

## 1 简介

本参考手册是对 SC32F15G 系列数据手册的补充，提供了应用（特别是软件开发）所需的信息，有关特定 SC32F15G 器件的功能集、订购信息以及机械和电气特征的信息，请参见其相应的数据手册。

## 目录

1	简介.....	1
	目录.....	2
2	文档约定.....	10
2.1	名词解释.....	10
2.2	外设可用性.....	10
3	资源框图.....	11
4	上电、复位和时钟控制（RCC）.....	12
4.1	上电过程.....	12
4.1.1	复位阶段.....	12
4.1.2	调入信息阶段.....	12
4.1.3	正常操作阶段.....	12
4.2	复位.....	12
4.2.1	硬件复位.....	13
4.2.2	软件复位.....	14
4.2.3	复位后的启动区域.....	14
4.2.4	复位初始状态.....	14
4.3	时钟.....	14
4.3.1	系统时钟源.....	14
4.3.2	内建高频 72MHz 振荡器（HIRC）.....	15
4.3.3	内建低频 32kHz 振荡器（LIRC）.....	15
4.3.4	内置低频振荡电路，可外接 32.768kHz 低频震荡器（LXT）.....	15
4.4	寄存器访问时钟和外设时钟源.....	16
4.4.1	寄存器访问时钟.....	16
4.4.2	外设时钟源.....	16
4.4.3	总线.....	16
4.4.4	外设时钟.....	18
4.4.5	外设时钟框图.....	18
4.4.6	外设时钟使能寄存器.....	19
4.4.7	低功耗模式时钟.....	19
4.5	RCC 寄存器.....	19
4.5.1	RCC 相关寄存器表.....	19
4.5.2	RCC 寄存器映射.....	30
5	中断.....	32
5.1	外部中断 INT0~15.....	32
5.2	中断与事件.....	32
5.3	中断向量表.....	33

5.4	外部中断寄存器 .....	34
5.4.1	外部中断相关寄存器表 .....	34
5.4.2	外部中断寄存器映射 .....	38
6	存储 .....	39
6.1	概述 .....	39
6.2	存储框图 .....	39
6.3	特性 .....	40
6.4	APROM (主存储区) .....	40
6.5	2 Kbytes 类 EEPROM (用户存储区) .....	41
6.6	4 Kbytes LDRAM (系统存储区) .....	41
6.6.1	BootLoader .....	42
6.7	SRAM .....	42
6.8	启动区域选择 (自举) .....	42
6.8.1	从主存储区自举 .....	42
6.8.2	从系统存储区自举 .....	43
6.8.3	从嵌入式 SRAM 自举 .....	43
6.8.4	自举模式设置 .....	43
6.9	96 Bits Unique ID .....	43
6.10	User ID 区域 .....	43
6.11	编程 .....	43
6.11.1	JTAG 专用模式 .....	44
6.11.2	常规模式 (JTAG 专用口无效) .....	44
6.12	安全加密 .....	45
6.12.1	安全加密操作权限 .....	45
6.13	In Application Programming (IAP) .....	45
6.13.1	IAP 操作相关寄存器 .....	46
6.14	选项字节区域 (Customer Option) .....	48
6.14.1	Customer Option 的映射寄存器 .....	49
7	模数转换器 (ADC) .....	51
7.1	概述 .....	51
7.2	时钟源 .....	51
7.3	特性 .....	51
7.4	ADC 采样和转换时间 .....	51
7.5	采样模式 .....	52
7.6	转换模式 .....	52
7.6.1	单次转换模式 .....	52
7.6.2	序列转换模式 .....	52
7.7	ADC 溢出 .....	53

7.8	ADC 与 DMA 控制器配合使用.....	54
7.9	ADC 转换步骤.....	54
7.10	ADC 连接电路图.....	55
7.11	ADC 中断.....	55
7.12	ADC 寄存器.....	55
7.12.1	ADC 相关寄存器表.....	55
7.12.2	ADC 寄存器映射.....	62
8	内部基准源 (VREF).....	63
8.1	概述.....	63
8.2	内部基准源模块配置.....	63
8.3	内部基准源输出.....	63
8.4	内部基准源功能框图.....	63
8.5	VREF 寄存器.....	64
8.5.1	VREF 相关寄存器表.....	64
8.5.2	VREF 寄存器映射.....	65
9	数模转换器 (DAC).....	66
9.1	概述.....	66
9.2	时钟源.....	66
9.3	特性.....	66
9.4	DAC 寄存器.....	66
9.4.1	DAC 相关寄存器表.....	66
9.4.2	DAC 寄存器映射.....	68
10	温度传感器.....	69
10.1	概述.....	69
10.2	温度传感器操作步骤.....	69
10.3	温度传感器寄存器.....	69
10.3.1	温度传感器相关寄存器表.....	69
10.3.2	温度传感器寄存器映射.....	70
11	运放及可编程增益放大器 (OP).....	71
11.1	概述.....	71
11.2	特性.....	71
11.3	OP0 框图.....	72
11.4	OP1 / OP2 框图.....	72
11.5	OP0 端口选择.....	73
11.5.1	OP0 精度调整.....	73
11.5.2	OP0 同相端输入.....	73
11.5.3	OP0 反相端输入.....	73

11.5.4	OP0 输出 .....	73
11.6	OP1/2 端口选择 .....	74
11.6.1	OP1/2 精度调整 .....	74
11.6.2	OP1/2 同相端输入 .....	74
11.6.3	OP1/2 反相端输入 .....	74
11.6.4	OP1/2 输出 .....	74
11.7	OP 寄存器 .....	74
11.7.1	OP0 相关寄存器表 .....	74
11.7.2	OP1/2 相关寄存器 .....	75
11.7.3	OP0/1/2 寄存器映射 .....	80
12	模拟比较器 (CMP) .....	81
12.1	概述 .....	81
12.2	CMP0/1/2 特性 .....	81
12.3	CMP3 特性 .....	81
12.4	模拟比较器结构框图 .....	82
12.5	CMP 中断 .....	82
12.6	CMP 寄存器 .....	82
12.6.1	CMP0/1/2 相关寄存器表 .....	82
12.6.2	CMP0/1/2 寄存器映射 .....	86
12.7	CMP3 寄存器 .....	87
12.7.1	CMP3 相关寄存器表 .....	87
12.7.2	CMP3 寄存器映射 .....	89
13	独立正交编码捕捉模块 (QEP) .....	90
13.1	概述 .....	90
13.2	时钟源 .....	90
13.3	特性 .....	90
13.4	正交计数 .....	90
13.4.1	正交模式结构框图 .....	90
13.4.2	正交模式真值表及波形 .....	91
13.5	方向计数 .....	92
13.5.1	方向计数模式结构框图 .....	93
13.5.2	方向计数模式波形 .....	93
13.6	双脉冲计数 .....	94
13.6.1	双脉冲计数模式结构框图 .....	94
13.6.2	双脉冲计数模式波形 .....	95
13.7	复位方式 .....	96
13.8	单位位置事件 .....	96

13.9	QEP 寄存器 .....	96
13.9.1	QEP 相关寄存器表 .....	96
13.9.2	QEP 寄存器映射 .....	100
14	16 位定时/计数器 (TIM) Timer0~Timer3.....	101
14.1	时钟源.....	101
14.2	特性.....	101
14.3	计数方式 .....	101
14.3.1	定时模式下 TIM 计数方式.....	101
14.3.2	PWM 输出模式下 TIM 计数方式 .....	101
14.4	定时器相关的信号口 .....	101
14.5	TIM 的中断及对应标志位 .....	102
14.6	TIM 的工作模式.....	102
14.6.1	工作模式 0: 16 位捕获模式 .....	103
14.6.2	工作模式 1: 16 位自动重载定时器/计数器模式 .....	103
14.6.3	工作模式 3: 可编程时钟输出模式.....	104
14.6.4	工作模式 4: PWM 输出模式 .....	104
14.7	TIM 中断 .....	104
14.8	TIM 寄存器.....	105
14.8.1	TIM 相关寄存器表.....	105
14.8.2	TIM 寄存器映射 .....	111
15	省电模式.....	114
16	GPIO .....	115
16.1	时钟源.....	115
16.2	特性.....	115
16.3	GPIO 结构图 .....	115
16.3.1	强推挽输出模式 .....	115
16.3.2	带上拉的输入模式 .....	115
16.3.3	高阻输入模式(Input only).....	116
16.4	GPIO 寄存器 .....	116
16.4.1	GPIO 相关寄存器表.....	116
16.4.2	GPIO 寄存器映射 .....	119
17	UART0~2 .....	121
17.1	时钟源.....	121
17.2	特性.....	121
17.3	UART2-LIN .....	122
17.3.1	LIN 帧结构 .....	122
17.3.2	LIN 主机模式 .....	122
17.3.3	LIN 从机模式 .....	122

17.4	UART 中断 .....	123
17.5	UART0/1 寄存器 .....	123
17.5.1	UART0/1 相关寄存器表 .....	123
17.5.2	UART0~1 寄存器映射 .....	127
17.6	UART2 寄存器 .....	128
17.6.1	UART2 相关寄存器表 .....	128
17.6.2	UART2 寄存器映射 .....	133
18	SPI0~1 .....	134
18.1	时钟源 .....	134
18.2	SPI0 特性 .....	134
18.3	SPI1 特性 .....	134
18.4	信号描述 .....	135
18.5	工作模式 .....	135
18.5.1	主模式 .....	135
18.5.2	从模式 .....	136
18.6	传送形式 .....	136
18.7	出错检测 .....	137
18.8	SPI0 和 SPI1 对比 .....	138
18.9	SPI 中断 .....	138
18.10	SPI0 寄存器 .....	139
18.10.1	SPI0 相关寄存器表 .....	139
18.10.2	SPI0 寄存器映射 .....	143
18.11	SPI1 寄存器 .....	143
18.11.1	SPI1 相关寄存器表 .....	143
18.11.2	SPI1 寄存器映射 .....	146
19	TWI0~1 .....	147
19.1	时钟源 .....	147
19.2	TWI0 特性 .....	147
19.3	TWI1 特性 .....	147
19.4	TWI 信号描述 .....	147
19.5	从机工作模式 .....	148
19.6	从机模式操作步骤 .....	150
19.7	主机工作模式 .....	151
19.8	主机模式操作步骤 .....	152
19.9	TWI0 中断 .....	153
19.10	TWI1 中断 .....	153
19.11	TWI0 寄存器 .....	153

19.11.1	TWI0 相关寄存器表 .....	153
19.11.2	TWI0 寄存器映射 .....	157
19.12	TWI1 寄存器 .....	158
19.12.1	TWI1 相关寄存器表 .....	158
19.12.2	TWI1 寄存器映射 .....	162
20	CAN 通信口 .....	163
20.1	概述 .....	163
20.2	时钟源 .....	163
20.3	特性 .....	163
20.4	CAN 协议 .....	164
20.4.1	CAN2.0 协议 .....	164
20.4.2	CAN_FD 协议 .....	166
20.5	功能描述 .....	169
20.5.1	配置波特率 .....	169
20.5.2	过滤器功能 .....	169
20.5.3	发送缓冲区 .....	170
20.5.4	自动重传屏蔽 .....	170
20.6	工作模式 .....	170
20.7	CAN 通信配置 .....	172
20.7.1	CAN 模块初始化 .....	172
20.7.2	发送报文配置 .....	172
20.7.3	发送流程 .....	173
20.7.4	接收流程 .....	173
20.7.5	报文收发优先级 .....	173
20.8	CAN 中断 .....	173
20.9	CAN 寄存器 .....	174
20.9.1	CAN 相关寄存器表 .....	174
20.9.2	CAN 寄存器映射 .....	189
21	硬件看门狗 WDT .....	192
21.1	概述 .....	192
21.2	时钟源 .....	192
21.3	WDT 寄存器 .....	192
21.3.1	WDT 相关寄存器表 .....	192
21.3.2	WDT 寄存器映射 .....	193
22	Base Timer (BTM) .....	194
22.1	概述 .....	194
22.2	时钟源 .....	194

22.3	特性.....	194
22.4	BTM 中断.....	194
22.5	BTM 寄存器.....	194
22.5.1	BTM 相关寄存器表.....	194
22.5.2	BTM 寄存器映射.....	196
23	内建 CRC 校验模块.....	197
23.1	概述.....	197
23.2	时钟源.....	197
23.3	特性.....	197
23.4	CRC 寄存器.....	198
23.4.1	CRC 相关寄存器表.....	198
23.4.2	CRC 寄存器映射.....	200
24	直接存储器访问(DMA).....	201
24.1	概述.....	201
24.2	时钟源.....	201
24.3	特性.....	201
24.4	功能说明.....	201
24.4.1	传输方向.....	201
24.4.2	DMA 访问区域限制.....	201
24.4.3	通道优先级.....	201
24.4.4	单一传输和批量传输.....	202
24.4.5	循环模式.....	202
24.4.6	DMA 通道使能后禁止操作的控制位.....	202
24.5	DMA 中断.....	202
24.6	DMA 寄存器.....	203
24.6.1	DMA 相关寄存器表.....	203
24.6.2	DMA 寄存器映射.....	208
25	SysTick.....	210
25.1	时钟源.....	210
25.2	SysTick 校准寄存器默认值.....	210
26	版本记录.....	211
27	声明.....	212

## 2 文档约定

### 2.1 名词解释

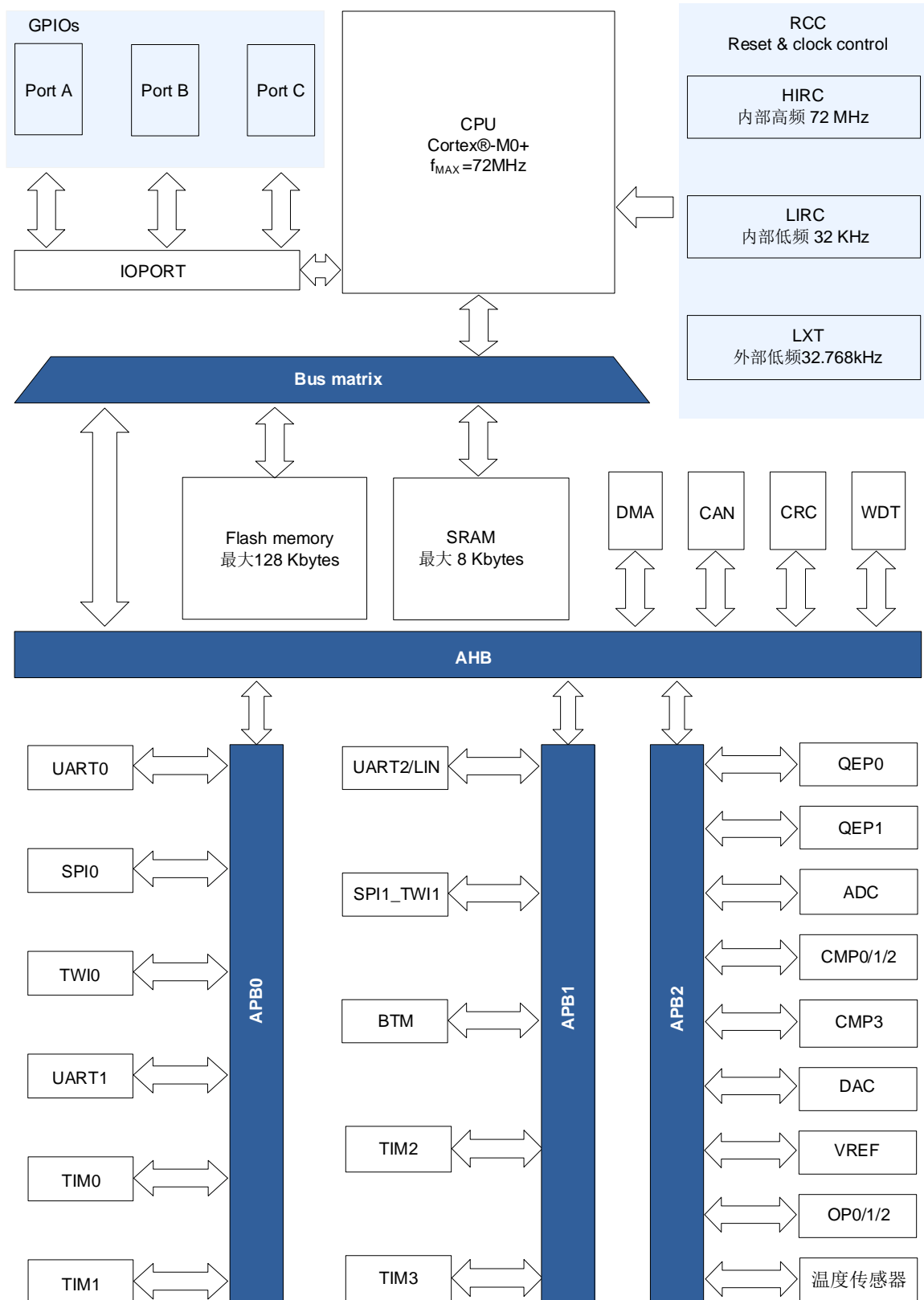
本节主要解释本文档中各首字母缩略词和缩写词的定义：

- 字：32 位数据
- 半字：16 位数据
- 字节：8 位数据
- 双字：64 位数据
- IAP（在应用中编程）：IAP 是指可以在用户程序运行期间对微控制器的 Flash 进行重新编程
- ICP（在线编程）：ICP 是指可以在器件安装于用户电路板上时使用 JTAG 协议、SWD 协议或自举程序对微控制器的 Flash 进行编程
- JTAG 协议：JTAG 协议是一种国际标准测试协议，主要用于芯片内部测试
- SWD 协议：SWD 协议是 ARM 设计的协议，代表串行线测试，用于对其微控制器进行编程和调试
- 选项字节：存储于 Flash 中的产品配置位
- AHB：高级高性能总线
- APB：高级外设总线

### 2.2 外设可用性

有关各型号产品的外设可用性及数量信息，请参考各产品最新的数据手册的产品外设资源表章节。

### 3 资源框图



## 4 上电、复位和时钟控制（RCC）

### 4.1 上电过程

SC32F15G 上电后，在客户端软件执行前，会经过以下三个阶段：

- ① 复位阶段；
- ② 调入信息阶段；
- ③ 正常操作阶段。

#### 4.1.1 复位阶段

复位阶段是指 SC32F15G 持续保持在复位状态，直到供应电压超过特定的阈值，其内部时钟机制才开始有效运作。

该阶段的具体时长受外部电源电压上升速率的影响，当外部电源提供的电压达到内建 POR 电压时，复位阶段即告结束。

#### 4.1.2 调入信息阶段

在 SC32F15G 内部设有一个预热计数器。复位阶段，此预热计数器一直被清零，供电电压上升至 POR 后，内部 HIRC 振荡器预热计数器开始计数。该预热计数器计数到一定数值后，周期性从 Flash ROM 中的 IFB（包含 Customer Option）读取数据至系统寄存器。所有 IFB 数据读取完毕后，预热及信息调入阶段结束，系统进入正常操作模式。

#### 4.1.3 正常操作阶段

完成信息调入后，SC32F15G 从 Flash 读取指令进入正常操作阶段。此时 LVR 电压值为用户写入 Customer Option 的设置值。

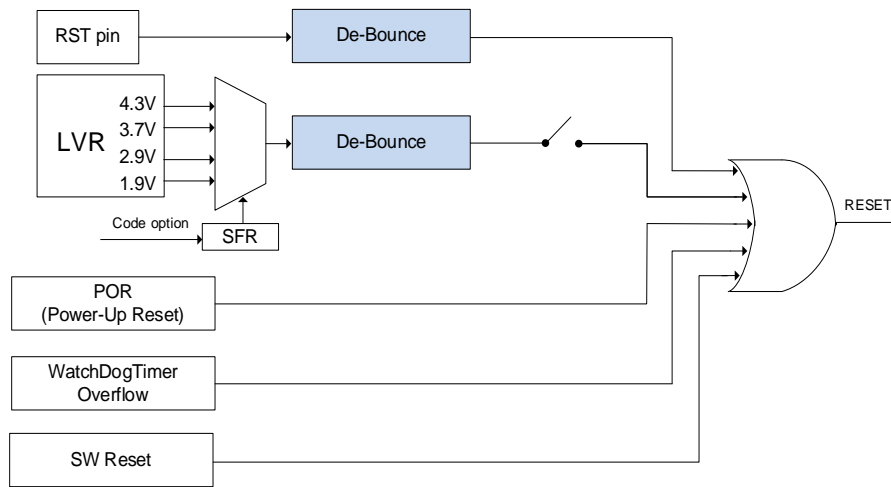
## 4.2 复位

SC32F15G 有两类复位方式：硬件复位和软件复位。当复位发生后，除 CAN 寄存器外的其他寄存器都将恢复成其复位值。CAN 寄存器复位值详见 [20.9 CAN 寄存器](#)。

SC32F15G 有 5 种复位方式，前四种为硬件复位：

- ① 外部 RST 复位
- ② 低电压复位 LVR
- ③ 上电复位 POR
- ④ 看门狗 WDT 复位
- ⑤ 软件复位

SC32F15G 的复位部分电路结构图如下：



SC32F15G 复位电路图

## 4.2.1 硬件复位

硬件复位可通过以下四种方式完成：

- ① 外部 RST 复位
- ② 低电压复位 LVR
- ③ 上电复位 POR
- ④ 看门狗 WDT 复位

### 4.2.1.1 外部 RST 复位

SC32F15G 的外部 RST 复位可通过在外部 RST 引脚上输入一个脉宽大于 18us 的低电平复位脉冲信号实现。

用户在烧录程序前可通过烧录上位机软件配置 Customer Option 项将 PC11 / NRST 管脚配置为 RST（复位脚）使用。

- 当该管脚配置为复位功能时，该管脚不能作为 GPIO 功能使用。
- 当该管脚配置为 GPIO 功能时，该管脚的复位功能不可用。

### 4.2.1.2 低电压复位 LVR

SC32F15G 内建一个低电压复位电路，支持 4 种门限电压选择：4.3V、3.7V、2.9V、1.9V。出厂时，默认的门限电压值为 1.9V，用户可通过烧录时设置 Customer Option 的值来设定缺省值。当 V<sub>DD</sub> 电压低于设定的门限值，且持续时间超过约 30μs 的消抖时间 T<sub>LVR</sub> 时，则触发复位操作。

### 4.2.1.3 上电复位 POR

SC32F15G 内部有上电复位电路，当 V<sub>DD</sub> 电压达到 POR 复位电压时，系统自动复位。

#### 4.2.1.4 看门狗复位 WDT

SC32F15G 有一个 WDT，其时钟源为内部的 32kHz 振荡器。用户可以通过编程器的 Customer Option 选择是否开启看门狗复位功能。

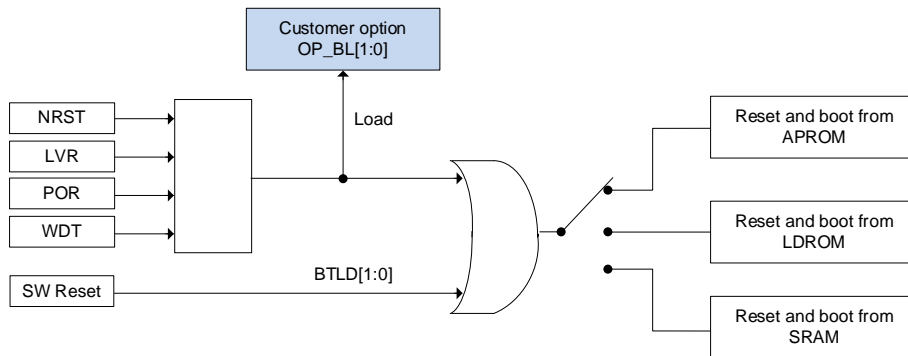
#### 4.2.2 软件复位

SC32F15G 提供软件复位功能，用户可以通过对 RST (IAP\_CON.8) 位写 1 后，使得系统立刻复位。

#### 4.2.3 复位后的启动区域

外部 RST 复位、低电压复位 LVR、上电复位 POR、看门狗 WDT 这四种硬件复位后，芯片从用户 OP\_BL[1:0] 设定的启动区域 (APROM / LDROM / SRAM) 启动。

软件复位后，芯片根据 BTLD[1:0] 设定的启动区域 (APROM / LDROM / SRAM) 启动。



SC32F15G 复位后启动区域切换示意图

#### 4.2.4 复位初始状态

当 SC32F15G 处于复位状态时，多数寄存器会回到其初始状态。看门狗 WDT 功能将在此阶段被禁用。“热启动”下的 Reset (如 WDT、LVR、软件复位等) 不会对 SRAM 中的数据产生影响，因此 SRAM 将保留复位前的值。

**注意：**当电源电压降低到无法维持 RAM 数据保存的程度时，SRAM 中的内容将会丢失。

### 4.3 时钟

#### 4.3.1 系统时钟源

系统时钟 SYSCLK 可通过三种不同的时钟源驱动：

- 内建高频 72MHz 振荡器 (HIRC)
- 内建低频 32kHz 振荡器 (LIRC)
- 外接低频晶振 (LXT)

对于每个振荡器，在未使用时可单独进行打开或关闭以降低功耗。

注意：

1. 上电默认的系统时钟源为 HIRC，上电默认的系统时钟频率为  $f_{HIRC}/2$ ，用户可在上电后的正常操作阶段，通过软件方式切换时钟源。切换前务必确保所选时钟源已处于稳定的工作状态；
2. 系统时钟源切换时，必须先将系统时钟源切换至 HIRC，再切换至目标时钟源。

#### 4.3.2 内建高频 72MHz 振荡器（HIRC）

内建高频时钟 HIRC 可通过对寄存器 RCC\_CFG0 的 HIRCEN 位置 1 开启，或清 0 关闭；且用户需要先对 RCC 保护寄存器 RCC\_KEY 写入一个大于 0x40 的值后才能对 RCC\_CFG0 和 RCC\_CFG1 进行改写。

HIRC 有以下功能及特性：

- 作为系统运行时钟
- 系统上电默认时钟频率  $f_{SYS}$  为  $f_{HIRC}/2$
- 频率误差：跨越（2.0V~5.5V）及（-40 ~ 105℃）应用环境下的频率误差  $\leq \pm 1\%$
- 可通过 32.768kHz 外接晶振进行自动校准，校准后 HIRC 精度可无限接近外接 32.768kHz 晶振的精度

#### 4.3.3 内建低频 32kHz 振荡器（LIRC）

内建低频时钟 LIRC 可通过对寄存器 RCC\_CFG0 的 LIRCEN 位置 1 开启，或清 0 关闭；且用户需要先对 RCC 保护寄存器 RCC\_KEY 写入一个大于 0x40 的值后才能对 RCC\_CFG0 和 RCC\_CFG1 进行改写。

LIRC 有以下功能及特性

- 作为系统运行时钟
- 作为 Base Timer 的时钟源
- 固定为 WDT 时钟源，WDT 使能后此时钟源必开启
- 频率误差：跨越（4.0 ~ 5.5V）及 25℃应用环境，经寄存器修正后频率误差  $\leq \pm 4\%$

#### 4.3.4 内置低频振荡电路，可外接 32.768kHz 低频震荡器（LXT）

外接低频晶振时钟 LXT 可通过对寄存器 RCC\_CFG0 的 LXTEN 位置 1 开启，或清 0 关闭；且用户需要先对 RCC 保护寄存器 RCC\_KEY 写入一个大于 0x40 的值后才能对 RCC\_CFG0 和 RCC\_CFG1 进行改写。

LXT 有以下功能及特性

- 作为系统运行时钟
- 作为 Base Timer 时钟源
- 外接 32.768kHz 低频振荡器
- 可通过 LXT 对 HIRC 进行自动校准

## 4.4 寄存器访问时钟和外设时钟源

### 4.4.1 寄存器访问时钟

寄存器访问时钟是驱动 Cortex<sup>®</sup>-M0+内核及内部总线操作的时钟信号，该信号直接驱动指令执行流水线、寄存器组访问以及内核与总线之间的数据传输。

寄存器访问时钟与 AHB (Advanced High-performance Bus) 和 APB (Advanced Peripheral Bus) 总线时钟保持严格同步，确保 CPU 核心操作与外设接口在统一时序框架下协同工作。用户需通过 AHB 总线外设时钟使能寄存器 (AHB\_CFG) 或 APB 总线外设时钟使能寄存器 (APBn\_CFG, n=0~2) 进行总线时钟使能和外设时钟门控操作。仅当对应外设的总线和寄存器访问时钟位均被使能 (Enable) 时，方可进行该外设寄存器的读写操作，否则将触发硬件保护机制导致访问无效。

### 4.4.2 外设时钟源

外设时钟源是为特定外设模块 (如 PWM、UART、ADC 等) 提供独立工作节拍的时钟信号源。它可以是系统时钟的分频/倍频产物，也可以是独立的内置或外部时钟源 (如内部 RC 振荡器、外部晶振等)。外设时钟源的存在使得不同外设可根据自身需求选择最优的时钟频率和稳定性。用户也可通过单独关闭未使用外设的时钟以降低功耗。

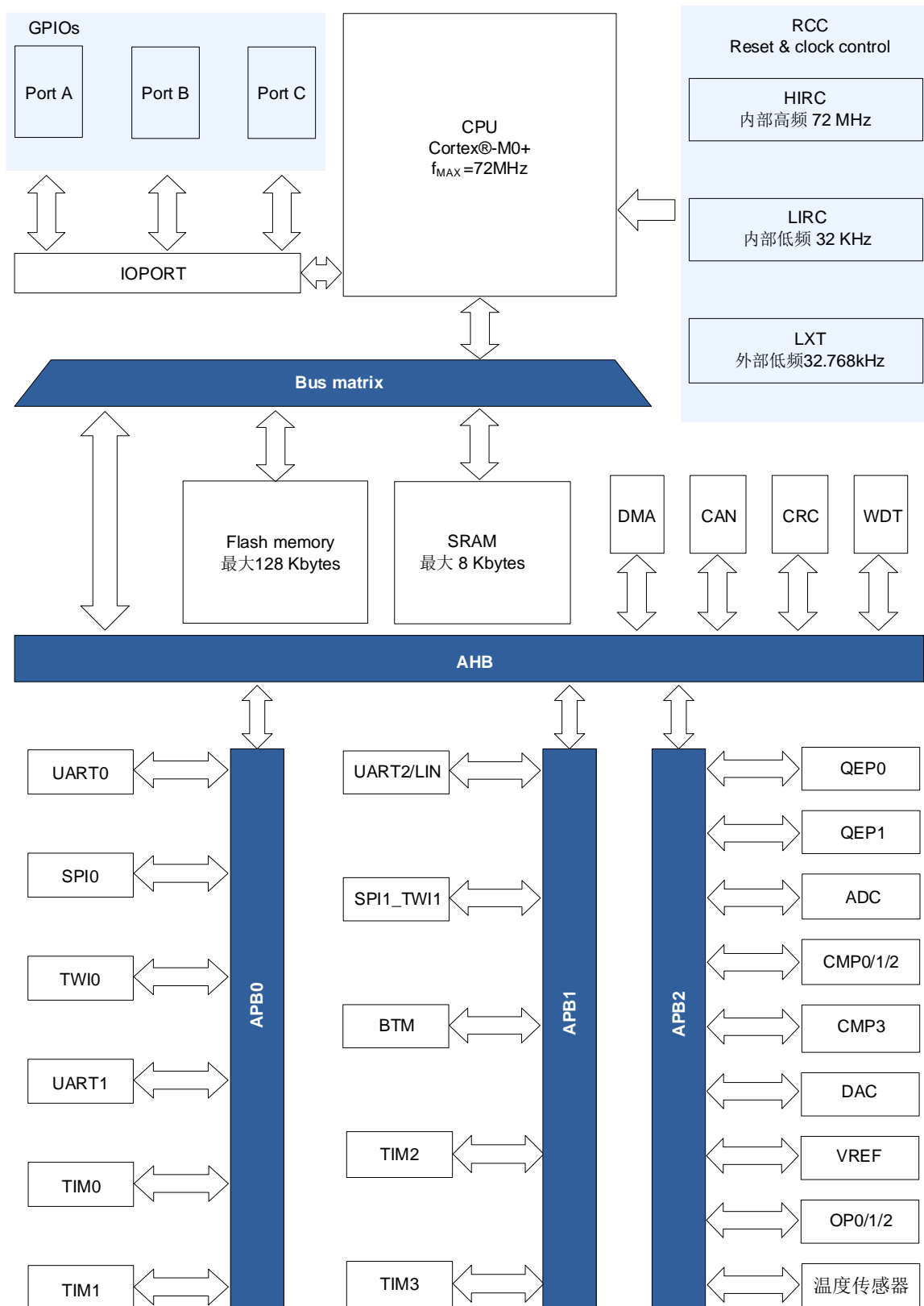
SC32F15G 的部分外设提供了多种可选的时钟源，用户可通过外设时钟源选择寄存器 (RCC\_CFGn, n=0~1) 为外设选择适合的时钟源。

### 4.4.3 总线

用户可通过多个预分频器配置 IOPORT、AHB、APB0、APB1、APB2 域的频率。

- IOPORT: GPIO 端口域主时钟，由系统时钟 SYSCLK 分频得到，最大频率是 72MHz，所有 GPIO 都由 IOPORT 驱动。
- HCLK: AHB 域主时钟，由系统时钟 SYSCLK 分频得到，最大频率是 72MHz，包括 Cortex<sup>®</sup>-M0+内核、内存、DMA 等都由 HCLK 驱动。
- PCLK0: APB0 域主时钟，由 HCLK 分频得到，最大频率是 HCLK 的频率，APB0 总线上的外设寄存器访问时钟都由 PCLK0 驱动，总线上的部分外设时钟源由 PCLK0 提供；
- PCLK1: APB1 域主时钟，由 HCLK 分频得到，最大频率是 HCLK 的频率，APB1 总线上的外设寄存器访问时钟都由 PCLK1 驱动，总线上的部分外设时钟源由 PCLK1 提供；
- PCLK2: APB2 域主时钟，由 HCLK 分频得到，最大频率是 HCLK 的频率，APB2 总线上的外设寄存器访问时钟都由 PCLK2 驱动，总线上的部分外设时钟源由 PCLK2 提供；

总线架构图如下图所示，外设寄存器读取频率取决于挂载该外设的总线频率。



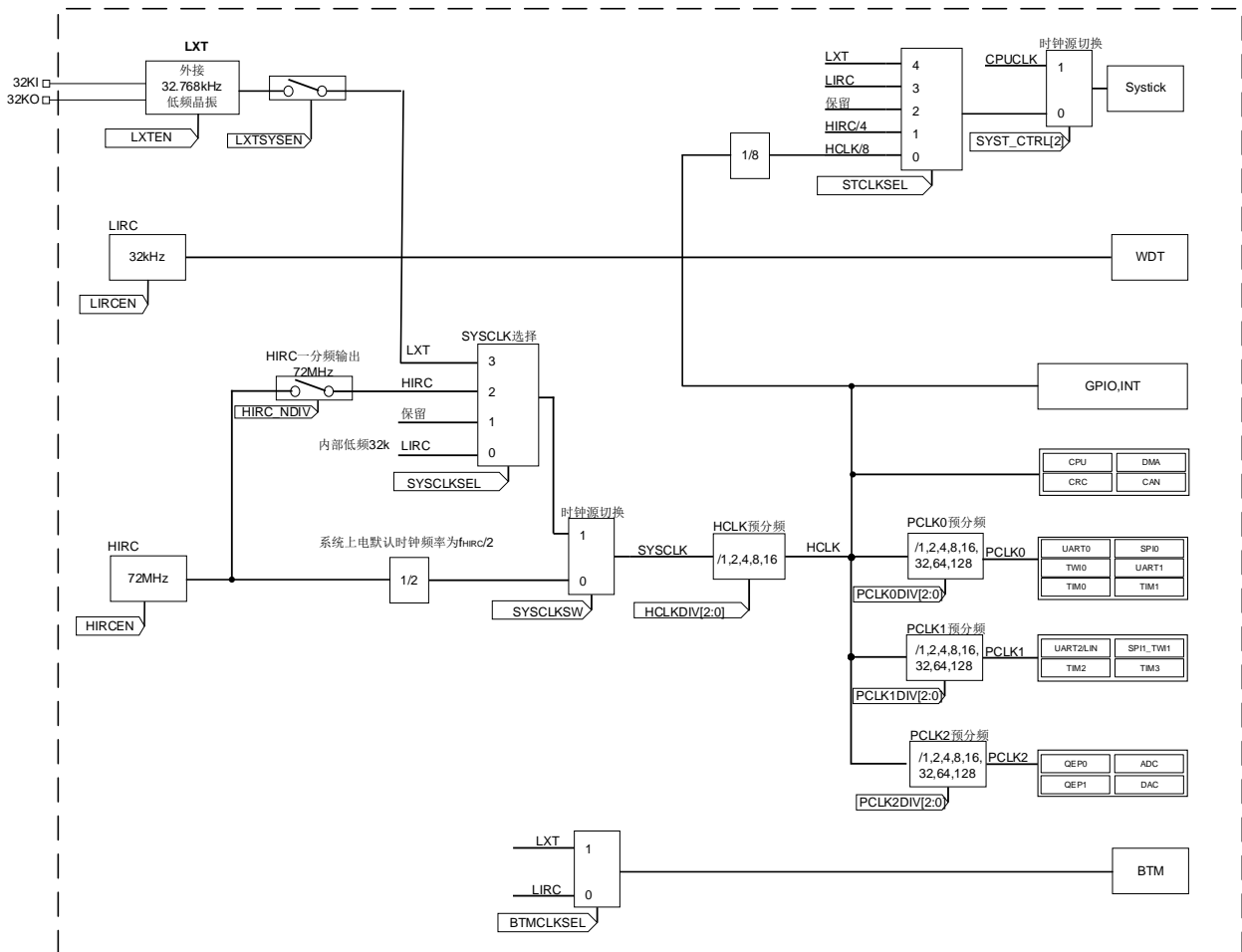
总线结构框图

#### 4.4.4 外设时钟

总线搭载的外设时钟一般由其对应的总线时钟提供，但以下外设可由其他时钟源提供时钟：

- BTM 有以下时钟源可选：
  - LIRC，为内部低频时钟
  - LXT，为外接低频晶振时钟
- WDT 时钟源固定为 LIRC，当 WDT 使能后，LIRC 会自动开启，WDT 工作的过程中 LIRC 始终保持振荡，用户无法关闭。
- Systick 有以下时钟源可选：
  - LXT，为外接低频晶振时钟
  - LIRC，为内部低频时钟
  - HIRC/4，为内部高频时钟 4 分频后的时钟
  - HCLK/8，为 AHB 总线上的时钟 8 分频后的时钟
  - CPUCLK，为内核时钟

#### 4.4.5 外设时钟框图



注意：系统上电后，默认的时钟频率  $f_{\text{SYS}}$  为  $f_{\text{HIRC}}/2$ ，用户可通过修改寄存器位 **SYSCLKSW** 与 **SYSCLKSEL** 选择所需的时钟源。

#### 4.4.6 外设时钟使能寄存器

每个外设时钟都可通过 AHB\_CFG 和 APBn\_CFG 寄存器的相应使能位使能，当外设时钟未激活时，不支持外设寄存器进行读写访问。

#### 4.4.7 低功耗模式时钟

在低功耗 STOP 模式下，CPU 时钟及大部分时钟源和外设时钟都会停止，且程序停止运行。但以下时钟源在低功耗模式下仍能正常工作：

- 内建低频时钟 LIRC
- 外接低频振荡器时钟 LXT

### 4.5 RCC 寄存器

#### 4.5.1 RCC 相关寄存器表

##### 4.5.1.1 RCC 保护寄存器 RCC\_KEY

寄存器	读/写	说明	复位值	上电初始值
RCC_KEY	读/写	RCC 保护寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
7~0	RCCKEY[7:0]	RCC_CFG0、RCC_CFG1 这两个寄存器的操作使能开关及时限设置。 写入一个大于等于 0x40 的值 n，代表： 1. 打开 RCC_CFG0、RCC_CFG1 这两个寄存器的写操作功能； 2. n 个系统时钟后如果接收不到寄存器写入命令，则 RCC 改写功能被重新关闭。
31~8	-	保留

##### 4.5.1.2 系统时钟源选择寄存器 RCC\_CFG0（写保护）

\*该寄存器受写保护，必须操作 RCC 保护寄存器 RCC\_KEY 才能修改。

寄存器	读/写	说明	复位值	上电初始值
RCC_CFG0	读/写	系统时钟源选择寄存器	0x0000_1040	0x0000_1040

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-

23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
INTEN	HIRC_NDIV	WAIT[1:0]		HPLDO_DP	-	SYSCLKSEL[1:0]	
7	6	5	4	3	2	1	0
SYSCLKSW	HIRCEN	-	-	-	-	LIRCEN	LXTEN

位编号	位符号	说明
15	INTEN	中断请求 CPU 的使能控制位 0: 禁止中断请求 1: 使能中断请求
14	HIRC_NDIV	内建高频晶振一分频输出 72MHz 使能位 0: 禁止 1: 使能
13~12	WAIT[1:0] 复位值=01	00: 保留, 不建议用户设置为“00”, 防止出现异常 01: 1 个 wait, 36MHz 主频推荐, 上电默认值 10: 2 个 wait, 72MHz 主频推荐 11: 3 个 wait, 72MHz 主频推荐 注意: 用户设置主频为 36M 时, 至少需要 1 个 wait; 设置主频为 72M 时, 至少需要 2 个 wait
11	HPLDO_DP	低频系统时钟功耗调整位 0: 系统时钟源非 LIRC 推荐设置 1: 系统时钟源选择 LIRC 推荐设置。当系统时钟选择 LIRC, 此位写 1 可降低整体功耗
9~8	SYSCLKSEL[1:0]	系统时钟源选择位 00: 系统时钟源来自 LIRC 01: 保留 10: 系统时钟源来自 HIRC 一分频, 72MHz 11: 系统时钟源来自 LXT 注意: 1. 上电默认的系统时钟源为 HIRC, 上电默认的系统时钟频率为 $f_{HIRC}/2$ , 用户可在上电后的正常操作阶段, 通过软件方式切换时钟源。切换前务必确保所选时钟源已处于稳定的工作状态; 2. 系统时钟源无论选择切换至哪一种, 都必须先将系统时钟源切换至 HIRC, 再切换至目标时钟源。
7	SYSCLKSW	系统时钟源切换位, 使能后系统时钟源从 HIRC 切换至 SYSCLKSEL 所选的时钟: 0: 系统时钟源为 HIRC, 上电默认系统时钟频率为 $f_{HIRC}/2$ 1: 系统时钟源为 SYSCLKSEL 设置项 对此位改写后, 必须内部电路切换成功才会更新改写的值, 否则读到的一直是改写前的状态。用户可以通过读取此位的方式判断时钟源是否已切换成功。复位/唤醒后此位将自动清零, 即复位/唤醒后由 HIRC 提供系统时钟。 注意: 1. 上电默认的系统时钟源为 HIRC, 上电默认的系统时钟频率为 $f_{HIRC}/2$ , 用户可在上电后的正常操作阶段, 通过软件方式切换时钟源。切换前务必确保所选时钟源已处于稳定的工作状态; 2. 系统时钟源无论选择切换至哪一种, 都必须先将系统时钟源切换至 HIRC, 再切换至目标时钟源。

位编号	位符号	说明
6	HIRCEN	内建高频 72MHz 振荡器 HIRC 使能位 0: 禁止 1: 使能 当 SYSCCLKSW = 0, 选择 HIRC 作为系统时钟时, 此位不可写入。 复位/唤醒后此位将由硬件置 1, 即复位/唤醒后由 HIRC 提供系统时钟。
1	LIRCEN	内建低频 32kHz 振荡器 LIRC 使能位 0: 禁止 1: 使能
0	LXTEN	外接低频晶振 LXT 使能位 0: 禁止 1: 使能
31~16 10 5~2	-	保留

#### 4.5.1.3 外设时钟源选择寄存器 RCC\_CFG1 (写保护)

\*该寄存器受写保护, 必须操作 RCC 保护寄存器 RCC\_KEY 才能修改。

寄存器	读/写	说明	复位值	上电初始值
RCC_CFG1	读/写	外设时钟源选择寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
7~5	STCLKSEL[2:0]	SysTick 时钟源选择位 000: 时钟源来自 HCLK/8 001: 时钟源来自 HIRC/4 010: 保留 011: 时钟源来自 LIRC 100: 时钟源来自 LXT 注意: 用户配置时注意, 如果 SysTick 时钟源不是来自 HCLK, SysTick 时钟源频率必须小于或等于 f <sub>HCLK</sub> /2。
0	BTMCLKSEL	BTM 时钟源选择位 0: 时钟源来自 LIRC 1: 时钟源来自 LXT 对此位改写后, 必须等待内部电路切换成功才会更新改写的值, 否则读到的一直是改写前的状态。 用户可以通过读取此位的方式判断时钟源是否已切换成功。
31~8 4~1	-	保留

#### 4.5.1.4 时钟状态寄存器 RCC\_STS

寄存器	读/写	说明	复位值	上电初始值
RCC_STS	读/写	时钟状态寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
3	SRAMPEIF	SRAM 奇偶校验错误标志位 检测到 SRAM 奇偶校验错误时，此位由硬件置 1。通过软件写入 1 可将此位清 0 0: 未检测到 SRAM 奇偶校验错误 1: 检测到 SRAM 奇偶校验错误
31~4 2~0	-	保留

#### 4.5.1.5 SysTick 校准参数寄存器 SYST\_CALIB

寄存器	读/写	说明	复位值	上电初始值
SYST_CALIB	只读	SysTick 校准参数寄存器	0x0000_2327	0x0000_2327

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

位编号	位符号	说明
23~0	CALIB[23:0]	校准寄存器默认值： 若，上电默认时钟为 $f_{HCLK}/n$ (MHz)， $n$ 是上电默认分频系数，上电默认时钟源为 HIRC 则，SysTick 校准值初始值为 $1000 * (f_{HCLK}/n)$ ，即保证默认可产生 1ms 时间基准
31~24	-	保留

#### 4.5.1.6 AHB 总线外设时钟使能寄存器 AHB\_CFG

寄存器	读/写	说明	复位值	上电初始值
AHB_CFG	读/写	AHB 总线外设时钟使能寄存器	0x0010_0000	0x0010_0000

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

位编号	位符号	说明
22~20	CLKDIV[2:0]	AHB 时钟分频设置位 AHB 总线时钟 HCLK 来自系统时钟 SYSCLK 的分频： 000: $f_{HCLK} = f_{SYS}$ 001: $f_{HCLK} = f_{SYS} / 2$ 010: $f_{HCLK} = f_{SYS} / 4$ 011: $f_{HCLK} = f_{SYS} / 8$ 100: $f_{HCLK} = f_{SYS} / 16$ 其它：保留
4	CANEN	CAN 模块时钟使能位 0: 禁止 1: 使能
2	IFBEN	Customer Option 映射寄存器时钟使能位 使用 OPINX 配合 OPREG 改写 IFB 映射寄存器之前，应先要打开时钟使能。 0: 禁止 1: 使能
1	CRCEN	CRC 模块时钟使能位 0: 禁止 1: 使能
0	DMAEN	DMA 时钟使能位 0: 禁止 1: 使能
31~23 19~5 3	-	保留

#### 4.5.1.7 APB0 总线外设时钟使能寄存器 APB0\_CFG

寄存器	读/写	说明	复位值	上电初始值
APB0_CFG	读/写	APB0 总线外设时钟使能寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
ENAPB	CLKDIV[2:0]			-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
UART1EN	UART0EN	SPI0EN	TWIOEN	-	-	TIM1EN	TIM0EN

位编号	位符号	说明
23	ENAPB	APB0 总线时钟开关控制位 0: 禁止 1: 使能
22~20	CLKDIV[2:0]	APB0 时钟分频设置位 APB0 总线时钟 PCLK0 来自 HCLK 的分频: 000: $f_{PCLK0} = f_{HCLK}$ 001: $f_{PCLK0} = f_{HCLK} / 2$ 010: $f_{PCLK0} = f_{HCLK} / 4$ 011: $f_{PCLK0} = f_{HCLK} / 8$ 100: $f_{PCLK0} = f_{HCLK} / 16$ 101: $f_{PCLK0} = f_{HCLK} / 32$ 110: $f_{PCLK0} = f_{HCLK} / 64$ 111: $f_{PCLK0} = f_{HCLK} / 128$
7	UART1EN	UART1 时钟使能位 0: 禁止 1: 使能
6	UART0EN	UART0 时钟使能位 0: 禁止 1: 使能
5	SPI0EN	SPI0 时钟使能位 0: 禁止 1: 使能
4	TWI0EN	TWI0 时钟使能位 0: 禁止 1: 使能
1	TIM1EN	Timer1 时钟使能位 0: 禁止 1: 使能
0	TIM0EN	Timer0 时钟使能位 0: 禁止 1: 使能
31~24 19~8 3~2	-	保留

#### 4.5.1.8 APB1 总线外设时钟使能寄存器 APB1\_CFG

寄存器	读/写	说明	复位值	上电初始值
APB1_CFG	读/写	APB1 总线外设时钟使能寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
ENAPB	CLKDIV[2:0]			-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
UART2EN	-	-	SPI_TWIEN	-	-	TIM3EN	TIM2EN

位编号	位符号	说明
23	ENAPB	APB1 总线时钟开关控制位 0: 禁止 1: 使能
22~20	CLKDIV[2:0]	APB1 时钟分频设置位 APB1 总线时钟 PCLK1 来自 HCLK 的分频: 000: $f_{PCLK1} = f_{HCLK}$ 001: $f_{PCLK1} = f_{HCLK} / 2$ 010: $f_{PCLK1} = f_{HCLK} / 4$ 011: $f_{PCLK1} = f_{HCLK} / 8$ 100: $f_{PCLK1} = f_{HCLK} / 16$ 101: $f_{PCLK1} = f_{HCLK} / 32$ 110: $f_{PCLK1} = f_{HCLK} / 64$ 111: $f_{PCLK1} = f_{HCLK} / 128$
7	UART2EN	UART2 时钟使能位 0: 禁止 1: 使能
4	SPI_TWIEN	SPI_TWI 时钟使能位 0: 禁止 1: 使能
1	TIM3EN	Timer3 时钟使能位 0: 禁止 1: 使能
0	TIM2EN	Timer2 时钟使能位 0: 禁止 1: 使能
31~24 19~8 6~5 3~2	-	保留

#### 4.5.1.9 APB2 总线外设时钟使能寄存器 APB2\_CFG

寄存器	读/写	说明	复位值	上电初始值
APB2_CFG	读/写	APB2 总线外设时钟使能寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
ENAPB	CLKDIV[2:0]			-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
-	ADCEN	-	-	QEP1EN	QEP0EN	-	-

位编号	位符号	说明
23	ENAPB	APB2 总线时钟开关控制位 0: 禁止 1: 使能
22~20	CLKDIV[2:0]	APB2 时钟分频设置位 APB2 总线时钟 PCLK2 来自 HCLK 的分频:

位编号	位符号	说明
		000: $f_{PCLK2} = f_{HCLK}$ 001: $f_{PCLK2} = f_{HCLK} / 2$ 010: $f_{PCLK2} = f_{HCLK} / 4$ 011: $f_{PCLK2} = f_{HCLK} / 8$ 100: $f_{PCLK2} = f_{HCLK} / 16$ 101: $f_{PCLK2} = f_{HCLK} / 32$ 110: $f_{PCLK2} = f_{HCLK} / 64$ 111: $f_{PCLK2} = f_{HCLK} / 128$
6	ADCEN	ADC 时钟使能位 0: 禁止 1: 使能
3	QEP1EN	QEP1 时钟使能位 0: 禁止 1: 使能
2	QEP0EN	QEP0 时钟使能位 0: 禁止 1: 使能
31~24 19~7 5~4 1~0	-	保留

#### 4.5.1.10 AHB 总线外设复位控制寄存器 AHB\_RST

寄存器	读/写	说明	复位值	上电初始值
AHB_RST	读/写	AHB 总线外设复位控制寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
4	CANRST	CAN 复位控制位 该位通过软件写 1，由硬件自动清 0。 0: 无影响 1: 复位 CAN
1	CRCRST	CRC 复位控制位 该位通过软件写 1，由硬件自动清 0。 0: 无影响 1: 复位 RCC
0	DMARST	DMA 复位控制位 该位通过软件写 1，由硬件自动清 0。 0: 无影响 1: 复位 DMA
31~5 3~2	-	保留

#### 4.5.1.11 APB0 总线外设复位控制寄存器 APB0\_RST

寄存器	读/写	说明	复位值	上电初始值
APB0_RST	读/写	APB0 总线外设复位控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
UART1RST	UART0RST	SPI0RST	TWI0RST	-	-	TIM1RST	TIM0RST

位编号	位符号	说明
7	UART1RST	UART1 复位控制位 该位通过软件写 1，由硬件自动清 0。 0: 无影响 1: 复位 UART1
6	UART0RST	UART0 复位控制位 该位通过软件写 1，由硬件自动清 0。 0: 无影响 1: 复位 UART0
5	SPI0RST	SPI0 复位控制位 该位通过软件写 1，由硬件自动清 0。 0: 无影响 1: 复位 SPI0
4	TWI0RST	TWI0 复位控制位 该位通过软件写 1，由硬件自动清 0。 0: 无影响 1: 复位 TWI0
1	TIM1RST	Timer1 复位控制位 该位通过软件写 1，由硬件自动清 0。 0: 无影响 1: 复位 Timer1
0	TIM0RST	Timer0 复位控制位 该位通过软件写 1，由硬件自动清 0。 0: 无影响 1: 复位 Timer0
31~8 3~2	-	保留

#### 4.5.1.12 APB1 总线外设复位控制寄存器 APB1\_RST

寄存器	读/写	说明	复位值	上电初始值
APB1_RST	读/写	APB1 总线外设复位控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8

-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
UART2RST	-	-	SPI_TWIRST	-	-	TIM3RST	TIM2RST

位编号	位符号	说明
7	UART2RST	UART2 复位控制位 该位通过软件写 1，由硬件自动清 0。 0: 无影响 1: 复位 UART2
4	SPI_TWIRST	SPI_TWI 复位控制位 该位通过软件写 1，由硬件自动清 0。 0: 无影响 1: 复位 SPI_TWI
1	TIM3RST	Timer3 复位控制位 该位通过软件写 1，由硬件自动清 0。 0: 无影响 1: 复位 Timer3
0	TIM2RST	Timer2 复位控制位 该位通过软件写 1，由硬件自动清 0。 0: 无影响 1: 复位 Timer2
31~8 6~5 3~2	-	保留

#### 4.5.1.13 APB2 总线外设复位控制寄存器 APB2\_RST

寄存器	读/写	说明	复位值	上电初始值
APB2_RST	读/写	APB2 总线外设复位控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
-	ADCRST	-	-	QEP1RST	QEP0RST	-	-

位编号	位符号	说明
6	ADCRST	ADC 复位控制位 该位通过软件写 1，由硬件自动清 0。 0: 无影响 1: 复位 ADC
3	QEP1RST	QEP1 复位控制位 该位通过软件写 1，由硬件自动清 0。 0: 无影响 1: 复位 QEP1
2	QEP0RST	QEP0 复位控制位 该位通过软件写 1，由硬件自动清 0。 0: 无影响 1: 复位 QEP0

位编号	位符号	说明
6	ADCRST	ADC 复位控制位 该位通过软件写 1，由硬件自动清 0。 0：无影响 1：复位 ADC
3	QEP1RST	QEP1 复位控制位 该位通过软件写 1，由硬件自动清 0。 0：无影响 1：复位 QEP1
31~3 1~0	-	保留

#### 4.5.1.14 NMI 中断配置寄存器 NMI\_CFG

寄存器	读/写	说明	复位值	上电初始值
NMI_CFG	读/写	非可屏蔽中断（NMI）中断配置寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
KEY[15:8]							
23	22	21	20	19	18	17	16
KEY[7:0]							
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
CMP0EN	CMP3EN	OP1EN	OP2EN	SRAMPEEN	INT0EN	-	-

位编号	位符号	说明
31~16	KEY[15:0]	NMI_CFG 寄存器写保护开关 向 KEY[15:0]写入 0xA05F 解锁后才能对当前寄存器的低位进行改写操作。
7	CMP0EN	CMP0 非屏蔽中断源使能位 0：禁止触发 NMI 1：非屏蔽中断源使能 使能后，CMP0IF 置起时会触发 NMI，手动清除 CMP0IF 标志后方可退出 NMI 中断 注意：如果 CMP0 中断使能已开启（CMPX_IDE->INTEN=1；CMPX_IDE->CMP0IE=1），仍会优先处理 NMI
6	CMP3EN	CMP3 非屏蔽中断源使能位 0：禁止触发 NMI 1：非屏蔽中断源使能 使能后，CMP3IF 置起时会触发 NMI，手动清除 CMP3IF 标志后方可退出 NMI 中断 注意：如果 CMP3 中断使能已开启（CMP3_IDE->INTEN=1），仍会优先处理 NMI
5	OP1EN	OP1_CMP 非屏蔽中断源使能位 0：禁止触发 NMI 1：非屏蔽中断源使能 使能后，OP1IF 置起时会触发 NMI，手动清除 OP1IF 标志后方可退出 NMI 中断 注意：如果 OP1 中断使能已开启（OPX_IDE->INTEN=1；OPX_IDE->OP1IE=1），仍会优先处理 NMI

位编号	位符号	说明
4	OP2EN	OP2_CMP 非屏蔽中断源使能位 0: 禁止触发 NMI 1: 非屏蔽中断源使能 使能后, OP2IF 置起时会触发 NMI, 手动清除 OP1IF 标志后方可退出 NMI 中断 注意: 如果 OP2 中断使能已开启 (OPX_IDE->INTEN=1; OPX_IDE->OP2IE=1), 仍会优先处理 NMI
3	SRAMPEEN	SRAM 奇偶校验错误中断使能位 0: SRAM 奇偶校验错误禁止触发 NMI 1: SRAM 奇偶校验错误非屏蔽中断源使能 使能后, 读取 SRAM 时检测到 SRAM 奇偶校验错误会触发 NMI, 手动清除相应标志位后方可退出 NMI 中断
2	INT0EN	外部中断 INT0 非屏蔽中断源使能位 0: INT0 禁止触发 NMI 1: INT0 非屏蔽中断源使能 使能后, INT0 口上的上升/下降沿中断均会触发 NMI, 手动清除相应标志位后方可退出 NMI 中断。 注意: 若 INT0 中断已使能, 仍会优先处理 NMI。
15~8 1~0	-	保留

