

用户手册

APM32F035x8

基于 Arm[®] Cortex[®]-M0+内核的 32 位微控制器

版本：V 0.3

目录

1	简介及文档描述规则	7
1.1	简介	7
1.2	文档描述规则.....	7
2	系统架构	10
2.1	术语全称、缩写描述.....	10
2.2	系统架构框图.....	10
2.3	存储器映射	11
2.4	启动配置	12
3	协处理器 (M0CP)	13
3.1	术语全称、缩写描述.....	13
3.2	简介	13
3.3	主要特征	13
3.4	结构框图	14
3.5	功能描述	14
3.6	寄存器地址映射	17
3.7	寄存器功能描述	17
4	FLASH 存储器	22
4.1	术语全称、缩写描述.....	22
4.2	主要特征	22
4.3	Flash 存储器结构.....	22
4.4	功能描述	23
4.5	寄存器地址映射	32
4.6	寄存器功能描述	33
5	系统配置控制器 (SYSCFG)	37
5.1	术语全称、缩写描述.....	37
5.2	寄存器地址映射	37
5.3	寄存器功能描述	37
6	复位与时钟管理 (RCM)	46
6.1	术语全称、缩写描述.....	46
6.2	复位功能描述.....	46
6.3	时钟管理功能描述	48

6.4	寄存器地址映射	54
6.5	寄存器功能描述	54
7	电源管理单元 (PMU)	70
7.1	术语全称、缩写描述	70
7.2	简介	70
7.3	结构框图	70
7.4	功能描述	71
7.5	寄存器地址映射	74
7.6	寄存器功能描述	74
8	嵌套向量中断控制器 (NVIC)	76
8.1	术语全称、缩写描述	76
8.2	简介	76
8.3	主要特征	76
8.4	中断和异常向量表	76
9	外部中断与事件控制器 (EINT)	78
9.1	简介	78
9.2	主要特征	78
9.3	功能描述	78
9.4	寄存器地址映射	80
9.5	寄存器功能描述	80
10	直接存储器存取 (DMA)	84
10.1	术语全称、缩写描述	84
10.2	简介	84
10.3	主要特征	84
10.4	功能描述	85
10.5	寄存器地址映射	90
10.6	寄存器功能描述	90
11	调试 MCU (DBGMCU)	94
11.1	术语全称、缩写描述	94
11.2	简介	94
11.3	主要特征	94
11.4	功能描述	95
11.5	寄存器地址映射	95

11.6	寄存器功能描述	95
12	通用/复用功能输入/输出引脚 (GPIO/AFIO)	99
12.1	术语全称、缩写描述	99
12.2	主要特征	99
12.3	结构框图	100
12.4	功能描述	100
12.5	寄存器地址映射	105
12.6	寄存器功能描述	105
13	定时器概述	110
13.1	术语全称、缩写描述	110
13.2	定时器类别及主要差异	110
14	高级定时器 (TMR1)	113
14.1	简介	113
14.2	主要特征	113
14.3	结构框图	114
14.4	功能描述	114
14.5	寄存器地址映射	132
14.6	寄存器功能描述	133
15	通用定时器 (TMR2/3/4)	158
15.1	简介	158
15.2	主要特征	158
15.3	结构框图	159
15.4	功能描述	159
15.5	寄存器地址映射	171
15.6	寄存器功能描述	172
16	基本定时器 (TMR6/7)	189
16.1	简介	189
16.2	主要特征	189
16.3	结构框图	189
16.4	功能描述	189
16.5	寄存器地址映射	191
16.6	寄存器功能描述	191
17	红外定时器 (IRTMR)	195

17.1	简介	195
17.2	功能描述	195
18	看门狗定时器 (WDT)	196
18.1	简介	196
18.2	独立看门狗	196
18.3	窗口看门狗	198
18.4	IWDT 寄存器地址映射	199
18.5	IWDT 寄存器功能描述	200
18.6	WWDT 寄存器地址映射	202
18.7	WWDT 寄存器功能描述	202
19	实时时钟 (RTC)	204
19.1	术语全称、缩写描述	204
19.2	简介	204
19.3	主要特征	204
19.4	结构框图	205
19.5	功能描述	205
19.6	寄存器地址映射	210
19.7	寄存器功能描述	211
20	控制器局域网 (CAN)	223
20.1	术语全称、缩写描述	223
20.2	简介	223
20.3	主要特征	223
20.4	功能描述	223
20.5	寄存器地址映射	231
20.6	寄存器功能描述	232
21	通用同步异步收发器 (USART)	247
21.1	术语全称、缩写描述	247
21.2	简介	247
21.3	主要特征	247
21.4	功能描述	249
21.5	寄存器地址映射	265
21.6	寄存器功能描述	265
22	内部集成电路接口 (I2C)	279

22.1	术语全称、缩写描述.....	279
22.2	简介	279
22.3	主要特征	279
22.4	结构框图	280
22.5	功能描述	281
22.6	寄存器地址映射	291
22.7	寄存器功能描述	291
23	串行外设接口/片上音频接口 (SPI/I2S)	302
23.1	术语全称、缩写描述.....	302
23.2	简介	302
23.3	SPI 主要特征	302
23.4	I2S 主要特征.....	303
23.5	SPI 功能描述	304
23.6	I2S 功能描述.....	315
23.7	寄存器地址映射	325
23.8	寄存器功能描述	325
24	模拟数字转换器 (ADC)	334
24.1	简介	334
24.2	主要特征	334
24.3	功能描述	335
24.4	寄存器地址映射	342
24.5	寄存器功能描述	342
25	比较器 (COMP)	355
25.1	术语全称、缩写描述.....	355
25.2	简介	355
25.3	主要特征	355
25.4	结构框图	356
25.5	功能描述	356
25.6	寄存器地址映射	357
25.7	寄存器功能描述	357
26	运算放大器 (OPA)	361
26.1	简介	361
26.2	主要特征	361

26.3	结构框图	361
26.4	寄存器地址映射	363
26.5	寄存器功能描述	363
27	循环冗余校验计算单元 (CRC)	365
27.1	简介	365
27.2	功能描述	365
27.3	寄存器地址映射	366
27.4	寄存器功能描述	366
28	芯片电子签名 (CHIP ELECTRONIC SIGNATURE)	368
28.1	简介	368
28.2	功能描述	368
28.3	寄存器功能描述	368
29	版本历史	370

1 简介及文档描述规则

1.1 简介

本参考手册向应用程序开发人员提供关于如何使用 MCU（微控制器）系统架构、存储器和外设所涉及的全部信息。

关于 Arm® Cortex®-M0+内核的相关信息，请参考 Arm® Cortex®-M0+技术参考手册；关于型号信息、尺寸和器件的电气特性等详细数据请参考对应的数据手册（Datasheet）；有关 MCU 系列全部型号中，存储器映射、外设存在情况及其数目请查阅相应的数据手册。

在此说明：珠海极海半导体有限公司，在以下正文中，简称“Geehy”。

1.2 文档描述规则

1.2.1 “寄存器功能描述”规则

- (1) 控制类（CTRL）寄存器，未特别说明，都是“软件置 1 和清 0”。
- (2) 控制类寄存器后面一般会有动词缩写以作区别，动词可以有：EN-Enable、CFG-Configure、D-Disable、SET-Setup、SEL-Select
- (3) 状态类寄存器缩写后面一般会有 FLG 以作区别。
- (4) 数值、数据类寄存器，一般会包括 V、VALUE、D、DATA，这些词后面不加动词，比如：xxPSC，CNT，后面一般不添加动词。

1.2.2 术语全称、缩写描述

表格 1 R/W 缩写及描述

R/W	描述	缩写
read/write	软件能读写此位。	R/W
read-only	软件只能读此位。	R
write-only	软件只能写此位，读此位将返回复位值。	W
read/clear	软件可以读此位，也可以通过写 1 清除此位，写 0 对此位无影响。	RC_W1
read/clear	软件可以读此位，也可以通过写 0 清除此位，写 1 对此位无影响。	RC_W0
read/clear by read	软件可以读此位，读此位将自动地清除它为 0，写此位无效。	RC_R
read/set	软件可以读也可以设置此位，写 0 对此位无影响。	R/S
read-only write trigger	软件可以读此位，写 0 或 1 触发一个事件但对此位数值没有影响。	RT_W
toggle	软件只能通过写 1 来翻转此位，写 0 对此位无影响。	T

表格 2 常用寄存器功能描述术语全称、缩写

中文全称	英文全称	英文缩写
使能	Enable	EN
禁止	Disable	D
清除	Clear	CLR
选择	Select	SEL
配置	Configure	CFG
控制	Contrl	CTRL
控制器	Controller	C
复位	Reset	RST
停止	Stop	STOP
设置	Set	SET
装载	Load	LD
校准	Calibration	CAL
初始化	Initialize	INIT
错误	Error	ERR
状态	Status	STS
准备	Ready	RDY
软件	Software	SW
硬件	Hardware	HW
源	Source	SRC
系统	System	SYS
外设	Peripheral	PER
地址	Address	ADDR
方向	Direction	DIR
时钟	Clock	CLK
输入	Input	I
输出	Output	O
中断	Interrupt	INT
数据	Data	DATA
大小	Size	SIZE
除法器	Divider	DIV
预分频器	Prescaler	PSC
乘法器	Multiplier	MUL

中文全称	英文全称	英文缩写
周期	Period	PRD
加扰/加密	Cipher	CIPH

表格 3 模块全称、简写

中文全称	英文全称	英文缩写
复位与时钟管理	Reset and Clock Management	RCM
电源管理单元	Power Management Unit	PMU
嵌套向量中断控制器	Nested Vector Interrupt Controller	NVIC
外部中断/事件控制器	External Interrupt /Event Controller	EINT
直接存储器存取	Direct Memory Access	DMA
调试 MCU	Debug MCU	DBG MCU
通用输入/输出引脚	General-Purpose Input Output Pin	GPIO
复用功能输入/输出引脚	Alternate Function Input Output Pin	AFIO
定时器	Timer	TMR
看门狗定时器	Watchdog Timer	WDT
独立看门狗	Independent Watchdog Timer	IWDT
窗口看门狗	Windows Watchdog Timer	WWDT
实时时钟	Real-Time Clock	RTC
通用同步异步收发器	Universal Synchronous Asynchronous Receiver Transmitter	USART
内部集成电路接口	Inter-integrated Circuit Interface	I2C
串行外设接口	Serial Peripheral Interface	SPI
片上音频接口	Inter-IC Sound Interface	I2S
模拟数字转换器	Analog-to-Digital Converter	ADC
循环冗余校验计算单元	Cyclic Redundancy Check Calculation Unit	CRC

2 系统架构

2.1 术语全称、缩写描述

表格 4 术语全称、缩写描述

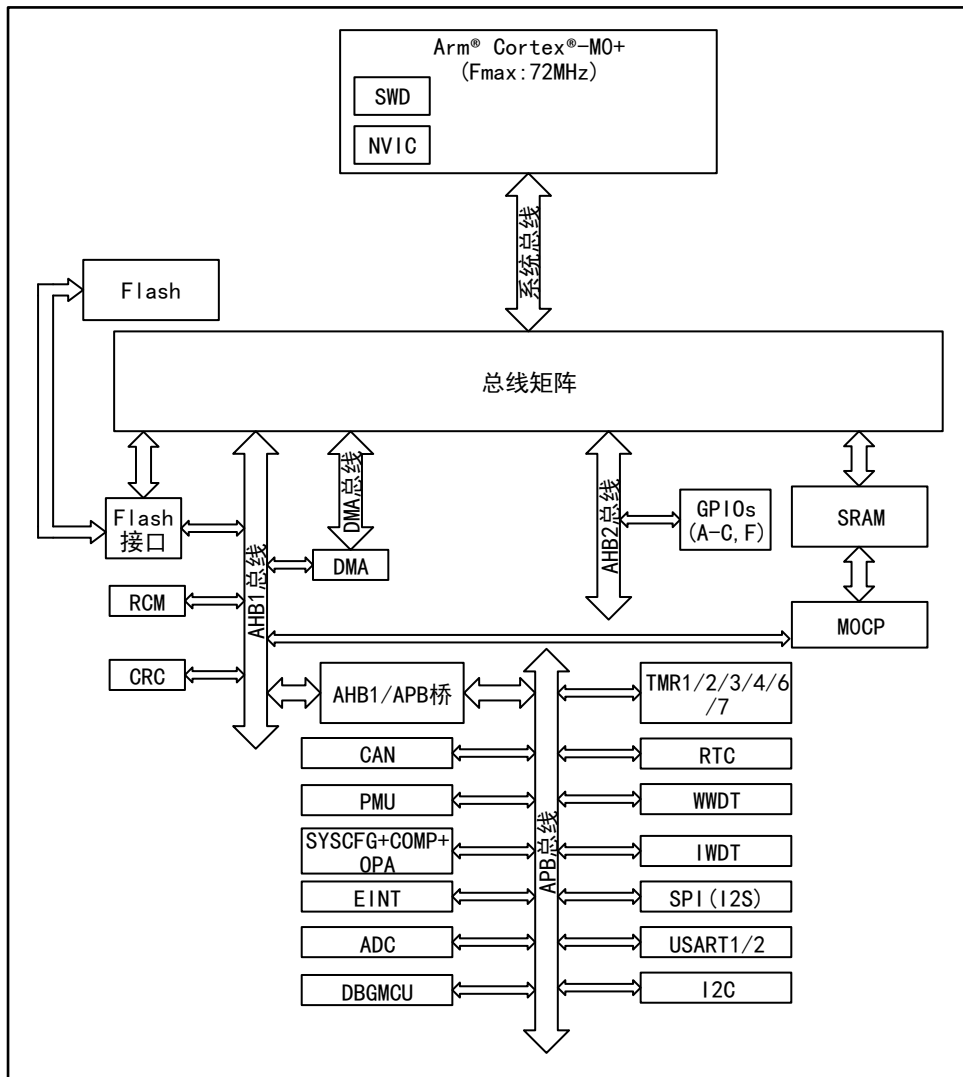
中文全称	英文全称	英文缩写
高级高性能总线	Advanced High-Performance Bus	AHB
高级外围总线	Advanced Peripheral Bus	APB

2.2 系统架构框图

主系统主要由三个主模块和四个从模块构成。主模块分别是 Arm® Cortex®-M0+ 内核、通用 DMA 和协处理器 M0CP。从模块分别是内部 SRAM、内部闪存存储器 FLASH、连接所有 GPIO 口的 AHB2 总线和在 AHB1 总线上的 AHB1/APB 桥，其中，AHB1/APB 桥连接所有外设设备。

这些都是通过一个多级的 AHB 总线构架相互连接的，如下图所示：

图 1 APM32F035x8T7 系统架构框图



表格 5 总线名称

名称	说明
系统总线	连接 Arm® Cortex®-M0+内核的系统总线（外设总线）与总线矩阵。
DMA 总线	连接 DMA 的 AHB 主控接口与总线矩阵。
总线矩阵	协调内核和 DMA 的访问；由 CPU AHB、系统总线、DMA 总线和 FMC、SRAM、AHB2 和 AHB1/APB 桥组成。AHB 外设通过总线矩阵与系统总线相连，允许 DMA 访问。
AHB/APB 桥	桥在 AHB 和 APB 总线间提供同步连接。 当对 APB 寄存器进行非 32 位访问时，访问会被自动转换成 32 位。

2.3 存储器映射

存储器映射地址总共是 4GB 地址，分配的地址包括内核（包括内核外设）、片上 Flash（包括主存储区、系统存储区、选项字节）、片上 SRAM、总线外设（包括 AHB、APB 外设），各类地址具体信息请参考对应型号的数据手册。

2.3.1 嵌入式 SRAM

内置静态 SRAM。它可以以字节、半字（16 位）或全字（32 位）访问。SRAM 的起始地址是 0x2000 0000。

SRAM 不支持奇偶校验。

2.4 启动配置

APM32F035 微控制器系列实现了一个特殊的机制，通过配置 BOOT0 引脚参数和 FMC_OBCS 中的 nBOOT1 位，可以用有三种不同的启动模式，即系统可以不仅仅从 Flash 存储器或系统存储器启动，还可以从内置 SRAM 启动。被选作启动区域的存储器是由选择的启动模式决定的。

表格 6 启动模式配置及其访问方式

启动模式选择引脚		启动模式	访问方式
BOOT1	BOOT0		
X	0	主闪存存储器 (Flash)	主闪存存储器被映射到启动空间，但仍然能够在它原有的地址访问它，即闪存存储器的内容可以在两个地址区域访问。
0	1	系统存储器	系统存储器被映射到启动空间 (0x0000 0000)，但仍然能够在它原有的地址访问它。
1	1	内置 SRAM	只能在开始的地址区访问 SRAM。

注意：

- (1) 启动空间地址为 0x0000 0000
- (2) Flash 原有地址为 0x0800 0000
- (3) 系统存储器原有地址为 0x1FFF EC00
- (4) SRAM 的起始地址为 0x2000 0000
- (5) BOOT1 的值为 nBOOT1 选项位取反

用户可以通过设置 BOOT1（配置 nBOOT1）和 BOOT0 引脚的状态，来选择在复位后的启动模式。

BOOT 引脚应在待机模式下保持用户需要的启动配置，当从待机模式退出时，引脚的值会被锁存。

如果选择从内置 SRAM 启动，那么在编写应用代码时，必须使用 NVIC 的异常表和偏移寄存器，重新将向量表映射至 SRAM 中。

内嵌的启动程序

内嵌的启动程序由 Geehy 在生产线上写入并存放在系统存储区域。

3 协处理器 (M0CP)

3.1 术语全称、缩写描述

表格 7 术语全称、缩写描述

中文全称	英文全称	术语
软件/硬件/固件	Software/Hardware/Firmware	SW/HW/FW
接口	Interface	I/F
协处理器	Coprocessor for the Arm Cortex-M0+	M0CP
程序存储器或指令存储器	Program memory or instruction memory	PM
数据存储器	Data memory	DM
用于指令或程序存储器访问的主总线接口	Master bus I/F for instruction or program memory accessing	IBUS
用于数据存储器访问的主总线接口	Master bus I/F for data memory accessing	DBUS
AHB 从接口	AHB slave interface	ahbs
面向硬件的预综合用户定义	Hardware-oriented, pre-synthesis user-defined	configurable
面向软件的, 通过软件接口的综合后用户编程	Software-oriented, post-synthesis user-programmed through SW I/F	programmable

3.2 简介

协处理器包含硬件除法、硬件开方、三角函数生成、SVPWM 生成五段和七段式功能。

3.3 主要特征

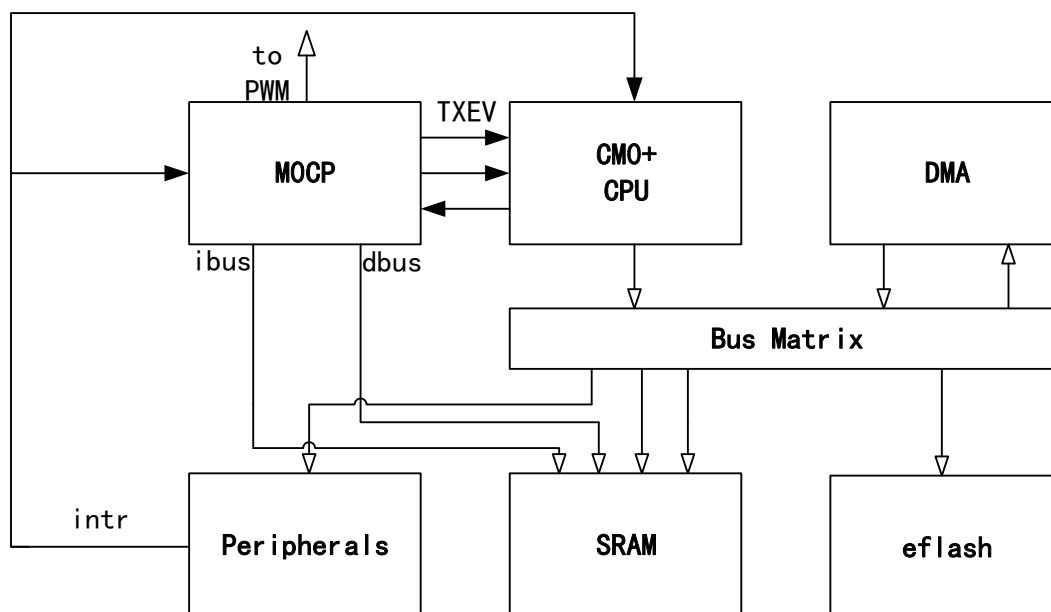
- (1) 1 个可配置 RV32ECM 内核。PM 最大为 8 KB。
- (2) 1 个 32 位 AHB 从接口用于寄存器配置。仅支持按 32 位数据对齐据访问。
- (3) 1 个 32 位 AHB 主接口用于访问 PM 和 DM。其中对 IBUS 主接口的响应必须为零等待。
- (4) 1 个紧耦合的接口用于访问与 SVPWM 算法对应的 PWM 模块。
- (5) 1 个 TX 中断 (计算结束产生中断)。
- (6) 两组参数/结果寄存器, 寄存器组的切换方式可编程, 由硬件或软件实现。
- (7) 支持 Arm® Cortex®-M0+和 TXEV 接口通信。

- (8) 可编程数据格式，Q0 至 Q15。
- (9) 支持并优化了 CORDIC 算法，实现部分三角函数加速。支持旋转模式和矢量模式。
- (10) 支持并优化 SVPWM 算法，支持五段式和七段式。
- (11) 支持和优化定点平方根和除法（32/32 位）。
- (12) 支持 atan2 运算。

3.4 结构框图

协处理器(M0CP) 旨在加速一些运行在 Arm® Cortex®-M0+上，与 FOC（Field Oriented Control，磁场定向控制）相关的应用。M0CP 与 MCU 的典型集成如下图所示。

图 2 结构框图



M0CP 采用两个 32 位 AHB lite 主接口分别访问 PM 和 DM。M0CP 可通过一个 32 位 AHB 从接口进行配置，通过 TXEV 与处理器通信，通过紧耦合握手协议接口直接更新 PWM 模块所需数据。

3.5 功能描述

3.5.1 CORDIC 算法描述

CORDIC (coordinate rotation digital computer, 坐标旋转数字算法) 是一种硬件高效的迭代方法，通过基本的加法和移位运算，实现矢量的旋转和定向计算不再需要复杂的函数就能得出结果。它使用旋转计算范围广泛的初等函数，同时支持旋转模式和向量模式，如下表所示。

表格 8 旋转模式和向量模式

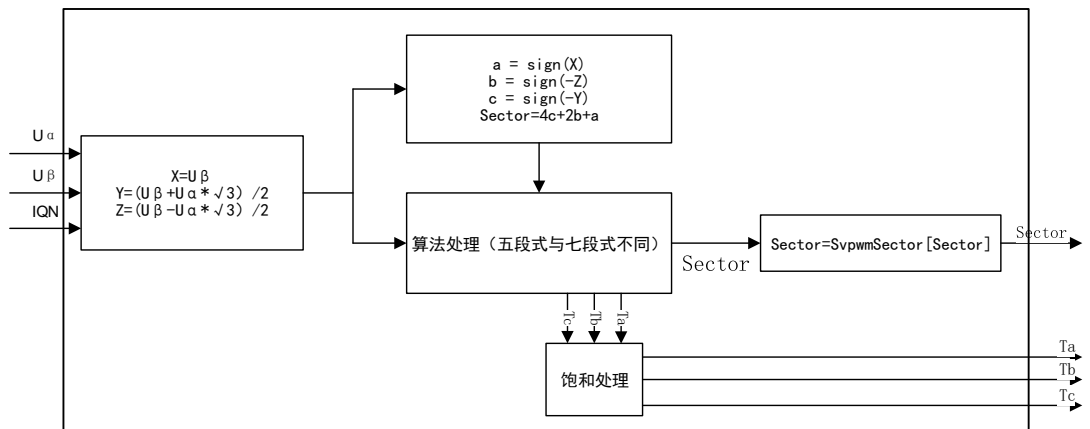
圆函数: $e_i = \arctan(2^{-i})$			
旋转模式	$d_i = \text{sign}(z_i), z_i \rightarrow 0$	$X_{result} = X \cos(Z) - Y \sin(Z)$ $Y_{result} = X \sin(Z) + Y \cos(Z)$ $Z_{result} = 0$	设置 $X=1, Y=0$ 则 $X_{result} = \cos(Z), Y_{result} = \sin(Z)$ CORDIC 内部采用预处理逻辑, 所以 X, Y 和 Z 可全范围 $[-2^{15}, (2^{15} - 1)]$ 取值
向量模式	$d_i = \text{sign}(z_i), z_i \rightarrow 0$	$X_{result} = \sqrt{X^2 + Y^2}$ $Y_{result} = 0$ $Z_{result} = Z + \arctan\left(\frac{Y}{X}\right)$	设置 $X=X, Y=Y,$ 则向量幅值 $X_{result} = \sqrt{X^2 + Y^2}$ 设置 $Z=0,$ 则 $Z_{result} = Z + \arctan\left(\frac{Y}{X}\right)$ CORDIC 内部采用预处理和后处理逻辑, 所以 X, Y 可全范围 $[-2^{15}, (2^{15} - 1)]$ 取值

注意: 操作数/结果 X, Y, Z 的数据存储在 $X_REG/Y_REG/Z_REG$ 寄存器中。

3.5.2 SVPWM 算法描述

SVPWM (Space Vector Pulse Width Modulation, 空间矢量脉宽调制) 用于调制电机所需要的三相电压。5 段和 7 段的算法流程如下图所示。

图 3 5 段和 7 段的算法流程



五段式算法处理
Sector=0, 7: $T_a=T_b=T_c=IQN(0.5)$;

Sector=1: $t_1=Z; t_2=Y;$
 $T_c=0;$
 $T_a=T_c+t_2;$
 $T_b=T_a+t_1;$

Sector=2: $t_1=Y; t_2=-X;$
 $T_b=0;$
 $T_c=T_b+t_2;$
 $T_a=T_c+t_1;$

Sector=3: $t_1=-Z; t_2=X;$
 $T_b=0;$
 $T_c=T_b+t_2;$
 $T_a=T_b+t_1;$

Sector=4: $t_1=-X; t_2=Z;$
 $T_a=0;$
 $T_b=T_a+t_2;$
 $T_c=T_b+t_1;$

Sector=5: $t_1=X; t_2=-Y;$
 $T_a=0;$
 $T_c=T_a+t_2;$
 $T_b=T_c+t_1;$

Sector=6: $t_1=-Y; t_2=-Z;$
 $T_b=0;$
 $T_a=T_b+t_2;$
 $T_c=T_a+t_1;$

七段式算法处理
Sector=0, 7: $T_a=T_b=T_c=IQN(0.5)$;

Sector=1: $t_1=Z; t_2=Y;$
 $T_c=(IQN(1)-t_1-t_2)>>1;$
 $T_a=T_c+t_2;$
 $T_b=T_a+t_1;$

Sector=2: $t_1=Y; t_2=-X;$
 $T_b=(IQN(1)-t_1-t_2)>>1;$
 $T_c=T_b+t_2;$
 $T_a=T_c+t_1;$

Sector=3: $t_1=-Z; t_2=X;$
 $T_b=(IQN(1)-t_1-t_2)>>1;$
 $T_c=T_b+t_2;$
 $T_a=T_b+t_1;$

Sector=4: $t_1=-X; t_2=Z;$
 $T_a=(IQN(1)-t_1-t_2)>>1;$
 $T_b=T_a+t_2;$
 $T_c=T_b+t_1;$

Sector=5: $t_1=X; t_2=-Y;$
 $T_a=(IQN(1)-t_1-t_2)>>1;$
 $T_c=T_a+t_2;$
 $T_b=T_c+t_1;$

Sector=6: $t_1=-Y; t_2=-Z;$
 $T_b=(IQN(1)-t_1-t_2)>>1;$
 $T_a=T_b+t_2;$
 $T_c=T_a+t_1;$

说明:
IQN: Qn的n值, 8位无符号数字
16位有符号数字

乘法: Qn格式乘法
SvpwmSector[]: 数组 [0, 1, 5, 0, 3, 2, 4, 0]

饱和处理: $x=a, b, c$
 $T_x < 0$ 时, $T_x=0$;
 $T_x > IQN(1)$ 时, $T_x=IQN(1)$

Sign函数: $x=\text{sign}(y)$
 $y > 0$ 时, $x=1$;
 $y < 0$ 时, $x=0$

注意：SVPWM 的 COUNT 模式需要将输出的占空比乘以 TMR1 的周期值，所以在不使能 TMR1 的情况下 M0CP 会一直处于 busy 状态。所以在 COUNT 模式下，应使能 TMR1。

注意：操作数/结果 X, Y, Z, W (Sector) 的数据存储在 X_REG/Y_REG/Z_REG/W_REG 寄存器中。

3.5.3 除法算法描述

进行 32bit/32bit 除法时，寄存器存放数据的情况如下：

- 除数应放在 X_REG 寄存器
- 被除数应放在 Y_REG 寄存器
- (计算完成后) 商 (32-bit) 存放在 X_REG 寄存器
- (计算完成后) 余数存放在 Y_REG 寄存器

通过配置 CTRL_REG 寄存器的 ALG 位选择有符号或无符号除法。

对于有符号除法运算，取余数规则描述如下：余数的符号与被除数相同，即

$$\text{余数} = \text{被除数} - \text{商} \times \text{除数}$$

例子 1: $-20 \div 6$ ，商为-3，余数为-2，而不是商为-4，余数为+4；

例子 2: $20 \div (-3)$ ，商为-7，余数为 1。

配置进来的输入的数据类型和读取出的结果的数据类型应和 ALG 的值选择对应；即计算有符号除法时，输入、输出为 int32 类型。计算无符号除法时，输入、输出为 uint32 类型。

对除法操作启动之前的除数和被除数的配置没有顺序要求。启动除法后，如果硬件运算完成，会自动将 STAT_REG 的 BUSY 位清 0。此时，可以从相应寄存器中读取运算结果，即商和余数。

3.5.4 I/O 寄存器分配和周期计数

表格 9 I/O 寄存器分配和周期计数表

算法	输入				输出				周期	执行时间 (72MHz)
	X_REG	Y_REG	Z_REG	Qn	X_REG	Y_REG	Z_REG	W_REG		
CORDIC rotate	X 输入 (int16)	Y 输入 (int16)	Z 输入 (int16)	15	X 输出 (int16)	Y 输出 (int16)	-	-	23	0.32us
CORDIC vector	X 输入 (int16)	Y 输入 (int16)	Z 输入 (int16)	n (1~15)	-	-	Z 输出 (int16)	-	n+3	0..28us
SVPWM duty	U_{β} (int16)	(0xddb3) (15-Qn) $*U_{\alpha}$ (int32)	-		T_a (uint16)	T_b (uint16)	T_c (uint16)	sector (uint3)	6	0.08us
SVPWM counter	U_{β} (int16)	(0xddb3) (15-Qn) $*U_{\alpha}$ (int32)	-		C_a (uint16)	C_b (uint16)	C_c (uint16)	sector (uint3)	≥ 12	$\geq 0.17us$

算法	输入				输出				周期	执行时间 (72MHz)
	X_REG	Y_REG	Z_REG	Qn	X_REG	Y_REG	Z_REG	W_REG		
Square Root	输入 (uint32)	-	-	-	输出 (uint16)	-	-	-	12	0.17us
DIV	除数 (uint/int32)	被除数 (uint/int32)	-	-	商 (uint32/int32)	余数 (uint32/int32)	-	-	10	0.14us
ATAN2	X 输入 (int16)	Y 输入 (int16)	Z 输入 (int16) (应为 0)	15	-	-	Z 输出 (int16)	-	26	0.36us

循环数从计算开始 (CTRL_REG.RUN=1) 开始计数, 当结果写入寄存器 (STAT_REG.BUSY=0) 时结束。在算法 SVPWM 计数器中, M0CP 需要通过有效的就绪协议接口更新 PWM。因此, 周期的数量根据 PWM 的响应时间而变化。

3.6 寄存器地址映射

仅支持按 32 位数据进行访问。

表格 10 协处理器寄存器地址映射

寄存器名	描述	地址偏移
CTRL_REG	控制寄存器	0x00
STAT_REG	状态寄存器	0x04
TXIS_REG	对外中断状态寄存器	0x08
TXIR_REG	原始操作状态寄存器	0x0C
TXIE_REG	对外中断使能寄存器	0x10
X_REG	X 变量寄存器	0x14
Y_REG	Y 变量寄存器	0x18
Z_REG	Z 变量寄存器	0x1C
W_REG	W 参数寄存器, 仅用于 SVPWM sector 模式	0x20
BANK_REG	寄存器组选择控制寄存器	0x34
REV_REG	修订或 IP 标签寄存器	0x3C
INIT_REG	初始化寄存器	0x40

3.7 寄存器功能描述

3.7.1 控制寄存器 (CTRL_REG)

偏移地址: 0x00

复位值: 0x0000 0078

位/域	名称	R/W	描述
0	RUN	W	开始 (Run) 操作开始, 由硬件清除, 软件设置。 写入 1 以启动运算, 下一个周期, 由硬件自动清除。
1	BANKAUTO	R/W	自动转换 (Bank Auto Conversion) 寄存器组转换是否是自动的。 0: 通过软件成组交换 1: 通过硬件成组交换
2	BANKHSEL	R/W	内核计算时的寄存器组选择 (Bank Select) 0: 使用寄存器组 0 1: 使用寄存器组 1 当 BANKAUTO 配置为 1 时, 该字段为只读, 表示该位由硬件管理。 当 BANKAUTO 配置为 0 时, 该字段可读可写。
6:3	QN	R/W	定点化信息设定 (Fixed-point information setting) 15:Q15 格式 14:Q14 \格式 0: 不做定点化 操作 。
7	CORDICOUT	R/W	CORDIC 输出 (CORDIC Output) 专用于 CORDIC 的 XY 输出模式。 0: 对于最终输出, X 和 Y 是原始结果。 1: 对于最终输出, X 和 Y 是原始结果除以 2.
8	SVPWMSEG	R/W	SVPWM 分式 (SVPWM Segmentation) 0: SVPWM 7 段式 1: SVPWM 5 段式
9	SVPWMOUT	R/W	SVPWM 输出 (SVPWM Output) 0: SVPWM 输出占空比结果。 1: SVPWM 输出 PWM 数值结果 (周期*占空比)。
10	TXEVEN	R/W	TX 中断使能 (TX Interrupt Enable) 0: 不会发送任何 TX 中断。 1: 操作完成后发送一个 TX 中断。
15:11	保留		
19:16	ALG	R/W	运算模式. (Algebra) 0x1: SVPWM. 0x2: CORDIC 旋转模式 0x3: CORDIC 向量模式 0x4: 平方根. 0x5: 除法, 无符号 0x6: 除法, 有符号. 0x7: ATAN2(复用上 CORDIC 向量模式硬件). 其他: 保留
31:20	保留		

3.7.2 状态寄存器 (STAT_REG)

偏移地址: 0x04

复位值: 0x00000000

位/域	名称	R/W	描述
0	BUSY	R	忙 (Busy) 内核是否繁忙 (计算或初始化时处于繁忙状态)。
1	DIVERR	R	除数 (Division) 是否出现除数为 0 的情况, 下一个操作开始时会被硬件清除。
2	OVF	R	溢出 (Overflow) 是否发生数据溢出, 下一个操作开始时会被硬件清除。
31:3	保留		

3.7.3 对外中断状态寄存器 (TXIS_REG)

偏移地址: 0x08

复位值: 0x00000000

位/域	名称	R/W	描述
0	DONE	R	完成状态 (Done Flag) 给到主 CPU 的 TX 中断状态位。当 TX 中断启用且操作完成时, 这一比特会被置位。 0: TX 中断被禁用或操作未完成(或者 m0cp 上没有运行任何操作)。 1: TX 中断已启用, 操作已完成。
31:1	保留		

3.7.4 原始中断状态寄存器 (TXIR_REG)

偏移地址: 0x0C

复位值: 0x00000000

位/域	名称	R/W	描述
0	DONE	RC_W1	完成状态 (Done Flag) 原始中断信号。在操作完成时会被置位, 与 TX 中断的使能与否无关。向该寄存器写入 1 以清 0。 0: 没有操作或操作未完成。 1: 操作完成。
31:1	保留		

3.7.5 对外中断使能寄存器 (TXIE_REG)

偏移地址: 0x10

复位值: 0x00000000

位/域	名称	R/W	描述
0	TXIEN	R/W	使能 TX 中断 (TX Interrupt Enable) 0: 禁用 TX 中断。当操作完成时, 不会拉高任何中断信号。 1: 使能 TX 中断。当操作完成时, 将拉高给主 CPU 的中断信号。
31:1	保留		

注意: 在启用 TXIE_REG 之前, 软件必须注意 TXIR_REG, 以免发生意外中断。

3.7.6 操作数/结果 X 寄存器 (X_REG)

偏移地址: 0x14

复位值: 0x00000000

位/域	名称	R/W	描述
31:0	DATA	R/W	参数 X 和结果 X 的数据 (Data X)

3.7.7 操作数/结果 Y 寄存器 (Y_REG)

偏移地址: 0x18

复位值: 0x00000000

位/域	名称	R/W	描述
31:0	DATA	R/W	参数 Y 和结果 Y 的数据 (Data Y)

3.7.8 操作数/结果 Z 寄存器 (Z_REG)

偏移地址: 0x1C

复位值: 0x00000000

位/域	名称	R/W	描述
31:0	DATA	R/W	参数 Z 和结果 Z 的数据 (Data Z)

3.7.9 结果 W 寄存器 (W_REG)

偏移地址: 0x20

复位值: 0x00000000

位/域	名称	R/W	描述
31:0	DATA	R	仅用于存储结果 W 的数据 (Data W)

注意: 由于 X/Y/Z/W 寄存器都是 32 位的, 当参数位宽小于 32 位时, 应该在正确的符号扩展后再配置进来。

例如, 在 CORDIC 中, 参数 X 的类型为 int16。当 X 的值为 -1 (十进制) 时, 配置到 X_REG 的值应该是 0xFFFF_FFFF, 而不是 0x0000_FFFF, 这一过程可以通过定义输入为 int32 数据类型, 由编译器完成。

当用作结果寄存器时, X/Y/Z/W 寄存器中的所有值都会根据符号值扩展为 32 位, 这一操作不会影响小于 32 位数据的正确性。

3.7.10 寄存器组寄存器 (BANK_REG)

偏移地址: 0x34

复位值: 0x00000000

当 CTRL_REG.BANKAUTO 配置为 1 时, 该寄存器仅读, 表示由硬件管理这些位。当 CTRL_REG.BANKAUTO 配置为 0 时, 该字段为可读可写。

位/域	名称	R/W	描述
0	BANKWSEL	R/W	选择写入的寄存器组 (Writing Bank Select) 0: 选择组 0 1: 选择组 1
1	BANKRSEL	R/W	选择读取的寄存器组 (Reading Bank Select) 0: 选择组 0 1: 选择组 1
31:2	保留		

3.7.11 版本寄存器 (REV_REG)

偏移地址: 0x3C

复位值：0x00050100

位/域	名称	R/W	描述
7:0	MIN	R	小型修订 (Minimum Reversion) 表示非 RTL 变更 (文档修改等)
15:8	MID	R	中期修订 (Medium Reversion) 表示固件不可见的 RTL 更改
23:16	MAJ	R	重大修订 (Major Reversion) 表示固件可见的 RTL 更改
31:24	保留		

3.7.12 初始化寄存器 (INIT_REG)

偏移地址：0x40

复位值：0x2000 0000

位/域	名称	R/W	描述
28:0	保留		
29	INIT	R/W	初始化 M0CP (M0CP Initialize) 用于初始化 M0CP, 先写 1 再写 0
31:30	保留		

4 Flash 存储器

本章主要介绍 Flash 的存储结构、读、擦、写、读/写保护、解锁/锁定特性，及涉及到的寄存器功能描述。

4.1 术语全称、缩写描述

表格 11 术语全称、缩写描述

中文全称	英文全称	英文缩写
闪存存储器控制器	Flash Memory Controller	FMC

4.2 主要特征

(1) Flash 存储器结构

- 分为主存储区、信息块
- 主存储区容量最高为 64KB
- 信息块分为系统存储区、OTP 区域、选项字节区域
- 系统存储区存放 BootLoader 程序，容量大小为 2944Bytes
- OTP 区域存放单次配置选项字，容量大小为 28Bytes，该区域用户只能写一次
- 芯片配置区包含 96 位唯一 UID、主存储区容量信息等，容量大小为 100Bytes
- 选项字节区容量大小为 64Bytes

(2) 功能说明

- 读取 Flash
- 页/片擦 Flash
- 写 Flash
- 读/写保护 Flash
- 配置选项字节
- Flash 存储内容加密

4.3 Flash 存储器结构

表格 12 APM32F035x8T7 型号产品 Flash 存储结构

块	名称	地址区域	大小 (字节)	扇区
主存储区	页 0	0x0800 0000 - 0x0800 03FF	1K	扇 0
主存储区	页 1	0x0800 0400 - 0x0800 07FF	1K	
主存储区	页 2	0x0800 0800 - 0x0800 0BFF	1K	
主存储区	页 3	0x0800 0C00 - 0x0800 0FFF	1K	
主存储区	

块	名称	地址区域	大小 (字节)	扇区
主存储区	页 28	0x0800 7000 - 0x0800 73FF	1K	扇 7
主存储区	页 29	0x0800 7400 - 0x0800 77FF	1K	
主存储区	页 30	0x0800 7800 - 0x0800 7BFF	1K	
主存储区	页 31	0x0800 7C00 - 0x0800 7FFF	1K	
主存储区	
主存储区	页 60	0x0800 F000 - 0x0800 F3FF	1K	扇 15
主存储区	页 61	0x0800 4000 - 0x0800 F7FF	1K	
主存储区	页 62	0x0800 8000 - 0x0800 FBFF	1K	
主存储区	页 63	0x0800 C000 - 0x0800 FFFF	1K	
信息块	系统存储区	0x1FFF EC00-0x1FFF F77F	2944	-
信息块	OTP 区域	0x1FFF F780-0x1FFF F79B	28	-
信息块	芯片配置区	0x1FFF F79C-0x1FFF F7FF	100	-
信息块	选项字节	0x1FFF F800-0x1FFF F83F	64	-

注:

- (1) OTP 区域被配置后无法擦除。

4.4 功能描述

描述对主存储器、信息块（包括系统存储区、选项字节）的操作，包括读、写、擦、读/写保护。

读取 Flash 包括主存储块、信息块，擦、写、读/写包括分开介绍；系统存储区在出厂时已写入，用户无法修改，不介绍该模块的擦、写、读/写保护。

4.4.1 读取 Flash

Flash 存储器可直接寻址，读取 Flash 受以下配置影响：

等待周期

针对不同的系统时钟需配置不同的等待周期：

- 0 个等待周期：0<系统时钟≤24MHz
- 1 个等待周期：24MHz<系统时钟≤48MHz
- 2 个等待周期：48MHz<系统时钟≤72MHz

预取缓存区

可提高读取速度，每次复位预取缓存区被自动打开；带预取缓存的读接口；只有系统时钟和 AHB 时钟一致、系统时钟小于 24MHz 时才能配置，系统时钟和 AHB 时钟一致时才能使用。

4.4.2 主存储块

4.4.2.1 擦主存储块

FMC 支持页擦除和整片擦除（全擦除）两种方式将主存储区的内容初始化为高电平（数据表现为 0xFFFF）。用户在对 Flash 进行写入前，建议对写入地址页先进行擦除操作，若写入地址的数据不是 0xFFFF 则会触发一个编程错误。

