时，向 DR 寄存器写入数据，硬件将该位置 1。如果 UARTx_CR1 寄存器的 TXORIE 位为 1，则 UARTx_ISR 寄存器的 TXORIR 位由硬件置 1，并产生中断。</p> <p>写 1 清零，写 0 无效。</p>
Bit1	<p>RXNE:读数据寄存器非空</p> <p>0:数据没有收到</p> <p>1:收到数据，可以读出</p> <p>当 RDR 移位寄存器中的数据被转移到 UARTx_DR 寄存器时，硬件将该位置 1。如果 UARTx_CR1 寄存器的 RXIE 位为 1，则 UARTx_ISR 寄存器的 RXIR 位由硬件置 1，并产生中断。</p> <p>该位只读。</p>
Bit0	<p>TXNE:发送数据寄存器非空</p> <p>0:数据已经被转移到移位寄存器</p> <p>1:数据还未被转移到移位寄存器</p> <p>当 TDR 寄存器中的数据被硬件转移到移位寄存器时，该位被硬件清零。如果 UARTx_CR1 寄存器的 TXIE 位为 1，则 UARTx_ISR 寄存器的 TXIR 位由硬件置 1，并产生中断。</p> <p>该位只读。</p>

### 9.4.3 控制寄存器 1 (UARTx\_CR1)

偏移地址: 0x08

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留				TCIE	DMAR	DMAT	HTEN	PEIE	FEIE	RXOR	TXOR	RXIE	TXIE	RE	TE
				rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

Bit31:Bit7	保留位, 硬件强制为 0.
Bit11	TCIE:发送完成中断使能 0:中断被禁止 1:中断使能 当 UARTx_ISR 寄存器的 TCIR 位为 1 时, 产生中断。
Bit10	DMAREN:DMA 使能接收 0:接收时的 DMA 模式被禁止 1:接收时的 DMA 模式使能
Bit9	DMATEN:DMA 使能发送 0:发送时的 DMA 模式被禁止 1:发送时的 DMA 模式使能
Bit8	HTEN:高速测试使能 0:高速测试被禁止 1:高速测试使能
Bit7	PEIE:校验错误中断使能 0:中断被禁止 1:中断使能 当 UARTx_ISR 寄存器的 PEIR 位为 1 时, 产生中断。
Bit6	FEIE:帧错误中断使能 0:中断被禁止 1:中断使能 当 UARTx_ISR 寄存器的 FEIR 位为 1 时, 产生中断。
Bit5	RXORIE:接收寄存器溢出中断使能 0:中断被禁止 1:中断使能 当 UARTx_ISR 寄存器的 RXORIR 位为 1 时, 产生中断。
Bit4	TXORIE:发送寄存器溢出中断使能 0:中断被禁止 1:中断使能 当 UARTx_ISR 寄存器的 TXORIR 位为 1 时, 产生中断。

Bit3	RXIE:接收缓冲区非空中断使能 0:中断被禁止 1:中断使能 当 UARTx_ISR 寄存器的 RXIR 位为 1 时, 产生中断。
Bit2	TXIE:发送缓冲区空中断使能 0:中断被禁止 1:中断使能 当 UARTx_ISR 寄存器的 TXIR 位为 1 时, 产生中断。
Bit1	RE:接收使能 0:接收被禁止 1:接收被使能, 开始检测 RX 起始位
Bit0	TE:发送使能 0:发送被禁止 1:发送被使能

#### 9.4.4 中断状态寄存器 (UARTx\_ISR)

偏移地址: 0x0C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留									TCIR	PEIR	FEIR	RXOR IR	TXOR IR	RXIR	TXIR
									rW	rW	rW	rW	rW	rW	rW

Bit31:Bit7	保留位, 硬件强制为 0.
Bit6	TCIR:发送完成中断状态标志位 0:未产生中断请求 1:产生中断请求 当 UARTx_CR1 寄存器 TCIE 位为 1 时, 如果 UARTx_SR 寄存器 TC 位为 1, 硬件将该位置 1。 写 1 清零, 写 0 无效。
Bit5	PEOR:校验错误中断状态标志位 0:未产生中断请求 1:产生中断请求 当 UARTx_CR1 寄存器 PEIE 位为 1 时, 如果 UARTx_SR 寄存器 PE 位为 1, 硬件将该位置 1。 写 1 清零, 写 0 无效。
Bit4	FEIR:帧错误中断状态标志位



	0:未产生中断请求 1:产生中断请求 当 UARTx_CR1 寄存器 FEIE 位为 1 时, 如果 UARTx_SR 寄存器 FE 位为 1, 硬件将该位置 1。 写 1 清零, 写 0 无效。
Bit3	RXORIR:接收溢出中断状态标志位 0:未产生中断请求 1:产生中断请求 当 UARTx_CR1 寄存器 RXORIE 位为 1 时, 如果 UARTx_SR 寄存器 RXOR 位为 1, 硬件将该位置 1。 写 1 清零, 写 0 无效。
Bit2	TXORIR:发送溢出中断状态标志位 0:未产生中断请求 1:产生中断请求 当 UARTx_CR1 寄存器 TXORIE 位为 1 时, 如果 UARTx_SR 寄存器 TXOR 位为 1, 硬件将该位置 1。 写 1 清零, 写 0 无效。
Bit1	RXIR:接收中断状态标志位 0:未产生中断请求 1:产生中断请求 当 UARTx_CR1 寄存器 RXIE 位为 1, 如果 UARTx_SR 寄存器 RXNE 位为 1, 硬件将该位置 1。 写 1 清零, 写 0 无效。
Bit0	TXIR:发送中断状态标志位 0:未产生中断请求 1:产生中断请求 当 UARTx_CR1 寄存器 TXIE 位为 1, 如果 UARTx_SR 寄存器 TXNE 位为 0, 硬件将该位置 1。 写 1 清零, 写 0 无效。

### 9.4.5 波特率配置寄存器(UARTx\_BRR)

偏移地址: 0x10

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留												BRR[19: 16]			
												rW	rW	rW	rW
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BRR[15: 0]															
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

Bit31:Bit20	保留位，硬件强制为 0。
Bit19:Bit0	BRR[19:0]: 波特率分频器除法因子，最小值为 16。

### 9.4.6 控制寄存器 2 (UARTx\_CR2)

偏移地址: 0x14

复位值: 0x0000 0000



Bit31:Bit5	保留位，硬件强制为 0。
Bit4:3	Parity: 校验类型选择 00: 奇校验 01: 偶校验 10: 强制 1 校验 11: 强制 0 校验
Bit2	PEN: 校验使能 0: 无校验 1: 使能校验
Bit1	STOP: 停止位选择 0: 1 位停止位 1: 2 位停止位
Bit0	CONV: 极性反转设置 0: 正常极性 1: 反转极性

### 9.5 IR 寄存器

红外功能寄存器基址: 0x4000 8000

### 9.5.1 IRO 控制寄存器 (IRO\_CR)

偏移地址: 0x00

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留														POL	EN
														rw	rw

Bit31:Bit2	保留位, 硬件强制为 0.
Bit1	POL: 红外 0 极性配置 0: 低电平 1: 高电平 当 IR 功能使能时, UART 输出“1”为 POL 所设电平, 输出“0”为 IR 调制频率的方波信号。
Bit0	EN: 红外 0 使能 0: 红外 0 未使能 1: 红外 0 功能使能 红外功能未使能时, UART 仅作串口通讯用; 红外功能使能后, UART 的 TX、RX 引脚作为红外通讯使用, TX 引脚输出红外调制信号。

### 9.5.2 IRO 调制频率设置寄存器 (IRO\_BRR)

偏移地址: 0x04

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留						BRR[9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit31:Bit10	保留位, 硬件强制为 0.
Bit9:0	BRR[9:0]: 红外 0 调制频率分频因子

### 9.5.3 IR1 控制寄存器 (IR1\_CR)

偏移地址: 0x08

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留														POL	EN
														rw	rw

Bit31:Bit2	保留位, 硬件强制为 0.
Bit1	POL: 红外 1 极性配置 0: 低电平 1: 高电平 当 IR 功能使能时, UART 输出“1”为 POL 所设电平, 输出“0”为 IR 调制频率的方波信号。
Bit0	EN: 红外 1 使能 0: 红外 1 未使能 1: 红外 1 功能使能 红外功能未使能时, UART 仅作串口通讯用; 红外功能使能后, UART 的 TX、RX 引脚作为红外通讯使用, TX 引脚输出红外调制信号。

### 9.5.4 IR1 调制频率设置寄存器 (IR1\_BRR)

偏移地址: 0x0C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留						BRR[9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit31:Bit10	保留位, 硬件强制为 0.
Bit9:0	BRR[9:0]: 红外 1 调制频率分频因子

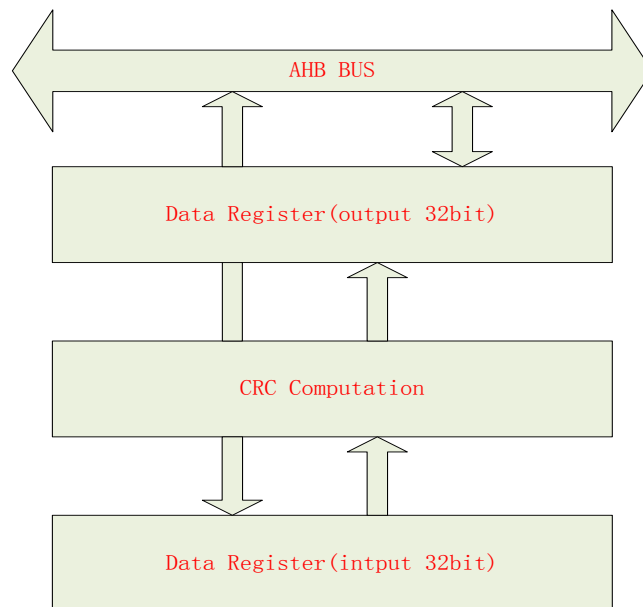
## 10. CRC 计算单元(CRC32/24)

### 10.1 主要特性

- 输入数据支持 8, 16, 32 位
- 支持输入数据按字节、半字、字翻转
- 支持输出数据按位翻转
- CRC 初始值可以预置
- 一个 32 位数据寄存器用于输入/输出
- 4 个 AHB 时钟周期(HCLK)内完成 CRC 运算
- 支持 CRC32 算法, 采用 CRC-32 多项式  $0x04C11DB7$  (以太网)
  - $X^{32}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^8+X^7+X^5+X^4+X^2+X+1$
- 支持 CRC24 算法, 采用 CRC-24 多项式  $0x1800063$ 
  - $X^{24}+X^{23}+X^6+X^5+X+1$

