

说明

CH32L103 系列产品是基于 32 位 RISC-V 指令集及架构设计的工业级通用微控制器。采用青稞 V4C 内核，支持硬件中断堆栈，提升中断响应效率；该系列产品挂载了丰富的外设接口和功能模块。其内部组织架构满足低成本低功耗嵌入式应用场景。

本手册针对用户的应用开发，提供了 CH32L103 产品的详细使用信息。

RISC-V 内核版本对比概览

内核版本	特点 指令集	硬件堆栈级数	中断嵌套级数	快速中断通道数	整数除法周期	向量表模式	扩展指令	内存保护
青稞 V4B	IMAC	2	2	4	9	地址或指令	支持	无
青稞 V4C	IMAC	2	2	4	5	地址或指令	支持	标准
青稞 V4F	IMAFC	3	8	4	5	地址或指令	支持	标准

寄存器中位属性缩写描述：

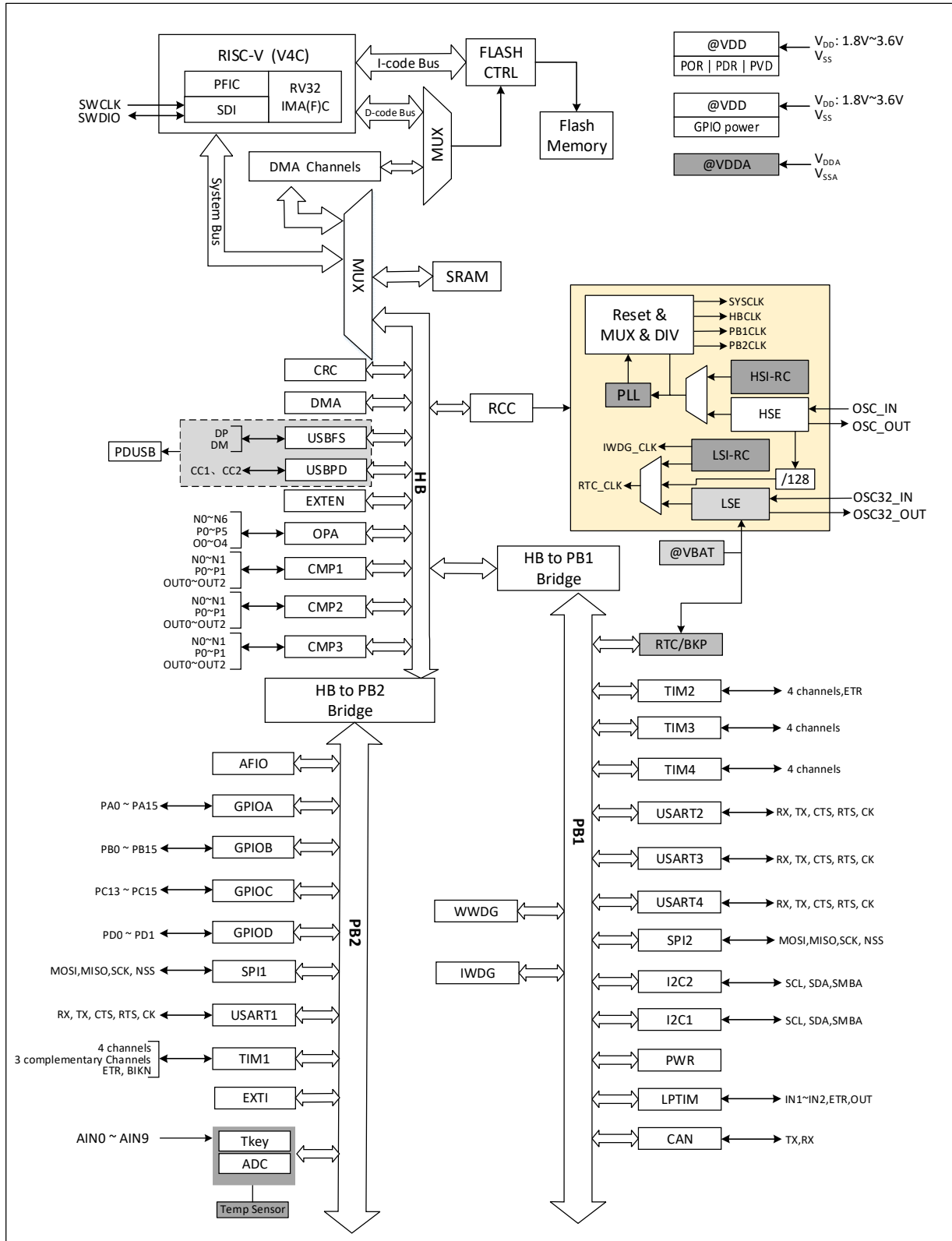
寄存器位属性	属性描述
RF	只读属性，读出固定值。
RO	只读属性，由硬件改变。
RZ	只读属性，读操作后自动位清 0
WO	只写属性（不可读，读值不确定）
WA	只写属性，安全模式下可写入。
WZ	只写属性，写操作后自动位清 0
RW	可读，可写。
RWA	可读，安全模式下可写入。
RW1	可读，写 1 有效，写 0 无效。
RW0	可读，写 0 有效，写 1 无效。
RW1T	可读，写 0 无效，写 1 翻转。
SC	自动清除。

第 1 章 存储器和总线架构

1.1 总线架构

该系列产品是基于 RISC-V 指令集设计的通用微控制器，其架构中的内核、仲裁单元、DMA 模块、SRAM 存储等部分通过多组总线实现交互。其系统框图见图 1-1。

图 1-1 系统框图



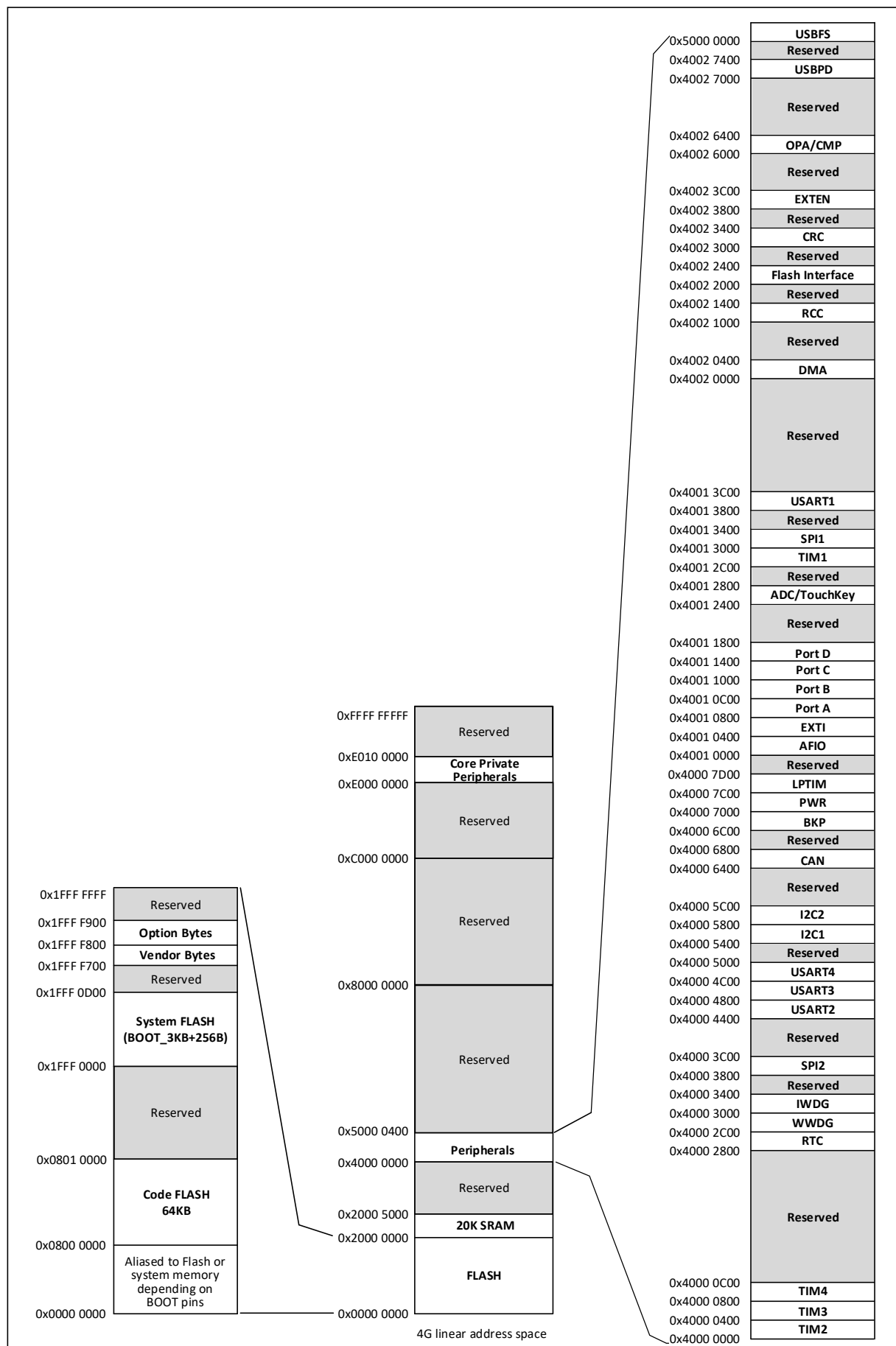
系统中设有：通用 DMA 控制器用以减轻 CPU 负担、提高效率；时钟树分级管理用以降低了外设总的运行功耗，同时还兼有数据保护机制，时钟安全系统保护机制等措施来增加系统稳定性。

1.2 存储器映像

该产品包含了程序存储器、数据存储器、内核寄存器和外设寄存器等等，它们都在一个 4GB 的线性空间寻址。

系统存储以小端格式存放数据，即低字节存放在低地址，高字节存放在高地址。

图 1-2 存储映像



1.2.1 存储器分配

内置最大 20K 字节的 SRAM，起始地址 0x20000000，支持字节、半字(2 字节)、全字(4 字节)访问。

内置最大 64K 字节的程序闪存存储区(CodeFlash)，用于存储用户应用程序。

内置 3K+256 字节的系统存储器(boot loader)，用于存储系统引导程序（厂家固化自举加载程序）。

内置 256 字节空间用于厂商配置字存储，出厂前固化，用户不可修改。

内置 256 字节空间用于用户选择字存储。

1.3 启动配置

系统可以通过 B00T0 和 B00T1 引脚来选择三种不同的启动模式。

表 1-1 启动模式

B00T0	B00T1	启动模式
0	X	从程序闪存存储器启动
1	0	从系统存储器启动
1	1	从内部 SRAM 启动

用户通过设置 B00T 引脚的状态值来选择复位后的启动模式。系统复位后或者电源复位都会导致 B00T 引脚的值被重新锁存。

启动模式不同，程序闪存存储器、系统存储器和内部 SRAM 有着不同的访问方式：

- 从程序闪存存储器启动时，程序闪存存储器地址被映射到 0x00000000 地址区域，同时也能够在原地址区域 0x08000000 访问。
- 从系统存储器启动时，系统存储器地址被映射到 0x00000000 地址区域，同时也能够在原地址区域 0x1FFF0000 访问。
- 从内部 SRAM 启动，只能够从 0x20000000 地址区域访问。

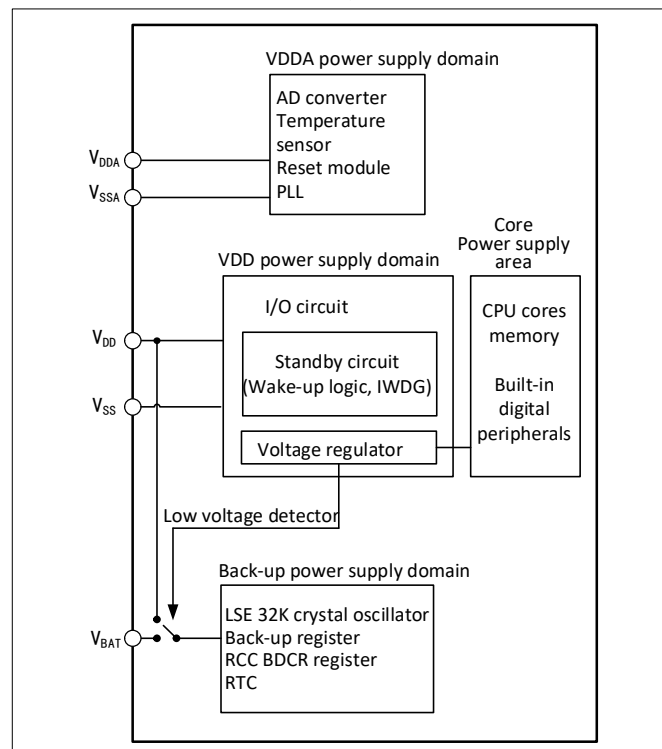
第2章 电源控制（PWR）

2.1 概述

系统工作电压 V_{DD} 范围为 1.8~3.6V，内置电压调节器提供内核所需的低压电源。当主电源 V_{DD} 掉电后，电池等后备电源可通过 V_{BAT} 引脚为实时时钟 (RTC) 和后备寄存器提供电源，如果无需后备电源，建议将 V_{DD} 直接连接到 V_{BAT} 引脚上。

V_{DDA} 和 V_{SSA} 引脚专门为系统中模拟相关电路供电，包括 ADC、温度传感器等。

图 2-1 电源结构框图



在主电源 V_{DD} 掉电后，模拟开关切换至 V_{BAT} ，后备区域由 V_{BAT} 引脚供电，此时 PC13~15 无法作为 GPIO，仅可使用如下功能：

- PC13 可以作为 TAMPER 引脚、RTC 闹钟或秒输出。
- PC14 和 PC15 只能用作 LSE 引脚。

当主电源 V_{DD} 上电稳定后，系统自动切换后备区域由 V_{DD} 供电，PC13~15 可以用作 GPIO 功能。

当 PC13~15 引脚作为 GPIO 输出时，速度必须限制在 2MHz 以下，最大负载电容为 30pF，并且禁止用在持续输出和吸入电流的场合，比如 LED 驱动。

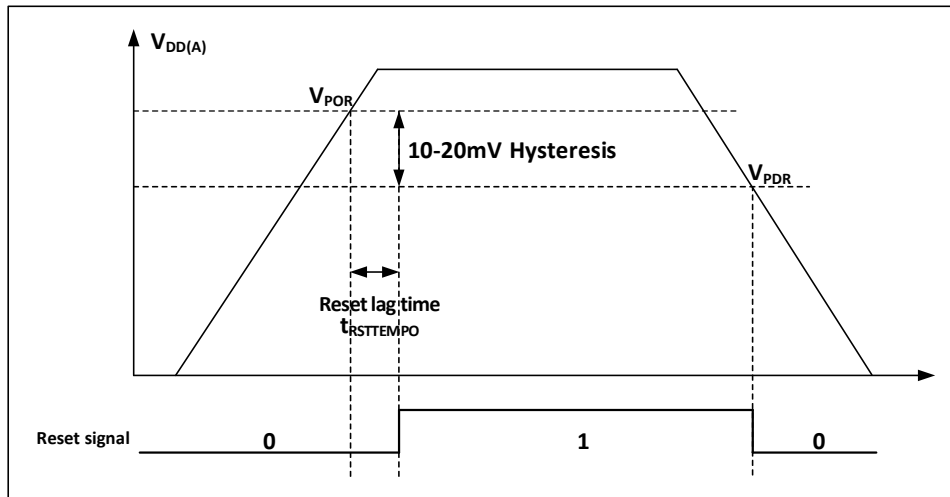
注：在主电源 V_{DD} 恢复供电过程中，内部 V_{BAT} 电源仍然通过对应的 V_{BAT} 引脚连在外部备用电源上，若 V_{DD} 在小于复位滞后时间 $t_{RSTTEMPO}$ 内就达到稳定，并且高于 V_{BAT} 的值 0.6V 以上，则有可能存在较短瞬间，电流通过 V_{DD} 与 V_{BAT} 之间的二极管灌入 V_{BAT} ，进而通过 V_{BAT} 引脚注入电池等后备电源，如果后备电源无法承受这样瞬时注入电流，建议在后备电源和 V_{BAT} 引脚之间加一只正向导通低压降二极管。

2.2 电源管理

2.2.1 上电复位和掉电复位

系统内部集成了上电复位 POR 和掉电复位 PDR 电路，当芯片供电电压 V_{DD} 和 V_{DDA} 低于对应门限电压时，系统被相关电路复位，无需外置额外的复位电路。上电门限电压 V_{POR} 和掉电门限电压 V_{PDR} 的参数请参考对应的数据手册。

图 2-2 POR 和 PDR 的工作示意图



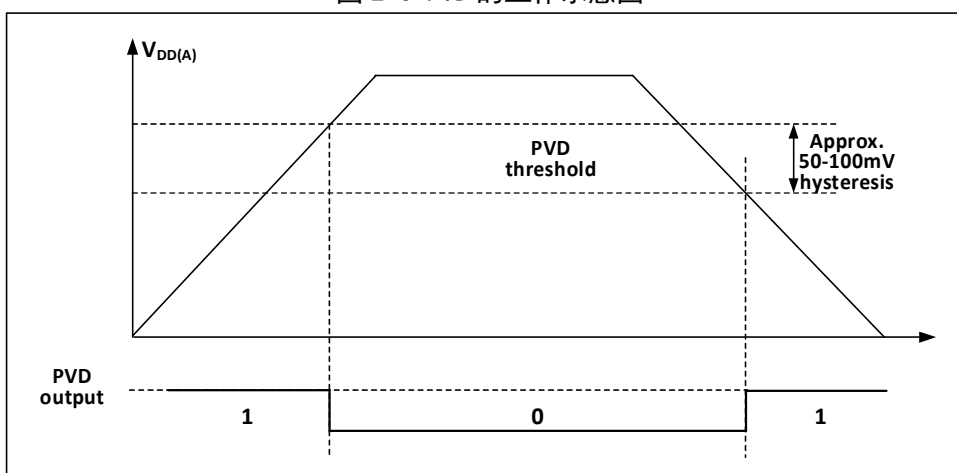
2.2.2 可编程电压监测器

可编程电压监测器 PVD，主要被用于监控系统主电源的变化，与电源控制寄存器 PWR_CTLR 的 PLS[2:0] 所设置的门檻电压相比较，配合外部中断寄存器（EXTI）设置，可产生相关中断，以便及时通知系统进行数据保存等掉电前操作。

具体配置如下：

- 1) 设置 PWR_CTLR 寄存器的 PLS[2:0] 域，选择要监控电压阈值。
- 2) 可选的中断处理。PVD 功能内部连接 EXTI 模块的第 16 线的上升/下降边沿触发设置，开启此中断（配置 EXTI），当 V_{DD} 下降到 PVD 阈值以下或上升到 PVD 阈值之上时就会产生 PVD 中断。
- 3) 设置 PWR_CTLR 寄存器的 PVDE 位来开启 PVD 功能。
- 4) 读取 PWR_CSR 状态寄存器的 PVD0 位可获取当前系统主电源与 PLS[2:0] 设置阈值关系，执行相应软处理。当 V_{DD} 电压高于 PLS[2:0] 设置阈值，PVD0 位置 0；当 V_{DD} 电压低于 PLS[2:0] 设置阈值，PVD0 位置 1。

图 2-3 PVD 的工作示意图



2.3 低功耗模式

在系统复位后，微控制器处于正常工作状态（运行模式），此时可以通过降低系统主频或者关闭不用外设时钟或者降低工作外设时钟来节省系统功耗。如果系统不需要工作，可设置系统进入低功耗模式，并通过特定事件让系统跳出此状态。

微控制器目前提供了 3 种低功耗模式，从处理器、外设、电压调节器的工作差异上分为：

- 睡眠模式：内核停止运行，所有外设（包含内核私有外设）仍在运行。

- 停止模式：停止所有时钟，唤醒后系统继续运行。停止模式分为四档，分别为：停止模式 1、停止模式 2、停止模式 3，停止模式 4，相应的配置方法如表 2-1 所示，停止模式四个档位配置后的电流参考 CH32L103DS0 手册。
- 待机模式：停止所有时钟，唤醒后微控制器复位（电源复位）。

表 2-1 低功耗模式一览

模式	进入	唤醒源	对时钟的影响	电压调节器
睡眠 (SLEEP)	WFI WFE	任意中断唤醒 唤醒事件唤醒	内核时钟关闭， 其他时钟无影响	正常
停止 (STOP)	SLEEPDEEP 置 1 PDDS 清 0 WFI 或 WFE	任一外部中断/事件（在外部中断寄存器中设置）、NRST 引脚复位、WKUP 引脚上升沿、PVD 的输出、RTC 闹钟事件、USB 的唤醒信号、USB PD 唤醒信号、CMP 唤醒信号、LPTIM 唤醒信号	关闭 HSE、HSI、PLL 和外设时钟	停止模式 1： LPDS=0, PDDS=0 停止模式 2（LDO 节能模式）： LPDS=0, PDDS=0, AUTO_LDO_EC=1 或 LPDS=0, PDDS=0, LDO_EC=1 停止模式 3： RAMLV=0, PDDS=0, LPDS=1 停止模式 4： RAMLV=1, PDDS=0, LPDS=1
待机 (STANDBY)	SLEEPDEEP 置 1 PDDS 置 1 WFI 或 WFE	WKUP 引脚上升沿、RTC 闹钟事件、NRST 引脚复位、IWDG 复位、PVD 的输出。 注：EXTI0~EXTI17 任一外部事件也可以唤醒系统，但唤醒后系统复位。	关闭 HSE、HSI、PLL 和外设时钟	关闭

注：SLEEPDEEP 位属于内核私有外设控制位，参考 PFIC_SCTLR 寄存器。

2.3.1 低功耗配置选项

● WFI 和 WFE 方式

WFI：微控制器被具有中断控制器响应的中断源唤醒，系统唤醒后，将最先执行中断服务函数（微控制器复位除外）。

WFE：唤醒事件触发微控制器将退出低功耗模式。唤醒事件包括：

- 1) 配置一个外部或内部的 EXTI 线为事件模式，此时无需配置中断控制器；
- 2) 或者配置某个中断源，等效为 WFI 唤醒，系统优先执行中断服务函数；
- 3) 或者配置 SEVONPEND 位，开启外设中断使能，但不开启中断控制器中的中断使能，系统唤醒后需要清除中断挂起位。

● SLEEPONEXIT

启用：执行 WFI 或 WFE 指令后，微控制器确保所有待处理的中断服务退出后进入低功耗模式。

不启用：执行 WFI 或 WFE 指令后，微控制器立即进入低功耗模式。

● SEVONPEND

启用：所有中断或者唤醒事件都可以唤醒通过执行 WFE 进入的低功耗。

不启用：只有在中断控制器中使能的中断或者唤醒事件可以唤醒通过执行 WFE 进入的低功耗。

2.3.2 睡眠模式（SLEEP）

此模式下，所有的 IO 引脚都保持他们运行模式下的状态，所有的外设时钟都正常，所以进入睡

眠模式前，尽量关闭无用的外设时钟，以减低功耗。该模式唤醒所需时间最短。

进入：配置内核寄存器控制位 SLEEPDEEP=0，电源控制寄存器 PDDS=0，LPDS 决定内部调压器状态，执行 WFI 或 WFE，可选 SEVONPEND 和 SLEEPONEXIT。

退出：任意中断或者唤醒事件。

2.3.3 停止模式 (STOP)

停止模式是在内核的深睡眠模式 (SLEEPDEEP) 基础上结合了外设的时钟控制机制，并让电压调节器的运行处于更低功耗的状态。此模式高频时钟 (HSE/HSI/PLL) 域被关闭，SRAM 和寄存器内容保持，IO 引脚状态保持。该模式唤醒后系统可以继续运行，HSI 为默认系统时钟。

如果正在进行闪存编程，直到对内存访问完成，系统才进入停止模式；如果正在进行对 PB 的访问，直到对 PB 访问完成，系统才进入停止模式。

停止模式下可工作模块：独立看门狗 (IWDG)、实时时钟 (RTC)、低频时钟 (LSI/LSE)、低功耗定时器 (LPTIM)。

进入：配置内核寄存器控制位 SLEEPDEEP=1，电源控制寄存器的 PDDS=0，可选 LPDS 位，执行 WFI 或 WFE，可选 SEVONPEND 和 SLEEPONEXIT。在进入 STOP 电压调节器低功耗模式 (PDDS=0, LPDS=1) 前，预置 FLASH_LP[1:0] = 10b，即 FLASH 低功耗模式 1。

退出：任一外部中断/事件 (在外部中断寄存器中设置)，NRST 引脚上外部复位，PVD 的输出，RTC 闹钟，USB 的唤醒信号，USB PD 唤醒信号，CMP 唤醒信号，LPTIM 唤醒信号等。

通过不同配置选择不同等级的停止模式。在停止模式下，即 PDDS=0，可选 LPDS 位，LPDS=0，电压调节器工作在停止模式 1；LPDS=0，AUTO_LDO_EC=1，MCU 进入停止模式时电源系统会自动节能，即进入停止模式 2，或者 LPDS=0，LDO_EC=1，也为停止模式 2；LPDS=1，电压调节器工作在停止模式 3。在停止模式 3 基础下，通过配置 PWR_CTLR 寄存器的 RAMLV=1，使能 RAM 低电压模式，功耗达到最低，称为停止模式 4。

2.3.4 待机模式 (STANDBY)

待机模式对比停止模式，唯一的差别在于：在某些指定的唤醒条件下退出后，微控制器将被复位，并且执行的是电源复位。

待机模式下可工作模块：独立看门狗 (IWDG)、实时时钟 (RTC)、低频时钟 (LSI/LSE)。

进入：配置内核寄存器控制位 SLEEPDEEP=1，电源控制寄存器的 PDDS=1，执行 WFI 或 WFE，可选 SEVONPEND 和 SLEEPONEXIT，且在进入 STANDBY 模式前，预置 FLASH_LP[1:0] = 10b，即 FLASH 低功耗模式 1。

退出：

- 1) EXTIO~EXTI17 任一外部事件 (在外部中断寄存器中设置)，此唤醒后微控制器执行电源复位。
- 2) WKUP 引脚的上升沿、RTC 闹钟事件的上升沿、NRST 引脚上外部复位、IWDG 复位，PVD 的输出，RTC 闹钟，此唤醒后微控制器执行电源复位。

在待机模式下，当正常供电时，通过配置 PWR_CTLR 寄存器的 R2KSTY=1 控制 2K 字节 RAM 不掉电，R18KSTY=1 控制 18K 字节 RAM 不掉电；当使用 VBAT 供电时，通过配置 PWR_CTLR 寄存器的 R2KVBAT=1 控制 2K 字节 RAM 不掉电，R18KVBAT=1 控制 18K 字节 RAM 不掉电。在该基础之上，可以通过配置 PWR_CTLR 寄存器的 RAMLV=1，使能 RAM 低电压模式，功耗达到最低。

注：调试模式下，使微处理器进入停止或待机模式，将失去调试连接。

R2KSTY=1 控制 2K 字节 RAM 的地址范围：0x20000000—0x20000000+2K

R18KSTY=1 控制 18K 字节 RAM 的地址范围：0x20000000+2K—0x20000000+2K+18K

2.3.5 RTC 自动唤醒

RTC 可以实现无需外部中断的情况下自动唤醒。通过对时间基数进行编程，可周期性地从停止或待机模式下唤醒。

可选择精准的外部低频 32.768KHz 晶振 LSE 作为 RTC 时钟源，也可以选择内部 LSI 振荡器作为 RTC 时钟源，LSI 的精度和功耗指标要差于 LSE。

RTC 闹钟事件能够把 MCU 从停机模式下唤醒，为了实现此功能，需要配置外部中断线 27，并且把 RTC 设置成可产生闹钟事件。而从待机模式下唤醒，仅需把 RTC 设置成可产生闹钟事件。

2.3.6 FLASH 低功耗模式

在进入 FLASH 低功耗模式前，需要先对电源控制寄存器（PWR_CTLR）的 FLASH_LP_REG 位置 1，使能 FLASH 进入低功耗模式，然后对电源控制寄存器（PWR_CTLR）的 FLASH_LP 位进行设置。此时 MCU 的 FLASH 低功耗模式配置完成。

2.4 寄存器描述

表 2-2 PWR 相关寄存器列表

名称	访问地址	描述	复位值
R32_PWR_CTLR	0x40007000	电源控制寄存器	0x00000400
R32_PWR_GSR	0x40007004	电源控制/状态寄存器	0x00000000

2.4.1 电源控制寄存器（PWR_CTLR）

偏移地址：0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved											RAM LV	R18K VBAT	R2K VBAT	R18K STY	R2K STY
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	LD0_EC	AUTO_LD0_EC	FLASH_LP	FLASH_LP_REG	DBP	PLS[2:0]	PVDE	CSBF	CWUF	PDDS	LPDS				

位	名称	访问	描述	复位值
[31:21]	Reserved	RO	保留。	0
20	RAMLV	RW	RAM 工作在低电压模式使能控制位（功耗相对更低）： 1：开启； 0：关闭。	0
19	R18KVBAT	RW	VBAT 供电时，Standby 模式下 18K RAM 电源开关控制位： 1：供电； 0：不供电。	0
18	R2KVBAT	RW	VBAT 供电时，Standby 模式下 2K RAM 电源开关控制位： 1：供电； 0：不供电。	0
17	R18KSTY	RW	Standby 模式下 18K RAM 电源开关控制位： 1：供电； 0：不供电。	0
16	R2KSTY	RW	Standby 模式下 2K RAM 电源开关控制位： 1：供电； 0：不供电。	0
[15:14]	Reserved	RO	保留。	0
13	LD0_EC	RW	电源系统 LD0 节能模式开启： 1：开启节能，约减少 10 μ A； 0：不开启。 注：该位开启时，仅支持最大总电流不超过 2mA	0
12	AUTO_LD0_EC	RW	预置进入停止模式后的 LD0 节能模式：	0

			1: 当进入停止模式后自动节能, 即停止模式 2; 0: 正常 LDO 模式。	
[11:10]	FLASH_LP	RW	配置 FLASH 模式: 00: 空闲模式; X1: FLASH 低功耗模式 0; 10: FLASH 低功耗模式 1。 注: 在进入 STANDBY 模式或 STOP 电压调节器低功耗模式 (PDDS=0, LPDS=1) 前, 此位预置为 FLASH 低功耗模式 1 (FLASH_LP[1:0] = 10b)。	01b
9	FLASH_LP_REG	RW	和 FLASH_LP 字段结合, 软件配置 FLASH 进入低功耗模式的使能: 1: 可以使 FLASH 进入低功耗模式; 0: 无法通过软件使 FLASH 进入低功耗模式。	0
8	DBP	RW	后备区域的写使能。当 RTC 时钟为外部时钟的 128 分频时, 该位必须设置为 1。 1: 允许写 RTC 和后备寄存器; 0: 禁止写 RTC 和后备寄存器。	0
[7:5]	PLS[2:0]	RW	PVD 电压监测阈值设置。详细说明见数据手册中电气特性部分。 000: 上升沿 1.75V/下降沿 1.70V; 001: 上升沿 1.93V/下降沿 1.87V; 010: 上升沿 2.14V/下降沿 2.08V; 011: 上升沿 2.35V/下降沿 2.28V; 100: 上升沿 2.54V/下降沿 2.46V; 101: 上升沿 2.72V/下降沿 2.63V; 110: 上升沿 2.92V/下降沿 2.83V; 111: 上升沿 3.10V/下降沿 3.01V。	000b
4	PVDE	RW	电源电压监测功能使能标志位: 1: 开启电源电压监测功能; 0: 禁止电源电压监测功能。	0
3	CSBF	RW1	清除待机状态标志位, 读出始终为 0。 1: 置 1 清除 SBF 待机状态标志位; 0: 清 0 无效。	0
2	CWUF	RW1	清除唤醒状态标志位, 读出始终为 0。 1: 置 1 后 2 个系统时钟周期后清除 WUF 标志位; 0: 清 0 无效。	0
1	PDDS	RW	掉电深睡眠情景下, 待机/停机模式选择位。 1: 进入待机模式; 0: 进入停机模式, 电压调节器状态由 LPDS 控制。	0
0	LPDS	RW	停机模式下, 电压调节器工作模式选择位。PDDS=0, 该位有效。 1: 电压调节器工作在低功耗模式; 0: 电压调节器工作在正常模式。	0

注: 此寄存器 BIT16~BIT20 只能由 backup 复位, 其他 BIT 从待机模式唤醒时复位。

2.4.2 电源控制/状态寄存器 (PWR_CSR)

偏移地址: 0x04

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

位	名称	访问	描述	复位值
[31:9]	Reserved	RO	保留。	0
8	EWUP	RW	WKUP 引脚使能位： 1: WKUP 强制配置为输入下拉状态，用于把 MCU 从待机状态下唤醒； 0: WKUP 引脚可用于通用 IO，无待机唤醒功能。	0
[7:3]	Reserved	RO	保留。	0
2	PVDO	RO	PVD 输出状态标志位。当 PWR_CTLR 寄存器的 PVDE=1 时，该位有效。 1: VDD 和 VDDA 低于 PLS[2:0] 设定的 PVD 阈值； 0: VDD 和 VDDA 高于 PLS[2:0] 设定的 PVD 阈值。	0
1	SBF	RO	待机状态标志位，可通过 CSBF 位置 1 清除。 1: MCU 进入待机模式； 0: MCU 不在待机模式。	0
0	WUF	RO	唤醒事件状态标志位，可通过 CWUF 位置 1 清除。 1: 在 WKUP 引脚检测到唤醒事件或 RTC 闹钟事件； 0: 没有唤醒事件发生。	0

注：此寄存器从待机模式唤醒后保持不变。

第 3 章 复位和时钟控制（RCC）

控制器根据电源区域的划分以及应用中的外设功耗管理考虑，提供了不同的复位形式以及可配置的时钟树结构。此章节描述了系统中各个时钟的作用域。

3.1 主要特性

- 多种复位形式
- 多路时钟源，总线时钟管理
- 内置外部晶体振荡监测和时钟安全系统
- 各外设时钟独立管理：复位、开启、关闭
- 支持内部时钟输出

3.2 复位

控制器提供了 3 种复位形式：电源复位、系统复位和后备区域复位。

3.2.1 电源复位

电源复位发生时，将复位除了后备区域外的所有寄存器（后备区域由 V_{BAT} 供电）。

其产生条件包括：

- 上电/掉电复位 (POR/PDR 复位)
- 从待机模式下唤醒

3.2.2 系统复位

系统复位发生时，将复位除了控制/状态寄存器 RCC_RSTSCKR 中的复位标志和后备区域外的所有寄存器。通过查看 RCC_RSTSCKR 寄存器中的复位状态标志位识别复位事件来源。

其产生条件包括：

- NRST 引脚上的低电平信号（外部复位）
- 窗口看门狗计数终止 (WWDG 复位)
- 独立看门狗计数终止 (IWDG 复位)
- 软件复位 (SW 复位)
- 低功耗管理复位
- 内核死锁复位
- OPA 复位
- USBPD 复位
- ADC 复位

窗口/独立看门狗复位：由窗口/独立看门狗外设定定时器计数周期溢出触发产生，详细描述看其相应章节。

软件复位：该产品通过可编程中断控制器 PFIC 中的中断配置寄存器 PFIC_CFGR 的 SYSRST 位置 1 复位系统或配置寄存器 PFIC_SCTLR 的 RSTSYS 位置 1 复位系统，具体参考对应章节。

低功耗管理复位：通过将用户选择字节中的 STANBY_RST 位置 0，将启用待机模式复位。这时执行了进入待机模式的过程后，将执行系统复位而不是进入待机模式。通过将用户选择字节中的 STOP_RST 位置 0，将启用停机模式复位。这时执行了进入停机模式的过程后，将执行系统复位而不是进入停机模式。

内核死锁复位：当 PFIC_SCTLR 寄存器的 LOCKUP 为 0 时，内核死锁使能，内核执行异常和 NMI 执行指令时会进入死锁。当 EXTEN_CTR 寄存器 LKUPEN 位启用时，一旦发生 Lock-up 情况，系统将进

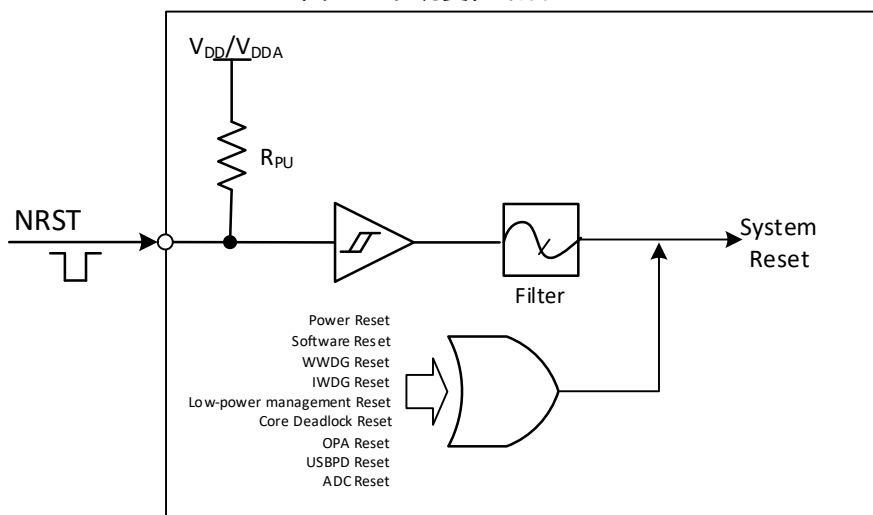
行复位。

OPA 复位：在 OPA 复位使能开启的情况下，运放输出高电平会产生 OPA 复位。

USBPD 复位：当 PD_RST_EN 为 1 时，CH32L103 支持 USB PD 信号帧 Hard Reset 产生的复位；如果 IE_RX_RESET 也为 1，则还支持信号帧 Cable Reset 产生的复位。USB PD 没有复位标志，但产生的复位效果同软件复位。

ADC 复位：在 ADC 看门狗复位使能开启的情况下，当 ADC 数据大于看门狗高阈值或者小于看门狗低阈值时会产生 ADC 复位。

图 3-1 系统复位结构



3.2.3 后备区域复位

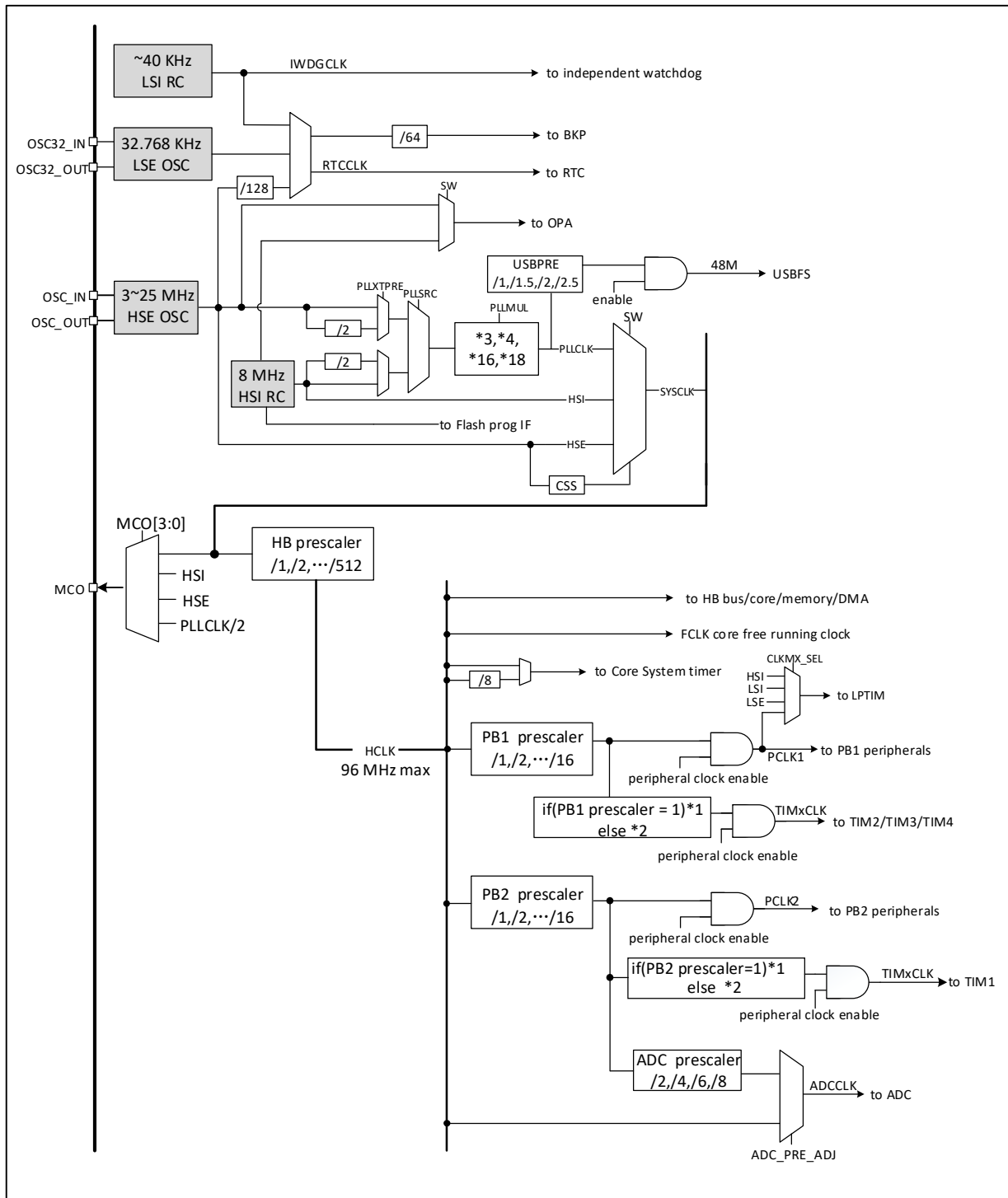
后备区域复位发生时，只会复位后备区域寄存器，包括后备寄存器、RCC_BDCTLR 寄存器（RTC 使能和 LSE 振荡器）。其产生条件包括：

- 在 V_{DD} 和 V_{BAT} 都掉电的前提下，由 V_{DD} 或 V_{BAT} 上电引起
- RCC_BDCTLR 寄存器的 BDRST 位置 1
- RCC_PB1PRSTR 寄存器的 BKPRST 位置 1

3.3 时钟

3.3.1 系统时钟结构

图 3-2 时钟树框图



注：当使用 USB 功能时，CPU 的频率必须是 48MHz、72MHz 或 96MHz。当系统从停止或待机模式唤醒时系统会自动切换为 HSI 做主频。

3.3.2 高速时钟（HSI/HSE）

HSI 是系统内部 8MHz 的 RC 振荡器产生的高速时钟信号。HSI RC 振荡器能够在不需要任何外部器件的条件下提供系统时钟。它的启动时间很短但时钟频率精度较差。HSI 通过设置 RCC_CTLR 寄存

器中的 HSION 位被启动和关闭, HSIRDY 位指示 HSI RC 振荡器是否稳定。系统默认 HSION 和 HSIRDY 置 1 (建议不要关闭)。如果设置了 RCC_INTR 寄存器的 HSIRDYIE 位, 将产生相应中断。

- HSI RC 振荡器可以通过 RCC_CTRL 寄存器将 HSILP 位置 1 进入内部低功耗模式, HSI 使能低功耗模式后, 输出频率从 8MHz 降为 1MHz。
- 出厂校准: 制造工艺的差异会导致每个芯片的 RC 振荡频率不同, 所以在芯片出厂前, 会为每颗芯片进行 HSI 校准。系统复位后, 工厂校准值被装载到 RCC_CTLR 寄存器的 HSICAL[7:0] 中。
- 用户调整: 基于不同的电压或环境温度, 应用程序可以通过 RCC_CTLR 寄存器里的 HSITRIM[4:0] 位来调整 HSI 频率。

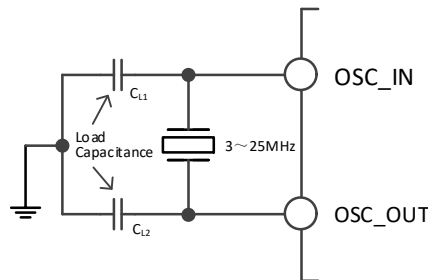
注: 如果 HSE 晶体振荡器失效, HSI 时钟会被作为备用时钟源 (时钟安全系统)。

HSE 是外部的高速时钟信号, 包括外部晶体/陶瓷谐振器产生或者外部高速时钟送入。

HSE 晶体振荡器可以通过 RCC_CTRL 寄存器将 HSELP 位置 1 进入低功耗模式。

- 外部晶体/陶瓷谐振器 (HSE 晶体): 外接 3~25MHz 外部振荡器为系统提供更为精确的时钟源。进一步信息可参考数据手册的电气特性部分。HSE 晶体可以通过设置 RCC_CTLR 寄存器中的 HSEON 位被启动和关闭, HSERDY 位指示 HSE 晶体振荡是否稳定, 硬件在 HSERDY 位置 1 后才将时钟送入系统。如果设置了 RCC_INTR 寄存器的 HSERDYIE 位, 将产生相应中断。

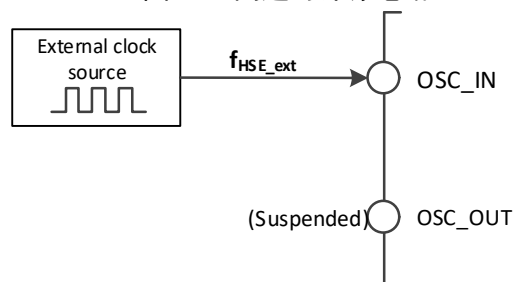
图 3-3 高速外部晶体电路



注: 负载电容需要尽可能地靠近振荡器引脚, 并根据晶体厂家参数选择容值。

- 外部高速时钟源 (HSE 旁路): 此模式从外部直接送入时钟源到 OSC_IN 引脚, OSC_OUT 引脚悬空。最高支持 25MHz 频率。应用程序需在 HSEON 位为 0 情况下, 置位 HSEBYP 位, 打开 HSE 旁路功能, 然后再置位 HSEON 位。

图 3-4 高速时钟源电路



3.3.3 低速时钟 (LSI/LSE)

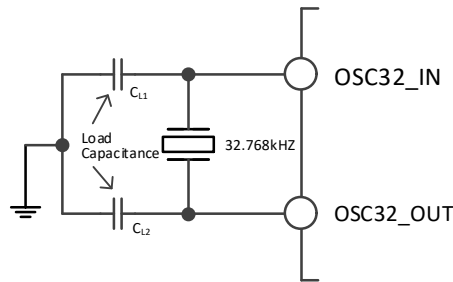
LSI 是系统内部约 40KHz 的 RC 振荡器产生的低速时钟信号。它可以在停机和待机模式下保持运行, 为 RTC 时钟、独立看门狗和唤醒单元提供时钟基准。进一步信息可参考数据手册的电气特性部分。LSI 可以通过设置 RCC_RSTSCKR 寄存器中的 LSION 位被启动和关闭, 然后通过查询 LSIRDY 位检测 LSI RC 振荡是否稳定, 硬件在 LSIRDY 位置 1 后才将时钟送入。如果设置了 RCC_INTR 寄存器的 LSIRDYIE 位, 将产生相应中断。

LSE 是外部的低速时钟信号, 包括外部晶体/陶瓷谐振器产生或者外部低速时钟送入。它为 RTC 时钟或者其他定时功能提供一个低功耗且精确的时钟源。

- 外部晶体/陶瓷谐振器 (LSE 晶体): 外接 32.768KHz 的外部低速振荡器。LSE 通过设置 RCC_BDCTL

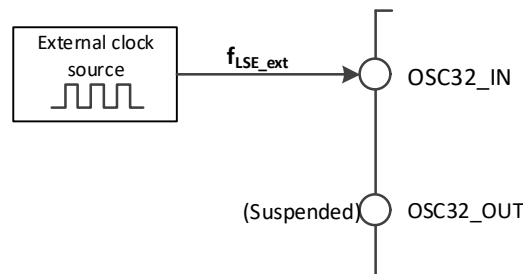
寄存器中的 LSEON 位被启动和关闭, LSERDY 位指示 LSE 晶体振荡是否稳定, 硬件在 LSERDY 位置 1 后才将时钟送入系统。如果设置了 RCC_INTR 寄存器的 LSERDYIE 位, 将产生相应中断。

图 3-5 低速外部晶体电路



- 外部低速时钟源（LSE 旁路）：此模式从外部直接送入时钟源到 OSC32_IN 引脚, OSC32_OUT 引脚悬空。应用程序需在 LSEON 位为 0 情况下, 置位 LSEBYP 位, 打开 LSE 旁路功能, 然后再置位 LSEON 位。

图 3-6 低速时钟源电路



3.3.4 PLL 时钟

通过配置 RCC_CFGR0 寄存器和扩展寄存器 EXTEN_CTR, 内部 PLL 时钟可以选择 3 种时钟来源和倍频系数, 这些设置必须在每个 PLL 被开启前完成, 一旦 PLL 被启动, 这些参数就不能被改动。设置 RCC_CTLR 寄存器中的 PLLON 位被启动和关闭, PLLRDY 位指示 PLL 时钟是否稳定, 硬件在 PLLRDY 位置 1 后才将时钟送入系统。如果设置了 RCC_INTR 寄存器的 PLLRDYIE 位, 将产生相应中断。

PLL 时钟来源:

- HSI 时钟送入
- HSI 经过 2 分频送入
- HSE 时钟送入
- HSE 时钟经过 2 分频送入

3.3.5 总线/外设时钟

3.3.5.1 系统时钟 (SYSCLK)

通过配置 RCC_CFGR0 寄存器 SW[1:0] 位配置系统时钟来源, SWS[1:0] 指示当前的系统时钟源。

- HSI 作为系统时钟
- HSE 作为系统时钟
- PLL 时钟作为系统时钟

控制器复位后, 默认 HSI 时钟被选为系统时钟源。时钟源之间的切换必须在目标时钟源准备就绪后才会发生。

3.3.5.2 HB/PB1/PB2 总线外设时钟 (HCLK/PCLK1/PCLK2)

HB 为高性能总线 (High Performance Bus), 其总线外设时钟为 HCLK; PB1 为外设总线 1 (Peripheral Bus 1), 其总线外设时钟为 PCLK1; PB2 为外设总线 2 (Peripheral Bus 2), 其总线外设时钟为 PCLK2。

通过配置 RCC_CFGR0 寄存器的 HPRE[3:0]、PPRE1[2:0]、PPRE2[2:0]位,可以分别配置 HB、PB1、PB2 总线的时钟。这些总线时钟决定了挂载在其下面的外设接口访问时钟基准。应用程序可以调整不同的数值,来降低部分外设工作时的功耗。

通过 RCC_HBRSTR、RCC_PB1PRSTR、RCC_PB2PRSTR 寄存器中各个位可以复位不同的外设模块,将其恢复到初始状态。

通过 RCC_HBPCENR、RCC_PB1PCENR、RCC_PB2PCENR 寄存器中各个位可以单独开启或关闭不同外设模块通讯时钟接口。使用某个外设时,首先需要开启其时钟使能位,才能访问其寄存器。

3.3.5.3 RTC 时钟 (RTCCLK)

通过设置 RCC_BDCTLR 寄存器的 RTCSEL[1:0]位,RTCCLK 时钟源可以由 HSE/128、LSE 或 LSI 时钟提供。修改此位前要保证电源控制寄存器 (PWR_CR) 中的 DBP 位置 1,只有后备区域复位,才能复位此位。

- LSE 作为 RTC 时钟: 由于 LSE 处于后备域由 V_{BAT} 供电,只要 V_{BAT} 维持供电,尽管 V_{DD} 供电被切断,RTC 仍继续工作。
- LSI 作为 RTC 时钟: 如果 V_{DD} 供电被切断,RTC 自动唤醒不能保证。
- HSE/128 作为 RTC 时钟: 如果 V_{DD} 供电被切断或内部电压调压器被关闭 (1.8V 域的供电被切断),则 RTC 状态不确定。

3.3.5.4 独立看门狗时钟

如果独立看门狗已经由硬件配置设置或软件启动,LSI 振荡器将被强制打开,并且不能被关闭。在 LSI 振荡器稳定后,时钟供应给 IWDG。

3.3.5.5 时钟输出 (MCO)

微控制器允许输出时钟信号到 MCO 引脚。在相应的 GPIO 端口寄存器配置复用推挽输出模式,通过配置 RCC_CFGR0 寄存器 MCO[2:0]位,可以选择以下 4 个时钟信号作为 MCO 时钟输出:

- 系统时钟 (SYSCLK) 输出
- HSI 时钟输出
- HSE 时钟输出
- PLL 时钟经过 2 分频输出

3.3.5.6 USB 时钟

USB 48MHz 时钟来自通过一个可配置的分频器的 PLL 时钟,此时 PLL 支持三种时钟配置,包括 48MHz、72MHz 和 96MHz,通过配置寄存器 RCC_CFGR0 的 USBPRE[1:0]位输出 48MHz 时钟到 USBFS。

3.3.5.7 ADC 时钟

ADC 可以通过 RCC_CFGR0 寄存器来选择输入时钟来源和控制 ADC 时钟的占空比。

在时钟配置寄存器 0 中 ADC_PRE_ADJ 位用来选择输入时钟,当该位置 1 时,ADC 输入时钟选择 HCLK 时钟输入。当该位置 0 时,ADC 输入时钟选择 PLCK2 经过 ADC 分频后 (ADCPRE[1:0]) 的时钟输入,并可以通过 ADC_DUTY_CHG 位相应的配置改变 ADC 时钟的占空比。

3.3.6 时钟安全系统

时钟安全系统是控制器的一种运行保护机制,它可以在 HSE 时钟发送故障的情况下,切换到 HSI 时钟下,并产生中断通知,允许应用程序软件完成营救操作。

通过设置 RCC_CTLR 寄存器的 CSSON 位置 1,激活时钟安全系统。此时,时钟监测器将在 HSE 振荡器启动 (HSERDY=1) 延迟后被使能,并在 HSE 时钟关闭后关闭。一旦系统运行过程中 HSE 时钟发生故障,HSE 振荡器将被关闭,时钟失效事件将被送到高级定时器 (TIM1) 的刹车输入端,并产生时

钟安全中断, CSSF 位置 1, 并且应用程序进入 NMI 不可屏蔽中断, 通过置位 CSSC 位, 可以清除 CSSF 位标志, 可撤销 NMI 中断挂起位。

如果当前 HSE 作为系统时钟, 或者当前 HSE 作为 PLL 输入时钟, PLL 作为系统时钟, 时钟安全系统将在 HSE 故障时自动将系统时钟切换到 HSI 振荡器, 并关闭 HSE 振荡器和 PLL。

3.4 寄存器描述

表 3-1 RCC 相关寄存器列表

名称	访问地址	描述	复位值
R32_RCC_CTLR	0x40021000	时钟控制寄存器	0x0000xx83
R32_RCC_CFGR0	0x40021004	时钟配置寄存器 0	0x00000000
R32_RCC_INTR	0x40021008	时钟中断寄存器	0x00000000
R32_RCC_PB2PRSTR	0x4002100C	PB2 外设复位寄存器	0x00000000
R32_RCC_PB1PRSTR	0x40021010	PB1 外设复位寄存器	0x00000000
R32_RCC_HBPCENR	0x40021014	HB 外设时钟使能寄存器	0x00000014
R32_RCC_PB2PCENR	0x40021018	PB2 外设时钟使能寄存器	0x00000000
R32_RCC_PB1PCENR	0x4002101C	PB1 外设时钟使能寄存器	0x00000000
R32_RCC_BDCTLR	0x40021020	后备域控制寄存器	0x00000000
R32_RCC_RSTSCKR	0x40021024	控制/状态寄存器	0x0C000000
R32_RCC_HBRSTR	0x40021028	HB 外设复位寄存器	0x00000000

3.4.1 时钟控制寄存器 (RCC_CTLR)

偏移地址: 0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved						PLL RDY	PLL ON	Reserved			HSELP	CSSON	HSE BYP	HSE RDY	HSEON
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HSICAL[7:0]								HSITRIM[4:0]				HSILP	HSI RDY	HSION	

位	名称	访问	描述	复位值
[31:26]	Reserved	RO	保留。	0
25	PLLRDY	RO	PLL 时钟就绪锁定标志位 (由硬件置位): 1: PLL 时钟锁定; 0: PLL 时钟未锁定。	0
24	PLLON	RW	PLL 时钟使能控制位: 1: 使能 PLL 时钟; 0: 关闭 PLL 时钟。 注: 进入停止或待机低功耗模式后, 此位由硬件清 0。	0
[23:21]	Reserved	RO	保留。	0
20	HSELP	RW	HSE 低功耗模式: 1: 开启 0: 不开启	1
19	CSSON	RW	时钟安全系统使能控制位:	0

			1: 使能时钟安全系统。当 HSE 准备好 (HSERDY 置 1), 硬件开启对 HSE 的时钟监测功能, 发现 HSE 异常触发 CSSF 标志及 NMI 中断; 当 HSE 没有准备好, 硬件关闭对 HSE 的时钟监测功能。 0: 关闭时钟安全系统。	
18	HSEBYP	RW	外部高速晶体旁路控制位: 1: 旁路外部高速晶体/陶瓷谐振器 (HSEON 位为 0 情况下, 置位 HSEBYP 位, 打开 HSE 旁路功能, 然后再置位 HSEON 位生效); 0: 不旁路高速外部晶体/陶瓷谐振器。	0
17	HSERDY	RO	外部高速晶体振荡稳定就绪标志位 (由硬件置位): 1: 外部高速晶体振荡稳定; 0: 外部高速晶体振荡没有稳定。 注: 在 HSEON 位清 0 后, 该位需要 6 个 HSE 周期清 0。	0
16	HSEON	RW	外部高速晶体振荡使能控制位: 1: 使能 HSE 振荡器; 0: 关闭 HSE 振荡器。 注: 进入停止或待机低功耗模式后, 此位由硬件清 0。	0
[15:8]	HSICAL[7:0]	RO	内部高速时钟校准值, 在系统启动时被自动初始化。	xxh
[7:3]	HSITRIM[4:0]	RW	内部高速时钟调整值: 用户可以输入一个调整值叠加到 HSICAL[7:0] 数值上, 根据电压和温度的变化调整内部 HSI RC 振荡器的频率。 默认值为 16, 可以把 HSI 调整到 8MHz \pm 1%; LP=0 时每步 HSICAL 的变化调整约 20KHz。LP=1 时, 每步 HSICAL 的变化调整约 2.5KHz。	10000b
2	HSILP	RW	HSI 内部低功耗模式: 1: 开启, HSI 频率降低至 1MHz。 0: 不开启 注: HSI 进入内部低功耗模式时要把 0x1FFFF72A 地址的值加载到 HSITRIM[4:0] 里, 实现低功耗模式下的修正	0
1	HSIRDY	RO	内部高速时钟 HSI 稳定就绪标志位 (由硬件置位): 1: 内部高速时钟 HSI 稳定; 0: 内部高速时钟 HSI 没有稳定。 注: 在 HSION 位清 0 后, 该位需要 6 个 HSI 周期清 0。	1
0	HSION	RW	内部高速时钟 HSI 使能控制位: 1: 使能 HSI 振荡器; 0: 关闭 HSI 振荡器。 注: 当从待机和停止模式返回或用作系统时钟的外部振荡器 HSE 发生故障时, 该位由硬件置 1 来启动内部 HSI 振荡器。	1

3.4.2 时钟配置寄存器 0 (RCC_CFGR0)

偏移地址: 0x04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADC_PRE_ADJ	ADC_DUTY_CHG			Reserved	MCO[2:0]			USBPRE[1:0]		PLLMUL[3:0]				PLL_XTPRE	PLL_SRC
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADCPRE[1:0]		PPRE2[2:0]			PPRE1[2:0]			HPRE[3:0]			SWS[1:0]		SW[1:0]		

位	名称	访问	描述	复位值
31	ADC_PRE_ADJ	RW	ADC 输入时钟选择： 1：选择 HCLK 时钟输入； 0：选择 PLCK2 经过 ADC 分频后（ADCPRE[1:0]）的时钟输入。	0
[30:28]	ADC_DUTY_CHG	RW	ADC 时钟占空比控制位： 000：高电平持续时间为 ADC 时钟周期的一半 001：高电平持续时间增加 1 个 HCLK 时钟周期 010：高电平持续时间增加 2 个 HCLK 时钟周期 011：高电平持续时间增加 3 个 HCLK 时钟周期 100：高电平持续时间增加 4 个 HCLK 时钟周期 101：高电平持续时间增加 5 个 HCLK 时钟周期 110：高电平持续时间增加 6 个 HCLK 时钟周期 111：高电平持续时间增加 7 个 HCLK 时钟周期	000
27	Reserved	RO	保留	0
[26:24]	MCO[2:0]	RW	微控制器 MCO 引脚时钟输出控制： 0xx：没有时钟输出； 100：系统时钟 (SYSCLK) 输出； 101：内部振荡器时钟 (HSI) 输出； 110：外部振荡器时钟 (HSE) 输出； 111：PLL 时钟 2 分频后输出；	000b
[23:22]	USBPRE[1:0]	RW	USB 时钟分频配置： 00：1 分频（适用于 PLLCLK=48MHz）； 01：2 分频（适用于 PLLCLK=96MHz）； 10：1.5 分频（适用于 PLLCLK=72MHz）； 11：保留。	00b
[21:18]	PLLMUL[3:0]	RW	PLL 时钟倍频系数（在 PLL 关闭才可写入）： 0000：PLL 2 倍频输出； 0001：PLL 3 倍频输出； 0010：PLL 4 倍频输出； 0011：PLL 5 倍频输出； 0100：PLL 6 倍频输出； 0101：PLL 7 倍频输出； 0110：PLL 8 倍频输出； 0111：PLL 9 倍频输出； 1000：PLL 10 倍频输出； 1001：PLL 11 倍频输出； 1010：PLL 12 倍频输出； 1011：PLL 13 倍频输出； 1100：PLL 14 倍频输出； 1101：PLL 15 倍频输出； 1110：PLL 16 倍频输出； 1111：PLL 18 倍频输出。	0000b
17	PLL_XTPRE	RW	HSE 分频送入 PLL 控制（在 PLL 关闭才可写入）： 1：HSE 2 分频送入 PLL； 0：HSE 不分频送入 PLL。	0

16	PLLSRC	RW	PLL 的输入时钟源（在 PLL 关闭才可写入）： 1：HSE 不分频或 2 分频送入 PLL； 0：HSI 不分频或 2 分频送入 PLL。 注：HSI 时钟的分频使用 EXTEN_CTR 寄存器 HSIPRE 位控制。	0
[15:14]	ADCPRE[1:0]	RW	ADC 时钟来源预分频控制： 00：PCLK2 2 分频后作为 ADC 时钟； 01：PCLK2 4 分频后作为 ADC 时钟； 10：PCLK2 6 分频后作为 ADC 时钟； 11：PCLK2 8 分频后作为 ADC 时钟。 注：ADC 时钟最高不要超过 48MHz。	00b
[13:11]	PPRE2[2:0]	RW	PB2 时钟来源预分频控制： 0xx：HCLK 不分频； 100：HCLK 2 分频； 101：HCLK 4 分频； 110：HCLK 8 分频； 111：HCLK 16 分频。	000b
[10:8]	PPRE1[2:0]	RW	PB1 时钟来源预分频控制： 0xx：HCLK 不分频； 100：HCLK 2 分频； 101：HCLK 4 分频； 110：HCLK 8 分频； 111：HCLK 16 分频。	000b
[7:4]	HPRE[3:0]	RW	HB 时钟来源预分频控制： 0xxx：SYSCLK 不分频； 1000：SYSCLK 2 分频； 1001：SYSCLK 4 分频； 1010：SYSCLK 8 分频； 1011：SYSCLK 16 分频； 1100：SYSCLK 64 分频； 1101：SYSCLK 128 分频； 1110：SYSCLK 256 分频； 1111：SYSCLK 512 分频。	0000b
[3:2]	SWS[1:0]	RO	系统时钟（SYSCLK）状态（硬件置位）： 00：系统时钟源是 HSI； 01：系统时钟源是 HSE； 10：系统时钟源是 PLL； 11：不可用。	00b
[1:0]	SW[1:0]	RW	选择系统时钟来源： 00：HSI 作为系统时钟； 01：HSE 作为系统时钟； 10：PLL 输出作为系统时钟； 11：不可用。 注：在使能了时钟安全系统下（CSSON=1），当从待机和停止模式返回或用作系统时钟的外部振荡器 HSE 发生故障时，由硬件强制选择 HSI 作为系统时钟。	00b

3.4.3 时钟中断寄存器 (RCC_INTR)

偏移地址: 0x08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved								CSSC	Reserved		PLL RDYC	HSE RDYC	HSI RDYC	LSE RDYC	LSI RDYC
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			PLL RDYIE	HSE RDYIE	HSI RDYIE	LSE RDYIE	LSI RDYIE	CSSF	Reserved		PLL RDYF	HSE RDYF	HSI RDYF	LSE RDYF	LSI RDYF

位	名称	访问	描述	复位值
[31:24]	Reserved	RO	保留。	0
23	CSSC	WO	清除时钟安全系统中断标志位 (CSSF) : 1: 清除 CSSF 中断标志; 0: 无动作。	0
[22:21]	Reserved	RO	保留	0
20	PLLRDYC	WO	清除 PLL 就绪中断标志位: 1: 清除 PLLRDYF 中断标志; 0: 无动作。	0
19	HSERDYC	WO	清除 HSE 振荡器就绪中断标志位: 1: 清除 HSERDYF 中断标志; 0: 无动作。	0
18	HSIRDYC	WO	清除 HSI 振荡器就绪中断标志位: 1: 清除 HSIRDYF 中断标志; 0: 无动作。	0
17	LSE RDYC	WO	清除 LSE 振荡器就绪中断标志位: 1: 清除 LSE RDYF 中断标志; 0: 无动作。	0
16	LSIRDYC	WO	清除 LSI 振荡器就绪中断标志位: 1: 清除 LSIRDYF 中断标志; 0: 无动作。	0
[15:13]	Reserved	RO	保留。	0
12	PLLRDYIE	RW	PLL 就绪中断使能位: 1: 使能 PLL 就绪中断; 0: 关闭 PLL 就绪中断。	0
11	HSE RDYIE	RW	HSE 就绪中断使能位: 1: 使能 HSE 就绪中断; 0: 关闭 HSE 就绪中断。	0
10	HSIRDYIE	RW	HSI 就绪中断使能位: 1: 使能 HSI 就绪中断; 0: 关闭 HSI 就绪中断。	0
9	LSE RDYIE	RW	LSE 就绪中断使能位: 1: 使能 LSE 就绪中断; 0: 关闭 LSE 就绪中断。	0

8	LSIRDYIE	RW	LSI 就绪中断使能位： 1：使能 LSI 就绪中断； 0：关闭 LSI 就绪中断。	0
7	CSSF	RO	时钟安全系统中断标志位： 1：HSE 时钟失效，产生了时钟安全中断 CSSI； 0：无时钟安全系统中断。 硬件置位，软件写 CSSC 位 1 清除。	0
[6:5]	Reserved	RO	保留。	0
4	PLLRDYF	RO	PLL 时钟就绪锁定中断标志： 1：PLL 时钟锁定产生中断； 0：无 PLL 时钟锁定中断。 硬件置位，软件写 PLLRDYC 位 1 清除。	0
3	HSERDYF	RO	HSE 时钟就绪中断标志： 1：HSE 时钟就绪产生中断； 0：无 HSE 时钟就绪中断。 硬件置位，软件写 HSERDYC 位 1 清除。	0
2	HSIRDYF	RO	HSI 时钟就绪中断标志： 1：HSI 时钟就绪产生中断； 0：无 HSI 时钟就绪中断。 硬件置位，软件写 HSIRDYC 位 1 清除。	0
1	LSERDYF	RO	LSE 时钟就绪中断标志： 1：LSE 时钟就绪产生中断； 0：无 LSE 时钟就绪中断。 硬件置位，软件写 LSERDYC 位 1 清除。	0
0	LSIRDYF	RO	LSI 时钟就绪中断标志： 1：LSI 时钟就绪产生中断； 0：无 LSI 时钟就绪中断。 硬件置位，软件写 LSIRDYC 位 1 清除。	0

3.4.4 PB2 外设复位寄存器 (RCC_PB2PRSTR)

偏移地址：0x0C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	USART1 RST	Reserved	SPI1 RST	TIM1 RST	Reserved	ADC RST	Reserved			IOPD RST	IOPC RST	IOPB RST	IOPA RST	Reserved	AFIO RST

位	名称	访问	描述	复位值
[31:15]	Reserved	RO	保留。	0
14	USART1RST	RW	USART1 接口复位控制： 1：复位模块； 0：无作用。	0
13	Reserved	RO	保留	0
12	SPI1RST	RW	SPI1 接口复位控制： 1：复位模块； 0：无作用。	0

11	TIM1RST	RW	TIM1 模块复位控制： 1：复位模块； 0：无作用。	0
10	Reserved	RO	保留。	0
9	ADCRST	RW	ADC 模块复位控制： 1：复位模块； 0：无作用。	0
[8:6]	Reserved	RO	保留。	0
5	IOPDRST	RW	IO 的 PD 端口模块复位控制： 1：复位模块； 0：无作用。	0
4	IOPCRST	RW	IO 的 PC 端口模块复位控制： 1：复位模块； 0：无作用。	0
3	IOPBRST	RW	IO 的 PB 端口模块复位控制： 1：复位模块； 0：无作用。	0
2	IOPARST	RW	IO 的 PA 端口模块复位控制： 1：复位模块； 0：无作用。	0
1	Reserved	RO	保留。	0
0	AFIORST	RW	IO 辅助功能模块复位控制： 1：复位模块； 0：无作用。	0

3.4.5 PB1 外设复位寄存器 (RCC_PB1PRSTR)

偏移地址：0x10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPTIM RST	Reserved	PWR RST	BKP RST	Reserved	CAN RST	Reserved	I2C2 RST	I2C1 RST	Reserved	USART 4 RST	USART3 RST	USART2 RST	Reserved		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	SPI2 RST	Reserved	WWDG RST	Reserved								TIM4 RST	TIM3 RST	TIM2 RST	

位	名称	访问	描述	复位值
31	LPTIMRST	RW	LPTIM 模块复位控制 1：复位模块； 0：无作用。	0
[30:29]	Reserved	RO	保留。	0
28	PWRRST	RW	电源接口模块复位控制： 1：复位模块； 0：无作用。	0
27	BKPRST	RW	后备单元复位控制： 1：复位模块； 0：无作用。	0
26	Reserved	RO	保留	0
25	CANRST	RW	CAN 模块复位控制： 1：复位模块； 0：无作用。	0
[24:23]	Reserved	RO	保留。	0
22	I2C2RST	RW	I2C 2 接口复位控制： 1：复位模块； 0：无作用。	0
21	I2C1RST	RW	I2C 1 接口复位控制： 1：复位模块； 0：无作用。	0

20	Reserved	R0	保留	0
19	USART4RST	RW	USART4 接口复位控制： 1：复位模块； 0：无作用。	0
18	USART3RST	RW	USART3 接口复位控制： 1：复位模块； 0：无作用。	0
17	USART2RST	RW	USART2 接口复位控制： 1：复位模块； 0：无作用。	0
[16:15]	Reserved	R0	保留。	0
14	SPI2RST	RW	SPI2 接口复位控制： 1：复位模块； 0：无作用。	0
[13:12]	Reserved	R0	保留。	0
11	WWDGRST	RW	窗口看门狗复位控制： 1：复位模块； 0：无作用。	0
[10:3]	Reserved	R0	保留。	0
2	TIM4RST	RW	定时器 4 模块复位控制： 1：复位模块； 0：无作用。	0
1	TIM3RST	RW	定时器 3 模块复位控制： 1：复位模块； 0：无作用。	0
0	TIM2RST	RW	定时器 2 模块复位控制： 1：复位模块； 0：无作用。	0

3.4.6 HB 外设时钟使能寄存器 (RCC_HBPCENR)

偏移地址：0x14

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved														USBP DEN	Reser ved
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			USBF SEN	Reserved				CRC EN	Reserved			SRAM EN	Rese rved	DMA EN	

位	名称	访问	描述	复位值
[31:18]	Reserved	R0	保留。	0
17	USBPDEN	RW	USBPD 时钟使能位： 1：USBPD 时钟开启； 0：USBPD 时钟关闭。	1
[16:13]	Reserved	R0	保留。	0
12	USBFSEN	RW	USBFS 模块时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
[11:7]	Reserved	R0	保留。	0
6	CRCEN	RW	CRC 模块时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
[5:3]	Reserved	R0	保留。	0
2	SRAMEN	RW	SRAM 接口模块时钟使能位： 1：睡眠模式时，SRAM 接口模块时钟开启； 0：睡眠模式时，SRAM 接口模块时钟关闭。	1

1	Reserved	RO	保留。	0
0	DMAEN	RW	DMA 模块时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0

3.4.7 PB2 外设时钟使能寄存器 (RCC_PB2PCENR)

偏移地址：0x18

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	USART1 EN	Reserved	SPI1 EN	TIM1 EN	Reserved	ADC EN	Reserved			IOPD EN	IOPC EN	IOPB EN	IOPA EN	Reserved	AFIO EN

位	名称	访问	描述	复位值
[31:15]	Reserved	RO	保留。	0
14	USART1EN	RW	USART1 接口时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
13	Reserved	RO	保留	0
12	SPI1EN	RW	SPI1 接口时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
11	TIM1EN	RW	TIM1 模块时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
10	Reserved	RO	保留。	0
9	ADCEN	RW	ADC 模块时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
[8:6]	Reserved	RO	保留。	0
5	IOPDEN	RW	IO 的 PD 端口模块时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
4	IOPCEN	RW	IO 的 PC 端口模块时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
3	IOPBEN	RW	IO 的 PB 端口模块时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
2	IOPAEN	RW	IO 的 PA 端口模块时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
1	Reserved	RO	保留。	0
0	AFIOEN	RW	IO 辅助功能模块时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0

3.4.8 PB1 外设时钟使能寄存器 (RCC_PB1PCENR)

偏移地址：0x1C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPTIM EN	Reserved		PWR EN	BKP EN	Reserved	CAN EN	Reserved		I2C2 EN	I2C1 EN	Reserved	USART 4 EN	USART 3EN	USART 2EN	Reserved

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	SPI2EN	Reserved	Reserved	WWDGEN	Reserved								TIM4EN	TIM3EN	TIM2EN

位	名称	访问	描述	复位值
31	LPTIMEN	RW	LPTIM 模块时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
[31:29]	Reserved	RO	保留。	0
28	PWREN	RW	电源接口模块时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
27	BKPEN	RW	后备单元时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
26	Reserved	RW	保留	0
25	CANEN	RW	CAN 模块时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
[24:23]	Reserved	RO	保留。	0
22	I2C2EN	RW	I2C 2 接口时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
21	I2C1EN	RW	I2C 1 接口时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
20	Reserved	RW	保留	0
19	USART4EN	RW	USART4 接口时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
18	USART3EN	RW	USART3 接口时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
17	USART2EN	RW	USART2 接口时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
[16:15]	Reserved	RO	保留。	0
14	SPI2EN	RW	SPI2 接口时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
[13:12]	Reserved	RO	保留。	0
11	WWDGEN	RW	窗口看门狗时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
[10:3]	Reserved	RO	保留。	0
2	TIM4EN	RW	定时器 4 模块时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
1	TIM3EN	RW	定时器 3 模块时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0
0	TIM2EN	RW	定时器 2 模块时钟使能位： 1：模块时钟开启； 0：模块时钟关闭。	0

3.4.9 后备域控制寄存器 (RCC_BDCTLR)

偏移地址：0x20

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
Reserved															BDRST	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
RTCEN	Reserved					RTCSEL[1:0]		Reserved					LSE BYP	LSE RDY	LSEON	

位	名称	访问	描述	复位值
[31:17]	Reserved	R0	保留。	0
16	BDRST	RW	后备域软件复位控制： 1：复位整个后备域。 0：撤销复位。	0
15	RTCEN	R0	RTC 时钟使能控制： 1：使能 RTC 时钟； 0：关闭 RTC 时钟。 注：RTCSEL!=0 的条件下才可以使能 RTC 时钟，否则硬件强制为 0。	0
[14:10]	Reserved	R0	保留。	0
[9:8]	RTCSEL[1:0]	R0	RTC 时钟源选择： 00：无时钟； 01：LSE 振荡器作为 RTC 时钟； 10：LSI 振荡器作为 RTC 时钟； 11：HSE 振荡器经 128 分频后作为 RTC 时钟。 注：一旦 RTC 时钟源被选定 (RTCEN=1)，直到下次后备域被复位，它不能再被改变。可通过设置 BDRST 位来恢复默认。	0
[7:3]	Reserved	R0	保留。	0
2	LSEBYP	R0	外部低速晶体 (LSE) 旁路控制位： 1：旁路外部低速晶体/陶瓷谐振器 (LSEON 位为 0 情况下，置位 LSEBYP 位，打开 HSE 旁路功能，然后再置位 LSEON 位生效)； 0：不旁路低速外部晶体/陶瓷谐振器。	0
1	LSERDY	R0	外部低速晶体振荡稳定就绪标志位 (由硬件置位)： 1：外部低速晶体振荡稳定； 0：外部低速晶体振荡没有稳定。 注：在 LSEON 位清 0 后，该位需要 6 个 LSE 周期清 0。	0
0	LSEON	R0	外部低速晶体振荡使能控制位： 1：使能 LSE 振荡器； 0：关闭 LSE 振荡器。	0

注：后备域控制寄存器中 (RCC_BDCTL) 的 LSEON、LSEBYP、RTCSEL 和 RTCEN 位处于后备域。因此，这些位在复位后处于写保护状态，只有在电源控制寄存器 (PWR_CR) 中的 DBP 位置 1 后，才能对这些位进行改动。这些位只能由后备域复位清除。任何内部或外部复位都不会影响这些位。

3.4.10 控制/状态寄存器 (RCC_RSTSCKR)

偏移地址：0x24

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPWR RSTF	WWDG RSTF	IWDG RSTF	SFT RSTF	POR RSTF	PIN RSTF	Reser ved	RMVF	Reserved							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved														LSI RDY	LSION

位	名称	访问	描述	复位值
31	LPWRRSTF	R0	低功耗复位标志： 1：发生低功耗复位； 0：无低功耗复位发生。 发生低功耗管理复位时由硬件置 1；软件写 RMVF 位清除。	0
30	WWDGRSTF	R0	窗口看门狗复位标志： 1：发生窗口看门狗复位； 0：无窗口看门狗复位发生。 发生窗口看门狗复位时由硬件置 1；软件写 RMVF 位清除。	0
29	IWDGRSTF	R0	独立看门狗复位标志： 1：发生独立看门狗复位； 0：无独立看门狗复位发生。 发生独立看门狗复位时由硬件置 1；软件写 RMVF 位清除。	0
28	SFTRSTF	R0	软件复位标志： 1：发生软件复位； 0：无软件复位发生。 发生软件复位时由硬件置 1；软件写 RMVF 位清除。	0
27	PORRSTF	R0	上电/掉电复位标志： 1：发生上电/掉电复位； 0：无上电/掉电复位发生。 发生上电/掉电复位时由硬件置 1；软件写 RMVF 位清除。	1
26	PINRSTF	R0	外部手动复位（RST引脚）标志： 1：发生RST引脚复位； 0：无RST引脚复位发生。 在RST引脚复位发生时由硬件置1；软件写RMVF位清除。	0
25	Reserved	R0	保留。	0
24	RMVF	W0	清除复位标志控制： 1：清除复位标志； 0：无作用。	0
[23:2]	Reserved	R0	保留。	0
1	LSIRDY	R0	内部低速时钟（LSI）稳定就绪标志位（由硬件置位）： 1：内部低速时钟（40KHz）稳定； 0：内部低速时钟（40KHz）没有稳定。	0

			注：在 LSION 位清 0 后，该位需要 3 个 LSI 周期清 0。	
0	LSION	RW	内部低速时钟（LSI）使能控制位： 1：使能 LSI（40KHz）振荡器； 0：关闭 LSI（40KHz）振荡器。	0

注：除复位标志只能由上电复位清除，其他由系统复位清除。

3.4.11 HB 外设复位寄存器（RCC_HBRSTR）

偏移地址：0x28

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

位	名称	访问	描述	复位值
[31:18]	Reserved	R0	保留。	0
17	USBPDRST	RW	USBPD 复位控制： 1：复位模块； 0：无作用。	0
[16:13]	Reserved	R0	保留	0
12	USBFSRST	RW	USBFS 模块复位控制： 1：复位模块； 0：无作用。	0
[11:0]	Reserved	R0	保留。	0

第4章 后备寄存器（BKP）

后备寄存器（BKP）提供了10个16位的后备数据寄存器，可以用来存储20字节的用户数据。在主电源（ V_{DD} ）掉电后，这些数据仍可以由 V_{BAT} 供电而保持，不受待机状态、系统复位或电源复位的影响。此外 BKP 单元还提供了侵入检测管理、RTC 时钟校准及脉冲输出功能。

4.1 主要特征

- 提供20字节的后备数据寄存器
- RTC 时钟校准功能
- 在 PC13 引脚上输出 RTC 时钟 64 分频，闹钟脉冲或者秒脉冲
- 侵入检测（TAMPER）功能

4.2 功能说明

微控制器复位后对后备寄存器和 RTC 的访问被禁止，需通过以下操作开启对后备寄存器的访问：

- 1) 置寄存器 RCC_PB1PCENR 的 PWREN 位和 BKPEN 位来打开电源和后备接口的操作时钟；
- 2) 置电源控制寄存器 PWR_CTLR 的 DBP 位，使能对后备寄存器和 RTC 寄存器的访问。

4.2.1 后备数据寄存器

后备数据寄存器可以作为通用数据缓存使用，由于其在 V_{DD} 掉电下靠 V_{BAT} 电源保存数据的特性，可以用来存一些重要的或敏感的数据。但这些数据在产生侵入事件后会被全部清除。

4.2.2 侵入检测

侵入检测就是当外界提供了一个信号（上升沿或下降沿）时，表示有“侵入事件”，硬件将自动清除当前系统中保留的重要信息。这种方式可以增加系统信息的安全性。

当侵入检测引脚上出现跳变沿（取决于 TPAL 位）时会产生一个侵入事件，如果使能了侵入检测中断，还会同时产生一个侵入检测中断。只要出现了侵入事件，后备数据寄存器就会被全部清除。此外，硬件检测采用记忆方式，即使侵入检测功能未开启（TPE=0），系统也会采样是否有跳变沿，并在满足 TPAL 位选择情况下，提前锁定侵入事件，并在 TPE 位置 1 下，触发侵入事件。

例如：当 TPAL=0 时，如果 TPE=0 未开启功能，但 TAMPER 引脚已经为高电平，一旦 TPE=1 后，则会产生一个额外的侵入事件（系统提前锁定了上升沿）。当 TPAL=1 时，如果 TPE=0 未开启功能，但 TAMPER 引脚已经为低电平，一旦 TPE=1 后，则会产生一个额外的侵入事件（系统提前锁定了下降沿）。

所以为了防止发生不必要的侵入事件，导致清除了后备寄存器，建议：在希望硬件检测侵入引脚的开始时刻，通过写 BKP_TPCSR 寄存器 CTE 位置 1，先清除硬件可能记忆过的侵入事件，并确保当前侵入检测引脚状态是无效的。

注：当 V_{DD} 电源断开时，侵入检测功能仍然有效。为了避免不必要的复位数据后备寄存器，TAMPER 引脚应该在片外连接到正确的电平。

4.2.3 RTC 校准

此功能必须配置侵入检测引脚作为普通 I/O 口使用。配置 BKP_TPCTLR 寄存器 TPE 位清 0。

- 脉冲输出

配置 BKP_OCTLR 寄存器的 ASOE 位，开启 RTC 脉冲输出，设置 ASOS 位，选择秒脉冲输出还是闹钟脉冲输出。

- RTC 校准

配置 BKP_OCTLR 寄存器的 COO 位后,内部的 RTC 时钟将经过 64 分频后输出到侵入检测引脚 (TAMPER) 上。通过实际测试,软件配合修改 CAL[6:0]位来调整时钟对 RTC 进行校准。

4.2.4 BKP 接口复位

BKP 区域可以在 V_{DD} 主电源掉电下,由 V_{BAT} 独立供电。应用代码控制 BKP 区域寄存器复位中,后备数据寄存器 BKP_DATAR1-10、ASOS 位、ASOE 位在软件配置 RCC_BDCTL 寄存器的 BDRST 位下复位,不受 RCC 外设接口控制 BKPRST 位影响。

4.3 寄存器描述

表 4-1 BKP 相关寄存器列表

名称	访问地址	描述	复位值
R16_BKP_DATAR1	0x40006C04	后备数据寄存器 1	0x0000
R16_BKP_DATAR2	0x40006C08	后备数据寄存器 2	0x0000
R16_BKP_DATAR3	0x40006C0C	后备数据寄存器 3	0x0000
R16_BKP_DATAR4	0x40006C10	后备数据寄存器 4	0x0000
R16_BKP_DATAR5	0x40006C14	后备数据寄存器 5	0x0000
R16_BKP_DATAR6	0x40006C18	后备数据寄存器 6	0x0000
R16_BKP_DATAR7	0x40006C1C	后备数据寄存器 7	0x0000
R16_BKP_DATAR8	0x40006C20	后备数据寄存器 8	0x0000
R16_BKP_DATAR9	0x40006C24	后备数据寄存器 9	0x0000
R16_BKP_DATAR10	0x40006C28	后备数据寄存器 10	0x0000
R16_BKP_OCTLR	0x40006C2C	RTC 校准寄存器	0x0000
R16_BKP_TPCTLR	0x40006C30	侵入检测控制寄存器	0x0000
R16_BKP_TPCSR	0x40006C34	侵入检测状态寄存器	0x0000

4.3.1 后备数据寄存器 (BKP_DATARx) (x=1-10)

偏移地址: 0x04-0x28

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

位	名称	访问	描述	复位值
[15:0]	D[15:0]	RW	后备数据,可以被用户程序调用。 注:它们仅由后备域复位来复位(BDRST)或(如果侵入检测引脚 TAMPER 功能被开启时)由侵入引脚事件复位。	0

4.3.2 RTC 校准寄存器 (BKP_OCTLR)

偏移地址: 0x2C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						ASOS	ASOE	CCO	CAL[6:0]						

位	名称	访问	描述	复位值
[15:10]	Reserved	RO	保留。	0

9	ASOS	RW	TAMPER 引脚闹钟/秒脉冲输出选择。 1: 输出秒脉冲; 0: 输出闹钟脉冲。 注: 此位只会由后备域复位 (BDRST) 来复位。	0
8	ASOE	RW	TAMPER 引脚使能脉冲输出位 0: 禁止输出闹钟脉冲或者秒脉冲; 1: 使能输出闹钟脉冲或者秒脉冲。 注: 此位只会由后备域复位 (BDRST) 来复位。	0
7	CCO	RW	校准时钟输出选择位 1: TEMPER 引脚输出经 64 分频的 RTC 时钟; 0: 不输出校准时钟。 注 1: 开启此功能必须关闭侵入检测功能。 注 2: 当 V_{DD} 供电断开时, 该位被清除。	0
[6:0]	CAL[6:0]	RW	校准值寄存器, 这个寄存器的值表示在每 2^{20} 个时钟脉冲中有多少个被跳过。这个功能用来校准 RTC 时钟。RTC 时钟可以被减慢 0~121ppm。	0

4.3.3 侵入检测控制寄存器 (BKP_TPCTLR)

偏移地址: 0x30

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

位	名称	访问	描述	复位值
[15:2]	Reserved	RO	保留。	0
1	TPAL	RW	侵入检测引脚 (TEMPER 引脚) 有效电平设置 0: 侵入检测引脚上的高电平会清除所有后备数据寄存器 (硬件锁定上升沿); 1: 侵入检测引脚上的低电平会清除所有后备数据寄存器 (硬件锁定下降沿)。	0
0	TPE	RW	侵入检测引脚使能位 0: TEMPER 引脚做普通 I/O 口用; 1: TEMPER 引脚做侵入检测用。	0

注: 同时将 TPAL 和 TPE 位清除会产生一个假的侵入事件, 推荐只在 TPE 为 0 时才改变 TPAL 位的状态。

4.3.4 侵入检测状态寄存器 (BKP_TPCSR)

偏移地址: 0x34

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						TIF	TEF	Reserved				TPIE	CTI	CTE	

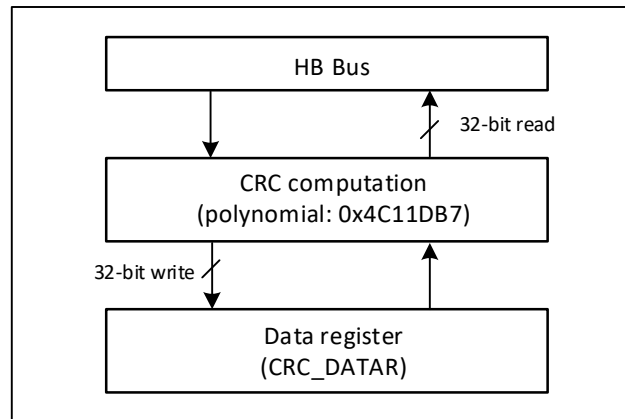
位	名称	访问	描述	复位值
[15:10]	Reserved	RO	保留。	0
9	TIF	RO	侵入中断标志位, 当检测到侵入事件且 TPIE 位置 1 时, 此位会被置位。通过向 CTI 位写 1	0

			来清除此标志位。如果 TPIE 位被复位，那么此位同时也会被复位。 注：仅当系统复位或由待机模式唤醒后才复位该位。	
8	TEF	RO	侵入事件标志位，当检测到侵入事件时，此位会被置位。通过向 CTE 位写 1 会清除此位。 注：当此位为 1 时，所有的 BKP_DATARx 寄存器的值会被清除，且在此位不复位前，所有对 BKP_DATARx 寄存器的写入操作都是无效的。	0
[7:3]	Reserved	RO	保留。	0
2	TPIE	RW	产生侵入中断使能位： 0：禁止侵入检测中断； 1：使能侵入检测中断（TPE 需置 1）。 注 1：侵入中断无法将内核从低功耗模式唤醒。 注 2：仅当系统复位或由待机模式唤醒后才复位该位。	0
1	CTI	WO	侵入检测中断清除位，写 1 清除，读取无效。	0
0	CTE	WO	侵入检测事件清除位，写 1 清除，读取无效。	0

第 5 章 循环冗余校验（CRC）

循环冗余校验 (CRC) 计算单元是根据固定的生成多项式得到任一 32 位数据的 CRC 计算结果。一般用于数据存储和数据通讯领域用来核实数据的正确性。系统提供硬件 CRC 计算单元可以大大节省 CPU 和 RAM 资源提高效率。

图 5-1 CRC 结构框图



5.1 主要特征

- 使用 CRC32 多项式 (0x4C11DB7) : $X^{32}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^8+X^7+X^5+X^4+X^2+X+1$;
- 同一个 32 位寄存器作为数据的输入和 CRC32 计算输出
- 单次转换时间: 4 个 HB 时钟周期 (HCLK)

5.2 功能描述

- CRC 单元复位

如果要开始一次新数据组的 CRC 计算，需要复位 CRC 计算单元。向控制寄存器 CRC_CTLR 的 RST 位写 1，硬件将复位数据寄存器，恢复初始值 0xFFFFFFFF。

- CRC 计算

CRC 单元的计算是前一次 CRC 计算结果和新参与的数据的 CRC 结果。CRC_DATAR 数据寄存器，对其执行写操作将送入新数据到硬件计算单元；执行读取操作，将得到最新一轮的 CRC 计算值。硬件计算时会中断系统的写操作，因此可以连续写入新的值。

注：CRC 单元是对整个 32 位数据进行计算，而不是逐字节计算。

- 独立数据缓冲区

CRC 单元提供了一个 8 位独立数据寄存器 CRC_IDATAR，用于应用代码临时存放 1 字节的数据，不受 CRC 单元复位影响。

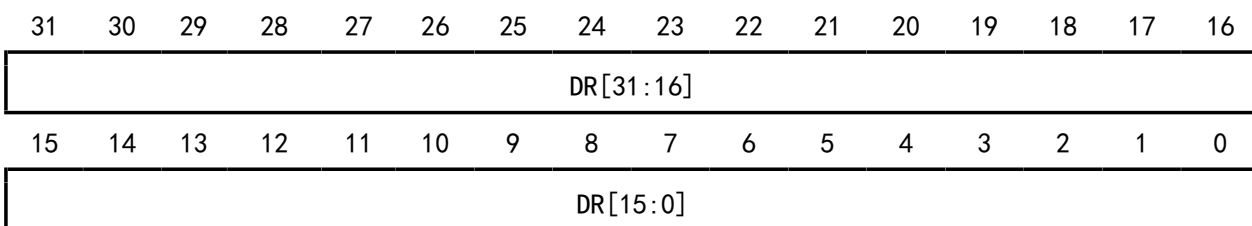
5.3 寄存器描述

表 5-1 CRC 相关寄存器列表

名称	访问地址	描述	复位值
R32_CRC_DATAR	0x40023000	数据寄存器	0xFFFFFFFF
R8_CRC_IDATAR	0x40023004	独立数据缓冲	0x00
R32_CRC_CTLR	0x40023008	控制寄存器	0x00000000

5.3.1 数据寄存器 (CRC_DATAR)

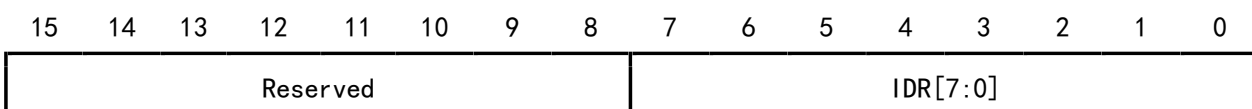
偏移地址: 0x00



位	名称	访问	描述	复位值
[31:0]	DR[31:0]	RW	写入原始数据; 读出计算结果。	0xFFFFFFFF

5.3.2 独立数据缓冲 (CRC_IDATAR)

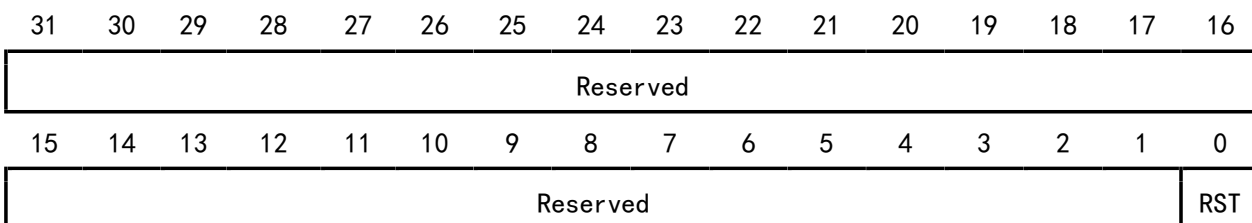
偏移地址: 0x04



位	名称	访问	描述	复位值
[7:0]	IDR[7:0]	RW	8 位通用寄存器, 可以用作数据缓存, 这个寄存器不受控制寄存器的 RST 域影响。	0

5.3.3 控制寄存器 (CRC_CTLR)

偏移地址: 0x08



位	名称	访问	描述	复位值
[31:1]	Reserved	RO	保留。	0
0	RST	WO	CRC 计算单元复位控制, 写 1 执行, 硬件自动清零, 执行完后, 数据寄存器为 0xFFFFFFFF。	0

第 6 章 实时时钟（RTC）

实时时钟（RTC）是一个独立的定时器模块，其可编程计数器最大可达到 32 位，配合软件即可以实现实时时钟功能，并且可以修改计数器的值来重新配置系统的当前时间和日期。RTC 模块在后备供电区域，系统复位和待机模式唤醒对其不造成影响。

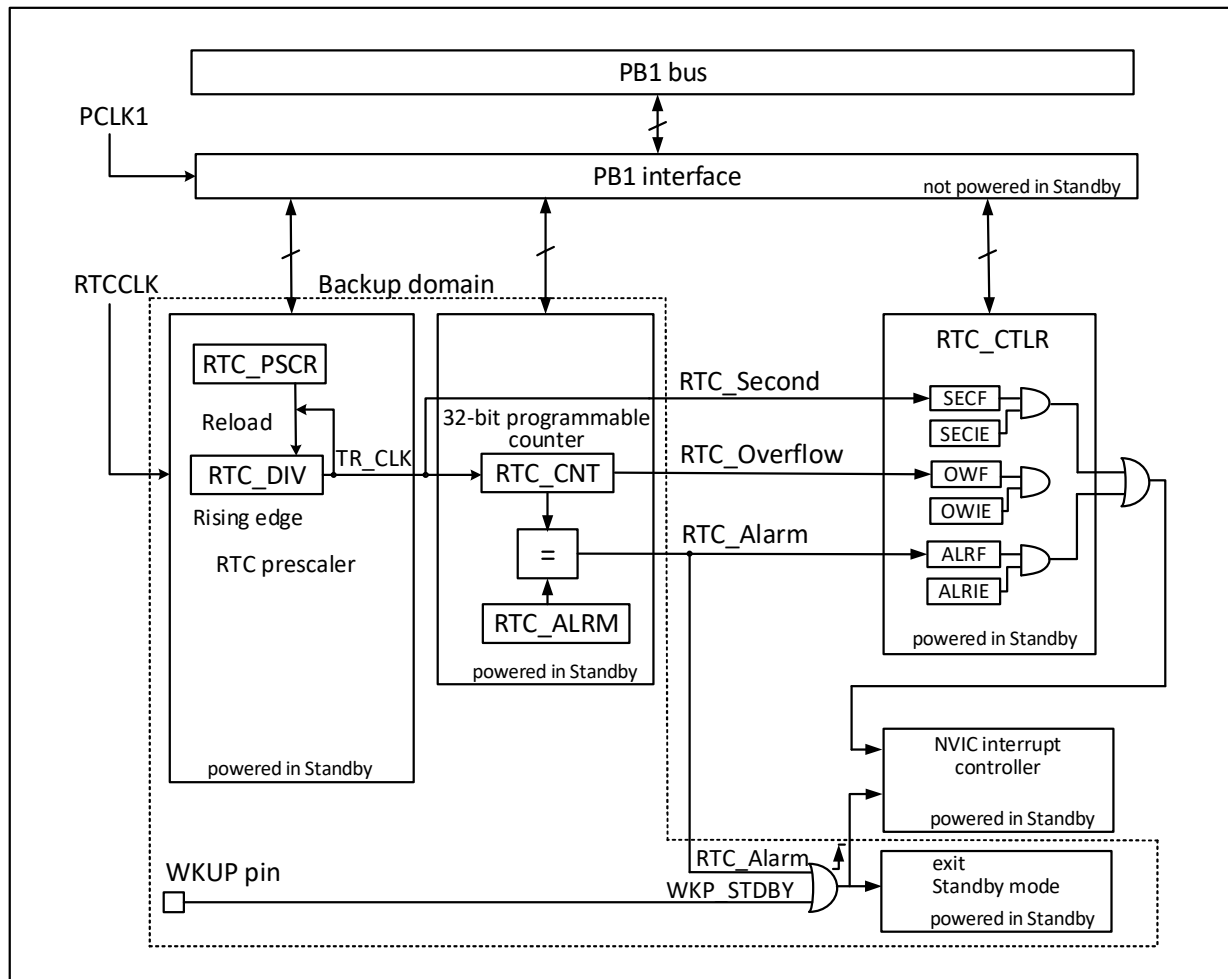
6.1 主要特征

- 最高为 2^{20} 的预分频系数
- 32 位可编程计数器
- 多种时钟源，中断
- 独立复位

6.2 功能描述

6.2.1 概述

图 6-1 RTC 结构框图



由图 6-1 所示，RTC 模块主要是 PB1 总线接口、分频器和计数器、控制和状态寄存器三部分组成，其中分频器和计数器部分在后备区域，可由 V_{BAT} 供电。RTCCLK 输入分频器（RTC_DIV）之后，被分频成 TR_CLK。值得注意的是，分频器（RTC_DIV）的内部是一个自减计数器，自减到溢出就会输出一个 TR_CLK，然后从重装值寄存器（RTC_PSCR）里取出预设值重装到分频器里，读分频器实际上是读取它的实时值（read only），写分频系数应该写到重装值寄存器（RTC_PSCR）里。一般 TR_CLK

的周期被设置为 1 秒，TR_CLK 会触发秒事件，同时会使主计数器（RTC_CNT）自增 1；当主计数器增加到和闹钟寄存器的值一致时，会触发闹钟事件；当主计数器自增到溢出时，会触发溢出事件。以上三种事件都可以触发中断，并对应相应中断使能位控制。

6.2.2 复位

由于实时时钟的特殊用途，其处于后备域的四组寄存器：预分频，预分频重装值，主计数器和闹钟，只能通过后备域的复位信号复位，参照 RCC 的后备域复位章节。实时时钟的控制寄存器受系统复位或电源复位控制。

6.2.3 较特别的读写寄存器操作

由于实时时钟的特殊用处，RTC 和 PB1 总线是独立的，PB1 对 RTC 的读取不一定是实时的，通过 PB1 读取 RTC 的寄存器必须在 PB1 启动后并经过了一个 RTC 上升沿，这种情形可能出现在系统复位和电源复位之后、从待机或者停机模式唤醒后。方便的做法是等待控制寄存器（CTLR）的 RSF 位被置高。对 RTC 的写操作必须等上一个写操作结束，且必须进入配置模式，具体的步骤为：

- 1) 查询 RTOFF 位，直到其变为 1；
- 2) 置 CNF 位，进入配置模式；
- 3) 对一个或者多个 RTC 寄存器进行写操作；
- 4) 清 CNF 位，退出配置模式，PB1 接口开始对 RTC 寄存器进行写入；
- 5) 查询 RTOFF 位，直到其变为 1 即为写完；

6.3 寄存器描述

表 6-1 RTC 相关寄存器列表

名称	访问地址	描述	复位值
R16_RTC_CTLRH	0x40002800	RTC 控制寄存器高位	0x0000
R16_RTC_CTLRL	0x40002804	RTC 控制寄存器低位	0x0020
R16_RTC_PSCRH	0x40002808	预分频器重装值寄存器高位	0x000X
R16_RTC_PSCRL	0x4000280C	预分频器重装值寄存器低位	0xFFFF
R16_RTC_DIVH	0x40002810	分频器寄存器高位	0x000X
R16_RTC_DIVL	0x40002814	分频器寄存器低位	0xFFFF
R16_RTC_CNTH	0x40002818	RTC 计数器高位	0xFFFF
R16_RTC_CNTL	0x4000281C	RTC 计数器低位	0xFFFF
R16_RTC_ALRMH	0x40002820	闹钟寄存器高位	0xFFFF
R16_RTC_ALRML	0x40002824	闹钟寄存器低位	0xFFFF

6.3.1 RTC 控制寄存器高位（RTC_CTLRH）

偏移地址：0x00

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													OWIE	ALRIE	SECI E

位	名称	访问	描述	复位值
[15:3]	Reserved	RO	保留。	0
2	OWIE	RW	溢出中断使能位。	0
1	ALRIE	RW	闹钟中断使能位。	0
0	SECI E	RW	秒中断使能位。	0

6.3.2 RTC 控制寄存器低位 (RTC_GTLRL)

偏移地址: 0x04

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										RTOFF	CNF	RSF	OWF	ALRF	SECF

位	名称	访问	描述	复位值
[15:6]	Reserved	RO	保留。	0
5	RTOFF	RO	RTC 操作状态指示位, 表示对 RTC 的最后一次操作的执行状态, 对 RTC 的操作必须等待此位为 1。 1: 上一次对 RTC 的操作已经完成; 0: 上一次对 RTC 的操作还在进行中。	1
4	CNF	RW	配置标志位, 将此位写 1 进入配置模式, 从而允许向计数器 (R16_RTC_CNTx)、闹钟寄存器 (R16_RTC_ALRMx) 和预分频器重装值寄存器 (R16_RTC_PSCRx) 写入值. 只有将该位写 1 并重新被软件清 0 后才会执行写的操作: 1: 进入配置模式; 0: 退出配置模式, 开始更新 RTC 寄存器。	0
3	RSF	RWO	寄存器同步标志位, 在对 RTC 模块的预分频 (PSCRx)、闹钟 (ALRMx)、计数器 (CNTx) 这些寄存器进行读写前, 都要先保证这个位已经被硬件置位, 以确定这些寄存器已经被同步; 在进行读写这些寄存器时, 或者 PB1 复位或 PB1 时钟停止后, 第一步应该将此位复位。 1: 寄存器已被同步; 0: 寄存器未被同步。	0
2	OWF	RWO	计数器溢出标志, 当 32 位计数器溢出时, 此位由硬件置位。如果置位了 OWIE 位, 还会产生一个溢出中断。此位只能由软件清零, 不能被软件置位。	0
1	ALRF	RWO	闹钟标志, 当计数器的值达到闹钟寄存器 (ALRMx) 的值, 此位会被硬件置位, 如果闹钟中断使能位 (ALRIE) 置位, 还会产生一个闹钟中断。此位只能由软件清零, 不能被软件置位。	0
0	SECF	RWO	秒事件标志, 当时钟经过预分频器分频后每产生一个下降沿, 就会使计数器自增一, 同时产生一个秒事件, 此位会被置位, 如果秒中断被使能 (SECIE 被置位), 同时还会产生一个秒中断。此位只能由软件清零, 不能被软件置位。	0