主存储器页擦除

页擦除是根据程序选择的主存储区页进行独立擦除，不会对未选择擦除的页产生任何影响。

正确的页擦除（或闪存写入操作）结束后，FMC_STS 寄存器的 OCF 将会被置位，若开启了 OCIE 中断则将触发一个操作完成中断。用户需要注意的是选择擦除的页必须是有效的页（主存储区有效的地址和未被写保护的地址）。

页擦除流程为：

- （1）检查 FMC_STS 寄存器的 BUSYF 为 0，表明此时没有其他操作正在进行。
- （2）设置 FMC_CTRL2 寄存器的 PAGEERA。
- （3）把需要擦除的页的地址写入到 FMC_ADDR 寄存器中。
- （4）设置 FMC_CTRL2 寄存器的 STA。
- （5）等待 FMC_STS 寄存器的 BUSYF 变为 0。
- （6）检查 FMC_STS 寄存器的 OCF 标志。
- （7）清除 FMC_STS 寄存器的 OCF 标志。

主存储器片擦除

片擦除操作将会使 Flash 主存储区内所有内容全部被擦除，片擦除会使主存储区所有数据全部擦除，用户在使用时需要特别注意，以避免误操作导致重要数据丢失。

片擦除流程为：

- （1）检查 FMC_STS 寄存器的 BUSYF 为 0，表明此时没有其他操作正在进行。
- （2）设置 FMC_CTRL2 寄存器的 MASSERA。
- （3）设置 FMC_CTRL2 寄存器的 STA。
- （4）等待 FMC_STS 寄存器的 BUSYF 变为 0。
- （5）检查 FMC_STS 寄存器的 OCF 标志。
- （6）清除 FMC_STS 寄存器的 OCF 标志。

4.4.2.2 写主存储块

FMC 支持对主存储区的 16 位（半字）数据写入，可选择通过 Debug、BootLoader、程序在 SRAM 中运行、直接读取已擦除的页等方式判断是否擦除成功。

为保证写入正确，需要在写入前检查目的地址是否已经被擦除，若未被擦除，则写入数据无效并将 FMC_STS 寄存器的 PEF 位置“1”。若目的地址存在写保护，则写入数据无效并触发一个写保护错误（FMC_STS 的 WPEF 位置“1”）。

4.4.2.3 读/写保护主存储块

闪存读/写保护用以防止非法读出/修改主存储区代码或数据，它由选项字节的读/写保护配置字节控制。对于 APM32F035x8T7 型号产品，写保护的基本单位是 4 页（即 4KBytes）。

读保护

读保护有三个级别，分布是：等级 0、等级 1、等级 2，具体情况如下：

表格 13 读保护级别区别

类别	READPROT	描述
等级 0	0xAA	主存储区、选项字节可擦、写、读。
等级 1	除了 0xAA 和 0xCC 的其它值	用户模式：允许对主存储区、选项字节擦、写、读。 Debug、SRAM 运行、系统储存区运行：禁止访问主存储区；选项字节可擦、写、读，但当等级修改为 0 时，会先执行主存储区片擦。
等级 2	0xCC	用户模式：允许对主存储区进行擦、写和读，允许对选项字节读和写（READPROT 除外，不能写），不可擦除选项字节 无法 Debug，系统不能从 SRAM、System memory 启动

写保护

可通过配置写保护选项字节 WRP0/1/2/3 的值来实现对主存储块对应的页进行写保护控制，写保护开启后，主存储区对应页的内容使用任何方式都不能被修改。

4.4.2.4 解锁/锁定主存储块

复位后 FMC 的 FMC_CTRL2 会被硬件锁定，此时不能直接对 FMC_CTRL2 进行写入操作，必须先按照正确的顺序向 FMC_KEY 写入对应的值来解锁 FMC。

KEY 值如下：

- KEY1=0x45670123
- KEY2=0xCDEF89AB

错误的写入顺序或错误的值均会使程序进入硬件错误，而且此时 FMC 将被锁定，所有 FMC 操作均无效，直到下次复位才能解除。用户也可以通过向控制寄存器 2（FMC_CTRL2）的 LOCK 位写“1”来软件锁定 FMC。

用户在每次 Flash 编程操作中必须按照“Flash 解锁—用户编程—Flash 上锁”的步骤进行，以避免在 Flash 编程操作结束后，因 Flash 未上锁而带来的用户代码/

数据被意外修改等风险。

4.4.3 选项字节

4.4.3.1 擦选项字节

支持擦除功能，正确的选项字节擦除（或选项字节写入操作）结束后，FMC_STS 寄存器的 OCF 将会被置位，若开启了 OCIE 中断则将触发一个操作完成中断。

选项字节擦除流程为：

- (1) 检查 FMC_STS 寄存器的 BUSYF 为 0，表明此时没有其他操作正在进行。
- (2) 解锁 FMC_CTRL2 寄存器的 OBWEN 位。
- (3) 设置 FMC_CTRL2 寄存器的 OBE。
- (4) 设置 FMC_CTRL2 寄存器的 STA。
- (5) 等待 FMC_STS 寄存器的 BUSYF 变为 0。
- (6) 检查 FMC_STS 寄存器的 OCF 标志。
- (7) 清除 FMC_STS 寄存器的 OCF 标志。

4.4.3.2 写选项字节

选项字节均支持写入功能。

选项字节写入流程为：

- (1) 检查 FMC_STS 寄存器的 BUSYF 为 0，表明此时没有其他操作正在进行。
- (2) 解锁 FMC_CTRL2 寄存器的 OBWEN 位。
- (3) 设置 FMC_CTRL2 寄存器的 OBP。
- (4) 向选项字节区地址写入 16 位（半字）数据（注意互补）。
- (5) 等待 FMC_STS 寄存器的 BUSYF 变为 0。
- (6) 检查 FMC_STS 寄存器的 OCF 标志。
- (7) 清除 FMC_STS 寄存器的 OCF 标志。

4.4.3.3 写保护选项字节

默认状态下，选项字节始终是可以读且被写保护。要想对选项字节块进行写操作（编程/擦除）首先要在 FMC_OBKEY 中写入正确的键序列（与上锁时一样），随后允许对选项字节块的写操作，FLASH_CTRL2 寄存器的 OBWEN 位标示允许写，清除这位将禁止写操作。

4.4.3.4 解锁/锁定选项字节

系统复位后选项字节默认处于锁定状态，只有进行正确的解锁操作才能修改选项字节。选项字节解锁与闪存解锁不同点在 KEY 值写入的是 FMC_OBKEY 寄存器而不是 FMC_KEY 寄存器。选项字节不支持“软件上锁”，用户需要特别注意的是每次修改选项字节的值后，需要系统复位才能使之生效。

4.4.4 选项字节功能描述

选项字节为用户提供了一些可供选择的功能，它主要由多个可配置的字节和对应的补码组成。在每次系统复位后，选项字节区将被重新加载到 FMC_OBCS 和 FMC_WRTPROT 寄存器（选项字节只有每次被重新加载到 FMC 后才会生效）。在重新加载过程中，若出现某个可配置字节与它的反码不匹配问题，则将触发一个选项字节错误（FMC_寄存器的 OBE 位置“1”），同时这个字节将会被置为“0xFF”。

APM32F035x8T7 加密可分为数据段和指令段加密：

数据段加密由安全配置字 dcode_cipher_page[63:0]来配置，每一个 bit 对应一个 page，dcode_cipher_page[0]~ dcode_cipher_page[63]分别对应 page0~page63。

指令段加密由安全配置字 icode_cipher_page[63:0]来配置，每一个 bit 对应一个 page，icode_cipher_page[0]~ icode_cipher_page[63]分别对应 page0~page63。

加密的密钥可以由 opt_enc_same_key, icode_cipher_key_sel 和 dcode_cipher_key_sel 配置，密钥可以选择来自 UID 或者来自寄存器配置。

加密模块的使能通过 SYSCFG 中的寄存器进行配置。

选项字节区的信息见下表。

表格 14 选项字节

地址	选项字节	初始值	R/W	功能描述
0x1FFF F800	READPROT	0xA5	R/W	读保护配置 位[7:0]: READPROT 0xAA: 等级 0 0xCC: 等级 2 其它: 等级 1
0x1FFF F801	nREADPROT	0x5A	R	READPROT 补码
0x1FFF F802	UOB	0xFF	R/W	用户选项字节 位 0: WDTSEL 0: 硬件看门狗 1: 软件看门狗 位 1: nRSTSTOP 0: 进入 Stop 模式时产生复位 1: 进入 Stop 模式时不产生复位 位 2: nRSTSTB

地址	选项字节	初始值	R/W	功能描述
				0: 进入 Standby 模式时产生复位 1: 进入 Standby 模式时不产生复位 位 3: 保留 位 4: nBOOT1 选择 BOOT 模式 位 5: VDDAMONI 0: V _{DDA} 供电检测器禁止 1: V _{DDA} 供电检测器使能 位 6: 保留 位 7: 保留
0x1FFF F803	nUOB	0x00	R	UOB 补码
0x1FFF F804	Data0	0xFF	R/W	用户数据字节 0
0x1FFF F805	nData0	0x00	R	Data0 补码
0x1FFF F806	Data1	0xFF	R/W	用户数据字节 1
0x1FFF F807	nData1	0x00	R	Data1 补码
0x1FFF F808	WRP0	0xFF	R/W	写保护配置 0
0x1FFF F809	nWRP0	0x00	R	WRP0 补码
0x1FFF F80A	WRP1	0xFF	R/W	写保护配置 1
0x1FFF F80B	nWRP1	0x00	R	WRP1 补码
0x1FFF F80C	WRP2	0xFF	R/W	写保护配置 2
0x1FFF F80D	nWRP2	0x00	R	WRP2 补码
0x1FFF F80E	WRP3	0xFF	R/W	写保护配置 3
0x1FFF F80F	nWRP3	0x00	R	WRP3 补码
0x1FFF F810	icode_cipher_page[7:0]	0xFF	W	icode_cipher_page[63:0] 为程序段加扰扇区号使能, 每个 bit 为对应 page 的加扰使能。 每一个 bit 加扰一个 page; icode_cipher_page[0] 对应 Page0, icode_cipher_page[63] 对应 Page63. 1'b0: 加扰使能 1'b1: 加扰不使能
0x1FFF F811	nicode_cipher_page[7:0]	0xFF	W	对应 bit 的补码
0x1FFF F812	icode_cipher_page[15:8]	0xFF	W	-
0x1FFF F813	nicode_cipher_page[15:8]	0xFF	W	对应 bit 的补码
0x1FFF F814	icode_cipher_page[23:16]	0xFF	W	-
0x1FFF F815	nicode_cipher_page[23:16]	0xFF	W	对应 bit 的补码
0x1FFF F816	icode_cipher_page[31:24]	0xFF	W	-

地址	选项字节	初始值	R/W	功能描述
0x1FFF F817	nicode_cipher_page[31:24]	0xFF	W	对应 bit 的补码
0x1FFF F818	icode_cipher_page[39:32]	0xFF	W	-
0x1FFF F819	nicode_cipher_page[39:32]	0xFF	W	对应 bit 的补码
0x1FFF F81A	icode_cipher_page[47:40]	0xFF	W	-
0x1FFF F81B	nicode_cipher_page[47:40]	0xFF	W	对应 bit 的补码
0x1FFF F81C	icode_cipher_page[55:48]	0xFF	W	-
0x1FFF F81D	nicode_cipher_page[55:48]	0xFF	W	对应 bit 的补码
0x1FFF F81E	icode_cipher_page[63:56]	0xFF	W	-
0x1FFF F81F	nicode_cipher_page[63:56]	0xFF	W	对应 bit 的补码
0x1FFF F820	dcode_cipher_page[7:0]	0xFF	W	dcode_cipher_page[63:0]为数据段加扰扇区号使能,每个 bit 为对应 page 的加扰使能。 每一个 bit 加扰一个 page; dcode_cipher_page[0] 对应 Page0, dcode_cipher_page[63]对应 Page63. 1'b0: 加扰使能 1'b1: 加扰不使能
0x1FFF F821	ndcode_cipher_page[7:0]	0xFF	W	对应 bit 的补码
0x1FFF F822	dcode_cipher_page[15:8]	0xFF	W	-
0x1FFF F823	ndcode_cipher_page[15:8]	0xFF	W	对应 bit 的补码
0x1FFF F824	dcode_cipher_page[23:16]	0xFF	W	-
0x1FFF F825	ndcode_cipher_page[23:16]	0xFF	W	对应 bit 的补码
0x1FFF F826	dcode_cipher_page[31:24]	0xFF	W	-
0x1FFF F827	ndcode_cipher_page[31:24]	0xFF	W	对应 bit 的补码
0x1FFF F828	dcode_cipher_page[39:32]	0xFF	W	-
0x1FFF F82A	dcode_cipher_page[47:40]	0xFF	W	-
0x1FFF F82B	ndcode_cipher_page[47:40]	0xFF	W	对应 bit 的补码
0x1FFF F82C	dcode_cipher_page[55:48]	0xFF	W	-
0x1FFF F82D	ndcode_cipher_page[55:48]	0xFF	W	对应 bit 的补码
0x1FFF F82E	dcode_cipher_page[63:56]	0xFF	W	-
0x1FFF F82F	ndcode_cipher_page[63:56]	0xFF	W	对应 bit 的补码
0x1FFF F830	icode_cipher_key_sel	0xFF	W	位 7-2: 保留 位 1-0: icode_cipher_key_sel 2'b11:保留 2'b10: icode_cipher_key 来自 UID

地址	选项字节	初始值	R/W	功能描述
				2'b0x: icode_cipher_key 来自 SYSCFG 寄存器
0x1FFF F831	nicode_cipher_key_sel	0xFF	W	位 7-3: 与 0x1FFF F830 位 7-3 互补 位 2: nopt_enc_same_key 位 1-0: nicode_cipher_key_sel
0x1FFF F832	dcode_cipher_key_sel	0xFF	W	位 7-3: 保留 位 2: opt_enc_same_key 1'b1: 数据加密和指令加密使用相同的密钥 (两个加密模块的加密由 icode_cipher_key 和 dcode_cipher_key 通过运算共同生成, 因此需要保证在程序运行过程中不要去改变 icode_cipher_key 和 dcode_cipher_key), 此时允许数据加密空间和指令加密空间 page 重合 1'b0: 数据加密和指令加密密钥独立, 此时不允许数据加密空间和指令加密空间 page 重合 位 1-0: dcode_cipher_key_sel 2'b11: 保留 2'b10: dcode_cipher_key 来自 UID 2'b0x: dcode_cipher_key 来自 SYSCFG 寄存器
0x1FFF F833	ndcode_cipher_key_sel	0xFF	W	位 7-2: 与 0x1FFF F832 位 7-2 互补 位 1-0: ndcode_cipher_key_sel

注意：当可配置字节和它的反码值为“0xFF”时，重加载过程中将不会再进行匹配验证。Flash 读/写保护配置是互相独立的，解除写保护不会强制丢失主存储区的内容，而是原样保留。

4.4.5 OTP 选项字节

4.4.5.1 擦 OTP 选项字节

不支持擦除功能。

4.4.5.2 写 OTP 选项字节

OTP 选项字节只在全 F 状态下支持写入功能。

4.4.5.3 写保护 OTP 选项字节

默认状态下，OTP 选项字节始终是被写保护。要想对选项字节块进行写操作（编程/擦除）首先要在 FMC_OBKEY 中写入正确的键序列（与上锁时一样），随后允许对选项字节块的写操作，FLASH_CTRL2 寄存器的 OBWEN 位标示允许写，清除这位将禁止写操作。

4.4.5.4 解锁/锁定 OTP 选项字节

系统复位后 OTP 选项字节默认处于锁定状态，只有进行正确的解锁操作才能修改 OTP 选项字节。OTP 选项字节解锁与闪存解锁不同点在 KEY 值写入的是 FMC_OBKEY 寄存器而不是 FMC_KEY 寄存器。OTP 选项字节不支持“软件上锁”，用户需要特别注意的是每次修改 OTP 选项字节的值后，需要系统复位才能使之生效。

4.4.6 OTP 选项字节功能描述

APM32F035x8T7 支持对 Flash 的存储内容进行加密。

max_security_page[63:0]: 最高安全区扇区号，每个 bit 保护一个 page。

max_security_page[0]对应 page0，max_security_page[63]对应 page63。

最高安全区域致此区域只能用于执行用户代码，即只允许 CPU 以取指令的方式去访问这块区域；针对此区域任何其他形式的读操作都会引起一次 flash main 区域整片擦除；一般情况下，建议最高安全区域覆盖程序加扰空间。

opt_valid_key[15:0]为此区域配置使能，配置好此区域的其他 option byte 再将其配置为 16'h0914 使配置生效。

表格 15 选项字节功能描述

地址	选项字节	初始值	R/W	功能描述
0x1FFF F780	opt_valid_key[7:0]	0xFF	W	当 opt_valid_key[15:0]=16'h0914 时，此区域的 option byte 才生效，因此配置时需注意先配置好此区域的 option byte，最后再配置这个 key
0x1FFF F781	nopt_valid_key[7:0]	0xFF	W	对应 bit 的补码
0x1FFF F782	opt_valid_key[15:8]	0xFF	W	-
0x1FFF F783	nopt_valid_key[15:8]	0xFF	W	对应 bit 的补码
0x1FFF F788	user_swid_key[7:0]	0xFF	W	user_swid_key[15:0]=16'h2021 时，SWD 调试接口将被关闭
0x1FFF F789	nuser_swid_key[7:0]	0xFF	W	对应 bit 的补码
0x1FFF F78A	user_swid_key[15:8]	0xFF	W	-
0x1FFF F78B	nuser_swid_key[15:8]	0xFF	W	对应 bit 的补码
0x1FFF F78C	max_security_page[7:0]	0xFF	W	max_security_page[63:0]: 最高安全区扇区号，每个 bit 保护一个 page。 max_security_page[0] 对应 page0， max_security_page[63] 对应 page63

地址	选项字节	初始值	R/W	功能描述
				最高安全区域致此区域只能用于执行用户代码，即程序段；针对此区域任何其他形式的读操作都会引起一次 flash main 区域整片擦除；一般情况下，建议最高安全区域覆盖程序加扰空间； 1'b0:最高安全区域使能 1'b1: 最高安全区域不使能
0x1FFF F78D	nmax_security_page[7:0]	0xFF	W	对应 bit 的补码
0x1FFF F78E	max_security_page[7:0]	0xFF	W	-
0x1FFF F78F	nmax_security_page[7:0]	0xFF	W	对应 bit 的补码
0x1FFF F790	max_security_page[15:8]	0xFF	W	-
0x1FFF F791	nmax_security_page[15:8]	0xFF	W	对应 bit 的补码
0x1FFF F792	max_security_page[23:16]	0xFF	W	-
0x1FFF F793	nmax_security_page[23:16]	0xFF	W	对应 bit 的补码
0x1FFF F794	max_security_page[31:24]	0xFF	W	-
0x1FFF F795	nmax_security_page[31:24]	0xFF	W	对应 bit 的补码
0x1FFF F796	max_security_page[39:32]	0xFF	W	-
0x1FFF F797	nmax_security_page[39:32]	0xFF	W	对应 bit 的补码
0x1FFF F798	max_security_page[47:40]	0xFF	W	-
0x1FFF F799	nmax_security_page[47:40]	0xFF	W	对应 bit 的补码
0x1FFF F79A	max_security_page[55:48]	0xFF	W	-
0x1FFF F79B	nmax_security_page[55:48]	0xFF	W	对应 bit 的补码

4.5 寄存器地址映射

基地址：0x40022000

表格 16 FMC 寄存器地址映射

寄存器名	描述	偏移地址
FMC_CTRL1	控制寄存器 1	0x00
FMC_KEY	关键字寄存器	0x04
FMC_OBKEY	选项字节寄存器	0x08
FMC_STS	状态寄存器	0x0C
FMC_CTRL2	控制寄存器 2	0x10
FMC_ADDR	闪存地址寄存器	0x14

寄存器名	描述	偏移地址
FMC_OBCS	选项字节控制/状态寄存器	0x1C
FMC_WRTPROT	写保护寄存器	0x20

4.6 寄存器功能描述

4.6.1 控制寄存器 1 (FMC_CTRL1)

偏移地址: 0x00

复位值: 0x0000 0000

位/域	名称	R/W	描述
2:0	WS	R/W	配置等待周期 (Wait State Configure) 000: 0 个等待周期, 0<系统时钟≤24MHz 001: 1 个等待周期, 24MHz<系统时钟≤48MHz 010: 2 个等待周期, 48MHz<系统时钟≤72MHz 其他: 保留
3	保留		
4	PBEN	R/W	使能预取缓存区 (Prefetch Buffer Enable) 0: 禁用 1: 使能
5	PBSF	R	预取缓存区状态标志 (Prefetch Buffer Status Flag) 0: 处于关闭状态 1: 处于打开状态
31:6	保留		

4.6.2 关键字寄存器 1 (FMC_KEY)

偏移地址: 0x04

复位值: xxxx xxxx

位/域	名称	R/W	描述
31:0	KEY	W	FMC 关键字 (FMC Key) 写入这些位所表示的关键字可以解锁 FMC, 这些位只能执行写操作, 执行读操作时返回 0。

4.6.3 选项字节关键字寄存器 (FMC_OBKEY)

偏移地址: 0x08

复位值: xxxx xxxx

位/域	名称	R/W	描述
31:0	OBKEY	W	选项字节关键字 (Option Byte Key) 写入这些位所表示的关键字可以解除选项字节写操作的锁定, 这些位只能执行写操作, 执行读操作时返回 0。

4.6.4 状态寄存器 (FMC_STS)

偏移地址: 0x0C

复位值: 0x0000 0000

位/域	名称	R/W	描述
0	BUSYF	R	忙碌标志 (Busy Flag) 该位表示正在进行闪存操作，这些位只能执行写操作，执行读操作时返回0。
1	保留		
2	PEF	R/W	编程错误标志 (Programming Error Flag) 地址被编辑前的值不是“0xFFFF”时，该位被软件置位。
3	保留		
4	WPEF	R/W	写保护错误标志 (Write Protection Error Flag) 当编程 FLASH 里写保护地址时，该位由硬件置位。
5	OCF	R/W	操作完成标志 (Operation Complete Flag) 当 FLASH 里的读/写操作完成时，由硬件置位。
31:6	保留		

4.6.5 控制寄存器 2 (FMC_CTRL2)

偏移地址: 0x10

复位值: 0x0000 0080

位/域	名称	R/W	描述
0	PG	R/W	编程 (Program) 该位置 1 进行 Flash 编程操作。
1	PAGEERA	R/W	页擦除 (Page Erase) 该位置 1 进行页擦除。
2	MASSERA	R/W	整片擦除 (Mass Erase) 该位置 1 进行整片擦除。
3	保留		
4	OBP	R/W	编程选项字节 (Option Byte Program) 该位置 1 进行选项字节编程操作。
5	OBE	R/W	擦除选项字节 (Option Byte Erase) 该位置 1 进行选项字节擦除操作。
6	STA	R/W	开始进行擦除操作 (Start Erase) 该位只能由软件置 1，在 STS_BUSYF 位清 0 时复位。
7	LOCK	R/W	锁定 (Lock) 该位只能写 1，该位置 1 表示 FMC 和 CTRL2 寄存器被锁定。
8	保留		
9	OBWEN	R/W	使能选项字节写操作 (Option Byte Write Enable) 当该位置 1 时，选项字节可以被编程。
10	ERRIE	R/W	使能错误中断 (Error Interrupt Enable) 0: 禁止中断 1: 使能中断 当 STS_PEF=1 或 STS_WPEF=1 时，设置该位产生中断。
11	保留		

位/域	名称	R/W	描述
12	OCIE	R/W	使能操作完成中断 (Operation Complete Interrupt Enable) 0: 操作完成中断禁用 1: 操作完成中断使能 当 STS_OCF=1 时, 设置该位产生中断。
13	OBLOAD	R/W	强制选项字节加载 (Force Option Byte Load) 该位置 1 时, 强制选项字节重装载, 生成系统复位。 0: 闲置 1: 强制加载
31:14	保留		

4.6.6 地址寄存器 (FMC_ADDR)

偏移地址: 0x14

复位值: 0x0000 0000

该寄存器由硬件修改为当前/最后使用的地址; 页擦除时, 需通过软件配置该寄存器。

位/域	名称	R/W	描述
31:0	ADDR	W	Flash 地址 (Flash Address) 在进行编程操作时, 该位写入要编程的地址, 在进行页擦除时, 该位写入要擦除的页。

4.6.7 选项字节控制/状态寄存器 (FMC_OBCS)

偏移地址: 0x1C

复位值: 0xXXXX XX0X

该寄存器的复位值与写入选项字节中的数值有关; OBE 位的复位值与加载的选项字节数值和它的反码是否一致的结果有关。

位/域	名称	R/W	描述
0	OBE	R	选项字节错误 (Option Byte Error) 1: 所加载的选项字节和它的补码不匹配, 选项字节和它的补码强制写入 0xFF
2:1	READPROT	R	读保护 (Read Protect) 显示使能了哪个等级的读保护。 bit1 置一时即为等级 1, bit2 置一时即为等级 2。 00: 等级 0 01: 等级 1 1X: 等级 2
7:3	保留		
8	WDTSEL	R	选择看门狗 (Watchdog Select) 0: 硬件看门狗 1: 软件看门狗
9	RSTSTOP	R	进入停机模式时产生复位 (nReset in STOP Mode) 0: 产生 1: 不产生

位/域	名称	R/W	描述
10	RSTSTDB	R	进入待机模式时产生复位 (nReset in STANDBY Mode) 0: 产生 1: 不产生
11	保留		
12	nBOOT1	R	配置 nBOOT1 模式 (nBoot1 Mode Configure)
13	VDDAMONI	R	V _{DDA} 电压监控 (V _{DDA} Monitor)
14	SRAMPARITY	R	SRAM 奇偶校验 (SRAM Parity Check)
15	保留		
23:16	DATA0	R	Data0
31:24	DATA1	R	Data1

4.6.8 写保护寄存器 (FMC_WRTPROT)

偏移地址: 0x20

复位值: 0xXXXX XXXX (复位值取决于选项字节中编程值)

位/域	名称	R/W	描述
31:0	WRTPROT	R	写保护 (Write Protect) 0: 有效 1: 无效

5 系统配置控制器（SYSCFG）

5.1 术语全称、缩写描述

表格 17 术语全称、缩写描述

中文全称	英文全称	英文缩写
超快模式	Fast Mode Plus	FM+
系统配置控制器	System Configuration Controller	SYSCFG

SYSCFG 主要用于管理地址映射和控制中断，具体是指：控制部分 IO 口上的 I2C 的超快模式；配置 DMA 触发源的重映射；存储器到代码起始区的重映射；管理连接到 GPIO 的外部中断；配置加密功能。

有关配置的详情请见 SYSCFG 寄存器配置。

5.2 寄存器地址映射

表格 18 SYSCFG 寄存器地址映射

寄存器名	描述	偏移地址
SYSCFG_CFG1	配置寄存器 1	0x00
SYSCFG_EINTCFG1	外部中断寄存器 1	0x08
SYSCFG_EINTCFG2	外部中断寄存器 2	0x0C
SYSCFG_EINTCFG3	外部中断寄存器 3	0x10
SYSCFG_EINTCFG4	外部中断寄存器 4	0x14
SYSCFG_CFG2	配置寄存器 2	0x18
SYSCFG_DCODE	数据加扰密钥寄存器	0x20
SYSCFG_ICODE	指令加扰密钥寄存器	0x24
SYSCFG_CIPHEN	加扰密钥寄存器	0x28
SYSCFG_ACCESSEN	处理器访问 IP 权限寄存器	0x2C

5.3 寄存器功能描述

5.3.1 配置寄存器 1（SYSCFG_CFG1）

偏移地址：0x00

复位值：0x0000 000X（X 指存储器模式，由 BOOT 控制。复位后，这些位通过 BOOT 引脚选择模式配置参数）

该寄存器用于配置内存和 DMA 请求的重映射以及控制特定 I/O 引脚。

两位用来配置地址为 0x0000 0000 的存储类型。

这些位全部都可以跳过硬件由软件选择物理映射，且可以通过软件进行控制和复位。

位/域	名称	R/W	描述
1:0	MMSEL	R/W	存储映射选择 (Memory Mapping Select) 控制内存映射地址 0x0000 0000。重置后这些位的参数由实际 BOOT 决定。 X0: 主闪存映射地址: 0x0000 0000 01: 系统闪存映射地址: 0x0000 0000 11: 嵌入式 SRAM 映射地址: 0x0000 0000
5:2	保留		
6	IRSEL	R/W	调制信号源选择 (IR Modulation Envelope Signal Select) 该位控制调制信号来自 TMR3 还是 USRAT1。 0: 调制信号来自 TMR3 1: 调制信号来自 USRAT1
7	保留		
8	ADCDMARMP	R/W	ADC DMA 请求重映射 (ADC DMA Request Remap) 控制 ADC DMA 的重映射请求。 0: 无重映射 ADC—DMA_CH1 1: 重映射 ADC—DMA_CH2
9	USART1TXRMP	R/W	USART1_TX DMA 请求重映射 (USART1_TX DMA Request Remap) 该位控制 USART1_TX DMA 的重映射请求。 0: 无重映射 USART1_RX—DMA_CH2 1: 重映射 USART1_RX—DMA_CH4
10	USART1RXRMP	R/W	USART1_RX DMA 请求重映射 (USART1_RX DMA Request Remap) 该位控制 USART1_RX DMA 的重映射请求。 0: 无重映射 USART1_TX—DMA_CH3 1: 重映射 USART1_TX—DMA_CH5
15:11	保留		
16	I2CPB6FMP	R/W	激活 PB6 引脚超快模式 (FM+) 驱动能力 (Fast Mode Plus Driving Capability Activate for PB6) 该位使能 PB6 接口开启 I2C 超快模式。 0: PB6 引脚被设置为标准模式。 1: PB6 引脚被设置为 I2C 超快模式, 且 I2C 速度控制被旁路 (被忽略)。
17	I2CPB7FMP	R/W	激活 PB7 引脚超快模式 (FM+) 驱动能力 (Fast Mode Plus Driving Capability Activate for PB7) 该位使能 PB7 接口开启 I2C 超快模式。 0: PB7 引脚被设置为标准模式。 1: PB7 引脚被设置为 I2C 超快模式, 且 I2C 速度控制被旁路 (被忽略)。

位/域	名称	R/W	描述
18	I2CPB8FMP	R/W	激活 PB8 引脚超快模式 (FM+) 驱动能力 (Fast Mode Plus Driving Capability Activate for PB8) 该位使能 PB8 接口开启 I2C 超快模式。 0: PB8 引脚被设置为标准模式。 1: PB8 引脚被设置为 I2C 超快模式, 且 I2C 速度控制被旁路 (被忽略)。
19	I2CPB9FMP	R/W	激活 PB9 引脚超快模式 (FM+) 驱动能力 (Fast Mode Plus Driving Capability Activate for PB9) 该位使能 PB9 接口开启 I2C 超快模式。 0: PB9 引脚被设置为标准模式。 1: PB9 引脚被设置为 I2C 超快模式, 且 I2C 速度控制被旁路 (被忽略)。
20	I2CPC4FMP	R/W	激活 PC4 引脚超快模式 (FM+) 驱动能力 (Fast Mode Plus Driving Capability Activate for PC4) 该位使能 PC4 接口开启 I2C 超快模式。 0: PC4 引脚被设置为标准模式 1: PC4 引脚被设置为 I2C 超快模式, 且 I2C 速度控制被旁路 (被忽略)
21	I2CPC5FMP	R/W	激活 PC5 引脚超快模式 (FM+) 驱动能力 (Fast Mode Plus Driving Capability Activate for PC5) 该位使能 PC5 接口开启 I2C 超快模式。 0: PC5 引脚被设置为标准模式 1: PC5 引脚被设置为 I2C 超快模式, 且 I2C 速度控制被旁路 (被忽略)
22	SPIDMARMP	R/W	SPI 的 DMA 请求重映射 (SPI DMA Request Remap) 该位控制 SPI 的 DMA 重映射请求。 0: 无重映射 SPI_RX—DMA_CH2, SPI_TX—DMA_CH3 1: 重映射 SPI_RX—DMA_CH4, SPI_TX—DMA_CH5
26:23	保留		
27	I2CDMARMP	R/W	I2C 的 DMA 请求重映射 (I2C DMA Request Remap) 该位控制 I2C 的 DMA 重映射请求。 0: 无重映射 I2C_RX—DMA_CH3, I2C_TX—DMA_CH2 1: 重映射 I2C_RX—DMA_CH7, I2C_TX—DMA_CH6
28	TMR1DMARMP	R/W	TMR1 的 DMA 请求重映射 (TMR1 DMA Request Remap) 该位控制 TMR1 的 DMA 重映射请求。 0: 无重映射 TMR1_CH1—DMA_CH2, TMR1_CH2—DMA_CH3, TMR1_CH3—DMA_CH5 1: 重映射 TMR1_CH1 — DMA_CH6, TMR1_CH2 — DMA_CH6, TMR1_CH3—DMA_CH6

位/域	名称	R/W	描述
29	TMR2DMARMP	R/W	TMR2 的 DMA 请求重映射 (TMR2 DMA Request Remap) 该位控制 TMR2 的 DMA 重映射请求。 0: 无重映射 TMR2_CH1—DMA_CH3, TMR2_CH2—DMA_CH4 1: 重映射 TMR2_CH1—DMA_CH7, TMR2_CH2—DMA_CH7
30	TMR3DMARMP	R/W	TMR3 的 DMA 请求重映射 (TMR3 DMA Request Remap) 该位控制 TMR3 的 DMA 重映射请求。 0: 无重映射 TMR3_CH1—DMA_CH4, TMR3_CH2—DMA_CH4 1: 重映射 TMR3_CH1—DMA_CH6, TMR3_CH2—DMA_CH6
31	保留		

5.3.2 外部中断寄存器 1 (SYSCFG_EINTCFG1)

这些位由软件控制进行改写来选择 EINTx(x=0...15)的外部中断源。EINTx [3:0] (注意: EINT3/7/11/15 为[2:0], 只有 3 个 bits) 的值所代表选择的外部中断源如下表所示。

注意, 具体的引脚数目请见数据手册的“引脚定义”章节。

表格 19 不同值所选择外部中断源

EINTx [3:0]	外部中断源
x000	PA[x] 引脚
x001	PB[x] 引脚
x010	PC[x] 引脚
x101	PF[x] 引脚
其他	保留

偏移地址: 0x08

复位值: 0x0000 0000

位/域	名称	R/W	描述
3:0	EINT0	R/W	EINT 0 配置位 (EINT0 Configure) 这些位由软件控制进行改写来选择 EINT0 的外部中断源。 位的值所表示选择的外部中断源如表格 19 所示。
7:4	EINT1	R/W	EINT1 配置位 (EINT1 Configure) 这些位由软件控制进行改写来选择 EINT1 的外部中断源。 位的值所表示选择的外部中断源如表格 19 所示。
11:8	EINT2	R/W	EINT2 配置位 (EINT2 Configure) 这些位由软件控制进行改写来选择 EINT2 的外部中断源。 位的值所表示选择的外部中断源如表格 19 所示。

位/域	名称	R/W	描述
14:12	EINT3	R/W	EINT3 配置位 (EINT3 Configure) 这些位由软件控制进行改写来选择 EINT3 的外部中断源。 位的值所表示选择的外部中断源如表格 19 所示。
31:15	保留		

5.3.3 外部中断寄存器 2 (SYSCFG_EINTCFG2)

这些位由软件控制进行改写来选择 EINTx(x=4...7)的外部中断源。EINTx [3:0]的值所代表选择的外部中断源如表格 19 所示。

偏移地址: 0x0C

复位值: 0x0000 0000

位/域	名称	R/W	描述
3:0	EINT4	R/W	EINT 4 配置位 (EINT4 Configure) 这些位由软件控制进行改写来选择 EINT4 的外部中断源。 位的值所表示选择的外部中断源如表格 19 所示。
7:4	EINT5	R/W	EINT5 配置位 (EINT5 Configure) 这些位由软件控制进行改写来选择 EINT5 的外部中断源。 位的值所表示选择的外部中断源如表格 19 所示。
11:8	EINT6	R/W	EINT6 配置位 (EINT6 Configure) 这些位由软件控制进行改写来选择 EINT6 的外部中断源。 位的值所表示选择的外部中断源如表格 19 所示。
14:12	EINT7	R/W	EINT7 配置位 (EINT7 Configure) 这些位由软件控制进行改写来选择 EINT7 的外部中断源。 位的值所表示选择的外部中断源如表格 19 所示。
31:15	保留		

5.3.4 外部中断寄存器 3 (SYSCFG_EINTCFG3)

这些位由软件控制进行改写来选择 EINTx(x=8...11)的外部中断源。EINTx [3:0]的值所代表选择的外部中断源如表格 19 所示。

偏移地址: 0x10

复位值: 0x0000 0000

位/域	名称	R/W	描述
3:0	EINT8	R/W	EINT8 配置位 (EINT8 Configure) 这些位由软件控制进行改写来选择 EINT8 的外部中断源。 位的值所表示选择的外部中断源如表格 19 所示。
7:4	EINT9	R/W	EINT9 配置位 (EINT9 Configure) 这些位由软件控制进行改写来选择 EINT9 的外部中断源。 位的值所表示选择的外部中断源如表格 19 所示。
11:8	EINT10	R/W	EINT10 配置位 (EINT10 Configure) 这些位由软件控制进行改写来选择 EINT10 的外部中断源。 位的值所表示选择的外部中断源如表格 19 所示。
14:12	EINT11	R/W	EINT11 配置位 (EINT11 Configure) 这些位由软件控制进行改写来选择 EINT11 的外部中断源。 位的值所表示选择的外部中断源如表格 19 所示。

位/域	名称	R/W	描述
31:15			保留

5.3.5 外部中断寄存器 4 (SYSCFG_EINTCFG4)

这些位由软件控制进行改写来选择 EINTx(x=12 to 15)的外部中断源。EINTx [3:0] 的值所代表选择的外部中断源如表格 19 所示。

偏移地址: 0x14

复位值: 0x0000 0000

位/域	名称	R/W	描述
3:0	EINT12	R/W	EINT12 配置位 (EINT12 Configure) 这些位由软件控制进行改写来选择 EINT12 的外部中断源。 位的值所表示选择的外部中断源如表格 19 所示。
7:4	EINT13	R/W	EINT13 配置位 (EINT13 Configure) 这些位由软件控制进行改写来选择 EINT13 的外部中断源。 位的值所表示选择的外部中断源如表格 19 所示。
11:8	EINT14	R/W	EINT14 配置位 (EINT14 Configure) 这些位由软件控制进行改写来选择 EINT14 的外部中断源。 位的值所表示选择的外部中断源如表格 19 所示。
14:12	EINT15	R/W	EINT15 配置位 (EINT15 Configure) 这些位由软件控制进行改写来选择 EINT15 的外部中断源。 位的值所表示选择的外部中断源如表格 19 所示。
31:15			保留

5.3.6 配置寄存器 2 (SYSCFG_CFG2)

偏移地址: 0x18

复位值: 0x0000 0000

位/域	名称	R/W	描述
0	LOCK	R/W	使能内核 LOCKUP (Core LOCKUP Enable) 该位由软件置位, 由系统复位清除。 它可以使能和锁定 Arm® Cortex®-M0+ LOCKUP 的 Hardfault(硬件错误)输出与 TMR1 刹车输入之间的连接。
1			保留
2	PVDLOCK	R/W	使能 PVD 锁定 (PVD Lock Enable) 能够使能和锁定 PVD 中断与 TMR1 刹车输入之间的连接, 并且能够锁定 PMU_CTRL 中的 PVDEN 位和 PLSEL 位的状态。 0: 锁定连接; PVDEN 位和 PLSEL 位为可编辑状态 1: 使能连接; PVDEN 位和 PLSEL 位为只读状态
31:3			保留

5.3.7 数据加扰密钥寄存器 (SYSCFG_DCODE)

偏移地址: 0x20

复位值: 0x0000 0000

位/域	名称	R/W	描述
31:0	DKEY	W	数据加扰密钥 (Data Code Cipher Key)

5.3.8 指令加扰密钥寄存器 (SYSCFG_ICODE)

偏移地址: 0x24

复位值: 0x0000 0000

位/域	名称	R/W	描述
31:0	IKEY	W	指令加扰密钥 (Instruction Code Cipher Key)

5.3.9 加扰使能寄存器 (SYSCFG_CIPHEN)

偏移地址: 0x28

复位值: 0x0000 0004

位/域	名称	R/W	描述
0	DCIPHEN	R/W	使能数据加扰 (Instruction Code Cipher Enable) 0: 禁止 1: 使能
1	ICIPHEN	R/W	使能指令加扰 (Instruction Code Cipher Enable) 0: 禁止 1: 使能
31:2	保留		

5.3.10 处理器和 DMA 访问 IP 权限寄存器 (SYSCFG_ACCESEN)

偏移地址: 0x2C

复位值: 0x17FF EEC9

位/域	名称	R/W	描述
0	DMAEN	R/W	使能处理器访问 DMA (CPU Access DMA Enable) 0: 禁止 1: 使能
2:1	保留		
3	CRCEN	R/W	使能处理器和 DMA 访问 CRC (CPU and DMA Access CRC Enable) 0: 禁止 1: 使能
5:4	保留		
6	ADCEN	R/W	使能处理器和 DMA 访问 ADC (CPU and DMA Access ADC Enable) 0: 禁止 1: 使能
7	CANEN	R/W	使能处理器和 DMA 访问 CAN (CPU and DMA Access CAN Enable) 0: 禁止 1: 使能
8	保留		

位/域	名称	R/W	描述
9	EINTEN	R/W	使能处理器和 DMA 访问 EINT (CPU and DMA Access EINT Enable) 0: 禁止 1: 使能
10	I2CEN	R/W	使能处理器和 DMA 访问 I2C (CPU and DMA Access I2C Enable) 0: 禁止 1: 使能
11	IWDTEN	R/W	使能处理器和 DMA 访问 IWDT (CPU and DMA Access IWDT Enable) 0: 禁止 1: 使能
12	保留		
13	RTCEN	R/W	使能处理器和 DMA 访问 RTC (CPU and DMA Access RTC Enable) 0: 禁止 1: 使能
14	SPIEN	R/W	使能处理器和 DMA 访问 SPI (CPU and DMA Access SPI Enable) 0: 禁止 1: 使能
15	TMR1EN	R/W	使能处理器和 DMA 访问 TMR1 (CPU and DMA Access TMR1 Enable) 0: 禁止 1: 使能
16	TMR2EN	R/W	使能处理器和 DMA 访问 TMR2 (CPU and DMA Access TMR2 Enable) 0: 禁止 1: 使能
17	TMR3EN	R/W	使能处理器和 DMA 访问 TMR3 (CPU and DMA Access TMR3 Enable) 0: 禁止 1: 使能
18	TMR4EN	R/W	使能处理器和 DMA 访问 TMR4 (CPU and DMA Access TMR4 Enable) 0: 禁止 1: 使能
19	TMR6EN	R/W	使能处理器和 DMA 访问 TMR6 (CPU and DMA Access TMR6 Enable) 0: 禁止 1: 使能
20	TMR7EN	R/W	使能处理器和 DMA 访问 TMR7 (CPU and DMA Access TMR7 Enable) 0: 禁止 1: 使能