### 10.2 功能描述

CRC计算单元包括一个可读/写的32位数据寄存器CRC\_DR, 写操作时, 输入要进行CRC计算的新数据, 读操作时, 输出CRC运算结果。CRC计算单元框图如下所示。



每次对数据寄存器的写操作都会使得CRC计算单元将所写入的新数据和上次计算得到的结果进行综合计算, 并得到一个新的计算结果。CRC 计算单元根据写入数据的格式不同来决定是整字计算还是一个字节一个字节计算。

32 位需要4 个AHB 时钟周期

16 位需要2 个AHB 时钟周期

8 位需要1个AHB 时钟周期

输入数据高低位可以颠倒，以适应各种不同的大小端体系。数据颠倒操作可以按找8 位、16位和32 位进行，这个功能由CRCSET寄存器中的REV\_IN[1:0] 位来选择设置。

例如：输入数据0x1A2B3C4D 用于CRC 计算

- ..... 按字节翻转就是：0x58D43CB2
- ..... 按半字翻转就是：0xD458B23C
- ..... 按整字翻转就是：0xB23CD458

输入数据也可以通过设置CRCSET 寄存器中的REV\_OUT 位来自动颠倒。该操作是按位进行的：例如，输出数据0x11223344转换过来就是0x22CC4488。

可以通过CRCSET寄存器中的RESET 控制位将CRC计算器的初值初始化为一个特定的值，这个值可以通过CRC\_INIT寄存器设置，默认为0xFFFFFFFF。

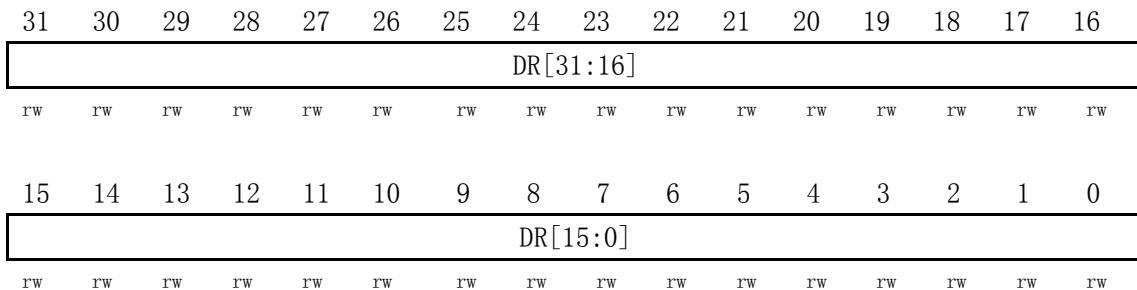
## 10.3 CRC 寄存器

CRC 寄存器基地址：0x40013000

### 10.3.1 数据寄存器(CRC\_DR)

偏移地址：0x00

复位值：0x0000 0000



Bit31:0	DR[31:0]:数据寄存器 写入：要进行 CRC 计算的新数据，支持 8 位、16 位、32 位 读取：返回 CRC 计算的结果
---------	---

### 10.3.2 控制寄存器(CRC\_CR)

偏移地址：0x04

复位值：0x0000 0000



保留															
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留												SEL	RV_0	RV_I	RST
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit31:5	保留位，硬件强制为 0。
Bit4	SEL: CRC 单元选择位 0: CRC32 1: CRC24
Bit3	RV_0: 输出数据按位翻转使能 0: 输出数据不按位翻转 1: 输出数据按位进行翻转
Bit2:1	RV_I: 输入数据翻转配置 00: 不翻转 01: 以字节为单位翻转 10: 以半字为单位翻转 11: 以字为单位翻转
Bit0	RST: 复位操作 0: 不复位 1: 复位整个 CRC 单元，将 CRC_INIT 寄存器值设为 CRC 初始值。 复位后，硬件自动清零。

### 10.3.3 CRC 初始值预置寄存器(CRC\_INIT)

偏移地址: 0x08

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INIT[31:16]															
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INIT[15:0]															
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit31:0	INIT[31:0]: CRC 初始值预置寄存器 写入 CRC 预置初值。 每次复位整个 CRC 单元后，硬件才会将该预置值作为 CRC 计算初始值。
---------	---

**注：** 更改 CRC 初值时，需先向 CRC\_INIT 写入初值，再使能 CRC\_CR 的 RST 位。

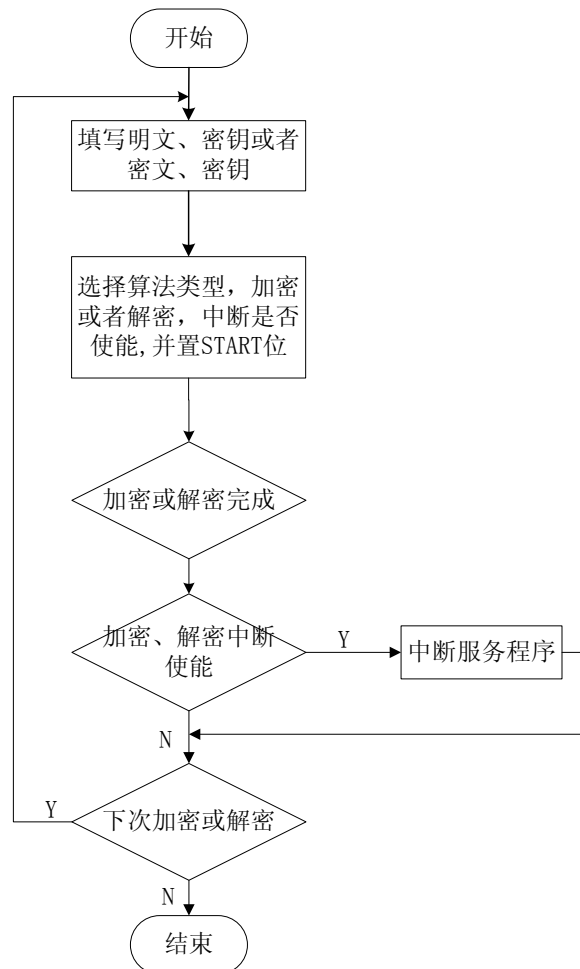
# 11. AES/3DES

## 11.1 主要特性

- 支持 AES128/3DES/DES 加解密算法
- 加密/解密是否完成可以通过查询或中断方式判断
- 中断使能可选
- 加密/解密启动设置

## 11.2 功能描述

- 在启动后，加密/解密自动完成。可通过查询或者中断的方式读取结果。
- 工作过程：填写密钥、明文或者密文后，选择算法类型，加密或者解密，是否中断使能，并置 START 位。
- 明文、密文和密钥的填写或读取方式，都采用低位在前。



加密/解密流程图



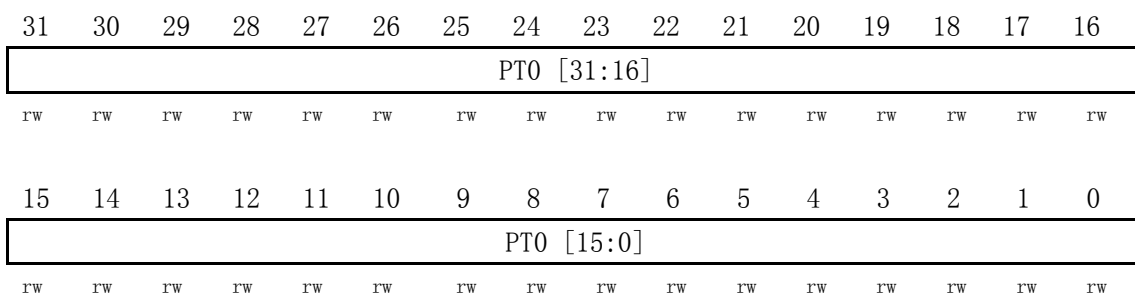
## 11.3 看门狗寄存器

AES 寄存器基地址：0x4000A800

### 11.3.1 数据输入寄存器 (AES\_PT)

偏移地址：0x00

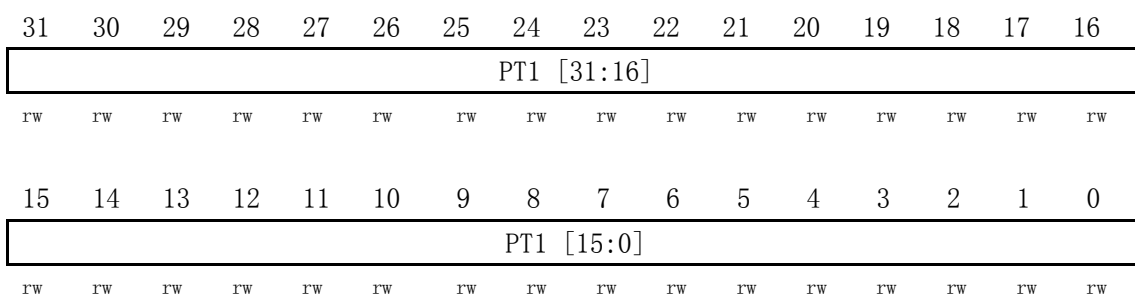
复位值：0x0000 0000



Bit31:0	PT0: AES 明文/密文输入寄存器[31:0]，低位在前。 当 SEL 为 ‘0’ 时，写入准备加密的明文。 当 SEL 为 ‘1’ 时，写入准备解密的密文。
---------	---

偏移地址：0x04

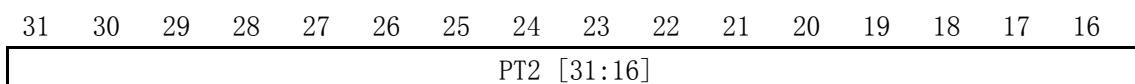
复位值：0x0000 0000



Bit31:0	PT1: AES 明文/密文输入寄存器[63:32]，低位在前。 当 SEL 为 ‘0’ 时，写入准备加密的明文。 当 SEL 为 ‘1’ 时，写入准备解密的密文。
---------	--

偏移地址：0x08

复位值：0x0000 0000



RW RW RW RW RW RW RW RW RW RW RW RW RW RW RW RW  
 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

PT2 [15:0]

RW RW RW RW RW RW RW RW RW RW RW RW RW RW RW RW

Bit31:0	PT2:AES 明文/密文输入寄存器[95:64]，低位在前。 当 SEL 为 ‘0’ 时，写入准备加密的明文。 当 SEL 为 ‘1’ 时，写入准备解密的密文。
---------	---

偏移地址：0x0C

复位值：0x0000 0000

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16

PT3 [31:16]

RW RW RW RW RW RW RW RW RW RW RW RW RW RW RW RW

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

PT3 [15:0]