#### 4.5.2 RCC 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值
AHB 基地址: 0x4000_3000					
AHB_CFG	0x00	读/写	AHB 总线外设时钟使能寄存器	0x0010_0000	0x0010_0000
AHB_RST	0x04	读/写	AHB 总线外设复位控制寄存器	0x0000_0000	0x0000_0000
RCC_KEY	0x0C	读/写	RCC 保护寄存器	0x0000_0000	0x0000_0000
RCC_CFG0	0x14	读/写	系统时钟源选择寄存器	0x0000_1040	0x0000_1040
RCC_CFG1	0x18	读/写	外设时钟源选择寄存器	0x0000_0000	0x0000_0000
RCC_STS	0x20	读/写	时钟状态寄存器	0x0000_0000	0x0000_0000
SYST_CALIB	0x28	读/写	SysTick 校准参数寄存器	0x0000_2327	0x0000_2327
NMI_CFG	0x2C	读/写	NMI 中断配置寄存器	0x0000_0000	0x0000_0000

寄存器	偏移地址	读/写	说明	复位值	上电初始值
APB0 基地址: 0x4002_0000					
APB0_CFG	0x00	读/写	APB0 总线外设时钟使能寄存器	0x0000_0000	0x0000_0000
APB0_RST	0x04	读/写	APB0 总线外设复位控制寄存器	0x0000_0000	0x0000_0000

寄存器	偏移地址	读/写	说明	复位值	上电初始值
APB1 基地址: 0x4002_1000					

寄存器	偏移地址	读/写	说明	复位值	上电初始值
APB1_CFG	0x00	读/写	APB1 总线外设时钟使能寄存器	0x0000_0000	0x0000_0000
APB1_RST	0x04	读/写	APB1 总线外设复位控制寄存器	0x0000_0000	0x0000_0000

寄存器	偏移地址	读/写	说明	复位值	上电初始值
APB2 基地址: 0x4002_2000					
APB2_CFG	0x00	读/写	APB2 总线外设时钟使能寄存器	0x0000_0000	0x0000_0000
APB2_RST	0x04	读/写	APB2 总线外设复位控制寄存器	0x0000_0000	0x0000_0000

## 5 中断

- M0+内核最多提供 32 个中断源，中断号为 0~31，SC32F15G 系列共 25 个中断源
- 四级中断优先级可设，中断优先级通过内核寄存器组 Interrupt priority registers 设置

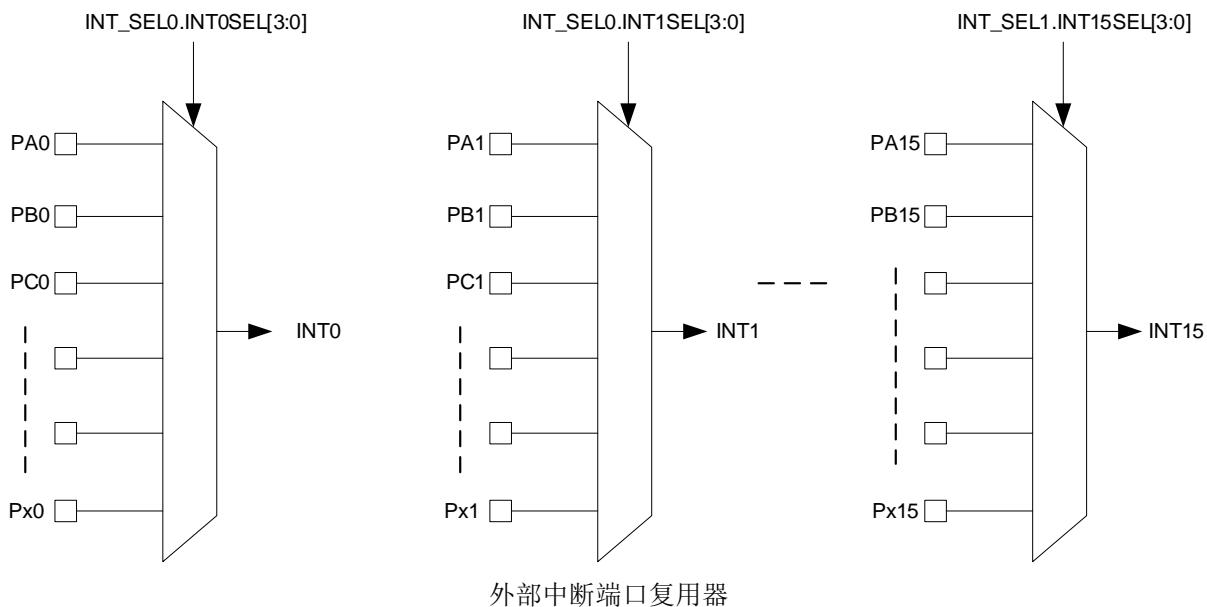
### 5.1 外部中断 INT0~15

外部中断有 16 个中断源，共占用 4 个中断向量，这 16 个外部中断源，全部可设上升沿、下降沿、双沿中断，经设置后可覆盖到所有的 GPIO 管脚，软件置起相应中断标志位（RIF/FIF 置 1），可触发进入相应的中断。

SC32F15G 系列外部中断的特性如下：

- 16 个 INT 中断源，共占用 4 个中断向量
- INT 经切换设置后可覆盖到所有的 GPIO 管脚
- 全部可设上升沿、下降沿、双沿中断，且均有独立对应中断标志位
- 软件置起相应中断标志位，可触发进入相应的中断

**注意：**切换 INT 功能时，用户需手动将 INTn（n=0~15）所在的 GPIO 端口设置为输入带上拉状态，端口输出状态下检测不到外部中断。



### 5.2 中断与事件

- NVIC 关闭，中断请求屏蔽位开启，可产生事件，不产生中断
- NVIC 开启后，中断请求屏蔽位做模块内的总中断开关

### 5.3 中断向量表

中断向量号	中断号	优先级	中断向量地址	中断源	内核/NVIC 使能位	中断请求屏蔽位	中断子开关	中断标志位	唤醒 stop
0	-	-	0x0000_0000	-		-	\	\	能
1	-	固定	0x0000_0004	RESET	PRIMASK	SCB	\	\	能
2	-	固定	0x0000_0008	NMI_Handler		SCB	\	\	能
3	-	固定	0x0000_000C	HardFault_Handler	PRIMASK	SCB	\	\	能
4-10	-	-	0x0000_0010 - 0x0000_0028	-		-	\	\	能
11	-	可设		SVC_Handler	PRIMASK	SCB	\	\	能
12-13	-	-	0x0000_0030 0x0000_0034	-		-	\	\	能
14	-	可设	0x0000_0038	PendSV_Handler	PRIMASK	SCB	\	\	能
15	-	可设	0x0000_003C	SysTick_Handler	PRIMASK	SysTick_CTRL	\	\	不能
16	0	可设	0x0000_0040	INT0	NVIC->ISER[0].0	INTF_IE->ENF <sub>x</sub> , x=0 INTR_IE->ENR <sub>x</sub>	\	INTF_STS->FIF <sub>x</sub> INTR_STS->RIF <sub>x</sub>	能
17	1	可设	0x0000_0044	INT1-7	NVIC->ISER[0].1	INTF_IE->ENF <sub>x</sub> , x=1-7 INTR_IE->ENR <sub>x</sub>	\	INTF_STS->FIF <sub>x</sub> INTR_STS->RIF <sub>x</sub>	能
18	2	可设	0x0000_0048	INT8-11	NVIC->ISER[0].2	INTF_IE->ENF <sub>x</sub> , x=8-11 INTR_IE->ENR <sub>x</sub>	\	INTF_STS->FIF <sub>x</sub> INTR_STS->RIF <sub>x</sub>	能
19	3	可设	0x0000_004C	INT12-15	NVIC->ISER[0].3	INTF_IE->ENF <sub>x</sub> , x=12-15 INTR_IE->ENR <sub>x</sub>	\	INTF_STS->FIF <sub>x</sub> INTR_STS->RIF <sub>x</sub>	能
20	4	可设	0x0000_0050	\	NVIC->ISER[0].4	\	\	\	
21	5	预留	0x0000_0054	\	NVIC->ISER[0].5	\	\	\	
22	6	可设	0x0000_0058	BTM	NVIC->ISER[0].6	BTM_CON->INTEN	\	BTM_STS->BTMIF	能
23	7	可设	0x0000_005C	UART0	NVIC->ISER[0].7	UART0_IDE->INTEN	UART0_IDE->TXIE UART0_IDE->RXIE	UART0_STS->TXIF UART0_STS->RXIF	能
				UART2/LIN		UART2_IDE->INTEN	UART2_IDE->TXIE UART2_IDE->RXIE UART2_IDE->BKIE UART2_IDE->SLVHEIE	UART2_STS->TXIF UART2_STS->RXIF UART2_STS->BKIF UART2_STS->SLVHEIF	不能
24	8	可设	0x0000_0060	UART1	NVIC->ISER[0].8	UART1_IDE->INTEN	UART1_IDE->TXIE UART1_IDE->RXIE	UART1_STS->TXIF UART1_STS->RXIF	能
25	9	可设	0x0000_0064	SPI0	NVIC->ISER[0].9	SPI0_IDE->INTEN	SPI0_IDE->RXNEIE SPI0_IDE->TBIE SPI0_IDE->RXIE SPI0_IDE->RXHIE SPI0_IDE->TXHIE	SPI0_STS->SPIF SPI0_STS->RXNEIF SPI0_STS->TXEIF SPI0_STS->RXFIF SPI0_STS->RXHIF SPI0_STS->TXHIF	不能
26	10	可设	0x0000_0068	SPI1	NVIC->ISER[0].10	SPI1_TWI1->INTEN	SPI1_TWI1_IDE->TBIE	SPI1_TWI1_STS->QTWIF SPI1_TWI1_STS->TXEIF	不能
				TWI1				SPI1_TWI1_STS->QTWIF	不能
27	11	可设	0x0000_006C	DMA0	NVIC->ISER[0].11	DMA0_CFG->INTEN	DMA0_CFG->TCIE DMA0_CFG->HTIE DMA0_CFG->TEIE	DMA0_STS->GIF DMA0_STS->TCIF DMA0_STS->HTIF DMA0_STS->TEIF	不能
28	12	可设	0x0000_0070	DMA1	NVIC->ISER[0].12	DMA1_CFG->INTEN	DMA1_CFG->TCIE DMA1_CFG->HTIE DMA1_CFG->TEIE	DMA1_STS->GIF DMA1_STS->TCIF DMA1_STS->HTIF DMA1_STS->TEIF	不能
29	13	可设	0x0000_0074	DMA2	NVIC->ISER[0].13	DMA2_CFG->INTEN	DMA2_CFG->TCIE DMA2_CFG->HTIE DMA2_CFG->TEIE	DMA2_STS->GIF DMA2_STS->TCIF DMA2_STS->HTIF DMA2_STS->TEIF	不能
30	14	可设	0x0000_0078	DMA3	NVIC->ISER[0].14	DMA3_CFG->INTEN	DMA3_CFG->TCIE DMA3_CFG->HTIE DMA3_CFG->TEIE	DMA3_STS->GIF DMA3_STS->TCIF DMA3_STS->HTIF DMA3_STS->TEIF	不能
31	15	可设	0x0000_007C	TIM0	NVIC->ISER[0].15	TIM0_IDE->INTEN	TIM0_IDE->TIE TIM0_IDE->EXFIE TIM0_IDE->EXRIE	TIM0_STS->TIF TIM0_STS->EXIF TIM0_STS->EXIR	不能
32	16	可设	0x0000_0080	TIM1	NVIC->ISER[0].16	TIM1_IDE->INTEN	TIM1_IDE->TIE TIM1_IDE->EXFIE TIM1_IDE->EXRIE	TIM1_STS->TIF TIM1_STS->EXIF TIM1_STS->EXIR	不能
33	17	可设	0x0000_0084	TIM2	NVIC->ISER[0].17	TIM2_IDE->INTEN	TIM2_IDE->TIE TIM2_IDE->EXFIE TIM2_IDE->EXRIE	TIM2_STS->TIF TIM2_STS->EXIF TIM2_STS->EXIR	不能

中断向量号	中断号	优先级	中断向量地址	中断源	内核/NVIC 使能位	中断请求屏蔽位	中断子开关	中断标志位	唤醒 stop
34	18	可设	0x0000_0088	TIM3	NVIC->ISER[0].18	TIM3_IDE->INTEN	TIM3_IDE->TIE TIM3_IDE->EXFIE TIM3_IDE->EXRIE	TIM3_STS->TIF TIM3_STS->EXIF TIM3_STS->EXIR	不能
35	19	预留	0x0000_008C	\	NVIC->ISER[0].19	\	\	\	不能
36	20	可设	0x0000_0090	QEP0	NVIC->ISER[0].20	QEP0_IDE->INTEN	QEP0_IDE->PCUIE QEP0_IDE->PCOIE QEP0_IDE->IERIE QEP0_IDE->UPEVNTIE	QEP0_STS->PCUIF QEP0_STS->PCOIF QEP0_STS->IERIF QEP0_STS->UPEVNTIF	不能
37	21	预留	0x0000_0094	\	NVIC->ISER[0].21	\	\	\	
38	22	可设	0x0000_0098	OP1_CMP	NVIC->ISER[0].22	OP_IDE->INTEN	OP_IDE->OP_CMP1IE	OP_STS->OP_CMP1IF	不能
				OP2_CMP			OP_IDE->OP_CMP2IE	OP_STS->OP_CMP2IF	不能
39	23	可设	0x0000_009C	TWI0	NVIC->ISER[0].23	TWI0_IDE->INTEN	\	TWI0_STS->TWIF	不能
40	24	可设	0x0000_00A0	QEP1	NVIC->ISER[0].24	QEP1_IDE->INTEN	QEP1_IDE->PCUIE QEP1_IDE->PCOIE QEP1_IDE->IERIE QEP1_IDE->UPEVNTIE	QEP1_STS->PCUIF QEP1_STS->PCOIF QEP1_STS->IERIF QEP1_STS->UPEVNTIF	不能
41	25	预留	0x0000_00A4	\	\	\	\	\	
42	26	预留	0x0000_00A8	\	\	\	\	\	
43	27	预留	0x0000_00AC	\	\	\	\	\	
44	28	可设	0x0000_00B0	CAN	NVIC->ISER[0].28	CAN_IDE->INTEN	CAN_RTIE->RIE CAN_RTIE->ROIE CAN_RTIE->RFIE CAN_RTIE->RAFIE CAN_RTIE->TPIE CAN_RTIE->TSIE CAN_RTIE->EIE CAN_RTIE->EPIE CAN_RTIE->ALIE CAN_RTIE->BEIE	CAN_RTIE->RIF CAN_RTIE->ROIF CAN_RTIE->RFIF CAN_RTIE->RAFIF CAN_RTIE->TPIF CAN_RTIE->TSIF CAN_RTIE->EIF CAN_RTIE->EPIF CAN_RTIE->ALIF CAN_RTIE->BEIF	不能
45	29	可设	0x0000_00B4	ADC	NVIC->ISER[0].29	ADC_IDE->INTEN	ADC_IDE->EOCIE ADC_IDE->EOSIE0 ADC_IDE->UPTHIE ADC_IDE->DOWTHIE	ADC_STS->ADCIF ADC_STS->EOSIF0 ADC_STS->UPTHIF ADC_STS->DOWTHIF	不能
46	30	可设	0x0000_00B8	CMP0	NVIC->ISER[0].30	CMPX_IDE->INTEN	CMPX_IDE->CMP0IE	CMPX_STS->CMP0IF	能
				CMP1			CMPX_IDE->CMP1IE	CMPX_STS->CMP1IF	
				CMP2			CMPX_IDE->CMP2IE	CMPX_STS->CMP2IF	
47	31	可设	0x0000_00BC	CMP3	NVIC->ISER[0].31	CMP3_IDE->INTEN	\	CMP3_STS->CMP3IF	能

## 5.4 外部中断寄存器

### 5.4.1 外部中断相关寄存器表

#### 5.4.1.1 INT 中断下降沿使能寄存器 INTF\_IE

寄存器	读/写	说明	复位值	上电初始值
INTF_IE	读/写	INT 中断下降沿使能寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
ENF15	ENF14	ENF13	ENF12	ENF11	ENF10	ENF9	ENF8
7	6	5	4	3	2	1	0
ENF7	ENF6	ENF5	ENF4	ENF3	ENF2	ENF1	ENF0

位编号	位符号	说明
15~0	ENF <sub>x</sub> (x=0~15)	INT <sub>x</sub> 下降沿中断使能控制位 (x=0~15) 0: 无效 1: 使能
31~16	-	保留

#### 5.4.1.2 INT 中断上升沿使能寄存器 INTR\_IE

寄存器	读/写	说明	复位值	上电初始值
INTR_IE	读/写	INT 中断上升沿使能寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
ENR15	ENR14	ENR13	ENR12	ENR11	ENR10	ENR9	ENR8
7	6	5	4	3	2	1	0
ENR7	ENR6	ENR5	ENR4	ENR3	ENR2	ENR1	ENR0

位编号	位符号	说明
15~0	ENR <sub>x</sub> (x=0~15)	INT <sub>x</sub> 上升沿中断使能控制位 (x=0~15) 0: 无效 1: 使能
31~16	-	保留

#### 5.4.1.3 外部中断端口选择寄存器 0 INT\_SELO

寄存器	读/写	说明	复位值	上电初始值
INT_SELO	读/写	外部中断端口选择寄存器 0	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
INT7SEL[3:0]				INT6SEL[3:0]			
23	22	21	20	19	18	17	16
INT5SEL[3:0]				INT4SEL[3:0]			
15	14	13	12	11	10	9	8
INT3SEL[3:0]				INT2SEL[3:0]			
7	6	5	4	3	2	1	0
INT1SEL[3:0]				INT0SEL[3:0]			

位编号	位符号	说明
31~0	INT <sub>x</sub> SEL[3:0] (x=0~7)	外部中断 INT <sub>x</sub> 端口选择位 (x=0~7) 0000: 选择 P <sub>A</sub> <sub>x</sub> 端口 0001: 选择 P <sub>B</sub> <sub>x</sub> 端口 0010: 选择 P <sub>C</sub> <sub>x</sub> 端口 其它: 保留  注意: 同一个外部中断口在同一时间内只能选择到一个 GPIO 口

#### 5.4.1.4 外部中断端口选择寄存器 1 INT\_SEL1

寄存器	读/写	说明	复位值	上电初始值
INT_SEL1	读/写	外部中断端口选择寄存器 1	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
INT15SEL[3:0]				INT14SEL[3:0]			
23	22	21	20	19	18	17	16
INT13SEL[3:0]				INT12SEL[3:0]			
15	14	13	12	11	10	9	8
INT11SEL[3:0]				INT10SEL[3:0]			
7	6	5	4	3	2	1	0
INT9SEL[3:0]				INT8SEL[3:0]			

位编号	位符号	说明
31~0	INTxSEL[3:0] (x=8~15)	外部中断 INTx 端口选择位 (x=8~15) 0000: 选择 PAX 端口 0001: 选择 PBx 端口 0010: 选择 PCx 端口 其它: 保留  注意: 同一个外部中断口在同一时间内只能选择到一个 GPIO 口

#### 5.4.1.5 外部中断下降沿控制寄存器 INTF\_CON

寄存器	读/写	说明	复位值	上电初始值
INTF_CON	读/写	外部中断下降沿控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
FT15	FT14	FT13	FT12	FT11	FT10	FT9	FT8
7	6	5	4	3	2	1	0
FT7	FT6	FT5	FT4	FT3	FT2	FT1	FT0

位编号	位符号	说明
15~0	FTx (x=0~15)	INTx 下降沿检测使能 (x=0~15) 0: 无效 1: 使能
31~16	-	保留

#### 5.4.1.6 外部中断上升沿控制寄存器 INTR\_CON

寄存器	读/写	说明	复位值	上电初始值
INTR_CON	读/写	外部中断上升沿控制寄存器	0x0000_0000	0x0000_0000

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

-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
RT15	RT14	RT13	RT12	RT11	RT10	RT9	RT8
7	6	5	4	3	2	1	0
RT7	RT6	RT5	RT4	RT3	RT2	RT1	RT0

位编号	位符号	说明
15~0	RTx (x=0~15)	INTx 上升沿检测使能 (x=0~15) 0: 无效 1: 使能
31~16	-	保留

#### 5.4.1.7 外部中断下降沿标志寄存器 INTF\_STS

寄存器	读/写	说明	复位值	上电初始值
INTF_STS	读/写	外部中断下降沿标志寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
FIF15	FIF14	FIF13	FIF12	FIF11	FIF10	FIF9	FIF8
7	6	5	4	3	2	1	0
FIF7	FIF6	FIF5	FIF4	FIF3	FIF2	FIF1	FIF0

位编号	位符号	说明
15~0	FIFx (x=0~15)	INTx 端口下降沿捕获标志位 (x=0~15) 检测到下降沿, 该位由硬件置 1, 通过软件写 0 清 0。 也可通过软件对该位写 1 触发下降沿捕获中断。
31~16	-	保留

#### 5.4.1.8 外部中断上升沿标志寄存器 INTR\_STS

寄存器	读/写	说明	复位值	上电初始值
INTR_STS	读/写	外部中断上升沿标志寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
RIF15	RIF14	RIF13	RIF12	RIF11	RIF10	RIF9	RIF8
7	6	5	4	3	2	1	0
RIF7	RIF6	RIF5	RIF4	RIF3	RIF2	RIF1	RIF0

位编号	位符号	说明
15~0	RIFx (x=0~15)	INTx 端口上升沿捕获标志位 (x=0~15) 检测到上升沿, 该位由硬件置 1, 通过软件写 0 清 0。 也可通过软件对该位写 1 触发上升沿捕获中断。
31~16	-	保留

### 5.4.2 外部中断寄存器映射

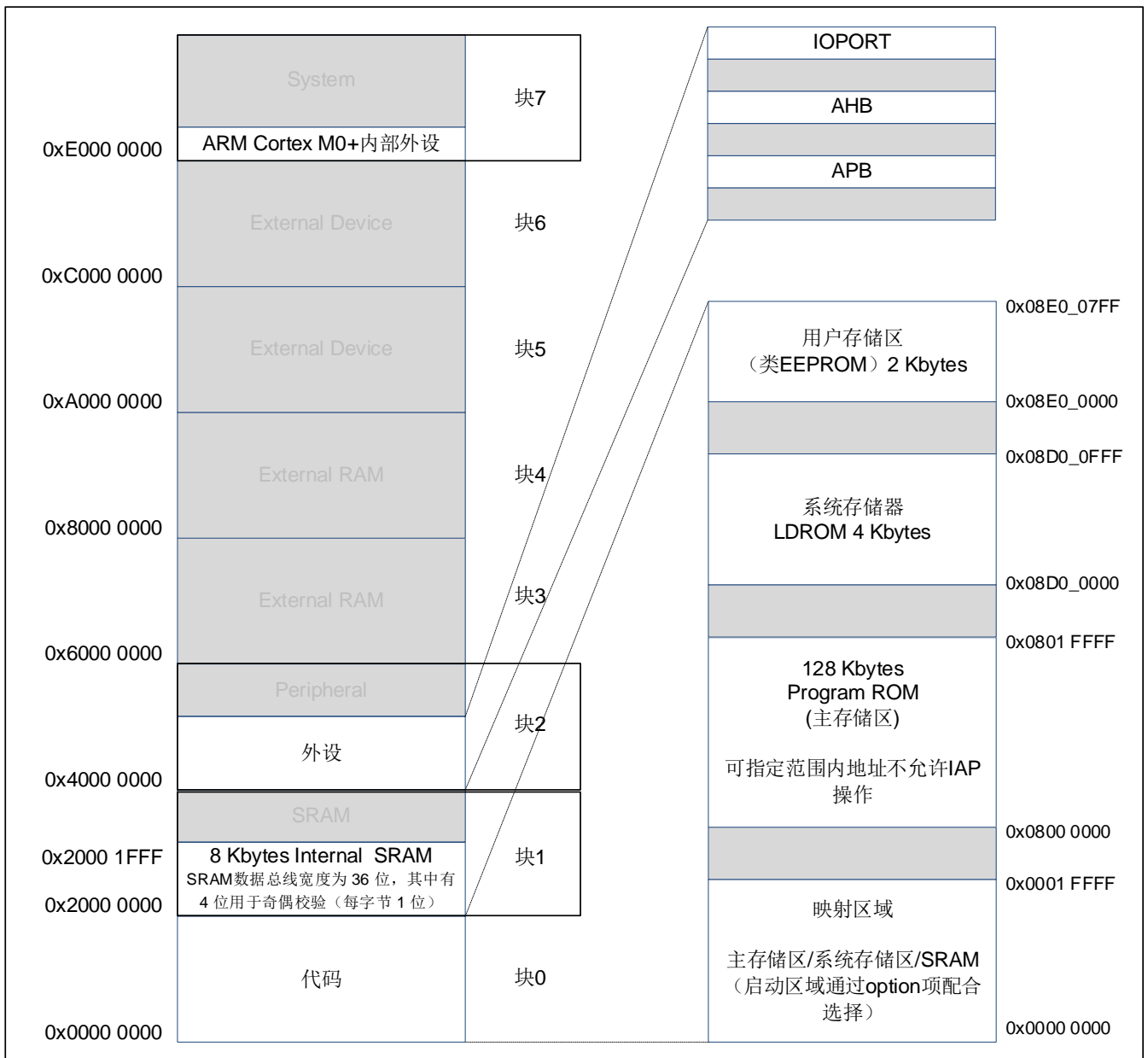
寄存器	偏移地址	读/写	说明	复位值	上电初始值
外部中断基地址: 0x4001_1800					
INTF_IE	0x00	读/写	INT 中断下降沿使能寄存器	0x0000_0000	0x0000_0000
INTR_IE	0x20	读/写	INT 中断上升沿使能寄存器	0x0000_0000	0x0000_0000
INT_SEL0	0x40	读/写	外部中断端口选择寄存器 0	0x0000_0000	0x0000_0000
INT_SEL1	0x60	读/写	外部中断端口选择寄存器 1	0x0000_0000	0x0000_0000
INTF_CON	0x80	读/写	外部中断下降沿控制寄存器	0x0000_0000	0x0000_0000
INTR_CON	0xA0	读/写	外部中断上升沿控制寄存器	0x0000_0000	0x0000_0000
INTF_STS	0xC0	读/写	外部中断下降沿标志寄存器	0x0000_0000	0x0000_0000
INTR_STS	0xE0	读/写	外部中断上升沿标志寄存器	0x0000_0000	0x0000_0000

## 6 存储

### 6.1 概述

程序存储器、数据存储器、寄存器排列在同一个线性（即地址连续）的 4 GB 地址空间内。各字节按小端格式在存储器中编码。字中编号最低的字节被视为该字的最低有效字节，而编号最高的字节被视为最高有效字节。可寻址的存储空间分为 8 个块，每个块为 512 MB。

### 6.2 存储框图



### 6.3 特性

- 存储 Flash 位宽为 32 Bits，可反复写入 10 万次
- 常温下数据保存时间为 100 年以上。
- Flash 的组成结构如下：
  - 最大 128 Kbytes APROM（主存储区）
  - 4 Kbytes LDROM（系统存储区）
  - 2 Kbytes 类 EEPROM（用户存储区）
  - 8 Kbytes 数据存储器 SRAM
  - 96 Bits Unique ID

### 6.4 APROM（主存储区）

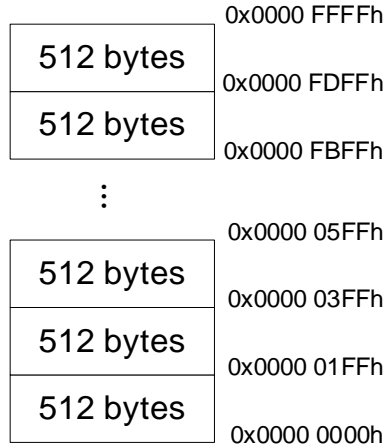
- 最大存储容量：128 Kbytes
- 扇区（sector）大小：512 bytes
- 支持操作：读/ 写/ 扇区擦除/ 全擦除/ 查空
- CPU（Cortex<sup>®</sup>-M0+）通过 AHB 总线访问 Flash
- 程序默认从主存储区启动，用户可通过 Customer Option OP\_BL[1:0]选择程序从 SRAM、LDROM 等其它区域启动
- 读保护：即读取加密，当读保护开启后，仅主存储区启动运行的程序可读取主存储区信息，其它区域启动或第三方工具均无法获取主存储区信息
- 写保护：提供两段禁止 IAP 操作的硬件写保护区域，用户可按扇区为单位设定该区域的范围

APROM（128 Kbytes）分为 256 个 512 bytes 的扇区（sector），用户烧录时，目标地址所属的 Sector 将被强制擦除后写入数据；用户写操作时，必须先擦除，再写入数据。



SC32F15G 系列 128 Kbytes APROM Sector 分区示意

APROM (64 Kbytes) 分为 128 个 512 bytes 的扇区 (sector)，用户烧录时，目标地址所属的 Sector 将被强制擦除后写入数据；用户写操作时，必须先擦除，再写入数据。

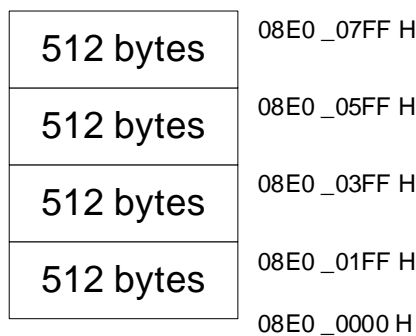


SC32F15G 系列 64 Kbytes APROM Sector 分区示意

## 6.5 2 Kbytes 类 EEPROM (用户存储区)

2K bytes 独立 EEPROM 区域地址为 0x08E0\_0000 H~ 0x08E0\_07FF H，由 IAPADE 寄存器设定。独立 EEPROM 可反复写入 10 万次，常温下数据保存时间为 100 年以上。独立 EEPROM 支持查空、编程、校验、擦除和读取功能。

EEPROM 共有 4 个 512 bytes 的扇区 (sector)：



SC32F15G EEPROM Sector 分区示意

**注意：EEPROM 擦写次数为 10 万次，用户擦写不要超过 EEPROM 的额定烧写次数，否则会出现异常！**

## 6.6 4 Kbytes LDROM (系统存储区)

- 系统存储区为 4 Kbytes LDROM，出厂固化 ISP 程序，该区域用户无法访问或改写

- 嵌入式自举程序为固化的 ISP 程序，该指令公开，允许用户通过 UART 重新编程 Flash。程序运行期间，若 500ms 内未接收到升级指令，则自动跳转至主存储区（0X0800 0000）执行

### 6.6.1 BootLoader

支持硬件和软件两种 BootLoad 方式，用户可根据需求灵活选择：

- 软件方式：通过软件划分 BootLoad 和 APP 区域、修改 VTOR 可轻易实现 BootLoad、APP 中断共用，调整各区大小
- 硬件方式：4 Kbytes 固定“系统存储区”作为专用 BootLoader 区域，用户不可对其进行读写操作：
  - 系统存储区作为一个固化的 BootLoader 空间，其中的程序在出厂前已烧录完成，用户不可读写。
  - 嵌入式自举程序位于系统存储区中，在生产阶段已完成编程。具有固化的 ISP 程序，该指令已公开，用户可利用该程序通过 UART 重新编程 Flash。

## 6.7 SRAM

- Internal SRAM: 8 Kbytes，地址 0x2000 0000 ~ 0x2000 1FFF
- 支持奇偶校验：
  - 额外的 1Kbytes RAM 用于奇偶校验：即 SRAM 数据总线宽度为 36 位，其中有 4 位用于奇偶校验（每字节 1 位）
  - 奇偶校验位在写入 SRAM 时进行计算和保存，在读取时自动进行校验。如果某一位失败，则将生成不可屏蔽中断（Cortex<sup>®</sup>-M0+ NMI）
  - 提供独立的 SRAM 奇偶校验错误标志 SRAMPEIF。

**注意：**当启用 SRAM 奇偶校验时，建议在代码开始处使用软件初始化整个 SRAM，以免在读取非初始化位置时出现奇偶校验错误。

- 用户可通过 Customer Option OP\_BL[1:0]选择程序从 SRAM 启动
- 能够以最大系统时钟频率按字节、半字（16 位）或全字（32 位）访问，无等待状态，因此可由 CPU 和 DMA 访问。

## 6.8 启动区域选择（自举）

复位后，用户可自行设置所需的自举模式配置。

退出待机模式后，还可以对启动模式配置进行重新采样。该启动延迟结束后，CPU 将从地址 0x0000 0000 获取栈顶值，然后从始于 0x0000 0004 的自举存储器开始执行代码。

自举区域选择有三种：主存储区、系统存储区和 SRAM，详细描述如下：

### 6.8.1 从主存储区自举

主存储区在自举存储器空间（0x0000 0000）中有别名，但也可从它原来的存储器空间（0x0800 0000）访问。换句话说：程序可从地址 0x0000 0000 或 0x0800 0000 开始访问。

## 6.8.2 从系统存储区自举

- 系统存储区（4 Kbytes LDR0M）作为一个固化的 BootLoader 空间，其中的程序是出厂前烧录好，用户不可读写。
- 嵌入式自举程序：嵌入式自举程序位于系统存储区中，在生产阶段编程。固化的 ISP 程序，该指令公开，可以利用该程序通过 UART 重新编程 Flash。

## 6.8.3 从嵌入式 SRAM 自举

SRAM 在自举存储器空间（0x0000 0000）中有别名，但也可从它原来的存储器空间（0x2000 0000）访问。

## 6.8.4 自举模式设置

通过寄存器控制位 BTLD[1:0]配合软件复位 RST 控制位可实现三种自举模式，BTLD 和 RST 受 IAP\_KEY 保护：

- ① 设置 BTLD[1:0]=0x00，芯片软件复位后从主存储区（APROM）启动
- ② 设置 BTLD[1:0]=0x01，芯片软件复位后从系统存储区（LDR0M）启动
- ③ 设置 BTLD[1:0]=0x10，芯片软件复位后从嵌入式 SRAM 启动

在芯片烧录阶段，通过 Customer Option 项 OP\_BL[1:0]实现芯片上电初始启动区域选择：

- ① 在 customer option 中设置 OP\_BL[1:0]=0x00，芯片复位后从主存储区（APROM）启动
- ② 在 customer option 中设置 OP\_BL[1:0]=0x01，芯片复位后从系统存储区（LDR0M）启动
- ③ 在 customer option 中设置 OP\_BL[1:0]=0x10，芯片复位后从嵌入式 SRAM 启动

## 6.9 96 Bits Unique ID

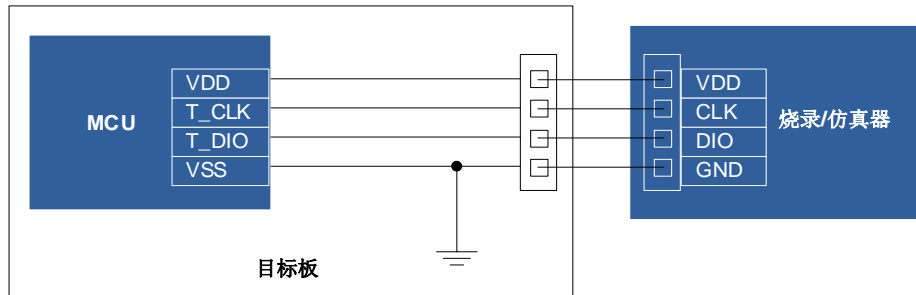
SC32F15G 提供了一个独立的 Unique ID 区域，出厂前会预烧一个 96 Bits 的唯一码，用以确保该芯片的唯一性。用户获得序列号的唯一方式是通过 IAP 指令读取。

## 6.10 User ID 区域

User ID 区域，出厂时写入用户定制 ID，用户可对其进行读操作，但禁止对 User ID 区域进行写操作。

## 6.11 编程

SC32F15G 的 Flash 通过 T\_DIO、T\_CLK、VDD、VSS 来进行编程，具体连接关系如下：



ICP 模式 Flash Writer 编程连接示意图

T\_DIO、T\_CLK 是 2 线 JTAG 烧写和仿真的信号线，用户在烧录时可通过 Customer Option 项配置这两端口的模式：JTAG 专用模式和常规模式（JTAG 专用口无效）。

**注意：UART0 的信号引脚支持两种映射方案：**

- 映射 1：RXD0 / TXD0
- 映射 2：RXD0A / TXD0A

当选择映射 1（RXD0/TXD0）时，该组引脚与系统烧录/调试接口（T\_CLK / T\_DIO）复用。在此映射下，若启用全双工通信，T\_CLK / T\_DIO 可能会与 UART0 的 RXD0 接收时序产生冲突，导致通信异常。因此，选择映射 1 时必须将 UART0 配置为半双工通信模式，以避免此硬件冲突并保证通信稳定性。

如需使用全双工 UART 通信，请将管脚映射至映射 2（RXD0A/TXD0A）。

### 6.11.1 JTAG 专用模式

JTAG 专用模式下，T\_DIO、T\_CLK 为烧写仿真专用口，与之复用的其它功能不可用。此模式一般用于在线调试阶段，方便用户仿真调试；JTAG 专用模式生效后，芯片无需重新上下电即可直接进入烧录或仿真模式。

### 6.11.2 常规模式（JTAG 专用口无效）

常规模式下，JTAG 功能不可用，端口上与之复用的其它功能可正常使用。此模式可防止烧录口占用 MCU 管脚，方便用户最大化利用 MCU 资源。

**注意：**当 JTAG 专用口无效的配置设定成功后，芯片必须彻底下电再重新上电后才能进入烧录或仿真模式，这样就会影响到带电模式下的烧录和仿真。赛元建议用户在量产烧录时选择 JTAG 专用口无效的配置，在研发调试阶段选择 JTAG 模式。

相关 Customer Option 如下：

寄存器	读/写	说明	复位值
COPT1_CFG@0xC2	读/写	Customer Option 映射寄存器 1	0x0000_0000

7	6	5	4	3	2	1	0
ENWDT	DISJTG	DISRST	-	-	-	OP_BL[1:0]	

位编号	位符号	说明
6	DISJTG	JTAG 口切换控制位 0: JTAG 模式使能, 对应管脚只能作为 T_CLK/ T_DIO 使用 1: 常规模式 (Normal), JTAG 功能无效

## 6.12 安全加密

SC32F15G 系列的安全加密功能主要是对 APROM 进行读保护加密: 用户可在烧录阶段通过赛元专用烧录套件的烧录上位机的 Customer Option 项配置读保护加密功能, 选择是否开启 Flash 读保护, 进入加密模式:

- 芯片默认出厂时 Flash 为非加密状态
- 读保护加密功能无映射寄存器, 用户只能在烧录阶段通过赛元专用烧录套件的烧录上位机的 Customer Option 项配置, 必须经过烧录才能完成修改。
- 加密失能: 可对主存储区执行读取、编程和擦除操作。也可对选项字节和备份寄存器进行所有操作。
- 加密使能:
  - 主存储区启动: 在用户模式下执行的代码 (从用户 APROM 自举) 可对主存储区执行所有操作。
  - 调试、从 SRAM 启动以及从系统存储区启动: 在调试模式下或当代码从 SRAM 或系统存储区启动时, 主存储区完全不可访问。
- 取消加密使能必须先对主存储区进行全擦操作。

### 6.12.1 安全加密操作权限

启动区域/工具	解锁状态					读保护加密状态				
	读	写	块擦	全擦	操作写保护区域	读	写	块擦	全擦	操作写保护区域
从 APROM 自举	√	√	√	\	禁止	√	√	√	\	禁止
调试/从 SRAM 自举	√	√	√	√	禁止	禁止	禁止	禁止	禁止	禁止
从系统存储区自举	√	√	√	√	√	禁止	禁止	禁止	√	禁止

## 6.13 In Application Programming (IAP)

SC32F15G 的 APROM 中的 IAP 区域可进行 In Application Programming (IAP) 操作, 用户可以通过 IAP 操作实现远程程序更新, 也可以通过 IAP 读操作获取 Unique ID 区域或 User ID 区域信息。进行 IAP 写数据操作前, 用户必须对目标地址所属的 Sector 进行扇区擦除操作。

芯片默认出厂时 APROM 允许全局 IAP 操作。芯片内部提供两组 APROM 写保护区域, 按照扇区单位设置起始, 被保护的区域禁止 IAP 操作, 设置规则如下:

IAPPORx 寄存器值 (x=A 或 B)	IAPPOR 保护区域
IAPPORx_ST = IAPPORx_ED	扇区 IAPPORx
IAPPORx_ST > IAPPORx_ED	无 (不受保护)

IAPPORx 寄存器值 (x=A 或 B)	IAPPOR 保护区域
IAPPORx_ST < IAPPORx_ED	从 IAPPORx_ST 到 IAPPORx_ED 的扇区

用户在烧录时可通过 Customer Option 项里的“Flash sectors protection”配置这两段 APROM 写保护区域。

### 6.13.1 IAP 操作相关寄存器

对写保护区域之外的 APROM 进行 IAP 操作，可通过以下寄存器实现：

#### 6.13.1.1 数据保护寄存器 IAP\_KEY

寄存器	读/写	说明	复位值	上电初始值
IAP_KEY	读/写	数据保护寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
31~0	IAPKEY[31:0]	数据保护锁 为防止因电气干扰等原因出现对 Flash 的意外操作，IAP_CON 寄存器执行写操作前需要通过 IAPKEY 解锁。解锁顺序如下： <ol style="list-style-type: none"> <li>写入 KEY1 = 0x1234_5678</li> <li>写入 KEY2 = 0xA05F_05FA</li> </ol> 如果操作顺序不正确，会锁定 IAP_CON 寄存器，直到下次系统复位才会解锁。

#### 6.13.1.2 IAP 扇区编号设置寄存器 IAP\_SNB

寄存器	读/写	说明	复位值	上电初始值
IAP_SNB	读/写	IAP 扇区编号设置寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
31~24	IAPADE[7:0]	IAP 操作区域扩展地址 通过向 IAPADE 写入不同的值，使得 IAP 操作指向不同的操作区域： 0x00：无效 0x4C：APROM 0x69：EEPROM 0xF1：customer option 其它：保留
8~0	IAPSNB[8:0]	扇擦/页擦时的 IAP 操作扇区编号设置： 实际操作扇区的起始地址= Flash 基址 + [ IAPSNB[8:0] x 0x200 ]
23~9	-	保留

### 6.13.1.3 IAP 控制寄存器 IAP\_CON (写保护)

\*该寄存器受写保护，必须操作数据保护寄存器 IAP\_KEY 才能修改。

寄存器	读/写	说明	复位值	上电初始值
IAP_CON	读/写	IAP 控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
LOCK	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	CONT[5:0]					
15	14	13	12	11	10	9	8
-	-	-	-	DMAEN	BTLD[1:0]		RST
7	6	5	4	3	2	1	0
ERASE	-	SERASE	PRG	-	-	CMD[1:0]	

位编号	位符号	说明
31	LOCK	对该位写 1 后，IAP_CON 寄存器被锁定。当检测到解锁序列时，由硬件将该位清 0。 如果解锁操作失败，该位仍保持置 1，直到下一次系统复位。
21~16	CONT[5:0]	IAP 写数据加速设置位 以 4bytes 连续地址为单位执行连续的 IAP 写操作，通过 CONT[5:0] 设置连续烧录长度，有效范围 0x01 ~ 0x20，即一次最多可连续烧录 32bits*32=128bytes 操作步骤： 1.待烧录的数据暂存在 RAM 区域，RAM buffer 最大 128bytes； 2.解锁 IAP_KEY，使能 IAP_CON.PRG。 3.使能 IAP_CON.DMAEN，设置 CONT[5:0]； 4.配置 DMA，选择其中一个空闲通道，通道源地址指向的 RAM buffer，目标地址指向待烧录区域的起始地址，注意烧录目标区域起始地址要 4bytes 对齐； 5.配置 DMA 通道，TPTYPE=1，选择批量模式，设置 TXWIDTH[1:0]=32bit，DMACNT[31:0]=CONT[5:0]； 6.触发 DMA 通道的软件搬运，SWREQ=1，开始执行连续烧录。
11	DMAEN	DMA 协助连续编程控制位 0：关闭 DMA 协助连续编程功能 1：开启 DMA 协助连续编程功能
10~9	BTLD[1:0]	软件复位后程序启动区域选择位： 00：芯片软件复位后从 APROM（主存储区）启动

位编号	位符号	说明
		01: 芯片软件复位后从 LDROM (系统存储区) 启动 10: 芯片软件复位后从嵌入式 SRAM 启动 11: 保留
8	RST	软件复位控制位 0: 程序正常运行 1: 对该位写 1 后系统立刻复位
7	ERASE	全擦 (All Erase) 控制位 0: 无操作 1: 对该位写 1 后再配置 CMD[1:0]=10, 则进入 APROM 全擦除操作, APROM 将全部擦除
5	SERASE	扇区擦除 (Sector Erase) 控制位 0: 无操作 1: 对该位写 1 后再配置 CMD[1:0]=10, 则进入 APROM 扇区擦除操作, APROM 的指定 Sector 将被擦除
4	PRG	编程 (Program) 控制位 0: 禁止 Flash 编程 1: 使能 Flash 编程
1~0	CMD[1:0]	IAP 命令使能控制位 10: 执行擦除操作命令 其它: 保留 注意: 1. 擦除操作命令位写 1 后, 必须配置 CMD[1:0]=10, 相应的操作才会开始执行 2. 一次只能执行 1 种 IAP 操作, 所以 ERASE/SERASE 位同一时间只能有一位置 1
30~22 15~12 6 3~2	-	保留

#### 6.13.1.4 IAP 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值
IAP 基地址: 0x4000_03C0					
IAP_KEY	0x00	读/写	数据保护寄存器	0x0000_0000	0x0000_0000
IAP_SNB	0x04	读/写	IAP 扇区编号设置寄存器	0x0000_0000	0x0000_0000
IAP_CON	0x0C	读/写	IAP 控制寄存器	0x0000_0000	0x0000_0000

## 6.14 选项字节区域 (Customer Option)

SC32F15G 有单独的一块 Flash 区域用于保存客户的上电初始值设置, 此区域称为选项字节区域

(Customer Option) 区域。用户在烧录时可通过上位机对 Customer Option 项进行配置, 在烧写过程中将配置值写入 Customer Option 区域, IC 在复位初始化阶段将调用 Customer Option 数据作为初始设置。

也可通过操作 Customer Option 的映射寄存器临时修改 Customer Option 项, 但是需要注意: 修改映射寄存

器仅可实现临时调整，不会对 Customer Option 区域的设置值造成任何影响，芯片复位后，仍会根据烧录时用户选择的 Customer Option 参数进行初始化。

Customer Option 相关映射寄存器的操作方式如下：

Customer Option 相关 SFR 的读写操作由 OPINX 和 OPREG 两个寄存器进行控制，各 Customer Option SFR 的具体位置由 OPINX 确定，如下表所示：

寄存器	地址	说明	复位值	上电初始值
OPINX	0x4000_03F8	Customer Option 指针	0x0000_0000	0x0000_0000
OPREG	0x4000_03FC	Customer Option 寄存器	0x0000_0000	0x0000_0000
COPT0_CFG	0XC1 @ OPINX	Customer Option 映射寄存器 0	0x0000_0000	0x0000_0000
COPT1_CFG	0XC2 @ OPINX	Customer Option 映射寄存器 1	0x0000_0000	0x0000_0000

### 6.14.1 Customer Option 的映射寄存器

使用 OPINX 配合 OPREG 改写 IFB 映射寄存器之前，应先要打开 Customer Option 寄存器的时钟使能开关 AHB\_CFG.IFBEN：

#### 6.14.1.1 AHB 总线外设时钟使能寄存器 AHB\_CFG

寄存器	读/写	说明	复位值	上电初始值
AHB_CFG	读/写	AHB 总线外设时钟使能寄存器	0x0010_0000	0x0010_0000

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

位编号	位符号	说明
2	IFBEN	Customer Option 映射寄存器时钟使能位 使用 OPINX 配合 OPREG 改写 IFB 映射寄存器之前，应先要打开时钟使能。 0：禁止 1：使能
31~23 19~3	-	保留

#### 6.14.1.2 Customer Option 映射寄存器 0 COPT\_CFG0

寄存器	地址	说明	复位值	上电初始值
COPT_CON0	读/写	Customer Option 映射寄存器 0	0x0000_0000	0x0000_0000

7	6	5	4	3	2	1	0
-	-	-	-	-	DISLVR	LVRS [1:0]	

位编号	位符号	说明
2	DISLVR	LVR 开关 0: LVR 有效 1: LVR 无效
1~0	LVRS [1:0]	LVR 电压选择控制 11: 4.3V 复位 10: 3.7V 复位 01: 2.9V 复位 00: 1.9V 复位
7~3	-	保留

### 6.14.1.3 Customer Option 映射寄存器 1 COPT\_CON1

寄存器	读/写	说明	复位值	上电初始值
COPT_CON1	读/写	Customer Option 映射寄存器 1	0x0000_0000	0x0000_0000

7	6	5	4	3	2	1	0
ENWDT	DISJTG	DISRST	-	-	-	OP_BL[1:0]	

位编号	位符号	说明
7	ENWDT	WDT 开关 1: WDT 开始工作 0: WDT 关闭
6	DISJTG	JTAG 口切换控制位 0: JTAG 模式使能, 对应管脚只能作为 T_CLK/ T_DIO 使用 1: 常规模式 (Normal), JTAG 功能无效
5	DISRST	复位信号口切换控制位 <b>该位只读, 用户不可改写。</b> 0: RST 对应管脚当复位脚使用 1: RST 所在管脚做正常的 GPIO 管脚使用
1~0	OP_BL[1:0]	芯片复位后启动区域选择 <b>该位只读, 用户不可改写。</b> 00: 芯片复位后从 APROM (主存储区) 启动 01: 芯片复位后从 LDRAM (系统存储区) 启动 10: 芯片复位后从嵌入式 SRAM 启动 11: 保留
4~2	-	保留

## 7 模数转换器 (ADC)

### 7.1 概述

SC32F15G 系列提供一个 12 位 ADC 逐次逼近型模数转换器。具有 18 个通道，可测量来自 16 个外部源和 2 个内部源的信号，2 个内部源分别是 V<sub>DD</sub> 电压和芯片温度。每个通道的 A/D 转换在采样后进行，ADC 的转换结果存储在一个 32 位数据寄存器中。

### 7.2 时钟源

- SC32F15G 系列的 ADC 的采样时钟为 PCLK2
- ADC 的单次转换时间约为 404ns

### 7.3 特性

- 精度：12 位
- 最多支持 18 路通道：
  - 外部 16 路 ADC 采样通道和 I/O 端口的其他功能复用
  - 外部 3 路 AIN 与 OP 复用，可测量 OP 模块的输出信号，分别为 OP0、OP1、OP2
  - 内部一路 ADC 可直接测量 V<sub>DD</sub> 电压
  - 内部一路芯片温度采样通道
- 提供 ADC 阈值看门狗功能，精确监测所有选定通道的转换电压，可同时设定上下阈值，当转换电压超出编程阈值时，可产生中断
- 两种转换模式可选：
  - 单次转换，通过软件触发
  - 序列转换，通过软件触发
- 可设 ADC 转换完成中断
- 单次转换时间约为 404ns
- 支持 DMA 传输：ADC 转换完成可产生 DMA 请求
- ADC 转换结果支持溢出提醒，当溢出发生时 OVERRUN 标志位置起，且 OVERRUN 标志位与 ADC 转换结果在同一寄存器 ADCV，用户可一次性读取

### 7.4 ADC 采样和转换时间

LOWSP[2:0] 设置值	采样时钟个数	采样时间 @F <sub>PCLK</sub> = 72MHz 单位：ns	转换时间典型值 单位：ns	ADC 从采样到完成转换 的总时间 ns
000	3	42	404	446
001	6	83	404	487

LOWSP[2:0] 设置值	采样时钟个数	采样时间 @F <sub>PCLK</sub> = 72MHz 单位: ns	转换时间典型值 单位: ns	ADC 从采样到完成转换 的总时间 ns
010	9	125	404	529
011	15	208	404	612
100	30	417	404	821
101	60	833	404	1237
110	120	1667	404	2071
111	480	6667	404	7071

## 7.5 采样模式

可选择以下两种方法触发采样：

- 手动触发：对 ADCS 写 1，一次采样一个 ADCISA[4:0]选中的通道并转换，转换结果存放在 ADCVA[11:0]
- 序列转换：符合序列转换触发条件后，按照序列选中的有效 DS<sub>n</sub> 编号从小到大的顺序逐个通道进行采样+转换

## 7.6 转换模式

用户可通过采样模式选择位 ADC\_CON.CONT 设置转换模式：

### 7.6.1 单次转换模式

CONT=0，为单次转换模式：

- 对 ADCS 写 1，一次采样一个 ADCISA[4:0]选中的通道并转换，转换结果存放在 ADCVA[11:0]

### 7.6.2 序列转换模式

CONT=1，为序列转换模式：

- 软件触发：对 ADCS 写 1，可触发一次序列的整体转换

#### 7.6.2.1 序列设置

用户可根据需求灵活设置序列的起始采样通道及序列长度，序列分段方式如下：

- 提供 16 个通道设置项 DS<sub>n</sub>[4:0]，n=0~15，所有出现在序列里的通道均由 DS<sub>n</sub> 设定；
- SQSTR0[3:0]定义序列的起始位置；
- SQCNT0[3:0]定义序列的采样数量。

例如：

设定 DS <sub>n</sub> , n=0~15 右侧序列可选通道均出自此 DS <sub>n</sub> 配置		SQSTR0[3:0]=0, 序列起始: DS0 SQCNT0[3:0]=13, 序列长度: 14			SQSTR0[3:0]=4, 序列起始: DS4 SQCNT0[3:0]=15, 序列长度: 16		
DS <sub>n</sub>	设定通道	顺序	DS <sub>n</sub>	采样通道	顺序	DS <sub>n</sub>	采样通道
DS0	AIN0	1	DS0	AIN0	13	DS0	AIN0
DS1	AIN1	2	DS1	AIN1	14	DS1	AIN1
DS2	AIN6	3	DS2	AIN6	15	DS2	AIN6
DS3	AIN7	4	DS3	AIN7	16	DS3	AIN7
DS4	AIN4	5	DS4	AIN4	1	DS4	AIN4
DS5	AIN5	6	DS5	AIN5	2	DS5	AIN5
DS6	AIN14	7	DS6	AIN14	3	DS6	AIN14
DS7	AIN15	8	DS7	AIN15	4	DS7	AIN15
DS8	AIN8	9	DS8	AIN8	5	DS8	AIN8
DS9	AIN9	10	DS9	AIN9	6	DS9	AIN9
DS10	AIN10	11	DS10	AIN10	7	DS10	AIN10
DS11	AIN11	12	DS11	AIN11	8	DS11	AIN11
DS12	AIN12	13	DS12	AIN12	9	DS12	AIN12
DS13	AIN13	14	DS13	AIN13	10	DS13	AIN13
DS14	1/4 V <sub>DD</sub>	\	DS14	1/4 V <sub>DD</sub>	11	DS14	1/4 V <sub>DD</sub>
DS15	温度采样	\	DS15	温度采样	12	DS15	温度采样

SQSTR0[3:0]=6, 序列起始: DS6 SQCNT0[3:0]=7, 序列长度: 8			SQSTR0[3:0]=12, 序列起始: DS12 SQCNT0[3:0]=5, 序列长度: 6			SQSTR0[3:0]=4, 序列起始: DS4 SQCNT0[3:0]=0, 序列长度: 1		
顺序	DS <sub>n</sub>	采样通道	顺序	DS <sub>n</sub>	采样通道	顺序	DS <sub>n</sub>	采样通道
\	DS0	AIN0	5	DS0	AIN0	\	DS0	AIN0
\	DS1	AIN1	6	DS1	AIN1	\	DS1	AIN1
\	DS2	AIN6	\	DS2	AIN6	\	DS2	AIN6
\	DS3	AIN7	\	DS3	AIN7	\	DS3	AIN7
\	DS4	AIN4	\	DS4	AIN4	1	DS4	AIN4
\	DS5	AIN5	\	DS5	AIN5	\	DS5	AIN5
1	DS6	AIN14	\	DS6	AIN14	\	DS6	AIN14
2	DS7	AIN15	\	DS7	AIN15	\	DS7	AIN15
3	DS8	AIN8	\	DS8	AIN8	\	DS8	AIN8
4	DS9	AIN9	\	DS9	AIN9	\	DS9	AIN9
5	DS10	AIN10	\	DS10	AIN10	\	DS10	AIN10
6	DS11	AIN11	\	DS11	AIN11	\	DS11	AIN11
7	DS12	AIN12	1	DS12	AIN12	\	DS12	AIN12
8	DS13	AIN13	2	DS13	AIN13	\	DS13	AIN13
\	DS14	1/4 V <sub>DD</sub>	3	DS14	1/4 V <sub>DD</sub>	\	DS14	1/4 V <sub>DD</sub>
\	DS15	温度采样	4	DS15	温度采样	\	DS15	温度采样

## 7.7 ADC 溢出

如果转换后的数据未由 CPU 或 DMA 及时读取，在新转换生成数据之前，会由溢出标志（OVERRUN）指示数据溢出事件。

发生溢出时，ADC 会保持工作状态并可继续进行转换。但 OVERRUN 标志将由硬件置 1，ADCV 的值会被

最新一次的转换结果覆盖，之前未读取的数据会丢失。

OVERRUN 标志在发生溢出时由硬件置 1，读取 ADCV 后自动清 0。

## 7.8 ADC 与 DMA 控制器配合使用

选择 DMA 其中一个通道的 REQSRC[5:0]=59，即选择该 DMA 通道的请求源为 ADC，同时将 ADC\_IDE 寄存器中的 DMAEN 位置 1，单采样模式下，每次 ADC 转换完成会生成 DMA 请求。在启动 DMA 及 ADC 后，DMA 便可将转换的数据从 ADCV 寄存器传输到用软件选择的目标位置。