位/域	名称	R/W	描述
21	USART1EN	R/W	使能处理器和 DMA 访问 USART1 (CPU and DMA Access USART1 Enable) 0: 禁止 1: 使能
22	USART2EN	R/W	使能处理器和 DMA 访问 USART2 (CPU and DMA Access USART2 Enable) 0: 禁止 1: 使能
23	WWDTEN	R/W	使能处理器和 DMA 访问 WWDT (CPU and DMA Access WWDT Enable) 0: 禁止 1: 使能
24	GPIOAEN	R/W	使能处理器和 DMA 访问 GPIOA (CPU and DMA Access GPIOA Enable) 0: 禁止 1: 使能
25	GPIOBEN	R/W	使能处理器和 DMA 访问 GPIOB (CPU and DMA Access GPIOB Enable) 0: 禁止 1: 使能
26	GPIOCEN	R/W	使能处理器和 DMA 访问 GPIOC (CPU and DMA Access GPIOC Enable) 0: 禁止 1: 使能
27	保留		
28	GPIOFEN	R/W	使能处理器和 DMA 访问 GPIOF (CPU and DMA Access GPIOF Enable) 0: 禁止 1: 使能
31:29	保留		

6 复位与时钟管理（RCM）

6.1 术语全称、缩写描述

表格 20 术语全称、缩写描述

中文全称	英文全称	英文缩写
复位与时钟管理	Reset and Clock Management	RCM
复位	Reset	RST
上电复位	Power-On Reset	POR
掉电复位	Power-Down Reset	PDR
高速外部时钟	High Speed External Clock	HSECLK
低速外部时钟	Low Speed External Clock	LSECLK
高速内部时钟	High Speed Internal Clock	HSICK
低速内部时钟	Low Speed Internal Clock	LSICK
锁相环	Phase Locked Loop	PLL
主要时钟输出	Main clock output	MCO
校准	Calibrate	CAL
调整	Trim	TRM
时钟安全系统	Clock Security System	CSS
不可屏蔽中断	Non Maskable Interrupt	NMI

6.2 复位功能描述

支持的复位分为三类：系统复位、电源复位、RTC 区域复位三种复位形式。

6.2.1 系统复位

6.2.1.1 “系统复位”复位源

复位源可以分为外部复位源和内部复位源。

外部复位源：

- NRST 引脚上的低电平

内部复位源：

- 窗口看门狗终止计数（WWDT 复位）
- 独立看门狗终止计数（IWDT 复位）
- 软件复位（SW 复位）
- 低功耗管理复位
- 载入选项字节复位

● 电源复位

以上任一事件发生时，都能产生一个系统复位。另外，可以通过查看 RCM_CSTS（控制/状态寄存器）中的复位标志位识别复位事件来源。

一般来说，系统复位时，会将除了 RCM_CSTS 的复位标志位和 RTC 域中寄存器以外的所有寄存器的值复位到复位值。

软件复位

将 Arm® Cortex®-M0+中断应用和复位控制寄存器中的 SYSRESETREQ 置“1”时，可实现软件复位。

低功耗管理复位

在以下两种情况下会产生低功耗管理复位，一种是进入待机模式时，另一种是进入停止模式时。在这两种情况下，如果把用户选择字节中的 RSTSTDB 位（待机模式时）或 RSTSTOP 位（停止模式时）清零，系统将被复位而不是进入待机模式或停止模式。

有关用户选择字节的详细信息，请参考“Flash 存储器”章节。

载入选项字节复位

载入字节复位由被软件控制的在 FMC_CTRL2 寄存器中 OBLOAD 位触发。

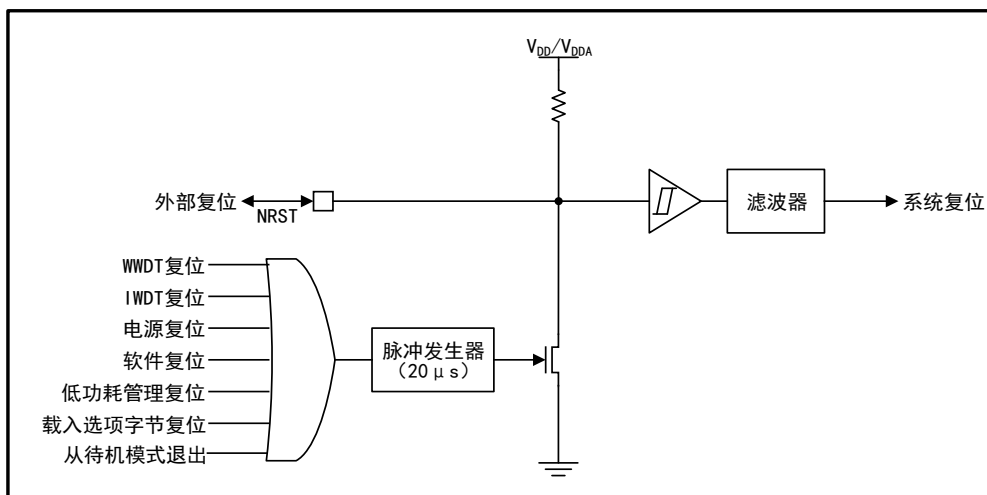
6.2.1.2 “系统复位”复位电路

复位源均作用于 NRST 引脚，该引脚并在复位过程中保持低电平。

内部复位源通过脉冲发生器在 NRST 引脚产生延时至少 20μs 的脉冲，引起 NRST 保持电平产生复位；外部复位源则直接将 NRST 引脚电平拉低产生复位。

“系统复位”复位电路见下图。

图 4 “系统复位”复位电路



6.2.2 电源复位

“电源复位”复位源如下：

- 上电（POR 复位）
- 掉电复位（PDR 复位）
- 从待机模式唤醒

以上任一事件发生时，产生电源复位。

电源复位将复位除了 RTC 以外的所有寄存器。

6.2.3 RTC 域复位

“RTC 域复位”复位源如下：

- 置位 RCM_RTCCTRL 中的 RTCRST 位触发的软件复位
- V_{DD} 掉电

以上任一事件发生时，产生 RTC 域复位。注意 RTC 域也是由 V_{DD} 供电。

RTC 区域复位只影响 RTC 区域。

RTC 域寄存器也能通过以下任一事件实现复位：

- RTC 检测到被修改
- 读保护级别从 1 级被改为 0 级

6.3 时钟管理功能描述

整个系统的时钟源有：HSECLK、LSECLK、HSICLK、HSICLK14、LSICLK、PLL。关于时钟源的特性，请参考数据手册的“电气特性”中的相关章节。

6.3.1 外部时钟源

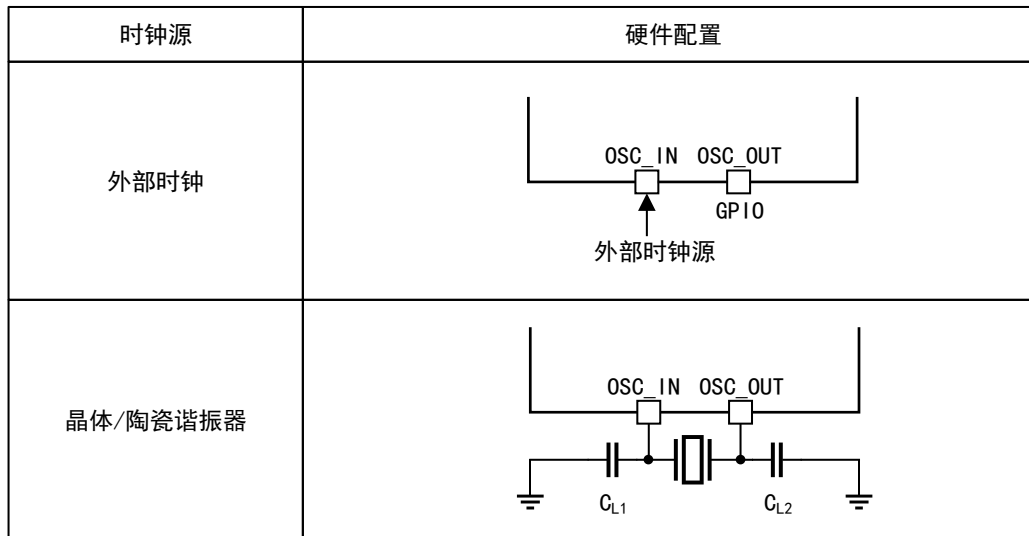
外部时钟信号包括 HSECLK（高速外部时钟信号）和 LSECLK（低速外部时钟信号）。

外部的时钟源有两种：

- 外部晶体/陶瓷谐振器
- 用户外部时钟

两种时钟源的硬件配置如下图所示。

图 5 HSECLK/LSECLK 的时钟源硬件配置



注意：

- (1) 根据所选择的振荡器来调整负载电容（CL1、CL2）的值。
- (2) 为了减少时钟输出的失真和缩短启动稳定时间，晶体/陶瓷谐振器和负载电容器必须尽可能地靠近振荡器引脚。

6.3.1.1 HSECLK 高速外部时钟信号

HSECLK 时钟信号由 HSECLK 外部晶体/陶瓷谐振器和 HSECLK 外部时钟两种时钟源产生。

表格 21 产生 HSECLK 的时钟源

名称	说明
外部时钟源 (HSECLK 旁路)	<p>通过 OSC_IN 引脚给 MCU 提供时钟。</p> <p>信号可以有普通的函数信号发送器（调试时）、晶体振荡器、其它信号发生器产生；波形可以是 40%-60% 占空比的方波、正弦波或三角波，最高频率可达 32MHz。</p> <p>硬件连接上，需连接必须连到 OSC_IN 引脚，同时保证 OSC_OUT 引脚悬空；MCU 配置上，用户可通过设置在 RCM_CTRL1（时钟控制寄存器 1）中的 HSEBCFG 和 HSEEN 位来选择这一模式。</p>
外部晶体/陶瓷谐振器 (HSECLK 晶体)	<p>通过谐振器给 MCU 提供时钟，谐振器包括晶体谐振器、陶瓷谐振器。频率范围是 4-32MHz。</p> <p>需要 OSC_IN、OSC_OUT 连接谐振器，可以通过设置时钟控制寄存器里 RCM_CTRL1（时钟控制寄存器）中的 HSEEN 位，启动和关闭。</p> <p>在时钟控制寄存器 RCM_CTRL1（时钟控制寄存器 1）中的 HSERDYFLG 位用来指示高速外部振荡器是否稳定。在启动后，直到这一位被硬件置“1”，时钟才被释放出来。如果在 RCM_INT（时钟中断寄存器）中允许产生中断，将会产生相应中断。</p>

6.3.1.2 LSECLK 低速外部时钟信号

LSECLK 时钟信号由 LSECLK 外部晶体/陶瓷谐振器和 LSECLK 外部时钟两种时钟源产生。

表格 22 产生 LSECLK 的时钟源

名称	说明
外部时钟源 (LSECLK 旁路)	通过 OSC32_IN 引脚给 MCU 提供时钟。 信号可以有普通的函数信号发送器 (调试时)、晶体振荡器、其它信号发生气产生; 波形可以是 50% 占空比的方波、正弦波或三角波, 信号频率需为 32.768kHz。 硬件连接上, 需连接必须连到 OSC32_IN 引脚, 同时保证 OSC32_OUT 引脚悬空; MCU 配置上, 用户可通过设置在 RCM_RTCCTRL 里的 LSEBCFG 和 LSEEN 位来选择这个模式。
外部晶体/陶瓷谐振器 (LSECLK 晶体)	通过谐振器给 MCU 提供时钟, 谐振器包括晶体谐振器、陶瓷谐振器。 频率是 32.768kHz。 需要 OSC32_IN、OSC32_OUT 连接谐振器, 可以通过在 RCM_RTCCTRL 里的 LSEEN 位启动和关闭。 在 RCM_RTCCTRL 里的 LSERDYFLG 指示 LSECLK 晶体振荡是否稳定。在启动阶段, 直到这个位被硬件置“1”后, LSECLK 时钟信号才被释放出来。 如果在时钟中断寄存器里被允许, 可产生中断申请。

6.3.2 内部时钟源

内部时钟包含 HSICLK (高速内部时钟信号) 和 LSICLK (低速内部时钟信号)。其中, HSICLK 有 8MHz 和 14MHz 两种规格, HSICLK14 主要用于向 ADC 提供时钟信号。

6.3.2.1 HSICLK 高速内部时钟信号

HSICLK 时钟信号由内部 8MHz 的 RC 振荡器产生。

不同芯片的 RC 振荡器频率不同, 且同一颗芯片随着温度、电压的变化也会存在差异; 每个芯片的 HSICLK 时钟频率在出厂前已经被厂家校准到 1% (25°C、 $V_{DD}=V_{DDA}=3.3V$), 系统复位时, 工厂校准值被装载到 RCM_CTRL1 (时钟控制寄存器) 中; 另外, 用户可以依据现场的应用环境 (温度、电压), 通过设置 RCM_CTRL1 中的 HSITRM 作进一步的频率调整。

HSIRDYFLG 位可以用来指示 HSICLK RC 振荡器是否稳定。在时钟启动过程中, 直到 HSIRDYFLG 位被硬件置“1”, HSICLK RC 输出时钟才被释放。HSICLK RC 振荡器可由 RCM_CTRL1 中的 HSIEN 位来启动或关闭。

与 HSECLK 晶体振荡器相比, 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; 它的启动时间比 HSECLK 晶体振荡器短; 即使在校准之后, 其时钟频率精度仍不如 HSECLK 晶体振荡器。

6.3.2.2 HSICLK14 高速内部时钟信号

HSICLK14 时钟信号由内部 14MHz 的 RC 振荡器产生, 主要用于向 ADC 提供时钟信号。HSI14RDYFLG 位可以用来指示 HSICLK14 RC 振荡器是否稳定。在时钟启动过程中, 直到 HSI14RDYFLG 位被硬件置“1”, HSICLK14 RC 输出时钟才被释放。HSICLK14 RC 振荡器可由 RCM_CTRL2 中的 HSI14EN 位来启动或关闭。

用户可以依据现场的应用环境 (温度、电压), 通过设置 RCM_CTRL2 中的 HSI14TRM 作进一步的频率调整。

6.3.2.3 LSICLK 低速内部时钟信号

LSICLK 主要特征

LSICLK 由 RC 振荡器产生，范围是 40kHz（30kHz 和 60kHz 之间）。随着温度、电压的变化，频率也会产生变化。可以在停机和待机模式下保持运行，为 IWDG（独立看门狗）和 RTC（实时时钟）提供时钟。

LSICLK 可以通过 RCM_CSTS（控制/状态寄存器）的 LSIEN 位来启动或关闭。在 RCM_CSTS 里的 LSIRDYFLG 位指示低速内部振荡器是否稳定。在启动阶段，直到这个位被硬件设置为“1”后，时钟才被释放。如果在 RCM_INT（时钟中断寄存器）里被允许，将产生 LSICLK 中断申请。

6.3.3 PLL 锁相环

内部 PLL 可以用来倍频 HSICLK 的输出时钟或 HSECLK 晶体输出时钟。

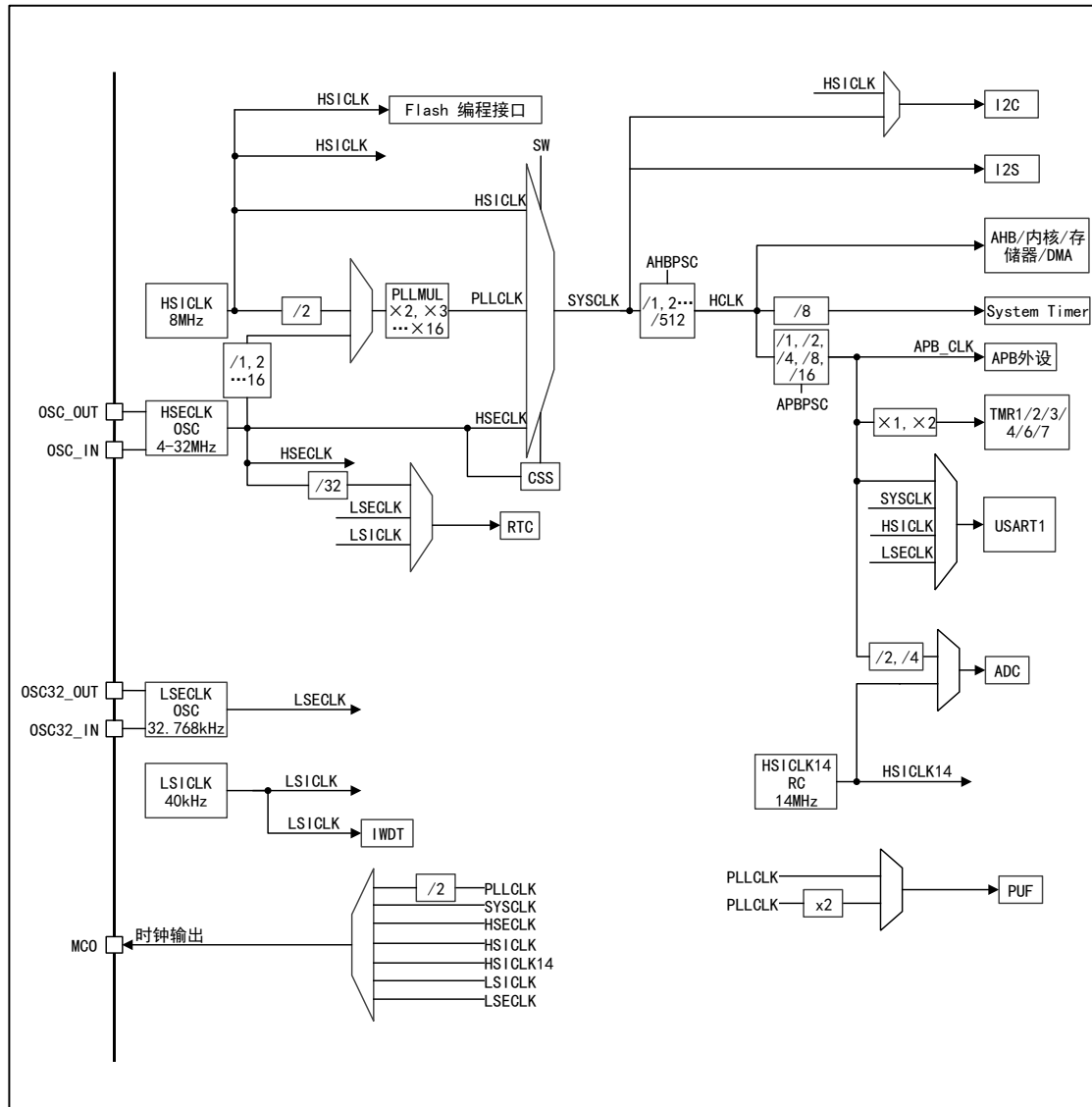
如果要配置 PLL 参数，先将 PLEN 位清 0，等到 PLLRDYFLG 被清 0（PLL 处于停用状态），然后改变参数，再将 PLEN 置 1 通过使能 PLL，等 PLLRDYFLG 被置 1 即可完成配置。

时钟源和倍频因子应在被激活前完成选择，一旦 PLL 被激活，选择将不能改变。

当 PLL 准备就绪且 PLL 中断在 RCM_INT 里被允许时，PLL 可发出中断申请。

6.3.4 时钟树

图 6 APM32F035x8T7 时钟树



注意:

- (1) HCLK 是指 AHB 时钟。
- (2) PCLK 是指连接 APB 的外设的时钟信号。
- (3) FCLK 是 Arm® Cortex®-M0+ 的运行时钟。
- (4) 可通过多个预分频器配置 AHB、APB 域的频率
- (5) 当需要运行连接 AHB, APB 的外设时, 都需要开启相应的使能端, 使外设获得时钟信号。
- (6) 所有 TMRxCLK (定时器时钟) 频率分配由硬件按以下 2 种情况自动设置:
 - 如果相应的 APB 预分频系数是 1, 定时器的时钟频率与所在 APB 总线频率一致。
 - 否则, 定时器的时钟频率被设为与其相连的 APB 总线频率的 2 倍。
- (7) 其中, TMRx (x=1,2,3,4,6,7) 时钟信号通过 APB 分频。
- (8) HSICKL14 RC 振荡器用于给 ADC 提供时钟 (详见“ADC 的时钟源选择”章节)。

6.3.5 RTC 的时钟源选择

通过设置 RCM_RTCCTRL 的 RTCSRCSEL 位选择 HSECLK/32、LSECLK 或 LSICLK 作为 RTCCLK 时钟源。只有 RTC 域复位时，时钟源的选择才可以改变。只有当 PCLK 大于等于 RTCCLK 时，系统才能对 RTC 进行正常操作。

6.3.6 IWDG 的时钟源选择

当 IWDG（独立看门狗）启动时，LSICLK 振荡器将被强制打开，等到稳定后把时钟信号供给 IWDG。LSICLK 在被强制打开后，将一直处于打开状态，不能被关闭。

6.3.7 MCO 的时钟源选择

当相应的 GPIO 端口寄存器被配置为对应功能时，可通过配置 RCM_CFG1（时钟配置寄存器）中的 MCOSEL 选择时钟信号输出到 MCO 引脚。具体时钟信号可参考时钟树或 RCM_CFG1 寄存器的 MCOSEL 位说明。

6.3.8 SYSCLK 的时钟源选择

SYSCLK 时钟源可以是 HSECLK、PLLCLK 或 HSICLK。

RCM_CFG1 的状态位可指示已准备好的时钟和已被选作的 SYSCLK 时钟源。

当进行系统复位后，HSICLK 振荡器被选作系统时钟，且当 PLL 直接或间接作为系统时钟时，该时钟源不能被停止。如果要切换 SYSCLK 时钟源，必须要等目标时钟源准备就绪（即目标时钟源稳定）。

6.3.9 CSS 时钟安全系统

为了防止由于外部晶振短路导致 MCU 无法正常运行的情况，MCU 可通过软件激活 CSS 时钟安全系统。安全系统被激活后，如果 HSECLK 振荡器直接或间接（作为 PLL 输入时钟且 PLL 作为系统时钟）被用作系统时钟，那么当 HSECLK 时钟发生故障时，外部 HSECLK 振荡器被关闭，系统时钟自动切换至 HSICLK，此时选择 HSECLK 为时钟输入且作为系统时钟输入源的 PLL 也会被关闭。

可通过软件激活 CSS，当 HSECLK 时钟出现故障时，产生 CSS 中断，并且 NMI 也自动产生。NMI 将被不断执行，直到 CSS 中断挂起位被清除。因此，在 NMI 的处理程序中必须通过设置 RCM_INT（时钟中断寄存器）里的 CSSCLR 位来清除 CSS 中断。

6.3.10 ADC 的时钟源选择

ADC 的时钟源通过 ADC_CFG2 控制。它能够选择 HSICLK14 或被 2/4 分频的 PCLK 作为时钟源。当 PCLK 作为 ADC 的时钟源时，HSICLK14 无法转接到 ADC 接口。

6.3.11 低功耗模式

PCLK 和 DMACLK 可以用软件禁止。

睡眠模式：

- 停止 CPU 时钟
- 可软件停止 Flash 和 RAM 接口时钟
- 所有连接 APB 总线的外设时钟被禁止时，硬件停止 AHB1/APB 桥时钟

停止模式和待机模式：

- 禁止所有 1.5V 电源域
- 禁止 PLLCLK、HSICLK、HSICLK14 和 HSECLK

深度睡眠模式：

- 通过设置 DBGMCU_CFG 中的 STOP_CLK_STS 位和 STANDBY_CLK_STS 位，可以对系统进行调试
- 系统通过中断（停止模式）或复位（待机模式）选择 HSICLK 作为 SYSCLK
- 如果正在进行 Flash 编程，则在完成所有编程操作后才会进入深度睡眠模式
- 如果正在使用 APB 域，则在完成所有操作后才会进入深度睡眠模式

6.4 寄存器地址映射

表格 23 RCM 寄存器地址映射

寄存器名	描述	偏移地址
RCM_CTRL1	时钟控制寄存器 1	0x00
RCM_CFG1	时钟配置寄存器 1	0x04
RCM_INT	时钟中断寄存器	0x08
RCM_APBRRST2	APB2 外设复位寄存器	0x0C
RCM_APBRRST1	APB1 外设复位寄存器	0x10
RCM_AHBCLKEN	AHB 外设时钟使能寄存器	0x14
RCM_APBCLKEN2	APB2 外设时钟使能寄存器	0x18
RCM_APBCLKEN1	APB1 外设时钟使能寄存器	0x1C
RCM_RTCCTRL	RTC 域控制寄存器	0x20
RCM_CSTS	控制/状态寄存器	0x24
RCM_AHRRST	AHB 外设复位寄存器	0x28
RCM_CFG2	时钟配置寄存器 2	0x2C
RCM_CFG3	时钟配置寄存器 3	0x30
RCM_CTRL2	时钟控制寄存器 2	0x34

6.5 寄存器功能描述

6.5.1 时钟控制寄存器 1 (RCM_CTRL1)

偏移地址：0x00

复位值：0x0000 XX83，X 代表未定义

访问：以字,半字和字节形式访问，无等待周期

位/域	名称	R/W	描述
0	HSIEN	R/W	<p>使能高速内部时钟（High Speed Internal Clock Enable）由软件置 1 或清 0。</p> <p>HSICKL 是 RC 振荡器，以下条件中的一个出现时，会硬件置 1：上电启动、软件复位、从待机模式唤醒、停机模式唤醒、外部高速时钟源（作为系统时钟或通过 PLL 提供系统时钟）发生故障。当 HSICKL 作为系统时钟源或通过 PLL 提供系统时钟时，该位不能清 0。</p> <p>0：关闭 HSICKL RC 振荡器 1：开启 HSICKL RC 振荡器</p>
1	HSIRDYFLG	R	<p>高速内部时钟就绪标志（High Speed Internal Clock Ready Flag）</p> <p>0：HSICKL RC 振荡器未稳定 1：HSICKL RC 振荡器已稳定</p>
2	保留		
7:3	HSITRM	R/W	<p>调整高速内部时钟（High Speed Internal Clock Trim）</p> <p>产品在出厂时会校准到 8MHz±1%，但随着温度、电压的变化而变化，可通过 HSITRM[4:0]调整 HSICKL RC 振荡器的频率。</p>
15:8	HSICAL	R	<p>校准内部高速时钟（High Speed Internal Clock Calibrate）</p> <p>在出厂时会校准到 8MHz±1%，在系统启动时，会将校准参数自动写入该寄存器。</p>
16	HSEEN	R/W	<p>使能高速外部时钟（High Speed External Clock Enable）</p> <p>进入待机或停机模式时，该位由硬件清 0，关闭 HSECLK；当 HSECLK 作为系统时钟源或通过 PLL 提供系统时钟时，该位不能清 0。</p> <p>0：HSECLK 关闭 1：HSECLK 开启</p>
17	HSERDYFLG	R	<p>高速外部时钟就绪标志（High Speed External Clock Ready Flag）</p> <p>HSECLK 稳定后有硬件置 1，软件清 0。</p> <p>0：HSECLK 未稳定 1：HSECLK 已稳定</p>
18	HSEBCFG	R/W	<p>配置高速外部时钟旁路模式（High Speed External Clock Bypass Configure）</p> <p>旁路模式指外部时钟作为 HSECLK 时钟源，否则谐振器作为 HSECLK 时钟源。</p> <p>0：非旁路模式 1：旁路模式</p>
19	CSSEN	R/W	<p>使能时钟安全系统（Clock Security System Enable）</p> <p>0：禁止 1：使能</p>
23:20	保留		
24	PLLEN	R/W	<p>使能 PLL（PLL Enable）</p> <p>当进入待机和停止模式时，该位由硬件清零；当 PLLCLK 已配置为（或者配置过程中）系统时钟的时钟源，该位不能清 0；其他情况可由软件置 1 或清 0。</p> <p>0：PLL 关闭 1：PLL 使能</p>

位/域	名称	R/W	描述
25	PLLRDYFLG	R	PLL 时钟就绪标志 (PLL Clock Ready Flag) PLL 锁定后由硬件置 1。 0: PLL 未锁定 1: PLL 锁定
31:26	保留		

6.5.2 时钟配置寄存器 1 (RCM_CFG1)

偏移地址: 0x04

复位值: 0x0000 0000

此寄存器所有位都由软件置位或清零。

访问: 以字,半字和字节形式访问, 0 到 2 个等待周期。

只有当访问发生在时钟切换时, 才会插入 1 或 2 个等待周期。

位/域	名称	R/W	描述
1:0	SCLKSEL	R/W	选择系统时钟时钟源 (System Clock Source Select) 在从停止或待机模式中返回时或直接或间接作为系统时钟的 HSECLK 出现故障时, 由硬件强制选择 HSICLK 作为系统时钟 (如果时钟安全系统已经启动) 00: HSICLK 作为系统时钟 01: HSECLK 作为系统时钟 10: PLLCLK 作为系统时钟 11: 保留
3:2	SCLKSELSTS	R	系统时钟时钟源选择状态 (System Clock Selection Status) 指示哪一个时钟源被作为系统时钟; 由硬件置 1 或清 0。 00: HSICLK 作为系统时钟 01: HSECLK 作为系统时钟 10: PLLCLK 输出作为系统时钟 11: 不可用
7:4	AHBPSC	R/W	配置 AHB 时钟预分频 (AHB Clock Prescaler Factor Configure) 控制 AHB 时钟的预分频系数。 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
10:8	APB1PSC	R/W	配置 APB1 时钟预分频系数 (APB1 Clock Prescaler Factor Configure) 来控制低速 APB1 时钟 (PCLK1) 的预分频系数。 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

位/域	名称	R/W	描述
13:11	保留		
14	ADCPSC	R/W	配置 ADC 时钟预分频 (ADCCLK Prescaler Factor Configure) 由 ADC 的配置寄存器相应位决定。
15	保留		
16	PLLSRCSEL	R/W	选择 PLL 时钟源 (PLL Clock Source Select) 只能在关闭 PLL 时才能改变此位。 0: HSICLK 2 分频后作为 PLL 时钟源 1: HSECLK 分频后作为 PLL 时钟源
17	PLLHSEPSC	R/W	配置作为 PLL 时钟源的 HSECLK 分频 (HSECLK Prescaler Factor for PLL Clock Source) 参考 RCM_CFG2 第 0 位。
21:18	PLLMULCFG	R/W	配置 PLL 倍频系数 (PLL Multiplication Factor Configure) 确定 PLL 倍频系数。只有在 PLL 关闭的情况下才可被写入。 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 16 倍频输出 注意: PLL 的输出频率不能超过 72MHz。
23:22	保留		
26:24	MCOSEL	R/W	选择主时钟输出 (Main Clock Output Select) 由软件置位或清零。 000: 没有时钟输出 001: HSICLK14 作为时钟输出 010: LSICLK 作为时钟输出 011: LSECLK 作为时钟输出 100: SYSCLK 作为时钟输出 101: HSICLK 作为时钟输出 110: HSECLK 作为时钟输出 111: PLLCLK2 分频后作为时钟输出
31:27	保留		

6.5.3 时钟中断寄存器 (RCM_INT)

偏移地址: 0x08

复位值：0x0000 0000

访问：以字,半字和字节形式访问，无等待周期。

位/域	名称	R/W	描述
0	LSIRDYFLG	R	LSICKLK 就绪中断标志 (LSICKLK Ready Interrupt Flag) LSICKLK 稳定且 LSIRDYEN 位被置 1 时，由硬件置 1；由软件置 1 LSIRDYCLR 清 0 该位。 0: 无 LSICKLK 就绪中断 1: 发生 LSICKLK 就绪中断
1	LSERDYFLG	R	LSECLK 就绪中断标志 (LSECLK Ready Interrupt Flag) LSECLK 稳定且 LSERDYEN 位被置 1 时，由硬件置 1；由软件置 1 LSERDYCLR 清 0 该位。 0: 无 LSECLK 就绪中断 1: 发生 LSECLK 就绪中断
2	HSIRDYFLG	R	HSICKLK 就绪中断标志 (HSICKLK Ready Interrupt Flag) HSICKLK 稳定且 HSIRDYEN 位被置 1 时，由硬件置 1；由软件置 1 HSIRDYCLR 清 0 该位。 0: 无 HSICKLK 就绪中断 1: 发生 HSICKLK 就绪中断
3	HSERDYFLG	R	HSECLK 就绪中断标志 (HSECLK Ready Interrupt Flag) HSECLK 稳定且 HSERDYEN 位被置 1 时，由硬件置 1；由软件置 1 HSERDYCLR 清 0 该位。 0: 无 HSECLK 就绪中断 1: 发生 HSECLK 就绪中断
4	PLLRDYFLG	R	PLL 就绪中断标志 (PLL Ready Interrupt Flag) PLL 稳定且 PLLRDYEN 位被置 1 时，由硬件置 1；由软件置 1 PLLRDYCLR 清 0 该位。 0: 无 PLL 锁定产生的时钟就绪中断 1: 发生 PLL 锁定产生的时钟就绪中断
5	HSI14RDYFLG	R	HSICKLK14 就绪中断标志 (HSICKLK14 Ready Interrupt Flag) 在内部高速时钟就绪且 HSI14RDYEN 位被置 1 时，由硬件置 1。 软件置 1HSI14RDYCLR 清 0 该位。 0: 无 HSECLK 失效产生的安全系统中断 1: 发生 HSECLK 失效产生的安全系统中断
6	保留		
7	CSSFLG	R	时钟安全系统中断标志 (Clock Security System Interrupt Flag) 在外部 4-16MHz 振荡器时钟出现故障时，由硬件置 1。 软件置 1CSSCLR 清 0 该位。 0: 无 HSE 时钟失效产生的安全系统中断 1: HSE 时钟失效导致了时钟安全系统中断
8	LSIRDYEN	R/W	使能 LSICKLK 就绪中断 (LSICKLK Ready Interrupt Enable) 使能或关闭内部 40KHz RC 振荡器就绪中断。 0: 禁止 1: 使能
9	LSERDYEN	R/W	使能 LSECLK 就绪中断 (LSECLK Ready Interrupt Enable) 使能外部 32KHz RC 振荡器就绪中断。 0: 禁止 1: 使能

位/域	名称	R/W	描述
10	HSIRDYEN	R/W	使能 HSICLK 就绪中断 (HSICLK Ready Interrupt Enable) 使能内部 8MHz RC 振荡器就绪中断。 0: 禁止 1: 使能。
11	HSERDYEN	R/W	使能 HSECLK 就绪中断 (HSECLK Ready Interrupt Enable) 使能外部 4-16MHz 振荡器就绪中断。 0: 禁止 1: 使能
12	PLLRDYEN	R/W	使能 PLL 就绪中断 (PLL Ready Interrupt Enable) 使能 PLL 就绪中断。 0: 禁止 1: 使能
13	HSI14RDYEN	R/W	使能 HSICLK14 就绪中断 (HSICLK14 Ready Interrupt Enable) 使能内部 14MHz RC 振荡器就绪中断。 0: 禁止 1: 使能。
15:14	保留		
16	LSIRDYCLR	W	清除 LSICLK 就绪中断 (LSICLK Ready Interrupt Clear) 清除 LSI 就绪中断标志位 LSIRDYFLG。 0: 无作用 1: 清除
17	LSERDYCLR	W	清除 LSECLK 就绪中断 (LSECLK Ready Interrupt Clear) 清除 LSE 就绪中断标志位 LSERDYFLG。 0: 无作用 1: 清除
18	HSIRDYCLR	W	清除 HSICLK 就绪中断 (HSICLK Ready Interrupt Clear) 清除 HSI 就绪中断标志位 HSIRDYFLG。 0: 无作用 1: 清除
19	HSERDYCLR	W	清除 HSECLK 就绪中断 (HSECLK Ready Interrupt Clear) 清除 HSE 就绪中断标志位 HSERDYFLG。 0: 无作用 1: 清除
20	PLLRDYCLR	W	清除 PLL 就绪中断 (PLL Ready Interrupt Clear) 清除 PLL 就绪中断标志位 PLLRDYFLG。 0: 无作用 1: 清除
21	HSI14RDYCLR	W	清除 HSICLK14 就绪中断 (HSICLK14 Ready Interrupt Clear) 清除 HSICLK14 就绪中断标志位 HSI14RDYFLG。 0: 无作用 1: 清除
22	保留		

位/域	名称	R/W	描述
23	CSSCLR	W	清除时钟安全系统中断（Clock Security System Interrupt Clear） 清除安全系统中断标志位 CSSFLG。 0: 无作用 1: 清除
31:24	保留		

6.5.4 APB 外设复位寄存器 2 (RCM_APBRS2)

偏移地址: 0x0C

复位值: 0x0000 0000

访问: 以字,半字和字节形式访问,无等待周期。

所有位都可以由软件置位或清 0。

位/域	名称	R/W	描述
0	SYSCFGRST	R/W	复位系统配置寄存器 (SYSCFG Reset) 0: 无作用 1: 复位
8:1	保留		
9	ADCRST	R/W	复位 ADC (ADC Reset) 0: 无作用 1: 复位 ADC
10	保留		
11	TMR1RST	R/W	复位 TMR1 定时器 (TMR1 Timer Reset) 0: 无作用 1: 复位
12	SPIRST	R/W	复位 SPI (SPI Reset) 0: 无作用 1: 复位
13	保留		
14	USART1RST	R/W	复位 USART1 (USART1 Reset) 0: 无作用 1: 复位
15	保留		
16	TMR7RST	R/W	复位 TMR7 (TMR7 Reset) 0: 无作用 1: 复位
21:17	保留		
22	DBGSRST	R/W	复位 Debug (Debug Reset) 0: 无作用 1: 复位
31:23	保留		

6.5.5 APB 外设复位寄存器 1 (RCM_APBRS1)

偏移地址: 0x10

复位值：0x0000 0000

访问：以字,半字和字节形式访问，无等待周期

位/域	名称	R/W	描述
0	TMR2RST	R/W	复位定时器 2 (Timer 2 Reset) 0: 无作用 1: 复位
1	TMR3RST	R/W	复位定时器 3 (Timer 3 Reset) 0: 无作用 1: 复位
3:2	保留		
4	TMR6RST	R/W	复位定时器 6 (Timer 6 Reset) 0: 无作用 1: 复位
7:5	保留		
8	TMR14RST	R/W	复位定时器 14 (Timer14 Reset) 0: 无作用 1: 复位
10:9	保留		
11	WWDTRST	R/W	复位窗口看门狗 (Window Watchdog Reset) 0: 无作用 1: 复位
16:12	保留		
17	USART2RST	R/W	复位 USART2 (USART2 Reset) 0: 无作用 1: 复位
20:18	保留		
21	I2CRST	R/W	复位 I2C (I2C Reset) 0: 无作用 1: 复位
24:22	保留		
25	CANRST	R/W	复位 CAN (CAN Reset) 0: 无作用 1: 复位
27:26	保留		
28	PMURST	R/W	复位电源接口 (Power Interface Reset) 0: 无作用 1: 复位
31:29	保留		

6.5.6 AHB 外设时钟使能寄存器 (RCM_AHBCLKEN)

偏移地址：0x14

复位值：0x0000 0014

访问：以字,半字和字节形式访问，无等待周期

所有位都可以由软件置位或清 0。

注：当外设时钟没有启用时，软件不能读出外设寄存器的数值，返回的数值始终是 0x0。

位/域	名称	R/W	描述
0	DMAEN	R/W	使能 DMA 时钟 (DMA Clock Enable) 0: 禁止 1: 使能
1	保留		
2	SRAMEN	R/W	使能 SRAM 时钟 (SRAM Interface Clock Enable) 使能睡眠模式时 SRAM 时钟。 0: 禁止 1: 使能
3	保留		
4	FMCEN	R/W	使能闪存接口电路时钟 (FMC Clock Enable) 使能睡眠模式时闪存接口电路时钟。 0: 禁止 1: 使能
5	保留		
6	CRCEN	R/W	使能 CRC 时钟 (CRC Clock Enable) 0: 禁止 1: 使能
16:7	保留		
17	PAEN	R/W	使能 I/O PortA 时钟 (I/O PortA Clock Enable) 0: 禁止 1: 使能
18	PBEN	R/W	使能 I/O PortB 时钟 (I/O PortB Clock Enable) 0: 禁止 1: 使能
19	PCEN	R/W	使能 I/O PortC 时钟 (I/O PortC Clock Enable) 0: 禁止 1: 使能
21:20	保留		
22	PFEN	R/W	使能 I/O PortF 时钟 (I/O PortF Clock Enable) 0: 禁止 1: 使能
23	保留		
24	M0CPEN	R/W	使能 M0CP 时钟 (M0CP Clock Enable) 0: 禁止 1: 使能
31:25	保留		

6.5.7 APB 外设时钟使能寄存器 2 (RCM_APBCLKEN2)

偏移地址: 0x18

复位值: 0x0000 0000

访问：以字，半字和字节形式访问

所有位都可以通过软件置位或清 0。

注：当外设时钟没有启用时，软件不能读出外设寄存器的数值，返回的数值始终是 0x0。

位/域	名称	R/W	描述
0	SCFGCOMPEN	R/W	使能 SYSCFG 时钟和 COMP 时钟 (SYSCFG Clock Enable) 0: 禁止 1: 使能
8:1	保留		
9	ADCEN	R/W	使能 ADC 接口时钟 (ADC Interface Clock Enable) 0: 禁止 1: 使能
10	保留		
11	TMR1EN	R/W	使能 TMR1 定时器时钟 (TMR1 Timer Clock Enable) 0: 禁止 1: 使能
12	SPIEN	R/W	使能 SPI 时钟 (SPI Clock Enable) 0: 禁止 1: 使能
13	保留		
14	USART1EN	R/W	使能 USART1 时钟 (USART1 Clock Enable) 0: 禁止 1: 使能
15	保留		
16	TMR7EN	R/W	使能 TMR7 定时器时钟 (TMR7 Timer Clock Enable) 0: 禁止 1: 使能
21:17	保留		
22	DBGEN	R/W	使能 Debug 时钟 (Debug Clock Enable) 0: 禁止 1: 使能
31:23	保留		

6.5.8 APB 外设时钟使能寄存器 1 (RCM_APBCLKEN1)

偏移地址：0x1C

复位值：0x0000 0000

访问：以字、半字和字节形式访问

通常无访问等待周期。但在 APB 总线上的外设被访问时，将插入等待状态直到 APB 外设访问结束。

所有位都可以通过软件置位或清 0。

注：当外设时钟没有启用时，软件不能读出外设寄存器的数值，返回的数值始终是 0x0。

位/域	名称	R/W	描述
0	TMR2EN	R/W	使能定时器 2 时钟 (Timer 2 Clock Enable) 0: 禁止 1: 使能
1	TMR3EN	R/W	使能定时器 3 时钟 (Timer 3 Clock Enable) 0: 禁止 1: 使能
3:2	保留		
4	TMR6EN	R/W	使能定时器 6 时钟 (Timer 6 Clock Enable) 0: 禁止 1: 使能
7:5	保留		
8	TMR4EN	R/W	使能定时器 4 时钟 (Timer4 Clock Enable) 0: 禁止 1: 使能
10:9	保留		
11	WWDTEN	R/W	使能窗口看门狗时钟 (Window Watchdog Clock Enable) 0: 禁止 1: 使能
16:12	保留		
17	USART2EN	R/W	使能 USART2 时钟 (USART 2 Clock Enable) 0: 禁止 1: 使能
20:18	保留		
21	I2CEN	R/W	使能 I2C1 时钟 (I2C1 Clock Enable) 0: 禁止 1: 使能
24:22	保留		
25	CANEN	R/W	使能 CAN 时钟 (CAN Clock Enable) 0: 禁止 1: 使能
27:26	保留		
28	PMUEN	R/W	使能电源接口时钟 (Power Interface Clock Enable) 0: 禁止 1: 使能
31:29	保留		