RW RW RW RW RW RW RW RW RW RW RW RW RW RW RW RW

Bit31:0	PT3:AES 明文/密文输入寄存器[127:96]，低位在前。 当 SEL 为 ‘0’ 时，写入准备加密的明文。 当 SEL 为 ‘1’ 时，写入准备解密的密文。
---------	--

### 11.3.2 AES 数据输出寄存器 (AES\_CT)

偏移地址：0x10

复位值：0x0000 0000

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16

CT0 [31:16]

RW RW RW RW RW RW RW RW RW RW RW RW RW RW RW RW

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

CT0 [15:0]

RW RW RW RW RW RW RW RW RW RW RW RW RW RW RW RW

Bit31:0	CT0:AES 明文/密文输出寄存器[31:0]，低位在前。 当 SEL 为 ‘0’ 时，读取为加密后的密文。 当 SEL 为 ‘1’ 时，读取为解密后的明文。
---------	--

偏移地址: 0x14

复位值: 0x0000 0000

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16

CT1 [31:16]

rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

CT1 [15:0]

rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW

Bit31:0

CT1:AES 明文/密文输出寄存器[63:32], 低位在前。

当 SEL 为 ‘0’ 时, 读取为加密后的密文。

当 SEL 为 ‘1’ 时, 读取为解密后的明文。

偏移地址: 0x18

复位值: 0x0000 0000

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16

CT2 [31:16]

rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

CT2 [15:0]

rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW

Bit31:0

CT2:AES 明文/密文输出寄存器[95:64], 低位在前。

当 SEL 为 ‘0’ 时, 读取为加密后的密文。

当 SEL 为 ‘1’ 时, 读取为解密后的明文。

偏移地址: 0x1C

复位值: 0x0000 0000

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16

CT3 [31:16]

rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

CT3 [15:0]

rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW

Bit31:0

CT3:AES 明文/密文输出寄存器[127:96], 低位在前。

当 SEL 为 ‘0’ 时, 读取为加密后的密文。

	当 SEL 为 ‘1’ 时，读取为解密后的明文。
--	--------------------------

### 11.3.3 AES 密钥寄存器 (AES\_KEY)

偏移地址: 0x20

复位值: 0x0000 0000

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16

KEY0 [31:16]
--------------

rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

KEY0 [15:0]
-------------

rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW

Bit31:0	KEY0: AES 密钥寄存器 [31:0], 低位在前。
---------	-------------------------------

偏移地址: 0x24

复位值: 0x0000 0000

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16

KEY1 [31:16]
--------------

rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

KEY1 [15:0]
-------------

rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW

Bit31:0	KEY1: AES 密钥寄存器 [63:32], 低位在前。
---------	--------------------------------

偏移地址: 0x28

复位值: 0x0000 0000

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16

KEY2 [31:16]
--------------

rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

KEY2 [15:0]
-------------

rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW rW

Bit31:0	KEY2:AES 密钥寄存器[95:64]，低位在前。
---------	-----------------------------

偏移地址：0x2C

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CI_KEY3 [31:16]															
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CI_KEY3 [15:0]															
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

Bit31:0	KEY3:AES 密钥寄存器[127:96]，低位在前。
---------	------------------------------

### 11.3.4 AES 控制/状态寄存器 (AES\_CSR)

偏移地址：0x30

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留								DCF	ECF	DCIE	ECIE	START	EDS	AMS	
								rW	rW	rW	rW	rW	rW	rW	rW

Bit31:8	保留位，始终读为 0。
Bit7	DCF：解密完成标志。 该位由硬件置‘1’，写‘1’可清除该位。 0：解密未完成。 1：解密完成。
Bit6	ECF：加密完成标志。 该位由硬件置‘1’，写‘1’可清除该位。 0：加密未完成。 1：加密完成。
Bit5	DCIE：加密完成中断使能。 0：禁止解密完成中断。 1：使能解密完成中断。
Bit4	ECIE：加密完成中断使能。 0：禁止加密完成中断。 1：使能加密完成中断。

Bit3	START: 加密/解密启动。 该位由软件设置, 由硬件自动清除。 0: 无效。 1: 启动。
Bit2	EDS: 加密/解密模式选择。 0: 加密模式。 1: 解密模式。
Bit1:0	AMS: 加解密算法选择。 0: 无。 1: AES 算法。 2: 3DES 算法。 3: DES 算法。

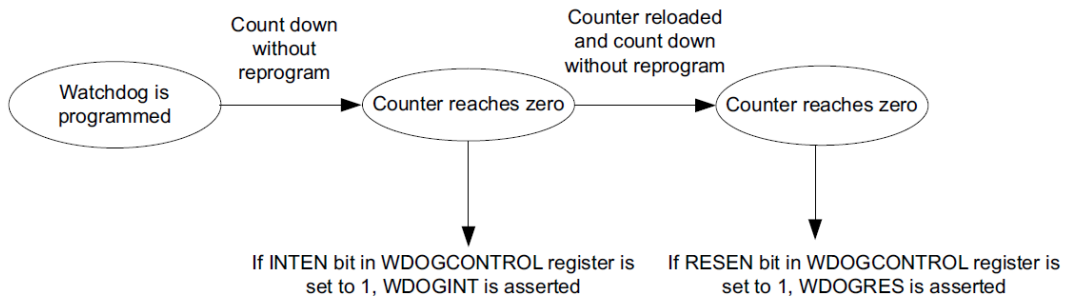
## 12. 独立看门狗(WDG)

### 12.1 主要特性

- 看门狗由专门的 RC 振荡器时钟驱动
- 由软件启动看门狗
- 看门狗启用之后不能被禁用, 直至下一次复位, 即只能够通过复位来禁止看门狗。
- 32 位递减计数器, 最长时间大于 10S。
- 具有安全锁, 防止误操作
- 看门狗复位能够影响复位状态寄存器, 即通过复位状态寄存器能够判断上电复位, 软件复位还是看门狗复位。

### 12.2 功能描述

- WDT 默认打开, 计数器计满后首先输出中断, 如果中断没有被处理, 计数器重新开始计数, 再次计满后输出复位信号, 复位输出之前 WDT 复位将不输出复位, 重新开始上述动作。
- 可以通过禁止看门狗中断使能来禁止看门狗功能。



## 12.3 看门狗寄存器

WDG 寄存器基地址：0x40004800

### 12.3.1 计数值加载寄存器 (WDG\_LDR)

偏移地址：0x00

复位值：0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LDR [31:16]															
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LDR [15:0]															
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW
Bit31:0	LDR:WDG 数据加载寄存器，加载计数器数值。														

### 12.3.2 计数值寄存器 (WDG\_CVR)

偏移地址：0x04

复位值：0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WCV [31:16]															
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WCV [15:0]															
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

Bit31:0	WCV:WDG 计数器当前值寄存器，计数器当前计数值。
---------	-----------------------------

### 12.3.3 控制寄存器(WDG\_CR)

偏移地址：0x08

复位值：0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留														WRE	WIE
														rw	rw

Bit31:2	保留。
Bit1	WRE: WDG 复位使能 0: 禁止 WDG 复位。 1: 使能 WDG 复位。
Bit0	WIE: WDG 中断使能 0: 禁止 WDG 中断。 1: 使能 WDG 中断。

### 12.3.4 中断清除寄存器(WDG\_ICR)

偏移地址：0x0C

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WIC [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WIC [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit31:0	WIC:WDG 中断清除，写任意值清除 WDG 中断，并重新加载计数值。
---------	--------------------------------------



### 12.3.5 原中断状态寄存器 (WDG\_RISR)

偏移地址: 0x0C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留															RIF
rw															
Bit31:1		保留。													
Bit0		RIF: WDG 原中断状态。													

### 12.3.6 中断状态寄存器 (WDG\_ISR)

偏移地址: 0x10

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留															WIF
rw															
Bit31:1		保留。													
Bit0		WIF: WDG 中断状态。													

### 12.3.7 安全锁寄存器 (WDG\_LOCKR)

偏移地址: 0x14

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WL [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WL [15:0]															

Bit31:0

WL: 写 0x1ACCE551 使能其他 WDG 寄存器写通道, 写其他值禁止。

## 13. 通用定时器(TIMx)

### 13.1 简介

通用定时器 TIMx (x=0, 1, 2, 3) 是由一个通过可编程预分频驱动的 32 位自动装载计数器构成。除定时功能外, 还可以测量输入信号的脉冲长度 (输入捕获功能), 可编程产生输出波形 (输出比较和 PWM 功能)。定时器之间是完全独立工作的, 可同步独立使用。

### 13.2 主要特性

- 32位自动装载计数器
- 32位可编程预分频值
- 定时器功能
  - 输入捕获
  - 输出比较
  - PWM输出
  - 定时功能
- 可配置定时器和预分频器在指定捕获事件清零。通过在输入脉冲前沿清零定时器并捕获定时器在后沿的计数值, 方便进行脉冲宽度测量
- 可在输入信号跃迁时快速捕获定时器值
- 匹配输出电平可编程设置
  - 匹配时设置低电平。
  - 匹配时设置高电平。
  - 匹配时切换电平。
  - 匹配时不执行任何操作

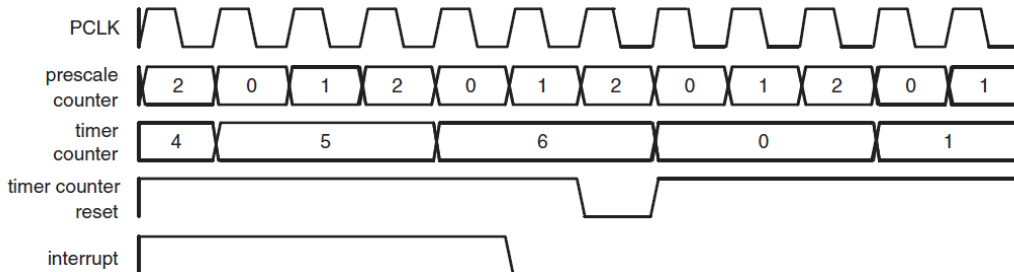
### 13.3 功能描述

#### 13.3.1 时基单元

通用定时器的主要部分是一个 32 位计数器和与其相关的自动装载寄存器, 计数器时钟由预分频器分频得到。

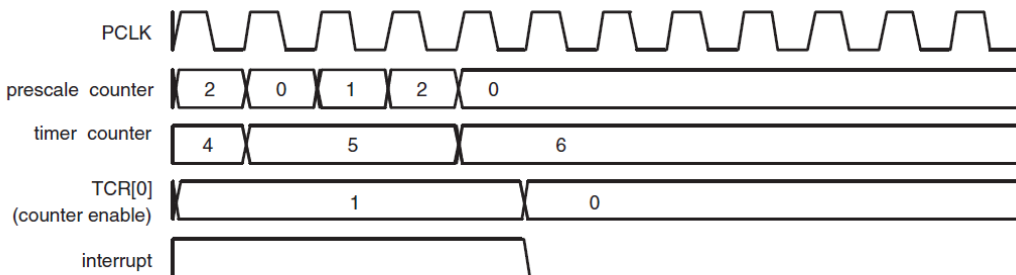
### 13.3.2 定时功能

定时器配置为在匹配时复位计数并产生中断。预分频器置为2，匹配寄存器置为6。在发生匹配的定时器周期结束时，定时器计数复位。这样就使匹配值具有完整长度的周期。在定时器到达匹配值后的下一个时钟产生指示匹配发生的中断。