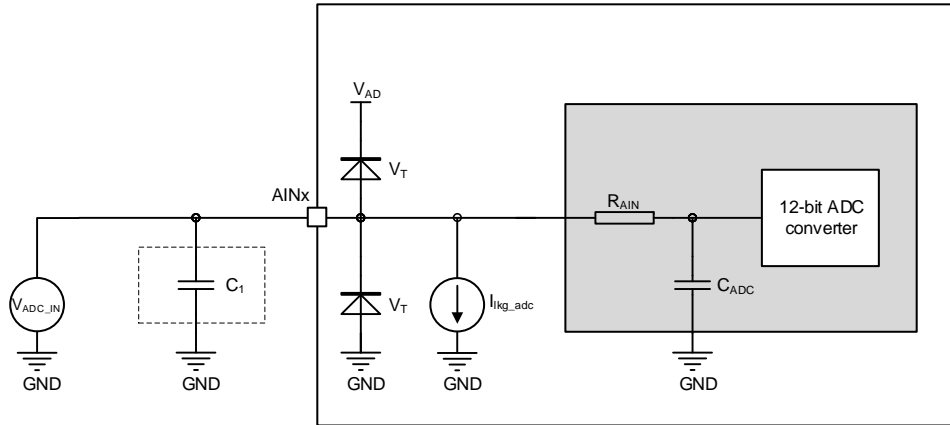
当 DMA 无法及时处理 DMA 传输请求，ADC 将产生溢出（OVERRUN=1），但不会影响到 DMA 传输请求，用户可以读取 RAM 区域的 ADCV 值，判断最高位是否为 1 来查看是否溢出。

## 7.9 ADC 转换步骤

用户实际进行 ADC 转换所需要的操作步骤如下：

- ① 设定 ADC 输入管脚；（设定 AINx 对应的位为 ADC 输入，通常 ADC 管脚会预先固定）；
- ② 通过 REFSEL 位设定 ADC 基准源，若选择 VREF 则需额外设定 VREF 的基准值
- ③ ADCEN 写 1，开启 ADC 模块电源；
- ④ 设置 ADCISA[4:0]，选择空闲及手动触发采样情况下选中的通道
- ⑤ 通过 UPTH[11:0]与 DOWTH[11:0]位设置 ADC 转换值得上下阈值，若 ADC 转换结果值超过阈值，则会置起对应标志位；用户还可通过 ADC\_TH\_CFG 寄存器自由设置通道是否进行阈值判断。
- ⑥ 选择单次转换或序列转换，若选择单次转换，则设置 CONT 为 0，并对 ADCS 写 1 以触发 ADCISA 选中的通道 ADC 转换。
- ⑦ 若选择序列转换，则需事先对 ADC\_SQ0 寄存器设置顺序，并通过 ADC\_SQCNT 寄存器的 SQSTR0 设置序列采样起始位置与 SQCNT0 设置采样个数。要开始序列转换，设置 CONT 为 1，且对 ADCS 写 1 即可开始一次序列的转换，转换将按照序列选中的有效 DS<sub>n</sub> 编号从小到大的顺序进行采样及转换。
- ⑧ ADCIF 置起，则说明一次转换完成，如果 ADC 中断使能且 EOCIE 使能，则会进入一次转换完成中断，用户需要软件清除 ADCIF 标志。
- ⑨ EOSIF0 置起，则说明对应序列采样及转换完成，如果 ADC 中断使能且 EOSIE0 使能，则会进入序列采样及转换完成中断，用户需软件清除 EOSIF0 标志。
- ⑩ 采样模式下采样通道对应转换结果会存放在 ADCVA[11:0]中，若不及时读取 ADCV 寄存器，下一次转换结果将会覆盖当前转换结果，且将 OVERRUN 位置 1，表明转换结果溢出；转换结果溢出不会影响采样
- ⑪ 若转换的 ADC 转换结果 ADCV 寄存器后，在 OVERRUN 位置 ADCVA[11:0]后，会将转换结果与上下阈值进行比较，若超过阈值则会置起 UPTHIF 上阈值溢出标志位或 DOWTHIF 下阈值溢出标志位；如果 ADC 中断使能且 UPTHIE/DOWTHIE 使能，则会进入对应阈值溢出中断
- ⑫ 可通过 DMA 传输转换数据。

## 7.10 ADC 连接电路图



说明:

- C<sub>1</sub> 为外接 0.01μF 电容，建议用户增加此电容以提升 ADC 性能;

## 7.11 ADC 中断

SC32F15G 系列的 ADC 在转换完成后，ADCIF 将置起，如果 ADC\_IDE.INTEN=1，将产生中断。每段序列也均有各自对应的转换完成中断开关及标志位。

中断事件	中断请求控制位	事件标志位	中断使能子开关
ADC 转换完成中断请求	ADC_IDE->INTEN	ADCIF	EOCIE
序列采样及转换完成中断请求		EOSIF0	EOSIE0
下阈值溢出中断请求		DOWTHIF	DOWTHIE
上阈值溢出中断请求		UPTHIF	UPTHIE

## 7.12 ADC 寄存器

### 7.12.1 ADC 相关寄存器表

#### 7.12.1.1 ADC 控制寄存器 ADC\_CON

寄存器	读/写	说明	复位值	上电初始值
ADC_CON	读/写	ADC 控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24	
-	-	-	-	-	-	-	-	
23	22	21	20	19	18	17	16	
-	-	-	ADCISA[4:0]				-	-
15	14	13	12	11	10	9	8	
-	-	-	-	-	LOWSP[2:0]			

7	6	5	4	3	2	1	0
ADCEN	-	CONT	REFSEL	-	-	-	ADCS

位编号	位符号	说明																																						
20~16	ADCISA[4:0]	<p>通道选择值同 DS<sub>n</sub> 定义</p> <table border="1"> <tr><td>00000</td><td>ADC0 / OP0</td></tr> <tr><td>00001</td><td>ADC1 / OP1</td></tr> <tr><td>00010</td><td>ADC2</td></tr> <tr><td>00011</td><td>ADC3</td></tr> <tr><td>00100</td><td>ADC4</td></tr> <tr><td>00101</td><td>ADC5</td></tr> <tr><td>00110</td><td>ADC6 / OP2</td></tr> <tr><td>00111</td><td>ADC7</td></tr> <tr><td>01000</td><td>ADC8</td></tr> <tr><td>01001</td><td>ADC9</td></tr> <tr><td>01010</td><td>ADC10</td></tr> <tr><td>01011</td><td>ADC11</td></tr> <tr><td>01100</td><td>ADC12</td></tr> <tr><td>01101</td><td>ADC13</td></tr> <tr><td>01110</td><td>ADC14</td></tr> <tr><td>01111</td><td>ADC15</td></tr> <tr><td>10000</td><td>ADC 输入为 1/4 V<sub>DD</sub></td></tr> <tr><td>10001</td><td>温度采样</td></tr> <tr><td>其它</td><td>保留</td></tr> </table>	00000	ADC0 / OP0	00001	ADC1 / OP1	00010	ADC2	00011	ADC3	00100	ADC4	00101	ADC5	00110	ADC6 / OP2	00111	ADC7	01000	ADC8	01001	ADC9	01010	ADC10	01011	ADC11	01100	ADC12	01101	ADC13	01110	ADC14	01111	ADC15	10000	ADC 输入为 1/4 V <sub>DD</sub>	10001	温度采样	其它	保留
00000	ADC0 / OP0																																							
00001	ADC1 / OP1																																							
00010	ADC2																																							
00011	ADC3																																							
00100	ADC4																																							
00101	ADC5																																							
00110	ADC6 / OP2																																							
00111	ADC7																																							
01000	ADC8																																							
01001	ADC9																																							
01010	ADC10																																							
01011	ADC11																																							
01100	ADC12																																							
01101	ADC13																																							
01110	ADC14																																							
01111	ADC15																																							
10000	ADC 输入为 1/4 V <sub>DD</sub>																																							
10001	温度采样																																							
其它	保留																																							
10~8	LOWSP[2:0]	<p>ADC 采样周期数选择</p> <p>000: 采样时间为 3 个系统时钟, (约 42ns @ f<sub>PCLK2</sub> =72MHz)</p> <p>001: 采样时间约 6 个系统时钟, (约 83ns @ f<sub>PCLK2</sub> =72MHz)</p> <p>010: 采样时间约 9 个系统时钟, (约 125ns @ f<sub>PCLK2</sub> =72MHz)</p> <p>011: 采样时间约 15 个系统时钟, (约 208ns @ f<sub>PCLK2</sub> =72MHz)</p> <p>100: 采样时间为 30 个系统时钟, (约 417ns @ f<sub>PCLK2</sub> =72MHz)</p> <p>101: 采样时间约 60 个系统时钟, (约 833ns @ f<sub>PCLK2</sub> =72MHz)</p> <p>110: 采样时间约 120 个系统时钟, (约 1667ns @ f<sub>PCLK2</sub> =72MHz)</p> <p>111: 采样时间约 480 个系统时钟, (约 6667ns @ f<sub>PCLK2</sub> =72MHz)</p> <p>说明: ADC 从采样到完成转换的总时间计算方式如下</p> $T_{ADC} = \text{采样时间} + \text{转换时间}$ <p>其中, ADC 转换时间约为 404ns</p>																																						
7	ADCEN	<p>启动 ADC 的电源</p> <p>0: 关闭 ADC 模块电源</p> <p>1: 开启 ADC 模块电源</p>																																						
5	CONT	<p>单次/序列转换模式选择位</p> <p>此位由软件置 1 和清零。</p> <p>0: 单次模式, 此模式下 ADC 采样及只能通过软件触发: 对 ADCS 写 1, 一次采样一个 ADCISA 选中的通道</p> <p>1: 序列模式:</p> <p>*软件触发: 对 ADCS 写 1, 可触发一次序列的整体转换</p>																																						
4	REFSEL	<p>ADC 模块基准源选择位:</p> <p>0: 模块基准源为 VDD</p> <p>1: 模块基准源为 VREF</p>																																						
0	ADCS	<p>ADC 开始触发控制 (ADC Start)</p> <p>对此位写 1, 开始做一次 ADC 的转换, 即该位只是 ADC 转换的触发信号。此位只可写入 1 有效。</p> <p>注意: 对 ADCS 写 1 后, 到中断标志 ADCIF 置起前不要对 ADCCON 寄存器进行写操作</p>																																						

位编号	位符号	说明
31~21 15~11 6 3~1	-	保留

### 7.12.1.2 ADC 标志位状态寄存器 ADC\_STS

寄存器	读/写	说明	复位值	上电初始值
ADC_STS	读/写	ADC 标志位状态寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	DOWTHIF	UPTHIF
7	6	5	4	3	2	1	0
BUSY	-	-	-	-	-	EOSIF0	ADCIF

位编号	位符号	说明
9	DOWTHIF	下阈值溢出标志位： 0: ADCVA[11:0] ≥ DOWTH[11:0] 1: ADCVA[11:0] < DOWTH[11:0]
8	UPTHIF	上阈值溢出标志位： 0: ADCVA[11:0] ≤ UPTH[11:0] 1: ADCVA[11:0] > UPTH[11:0]
7	BUSY	硬件触发 ADC 状态位 0: ADC 空闲 1: 硬件触发的 ADC 序列正在采样/转换  ADC 采样/转换进行中，BUSY 清零前对 ADCCON 寄存器进行的所有写操作均无效
1	EOSIF0	序列采样及转换完成中断标志位 该位由硬件置 1，通过软件写 1 清 0。 0: 序列采样及转换未完成 1: 序列采样及转换完成 当序列最后一个通道的转换结果出现在 ADCV 寄存器时，会通过硬件将该位置 1；若此时 ADC_IDE.EOSIE0=1，将产生中断。
0	ADCIF	ADC 中断请求标志位 该位由硬件置 1，通过软件写 1 清 0。 0: 无 ADC 通道转换完成 1: 有 ADC 通道转换完成 通道的每次转换结束，新数据结果出现在 ADCV 寄存器时，会通过硬件将该位置 1，如果 ADC_IDE.EOCIE=1，将产生中断。
31~10 6~2	-	保留

### 7.12.1.3 ADC 转换数值寄存器 ADCV

寄存器	读/写	说明	复位值	上电初始值
ADCV	读/写	ADC 转换数值寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
31	OVERRUN	该位只读 无法及时处理请求而发生溢出时由硬件置 1，读取 ADCV 后自动清零 <b>注意：</b> 1.上一次的 ADC 转换结果将被新的 ADC 转换结果覆盖 2.发生溢出后，DMA 传输不停止 3.不论 DMA 还是 CPU，只要读取了 ADCV 寄存器，OVERRUN 位就会清零
11~0	ADCVA[11:0]	ADC 转换结果 当前采样通道对应转换结果，转换结果存放在寄存器低 11 位，即 ADCVA 中
30~12	-	保留

#### 7.12.1.4 ADC 端口设置寄存器 ADC\_CFG

寄存器	读/写	说明	复位值	上电初始值
ADC_CFG	读/写	ADC 端口设置寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
AIN15	AIN14	AIN13	AIN12	AIN11	AIN10	AIN9	AIN8
7	6	5	4	3	2	1	0
AIN7	AIN6	AIN5	AIN4	AIN3	AIN2	AIN1	AIN0

位编号	位符号	说明
15~0	AINx	ADC 端口设置寄存器 0: AINx 对应端口不可作为 ADC 输入通道 1: AINx 对应端口可作为 ADC 输入通道，当 ADCISA[4:0]选择 AINx 作为 ADC 输入通道时，AINx 对应端口的上拉电阻将自动移除。
31~16	-	保留

#### 7.12.1.5 ADC 通道阈值使能寄存器 ADC\_TH\_CFG

寄存器	读/写	说明	复位值	上电初始值
ADC_TH_CFG	读/写	ADC 通道阈值使能寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
AIN15	AIN14	AIN13	AIN12	AIN11	AIN10	AIN9	AIN8
7	6	5	4	3	2	1	0
AIN7	AIN6	AIN5	AIN4	AIN3	AIN2	AIN1	AIN0

位编号	位符号	说明
15~0	AINx	ADC 通道阈值使能控制位，x=0~15 0: AINx 通道禁止使用阈值功能 1: AINx 通道需要进行阈值判断
31~16	-	保留

### 7.12.1.6 ADC 阈值下限设置寄存器 ADC\_DOWTH

寄存器	读/写	说明	复位值	上电初始值
ADC_DOWTH	读/写	ADC 阈值下限设置寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
11~0	DOWTH[11:0]	ADC 阈值下限设置位 当前转换的 ADCVA[11:0] < DOWTH[11:0] 设置的值时，DOWTHIF 标志位置起，如果此时 DOWTHIE 使能，可以触发 ADC 下阈值比较中断。
31~12	-	保留

### 7.12.1.7 ADC 阈值上限设置寄存器 ADC\_UPTH

寄存器	读/写	说明	复位值	上电初始值
ADC_UPTH	读/写	ADC 阈值上限设置寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
11~0	UPTH[11:0]	ADC 阈值上限设置位 当前转换的 ADCVA[11:0]>UPTH[11:0] 设置的值时，UPTHIF 标志位置起，如果此时 UPTHIE 使能，可以触发 ADC 上阈值比较中断。
31~12	-	保留

### 7.12.1.8 ADC 的中断使能及 DMA 控制寄存器 ADC\_IDE

寄存器	读/写	说明	复位值	上电初始值
ADC_IDE	读/写	ADC 的中断使能及 DMA 控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	DOWTHIE	UPTHIE
7	6	5	4	3	2	1	0
INTEN	DMAEN	-	-	-	-	EOSIE0	EOCIE

位编号	位符号	说明
9	DOWTHIE	下阈值溢出中断使能位： 0: DOWTHIF 置起时不允许产生中断 1: DOWTHIF 置起时，产生中断
8	UPTHIE	上阈值溢出中断使能位： 0: UPTHIF 置起时不允许产生中断 1: UPTHIF 置起时，产生中断
7	INTEN	中断请求 CPU 的使能控制位 0: 禁止中断请求 1: 使能中断请求
6	DMAEN	直接存储器访问使能 由软件置 1 和清零，用于使能 DMA 请求的生成。这样便可使用 DMA 控制器自动管理转换的数据。 0: 禁止 DMA 1: 使能 DMA 注：确保当前未进行任何转换，才允许通过软件对此位执行写操作。
1	EOSIE0	序列采样及转换完成中断使能位 0: EOSIF0 置起时，不允许产生中断 1: EOSIF0 置起时，产生中断
0	EOCIE	ADC（每一次）转换完成中断使能位 0: ADCIF 置起时，不允许产生中断 1: ADCIF 置起时，产生中断
31~10 5~2	-	保留

### 7.12.1.9 ADC 序列通道设置寄存器 ADC\_SQCNT

寄存器	读/写	说明	复位值	上电初始值
ADC_SQCNT	读/写	ADC 序列通道设置寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
SQSTR0[3:0]				SQCNT0[3:0]			

位编号	位符号	说明
7~4	SQSTR0[3:0]	序列的起始位置设置位 用于设定序列每次启动时的起始 DS <sub>n</sub> , n=0~15
3~0	SQCNT0[3:0]	序列的采样个数设置位 采样序列的采样个数=SQCNT0[3:0]+1 即：序列最多可支持 16 个采样
31~8	-	保留

### 7.12.1.10 ADC 序列设置寄存器 ADC\_SQ0

寄存器	读/写	说明	复位值	上电初始值
ADC_SQ0	读/写	ADC 序列设置寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24	
-	-	-	DS15[4:0]				-	-
23	22	21	20	19	18	17	16	
-	-	-	DS14[4:0]				-	-
15	14	13	12	11	10	9	8	
-	-	-	DS13[4:0]				-	-
7	6	5	4	3	2	1	0	
-	-	-	DS12[4:0]				-	-
31	30	29	28	27	26	25	24	
-	-	-	DS11[4:0]				-	-
23	22	21	20	19	18	17	16	
-	-	-	DS10[4:0]				-	-
15	14	13	12	11	10	9	8	
-	-	-	DS9[4:0]				-	-
7	6	5	4	3	2	1	0	
-	-	-	DS8[4:0]				-	-
31	30	29	28	27	26	25	24	
-	-	-	DS7[4:0]				-	-
23	22	21	20	19	18	17	16	
-	-	-	DS6[4:0]				-	-
15	14	13	12	11	10	9	8	
-	-	-	DS5[4:0]				-	-
7	6	5	4	3	2	1	0	
-	-	-	DS4[4:0]				-	-
31	30	29	28	27	26	25	24	
-	-	-	DS3[4:0]				-	-
23	22	21	20	19	18	17	16	
-	-	-	DS2[4:0]				-	-
15	14	13	12	11	10	9	8	
-	-	-	DS1[4:0]				-	-
7	6	5	4	3	2	1	0	
-	-	-	DS0[4:0]				-	-

位编号	位符号	说明	
28~24 20~16 12~8 4~0	DSn[4:0]	DSn[4:0]: n=0~15, ADC 采样序列信号选择	
		00000	ADC0 / OP0
		00001	ADC1 / OP1
		00010	ADC2
		00011	ADC3
		00100	ADC4
		00101	ADC5
		00110	ADC6 / OP2
		00111	ADC7
		01000	ADC8
		01001	ADC9
		01010	ADC10
		01011	ADC11
		01100	ADC12
		01101	ADC13
01110	ADC14		
01111	ADC15		
	10000	ADC 输入为 1/4 V <sub>DD</sub>	
	10001	温度采样	
	其它	保留	
31~29 23~21 15~13 7~5	-	保留	

### 7.12.2 ADC 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值
ADC 基地址: 0x4002_2100					
ADC_CON	0x00	读/写	ADC 控制寄存器	0x0000_0000	0x0000_0000
ADC_STS	0x04	读/写	ADC 标志位状态寄存器	0x0000_0000	0x0000_0000
ADCV	0x08	读/写	ADC 转换数值寄存器	0x0000_0000	0x0000_0000
ADC_CFG	0x0C	读/写	ADC 端口设置寄存器	0x0000_0000	0x0000_0000
ADC_TH_CFG	0x10	读/写	ADC 通道阈值使能寄存器	0x0000_0000	0x0000_0000
ADC_LOWTH	0x14	读/写	ADC 阈值下限设置寄存器	0x0000_0000	0x0000_0000
ADC_UPTH	0x18	读/写	ADC 阈值上限设置寄存器	0x0000_0000	0x0000_0000
ADC_IDE	0x1C	读/写	ADC 的中断使能及 DMA 控制寄存器	0x0000_0000	0x0000_0000
ADC_SQCNT	0x24	读/写	ADC 序列通道设置寄存器	0x0000_0000	0x0000_0000
ADC_SQ0	0x28	读/写	ADC 序列设置寄存器	0x0000_0000	0x0000_0000

## 8 内部基准源 (VREF)

### 8.1 概述

SC32F15G 系列内部集成一个独立的内部基准模块 (VREF)，可作为多个外设的基准源。

### 8.2 内部基准源模块配置

系统模拟电路基准模块有四种配置方式：

- VREFCFG1=0、VREFCFG0=0，Vref PIN 端口不使能、内建基准模块关闭；
- VREFCFG1=0、VREFCFG0=1，模拟电路使用内建基准，Vref 电压为 VREFS[1:0]选定项；
- VREFCFG1=1、VREFCFG0=0，模拟电路使用外接基准，Vref 由外部 Vref PIN 输入；
- VREFCFG1=1、VREFCFG0=1，模拟电路使用内建基准，Vref 电压为 VREFS[1:0]选定项。

### 8.3 内部基准源输出

内部基准源模块使能后，VREF 可作为 ADC/DAC/OP/CMP 的基准选择，也可二分之一分压后通过 VMID 引脚输出。

具体设置方式如下：

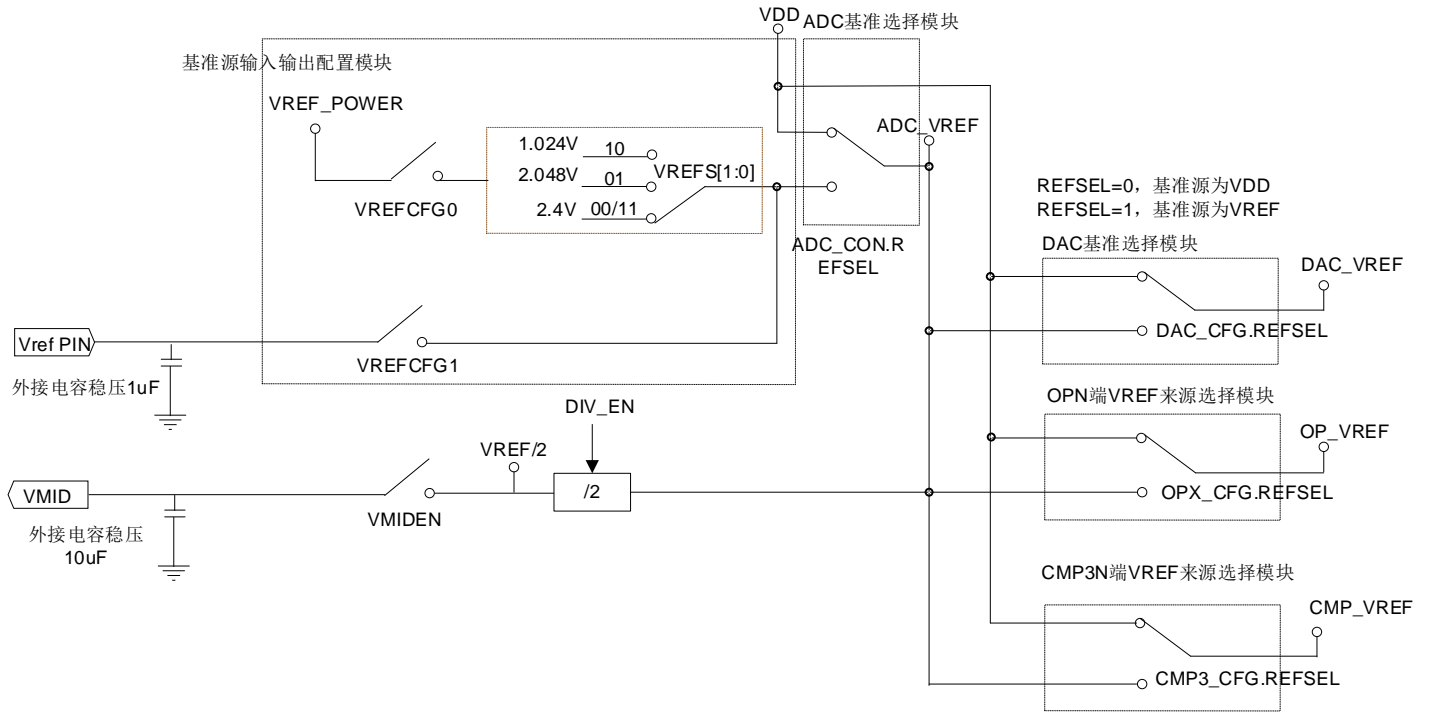
- ADC/DAC/OP/CMP 模块默认基准源为 VDD，当选择 VREF 作为基准源时，需使能相关外设寄存器内的模块基准源选择位 REFSEL。例如：当选择 VREF 作为 DAC 模块基准源时，需使能 DAC 配置寄存器 DAC\_CFG 的 DAC 模块基准源选择位 REFSEL。

注意：

1. 若只开启 DAC/OP/CMP，外设基准源的选择不受影响。
  2. 若同时开启 ADC 和其他外设 (DAC/OP/CMP)，当 ADC 的基准源为 VDD 时，其他外设基准源只能选择 VDD；若要修改其他外设基准源为 VREF，需先将 ADC 的基准源设置为 VREF！
- 当选择 VREF/2 通过 VMID 引脚输出时，需先将内部基准分压电路使能位 DIV\_EN 置 1，使 VREF/2 点输出电压为 VREF 的一半（可作为 OP 输入差分模式的偏置电压），再将 VMID 端口使能位 VMIDEN 置 1。

### 8.4 内部基准源功能框图

Vref PIN 可作为输入输出引脚，VMID 只能作为输出引脚。



## 8.5 VREF 寄存器

### 8.5.1 VREF 相关寄存器表

#### 8.5.1.1 VREF 模块配置寄存器 VREF\_CFG

寄存器	读/写	说明	复位值	上电初始值
VREF_CFG	读/写	VREF 模块配置寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
-	DIV_EN	VMIDEN	-	VREFS[1:0]		VREFCFG1	VREFCFG0

位编号	位符号	说明
6	DIV_EN	内部基准分压电路使能位 0: 不使能, VREF/2 点无输出 1: 使能, VREF/2 点输出电压为 VREF 的一半
5	VMIDEN	VMID 端口使能位 0: VMID 所在端口为其它复用功能 1: VMID 所在端口输出 VREF/2
3~2	VREFS[1:0]	系统模拟电路 Vref 选择 00: 保留 (默认接 2.4V) 01: 设定 ADC 的 Vref 为内部准确的 2.048V

位编号	位符号	说明
		10: 设定 ADC 的 Vref 为内部准确的 1.024V 11: 设定 ADC 的 Vref 为内部准确的 2.4V
1~0	VREFCFG1、VREFCFG0	VREFCFG[1:0]系统模拟电路基准模块 VREF 设置位 00: Vref PIN 端口不使能、内建基准模块关闭 01: 模拟电路使用内建基准, Vref 电压为 VREFS[1:0]选定项 10: 模拟电路使用外接基准, Vref 由外部 Vref PIN 输入 11: 模拟电路使用内建基准, Vref 电压为 VREFS[1:0]选定项
31~7 4	-	保留

### 8.5.2 VREF 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值
VREF 基地址: 0x4002_2190					
VREF_CFG	0x0C	读/写	VREF 模块配置寄存器	0x0000_0000	0x0000_0000

## 9 数模转换器 (DAC)

### 9.1 概述

SC32F15G 内部集成一个独立的 10 Bits 数模转换器 (DAC)。此 DAC 有两个独立的输出端口 DACOUT0 和 DACOUT1，DAC 也可在芯片内部输出到 OP1/OP2 的反相端。

### 9.2 时钟源

SC32F15G 系列的 DAC 时钟源仅来自 PCLK2

### 9.3 特性

- 基准源可选择 VDD 或 VREF
- 输出方式有两种：
  - 两个独立的输出端口 DACOUT0 或 DACOUT1 输出
  - 在芯片内部输出到 OP1/OP2 的反相端
  - 在芯片内部输出到 CMP0/1/2/3 的负向输入端

### 9.4 DAC 寄存器

#### 9.4.1 DAC 相关寄存器表

##### 9.4.1.1 DAC 状态寄存器 DAC\_STS

寄存器	读/写	说明	复位值	上电初始值
DAC_STS	读/写	DAC 状态寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
0	STA	DAC 转换状态位 此位为状态位，由硬件置起或清零 0: DAC 模块空闲/已转换完成； 1: DAC 模块转换中
31~1	-	保留

### 9.4.1.2 DAC 转换寄存器 DAC\_IN

寄存器	读/写	说明	复位值	上电初始值
DAC_IN	读/写	DAC 转换寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
9~0	DACV[9:0]	DAC 输出电压： $VDACOUT=(Vref / 1024) * DACV[9:0]$ 注意：转换值写入此寄存器后即生效
31~10	-	保留

### 9.4.1.3 DAC 配置寄存器 DAC\_CFG

寄存器	读/写	说明	复位值	上电初始值
DAC_CFG	读/写	DAC 配置寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
-	-	-	-	REFSEL	DACOUT1	DACOUT0	DACEN

位编号	位符号	说明
3	REFSEL	DAC 模块基准源选择位 0: 模块基准源为 VDD 1: 模块基准源为 VREF
2	DACOUT1	DACOUT1 端口使能位 0: DACOUT1 所在端口为其它复用功能 1: DACOUT1 所在端口输出 DAC 当前转换电压
1	DACOUT0	DACOUT0 端口使能位 0: DACOUT0 所在端口为其它复用功能 1: DACOUT0 所在端口输出 DAC 当前转换电压
0	DACEN	DAC 使能控制位 0: 关闭 1: 使能
31~4	-	保留

#### 9.4.2 DAC 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值
DAC 基地址: 0x4002_2190					
DAC_STS	0x00	读/写	DAC 状态寄存器	0x0000_0000	0x0000_0000
DAC_IN	0x04	读/写	DAC 转换寄存器	0x0000_0000	0x0000_0000
DAC_CFG	0x08	读/写	DAC 配置寄存器	0x0000_0000	0x0000_0000

## 10 温度传感器

### 10.1 概述

SC32F15G 内建一个温度传感器，可通过 ADC 电路测量温度传感器电压。

### 10.2 温度传感器操作步骤

使用温度传感器时，ADC 参考电压选择内部 2.4V 作为参考，温度传感器每增加 1℃，ADC 转换值会增加固定值。赛元出厂时已将每颗芯片 25℃对应的 ADC 转换结果写入到对应地址中。

用户使用温度传感器的操作步骤如下：

- ① 设定 ADC 参考电压 Vref 为内部 2.4V 基准源，设定 ADC 采样周期，建议选择 60 个以上采样时钟，之后开启 ADC 模块电源；
- ② 选择 ADC 输入通道为温度传感器通道；
- ③ 使能温度传感器，TS\_EN 写 1；
- ④ 延时 20μs
- ⑤ TS\_CHOP 写 0，启动 ADC 转换，一次转换完成，记录转换值 ADC<sub>Value1</sub>；
- ⑥ TS\_CHOP 写 1，启动 ADC 转换，一次转换完成，记录转换值 ADC<sub>Value2</sub>；
- ⑦ 将两次转换值求平均：

$$ADC_{Value} = \frac{(ADC_{Value1} + ADC_{Value2})}{2}$$

- ⑧ 从对应地址读取出厂时写入的 25 摄氏度 ADC 转换值 ADC<sub>ValueTest</sub>；
- ⑨ 代入公式计算得到当前温度：

$$Temperature = 25^{\circ}C + \frac{(ADC_{Value} - ADC_{ValueTest})}{8.53}$$

用户若需获取更多温度传感器相关信息，请参考《赛元 SC32F1XXX 系列 MCU 应用指南 V1.4》

### 10.3 温度传感器寄存器

#### 10.3.1 温度传感器相关寄存器表

##### 10.3.1.1 温度传感器设置寄存器 TS\_CFG

寄存器	读/写	说明	复位值	上电初始值
TS_CFG	读/写	温度传感器设置寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
7	TS_EN	温度传感器使能控制位 0: 关闭温度传感器 1: 使能温度传感器
0	TS_CHOP	温度传感器的抵消 offset 的应用控制位 TS_CHOP 写 0 后启动一次 ADC 转换得到一个数值，TS_CHOP 再写 1 后启动一次 ADC 转换得到第二个数值，两个数做平均得到最终数
31~8 6~1	-	保留

### 10.3.2 温度传感器寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值
温度传感器基地址: 0x4002_21E0					
TS_CFG	0x00	读/写	温度传感器设置寄存器	0x0000_0000	0x0000_0000

## 11 运放及可编程增益放大器（OP）

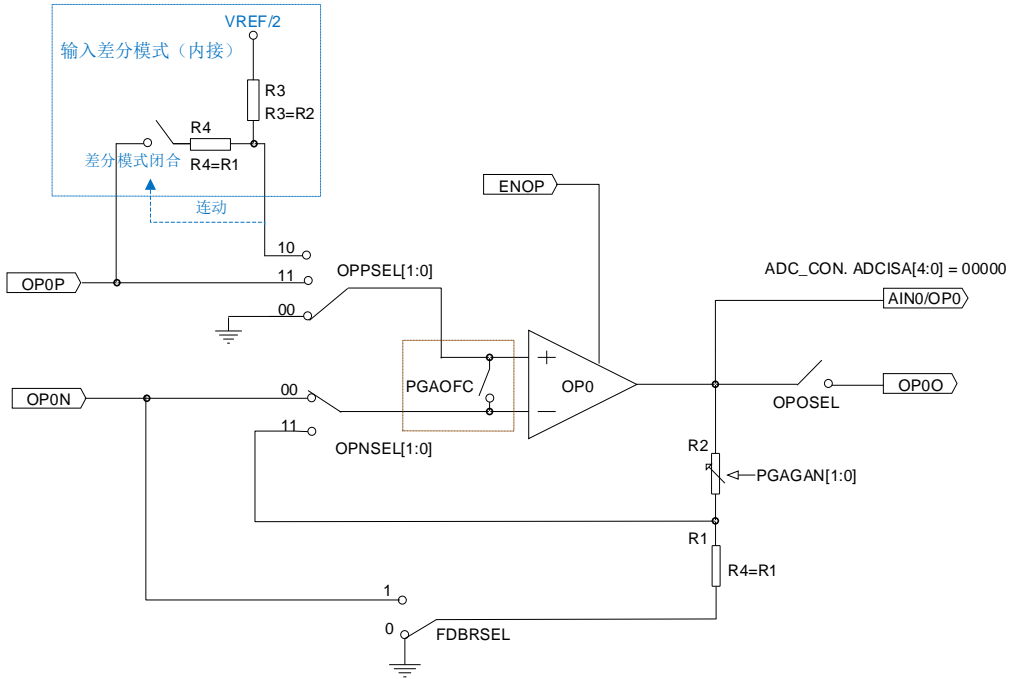
### 11.1 概述

SC32F15G 系列内建三个独立的 Rail-to-Rail 可配增益放大器：OP0/OP1/OP2。

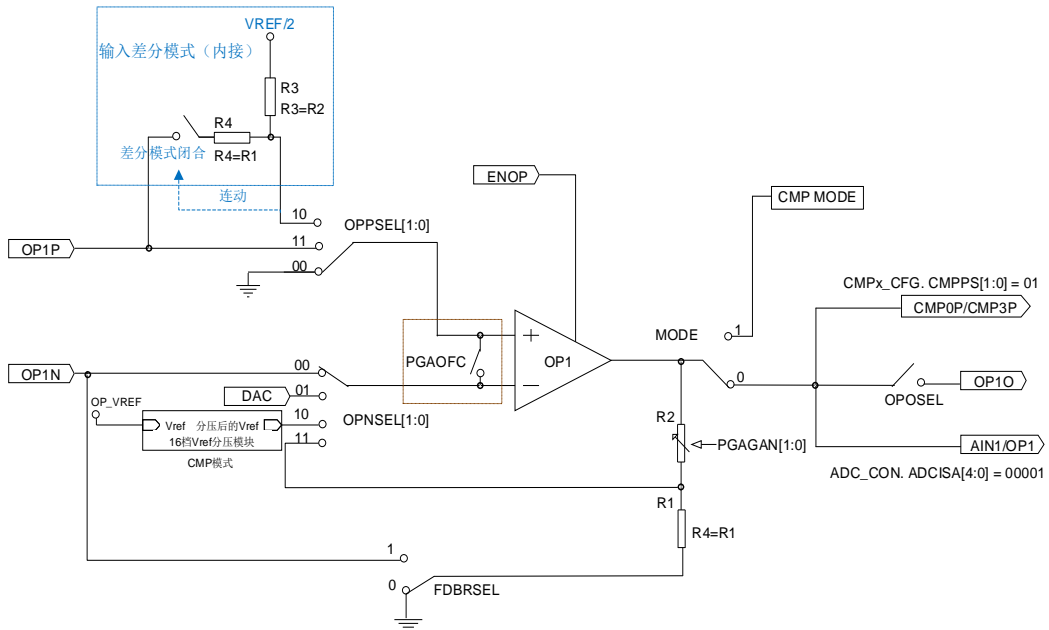
### 11.2 特性

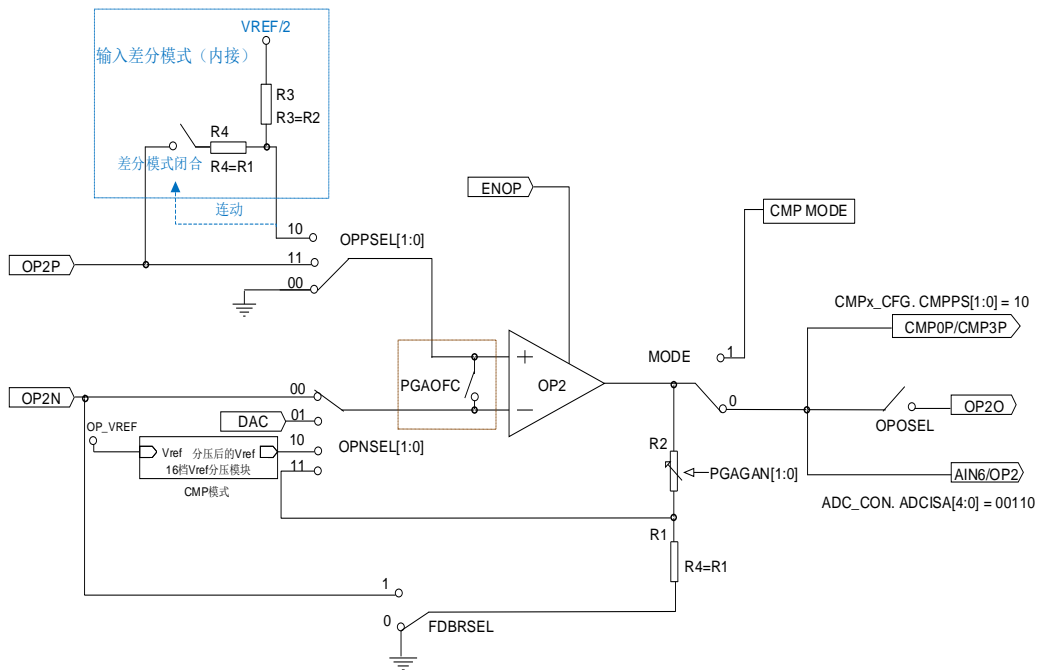
- 三个 OP 均可配置为 PGA 模式，单端模式增益如下
  - 同相输入增益：4/8/16/32
  - 反相输入增益：3/7/15/31
- 三个 OP 的同相端、反相端以及输出端均有独立的对外端口
- 三个 OP 的输出分别与三路 ADC 通道复用，输出结果可通过 ADC 结果寄存器读出
- OP1/OP2 可设置为比较器（CMP）模式：
  - CMP 模式下迟滞电压固定为 10~15mV
  - CMP 模式下的响应时间：典型值 50ns
- OP1/OP2 均可输出到 CMP0 和 CMP3 的正端
- 参数
  - 带宽 10MHz
  - 输入失调电压 $\leq 10\text{mV}$ ，需调零
  - 压摆率 $\geq 10\text{V}/\mu\text{s}$

### 11.3 OP0 框图



### 11.4 OP1 / OP2 框图





## 11.5 OP0 端口选择

### 11.5.1 OP0 精度调整

可通过设置 PGA 输入端 offset 调整控制位  $PGA_{OFC}=1$ ，将 OP 模块的同相端与反相端输入短接来实现精度调整。其他情况下， $PGA_{OFC}$  设置为 0。

### 11.5.2 OP0 同相端输入

OP0 的同相端输入有三种：OP0P 外部引脚、内部 VSS 和输入差分模式，可通过  $OPPSEL[1:0]$  切换选择。当选择差分模式时，需同步使能  $VREF\_CFG.DIV\_EN$ ，偏置电压  $VREF/2$  才有输出。

### 11.5.3 OP0 反相端输入

OP0 的反相端输入有两种：OP0N 外部引脚和内部反馈电阻。选择 OP0N 外部引脚为反相端输入时，需设置 OP0 输入控制位  $OPNSEL[1:0]=00$ ，反馈电阻端连接选择位  $FDBRSEL=1$ ；选择内部反馈电阻为负端输入时，需设置  $OPNSEL[1:0]=11$ ， $FDBRSEL=0$  或 1，并通过内部增益档位选择位  $PGAGAN[1:0]$  进行内部增益档位选择。

### 11.5.4 OP0 输出

OP0 的输出有两种：用于 AD 转换器的模拟输入或者通过 OP0O 外部引脚输出。

具体设置方式如下：

- OP0 通过 OP0O 外部引脚输出时，需设置  $OPOSEL=1$ ；
- OP0 的输出端默认与 ADC 输入相连，通过设置  $ADCISA[4:0]=00000$  选择 OP0 输出作为 ADC 输入，使能 ADC 后，OP 的转换结果可直接在  $ADCV$  寄存器获取。

## 11.6 OP1/2 端口选择

### 11.6.1 OP1/2 精度调整

可通过设置 PGA 输入端 offset 调整控制位 PGOFC=1，将 OP 模块的同相端与反相端输入短接来实现精度调整。其他情况下，PGOFC 设置为 0。

### 11.6.2 OP1/2 同相端输入

OP1/2 的同相端输入有三种：OP1P/OP2P 外部引脚、内部 VSS 和输入差分模式，可通过 OPPSEL[1:0]切换选择。当选择差分模式时，需同步使能 VREF\_CFG. DIV\_EN，偏置电压 VREF/2 才有输出。

### 11.6.3 OP1/2 反相端输入

OP1/2 的反相端输入有四种：OP1N/OP2N 外部引脚、DAC 输出、OPRF[3:0]设定值和内部反馈电阻。

具体设置方式如下：

- 选择 OP1N/OP2N 外部引脚为反相端输入时，需设置 OP1/2 反相端输入控制位 OPNSEL[1:0]=00，反馈电阻端连接选择位 FDBRSEL=1；
- 选择 DAC 为反相端输入时，需使能 DAC 模块，并设置 OP1/2 反相端输入控制位 OPNSEL[1:0]=01；
- 选择 OPRF[3:0]设定值为反相端输入时，需设置 OP1/2 反相端输入控制位 OPNSEL[1:0]=10；
- 选择内部反馈电阻为反相端输入时，需设置 OPNSEL[1:0]=11，并通过内部增益档位选择位 PGAGAN[1:0]进行内部增益档位选择。

### 11.6.4 OP1/2 输出

OP1/2 的输出有三种：AD 转换器的模拟输入、CMP0/CMP3 的正端输入或者通过 OP1O/OP2O 外部引脚输出。

具体设置方式如下：

- OP1/2 输出作 AD 转换器的模拟输入或者 CMP0/CMP3 的正端输入，需设置 MODE=0 使 OP1/2 为运放模式。
- OP1/2 在运放模式下，可输出到外部引脚 OP1O/OP2O，此时需设置 OPOSEL=1。

## 11.7 OP 寄存器

### 11.7.1 OP0 相关寄存器表

#### 11.7.1.1 OP0 控制寄存器 OP0\_CON

寄存器	读/写	说明	复位值	上电初始值
OP0_CON	读/写	OP0 控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
----	----	----	----	----	----	----	----

-	FDBRSEL	-	TRIMOFFSETN[4:0]				
23	22	21	20	19	18	17	16
PGAOFC	-	-	TRIMOFFSETP[4:0]				
15	14	13	12	11	10	9	8
-	-	-	-	-	-	PGAGAN[1:0]	
7	6	5	4	3	2	1	0
ENOP	-	OPPSEL[1:0]		OPNSEL[1:0]		-	OPOSEL

位编号	位符号	说明
30	FDBRSEL	运放模块的反馈电阻 R1 连接选择位 0: 内部接 VSS, 0V 1: OPxN (外部引脚)
28~24	TRIMOFFSETN[4:0]	Trim for NMOS differential pairs 运放的 NMOS 差分 offset 校准值
23	PGAOFC	OP 输入端短接控制位 0: 同相和反相输入端不短接 1: 同相和反相输入端短接
20~16	TRIMOFFSETP[4:0]	Trim for PMOS differential pairs 运放的 PMOS 差分 offset 校准值
9~8	PGAGAN[1:0]	运放 PGA 模式内部增益档位选择 00: 同相 4, 反相 3 01: 同相 8, 反相 7 10: 同相 16, 反相 15 11: 同相 32, 反相 31
7	ENOP	OP0 模块使能位 0: 关闭 OP0 模块电源 1: 使能 OP0 模块电源
5~4	OPPSEL[1:0]	运放同相端输入选择位 00: 内部接 VSS, 0V 10: 输入差分模式, 偏置电压为 VREF/2, 注意此时需同步使能 VREF_CFG.DIV_EN, VREF/2 才有电压输出 11: 选用 OP0P (外部引脚)
3~2	OPNSEL[1:0]	运放反相端输入选择位: 00: 选用 OP0N (外部引脚) 01: 保留 10: 保留 11: 接反馈电阻 R2
0	OPOSEL	运放输出端连接选择位 1: 运放输出连接到 OP0O (外部引脚) 0: 运放输出与 OP0O 的连接断开 说明: 运放的输出始终连接到 ADC 和 CMPxPS 的可选项
31 29 22~21 15~10 6 1	-	保留

## 11.7.2 OP1/2 相关寄存器

### 11.7.2.1 OP1 控制寄存器 OP1\_CON

寄存器	读/写	说明	复位值	上电初始值
OP1_CON	读/写	OP1 控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	FDBRSEL	-	TRIMOFFSETN[4:0]				-
23	22	21	20	19	18	17	16
PGAOFC	-	-	TRIMOFFSETP[4:0]				-
15	14	13	12	11	10	9	8
OPRF[3:0]				MODE	-	PGAGAN[1:0]	
7	6	5	4	3	2	1	0
ENOP	-	OPPSEL[1:0]		OPNSEL[1:0]		-	OPOSEL

位编号	位符号	说明
30	FDBRSEL	运放模块的反馈电阻 R1 连接选择位 0: 内部接 VSS, 0V 1: OP1N (外部引脚)
28~24	TRIMOFFSETN[4:0]	Trim for NMOS differential pairs 运放的 NMOS 差分 offset 校准值
23	PGAOFC	OP 输入端短接控制位 0: 同相和反相输入端不短接 1: 同相和反相输入端短接
20~16	TRIMOFFSETP[4:0]	Trim for PMOS differential pairs 运放的 PMOS 差分 offset 校准值
15~12	OPRF[3:0]	OP 作比较器模式下, 运放反相端输入电压选择位, 当 OPNSEL[1:0]=10 时生效: 0000: 1/16 OPx_VREF 0001: 1/16 OPx_VREF 0010: 2/16 OPx_VREF 0011: 3/16 OPx_VREF 0100: 4/16 OPx_VREF 0101: 5/16 OPx_VREF 0110: 6/16 OPx_VREF 0111: 7/16 OPx_VREF 1000: 8/16 OPx_VREF 1001: 9/16 OPx_VREF 1010: 10/16 OPx_VREF 1011: 11/16 OPx_VREF 1100: 12/16 OPx_VREF 1101: 13/16 OPx_VREF 1110: 14/16 OPx_VREF 1111: 15/16 OPx_VREF
11	MODE	OP1 模式选择位 0: 运放模式 1: 比较器模式
8~9	PGAGAN[1:0]	运放 PGA 模式内部增益档位选择 00: 同相 4, 反相 3 01: 同相 8, 反相 7 10: 同相 16, 反相 15 11: 同相 32, 反相 31
7	ENOP	0: 关闭 OP1 模块电源 1: 使能 OP1 模块电源
5~4	OPPSEL[1:0]	运放同相端输入选择位 00: 内部接 VSS, 0V 10: 输入差分模式, 偏置电压为 VREF/2, 注意此时需同步使能 VREF_CFG.DIV_EN, VREF/2 才有电压输出 11: 选用 OP1P (外部引脚)

位编号	位符号	说明
3~2	OPNSEL[1:0]	运放反相端输入选择位： 00: 选用 OP1N（外部引脚） 01: 选用 DAC 输出 10: 选用 OPRF[3:0]设定值 11: 接反馈电阻 R2
0	OPOSEL	运放输出端连接选择位 1: 运放输出连接到 OP1O（外部引脚） 0: 运放输出与 OP1O 的连接断开 说明：运放的输出始终连接到 ADC 和 CMPxPS 的可选项
31 29 22~21 10 6 1	-	保留

### 11.7.2.2 OP2 控制寄存器 OP2\_CON

寄存器	读/写	说明	复位值	上电初始值
OP2_CON	读/写	OP2 控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24	
-	FDBRSEL	-	TRIMOFFSETN[4:0]				-	-
23	22	21	20	19	18	17	16	
PGAOFc	-	-	TRIMOFFSETP[4:0]				-	-
15	14	13	12	11	10	9	8	
OPRF[3:0]			MODE		-	PGAGAN[1:0]		
7	6	5	4	3	2	1	0	
ENOP	-	OPPSEL[1:0]		OPNSEL[1:0]		-	OPOSEL	

位编号	位符号	说明
30	FDBRSEL	运放模块的反馈电阻 R1 连接选择位 0: 内部接 VSS, 0V 1: OP2N（外部引脚）
28~24	TRIMOFFSETN[4:0]	Trim for NMOS differential pairs 运放的 NMOS 差分 offset 校准值
23	PGAOFc	OP 输入端短接控制位 0: 同相和反相输入端断开 1: 同相和反相输入端短接
20~16	TRIMOFFSETP[4:0]	Trim for PMOS differential pairs 运放的 PMOS 差分 offset 校准值
15~12	OPRF[3:0]	OP 作比较器模式下，运放反相端输入电压选择位，当 OPNSEL[1:0]=10 时生效： 0000: 1/16 OP <sub>x</sub> _VREF 0001: 1/16 OP <sub>x</sub> _VREF 0010: 2/16 OP <sub>x</sub> _VREF 0011: 3/16 OP <sub>x</sub> _VREF 0100: 4/16 OP <sub>x</sub> _VREF 0101: 5/16 OP <sub>x</sub> _VREF 0110: 6/16 OP <sub>x</sub> _VREF 0111: 7/16 OP <sub>x</sub> _VREF 1000: 8/16 OP <sub>x</sub> _VREF 1001: 9/16 OP <sub>x</sub> _VREF

位编号	位符号	说明
		1010: 10/16 OPx_VREF 1011: 11/16 OPx_VREF 1100: 12/16 OPx_VREF 1101: 13/16 OPx_VREF 1110: 14/16 OPx_VREF 1111: 15/16 OPx_VREF
11	MODE	OP2 模式选择位 0: 运放模式 1: 比较器模式
8~9	PGAGAN[1:0]	运放 PGA 模式内部增益档位选择 00: 同相 4, 反相 3 01: 同相 8, 反相 7 10: 同相 16, 反相 15 11: 同相 32, 反相 31
7	ENOP	0: 关闭 OP2 模块电源 1: 使能 OP2 模块电源
5~4	OPPSEL[1:0]	运放同相端输入选择位 00: 内部接 VSS, 0V 10: 输入差分模式, 偏置电压为 VREF/2, 注意此时需同步使能 VREF_CFG.DIV_EN, VREF/2 才有电压输出 11: 选用 OP2P (外部引脚)
3~2	OPNSEL[1:0]	运放反相端输入选择位: 00: 选用 OP2N (外部引脚) 01: 选用 DAC 输出 10: 选用 OPRF[3:0] 设定值 11: 接反馈电阻 R2
0	OPOSEL	运放输出端连接选择位 1: 运放输出连接到 OP2O (外部引脚) 0: 运放输出与 OP2O 的连接断开 说明: 运放的输出始终连接到 ADC 和 CMPxPS 的可选项
31 29 22~21 10 6 1	-	保留

### 11.7.2.3 OP1/2 配置寄存器 OPX\_CFG

寄存器	读/写	说明	复位值	上电初始值
OPX_CFG	读/写	OP1/2 配置寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
REFSEL	-	OP_CMPIM2[1:0]	OP_CMPIM1[1:0]	-	-	-	-

位编号	位符号	说明
7	REFSEL	运放模块基准 OPx_VREF 源选择位 (x=1~2)

位编号	位符号	说明
		0: 模块基准源为 VDD 1: 模块基准源为 VREF
5~4	OP_CMPIM2[1:0]	OP2 比较器模式，比较触发条件选择位 00: 不触发 01: 上升沿触发: IN+从小于 IN- 到大于 IN- 后触发; 10: 下降沿触发: IN+从大于 IN- 到小于 IN- 后触发; 11: 双沿触发: IN+从小于 IN- 到大于 IN- , 或 IN+从大于 IN- 到小于 IN-后均会触发
3~2	OP_CMPIM1[1:0]	OP1 比较器模式，比较触发条件选择位 00: 不触发 01: 上升沿触发: IN+从小于 IN- 到大于 IN- 后触发; 10: 下降沿触发: IN+从大于 IN- 到小于 IN- 后触发; 11: 双沿触发: IN+从小于 IN- 到大于 IN- , 或 IN+从大于 IN- 到小于 IN-后均会触发
31~8 6 1~0	-	保留

#### 11.7.2.4 OP1/2 比较器状态寄存器 OPX\_STS

寄存器	读/写	说明	复位值	上电初始值
OPX_STS	读/写	OP1/2 比较器状态寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
-	-	OP_CMP2STA	OP_CMP1STA	-	OP_CMP2IF	OP_CMP1IF	-

位编号	位符号	说明
5	OP_CMP2STA	OP2 比较器模式输出状态 0: OP2 比较器正端电压小于负端电压 1: OP2 比较器正端电压大于负端电压
4	OP_CMP1STA	OP1 比较器模式输出状态 0: OP1 比较器正端电压小于负端电压 1: OP1 比较器正端电压大于负端电压
2	OP_CMP2IF	OP2 比较器模式中断标志位 0: OP2 比较器中断未被触发 1: 当 OP2 比较器满足中断触发条件时，此位会被硬件自动设定成 1。如果此时 OP_CMP2IE 使能，OP2 比较器中断产生。在 OP2 比较器中断发生后，硬件并不会自动清除此位，此位必须由使用者的软件负责清除。
1	OP_CMP1IF	OP1 比较器模式中断标志位 0: OP1 比较器中断未被触发 1: 当 OP1 比较器满足中断触发条件时，此位会被硬件自动设定成 1。如果此时 OP_CMP1IE 使能，OP1 比较器中断产生。在 OP1 比较器中断发生后，硬件并不会自动清除此位，此位必须由使用者的软件负责清除。
31~6 3	-	保留

位编号	位符号	说明
0		

### 11.7.2.5 OP1/2 比较器中断使能寄存器 OPX\_IDE

寄存器	读/写	说明	复位值	上电初始值
OPX_IDE	读/写	OP1/2 比较器中断使能寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
INTEN	-	-	-	-	OP_CMP2IE	OP_CMP1IE	-

位编号	位符号	说明
7	INTEN	中断请求 CPU 的使能控制位 0: 禁止中断请求 1: 使能中断请求
2	OP_CMP2IE	OP2 比较器模式中断使能位 0: OP_CMP2IF 置起时, 不允许产生中断 1: OP_CMP2IF 置起时, 允许产生中断
1	OP_CMP1IE	OP1 比较器模式中断使能位 0: OP_CMP1IF 置起时, 不允许产生中断 1: OP_CMP1IF 置起时, 允许产生中断
31~8 6~3 0	-	保留

### 11.7.3 OP0/1/2 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值
OP0/1/2 基地址: 0x4002_21B0					
OP0_CON	0x00	读/写	OP0 控制寄存器	0x0000_0000	0x0000_0000
OP1_CON	0x04	读/写	OP1 控制寄存器	0x0000_0000	0x0000_0000
OP2_CON	0x0C	读/写	OP2 控制寄存器	0x0000_0000	0x0000_0000
OPX_CFG	0x10	读/写	OP1/2 配置寄存器	0x0000_0000	0x0000_0000
OPX_STS	0x14	读/写	OP1/2 比较器状态寄存器	0x0000_0000	0x0000_0000
OPX_IDE	0x18	读/写	OP1/2 比较器中断使能寄存器	0x0000_0000	0x0000_0000