6.5.9 RTC 域控制寄存器 (RCM_RTCCTRL)

偏移地址: 0x20

复位值: 0x0000 0018, 只能由 RTC 域复位有效复位

访问: 以字、半字和字节形式访问, 0 到 3 等待周期

当连续对该寄存器进行访问时, 将插入等待状态。

注意：当 PMU_CTRL 中的 BPWEN 位被置 1 后，LSEEN、LSEBCFG、RTCSRCSEL 和 RTCCLKEN 才能进行改动。

位/域	名称	R/W	描述
0	LSEEN	R/W	使能 LSECLK (Low-Speed External Oscillator Enable) 0: 禁止 1: 使能
1	LSERDYFLG	R	LSECLK 就绪标志 (Low-Speed External Clock Ready Flag) LSECLK 稳定时硬件置 1，不稳定时硬件清 0。 0: 未就绪 1: 就绪
2	LSEBCFG	R/W	配置 LSECLK 为旁路模式 (Low-Speed External Clock Bypass Mode Configure) 旁路模式指外部时钟作为 LSECLK 时钟源，否则谐振器作为 LSECLK 时钟源。 0: 非旁路模式 1: 旁路模式
4:3	LSEDRVCFG	R/W	配置 LSECLK 振荡器驱动能力 (LSE Oscillator Drive Capability Configure) 有软件置位或清零，设置 LSECLK 振荡器的驱动能力 (晶体模式不被旁路)，复位 RTC 域时，该位被复原成缺省值。 00: 弱 01: 中低 10: 中高 11: 强
7:5	保留		
9:8	RTCSRCSEL	R/W	选择 RTC 时钟源 (RTC Clock Source Select) 先设置 RTCRST 位复位 RTC 域，再选择 RTC 时钟源，无法直接配置该寄存器修改。 00: 无时钟 01: LSECLK 作为 RTC 时钟 10: LSICLK 作为 RTC 时钟 11: HSECLK 在 32 分频后作为 RTC 时钟
14:10	保留		
15	RTCCLKEN	R/W	使能 RTC 时钟 (RTC Clock Enable) 0: 禁止 1: 使能
16	RTCRST	R/W	复位 RTC 域软件 (RTC Domain Software Reset) 由软件置 1 或清 0 0: 复位未激活 1: 复位 RTC 域 (只影响 LSECLK 振荡器、RTC 实时时钟和寄存器 RCM_RTCCTRL)
31:17	保留		

6.5.10 控制/状态寄存器 (RCM_CSTS)

偏移地址: 0x24

复位值: 0xXXX0 0000，除复位标志外由系统复位清除，复位标志只能由电源复位清除。

访问：以字、半字和字节形式访问，0 到 3 等待周期。

当连续对该寄存器进行访问时，将插入等待状态。

位/域	名称	R/W	描述
0	LSIEN	R/W	使能内部低速振荡器（Low-Speed Internal Oscillator Enable） 由软件置 1 或清 0。 0: 禁止 1: 使能
1	LSIRDYFLG	R	内部低速振荡器就绪（Low-Speed Internal Oscillator Ready Flag） LSICLK 稳定时硬件置 1，不稳定时硬件清 0。 0: 未就绪 1: 就绪
22:2	保留		
23	PWRRSTFLG	R	1.5V 区域复位标志（Reset Flag of The 1.5V Domain） 由软件置位，通过将 RSTFLGCLR 置位清除。
24	RSTFLGCLR	RT_W	清除复位标志（Reset Flag Clear） 由软件置位或清除复位标志，包括 RSTFLGCLR。 0: 无作用 1: 清除复位标志
25	OBRSTFLG	R	选项字节加载复位标志（Option Byte Loader Reset Flag） 当选项字节加载复位发生时由硬件置位，否则通过将 RSTFLGCLR 置位清除。 0: 没有发生复位 1: 复位发生
26	PINRSTFLG	R	引脚复位标志（PIN Reset Flag） 在引脚复位发生时由硬件置位，否则通过将 RSTFLGCLR 置位清除。 0: 没有发生复位 1: 复位发生
27	PODRSTFLG	R	上电/掉电复位发生标志（POR/PDR Reset Occur Flag） 由硬件置 1；由软件通过写 RSTFLGCLR 位清除。 0: 无上电/掉电复位发生 1: 发生上电/掉电复位
28	SWRSTFLG	R	软件复位发生标志（Software Reset Occur Flag） 由硬件置 1；由软件通过写 RSTFLGCLR 位清除。 0: 没有发生 1: 发生
29	IWDTRSTFLG	R	发生独立看门狗复位标志（Independent Watchdog Reset Occur Flag） 当独立看门狗复位发生在 V _{DD} 区域时由硬件置 1；由软件通过写 RSTFLGCLR 位清除。 0: 没有发生 1: 发生

位/域	名称	R/W	描述
30	WWDTRSTFLG	R	发生窗口看门狗复位标志 (Window Watchdog Reset Occur Flag) 当窗口看门狗复位发生时由硬件置 1; 由软件通过写 RSTFLGCLR 位清除。 0: 没有发生 1: 发生
31	LPWRRSTFLG	R	低功耗复位发生标志 (Low Power Reset Occur Flag) 当低功耗管理复位发生时由硬件置 1; 由软件通过写 RSTFLGCLR 位清除。 0: 没有发生 1: 发生

6.5.11 AHB 外设复位寄存器 (RCM_IORST)

偏移地址: 0x28

复位值: 0x0000 0000

访问: 以字,半字和字节形式访问, 无等待周期
由软件置 1 或清零。

位/域	名称	R/W	描述
16:0	保留		
17	PARST	R/W	复位 I/O 端口 PA (I/O Port A Reset) 0: 无效 1: 复位
18	PBRST	R/W	复位 I/O 端口 PB (I/O Port B Reset) 0: 无效 1: 复位
19	PCRST	R/W	复位 I/O 端口 PC (I/O Port C Reset) 0: 无效 1: 复位
21:20	保留		
22	PFRST	R/W	复位 I/O 端口 PF (I/O Port F Reset) 0: 无效 1: 复位
23	保留		
24	M0CPRST	R/W	复位 M0CP (M0CP Reset) 0: 无效 1: 复位
31:23	保留		

6.5.12 时钟配置寄存器 2 (RCM_CFG2)

偏移地址: 0x2C

复位值: 0x0000 0000

访问: 以字,半字和字节形式访问, 无等待周期

位/域	名称	R/W	描述
3:0	PLLDIVCFG	R/W	配置 PLLCLK 输入分频系数 (PLLCLK Input Division Factor Configure) 配置 PLLCLK 的输入时钟信号分频系数。 0000: 无分频 0001: 2 分频 0010: 3 分频 1111: 16 分频
31:4	保留		

6.5.13 时钟配置寄存器 3 (RCM_CFG3)

偏移地址: 0x30

复位值: 0x0000 0000

访问: 以字,半字和字节形式访问, 无等待周期

位/域	名称	R/W	描述
1:0	USART1SEL	R/W	选择 USART1 的时钟源 (USART1 Clock Source Select) 由软件置位或清零。默认值为 00。 00: PCLK 作为 USART1CLK 01: SYSCLK 作为 USART1CLK 10: LSECLK 作为 USART1CLK 11: HSICLK 作为 USART1CLK
3:2	保留		
4	I2CSEL	R/W	选择 I2C 的时钟源 (I2C Clock Source Select) 由软件置位或清零。默认值为 0。 0: HSICLK 作为 I2CCLK 1: SYSCLK 作为 I2CCLK
7:5	保留		
8	ADCSEL	R/W	选择 ADC 的时钟源 (ADC Clock Source Select) 保持复位值, HSICLK14 作为 ADCCLK 的异步时钟输入, ADCCLK 的时钟源由 ADC_CFG2 决定。
31:9	保留		

6.5.14 时钟控制寄存器 2 (RCM_CTRL2)

偏移地址: 0x34

复位值: 0xXX00 XX80, X 代表未定义

访问: 以字,半字和字节形式访问, 无等待周期

位/域	名称	R/W	描述
0	HSI14EN	R/W	使能 HSICLK14 (HSICLK14 Enable) 由软件置 1 或清 0。 0: 内部 14MHz 振荡器关闭 1: 内部 14MHz 振荡器开启

位/域	名称	R/W	描述
1	HSI14RDFLG	R	HSICLK14 准备标志 (HSICLK14 Ready Flag) 由硬件置位, 用于指示 HSI14 振荡器的状态。 0: 未准备好 1: 准备好
2	HSI14TO	R/W	ADC 启动 HSI14 (ADC Interface Turn On HSI14) ADC 接口可以启动 HSI14 振荡器, 由硬件置位或清零。 0: 能够启动 1: 不能启动
7:3	HSI14TRM	R/W	HSICLK14 调整 (HSICLK14 Trim) 产品在出厂时会校准到 $14\text{MHz} \pm 1\%$, 但随着温度、电压的变化而变化, 可通过 HSI14TRM 调整 HSI14 RC 振荡器的频率。
15:8	HSI14CAL	R	HSICLK14 校准 (HSICLK14 Calibrate) 在出厂时会校准到 $14\text{MHz} \pm 1\%$, 在系统启动时, 会将校准参数自动写入该寄存器。
31:16	保留		

7 电源管理单元 (PMU)

7.1 术语全称、缩写描述

表格 24 术语全称、缩写描述

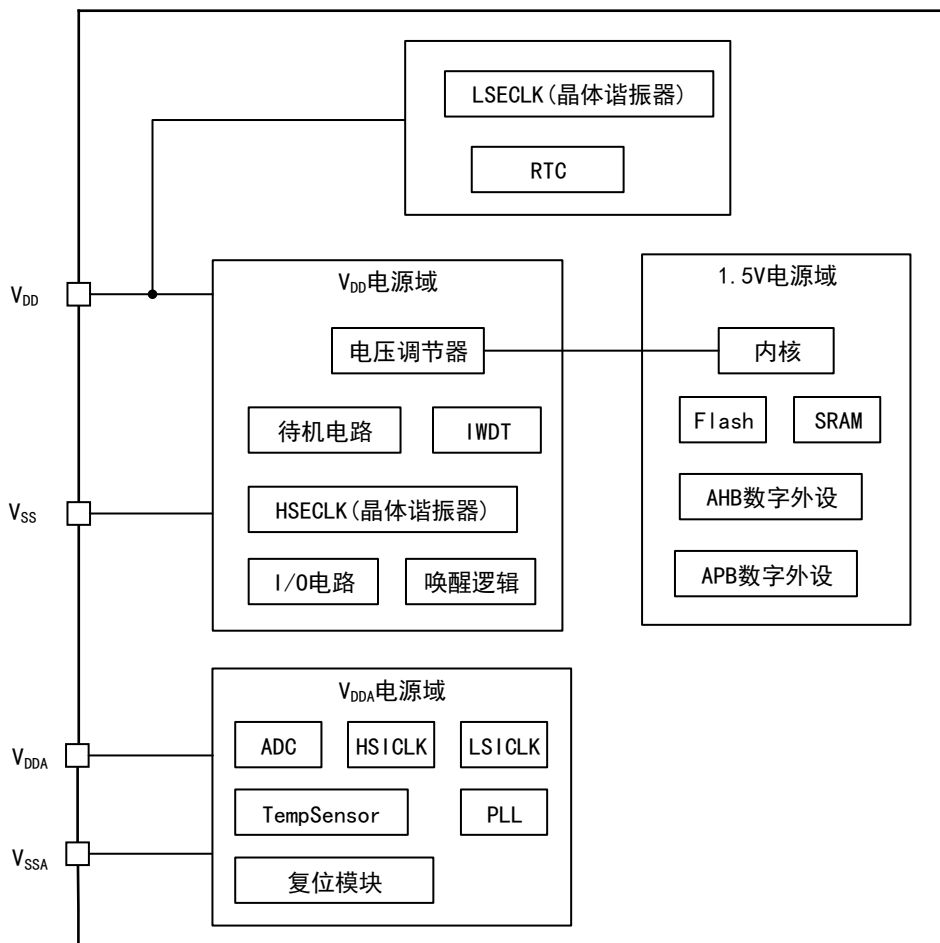
中文全称	英文全称	英文缩写
电源管理单元	Power Management Unit	PMU
上电复位	Power On Reset	POR
掉电复位	Power Down Reset	PDR

7.2 简介

电源是一个系统稳定运行的基础，工作电压为 2.0~3.6V，可以通过内置的电压调节器提供 1.5V 的电源。

7.3 结构框图

图 7 电源控制结构框图



7.4 功能描述

7.4.1 电源域

产品的电源域有： V_{DD} 电源域、 V_{DDA} 电源域、1.5V 电源域。

7.4.1.1 V_{DD} 电源域

通过 V_{DD}/V_{SS} 引脚供电，给电压调节器、待机电路、IWDT、HSECLK、I/O（除了 PC13、PC14、PC15 引脚）、唤醒逻辑供电。

电压调节器

给 1.5V 电源域供电，有以下几种工作模式：

- 正常模式：此模式下 1.5V 供电区域全功率运行
- 停止模式：此模式下 1.5V 供电区域工作在低功耗状态，所有时钟关闭，外设停止工作
- 待机模式：此模式下 1.5V 供电区域停止供电，除了备用电路，寄存器和 SRAM 内容都会丢失

7.4.1.2 V_{DDA} 电源域

通过 V_{DDA}/V_{SSA} 引脚供电，给 ADC、HSICLK、LSICLK、TempSensor、PLL、复位模块供电。

独立 ADC 电源

独立的 ADC 电源可以提高转换精度，具体电源引脚如下：

- V_{DDA} ：ADC 的电源引脚
- V_{SSA} ：独立电源地引脚

7.4.1.3 1.5V 电源域

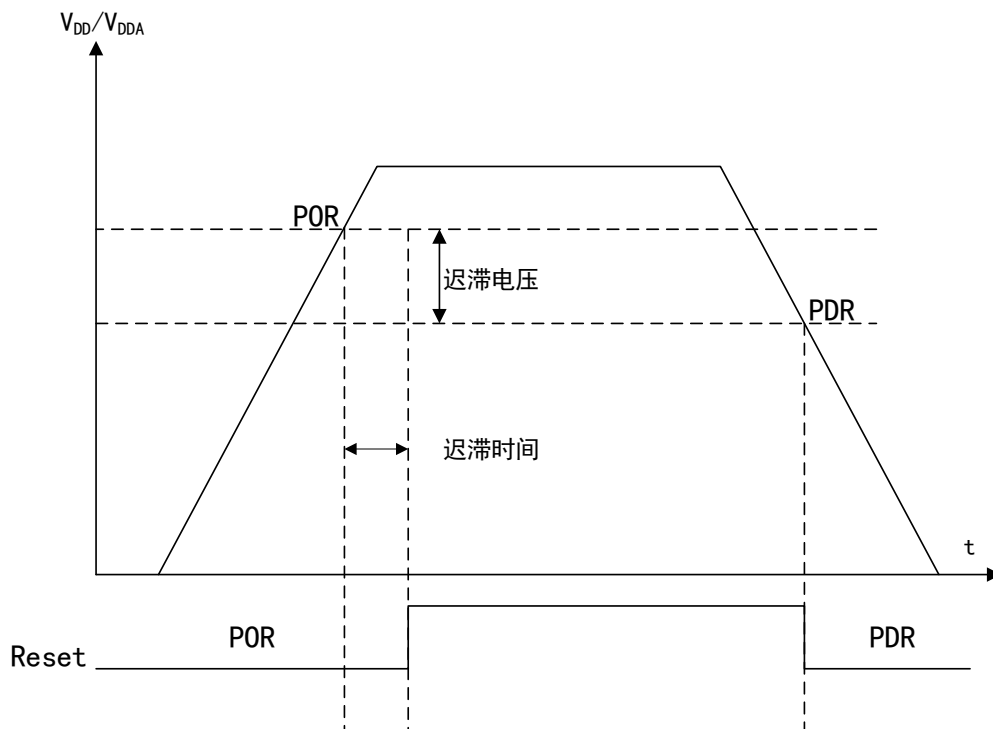
由电压调节器供电，给内核、Flash、SRAM、数字外设供电。

7.4.2 电源管理

7.4.2.1 上电复位与掉电复位（POR 与 PDR）

当检测到 V_{DD}/V_{DDA} 低于阈值电压 V_{POR} 和 V_{PDR} 时，芯片将会自动保持为复位状态，上电复位和掉电复位的波形图如下，POR、PDR、迟滞电压、迟滞时间请参考“数据手册”。

图 8 上电复位和掉电复位的波形图



7.4.3 功耗控制

7.4.3.1 低功耗模式降低功耗

低功耗模式有三种：睡眠模式、停止模式和待机模式。通过关闭内核、时钟源、设置调压器来降低功耗。

每种低功耗模式的功耗、唤醒启动时间、唤醒方式、唤醒后数据的保存存在差异；功耗越低，唤醒时间越长，唤醒方式越少，唤醒后保存的数据越少，用户可以根据需求选择最合适的低功耗模式。下图是三种低功耗模式的差异。

表格 25 “睡眠模式、停止模式和待机模式”差异

模式	说明	进入方式	唤醒方式	电压调节器	对 1.5V 区域时钟的影响	对 V _{DD} 区域时钟的影响
睡眠	Arm® Cortex®-M0+内核停止，所有外设包括内核的外设仍在工作	调用 WFI 命令	任一中断	开	只关闭内核时钟，对其它时钟以及 ADC 的时钟没有影响	无
		调用 WFE 命令	唤醒事件	开		无
停止	所有的时钟都已停止	PDDSCFG 和 LPDSCFG 位 +SLEEPDEEP 位+WFI 或 WFE	任一外部中断	开启或处于低功耗模式	关闭所有 1.5V 区域的时钟	HSICLK 和 HSECLK 的振荡器关闭

模式	说明	进入方式	唤醒方式	电压调节器	对 1.5V 区域时钟的影响	对 V _{DD} 区域时钟的影响
待机	1.5V 电源关闭	PDDSCFG 位 +SLEEPDEEP 位+WFI 或 WFE	WKUP 引脚的上升沿、RTC 闹钟事件、NRST 引脚上的外部复位、IWDT 复位	关		

睡眠模式

睡眠模式特点见下表：

表格 26 睡眠模式特点

特性	说明
进入	通过执行 WFI 或 WFE 指令时立即进入睡眠模式； 当 SLEEPONEXIT 置 0，且执行 WFI 或 WFE 指令，立即进入睡眠模式；当 SLEEPONEXIT 置 1，系统先退出中断程序，立即进入睡眠模式。
唤醒	若执行 WFI 指令进入睡眠模式，则通过任意中断唤醒；若执行 WFE 指令进入睡眠模式，则通过事件唤醒。
睡眠时	内核停止工作，所有外设仍在运行，且保存睡眠前内核寄存器、内存的数据。
唤醒延时	无
唤醒后	若通过中断唤醒，先进入中断，然后退出中断，之后才执行 WFI 指令后的程序，若通过事件唤醒，直接直接执行 WFE 指令后的程序。

停止模式

停止模式特点见下表：

表格 27 停止模式特点

特性	说明
进入	内核寄存器的 SLEEPDEEP 位置 1，寄存器 PMU_CTRL 中的 PDDSCFG 位置 0，然后执行 WFI 或 WFE 指令时立即进入停止模式； 寄存器 PMU_CTRL 的 LPDSCFG 位置 0 时调压器工作在正常模式，寄存器 PMU_CTRL 的 LPDSCFG 位置 1 时调节器工作在低功耗模式。
唤醒	若执行 WFI 指令进入睡眠模式，则通过任意中断唤醒；若执行 WFE 指令进入睡眠模式，则通过事件唤醒。
停止时	内核停止工作，外设也停止工作，保存停止前内核寄存器、内存的数据。
唤醒延时	HSICLK 振荡器唤醒时间+电压调节器从低功耗唤醒的时间。
唤醒后	若通过中断唤醒，先进入中断，然后退出中断，之后才执行 WFI 指令后的程序，若通过事件唤醒，直接直接执行 WFE 指令后的程序。

待机模式

待机模式特点见下表：

表格 28 待机模式

特性	说明
进入	内核寄存器的 SLEEPDEEP 位置 1，寄存器 PMU_CTRL 中的 PDDSCFG 位置 1，WUEFLG 位置 0，然后执行 WFI 或 WFE 指令时立即进入待机模式。
唤醒	通过 WKUP 引脚的上升沿，RTC 闹钟、唤醒、入侵、时间戳事件或 NRST 引脚外部复位及 IWDT 复位唤醒。
待机时	内核停止工作，外设也停止工作，内核寄存器、内存的数据会丢失。
唤醒延时	芯片复位的时间。
唤醒后	程序从头开始执行。

7.4.3.2 运行模式降低功耗

在运行模式，可通过降低系统时钟、关闭或者降低 APB/AHB 总线上的外设时钟降低运行模式功耗。

7.5 寄存器地址映射

表格 29 PMU 寄存器地址映射表

寄存器名	描述	偏移地址
PMU_CTRL	电源控制寄存器	0x00
PMU_CSTS	电源控制/状态寄存器	0x04

7.6 寄存器功能描述

7.6.1 电源控制寄存器 (PMU_CTRL)

偏移地址：0x00

复位值：0x0000 0000（从待机模式唤醒时清除）

位/域	名称	R/W	描述
0	LPDSCFG	R/W	低功耗深度睡眠配置 (Low Power Deep Sleep Configure) 配置在停机模式下调压器的工作状态。 0: 开启 1: 低功耗模式
1	PDDSCFG	R/W	配置掉电深度睡眠 (Pown Down Deep Sleep Configure) 在 CPU 进入深度睡眠下，在待机、停机模式下配置调压器的状态。 0: 进入停机模式时，调压器由 LPDSCFG 位控制 1: 进入待机模式
2	WUFLGCLR	RC_W1	清除唤醒标志 (Wakeup Flag Clear) 0: 无效 1: 通过写 1 在 2 个系统时钟周期后清除唤醒标志
3	SBFLGCLR	RC_W1	清除待机标志 (Standby Flag Clear) 0: 无效 1: 写 1 清除待机标志
7:4	保留		

位/域	名称	R/W	描述
8	BPWEN	R/W	使能写 RTC 区域 (RTC Domain Write Access Enable) RTC 区域指 RTC、RTC 寄存器, 复位后禁止写访问, 写 1 允许写访问。 0: 禁止写 1: 使能写
31:9	保留		

7.6.2 电源控制/状态寄存器 (PMU_CSTS)

偏移地址: 0x04

复位值: 0x0000 000X (从待机模式唤醒时不被清除)

与标准的 APB 读相比, 读此寄存器需要额外的 APB 周期

位/域	名称	R/W	描述
0	WUEFLG	R	唤醒事件产生标志 (Wakeup Event Flag) 该位由硬件设置, 标志是否在 WKUP 引脚上发生唤醒事件或者 RTC 闹钟唤醒事件。 0: 未发生 1: 已发生 注: 使能 WKUP 引脚, 当 WKUP 引脚已经是高电平时, 会检测到事件。
1	SBFLG	R	待机标志 (Standby Flag) 该位由硬件置 1, 只能由 POR/PDR (上电/掉电复位) 或设置电源控制寄存器 (PMU_CTRL) 的 SBFLGCLR 位清除。 0: 未进入过待机模式 1: 已进入过待机模式
7:2	保留		
9:8	WKUPCFGx	R/W	WKUPx 引脚配置 (WKUPxPin Configure) WKUPx 作为普通 I/O 时, WKUPx 引脚上的事件不能唤醒处在待机模式下的 CPU; 不当做普通 I/O 时, 才能唤醒 CPU。 0: 配置普通 I/O 1: 可唤醒 MCU 注: 在系统复位时清除这一位。
31:10	保留		

8 嵌套向量中断控制器（NVIC）

8.1 术语全称、缩写描述

表格 30 术语全称、缩写描述

中文全称	英文全称	英文缩写
不可屏蔽中断	Non Maskable Interrupt	NMI
嵌套向量中断控制器	Nested Vectored Interrupt Controller	NVIC

8.2 简介

产品中的 Cortex-M0+内核集成了嵌套向量中断控制器（Nested Vectored Interrupt Controller (NVIC)），它和内核紧密耦合，能高效、低延迟处理异常和中断、电源管理控制。更多关于 NVIC 的说明请参考《Cortex-M0+技术参考手册》。

8.3 主要特征

- (1) 32 个可屏蔽中断通道（不包括 16 个 Cortex-M0+中断线）
- (2) 4 个可编程的优先级（使用 2 位的中断优先级）
- (3) 低延时的异常和中断处理
- (4) 电源管理控制
- (5) 系统控制寄存器的实现

8.4 中断和异常向量表

表格 31 中断和异常向量表

名称	向量编号	优先级	向量地址	描述
-	-	-	0x0000_0000	保留
RST	-	-3	0x0000_0004	复位
NMI	-	-2	0x0000_0008	不可屏蔽中断
硬件故障（HardFault）	-	-1	0x0000_000C	各种硬件故障
SVCall	-	可设置	0x0000_002C	通用 SWI 指令调用的系统服务
PendSV	-	可设置	0x0000_0038	可挂起的系统服务
SysTick	-	可设置	0x0000_003C	系统滴答定时器
WWDT	0	可设置	0x0000_0040	窗口看门狗中断
PVD	1	可设置	0x0000_0044	PVD 中断

名称	向量编号	优先级	向量地址	描述
RTC	2	可设置	0x0000_0048	RTC 中断
FLASH	3	可设置	0x0000_004C	FLASH 中断
RCM	4	可设置	0x0000_0050	RCM 中断
EINT0_1	5	可设置	0x0000_0054	EINT 线[1:0]中断
EINT2_3	6	可设置	0x0000_0058	EINT 线[3:2]中断
EINT4_15	7	可设置	0x0000_005C	EINT 线[15:4]中断
M0CP	8	可设置	0x0000_0060	M0CP 中断
DMA_CH1	9	可设置	0x0000_0064	DMA 通道 1 中断
DMA_CH2_3	10	可设置	0x0000_0068	DMA 通道 2 和 3 中断
DMA_CH4_7	11	可设置	0x0000_006C	DMA 通道 4/5/6/7 中断
ADC_COMP	12	可设置	0x0000_0070	ADC 和 COMP 中断
TMR1_BRK_UP_TRG_COM	13	可设置	0x0000_0074	TMR1、BRK、UP、TRG 和 COM 中断
TMR1_CC	14	可设置	0x0000_0078	TMR1 捕获比较中断
TMR2	15	可设置	0x0000_007C	TMR2 中断
TMR3	16	可设置	0x0000_0080	TMR3 中断
TMR6	17	可设置	0x0000_0084	TMR6 中断
-	-	-	0x0000_0088	保留
TMR4	19	可设置	0x0000_008C	TMR4 中断
TMR7	20	可设置	0x0000_0090	TMR7 中断
-	-	-	0x0000_0094	保留
-	-	-	0x0000_0098	保留
I2C1	23	可设置	0x0000_009C	I2C1 中断
-	-	-	0x0000_00A0	保留
SPI1	25	可设置	0x0000_00A4	SPI1 中断
-	-	-	0x0000_00A8	保留
USART1	27	可设置	0x0000_00AC	USART1 中断
USART2	28	可设置	0x0000_00B0	USART2 中断
-	-	-	0x0000_00B4	保留
CAN	30	可设置	0x0000_00B8	CAN 中断
-	-	-	0x0000_00BC	保留

9 外部中断与事件控制器 (EINT)

9.1 简介

中断/事件分为内部中断/事件、外部中断/事件。在该手册中，外部中断指从 I/O 引脚输入信号引起的中断/事件，在中断向量表中指 EINTx；其它中断指内部中断/事件。

事件可分为硬件事件、软件事件。硬件事件是通过外部/内核硬件信号产生事件，软件事件是通过指令产生事件。

中断需经过中断处理函数实现需要处理的工作；事件不需要经过中断处理函数，可硬件触发预先设置的工作。外部事件例如可通过事件是 GPIO 输出脉冲，内部事件例如通过一个 TMR 的更新事件触发另一个 TMR 工作。

9.2 主要特征

- (1) 支持 28 个事件/中断请求
- (2) 可独立配置作为外部/内部事件请求的线
- (3) 每个事件/中断线都可独立屏蔽
- (4) 系统不处于停机模式时自动禁止内部的线
- (5) 每个外部事件/中断线都可独立触发
- (6) 每个外部中断线都有专用的状态位
- (7) 仿真所有外部事件中断

9.3 功能描述

9.3.1 “外部中断与事件”类别及差异点

“外部中断与事件”按照触发源、配置及执行过程，可分为：外部硬件中断、外部硬件事件、外部软件事件、外部软件中断，差异点见下表：

表格 32 “外部中断与事件”分类及差异点

名称	触发源	配置及执行过程
外部硬件中断	外部信号	(1) 设置触发方式，允许中断请求，使能对应外设中断线（在 NVIC 中使能）； (2) 当外部中断线上产生了和配置一致的边沿时，产生中断请求，对应的挂起位被置 1，在挂起寄存器对应位写 1，将清除该中断请求。
外部硬件事件	外部信号	(1) 设置触发方式，使能事件线； (2) 当外部事件线上产生了和配置一致的边沿时，产生 1 个事件请求脉冲，对应的挂起位不被置 1。

名称	触发源	配置及执行过程
外部软件事件	软件中断寄存器/发送事件(SEV)指令	(1) 使能事件线; (2) 对应事件线的软件中断事件寄存器写 1, 产生 1 个事件请求脉冲, 对应的挂起位不被置 1。
外部软件中断	软件中断寄存器	(1) 允许中断请求, 使能对应外设中断线 (在 NVIC 中使能); (2) 对应中断线的软件中断事件寄存器写 1, 产生中断请求, 对应的挂起位被置 1, 在挂起寄存器对应位写 1, 将清除该中断请求。

9.3.2 内核唤醒

使用 WFI、WFE 指令都可以使内核停止工作。使用 WFI 指令, 任一中断可唤醒内核; 使用 WFE 指令, 需通过事件唤醒。

使用中断唤醒, 会触发中断处理函数, 正常的中断配置即可唤醒内核。使用事件唤醒内核不触发中断处理函数, 会减少唤醒时间, 配置方法是:

- (1) 触发内部中断 (内部硬件事件) 但不触发中断处理函数唤醒
 - 使能外设中的一个内部中断, 但不使能 NVIC 中对应的中断, 避免触发中断处理函数
 - 在内核的系统控制器中使能 SEVONPEND 位, 执行 WFE 指令使内核进入睡眠
 - 产生中断唤醒内核, 当内核从 WFE 恢复后, 需要清除相应外设的中断挂起位和外设 NVIC 中断通道挂起位 (在 NVIC 中断清除挂起寄存器中)
- (2) 通过 EINT 线事件 (外部硬件事件) 唤醒
 - 配置 EINT 线为事件模式
 - 执行 WFE 指令使内核进入睡眠
 - 产生中断唤醒内核, CPU 从 WFE 恢复后, 因为对应事件线的挂起位没有被置位, 不必清除相应外设的中断挂起位或 NVIC 中断通道挂起位

9.3.3 外部中断与事件线映射

表格 33 外部中断与事件线映射

外部中断与事件通道名称	外部中断与事件线编号
PA0/PB0/PC0/PF0	EINT 0
PA1/PB1/PC1/PF1	EINT 1
...	...
PA15/PB15/PC15	EINT 15
PVD 输出	EINT 16
RTC 闹钟事件	EINT 17
保留	EINT 18
RTC 篡改和时间戳事件	EINT 19
保留	EINT 20

外部中断与事件通道名称	外部中断与事件线编号
COMP1 输出	EINT 21
COMP2 输出	EINT 22
内部 I2C1 唤醒事件	EINT 23
保留	EINT 24
内部 USART1 唤醒事件	EINT 25
保留	EINT 26
保留	EINT 27

9.4 寄存器地址映射

表格 34 外部中断/事件控制器寄存器地址映射

寄存器名	描述	偏移地址
EINT_IMASK	中断屏蔽寄存器	0x00
EINT_EMASK	事件屏蔽寄存器	0x04
EINT_RTEN	上升沿触发选择寄存器	0x08
EINT_FTEN	下降沿触发选择寄存器	0x0C
EINT_SWINTE	软件中断事件寄存器	0x10
EINT_IPEND	挂起寄存器	0x14

9.5 寄存器功能描述

9.5.1 中断屏蔽寄存器 (EINT_IMASK)

偏移地址: 0x00

复位值: 0x0F94 0000

位/域	名称	R/W	描述
27:0	IMASKx	R/W	屏蔽线 x 上的中断请求 (Interrupt Request Mask on Line x) 0: 屏蔽 1: 开放
31:28	保留		

9.5.2 事件屏蔽寄存器 (EINT_EMASK)

偏移地址: 0x04

复位值: 0x0000 0000

位/域	名称	R/W	描述
27:0	EMASKx	R/W	屏蔽线 x 上的事件请求 (Event Request Mask on Line x) 0: 屏蔽 1: 开放
31:28	保留		

9.5.3 使能上升沿触发选择寄存器 (EINT_RTEN)

偏移地址: 0x08

复位值: 0x0000 0000

位/域	名称	R/W	描述
17:0	RTENx	R/W	使能线 x 上的上升沿触发事件和中断 (Rising Trigger Event Enable and Interrupt of Line x) 0: 禁止 1: 使能
18	保留		
19	RTEN19	R/W	使能线 19 上的上升沿触发事件和中断 (Rising Trigger Event Enable and Interrupt of Line 19) 0: 禁止 1: 使能
20	保留		
22:21	RTENx	R/W	使能线 x 上的上升沿触发事件和中断 (Rising Trigger Event Enable and Interrupt of Line x) 0: 禁止 1: 使能
31:23	保留		

注意: 由于外部唤醒线是边沿触发, 所以这些线上不能有毛刺信号; 在写 EINT_RTEN 寄存器时, 若上升沿信号在外部中断线上则不能被识别, 置位挂起位也不会置位; 在同一中断线上, 上升沿触发和下降沿触发可以同时被设置。

9.5.4 使能下降沿触发选择寄存器 (EINT_FTEN)

偏移地址: 0x0C

复位值: 0x0000 0000

位/域	名称	R/W	描述
17:0	FTENx	R/W	使能线 x 上的下降沿触发事件和中断 (Falling Trigger Event Enable and Interrupt of Line x) 0: 禁止 1: 使能
18	保留		
19	FTEN19	R/W	使能线 19 上的下降沿触发事件和中断 (Falling Trigger Event Enable and Interrupt of Line 19) 0: 禁止 1: 使能
20	保留		
22:21	FTENx	R/W	使能线 x 上的下降沿触发事件和中断 (Falling Trigger Event Enable and Interrupt of Line x) 0: 禁止 1: 使能
31:23	保留		

注意：由于外部唤醒线是边沿触发，所以这些线上不能有毛刺信号；在写 EINT_FTEN 寄存器时，若上升沿信号在外部中断线上则不能被识别，置位挂起位也不会置位；在同一中断线上，上升沿触发和下降沿触发可以同时被设置。

9.5.5 软件中断事件寄存器 (EINT_SWINTE)

偏移地址：0x10

复位值：0x0000 0000

位/域	名称	R/W	描述
17:0	SWINTE _x	R/W	<p>线 x 上的软件中断 (Software Interrupt Event on Line x)</p> <p>软件置 1，对 EINT_IPEND 的对应位写入 1 清 0。</p> <p>当该位为 0 时，写 1 将置位 EINT_IPEND 的挂起位。若置位 EINT_IMASK (EINT_EMASK) 开放中断 (事件) 请求，则此时将产生一个中断 (事件)。</p> <p>0: 无作用</p> <p>1: 软件产生中断 (事件)</p>
18	保留		
19	SWINTE19	R/W	<p>线 19 上的软件中断 (Software Interrupt Event on Line 19)</p> <p>软件置 1，对 EINT_IPEND 的对应位写入 1 清 0。</p> <p>当该位为 0 时，写 1 将置位 EINT_IPEND 的挂起位。若置位 EINT_IMASK (EINT_EMASK) 开放中断 (事件) 请求，则此时将产生一个中断 (事件)。</p> <p>0: 无作用</p> <p>1: 软件产生中断 (事件)</p>
20	保留		
22:21	SWINTE _x	R/W	<p>线 x 上的软件中断 (Software Interrupt Event on Line x)</p> <p>软件置 1，对 EINT_IPEND 的对应位写入 1 清 0。</p> <p>当该位为 0 时，写 1 将置位 EINT_IPEND 的挂起位。若置位 EINT_IMASK (EINT_EMASK) 开放中断 (事件) 请求，则此时将产生一个中断 (事件)。</p> <p>0: 无作用</p> <p>1: 软件产生中断 (事件)</p>
31:23	保留		

9.5.6 中断挂起寄存器 (EINT_IPEND)

偏移地址：0x14

复位值：0xXXXX XXXX

位/域	名称	R/W	描述
17:0	IPEND _x	RC_W1	<p>发生线 x 上的中断挂起标志 (Interrupt Pending Occur of Line x Flag)</p> <p>是否发生可选择的触发请求</p> <p>0: 无</p> <p>1: 发生</p> <p>当在外部中断线上发生了 EINT_RTEN/EINT_FTEN 对应的边沿触发请求时，由硬件置 1；可通过改变边沿检测的极性清 0，或通过向该位写入 1 清 0。</p>
18	保留		

位/域	名称	R/W	描述
19	IPEND19	RC_W1	<p>发生线 19 上的中断挂起标志 (Interrupt Pending Occur of Line 19 Flag)</p> <p>是否发生可选择的触发请求</p> <p>0: 无</p> <p>1: 发生</p> <p>当在外部中断线上发生了 EINT_RTEN/EINT_FTEN 对应的边沿触发请求时, 由硬件置 1; 可通过改变边沿检测的极性清 0, 或通过向该位写入 1 清 0。</p>
20	保留		
22:21	IPENDx	RC_W1	<p>发生线 x 上的中断挂起标志 (Interrupt Pending Occur of Line x Flag)</p> <p>是否发生可选择的触发请求</p> <p>0: 无</p> <p>1: 发生</p> <p>当在外部中断线上发生了 EINT_RTEN/EINT_FTEN 对应的边沿触发请求时, 由硬件置 1; 可通过改变边沿检测的极性清 0, 或通过向该位写入 1 清 0。</p>
31:23	保留		

10 直接存储器存取（DMA）

10.1 术语全称、缩写描述

表格 35 术语全称、缩写描述

中文全称	英文全称	英文缩写
全局	Global	G
传输	Transfer	T
半	Half	H
完成	Complete	C
错误	Error	E
通道	Channel	CH
循环	Circular	CIR
外设	Peripheral	PER
增量	Increment	I
存储器	Memory	M
优先级	Priority	PRI
数量	Number	N
地址	Address	ADDR

10.2 简介

DMA（Direct Memory Access：直接存储器存取）在无须 CPU 干预的情况下，可实现外设与存储器或存储器与存储器之间数据的高速传输，从而节省 CPU 资源来做其他操作。

产品有一个 DMA 控制器共 7 个通道。每个通道可管理多个 DMA 请求，但每个通道同一时刻只能响应 1 个 DMA 请求。每个通道可设置优先级，仲裁器可根据通道的优先级协调各个 DMA 通道对应的 DMA 请求的优先级。

10.3 主要特征

- (1) DMA 有 7 个通道
- (2) 数据传输有三种：外设到存储器、存储器到外设、存储器到存储器
- (3) 每个通道都有连接专门的硬件 DMA 请求
- (4) 多个请求同时发生时支持软件优先级和硬件优先级
- (5) 每个通道都有 3 个事件标志和独立中断

- (6) 支持循环传输模式
- (7) 数据传输数目可编程，最大到 65535

10.4 功能描述

10.4.1 DMA 请求

若外设或存储器需要使用 DMA 传输数据，就必须先发送 DMA 请求，等待 DMA 同意之后才开始数据传输。

DMA 一共有 7 个通道，每个通道都连接着不同的外设，每个通道都有 3 个事件标志（DMA 半传输、DMA 传输完成和 DMA 传输出错），3 个事件标志的逻辑或成为一个单独的中断请求，且都支持软件触发。

多个外设请求同一个通道时，需要配置对应寄存器，开启或关闭每个外设的请求，以保证一个通道仅能开启一个外设请求。

表格 36 DMA 请求映射表

外设	通道 1	通道 2	通道 3	通道 4	通道 5	通道 6	通道 7
TMR1	—	TMR1_CH1	TMR1_CH2	TMR1_CH4 TMR1_TRIG TMR1_COM	TMR1_CH3 ⁽¹⁾ TMR1_UP	TMR1_CH1 ⁽²⁾ TMR1_CH2 ⁽²⁾ TMR1_CH3 ⁽²⁾	—
TMR2	TMR2_CH3	TMR2_UP	TMR2_CH2 ⁽¹⁾	TMR2_CH4 ⁽¹⁾	TMR2_CH1	—	TMR2_CH2 ⁽²⁾ TMR2_CH4 ⁽²⁾
TMR3	—	TMR3_CH3	TMR3_CH4 TMR3_UP	TMR3_CH1 ⁽¹⁾ TMR3_TRIG ⁽¹⁾	—	TMR3_CH1 ⁽²⁾ TMR3_TRIG ⁽²⁾	—
TMR4	—	—	—	—	TMR4_CH3	TMR4_CH4 TMR4_UP	TMR4_CH1 TMR4_TRIG
TMR7	—	—	—	—	TMR7_UP	—	—
TMR6	—	—	TMR6_UP	—	—	—	—
ADC	ADC ⁽¹⁾	ADC ⁽²⁾	—	—	—	—	—
SPI	—	SPI1_RX ⁽¹⁾	SPI1_TX ⁽¹⁾	SPI1_RX ⁽²⁾	SPI1_TX ⁽²⁾	—	—
USART	—	USART1_TX ⁽¹⁾	USART1_RX ⁽¹⁾	USART1_TX ⁽²⁾ USART2_TX	USART1_RX ⁽²⁾ USART2_RX	—	—
I2C	—	I2C1_TX ⁽¹⁾	I2C1_RX ⁽¹⁾	—	—	I2C1_TX ⁽²⁾	I2C1_RX ⁽²⁾

注：

- (1) 只有在 SYSCFG_CFG1 寄存器相应的位清 0 时该 DMA 请求映射到这个 DAM 通道。
- (2) 只有在 SYSCFG_CFG1 寄存器相应的重映射位置位时该 DMA 请求映射到这个 DAM 通道。

10.4.2 DMA 通道

10.4.2.1 传输数据可编程

DMA 传输的数据支持可编程，最大可达到 65535，通过配置 DMA_CHCFGx 寄存器的 PERSIZE 位和 MEMSIZE 位可设置外设和存储器的传输数据位宽。