**定时器周期，其中 PR = 2、MRx = 6，并且启用了在匹配时同时执行中断和复位操作**

定时器配置为在匹配时停止计数并产生中断。预分频器再次置为2，匹配寄存器置为6。在定时器达到匹配值的下一个时钟，TCR 中的定时器使能位被清零，产生指示匹配发生的中断。



**定时器周期，其中 PR = 2、MRx = 6，并且启用了在匹配时同时执行中断和停止操作**

### 13.3.3 PWM 功能

单边沿控制的 PWM 输出规则

1. 所有单边沿控制的PWM 输出在PWM 周期开始时都为低电平（定时器置为0）。
2. 每个PWM 输出在达到其匹配值时将转入高电平。如果没有发生匹配（即匹配值大于PWM 周期长度），则PWM 输出将继续保持低电平。
3. 如果将大于PWM 周期长度的匹配值写入到匹配寄存器，且PWM 信号已经为高电平，则在下一个PWM 周期开始时PWM 信号将被清零。
4. 如果匹配寄存器中包含与定时器复位值（PWM 周期长度）相同的值，则PWM 输出将继续保持低电平。
5. 如果匹配寄存器置0，则PWM 输出将在定时器为0 后变为高电平，并继续保持高电平。

例：设置占空比为50%的PWM波形，可以将TC寄存器设定为99，匹配寄存器设定为49，占空比计算公式为： $1 - (49 + 1) / (99 + 1)$

## 13.4 通用定时器寄存器

TIM0 寄存器基地址: 0x4000 0000

TIM1 寄存器基地址: 0x4000 0800

TIM2 寄存器基地址: 0x4000 1000

TIM3 寄存器基地址: 0x4000 1800

### 13.4.1 控制寄存器(TIMx\_CR)

偏移地址: 0x00

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留											TIE	ESCL	ESCA	ESCE	TE
											RW	RW	RW	RW	RW

位 31:5	保留。
位 4	TIE: 定时器中断使能 0: 禁止 TIMx 中断。 1: 使能 TIMx 中断。
位 3	ESCL: 外部信号作为捕获信号选择 0: 内部信号作为捕获信号。 1: 外部信号作为捕获信号。
位 2	ESCA: 外部信号作为时钟选择 0: 内部信号作为时钟。 1: 外部信号作为时钟。
位 1	ESCE: 外部信号作计数使能 0: 禁止外部信号作计数。 1: 使能外部信号作计数。
位 0	TE: 定时器 TIEx 使能 0: 禁止 TIEx。 1: 使能 TIMx。

### 13.4.2 计数器当前值寄存器(TIMx\_CNT)

偏移地址: 0x04

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

CNT [31:16]															
r	w	r	w	r	w	r	w	r	w	r	w	r	w	r	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT [15:0]															
r	w	r	w	r	w	r	w	r	w	r	w	r	w	r	w
位 31:0		CNT: 计数器当前值。													

### 13.4.3 计数器值设定寄存器(TIMx\_CSR)

偏移地址: 0x08

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CSR [31:16]															
r	w	r	w	r	w	r	w	r	w	r	w	r	w	r	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CSR [15:0]															
r	w	r	w	r	w	r	w	r	w	r	w	r	w	r	w
位 31:0		CSR: 计数器设定值, 也作为 PWM 周期设定值。													

### 13.4.4 中断状态寄存器(TIMx\_ISR)

偏移地址: 0x0C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留												CIS	MIS	TIS	
												r	w	r	
位 31:3		保留													
位 2		CIS: 捕获中断, 当捕获功能使能并发生捕获事件时产生中断由硬件置 ‘1’, 写 ‘1’ 清零, 写 ‘0’ 无效 0: 无中断 1: 产生中断													
位 1		MIS: 匹配中断标志, 当匹配功能使能并发生匹配事件时产生中断由硬件置 ‘1’, 写 ‘1’ 清零, 写 ‘0’ 无效 0: 无中断 1: 产生中断													
位 0		TIS: 定时器中断标志, 由硬件置 ‘1’, 写 ‘1’ 清零, 写 ‘0’ 无效													

	0: 无中断 1: 产生中断
--	-------------------

### 13.4.5 中断状态寄存器 (TIMx\_RST)

偏移地址: 0x10

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留															RST

rw

位 31:1	保留
位 0	RST: 定时器/计数器复位设置 0: 无动作 1: 复位定时器/计数器

### 13.4.6 预分频值设置寄存器 (TIMx\_PSV)

偏移地址: 0x14

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PSV [31:16]															

rw    rw    rw    rw    rw    rw    rw    rw    rw    rw    rw    rw    rw    rw    rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSV [15:0]															

rw    rw    rw    rw    rw    rw    rw    rw    rw    rw    rw    rw    rw    rw    rw

位 31:0	PSV: 预分频设定值
--------	-------------

### 13.4.7 预分频值寄存器 (TIMx\_PV)

偏移地址: 0x18

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PV [31:16]															

r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w

15   14   13   12   11   10   9   8   7   6   5   4   3   2   1   0

PV [15:0]

r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w

位 31:0	PV: 预分频计数当前值
--------	--------------

### 13.4.8 匹配控制寄存器 (TIMx\_MCR)

偏移地址: 0x1C

复位值: 0x0000 0000

31   30   29   28   27   26   25   24   23   22   21   20   19   18   17   16

保留

15   14   13   12   11   10   9   8   7   6   5   4   3   2   1   0

保留

MSC

MRC

MIE

r w

r w

r w

位 31:3	保留
位 2	MSC: 发生匹配时停止计数 0: 无动作 1: 发生匹配时停止计数
位 1	MRC: 发生匹配时, 复位计数器 0: 无动作 1: 发生匹配时, 复位计数器
位 0	MIE: 匹配中断使能, 匹配时产生中断 0: 禁止中断 1: 使能中断

### 13.4.9 匹配值设置寄存器 (TIMx\_MVS)

偏移地址: 0x20

复位值: 0x0000 0000

31   30   29   28   27   26   25   24   23   22   21   20   19   18   17   16

MVS [31:16]

r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w

15   14   13   12   11   10   9   8   7   6   5   4   3   2   1   0

MVS [15:0]

r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w

位 31:0	MVS: 设定发生匹配设定值, 同时为 PWM 占空比设定值
--------	--------------------------------

### 13.4.10 捕获控制寄存器(TIMx\_CAC)

偏移地址: 0x24

复位值: 0x0000 0000

31   30   29   28   27   26   25   24   23   22   21   20   19   18   17   16

保留															
----	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

15   14   13   12   11   10   9   8   7   6   5   4   3   2   1   0

保留													CEIE	CDE	CUC
													r w	r w	r w

位 31:3	保留
位 2	CEIE: 捕获事件产生中断 0: 禁止中断 1: 使能中断
位 1	CDE: 捕获下降沿 0: 无动作 1: 捕获下降沿
位 0	CUE: 捕获上升沿 0: 无动作 1: 捕获上升沿

### 13.4.11 捕获值寄存器(TIMx\_CV)

偏移地址: 0x28

复位值: 0x0000 0000

31   30   29   28   27   26   25   24   23   22   21   20   19   18   17   16

CV [31:16]															
------------	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

r   r   r   r   r   r   r   r   r   r   r   r   r   r   r   r

15   14   13   12   11   10   9   8   7   6   5   4   3   2   1   0

CV [15:0]															
-----------	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

r   r   r   r   r   r   r   r   r   r   r   r   r   r   r   r

位 31:0	CV: 捕获事件时, 计数器当前值
--------	-------------------



### 13.4.12 外部匹配控制寄存器 (TIMx\_EMC)

偏移地址: 0x2C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留													MOP	MO	
													rW	rW	rW

位 31:3	保留
位 2:1	MOP: 匹配操作选择: 00: 不动作 01: 输出低电平 10: 输出高电平 11: 输出切换
位 0	MO: 匹配输出 0: 不动作 1: 发生匹配时输出

### 13.4.13 计数器/定时器控制寄存器 (TIMx\_CCR)

偏移地址: 0x30

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留												RES	CC	CES	
												rW	rW	rW	rW

位 31:4	保留
位 3	RES: 计数器清 0 信号沿选择, 当位 2 设定时本位有效
位 2	CC: 捕获发生时, 计数器清 0
位 1:0	CES: 计数器时钟沿选择 00: PCLK 上升沿 01: 外部输入上升沿 10: 外部输入下降沿 11: 外部输入上升沿和下降沿同时

## 13.4.14 PWM 输出控制寄存器 (TIMx\_PCR)

偏移地址: 0x34

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PCR [31:16]															
r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PCR [15:0]															
r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w

位 31:0	PCR: PWM 输出选择 0: 输出匹配结果 1: 输出 PWM 波形
--------	--

**注:** 配置寄存器时需要先配置好设定值再使能计数器。

# 14. 串行外设接口 (SPI)

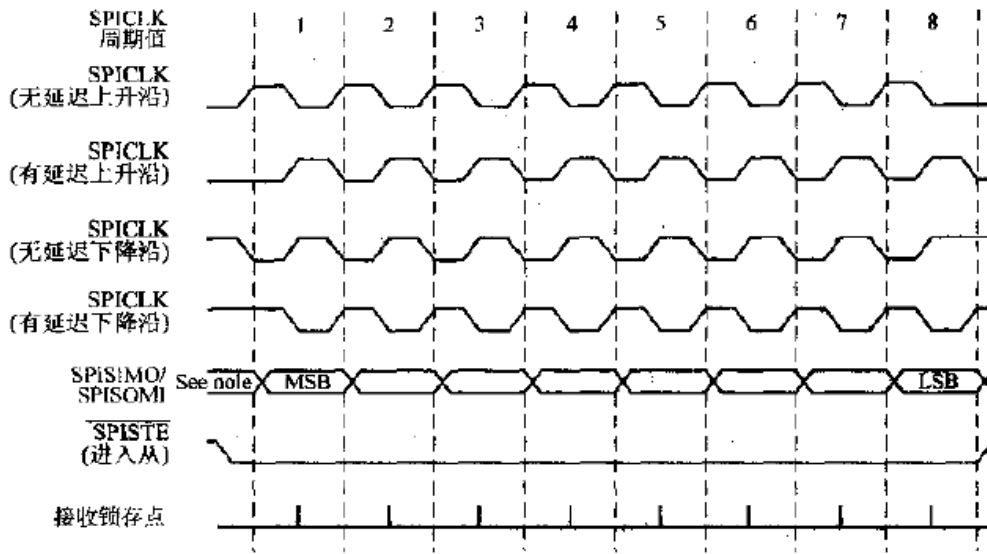
## 14.1 概述

- 工作在 SPI 主控模式下
- 波特率设定: [31:0]
- 接受和发送可以通过查询或中断方式实现
- 支持数据宽度设置, 支持 8bits, 16bits, 24bits, 32bits 传输
- 支持 MSB/LSB 传输设置
- 支持时钟相位及极性设置