## 12 模拟比较器（CMP）

### 12.1 概述

SC32F15G 内建四个模拟比较器 CMP0/1/2/3，其中 CMP0/1/2 共用反相端，CMP3 完全独立。

CMP 中断可唤醒 STOP 模式。可用于报警器电路、电源电压监测电路、过零检测电路等。

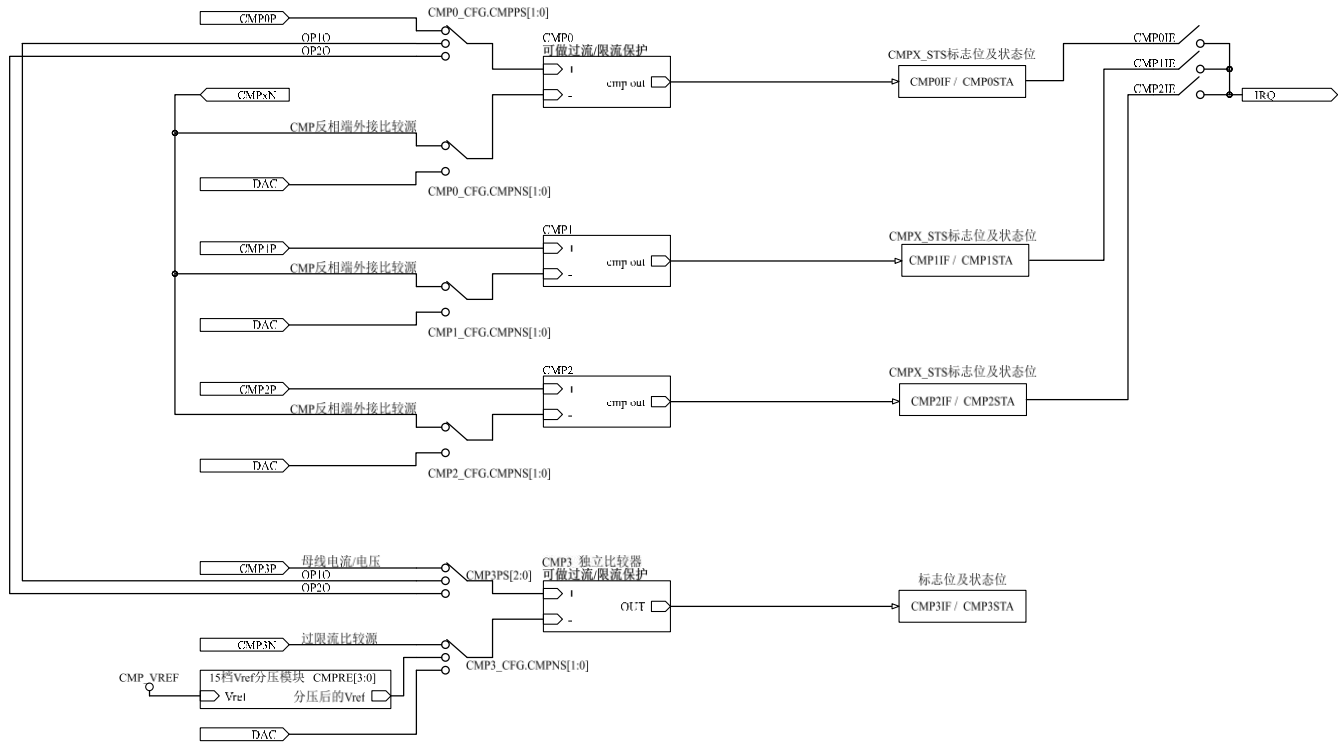
### 12.2 CMP0/1/2 特性

- 三个 CMP 正端均有独立的外部输入端口
- CMP0 的可使用 OP1/OP2 的输出作为正端输入
- 三个 CMP 的负端均可独立切换至：
  - 三个 CMP 共用的外部输入端口 CMPxN
  - 内建 DAC 输出
- CMP0/1/2 中断可唤醒 STOP Mode
- 迟滞电压四档可选：0/5/10/20mV
- 响应时间约为 50ns

### 12.3 CMP3 特性

- CMP3 正端可切换至
  - 外部输入端口 CMP3P
  - OP1/OP2 的输出
- CMP3 负端可切换至
  - 外部输入端口 CMP3N
  - 内建的 DAC 输出
  - VREF 的 16 档分压模块输出
- CMP3 中断可唤醒 STOP Mode
- 迟滞电压四档可选：0/5/10/20mV
- 响应时间约为 50ns

## 12.4 模拟比较器结构框图



CMP 结构框图

## 12.5 CMP 中断

对于 CMP0~3，在满足设定的 CMPIM[1:0]设定条件时会触发中断。可以使用单独的中断使能位以提高灵活性。

中断事件	中断请求控制位	事件标志位	中断使能子开关
CMP0 满足 CMPIM[1:0]设定的中断触发条件	CMPX_IDE->INTEN	CMP0IF	CMP0IE
CMP1 满足 CMPIM[1:0]设定的中断触发条件		CMP1IF	CMP1IE
CMP2 满足 CMPIM[1:0]设定的中断触发条件		CMP2IF	CMP2IE
CMP3 满足 CMPIM[1:0]设定的中断触发条件	CMP3_IDE ->INTEN	CMP3IF	CMP3IE

## 12.6 CMP 寄存器

### 12.6.1 CMP0/1/2 相关寄存器表

#### 12.6.1.1 CMP0/1/2 状态寄存器 CMPX\_STS

寄存器	读/写	说明	复位值	上电初始值
CMPX_STS	读/写	CMP0/1/2 状态寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
-	-	CMP2STA	CMP1STA	CMP0STA	CMP2IF	CMP1IF	CMP0IF

位编号	位符号	说明
5	CMP2STA	CMP2 输出状态位 0: CMP2 正端电压小于负端电压 1: CMP2 正端电压大于负端电压
4	CMP1STA	CMP1 输出状态位 0: CMP1 正端电压小于负端电压 1: CMP1 正端电压大于负端电压
3	CMP0STA	CMP0 输出状态位 0: CMP0 正端电压小于负端电压 1: CMP0 正端电压大于负端电压
2	CMP2IF	CMP2 中断标志位 该位由硬件置 1，通过软件写 1 清 0。 0: CMP2 中断未被触发； 1: 当 CMP2 满足中断触发条件时，此位会被硬件置 1。如果此时 CMP2IE 使能，CMP2 中断产生。
1	CMP1IF	CMP1 中断标志位 该位由硬件置 1，通过软件写 1 清 0。 0: CMP1 中断未被触发； 1: 当 CMP1 满足中断触发条件时，此位会被硬件置 1。如果此时 CMP1IE 使能，CMP1 中断产生。
0	CMP0IF	CMP0 中断标志位 该位由硬件置 1，通过软件写 1 清 0。 0: CMP0 中断未被触发； 1: 当 CMP0 满足中断触发条件时，此位会被硬件置 1。如果此时 CMP0IE 使能，CMP0 中断产生。
31~6	-	保留

### 12.6.1.2 CMP0/1/2 控制寄存器 CMPX\_CON

寄存器	读/写	说明	复位值	上电初始值
CMPX_CON	读/写	CMP0/1/2 控制寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
1~0	HYS[1:0]	CMP0/CMP1/CMP2 迟滞（回差）电压选择位

位编号	位符号	说明
		00: 0V 01: 5mV 10: 10mV 11: 20mV
31~2	-	保留

### 12.6.1.3 CMP0/1/2 中断使能寄存器 CMPX\_IDE

寄存器	读/写	说明	复位值	上电初始值
CMPX_IDE	读/写	CMP0/1/2 中断使能寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
INTEN	-	-	-	-	CMP2IE	CMP1IE	CMP0IE

位编号	位符号	说明
7	INTEN	中断请求 CPU 的使能控制位 0: 禁止中断请求 1: 使能中断请求
2	CMP2IE	CMP2 中断使能位 0: CMP2IF 置起时, 不允许产生中断 1: CMP2IF 置起时, 允许产生中断
1	CMP1IE	CMP1 中断使能位 0: CMP1IF 置起时, 不允许产生中断 1: CMP1IF 置起时, 允许产生中断
0	CMP0IE	CMP0 中断使能位 0: CMP0IF 置起时, 不允许产生中断 1: CMP0IF 置起时, 允许产生中断
31~8 6~3	-	保留

### 12.6.1.4 CMP0 配置寄存器 CMP0\_CFG

寄存器	读/写	说明	复位值	上电初始值
CMP0_CFG	读/写	CMP0 配置寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
7	CMPEN	CMP0 使能位 0: 关闭 CMP0 1: 使能 CMP0
6~5	CMPIM[1:0]	CMP0 中断模式选择位 00: 不产生中断 01: 上升沿中断: IN+从小于 IN- 到大于 IN- 后会产生中断; 10: 下降沿中断: IN+从大于 IN- 到小于 IN- 后会产生中断; 11: 双沿中断: IN+从小于 IN- 到大于 IN- , 或 IN+从大于 IN- 到小于 IN-后均会产生中断;
3~2	CMPPS[1:0]	CMP0 正端信号选择位 00: 选用 CMP0P 01: 选用 OP1O 10: 选用 OP2O 11: 保留
1~0	CMPNS[1:0]	CMP0 负端信号选择位 00: 选用 CMPxN 01: 选用 DAC 输出 10: 保留 11: 保留
31~8 4	-	保留

#### 12.6.1.5 CMP1 配置寄存器 CMP1\_CFG

寄存器	读/写	说明	复位值	上电初始值
CMP1_CFG	读/写	CMP1 配置寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
7	CMPEN	CMP1 使能位 0: 关闭 CMP1 1: 使能 CMP1
6~5	CMPIM[1:0]	CMP1 中断模式选择位 00: 不产生中断 01: 上升沿中断: IN+从小于 IN- 到大于 IN- 后会产生中断; 10: 下降沿中断: IN+从大于 IN- 到小于 IN- 后会产生中断; 11: 双沿中断: IN+从小于 IN- 到大于 IN- , 或 IN+从大于 IN- 到小于 IN-后均会产生中断;
1~0	CMPNS[1:0]	CMP1 负端信号选择位 00: 选用 CMPxN 01: 选用 DAC 输出 10: 保留 11: 保留
31~8	-	保留

位编号	位符号	说明
4~2		

### 12.6.1.6 CMP2 配置寄存器 CMP2\_CFG

寄存器	读/写	说明	复位值	上电初始值
CMP2_CFG	读/写	CMP2 配置寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
7	CMPEN	CMP2 使能位 0: 关闭 CMP2 1: 使能 CMP2
6~5	CMPIM[1:0]	CMP2 中断模式选择位 00: 不产生中断 01: 上升沿中断: IN+从小于 IN- 到大于 IN- 后会产生中断; 10: 下降沿中断: IN+从大于 IN- 到小于 IN- 后会产生中断; 11: 双沿中断: IN+从小于 IN- 到大于 IN- , 或 IN+从大于 IN- 到小于 IN-后均会产生中断;
1~0	CMPNS[1:0]	CMP2 负端信号选择位 00: 选用 CMPxN 01: 选用 DAC 输出 10: 保留 11: 保留
31~8 4~2	-	保留

### 12.6.2 CMP0/1/2 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值
CMP 基地址: 0x4002_2150					
CMPX_STS	0x00	读/写	CMP0/1/2 状态寄存器	0x0000_0000	0x0000_0000
CMPX_CON	0x04	读/写	CMP0/1/2 控制寄存器	0x0000_0000	0x0000_0000
CMPX_IDE	0x08	读/写	CMP0/1/2 中断使能寄存器	0x0000_0000	0x0000_0000
CMP0_CFG	0x0C	读/写	CMP0 配置寄存器	0x0000_0000	0x0000_0000
CMP1_CFG	0x10	读/写	CMP1 配置寄存器	0x0000_0000	0x0000_0000
CMP2_CFG	0x14	读/写	CMP2 配置寄存器	0x0000_0000	0x0000_0000

## 12.7 CMP3 寄存器

### 12.7.1 CMP3 相关寄存器表

#### 12.7.1.1 CMP3 状态寄存器 CMP3\_STS

寄存器	读/写	说明	复位值	上电初始值
CMP3_STS	读/写	CMP3 状态寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
-	-	-	-	-	-	CMP3STA	CMP3IF

位编号	位符号	说明
1	CMP3STA	CMP3 输出状态 0: CMP3 正端电压小于负端电压 1: CMP3 正端电压大于负端电压
0	CMP3IF	CMP3 中断标志位 该位由硬件置 1，通过软件写 1 清 0。 0: CMP3 中断未被触发； 1: 当 CMP3 满足中断触发条件时，此位会被硬件置 1。如果此时 CMP3IE 使能，CMP3 中断产生。
31~2	-	保留

#### 12.7.1.2 CMP3 控制寄存器 CMP3\_CON

寄存器	读/写	说明	复位值	上电初始值
CMP3_CON	读/写	CMP3 控制寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
7	REFSEL	CMP 模块基准 CMP_VREF 源选择位 0: 模块基准源为 VDD 1: 模块基准源为 VREF
1~0	HYS[1:0]	CMP3 迟滞（回差）电压选择位 00: 0V 01: 5mV

位编号	位符号	说明
		10: 10mV 11: 20mV
31~8 6~2	-	保留

### 12.7.1.3 CMP3 中断使能寄存器 CMP3\_IDE

寄存器	读/写	说明	复位值	上电初始值
CMP3_IDE	读/写	CMP3 中断使能寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
7	INTEN	中断请求 CPU 的使能控制位 0: 禁止中断请求 1: 使能中断请求
31~8 6~0	-	保留

### 12.7.1.4 CMP3 配置寄存器 CMP3\_CFG

寄存器	读/写	说明	复位值	上电初始值
CMP3_CFG	读/写	CMP3 配置寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
11~8	CMPRF[3:0]	模拟比较器负端比较电压选择位 模拟比较器负端比较电压设置项如下： 0000: 选用 1/16 CMP_VREF 为模拟比较器的比较电压 0001: 选用 1/16 CMP_VREF 为模拟比较器的比较电压 0010: 选用 2/16 CMP_VREF 为模拟比较器的比较电压 0011: 选用 3/16 CMP_VREF 为模拟比较器的比较电压 0100: 选用 4/16 CMP_VREF 为模拟比较器的比较电压 0101: 选用 5/16 CMP_VREF 为模拟比较器的比较电压 0110: 选用 6/16 CMP_VREF 为模拟比较器的比较电压 0111: 选用 7/16 CMP_VREF 为模拟比较器的比较电压 1000: 选用 8/16 CMP_VREF 为模拟比较器的比较电压

位编号	位符号	说明
		1001: 选用 9/16 CMP_VREF 为模拟比较器的比较电压 1010: 选用 10/16 CMP_VREF 为模拟比较器的比较电压 1011: 选用 11/16 CMP_VREF 为模拟比较器的比较电压 1100: 选用 12/16 CMP_VREF 为模拟比较器的比较电压 1101: 选用 13/16 CMP_VREF 为模拟比较器的比较电压 1110: 选用 14/16 CMP_VREF 为模拟比较器的比较电压 1111: 选用 15/16 CMP_VREF 为模拟比较器的比较电压
7	CMPEN	CMP3 使能位 0: 关闭 CMP3 1: 使能 CMP3
6~5	CMPIM[1:0]	CMP3 中断模式选择位 00: 不产生中断 01: 上升沿中断: IN+从小于 IN- 到大于 IN- 后会产生中断; 10: 下降沿中断: IN+从大于 IN- 到小于 IN- 后会产生中断; 11: 双沿中断: IN+从小于 IN- 到大于 IN- , 或 IN+从大于 IN- 到小于 IN-后均会产生中断;
3~2	CMPPS[1:0]	CMP3 正端信号选择位 00: 选用 CMP3P 01: 选用 OP1O 10: 选用 OP2O 11: 保留
1~0	CMPNS[1:0]	CMP3 负端信号选择位 00: 选用 CMP3N 01: 选用 DAC 输出 10: 选用 CMPRF[3:0]设定值 11: 保留
31~12 4	-	保留

### 12.7.2 CMP3 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值
CMP 基地址: 0x4002_2170					
CMP3_STS	0x00	读/写	CMP3 模拟比较器状态寄存器	0x0000_0000	0x0000_0000
CMP3_CON	0x04	读/写	CMP3 控制寄存器	0x0000_0000	0x0000_0000
CMP3_IDE	0x08	读/写	CMP3 中断使能寄存器	0x0000_0000	0x0000_0000
CMP3_CFG	0x0C	读/写	CMP3 配置寄存器	0x0000_0000	0x0000_0000

## 13 独立正交编码捕捉模块（QEP）

### 13.1 概述

SC32F15G 内部集成两个正交编码捕捉模块（简称 QEP），可以与线性或增量编码器等设备连接，用于获取机器的位置、方向等信息。用户可以通过配置寄存器 QEPn\_CON（n=0~1）的 QSRC[1:0]，选择与外接设备匹配的位置计数器的计数模式，SC32F15G 提供 3 种计数模式：正交计数、方向计数和双脉冲计数。

### 13.2 时钟源

SC32F15G 系列的 QEP 时钟源仅来自 PCLK2

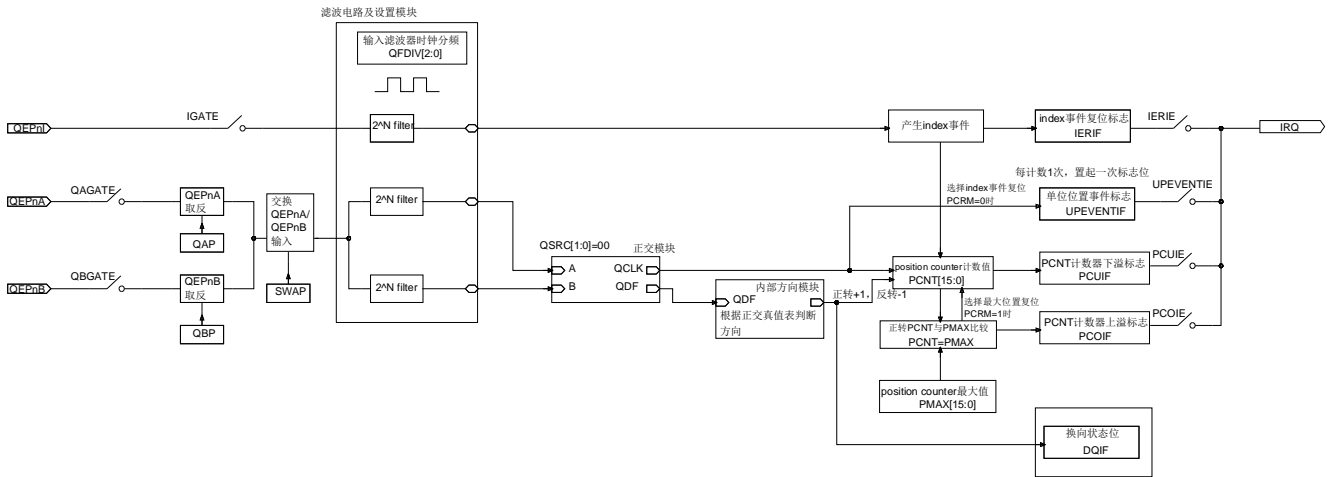
### 13.3 特性

- 提供 3 路输入信号口：QEPnA、QEPnB 和 QEPnI（n=0~1），分别有各自的选通门限 QAGATE、QBGATE 和 IGATE
- 输入信号 QEPnA、QEPnB 可交换输入方向
- 输入信号 QEPnA、QEPnB 可单独配置各自的输入极性
- 为输入信号 QEPnA、QEPnB 和 QEPnI 提供最大 128 分频的数字输入滤波器
- 方向计数和双脉冲计数模式下可选择上升沿、下降沿或双沿计数
- 位置计数器提供 2 种复位模式：index 事件复位、最大值复位（即 PCNT=PMAX 时复位）

### 13.4 正交计数

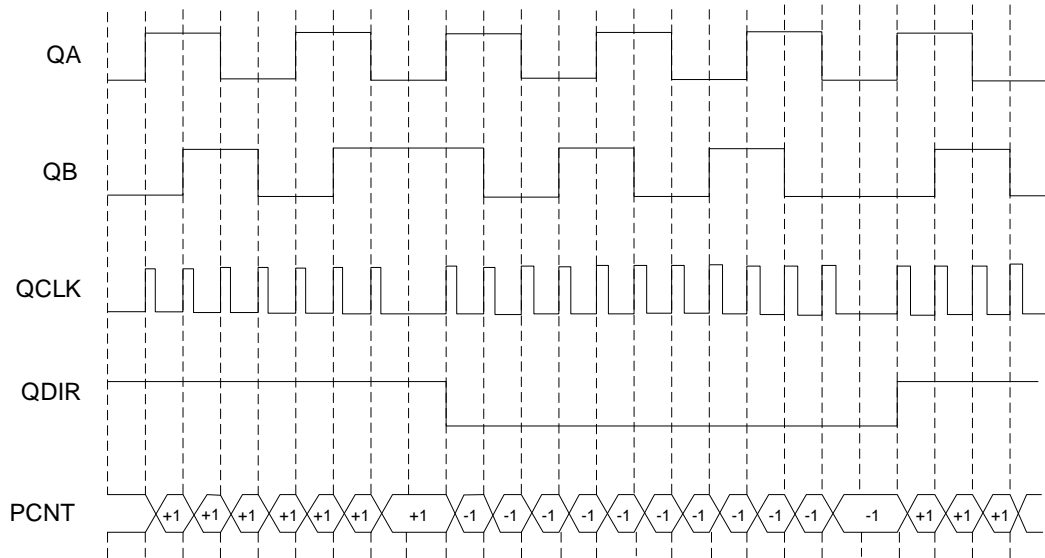
当位置计数源选择正交计数时，即 QSRC[1:0]=00，正交计数要求 QEPnA、QEPnB 两个输入信号相位差 90°，QEP 模块通过 QEPnA 和 QEPnB 的上升/下降沿生成一个正交脉冲 QCLK，若 QEPnA 波形优先，则为正方向，PCNT 向上计数，方向标志位 QDF 为 1，QCLK 每来一个脉冲，PCNT+1；如果 QEPnB 波形优先于 QEPnA，则为反方向，PCNT 向下计数，方向标志位 QDF 为 0，QCLK 每来一个脉冲，PCNT-1。

#### 13.4.1 正交模式结构框图



### 13.4.2 正交模式真值表及波形

上一边沿	当前边沿	DQIF	PCNT
QEPnA↑	QEPnB↑	1	增加
	QEPnB↓	0	减少
	QEPnA↓	翻转	增加或减少
QEPnA↓	QEPnB↓	1	增加
	QEPnB↑	0	减少
	QEPnA↑	翻转	增加或减少
QEPnB↑	QEPnA↑	1	增加
	QEPnA↓	0	减少
	QEPnB↓	翻转	增加或减少
QEPnB↓	QEPnA↓	1	增加
	QEPnA↑	0	减少
	QEPnB↑	翻转	增加或减少

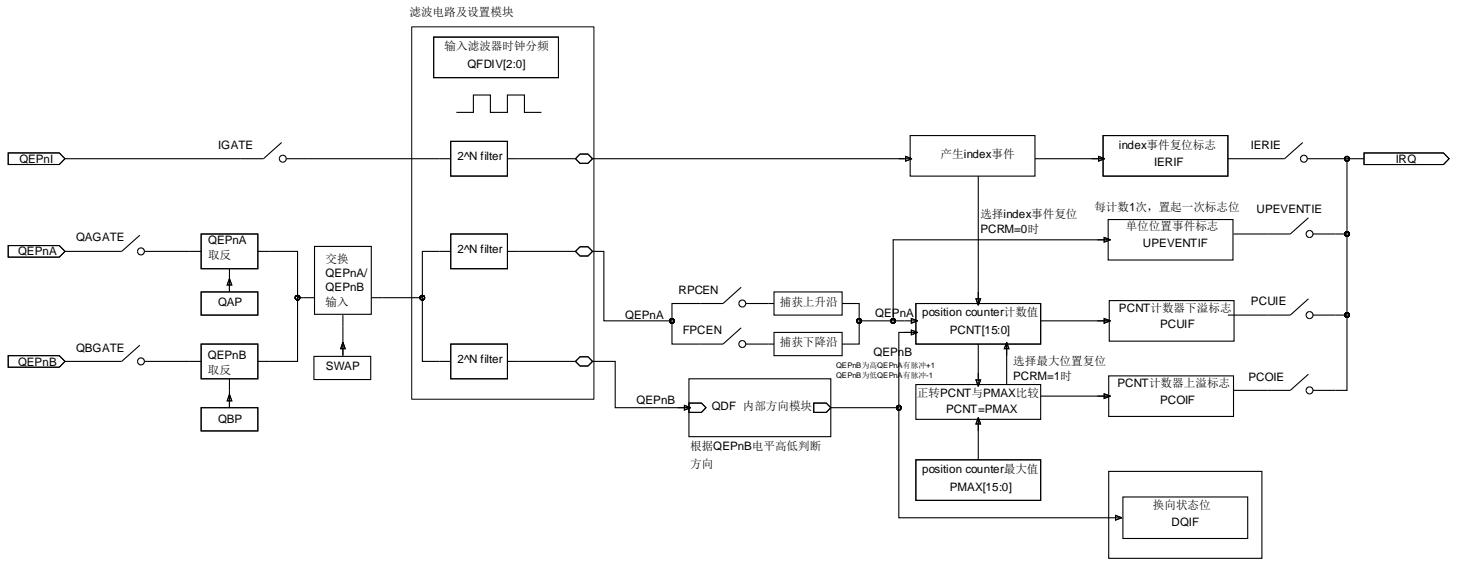


上图波形中，QEPnA、QEPnB 为两个相位差 90°的输入信号，QCLK 信号来源于 A、B 两相信号的跳变沿，DQIF 方向信号可根据 QEPnA、QEPnB 的真值表判断方向，QCLK 信号和 DQIF 信号一起确定位置计数器（QEPn\_PCNT）的计数增减：正向时，一个 QCLK+1；反向时，一个 QCLK-1。

### 13.5 方向计数

当位置计数源选择方向计数时，即 QSRC[1:0]=01，方向计数要求输入信号 QEPnA 为计数脉冲，输入信号 QEPnB 指示方向。可通过使能上升沿计数 RPCEN、下降沿计数 FPCEN 或同时使能实现双沿计数。当输入信号 QEPnB 为高电平时，方向模块判断为正方向，方向标志位 QDF 为 1，此时检测到 QEPnA 对应的计数脉冲，则位置计数器 PCNT+1；当输入信号 QEPnB 为低电平时，方向模块判断为反方向，方向标志位 QDF 为 0，此时检测到 QEPnA 对应的计数脉冲，则位置计数器 PCNT-1。方向计数模式下框图如下：

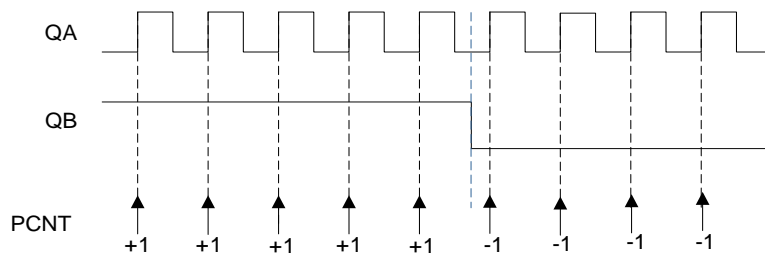
### 13.5.1 方向计数模式结构框图



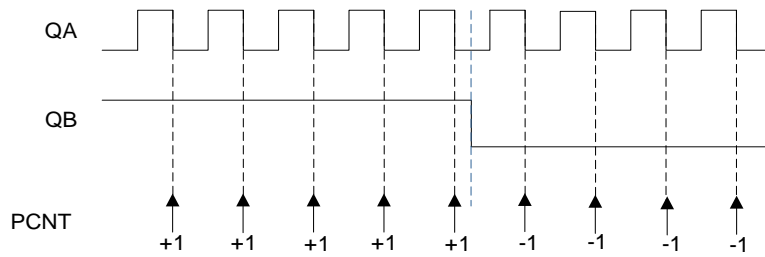
### 13.5.2 方向计数模式波形

方向计数模式根据 QEPnB 的电平判断方向。

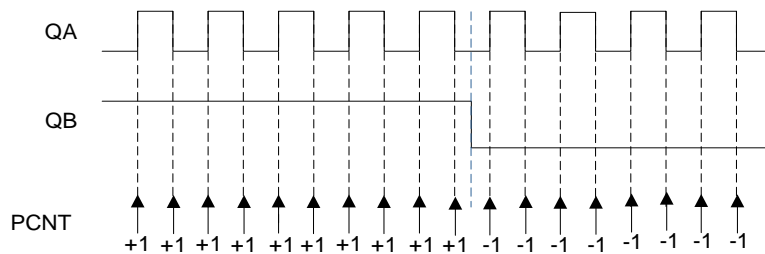
若使能上升沿计数，当 QEPnB 为高电平时（即 CW 方向），位置计数器（QEPn\_PCNT）在 QEPnA 信号的每个上升沿+1；当 QEPnB 为低电平时（即 CCW 方向），位置计数器（QEPn\_PCNT）在 QEPnA 信号的每个上升沿-1。波形图如下：



若使能下降沿计数，当 QEPnB 为高电平时（即 CW 方向），位置计数器（QEPn\_PCNT）在 QEPnA 信号的每个下降沿+1；当 QEPnB 为低电平时（即 CCW 方向），位置计数器（QEPn\_PCNT）在 QEPnA 信号的每个下降沿-1。波形图如下：



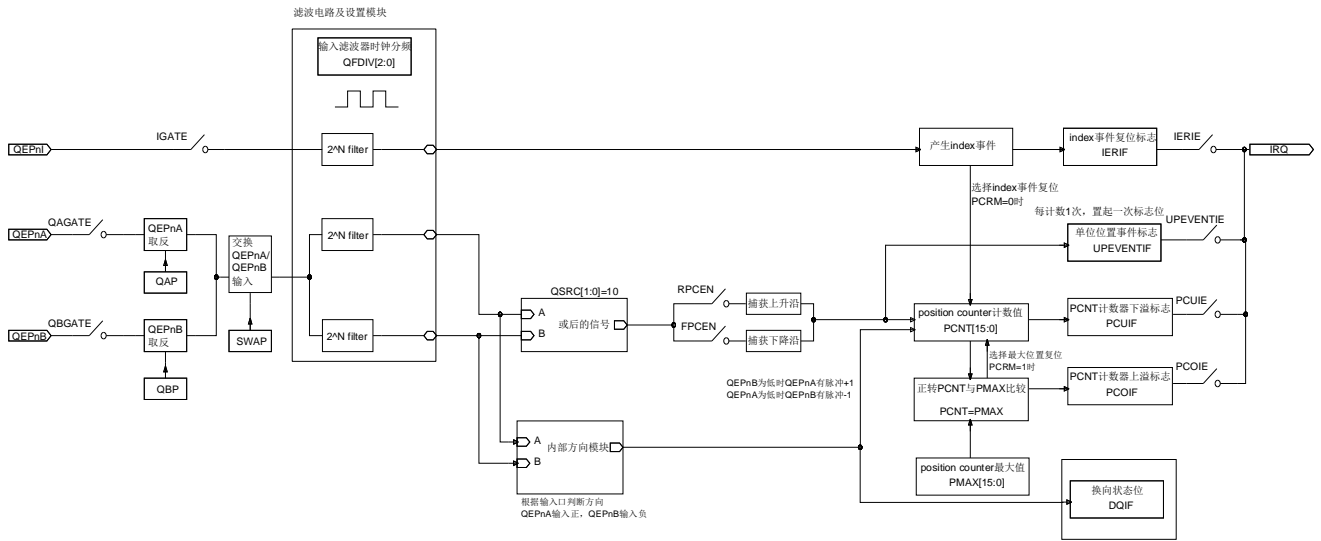
若使能双沿计数，当 QEPnB 为高电平时（即 CW 方向），位置计数器（QEPn\_PCNT）在 QEPnA 信号的每个上升沿和下降沿+1；当 QEPnB 为低电平时（即 CCW 方向），位置计数器（QEPn\_PCNT）在 QEPnA 信号的每个上升沿和下降沿-1。波形图如下：



## 13.6 双脉冲计数

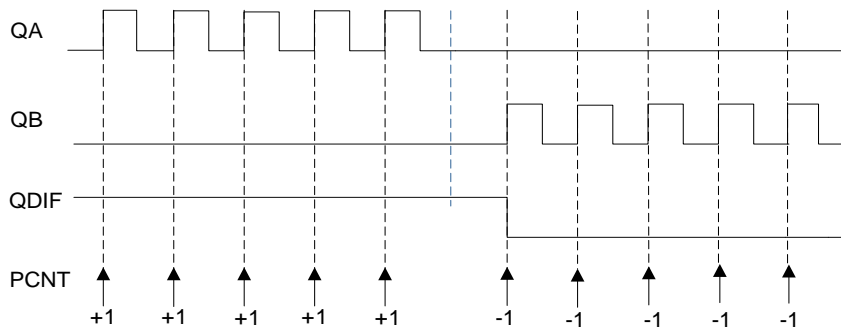
当位置计数源选择双脉冲计数时，即 QSRC[1:0]=10，输入信号 QEPnA 有脉冲 QEPnB 为低电平时，为正方向计数，方向标志位 QDF 为 1，每来一个脉冲，位置计数器 PCNT+1；输入信号 QEPnB 有脉冲 QEPnA 为低电平时，为反方向计数，方向标志位 QDF 为 0，QEPnB 每来一个脉冲，位置计数器 PCNT-1。可通过使能上升沿计数 RPCEN、下降沿计数 FPCEN 或同时使能实现双沿计数。

### 13.6.1 双脉冲计数模式结构框图

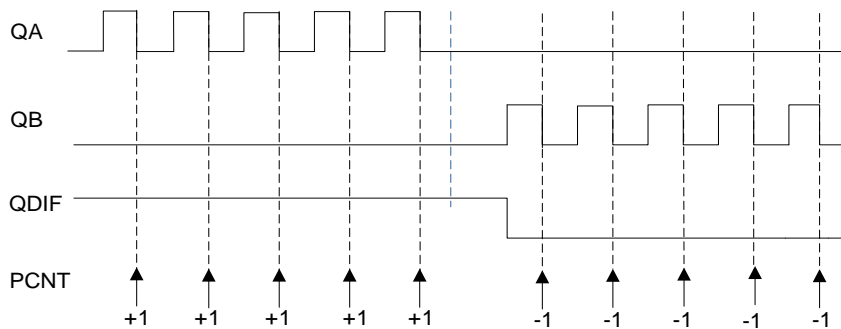


### 13.6.2 双脉冲计数模式波形

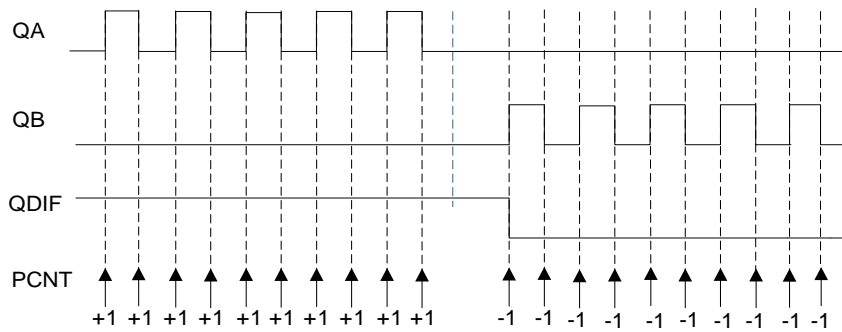
使能上升沿计数时，波形图如下：



使能下降沿计数时，波形图如下：



同时使能上升沿和下降沿计数时，波形图如下：



## 13.7 复位方式

当选择 index 事件复位时，即 PCRM=0 时，index 输入引脚检测到 index 信号，位置计数器 PCNT 复位，正转时复位为 0，反转时复位为 PMAX 值，此时 index 事件复位标志 IERIF 置起，若使能 IERIE 可同时触发中断。

当选择最大值复位时，即 PCRM=1，位置计数器 PCNT 向上计数到 PMAX 值时，PCNT 计数器上溢标志位置起，PCNT 复位为 0；位置计数器 PCNT 向下计数到 0 时，PCNT 计数器下溢标志位置起，PCNT 复位为 PMAX 值。

## 13.8 单位位置事件

为方便测量速度信息，SC32F15G QEP 模块提供单位位置事件中断，在不同计数模式下，QEP 模块每捕获到一个计数沿就会进入一次单位位置事件中断，方便通过外部 Timer 捕获单位位置事件的时间间隔，从而利用以下公式测量速度信息。

$$v(k) \approx \frac{X}{t(k) - t(k-1)} = \frac{X}{\Delta T}$$

X 表示  $t(k)$  到  $t(k-1)$  时间段采集到的计数值。

## 13.9 QEP 寄存器

### 13.9.1 QEP 相关寄存器表

#### 13.9.1.1 QEPn 控制寄存器 QEPn\_CON

寄存器	读/写	说明	复位值	上电初始值
QEPn_CON	读/写	QEPn 控制寄存器 (n=0~1)	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	SPOS	-	-	-	-	QIP
15	14	13	12	11	10	9	8

SWAP	PCRM	QBP	QAP	-	QFDIV[2:0]		
7	6	5	4	3	2	1	0
QEPEN	IGATE	QBGATE	QAGATE	FPCEN	RPCEN	QSRC[1:0]	

位编号	位符号	说明												
21	SPOS	QEP1 信号口映射控制位@QEP1_CON												
		<table border="1"> <tr> <td>信号</td> <td>QEP1A</td> <td>QEP1B</td> <td>QEP1I</td> </tr> <tr> <td>SPOS=0</td> <td>PA0</td> <td>PA1</td> <td>PA2</td> </tr> <tr> <td>SPOS=1</td> <td>PC4</td> <td>PC5</td> <td>PC6</td> </tr> </table>	信号	QEP1A	QEP1B	QEP1I	SPOS=0	PA0	PA1	PA2	SPOS=1	PC4	PC5	PC6
		信号	QEP1A	QEP1B	QEP1I									
		SPOS=0	PA0	PA1	PA2									
SPOS=1	PC4	PC5	PC6											
16	QIP	QEPnI 输入极性 0 正向 1 反向												
15	SWAP	QEPnA 和 QEPnB 输入交换位 0: QEPnA 和 QEPnB 输入不交换 1: QEPnA 和 QEPnB 在滤波前输入交换												
14	PCRM	index 事件复位选择位 0: 使能 index 事件, index 事件发生时复位位置计数器, 正向复位时 PCNT=0, 反向复位时 PCNT=PMAX 1: 屏蔽 index 事件												
13	QBP	QEPnB 输入极性 0 正向 1 反向												
12	QAP	QEPnA 输入极性 0 正向 1 反向												
10~8	QFDIV[2:0]	QEPnA、QEPnB、QEPnI 数字输入滤波器滤波宽度选择位 111 = 1:128 时钟分频 110 = 1:64 时钟分频 101 = 1:32 时钟分频 100 = 1:16 时钟分频 011 = 1:8 时钟分频 010 = 1:4 时钟分频 001 = 1:2 时钟分频 000 = 1:1 时钟分频												
7	QEPEN	QEP 模块使能 0: 不使能 1: 使能												
6	IGATE	QEPnI 选通 0: 禁用 QEPnI pin 1: 使能 QEPnI pin												
5	QBGATE	QEPnB 引脚选通 0: 禁用 QEPnB pin 1: QEPnB 引脚设置为 QEP 输入状态												
4	QAGATE	QEPnA 引脚选通 0: 禁用 QEPnA pin 1: QEPnA 引脚设置为 QEP 输入状态												
3	FPCEN	下降沿计数使能 0: 不使能 1: 使能												
2	RPCEN	上升沿计数使能 0: 不使能 1: 使能												

位编号	位符号	说明
1~0	QSRC[1:0]	位置计数源选择 00: 正交计数模式: 根据 QEPnA、QEPnB 的真值表判断方向; 位置计数器 (QPOSCNT) 在正向时, 一个 QCLK+1; 反向时, 一个 QCLK-1。 01: 方向计数模式: QEPnA 为位置计数器提供时钟, QEPnB 提供方向信息。 10: 双脉冲计数模式: 此模式下, 周期性计数得到的就是速度信号。 11: 保留
31~22 20~17 11	-	保留

### 13.9.1.2 QEPn 位置计数器计数值寄存器 QEPn\_PCNT

寄存器	读/写	说明	复位值	上电初始值
QEPn_PCNT	读/写	QEPn 位置计数器计数值寄存器 (n=0~1)	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
15~0	PCNT[15:0]	位置计数器计数值
31~16	-	保留

### 13.9.1.3 QEPn 位置计数器最大值寄存器 QEPn\_PMAX

寄存器	读/写	说明	复位值	上电初始值
QEPn_PMAX	读/写	QEPn 位置计数器最大值寄存器 (n=0~1)	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
15~0	PMAX[15:0]	位置计数器最大值 向上计数时, 如已配置 PMAX, 则 PCNT 计数值计到 PMAX 后清零。 向下计数时, PCNT 计到 0 后载入最大值。

位编号	位符号	说明
31~16	-	保留

#### 13.9.1.4 QEPn 状态标志寄存器 QEPn\_STS

寄存器	读/写	说明	复位值	上电初始值
QEPn_STS	读/写	QEPn 状态标志寄存器 (n=0~1)	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
4	DQIF	正交方向标志 0: 逆时针旋转 (或反向运动) 1: 顺时针旋转 (或正向移动)
3	UPEVNTIF	单位位置事件标志 0: 未检测到单元位置事件 1: 检测到单位位置事件。写 1 清除。
2	IERIF	index 事件重置中断标志 0: 没有产生中断 1: 复位 PCNT 后该位置起
1	PCOIF	位置计数器溢出中断标志 0: 没有产生中断 1: 位置计数器向上溢出时该位置起
0	PCUIF	位置计数器下溢中断标志 0: 没有产生中断 1: 位置计数器向下溢出时该位置起
31~5	-	保留

#### 13.9.1.5 QEPn 中断使能寄存器 QEPn\_IDE

寄存器	读/写	说明	复位值	上电初始值
QEPn_IDE	读/写	QEPn 中断使能寄存器 (n=0~1)	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
7	INTEN	中断请求 CPU 的屏蔽位 0: 中断请求屏蔽 1: 中断请求开启
3	UPEVNTIE	单位位置事件中断使能 0: 禁用中断 1: 启用中断
2	IERIE	启用 index 事件重置中断 0: 禁用中断 1: 启用中断
1	PCOIE	位置计数器溢出中断启用 0: 禁用中断 1: 启用中断
0	PCUIE	位置计数器下溢中断使能 0: 禁用中断 1: 启用中断
31~8 6~4	-	保留

### 13.9.2 QEP 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值
QEP0 基地址: 0x4002_2020					
QEP0_CON	0x00	读/写	QEP0 控制寄存器	0x0000_0000	0x0000_0000
QEP0_PCNT	0x04	读/写	QEP0 位置计数器计数值寄存器	0x0000_0000	0x0000_0000
QEP0_PMAX	0x08	读/写	QEP0 位置计数器最大值寄存器	0x0000_0000	0x0000_0000
QEP0_STS	0x0C	读/写	QEP0 状态标志寄存器	0x0000_0000	0x0000_0000
QEP0_IDE	0x10	读/写	QEP0 中断使能寄存器	0x0000_0000	0x0000_0000

寄存器	偏移地址	读/写	说明	复位值	上电初始值
QEP1 基地址: 0x4002_2040					
QEP1_CON	0x00	读/写	QEP1 控制寄存器	0x0000_0000	0x0000_0000
QEP1_PCNT	0x04	读/写	QEP1 位置计数器计数值寄存器	0x0000_0000	0x0000_0000
QEP1_PMAX	0x08	读/写	QEP1 位置计数器最大值寄存器	0x0000_0000	0x0000_0000
QEP1_STS	0x0C	读/写	QEP1 状态标志寄存器	0x0000_0000	0x0000_0000
QEP1_IDE	0x10	读/写	QEP1 中断使能寄存器	0x0000_0000	0x0000_0000

## 14 16 位定时/计数器 (TIM) Timer0~Timer3

### 14.1 时钟源

- 定时模式/PWM 输出模式下，TIM 时钟源来自 PCLK
- 计数模式下，Tn 引脚为计数源输入

### 14.2 特性

- 支持 8 档 TIM 时钟预分频
- 4 个独立 16 Bits 自动重载计数器 Timer0~Timer3
- 16 位递增、递减、递增/递减自动重载计数器
- 支持上升沿/下降沿捕获，可实现 PWM duty 和周期捕获
- TIM1/2 溢出及捕获事件可产生 DMA 请求
- 所有 Timer 的 Tn/TnCAP/ TnPWMA、和 TnEX / TnPWMB 管脚均可以重映射

### 14.3 计数方式

#### 14.3.1 定时模式下 TIM 计数方式

- 向上计数：从设定值开始向上计数，至 0xFFFF 溢出
- 向下计数：从 0xFFFF 开始向下计数至设定值

#### 14.3.2 PWM 输出模式下 TIM 计数方式

PWM 输出模式下只能选择向上计数：从 0 开始向上计数，至占空比设置项 PDT 时 PWM 输出波形切换高低电平，之后继续向上计数到设定的重载值 RLD，产生溢出并从 0 重新开始计数。

TIM 输出的 PWM 周期  $T_{PWM}$  计算公式如下：

$$T_{PWM} = \frac{RLD[15:0] + 1}{PCLK}$$

占空比 duty 计算公式：

$$duty = \frac{PDT[15:0]}{RLD[15:0] + 1}$$

### 14.4 定时器相关的信号口

- TnCAP/Tn, n=0~3
  - Tn 时钟输入/输出

- TnCAP 上升沿/下降沿均可捕获
- 注意：Tn 和 TnCAP 为复用功能，不能同时使用
- TnEX, n=0~3
  - 重载模式下，TnEX 引脚上的外部事件输入（下降沿）用作重载允许/禁止控制
  - 捕获模式，当 FSEL = 1 时为下降沿捕获信号输入脚，检测到 TnEX 引脚上一个下降沿，产生一个捕获，EXIF 被置起，TnCNT 寄存器的值捕获到寄存器 FCAP 里
- TnPWM, n=0~3
  - Timer0~3 可通过各自的 Tn 端口提供 duty 可单独调的 PWM：TnPWMA
  - Timer0~3 可通过各自的 TnEX 端口提供 duty 可单独调的 PWM：TnPWMB
  - TnPWMA 和 TnPWMB 共周期，时钟源随 TIM

注意：TIM 的 PWM 捕获功能与 PWM 输出功能不可同时开启

## 14.5 TIM 的中断及对应标志位

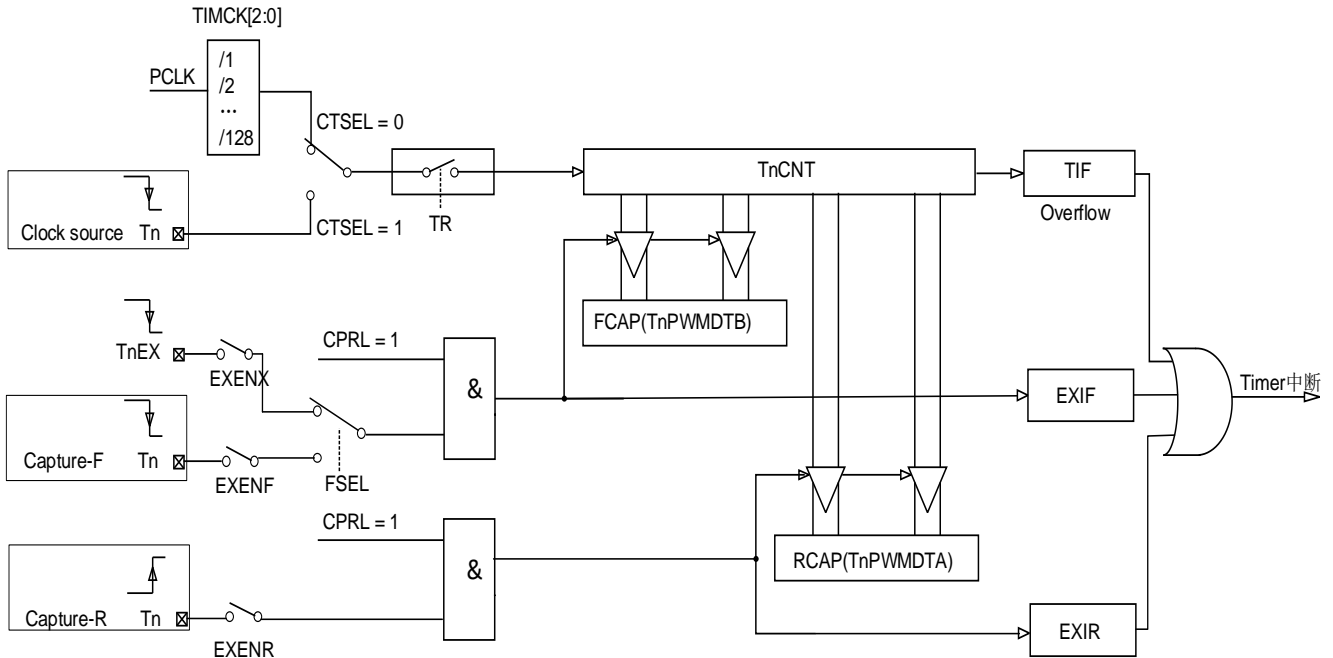
- 计数器上溢/下溢——共用中断标志位 TIF
- 捕获状态标志：
  - EXIF 外部事件输入下降沿被检测到的标志位
  - EXIR 外部事件输入上升沿被检测到的标志位
- 中断及优先级配置控制位合并至 NVIC 模块

## 14.6 TIM 的工作模式

- 模式 0：16 位捕获模式，可实现 PWM 双沿捕获
- 模式 1：16 位自动重载定时/计数器模式
- 模式 3：可编程时钟输出模式
- 模式 4：PWM 输出模式

## 14.6.1 工作模式 0：16 位捕获模式

### 14.6.1.1 双沿捕获结构图



模式 0：16 位捕获

### 14.6.1.2 PWM 双沿捕获模式

TnEX 为 PWM 捕获口。内部测量分个通道，分别采样上升沿和下降沿：

- 专门一个下降沿捕获计数寄存器 16 Bits——与 PWM duty 寄存器复用
- 专门一个上升沿捕获计数寄存器 16 Bits——与 PWM duty 寄存器复用
- 捕获标志位：
  - 单独的上/下沿捕获使能位
  - 单独的上/下捕获标志

## 14.6.2 工作模式 1：16 位自动重载定时器/计数器模式

在 16 位自动重载方式下，定时器可以被选为递增计数或递减计数。这个功能通过将 TIMn\_CON (n=0~3) 中的 DEC 置 1 后，通过 TnEX 选择计数方向。系统复位后，DEC 位复位值为 0，定时器 n 默认递增计数；当 DEC 置 1 时，定时器 n 递增计数或递减计数取决于 TnEX 引脚上的电平。

当 DEC=0，通过在 TIMn\_CON 中的 EXENX 位选择两个选项：

若 EXENX = 0，当 TIMn\_CNT 递增到 0xFFFF 时，定时器溢出位 TIF 置起，同时定时器自动将用户软件写好的重载寄存器 TIMn\_RLD 中的 16 位值装入 TIMn\_CNT 寄存器。

若 EXENX = 1，溢出或在外部输入 TnEX 上的下降沿都能触发一个 16 位重载。TnEX 上有下降沿产生时，

EXIF 位置起。如果 TIE 被使能，TIF 和 EXIF 位都能产生一个中断。

当 DCEN = 1 时，TnEX 引脚控制计数的方向，而 EXENX 控制无效。

若 TnEX = 1，则 TIMn 递增计数。当 TIMn\_CNT 递增到 0xFFFF 时，定时器溢出位 TIF 置起，同时定时器自动将用户软件写好的重载寄存器 TIMn\_RLD 中的 16 位值装入 TIMn\_CNT 寄存器。

若 TnEX = 0，则 TIMn 递减计数。当 TIMn\_CNT 的值从 0xFFFF 递减至等于 TIMn\_RLD 的值时，定时器溢出，且定时器溢出位 TIF 置起，同时 0xFFFF 重载入 TIMn\_CNT。

在此工作方式下，无论 TIMn 溢出与否，EXFIF 不作为中断标志。

### 14.6.3 工作模式 3：可编程时钟输出模式

在这种方式中，TIMn (n=0~3) 可以编程为输出 50% 占空比时钟周期：当 CTSEL=0, TnOE=1 时，使能 TIMn 作为时钟发生器。

在这种方式下 Tn 输出的时钟频率为：

$$f_{OUT} = \frac{f_{TIM}}{(65536 - TIMn_{RLD}) * 4}$$

### 14.6.4 工作模式 4：PWM 输出模式

- PWM 占空比变化特性

更改 PDTx[15:0] 的值，占空比不会立即改变，而是等待本周期结束，在下个周期改变。

- PWM 周期变化特性

通过改变周期设置寄存器[RCAPXL / RCAPXH]的值实现。定义当前周期计数值为 Tn，写入周期寄存器时，定时器计到的值为 Tm，待更新的周期计数值为 Tx，则：

- Tm ≤ Tx：周期按照 Tx 实时改变；
- Tm > Tx：此时周期变化会分为两个阶段。第一个阶段，写入周期寄存器之后，周期计数器会从当前计数值累加至溢出清零。第二个阶段，周期按照 Tx 改变。

## 14.7 TIM 中断

定时或计数模式下，CNT 计数达到 TIMn 计数值，TIF 将置起，如果 TIMn\_IDE.INTEN=1，将产生中断。

外部事件输入模式下，检测到有效跳变沿，EXIR/EXIF 将置起，如果 TIMn\_IDE.INTEN=1，将产生中断。

中断事件	事件标志位	中断使能总控制位	模块中断使能子控制位
定时器溢出	TIF	TIMn_IDE->INTEN (n=0~3)	TIMn_IDE->TIE
外部事件输入上升沿中断	EXIR		TIMn_IDE->EXRIE
外部事件输入下降沿中断	EXIF		TIMn_IDE->EXFIE

## 14.8 TIM 寄存器

### 14.8.1 TIM 相关寄存器表

#### 14.8.1.1 定时器控制寄存器 TIMn\_CON

寄存器	读/写	说明	复位值	上电初始值
TIMn_CON (n=0~3)	读/写	定时器控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	SPOS[1:0]		-	-	-	-	-
15	14	13	12	11	10	9	8
TXOE	EPWMNA	EPWMNB	INVNA	INVNB	TIMCK[2:0]		
7	6	5	4	3	2	1	0
TR	DEC	EXENX	FSEL	EXENF	EXENR	CTSEL	CPRL

位编号	位符号	说明															
22~21	SPOS[1:0]	<ul style="list-style-type: none"> <li>● TIM0 信号口映射控制位 @TIM0_CON</li> <table border="1"> <thead> <tr> <th>信号</th> <th>T0CAP/T0/T0PWMA</th> <th>T0EX/T0PWMB</th> </tr> </thead> <tbody> <tr> <td>SPOS 值</td> <td></td> <td></td> </tr> <tr> <td>SPOS[1:0]=00</td> <td>PA4</td> <td>PA5</td> </tr> <tr> <td>SPOS[1:0]=01</td> <td>PB0</td> <td>PB1</td> </tr> <tr> <td>SPOS[1:0]=10</td> <td>PC6</td> <td>PC7</td> </tr> </tbody> </table> </ul>	信号	T0CAP/T0/T0PWMA	T0EX/T0PWMB	SPOS 值			SPOS[1:0]=00	PA4	PA5	SPOS[1:0]=01	PB0	PB1	SPOS[1:0]=10	PC6	PC7
		信号	T0CAP/T0/T0PWMA	T0EX/T0PWMB													
		SPOS 值															
		SPOS[1:0]=00	PA4	PA5													
		SPOS[1:0]=01	PB0	PB1													
		SPOS[1:0]=10	PC6	PC7													
		<ul style="list-style-type: none"> <li>● TIM1 信号口映射控制位 @TIM1_CON</li> <table border="1"> <thead> <tr> <th>信号</th> <th>T1CAP/T1/T1PWMA</th> <th>T1EX/T1PWMB</th> </tr> </thead> <tbody> <tr> <td>SPOS 值</td> <td></td> <td></td> </tr> <tr> <td>SPOS[1:0]=00</td> <td>PC11</td> <td>PC12</td> </tr> <tr> <td>SPOS[1:0]=01</td> <td>PB2</td> <td>PB3</td> </tr> </tbody> </table> </ul>	信号	T1CAP/T1/T1PWMA	T1EX/T1PWMB	SPOS 值			SPOS[1:0]=00	PC11	PC12	SPOS[1:0]=01	PB2	PB3			
		信号	T1CAP/T1/T1PWMA	T1EX/T1PWMB													
		SPOS 值															
		SPOS[1:0]=00	PC11	PC12													
		SPOS[1:0]=01	PB2	PB3													
		<ul style="list-style-type: none"> <li>● TIM2 信号口映射控制位 @TIM2_CON</li> <table border="1"> <thead> <tr> <th>信号</th> <th>T2CAP/T2/T2PWMA</th> <th>T2EX/T3PWMB</th> </tr> </thead> <tbody> <tr> <td>SPOS 值</td> <td></td> <td></td> </tr> <tr> <td>SPOS[1:0]=00</td> <td>PC14</td> <td>PC13</td> </tr> <tr> <td>SPOS[1:0]=01</td> <td>PB4</td> <td>PB5</td> </tr> </tbody> </table> </ul>	信号	T2CAP/T2/T2PWMA	T2EX/T3PWMB	SPOS 值			SPOS[1:0]=00	PC14	PC13	SPOS[1:0]=01	PB4	PB5			
		信号	T2CAP/T2/T2PWMA	T2EX/T3PWMB													
		SPOS 值															
		SPOS[1:0]=00	PC14	PC13													
		SPOS[1:0]=01	PB4	PB5													
<ul style="list-style-type: none"> <li>● TIM3 信号口映射控制位 @TIM3_CON</li> <table border="1"> <thead> <tr> <th>信号</th> <th>T3CAP/T3/T3PWMA</th> <th>T3EX/T3PWMB</th> </tr> </thead> <tbody> <tr> <td>SPOS 值</td> <td></td> <td></td> </tr> <tr> <td>SPOS[1:0]=00</td> <td>PC2</td> <td>PC1</td> </tr> <tr> <td>SPOS[1:0]=01</td> <td>PB6</td> <td>PB7</td> </tr> </tbody> </table> </ul>	信号	T3CAP/T3/T3PWMA	T3EX/T3PWMB	SPOS 值			SPOS[1:0]=00	PC2	PC1	SPOS[1:0]=01	PB6	PB7					
信号	T3CAP/T3/T3PWMA	T3EX/T3PWMB															
SPOS 值																	
SPOS[1:0]=00	PC2	PC1															
SPOS[1:0]=01	PB6	PB7															
15	TXOE	Tn 引脚信号方向控制位 0: Tn 作为时钟输入或 I/O 端口 1: Tn 用于可编程时钟输出															
14	EPWMNA	Tn_PWMA 所在管脚 PWM 波形输出使能位 0: 关闭 1: 使能															