10.4.2.2 传输宽度、对齐方式可编程

可编程数据传输宽度的 DMA 传输操作：

图 9 源为 8bits 目标为 8bits 的传输宽度

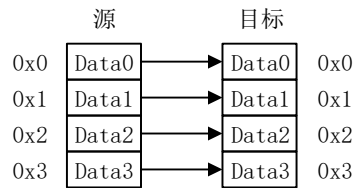


图 10 源为 8bits 目标为 16bits 的传输宽度

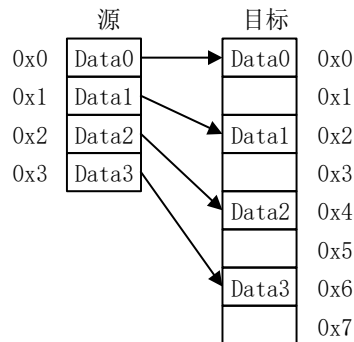


图 11 源为 8bits 目标为 32bits 的传输宽度

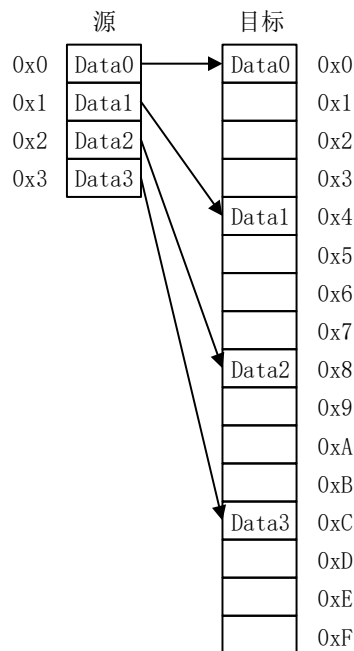


图 12 源为 32bits 目标为 8bits 的传输宽度

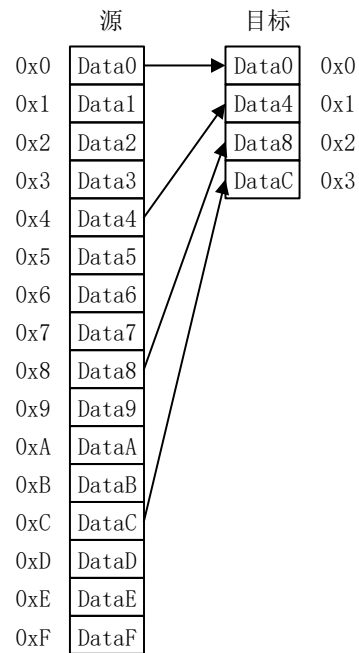
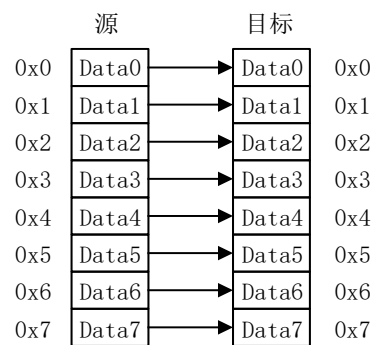


图 13 源为 16bits 目标为 16bits 的传输宽度



10.4.2.4 传输模式

有两种通道配置模式：非循环模式、循环模式。

非循环模式

数据传输结束后不再进行 DMA 操作，将重新开始新的 DMA 传输，在 DMA 通道不工作时寄存器 DMA_CHNDATAx 重新写入传输数值。

循环模式

数据传输结束后，寄存器 DMA_CHNDATAx 的内容被自动重新加载为之前配置的数值，外设地址寄存器 DMA_CHPADDRx 和存储器地址寄存器 DMA_CHMADDRx 也被重新加载为初始基地址。

配置方法如下：

- 配置寄存器 DMA_CHCFGx 的 CIRMODE 位置 1 开启循环模式；
- 此模式用来处理连续的外设请求，当数据传输的数目变成 0，将会自动恢复成初始值，持续进行 DMA 操作，直到 CIRMODE 位清 0 退出循环模式。

10.4.2.5 DMA 请求优先级设置

仲裁器

发生多个 DMA 通道请求时，需要用到仲裁器来管理先后响应的顺序。管理分两个阶段：第一阶段软件阶段分为最高、高、中等和低四个优先级；第二阶段硬件阶段，在软件优先级相同的情况下，通道编号越低优先级越高。

10.4.2.6 传输方向

支持三种方向：存储器到存储器、存储器到外设、外设到存储器。

如果对存储器执行的是写操作（目标地址），存储器包括内部 SRAM、EMMC 支持的外部 RAM（例如外部 SRAM）；如果对存储器执行的是读操作（源地址），地址包括内部 Flash、内部 SRAM。

“存储器到存储器”的配置举例如下：

- 配置寄存器 DMA_CHCFGx 的 M2MMODE 位启动存储器到存储器模式；
- 此模式下的 DMA 操作是在没有外设请求下进行的，配置寄存器 DMA_CHCFGx 的 CHEN 位置 1 通道开启后开始传输数据，直到传输数量寄存器 DMA_CHNDATAx 变为 0，传输结束。

10.4.3 中断

每一个 DMA 通道都有三种类型的中断事件，分别是：传输过半（HT）、传输完成（TC）、传输错误（TE）。

- （1）传输过半的中断事件标志位为 HTFLG，中断使能控制位为 HTINTEN

(2) 传输完成的中断事件标志位为 TCFLG，中断使能控制位为 TCINTEN

(3) 传输错误的中断事件标志位为 TERRFLG，中断使能控制位为 TERRINTEN

10.5 寄存器地址映射

表格 37 寄存器地址映射

寄存器名	描述	偏移地址
DMA_INTSTS	DMA 中断状态寄存器	0x00
DMA_INTFCLR	DMA 中断标志复位寄存器	0x04
DMA_CHCFGx	DMA 通道 x 配置寄存器	0x08+20 x
DMA_CHNDATAx	DMA 通道 x 传输数量寄存器	0x0C+20 x
DMA_CHPADDRx	DMA 通道 x 外设地址寄存器	0x10+20 x
DMA_CHMADDRx	DMA 通道 x 存储器地址寄存器	0x14+20 x

10.6 寄存器功能描述

10.6.1 DMA 中断状态寄存器 (DMA_INTSTS)

偏移地址：0x00

复位值：0x0000 0000

位/域	名称	R/W	描述
24,20,16, 12,8,4,0	GINTFLGx	R	通道 x 发生全局中断标志 (x=1..7) (Channel x Global Interrupt Occur Flag) 表示在通道上是否产生 TC、HT 或 TE 中断；这些位由硬件置 1，在 DMA_INTFCLR 的对应位上写 1 清 0。 0: 没有产生 1: 产生
25,21,17, 13,9,5,1	TCFLGx	R	通道 x 的全部传输完成标志 (x=1..7) (Channel x All Transfer Complete Flag) 表示在通道上是否产生传输完成中断(TC)；这些位由硬件置 1，在 DMA_INTFCLR 的对应位上写 1 清 0。 0: 未完成 1: 已完成
26,22,18, 14,10,6,2	HTFLGx	R	通道 x 的一半传输完成标志 (x=1..7) (Channel x Half Transfer Complete Flag) 表示在通道上是否产生半传输中断(HT)；这些位由硬件置 1，在 DMA_INTFCLR 的对应位上写 1 清 0。 0: 没有产生 1: 产生

位/域	名称	R/W	描述
27,23,19, 15,11,7,3	TERRFLGx	R	通道 x 发生传输错误标志 (x=1..7) (Channel x Transfer Error Occur Flag) 表示在通道上是否产生传输错误中断(TE); 这些位由硬件置 1, 在 DMA_INTFCLR 的对应位上写 1 清 0。 0: 没有产生 1: 产生
31:28	保留		

10.6.2 DMA 中断标志清除寄存器 (DMA_INTFCLR)

偏移地址: 0x04

复位值: 0x0000 0000

位/域	名称	R/W	描述
24,20,16,12, 8,4,0	GINTCLRx	R/W	清除通道 x 发生全局中断标志 (x=1..7) (Channel x Global Interrupt Occur Flag Clear) 清除中断状态寄存器中对应的 GINTFLG、TCFLG、HTFLG 和 TERRFLG 标志。 0: 无效 1: 清除 GINTFLG 标志
25,21, 17,13, 9,5,1	TCCLRx	R/W	清除通道 x 全部传输完成标志 (x=1..7) (Channel x Transfer Complete Clear) 清除中断状态寄存器中对应的 TCFLG 标志。 0: 无效 1: 清除 TCFLG 标志
26,22 18,14, 10,6,2	HTCLRx	R/W	清除通道 x 一半传输完成标志 (x=1..7) (Channel x Half Transfer Complete Clear) 清除中断状态寄存器中对应的 HTFLG 标志。 0: 无效 1: 清除 HTFLG 标志
27,23, 19,15, 11,7,3	TERRCLRx	R/W	清除通道 x 发生传输错误标志 (x=1..7) (Channel x Transfer Error Occur Clear) 清除中断状态寄存器中对应的 TERRFLG 标志。 0: 无效 1: 清除 TERRFLG 标志
31:28	保留		

10.6.3 DMA 通道 x 配置寄存器 (DMA_CHCFGx) (x=1..7)

偏移地址: 0x08+20 x (通道编号-1)

复位值: 0x0000 0000

位/域	名称	R/W	描述
0	CHEN	R/W	使能 DMA 通道 (DMA Channel Enable) 0: 禁止 1: 使能
1	TCINTEN	R/W	使能全部传输完成中断 (All Transfer Complete Interrupt Enable) 0: 禁止 1: 使能

位/域	名称	R/W	描述
2	HTINTEN	R/W	使能一般半传输完成中断 (Half Transfer Complete Interrupt Enable) 0: 禁止 1: 使能
3	TERRINTEN	R/W	使能传输错误发生中断 (Transfer Error Occur Interrupt Enable) 0: 禁止 1: 使能
4	DIRCFG	R/W	配置数据传输方向 (Data Transfer Direction Configure) 0: 从外设读至存储器 1: 从存储器读至外设
5	CIRMODE	R/W	使能循环模式 (Circular Mode Enable) 0: 禁止 1: 使能
6	PERIMODE	R/W	使能外设地址增量模式 (Peripheral Address Increment Mode Enable) 0: 禁止 1: 使能
7	MIMODE	R/W	使能存储器地址增量模式 (Memory Address Increment Mode Enable) 0: 禁止 1: 使能
9:8	PERSIZE	R/W	配置外设数据宽度 (Peripheral Data Size Configure) 00: 8 位 01: 16 位 10: 32 位 11: 保留
11:10	MEMSIZE	R/W	配置存储器数据宽度 (Memory Data Size Configure) 00: 8 位 01: 16 位 10: 32 位 11: 保留
13:12	CHPL	R/W	配置通道优先级 (Channel Priority Level Configure) 00: 低 01: 中 10: 高 11: 最高
14	M2MMODE	R/W	使能存储器到存储器模式 (Memory to Memory Mode Enable) 0: 禁止 1: 使能
31:15	保留		

10.6.4 DMA 通道 x 传输数量寄存器 (DMA_CHNDATAx) (x=1...7)

偏移地址: 0x0C+20 x (通道编号-1)

复位值: 0x0000 0000

位/域	名称	R/W	描述
15:0	NDATAT	R/W	设置数据传输数量 (Number of Data to Transfer Setup) 该寄存器指示要被传输的字节数目, 数据传输数量范围为 0 至 65535。 此寄存器只能在通道不工作时写入; 一旦通道被启用该寄存器变为只读, 指示剩余的要被传输的字节数目。 寄存器在每次 DMA 传输后递减; 数据传输完成, 寄存器变为 0 或者当该通道配置为自动重新加载模式时被自动重新加载为之前配置的数值; 若该寄存器为 0, 无论通道是否开启, 都不会发生任何数据传输。
31:16	保留		

10.6.5 DMA 通道 x 外设地址寄存器 (DMA_CHPADDRx) (x=1...7)

偏移地址: $0x10+20x$ (通道编号-1)

复位值: 0x0000 0000

当开启通道 (DMA_CHCFGx 的 CHEN=1) 时不能写该寄存器。

位/域	名称	R/W	描述
31:0	PERADDR	R/W	设置外设基地址 (Peripheral Basic Address Setup) 当 PERSIZE='01' (16 位), 不使用 PERADDR[0]位, 进行传输时会自动与 16 位地址对齐。 当 PERSIZE='10' (32 位), 不使用 PERADDR[1:0]位, 进行传输时会自动与 32 位地址对齐。

10.6.6 DMA 通道 x 存储器地址寄存器 (DMA_CHMADDRx) (x=1...7)

偏移地址: $0x14+20x$ (通道编号-1)

复位值: 0x0000 0000

当开启通道 (DMA_CHCFGx 的 CHEN=1) 时不能写该寄存器。

位/域	名称	R/W	描述
31:0	MEMADDR	R/W	设置存储器基地址 (Memory Basic Address Setup) 当 MEMSIZE='01' (16 位), 不使用 MEMADDR[0]位, 进行传输时会自动与 16 位地址对齐。 当 MEMSIZE='10' (32 位), 不使用 MEMADDR[1:0]位, 进行传输时会自动与 32 位地址对齐。

11 调试 MCU (DBGMCU)

11.1 术语全称、缩写描述

表格 38 术语全称、缩写描述

中文全称	英文全称	英文缩写
帧时钟	Frame Clock	FCLK
数据触发	Data Watchpoint Trigger	DWT
断点单元	Break Point Unit	BPU

11.2 简介

APM32F0xx 微控制器系列使用 Arm® Cortex®-M0+内核，Arm® Cortex®-M0+内核内含硬件调试模块，支持复杂的调试操作。在调试时该模块可以使运行的内核在断点时停下来，达到查询内核内部状态和系统外部状态的效果，并且在查询完成后恢复内核与外设的运行，继续执行程序。

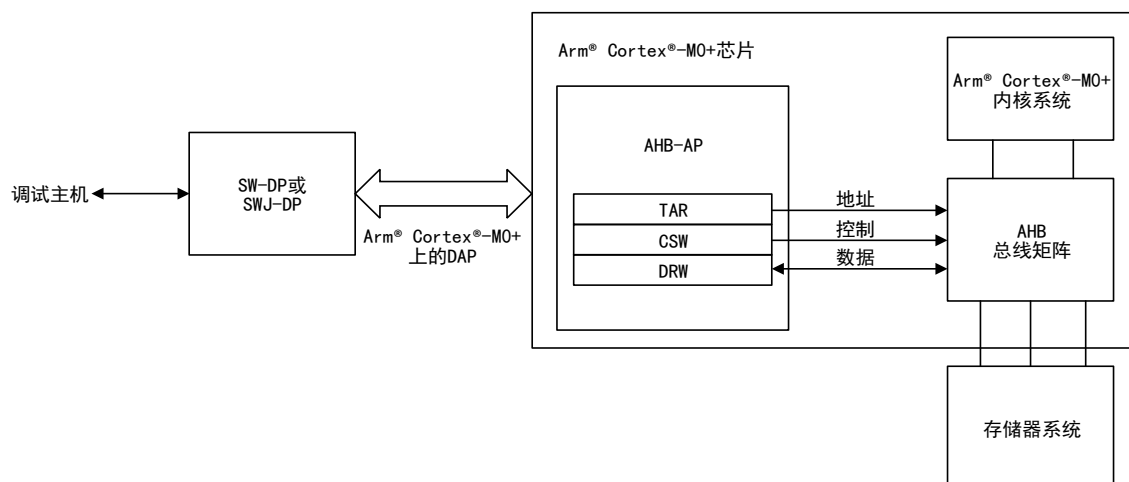
支持的调试接口：串行接口

注意：Arm® Cortex®-M0+内核内含的硬件调试模块是 Arm CoreSight 开发工具集的子集。更多 Arm® Cortex®-M0+内核的调试功能信息，请参考 Cortex®-M0+ (r1p1 版) 技术参考手册 (TRM) 和 CoreSight 开发工具集 (r1p0 版) TRM。

11.3 主要特征

- (1) 灵活的调试引脚分配
- (2) MCU 调试盒(支持低电源模式，控制外设时钟等)

图 16 APM32F0xx 级别和 Arm® Cortex®-M0+级别的调试框图



11.4 功能描述

- (1) 实现芯片的在线编程和调试
- (2) 利用 KEIL/IAR 等软件实现在线调试及下载编程
- (3) 灵活实现离线编程器的制作

11.5 寄存器地址映射

表格 39 DBGMCU 寄存器地址映射

寄存器名	描述	偏移地址
DBGMCU_IDCODE	调试 MCU 设备 ID 寄存器	0x00
DBGMCU_CFG	调试 MCU 配置寄存器	0x04
DBGMCU_APB1F	调试 MCU APB1 冻结寄存器	0x08
DBGMCU_APB2F	调试 MCU APB2 冻结寄存器	0x0C

11.6 寄存器功能描述

11.6.1 调试 MCU 设备 ID 寄存器 (DBGMCU_IDCODE)

地址: 0x00

只支持 32 位访问

复位值: 0x0001 0013

位/域	名称	R/W	描述
15:0	EQR	R	设备识别 (Equipment Recognition) 此字段表示设备 ID: 0x001
31:16	WVR	R	版本识别 (Wafer Version Recognition) 此字段指示设备的版本: 0x0013

11.6.2 设备 ID 寄存器 (DBGMCU_CFG)

此寄存器允许在调试中对 MCU 进行配置, 支持低功耗模式。

它是由 POR 异步重置 (而不是系统重置), 它可以由调试器在系统重置下编写。

如果调试主机不支持这些特性, 用户软件可以写入这些寄存器。

只支持 32 位访问

地址: 0x04

复位值: 0x0000 0000 (不受系统复位影响)

位/域	名称	R/W	描述
0	保留		
1	STOP_CLK_STS	R/W	配置调试停止模式 (Debug Stop Mode Configure) 0: FCLK 和 HCLK 都关闭的停止模式下, 所有的时钟都被时钟控制器禁止。退出停止模式时, 时钟配置和复位之后的配置一样 (由 8MHZ 的内部 RC 振荡器 HSICLK 提

位/域	名称	R/W	描述
			供时钟), 所以软件需要重新配置时钟控制器启动 PLL, 晶振等。 1: FCLK 和 HCLK 都开启的停止模式下, FCLK 和 HCLK 由内部 RC 振荡器提供。内部 RC 振荡器在停止模式下保持或活跃, 退出停止模式时, 软件必须重新配置时钟控制器启用 PLL, 晶振等。
2	STANDBY_CLK_STS	R/W	配置调试待机模式 (Debug Standby Mode) 0: FCLK 和 HCLK 都关闭时, 数字部分未通电, 从软件层面看, 处理一些表示了微控制器刚从待机状态退出时, 其他退出调试模式和复位是相同的 1: FCLK 和 HCLK 都开启时, 数字部分通电, 内部 RC 振荡器提供 FCLK 和 HCLK 时钟, 此外, 微控制器通过系统复位来退出待机模式和复位是一样的。
31:3	保留		

11.6.3 调试 MCU APB1 冻结寄存器 (DBGMCU_APB1F)

此寄存器用于在调试时配置 MCU。

涉及一些 APB 外设:

- 冻结定时器计数器
- 冻结 I2C SMBus 超时
- 冻结支持系统窗口监管机构 and 独立的看门狗计数器

此寄存器由 POR 异步重置 (而不是系统重置), 它可以由调试器在系统重置下编写。

只支持 32 位访问

地址: 0x08

复位值: 0x0000 0000 (不受系统复位影响)

位/域	名称	R/W	描述
0	TMR2_STS	R/W	内核停止时, 配置 TMR2 的工作状态 (ConfigureTimer2 Work Status When Core is in Halted) 内核停止时, TMR2 计数器是否继续工作 0: 继续工作 1: 停止工作
1	TMR3_STS	R/W	内核停止时, 配置 TMR3 的工作状态 (ConfigureTimer3 Work Status When Core is in Halted) 内核停止时, TMR3 计数器是否继续工作 0: 继续工作 1: 停止工作
3:2	保留		
4	TMR6_STS	R/W	内核停止时, 配置 TMR6 的工作状态 (ConfigureTimer6 Work Status When Core is in Halted) 内核停止时, TMR6 计数器是否继续工作 0: 继续工作 1: 停止工作
7:5	保留		

位/域	名称	R/W	描述
8	TMR4_STS	R/W	内核停止时，配置 TMR4 的工作状态（Configure Timer4 Work Status When Core is in Halted） 内核停止时，TMR4 计数器是否继续工作 0：继续工作 1：停止工作
9	保留		
10	RTC_STS	R/W	内核停止时，配置 RTC 的工作状态（Configure RTC Work Status When Core is in Halted） 内核停止时，RTC 计数器是否继续工作 0：继续工作 1：停止工作
11	WWDT_STS	R/W	内核停止时，配置窗口看门狗的工作状态（Configure Window Watchdog Work Status When Core is in Halted） 内核停止时，WWDT 是否继续工作 0：继续工作 1：停止工作
12	IWDT_STS	R/W	内核停止时，配置独立看门狗的工作状态（Configure Independent Watchdog Work Status When Core is in Halted） 内核停止时，IWDT 是否继续工作 0：继续工作 1：停止工作
20:13	保留		
21	I2C1_SMBUS_TIMEOUT_STS	R/W	内核停止时，配置 I2C1_SMBUS_TIMEOUT 的工作状态（Configure I2C1_SMBUS_TIMEOUT Work Status When Core is in Halted） 0：正常工作 1：冻结 SMBUS 的超时模式
24:22	保留		
25	CAN_STS	R/W	内核停止时，配置 CAN 的工作状态（Configure CAN Work Status When Core is in Halted） 内核停止时，CAN 是否继续工作 0：继续工作 1：停止工作
31:22	保留		

11.6.4 调试 MCU APB2 冻结寄存器（DBGMCU_APB2F）

此寄存器用于在调试时配置 MCU。

涉及一些 APB 外设：

- 冻结定时器计数器

此寄存器由 POR 异步重置（而不是系统重置），它可以由调试器在系统重置下编写。

只支持 32 位访问

地址：0x0C

复位值：0x0000 0000（不受系统复位影响）

位/域	名称	R/W	描述
10:0	保留		
11	TMR1_STS	R/W	内核停止时，配置 TMR1 的工作状态（ConfigureTimer1 Work Status When Core is in Halted） 内核停止时，TMR1 计数器是否继续工作 0：继续工作 1：停止工作
15:12	保留		
16	TMR7_STS	R/W	内核停止时，配置 TMR7 的工作状态（ConfigureTimer7 Work Status When Core is in Halted） 内核停止时，TMR7 计数器是否继续工作 0：继续工作 1：停止工作
31:17	保留		

12 通用/复用功能输入/输出引脚（GPIO/AFIO）

12.1 术语全称、缩写描述

表格 40 术语全称、缩写描述

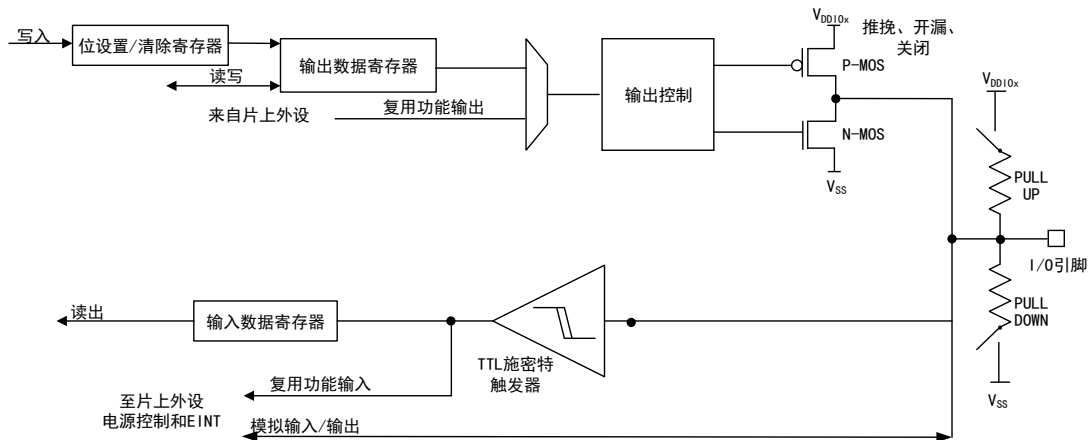
中文全称	英文全称	英文缩写
P 通道金属氧化物半导体	P-channel Metal Oxide Semiconductor	P-MOS
N 通道金属氧化物半导体	N-channel Metal Oxide Semiconductor	N-MOS

12.2 主要特征

- (1) 输入模式
 - 浮空输入
 - 上拉输入
 - 下拉输入
- (2) 输出模式
 - 推挽输出
 - 开漏输出
 - 可配置最大输出速率
- (3) 复用模式
 - 推挽复用功能
 - 开漏复用功能
- (4) 模拟模式
- (5) GPIO 都可以作为外部中断/唤醒线
- (6) 支持锁定 I/O 配置功能

12.3 结构框图

图 17 GPIO 结构框图



12.4 功能描述

GPIO 的每个引脚都可以通过软件配置上拉、下拉、浮空和模拟输入，或者推挽/开漏输出输入模式以及复用功能。所有的 GPIO 接口都具有外部中断能力。

12.4.1 复位期间和刚复位后的 IO 状态

GPIO 在复位期间和刚复位后，复用功能未开启，I/O 端口将会被配置为浮空输入模式。

在复位后，调试引脚处于 AF 上拉或下拉状态：

- PA14: SWCLK 置于下拉模式
- PA13: SWDIO 置于上拉模式

12.4.2 输入模式

在输入模式中可以设置为上拉、下拉、浮空和模拟输入。

当 GPIO 配置为输入模式时，所有的 GPIO 引脚内部都有一个内部弱上拉和弱下拉电阻，它们可以被激活也可以被断开。

上拉、下拉、浮空模式

在（上拉、下拉、浮空）输入模式中

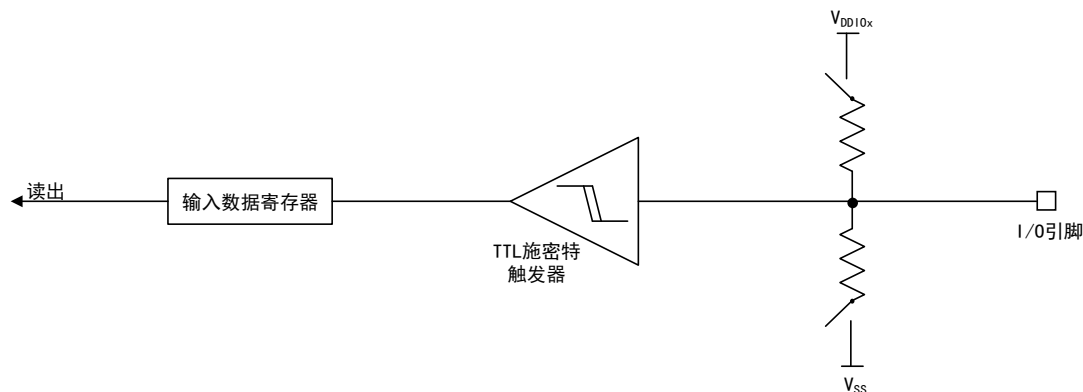
- 施密特触发器打开
- 禁止输出缓冲器
- 通过配置上拉/下拉寄存器 $GPIOx_PUPD$ 选择是否使用上拉/下拉电阻
- 输入数据寄存器 $GPIOx_IDATA$ 在每个 AHB 时钟周期捕捉 I/O 引脚上的数据。
- 通过输入数据寄存器 $GPIOx_IDATA$ 读取 I/O 状态

浮空输入模式的初始电平状态不确定，且易受外界干扰；连接设备时，由外部的

输入电平决定（阻抗非常高的除外）。

上拉/下拉输入模式的初始电平状态，如果是上拉则为高电平，如果为下拉则为低电平；连接设备时，由外部的输入电平及负载阻抗决定。

图 18 输入模式 I/O 结构



12.4.3 输出模式

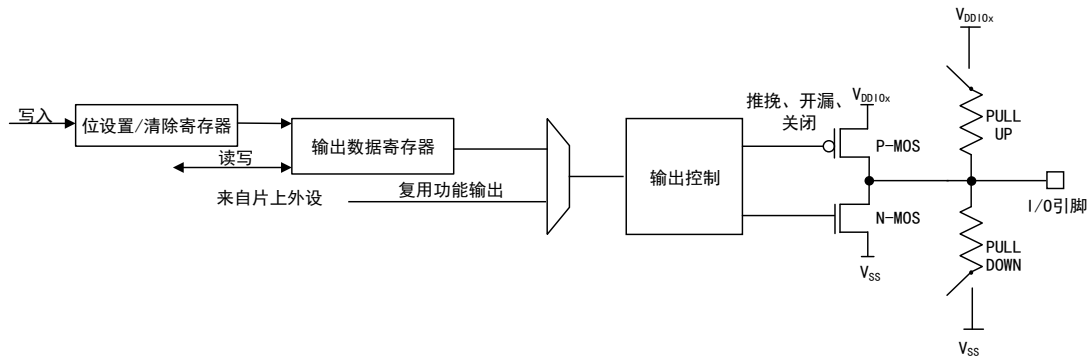
在输出模式中可以为推挽输出和开漏输出。

当 GPIO 配置为输出引脚时，可以配置端口的输出速度和选择输出驱动模式（推挽/开漏）。

在输出模式中：

- 施密特触发器打开
- 激活输出缓冲器
- 通过配置上拉/下拉寄存器 GPIOx_PUPD 选择是否使用上拉/下拉电阻
- 推挽模式：
 - 双 MOS 管以轮流方式工作，输出数据寄存器可控制 I/O 输出高低电平
 - 通过输出数据寄存器 GPIOx_ODATA 读取最后写入的值
- 开漏模式：
 - 只有 N-MOS 管工作，输出数据寄存器可控制 I/O 输出高阻态或低电平
 - 输入数据寄存器 GPIOx_IDATA 在每个 AHB 时钟周期捕捉 I/O 引脚上的数据
 - 通过输入数据寄存器 GPIOx_IDATA 读取 I/O 的实际状态

图 19 输出模式 I/O 结构



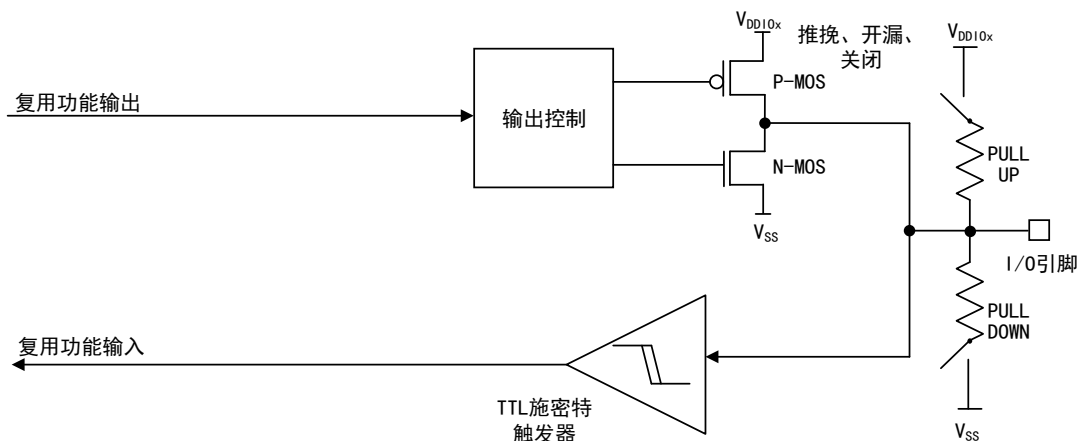
12.4.4 复用模式

在复用模式中可以为推挽复用和开漏复用

在推挽/开漏复用模式中：

- 打开输出缓冲器
- 由外设驱动输出缓冲器
- 激活施密特触发输入
- 通过配置上拉/下拉寄存器 GPIOx_PUPD 选择是否使用上拉/下拉电阻
- I/O 引脚上的数据在每个 AHB 时钟周期采样并存入端口输入状态寄存器
- 通过输入数据寄存器 GPIOx_IDATA 读取 I/O 的实际状态

图 20 复用模式 I/O 结构

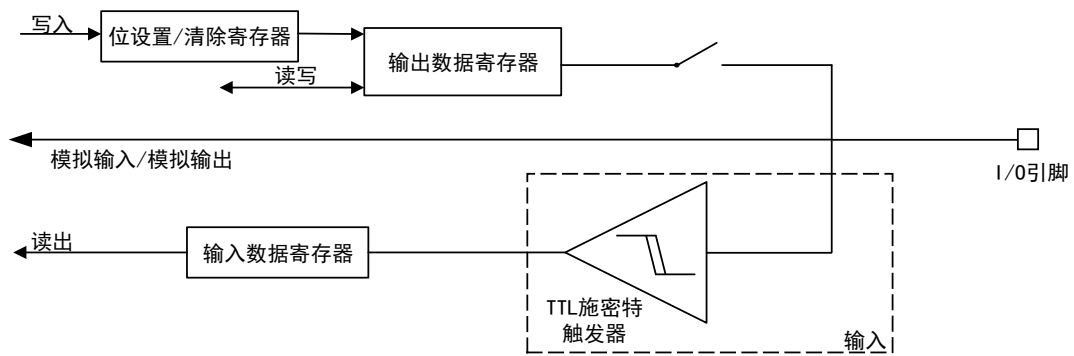


12.4.5 模拟模式

在模拟功能模式中：

- 禁止输出缓冲器
- 禁止施密特触发器输入，施密特触发器的输出值强置为 0
- 禁用弱上拉和下拉电阻
- 读取输入数据寄存器的值为 0

图 21 模拟功能 I/O 结构



12.4.6 外部中断/唤醒线

所有的 GPIO 端口都有外部中断功能，如果要使用外部中断线，端口必须要配置成输入模式。

12.4.7 I/O 数据位处理

GPIO 端口置位/复位寄存器（GPIOx_BSC）允许对输出数据寄存器（GPIOx_ODATA）的每一个位进行置位/复位操作。置位/复位寄存器的有效数据宽度是 GPIOx_ODATA 有效数据宽度的两倍。

GPIOx_BSC 中的任意位写 0 不会影响 GPIOx_ODATA 寄存器的值。如果 GPIOx_BSC 的 BS 和 BC 位同时置 1，则 BS 位优先。GPIOx_BSC 寄存器可以改变 GPIOx_ODATA 寄存器的相应位，GPIOx_ODATA 位可以直接从 GPIOx_BSC 寄存器进行访问。

GPIOx_ODATA 在用 GOIOx_BSC 寄存器置位或复位访问机制时，不需要通过软件关闭中断访问 GPIOx_ODATA。

12.4.8 复用功能与重映射

多路复用器

多路复用器用来将器件 I/O 口线连接内嵌的外设模块，并且在同一时刻上只能一对一。

每个 I/O 引脚配有一个多路复用器，多路复用器的复用功能输入多达 16 个，但是实际上最多用到 8 个（AF0-AF7），由 GPIOx_ALFL 和 GPIOx_ALFH 寄存器配置。当 I/O 引脚复位后，所有引脚口都连接到 AF0。

重映射

每个外设都有多个复用功能，但由于一个引脚只能选择一个复用功能输入，所以可以将外设的复用功能映射到其他 I/O 引脚上，即把复用功能信号重新分配一个引脚地址。

引脚的复用功能和重映射地址表见数据手册。

I/O 复用配置

当 I/O 口要连接到外设复用功能时，需要做以下调试：

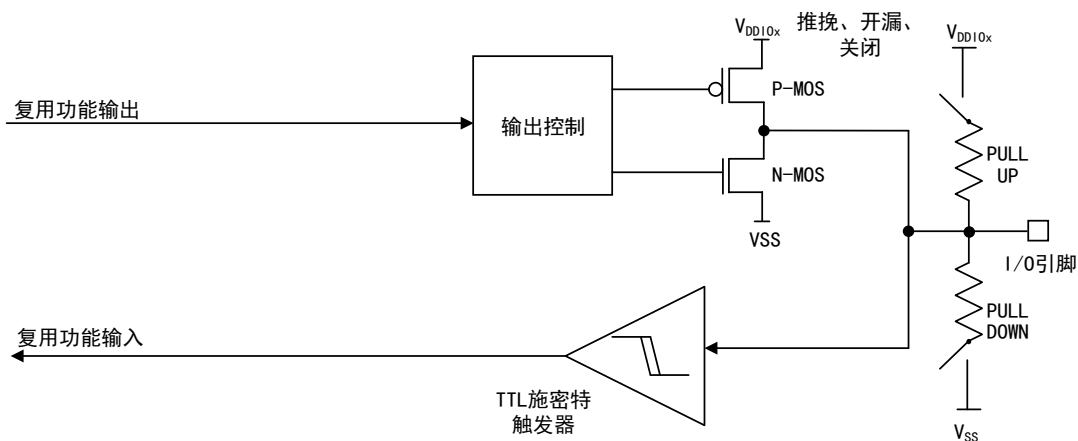
- 复位后，引脚配置为复用功能
- I/O 口配置为输入、输出或模拟输入
- I/O 口连接到定义的 AFx
- 配置引脚上拉/下拉和输出速度
- GPIOx_MODE 中配置 I/O 为复用功能

当 I/O 端口被配置为复用功能时，其输入输出模式：

- 打开输出缓冲器
- 由外设驱动输出缓冲器
- 激活施密特触发输入
- 通过配置上拉/下拉寄存器 GPIOx_PUPD 选择是否使用上拉/下拉电阻
- I/O 引脚上的数据在每个 AHB 时钟周期采样并存入端口输入状态寄存器
- 通过输入数据寄存器 GPIOx_IDATA 读取 I/O 的实际状态

复用模式 I/O 结构如下图所示：

图 22 复用模式下的 I/O 结构



12.4.9 GPIO 锁定功能

GPIO 的锁定机制可以保护 I/O 端口的配置。

通过对 GPIOx_LOCK 寄存器进行写序列（特定的）来冻结端口 A 和端口 B 的控制寄存器。如果要写 GPIOx_LOCK 寄存器，要发出一个特定的写/读序列。

通过配置锁定寄存器（GPIOx_LOCK）可以锁定 I/O 的配置，当一个端口位执行了锁定程序，到下一次复位之前，将不能再修改端口位的配置。

12.4.10 HSECLK 或 LSECLK 引脚用作 GPIO

通过配置 RCM_CTRL1 和 RCM_RTCCTRL 寄存器中的 HSEEN/LSEEN 设置 HSECLK/LSECLK RC 振荡器是否开启。

HSECLK/LSECLK RC 振荡器开启时，振荡器控制相关引脚，同时其相关引脚与

GPIO 配置无关；HSECLK/LSECLK RC 振荡器关闭时，相关振荡器可以用作普通 GPIO 接口。

12.4.11 GPIO 在 RTC 供电域下的使用

当核心电源域断电时，失去 PC13/PC14/PC15 GPIO 功能，此时，如果 GPIO 的配置没有被 RTC 配置，PC13/14/PC15 引脚被设置为模拟输入模式。

关于 RTC 控制 I/O 引脚的详细信息，参见 RTC 控制的 I/O 引脚。

12.5 寄存器地址映射

表格 41 GPIO 寄存器地址映射

寄存器名	描述	偏移地址
GPIOx_MODE	端口模式寄存器	0x00
GPIOx_OMODE	端口输出模式寄存器	0x04
GPIOx_OSSEL	端口输出速度寄存器	0x08
GPIOx_PUPD	端口上拉/下拉寄存器	0x0C
GPIOx_IDATA	端口位输入数据寄存器	0x10
GPIOx_ODATA	端口位输出清除寄存器	0x14
GPIOx_BSC	端口置位/复位寄存器	0x18
GPIOx_LOCK	端口锁定寄存器	0x1C
GPIOx_ALFL	端口复用功能低 8 位寄存器	0x20
GPIOx_ALFH	端口复用功能高 8 位寄存器	0x24
GPIOx_BR	端口复位寄存器	0x28

12.6 寄存器功能描述

12.6.1 端口模式寄存器 (GPIOx_MODE) (x=A..C, F)

偏移地址：0x00

复位值：0x2800 0000 端口 A

0x0000 0000 其他口

位/域	名称	R/W	描述
31:0	MODEy[1:0]	R/W	配置端口 x 引脚 y 模式 (y=0...15) (PortxPin y mode Configure) 00: 输入模式 (复位后的状态) 01: 通用输出模式 10: 复用功能模式 11: 模拟模式

12.6.2 端口输出模式寄存器 (GPIOx_OMODE) (x=A..C, F)

偏移地址: 0x04

复位值: 0x0000 0000

位/域	名称	R/W	描述
15:0	OMODEy	R/W	配置端口 x 引脚 y 输出模式 (y=0...15) (PortxPin y Output Mode Configure) 0: 推挽输出 (复位状态) 1: 开漏输出
31:16	保留		

12.6.3 端口输出速度寄存器 (GPIOx_OSSEL) (x=A..C, F)

偏移地址: 0x08

复位值: 0x0C00 0000 端口 A

0x0000 0000 其他端口

位/域	名称	R/W	描述
31:0	OSSELy[1:0]	R/W	选择端口 x 引脚 y 输出速度 (y=0...15) (PortxPin y Output Speed Select) x0: 低速 01: 中速 11: 高速 由软件写配置 I/O 口的速度

12.6.4 GPIO 口上拉/下拉寄存器 (GPIOx_PUPD) (x=A..C, F)

偏移地址: 0x0C

复位值: 0x2400 0000 端口 A

0x0000 0000 其他端口

位/域	名称	R/W	描述
31:0	PUPDy[1:0]	R/W	配置端口 x 引脚 y 上拉/下拉 (y=0...15) (PortxPin y Pull-up/Pull-down Configure) 这些位通过软件写操作来配置端口位的上拉/下拉 00: 禁止上拉/下拉 01: 上拉 10: 下拉 11: 复位

12.6.5 GPIO 端口输入数据寄存器 (GPIOx_IDATA) (x=A..C, F)

偏移地址: 0x10

复位值: 0x0000 XXXX

位/域	名称	R/W	描述
15:0	IDATAy	R	端口 x 引脚 y 输入数据 (y=0...15) (PortxPin y Input Data) 这些位只能进行读操作, 用来存储相应的 I/O 端口的输入值
31:16	保留		

12.6.6 GPIO 端口输出数据寄存器 (GPIOx_ODATA) (x=A..C, F)

偏移地址: 0x14

复位值: 0x0000 0000

位/域	名称	R/W	描述
15:0	ODATy	R/W	端口 x 引脚 y 输出数据 (y=0...15) (PortxPin y Output Data) 可以通过软件进行读写操作 对于原子位设置/设置, 可以通过写入 GPIOx_BSC 或 GPIOx_BR 寄存器来单独设置 ODATy 位
31:16	保留		

12.6.7 GPIO 端口置位/复位寄存器 (GPIOx_BSC) (x=A..C, F)

偏移地址: 0x18

复位值: 0x0000 0000

位/域	名称	R/W	描述
15:0	BSy	W	端口 x 引脚 y 的设置位 (y=0...15) (PortxPin y Set Bit) 这些位只能进行写操作, 读这些位时返回 0x0000 的数值。 这些位用来影响对应的 ODATy 位 0: 无影响 1: 设置对应的 ODATy 位
31:16	BCy	W	端口 x 引脚 y 复位位 (y=0...15) (PortxPin y Reset Bit) 这些位只能进行写操作, 读这些位时返回 0x0000 的数值。 这些位用来影响对应的 ODATy 位 0: 无影响 1: 对应的 ODATy 位清 0 若 BSy 位和 BCy 位同时设置, BSy 有优先权

12.6.8 GPIO 端口锁定寄存器 (GPIOx_LOCK) (x=A..B)

该寄存器保护 GPIO 的配置在程序运行期间误修改; 若再次修改 GPIO 配置, 需系统复位后才能修改。配置 GPIO 锁定功能时, 需对该寄存器执行指定的序列才能启动 GPIO 锁定功能。

偏移地址: 0x1C

复位值: 0x0000 0000

位/域	名称	R/W	描述
15:0	LOCKy	R/W	配置端口 x 引脚 y 的锁定位 (y=0...15) (PortxLock bit y Configure) 0: 不锁定端口 x 引脚 y 的配置 1: 锁定端口 x 引脚 y 的配置 这些位可以进行读写操作, 但是只能在 LOCKKEY=0 时写入。