## 14.2 详细功能说明

1. 主机控制具有完全的主导地址。它决定着通信的速度, 也决定着何时可以开始和结束一次通信, 从机只能被动响应主机发起的传输。
2. SPI通信是一种全双工高速的通信方式。从通信的任意一方来看, 读操作和写操作都是同步完成的。
3. SPI的传输始终是在主机控制下, 进行双向同步的数据交换。
4. 根据不同的时钟极性和相位选择可以有四种时钟方案。

### 14.3 接口传输格式



### 14.4 SPI 寄存器

SPI0 寄存器基地址: 0x4000 5000  
 SPI1 寄存器基地址: 0x4000 5800  
 SPI2 寄存器基地址: 0x4000 6000

#### 14.4.1 SPI 数据寄存器 (SPI<sub>x</sub>\_DR)

偏移地址: 0x00  
 复位值: 0x0000 0000



## 14.4.2 SPI 状态寄存器 (SPIx\_SR)

偏移地址: 0x04

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留												RO	TO	RF	TF
												rw	rw	r	r

位 31:4	保留
位 3	RO: 接收寄存器溢出标志, 硬件置 1, 软件清零, 写 1 清零, 写 0 无效 0: 接收寄存器未溢出 1: 接收寄存器溢出
位 2	TO: 发送寄存器溢出标志, 硬件置 1, 软件清零, 写 1 清零, 写 0 无效 0: 发送寄存器未溢出 1: 发送寄存器溢出
位 1	RF: 接收寄存器满标志, 只读 0: 接收寄存器未满 1: 接收寄存器满
位 0	TF: 发送寄存器满标志, 只读 0: 发送寄存器未满 1: 发送寄存器满

## 14.4.3 SPI 控制寄存器 (SPIx\_CR)

偏移地址: 0x08

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留												ROIE	TOIE	IE	EN
												rw	rw	rw	rw

位 31:4	保留
位 3	ROIE: 接收溢出中断使能 0: 禁止中断 1: 使能中断
位 2	TOIE: 发送溢出中断使能

	0: 禁止中断 1: 使能中断
位 1	IE: SPI 中断使能 0: 禁止中断 1: 使能中断
位 0	EN: SPI 使能 0: 禁止 SPI 1: 使能 SPI

#### 14.4.4 SPI 控制寄存器 (SPIx\_ISR)

偏移地址: 0x0C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留												ROIS	TOIS	保留	IS
												rW	rW	rW	rW

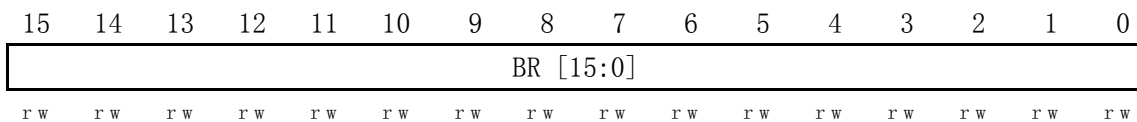
位 31:4	保留
位 3	接收溢出中断, 写 1 清零, 写 0 无效 0: 无中断 1: 产生中断
位 2	发送溢出中断, 写 1 清零, 写 0 无效 0: 无中断 1: 产生中断
位 1	保留
位 0	IS: SPI 中断标志, 写 1 清零, 写 0 无效 0: 无中断 1: 产生中断

#### 14.4.5 SPI 波特率寄存器 (SPIx\_BR)

偏移地址: 0x10

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BR [31:16]															
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

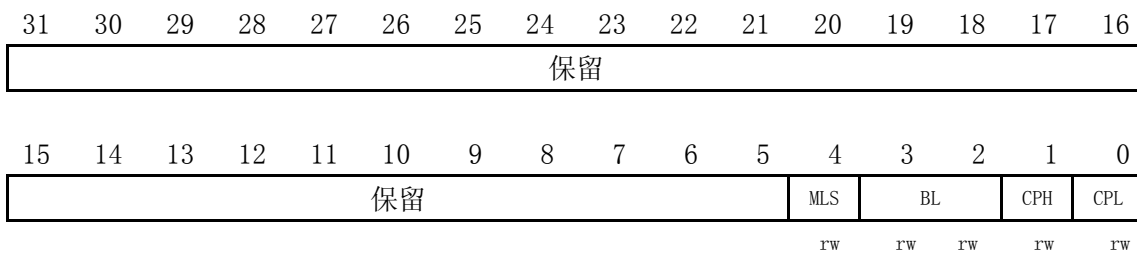


位 31:0	BR: SPI 时钟波特率
--------	---------------

### 14.4.6 SPI 线路控制寄存器 (SPIx\_LCR)

偏移地址: 0x14

复位值: 0x0000 0000



位 31:5	保留
位 4	MLS: MSB/LSB 选择 0: 选择 LSB 1: 选择 MSB
位 3:2	BL: 传输数据位宽选择 00: 8bits 01: 16bits 10: 24bits 11: 32bits
位 1	CPH: 时钟相位选择 0: 不延迟 1: 延迟
位 0	CPL: 时钟极性选择 0: 上升沿 1: 下降沿

# 15. I<sup>2</sup>C 模块

## 15.1 概述

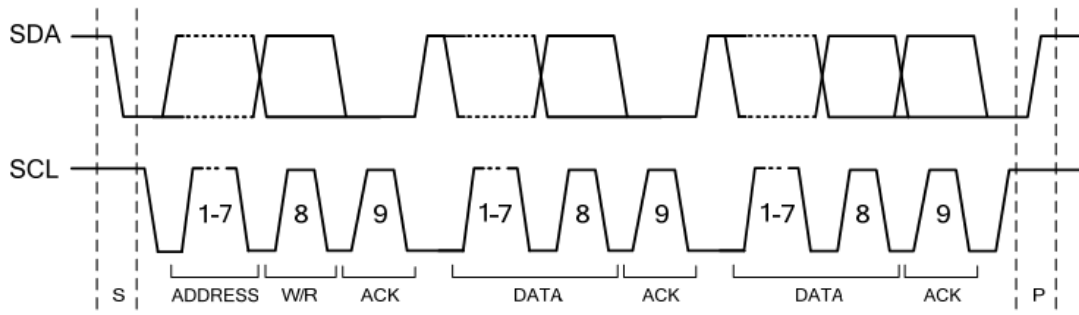
- 工作在 I<sup>2</sup>C 主控模式下
- 波特率设定：〈4, 1048575〉
- 接收和发送可以通过查询或中断方式实现
- 支持读写数据等待，可以设置超时时间
- 支持单字节和多字节读取

## 15.2 功能描述

### I<sup>2</sup>C 协议

通常标准 I<sup>2</sup>C 传输协议包含四个部分：

- 1) 起始信号或重复起始信号的产生
- 2) 从机地址和 R/W 位传输
- 3) 数据传输
- 4) 停止信号的产生



### 从机地址传输

起始信号后传输的第一个字节是从机地址，从机地址的头 7 位是呼叫地址，紧跟 7 位地址后的是 RW 位。RW 位通知从机数据传输方向。系统当中不会有两台从机有相同的地址。只有地址匹配的从机才会在 SCL 的第 9 个时钟周期拉低 SDA 作为应答信号来响应主机。

### 数据传输

当从机寻址成功完成，就可以根据主机发送的 R/W 位所决定的方向，开始一字节一字节的数据传输，每一个传输的字节会在第九个 SCL 时钟周期跟随一个应答位，如果从机上产生无应答信号 (NACK)，主机产生一个停止信号来中止本次数据传输。

如果主机作为接收设备，没有应答 (NACK) 从机，则从机释放 SDA 线，以便于主机产生一个停止信号。

用户在使用连续读数时需要在读倒数第二个数时给出不应答标志，读最后一个数时给出

结束标志或重新开始传输。

## 15.3 I<sup>2</sup>C 寄存器

IIC0 寄存器基地址: 0x4000 6800

IIC1 寄存器基地址: 0x4000 7000

### 15.3.1 I<sup>2</sup>C 数据寄存器 (I<sup>2</sup>Cx\_DR)

偏移地址: 0x00

复位值: 0x0000 0000

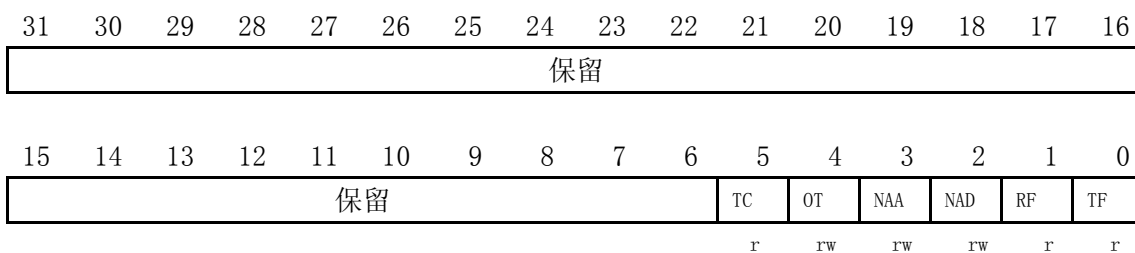


位 31:8	保留
位 7:0	DR: 8 位 I <sup>2</sup> C 串口传输数据

### 15.3.2 I<sup>2</sup>C 状态寄存器 (I<sup>2</sup>Cx\_SR)

偏移地址: 0x04

复位值: 0x0000 0020



位 31:7	保留
位 5	TC: 发送完成标志, 当 I <sup>2</sup> C 数据全部发送完成时该标志置 1, 该标志只读, 写数据后该位自动清零 0: 发送未完成 1: 发送完成
位 4	OT: 等待 CPU 读写数超时, 硬件置一, 软件清零 0: 未超时