位编号	位符号	说明
13	EPWMNB	TnEX_PWMB 所在管脚 PWM 波形输出使能位 0: 关闭 1: 使能
12	INVNA	TPWMnA 波形输出反向控制位 0: 正常 1: 波形输出反向
11	INVNB	TPWMnB 波形输出反向控制位 0: 正常 1: 波形输出反向
10~8	TIMCK[2:0]	TIM 时钟频率档位控制位 用于设定 TIM 时钟频率 $f_{TIM}$ 为: 000: $f_{SOURCE}/1$ 001: $f_{SOURCE}/2$ 010: $f_{SOURCE}/4$ 011: $f_{SOURCE}/8$ 100: $f_{SOURCE}/16$ 101: $f_{SOURCE}/32$ 110: $f_{SOURCE}/64$ 111: $f_{SOURCE}/128$  $f_{SOURCE}$ 对应的时钟可为 PCLK 或 Tn 输入
7	TR	TIMn 开始/停止控制位, n=0~3 0: 停止 TIMn / TPWMn 计数器 1: 开始 TIMn / TPWMn 计数器
6	DEC	递增/递减方向控制位 0: TIMn 为递增的定时/计数器 1: TIMn 作为递增/递减的定时/计数器, TnEX 用来选择计数方向
5	EXENX	TnEX 设置位, n=0~3 该位在不同模式下作用不同: ● 重载模式: (CPRL = 0) 该位用于控制 TnEX 引脚上的外部事件输入 (下降沿), 用作重载允许/禁止控制: 0: 忽略 TnEX 引脚上的事件 1: 检测到 TnEX 引脚上一个下降沿, 产生一个重载 ● 捕获模式: (CPRL = 1) 该位用作 TnEX 下降沿信号捕获选通位: 0: 忽略 TnEX 引脚上的事件 1: 当 FSEL = 1, 检测到 TnEX 引脚上一个下降沿, 产生一个捕获, EXIF 被置起, TnCNT 寄存器的值捕获到寄存器 FCAP 里
4	FSEL	下降沿信号选择位 该位仅在捕获模式 (CPRL=1) 下有效 0: 检测到 Tn 引脚上一个下降沿, 产生一个捕获。忽略 TnEX 引脚上的事件 1: 检测到 TnEX 引脚上一个下降沿, 产生一个捕获。忽略 Tn 引脚上的事件
3	EXENF	下降沿信号捕获使能位: 0: 忽略 Tn 引脚上的事件 1: 检测到 Tn 引脚上一个下降沿, 产生一个捕获, EXIF 被置起, TnCNT 寄存器的值捕获到寄存器 FCAP 里
2	EXENR	Tn 引脚上的上升沿信号捕获使能位 0: 忽略 Tn 引脚上的事件

位编号	位符号	说明
		1: 检测到 Tn 引脚上一个上升沿, 产生一个捕获, EXIR 被置起, TnCNT 寄存器的值捕获到寄存器 RCAP 里
1	CTSEL	定时器/计数器选定位 0: 定时器方式 1: 计数器方式
0	CPRL	捕获/重载方式设置位 0: 重载功能 1: 捕获功能
31~23 20~16	-	保留

#### 14.8.1.2 定时器计数值寄存器 TIMn\_CNT

寄存器	读/写	说明	复位值	上电初始值
TIMn_CNT (n=0~3)	读/写	定时器计数值寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
15~0	CNT[15:0]	TIMn 计数值
31~16	-	保留

#### 14.8.1.3 定时器重载寄存器 TIMn\_RLD

寄存器	读/写	说明	复位值	上电初始值
TIMn_RLD (n=0~3)	读/写	定时器重载寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
15~0	RLD[15:0]	定时器溢出或在外部输入 TnEX 上的下降沿都能触发一个 16 位重载。产生重载时，定时器自动将用户软件写好的 RLD[15:0]值装入 TnCNT 寄存器。
31~16	-	保留

#### 14.8.1.4 定时器标志位寄存器 TIMn\_STS

寄存器	读/写	说明	复位值	上电初始值
TIMn_STS (n=0~3)	读/写	定时器标志制寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
2	EXIF	外部事件输入下降沿被检测到的标志位 该位由硬件置 1，通过软件写 1 清 0。 0: 无外部事件输入 1: 检测到外部输入（如果 EXENF = 1，由硬件设 1） 注意：捕获模式下，该位软件清 0 之前不允许更新 TnFCAP 值
1	EXIR	Tn 引脚外部事件输入上升沿被检测到的标志位 该位由硬件置 1，通过软件写 1 清 0。 0: 无外部事件输入 1: 检测到外部输入（如果 EXENR = 1，由硬件设 1） 注意：捕获模式下，该位软件清 0 之前不允许更新 TnRCAP 值
0	TIF	定时器溢出标志位该位由硬件置 1，通过软件写 1 清 0。 0: 无溢出（必须由软件清 0） 1: 溢出（如果 RCLK = 0 和 TCLK = 0，由硬件设 1）
31~3	-	保留

#### 14.8.1.5 TnPWMA 占空比设定寄存器 TIMn\_PDTA (@CPRL = 0)

寄存器	读/写	说明	复位值	上电初始值
TIMn_PDTA (n=0~3)	读/写	TnPWMA 占空比设定寄存器 TIMn_PDTA (@CPRL = 0)	0x0000_0000	0x0000_0000

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

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

位编号	位符号	说明
15~0	PDT[15:0]	TPWMnA 占空比寄存器, n=0~3 TPWMnA 的波形的高电平宽度是 PDT[15:0]个 TIM 时钟
31~16	-	保留

#### 14.8.1.6 TnPWMB 占空比设定寄存器 TIMn\_PDTB (@CPRL = 0)

寄存器	读/写	说明	复位值	上电初始值
TIMn_PDTB (n=0~3)	读/写	TnPWMB 占空比设定寄存器 TIMn_PDTB (@CPRL = 0)	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
15~0	PDT[15:0]	TPWMnB 占空比寄存器, n=0~3 TPWMnB 的波形的高电平宽度是 PDT[15:0]个 TIM 时钟
31~16	-	保留

#### 14.8.1.7 上升沿数据捕获寄存器 TIMn\_RCAP (@CPRL = 1)

寄存器	读/写	说明	复位值	上电初始值
TIMn_RCAP (n=0~3)	读/写	上升沿数据捕获寄存器 TIMn_RCAP (@CPRL = 1)	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
RCAP[15:8]							

7	6	5	4	3	2	1	0
RCAP[7:0]							

位编号	位符号	说明
15~0	RCAP [15:0]	TIMn 的 PWM 捕获模式下，当上升沿捕获条件发生，CNT 计数器值将被保存到该寄存器。
31~16	-	保留

#### 14.8.1.8 下降沿数据捕获寄存器 TIMn\_FCAP (@CPRL = 1)

寄存器	读/写	说明	复位值	上电初始值
TIMn_FCAP (n=0~3)	读/写	下降沿数据捕获寄存器 TIMn_FCAP (@CPRL = 1)	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
15~0	FCAP[15:0]	TIMn 的 PWM 捕获模式下，当下降沿捕获条件发生，CNT 计数器值将被保存到该寄存器。
31~16	-	保留

#### 14.8.1.9 TIMn 的中断使能及 DMA 控制寄存器 TIMn\_IDE

寄存器	读/写	说明	复位值	上电初始值
TIMn_IDE n=0~3	读/写	TIMn 的中断使能及 DMA 控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
-	CAPFDE	CAPRDE	TIDE	EXFIE	EXRIE	TIE	INTEN

位编号	位符号	说明
6	CAPFDE	下降沿捕获事件触发 DMA 请求使能位 0: 下降沿捕获事件禁止产生 DMA 请求 1: 产生新的下降沿捕获时, 触发 DMA 请求
5	CAPRDE	上升沿捕获事件触发 DMA 请求使能位 0: 上升沿捕获事件禁止产生 DMA 请求 1: 产生新的上升沿捕获时, 触发 DMA 请求
4	TIDE	定时器溢出事件触发 DMA 请求使能位 0: 定时器溢出禁止产生 DMA 请求 1: 定时器溢出允许产生 DMA 请求
3	EXFIE	外部事件输入下降沿中断开关 0: 禁止下降沿中断 1: 使能下降沿中断
2	EXRIE	外部事件输入上升沿中断开关 0: 禁止上升沿中断 1: 使能上升沿中断
1	TIE	定时器溢出中断开关 0: 禁止溢出中断 1: 使能溢出中断
0	INTEN	中断请求 CPU 的使能控制位 0: 禁止中断请求 1: 使能中断请求
31~7	-	保留

#### 14.8.2 TIM 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值
TIM0 基地址: 0x4002_0100					
TIM0_CON	0x00	读/写	定时器控制寄存器	0x0000_0000	0x0000_0000
TIM0_CNT	0x04	读/写	定时器计数值寄存器	0x0000_0000	0x0000_0000
TIM0_RLD	0x08	读/写	定时器重载寄存器	0x0000_0000	0x0000_0000
TIM0_STS	0x0C	读/写	定时器标志位寄存器	0x0000_0000	0x0000_0000
TIM0_PDTA	0x10	读/写	TnPWMA 占空比设定寄存器 (@CPRL = 0)	0x0000_0000	0x0000_0000
TIM0_RCAP	0x10	读/写	上升沿数据捕获寄存器 (@CPRL = 1)	0x0000_0000	0x0000_0000
TIM0_PDTB	0x14	读/写	TnPWMB 占空比设定寄存器 (@CPRL = 0)	0x0000_0000	0x0000_0000
TIM0_FCAP	0x14	读/写	下降沿数据捕获寄存器 (@CPRL = 1)	0x0000_0000	0x0000_0000
TIM0_IDE	0x18	读/写	TIMn 的中断使能及 DMA 控制寄存器	0x0000_0000	0x0000_0000
TIM1 基地址: 0x4002_0140					
TIM1_CON	0x00	读/写	定时器控制寄存器	0x0000_0000	0x0000_0000
TIM1_CNT	0x04	读/写	定时器计数值寄存器	0x0000_0000	0x0000_0000

寄存器	偏移地址	读/写	说明	复位值	上电初始值
TIM1_RLD	0x08	读/写	定时器重载寄存器	0x0000_0000	0x0000_0000
TIM1_STS	0x0C	读/写	定时标志位寄存器	0x0000_0000	0x0000_0000
TIM1_PDTA	0x10	读/写	TnPWMA 占空比设定寄存器 (@CPRL = 0)	0x0000_0000	0x0000_0000
TIM1_RCAP	0x10	读/写	上升沿数据捕获寄存器 (@CPRL = 1)	0x0000_0000	0x0000_0000
TIM1_PDTB	0x14	读/写	TnPWMB 占空比设定寄存器 (@CPRL = 0)	0x0000_0000	0x0000_0000
TIM1_FCAP	0x14	读/写	下降沿数据捕获寄存器 (@CPRL = 1)	0x0000_0000	0x0000_0000
TIM1_IDE	0x18	读/写	TIMn 的中断使能及 DMA 控制寄存器	0x0000_0000	0x0000_0000
TIM2 基地址: 0x4002_1100					
TIM2_CON	0x00	读/写	定时器控制寄存器	0x0000_0000	0x0000_0000
TIM2_CNT	0x04	读/写	定时器计数值寄存器	0x0000_0000	0x0000_0000
TIM2_RLD	0x08	读/写	定时器重载寄存器	0x0000_0000	0x0000_0000
TIM2_STS	0x0C	读/写	定时标志位寄存器	0x0000_0000	0x0000_0000
TIM2_PDTA	0x10	读/写	TnPWMA 占空比设定寄存器 (@CPRL = 0)	0x0000_0000	0x0000_0000
TIM2_RCAP	0x10	读/写	上升沿数据捕获寄存器 (@CPRL = 1)	0x0000_0000	0x0000_0000
TIM2_PDTB	0x14	读/写	TnPWMB 占空比设定寄存器 (@CPRL = 0)	0x0000_0000	0x0000_0000
TIM2_FCAP	0x14	读/写	下降沿数据捕获寄存器 (@CPRL = 1)	0x0000_0000	0x0000_0000
TIM2_IDE	0x18	读/写	TIMn 的中断使能及 DMA 控制寄存器	0x0000_0000	0x0000_0000
TIM3 基地址: 0x4002_1140					
TIM3_CON	0x00	读/写	定时器控制寄存器	0x0000_0000	0x0000_0000
TIM3_CNT	0x04	读/写	定时器计数值寄存器	0x0000_0000	0x0000_0000
TIM3_RLD	0x08	读/写	定时器重载寄存器	0x0000_0000	0x0000_0000
TIM3_STS	0x0C	读/写	定时标志位寄存器	0x0000_0000	0x0000_0000
TIM3_PDTA	0x10	读/写	TnPWMA 占空比设定寄存器 (@CPRL = 0)	0x0000_0000	0x0000_0000
TIM3_RCAP	0x10	读/写	上升沿数据捕获寄存器 (@CPRL = 1)	0x0000_0000	0x0000_0000
TIM3_PDTB	0x14	读/写	TnPWMB 占空比设定寄存器 (@CPRL = 0)	0x0000_0000	0x0000_0000
TIM3_FCAP	0x14	读/写	下降沿数据捕获寄存器 (@CPRL = 1)	0x0000_0000	0x0000_0000

寄存器	偏移地址	读/写	说明	复位值	上电初始值
TIM3_IDE	0x18	读/写	TIMn 的中断使能及 DMA 控制寄存器	0x0000_0000	0x0000_0000

## 15 省电模式

初始上电默认运行在常规模式，即 Normal Mode，额外提供三种省电模式：

- 低速模式：系统时钟源可选择 LIRC，CPU 可工作在 32kHz
- IDLE Mode，可由任何中断唤醒
- STOP Mode，可由 INT0~15、Base Timer 和 CMP 唤醒

## 16 GPIO

### 16.1 时钟源

M0+内核可通过 IOPORT 总线实现单周期访问 GPIO，数据传输效率极高。IOPORT 总线的时钟来自 HCLK。

### 16.2 特性

SC32F15G 系列 GPIO 端口特性如下：

- 最大 44 个双向可独立控制的 GPIO
- CPU 可在单周期内通过 IOPORT 总线访问 GPIO 端口
- 可独立设定上拉电阻
- 所有口源驱动能力分四级控制
- 全部 I/O 具有大灌电流驱动能力（50mA）
- 16 个 I/O 一组
- I/O 端口在输入或输出状态下，从端口数据寄存器里读到的都是端口的实际状态值

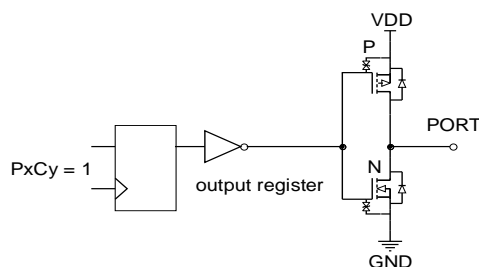
注意：未使用及封装未引出的端口均要设置为强推挽输出模式。

### 16.3 GPIO 结构图

#### 16.3.1 强推挽输出模式

强推挽输出模式下，能够提供持续的大电流驱动：相关电气参数详见《GPIO 参数》章节

强推挽输出模式的端口结构示意图如下：



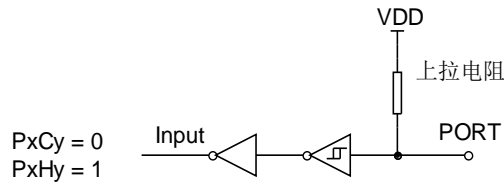
强推挽输出模式

#### 16.3.2 带上拉的输入模式

带上拉的输入模式下，输入口上恒定接一个上拉电阻，仅当输入口上电平被拉低时，才会检测到低电平信

号。

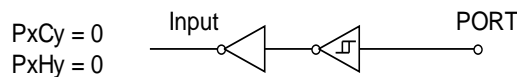
带上拉的输入模式的端口结构示意图如下：



带上拉的输入模式

### 16.3.3 高阻输入模式 (Input only)

高阻输入模式的端口结构示意图如下所示：



高阻输入模式

## 16.4 GPIO 寄存器

### 16.4.1 GPIO 相关寄存器表

#### 16.4.1.1 PX 端口数据寄存器 PX

寄存器	读/写	说明	复位值	上电初始值
PX X=A,B,C	读/写	PX 端口数据寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8
7	6	5	4	3	2	1	0
PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0

位编号	位符号	说明
15~0	PDn (n=0~15)	PXn 端口数据寄存器, X=A,B,C, n=0~15 端口锁存寄存器数据, 从端口数据寄存器里读到的都是端口的实际状态值。

位编号	位符号	说明
31~16	-	保留

#### 16.4.1.2 PX 端口数据寄存器 PXn\_BIT

寄存器	读/写	说明	复位值	上电初始值
PXn_BIT X=A,B,C n=0~15	读/写	PX 端口数据寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
0	BSRn	PXn 端口位赋值控制, n=0~15 用于对 PXn 端口单独赋值
31~1	-	保留

#### 16.4.1.3 PX 端口数据寄存器 PXn\_XR

寄存器	读/写	说明	复位值	上电初始值
PXn_XR X=A,B,C n=0~15	读/写	对 PXn 翻转	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
0	XRn	PXn 端口位翻转控制, n=0~15 0: 无效 1: 对 PXn 输出进行翻转
31~1	-	保留

#### 16.4.1.4 PX 端口输入/输出控制寄存器 PXCON

寄存器	读/写	说明	复位值	上电初始值
PXCON X=A,B,C	读/写	PX 端口输入/输出控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
MODE15	MODE14	MODE13	MODE12	MODE11	MODE10	MODE9	MODE8
7	6	5	4	3	2	1	0
MODE7	MODE6	MODE5	MODE4	MODE3	MODE2	MODE1	MODE0

位编号	位符号	说明
15~0	MODEn (n=0~15)	PXn 端口强推挽模式使能位, n=0~15 0: PXn 为输入模式 (上电初始值) 1: PXn 为强推挽输出模式
31~16	-	保留

#### 16.4.1.5 PX 端口上拉电阻控制寄存器 PXPB

寄存器	读/写	说明	复位值	上电初始值
PXPB X=A,B,C	读/写	PX 端口上拉电阻控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
PUPD15	PUPD14	PUPD13	PUPD12	PUPD11	PUPD10	PUPD9	PUPD8
7	6	5	4	3	2	1	0
PUPD7	PUPD6	PUPD5	PUPD4	PUPD3	PUPD2	PUPD1	PUPD0

位编号	位符号	说明
15~0	PUPDn (n=0~15)	PXn 端口上拉电阻使能位, n=0~15 0: PXn 为高阻输入模式 (上电初始值), 上拉电阻关闭; 1: PXn 上拉电阻打开
31~16	-	保留

#### 16.4.1.6 GPIO 驱动等级寄存器 PXLEV

寄存器	读/写	说明	复位值	上电初始值
PXLEV X=A,B,C	读/写	GPIO 驱动等级寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
LEV15[1:0]		LEV14[1:0]		LEV13[1:0]		LEV12[1:0]	
23	22	21	20	19	18	17	16
LEV11[1:0]		LEV10[1:0]		LEV9[1:0]		LEV8[1:0]	
15	14	13	12	11	10	9	8
LEV7[1:0]		LEV6[1:0]		LEV5[1:0]		LEV4[1:0]	
7	6	5	4	3	2	1	0
LEV3[1:0]		LEV2[1:0]		LEV1[1:0]		LEV0[1:0]	

位编号	位符号	说明
31~0	LEVn[1:0] (n=0~15)	PXn 端口等级控制位, n=0~15 用于设置 PXn 端口的 IOH 等级: 00: 等级 0 (最大); 01: 等级 1; 10: 等级 2; 11: 等级 3 (最小);

#### 16.4.2 GPIO 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值
PA 基地址: 0x4001_1000					
PA	0x00	读/写	PA 端口数据寄存器	0x0000_0000	0x0000_0000
PACON	0x20	读/写	PA 口输入/输出控制寄存器	0x0000_0000	0x0000_0000
PAPH	0x40	读/写	PA 口上拉电阻控制寄存器	0x0000_0000	0x0000_0000
PALEV	0x60	读/写	IOH 设置寄存器 IO 驱动等级	0x0000_0000	0x0000_0000

寄存器	偏移地址	读/写	说明	复位值	上电初始值
PB 基地址: 0x4001_1100					
PB	0x00	读/写	PB 端口数据寄存器	0x0000_0000	0x0000_0000
PBCON	0x20	读/写	PB 口输入/输出控制寄存器	0x0000_0000	0x0000_0000
PBPH	0x40	读/写	PB 口上拉电阻控制寄存器	0x0000_0000	0x0000_0000
PBLEV	0x60	读/写	IOH 设置寄存器 IO 驱动等级	0x0000_0000	0x0000_0000

寄存器	偏移地址	读/写	说明	复位值	上电初始值
PC 基地址: 0x4001_1200					
PC	0x00	读/写	PC 端口数据寄存器	0x0000_0000	0x0000_0000
PCCON	0x20	读/写	PC 口输入/输出控制寄存器	0x0000_0000	0x0000_0000

寄存器	偏移地址	读/写	说明	复位值	上电初始值
PCPH	0x40	读/写	PC 口上拉电阻控制寄存器	0x0000_0000	0x0000_0000
PCLEV	0x60	读/写	IOH 设置寄存器 IO 驱动等级	0x0000_0000	0x0000_0000

## 17 UART0~2

### 17.1 时钟源

- SC32F15G 系列 UART 的时钟源仅一种，来自 PCLK

### 17.2 特性

- 三个 UART：UART0~2
- UART2 具有完整的 LIN 接口：
  - 主从模式可切换
  - 支持主机模式下硬件 break 发送（10/13 Bits）
  - 支持从机模式下硬件 break 检测（10/11 Bits）
  - 支持从机模式下波特率同步
  - 提供相关中断/状态位/标志位/容错范围
- 每个 UART 有三种通讯模式可选：
  - 模式 0，8 位半双工同步通信模式，在 RXD 引脚上收发串行数据。TXD 引脚用作发送移位时钟。每帧收发 8 位，低位先接收或发送；
  - 模式 1，10 位全双工异步通信，由 1 个起始位，8 个数据位和 1 个停止位组成，通信波特率可变；
  - 模式 3，11 位全双工异步通信，由 1 个起始位，8 个数据位，一个可编程的第 9 位和 1 个停止位组成，通信波特率可变。
- 发送和接收完成可产生中断并置起对应的标志位 TXIF 和 RXIF，中断标志需要软件清除
- UART0 和 UART1 可产生 DMA 请求
- UART2 不能产生 DMA 请求
- UART0/1/2 均支持信号口映射，均可映射到另外一组 IO
- 独立波特率发生器
- UART0/1 支持从 STOP Mode 唤醒：
  - START 位下降沿可唤醒 STOP Mode
  - 提供对应的唤醒中断使能位 WKIE 及唤醒中断标志位 WKIF

**注意：UART0 的信号引脚支持两种映射方案：**

- 映射 1：RXD0 / TXD0
- 映射 2：RXD0A / TXD0A

当选择映射 1（RXD0/TXD0）时，该组引脚与系统烧录/调试接口（T\_CLK / T\_DIO）复用。在此映射下，若启用全双工通信，T\_CLK / T\_DIO 可能会与 UART0 的 RXD0 接收时序产生冲突，导致通信异常。因此，选择映射 1 时必须将 UART0 配置为半双工通信模式，以避免此硬件冲突并保证通信稳定性。

如需使用全双工 UART 通信，请将管脚映射至映射 2（RXD0A/TXD0A）。

## 17.3 UART2-LIN

UART2 支持标准的 LIN 通信协议

### 17.3.1 LIN 帧结构

根据 LIN 协议，所有的传输信息被打包为帧。一个帧由一个报头（主机任务提供）和一个紧跟其后的应答（从机任务提供）组成。报头（主机任务提供）由一个 **break** 域和一个 **sync**（同步）域再跟一个帧识别码（**frame ID**）组成。帧 ID 仅作为定义帧的用途。从机任务负责回应相关的帧 ID。响应由一个数据域和一个校验域组成。下图是 LIN 帧的结构



### 17.3.2 LIN 主机模式

通过设置 FUNCSEL=1, SLVEN=0, UART 控制器支持 LIN 主机模式。在 LIN 模式，根据 LIN 的标准，每个字节由值为 0（显性）的 START 位开始，接着是 8 位数据位，没有校验位，LSB 优先，由一个值为 1（隐性）的 STOP 位结束。使能并初始化 LIN 主机模式需要如下步骤：

- ① 设置 UART\_BAUD 寄存器设定波特率。
- ② 设置 FUNCSEL=1, 选择 LIN 功能模式
- ③ 设置 SM[1:0]=01 配置 UART 为模式 1

一个完整的报头由一个 **break** 域和同步域再跟一个帧标识符（帧 ID）组成。UART 控制器可以选择“**break** 域”作为发送的报头。“同步域”和“帧 ID 域”需要用户通过软件写入，即：发送一个完整的报头到总线，软件必须依次填同步数据（0x55）和帧 ID 数据到 UART\_DAT 寄存器。

### 17.3.3 LIN 从机模式

通过设置 FUNCSEL=1, SLVEN=1, UART 控制器支持 LIN 从机模式。在 LIN 模式，根据 LIN 的标准，每一个字节域都是由一个值为 0 的显性位开始的，跟着 8 个数据位，没有校验位，最低位在前，由一个值为 1 的隐性 stop 位结束。

LIN 从机模式初始化的流程如下：

- ① 设置 UART\_BAUD 寄存器设定波特率。
- ② 设置 FUNCSEL=1, 选择 LIN 功能模式。
- ③ 设定 SM[1:0]=1 配置 UART 为模式 1
- ④ 设定 SLVEN=1, 使能 LIN 从机模式

LIN 从模式下，通过设定 LBDL 使能从机 **break** 域检测功能来侦测接收“**break** 域”。接收到一个 **break** 后，BKIF 标志将被置位。如果 BKIE =1 中断将发生。为了避免比特率偏差，控制器支持自动重同步功能，

避免时钟误差错误，通过设定 SLVAREN 使能该特性。

### 17.3.3.1 同步域误差错误

自动重新同步模式下，控制器将检测同步域的误差错误。误差错误检测比较当前波特率和接收到的同步域的波特率。两个检测被同步执行。

检查 1：根据同步域的第一个下降沿和最后一个下降沿的测量值

- 如果误差大于 15%，报头错误标志 SLVHEIF 将被置位
- 如果误差在 14% 和 15%之间，报头错误标志 SLVHEIF 可能被置位也可能没有被置位（取决于数据失相）

检查 2：根据同步域的每一个下降沿的测量值

- 如果误差大于 19%，报头错误标志 SLVHEIF 将被置位
- 如果误差在 15% 和 19%之间，报头错误标志 SLVHEIF 可能被置位也可能没有被置位（取决于数据失相）

注：误差检测基于当前波特率时钟。因而，为了保证误差检测的正确性，建议用户在新的 break 域收到之前，通过软件将波特率重新加载为初始值。

## 17.4 UART 中断

对于 UARTn, n=0~2，在发生“唤醒”或“数据收发完成”时都会生成中断。可以使用单独的中断使能位以提高灵活性。

中断事件	中断请求控制位	事件标志位	中断使能子开关
UART 从 STOP Mode 唤醒	UARTn_IDE ->INTEN	WKIF	WKIE
数据发送完成		TXIF	TXIE
数据接收完成		RXIF	RXIE
断开中断	UART2_IDE ->INTEN	BKIF	BKIE
LIN 从机侦测到报头错误		SLVHEIF	SLVHEIE
波特率同步完成		SYNCIF	SYNCIE

## 17.5 UART0/1 寄存器

### 17.5.1 UART0/1 相关寄存器表

#### 17.5.1.1 UART 控制寄存器 UARTn\_CON

寄存器	读/写	说明	复位值	上电初始值
UARTn_CON (n=0/1)	读/写	UART 控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	SPOS	-	-	-	-	-
7	6	5	4	3	2	1	0
TXEN	RXEN	-	PRESCALER	-	SM2	SM1	SM0

位编号	位符号	说明																		
13	SPOS	<p>● UART0 信号口映射控制位@UART0_CON</p> <table border="1"> <tr> <td style="text-align: center;">信号</td> <td>RXD0</td> <td>TXD0</td> </tr> <tr> <td>SPOS=0</td> <td>PC7</td> <td>T_DIO 所在端口</td> </tr> <tr> <td>SPOS=1</td> <td>PB6</td> <td>PB5</td> </tr> </table> <p>● UART1 信号口映射控制位@UART1_CON</p> <table border="1"> <tr> <td style="text-align: center;">信号</td> <td>RXD1</td> <td>TXD1</td> </tr> <tr> <td>SPOS=0</td> <td>PC1</td> <td>PC2</td> </tr> <tr> <td>SPOS=1</td> <td>PB3</td> <td>PB4</td> </tr> </table>	信号	RXD0	TXD0	SPOS=0	PC7	T_DIO 所在端口	SPOS=1	PB6	PB5	信号	RXD1	TXD1	SPOS=0	PC1	PC2	SPOS=1	PB3	PB4
信号	RXD0	TXD0																		
SPOS=0	PC7	T_DIO 所在端口																		
SPOS=1	PB6	PB5																		
信号	RXD1	TXD1																		
SPOS=0	PC1	PC2																		
SPOS=1	PB3	PB4																		
7	TXEN	<p>UART 发送控制位</p> <p>0: TXD 信号不再影响所在管脚的状态。此时若用户程序限制发送功能，仅用到了接收，那么与 TXD 脚复用的其它功能将不受影响；但此状态下向发送缓存中写值依然能触发 UART 发送，发送时 TXD 脚切换为 TXD 信号口，发送完毕后 TXD 脚切换回默认复用状态。</p> <p>1: 允许发送数据，TXD 所在管脚切换为 TXD 信号口模式</p>																		
6	RXEN	<p>UART 接收控制位</p> <p>0: 不允许接收数据</p> <p>1: 允许接收数据</p>																		
4	PRESCALER	<p>波特率倍率设置位</p> <p>该位在 UART 的不同模式下有不同定义：</p> <ul style="list-style-type: none"> <li>● 当 SM0~1 = 01 (UART 模式 1) 或 SM0~1 = 11 (UART 模式 3) : <ul style="list-style-type: none"> <li>■ 0: 串行端口在系统时钟的 1 分频下运行</li> <li>■ 1: 串行端口在系统时钟的 16 分频下运行</li> </ul> </li> <li>● 当 SM0~1 = 00 (UART 模式 0) 波特率倍率设置位: <ul style="list-style-type: none"> <li>■ 0: 串行端口在系统时钟的 12 分频下运行</li> <li>■ 1: 串行端口在系统时钟的 4 分频下运行</li> </ul> </li> </ul>																		
2	SM2	<p>SBUF8 置位中断允许位</p> <p>该位仅在模式 3 有效</p> <p>0: 每收到一个完整的数据帧就置位 RI 产生中断请求；</p> <p>1: 收到一个完整的数据帧时，只有当 SBUF8=1 时才会置位 RI 产生中断请求</p>																		

位编号	位符号	说明
1~0	SM[1:0]	<p>UART 通信模式控制位</p> <p>00: 模式 0, 8 位半双工同步通信模式, 在 RXD 引脚上收发串行数据。TXD 引脚用作发送移位时钟。每帧收发 8 位, 低位先接收或发送。在该模式下使能 RXEN 位, UART 将发出一个完整一帧的时钟, 并且将 RXIF 置 1;</p> <p>01: 模式 1, 10 位全双工异步通信, 由 1 个起始位, 8 个数据位和 1 个停止位组成, 通信波特率可变;</p> <p>10: 保留</p> <p>11: 模式 3, 11 位全双工异步通信, 由 1 个起始位, 8 个数据位, 一个可编程的第 9 位和 1 个停止位组成, 通信波特率可变</p>
31~14 12~8 5 3	-	保留

### 17.5.1.2 UART 标志状态位寄存器 UARTn\_STS

寄存器	读/写	说明	复位值	上电初始值
UARTn_STS (n=0/1)	读/写	UART 标志状态位寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
4	WKIF	<p>UART 唤醒标志位</p> <p>UART 从 STOP Mode 唤醒后, 该位由硬件置 1, 如果此时 WKIE=1, 将产生中断。</p> <p>该位由软件写 1 清 0。</p>
1	TXIF	<p>发送中断标志位</p> <p>数据发送完成该位由硬件置 1, 如果此时 TXIE = 1, 将产生中断。</p> <p>该位由软件写 1 清 0。</p> <p>注意: 在 DMA 模式下, DMA 写入发送缓存后, 该位由 DMA 模块清 0, 此时用户无需通过软件清 0。</p>
0	RXIF	<p>接收中断标志位</p> <p>数据接收完成该位由硬件置 1, 如果此时 RXIE = 1, 将产生中断。</p> <p>该位由软件写 1 清 0。</p> <p>注意: 在 DMA 模式下, DMA 读取接收缓存后, 该位由 DMA 模块清 0, 此时用户无需通过软件清 0。</p>
31~5 3~2	-	保留

### 17.5.1.3 UART 波特率配置寄存器 UARTn\_BAUD

寄存器	读/写	说明	复位值	上电初始值
UARTn_BAUD (n=0/1)	读/写	UART 波特率配置寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
15~0	BAUD[15:0]	UART 波特率配置位 写入 BAUD[15:0]后, UART 的波特率将按照以下公式配置: $BaudRate=f_{UART} / BAUD[15:0]$ 其中 $f_{UART}$ 为 UART 时钟源经预分频之后的最终频率, 见 PRESCALER 位说明。 <b>注意: BAUD[15:0]必须大于 0x0010。</b>
31~16	-	保留

### 17.5.1.4 UART 数据寄存器 UARTn\_DATA

寄存器	读/写	说明	复位值	上电初始值
UARTn_DATA (n=0/1)	读/写	UART 数据寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	SBUF8
7	6	5	4	3	2	1	0
SBUF[7:0]							

位编号	位符号	说明
8	SBUF8	UART 的发送/接收数据的第 9 位 该位仅在模式 3 有效。
7~0	SBUF[7:0]	UART 数据缓存 读操作: 将返回接收锁存器中的内容 写操作: SBUF 的数据将送至发送移位寄存器, 并启动发送流程
31~9	-	保留

### 17.5.1.5 UART 的中断使能及 DMA 控制寄存器 UARTn\_IDE

寄存器	读/写	说明	复位值	上电初始值
UARTn_IDE (n=0/1)	读/写	UART 的中断使能及 DMA 控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
----	----	----	----	----	----	----	----

-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
TXDMAEN	RXDMAEN	-	WKIE	-	TXIE	RXIE	INTEN

位编号	位符号	说明
7	TXDMAEN	DMA 发送通道使能位 0: 禁用 DMA 发送功能 1: 使能 DMA 发送功能 该位使能后, TXIF 置起可触发 DMA 通道发送请求
6	RXDMAEN	DMA 接收通道使能位 0: 禁用 DMA 接收功能 1: 使能 DMA 接收功能 该位使能后, RXIF 置起可触发 DMA 通道接收请求
4	WKIE	UART 唤醒中断使能位 0: WKIF 置起时, 不允许产生中断 1: WKIF 置起时, 产生中断
2	TXIE	UART 发送中断使能位 0: TXIF 置起时, 不允许产生中断 1: TXIF 置起时, 产生中断
1	RXIE	UART 接收中断使能位 0: RXIF 置起时, 不允许产生中断 1: RXIF 置起时, 产生中断
0	INTEN	中断请求 CPU 的使能控制位 0: 禁止中断请求 1: 使能中断请求
31~8 5 3	-	保留

### 17.5.2 UART0~1 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值	访问限制
UART0 基地址: 0x4002_0020						
UART0_CON	0x00	读/写	UART 控制寄存器	0x0000_0000	0x0000_0000	-
UART0_STS	0x04	读/写	UART 标志位寄存器	0x0000_0000	0x0000_0000	-
UART0_BAUD	0x08	读/写	UART 波特率寄存器	0x0000_0000	0x0000_0000	-
UART0_DATA	0x0C	读/写	UART 数据寄存器	0x0000_0000	0x0000_0000	不支持字节/ 半字访问
UART0_IDE	0x10	读/写	通信口 DMA 控制寄存器	0x0000_0000	0x0000_0000	-

寄存器	偏移地址	读/写	说明	复位值	上电初始值	访问限制
UART1 基地址: 0x4002_0080						
UART1_CON	0x00	读/写	UART 控制寄存器	0x0000_0000	0x0000_0000	-
UART1_STS	0x04	读/写	UART 标志位寄存器	0x0000_0000	0x0000_0000	-
UART1_BAUD	0x08	读/写	UART 波特率寄存器	0x0000_0000	0x0000_0000	-
UART1_DATA	0x0C	读/写	UART 数据寄存器	0x0000_0000	0x0000_0000	不支持字节/半字访问
UART1_IDE	0x10	读/写	通信口 DMA 控制寄存器	0x0000_0000	0x0000_0000	-

## 17.6 UART2 寄存器

### 17.6.1 UART2 相关寄存器表

#### 17.6.1.1 UART 控制寄存器 UARTn\_CON

寄存器	读/写	说明	复位值	上电初始值
UARTn_CON (n=2)	读/写	UART 控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	BKSIZE	-	BKTR
23	22	21	20	19	18	17	16
-	-	-	-	-	LBDL	SLVAREN	SLVEN
15	14	13	12	11	10	9	8
-	-	SPOS	-	-	-	-	FUNCSEL
7	6	5	4	3	2	1	0
TXEN	RXEN	-	PRESCALER	-	SM2	SM1	SM0

位编号	位符号	说明
26	BKSIZE	间隔段生成位长度选择位 0: 选择用 10 位时间长度发送 1: 选择用 13 位时间长度发送
24	BKTR	LIN 模式, 断开字符发送触发位 0: 不发送断开字符 1: 发送断开字符 <b>注意:</b> 1. 当 LIN 是从机模式 (SLVEN=1) 时, 此位无效 2. 当 break 域发送完之后, 此位自动清除
18	LBDL	LIN 断路检测长度选择位 0: 选择 10 位中断检测 1: 选择 11 位中断检测 <b>注意:</b>

位编号	位符号	说明									
		<ol style="list-style-type: none"> <li>1. 检测长度需要配合主机发送间隔长度进行选择；</li> <li>2. 当 LIN 是主机模式 (SLVEN=0) 时，此位无效。</li> </ol>									
17	SLVAREN	<p>波特率自动重同步使能位</p> <p>0: LIN 从机自动重同步模式禁止 1: LIN 从机自动重同步模式使能</p> <p>说明：当自动重新同步功能使能后，每个 LIN break 域后面，用 LIN 的工作时钟持续采样 5 个下降沿的时间，测量的结果储存在内部波特率缓存寄存器中，UARTn_BAUD 寄存器的值将被自动更新</p> <p>注意：当 LIN 是主机模式 (SLVEN=0) 时，此位无效。</p>									
16	SLVEN	<p>LIN 从模式使能位</p> <p>0: LIN 从模式禁止 (LIN 主模式使能) 1: LIN 从模式使能 (LIN 主模式禁止)</p> <p>说明：</p> <ol style="list-style-type: none"> <li>1. LIN 主模式下，SLVAREN、LBDL 无效</li> <li>2. LIN 从模式下，可检测断路</li> </ol>									
13	SPOS	<p>● UART2 信号口映射控制位 @UART2_CON</p> <table border="1" style="margin-left: 20px;"> <thead> <tr> <th style="text-align: center;">信号</th> <th>RXD2</th> <th>TXD2</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">SPOS=0</td> <td style="text-align: center;">PA1</td> <td style="text-align: center;">PA0</td> </tr> <tr> <td style="text-align: center;">SPOS=1</td> <td style="text-align: center;">PB1</td> <td style="text-align: center;">PB0</td> </tr> </tbody> </table>	信号	RXD2	TXD2	SPOS=0	PA1	PA0	SPOS=1	PB1	PB0
信号	RXD2	TXD2									
SPOS=0	PA1	PA0									
SPOS=1	PB1	PB0									
8	FUNCSEL	<p>通讯模式选择位</p> <p>0: UART 功能 1: LIN 功能使能，此时 LIN 硬件模块和 UART 模块是同时开启，LIN 模块负责 break 检测/发送、波特率同步/更新</p> <p>注意：FUNCSEL = 1 时，UART2_CON 的 bit 31~16 才有效</p>									
7	TXEN	<p>UART 发送控制位</p> <p>0: TXD 信号不再影响所在管脚的状态。此时若用户程序限制发送功能，仅用到了接收，那么与 TXD 脚复用的其它功能将不受影响；但此状态下向发送缓存中写值依然能触发 UART 发送，发送时 TXD 脚切换为 TXD 信号口，发送完毕后 TXD 脚切换回默认复用状态。</p> <p>1: 允许发送数据，TXD 所在管脚切换为 TXD 信号口模式</p>									
6	RXEN	<p>UART 接收控制位</p> <p>0: 不允许接收数据 1: 允许接收数据</p>									
4	PRESCALER	<p>波特率倍率设置位</p> <p>该位在 UART 的不同模式下有不同定义：</p> <ul style="list-style-type: none"> <li>● 当 SM0~1 = 01 (UART 模式 1) 或 SM0~1 = 11 (UART 模式 3)： <ul style="list-style-type: none"> <li>■ 0: 串行端口在系统时钟的 1 分频下运行</li> <li>■ 1: 串行端口在系统时钟的 16 分频下运行</li> </ul> </li> <li>● 当 SM0~1 = 00 (UART 模式 0) 波特率倍率设置位： <ul style="list-style-type: none"> <li>■ 0: 串行端口在系统时钟的 12 分频下运行</li> <li>■ 1: 串行端口在系统时钟的 4 分频下运行</li> </ul> </li> </ul>									

位编号	位符号	说明
2	SM2	<b>SBUF8 置位中断允许位</b> 该位仅在模式 3 有效 0: 每收到一个完整的数据帧就置位 RI 产生中断请求; 1: 收到一个完整的数据帧时, 只有当 SBUF8=1 时才会置位 RI 产生中断请求
1~0	SM[1:0]	<b>UART 通信模式控制位</b> 00: 保留; 01: 模式 1, 10 位全双工异步通信, 由 1 个起始位, 8 个数据位和 1 个停止位组成, 通信波特率可变; 10: 保留 11: 模式 3, 11 位全双工异步通信, 由 1 个起始位, 8 个数据位, 一个可编程的第 9 位和 1 个停止位组成, 通信波特率可变
31~27 25 23~19 15~14 12~9 5 3	-	保留

### 17.6.1.2 UART 标志状态位寄存器 UARTn\_STS

寄存器	读/写	说明	复位值	上电初始值
UARTn_STS (n=2)	读/写	UART 标志状态位寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
11	SYNCIF	<b>LIN 模式下, 波特率同步完成标志位</b> 该位会在同步域 0x55 后置起
10	SLVSYNIF	<b>LIN 从机同步域状态位</b> 该位表示在自动重同步模式, LIN 同步域正在被分析。 正常通讯情况下, 该位只读, 作状态位使用 当接收器报头检测到一些错误, 用户可以对该位写 1 以复位内部电路来重新搜索新的帧报头  0: 当前字符不在 LIN 同步状态 1: 当前字符在 LIN 同步状态  注意: 1. 该位仅在 LIN 从机模式有效 2. 当向该位写 1, 硬件会重载初始波特率并重新搜索新的帧报头
9	SLVHEIF	<b>LIN 从机报头错误标志</b> 在 LIN 从机模式, 当检测到一个 LIN 报头错误时, 该位由硬件置 1, 向该位写 1 清 0

位编号	位符号	说明
		<p>0: LIN 未检测到报头错误 1: LIN 检测到报头错误</p> <p>错误情况包括:</p> <ol style="list-style-type: none"> <li>1. break 间隔符太短 (小于 0.5 位的时间)</li> <li>2. 在非自动重同步模式同步域数据不是 0x55</li> <li>3. 自动重同步模式同步域偏离错误</li> </ol> <p>注意: 该位仅在 LIN 从机模式时有效。</p>
8	BKIF	<p>LIN 模式断开中断标志位 break 域发送或接收完毕后该位由硬件置 1, 如果此时 BKIE = 1, 将产生中断。该位由软件写 1 清 0。</p>
1	TXIF	<p>发送中断标志位 数据发送完成该位由硬件置 1, 如果此时 TXIE = 1, 将产生中断。该位由软件写 1 清 0。 <b>注意: 在 DMA 模式下, DMA 写入发送缓存后, 该位由 DMA 模块清 0, 此时用户无需通过软件清 0。</b></p>
0	RXIF	<p>接收中断标志位 数据接收完成该位由硬件置 1, 如果此时 RXIE = 1, 将产生中断。该位由软件写 1 清 0。 <b>注意: 在 DMA 模式下, DMA 读取接收缓存后, 该位由 DMA 模块清 0, 此时用户无需通过软件清 0。</b></p>
31~12 7~2	-	保留

### 17.6.1.3 UART 波特率配置寄存器 UARTn\_BAUD

寄存器	读/写	说明	复位值	上电初始值
UARTn_BAUD (n=2)	读/写	UART 波特率配置寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
15~0	BAUD[15:0]	<p>UART 波特率配置位 写入 BAUD[15:0]后, UART 的波特率将按照以下公式配置: <math>BaudRate=f_{UART} / BAUD[15:0]</math> 其中 <math>f_{UART}</math> 为 UART 时钟源经预分频之后的最终频率, 见 PRESCALER 位说明。 <b>注意: BAUD[15:0]必须大于 0x0010。</b></p>
31~16	-	保留

#### 17.6.1.4 UART 数据寄存器 UARTn\_DATA

寄存器	读/写	说明	复位值	上电初始值
UARTn_DATA (n=2)	读/写	UART 数据寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	SBUF8
7	6	5	4	3	2	1	0
SBUF[7:0]							

位编号	位符号	说明
8	SBUF8	UART 的发送/接收数据的第 9 位 该位仅在模式 3 有效。
7~0	SBUF[7:0]	UART 数据缓存 读操作：将返回接收锁存器中的内容 写操作：SBUF 的数据将送至发送移位寄存器，并启动发送流程
31~9	-	保留

#### 17.6.1.5 UART 的中断使能及 DMA 控制寄存器 UARTn\_IDE

寄存器	读/写	说明	复位值	上电初始值
UARTn_IDE (n=2)	读/写	UART 的中断使能及 DMA 控制寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
11	SYNCIE	LIN 模式下，波特率同步完成中断使能位
9	SLVHEIE	LIN 从机报头错误中断使能，仅在 LIN 从机模式时有效 LIN 从机侦测到一个 LIN 报头错误时，SLVHEIE 位由硬件置 1，如果此时 SLVHEIE=1，将产生中断。  错误情况包括： 1. break 间隔符太短（小于 0.5 位的时间） 2. 在非自动重同步模式同步域数据不是 0x55 3. 自动重同步模式同步域偏离错误
8	BKIE	LIN 模式断开中断控制位 0: BKIF 置起时，不允许产生中断 1: BKIF 置起时，产生中断

位编号	位符号	说明
2	TXIE	UART 发送中断使能位 0: TXIF 置起时, 不允许产生中断 1: TXIF 置起时, 产生中断
1	RXIE	UART 接收中断使能位 0: RXIF 置起时, 不允许产生中断 1: RXIF 置起时, 产生中断
0	INTEN	中断请求 CPU 的使能控制位 0: 禁止中断请求 1: 使能中断请求
31~12 10 7~3	-	保留

### 17.6.2 UART2 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值	访问限制
UART2 基地址: 0x4002_1020						
UART2_CON	0x00	读/写	UART 控制寄存器	0x0000_0000	0x0000_0000	-
UART2_STS	0x04	读/写	UART 标志位寄存器	0x0000_0000	0x0000_0000	-
UART2_BAUD	0x08	读/写	UART 波特率寄存器	0x0000_0000	0x0000_0000	-
UART2_DATA	0x0C	读/写	UART 数据寄存器	0x0000_0000	0x0000_0000	不支持字节/ 半字访问
UART2_IDE	0x10	读/写	通信口 DMA 控制寄存器	0x0000_0000	0x0000_0000	-

## 18 SPI0~1

### 18.1 时钟源

- SC32F15G 系列的 SPI 时钟仅一种，来自 PCLK

### 18.2 SPI0 特性

- 支持 11 档 SPI 时钟预分频，时钟预分频允许用户设置到较低频率，最小分频档位为  $f_{PCLK}/1024$ 。
- SPI0 信号口共两组 IO 映射可选
- SPI0 信号口强驱动：
  - SPI 通信模式下相应的信号口管脚输出驱动能力增强，其它模式下跟普通 IO 特性一致。
  - 映射信号口也可以变成强驱动，以保证 SPI0 在任意端口上的一致性
  - 具有 16 位 8 级 FIFO 缓存，发送接收独立
  - SPI0 的 FIFO 功能可以实现：连续向 SPI 发送缓存（SPI0\_DATA）写入 8 个或 8 个以内的 16 位发送数据，SPI 发送的时候，最先写入的数据也最先被发送。当用户写入 FIFO 的数据被发送完成，发送缓存器空标志 TXEIF 置 1；若 FIFO 的数据已满，则写入冲突标志位 WCOL 置位，用户无法向 FIFO 写入数据，直至 FIFO 内的数据被发送出去、FIFO 不满，用户才能写入数据。当 FIFO 内的数据全部发送完毕才置起中断标志 SPIF。
  - 连续从 SPI 接收缓存（SPI0\_DATA）读取 8 个或 8 个以内的 16 位接收数据，最先接收到的数据也最先被读取到。
  - FIFO 数据传输一半中断及对应标志位，方便用户及时读取/写入数据：
    - ◆ 提供发送 FIFO 有效数据不满一半中断及对应标志位 TXHIF
    - ◆ 提供接收 FIFO 超一半中断及对应标志位标志 RXHIF
  - 接收缓存溢出中断及对应标志位，及时通知异常
- 支持 DMA
  - 使能 TXDMAEN，发送缓存器空标志位 TXEIF 置起后可触发 DMA 请求，DMA 写入发送缓存后，自动清除 TXEIF 标志位
  - 使能 RXDMAEN，接收缓冲区非空标志位 RXNEIF 置起后可触发 DMA 请求，DMA 读取接收缓存后，自动清除 RXEIF 标志位

### 18.3 SPI1 特性

- 与 TWI1 共用寄存器地址和信号口，但功能完全独立
- 支持 13 档 SPI 时钟预分频，时钟预分频允许用户设置到较低频率，最小分频档位为  $f_{PCLK}/4096$ 。
- 信号口共四组 IO 映射可选
- 无 FIFO 缓存
- 支持 DMA：一帧结束统一置位请求。

## 18.4 信号描述

### 主输出从输入 (MOSI) :

该路信号连接主设备和一个从设备。数据通过 MOSI 从主设备串行传送到从设备，主设备输出，从设备输入。

### 主输入从输出 (MISO) :

该路信号连接从设备和主设备。数据通过 MISO 从从设备串行传送到主设备，从设备输出，主设备输入。当 SPI 配置为从设备并未被选中，从设备的 MISO 引脚处于高阻状态。

### SPI 串行时钟 (SCK) :

SCK 信号用作控制 MOSI 和 MISO 线上输入输出数据的同步移动。每 8 时钟周期线上传送一个字节。如果从设备未被选中，SCK 信号被此从设备忽略。

## 18.5 工作模式

SPI 可配置为主模式或从模式中的一种。SPI 模块的配置和初始化通过设置 SPI 控制寄存器 SPI0\_CON/SPI1\_TWI1\_CON 和 SPI 中断使能及 DMA 控制寄存器 SPI0\_IDE/SPI1\_TWI1\_IDE 来完成。配置完成后，通过设置 SPI 数据寄存器 SPI0\_DATA/SPI1\_TWI1\_DATA (以下简称 SPD) 来完成数据传送。

在 SPI 通讯期间，数据同步地被串行的移进移出。串行时钟线 (SCK) 使两条串行数据线 (MOSI 和 MISO) 上数据的移动和采样保持同步。如果从设备没有被选中，则不能参与 SPI 总线上的活动。

当 SPI 主设备通过 MOSI 线传送数据到从设备时，从设备通过 MISO 线发送数据到主设备作为响应，这就实现了在同一时钟下数据发送和接收的同步全双工传输。发送移位寄存器和接收移位寄存器使用相同的特殊功能器地址，对 SPI 数据寄存器 SPD 进行写操作将写入发送移位寄存器，对 SPD 进行读操作将获得接收移位寄存器的数据。

有些设备的 SPI 接口会引出 SS 脚 (从设备选择引脚，低有效)，与 SC32F15G 的 SPI 通信时，SPI 总线上其它设备的 SS 脚的连接方式需根据不同的通信模式进行连接。下表列出了 SC32F15G 的 SPI 不同通信模式下，SPI 总线上其它设备 SS 脚的连接方式：

SC32F15G SPI	SPI 总线上其它设备	模式	从机的 SS (从设备选择引脚)
主模式	从模式	一主一从	拉低
		一主多从	SC32F15G 引出多根 I/O，分别接至从机的 SS 脚。在数据传送之前，从设备的 SS 引脚必须被置低
从模式	主模式	一主一从	拉高

### 18.5.1 主模式

- 模式启动：

SPI 主设备控制 SPI 总线上所有数据传送的启动。当 SPI 控制寄存器 SPI0\_CON/SPI1\_TWI1\_CON 中的 MSTR/SMSTR 位置 1 时，SPI 在主模式下运行，只有一个主设备可以启动传送。

- 发送:

在 SPI 主模式下, 对 SPD 进行以下操作: 在 8 位模式下写一个字节数据到 SPD[7:0]或在 16 位模式下将数据写到 SPD[15:0], 再将低字节写入 SPDL, 数据将会写入发送移位缓冲器。如果发送移位寄存器已经存在一个数据, 那么主 SPI 产生一个 WCOL 信号以表明写入太快。但是在发送移位寄存器中的数据不会受到影响, 发送也不会中断。另外如果发送移位寄存器不为空, 那么主设备立即按照 SCK 上的 SPI 时钟频率串行地移出发送移位寄存器中的数据到 MOSI 线上。当传送完毕, SPI 标志状态位寄存器 SPI0\_STS/SPI1\_TWI1\_STS 中的 SPIF/QTWIF 位被置 1。如果 SPI 中断被允许, 当 SPIF/QTWIF 位置 1 时, 也会产生一个中断。

- 接收:

当主设备通过 MOSI 线传送数据给从设备时, 相对应的从设备同时也通过 MISO 线将其发送移位寄存器的内容传送给主设备的接收移位寄存器, 实现全双工操作。因此, SPIF/QTWIF 标志位置 1 即表示传送完成也表示接收数据完毕。从设备接收的数据按照 MSB 或 LSB 优先的传送方向存入主设备的接收移位寄存器。当一个字节的数据完全被移入接收寄存器时, 处理器可以通过读 SPD 获得该数据。

## 18.5.2 从模式

- 模式启动:

当 SPI 控制寄存器 SPI0\_CON/SPI1\_TWI1\_CON 中的 MSTR/SMSTR 位清 0, SPI 在从模式下运行。

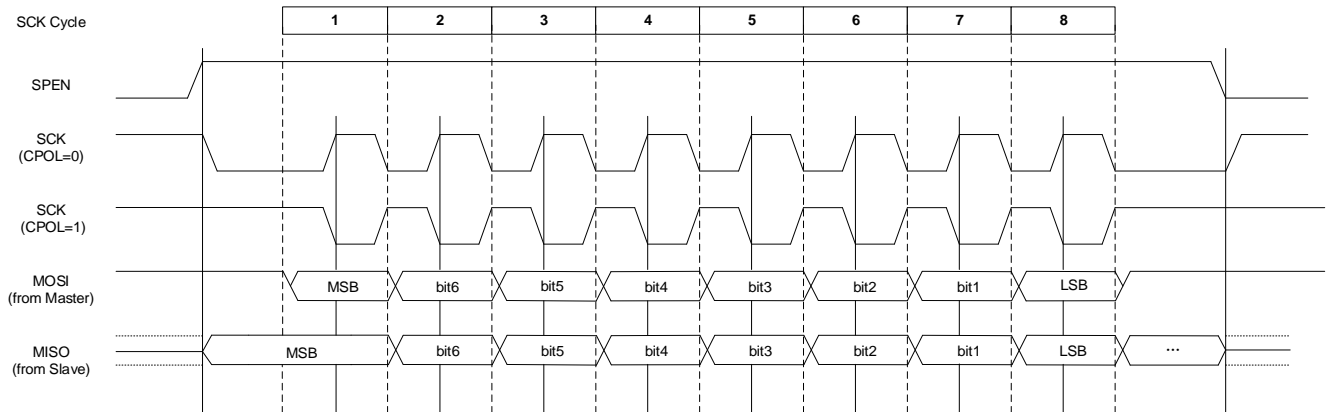
- 发送与接收:

从属模式下, 按照主设备控制的 SCK 信号, 数据通过 MOSI 引脚移入, MISO 引脚移出。一个位计数器记录 SCK 的边沿数, 当接收移位寄存器移入 8 位数据 (一个字节) 同时发送移位寄存器移出 8 位数据 (一个字节), SPIF/QTWIF 标志位被置 1。数据可以通过读取 SPD 寄存器获得。如果 SPI 中断被允许, 当 SPIF/QTWIF 置 1 时, 也会产生一个中断。此时接收移位寄存器保持原有数据并且 SPIF/QTWIF 位置 1, 这样 SPI 从设备将不会接收任何数据直到 SPIF/QTWIF 清 0。SPI 从设备必须在主设备开始一次新的数据传送之前将要传送的数据写入发送移位寄存器。如果在开始发送之前未写入数据, 从设备将传送 “0x00” 字节给主设备。如果写 SPD 操作发生在传送过程中, 那么 SPI 从设备的 WCOL 标志位置 1, 即如果传送移位寄存器已经含有数据, SPI 从设备的 WCOL 位置 1, 表示写 SPD 冲突。但是移位寄存器的数据不受影响, 传送也不会被中断。