6.3.3 预分频器重装值寄存器高位 (RTC_PSCRH)

偏移地址: 0x08

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

Reserved	PRL[19:16]
----------	------------

位	名称	访问	描述	复位值
[15:4]	Reserved	RO	保留。	0
[3:0]	PRL[19:16]	WO	重装值高位。	x

6.3.4 预分频器重装值寄存器低位 (RTC_PSCRL)

偏移地址: 0x0C

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

PRL[15:0]

位	名称	访问	描述	复位值
[15:0]	PRL[15:0]	WO	重装值低位。实际的分频系数就是 (PRL[19:0]+1)，比如如果 RTC 输入频率为 32768Hz，那么这个值设为 0x7fff 就可以分频出 1 秒周期的信号。	xxxxh

6.3.5 分频器寄存器高位 (RTC_DIVH)

偏移地址: 0x10

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

Reserved	DIV[19:16]
----------	------------

位	名称	访问	描述	复位值
[15:4]	Reserved	RO	保留。	0
[3:0]	DIV[19:16]	RO	分频器寄存器高位。	x

6.3.6 分频器寄存器低位 (RTC_DIVL)

偏移地址: 0x14

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

DIV[15:0]

位	名称	访问	描述	复位值
[15:0]	DIV[15:0]	RO	分频器寄存器低位。DIV 实际上是一个自减计数器，RTC_CLK 每来一个时钟 DIV 计数器就会减 1，溢出后就会输出一个 TR_CLK，同时从 PSCR 中重装值。DIV 只能读取，读出的是当前分频器的计数器的剩余值。	xxxxh

6.3.7 RTC 计数器高位 (RTC_CNTH)

偏移地址: 0x18

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

CNT[31:16]

位	名称	访问	描述	复位值
[15:0]	CNT[31:16]	RW	计数器高位。	xxxxh

6.3.8 RTC 计数器低位 (RTC_CNTL)

偏移地址: 0x1C

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

位	名称	访问	描述	复位值
[15:0]	CNT[15:0]	RW	计数器低位, RTC 定时器的核心器件, 由 TRCLK (周期一般设为 1 秒) 提供时钟。通过读取 CNT[31:0] 来计算出当前的时间。写这个值需要进入配置模式。	xxxxh

6.3.9 闹钟寄存器高位 (RTC_ALRMH)

偏移地址: 0x20

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ALR[31:16]															

位	名称	访问	描述	复位值
[15:0]	ALR[31:16]	WO	闹钟寄存器高位。	xxxxh

6.3.10 闹钟寄存器低位 (RTC_ALRML)

偏移地址: 0x24

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

位	名称	访问	描述	复位值
[15:0]	ALR[15:0]	WO	闹钟寄存器低位。当闹钟寄存器 ALRM[31:0] 的值和计数器 CNT[31:0] 的值一致时会产生一个闹钟事件。更改这个值需要进入配置模式。	xxxxh

第 7 章 独立看门狗（IWDG）

系统设有独立看门狗（IWDG）用来检测逻辑错误和外部环境干扰引起的软件故障。IWDG 时钟源来自于 LSI，可独立于主程序之外运行，适用于对精度要求低的场合。

7.1 主要特征

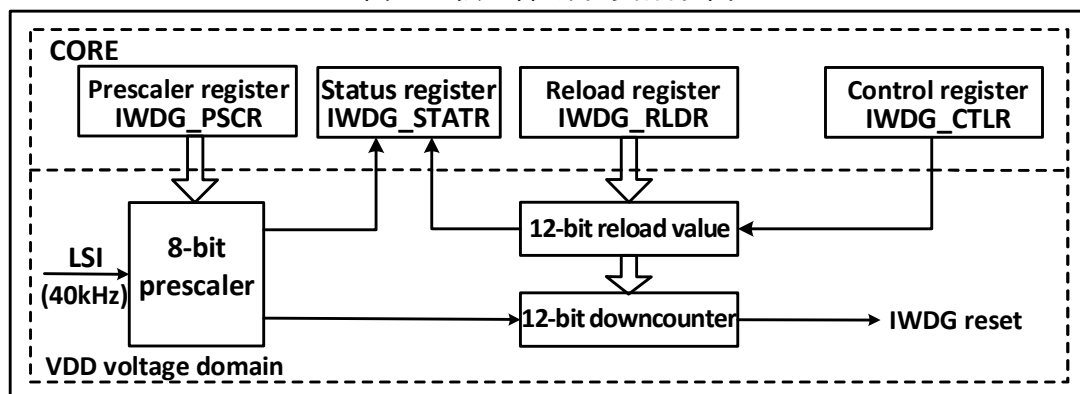
- 12 位自减型计数器
- 时钟来源 LSI 分频，可以在低功耗模式下运行
- 复位条件：计数器值减到 0

7.2 功能说明

7.2.1 原理和用法

独立看门狗的时钟来源 LSI 时钟，其功能在停机和待机模式时仍能正常工作。当看门狗计数器自减到 0 时，将会产生系统复位，所以超时时间为（重装载值+1）个时钟。

图 7-1 独立看门狗的结构框图



● 启动独立看门狗

系统复位后，看门狗处于关闭状态，向 IWDG_CTLR 寄存器写 0xCCCC 开启看门狗，随后它不能再被关闭，除非发生复位。

如果在用户选择字开启了硬件独立看门狗使能位（IWDG_SW），在微控制器复位后将固定开启 IWDG。

● 看门狗配置

看门狗内部是一个递减运行的 12 位计数器，当计数器的值减为 0 时，将发生系统复位。开启 IWDG 功能，需要执行下面几点操作：

- 1) 计数时基：IWDG 时钟来源 LSI，通过 IWDG_PSCR 寄存器设置 LSI 分频值时钟作为 IWDG 的计数时基。操作方法先向 IWDG_CTLR 寄存器写 0x5555，再修改 IWDG_PSCR 寄存器中的分频值。IWDG_STATR 寄存器中的 PVU 位指示了分频值更新状态，在更新完成的情况下才可以进行分频值的修改和读出。
- 2) 重装载值：用于更新独立看门狗中计数器当前值，并且计数器由此值进行递减。操作方法先向 IWDG_CTLR 寄存器写 0x5555，再修改 IWDG_RLDR 寄存器设置目标重装载值。IWDG_STATR 寄存器中的 RUV 位指示了重装载值更新状态，在更新完成的情况下才可以进行 IWDG_RLDR 寄存器的修改和读出。
- 3) 看门狗使能：向 IWDG_CTLR 寄存器写 0xCCCC，即可开启看门狗功能。
- 4) 喂狗：即在看门狗计数器递减到 0 前刷新当前计数器值防止发生系统复位。向 IWDG_CTLR 寄存

器写 0xAAAA，让硬件将 IWDG_RLDR 寄存值更新到看门狗计数器中。此动作需要在看门狗功能开启后定时执行，否则会出现看门狗复位动作。

7.2.2 调试模式

系统进入调试模式时，可以由调试模块寄存器配置 IWDG 的计数器继续工作或停止。

7.3 寄存器描述

表 7-1 IWDG 相关寄存器列表

名称	访问地址	描述	复位值
R16_IWDG_CTLR	0x40003000	控制寄存器	0x0000
R16_IWDG_PSCR	0x40003004	分频因子寄存器	0x0000
R16_IWDG_RLDR	0x40003008	重装载值寄存器	0x0FFF
R16_IWDG_STATR	0x4000300C	状态寄存器	0x0000

7.3.1 IWDG 控制寄存器（IWDG_CTLR）

偏移地址：0x00

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

位	名称	访问	描述	复位值
[15:0]	KEY[15:0]	WO	操作键值锁。 0xAAAA：喂狗。加载 IWDG_RLDR 寄存器值到独立看门狗计数器中； 0x5555：允许修改 R16_IWDG_PSCR 和 R16_IWDG_RLDR 寄存器； 0xCCCC：启动看门狗，如果启用了硬件看门狗（用户选择字配置）则不受这个限制。	0

7.3.2 分频因子寄存器（IWDG_PSCR）

偏移地址：0x04

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

位	名称	访问	描述	复位值
[15:3]	Reserved	RO	保留。	0
[2:0]	PR[2:0]	RW	IWDG 时钟分频系数，修改此域前要向 KEY 中写 0x5555。 000：4 分频； 001：8 分频； 010：16 分频； 011：32 分频； 100：64 分频； 101：128 分频； 110：256 分频； 111：保留。 IWDG 计数时基=LSI/分频系数。 注：读该域值前，要确保 IWDG_STATR 寄存器	000b

			中的 PVU 位为 0，否则读出值无效。	
--	--	--	----------------------	--

7.3.3 重装载值寄存器 (IWDG_RLDR)

偏移地址: 0x08

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

位	名称	访问	描述	复位值
[15:12]	Reserved	R0	保留。	0
[11:0]	RL[11:0]	RW	计数器重装载值。修改此域前要向 KEY 中写 0x5555。 当向 KEY 中写 0xAAAA 后，此域的值将会被硬件装载到计数器中，随后计数器从这个值开始递减计数。 注：读写该域值前，要确保 IWDG_STATR 寄存器中的 RVU 位为 0，否则读写此域无效。	FFFh

注：此寄存器在待机模式下会被复位。

7.3.4 状态寄存器 (IWDG_STATR)

偏移地址: 0x0C

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

位	名称	访问	描述	复位值
[15:2]	Reserved	R0	保留。	0
1	RVU	R0	重装值更新标志位。硬件置位或清 0。 1：重装载值更新正在进行中； 0：重装载更新结束（最多 5 个 LSI 周期）。 注：重装载值寄存器 IWDG_RLDR 只有在 RVU 位被清 0 后才可读写访问。	0
0	PVU	R0	时钟分频系数更新标志位。硬件置位或清 0。 1：时钟分频值更新正在进行中； 0：时钟分频值更新结束（最多 5 个 LSI 周期）。 注：分频因子寄存器 IWDG_PSCR 只有在 PVU 位被清 0 后才可读写访问。	0

注：在预分频或重装值更新后，不必等待 RVU 或 PVU 复位，可继续执行下面的代码。（即使在低功耗模式下，此写操作仍会被继续执行完成。）

第 8 章 窗口看门狗（WWDG）

窗口看门狗一般用来监测系统运行的软件故障，例如外部干扰、不可预见的逻辑错误等情况。它需要在一个特定的窗口时间（有上下限）内进行计数器刷新（喂狗），否则早于或者晚于这个窗口时间看门狗电路都会产生系统复位。

8.1 主要特征

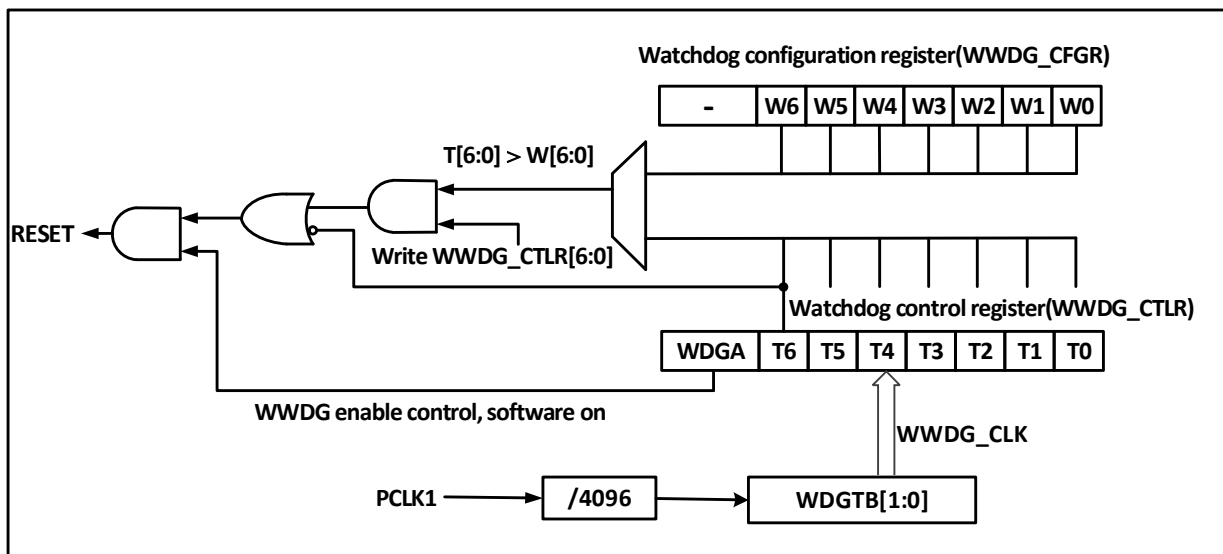
- 可编程的 7 位自减型计数器
- 双条件复位：当前计数器值小于 0x40，或者计数器值在窗口时间外被重装载
- 唤醒提前通知功能（EWI），用于及时喂狗动作防止系统复位

8.2 功能说明

8.2.1 原理和用法

窗口看门狗运行基于一个 7 位的递减计数器，其挂载在 PB1 总线下，计数时基 WWDG_CLK 来源（PCLK1/4096）时钟的分频，分频系数在配置寄存器 WWDG_CFGR 中的 WDG TB[1:0] 域设置。递减计数器处于自由运行状态，无论看门狗功能是否开启，计数器一直循环递减计数。如图 8-1 所示，窗口看门狗内部结构框图。

图 8-1 窗口看门狗结构框图



● 启动窗口看门狗

系统复位后，看门狗处于关闭状态，设置 WWDG_CTLR 寄存器的 WDGA 位能够开启看门狗，随后它不能再被关闭，除非发生复位。

注：可以通过设置 RCC_PB1PCENR 寄存器关闭 WWDG 的时钟来源，暂停 WWDG_CLK 计数，间接停止看门狗功能，或者通过设置 RCC_PB1PRSTR 寄存器复位 WWDG 模块，等效为复位的作用。

● 看门狗配置

看门狗内部是一个不断循环递减运行的 7 位计数器，支持读写访问。使用看门狗复位功能，需要执行下面几点操作：

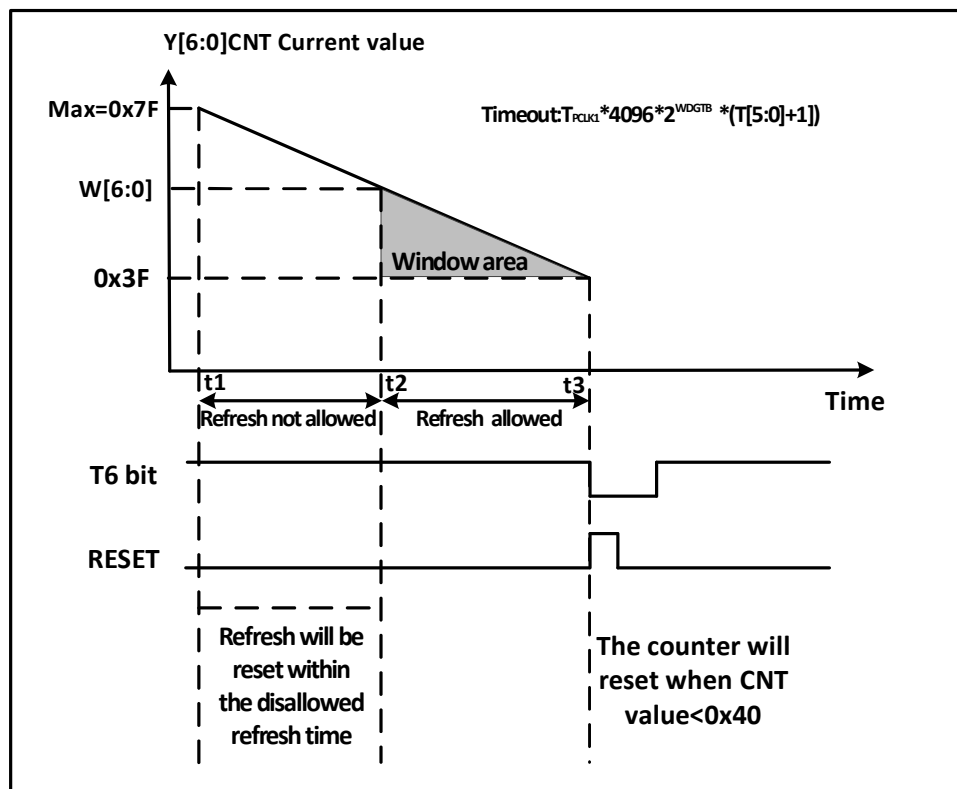
- 1) 计数时基：通过 WWDG_CFGR 寄存器的 WDG TB[1:0] 位域，注意要开启 RCC 单元的 WWDG 模块时钟。
- 2) 窗口计数器：设置 WWDG_CFGR 寄存器的 W[6:0] 位域，此计数器由硬件用作和当前计数器比较使用，数值由用户软件配置，不会改变。作为窗口时间的上限值。

- 3) 看门狗使能：WWDG_CTLR 寄存器 WDGA 位软件置 1，开启看门狗功能，可以系统复位。
- 4) 喂狗：即刷新当前计数器值，配置 WWDG_CTLR 寄存器的 T[6:0] 位域。此动作需要在看门狗功能开启后，在周期性的窗口时间内执行，否则会出现看门狗复位动作。

● 喂狗窗口时间

如图 8-2 所示，灰色区域为窗口看门狗的监测窗口区域，其上限时间 t_2 对应当前计数器值达到窗口值 $W[6:0]$ 的时间点；其下限时间 t_3 对应当前计数器值达到 $0x3F$ 的时间点。此区域时间内 $t_2 < t < t_3$ 可以进行喂狗操作（写 T[6:0]），刷新当前计数器的数值。

图 8-2 窗口看门狗的计数模式



● 看门狗复位：

- 1) 当没有及时喂狗操作，导致 T[6:0] 计数器的值由 $0x40$ 变成 $0x3F$ ，将出现“窗口看门狗复位”，产生系统复位。即 T6-bit 被硬件检测为 0，将出现系统复位。

注：应用程序可以通过软件写 T6-bit 为 0，实现系统复位，等效软件复位功能。

- 2) 当在不允许喂狗时间内执行计数器刷新动作，即在 $t_1 \leq t \leq t_2$ 时间内操作写 T[6:0] 位域，将出现“窗口看门狗复位”，产生系统复位。

● 提前唤醒

为了防止没有及时刷新计数器导致系统复位，看门狗模块提供了早期唤醒中断（EWI）通知。当计数器自减到 $0x40$ 时，产生提前唤醒信号，WEIF 标志置 1，如果置位了 EWI 位，会同时触发窗口看门狗中断。此时距离硬件复位有 1 个计数器时钟周期（自减为 $0x3F$ ），应用程序可在此时间内即时进行喂狗操作。

8.2.2 调试模式

系统进入调试模式时，可以由调试模块寄存器配置 WWDG 的计数器继续工作或停止。

8.3 寄存器描述

表 8-1 WWDG 相关寄存器列表

名称	访问地址	描述	复位值
R16_WWDG_CTLR	0x40002C00	控制寄存器	0x007F
R16_WWDG_CFGR	0x40002C04	配置寄存器	0x007F
R16_WWDG_STATR	0x40002C08	状态寄存器	0x0000

8.3.1 WWDG 控制寄存器 (WWDG_CTLR)

偏移地址: 0x00

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

位	名称	访问	描述	复位值
[15:8]	Reserved	R0	保留。	0
7	WDGA	RW1	窗口看门狗复位使能位。 1: 开启看门狗功能 (可产生复位信号); 0: 禁止看门狗功能。 软件写 1 开启, 但是只允许复位后硬件清 0。	0
[6:0]	T[6:0]	RW	7 位自减计数器, 每 4096×2^{WDGTB} 个 PCLK1 周期自减 1。当计数器从 0x40 自减到 0x3F 时, 即 T6 跳变为 0 时, 产生看门狗复位。	7Fh

8.3.2 WWDG 配置寄存器 (WWDG_CFGR)

偏移地址: 0x04

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							EWI	WDGTB[1:0]	W[6:0]						

位	名称	访问	描述	复位值
[15:10]	Reserved	R0	保留。	0
9	EWI	RW1	提前唤醒中断使能位。 若此位置 1, 则在计数器的值达到 0x40 时产生中断。此位只能在复位后由硬件清 0。	0
[8:7]	WDGTB[1:0]	RW	窗口看门狗时钟分频选择: 00: 1 分频, 计数时基 = PCLK1/4096; 01: 2 分频, 计数时基 = PCLK1/4096/2; 10: 4 分频, 计数时基 = PCLK1/4096/4; 11: 8 分频, 计数时基 = PCLK1/4096/8。	00b
[6:0]	W[6:0]	RW	窗口看门狗 7 位窗口值。用来与计数器的值做比较。喂狗操作只能在计数器的值小于窗口值且大于 0x3F 时进行。	7Fh

8.3.3 WWDG 状态寄存器 (WWDG_STATR)

偏移地址：0x08

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

位	名称	访问	描述	复位值
[15:1]	Reserved	WO	保留。	0
0	EWIF	RW0	提前唤醒中断标志位。 当计数器到达 0x40 时，此位会被硬件置位，必须通过软件清 0，用户置位是无效的。即使 EWI 未被置位，此位在事件发生时仍会照常被置位。	0

第9章 中断和事件（PFIC）

内置可编程快速中断控制器（PFIC - Programmable Fast Interrupt Controller），最多支持 255 个中断向量。当前系统管理了 52 个外设中断通道和 4 个内核中断通道，其他保留。

9.1 主要特征

9.1.1 PFIC 控制器

- 52 个外设中断，每个中断请求都有独立的触发和屏蔽控制位，有专用的状态位
- 可编程多级中断嵌套，最大嵌套深度 2 级，硬件压栈深度 2 级
- 特有快速中断进出机制，硬件自动压栈和恢复，无需指令开销
- 特有免表 VTF（Vector Table Free）中断响应机制，4 路可编程直达中断向量地址

9.2 系统定时器

内核自带了一个 64 位加减计数器（SysTick），支持 HCLK 或者 HCLK/8 作为时基，具有较高优先级，校准后可用于时间基准。

9.3 中断和异常的向量表

表 9-1 向量表

编号	优先级	类型	名称	描述	入口地址
0	-	-	-	-	0x00000000
1	-	-	-	-	0x00000004
2	-5	固定	NMI	不可屏蔽中断	0x00000008
3	-4	固定	HardFault	异常中断	0x0000000C
4	-	-	-	保留	0x00000010
5	-3	固定	Ecall-M	机器模式回调中断	0x00000014
6-7	-	-	-	保留	0x00000018- 0x0000001C
8	-2	固定	Ecall-U	用户模式回调中断	0x00000020
9	-1	固定	BreakPoint	断点回调中断	0x00000024
10-11	-	-	-	保留	0x00000028- 0x0000002C
12	0	可编程	SysTick	系统定时器中断	0x00000030
13	-	-	-	保留	0x00000034
14	1	可编程	SW	软件中断	0x00000038
15	-	-	-	保留	0x0000003C
16	2	可编程	WWDG	窗口定时器中断	0x00000040
17	3	可编程	PVD	电源电压检测中断（EXTI）	0x00000044
18	4	可编程	TAMPER	侵入检测中断	0x00000048
19	5	可编程	RTC	实时时钟中断	0x0000004C
20	6	可编程	FLASH	闪存全局中断	0x00000050
21	7	可编程	RCC	复位和时钟控制中断	0x00000054
22	8	可编程	EXTI0	EXTI 线 0 中断	0x00000058

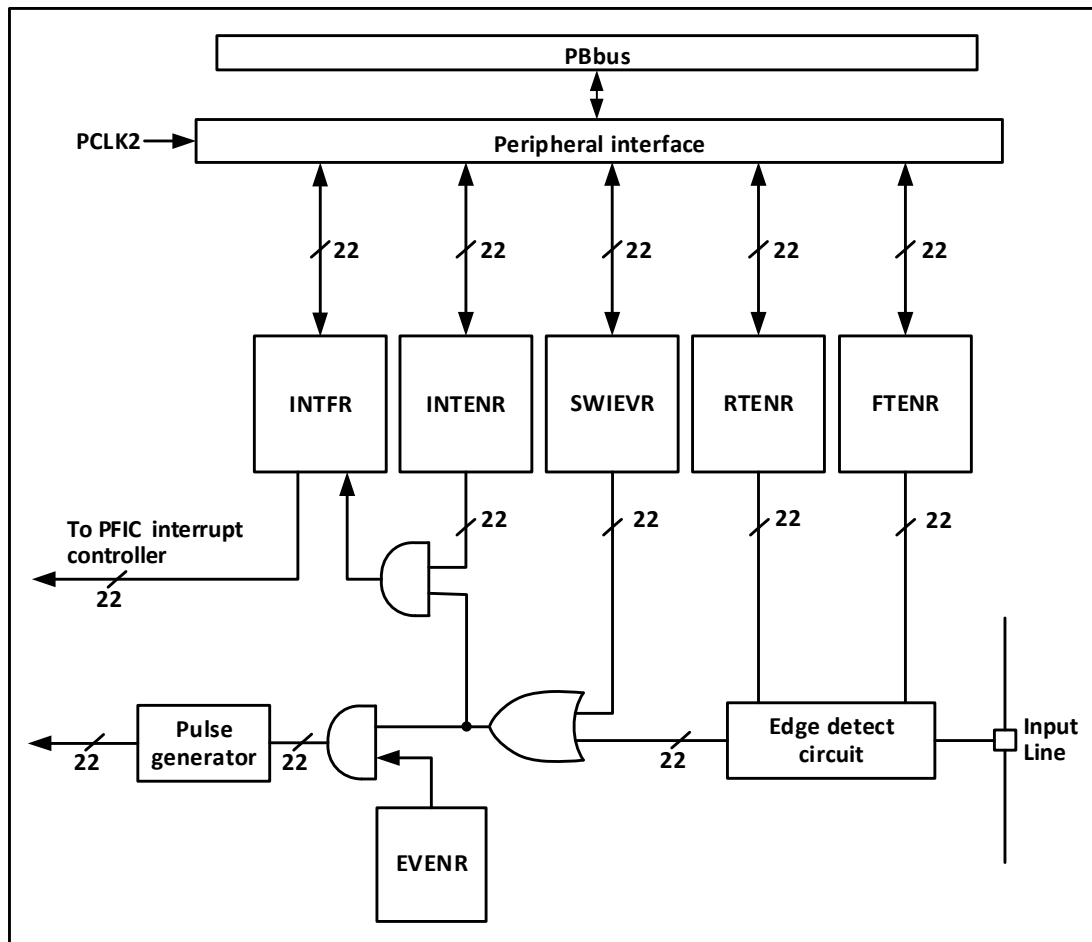
23	9	可编程	EXTI1	EXTI 线 1 中断	0x0000005C
24	10	可编程	EXTI2	EXTI 线 2 中断	0x00000060
25	11	可编程	EXTI3	EXTI 线 3 中断	0x00000064
26	12	可编程	EXTI4	EXTI 线 4 中断	0x00000068
27	13	可编程	DMA_CH1	DMA 通道 1 全局中断	0x0000006C
28	14	可编程	DMA_CH2	DMA 通道 2 全局中断	0x00000070
29	15	可编程	DMA_CH3	DMA 通道 3 全局中断	0x00000074
30	16	可编程	DMA_CH4	DMA 通道 4 全局中断	0x00000078
31	17	可编程	DMA_CH5	DMA 通道 5 全局中断	0x0000007C
32	18	可编程	DMA_CH6	DMA 通道 6 全局中断	0x00000080
33	19	可编程	DMA_CH7	DMA 通道 7 全局中断	0x00000084
34	20	可编程	ADC	ADC 全局中断	0x00000088
35	21	可编程	USB_HP 或 CAN_TX	USB_HP 或 CAN_TX 全局中断	0x0000008C
36	22	可编程	USB_LP 或 CAN_RX0	USB_LP 或 CAN_RX0 全局中断	0x00000090
37	23	可编程	CAN_RX1	CAN_RX1 全局中断	0x00000094
38	24	可编程	CAN_SCE	CAN_SCE 全局中断	0x00000098
39	25	可编程	EXTI9_5	EXTI 线[9:5]中断	0x0000009C
40	26	可编程	TIM1_BRK	TIM1 刹车中断	0x000000A0
41	27	可编程	TIM1_UP	TIM1 更新中断	0x000000A4
42	28	可编程	TIM1_TRG_COM	TIM1 触发和通信中断	0x000000A8
43	29	可编程	TIM1_CC	TIM1 捕获比较中断	0x000000AC
44	30	可编程	TIM2	TIM2 全局中断	0x000000B0
45	31	可编程	TIM3	TIM3 全局中断	0x000000B4
46	32	可编程	TIM4	TIM4 全局中断	0x000000B8
47	33	可编程	I2C1_EV	I ² C1 事件中断	0x000000BC
48	34	可编程	I2C1_ER	I ² C1 错误中断	0x000000C0
49	35	可编程	I2C2_EV	I ² C2 事件中断	0x000000C4
50	36	可编程	I2C2_ER	I ² C2 错误中断	0x000000C8
51	37	可编程	SPI1	SPI1 全局中断	0x000000CC
52	38	可编程	SPI2	SPI2 全局中断	0x000000D0
53	39	可编程	USART1	USART1 全局中断	0x000000D4
54	40	可编程	USART2	USART2 全局中断	0x000000D8
55	41	可编程	USART3	USART3 全局中断	0x000000DC
56	42	可编程	EXTI15_10	EXTI 线[15:10]中断	0x000000E0
57	43	可编程	RTCAlarm	RTC 闹钟中断 (EXTI)	0x000000E4
58	44	可编程	LPTIM_WKUP	LPTIM 唤醒中断	0x000000E8
59	45	可编程	USBFS	USBFS 全局中断	0x000000EC
60	46	可编程	USBFS_WKUP	USBFS 唤醒中断	0x000000F0
61	47	可编程	USART4	USART4 全局中断	0x000000F4
62	48	可编程	DMA_CH8	DMA 通道 8 全局中断	0x000000F8
63	49	可编程	LPTIM	LPTIM 全局中断	0x000000FC
64	50	可编程	OPA	OPA 全局中断	0x00000100
65	51	可编程	USBPD	USBPD 全局中断	0x00000104
66	-	-	-	-	0x00000108
67	52	可编程	USBPD_WKUP	USBPD 唤醒中断	0x0000010C

68	53	可编程	CMP_WKUP	CMP 唤醒中断	0x00000110
----	----	-----	----------	----------	------------

9.4 外部中断和事件控制器 (EXTI)

9.4.1 概述

图 9-1 外部中断(EXTI)接口框图



由图 9-1 可以看出，外部中断的触发源既可以是软件中断 (SWIEVR) 也可以是实际的外部中断通道，外部中断通道的信号会先经过边沿检测电路 (edge detect circuit) 的筛选。只要产生软件中断或外部中断信号其一，就会通过图中的或门电路输出给事件使能和中断使能两个与门电路，只要有中断被使能或事件被使能，就会产生中断或事件。EXTI 的六个寄存器由处理器通过 PB2 接口访问。

9.4.2 唤醒事件说明

系统可以通过唤醒事件来唤醒由 WFE 指令引起的睡眠模式。唤醒事件通过以下两种配置产生：

- 在外设的寄存器里使能一个中断，但不在内核的 PFIC 里使能这个中断，同时在内核里使能 SEVONPEND 位。体现在 EXTI 中，就是使能 EXTI 中断，但不在 PFIC 中使能 EXTI 中断，同时使能 SEVONPEND 位。当 CPU 从 WFE 中唤醒后，需要清除 EXTI 的中断标志位和 PFIC 挂起位。
- 使能一个 EXTI 通道为事件通道，CPU 从 WFE 唤醒后无需清除中断标志位和 PFIC 挂起位的操作。

9.4.3 说明

使用外部中断需要配置相应外部中断通道，即选择相应触发沿，使能相应中断。当外部中断通道上出现了设定的触发沿时，将产生一个中断请求，对应的中断标志位也会被置位。对标志位写 1 可以清除该标志位。

使用外部硬件中断步骤：

- 1) 配置 GPIO 操作；
- 2) 配置对应的外部中断通道的中断使能位（EXTI_INTENR）；
- 3) 配置触发沿（EXTI_RTENR 或 EXTI_FTENR），选择上升沿触发、下降沿触发或双边沿触发；
- 4) 在内核的 PFIC 中配置 EXTI 中断，以保证其可以正确响应。

使用外部硬件事件步骤：

- 1) 配置 GPIO 操作；
- 2) 配置对应的外部中断通道的事件使能位（EXTI_EVENTR）；
- 3) 配置触发沿（EXTI_RTENR 或 EXTI_FTENR），选择上升沿触发、下降沿触发或双边沿触发。

使用软件中断/事件步骤：

- 1) 使能外部中断（EXTI_INTENR）或外部事件（EXTI_EVENTR）；
- 2) 如果使用中断服务函数，需要设置内核的 PFIC 里 EXTI 中断；
- 3) 设置软件中断触发（EXTI_SWIEVR），即会产生中断。

9.4.4 外部事件映射

表 9-2 EXTI 中断映射

外部中断/事件线路	映射事件描述
EXTI0~EXTI15	Px0~Px15 (x=A/B/C/D)，任何一个 I/O 口都可以启用外部中断/事件功能，由 AFIO_EXTICRx 寄存器配置。
EXTI16	PVD 事件：超出电压监控阈值
EXTI17	RTC 闹钟事件
EXTI19	USBPD 唤醒事件
EXTI20	USBFS 唤醒事件
EXTI21	LPTIM 唤醒事件
EXTI22	COMP 唤醒事件

9.5 寄存器描述

9.5.1 EXTI 寄存器描述

表 9-3 EXTI 相关寄存器列表

名称	访问地址	描述	复位值
R32_EXTI_INTENR	0x40010400	中断使能寄存器	0x00000000
R32_EXTI_EVENTR	0x40010404	事件使能寄存器	0x00000000
R32_EXTI_RTENR	0x40010408	上升沿触发使能寄存器	0x00000000
R32_EXTI_FTENR	0x4001040C	下降沿触发使能寄存器	0x00000000
R32_EXTI_SWIEVR	0x40010410	软中断事件寄存器	0x00000000
R32_EXTI_INTFR	0x40010414	中断标志位寄存器	0x0000XXXX

9.5.1.1 中断使能寄存器（EXTI_INTENR）

偏移地址：0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved									MR22	MR21	MR20	MR19	Reserved	MR17	MR16

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MR15	MR14	MR13	MR12	MR11	MR10	MR9	MR8	MR7	MR6	MR5	MR4	MR3	MR2	MR1	MR0

位	名称	访问	描述	复位值
[31:23]	Reserved	R0	保留。	0
[22:0]	MRx	RW	使能外部中断通道 x 的中断请求信号： 1：使能此通道的中断； 0：屏蔽此通道的中断。	0

9.5.1.2 事件使能寄存器 (EXTI_EVENR)

偏移地址：0x04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved									MR22	MR21	MR20	MR19	Reserved	MR17	MR16

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MR15	MR14	MR13	MR12	MR11	MR10	MR9	MR8	MR7	MR6	MR5	MR4	MR3	MR2	MR1	MR0

位	名称	访问	描述	复位值
[31:23]	Reserved	R0	保留。	0
[22:0]	MRx	RW	使能外部中断通道 x 的事件请求信号： 1：使能此通道的事件； 0：屏蔽此通道的事件。	0

9.5.1.3 上升沿触发使能寄存器 (EXTI_RTENR)

偏移地址：0x08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved									TR22	TR21	TR20	TR19	Reserved	TR17	TR16

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TR15	TR14	TR13	TR12	TR11	TR10	TR9	TR8	TR7	TR6	TR5	TR4	TR3	TR2	TR1	TR0

位	名称	访问	描述	复位值
[31:23]	Reserved	R0	保留。	0
[22:0]	TRx	RW	使能外部中断通道 x 的上升沿触发： 1：使能此通道的上升沿触发； 0：禁止此通道的上升沿触发。	0

9.5.1.4 下降沿触发使能寄存器 (EXTI_FTENR)

偏移地址：0x0C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved									TR22	TR21	TR20	TR19	Reserved	TR17	TR16

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TR15	TR14	TR13	TR12	TR11	TR10	TR9	TR8	TR7	TR6	TR5	TR4	TR3	TR2	TR1	TR0

位	名称	访问	描述	复位值
[31:23]	Reserved	R0	保留	0
[22:0]	TRx	RW	使能外部中断通道 x 的下降沿触发： 0：禁止此通道的下降沿触发； 1：使能此通道的下降沿触发。	0

9.5.1.5 软中断事件寄存器 (EXTI_SWIEVR)

偏移地址：0x10

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

位	名称	访问	描述	复位值
[31:23]	Reserved	R0	保留。	0
[22:0]	SWIERx	RW	在相对应的外部触发中断通道上设置一个软件中断。这里置位会使中断标志位 (EXTI_INTFR) 对应位置位，如果中断使能 (EXTI_INTENR) 或事件使能 (EXTI_EVENTR) 开启，那么就会产生中断或事件。	0

9.5.1.6 中断标志位寄存器 (EXTI_INTFR)

偏移地址：0x14

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved									IF22	IF21	IF20	IF19	Reser ved	IF17	IF16
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IF15	IF14	IF13	IF12	IF11	IF10	IF9	IF8	IF7	IF6	IF5	IF4	IF3	IF2	IF1	IF0

位	名称	访问	描述	复位值
[31:23]	Reserved	R0	保留。	0
[22:0]	IFx	RW1	中断标志位，该位置位标志表示发生了对应的外部中断。写 1 可以清除此位。	0

9.5.2 PFIC 寄存器描述

表 9-4 PFIC 相关寄存器列表

名称	访问地址	描述	复位值
----	------	----	-----

R32_PFIIC_ISR1	0xE000E000	PFIC 中断使能状态寄存器 1	0x0000000C
R32_PFIIC_ISR2	0xE000E004	PFIC 中断使能状态寄存器 2	0x00000000
R32_PFIIC_ISR3	0xE000E008	PFIC 中断使能状态寄存器 3	0x00000000
R32_PFIIC_IPR1	0xE000E020	PFIC 中断挂起状态寄存器 1	0x00000000
R32_PFIIC_IPR2	0xE000E024	PFIC 中断挂起状态寄存器 2	0x00000000
R32_PFIIC_IPR3	0xE000E028	PFIC 中断挂起状态寄存器 3	0x00000000
R32_PFIIC_ITHRESDR	0xE000E040	PFIC 中断优先级阈值配置寄存器	0x00000000
R32_PFIIC_CFGR	0xE000E048	PFIC 中断配置寄存器	0x00000000
R32_PFIIC_GISR	0xE000E04C	PFIC 中断全局状态寄存器	0x00000000
R32_PFIIC_VTFIDR	0xE000E050	PFIC VTF 中断 ID 配置寄存器	0x00000000
R32_PFIIC_VTFADDRR0	0xE000E060	PFIC VTF 中断 0 地址寄存器	0x00000000
R32_PFIIC_VTFADDRR1	0xE000E064	PFIC VTF 中断 1 地址寄存器	0x00000000
R32_PFIIC_VTFADDRR2	0xE000E068	PFIC VTF 中断 2 地址寄存器	0x00000000
R32_PFIIC_VTFADDRR3	0xE000E06C	PFIC VTF 中断 3 地址寄存器	0x00000000
R32_PFIIC_IENR1	0xE000E100	PFIC 中断使能设置寄存器 1	0x00000000
R32_PFIIC_IENR2	0xE000E104	PFIC 中断使能设置寄存器 2	0x00000000
R32_PFIIC_IENR3	0xE000E108	PFIC 中断使能设置寄存器 3	0x00000000
R32_PFIIC_IRER1	0xE000E180	PFIC 中断使能清除寄存器 1	0x00000000
R32_PFIIC_IRER2	0xE000E184	PFIC 中断使能清除寄存器 2	0x00000000
R32_PFIIC_IRER3	0xE000E188	PFIC 中断使能清除寄存器 3	0x00000000
R32_PFIIC_IPSR1	0xE000E200	PFIC 中断挂起设置寄存器 1	0x00000000
R32_PFIIC_IPSR2	0xE000E204	PFIC 中断挂起设置寄存器 2	0x00000000
R32_PFIIC_IPSR3	0xE000E208	PFIC 中断挂起设置寄存器 3	0x00000000
R32_PFIIC_IPRR1	0xE000E280	PFIC 中断挂起清除寄存器 1	0x00000000
R32_PFIIC_IPRR2	0xE000E284	PFIC 中断挂起清除寄存器 2	0x00000000
R32_PFIIC_IPRR3	0xE000E288	PFIC 中断挂起清除寄存器 3	0x00000000
R32_PFIIC_IACR1	0xE000E300	PFIC 中断激活状态寄存器 1	0x00000000
R32_PFIIC_IACR2	0xE000E304	PFIC 中断激活状态寄存器 2	0x00000000
R32_PFIIC_IACR3	0xE000E308	PFIC 中断激活状态寄存器 3	0x00000000
R32_PFIIC_IPRIORx	0xE000E400	PFIC 中断优先级配置寄存器	0x00000000
R32_PFIIC_SCTLR	0xE000ED10	PFIC 系统控制寄存器	0x00000000

注：1. NMI、EXC、ECALL-M、ECALL-U、BREAKPOINT 中断默认总是使能。

2. ECALL-M、ECALL-U、BREAKPOINT 均为 EXC 的一种情况，状态由 EXC 的状态位 bit3 表示。

3. NMI、EXC 支持中断挂起清除和设置操作，不支持中断使能清除和设置操作。

4. ECALL-M、ECALL-U、BREAKPOINT 不支持中断挂起清除和设置、中断使能清除和设置操作。

9.5.2.1 PFIC 中断使能状态寄存器 1 (PFIC_ISR1)

偏移地址：0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INTENSTA[31:16]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	INTENSTA14	Reserved	INTENSTA12	Reserved								INTENSTA3	INTENSTA2	Reserved	

位	名称	访问	描述	复位值
[31:16]	INTENSTA	RO	16#-31#中断当前使能状态。 1: 当前编号中断已使能; 0: 当前编号中断未启用。	0
15	Reserved	RO	保留	0
14	INTENSTA	RO	14#中断当前使能状态。 1: 当前编号中断已使能; 0: 当前编号中断未启用。	0
13	Reserved	RO	保留	0
12	INTENSTA	RO	12#中断当前使能状态。 1: 当前编号中断已使能; 0: 当前编号中断未启用。	0
[11:4]	Reserved	RO	保留	0
[3:2]	INTENSTA	RO	2#-3#中断当前使能状态。 1: 当前编号中断已使能; 0: 当前编号中断未启用。	1
[1:0]	Reserved	RO	保留	0

9.5.2.2 PFIC 中断使能状态寄存器 2 (PFIC_ISR2)

偏移地址: 0x04

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

位	名称	访问	描述	复位值
[31:0]	INTENSTA	RO	32#-63#中断当前使能状态。 1: 当前编号中断已使能; 0: 当前编号中断未启用。	0

9.5.2.3 PFIC 中断使能状态寄存器 3 (PFIC_ISR3)

偏移地址: 0x08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved											INTENSTA[68:64]				

位	名称	访问	描述	复位值
[31:5]	Reserved	RO	保留	0
[4:0]	INTENSTA	RO	64#-68#中断当前使能状态。 1: 当前编号中断已使能; 0: 当前编号中断未启用。	0

9.5.2.4 PFIC 中断挂起状态寄存器 1 (PFIC_IPR1)

偏移地址: 0x20

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PENDSTA[31:16]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	PENDSTA14	Reserved	PENDSTA12	Reserved								PENDSTA3	PENDSTA2	Reserved	

位	名称	访问	描述	复位值
[31:12]	PENDSTA	R0	16#-31#中断当前挂起状态。 1: 当前编号中断已挂起; 0: 当前编号中断未挂起。	0
15	Reserved	R0	保留。	0
14	PENDSTA	R0	14#中断当前挂起状态。 1: 当前编号中断已挂起; 0: 当前编号中断未挂起。	0
13	Reserved	R0	保留。	0
12	PENDSTA	R0	12#中断当前挂起状态。 1: 当前编号中断已挂起; 0: 当前编号中断未挂起。	0
[11:4]	Reserved	R0	保留。	0
[3:2]	PENDSTA	R0	2#-3#中断当前挂起状态。 1: 当前编号中断已挂起; 0: 当前编号中断未挂起。	0
[1:0]	Reserved	R0	保留。	0

9.5.2.5 PFIC 中断挂起状态寄存器 2 (PFIC_IPR2)

偏移地址: 0x24

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

位	名称	访问	描述	复位值
[31:0]	PENDSTA	R0	32#-63#中断当前挂起状态。 1: 当前编号中断已挂起; 0: 当前编号中断未挂起。	0

9.5.2.6 PFIC 中断挂起状态寄存器 3 (PFIC_IPR3)

偏移地址: 0x28

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

Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved											PENDSTA[68:64]				

位	名称	访问	描述	复位值
[31:5]	Reserved	RO	保留。	0
[4:0]	PENDSTA	RO	64#~68#中断当前挂起状态。 1: 当前编号中断已挂起; 0: 当前编号中断未挂起。	0

9.5.2.7 PFIC 中断优先级阈值配置寄存器 (PFIC_ITHRESDR)

偏移地址: 0x40

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
Reserved																								THRESHOLD[7:0]							

位	名称	访问	描述	复位值
[31:8]	Reserved	RO	保留。	0
[7:0]	THRESHOLD	RW	中断优先级阈值设置值。 低于当前设置值的中断优先级值, 当挂起时不执行中断服务; 此寄存器为 0 时表示阈值寄存器功能无效。 [7:5]: 优先级阈值。 [4:0]: 保留, 固定为 0, 写无效。	0

9.5.2.8 PFIC 中断配置寄存器 (PFIC_CFGR)

偏移地址: 0x48

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

位	名称	访问	描述	复位值
[31:16]	KEYCODE	WO	对应不同的目标控制位, 需要同步写入相应的安全访问标识数据才能修改, 读出数据固定为 0。 KEY1 = 0xFA05; KEY2 = 0xBCAF; KEY3 = 0xBEEF。	0
[15:8]	Reserved	RO	保留。	0
7	RSTSYS	WO	系统复位 (同步写入 KEY3)。自动清 0。 写 1 有效, 写 0 无效。 注: 与 PFIC_SCTLR 寄存器 SYSRST 位作用相同。	0

[6:0]	Reserved	R0	保留。	0
-------	----------	----	-----	---

9.5.2.9 PFIC 中断全局状态寄存器 (PFIC_GISR)

偏移地址: 0x4C

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

位	名称	访问	描述	复位值
[31:10]	Reserved	R0	保留。	0
9	GPENDSTA	R0	当前是否有中断处于挂起: 1: 有; 0: 没有。	0
8	GACTSTA	R0	当前是否有中断被执行: 1: 有; 0: 没有。	0
[7:0]	NESTSTA	R0	当前中断嵌套状态, 目前最大支持 2 级嵌套, 硬件压栈深度最大为 2 级。 0x03: 第 2 级中断中; 0x01: 第 1 级中断中; 0x00: 没有中断发生; 其他: 不可能情况。	0x00

9.5.2.10 PFIC VTF 中断 ID 配置寄存器 (PFIC_VTFIDR)

偏移地址: 0x50

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
VTFID3								VTFID2							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VTFID1								VTFID0							

位	名称	访问	描述	复位值
[31:24]	VTFID3	RW	配置 VTF 中断 3 的中断编号。	0
[23:16]	VTFID2	RW	配置 VTF 中断 2 的中断编号。	0
[15:8]	VTFID1	RW	配置 VTF 中断 1 的中断编号。	0
[7:0]	VTFID0	RW	配置 VTF 中断 0 的中断编号。	0

9.5.2.11 PFIC VTF 中断 0 地址寄存器 (PFIC_VTFADDR0)

偏移地址: 0x60

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADDR0[31:16]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

ADDR0[15:1]	VTF0EN
-------------	--------

位	名称	访问	描述	复位值
[31:1]	ADDR0	RW	VTF 中断 0 服务程序地址 bit[31:1], bit0 为 0。	0
0	VTF0EN	RW	VTF 中断 0 使能位: 1: 启用 VTF 中断 0 通道; 0: 关闭。	0

9.5.2.12 PFIC VTF 中断 1 地址寄存器 (PFIC_VTFADDRR1)

偏移地址: 0x64

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADDR1[31:16]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR1[15:1]															VTF1EN

位	名称	访问	描述	复位值
[31:1]	ADDR1	RW	VTF 中断 1 服务程序地址 bit[31:1], bit0 为 0。	0
0	VTF1EN	RW	VTF 中断 1 使能位: 1: 启用 VTF 中断 1 通道; 0: 关闭。	0

9.5.2.13 PFIC VTF 中断 2 地址寄存器 (PFIC_VTFADDRR2)

偏移地址: 0x68

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADDR2[31:16]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR2[15:1]															VTF2EN

位	名称	访问	描述	复位值
[31:1]	ADDR2	RW	VTF 中断 2 服务程序地址 bit[31:1], bit0 为 0。	0
0	VTF2EN	RW	VTF 中断 2 使能位: 1: 启用 VTF 中断 2 通道; 0: 关闭。	0

9.5.2.14 PFIC VTF 中断 3 地址寄存器 (PFIC_VTFADDRR3)

偏移地址: 0x6C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADDR3[31:16]															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR3[15:1]														VTF3EN	

位	名称	访问	描述	复位值
[31:1]	ADDR3	RW	VTF 中断 3 服务程序地址 bit[31:1], bit0 为 0。	0
0	VTF3EN	RW	VTF 中断 3 使能位: 1: 启用 VTF 中断 3 通道; 0: 关闭。	0

9.5.2.15 PFIC 中断使能设置寄存器 1 (PFIC_IENR1)

偏移地址: 0x100

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INTEN[31:16]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	INTEN14	Reserved	INTEN12	Reserved											

位	名称	访问	描述	复位值
[31:16]	INTEN	WO	16#-31#中断使能控制。 1: 当前编号中断使能; 0: 无影响。	0
15	Reserved	RO	保留。	0
14	INTEN	WO	14#中断使能控制。 1: 当前编号中断使能; 0: 无影响。	0
13	Reserved	RO	保留。	0
12	INTEN	WO	12#中断使能控制。 1: 当前编号中断使能; 0: 无影响。	0
[11:0]	Reserved	RO	保留。	0

9.5.2.16 PFIC 中断使能设置寄存器 2 (PFIC_IENR2)

偏移地址: 0x104

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

位	名称	访问	描述	复位值
[31:0]	INTEN	WO	32#-63#中断使能控制。 1: 当前编号中断使能;	0

			0: 无影响。	
--	--	--	---------	--

9.5.2.17 PFIC 中断使能设置寄存器 3 (PFIC_IENR3)

偏移地址: 0x108

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved											INTEN[68:64]				

位	名称	访问	描述	复位值
[31:5]	Reserved	RO	保留。	0
[4:0]	INTEN	WO	64#-68#中断使能控制。 1: 当前编号中断使能; 0: 无影响。	0

9.5.2.18 PFIC 中断使能清除寄存器 1 (PFIC_IRER1)

偏移地址: 0x180

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INTRST[31:16]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	INTRSET 14	Reserved	INTRSET1 2	Reserved											

位	名称	访问	描述	复位值
[31:16]	INTRSET	WO	16#-31#中断关闭控制。 1: 当前编号中断关闭; 0: 无影响。	0
15	Reserved	RO	保留。	0
14	INTRSET	WO	14#中断关闭控制。 1: 当前编号中断关闭; 0: 无影响。	0
13	Reserved	RO	保留。	0
12	INTRSET	WO	12#中断关闭控制。 1: 当前编号中断关闭; 0: 无影响。	0
[11:0]	Reserved	RO	保留。	0

9.5.2.19 PFIC 中断使能清除寄存器 2 (PFIC_IRER2)

偏移地址: 0x184

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

INTRSET[47:32]

位	名称	访问	描述	复位值
[31:0]	INTRSET	WO	32#-63#中断关闭控制。 1: 当前编号中断关闭; 0: 无影响。	0

9.5.2.20 PFIC 中断使能清除寄存器 3 (PFIC_IRER3)

偏移地址: 0x188

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved											INTRSET[68:64]				

位	名称	访问	描述	复位值
[31:5]	Reserved	RO	保留。	0
[4:0]	INTRSET	WO	64#-68#中断关闭控制。 1: 当前编号中断关闭; 0: 无影响。	0

9.5.2.21 PFIC 中断挂起设置寄存器 1 (PFIC_IPSR1)

偏移地址: 0x200

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PENDSET[31:16]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	PEND SET14	Reserved	PEND SET12	Reserved								PEND SET3	PEND SET2	Reserved	

位	名称	访问	描述	复位值
[31:12]	PENDSET	WO	16#-31#中断挂起设置。 1: 当前编号中断挂起; 0: 无影响。	0
15	Reserved	RO	保留。	0
14	PENDSET	WO	14#中断挂起设置, 13#和 15#保留。 1: 当前编号中断挂起; 0: 无影响。	0
13	Reserved	RO	保留。	0
12	PENDSET	WO	12#中断挂起设置, 13#和 15#保留。 1: 当前编号中断挂起; 0: 无影响。	0
[11:4]	Reserved	RO	保留。	0
[3:2]	PENDSET	WO	2#-3#中断挂起设置。	0

			1: 当前编号中断挂起; 0: 无影响。	
[1:0]	Reserved	R0	保留。	0

9.5.2.22 PFIC 中断挂起设置寄存器 2 (PFIC_IPSR2)

偏移地址: 0x204

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

位	名称	访问	描述	复位值
[31:0]	PENDSET	WO	32#~63#中断挂起设置。 1: 当前编号中断挂起; 0: 无影响。	0

9.5.2.23 PFIC 中断挂起设置寄存器 3 (PFIC_IPSR3)

偏移地址: 0x208

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved											PENDSET [68:64]				

位	名称	访问	描述	复位值
[31:5]	Reserved	R0	保留	0
[4:0]	PENDSET	WO	64#~68#中断挂起设置。 1: 当前编号中断挂起; 0: 无影响。	0

9.5.2.24 PFIC 中断挂起清除寄存器 1 (PFIC_IPRR1)

偏移地址: 0x280

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PENDRST [31:16]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	PEND RST14	Reserved	PEND RST12	Reserved								PEND RST3	PEND RST2	Reserved	

位	名称	访问	描述	复位值
[31:16]	PENDRST	WO	16#~31#中断挂起清除。 1: 当前编号中断清除挂起状态; 0: 无影响。	0

15	Reserved	RO	保留。	0
14	PENDRST	WO	14#中断挂起清除。 1: 当前编号中断清除挂起状态; 0: 无影响。	0
13	Reserved	RO	保留。	0
12	PENDRST	WO	12#中断挂起清除。 1: 当前编号中断清除挂起状态; 0: 无影响。	0
[11:4]	Reserved	RO	保留。	0
[3:2]	PENDRST	WO	2#-3#中断挂起清除。 1: 当前编号中断清除挂起状态; 0: 无影响。	0
[1:0]	Reserved	RO	保留。	0

9.5.2.25 PFIC 中断挂起清除寄存器 2 (PFIC_IPRR2)

偏移地址: 0x284

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

位	名称	访问	描述	复位值
[31:0]	PENDRST	WO	32#-63#中断挂起清除。 1: 当前编号中断清除挂起状态; 0: 无影响。	0

9.5.2.26 PFIC 中断挂起清除寄存器 3 (PFIC_IPRR3)

偏移地址: 0x288

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												PENDRST [68:64]			

位	名称	访问	描述	复位值
[31:5]	Reserved	RO	保留。	0
[4:0]	PENDRST	WO	64#-68#中断挂起清除。 1: 当前编号中断清除挂起状态; 0: 无影响。	0

9.5.2.27 PFIC 中断激活状态寄存器 1 (PFIC_IACR1)

偏移地址: 0x300

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

IACTS[31:16]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	IACTS14	Reserved	IACTS12	Reserved								IACTS3	IACTS2	Reserved	

位	名称	访问	描述	复位值
[31:16]	IACTS	R0	16#-31#中断执行状态。 1: 当前编号中断执行中; 0: 当前编号中断没执行。	0
15	Reserved	R0	保留。	0
14	IACTS	R0	14#中断执行状态。 1: 当前编号中断执行中; 0: 当前编号中断没执行。	0
13	Reserved	R0	保留。	0
12	IACTS	R0	12#中断执行状态。 1: 当前编号中断执行中; 0: 当前编号中断没执行。	0
[11:4]	Reserved	R0	保留。	0
[3:2]	IACTS	R0	2#-3#中断执行状态。 1: 当前编号中断执行中; 0: 当前编号中断没执行。	0
[1:0]	Reserved	R0	保留。	0

9.5.2.28 PFIC 中断激活状态寄存器 2 (PFIC_IACR2)

偏移地址: 0x304

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

位	名称	访问	描述	复位值
[31:0]	IACTS	R0	32#-63#中断执行状态。 1: 当前编号中断执行中; 0: 当前编号中断没执行。	0

9.5.2.29 PFIC 中断激活状态寄存器 3 (PFIC_IACR3)

偏移地址: 0x308

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved											IACTS[68:64]				

位	名称	访问	描述	复位值
[31:5]	Reserved	R0	保留	0
[4:0]	IACTS	R0	64#~68#中断执行状态。 1: 当前编号中断执行中; 0: 当前编号中断没执行。	0

9.5.2.30 PFIC 中断优先级配置寄存器 (PFIC_IPRIORx) (x=0~17)

偏移地址: 0x400 ~ 0x4FF

控制器支持 256 个中断 (0~255)，每个中断使用 8bit 来设置控制优先级。

	31	24	23	16	15	8	7	0
IPRIOR17	Reserved		Reserved		Reserved		PRIO_68	
IPRIOR16	PRIO_67		PRIO_66		PRIO_65		PRIO_64	
...	
IPRIOR0	PRIO_3		PRIO_2		PRIO_1		PRIO_0	

位	名称	访问	描述	复位值
[551:544]	IP_68	RW	同 IP_0 描述。	0
...
[31:24]	IP_3	RW	同 IP_0 描述。	0
[23:16]	IP_2	RW	同 IP_0 描述。	0
[15:8]	IP_1	RW	同 IP_0 描述。	0
[7:0]	IP_0	RW	编号 0 中断优先级配置: [7:5]: 优先级控制位。 若配置无嵌套, 无抢占位; 若配置 2 级嵌套, bit7 为抢占位; 优先级数值越小则优先级越高, 同一抢占优先级中断若同时挂起, 优先执行优先级高的中断。 [4:0]: 保留, 固定为 0, 写无效。	0

9.5.2.31 PFIC 系统控制寄存器 (PFIC_SCTLR)

偏移地址: 0xD10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								LOCKUP	SET	SEV	WFIT0	SLEEP	SLEEP	Reserved	
								P	EVENT	ONPEND	WFE	DEEP	ONEXIT		

位	名称	访问	描述	复位值
31	SYSRST	WO	系统复位, 自动清 0。写 1 有效, 写 0	0

			无效，与 PFIC_CFGR 寄存器相同效果	
[30:5]	Reserved	RO	保留。	0
6	LOCKUP	RW	内核死锁使能(内核执行异常和 NMI 执行指令时会进入死锁) 1: 关闭使能 0: 开启使能	0
5	SETEVENT	WO	设置事件，可以唤醒 WFE 的情况。	0
4	SEVONPEND	RW	当发生事件或者中断挂起状态时，可以从 WFE 指令后唤醒系统,如果未执行 WFE 指令，将在下次执行该指令后立即唤醒系统。 1: 启用的事件和所有中断（包括未开启中断）都能唤醒系统； 0: 只有启用的事件和启用的中断可以唤醒系统。	0
3	WFIWFE	RW	将 WFI 指令当成是 WFE 执行。 1: 将之后的 WFI 指令当做 WFE 指令； 0: 无作用。	0
2	SLEEPDEEP	RW	控制系统的低功耗模式： 1: deepsleep 0: sleep	0
1	SLEEPONEXIT	RW	控制离开中断服务程序后，系统状态： 1: 系统进入低功耗模式； 0: 系统进入主程序。	0
0	Reserved	RO	保留。	0

9.5.3 专用 CSR 寄存器

RISC-V 架构中定义了一些控制和状态寄存器（Control and Status Register, CSR），用于配置或标识或记录运行状态。CSR 寄存器属于内核内部的寄存器，使用专用的 12 位地址空间。该芯片除了 RISC-V 特权架构文档中定义的标准寄存器外，还增加了一些厂商自定义寄存器，需要使用 csr 指令进行访问。

注：此类寄存器标注为“MRW, MRO, MRW1”属性的需要系统在机器模式下才能访问。

9.5.3.1 中断系统控制寄存器（INTSYSCR）

CSR 地址：0x804

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

位	名称	访问	描述	复位值
[31:6]	Reserved	MRO	保留。	0
5	GIHWSTKNEN	MRW1	全局中断和硬件压栈关闭使能。 注：该位常使用于实时操作系统中，中	0

			断切换上下文时，置位该位，可关闭全局中断和硬件压栈出栈，当上下文切换完成，执行完中断返回后，硬件自动清除该位。	
[4:2]	Reserved	MRO	保留。	0
1	INESTEN	MRW	中断嵌套使能： 0：中断嵌套功能关闭； 1：中断嵌套功能使能。	0
0	HWSTKEN	MRW	硬件压栈使能： 0：硬件压栈功能关闭； 1：硬件压栈功能使能。	0

9.5.3.2 异常入口基地址寄存器（MTVEC）

CSR 地址：0x305

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BASEADDR[31:16]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BASEADDR[15:2]														MODE1	MODE0

位	名称	访问	描述	复位值
[31:2]	BASEADDR[31:2]	MRW	中断向量表基地址。	0
1	MODE1	MRW	中断向量表识别模式： 0：按跳转指令识别，有限范围，支持非跳指令； 1：按绝对地址识别，支持全范围，但必须跳转。	0
0	MODE0	MRW	中断或异常入口地址模式选择： 0：使用统一入口地址； 1：根据中断编号*4 进行地址偏移。	0

9.5.4 物理内存保护单元（PMP）

为了提高系统安全，RISC-V 的架构中定义了一套物理地址访问限制，可以为区域内物理内存设置其读、写、执行属性，区域长度最小 4 字节保护。PMP 单元在用户模式下一直生效，在机器模式下可选生效，如果违背了当前内存限制，将会产生系统异常中断（EXC）。

PMP 单元包含 4 组 8-bit 的配置寄存器（32bit）和 4 组地址寄存器，需要使用 csr 指令进行访问，并且在机器模式下进行。

9.5.4.1 PMP 配置寄存器（PMPCFG0）

CSR 地址：0x3A0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
pmp3cfg								pmp2cfg							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
pmp1cfg								pmp0cfg							

位	名称	访问	描述			复位值
[31:24]	pmp3cfg	MRW	见 pmp0cfg。			0
[23:16]	pmp2cfg	MRW	见 pmp0cfg。			0
[15:8]	pmp1cfg	MRW	见 pmp0cfg。			0
[7:0]	pmp0cfg	MRW	位	名称	描述	0
			7	L	锁定使能，机器模式下可解锁 0：不锁定； 1：锁定相关寄存器。	
			[6:5]	—	保留。	
			[4:3]	A	地址对齐及保护区域范围选择。	
			2	X	可执行属性。	
			1	W	可写入属性。	
			0	R	可读出属性。	

其中，地址对齐及保护区域范围选择，对于 $A_ADDR \leq region < B_ADDR$ 区域进行内存保护（要求 A_ADDR 和 B_ADDR 均为 4 字节对齐）：

- 1、如果 $B_ADDR - A_ADDR == 2^2$ ，则采用 NA4 方式；
- 2、如果 $B_ADDR - A_ADDR == 2^{(G+2)}$ ， $G \geq 1$ ，且 A_ADDR 为 $2^{(G+2)}$ 对齐则采用 NAPOT 方式；
- 3、否则采用 TOR 方式。

A 值	名称	描述
00b	OFF	没有区域要保护
01b	TOR	顶端对齐区域保护： pmp0cfg 下， $0 \leq region < pmpaddr0$ ； pmp1cfg 下， $pmpaddr0 \leq region < pmpaddr1$ ； pmp2cfg 下， $pmpaddr1 \leq region < pmpaddr2$ ； pmp3cfg 下， $pmpaddr2 \leq region < pmpaddr3$ 。 $pmpaddr_{i-1} = A_ADDR \gg 2$ ； $pmpaddr_i = B_ADDR \gg 2$ 。
10b	NA4	固定 4 字节区域保护。 pmp0cfg~pmp3cfg 对应 pmpaddr0~pmpaddr3 作为起始地址。 $pmpaddr_i = A_ADDR \gg 2$ 。
11b	NAPOT	保护 $2^{(G+2)}$ 区域， $G \geq 1$ ，此时 A_ADDR 为 $2^{(G+2)}$ 对齐。 $pmpaddr_i = ((A_ADDR (2^{(G+2)} - 1)) \& \sim(1 \ll G + 1)) \gg 2$ 。

9.5.4.2 PMP 地址 0 寄存器（PMPADDR0）

CSR 地址：0x3B0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADDR0[33:18]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR0[17:2]															

位	名称	访问	描述	复位值
[31:0]	ADDR0	MRW	PMP 设置地址 0 的 bit[33:2], 实际高 2 位未用。	0

9.5.4.3 PMP 地址 1 寄存器 (PMPADDR1)

CSR 地址: 0x3B1

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADDR1[33:18]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR1[17:2]															

位	名称	访问	描述	复位值
[31:0]	ADDR1	MRW	PMP 设置地址 1 的 bit[33:2], 实际高 2 位未用。	0

9.5.4.4 PMP 地址 2 寄存器 (PMPADDR2)

CSR 地址: 0x3B2

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADDR2[33:18]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR2[17:2]															

位	名称	访问	描述	复位值
[31:0]	ADDR2	MRW	PMP 设置地址 2 的 bit[33:2], 实际高 2 位未用。	0

9.5.4.5 PMP 地址 3 寄存器 (PMPADDR3)

CSR 地址: 0x3B3

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADDR3[33:18]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR3[17:2]															

位	名称	访问	描述	复位值
[31:0]	ADDR3	MRW	PMP 设置地址 3 的 bit[33:2], 实际高 2 位未用。	0

9.5.5 STK 寄存器描述

表 9-5 STK 相关寄存器列表

名称	访问地址	描述	复位值
R32_STK_CTLR	0xE000F000	系统计数控制寄存器	0x00000000
R32_STK_SR	0xE000F004	系统计数状态寄存器	0x00000000
R32_STK_CNTL	0xE000F008	系统计数器低位寄存器	0x00000000
R32_STK_CNTH	0xE000F00C	系统计数器高位寄存器	0x00000000
R32_STK_CMPLR	0xE000F010	计数比较低位寄存器	0x00000000

R32_STK_CMPHR	0xE000F014	计数比较高位寄存器	0x00000000
---------------	------------	-----------	------------

注：适用于基于 32 位 RISC-V 指令集及架构设计的通用微控制器。

9.5.5.1 系统计数控制寄存器 (STK_CTLR)

偏移地址：0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SWIE	Reserved														
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										INIT	MODE	STRE	STCLK	STIE	STE

位	名称	访问	描述	复位值
31	SWIE	RW	软件中断触发使能 (SWI)： 1：触发软件中断； 0：关闭触发。 进入软件中断后，需软件清 0，否则持续触发。	0
[30:6]	Reserved	R0	保留。	0
5	INIT	W1	计数器初始值更新： 1：向上计数时更新为 0，向下计数时更新为比较值； 0：无效。	0
4	MODE	RW	计数模式： 1：向下计数； 0：向上计数。	0
3	STRE	RW	自动重载计数使能位： 1：向上计数到比较值后重新从 0 开始计数，向下计数到 0 后，重新从比较值开始计数； 0：向上计数到比较值后继续向上计数，向下计数到 0 后，重新从最大值开始向下计数。	0
2	STCLK	RW	计数器时钟源选择位： 1：HCLK 做时基； 0：HCLK/8 做时基；	0
1	STIE	RW	计数器中断使能控制位： 1：使能计数器中断； 0：关闭计数器中断。	0
0	STE	RW	系统计数器使能控制位： 1：启动系统计数器 STK； 0：关闭系统计数器 STK，计数器停止计数。	0

9.5.5.2 系统计数状态寄存器 (STK_SR)

偏移地址：0x04

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

Reserved	CNTIF
----------	-------

位	名称	访问	描述	复位值
[31:1]	Reserved	R0	保留	0
0	CNTIF	RW0	计数值比较标志，写 0 清除，写 1 无效： 1：向上计数达到比较值，向下计数到 0； 0：未达到比较值。	0

9.5.5.3 系统计数器低位寄存器（STK_CNTL）

偏移地址：0x08

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

位	名称	访问	描述	复位值
[31:0]	CNT[31:0]	RW	当前计数器计数值低 32 位。	0

注：寄存器 STK_CNTL 和寄存器 STK_CNTH 共同构成了 64 位系统计数器。

9.5.5.4 系统计数器高位寄存器（STK_CNTH）

偏移地址：0x0C

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

位	名称	访问	描述	复位值
[31:0]	CNT[63:32]	RW	当前计数器计数值高 32 位。	0

注：寄存器 STK_CNTL 和寄存器 STK_CNTH 共同构成了 64 位系统计数器。

9.5.5.5 计数比较低位寄存器（STK_CMPLR）

偏移地址：0x10

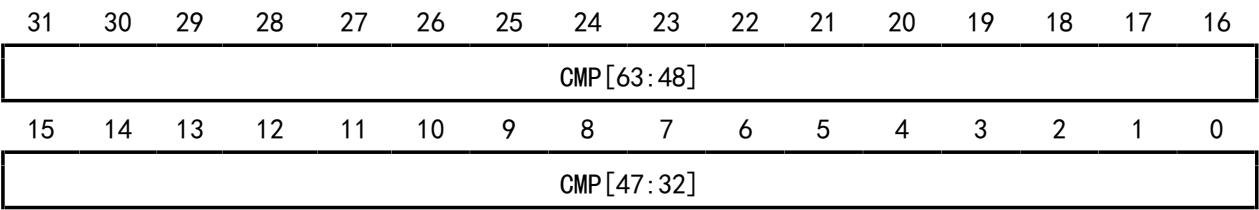
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CMP[31:16]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMP[15:0]															

位	名称	访问	描述	复位值
[31:0]	CMP[31:0]	RW	设置比较计数器值低 32 位。	0

注：寄存器 STK_CMPLR 和寄存器 STK_CMPHR 共同构成了 64 位计数器比较值。

9.5.5.6 计数比较高位寄存器（STK_CMPHR）

偏移地址：0x14



位	名称	访问	描述	复位值
[31:0]	CMP[63:32]	RW	设置比较计数器值高 32 位。	0

注：寄存器 STK_CMPLR 和寄存器 STK_CMPHR 共同构成了 64 位计数器比较值。

第 10 章 GPIO 及其复用功能（GPIO/AFIO）

GPIO 口可以配置成多种输入或输出模式，内置可关闭的上拉或下拉电阻，可以配置成推挽或开漏功能。GPIO 口还可以复用成其他功能。

PA11/PA12 可以复用作为 USB 的 I/O 引脚，有两套互斥的上拉和下拉电阻。当 RB_UC_RST_SIE=1 时作为普通 GPIO，上拉和下拉电阻的控制方法和特性与其他 GPIO 相同。当 RB_UC_RST_SIE=0 时作为 USB 专用引脚 UDM/UDP，USB 上拉电阻约 1.5K，参考表 20-2 实现控制，USB 下拉电阻约 15K，由 R8_USB_CTRL 中的 RB_UH_PD_DIS 控制，均不受 GPIO 控制。

PB6/PB7 可以复用作为 USB PD 的 I/O 引脚，有两套相互独立的上拉和下拉电阻。其中，与其他 GPIO 特性相同的上拉和下拉电阻的控制方法与其他 GPIO 相同。另有一套上拉电流和下拉电阻 R_d （部分封装形式的芯片未内置）用于 Type-C 引脚，由 R16_PORT_CC1/R16_PORT_CC2 控制。另外，AFIO_CR 中的 USBPD_IN_HVT 位，可为 PB6/PB7 选择高阈值输入模式。

10.1 主要特征

端口的每个引脚都可以配置成以下的多种模式之一：

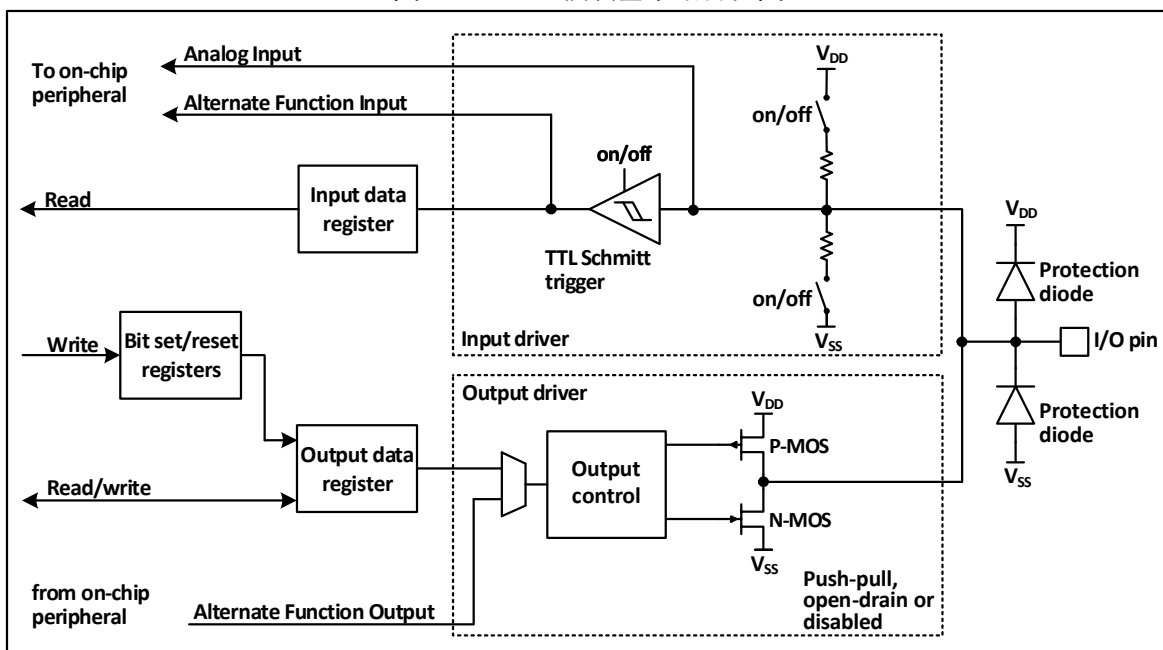
- 浮空输入
- 上拉输入
- 下拉输入
- 模拟输入
- 开漏输出
- 推挽输出
- 复用功能的输入和输出

许多引脚拥有复用功能，很多其他的外设把自己的输出和输入通道映射到这些引脚上，这些复用引脚具体用法需要参照各个外设，而对这些引脚是否复用和是否重映射的内容由本章说明。

10.2 功能描述

10.2.1 概述

图 10-1 GPIO 模块基本结构框图



如图 10-1 所示 I/O 口结构，每个引脚在芯片内部都有两只保护二极管，I/O 口内部可分为输入和输出驱动模块。其中输入驱动有弱上下拉电阻可选，可连接到 AD 等模拟输入的外设；如果输入到数字外设，就需要经过一个 TTL 施密特触发器，再连接到 GPIO 输入寄存器或其他复用外设。而输出驱

动有一对 MOS 管，可通过配置上下的 MOS 管是否使能来将 I/O 口配置成开漏或推挽输出；输出驱动内部也可以配置成由 GPIO 控制输出还是由复用的其他外设控制输出。

10.2.2 GPIO 的初始化功能

刚复位后，GPIO 口运行在初始状态，这时大多数 I/O 口都是运行在浮空输入状态，但也有 HSE 等外设相关的引脚是运行在外设复用的功能上。具体的初始化功能请参照引脚描述相关的章节。

10.2.3 外部中断

所有的 GPIO 口都可以被配置外部中断输入通道，但一个外部中断输入通道最多只能映射到一个 GPIO 引脚上，且外部中断通道的序号必须和 GPIO 端口的位号一致，比如 PA1（或 PB1、PC1、PD1）只能映射到 EXTI1 上，且 EXTI1 只能接受 PA1、PB1、PC1 或 PD1 等其中之一的映射，两方都是一对一的关系。

10.2.4 复用功能

使用复用功能必须要注意：

- 使用输入方向的复用功能，端口必须配置成复用输入模式，上下拉设置可根据实际需要来设置
- 使用输出方向的复用功能，端口必须配置成复用输出模式，推挽或开漏可根据实际情况设置
- 对于双向的复用功能，端口必须配置成复用输出模式，这时驱动器被配置成浮空输入模式

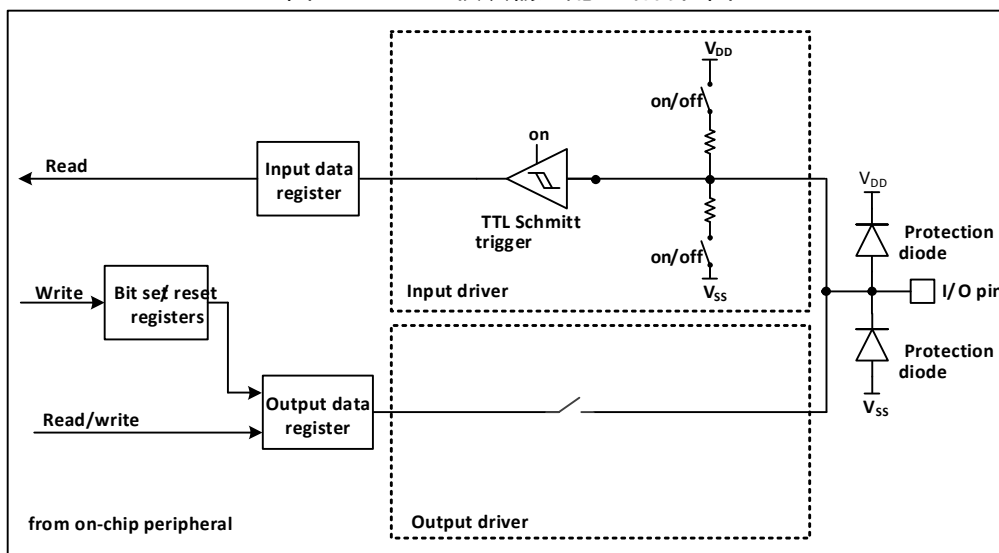
同一个 I/O 口可能有多个外设复用到此管脚，因此为了使各个外设都有最大的发挥空间，外设的复用引脚除了默认复用引脚，还可以进行重映射，重映射到其他的引脚，避开被占用的引脚。

10.2.5 锁定机制

锁定机制可以锁定 I/O 口的配置。经过特定的一个写序列后，选定的 I/O 引脚配置将被锁定，在下一个复位前无法更改。

10.2.6 输入配置

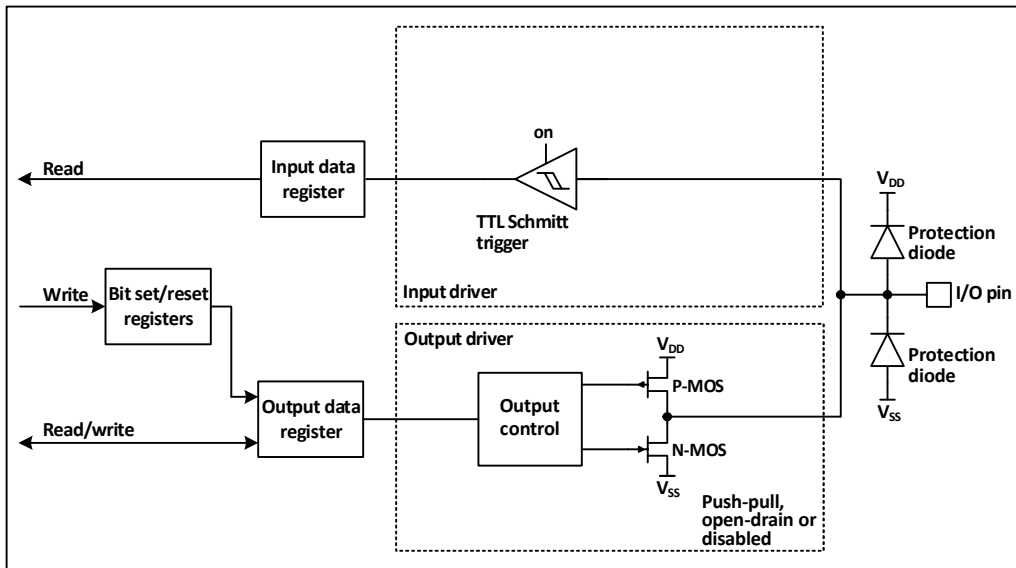
图 10-2 GPIO 模块输入配置结构框图



当 I/O 口配置成输入模式时，输出驱动断开，输入上下拉可选，不连接复用功能和模拟输入。在每个 I/O 口上的数据在每个 PB2 时钟被采样到输入数据寄存器，读取输入数据寄存器对应位即获取了对应引脚的电平状态。

10.2.7 输出配置

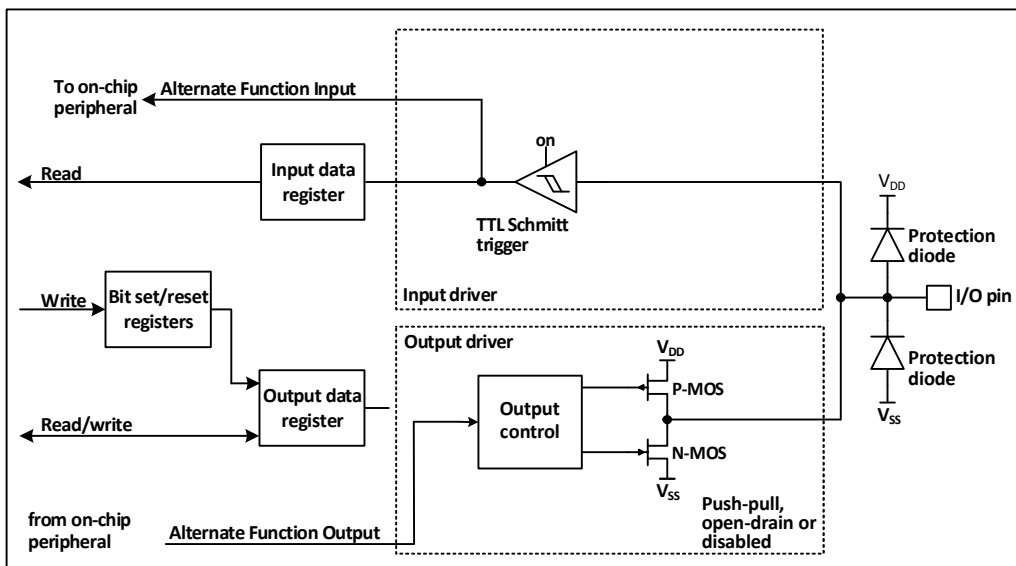
图 10-3 GPIO 模块输出配置结构框图



当 I/O 口配置成输出模式时，输出驱动器中的一对 MOS 可根据需要被配置成推挽或开漏模式，不使用复用功能。输入驱动的上下拉电阻被禁用，TTL 施密特触发器被激活，出现在 I/O 引脚上的电平将会在每个 PB2 时钟被采样到输入数据寄存器，所以读取输入数据寄存器将会得到 I/O 状态，在推挽输出模式时，对输出数据寄存器的访问就会得到最后一次写入的值。

10.2.8 复用功能配置

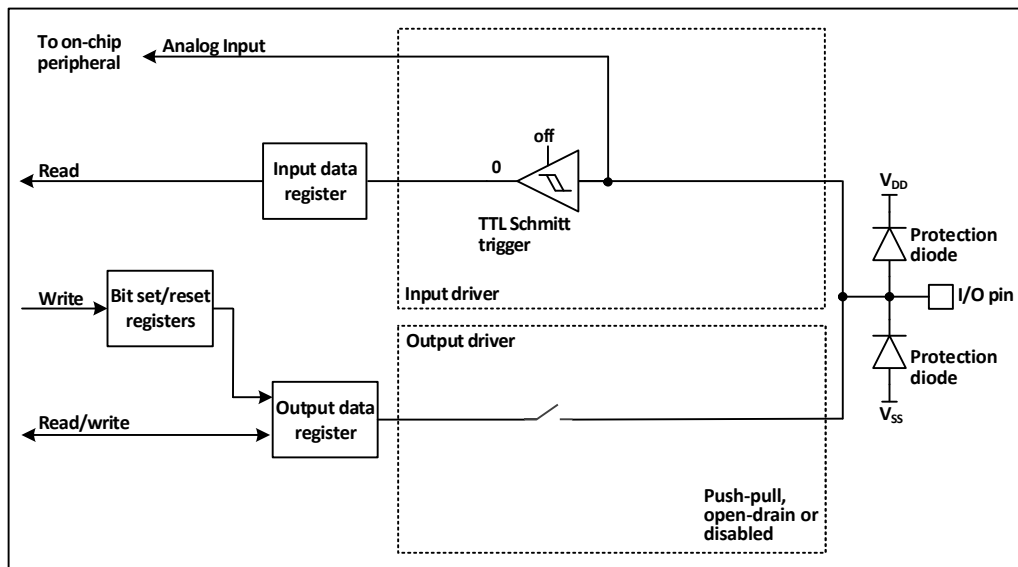
图 10-4 GPIO 模块被其他外设复用时的结构框图



在启用复用功能时，输出驱动器被使能，可以按需要被配置成开漏或推挽模式，施密特触发器也被打开，复用功能的输入和输出线都被连接，但是输出数据寄存器被断开，出现在 I/O 引脚上的电平将会在每个 PB2 时钟被采样到输入数据寄存器，在开漏模式下，读取输入数据寄存器将会得到 I/O 口当前状态；在推挽模式下，读取输出数据寄存器将会得到最后一次写入的值。

10.2.9 模拟输入配置

图 10-5 GPIO 模块作为模拟输入时的配置结构框图



在启用模拟输入时，输出缓冲器被断开，输入驱动中施密特触发器的输入被禁止以防止产生 I/O 口上的消耗，上下拉电阻被禁止，读取输入数据寄存器将一直为 0。

10.2.10 外设的 GPIO 设置

下列表格推荐了各个外设的引脚相应的 GPIO 口配置。

表 10-1 高级定时器（TIM1）

TIM1	配置	GPIO 配置
TIM1_CHx	输入捕获通道 x	浮空输入
	输出比较通道 x	推挽复用输出
TIM1_CHxN	互补输出通道 x	推挽复用输出
TIM1_BKIN	刹车输入	浮空输入
TIM1_ETR	外部触发时钟输入	浮空输入

表 10-2 通用定时器（TIM2/3/4）

TIM2/3/4 引脚	配置	GPIO 配置
TIM2/3/4_CHx	输入捕获通道 x	浮空输入
	输出比较通道 x	推挽复用输出
TIM2/3/4_ETR	外部触发时钟输入	浮空输入

表 10-3 通用同步异步串行收发器（USART）

USART 引脚	配置	GPIO 配置
USARTx_TX	全双工模式	推挽复用输出
	半双工同步模式	开漏复用输出
USARTx_RX	全双工模式	浮空输入或带上拉输入
	半双工同步模式	未使用
USARTx_CK	同步模式	推挽复用输出
USARTx_RTS	硬件流量控制	推挽复用输出
USARTx_CTS	硬件流量控制	浮空输入或带上拉输入

表 10-4 串行外设接口（SPI）模块

SPI 引脚	配置	GPIO 配置
SPIx_SCK	主模式	推挽复用输出
	从模式	浮空输入
SPIx_MOSI	全双工主模式	推挽复用输出
	全双工从模式	浮空输入或带上拉输入
	简单的双向数据线/主模式	推挽复用输出
	简单的双向数据线/从模式	未使用
SPIx_MISO	全双工主模式	浮空输入或带上拉输入
	全双工从模式	推挽复用输出
	简单的双向数据线/主模式	未使用
	简单的双向数据线/从模式	推挽复用输出
SPIx_NSS	硬件主/从模式	浮空、上拉或下拉输入
	硬件主模式/NSS 输出使能	推挽复用输出
	软件模式	未使用

表 10-5 内部集成总线（I2C）模块

I ² C 引脚	配置	GPIO 配置
I ² C_SCL	I ² C 时钟	开漏复用输出
I ² C_SDA	I ² C 数据	开漏复用输出

表 10-6 控制器局域网（CAN）模块

CAN 引脚	GPIO 配置
CANx_TX	推挽复用输出
CANx_RX	浮空输入或上拉输入

表 10-7 USB 主机设备（USBFS）控制器

USBFS 引脚	GPIO 配置
USBFS_DM/USBFS_DP	使能了 USB 模块之后，复用 I/O 口会自动连接到内部 USBFS 收发器

表 10-8 模拟转数字转换器（ADC）

ADC 引脚	GPIO 配置
ADC	模拟输入

表 10-9 其他的 I/O 功能设置

引脚	配置功能	GPIO 配置
TAMPER_RTC	RTC 输出	硬件自动设置
	侵入事件输入	
MC0	时钟输出	推挽复用输出
EXTI	外部中断输入	浮空、上拉或下拉输入

表 10-10 USB PD / type C 控制器

USBPD 引脚	GPIO 配置
CC1/CC2	使能了 USB PD 模块之后，复用 I/O 口会自动连接到内部 PD 收发器

10.2.11 复用功能重映射 GPIO 设置

10.2.11.1 OSC32_IN/OSC32_OUT 作为 GPIO 端口 PC14/PC15

当 LSEON=0 时, LSE 振荡器引脚 OSC32_IN/OSC32_OUT 可以分别用做 GPIO 的 PC14/PC15。

当 LSEON=1 时, 作为 LSE 引脚。

注: 上电默认做普通 IO 功能

10.2.11.2 OSC_IN/OSC_OUT 引脚作为 GPIO 端口 PD0/PD1

OSC_IN/OSC_OUT 可以用做 GPIO 的 PD0/PD1, 通过设置重映射寄存器 1 (AFIO_PCFR1) 实现。

10.2.11.3 定时器复用功能重映射

表 10-11 TIM1 复用功能重映射

复用功能	000	001	010	011	100	101
TIM1_ETR	PA12	PA12	PB6	PA3	PB6	PA13
TIM1_CH1	PA8	PA8	PA1	PA1	PB7	PB1
TIM1_CH2	PA9	PA9	PA7	PA7	PB8	PB8
TIM1_CH3	PA10	PA10	PA14	PA14	PB12	PB6
TIM1_CH4	PA11	PA11	PB1	PB1	PA2	PA3
TIM1_BKIN	PB12	PA6	PA13	PA13	PA12	PA12
TIM1_CH1N	PB13	PA7	PB11	PB11	PA14	PA14
TIM1_CH2N	PB14	PB0	PB0	PB0	PB1	PA1
TIM1_CH3N	PB15	PB1	PB9	PB9	PB9	PB7

表 10-12 TIM2 复用功能重映射

复用功能	000	001	010	011	100	101	111
TIM2_ETR	PA0	PA15	PA0	PA15	PA3	PA12	PA12
TIM2_CH1	PA0	PA15	PA0	PA15	PA3	PA12	PA12
TIM2_CH2	PA1	PB3	PA1	PB3	PA2	PA2	PB8
TIM2_CH3	PA2	PA2	PB10	PB10	PB12	PB12	PA5
TIM2_CH4	PA3	PA3	PB11	PB11	PA6	PA6	PA4

表 10-13 TIM3 复用功能重映射

复用功能	0	1
TIM3_CH1	PA6	PB4
TIM3_CH2	PA7	PB5
TIM3_CH3	PB0	PB0
TIM3_CH4	PB1	PB1

表 10-14 TIM4 复用功能重映射

复用功能	0	1
TIM4_CH1	PB6	PB10
TIM4_CH2	PB7	PB11
TIM4_CH3	PB8	PB8
TIM4_CH4	PB9	PB9

表 10-15 LPTIM 复用功能重映射

复用功能	0	1
LPT_IN1	PB12	PB5
LPT_IN2	PB13	PB7
LPT_ETR	PB14	PB6
LPT_OUT	PB15	PB2

10.2.11.4 USART 复用功能重映射

表 10-16 USART1 复用功能重映射

复用功能	000	001	010	011	100	101
USART1_TX	PA9	PB6	PA4	PA5	PB11	PB12
USART1_RX	PA10	PB7	PA5	PA4	PB9	PA12
USART1_CK	PA8	PA8	PA3	PA6	PA6	PB6
USART1_CTS	PA11	PA11	PA2	PB7	PA14	PB7
USART1_RTS	PA12	PA12	PA13	PB8	PA13	PB8

表 10-17 USART2 复用功能重映射

复用功能	00	10	11
USART2_TX	PA2	PA11	PA12
USART2_RX	PA3	PA12	PA11
USART2_CK	PA4	PA4	PA4
USART2_CTS	PA0	PA0	PA0
USART2_RTS	PA1	PA1	PA1

表 10-18 USART3 复用功能重映射

复用功能	00	10	11
USART3_TX	PB10	PD1	PD0
USART3_RX	PB11	PD0	PD1
USART3_CK	PB12	PB12	PB12
USART3_CTS	PB13	PB13	PB13
USART3_RTS	PB14	PB14	PB14

表 10-19 USART4 复用功能重映射

复用功能	0	1
USART4_TX	PB0	PA5
USART4_RX	PB1	PB5
USART4_CK	PB2	PA6
USART4_CTS	PB3	PA7
USART4_RTS	PB4	PA15

10.2.11.5 SPI 复用功能重映射

表 10-20 SPI1 复用功能重映射

复用功能	00	01	10	11
------	----	----	----	----

SPI1_NSS	PA4	PA15	PA12	PB12
SPI1_SCK	PA5	PB3	PB6	PB6
SPI1_MISO	PA6	PB4	PB8	PB8
SPI1_MOSI	PA7	PB5	PB7	PB7

10.2.11.6 I2C 复用功能重映射

表 10-21 I2C1 复用功能重映射

复用功能	00	10	11
I2C1_SCL	PB6	PA13	PB9
I2C1_SDA	PB7	PA12	PB11

10.2.11.7 CAN 复用功能重映射

表 10-22 CAN 复用功能重映射

复用功能	00	10	11
TX	PA12	PB9	PD1
RX	PA11	PB8	PD0

10.2.12 BC 功能配置

AFIO_CR 寄存器包括 BC 接口 UDM/UDP 引脚的控制位、比较器及电压输出的控制位。

设置 RB_UC_RST_SIE=0 开启 USB、设置 R8_UHOST_CTRL 中的 RB_UH_PD_DIS=0 使能内部 15K 下拉、设置 UDM/UDP 的 BC_VSRC=1，可使能 BC 协议源电压 V_{BC_SRC} 输出。

设置 UDM/UDP 的 BC_CMPE=1，可使能 BC 协议比较器；对 UDM/UDP 的 BC_CMP0 位进行读操作，可获取引脚电压与 BC 协议参考值 V_{BC_REF} 的比较状态。

AFIO_CR 寄存器中还包括了 USBPD_IN_HVT 位，该位可对 PD 引脚 PB6/PB7 高阈值输入模式进行设置。该位置 1 时为高阈值输入，典型值约 2.2V，可降低 PD 通讯时的 I/O 功耗；该位置 0 时为正常 GPIO 阈值输入。

10.3 寄存器描述

10.3.1 GPIO 的寄存器描述

除非特殊说明，GPIO 的寄存器必须以字的方式操作（以 32 位来操作这些寄存器）。

表 10-23 GPIO 相关寄存器列表

名称	访问地址	描述	复位值
R32_GPIOA_CFGLR	0x40010800	PA 端口配置寄存器低位	0x44444444
R32_GPIOB_CFGLR	0x40010C00	PB 端口配置寄存器低位	0x44444444
R32_GPIOC_CFGLR	0x40011000	PC 端口配置寄存器低位	0x44444444
R32_GPIOD_CFGLR	0x40011400	PD 端口配置寄存器低位	0x44444444
R32_GPIOA_CFGHR	0x40010804	PA 端口配置寄存器高位	0x44444444
R32_GPIOB_CFGHR	0x40010C04	PB 端口配置寄存器高位	0x44444444
R32_GPIOC_CFGHR	0x40011004	PC 端口配置寄存器高位	0x44444444
R32_GPIOD_CFGHR	0x40011404	PD 端口配置寄存器高位	0x44444444
R32_GPIOA_INDR	0x40010808	PA 端口输入数据寄存器	0x0000XXXX
R32_GPIOB_INDR	0x40010C08	PB 端口输入数据寄存器	0x0000XXXX
R32_GPIOC_INDR	0x40011008	PC 端口输入数据寄存器	0x0000XXXX

R32_GPIOD_INDR	0x40011408	PD 端口输入数据寄存器	0x0000XXXX
R32_GPIOA_OUTDR	0x4001080C	PA 端口输出数据寄存器	0x00000000
R32_GPIOB_OUTDR	0x40010C0C	PB 端口输出数据寄存器	0x00000000
R32_GPIOC_OUTDR	0x4001100C	PC 端口输出数据寄存器	0x00000000
R32_GPIOD_OUTDR	0x4001140C	PD 端口输出数据寄存器	0x00000000
R32_GPIOA_BSHR	0x40010810	PA 端口置位/复位寄存器	0x00000000
R32_GPIOB_BSHR	0x40010C10	PB 端口置位/复位寄存器	0x00000000
R32_GPIOC_BSHR	0x40011010	PC 端口置位/复位寄存器	0x00000000
R32_GPIOD_BSHR	0x40011410	PD 端口置位/复位寄存器	0x00000000
R32_GPIOA_BCR	0x40010814	PA 端口复位寄存器	0x00000000
R32_GPIOB_BCR	0x40010C14	PB 端口复位寄存器	0x00000000
R32_GPIOC_BCR	0x40011014	PC 端口复位寄存器	0x00000000
R32_GPIOD_BCR	0x40011414	PD 端口复位寄存器	0x00000000
R32_GPIOA_LCKR	0x40010818	PA 端口锁定配置寄存器	0x00000000
R32_GPIOB_LCKR	0x40010C18	PB 端口锁定配置寄存器	0x00000000
R32_GPIOC_LCKR	0x40011018	PC 端口锁定配置寄存器	0x00000000
R32_GPIOD_LCKR	0x40011418	PD 端口锁定配置寄存器	0x00000000

10.3.1.1 GPIO 配置寄存器低位 (GPIOx_CFGLR) (x=A/B/C/D)

偏移地址: 0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CNF7[1:0]	MODE7[1:0]	CNF6[1:0]	MODE6[1:0]	CNF5[1:0]	MODE5[1:0]	CNF4[1:0]	MODE4[1:0]								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNF3[1:0]	MODE3[1:0]	CNF2[1:0]	MODE2[1:0]	CNF1[1:0]	MODE1[1:0]	CNF0[1:0]	MODE0[1:0]								

位	名称	访问	描述	复位值
[31:30] [27:26] [23:22] [19:18] [15:14] [11:10] [7:6] [3:2]	CNFy[1:0]	RW	(y=0-7), 端口 x 的配置位, 通过这些位配置相应的端口。 在输入模式时 (MODE=00b): 00: 模拟输入模式; 01: 浮空输入模式; 10: 带有上下拉模式。 11: 保留。 在输出模式 (MODE>00b): 00: 通用推挽输出模式; 01: 通用开漏输出模式; 10: 复用功能推挽输出模式; 11: 复用功能开漏输出模式。	01b
[29:28] [25:24] [21:20] [17:16] [13:12] [9:8]	MODEy[1:0]	RW	(y=0-7), 端口 x 模式选择, 通过这些位配置相应的端口。 00: 输入模式; 01: 输出模式, 最大速度 10MHz; 10: 输出模式, 最大速度 2MHz; 11: 输出模式, 最大速度 50MHz。	00b

[5:4]				
[1:0]				

10.3.1.2 GPIO 配置寄存器高位 (GPIOx_CFGHR) (x=A/B/C/D)

偏移地址: 0x04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CNF15[1:0]	MODE15[1:0]	CNF14[1:0]	MODE14[1:0]	CNF13[1:0]	MODE13[1:0]	CNF12[1:0]	MODE12[1:0]								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNF11[1:0]	MODE11[1:0]	CNF10[1:0]	MODE10[1:0]	CNF9[1:0]	MODE9[1:0]	CNF8[1:0]	MODE8[1:0]								

位	名称	访问	描述	复位值
[31:30] [27:26] [23:22] [19:18] [15:14] [11:10] [7:6] [3:2]	CNFy[1:0]	RW	(y=8-15), 端口 x 的配置位, 通过这些位配置相应的端口。 在输入模式时 (MODE=00b): 00: 模拟输入模式; 01: 浮空输入模式; 10: 带有上下拉模式。 11: 保留。 在输出模式 (MODE>00b): 00: 通用推挽输出模式; 01: 通用开漏输出模式; 10: 复用功能推挽输出模式; 11: 复用功能开漏输出模式。	01b
[29:28] [25:24] [21:20] [17:16] [13:12] [9:8] [5:4] [1:0]	MODEy[1:0]	RW	(y=8-15), 端口 x 的模式位, 通过这些位配置相应的端口。 00: 输入模式; 01: 输出模式, 最大速度 10MHz; 10: 输出模式, 最大速度 2MHz; 11: 输出模式, 最大速度 50MHz。	00b

10.3.1.3 端口输入寄存器 (GPIOx_INDR) (x=A/B/C/D)

偏移地址: 0x08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDR15	IDR14	IDR13	IDR12	IDR11	IDR10	IDR9	IDR8	IDR7	IDR6	IDR5	IDR4	IDR3	IDR2	IDR1	IDR0

位	名称	访问	描述	复位值
[31:16]	Reserved	RO	保留。	0
[15:0]	IDRy	RO	(y=0-15), 端口输入数据。这些位只读并只能以 16 位形式读出。读出的值就是对应位的高	x

			低状态。	
--	--	--	------	--

10.3.1.4 端口输出寄存器 (GPIOx_OUTDR) (x=A/B/C/D)

偏移地址: 0x0C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ODR15	ODR14	ODR13	ODR12	ODR11	ODR10	ODR9	ODR8	ODR7	ODR6	ODR5	ODR4	ODR3	ODR2	ODR1	ODR0

位	名称	访问	描述	复位值
[31:16]	Reserved	RO	保留。	0
[15:0]	ODRy	RW	对于输出模式: (y=0-15), 端口输出的数据。这些数据只能以 16 位的形式操作。IO 口对外输出这些寄存器的值。 对于带有上下拉的输入模式: 0: 下拉输入; 1: 上拉输入。	0

10.3.1.5 端口复位/置位寄存器 (GPIOx_BSHR) (x=A/B/C/D)

偏移地址: 0x10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BS15	BS14	BS13	BS12	BS11	BS10	BS9	BS8	BS7	BS6	BS5	BS4	BS3	BS2	BS1	BS0

位	名称	访问	描述	复位值
[31:16]	BRy	WO	(y=0-15), 对这些位置位会清除对应的 OUTDR 位, 写 0 不产生影响。这些位只能以 16 位的形式访问。如果同时设置了 BR 和 BS 位, 则 BS 位起作用。	0
[15:0]	BSy	WO	(y=0-15), 对这些位置位会使对应的 OUTDR 位置位, 写 0 不产生影响。这些位只能以 16 位的形式访问。如果同时设置了 BR 和 BS 位, 则 BS 位起作用。	0

10.3.1.6 端口复位寄存器 (GPIOx_BCR) (x=A/B/C/D)

偏移地址: 0x14

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0

位	名称	访问	描述	复位值
[31:16]	Reserved	R0	保留。	0
[15:0]	BRy	WO	(y=0-15)，对这些位置位会清除对应的 OUTDR 位，写 0 不产生影响。这些位只能以 16 位的形式访问。	0

10.3.1.7 配置锁定寄存器 (GPIOx_LCKR) (x=A/B/C/D)

偏移地址: 0x18

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															LCKK
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LCK15	LCK14	LCK13	LCK12	LCK11	LCK10	LCK9	LCK8	LCK7	LCK6	LCK5	LCK4	LCK3	LCK2	LCK1	LCK0