	1: 超时
位 3	NAA: 从机 NACK 标志 1: 匹配从机地址时, 从机没有 ACK。发送一个停止状态, I <sup>2</sup> C 停止 硬件置一, 软件清零 0: 应答状态正常 1: 没有收到匹配地址应答
位 2	NAD: 从机 NACK 标志 1: 主机发数据时, 从机没有 ACK。发送一个停止状态, I <sup>2</sup> C 停止 硬件置一, 软件清零 0: 应答状态正常 1: 没有收到发送数据应答
位 1	RF: 接收数据标志, 当该标志为 1 时表示 I <sup>2</sup> C 接收到一个字节, CPU 读数据该标志自动清除。该位为只读位 0: 接收数据寄存器空 1: 接收数据寄存器满
位 0	TF: 发送数据寄存器满标志, 当该位为 0 时 CPU 可以写发送数据。该位为只读位 0: 发送数据寄存器空 1: 发送数据寄存器满

### 15.3.3 I<sup>2</sup>C 控制寄存器 (I<sup>2</sup>Cx\_CR)

偏移地址: 0x08

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留								RNA	保留	STAR	SRT	保留	TE	IE	EN
								rw		rw	rw		rw	rw	rw

位 31:8	保留
位 7	RNA: I <sup>2</sup> C 读阶段, 停止接收控制位, 0: ACK 应答 (SDA 为低) 1: 没有 ACK (SDA 为高) 软件置一, 硬件清零
位 6	保留
位 5	STAR: START 控制位 0: 不动作 1: IDEL 后, 软件置一, 系统准备传数 软件置一, 硬件清零
位 4	SRT: 软件复位控制位

	0: 不复位 1: 整个系统复位 软件置一, 硬件清零
位 3	保留
位 2	TE: 定时器使能 0: 禁止 I <sup>2</sup> C 定时器 1: 使能 I <sup>2</sup> C 定时器
位 1	IE: I <sup>2</sup> C 中断使能 0: 禁止 I <sup>2</sup> C 中断功能 1: 使能 I <sup>2</sup> C 中断功能
位 0	EN: I <sup>2</sup> C 使能 0: 禁止 I <sup>2</sup> C 功能 1: 使能 I <sup>2</sup> C 功能

### 15.3.4 I<sup>2</sup>C 波特率寄存器 (I<sup>2</sup>Cx\_BD)

偏移地址: 0x0C

复位值: 0x0000 0200

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留												BD [19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BD [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位 31:20	保留
位 19:0	BD: I <sup>2</sup> C 时钟波特率, 最小为 4

### 15.3.5 I<sup>2</sup>C 超时设置寄存器 (I<sup>2</sup>Cx\_TR)

偏移地址: 0x10

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TR [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TR [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位 31:0	TR: 等待超时时间设定, 值为设定值乘以波特率除 4
--------	-----------------------------

### 15.3.6 I<sup>2</sup>C 停止控制寄存器 (I<sup>2</sup>Cx\_SCR)

偏移地址: 0x14

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留															SR

rw

位 31:1	保留
位 0	SR: STOP 控制位, 软件置一, 硬件清零 0: 无动作 1: 在 ACK 状态后产生 STOP 状态, 系统停止

### 15.3.7 I<sup>2</sup>C 中断状态寄存器 (I<sup>2</sup>Cx\_ISR)

偏移地址: 0x18

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留 [15:6]										TCIS	TOIS	NAIS	NDIS	RIS	TIS

rw    rw    rw    rw    rw    rw

位 31:6	保留
位 5	TCIS: 发送完成中断标志, 当 I <sup>2</sup> C 数据全部发送完成时, 写数据或写 1 后该位自动清零 0: 发送未完成 1: 发送完成
位 4	TOIS: 等待 CPU 写数超时中断标志, 写 1 清零, 写 0 无效 0: 未超时 1: 超时
位 3	NAIS: 从机 NACK 中断标志, 匹配从机地址时, 从机没有 ACK。发送一个停止状态, I <sup>2</sup> C 停止, 进中断。硬件置一, 软件清零, 写 1 清零, 写 0 无效 0: 正常 1: 匹配从机地址时, 从机没有 ACK

位 2	NDIS: 从机 NACK 中断标志, 主机发数据时, 从机没有 ACK 发送一个停止状态, I <sup>2</sup> C 停止, 进中断。硬件置一, 软件清零, 写 1 清零, 写 0 无效 0: 正常 1: 主机发数据时, 从机没有 ACK
位 1	RIS: 接收数据非空中断标志, 当该标志为 1 时表示 I <sup>2</sup> C 接收到一个字节, 读数据或者对该位写 1 清除该标志。 0: 空 1: 接收数据非空
位 0	TIS: 发送数据寄存器空中断标志, 当该位为 1 时表示寄存器中的数据被硬件转移到发送移位寄存器, CPU 可以通过写发送数据或对该位写 1 来清除该标志。 0: 非空 1: 发送数据寄存器空

**注:** 配置寄存器需要先使能 I<sup>2</sup>C, 再填写数据寄存器。

## 16. DMA

### 16.1 概述

单个 AHB-Lite 主机使用 32 位地址和数据总线传输数据。

- 支持按字传输数据。
- 13 个 DMA 通道。
- 每个通道都带有握手信号。
- 每个优先级仲裁使用一个由 DMA 通道号决定的固定优先级。
- 支持存储器到存储器、存储器到外设、外设到存储器的传输。
- 单个 DMA 周期的传输数可在 1 到 4095 之间设置。
- AHB 上出现错误情况时可以通过 dma\_err 信号进行指示。

DMA 控制器包含 APB 寄存器接口、到 AHB 接口以及 DMA 控制模块

- 使用硬件或软件请求 DMA 传输。不支持猝发传输。
- 目标数据宽度与源数据宽度相等。

#### 1. APB 模块:

APB 模块用于配置控制传输的寄存器。

#### 2. DMA 控制模块:

- 仲裁接收到的请求。
- 指示哪个通道活跃。
- 通道完成时给予提示。
- AHB-Lite 接口上发生错误时进行指示。

- 允许较慢的外设中断 DMA 周期。

### 3. AHB 模块:

AHB 模块用于传输数据

## 16.2 功能描述

一个典型的 DMA 传输过程如下:

- (1) I/O 设备向 DMAC (DMA-Controller, DMA 控制器) 发出 DMA 请求。
- (2) DMAC 在接到 I/O 设备发出的 DMA 请求后, 向 CPU 发出总线请求, 请求 CPU 脱离对总线的控制, 而由 DMAC 接管系统总线。
- (3) CPU 在执行完当前指令的当前总线周期后, 向 DMAC 发出总线响应信号。
- (4) CPU 脱离对系统总线的控制, 处于等待状态 (但一直监视 DMAC), 由 DMAC 接管对系统总线的控制。
- (5) DMAC 向 I/O 设备发出 DMA 应答信号。
- (6) DMAC 在存储器与 I/O 设备之间进行数据传输, 通过地址总线送出地址, 通过数据总线送出数据, 通过控制总线向存储器和 I/O 设备发出读, 写信号, 从而完成一个字节的传输, 并修改地址指针继续传输。
- (7) 当设定的字节数据传输完毕后 (由 DMAC 自动计数), DMAC 将总线请求信号变成无效, 同时脱离对总线的控制, CPU 检测到总线请求信号变成无效后, 也将总线响应信号变成无效, CPU 回复对系统总线的控制, 继续执行被 DMAC 中断的当前指令的当前总线周期。

使用说明:

- 1 打开写保护
- 2 配置通道数据结构
- 3 配置通道请求寄存器

### DMA 优先级

通道编号 0 具有最高优先级, 随着通道编号增大, 通道优先级递减。

通道编号	优先级设置	仲裁优先级降序
0	默认	
1	默认	最高
...	...	...
12	默认	最低

### 传输通道

外设	DMA 通道号	是否软件请求	是否硬件请求
UART0 RX	0	否	是
UART1 RX	1	否	是
UART2 RX	2	否	是
SPI0 RX	3	否	是
SPI1 RX	4	否	是

SSPI RX	5	否	是
UART0 TX	6	是	是
UART1 TX	7	是	是
UART2 TX	8	是	是
SPI0 TX	9	是	是
SPI1 TX	10	是	是
SSPI TX	11	是	是
存储器到存储器	12	是	否

## 16.3 DMA 寄存器

DMA 寄存器基地址：0x4000 8800

### 16.3.1 DMA 写保护寄存器 (DMA\_LOCK)

偏移地址：0x00

复位值：0x0000 0000

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16

LOCK [31:16]
--------------

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

LOCK [15:0]
-------------

位 31:13	LOCK: DMA 写保护寄存器, 写入 0x000000FF 解除写保护, 写入其它值启动保护
---------	--

### 16.3.2 DMA 通道软件请求寄存器 (DMA\_SRR)

偏移地址：0x04

复位值：0x0000 0000

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16

保留
----

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

保留	SRR[12:0]
----	-----------

w w w w w w w w w w w w w w w

位 31:13	保留
---------	----

位 12:0	SRR: 通道软件请求寄存器, 该寄存器为只写寄存器, 允许通道 c(0-12) 产生软件 DMA 请求。 c=0: 通道 c 不创建 DMA 请求 c=1: 通道 c 创建 DMA 请求
--------	--

### 16.3.3 DMA 通道使能寄存器 (DMA\_ER)

偏移地址: 0x08

复位值: 0x0000 0000

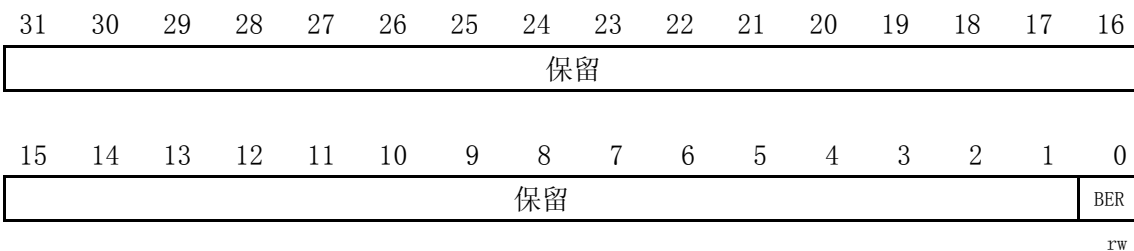


位 31:13	保留
位 12:0	ER: 通道使能设置寄存器, 在通道软件请求寄存器前一个配置, 该寄存器为读/写寄存器, 使能 DMA 通道 c(0-12)。对该寄存器的读操作将返回通道的使能状态。 c=0: 通道 c 被禁能 c=1: 通道 c 被使能

### 16.3.4 DMA 总线错误寄存器 (DMA\_BER)

偏移地址: 0x0C

复位值: 0x0000 0000



位 31:1	保留
位 0	BER: 总线错误清除寄存器, 该寄存器为读/写寄存器, 返回总线状态, 软件写 1 清除错误状态。 0: 总线未发生错误 1: 总线发生错误