## 18.6 传送形式

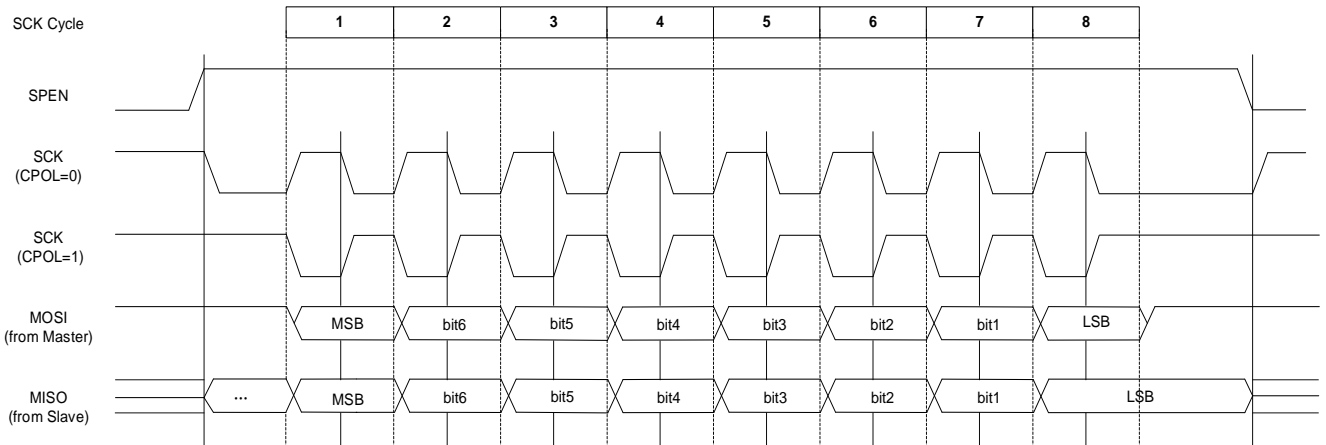
通过软件设置 SPI 控制寄存器 SPI0\_CON/SPI1\_TWI1\_CON 的 CPOL 位和 CPHA 位, 用户可以选择 SPI 时钟极性和相位的四种组合方式。CPOL 位定义时钟的极性, 即空闲时的电平状态, 它对 SPI 传输格式影响不大。CPHA 位定义时钟的相位, 即定义允许数据采样移位的时钟边沿。在主从通讯的两个设备中, 时钟极性相位的设置应一致。

当 CPHA = 0, SCK 的第一个沿捕获数据, 从设备必须在 SCK 的第一个沿之前将数据准备好。



CPHA = 0 数据传输图

当 CPHA = 1，主设备在 SCK 的第一个沿将数据输出到 MOSI 线上，从设备把 SCK 的第一个沿作为开始发送信号，SCK 的第二沿开始捕获数据，因此用户必须在第一个 SCK 的两个沿内完成写 SPD 的操作。这种数据传输形式是一个主设备一个从设备之间通信的首选形式。



CPHA = 1 数据传输图

## 18.7 出错检测

在发送数据序列期间写入 SPD 会引起写冲突，SPI 标志状态位寄存器 SPI0\_STS/SPI1\_TWI1\_STS 中的 WCOL 位置 1。WCOL 位置 1 不会引起中断，发送也不会中止。WCOL 位需由软件清 0。

## 18.8 SPI0 和 SPI1 对比

对比 BIT 位	SPI0	SPI1
信号口强驱动	有	无
WCOL	当发送 FIFO 写满后，对 FIFO 进行写操作将无法写入，WCOL 也会置起，代表缓存写入冲突	当一帧正在发送，对发送缓存进行写操作将无法写入，WCOL 也会置起，代表缓存写入冲突
SPIF	该位置起，代表一帧数据接收/发送完成	无
QTWIF	无	该位置起，代表一帧数据接收/发送完成
RXHIE	接收 FIFO 内有效数据超过一半中断使能位	无
TXHIE	发送 FIFO 内有效数据不满一半中断使能位	无
RXIE	接收 FIFO 已满中断使能位	无
TBIE	发送 FIFO 为空中断使能位	发送缓存为空时中断使能位
RXNEIE	接收 FIFO 非空中断使能位	无
RXHIF	该位置起，代表接收 FIFO 内有效数据超过一半	无
TXHIF	该位置起，代表发送 FIFO 内有效数据不满一半	无
RXFIF	该位置起，代表接收 FIFO 已满	无
TXEIF	该位置起，代表发送 FIFO 为空	该位置起，代表发送缓存为空
RXNEIF	接收 FIFO 非空标志位	无
DMA	通过发送缓存器空标志位 TXEIF 和接收缓冲区非空状态位 RXNEIF 触发 DMA 请求	一帧结束统一置位请求

## 18.9 SPI 中断

对于 SPI0，在发生“传输完成”、“FIFO 半传输”或“发送缓存为空”时都会生成中断。可以使用单独的中断使能位以提高灵活性。

中断事件	事件标志位	中断请求控制位	子事件标志位	中断使能子开关
接收 FIFO 内有效数据超过一半	SPIF	SPI0_IDE ->INTEN	RXHIF	RXHIE
发送 FIFO 内有效数据不满一半			TXHIF	TXHIE
接收 FIFO 已满			RXFIF	RXIE
发送 FIFO 为空			TXEIF	TBIE
接收 FIFO 非空			RXNEIF	RXNEIE

对于 SPI1，在发生“传输完成”或“发送缓存”时都会生成中断。可以使用单独的中断使能位以提高灵活性。

中断事件	事件标志位	中断请求控制位	子事件标志位	中断使能子开关
一帧数据接收/发送完成	QTWIF	SPI1_TWI1_IDE	\	\

中断事件	事件标志位	中断请求控制位	子事件标志位	中断使能子开关
发送缓存为空		->INTEN	TXEIF	TBIE

## 18.10 SPI0 寄存器

### 18.10.1 SPI0 相关寄存器表

#### 18.10.1.1 SPI0 控制寄存器 SPI0\_CON

寄存器	读/写	说明	复位值	上电初始值
SPI0_CON	读/写	SPI0 控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	SPOS	-	SPR[3:0]			
7	6	5	4	3	2	1	0
SPEN	-	-	CPOL	CPHA	DORD	SPMD	MSTR

位编号	位符号	说明												
13	SPOS	SPI0 信号口映射控制位												
		<table border="1"> <thead> <tr> <th>信号</th> <th>MISO0</th> <th>MOSI0</th> <th>SCK0</th> </tr> </thead> <tbody> <tr> <td>SPOS=0</td> <td>PC1</td> <td>PC2</td> <td>PC3</td> </tr> <tr> <td>SPOS=1</td> <td>PB5</td> <td>PB6</td> <td>PB7</td> </tr> </tbody> </table>	信号	MISO0	MOSI0	SCK0	SPOS=0	PC1	PC2	PC3	SPOS=1	PB5	PB6	PB7
		信号	MISO0	MOSI0	SCK0									
SPOS=0	PC1	PC2	PC3											
SPOS=1	PB5	PB6	PB7											
11~8	SPR[3:0]	SPI 时钟预分频扩展一位 0000: f <sub>PCLK0</sub> 0001: f <sub>PCLK0</sub> /2 0010: f <sub>PCLK0</sub> /4 0011: f <sub>PCLK0</sub> /8 0100: f <sub>PCLK0</sub> /16 0101: f <sub>PCLK0</sub> /32 0110: f <sub>PCLK0</sub> /64 0111: f <sub>PCLK0</sub> /128 1000: f <sub>PCLK0</sub> /256 1001: f <sub>PCLK0</sub> /512 1010: f <sub>PCLK0</sub> /1024 其它: f <sub>PCLK0</sub> /1024 注意: 为确保 SC32F15G 系列 SPI0 正确通信, 通信频率请选择 16MHz 以下												
7	SPEN	SPI 使能控制位 0: 关闭 SPI0												

位编号	位符号	说明
		1: 打开 SPI0
4	CPOL	SPI 时钟极性控制位 0: SCK 在空闲状态下为低电平 1: SCK 在空闲状态下为高电平
3	CPHA	SPI 时钟相位控制位 0: SCK 周期的第一沿采集数据 1: SCK 周期的第二沿采集数据
2	DORD	SPI 传送方向选择位 0: MSB 优先发送 1: LSB 优先发送
1	SPMD	SPI 传输模式选择位 0: 8 位模式 1: 16 位模式
0	MSTR	SPI 主从机模式选择位 0: SPI0 为从设备 1: SPI0 为主设备
31~14 12 6~5	-	保留

#### 18.10.1.2 SPI0 标志状态位寄存器 SPI0\_STS

寄存器	读/写	说明	复位值	上电初始值
SPI0_STS	读/写	SPI0 标志状态位寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
WCOL	-	TXHIF	RXHIF	RXFIF	TXEIF	RXNEIF	SPIF

位编号	位符号	说明
7	WCOL	写入冲突标志位 该位由硬件置 1，通过软件写 1 清 0，用于指示是否发生写入冲突： 0: 未检测到写入冲突 1: 检测到一个写入冲突
5	TXHIF	发送 FIFO 内有效数据不满一半状态位/中断状态位 该位只读，由硬件置起或清零，用于指示当前发送 FIFO 的状态： 0: 发送 FIFO 中的有效数据个数大于 FIFO 一半 1: 发送 FIFO 中的有效数据个数少于或等于 FIFO 一半，如果此时 TXHIE = 1，将产生中断

位编号	位符号	说明
4	RXHIF	接收 FIFO 内有效数据超过一半状态位/中断状态位 该位只读，由硬件置起或清零，用于指示当前接收 FIFO 的状态： 0: 接收 FIFO 的有效数据个数少于或等于 FIFO 一半 1: 接收 FIFO 的有效数据个数大于 FIFO 一半，如果此时 RXHIE = 1，将产生中断
3	RXFIF	接收 FIFO 已满状态位 该位只读，由硬件置起或清零，用于指示当前接收 FIFO 是否已满： 0: 接收 FIFO 未滿 1: 接收 FIFO 已滿
2	TXEIF	发送 FIFO 为空标志位 用于指示当前发送 FIFO 是否为空： 0: 发送 FIFO 非空 1: 发送 FIFO 为空 该位由硬件置 1，通过软件写 1 清 0。 注意：在 DMA 模式下，DMA 写入发送缓存后，该位由 DMA 模块清 0，此时用户无需通过软件清 0。
1	RXNEIF	接收 FIFO 非空状态位 用于指示当前接收 FIFO 是否为空： 该位只读，由硬件置起或清零，用于指示当前接收 FIFO 的状态： 0: 接收 FIFO 为空 1: 接收 FIFO 非空
0	SPIF	SPI 数据传送标志位 该位由硬件置 1，通过软件写 1 清 0，用于指示当前 SPI 数据传输是否完成： 0: 数据传输未完成/进行中 1: 数据传输已完成
31~8 6	-	保留

### 18.10.1.3 SPI0 数据寄存器 SPI0\_DATA

寄存器	读/写	说明	复位值	上电初始值
SPI0_DATA	读/写	SPI0 数据寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
15~0	SPD[15:0]	SPI 数据缓存

位编号	位符号	说明
		读操作：从 SPI0 接收 FIFO 读取接收到的数据 写操作：向 SPI0 发送 FIFO 写入待发送的数据
31~16	-	保留

#### 18.10.1.4 SPI0 的中断使能及 DMA 控制寄存器 SPI0\_IDE

寄存器	读/写	说明	复位值	上电初始值
SPI0_IDE	读/写	SPI0 的中断使能及 DMA 控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
TXDMAEN	RXDMAEN	TXHIE	RXHIE	RXIE	TBIE	RXNEIE	INTEN

位编号	位符号	说明
7	TXDMAEN	DMA 发送通道使能位 0: 禁用 DMA 发送功能 1: 使能 DMA 发送功能 该位使能后, TXEIF 置起可触发 DMA 通道发送请求
6	RXDMAEN	DMA 接收通道使能位 0: 禁用 DMA 接收功能 1: 使能 DMA 接收功能 该位使能后, RXNEIF 置起可触发 DMA 通道接收请求
5	TXHIE	发送 FIFO 内有效数据不满一半中断使能位 0: TXHIF 置起时, 不允许产生中断 1: TXHIF 置起时, 允许产生中断
4	RXHIE	接收 FIFO 内有效数据超过一半中断使能位 0: RXHIF 置起时, 不允许产生中断 1: RXHIF 置起时, 允许产生中断
3	RXIE	接收 FIFO 已满中断使能位 0: RXFIF 置起时, 不允许产生中断 1: RXFIF 置起时, 产生中断
2	TBIE	发送 FIFO 为空时的中断使能位 0: TXEIF 置起时, 不允许产生中断 1: TXEIF 置起时, 允许产生中断
1	RXNEIE	接收 FIFO 非空中断使能位 0: RXNEIF 置起, 不允许产生中断 1: RXNEIF 置起, 产生中断
0	INTEN	中断请求 CPU 的使能控制位 0: 禁止中断请求

位编号	位符号	说明
		1: 使能中断请求
31~8	-	保留

## 18.10.2 SPI0 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值	访问限制
SPI0 基地址: 0x4002_0040						
SPI0_CON	0x00	读/写	SPI 控制寄存器	0x0000_0000	0x0000_0000	-
SPI0_STS	0x04	读/写	SPI 标志位寄存器	0x0000_0000	0x0000_0000	-
SPI0_DATA	0x0C	读/写	SPI 数据寄存器	0x0000_0000	0x0000_0000	不支持字节/半字访问
SPI0_IDE	0x10	读/写	通信口 DMA 控制寄存器	0x0000_0000	0x0000_0000	-

## 18.11 SPI1 寄存器

### 18.11.1 SPI1 相关寄存器表

#### 18.11.1.1 SPI1\_TWI1 二合一控制寄存器 SPI1\_TWI1\_CON

寄存器	读/写	说明	复位值	上电初始值
SPI1_TWI1_CON	读/写	SPI1_TWI1 二合一控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
MODE[1:0]		-	CPOL	CPHA	DORD	SPMD	SMSTR
15	14	13	12	11	10	9	8
-	SPOS[1:0]		-	QWCK[3:0]			
7	6	5	4	3	2	1	0
QWEN	-	START	STOP	-	-	AA	STRETCH

位编号	位符号	说明
23~22	MODE[1:0]	SPI1_TWI1 模式选择位 00: TWI 模式 01: SPI 模式 10: 保留 11: 保留
20	CPOL	SPI 时钟极性控制位 0: SCK 在空闲状态下为低电平 1: SCK 在空闲状态下为高电平
19	CPHA	SPI 时钟相位控制位 0: SCK 周期的第一沿采集数据 1: SCK 周期的第二沿采集数据

位编号	位符号	说明																				
18	DORD	SPI 传送方向选择位 0: MSB 优先发送 1: LSB 优先发送																				
17	SPMD	SPI 传输模式选择位 0: 8 位模式 1: 16 位模式																				
16	SMSTR	SPI 主从机模式选择位 0: SPI1 为从设备 1: SPI1 为主设备																				
14~13	SPOS[1:0]	<p>● SPI1 信号口映射控制位@SPI1_TW11_CON</p> <table border="1"> <thead> <tr> <th>信号</th> <th>MISO1</th> <th>MOSI1</th> <th>SCK1</th> </tr> </thead> <tbody> <tr> <td>SPOS[1:0]=00</td> <td>PC0</td> <td>PB15</td> <td>PB14</td> </tr> <tr> <td>SPOS[1:0]=01</td> <td>PA8</td> <td>PA7</td> <td>PA6</td> </tr> <tr> <td>SPOS[1:0]=10</td> <td>PA13</td> <td>PA12</td> <td>PA11</td> </tr> <tr> <td>SPOS[1:0]=11</td> <td>PB4</td> <td>PB3</td> <td>PB2</td> </tr> </tbody> </table>	信号	MISO1	MOSI1	SCK1	SPOS[1:0]=00	PC0	PB15	PB14	SPOS[1:0]=01	PA8	PA7	PA6	SPOS[1:0]=10	PA13	PA12	PA11	SPOS[1:0]=11	PB4	PB3	PB2
信号	MISO1	MOSI1	SCK1																			
SPOS[1:0]=00	PC0	PB15	PB14																			
SPOS[1:0]=01	PA8	PA7	PA6																			
SPOS[1:0]=10	PA13	PA12	PA11																			
SPOS[1:0]=11	PB4	PB3	PB2																			
11~8	QTWCK[3:0]	<p>主机模式下 SPI 的通信速率设定:</p> <p>0000: f<sub>PCLK1</sub>            0001: f<sub>PCLK1</sub> /2            0010: f<sub>PCLK1</sub> /4            0011: f<sub>PCLK1</sub> /8            0100: f<sub>PCLK1</sub> /16            0101: f<sub>PCLK1</sub> /32            0110: f<sub>PCLK1</sub> /64            0111: f<sub>PCLK1</sub> /128            1000: f<sub>PCLK1</sub> /256            1001: f<sub>PCLK1</sub> /512            1010: f<sub>PCLK1</sub> /1024            1011: f<sub>PCLK1</sub> /2048            1100: f<sub>PCLK1</sub> /4096            其它: f<sub>PCLK1</sub> /4096</p> <p>注意: 为确保 SC32F15G 系列 SPI1 正确通信, 通信频率请选择 8MHz 以下</p>																				
7	QTWEN	SPI1_TW11 模块使能控制位 0: 关闭模块 1: 打开模块																				
31~24 21 15 12 6 3~2	-	保留																				

### 18.11.1.2 SPI1\_TW11 二合一标志状态位寄存器 SPI1\_TW11\_STS

寄存器	读/写	说明	复位值	上电初始值
SPI1_TW11_STS	读/写	SPI1_TW11 二合一标志状态位寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-

23	22	21	20	19	18	17	16
NBYTES[7:0]							
15	14	13	12	11	10	9	8
-	-	-	-	-	STATE[2:0]		
7	6	5	4	3	2	1	0
WCOL	TXEIF	-	-	TMSTR	GCA	TXnE/RXnE	QTWIF

位编号	位符号	说明
7	WCOL	写入冲突标志位 该位由硬件置 1，通过软件写 1 清 0，用于指示是否发生写入冲突： 0：未检测到写入冲突 1：检测到一个写入冲突
6	TXEIF	发送缓存器为空标志位 用于指示当前发送缓存器是否为空： 0：发送缓存器非空 1：发送缓存器为空 该位由硬件置 1，通过软件写 1 清 0。
0	QTWIF	SPI 数据传送标志位 该位由硬件置 1，通过软件写 1 清 0，用于指示当前 SPI 数据传输是否完成： 0：数据传输未完成/进行中 1：数据传输已完成
31~24 15~11 5~4	-	保留

### 18.11.1.3 SPI1\_TWI1 二合一数据寄存器 SPI1\_TWI1\_DATA

寄存器	读/写	说明	复位值	上电初始值
SPI1_TWI1_DATA	读/写	SPI1_TWI1 二合一数据寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
15~0	QTWIDAT[15:0]	SPI 数据缓存 读操作：从 SPI1 接收缓存器读取接收到的数据 写操作：向 SPI1 发送缓存器写入待发送的数据
31~16	-	保留

### 18.11.1.4 SPI1\_TWI1 二合一中断使能及 DMA 控制寄存器 SPI1\_TWI1\_IDE

寄存器	读/写	说明	复位值	上电初始值
SPI1_TWI1_IDE	读/写	SPI1_TWI1 二合一中断使能及 DMA 控制寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
7	TXDMAEN	DMA 发送通道使能位 0: 禁用 DMA 发送功能 1: 使能 DMA 发送功能 该位使能后, TXEIF 置起可触发 DMA 通道发送请求
6	RXDMAEN	DMA 接收通道使能位 0: 禁用 DMA 接收功能 1: 使能 DMA 接收功能 该位使能后, QTWIF 置起可触发 DMA 通道接收请求
1	TBIE	发送缓存为空时的中断使能位 0: TXEIF 置起时, 不允许产生中断 1: TXEIF 置起时, 产生中断
0	INTEN	中断请求 CPU 的使能控制位 0: 禁止中断请求 1: 使能中断请求
31~8 5~2	-	保留

### 18.11.2 SPI1 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值	访问限制
SPI1_TWI1 基地址: 0x4002_1040						
SPI1_TWI1_CON	0x00	读/写	SPI1_TWI1 二合一控制寄存器	0x0000_0000	0x0000_0000	-
SPI1_TWI1_STS	0x04	读/写	SPI1_TWI1 二合一标志位寄存器	0x0000_0000	0x0000_0000	-
SPI1_TWI1_DATA	0x0C	读/写	SPI1_TWI1 二合一数据寄存器	0x0000_0000	0x0000_0000	不支持字节/半字访问
SPI1_TWI1_IDE	0x10	读/写	SPI1_TWI1 中断使能及 DMA 控制寄存器	0x0000_0000	0x0000_0000	-

## 19 TWI0~1

### 19.1 时钟源

- SC32F15G 系列的 TWI 的时钟源仅一种，来自 PCLK

### 19.2 TWI0 特性

- 支持 11 档 TWI 时钟预分频，主机模式下 TWI 通信速率，默认为最小分频档位（ $f_{PCLK}/4$ ）
- 信号口共三组映射可选
- 可配置为主机模式或从机模式
- 主从机之间双向数据传输
- 速率可达 1Mbps
- 支持 DMA

### 19.3 TWI1 特性

- 与 SPI1 共用寄存器地址和信号口，但功能完全独立
- 支持 11 档 TWI 时钟预分频，主机模式下 TWI 通信速率，默认为最小分频档位（ $f_{PCLK}/4$ ）
- 信号口共四组映射可选
- 可配置为主机模式或从机模式
- 主从机之间双向数据传输
- 速率可达 1Mbps

### 19.4 TWI 信号描述

在 TWI 总线上，数据通过时钟线 SCL 和数据线 SDA 在主从机间逐一字节同步传送。每个字节数据长度是 8 位，一个 SCL 时钟脉冲传输一个数据位，数据由最高位 MSB 开始传输，每个字节传输后跟随一个应答位，每个位在 SCL 为高时采样。

因此，SDA 线可能在 SCL 为低时改变，但在 SCL 为高时必须保持稳定。当 SCL 为高时，SDA 线上的跳变视为一个命令（START 或 STOP）。

- TWI 时钟信号线（SCL）

该时钟信号由主机发出，连接到所有的从机。每 9 个时钟周期传送一个字节数据。前 8 个周期作数据的传送，最后一个时钟作为接收方应答时钟。空闲时应为高电平，由 SCL 线上的上拉电阻拉高。

- TWI 数据信号线（SDA）

SDA 是双向信号线，空闲时应为高电平，由 SDA 线上的上拉电阻拉高。

## 19.5 从机工作模式

- 模式启动:

当 TWI 使能标志位打开（TWI0 为 TWEN = 1，TWI1 为 QTWEN=1），同时接收到主机发送的启动信号时，模式启动。

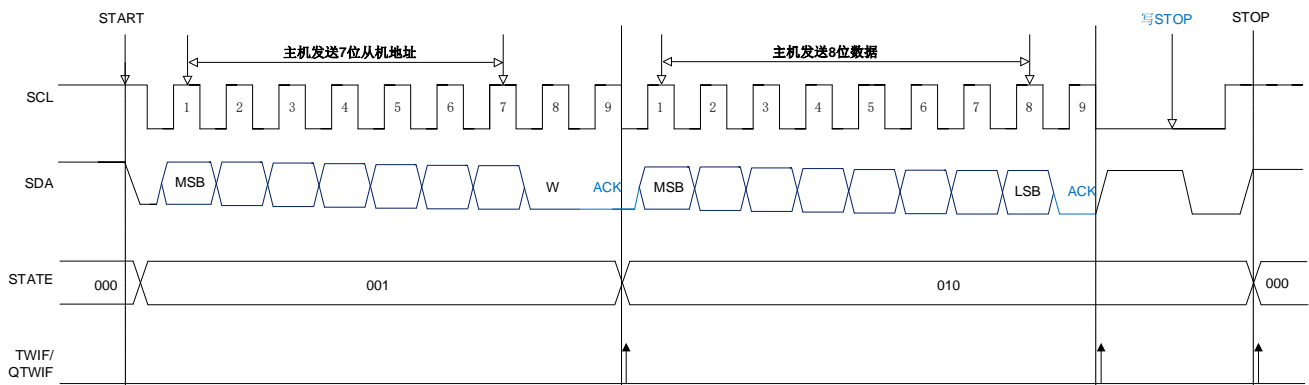
从机从空闲模式（STATE[2:0] = 000）进入接收第一帧地址（STATE[2:0] = 001）状态，等待主机的第一帧数据。第一帧数据由主机发送，包括了 7 位地址位和 1 位读写位，TWI 总线上所有从机都会收到主机的第一帧数据。主机发送完第一帧数据后释放 SDA 信号线。若主机所发地址与某一从机自身地址寄存器中的值相同，说明该从机被选中，被选中的从机会判断接总线上的第 8 位，即数据读写位（=1，读命令；=0，写命令），然后占用 SDA 信号线，在 SCL 的第 9 个时钟周期给主机一个低电平的应答信号，之后会释放总线。从机被选中后，会根据读写位的不同而进入不同的状态：

- 非通用地址响应，从机接收模式:

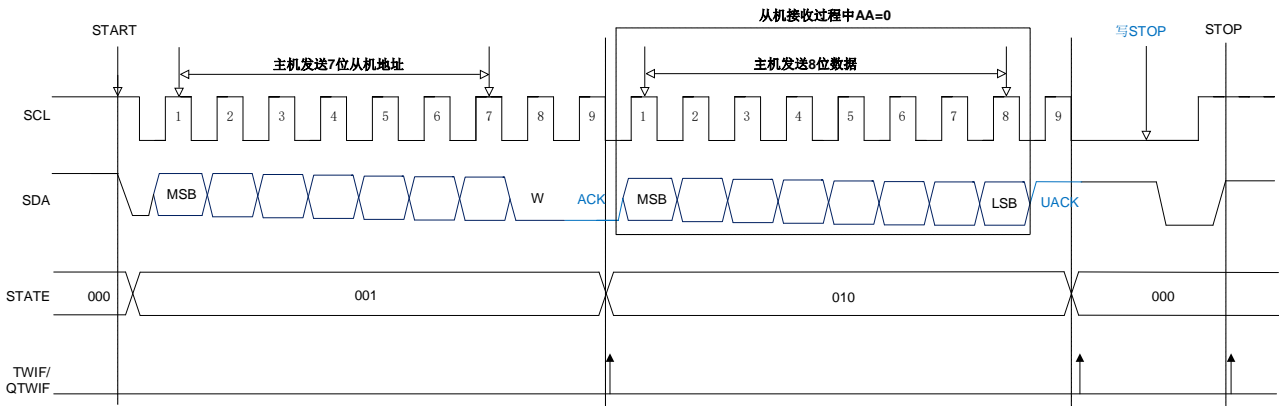
如果第一帧接收到的读写位是写（0），则从机进入到从机接收状态（STATE[2:0] = 010）等待接收主机发送的数据。主机每发送 8 位，都要释放总线，等待第 9 个周期从机的应答信号。

如果从机的应答信号是低电平，主机的通信可以有以下几种方式：

- 继续发送数据；
- 重新发送启动信号（start），此时从机重新进入接收第一帧地址（STATE[2:0] = 001）状态；
- 发送停止信号，表示本次传输结束，从机回到空闲状态，等待主机下一次的启动信号。



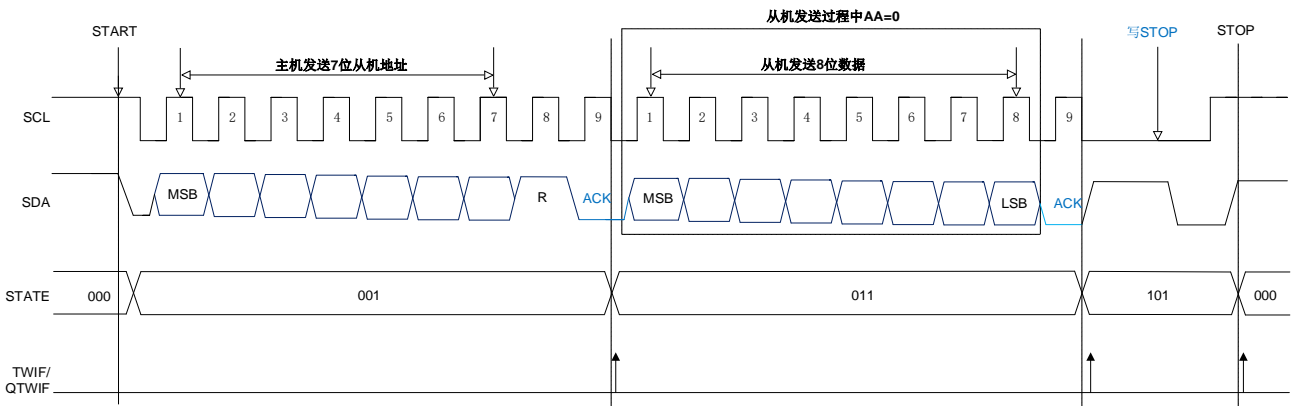
如果从机应答的是高电平（在接收过程中，从机寄存器中的 AA 值改写为 0），表示当前字节传输完以后，从机会主动结束本次传输，回到空闲状态（STATE[2:0] = 000），不再接收主机发送的数据。



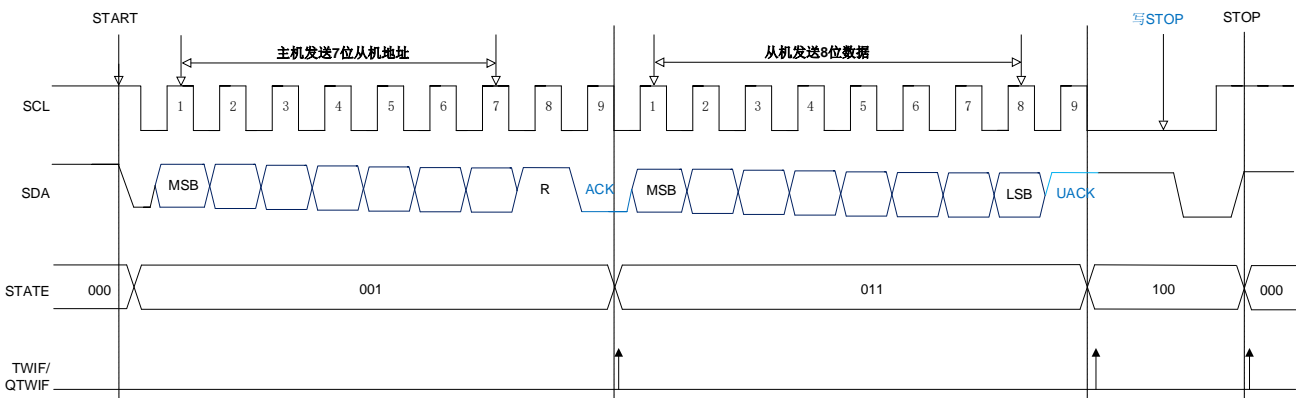
● 非通用地址响应，从机发送模式：

如果第一帧接收到的读写位是读（1），则从机会占用总线，向主机发送数据。每发送 8 位数据，从机释放总线，等待主机的应答：

如果主机应答的是低电平，则从机继续发送数据。在发送过程中，如果从机寄存器中的 AA 值被改写为 0，则传输完当前字节从机会主动结束传输并释放总线，等待主机的停止信号或重新启动信号（STATE[2:0] = 101）。



如果主机应答的是高电平，则从机 STATE[2:0] = 100，等待主机的停止信号或重新启动信号。

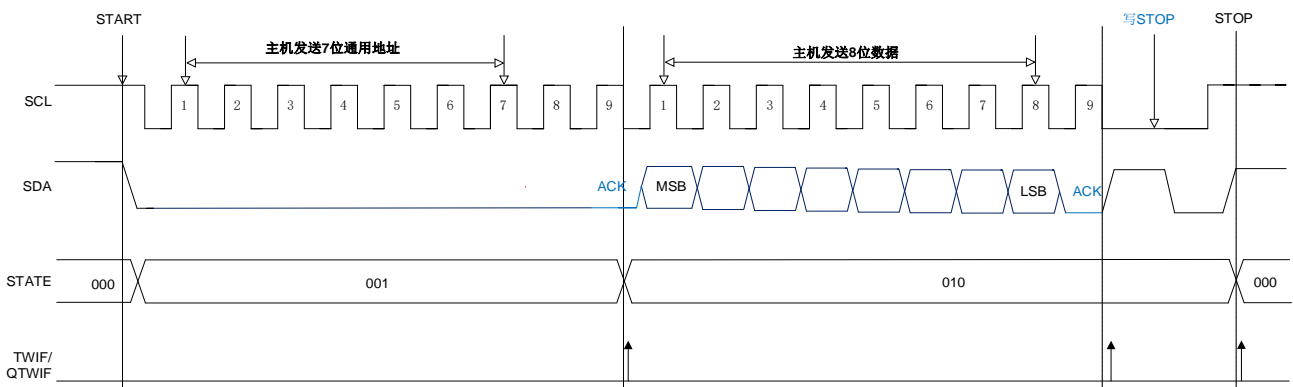


- 通用地址的响应:

GC=1 时, 此时通用地址允许使用。从机进入到接收第一帧地址 (STATE[2:0] = 001) 状态, 接收的第一帧数据中的地址位数据为 0x00, 此时所有从机响应主机。主机发送的读写位必须是写 (0), 所有从机接收后进入接收数据 (STATE[2:0] = 010) 状态。主机每发送 8 个数据释放一次 SDA 线, 并读取 SDA 线上的状态:

如果有从机应答, 则主机的通信可以有以下三种方式:

- 继续发送数据;
- 重新启动;
- 发送停止信号, 结束本次通讯。



如果无从机应答, 则 SDA 为空闲状态。

注意: 在一主多从模式下使用通用地址时, 主机发送的读写位不能为读 (1) 状态, 否则除发送数据的设备, 总线上其它设备均会响应。

## 19.6 从机模式操作步骤

- ① 配置 TWI0 控制寄存器 TWI0\_CON: TWEN = 1, 或 TWI1 控制寄存器 SPI1\_TWI1\_CON: QTWEN=1, 使能 TWI
- ② 配置 TWI0 控制寄存器 TWI0\_CON, 或 TWI1 控制寄存器 SPI1\_TWI1\_CON;
- ③ 配置 TWI0 地址寄存器 TWI0\_ADD, 或 TWI1 地址寄存器 TWI1\_ADD;
- ④ 如果从机接收数据, 则等待 TWI0\_STS 中的中断标志位 TWIF 置 1, 或等待 SPI1\_TWI1\_STS 中的中断标志位 QTWIF 置 1。从机每接收到 8 位数据, TWIF/QTWIF 会被置 1。中断标志位 TWIF/QTWIF 需手动清零;
- ⑤ 如果从机发送数据, 则要将待发送的数据写进 TWI0 的 TWIDAT 或 SPI1\_TWI1 的 QTWIDAT 中, TWI 会自动将数据发送出去。每发送 8 位, 中断标志位 TWIF/QTWIF 就会被置 1。

## 19.7 主机工作模式

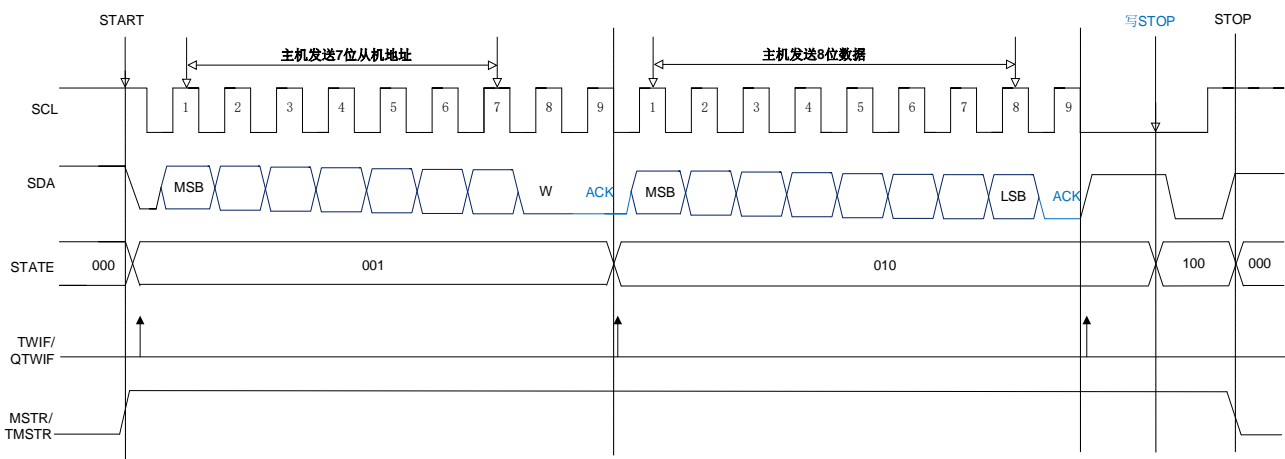
- 模式启动

当 TWI 接口向总线发出起始条件后，会自动切换为主模式，同时硬件将 TWIO 的 MSTR 位置 1 或 SPI1\_TW11 的 TMSTR 位置 1。主机状态位 STATE[2:0] 从 000 切换到 001，同时中断条件 TWIF/QTWIF 被置 1。

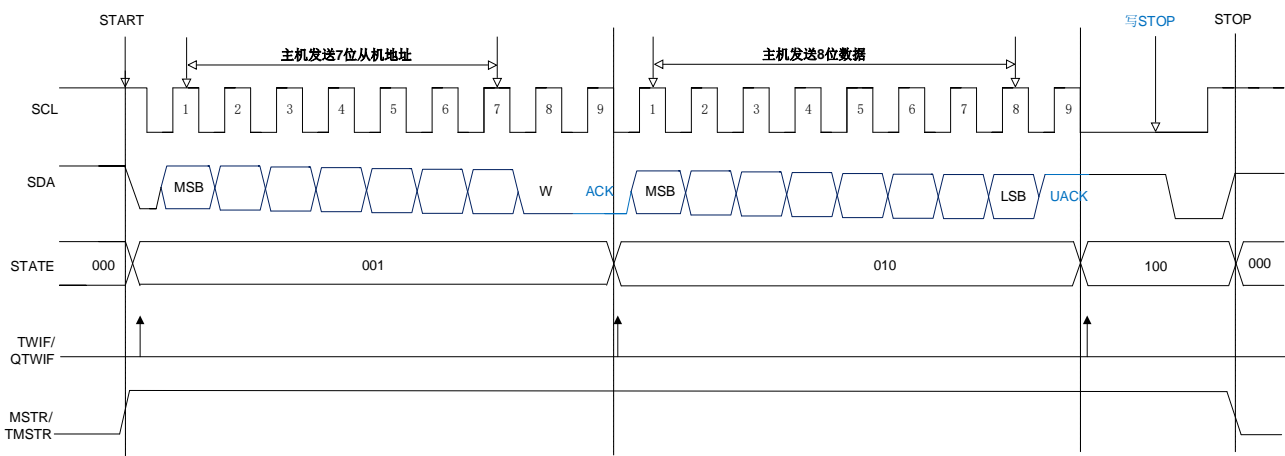
- TWI 主机发送模式

主机发送模式下，主机发送的第一帧数据包括了 7 位地址位（被选中的从机地址）和 1 位读写位（=0，写命令），TWI 总线上所有从机都会收到主机的第一帧数据。主机发送完第一帧数据后释放 SDA 信号线。被选中的从机在 SCL 的第 9 个时钟周期给主机一个应答信号，之后会释放总线并进入到从机接收状态等待接收主机发送的数据。主机每发送 8 位，都要释放总线，等待第 9 个周期从机的应答信号。

如果从机应答低电平，主机可以继续发送数据。也可以重新发送启动信号：



如果从机应答高电平，表示当前字节传输完以后，从机会主动结束本次传输，不再接收主机发送的数据，主机 STATE[2:0] 从发送数据状态 010 切换为 100：

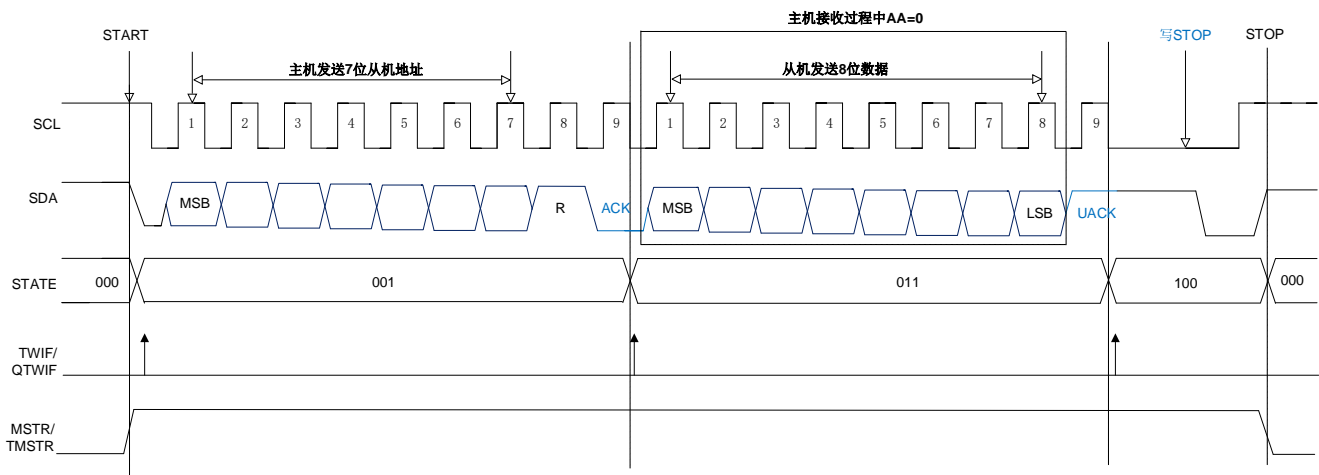


● TWI 主机接收模式

主机发送模式下，主机发送的第一帧数据包括了包括了 7 位地址位（被选中的从机地址）和 1 位读写位（=1，读命令），TWI 总线上所有从机都会收到主机的第一帧数据。主机发送完第一帧数据后释放 SDA 信号线。被选中的从机在 SCL 的第 9 个时钟周期给主机一个应答信号，之后会占用总线，向主机发送数据。每发送 8 位数据，从机释放总线，等待主机的应答。主机接收到从机地址匹配成功后的应答信号 ACK，并开始接收从机数据（STATE=011）：

1. 若主机应答位使能（AA=1），则每接收到一 byte 数据，主机回复应答信号 ACK，TWIF 被置位；
2. 在接收最后一 byte 数据前，若应答使能位关闭（AA=0），则主机接收完最后一 byte 数据后回复 UACK，然后主机可发送停止信号。

主机接收模式下，主动释放总线方式如下：



## 19.8 主机模式操作步骤

- ① 配置 TWI0 控制寄存器 TWI0\_CON：TWEN = 1，或 TWI1 控制寄存器 SPI1\_TWI1\_CON:QTWEN = 1，使能 TWI
- ② 配置 TWI0 控制寄存器 TWI0\_CON：配置 TWI0 通信速率（TWCK[3:0]），将起始位 STA 置“1”；或配置 TWI1 控制寄存器 SPI1\_TWI1\_CON：配置 TWI1 通信速率（QTWCK[3:0]），将起始位 START 置“1”
- ③ 配置 TWI0 数据寄存器 TWI0\_DATA：将“从机地址+读写位”写入 TWIDAT，总线上发出地址帧；或配置 SPI1\_TWI1 二合一数据寄存器 SPI1\_TWI1\_DATA：将“从机地址+读写位”写入 QTWIDAT，总线上发出地址帧
- ④ 如果主机接收数据，则等待 TWI0\_STS 中的中断标志位 TWIF 置 1，或等待 SPI1\_TWI1\_STS 中的中断标志位 QTWIF 置 1。主机每接收到 8 位数据，中断标志位会被置 1。中断标志位需手动清零；
- ⑤ 如果主机发送数据，则要将待发送的数据写进 TWIDAT/QTWIDAT 中，TWI 会自动将数据发送出去。每发送 8 位，中断标志位 TWIF/QTWIF 就会被置 1。
- ⑥ 数据接收发送完成，主机可发送停止条件（TWI0 为 STO=1，TWI1 为 STOP=1），主机状态切换为 000。或发送重复起始信号，开始新一轮的数据传输。

注意：主机产生 stop 之后主机的 TWIF/QTWIF 不会置位！

## 19.9 TWI0 中断

对于 TWI0，在发生以下事件都可以触发中断，所有的 TWI 事件共用一个中断标志位。

中断事件	事件标志位	中断请求控制位
主机模式，发送完启动信号	TWIF	TWI0_IDE ->INTEN
主机模式，发送完地址帧		
主机模式，接收或发送完数据帧		
从机模式，第一帧地址匹配成功		
从机模式，成功接收或发送 8 位数据		
从机模式，接收到重复起始条件		
从机模式，收到停止信号		

## 19.10 TWI1 中断

对于 TWI1，在发生以下事件都可以触发中断，所有的 TWI 事件共用一个中断标志位。

中断事件	事件标志位	中断请求控制位
主机模式，发送完启动信号	QTWIF	SPI1_TWI1_IDE ->INTEN
主机模式，发送完地址帧		
主机模式，接收或发送完数据帧		
从机模式，第一帧地址匹配成功		
从机模式，成功接收或发送 8 位数据		
从机模式，接收到重复起始条件		
从机模式，收到停止信号		

## 19.11 TWI0 寄存器

### 19.11.1 TWI0 相关寄存器表

#### 19.11.1.1 TWI0 控制寄存器 TWI0\_CON

寄存器	读/写	说明	复位值	上电初始值
TWI0_CON	读/写	TWI0 控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8

-	SPOS[1:0]		-	TWCK[3:0]			
7	6	5	4	3	2	1	0
TWEN	-	STA	STO	-	-	AA	STRETCH

位编号	位符号	说明												
14~13	SPOS[1:0]	<p>● TWI0 信号口映射控制位 @TWI0_CON</p> <table border="1"> <thead> <tr> <th>信号</th> <th>SCL0</th> <th>SDA0</th> </tr> </thead> <tbody> <tr> <td>SPOS[1:0]=00</td> <td>PA0</td> <td>PA1</td> </tr> <tr> <td>SPOS[1:0]=01</td> <td>PB7</td> <td>PB6</td> </tr> <tr> <td>SPOS[1:0]=10</td> <td>PB3</td> <td>PB4</td> </tr> </tbody> </table>	信号	SCL0	SDA0	SPOS[1:0]=00	PA0	PA1	SPOS[1:0]=01	PB7	PB6	SPOS[1:0]=10	PB3	PB4
信号	SCL0	SDA0												
SPOS[1:0]=00	PA0	PA1												
SPOS[1:0]=01	PB7	PB6												
SPOS[1:0]=10	PB3	PB4												
11~8	TWCK[3:0]	<p>主机模式下 TWI 的通信速率设定:</p> <p>0000: f<sub>PCLK</sub> /4096 0001: f<sub>PCLK</sub> /2048 0010: f<sub>PCLK</sub> /1024 0011: f<sub>PCLK</sub> /512 0100: f<sub>PCLK</sub> /256 0101: f<sub>PCLK</sub> /128 0110: f<sub>PCLK</sub> /64 0111: f<sub>PCLK</sub> /32 1000: f<sub>PCLK</sub> /16 1001: f<sub>PCLK</sub> /8 1010: f<sub>PCLK</sub> /4 其它: f<sub>PCLK</sub> /4</p> <p>TWI0 的通信速率 f<sub>TWI0</sub> = f<sub>PCLK0</sub></p> <p><b>注意: TWI0 作主机时极限频率为 4M, 作从机时极限频率为 4M</b></p>												
7	TWEN	<p>TWI 使能控制位</p> <p>0: 关闭 TWI 1: 打开 TWI</p>												
5	STA	<p>TWI 起始位触发开关</p> <p>该位写 1 将产生起始条件, TWI 将切换为主机模式。 软件可以设置或清除该位, 或当起始条件发出后, 由硬件清 0。</p>												
4	STO	<p>TWI 停止位触发开关</p> <p>主机模式下, 对该位写 1, 在当前字节传输或起始条件发出后产生停止条件。 软件可以设置或清除该位, 或当检测到停止条件时, 由硬件清除。</p>												
1	AA	<p>TWI 应答使能位</p> <p>0: 无应答, 返回 UACK (应答位为高电平) 1: 在接收到一个匹配的地址或数据后返回一个应答 ACK</p>												
0	STRETCH	<p>TWI 时钟延长使能位</p> <p>该位仅在从机模式下有效</p> <p>0: 禁止时钟延长 1: 允许时钟延长, 主机需要支持时钟延长功能</p> <p>说明: 在数据传输完成之后, 且 ACK 为 0, 此时时钟延长发生。</p>												
31~15 12 6 3~2	-	保留												

### 19.11.1.2 TWI0 标志状态位寄存器 TWI0\_STS

寄存器	读/写	说明	复位值	上电初始值
TWI0_STS	读/写	TWI0 标志状态位寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
NBYTES[7:0]							
15	14	13	12	11	10	9	8
-	-	-	-	-	STATE[2:0]		
7	6	5	4	3	2	1	0
-	-	-	-	MSTR	GCA	TXnE/RXnE	TWIF

位编号	位符号	说明
23~16	NBYTES[7:0]	收发缓存字节数设置位 用于设置待发送/接收的字节数。 每发送/接收成功一个自己 NBYTES 自动减 1，当 NBYTES 为 0 时 TC 标志位将置起。 <b>注意：STA 置 1 时不允许修改。</b>
10~8	STATE[2:0]	TWI 状态位 用于指示 TWI 状态，主/从机模式下状态位含义不同。 <ul style="list-style-type: none"> <li>从机模式： <ul style="list-style-type: none"> <li>000: 从机处于空闲状态，等待 TWEN 置 1，检测 TWI 启动信号。当从机接收到停止条件后跳转到此状态</li> <li>001: 从机正在接收第一帧地址和读写位（第 8 位为读写位，1 为读，0 为写）。从机接收到起始条件后会跳转到此状态</li> <li>010: 从机接收数据状态</li> <li>011: 从机发送数据状态</li> <li>100: 在从机发送数据状态中，当主机回 UACK 时跳转到此状态，等待重新启动信号或停止信号</li> <li>101: 从机处于发送状态时，将 AA 写 0 会进入此状态，等待重新启动信号或停止信号</li> <li>110: 从机的地址与主机发送的地址不匹配会跳转到此状态，等待新的起始条件或停止条件</li> </ul> </li> <li>主机模式： <ul style="list-style-type: none"> <li>000: 状态机为空闲状态</li> <li>001: 主机发送起始条件或主机正在发送从设备地址</li> <li>010: 主机发送数据</li> <li>011: 主机接收数据</li> <li>100: 主机发送停止条件或接收到从机的 UACK 信号</li> </ul> </li> </ul>
3	MSTR	TWI 主/从机模式标志位 0: 从机模式 1: 主机模式 <b>说明：</b> 1. 当 TWI 接口向总线发出起始条件后，会自动切换为主机模式，同时硬件将该位置位； 2. 当总线上检测到一个停止条件时，硬件清除该位。
2	GCA	TWI 通用地址响应标志位 0: 非响应通用地址 1: 当 GC 置 1，同时通用地址匹配时该位由硬件置 1，并自动清 0
1	TXnE/RXnE	TWI 传输完成标志位 以下情况，TXnE/RXnE 由硬件置 1 <ul style="list-style-type: none"> <li>主机模式： <ul style="list-style-type: none"> <li>■ 主机发送地址帧（写），且收到从机的 ACK</li> <li>■ 主机发送完数据，且接收到从机 ACK</li> <li>■ 主机接收到数据，且主机回从机 ACK</li> </ul> </li> <li>从机模式：</li> </ul>

位编号	位符号	说明
		<ul style="list-style-type: none"> <li>■ 从机接收地址帧（读），且和从机地址（TWA）匹配</li> <li>■ 从机接收到数据，且从机回主机 ACK</li> <li>■ 从机发送完数据，且接收到主机 ACK（AA=1）</li> </ul> 对 TWIDAT 进行读写操作后，该位将被硬件清除。
0	TWIF	TWI 中断标志位 该位由硬件置 1，通过软件写 1 清 0。 <ul style="list-style-type: none"> <li>● 主机模式：                             <ul style="list-style-type: none"> <li>■ 发送启动信号</li> <li>■ 发送完地址帧</li> <li>■ 接收或发送完数据帧</li> </ul> </li> <li>● 从机模式：                             <ul style="list-style-type: none"> <li>■ 第一帧地址匹配成功</li> <li>■ 成功接收或发送 8 位数据</li> <li>■ 接收到重复起始条件</li> <li>■ 从机收到停止信号</li> </ul> </li> </ul>
31~24 15~11 7~4	-	保留

### 19.11.1.3 TWI0 地址寄存器 TWI0\_ADD

寄存器	读/写	说明	复位值	上电初始值
TWI0_ADD	读/写	TWI0 地址寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
7~1	TWA[6:0]	TWI 地址寄存器 TWA[6:0]不能写为全 0，00H 为通用地址寻址专用。 该位在主机模式下无效。
0	GC	TWI 通用地址响应使能位 0：禁止响应通用地址 00H 1：允许响应通用地址 00H
31~8	-	保留

### 19.11.1.4 TWI0 数据寄存器 TWI0\_DATA

寄存器	读/写	说明	复位值	上电初始值
TWI0_DATA	读/写	TWI0 数据寄存器	0x0000_0000	0x0000_0000

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

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

位编号	位符号	说明
7~0	TWIDAT[7:0]	TWI 数据缓存 读操作：从 TWI 接收缓存读取接收到的数据 写操作：向 TWI 发送缓存写入待发送的数据
31~8	-	保留

### 19.11.1.5 TWI0 的中断使能及 DMA 控制寄存器 TWI0\_IDE

寄存器	读/写	说明	复位值	上电初始值
TWI0_IDE	读/写	TWI0 的中断使能及 DMA 控制寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
7	TXDMAEN	DMA 发送通道使能位 0: 禁用 DMA 发送功能 1: 使能 DMA 发送功能 该位使能后, TXnE 置起可触发 DMA 通道发送请求
6	RXDMAEN	DMA 接收通道使能位 0: 禁用 DMA 接收功能 1: 使能 DMA 接收功能 该位使能后, RXnE 置起可触发 DMA 通道接收请求
0	INTEN	中断请求 CPU 的使能控制位 0: 禁止中断请求 1: 使能中断请求
31~8 5~1	-	保留

### 19.11.2 TWI0 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值
TWI0 基地址: 0x4002_0060					
TWI0_CON	0x00	读/写	TWI0 控制寄存器	0x0000_0000	0x0000_0000
TWI0_STS	0x04	读/写	TWI0 标志位寄存器	0x0000_0000	0x0000_0000
TWI0_ADD	0x08	读/写	TWI0 地址寄存器	0x0000_0000	0x0000_0000

寄存器	偏移地址	读/写	说明	复位值	上电初始值
TWI0_DATA	0x0C	读/写	TWI0 数据寄存器	0x0000_0000	0x0000_0000
TWI0_IDE	0x10	读/写	通信口 DMA 控制寄存器	0x0000_0000	0x0000_0000

## 19.12 TWI1 寄存器

### 19.12.1 TWI1 相关寄存器表

#### 19.12.1.1 SPI1\_TWI1 二合一控制寄存器 SPI1\_TWI1\_CON

寄存器	读/写	说明	复位值	上电初始值
SPI1_TWI1_CON	读/写	SPI1_TWI1 二合一控制寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
MODE[1:0]		-	CPOL	CPHA	DORD	SPMD	SMSTR
15	14	13	12	11	10	9	8
-	SPOS[1:0]		-	QWCK[3:0]			
7	6	5	4	3	2	1	0
QWEN	-	START	STOP	-	-	AA	STRETCH

位编号	位符号	说明															
23~22	MODE[1:0]	SPI1_TWI1 模式选择位 00: TWI 模式 01: SPI 模式 10: 保留 11: 保留															
14~13	SPOS[1:0]	<p>● TWI1 信号口映射控制位@SPI1_TWI1_CON</p> <table border="1"> <thead> <tr> <th>信号</th> <th>SCL1</th> <th>SDA1</th> </tr> </thead> <tbody> <tr> <td>SPOS[1:0]=00</td> <td>PB14</td> <td>PB15</td> </tr> <tr> <td>SPOS[1:0]=01</td> <td>PA6</td> <td>PA7</td> </tr> <tr> <td>SPOS[1:0]=10</td> <td>PA11</td> <td>PA12</td> </tr> <tr> <td>SPOS[1:0]=11</td> <td>PB2</td> <td>PB3</td> </tr> </tbody> </table>	信号	SCL1	SDA1	SPOS[1:0]=00	PB14	PB15	SPOS[1:0]=01	PA6	PA7	SPOS[1:0]=10	PA11	PA12	SPOS[1:0]=11	PB2	PB3
信号	SCL1	SDA1															
SPOS[1:0]=00	PB14	PB15															
SPOS[1:0]=01	PA6	PA7															
SPOS[1:0]=10	PA11	PA12															
SPOS[1:0]=11	PB2	PB3															
11~8	QWCK[3:0]	<p>主机模式下 TWI 的通信速率设定:</p> <p>0000: f<sub>PCLK1</sub>                      0001: f<sub>PCLK1</sub> /2                      0010: f<sub>PCLK1</sub> /4                      0011: f<sub>PCLK1</sub> /8                      0100: f<sub>PCLK1</sub> /16                      0101: f<sub>PCLK1</sub> /32                      0110: f<sub>PCLK1</sub> /64                      0111: f<sub>PCLK1</sub> /128                      1000: f<sub>PCLK1</sub> /256                      1001: f<sub>PCLK1</sub> /512                      1010: f<sub>PCLK1</sub> /1024                      1011: f<sub>PCLK1</sub> /2048                      1100: f<sub>PCLK1</sub> /4096</p>															