位	名称	访问	描述	复位值
[31:17]	Reserved	R0	保留	0
16	LCKK	RW	锁定键，它可以通过特定的序列写入实现锁定，但它可以随时读出。它读出为 0 时表示未锁定生效，读出 1 时表示锁定生效。 锁定键的写入序列为：写 1-写 0-写 1-读 0-读 1，最后一步非必要，但是可以用以确认锁定键已经激活。 在写入序列时任何错误都不会使激活锁定，且在写入序列时，不能更改 LCK[15:0] 的值。锁定生效后，只有在下次复位后才能更改端口的配置。	0
[15:0]	LCKy	RW	(y=0-15)，这些位为 1 时表示锁定对应端口的配置。只能在 LCKK 未锁定前改变这些位。锁定的配置指的是配置寄存器 GPIOx_CFGLR 和 GPIOx_CFGHR。	0

注：当对相应的端口位执行了 LOCK 序列后，在下次系统复位之前将不能再更改端口位的配置。

10.3.2 AFIO 寄存器

除非特殊说明，AFIO 的寄存器必须以字的方式操作（以 32 位来操作这些寄存器）。

表 10-24 AFIO 相关寄存器列表

名称	访问地址	描述	复位值
R32_AFIO_ECR	0x40010000	事件控制寄存器	0x00000000
R32_AFIO_PCFR1	0x40010004	重映射寄存器 1	0x00000000
R32_AFIO_EXTICR1	0x40010008	外部中断配置寄存器 1	0x00000000
R32_AFIO_EXTICR2	0x4001000C	外部中断配置寄存器 2	0x00000000
R32_AFIO_EXTICR3	0x40010010	外部中断配置寄存器 3	0x00000000
R32_AFIO_EXTICR4	0x40010014	外部中断配置寄存器 4	0x00000000
R32_AFIO_CR	0x40010018	控制寄存器	0x00000000
R32_AFIO_PCFR2	0x4001001C	重映射寄存器 2	0x00000000

10.3.2.1 事件控制寄存器 (AFIO_ECR)

偏移地址: 0x00

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

位	名称	访问	描述	复位值
[31:8]	Reserved	R0	保留。	0
7	EVOE	RW	允许事件输出位, 对该位置位会使内核的 EVENTOUT 连接到 PORT 和 PIN 选定的 IO 口。	0
[6:4]	PORT[2:0]	RW	用于选择内核输出 EVENTOUT 的端口: 000: 选择 PA 口; 001: 选择 PB 口; 010: 选择 PC 口; 011: 选择 PD 口; 其他: 保留。	000b
[3:0]	PIN[3:0]	RW	此位的值用来确定选择内核输出 EVENTOUT 到端口的具体引脚号, 值 0~15 分别对应 PORT 中选定的 Px 的第 0~15 号引脚。	0

10.3.2.2 重映射寄存器 1 (AFIO_PCFR1)

偏移地址: 0x04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved					SW_CFG[2:0]			Reserved							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD01_RM	CAN_RM[1:0]	TIM4_RM	Reserved	TIM3_RM	TIM2_RM[1:0]	TIM1_RM[1:0]	TIM1_RM[1:0]	USART3_RM[1:0]	USART2_RM	USART1_RM	I2C1_RM	SPI1_RM			

位	名称	访问	描述	复位值
[31:27]	Reserved	R0	保留。	0
[26:24]	SW_CFG[2:0]	WO	这些位用以配置 SW 功能和跟踪功能的 IO 口。SDI 是访问内核的调试接口。系统复位后总是作为 SDI 端口。 0xx: 启用 SDI; 100: 关闭 SDI, 作为 GPIO 功能; 其他: 无效。	000b
[23:16]	Reserved	R0	保留。	0
15	PD01_RM	RW	引脚 PD0&PD1 重映射位, 该位可由用户读写。它控制 PD0 和 PD1 的 GPIO 功能是否进行重映射, 即 PD0&PD1 映射到 OSC_IN&OSC_OUT。 0: 引脚作为晶振引脚使用; 1: 引脚作为 GPIO 口使用;	0
[14:13]	CAN_RM[1:0]	RW	CAN 复用功能重映射位, 这些位可由用户读写。	00b

			控制 CAN_RX 和 CAN_TX 的重映射： 00: CAN_RX 映射到 PA11, CAN_TX 映射到 PA12; 10: CAN_RX 映射到 PB8, CAN_TX 映射到 PB9; 01: 保留; 11: CAN_RX 映射到 PD0, CAN_TX 映射到 PD1;	
12	TIM4_RM	RW	定时器 4 的重映射位，该位可由用户读写。它控制定时器 4 的通道 1 至 4 在 GPIO 端口的重映射： 0: 默认映射 (CH1/PB6, CH2/PB7, CH3/PB8, CH4/PB9) ; 1: 重映射 (CH1/PB10, CH2/PB11, CH3/PPB8, CH4/PB9) 。	0
11	Reserved	RO	保留。	0
10	TIM3_RM	RW	定时器 3 的重映射位，这些位可由用户读写。它控制定时器 3 的通道 1 至 4 在 GPIO 端口的重映射： 0: 默认映射 (CH1/PA6, CH2/PA7, CH3/PB0, CH4/PB1) ; 1: 部分映射 (CH1/PB4, CH2/PB5, CH3/PB0, CH4/PB1) 。	0
[9:8]	TIM2_RM[1:0]	RW	定时器 2 的重映射位，结合重映射寄存器 2 的 TIM2_RM_H 字段使用，{TIM2_RM_H, TIM2_RM}。这些位可由用户读写。它控制定时器 2 的通道 1 至 4 和外部触发 (ETR) 在 GPIO 端口的映射： 000: 默认映射 (CH1/ETR/PA0, CH2/PA1, CH3/PA2, CH4/PA3) ; 001: 部分映射 (CH1/ETR/PA15, CH2/PB3, CH3/PA2, CH4/PA3) ; 010: 部分映射 (CH1/ETR/PA0, CH2/PA1, CH3/PB10, CH4/PB11) ; 011: 完全映射 (CH1/ETR/PA15, CH2/PB3, CH3/PB10, CH4/PB11) ; 100: 完全映射 (CH1/ETR/PA3, CH2/PA2, CH3/PB12, CH4/PA6) ; 101: 完全映射 (CH1/ETR/PA12, CH2/PA2, CH3/PB12, CH4/PA6) ; 111: 完全映射 (CH1/ETR/PA12, CH2/PB8, CH3/PA5, CH4/PA4) 。	00b
[7:6]	TIM1_RM[1:0]	RW	定时器 1 的重映射位，结合重映射寄存器 2 的 TIM1_RM_H 字段使用，{TIM1_RM_H, TIM1_RM}。这些位可由用户读写。 它控制定时器 1 的通道 1 至 4、1N 至 3N、外部触发 (ETR) 和刹车输入 (BKIN) 在 GPIO 端口的映射： 000: 默认映射 (ETR/PA12, CH1/PA8, CH2/PA9, CH3/PA10, CH4/PA11, BKIN/PB12, CH1N/PB13,	00b

			<p>CH2N/PB14, CH3N/PB15)</p> <p>001: 部分映射 (ETR/PA12, CH1/PA8, CH2/PA9, CH3/PA10, CH4/PA11, BKIN/PA6, CH1N/PA7, CH2N/PB0, CH3N/PB1)</p> <p>010: 完全映射 (ETR/PB6, CH1/PA1, CH2/PA7, CH3/PA14, CH4/PB1, BKIN/PA13, CH1N/PB11, CH2N/PB0, CH3N/PB9)</p> <p>011: 完全映射 (ETR/PA3, CH1/PA1, CH2/PA7, CH3/PA14, CH4/PB1, BKIN/PA13, CH1N/PB11, CH2N/PB0, CH3N/PB9)</p> <p>100: 完全映射 (ETR/PB6, CH1/PB7, CH2/PB8, CH3/PB12, CH4/PA2, BKIN/PA12, CH1N/PA14, CH2N/PB1, CH3N/PB9)</p> <p>101: 完全映射 (ETR/PA13, CH1/PB1, CH2/PB8, CH3/PB6, CH4/PA3, BKIN/PA12, CH1N/PA14, CH2N/PA1, CH3N/PB7)</p> <p>111: 仅控制 LSI 输入, 用于 LSI 校准</p>	
[5:4]	USART3_RM[1:0]	RW	<p>USART3 的重映射位, 这些位可由用户读写。它控制 USART3 的 TX、RX、CK、CTS 和 RTS 复用功能在 GPIO 端口的映射:</p> <p>00: 默认映射 (TX/PB10, RX/PB11, CK/PB12, CTS/PB13, RTS/PB14)</p> <p>10: 部分重映射 (TX/PD1, RX/PD0, CK/PB12, CTS/PB13, RTS/PB14)</p> <p>11: 部分重映射 (TX/PD0, RX/PD1, CK/PB12, CTS/PB13, RTS/PB14)</p>	00b
3	USART2_RM	RW	<p>USART2 的重映射位, 和重映射寄存器 2 的 USART2_RM_H 字段结合, {USART2_RM_H, USART2_RM}。该位可由用户读写。它控制 USART2 的 TX、RX、CK、CTS 和 RTS 复用功能在 GPIO 端口的映射:</p> <p>00: 默认映射 (TX/PA2, RX/PA3, CK/PA4, CTS/PA0, RTS/PA1)</p> <p>01: 保留</p> <p>10: 部分映射 (TX/PA11, RX/PA12, CK/PA4, CTS/PA0, RTS/PA1)</p> <p>11: 部分映射 (TX/PA12, RX/PA11, CK/PA4, CTS/PA0, RTS/PA1)</p>	0
2	USART1_RM	RW	<p>USART1 的重映射位, 和重映射寄存器 2 的 USART1_RM_H 字段结合, {USART1_RM_H, USART1_RM}。该位可由用户读写。它控制 USART1 的 TX、RX、CK、CTS 和 RTS 复用功能在 GPIO 端口的映射:</p> <p>000: 默认映射 (TX/PA9, RX/PA10, CK/PA8, CTS/PA11, RTS/PA12)</p> <p>001: 部分映射 (TX/PB6, RX/PB7, CK/PA8,</p>	0

			CTS/PA11, RTS/PA12) 010: 完全映射 (TX/PA4, RX/PA5, CK/PA3, CTS/PA2, RTS/PA13) 011: 完全映射 (TX/PA5, RX/PA4, CK/PA6, CTS/PB7, RTS/PB8) 100: 完全映射 (TX/PB11, RX/PB9, CK/PA6, CTS/PA14, RTS/PA13) 101: 完全映射 (TX/PB12, RX/PA12, CK/PB6, CTS/PB7, RTS/PB8)	
1	I2C1_RM	RW	I2C1 的重映射, 和重映射寄存器 2 的 I2C1_RM_H 字段结合, {I2C1_RM_H, I2C1_RM}。该位可由用户读写。它控制 I2C1 的 SCL 和 SDA 复用功能在 GPIO 端口的映射: 00: 默认映射 (SCL/PB6, SDA/PB7) 10: 完全映射 (SCL/PA13, SDA/PA12) 11: 完全映射 (SCL/PB9, SDA/PB11)	0
0	SPI1_RM	RW	SPI1 的重映射, 和重映射寄存器 2 的 SPI1_RM_H 字段结合, {SPI1_RM_H, SPI1_RM}。该位可由用户读写。它控制 SPI1 的 NSS、SCK、MISO 和 MOSI 复用功能在 GPIO 端口的映射: 00: 默认映射 (NSS/PA4, SCK/PA5, MISO/PA6, MOSI/PA7) 01: 完全映射 (NSS/PA15, SCK/PB3, MISO/PB4, MOSI/PB5) 10: 完全映射 (NSS/PA12, SCK/PB6, MISO/PB8, MOSI/PB7) 11: 完全映射 (NSS/PB12, SCK/PB6, MISO/PB8, MOSI/PB7)	0

10.3.2.3 外部中断配置寄存器 1 (AFIO_EXTICR1)

偏移地址: 0x08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI3[3:0]				EXTI2[3:0]				EXTI1[3:0]				EXTI0[3:0]			

位	名称	访问	描述	复位值
[31:16]	Reserved	RO	保留。	0
[15:12] [11:8] [7:4] [3:0]	EXTIx[3:0]	RW	(x=0-3), 外部中断输入引脚配置位。用以决定外部中断引脚映射到哪个端口的引脚上: 0000: PA 引脚的第 x 个引脚; 0001: PB 引脚的第 x 个引脚; 0010: PC 引脚的第 x 个引脚; 0011: PD 引脚的第 x 个引脚;	0000b

			其他：保留。	
--	--	--	--------	--

10.3.2.4 外部中断配置寄存器 2 (AFIO_EXTICR2)

偏移地址：0x0C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI7[3:0]				EXTI6[3:0]				EXTI5[3:0]				EXTI4[3:0]			

位	名称	访问	描述	复位值
[31:16]	Reserved	R0	保留。	0
[15:12] [11:8] [7:4] [3:0]	EXTI _x [3:0]	RW	(x=4-7)，外部中断输入引脚配置位。用以决定外部中断引脚映射到哪个端口的引脚上： 0000：PA 引脚的第 x 个引脚； 0001：PB 引脚的第 x 个引脚； 0010：PC 引脚的第 x 个引脚； 0011：PD 引脚的第 x 个引脚； 其他：保留。	0000b

10.3.2.5 外部中断配置寄存器 3 (AFIO_EXTICR3)

偏移地址：0x10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI11[3:0]				EXTI10[3:0]				EXTI9[3:0]				EXTI8[3:0]			

位	名称	访问	描述	复位值
[31:16]	Reserved	R0	保留。	0
[15:12] [11:8] [7:4] [3:0]	EXTI _x [3:0]	RW	(x=8-11)，外部中断输入引脚配置位。用以决定外部中断引脚映射到哪个端口的引脚上： 0000：PA 引脚的第 x 个引脚； 0001：PB 引脚的第 x 个引脚； 0010：PC 引脚的第 x 个引脚； 0011：PD 引脚的第 x 个引脚； 其他：保留。	0000b

10.3.2.6 外部中断配置寄存器 4 (AFIO_EXTICR4)

偏移地址：0x14

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI15[3:0]				EXTI14[3:0]				EXTI13[3:0]				EXTI12[3:0]			

位	名称	访问	描述	复位值
[31:16]	Reserved	R0	保留。	0
[15:12] [11:8] [7:4] [3:0]	EXTI _x [3:0]	RW	(x=12-15), 外部中断输入引脚配置位。用以决定外部中断引脚映射到哪个端口的引脚上: 0000: PA 引脚的第 x 个引脚; 0001: PB 引脚的第 x 个引脚; 0010: PC 引脚的第 x 个引脚; 0011: PD 引脚的第 x 个引脚; 其他: 保留。	0000b

10.3.2.7 控制寄存器 (AFIO_CR)

偏移地址: 0x18

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved										UDM_BC_CMPO	UDP_BC_CMPO	UDM_BC_CMPE	UDP_BC_CMPE	UDM_BC_VSRC	UDP_BC_VSRC
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										USBPD_IN_HVT	Reserved				

位	名称	访问	描述	复位值
[31:22]	Reserved	R0	保留。	0
21	UDM_BC_CMPO	R0	PA11/UDM 引脚 BC 协议比较器状态: 1: PA11 电压高于 BC 协议参考值 V_{BC_REF} ; 0: PA11 电压低于 BC 协议参考值 V_{BC_REF} 。	0
20	UDP_BC_CMPO	R0	PA12/UDP 引脚 BC 协议比较器状态: 1: PA12 电压高于 BC 协议参考值 V_{BC_REF} ; 0: PA12 电压低于 BC 协议参考值 V_{BC_REF} 。	0
19	UDM_BC_CMPE	RW	PA11/UDM 引脚 BC 协议比较器使能: 1: 使能 0: 关闭	0
18	UDP_BC_CMPE	RW	PA12/UDP 引脚 BC 协议比较器使能: 1: 使能 0: 关闭	0
17	UDM_BC_VSRC	RW	PA11/UDM 引脚 BC 协议源电压使能: 1: PA11 输出 BC 协议源电压 V_{BC_SRC} ; 0: 禁止输出。	0
16	UDP_BC_VSRC	RW	PA12/UDP 引脚 BC 协议源电压使能: 1: PA12 输出 BC 协议源电压 V_{BC_SRC} ; 0: 禁止输出。	0
[15:10]	Reserved	R0	保留。	0
9	USBPD_IN_HVT	RW	PD 引脚 PB6/PB7 高阈值输入模式: 1: 高阈值输入, 典型值约 2.2V, 可降低 PD 通讯时的 I/O 功耗;	0

			0: 正常 GPIO 阈值输入。	
[8:0]	Reserved	RW	保留。	0

10.3.2.8 重映射寄存器 2 (AFIO_PCFR2)

偏移地址: 0x10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved						LPTIM_RM	SPI1_RM_H	I2C1_RM_H	TIM1_RM_H	TIM2_RM_H	USART1_RM_H	USART2_RM_H	Reserved	USART4_RM	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved															

位	名称	访问	描述	复位值
[31:26]	Reserved	RO	保留。	0
25	LPTIM_RM	RW	LPTIM 重映射 0: 默认映射 (CH1/PB12, CH2/PB13, ETR/PB14, 0C/PB15) 1: 重映射 (CH1/PB5, CH2/PB7, ETR/PB6, 0C/PB2)	0
24	SPI1_RM_H	RW	SPI1 重映射, 和重映射寄存器 1 的 SPI1_RM 字段结合 {SPI1_RM_H, SPI1_RM} 00: 默认映射 (NSS/PA4, SCK/PA5, MISO/PA6, MOSI/PA7) 01: 完全映射 (NSS/PA15, SCK/PB3, MISO/PB4, MOSI/PB5) 10: 完全映射 (NSS/PA12, SCK/PB6, MISO/PB8, MOSI/PB7) 11: 完全映射 (NSS/PB12, SCK/PB6, MISO/PB8, MOSI/PB7)	0
23	I2C1_RM_H	RW	I2C1 的重映射, 和重映射寄存器 1 的 I2C1_RM 字段结合, {I2C1_RM_H, I2C1_RM}。该位可由用户读写。它控制 I2C1 的 SCL 和 SDA 复用功能在 GPIO 端口的映射: 00: 默认映射 (SCL/PB6, SDA/PB7, SMBA/PB5) 10: 完全映射 (SCL/PA13, SDA/PA12, SMBA/PB5) 11: 完全映射 (SCL/PB9, SDA/PB11, SMBA/PB5)	0
22	TIM1_RM_H	RW	定时器 1 的重映射位, 结合重映射寄存器 1 的 TIM1_RM 字段使用, {TIM1_RM_H, TIM1_RM}。这些位可由用户读写。 它控制定时器 1 的通道 1 至 4、1N 至 3N、外部触发 (ETR) 和刹车输入 (BKIN) 在 GPIO 端口的映射: 000: 默认映射 (ETR/PA12, CH1/PA8, CH2/PA9, CH3/PA10, CH4/PA11, BKIN/PB12, CH1N/PB13, CH2N/PB14, CH3N/PB15) 001: 完全映射 (ETR/PA12, CH1/PA8, CH2/PA9, CH3/PA10, CH4/PA11, BKIN/PA6, CH1N/PA7, CH2N/PB0, CH3N/PB1) 010: 完全映射 (ETR/PB6, CH1/PA1, CH2/PA7, CH3/PA14, CH4/PB1, BKIN/PA13, CH1N/PB11, CH2N/PB0, CH3N/PB9)	0

			<p>011: 完全映射 (ETR/PA3, CH1/PA1, CH2/PA7, CH3/PA14, CH4/PB1, BKIN/PA13, CH1N/PB11, CH2N/PB0, CH3N/PB9)</p> <p>100: 完全映射 (ETR/PB6, CH1/PB7, CH2/PB8, CH3/PB12, CH4/PA2, BKIN/PA12, CH1N/PA14, CH2N/PB1, CH3N/PB9)</p> <p>101: 完全映射 (ETR/PA13, CH1/PB1, CH2/PB8, CH3/PB6, CH4/PA3, BKIN/PA12, CH1N/PA14, CH2N/PA1, CH3N/PB7)</p> <p>111: 仅控制 LSI 输入, 用于 LSI 校准</p>	
21	TIM2_RM_H	RW	<p>定时器 2 的重映射位, 结合重映射寄存器 1 的 TIM2_RM 字段使用, {TIM2_RM_H, TIM2_RM}。这些位可由用户读写。它控制定时器 2 的通道 1 至 4 和外部触发 (ETR) 在 GPIO 端口的映射:</p> <p>000: 默认映射 (CH1/ETR/PA0, CH2/PA1, CH3/PA2, CH4/PA3);</p> <p>001: 部分映射 (CH1/ETR/PA15, CH2/PB3, CH3/PA2, CH4/PA3);</p> <p>010: 部分映射 (CH1/ETR/PA0, CH2/PA1, CH3/PB10, CH4/PB11);</p> <p>011: 完全映射 (CH1/ETR/PA15, CH2/PB3, CH3/PB10, CH4/PB11);</p> <p>100: 完全映射 (CH1/ETR/PA3, CH2/PA2, CH3/PB12, CH4/PA6);</p> <p>101: 完全映射 (CH1/ETR/PA12, CH2/PA2, CH3/PB12, CH4/PA6);</p> <p>111: 完全映射 (CH1/ETR/PA12, CH2/PB8, CH3/PA5, CH4/PA4)。</p>	0
[20:19]	USART1_RM_H	RW	<p>USART1 的重映射位, 和重映射寄存器 1 的 USART1_RM 字段结合, {USART1_RM_H, USART1_RM}。该位可由用户读写。它控制 USART1 的 CTS、RTS、CK、TX 和 RX 复用功能在 GPIO 端口的映射:</p> <p>000: 默认映射 (TX/PA9, RX/PA10, CK/PA8, CTS/PA11, RTS/PA12)</p> <p>001: 完全映射 (TX/PB6, RX/PB7, CK/PA8, CTS/PA11, RTS/PA12)</p> <p>010: 完全映射 (TX/PA4, RX/PA5, CK/PA3, CTS/PA2, RTS/PA13)</p> <p>011: 完全映射 (TX/PA5, RX/PA4, CK/PA6, CTS/PB7, RTS/PB8)</p> <p>100: 完全映射 (TX/PB11, RX/PB9, CK/PA6, CTS/PA14, RTS/PA13)</p> <p>101: 完全映射 (TX/PB12, RX/PA12, CK/PB6, CTS/PB7, RTS/PB8)</p>	00b
18	USART2_RM_H	RW	<p>USART2 的重映射位, 和重映射寄存器 1 的 USART2_RM 字段结合, {USART2_RM_H, USART2_RM}。该位可由用户读写。它控制 USART2 的 CTS、RTS、CK、TX 和 RX 复用功能在 GPIO 端口的映射:</p> <p>00: 默认映射 (TX/PA2, RX/PA3, CK/PA4, CTS/PA0,</p>	0

			RTS/PA1) 01: 保留 10: 部分映射 (TX/PA11, RX/PA12, CK/PA4, CTS/PA0, RTS/PA1) 11: 部分映射 (TX/PA12, RX/PA11, CK/PA4, CTS/PA0, RTS/PA1)	
17	Reserved	RW	保留。	0
16	USART4_RM	RW	USART4 的重映射位。该位可由用户读写。它控制 USART2 的 TX、RX、CK、CTS 和 RTS 复用功能在 GPIO 端口的映射: 0: 默认映射 (TX/PB0, RX/PB1, CK/PB2, CTS/PB3, RTS/PB4) 1: 完全映射 (TX/PA5, RX/PB5, CK/PA6, CTS/PA7, RTS/PA15)	0
[15:0]	Reserved	R0	保留。	0

第 11 章 直接存储器访问控制（DMA）

直接存储器访问控制器（DMA）提供在外设和存储器之间或存储器和存储器之间的高速数据传输方式，无须 CPU 干预，数据可以通过 DMA 快速地移动，以节省 CPU 的资源来做其他操作。

DMA 控制器每个通道专门用来管理来自于一个或多个外设对存储器访问的请求。还有一个仲裁器来协调各通道之间的优先级。

11.1 主要特性

- 多个独立可配置通道
- 每个通道都直接连接专用的硬件 DMA 请求，并支持软件触发
- 支持循环的缓冲器管理
- 多个通道之间的请求优先权可以通过软件编程设置（最高、高、中和低），优先权设置相等时由通道号决定（通道号越低优先级越高）
- 支持外设到存储器、存储器到外设、存储器到存储器之间的传输
- 闪存、SRAM、外设的 SRAM、PB1、PB2 和 HB 外设均可作为访问的源和目标
- 可编程的数据传输数目：最大为 65535

11.2 功能描述

11.2.1 DMA 通道处理

1) 仲裁优先级

多个独立的通道产生的 DMA 请求通过逻辑或结构输入到 DMA 控制器，当前只会有一个通道的请求得到响应。模块内部的仲裁器根据通道请求的优先级来选择要启动的外设/存储器的访问。

软件管理中，应用程序通过对 DMA_CFGRx 寄存器的 PL[1:0] 位设置，可以为每个通道独立配置优先等级，包括最高、高、中、低 4 个等级。当通道间的软件设置等级一致时，模块会按固定的硬件优先级选择，通道编号偏低的要比偏高的有较高优先权。

2) DMA 配置

当 DMA 控制器收到一个请求信号时，会访问发出请求的外设或存储器，建立外设或存储器和存储器之间的数据传输。主要包括下面 3 个操作步骤：

- 1) 从外设数据寄存器或当前外设/存储器地址寄存器指示的存储器地址取数据，第一次传输时的开始地址是 DMA_PADDRx 或 DMA_MADDRx 寄存器指定的外设基地址或存储器地址。
- 2) 存数据到外设数据寄存器或当前外设/存储器地址寄存器指示的存储器地址，第一次传输时的开始地址是 DMA_PADDRx 或 DMA_MADDRx 寄存器指定的外设基地址或存储器地址。
- 3) 执行一次 DMA_CNTRx 寄存器中数值的递减操作，该寄存器指示当前未完成转移的操作数目。

每个通道包括 3 种 DMA 数据转移方式：

- 外设到存储器（MEM2MEM=0，DIR=0）
- 存储器到外设（MEM2MEM=0，DIR=1）
- 存储器到存储器（MEM2MEM=1）

注：存储器到存储器方式无需外设请求信号，配置为此模式后（MEM2MEM=1），通道开启（EN=1）即可启动数据传输。此方式不支持循环模式。

配置过程如下：

- 1) 在 DMA_PADDRx 寄存器中设置外设寄存器的首地址或存储器到存储器方式（MEM2MEM=1）下存储

器数据地址。发生 DMA 请求时，这个地址将是数据传输的源或目标地址。

- 2) 在 DMA_MADDRx 寄存器中设置存储器数据地址。发生 DMA 请求时，传输的数据将从这个地址读出或写入这个地址。
- 3) 在 DMA_CNTRx 寄存器中设置要传输的数据数量。在每个数据传输后，这个数值递减。
- 4) 在 DMA_CFGRx 寄存器的 PL[1:0]位中设置通道的优先级。
- 5) 在 DMA_CFGRx 寄存器中设置数据传输的方向、循环模式、外设和存储器的增量模式、外设和存储器的数据宽度、传输过半、传输完成、传输错误中断使能位，
- 6) 设置 DMA_CCRx 寄存器的 ENABLE 位，启动通道 x。

注：DMA_PADDRx/DMA_MADDRx/DMA_CNTRx 寄存器以及 DMA_CFGRx 寄存器中的数据传输的方向（DIR）、循环模式（位置）、外设和存储器的增量模式（MINC/PINC）等控制位只有在 DMA 通道被关闭下才可以配置写入。

3) 循环模式

设置 DMA_CFGRx 寄存器的 CIRC 位置 1，可以启用通道数据传输的循环模式功能。循环模式下，当数据传输的数目变为 0 时，DMA_CNTRx 寄存器的内容会自动被重新加载为其初始数值，内部的外设和存储器地址寄存器也被重新加载为 DMA_PADDRx 和 DMA_MADDRx 寄存器设定的初始地址值，DMA 操作将继续进行，直到通道被关闭或关闭 DMA 模式。

4) DMA 处理状态

- 传输过半：对应 DMA_INTFR 寄存器中的 HTIFx 位硬件置位。当 DMA 的传输数目减至初始设定值一半以下将会产生 DMA 传输过半标志，如果在 DMA_CCRx 寄存器中置位了 HTIE，则将产生中断。硬件通过此标志提醒应用程序，可以为新一轮数据传输做准备。
- 传输完成：对应 DMA_INTFR 寄存器中的 TCIFx 位硬件置位。当 DMA 的传输数目减至 0 将会产生 DMA 传输完成标志，如果在 DMA_CCRx 寄存器中置位了 TCIE，则将产生中断。
- 传输错误：对应 DMA_INTFR 寄存器中的 TEIFx 位硬件置位。读写一个保留的地址区域，将会产生 DMA 传输错误。同时模块硬件会自动清 0 发生错误的通道所对应的 DMA_CCRx 寄存器的 EN 位，该通道被关闭。如果在 DMA_CCRx 寄存器中置位了 TEIE，则将产生中断。

应用程序在查询 DMA 通道状态时，可以先访问 DMA_INTFR 寄存器的 GIFx 位，判断出当前哪个通道发生了 DMA 事件，进而处理该通道的具体 DAM 事件内容。

11.2.2 可编程的数据传输总大小/数据位宽/对齐方式

DMA 每个通道一轮传输的数据量总大小可编程，最大 65535 次。DMA_CNTRx 寄存器中指示待传输数目。在 EN=0 时，写入设置值，在 EN=1 开启 DMA 传输通道后，此寄存器变为只读属性，在每次传输后数值递减。

外设和存储器的传输数据取值支持地址指针自动递增功能，指针增量可编程。它们访问的第一个传输的数据地址存放在 DMA_PADDRx 和 DMA_MADDRx 寄存器中，通过设置 DMA_CFGRx 寄存器的 PINC 位或 MINC 位置 1，可以分别开启外设地址自增模式或存储器地址自增模式，PSIZE[1:0]设置外设地址取数据大小及地址自增大小，MSIZE[1:0]设置存储器地址取数据大小及地址自增大小，包括 3 种选择：8 位、16 位、32 位。具体数据转移方式如下表：

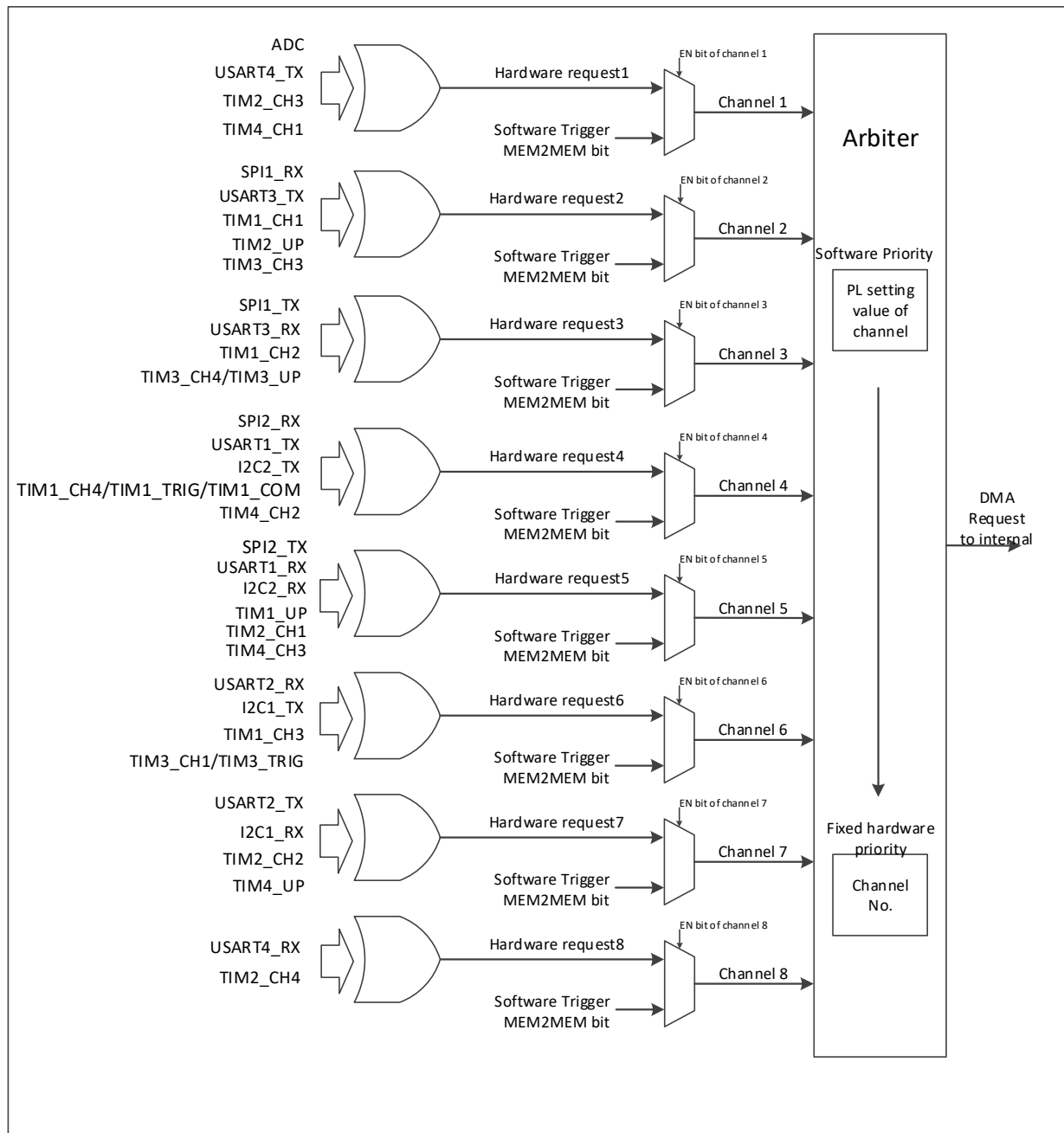
表 11-1 不同数据位宽下 DMA 转移（PINC=MINC=1）

源端位宽	目标位宽	传输数目	源：地址/数据	目标：地址/数据	传输操作
8	8	4	0x00/B0	0x00/B0	<ul style="list-style-type: none"> ● 源端地址递增量与源端设置的数据位宽对齐，取值大小等于源端数据位宽 ● 目标地址递增量与目标设置数
			0x01/B1	0x01/B1	
			0x02/B2	0x02/B2	
			0x03/B3	0x03/B3	

8	16	4	0x00/B0 0x01/B1 0x02/B2 0x03/B3	0x00/00B0 0x02/00B1 0x04/00B2 0x06/00B3	<p>据的位宽对齐，取值大小等于目标数据位宽</p> <ul style="list-style-type: none"> ● DMA 转移送入目标端的数据依据原则：数据大小不足高位补0，数据大小溢出高位去掉 ● 存储数据方式：小端模式，低地址存放低字节，高地址存放高字节
8	32	4	0x00/B0 0x01/B1 0x02/B2 0x03/B3	0x00/000000B0 0x04/000000B1 0x08/000000B2 0x0C/000000B3	
16	8	4	0x00/B1B0 0x02/B3B2 0x04/B5B4 0x06/B7B6	0x00/B0 0x01/B2 0x02/B4 0x03/B6	
16	16	4	0x00/B1B0 0x02/B3B2 0x04/B5B4 0x06/B7B6	0x00/B1B0 0x02/B3B2 0x04/B5B4 0x06/B7B6	
16	32	4	0x00/B1B0 0x02/B3B2 0x04/B5B4 0x06/B7B6	0x00/0000B1B0 0x04/0000B3B2 0x08/0000B5B4 0x0C/0000B7B6	
32	8	4	0x00/B3B2B1B0 0x04/B7B6B5B4 0x08/BBB9B8 0x0C/BFBEBDBC	0x00/B0 0x01/B4 0x02/B8 0x03/BC	
32	16	4	0x00/B3B2B1B0 0x04/B7B6B5B4 0x08/BBB9B8 0x0C/BFBEBDBC	0x00/B1B0 0x02/B5B4 0x04/B9B8 0x06/BDBC	
32	32	4	0x00/B3B2B1B0 0x04/B7B6B5B4 0x08/BBB9B8 0x0C/BFBEBDBC	0x00/B3B2B1B0 0x04/B7B6B5B4 0x08/BBB9B8 0x0C/BFBEBDBC	

11.2.3 DMA 请求映射

图 11-1 DMA 请求映像



DMA 控制器提供 8 个通道，每个通道对应多个外设请求，通过设置相应外设寄存器中对应 DMA 控制位，可以独立的开启或关闭各个外设的 DMA 功能，具体对应关系如下。

外设	通道 1	通道 2	通道 3	通道 4	通道 5	通道 6	通道 7	通道 8
ADC	ADC							
SPI1		SPI1_RX	SPI1_TX					
SPI2				SPI2_RX	SPI2_TX			
USART1				USART1_TX	USART1_RX			
USART2						USART2_RX	USART2_TX	
USART3		USART3_TX	USART3_RX					
USART4	USART4_TX							USART4_RX
I2C1						I2C1_TX	I2C1_RX	

I2C2				I2C2_TX	I2C2_RX			
TIM1		TIM1_CH1	TIM1_CH2	TIM1_CH4 TIM1_TRIG TIM1_COM	TIM1_UP	TIM1_CH3		
TIM2	TIM2_CH3	TIM2_UP			TIM2_CH1		TIM2_CH2	TIM2_CH4
TIM3		TIM3_CH3	TIM3_CH4 TIM3_UP			TIM3_CH1 TIM3_TRIG		
TIM4	TIM4_CH1			TIM4_CH2	TIM4_CH3		TIM4_UP	

11.3 寄存器描述

表 11-2 DMA 相关寄存器列表

名称	访问地址	描述	复位值
R32_DMA_INTFR	0x40020000	DMA 中断状态寄存器	0x00000000
R32_DMA_INTFCR	0x40020004	DMA 中断标志清除寄存器	0x00000000
R32_DMA_CFGR1	0x40020008	DMA 通道 1 配置寄存器	0x00000000
R32_DMA_CNTR1	0x4002000C	DMA 通道 1 传输数据数目寄存器	0x00000000
R32_DMA_PADDR1	0x40020010	DMA 通道 1 外设地址寄存器	0x00000000
R32_DMA_MADDR1	0x40020014	DMA 通道 1 存储器地址寄存器	0x00000000
R32_DMA_CFGR2	0x4002001C	DMA 通道 2 配置寄存器	0x00000000
R32_DMA_CNTR2	0x40020020	DMA 通道 2 传输数据数目寄存器	0x00000000
R32_DMA_PADDR2	0x40020024	DMA 通道 2 外设地址寄存器	0x00000000
R32_DMA_MADDR2	0x40020028	DMA 通道 2 存储器地址寄存器	0x00000000
R32_DMA_CFGR3	0x40020030	DMA 通道 3 配置寄存器	0x00000000
R32_DMA_CNTR3	0x40020034	DMA 通道 3 传输数据数目寄存器	0x00000000
R32_DMA_PADDR3	0x40020038	DMA 通道 3 外设地址寄存器	0x00000000
R32_DMA_MADDR3	0x4002003C	DMA 通道 3 存储器地址寄存器	0x00000000
R32_DMA_CFGR4	0x40020044	DMA 通道 4 配置寄存器	0x00000000
R32_DMA_CNTR4	0x40020048	DMA 通道 4 传输数据数目寄存器	0x00000000
R32_DMA_PADDR4	0x4002004C	DMA 通道 4 外设地址寄存器	0x00000000
R32_DMA_MADDR4	0x40020050	DMA 通道 4 存储器地址寄存器	0x00000000
R32_DMA_CFGR5	0x40020058	DMA 通道 5 配置寄存器	0x00000000
R32_DMA_CNTR5	0x4002005C	DMA 通道 5 传输数据数目寄存器	0x00000000
R32_DMA_PADDR5	0x40020060	DMA 通道 5 外设地址寄存器	0x00000000
R32_DMA_MADDR5	0x40020064	DMA 通道 5 存储器地址寄存器	0x00000000
R32_DMA_CFGR6	0x4002006C	DMA 通道 6 配置寄存器	0x00000000
R32_DMA_CNTR6	0x40020070	DMA 通道 6 传输数据数目寄存器	0x00000000
R32_DMA_PADDR6	0x40020074	DMA 通道 6 外设地址寄存器	0x00000000
R32_DMA_MADDR6	0x40020078	DMA 通道 6 存储器地址寄存器	0x00000000
R32_DMA_CFGR7	0x40020080	DMA 通道 7 配置寄存器	0x00000000
R32_DMA_CNTR7	0x40020084	DMA 通道 7 传输数据数目寄存器	0x00000000
R32_DMA_PADDR7	0x40020088	DMA 通道 7 外设地址寄存器	0x00000000
R32_DMA_MADDR7	0x4002008C	DMA 通道 7 存储器地址寄存器	0x00000000
R32_DMA_CFGR8	0x40020094	DMA 通道 8 配置寄存器	0x00000000
R32_DMA_CNTR8	0x40020098	DMA 通道 8 传输数据数目寄存器	0x00000000

R32_DMA_PADDR8	0x4002009C	DMA 通道 8 外设地址寄存器	0x00000000
R32_DMA_MADDR8	0x400200A0	DMA 通道 8 存储器地址寄存器	0x00000000

11.3.1 DMA 中断状态寄存器 (DMA_INTFR)

偏移地址: 0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TEIF8	HTIF8	TCIF8	GIF8	TEIF7	HTIF7	TCIF7	GIF7	TEIF6	HTIF6	TCIF6	GIF6	TEIF5	HTIF5	TCIF5	GIF5
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TEIF4	HTIF4	TCIF4	GIF4	TEIF3	HTIF3	TCIF3	GIF3	TEIF2	HTIF2	TCIF2	GIF2	TEIF1	HTIF1	TCIF1	GIF1

位	名称	访问	描述	复位值
31/27/23/ 19/15/11/ 7/3	TEIFx	RO	通道 x 的传输错误标志 (x=1/2/3/4/5/6/7/8) : 1: 在通道 x 上发生了传输错误; 0: 在通道 x 上没有传输错误。 硬件置位, 软件写 CTEIFx 位清除此标志。	0
30/26/22/ 18/14/10/ 6/2	HTIFx	RO	通道 x 的传输过半标志 (x=1/2/3/4/5/6/7/8) : 1: 在通道 x 上产生了传输过半事件; 0: 在通道 x 上没有传输过半。 硬件置位, 软件写 CHTIFx 位清除此标志。	0
29/25/21/ 17/13/9/5/ /1	TCIFx	RO	通道 x 的传输完成标志 (x=1/2/3/4/5/6/7/8) : 1: 在通道 x 上产生了传输完成事件; 0: 在通道 x 上没有传输完成事件。 硬件置位, 软件写 CTCIFx 位清除此标志。	0
28/24/20/ 16/12/8/4/ /0	GIFx	RO	通道 x 的全局中断标志 (x=1/2/3/4/5/6/7/8) : 1: 在通道 x 上产生了 TEIFx 或 HTIFx 或 TCIFx; 0: 在通道 x 上没有发生 TEIFx 或 HTIFx 或 TCIFx。 硬件置位, 软件写 CGIFx 位清除此标志。	0

11.3.2 DMA 中断标志清除寄存器 (DMA_INTFCR)

偏移地址: 0x04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CTEIF8	CHTIF8	CTCIF8	CGIF8	CTEIF7	CHTIF7	CTCIF7	CGIF7	CTEIF6	CHTIF6	CTCIF6	CGIF6	CTEIF5	CHTIF5	CTCIF5	CGIF5
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CTEIF4	CHTIF4	CTCIF4	CGIF4	CTEIF3	CHTIF3	CTCIF3	CGIF3	CTEIF2	CHTIF2	CTCIF2	CGIF2	CTEIF1	CHTIF1	CTCIF1	CGIF1

位	名称	访问	描述	复位值
31/27/23/ 19/15/11/ 7/3	CTEIFx	WO	清除通道 x 的传输错误标志 (x=1/2/3/4/5/6/8) : 1: 清除 DMA_INTFR 寄存器中的 TEIFx 标志; 0: 无作用。	0
30/26/22/ 18/14/10/ 6/2	CHTIFx	WO	清除通道 x 的传输过半标志 (x=1/2/3/4/5/6/8) : 1: 清除 DMA_INTFR 寄存器中的 HTIFx 标志; 0: 无作用。	0
29/25/21/ /1	CTCIFx	WO	清除通道 x 的传输完成标志 (x=1/2/3/4/5/6/7/8) :	0

17/13/9/5 /1			1: 清除 DMA_INTFR 寄存器中的 TCIFx 标志; 0: 无作用。	
28/24/20/ 16/12/8/4 /0	CGIFx	WO	清除通道 x 的全局中断标志 (x=1/2/3/4/5/6/7/8) : 1: 清除 DMA_INTFR 寄存器中的 TEIFx/HTIFx/TCIFx/ GIFx 标志; 0: 无作用。	0

11.3.3 DMA 通道 x 配置寄存器 (DMA_CFGRx) (x=1/2/3/4/5/6/7/8)

偏移地址: $0x08 + (x-1)*20$

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	MEM2MEM	PL[1:0]	MSIZE[1:0]	PSIZE[1:0]	MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN			

位	名称	访问	描述	复位值
[31:15]	Reserved	RO	保留。	0
14	MEM2MEM	RW	存储器到存储器模式使能: 1: 使能存储器到存储器数据传输模式; 0: 非存储器到存储器数据传输。	0
[13:12]	PL[1:0]	RW	通道优先级设置: 00: 低; 01: 中; 10: 高; 11: 最高。	00b
[11:10]	MSIZE[1:0]	RW	存储器地址数据宽度设置: 00: 8 位; 01: 16 位; 10: 32 位; 11: 保留。	00b
[9:8]	PSIZE[1:0]	RW	外设地址数据宽度设置: 00: 8 位; 01: 16 位; 10: 32 位; 11: 保留。	00b
7	MINC	RW	存储器地址增量递增模式使能: 1: 使能存储器地址增量递增操作; 0: 存储器地址保持不变操作。	0
6	PINC	RW	外设地址增量递增模式使能: 1: 使能外设地址增量递增操作; 0: 外设地址保持不变操作。	0
5	CIRC	RW	DMA 通道循环模式使能: 1: 使能循环操作; 0: 执行单次操作。	0
4	DIR	RW	数据传输方向: 1: 从存储器读; 0: 从外设读。	0
3	TEIE	RW	传输错误中断使能控制: 1: 使能传输错误中断; 0: 禁止传输错误中断。	0

2	HTIE	RW	传输过半中断使能控制： 1：使能传输过半中断； 0：禁止传输过半中断。	0
1	TCIE	RW	传输完成中断使能控制： 1：使能传输完成中断； 0：禁止传输完成中断。	0
0	EN	RW	通道使能控制： 1：通道开启； 0：通道关闭。 发生 DMA 传输错误时，硬件自动将此位清 0，关闭通道。	0

11.3.4 DMA 通道 x 传输数据数目寄存器 (DMA_CNTRx) (x=1/2/3/4/5/6/7/8)

偏移地址：0x0C + (x-1)*20

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

位	名称	访问	描述	复位值
[31:16]	Reserved	RO	保留。	0
[15:0]	NDT[15:0]	RW	数据传输数量，范围 0~65535。 指示剩余的待传输数目（寄存器内容在每次 DMA 传输后递减）。 在通道为循环模式下，寄存器的内容将被自动重新加载为之前配置的数值。	0

注：表示当前待传输数目。当寄存器内容为 0 时，无论通道是否开启，都不会发生任何数据传输。

11.3.5 DMA 通道 x 外设地址寄存器 (DMA_PADDRx) (x=1/2/3/4/5/6/7/8)

偏移地址：0x10 + (x-1)*20

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
PA[31:0]																															

位	名称	访问	描述	复位值
[31:0]	PA[31:0]	RW	外设基地址，作为外设数据传输的源或目标地址。 当 PSIZE[1:0]= '01'（16 位），模块自动忽略 bit0，操作地址自动 2 字节对齐；当 PSIZE[1:0]= '10'（32 位），模块自动忽略 bit[1:0]，操作地址自动 4 字节对齐。	0

注：此寄存器只能在 EN=0 时更改，EN=1 时不可写。

11.3.6 DMA 通道 x 存储器地址寄存器 (DMA_MADDRx) (x=1/2/3/4/5/6/7/8)

偏移地址：0x14 + (x-1)*20

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
MA[31:0]																															

位	名称	访问	描述	复位值
[31:0]	MA[31:0]	RW	存储器数据地址，作为数据传输的源或目标地址。 当 MSIZE[1:0]= ‘01’ （16 位），模块自动忽略 bit0， 操作地址自动 2 字节对齐；当 MSIZE[1:0]= ‘10’ （32 位），模块自动忽略 bit[1:0]，操作地址自动 4 字节 对齐。	0

注：此寄存器只能在 EN=0 时更改，EN=1 时不可写。

第 12 章 模拟/数字转换（ADC）

ADC 模块包含 1 个 12 位的逐次逼近型的模拟数字转换器，最高 48MHz 的输入时钟。支持 10 个外部通道和 3 个内部信号源采样源。可完成通道的单次转换、连续转换，通道间自动扫描模式、间断模式、外部触发模式等功能。可以通过模拟看门狗功能监测通道电压是否在阈值范围内。

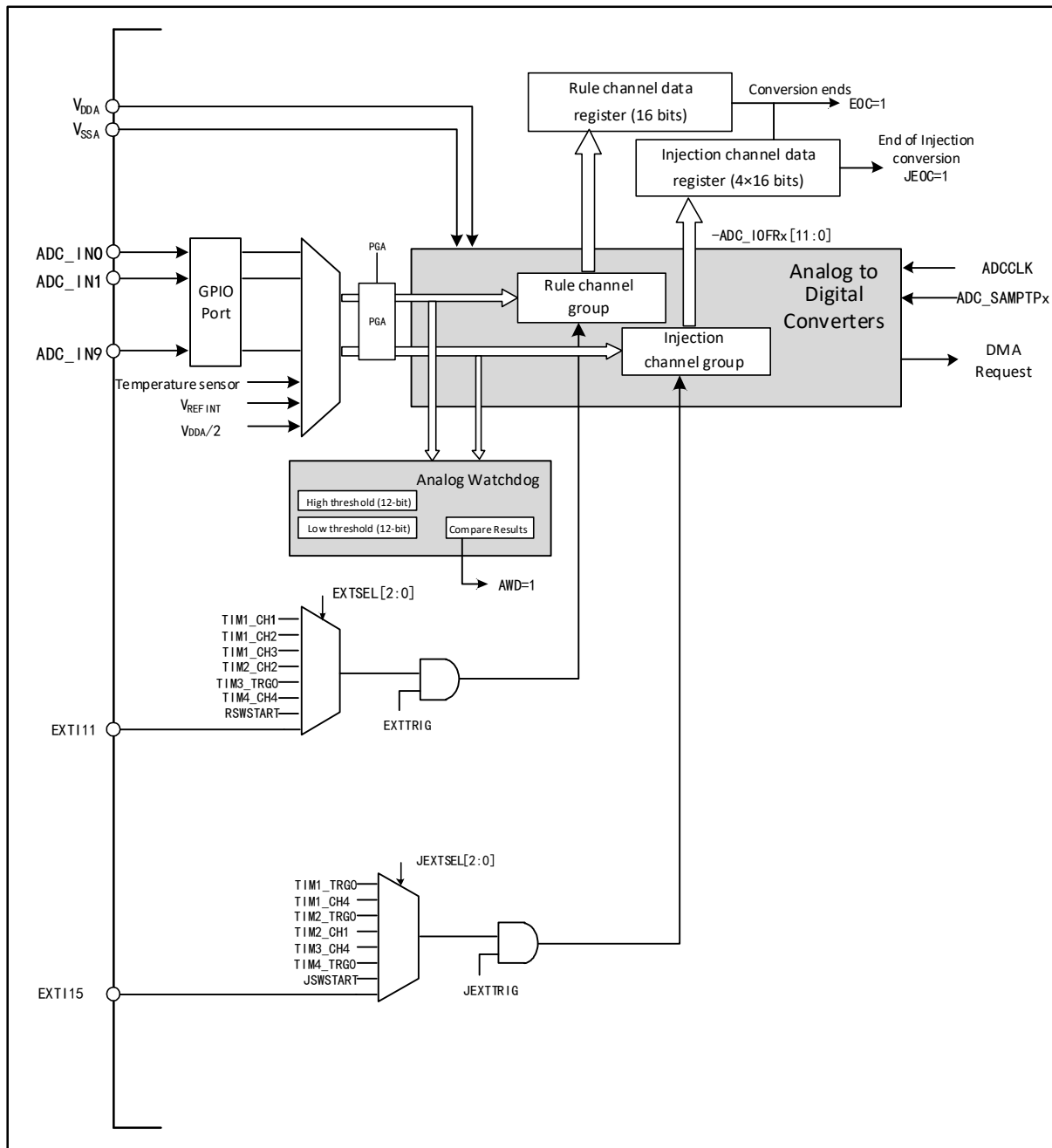
12.1 主要特性

- 12 位分辨率
- 支持 10 个外部通道和 3 个内部信号源采样
- 多通道的多种采样转换方式：单次、连续、扫描、触发、间断等
- 数据对齐模式：左对齐、右对齐
- 采样时间可按通道分别编程
- 规则转换和注入转换均支持外部触发
- 模拟看门狗监测通道电压，自校准功能
- ADC 通道输入范围： $0 \leq V_{IN} \leq V_{DDA}$
- 输入增益可调，可实现小信号放大采样

12.2 功能描述

12.2.1 模块结构

图 12-1 ADC 模块框图



12.2.2 ADC 配置

1) 模块上电

ADC_CTLR2 寄存器的 ADON 位为 1 表示 ADC 模块上电。当 ADC 模块从断电模式 (ADON=0) 下进入上电状态 (ADON=1) 后, 需要延迟一段时间 t_{STAB} 用于模块稳定时间。之后再次写入 ADON 位为 1, 用于作为软件启动 ADC 转换的启动信号。通过清除 ADON 位为 0, 可以终止当前转换并将 ADC 模块置于断电模式, 这个状态下, ADC 几乎不耗电。

2) 采样时钟

模块的寄存器操作基于 PCLK2 (PB2 总线) 时钟, 其转换单元的时钟基准 ADCCLK 与 PCLK2 同步,

由 RCC_CFGR0 寄存器的 ADCPRE[1:0]域配置分频，最大不能超过 48MHz。

3) 通道配置

ADC 模块提供了 13 个通道采样源，包括 10 个外部通道和 3 个内部通道。它们可以配置到两种转换组中：规则组和注入组。以实现任意多个通道上以任意顺序进行一系列转换构成的组转换。

转换组：

- 规则组：由多达 16 个转换组成。规则通道和它们的转换顺序在 ADC_RSQRx 寄存器中设置。规则组中转换的总数量应写入 ADC_RSQR1 寄存器的 L[3:0]中。
- 注入组：由多达 4 个转换组成。注入通道和它们的转换顺序在 ADC_ISQR 寄存器中设置。注入组里的转换总数量应写入 ADC_ISQR 寄存器的 JL[1:0]中。

注：如果 ADC_RSQRx 或 ADC_ISQR 寄存器在转换期间被更改，当前的转换被终止，一个新的启动信号将发送到 ADC 以转换新选择的组。

3 个内部通道：

- 温度传感器：连接 ADC_IN16 通道，用来测量芯片内部温度。
- V_{REFINT} 内部参考电压：连接 ADC_IN17 通道。
- $V_{DDA}/2$ ：连接 ADC_IN18 通道。

4) 校准

ADC 有一个内置自校准模式。经过校准环节可大幅减小因内部电容器组的变化而造成的精度误差。在校准期间，在每个电容器上都会计算出一个误差修正码，用于消除在随后的转换中每个电容器上产生的误差。

ADC_CFG 寄存器的 FIFO_EN 位置 1，通过写 ADC_CTLR2 寄存器的 RSTCAL 位置 1 初始化校准寄存器，等待 RSTCAL 硬件清 0 表示初始化完成。置位 CAL 位，启动校准功能，一旦校准结束，硬件会自动清除 CAL 位。转换内部通道 18 中间电压值，2048 数值与转换值进行偏移比较。之后可以开始正常的转换功能，每次 ADC 转后加上偏移值。建议在 ADC 模块上电时执行一次 ADC 校准。

注：启动校准前，必须保证 ADC 模块处于上电状态 (ADON=1) 超过至少两个 ADC 时钟周期。

5) 可编程采样时间

ADC 使用若干个 ADCCLK 周期对输入电压采样，通道的采样周期数目可以通过 ADC_SAMPTR1 和 ADC_SAMPTR2 寄存器中的 SMPx[2:0]位更改。每个通道可以分别使用不同的时间采样。

总转换时间如下计算：

$$T_{CONV} = \text{采样时间} + 12.5T_{ADCCLK}$$

而采样时间可以根据 SMPx[2:0]决定

SMPx[2:0]	采样时间 (ADC_LP=0)	采样时间 (ADC_LP=1)
000	1.5 个周期	7.5 个周期
001	7.5 个周期	11.5 个周期
010	13.5 个周期	17.5 个周期
011	28.5 个周期	27.5 个周期
100	41.5 个周期	47.5 个周期
101	55.5 个周期	
110	71.5 个周期	
111	239.5 个周期	

ADC 的规则通道转换支持 DMA 功能。规则通道转换的值储存在一个仅有的数据寄存器 ADC_RDATAR 中，为防止连续转换多个规则通道时，没有及时取走 ADC_RDATAR 寄存器中的数据，可以开启 ADC 的 DMA 功能。硬件会在规则通道的转换结束时 (EOC 置位) 产生 DMA 请求，并将转换的数据从

ADC_RDATAR 寄存器传输到用户指定的目的地址。

对 DMA 控制器模块的通道配置完成后，写 ADC_CTLR2 寄存器的 DMA 位置 1，开启 ADC 的 DMA 功能。

注：注入组转换不支持 DMA 功能。

6) 数据对齐

ADC_CTLR2 寄存器中的 ALIGN 位选择 ADC 转换后的数据存储对齐方式。12 位数据支持左对齐和右对齐模式。

规则组通道的数据寄存器 ADC_RDATAR 保存的是实际转换的 12 位数字值；而注入组通道的数据寄存器 ADC_IDATARx 是实际转换的数据减去 ADC_IOFRx 寄存器的定义的偏移量后写入的值，会存在正负情况，所以有符号位（SIGNB）。

图 12-2 数据左对齐

规则组数据寄存器															
D11	D10	D9	D8	D7	D6	D5	D4	D4	D2	D1	D0	0	0	0	0
注入组数据寄存器															
SIGNB	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	0	0	0

图 12-3 数据右对齐

规则组数据寄存器															
0	0	0	0	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
注入组数据寄存器															
SIGNB	SIGNB	SIGNB	SIGNB	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

7) ADC 配置使能

ADC 模块提供了一个 ADC_CFG 寄存器，此寄存器通过置位相应的控制位进行使能某个模块功能。

ADC_DUTY_EN 位为 ADC 时钟占空比控制位，当该位置 1 时将输入时钟延迟 4ns 得到约 75%占空比的 ADC 时钟，置 0 时输入时钟不做处理。

FIFO_EN 位在高电平下使能 ADC FIFO。

ADC_LP 位控制 ADC 低功耗模式，置 1 时 vcmbuffer 和比较器功耗较高，适用于 1M 及以上采样率，置 0 时进入低功耗模式，vcmbuffer 和比较器功耗较低，适用于 1M 及以下采样率。

ADC_BUFTRIM 的 [3:0] 位为 ADC BUFFER 失调校准控制位，其中最高位选择失调校准极性正负，[2:0] 控制校准档位。

12.2.3 外部触发源

ADC 转换的启动事件可以由外部事件触发。如果设置了 ADC_CTLR2 寄存器的 EXTTRIG 或 JEXTTRIG 位，则可分别通过外部事件触发规则组或注入组通道的转换。此时，EXTSEL [2:0] 和 JEXTSEL [2:0] 位的配置决定规则组和注入组的外部事件源。

表 12-1 规则组通道的外部触发源

EXTSEL [2:0]	触发源	类型
000	定时器 1 的 CC1 事件	来自片上定时器的内部信号
001	定时器 1 的 CC2 事件	
010	定时器 1 的 CC3 事件	
011	定时器 2 的 CC2 事件	
100	定时器 3 的 TRGO 事件	
101	定时器 4 的 CC4 事件	

110	EXTI 线 11	来自外部引脚
111	RSWSTART 位置 1 软件触发	软件控制位

表 12-2 注入组通道的外部触发源

JEXTSEL[2:0]	触发源	类型
000	定时器 1 的 TRG0 事件	来自片上定时器的内部信号
001	定时器 1 的 CC4 事件	
010	定时器 2 的 TRG0 事件	
011	定时器 2 的 CC1 事件	
100	定时器 3 的 CC4 事件	
101	定时器 4 的 TRG0 事件	
110	EXTI 线 15	来自外部引脚
111	JSWSTART 位置 1 软件触发	软件控制位

12.2.4 转换模式

表 12-3 转换模式组合

ADC_CTLR1 和 ADC_CTLR2 寄存器控制位					ADC 转换模式
CONT	SCAN	RDISCEN/JDISCEN	IAUTO	启动事件	
0	0	0	0	ADON 位置 1	单次单通道模式：某一规则通道执行单次转换。
				外部触发方式	单次单通道模式：规则通道或注入通道的某一通道执行单次转换。
	1	0	0	ADON 位置 1 或外部触发方式	单次扫描模式：按顺序对选中的所有规则组通道(ADC_RSQRx)或所有注入组通道(ADC_ISQR)逐个执行单次转换。 触发注入方式：当规则组通道转换过程中可以插入注入组通道所有转换，之后再继续规则组通道转换；但转换注入组通道时不会插入规则组通道转换。
				ADON 位置 1 或外部触发方式	单次扫描模式：按顺序对选中的所有规则组通道(ADC_RSQRx)或所有注入组通道(ADC_ISQR)逐个执行单次转换。 自动注入方式：在规则组通道转换完之后，注入组通道被自动转换。 注：转换过程中不允许出现注入通道的外部触发信号。
	0	1 (RDISCEN 和 JDISCEN 不能同时为 1)	0	外部触发方式	单次中断模式：每次启动事件，执行一个短序列(DISCNUM[2:0]定义数量)的通道数量转换，直到所有选中通道转换完成才能重头开始。 注：规则组和注入组选中此模式控制位分别为 JDISCEN 和 RDISCEN，不能同时为规则组和注入组配置中断模式，中断模式只能用于一组转换。
			1	—	禁止此模式。
	1	1	X	—	无此模式。

1	0	0	0	ADON 位置 1 或 外部 触 发方式	连续单通道/扫描模式：每轮结束后重复新一轮的转换，直到 CONT 清 0 才能终止。
	1	0	0		
			1		

注：规则组和注入组的外部触发事件是不一样的，而且‘ADON’位只能启动规则组通道转换，所以规则组和注入组通道转换的启动事件独立。

1) 单次单通道转换模式

此模式下，对当前 1 个通道只执行一次转换。该模式对规则组或注入组中排序第 1 的通道执行转换，其中通过设置 ADC_CTLR2 寄存器的 ADON 位置 1 (只适用于规则通道) 启动也可通过外部触发启动 (适用于规则通道或注入通道)。一旦选择通道的转换完成将：

如果转换的是规则组通道，则转换数据被储存在 16 位 ADC_RDATAR 寄存器中，EOC 标志被置位，如果设置了 EOCIE 位，将触发 ADC 中断。

如果转换的是注入组通道，则转换数据被储存在 16 位 ADC_IDATAR1 寄存器中，EOC 和 JE0C 标志被置位，如果设置了 JE0CIE 或 EOCIE 位，将触发 ADC 中断。

2) 单次扫描模式转换

通过设置 ADC_CTLR1 寄存器的 SCAN 位为 1 进入 ADC 扫描模式。此模式用来扫描一组模拟通道，对被 ADC_RSQRx 寄存器 (对规则通道) 或 ADC_ISQR (对注入通道) 选中的所有通道逐个执行单次转换，当前通道转换结束时，同一组的下一个通道被自动转换。

在扫描模式里，根据 IAUTO 位的状态，又分为触发注入方式和自动注入方式。

● 触发注入

IAUTO 位为 0，当在扫描规则组通道过程中，发生了注入组通道转换的触发事件，当前转换被复位，注入通道的序列被以单次扫描方式进行，在所有选中的注入组通道扫描转换结束后，恢复上次被中断的规则组通道转换。

如果当前在扫描注入组通道序列时，发生了规则通道的启动事件，注入组转换不会被中断，而是在注入序列转换完成后再执行规则序列的转换。

注：使用触发的注入转换时，必须保证触发事件的间隔长于注入序列。例如，完成注入序列的转换总体时间需要 28 个 ADCCLK，那么触发注入通道的事件间隔时间最小值为 29 个 ADCCLK。

● 自动注入

IAUTO 位为 1，在扫描完规则组选中的所有通道转换后，自动进行注入组选中通道的转换。这种方式可以用来转换 ADC_RSQRx 和 ADC_ISQR 寄存器中多达 20 个转换序列。

此模式里，必须禁止注入通道的外部触发 (JEXTTRIG=0)。

注：对于 ADC 时钟预分频系数 (ADCPRE[1:0]) 为 4 至 8 时，当从规则转换切换到注入序列或从注入转换切换到规则序列时，会自动插入 1 个 ADCCLK 间隔；当 ADC 时钟预分频系数为 2 时，则有 2 个 ADCCLK 间隔的延迟。

3) 单次中断模式转换

通过设置 ADC_CTLR1 寄存器的 RDISCEN 或 JDISCEN 位为 1 进入规则组或注入组的中断模式。此模式区别扫描模式中扫描完整的一组通道，而是将一组通道分为多个短序列，每次外部触发事件将执行一个短序列的扫描转换。

短序列的长度 n (n≤8) 定义在 ADC_CTLR1 寄存器的 DISCNUM[2:0] 中，当 RDISCEN 为 1，则是规则组的中断模式，待转换总长度定义在 ADC_RSQR1 寄存器的 L[3:0] 中；当 JDISCEN 为 1，则是注入组的中断模式，待转换总长度定义在 ADC_ISQR 寄存器的 JL[1:0] 中。不能同时将规则组和注入组设置为中断模式。

规则组中断模式举例：

RDISCEN=1, DISCNUM[2:0]=3, L[3:0]=8, 待转换通道=1, 3, 2, 5, 8, 4, 7, 6

第 1 次外部触发：转换序列为：1，3，2
第 2 次外部触发：转换序列为：5，8，4
第 3 次外部触发：转换序列为：7，6，同时产生 EOC 事件
第 4 次外部触发：转换序列为：1，3，2

注入组间断模式举例：

JDISCEN=1，DISCNUM[2:0]=1，JL[1:0]=3，待转换通道=1，3，2

第 1 次外部触发：转换序列为：1
第 2 次外部触发：转换序列为：3
第 3 次外部触发：转换序列为：2，同时产生 EOC 和 JE0C 事件
第 4 次外部触发：转换序列为：1

注：1. 当以间断模式转换一个规则组或注入组时，转换序列结束后不自动从头开始。当所有子组被转换完成，下一次触发事件启动第一个子组的转换。
2. 不能同时使用自动注入（IAUTO=1）和间断模式。
3. 不能同时为规则组和注入组设置间断模式，间断模式只能用于一组转换。

4) 连续转换

在连续转换模式中，当前面 ADC 转换一结束马上就启动另一次转换，转换不会在选择组的最后一个通道上停止，而是再次从选择组的第一个通道继续转换。此模式的启动事件包括外部触发事件和 ADON 位置 1，设置启动后，需将 CONT 位置 1。

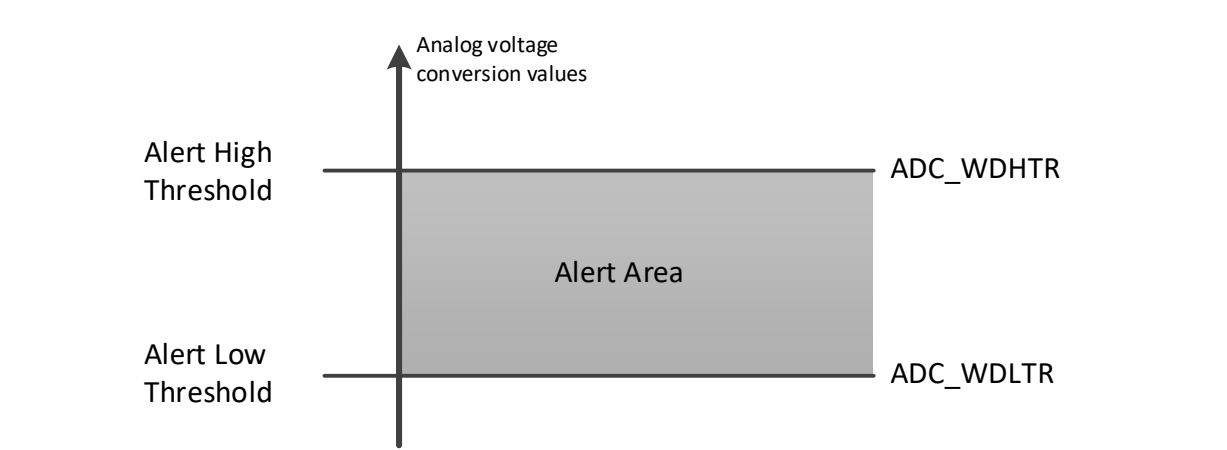
如果一个规则通道被转换，转换数据被存储于 ADC_RDATAR 寄存器中，转换结束标志 EOC 被置位，如果设置了 EOCIE，则产生中断。

如果一个注入通道被转换，转换数据被存储于 ADC_IDATARx 寄存器中，注入转换结束标志 JE0C 被置位，如果设置了 JE0CIE，则产生中断。

12.2.5 模拟看门狗

如果被 ADC 转换的模拟电压低于低阈值或高于高阈值，AWD 模拟看门狗状态位被设置。阈值设置位于 ADC_WDHTR 和 ADC_WDLTR 寄存器的最低 12 个有效位中。通过设置 ADC_CTLR1 寄存器的 AWDIE 位以允许产生相应中断。通过设置 ADC_CFG 寄存器的 AWRST_EN 位可开启模拟看门狗复位。

图 12-4 模拟看门狗阈值区



配置 ADC_CTLR1 寄存器的 AWDSGL、AWDEN、JAWDEN 及 AWDCH[4:0] 位选择模拟看门狗警戒的通道，具体关系见下表：

表 12-4 模拟看门狗通道选择

模拟看门狗警戒通道	ADC_CTLR1 寄存器控制位
-----------	------------------

	AWDSGL	AWDEN	JAWDEN	AWDCH[4:0]
不警戒	忽略	0	0	忽略
所有注入通道	0	0	1	忽略
所有规则通道	0	1	0	忽略
所有注入和规则通道	0	1	1	忽略
单一注入通道	1	0	1	决定通道编号
单一规则通道	1	1	0	决定通道编号
单一注入和规则通道	1	1	1	决定通道编号

12.2.6 温度传感器

芯片内置温度传感器，连接 ADC_INT16 通道，通过 ADC 将传感器输出的电压转换成数字值来反馈芯片内部温度，推荐设置采样时间是 $17.1\mu s$ 。温度传感器输出的电压随温度线性变化，由于制造离散性，其线性变化的曲线斜率和偏移有所不同，所以内部温度传感器更适合于检测温度的变化，而不是测量绝对的温度。如果需要测量精确的温度，应该使用一个外置的温度传感器。

通过设置 ADC_CTLR2 寄存器的 TSVREFE 位置 1，唤醒 ADC 内部采样通道，软件启动或外部触发启动 ADC 的温度传感器通道转换，读取数据结果 (mV)。其中，数字值和温度 (°C) 换算公式如下：

$$\text{温度} (^{\circ}\text{C}) = ((V_{\text{SENSE}} - V_{25}) / \text{Avg_Slope}) + 25$$

V25：温度传感器在 25°C 下的电压值

Avg_Slope：温度与 V_{SENSE} 曲线的平均斜率 (mV/°C)

参考数据手册电气特性章节中 V_{25} 和 Avg_Slope 的实际值。

注：内部温度传感器上电 (TSVREFE 位从 0 改为 1) 需要一个建立时间，而 ADC 模块上电也需要一个建立时间 (ADON 位从 0 改为 1)，所以为了缩短等待时间，可以同时设置 ADON 和 TSVREFE 位。

12.3 寄存器描述

表 12-5 ADC 相关寄存器列表

名称	访问地址	描述	复位值
R32_ADC_STATR	0x40012400	ADC 状态寄存器	0x00000000
R32_ADC_CTLR1	0x40012404	ADC 控制寄存器 1	0x00000000
R32_ADC_CTLR2	0x40012408	ADC 控制寄存器 2	0x00000000
R32_ADC_SAMPTR1	0x4001240C	ADC 采样时间配置寄存器 1	0x00000000
R32_ADC_SAMPTR2	0x40012410	ADC 采样时间配置寄存器 2	0x00000000
R32_ADC_I0FR1	0x40012414	ADC 注入通道数据偏移寄存器 1	0x00000000
R32_ADC_I0FR2	0x40012418	ADC 注入通道数据偏移寄存器 2	0x00000000
R32_ADC_I0FR3	0x4001241C	ADC 注入通道数据偏移寄存器 3	0x00000000
R32_ADC_I0FR4	0x40012420	ADC 注入通道数据偏移寄存器 4	0x00000000
R32_ADC_WDHTR	0x40012424	ADC 看门狗高阈值寄存器	0x00000FFF
R32_ADC_WDLTR	0x40012428	ADC 看门狗低阈值寄存器	0x00000000
R32_ADC_RSQR1	0x4001242C	ADC 规则通道序列寄存器 1	0x00000000
R32_ADC_RSQR2	0x40012430	ADC 规则通道序列寄存器 2	0x00000000
R32_ADC_RSQR3	0x40012434	ADC 规则通道序列寄存器 3	0x00000000
R32_ADC_ISQR	0x40012438	ADC 注入通道序列寄存器	0x00000000
R32_ADC_IDATAR1	0x4001243C	ADC 注入数据寄存器 1	0x00000000
R32_ADC_IDATAR2	0x40012440	ADC 注入数据寄存器 2	0x00000000
R32_ADC_IDATAR3	0x40012444	ADC 注入数据寄存器 3	0x00000000

R32_ADC_IDATAR4	0x40012448	ADC 注入数据寄存器 4	0x00000000
R32_ADC_RDATAR	0x4001244C	ADC 规则数据寄存器	0x00000000
R32_ADC_CFG	0x40012450	ADC 配置寄存器	0x00000000

12.3.1 ADC 状态寄存器 (ADC_STATR)

偏移地址: 0x00

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

位	名称	访问	描述	复位值
[31:5]	Reserved	R0	保留。	0
4	STRT	RWO	规则通道转换开始状态: 1: 规则通道转换已开始; 0: 规则通道转换未开始。 该位由硬件置 1, 由软件清 0 (写 1 无效)。	0
3	JSTRT	RWO	注入通道转换开始状态: 1: 注入通道转换已开始; 0: 注入通道转换未开始。 该位由硬件置 1, 由软件清 0 (写 1 无效)。	0
2	JEOC	RWO	注入通道组转换结束状态: 1: 转换完成; 0: 转换未完成。 该位由硬件置 1 (所有注入通道转换完), 由软件清 0 (写 1 无效)。	0
1	EOC	RWO	转换结束状态: 1: 转换完成; 0: 转换未完成。 该位由硬件置 1 (规则或注入通道组转换结束), 由软件清 0 (写 1 无效) 或读 ADC_RDATAR 时清除。	0
0	AWD	RWO	模拟看门狗标志位: 1: 发生模拟看门狗事件; 0: 没有发生模拟看门狗事件。 该位由硬件置 1 (转换值超出 ADC_WDHTR 和 ADC_WDLTR 寄存器范围), 由软件清 0 (写 1 无效)。	0

12.3.2 ADC 控制寄存器 1 (ADC_CTLR1)

偏移地址: 0x04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved			PGA[1:0]		BUFE N	TKIT UNE	TKENABLE	RAWDEN	JAWDEN	Reserved					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

DISCNUM[2:0]	JDISC EN	RDISC CEN	JAUTO	AWD SGL	SCAN	JEOCI E	AWDIE	EOC IE	AWDCH[4:0]
--------------	-------------	--------------	-------	------------	------	------------	-------	-----------	------------

位	名称	访问	描述	复位值
[31:29]	Reserved	RO	保留。	0
[28:27]	PGA[1:0]	RW	ADC通道增益配置 00: x1 01: x4 10: x16 11: x64 注: 输入增益可调, 可实现小信号放大采样。使用此功能需开启ADC_Buffer。	00b
26	BUFEN	RW	ADC BUFFER使能 0: 关闭输入Buffer 1: 使能输入Buffer	0
25	TKITUNE	RW	TKEY模块充电电流配置 0: 充电电流为35 μ A 1: 充电电流减半	0
24	TKENABLE	RW	TKEY模块使能控制, 包括TKEY_F和TKEY_V单元: 1: 开启TKEY模块; 0: 关闭 TKEY 模块。	0
23	AWDEN	RW	在规则通道上模拟看门狗功能使能位: 1: 规则通道上使能模拟看门狗; 0: 规则通道上关闭模拟看门狗。	0
22	JAWDEN	RW	在注入通道上模拟看门狗功能使能位: 1: 注入通道上使能模拟看门狗; 0: 注入通道上关闭模拟看门狗。	0
[21:16]	Reserved	RO	保留	0000b
[15:13]	DISCNUM[2:0]	RW	中断模式下, 外部触发后要转换的规则通道数目: 000: 1 个通道; ... 111: 8 个通道。	000b
12	JDISCEN	RW	注入通道上的中断模式使能位: 1: 使能注入通道上的中断模式; 0: 关闭注入通道上的中断模式。	0
11	RDISCEN	RW	规则通道上的中断模式使能位: 1: 使能规则通道上的中断模式; 0: 关闭规则通道上的中断模式。	0
10	JAUTO	RW	开启规则通道完成后, 自动转换注入通道组使能位: 1: 使能自动的注入通道组转换; 0: 关闭自动的注入通道组转换。 注: 此模式需要禁止注入通道的外部触发功能。	0
9	AWDSGL	RW	扫描模式下, 在单一通道上使用模拟看门狗使能位: 1: 在单一通道上使用模拟看门狗 (AWDCH[4:0] 选择); 0: 在所有通道上使用模拟看门狗。	0

8	SCAN	RW	扫描模式使能位： 1：使能扫描模式（连续转换 ADC_I0FRx 和 ADC_RSQRx 选择的所有通道）； 0：关闭扫描模式。	0
7	JE0CIE	RW	注入通道组转换结束中断使能位： 1：使能注入通道组转换完成中断（JE0C 标志）； 0：关闭注入通道组转换完成中断。	0
6	AWDIE	RW	模拟看门狗中断使能位： 1：使能模拟看门狗中断； 0：关闭模拟看门狗中断。 <i>注：在扫描模式下，如果发生此中断将中止扫描。</i>	0
5	EOCIE	RW	转换结束（规则或注入通道组）中断使能位： 1：使能转换结束中断（EOC 标志）； 0：关闭转换结束中断。	0
[4:0]	AWDCH[4:0]	RW	模拟看门狗通道选择位： 00000：模拟输入通道 0； 00001：模拟输入通道 1； ... 01001：模拟输入通道 9； 01010：保留； 01011：保留； 01100：保留； 01101：保留； 01110：保留； 01111：保留； 10000：模拟输入通道 16； 10001：模拟输入通道 17； 10010：模拟输入通道 18； 其他：保留。	00000b

12.3.3 ADC 控制寄存器 2 (ADC_CTLR2)

偏移地址：0x08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved								TS VREFE	SW START	JSW START	EXT TRIG	EXTSEL[2:0]			Reser ved
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JEXT TRIG	JEXTSEL[2:0]			ALIGN	Reserved	DMA	Reserved					RST CAL	CAL	CONT	ADON

位	名称	访问	描述	复位值
[31:24]	Reserved	RO	保留。	0
23	TSVREFE	RW	温度传感器和内部电压 (V_{REFINT}) 通道使能位： 1：使能温度传感器和 V_{REFINT} 通道； 0：禁止温度传感器和 V_{REFINT} 通道。	0

22	RSWSTART	RW	启动一个规则通道转换，需要设置软件触发： 1：启动规则通道转换； 0：复位状态。 此位由软件置位，转换开始后硬件清 0。	0
21	JSWSTART	RW	启动一个注入通道转换，需要设置软件触发： 1：启动注入通道转换； 0：复位状态。 此位由软件置位，转换开始后硬件清 0 或者软件清 0。	0
20	EXTTRIG	RW	规则通道的外部触发转换模式使能： 1：使用外部事件启动转换； 0：关闭外部事件启动功能。	0
[19:17]	EXTSEL[2:0]	RW	启动规则通道转换的外部触发事件选择： 000：定时器 1 的 CC1 事件； 001：定时器 1 的 CC2 事件； 010：定时器 1 的 CC3 事件； 011：定时器 2 的 CC2 事件； 100：定时器 3 的 TRGO 事件； 101：定时器 4 的 CC4 事件； 110：EXTI 线 11； 111：RSWSTART 软件触发。	000b
16	Reserved	RO	保留。	0
15	JEXTTRIG	RW	注入通道的外部触发转换模式使能： 1：使用外部事件启动转换； 0：关闭外部事件启动功能。	0
[14:12]	JEXTSEL[2:0]	RW	启动注入通道转换的外部触发事件选择： 000：定时器 1 的 TRGO 事件； 001：定时器 1 的 CC4 事件； 010：定时器 2 的 TRGO 事件； 011：定时器 2 的 CC1 事件； 100：定时器 3 的 CC4 事件； 101：定时器 4 的 TRGO 事件； 110：EXTI 线 15； 111：JSWSTART 软件触发。	000b
11	ALIGN	RW	数据对齐方式： 1：左对齐； 0：右对齐。	0
[10:9]	Reserved	RO	保留。	0
8	DMA	RW	直接存储访问（DMA）模式使能： 1：使能 DMA 模式； 0：关闭 DMA 模式。	0
[7:4]	Reserved	RO	保留。	0
3	RSTCAL	RW	复位校准，此位由软件置位，复位完成后由硬件清 0： 1：初始化校准寄存器； 0：校准寄存器已初始化。 注：如果正在进行转换时设置 RSTCAL，清除校准寄存器需要额外的周期。	0
2	CAL	RW	A/D 校准，该位由软件置位，校准结束时由硬件清 0。	0

			1: 开始校准; 0: 校准完成。	
1	CONT	RW	连续转换使能: 1: 连续转换模式; 0: 单次转换模式。 如果设置了此位, 则转换将连续进行直到该位被清除。	0
0	ADON	RW	开/关 A/D 转换器 当该位为 0 时, 写入 1 将把 ADC 从断电模式下唤醒; 当该位为 1 时, 写入 1 将启动转换。 1: 开启 ADC 并启动转换; 0: 关闭 ADC 转换/校准, 并进入断电模式。 注: 当寄存器只有 ADON 改变时, 才会启动一次转换, 如果还有其他任意位发送变化, 则不会启动新的转换。	0

12.3.4 ADC 采样时间配置寄存器 1 (ADC_SAMPTR1)

偏移地址: 0x0C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved					SMP18[2:0]			SMP17[2:0]			SMP16[2:0]			Reserved	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved															

位	名称	访问	描述	复位值
[31:27]	Reserved	RO	保留。	0
[26:18]	SMPx[2:0]	RW	SMPx[2:0]: 通道 x 的采样时间配置: 000: 1.5 周期; 001: 7.5 周期; 010: 13.5 周期; 011: 28.5 周期; 100: 41.5 周期; 101: 55.5 周期; 110: 71.5 周期; 111: 239.5 周期; SMPx[2:0]: 通道 x 的采样时间配置 (ADC_LP=1) : 000: 7.5 周期; 001: 11.5 周期; 010: 17.5 周期; 011: 27.5 周期; 100: 47.5 周期; 101: 55.5 周期; 110: 71.5 周期; 111: 239.5 周期; 这些位用于独立地选择每个通道的采样时间, 在采样周期中通道配置值必须保持不变。	000b
[17:0]	Reserved	RO	保留。	0

12.3.5 ADC 采样时间配置寄存器 2 (ADC_SAMPTR2)

偏移地址: 0x10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved		SMP9[2:0]			SMP8[2:0]			SMP7[2:0]			SMP6[2:0]			SMP5[2:1]	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMP5[0]		SMP4[2:0]			SMP3[2:0]			SMP2[2:0]			SMP1[2:0]			SMP0[2:0]	

位	名称	访问	描述	复位值
[31:30]	Reserved	R0	保留。	0
[29:0]	SMPx[2:0]	RW	SMPx[2:0]: 通道 x 的采样时间配置 (ADC_LP=0) : 000: 1.5 周期; 001: 7.5 周期; 010: 13.5 周期; 011: 28.5 周期; 100: 41.5 周期; 101: 55.5 周期; 110: 71.5 周期; 111: 239.5 周期; SMPx[2:0]: 通道 x 的采样时间配置 (ADC_LP=1) : 000: 7.5 周期; 001: 11.5 周期; 010: 17.5 周期; 011: 27.5 周期; 100: 47.5 周期; 101: 55.5 周期; 110: 71.5 周期; 111: 239.5 周期; 这些位用于独立地选择每个通道的采样时间, 在采样周期中通道配置值必须保持不变。	000b

12.3.6 ADC 注入通道数据偏移寄存器 x (ADC_IOFRx) (x=1/2/3/4)

偏移地址: $0x14 + (x-1)*4$

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

位	名称	访问	描述	复位值
[31:12]	Reserved	R0	保留。	0
[11:0]	JOFFSETx[11:0]	RW	注入通道 x 的数据偏移值。 转换注入通道时, 这个值定义了用于从原始转换数据中减去的数值。转换的结果可以在 ADC_IDATARx 寄存器中读出。	0

12.3.7 ADC 看门狗高阈值寄存器 (ADC_WDHTR)

偏移地址: $0x24$

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

位	名称	访问	描述	复位值
[31:12]	Reserved	R0	保留。	0
[11:0]	HT[11:0]	RW	模拟看门狗高阈值设置值。	0xFFFF

注: 可以在转换过程中更改 WDHTR 和 LTR 的值, 但它们将在下次转换时生效。

12.3.8 ADC 看门狗低阈值寄存器 (ADC_WDLTR)

偏移地址：0x28

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

位	名称	访问	描述	复位值
[31:12]	Reserved	RO	保留。	0
[11:0]	LT[11:0]	RW	模拟看门狗低阈值设置值。	0

注：可以在转换过程中更改 WDHTR 和 LTR 的值，但它们将在下次转换时生效。

12.3.9 ADC 规则通道序列寄存器 1 (ADC_RSQR1)

偏移地址：0x2C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved								L[3:0]				RSQ16[4:1]			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ16[0]	SQ15[4:0]				SQ14[4:0]				SQ13[4:0]						

位	名称	访问	描述	复位值
[31:24]	Reserved	RO	保留。	0
[23:20]	L[3:0]	RW	规则通道转换序列中需要转换的通道数目： 0000-1111：1-16 个转换。	0
[19:15]	SQ16[4:0]	RW	规则序列中的第 16 个转换通道的编号（0-9，16-18）。	0
[14:10]	SQ15[4:0]	RW	规则序列中的第 15 个转换通道的编号（0-9，16-18）。	0
[9:5]	SQ14[4:0]	RW	规则序列中的第 14 个转换通道的编号（0-9，16-18）。	0
[4:0]	SQ13[4:0]	RW	规则序列中的第 13 个转换通道的编号（0-9，16-18）。	0

12.3.10 ADC 规则通道序列寄存器 2 (ADC_RSQR2)

偏移地址：0x30

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved		SQ12[4:0]				SQ11[4:0]				SQ10[4:1]					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ10[0]	SQ9[4:0]				SQ8[4:0]				SQ7[4:0]						

位	名称	访问	描述	复位值
[31:30]	Reserved	RO	保留。	0
[29:25]	SQ12[4:0]	RW	规则序列中的第 12 个转换通道的编号（0-9，16-18）。	0
[24:20]	SQ11[4:0]	RW	规则序列中的第 11 个转换通道的编号（0-9，16-18）。	0
[19:15]	SQ10[4:0]	RW	规则序列中的第 10 个转换通道的编号（0-9，16-18）。	0
[14:10]	SQ9[4:0]	RW	规则序列中的第 9 个转换通道的编号（0-9，16-18）。	0
[9:5]	SQ8[4:0]	RW	规则序列中的第 8 个转换通道的编号（0-9，16-18）。	0

[4:0]	SQ7[4:0]	RW	规则序列中的第 7 个转换通道的编号 (0-9, 16-18)。	0
-------	----------	----	----------------------------------	---

12.3.11 ADC 规则通道序列寄存器 3 (ADC_RSQR3)

偏移地址: 0x34

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				SQ6[4:0]				SQ5[4:0]				SQ4[4:1]			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ4[0]		SQ3[4:0]				SQ2[4:0]				SQ1[4:0]					

位	名称	访问	描述	复位值
[31:30]	Reserved	RO	保留。	0
[29:25]	SQ6[4:0]	RW	规则序列中的第 6 个转换通道的编号 (0-9, 16-18)。	0
[24:20]	SQ5[4:0]	RW	规则序列中的第 5 个转换通道的编号 (0-9, 16-18)。	0
[19:15]	SQ4[4:0]	RW	规则序列中的第 4 个转换通道的编号 (0-9, 16-18)。	0
[14:10]	SQ3[4:0]	RW	规则序列中的第 3 个转换通道的编号 (0-9, 16-18)。	0
[9:5]	SQ2[4:0]	RW	规则序列中的第 2 个转换通道的编号 (0-9, 16-18)。	0
[4:0]	SQ1[4:0]	RW	规则序列中的第 1 个转换通道的编号 (0-9, 16-18)。	0

12.3.12 ADC 注入通道序列寄存器 (ADC_ISQR)

偏移地址: 0x38

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved										JL[1:0]		JSQ4[4:1]			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JSQ4[0]		JSQ3[4:0]				JSQ2[4:0]				JSQ1[4:0]					

位	名称	访问	描述	复位值
[31:22]	Reserved	RO	保留。	0
[21:20]	JL[1:0]	RW	注入通道转换序列中需要转换的通道数目: 00-11: 1-4 个转换。	0
[19:15]	JSQ4[4:0]	RW	注入序列中的第 4 个转换通道的编号 (0-9, 16-18)。 注: 软件写入, 并将通道编号 (0-9, 16-18) 分配为要转换的序列中的第 4 个。	0
[14:10]	JSQ3[4:0]	RW	注入序列中的第 3 个转换通道的编号 (0-9, 16-18)。	0
[9:5]	JSQ2[4:0]	RW	注入序列中的第 2 个转换通道的编号 (0-9, 16-18)。	0
[4:0]	JSQ1[4:0]	RW	注入序列中的第 1 个转换通道的编号 (0-9, 16-18)。	0

注: 不同于规则转换序列, 如果 JL[1:0] 的长度小于 4, 则转换的序列顺序是从 (4-JL) 开始。

例如, 当 JL[1:0]=3 (定序器中有 4 次注入转换) 时, ADC 将按以下顺序转换通道: JSQ1[4:0]、JSQ2[4:0]、JSQ3[4:0] 和 JSQ4[4:0];

当 JL[1:0]=2 (定序器中有 3 次注入转换) 时, ADC 将按以下顺序转换通道: JSQ2[4:0]、JSQ3[4:0] 和 JSQ4[4:0];

当 JL[1:0]=1 (定序器中有 2 次注入转换) 时, ADC 转换通道的顺序为: 先是 JSQ3[4:0], 后是 JSQ4[4:0];

当 JL[1:0]=0 (定序器中有 1 次注入转换) 时, ADC 将仅转换 JSQ4[4:0] 通道。

如果 ADCx_ISQR[21:0]=10 0011 0001 0011 0001 0, ADC 将按以下顺序转换通道: JSQ2[4:0]、

JSQ3[4:0]和JSQ4[4:0]，表示扫描转换按以下通道顺序进行：7、3、7。

12.3.13 ADC 注入数据寄存器 x (ADC_IDATARx) (x=1/2/3/4)

偏移地址：0x3C + (x-1)*4

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

位	名称	访问	描述	复位值
[31:16]	Reserved	R0	保留。	0
[15:0]	JDATA[15:0]	R0	注入通道转换数据（数据左对齐或右对齐）。	0

12.3.14 ADC 规则数据寄存器 (ADC_RDATAR)

偏移地址：0x4C

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

位	名称	访问	描述	复位值
[31:16]	Reserved	R0	保留。	0
[15:0]	DATA	R0	规则通道转换数据（数据左对齐或右对齐）。	0

12.3.15 ADC 配置寄存器 (ADC_CFG)

偏移地址：0x50

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved													TKEY_DRV_OUTEN		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TKEY_DRV_OUTEN							TKEY_DRV_EN	ADC_DUTY_EN	FIFO_EN	ADC_LP	AWDRST_EN	ADC_BUFTRIM			

位	名称	访问	描述	复位值
[31:19]	Reserved	R0	保留。	0
[18:9]	TKEY_DRV_OUTEN	RW	TOUCHKEY 多路屏蔽各通道使能，高电平有效。	0
8	TKEY_DRV_EN	RW	TOUCHKEY 多路屏蔽使能，高电平有效。	0
7	ADC_DUTY_EN	RW	ADC 时钟占空比控制位 0：输入时钟不做处理； 1：将 ADC 时钟的高电平延长 4ns，周期不变。	0
6	FIFO_EN	RW	ADC FIFO 使能，高电平有效。	0
5	ADC_LP	RW	ADC 低功耗模式控制位：	0

			0: 低功耗模式, <code>vcmbuffer</code> 和比较器功耗较低, 适用于 1M 及以下采样率; 1: <code>vcmbuffer</code> 和比较器功耗较高, 适用于 1M 及以上采样率。	
4	<code>AWDRST_EN</code>	RW	模拟看门狗复位使能位: 0: 禁止模拟看门狗复位; 1: 使能模拟看门狗复位。	0
[3:0]	<code>ADC_BUFTRIM</code>	RW	ADC BUFFER 失调校准控制位, 其中最高位选择失调校准极性正负, [2:0]控制校准档位。	0

第 13 章 触摸按键检测 (TKEY)

触摸检测控制 (TKEY) 单元, 借助 ADC 模块的电压转换功能, 通过将电容量转换为电压量进行采样, 实现触摸按键检测功能。检测通道复用 ADC 的 10 个外部通道, 通过 ADC 模块的单次转换模式实现触摸按键检测。

13.1 TKEY 功能描述

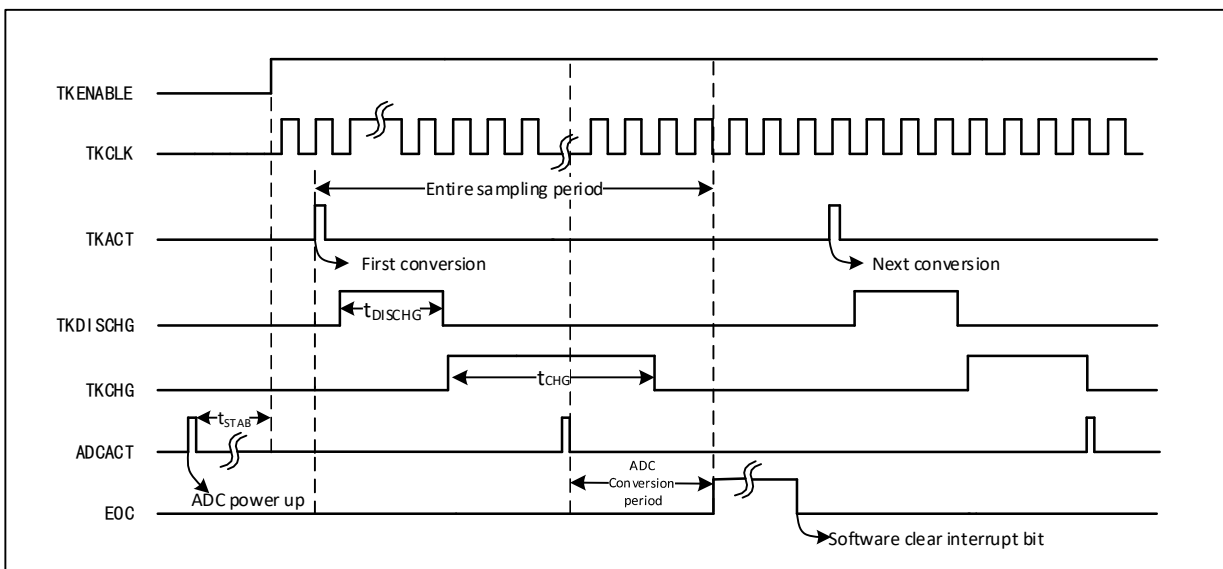
● TKEY 开启

TKEY 检测过程需要 ADC 模块配合进行, 所以使用 TKEY 功能时, 需要保证 ADC 模块处于上电状态 (ADON=1), 然后将 ADC_CTLR1 寄存器的 TKENABLE 位置 1, 打开 TKEY 单元功能, 且可以通过 TKITUNE 位调整 TKEY 模块的充电电流。

TKEY 只支持单次单通道转换模式, 将待转换的通道配置到 ADC 模块的规则组序列第一个, 软件启动转换 (写 TKEY_ACT_DCG 寄存器)。

注: 不进行 TKEY 转换时, 仍然可以保留 ADC 通道配置转换功能。

图 13-1 TKEY 工作时序图



● 可编程采样时间

TKEY 单元转换需要先使用若干个 ADCCLK 时钟周期 (t_{DISCHG}) 进行放电, 然后再通过若干个 ADCCLK 周期 (t_{CHG}) 对通道进行充电进行电压采样, 充电周期数为 ADC_SAMPTR2 寄存器中的 SMPx[2:0] 配置值加上 TKEY_CHGOFFSET 偏移量之和, 每个通道可以分别用不同的充电周期来调整采样电压。

● TKEY 多路屏蔽

TKEY_DRV_EN 位在高电平下有效, 置位时 TOUCHKEY 多路屏蔽使能, 为控制通道总开关; TKEY_DRV_OUTEN 位在高电平下有效, 使能时进行单独控制每一路通道使能。

13.2 TKEY 操作步骤

TKEY 检测属于 ADC 模块下的扩展功能, 其工作原理是通过“触摸”和“非触摸”方式让硬件通道感知的电容量发生变化, 进而通过可设置的充放电周期数将电容量的变化转换为电压的变化, 最后通过 ADC 模块转换为数字值。

采样时, 需要将 ADC 配置为单次单通道工作模式, 由 TKEY_ACT 寄存器的“写操作”启动一次转

换，具体流程如下：

- 1) 初始化 ADC 功能，配置 ADC 模块为单次转换模块，置 ADON 位为 1，唤醒 ADC 模块。将 ADC_CTLR1 寄存器的 TKENABLE 位置 1，打开 TKEY 单元。
- 2) 设置要转换的通道，将通道号写入 ADC 规则组序列中第一个转换位置（ADC_RSQR3[4:0]），设置 L[3:0] 为 1。
- 3) 设置通道的充电采样时间，写 ADC_SMPSTR2 寄存器，可为每个通道配置不同的充电时间。
- 4) 写 TKEY_CHGOFFSET 寄存器，设置通道的充电时间偏移量（低 10 位有效），以调整充电时间。
- 5) 写 TKEY_ACT_DCG 寄存器，设置放电时间（低 10 位有效），并启动一次 TKEY 的采样和转换。
- 6) 等待 ADC 状态寄存器的 EOC 转换结束标志位置 1，读取 ADC_DR 寄存器得到此次转换值。
- 7) 如果需要进行下次转换，重复 2-6 步骤。如果不需修改通道充电采样时间，可省略步骤 3 或 4。

13.3 TKEY 寄存器描述

表 13-1 TKEY 相关寄存器列表

名称	访问地址	描述	复位值
R32_TKEY_CHGOFFSET	0x4001243C	TKEY 充电时间偏移量寄存器	0x00000000
R32_TKEY_ACT_DCG	0x4001244C	TKEY 启动和放电时间寄存器	0x00000000
R32_TKEY_DR	0x4001244C	TKEY 数据寄存器	0x00000000

13.3.1 TKEY 充电时间偏移量寄存器（TKEY_CHGOFFSET）

偏移地址：0x3C

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

位	名称	访问	描述	复位值
[31:10]	Reserved	RO	保留。	0
[9:0]	TKCGOFFSET[9:0]	WO	TKEY 充电时间偏移量配置值。 总充电时间 TCHG=TKCGOFFSET+ SMPx	0

注：此寄存器映射 ADC 模块的注入数据寄存器 1（ADC_IDATAR1）。因此当该地址寄存器进行“写操作”时，作为 TKEY 充电时间偏移量（TKEY_CHGOFFSET）执行；进行“读操作”时，作为 ADC 模块的注入数据寄存器 1（ADC_IDATAR1）执行。

13.3.2 TKEY 启动和放电时间寄存器（TKEY_ACT_DCG）

偏移地址：0x4C

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

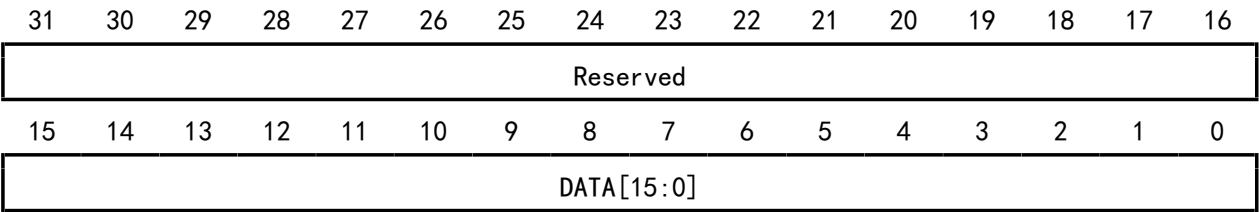
位	名称	访问	描述	复位值
[31:10]	Reserved	RO	保留。	0

[9:0]	TKACT_DCG[9:0]	WO	写放电时间并启动一次 TKEY 通道检测。 单位：系统时钟。	0
-------	----------------	----	-----------------------------------	---

注：此寄存器映射 ADC 模块的规则数据寄存器（ADC_RDATAR）。

13.3.3 TKEY 数据寄存器（TKEY_DR）

偏移地址：0x4C



位	名称	访问	描述	复位值
[31:16]	Reserved	RO	保留。	0
[15:0]	DATA[15:0]	RO	转换的数据。	0

注：此寄存器映射 ADC 模块的规则数据寄存器（ADC_RDATAR）。

第 14 章 高级定时器 (ADTM)

高级定时器模块包含一个功能强大的 16 位自动重装定时器 (TIM1)，可用于测量脉冲宽度或产生脉冲、PWM 波等。用于电机控制、电源等领域。

14.1 主要特征

高级定时器 (TIM1) 的主要特征包括：

- 16 位自动重装计数器，支持增计数模式，减计数模式和增减计数模式；
- 16 位预分频器，分频系数从 1~65536 之间动态可调；
- 支持四路独立的比较捕获通道；
- 每路比较捕获通道支持多种工作模式，比如：输入捕获，输出比较，PWM 生成和单脉冲输出；
- 支持可编程死区时间的互补输出；
- 支持外部信号控制定时器；
- 支持使用重复计数器在确定周期后更新定时器；
- 支持使用刹车信号将定时器复位或置其于确定状态；
- 支持在多种模式下使用 DMA；
- 支持增量式编码器；
- 支持定时器之间的级联和同步