### 16.3.5 DMA 中断状态寄存器 (DMA\_IS)

偏移地址: 0x10

复位值: 0x0000 0000



位 31:13	保留
位 12:0	ER: 通道 DMA 中断状态寄存器 该寄存器为读/写寄存器, 显示每个 DMA 通道 c(0-12)的 DMA 完成中断状态。 写“1”将清除状态位。 c=0:通道 c 未中断 c=1:通道 c 产生 DMA 完成中断

### 16.3.6 DMA 错误中断使能寄存器 (DMA\_EIE)

偏移地址: 0x14

复位值: 0x0000 0000



位 31:1	保留
位 0	EIE: DMA 错误中断使能寄存器, 该寄存器为读/写寄存器, 使能 DMA 错误中断。 0:禁能 DMA 错误中断 1:使能 DMA 错误中断

### 16.3.7 DMA 完成中断使能寄存器 (DMA\_CIE)

偏移地址: 0x18

复位值: 0x0000 0000



31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留			CIE[12:0]												
			rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

位 31:13	保留
位 12:0	CIE: 通道 DMA 中断使能寄存器, 该寄存器为读/写寄存器, 允许 DMA 通道 c(0-12)在完成 DMA 传输后创建中断。 c=0: 禁能通道 c 的 DMA 完成中断 c=1: 使能通道 c 的 DMA 完成中断

### 16.3.8 DMA 通道模式选择寄存器 (DMA\_CMSR)

偏移地址: 0x1C

复位值: 0x0021 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留								DA[23:20]				保留		DU[17:16]	
								rW	rW	rW	rW			rW	rW

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留			CD[12:0]												
			rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

位 31:24	保留
位 23:20	DA: 将这些位置位以控制在控制器重新仲裁之前可发生的 DMA 传输次数 (根据源数据大小确定, 每字节/半字/字为一次), 只适用于内存到内存的传输。 可能的仲裁率设置为: 0000: 每次完成 DMA 传输后仲裁 0001: 完成 2 次 DMA 传输后仲裁 0010: 完成 4 次 DMA 传输后仲裁 0011: 完成 8 次 DMA 传输后仲裁 0100: 完成 16 次 DMA 传输后仲裁 0101: 完成 32 次 DMA 传输后仲裁 0110: 完成 64 次 DMA 传输后仲裁 0111: 完成 128 次 DMA 传输后仲裁 1000: 完成 256 次 DMA 传输后仲裁 1001: 完成 512 次 DMA 传输后仲裁 1010: 完成 1024 次 DMA 传输后仲裁
位 19:18	保留
位 17:16	DU: dma 与 cpu 的占空比: 00: delay8 个周期

	01: delay16 个周期 10: delay32 个周期 11: delay64 个周期
位 15:13	保留
位 12:0	CD: 通道 0-12DMA 方向 1: 内存到外设 0: 外设到内存 (内存到内存设置为 0)

### 16.3.9 DMA 源数据起始地址寄存器 (DMA\_SDSA)

偏移地址:  $0x20+12x$  (通道编号 0..12)

复位值:  $0x0000\ 0000$

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SDSA [31:16]															
r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SDSA [15:0]															
r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w

位 31:0	SDSA: 源数据起始地址, 每个通道都包含源数据起始地址
--------	-------------------------------

### 16.3.10 DMA 目的数据起始地址寄存器 (DMA\_DDSA)

偏移地址:  $0x24+12x$  (通道编号 0..12)

复位值:  $0x0000\ 0000$

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DDSA [31:16]															
r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DDSA [15:0]															
r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w

位 31:0	DDSA: 目的数据起始地址, 每个通道都包含目的数据起始地址
--------	---------------------------------

### 16.3.11 DMA 数据长度配置寄存器 (DMA\_LCR)

偏移地址:  $0x28+12x$  (通道编号 0..12)

复位值:  $0x0000\ 0000$

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留												DI[19:18]	DS[16:16]		
												rW	rW	rW	rW
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SI[15:14]	SS[13:12]	TL[11:0]													
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

位 31:20	保留
位 19:18	DI: 目标地址增量, 地址增量取决于源数据宽度, 具体如下: 源数据宽度=字节 00=字节 01=半字 10=字 11=无增量 源数据宽度=半字 00=保留 01=半字 10=字 11=无增量 源数据宽度=字 00=保留 01=保留 10=字 11=无增量
位 17:16	DS: 目标数据大小。必须设为与源数据大小相同的值
位 15:14	SI: 源地址增量, 地址增量取决于源数据宽度, 具体如下: 源数据宽度=字节 00=字节 01=半字 10=字 11=无增量 源数据宽度=半字 00=保留 01=半字 10=字 11=无增量 源数据宽度=字 00=保留 01=保留 10=字 11=无增量
位 13:12	SS: 匹配源数据大小:

	00=字节 01=半字 10=字 11=保留
位 11:0	TL:需要传输的字的长度

注：载波通道只支持按字传输

## 17. SSPI 从机模块

### 17.1 概述

- 工作在 SSPI 从机模式下
- 接收和发送可以通过查询或中断方式实现
- 支持 8bits 传输
- 支持 MSB/LSB 传输设置

### 17.2 功能说明

- ① 主机控制具有完全的主导地址。它决定着通信的速度，也决定着何时可以开始和结束一次通信，从机只能被动响应主机发起的传输。
- ② SPI通信是一种全双工高速的通信方式。从通信的任意一方来看，读操作和写操作都是同步完成的。
- ③ SPI的传输始终是在主机控制下，进行双向同步的数据交换。

### 17.3 SSPI 寄存器

SSPI 寄存器基地址：0x4000 A000

#### 17.3.1 SSPI 数据寄存器 (SSPI<sub>x</sub>\_DR)

偏移地址：0x00

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DR [31:16]															

r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w

15   14   13   12   11   10   9   8   7   6   5   4   3   2   1   0

DR [15:0]
-----------

r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w   r w

位 31:0	DR: SSPI 传输数据寄存器
--------	------------------

### 17.3.2 SSPI 控制寄存器 (SSPIx\_CR)

偏移地址: 0x04

复位值: 0x0000 0000

31   30   29   28   27   26   25   24   23   22   21   20   19   18   17   16

保留
----

15   14   13   12   11   10   9   8   7   6   5   4   3   2   1   0

保留	DRE	DTE	TCIE	RIE	TIE
	r w	r w	r w	r w	r w

位 31:5	保留
位 4	DRE: DMA 接收请求使能 0: 禁止接收请求 1: 使能接收请求
位 3	DTE: DMA 发送请求使能 0: 禁止发送请求 1: 使能发送请求
位 2	TCIE: SPI 完成中断使能, 当该位置高时, 表示一次 SPI 传输结束, CS 拉高 0: 禁止完成中断 1: 使能完成中断
位 1	RIE: SPI 接收中断使能, 当该位置高时, SPI 从机接收完一组数据后产生中断 0: 禁止接收中断 1: 使能接收中断
位 0	TIE: SPI 发送中断使能, 当该位置高时, SPI 从机发送完一组数据后产生中断 0: 禁止发送中断 1: 使能发送中断

### 17.3.3 SSPI 状态寄存器 (SSPIx\_SR)

偏移地址: 0x08

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留													TCIS	RIS	TIS
													rw	rw	rw

位 31:3	保留
位 2	TCIE: SPI 完成标志, 当 CS 输入上升沿时, 该标志置高, 写 1 清零 0: 无中断 1: SPI 完成标志
位 1	RIE: SPI 接收完成标志, 当 SPI 从机接收完一组数据后, 该标志置高, 读数据或写 1 清零 0: 无中断 1: SPI 接收完成
位 0	TIE: SPI 发送完成标志, 当 SPI 从机发送完一组数据后, 该标志置高, 写入数据或写 1 清零 0: 无中断 1: SPI 发送完成

### 17.3.4 SSPI 线路控制寄存器 (SSPIx\_LCR)

偏移地址: 0x0C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留													MLS	EN	
													rw	rw	

位 31:2	保留
位 1	MLS: MSB/LSB 有效位选择 0: LSB 1: MSB
位 0	EN: SPI 从机工作使能 0: 禁止 SPI 从机

## 18. 宽带载波通信模块(BPL)

### 18.1 主要特性

- 每帧数据长度为 136Byte 到 2080Byte
- 物理层峰值通信速率高达 23Mbps
- 调制方式为 OFDM
- 载波信号幅度可调，最大幅度为 1V（峰峰值）

### 18.2 功能概述

#### 1. 载波工作描述

发送过程：用户根据需要设置各配置寄存器，将要发送的帧控制信息 104bits 按照选定的协议标准填写在帧控制信息寄存器中，将要传输的载荷数据放在相应的 RAM 空间，将物理状态寄存器配置为发送状态即使能发送，硬件自动识别帧控制信息中的调制方式并按照用户配置的各参数信息将帧控制及发送数据经过一系列编码调制等过程经由 DAC 发送。

接收过程：用户将物理状态寄存器配置为接收状态后，载波模块自动使能接收，将 ADC 数据进行分析判断，当检测到有效帧后给出标志信息。将帧控制信息进行解调解码，得到帧控制信息并给出相应标志，用户可以在相应寄存器读出帧控制信息。硬件解析帧控制信息中包含的解调方式，按该方式进行数据的解调解码，给出完成标志并将有效的数据存储于指定的 RAM 空间。

#### 2. 载波发送信号幅度计算

载波信号幅度可调，最大幅度为 1V（峰峰值）。共有 200 档。

信号幅度(V)= 设置档位/200x1V

**注：**由于宽带载波模块使用比较复杂，晓程公司会提供相应的驱动程序和技术支持，手册中不再提供寄存器及其使用方式。

# 19. 系统控制模块

## 19.1 简介

系统控制模块包括复位检测，外部掉电、过压检测，系统时钟控制和程序加载等功能。

## 19.2 功能概述

### 功能概述

1. 看门狗，软复位和手动复位状态查询
2. 程序校验中断控制和错误位置查询
3. 程序加载使能
4. 程序加载错误标志查询和位置查询
5. CPU 时钟分频设置
6. 过压和欠压中断
7. 过压检测阈值设置
8. 外部晶振微调控制
9. 外部晶振时钟输出分频

## 19.3 系统控制寄存器

### 19.3.1 SYS 复位状态寄存器 (SYS\_RSTS)

地址：0x4001F010

复位值：0x0000 ----

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留												ER	LS	WR	SR
												r w	r w	r w	r w

位 31:4	保留
位 3	ER: 外部引脚复位标志，外部引脚复位自动置 1，软件清 0
位 2	LS: 锁定复位标志，锁定复位自动置 1，软件清 0
位 1	WR: 看门狗复位标志，看门狗复位自动置 1，软件清 0
位 0	SR: 软复位标志，软复位自动置 1，软件清 0