位编号	位符号	说明
		其它: f <sub>PCLK1</sub> /4096 <b>注意: TWI1 作主机时极限频率为 4M, 作从机时极限频率为 2M</b>
7	QTWEN	SPI1_TWI1 模块使能控制位 0: 关闭模块 1: 打开模块
5	START	TWI 起始位触发开关 该位写 1 将产生起始条件, TWI 将切换为主机模式。 软件可以设置或清除该位, 或当起始条件发出后, 由硬件清 0。
4	STOP	TWI 停止位触发开关 主机模式下, 对该位写 1, 在当前字节传输或起始条件发出后产生停止条件。 软件可以设置或清除该位, 或当检测到停止条件时, 由硬件清除。
1	AA	TWI 应答使能位 0: 无应答, 返回 UACK (应答位为高电平) 1: 在接收到一个匹配的地址或数据后返回一个应答 ACK
0	STRETCH	TWI 时钟延长使能位 该位仅在从机模式下有效 0: 禁止时钟延长 1: 允许时钟延长, 主机需要支持时钟延长功能 说明: 在数据传输完成之后, 且 ACK 为 0, 此时时钟延长发生。
31~24 21 15 12 6 3~2	-	保留

### 19.12.1.2 SPI1\_TWI1 二合一标志状态位寄存器 SPI1\_TWI1\_STS

寄存器	读/写	说明	复位值	上电初始值
SPI1_TWI1_STS	读/写	SPI1_TWI1 二合一标志状态位寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
NBYTES[7:0]							
15	14	13	12	11	10	9	8
-	-	-	-	-	STATE[2:0]		
7	6	5	4	3	2	1	0
WCOL	TXEIF	-	-	TMSTR	GCA	TXnE/RXnE	QTWIF

位编号	位符号	说明
23~16	NBYTES[7:0]	收发缓存字节数设置位 用于设置待发送/接收的字节数。 每发送/接收成功一个自己 NBYTES 自动减 1, 当 NBYTES 为 0 时 TC 标志位将置起。 <b>注意: START 置 1 时不允许修改。</b>
10~8	STATE[2:0]	TWI 状态位 用于指示 TWI 状态, 主/从机模式下状态位含义不同。 ● 从机模式:

位编号	位符号	说明
		<p>000: 模块处于空闲状态, 等待 TWEN 置 1, 检测 TWI 启动信号。模块检测到停止条件后跳转到此状态</p> <p>001: 从机正在接收第一帧地址和读写位 (第 8 位为读写位, 1 为读, 0 为写)。从机接收到起始条件后会跳转到此状态</p> <p>010: 从机接收数据状态</p> <p>011: 从机发送数据状态</p> <p>111: 从机处于意外状态</p> <p>从机进入意外状态的条件如下:</p> <ul style="list-style-type: none"> <li>■ 在从机发送数据状态中, 当主机回 UACK 时跳转到此状态, 等待重新启动信号或停止信号</li> <li>■ 从机处于发送状态时, 将 AA 写 0 会进入此状态, 等待重新启动信号或停止信号</li> <li>■ 从机处于接收状态时, 回复 UACK 时跳转到此状态, 等待重新启动信号或停止信号</li> <li>■ 从机的地址与主机发送的地址不匹配会跳转到此状态, 等待新的起始条件或停止条件</li> </ul> <p>● 主机模式:</p> <p>000: 状态机为空闲状态</p> <p>001: 主机发送起始条件或主机正在发送从设备地址</p> <p>010: 主机发送数据</p> <p>011: 主机接收数据</p> <p>111: 模块处于意外状态</p> <p>主机进入意外状态的条件如下:</p> <ul style="list-style-type: none"> <li>■ 主机发送停止条件或接收到从机的 UACK 信号会跳转到此状态</li> <li>■ 从机的地址与主机发送的地址不匹配会跳转到此状态</li> </ul>
3	TMSTR	<p>TWI 主/从机模式标志位</p> <p>0: 从机模式</p> <p>1: 主机模式</p> <p>说明: 当 TWI 接口向总线发出起始条件后, 会自动切换为主机模式, 同时硬件将该位置位; 当总线上检测到一个停止条件时, 硬件清除该位。</p>
2	GCA	<p>TWI 通用地址响应标志位</p> <p>0: 非响应通用地址</p> <p>1: 当 GC 置 1, 同时通用地址匹配时该位由硬件置 1, 并自动清 0</p>
1	TXnE/RXnE	<p>TWI 传输完成标志位</p> <p>以下情况, TXnE/RXnE 由硬件置 1</p> <p>● 主机模式:</p> <ul style="list-style-type: none"> <li>■ 主机发送地址帧 (写), 且收到从机的 ACK</li> <li>■ 主机发送完数据, 且接收到从机 ACK</li> <li>■ 主机接收到数据, 且主机回从机 ACK</li> </ul> <p>● 从机模式:</p> <ul style="list-style-type: none"> <li>■ 从机接收地址帧 (读), 且和从机地址 (TWA) 匹配</li> <li>■ 从机接收到数据, 且从机回主机 ACK</li> <li>■ 从机发送完数据, 且接收到主机 ACK (AA=1)</li> </ul> <p>对 QTWIDAT 进行读写操作后, 该位将被硬件清除。</p>
0	QTWIF	<p>TWI 中断标志位</p> <p>该位由硬件置 1, 通过软件写 1 清 0。</p> <p>● 主机模式:</p> <ul style="list-style-type: none"> <li>■ 发送启动信号</li> <li>■ 发送完地址帧</li> </ul>

位编号	位符号	说明
		<ul style="list-style-type: none"> <li>■ 接收或发送完数据帧</li> <li>● 从机模式： <ul style="list-style-type: none"> <li>■ 第一帧地址匹配成功</li> <li>■ 成功接收或发送 8 位数据</li> <li>■ 接收到重复起始条件</li> <li>■ 从机收到停止信号</li> </ul> </li> </ul>
31~24 15~11 5~4	-	保留

### 19.12.1.3 TWI1 地址寄存器 TWI1\_ADD

寄存器	读/写	说明	复位值	上电初始值
TWI1_ADD	读/写	TWI1 地址寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
7~1	QTWADD[6:0]	TWI 地址寄存器 QTWADD[6:0]不能写为全 0，00H 为通用地址寻址专用。 该位在主机模式下无效。
0	GC	TWI 通用地址响应使能位 0：禁止响应通用地址 00H 1：允许响应通用地址 00H
31~8	-	保留

### 19.12.1.4 SPI1\_TWI1 二合一数据寄存器 SPI1\_TWI1\_DATA

寄存器	读/写	说明	复位值	上电初始值
SPI1_TWI1_DATA	读/写	SPI1_TWI1 二合一数据寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
7~0	QTWIDAT[7:0]	TWI 数据缓存 读操作：从 TWI 接收缓存读取接收到的数据 写操作：向 TWI 发送缓存写入待发送的数据
31~16	-	保留

### 19.12.1.5 SPI1\_TW11 二合一中断使能及 DMA 控制寄存器 SPI1\_TW11\_IDE

寄存器	读/写	说明	复位值	上电初始值
SPI1_TW11_IDE	读/写	SPI1_TW11 二合一中断使能及 DMA 控制寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
0	INTEN	中断请求 CPU 的使能控制位 0：禁止中断请求 1：使能中断请求
31~8 5~2	-	保留

### 19.12.2 TWI1 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值
SPI1_TW11 基地址：0x4002_1040					
SPI1_TW11_CON	0x00	读/写	SPI1_TW11 二合一控制寄存器	0x0000_0000	0x0000_0000
SPI1_TW11_STS	0x04	读/写	SPI1_TW11 二合一标志位寄存器	0x0000_0000	0x0000_0000
TWI1_ADD	0x08	读/写	TWI1 地址寄存器	0x0000_0000	0x0000_0000
SPI1_TW11_DATA	0x0C	读/写	SPI1_TW11 二合一数据寄存器	0x0000_0000	0x0000_0000
SPI1_TW11_IDE	0x10	读/写	SPI1_TW11 中断使能及 DMA 控制寄存器	0x0000_0000	0x0000_0000

## 20 CAN 通信口

### 20.1 概述

SC32F15G 系列的控制器局域网（CAN）支持 CAN2.0B 协议与 CAN\_FD 协议的通信，相比于 CAN2.0B 协议，CAN\_FD 具有更高的灵活性，其位速率从原来的仅 1Mbit/s 更改为可变，数据段长度最多可达 64 字节。支持四种不同的工作模式，可设置低功耗待机模式，支持待机唤醒。

发送缓冲区支持 PTB 主传输缓冲区与 STB 次传输缓冲区这两种发送缓冲区可供选择，可选用 FIFO 模式或优先权模式决定发送顺序；接收缓冲区可同时容纳 8 帧数据，且每个接收帧都有单独时间戳。接收过滤器有 8 组可供选择，每个过滤器都可单独启用，设置过滤条件。

### 20.2 时钟源

AHB 时钟总线，时钟源跟随 HCLK。

### 20.3 特性

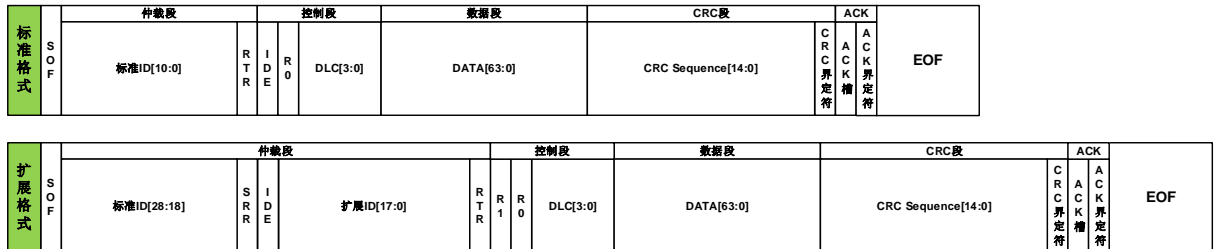
- 协议支持：
  - CAN 2.0B
    - ◆ 支持标准格式和扩展格式，最多可负载 8bytes 数据
    - ◆ 速率可到 1Mbit/s
  - CAN FD
    - ◆ 支持标准格式和扩展格式，最多可负载 64bytes 数据
    - ◆ 速率可变
- 待机模式：使能此模式后，CAN 收发器将进入低功耗状态并不再接收数据帧，仅检测 CAN 总线上的显性电平
- 时间戳：
  - CiA 603：64 位时间戳，发送帧（TTS）支持一个时间戳，存放在寄存器里，但所有的接收帧（RTS）都有单独的时间戳
- 收发缓存：
  - 8 组接收缓存（RB），每个接收帧（RTS）有单独时间戳，与数据一同存放在 RB 中。RB 的工作方式同 FIFO
  - 9 组发送缓存（TB）：
    - ◆ 1 个 Primary Transmit Buffer PTB
    - ◆ 8 个 Secondary Transmit Buffer STB，支持两种传输模式：FIFO 模式和优先权决定模式
  - 8 组接收过滤器（支持 29bit ID）

## 20.4 CAN 协议

### 20.4.1 CAN2.0 协议

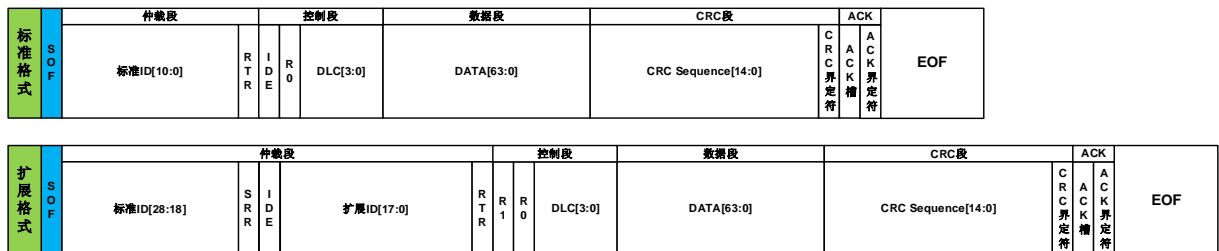
CAN2.0 传统协议具有以下特性：

- 具有帧起始段，仲裁段，控制段，数据段，CRC 段，ACK 段，帧结束段
- ID 标识符可选择标准帧或扩展帧
- 可以选择发送数据帧或远程帧
- 最大数据长度为 8 字节
- 比特率固定为低速比特率寄存器配置
- CRC 校验位为 15 位



#### 20.4.1.1 帧起始段

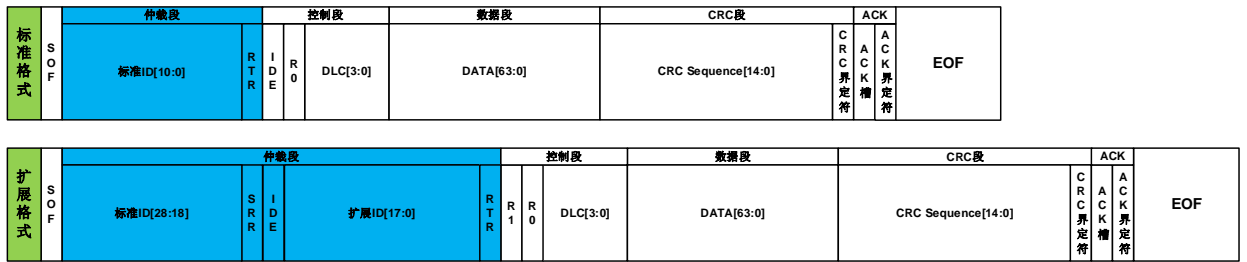
在 CAN2.0 协议中使用 SOF 显性位表示报文的起始，并在总线中起同步作用。



#### 20.4.1.2 仲裁段

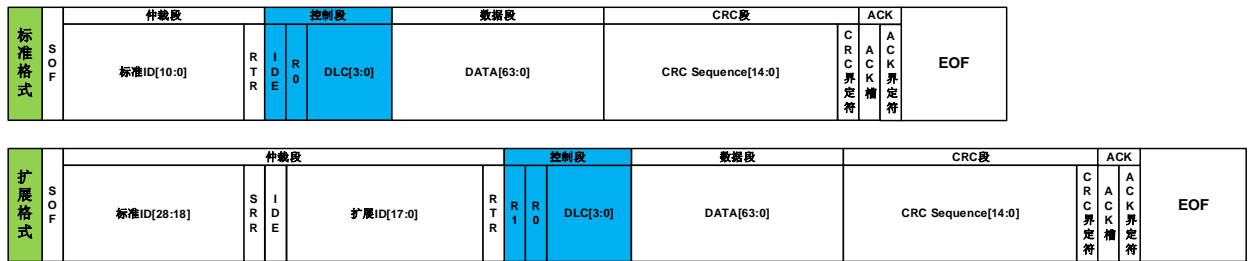
在 CAN2.0 协议中仲裁段包括 ID 标识符段，RTR 远程帧控制位以及 IDE 标识符扩展位（扩展帧情况下），用户可通过设置 IDE 控制该帧是否启用标识符扩展位。

仲裁段中的 ID 标识符可作为优先级判断的依据，当同一时刻有多个节点需要占用总线时，根据显性优先级高于隐性的原则，ID 标识符越小其优先级越高；ID 相同时，RTR 表示为显性的数据帧比 RTR 表示为隐性的远程帧优先级高。除此之外，ID 标识符还可以作为过滤器使用过滤功能的判断依据。



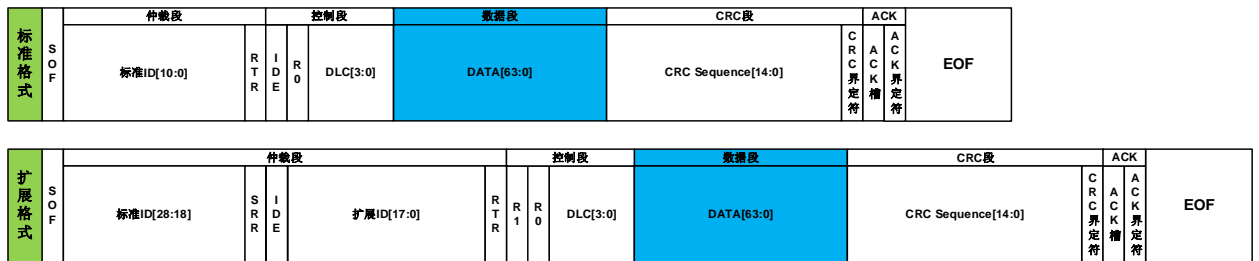
### 20.4.1.3 控制段

在 CAN2.0 协议中控制段包括 IDE 标识符扩展位（标准帧情况下），保留位以及 DLC 数据长度位，在 CAN2.0 帧中数据长度最大值可设定为 8 字节。



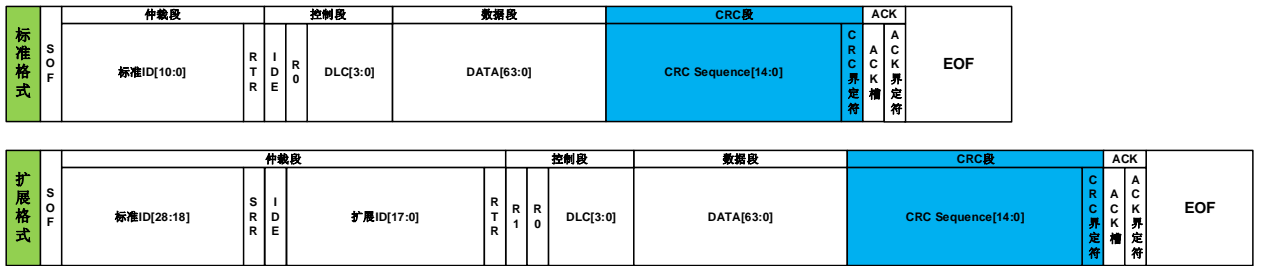
### 20.4.1.4 数据帧与远程帧

在 CAN2.0 协议中存在 RTR 位可以控制发送的报文包含数据帧或者是远程帧。选择数据帧时，用户将需要发送的数据写入数据段，并正常通过发送流程发送；选择远程帧时，当某一节点需要数据时，可以发送远程帧请求另一节点发送相同 ID 的数据帧。相比于数据帧，远程帧的 RTR 位为隐性，而数据帧的 RTR 位为显性。且远程帧没有数据段，其 DLC 数据长度为请求的数据帧数据长度。



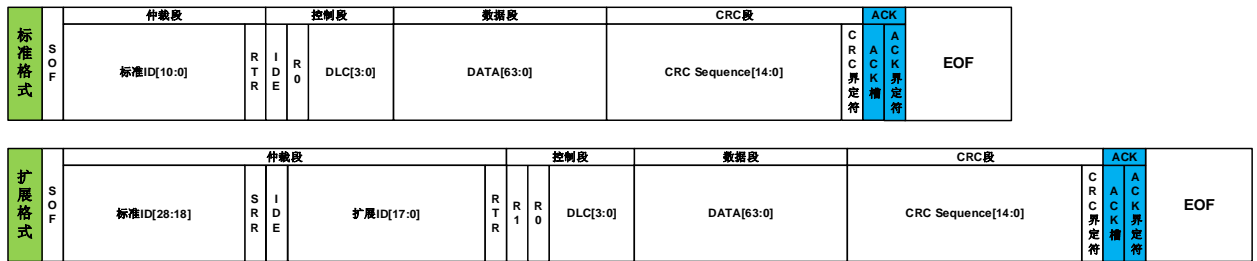
### 20.4.1.5 CRC 段

CRC 段即循环冗余校验段，在 CAN2.0 协议中 CRC 段包含 CRC 序列与 CRC 界定符组成，CRC 序列是根据特定多项式计算得到的值，计算范围包括帧起始段，仲裁段，控制段以及数据段，发送方与接收方会以同样的算法计算 CRC 序列值，对比若不一致则说明发生通讯错误。CRC 界定符是 CRC 序列后一个单独的隐性位，用于分隔 CRC 段与 ACK 段。



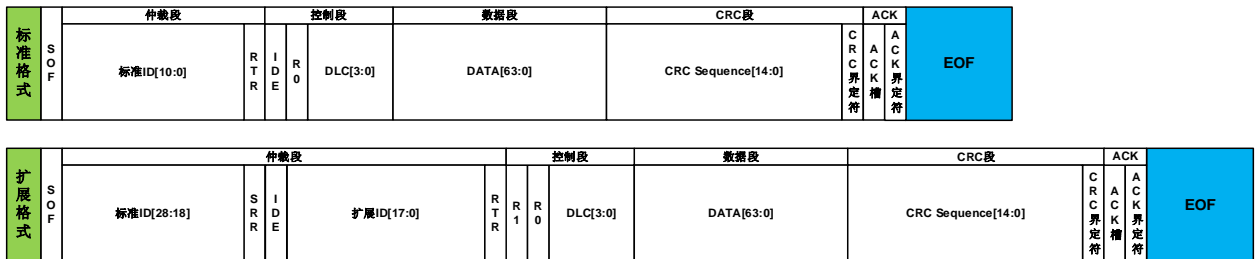
### 20.4.1.6 ACK 段

在 CAN2.0 的 ACK 段由一个 ACK 位和一个 ACK 界定符组成。ACK 位在发送帧中为隐性，若接收方确认接收无误，则会回复一个显性 ACK 位，表示传输正确完成。ACK 界定符恒为隐性，起分隔作用。



### 20.4.1.7 帧结束段

CAN2.0 中帧结束段为 7 个隐性位，表示一帧报文结束。



## 20.4.2 CAN\_FD 协议

CAN\_FD 协议又称 CAN 灵活数据率协议，其具有以下特点：

- 可兼容 CAN2.0 协议
- 具有帧起始段，仲裁段，控制段，数据帧，CRC 段，ACK 段，帧结束段
- ID 标识符可选择标准帧或扩展帧
- 只能选择发送数据帧
- 最大数据长度为 64 字节
- 可通过设置 BRS 位使数据段获得比仲裁段更高的比特率
- 控制段中有 ESI 位表示错误状态
- CRC 将位填充加入计算，最多可有 21 个 CRC 校验位

标准格式	SOF	仲裁段			控制段			数据段			CRC段		ACK		EOF
		标准ID[10:0]	RRS	IDE	EDL	RO	BRS	ESI	DLC[3:0]	最多64个字节	CRC Sequence[16:0]或 CRC Sequence[20:0]	CRC 界定符	ACK 槽	ACK 界定符	
扩展格式	SOF	仲裁段			控制段			数据段			CRC段		ACK		EOF
		标准ID[28:18]	RRS	IDE	EDL	RO	BRS	ESI	DLC[3:0]	最多64个字节	CRC Sequence[16:0]或 CRC Sequence[20:0]	CRC 界定符	ACK 槽	ACK 界定符	

### 20.4.2.1 帧起始段

在 CAN\_FD 协议与 CAN2.0 协议中都使用 SOF 显性位表示报文的起始，并在总线中起同步作用。

标准格式	SOF	仲裁段			控制段			数据段			CRC段		ACK		EOF
		标准ID[10:0]	RRS	IDE	EDL	RO	BRS	ESI	DLC[3:0]	最多64个字节	CRC Sequence[16:0]或 CRC Sequence[20:0]	CRC 界定符	ACK 槽	ACK 界定符	
扩展格式	SOF	仲裁段			控制段			数据段			CRC段		ACK		EOF
		标准ID[28:18]	RRS	IDE	EDL	RO	BRS	ESI	DLC[3:0]	最多64个字节	CRC Sequence[16:0]或 CRC Sequence[20:0]	CRC 界定符	ACK 槽	ACK 界定符	

### 20.4.2.2 仲裁段

CAN\_FD 中仲裁段包括 ID 标识符段与 IDE 标识符扩展位（扩展帧情况下），与 CAN2.0 相比 RTR 位被替换成了 RRS 位（下图中称为 r1），恒为显性。用户可通过设置 IDE 控制该帧是否启用标识符扩展位。

仲裁段中的 ID 标识符可作为优先级判断的依据，当同一时刻有多个节点需要占用总线时，根据显性优先级高于隐性的原则，ID 标识符越小其优先级越高。除此之外，ID 标识符还可以作为过滤器使用过滤功能的判断依据。

标准格式	SOF	仲裁段			控制段			数据段			CRC段		ACK		EOF
		标准ID[10:0]	RRS	IDE	EDL	RO	BRS	ESI	DLC[3:0]	最多64个字节	CRC Sequence[16:0]或 CRC Sequence[20:0]	CRC 界定符	ACK 槽	ACK 界定符	
扩展格式	SOF	仲裁段			控制段			数据段			CRC段		ACK		EOF
		标准ID[28:18]	RRS	IDE	EDL	RO	BRS	ESI	DLC[3:0]	最多64个字节	CRC Sequence[16:0]或 CRC Sequence[20:0]	CRC 界定符	ACK 槽	ACK 界定符	

### 20.4.2.3 控制段

CAN\_FD 中控制段包括 IDE 标识符扩展位（标准帧情况下），保留位，EDL 扩展数据长度位，BRS 比特率切换位，ESI 错误状态指示位以及 DLC 数据长度位，且 CAN\_FD 帧中数据长度最大值增加到 64 字节。

CAN\_FD 协议中新增的 EDL 位为隐性时表示该帧为 CAN\_FD 帧，BRS 位可控制仲裁段与数据段传输选用不同波特率，ESI 位可指示节点处于主动错误模式或者是被动错误模式。

标准格式	SOF	仲裁段			控制段				数据段				CRC段		ACK		EOF											
		标准ID[10:0]	R	R	R	I	D	E	L	R	0	B	R	S	E	S		I	DLC[3:0]	最多64个字节	CRC Sequence[16:0]或 CRC Sequence[20:0]	CRC 界定符	ACK 槽	ACK 界定符				
扩展格式	SOF	仲裁段			控制段				数据段				CRC段		ACK		EOF											
		标准ID[28:18]	S	R	R	I	D	E	L	扩展ID[17:0]	R	R	R	E	D	L		R	0	B	R	S	E	S	I	DLC[3:0]	最多64个字节	CRC Sequence[16:0]或 CRC Sequence[20:0]

#### 20.4.2.4 数据帧

CAN\_FD 协议中只能选择数据帧，不能选择远程帧。选择数据帧时，用户将需要发送的数据写入数据段，并通过发送流程发送。

标准格式	SOF	仲裁段			控制段				数据段				CRC段		ACK		EOF											
		标准ID[10:0]	R	R	R	I	D	E	L	R	0	B	R	S	E	S		I	DLC[3:0]	最多64个字节	CRC Sequence[16:0]或 CRC Sequence[20:0]	CRC 界定符	ACK 槽	ACK 界定符				
扩展格式	SOF	仲裁段			控制段				数据段				CRC段		ACK		EOF											
		标准ID[28:18]	S	R	R	I	D	E	L	扩展ID[17:0]	R	R	R	E	D	L		R	0	B	R	S	E	S	I	DLC[3:0]	最多64个字节	CRC Sequence[16:0]或 CRC Sequence[20:0]

#### 20.4.2.5 CRC 段

在 CAN\_FD 协议中 CRC 段组成基本与 CAN2.0 中一致，但其在 CAN2.0 的基础上对 CRC 算法进行了改进，首先是由于 CAN\_FD 支持最大数据长度上升至 64 字节，所以 CRC 序列位上升至 17 位，最大可达 21 位，其位数由发送数据长度决定；其次 CAN\_FD 中支持位填充机制，将填充位加入了 CRC 序列计算，确保了 CAN\_FD 的稳健性。

标准格式	SOF	仲裁段			控制段				数据段				CRC段		ACK		EOF											
		标准ID[10:0]	R	R	R	I	D	E	L	R	0	B	R	S	E	S		I	DLC[3:0]	最多64个字节	CRC Sequence[16:0]或 CRC Sequence[20:0]	CRC 界定符	ACK 槽	ACK 界定符				
扩展格式	SOF	仲裁段			控制段				数据段				CRC段		ACK		EOF											
		标准ID[28:18]	S	R	R	I	D	E	L	扩展ID[17:0]	R	R	R	E	D	L		R	0	B	R	S	E	S	I	DLC[3:0]	最多64个字节	CRC Sequence[16:0]或 CRC Sequence[20:0]

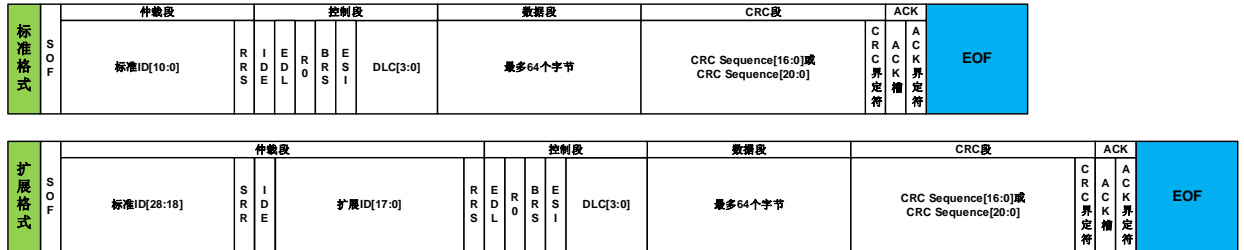
#### 20.4.2.6 ACK 段

在 CAN\_FD 与 CAN2.0 协议中 ACK 段构成相同，都由一个 ACK 位和一个 ACK 界定符组成。ACK 位在发送帧中为隐性，若接收方确认接收无误，则会回复一个显性 ACK 位，表示传输正确完成。ACK 界定符恒为隐性，起分隔作用。

标准格式	SOF	仲裁段			控制段				数据段				CRC段		ACK		EOF											
		标准ID[10:0]	R	R	R	I	D	E	L	R	0	B	R	S	E	S		I	DLC[3:0]	最多64个字节	CRC Sequence[16:0]或 CRC Sequence[20:0]	CRC 界定符	ACK 槽	ACK 界定符				
扩展格式	SOF	仲裁段			控制段				数据段				CRC段		ACK		EOF											
		标准ID[28:18]	S	R	R	I	D	E	L	扩展ID[17:0]	R	R	R	E	D	L		R	0	B	R	S	E	S	I	DLC[3:0]	最多64个字节	CRC Sequence[16:0]或 CRC Sequence[20:0]

### 20.4.2.7 帧结束段

在 CAN\_FD 与 CAN2.0 中帧结束段都为 7 个隐性位，表示一帧报文结束。



## 20.5 功能描述

### 20.5.1 配置波特率

CAN 通信时的波特率可通过低速寄存器 CAN\_S\_SEG 和高速寄存器 CAN\_F\_SEG 配置，在 CAN\_FD 模式下使能 BRS 位可支持数据段和仲裁段切换不同波特率，以提高数据段发送速率；其中低速寄存器 CAN\_S\_SEG 配置仲裁段波特率，高速寄存器 CAN\_F\_SEG 配置数据段波特率。而在 CAN2.0 模式下仲裁段与数据段的波特率只能通过低速寄存器 CAN\_S\_SEG 配置，不使用高速寄存器 CAN\_F\_SEG。

波特率计算公式如下：

$$\text{Baud} = \text{clock} / (((\text{SEG1} + 2) + (\text{SEG2} + 1)) * (\text{PRESC} + 1))$$

其中 Baud 为波特率，clock 为 CAN 时钟，SEG1 为采样点前时间段减去同步段的相位段 1，SEG2 为采样点后的相位段 2，PRESC 为预分频值。此外寄存器中还需配置 SJW，SJW 为再同步跳变宽度的最大值，不直接影响波特率，作用是在每个报文的 bit 位由于不可控因素出现相位差的情况下，根据相位差调整 SEG1 和 SEG2 的宽度，因此要求 SJW 的值小于等于 SEG1。

### 20.5.2 过滤器功能

当用户想接收特定 ID 的报文时，可启用过滤器功能进行筛选。过滤器主要通过 ACODE 与 AMASK 配合实现过滤功能，ACODE 和 AMASK 都在 CAN\_ACF 寄存器中配置。ID 会与 ACODE 中写入值的每一位进行对比，若全部相同，则通过过滤器；而 AMASK 中写入值中为 1 的位会掩盖 ACODE 中对应位的对比，使得 ID 与 ACODE 对比时无论 AMASK 写 1 的对应位是否相同都认为对比结果是相同的。如果 ID 为拓展帧，则会对比 ID28~0；若 ID 为标准帧，则只对比 ID10~0，ID28~11 不影响结果。例如，在帧格式为标准帧的情况下，ACODE 设为 0x212，AMASK 设为 0x000，那么只能筛选通过 ID 号为 0x212 的报文；若将 AMASK 设为 0x7F0，那么过滤器会筛选通过所有尾号为 2（即 ID 为 0xXX2）的报文。

共有 8 个过滤器供用户选择，ID 会从编号小的已使能过滤器开始依次向编号大的已使能过滤器对比，若通过则被接收，不通过则向上对比，都不通过则都不接收。启用硬件重置，即将 CAN\_CFG\_STAT 寄存器的 RESET 位置 1，默认情况只能使能 0 号过滤器，且默认接收所有 ID 的报文，同时禁用其他过滤器。只有过滤器 0 受开机复位影响，所有其他过滤器均保持未初始化状态。过滤器的选择与使能在 CAN\_ACFCTRL 寄存器中配置，且 AMASK 与 ACODE 的指针指向也在 CAN\_ACFCTRL 寄存器中配置，当指针指向 AMASK 时，CAN\_ACF 寄存器中还可以通过配置 AIDE 与 AIDEE 位实现只接收标准帧或只接受扩展帧的功能；当指针指向 ACODE 时，CAN\_ACF 寄存器中配置 AIDE 与 AIDEE 位不会产生任何影响。

### 20.5.3 发送缓冲区

CAN 模块中有两种发送缓冲区供用户选择，分别为 PTB（Primary Transmit Buffer）主传输缓冲区和 STB（Secondary Transmit Buffer）次传输缓冲区。其中 PTB 只有 1 个 slot，即只能存储 1 帧报文；而 STB 有 8 个 slot，即可存储 8 帧报文。

当 PTB 写入 1 帧报文写满后，需通过使能 TPE 进行报文传输。

当 STB 的 1 个 slot 被写入 1 帧报文后，需要将 TSNEXT 位置 1，将 STB 指向下个 slot，以存储下 1 帧报文，且每次 STB 指向新的 slot 后都需清零发送相关寄存器中的随机数避免随机数对报文写入产生影响。STB 内报文存储数可通过 TSSTAT 位进行查看，无论其是否写满，都可以通过使能 TSONE 或 TSALL 进行报文传输。TSONE 与 TSALL 的区别在于 TSONE 是每次传输 STB 缓冲区 1 个 slot 里的 1 帧报文，而 TSALL 是将所有报文一起传输。TSONE 每传输完毕 1 帧报文都可产生 1 个中断，而 TSALL 需传输完所有已填满的 slot 中的报文后才产生 1 个中断。在 STB 发送数据时，发送顺序可由 TSMODE 位决定，发送模式可选择 FIFO 模式，根据报文被写入 STB 的顺序进行发送；也可以选择根据 ID 优先级顺序发送，ID 标识符越小的报文发送优先级越高。

由于 PTB 的传输优先级高于 STB，若在 STB 传输过程中使能了 TPE 开启 PTB 传输，那么会在 TPE 开启时的那帧报文传输完毕后立即开始 PTB 中的报文传输，STB 的报文传输将被滞后直至 PTB 传输完成或 TPA 终止 PTB 报文传输。

### 20.5.4 自动重传屏蔽

CAN 模块中包含 PTB 与 STB 两种缓冲区的自动重传屏蔽位（CAN\_CFG\_STAT 的 TPSS 与 TSSS 位），且两个屏蔽位控制各自发送缓冲区的自动重传功能。在默认模式下屏蔽位关闭，CAN 会为发生错误或仲裁丢失的报文执行自动重传功能；使能自动重传屏蔽后，对应缓冲区的自动重传功能将被禁止，发生错误与仲裁丢失的报文不会自动重新发送。

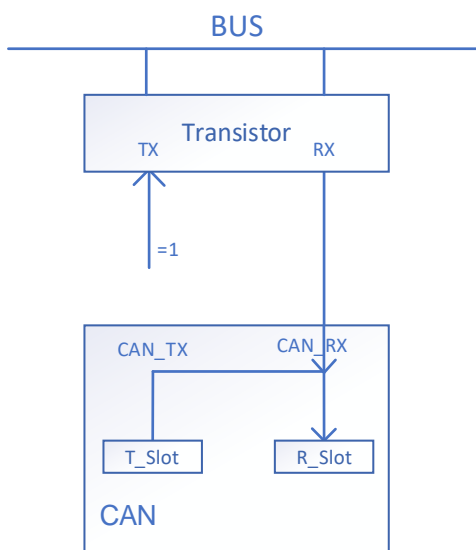
## 20.6 工作模式

CAN 可以选择四种不同的工作模式可供选择，分别是只听模式，外部回环模式，内部回环模式与待机模式，具体模式内容如下：

- 只听模式

当 CAN 运行在只听模式下时，不能向 CAN 总线发送报文，但可以将总线上的报文接收到接收缓存中，不能回复 ACK，也不能置起接收中断。

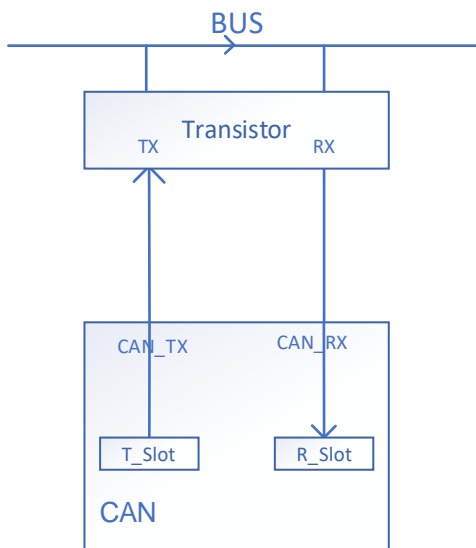
当 CAN 运行在只听模式下并且同时开启外部回环模式时，MCU 可向总线发送报文，但只能对自己发送的报文回复 ACK 并置起中断；其他总线上的报文可被存储在缓存中但是不会对其回复 ACK。



只听模式示意图

- 外部回环模式

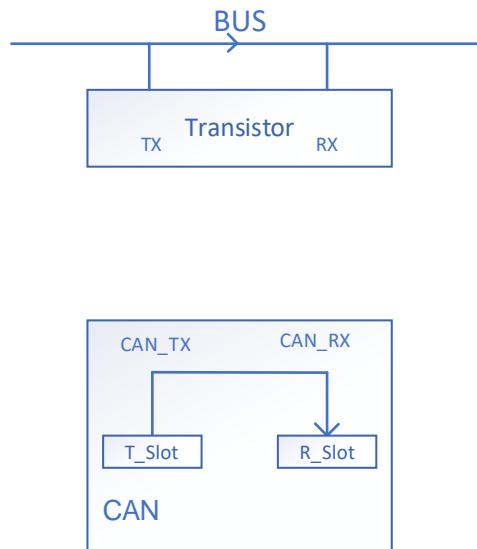
当 CAN 运行在外部回环模式下时，如果 MCU 设备接入总线，那么 MCU 会接收到自己发送的报文和外部报文，且中断正常置起；若 MCU 没有接入总线，那么若将 CAN\_TX 与 CAN\_RX 连接且将 SACK 位写 1，MCU 也会接收到自己发送的报文且中断正常置起。



外部回环模式示意图

- 内部回环模式

当 CAN 运行在内部回环模式下时，MCU 不经过收发器直接在内部将报文发送给接收端，且接收端只会接收到自己发送的报文，不会接收到总线上的报文。



内部回环模式示意图

- 待机模式

当 CAN 运行在待机模式下时，MCU 上电后 CAN 模块不会启用发送报文功能，当 MCU 接收到总线上的报文后，即检测到总线上的显性电平时，无论是否通过过滤器，CAN 都会退出待机模式，之后可正常发送报文。

## 20.7 CAN 通信配置

### 20.7.1 CAN 模块初始化

在启用 MCU 的 CAN 模块相关功能前，需对 CAN 模块进行初始化，初始化步骤包括：

- 使能 AHB 时钟支线上的 CAN 时钟，进行时钟初始化；
- 根据实际功能需要对 IO 进行初始化；
- 对 CAN 结构进行初始化；

由于发送与接收缓存器 TBUF 与 RBUF 都由 RAM 组成，每次上电或复位后 TBUF 与 RBUF 内都为随机数，且接收相关寄存器都是只读状态，无法清零，故用户应避免在 CAN 模块没有接收到任何数据之前读取接收缓存器，否则将会读到无意义随机数。

### 20.7.2 发送报文配置

用户在配置发送报文时需要对以下内容寄存器清零并进行配置：

- 通过配置低速波特率寄存器 CAN\_S\_SEG 与高速波特率寄存器 CAN\_F\_SEG 设置波特率
- 通过配置发送缓存区选择位 TBSEL，选择 PTB 缓存或 STB 缓存
- 通过配置 CAN\_FD 模式位 FDF，选择 CAN2.0 与 CAN\_FD
- 通过配置标识符扩展位 IDE，选择 ID 为标准帧或扩展帧
- 通过配置远程传输请求位 RTR，选择远程帧或扩展帧（若选择 CAN\_FD 模式则可忽略此位）
- 通过配置数据长度位 DLC，设置传输的数据长度
- 通过配置 TBUF 缓存区，设置需要发送的数据

### 20.7.3 发送流程

- ① 开启发送使能后，用户将配置好的报文写入缓冲区，若选择 STB 缓冲区，则需要每次写入 1 帧报文后使能 TSNEXT 表示当前帧已配置完成，并将 TBUF 指向下一个 slot 写入报文。
- ② 需传输的报文写入完毕后，若选择 PTB 则使能 PTE 位开始发送，若选择 STB 则使能 TSONE 逐帧发送或 TSALL 将所有帧一起发送。
- ③ 若使能对应中断，那么发送报文会置起对应中断标志位并进入中断服务函数；若不使能中断，那么对应中断标志位不会置起。

### 20.7.4 接收流程

- ① 开启 CAN\_IDE 寄存器的 TXEN 发送使能与 RXEN 接收使能，开启 TXEN 发送使能的目的是保证接收完毕 ACK 位正确回复。
- ② 设置低速波特率寄存器 CAN\_S\_SEG 和高速波特率寄存器 CAN\_F\_SEG。
- ③ RBUF 有 8 个 slot，最多可存储 8 帧报文。报文的接收可配合过滤器筛选具有特定标识符的报文，没有通过筛选的报文也会进入接收缓存器，但会被下一帧报文覆写，直到通过筛选的报文到来并使 RBUF 指向下一个 slot。
- ④ 读取接收缓冲器 RBUF 中的数据后，需要通过设置 CAN\_CFG\_STAT 寄存器的 RREL 位在读取完成后释放当前 slot，并将 RBUF 指向下一个 slot，才能成功读取下一帧报文。
- ⑤ 若使能对应中断，那么发送报文会置起对应中断标志位并进入中断服务函数；若不使能中断，那么对应中断标志位不会置起。  
若 RBUF 已满后还有新的报文需要存入接收缓冲器，可在初始化步骤设置 ROM 位选择丢弃最新的报文或最旧的报文。

### 20.7.5 报文收发优先级

报文收发优先级与报文标识符有关，ID 标识符越小其优先级越高。当同一时刻不止一个报文请求在总线中等待时，这些报文会根据优先级判断顺序进行处理。

## 20.8 CAN 中断

当 CAN 的中断使能总控制位开启且使能子中断控制位时，若有中断事件来临则会置起对应子中断标志位并进入中断服务函数；若不使能子中断控制位，则无法置起对应子中断标志位。

中断事件	事件标志位	中断使能总控制位	模块中断使能子控制位
接收完成	RIF	CAN_IDE_INTEN	CAN_RTIE->RIE
RBUF 已满	ROIF		CAN_RTIE -> ROIE
RBUF 溢出	RFIF		CAN_RTIE -> RFIE
RBUF slot 达到设定阈值	RAFIF		CAN_RTIE -> RAFIE
PTB 传输完成	TPIF		CAN_RTIE -> TPIE
STB 传输完成	TSIF		CAN_RTIE -> TSIE
错误中断	EIF		CAN_RITE->EIE

中断事件	事件标志位	中断使能总控制位	模块中断使能子控制位
仲裁丢失错误	ALIF		CAN_RTIE -> ALIE
总线错误	BEIF		CAN_RTIE -> BEIE
错误被动中断	EPIF		CAN_RTIE -> EPIE

## 20.9 CAN 寄存器

### 20.9.1 CAN 相关寄存器表

#### 20.9.1.1 接收帧 ID 寄存器 CAN\_RX\_ID——标准帧

寄存器	读/写	说明	复位值	上电初始值
CAN_RX_ID	只读	接收帧 ID 寄存器	0xnnnn_nnnn	0xnnnn_nnnn

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

位编号	位符号	说明
31	ESI	错误状态指示标志 0: CANFD 节点主动错误 1: CANFD 节点被动错误
10~0	ID[10:0]	接收到的标准帧 11 位 ID 标识符
30~11	-	保留

#### 20.9.1.2 接收帧 ID 寄存器 CAN\_RX\_ID——扩展帧

寄存器	读/写	说明	复位值	上电初始值
CAN_RX_ID	只读	接收帧 ID 寄存器	0xnnnn_nnnn	0xnnnn_nnnn

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

位编号	位符号	说明
31	ESI	错误状态指示标志 0: CANFD 节点主动错误 1: CANFD 节点被动错误
28~0	ID[28:0]-extend format	接收到的扩展帧 29 位 ID 标识符
30~29	-	保留

### 20.9.1.3 接收帧控制及状态寄存器 CAN\_RX\_CTRL

寄存器	读/写	说明	复位值	上电初始值
CAN_RX_CTRL	只读	接收帧控制及状态寄存器	0xnxxx_nxxx	0xnxxx_nxxx

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
KOER[2:0]			TX	-	-	-	-
7	6	5	4	3	2	1	0
IDE	RTR	FDL	BRS	DLC[3:0]			

位编号	位符号	说明
15~13	KOER[2:0]	错误代码 000: 无错误 001: 位错误 010: 帧格式错误 011: 填充错误 100: 应答错误 101: CRC 错误 110: 其他错误（自身错误标志后的显性位、收到的主动错误标志时间过长、ACK 错误后被动错误标志期间的显性位）。 111: 未使用 KOER 随每次新错误更新。因此，在成功发送或接收帧时，KOER 将保持不变。
12	TX	回环模式下的接收状态位 0: 未接收 1: 回环模式下，接收到自己发送的帧
7	IDE	标识符扩展名 0: 标准模式 ID[10:0] 1: 扩展模式 ID[28:0]
6	RTR	远程传输请求 0: 数据帧 1: 远程帧
5	FDL	CAN FD 帧标志 0: CAN 2.0 帧 1: CAN FD 帧
4	BRS	位速率切换使能位 0: 禁止位速率切换 1: 速率切换使能

位编号	位符号	说明																														
3~0	DLC[3:0]	接收帧数据字节长度 CAN 2.0 接收帧的数据字节长度最大为 8 bytes，最小为 0 byte； CAN FD 接收帧的数据字节长度最大为 64 bytes，最小为 0 byte。 CAN 2.0 及 CAN FD 的数据字节长度对应关系如下表：																														
		<table border="1"> <thead> <tr> <th>DLC</th> <th>Frame Type</th> <th>Payload in Bytes</th> </tr> </thead> <tbody> <tr> <td>0000 to 1000</td> <td>CAN 2.0 and CAN FD</td> <td>0 to 8</td> </tr> <tr> <td>1001 to 1111</td> <td>CAN 2.0</td> <td>8</td> </tr> <tr> <td>1001</td> <td>CAN FD</td> <td>12</td> </tr> <tr> <td>1010</td> <td>CAN FD</td> <td>16</td> </tr> <tr> <td>1011</td> <td>CAN FD</td> <td>20</td> </tr> <tr> <td>1100</td> <td>CAN FD</td> <td>24</td> </tr> <tr> <td>1101</td> <td>CAN FD</td> <td>32</td> </tr> <tr> <td>1110</td> <td>CAN FD</td> <td>48</td> </tr> <tr> <td>1111</td> <td>CAN FD</td> <td>64</td> </tr> </tbody> </table>	DLC	Frame Type	Payload in Bytes	0000 to 1000	CAN 2.0 and CAN FD	0 to 8	1001 to 1111	CAN 2.0	8	1001	CAN FD	12	1010	CAN FD	16	1011	CAN FD	20	1100	CAN FD	24	1101	CAN FD	32	1110	CAN FD	48	1111	CAN FD	64
		DLC	Frame Type	Payload in Bytes																												
		0000 to 1000	CAN 2.0 and CAN FD	0 to 8																												
		1001 to 1111	CAN 2.0	8																												
		1001	CAN FD	12																												
		1010	CAN FD	16																												
		1011	CAN FD	20																												
		1100	CAN FD	24																												
		1101	CAN FD	32																												
1110	CAN FD	48																														
1111	CAN FD	64																														
31~16	-	保留																														
11~8	-																															

#### 20.9.1.4 接收帧数据寄存器 CAN\_RBUF0~CAN\_RBUF15

寄存器	读/写	说明	复位值	上电初始值
CAN_RBUF0~CAN_RBUF15	只读	接收帧数据寄存器共 64Bytes	0xnnnn_nnnn	0xnnnn_nnnn

31	30	29	28	27	26	25	24
Datax[7:0] (x=N*4+4 N=0~15)							
23	22	21	20	19	18	17	16
Datax[7:0] (x=N*4+3 N=0~15)							
15	14	13	12	11	10	9	8
Datax[7:0] (x=N*4+2 N=0~15)							
7	6	5	4	3	2	1	0
Datax[7:0] (x=N*4+1 N=0~15)							

位编号	位符号	说明
31~24	Datax[7:0] (x=N*4+4 N=0~15)	接收帧数据第 N 个字的第 4 个字节
23~16	Datax[7:0] (x=N*4+3 N=0~15)	接收帧数据第 N 个字的第 3 个字节
15~8	Datax[7:0] (x=N*4+2 N=0~15)	接收帧数据第 N 个字的第 2 个字节
7~0	Datax[7:0] (x=N*4+1 N=0~15)	接收帧数据第 N 个字的第 1 个字节

#### 20.9.1.5 接收帧时间戳存储寄存器低位 CAN\_RTSL

寄存器	读/写	说明	复位值	上电初始值
CAN_RTSL	只读	接收帧时间戳存储寄存器低位	0xnnnn_nnnn	0xnnnn_nnnn

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

RTS[23:16]							
15	14	13	12	11	10	9	8
RTS[15:8]							
7	6	5	4	3	2	1	0
RTS[7:0]							

位编号	位符号	说明
31~0	RTS[31:0]	接收帧时间戳存储单元低 32 位。CAN 模块每接收一帧报文，硬件电路会按照 CiA 603 标准将当前接收帧时间戳计时器值写入 RTS 寄存器内，时间戳获取位置由 CAN_ACFCTRL 的 TIMEPOS 寄存器控制。时间戳可以是 32 位或 64 位，未使用的位强制为 0。 CAN 模块提供了 8 个接收帧 slot，每个 slot 均有独立的 RTS 存储空间

### 20.9.1.6 接收帧时间戳存储寄存器高位 CAN\_RTSH

寄存器	读/写	说明	复位值	上电初始值
CAN_RTSH	只读	接收帧时间戳存储寄存器高位	0xnnnn_nnnn	0xnnnn_nnnn

31	30	29	28	27	26	25	24
RTS[63:56]							
23	22	21	20	19	18	17	16
RTS[55:48]							
15	14	13	12	11	10	9	8
RTS[47:40]							
7	6	5	4	3	2	1	0
RTS[39:32]							

位编号	位符号	说明
31~0	RTS[63:32]	接收帧时间戳存储单元高 32 位。CAN 模块每接收一帧报文，硬件电路会按照 CiA 603 标准将当前接收帧时间戳计时器值写入 RTS 寄存器内，时间戳获取位置由 CAN_ACFCTRL 的 TIMEPOS 寄存器控制。时间戳可以是 32 位或 64 位，未使用的位强制为 0。 CAN 模块提供了 8 个接收帧 slot，每个 slot 均有独立的 RTS 存储空间

### 20.9.1.7 发送帧 ID 寄存器 CAN\_TX\_ID——标准帧

寄存器	读/写	说明	复位值	上电初始值
CAN_TX_ID	读/写	发送帧 ID 寄存器	0xnnnn_nnnn	0xnnnn_nnnn

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

位编号	位符号	说明
31	TTSSEN	发送帧时间戳信息生成使能位

位编号	位符号	说明
		使能后硬件电路将按照 CiA 603 标准记录当前发送帧的时间戳 (TTS) 信息 0: 此帧不获取发送时间戳 1: TTS 更新使能
10~0	ID[10:0]	发送的标准帧 11 位 ID 标识符
30~11	-	保留

### 20.9.1.8 发送帧 ID 寄存器 CAN\_TX\_ID——扩展帧

寄存器	读/写	说明	复位值	上电初始值
CAN_TX_ID	读/写	发送帧 ID 寄存器	0xnxxx_nxxx	0xnxxx_nxxx

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

位编号	位符号	说明
31	TTSEN	发送帧时间戳信息生成使能位 使能后硬件电路将按照 CiA 603 标准记录当前发送帧的时间戳 (TTS) 信息 0: 此帧不获取发送时间戳 1: TTS 更新使能
28~0	ID[28:0]	发送的标准帧 29 位 ID 标识符
30~29	-	保留

### 20.9.1.9 发送帧控制及状态寄存器 CAN\_TX\_CTRL

寄存器	读/写	说明	复位值	上电初始值
CAN_TX_CTRL	读/写	发送帧控制及状态寄存器	0xnxxx_nxxx	0xnxxx_nxxx

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

位编号	位符号	说明
7	IDE	标识符扩展名 0: 标准格式: ID (10:0) 1: 扩展格式: ID (28:0)
6	RTR	远程传输请求 0: 数据帧

位编号	位符号	说明																														
		1: 远程帧 只有 CAN 2.0 帧可以是远程帧。CAN FD 没有远程帧。																														
5	FDF	CAN 2.0 / CAN FD 帧模式位 0: CAN 2.0 帧 (数据域最多为 8 bytes) 1: CAN FD 帧 (数据域最多为 64 bytes)																														
4	BRS	位速率切换使能位: 0: 禁止位速率切换, 数据段与控制段速率一致, 统一为常规或慢速通信模式 1: 速率切换使能, 每一帧的数据段 (包含数据域和 CRC 字段) 与控制段速率不同, 数据段会使用更高的速率																														
3~0	DLC[3:0]	发送帧数据域长度位 CAN 2.0 发送帧的数据域长度最大为 8 bytes, 最小为 0 byte; CAN FD 发送帧的数据域长度最大为 64 bytes, 最小为 0 byte。 CAN 2.0 及 CAN FD 的数据域长度对应关系如下表: <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>DLC</th> <th>Frame Type</th> <th>Payload in Bytes</th> </tr> </thead> <tbody> <tr> <td>0000 to 1000</td> <td>CAN 2.0 and CAN FD</td> <td>0 to 8</td> </tr> <tr> <td>1001 to 1111</td> <td>CAN 2.0</td> <td>8</td> </tr> <tr> <td>1001</td> <td>CAN FD</td> <td>12</td> </tr> <tr> <td>1010</td> <td>CAN FD</td> <td>16</td> </tr> <tr> <td>1011</td> <td>CAN FD</td> <td>20</td> </tr> <tr> <td>1100</td> <td>CAN FD</td> <td>24</td> </tr> <tr> <td>1101</td> <td>CAN FD</td> <td>32</td> </tr> <tr> <td>1110</td> <td>CAN FD</td> <td>48</td> </tr> <tr> <td>1111</td> <td>CAN FD</td> <td>64</td> </tr> </tbody> </table>	DLC	Frame Type	Payload in Bytes	0000 to 1000	CAN 2.0 and CAN FD	0 to 8	1001 to 1111	CAN 2.0	8	1001	CAN FD	12	1010	CAN FD	16	1011	CAN FD	20	1100	CAN FD	24	1101	CAN FD	32	1110	CAN FD	48	1111	CAN FD	64
DLC	Frame Type	Payload in Bytes																														
0000 to 1000	CAN 2.0 and CAN FD	0 to 8																														
1001 to 1111	CAN 2.0	8																														
1001	CAN FD	12																														
1010	CAN FD	16																														
1011	CAN FD	20																														
1100	CAN FD	24																														
1101	CAN FD	32																														
1110	CAN FD	48																														
1111	CAN FD	64																														
31~8	-	保留																														

#### 20.9.1.10 发送帧数据寄存器 CAN\_TBUF0~CAN\_TBUF15

寄存器	读/写	说明	复位值	上电初始值
CAN_TBUF0~CAN_TBUF15	读/写	发送帧数据寄存器共 64Bytes	0xnnnn_nnnn	0xnnnn_nnnn

31	30	29	28	27	26	25	24
Datax[7:0] (x=N*4+4 N=0~15)							
23	22	21	20	19	18	17	16
Datax[7:0] (x=N*4+3 N=0~15)							
15	14	13	12	11	10	9	8
Datax[7:0] (x=N*4+2 N=0~15)							
7	6	5	4	3	2	1	0
Datax[7:0] (x=N*4+1 N=0~15)							

位编号	位符号	说明
31~24	Datax[7:0] (x=N*4+4 N=0~15)	发送帧数据第 N 个字的第 4 个字节
23~16	Datax[7:0] (x=N*4+3 N=0~15)	发送帧数据第 N 个字的第 3 个字节
15~8	Datax[7:0] (x=N*4+2 N=0~15)	发送帧数据第 N 个字的第 2 个字节
7~0	Datax[7:0] (x=N*4+1 N=0~15)	发送帧数据第 N 个字的第 1 个字节

### 20.9.1.11 发送帧时间戳存储寄存器低位 CAN\_TTSL

寄存器	读/写	说明	复位值	上电初始值
CAN_TTSL	只读	发送帧时间戳存储寄存器低位	0xnnnn_nnnn	0xnnnn_nnnn

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

位编号	位符号	说明
31~0	TTS[31:0]	发送帧时间戳存储单元低 32 位。发送帧时间戳存储单元用于存储发送帧的时间戳（TTS），时间主机（time master）可通过 TTS 获取同步信息（SYNC message）。 当发送帧时间戳信息生成位使能（TTSEN=1），则每发送一帧，TTS 都会更新。时间戳可以是 32 位或 64 位，未使用的位强制为 0。

### 20.9.1.12 发送帧时间戳存储寄存器高位 CAN\_TTSH

寄存器	读/写	说明	复位值	上电初始值
CAN_TTSH	只读	发送帧时间戳存储寄存器高位	0xnnnn_nnnn	0xnnnn_nnnn

31	30	29	28	27	26	25	24
TTS[63:56]							
23	22	21	20	19	18	17	16
TTS[55:48]							
15	14	13	12	11	10	9	8
TTS[47:40]							
7	6	5	4	3	2	1	0
TTS[39:32]							