14.2 原理和结构

本节主要论述高级定时器的内部构造。

14.2.1 概述

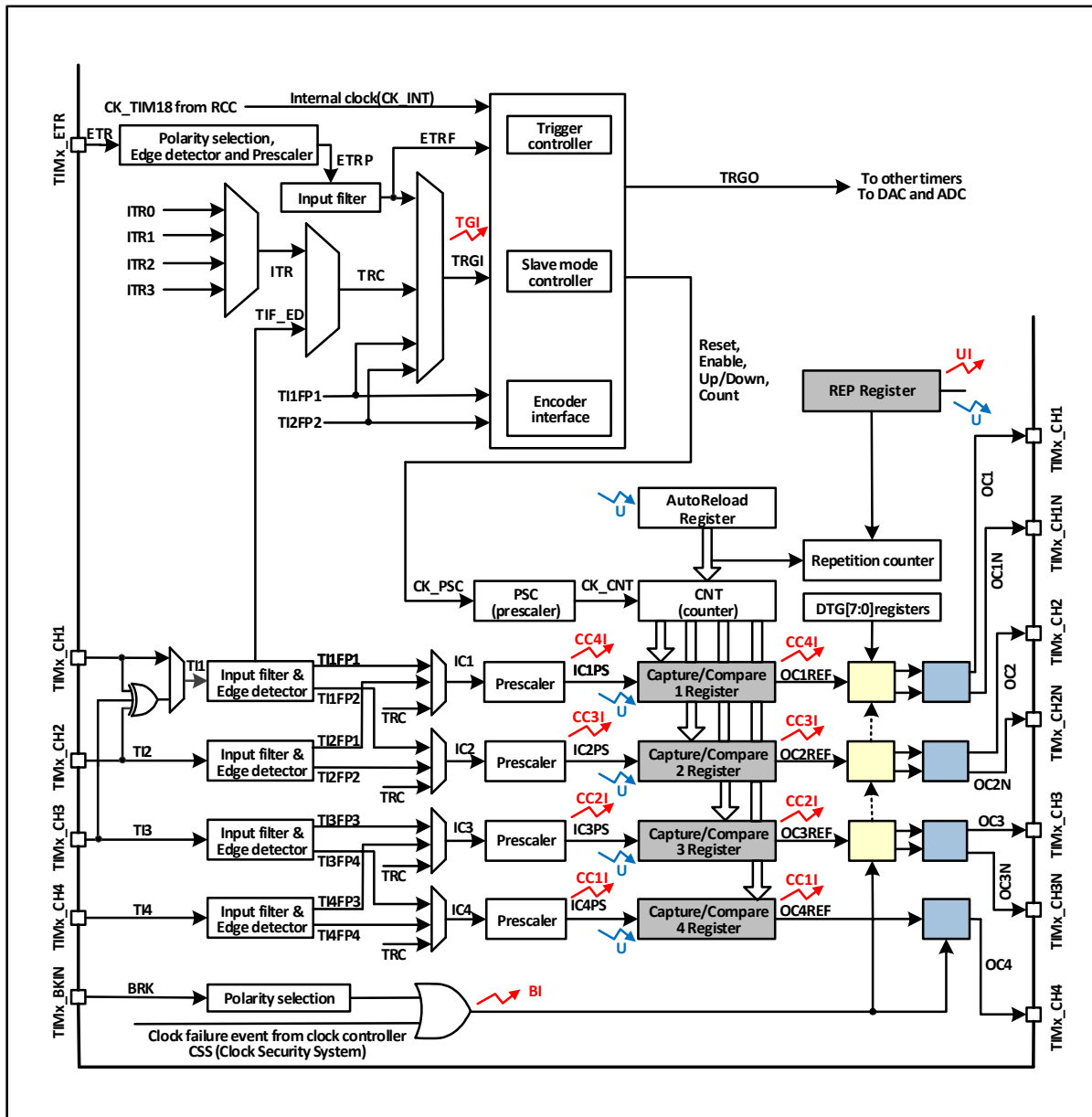
如图 14-1，高级定时器的结构大致可以分为三部分，即输入时钟部分，核心计数器部分和比较捕获通道部分。

高级定时器的时钟可以来自于 PB 总线时钟 (CK_INT)，可以来自外部时钟输入引脚 (TIMx_ETR)，亦可以来自于其他具有时钟输出功能的定时器 (ITRx)，还可以来自于比较捕获通道的输入端 (TIMx_CHx)。这些输入的时钟信号经过各种设定的滤波分频等操作后成为 CK_PSC 时钟，输出给核心计数器部分。另外，这些复杂的时钟来源还可以作为 TRGO 输出给其他的定时器、ADC 等外设。

高级定时器的核心是一个 16 位计数器 (CNT)。CK_PSC 经过预分频器 (PSC) 分频后，成为 CK_CNT 并输出给 CNT，CNT 支持增计数模式、减计数模式和增减计数模式，并有一个自动重装值寄存器 (ATRLR) 在每个计数周期结束后为 CNT 重装初始值。另外还有个辅助计数器在一旁计数 ATRLR 为 CNT 重装初始值的次数，当次数达到重复计数值寄存器 (RPTCR) 里设置的次数时，可以产生特定事件。

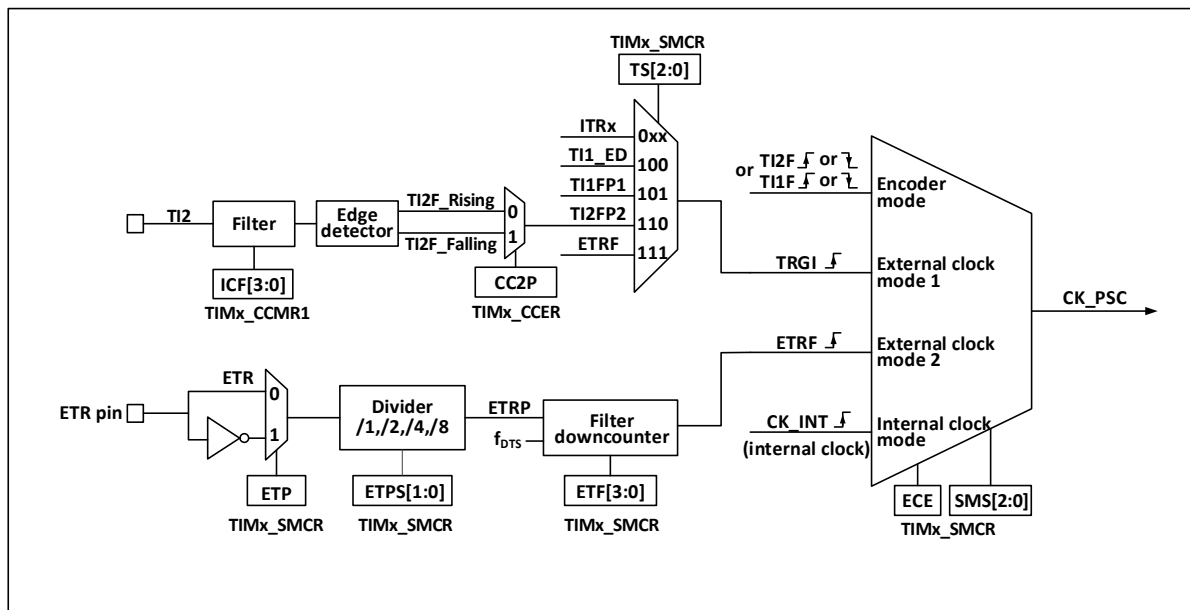
高级定时器拥有四组比较捕获通道，每组比较捕获通道都可以从专属的引脚上输入脉冲，也可以向引脚输出波形，即比较捕获通道支持输入和输出模式。比较捕获寄存器每个通道的输入都支持滤波、分频和边沿检测等操作，并支持通道间的互触发，还能为核心计数器 CNT 提供时钟。每个比较捕获通道都拥有一组比较捕获寄存器 (CHxCVR)，支持与主计数器 (CNT) 进行比较而输出脉冲。

图 14-1 高级定时器的结构框图



14.2.2 时钟输入

图 14-2 高级定时器的 CK_PSC 来源框图



高级定时器 CK_PSC 的时钟来源很多，可以分为 4 类：

- 1) 外部时钟引脚（ETR）输入时钟的路线：ETR→ETRP→ETRF；
- 2) 内部 PB 时钟输入路线：CK_INT；
- 3) 来自比较捕获通道引脚（TIMx_CHx）的路线：TIMx_CHx→Tl_x→Tl_xFP_x，此路线也用于编码器模式；
- 4) 来自内部其他定时器的输入：ITRx；

通过决定 CK_PSC 来源的 SMS 的输入脉冲选择可以将实际的操作分为 4 类：

- 1) 选择内部时钟源（CK_INT）；
- 2) 外部时钟源模式 1；
- 3) 外部时钟源模式 2；
- 4) 编码器模式；

上文提到的 4 种时钟源来源都可通过这 4 种操作选定。

14.2.2.1 内部时钟源（CK_INT）

如果将 SMS 域保持 000b 时启动高级定时器，那么就是选定内部时钟源（CK_INT）为时钟。此时 CK_INT 就是 CK_PSC。

14.2.2.2 外部时钟源模式 1

如果将 SMS 域设置为 111b 时，就会启用外部时钟源模式 1。启用外部时钟源 1 时，TRGI 被选定为 CK_PSC 的来源，值得注意的，还需要通过配置 TS 域来选择 TRGI 的来源。TS 域可选择以下几种脉冲作为时钟来源：

- 1) 内部触发（ITRx，x 为 0, 1, 2, 3）；
- 2) 比较捕获通道 1 经过边缘检测器后的信号（TI1F_ED）；
- 3) 比较捕获通道的信号 TI1FP1、TI2FP2；
- 4) 来自外部时钟引脚输入的信号 ETRF。

14.2.2.3 外部时钟源模式 2

使用外部触发模式 2 能在外部时钟引脚输入的每一个上升沿或下降沿计数。将 ECE 位置位时，将使用外部时钟源模式 2。使用外部时钟源模式 2 时，ETRF 被选定为 CK_PSC。ETR 引脚经过可选的

反相器（ETP），分频器（ETPS）后成为 ETRP，再经过滤波器（ETF）后即成为 ETRF。

在 ECE 位置位且将 SMS 设为 111b 时，相当于 TS 选择 ETRF 为输入。

14.2.2.4 编码器模式

将 SMS 置为 001b, 010b, 011b 将会启用编码器模式。启用编码器模式可以选择在 TI1FP1 和 TI2FP2 中某一个特定的电平下以另一个跳变沿作为信号进行信号输出。此模式用于外接编码器使用的情况下。具体功能参考 14.3.10 节。

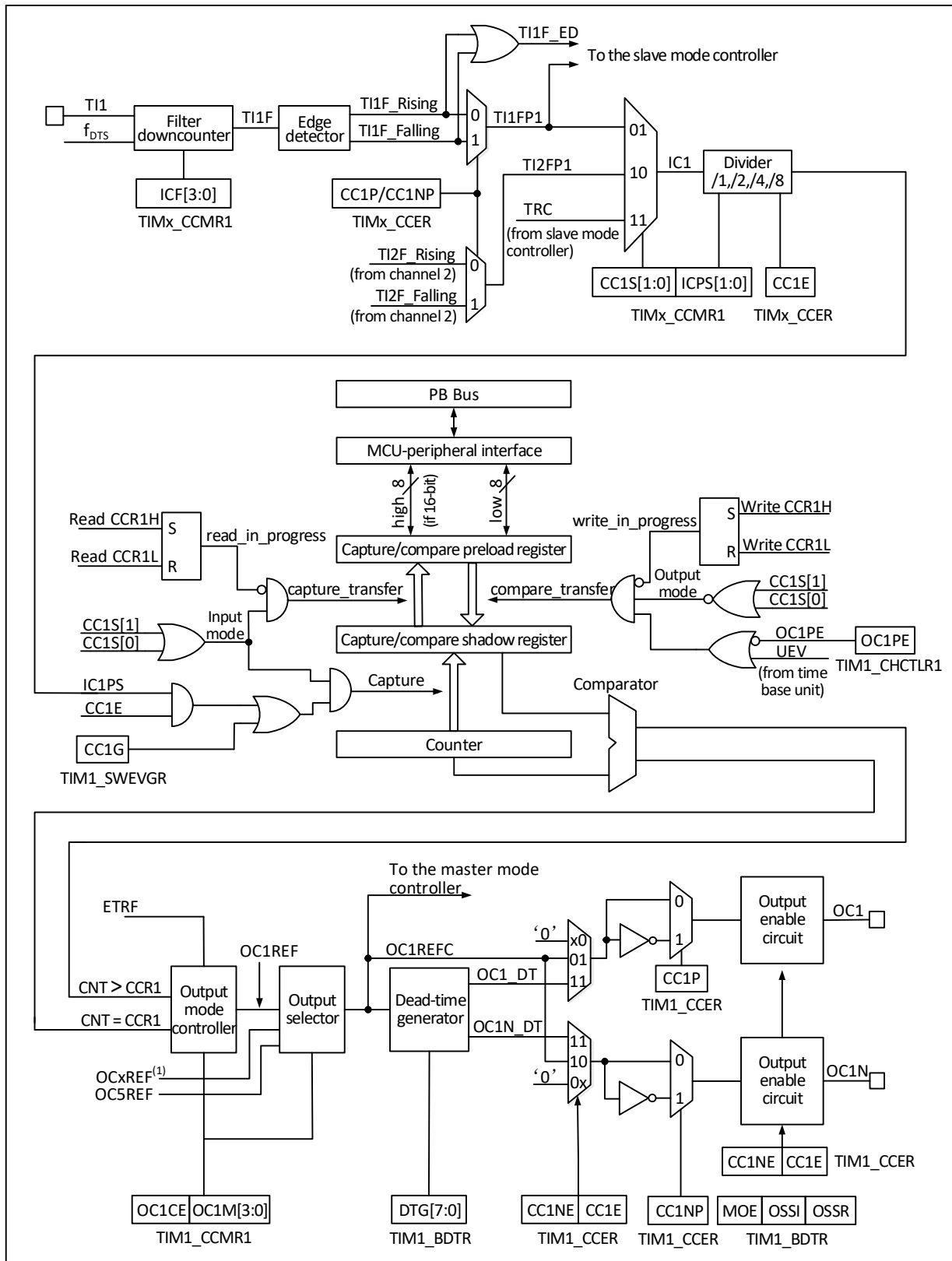
14.2.3 计数器和周边

CK_PSC 输入给预分频器（PSC）进行分频。PSC 是 16 位的，实际的分频系数相当于 R16_TIMx_PSC 的值+1。CK_PSC 经过 PSC 会成为 CK_INT。更改 R16_TIM1_PSC 的值并不会实时生效，而会在更新事件后更新给 PSC。更新事件包括 UG 位清零和复位。定时器的核心是一个 16 位计数器（CNT），CK_CNT 最终会输入给 CNT，CNT 支持增计数模式、减计数模式和增减计数模式，并有一个自动重装值寄存器（ATRLR）在每个计数周期结束后为 CNT 重新装载初始值。另外还有个辅助计数器在一旁记录 ATRLR 为 CNT 重新装载初值的次数，当达到重复计数值寄存器（RPTCR）里设置的次数时，可以产生特定事件。

14.2.4 比较捕获通道和周边

比较捕获通道是定时器实现复杂功能的主要组件，它的核心是比较捕获寄存器，辅以外围输入部分的数字滤波，分频和通道间复用、输出部分的比较器和输出控制组成。

图 14-3 比较捕获通道的结构框图



比较捕获通道的结构框图如图 14-3 所示。信号从通道 x 引脚输入进来后可选做为 TIx ($TI1$ 的来源可以不只是 $CH1$, 见定时器的结构框图 14-1), $TI1$ 经过滤波器 (ICF[3:0]) 生成 $TI1F$, 再经过边沿检测器分成 $TI1F_Rising$ 和 $TI1F_Falling$, 这两个信号经过选择 ($CC1P$) 生成 $TI1FP1$, $TI1FP1$ 和来自通道 2 的 $TI2FP1$ 一起送给 $CC1S$ 选择成为 $IC1$, 经过 $ICPS$ 分频后送给比较捕获寄存器。

比较捕获寄存器由一个预装载寄存器和一个影子寄存器组成, 读写过程仅操作预装载寄存器。

在捕获模式下，捕获发生在影子寄存器上，然后复制到预装载寄存器；在比较模式下，预装载寄存器的内容被复制到影子寄存器中，然后影子寄存器的内容与核心计数器（CNT）进行比较。

14.3 功能和实现

高级定时器复杂功能的实现都是对定时器的比较捕获通道、时钟输入电路和计数器及周边部分的操作实现的。定时器的时钟输入可以来自于包括比较捕获通道的输入在内的多个时钟源。对比较捕获通道和时钟源选择的操作直接决定其功能。比较捕获通道是双向的，可以工作在输入和输出模式。

14.3.1 计数器模式

递增计数模式

在递增计数模式下，计数器从 0 计数到自动重载值（R16_TIMx_ATRLR 寄存器的内容），然后重新从 0 开始计数并生成计数器上溢事件。

如果使用重复计数器，则当递增计数的重复次数达到重复计数器寄存器中编程的次数加一次（R16_TIMx_RPTCR+1）后，将生成更新事件（UEV）。否则，将在每次计数器上溢时产生更新事件。

将 TIMx_SWEVGR 寄存器的 UG 位置 1（通过软件或使用从模式控制器）时，也将产生更新事件。

通过软件将 R16_TIMx_CTLR1 寄存器中的 UDIS 位置 1 可禁止 UEV 事件。这可避免向预装载寄存器写入新值时更新影子寄存器。在 UDIS 位写入 0 之前不会产生任何更新事件。不过，计数器和预分频器计数器都会重新从 0 开始计数（而预分频比保持不变）。此外，如果 R16_TIMx_CTLR1 寄存器中的 URS 位（更新请求选择）已置 1，则将 UG 位置 1 会生成更新事件 UEV，但不会将 UIF 标志置 1（因此，不会发送任何中断或 DMA 请求）。这样一来，如果在发生捕获事件时将计数器清零，将不会同时产生更新中断和捕获中断。

发生更新事件时，将更新所有寄存器且将更新标志（R16_TIMx_INTFR 寄存器中的 UIF 位）置 1（取决于 URS 位）：

- 1) 重复计数器中将重新装载 R16_TIMx_RPTCR 寄存器的内容
- 2) 自动重载影子寄存器将以预装载值（R16_TIMx_ATRLR）进行更新
- 3) 预分频器的缓冲区中将重新装载预装载值（R16_TIMx_PSC 寄存器的内容）

递减计数模式

在递减计数模式下，计数器从自动重载值（R16_TIMx_ATRLR 寄存器的内容）开始递减计数到 0，然后重新从自动重载值开始计数并生成计数器下溢事件。

如果使用重复计数器，则当递减计数的重复次数达到重复计数器寄存器中编程的次数加一次（R16_TIMx_RPTCR+1）后，将生成更新事件（UEV）。否则，将在每次计数器下溢时产生更新事件。

将 R16_TIMx_EGR 寄存器的 UG 位置 1（通过软件或使用从模式控制器）时，也将产生更新事件。

通过软件将 R16_TIMx_CTLR1 寄存器中的 UDIS 位置 1 可禁止 UEV 更新事件。这可避免向预装载寄存器写入新值时更新影子寄存器。在 UDIS 位写入 0 之前不会产生任何更新事件。不过，计数器会重新从当前自动重载值开始计数，而预分频器计数器则重新从 0 开始计数（但预分频比保持不变）。

此外，如果 R16_TIMx_CTLR1 寄存器中的 URS 位（更新请求选择）已置 1，则将 UG 位置 1 会生成更新事件 UEV，但不会将 UIF 标志置 1（因此，不会发送任何中断或 DMA 请求）。这样一来，如果在发生捕获事件时将计数器清零，将不会同时产生更新中断和捕获中断。

发生更新事件时，将更新所有寄存器且将更新标志（R16_TIMx_INTFR 寄存器中的 UIF 位）置 1（取决于 URS 位）：

- 1) 重复计数器中将重新装载 R16_TIMx_RPTCR 寄存器的内容
- 2) 预分频器的缓冲区中将重新装载预装载值（R16_TIMx_PSC 寄存器的内容）
- 3) 自动重载活动寄存器将以预装载值（R16_TIMx_ATRLR 寄存器的内容）进行更新。

注：自动重载寄存器会在计数器重载之前得到更新，因此，下一个计数周期就是我们所希望的新的周期长度。

中心对齐模式（递增/递减计数）

在中心对齐模式下，计数器从 0 开始计数到自动重载值（R16_TIMx_ATRLR 寄存器的内容）—1，生成计数器上溢事件；然后从自动重载值开始向下计数到 1 并生成计数器下溢事件。之后从 0 开始重新计数。

当 R16_TIMx_CTLR1 寄存器中的 CMS 位不为“00”时，中心对齐模式有效。将通道配置为输出模式时，其输出比较中断标志将在以下模式下置 1，即：计数器递减计数（中心对齐模式 1，CMS=“01”）、计数器递增计数（中心对齐模式 2，CMS=“10”）以及计数器递增/递减计数（中心对齐模式 3，CMS=“11”）。

在此模式下，R16_TIMx_CTLR1 寄存器的 DIR 方向位不可写入值，而是由硬件更新并指示当前计数器方向。

每次发生计数器上溢和下溢时都会生成更新事件，或将 R16_TIMx_SWEVGR 寄存器中的 UG 位置 1（通过软件或使用从模式控制器）也可以生成更新事件。这种情况下，计数器以及预分频器计数器将重新从 0 开始计数。

通过软件将 R16_TIMx_CTLR1 寄存器中的 UDIS 位置 1 可禁止 UEV 更新事件。这可避免向预装载寄存器写入新值时更新影子寄存器。在 UDIS 位写入 0 之前不会产生任何更新事件。不过，计数器仍会根据当前自动重载值进行递增和递减计数。

此外，如果 R16_TIMx_CTLR1 寄存器中的 URS 位（更新请求选择）已置 1，则将 UG 位置 1 会生成 UEV 更新事件，但不会将 UIF 标志置 1（因此，不会发送任何中断或 DMA 请求）。这样一来，如果在发生捕获事件时将计数器清零，将不会同时产生更新中断和捕获中断。

发生更新事件时，将更新所有寄存器且将更新标志（R16_TIMx_INTFR 寄存器中的 UIF 位）置 1（取决于 URS 位）：

- 1) 重复计数器中将重新装载 R16_TIMx_RPTCR 寄存器的内容
- 2) 预分频器的缓冲区中将重新装载预装载值（R16_TIMx_PSC 寄存器的内容）
- 3) 自动重载活动寄存器将以预装载值（R16_TIMx_ATRLR 寄存器的内容）进行更新。注意，如果更新操作是由计数器上溢触发的，则自动重载寄存器在重载计数器之前更新，因此，下一个计数周期就是我们所希望的新的周期长度（计数器被重载新的值）。

14.3.2 输入捕获模式

输入捕获模式是定时器的基本功能之一。输入捕获模式的原理是，当检测到 ICxPS 信号上确定的边沿后，则发生捕获事件，计数器当前的值会被锁存到比较捕获寄存器（R16_TIMx_CHCTLRx）中。发生捕获事件时，CCxIF（在 R16_TIMx_INTFR 中）被置位，如果使能了中断或 DMA，还会产生相应中断或 DMA。如果发生捕获事件时，CCxIF 已经被置位了，那么 CCxOF 位会被置位。CCxIF 可由软件清除，也可以通过读取比较捕获寄存器由硬件清除。CCxOF 由软件清除。

举个通道 1 的例子来说明使用输入捕获模式的步骤，如下：

- 1) 配置 CCxS 域，选择 ICx 信号的来源。比如设为 10b，选择 TI1FP1 作为 IC1 的来源，而不可以使用默认设置，CCxS 域默认是使比较捕获模块作为输出通道；
- 2) 配置 ICxF 域，设定 TI 信号的数字滤波器。数字滤波器会以确定的频率，采样确定的次数，再输出一个跳变。这个采样频率和次数是通过 ICxF 来确定的；
- 3) 配置 CCxP 位，设定 TIxFPx 的极性。比如保持 CC1P 位为低，选择上升沿跳变；
- 4) 配置 ICxPS 域，设定 ICx 信号成为 ICxPS 之间的分频系数。比如保持 ICxPS 为 00b，不分频；
- 5) 配置 CCxE 位，允许捕获核心计数器（CNT）的值到比较捕获寄存器中。置 CC1E 位；
- 6) 根据需要配置 CCxIE 和 CCxDE 位，决定是否允许使能中断或 DMA。

至此已经将比较捕获通道配置完成。

当 TI1 输入了一个被捕获的脉冲时,核心计数器(CNT)的值会被记录到比较捕获寄存器中,CC1IF 被置位,当 CC1IF 在之前就已经被置位时,CC1OF 位也会被置位。如果 CC1IE 位,那么会产生一个中断;如果 CC1DE 被置位,会产生一个 DMA 请求。可以通过写事件产生寄存器(TIMx_SWEVGR)的方式由软件产生一个输入捕获事件。

14.3.3 比较输出模式

比较输出模式是定时器的基本功能之一。比较输出模式的原理是在核心计数器(CNT)的值与比较捕获寄存器的值一致时,输出特定的变化或波形。OCxM 域(在 R16_TIMx_CHCTLRx 中)和 CCxP 位(在 R16_TIMx_CCER 中)决定输出的是确定的高低电平还是电平翻转。产生比较一致事件时还会置 CCxIF 位,如果预先置了 CCxIE 位,则会产生一个中断;如果预先设置了 CCxDE 位,则会产生一个 DMA 请求。

配置为比较输出模式的步骤为下:

- 1) 配置核心计数器(CNT)的时钟源和自动重装值;
- 2) 设置需要对比的计数值到比较捕获寄存器(R16_TIMx_CHxCVR)中;
- 3) 如果需要产生中断,置 CCxIE 位;
- 4) 保持 OCxPE 为 0,禁用比较寄存器的预装载寄存器;
- 5) 设定输出模式,设置 OCxM 域和 CCxP 位;
- 6) 使能输出,置 CCxE 位;
- 7) 置 CEN 位启动定时器。

14.3.4 强制输出模式

定时器的比较捕获通道的输出模式可以由软件强制输出确定的电平,而不依赖比较捕获寄存器的影子寄存器和核心计数器的比较。

具体的做法是将 OCxM 置为 100b,即为强制将 OCxREF 置为低;或者将 OCxM 置为 101b,即为强制将 OCxREF 置为高。

需要注意的是,将 OCxM 强制置为 100b 或者 101b,内部核心计数器和比较捕获寄存器的比较过程还在进行,相应的标志位还在置位,中断和 DMA 请求还在产生。

14.3.5 PWM 输入模式

PWM 输入模式是用来测量 PWM 的占空比和频率的,是输入捕获模式的一种特殊情况。除下列区别外,操作和输入捕获模式相同: PWM 占用两个比较捕获通道,且两个通道的输入极性设为相反,其中一个信号被设为触发输入, SMS 设为复位模式。

例如,测量从 TI1 输入的 PWM 波的周期和频率,需要进行以下操作:

- 1) 将 TI1(TI1FP1)设为 IC1 信号的输入。将 CC1S 置为 01b;
- 2) 将 TI1FP1 置为上升沿有效。将 CC1P 保持为 0;
- 3) 将 TI1(TI1FP2)置为 IC2 信号的输入。将 CC2S 置为 10b;
- 4) 选 TI1FP2 置为下降沿有效。将 CC2P 置为 1;
- 5) 时钟源的来源选择 TI1FP1。将 TS 设为 101b;
- 6) 将 SMS 设为复位模式,即 100b;
- 7) 使能输入捕获。CC1E 和 CC2E 置位;

这样比较捕获寄存器 1 的值就是 PWM 的周期,而比较捕获寄存器 2 的值就是其占空比。

14.3.6 PWM 输出模式

PWM 输出模式是定时器的基本功能之一。PWM 输出模式最常见的是使用重装值确定 PWM 频率,使用捕获比较寄存器确定占空比的方法。将 OCxM 域中置 110b 或 111b 使用 PWM 模式 1 或模式 2,置 OCxPE 位使能预装载寄存器,最后置 ARPE 位使能预装载寄存器的自动重装。由于在发生一个更新事件时,

预装载寄存器的值才能被送到影子寄存器，所以在核心计数器开始计数之前，需要置 UG 位来初始化所有寄存器。在 PWM 模式下，核心计数器和比较捕获寄存器一直在进行比较，根据 CMS 位，定时器能够输出边沿对齐或中央对齐的 PWM 信号。

● 边沿对齐

使用边沿对齐时，核心计数器增计数或减计数，在 PWM 模式 1 的情景下，在核心计数器的值大于比较捕获寄存器时，OCxREF 为高；当核心计数器的值小于比较捕获寄存器时（比如核心计数器增长到 R16_TIMx_ATRLR 的值而恢复成全 0 时），OCxREF 为低。

● 中央对齐

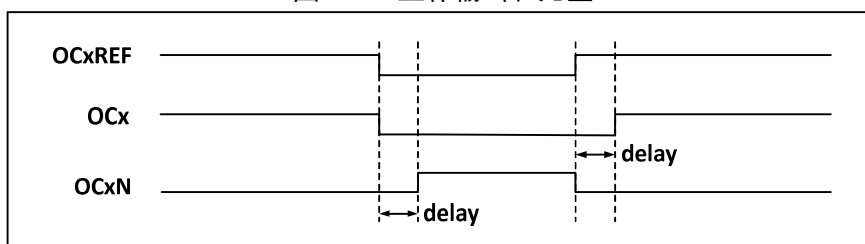
使用中央对齐模式时，核心计数器运行在增计数和减计数交替进行的模式下，OCxREF 在核心计数器和比较捕获寄存器的值一致时进行上升和下降的跳变。但比较标志在三种中央对齐模式下，置位的时机有所不同。在使用中央对齐模式时，最好在启动核心计数器之前产生一个软件更新标志（置 UG 位）。

14.3.7 互补输出和死区

比较捕获通道一般有两个输出引脚（比较捕获通道 4 只有一个输出引脚），能输出两个互补的信号（OCx 和 OCxN），OCx 和 OCxN 可以通过 CCxP 和 CCxNP 位独立地设置极性，通过 CCxE 和 CCxNE 独立地设置输出使能，通过 MOE、OIS、OISN、OSS1、OSSR 位进行死区和其他的控制。同时使能 OCx 和 OCxN 输出将插入死区，每个通道都有一个 10 位的死区发生器。如果存在刹车电路则还要设置 MOE 位。OCx 和 OCxN 由 OCxREF 关联产生，如果 OCx 和 OCxN 都是高有效，那么 OCx 与 OCxREF 相同，只是 OCx 的上升沿相当于 OCxREF 有一个延迟，OCxN 与 OCxREF 相反，它的上升沿相对参考信号的下降沿会有一个延迟，如果延迟大于有效输出宽度，则不会产生相应的脉冲。

如图 14-4 展示了 OCx 和 OCxN 与 OCxREF 的关系，并展示出死区。

图 14-4 互补输出和死区



14.3.8 刹车信号

当产生刹车信号时，输出使能信号和无效电平都会根据 MOE、OIS、OISN、OSS1 和 OSSR 等位进行修改。但 OCx 和 OCxN 不会在任何时间都处在有效电平。刹车事件源可以来自于刹车输入引脚，也可以是一个时钟失败事件，而时钟失败事件由 CSS（时钟安全系统）产生。

在系统复位后，刹车功能被默认禁止（MOE 位为低），置 BKE 位可以使能刹车功能，输入的刹车信号的极性可以通过设置 BKP 设置，BKE 和 BKP 信号可以被同时写入，在真正写入之前会有一个 PB 时钟的延迟，因此需要等一个 PB 周期才能正确读出写入值。

在刹车引脚出现选定的电平系统将产生如下动作：

- 1) MOE 位被异步清零，根据 S00I 位的设置将输出置为无效状态、空闲状态或复位状态；
- 2) 在 MOE 被清零后，每一个输出通道输出由 OISx 确定的电平；
- 3) 当使用互补输出时：输出被置于无效状态，具体取决于极性；
- 4) 如果 BIE 被置位，当 BIF 置位，会产生一个中断；如果设置了 BDE 位，则会产生一个 DMA 请求；
- 5) 如果 AOE 被置位，在下一个更新事件 UEV 时，MOE 位被自动置位。

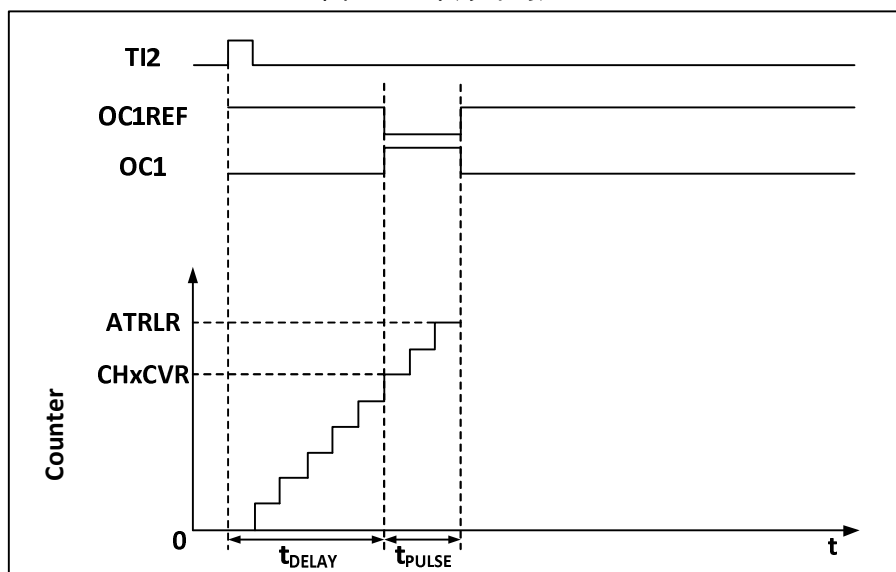
14.3.9 单脉冲模式

单脉冲模式可以用于让微控制器响应一个特定的事件，使之在一个延迟之后产生一个脉冲，延

迟和脉冲的宽度可编程。置 OPM 位可以使核心计数器在产生下一个更新事件 UEV 时（计数器翻转到 0）停止。

如图 14-5，需要在 TI2 输入引脚上检测到一个上升沿开始，延迟 T_{delay} 之后，在 OC1 上产生一个长度为 T_{pulse} 的正脉冲：

图 14-5 单脉冲的产生



- 1) 设定 TI2 为触发。置 CC2S 域为 01b，把 TI2FP2 映射到 TI2；置 CC2P 位为 0b，TI2FP2 设为上升沿检测；置 TS 域为 110b，TI2FP2 设为触发源；置 SMS 域为 110b，TI2FP2 被用来启动计数器；
- 2) T_{delay} 由比较捕获寄存器的值确定， T_{pulse} 由自动重装值寄存器的值和比较捕获寄存器的值确定。

14.3.10 编码器模式

编码器模式是定时器的一个典型应用，可以用来接入编码器的双相输出，核心计数器的计数方向和编码器的转轴方向同步，编码器每输出一个脉冲就会使核心计数器加一或减一。使用编码器的步骤为：将 SMS 域置为 001b（只在 TI2 边沿计数）、010b（只在 TI1 边沿计数）或 011b（在 TI1 和 TI2 双边沿计数），将编码器接到比较捕获通道 1、2 的输入端，给重装值寄存器设一个值，这个值可以设的大一点。在编码器模式时，定时器内部的比较捕获寄存器，预分频器，重复计数寄存器等都正常工作。下表表明了计数方向和编码器信号的关系。

表 14-1 定时器编码器模式的计数方向和编码器信号之间的关系

计数有效边沿	相对信号的 电平	TI1FP1 信号边沿		TI2FP2 信号	
		上升沿	下降沿	上升沿	下降沿
仅在 TI1 边沿计数	高	向下计数	向上计数	不计数	
	低	向上计数	向下计数		
仅在 TI2 边沿计数	高	不计数		向上计数	向下计数
	低			向下计数	向上计数
在 TI1 和 TI2 双边沿计数	高	向下计数	向上计数	向上计数	向下计数
	低	向上计数	向下计数	向下计数	向上计数

14.3.11 TIMx 定时器和外部触发的同步

定时器能够在复位模式、门控模式和触发模式下和一个外部触发同步。

从模式：复位模式

计数器及其预分频器可以响应触发输入事件而被重新初始化；如果 R16_TIMx_CTLR1 寄存器的

URS 位为低，则产生一个更新事件 UEV；然后更新所有的预装载寄存器 (R16_TIMx_ATRLR, R16_TIMx_CHxCVR)。

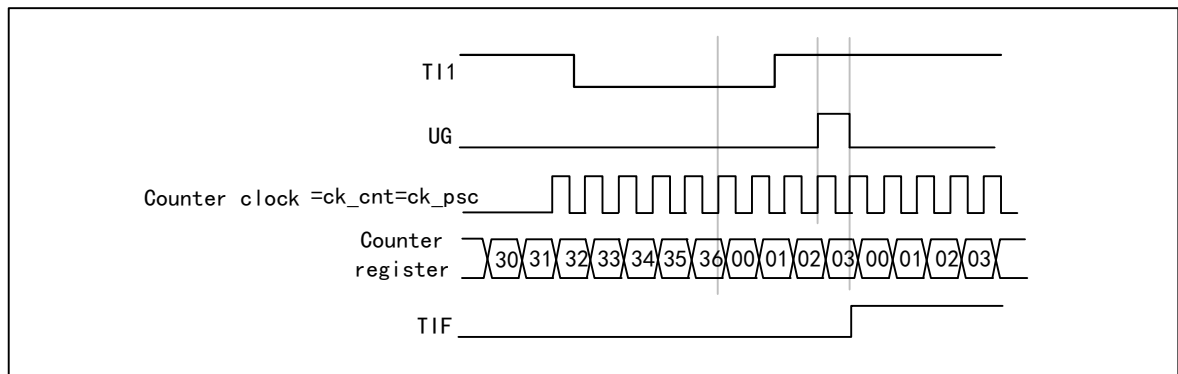
以下示例中，当 TI1 输入出现上升沿，向上计数器清零：

- 1) 配置通道 1 以检测 TI1 的上升沿。配置输入滤波器带宽 (本例不需要任何滤波器，因此保持 IC1F=0000)。无需配置捕获分频器，因为其不用于触发操作。CC1S 位只选择输入捕获源，即 CC1S=01 (在 R16_TIMx_CCMR1 中)。将 CC1P=0 和 CC1NP='0' 写入 R16_TIMx_CCER 寄存器以验证极性 (只检测上升沿)。
- 2) 将 SMS=100 写入 R16_TIMx_SMCGR，配置定时器为复位模式；将 TS=101 写入 R16_TIMx_SMCGR，选择 TI1 作为输入源。
- 3) 将 CEN=1 写入 R16_TIMx_CTLR1，启动计数器。

计数器使用内部时钟计数，然后正常运转，当出现一个 TI1 上升沿，计数器清零并从 0 开始重新计数。同时，触发标志 TIF 位置 1，使能中断或 DMA 后，可以发送中断或 DMA 请求。(取决于 R16_TIMx_DMAINTENR 寄存器中 TIE (中断使能) 位和 TDE (DMA 使能) 位)。

下图显示当自动重载寄存器 R16_TIMx_ARR=0x36 时的动作。TI1 上升沿与实际计数器复位之间的延时是 TI1 输入端的重新同步电路造成的。

图 14-6 复位模式下的控制电路



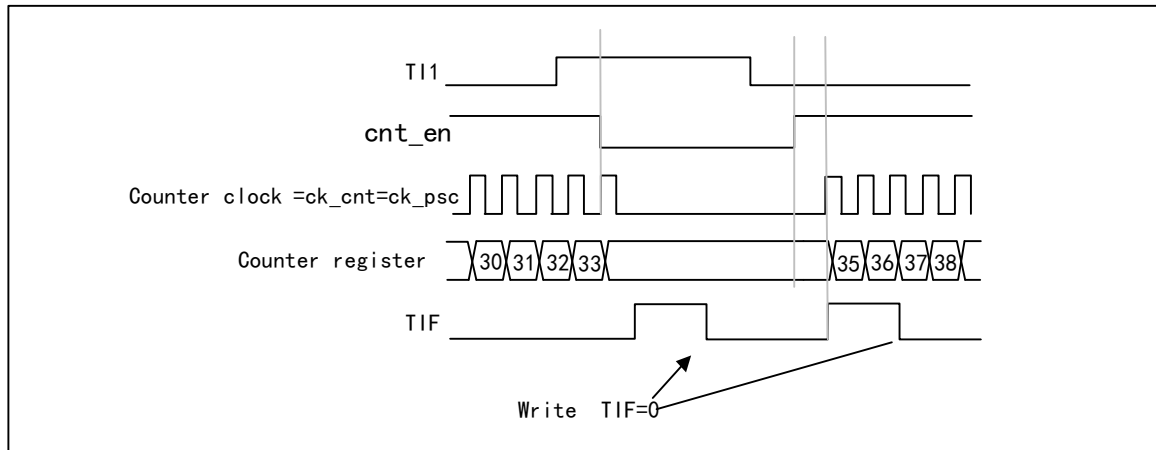
从模式：门控模式

输入信号的电平使能计数器。以下的例子中，仅 TI1 为低时计数器向上计数：

- 1) 配置通道 1 以检测 TI1 上的低电平。配置输入滤波器带宽 (本例不需要任何滤波器，因此保持 IC1F=0000)。无需配置捕获分频器，因为其不用于触发操作。CC1S 位只选择输入捕获源，即 CC1S=01 (在 R16_TIMx_CCMR1 中)。将 CC1P=1 和 CC1NP='0' 写入 R16_TIMx_CCER 寄存器以验证极性 (只检测低电平)。
- 2) 将 SMS=101 写入 R16_TIMx_SMCGR，配置定时器为门控模式；将 TS=101 写入 R16_TIMx_SMCGR，选择 TI1 作为输入源。
- 3) 将 CEN=1 写入 R16_TIMx_CTLR1，启动计数器。门控模式下，如果 CEN=0，无论触发输入电平如何，计数器都不会启动。

只要 TI1 为低，计数器开始依据内部时钟计数，TI1 变高时停止计数。当计数器开始或停止时都将 R16_TIMx_INTFR 中的 TIF 位置 1。TI1 上升沿与实际计数器复位之间的延时是 TI1 输入端的重新同步电路造成的。

图 14-7 门控模式下的控制电路



从模式：触发模式

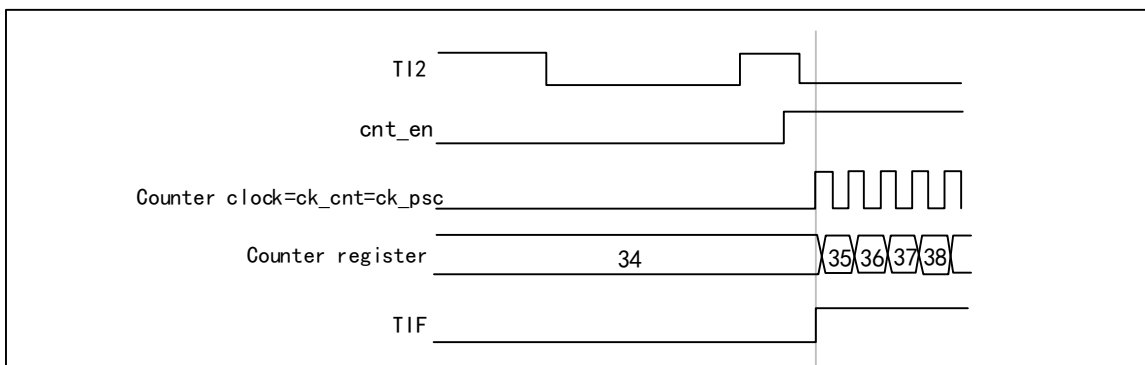
选择的输入端上发生事件时将使能计数器。以下示例中，当 TI2 输入出现上升沿，向上计数器启动：

- 1) 配置通道 2 以检测 TI2 的上升沿。配置输入滤波器带宽(本例不需要任何滤波器，因此保持 IC2F=0000)。无需配置捕获分频器，因为其不用于触发操作。CC2S 位只选择输入捕获源，置 CC2S=01 (在 R16_TIMx_CCMR1 中)。将 CC2P=1 和 CC2NP= '0' 写入 R16_TIMx_CCER 寄存器以验证极性 (只检测低电平)
- 2) 将 SMS=110 写入 R16_TIMx_SMCFGR 寄存器，配置定时器为触发模式；将 TS=110 写入 R16_TIMx_SMCFGR 寄存器，选择 TI2 作为输入源。

当 TI2 出现上升沿，计数器开始在内部时钟驱动下计数，同时 TIF 置 1。

TI2 上升沿和实际计数器启动之间的延时，是由 TI2 输入端的重同步电路造成。

图 14-8 触发模式下的控制电路



从模式：外部时钟模式 2+触发模式

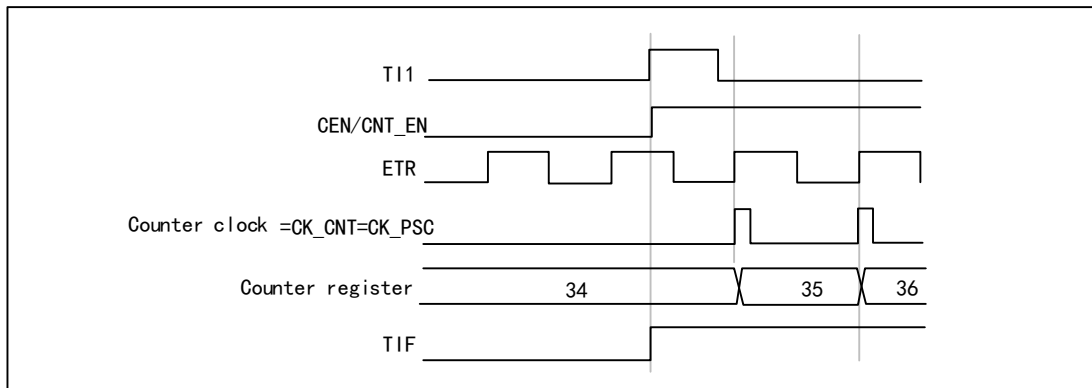
外部时钟模式 2 可结合除外部时钟模式 1 和编码器模式以外的另一种从模式一起使用。此时，ETR 信号用作外部时钟的输入，在复位模式、门控模式或触发模式下，另一个输入可作为触发输入。不建议通过 R16_TIMx_SMCFGR 寄存器的 TS 位来选择 ETR 作为 TRGI。以下示例中，一旦在 TI1 上出现一个上升沿，向上计数器即在 ETR 的每一个上升沿递增：

- 1) 配置 R16_TIMx_SMCFGR 寄存器以配置外部触发输入电路：
 - ETF=0000：没有滤波；
 - ETPS=00：不用预分频器；
 - ETP=0：检测 ETR 的上升沿，置 ECE=1 使能外部时钟模式 2。
- 2) 配置通道 1 以检测 TI 的上升沿：

- IC1F=0000：没有滤波；
 - 无需配置捕获分频器，因为其不用于触发操作；
 - 置 R16_TIMx_CHCTLR1 寄存器中 CC1S=01，选择输入捕获源；
 - 置 R16_TIMx_CCER 寄存器中 CC1P=0 以确定极性（只检测上升沿）。
- 3) 将 SMS=110 写入 R16_TIMx_SMCFGR 寄存器，配置定时器为触发模式。将 TS=101 写入 R16_TIMx_SMCFGR 寄存器，选择 TI1 作为输入源。

当 TI1 出现上升沿，使能计数器，TIF 置 1，计数器开始在 ETR 的上升沿计数。ETR 信号的上升沿和实际计数器复位间的延时，由 ETRP 输入端的重同步电路造成。

图 14-9 外部时钟模式 2+触发模式下的控制电路

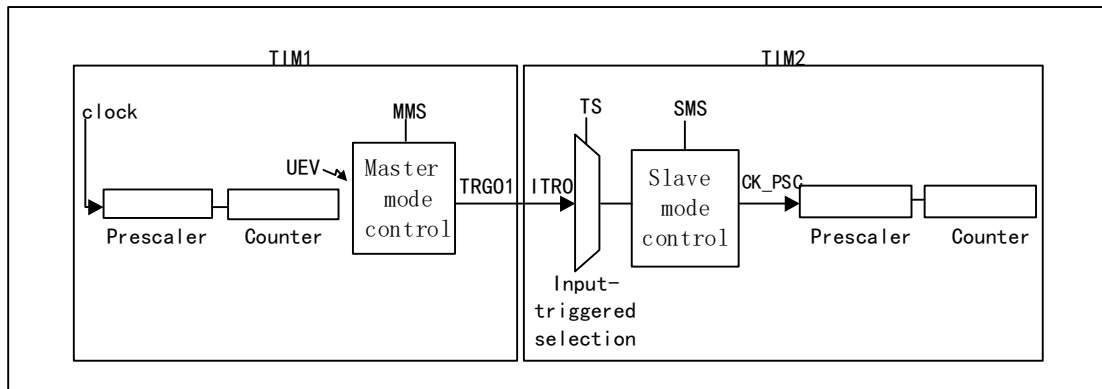


14.3.12 定时器同步模式

TIMx 定时器从内部连接在一起，以实现定时器同步或级联。当某个定时器配置为主模式时，可对另一个配置为从模式的定时器的计数器执行复位、启动、停止操作或为其提供时钟。

将一个定时器用作另一个定时器的预分频器

图 14-10 主/从定时器示例



例如，可以将定时器 1 配置为定时器 2 的预分频器。为此：

- 1) 将定时器 1 配置为主模式，每次发生更新事件 UEV 时都输出一个周期性触发信号。如果将 MMS=010 写入 R16_TIM1_CTLR2，则每当生成更新事件，TRG01 都会输出一个上升沿。
- 2) 要将定时器 1 的 TRG01 输出连接到定时器 2，必须将定时器 2 配置为从模式，使用 ITRO 作为内部触发。通过 R16_TIM2_SMCFGR 寄存器中的 TS 位（写入 TS=000）可对此进行选择。
- 3) 然后将从模式控制器设为外部时钟模式 1（在 R16_TIM2_SMCFGR 寄存器中写入 SMS=111）。这样一来，定时器 2 的时钟将由定时器 1 周期性触发信号的上升沿（与定时器 1 的计数器上溢对应）提供。
- 4) 最后必须通过将这两个定时器的相应 CEN 位（R16_TIMx_CTLR1 寄存器）置 1 同时使能二者。

注意：如果选择定时器 1 的 OCx 信号作为触发输出 (MMS=1xx)，该信号的上升沿将用于驱动定时器 2

的计数器。

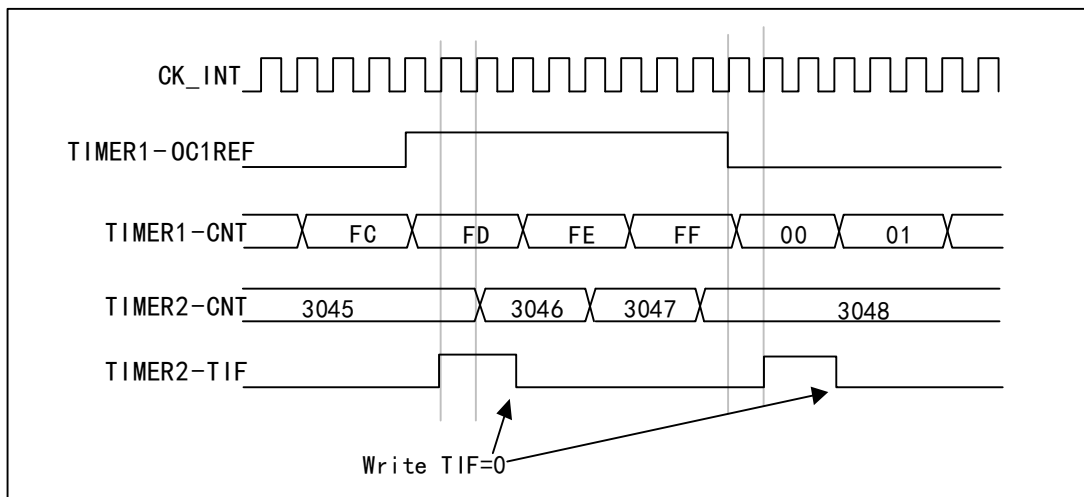
使用一个定时器使能另一个定时器

本例中通过定时器 1 的输出比较 1 来使能定时器 2。仅当定时器 1 的 OC1REF 为高电平时，定时器 2 才根据分频后的内部时钟进行计数。两个计数器的时钟频率都基于 CK_INT 通过预分频器执行 3 分频 ($f_{CK_CNT}=f_{CK_INT}/3$)。

- 1) 将定时器 1 配置为主模式，发送其输出比较 1 参考信号 (OC1REF) 作为触发输出 (R16_TIM1_CR2 寄存器中的 MMS=100)。
- 2) 配置定时器 1 的 OC1REF 波形 (R16_TIM1_CCMR1 寄存器)。
- 3) 配置定时器 2 以接收来自定时器 1 的输入触发 (R16_TIM2_SMCFGR 寄存器中的 TS=000)。
- 4) 将定时器 2 配置为门控模式 (R16_TIM2_SMCFGR 寄存器中的 SMS=101)。
- 5) 通过向 CEN 位 (R16_TIM2_CTLR1 寄存器) 写入 “1” 使能定时器 2。
- 6) 通过向 CEN 位 (R16_TIM1_CTLR1 寄存器) 写入 “1” 启动定时器 1。

注：计数器 2 的时钟与计数器 1 不同步，此模式仅影响定时器 2 的计数器使能信号。

图 14-11 使用定时器 1 的 OC1REF 对定时器 2 实施门控控制

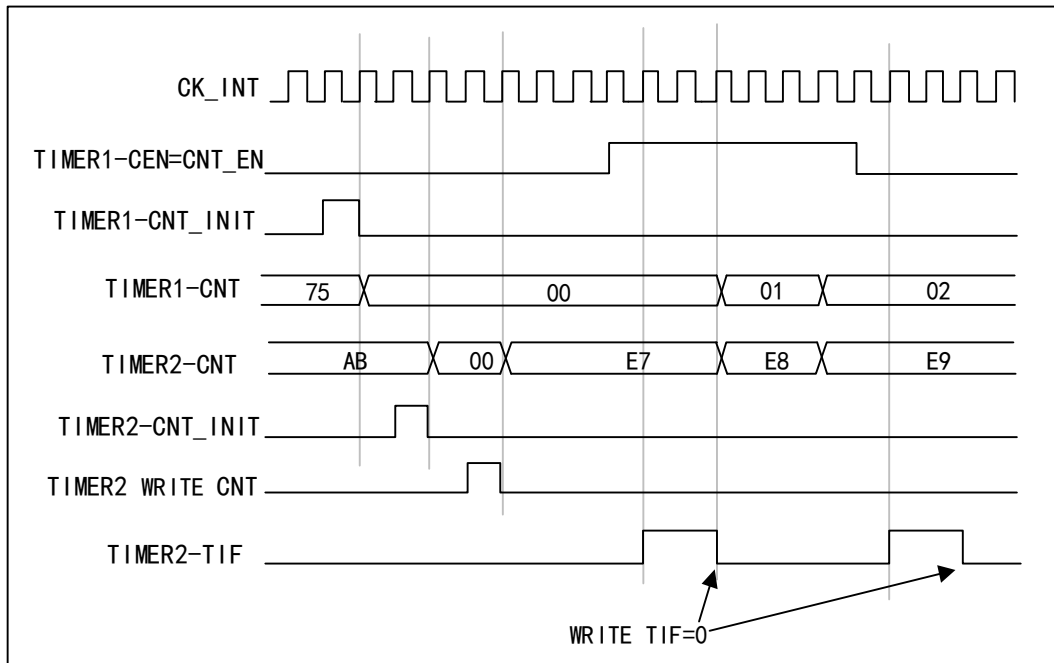


定时器 2 的计数器和预分频器在启动前未进行初始化。因此从各自的当前值开始计数。启动定时器 1 之前，通过复位这两个定时器可以从指定值开始计数。这样便可以在定时器计数器中写入所需的任意值。两个定时器都可通过软件使用 R16_TIMx_SWEVGR 寄存器中的 UG 位轻松复位。

在下一示例中，定时器 1 与定时器 2 同步。定时器 1 为主模式，从 0 开始计数。定时器 2 为从模式，从 0xE7 开始计数。两个定时器的预分频比相同。在 R16_TIM1_CTLR1 寄存器中通过向 CEN 位写入 “0” 来禁止定时器 1 时，定时器 2 将停止：

- 1) 将定时器 1 配置为主模式，发送其输出比较 1 参考信号 (OC1REF) 作为触发输出 (R16_TIM1_CTLR2 寄存器中的 MMS=100)。
- 2) 配置定时器 1 的 OC1REF 波形 (R16_TIM1_CHCTLR1 寄存器)。
- 3) 配置定时器 2 以接收来自定时器 1 的输入触发 (R16_TIM2_SMCFGR 寄存器中的 TS=000)。
- 4) 将定时器 2 配置为门控模式 (R16_TIM2_SMCFGR 寄存器中的 SMS=101)。
- 5) 通过向 UG 位 (R16_TIM1_SWEVGR 寄存器) 写入 “1” 复位定时器 1。
- 6) 通过向 UG 位 (R16_TIM2_SWEVGR 寄存器) 写入 “1” 复位定时器 2。
- 7) 通过在定时器 2 的计数器 (R16_TIM2_CNTL) 中写入 “0xE7” 使定时器 2 初始化为 0xE7。
- 8) 通过向 CEN 位 (R16_TIM2_CTLR1 寄存器) 写入 “1” 使能定时器 2。
- 9) 通过向 CEN 位 (R16_TIM1_CTLR1 寄存器) 写入 “1” 启动定时器 1。
- 10) 通过向 CEN 位 (R16_TIM1_CTLR1 寄存器) 写入 “0” 停止定时器 1。

图 14-12 使用定时器 1 的使能信号对定时器 2 实施门控控制

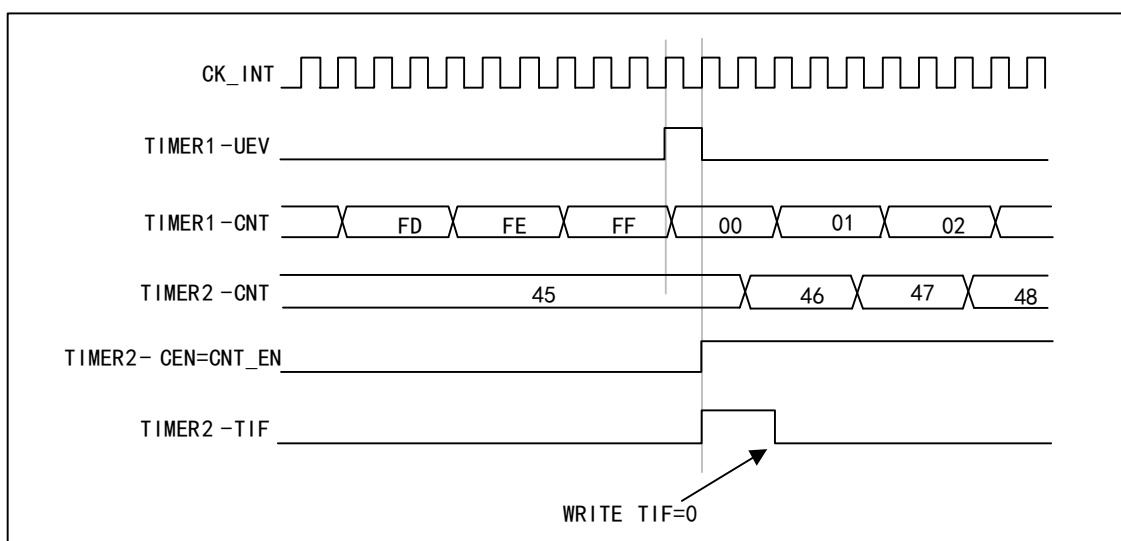


使用一个定时器启动另一个定时器

本例中使用定时器 1 的更新事件使能定时器 2。只要定时器 1 生成更新事件，定时器 2 便根据分频后的内部时钟从当前值（可以为 0）开始计数。定时器 2 收到触发信号时，其 CEN 位自动置 1，并且计数器开始计数，直到向 R16_TIM2_CTLR1 寄存器的 CEN 位写入“0”后停止计数。两个计数器的时钟频率都基于 CK_INT 通过预分频器执行 3 分频 ($f_{CK_CNT} = f_{CK_INT}/3$)。

- 1) 将定时器 1 配置为主模式，发送其更新事件 (UEV) 作为触发输出 (R16_TIM1_CTLR2 寄存器中的 MMS=010)。
- 2) 配置定时器 1 的周期 (R16_TIM1_ATRLR 寄存器)。
- 3) 配置定时器 2 以接收来自定时器 1 的输入触发 (R16_TIM2_SMCFG 寄存器中的 TS=000)。
- 4) 将定时器 2 配置为触发模式 (R16_TIM2_SMCFG 寄存器中的 SMS=110)。
- 5) 通过向 CEN 位 (R16_TIM1_CTLR1 寄存器) 写入“1”启动定时器 1。

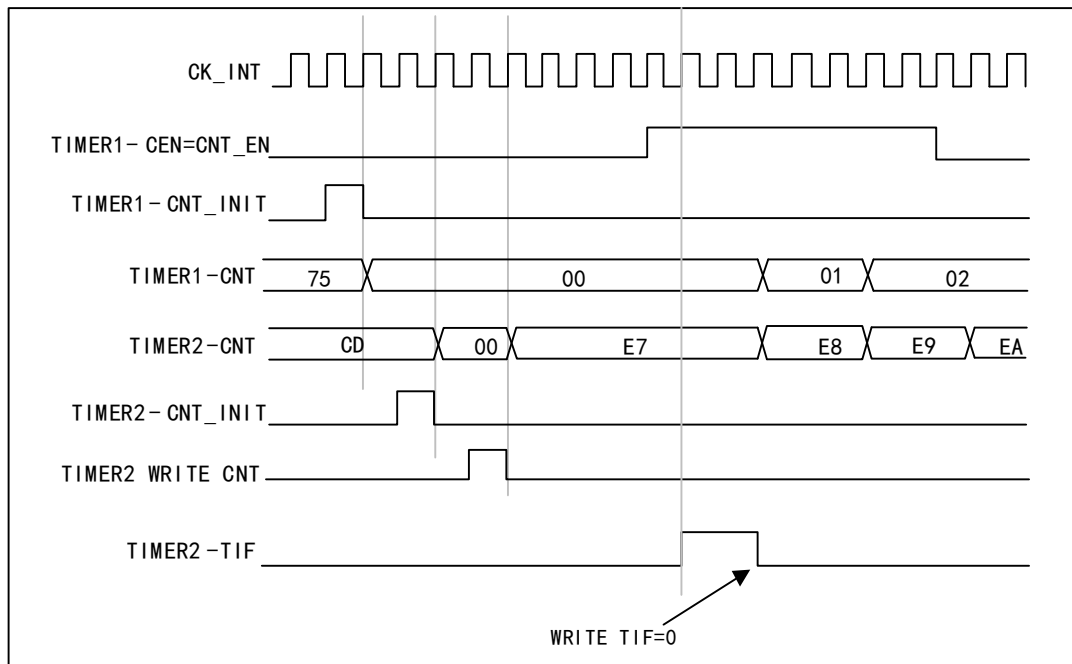
图 14-13 使用定时器 1 的更新事件触发定时器 2



如上述示例所示，用户可以在开始计数之前初始化两个计数器。图 14-14 显示了与图 14-13 具

有相同配置，只不过处于触发模式（R16_TIM2_SMCFG 寄存器中的 SMS=110）而非门控模式的计数行为。

图 14-14 使用定时器 1 的使能信号触发定时器 2



将一个定时器用作另一个定时器的预分频器

例如，可以将定时器 1 配置为定时器 2 的预分频器。为此：

- 1) 将定时器 1 配置为主模式，发送其更新事件 (UEV) 作为触发输出（R16_TIM1_CTLR2 寄存器中的 MMS=010）。这样便会在计数器每次发生上溢时输出一个周期性信号。
- 2) 配置定时器 1 的周期（R16_TIM1_ATRLR 寄存器）。
- 3) 配置定时器 2 以接收来自定时器 1 的输入触发（R16_TIM2_SMCFG 寄存器中的 TS=000）。
- 4) 将定时器 2 配置为外部时钟模式（R16_TIM2_SMCFG 寄存器中的 SMS=111）。
- 5) 通过向 CEN 位（R16_TIM2_CTLR1 寄存器）写入“1”启动定时器 2。
- 6) 通过向 CEN 位（R16_TIM1_CTLR1 寄存器）写入“1”启动定时器 1。

使用一个外部触发同步的启动 2 个定时器

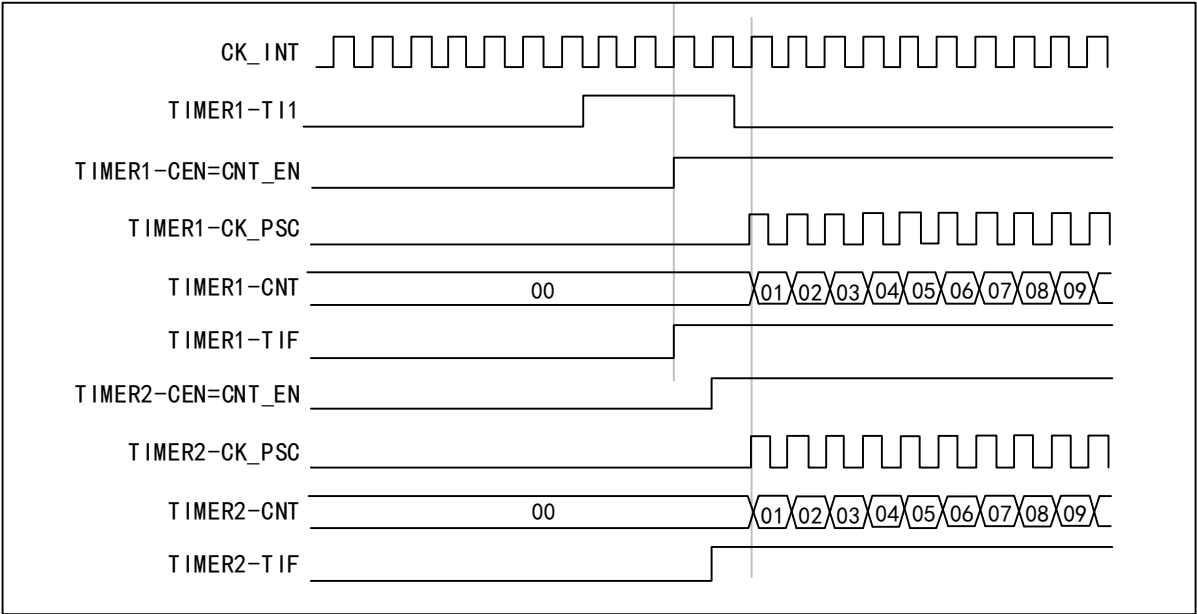
本例中，定时器 1 的 TI1 输入出现上升沿时使能定时器 1，使能定时器 1 的同时使能定时器 2。要确保两个计数器对齐，定时器 1 必须配置为主/从模式（对应的 TI1 为从，对应定时器 2 为主）：

- 1) 将定时器 1 配置为主模式，发送其使能信号作为触发输出（R16_TIM1_CTLR2 寄存器中的 MMS=001）。
- 2) 将定时器 1 配置为从模式以接收来自 TI1 的输入触发（R16_TIM1_SMCFG 寄存器中的 TS=100）。
- 3) 将定时器 1 配置为触发模式（R16_TIM1_SMCFG 寄存器中的 SMS=110）。
- 4) 通过写入 MSM=1（R16_TIMx_SMCR 寄存器）将定时器 1 配置为主/从模式。
- 5) 配置定时器 2 以接收来自定时器 1 的输入触发（R16_TIM2_SMCFG 寄存器中的 TS=000）。
- 6) 将定时器 2 配置为触发模式（R16_TIM2_SMCFG 寄存器中的 SMS=110）。

当 TI1（定时器 1）出现上升沿时，两个计数器开始根据内部时钟同步计数，并且两个 TIF 标志都置 1。

注：本例中，两个定时器都在启动之前进行了初始化（通过将各自的 UG 位置 1）。两个计数器都从 0 开始计数，但可以通过对任意一个计数器寄存器（R16_TIMx_CNT）进行写操作，在二者之间轻松插入一个偏移量。可注意到主/从模式在定时器 1 的 CNT_EN 与 CK_PSC 之间产生了延迟。

图 14-15 使用定时器 1 的 TI1 输入触发定时器 1 和定时器 2



定时器能够输出时钟脉冲（TRGO），也能接收其他定时器的输入（ITRx）。不同的定时器的 ITRx 的来源（别的定时器的 TRGO）是不一样的。定时器内部触发连接如表 14-2 所示。

表 14-2 TIMx 内部触发连接

从定时器	ITR0 (TS=000)	ITR1 (TS=001)	ITR2 (TS=010)	ITR3 (TS=011)
TIM1	-	TIM2	TIM3	TIM4

14.3.13 调试模式

当系统进入调试模式时，定时器根据 DBG 模块的设置继续运转或停止。

14.4 寄存器描述

表 14-3 TIM1 相关寄存器列表

名称	访问地址	描述	复位值
R16_TIM1_CTLR1	0x40012C00	控制寄存器 1	0x0000
R16_TIM1_CTLR2	0x40012C04	控制寄存器 2	0x0000
R16_TIM1_SMCFR	0x40012C08	从模式控制寄存器	0x0000
R16_TIM1_DMAINTENR	0x40012C0C	DMA/中断使能寄存器	0x0000
R16_TIM1_INTFR	0x40012C10	中断状态寄存器	0x0000
R16_TIM1_SWEVGR	0x40012C14	事件产生寄存器	0x0000
R16_TIM1_CHCTLR1	0x40012C18	比较/捕获控制寄存器 1	0x0000
R16_TIM1_CHCTLR2	0x40012C1C	比较/捕获控制寄存器 2	0x0000
R16_TIM1_CCER	0x40012C20	比较/捕获使能寄存器	0x0000
R16_TIM1_CNT	0x40012C24	计数器	0x0000
R16_TIM1_PSC	0x40012C28	计数时钟预分频器	0x0000
R16_TIM1_ATRLR	0x40012C2C	自动重装值寄存器	0xFFFF
R16_TIM1_RPTCR	0x40012C30	重复计数值寄存器	0x0000
R32_TIM1_CH1CVR	0x40012C34	比较/捕获寄存器 1	0x00000000
R32_TIM1_CH2CVR	0x40012C38	比较/捕获寄存器 2	0x00000000

R32_TIM1_CH3CVR	0x40012C3C	比较/捕获寄存器 3	0x00000000
R32_TIM1_CH4CVR	0x40012C40	比较/捕获寄存器 4	0x00000000
R16_TIM1_BDTR	0x40012C44	刹车和死区寄存器	0x0000
R16_TIM1_DMACHFR	0x40012C48	DMA 控制寄存器	0x0000
R32_TIM1_DMAADR	0x40012C4C	连续模式的 DMA 地址寄存器	0x00000000

14.4.1 控制寄存器 1 (TIM1_CTLR1)

偏移地址: 0x00

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMR_CAP_LVL_EN	TMR_CAP_OV_EN	Reserved	BKSEL	Reserved	CKD[1:0]	ARPE	CMS[1:0]	DIR	OPM	URS	UDIS	CEN			

位	名称	访问	描述	复位值
15	TMR_CAP_LVL_EN	RW	双沿捕获模式下, 捕获电平指示使能: 0: 关闭指示功能 1: 使能指示功能。 注: 使能后, CHxCVR 的[16]指示捕获值对应的电平。	0
14	TMR_CAP_OV_EN	RW	捕获值模式配置: 0: 捕获值为实际计数器的值 1: 当捕获前产生计数器溢出时, CHxCVR 值为 0xFFFF。	0
13	Reserved	RO	保留。	0
12	BKSEL	RW	TIM1 刹车输入源选择: 0: 刹车来自 IO 或者 OPA; 1: 刹车信号来自比较器 (若 CMP3 使能, 则刹车来自于 CMP3 的输出; 若 CMP3 未使能且 CMP2 使能, 则刹车来自 CMP2 的输出; 若 CMP3 和 CMP2 未使能且 CMP1 使能, 则刹车来自于 CMP1 的输出)	0
[11:10]	Reserved	RO	保留。	0
[9:8]	CKD[1:0]	RW	这 2 位定义在定时器时钟(CK_INT)频率、死区时间和由死区发生器与数字滤波器(ETR, T1x)所用的采样时钟之间的分频比例: 00: Tdts=Tck_int 01: Tdts = 2 x Tck_int 10: Tdts = 4 x Tck_int 11: 保留。	00b
7	ARPE	RW	自动重装预装使能位: 0: 禁止自动重装值寄存器 (ATRLR); 1: 使能自动重装值寄存器 (ATRLR)。	0
[6:5]	CMS[1:0]	RW	中央对齐模式选择: 00: 边沿对齐模式。计数器依据方向位(DIR)向上或向下计数。 01: 中央对齐模式 1。计数器交替地向上和向下计数。配置为输出的通道(CHCTLRx 寄存器中 CCxS=00)的输出比较中断标志位, 只在计数器向下计数时被	00b

			<p>设置。</p> <p>10: 中央对齐模式 2。计数器交替地向上和向下计数。配置为输出的通道 (CHCTLRx 寄存器中 CCxS=00) 的输出比较中断标志位, 只在计数器向上计数时被设置。</p> <p>11: 中央对齐模式 3。计数器交替地向上和向下计数。配置为输出的通道 (CHCTLRx 寄存器中 CCxS=00) 的输出比较中断标志位, 在计数器向上和向下计数时均被设置。</p> <p>注: 在计数器使能时 (CEN=1), 不允许从边沿对齐模式转换到中央对齐模式。</p>	
4	DIR	RW	<p>计数器方向:</p> <p>0: 计数器的计数模式为增计数;</p> <p>1: 计数器的计数模式为减计数。</p> <p>注: 当计数器配置为中央对齐模式或编码器模式时, 该位无效。</p>	0
3	OPM	RW	<p>单脉冲模式:</p> <p>0: 在发生下一次更新事件时, 计数器不停止;</p> <p>1: 在发生下一次更新事件时, 计数器停止 (清除 CEN 位)。</p>	0
2	URS	RW	<p>更新请求源, 软件通过该位选择 UEV 事件的源。</p> <p>0: 如果使能了更新中断或 DMA 请求, 则下述任一事件产生更新中断或 DMA 请求。</p> <ul style="list-style-type: none"> -计数器溢出/下溢 -设置 UG 位 -从模式控制器产生的更新 <p>1: 如果使能了更新中断或 DMA 请求, 则只有计数器溢出/下溢才产生更新中断或 DMA 请求。</p>	0
1	UDIS	RW	<p>禁止更新, 软件通过该位允许/禁止 UEV 事件的产生。</p> <p>0: 允许 UEV。更新 (UEV) 事件由下述任一事件产生:</p> <ul style="list-style-type: none"> -计数器溢出/下溢 -设置 UG 位 -从模式控制器产生的更新具有缓存的寄存器被装入它们的预装载值。 <p>1: 禁止 UEV。不产生更新事件, 各寄存器 (ARR、PSC、CCRx) 保持它们的值。如果设置了 UG 位或从模式控制器发出了一个硬件复位, 则计数器和预分频器被重新初始化。</p>	0
0	CEN	RW	<p>使能计数器。</p> <p>0: 禁止计数器;</p> <p>1: 使能计数器。</p> <p>注: 在软件设置了 CEN 位后, 外部时钟、门控模式和编码器模式才能工作。触发模式可以自动地通过硬件设置 CEN 位。</p>	0

14.4.2 控制寄存器 2 (TIM1_CTLR2)

偏移地址: 0x04

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Reserved	CCPC

位	名称	访问	描述	复位值
15	Reserved	RO	保留。	0
14	OIS4	RW	输出空闲状态 4: 1: 当 MOE=0 时, 如果实施了 OC4N, 则死区后 OC4=1; 0: 当 MOE=0 时, 如果实施了 OC4N, 则死区后 OC4=0。 注: 已经设置了 LOCK (TIMx_BDTR 寄存器) 级别 1、2 或 3 后, 该位不能被修改。	0
13	OIS3N	RW	输出空闲状态 3: 1: 当 MOE=0 时, 死区后 OC3N=1; 0: 当 MOE=0 时, 死区后 OC3N=0。 注: 已经设置了 LOCK (TIMx_BDTR 寄存器) 级别 1、2 或 3 后, 该位不能被修改。	0
12	OIS3	RW	输出空闲状态 3, 参见 OIS4。	0
11	OIS2N	RW	输出空闲状态 2, 参见 OIS3N。	0
10	OIS2	RW	输出空闲状态 2, 参见 OIS4。	0
9	OIS1N	RW	输出空闲状态 1, 参见 OIS3N。	0
8	OIS1	RW	输出空闲状态 1, 参见 OIS4。	0
7	TI1S	RW	TI1 选择: 1: TIMx_CH1、TIMx_CH2 和 TIMx_CH3 引脚经异或后连到 TI1 输入; 0: TIMx_CH1 引脚直连到 TI1 输入。	0
[6:4]	MMS[2:0]	RW	主模式选择: 这 3 位用于选择在主模式下送到从定时器的同步信息 (TRGO)。 可能的组合如下: 000: 复位 - TIMx_EGR 寄存器的 UG 位被用于作为触发输出 (TRGO)。如果是触发输入产生的复位 (从模式控制器处于复位模式), 则 TRGO 上的信号相对实际的复位会有一个延迟; 001: 使能 - 计数器使能信号 CNT_EN 被用于作为触发输出 (TRGO)。有时需要在同一时间启动多个定时器或控制在一段时间内使能从定时器。计数器使能信号是通过 CEN 控制位和门控模式下的触发输入信号的逻辑或产生。当计数器使能信号受控于触发输入时, TRGO 上会有一个延迟, 除非选择了主/从模式 (见 TIMx_SMCR 寄存器中 MSM 位的描述); 010: 更新 - 更新事件被选为触发输入 (TRGO)。例如, 一个主定时器的时钟可以被用作一个从定时器的预分频器; 011: 比较脉冲 - 在发生一次捕获或一次比较成功时, 当要设置 CC1IF 标志时 (即使它已经为高), 触	000b

			发输出送出一个正脉冲 (TRGO) ; 100: 比较 - OC1REF 信号被用于作为触发输出 (TRGO) ; 101: 比较 - OC2REF 信号被用于作为触发输出 (TRGO) ; 110: 比较 - OC3REF 信号被用于作为触发输出 (TRGO) ; 111: 比较 - OC4REF 信号被用于作为触发输出 (TRGO) 。	
3	CCDS	RW	捕获比较的 DMA 选择。 1: 当发生更新事件时, 送出 CHxCVR 的 DMA 请求; 0: 当发生 CHxCVR 时, 产生 CHxCVR 的 DMA 请求。	0
2	CCUS	RW	比较捕获控制更新选择位。 1: 如果 CCPC 置位, 可以通过设置 COM 位或 TRGI 上的一个上升沿更新它们; 0: 如果 CCPC 置位, 只能通过设置 COM 位更新它们。 <i>注: 该位只对具有互补输出的通道起作用。</i>	0
1	Reserved	RO	保留。	0
0	CCPC	RW	比较捕获预装载控制位。 1: CCxE, CCxNE 和 OCxM 位是预装载的, 设置该位后, 它们只在设置了 COM 位后被更新; 0: CCxE, CCxNE 和 OCxM 位不是预装载的。 <i>注: 该位只对具有互补输出的通道起作用。</i>	0

14.4.3 从模式控制寄存器 (TIM1_SMCFR)

偏移地址: 0x08

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	ECE	ETPS[1:0]	ETF[3:0]			MSM	TS[2:0]		Reserved	SMS[2:0]					

位	名称	访问	描述	复位值
15	ETP	RO	ETR 触发极性选择, 该位选择是直接输入 ETR 还是输入 ETR 的反相。 1: 将 ETR 反相, 低电平或下降沿有效; 0: ETR, 高电平或上升沿有效。	0
14	ECE	RW	外部时钟模式 2 启用选择: 1: 使能外部时钟模式 2; 0: 禁用外部时钟模式 2。 <i>注 1: 从模式可以与外部时钟模式 2 同时使用: 复位模式, 门控模式和触发模式; 但是, 这时 TRGI 不能连到 ETRF (TS 位不能是 '111')。</i> <i>注 2: 外部时钟模式 1 和外部时钟模式 2 同时被使能时, 外部时钟的输入是 ETRF。</i>	0
[13:12]	ETPS[1:0]	RW	外部触发信号 (ETRP) 分频, 这个信号频率最大不能超过 TIMxCLK 频率的 1/4, 可以通过这个域来降频: 00: 关闭预分频;	00b

			01: ETRP 频率除以 2; 10: ETRP 频率除以 4; 11: ETRP 频率除以 8。	
[11:8]	ETF[3:0]	RW	外部触发滤波, 实际上, 数字滤波器是一个事件计数器, 它使用一定的采样的频率, 记录到 N 个事件后会产生一个输出的跳变。 0000: 无滤波器, 以 Fdts 采样; 0001: 采样频率 $F_{sampling}=F_{ck_int}$, N=2; 0010: 采样频率 $F_{sampling}=F_{ck_int}$, N=4; 0011: 采样频率 $F_{sampling}=F_{ck_int}$, N=8; 0100: 采样频率 $F_{sampling}=F_{dts}/2$, N=6; 0101: 采样频率 $F_{sampling}=F_{dts}/2$, N=8; 0110: 采样频率 $F_{sampling}=F_{dts}/4$, N=6; 0111: 采样频率 $F_{sampling}=F_{dts}/4$, N=8; 1000: 采样频率 $F_{sampling}=F_{dts}/8$, N=6; 1001: 采样频率 $F_{sampling}=F_{dts}/8$, N=8; 1010: 采样频率 $F_{sampling}=F_{dts}/16$, N=5; 1011: 采样频率 $F_{sampling}=F_{dts}/16$, N=6; 1100: 采样频率 $F_{sampling}=F_{dts}/16$, N=8; 1101: 采样频率 $F_{sampling}=F_{dts}/32$, N=5; 1110: 采样频率 $F_{sampling}=F_{dts}/32$, N=6; 1111: 采样频率 $F_{sampling}=F_{dts}/32$, N=8。	0000b
7	MSM	RW	主/从模式选择: 1: 触发输入 (TRGI) 上的事件被延迟了, 以允许在当前定时器 (通过 TRGO) 与它的从定时器间的完美同步。这对要求把几个定时器同步到一个单一的外部事件时是非常有用的; 0: 不发挥作用。	0
[6:4]	TS[2:0]	RW	触发选择域, 这 3 位选择用于同步计数器的触发输入源: 000: 内部触发 0 (ITR0); 001: 内部触发 1 (ITR1); 010: 内部触发 2 (ITR2); 011: 内部触发 3 (ITR3); 100: TI1 的边沿检测器 (TI1F_ED); 101: 滤波后的定时器输入 1 (TI1FP1); 110: 滤波后的定时器输入 2 (TI2FP2); 111: 外部触发输入 (ETRF); 以上只有在 SMS 为 0 时改变。 注: 具体见表 14-2。	000b
3	Reserved	RO	保留。	0
[2:0]	SMS[2:0]	RW	输入模式选择域。选择核心计数器的时钟和触发模式。 000: 由内部时钟 CK_INT 驱动; 001: 编码器模式 1, 根据 TI1FP1 的电平, 核心计数器在 TI2FP2 的边沿增减计数;	000b

		<p>010: 编码器模式 2, 根据 TI2FP2 的电平, 核心计数器在 TI1FP1 的边沿增减计数;</p> <p>011: 编码器模式 3, 根据另一个信号的输入电平, 核心计数器在 TI1FP1 和 TI2FP2 的边沿增减计数;</p> <p>100: 复位模式, 触发输入 (TRGI) 的上升沿将初始化计数器, 并且产生一个更新寄存器的信号;</p> <p>101: 门控模式, 当触发输入 (TRGI) 为高时, 计数器的时钟开启; 在触发输入变为低, 计数器停止, 计数器的启停都是受控的;</p> <p>110: 触发模式, 计数器在触发输入 TRGI 的上升沿启动, 只有计数器的启动是受控的;</p> <p>111: 外部时钟模式 1, 选中的触发输入 (TRGI) 的上升沿驱动计数器。</p>	
--	--	--	--

14.4.4 DMA/中断使能寄存器 (TIM1_DMAINTENR)

偏移地址: 0x0C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UIE

位	名称	访问	描述	复位值
15	Reserved	RO	保留。	0
14	TDE	RW	触发 DMA 请求使能位。 1: 允许触发 DMA 请求; 0: 禁止触发 DMA 请求。	0
13	COMDE	RW	COM 的 DMA 请求使能位。 1: 允许 COM 的 DMA 请求; 0: 禁止 COM 的 DMA 请求。	0
12	CC4DE	RW	比较捕获通道 4 的 DMA 请求使能位。 1: 允许比较捕获通道 4 的 DMA 请求; 0: 禁止比较捕获通道 4 的 DMA 请求。	0
11	CC3DE	RW	比较捕获通道 3 的 DMA 请求使能位。 1: 允许比较捕获通道 3 的 DMA 请求; 0: 禁止比较捕获通道 3 的 DMA 请求。	0
10	CC2DE	RW	比较捕获通道 2 的 DMA 请求使能位。 1: 允许比较捕获通道 2 的 DMA 请求; 0: 禁止比较捕获通道 2 的 DMA 请求。	0
9	CC1DE	RW	比较捕获通道 1 的 DMA 请求使能位。 1: 允许比较捕获通道 1 的 DMA 请求; 0: 禁止比较捕获通道 1 的 DMA 请求。	0
8	UDE	RW	更新的 DMA 请求使能位。 1: 允许更新的 DMA 请求; 0: 禁止更新的 DMA 请求。	0
7	BIE	RW	刹车中断使能位。 1: 允许刹车中断; 0: 禁止刹车中断。	0

6	TIE	RW	触发中断使能位。 1: 使能触发中断; 0: 禁止触发中断。	0
5	COMIE	RW	COM 中断允许位。 1: 允许 COM 中断; 0: 禁止 COM 中断。	0
4	CC4IE	RW	比较捕获通道 4 中断使能位。 1: 允许比较捕获通道 4 中断; 0: 禁止比较捕获通道 4 中断。	0
3	CC3IE	RW	比较捕获通道 3 中断使能位。 1: 允许比较捕获通道 3 中断; 0: 禁止比较捕获通道 3 中断。	0
2	CC2IE	RW	比较捕获通道 2 中断使能位。 1: 允许比较捕获通道 2 中断; 0: 禁止比较捕获通道 2 中断。	0
1	CC1IE	RW	比较捕获通道 1 中断使能位。 1: 允许比较捕获通道 1 中断; 0: 禁止比较捕获通道 1 中断。	0
0	UIE	RW	更新中断使能位。 1: 允许更新中断; 0: 禁止更新中断。	0

14.4.5 中断状态寄存器 (TIM1_INTFR)

偏移地址: 0x10

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	CC40F	CC30F	CC20F	CC10F	Reserved	BIF	TIF	COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UIF		

位	名称	访问	描述	复位值
[15:13]	Reserved	RO	保留。	0
12	CC40F	RW0	比较捕获通道 4 重复捕获标志位。	0
11	CC30F	RW0	比较捕获通道 3 重复捕获标志位。	0
10	CC20F	RW0	比较捕获通道 2 重复捕获标志位。	0
9	CC10F	RW0	比较捕获通道 1 重复捕获标志位, 仅用于比较捕获通道被配置为输入捕获模式时。该标记由硬件置位, 软件写 0 可清除此位。 1: 计数器的值被捕获到捕获比较寄存器时, CC1IF 的状态已经被置位; 0: 无重复捕获产生。	0
8	Reserved	RO	保留。	0
7	BIF	RW0	刹车中断标志位, 一旦刹车输入有效, 由硬件对该位置位, 可由软件清零。 1: 刹车引脚输入上检测到设定的有效电平; 0: 无刹车事件产生。	0
6	TIF	RW0	触发器中断标志位, 当发生触发事件时由硬件对该位置位, 由软件清零。触发事件包括从除门控模式	0

			外的其它模式时，在 TRGI 输入端检测到有效边沿，或门控模式下的任一边沿。 1：触发器事件产生； 0：无触发器事件产生。	
5	COMIF	RW0	COM 中断标志位，一旦产生 COM 事件，该位由硬件置位，由软件清零。COM 事件包括 CCxE、CCxNE、OCxM 被更新。 1：COM 事件产生； 0：无 COM 事件产生。	0
4	CC4IF	RW0	比较捕获通道 4 中断标志位。	0
3	CC3IF	RW0	比较捕获通道 3 中断标志位。	0
2	CC2IF	RW0	比较捕获通道 2 中断标志位。	0
1	CC1IF	RW0	比较捕获通道 1 中断标志位。 如果比较捕获通道配置为输出模式： 当计数器值与比较值匹配时该位由硬件置位，但在中心对称模式下除外。该位由软件清零。 1：核心计数器的值与比较捕获寄存器 1 的值匹配； 0：无匹配发生。 如果比较捕获通道 1 配置为输入模式： 当捕获事件发生时该位由硬件置位，它由软件清零或通过读比较捕获寄存器清零。 1：计数器值已被捕获比较捕获寄存器 1； 0：无输入捕获产生。	0
0	UIF	RW0	更新中断标志位，当产生更新事件时该位由硬件置位，由软件清零。 1：更新中断产生； 0：无更新事件产生。 以下情形会产生更新事件： 若 UDIS=0，当重复计数器数值上溢或下溢时； 若 URS=0、UDIS=0，当置 UG 位时，或当通过软件对计数器核心计数器重新初始化时； 若 URS=0、UDIS=0，当计数器 CNT 被触发事件重新初始化时；	0

14.4.6 事件产生寄存器 (TIM1_SWEVGR)

偏移地址：0x14

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								BG	TG	COMG	CC4G	CC3G	CC2G	CC1G	UG

位	名称	访问	描述	复位值
[15:8]	Reserved	R0	保留。	0
7	BG	WO	刹车事件产生位，此位由软件置位和清零，用来产生一个刹车事件。 1：产生一个刹车事件。此时 MOE=0、BIF=1，若使能对应的中断和 DMA，则产生相应的中断和 DMA；	0

			0: 无动作。	
6	TG	WO	触发事件产生位, 该位由软件置位, 硬件清零, 用于产生一个触发事件。 1: 产生一个触发事件, TIF 被置位, 若使能对应的中断和 DMA, 则产生相应的中断和 DMA; 0: 无动作。	0
5	COMG	WO	比较捕获控制更新产生位。产生比较捕获控制更新事件。该位由软件置位, 由硬件自动清零。 1: 当 CCPC=1, 允许更新 CCxE、CCxNE、OCxM 位; 0: 无动作。 注: 该位只对拥有互补输出的通道 (通道 1, 2, 3) 有效。	0
4	CC4G	WO	比较捕获事件产生位 4。产生比较捕获事件 4。	0
3	CC3G	WO	比较捕获事件产生位 3。产生比较捕获事件 3。	0
2	CC2G	WO	比较捕获事件产生位 2。产生比较捕获事件 2。	0
1	CC1G	WO	比较捕获事件产生位 1, 产生比较捕获事件 1。 该位由软件置位, 由硬件清零。用于产生一个比较捕获事件。 1: 在比较捕获通道 1 上产生一个比较捕获事件: 若比较捕获通道 1 配置为输出: 置 CC1IF 位。若使能对应的中断和 DMA, 则产生相应的中断和 DMA; 若比较捕获通道 1 配置为输入: 当前核心计数器的值被捕获至比较捕获寄存器 1; 置 CC1IF 位, 若使能了对应的中断和 DMA, 则产生相应的中断和 DMA。若 CC1IF 已经置位, 则置 CC1OF 位。 0: 无动作。	0
0	UG	WO	更新事件产生位, 产生更新事件。该位由软件置位, 由硬件自动清零。 1: 初始化计数器, 并产生一个更新事件; 0: 无动作。 注: 预分频器的计数器也被清零, 但是预分频系数不变。若在中心对称模式下或增计数模式下则核心计数器被清零; 若减计数模式下则核心计数器取重装值寄存器的值。	0

14.4.7 比较/捕获控制寄存器 1 (TIM1_CHCTLR1)

偏移地址: 0x18

通道可用于输入 (捕获模式) 或输出 (比较模式), 通道的方向由相应的 CCxS 位定义。该寄存器其它位的作用在输入和输出模式下不同。OCxx 描述了通道在输出模式下的功能, ICxx 描述了通道在输入模式下的功能。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
OC2CE		OC2M[2:0]			OC2PE	OC2FE	GC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	GC1S[1:0]	
IC2F[3:0]				IC2PSC[1:0]		IC1F[3:0]			IC1PSC[1:0]							

比较模式（引脚方向为输出）：

位	名称	访问	描述	复位值
15	OC2CE	RW	比较捕获通道 2 清零使能位。 1：一旦检测到 ETRF 输入高电平，清除 OC2REF 位零； 0：OC2REF 不受 ETRF 输入的影响。	0
[14:12]	OC2M[2:0]	RW	比较捕获通道 2 模式设置域。 该 3 位定义了输出参考信号 OC2REF 的动作，而 OC2REF 决定了 OC2、OC2N 的值。OC2REF 是高电平有效，而 OC2 和 OC2N 的有效电平取决于 CC2P、CC2NP 位。 000：冻结。比较捕获寄存器的值与核心计数器间的比较值对 OC1REF 不起作用； 001：强制设为有效电平。当核心计数器与比较捕获寄存器 1 的值相同时，强制 OC1REF 为高； 010：强制设为无效电平。当核心计数器的值与比较捕获寄存器 1 相同时，强制 OC1REF 为低； 011：翻转。当核心计数器与比较捕获寄存器 1 的值相同时，翻转 OC1REF 的电平。 100：强制为无效电平。强制 OC1REF 为低。 101：强制为有效电平。强制 OC1REF 为高。 110：PWM 模式 1：在向上计数时，一旦核心计数器大于比较捕获寄存器的值时通道 1 为无效电平，否则为有效电平；在向下计数时，一旦核心计数器大于比较捕获寄存器的值时通道 1 为有效电平，否则为无效电平。 111：PWM 模式 2：在向上计数时，一旦核心计数器大于比较捕获寄存器的值时，通道 1 为有效电平，否则为无效电平；在向下计数时，一旦核心计数器大于比较捕获寄存器的值时，通道 1 为无效电平，否则为有效电平 (OC1REF=1)。 注：一旦 LOCK 级别设为 3 并且 CC1S=00b 则该位不能被修改。在 PWM 模式 1 或 PWM 模式 2 中，只有当比较结果改变了或在输出比较模式中从冻结模式切换到 PWM 模式时，OC1REF 电平才改变。	000b
11	OC2PE	RW	比较捕获寄存器 2 预装载使能位。 1：开启比较捕获寄存器的预装载功能，读写操作仅对预装载寄存器操作，比较捕获寄存器 1 的预装载值在更新事件到来时被加载至当前影子寄存器中； 0：禁止比较捕获寄存器 2 的预装载功能，可随时写入比较捕获寄存器 2，并且新写入的数值立即起作用。 注：一旦 LOCK 级别设为 3 并且 CC1S=00，则该位不能被修改；仅仅在单脉冲模式下 (OPM=1) 可以在未确认预装载寄存器情况下使用 PWM 模式，否则其动作不确定。	0
10	OC2FE	RW	比较捕获通道 2 快速使能位，该位用于加快比较捕	0

			<p>获通道输出对触发输入事件的响应。</p> <p>1: 输入到触发器的有效沿的作用就像发生了一次比较匹配。因此, OC 被设置为比较电平而与比较结果无关。采样触发器的有效沿和比较捕获通道 2 输出间的延时被缩短为 3 个时钟周期;</p> <p>0: 根据计数器与比较捕获寄存器 2 的值, 比较捕获通道 2 正常操作, 即使触发器是打开的。当触发器的输入有一个有效沿时, 激活比较捕获通道 2 输出的最小延时为 5 个时钟周期。</p> <p>OC2FE 只在通道被配置成 PWM1 或 PWM2 模式时起作用。</p>	
[9:8]	CC2S[1:0]	RW	<p>比较捕获通道 2 输入选择域。</p> <p>00: 比较捕获通道 2 被配置为输出;</p> <p>01: 比较捕获通道 2 被配置为输入, IC2 映射在 TI2 上;</p> <p>10: 比较捕获通道 2 被配置为输入, IC2 映射在 TI1 上;</p> <p>11: 比较捕获通道 2 被配置为输入, IC2 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时(由 TS 位选择)。</p> <p><i>注: 比较捕获通道 2 仅在通道关闭时(CC2E 为零时)才是可写的。</i></p>	00b
7	OC1CE	RW	比较捕获通道 1 清零使能位。	0
[6:4]	OC1M[2:0]	RW	比较捕获通道 1 模式设置域。	0
3	OC1PE	RW	比较捕获寄存器 1 预装载使能位。	0
2	OC1FE	RW	比较捕获通道 1 快速使能位。	0
[1:0]	CC1S[1:0]	RW	比较捕获通道 1 输入选择域。	0

捕获模式（引脚方向为输入）：

位	名称	访问	描述	复位值
[15:12]	IC2F[3:0]	RW	<p>输入捕获滤波器 2 配置域, 这几位设置了 TI1 输入的采样频率及数字滤波器长度。数字滤波器由一个事件计数器组成, 它记录到 N 个事件后会产生一个输出的跳变。</p> <p>0000: 无滤波器, 以 fDTS 采样;</p> <p>1000: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/8$, $N=6$;</p> <p>0001: 采样频率 $F_{\text{sampling}} = F_{\text{ck_int}}$, $N=2$;</p> <p>1001: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/8$, $N=8$;</p> <p>0010: 采样频率 $F_{\text{sampling}} = F_{\text{ck_int}}$, $N=4$;</p> <p>1010: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/16$, $N=5$;</p> <p>0011: 采样频率 $F_{\text{sampling}} = f = F_{\text{ck_int}}$, $N=8$;</p> <p>1011: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/16$, $N=6$;</p> <p>0100: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/2$, $N=6$;</p> <p>1100: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/16$, $N=8$;</p> <p>0101: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/2$, $N=8$;</p> <p>1101: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/32$, $N=5$;</p>	0000b

			0110: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/4$, $N=6$; 1110: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/32$, $N=6$; 0111: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/4$, $N=8$; 1111: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/32$, $N=8$ 。	
[11:10]	IC2PSC[1:0]	RW	比较捕获通道 2 预分频配置域, 这 2 位定义了比较捕获通道 2 的预分频系数。一旦 CC1E=0, 则预分频器复位。 00: 无预分频器, 捕获输入口上检测到的每一个边沿都触发一次捕获; 01: 每 2 个事件触发一次捕获; 10: 每 4 个事件触发一次捕获; 11: 每 8 个事件触发一次捕获。	00b
[9:8]	CC2S[1:0]	RW	比较捕获通道 2 输入选择域, 这 2 位定义通道的方向(输入/输出), 及输入脚的选择。 00: 比较捕获通道 1 通道被配置为输出; 01: 比较捕获通道 1 通道被配置为输入, IC1 映射在 TI1 上; 10: 比较捕获通道 1 通道被配置为输入, IC1 映射在 TI2 上; 11: 比较捕获通道 1 通道被配置为输入, IC1 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时(由 TS 位选择)。 注: CC1S 仅在通道关闭时(CC1E 为 0)才是可写的。	00b
[7:4]	IC1F[3:0]	RW	输入捕获滤波器 1 配置域。	0
[3:2]	IC1PSC[1:0]	RW	比较捕获通道 1 预分频配置域。	0
[1:0]	CC1S[1:0]	RW	比较捕获通道 1 输入选择域。	0

14.4.8 比较/捕获控制寄存器 2 (TIM1_CHCTLR2)

偏移地址: 0x1C

通道可用于输入(捕获模式)或输出(比较模式), 通道的方向由相应的 CCxS 位定义。该寄存器其它位的作用在输入和输出模式下不同。OCxx 描述了通道在输出模式下的功能, ICxx 描述了通道在输入模式下的功能。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC4CE	OC4M[2:0]			OC4PE	OC4FE	GC4S[1:0]		OC3CE	OC3M[2:0]			OC3PE	OC3FE	GC3S[1:0]	
IC4F[3:0]				IC4PSC[1:0]				IC3F[3:0]			IC3PSC[1:0]				

比较模式(引脚方向为输出):

位	名称	访问	描述	复位值
15	OC4CE	RW	比较捕获通道 4 清零使能位。	0
[14:12]	OC4M[2:0]	RW	比较捕获通道 4 模式设置域。	0
11	OC4PE	RW	比较捕获寄存器 4 预装载使能位。	0
10	OC4FE	RW	比较捕获通道 4 快速使能位。	0
[9:8]	CC4S[1:0]	RW	比较捕获通道 4 输入选择域。	0
7	OC3CE	RW	比较捕获通道 3 清零使能位。	0

[6:4]	OC3M[2:0]	RW	比较捕获通道 3 模式设置域。	0
3	OC3PE	RW	比较捕获寄存器 3 预装载使能位。	0
2	OC3FE	RW	比较捕获通道 3 快速使能位。	0
[1:0]	CC3S[1:0]	RW	比较捕获通道 3 输入选择域。	0

捕获模式（引脚方向为输入）：

位	名称	访问	描述	复位值
[15:12]	IC4F[3:0]	RW	输入捕获滤波器 4 配置域。	0
[11:10]	IC4PSC[1:0]	RW	比较捕获通道 4 预分频配置域。	0
[9:8]	CC4S[1:0]	RW	比较捕获通道 4 输入选择域。	0
[7:4]	IC3F[3:0]	RW	输入捕获滤波器 3 配置域。	0
[3:2]	IC3PSC[1:0]	RW	比较捕获通道 3 预分频配置域。	0
[1:0]	CC3S[1:0]	RW	比较捕获通道 3 输入选择域。	0

14.4.9 比较/捕获使能寄存器 (TIM1_CCER)

偏移地址：0x20

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E	

位	名称	访问	描述	复位值
[15:14]	Reserved	RO	保留。	0
13	CC4P	RW	比较捕获通道 4 输出极性设置位。	0
12	CC4E	RW	比较捕获通道 4 输出使能位。	0
11	CC3NP	RW	比较捕获通道 3 互补输出极性设置位。	0
10	CC3NE	RW	比较捕获通道 3 互补输出使能位。	0
9	CC3P	RW	比较捕获通道 3 输出极性设置位。	0
8	CC3E	RW	比较捕获通道 3 输出使能位。	0
7	CC2NP	RW	比较捕获通道 2 互补输出极性设置位。	0
6	CC2NE	RW	比较捕获通道 2 互补输出使能位。	0
5	CC2P	RW	比较捕获通道 2 输出极性设置位。	0
4	CC2E	RW	比较捕获通道 2 输出使能位。	0
3	CC1NP	RW	比较捕获通道 1 互补输出极性设置位。	0
2	CC1NE	RW	比较捕获通道 1 互补输出使能位。	0
1	CC1P	RW	比较捕获通道 1 输出极性设置位。 CC1 通道配置为输出： 1：OC1 低电平有效； 0：OC1 高电平有效。 CC1 通道配置为输入： 该位选择是 IC1 还是 IC1 的反相信号作为触发或捕获信号。 1：反相：捕获发生在 IC1 的下降沿；当用作外部触发器时，IC1 反相。 0：不反相：捕获发生在 IC1 的上升沿；当用作外部触发器时，IC1 不反相。	0

			注:一旦 LOCK 级别 (TIMx_BDTR 寄存器中的 LOCK 位) 设为 3 或 2, 则该位不能被修改。	
0	CC1E	RW	比较捕获通道 1 输出使能位。 CC1 通道配置为输出: 1: 开启。OC1 信号输出到对应的输出引脚, 其输出电平依赖于 MOE、OSSI、OSSR、OIS1、OIS1N 和 CC1NE 位的值。 0: 关闭。OC1 禁止输出, 因此 OC1 的输出电平依赖于 MOE、OSSI、OSSR、OIS1、OIS1N 和 CC1NE 位的值。 CC1 通道配置为输入: 该位决定了计数器的值是否能捕获入 TIMx_CCR1 寄存器。 1: 捕获使能; 0: 捕获禁止。	0

14.4.10 高级定时器的计数器 (TIM1_CNT)

偏移地址: 0x24

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

CNT[15:0]

位	名称	访问	描述	复位值
[15:0]	CNT[15:0]	RW	定时器的计数器的实时值。	0

14.4.11 计数时钟预分频器 (TIM1_PSC)

偏移地址: 0x28

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

PSC[15:0]

位	名称	访问	描述	复位值
[15:0]	PSC[15:0]	RW	定时器的预分频器的分频系数; 计数器的时钟频率等于分频器的输入频率/(PSC+1)。	0

14.4.12 自动重装值寄存器 (TIM1_ATRLR)

偏移地址: 0x2C

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

ARR[15:0]

位	名称	访问	描述	复位值
[15:0]	ARR[15:0]	RW	此域的值将会被装入计数器, ATRLR 何时动作和更新见 14.2.3 章节; ATRLR 为空时, 计数器停止。	0xFFFF

14.4.13 重复计数值寄存器 (TIM1_RPTCR)

偏移地址: 0x30

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

Reserved	REP[7:0]
----------	----------

位	名称	访问	描述	复位值
[15:8]	Reserved	R0	保留。	0
[7:0]	REP[7:0]	RW	重复计数器的值。	0

14.4.14 比较/捕获寄存器 1 (TIM1_CH1CVR)

偏移地址: 0x34

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															LEVEL1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1[15: 0]															

位	名称	访问	描述	复位值
[31:17]	Reserved	R0	保留。	0
16	LEVEL1	R0	捕获值对应的电平指示 bit	0
[15:0]	CCR1[15:0]	RW	比较捕获寄存器通道 1 的值。	0

14.4.15 比较/捕获寄存器 2 (TIM1_CH2CVR)

偏移地址: 0x38

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															LEVEL2
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2[15: 0]															

位	名称	访问	描述	复位值
[31:17]	Reserved	R0	保留。	0
16	LEVEL2	R0	捕获值对应的电平指示 bit	0
[15:0]	CCR2[15:0]	RW	比较捕获寄存器通道 2 的值。	0