位/域	名称	R/W	描述
16	LOCKKEY	R/W	<p>锁定键 (LOCK Key)</p> <p>该位决定端口配置锁键位是否被激活</p> <p>0: 不激活</p> <p>1: 激活, GPIOx_LOCK 寄存器锁定直到下一次 MUC 复位产生。</p> <p>锁定键写序列:</p> <p>写 LOCK[16]=1+LOCK[15:0]</p> <p>写 LOCK[16]=0+LOCK[15:0]</p> <p>写 LOCK[16]=1+LOCK[15:0]</p> <p>读 LOCK</p> <p>读 LOCK[16]=1 (此读操作可以选择, 要确认是否激活锁键)</p> <p>注意:</p> <p>(1) 在操作锁键的写入序列期间, LOCKy 的值不能改变。</p> <p>(2) 操作锁键写入序列中的任何错误将中止锁键。</p> <p>(3) 在端口任意位上的第一个锁序列之后, LOCKKEY 位上的任何读访问都将返回“1”, 直到下一个 MCU 重置或外围设备重置。</p>
31:17			保留

12.6.9 GPIO 复用功能低 8 位寄存器 (GPIOx_ALFL) (x=A...C, F)

偏移地址: 0x20

复位值: 0x0000 0000

位/域	名称	R/W	描述
31:0	ALFSEly	R/W	<p>选择端口 x 引脚 y 的复用功能 (y=0...7) (PortxPin y Alternate Function Select)</p> <p>这些位可以通过软件进行读操作, 用来配置端口的复用功能。</p> <p>ALFSEly 选择:</p> <p>0000: AF0</p> <p>0001: AF1</p> <p>0010: AF2</p> <p>0011: AF3</p> <p>0100: AF4</p> <p>0101: AF5</p> <p>0110: AF6</p> <p>0111: AF7</p> <p>1000: 保留</p> <p>1001: 保留</p> <p>1010: 保留</p> <p>1011: 保留</p> <p>1100: 保留</p> <p>1101: 保留</p> <p>1111: 保留</p> <p>1110: 保留</p>

12.6.10 GPIO 复用功能高 8 位寄存器 (GPIOx_ALFH) (x=A...C, F)

偏移地址: 0x24

复位值: 0x0000 0000

位/域	名称	R/W	描述
31:0	ALFSELY	R/W	<p>选择端口 x 引脚 y 的复用功能 (y=8...15) (PortxPin y Alternate Function Select)</p> <p>这些位可以通过软件进行读操作，用来配置端口的复用功能。</p> <p>ALFSELY 选择:</p> <ul style="list-style-type: none"> 0000: AF0 0001: AF1 0010: AF2 0011: AF3 0100: AF4 0101: AF5 0110: AF6 0111: AF7 1000: 保留 1001: 保留 1010: 保留 1011: 保留 1100: 保留 1101: 保留 1111: 保留 1110: 保留

12.6.11 GPIO 端口复位寄存器 (GPIOx_BR) (x=A...C, F)

偏移地址: 0x28

复位值: 0x0000 0000

位/域	名称	R/W	描述
15:0	BRy	W	<p>配置端口 x 引脚 y 复位 (y=0...15) (PortxPin y Reset Configure)</p> <p>这些位只能进行写操作，读这些位时返回值为 0x0000。</p> <p>这些位用来影响对应的 ODATA</p> <ul style="list-style-type: none"> 0: 无影响 1: 对应的 ODATA 位清 0
31:16	保留		

13 定时器概述

13.1 术语全称、缩写描述

表格 42 术语全称、缩写描述

中文全称	英文全称	英文缩写
定时器	Timer	TMR
更新	Update	U
请求	Request	R
事件	Event	EV
捕捉	Capture	C
比较	Compare	C
长度	Length	LEN

13.2 定时器类别及主要差异

在本次系列产品中，一共可分为三类定时器：高级定时器、通用定时器和基本定时器（看门狗定时器在其它章节中描述）。

高级定时器包含了通用定时器和基本定时器的功能，高级定时器是有四个捕获/比较通道，支持定时功能、输入捕获和输出比较功能、刹车以及互补输出功能，是一个 16 位可以向上/下计数的定时器。

通用定时器的功能比高级定时器简单，主要差异点在通道总数、互补输出通道组数、刹车功能。

而基本定时器是一个只能实现定时功能、没有外部接口的定时器。

产品包含的定时器主要差异见下表：

表格 43 产品包含的定时器主要差异

项目	具体内容/类别	高级定时器			通用定时器		基本定时器	
		TMR1	TMR2	TMR3/4	TMR6	TMR7		
时基单元	计数器	16 位	32 位	16 位	16 位			
	预分频器	16 位	16 位	16 位	16 位			
	计数模式	向上 向下 中央对齐	向上 向下 中央对齐	向上 向下 中央对齐	向上			
通道	输入通道	4	4	4	0			
	捕获比较通道	4	4	4	0			
	输出通道	8	4	4	0			

项目	具体内容/类别	高级定时器	通用定时器		基本定时器
	互补输出通道	3 组	0	0	0
功能	产生 DMA 请求	可以	可以	可以	可以
	PWM 模式	有	有	有	无
	单脉冲模式	有	有	有	无
	强制输出模式	有	有	有	无
	死区插入	有	无	无	无

定时器术语

表格 44 引脚定义术语

名称	描述
TMRx_ETR	定时器 x 外部触发信号
TMRx_CH1、TMRx_CH2、TMRx_CH3、 TMRx_CH4	定时器 x 通道 1/2/3/4
TMRx_CHyN	定时器 x 互补输出通道 y
TMRx_BKIN	定时器 x 刹车信号

表格 45 内部信号定义术语

名称	描述
ETR	TMRx_ETR 外部触发信号
ETRF	外部触发滤波
ETRP	外部触发分频
-	
ITR, ITR0, ITR1	内部触发
TRGI	时钟/触发器/从模式控制器的触发输入
TIF_ED	定时器输入滤波边缘检测
-	
CK_PSC	分频时钟
CK_CNT	计数器时钟
PSC	预分频器
CNT	计数器
AUTORLD	自动装载寄存器
-	
TIx, TI1	定时器输入
TIF, TI1F	定时器输入滤波

名称	描述
TI1_ED	定时器输入边缘检测
TIxFPx, TI1FP1	定时器输入滤波极性
ICx, IC1	输入捕获
ICxPS, IC1PS	输入捕获预分频
TRC	触发捕获
BRK	刹车信号
-	
OCx, OC1	定时器输出比较通道
OCxREF, OC1REF	输出比较参考信号
-	
TGI	触发中断
BI	刹车中断
CCxI, CC1I	捕获/比较中断
UEV	更新事件
UIFLG	更新中断标志

14 高级定时器 (TMR1)

14.1 简介

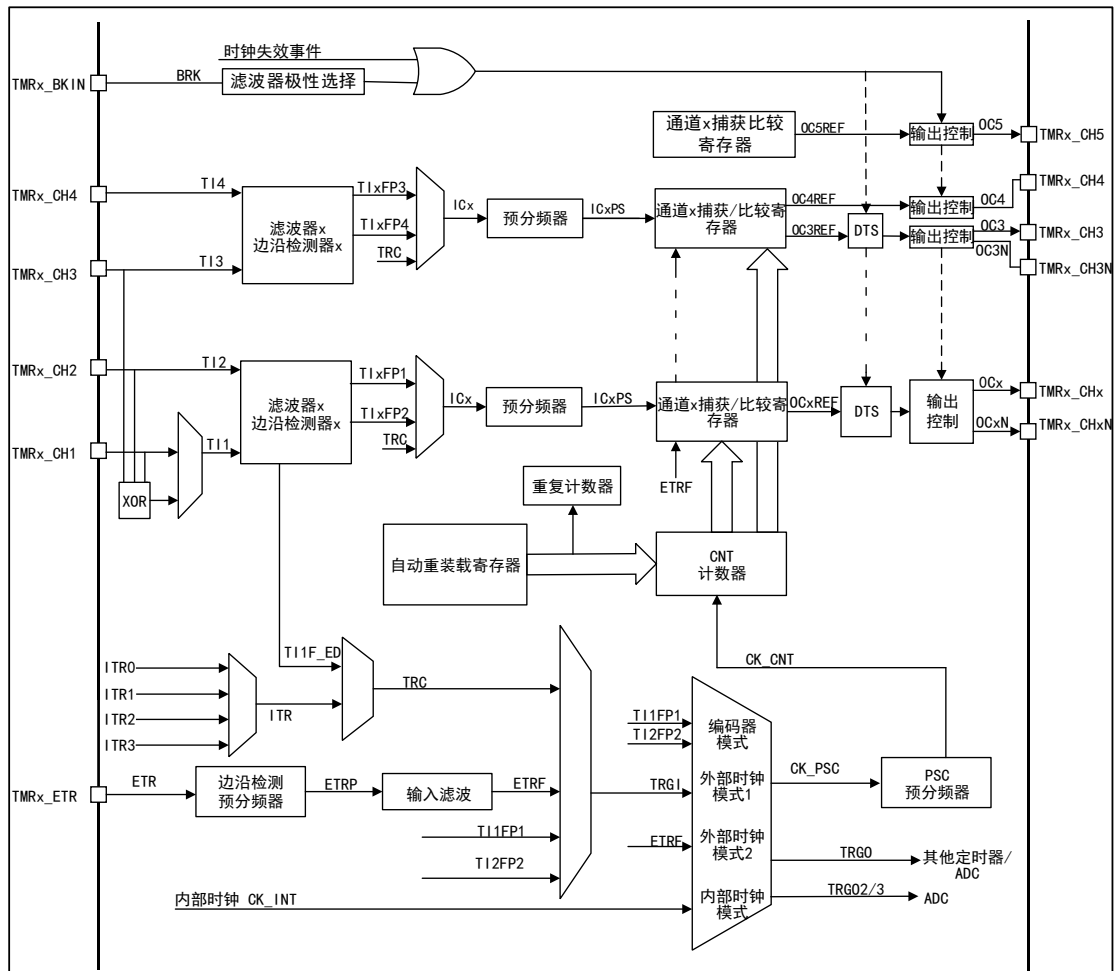
高级定时器 TMR1 以时基单元为核心，拥有输入捕获、输出比较和刹车输入等功能，含有一个 16 位的自动装载计数器。高级定时器含有互补输出、重复计数以及可编程的死区插入等功能，更加适合用于电机的控制。

14.2 主要特征

- (1) 时基单元
 - 计数器：16 位计数器，可以向上计数，向下计数，中央对齐计数
 - 预分频器：16 位可编程预分频器
 - 重复计数器：16 位的重复计数器
 - 自动重装载功能
- (2) 时钟源选择
 - 内部时钟
 - 外部输入
 - 外部触发
 - 内部触发
- (3) 输入捕获功能
 - 计数功能
 - PWM 输入模式（脉冲宽度、频率、占空比测量）
 - 编码器接口模式
- (4) 输出比较功能
 - PWM 输出模式
 - 强制输出模式
 - 单脉冲模式
 - 互补输出和死区插入
- (5) 定时功能
- (6) 刹车功能
- (7) 定时器的主/从模式控制器
 - 定时器之间可以同步和级联
 - 支持多种从模式、同步信号
- (8) 中断输出和 DMA 请求事件
 - 更新事件（计数器上/下溢出，计数器初始化）
 - 触发事件（计数器启动、停止、内/外部触发）
 - 捕获/比较事件
 - 刹车信号输入事件

14.3 结构框图

图 23 TMR1 结构框图



14.4 功能描述

14.4.1 时钟源选择

高级定时器一共有四种时钟源

内部时钟

是来自 RCM 的 TMRx_CLK，即定时器本身的驱动时钟，当禁止从模式控制器，则预分频的时钟源 CK_PSC 由内部时钟 CK_INT 驱动。

外部时钟模式 1

来自定时器自身的输入通道 T11/2/3/4，经过极性选择和滤波以后生成的触发信号，连接到从模式控制器，进而控制计数器的工作。其中通道 1 的输入经过上升沿、下降沿双沿检测而生成脉冲信号进行逻辑相或以后的信号就是 T11F_ED

信号，即 TIF_ED 双边沿信号。特别的是 PWM 输入只能由 TI1/2 输入。

外部时钟模式 2

来自于外部触发接口（ETR）经过极性选择、分频、滤波以后的信号，经过触发输入选择器，连接到从模式控制器，从而控制计数器的工作。

内部触发输入

设置定时器工作于从模式，时钟源为其他定时器的输出信号，此时钟源没有滤波，可以实现定时器之间的同步或级联。主模式的定时器可以对从模式定时器执行复位、启动、停止或提供时钟。

14.4.2 时基单元

高级定时器里的时基单元包含四个寄存器

- 计数器寄存器（CNT）16 位
- 自动重载寄存器（AUTORLD）16 位
- 预分频器（PSC）16 位
- 重复次数寄存器（REPCNT）8 位

计数器 CNT

高级定时器中的计数器中一共有三种计数模式

- 向上计数模式
- 向下计数模式
- 中央对齐模式

向上计数模式

通过配置控制寄存器（TMRx_CTRL1）中的 CNTDIR 位，设置为向上计数模式。

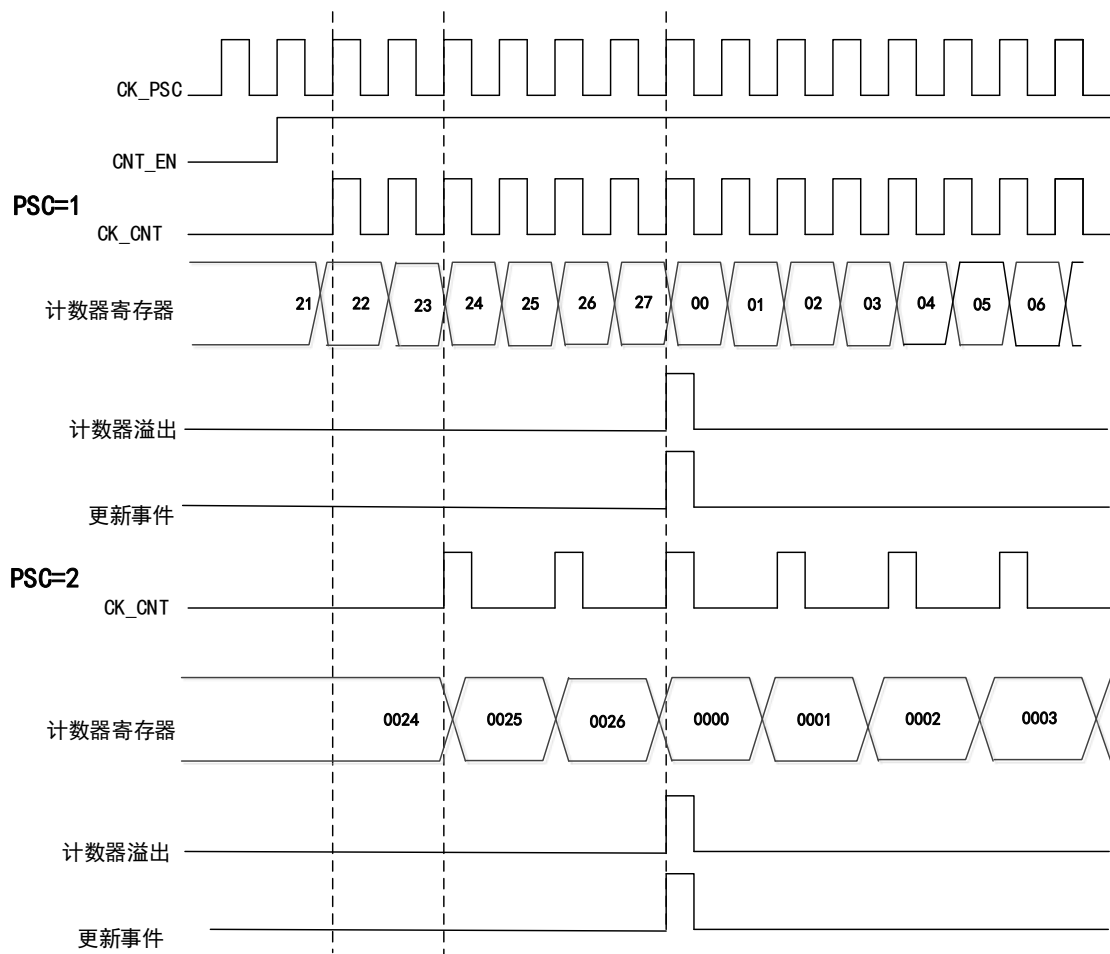
当计数器处于向上计数模式时，计数器从 0 开始向上计数，当每来一个脉冲计数器就会增加 1，一直到计数器（TMRx_CNT）值与自动重载

（TMRx_AUTORLD）的值相等时，计数器会再次从 0 开始计数，此时产生一个计数器向上溢出事件，其中自动重载的值（TMRx_AUTORLD）是提前写入的。

当计数器溢出时，会产生更新事件，此时重复计数的影子寄存器、自动重载的影子寄存器和预分频的缓冲区都将会被更新。可以通过配置控制寄存器 TMRx_CTRL1 中的 UD 位，禁止更新事件。

下图为向上计数模式下，分频因子为 1 或 2 的时序图

图 24 向上计数模式下，分频因子为 1 或 2 的时序图



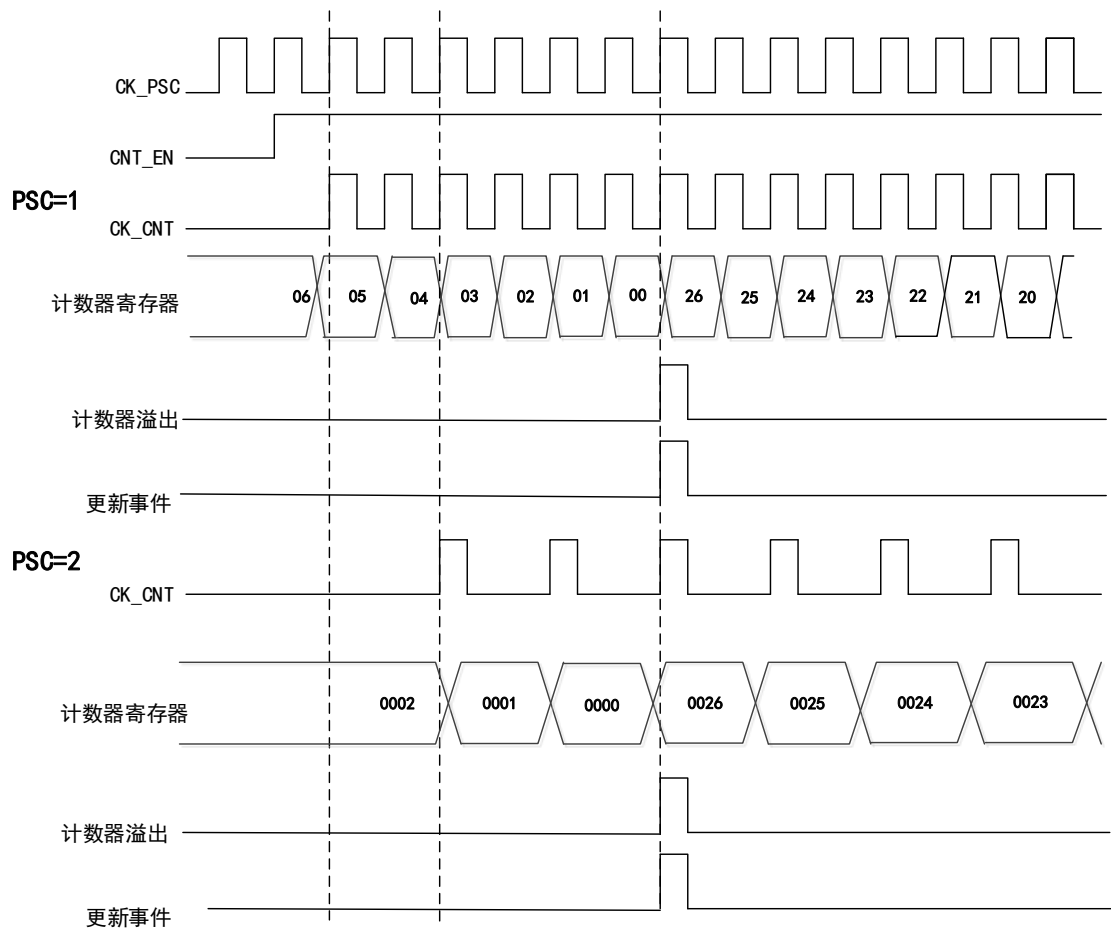
向下计数模式

通过配置控制寄存器（TMRx_CTRL1）中的 CNTDIR 位，设置为向下计数模式。

当计数器处于向下计数模式时，计数器从自动重载的值（TMRx_AUTORLD）开始向下计数，当每来一个脉冲计数器就会减 1，一直减到 0 时，计数器会重新从（TMRx_AUTORLD）开始计数，与此同时便会产生一个计数器向下溢出事件，其中自动重载的值（TMRx_AUTORLD）是提前写入的。

计数器溢出时，会产生更新事件，此时，重复计数的影子寄存器、自动重载的影子寄存器和预分频的缓冲区都将会被更新。可以配置 TMRx_CTRL1 寄存器中的 UD 位，禁止更新事件。

图 25 向下计数模式下，分频因子为 1 或 2 的时序图

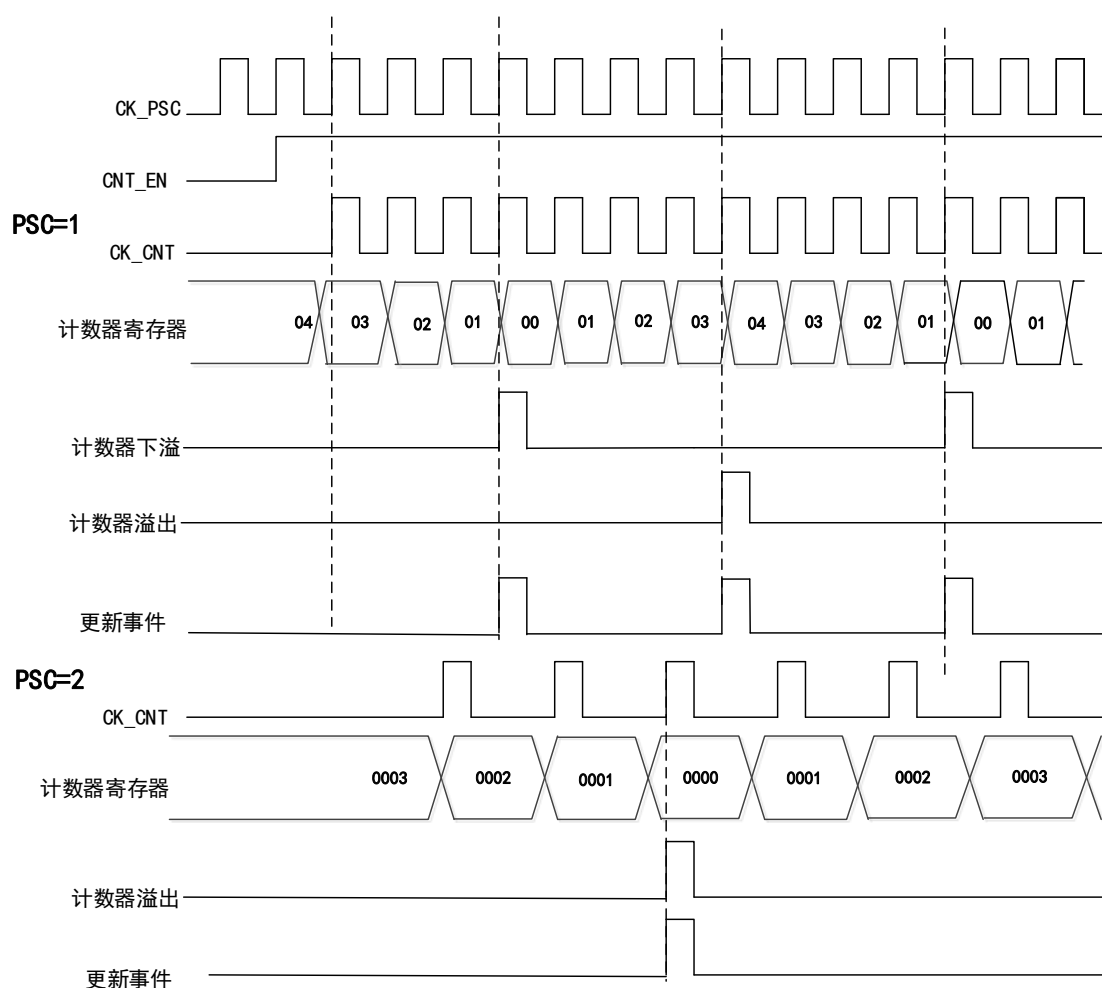


中央对齐模式

通过配置控制寄存器（TMRx_CTRL1）中的 CNTDIR 位，设置为中央对齐模式。

当计数器处于中央对齐模式时，计数器从 0 开始向上计数到自动重载的值（TMRx_AUTORLD），然后从自动重载的值（TMRx_AUTORLD）再向下计数到 0，以此往复，在向上计数时当计数器的值为（AUTORLD-1）时会产生一个计数器上溢事件，在向下计数时计数器的值为 1 时会产生一个计数器下溢事件。

图 26 中央对齐模式下，分频因子为 1 或 2 的时序图



重复计数器 REPCNT

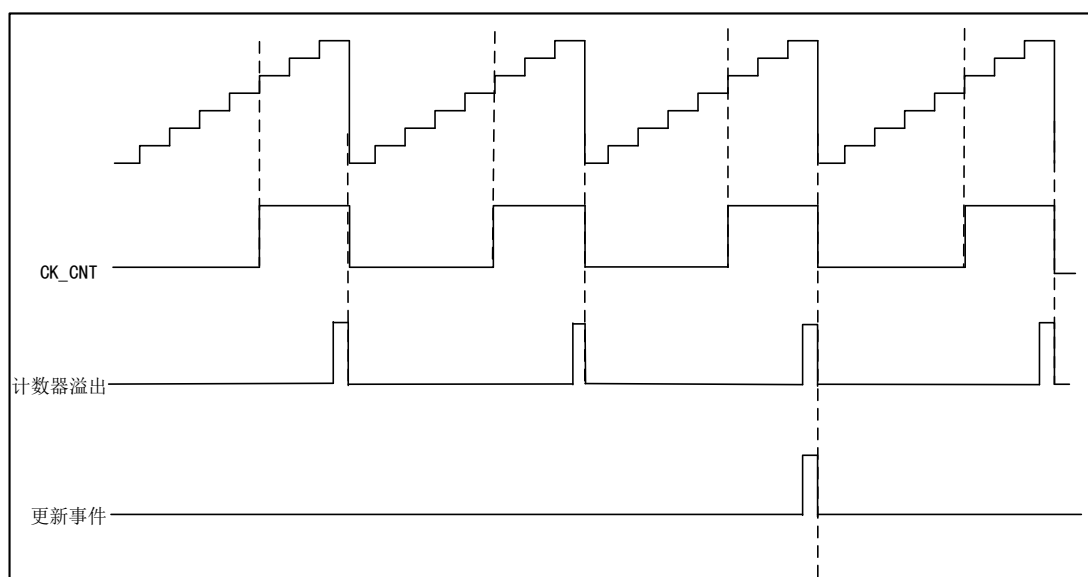
在基本/通用定时器中没有重复计数器 REPCNT，也就表明了基本/通用定时器中发生上溢事件或者下溢事件时，直接就会产生更新事件，而在高级定时器里，因为重复计数器的存在，高级定时器发生上/下溢事件时，只有当重复计数器的值为 0 时才会产生更新事件。

例如，如果高级定时器需要在发生上/下溢事件时就产生更新事件，应将重复计数器的值置 0。

如果在向上计数模式时，使用了重复计数器功能，每当计数器向上计数到 **AUTORLD** 时，发生上溢事件，此时重复计数器的值会减 1，直到重复计数器的值为 0 时会产生更新事件。

即在发生 **N+1** 个（**N** 为重复计数器的值）上/下溢事件时产生更新事件。

图 27 向上计数模式下，设置 REPCNT=2 的时序图



预分频器 PSC

16 位可编程预分频器可以将计数器的时钟频率进行 1~65536 之间任意值的分频（由 TMRx_PSC 寄存器控制），经过分频后的时钟将会驱动计数器 CNT 计数。

预分频器带有缓冲器，它能够在运行中被改变。

14.4.3 输入捕获

输入捕获通道

高级定时器有四个独立的捕获/比较独立通道，每一个捕获/比较的通道都围绕着一个捕获/比较寄存器。

在输入捕获中，被测量的信号会从定时器的外部引脚 T1/2/3/4 进入首先经过边沿检测器和输入滤波器，然后进入捕获通道，每个捕获通道都有相对应的捕获寄存器，当发生捕获时，计数器 CNT 的值将会被锁存在捕获寄存器 CCx 中。在进入捕获寄存器之前，信号还会经过预分频器，用于设定经过多少事件进行一次捕获。

输入捕获应用

输入捕获用来捕获外部事件，并且可以赋予时间标记表明事件的发生时刻，可以测量脉冲跳变沿事件（测量频率或者脉宽），如：在输入引脚上如果出现了被选择的边沿，TMRx_CCx 寄存器会捕获计数器当前的值，同时状态寄存器 TMRx_STS 的 CCxIFLG 位被置 1，如果 CCxIEN=1，便会产生中断。

捕获模式下可以测量一个波形的时序、频率、周期和占空比。在输入捕获模式中将边沿选择设定为上升沿检测，当捕获通道出现上升沿时，发生第一次捕获，此时计数器 CNT 的值会被锁存在捕获寄存器 CCx 中，同时会进入捕获中断，在中断服务程序中记录一次捕获，记下此时的值，当检测到下一个上升沿时，发生第

二次捕获，计数器 CNT 的值会再次锁存在捕获寄存器 CCx 中，此时再次进入捕获中断，读取捕获寄存器的值，通过捕获就会得出此脉冲信号的周期。

14.4.4 输出比较

输出比较一共有八种模式：冻结，匹配时通道 x 为有效电平、匹配时通道 x 为无效电平、翻转、强制为无效、强制为有效、PWM1 和 PWM2 模式，由 TMRx_CCMx 寄存器中的 OCxMOD 位配置，在输出比较模式中控制输出信号的波形。

输出比较应用

输出比较模式中，定时器产生脉冲的位置、极性、频率和时间都是可以控制的。

当计数器的值和捕获/比较寄存器的值相等时，通过配置 TMRx_CCMx 寄存器中的 OCxMOD 位和输出极性 TMRx_CCEN 寄存器中的 CCxPOL 位，通道的输出可以被置高电平、低电平或者翻转。

在 TMRx_STS 寄存器中的 CCxIFLG=1 时，如果 TMRx_DIEN 寄存器中的 CCxIEN=1 产生中断；TMRx_CTRL2 寄存器中的 CCDSEL=1 产生 DMA 请求。

14.4.5 PWM 输出模式

PWM 模式是定时器对外输出可以调节的脉冲信号，其中信号的脉宽是由比较寄存器 CCx 的值决定，周期是由自动重载 AUTORLD 的值决定。

PWM 输出模式分为 PWM 模式 1 和 PWM 模式 2；PWM 模式 1 和 PWM 模式 2 分为向上计数和向下计数和边沿对齐计数；PWM 模式 1 中如果计数器 CNT 的值小于比较寄存器 CCx 的值，输出有效电平，否则反之。

设置 $CCx=5, AUTORLD=7$, 在 PWM 模式下的时序图

图 28 PWM1 向上计数模式的时序图

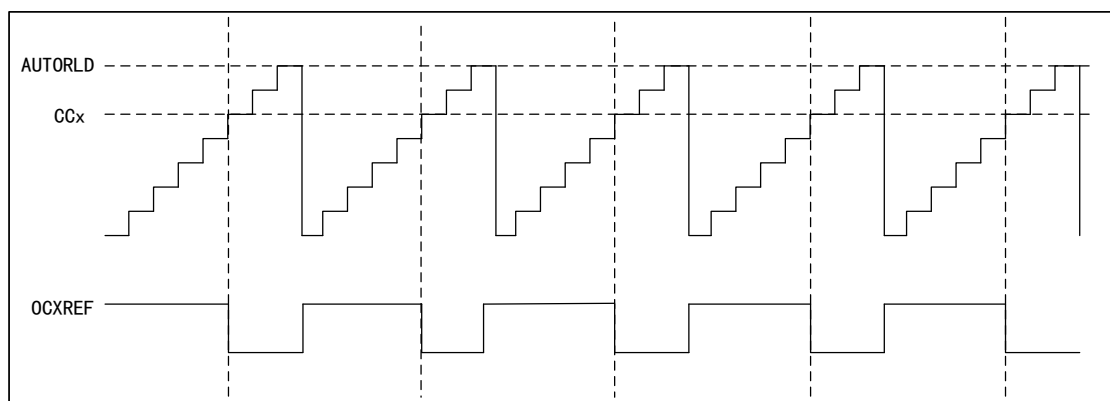


图 29 PWM1 向下计数模式的时序图

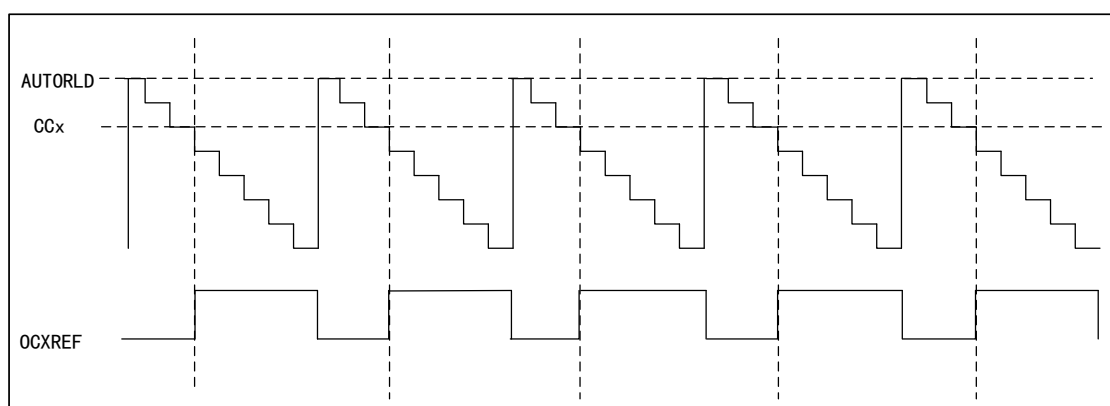
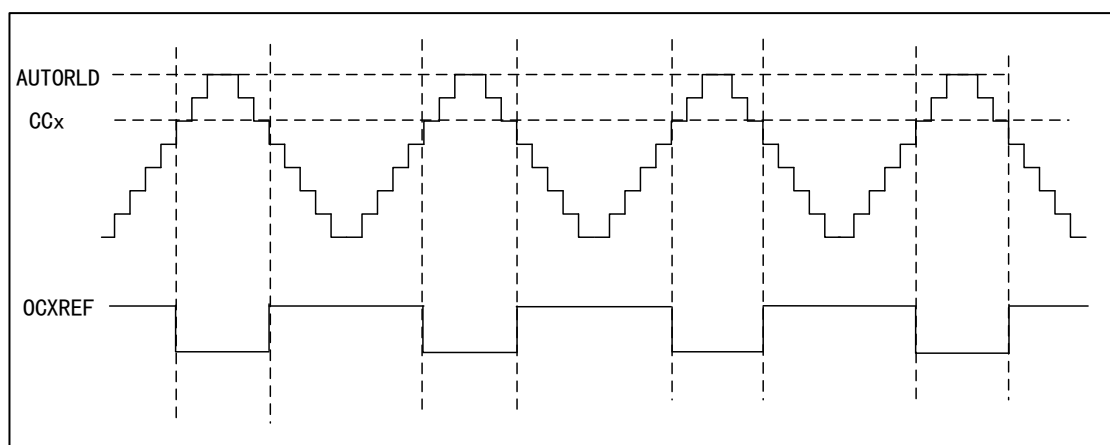


图 30 PWM1 中央对齐模式的时序图



PWM 模式 2 中如果计数器 CNT 的值小于比较寄存器 CCx 的值，输出无效电平，否则反之。

设置 $CCx=5, \text{AUTORLD}=7$, 在 PWM 模式 2 下的时序图

图 31 PWM2 向上计数模式的时序图

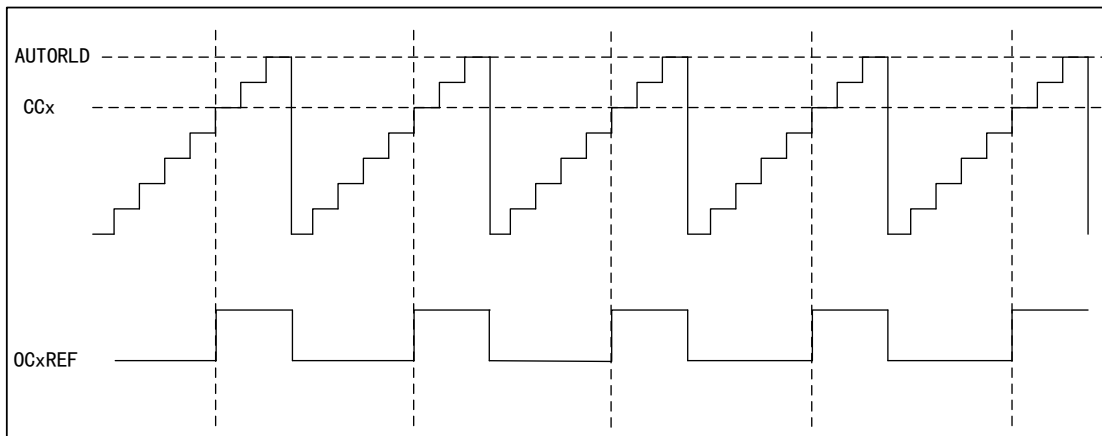


图 32 PWM2 向下计数模式的时序图

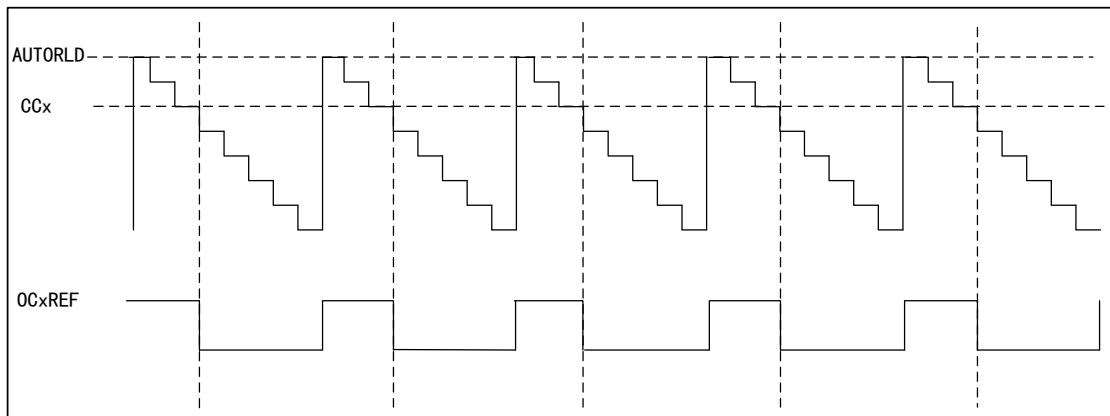
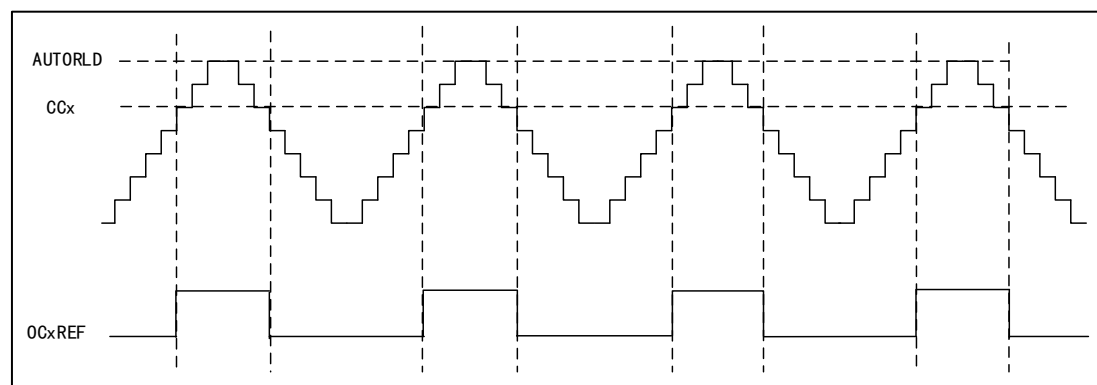


图 33 PWM2 中央对齐模式的时序图



14.4.6 PWM 输入模式

PWM 输入模式是输入捕获的一个特例。

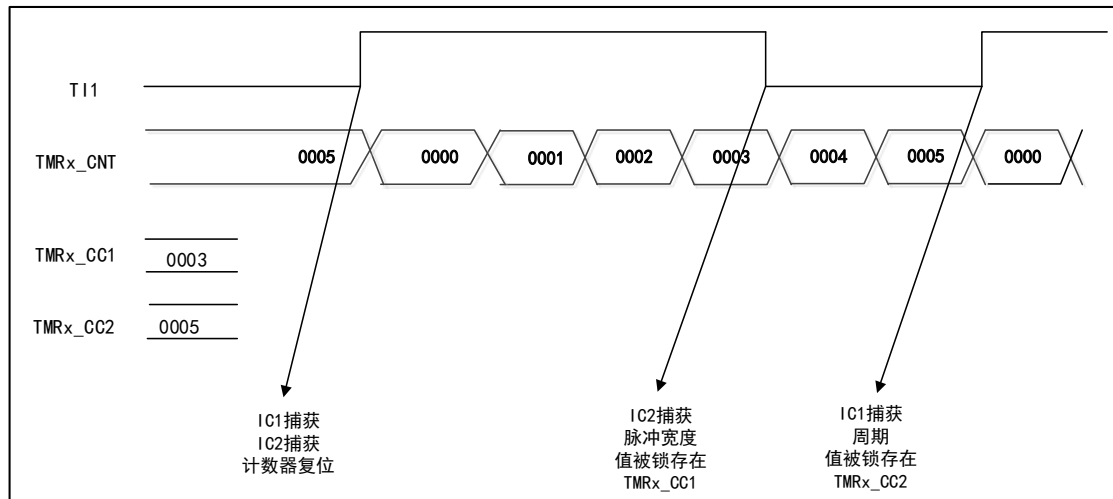
PWM 输入模式，只有 $T11FP1$ 、 $T11FP2$ 连接到了从模式控制器，所以只能从通

道 TMRx_CH1 和 TMRx_CH2 输入，且需要占用 CH1、CH2 的捕获寄存器。

在 PWM 输入模式中，PWM 信号从 TMRx_CH1 进入，信号会被分成两路，一路可以测量周期，一路可以测量占空比。在配置中只需设置其中一路的极性，另一路会自动配置相反的极性。