位编号	位符号	说明
31~0	TTS[63:32]	发送帧时间戳存储单元高 32 位。发送帧时间戳存储单元用于存储发送帧的时间戳（TTS），时间主机（time master）可通过 TTS 获取同步信息（SYNC message）。 当发送帧时间戳信息生成位使能（TTSEN=1），则每发送一帧，TTS 都会更新。时间戳可以是 32 位或 64 位，未使用的位强制为 0。

### 20.9.1.13 CAN 状态位与配置寄存器 CAN\_CFG\_STAT

寄存器	读/写	说明	复位值	上电初始值
CAN_CFG_STAT	读/写	CAN 状态位与配置寄存器	0x0090_0080	0xnnnn_nnnn

31	30	29	28	27	26	25	24
SACK	ROM	ROV	RREL	RBALL	-	RSTAT[1:0]	

23	22	21	20	19	18	17	16
FD_ISO	TSNEXT	TSMODE	-	-	-	TSSTAT[1:0]	
15	14	13	12	11	10	9	8
TBSEL	LOM	STBY	TPE	TPA	TSONE	TSALL	TSA
7	6	5	4	3	2	1	0
RESET	LBME	LBMI	TPSS	TSSS	RACTIVE	TACTIVE	BUSOFF

位编号	位符号	说明
31	SACK	自应答使能位 0: 禁止自应答 1: 回环模式下 (LBME=1) 时要求自应答 该位仅在 RESET=0 时可以操作
30	ROM	接收缓冲区 (R_slot) 溢出模式选择位 当 8 个接收 R_slot 已满, 新的报文到来后有两种处理方式可选: 0: 最早的报文将被丢弃 1: 丢弃新接收的报文
29	ROV	接收缓冲区 (R_slot) 溢出标志 (只读) 0: 无溢出 1: R_slot 溢出 该位在 RREL 写 1 后清 0
28	RREL	接收缓冲区 (R_slot) 释放控制位 每次读取 RBUF, 需要对此位写 1 以释放当前读取的 R_slot, 并指向指向下一个 R_slot, RSTAT (1:0) 也会有相应的更新。 0: 当前 R_slot 空间未释放 1: 当前 R_slot 内容已被读取, 当前 RBUF 空间已释放
27	RBALL	接收缓冲区 (R_slot) 存储模式设置位 0: 常规模式: 数据帧判断为正确后才会存储到 R_slot 1: 数据帧无论正确与否, 均存储在 R_slot
25~24	RSTAT[1:0]	接收缓冲区状态 00: R_slot 为空 01: R_slot 非空, 未满足且无溢出, 存储帧数量 < AFWL 编程阈值 10: R_slot 未满足且无溢出, 但存储帧数量 ≥ AFWL 编程阈值 11: R_slot 已满, 如果此时 ROV=1, 代表 RBUF 已溢出
23	FD_ISO	CAN FD ISO 模式 0: Bosch CAN FD (非 ISO) 模式 1: ISO CAN FD 模式 (ISO 11898-1:2015)
22	TSNEXT	次级发送缓冲区 (STB slot) 切换控制位 每次 TBUF 填写完成后, 需要对此位写 1 以指示当前 STB 已被填满, 并将 TBUF 指向下一个 STB, TSSTAT (1:0) 也会有相应的更新。 CAN 模块复位后, 该位会自动清零。 0: 无操作 1: 当前 STB slot 填写完成, 指向下一个 slot
21	TSMODE	次级发送缓冲区发送模式选择位 0: FIFO 模式 1: 优先级决定模式
17~16	TSSTAT[1:0]	次级传输状态位 该位代表 STB 存储的报文数量: 00: STB 存储的报文数量为空 01: STB 存储的报文数量 ≤ 4 10: STB 存储的报文数量 ≥ 5 11: STB 存储的报文数量 = 8 STB 被禁用时 TSSTAT 固定为 00
15	TBSEL	发送缓冲区选择 选择要载入信息的发送缓冲区。使用 TBUF 寄存器进行访问。

位编号	位符号	说明
		0: 当前报文写入区域为 PTB (高优先级的缓冲器) 1: 当前报文写入区域为 STB
14	LOM	只听模式使能位 0: 禁用 1: 启用 该位仅在 RESET=0 时可以操作
13	STBY	收发器待机模式 0: 禁用 1: 启用 该位仅在 RESET=0 时可以操作
12	TPE	PTB (高优先级) 传输模式使能位 0: 不启用 PTB 1: 对 PTB 信息进行传输
11	TPA	PTB 传输中止 0: 不中止 1: 中止已由 TPE=1 请求但尚未开始的 PTB 传输。(数据信息仍留在 PTB 中)。该位使能后, TPE 将被关联清 0
10	TSONE	STB 传输一帧报文使能位 0: STB 不传输 1: 启用 STB 中一帧报文的传输
9	TSALL	STB (次优先级) 传输所有报文帧使能位 0: STB 不传输 1: STB 中所有报文的传输启用
8	TSA	STB 传输中止 0: 不中止 1: 中止 STB 已请求但尚未开始的传输
7	RESET	复位请求位 0: 不复位 1: 复位 部分控制位只有在 RESET=1 时才允许修改, 部分控制位只有在 RESET=0 时才允许修改 (例如 LBME 等模式相关控制位)
6	LBME	回环模式 0: 禁用 1: 启用 当传输处于活动状态时, 不应该启用 LBME 该位仅在 RESET=0 时可以操作
5	LBMI	内部回环模式 0: 禁用 1: 已启用 当传输处于活动状态时, 不应该启用 LBMI 该位仅在 RESET=0 时可以操作
4	TPSS	PTB 自动重传 (single shot) 屏蔽位 0: PTB 为自动重传模式, 发生错误或仲裁丢失时会自动执行重传。 1: PTB 为禁止自动重传模式, 发生错误或仲裁丢失时不会执行重传。
3	TSSS	STB 自动重传 (single shot) 屏蔽位 0: STB 为自动重传模式, 发生错误或仲裁丢失时会自动执行重传。 1: STB 为禁止自动重传模式, 发生错误或仲裁丢失时不会执行重传。
2	RACTIVE	接收状态位 0: 当前无接收活动 1: 正在接收帧
1	TACTIVE	发送状态位 0: 当前无发送活动

位编号	位符号	说明
		1: 正在发送帧
0	BUSOFF	CAN 模块总线关闭控制 0: 总线开启 1: 总线关闭 该位写 1 后将重置收发错误计数器 RECNT 和 TECNT，建议用户仅在调试阶段操作此位
26 20~18	-	保留

#### 20.9.1.14 CAN 中断控制和中断标志位寄存器 CAN\_RTIE

寄存器	读/写	说明	复位值	上电初始值
CAN_RTIE	读/写	CAN 中断控制	0x1B00_00FE	0xnxxx_nxxx

31	30	29	28	27	26	25	24
AFWL[3:0]				EWL[3:0]			
23	22	21	20	19	18	17	16
EWARN	EPASS	EPIE	EPIF	ALIE	ALIF	BEIE	BEIF
15	14	13	12	11	10	9	8
RIF	ROIF	RFIF	RAFIF	TPIF	TSIF	EIF	AIF
7	6	5	4	3	2	1	0
RIE	ROIE	RFIE	RAFIE	TPIE	TSIE	EIE	TSFF

位编号	位符号	说明
31~28	AFWL[3:0]	接收缓存阈值 当已填充完成的 RB slots 数量与 AFWL (3:0) 设置值相等，RB slots 阈值警告标志位 RAFIF 会置起，如果此时 RAFIE=1，将触发中断。
27~24	EWL[3:0]	错误计数阈值=(EWL[3:0]+1)*8，可设置的范围为 8~128 RECNT 和 TECNT 对收发期间的错误进行计数，这两种错误计数器任意一个达到错误计数阈值，EWARN 和 EIF 都会置起，如果此时 EIE=1，将触发中断。
23	EWARN	错误计数阈值警告标志位 0: RECNT 和 TECNT 两种错误计数器的值均小于错误计数阈值 (EWL[3:0]+1)*8 1: RECNT 和 TECNT 两种错误计数器任意一个达到错误计数阈值 (EWL[3:0]+1)*8
22	EPASS	错误被动模式启动位 0: 不启动（节点为主动错误）。 1: 启动（节点为被动错误）
21	EPIE	错误被动中断使能位 0: 禁用 1: 启用
20	EPIF	错误被动中断标志。如果错误状态从主动错误变为被动错误，或者相反，如果该中断被使能，EPIF 将被置起。
19	ALIE	仲裁丢失中断使能 0: 禁用 1: 启用
18	ALIF	仲裁丢失中断标志
17	BEIE	总线错误中断使能 0: 禁用 1: 启用

位编号	位符号	说明
16	BEIF	总线错误中断标志
15	RIF	接收中断标志 0: 未收到帧 1: 数据或远程帧已接收完成, 可从接收缓存读取
14	ROIF	RB 溢出标志位 0: RB slots 未发生溢出覆盖 1: RB slots 已满且至少有一条接收信息被覆盖。 接收溢出发生时, ROIF 和 RFIF 都会被置起, 如果此时 ROIE=1, 将触发中断
13	RFIF	RB 已满标志位 0: RB slots 未滿 1: RB slots 已滿。在下一条接收信息到来前请及时是否 RB slots, 否则 RB slots 中最早的信息将被丢弃。
12	RAFIF	RB slots 阈值警告标志位 0: 已填充的 RB slot < AFWL 1: 已填满的 RB slot ≥ AFWL
11	TPIF	PTB 传输完成标志位 0: PTB 传输未完成 1: PTB 传输请求已响应, 传输完成
10	TSIF	STB 传输完成标志位 0: STB 传输未完成 1: STB 传输请求已响应, 传输完成
9	EIF	错误中断标志位 当满足以下任一个条件, 该位都会置起: a) RECNT 错误计数值达到错误计数阈值(EWL[3:0]+1)*8, b) TECNT 错误计数值达到错误计数阈值(EWL[3:0]+1)*8 c) BUSOFF 位被改写
8	AIF	传输中止标志位 0: 未执行中止操作 1: TPA 或 TSA 写 1 后, 该位将置起, 相关的发送请求被中止
7	RIE	接收中断使能 0: 禁用 1: 启用
6	ROIE	RB 溢出中断使能 0: 禁用 1: 启用
5	RFIE	RB 已满中断使能 0: 禁用 1: 启用
4	RAFIE	RB slots 阈值警告中断使能 0: 禁用 1: 启用
3	TPIE	PTB 传输完成中断使能 0: 禁用 1: 启用
2	TSIE	STB 传输完成中断使能 0: 禁用 1: 启用
1	EIE	错误中断使能 0: 禁用 1: 启用
0	TSFF	发送区域满标志位:

位编号	位符号	说明
		次级发送缓冲区满标志位 0: STB 未装满最大数量的报文 1: STB 已装满最大数量的报文

### 20.9.1.15 低速时钟配置寄存器 CAN\_S\_SEG

寄存器	读/写	说明	复位值	上电初始值
CAN_S_SEG	读/写	低速时钟配置寄存器	0x0102_0203	0xnxxx_nxxx

31	30	29	28	27	26	25	24
S_PRESC[7:0]							
23	22	21	20	19	18	17	16
S_SJW[6:0]							
15	14	13	12	11	10	9	8
S_Seg_2[6:0]							
7	6	5	4	3	2	1	0
S_Seg_1[7:0]							

位编号	位符号	说明
31~24	S_PRESC[7:0]	低速预分频器 预分频器对系统时钟进行分频，以获得时间量子时钟 tq_clk。 有效范围 PRESC=[0x00, 0xff]，分频值为 1 至 256 分频值= S_PRESC+1
22~16	S_SJW[6:0]	低速同步跳变宽度 $t_{SJW} = (SJW+1)*TQ$ 同步跳变宽度是缩短或延长重新同步比特时间的最长时间，其中 TQ 是时间量子
14~8	S_Seg_2[6:0]	低速相位段 2 相位段 2 用于配置采样点后的时间段，采样点将设置为 $t_{Seg\_2} = (Seg\_2+1)*TQ$
7~0	S_Seg_1[7:0]	低速相位段 1 相位段 1 用于配置采样点前的时间段减去同步段的时间，采样点将设置为 $t_{Seg\_1} = (Seg\_1+2)*TQ$
23 15	-	保留

### 20.9.1.16 高速时钟配置寄存器 CAN\_F\_SEG

寄存器	读/写	说明	复位值	上电初始值
CAN_F_SEG	读/写	高速时钟配置寄存器	0x0102_0203	0xnxxx_nxxx

31	30	29	28	27	26	25	24
F_PRESC[7:0]							
23	22	21	20	19	18	17	16
F_SJW[3:0]							
15	14	13	12	11	10	9	8
F_Seg_2[3:0]							
7	6	5	4	3	2	1	0
F_Seg_1[4:0]							

位编号	位符号	说明
31~24	F_PRESC[7:0]	高速预分配器 预分频器对系统时钟进行分频，以获得时间量子时钟 tq_clk。 有效范围 PRESC=[0x00, 0xff]，分频值为 1 至 256 分频值 = F_PRESC+1
19~16	F_SJW[3:0]	高速同步跳变宽度 $t_{SJW} = (SJW+1)*TQ$ 同步跳变宽度是缩短或延长重新同步比特时间的最长时间，其中 TQ 是时间量子
11~8	F_Seg_2[3:0]	高速相位段 2 相位段 2 用于配置采样点后的时间段，采样点将设置为 $t_{Seg\_2} = (Seg\_2+1)*TQ$
4~0	F_Seg_1[4:0]	高速相位段 1 相位段 1 用于配置采样点前的时间段减去同步段的时间，采样点将设置为 $t_{Seg\_1} = (Seg\_1+2)*TQ$
23~20 15~12 7~5	-	保留

#### 20.9.1.17 延迟补偿与收发错误寄存器 CAN\_EALCAP

寄存器	读/写	说明	复位值	上电初始值
CAN_EALCAP	读/写	延迟补偿与收发错误寄存器	0x0000_0000	0xnxxxx_nnnn

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

位编号	位符号	说明
31~24	TECNT[7:0]	发送错误计数器，代表传输过程中的发送错误次数
23~16	RECNT[7:0]	接收错误计数器，代表接收过程中的错误次数
15	TDCEN	发送延迟补偿使能位 0: 禁用发送器延迟补偿 1: 启用发送器延迟补偿
14~8	SSPOFF	二次采样点偏移
7~5	KOER[2:0]	错误类型 错误类型（错误代码） 000: 无错误 001: 位错误 010: 格式错误 011: 填充错误 100: 应答错误 101: CRC 错误 110: 其他错误（自身错误标志后的显性位、收到的主动错误标志时间过长、ACK 错误后被动错误标志期间的显性位） 111: 未使用 KOER 随每次新错误更新。因此，在成功发送或接收帧

位编号	位符号	说明
		时, KOER 将保持不变
4~0	ALC[4:0]	仲裁丢失捕获 (仲裁丢失所在帧的位位置) ALC 的值定义如下: 帧以 SOF 位开始, 然后传输 ID 的第一位。第一个 ID 位的 ALC 值为 0, 第二个 ID 位的 ALC 值为 1, 以此类推。

### 20.9.1.18 过滤器控制寄存器 CAN\_ACFCTRL

寄存器	读/写	说明	复位值	上电初始值
CAN_ACFCTRL	读/写	过滤器控制寄存器	0x00010200	0xnxxx_nxxx

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
AE_7	AE_6	AE_5	AE_4	AE_3	AE_2	AE_1	AE_0
15	14	13	12	11	10	9	8
						TIMEPOS	TIMEEN
7	6	5	4	3	2	1	0
-	-	SELMASK	-	ACFADR[3:0]			

位编号	位符号	说明
23~16	AE_X X= (0~7)	X 号接收过滤器使能 0: 禁用接收过滤器 1: 使能接收过滤器
9	TIMEPOS	时间戳位置 0: SOF 1: EOF 只有在 TIMEEN=0 时才能修改 TIMEPOS, 但可以通过设置 TIMEEN=1 的获取写入访问修改 TIMEPOS。
8	TIMEEN	时间戳启用 0: 禁用 1: 启用
5	SELMASK	选择接收 MASK 0: 将 ACF_x 指向接收 code 1: 将 ACF_x 指向接收 mask
3~0	ACFADR[3:0]	接收过滤器地址 ACFADR 指向特定的接收过滤器。所选滤波器可通过寄存器 ACF_x 访问。位 SELMASK 用于在接收代码和屏蔽之间选择所选接收滤波器。
31~24 15~10 7~6 4	-	保留

### 20.9.1.19 过滤器数据寄存器 CAN\_ACF

寄存器	读/写	说明	复位值	上电初始值
CAN_ACF	读/写	过滤器数据寄存器	0x0000_0000	0xnxxx_nxxx

31	30	29	28	27	26	25	24

-	AIDEE	AIDE	AMASK/ACODE[28:24]				
23	22	21	20	19	18	17	16
AMASK/ACODE[23:16]							
15	14	13	12	11	10	9	8
AMASK/ACODE[15:8]							
7	6	5	4	3	2	1	0
AMASK/ACODE[7:0]							

位编号	位符号	说明
30	AIDEE	接收掩码 IDE 位检查使能位 0: 接收过滤器不受 AIDE 限制, 标准帧或扩展帧均允许接收 1: 接收过滤器根据 AIDE 定义选择仅接收标准帧或仅接收扩展帧
29	AIDE	接收掩码 IDE 位值, 该位仅在 AIDEE=1 时有效 0: 接收过滤器只接收标准帧 1: 接收过滤器只接收扩展帧
28~0	AMASK /ACODE[28:0]	接收 MASK 0: 使能对接收标识符的这些位进行接收检查 1: 禁用对接收标识符这些位的接收检查 接收 CODE 0: 与接收报文 ID 位比较的 ACC 位值 1: 与接收报文 ID 位比较的 ACC 位值
31	-	保留

### 20.9.1.20 CAN 中断及控制使能寄存器 CAN\_IDE

寄存器	读/写	说明	复位值	上电初始值
CAN_IDE	读/写	CAN 中断及控制使能寄存器	0x8000_0000	0xnxxx_nxxx

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

位编号	位符号	说明
31	FDEN	FD 帧格式使能位 0: 关闭使能 1: 打开使能
15	TXEN	CAN 发送使能位 0: 关闭使能 1: 打开使能
14	RXEN	CAN 接收使能位 0: 关闭使能 1: 打开使能
11	TIM_EN	时间戳计数器使能位 0: 关闭使能 1: 打开使能
0	INTEN	中断请求 CPU 屏蔽位

位编号	位符号	说明
		0: 中断请求屏蔽 1: 中断请求开启
30~16 13~12 10~1	-	保留

### 20.9.1.21 CAN 时间戳定时器低位寄存器 CAN\_TIML

寄存器	读/写	说明	复位值	上电初始值
CAN_TIML	读/写	CAN 时间戳定时器低位读写寄存器	0x0000_0000	0xnxxxx_nnnn

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

位编号	位符号	说明
31~0	CAN_TIML[31:0]	64 位时间戳定时器低 32 位

### 20.9.1.22 CAN 时间戳定时器高位寄存器 CAN\_TIMH

寄存器	读/写	说明	复位值	上电初始值
CAN_TIMH	读/写	CAN 时间戳定时器高位读写寄存器	0x0000_0000	0xnxxxx_nnnn

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

位编号	位符号	说明
31~0	CAN_TIMH[31:0]	64 位时间戳定时器高 32 位

## 20.9.2 CAN 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值
CAN 基地址: 0x4001_0C00					
CAN_RX_ID	0x00	只读	接收帧 ID 寄存器	0xnxxxx_nnnn	0xnxxxx_nnnn
CAN_RX_CTRL	0x04	只读	接收帧控制及状态寄存器	0xnxxxx_nnnn	0xnxxxx_nnnn

寄存器	偏移地址	读/写	说明	复位值	上电初始值
CAN_RBUF0	0x08	只读	接收帧数据寄存器 0	0xnnnn_nnnn	0xnnnn_nnnn
CAN_RBUF1	0x0C	只读	接收帧数据寄存器 1	0xnnnn_nnnn	0xnnnn_nnnn
CAN_RBUF2	0x10	只读	接收帧数据寄存器 2	0xnnnn_nnnn	0xnnnn_nnnn
CAN_RBUF3	0x14	只读	接收帧数据寄存器 3	0xnnnn_nnnn	0xnnnn_nnnn
CAN_RBUF4	0x18	只读	接收帧数据寄存器 4	0xnnnn_nnnn	0xnnnn_nnnn
CAN_RBUF5	0x1C	只读	接收帧数据寄存器 5	0xnnnn_nnnn	0xnnnn_nnnn
CAN_RBUF6	0x20	只读	接收帧数据寄存器 6	0xnnnn_nnnn	0xnnnn_nnnn
CAN_RBUF7	0x24	只读	接收帧数据寄存器 7	0xnnnn_nnnn	0xnnnn_nnnn
CAN_RBUF8	0x28	只读	接收帧数据寄存器 8	0xnnnn_nnnn	0xnnnn_nnnn
CAN_RBUF9	0x2C	只读	接收帧数据寄存器 9	0xnnnn_nnnn	0xnnnn_nnnn
CAN_RBUF10	0x30	只读	接收帧数据寄存器 10	0xnnnn_nnnn	0xnnnn_nnnn
CAN_RBUF11	0x34	只读	接收帧数据寄存器 11	0xnnnn_nnnn	0xnnnn_nnnn
CAN_RBUF12	0x38	只读	接收帧数据寄存器 12	0xnnnn_nnnn	0xnnnn_nnnn
CAN_RBUF13	0x3C	只读	接收帧数据寄存器 13	0xnnnn_nnnn	0xnnnn_nnnn
CAN_RBUF14	0x40	只读	接收帧数据寄存器 14	0xnnnn_nnnn	0xnnnn_nnnn
CAN_RBUF15	0x44	只读	接收帧数据寄存器 15	0xnnnn_nnnn	0xnnnn_nnnn
CAN_RTSL	0x48	只读	接收帧时间戳存储寄存器低位	0xnnnn_nnnn	0xnnnn_nnnn
CAN_RTSH	0x4C	只读	接收帧时间戳存储寄存器高位	0xnnnn_nnnn	0xnnnn_nnnn
CAN_TX_ID	0x50	读/写	发送帧 ID 寄存器	0xnnnn_nnnn	0xnnnn_nnnn
CAN_TX_CTRL	0x54	读/写	发送帧控制及状态寄存器	0xnnnn_nnnn	0xnnnn_nnnn
CAN_TBUF0	0x58	读/写	发送帧数据寄存器 0	0xnnnn_nnnn	0xnnnn_nnnn
CAN_TBUF1	0x5C	读/写	发送帧数据寄存器 1	0xnnnn_nnnn	0xnnnn_nnnn
CAN_TBUF2	0x60	读/写	发送帧数据寄存器 2	0xnnnn_nnnn	0xnnnn_nnnn
CAN_TBUF3	0x64	读/写	发送帧数据寄存器 3	0xnnnn_nnnn	0xnnnn_nnnn
CAN_TBUF4	0x68	读/写	发送帧数据寄存器 4	0xnnnn_nnnn	0xnnnn_nnnn
CAN_TBUF5	0x6C	读/写	发送帧数据寄存器 5	0xnnnn_nnnn	0xnnnn_nnnn
CAN_TBUF6	0x70	读/写	发送帧数据寄存器 6	0xnnnn_nnnn	0xnnnn_nnnn

寄存器	偏移地址	读/写	说明	复位值	上电初始值
CAN_TBUF7	0x74	读/写	发送帧数据寄存器 7	0xnnnn_nnnn	0xnnnn_nnnn
CAN_TBUF8	0x78	读/写	发送帧数据寄存器 8	0xnnnn_nnnn	0xnnnn_nnnn
CAN_TBUF9	0x7C	读/写	发送帧数据寄存器 9	0xnnnn_nnnn	0xnnnn_nnnn
CAN_TBUF10	0x80	读/写	发送帧数据寄存器 10	0xnnnn_nnnn	0xnnnn_nnnn
CAN_TBUF11	0x84	读/写	发送帧数据寄存器 11	0xnnnn_nnnn	0xnnnn_nnnn
CAN_TBUF12	0x88	读/写	发送帧数据寄存器 12	0xnnnn_nnnn	0xnnnn_nnnn
CAN_TBUF13	0x8C	读/写	发送帧数据寄存器 13	0xnnnn_nnnn	0xnnnn_nnnn
CAN_TBUF14	0x90	读/写	发送帧数据寄存器 14	0xnnnn_nnnn	0xnnnn_nnnn
CAN_TBUF15	0x94	读/写	发送帧数据寄存器 15	0xnnnn_nnnn	0xnnnn_nnnn
CAN_TTSL	0x98	只读	发送帧时间戳存储寄存器低位	0xnnnn_nnnn	0xnnnn_nnnn
CAN_TTSH	0x9C	只读	发送帧时间戳存储寄存器高位	0xnnnn_nnnn	0xnnnn_nnnn
CAN_CFG_STAT	0xA0	读/写	CAN 状态位与配置寄存器	0x0090_0080	0xnnnn_nnnn
CAN_RTIE	0xA4	读/写	CAN 中断控制和中断标志位寄存器	0x1B00_00FE	0xnnnn_nnnn
CAN_S_SEG	0xA8	读/写	低速时钟配置寄存器	0x0102_0203	0xnnnn_nnnn
CAN_F_SEG	0xAC	读/写	高速时钟配置寄存器	0x0102_0203	0xnnnn_nnnn
CAN_EALCAP	0xB0	读/写	延迟补偿与收发错误寄存器	0x0000_0000	0xnnnn_nnnn
CAN_ACFCTRL	0xB4	读/写	过滤器控制寄存器	0x0001_0200	0xnnnn_nnnn
CAN_ACF	0xB8	读/写	过滤器数据寄存器	0x0000_0000	0xnnnn_nnnn
CAN_IDE	0xCC	读/写	CAN 中断及控制使能寄存器	0x8000_0000	0xnnnn_nnnn
CAN_TIML	0xD0	读/写	CAN 时间戳定时器低位寄存器	0x0000_0000	0xnnnn_nnnn
CAN_TIMH	0xD4	读/写	CAN 时间戳定时器高位寄存器	0x0000_0000	0xnnnn_nnnn

## 21 硬件看门狗 WDT

### 21.1 概述

SC32F15G 系列内建一个独立的硬件看门狗 WDT，其时钟源为内部的 32kHz 振荡器 LIRC。用户可以通过编程器的 Customer Option 中的 ENWDT 控制位选择是否开启看门狗复位功能。

硬件看门狗 WDT，具有安全性高、定时准确及使用灵活的优点。此看门狗外设可检测并解决由软件错误导致的故障，并在计数器达到给定的溢出时间时触发系统复位。

WDT 由其内部低频振荡器驱动，因此即便在主时钟发生故障时仍然保持工作状态。

### 21.2 时钟源

SC32F15G 系列的 WDT 的时钟源固定为 LIRC。WDT 使能后，LIRC 会自动开启，WDT 工作的过程中 LIRC 始终保持振荡，用户无法关闭。

### 21.3 WDT 寄存器

#### 21.3.1 WDT 相关寄存器表

##### 21.3.1.1 WDT 控制寄存器 WDTCON

寄存器	读/写	说明	复位值	上电初始值
WDTCON	读/写	WDT 控制寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
0	CLRWDT	WDT 计数清零位 该位通过软件写 1，由硬件自动清 0。 0: 无影响 1: WDT 计数器从 0 开始计数
31~1	-	保留

### 21.3.1.2 WDT 设置寄存器 WDTCFG

寄存器	读/写	说明	复位值	上电初始值
WDTCFG	读/写	WDT 设置寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明	
2~0	WDTCKS[2:0]	看门狗时钟选择:	
		WDTCKS[2:0]	WDT 溢出时间
		000	500ms
		001	250ms
		010	125ms
		011	62.5ms
		100	31.5ms
		101	15.75ms
		110	7.88ms
		111	3.94ms
31~3	-	保留	

### 21.3.2 WDT 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值	访问限制
WDT 基地址: 0x4000_0330						
WDTCON	0x0C	读/写	WDT 控制寄存器	0x0000_0000	0x0000_0000	不支持字节/半字访问
WDTCFG	0x10	读/写	WDT 设置寄存器	0x0000_0000	0x0000_0000	不支持字节/半字访问

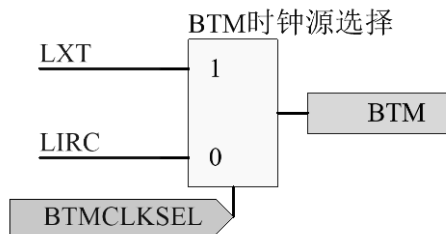
## 22 Base Timer (BTM)

### 22.1 概述

SC32F15G 系列内建一个 Base Timer (BTM)，可以按照 15.625ms ~ 32s 的间隔产生中断。32kHz LIRC 及外接 32.768kHz 晶体振荡器 LXT 都可作为 BTM 的时钟源。BTM 产生的中断可以将 CPU 从 STOP mode 唤醒。

### 22.2 时钟源

- SC32F15G 系列的 BTM 时钟源有两种：LXT 和 LIRC 可选



### 22.3 特性

- 中断频率间隔 15.625ms ~ 32s 可选
- 可唤醒 STOP Mode

### 22.4 BTM 中断

SC32F15G 系列的 BTM 计数满足 BTMFS 设置条件时，BTMIF 置起，如果 BTM\_CON.INTEN=1，将产生中断。

中断事件	事件标志位	中断使能控制位
BTM 中断请求	BTMIF	BTM_CON->INTEN

### 22.5 BTM 寄存器

#### 22.5.1 BTM 相关寄存器表

##### 22.5.1.1 低频定时器控制寄存器 BTM\_CON

寄存器	读/写	说明	复位值	上电初始值
BTM_CON	读/写	低频定时器控制寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
7	ENBTM	Base Timer 启动控制位 0: Base Timer 不启动 1: Base Timer 启动
6	INTEN	中断请求 CPU 的使能控制位 0: 禁止中断请求 1: 使能中断请求
3~0	BTMFS[3:0]	低频时钟中断频率选择 0000: 每 15.625ms 产生一个中断 0001: 每 31.25ms 产生一个中断 0010: 每 62.5ms 产生一个中断 0011: 每 125ms 产生一个中断 0100: 每 0.25 s 产生一个中断 0101: 每 0.5 s 产生一个中断 0110: 每 1.0 s 产生一个中断 0111: 每 2.0 s 产生一个中断 1000: 每 4.0 s 产生一个中断 1001: 每 8.0 s 产生一个中断 1010: 每 16.0 s 产生一个中断 1011: 每 32.0 s 产生一个中断 1100~1111: 保留
31~8 5~4	-	保留

### 22.5.1.2 BTM 标志位寄存器 BTM\_STS

寄存器	读/写	说明	复位值	上电初始值
BTM_STS	读/写	BTM 标志位寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
0	BTMIF	Base Timer 中断标志位 该位由硬件置 1，通过软件写 1 清 0。 BTM 计数满足 BTMFS 设置条件时该位自动置起。
31~1	-	保留

### 22.5.2 BTM 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值
BTM 基地址: 0x4002_1080					
BTM_CON	0x00	读/写	低频定时器控制寄存器	0x0000_0000	0x0000_0000
BTM_STS	0x04	读/写	BTM 标志位寄存器	0x0000_0000	0x0000_0000

## 23 内建 CRC 校验模块

### 23.1 概述

SC32F15G 系列内建一个 CRC 校验模块，使用多项式发生器从一个 8 位/16 位/32 位的数据字中产生 CRC 码。在众多的应用中，基于 CRC 的技术还常用来验证数据传输或存储的完整性。根据功能安全标准的规定，这些技术提供了验证 Flash 完整性的方法。CRC 计算单元有助于在运行期间计算软件的签名，并将该签名与链接时生成并存储在指定存储单元的参考签名加以比较。

### 23.2 时钟源

- SC32F15G 系列的 CRC 时钟源来自 HCLK

### 23.3 特性

- 内建了 1 个硬件 CRC 模块
- 初始值可设，默认为 0xFFFF\_FFFF
- 支持 8 Bits/16 Bits/32 Bits 数据单元
- 多项式可编程，默认为 0x04C1\_1DB7
- 仅支持软件送数计算模式
- 支持 DMA: CRC\_DR 可作为 DMA 的目标地址，也可直接寄存器访问
- 单独一个 byte 计算 CRC 需要 1 个系统时钟。

CRC 算法名称	CRC-32/MPEG-2
多项式公式	$x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$
数据宽度	32bit
初始值	0xFFFF_FFFF
结果异或值	0x0000_0000
输入值反转	false
输出值反转	false
LSB/MSB	MSB

注意：CRCDR 写入数据和读出不是同一数据。

## 23.4 CRC 寄存器

### 23.4.1 CRC 相关寄存器表

#### 23.4.1.1 CRC 数据寄存器 CRC\_DR

寄存器	读/写	说明	复位值	上电初始值
CRC_DR	读/写	CRC 数据寄存器（被计算的数/结果）	0xFFFF_FFFF	0x0000_0000

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

位编号	位符号	说明
31~0	CRCDR[31:0]	<p>CRC 数据寄存器位</p> <p>该寄存器用于向 CRC 计算器写入新数据。读取寄存器时可读出之前的 CRC 计算结果。如果数据大小小于 32 位，则最低有效位可用于写入 / 读取正确值。该寄存器的操作要求如下：</p> <ol style="list-style-type: none"> <li>1. 首先必须对 CRC_CON.CRCRST 写 1，将 CRCDR 复位</li> <li>2. 当 CRCREG 被写入时，硬件自动计算 CRC 结果，并继续存放于 CRCDR 内</li> </ol> <p>当需要时，即时读出 CRC 计算结果。</p>

#### 23.4.1.2 CRC 控制寄存器 CRC\_CON

寄存器	读/写	说明	复位值	上电初始值
CRC_CON	读/写	CRC 控制寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
7~6	POLYSIZE[1:0]	CRC 多项式大小设置位 00: 32 位多项式 01: 16 位多项式 10: 8 位多项式 11: 7 位多项式
0	CRCRST	CRCDR 寄存器复位 (Q31~Q0) 该位通过软件写 1, 由硬件自动清 0。 0: 无影响 1: 复位 CRCDR, 复位的值为用户写入 CRC_INIT 寄存器的值
31~8 5~1	-	保留

### 23.4.1.3 CRC 初始值寄存器 CRC\_INT

寄存器	读/写	说明	复位值	上电初始值
CRC_INT	读/写	CRC 初始值寄存器	0xFFFF_FFFF	0x0000_0000

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

位编号	位符号	说明
31~0	CRC_INIT[31:0]	可编程 CRC 初始值, 复位值: 0xFFFF_FFFF 此寄存器用于写入 CRC 初始值

### 23.4.1.4 CRC 多项式设置寄存器 CRC\_POL

寄存器	读/写	说明	复位值	上电初始值
CRC_POL	读/写	CRC 多项式设置寄存器	0x04C1_1DB7	0x0000_0000

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

位编号	位符号	说明
31~0	POL[31:0]	可编程多项式，复位值：0x04C1_1DB7 此寄存器用于写入要用于 CRC 计算的多项式系数，如果多项式数值小于 32 位，则必须使用最低有效位编程正确值。

### 23.4.2 CRC 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值	访问限制
CRC 基地址：0x4000_2000						
CRC_DR	0x00	读/写	CRC 数据寄存器	0xFFFF_FFFF	0x0000_0000	-
CRC_CON	0x04	读/写	CRC 控制寄存器	0x0000_0000	0x0000_0000	-
CRC_INT	0x08	读/写	CRC 初始值寄存器	0xFFFF_FFFF	0x0000_0000	不支持字节/ 半字访问
CRC_POL	0x0C	读/写	CRC 多项式设置寄存器	0x04C1_1DB7	0x0000_0000	不支持字节/ 半字访问

## 24 直接存储器访问 (DMA)

### 24.1 概述

直接存储器访问 (DMA) 控制器用于高速数据传输。DMA 控制器可以从一个地址到另一个地址传输数据，无需 CPU 介入。通过 DMA 进行数据传输可减少 CPU 的工作量，将节省下的 CPU 资源做其他应用。DMA 控制器包含 4 个通道，每个通道都直接连接专用的硬件 DMA 请求，每个通道都同样支持软件触发。DMA 控制器支持 4 级通道优先级，用于处理 DMA 请求间的优先级，确保同一时刻只有一个 DMA 通道工作。DMA 控制器也支持单一传输和批量传输，请求源可以是软件请求或接口请求，内存之间的数据传输是使用软件请求。

注：对于一个双向数据传输应用，需要 2 个 DMA 通道分别完成发送和接收。

### 24.2 时钟源

DMA 的时钟源为 HCLK，通过 AHB\_CFG.DMAEN 使能 DMA 的外设时钟。

### 24.3 特性

- 支持 4 个可独立配置的通道
- 支持 4 级通道优先级
- 支持 8 位，16 位，32 位数据传输
- 支持源和目标地址自动增加/减少或者固定，数据宽度支持字节，半字，字
- 支持单次和批量传输方式

### 24.4 功能说明

#### 24.4.1 传输方向

DMA 外设与存储之间传输无限制：

内存到内存	内存到外设	外设到内存	外设到外设
无限制	无限制	无限制	无限制

#### 24.4.2 DMA 访问区域限制

用户操作 DMA 时，不允许对 Flash 进行写操作，也不允许通过 DMA 操作内核，否则将产生无法预估的异常。

#### 24.4.3 通道优先级

通过 PL[1:0]位可设置四个级别的优先级：

- 00: 低
- 01: 中
- 10: 高
- 11: 非常高

#### 24.4.4 单一传输和批量传输

DMA 控制器支持单一和成组数据的传输类型，请求源可以是软件请求，接口请求，内存之间的数据传输是使用软件请求。单一传输的意思是软件或接口准备好传输一个数据（每个数据需要一次请求），批量传输的意思是软件或接口将传输多个数据（多个数据仅需一次请求）。

单一传输和批量传输模式可通过寄存器 TPTYPE（DMA<sub>n</sub>\_CFG[15]）设定。

当 DMA 控制器运行在单一传输模式，每搬移一个数据需要一次请求，当搬移一次数据，寄存器 DMA<sub>n</sub>\_CNT[31:0]，n=0~3 会减 1，直到 DMA<sub>n</sub>\_CNT[31:0] 中的数目递减为 0，搬移才会完成。在该模式，BURSIZE（DMA<sub>n</sub>\_CFG[14:12]）不用于控制搬移数据量大小，它的值固定为 1。

在批量搬移模式，DMA 控制器搬移 DMA<sub>n</sub>\_CNT[31:0] 个数据，仅需一次请求。当搬移 BURSIZE（DMA<sub>n</sub>\_CFG[14:12]）数据后，DMA<sub>n</sub>\_CNT[31:0] 中的数目会减去 BURSIZE。直到 DMA<sub>n</sub>\_CNT[31:0] 中的数目递减为 0，搬移数据才完成。

#### 24.4.5 循环模式

循环模式可用于处理循环缓冲区和连续数据流（例如 ADC 扫描模式）。在循环模式传输过程中，待传输数据的数目将自动重新装载为在通道配置阶段设置的初始值，并继续响应 DMA 请求。为停止循环传输，软件需要在禁止 DMA 通道前使外设停止生成 DMA 请求（例如退出 ADC 扫描模式）。软件必须在启动/使能传输前，以及在停止循环传输后，明确设定 DMACNT 值。

SC32F15G 系列的 DMA 控制器支持常规模式和循环模式：

- 当 CIRC=0（DMA 通道处于非循环模式）时，在达到设定的待传输数据数目时，将不再接受任何 DMA 请求；
- 当 CIRC=1（DMA 通道处于循环模式）时，在传输完成后该通道的 DMACNT 会自动重新装载之前设定的值，等待下一次循环。

用户可以根据实际需求灵活选择。

#### 24.4.6 DMA 通道使能后禁止操作的控制位

状态位、标志清零位不受限制，其余控制位均受 CHEN=1 操作限制，其目的应是为了防止传输过程修改配置导致数据传输不确定。

DMA 通道使能后，寄存器位域/位、源/目标地址、优先级、传输控制相关均不可改写。

### 24.5 DMA 中断

对于每个 DMA 通道 n，n=0~3，在发生“传输完成”、“半传输”或“传输错误”时都会生成中断。可以使用单独的中断使能位以提高灵活性。

中断事件	事件标志位	中断请求控制位	子事件标志位	中断使能子开关
DMA 通道 n 传输完成	GIF	DMA <sub>n</sub> _CFG ->INTEN	TCIF	TCIE
DMA 通道 n 传输一半			HTIF	HTIE
DMA 通道 n 传输错误			TEIF	TEIE

## 24.6 DMA 寄存器

### 24.6.1 DMA 相关寄存器表

#### 24.6.1.1 DMA 通道 n 传输源地址缓存寄存器 DMA<sub>n</sub>\_SADR

寄存器	读/写	说明	复位值	上电初始值
DMA <sub>n</sub> _SADR n = 0~3	读/写	DMA 通道 n 传输源地址缓存寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
31~0	SADR[31:0]	<p>DMA 传输源地址缓存</p> <ul style="list-style-type: none"> <li>● 读取： <ul style="list-style-type: none"> <li>■ 当通道打开时，读到的是内部的源地址工作寄存器；</li> <li>■ 当通道禁止时，读到的是表面上的源地址缓存寄存器。</li> </ul> </li> <li>● 更新： <ul style="list-style-type: none"> <li>■ 每搬运完一次，源地址工作寄存器会根据 SAINC[1:0]设置自动变化，其变化的宽度根据 TXWIDTH[1:0]决定。</li> <li>■ 循环模式下（SAINC == 11），源地址缓存寄存器会重载至源地址工作寄存器。</li> </ul> </li> <li>● 写入： <ul style="list-style-type: none"> <li>■ 写源地址缓存寄存器的条件：CHEN=0，或 CHEN=1，但 DMA 通道已传输完成，且处于 IDLE 状态。</li> </ul> </li> </ul>

#### 24.6.1.2 DMA 通道 n 传输目标地址缓存寄存器 DMA<sub>n</sub>\_DADR

寄存器	读/写	说明	复位值	上电初始值
DMA <sub>n</sub> _DADR n = 0~3	读/写	DMA 通道 n 目标地址缓存寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
DADR[31:24]							
23	22	21	20	19	18	17	16
DADR[23:16]							
15	14	13	12	11	10	9	8
DADR[15:8]							

7	6	5	4	3	2	1	0
DADR[7:0]							

位编号	位符号	说明
31~0	DADR[31:0]	<p>DMA 传输目标地址缓存</p> <ul style="list-style-type: none"> <li>● 读取： <ul style="list-style-type: none"> <li>■ 当通道打开时，读到的是内部的目标地址工作寄存器；</li> <li>■ 当通道禁止时，读到的是表面上的目标地址缓存寄存器。</li> </ul> </li> <li>● 更新： <ul style="list-style-type: none"> <li>■ 每搬运完一次，目标地址工作寄存器会根据 DAINC[1:0]设置自动变化，其变化的宽度根据 TXWIDTH[1:0]决定。</li> <li>■ 循环模式下（SAINC == 11），目标地址缓存寄存器会重载至目标地址工作寄存器。</li> </ul> </li> <li>● 写入： <ul style="list-style-type: none"> <li>■ 写目标地址缓存寄存器的条件：CHEN=0，或 CHEN=1，但 DMA 通道已传输完成，且处于 IDLE 状态。</li> </ul> </li> </ul>

### 24.6.1.3 DMA 通道 n 控制/配置寄存器 DMA<sub>n</sub>\_CFG

寄存器	读/写	说明	复位值	上电初始值
DMA <sub>n</sub> _CFG n = 0~3	读/写	DMA 通道 n 控制/配置寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	REQSRC[5:0]					
23	22	21	20	19	18	17	16
CHRQ	-	-	-	TEIE	HTIE	TCIE	INTEN
15	14	13	12	11	10	9	8
TPTYPE	BURSIZE[2:0]			SAINC[1:0]		DAINC[1:0]	
7	6	5	4	3	2	1	0
CHEN	CHRST	PAUSE	CIRC	TXWIDTH[1:0]		PL[1:0]	

位编号	位符号	说明
29~24	REQSRC[5:0]	<p>DMA 通道请求源选择位</p> <p>0: 禁用当前 DMA 通道的外设请求</p> <p>选择以下配置值，当前 DMA 通道的外设请求源将配合设置项中的外设 DMA 请求使能开关产生：</p> <p>2: UART0_IDE-&gt;TXDMAEN  3: UART0_IDE-&gt;RXDMAEN  4: UART1_IDE-&gt;TXDMAEN  5: UART1_IDE-&gt;RXDMAEN  12: SPI0_IDE-&gt;TXDMAEN  13: SPI0_IDE-&gt;RXDMAEN  14: SPI1_IDE-&gt;TXDMAEN  15: SPI1_IDE-&gt;RXDMAEN  20: TWI0_IDE-&gt;TXDMAEN  21: TWI0_IDE-&gt;RXDMAEN  33: TIM1_IDE-&gt;TIDE  34: TIM1_IDE-&gt;CAPFDE  35: TIM1_IDE-&gt;CAPRDE  36: TIM2_IDE-&gt;TIDE  37: TIM2_IDE-&gt;CAPFDE</p>

位编号	位符号	说明
		38: TIM2_IDE->CAPRDE 59: ADCCON->DMAEN 60: DMA0_CFG->CHRQ 61: DMA1_CFG->CHRQ 62: DMA2_CFG->CHRQ 63: DMA3_CFG->CHRQ 其它: 禁用 DMA 外设请求
23	CHRQ	DMA 通道的 DMA 请求使能位 0: 禁止, 当前 DMA 通道禁止作为其它 DMA 通道的请求源 1: 使能, 当前 DMA 通道可作为其它 DMA 通道的请求源, 即当前 DMA 通道和其它外设一样, 可产生 DMA 请求。 该位使能后, 可以实现 DMA 请求 DMA, 例如: CHRQ =1, DMA 通道 n 完成数据搬运后, 向 DMA 通道 m 产生一个 DMA 请求, 通道 m 响应请求, 将预先配置好的参数表更新至通道 n 的寄存器, 从而实现通道 n 的参数自动更新。 <b>注意: CHRQ 置起后, 作为请求源的 DMA 能够进行数据搬运, 但不会置起标志位和进入相应中断, 需要 CHRQ 位置 0 后, 才会置起标志位及进入中断。</b>
19	TEIE	DMA 传输错误中断使能位 0: DMA 传输错误中断除能 1: DMA 传输错误中断使能
18	HTIE	DMA 传输一半中断使能位 0: DMA 传输一半中断除能 1: DMA 传输一半中断使能
17	TCIE	DMA 传输完成中断使能位 0: DMA 传输完成中断除能 1: DMA 传输完成中断使能
16	INTEN	中断请求 CPU 的使能控制位 0: 禁止中断请求 1: 使能中断请求
15	TPTYPE	DMA 通道传输类型选择位 0: 单次传输 1: 批量传输。批量传输模式下, DMA 控制器搬运 DMACNT 个数据仅需一次请求, 通道响应该请求后, 数据将以 Burst 方式进行传输, 即以 BURSIZE 为单位进行数据搬运直到 DMACNT 递减为 0, 一次批量传输模式下的数据处理才算完成。
14~12	BURSIZE[2:0]	批量传输时, 基于 Burst 传输方式下的定义, Burst 大小可选择: 000: 128 001: 64 010: 32 011: 16 100: 8 101: 4 110: 2 111: 1
11~10	SAINC[1:0]	DMA 通道传输源地址增减模式设置位 00: 无增量 (固定地址模式) 01: 增量模式 10: 减量模式 11: 递增循环模式 (见 DMA 传输源地址缓存寄存器) SAINC[1:0]的值可以任意修改, 在通道禁止时立即生效; 在通道使能时, 修改值在循环模式重装载时生效

位编号	位符号	说明
9~8	DAINC[1:0]	DMA 传输目标地址增减模式设置位 00: 无增量（固定地址模式） 01: 增量模式 10: 减量模式 11: 递增循环模式（见 DMA 传输目标地址缓存寄存器） DAINC[1:0]的值可以任意修改，在通道禁止时立即生效；在通道使能时，修改值在循环模式重装载时生效
7	CHEN	DMA 通道使能位 0: DMA 通道禁止 1: DMA 通道使能
6	CHRST	DMA 通道复位控制位，该位用于控制 DMA 通道复位。 0: 无效 1: 当前 DMA 通道复位。此时，当前 DMA 通道的 CHEN 被除能，中断标志为被清除，其他寄存器的值保持不变
5	PAUSE	DMA 通道传输暂停控制位 0: 无效 1: 当前 DMA 通道暂停。此时，当前 DMA 通道的 CHEN 被除能，状态机在完成当前读写周期后回到 state=1，内部寄存器的值处于保持状态（源/目的地址，计数器）。暂停后若要恢复搬运，需同时对 CHEN 和 PAUSE 赋值：CHEN=1,PAUSE=0。
4	CIRC	DMA 通道循环模式使能位 0: 通道未处于循环模式，在达到设定的待传输数据数目时，该通道的 DMACNT 会保持为零； 1: 通道处于循环模式，在传输完成后该通道的 DMACNT 会自动重新装载之前设定的值。 循环模式可用于处理循环缓冲区和连续数据流（例如 ADC 扫描模式）。在循环模式传输过程中，待传输数据的数目将自动重新装载为在通道配置阶段设置的初始值，并继续响应 DMA 请求。为停止循环传输，软件需要在禁止 DMA 通道前使外设停止生成 DMA 请求（例如退出 ADC 扫描模式）。软件必须在启动/使能传输前，以及在停止循环传输后，明确设定 DMACNT 值。
3~2	TXWIDTH[1:0]	DMA 通道传输宽度选择位 选择当前 DMA 通道的源地址及目标地址每次传输的数据宽度： 00: 8bit 01: 16bit 10: 32bit 11: 32bit TXWIDTH[1:0] 的值可以任意修改，在通道禁止时，立即生效；在通道打开时，修改值在循环模式重装载时生效。
1~0	PL[1:0]	DMA 通道优先级设置位 在 DMA 已经有通道在工作，且其它通道也接收到请求正挂起，当正在工作的通道结束后将启动优先级仲裁。 00: 低 01: 中 10: 高 11: 非常高 <b>注意：同等优先级配置，通道号越小优先级越高。</b>
31~30 22~20	-	保留

#### 24.6.1.4 DMA 通道 n 计数器缓存寄存器 DMA<sub>n</sub>\_CNT

寄存器	读/写	说明	复位值	上电初始值
DMA <sub>n</sub> _CNT n = 0~3	读/写	DMA 通道 n 计数器缓存寄存器	0x0000_0000	0x0000_0000

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

位编号	位符号	说明
31~0	DMACNT[31:0]	<p>DMA 通道计数器缓存寄存器</p> <ul style="list-style-type: none"> <li>● 写入： <ul style="list-style-type: none"> <li>■ DMACNT 的值等于当前 DMA 通道的剩余传输次数；</li> <li>■ 每个 DMA 通道内部都有一个“工作计数器”，该计数器会在每次搬运后以为 TXWIDTH 为单位递减： <ul style="list-style-type: none"> <li>◆ 当 CIRC=0（DMA 通道处于非循环模式）时，“工作计数器”递减到 0 后，将不再接受任何 DMA 请求。</li> <li>◆ 当 CIRC=1（DMA 通道处于循环模式）时，“工作计数器”递减到 0 后，会将 DMACNT 的值重载到“工作计数器”内，等待下一次循环。</li> </ul> </li> </ul> </li> <li>● 读取： <ul style="list-style-type: none"> <li>■ 当通道禁止时，读出的是 DMACNT 的值；</li> <li>■ 当通道打开时，读出的是内部的“工作计数器”的实时数据。</li> </ul> </li> </ul>

#### 24.6.1.5 DMA 通道 n 状态寄存器 DMA<sub>n</sub>\_STS

寄存器	读/写	说明	复位值	上电初始值
DMA <sub>n</sub> _STS n = 0~3	读/写	DMA 通道 n 状态寄存器	0x0000_0000	0x0000_0000

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	SWREQ
7	6	5	4	3	2	1	0
STATUS[3:0]				TEIF	HTIF	TCIF	GIF

位编号	位符号	说明
8	SWREQ	DMA 通道的软件请求触发位 该位写 1 后，当前 DMA 通道会一直挂起软件请求，直到当前 DMA 通道响应，并将该位自动硬件清 0。
7~4	STATUS[3:0]	DMA 通道状态位 0000：空闲。 0001：写入源地址 0010：读取源地址数据，并写入目的地址 0011：写入目的地址数据

位编号	位符号	说明
		0100: 保留 0101: 挂起等待中 (有通道在忙, 其他通道请求挂起) 0110: 暂停等待中 (批量传输模式时 PAUSE 写 1 后) 0111: burst 传输中 1000: burst 传输停止: PAUSE 使能、DMACNT 计数到 0, 或 bursize 计数到 0 均会进入此状态
3	TEIF	DMA 传输错误中断标志位 当 DMA 读写到未定义的地址时, TEIF 会被硬件置 1。 该位写 1 清零。
2	HTIF	DMA 传输一半中断标志位 当 DMACNT 的计数值计数到 DMACNT/2 时, HTIF 会被硬件置 1。 该位写 1 清零。
1	TCIF	DMA 传输完成中断标志位 当 DMACNT 的计数值计到 0 时, TCIF 会被硬件置 1。 该位写 1 清零。
0	GIF	DMA 通道全局中断标志位 0: 当前 DMA 通道无中断产生 1: 当前 DMA 通道产生中断: 传输错误、传输到一半或传输完成
31~9	-	保留

#### 24.6.2 DMA 寄存器映射

寄存器	偏移地址	读/写	说明	复位值	上电初始值
DMA0 基地址: 0x4001_0800					
DMA0_SADR	0x00	读/写	DMA 源地址缓存寄存器	0x0000_0000	0x0000_0000
DMA0_DADR	0x04	读/写	DMA 目标地址缓存寄存器	0x0000_0000	0x0000_0000
DMA0_CFG	0x08	读/写	DMA 控制/配置寄存器	0x0000_0000	0x0000_0000
DMA0_CNT	0x0C	读/写	DMA 计数器缓存寄存器	0x0000_0000	0x0000_0000
DMA0_STS	0x10	读/写	DMA 状态寄存器	0x0000_0000	0x0000_0000
DMA1 基地址: 0x4001_0840					
DMA1_SADR	0x00	读/写	DMA 源地址缓存寄存器	0x0000_0000	0x0000_0000
DMA1_DADR	0x04	读/写	DMA 目标地址缓存寄存器	0x0000_0000	0x0000_0000
DMA1_CFG	0x08	读/写	DMA 控制/配置寄存器	0x0000_0000	0x0000_0000
DMA1_CNT	0x0C	读/写	DMA 计数器缓存寄存器	0x0000_0000	0x0000_0000
DMA1_STS	0x10	读/写	DMA 状态寄存器	0x0000_0000	0x0000_0000
DMA2 基地址: 0x4001_0880					
DMA2_SADR	0x00	读/写	DMA 源地址缓存寄存器	0x0000_0000	0x0000_0000

寄存器	偏移地址	读/写	说明	复位值	上电初始值
DMA2_DADR	0x04	读/写	DMA 目标地址缓存寄存器	0x0000_0000	0x0000_0000
DMA2_CFG	0x08	读/写	DMA 控制/配置寄存器	0x0000_0000	0x0000_0000
DMA2_CNT	0x0C	读/写	DMA 计数器缓存寄存器	0x0000_0000	0x0000_0000
DMA2_STS	0x10	读/写	DMA 状态寄存器	0x0000_0000	0x0000_0000
DMA3 基地址: 0x4001_08C0					
DMA3_SADR	0x00	读/写	DMA 源地址缓存寄存器	0x0000_0000	0x0000_0000
DMA3_DADR	0x04	读/写	DMA 目标地址缓存寄存器	0x0000_0000	0x0000_0000
DMA3_CFG	0x08	读/写	DMA 控制/配置寄存器	0x0000_0000	0x0000_0000
DMA3_CNT	0x0C	读/写	DMA 计数器缓存寄存器	0x0000_0000	0x0000_0000
DMA3_STS	0x10	读/写	DMA 状态寄存器	0x0000_0000	0x0000_0000

## 25 SysTick

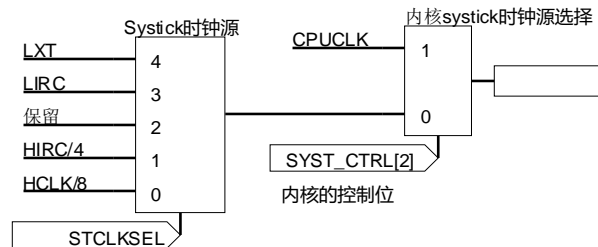
SysTick 是一个简单、24 位写入清 0、递减、带灵活控制机制的自动装载计数器。该计数器可以用作实时操作系统（RTOS）的滴答定时器或作为一个简单的计数器。

### 25.1 时钟源

SysTick（Cortex<sup>®</sup>-M0+内核系统定时器）的时钟源分为内部时钟源和外部时钟源：

- 内部时钟源，即 CPU 时钟
- 4 个外部时钟源

SysTick 时钟源框图如下：



### 25.2 SysTick 校准寄存器默认值

SysTick 校准寄存器的校准值设置方法如下：

- 若上电默认时钟为  $f_{HCLK}/n$  (MHz)， $n$  是上电默认分频系数，上电默认时钟源为 HIRC；
- 则当 SysTick 校准值初始值为  $1000 * (f_{HCLK}/n)$  时，可产生 1ms 时间基准。

## 26 版本记录

版本	记录	日期
V1.0	正式版	2026 年 03 月 18 日
V0.1	初版	2024 年 06 月 18 日

## 27 声明

深圳市赛元微电子股份有限公司（以下简称赛元）保留随时对赛元产品、文档或服务进行变更、更正、增强、修改和改进的权利，恕不另行通知。赛元认为提供的信息是准确可信的。本文档信息于 2024 年 6 月开始使用。在实际进行生产设计时，请参阅各产品最新的数据手册等相关资料。