14.4.16 比较/捕获寄存器 3 (TIM1_CH3CVR)

偏移地址: 0x3C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															LEVEL3
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR3[15: 0]															

位	名称	访问	描述	复位值
[31:17]	Reserved	R0	保留。	0
16	LEVEL3	R0	捕获值对应的电平指示 bit	0

[15:0]	CCR3[15:0]	RW	比较捕获寄存器通道 3 的值。	0
--------	------------	----	-----------------	---

14.4.17 比较/捕获寄存器 4 (TIM1_CH4CVR)

偏移地址: 0x40

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															LEVEL4
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR4[15: 0]															

位	名称	访问	描述	复位值
[31:17]	Reserved	RO	保留。	0
16	LEVEL4	RO	捕获值对应的电平指示 bit	0
[15:0]	CCR4[15:0]	RW	比较捕获寄存器通道 4 的值。	0

14.4.18 刹车和死区寄存器 (TIM1_BDTR)

偏移地址: 0x44

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]	DTG[7:0]								

位	名称	访问	描述	复位值
15	MOE	RW	主输出使能位。一旦刹车信号有效, 将被异步清零。 1: 允许 OCx 和 OCxN 设为输出; 0: 禁止 OCx 和 OCxN 的输出或者强制为空闲状态。	0
14	AOE	RW	自动输出使能。 1: MOE 可以被软件置位或者在下一个更新事件中被置位; 0: MOE 只能被软件置位。	0
13	BKP	RW	刹车输入极性设置位。 1: 刹车输入高电平有效; 0: 刹车输入低电平有效。 注: 当设置了 LOCK 级别 1 后, 该位不能被修改。对该位的写需要一个 PB 时钟以后才能生效。	0
12	BKE	RW	刹车功能使能位。 1: 开启刹车输入; 0: 禁止刹车输入。 注: 当设置了 LOCK 级别 1 后, 该位不能被修改。对该位的写需要一个 PB 时钟以后才能生效。	0
11	OSSR	RW	1: 当定时器不工作时, 一旦 CCxE=1 或 CCxNE=1, 首先开启 OC/OCN 并输出无效电平, 然后置 OCx、OCxN 使能输出信号=1; 0: 当定时器不工作时, 禁止 OC/OCN 输出。 注: 当设置了 LOCK 级别 1 后, 该位不能被修改。	0
10	OSSI	RW	1: 当定时器不工作时, 一旦 CCxE=1 或 CCxNE=1,	0

			OC/OCN 首先输出其空闲电平, 然后 OCx、OCxN 使能输出信号=1; 0: 当定时器不工作时, 禁止 OC/OCN 输出。 注: 当设置了 LOCK 级别 1 后, 该位不能被修改。	
[9:8]	LOCK[1:0]	RW	锁定功能设置域。 00: 关闭锁定功能; 01: 锁定级别 1, 不能写 DTG、BKE、BKP、AOE、OISx 和 OISxN 位; 10: 锁定级别 2, 不能写入锁定级别 1 中的各位, 也不能写入 CC 极性位以及 OSSR 和 OSSI 位; 11: 锁定级别 3, 不能写入锁定级别 2 中的各位, 也不能写入 CC 控制位。 注: 在系统复位后, 只能写一次 LOCK 位, 无法再次修改直到复位。	00b
[7:0]	DTG[7:0]	RW	死区设置位, 这些位定义了互补输出之间的死区持续时间。 假设 DT 表示其持续时间: DTG[7:5]=0xx=>DT=DTG[7:0]*Tdtg, Tdtg =TDS; DTG[7:5]=10x=>DT=(64+DTG[5:0])*Tdtg, Tdtg=2*TDS; DTG[7:5]=110=>DT=(32+DTG[4:0])*Tdtg, Tdtg =8×TDS; DTG[7:5]=111=>DT=(32+DTG[4:0])*Tdtg, Tdtg =16*TDS。 注: 一旦 LOCK 级别 (TIM1_BDTR 寄存器中的 LOCK[1:0]位) 设为 1、2 或 3, 则不能修改这些位。	0

14.4.19 DMA 控制寄存器 (TIM1_DMACFGR)

偏移地址: 0x48

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

位	名称	访问	描述	复位值
[15:13]	Reserved	RO	保留。	0
[12:8]	DBL[4:0]	RW	DMA 连续传送的长度, 实际值为此域的值+1。	0
[7:5]	Reserved	RO	保留。	0
[4:0]	DBA[4:0]	RW	这些位定义了 DMA 在连续模式下从控制寄存器 1 所在地址的偏移量。	0

14.4.20 连续模式的 DMA 地址寄存器 (TIM1_DMAADR)

偏移地址: 0x4C

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

DMAB[15:0]

位	名称	访问	描述	复位值
[31:0]	DMAB[31:0]	RW	连续模式下，DMA 的地址。	0

第 15 章 通用定时器 (GPTM)

通用定时器模块包含两个 16 位可自动重装的定时器 (TIM2、TIM3) 和一个 32 位可自动重装的定时器 (TIM4)，用于测量脉冲宽度或者产生特定频率的脉冲、PWM 波等。可用于自动化控制、电源等领域。

15.1 主要特征

16 位通用定时器的主要特征包括：

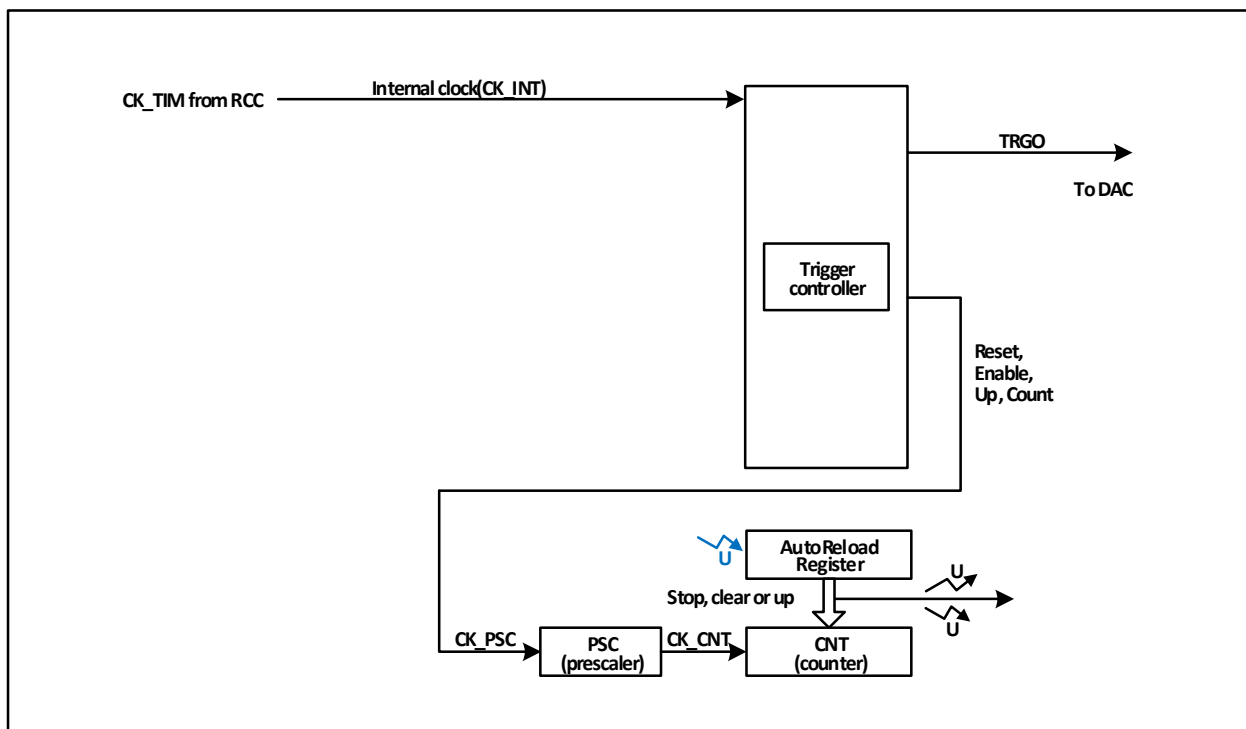
- 16 位自动重装计数器，支持增计数模式，减计数模式和增减计数模式
- 16 位预分频器，分频系数从 1~65536 之间动态可调
- 支持四路独立的比较捕获通道
- 每路比较捕获通道支持多种工作模式，比如：输入捕获、输出比较、PWM 生成和单脉冲输出
- 支持外部信号控制定时器
- 支持在多种模式下使用 DMA
- 支持增量式编码，定时器之间的级联和同步

32 位通用定时器的主要特征包括：

- 32 位自动重装计数器，支持增计数模式，减计数模式和增减计数模式
- 16 位预分频器，分频系数从 1~65536 之间动态可调
- 支持四路独立的比较捕获通道
- 每路比较捕获通道支持多种工作模式，比如：输入捕获、输出比较、PWM 生成和单脉冲输出
- 支持外部信号控制定时器
- 支持在多种模式下使用 DMA
- 支持增量式编码，定时器之间的级联和同步

15.2 原理和结构

图 15-1 通用定时器的结构框图



15.2.1 概述

如图 15-1 所示，通用定时器的结构大致可以分为三部分，即输入时钟部分，核心计数器部分和比较捕获通道部分。

通用定时器的时钟可以来自于 HB 总线时钟 (CK_INT)，可以来自外部时钟输入引脚 (TIMx_ETR)，可以来自于其他具有时钟输出功能的定时器 (ITRx)，还可以来自于比较捕获通道的输入端 (TIMx_CHx)。这些输入的时钟信号经过各种设定的滤波分频等操作后成为 CK_PSC 时钟，输出给核心计数器部分。另外，这些复杂的时钟来源还可以作为 TRGO 输出给其他的定时器、ADC 等外设。

通用定时器的核心是一个 16 位计数器 (CNT)。CK_PSC 经过预分频器 (PSC) 分频后，成为 CK_CNT 再最终输给 CNT，CNT 支持增计数模式、减计数模式和增减计数模式，并有一个自动重装值寄存器 (ATRLR) 在每个计数周期结束后为 CNT 重装初始值。

通用定时器拥有四组比较捕获通道，每组比较捕获通道都可以从专属的引脚上输入脉冲，也可以向引脚输出波形，即比较捕获通道支持输入和输出模式。比较捕获寄存器每个通道的输入都支持滤波、分频、边沿检测等操作，并支持通道间的互触发，还能为核心计数器 CNT 提供时钟。每个比较捕获通道都拥有一组比较捕获寄存器 (CHxCVR)，支持与主计数器 (CNT) 进行比较而输出脉冲。

15.2.2 通用定时器和高级定时器的区别

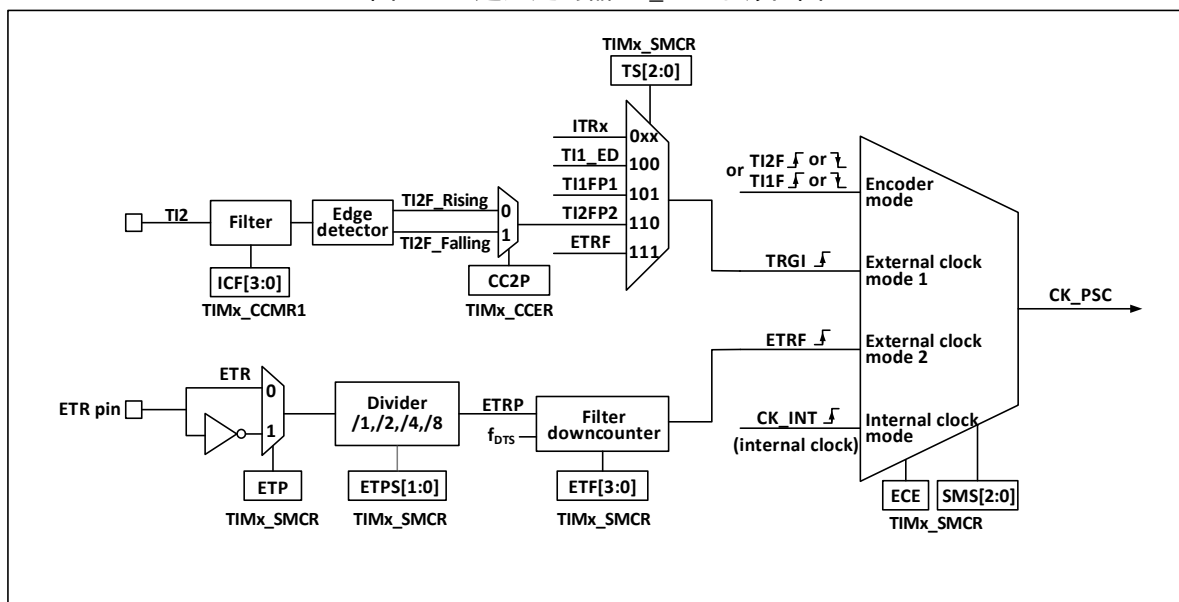
与高级定时器相比，通用定时器缺少以下功能：

- 1) 通用定时器缺少对核心计数器的计数周期进行计数的重复计数寄存器。
- 2) 通用定时器的比较捕获通道缺少死区产生，没有互补输出。
- 3) 通用定时器没有刹车信号机制。
- 4) 通用定时器的默认时钟 CK_INT 都来自 PB1，而高级定时器的 CK_INT 都来自 PB2。

15.2.3 时钟输入

本节论述 CK_PSC 的来源。此处截取通用定时器的整体结构框图的时钟源部分。

图 15-2 通用定时器 CK_PSC 来源框图



可选的输入时钟可以分为 4 类：

- 1) 外部时钟引脚 (ETR) 输入的路线：ETR→ETRP→ETRF；
- 2) 内部 PB 时钟输入路线：CK_INT；
- 3) 来自比较捕获通道引脚 (TIMx_CHx) 的路线：TIMx_CHx→TIx→TIxFPx，此路线也用于编码器模式；
- 4) 来自内部其他定时器的输入：ITRx。

通过决定 CK_PSC 来源的 SMS 的输入脉冲选择可以将实际的操作分为三类：

- 1) 选择内部时钟源 (CK_INT)；
- 2) 外部时钟源模式 1；
- 3) 外部时钟源模式 2；
- 4) 编码器模式。

上文提到的 4 种时钟源来源都可通过这 4 种操作选定。

15.2.3.1 内部时钟源 (CK_INT)

如果将 SMS 域保持为 000b 时启动通用定时器，那么就是选定内部时钟源 (CK_INT) 为时钟。此时 CK_INT 就是 CK_PSC。

15.2.3.2 外部时钟源模式 1

如果将 SMS 域设置为 111b 时，就会启用外部时钟源模式 1。启用外部时钟源 1 时，TRGI 被选定为 CK_PSC 的来源，值得注意的，用户还需要通过配置 TS 域来选择 TRGI 的来源。TS 域可选择以下几种脉冲作为时钟来源：

- 1) 内部触发 (ITRx, x 为 0, 1, 2, 3)；
- 2) 比较捕获通道 1 经过边缘检测器后的信号 (TI1F_ED)；
- 3) 比较捕获通道的信号 TI1FP1、TI2FP2；
- 4) 来自外部时钟引脚输入的信号 ETRF。

15.2.3.3 外部时钟源模式 2

使用外部触发模式 2 能在外部时钟引脚输入的每一个上升沿或下降沿计数。将 ECE 位置位时，将使用外部时钟源模式 2。使用外部时钟源模式 2 时，ETRF 被选定为 CK_PSC。ETR 引脚经过可选的反相器 (ETP)，分频器 (ETPS) 后成为 ETRP，再经过滤波器 (ETF) 后即成为 ETRF。

在 ECE 位置位且将 SMS 设为 111b 时，那么，相当于 TS 选择 ETRF 为输入。

15.2.3.4 编码器模式

将 SMS 置为 001b, 010b, 011b 将会启用编码器模式。启用编码器模式可以选择在 TI1FP1 和 TI2FP2 中某一个特定的电平下以另一个跳变沿作为信号进行信号输出。此模式用于外接编码器使用的情况下。具体功能参考 15.3.7 节。

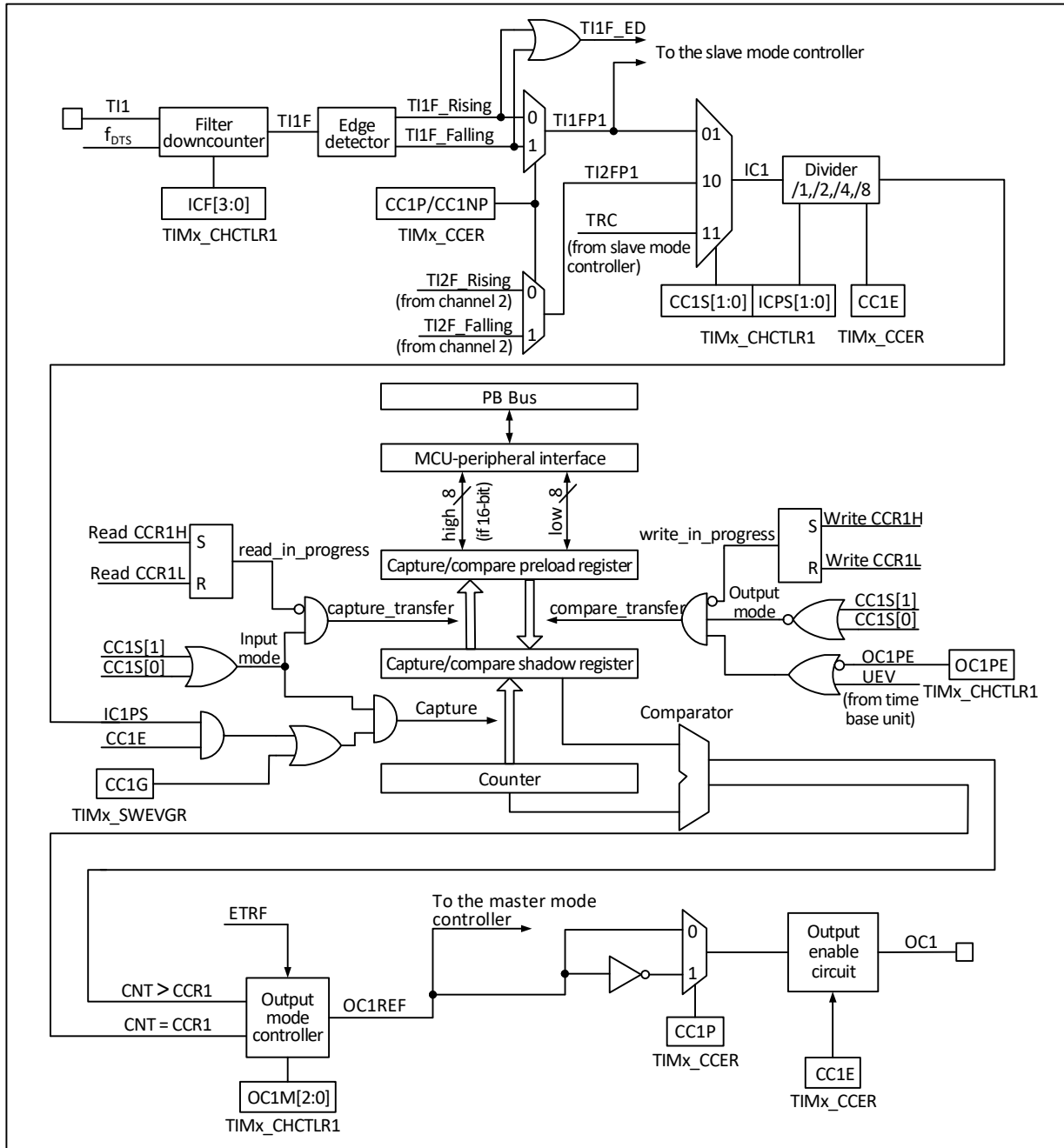
15.2.4 计数器和周边

CK_PSC 输入给预分频器 (PSC) 进行分频。PSC 是 16 位的，实际的分频系数相当于 R16_TIMx_PSC 的值+1。CK_PSC 经过 PSC 会成为 CK_INT。更改 R16_TIM1_PSC 的值并不会实时生效，而会在更新事件后更新给 PSC。更新事件包括 UG 位清零和复位。

15.2.5 比较捕获通道

比较捕获通道是定时器实现复杂功能的核心，它的核心是比较捕获寄存器，辅以外围输入部分的数字滤波，分频和通道间复用，输出部分的比较器和输出控制组成。比较捕获通道的结构框图如图 15-3 所示。

图 15-3 比较捕获通道的结构框图



信号从通道 x 引脚输入进来后可选做为 TI_x (TI_1 的来源可以不只是 CH1, 见定时器的框图 15-1), TI_1 经过滤波器 (ICF[3:0]) 生成 TI_{1F} , 再经过边沿检测器分成 TI_{1F_Rising} 和 $TI_{1F_Falling}$, 这两个信号经过选择 (CC1P) 生成 TI_{1FP1} , TI_{1FP1} 和来自通道 2 的 TI_{2FP1} 一起送给 CC1S 选择成为 IC1, 经过 ICPS 分频后送给比较捕获寄存器。

比较捕获寄存器由一个预装载寄存器和一个影子寄存器组成, 读写过程仅操作预装载寄存器。在捕获模式下, 捕获发生在影子寄存器上, 然后复制到预装载寄存器; 在比较模式下, 预装载寄存器的内容被复制到影子寄存器中, 然后影子寄存器的内容与核心计数器 (CNT) 进行比较。

15.3 功能和实现

通用定时器复杂功能的实现都是对定时器的比较捕获通道、时钟输入电路和计数器及周边组件进行操作实现的。定时器的时钟输入可以来自于包括比较捕获通道的输入在内的多个时钟源。对比

较捕获寄存通道和时钟源选择的操作直接决定其功能。比较捕获通道是双向的，可以工作在输入和输出模式。

15.3.1 输入捕获模式

输入捕获模式是定时器的基本功能之一。输入捕获模式的原理是，当检测到 ICxPS 信号上确定的边沿后，则产生捕获事件，计数器当前的值会被锁存到比较捕获寄存器（R16_TIMx_CHCTLRx）中。发生捕获事件时，CCxIF（在 R16_TIMx_INTFR 中）被置位，如果使能了中断或者 DMA，还会产生相应中断或者 DMA。如果发生捕获事件时，CCxIF 已经被置位了，那么 CCxOF 位会被置位。CCxIF 可由软件清除，也可以通过读取比较捕获寄存器由硬件清除。CCxOF 由软件清除。

举个通道 1 的例子来说明使用输入捕获模式的步骤，如下：

- 1) 配置 CCxS 域，选择 ICx 信号的来源。比如设为 10b，选择 TI1FP1 作为 IC1 的来源，不可以使用默认设置，CCxS 域默认是使比较捕获模块作为输出通道；
- 2) 配置 ICxF 域，设定 TI 信号的数字滤波器。数字滤波器会以确定的频率，采样确定的次数，再输出一个跳变。这个采样频率和次数是通过 ICxF 来确定的；
- 3) 配置 CCxP 位，设定 TIxFPx 的极性。比如保持 CC1P 位为低，选择上升沿跳变；
- 4) 配置 ICxPS 域，设定 ICx 信号成为 ICxPS 之间的分频系数。比如保持 ICxPS 为 00b，不分频；
- 5) 配置 CCxE 位，允许捕获核心计数器（CNT）的值到比较捕获寄存器中。置 CC1E 位；
- 6) 根据需要配置 CCxIE 和 CCxDE 位，决定是否允许使能中断或者 DMA。

至此已经将比较捕获通道配置完成。

当 TI1 输入了一个被捕获的脉冲时，核心计数器（CNT）的值会被记录到比较捕获寄存器中，CC1IF 被置位，当 CC1IF 在之前就已经被置位时，CC1OF 位也会被置位。如果 CC1IE 位，那么会产生一个中断；如果 CC1DE 被置位，会产生一个 DMA 请求。可以通过写事件产生寄存器的方式（R16_TIMx_SWEVGR）的方式由软件产生一个输入捕获事件。

15.3.2 比较输出模式

比较输出模式是定时器的基本功能之一。比较输出模式的原理是在核心计数器（CNT）的值与比较捕获寄存器的值一致时，输出特定的变化或波形。OCxM 域（在 R16_TIMx_CHCTLRx 中）和 CCxP 位（在 R16_TIMx_CCER 中）决定输出的是确定的高低电平还是电平翻转。产生比较一致事件时还会置 CCxIF 位，如果预先置了 CCxIE 位，则会产生一个中断；如果预先设置了 CCxDE 位，则会产生一个 DMA 请求。

配置为比较输出模式的步骤为下：

- 1) 配置核心计数器（CNT）的时钟源和自动重装值；
- 2) 设置好需要对比的计数值到比较捕获寄存器（R16_TIMx_CHxCVR）中；
- 3) 如果需要产生中断，置 CCxIE 位；
- 4) 保持 OCxPE 为 0，禁用比较捕获寄存器的预装载寄存器；
- 5) 设定输出模式，设置 OCxM 域和 CCxP 位；
- 6) 使能输出，置 CCxE 位；
- 7) 置 CEN 位启动定时器；

15.3.3 强制输出模式

定时器的比较捕获通道的输出模式可以由软件强制输出确定的电平，而不依赖比较捕获寄存器的影子寄存器和核心计数器的比较。

具体的做法是将 OCxM 置为 100b，即为强制将 OCxREF 置为低；或者将 OCxM 置为 101b，即为强制将 OCxREF 置为高。

需要注意的是，将 OCxM 强制置为 100b 或者 101b，内部主计数器和比较捕获寄存器的比较过程还在进行，相应的标志位还在置位，中断和 DMA 请求还在产生。

15.3.4 PWM 输入模式

PWM 输入模式是用来测量 PWM 的占空比和频率的，是输入捕获模式的一种特殊情况。除下列区别外，操作和输入捕获模式相同：PWM 占用两个比较捕获通道，且两个通道的输入极性设为相反，其中一个信号被设为触发输入，SMS 设为复位模式。

例如，测量从 TI1 输入的 PWM 波的周期和频率，需要进行以下操作：

- 1) 将 TI1 (TI1FP1) 设为 IC1 信号的输入。将 CC1S 置为 01b；
- 2) 将 TI1FP1 置为上升沿有效。将 CC1P 保持为 0；
- 3) 将 TI1 (TI1FP2) 置为 IC2 信号的输入。将 CC2S 置为 10b；
- 4) 选 TI1FP2 置为下降沿有效。将 CC2P 置为 1；
- 5) 时钟源的来源选择 TI1FP1。将 TS 设为 101b；
- 6) 将 SMS 设为复位模式，即 100b；
- 7) 使能输入捕获。CC1E 和 CC2E 置位。

15.3.5 PWM 输出模式

PWM 输出模式是定时器的基本功能之一。PWM 输出模式最常见的是使用重装值确定 PWM 频率，使用捕获比较寄存器确定占空比的方法。将 OCxM 域中置 110b 或者 111b 使用 PWM 模式 1 或者模式 2，置 OCxPE 位使能预装载寄存器，最后置 ARPE 位使能预装载寄存器的自动重装。在发生一个更新事件时，预装载寄存器的值才能被送到影子寄存器，所以在核心计数器开始计数之前，需要置 UG 位来初始化所有寄存器。在 PWM 模式下，核心计数器和比较捕获寄存器一直在进行比较，根据 CMS 位，定时器能够输出边沿对齐或者中央对齐的 PWM 信号。

- 边沿对齐

使用边沿对齐时，核心计数器增计数或者减计数，在 PWM 模式 1 的情景下，在核心计数器的值大于比较捕获寄存器时，OCxREF 上升为高；当核心计数器的值小于比较捕获寄存器时（比如核心计数器增长到 R16_TIMx_ATRLR 的值而恢复成全 0 时），OCxREF 下降为低。

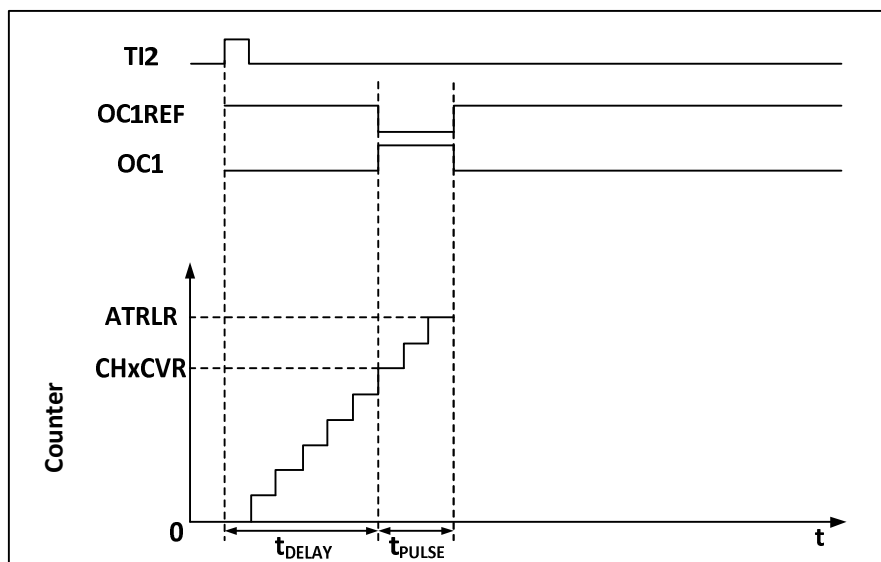
- 中央对齐

使用中央对齐模式时，核心计数器运行在增计数和减计数交替进行的模式下，OCxREF 在核心计数器和比较捕获寄存器的值一致时进行上升和下降的跳变。但比较标志在三种中央对齐模式下，置位的时机有所不同。在使用中央对齐模式时，最好在启动核心计数器之前产生一个软件更新标志（置 UG 位）。

15.3.6 单脉冲模式

单脉冲模式可以响应一个特定的事件，在一个延迟之后产生一个脉冲，延迟和脉冲的宽度可编程。置 OPM 位可以使核心计数器在产生下一个更新事件 UEV 时（计数器翻转到 0）停止。

图 15-4 事件产生和脉冲响应



如图 15-4 所示，需要在 TI2 输入引脚上检测到一个上升沿开始，延迟 T_{delay} 之后，在 OC1 上产生一个长度为 T_{pulse} 的正脉冲：

- 1) 设定 TI2 为触发。置 CC2S 域为 01b，把 TI2FP2 映射到 TI2；置 CC2P 位为 0b，TI2FP2 设为上升沿检测；置 TS 域为 110b，TI2FP2 设为触发源；置 SMS 域为 110b，TI2FP2 被用来启动计数器；
- 2) T_{delay} 由比较捕获寄存器定义， T_{pulse} 由自动重装值寄存器的值和比较捕获寄存器的值确定。

15.3.7 编码器模式

编码器模式是定时器的一个典型应用，可以用来接入编码器的双相输出，核心计数器的计数方向和编码器的转轴方向同步，编码器每输出一个脉冲就会使核心计数器加一或减一。使用编码器的步骤为：将 SMS 域置为 001b（只在 TI2 边沿计数）、010b（只在 TI1 边沿计数）或者 011b（在 TI1 和 TI2 双边沿计数），将编码器接到比较捕获通道 1、2 的输入端，设一个重装值计数器的值，这个值可以设的大一点。在编码器模式时，定时器内部的比较捕获寄存器，预分频器，重复计数寄存器等都正常工作。下表表明了计数方向和编码器信号的关系。

表 15-1 定时器编码器模式的计数方向和编码器信号之间的关系

计数有效边沿	相对信号的电平	TI1FP1 信号边沿		TI2FP2 信号	
		上升沿	下降沿	上升沿	下降沿
仅在 TI1 边沿计数	高	向下计数	向上计数	不计数	
	低	向上计数	向下计数		
仅在 TI2 边沿计数	高	不计数		向上计数	向下计数
	低			向下计数	向上计数
在 TI1 和 TI2 双边沿计数	高	向下计数	向上计数	向上计数	向下计数
	低	向上计数	向下计数	向下计数	向上计数

15.3.8 定时器同步模式

定时器能够输出时钟脉冲（TRG0），也能接收其他定时器的输入（ITRx）。不同的定时器的 ITRx 的来源（别的定时器的 TRG0）是不一样的。定时器内部触发连接如表 15-2 所示。

表 15-2 GTPM 内部触发连接

从定时器	ITR0 (TS=000)	ITR1 (TS=001)	ITR2 (TS=010)	ITR3 (TS=011)
TIM2	TIM1	USB	TIM3	TIM4

TIM3	TIM1	TIM2	—	TIM4
TIM4	TIM1	TIM2	TIM3	—

15.3.9 调试模式

当系统进入调试模式时，根据 DBG 模块的设置可以控制定时器继续运转或者停止。

15.4 寄存器描述

表 15-3 TIM2 相关寄存器列表

名称	偏移地址	描述	复位值
R16_TIM2_CTLR1	0x40000000	TIM2 控制寄存器 1	0x0000
R16_TIM2_CTLR2	0x40000004	TIM2 控制寄存器 2	0x0000
R16_TIM2_SMCFR	0x40000008	TIM2 从模式控制寄存器	0x0000
R16_TIM2_DMAINTENR	0x4000000C	TIM2 DMA/中断使能寄存器	0x0000
R16_TIM2_INTFR	0x40000010	TIM2 中断状态寄存器	0x0000
R16_TIM2_SWEVGR	0x40000014	TIM2 事件产生寄存器	0x0000
R16_TIM2_CHCTLR1	0x40000018	TIM2 比较/捕获控制寄存器 1	0x0000
R16_TIM2_CHCTLR2	0x4000001C	TIM2 比较/捕获控制寄存器 2	0x0000
R16_TIM2_CCER	0x40000020	TIM2 比较/捕获使能寄存器	0x0000
R16_TIM2_CNT	0x40000024	TIM2 计数器	0x0000
R16_TIM2_PSC	0x40000028	TIM2 计数时钟预分频器	0x0000
R16_TIM2_ATRLR	0x4000002C	TIM2 自动重装值寄存器	0xFFFF
R32_TIM2_CH1CVR	0x40000034	TIM2 比较/捕获寄存器 1	0x00000000
R32_TIM2_CH2CVR	0x40000038	TIM2 比较/捕获寄存器 2	0x00000000
R32_TIM2_CH3CVR	0x4000003C	TIM2 比较/捕获寄存器 3	0x00000000
R32_TIM2_CH4CVR	0x40000040	TIM2 比较/捕获寄存器 4	0x00000000
R16_TIM2_DMAR	0x40000048	TIM2 DMA 控制寄存器	0x0000
R32_TIM2_DMAADR	0x4000004C	TIM2 连续模式的 DMA 地址寄存器	0x00000000

表 15-4 TIM3 相关寄存器列表

名称	偏移地址	描述	复位值
R16_TIM3_CTLR1	0x40000400	TIM3 控制寄存器 1	0x0000
R16_TIM3_CTLR2	0x40000404	TIM3 控制寄存器 2	0x0000
R16_TIM3_SMCFR	0x40000408	TIM3 从模式控制寄存器	0x0000
R16_TIM3_DMAINTENR	0x4000040C	TIM3 DMA/中断使能寄存器	0x0000
R16_TIM3_INTFR	0x40000410	TIM3 中断状态寄存器	0x0000
R16_TIM3_SWEVGR	0x40000414	TIM3 事件产生寄存器	0x0000
R16_TIM3_CHCTLR1	0x40000418	TIM3 比较/捕获控制寄存器 1	0x0000
R16_TIM3_CHCTLR2	0x4000041C	TIM3 比较/捕获控制寄存器 2	0x0000
R16_TIM3_CCER	0x40000420	TIM3 比较/捕获使能寄存器	0x0000
R16_TIM3_CNT	0x40000424	TIM3 计数器	0x0000
R16_TIM3_PSC	0x40000428	TIM3 计数时钟预分频器	0x0000
R16_TIM3_ATRLR	0x4000042C	TIM3 自动重装值寄存器	0xFFFF
R32_TIM3_CH1CVR	0x40000434	TIM3 比较/捕获寄存器 1	0x00000000

R32_TIM3_CH2CVR	0x40000438	TIM3 比较/捕获寄存器 2	0x00000000
R32_TIM3_CH3CVR	0x4000043C	TIM3 比较/捕获寄存器 3	0x00000000
R32_TIM3_CH4CVR	0x40000440	TIM3 比较/捕获寄存器 4	0x00000000
R16_TIM3_DMACHFR	0x40000448	TIM3 DMA 控制寄存器	0x0000
R32_TIM3_DMAADR	0x4000044C	TIM3 连续模式的 DMA 地址寄存器	0x00000000

表 15-5 TIM4 相关寄存器列表

名称	偏移地址	描述	复位值
R16_TIM4_CTLR1	0x40000800	TIM4 控制寄存器 1	0x0000
R16_TIM4_CTLR2	0x40000804	TIM4 控制寄存器 2	0x0000
R16_TIM4_SMCFGR	0x40000808	TIM4 从模式控制寄存器	0x0000
R16_TIM4_DMAINTENR	0x4000080C	TIM4 DMA/中断使能寄存器	0x0000
R16_TIM4_INTFR	0x40000810	TIM4 中断状态寄存器	0x0000
R16_TIM4_SWEVGR	0x40000814	TIM4 事件产生寄存器	0x0000
R16_TIM4_CHCTL1	0x40000818	TIM4 比较/捕获控制寄存器 1	0x0000
R16_TIM4_CHCTL2	0x4000081C	TIM4 比较/捕获控制寄存器 2	0x0000
R16_TIM4_CCER	0x40000820	TIM4 比较/捕获使能寄存器	0x0000
R32_TIM4_CNT	0x40000824	TIM4 计数器	0x00000000
R16_TIM4_PSC	0x40000828	TIM4 计数时钟预分频器	0x0000
R32_TIM4_ATRLR	0x4000082C	TIM4 自动重装值寄存器	0xFFFFFFFF
R32_TIM4_CH1CVR	0x40000834	TIM4 比较/捕获寄存器 1	0x00000000
R32_TIM4_CH2CVR	0x40000838	TIM4 比较/捕获寄存器 2	0x00000000
R32_TIM4_CH3CVR	0x4000083C	TIM4 比较/捕获寄存器 3	0x00000000
R32_TIM4_CH4CVR	0x40000840	TIM4 比较/捕获寄存器 4	0x00000000
R16_TIM4_DMACHFR	0x40000848	TIM4 DMA 控制寄存器	0x0000
R32_TIM4_DMAADR	0x4000084C	TIM4 连续模式的 DMA 地址寄存器	0x00000000

15.4.1 控制寄存器 1 (TIMx_CTLR1) (x=2/3/4)

偏移地址：0x00

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMR_CAP_LVL_EN	TMR_CAP_OV_EN	Reserved				CKD[1:0]	ARPE	CMS[1:0]	DIR	OPM	URS	UDIS	CEN		

位	名称	访问	描述	复位值
15	TMR_CAP_LVL_EN	RW	双沿捕获模式下，捕获电平指示使能： 0：关闭指示功能 1：使能指示功能。 注：使能后，CHxCVR 的[16]指示捕获值对应的电平。	0
14	TMR_CAP_OV_EN	RW	捕获值模式配置： 0：捕获值为实际计数器的值 1：当捕获前产生计数器溢出时，CHxCVR 值为 0xFFFF。	0
[13:10]	Reserved	RO	保留。	0
[9:8]	CKD[1:0]	RW	这 2 位定义在定时器时钟(CK_INT)频率、数字滤波器所用的采样时钟之间的分频比例：	00b

			00: Tdts=Tck_int; 01: Tdts= 2xTck_int; 10: Tdts= 4xTck_int; 11: 保留。	
7	ARPE	RW	自动重装预装使能位: 1: 使能自动重装值寄存器 (ATRLR) ; 0: 禁止自动重装值寄存器 (ATRLR) 。	0
[6:5]	CMS[1:0]	RW	中央对齐模式选择: 00: 边沿对齐模式。计数器依据方向位(DIR) 向上或向下计数。 01: 中央对齐模式 1。计数器交替地向上和向下计数。配置为输出的通道(CHCTLRx 寄存器中 CCxS=00) 的输出比较中断标志位, 只在计数器向下计数时被设置。 10: 中央对齐模式 2。计数器交替地向上和向下计数。配置为输出的通道(CHCTLRx 寄存器中 CCxS=00) 的输出比较中断标志位, 只在计数器向上计数时被设置。 11: 中央对齐模式 3。计数器交替地向上和向下计数。配置为输出的通道(CHCTLRx 寄存器中 CCxS=00) 的输出比较中断标志位, 在计数器向上和向下计数时均被设置。 注: 在计数器使能时(CEN=1), 不允许从边沿对齐模式转换到中央对齐模式。	00b
4	DIR	RW	计数器方向: 0: 计数器的计数模式为增计数; 1: 计数器的计数模式为减计数。 注: 当计数器配置为中央对齐模式或编码器模式时, 该位无效。	0
3	OPM	RW	单脉冲模式。 1: 在发生下一次更新事件(清除 CEN 位)时, 计数器停止; 0: 在发生下一次更新事件时, 计数器不停止。	0
2	URS	RW	更新请求源, 软件通过该位选择 UEV 事件的源。 1: 如果使能了更新中断或 DMA 请求, 则只有计数器溢出/下溢才产生更新中断或 DMA 请求; 0: 如果使能了更新中断或 DMA 请求, 则下述任一事件产生更新中断或 DMA 请求: -计数器溢出/下溢 -设置 UG 位 -从模式控制器产生的更新	0
1	UDIS	RW	禁止更新, 软件通过该位允许/禁止 UEV 事件的产生。 1: 禁止 UEV。不产生更新事件, 各寄存器(ATRLR、PSC、CHCTLRx)保持它们的值。如果设置了 UG 位或从模式控制器发出了一个硬件复位, 则计数器和预	0

			分频器被重新初始化。 0: 允许 UEV。更新 (UEV) 事件由下述任一事件产生： - 计数器溢出/下溢 - 设置 UG 位 - 从模式控制器产生的更新具有缓存的寄存器被装入它们的预装载值。	
0	CEN	RW	使能计数器 (Counter enable)。 1: 使能计数器； 0: 禁止计数器。 注：在软件设置了 CEN 位后，外部时钟、门控模式和编码器模式才能工作。触发模式可以自动地通过硬件设置 CEN 位。	0

15.4.2 控制寄存器 2 (TIMx_CTLR2) (x=2/3/4)

偏移地址：0x04

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								TI1S	MMS[2:0]		CCDS	Reserved			

位	名称	访问	描述	复位值
[15:8]	Reserved	RO	保留。	0
7	TI1S	RW	TI1 选择： 1: TIMx_CH1、TIMx_CH2 和 TIMx_CH3 引脚经异或后连到 TI1 输入； 0: TIMx_CH1 引脚直连到 TI1 输入。	0
[6:4]	MMS[2:0]	RW	主模式选择：这 3 位用于选择在主模式下送到从定时器的同步信息 (TRGO)。可能的组合如下： 000: 复位 - UG 位被用于作为触发输出 (TRGO)。如果是触发输入产生的复位 (从模式控制器处于复位模式)，则 TRGO 上的信号相对实际的复位会有一个延迟； 001: 使能 - 计数器使能信号 CNT_EN 被用于作为触发输出 (TRGO)。有时需要在同一时间启动多个定时器或控制在一段时间内使能从定时器。计数器使能信号是通过 CEN 控制位和门控模式下的触发输入信号的逻辑或产生。当计数器使能信号受控于触发输入时，TRGO 上会有一个延迟，除非选择了主/从模式 (见 TIMx_SMCGR 寄存器中 MSM 位的描述)； 010: 更新事件被选为触发输入 (TRGO)。例如，一个主定时器的时钟可以被用作一个从定时器的预分频器； 011: 比较脉冲，在发生一次捕获或一次比较成功时，当要设置 CC1IF 标志时 (即使它已经为高)，触发输出送出一个正脉冲 (TRGO)； 100: OC1REF 信号被用于作为触发输出 (TRGO)； 101: OC2REF 信号被用于作为触发输出 (TRGO)；	000b

			110: OC3REF 信号被用于作为触发输出 (TRGO); 111: OC4REF 信号被用于作为触发输出 (TRGO)。	
3	CCDS	RW	1: 当发生更新事件时, 送出 CHxCVR 的 DMA 请求; 0: 当发生 CHxCVR 时, 产生 CHxCVR 的 DMA 请求。	0
[2:0]	Reserved	RO	保留。	0

15.4.3 从模式控制寄存器 (TIMx_SMCFGR) (x=2/3/4)

偏移地址: 0x08

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP		ECE		ETPS[1:0]		ETF[3:0]			MSM	TS[2:0]		Reserved		SMS[2:0]	

位	名称	访问	描述	复位值
15	ETP	RO	ETR 触发极性选择, 该位选择是直接输入 ETR 还是输入 ETR 的反相。 1: 将 ETR 反相, 低电平或下降沿有效; 0: ETR, 高电平或上升沿有效。	0
14	ECE	RW	外部时钟模式 2 启用选择。 1: 使能外部时钟模式 2; 0: 禁用外部时钟模式 2。 <i>注 1: 从模式可以与外部时钟模式 2 同时使用: 复位模式, 门控模式和触发模式; 但是, 这时 TRGI 不能连到 ETRF (TS 位不能是 111b)。</i> <i>注 2: 外部时钟模式 1 和外部时钟模式 2 同时被使能时, 外部时钟的输入是 ETRF。</i>	0
[13:12]	ETPS[1:0]	RW	外部触发信号 (ETRP) 分频, 这个信号频率最大不能超过是 TIMxCLK 频率的 1/4, 可以通过这个域来降频。 00: 关闭预分频; 01: ETRP 频率除以 2; 10: ETRP 频率除以 4; 11: ETRP 频率除以 8。	00b
[11:8]	ETF[3:0]	RW	外部触发滤波, 实际上, 数字滤波器是一个事件计数器, 它使用一定的采样的频率, 记录到 N 个事件后会产生一个输出的跳变。 0000: 无滤波器, 以 Fdts 采样; 0001: 采样频率 F _{sampling} =F _{ck_int} , N=2; 0010: 采样频率 F _{sampling} =F _{ck_int} , N=4; 0011: 采样频率 F _{sampling} =F _{ck_int} , N=8; 0100: 采样频率 F _{sampling} =F _{dts} /2, N=6; 0101: 采样频率 F _{sampling} =F _{dts} /2, N=8; 0110: 采样频率 F _{sampling} =F _{dts} /4, N=6; 0111: 采样频率 F _{sampling} =F _{dts} /4, N=8; 1000: 采样频率 F _{sampling} =F _{dts} /8, N=6; 1001: 采样频率 F _{sampling} =F _{dts} /8, N=8; 1010: 采样频率 F _{sampling} =F _{dts} /16, N=5;	0000b

			1011: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/16$, $N=6$; 1100: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/16$, $N=8$; 1101: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/32$, $N=5$; 1110: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/32$, $N=6$; 1111: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/32$, $N=8$ 。	
7	MSM	RW	主/从模式选择: 1: 触发输入 (TRGI) 上的事件被延迟了, 以允许在当前定时器 (通过 TRGO) 与它的从定时器间的完美同步。这对要求把几个定时器同步到一个单一的外部事件时是非常有用的; 0: 不发挥作用。	0
[6:4]	TS[2:0]	RW	触发选择域, 这 3 位选择用于同步计数器的触发输入源。 000: 内部触发 0 (ITR0); 001: 内部触发 1 (ITR1); 010: 内部触发 2 (ITR2); 011: 内部触发 3 (ITR3); 100: TI1 的边沿检测器 (TI1F_ED); 101: 滤波后的定时器输入 1 (TI1FP1); 110: 滤波后的定时器输入 2 (TI2FP2); 111: 外部触发输入 (ETRF); 以上只有在 SMS 为 0 时改变。	000b
3	Reserved	RO	保留。	0
[2:0]	SMS[2:0]	RW	输入模式选择域。选择核心计数器的时钟和触发模式。 000: 由内部时钟 CK_INT 驱动; 001: 编码器模式 1, 根据 TI1FP1 的电平, 核心计数器在 TI2FP2 的边沿增减计数; 010: 编码器模式 2, 根据 TI2FP2 的电平, 核心计数器在 TI1FP1 的边沿增减计数; 011: 编码器模式 3, 根据另一个信号的输入电平, 核心计数器在 TI1FP1 和 TI2FP2 的边沿增减计数; 100: 复位模式, 触发输入 (TRGI) 的上升沿将初始化计数器, 并且产生一个更新寄存器的信号; 101: 门控模式, 当触发输入 (TRGI) 为高时, 计数器的时钟开启; 在触发输入变为低, 计数器停止, 计数器的启停都是受控的; 110: 触发模式, 计数器在触发输入 TRGI 的上升沿启动, 只有计数器的启动是受控的; 111: 外部时钟模式 1, 选中的触发输入 (TRGI) 的上升沿驱动计数器。	000b

15.4.4 DMA/中断使能寄存器 (TIMx_DMAINTENR) (x=2/3/4)

偏移地址: 0x0C

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

Reserved	TDE	Reserved	CC4DE	CC3DE	CC2DE	CC1DE	UDE	Reserved	TIE	Reserved	CC4IE	CC3IE	CC2IE	CC1IE	UIE
----------	-----	----------	-------	-------	-------	-------	-----	----------	-----	----------	-------	-------	-------	-------	-----

位	名称	访问	描述	复位值
15	Reserved	RO	保留。	0
14	TDE	RW	触发 DMA 请求使能位。 1: 允许触发 DMA 请求; 0: 禁止触发 DMA 请求。	0
13	Reserved	RW	保留	0
12	CC4DE	RW	比较捕获通道 4 的 DMA 请求使能位。 1: 允许比较捕获通道 4 的 DMA 请求; 0: 禁止比较捕获通道 4 的 DMA 请求。	0
11	CC3DE	RW	比较捕获通道 3 的 DMA 请求使能位。 1: 允许比较捕获通道 3 的 DMA 请求; 0: 禁止比较捕获通道 3 的 DMA 请求。	0
10	CC2DE	RW	比较捕获通道 2 的 DMA 请求使能位。 1: 允许比较捕获通道 2 的 DMA 请求; 0: 禁止比较捕获通道 2 的 DMA 请求。	0
9	CC1DE	RW	比较捕获通道 1 的 DMA 请求使能位。 1: 允许比较捕获通道 1 的 DMA 请求; 0: 禁止比较捕获通道 1 的 DMA 请求。	0
8	UDE	RW	更新的 DMA 请求使能位。 1: 允许更新的 DMA 请求; 0: 禁止更新的 DMA 请求。	0
7	Reserved	RO	保留。	0
6	TIE	RW	触发中断使能位。 1: 使能触发中断; 0: 禁止触发中断。	0
5	Reserved	RO	保留。	0
4	CC4IE	RW	比较捕获通道 4 中断使能位。 1: 允许比较捕获通道 4 中断; 0: 禁止比较捕获通道 4 中断。	0
3	CC3IE	RW	比较捕获通道 3 中断使能位。 1: 允许比较捕获通道 3 中断; 0: 禁止比较捕获通道 3 中断。	0
2	CC2IE	RW	比较捕获通道 2 中断使能位。 1: 允许比较捕获通道 2 中断; 0: 禁止比较捕获通道 2 中断。	0
1	CC1IE	RW	比较捕获通道 1 中断使能位。 1: 允许比较捕获通道 1 中断; 0: 禁止比较捕获通道 1 中断。	0
0	UIE	RW	更新中断使能位。 1: 允许更新中断; 0: 禁止更新中断。	0

15.4.5 中断状态寄存器 (R16_TIMx_INTFR) (x=2/3/4)

偏移地址：0x10

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	CC40F	CC30F	CC20F	CC10F	Reserved	TIF	Reserved	CC41F	CC31F	CC21F	CC11F	UIF			

位	名称	访问	描述	复位值
[15:13]	Reserved	RO	保留。	0
12	CC40F	RWO	比较捕获通道 4 重复捕获标志位。	0
11	CC30F	RWO	比较捕获通道 3 重复捕获标志位。	0
10	CC20F	RWO	比较捕获通道 2 重复捕获标志位。	0
9	CC10F	RWO	比较捕获通道 1 重复捕获标志位，仅用于比较捕获通道被配置为输入捕获模式时。该标记由硬件置位，软件写 0 可清除此位。 1：计数器的值被捕获到捕获比较寄存器时，CC11F 的状态已经被置位； 0：无重复捕获产生。	0
[8:7]	Reserved	RO	保留。	0
6	TIF	RWO	触发器中断标志位，当发生触发事件时由硬件对该位置位，由软件清零。触发事件包括从除门控模式外的其它模式时，在 TRGI 输入端检测到有效边沿，或门控模式下的任一边沿。 1：触发器事件产生； 0：无触发器事件产生。	0
5	Reserved	RO	保留。	0
4	CC41F	RWO	比较捕获通道 4 中断标志位。	0
3	CC31F	RWO	比较捕获通道 3 中断标志位。	0
2	CC21F	RWO	比较捕获通道 2 中断标志位。	0
1	CC11F	RWO	比较捕获通道 1 中断标志位。 如果比较捕获通道配置为输出模式，当计数器值与比较值匹配时该位由硬件置位，但在中心对称模式下除外。该位由软件清零。 1：核心计数器的值与比较捕获寄存器 1 的值匹配； 0：无匹配发生。 如果比较捕获通道 1 配置为输入模式，当捕获事件发生时该位由硬件置位，它由软件清零或通过读比较捕获寄存器清零。 1：计数器值已被捕获比较捕获寄存器 1； 0：无输入捕获产生。	0
0	UIF	RWO	更新中断标志位，当产生更新事件时该位由硬件置位，由软件清零。 1：更新中断产生； 0：无更新事件产生。 以下情形会产生更新事件： 若 UDIS=0，当重复计数器数值上溢或下溢时； 若 URS=0、UDIS=0，当置 UG 位时，或当通过软件对计数器核心计数器重新初始化时；	0

			若 URS=0、UDIS=0，当计数器 CNT 被触发事件重新初始化时；	
--	--	--	--------------------------------------	--

15.4.6 事件产生寄存器 (TIMx_SWEVGR) (x=2/3/4)

偏移地址：0x14

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

Reserved					TG	Reserved	CC4G	CC3G	CC2G	CC1G	UG
----------	--	--	--	--	----	----------	------	------	------	------	----

位	名称	访问	描述	复位值
[15:7]	Reserved	RO	保留。	0
6	TG	WO	触发事件产生位，该位由软件置位，硬件清零，用于产生一个触发事件。 1：产生一个触发事件，TIF 被置位，若使能对应的中断和 DMA，则产生相应的中断和 DMA； 0：无动作。	0
5	Reserved	RO	保留。	0
4	CC4G	WO	比较捕获事件产生位 4。产生比较捕获事件 4。	0
3	CC3G	WO	比较捕获事件产生位 3。产生比较捕获事件 3。	0
2	CC2G	WO	比较捕获事件产生位 2。产生比较捕获事件 2。	0
1	CC1G	WO	比较捕获事件产生位 1，产生比较捕获事件 1。该位由软件置位，由硬件清零。用于产生一个比较捕获事件。 1：在比较捕获通道 1 上产生一个比较捕获事件：若比较捕获通道 1 配置为输出：置 CC1IF 位。若使能对应的中断和 DMA，则产生相应的中断和 DMA；若比较捕获通道 1 配置为输入：当前核心计数器的值被捕获至比较捕获寄存器 1；置 CC1IF 位，若使能了对应的中断和 DMA，则产生相应的中断和 DMA。若 CC1IF 已经置位，则置 CC1OF 位。 0：无动作。	0
0	UG	WO	更新事件产生位，产生更新事件。该位由软件置位，由硬件自动清零。 1：初始化计数器，并产生一个更新事件； 0：无动作。 注：预分频器的计数器也被清零，但是预分频系数不变。若在中心对称模式下或增计数模式下则核心计数器被清零；若减计数模式下则核心计数器取重装值寄存器的值。	0

15.4.7 比较/捕获控制寄存器 1 (TIMx_CHCTLR1) (x=2/3/4)

偏移地址：0x18

通道可用于输入(捕获模式)或输出(比较模式)，通道的方向由相应的 CCxS 位定义。该寄存器其它位的作用在输入和输出模式下不同。OCxx 描述了通道在输出模式下的功能，ICxx 描述了通道在输入模式下的功能。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2CE	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
IC2F[3:0]				IC2PSC[1:0]				IC1F[3:0]			IC1PSC[1:0]				

比较模式（引脚方向为输出）：

位	名称	访问	描述	复位值
15	OC2CE	RW	比较捕获通道 2 清零使能位。 1：一旦检测到 ETRF 输入高电平，清除 OC2REF 位零； 0：OC2REF 不受 ETRF 输入的影响。	0
[14:12]	OC2M[2:0]	RW	比较捕获通道 2 模式设置域。 该 3 位定义了输出参考信号 OC2REF 的动作，而 OC2REF 决定了 OC2、OC2N 的值。OC2REF 是高电平有效，而 OC2 和 OC2N 的有效电平取决于 CC2P、CC2NP 位。 000：冻结。比较捕获寄存器的值与核心计数器间的比较值对 OC1REF 不起作用； 001：强制设为有效电平。当核心计数器与比较捕获寄存器 1 的值相同时，强制 OC1REF 为高； 010：强制设为无效电平。当核心计数器的值与比较捕获寄存器 1 相同时，强制 OC1REF 为低； 011：翻转。当核心计数器与比较捕获寄存器 1 的值相同时，翻转 OC1REF 的电平。 100：强制为无效电平。强制 OC1REF 为低。 101：强制为有效电平。强制 OC1REF 为高。 110：PWM 模式 1：在向上计数时，一旦核心计数器大于比较捕获寄存器的值时通道 1 为无效电平，否则为有效电平；在向下计数时，一旦核心计数器大于比较捕获寄存器的值时通道 1 为有效电平，否则为无效电平。 111：PWM 模式 2：在向上计数时，一旦核心计数器大于比较捕获寄存器的值时，通道 1 为有效电平，否则为无效电平；在向下计数时，一旦核心计数器大于比较捕获寄存器的值时，通道 1 为无效电平，否则为有效电平 (OC1REF=1)。 注：一旦 LOCK 级别设为 3 并且 CC1S=00b 则该位不能被修改。在 PWM 模式 1 或 PWM 模式 2 中，只有当比较结果改变了或在输出比较模式中从冻结模式切换到 PWM 模式时，OC1REF 电平才改变。	000b
11	OC2PE	RW	比较捕获寄存器 2 预装载使能位。 1：开启比较捕获寄存器的预装载功能，读写操作仅对预装载寄存器操作，比较捕获寄存器 1 的预装载值在更新事件到来时被加载至当前影子寄存器中； 0：禁止比较捕获寄存器 2 的预装载功能，可随时写入比较捕获寄存器 2，并且新写入的数值立即起作用。	0

			用。 注：一旦 LOCK 级别设为 3 并且 CC1S=00，则该位不能被修改。仅仅在单脉冲模式下 (OPM=1) 可以在未确认预装载寄存器情况下使用 PWM 模式，否则其动作不确定。	
10	OC2FE	RW	比较捕获通道 2 快速使能位，该位用于加快比较捕获通道输出对触发输入事件的响应。 1：输入到触发器的有效沿的作用就像发生了一次比较匹配。因此，OC 被设置为比较电平而与比较结果无关。采样触发器的有效沿和比较捕获通道 2 输出间的延时被缩短为 3 个时钟周期； 0：根据计数器与比较捕获寄存器 2 的值，比较捕获通道 2 正常操作，即使触发器是打开的。当触发器的输入有一个有效沿时，激活比较捕获通道 2 输出的最小延时为 5 个时钟周期。 OC2FE 只在通道被配置成 PWM1 或 PWM2 模式时起作用；	0
[9:8]	CC2S[1:0]	RW	比较捕获通道 2 输入选择域。 00：比较捕获通道 2 被配置为输出； 01：比较捕获通道 2 被配置为输入，IC2 映射在 TI2 上； 10：比较捕获通道 2 被配置为输入，IC2 映射在 TI1 上； 11：比较捕获通道 2 被配置为输入，IC2 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时 (由 TS 位选择)。 注：比较捕获通道 2 仅在通道关闭时 (CC2E 为零时) 才是可写的。	00b
7	OC1CE	RW	比较捕获通道 1 清零使能位。	0
[6:4]	OC1M[2:0]	RW	比较捕获通道 1 模式设置域。	0
3	OC1PE	RW	比较捕获寄存器 1 预装载使能位。	0
2	OC1FE	RW	比较捕获通道 1 快速使能位。	0
[1:0]	CC1S[1:0]	RW	比较捕获通道 1 输入选择域。	0

捕获模式（引脚方向为输入）：

位	名称	访问	描述	复位值
[15:12]	IC2F[3:0]	RW	输入捕获滤波器 2 配置域，这几位设置了 TI1 输入的采样频率及数字滤波器长度。数字滤波器由一个事件计数器组成，它记录到 N 个事件后会产生一个输出的跳变。 0000：无滤波器，以 fDTS 采样； 1000：采样频率 $F_{\text{sampling}} = F_{\text{dts}}/8$ ，N=6； 0001：采样频率 $F_{\text{sampling}} = F_{\text{ck_int}}$ ，N=2； 1001：采样频率 $F_{\text{sampling}} = F_{\text{dts}}/8$ ，N=8； 0010：采样频率 $F_{\text{sampling}} = F_{\text{ck_int}}$ ，N=4；	0000b

			1010: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/16$, $N=5$; 0011: 采样频率 $F_{\text{sampling}} = f = F_{\text{ck_int}}$, $N=8$; 1011: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/16$, $N=6$; 0100: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/2$, $N=6$; 1100: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/16$, $N=8$; 0101: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/2$, $N=8$; 1101: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/32$, $N=5$; 0110: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/4$, $N=6$; 1110: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/32$, $N=6$; 0111: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/4$, $N=8$; 1111: 采样频率 $F_{\text{sampling}} = F_{\text{dts}}/32$, $N=8$ 。	
[11:10]	IC2PSC[1:0]	RW	比较捕获通道 2 预分频配置域, 这 2 位定义了比较捕获通道 2 的预分频系数。一旦 CC1E=0, 则预分频器复位。 00: 无预分频器, 捕获输入口上检测到的每一个边沿都触发一次捕获; 01: 每 2 个事件触发一次捕获; 10: 每 4 个事件触发一次捕获; 11: 每 8 个事件触发一次捕获。	00b
[9:8]	CC2S[1:0]	RW	比较捕获通道 2 输入选择域, 这 2 位定义通道的方向(输入/输出), 及输入脚的选择。 00: 比较捕获通道 1 通道被配置为输出; 01: 比较捕获通道 1 通道被配置为输入, IC1 映射在 TI1 上; 10: 比较捕获通道 1 通道被配置为输入, IC1 映射在 TI2 上; 11: 比较捕获通道 1 通道被配置为输入, IC1 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时(由 TS 位选择)。 <i>注: CC1S 仅在通道关闭时(CC1E 为 0)才是可写的。</i>	00b
[7:4]	IC1F[3:0]	RW	输入捕获滤波器 1 配置域。	0
[3:2]	IC1PSC[1:0]	RW	比较捕获通道 1 预分频配置域。	0
[1:0]	CC1S[1:0]	RW	比较捕获通道 1 输入选择域。	0

15.4.8 比较/捕获控制寄存器 2 (TIMx_CHCTLR2) (x=2/3/4)

偏移地址: 0x1C

通道可用于输入(捕获模式)或输出(比较模式), 通道的方向由相应的 CCxS 位定义。该寄存器其它位的作用在输入和输出模式下不同。OCxx 描述了通道在输出模式下的功能, ICxx 描述了通道在输入模式下的功能。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
OC4CE		OC4M[2:0]			OC4PE	OC4FE	CC4S[1:0]		OC3CE		OC3M[2:0]			OC3PE	OC3FE	CC3S[1:0]	
IC4F[3:0]					IC4PSC[1:0]				IC3F[3:0]					IC3PSC[1:0]			

比较模式（引脚方向为输出）：

位	名称	访问	描述	复位值
15	OC4CE	RW	比较捕获通道 4 清零使能位。	0
[14:12]	OC4M[2:0]	RW	比较捕获通道 4 模式设置域。	0
11	OC4PE	RW	比较捕获寄存器 4 预装载使能位。	0
10	OC4FE	RW	比较捕获通道 4 快速使能位。	0
[9:8]	CC4S[1:0]	RW	比较捕获通道 4 输入选择域。	0
7	OC3CE	RW	比较捕获通道 3 清零使能位。	0
[6:4]	OC3M[2:0]	RW	比较捕获通道 3 模式设置域。	0
3	OC3PE	RW	比较捕获寄存器 3 预装载使能位。	0
2	OC3FE	RW	比较捕获通道 3 快速使能位。	0
[1:0]	CC3S[1:0]	RW	比较捕获通道 3 输入选择域。	0

捕获模式（引脚方向为输入）：

位	名称	访问	描述	复位值
[15:12]	IC4F[3:0]	RW	输入捕获滤波器 4 配置域。	0
[11:10]	IC4PSC[1:0]	RW	比较捕获通道 4 预分频配置域。	0
[9:8]	CC4S[1:0]	RW	比较捕获通道 4 输入选择域。	0
[7:4]	IC3F[3:0]	RW	输入捕获滤波器 3 配置域。	0
[3:2]	IC3PSC[1:0]	RW	比较捕获通道 3 预分频配置域。	0
[1:0]	CC3S[1:0]	RW	比较捕获通道 3 输入选择域。	0

15.4.9 比较/捕获使能寄存器（TIMx_CCER）（x=2/3/4）

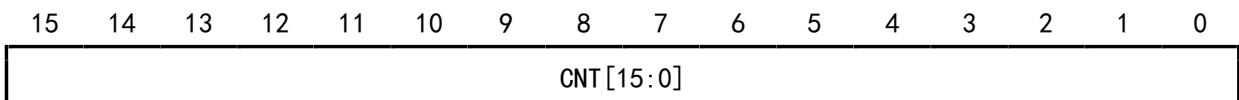
偏移地址：0x20

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	CC4P	CC4E	Reserved	CC3P	CC3E	Reserved	CC2P	CC2E	Reserved	CC1P	CC1E	Reserved	Reserved	Reserved	Reserved

位	名称	访问	描述	复位值
[15:14]	Reserved	RO	保留。	0
13	CC4P	RW	比较捕获通道 4 输出极性设置位。	0
12	CC4E	RW	比较捕获通道 4 输出使能位。	0
[11:10]	Reserved	RO	保留。	0
9	CC3P	RW	比较捕获通道 3 输出极性设置位。	0
8	CC3E	RW	比较捕获通道 3 输出使能位。	0
[7:6]	Reserved	RO	保留。	0
5	CC2P	RW	比较捕获通道 2 输出极性设置位。	0
4	CC2E	RW	比较捕获通道 2 输出使能位。	0
[3:2]	Reserved	RO	保留。	0
1	CC1P	RW	比较捕获通道 1 输出极性设置位。	0
0	CC1E	RW	比较捕获通道 1 输出使能位。	0

15.4.10 通用定时器的计数器 (TIMx_CNT) (x=2/3)

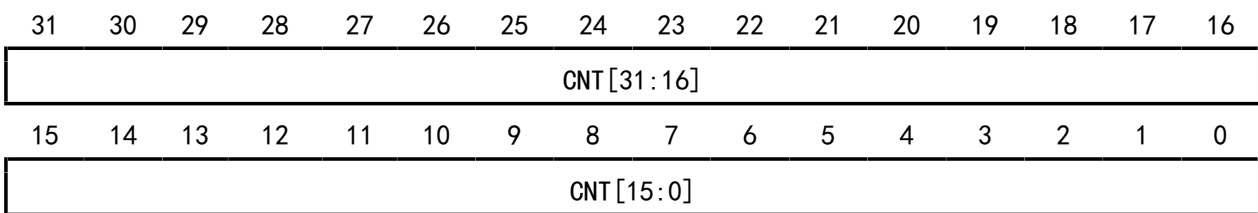
偏移地址: 0x24



位	名称	访问	描述	复位值
[15:0]	CNT[15:0]	RW	定时器的计数器的实时值。	0

15.4.11 通用定时器的计数器 (TIMx_CNT) (x=4)

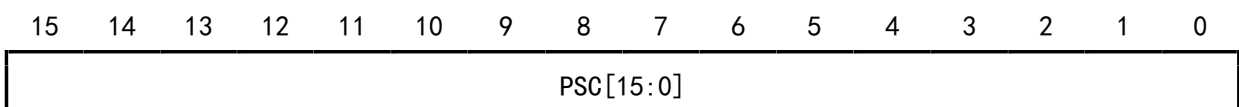
偏移地址: 0x24



位	名称	访问	描述	复位值
[31:0]	CNT[31:0]	RW	定时器的计数器的实时值。	0

15.4.12 计数时钟预分频器 (TIMx_PSC) (x=2/3/4)

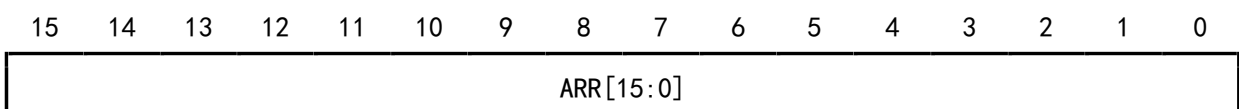
偏移地址: 0x28



位	名称	访问	描述	复位值
[15:0]	PSC[15:0]	RW	定时器的预分频器的分频系数; 计数器的时钟频率等于分频器的输入频率/(PSC+1)。	0

15.4.13 自动重装值寄存器 (TIMx_ARRLR) (x=2/3)

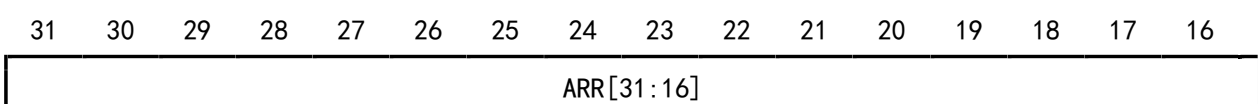
偏移地址: 0x2C



位	名称	访问	描述	复位值
[15:0]	ARR[15:0]	RW	ATRLR[15:0]的值将会被装入计数器, ATRLR 何时动作和更新请阅读 15.2.4 节; ATRLR 为空时, 计数器停止。	0xFFFF

15.4.14 自动重装值寄存器 (TIMx_ARRLR) (x=4)

偏移地址: 0x2C



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

位	名称	访问	描述	复位值
[31:0]	ARR[31:0]	RW	ATRLR[31:0]的值将会被装入计数器, ATRLR 何时动作和更新请阅读 15.2.4 节; ATRLR 为空时, 计数器停止。	0xFFFFFFFF

15.4.15 比较/捕获寄存器 1 (TIMx_CH1CVR) (x=2/3)

偏移地址: 0x34

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															LEVEL1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1[15:0]															

位	名称	访问	描述	复位值
[31:17]	Reserved	RO	保留。	0
16	LEVEL1	RO	捕获值对应的电平指示 bit	0
[15:0]	CCR1[15:0]	RW	比较捕获寄存器通道 1 的值。	0

15.4.16 比较/捕获寄存器 1 (TIMx_CH1CVR) (x=4)

偏移地址: 0x34

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR1[31:16]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1[15:0]															

位	名称	访问	描述	复位值
[31:0]	CCR1[31:0]	RW	比较捕获寄存器通道 1 的值。	0

15.4.17 比较/捕获寄存器 2 (TIMx_CH2CVR) (x=2/3)

偏移地址: 0x38

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															LEVEL2
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2[15:0]															

位	名称	访问	描述	复位值
[31:17]	Reserved	RO	保留。	0
16	LEVEL2	RO	捕获值对应的电平指示 bit	0

[15:0]	CCR2[15:0]	RW	比较捕获寄存器通道 2 的值。	0
--------	------------	----	-----------------	---

15.4.18 比较/捕获寄存器 2 (TIMx_CH2CVR) (x=4)

偏移地址: 0x38

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR2[31: 16]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2[15: 0]															

位	名称	访问	描述	复位值
[31:0]	CCR2[15:0]	RW	比较捕获寄存器通道 2 的值。	0

15.4.19 比较/捕获寄存器 3 (TIMx_CH3CVR) (x=2/3)

偏移地址: 0x3C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															LEVEL3
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR3[15: 0]															

位	名称	访问	描述	复位值
[31:17]	Reserved	RO	保留。	0
16	LEVEL3	RO	捕获值对应的电平指示 bit	0
[15:0]	CCR3[15:0]	RW	比较捕获寄存器通道 3 的值。	0

15.4.20 比较/捕获寄存器 3 (TIMx_CH3CVR) (x=4)

偏移地址: 0x3C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR3[31: 16]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR3[15: 0]															

位	名称	访问	描述	复位值
[31:0]	CCR3[31:0]	RW	比较捕获寄存器通道 3 的值。	0

15.4.21 比较/捕获寄存器 4 (TIMx_CH4CVR) (x=2/3)

偏移地址: 0x40

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															LEVEL4
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

CCR4[15:0]

位	名称	访问	描述	复位值
[31:17]	Reserved	R0	保留。	0
16	LEVEL4	R0	捕获值对应的电平指示 bit	0
[15:0]	CCR4[15:0]	RW	比较捕获寄存器通道 4 的值。	0

15.4.22 比较/捕获寄存器 4 (TIMx_CH4CVR) (x=4)

偏移地址: 0x40

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR4[31:16]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR4[15:0]															

位	名称	访问	描述	复位值
[31:0]	CCR4[31:0]	RW	比较捕获寄存器通道 4 的值。	0

15.4.23 DMA 控制寄存器 (TIMx_DMACFGR) (x=2/3/4)

偏移地址: 0x48

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

位	名称	访问	描述	复位值
[15:13]	Reserved	R0	保留。	0
[12:8]	DBL[4:0]	RW	DMA 连续传送的长度, 实际值为此域的值+1。	0
[7:5]	Reserved	R0	保留。	0
[4:0]	DBA[4:0]	RW	这些位定义了 DMA 在连续模式下从控制寄存器 1 所在地址的偏移量。	0

15.4.24 连续模式的 DMA 地址寄存器 (TIMx_DMAADR) (x=2/3/4)

偏移地址: 0x4C

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

位	名称	访问	描述	复位值
[31:0]	DMAB[31:0]	RW	连续模式下, DMA 的地址。	0

第 16 章 低功耗定时器 (LPTIM)

LPTIM 是一个 16 位上行计数的定时器。LPTIM 具有多种可选的时钟源，使得 LPTIM 能在除待机模式外的所有电源模式下运行。LPTIM 在没有内部时钟源的情况下也能运行，依此可以将 LPTIM 当作“脉冲计数器”使用。除此之外，LPTIM 还能将系统从低功耗模式唤醒，所以 LPTIM 很适合以极低的功耗实现“超时功能”。

16.1 主要特征

- 16 位上行计数器
- 3 位预分频器，支持 8 种分频系数（1、2、4、8、16、32、64、128）
- 可选时钟源

内部时钟源: LSE、LSI、HSI 或 PB1 时钟

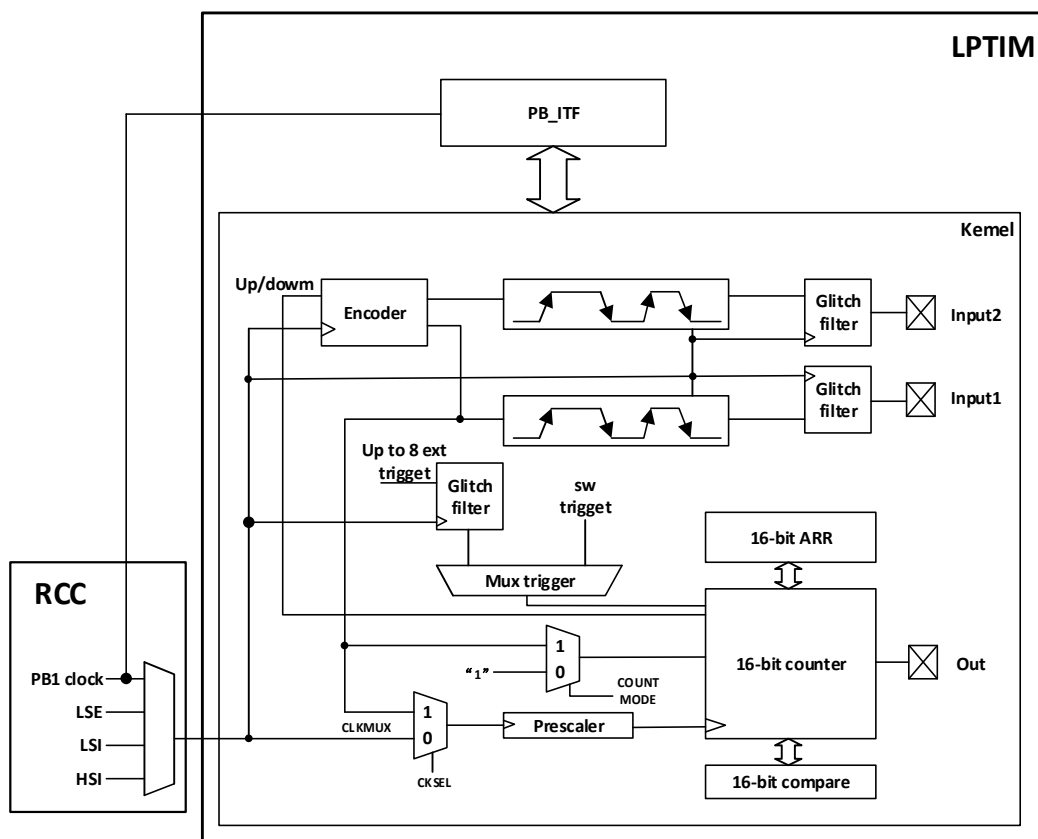
外部时钟源：LPTIM 输入上的外部时钟

- 16 位 ARR 自动重载寄存器
- 16 位比较寄存器
- 连续/单触发模式
- 可选的软件/硬件输入触发器
- 可编程数字干扰滤波器
- 可配置输出 PWM 波
- 可配置 I/O 极性
- 编码器模式

16.2 LPTIM 功能描述

16.2.1 LPTIM 框图

图 16-1 低功耗定时器框图



16.3 LPTIM 触发器映射

LPTIM 外部触发器连接的信息如下所示：

表 16-1 LPTIM 外部触发器连接

TRIGSEL[1:0]	外部触发
LPTIM_TRG_00	LPTIM_ETR (PB6/PB14)
LPTIM_TRG_01	RTC_ALARM
LPTIM_TRG_10	TAMP (PC13)

16.3.1 LPTIM 复位和时钟

LPTIM 模块复位由 RCC_PB1PRSTR 寄存器的 LPTIMRST 位控制，置 0 时无作用，置 1 时复位模块。

LPTIM 模块时钟使能由 RCC_PB1PCENR 寄存器的 LPTIMEN 位控制，置 0 时模块时钟关闭，置 1 时模块时钟开启。

LPTIM 的计数时钟由多个可选的时钟源提供，可分为内部时钟源和外部时钟源。

当使用内部时钟源计数时，内部时钟源可以通过 LPTIM_CFGR 寄存器的 CLKMX_SEL 位中对 PB1、LSI、LSE、HSI 四种时钟源进行选择。此外，LPTIM 可以使用外部输入 LPTIM_CH1 (PB5/PB12) 上注入的外部时钟信号来计时。

表 16-2 LPTIM 内部时钟源

寄存器 LPTIM_CFGR[26:25]	时钟源
00	TIM_CLK (来源于 PB1_CLK)
01	HSI_CLK
10	LSE_CLK
11	LSI_CLK

当使用外部时钟源计数时，LPTIM 可能以以下两种配置之一运行：

第一种配置是由外部信号对 LPTIM 提供时钟，内部时钟信号同时由可配置的内部时钟源 (PB1、LSI、LSE、HSI) 等提供。

第二种配置是 LPTIM 仅由外部时钟源通过其外部输入通道 1，该配置在进入低功率模式后所有内部时钟源关闭时，实现脉冲计数器功能或超时功能。

表 16-3 LPTIM 外部时钟源

寄存器 LPTIM_CFGR[2:1]	时钟源
00	LPTIM_CH1 (PB5/PB12)
01	~LPTIM_CH1 (PB5/PB12)
其他	无

对 CKSEL 和 COUNTMODE 位进行编程可以控制 LPTIM 时钟是使用外部时钟源还是内部时钟源。当配置为选择外部时钟源时，CKPOL 位用于选择外部时钟信号有效边沿。如果有效边沿设置为上升沿和下降沿（双边沿），则还要提供内部时钟信号。在这种情况下，内部时钟信号频率应该比外部时钟信号频率高四倍。

16.3.2 滤波器

LPTIM 输入无论是外部输入还是内部输入，都受到数字滤波器的保护，以防止任何毛刺和噪声干扰在 LPTIM 内部传播，进而避免产生意外计数或触发。

在激活数字滤波器之前，应先向 LPTIM 提供内部时钟源，保证过滤器正常运行。

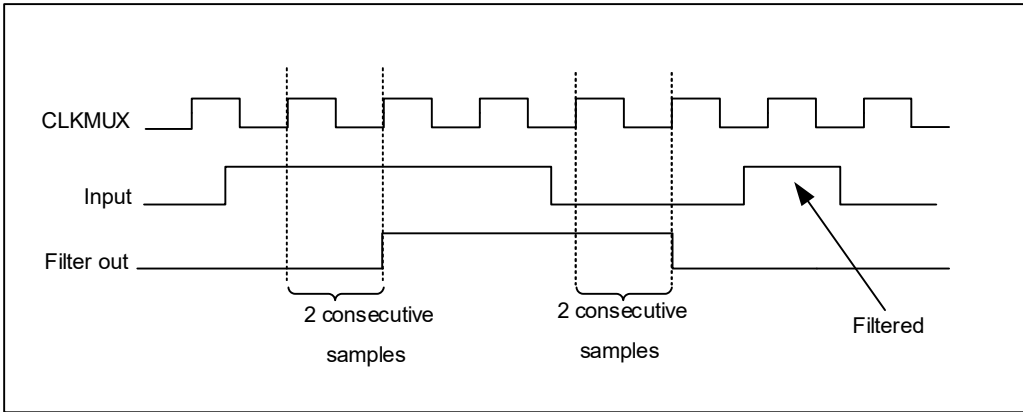
数字滤波器分为两组：

一组为数字滤波器保护 LPTIM 外部输入，由 CKFLT 位控制数字滤波器的灵敏度；另一组数字滤

波器保护 LPTIM 内部触发输入，由 TRGFLT 位控制数字滤波器的灵敏度。

滤波器的灵敏度会影响相同的连续采样的数量,在其中一个 LPTIM 输入上检测到此类连续的采样时，才能将某信号电平变化视为有效切换。下图给出了编程 2 个连续采样时，干扰滤波器行为的示例。

图 16-2 干扰滤波器时序框图



注：在不使用内部时钟信号的时候，数字滤波器的停用必须通过将 CKFLT 和 TRGFLT 位清零来实现，并使用外部模拟滤波器避免 LPTIM 外部输入带来的干扰。

16.3.3 预分频器

LPTIM 16 位计数器前面要有一个可配置的 2ⁿ 预分频器。由 PRESC[2:0] 3 位字段控制预分频器的分频比，表 16-2 列出了所有可能的分频比

表 16-4 预分频器的分频比

PRESC[2:0]	分频比
000	1
001	1/2
010	1/4
011	1/8
100	1/16
101	1/32
110	1/64
111	1/128

16.3.4 触发器多路复用器

LPTIM 计数器有两种启动方式，一是通过软件启动，二是在检测到触发输入中一个以上的有效边沿后启动。LPTIM 触发方式由 TRIGEN[1:0] 控制，触发源由 TRIGSEL[1:0] 位控制。

表 16-5 触发方式

TRIGEN[1:0]	触发方式
00	无效
01	上升沿
10	下降沿
11	双边沿

表 16-6 触发源

TRIGSEL [1:0]	触发源
00	LPTIM_ETR (PB6/PB14)
01	RTC_ALARM
10	TAMP (PC13)
11	无效

16.3.5 操作模式

LPTIM 具有两种操作模式：

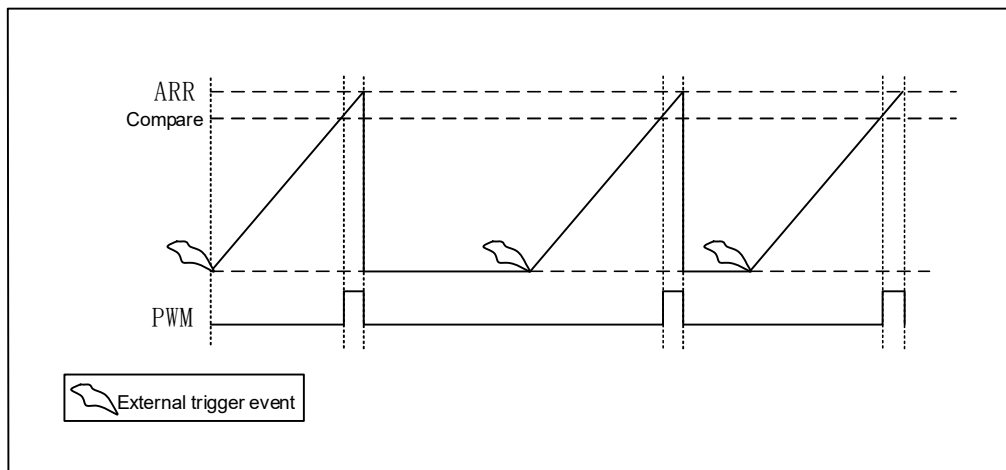
连续模式：计时器自由运行，从触发事件启动，直到计时器被禁用才停止。

单触发模式：计时器从触发事件启动，当达到 ARR 值时停止。

在单触发模式中，要启用单次计数，必须将 SNGSTRT 位置 1，一个新的触发事件将重新启动计时器，在计数器启动之后至计数器达到 ARR 之前发生的任何触发事件都将被丢失。

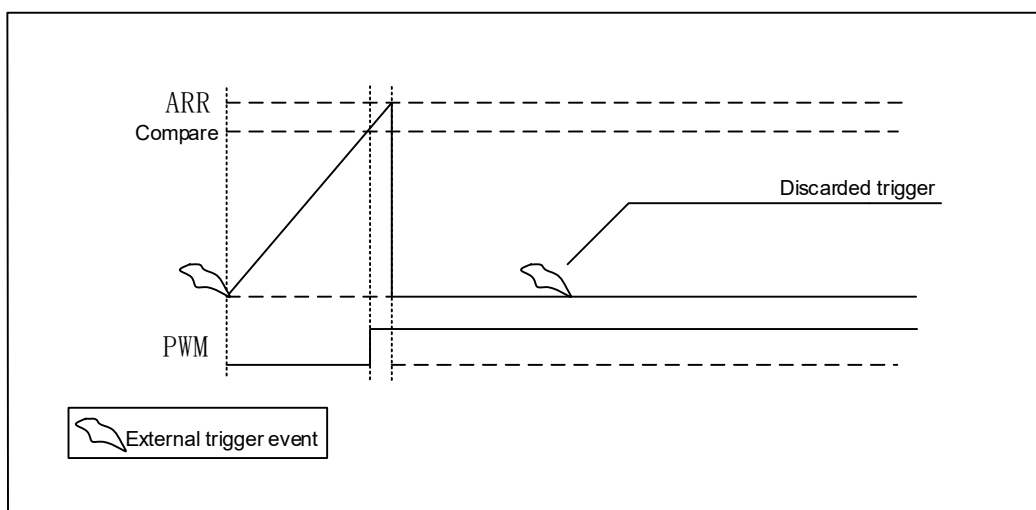
在选择用外部触发时，则在置位 SNGSTRT 位后以及计数器寄存器停止后到达的每个外部触发器事件将启动计数器进行新的一次计数循环。

图 16-3 LPTIM 输出波形，单次计数模式配置



激活单触发设置模式，应该注意的是，当 LPTIM_CFGR 寄存器中的 WAVE 位字段被设置时，单触发模式被激活，在这种情况下，计数器只在第一个之后启动一次触发器，并且丢弃任何后续的触发器事件。

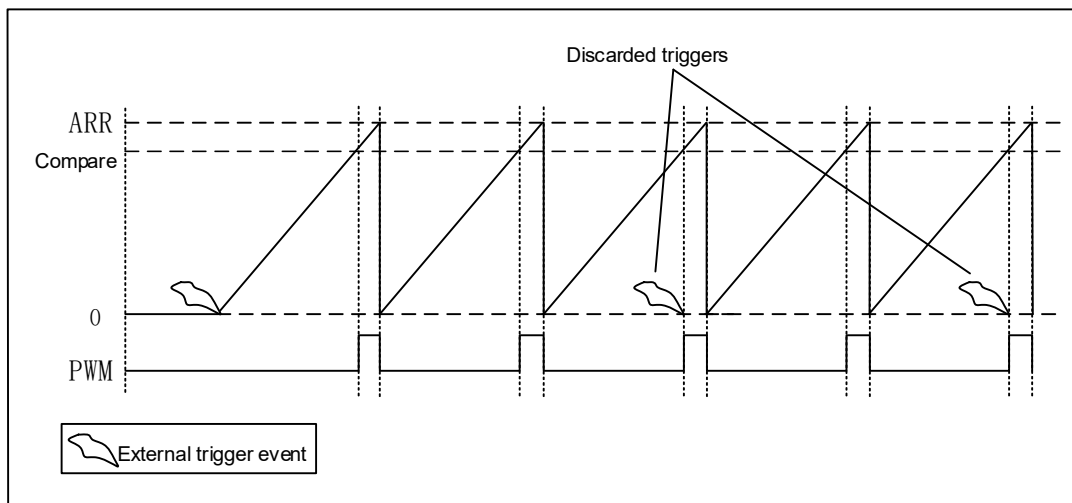
图 16-4 LPTIM 输出波形，单次计数模式配置且激活置 1



连续模式，要启用连续计数，必须设置 CNTSTRT 位，如果选择了外部触发器，则在设置 CNTSTRT 之后到达的外部触发器事件将启动计数器进行连续计数。任何后续的外部触发事件都将被丢弃，在

软件启动的情况下 (TRIGEN=00) 设置 CNTSTRT 将启动计数器连续计数。

图 16-5 LPTIM 输出波形、连续计数模式配置



SNGSTRT 和 CNTSTRT 位只能在定时器启用时设置 (ENABLE 位设置为 1)，可以将更改 LPTIM 计数器模式，如果先前选择了连续模式，则设置 SNGSTRT 将把 LPTIM 切换到单触发模式，计数器达到 ARR 值停止计数，如果先前选择了单触发模式，则设置 CNTSTRT 会将 LPTIM 切换到连续模式。计数器达到 ARR 值时立即重新启动。

16.3.6 超时功能

在一个选定的触发输入上检测到有效边沿可用于重置计数器，此功能通过 TIMOUT 位进行控制。第一个触发事件将启动计数器，任何连续的触发事件都将重置计数器并且计时器将重新启动，可以实现低功耗超时功能，如果没有触发事件发生，则 MCU 被比较匹配事件唤醒。

16.3.7 波形产生

两个 16 位寄存器 LPTIM_ARR 和 LPTIM_CMP，用于在 LPTIM 输出上生成几个不同的波形。计时器可以生成以下波形：

- (1) PWM 模式：一旦 LPTIM_CNT 中的计数器值超过 LPTIM_CMP 中的比较值，就设置 LPTIM 输出。一旦 LPTIM_ARR 和 LPTIM_CNT 寄存器值相等，LPTIM 输出就会重置。
- (2) 单脉冲模式：输出波形与第一个脉冲的 PWM 模式相似，然后永久复位。
- (3) 一次设置模式：输出波形与单脉冲模式相似，只是输出保持在最后一个信号电平（取决于输出配置的极性）。

上述模式要求 LPTIM_ARR 寄存器值严格大于 LPTIM_CMP 寄存器值。

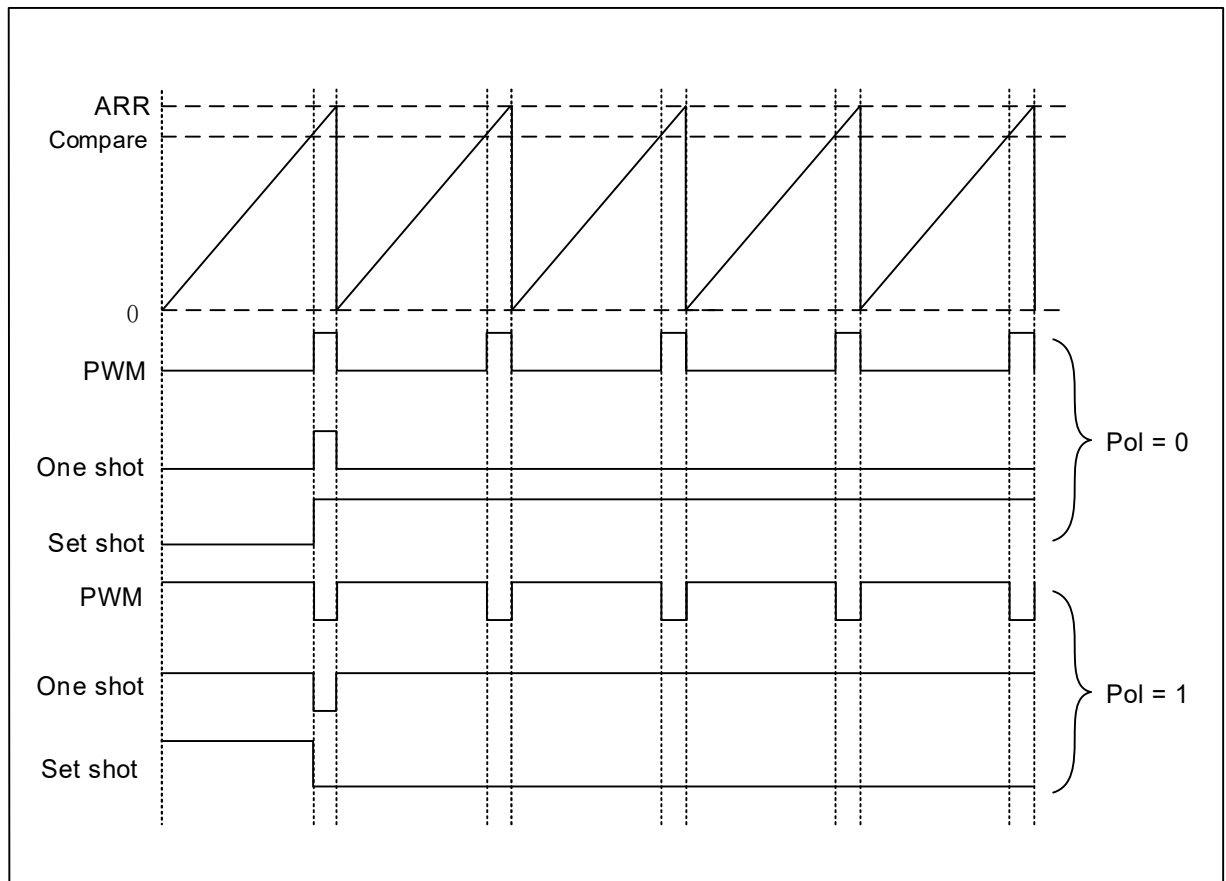
LPTIM 输出波形可以通过 WAVE 位配置如下：

- (1) 将 WAVE 位重置为 0 将强制 LPTIM 生成 PWM 波形或单脉冲波形，具体取决于设置的位：CNTSTRT 或 SNGSTRT。
- (2) 将 WAVE 位设置为 1 将强制 LPTIM 生成一次设置模式。

WAVPOL 位控制 LPTIM 输出极性，更改会立即生效，因此在重新配置极性后，甚至在启用计时器之前，输出默认值也会立即更改。

生成的信号的频率高达 LPTIM 时钟频率 2 分频。图 16-6 给出了可能在 LPTIM 输出上生成的三种波形。此外，此图还显示了通过 WAVPOL 位更改极性所产生的效果。

图 16-6 生成波形



16.3.8 寄存器更新

LPTIM_ARR 寄存器和 LPTIM_CMP 寄存器在 PB 总线写入操作之后立即更新，或者如果计时器已经启动，则在当前周期结束时更新。

PRELOAD 位控制 LPTIM_ARR 和 LPTIM_CMP 寄存器的更新方式：

当 PRELOAD 位设置为 0 时，LPTIM_ARR 和 LPTIM_CMP 寄存器将在任何写入访问后立即更新。

当 PRELOAD 位设置为 1 时，如果计时器已经启动，则在当前周期结束时更新 LPTIM_ARR 和 LPTIM_CMP 寄存器。

LPTIM PB 接口和 LPTIM 逻辑使用不同的时钟，因此在 PB 写入和这些值可用于计数器比较器之间存在一些延迟，在这个延迟期内，必须避免对这些寄存器进行任何额外的写入。

LPTIM_ISR 寄存器中的 ARROK 标志和 CMPOK 标志分别指示何时完成对 LPTIM_ARR 寄存器和 LPTIM_CMP 寄存器的写入操作。

在对 LPTIM_ARR 寄存器或 LPTIM_CMP 寄存器进行写入之后，只有在前一次写入操作完成时，才能对同一寄存器执行新的写入操作。

在设置 ARROK 标志或 CMPOK 标志之前的任何连续写入都将导致不可预测的结果。

16.3.9 计数器模式

LPTIM 计数器可用于对 LPTIM_CH1 上的外部事件进行计数，也可用于对内部时钟周期进行计数。CKSEL 和 COUNTMODE 位控制哪个源将用于更新计数器。

如果 LPTIM 被配置为 LPTIM_CH1 上的外部事件进行计数，则可以根据写入 CKPOL[1:0] 位的值在上升沿，下降沿或双边沿之后更新计数。

根据 CKSEL 和 COUNTMODE 值，可以选择以下计数模式：

(1) CKSEL=0：LPTIM 时钟由内部时钟源计数

COUNTMODE=0，LPTIM 被配置为由内部时钟源计数时，并且 LPTIM 计数器被配置为跟随每个内部时钟

脉冲之后更新。

COUNTMODE=1, 使用提供给 LPTIM 的内部时钟对 LPTIM_CH1 上的数据进行采样。因此为了不遗漏任何事件, LPTIM_CH1 上的信号的变化频率不应超过 LPTIM 内部时钟的频率。

(2) CKSEL=1: LPTIM 由外部时钟源计数时, COUNTMODE 的值无影响。

在此配置中, LPTIM 不需要内部时钟源 (除非启用数字滤波器), 在 LPTIM_CH1 上注入的信号用作 LPTIM 的系统时钟。此配置适用于未启用内置振荡器的工作模式。

对于此配置, LPTIM 计数器可以在 LPTIM_CH1 时钟信号的上升沿或下降沿更新, 但不能在双边沿更新。由于在 LPTIM_CH1 注入的信号也用于 LPTIM 计数时, 因此在计数器递增之前会有一些初始延迟 (在启用 LPTIM 之后)。准确来说, 在使能 LPTIM 之后, LPTIM 外部 LPTIM_CH1 的前五个有效边沿将丢失。

16.3.10 定时器启用

位于 LPTIM_CR 寄存器中的 ENABLE 位用于启用/禁用 LPTIM 逻辑, 在设置 ENABLE 位后, 在实际启用 LPTIM 之前需要两个计数器时钟的延迟。

16.3.11 编码器模式

该模式允许处理来自用于检测旋转元件的位置的正交编码器的信号。编码器接口模式只是作为一个有方向选择的外部时钟。这意味着计数器仅在 0 和 LPTIM_ARR 寄存器中的自动重载值之间连续计数。因此, 在启动之前必须配置 LPTIM_ARR 根据两个外部输入信号 LPTIM_CH1 和 LPTIM_CH2 生成时钟信号以对 LPTIM 计数器进行计时。这两个信号之间的相位决定了计数方向。

只有当 LPTIM 由内部时钟源计数时, 编码器模式才可用。LPTIM_CH1 和 LPTIM_CH2 上的信号频率不得超过 LPTIM 内部时钟的频率 4 分频。满足以上条件才能确保 LPTIM 正常运行。

方向变化由 LPTIM_ISR 寄存器中的两个向上和向下标志发出信号。此外, 如果通过 DOWNIE 位启用, 则可以为两个方向改变事件生成中断。

要激活编码器模式, 必须将 ENC 位设置为 1, 必须首先将 LPTIM 配置为连续模式。

当编码器模式处于激活状态, LPTIM 计数器会根据增量编码器的速度和方向自动修改。因此, 它的内容总是代表编码器的位置。由向上和向下标志指示计数方向, 对应于编码器转子的旋转方向。根据 CKPOL[1:0] 位配置的边缘灵敏度, 得到了以下可能的组合方案, LPTIM_CH1 和 LPTIM_CH2 不同时切换。

表 16-7 配置的边缘灵敏度组合方案

活动边沿 (CKPOL[1:0])	相反信号的电平 (LPTIM_CH1 对应的输入是 LPTIM_CH2, LPTIM_CH2 对应的输入是 LPTIM_CH1,)	LPTIM_CH1		LPTIM_CH2	
		上升	下降	上升	下降
上升沿 (00)	高	递减	不计数	递增	不计数
	低	递增	不计数	递减	不计数
下降沿 (01)	高	不计数	递增	不计数	递减
	低	不计数	递减	不计数	递增
双边沿 (10)	高	递减	递增	递增	递减
	低	递增	递减	递减	递增
11		无效			

16.3.12 调试模式

当 MCU 进入调式模式（内核停止），LPTIM 继续正常工作。

16.3.13 LPTIM 低功耗模式

表 16-8 低功耗模式对 LPTIM 的影响

模式	描述
睡眠模式	无影响，LPTIM 中断会导致设备退出睡眠模式
停止模式	LPTIM 外围设备在由 LSE 或 LSI 计时时处于活动状态，LPTIM 中断导致设备退出停止模式
待机模式	LPTIM 外围设备已断电，必须在退出待机模式后重新初始化

16.4 LPTIM 中断

如果通过 LPTIM_IER 寄存器使能以下事件，则会产生中断/唤醒：

- 比较匹配
- 自动重载匹配
- 外部触发事件
- 自动重载寄存器写入完成
- 比较寄存器写入完成
- 方向改变（编码器模式）

表 16-9 中断事件

中断事件	描述
比较匹配	当计数器寄存器 LPTIM_CNT 的值与比较寄存器 LPTIM_CMP 的值相等时会产生中断标志
自动重载匹配	当计数器寄存器 LPTIM_CNT 的值与自动重载寄存器 LPTIM_ARR 的值相等时会产生中断标志
外部触发事件	当检测到外部触发事件时会产生中断标志
自动重载寄存器写入完成	对 LPTIM_ARR 寄存器的写入操作完成时会产生中断标志
比较寄存器写入完成	对 LPTIM_CMP 寄存器的写入操作完成时会产生中断标志
计数方向改变	在编码器模式下使用，两个中断标志嵌入到信号的方向改变： 向上标志：表示向上计数方向改变； 向下标志：表示向下计数方向改变。

16.5 寄存器描述

表 16-10 LPTIM 相关寄存器列表

名称	访问地址	描述	复位值
R32_LPTIM_ISR	0x40007C00	中断状态寄存器	0x00000000
R32_LPTIM_ICR	0x40007C04	中断清除寄存器	0x00000000
R32_LPTIM_IER	0x40007C08	中断使能寄存器	0x00000000
R32_LPTIM_CFGR	0x40007C0C	配置寄存器	0x00000000
R32_LPTIM_CR	0x40007C10	控制寄存器	0x00000000
R32_LPTIM_CMR	0x40007C14	比较寄存器	0x00000000
R32_LPTIM_ARR	0x40007C18	自动重载寄存器	0x00000001
R32_LPTIM_CNT	0x40007C1C	计数寄存器	0x00000000

16.5.1 中断状态寄存器 (LPTIM_ISR)

偏移地址: 0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								DIR _SY NC	DOW N	UP	ARR OK	CMP OK	EXT TRIG	ARR M	CMP M

位	名称	访问	描述	复位值
[31:8]	Reserved	R0	保留。	0
7	DIR_SYNC	R0	编码器模式下, 计数器计数方向 0: 向下计数 1: 向上计数 该位只在编码器模式下有效, 即 ENC 位置 1, 使用前还需将 DIR_EXTEN 置 1 使能计数方向功能。	0
6	DOWN	R0	向下计数 1: 计数方向从上到下 0: 无效 在编码器模式下, 该位由硬件设置, 以通知应用程序计数器方向从上到下变化, 为 1 之后可以通过向 LPTIM_ICR[6] 位写 1 来清除。	0
5	UP	R0	向上计数 1: 计数方向从下到上 0: 无效 在编码器模式下, 该位由硬件设置, 以通知应用程序计数器方向从下到上变化, 为 1 之后可以通过向 LPTIM_ICR[5] 位写 1 来清除。	0
4	ARROK	R0	自动重加载寄存器数据更新成功 1: 数据更新成功 0: 无效 该位由硬件设置, 以通知应用程序 PB 总线对 LPTIM_ARR 的写入操作已成功完成, 为 1 之后可以通过向 LPTIM_ICR[4] 位写 1 来清除。	0
3	CMPOK	R0	比较寄存器数据更新成功 1: 数据更新成功 0: 无效 该位由硬件设置, 以通知应用程序 PB 总线对 LPTIM_CMR 的写入操作已成功完成, 为 1 之后可以通过向 LPTIM_ICR[3] 位写 1 来清除。	0
2	EXTTRIG	R0	外部触发边缘事件 1: 有效边缘输入已发生 0: 无效 该位由硬件设置, 以通知应用程序所选外部触发器上	0