在此模式中，从模式控制器要配置成复位模式（TMRx_SMCTRL 寄存器的 SMFSEL 位）。

图 34 PWM 输入模式时序图



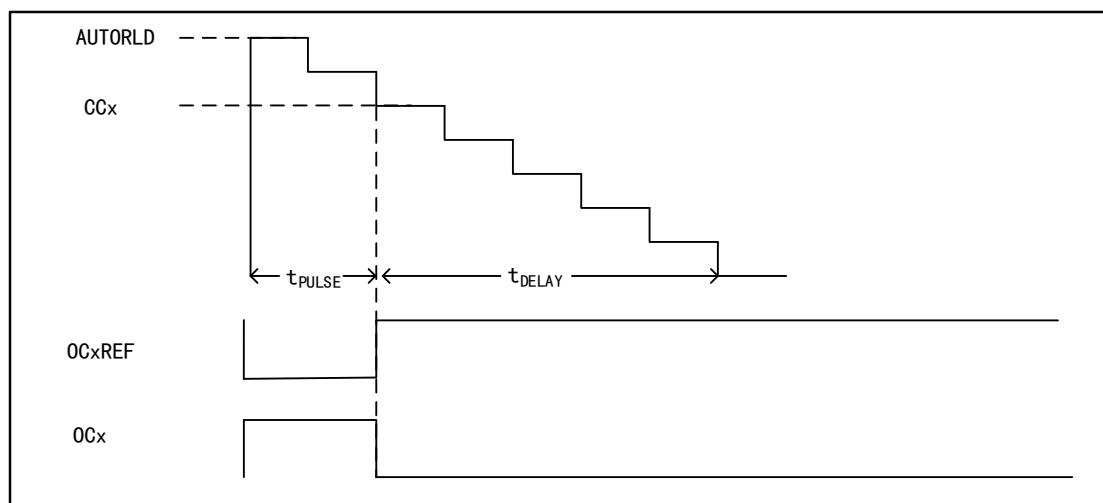
14.4.7 单脉冲模式

单脉冲模式是定时器比较输出中一种特殊情况，也是 PWM 输出模式的特例。

设置 TMRx_CTRL1 寄存器的 SPMEN 位选择单脉冲模式，计数器启动后，在未发生更新事件之前有一定个数的脉冲输出，当发生更新事件后计数器停止计数，后续不再有变化的 PWM 波形输出。

单脉冲模式通过程序在一定可控延迟后，产生一个脉宽可控的脉冲，延时时间由 TMRx_CCx 寄存器的值定义；在增计数模式下延时时间为 CCx，脉冲宽度为 AUTORLD-CCx；在减计数模式下延时时间为 AUTORLD-CCx，脉冲宽度为 CCx。

图 35 单脉冲模式下的时序图



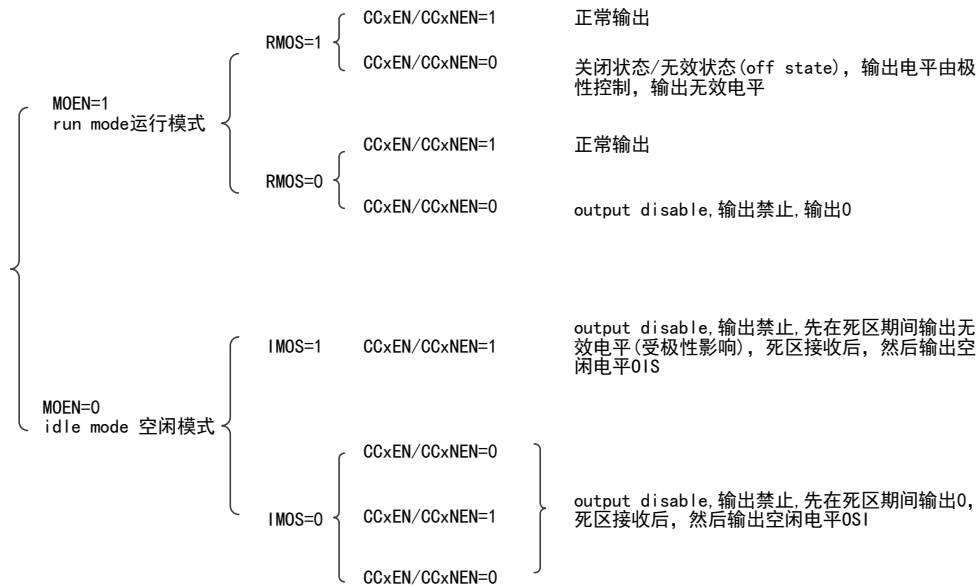
14.4.8 寄存器对输出波形的影响

以下寄存器会影响定时器输出波形的电平，详细信息可参照“寄存器功能描述”。

- (1) TMRx_CCEN 寄存器中的 CCxEN、CCxNEN 位
 - CCxNEN=0、CCxEN=0: 关闭输出（输出禁止，无效状态）
 - CCxNEN=1、CCxEN=1: 开启输出（输出使能，正常输出）
- (2) TMRx_BDT 寄存器中的 MOEN 位
 - MOEN=0: 空闲模式
 - MOEN=1: 运行模式
- (3) TMRx_CTRL2 寄存器中的 OCxOIS、OCxNOIS 位
 - OCxOIS=0、OCxNOIS=0: 空闲时（MOEN=0）死区后的输出电平为 0
 - OCxOIS=1、OCxNOIS=1: 空闲时（MOEN=0）死区后的输出电平为 1
- (4) TMRx_BDT 寄存器中的 RMOS 位
 - RMOS 的应用环境：对应互补通道、定时器运行模式下（MOEN=1）、定时器不工作（CCxEN=0、CCxNEN=0）或工作（CCxEN=1、CCxNEN=1）情况下
- (5) TMRx_BDT 寄存器中的 IMOS 位
 - IMOS 的应用环境：对应互补通道、定时器空闲模式下（MOEN=0）、定时器不工作（CCxEN=0、CCxNEN=0）或工作（CCxEN=1、CCxNEN=1）情况下
- (6) TMRx_CCEN 寄存器的 CCxPOL、CCxNPOL 位
 - CCxPOL=0、CCxNPOL=0: 输出极性，高电平有效
 - CCxPOL=1、CCxNPOL=1: 输出极性，低电平有效

下图罗列了影响输出波形的寄存器结构关系

图 36 影响输出波形的寄存器结构关系



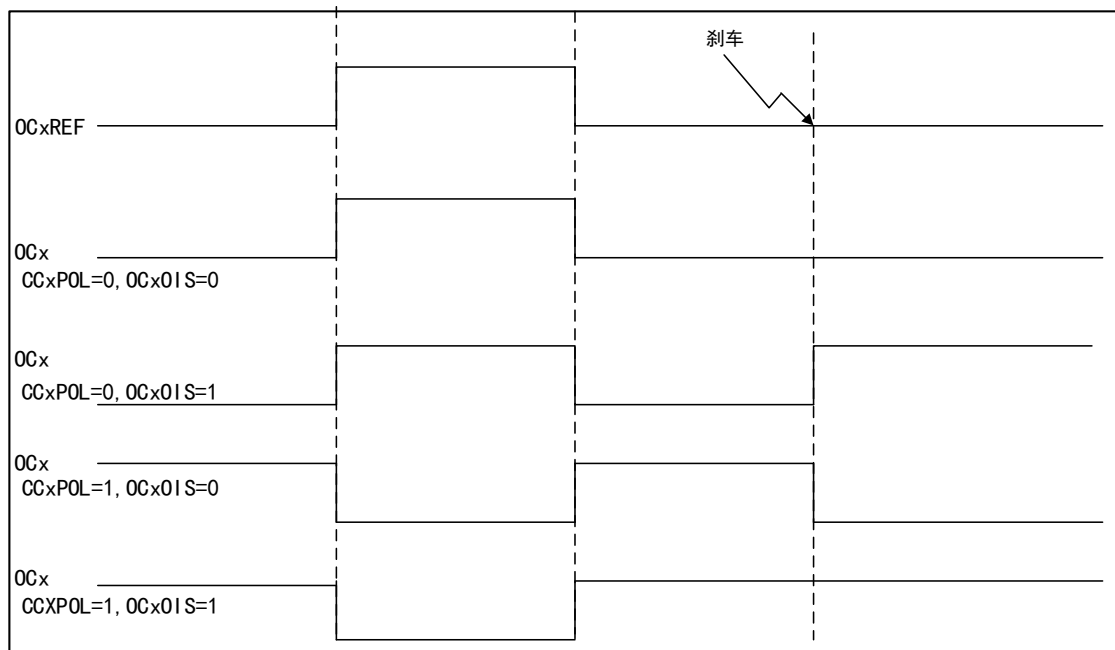
14.4.9 刹车功能

刹车的信号源为时钟故障事件和外部输入接口。

其中 TMRx_BDT 寄存器中的 BRKEN 位可以使能刹车功能, BRKPOL 位配置刹车输入信号的极性。TMRx_BKFT 寄存器可以配置刹车输入信号滤波器。

发生刹车事件时, 可以根据相关控制位的状态修改输出脉冲信号电平。

图 37 发生刹车事件的时序图

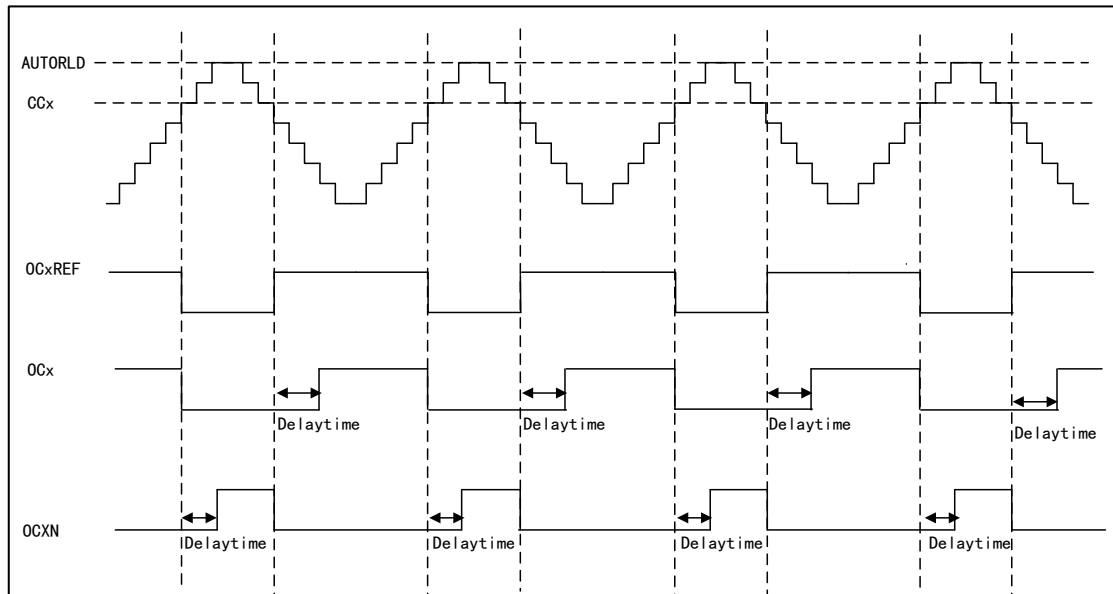


14.4.10 互补输出和死区插入

TMR1 定时器有三组互补输出通道。插入死区时间用于生成互补的输出信号，确保通道互补的两路信号不会同时有效。根据定时器连接的输出器件以及特性来设定死区时间

配置 TMRx_BDT 寄存器的 DTS 位可以控制死区的持续时间

图 38 带死区插入的互补输出



14.4.11 强制输出模式

强制输出模式下无视比较结果，直接根据配置指令输出相应电平。

- TMRx_CCMx 寄存器的 CCxSEL=00, 设定 CCx 通道为输出
- TMRx_CCMx 寄存器的 OCxMOD=100/101, 设定强制 OCxREF 信号为无效/有效状态

在此模式中仍旧会产生相应的中断和 DMA 请求。

14.4.12 编码器接口模式

编码器接口模式相当于是一个带有方向选择的外部时钟，在编码器接口模式中，计数器的内容可一直指示编码器的位置。

选择编码器接口的方法如下：

- 通过设置 TMRx_SMCTRL 寄存器的 SMFSEL 位，可以设定计数器是在 TI1 通道/TI2 通道边沿计数，或者同时在 TI1 和 TI2 的边沿计数。
- 通过设置 TMRx_CCEN 寄存器中的 CC1POL 和 CC2POL 位，可以选择 TI1 和 TI2 的极性。
- 通过设置 TMRx_CCM1 寄存器中的 IC1F 和 IC2F 位，可以选择是否进行滤波。

两个输入 TI1 和 TI2 可以作为增量编码器的接口，计数器由在 TI1 和 TI2 经过滤波和边沿选择后的信号 TI1FP1 和 TI2FP2 的有效跳变驱动。

根据 T11 和 T12 的输入信号，产生计数脉冲和方向信号

- 根据输入信号的跳变顺序,计数器会向上/向下计数
- 将控制寄存器 TMRx_CTRL1 的 CNTDIR 设置为只读（任一输入端的跳变都会重新计算 CNTDIR）

计数器计数方向改变机制如下图

表格 46 计数方向与编码器的关系

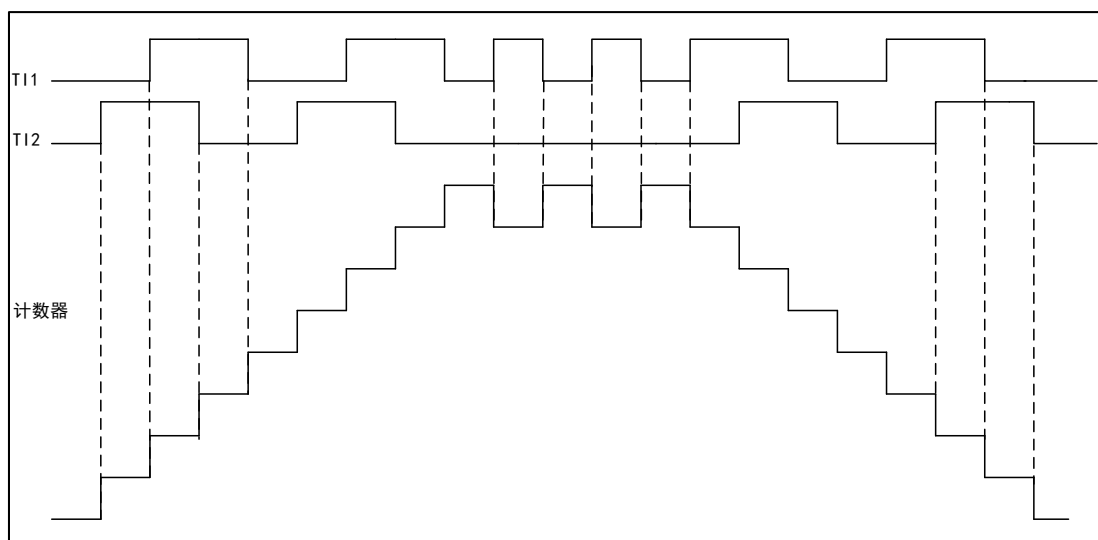
有效边沿		仅在 T11 计数		仅在 T12 计数		在 T11 和 T12 计数	
相对信号的电平		高	低	高	低	高	低
T11FP1	上升沿	—		向下计数	向上计数	向下计数	向上计数
	下降沿			向上计数	向下计数	向上计数	向下计数
T12FP2	上升沿	向上计数	向下计数	—		向上计数	向下计数
	下降沿	向下计数	向上计数			向下计数	向上计数

外部的增量编码器可以不用外部接口逻辑直接与 MCU 连接，所以使用比较器将编码器的差动输出转换到数字信号来增加抗噪声干扰。

在下图的实例中

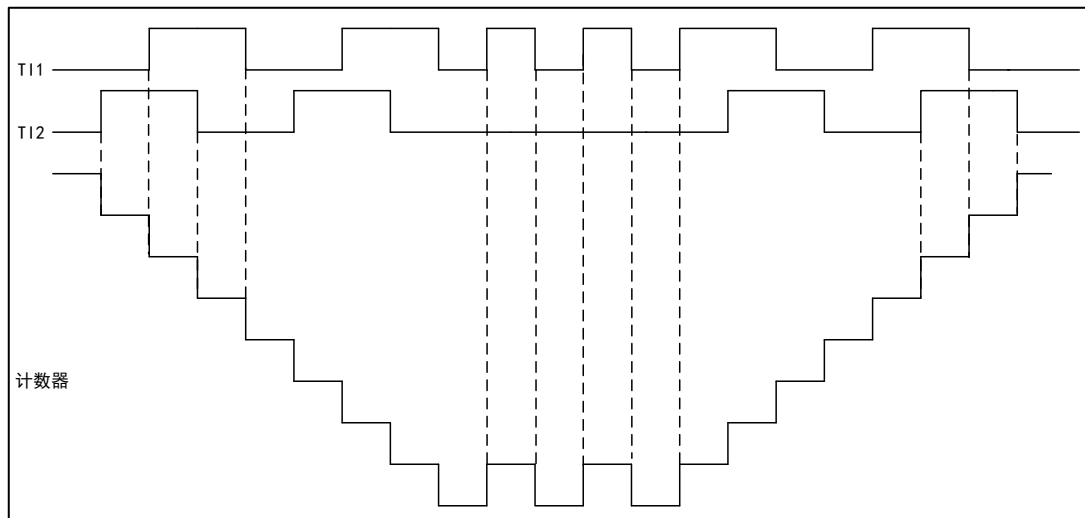
- 将 T11FP1 映射到 T11
- 将 T12FP2 映射到 T12
- T11FP1 和 T12FP2 都不反相
- 输入信号在上升沿和下降沿有效
- 使能计数器

图 39 编码器模式下的计数器操作实例



例如：当 T11 处在低电平时，如果 T12 出现上升沿状态，计数器向上计数。

图 40 TI1FP1 反相的编码器接口模式实例



例如：当 TI1 处于低电平时，TI2 发生上升沿跳变，计数器向下计数。

14.4.13 从模式

TMR1 定时器可以进行外部的触发同步

- 复位模式
- 门控模式
- 触发模式

可设置 TMRx_SMCTRL 寄存器中的 SMFSEL 位来选择是哪种模式

SMFSEL=100 设定复位模式，SMFSEL=101 设定门控模式，SMFSEL=110 设定触发模式。

复位模式下，在发生一个触发输入事件时，计数器和预分频器会被初始化，选中的触发输入（TRGI）的上升沿重新初始化计数器，并且产生一个更新寄存器的信号。

门控模式下，计数器的使能依赖于选中的输入端的高电平，当触发输入为高时，计数器的时钟开启，一旦触发输入变为低，则计数器停止（但不复位），计数器的启动和停止都是受控制的。

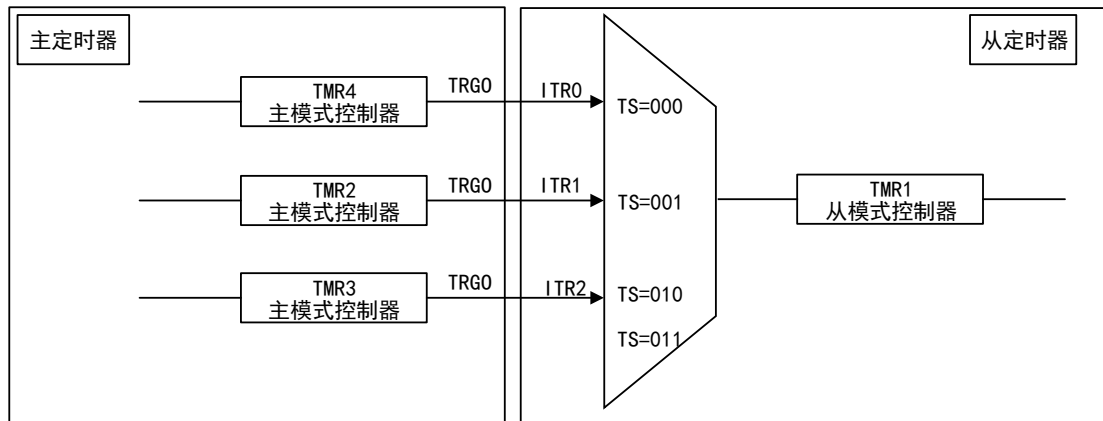
触发模式下，计数器的使能依赖于选中的输入端上的事件，计数器在触发输入的上升沿启动（但不复位），只有计数器的启动是受控制的。

14.4.14 定时器互连

TMR1 各个定时器可以互相连接实现定时器之间的同步或级联。需要配置一个定时器处于主模式，另一个定时器处于从模式。

定时器处于主模式时可以对从模式定时器的计数器进行复位、启动、停止和提供时钟源等。

图 41 TMR1 与其他定时器互连



在定时器互连中可以：

- 将一个定时器作为另一个寄存器的预分频器
- 用一个定时器的使能信号启动另一个寄存器
- 用一个定时器的更新事件启动另一个寄存器
- 用一个定时器的使能选通另一个定时器
- 用一个外部触发同步两个定时器

14.4.15 中断和 DMA 请求

定时器在工作时产生事件时会发生中断

- 更新事件（计数器上/下溢出，计数器初始化）
- 触发事件（计数器启动、停止、内/外部触发）
- 捕获/比较事件
- 刹车信号输入事件。

其中一些内部中断事件可以产生 DMA 请求，还有专门的接口允许或者禁止触发 DMA 请求。

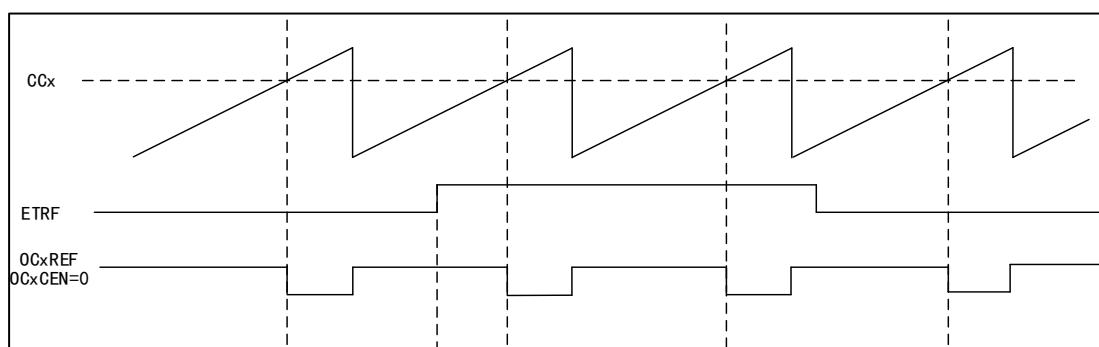
14.4.16 在外部事件时清除 OCxREF 信号

此功能是用用于输出比较和 PWM 模式。

在一个通道中，用 ETRF 输入端口的高电平将 OCxREF 的信号降为低电平，捕获/比较寄存器 TMRx_CCMx 中的 OCxCEN 的位置 1，OCxREF 信号会保持为低电平直到下一次发生更新事件。

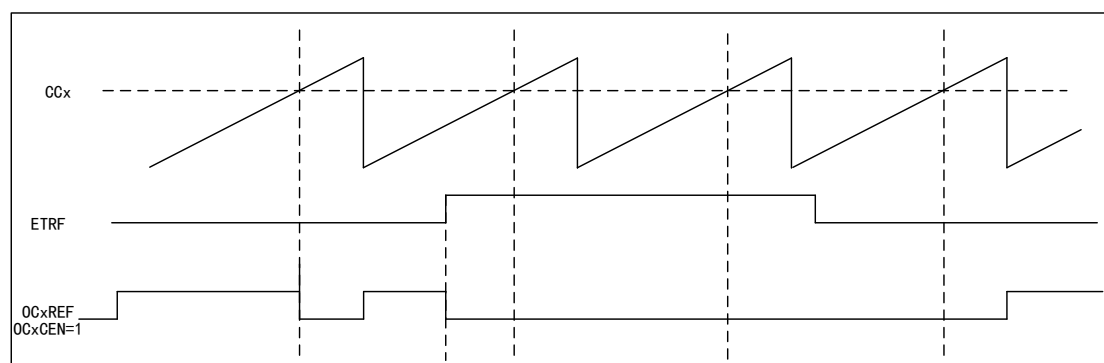
将 TMR1 置于 PWM 模式，关闭外部触发预分频器，禁止外部触发模式 2，当 ETRF 输入为高时，通过设置 OCxCEN=0，输出的 OCxREF 信号如下图。

图 42 OCxREF 时序图



将 TMR1 置于 PWM 模式，关闭外部触发预分频器，禁止外部触发模式 2，当 ETRF 输入为高时，通过设置 OCxREN=1，输出的 OCxREF 信号如下图。

图 43 OCxREF 时序图

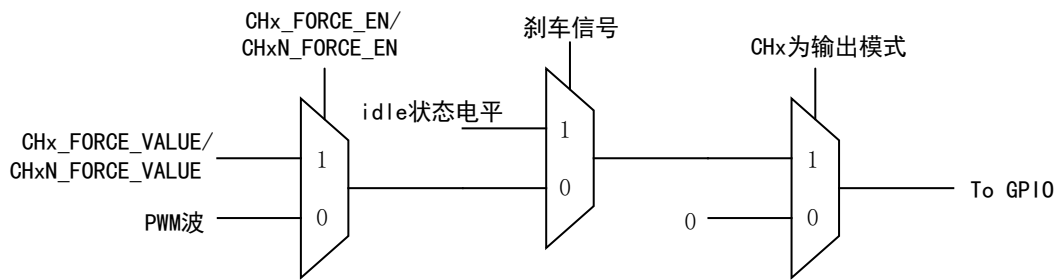


14.4.17 手动控制 PWM 输出

当 CCxEN/CCxNEN 位为 1，且 MOEN 位为 1 时，若 TMRx_OUTPUTCTRL1 寄存器中的 CHx_FORCE_EN/CHxN_FORCE_EN 位为 1，则相应通道将输出 TMRx_OUTPUTCTRL2 寄存器 CHx_FORCE_VALUE/CHxN_FORCE_VALUE 位的值；若 TMRx_OUTPUTCTRL1 寄存器中的 CHx_FORCE_EN/CHxN_FORCE_EN 位为 0，则相应通道将输出 PWM 波形。注意，在手动控制 PWM 输出模式下，其输出会受到 PWM 刹车信号控制，而在 I/O 输出模式下，不受该信号控制。具体如下图所示。

更改 TMRx_OUTPUTCTRL1 和 TMRx_OUTPUTCTRL2 寄存器时，若 TMRx_OUTPUTCTRL1 的 OUTPUTCTRL_BUF 位为 0 时，手动控制引脚输出的改变与系统时钟同步，输出状态改变立即生效；若 TMRx_OUTPUTCTRL1 的 OUTPUTCTRL_BUF 位为 1 时，手动控制引脚输出的改变与更新事件同步，输出状态在下次更新事件时生效。

图 44 输出控制模块逻辑图



注意：OUTPUTCTRL_BUF 位不可与 TMRx_OUTPUTCTRL1 寄存器其他位同时写入，如果需要修改 OUTPUTCTRL_BUF 位和 TMRx_OUTPUTCTRL1 寄存器中其他控制位，需要进行两次写操作。任意情况下，对 OUTPUTCTRL_BUF 位的修改与时钟同步生效。

14.4.18 TMR1 与 M0CP 交互

在多数电机控制运用中，SVPWM 输出的 Ta, Tb, Tc 实际上是 TMR1 的三相占空比，为一个小于 1 的正数。软件要将 3 个值乘以 PWM 的周期后放入 PWM 占空比寄存器中。为提高效率，APM32F035x8T7 提供了可选的硬件方案，将 M0CP 配置为 SVPWM 计算模式（详见协处理器(M0CP)），并置位 RUN 启动运算，硬件会自动将 Ta、Tb、Tc 按 Qn 格式乘法乘以 PWM 的周期，随后将乘积自动放入 CCx 和 CCxC（x=0-2）。

注意：M0CP 将计算结果放入 CCx/CCxC 后的计算结果生效时间可由 TMRx_M0CP 寄存器的 CCx_NO_BUFFER 位控制：此位置 0 时，CCx/CCxC 所写入的值在下次更新事件时生效；此位置 1 时，CCx/CCxC 所写入的值立即生效。

需要注意的是当 TMR1 模块中 PDCON 位为 0 时，Ta、Tb、Tc 与 PWM 的周期 PWM_PERIOD 相乘，而当 PDCON 为 1 时，Ta、Tb、Tc 先被 1 减之后再与 PWM 的周期 PWM_PERIOD 相乘，即如下表所示：

表格 47 PWMx 周期与 PDCON、Ta、Tb、Tc 的关系

CCx	周期 (PDCON = 0)	周期 (PDCON = 1)
CC1/CC1C	Ta * PWM_PERIOD	(1 - Ta) * PWM_PERIOD
CC2/CC2C	Tb * PWM_PERIOD	(1 - Tb) * PWM_PERIOD
CC3/CC3C	Tc * PWM_PERIOD	(1 - Tc) * PWM_PERIOD

注意：PWM_PERIOD 可由 TMRx_M0CP 的 PWM_PERIOD_SEL 位选择来自 TMRx_AUTORLDR 寄存器或 TMRx_AUTORLDR 影子寄存器。PWM_PDCON 的值由 TMRx_M0CP 的 PWM_PDCON_SEL 位控制。

实际运用中，必须先配置好 TMRx_M0CP 寄存器。接着将 Ta, Tb, Tc 放入 M0CP 的操作数寄存器 X, Y, Z 内（同 SVPWM 运算完成后结果的放置顺序），然后配置 M0CP 的 CTRL_REG 开始运算。

运算过程中 M0CP 的 STAT_REG 的 BUSY 位始终为 1，运算完成后硬件清零。

14.5 寄存器地址映射

下表中将高级定时器的所有寄存器映射到一个 16 位可寻址（编址）空间。

表格 48 TMR1 寄存器地址映射

寄存器名	描述	偏移地址
TMRx_CTRL1	控制寄存器 1	0x00
TMRx_CTRL2	控制寄存器 2	0x04
TMRx_SMCTRL	从模式控制寄存器	0x08
TMRx_DIEN	DMA/中断使能寄存器	0x0C
TMRx_STS	状态寄存器	0x10
TMRx_CEG	控制事件产生寄存器	0x14
TMRx_CCM1	捕获/比较模式寄存器 1	0x18
TMRx_CCM2	捕获/比较模式寄存器 2	0x1C
TMRx_CCEN	捕获/比较使能寄存器	0x20
TMRx_CNT	计数器寄存器	0x24
TMRx_PSC	预分频寄存器	0x28
TMRx_AUTORLD	自动重装载寄存器	0x2C
TMRx_REPCNT	重复计数寄存器	0x30
TMRx_CC1	通道 1 捕获/比较寄存器	0x34
TMRx_CC2	通道 2 捕获/比较寄存器	0x38
TMRx_CC3	通道 3 捕获/比较寄存器	0x3C
TMRx_CC4	通道 4 捕获/比较寄存器	0x40
TMRx_BDT	刹车和死区寄存器	0x44
TMRx_DCTRL	DMA 控制寄存器	0x48
TMRx_DMADDR	连续模式的 DMA 地址寄存器	0x4C
TMRx_CC1C	通道 1 捕获/比较寄存器互补寄存器	0x50
TMRx_CC2C	通道 2 捕获/比较寄存器互补寄存器	0x54
TMRx_CC3C	通道 3 捕获/比较寄存器互补寄存器	0x58
TMRx_CC5	通道 5 捕获/比较寄存器	0x5C
TMRx_CCM3	捕获/比较模式寄存器 3	0x60
TMRx_CTRL3	控制寄存器 3	0x64
TMRx_BKFT	刹车滤波寄存器	0x68
TMRx_M0CP	协处理器寄存器	0x6C
TMRx_OUTPUTCTRL1	输出控制寄存器 1	0x70

寄存器名	描述	偏移地址
TMRx_OUTPUTCTRL2	输出控制寄存器 2	0x74
TMRx_CTRL4	控制寄存器 4	0x78

14.6 寄存器功能描述

14.6.1 控制寄存器 1 (TMRx_CTRL1)

偏移地址: 0x00

复位值: 0x0000

位/域	名称	R/W	描述
0	CNTEN	R/W	<p>使能计数器 (Counter Enable)</p> <p>0: 禁止</p> <p>1: 使能</p> <p>定时器配置为外部时钟、门控模式和编码器模式时, 需要通过软件对该位写 1 启动定期工作; 配置为触发模式时, 可硬件写 1。</p>
1	UD	R/W	<p>禁止更新 (Update Disable)</p> <p>更新事件可引起 AUTORLD、PSC、CCx 产生更新设置的数值。</p> <p>0: 允许更新事件 (UEV)</p> <p>更新事件可以由以下任一情况产生:</p> <ul style="list-style-type: none"> 计数器溢出/下溢; 设置 UEG 位; 从模式控制器产生的更新。 <p>1: 禁止更新事件</p>
2	URSSEL	R/W	<p>更新请求源 (Update Request Source Select)</p> <p>如果使能了中断或 DMA, 更新事件可产生更新中断或 DMA 请求, 通过该位可选择不同的更新请求源。</p> <p>0: 计数器上溢或下溢</p> <ul style="list-style-type: none"> 设置 UEG 位 通过从模式控制器产生的更新 <p>1: 计数器上溢或下溢</p>
3	SPMEN	R/W	<p>使能单脉冲模式 (Single Pulse Mode Enable)</p> <p>产生更新事件时, 可改变通道的输出电平; 在该模式下, 会清除 CNTEN 位, 停止计数器, 后续不再改变通道的输出电平。</p> <p>0: 禁用</p> <p>1: 使能</p>
4	CNTDIR	R/W	<p>配置计数器计数方向 (Counter Direction)</p> <p>当计数器配置为中央对齐模式或编码器模式时, 该位为只读。</p> <p>0: 向上计数</p> <p>1: 向下计数</p>
6:5	CAMSEL	R/W	<p>选择中央对齐模式 (Center Aligned Mode Select),</p> <p>中央对齐模式下, 计数器交替的向上向下计数; 否则只向上或向下计数。不同的中央对齐模式, 影响输出输出通道的输出比较中断标志位置 1 的时机; 在计数器禁止时 (CNTEN=0) 时, 选择中央对齐模式。</p> <p>00: 边沿对齐模式</p> <p>01: 中心对齐模式 1 (在向下计数时, 输出通道的输出比较中断标志位置 1)</p>

位/域	名称	R/W	描述
			10: 中心对齐模式 2 (在向上计数时, 输出通道的输出比较中断标志位置 1) 11: 中心对齐模式 3 (在向上/下计数时, 输出通道的输出比较中断标志位置 1)
7	ARPEN	R/W	TMRx_AUTORLD 寄存器自动重装缓冲使能 (Auto-reload Preload Enable) 禁止缓存区时, 程序修改 TMRx_AUTORLD 会立刻修改装入计数器的数值; 使能缓存区时, 程序修改 TMRx_AUTORLD 会在下一个更新事件修改装入计数器的数值。 0: 禁止 1: 使能
9:8	CLKDIV	R/W	时钟分频系数 (Clock Division) 死区、数字滤波器的配置由 CK_INT 提供时钟, 通过设置改位可调整死区时间、数字滤波器的采用时钟。 00: $t_{DTS}=t_{CK_INT}$ 01: $t_{DTS}=2 \times t_{CK_INT}$ 10: $t_{DTS}=4 \times t_{CK_INT}$ 11: 保留
15:10			保留

14.6.2 控制寄存器 2 (TMRx_CTRL2)

偏移地址: 0x04

复位值: 0x0000

位/域	名称	R/W	描述
0	CCPEN	R/W	使能捕获/比较预装载 (Capture/Compare Preloaded Enable) 该位影响 CCxEN、CCxNEN、OCxMOD 数值的改变, 禁止预装载时, 程序修改会立刻影响定时器的设置; 使能预装载时, 只在设置了 COMG 会后更新, 从而影响定时器的设置; 该位只在具有互补输出的通道起作用。 0: 禁止 1: 使能
1			保留
2	CCUSEL	R/W	选择捕获/比较控制更新 (Capture/Compare Control Update Select) 仅在捕获比较预装载使能 (CCPEN=1) 时, 且只对互补输出通道起作用。 0: 只能通过设置 COMG 位更新 1: 可以通过设置 COMG 位或者 TRGI 上的上升沿更新
3	CCDSEL	R/W	选择发出捕获/比较的 DMA 请求 (Capture/Compare DMA Select) 0: 当发生 CCx 事件时, 送出 CCx 的 DMA 请求 1: 当发生更新事件时, 送出 CCx 的 DMA 请求
6:4	MMSEL	R/W	选择定时器主模式下用于 TRGO 的信号 (Master Mode Signal Select) 工作在主模式的定时器的信号可用于 TRGO, 从而影响处在从模式且与主定时器级联的的定时器工作, 具体影响可从模式的定时器配置有关。 000: 复位, 主模式定时器的复位信号用于 TRGO 001: 使能, 主模式定时器的计数器使能信号用于 TRGO 010: 更新, 主模式定时器的更新事件用于 TRGO 011: 比较脉冲, 主模式定时器捕获/比较成功 (CCxIFLG=1) 时输出一个脉冲信号用于 TRGO 100: 比较模式 1, OC1REF 用于触发 TRGO

位/域	名称	R/W	描述
			101: 比较模式 2, OC2REF 用于触发 TRGO 110: 比较模式 3, OC3REF 用于触发 TRGO 111: 比较模式 4, OC4REF 用于触发 TRGO
7	TI1SEL	R/W	TI1 选择 (Timer Input 1 Select) 0: TMRx_CH1 引脚连到 TI1 输入 1: TMRx_CH1、TMRx_CH2 和 TMRx_CH3 引脚经异或后连到 TI1 输入
8	OC1OIS	R/W	配置 OC1 输出空闲状态 (OC1 Output Idel State Configure) 仅在当 MOEN=0 时、实现了 OC1N, 只影响 OC1 死区时间后的电平状态。 0: OC1=0 1: OC1=1 注: 当 TMRx_BDT 寄存器中 LOCKCFG 位级别为 1、2 或 3 时, 该位不能修改。
9	OC1NOIS	R/W	配置 OC1N 输出空闲状态 (OC1N Output Idel State Configure) 仅在当 MOEN=0、实现了 OC1N, 只影响 OC1N 死区时间后的电平状态。 0: OC1N=0 1: OC1N=1 注: 当 TMRx_BDT 寄存器中 LOCKCFG 位级别为 1、2 或 3 时, 该位不能修改。
10	OC2OIS	R/W	配置 OC2 输出空闲状态。参考 OC1OIS 位
11	OC2NOIS	R/W	配置 OC2N 输出空闲状态。参考 OC1NOIS 位
12	OC3OIS	R/W	配置 OC3 输出空闲状态。参考 OC1OIS 位
13	OC3NOIS	R/W	配置 OC3N 输出空闲状态。参考 OC1NOIS 位
14	OC4OIS	R/W	配置 OC4 输出空闲状态。参考 OC1OIS 位
15	保留		

14.6.3 从模式控制寄存器 (TMRx_SMCTRL)

偏移地址: 0x08

复位值: 0x0000

位/域	名称	R/W	描述
2:0	SMFSEL	R/W	选择从模式功能 (Slave Mode Function Select) 000: 禁止从模式, 定时器可作为主模式定时器影响从模式定时器的工 作; 如果 CTRL1_CNTEN=1, 则预分频器直接由内部时钟驱 动。 001: 编码器模式 1, 根据 TI1FP1 的电平, 计数器在 TI2FP2 的边沿 计数。 010: 编码器模式 2, 根据 TI2FP2 的电平, 计数器在 TI1FP1 的边沿 计数。 011: 编码器模式 3, 根据另一个信号的输入电平, 计数器在 TI1FP1、TI2FP2 的边沿计数。 100: 复位模式, 从模式定时器在收到 TRGI 的上升沿信号后复位计数 器, 并产生更新寄存器的信号。 101: 门控模式, 从模式定时器在收到 TRGI 高电平信号时, 启动计数 器工作; 收到 TRGI 低电平时停止计数器工作; 再收到 TRGI 高 电平信号时, 继续工作; 整个期间不复位计数器。

位/域	名称	R/W	描述
			<p>110: 触发模式, 从模式定时器在收到 TRGI 的上升沿信号后, 启动计数器工作。</p> <p>111: 外部时钟模式 1, 选择 TRGI 的上升沿信号作为时钟源驱动计数器工作。</p>
3	OCCSEL	R/W	<p>选择 OCREF 信号清除源 (OCREF Clear Source Select)</p> <p>该位用来选择 OCREF 清除源</p> <p>0: OCREF_CLR</p> <p>1: ETRF</p>
6:4	TRGSEL	R/W	<p>选择触发输入信号 (Trigger Input Signal Select)</p> <p>为了避免在改变该位值时产生错误的边沿检测, 须在 SMFSEL=0 时改变。</p> <p>000: 内部触发 ITR0</p> <p>001: 内部触发 ITR1</p> <p>010: 内部触发 ITR2</p> <p>011: 内部触发 ITR3</p> <p>100: 通道 1 输入边沿检测器 TIF_ED</p> <p>101: 通道 1 滤波后定时器输入 TI1FP1</p> <p>110: 通道 2 滤波后的定时器输入 TI2FP2</p> <p>111: 外部触发输入 (ETRF)</p>
7	MSMEN	R/W	<p>使能主/从模式 (Master/slave Mode Enable)</p> <p>0: 无效</p> <p>1: 使能主/从模式</p>
11:8	ETFCFG	R/W	<p>配置外部触发滤波器 (External Trigger Filter Configure)</p> <p>0000: 禁用滤波器, 以 f_{DTS} 采样</p> <p>0001: DIV=1, N=2</p> <p>0010: DIV=1, N=4</p> <p>0011: DIV=1, N=8</p> <p>0100: DIV=2, N=6</p> <p>0101: DIV=2, N=8</p> <p>0110: DIV=4, N=6</p> <p>0111: DIV=4, N=8</p> <p>1000: DIV=8, N=6</p> <p>1001: DIV=8, N=8</p> <p>1010: DIV=16, N=5</p> <p>1011: DIV=16, N=6</p> <p>1100: DIV=16, N=8</p> <p>1101: DIV=32, N=5</p> <p>1110: DIV=32, N=6</p> <p>1111: DIV=32, N=8</p> <p>采样频率=定时器时钟频率/DIV; 滤波长度=N, 每 N 个事件产生一个跳变。</p>
13:12	ETPCFG	R/W	<p>配置外部触发信号预分频器 (External Trigger Prescaler Configure)</p> <p>ETR (外部触发输入) 的信号经过分频后为 ETRP, ETRP 的信号频率最多是 TMR1CLK 频率的 1/4; 当 ETR 频率过高时, 须经过分频降低 ETRP 的频率。</p> <p>00: 禁用预分频器;</p> <p>01: ETR 信号 2 分频</p>

位/域	名称	R/W	描述
			10: ETR 信号 4 分频 11: ETR 信号 8 分频
14	ECEN	R/W	使能外部时钟模式 2 (External Clock Enable Mode2) 0: 禁止 1: 使能 设置 ECEN 位与选择外部时钟模式 1 将 TRGI 连接到 ETRF 具有相同作用; 从模式 (复位、门控、触发) 可以与外部时钟模式 2 同时使用, 但此时 TRGI 不能连到 ETRF; 当外部时钟模式 1 和外部时钟模式 2 同时使能时, 外部时钟的输入是 ETRF。
15	ETPOL	R/W	配置外部触发极性 (External Trigger Polarity Configure) 该位决定外部触发 ETR 是否反相。 0: 外部触发 ETR 不反相, 高电平或上升沿有效 1: 外部触发 ETR 反相, 低电平或下降沿有效