### 19.3.2 SYS 锁定寄存器(SYS\_ULOCK)

地址: 0x4001F800

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留															ULK

r w

位 31:1	保留
位 0	ULK: 读出 0: 寄存器锁定, 不能进行写操作 ; 读出 1: 解锁 默认为锁定状态, 向锁定寄存器写入 0x05500660 解锁

### 19.3.3 SYS 程序存储器校验控制寄存器(SYS\_PMCC)

地址: 0x4001F804

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留														IE	IS

r w      r w

位 31:2	保留
位 1	IE: 程序存储器校验中断使能, 高有效, 默认为 0
位 0	IS: 程序存储器校验中断标志位, 写 1 清 0

### 19.3.4 SYS 校验错误地址寄存器(SYS\_MFCA)

地址: 0x4001F808

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MFC[15:0]															

r r r r r r r r r r r r r r r r

位 31:16	保留
位 15:0	MFCA: 程序存储器校验错误位置

### 19.3.5 SYS FLASH 程序加载控制寄存器 (SYS\_FLC)

地址: 0x4001F80C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	保留										
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	保留										FLC

rw

位 31:1	保留
位 0	FLC: 1: 复位后重新加载程序 0: 复位后不重新加载程序

### 19.3.6 SYS FLASH 程序加载错误寄存器 (SYS\_FLE)

地址: 0x4001F810

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	保留										
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	保留										FLE

r

位 31:1	保留
位 0	FLE: FLASH 程序加载错误标志位 1: 加载时 CRC 校验错误

### 19.3.7 SYS FLASH 程序加载错误地址寄存器 (SYS\_FLEA)

地址: 0x4001F814

复位值: 0x0000 0000

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16

保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FLEA [15:4]															
r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w	r w

位 31:16	保留
位 15:0	FLE: FLASH 加载校验错误的位置

### 19.3.8 SYS CPU 时钟分频寄存器 (SYS\_CCD)

地址: 0x4001F818

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留														CCD	
															r w

位 31:1	保留
位 0	CCD: 0: CPU 时钟为 37.5Mhz 1: CPU 时钟为 75Mhz

### 19.3.9 SYS 电源中断控制寄存器 (SYS\_PIE)

地址: 0x4001F81C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留												OVIE	OVDE	PDIE	
													r w	r w	r w

位 31:3	保留
位 2	OVIE: 过压检测中断使能, 高有效, 默认为 0
位 1	OVDE: 过压检测使能, 高有效, 默认为 0
位 0	PDIE: 掉电检测中断使能, 高有效, 默认为 0

### 19.3.10 SYS 电源中断状态寄存器(SYS\_PIS)

地址: 0x4001F820

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留 [15:4]												OVIS	OVS	PDIS	PDS
												rw	rw	rw	rw

位 31:4	保留
位 3	OVIS: 过压检测中断标志位, 写 1 清零
位 2	OVS: 过压标志位, 1: 12V 电压升高超过设定阈值
位 1	PDIS: 掉电检测中断标志位, 写 1 清零
位 0	PDS: 掉电标志位, 1: 12V 电压降至设定阈值

### 19.3.11 SYS 过压检测阈值设置寄存器(SYS\_OCVS)

地址: 0x4001F824

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留													OCVS		
													rw	rw	rw

位 31:3	保留
位 2:0	OCDS: 过压比较器阈值控制信号 000~111 : 0.3V~1V 0.1V /step 默认设置: 000 (0.3V)

### 19.3.12 SYS 外部晶振微调控制寄存器 (SYS\_XTC)

地址: 0x4001F828

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留										XTTE	XTT				
										rw	rw	rw	rw	rw	rw
位 31:6	保留														
位 5	XTTE: 外部晶振微调使能														
位 4:0	XTT: XTAL BUF 补偿电容修调值控制位 修调范围: 0fF - 1550fF (未考虑寄生效应) 50fF /step 01111: 750fF														

### 19.3.13 SYS 外部晶振输出分频寄存器 (SYS\_XTD)

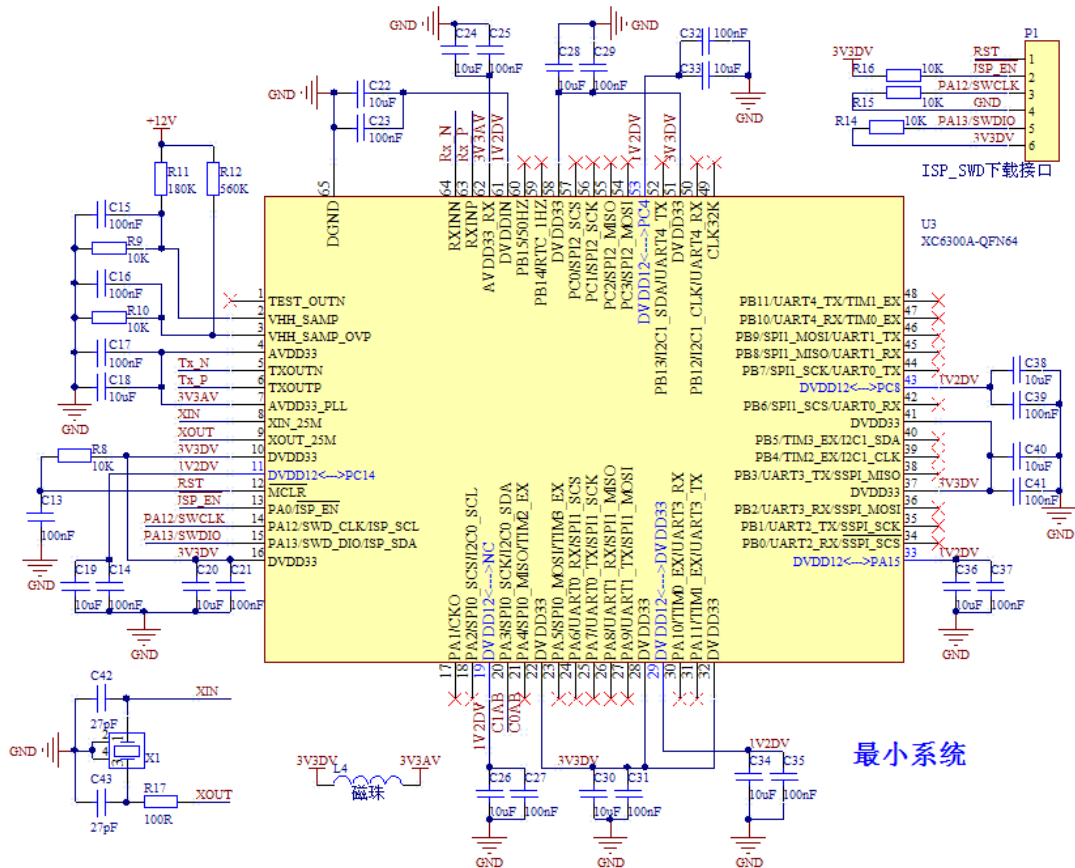
地址: 0x4001F82C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留													XTD		
													rw	rw	rw
位 31:3	保留														
位 2:0	XTD: XTAL_25M 输出分频器 000: 不分频 001: 2分频 010: 4分频 011: 8分频 100: 16分频 101: 32分频														

## 20. 参考电路

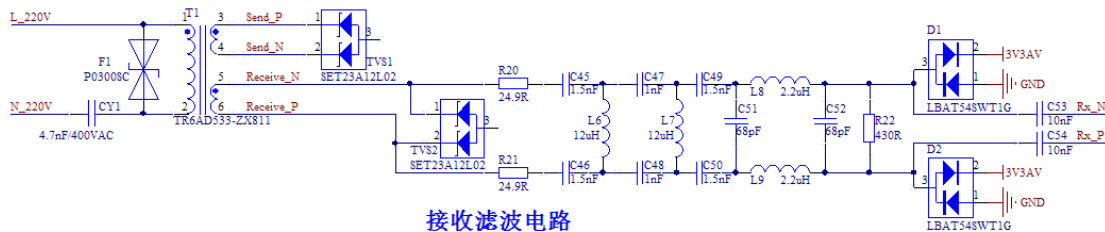
### 20.1 最小系统参考



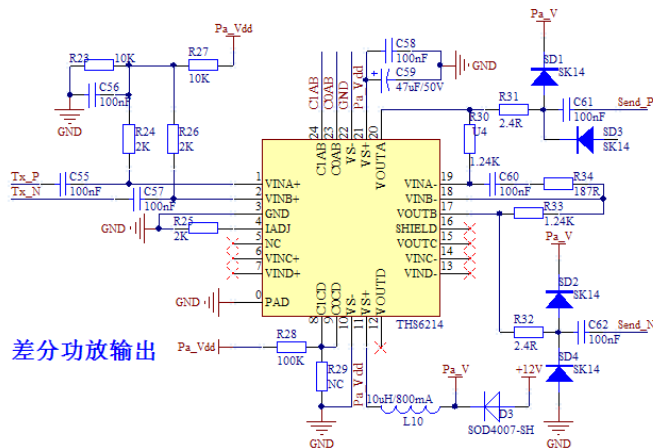
最小系统电路网络说明:

3V3DV	3.3V 数字电源输入，与外部 3.3V 直流电源相连
3V3AV	3.3V 模拟电源输入，通过磁珠与 3V3DV 相连
1V2DV	1.2V 数字电源输入，与外部 1.2V 直流电源相连
+12V	12V 电源，分压后与 VHH_SAMP 和 VHH_SAMP_OVP 引脚相连。这两个引脚可分别检测 12V 电压的欠压和过压。
GND	GND
XIN	25M 晶振输入端，与 25M 无源晶振 X1 相连
XOUT	25M 晶振输出端，与 25M 无源晶振 X1 相连
Tx_N	载波通信差分信号反向输出端
Tx_P	载波通信差分信号同向输出端
Rx_N	载波通信差分信号反向输入端
Rx_P	载波通信差分信号同向输入端
COAB	PA 芯片 THS6214 控制引脚
C1AB	PA 芯片 THS6214 控制引脚

## 20.2 载波电路参考



接收滤波电路



差分功放输出

载波电路网络说明:

Tx_N	载波通信差分信号反向输出端
Tx_P	载波通信差分信号同向输出端
Rx_N	载波通信差分信号反向输入端
Rx_P	载波通信差分信号同向输入端
COAB	载波发射用 PA 控制引脚
C1AB	载波发射用 PA 控制引脚
+12V	载波发射用 PA 芯片 THS6214 的供电电源
L_220V	220V 交流市电端口
N_220V	220V 交流市电端口