			的有效边缘输入已发生，如果由于计时器已经启动而忽略触发器，则此标志为未设置。为 1 之后可以通过向 LPTIM_ICR[2] 位写 1 来清除。	
1	ARRM	RO	自动重加载寄存器数据与 LPTIM_CNT 寄存器的数据匹配 1: 匹配成功 0: 无效 该位由硬件设置，以通知应用程序 LPTIM_CNT 寄存器的值达到 LPTIM_ARR 寄存器的值，为 1 之后可以通过向 LPTIM_ICR[1] 位写 1 来清除。	0
0	CMPM	RO	比较寄存器的数据与 LPTIM_CNT 寄存器的数据匹配 1: 匹配成功 0: 无效 该位由硬件设置，以通知应用程序 LPTIM_CNT 寄存器的值达到 LPTIM_CMR 寄存器的值，为 1 之后可以通过向 LPTIM_ICR[0] 位写 1 来清除。	0

16.5.2 中断清除寄存器 (LPTIM_ICR)

偏移地址: 0x04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									DOW NCF	UPN CF	ARRO KNCF	CMP OKN CF	EXTT RIGN CF	ARR MNC F	CM PM NC F

位	名称	访问	描述	复位值
[31:7]	Reserved	RO	保留。	0
6	DOWNCF	W1	清除向下标志位 注意：写 0 无效，写 1 清 0 状态寄存器对应位	x
5	UPCF	W1	清除向上标志位 注意：写 0 无效，写 1 清 0 状态寄存器对应位	x
4	ARROKCF	W1	清除自动重加载数据更新标志位 注意：写 0 无效，写 1 清 0 状态寄存器对应位	x
3	CMPOKCF	W1	清除比较器数据更新标志位 注意：写 0 无效，写 1 清 0 状态寄存器对应位	x
2	EXTTRIGCF	W1	清除外部触发器边缘事件标志位 注意：写 0 无效，写 1 清 0 状态寄存器对应位	x
1	ARRMCF	W1	清除自动重加载寄存器匹配标志位 注意：写 0 无效，写 1 清 0 状态寄存器对应位	x
0	CMPMCF	W1	清除比较寄存器匹配标志位 注意：写 0 无效，写 1 清 0 状态寄存器对应位	x

16.5.3 中断使能寄存器 (LPTIM_IER)

偏移地址: 0x08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								DOWNIE	UPIE	ARROKIE	CMPOKIE	EXTTRIGIE	ARRMIE	CMPMIE	

位	名称	访问	描述	复位值
[31:7]	Reserved	RO	保留。	0
6	DOWNIE	RW	向下中断使能: 0: 关闭 1: 开启	0
5	UPIE	RW	向上中断使能: 0: 关闭 1: 开启	0
4	ARROKIE	RW	自动重加载寄存器数据更新成功中断使能 0: 关闭 1: 开启	0
3	CMPOKIE	RW	比较寄存器数据更新成功中断使能 0: 关闭 1: 开启	0
2	EXTTRIGIE	RW	外部触发边缘事件输入中断使能 0: 关闭 1: 开启	0
1	ARRMIE	RW	自动重加载寄存器数据匹配成功中断使能 0: 关闭 1: 开启	0
0	CMPMIE	RW	比较寄存器数据匹配成功中断使能 0: 关闭 1: 开启	0

16.5.4 配置寄存器 (LPTIM_CFGR)

偏移地址: 0x0C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				FORCE_PWM	CLKMX_SEL	ENC	COUNT_MODE	PRELOAD	WAVPOL	WAVE	TIMOUT	TRIGEN	Reserved		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	TRIGSEL		Reserved	PRESC			Reserved	TRGFLT		Reserved	CKFLT		CKPOL		CKSEL

位	名称	访问	描述	复位值
[31:28]	Reserved	RO	保留。	0
27	FORCE_PWM	RW	强制 PWM 输出	0

			0: 无效 1: 强制 PWM 输出高电平	
[26:25]	CLKMX_SEL	RW	LPTIM 计数器内部时钟源选择 00: TIM_CLK (来源于 PB1_CLK) 01: HSI_CLK 10: LSE_CLK 11: LSI_CLK	0
24	ENC	RW	编码器模式 0: 关闭 1: 开启	0
23	COUNTMODE	RW	计数器模式, 选择 LPTIM 使用哪个时钟源为计数器计时 0: 计数器在每个内部时钟脉冲后递增 1: 当 LPTIM 外部输入每个有效脉冲后, 计数器递增	0
22	PRELOAD	RW	寄存器更新模式, 控制 LPTIM_ARR 和 LPTIM_CMP 寄存器更新模式 0: 每次 PB 总线写入访问后更新寄存器 1: 寄存器在当前 LPTIM 周期结束时更新	0
21	WAVPOL	RW	PWM 波形极性 0: 输出反映 LPTIM_ARR 和 LPTIM_CMP 寄存器之间比较结果 1: 输出反映 LPTIM_ARR 和 LPTIM_CMP 寄存器之间比较结果的取反 注意: 如果计数器值大于比较器值比较结果为 1, 否则为 0。	0
20	WAVE	RW	PWM 波形 0: 关闭一次模式 1: 设置一次模式 (一个脉冲波形)	0
19	TIMOUT	RW	超时启用, 控制超时功能 0: 计时器已启动时到达的触发事件被忽略 1: 计时器已启动时到达的触发事件将重置并重新启动计数器	0
[18:17]	TRIGEN	RW	触发启用和极性, 控制 LPTIM 计数器是否由外部触发器启动, 如果选择了外部触发器选项, 则触发器活动边缘有三种配置。 00: 软件触发 (计数器启动由软件启动) 01: 上升沿触发 10: 下降沿触发 11: 双边沿触发	0
[16:15]	Reserved	RO	保留	0
[14:13]	TRIGSEL	RW	触发源选择 00: LPTIM_ETR (PB6/PB14) 01: RTC_ALARM 10: TAMP (PC13) 11: 无效	0
12	Reserved	RO	保留	0

[11:9]	PRESC	RW	时钟预分频器，配置预分频系数 000: 1 分频 001: 2 分频 010: 4 分频 011: 8 分频 100: 16 分频 101: 32 分频 110: 64 分频 111: 128 分频	0
8	Reserved	RO	保留	0
[7:6]	TRGFLT	RW	用于触发器的可配置数字滤波器 TRGFLT 值设置在内部触发器上发生变化时应检测到的连续相等样本的数量，然后将其视为有效的电平转换。必须有内部时钟源才能使用此功能 00: 任何触发器的更改都被视为有效触发器 01: 触发激活电平变化必须稳定至少 2 个时钟周期，然后才视为有效触发器 10: 触发激活电平变化必须稳定至少 4 个时钟周期，然后才视为有效触发器 11: 触发激活电平变化必须稳定至少 8 个时钟周期，然后才视为有效触发器	0
5	Reserved	RO	保留。	0
[4:3]	CKFLT	RW	外部时钟的可配置数字滤波器 CKFLT 值用于设置连续相等样本的数量。当外部时钟信号发生电平变化时应检测到此类连续样本，才会将此电平变化视为有效电平转换。必须有内部时钟源才能使用此功能 00: 任何外部时钟信号电平变化都被视为有效转换 01: 外部时钟信号电平变化必须稳定至少 2 个时钟周期，然后才视为有效 10: 外部时钟信号电平变化必须稳定至少 4 个时钟周期，然后才视为有效 11: 外部时钟信号电平变化必须稳定至少 8 个时钟周期，然后才视为有效	0
[2:1]	CKPOL	RW	如果 LPTIM 选择外部时钟源计数, CKPOL 位用于配置有效边沿: 00: 上升沿用于计数 01: 下降沿用于计数 10: 双边沿用于计数, 当外部时钟信号的双边沿都有效时, LPTIM 也必须由频率至少等于外部时钟频率四倍的内部时钟源进行计时。 11: 无效 如果 LPTIM 配置为编码器模式时 (ENC 位置 1), CKPOL 位用于选择编码器模式: 00: 编码器模式 1 激活 01: 编码器模式 2 激活	0

			10: 编码器模式 3 激活 11: 无效	
0	CKSEL	RW	时钟选择器, 用于选择 LPTIM 使用的时钟 0: LPTIM 使用内部时钟 1: LPTIM 使用外部时钟	0

16.5.5 控制寄存器 (LPTIM_CR)

偏移地址: 0x10

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

位	名称	访问	描述	复位值
[31:5]	Reserved	RO	保留。	0
4	DIR_EXTEN	RW	外部触发计数方向使能 0: 关闭 1: 开启 注意: 编码器模式开启后使用	0
3	OUTEN	RW	PWM 波输出使能 0: 关闭 1: 开启 注意: 在非编码器模式下, 该位使能输出 PWM 波需要将 ENABLE 同时使能才有效。	0
2	CNTSTRT	RW	在连续模式下启动 该位由软件设置, 并由硬件清除, 在软件启动的情况下 (TRIGEN==00), 设置此位以连续模式启动 LPTIM, 如果软件启动被禁用 (TRIGEN!=00) 则设置此位会在检测到外部触发器后立即以连续模式启动 LPTIM。如果在 LPTIM 处于单脉冲计数模式时设置此位, 则 LPTIM 将在计数器达到 ARR 值时不会停止计数。 注意: ENABLE=1 时才可以写	0
1	SNGSTRT	RW	LPTIM 在单触发模式下启动 该位由软件设置, 并由硬件清除, 在软件启动的情况下 (TRIGEN==00), 设置此位以单脉冲模式启动 LPTIM, 如果软件启动被禁用 (TRIGEN!=00) 则设置此位会在检测到外部触发器后立即以单脉冲模式启动 LPTIM。如果在 LPTIM 处于连续计数模式时设置此位, 则 LPTIM 将在计数器达到 ARR 值时停止。 注意: ENABLE=1 时才可以写。	0
0	ENABLE	RW	定时器使能, 当定时器使能位被清 0 时, 将会复位内部逻辑, 并且无法操作 LPT_CR[2:1] 位。 0: 低功耗定时器禁止并且复位内部逻辑	0

			1:低功耗定时器开启	
--	--	--	------------	--

16.6.6 比较寄存器 (LPTIM_CMP)

偏移地址: 0x14

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

位	名称	访问	描述	复位值
[31:16]	Reserved	RO	保留。	0
[15:0]	CMP	RW	比较值, 当计数器的值等于比较寄存器中的值, 定时器比较标志会被置 1, 如果在置 1 之前打开相应的使能信号会产生中断源和低功耗唤醒信号。	0

16.5.7 自动重载寄存器 (LPTIM_ARR)

偏移地址: 0x18

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

位	名称	访问	描述	复位值
[31:16]	Reserved	RO	保留。	0
[15:0]	ARR	RW	计数寄存器重新装载的计数值, 向上计数时如果计数值等于重新装载的计数值, 则计数器从 0 开始计数; 向下计数时如果计数器的值等于 0 则从重新装载的计数值开始计数。	1

16.5.8 计数寄存器 (LPTIM_CNT)

偏移地址: 0x1C

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

位	名称	访问	描述	复位值
[31:16]	Reserved	RO	保留。	0
[15:0]	COUNT	RO	当前定时器计数器寄存器的计数值 当 LPTIM 使用异步时钟运行时, 读取 LPTIM_CNT 寄存器可能返回不可靠的值, 因此, 在这种情况下, 有必	0

			要执行两次连续的读取访问，并验证两个返回值是否相同。应该注意的是， <u>对于可靠的 LPTIM_CNT 寄存器读取访问，两次连续读取必须执行访问并进行比较。</u>	
--	--	--	---	--

第 17 章 通用同步异步收发器 (USART)

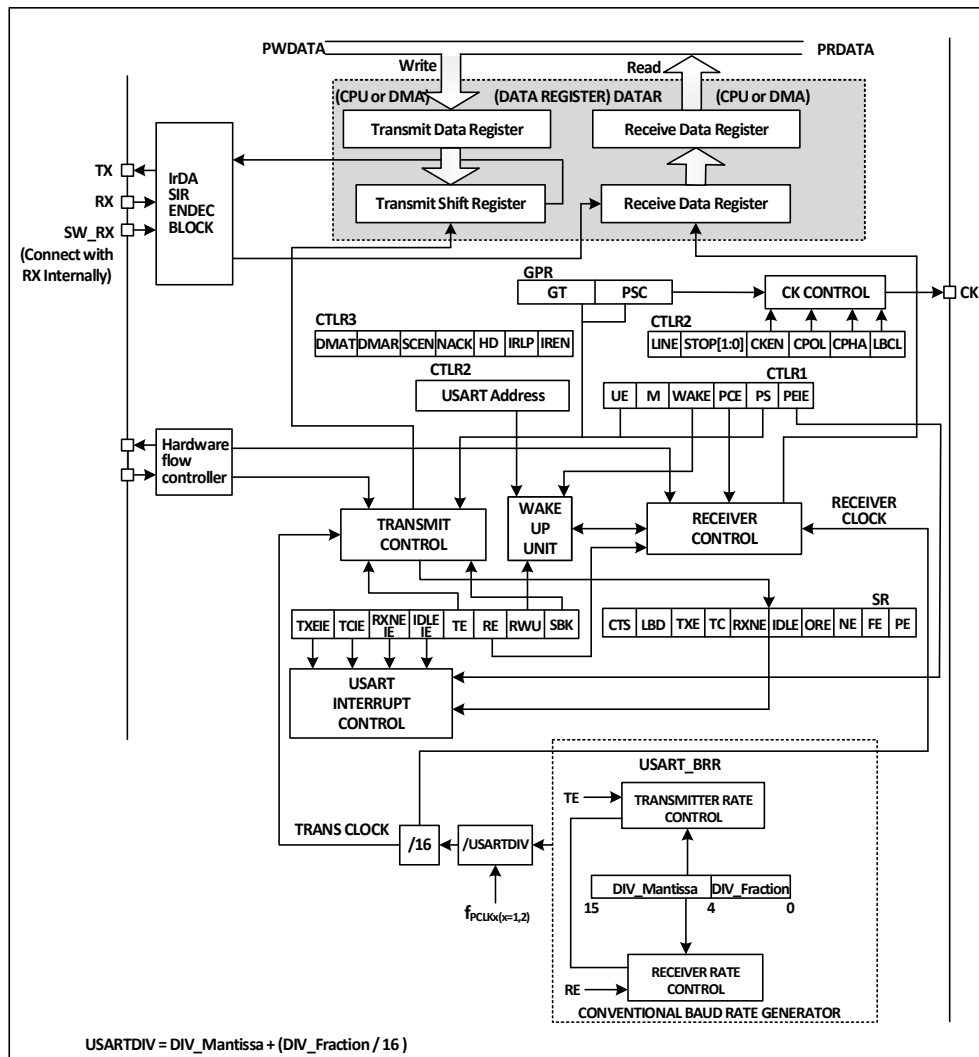
该模块包含 4 个通用同步异步收发器 (USART1/2/3/4)。

17.1 主要特征

- 全双工或半双工的同步或异步通信
- NRZ 数据格式
- 分数波特率发生器，最高 6Mbps
- 可编程数据长度
- 可配置的停止位
- 支持 LIN, IrDA 编码器，智能卡
- 支持 DMA
- 多种中断源

17.2 概述

图 17-1 通用同步/异步收发器的结构框图



当 TE (发送使能位) 置位时，发送移位寄存器里的数据在 TX 引脚上输出，时钟在 CK 引脚上输出。在发送时，最先移出的是最低有效位，每个数据帧都由一个低电平的起始位开始，然后发送器根据 M (字长) 位上的设置发送八位或九位的数据字，最后是数目可配置的停止位。如果配有奇偶

检验位，数据字的最后一位为校验位。在 TE 置位后会发送一个空闲帧，空闲帧是 10 位或 11 位高电平，包含停止位。断开帧是 10 位或 11 位低电平，后跟着停止位。

17.3 波特率发生器

收发器的波特率 = $FCLK / (16 * USARTDIV)$ ，FCLK 是 PBx 的时钟，即 PCLK1 或 PCLK2，USART1 模块使用 PCLK2，其余的使用 PCLK1。USARTDIV 的值是根据 USART_BRR 中的 DIV_M 和 DIV_F 两个域决定的，具体计算的公式为：

$$USARTDIV = DIV_M + (DIV_F / 16)$$

需要注意的是，波特率产生器产生的比特率不一定能刚好生成用户所需要的波特率，这其中可能是存在偏差。除了尽量取接近的值，减小偏差的方法还可以是增大 PBx 的时钟。比如设定波特率为 115200bps 的时，USARTDIV 的值设为 39.0625，在最高频率时可以得到刚好 115200bps 的波特率，但是如果你需要 921600bps 的波特率时，计算的 USARTDIV 是 4.88，但是实际上在 USART_BRR 里填入的值最接近只能是 4.875，实际产生的波特率是 923076bps，误差达到 0.16%。

发送方发出的串口波形传到接收端时，接收方和发送方的波特率是有一定误差的。误差主要来自三个方面：接收方和发送方实际的波特率不一致；接收方和发送方的时钟有误差；波形在线路中产生的变化。外设模块的接收器是有一定接收容差能力的，当以上三个方面产生的总偏差之和小于模块的容差能力极限时，这个总偏差不影响收发。模块的容差能力极限受是否采用分数波特率和 M 位（数据域字长）影响，采用分数波特率和使用 9 位数据域长度会使容差能力极限降低，但不低于 3%。

17.4 同步模式

同步模式使得系统在使用 USART 模块时可以输出时钟信号。在开启同步模式对外发送数据时，CK 引脚会同时对外输出时钟。

开启同步模式的方式是对控制寄存器 2 (R16_USARTx_CTLR2) 的 CLKEN 位置位，但同时需要关闭 LIN 模式、智能卡模式、红外模式和半双工模式，即保证 SCEN、HDSEL 和 IREN 位处于复位状态，这三位在控制寄存器 3 (R16_USARTx_CTLR3) 中。

同步模式使用的要点在于时钟的输出控制。有以下几点需要注意：

USART 模块同步模式只工作在主模式，即 CK 引脚只输出时钟，不接收输入；

只在 TX 引脚输出数据时输出时钟信号；

LBCL 位决定在发送最后一位数据位时是否输出时钟，CPOL 位决定时钟的极性，CPHA 决定时钟的相位，这三个位在控制寄存器 2 (R16_USARTx_CTLR2) 中，这三个位需要在 TE 和 RE 未被使能的情况下设置，具体区别见图 17-2。

接收器在同步模式下只会在输出时钟时采样，需要从设备保持一定的信号建立时间和保持时间，具体见图 17-3。

图 17-2 USART 时钟时序示例 (M=0)

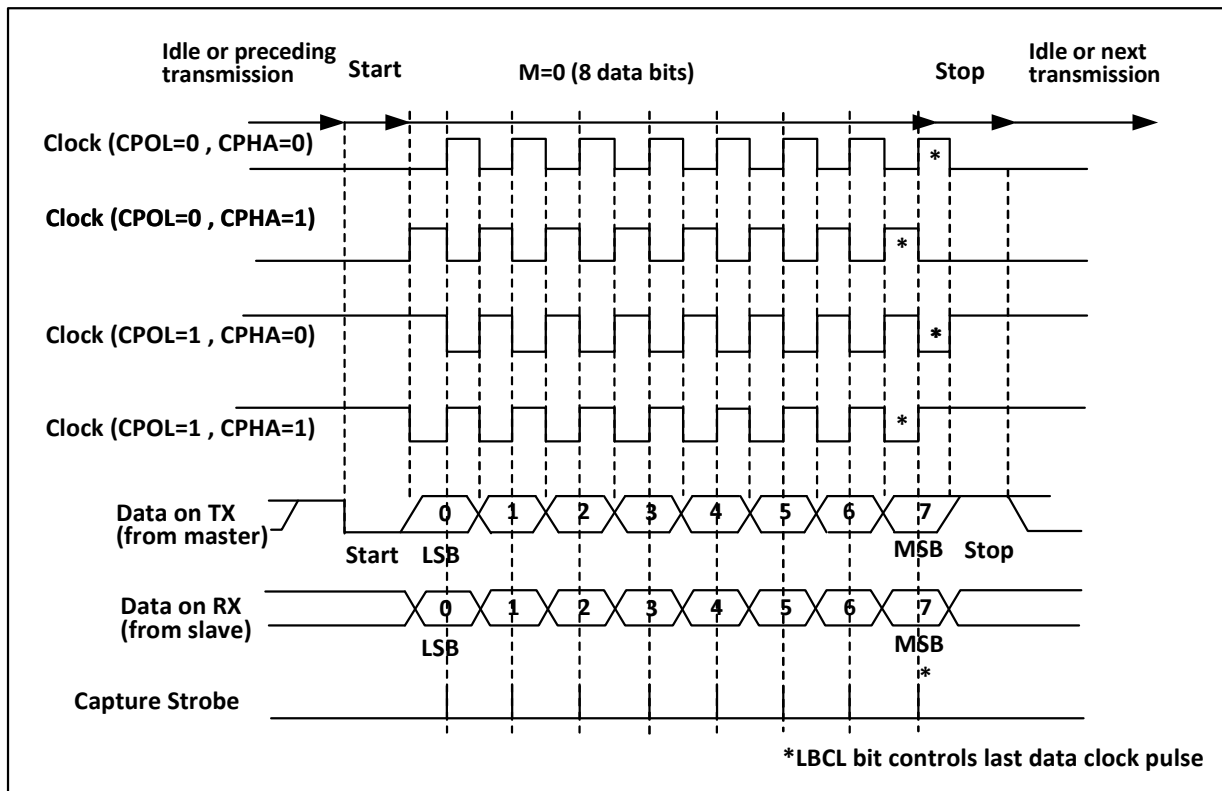
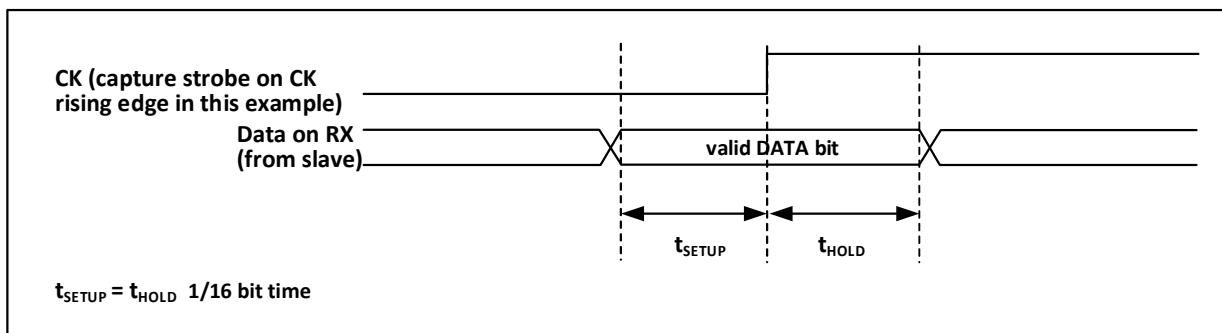


图 17-3 数据采样保持时间



17.5 单线半双工模式

半双工模式支持使用单个引脚（只使用 TX 引脚）来接收和发送，TX 引脚和 RX 引脚在芯片内部连接。

开启半双工模式的方式是对控制寄存器 3 (R16_USARTx_CTLR3) 的 HDSEL 位置位，但同时需要关闭 LIN 模式、智能卡模式、红外模式和同步模式，即保证 SCEN、CLKEN 和 IREN 位处于复位状态，这三位在控制寄存器 2 和 3 (R16_USARTx_CTLR2 和 R16_USARTx_CTLR3) 中。

设置成半双工模式之后，需要把 TX 的 I/O 口设置成开漏输出高模式。在 TE 置位的情况下，只要将数据写到数据寄存器上，就会发送出去。特别要注意的是，半双工模式可能会出现多设备使用单总线收发时的总线冲突，这需要用户用软件自行避免。

17.6 智能卡

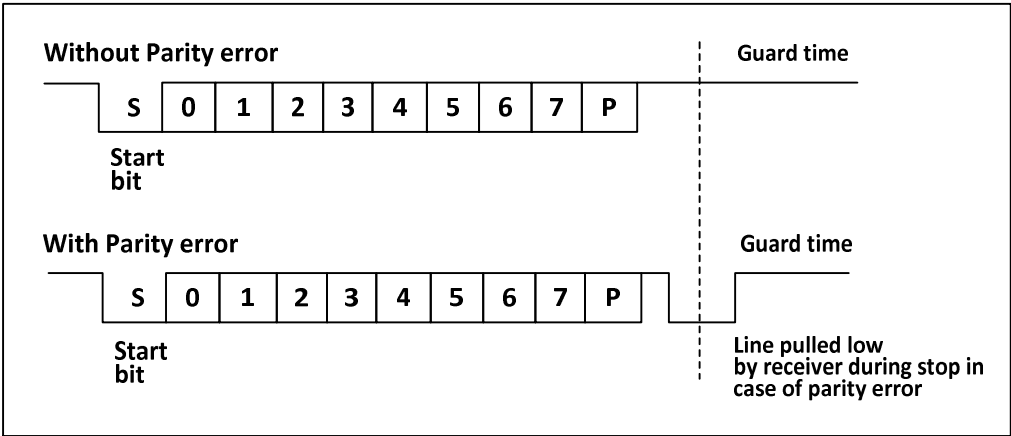
智能卡模式支持 ISO7816-3 协议访问智能卡控制器。

开启智能卡模式的方式是对控制寄存器 3 (R16_USARTx_CTLR3) 的 SCEN 位置位，但同时需要关闭 LIN 模式、半双工模式和红外模式，即保证 LINEN、HDSEL 和 IREN 位处于复位状态，但是可以开

启 CLKEN 来输出时钟，这些位在控制寄存器 2 和 3（R16_USARTx_CTLR2 和 R16_USARTx_CTLR3）中。

为了支持智能卡模式，USART 应当被置为 8 位数据位外加 1 位校验位，它的停止位建议配置成发送和接收都为 1.5 位，智能卡模式是一种单线半双工的协议，它使用 TX 线作为数据通讯，应当被配置为开漏输出加上拉。当接收方接收一帧数据检测到奇偶校验错误时，会在停止位时，发出一个 NACK 信号，即在停止位期间主动把 TX 拉低一个周期，发送方检测到 NACK 信号后，会产生帧错误，应用程序据此可以重发。图 17-4 展示了正确情况下和发生奇偶校验错误情况下的 TX 引脚上的波形图。USART 的 TC 标志（发送完成标志）可以延迟 GT（保护时间）个时钟产生，接收方也不会将自己置的 NACK 信号认成起始位。

图 17-4（未）发生奇偶校验错误示意图



在智能卡模式下，CK 引脚使能后输出的波形和通讯无关，它仅仅是给智能卡提供时钟的，它的值是 PB 时钟再经过五位可设置的时钟分频（分频值为 PSC 的两倍，最高 62 分频）。

17.7 IrDA

USART 模块支持控制 IrDA 红外收发器进行物理层通信。使用 IrDA 必须清除 LINEN、STOP、CLKEN、SCEN 和 HDSEL 位。USART 模块和 SIR 物理层（红外收发器）之间使用 NRZ（不归零）编码，最高支持到 115200bps 速率。

IrDA 是一个半双工的协议，如果 UASRT 正在给 SIR 物理层发数据，那么 IrDA 解码器将会忽视新发来的红外信号，如果 USART 正在接受从 SIR 发来的数据，那么 SIR 不会接受来自 USART 的信号。USART 发给 SIR 和 SIR 发给 USART 的电平逻辑是不一样的，SIR 接收逻辑中，高电平为 1，低电平为 0，但是在 SIR 发送逻辑中，高电平为 0，低电平为 1。

17.8 DMA

USART 模块支持 DMA 功能，可以利用 DMA 实现快速连续收发。当启用 DMA 时，TXE 被置位时，DMA 就会从设定的内存空间向发送缓冲区写数据。当使用 DMA 接收时，每次 RXNE 置位后，DMA 就会将接收缓冲区里的数据转移到特定的内存空间。

17.9 中断

USART 模块支持多种中断源，包括发送数据寄存器空（TXE）、CTS、发送完成（TC）、接收数据就绪（RXNE）、数据溢出（ORE）、线路空闲（IDLE）、奇偶校验出错（PE）、断开标志（LBD）、噪声（NE）、多缓冲通信的溢出（ORT）和帧错误（FE）等等。

表 17-1 中断和对应的使能位的关系

中断源	使能位
-----	-----

数据寄存器空 (TXE)	TXEIE
允许发送 (CTS)	CTSIE
发送完成 (TC)	TCIE
接收数据就绪 (RXNE)	RXNEIE
数据溢出 (ORE)	
线路空闲 (IDLE)	IDLEIE
奇偶校验出错 (PE)	PEIE
断开标志 (LBD)	LBDIE
噪声 (NE)	EIE
多缓冲通信的溢出 (ORT)	
多缓冲通信的帧错误 (FE)	

17.10 寄存器描述

表 17-2 USART1 相关寄存器列表

名称	访问地址	描述	复位值
R32_USART1_STATR	0x40013800	USART1 状态寄存器	0x000000C0
R32_USART1_DATAR	0x40013804	USART1 数据寄存器	0x000000XX
R32_USART1_BRR	0x40013808	USART1 波特率寄存器	0x00000000
R32_USART1_CTLR1	0x4001380C	USART1 控制寄存器 1	0x00000000
R32_USART1_CTLR2	0x40013810	USART1 控制寄存器 2	0x00000000
R32_USART1_CTLR3	0x40013814	USART1 控制寄存器 3	0x00000000
R32_USART1_GPR	0x40013818	USART1 保护时间和预分频寄存器	0x00000000

表 17-3 USART2 相关寄存器列表

名称	访问地址	描述	复位值
R32_USART2_STATR	0x40004400	USART2 状态寄存器	0x000000C0
R32_USART2_DATAR	0x40004404	USART2 数据寄存器	0x000000XX
R32_USART2_BRR	0x40004408	USART2 波特率寄存器	0x00000000
R32_USART2_CTLR1	0x4000440C	USART2 控制寄存器 1	0x00000000
R32_USART2_CTLR2	0x40004410	USART2 控制寄存器 2	0x00000000
R32_USART2_CTLR3	0x40004414	USART2 控制寄存器 3	0x00000000
R32_USART2_GPR	0x40004418	USART2 保护时间和预分频寄存器	0x00000000

表 17-4 USART3 相关寄存器列表

名称	访问地址	描述	复位值
R32_USART3_STATR	0x40004800	USART3 状态寄存器	0x000000C0
R32_USART3_DATAR	0x40004804	USART3 数据寄存器	0x000000XX
R32_USART3_BRR	0x40004808	USART3 波特率寄存器	0x00000000
R32_USART3_CTLR1	0x4000480C	USART3 控制寄存器 1	0x00000000
R32_USART3_CTLR2	0x40004810	USART3 控制寄存器 2	0x00000000
R32_USART3_CTLR3	0x40004814	USART3 控制寄存器 3	0x00000000
R32_USART3_GPR	0x40004818	USART3 保护时间和预分频寄存器	0x00000000

表 17-5 USART4 相关寄存器列表

名称	访问地址	描述	复位值
R32_USART4_STATR	0x40004C00	USART4 状态寄存器	0x000000C0
R32_USART4_DATAR	0x40004C04	USART4 数据寄存器	0x000000XX
R32_USART4_BRR	0x40004C08	USART4 波特率寄存器	0x00000000
R32_USART4_CTLR1	0x40004C0C	USART4 控制寄存器 1	0x00000000
R32_USART4_CTLR2	0x40004C10	USART4 控制寄存器 2	0x00000000
R32_USART4_CTLR3	0x40004C14	USART4 控制寄存器 3	0x00000000
R32_USART4_GPR	0x40004C18	USART4 保护时间和预分频寄存器	0x00000000

17.10.1 USART 状态寄存器 (USART_x_STATR) (x=1/2/3/4)

偏移地址: 0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						CTS	LBD	TXE	TC	RXNE	IDLE	ORE	NE	FE	PE

位	名称	访问	描述	复位值
[31:10]	Reserved	R0	保留。	0
9	CTS	RW0	CTS 状态改变标志。如果设置了 CTSE 位, 当 nCTS 输出状态改变时, 该位将由硬件置高。由软件清零。如果 CTSIE 位已经被置位, 则会产生中断。 1: nCTS 状态线上存在变化; 0: nCTS 状态线上没有变化。	0
8	LBD	RW0	LIN 断开检测标志。当检测到 LIN 断开时, 该位被硬件置位。由软件清零。如果 LBDIE 已经被置位, 则将会产生中断。 1: 检测到 LIN 断开; 0: 没有检测待 LIN 断开。	0
7	TXE	R0	发送数据寄存器空标志。当 TDR 寄存器中的数据被硬件转移到移位寄存器的时候, 该位被硬件置位。如果 TXEIE 已经被置位时, 就会产生中断, 对数据寄存器进行写操作, 此位将会被复位。 1: 数据已经被转移到移位寄存器; 0: 数据还没被转移到移位寄存器。	1
6	TC	RW0	发送完成标志。当含有数据的一帧发送完成后, 并且 TXE 被置位, 则硬件将会此位置位, 如果 TCIE 被置位, 还会产生对应中断, 软件读了此位再写数据寄存器则会清除此位。也可以直接写 0 来清除此位。 1: 发送完成; 0: 发送还未完成。	1

5	RXNE	RW0	<p>读数据寄存器非空标志，当移位寄存器中的数据被转移到数据寄存器中，该位会被硬件置位。如果 RXNEIE 已经被置位，则还会产生对应的中断。对数据寄存器的读操作可以将该位清除。也可以直接写 0 来清除该位。</p> <p>1：数据收到，能够读出； 0：数据还没收到。</p>	0
4	IDLE	RO	<p>总线空闲标志。当总线空闲时，该位将会被硬件置位。如果 IDLEIE 已经被置位，则会产生对应的中断。读状态寄存器再读数据寄存器的操作会清除此位。</p> <p>1：总线正空闲； 0：没有检测到总线空闲。</p> <p>注：此位不会被再次置位直到 RXNE 被置位。</p>	0
3	ORE	RO	<p>过载错误标志。当接收移位寄存器存在数据需要转到数据寄存器时，但是数据寄存器的接收域还有数据未读出时，此位将会被置位。如果 RXNEIE 被置位了，还会产生对应中断。</p> <p>1：发生过载错误； 0：没有过载错误。</p> <p>注：发生过载错误时，数据寄存器的值不会丢失，但是移位寄存器的值会被覆盖。如果设置可 EIE 位，在多缓冲区通讯模式下，ORE 标志位置位会产生中断。</p>	0
2	NE	RO	<p>噪声错误标志。当检测到噪声错误标志时，由硬件置位。读状态寄存器后，再读数据寄存器的操作会复位此位。</p> <p>1：检测到噪声； 0：没有检测到噪声。</p> <p>注：该位不会产生中断。如果设置了 EIE 位，在多缓冲区通讯模式下，FE 标志位置位会产生中断。</p>	0
1	FE	RO	<p>帧错误标志。当检测到同步错误，过多的噪声或者断开符，该位将会被硬件置位。读此位再读数据寄存器的操作会复位此位。</p> <p>1：检测到帧错误； 0：没有检测到帧错误。</p> <p>注：该位不会产生中断，如果设置了 EIE 位，在多缓冲区通讯模式下，FE 标志位置位会产生中断。</p>	0
0	PE	RO	<p>校验错误标志。在接收模式下，如果产生奇偶检验错误，硬件置位此位。读此位再读数据寄存器的操作会复位此位。在清除此位前，软件必须等 RXNE 标志位被置位。如果 PEIE 之前已经被置位，那么此位被置位会产生对应的中断。</p>	0

			1: 出现奇偶校验错误; 0: 没有检验错误。	
--	--	--	----------------------------	--

17.10.2 USART 数据寄存器 (USARTx_Datar) (x=1/2/3/4)

偏移地址: 0x04

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

位	名称	访问	描述	复位值
[31:9]	Reserved	R0	保留。	0
[8:0]	DR[8:0]	RW	数据寄存器。这个寄存器实际上是接收数据寄存器 (RDR) 和发送寄存器 (TDR) 两个寄存器组成, DR 的读写操作起始分别是读接收寄存器 (RDR) 和写发送寄存器 (TDR)。	X

17.10.3 USART 波特率寄存器 (USARTx_BRR) (x=1/2/3/4)

偏移地址: 0x08

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

位	名称	访问	描述	复位值
[31:16]	Reserved	R0	保留。	0
[15:4]	DIV_Mantissa [11:0]	RW	这 12 位定义了分频器除法因子的整数部分。	0
[3:0]	DIV_Fraction[3:0]	RW	这 4 位定义了分频器除法因子的小数部分。	0

17.10.4 USART 控制寄存器 1 (USARTx_CTLR1) (x=1/2/3/4)

偏移地址: 0x0C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	UE	M	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXEIE	IDLEIE	TE	RE	RWU	SBK	

位	名称	访问	描述	复位值
[31:14]	Reserved	R0	保留。	0
13	UE	RW	USART 使能位。当此位被置位后, 在当前字节传输完成后, USART 的分频器和输出都会停止	0

			工作。	
12	M	RW	字长位。 1: 9 个数据位; 0: 8 个数据位。	0
11	WAKE	RW	唤醒位。此位决定了把 USART 唤醒的方法: 1: 地址标记; 0: 总线空闲。	0
10	PCE	RW	校验位使能。对于接收方, 就是进行对数据的奇偶校验; 对于发送方, 就是插入校验位。一旦设置了此位, 只有当前字节传输完成后, 校验位使能才生效。	0
9	PS	RW	奇偶校验选择。0 表示偶校验, 1 表示奇校验。设置了该位后, 只有当前字节传输完成后, 校验位使能才生效。	0
8	PEIE	RW	奇偶检验中断使能位。对此位置位表示允许产生奇偶检验错误中断。	0
7	TXEIE	RW	发送缓冲区空中断使能。对此位置位表示允许产生发送缓冲区空中断。	0
6	TCIE	RW	发送完成中断使能。对此位置位表示允许产生发送完成中断。	0
5	RXNEIE	RW	接收缓冲区非空中断使能。对此位置位表示允许产生接收缓冲区非空中断。	0
4	IDLEIE	RW	总线空闲中断使能。对此位置位表示允许产生总线空闲中断。	0
3	TE	RW	发送使能。置此位会使能发送器。	0
2	RE	RW	接收使能。置此位会使能接收器, 接收器开始检测 RX 引脚上的起始位。	0
1	RWU	RW	接收唤醒。该位决定是否把 USART 置于静默模式: 1: 接收器处于静默模式; 0: 接收器处于正常工作模式。 注 1: 置 RWU 位之前, USART 需要先接收一个数据字节, 否则在静默模式下, 不能被总线空闲唤醒; 注 2: 当配置成地址标记唤醒时, 在 RXNE 被置位时, 不能用软件修改 RWU 位。	0
0	SBK	RW	发送帧断开字符控制位。置此位来发送一个帧断开字符。在断开帧的停止位时, 由硬件复位。 1: 发送; 0: 不发送。	0

17.10.5 USART 控制寄存器 2 (USARTx_CTLR2) (x=1/2/3/4)

偏移地址: 0x10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	LINEN	STOP	CLKEN	CPOL	CPHA	LBCL	Reserved	LBDIE	LBDL	Reserved	ADD[3:0]				

位	名称	访问	描述	复位值
[31:15]	Reserved	R0	保留。	0
14	LINEN	RW	LIN 模式使能位，置位则使能 LIN 模式。在 LIN 模式下，可以使用 SBK 位发送 LIN 同步断开符号，以及检测 LIN 同步断开符。	0
[13:12]	STOP	RW	停止位设置域。这两位来设置停止位。 00: 1 个停止位； 01: 0.5 个停止位； 10: 2 个停止位； 11: 1.5 个停止位。	00b
11	CLKEN	RW	时钟使能，使能 CK 引脚。 1: 使能； 0: 禁止。	0
10	CPOL	RW	时钟极性设置位。在同步模式下，可以用该位选择 SLCK 引脚上时钟输出的极性，和 CPHA 一起配合来产生需要的时钟/数据的采样关系。 1: 总线空闲时 CK 引脚上保持高电平； 0: 总线空闲时 CK 引脚上保持低电平。 注：使能发送后此位不可被修改。	0
9	CPHA	RW	时钟相位设置位。在同步模式下，可以用该位选择 SLCK 引脚上的时钟输出的相位，和 CPOL 位一起配合来产生需要的时钟/数据的采样关系。 1: 在时钟的第二个边沿进行数据捕获； 0: 在时钟的第一个边沿进行数据捕获。 注：使能发送后此位不可被修改。	0
8	LBCL	RW	最后一个时钟脉冲控制位。 在同步模式下，使用该位来控制是否在 CK 引脚上输出最后发送的那个数据字节对应的时钟脉冲； 1: 最后一位数据的时钟脉冲不从 CK 输出； 0: 最后一位数据的时钟脉冲会从 CK 输出。 注：使能发送后此位不可被修改。	0
7	Reserved	RW	保留。	0
6	LBDIE	RW	LIN 断开符检测中断使能，该位置位会使能 LBD 引起的中断；	0
5	LBDL	RW	LIN 断开符检测长度，该位用来选择是 11 位还是 10 位的断开符检测。 1: 11 位的断开符检测； 0: 10 位的断开符检测。	0
4	Reserved	RW	保留。	0
[3:0]	ADD[3:0]	RW	地址域，用来设置本设备的 USART 节点地址。在多处理器通讯下的静默模式中使用的，使用地址标记来唤醒某个 USART 设备。	0

17.10.6 USART 控制寄存器 3 (USARTx_CTLR3) (x=1/2/3/4)

偏移地址: 0x14

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						CTSIE	CTSE	RTSE	DMAT	DMAR	SCEN	NACK	HDSEL	IRLP	EIE

位	名称	访问	描述	复位值
[31:11]	Reserved	RO	保留。	0
10	CTSIE	RW	CTSIE 中断使能位, 置此位时在 CTS 被置位时会产生中断。	0
9	CTSE	RW	CTS 使能位, 置此位会使能 CTS 流控。	0
8	RTSE	RW	RTS 使能位, 置此位会使能 RTS 流控。	0
7	DMAT	RW	DMA 发送使能位。此位置 1 在发送时使用 DMA。	0
6	DMAR	RW	DMA 接收使能位。此位置 1 在接收时使用 DMA。	0
5	SCEN	RW	智能卡模式使能位, 置 1 使能智能卡模式。	0
4	NACK	RW	智能卡 NACK 使能位, 置此位在校验错误出现时, 发送 NACK。	0
3	HDSEL	RW	半双工模式选择位, 置此位选择半双工模式。	0
2	IRLP	RW	红外低功耗选择位, 置此位在选择红外线时, 启用低功耗模式。	0
1	IREN	RW	红外线使能位, 置此位使能红外模式。	0
0	EIE	RW	错误使能中断位, 置此位后, 在 DMAR 被置位的前提下, 如果 FE、ORE 或 NE 被置位, 就会产生中断。	0

17.10.7 USART 保护时间和预分频寄存器 (USARTx_GPR) (x=1/2/3/4)

偏移地址: 0x18

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

位	名称	访问	描述	复位值
[31:16]	Reserved	RO	保留。	0
[15:8]	GT[7:0]	RW	保护时间值域。该域规定了以波特率时钟为单位的保护时间。在智能卡模式下, 当保护时间过去后, 才会设置发送完成标志。	0
[7:0]	PSC[7:0]	RW	预分频器值域。 在红外低功耗模式下, 源时钟被该值 (全部 8 位有效) 分频, 值为 0 时表示保留; 在红外正常模式下, 此位只能被设置为 1; 在智能卡模式下, 源时钟被该值 (低 5 位有效)	0

			的两倍分频，来给智能卡提供时钟，值为 0 表示保留。	
--	--	--	----------------------------	--

第 18 章 两线通信总线（I2C）

内部集成电路总线（I2C）广泛用在微控制器和传感器及其他片外模块的通讯上，它本身支持多主从模式，仅仅使用两根线（SDA 和 SCL）就能以 100KHz（标准）和 400KHz（快速）两种速度通讯。I2C 总线还兼容 SMBus 协议，不仅支持 I2C 的时序，还支持仲裁、定时和 DMA，拥有 CRC 校验功能。

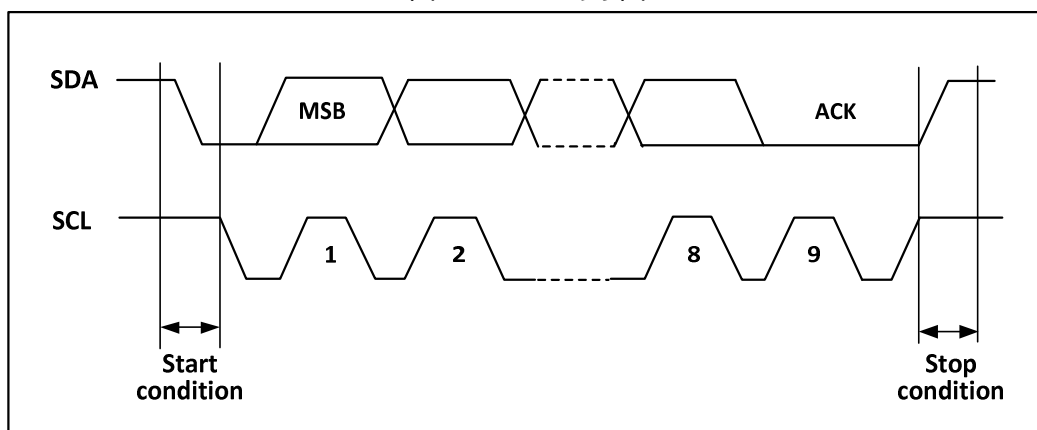
18.1 主要特征

- 支持主模式和从模式
- 支持 7 位或 10 位地址
- 从设备支持双 7 位地址
- 支持两种速度模式：100KHz 和 400KHz
- 多种状态模式，多种错误标志
- 支持加长的时钟功能
- 2 个中断向量
- 支持 DMA
- 支持 PEC
- 兼容 SMBus

18.2 概述

I2C 是半双工的总线，它同时只能运行在下列四种模式之一：主设备发送模式、主设备接收模式、从设备发送模式和从设备接收模式。I2C 模块默认工作在从模式，在产生起始条件后，会自动地切换到主模式，当仲裁丢失或产生停止信号后，会切换到从模式。I2C 模块支持多主机功能。工作在主模式时，I2C 模块会主动发出数据和地址。数据和地址都以 8 位为单位进行传输，高位在前，低位在后，在起始事件后的是一个字节（7 位地址模式下）或两个字节（10 位地址模式下）地址，主机每发送 8 位数据或地址，从机需要回复一个应答 ACK，即把 SDA 总线拉低，如图 18-1 所示。

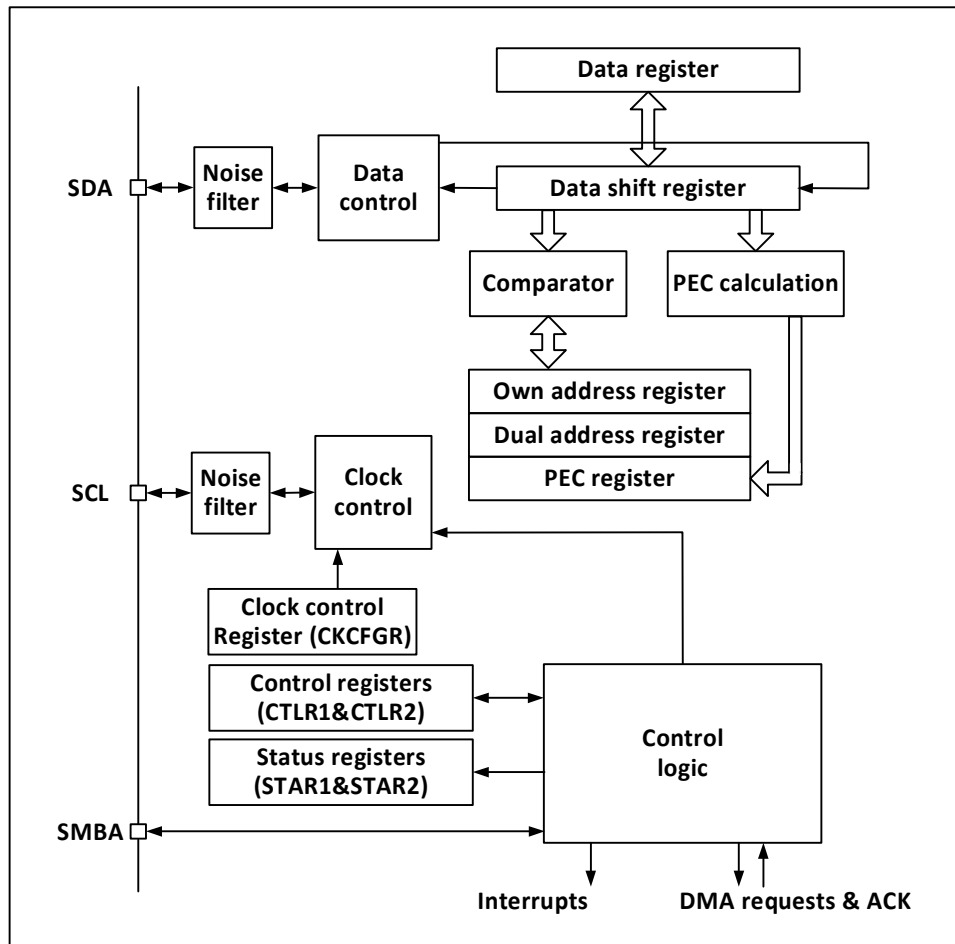
图 18-1 I2C 时序图



为了正常使用必须给 I2C 输入正确的时钟，其中标准模式下，输入时钟最低为 2MHz，在快速模式下，输入时钟最低为 4MHz。

图 18-2 是 I2C 模块功能框图。

图 18-2 I2C 功能框图



18.3 主模式

主模式时，I2C 模块主导数据传输并输出时钟信号，数据传输以开始事件开始，以结束事件结束。使用主模式通讯的步骤为：

在控制寄存器 2 (R16_I2Cx_CTLR2) 和时钟控制寄存器 (R16_I2Cx_CKCFGR) 中设置正确的时钟；
在上升沿寄存器 (R16_I2Cx_RTR) 设置合适的上升沿；

在控制寄存器 (R16_I2Cx_CTLR1) 中置 PE 位启动外设；

在控制寄存器 (R16_I2Cx_CTLR1) 中置 START 位，产生起始事件。在置 START 位后，I2C 模块会自动切换到主模式，MSL 位会置位，产生起始事件，在产生起始事件后，SB 位会置位，如果 ITEVTEN 位（在 R16_I2Cx_CTLR2）被置位，则会产生中断。此时应该读取状态寄存器 1 (R16_I2Cx_STAR1)，写从地址到数据寄存器后，SB 位会自动清除；

如果是使用 10 位地址模式，那么写数据寄存器发送头序列（头序列为 11110xx0b，其中的 xx 位是 10 位地址的最高两位）。

在发送完头序列之后，状态寄存器的 ADD10 位会被置位，如果 ITEVTEN 位已经置位，则会产生中断，此时应读取 R16_I2Cx_STAR1 寄存器后，写第二个地址字节到数据寄存器后，清除 ADD10 位。

然后写数据寄存器发送第二个地址字节，在发送完第二个地址字节后，状态寄存器的 ADDR 位会被置位，如果 ITEVTEN 位已经置位，则会产生中断，此时应读取 R16_I2Cx_STAR1 寄存器后再读一次 R16_I2Cx_STAR2 寄存器以清除 ADDR 位；

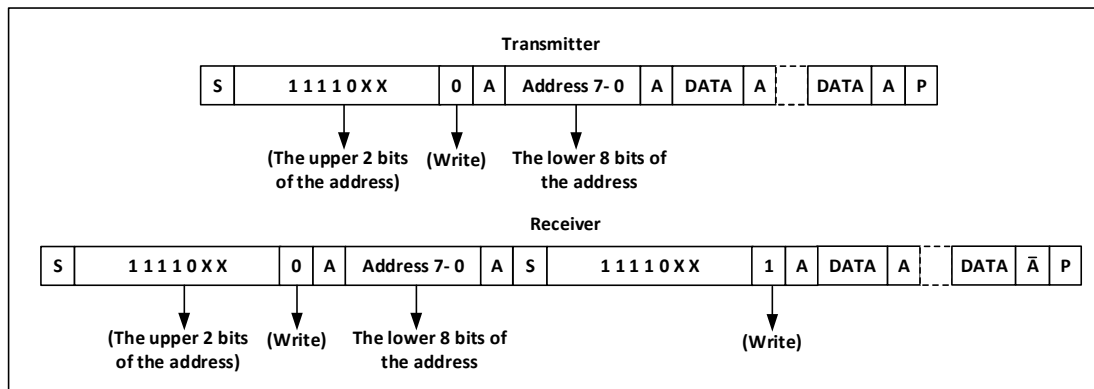
如果使用的是 7 位地址模式，那么写数据寄存器发送地址字节，在发送完地址字节后，状态寄存器的 ADDR 位会被置位，如果 ITEVTEN 位已经置位，则会产生中断，此时应读取 R16_I2Cx_STAR1 寄存器后再读一次 R16_I2Cx_STAR2 寄存器以清除 ADDR 位；

在 7 位地址模式下，发送的第一个字节为地址字节，头 7 位代表的是目标从设备地址，第 8 位

决定了后续报文的方向，0 代表是主设备写入数据到从设备，1 代表是主设备向从设备读取信息。

在 10 位地址模式下，如图 18-3 所示，在发送地址阶段，第一个字节为 11110xx0，xx 为 10 位地址的最高 2 位，第二个字节为 10 位地址的低 8 位。若后续进入主设备发送模式，则继续发送数据；若后续准备进入主设备接收模式，则需要重新发送一个起始条件，跟随发送一个字节为 11110xx1，然后进入主设备接收模式。

图 18-3 10 位地址时主机收发数据示意图



发送模式时，主设备内部的移位寄存器将数据从数据寄存器发送到 SDA 线上，当主设备接收到 ACK 时，状态寄存器 1 (R16_I2Cx_STAR1) 的 Tx E 被置位，如果 ITEVTEN 和 ITBUFEN 被置位，还会产生中断。向数据寄存器写入数据将会清除 Tx E 位。如果 Tx E 位被置位且上次发送数据之前没有新的数据被写入数据寄存器，那么 BTF 位会被置位，在其被清除之前，SCL 将保持低电平，读 R16_I2Cx_STAR1 后，向数据寄存器写入数据将会清除 BTF 位。

在接收模式时，I2C 模块会从 SDA 线接收数据，通过移位寄存器写进数据寄存器。在每个字节之后，如果 ACK 位被置位，那么 I2C 模块将会发出一个应答低电平，同时 RxNE 位会被置位，如果 ITEVTEN 和 ITBUFEN 被置位，还会产生中断。如果 RxNE 被置位且在新的数据被接收前，原有的数据没有被读出，则 BTF 位将被置位，在清除 BTF 之前，SCL 将保持低电平，读取 R16_I2Cx_STAR1 后，再读取数据寄存器将会清除 BTF 位。

主设备在结束发送数据时，会主动发一个结束事件，即置 STOP 位。在接收模式时，主设备需要在最后一个数据位的应答位置 NAK。注意，产生 NAK 后，I2C 模块将会切换至从模式。

18.4 从模式

从模式时，I2C 模块能识别它自己的地址和广播呼叫地址。软件能控制开启或禁止广播呼叫地址的识别。一旦检测到起始事件，I2C 模块将 SDA 的数据通过移位寄存器与自己的地址（位数取决于 ENDUAL 和 ADDMODE）或广播地址（ENG 置位时）相比较，如果不匹配将会忽略，直到产生新的起始事件。如果与头序列相匹配，则会产生一个 ACK 信号并等待第二个字节的地址；如果第二个字节的地址也匹配或 7 位地址情况下全段地址匹配，那么：

首先产生一个 ACK 应答；ADDR 位被置位，如果 ITEVTEN 位已经置位，那么还会产生相应的中断；

如果使用的是双地址模式（ENDUAL 位被置位），还需要读取 DUALF 位来判断主机唤起的是哪一个地址。

从模式默认是接收模式，在接收的头序列的最后一位为 1，或 7 位地址最后一位为 1 时（取决于第一次接收到头序列还是普通的 7 位地址），I2C 模块将进入到发送器模式，TRA 位将指示当前是接收器还是发送器模式。

发送模式时，在清除 ADDR 位后，I2C 模块将字节从数据寄存器通过移位寄存器发送到 SDA 线上。在收到一个应答 ACK 后，Tx E 位将被置位，如果设置了 ITEVTEN 和 ITBUFEN，还会产生一个中断。如果 Tx E 被置位但在下一个数据发送结束前没有新的数据被写入数据寄存器时，BTF 位将被置位。在清除 BTF 前，SCL 将保持低电平，读取状态寄存器 1 (R16_I2Cx_STAR1) 后，再向数据寄存器

写入数据将会清除 BTF 位。

接收模式时，在 ADDR 被清除后，I2C 模块将 SDA 上的数据通过移位寄存器存进数据寄存器，在每接收到一个字节后，I2C 模块都会置一个 ACK 位，并置 RxNE 位，如果设置了 ITEVTEN 和 ITBUFEN，还会产生一个中断。如果 RxNE 被置位，且在接收到新的数据前旧的数据没有被读出，那么 BTF 会被置位。在清除 BTF 位之前 SCL 会保持低电平。读取状态寄存器 1 (R16_I2Cx_STAR1) 并读取数据寄存器里的数据会清除 BTF 位。

当 I2C 模块检测到停止事件时，将置 STOPF 位，如果设置了 ITEVFEN 位，还会产生一个中断。用户需要读取状态寄存器 (R16_I2Cx_STAR1) 再写控制寄存器 (比如复位控制字 SWRST) 来清除。

18.5 错误

18.5.1 总线错误 BERR

在传输地址或数据期间，I2C 模块检测到外部的起始或停止事件时，将产生一个总线错误。产生总线错误时，BERR 位被置位，如果设置了 ITERREN 还会产生一个中断。在从模式下，数据被丢弃，硬件释放总线。如果是起始信号，硬件会认为是重启信号，开始等待地址或停止信号；如果是停止信号，则提前按正常的停止条件操作。在主模式下，硬件不会释放总线，同时不影响当前传输，由用户代码决定是否中止传输。

18.5.2 应答错误 AF

当 I2C 模块检测到一个字节后没有应答时，会产生应答错误。产生应答错误时：AF 会被置位，如果设置了 ITERREN 还会产生一个中断；遇到 AF 错误，如果 I2C 模块工作在从模式，硬件必须释放总线，如果处于主模式，软件必须生成一个停止事件。

18.5.3 仲裁丢失 ARLO

当 I2C 模块检测到仲裁丢失时，产生仲裁丢失错误。产生仲裁丢失错误时：ARLO 位被置位，如果设置了 ITERREN 还会产生一个中断；I2C 模块切换到从模式，并不再响应针对它的从地址发起的传输，除非有主机发起新的起始事件；硬件会释放总线。

18.5.4 过载/欠载错误 OVR

● 过载错误：

在从机模式下，如果禁止时钟延长，I2C 模块正在接收数据，如果已经接受到一个字节的数据，但是上一次接收到数据还没有被读出，则会产生过载错误。发生过载错误时，最后收到的字节将被丢弃，发送方应当重发最后一次发送的字节。

● 欠载错误：

在从模式下，如果禁止时钟延长，I2C 模块正在发送数据，如果在下一个字节的时钟到来之前新的数据还没有被写入到数据寄存器，那么将产生欠载错误。在发生欠载错误时，前一次数据寄存器里的数据将被发送两次，如果发生欠载错误，那么接收方应该丢弃重复收到的数据。为了不产生欠载错误，I2C 模块应当在下一个字节的第一个上升沿之前将数据写入数据寄存器。

18.6 时钟延长

如果禁止时钟延长，那么就存在发生过载/欠载错误的可能。但如果使能了时钟延长：

- 在发送模式下，如果 TxNE 置位且 BTF 置位，SCL 将一直为低，一直等待用户读取状态寄存器，并向数据寄存器写入待发送的数据；
- 在接收模式下，如果 RxNE 置位且 BTF 置位，那么 SCL 在接收到数据后将保持低，直到用户读取

状态寄存器，并读取数据寄存器；

由此可见，使能时钟延长可以避免出现过载/欠载错误。

18.7 SMBus

SMBus 也是一种双线接口，它一般应用在系统和电源管理之间。SMBus 和 I2C 有很多相似的地方，例如 SMBus 使用和 I2C 一样的 7 位地址模式，以下是 SMBus 和 I2C 的共同点：

- 1) 主从通信模式，主机提供时钟，支持多主多从；
- 2) 两线通讯结构，其中 SMBus 可选一个警示线；
- 3) 都支持 7 位地址格式。

同时 SMBus 和 I2C 也存在区别：

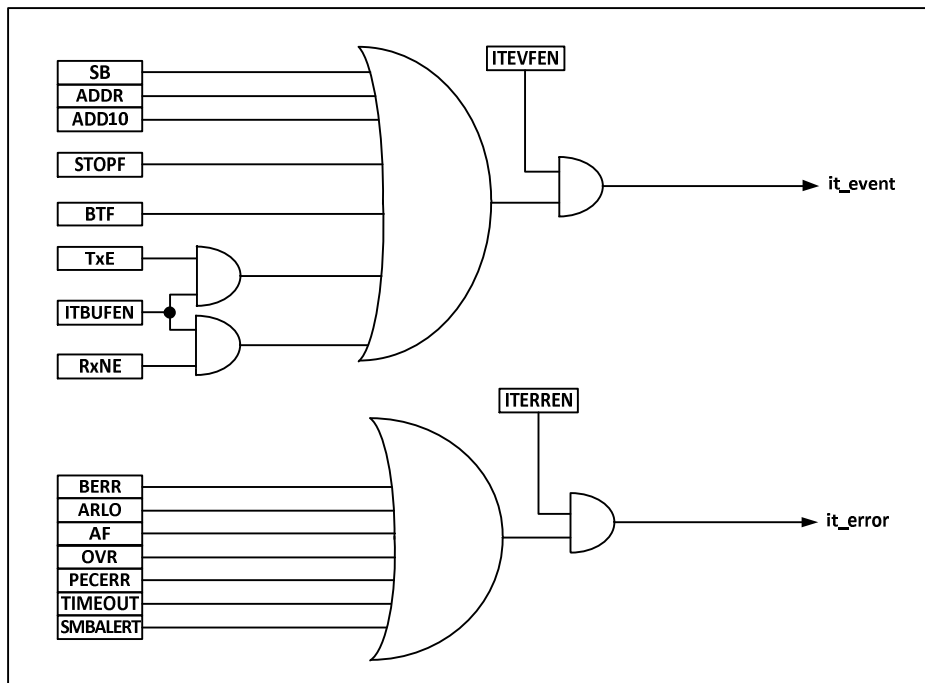
- 1) I2C 支持的速度最高 400KHz，而 SMBus 支持的最高是 100KHz，且 SMBus 有最小 10KHz 的速度限制；
- 2) SMBus 的时钟为低超过 35mS 时，会报超时，但 I2C 无此限制；
- 3) SMBus 有固定的逻辑电平，而 I2C 没有，取决于 VDD；
- 4) SMBus 有总线协议，而 I2C 没有。

SMBus 还包括设备识别、地址解析协议、唯一的设备标识符、SMBus 提醒和各种总线协议，具体请参考 SMBus 规范 2.0 版本。当使用 SMBus 时，只需要置控制寄存器的 SMBus 位，按需配置 SMBTYPE 位和 ENAARP 位。

18.8 中断

每个 I2C 模块都有两种中断向量，分别是事件中断和错误中断。两种中断支持图 18-4 的中断源。

图 18-4 I2C 中断请求



18.9 DMA

可以使用 DMA 来进行批量数据的收发。使用 DMA 时不能对控制寄存器的 ITBUFEN 位进行置位。

- 利用 DMA 发送

通过将 CTRL2 寄存器的 DMAEN 位置位可以激活 DMA 模式。只要 TxE 位被置位，数据将由 DMA 从

设定的内存装载进 I2C 的数据寄存器。需要进行以下设定来为 I2C 分配通道。

- 1) 向 DMA_PADDRx 寄存器设置 I2Cx_DATAR 寄存器地址, DMA_MADDRx 寄存器中设置存储器地址, 这样在每个 TxE 事件后, 数据将从存储器送至 I2Cx_DATAR 寄存器。
- 2) 在 DMA_CNTRx 寄存器中设置所需的传输字节数。在每个 TxE 事件后, 此值将被递减。
- 3) 利用 DMA_CFGRx 寄存器中的 PL[0:1]位配置通道优先级。
- 4) 设置 DMA_CFGRx 寄存器中的 DIR 位, 并根据应用要求可以配置在整个传输完成一半或全部完成时发出中断请求。
- 5) 通过设置 DMA_CFGRx 寄存器上的 EN 位激活通道。

当 DMA 控制器中设置的数据传输数目已经完成时, DMA 控制器给 I2C 接口发送一个传输结束的 EOT/ EOT_1 信号。在中断允许的情况下, 将产生一个 DMA 中断。

● 利用 DMA 接收

置位 CTLR2 寄存器的 DMAEN 后即可进行 DMA 接收模式。使用 DMA 接收时, DMA 将数据寄存器里的数据传送到预设的内存区域。需要以下步骤来为 I2C 分配通道。

- 1) 向 DMA_PADDRx 寄存器设置 I2Cx_DATAR 寄存器地址, DMA_MADDRx 寄存器中设置存储器地址, 这样在每个 RxNE 事件后, 数据将从 I2Cx_DATAR 寄存器写入存储器。
- 2) 在 DMA_CNTRx 寄存器中设置所需的传输字节数。在每个 RxNE 事件后, 此值将被递减。
- 3) 用 DMA_CFGRx 寄存器中的 PL[0:1]配置通道优先级。
- 4) 清除 DMA_CFGRx 寄存器中的 DIR 位, 根据应用要求可以设置在数据传输完成一半或全部完成时发出中断请求。
- 5) 设置 DMA_CFGRx 寄存器中的 EN 位激活该通道。

当 DMA 控制器中设置的数据传输数目已经完成时, DMA 控制器给 I2C 接口发送一个传输结束的 EOT/EOT_1 信号。在中断允许的情况下, 将产生一个 DMA 中断。

18.10 包校验错误

包错误校验 (PEC) 是为了提供传输的可靠性而增加一项 CRC8 校验的步骤, 使用以下多项式对每一位串行数据进行计算:

$$C=X^8+X^2+X+1$$

PEC 计算是由控制寄存器的 ENPEC 位激活, 对所有信息字节进行计算, 包括地址和读写位在内。在发送时, 启用 PEC 会在最后一字节数据之后加上一个字节的 CRC8 计算结果; 而在接收模式, 在最后一字节被认为是 CRC8 校验结果, 如果和内部的计算结果不符合, 就会回复一个 NAK, 如果是主接收器, 无论校验结果正确与否, 都会回复一个 NAK。

18.11 调试模式

当系统进入调试模式之后, 可以通过 DEBUG 模块的 DBG_I2Cx_SMBUS_TIMEOUT 位来决定 I2CSMBus 的超时控制是继续工作还是停止。

18.12 寄存器描述

表 18-1 I2C1 相关寄存器列表

名称	访问地址	描述	复位值
R16_I2C1_CTLR1	0x40005400	I2C1 控制寄存器 1	0x0000
R16_I2C1_CTLR2	0x40005404	I2C1 控制寄存器 2	0x0000
R16_I2C1_OADDR1	0x40005408	I2C1 地址寄存器 1	0x0000

R16_I2C1_OADDR2	0x4000540C	I2C1 地址寄存器 2	0x0000
R16_I2C1_DATAR	0x40005410	I2C1 数据寄存器	0x0000
R16_I2C1_STAR1	0x40005414	I2C1 状态寄存器 1	0x0000
R16_I2C1_STAR2	0x40005418	I2C1 状态寄存器 2	0x0000
R16_I2C1_CKCFGR	0x4000541C	I2C1 时钟寄存器	0x0000
R16_I2C1_RTR	0x40005420	I2C1 上升时间寄存器	0x0002

表 18-2 I2C2 相关寄存器列表

名称	访问地址	描述	复位值
R16_I2C2_CTLR1	0x40005800	I2C2 控制寄存器 1	0x0000
R16_I2C2_CTLR2	0x40005804	I2C2 控制寄存器 2	0x0000
R16_I2C2_OADDR1	0x40005808	I2C2 地址寄存器 1	0x0000
R16_I2C2_OADDR2	0x4000580C	I2C2 地址寄存器 2	0x0000
R16_I2C2_DATAR	0x40005810	I2C2 数据寄存器	0x0000
R16_I2C2_STAR1	0x40005814	I2C2 状态寄存器 1	0x0000
R16_I2C2_STAR2	0x40005818	I2C2 状态寄存器 2	0x0000
R16_I2C2_CKCFGR	0x4000581C	I2C2 时钟寄存器	0x0000
R16_I2C2_RTR	0x40005820	I2C2 上升时间寄存器	0x0002

18.12.1 I2C 控制寄存器 (I2Cx_CTLR1) (x=1/2)

偏移地址: 0x00

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWRST	Reserved	ALERT	PEC	POS	ACK	STOP	START	NOSTRETCH	ENG	ENPEC	ENARP	SMBTYPE	Reserved	SMBUS	PE

位	名称	访问	描述	复位值
15	SWRST	RW	软件重置, 用户代码置此位会使 I2C 外设重置。在复位前确定 I2C 总线的引脚被释放, 总线处于空闲状态。 注: 该位可以在总线上没有检测到停止条件但是 busy 位为 1 时, 重置 I2C 模块。	0
14	Reserved	RO	保留。	0
13	ALERT	RW	SMBus 提醒位, 用户代码可以设置此位或清除此位; 当 PE 置位后, 此位可以被硬件清除。 1: 驱动 SMBusALERT 引脚使其变低, 响应地址头应紧跟在 ACK 信号后面; 0: 释放 SMBusALERT 引脚使其变高, 响应地址头应紧跟在 NACK 信号后面。	0
12	PEC	RW	数据包出错检测使能位, 置此位启用数据包出错检测。用户代码可以对此位置位或清零; 当 PEC 被传输后, 产生开始或结束信号, 或 PE 位清 0 时, 硬件清零该位; 1: 带 PEC; 0: 不带 PEC。	0

			注：仲裁丢失时，PEC 失效。	
11	POS	RW	<p>ACK 和 PEC 位置设置位，该位可以被用户代码置位或清零，在 PE 被清零后，可以被硬件清除；</p> <p>1: ACK 位控制在移位寄存器里接收的下一个字节的 ACK 或 NAK。PEC 移位寄存器里接收的下一字节是 PEC；</p> <p>0: ACK 位控制当前移位寄存器内正在接受的字节的 ACK 或 NAK。PEC 位表明当前移位寄存器的字节是 PEC。</p> <p>注：POS 位在 2 字节数据接收中的用法如下：必须在接收之前配置好。为了 NACK 第 2 个字节，必须在清除 ADDR 位后立刻清除 ACK 位；为了检测第二个字节的 PEC，必须在 ADDR 事件发生后，配置 POS 位后设置 PEC 位。</p>	0
10	ACK	RW	<p>应答使能位，该位可以被用户代码置位或清零，当 PE 位被置位时，该位可以被硬件清除；</p> <p>1: 在接收到一个字节后返回一个应答；</p> <p>0: 不设应答。</p>	0
9	STOP	RW	<p>停止事件产生位，该位可以被用户代码置位或清零，或当检测到停止事件时，由硬件清除，或检测到超时错误时，由硬件将其置位。</p> <p>主模式下：</p> <p>1: 在当前字节传输或当前起始条件发出后产生停止事件；</p> <p>0: 无停止事件产生。</p> <p>从模式下：</p> <p>1: 在当前字节传输后释放 SCL 和 SDA 线；</p> <p>0: 无停止事件产生。</p>	0
8	START	RW	<p>起始事件产生位，该位可以被用户代码置位或清零，当起始条件发出后或 PE 被清零时，由硬件清零。</p> <p>主模式下：</p> <p>1: 重复产生起始事件；</p> <p>0: 无起始事件产生。</p> <p>从模式下：</p> <p>1: 当总线空闲时，产生起始事件；</p> <p>0: 无起始事件产生。</p>	0
7	NOSTRETCH	RW	<p>禁止时钟延长位，此位用于在 ADDB 或 BTF 标志被置位的情况下，禁止从模式下的时钟延长，直至被软件清零。</p> <p>1: 禁止时钟延长；</p> <p>0: 允许时钟延长。</p>	0
6	ENGCG	RW	广播呼叫使能位，置此位使能广播呼叫，应答广播地址 00h。	0
5	ENPEC	RW	PEC 使能位，置此位开启 PEC 计算。	0

4	ENARP	RW	ARP 使能位，置此位使能 ARP。 如果 SMBTYPE=0，则使用 SMBus 设备的默认地址；如果 SMBTYPE=1，则使用 SMBus 的主地址。	0
3	SMBTYPE	RW	SMBus 设备类型，置 1 为 SMBus 主设备，置 0 为 SMBus 从设备。	0
2	Reserved	RO	保留。	0
1	SMBUS	RW	SMBus 模式选择位，置 1 为使用 SMBus 模式，置 0 为使用 I2C 模式。	0
0	PE	RW	I2C 外设使能位。 1：启用 I2C 模块； 0：禁用 I2C 模块。	0

18.12.2 I2C 控制寄存器 2 (I2Cx_CTLR2) (x=1/2)

偏移地址：0x04

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			LAST	DMAEN	ITBUFEN	ITEVTEN	ITERREN	Reserved			FREQ[5:0]				

位	名称	访问	描述	复位值
[15:13]	Reserved	RO	保留。	0
12	LAST	RW	DMA 最后一次传输设置位。 1：下一次 DMA 的 EOT 是最后的传输； 0：下一次 DMA 的 EOT 不是最后的传输。 注：该位在主接收模式使用，可以在最后一次接收数据时产生一个 NAK。	0
11	DMAEN	RW	DMA 请求使能位，置此位在 TxEN 或 RxEN 被置位时允许 DMA 请求。	0
10	ITBUFEN	RW	缓冲器中断使能位。 1：当 TxEN 或 RxEN 被置位时，产生事件中断； 0：当 TxEN 或 RxEN 被置位时，不产生中断。	0
9	ITEVTEN	RW	时间中断使能位，置此位使能事件中断。 在下列条件下，将产生此中断： SB=1 (主模式)； ADDR=1 (主从模式)； ADDR10=1 (主模式)； STOPF=1 (从模式)； BTF=1，但是没有 TxEN 或 RxEN 事件； 如果 ITBUFEN=1，TxEN 事件为 1； 如果 ITBUFEN=1，RxNE 事件为 1。	0
8	ITERREN	RW	出错中断使能位，置位表示允许出错中断。 在下列条件下，将产生该中断： BERR=1；ARL0=1；AF=1；OVR=1；PECERR=1； TIMEOUT=1；SMBAlert=1。	0
[7:6]	Reserved	RO	保留。	0
[5:0]	FREQ[5:0]	RW	I2C 模块时钟频率域，必须输入正确的时钟频	0

			率以产生正确的时序，允许的范围在 4~60MHz 之间。必须设置在 000100b 到 111100b 之间，单位为 MHz。	
--	--	--	---	--

18.12.3 I2C 地址寄存器 1 (I2Cx_OADDR1) (x=1/2)

偏移地址：0x08

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADD MODE	Reserved						ADD[9:8]		ADD[7:1]							ADD0

位	名称	访问	描述	复位值
15	ADDMODE	RW	地址模式。 1: 10 位从机地址（不响应 7 位地址）； 0: 7 位从机地址（不响应 10 位地址）。	0
[14:10]	Reserved	RO	保留。	0
[9:8]	ADD[9:8]	RW	接口地址，在使用 10 位地址时为第 9-8 位，在使用 7 位地址时忽略。	0
[7:1]	ADD[7:1]	RW	接口地址，第 7-1 位。	0
0	ADD0	RW	接口地址，使用 10 位地址时为第 0 位，在使用 7 位地址时忽略。	0

18.12.4 I2C 地址寄存器 2 (I2Cx_OADDR2) (x=1/2)

偏移地址：0x0C

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									ADD2[7:1]							ENDUAL

位	名称	访问	描述	复位值
[15:8]	Reserved	RO	保留。	0
[7:1]	ADD2[7:1]	RW	接口地址，双地址模式下地址的 7-1 位。	0
0	ENDUAL	RW	双地址模式使能位，置此位可以让 ADD2 也能被识别。	0

18.12.5 I2C 数据寄存器 (I2Cx_DATAR) (x=1/2)

偏移地址：0x10

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

位	名称	访问	描述	复位值
[15:8]	Reserved	RO	保留。	0
[7:0]	DR[7:0]	RW	数据寄存器，该域用来存放接收到的数据或存放用于发送到总线的数据。	0

18.12.6 I2C 状态寄存器 1 (I2Cx_STAR1) (x=1/2)

偏移地址：0x14

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMBALERT	TIMEOUT	Reserved	PECERR	OVR	AF	ARLO	BERR	TxE	RxNE	Reserved	STOPF	ADD10	BTF	ADDR	SB

位	名称	访问	描述	复位值
15	SMBALERT	RW0	SMBus 警示位，该位可以由用户写 0 复位，或在 PE 变低时由硬件复位。 在 SMBus 主机模式下： 1：在引脚上产生了 SMBus 警示； 0：无 SMBus 警示。 在 SMBus 从机模式下： 1：收到 SMBAlert 响应地址头序列直到 SMBAlert 变低； 0：没有收到 SMBAlert 响应地址头序列。	0
14	TIMEOUT	RW0	超时或 Tlow 错误标志位，该位可以由用户写 0 复位，或在 PE 变低时由硬件复位。 1：SCL 处于低已达到 25mS，或主机低电平累计时钟扩招时间超过 10mS，或从设备低电平累计时间超过 25mS； 0：无超时错误。 注：在从模式下此位被置位，从设备会复位通讯，硬件会释放总线；在主模式下此位被置位，硬件会发出停止条件。	0
13	Reserved	R0	保留。	0
12	PECERR	RW0	在接收时发生 PEC 错误标志位，该位可以由用户写 0 复位，或在 PE 变低时由硬件复位。 1：有 PEC 错误，接收到 PEC 后，返回 NAK； 0：无 PEC 错误。	0
11	OVR	RW0	过载、欠载标志位。 1：有过载、欠载事件发生：当 NOSTRETCH=1 时，在接收模式中收到一个新的字节时，数据寄存器里的内容还未被读出，则新接收的字节将丢失；在发送模式时，没有新的数据写入数据寄存器，同样的字节将被发送两次； 0：无过载、欠载事件。	0
10	AF	RW0	应答失败标志位，该位可以由用户写 0 复位，或在 PE 变低时由硬件复位。 1：应答错误； 0：应答正常。	0
9	ARLO	RW0	仲裁丢失标志位，该位可以由用户写 0 复位，或在 PE 变低时由硬件复位。 1：检测到仲裁丢失，模块失去对总线的控制； 0：仲裁正常。	0
8	BERR	RW0	总线出错标志位，该位可以由用户写 0 复位，	0

			或在 PE 变低时由硬件复位。 1: 起始或停止条件出错; 0: 正常。	
7	TxE	R0	数据寄存器为空标志位, 向数据寄存器写数据可以清除, 或产生一个起始或停止位后, 或当 PE 为 0 后, 由硬件自动清除。 1: 发送数据时, 发送数据寄存器为空; 0: 数据寄存器非空。	0
6	RxNE	R0	数据寄存器非空标志位, 对数据寄存器的读写操作将清除此位, 或当 PE 为 0 后, 由硬件清除此位。 1: 接收数据时, 数据寄存器不为空; 0: 正常。	0
5	Reserved	R0	保留。	0
4	STOPF	R0	停止事件标志位, 用户读取状态寄存器 1 之后, 对控制寄存器 1 的写操作将会清除该位, 或当 PE 为 0 后, 由硬件清除此位。 1: 在应答之后, 从设备在总线上检测到停止事件; 0: 没有检测到停止事件。	0
3	ADD10	R0	10 位地址头序列发送标志位, 用户读取状态寄存器 1 之后, 对控制寄存器 1 的写操作将会清除该位, 或当 PE 为 0 后, 由硬件清除此位。 1: 在 10 位地址模式下, 主设备已经将第一个地址字节发送出去; 0: 无。	0
2	BTF	R0	字节发送结束标志位, 用户读取状态寄存器 1 后, 对数据寄存器的读写将清除此位; 在传输中, 发起一个起始或者停止事件后, 或当 PE 为 0 后, 由硬件清除此位。 1: 字节发送结束。当 NOSTRETCH=0 时: 发送时, 当一个新数据被发送且数据寄存器还未被写入新数据; 接收时, 当接收一个新的字节但是数据寄存器还未被读取; 0: 无。	0
1	ADDR	RWO	地址被发送/地址匹配标志位, 用户读取状态寄存器 1 后, 对状态寄存器 2 的读操作将会清除此位, 或当 PE 为 0 时, 由硬件清除此位。 主模式: 1: 地址发送结束: 在 10 位地址模式下, 当收到地址的第二个字节的 ACK 后改为被置位; 在 7 位地址模式下, 当收到地址的 ACK 后被置位; 0: 地址发送没有结束。 从模式: 1: 收到的地址匹配; 0: 地址不匹配或没有收到地址。	0

0	SB	R0	起始位发送标志位，读取状态寄存器 1 后写数据寄存器的操作将清除该位，或当 PE 为 0 时，硬件将会清除此位。 1：已发送起始位； 0：未发送起始位。	0
---	----	----	--	---

18.12.7 I2C 状态寄存器 2 (I2Cx_STAR2) (x=1/2)

偏移地址：0x18

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PEC[7:0]								DUALF	SMBHOST	SMBDEFAULT	GENCALL	Reserved	TRA	BUSY	MSL

位	名称	访问	描述	复位值
[15:8]	PEC[7:0]	R0	包错误检查域，当 PEC 使能时（ENPEC 置位），此域存放 PEC 的值。	0
7	DUALF	R0	匹配检测标志位，在产生停止位或起始位时，或在 PE=0 时，硬件会将该位清零。 1：接收到的地址与 OAR2 中的内容相符； 0：接收到的地址与 OAR1 中的内容相符。	0
6	SMBHOST	R0	SMBus 主机头标志位，在产生停止位或起始位时，或在 PE=0 时，硬件会将该位清零。 1：当 SMBTYPE=1 且 ENARP=1 时，收到了 SMBus 主机地址； 0：未接收到 SMBus 主机地址。	0
5	SMBDEFAULT	R0	SMBus 设备默认地址标志位，在产生停止位或起始位时，或在 PE=0 时，硬件会将该位清零。 1：当 ENARP=1，收到 SMBus 设备的默认地址； 0：未收到地址。	0
4	GENCALL	R0	广播呼叫地址标志位，在产生停止位或起始位时，或者在 PE=0 时，硬件会将该位清零。 1：当 ENG=1 时，收到广播呼叫的地址； 0：未收到广播呼叫地址。	0
3	Reserved	R0	保留。	0
2	TRA	R0	发送/接收标志位，在检测到停止事件（STOPF=1），重复的起始条件、总线仲裁丢失（ARLO=1）或 PE=0 时，硬件会将其清零。 1：数据已发送； 0：接收到数据。 该位根据地址字节的 R/W 位来决定。	0
1	BUSY	R0	总线忙标志位，该位在检测到一个停止位时会被清零。在接口被禁用时（PE=0），该信息仍被更新。 1：总线忙：SDA 或 SCL 存在低电平； 0：总线空闲无通讯。	0
0	MSL	R0	主从模式指示位，当接口处于主模式时	0

			(SB=1)，硬件将该位置位；当总线检测到一个停止位，仲裁丢失时，或 PE=0 时，硬件会清除该位。	
--	--	--	--	--

18.12.8 I2C 时钟寄存器 (I2Cx_CKCFGR) (x=1/2)

偏移地址: 0x1C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
F/S	DUTY	Reserved	CCR[11:0]												

位	名称	访问	描述	复位值
15	F/S	RW	主模式选择位。 1: 快速模式; 0: 标准模式。	0
14	DUTY	RW	快速模式时的占空比: 1: $T_{\text{低电平}}/T_{\text{高电平}} = 16/9$; 0: $T_{\text{低电平}}/T_{\text{高电平}} = 2$ 。	0
[13:12]	Reserved	RO	保留。	0
[11:0]	CCR[11:0]	RW	时钟分频系数域, 决定 SCL 时钟的频率波形。	0

18.12.9 I2C 上升时间寄存器 (I2Cx_RTR) (x=1/2)

偏移地址: 0x20

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

位	名称	访问	描述	复位值
[15:6]	Reserved	RO	保留。	0
[5:0]	TRISE[5:0]	RW	最大上升时间域。这个位设置主模式的 SCL 的上升时间。最大的上升沿时间等于 TRISE-1 个时钟周期。此位只能在 PE 清零下设置。比如如果 I2C 模块的输入时钟周期为 125nS, 而 TRISE 的值为 9, 那么最大上升沿时间为 (9-1)*125nS, 即 1000nS。	000010b

第 19 章 串行外设接口（SPI）

SPI 支持以三线同步串行模式进行数据交互，加上片选线支持硬件切换主从模式，支持以单根数据线通讯。

19.1 主要特征

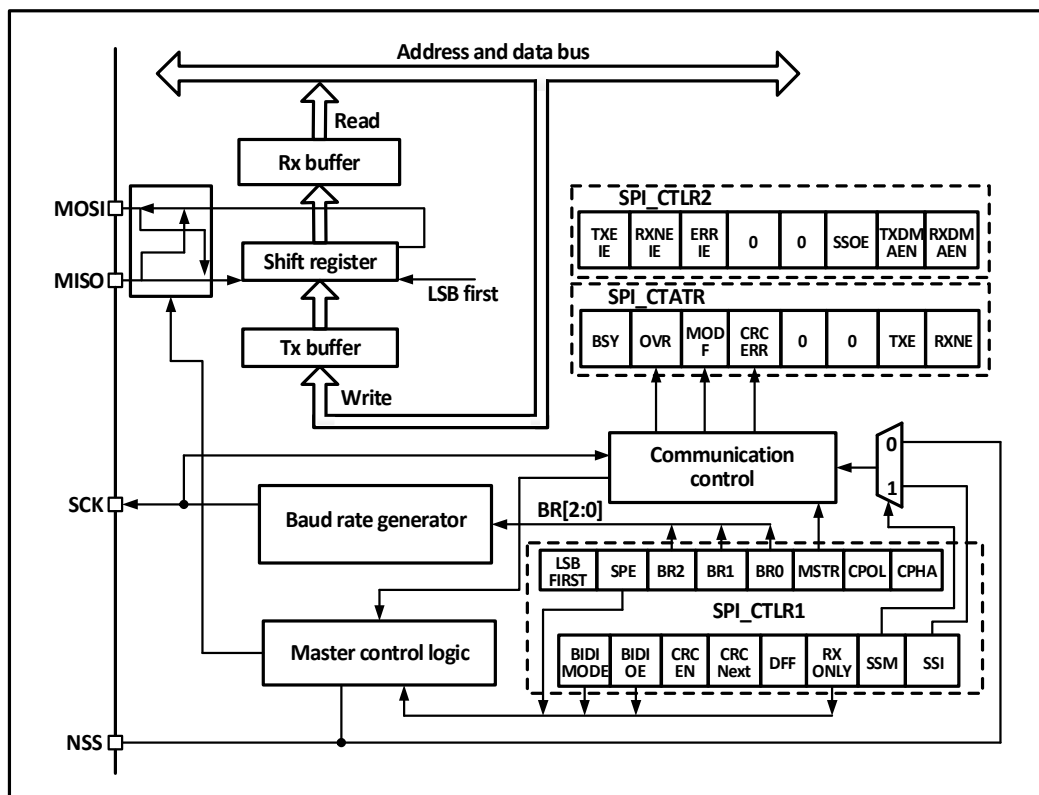
19.1.1 SPI 特征

- 支持全双工同步串行模式
- 支持单线半双工模式
- 支持主模式和从模式，多从模式
- 支持 8 位或 16 位数据结构
- 最高时钟频率支持到 F_{pclk} 的一半
- 数据顺序支持 MSB 或 LSB 在前
- 支持硬件或软件控制 NSS 引脚
- 收发支持硬件 CRC 校验
- 收发缓冲器支持 DMA 传输
- 支持修改时钟相位和极性

19.2 SPI 功能描述

19.2.1 概述

图 19-1 SPI 结构框图



由图 19-1 可以看出，与 SPI 相关的主要是 MISO、MOSI、SCK 和 NSS 四个引脚。其中 MISO 引脚在 SPI 模块工作在主模式下时，是数据输入引脚；工作在从模式下时，是数据输出引脚。MOSI 引脚工作在主模式下时，是数据输出引脚；工作在从模式时，是数据输入引脚。SCK 是时钟引脚，时钟

- 信号一直由主机输出，从机接收时钟信号并同步数据收发。NSS 引脚是片选引脚，有以下用法：
- 1) NSS 由软件控制：此时 SSM 被置位，内部 NSS 信号由 SSI 决定输出高还是低，这种情况一般用于 SPI 主模式；
 - 2) NSS 由硬件控制：在 NSS 输出使能时，即 SS0E 置位时，在 SPI 主机向外发送输出时会主动拉低 NSS 引脚，如果拉低 NSS 脚，则会产生一个硬件错误；SS0E 不置位，则可以用于多主机模式，如果它被拉低则会强行进入从机模式，MSTR 位会被自动清除。

可以通过 CPHA 和 CPOL 配置 SPI 的工作模式。CPHA 置位表示模块在时钟的第二个边沿进行数据采样，数据被锁存，CPHA 不置位表示 SPI 模块在时钟的第一个边沿进行采样，数据被锁存。CPOL 则表示无数据时时钟保持高电平还是低电平。

主机和设备需要设置为相同的 SPI 模式，在配置 SPI 模式前，需要清除 SPE 位。DEF 位可以决定 SP 的单个数据长度是 8 位还是 16 位。LSBFIRST 可以控制单个数据字是高位在前还是低位在前。

表 19-1 SPI 模式区分

	模式 0	模式 1	模式 2	模式 3
CPOL	0	1	1	1
CPHA	0	0	0	1

19.2.2 主模式

- 在 SPI 模块工作在主模式时，由 SCK 产生串行时钟。配置成主模式进行以下步骤：
- 配置控制寄存器的 BR[2:0]域来确定时钟；
 - 配置 CPOL 和 CPHA 位来确定 SPI 模式；
 - 配置 DEF 确定数据字长；
 - 配置 LSBFIRST 确定帧格式；
 - 配置 NSS 引脚，比如置 SS0E 位让硬件去置 NSS。也可以置 SSM 位并把 SSI 位置高；
 - 置 MSTR 位和 SPE 位，需要保证 NSS 此时已经是高。
- 需要发送数据时只需要向数据寄存器写要发送的数据就行了。SPI 会从发送缓冲区并行地把数据送到移位寄存器，然后按照 LSBFIRST 的设置将数据从移位寄存器发出去，当数据已经到了移位寄存器时，TXE 标志会被置位，如果已经置位了 TXEIE，那么会产生中断。如果 TXE 标志位置位需要向数据寄存器里填数据，维持完整的数据流。
- 当接收器接收数据时，当数据字的最后一个采样时钟沿到来时，数据从移位寄存器并行地转移到接收缓冲区，RXNE 位被置位，如果之前置位了 RXNEIE 位，还会产生中断。此时应该尽快读取数据寄存器取走数据。

图 19-2 SPI 主模式读写模式 0

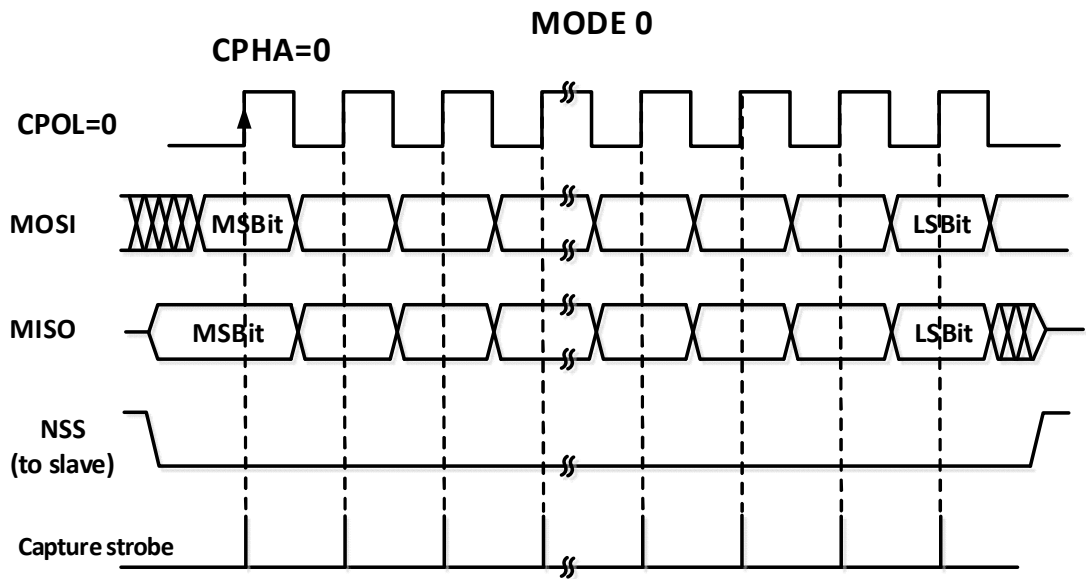


图 19-3 SPI 主模式读写模式 1

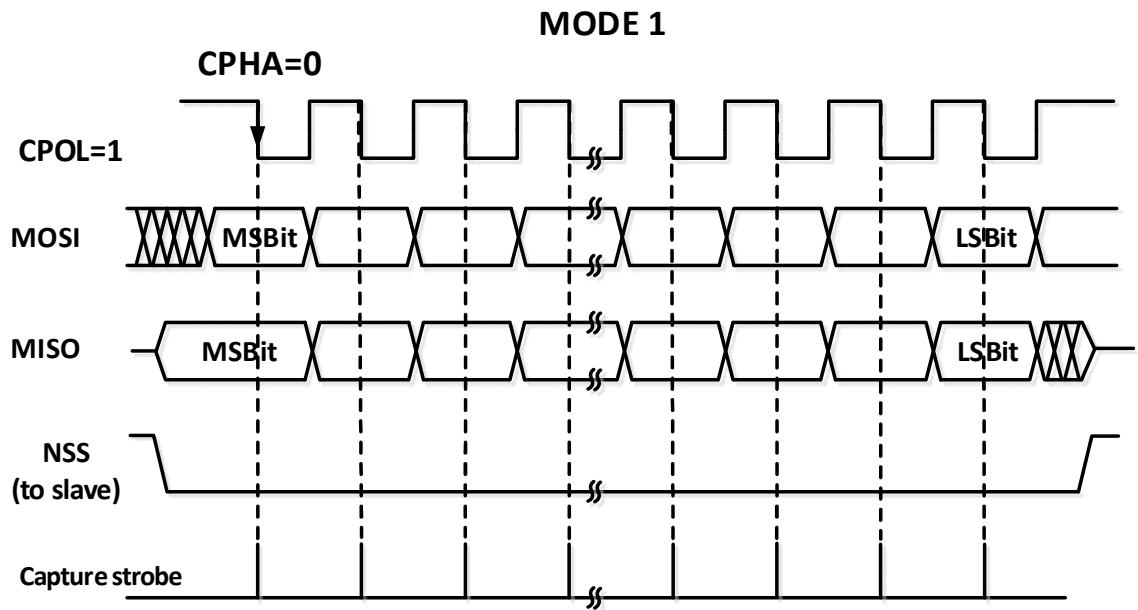


图 19-4 SPI 主模式读写模式 2

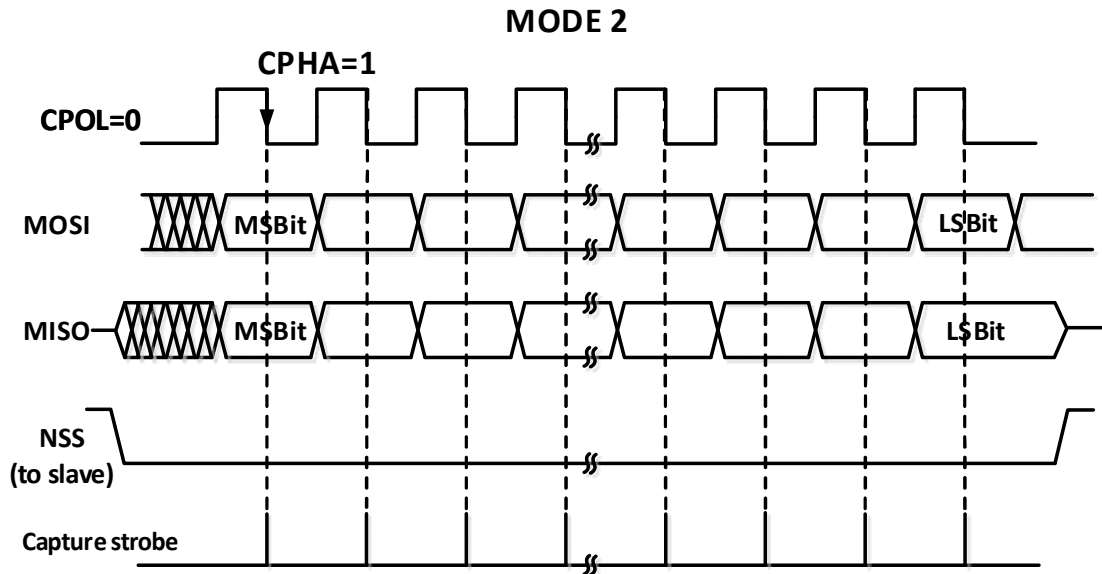
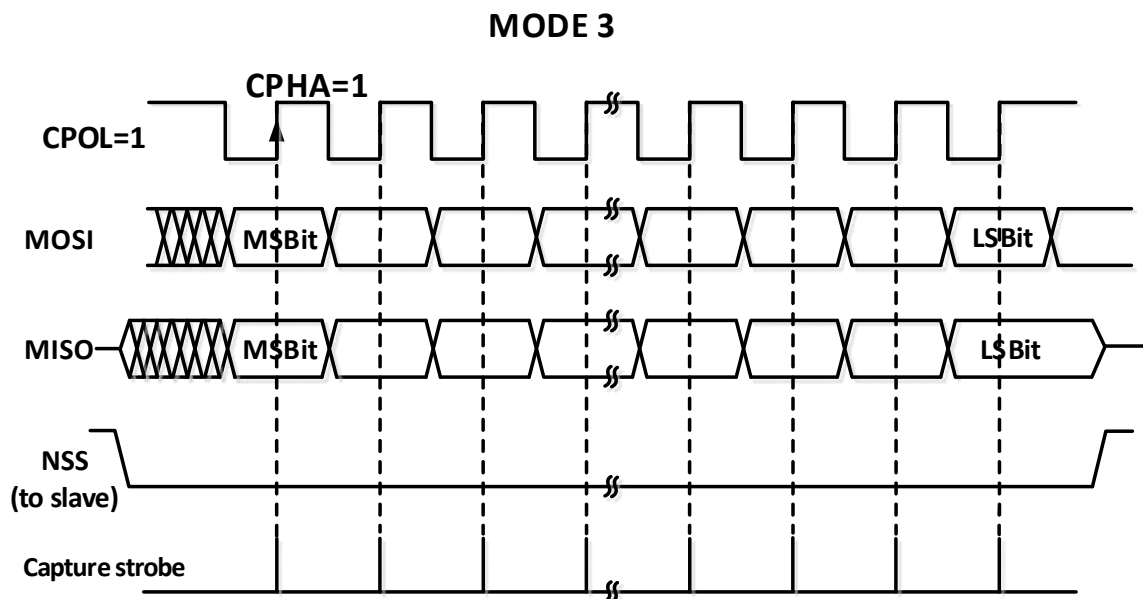


图 19-5 SPI 主模式读写模式 3



19.2.3 从模式

当 SPI 模块工作在从模式时，SCK 用于接收主机发来的时钟，自身的波特率设置无效。配置成从模式的步骤如下：

配置 DEF 位设置数据位长度；

配置 CPHA 匹配主机工作模式；

根据收发的配置和 CPOL 来决定 SPI 的工作模式；

若需要在从模式进行发送，则需要将 CPOL 置位，配置为模式 2 或模式 3，主机根据需要更改配置；

若只需要在从模式进行接收，则只需要匹配主机 CPOL 模式即可；

配置 LSBFIRST 匹配主机数据帧格式；

硬件管理模式，NSS 管脚需要保持为低电平，如果设置 NSS 为软件管理（SSM 置位），那么请保持 SSI 不被置位；

清除 MSTR 位，置 SPE 位，开启 SPI 模式。

在发送时，当 SCK 出现第一个从机接收采样沿时，从机开始发送。发送的过程就是发送缓冲区的数据移到发送移位寄存器，当发送缓冲区的数据移到了移位寄存器之后，会置位 TXE 标志，如果之前置位了 TXEIE 位，那么会产生中断。

在接收时，最后一个时钟采样沿之后，RXNE 位被置位，移位寄存器接收到的字节被转移到接收缓冲区，读数据寄存器的读操作可以获得接收缓冲区里的数据。如果在 RXNE 置位之前 RXNEIE 已经被置位，那么会产生中断。

图 19-6 SPI 从模式读模式 0

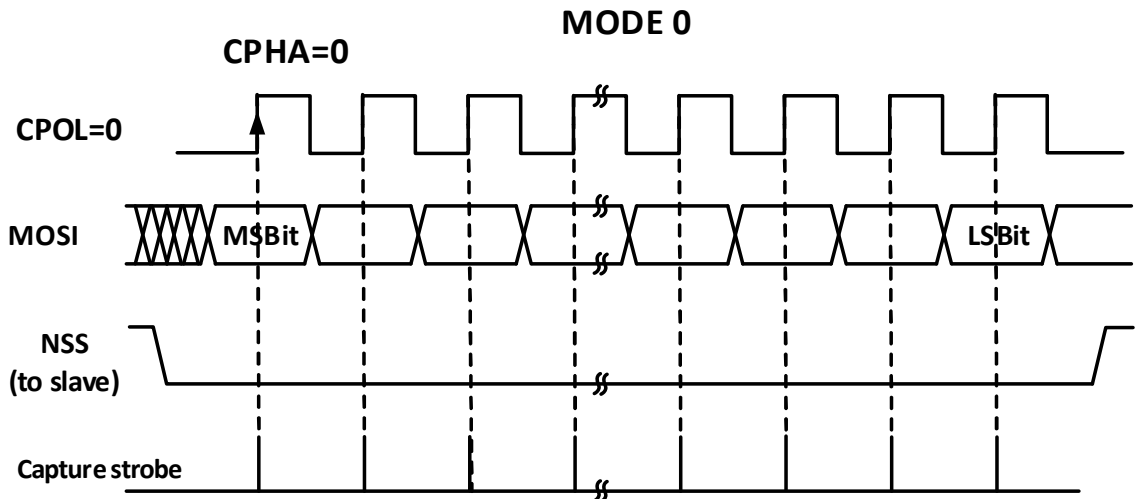


图 19-7 SPI 从模式读模式 1

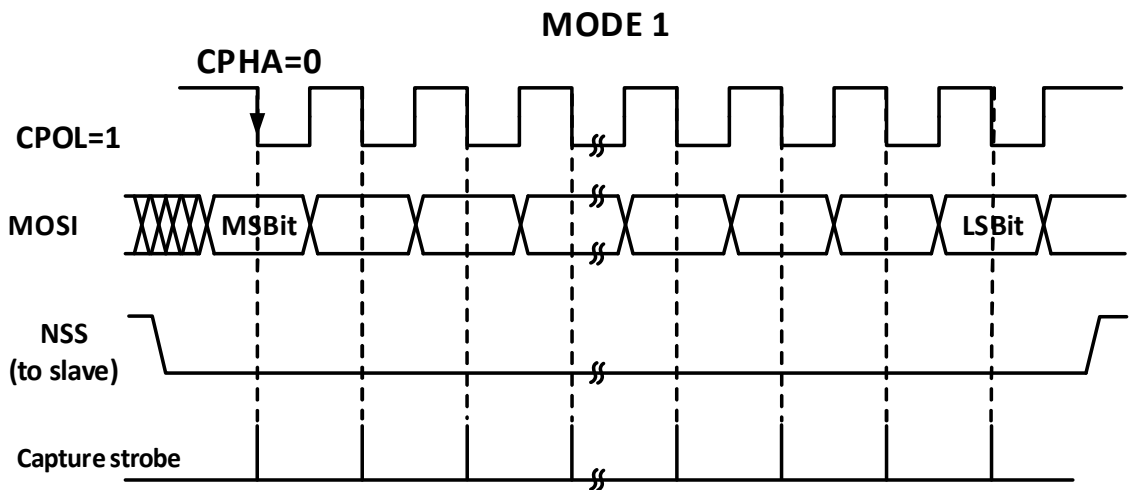


图 19-8 SPI 从模式读写模式 2

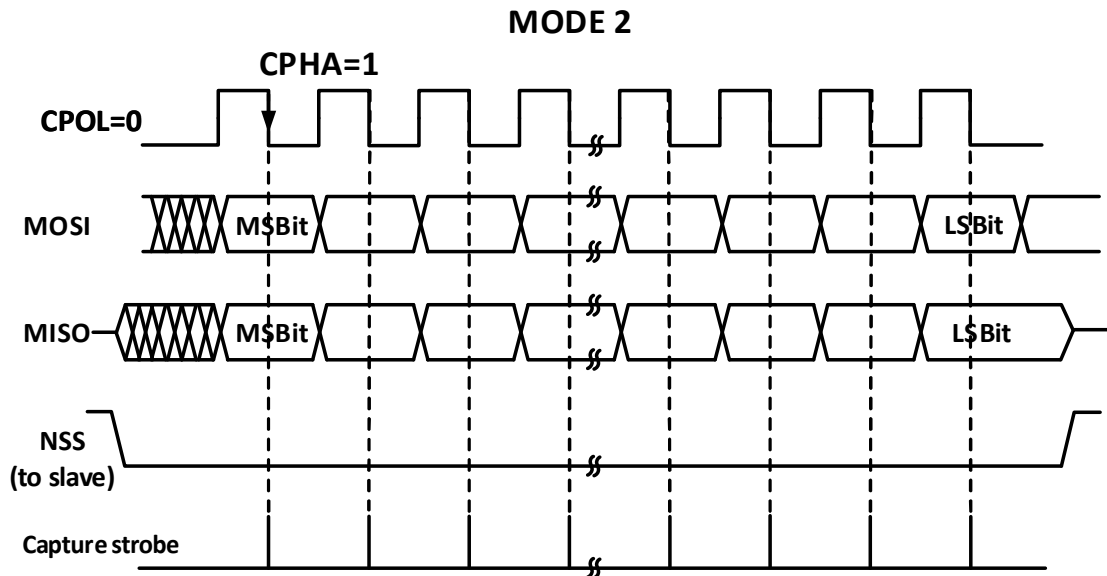
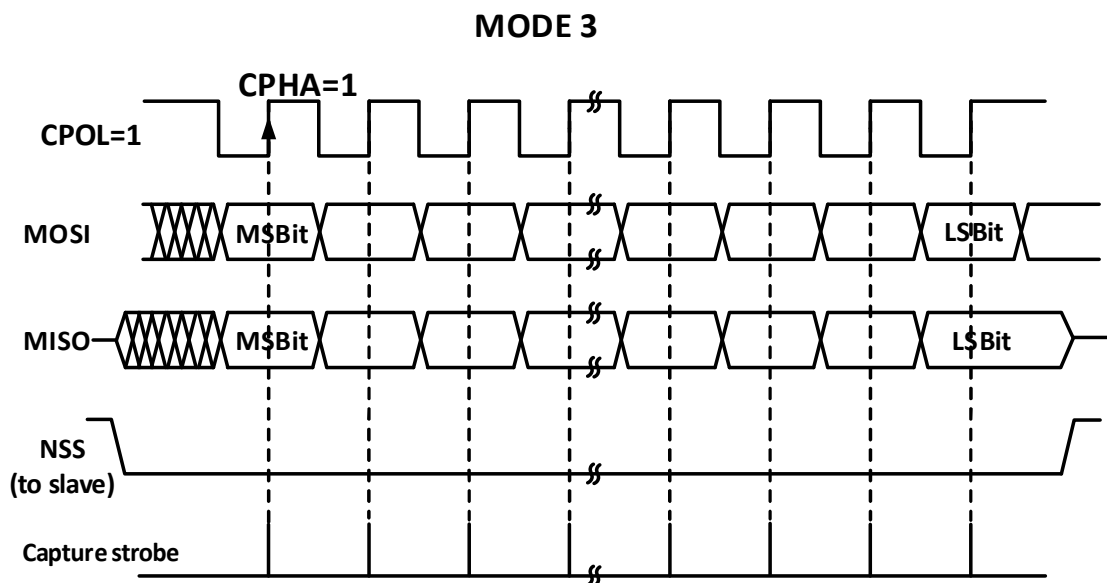


图 19-9 SPI 从模式读写模式 3



19.2.4 单工模式

SPI 接口可以工作在半双工模式，即主设备使用 MOSI 引脚，从设备使用 MISO 引脚进行通讯。使用半双工通讯时需要把 BIDIMODE 置位，使用 BIDIOE 控制传输方向。

在正常全双工模式下将 RXONLY 位置位可以将 SPI 模块设置为仅仅接收的单工模式，在 RXONLY 置位之后会释放一个数据脚，主模式和从模式释放的引脚并不相同。也可以不理睬接收的数据将 SPI 置成只发送的模式。

19.2.5 CRC

SPI 模块使用 CRC 校验保证全双工通信的可靠性，数据收发分别使用单独的 CRC 计算器。CRC 计算的多项式由多项式寄存器决定，对于 8 位数据宽度和 16 位数据宽度，会分别使用不同的计算方法。

设置 CRGEN 位会启用 CRC 校验，同时会使 CRC 计算器复位。在发送完最后一个数据字节后，置

CRCNEXT 位会在当前字节发送结束后发送 TXCRCR 计算器的计算结果，同时最后接收到的接收移位寄存器的值如果与本地算出来的 RXCRCR 的计算值不相符，那么 CRCERR 位会被置位。使用 CRC 校验需要在配置 SPI 工作模式时设置多项式计算器并置 CRCEN 位，并在最后一个字或半字置 CRCNEXT 位发送 CRC 并进行接收 CRC 的校验。注意，收发双方的 CRC 计算多项式应该统一。

19.2.6 DMA

SPI 模块支持使用 DMA 加快数据通讯速度，可以使用 DMA 向发送缓冲区填写数据，或使用 DMA 从接收缓冲区及时取走数据。DMA 会以 RXNE 和 TXE 为信号及时取走或发来数据。DMA 也可以工作在单工或加 CRC 校验的模式。

19.2.7 错误

● 主模式失效错误

当 SPI 工作在 NSS 引脚硬件管理模式，发生了外部拉低 NSS 引脚的操作；或在 NSS 引脚软件管理模式，SSI 位被清零；或 SPE 位被清零，导致 SPI 被关闭；或 MSTR 位被清零，SPI 进入从模式。如果 ERRIE 位已经被置位，还会产生中断。清除 MODF 位步骤：首先执行一次对 R16_SPI1_STATR 的读或写操作，然后写 R16_SPI1_CTLR1。

● 溢出错误

如果主机发送了数据，而从设备的接收缓冲区中还有未读取的数据，就会发生溢出错误，OVR 位被置位，如果 ERRIE 被置位还会产生中断。发送溢出错误应该重新开始当前传输。读取数据寄存器再读取状态寄存器会消除此位。

● CRC 错误

当接收到的 CRC 校验字和 RXCRCR 的值不匹配时，会产生 CRC 校验错误，CRCERR 位会被置位。

19.2.8 中断

SPI 模块的中断支持五个中断源，其中发送缓冲区空、接收缓冲区非空这两个事件分别会置位 TXE 和 RXNE，在分别置位了 TXEIE 和 RXNEIE 位的情况下会产生中断。除此之外上面提到的三种错误也会产生中断，分别是 MODF、OVR 和 CRCERR，在使能了 ERRIE 位之后，这三种错误也会产生错误中断。

19.3 寄存器描述

表 19-2 SPI1 相关寄存器列表

名称	访问地址	描述	复位值
R16_SPI1_CTLR1	0x40013000	SPI1 控制寄存器 1	0x0000
R16_SPI1_CTLR2	0x40013004	SPI1 控制寄存器 2	0x0000
R16_SPI1_STATR	0x40013008	SPI1 状态寄存器	0x0002
R16_SPI1_DATAR	0x4001300C	SPI1 数据寄存器	0x0000
R16_SPI1_CRCR	0x40013010	SPI1 多项式寄存器	0x0007
R16_SPI1_RCRCR	0x40013014	SPI1 接收 CRC 寄存器	0x0000
R16_SPI1_TCRCR	0x40013018	SPI1 发送 CRC 寄存器	0x0000
R16_SPI1_HSCR	0x40013024	SPI1 高速控制寄存器	0x0000

表 19-2 SPI2 相关寄存器列表

名称	访问地址	描述	复位值
R16_SPI2_CTLR1	0x40003800	SPI2 控制寄存器 1	0x0000
R16_SPI2_CTLR2	0x40003804	SPI2 控制寄存器 2	0x0000
R16_SPI2_STATR	0x40003808	SPI2 状态寄存器	0x0002
R16_SPI2_DATAR	0x4000380C	SPI2 数据寄存器	0x0000
R16_SPI2_CRCCR	0x40003810	SPI2 多项式寄存器	0x0007
R16_SPI2_RCRCR	0x40003814	SPI2 接收 CRC 寄存器	0x0000
R16_SPI2_TCRCR	0x40003818	SPI2 发送 CRC 寄存器	0x0000
R16_SPI2_HSCR	0x40003824	SPI2 高速控制寄存器	0x0000

19.3.1 SPI 控制寄存器 1 (SPIx_CTLR1) (x=1/2)

偏移地址: 0x00

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BIDI MODE	BIDI OE	CRCEN	CRC NEXT	DFF	RX ONLY	SSM	SSI	LSB FIRST	SPE	BR[2:0]			MSTR	CPOL	CPHA

位	名称	访问	描述	复位值
15	BIDIMODE	RW	单向数据模式使能位。 1: 选择单线双向模式; 0: 选择双线双向模式。	0
14	BIDIOE	RW	单线输出使能位, 和 BIDIMODE 配合使用。 1: 使能输出, 仅发送; 0: 禁止输出, 仅接收。	0
13	CRCEN	RW	硬件 CRC 校验使能位, 该位只能在 SPE 为 0 时写入, 该位只能在全双工模式下使用。 1: 启动 CRC 计算; 0: 禁止 CRC 计算。	0
12	CRCNEXT	RW	在接下来的一次数据传输后, 发送 CRC 寄存器的值。这位应该在向数据寄存器写入最后一个数据后立刻置位。 1: 发送 CRC 校验结果; 0: 继续发送数据寄存器的数据。	0
11	DFF	RW	数据帧长度位, 此位只能在 SPE 为 0 时写入。 1: 使用 16 位数据长度进行收发; 0: 使用 8 位数据长度进行收发。	0
10	RXONLY	RW	双线模式下只接收位, 该位和 BIDIMODE 配合使用。置此位可以让设备只接收不发送。 1: 只接收, 单工模式; 0: 全双工模式。	0
9	SSM	RW	片选引脚管理位, 此位决定 NSS 引脚的电平由硬件还是软件控制。 1: 软件控制 NSS 引脚; 0: 硬件控制 NSS 引脚。	0

8	SSI	RW	片选引脚控制位，在 SSM 置位的情况下，此位决定 NSS 引脚的电平。 1: NSS 为高电平； 0: NSS 为低电平。	0
7	LSBFIRST	RW	帧格式控制位。不可以在通讯时修改此位。 1: 先发送 LSB； 0: 先发送 MSB。	0
6	SPE	RW	SPI 使能位。 1: 启用 SPI； 0: 禁用 SPI。	0
[5:3]	BR[2:0]	RW	波特率设置域，在通讯时不可以修改此域。 000: $F_{PCLK}/2$ ； 001: $F_{PCLK}/4$ ； 010: $F_{PCLK}/8$ ； 011: $F_{PCLK}/16$ ； 100: $F_{PCLK}/32$ ； 101: $F_{PCLK}/64$ ； 110: $F_{PCLK}/128$ ； 111: $F_{PCLK}/256$ 。 注：此位仅适用于 HSRXEN 位为 0 的情况；当 HSRXEN 位为 1 时，SCK 频率为 $F_{PCLK}/(BR+2)$ 。	000b
2	MSTR	RW	主从设置位，在通讯时不可以修改此位。 1: 配置为主设备； 0: 配置为从设备。	0
1	CPOL	RW	时钟极性选择位，在通讯时不可以修改此位。 1: 空闲状态时，SCK 保持高电平； 0: 空闲状态时，SCK 保持低电平。	0
0	CPHA	RW	时钟相位设置位，在通讯时不可以修改此位。 1: 数据采样从第二个时钟沿开始； 0: 数据采样从第一个时钟沿开始。	0

19.3.2 SPI 控制寄存器 2 (SPIx_CTLR2) (x=1/2)

偏移地址：0x04

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								TXEIE	RXNEIE	ERRIE	Reserved	SSOE	TXDMAEN	RXDMAEN	

位	名称	访问	描述	复位值
[15:8]	Reserved	RO	保留。	0
7	TXEIE	RW	发送缓冲区空中断使能位。置此位允许 TXE 被置位时产生中断。	0
6	RXNEIE	RW	接收缓冲区非空中断使能位。置此位允许 RXNE 被置位时产生中断。	0
5	ERRIE	RW	错误中断使能位。置此位允许在产生错误 (CRCERR, OVR, MODF) 时产生中断。	0
[4:3]	Reserved	RO	保留。	0
2	SSOE	RW	SS 输出使能。禁止 SS 输出可以工作在多主模式下。 1: 使能 SS 输出；	0

			0: 禁止主模式下的 SS 输出。	
1	TXDMAEN	RW	发送缓冲区 DMA 使能位。 1: 启用发送缓冲区 DMA; 0: 禁用发送缓冲区 DMA。	0
0	RXDMAEN	RW	接收缓冲区 DMA 使能位。 1: 启用接收缓冲区 DMA; 0: 禁用接收缓冲区 DMA。	0

19.3.3 SPI 状态寄存器 (SPIx_STATR) (x=1/2)

偏移地址: 0x08

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								BSY	OVR	MODF	CRC ERR	Reserved	TXE	RXNE	

位	名称	访问	描述	复位值
[15:8]	Reserved	R0	保留。	0
7	BSY	R0	忙标志位, 该位由硬件置位或复位。 1: SPI 正在通讯, 或发送缓冲区非空; 0: SPI 不在通讯。	0
6	OVR	RW0	溢出标志位, 该位由硬件置位, 软件复位。 1: 出现溢出错误; 0: 没有出现溢出错误。	0
5	MODF	R0	模式错误标志位, 该位由硬件置位, 软件复位。 1: 出现了模式错误; 0: 没有出现模式错误。	0
4	CRCERR	RW0	CRC 错误标志位, 该位由硬件置位, 软件复位。 1: 收到的 CRC 值与 RCRCR 的值不一致; 0: 收到的 CRC 值与 RCRCR 的值一致。	0
[3:2]	Reserved	R0	保留。	0
1	TXE	R0	发送缓冲区为空标志位。 1: 发送缓冲区为空; 0: 发送缓冲区非空。	1
0	RXNE	R0	接收缓冲区非空标志位。 1: 接收缓冲区非空; 0: 接收缓冲区为空。	0

19.3.4 SPI 数据寄存器 (SPIx_DATAR) (x=1/2)

偏移地址: 0x0C

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

位	名称	访问	描述	复位值
[15:0]	DR[15:0]	RW	数据寄存器。数据寄存器用于存放接收到的数	0

			据或预存将要发送出去的数据，因此数据寄存器的读写实际上是对应操作不同的区域，其中读对应接收缓冲区，写对应发送缓冲区。数据的接收和发送可以是 8 位或者 16 位的，需要在传输之前就确定使用多少位的数据。使用 8 位进行数据传输时，只有数据寄存器的低 8 位被使用，接收时高 8 位强制为 0。使用 16 位数据结构则会使全部 16 位数据寄存器被使用。	
--	--	--	--	--

19.3.5 SPI 多项式寄存器 (SPIx_CRCR) (x=1/2)

偏移地址: 0x10

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

位	名称	访问	描述	复位值
[15:0]	CRCPOLY[15:0]	RW	CRC 多项式。此域定义 CRC 计算用到的多项式。	0007h

19.3.6 SPI 接收 CRC 寄存器 (SPIx_RXCRCR) (x=1/2)

偏移地址: 0x14

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

位	名称	访问	描述	复位值
[15:0]	RXCRC[15:0]	RO	接收 CRC 值。存储着计算出来的接收到的字节的 CRC 校验的结果。对 CRCEN 置位会复位该寄存器。计算方法使用 CRCPOLY 用到的多项式。8 位模式下只有低 8 位参与计算，16 位模式下全部 16 位都会参与计算。需要在 BSY 为 0 时去读取这个寄存器。	0

19.3.7 发送 CRC 寄存器 (SPIx_TXCRCR) (x=1/2)

偏移地址: 0x18

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

位	名称	访问	描述	复位值
[15:0]	TXCRC[15:0]	RO	发送 CRC 值。存储着计算出来的已经发送出去的字节的 CRC 校验的结果。对 CRCEN 置位会复位该寄存器。计算方法使用 CRCPOLY 用到的多项式。8 位模式下只有低 8 位参与计算，16 位模式下全部 16 位都会参与计算。需要在 BSY 为 0 时去读取这个寄存器。	0

19.3.8 SPI 高速控制寄存器（SPIx_HSCR）（x=1/2）

偏移地址：0x24

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

位	名称	访问	描述	复位值
[15:1]	Reserved	R0	保留	0
0	HSRXEN	RW	SPI 高速模式下读使能： 1：使能高速读模式； 0：关闭高速读模式。	0

第 20 章 USB 全速主机/设备控制器（USBFS）

20.1 USB 控制器简介

内嵌 USB2.0 控制器和 USB-PHY，具有主机控制器和 USB 设备控制器双重角色。当作为主机控制器时，它可支持低速、全速和高速的 USB 设备/HUB。当作为设备控制器时，可以灵活设置为低速、全速或高速模式以适应各种应用。

芯片内嵌 USB 控制器及收发器，特性如下：

- 双重角色设备控制器，支持 USB Host 主机功能和 USB Device 设备功能。
- 遵循 On-The-Go Supplement to the USB2.0 规范，主机和设备模式均支持 USB2.0 全速 12Mbps 或低速 1.5Mbps。
- 支持软件 HNP 和 SRP 协议。
- 支持 USB 控制传输、批量传输、中断传输和同步/实时传输。
- 支持最大 64 字节的数据包，内置 FIFO，支持中断和 DMA。

20.2 寄存器描述

USB 相关寄存器分为 3 个部分，部分寄存器是在主机和设备模式下进行复用的。

- USB 全局寄存器
- USB 设备控制寄存器
- USB 主机控制寄存器

20.2.1 全局寄存器描述

表 20-1 USBFS 相关寄存器列表

名称	访问地址	描述	复位值
R8_USB_CTRL	0x50000000	USB 控制寄存器	0x06
R8_USB_INT_EN	0x50000002	USB 中断使能寄存器	0x00
R8_USB_DEV_AD	0x50000003	USB 设备地址寄存器	0x00
R8_USB_MIS_ST	0x50000005	USB 杂项状态寄存器	0xXX
R8_USB_INT_FG	0x50000006	USB 中断标志寄存器	0x20
R8_USB_INT_ST	0x50000007	USB 中断状态寄存器	0xXX
R16_USB_RX_LEN	0x50000008	USB 接收长度寄存器	0xXX

20.2.1.1 USB 控制寄存器（R8_USB_CTRL）

位	名称	访问	描述	复位值
7	RB_UC_HOST_MODE	RW	USB 工作模式选择位： 1：主机模式（HOST）； 0：设备模式（DEVICE）。	0
6	RB_UC_LOW_SPEED	RW	USB 总线信号传输速率选择位： 1：1.5Mbps； 0：12Mbps。	0
5	RB_UC_DEV_PU_EN	RW	USB 设备模式下，USB 设备使能和内部上拉电阻控制位，为 1 则使能 USB 设备传输并且启用内部上拉电阻。	0
[5:4]	MASK_UC_SYS_CTRL	RW	见下表配置 USB 系统。	0
3	RB_UC_INT_BUSY	RW	USB 传输完成中断标志未清零前自动暂停	0

			使能位： 1：在中断标志 UIF_TRANSFER 未清零前自动暂停，设备模式下自动应答忙 NAK，主机模式下自动暂停后续传输； 0：不暂停。	
2	RB_UC_RST_SIE	RW	USB 协议处理器复位控制位： 1：关闭 USB 并复位 USB 协议处理器(SIE)，PA11/PA12 作为普通 GPIO； 0：启用 USB，PA11/PA12 作为 USB 专用引脚。	1
1	RB_UC_CLR_ALL	RW	USB 的 FIFO 和中断标志清零： 1：强制清空和清零； 0：不清。	1
0	RB_UC_DMA_EN	RW	USB 的 DMA 和 DMA 中断控制位： 1：使能 DMA 功能和 DMA 中断； 0：关闭 DMA。	0

当 RB_UC_RST_SIE=0 时，由 RB_UC_HOST_MODE 和 MASK_UC_SYS_CTRL 组成 USB 系统控制组合：

表 20-2 USB 系统控制组合

RB_UC_HOST_MODE	MASK_UC_SYS_CTRL	USB 系统控制描述
0	00	禁止 USB 设备功能，关闭内部上拉电阻。
0	01	使能 USB 设备功能，关闭内部上拉电阻，需加外部上拉。
0	1x	使能 USB 设备功能，启用内部 1.5K 上拉电阻。该上拉电阻优先于下拉电阻，也可用于 GPIO 模式。
1	00	USB 主机模式，正常工作状态。
1	01	USB 主机模式，强制 DP/DM 输出 SE0 状态。
1	10	USB 主机模式，强制 DP/DM 输出 J 状态。
1	11	USB 主机模式，强制 DP/DM 输出 K 状态/唤醒。

20.2.1.2 USB 中断使能寄存器 (R8_USB_INT_EN)

位	名称	访问	描述	复位值
7	Reserved	RO	保留	0
6	RB_UIE_DEV_NAK	RW	USB 设备模式，接收到 NAK 中断： 1：使能中断； 0：禁止中断。	0
5	RB_UID_1_WIRE	RW	USB 单线模式使能 1：开启； 0：禁止。	0
4	RB_UIE_FIFO_OV	RW	FIFO 溢出中断： 1：使能中断； 0：禁止中断。	0
3	RB_UIE_HST_SOF	RW	USB 主机模式，SOF 定时中断： 1：使能中断； 0：禁止中断。	0
2	RB_UIE_SUSPEND	RW	USB 总线挂起或唤醒事件中断： 1：使能中断； 0：禁止中断。	0
1	RB_UIE_TRANSFER	RW	USB 传输完成中断： 1：使能中断； 0：禁止中断。	0
0	RB_UIE_DETECT	RW	USB 主机模式，USB 设备连接或断开事件中断：	0

			1: 使能中断; 0: 禁止中断。	
	RB_UIE_BUS_RST	RW	USB 设备模式, USB 总线复位事件中断: 1: 使能中断; 0: 禁止中断。	0

20.2.1.3 USB 设备地址寄存器 (R8_USB_DEV_AD)

位	名称	访问	描述	复位值
7	RB_UDA_GP_BIT	RW	USB 通用标志位, 用户自定义。	0
[6:0]	MASK_USB_ADDR	RW	主机模式: 当前操作的 USB 设备地址; 设备模式: 该 USB 自身地址。	0

20.2.1.4 USB 杂项状态寄存器 (R8_USB_MIS_ST)

位	名称	访问	描述	复位值
7	RB_UMS_SOF_PRES	RO	USB 主机模式下 SOF 包预示状态位: 1: 将要发送 SOF 包, 此时如有其它 USB 数据包将被自动延后; 0: 无 SOF 包发送。	x
6	RB_UMS_SOF_ACT	RO	USB 主机模式下 SOF 包传输状态位: 1: 正在发出 SOF 包; 0: 发送完成或者空闲。	x
5	RB_UMS_SIE_FREE	RO	USB 协议处理器的空闲状态位: 1: 协议器空闲; 0: 忙, 正在进行 USB 传输。	1
4	RB_UMS_R_FIFO_RDY	RO	USB 接收 FIFO 数据就绪状态位: 1: 接收 FIFO 非空; 0: 接收 FIFO 为空。	0
3	RB_UMS_BUS_RST	RO	USB 总线复位状态位: 1: 当前 USB 总线处于复位态; 0: 当前 USB 总线处于非复位态。	x
2	RB_UMS_SUSPEND	RO	USB 挂起状态位: 1: USB 总线处于挂起态, 有一段时间没有 USB 活动; 0: USB 总线处于非挂起态。	0
1	RB_UMS_DM_LEVEL	RO	USB 主机模式下, 设备刚连入 USB 端口时 DM 引脚的电平状态, 用于判断速度: 1: 高电平/低速; 0: 低电平/全速。	0
0	RB_UMS_DEV_ATTACH	RO	USB 主机模式下端口的 USB 设备连接状态位: 1: 端口已经连接 USB 设备; 0: 端口没有 USB 设备连接。	0

20.2.1.5 USB 中断标志寄存器 (R8_USB_INT_FG)

位	名称	访问	描述	复位值
7	RB_U_IS_NAK	RO	USB 设备模式下, NAK 响应状态位: 1: 当前 USB 传输过程中回应 NAK;	0

			0: 无 NAK 响应。	
6	RB_U_TOG_OK	RO	当前 USB 传输 DATA0/1 同步标志匹配状态位: 1: 同步; 0: 不同步。	0
5	RB_U_SIE_FREE	RO	USB 协议处理器空闲状态位: 1: USB 空闲; 0: 忙, 正在进行 USB 传输。	1
4	RB_UIF_FIFO_OV	RW	USB FIFO 溢出中断标志位, 写 1 清零: 1: FIFO 溢出触发; 0: 无事件。	0
3	RB_UIF_HST_SOF	RW	USB 主机模式下 SOF 定时中断标志位, 写 1 清零: 1: SOF 包传输完成触发; 0: 无事件。	0
2	RB_UIF_SUSPEND	RW	USB 总线挂起或唤醒事件中断标志位, 写 1 清零: 1: USB 挂起事件或唤醒事件触发; 0: 无事件。	0
1	RB_UIF_TRANSFER	RW	USB 传输完成中断标志位, 写 1 清零: 1: 一个 USB 传输完成触发; 0: 无事件。	0
0	RB_UIF_DETECT	RW	USB 主机模式下 USB 设备连接或断开事件中断标志位, 写 1 清零: 1: 检测到 USB 设备连接或断开触发; 0: 无事件。	0
	RB_UIF_BUS_RST	RW	USB 设备模式下 USB 总线复位事件中断标志位, 写 1 清零: 1: USB 总线复位事件触发; 0: 无事件。	0

20.2.1.6 USB 中断状态寄存器 (R8_USB_INT_ST)

位	名称	访问	描述	复位值
7	RB_UIS_SETUP_ACT	RO	USB 设备模式下, 该位为 1 表示当前已成功接收到 8 字节的 SETUP 请求包, SETUP 令牌不影响 RB_UIS_TOG_OK 和 MASK_UIS_TOKEN、MASK_UIS_ENDP 及 RB_USB_RX_LEN。	0
6	RB_UIS_TOG_OK	RO	当前 USB 传输 DATA0/1 同步标志匹配状态位, 同 RB_U_TOG_OK: 1: 同步; 0: 不同步。	0
[5:4]	MASK_UIS_TOKEN	RO	设备模式下, 当前 USB 传输事务的令牌 PID 标识。	x
[3:0]	MASK_UIS_ENDP	RO	设备模式下, 当前 USB 传输事务的端点号。	x
	MASK_UIS_H_RES	RO	主机模式下, 当前 USB 传输事务的应答 PID 标识, 0000 表示设备无应答或超时; 其它值表示应答 PID。	x

MASK_UIS_TOKEN 用于 USB 设备模式下标识当前 USB 传输事务的令牌 PID：00 表示 OUT 包；01 保留；10 表示 IN 包；11 表示 SETUP 包。

MASK_UIS_H_RES 仅在主机模式下有效。在主机模式下，若主机发送 OUT/SETUP 令牌包时，则该 PID 是握手包 ACK/NAK/STALL，或是设备无应答/超时。若主机发送 IN 令牌包，则该 PID 是数据包的 PID（DATA0/DATA1）或握手包 PID。

20.2.1.7 USB 接收长度寄存器（R16_USB_RX_LEN）

位	名称	访问	描述	复位值
[15:10]	Reserved	R0	保留	0
[9:0]	R16_USB_RX_LEN	R0	当前 USB 端点接收的数据字节数	x

20.2.2 设备寄存器描述

USBFS 模块在 USB 设备模式下，提供了端点 0~7 共 8 组双向端点，除端点 3 之外的所有端点的最大数据包长度都是 64 字节，端点 3 的最大数据包长度为 1023 字节。

- 端点 0 是默认端点，支持控制传输，发送和接收共用一个 64 字节数据缓冲区。
- 端点 1~7 各自包括一个发送端点 IN 和一个接收端点 OUT，发送和接收各有一个独立的数据缓冲区，支持批量传输、中断传输和实时/同步传输。
- 端点 0 具有独立的 DMA 地址，收发共用，端点 1~7 的发送和接收各有一个 DMA 地址。通过入 R32_UEPn_BUF_MOD 寄存器可以设置数据缓冲区的模式为双缓冲或单缓冲。若使用双缓冲区模式，该端点只能使用单方向传输。
- 每组端点都具有收发控制寄存器 R8_UEPn_TX_CTRL、R8_UEPn_RX_CTRL 和发送长度寄存器 R16_UEPn_T_LEN 和 R32_UEPn*_DMA（n=0~7），用于配置该端点的同步触发位、对 OUT 事务和 IN 事务的响应以及发送数据的长度等。

作为 USB 设备所必要的 USB 总线上拉电阻可以由软件随时设置是否启用，当 USB 控制寄存器 R8_USB_CTRL 中的 RB_UC_DEV_PU_EN 置 1 时，控制器根据 RB_UC_SPEED_TYPE 的速度设置，在内部为 USB 总线的 DP/DM 引脚连接上拉电阻，并启用 USB 设备功能。

当检测到 USB 总线复位、USB 总线挂起或唤醒事件，或当 USB 成功处理完数据发送或数据接收后，USB 协议处理器都将设置相应的中断标志，如果中断使能打开，还会产生相应的中断请求。应用程序可以直接查询或在 USB 中断服务程序中查询并分析中断标志寄存器 R8_USB_INT_FG，根据 RB_UIF_BUS_RST 和 RB_UIF_SUSPEND 进行相应的处理；并且，如果 RB_UIF_TRANSFER 有效，那么还需要继续分析 USB 中断状态寄存器 R8_USB_INT_ST，根据当前端点号 MASK_UIS_ENDP 和当前事务令牌 PID 标识 MASK_UIS_TOKEN 进行相应的处理。如果事先设定了各个端点的 OUT 事务的同步触发位 RB_UEP_R_TOG，那么可以通过 RB_U_TOG_OK 或 RB_UIS_TOG_OK 判断当前所接收到的数据包同步触发位是否与该端点的同步触发位匹配，如果数据同步，则数据有效；如果数据不同步，则数据应该被丢弃。每次处理完 USB 发送或接收中断后，都应该正确修改相应端点的同步触发位，用于下次所发送的数据包或下次所接收的数据包是否同步检测；另外，设置 RB_UEP_T_TOG_AUTO 或 RB_UEP_R_TOG_AUTO 可以实现在发送成功或接收成功后自动修改相应的同步触发位（翻转或自减）。

各个端点准备发送的数据在各自的缓冲区中，准备发送的数据长度是独立设定在 R16_UEPn_T_LEN 中；各个端点接收到的数据在各自的缓冲区中，但是接收到的数据长度都在 USB 接收长度寄存器 R16_USB_RX_LEN 中，可以在 USB 接收中断时根据当前端点号区分。

表 20-3 设备相关寄存器列表

名称	访问地址	描述	复位值
R8_UDEV_CTRL	0x50000001	USB 设备物理端口控制寄存器	0xX0
R8_UEP4_1_MOD	0x5000000C	端点 1/4 模式控制寄存器	0x00
R8_UEP2_3_MOD	0x5000000D	端点 2/3 模式控制寄存器	0x00

R8_UEP5_6_MOD	0x5000000E	端点 5/6 模式控制寄存器	0x00
R8_UEP7_MOD	0x5000000F	端点 7 模式控制寄存器	0x00
R32_UEP0_DMA	0x50000010	端点 0 缓冲区起始地址	0x0000XXXX
R32_UEP1_DMA	0x50000014	端点 1 缓冲区起始地址	0x0000XXXX
R32_UEP2_DMA	0x50000018	端点 2 缓冲区起始地址	0x0000XXXX
R32_UEP3_DMA	0x5000001C	端点 3 缓冲区起始地址	0x0000XXXX
R32_UEP4_DMA	0x50000020	端点 4 缓冲区起始地址	0x0000XXXX
R32_UEP5_DMA	0x50000024	端点 5 缓冲区起始地址	0x0000XXXX
R32_UEP6_DMA	0x50000028	端点 6 缓冲区起始地址	0x0000XXXX
R32_UEP7_DMA	0x5000002C	端点 7 缓冲区起始地址	0x0000XXXX
R32_USB_EP0_CTRL	0x50000030	端点 0 发送长度和控制寄存器	0x000000XX
R16_UEP0_T_LEN	0x50000030	端点 0 发送长度寄存器	0x00XX
R16_UEP0_CTRL	0x50000032	端点 0 控制寄存器	0x0000
R32_USB_EP1_CTRL	0x50000034	端点 1 发送长度和控制寄存器	0x000000XX
R16_UEP1_T_LEN	0x50000034	端点 1 发送长度寄存器	0x00XX
R16_UEP1_CTRL	0x50000036	端点 1 控制寄存器	0x0000
R32_USB_EP2_CTRL	0x50000038	端点 2 发送长度和控制寄存器	0x000000XX
R16_UEP2_T_LEN	0x50000038	端点 2 发送长度寄存器	0x00XX
R16_UEP2_CTRL	0x5000003A	端点 2 控制寄存器	0x0000
R32_USB_EP3_CTRL	0x5000003C	端点 3 发送长度和控制寄存器	0x000000XX
R16_UEP3_T_LEN	0x5000003C	端点 3 发送长度寄存器	0x00XX
R16_UEP3_CTRL	0x5000003E	端点 3 控制寄存器	0x0000
R32_USB_EP4_CTRL	0x50000040	端点 4 发送长度和控制寄存器	0x000000XX
R16_UEP4_T_LEN	0x50000040	端点 4 发送长度寄存器	0x00XX
R16_UEP4_CTRL	0x50000042	端点 4 控制寄存器	0x0000
R32_USB_EP5_CTRL	0x50000044	端点 5 发送长度和控制寄存器	0x000000XX
R16_UEP5_T_LEN	0x50000044	端点 5 发送长度寄存器	0x00XX
R16_UEP5_CTRL	0x50000046	端点 5 控制寄存器	0x0000
R32_USB_EP6_CTRL	0x50000048	端点 6 发送长度和控制寄存器	0x000000XX
R16_UEP6_T_LEN	0x50000048	端点 6 发送长度寄存器	0x00XX
R16_UEP6_CTRL	0x5000004A	端点 6 控制寄存器	0x0000
R32_USB_EP7_CTRL	0x5000004C	端点 7 发送长度和控制寄存器	0x000000XX
R16_UEP7_T_LEN	0x5000004C	端点 7 发送长度寄存器	0x00XX
R16_UEP7_CTRL	0x5000004E	端点 7 控制寄存器	0x0000

20.2.2.1 USB 设备物理端口控制寄存器 (R8_UDEV_CTRL)

位	名称	访问	描述	复位值
7	RB_UD_PD_DIS	RW	USB 设备端口 UDP/UDM 引脚内部下拉电阻控制位： 1：禁用内部下拉； 0：使能内部下拉，也可用于 GPIO 模式提供下拉电阻。 <i>注：已改由 GPIOC_CFGXR 的 MODE 和 CNF 及 GPIOC_OUTDR 设置下拉，此位保留</i>	1
6	Reserved	RO	保留。	0

5	RB_UD_DP_PIN	R0	当前 UDP 引脚状态： 1：高电平； 0：低电平。	X
4	RB_UD_DM_PIN	R0	当前 UDM 引脚状态： 1：高电平； 0：低电平。	X
3	Reserved	R0	保留。	0
2	RB_UD_LOW_SPEED	RW	USB 设备物理端口低速模式使能位： 1：选择 1.5Mbps 低速模式； 0：选择 12Mbps 全速模式。	0
1	RB_UD_GP_BIT	RW	USB 设备模式通用标志位，用户自定义。	0
0	RB_UD_PORT_EN	RW	USB 设备物理端口使能位： 1：使能物理端口； 0：禁用物理端口。	0

20.2.2.2 端点 1/4 模式控制寄存器 (R8_UEP4_1_MOD)

位	名称	访问	描述	复位值
7	RB_UEP1_RX_EN	RW	1：使能端点 1 接收 (OUT)； 0：禁止端点 1 接收。	0
6	RB_UEP1_TX_EN	RW	1：使能端点 1 发送 (IN)； 0：禁止端点 1 发送。	0
5	Reserved	R0	保留。	0
4	RB_UEP1_BUF_MOD	RW	端点 1 数据缓冲区模式控制位。 注：该位为 1 时，UEP1_RX_EN 和 UEP1_TX_EN 不能同时为 1。	0
3	RB_UEP4_RX_EN	RW	1：使能端点 4 接收 (OUT)； 0：禁止端点 4 接收。	0
2	RB_UEP4_TX_EN	RW	1：使能端点 4 发送 (IN)； 0：禁止端点 4 发送。	0
1	Reserved	R0	保留。	0
0	RB_UEP4_BUF_MOD	RW	端点 4 数据缓冲区模式控制位。 注：该位为 1 时，UEP4_RX_EN 和 UEP4_TX_EN 不能同时为 1。	0

20.2.2.3 端点 2/3 模式控制寄存器 (R8_UEP2_3_MOD)

位	名称	访问	描述	复位值
7	RB_UEP3_RX_EN	RW	1：使能端点 3 接收 (OUT)； 0：禁止端点 3 接收。	0
6	RB_UEP3_TX_EN	RW	1：使能端点 3 发送 (IN)； 0：禁止端点 3 发送。	0
5	Reserved	R0	保留。	0
4	RB_UEP3_BUF_MOD	RW	端点 3 数据缓冲区模式控制位。 注：该位为 1 时，UEP3_RX_EN 和 UEP3_TX_EN 不能同时为 1。	0
3	RB_UEP2_RX_EN	RW	1：使能端点 2 接收 (OUT)；	0

			0: 禁止端点 2 接收。	
2	RB_UEP2_TX_EN	RW	1: 使能端点 2 发送 (IN) ; 0: 禁止端点 2 发送。	0
1	Reserved	RO	保留。	0
0	RB_UEP2_BUF_MOD	RW	端点 2 数据缓冲区模式控制位。 注: 该位为 1 时, UEP2_RX_EN 和 UEP2_TX_EN 不能同时为 1。	0

20.2.2.4 端点 5/6 模式控制寄存器 (R8_UEP5_6_MOD)

位	名称	访问	描述	复位值
7	RB_UEP6_RX_EN	RW	1: 使能端点 6 接收 (OUT) ; 0: 禁止端点 6 接收。	0
6	RB_UEP6_TX_EN	RW	1: 使能端点 6 发送 (IN) ; 0: 禁止端点 6 发送。	0
5	Reserved	RO	保留。	0
4	RB_UEP6_BUF_MOD	RW	端点 6 数据缓冲区模式控制位。 注: 该位为 1 时, UEP6_RX_EN 和 UEP6_TX_EN 不能同时为 1。	0
3	RB_UEP5_RX_EN	RW	1: 使能端点 5 接收 (OUT) ; 0: 禁止端点 5 接收。	0
2	RB_UEP5_TX_EN	RW	1: 使能端点 5 发送 (IN) ; 0: 禁止端点 5 发送。	0
1	Reserved	RO	保留。	0
0	RB_UEP5_BUF_MOD	RW	端点 5 数据缓冲区模式控制位。 注: 该位为 1 时, UEP5_RX_EN 和 UEP5_TX_EN 不能同时为 1	0

20.2.2.5 端点 7 模式控制寄存器 (R8_UEP7_MOD)

位	名称	访问	描述	复位值
[7:4]	Reserved	RO	保留。	0
3	RB_UEP7_RX_EN	RW	1: 使能端点 7 接收 (OUT) ; 0: 禁止端点 7 接收。	0
2	RB_UEP7_TX_EN	RW	1: 使能端点 7 发送 (IN) ; 0: 禁止端点 7 发送。	0
1	Reserved	RO	保留。	0
0	RB_UEP7_BUF_MOD	RW	端点 7 数据缓冲区模式控制位。	0

由 RB_UEPn_RX_EN 和 RB_UEPn_TX_EN 以及 RB_UEPn_BUF_MOD 组合分别配置 USB 端点 1-15 的数据缓冲区模式, 具体参考表 20-4。其中, 在双 64 字节缓冲区模式下, USB 数据传输时将根据 RB_UEP*_TOG=0 选择前 64 字节缓冲区, 根据 RB_UEP*_TOG=1 选择后 64 字节缓冲区, 设置 RB_UEP_AUTO_TOG=1 可实现自动切换。

表 20-4 端点 n 缓冲区模式 (n=1-7)

RB_UEPn_RX_EN	RB_UEPn_TX_EN	RB_UEPn_BUF_MOD	描述: 以 R16_UEPn_DMA 为起始地址由低向高排列
0	0	X	端点被禁用, 未用到 R16_UEPn_DMA 缓冲区。

1	0	0	单 64 字节接收缓冲区 (OUT)。
1	0	1	双 64 字节接收缓冲区 (OUT)，由 RB_UEP_R_TOG 选择。
0	1	0	单 64 字节发送缓冲区 (IN)。
0	1	1	双 64 字节发送缓冲区 (IN)，由 RB_UEP_T_TOG 选择。
1	1	0	单 64 字节接收缓冲区 (OUT)，单 64 字节发送缓冲区 (IN)。
1	1	1	双 64 字节接收缓冲区 (OUT)，通过 RB_UEP_R_TOG 选择， 双 64 字节发送缓冲区 (IN)，通过 RB_UEP_T_TOG 选择。 全部 256 字节排列如下： UEPn_DMA+0 地址：RB_UEP_R_TOG=0 时端点接收地址； UEPn_DMA+64 地址：RB_UEP_R_TOG=1 时端点接收地址； UEPn_DMA+128 地址：RB_UEP_T_TOG=0 时端点发送地址； UEPn_DMA+192 地址：RB_UEP_T_TOG=1 时端点发送地址。

20.2.2.6 端点 n 缓冲区起始地址 (R32_UEPn_DMA) (n=0-7)

位	名称	访问	描述	复位值
[31:15]	Reserved	R0	保留	0
[14:0]	UEPn_DMA	RW	端点 n 缓冲区起始地址。 低 15 位有效，地址必须 4 字节对齐。	x

20.2.2.7 端点 n 发送长度和控制寄存器 (R32_USB_EPn_CTRL) (n=0-7)

位	名称
[31:16]	R16_UEPn_CTRL
[15:0]	R16_UEPn_T_LEN

20.2.2.8 端点 n 发送长度寄存器 (R16_UEPn_T_LEN) (n=0-1)

位	名称	访问	描述	复位值
[15:7]	Reserved	R0	保留。	0
[6:0]	R8_UEPn_T_LEN	RW	设置 USB 端点 n 准备发送的数据字节数 n=0、1。	x

20.2.2.9 端点 n 发送长度寄存器 (R16_UEPn_T_LEN) (n=2)

位	名称	访问	描述	复位值
[15:8]	Reserved	R0	保留。	0
7	HOST_PID3	RW	主机模式下的 PID[3]。	0
[6:0]	R8_UEPn_T_LEN	RW	设置 USB 端点 n 准备发送的数据字节数 n=2。	x

20.2.2.10 端点 n 发送长度寄存器 (R16_UEPn_T_LEN) (n=3)

位	名称	访问	描述	复位值
[15:10]	Reserved	R0	保留。	0
[9:0]	R8_UEPn_T_LEN	RW	设置 USB 端点 n 准备发送的数据字节数 n=3。	x

20.2.2.11 端点 n 发送长度寄存器 (R16_UEPn_T_LEN) (n=4-7)

位	名称	访问	描述	复位值
[15:7]	Reserved	R0	保留	0
[6:0]	R8_UEPn_T_LEN	RW	设置 USB 端点 n 准备发送的数据字节数 n=4、5、6、7。	x

20.2.2.12 端点 n 控制寄存器 (R16_UEPn_CTRL) (n=0-7)

位	名称	访问	描述	复位值
[15:12]	Reserved	R0	保留。	0
11	RB_UEP_R_AUTO_TOG	RW	同步触发位自动翻转使能控制位： 1：数据接收成功后自动翻转相应的同步触发位； 0：不自动翻转，可以手动切换。 注：端点 0 此位保留。	0
10	MASK_UEP_R_TOG	RW	USB 端点 n 的接收器（处理 OUT 事务）期望的同步触发位： 1：期望 DATA1； 0：期望 DATA0。	0
[9:8]	MASK_UEP_R_RES	RW	端点 n 的接收器对 OUT 事务的响应控制： 00：应答 ACK； 01：超时/无响应，用于非端点 0 的实时/同步传输； 10：应答 NAK 或忙； 11：应答 STALL 或错误。	00b
[7:4]	Reserved	R0	保留。	0
3	RB_UEP_T_AUTO_TOG	RW	同步触发位自动翻转使能控制位： 1：数据发送成功后自动翻转相应的同步触发位； 0：不自动翻转，可以手动切换。 注：端点 0 此位保留。	0
2	RB_UEP_T_TOG	RW	USB 端点 n 的发送器（处理 IN 事务）准备的同步触发位： 1：发送 DATA1； 0：发送 DATA0。	0
[1:0]	MASK_UEP_T_RES	RW	端点 n 的发送器对 IN 事务的响应控制： 00：DATA0/DATA1 数据就绪并期望 ACK； 01：应答 DATA0/DATA1 并期望无响应，用于非端点 0 的实时/同步传输； 10：应答 NAK 或忙； 11：应答 STALL 或错误。	00b

20.2.3 USB 主机寄存器

在 USB 主机模式下，芯片提供了一组双向主机端点，包括一个发送端点 OUT 和一个接收端点 IN，一个数据包的最大长度是 1024 字节（同步传输），支持控制传输、中断传输、批量传输和实时/同步传输。

主机端点发起的每一个 USB 事务，在处理结束后总是自动设置 RB_UIF_TRANSFER 中断标志。应用程序可以直接查询或在 USB 中断服务程序中查询并分析中断标志寄存器 R8_USB_INT_FG，根据各中断标志分别进行相应的处理；并且，如果 RB_UIF_TRANSFER 有效，那么还需要继续分析 USB 中断状态寄存器 R8_USB_INT_ST，根据当前 USB 传输事务的应答 PID 标识 MASK_UIS_H_RES 进行相应的处

理。

如果事先设定了主机接收端点的 IN 事务的同步触发位 (RB_UH_R_TOG)，那么可以通过 RB_U_TOG_OK 或 RB_UIS_TOG_OK 判断当前所接收到的数据包的同步触发位是否与主机接收端点的同步触发位匹配，如果数据同步，则数据有效；如果数据不同步，则数据应该被丢弃。每次处理完 USB 发送或接收中断后，都应该正确修改相应主机端点的同步触发位，用于同步下次所发送的数据包和检测下次所接收的数据包是否同步；另外，通过设置 RB_UH_T_AUTO_TOG 和 RB_UH_R_AUTO_TOG 可以实现在发送成功或接收成功后自动翻转相应的同步触发位。

USB 主机令牌设置寄存器 R8_UH_EP_PID 用于设置被操作的目标设备的端点号和本次 USB 传输事务的令牌 PID 包标识。SETUP 令牌和 OUT 令牌所对应的数据由主机发送端点提供，准备发送的数据在 R16_UH_TX_DMA 缓冲区中，准备发送的数据长度设置在 R16_UH_TX_LEN 中；IN 令牌所对应数据由目标设备返回给主机接收端点，接收到数据存放 R16_UH_RX_DMA 缓冲区中，接收到的数据长度存放在 R16_USB_RX_LEN 中。

表 20-5 主机相关寄存器列表

名称	访问地址	描述	复位值
R8_UHOST_CTRL	0x50000001	USB 主机物理端口控制寄存器	0xX0
R8_UH_EP_MOD	0x5000000D	USB 主机端点模式控制寄存器	0x00
R16_UH_RX_DMA	0x50000018	USB 主机接收缓冲区起始地址	X
R16_UH_TX_DMA	0x5000001C	USB 主机发送缓冲区起始地址	X
R16_UH_SETUP	0x50000036	USB 主机辅助设置寄存器	0x0000
R8_UH_EP_PID	0x50000038	USB 主机令牌设置寄存器	0x00
R8_UH_RX_CTRL	0x5000003B	USB 主机接收端点控制寄存器	0x00
R16_UH_TX_LEN	0x5000003C	USB 主机发送长度寄存器	X
R8_UH_TX_CTRL	0x5000003E	USB 主机发送端点控制寄存器	0x00

20.2.3.1 USB 主机物理端口控制寄存器 (R8_UHOST_CTRL)

位	名称	访问	描述	复位值
7	RB_UH_PD_DIS	RW	USB 主机端口 UD+/UD-引脚内部 15K 下拉电阻控制位： 1：禁用内部下拉； 0：使能内部下拉。 也可用于 GPIO 模式提供 15K 下拉电阻。	1
6	Reserved	RO	保留。	0
5	RB_UH_DP_PIN	RO	当前 UD+引脚状态： 1：高电平； 0：低电平。	x
4	RB_UH_DM_PIN	RO	当前 UD-引脚状态： 1：高电平； 0：低电平。	x
3	Reserved	RO	保留。	0
2	RB_UH_LOW_SPEED	RW	USB 主机端口低速模式使能位： 1：选择 1.5Mbps 低速模式； 0：选择 12Mbps 全速模式。	0
1	RB_UH_BUS_RESET	RW	USB 主机模式总线复位控制位： 1：强制输出 USB 总线复位； 0：结束输出。	0
0	RB_UH_PORT_EN	RW	USB 主机端口使能位： 1：使能主机端口；	0

			0: 禁用主机端口。 当 USB 设备断开连接时, 该位自动清 0。	
--	--	--	---------------------------------------	--

20.2.3.2 USB 主机端点模式控制寄存器 (R8_UH_EP_MOD)

位	名称	访问	描述	复位值
7	Reserved	R0	保留。	0
6	RB_UH_EP_TX_EN	RW	主机发送端点发送 (SETUP/OUT) 使能位: 1: 使能端点发送; 0: 禁止端点发送。	0
5	Reserved	R0	保留。	0
4	RB_UH_EP_TBUF_MOD	RW	主机发送端点发送数据缓冲区模式控制位。	0
3	RB_UH_EP_RX_EN	RW	主机接收端点接收 (IN) 使能位: 1: 使能端点接收; 0: 禁止端点接收。	0
[2:1]	Reserved	R0	保留。	00b
0	RB_UH_EP_RBUF_MOD	RW	USB 主机接收端点接收数据缓冲区模式控制位。	0

由 RB_UH_EP_TX_EN 和 RB_UH_EP_TBUF_MOD 组合控制主机发送端点数据缓冲区模式, 参考下表。

表 20-6 主机发送缓冲区模式

RB_UH_EP_TX_EN	RB_UH_EP_TBUF_MOD	描述: 以 R16_UH_TX_DMA 为起始地址
0	X	端点被禁用, 未用到 R16_UH_TX_DMA 缓冲区。
1	0	单 64 字节发送缓冲区 (SETUP/OUT)。
1	1	双 64 字节发送缓冲区, 通过 RB_UH_T_TOG 选择: 当 RB_UH_T_TOG=0 时选择前 64 字节缓冲区; 当 RB_UH_T_TOG=1 时选择后 64 字节缓冲区。

由 RB_UH_EP_RX_EN 和 RB_UH_EP_RBUF_MOD 组合控制主机接收端点数据缓冲区模式, 参考下表。

表 20-7 主机接收缓冲区模式

RB_UH_EP_RX_EN	RB_UH_EP_RBUF_MOD	结构描述: 以 R16_UH_TX_DMA 为起始地址
0	X	端点被禁用, 未用到 R16_UH_RX_DMA 缓冲区。
1	0	单 64 字节接收缓冲区 (IN)。
1	1	双 64 字节接收缓冲区, 通过 RB_UH_R_TOG 选择: 当 RB_UH_R_TOG=0 时选择前 64 字节缓冲区; 当 RB_UH_R_TOG=1 时选择后 64 字节缓冲区。

20.2.3.3 USB 主机接收缓冲区起始地址 (R16_UH_RX_DMA)

位	名称	访问	描述	复位值
[15:0]	R16_UH_RX_DMA	RW	主机端点数据接收缓冲区起始地址。 低 15 位有效, 地址必须 4 字节对齐。	XXXXh

20.2.3.4 USB 主机发送缓冲区起始地址 (R16_UH_TX_DMA)

位	名称	访问	描述	复位值
[15:0]	R16_UH_TX_DMA	RW	主机端点数据发送缓冲区起始地址。 低 15 位有效, 地址必须 4 字节对齐。	XXXXh

20.2.3.5 USB 主机辅助设置寄存器(R16_UH_SETUP)

位	名称	访问	描述	复位值
[15:11]	Reserved	R0	保留。	0
10	RB_UH_PRE_PID_EN	RW	低速前导包 PRE PID 使能位： 1：使能，用于通过外部 HUB 与低速 USB 设备通讯。 0：禁用低速前导包。	0
[9:3]	Reserved	R0	保留。	0
2	RB_UH_SOF_EN	RW	自动产生 SOF 包使能位： 1：主机自动产生 SOF 包； 0：不自动产生，但可手工产生。	0
[1:0]	Reserved	R0	保留。	0

20.2.3.6 USB 主机令牌设置寄存器(R8_UH_EP_PID)

位	名称	访问	描述	复位值
[7:4]	MASK_UH_TOKEN	RW	设置本次 USB 传输事务的令牌 PID 包标识。	0000b
[3:0]	MASK_UH_ENDP	RW	设置本次被操作的目标设备的端点号。	0000b

20.2.3.7 USB 主机接收端点控制寄存器(R8_UH_RX_CTRL)

位	名称	访问	描述	复位值
[7:4]	Reserved	R0	保留。	0
3	RB_UH_R_AUTO_TOG	RW	同步触发位自动翻转使能控制位： 1：数据接收成功后自动翻转相应的期待同步触发位 (RB_UH_R_TOG)； 0：不自动翻转，可以手动切换。	0
2	RB_UH_R_TOG	RW	USB 主机接收器（处理 IN 事务）期望的同步触发位： 1：期望 DATA1； 0：期望 DATA0。	0
1	Reserved	R0	保留。	0
0	RB_UH_R_RES	RW	主机接收器对 IN 事务的响应控制位： 1：无响应，用于非 0 端点的实时/同步传输； 0：应答 ACK。	0

20.2.3.8 USB 主机发送长度寄存器(R16_UH_TX_LEN)

位	名称	访问	描述	复位值
[15:0]	R8_UH_TX_LEN	RW	设置 USB 主机发送端点准备发送的数据字节数。	XXh

20.2.3.9 USB 主机发送端点控制寄存器(R8_UH_TX_CTRL)

位	名称	访问	描述	复位值
---	----	----	----	-----

[7:4]	Reserved	RO	保留。	0
3	RB_UH_T_AUTO_TOG	RW	同步触发位自动翻转使能控制位： 1：数据发送成功后自动翻转相应的同步触发位 (RB_UH_T_TOG)； 0：不自动翻转，可以手动切换。	0
2	RB_UH_T_TOG	RW	USB 主机发送器（处理 SETUP/OUT 事务）准备的同步触发位： 1：表示发送 DATA1； 0：表示发送 DATA0。	0
1	Reserved	RO	保留。	0
0	RB_UH_T_RES	RW	USB 主机发送器对 SETUP/OUT 事务的响应控制位： 1：期望无响应，用于非 0 端点的实时/同步传输； 0：期望应答 ACK。	0

第 21 章 USB PD 控制器（USBPD）

21.1 USB PD 控制器简介

芯片内置 USB Power Delivery 控制器和 PD 收发器 PHY，支持 USB type-C 主从检测，自动 BMC 编解码和 CRC，硬件边沿控制，支持 USB PD2.0 和 PD3.0 电力传输控制，支持快充，支持 UFP/DFP/DRD/DRP 和 PDUSB，支持 PD 受电端和 PD 供电端应用。

- 内置 USB type-C 接口，支持主从检测，支持 DRP、Sink/Consumer 和 Source/Provider；
- 内置 USB PD 收发器 PHY，集成硬件边沿斜率控制；
- 内置 USB Power Delivery 控制器，自动 BMC 编解码、4b5b 编解码和 CRC；
- 支持 S0P、S0P'、S0P'' 等 PD 包，支持 USB PD 复位信号帧硬件复位；
- 支持最大包长度 510 字节，支持 DMA；
- 支持 USB PD 2.0 和 3.0 电力传输协议，USB 端口支持 BC 等充电协议。

21.2 寄存器描述

表 21-1 USBPD 相关寄存器列表

名称	访问地址	描述	复位值
R32_USBPD_CONFIG	0x40027000	PD 配置寄存器	0x00000X02
R16_CONFIG	0x40027000	PD 中断使能寄存器	0x0X02
R16_BMC_CLK_CNT	0x40027002	BMC 采样时钟计数器	0x0000
R32_USBPD_CONTROL	0x40027004	PD 控制寄存器	0x00000000
R16_CONTROL	0x40027004	PD 收发控制寄存器	0x0000
R8_CONTROL	0x40027004	PD 收发使能寄存器	0x00
R8_TX_SEL	0x40027005	PD 发送 SOP 选择寄存器	0x00
R16_BMC_TX_SZ	0x40027006	PD 发送长度寄存器	0x0000
R32_USBPD_STATUS	0x40027008	PD 状态寄存器	0x000000XX
R16_STATUS	0x40027008	PD 中断和数据寄存器	0x00XX
R8_DATA_BUF	0x40027008	DMA 缓存数据寄存器	0xXX
R8_STATUS	0x40027009	PD 中断标志寄存器	0x00
R16_BMC_BYTE_CNT	0x4002700A	字节计数器	0x0000
R32_USBPD_PORT	0x4002700C	端口控制寄存器	0x00030003
R16_PORT_CC1	0x4002700C	CC1 端口控制寄存器	0x0003
R16_PORT_CC2	0x4002700E	CC2 端口控制寄存器	0x0003
R32_USBPD_DMA	0x40027010	DMA 缓存地址寄存器	0x0000XXXX
R16_DMA	0x40027010	PD 缓冲区起始地址寄存器	0xFFFF

21.2.1 PD 配置寄存器（R32_USBPD_CONFIG）

偏移地址：0x00

位	名称
[31:16]	R16_BMC_CLK_CNT
[15:0]	R16_CONFIG

21.2.2 PD 中断使能寄存器（R16_CONFIG）

偏移地址：0x00

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IE_TX_END	IE_RX_RESET	IE_RX_ACT	IE_RX_BYTE	IE_RX_BYTE	IE_PD_I0	RTX_BIT0	MULTI_0	Reserved		WAKE_POLAR	PD_RST_EN	PD_DMA_EN	CC_SEL	PD_ALL_CLR	Reserved

位	名称	访问	描述	复位值
15	IE_TX_END	RW	发送结束中断使能。	0
14	IE_RX_RESET	RW	接收复位中断使能。	0
13	IE_RX_ACT	RW	接收完成中断使能。	0
12	IE_RX_BYTE	RW	接收字节中断使能。	0
11	IE_RX_BIT	RW	接收 bit 中断使能。	0
10	IE_PD_I0	RW	PD I0 中断使能。	0
9	RTX_BIT0	RO	收发移位寄存器位 0 的当前值： 0：当前时刻位 0 的值为 1； 1：当前时刻位 0 的值为 0。	x
8	MULTI_0	RO	接收到连续多 bit 的 0 指示信号，此位为 1 表示已接收到连续 5bit 的 0 值。	0
[7:6]	Reserved	RO	保留。	0
5	WAKE_POLAR	RW	PD 端口唤醒电平： 0：低电平有效； 1：高电平有效。	0
4	PD_RST_EN	RW	PD 模式复位命令使能： 0：无效； 1：复位。	0
3	PD_DMA_EN	RW	使能 USBPD 的 DMA，正常传输模式下该位必须设置为 1： 0：关闭 DMA； 1：使能 DMA 功能和 DMA 中断。	0
2	CC_SEL	RW	选择当前 PD 通讯端口： 0：使用 CC1 端口通讯； 1：使用 CC2 端口通讯	0
1	PD_ALL_CLR	RW	PD 模式清除所有的中断标志位： 0：无效； 1：清除中断标志位。	1
0	Reserved	RO	保留。	0

21.2.3 BMC 采样时钟计数器 (R16_BMC_CLK_CNT)

偏移地址：0x02

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

位	名称	访问	描述	复位值
[15:9]	Reserved	RO	保留。	0

[8:0]	BMC_CLK_CNT	RW	BMC 发送或者接收采样时钟计数器。	0
-------	-------------	----	--------------------	---

21.2.4 PD 控制寄存器 (R32_USBPD_CONTROL)

偏移地址: 0x04

位	名称
[31:16]	R16_BMC_TX_SZ
[15:0]	R16_CONTROL

21.2.5 PD 收发控制寄存器 (R16_CONTROL)

偏移地址: 0x04

位	名称
[15:8]	R8_TX_SEL
[7:0]	R8_CONTROL

21.2.6 PD 收发使能寄存器 (R8_CONTROL)

偏移地址: 0x04

7	6	5	4	3	2	1	0
RX_ST_H	RX_ST_L	DATA_FLAG	Reserved			BMC_START	PD_TX_EN

位	名称	访问	描述	复位值
7	RX_ST_H	RO	接收状态高位。	0
6	RX_ST_L	RO	接收状态低位。	0
5	DATA_FLAG	RO	缓存数据有效标志位。	0
[4:2]	Reserved	RO	保留。	0
1	BMC_START	RW	BMC 发送开始信号。	0
0	PD_TX_EN	RW	USBPD 收发模式和发送使能: 0: PD 接收使能; 1: PD 发送使能。	0

21.2.7 PD 发送 SOP 选择寄存器 (R8_TX_SEL)

偏移地址: 0x05

7	6	5	4	3	2	1	0
TX_SEL4		TX_SEL3		TX_SEL2		Reserved	TX_SEL1

位	名称	访问	描述	复位值
[7:6]	TX_SEL4	RW	PD 发送模式下 K-CODE4 类型选择: 00: SYNC2; 01: SYNC3; 1x: RST2。	0
[5:4]	TX_SEL3	RW	PD 发送模式下 K-CODE3 类型选择: 00: SYNC1; 01: SYNC3;	0

			1x: RST1。	
[3:2]	TX_SEL2	RW	PD 发送模式下 K-CODE2 类型选择: 00: SYNC1; 01: SYNC3; 1x: RST1。	0
1	Reserved	RO	保留。	0
0	TX_SEL1	RW	PD 发送模式下 K-CODE1 类型选择: 0: SYNC1; 1: RST1。	0

21.2.8 PD 发送长度寄存器 (R16_BMC_TX_SZ)

偏移地址: 0x06

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

位	名称	访问	描述	复位值
[15:9]	Reserved	RO	保留。	0
[8:0]	BMC_TX_SZ	RW	PD 模式下发送的总长度。	0

21.2.9 PD 状态寄存器 (R32_USBPDP_STATUS)

偏移地址: 0x08

位	名称
[31:16]	R16_BMC_BYTE_CNT
[15:0]	R16_STATUS

21.2.10 PD 中断和数据寄存器 (R16_STATUS)

偏移地址: 0x08

位	名称
[15:8]	R8_STATUS
[7:0]	R8_DATA_BUF

21.2.11 DMA 缓存数据寄存器 (R8_DATA_BUF)

偏移地址: 0x08

7	6	5	4	3	2	1	0
DATA_BUF							

位	名称	访问	描述	复位值
[7:0]	DATA_BUF	RO	DMA 缓存数据。	X

21.2.12 PD 中断标志寄存器 (R8_STATUS)

偏移地址: 0x09

7	6	5	4	3	2	1	0
IF_TX_END	IF_RX_RESET	IF_RX_ACT	IF_RX_BYTE	IF_RX_BIT	BUF_ERR	BMC_AUX	

位	名称	访问	描述	复位值
7	IF_TX_END	RW1	传送完成中断标志, 写 1 清 0, 写 0 无效。	0
6	IF_RX_RESET	RW1	接收复位中断标志, 写 1 清 0, 写 0 无效。	0
5	IF_RX_ACT	RW1	接收完成中断标志, 写 1 清 0, 写 0 无效。	0
4	IF_RX_BYTE	RW1	接收字节或者 SOP 中断标志, 写 1 清 0, 写 0 无效。	0
3	IF_RX_BIT	RW1	接收 bit 或者 5bit 中断标志, 写 1 清 0, 写 0 无效。	0
2	BUF_ERR	RW1	BUFFER 或者 DMA 错误中断标志, 写 1 清 0, 写 0 无效。	0
[1:0]	BMC_AUX	R0	指示当前 PD 状态: 在 PD 接收时或者接收完成后, 状态如下: 00: 接收空闲或者没有接收到有效的数据包; 01: 接收到 SOP 即 SOP0; 10: 接收到 SOP' 即 SOP1 或者 Hard Reset; 11: 接收到 SOP'' 即 SOP2 或者 Cable Reset。 在 PD 发送时, 状态如下: 00: 正在发送CRC32[7:0]; 01: 正在发送CRC32[15:8]; 10: 正在发送 CRC32[23:16]; 11: 正在发送 CRC32[31:24]。	00

21.2.13 字节计数器 (R16_BMC_BYTE_CNT)

偏移地址: 0x0A

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

位	名称	访问	描述	复位值
[15:9]	Reserved	R0	保留。	0
[8:0]	BMC_BYTE_CNT	R0	字节计数器。	0

21.2.14 端口控制寄存器 (R32_USBDPD_PORT)

偏移地址: 0x0C

位	名称
[15:8]	R16_PORT_CC2
[7:0]	R16_PORT_CC1

21.2.15 CC1 端口控制寄存器 (R16_PORT_CC1)

偏移地址: 0x0C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								CC1_CE			CC1_LVE	CC1_PU	CC1_PD	PA_CC1_AI	

位	名称	访问	描述	复位值
[15:8]	Reserved	RO	保留。	0
[7:5]	CC1_CE	RW	CC1 端口电压比较器的使能： 000：关闭； 001：保留； 010：0.22V； 011：0.43V； 100：0.55V； 101：0.66V； 110：0.96V； 111：1.23V。	000
4	CC1_LVE	RW	CC1 端口输出低电压使能： 0：正常 VDD 电压驱动输出； 1：低电压驱动输出。	0
[3:2]	CC1_PU	RW	CC1 端口上拉电流选择： 00：禁止上拉电流； 01：330 μ A； 10：180 μ A； 11：80 μ A。 注：（1）CC 端口的上拉电流独立于 GPIO，可单独控制； （2）CHIPID 倒数第二位为 1 的芯片，在作为 Source 使用时，端口配置为上拉输入。	00
1	CC1_PD	RW	CC1 端口下拉电阻 Rd 使能： 0：关闭下拉电阻（注：关闭后仍有约 800k Ω 弱下拉） 1：启用 Rd 下拉电阻，约 5.1K Ω 。 注：CC 端口的下拉电阻独立于 GPIO，可单独控制，但部分封装形式的芯片未内置 Rd，具体参考 CH32L103DS0.PDF 中选型表	1
0	PA_CC1_AI	RO	CC1 端口比较器模拟输入。	1

21.2.16 CC2 端口控制寄存器（R16_PORT_CC2）

偏移地址：0x0E

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								CC2_CE		CC2_LVE	CC2_PU	CC2_PD	PA_CC2_AI		

位	名称	访问	描述	复位值
[15:8]	Reserved	RO	保留。	0
[7:5]	CC2_CE	RW	CC2 端口电压比较器的使能： 000：关闭； 001：保留； 010：0.22V； 011：0.43V； 100：0.55V；	000

			101: 0.66V; 110: 0.96V; 111: 1.23V。	
4	CC2_LVE	RW	CC2 端口输出低电压使能: 0: 正常 VDD 电压驱动输出; 1: 低电压驱动输出。	0
[3:2]	CC2_PU	RW	CC2 端口上拉电流选择: 00: 禁止上拉电流; 01: 330 μ A; 10: 180 μ A; 11: 80 μ A。 注: (1) CC 端口的上拉电流独立于 GPIO, 可单独控制; (2) CHIPID 倒数第二位为 1 的芯片, 在作为 Source 使用时, 端口配置为上拉输入	00
1	CC2_PD	RW	CC2 端口下拉电阻 Rd 使能: 0: 关闭下拉电阻 (注: 关闭后仍有约 800k Ω 弱下拉); 1: 启用 Rd 下拉电阻, 约 5.1K Ω 。 注: CC 端口的下拉电阻独立于 GPIO, 可单独控制, 但部分封装形式的芯片未内置 Rd, 具体参考 CH32L103DS0.PDF 中选型表	1
0	PA_CC2_AI	RO	CC2 端口比较器模拟输入。	1

21.2.17 DMA 缓存地址寄存器 (R32_USBPD_DMA)

偏移地址: 0x10

位	名称
[31:15]	Reserved
[15:0]	R16_DMA

21.2.18 PD 缓冲区起始地址寄存器 (R16_DMA)

偏移地址: 0x10

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

位	名称	访问	描述	复位值
[15:0]	USBPD_DMA_ADDR	RW	USBPD_DMA 缓存地址。 低 16 位有效, 地址必须 4 字节对齐。	X

第 22 章 控制器局域网（CAN）

控制器局域网是一种用于串行数据通信的高性能通信协议。CAN 控制器提供了一个完整的 CAN 协议实现方案，支持 CAN 协议 2.0A 和 2.0B。CAN 控制器可以用来构建强大的局域网来实现安全的分布式实时控制，以较小的 CPU 负荷来处理大量的数据报文，在工业和汽车领域有着广泛的应用。

22.1 主要特性

- 兼容 CAN 规范 2.0A 和 2.0B
- 可编程的传输速率，最高可达 8Mbit/s
- 支持时间触发通信功能，避免低优先级消息阻塞
- 支持三个发送邮箱，发送报文优先级可由报文标识符或发送请求的次序决定，并可记录发送报文 SOF 时刻的时间戳
- 支持三级邮箱深度的 2 个接收 FIFO，14 个报文过滤器组可供配置，每个过滤器组可配置成 32 或 16 位模式，屏蔽位或标识符列表模式，能够尽量减少软件对报文筛选的干预，FIFO 溢出处理方式灵活，并可记录接收报文 SOF 时刻的时间戳
- 占用 4 个中断向量，每个中断源可以独立配置

22.2 CAN 控制器工作模式

CAN 控制器可以对寄存器 CAN_CTLR 中的 SLEEP 或 INRQ 位进行操作，实现在初始化模式、睡眠模式和正常模式 3 个工作模式下切换。

22.2.1 初始化模式

在复位后，CAN 默认工作在睡眠模式以减低功耗，此时禁止报文收发，TX 引脚的内部上拉电阻使能，TX 引脚输出隐性位。对寄存器 CAN_CTLR 中的 INRQ 位置 1，请求 CAN 控制器进入初始化模式，当寄存器 CAN_STATR 的 INAK 位自动置 1 则成功进入初始化状态。同样对寄存器 CAN_CTLR 中的 INRQ 位清零，请求 CAN 控制器退出初始化模式，当寄存器 CAN_STATR 的 INAK 位自动清 0 则成功退出初始化状态。

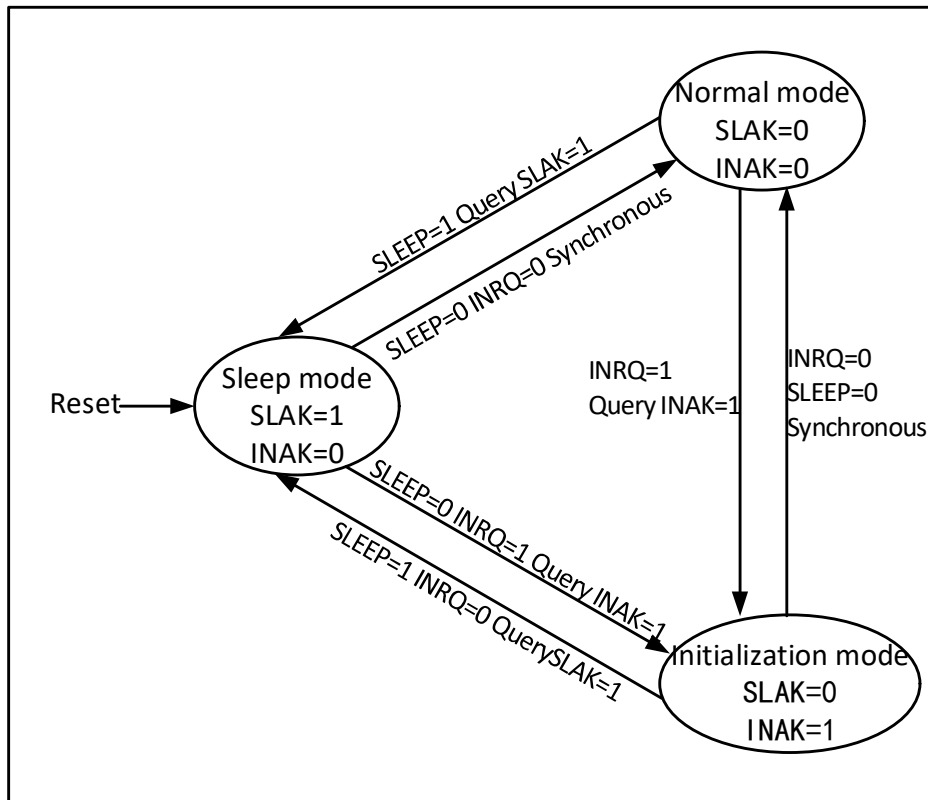
对过滤器组进行初始化，可以在非初始化模式下进行，不过必须对寄存器 CAN_FCTL 的 FINIT 位进行置 1，此时禁止接收报文。

22.2.2 睡眠模式

对寄存器 CAN_CTLR 中的 SLEEP 位置 1，请求 CAN 控制器进入睡眠模式，当寄存器 CAN_STATR 的 SNAK 位自动置 1 则 CAN 成功进入睡眠模式，此时 CAN 控制器的时钟停止，但邮箱寄存器仍可访问。由睡眠模式进入初始化模式，必须对 CAN_CTLR 的 SLEEP 位清 0，INRQ 位置 1，当寄存器 CAN_STATR 的 INAK 位自动置 1 则切换为初始化状态完成。

由睡眠模式进入正常模式，必须对 CAN_CTLR 的 SLEEP 位清 0，当寄存器 CAN_STATR 的 SNAK 位自动清 0 则进入正常模式。

图 22-1 CAN 工作模式切换



22.3 CAN 控制器测试模式

在初始化模式下，对寄存器 CAN_BTMR 的 SILM 和 LBKM 位进行操作，可以选择一种测试模式，然后通过对寄存器 CAN_CTLR 的 INRQ 位清零，退出初始化模式，进入测试模式。测试模式分为静默模式、环回模式和静默环回模式三种。

22.3.1 静默模式

对寄存器 CAN_BTMR 的 SILM 位置 1，可选择进入静默模式。该模式下，CAN 控制器可以接收，不能对外发送报文，对外总是处于隐性位，可以避免对总线产生影响，但是报文能够被所在节点的控制器的接收。通常静默模式被用于 CAN 总线的状态分析。

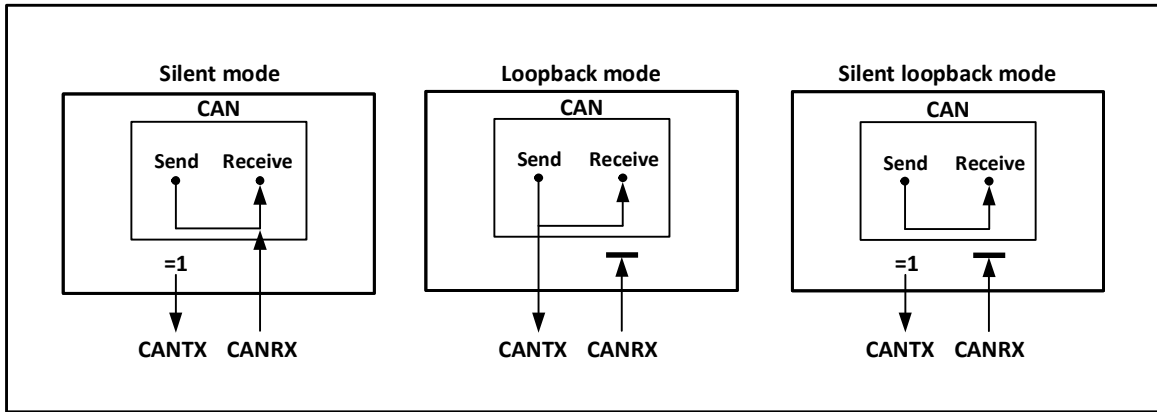
22.3.2 环回模式

对寄存器 CAN_BTMR 的 LBKM 位置 1，可选择进入环回模式。该模式下，CAN 控制器可以对外发送报文，不能接收外部报文，但是发送报文能够被所在节点的控制器的接收，接收过滤机制有效。通常环回模式被用于 CAN 控制器的收发测试。

22.3.3 静默环回模式

对寄存器 CAN_BTMR 的 SILM 和 LBKM 位置 1，可选择进入静默环回模式。该模式通常用于 CAN 控制器封闭自测试，在该模式下，对 CAN 总线无影响，RX 引脚与总线断开，TX 引脚置隐性位。

图 22-2 CAN 总线的三种测试模式



22.4 MCU 处于调试模式下 CAN 控制器的工作状态

当 MCU 进入调试模式后，内核处于暂停状态，但可以通过调试模块中配置位来决定 CAN 控制器是处于正常运行或停止状态。

22.5 CAN 控制器功能描述

22.5.1 发送处理流程

发送处理流程如下：如果三个发送邮箱中有空置的邮箱，应用层软件仅对空置邮箱的寄存器具有写入权限，对寄存器 CAN_TXMIRx、CAN_TXMDTRx、CAN_TXMDLRx 和 CAN_TXMDHRx 进行操作，可以设置报文标识符、报文长度、时间戳和报文数据等。在数据准备好之后，对寄存器 CAN_TXMIRx 的 TXRQ 位置 1 请求发送，邮箱进入挂号状态，并进行优先级排队；一旦成为最高优先级邮箱，则变为预定发送状态，等待 CAN 总线空闲；当 CAN 总线空闲时，预定发送邮箱的报文立刻进入发送状态；报文发送完毕后，邮箱重新成为空置邮箱，并且寄存器 CAN_TSTATR 的 RQCP 和 TXOK 位置 1，来指示发送成功；若发送时仲裁失败，寄存器 CAN_TSTATR 的 ALST 位置 1，若发送错误，则 TERR 位置 1。

22.5.2 发送优先级

发送优先级可以由标识符或发送请求先后次序决定，寄存器 CAN_CTLR 的 TXFP 位置 1 按发送请求先后次序发送，按发送请求先后次序主要应用于分段发送；TXFP 位清 0 按标识符优先级决定发送次序，标识符越小则优先级越高，同标识符的情况下，则低编号的邮箱有更高优先级。

22.5.3 发送中止处理

若对寄存器 CAN_TSTATR 的 ABRQ 位置 1，则可以中止发送请求。当邮箱状态为挂号或预定发送状态时，发送请求直接中止；当邮箱处于发送状态时，中止请求可能会成功（停止发送），也有可能失败（发送完成），结果可由寄存器 CAN_TSTATR 的 TXOK 位来查询。

22.5.4 基于时间触发模式

传统的 CAN 通信总线繁忙时，容易造成低优先级的消息长时间阻塞，甚至无法满足其时限的要求。为了解决该瓶颈，推出了基于时间触发模式的相关协议，此类协议在工业上有一定规模的应用，基于时间触发模式的功能即为配合此类协议的应用。

在时间触发模式下有两种模式可供选择，使用该模式需关闭自动重传功能，通过配置 CAN_TTCTL 寄存器的 MODE 位来选择默认模式和增强模式。对寄存器 CAN_CTLR 的 TTCM 和 NART 位置 1，使能时间触发模式并禁止自动重传，CAN_TTCTL 寄存器的 MODE 位默认为 0，此时工作在默认模式，内部定时器被激活用来产生发送和接收邮箱的时间戳，定时器在 CAN 位时间累加，内部定时器

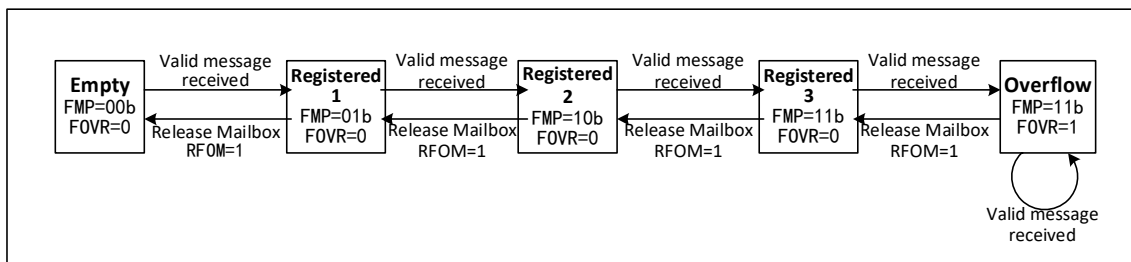
在接收和发送的帧起始位的采样点位置被采样并产生时间戳。若使用增强模式，需要将配置 CAN_TTCTLR 寄存器的 MODE 位为 1，来开启增强模式，使用该模式，在整个 CAN 网络中，必须存在三个或三个以上的节点，其中一个节点发送时间基准，其他节点收到该基准节点的时间戳后，通过向 CAN_TTCTLR 寄存器的 TIMRST 位写 1 复位内部计数器，将内部计数器进行同步，这样除了发送时间基准的节点之外，其余 CAN 节点实现了时间同步，之后将待发送的数据写入发送邮箱，依次配置各节点的时间触发计数值（CAN_TTCNT 寄存器的 TIMCNT）和内部计数器计数终值（CAN_TTCTLR 寄存器 TIMCMV），时间触发计数值和内部计数器计数终值由 CAN 节点的个数、CAN 通信速率和一帧数据位数决定，配置完成后，各节点等待内部计数器计数到时间触发计数值后，触发发送动作。

22.5.5 接收处理流程

CAN 总线报文的接收，由控制器硬件来完成，无需 MCU 的干涉，减轻了 MCU 的处理负荷。所接收到的报文，根据寄存器 CAN_FAFIFOR 的设置，分别被存储到两个具有 3 级邮箱深度的 FIFO 中，应用层如需获取报文，只能通过接收 FIFO 邮箱来读取有效接收报文。

初始时，接收 FIFO 为空，接收 FIFO 寄存器 CAN_RFIF0x 的 FMR[1:0] 值为二进制 00b，接到一个有效接收报文后，变为挂号 1 状态，控制器自动把接收 FIFO 寄存器 CAN_RFIF0x 的 FMR[1:0] 设置二进制 01b；若此时读取邮箱数据寄存器 CAN_RXMDLRx 和 CAN_RXMDHRx，通过对接收 FIFO 寄存器 CAN_RFIF0x 的 RFOM 位置 1 来释放邮箱，接收 FIFO 状态又变为空；如果在挂号 1 状态时不释放邮箱，下一个有效接收报文被接到后，接收 FIFO 状态切换为挂号 2 状态，此时接收 FIFO 寄存器 CAN_RFIF0x 的 FMR[1:0] 自动置二进制 10b；若读取邮箱数据寄存器并释放邮箱，则状态回到挂号 1；如果在挂号 2 状态不释放邮箱，则接收 FIFO 进入挂号 3 状态；同样在挂号 3 状态下读取报文并释放邮箱，则返回挂号 2 状态；若在挂号 3 状态不释放邮箱，则在接收到下一个有效报文时，必然导致报文丢失情况出现。

图 22-3 接收 FIFO 状态切换图



上文中的报文丢失情况，即接收 FIFO 为满，报文溢出导致报文丢失，接收 FIFO 寄存器 CAN_RFIF0x 的 FOVR 位会硬件自动置 1，以供溢出查询。寄存器 CAN_CTLR 的 RFLM 位置 1，则接收 FIFO 锁定功能启用，丢弃的报文为新接收报文；寄存器 CAN_CTLR 的 RFLM 位清 0，则接收 FIFO 锁定功能停用，接收 FIFO 的三个原报文中，最后接收的报文会被新报文覆盖。

当寄存器 CAN_INTENR 相关位置位，可以使接收 FIFO 状态切换时产生中断，以便更高效的处理接收报文，详见 22.6 节 CAN 中断。

22.5.6 接收报文标识符过滤

模块中有着多达 28 个过滤器组，通过设置过滤器组，每个 CAN 节点都可以接收到符合过滤规则的报文，不符合过滤规则的报文被硬件丢弃，无需软件干涉。

每个过滤器组由 2 个 32 位寄存器 CAN_FxR0 和 CAN_FxR1 组成。过滤器组的位宽都可以通过设置寄存器 CAN_FSCFGR 的各个位独立配置成 1 个 32 位过滤器或两个 16 位过滤器。每个过滤器组可通过设置寄存器 CAN_FMCFGR 的各个位配置为屏蔽位或标识符列表模式，各个过滤器组可以通过设置寄存器 CAN_FWR 的各个位选择启用或禁用。设置寄存器 CAN_FAFIFOR 的各个位可以把选择通过过滤器的报文存放到哪个接收 FIFO。

如下表 22-1 所示，屏蔽位模式下，两个寄存器分别为标识符寄存器和屏蔽寄存器，两者需要配

合使用，标识符寄存器每一位指示相应的位期望值为显性或隐性，屏蔽寄存器每一位指示相应位是否需要对应标识符寄存器位期望值一致。

表 22-1 32 位屏蔽位模式

标识符寄存器	CAN_FxR1[31:24]	CAN_FxR1[23:16]		CAN_FxR1[15:8]	CAN_FxR1[7:0]			
屏蔽位寄存器	CAN_FxR2[31:24]	CAN_FxR2[23:16]		CAN_FxR2[15:8]	CAN_FxR2[7:0]			
映射	STID[10:3]	STID[2:0]	EXID[17:13]	EXID[12:5]	EXID[4:0]	IDE	RTR	0

标识符列表模式下，两个寄存器都被用作标识符寄存器，接收报文标识符必须与其中一个寄存器保持一致才能通过筛选。

表 22-2 32 位标识符列表模式

标识符寄存器	CAN_FxR1[31:24]	CAN_FxR1[23:16]		CAN_FxR1[15:8]	CAN_FxR1[7:0]			
屏蔽位寄存器	CAN_FxR2[31:24]	CAN_FxR2[23:16]		CAN_FxR2[15:8]	CAN_FxR2[7:0]			
映射	STID[10:3]	STID[2:0]	EXID[17:13]	EXID[12:5]	EXID[4:0]	IDE	RTR	0

在 16 位模式下，寄存器组被拆分成四个寄存器，屏蔽位模式每组过滤器的屏蔽位模式可以有 2 个过滤器，每个过滤器里各包含一个 16 位标识符寄存器和 16 位屏蔽寄存器；标识符列表模式下四个寄存器都用作标识符寄存器。

表 22-3 16 位屏蔽位模式

标识符寄存器 n	CAN_FxR1[15:8]	CAN_FxR1[7:0]			
屏蔽位寄存器 n	CAN_FxR1[31:24]	CAN_FxR1[23:16]			
标识符寄存器 n+1	CAN_FxR2[15:8]	CAN_FxR2[7:0]			
屏蔽位寄存器 n+1	CAN_FxR2[31:24]	CAN_FxR2[23:16]			
映射	STID[10:3]	STID[2:0]	RTR	IDE	EXID[17:15]

表 22-4 16 位标识符列表模式

标识符寄存器 n	CAN_FxR1[15:8]	CAN_FxR1[7:0]			
屏蔽位寄存器 n	CAN_FxR1[31:24]	CAN_FxR1[23:16]			
标识符寄存器 n+1	CAN_FxR2[15:8]	CAN_FxR2[7:0]			
屏蔽位寄存器 n+1	CAN_FxR2[31:24]	CAN_FxR2[23:16]			
映射	STID[10:3]	STID[2:0]	RTR	IDE	EXID[17:15]

报文进入 FIFO 邮箱中，会被应用程序读取并存放，通常应用程序根据报文标识符来区分报文数据。CAN 控制器对接收 FIFO 中通过不同过滤器筛选的报文，提供了过滤器编号，编号被存放在寄存器 CAN_RXMDTRx 的 FMI[7:0] 中，编号时不考虑过滤器组是否启用。编号规则详见图 22-4 的示例。

当出现某个报文能通过多个过滤器的过滤，则接收邮箱中存放的过滤器编号根据过滤器优先级规则来决定存放哪个过滤器的编号，过滤器优先级规则如下：

- 所有 32 位的过滤器优先级均高于 16 位的过滤器
- 对于同样宽度的过滤器，标识符列表的过滤器优先级高于屏蔽位模式的过滤器
- 宽度和模式都一致的过滤器，编号小的过滤器优先级更高

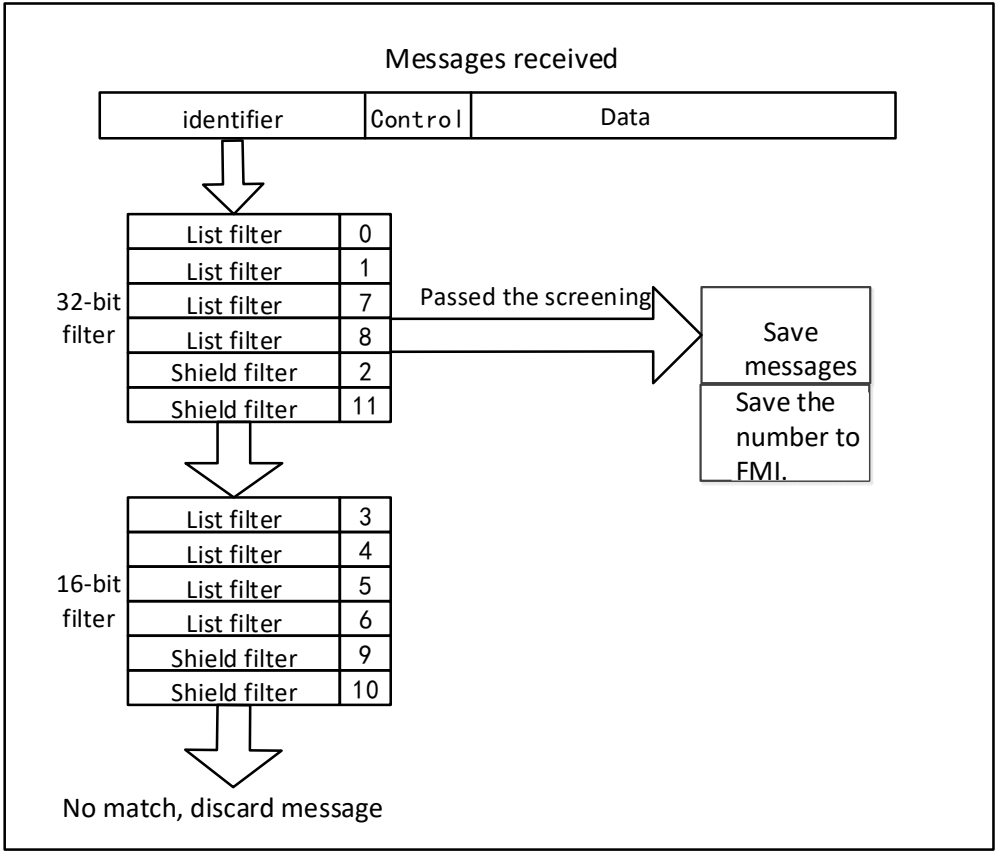
如图 22-5 所示：在接收报文时，先把标识符与 32 位标识符列表模式过滤器进行匹配筛选，没有匹配再与 32 位屏蔽位模式过滤器进行匹配筛选，没有匹配则继续与 16 位标识符列表模式过滤器

进行匹配筛选，没有匹配最后与 16 位屏蔽位模式过滤器进行匹配筛选，最后如果都没有匹配则丢弃报文，出现匹配则报文存入接收 FIFO 的邮箱，标识符编号存入寄存器 CAN_RXMDTRx 的 FMI 中。

图 22-4 过滤器编号的示例

Filter group number	FIFO0	Filter number	Filter group number	FIFO1	Filter number
0	32-bit shield mode	0	1	16-bit list mode	0 1 2 3
2	16-bit list mode	1 2 3 4	4	16-bit shield mode	4 5
3	32-bit list mode	5 6	6	32-bit list mode	6 7
5	Disabled 16-bit shield Mode	7 8	9	32-bit shield mode	8
7	32-bit shield mode	9	11	Disabled 16-bit shield Mode	9 10
8	32-bit list mode	10 11	12	32-bit list mode	11 12
10	16-bit shield mode	12 13	13	32-bit list mode	13 14

图 22-5 过滤器过滤示例



22. 5. 7 出错处理

CAN 控制器依靠状态错误寄存器 CAN_ERRSR, 对于总线上的出错管理。状态错误寄存器 CAN_ERRSR 里的 TEC 和 REC, 分别代表发送和接收错误计数值, 根据随着收发错误的增加而增加, 收发成功而

减小，可以根据它们的值来判断 CAN 总线的稳定性。

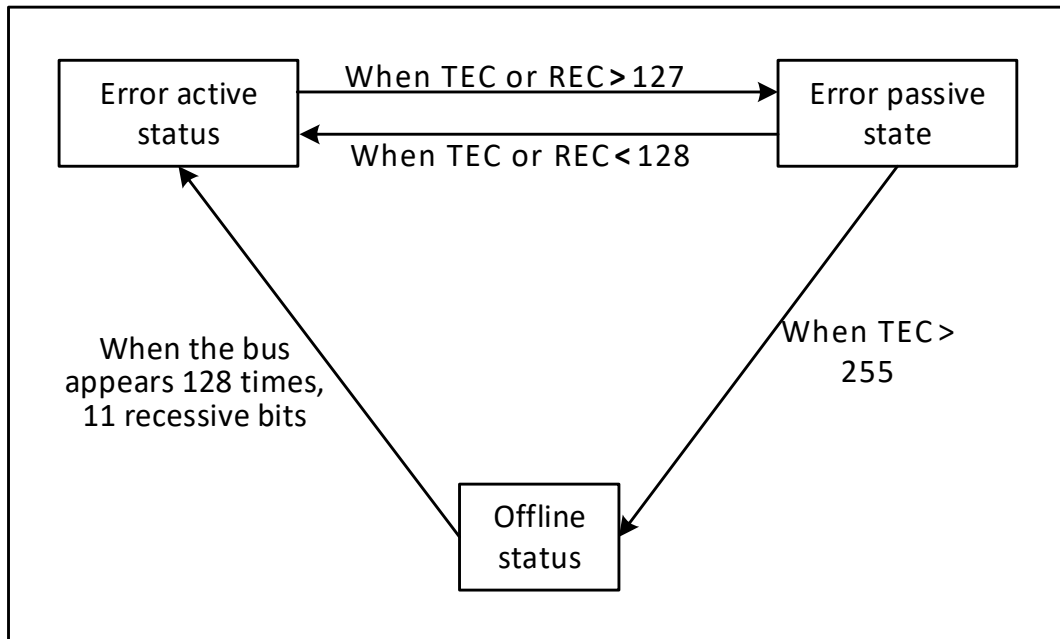
当状态错误寄存器 CAN_ERRSR 里的 TEC 和 REC 小于 128 时，当前 CAN 节点处于错误主动状态，可以正常参与总线通信，并且在检测到错误的时候发出主动错误标志。

当状态错误寄存器 CAN_ERRSR 里的 TEC 和 REC 大于 127 时，当前 CAN 节点处于错误被动状态，并且在检测到错误的时候不允许发出主动错误标志，只能发出被动错误标志。

当状态错误寄存器 CAN_ERRSR 里的 TEC 大于 255 时，当前 CAN 节点进入离线状态。

当总线监测到 128 次出现 11 个连续的隐性位时，恢复到错误主动状态，该恢复方式受主控制寄存器 CAN_CTLR 里的 ABOM 位影响。若 ABOM 置 1，则硬件自动退出离线状态。若 ABOM 为 0，则需要软件操作 INRQ 位进入初始化模式，随后退出初始化，才能退出离线状态。

图 22-6 CAN 错误状态切换图



22.5.8 位时序

按照 CAN 总线的标准，将每一位时间分为四段：分别为同步段、传播时间段、相位缓冲段 1 和相位缓冲段 2。这些段由最小时间单元 T_q 组成。CAN 控制器通过采样来监测 CAN 总线变化，通过帧起始位的边沿进行同步。

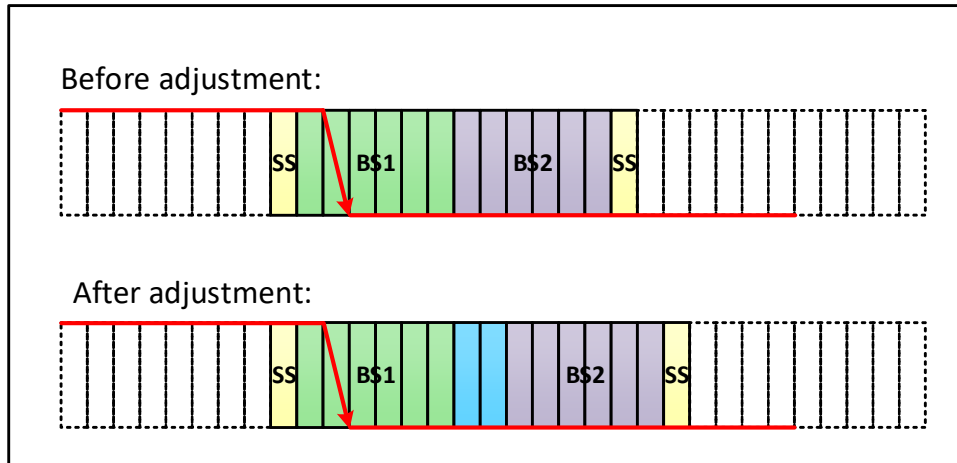
CAN 控制器把上述四段重新划分为三段，分别为：

- 同步段(SS)：也就是 CAN 标准里的同步段，固定为 1 个最小时间单元，正常情况下所期望的位跳变发生在本时间段内。
- 时间段 1(BS1)：包含 CAN 标准里的传播时间段和相位缓冲段 1，可以被设置为包含 1 到 16 最小时间单元，可以被自动延长，用于补偿 CAN 总线上不同节点频率精度误差带来的相位正向漂移。该时间段结束为采样点位置。
- 时间段 2(BS2)：也就是 CAN 标准里的相位缓冲段 2，可以被设置为 1 到 8 个最小时间单元，可以被自动缩短，以补偿 CAN 总线上不同节点频率精度误差带来的相位负向漂移。

重新同步跳转宽度(SJW)，是每位中可以延长和缩小的最小时间单元数量上限，范围可设置为 1 到 4 个最小时间单元。

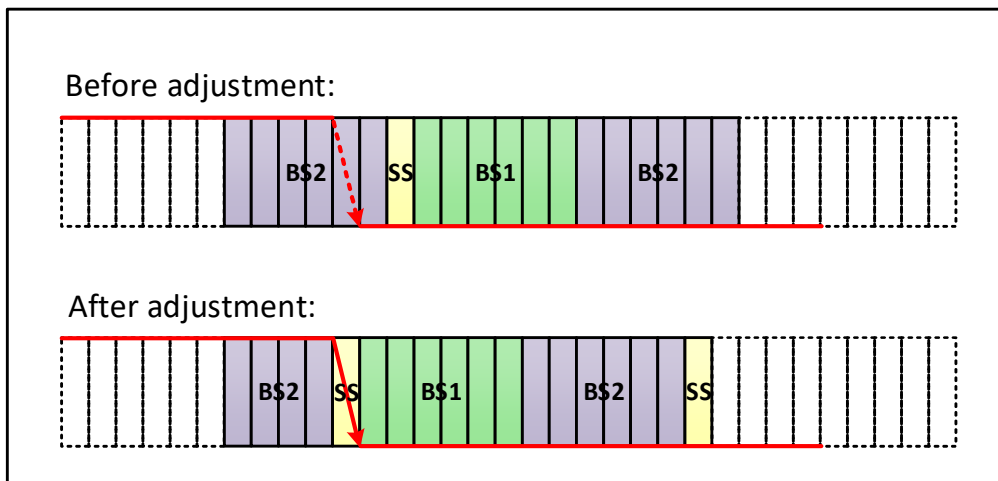
上述参数都可以在 CAN 总线时序寄存器 CAN_BTIMR 里配置。

图 22-7 跳变出现在 BS1 中



如图 22-7，SJW 为 2，总线电平跳变在时间段 1 被检测到，则需要延长时间段 1 的长度，最大延长 SJW，从而延迟采样点的位置。

图 22-8 跳变出现在 BS2 中



如图 22-8，SJW 为 2，总线电平跳变在时间段 2 被检测到，则需要缩小时间段 2 的长度，最大缩小 SJW，从而提前采样点的位置。

22.6 CAN 中断

CAN 控制器有四个中断向量，分别为发送中断、FIFO_0 中断、FIFO_1 中断、错误及状态变化中断。

设置 CAN 中断允许寄存器 CAN_INTENR，可以允许或禁用各个中断源。

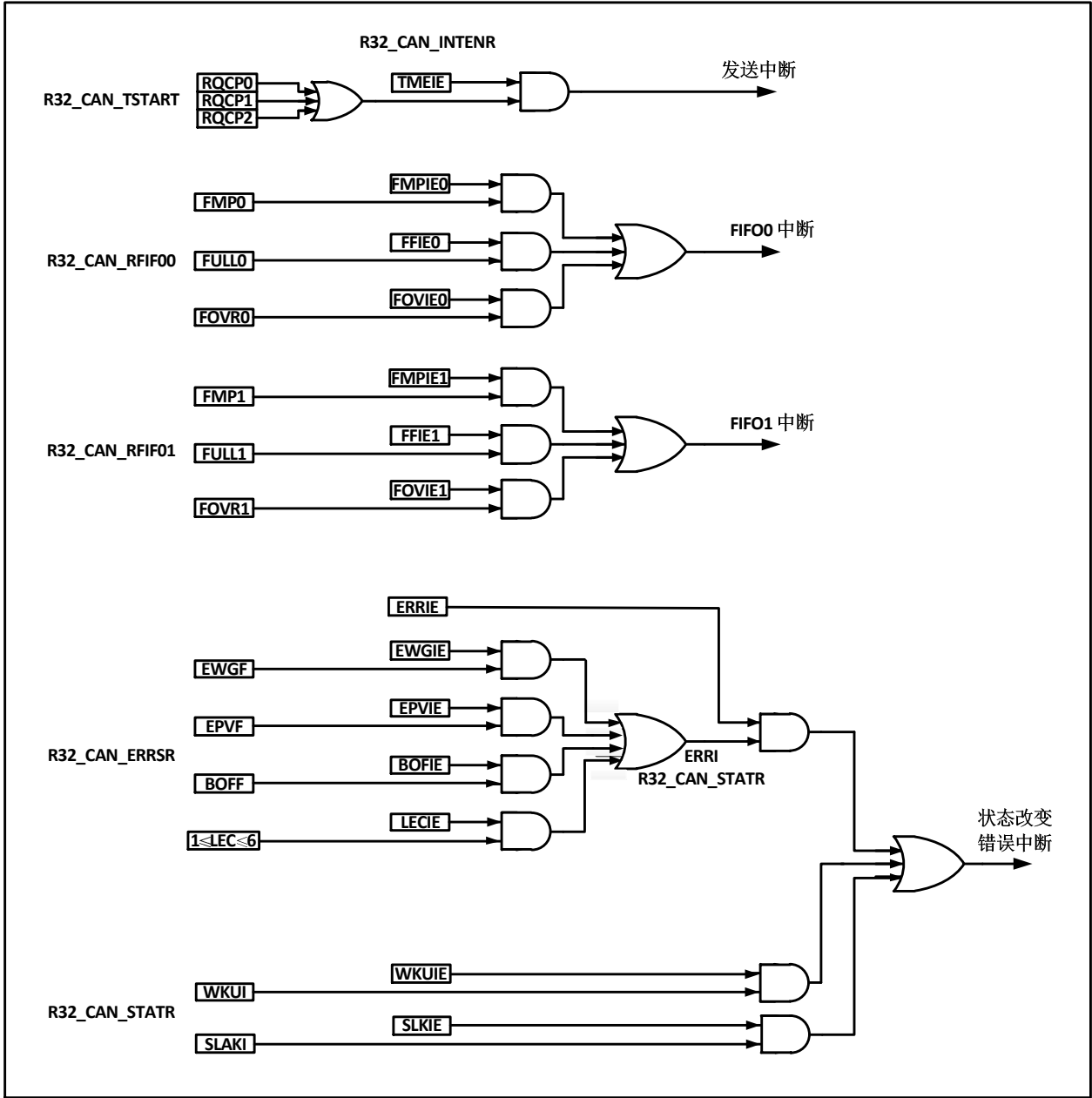
发送中断由发送邮箱变空事件产生，中断产生后，查询寄存器 CAN_TSTATR 的 RQCP0、RQCP1 和 RQCP2 位来判断是哪个邮箱变空事件产生。

FIFO0 中断由接收新报文、接收邮箱变满和溢出事件产生，中断产生后，查询寄存器 CAN_RFIF00 的 FMP0、FULL0 和 FOVER0 位来判断是哪个邮箱变空事件产生。

FIFO1 中断由接收新报文、接收邮箱变满和溢出事件产生，中断产生后，查询寄存器 CAN_RFIF01 的 FMP1、FULL1 和 FOVER1 位来判断是哪个邮箱变空事件产生。

错误及状态变化中断由出错、唤醒和睡眠事件产生。

图 22-9 CAN 中断逻辑图



22. 7 CAN FD 功能说明

22. 7. 1 FD 帧操作

1) FD 帧的发送:

只需将 CANFD_CR 寄存器的[7:0]位配置为 0x0F, 同时将对邮箱的 CANFD_DMA_T0/1/2 缓冲区填入发送数据, 并且配置 DMA 地址, 即可发出 FD 格式的帧, 其他寄存器按需配置即可。

2) FD 帧的接收:

只需配置对应接收 FIFO 的 CANFD_DMA_R0/1 地址, 并将 CANFD_BTR 寄存器配置为正确的比特率, 即可接收到 FD 帧。

在 FDCAN 格式中, DLC 的编码与标准 CAN 格式有所不同。其中 DLC 代码 0 至 8 的编码与标准 CAN 相同, 而代码 9 至 15 (在标准 CAN 中均编码一个 8 字节的数据字段) 的编码与标准 CAN 不同, 如表所示。

表 22-5 CAN FD 模式下的 DLC 编码

DLC	9	10	11	12	13	14	15
数据字节数	12	16	20	24	32	48	64

注：FD 帧接收 FIFO 深度为 1，也即每次接收数据到 DMA_R0/1 后，软件必须及时取走 RAM 中的数据包，否则数据会被覆盖。

22.7.2 发送器延时补偿

CAN FD 提供了一个发送器延时计时过滤功能，使用该功能时，需要将 CANFD_TDCT 寄存器的 TDC_FILTER 位进行设置，用来定义内部延时计数器的最小延时，防止 RX 上的显性位毛刺使得延时计数器过早结束。

可以通过 TDC0 位对发送器延时补偿偏移，大小为 FD_TDC0+1。CANFD_PSR 寄存器 TDCV[23:16] 位为只读状态，返回实际值；FD 发送延时补偿实际值为内部延时计数器的值加 FD_TDC0。

22.8 寄存器描述

CAN 控制器相关的寄存器必须用 32 位字的方式来操作。为了避免当前节点对整个 CAN 总线的影响，所以应用软件只能在初始化模式下修改位时序寄存器 CAN_BTMR。

表 22-6 CAN 相关寄存器列表

名称	访问地址	描述	复位值
R32_CAN_CTLR	0x40006400	CAN 主控制寄存器	0x00010002
R32_CAN_STATR	0x40006404	CAN 主状态寄存器	0x00000x02
R32_CAN_TSTATR	0x40006408	CAN 发送状态寄存器	0x1C000000
R32_CAN_RFIF00	0x4000640C	CAN 接收 FIFO0 控制和状态寄存器	0x00000000
R32_CAN_RFIF01	0x40006410	CAN 接收 FIFO1 控制和状态寄存器	0x00000000
R32_CAN_INTENR	0x40006414	CAN 中断使能寄存器	0x00000000
R32_CAN_ERRSR	0x40006418	CAN 错误状态寄存器	0x00000000
R32_CAN_BTMR	0x4000641C	CAN 位时序寄存器	0x01230000
R32_CAN_TTCTLR	0x40006420	CAN 时间触发控制寄存器	0x0000FFFF
R32_CAN_TTCNT	0x40006424	CAN 时间触发计数值寄存器	0x00000000
R32_CAN_TERR_CNT	0x40006428	CAN 离线恢复错误计数器	0x00000000
R32_CANFD_CR	0x4000642C	CANFD 控制寄存器	0x0000000E
R32_CANFD_BTR	0x40006430	CANFD 时序寄存器	0x60800637
R32_CANFD_TDCT	0x40006434	CANFD 发送延迟补偿寄存器	0x00000002
R32_CANFD_PSR	0x40006438	CANFD 发送延迟补偿值寄存器	0x00000000
R32_CANFD_DMA_T0	0x4000643C	CANFD DMA 发送邮箱 0 缓存寄存器	0x00000000
R32_CANFD_DMA_T1	0x40006440	CANFD DMA 发送邮箱 1 缓存寄存器	0x00000000
R32_CANFD_DMA_T2	0x40006444	CANFD DMA 发送邮箱 2 缓存寄存器	0x00000000
R32_CANFD_DMA_R0	0x40006448	CANFD DMA 接收邮箱 0 缓存寄存器	0x00000000
R32_CANFD_DMA_R1	0x4000644C	CANFD DMA 接收邮箱 1 缓存寄存器	0x00000000

表 22-7 CAN 邮箱相关寄存器列表

名称	访问地址	描述	复位值
R32_CAN_TXMIRO	0x40006580	CAN 发送邮箱 0 标识符寄存器	X
R32_CAN_TXMDTRO	0x40006584	CAN 发送邮箱 0 数据长度和时间戳寄存器	X

R32_CAN_TXMDLR0	0x40006588	CAN 发送邮箱 0 低字节数据寄存器	X
R32_CAN_TXMDHR0	0x4000658C	CAN 发送邮箱 0 高字节数据寄存器	X
R32_CAN_TXMIR1	0x40006590	CAN 发送邮箱 1 标识符寄存器	X
R32_CAN_TXMDTR1	0x40006594	CAN 发送邮箱 1 数据长度和时间戳寄存器	X
R32_CAN_TXMDLR1	0x40006598	CAN 发送邮箱 1 低字节数据寄存器	X
R32_CAN_TXMDHR1	0x4000659C	CAN 发送邮箱 1 高字节数据寄存器	X
R32_CAN_TXMIR2	0x400065A0	CAN 发送邮箱 2 标识符寄存器	X
R32_CAN_TXMDTR2	0x400065A4	CAN 发送邮箱 2 数据长度和时间戳寄存器	X
R32_CAN_TXMDLR2	0x400065A8	CAN 发送邮箱 2 低字节数据寄存器	X
R32_CAN_TXMDHR2	0x400065AC	CAN 发送邮箱 2 高字节数据寄存器	X
R32_CAN_RXMIR0	0x400065B0	CAN 接收 FIFO0 邮箱标识符寄存器	X
R32_CAN_RXMDTR0	0x400065B4	CAN 接收 FIFO0 邮箱数据长度和时间戳寄存器	X
R32_CAN_RXMDLR0	0x400065B8	CAN 接收 FIFO0 邮箱低字节数据寄存器	X
R32_CAN_RXMDHR0	0x400065BC	CAN 接收 FIFO0 邮箱高字节数据寄存器	X
R32_CAN_RXMIR1	0x400065C0	CAN 接收 FIFO1 邮箱标识符寄存器	X
R32_CAN_RXMDTR1	0x400065C4	CAN 接收 FIFO1 邮箱数据长度和时间戳寄存器	X
R32_CAN_RXMDLR1	0x400065C8	CAN 接收 FIFO1 邮箱低字节数据寄存器	X
R32_CAN_RXMDHR1	0x400065CC	CAN 接收 FIFO1 邮箱高字节数据寄存器	X

表 22-8 CAN 过滤器相关寄存器列表

名称	访问地址	描述	复位值
R32_CAN_FCTL	0x40006600	CAN 过滤器主控制寄存器	0x2A1C0E01
R32_CAN_FMCGR	0x40006604	CAN 过滤器模式寄存器	0x00000000
R32_CAN_FSCGR	0x4000660C	CAN 过滤器位宽寄存器	0x00000000
R32_CAN_FAFIFOR	0x40006614	CAN 过滤器 FIFO 关联寄存器	0x00000000
R32_CAN_FWR	0x4000661C	CAN 过滤器激活寄存器	0x00000000
R32_CAN_F0R1	0x40006640	CAN 过滤器组 0 寄存器 1	X
R32_CAN_F0R2	0x40006644	CAN 过滤器组 0 寄存器 2	X
R32_CAN_F1R1	0x40006648	CAN 过滤器组 1 寄存器 1	X
R32_CAN_F1R2	0x4000664C	CAN 过滤器组 1 寄存器 2	X
R32_CAN_F2R1	0x40006650	CAN 过滤器组 2 寄存器 1	X
R32_CAN_F2R2	0x40006654	CAN 过滤器组 2 寄存器 2	X
R32_CAN_F3R1	0x40006658	CAN 过滤器组 3 寄存器 1	X
R32_CAN_F3R2	0x4000665C	CAN 过滤器组 3 寄存器 2	X
R32_CAN_F4R1	0x40006660	CAN 过滤器组 4 寄存器 1	X
R32_CAN_F4R2	0x40006664	CAN 过滤器组 4 寄存器 2	X
R32_CAN_F5R1	0x40006668	CAN 过滤器组 5 寄存器 1	X
R32_CAN_F5R2	0x4000666C	CAN 过滤器组 5 寄存器 2	X
R32_CAN_F6R1	0x40006670	CAN 过滤器组 6 寄存器 1	X
R32_CAN_F6R2	0x40006674	CAN 过滤器组 6 寄存器 2	X
R32_CAN_F7R1	0x40006678	CAN 过滤器组 7 寄存器 1	X
R32_CAN_F7R2	0x4000667C	CAN 过滤器组 7 寄存器 2	X
R32_CAN_F8R1	0x40006680	CAN 过滤器组 8 寄存器 1	X
R32_CAN_F8R2	0x40006684	CAN 过滤器组 8 寄存器 2	X

R32_CAN_F9R1	0x40006688	CAN 过滤器组 9 寄存器 1	X
R32_CAN_F9R2	0x4000668C	CAN 过滤器组 9 寄存器 2	X
R32_CAN_F10R1	0x40006690	CAN 过滤器组 10 寄存器 1	X
R32_CAN_F10R2	0x40006694	CAN 过滤器组 10 寄存器 2	X
R32_CAN_F11R1	0x40006698	CAN 过滤器组 11 寄存器 1	X
R32_CAN_F11R2	0x4000669C	CAN 过滤器组 11 寄存器 2	X
R32_CAN_F12R1	0x400066A0	CAN 过滤器组 12 寄存器 1	X
R32_CAN_F12R2	0x400066A4	CAN 过滤器组 12 寄存器 2	X
R32_CAN_F13R1	0x400066A8	CAN 过滤器组 13 寄存器 1	X
R32_CAN_F13R2	0x400066AC	CAN 过滤器组 13 寄存器 2	X

22.8.1 CAN 主控制寄存器 (CAN_CTLR)

偏移地址: 0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
Reserved															DBF	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
RST	Reserved							TTCM	ABOM	AWUM	NART	RFLM	TXFP	SLEEP	INRQ	

位	名称	访问	描述	复位值
[31:17]	Reserved	RO	保留。	0
16	DBF	RW	调试是否禁止 CAN 总线工作 1: 调试时, CAN 的收发被禁止, 但是接收 FIFO 的控制和读写操作一切正常; 0: 调试时, CAN 控制器正常工作。	1
15	RST	RW1	CAN 控制器软件复位请求, 该位写 0 无效 1: 对 CAN 控制器进行复位, 复位后控制器进入睡眠模式, 然后硬件自动清 0; 0: CAN 控制器正常状态。	0
[14:8]	Reserved	RO	保留	0
7	TTCM	RW	是否允许时间触发模式 1: 使能时间触发模式; 0: 禁止时间触发模式。 时间触发模式主要是配合 TTCAN 协议使用。	0
6	ABOM	RW	离线自动退出控制 1: 硬件检测到 128 次连续 11 个隐性位, 自动退出离线状态; 0: 需要软件操作寄存器 CAN_CTLR 的 INRQ 位置 1 然后清 0, 当检测到 128 次连续 11 个隐性位后, 退出离线状态。	0
5	AWUM	RW	CAN 控制器自动唤醒使能 1: 当检测到报文时, 硬件自动唤醒, 寄存器 CAN_STATR 的 SLEEP 和 SLAK 位自动清 0; 0: 需要软件操作寄存器 CAN_CTLR 的 SLEEP 位清 0, 唤醒 CAN 控制器。	0

4	NART	RW	报文自动重传功能禁止 1: 无论发送成功与否, 报文只能被发送一次; 0: CAN 控制器一直重传至发送成功为止。	0
3	RFLM	RW	接收 FIFO 报文锁定模式使能 1: 当接收 FIFO 溢出时, 已接收邮箱报文未读出, 邮箱未释放时, 新接收到的报文被丢弃; 0: 当接收 FIFO 溢出时, 已接收邮箱报文未读出, 邮箱未释放时, 新接收到的报文会覆盖原有报文。 注: 该位只用于传统 CAN	0
2	TXFP	RW	发送邮箱优先级方式选择 1: 优先级由发送请求的先后顺序决定; 0: 优先级由报文标识符来决定。	0
1	SLEEP	RW	睡眠模式请求位 1: 置 1 请求 CAN 控制器进入睡眠模式, 当前活动完成后, 控制器进入睡眠模式, 若 AWUM 位置 1, 则在接收到报文时, 控制器把 SLEEP 位清 0; 0: 软件清 0 后, 控制器退出睡眠模式。	1
0	INRQ	RW	初始化模式请求位 1: 置 1 请求 CAN 控制器进入初始化模式, 当前活动完成后, 控制器进入初始化模式, 硬件对寄存器 CAN_STATR 的 INAK 位置 1; 0: 置 0 请求 CAN 控制器退出初始化模式, 进入正常模式, 硬件对寄存器 CAN_STATR 的 INAK 位清 0。	0

22.8.2 CAN 主状态寄存器 (CAN_STATR)

偏移地址: 0x04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				RX	SAMP	RXM	TXM	Reserved				SLAKI	WKUI	ERRI	SLAK INAK

位	名称	访问	描述	复位值
[31:12]	Reserved	RO	保留。	0
11	RX	RO	CAN 控制器接收引脚 RX 当前实际电平。	1
10	SAMP	RO	CAN 控制器接收引脚 RX 上一个接收位的电平	1
9	RXM	RO	接收模式查询位 1: 当前 CAN 控制器为接收模式; 0: 当前 CAN 控制器非接收模式。	0
8	TXM	RO	发送模式查询位 1: 当前 CAN 控制器为发送模式; 0: 当前 CAN 控制器非发送模式。	0

[7:5]	Reserved	RO	保留	0
4	SLAKI	RW1	睡眠中断使能时，即寄存器 CAN_INTENR 的 SLKIE 位置 1 时，中断产生标志位，写 1 清 0，写 0 无效。 1：进入睡眠模式时，中断产生，硬件置 1； 0：退出睡眠模式时，硬件清 0 也可软件清 0。	0
3	WKUI	RW1	唤醒中断标志位。当寄存器 CAN_INTENR 的 WKUI 位置 1 时，若 CAN 控制器处于睡眠模式时，检测到 SOF 位，则硬件置 1。软件置 1 清 0，置 0 无效。	0
2	ERRI	RW1	出错中断状态标志位。当寄存器 CAN_INTENR 的 ERRIE 位置 1 时，产生错误及状态变化中断。该位软件置 1 清 0，置 0 无效。	0
1	SLAK	RO	睡眠模式指示位。 1：CAN 控制器正处于睡眠模式； 0：CAN 控制器不在睡眠模式。	1
0	INAK	RO	初始化模式指示位。 1：CAN 控制器正在初始化模式； 0：CAN 控制器工作在非初始化模式。	0

22.8.3 CAN 发送状态寄存器 (CAN_TSTATR)

偏移地址：0x08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOW2	LOW1	LOW0	TME2	TME1	TME0	CODE[1:0]	ABRQ2	Reserved			TERR2	ALST2	TXOK2	RQCP2	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ABRQ1	Reserved		TERR1	ALST1	TXOK1	RQCP1	ABRQ0	Reserved			TERR0	ALST0	TXOK0	RQCP0	

位	名称	访问	描述	复位值
31	LOW2	RO	表示发送邮箱 2 的最低优先级标志位 1：表示发送邮箱 2 的优先级最低； 0：表示发送邮箱 2 的优先级非最低。	0
30	LOW1	RO	发送邮箱 1 的最低优先级标志位 1：表示发送邮箱 1 的优先级最低； 0：表示发送邮箱 1 的优先级非最低。	0
29	LOW0	RO	发送邮箱 0 的最低优先级标志位 1：表示发送邮箱 0 的优先级最低； 0：表示发送邮箱 0 的优先级非最低。	0
28	TME2	RO	表示发送邮箱 2 的空标志位 1：表示发送邮箱 2 无等待发送报文； 0：表示发送邮箱 2 有等待发送报文。	1
27	TME1	RO	表示发送邮箱 1 的空标志位 1：表示发送邮箱 1 无等待发送报文； 0：表示发送邮箱 1 有等待发送报文。	1
26	TME0	RO	表示发送邮箱 0 的空标志位	1

			1: 表示发送邮箱 0 无等待发送报文; 0: 表示发送邮箱 0 有等待发送报文。	
[25:24]	CODE[1:0]	R0	邮箱编号 当有 1 个以上邮箱为空时, 表示下一个为空的邮箱号; 当邮箱全空时, 表示优先级最低的邮箱号。	0
23	ABRQ2	RW1	发送邮箱 2 的发送中止请求。软件置 1, 可以中止邮箱 2 的发送请求, 发送报文被清除时硬件清 0, 若邮箱 2 清空, 软件置 1 无效。	0
[22:20]	Reserved	R0	保留	0
19	TERR2	RW1	发送邮箱 2 发送失败标志位, 当发送邮箱 2 发送失败, 该位自动置 1。软件置 1 清 0, 软件写 0 无效。	0
18	ALST2	RW1	发送邮箱 2 仲裁失败标志位, 当发送邮箱 2 仲裁优先级低导致发送失败, 该位自动置 1。软件置 1 清 0, 软件写 0 无效。	0
17	TXOK2	RW1	发送邮箱 2 发送成功标志位 1: 上次发送成功; 0: 上次发送失败。 软件置 1 清 0, 软件写 0 无效。	0
16	RQCP2	RW1	发送邮箱 2 请求完成标志位, 当发送邮箱 2 的发送或中止请求完成时, 该位自动置 1。软件置 1 清 0, 软件写 0 无效。	0
15	ABRQ1	RW1	发送邮箱 1 的发送中止请求。软件置 1, 可以中止邮箱 1 的发送请求, 发送报文被清除时硬件清 0。软件写 0 无效。	0
[14:12]	Reserved	R0	保留	0
11	TERR1	RW1	发送邮箱 1 发送失败标志位, 当发送邮箱 1 发送失败, 该位自动置 1。软件置 1 清 0, 软件写 0 无效。	0
10	ALST1	RW1	发送邮箱 1 仲裁失败标志位, 当发送邮箱 1 仲裁优先级低导致发送失败, 该位自动置 1。	0
9	TXOK1	RW1	发送邮箱 1 发送成功标志位 1: 上次发送成功; 0: 上次发送失败。 软件置 1 清 0, 软件写 0 无效。	0
8	RQCP1	RW1	发送邮箱 1 请求完成标志位, 当发送邮箱 1 的发送或中止请求完成时, 该位自动置 1。软件置 1 清 0, 软件写 0 无效。	0
7	ABRQ0	RW1	发送邮箱 0 的发送中止请求。软件置 1, 可以中止邮箱 0 的发送请求, 发送报文被清除时硬件清 0。软件写 0 无效。	0
[6:4]	Reserved	R0	保留	0
3	TERR0	RW1	发送邮箱 0 发送失败标志位, 当发送邮箱 0 发送失败, 该位自动置 1。软件置 1 清 0, 软	0

			件写 0 无效。	
2	ALST0	RW1	发送邮箱 0 仲裁失败标志位，当发送邮箱 0 仲裁优先级低导致发送失败，该位自动置 1。软件置 1 清 0，软件写 0 无效。	0
1	TXOK0	RW1	发送邮箱 0 发送成功标志位 1：上次发送成功； 0：上次发送失败。 软件置 1 清 0，软件写 0 无效。	0
0	RQCP0	RW1	发送邮箱 0 请求完成标志位，当发送邮箱 0 的发送或中止请求完成时，该位自动置 1。软件置 1 清 0，软件写 0 无效。	0

22.8.4 CAN 接收 FIFO 0 状态寄存器 (CAN_RFIF00)

偏移地址：0x0C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										RFOM0	FOVR0	FULL0	Reserved	FMP0[1:0]	

位	名称	访问	描述	复位值
[31:6]	Reserved	R0	保留。	0
5	RFOM0	RW1	软件对该位置 1，则释放接收 FIFO_0 的当前邮箱报文，释放完后自动清 0，软件写 0 无效。	0
4	FOVR0	RW1	接收 FIFO_0 溢出标志位。当 FIFO_0 中有三个报文时，又接到新报文，硬件置 1。该位需要软件置 1 清 0，软件写 0 无效。	0
3	FULL0	RW1	接收 FIFO_0 满标志位。当 FIFO_0 中有三个报文时，硬件置 1。该位需要软件置 1 清 0，软件写 0 无效。	0
2	Reserved	R0	保留。	0
[1:0]	FMP0[1:0]	R0	接收 FIFO_0 报文数目。	0

22.8.5 CAN 接收 FIFO 1 状态寄存器 (CAN_RFIF01)

偏移地址：0x10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										RFOM1	FOVR1	FULL1	Reserved	FMP1[1:0]	

位	名称	访问	描述	复位值
---	----	----	----	-----

[31:6]	Reserved	RO	保留。	0
5	RFOM1	RW1	软件对该位置 1, 则释放接收 FIFO_1 的当前邮箱报文, 释放完后自动清 0, 软件写 0 无效。	0
4	FOVR1	RW1	接收 FIFO_1 溢出标志位。当 FIFO_1 中有三个报文时, 又接到新报文, 硬件置 1。该位需要软件置 1 清 0, 软件写 0 无效。	0
3	FULL1	RW1	接收 FIFO_1 满标志位。当 FIFO_1 中有三个报文时, 硬件置 1。该位需要软件置 1 清 0, 软件写 0 无效。	0
2	Reserved	RF	保留	0
[1:0]	FMP1[1:0]	RO	接收 FIFO_1 报文数目。	0

22.8.6 CAN 中断使能寄存器 (CAN_INTENR)

偏移地址: 0x14

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved														SLKIE	WKUIE
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ERRIE	Reserved			LECIE	BOFIE	EPVIE	EWGIE	Reserved	FOVIE1	FFIE1	FMPIE1	FOVIE0	FFIE0	FMPIE0	TMEIE

位	名称	访问	描述	复位值
[31:18]	Reserved	RO	保留。	0
17	SLKIE	RW	睡眠中断使能位。 1: 进入睡眠状态时, 产生中断; 0: 进入睡眠状态时, 不产生中断。	0
16	WKUIE	RW	唤醒中断使能位。 1: 当 CAN 控制器被唤醒时, 产生中断; 0: 当 CAN 控制器被唤醒时, 不产生中断。	0
15	ERRIE	RW	错误中断使能位, CAN 错误中断总使能位。 1: 当 CAN 控制器产生错误时, 产生中断; 0: 当 CAN 控制器产生错误时, 不产生中断。	0
[14:12]	Reserved	RF	保留。	0
11	LECIE	RW	上次错误号中断使能位。 1: 检测到错误时, 硬件更新 LEC[2:0], 更新 ERRI 位为 1, 触发错误中断; 0: 检测到错误时, 硬件更新 LEC[2:0], 不更新 ERRI 位, 不触发错误中断。	0
10	BOFIE	RW	离线中断使能位。 1: 进入离线状态时, 更新 ERRI 位为 1, 触发错误中断; 0: 进入离线状态时, 不更新 ERRI 位, 不触发错误中断。	0
9	EPVIE	RW	错误被动中断使能位。	0

			1: 进入错误被动状态时, 更新 ERR1 位为 1, 触发错误中断; 0: 进入错误被动状态时, 不更新 ERR1 位, 不触发错误中断。	
8	EWGIE	RW	错误警告中断使能位。 1: 出错次数达到警告阈值时, 更新 ERR1 位为 1, 触发错误中断; 0: 出错次数达到警告阈值时, 不更新 ERR1 位, 不触发错误中断。	0
7	Reserved	RF	保留。	0
6	FOVIE1	RW	接收 FIFO_1 溢出中断使能位。 1: 当 FIFO_1 溢出, 触发 FIFO_1 中断; 0: 当 FIFO_1 溢出, 不触发 FIFO_1 中断。	0
5	FFIE1	RW	接收 FIFO_1 满中断使能位。 1: 当 FIFO_1 为满, 触发 FIFO_1 中断; 0: 当 FIFO_1 为满, 不触发 FIFO_1 中断。	0
4	FMPIE1	RW	接收 FIFO_1 消息挂号中断使能位。 1: 当 FIFO_1 更新 FMP 位, 且不为 0, 触发 FIFO_1 中断; 0: 当 FIFO_1 更新 FMP 位, 且不为 0, 不触发 FIFO_1 中断。	0
3	FOVIE0	RW	接收 FIFO_0 溢出中断使能位。 1: 当 FIFO_0 溢出, 触发 FIFO_0 中断; 0: 当 FIFO_0 溢出, 不触发 FIFO_0 中断。	0
2	FFIE0	RW	接收 FIFO_0 满中断使能位。 1: 当 FIFO_0 为满, 触发 FIFO_0 中断; 0: 当 FIFO_0 为满, 不触发 FIFO_0 中断。	0
1	FMPIE0	RW	接收 FIFO_0 消息挂号中断使能位。 1: 当 FIFO_0 更新 FMP 位, 且不为 0, 触发 FIFO_0 中断; 0: 当 FIFO_0 更新 FMP 位, 且不为 0, 不触发 FIFO_0 中断。	0
0	TMEIE	RW	发送邮箱空中断。 1: 当发送邮箱为空时, 产生中断; 0: 当发送邮箱为空时, 不产生中断。	0

22.8.7 CAN 错误状态寄存器 (CAN_ERRSR)

偏移地址: 0x18

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

位	名称	访问	描述	复位值
[31:24]	REC[7:0]	RO	接收错误计数器。 当 CAN 接收出错时, 根据出错条件, 该计数器加 1 或 8; 接收成功后, 该计数器减 1 或设为 120 (错误计数值大于 127)。计数器值超过 127 时, CAN 进入错误被动状态。	0
[23:16]	TEC[7:0]	RO	发送错误计数器。 当 CAN 发送出错时, 根据出错条件, 该计数器加 1 或 8; 发送成功后, 该计数器减 1 或设为 120 (错误计数值大于 127)。计数器值超过 127 时, CAN 进入错误被动状态。	0
[15:7]	Reserved	RO	保留。	0
[6:4]	LEC[2:0]	RW	上次错误代号。 检测到 CAN 总线上发送错误时, 控制器根据出错情况设置, 当正确收发报文时, 置 000b。 000: 无错误; 001: 位填充错误; 010: FORM 格式错误; 011: ACK 确认错误; 100: 隐性位错误; 101: 显性位错误; 110: CRC 错误; 111: 软件设置。 通常应用软件读取到错误时, 把代号设置为 111b, 可以检测到代号更新。	0
3	Reserved	RO	保留。	0
2	BOFF	RO	离线状态标志位。 当 CAN 控制器进入离线状态时, 硬件自动置 1; 退出离线状态时, 硬件自动清 0。	0
1	EPVF	RO	错误被动标志位。 当收发错误计数器达到错误被动阈值时, 即大于 127 时, 硬件置 1。	0
0	EWGF	RO	错误警告标志位。 当收发错误计数器达到警告阈值时, 即大于等于 96 时, 硬件置 1。	0

22.8.8 CAN 位时序寄存器 (CAN_BTMR)

偏移地址: 0x1C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SILM	LBKM	Reserved	SJW[3:0]				TS2[3:0]				TS1[3:0]				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BTR_TS1_T				Reserved		BRP[9:0]									

位	名称	访问	描述	复位值
---	----	----	----	-----

31	SILM	RW	静默模式设置位。 1: 进入静默模式; 0: 退出静默模式。	0
30	LBKM	RW	环回模式设置位。 1: 进入环回模式; 0: 退出环回模式。	0
[29:28]	Reserved	RO	保留。	0
[27:24]	SJW[3:0]	RW	定义了重新同步跳转宽度设置值。 实现重新同步时, 位中可以延长和缩小的最小时间单元数量上限, 实际值为 (SJW[1:0]+1), 范围可设置为 1 到 4 个最小时间单元。	0001b
[23:20]	TS2[3:0]	RW	时间段 2 设置值。 定义了时间段 2 占用了多少个最小时间单元, 实际值为 (TS2[1:0]+1)。	0010b
[19:16]	TS1[3:0]	RW	时间段 1 设置值。 定义了时间段 1 占用了多少个最小时间单元, 实际值为 (TS1[1:0]+1)。	0011b
[15:12]	BTR_TS1_T	RW	对于传统 CAN, CLAS_LONG_TS1=0, 则 TS1 为 TS[3:0] (4bit); CLAS_LONG_TS1=1, 则 TS1 为 TS[1:0]+BTR_TS1_T[15:12] (6bit)。	0
[11:10]	Reserved	RO	保留。	0
[9:0]	BRP[9:0]	RW	最小时间单元长度设置值 $T_q = (BRP[9:0]+1) \times t_{pclkS}$	0

22.8.9 CAN 时间触发控制寄存器 (CAN_TTCTLR)

偏移地址: 0x20

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

位	名称	访问	描述	复位值
[31:18]	Reserved	RO	保留。	0
17	MODE	RW	时间触发模式选择位。 1: 增强模式; 0: 默认模式。	0
16	TIMRST	WZ	内部计数器复位控制位。 写 1 复位内部计数器, 硬件自动清 0	0
[15:0]	TIMCMV[15:0]	RW	内部计数器计数终值	ffffh

22.8.10 CAN 时间触发计数值寄存器 (CAN_TTCNT)

偏移地址: 0x24

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

位	名称	访问	描述	复位值
[31:16]	Reserved	RO	保留。	0
[15:0]	TIMCNT[15:0]	RW	时间触发计数值	0

22.8.11 CAN 离线恢复错误计数器 (CAN_TERR_CNT)

偏移地址: 0x28

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

位	名称	访问	描述	复位值
[31:9]	Reserved	RO	保留。	0
[8:0]	TX_ERR_CNT	RW	当前离线恢复错误计数值, 修改该计数值可从离线立即恢复	0

22.8.12 CANFD 控制寄存器 (CANFD_CR)

偏移地址: 0x2C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						RESTR ICT_M ODE	CLAS _LONG _TS1	RES_E XCEPT	USER_ESI_B			TX_BRS_B		TX_FD	

位	名称	访问	描述	复位值
[31:10]	Reserved	RO	保留。	0
9	RESTRICT_MODE	RW	CANFD 的受限操作模式使能 1: 错误计数器不会增加或减少, 并且只能发送隐性的错误帧和过载帧, 接收不受影响。 0: 无效	0
8	CLAS_LONG_TS1	RW	选择 CAN 的 TS1 阶段长度 1: 6bit 0: 4bit	0
7	RES_EXCEPT	RW	FD 帧 RES 位协议例外使能: 1: 收到隐性的 RES 位不会产生错误; 0: 收到隐性的 RES 位产生形式错误。	0

[6:4]	USER_ESI_B	RW	发送 FD 帧时，三个发送邮箱的 ESI 位： 1：软件强制发送隐性的 ESI 位； 0：硬件自动配置，错误主动时发送显性的 ESI，错误被动时发送隐性的 ESI；	0
[3:1]	TX_BRS_B	RW	发送 FD 帧时，三个发送邮箱的 BRS 位： 1：对应通道使能位速率切换； 0：对应通道不使能位速率切换。	111b
0	TX_FD	RW	发送 FD 帧使能位： 1：发送 FD 帧； 0：发送传统帧。	0

22.8.13 CANFD 时序寄存器 (CANFD_BTR)

偏移地址：0x30

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved								TDCE	Reserved			BTR_BRP_FD			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			BTR_TS1_FD					BTR_TS2_FD				BTR_SJW_FD			

位	名称	访问	描述	复位值
[31:24]	Reserved	RO	保留。	0
23	TDCE	RW	FD 帧发送延时补偿使能	1
[22:21]	Reserved	RO	保留。	0
[20:16]	BTR_BRP_FD	RW	最小时间单元长度设置值 $T_q = (BRP+1) \times t_{pclk}$ 注：CAN 波特率计算公式为： $CANbps = PCLK1 / ((TS1+1+TS2+1+1) * (BPR+1))$	0
[15:13]	Reserved	RO	保留。	0
[12:8]	BTR_TS1_FD	RW	时间段 1 设置值： 定义了时间段 1 占用了多少个最小时间单元，实际值为 TS1+1	00110b
[7:4]	BTR_TS2_FD	RW	时间段 2 设置值： 定义了时间段 2 占用了多少个最小时间单元，实际值为 TS2+1	0011b
[3:0]	BTR_SJW_FD	RW	重新同步跳跃宽度设置值： 实现重新同步时，位中可以延长和缩小的最小时间单元数量上限，实际值为 FD_SJW+1，范围可设置为 1 到 4 个最小时间单元。	0111b

22.8.14 CANFD 发送延迟补偿寄存器 (CANFD_TDCT)

偏移地址：0x34

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

Reserved	TDC_FILTER	Reserved	TDC0
----------	------------	----------	------

位	名称	访问	描述	复位值
[31:14]	Reserved	RO	保留。	0
[13:8]	TDC_FILTER	RW	发送器延时计时过滤： 定义内部延时计数器的最小延时，防止 RX 上的显性位毛刺使得延时计数器过早结束。	0
[7:6]	Reserved	RO	保留。	0
[5:0]	TDC0	RW	发送器延时补偿偏移： FD_TDC0+1	000010b

22.8.15 CANFD 发送延迟补偿值寄存器 (CANFD_PSR)

偏移地址：0x38

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

位	名称	访问	描述	复位值
[31:24]	Reserved	RO	保留。	0
[23:16]	TDCV	RO	FD 发送延时补偿实际值：内部延时计数器的值加 FD_TDC0	0
[15:0]	Reserved	RO	保留	0

22.8.16 CANFD DMA 发送邮箱 0 地址寄存器 (CANFD_DMA_T0)

偏移地址：0x3C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	DMA_ADDR_T0														

位	名称	访问	描述	复位值
[31:15]	Reserved	RO	保留。	0
[14:0]	DMA_ADDR_T0	RW	FD 帧发送邮箱 0 对应的发送缓冲区，地址必须 4 字节对齐。	0

22.8.17 CANFD DMA 发送邮箱 1 地址寄存器 (CANFD_DMA_T1)

偏移地址：0x40

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

Reserved	DMA_ADDR_T1
----------	-------------

位	名称	访问	描述	复位值
[31:15]	Reserved	RO	保留。	0
[14:0]	DMA_ADDR_T1	RW	FD 帧发送邮箱 1 对应的发送缓冲区，地址必须 4 字节对齐。	0

22.8.18 CANFD DMA 发送邮箱 2 地址寄存器 (CANFD_DMA_T2)

偏移地址：0x44

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	DMA_ADDR_T2														

位	名称	访问	描述	复位值
[31:15]	Reserved	RO	保留。	0
[14:0]	DMA_ADDR_T2	RW	FD 帧发送邮箱 2 对应的发送缓冲区，地址必须 4 字节对齐。	0

22.8.19 CANFD DMA 接收邮箱 0 地址寄存器 (CANFD_DMA_R0)

偏移地址：0x48

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	DMA_ADDR_R0														

位	名称	访问	描述	复位值
[31:15]	Reserved	RO	保留。	0
[14:0]	DMA_ADDR_R0	RW	FD 帧接收 FIFO0 对应的接收缓冲区，地址必须 4 字节对齐。	0

22.8.20 CANFD DMA 接收邮箱 1 地址寄存器 (CANFD_DMA_R1)

偏移地址：0x4C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	DMA_ADDR_R1														

位	名称	访问	描述	复位值
[31:15]	Reserved	RO	保留。	0
[14:0]	DMA_ADDR_R1	RW	FD 帧接收 FIFO1 对应的接收缓冲区，地址必须 4 字节对齐。	0

22.8.21 CAN 发送邮箱标识符寄存器 (CAN_TXMIRy) (y=0/1/2)

偏移地址：0x180, 0x190, 0x1A0

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

位	名称	访问	描述	复位值
[31:21]	STID[10:0] /EXID[28:18]	RW	标准标识符或扩展标识符的高 11 位。	x
[20:3]	EXID[17:0]	RW	扩展标识符的低 18 位。	x
2	IDE	RW	标识符选择标志位。 1: 选用扩展标识符; 0: 选用标准标识符。	x
1	RTR	RW	远程帧(也称遥控帧)选择标志位。 1: 当前为远程帧; 0: 当前为数据帧。	x
0	TXRQ	RW	数据发送请求标志位。 软件置 1 时, 请求发送邮箱里的数据, 发送完毕邮箱为空时, 硬件清 0。	0

22.8.22 CAN 发送邮箱数据长度和时间戳寄存器 (CAN_TXMDTRY) (y=0/1/2)

偏移地址：0x184, 0x194, 0x1A4

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

位	名称	访问	描述	复位值
[31:16]	TIME[15:0]	RW	用于发送报文 SOF 时刻的 16 位定时器值。	x
[15:9]	Reserved	RO	保留。	0
8	TGT	RW	报文时间戳发送选择标志位。该位在 TTCM 置 1, 并报文长度为 8 时有效。 1: 发送时间戳, 值为 TIME[15:0] 的即时值, 替换了 8 字节报文的最后两个字节; 0: 不发送时间戳。	x
[7:4]	Reserved	RO	保留。	0

[3:0]	DLC[3:0]	RW	数据帧的数据长度或远程帧请求数据长度 数据长度可设置范围为 0 到 64。	0
-------	----------	----	--	---

22.8.23 CAN 发送邮箱低字节数据寄存器 (CAN_TXMDLRy) (y=0/1/2)

偏移地址: 0x188, 0x198, 0x1A8

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA3[7:0]								DATA2[7:0]							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA1[7:0]								DATA0[7:0]							

位	名称	访问	描述	复位值
[31:24]	DATA3[7:0]	RW	发送数据字节 3 的内容。	x
[23:16]	DATA2[7:0]	RW	发送数据字节 2 的内容。	x
[15:8]	DATA1[7:0]	RW	发送数据字节 1 的内容。	x
[7:0]	DATA0[7:0]	RW	发送数据字节 0 的内容。	x

22.8.24 CAN 发送邮箱高字节数据寄存器 (CAN_TXMDHRy) (y=0/1/2)

偏移地址: 0x18C, 0x19C, 0x1AC

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA7[7:0]								DATA6[7:0]							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA5[7:0]								DATA4[7:0]							

位	名称	访问	描述	复位值
[31:24]	DATA7[7:0]	RW	发送数据字节 7 的内容。	x
[23:16]	DATA6[7:0]	RW	发送数据字节 6 的内容。	x
[15:8]	DATA5[7:0]	RW	发送数据字节 5 的内容。	x
[7:0]	DATA4[7:0]	RW	发送数据字节 4 的内容。	x

22.8.25 CAN 接收邮箱标识符寄存器 (CAN_RXMIRy) (y=0/1)

偏移地址: 0x1B0, 0x1C0

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

位	名称	访问	描述	复位值
[31:21]	STID[10:0] /EXIDH[28:18]	RO	标准标识符或扩展标识符的高 11 位。	x
[20:3]	EXIDL[17:0]	RO	扩展标识符的低 18 位。	x

2	IDE	RO	标识符选择标志位。 1: 选用扩展标识符; 0: 选用标准标识符。	x
1	RTR	RO	远程帧(也称遥控帧)选择标志位。 1: 当前为远程帧; 0: 当前为数据帧。	x
0	FDF	RO	当前接收帧的 FD 帧指示标志 1: FD 帧 0: 传统帧	0

22.8.26 CAN 接收邮箱数据长度和时间戳寄存器 (CAN_RXMDTRY) (y=0/1)

偏移地址: 0x1B4, 0x1C4

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

位	名称	访问	描述	复位值
[31:16]	TIME[15:0]	RO	用于接收报文 SOF 时刻的 16 位定时器值。	0
[15:8]	FMI[7:0]	RO	报文所匹配的过滤器编号。	x
7	Reserved	RO	保留。	0
6	RES	RO	当前接收帧的 RES 位	0
5	ESI	RO	当前接收帧的 ESI 位	0
4	BRS	RO	当前接收帧的 BRS 位	0
[3:0]	DLC[3:0]	RO	接收报文数据长度。 DLC=0-8: 标准 CAN 与 CANFD 相同, 表示数据帧长度 0 到 8, 远程帧为 0。 DLC=9-15: CANFD 独有, 表示数据帧长度详见表 22-5。	x

22.8.27 CAN 接收邮箱低字节数据寄存器 (CAN_RXMDLRy) (y=0/1)

偏移地址: 0x1B8, 0x1C8

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA3[7:0]								DATA2[7:0]							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA1[7:0]								DATA0[7:0]							

位	名称	访问	描述	复位值
[31:24]	DATA3[7:0]	RO	接收报文的数据字节 3。	x
[23:16]	DATA2[7:0]	RO	接收报文的数据字节 2。	x
[15:8]	DATA1[7:0]	RO	接收报文的数据字节 1。	x
[7:0]	DATA0[7:0]	RO	接收报文的数据字节 0。	x

22.8.28 CAN 接收邮箱高字节数据寄存器 (CAN_RXMDHRy) (y=0/1)

偏移地址: 0x1BC, 0x1CC

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA7[7:0]								DATA6[7:0]							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA5[7:0]								DATA4[7:0]							

位	名称	访问	描述	复位值
[31:24]	DATA7[7:0]	RO	接收报文的数据字节 7。	x
[23:16]	DATA6[7:0]	RO	接收报文的数据字节 6。	x
[15:8]	DATA5[7:0]	RO	接收报文的数据字节 5。	x
[7:0]	DATA4[7:0]	RO	接收报文的数据字节 4。	x

22.8.29 CAN 过滤器主控制寄存器 (CAN_FCTLr)

偏移地址: 0x200

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
16' h2a1c															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								Reserved							FINIT

位	名称	访问	描述	复位值
[31:16]		RO	固化为 16' h2a1c	16' h2a1c
[15:8]	Reserved	RO	保留。	0
[7:1]	Reserved	RO	保留。	0
0	FINIT	RW	过滤器初始化模式使能标志位。 1: 过滤器组为初始化模式; 0: 过滤器组为正常模式。	1

22.8.30 CAN 过滤器模式寄存器 (CAN_FMCFR)

偏移地址: 0x204

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				FBM27	FBM26	FBM25	FBM24	FBM23	FBM22	FBM21	FBM20	FBM19	FBM18	FBM17	FBM16
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FBM15	FBM14	FBM13	FBM12	FBM11	FBM10	FBM9	FBM8	FBM7	FBM6	FBM5	FBM4	FBM3	FBM2	FBM1	FBM0

位	名称	访问	描述	复位值
[31:28]	Reserved	RO	保留。	0
[27:0]	FBMx	RW	过滤器组 x 的工作模式控制位, FINIT 为 1 才能写入。	0

			0: 过滤器组 x 的寄存器为屏蔽位模式; 1: 过滤器组 x 的寄存器为标识符列表模式。	
--	--	--	--	--

22.8.31 CAN 过滤器位宽寄存器 (CAN_FSCFGR)

偏移地址: 0x20C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				FSC27	FSC26	FSC25	FSC24	FSC23	FSC22	FSC21	FSC20	FSC19	FSC18	FSC17	FSC16
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FSC15	FSC14	FSC13	FSC12	FSC11	FSC10	FSC9	FSC8	FSC7	FSC6	FSC5	FSC4	FSC3	FSC2	FSC1	FSC0

位	名称	访问	描述	复位值
[31:28]	Reserved	RO	保留。	0
[27:0]	FSCx	RW	过滤器组 x 的位宽控制位, FINT 为 1 才能写入。 1: 过滤器组 x 的寄存器为单个 32 位; 0: 过滤器组 x 的寄存器为 2 个 16 位。	0

22.8.32 CAN 过滤器 FIFO 关联寄存器 (CAN_FAFIFOR)

偏移地址: 0x214

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				FFA27	FFA26	FFA25	FFA24	FFA23	FFA22	FFA21	FFA20	FFA19	FFA18	FFA17	FFA16
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FFA15	FFA14	FFA13	FFA12	FFA11	FFA10	FFA9	FFA8	FFA7	FFA6	FFA5	FFA4	FFA3	FFA2	FFA1	FFA0

位	名称	访问	描述	复位值
[31:28]	Reserved	RO	保留。	0
[27:0]	FFAx	RW	过滤器组 x 的关联 FIFO 控制位, FINT 为 1 才能写入。 1: 过滤器组 x 被关联到 FIFO_1; 0: 过滤器组 x 被关联到 FIFO_0。	0

22.8.33 CAN 过滤器激活寄存器 (CAN_FWR)

偏移地址: 0x21C

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

位	名称	访问	描述	复位值
[31:28]	Reserved	RO	保留。	0

[27:0]	FACT _x	RW	过滤器组 x 的激活控制位。 1: 过滤器组 x 激活; 0: 过滤器组 x 禁用。	0
--------	-------------------	----	--	---

22. 8. 34 CAN 过滤器组的过滤寄存器（CAN_FiRx）（i=0-13, x=1/2）

偏移地址：0x240-0x31C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FB31	FB30	FB29	FB28	FB27	FB26	FB25	FB24	FB23	FB22	FB21	FB20	FB19	FB18	FB17	FB16
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FB15	FB14	FB13	FB12	FB11	FB10	FB9	FB8	FB7	FB6	FB5	FB4	FB3	FB2	FB1	FB0

位	名称	访问	描述	复位值
[31:0]	FB	RW	过滤器组中寄存器的标志位，FINT 为 1 才能写入。 标识符模式 1: 对应位期望电平为隐性位; 0: 对应位期望电平为显性位。 屏蔽位模式 1: 必须和对应的标识符寄存器位一致; 0: 不需要和对应的标识符寄存器位一致。	0

第 23 章 运放（OPA）和比较器（CMP）

该模块包含 1 个可独立配置的运算放大器（OPA 或 PGA）和 3 个可独立配置的电压比较器（CMP），其中运算放大器（OPA 或 PGA）支持增益选择，也可改用于电压比较器。

每个运算放大器的输入和输出均连接至 I/O 口，且输入引脚或增益可选择，输出引脚可选择配置到通用 I/O 口或复用为 ADC 采样通道的 I/O，支持将外部模拟小信号放大送入 ADC 以实现小信号 ADC 转换。

每个电压比较器的输入和输出均连接至 I/O 口，且输入引脚可选择，输出引脚可选择配置到通用 I/O 口或复用为 TIM 内部采样通道（不占用 I/O 引脚）。

23.1 主要特性

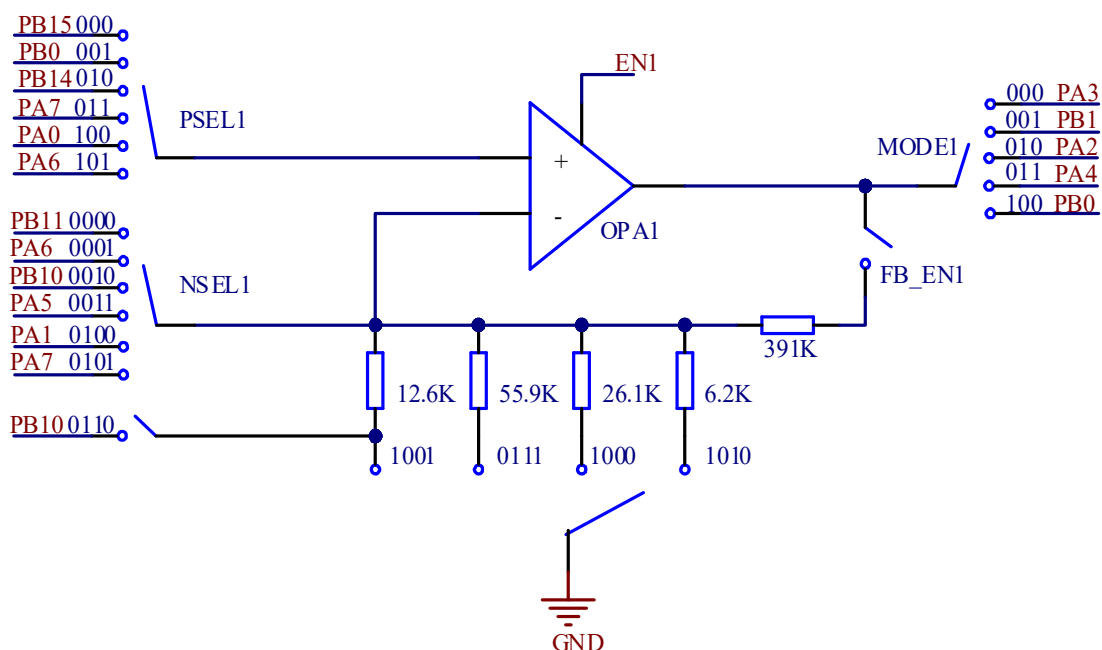
- OPA 输入引脚或通道可选择
- OPA 输出引脚可选择通用 I/O 口或 ADC 采样通道
- OPA 支持正端输入轮询功能
- OPA 支持 PGA 增益选择
- CMP 输入引脚可选择，负端输入通道可选公用引脚
- CMP 输出引脚可选择通用 I/O 口或 TIM 内部采样通道
- OPA 中断可使系统从睡眠模式中唤醒
- CMP 中断可使系统从睡眠和停止模式中唤醒

23.2 功能描述

23.2.1 运放 OPA

置位 OPA_CTLR1 寄存器中的 EN1，即可使能对应的 OPA1，配置 OPA_CTLR1 寄存器中的 MODE1 可选择 OPA1 的输出通道为 ADC 采样通道或者普通 I/O 口，配置 OPA_CTLR1 寄存器中的 PSEL1，可选择 OPA1 的正端输入引脚，配置 OPA_CTLR1 寄存器中的 NSEL1，可选择 OPA1 的负端输入通道、或作为 PGA 使用时的增益。

图 23-1 OPA 结构图



23.2.2 运放正输入端轮询

每个 OPA 的 P 端可从 OPA_P0/OPA_P1/OPA_P2/OPA_P3/OPA_P4/OPA_P5 中选择，OPA 的轮询功能可实现定时依次选择 OPA_P0/OPA_P1/OPA_P2/OPA_P3/OPA_P4/OPA_P5，轮流选中所有的 P 端；可通过配置 OPA_CFGR1 寄存器中的 POLL_EN 位来选择使能轮询功能的 OPA1。（注：P 端轮询顺序不可更改，设置轮询顺序详情见 OPA_CFGR2 寄存器中的 POLL1_NUM[2:0] 位）

其中轮询的通道数量可通过 OPA_CFGR2 寄存器中的 POLL1_NUM[2:0] 配置，轮询间隔可通过 OPA_CFGR2 寄存器中的 POLL_VLU[8:0] 配置，轮询 P 端查询可以通过 OPA_CFGR2 寄存器中 POLL1_CNT 配置；例如 POLL1_CNT=001b，对应的 P 端为 OPA_P1，依次类推可查询对应的轮询 P 端。

23.2.3 运放 OPA 中断

OPA 中断只可唤醒 SLEEP 睡眠模式，OPA 中断配置：

- 1) 在内核的 PFIC 中配置 OPA 中断，以保证其可以正确响应；
- 2) 配置 POLL_EN=1，使能 OPA 轮询功能，配置 POLL_VLU[8:0]，设置进入中断时间。
- 3) 使能 OPA，使能 OPA 中断。

中断配置：

(1) OPA 中断

设置 IE_OUT=1，打开 OPA1 中断使能，当 OPA 输出高电平时进入中断。

(2) OPA 轮询中断

设置 IE_CNT=1，打开 OPA1 轮询间隔结束中断使能，当 OPA 的 P 端轮询一次进入中断。

(3) OPANMI

设置 NMI_EN=1，打开 OPA1 NMI 中断使能，当 OPA 输出为高电平，进入 NMI 中断。

23.2.4 运放 OPA 复位

使能 OPA 功能后，设置 RST_EN=1 开启 OPA 复位功能，OPA 输出为高电平时，系统复位。

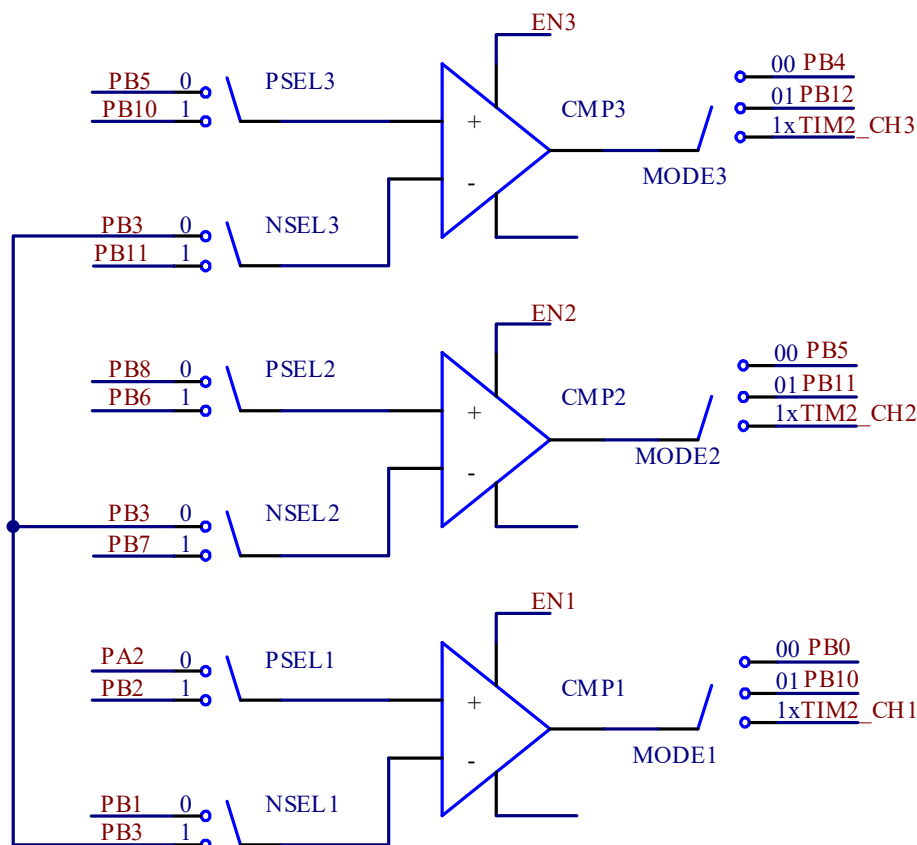
23.2.5 运放 OPA 刹车

通过设置 OPA_CFGR1 寄存器中的 BKIN_EN 位可选择刹车信号源，当 BKIN_EN=1 时，TIM1 的刹车源来自 OPA，此时使用 IO 引脚刹车无效；刹车输入极性当且仅当 OPA 输出为高电平时，刹车信号有效。

23.2.6 比较器 CMP

置位 OPA_CTLR2 寄存器中的 ENx，即可使能对应的 CMPx，配置 OPA_CTLR2 寄存器中的 MODEx 可选择 CMPx 的输出通道为普通 I/O 口或内部定时器通道。配置 OPA_CTLR2 寄存器中的 PSELx，可选择 CMPx 的正端输入引脚，配置 OPA_CTLR2 寄存器中的 NSELx，可选择 CMPx 的负端输入引脚。

图 23-2 CMP 结构图



注：如果比较器输出使用定时通道，通过定时器捕获用于查看比较器输出状态。

23.2.7 比较器 CMP 中断

比较器专属的外部中断(EXTI)线 — EXTI22 (COMP 唤醒事件), 可产生中断或事件, 可用于 SLEEP、STOP 低功耗模式下唤醒。

使用外部中断唤醒时, 所需条件为:

- 1) 配置对应的外部中断通道的事件使能位 (EXTI_EVENR);
- 2) 配置比较器输出电平触发沿, 选择上升沿触发、下降沿触发或双边沿触发;
- 3) 在内核的 PFIC 中配置 EXTI 中断, 以保证其可以正确响应。
- 4) 使能 CMP。

使用事件时, 所需条件为:

- 1) 配置对应的外部中断通道的事件使能位 (EXTI_EVENR);
- 2) 配置比较器输出电平触发沿, 选择上升沿触发、下降沿触发或双边沿触发;
- 3) 使能 CMP。

通过配置 WKUP_MD[1:0] 来选择 CMP 输出信号的唤醒源。

表 23-1 CMP 输出信号的唤醒源选择

设置 WKUP_MD[1:0]	01	10	11	00
唤醒 CMP 输出信号的电平	双边沿	上升沿	下降沿	无效

注：当 CMP 将产生一个中断请求, 对应的中断线 22 标志位也会被置位。对标志位写 1 可以清除该标志位。

23.2.8 比较器 CMP 低功耗

置位 LPx 位, CMPx 进入低功耗模式, 电流对应减少约 3.2 μ A, 且会延长 CMP 的比较时间, 增加输出失调电压。其中对应参数值请参考《CH32L103DS0》手册。

23.3 寄存器描述

表 23-2 OPA 相关寄存器列表

名称	访问地址	描述	复位值
R16_OPA_CFGR1	0x40026000	OPA 配置寄存器 1	0x0080
R16_OPA_CFGR2	0x40026002	OPA 配置寄存器 2	0x0000
R32_OPA_CTLR1	0x40026004	OPA 控制寄存器 1	0x01D801D8
R32_OPA_CTLR2	0x40026008	OPA 控制寄存器 2	0x00000000
R32_OPCM_KEY	0x40026014	OPA 及 CMP 解锁键寄存器	0xFFFFFFFF

23.3.1 OPA 配置寄存器 1 (OPA_CFGR1)

偏移地址: 0x00

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	IF_CNT	Reserved	IF_OUT	NMI_EN	IE_CNT	Reserved	IE_OUT	OPCM_LOCK	Reserved	RST_EN	Reserved	BKIN_EN	Reserved	POLL_EN	

位	名称	访问	描述	复位值
15	Reserved	RO	保留。	0
14	IF_CNT	RWO	OPA 轮询间隔结束的中断标志: 0: 无效; 1: 轮询间隔结束。 写 0 清零、写 1 无效。	0
13	Reserved	RO	保留	0
12	IF_OUT	RWO	轮询到 OPA1 输出为高电平的中断标志: 0: 无效; 1: 轮询到 OPA1 输出为高电平。 写 0 清零、写 1 无效。	0
11	NMI_EN	RW	OPA 连接 NMI 中断使能: 0: 关闭; 1: 开启。	0
10	IE_CNT	RW	OPA 轮询间隔结束的中断使能: 0: 关闭中断使能; 1: 打开中断使能。	0
9	Reserved	RO	保留	0
8	IE_OUT	RW	OPA1 中断使能: 0: 关闭中断使能; 1: 打开中断使能	0
7	OPCM_LOCK	RW	OPA 及 CMP 锁(写 1 上锁, 写 0 无效) 0: 解锁, 可以对配置寄存器的其他位进行写操作 1: 上锁, 无法对配置寄存器的其他位进行写操作	1
[6:5]	Reserved	RO	保留	0

4	RST_EN	RW	OPA1 复位系统使能： 0：关闭复位使能； 1：打开复位使能。	0
3	Reserved	R0	保留	0
2	BKIN_EN	RW	TIM1 的 BKIN 输入源选择位： 0：TIM1 的 BKIN 来自 IO； 1：TIM1 的 BKIN 来自 OPA 输出。	0
1	Reserved	R0	保留	0
0	POLL_EN	RW	OPA1 正端轮询使能 0：关闭 1：开启	0

23.3.2 OPA 配置寄存器 2 (OPA_CFGR2)

偏移地址：0x02

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	POLL1_CNT[2:0]			POLL1_NUM[2:0]			POLL_VLU[8:0]								

位	名称	访问	描述	复位值
15	Reserved	R0	保留	0
[14:12]	POLL1_CNT[2:0]	R0	查询 OPA1 轮询的正端数： 000：01P0 001：01P1 010：01P2 011：01P3 100：01P4 101：01P5 其他：保留	0
[11:9]	POLL1_NUM[2:0]	RW	配置 OPA1 轮询的正端个数 000：1 个，01P0 001：2 个，01P0+01P1 010：3 个，01P0+01P1+01P2 011：4 个，01P0+01P1+01P2+01P3 100：5 个，01P0+01P1+01P2+01P3+01P4 101：6 个，01P0+01P1+01P2+01P3+01P4+01P5 其他：保留	
[8:0]	POLL_VLU[8:0]	RW	配置 OPA1 轮询采样间隔时间 轮询间隔= (POLL_VLU+1) *1us。	0

23.3.3 OPA 控制寄存器 1 (OPA_CTLR1)

偏移地址：0x04

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

Reserved	LP1	NSEL1[3:0]	FB_EN1	PSEL1[2:0]	MODE1[2:0]	EN1
----------	-----	------------	--------	------------	------------	-----

位	名称	访问	描述	复位值
[31:30]	Reserved	R0	保留。	0
[21:16]	ITRIMN[5:0]	RW	OPA1 低电平修调电压	0
[23:22]	Reserved	R0	保留。	0
[21:16]	ITRIMP[5:0]	RW	OPA1 高电平修调电压	0
[15:13]	Reserved	R0	保留	0
12	LP1	RW	OPA1 低功耗模式选择: 0: 关闭 1: 打开	0
[11:8]	NSEL1[3:0]	RW	OPA1 负端通道选择与 PGA 增益选择 0000: PB11 0001: PA6 0010: PB10 0011: PA5 0100: PA1 0101: PA7 0110: PB10, PGA 模式, 内部增益为 32, 反馈电阻 391k Ω 0111: PGA 模式, 无负端输入通道, 内部增益为 8 1000: PGA 模式, 无负端输入通道, 内部增益为 16 1001: PGA 模式, 无负端输入通道, 内部增益为 32 1010: PGA 模式, 无负端输入通道, 内部增益为 64 1111: 均不选择, 负端通道处于高阻态 其他: 保留	1111b
7	FB_EN1	RW	OPA1 内部反馈电阻使能 0: 禁止 1: 使能	0
[6:4]	PSEL1[2:0]	RW	OPA1 正向输入端选择 000: PB15 001: PB0 010: PB14 011: PA7 100: PA0 101: PA6 其他: 均不选择, 正端通道处于高阻态	111b
[3:1]	MODE1[2:0]	RW	OPA1 输出通道选择 000: 输出信号通过 PA3 输出 001: 输出信号通过 PB1 输出 010: 输出信号通过 PA2 输出 011: 输出信号通过 PA4 输出 100: 输出信号通过 PB0 输出 其他: 均不选择, 输出通道均处于高阻态	111b
0	EN1	RW	OPA1 使能 0: 关闭	0

			1: 开启	
--	--	--	-------	--

23.3.4 OPA 控制寄存器 2 (OPA_CTLR2)

偏移地址: 0x08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved						WKUP_MD[1:0]	Reserved	LP3	HYEN3	PSEL3	NSEL3	MODE3[1:0]	EN3		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	LP2	HYEN2	PSEL2	NSEL2	MODE2[1:0]	EN2	Reserved	LP1	HYEN1	PSEL1	NSEL1	MODE1[1:0]	EN1		

位	名称	访问	描述	复位值
[31:26]	Reserved	R0	保留	0
[25:24]	WKUP_MD[1:0]	RW	CMP 唤醒电平配置: 00: 关闭 CMP 唤醒功能; 01: 比较器的输出的上升沿和下降沿都可唤醒系统; 10: 比较器的输出的上升沿唤醒系统; 11: 比较器的输出的下降沿唤醒系统。	0
23	Reserved	R0	保留	0
22	LP3	RW	CMP3 低功耗开关 0: 关闭 1: 打开, 3.2 μ A	0
21	HYEN3	RW	CMP3 比较器迟滞功能选择 0: 关闭 1: 打开, ± 15 mV	0
20	PSEL3	RW	CMP3 正端输入通道选择 0: PB5 1: PB10	0
19	NSEL3	RW	CMP3 负端输入通道选择 0: PB3 1: PB11	0
[18:17]	MODE3[1:0]	RW	CMP3 输出通道选择 00: 输出通道为 PB4 01: 输出通道为 PB12 1x: 输出通道为内部通道 TIM2_CH3	0
16	EN3	RW	CMP3 使能 0: 禁止 CMP3 1: 使能 CMP3	0
15	Reserved	R0	保留	0
14	LP2	RW	CMP2 低功耗开关 0: 关闭 1: 打开, 3.2 μ A	0
13	HYEN2	RW	CMP2 比较器迟滞功能选择 0: 关闭	0

			1: 打开, $\pm 15\text{mV}$	
12	PSEL2	RW	CMP2 正端输入通道选择 0: PB8 1: PB6	0
11	NSEL2	RW	CMP2 负端输入通道选择 0: PB3 1: PB7	0
[10:9]	MODE2[1:0]	RW	CMP2 输出通道选择 00: 输出通道为 PB5 01: 输出通道为 PB11 1x: 输出通道为内部通道 TIM2_CH2	0
8	EN2	RW	CMP2 使能: 0: 禁止 CMP2 1: 使能 CMP2	0
7	Reserved	RO	保留	0
6	LP1	RW	CMP1 低功耗开关 0: 关闭 1: 打开, $3.2\mu\text{A}$	0
5	HYEN1	RW	CMP1 比较器迟滞功能选择 0: 关闭 1: 打开, $\pm 15\text{mV}$	0
4	PSEL1	RW	CMP1 正端输入通道选择 0: PA2 1: PB2	0
3	NSEL1	RW	CMP1 负端输入通道选择 0: PB1 1: PB3	0
[2:1]	MODE1[1:0]	RW	CMP1 输出通道选择 00: 输出通道为 PB0 01: 输出通道为 PB10 1x: 输出通道为内部通道 TIM2_CH1	0
0	EN1	RW	CMP1 使能 0: 禁止 CMP1 1: 使能 CMP1	0

23.3.5 OPA/CMP 解锁键寄存器 (OPCM_KEY)

偏移地址: $0\text{x}14$

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

位	名称	访问	描述	复位值
[31:0]	OPCM_KEY[31:0]	RW	OPA 键/CMP 键, 用于输入 OPA/CMP 的解锁键包括:	X

			KEY1 = 0x45670123; KEY2 = 0xCDEF89AB。	
--	--	--	--	--

第 24 章 闪存及用户选择字（FLASH）

24.1 闪存组织

芯片内部闪存组织结构如下：

表 24-1 闪存组织结构

块	名称	地址范围	大小（字节）
主存储器	页 0	0x0800 0000 - 0x0800 00FF	256
	页 1	0x0800 0100 - 0x0800 01FF	256
	页 2	0x0800 0200 - 0x0800 02FF	256
	页 3	0x0800 0300 - 0x0800 03FF	256

	页 255	0x0800 FF00 - 0x0800 FFFF	256
信息块	启动程序代码	0x1FFF 0000 - 0x1FFF 0CFF	3K+256
	用户选择字	0x1FFF F800 - 0x1FFF F8FF	256
	厂商配置字	0x1FFF F700 - 0x1FFF F7FF	256

注：1) 上述主存储器区域用于用户的应用程序存储，以 2K 字节（8 页）单位进行写保护划分；除了“厂商配置字”区域出厂锁定，用户不可访问，其他区域在一定条件下用户可操作。

24.2 闪存编程及安全性

24.2.1 编程/擦除方式

- 快速编程：此方式采用页操作方式。经过特定序列解锁后，执行单次 256 字节的编程及 256 字节擦除、1K 字节擦除及整片擦除。

24.2.2 安全性-防止非法访问（读、写、擦）

- 页写入保护
- 读保护

芯片处于读保护状态下时：

- 1) 主存储器 0-7 页（2K 字节）自动写保护状态，不受 FLASH_WPR 寄存器控制；解除读保护状态，所有主存储页都由 FLASH_WPR 寄存器控制。
- 2) 系统引导代码区、SDI 模式、RAM 区域都不可对主存储器进行擦除或编程，整片擦除除外。可擦除或编程用户选择字区域。如果试图解除读保护（编程用户字），芯片将自动擦除整片用户区。

注：进行闪存的编程/擦除操作时，必须打开内部 RC 振荡器（HSI）。

24.3 寄存器描述

表 24-2 FLASH 相关寄存器列表

名称	访问地址	描述	复位值
R32_FLASH_ACTLR	0x40022000	访问控制寄存器	0x00000000
R32_FLASH_KEYR	0x40022004	FPEC 键寄存器	0xFFFFFFFF
R32_FLASH_OBKEYR	0x40022008	OBKEY 寄存器	0xFFFFFFFF
R32_FLASH_STATR	0x4002200C	状态寄存器	0x00000000
R32_FLASH_CTLR	0x40022010	控制寄存器	0x00008080

R32_FLASH_ADDR	0x40022014	地址寄存器	0x00000000
R32_FLASH_OBR	0x4002201C	选择字寄存器	0x0XXXXXFE
R32_FLASH_WPR	0x40022020	写保护寄存器	0xFFFFFFFF
R32_FLASH_MODEKEYR	0x40022024	扩展键寄存器	0xFFFFFFFF

24.3.1 访问控制寄存器 (FLASH_ACTLR)

偏移地址：0x00

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

位	名称	访问	描述	复位值
[31:2]	Reserved	RO	保留	0
[1:0]	LATENCY	RW	FLASH 等待状态数 00: 0 等待 (HCLK≤40MHz) 01: 1 等待 (40MHz<HCLK≤72MHz) 10: 2 等待 (HCLK>72MHz) 其他: 无效	00b

24.3.2 FPEC 键寄存器 (FLASH_KEYR)

偏移地址：0x04

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

位	名称	访问	描述	复位值
[31:0]	KEYR[31:0]	WO	FPEC 键，用于输入 FPEC 的解锁键包括： KEY1 = 0x45670123； KEY2 = 0xCDEF89AB。	x

24.3.3 OBKEY 寄存器 (FLASH_OBKEYR)

偏移地址：0x08

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

位	名称	访问	描述	复位值
[31:0]	OBKEYR[31:0]	WO	选择字键，用于输入选择字键解除 OPTWRE。	x

			KEY1 = 0x45670123; KEY2 = 0xCDEF89AB。 (注意: 需要先解锁 FLASH)	
--	--	--	---	--

24.3.4 状态寄存器 (FLASH_STATR)

偏移地址: 0x0C

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

位	名称	访问	描述	复位值
[31:8]	Reserved	RO	保留。	0
7	TURBO	RO	TURBO 1: 设置进入 TURBO 模式, 增加约 100 μ A 静态电流, 可加速 SLEEP 睡眠后醒来的过程 0: 无影响	0
6	FWAKE_FLAG	RWO	FLASH 唤醒标志, 写 0 清零 1: FLASH 被唤醒 0: 无影响	0
5	EOP	RW1	指示操作结束, 写 1 清零。 每次成功擦除或编程时, 硬件会置位。	0
4	WRPRTERR	RW1	指示写保护错误, 写 1 清零。 如果对写保护的地址编程时, 硬件会置位。	0
[3:1]	Reserved	RO	保留。	0
0	BSY	RO	指示忙状态: 1: 表示闪存操作正在进行; 0: 操作结束。	0

注: 进行编程操作时, 需要确定 FLASH_CTLR 寄存器的 STRT 位为 0。

24.3.5 控制寄存器 (FLASH_CTLR)

偏移地址: 0x10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved								BANK_ERASE	Reserved			BUFRST	BUFLoad	FTER	FTPG
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FLOCK	Reserved	FWAKEIE	EOPIE	Reserved	ERRIE	OBWRE	Reserved	LOCK	STRT	OBER	OBPG	Reserved	MER	SER	Reserved

位	名称	访问	描述	复位值
[31:25]	Reserved	RO	保留。	0
23	BER32	RW	执行块擦除 32KB	0
[22:20]	Reserved	RO	保留。	0
19	BUFRST	RW	BUF 复位操作	0

18	BUFLOAD	RW	将数据缓存到 BUF 中	0
17	FTER	RW	执行快速页（256Byte）擦除操作。	0
16	FTPG	RW	执行快速页编程操作。	0
15	FLOCK	RW1	快速编程锁。只能写‘1’。当该位为‘1’时表示快速编程/擦除模式不可用。在检测到正确的解锁序列后，硬件清除此位为‘0’。软件置1，重新加锁。	1
14	Reserved	RO	保留。	0
13	FWAKEIE	RW	FLASH 唤醒中断使能： 1：允许产生中断 0：禁止产生中断	0
12	EOPIE	RW	操作完成中断控制（FLASH_STATR 寄存器中 EOP 置位）： 1：允许产生中断； 0：禁止产生中断。	0
11	Reserved	RO	保留。	0
10	ERRIE	RW	错误状态中断控制（FLASH_STATR 寄存器中 PGERR/WRPRTERR 置位）： 1：允许产生中断； 0：禁止产生中断。	0
9	OBWRE	RW0	用户选择字锁，软件清0： 1：表示可以对用户选择字进行编程操作。需要在 FLASH_OBKEYR 寄存器中写入正确序列后由硬件置位。 0：软件清零后重新加锁用户选择字。	0
8	Reserved	RO	保留。	0
7	LOCK	RW1	锁。只能写‘1’。当该位为‘1’时表示 FPEC 和 FLASH_CTLR 被锁住不可写。在检测到正确的解锁序列后，硬件清除此位为‘0’。在一次不成功的解锁操作后，直到下次系统复位前，该位不会再改变。	1
6	STRT	RW1	开始。置1启动一次擦除动作，硬件自动清0（BSY 变‘0’）。	0
5	OBER	RW	执行用户选择字擦除	0
4	OBPG	RW	执行用户选择字编程	0
3	Reserved	RO	保留。	0
2	MER	RW	执行全擦除操作（擦除整个用户区）。	0
1	PER	RW	执行扇区擦除	0
0	Reserved	RO	保留。	0

24.3.6 地址寄存器（FLASH_ADDR）

偏移地址：0x14

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

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

位	名称	访问	描述	复位值
[31:0]	FAR	RW	闪存地址，进行编程时为编程的地址，进行擦除时为擦除的起始地址。 当 FLASH_SR 寄存器中的 BSY 位为 ‘1’ 时，不能写此寄存器。	0

24.3.7 选择字寄存器 (FLASH_OBR)

偏移地址：0x1C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved						DATA1								DATA0	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA0						FIX_11		CFGCANM	Reserved		STANDY RST	STOP RST	IWDG SW	RDPRT	OBERR

位	名称	访问	描述	复位值
[31:26]	Reserved	RO	保留。	0
[25:18]	DATA1	RO	数据字节 1	X
[17:10]	DATA0	RO	数据字节 0	X
[9:8]	FIX_11	RO	固定为 11	11b
7	USER	CFGCANM	配置 CAN 离线恢复时间， 1：从离线恢复到正常更快一些， 0：从离线恢复到正常符合 CAN 协议	1
[6:5]		Reserved	保留	11b
4		STANDY_RST	待机模式下系统复位控制，低电平有效。	1
3		STOP_RST	停止模式下系统复位控制，低电平有效。	1
2		IWDG_SW	独立看门狗（IWDG）硬件使能位，低电平有效。	1
1	RDPRT		读保护状态。 1：表示闪存当前读保护有效。	1
0	OBERR		选择字错误。 1：表示选择字和它的反码不匹配。	0

注：USER 和 RDPRT 在系统复位后从用户选择字区域加载。

24.3.8 写保护寄存器 (FLASH_WPR)

偏移地址：0x20

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

WRP[15:0]				
-----------	--	--	--	--

位	名称	访问	描述	复位值
[31:0]	WRP	RO	闪存写保护状态。 1：写保护失效； 0：写保护有效。 每个比特位代表 2K 字节（8 页）存储写保护状态。	X

注：WPR 在系统复位后从用户选择字区域加载。

24.3.9 扩展键寄存器（FLASH_MODEKEYR）

偏移地址：0x24

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

位	名称	访问	描述	复位值
[31:0]	MODEKEYR	WO	输入下面序列解锁快速编程/擦除模式： KEY1 = 0x45670123； KEY2 = 0xCDEF89AB。	X

24.4 闪存操作流程

24.4.1 读操作

在通用地址空间内进行直接寻址，任何 8/16/32 位数据的读操作都能访问闪存模块的内容并得到相应的数据。

24.4.2 解除闪存锁

系统复位后，闪存控制器（FPEC）和 FLASH_CTLR 寄存器是被锁定的，不可访问。通过写入序列到 FLASH_KEYR 寄存器可解锁闪存控制器模块。

解锁序列：

- 1) 向 FLASH_KEYR 寄存器写入 KEY1 = 0x45670123（第 1 步必须是 KEY1）；
- 2) 向 FLASH_KEYR 寄存器写入 KEY2 = 0xCDEF89AB（第 2 步必须是 KEY2）。

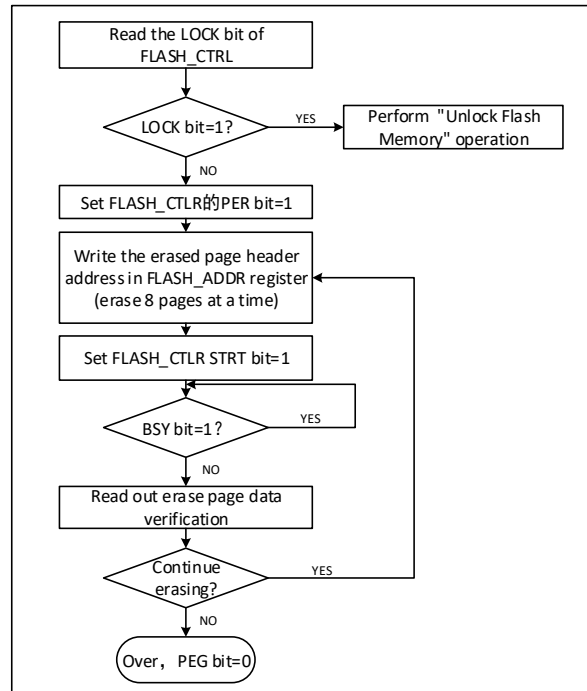
上述操作必须按序并连续执行，否则属于错误操作，会锁定 FPEC 模块和 FLASH_CTLR 寄存器并产生总线错误，直到下次系统复位。

闪存控制器（FPEC）和 FLASH_CTLR 寄存器可以通过将 FLASH_CTLR 寄存器的“LOCK”位，置 1 来再次锁定。

24.4.3 主存储器标准擦除

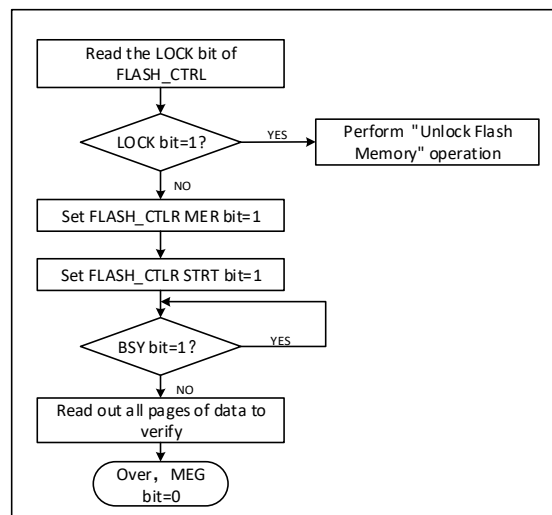
闪存可以按标准页（1K 字节）擦除，也可以整片擦除。

图 24-1 FLASH 页擦除



- 1) 检查 FLASH_CTRL 寄存器 LOCK 位，如果为 1，需要执行“解除闪存锁”操作。
 - 2) 设置 FLASH_CTRL 寄存器的 PER 位为 ‘1’，开启标准页擦除模式。
 - 3) 向 FLASH_ADDR 寄存器写入选择擦除的页首地址。
 - 4) 设置 FLASH_CTRL 寄存器的 STAT 位为 ‘1’，启动一次擦除动作。
 - 5) 等待 BSY 位变为 ‘0’ 或 FLASH_STATR 寄存器的 EOP 位为 ‘1’ 表示擦除结束，将 EOP 位清 0。
 - 6) 读擦除页的数据进行校验。
 - 7) 继续标准页擦除可以重复 3-5 步骤，结束擦除将 PEG 位清 0。
- 注：擦除成功后，字读- 0xFF。

图 24-2 FLASH 整片擦除



- 1) 检查 FLASH_CTRL 寄存器 LOCK 位，如果为 1，需要执行“解除闪存锁”操作。
- 2) 设置 FLASH_CTRL 寄存器的 MER 位为 ‘1’，开启整片擦除模式。
- 3) 设置 FLASH_CTRL 寄存器的 STAT 位为 ‘1’，启动擦除动作。
- 4) 等待 BSY 位变为 ‘0’ 或 FLASH_STATR 寄存器的 EOP 位为 ‘1’ 表示擦除结束，将 EOP 位清 0。

- 5) 读擦除页的数据进行校验。
- 6) 将 MER 位清 0。

24.4.4 快速编程模式解锁

通过写入序列到 FLASH_MODEKEYR 寄存器可解锁快速编程模式操作。解锁后, FLASH_CTLR 寄存器的 FLOCK 位将清 0, 表示可以进行快速擦除和编程操作。通过将 FLASH_CTLR 寄存器的“FLOCK”位软件置 1 来再次锁定。

解锁序列:

- 1) 向 FLASH_MODEKEYR 寄存器写入 KEY1 = 0x45670123;
- 2) 向 FLASH_MODEKEYR 寄存器写入 KEY2 = 0xCDEF89AB。

上述操作必须按序并连续执行, 否则属于错误操作会锁定, 直到下次系统复位才能重新解锁。

注: 快速编程操作需要解除“LOCK”和“FLOCK”两层锁定。

24.4.5 主存储器快速编程

快速编程按页(256 字节)进行编程。

- 1) 检查 FLASH_CTLR 寄存器 LOCK 位, 如果为‘1’, 需要执行“解除闪存锁”操作。
- 2) 检查 FLASH_CTLR 寄存器 FLOCK 位, 如果为‘1’, 需要执行“快速编程模式解锁”操作。
- 3) 检查 FLASH_STATR 寄存器的 BSY 位, 以确认没有其他正在进行的编程操作。
- 4) 设置 FLASH_CTLR 寄存器的 FTPG 位为‘1’, 使能快速页编程模式。
- 5) 设置 FLASH_CTLR 寄存器的 BUFRST 位, 执行清除内部 256 字节缓存区操作。
- 6) 等待 BYS 位变为‘0’或 FLASH_STATR 寄存器的 EOP 位为‘1’表示清除结束, 将 EOP 位清 0
- 7) 使用 32 位方式向 FLASH 地址写入数据, 例如
* (uint32_t*) 0x80000000 = 0x12345678;
- 8) 然后设置 FLASH_CTLR 寄存器的 BUFLD 位, 执行加载到缓存区。
- 9) 等待 FLASH_STATR 寄存器的 WR_BSY 为‘0’, 写入下个数据。
- 10) 重复步骤 7-9 共 4 次, 将 16 字节数据都加载到缓存区(主要 4 轮操作地址要连续)。
- 11) 向 FLASH_ADDR 寄存器写入快速编程页的首地址。
- 12) 设置 FLASH_CTLR 寄存器的 STRT 位为‘1’, 启动快速页编程。
- 13) 等待 BSY 位变为‘0’或 FLASH_STATR 寄存器的 EOP 位为‘1’表示一次快速页编程完成, 将 EOP 位清 0。
- 14) 查询 FLASH_STATR 寄存器看是否有错误, 或者读编程地址数据校验。
- 15) 继续快速页编程可以重复 5-14 步骤, 结束编程将 FTPG 位清 0。

24.4.6 主存储器快速擦除

快速擦除按页(256 字节)进行擦除。

- 1) 检查 FLASH_CTLR 寄存器 LOCK 位, 如果为 1, 需要执行“解除闪存锁”操作。
- 2) 检查 FLASH_CTLR 寄存器 FLOCK 位, 如果为 1, 需要执行“快速编程模式解锁”操作。
- 3) 检查 FLASH_STATR 寄存器的 BSY 位, 以确认没有其他正在进行的编程操作。
- 4) 设置 FLASH_CTLR 寄存器的 FTER 位为‘1’, 开启快速页擦除(256 字节)模式功能。
- 5) 向 FLASH_ADDR 寄存器写入快速擦除页的首地址。
- 6) 设置 FLASH_CTLR 寄存器的 STAT 位为‘1’, 启动一次快速页擦除(256 字节)动作。
- 7) 等待 BSY 位变为‘0’或 FLASH_STATR 寄存器的 EOP 位为‘1’表示擦除结束, 将 EOP 位清 0。
- 8) 查询 FLASH_STATR 寄存器看是否有错误, 或者读擦除页地址数据校验。
- 9) 继续快速页擦除可以重复 5-8 步骤, 结束擦除将 FTER 位清 0。

注: 擦除成功后, 字读- 0xFF。

快速擦除按块（32K 字节）进行擦除。

- 1) 检查 FLASH_CTLR 寄存器 LOCK 位，如果为 1，需要执行“解除闪存锁”操作。
- 2) 检查 FLASH_CTLR 寄存器 FLOCK 位，如果为 1，需要执行“快速编程模式解锁”操作。
- 3) 检查 FLASH_STATR 寄存器的 BSY 位，以确认没有其他正在进行的编程操作。
- 4) 设置 FLASH_CTLR 寄存器的 BER32 位为 ‘1’，开启快速块擦除（32K 字节）模式功能。
- 5) 向 FLASH_ADDR 寄存器写入快速擦除块的首地址。
- 6) 设置 FLASH_CTLR 寄存器的 STAT 位为 ‘1’，启动一次快速块擦除（32K 字节）动作。
- 7) 等待 BYS 位变为 ‘0’ 或 FLASH_STATR 寄存器的 EOP 位为 ‘1’ 表示擦除结束，将 EOP 位清 0。
- 8) 查询 FLASH_STATR 寄存器看是否有错误，或者读擦除页地址数据校验。
- 9) 继续快速页擦除可以重复 5-8 步骤，结束擦除将 BER32 位清 0。

注：擦除成功后，字读- 0xFF。

24.5 用户选择字

用户选择字固化在 FLASH 中，在系统复位后会被重新装载到相应寄存器，用户可以任意的进行擦除和编程。用户选择字信息块总共有 8 个字节（4 个字节为写保护，1 个字节为读保护，1 个字节为配置选项，2 个字节存储用户数据），每个位都有其反码位用于装载过程中的校验。下面描述了选择字信息结构和意义。

表 24-3 32 位选择字格式划分

[31:24]	[23:16]	[15:8]	[7:0]
选择字字节 1 反码	选择字字节 1	选择字字节 0 反码	选择字字节 0

表 24-4 用户选择字信息结构

地址 位	[31:24]	[23:16]	[15:8]	[7:0]
0x1FFFF800	nUSER	USER	nRDPR	RDPR
0x1FFFF804	nData1	Data1	nData0	Data0
0x1FFFF808	nWRPR1	WRPR1	nWRPR0	WRPR0
0x1FFFF80C	nWRPR3	WRPR3	nWRPR2	WRPR2

名称/字节			描述	复位值
RDPR			读保护控制位，配置是否可以读出闪存中的代码。 0xA5：若此字节为 0xA5（nRDP 必须为 0x5A），表示当前代码处于非读保护状态，可以读出； 其他值：表示代码读保护状态，不可读，0-31 页（4K）将自动写保护，不受 WRPR0 控制。	0xA5
USER	[7:6]	Reserved	保留。	11b
	5	CFGCANM	配置 CAN 离线恢复时间： 1：从离线恢复到正常更快一些， 0：从离线恢复到正常符合 CAN 协议	1
	[4:3]	Reserved	保留。	11b
	2	STANDYRST	待机模式下系统复位控制： 1：不启用，进入待机模式系统不复位； 0：启用，进入待机模式产生系统复位。	1

	1	STOPRST	停止模式下系统复位控制： 1：不启用，进入停止模式不复位系统； 0：启用，进入停止模式产生系统复位。	1
	0	IWDGSW	独立看门狗（IWDG）硬件使能位： 1：IWDG 功能由软件开启，禁止硬件开启； 0：IWDG 功能由硬件开启（随 LSI 时钟决定）。	1
Data0 - Data1			存储用户数据 2 字节。	FFFFh
WRPRO - WRPR3			写保护控制位。每个比特位用于控制主存储器中 2 个扇区（1K 字节/扇区）的写保护状态： 1：关闭写保护； 0：启用写保护。 4 个字节用于保护总共 64K 字节的主存储器。 WRP0：第 0-15 扇区存储写保护控制； WRP1：第 16-31 扇区存储写保护控制； WRP2：第 32-47 扇区存储写保护控制； WRP3：第 48-63 扇区存储写保护控制。	FFFFFFFFh

24.5.1 用户选择字解锁

通过写入序列到 FLASH_OBKEYR 寄存器可解锁用户选择字操作。解锁后，FLASH_CTLR 寄存器的 OBWRE 位将置 1，表示可以进行用户选择字的擦除和编程。通过将 FLASH_CTLR 寄存器的“OBWRE”位，软件清 0 来再次锁定。

解锁序列：

- 1) 向 FLASH_OBKEYR 寄存器写入 KEY1 = 0x45670123；
- 2) 向 FLASH_OBKEYR 寄存器写入 KEY2 = 0xCDEF89AB。

注：用户选择字操作需要解除“LOCK”和“OBWRE”两层锁定。

24.5.2 用户选择字编程

- 1) 检查 FLASH_CTLR 寄存器 LOCK 位，如果为‘1’，需要执行“解除闪存锁”操作。
 - 2) 检查 FLASH_CTLR 寄存器 FLOCK 位，如果为‘1’，需要执行“快速编程模式解锁”操作。
 - 3) 检查 FLASH_STATR 寄存器的 BSY 位，以确认没有其他正在进行的编程操作。
 - 4) 设置 FLASH_CTLR 寄存器的 FTPG 位为‘1’，使能快速页编程模式。
 - 5) 设置 FLASH_CTLR 寄存器的 BUFRST 位，执行清除内部 256 字节缓存区操作。
 - 6) 等待 BYS 位变为‘0’或 FLASH_STATR 寄存器的 EOP 位为‘1’表示清除结束，将 EOP 位清 0
 - 7) 使用 32 位方式向 FLASH 地址写入数据，例如
* (uint32_t*) 0x1FFFF804= 0x5AA55AA5；
 - 8) 然后设置 FLASH_CTLR 寄存器的 BUFLD 位，执行加载到缓存区。
 - 9) 等待 FLASH_STATR 寄存器的 WR_BSY 为‘0’，写入下个数据。
 - 10) 重复步骤 5-9 共 4 次，将 256 字节数据都加载到缓存区（主要 16 轮操作地址要连续）。
 - 11) 向 FLASH_ADDR 寄存器写入快速编程页的首地址。
 - 12) 设置 FLASH_CTLR 寄存器的 STRT 位为‘1’，启动快速页编程。
 - 13) 等待 BSY 位变为‘0’或 FLASH_STATR 寄存器的 EOP 位为‘1’表示一次快速页编程完成，将 EOP 位清 0。
 - 14) 查询 FLASH_STATR 寄存器看是否有错误，或者读编程地址数据校验。
 - 15) 继续快速页编程可以重复 5-14 步骤，结束编程将 FTPG 位清 0。
- 注：当修改选择字中的“读保护”变成“非保护”状态时，会自动执行一次整片擦除主存储区操作。如果修改“读保护”之外的选型，则不会出现整片擦除的操作。

24.5.3 用户选择字擦除

直接擦除整个 256 字节用户选择字区域。

- 1) 检查 FLASH_CTLR 寄存器 LOCK 位，如果为 1，需要执行“解除闪存锁”操作。
- 2) 检查 FLASH_STATR 寄存器的 BSY 位，以确认没有正在进行的编程操作。
- 3) 检查 FLASH_CTLR 寄存器 OBWRE 位，如果为 0，需要执行“用户选择字解锁”操作。
- 4) 设置 FLASH_CTLR 寄存器的 OBER 位为 ‘1’，之后设置 FLASH_CTLR 寄存器的 STAT 位为 ‘1’，开启用户选择字擦除。
- 5) 等待 BSY 位变为 ‘0’ 或 FLASH_STATR 寄存器的 EOP 位为 ‘1’ 表示擦除结束，将 EOP 位清 0
- 6) 读擦除地址数据校验。
- 7) 结束将 OBER 位清 0。

注：擦除成功后，字读- 0xFF。

24.5.4 解除读保护

闪存是否读保护，由用户选择字决定。读取 FLASH_OBR 寄存器，当 RDPRT 位为 ‘1’ 表示当前闪存处于读保护状态，闪存操作上受到读保护状态的一系列安全防护。解除读保护过程如下：

- 1) 擦除整个用户选择字区域，此时读保护字段 RDPR，此时读保护仍然有效。
- 2) 用户选择字编程，写入正确的 RDPR 代码 0xA5 以解除闪存的读保护。（此步骤首先将导致系统自动对闪存执行整片擦除操作）
- 3) 进行上电复位以重新加载选择字节（包括新的 RDPR 码），此时读保护被解除。

24.5.5 解除写保护

闪存是否写保护，由用户选择字决定。读取 FLASH_WPR 寄存器，每个比特位代表 2K 字节闪存空间，当比特位为 ‘1’ 表示非写保护状态，为 ‘0’ 表示写保护。解除写保护过程如下：

- 1) 擦除整个用户选择字区域。
- 2) 写入正确的 RDPR 码 0xA5，允许读访问；
- 3) 进行系统复位，重新加载选择字节（包括新的 WRPR[3:0] 字节），写保护被解除。

第 25 章 扩展配置 (EXTEN)

25.1 扩展配置

系统提供了 EXTEN 扩展配置单元 (EXTEN_CTR 寄存器)。该单元使用 HB 时钟，只在系统复位执行复位动作。主要包括以下几个扩展控制位功能：

- 1) 调节内核电压：LDOTRIM 和 ULLDOTRIM 字段选择默认值，在调节性能和功耗时可以修改其值。
- 2) PLL 时钟选择：HSIPRE 字段配合原有的时钟配置寄存器，提供了 HSI 时钟进行分频或不分频作为 PLL 的输入时钟的选择。
- 3) Lock-up 功能监控：LKUPEN 字段启用，将打开系统的 Lock-up 情况监控，一旦发生 Lock-up 情况，系统将进行复位，并将 LKUPRST 字段置 1，读取后可以写 1 清除此标志。

25.2 寄存器描述

表 25-1 EXTEN 相关寄存器列表

名称	访问地址	描述	复位值
R32_EXTEN_CTR	0x40023800	配置扩展控制寄存器	0x00002440

25.2.1 配置扩展控制寄存器 (EXTEN_CTR)

偏移地址：0x00

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

位	名称	访问	描述	复位值
[31:14]	Reserved	RO	保留。	0
[13:12]	LDOTRIM[1:0]	RW	调整数字内核电压值，LD0 电压值	10b
11	Reserved	RO	保留	0
[10:8]	ULLDOTRIM[2:0]	RW	调整低功耗模式下，ULLD0 电压值	100b
7	LKUPRST	RW1	LOCKUP 复位标志： 1：发生 LOCKUP 导致系统复位，写 1 清除； 0：正常。	0
6	LKUPEN	RW	LOCKUP 监测功能： 1：启用，系统发生 lock-up 时执行复位并将 LKUPRST 置位； 0：不启用。	1
5	Reserved	RO	保留。	0
4	HSIPRE	RW	HSI 时钟是否分频：（只能在 PLL 关闭下写入） 1：HSI 时钟作为 PLL 输入时钟； 0：HSI 时钟经 2 分频作为 PLL 输入时钟。	0
[3:0]	Reserved	RO	保留	0

第 26 章 调试支持 (DBG)

26.1 主要特征

此寄存器允许在调试状态下配置 MCU。包括：

- 支持独立看门狗 (IWDG) 的计数器
- 支持窗口看门狗 (WWDG) 的计数器
- 支持定时器的计数器
- 支持 I2CSMBus 的超时控制
- 支持 bxCAN 通信

26.2 寄存器描述

26.2.1 调试 MCU 配置寄存器 (DBGMCU_CR)

地址：0x7C0 (CSR)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved											CAN_S TOP	Reserved			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIM4_ STOP	TIM3_ STOP	TIM2_ STOP	TIM1_ STOP	I2C2_ SMBUS_ TIME OUT	I2C1_ SMBUS_ TIME OUT	WWDG_ STOP	IWDG_ STOP	Reserved					STAND BY	STOP	SLEEP

位	名称	访问	描述	复位值
[31:21]	Reserved	RO	保留	0
20	CAN_STOP	RW	CAN 调试停止位。当内核进入调试状态时 CAN1 停止运行。 1: CAN 的接收寄存器不继续接收数据; 0: CAN 仍然正常运行。	0
[19:16]	Reserved	RO	保留	0
15	TIM4_STOP	RW	定时器 4 调试停止位。当内核进入调试状态时计数器停止工作。 1: 定时器 4 的计数器停止工作; 0: 定时器 4 的计数器仍然正常工作。	0
14	TIM3_STOP	RW	定时器 3 调试停止位。当内核进入调试状态时计数器停止工作。 1: 定时器 3 的计数器停止工作; 0: 定时器 3 的计数器仍然正常工作。	0
13	TIM2_STOP	RW	定时器 2 调试停止位。当内核进入调试状态时计数器停止工作。 1: 定时器 2 的计数器停止工作; 0: 定时器 2 的计数器仍然正常工作。	0
12	TIM1_STOP	RW	定时器 1 调试停止位。当内核进入调试状态时计数器停止工作。	0

			1: 定时器 1 的计数器停止工作; 0: 定时器 1 的计数器仍然正常工作。	
11	I2C2_SMBUS_TIMEOUT	RW	SMBUS 超时模式调试停止位。当内核进入调试状态时停止 SMBUS 超时模式。 1: 冻结 SMBUS 的超时控制; 0: 与正常模式操作相同。	0
10	I2C1_SMBUS_TIMEOUT	RW	SMBUS 超时模式调试停止位。当内核进入调试状态时停止 SMBUS 超时模式。 1: 冻结 SMBUS 的超时控制; 0: 与正常模式操作相同。	0
9	WWDG_STOP	RW	窗口看门狗调试停止位。当内核进入调试状态时调试窗口看门狗停止工作。 1: 窗口看门狗计数器停止工作; 0: 窗口看门狗计数器仍然正常工作。	0
8	IWDG_STOP	RW	独立看门狗调试停止位。当内核进入调试状态时看门狗停止工作。 1: 看门狗计数器停止工作; 0: 看门狗计数器仍然正常工作。	0
[7:3]	Reserved	RW	保留	0
2	STANDBY	RW	调试待机模式位。 1: (FCLK 开, HCLK 开) 数字电路部分不下电, FCLK 和 HCLK 时钟由内部 RL 振荡器提供时钟。另外, 微控制器通过产生系统复位来退出 STANDBY 模式和复位是一样的; 0: (FCLK 关, HCLK 关) 整个数字电路部分都断电。从软件的观点看, 退出 STANDBY 模式与复位是一样的 (除了一些状态位指示了微控制器刚从 STANDBY 状态退出)。	0
1	STOP	RW	调试停止模式位。 1: (FCLK 开, HCLK 开) 在停止模式时, FCLK 和 HCLK 时钟由内部 RC 振荡器提供。当退出停止模式时, 软件必需重新配置时钟系统启动 PLL, 晶振等 (与配置此比特位为 0 时的操作一样); 0: (FCLK 关, HCLK 关) 在停止模式时, 时钟控制器禁止一切时钟 (包括 HCLK 和 FCLK)。当从 STOP 模式退出时, 时钟的配置和复位之后的配置一样 (微控制器由 8MHz 的内部 RC 振荡器 (HSI) 提供时钟)。因此, 软件必需重新配置时钟控制系统启动 PLL, 晶振等。	0
0	SLEEP	RW	调试睡眠模式位。 1: (FCLK 开, HCLK 开) 在睡眠模式时, FCLK 和 HCLK 时钟都由原先配置好的系统时钟提供; 0: (FCLK 开, HCLK 关) 在睡眠模式时, FCLK 由原先已配置好的系统时钟提供, HCLK 则关闭。由于睡眠模式不会复位已配置好的时钟系统, 因此从睡眠模式退出时, 软件不需要重新配置时钟系统。	0

第 27 章 电子签名 (ESIG)

电子签名包含了芯片识别信息：闪存区容量和唯一身份标识。它由厂家在出厂时烧录到存储器模块的系统存储区域，可以通过 SWD (SDI) 或者应用代码读取。

27.1 功能描述

闪存区容量：指示当前芯片用户应用程序可以使用大小。

唯一身份标识：96 位二进制码，对任意一个微控制器都是唯一的，用户只能读访问不能修改。此唯一标识信息可以用作微控制器（产品）的安全密码、加解密钥、产品序列号等，用来提高系统安全机制或表明身份信息。

以上内容用户都可以按 8/16/32 位进行读访问。

27.2 寄存器描述

表 27-1 ESIG 相关寄存器列表

名称	访问地址	描述	复位值
R16_ESIG_FLACAP	0x1FFFF7E0	闪存容量寄存器	0xFFFF
R32_ESIG_UNIID1	0x1FFFF7E8	UID 寄存器 1	0xFFFFFFFF
R32_ESIG_UNIID2	0x1FFFF7EC	UID 寄存器 2	0xFFFFFFFF
R32_ESIG_UNIID3	0x1FFFF7F0	UID 寄存器 3	0xFFFFFFFF

27.2.1 闪存容量寄存器 (ESIG_FLACAP)

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

位	名称	访问	描述	复位值
[15:0]	F_SIZE	RO	以 Kbyte 为单位的闪存容量。 例：0x0080 = 128 K 字节	X

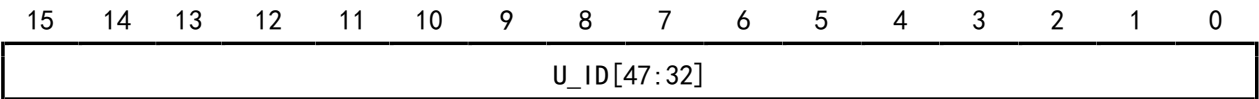
27.2.2 UID 寄存器 (ESIG_UNIID1)

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

位	名称	访问	描述	复位值
[31:0]	U_ID[31:0]	RO	UID 的 0-31 位。	X

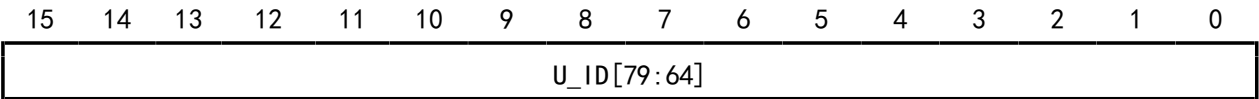
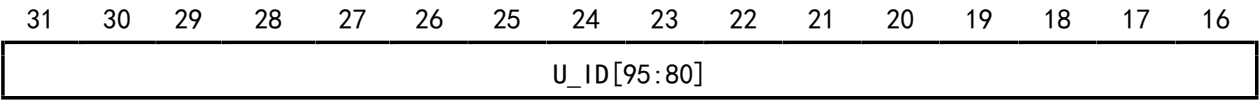
27.2.3 UID 寄存器 (ESIG_UNIID2)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
U_ID[63:48]															



位	名称	访问	描述	复位值
[31:0]	U_ID[63:32]	R0	UID 的 32-63 位。	X

27.2.4 UID 寄存器 (ESIG_UNIID3)



位	名称	访问	描述	复位值
[31:0]	U_ID[95:64]	R0	UID 的 64-95 位。	X