表格 49 TMR1 内部触发连接

从定时器	ITR0 (TS=000)	ITR1 (TS=001)	ITR2 (TS=010)	ITR3 (TS=011)
TMR1	TMR4	TMR2	TMR3	-

14.6.4 DMA/中断使能寄存器 (TMRx_DIEN)

偏移地址: 0x0C

复位值: 0x0000

位/域	名称	R/W	描述
0	UIEN	R/W	使能更新中断 (Update Interrupt Enable) 0: 禁止 1: 使能
1	CC1IEN	R/W	使能捕获/比较通道 1 中断 (Capture/Compare Channel1 Interrupt Enable) 0: 禁止 1: 使能
2	CC2IEN	R/W	使能捕获/比较通道 2 中断 (Capture/Compare Channel2 Interrupt Enable) 0: 禁止 1: 使能
3	CC3IEN	R/W	使能捕获/比较通道 3 中断 (Capture/Compare Channel3 Interrupt Enable) 0: 禁止 1: 使能
4	CC4IEN	R/W	使能捕获/比较通道 4 中断 (Capture/Compare Channel4 Interrupt Enable) 0: 禁止 1: 使能
5	COMIEN	R/W	使能 COM 中断 (COM Interrupt Enable) 0: 禁止 1: 使能
6	TRGIEN	R/W	使能触发中断 (Trigger Interrupt Enable) 0: 禁止

位/域	名称	R/W	描述
			1: 使能
7	BRKIEN	R/W	使能刹车中断 (Break Interrupt Enable) 0: 禁止 1: 使能
8	UDIEN	R/W	使能更新的 DMA 请求 (Update DMA Request Enable) 0: 禁止 1: 使能
9	CC1DEN	R/W	使能捕获/比较通道 1 的 DMA 请求 (Capture/Compare Channel1 DMA Request Enable) 0: 禁止 1: 使能
10	CC2DEN	R/W	使能捕获/比较通道 2 的 DMA 请求 (Capture/Compare Channel2 DMA Request Enable) 0: 禁止 1: 使能
11	CC3DEN	R/W	使能捕获/比较通道 3 的 DMA 请求 (Capture/Compare Channel3 DMA Request Enable) 0: 禁止 1: 使能
12	CC4DEN	R/W	使能捕获/比较通道 4 的 DMA 请求 (Capture/Compare Channel4 DMA Request Enable) 0: 禁止 1: 使能
13	COMDEN	R/W	使能的 DMA 请求 (COM DMA Request Enable) 0: 禁止 1: 使能
14	TRGDEN	R/W	使能触发 DMA 请求 (Trigger DMA Request Enable) 0: 禁止 1: 使能
15	保留		

14.6.5 状态寄存器 (TMRx_STS)

偏移地址: 0x10

复位值: 0x0000

位/域	名称	R/W	描述
0	UIFLG	RC_W0	产生更新事件中断标志位 (Update Event Interrupt Generate Flag) 0: 没有发生更新事件中断 1: 发生更新事件中断 计数器数值重新装载或重新初始化时, 会产生更新事件, 该位由硬件置 1, 软件清 0; 更新事件的产生的情况有以下情况: (1) TMRx_CTRL1 寄存器的 UD=0, 重复计数器数值上/下溢时产生更新事件; (2) TMRx_CTRL1 寄存器的 URSSEL=0 和 UD=0, 配置 TMRx_CEG 寄存器的 UEG=1 产生更新事件, 需要通过软件初始化计数器;

位/域	名称	R/W	描述
			(3) TMRx_CTRL1 寄存器的 URSSEL=0 和 UD=0, 计数器被触事件初始化时产生更新事件。
1	CC1IFLG	RC_W0	捕获/比较通道 1 中断标志 (Captuer/Compare Channel1 Interrupt Flag) 当捕获比较通道 1 配置为输出时: 0: 无匹配发生 1: TMRx_CNT 的值与 TMRx_CC1 的值相匹配 当捕获比较通道 1 配置为输入时: 0: 没有发生输入捕获 1: 发生输入捕获 捕获事件发生时由硬件置 1, 可以由软件清 0 或者读 TMRx_CC1 寄存器时清 0。
2	CC2IFLG	RC_W0	捕获/比较通道 2 中断标志 (Captuer/Compare Channel2 Interrupt Flag) 参考 STS_CC1IFLG
3	CC3IFLG	RC_W0	捕获/比较通道 3 中断标志 (Captuer/Compare Channel3 Interrupt Flag) 参考 STS_CC1IFLG
4	CC4IFLG	RC_W0	捕获/比较通道 4 中断标志 (Captuer/Compare Channel4 Interrupt Flag) 参考 STS_CC1IFLG
5	COMIFLG	RC_W0	产生 COM 事件中断标志 (COM Event Interrupt Generate Flag) 0: 无 COM 事件产生 1: COM 中断等待响应 产生 COM 事件后, 该位由硬件置 1, 软件清 0。
6	TRGIFLG	RC_W0	产生触发事件中断标志 (Trigger Event Interrupt Generate Flag) 0: 没有发生触发事件中断 1: 发生触发事件中断 发生触发事件时, 该位由硬件置 1, 软件清 0。
7	BRKIFLG	RC_W0	产生刹车事件中断标志 (Brake Event Interrupt Generate Flag) 0: 没有发生刹车事件 1: 发生刹车事件 刹车输入有效的情况下, 该位由硬件置 1; 无效的情况下, 可以通过软件清 0。
8	保留		
9	CC1RCFLG	RC_W0	捕获/比较通道 1 重复捕获标志 (Captuer/Compare Channel1 Repetition Capture Flag) 0: 没有发生重复捕获 1: 发生重复捕获 计数器的值被捕获到 TMRx_CC1 寄存器中, 此时 CC1IFLG=1; 只有当通道被配置为输入捕获时, 该位由硬件置 1, 软件清 0。
10	CC2RCFLG	RC_W0	捕获/比较通道 2 重复捕获标志 (Captuer/Compare Channel2 Repetition Capture Flag) 参考 STS_CC1RCFLG
11	CC3RCFLG	RC_W0	捕获/比较通道 3 重复捕获标志 (Captuer/Compare Channel3 Repetition Capture Flag) 参考 STS_CC1RCFLG

位/域	名称	R/W	描述
12	CC4RCFLG	RC_W0	捕获/比较通道 4 重复捕获标志 (Captuer/Compare Channel4 Repetition Capture Flag) 参考 STS_CC1RCFLG
15:13	保留		

14.6.6 控制事件产生寄存器 (TMRx_CEG)

偏移地址: 0x14

复位值: 0x0000

位/域	名称	R/W	描述
0	UEG	W	产生更新事件 (Update Event Generate) 0: 无效 1: 初始化计数器, 产生更新事件 此位由软件置 1, 硬件清 0。 注意: 产生更新事件时, 预分频器的计数器会清 0, 但是预分频系数不变。如果在向下计数模式下计数器会读取 TMRx_AUTORLD 的值; 如果在中央对齐模式下或者向上计数模式中计数器会被清 0。
1	CC1EG	W	产生捕获/比较通道 1 事件 (Capture/Compare Channel1 Event Generation) 0: 无效 1: 产生捕获/比较事件 该位由软件置 1, 硬件自动清 0。 如果通道 1 处于输出模式 当 CC1IFLG=1 时, 如果设置了 CC1IEN 和 CC1DEN 位, 则产生相应的中断和 DMA 请求。 如果通道 1 处于输入模式 捕获计数器的值存储在 TMRx_CC1 寄存器中; 配置 CC1IFLG=1, 如果还设置了 CC1IEN 和 CC1DEN 位, 则产生相应的中断和 DMA 请求; 如果此时 CC1IFLG=1, 则需要配置 CC1RCFLG=1。
2	CC2EG	W	产生捕获/比较通道 2 事件 (Capture/Compare Channel2 Event Generation) 参考 CC1EG 描述
3	CC3EG	W	产生捕获/比较通道 3 事件 (Capture/Compare Channel3 Event Generation) 参考 CC1EG 描述
4	CC4EG	W	产生捕获/比较通道 4 事件 (Capture/Compare Channel4 Event Generation) 参考 CC1EG 描述
5	COMG	W	产生捕获/比较控制更新事件 (Capture/Compare Control Update Event Generate) 0: 无效 1: 产生捕获/比较更新事件 该位由软件置 1, 硬件自动清 0。 注: COMG 位只有在互补输出的通道有效。
6	TEG	W	产生触发事件 (Trigger Event Generate) 0: 无效 1: 产生触发事件 该位由软件置 1, 硬件自动清 0。

位/域	名称	R/W	描述
7	BEG	W	产生刹车事件 (Brake Event Generate) 0: 无效 1: 产生刹车事件 该位由软件置 1, 硬件自动清 0。
15:8	保留		

14.6.7 捕获/比较模式寄存器 1 (TMRx_CCM1)

偏移地址: 0x18

复位值: 0x0000

可通过 CCxSEL 位配置定时器为输入 (捕获模式) 或输出 (比较模式)。该寄存器其它位的作用在输入和输出模式下不同, 同一个位在输出模式和输入模式下的功能是不同的。寄存器中的 OCxx 描述了通道在输出模式下的功能, 寄存器中的 ICxx 描述了通道在输入模式下的功能。

输出比较模式:

位/域	名称	R/W	描述
1:0	CC1SEL	R/W	选择捕获/比较通道 1 (Capture/Compare Channel1 Select) 该位定义了输入/输出的方向以及选择输入引脚。 00: CC1 通道为输出 01: CC1 通道为输入, IC1 映射在 TI1 上 10: CC1 通道为输入, IC1 映射在 TI2 上 11: CC1 通道为输入, IC1 映射在 TRC 上, 仅工作在内部触发输入 注: 该位仅在通道关闭时 (TMRx_CCEN 寄存器的 CC1EN=0 时) 可写。
2	OC1FEN	R/W	快速使能输出比较通道 1 (Output Compare Channel1 Fast Enable) 0: 禁止 1: 使能 该位用来提高捕获/比较输出对触发输入事件的响应。
3	OC1PEN	R/W	使能输出比较通道 1 预装载 (Output Compare Channel1 Preload Enable) 0: 禁止预装载功能, 通过程序写入 TMRx_CC1 寄存器的数值, 会马上起作用。 1: 启用预装载功能, 通过程序写入 TMRx_CC1 寄存器的数值, 会在产生更新事件后起作用。 注: 当保护级别为 3 级时且通道配置为输出时, 该位不能被修改。当不确定预装载寄存器情况, 仅在单脉冲模式 (SPMEN=1) 下, 可以使用 PWM 模式, 否则不确定其接下来的输出比较结果。
6:4	OC1MOD	R/W	配置输出比较通道 1 模式 (Output Compare Channel1 Mode Configure) 000: 冻结。输出比较对 OC1REF 无影响 001: 匹配时输出置为高。计数器的值和捕获比较寄存器的值 CCx 发生匹配时, 强制 OC1REF 为高电平 010: 匹配时输出置为低。计数器的值和捕获比较寄存器的值发生匹配时, 强制 OC1REF 为低电平 011: 匹配时输出翻转。计数器的值和捕获比较寄存器的值发生匹配时, 翻转 OC1REF 的电平 100: 强制输出为低。强制 OC1REF 为低电平 101: 强制输出为高。强制 OC1REF 为高电平 110: PWM 模式 1 (计数器值 < 输出比较值时置为高, 否则反之)

位/域	名称	R/W	描述
			111: PWM 模式 2 (计数器值>输出比较值时置为高, 否则反之) 注: 当保护级别为 3 级时且通道配置为输出时, 该位不能被修改。在 PWM 模式 1 和 2 中, OC1REF 电平在比较结果改变或者输出比较模式从冻结模式转换到 PWM 模式时改变。
7	OC1CEN	R/W	使能输出比较通道 1 清除 (Output Compare Channel1 Clear Enable) 0: OC1REF 不受 ETRF 输入影响。 1: 检测到 ETRF 输入高电平时, OC1REF=0
9:8	CC2SEL	R/W	通道 2 模式选择 (Capture/Compare Channel2 Select) 该位定义了输入/输出的方向以及选择输入引脚。 00: CC2 通道为输出 01: CC2 通道为输入, IC2 映射在 TI2 上 10: CC2 通道为输入, IC2 映射在 TI1 上 11: CC2 通道为输入, IC2 映射在 TRC 上, 仅工作在内部触发输入 注意: 该位仅在通道关闭时 (TMRx_CCEN 寄存器的 CC2EN=0 时) 可写。
10	OC2FEN	R/W	快速使能输出比较通道 2 (Output Compare Channel2 Preload Enable)
11	OC2PEN	R/W	使能输出比较通道 2 缓冲 (Output Compare Channel2 Buffer Enable)
14:12	OC2MOD	R/W	输出比较通道 2 模式 (Output Compare Channel1 Mode)
15	OC2CEN	R/W	使能输出通道 2 比较清除 (Output Compare Channel2 Clear Enable)

输入捕获模式:

位/域	名称	R/W	描述
1:0	CC1SEL	R/W	选择输入/捕获通道 1 (Capture/Compare Channel1 Select) 00: CC1 通道为输出 01: CC1 通道为输入, IC1 映射在 TI1 上 10: CC1 通道为输入, IC1 映射在 TI2 上 11: CC1 通道为输入, IC1 映射在 TRC 上, 仅工作在内部触发输入 注意: 该位仅在通道关闭时 (TMRx_CCEN 位的 CC1EN=0 时) 可写。
3:2	IC1PSC	R/W	配置输入捕获通道 1 预分频因子 (Input Capture Channel1 Prescaler Configure) 00: PSC=1 01: PSC=2 10: PSC=4 11: PSC=8 PSC 是预分频因子, 每 PSC 个事件触发一次捕获。
7:4	IC1F	R/W	配置输入捕获通道 1 滤波器 (Input Capture Channel1 Filter Configure) 0000: 禁用滤波器, 以 f_{DTS} 采样 0001: DIV=1, N=2 0010: DIV=1, N=4 0011: DIV=1, N=8 0100: DIV=2, N=6 0101: DIV=2, N=8 0110: DIV=4, N=6 0111: DIV=4, N=8 1000: DIV=8, N=6 1001: DIV=8, N=8

位/域	名称	R/W	描述
			1010: DIV=16, N=5 1011: DIV=16, N=6 1100: DIV=16, N=8 1101: DIV=32, N=5 1110: DIV=32, N=6 1111: DIV=32, N=8 采样频率=定时器时钟频率/DIV; 滤波器长度=N, 表示每 N 个事件产生一个跳变。
9:8	CC2SEL	R/W	选择捕获/比较通道 2 (Capture/Compare Channel2 Select) 00: CC2 通道为输出 01: CC2 通道为输入, IC2 映射在 TI1 上 10: CC2 通道为输入, IC2 映射在 TI2 上 11: CC2 通道为输入, IC2 映射在 TRC 上, 仅工作在内部触发输入 注意: 该位仅在通道关闭时 (TMRx_CCEN 寄存器的 CC2EN=0 时) 可写。
11:10	IC2PSC	R/W	配置输入捕获通道 2 预分频因子 ((Input Capture Channel2 Prescaler Configure))
15:12	IC2F	R/W	配置输入捕获通道 2 滤波器 (Input Capture Channel2 Filter Configure)

14.6.8 捕获/比较模式寄存器 2 (TMRx_CCM2)

偏移地址: 0x1C

复位值: 0x0000

参看以上 CCM1 寄存器的描述。

输出比较模式:

位/域	名称	R/W	描述
1:0	CC3SEL	R/W	选择捕获/比较通道 3 (Capture/Compare Channel1 Select) 该位定义了输入/输出的方向以及选择输入引脚。 00: CC3 通道为输出 01: CC3 通道为输入, IC3 映射在 TI3 上 10: CC3 通道为输入, IC3 映射在 TI4 上 11: CC3 通道为输入, IC3 映射在 TRC 上, 仅工作在内部触发输入 注意: 该位仅在通道关闭时 (TMRx_CCEN 寄存器的 CC3EN=0 时) 可写。
2	OC3FEN	R/W	快速使能输出比较通道 3 (Output Compare Channel3 Fast Enable) 0: 禁止 1: 使能 该位用来提高捕获/比较输出对触发输入事件的响应。
3	OC3PEN	R/W	使能输出比较通道 3 预装载 (Output Compare Channel3 Preload Enable)
6:4	OC3MOD	R/W	配置输出比较通道 3 模式 (Output Compare Channel3 Mode Configure)
7	OC3CEN	R/W	使能输出比较通道 3 清除 (Output Compare Channel3 Clear Enable) 0: OC3REF 不受 ETRF 输入影响。 1: 检测到 ETRF 输入高电平时, OC1REF=0

位/域	名称	R/W	描述
9:8	CC4SEL	R/W	选择通道 4 模式 (Capture/Compare Channel4 Select) 该位定义了输入/输出的方向以及选择输入引脚。 00: CC4 通道为输出 01: CC4 通道为输入, IC4 映射在 TI4 上 10: CC4 通道为输入, IC4 映射在 TI3 上 11: CC4 通道为输入, IC4 映射在 TRC 上, 仅工作在内部触发输入 注意: 该位仅在通道关闭时 (TMRx_CCEN 寄存器的 CC4EN=0 时) 可写。
10	OC4FEN	R/W	快速使能输出比较通道 4 (Output Compare Channel4 Preload Enable)
11	OC4PEN	R/W	使能输出比较通道 4 缓冲 (Output Compare Channel4 Buffer Enable)
14:12	OC4MOD	R/W	配置输出比较通道 4 模式 (Output Compare Channel4 Mode Configure)
15	OC4CEN	R/W	使能输出通道 4 比较清除 (Output Compare Channel4 Clear Enable)

输入捕获模式:

位/域	名称	R/W	描述
1:0	CC3SEL	R/W	选择输入/捕获通道 3 (Capture/Compare Channel3 Select) 00: CC3 通道为输出 01: CC3 通道为输入, IC3 映射在 TI3 上 10: CC3 通道为输入, IC3 映射在 TI4 上 11: CC3 通道为输入, IC3 映射在 TRC 上, 仅工作在内部触发输入 注意: 该位仅在通道关闭时 (TMRx_CCEN 寄存器的 CC3EN=0 时) 可写。
3:2	IC3PSC	R/W	配置输入捕获通道 3 预分频因子 (Input Capture Channel3 Prescaler Configure) 00: PSC=1 01: PSC=2 10: PSC=4 11: PSC=8 PSC 是预分频因子, 每 PSC 个事件触发一次捕获。
7:4	IC3F	R/W	配置输入捕获通道 3 滤波器 (Input Capture Channel3 Filter Configure)
9:8	CC4SEL	R/W	选择捕获/比较通道 4 (Capture/Compare Channel4 Select) 00: CC4 通道为输出 01: CC4 通道为输入, IC4 映射在 TI4 上 10: CC4 通道为输入, IC4 映射在 TI3 上 11: CC4 通道为输入, IC4 映射在 TRC 上, 仅工作在内部触发输入 注意: 该位仅在通道关闭时 (TMRx_CCEN 寄存器的 CC4EN=0 时) 可写。
11:10	IC4PSC	R/W	配置输入捕获通道 4 预分频因子 ((Input Capture Channel4 Prescaler Configure)
15:12	IC4F	R/W	配置输入捕获通道 4 滤波器 (Input Capture Channel4 Filter Configure)

14.6.9 捕获/比较使能寄存器 (TMRx_CCEN)

偏移地址: 0x20

复位值: 0x0000

位/域	名称	R/W	描述
0	CC1EN	R/W	<p>使能捕获/比较通道 1 输出 (Capture/Compare Channel1 Output Enable)</p> <p>捕获/比较通道 1 配置为输出时:</p> <p>0: 禁止输出</p> <p>1: 开启输出</p> <p>捕获/比较通道 1 配置为输入时:</p> <p>该位决定了计数器的值 CNT 是否能捕获进入 TMRx_CC1 寄存器中</p> <p>0: 禁止捕获</p> <p>1: 开启捕获</p>
1	CC1POL	R/W	<p>配置捕获/比较通道 1 输出极性 (Capture/Compare Channel1 Output Polarity Configure)</p> <p>CC1 通道配置为输出时:</p> <p>0: OC1 高电平有效</p> <p>1: OC1 低电平有效</p> <p>CC1 通道配置为输入时:</p> <p>CC1POL 和 CC1NPOL 同时控制触发或捕获的信号 TI1FP1 和 TI2FP1 的极性</p> <p>00: 不反相/上升沿:</p> <p>TIxFP1 不反相 (门控、编码器模式下触发), 在 TIxFP1 的上升沿捕获 (复位触发、捕获、外部时钟和触发模式)。</p> <p>01: 反相/下降沿:</p> <p>TIxFP1 反相 (门控、编码器模式下触发), 在 TIxFP1 的上升沿捕获 (复位触发、捕获、外部时钟和触发模式)。</p> <p>10: 保留</p> <p>11: 不反相/上升和下降沿:</p> <p>TIxFP1 不反相 (门控模式下触发, 不能用于编码器模式), 在 TIxFP1 的上升沿捕获 (复位触发、捕获、外部时钟和触发模式)。</p>
2	CC1NEN	R/W	<p>使能捕获/比较通道 1 互补输出 (Capture/Compare Channel1 Complementary Output Enable)</p> <p>0: 禁止</p> <p>1: 开启</p>
3	CC1NPOL	R/W	<p>捕获/比较通道 1 互补输出极性 (Capture/Compare Channel1 Complementary Output Polarity)</p> <p>CC1 通道配置为输出时</p> <p>0: OC1N 高电平有效</p> <p>1: OC1N 低电平有效</p> <p>CC1 通道配置为输入时</p> <p>该位与 CC1POL 一起定义 TI1FP1 和 TI2FP1 的极性</p> <p>注:</p> <p>在互补输出的通道上, 该位时预装载的, 如果 TMRx_CTRL2 的 CCPEN=1, CC1NPOL 只有在生成换向事件时才能从预加载位中获取新值。</p> <p>当保护级别为 2 或者 3 时, 该位不能被修改</p>
4	CC2EN	R/W	<p>使能捕获/比较通道 2 输出 (Capture/Compare Channel2 Output Enable)</p> <p>参考 CCEN_CC1EN</p>

位/域	名称	R/W	描述
5	CC2POL	R/W	配置捕获/比较通道 2 输出极性 (Capture/Compare Channel2 Output Polarity Configure) 参考 CCEN_CC1POL
6	CC2NEN	R/W	使能捕获/比较通道 1 互补输出 (Capture/Compare Channel1 Complementary Output Enable) 参考 CCEN_CC1NEN
7	CC2NPOL	R/W	配置捕获/比较通道 2 互补输出极性 (Capture/Compare Channel2 Complementary Output Polarity Configure) 参考 CCEN_CC1NPOL
8	CC3EN	R/W	使能捕获/比较通道 3 输出 (Capture/Compare Channel3 Output Enable) 参考 CCEN_CC1EN
9	CC3POL	R/W	配置捕获/比较通道 3 输出极性 (Capture/Compare Channel3 Output Polarity Configure) 参考 CCEN_CC1POL
10	CC3NEN	R/W	使能捕获/比较通道 3 互补输出 (Capture/Compare Channel3 Complementary Output Enable) 参考 CCEN_CC1NEN
11	CC3NPOL	R/W	配置捕获/比较通道 3 互补输出极性 (Capture/Compare Channel3 Complementary Output Polarity Configure) 参考 CCEN_CC1NPOL
12	CC4EN	R/W	使能捕获/比较通道 4 输出 (Capture/Compare Channel4 Output Enable) 参考 CCEN_CC1EN
13	CC4POL	R/W	捕获/比较通道 4 输出极性 (Capture/Compare Channel4 Output Polarity) 参考 CCEN_CC1POL
15:14	保留		

14.6.10 计数器寄存器 (TMRx_CNT)

偏移地址: 0x24

复位值: 0x0000

位/域	名称	R/W	描述
15:0	CNT	R/W	计数器数值 (Counter Value)

14.6.11 预分频寄存器 (TMRx_PSC)

偏移地址: 0x28

复位值: 0x0000

位/域	名称	R/W	描述
15:0	PSC	R/W	预分频器数值 (Prescaler Value) 计数器的时钟频率 (CK_CNT) = $f_{CK_PSC} / (PSC+1)$

14.6.12 自动重装载寄存器 (TMRx_AUTORLD)

偏移地址: 0x2C

复位值: 0xFFFF

位/域	名称	R/W	描述
15:0	AUTORLD	R/W	自动重装数值 (Auto Reload Value) 自动重装载的值为空时, 计数器不进行计数。

14.6.13 重复计数寄存器 (TMRx_REPCNT)

偏移地址: 0x30

复位值: 0x0000

位/域	名称	R/W	描述
7:0	REPCNT	R/W	重复计数数值 (Repetition Counter Value) 当重复计数器计数值减为 0 时产生更新事件, 计数器重新从 REPCNT 数值开始计数; 新写入该寄存器的新值只有在下次周期更新事件发生时才有效。
15:8			保留

14.6.14 通道 1 捕获/比较寄存器 (TMRx_CC1)

偏移地址: 0x34

复位值: 0x0000

位/域	名称	R/W	描述
15:0	CC1	R/W	捕获/比较通道 1 数值 (Capture/Compare Channel1 Value) 捕获/比较通道 1 配置为输入模式: CC1 包含上次输入捕获通道 1 事件传输的计数器数值。 捕获/比较通道 1 配置为输出模式 CC1 包含了当前装入捕获/比较寄存器数值。 捕获比较通道 1 的值 CC1 与计数器的值 CNT 比较, 在 OC1 上产生输出信号。 当输出比较预装载禁止 (TMRx_CCM1 寄存器的 OC1PEN=0) 时, 写入的数值会立即影响输出比较结果; 当输出比较预装载使能 (TMRx_CCM1 寄存器的 OC1PEN=1) 时, 写入的值会在产生更新事件时影响输出比较结果。

14.6.15 通道 2 捕获/比较寄存器 (TMRx_CC2)

偏移地址: 0x38

复位值: 0x0000

位/域	名称	R/W	描述
15:0	CC2	R/W	捕获/比较通道 2 数值 (Capture/Compare Channel2 Value) 参考 TMRx_CC1

14.6.16 通道 3 捕获/比较寄存器 (TMRx_CC3)

偏移地址: 0x3C

复位值: 0x0000

位/域	名称	R/W	描述
15:0	CC3	R/W	捕获/比较通道 3 数值 (Capture/Compare Channel3 Value) 参考 TMRx_CC1

14.6.17 通道 4 捕获/比较寄存器 (TMRx_CC4)

偏移地址: 0x40

复位值: 0x0000

位/域	名称	R/W	描述
15:0	CC4	R/W	捕获/比较通道 4 数值 (Capture/Compare Channel4 Value) 参考 TMRx_CC1

14.6.18 刹车和死区寄存器 (TMRx_BDT)

偏移地址: 0x44

复位值: 0x0000

注: 根据锁定设置, AOEN、BRKPOL、BRKEN、IMOS、RMOS 和 DTS[7:0]位均可被写保护, 有必要在第一次写入 TMRx_BDT 寄存器时对它们进行配置。

位/域	名称	R/W	描述
7:0	DTS	R/W	<p>设置互补输出通道的死区持续时间 (Dead Time Setup)</p> <p>DT 为死区持续时间, DT 与寄存器 DTS 的关系如下:</p> <p>DTS[7:5]=0xx=>DT=DTS[7:0]×T_{DTS}, T_{DTS}=TDTS;</p> <p>DTS[7:5]=10x=>DT= (64+DTS[5:0])×T_{DTS}, T_{DTS}=2×T_{DTS};</p> <p>DTS[7:5]=110=>DT= (32+DTS[4:0])×T_{DTS}, T_{DTS}=8×T_{DTS};</p> <p>DTS[7:5]=111=>DT= (32+DTS[4:0]) ×T_{DTS}, T_{DTS}=16×T_{DTS};</p> <p>例: 假设 T_{DTS}=125ns (8MHz), 死区时间设置如下:</p> <p>若步长时间为 125ns, 可设置死区时间范围是 0 到 15875ns;</p> <p>若步长时间为 250ns, 可设置死区时间范围是 16μs 到 31750ns;</p> <p>若步长时间为 1μs, 可设置死区时间范围是 32μs 到 63μs;</p> <p>若步长时间为 2μs, 可设置死区时间范围是 64μs 到 126μs。</p> <p>注: 一旦 LOCK 级别 (TMRx_BDT 寄存器中的 LOCKCFG 位) 设为 1、2 或 3, 则不能修改这些位。</p>
9:8	LOCKCFG	R/W	<p>配置锁定写保护模式 (Lock Write Protection Mode Configure)</p> <p>00: 无锁定写保护, 可直接写寄存器</p> <p>01: 锁定写保护级别 1</p> <p>不能写入 TMRx_BDT 的 DTS、BRKEN、BRKPOL、AOEN 位和 TMRx_CTRL2 寄存器的 OCxOIS 和 OCxNOIS 位。</p> <p>10: 锁定写保护级别保护级别 2</p> <p>不能写入保护级别 1 的所有位, 也不能写入 TMRx_CCEN 寄存器中的 CCxPOL 和 OCxNPOL 位、TMRx_BDT 寄存器的 RMOS 和 IMOS 位。</p> <p>11: 锁定写保护级别保护级别 3</p> <p>不能写入保护级别 2 的所有位, 也不能写入 TMRx_CCMx 的寄存器的 OCxMOD 和 OCxPEN 位。</p> <p>注意: 在系统复位后, 只能写一次锁定写保护位。</p>
10	IMOS	R/W	<p>配置空闲模式下的关闭状态 (Idle Mode Off-state Configure)</p> <p>空闲模式指 MOEN=0, 关闭是指 CCxEN=0; 该位描述的是在 MOEN=0, CCxEN 由 0 变为 1, 配置该位不同的数值, 对输出波形的影响。</p> <p>0: 禁止 OCx/OCxN 输出</p> <p>1: 若 CCxEN=1, 先在死区期间输出无效电平 (具体电平数值受极性配置影响), 在死区结束后, 输出空闲电平</p>
11	RMOS	R/W	<p>配置运行模式下的关闭状态 (Run Mode Off-state Configure)</p> <p>运行模式指 MOEN=1, 关闭是指 CCxEN=0; 该位描述的是在 MOEN=1, CCxEN 由 0 变为 1 时, 配置该位不同的数值, 对输出波形的影响。</p> <p>0: 禁止 OCx/OCxN 输出</p> <p>1: OCx/OCxN 先输出无效电平 (具体电平数值受极性配置影响)</p>

位/域	名称	R/W	描述
12	BRKEN	R/W	使能刹车功能 (Brake Function Enable) 0: 禁止 1: 使能 注: 当保护级别为 1 时, 该位不能修改。
13	BRKPOL	R/W	配置刹车输入极性 (Brake Polarity Configure) 0: 刹车输入 BRK 在低电平有效 1: 刹车输入 BRK 在高电平有效 注: 当保护级别为 1 时, 该位不能被修改。对该位的写操作需要一个 APB 时钟延迟后才能使用。
14	AOEN	R/W	使能自动输出 (Automatic Output Enable) 0: MOEN 只能被软件置 1 1: MOEN 可以被软件置 1 或者在下一个更新事件时自动置 1 (刹车输入无效) 注: 当保护级别为 1 时, 该位不能修改。
15	MOEN	R/W	使能 PWM 主输出 (PWM Main Output Enable) 0: 禁止 OCx 和 OCxN 输出或者强制输出空闲状态 1: 当设置了 TMRx_CCEN 寄存器的 CCxEN 和 CCxNEN 位, 开启 OCx 和 OCxN 输出 刹车输入有效时硬件异步清 0。 注: 由软件置 1 还是自动置 1 取决于 TMRx_BDT 寄存器的 AOEN 位。

14.6.19 DMA 控制寄存器 (TMRx_DCTRL)

偏移地址: 0x48

复位值: 0x0000

位/域	名称	R/W	描述
4:0	DBADDR	R/W	设置 DMA 基地址 (DMA Base Address Setup) 这些位定义了 DMA 在连续模式下的基地址 (当对 TMRx_DMADDR 寄存器进行读或写时), DBADDR 定义为从 TMRx_CTRL1 寄存器所在地址开始的偏移量: 00000: TMRx_CTRL1 00001: TMRx_CTRL2 00010: TMRx_SMCTRL
7:5	保留		
12:8	DBLEN	R/W	设置 DMA 突发传输长度 (DMA Burst Transfer Length Setup) 这些位定义 DMA 在连续模式下的传输长度、传输的次数, 其中传输的数据可以是 16 位和 8 位。 当读写 TMRx_DMADDR 寄存器时, 定时器进行一次连续传输传输: 00000: 1 次传输 00001: 2 次传输 00010: 3 次传输 10001: 18 次传输 传输的地址公式如下: 传输的地址=TMRx_CTRL1 的地址 (从地址)+DBADDR+DMA 索引; DMA 索引=DBLEN

位/域	名称	R/W	描述
			<p>例如：DBLEN=7, DBADDR=TMR2_CTRL1（从地址）表示待传输数据的地址，则 TMRx_CTRL1 的地址+DBADDR+7，表示了将要写入/读出数据的地址</p> <p>数据的传输将发生在：TMRx_CTRL1 的地址+从 DBADDR 开始的 7 个寄存器。</p> <p>根据设置的 DMA 数据长度不同，数据传输也会发生变化： 当传输数据设置为 16 位时，数据会传输给 7 个寄存器 当传输数据设置为 8 位时，第一个寄存器的数据是第一个的数据的 MSB 位，第二个寄存器的数据是第一个数据的 LSB 位，数据仍然会传输给 7 个寄存器。</p>
15:13			保留

14.6.20 连续模式的 DMA 地址寄存器 (TMRx_DMADDR)

偏移地址：0x4C

复位值：0x0000

位/域	名称	R/W	描述
15:0	DMADDR	R/W	<p>DMA 突发传送寄存器 (DMA Register for Burst Transfer)</p> <p>TMRx_DMADDR 寄存器的读或写操作访问会导致对以下地址所在寄存器的存取操作： TMRx_CTRL1 地址+ (DBADDR+DMA 索引) × 4</p> <p>其中： “TMRx_CTRL1 地址”是控制寄存器 1 (TMRx_CTRL1) 所在的地址； “DBADDR”是 TMRx_DCTRL 寄存器中定义的基地址； “DMA 索引”是由 DMA 自动控制的偏移量，它取决于 TMRx_DCTRL 寄存器中定义的 DBLEN。</p>

14.6.21 通道 1 捕获/比较寄存器互补寄存器 (TMRx_CC1C)

偏移地址：0x50

复位值：0x0000

位/域	名称	R/W	描述
15:0	CC1C	R/W	<p>通道 1 捕获/比较寄存器互补寄存器 (Channel 1 Capture/Compare Register Complementary Register)</p> <p>捕获/比较通道 1 配置为 PWM 输出模式，计数器为中心对齐模式，且非对称 PWM 输出模式使能(TMRx_CTRL3 寄存器的 OC1AEN=1)时，CC1C 包含了当前装入捕获/比较寄存器互补寄存器的数值当计数器为增计数时，捕获比较通道 1 的值 CC1 与计数器的值 CNT 比较，在 OC1 上产生输出信号。 当计数器为减计数时，CC1C 的值与计数器的值 CNT 比较，在 OC1 上产生输出信号。</p> <p>当输出比较预装载禁止 (TMRx_CCM1 寄存器的 OC1PEN=0) 时，写入的数值会立即影响输出比较结果； 当输出比较预装载使能 (TMRx_CCM1 寄存器的 OC1PEN=1) 时，写入的值会在产生更新事件时影响输出比较结果。</p>

14.6.22 通道 2 捕获/比较寄存器互补寄存器 (TMRx_CC2C)

偏移地址：0x54

复位值：0x0000

位/域	名称	R/W	描述
15:0	CC2C	R/W	通道 2 捕获/比较寄存器互补寄存器 (Channel 3 Capture/Compare Register Complementary Register) 参考 TMRx_CC1C

14.6.23 通道 3 捕获/比较寄存器互补寄存器 (TMRx_CC3C)

偏移地址: 0x58

复位值: 0x0000

位/域	名称	R/W	描述
15:0	CC3C	R/W	通道 3 捕获/比较寄存器互补寄存器 (Channel 3 Capture/Compare Register Complementary Register) 参考 TMRx_CC1C

14.6.24 通道 5 捕获/比较寄存器 (TMRx_CC5)

偏移地址: 0x5C

复位值: 0x0000

位/域	名称	R/W	描述
15:0	CC5	R/W	捕获/比较通道 5 数值 (Capture/Compare Channel5 Value) 参考 TMRx_CC1

14.6.25 捕获/比较模式寄存器 3 (TMRx_CCM3)

偏移地址: 0x60

复位值: 0x0000

通道 5 仅有输出比较功能:

位/域	名称	R/W	描述
0	CC5EN	R/W	使能捕获/比较通道 5 输出 (Compare Channel5 Output Enable) 0: 禁止输出 1: 开启输出
1	CC5POL	R/W	配置捕获/比较通道 5 输出极性 (Capture/Compare Channel5 Output Polarity Configure) 0: OC5 高电平有效 1: OC5 低电平有效 注: 一旦 LOCK 级别 (TMRx_BDT 寄存器中的 LOCKCFG 位) 设为 2 或 3, 则不能修改这些位。
2	OC5FEN	R/W	快速使能输出比较通道 5 (Output Compare Channel5 Fast Enable) 0: 禁止 1: 使能 该位用来提高捕获/比较输出对触发输入事件的响应。
3	OC5PEN	R/W	使能输出比较通道 5 预装载 (Output Compare Channel5 Preload Enable) 参考 OC1PEN 位。 注: 一旦 LOCK 级别 (TMRx_BDT 寄存器中的 LOCKCFG 位) 设为 3, 则不能修改这些位。
6:4	OC5MOD	R/W	配置输出比较通道 5 模式 (Output Compare Channel5 Mode Configure) 参考 OC1MOD 位。 注: 一旦 LOCK 级别 (TMRx_BDT 寄存器中的 LOCKCFG 位) 设为 3, 则不能修改这些位。

位/域	名称	R/W	描述
7	OC5CEN	R/W	使能输出比较通道 5 清除 (Output Compare Channel5 Clear Enable) 0: OC5REF 不受 ETRF 输入影响。 1: 检测到 ETRF 输入高电平时, OC5REF=0
8	OC5OIS	R/W	配置 OC5 输出空闲状态 (OC5 Output Idle State Configure) 注: 一旦 LOCK 级别 (TMRx_BDT 寄存器中的 LOCKCFG 位) 设为 1、2 或 3, 则不能修改这些位。
15:9	保留		

14.6.26 控制寄存器 3 (TMRx_CTRL3)

偏移地址: 0x64

复位值: 0x0000

位/域	名称	R/W	描述
0	MMSZE	R/W	计数器归 0 时刻产生 TRGO 信号 (When the counter returns to 0, TRGO signal is generated) 产生的 TRGO 与 MMSEL 和 MMSEL2 选择的 TRGO 信号相叠加。 只在计数器为中心对齐模式下有效, 只在 MMSEL 或 MMSEL2 选择的 TRGO 为脉冲信号时有效(在使能模式、比较模式 1-5 时无效)。 1: 计数器归 0 时产生 TRGO 0: 计数器归 0 时不产生 TRGO
1	MMSPE	R/W	计数器与自动重载寄存器匹配时产生 TRGO 信号 (TRGO Signal is Generated when the Counter Matches the Autoreload Register) 产生的 TRGO 与 MMSEL 和 MMSEL2 选择的 TRGO 信号相叠加。 只在计数器为中心对齐模式下有效, 只在 MMSEL 或 MMSEL2 选择的 TRGO 为脉冲信号时有效(在使能模式、比较模式 1-5 时无效)。 1: 计数器与自动重载寄存器匹配时产生 TRGO 0: 计数器与自动重载寄存器匹配时不产生 TRGO
3:2	保留		
7:4	MMSEL2	R/W	选择定时器主模式下用于 TRGO 的信号 (Master Mode Signal Select) 工作在主模式的定时器的信号可用于 TRGO, 从而影响处在从模式且与主定时器级联的的定时器工作, 具体影响可从模式的定时器配置有关。 0xxx: 由 MMSEL [2:0]选择 TRGO 1000: 不产生 TRGO(只会根据 MMSZE 和 MMSPE 产生 TRGO) 1001: 比较模式 5, OC5REF 用于触发 TRGO 1010: OC4REF 上升沿和下降沿产生 TRGO 1011: OC5REF 上升沿和下降沿产生 TRGO 1100: OC4REF 上升沿和 OC5REF 上升沿产生 TRGO 1101: OC4REF 下降沿和 OC5REF 下降沿产生 TRGO 1110: OC4REF 上升沿和 OC5REF 下降沿产生 TRGO 1111: OC4REF 下降沿和 OC5REF 上升沿产生 TRGO
8	OC1AEN	R/W	使能通道 1 非对称 PWM 输出模式 (Channel 1 Asymmetric PWM Output Mode Enable) 仅在计数器为中心对称模式, 通道 1 配置为 PWM 输出模式时有效。 非对称 PWM 模式下, 计数器增计数时 OC1REF 由 CC1 控制, 计数器减计数时 OC1REF 由 CC1C 控制。 1: 使能非对称 PWM 输出模式 0: CC1C 不影响 OC1REF 输出 注: 一旦 LOCK 级别 (TMRx_BDT 寄存器中的 LOCKCFG 位) 设为 3, 则不能修改这些位。
9	OC2AEN	R/W	配置输出比较通道 2 为非对称模式 (Output Compare Channel2 Asynchronous Mode Configure)

位/域	名称	R/W	描述
			参考 OC1AEN。 注：一旦 LOCK 级别（TMRx_BDT 寄存器中的 LOCKCFG 位）设为 3，则不能修改这些位。
10	OC3AEN	R/W	配置输出比较通道 3 为非对称模式（Output Compare Channel3 Asynchronous Mode Configure） 参考 OC1AEN。 注：一旦 LOCK 级别（TMRx_BDT 寄存器中的 LOCKCFG 位）设为 3，则不能修改这些位。
15:11	保留		

14.6.27 刹车滤波寄存器（TMRx_BKFT）

偏移地址：0x68

复位值：0x0000

位/域	名称	R/W	描述
3:0	BKF	R/W	外部刹车输入滤波器(External Brake Input Filter) 0000: 禁用滤波器，以 f _{DTs} 采样 0001: DIV=1, N=2 0010: DIV=1, N=4 0011: DIV=1, N=8 0100: DIV=2, N=6 0101: DIV=2, N=8 0110: DIV=4, N=6 0111: DIV=4, N=8 1000: DIV=8, N=6 1001: DIV=8, N=8 1010: DIV=16, N=5 1011: DIV=16, N=6 1100: DIV=16, N=8 1101: DIV=32, N=5 1110: DIV=32, N=6 1111: DIV=32, N=8 采样频率=定时器时钟频率/DIV；滤波长度=N，每 N 个事件产生一个跳变。 注：当保护级别为 1 时，该位不能被修改。
15:4	保留		

14.6.28 协处理器寄存器（TMRx_M0CP）

偏移地址：0x6C

复位值：0x4824

位/域	名称	R/W	描述
1:0	CH1_WDATA_SEL	R/W	通道 1 数据选择寄存器(Channel 1 Data Selection Register) 00: CCR1 寄存器写入 M0CP 传出数据的 15-0 位 01: CCR1 寄存器写入 M0CP 传出数据的 31-16 位 10: CCR1 寄存器写入 M0CP 传出数据的 47-32 位 11: 保留
3:2	CH2_WDATA_SEL	R/W	通道 2 数据选择寄存器(Channel 2 Data Selection Register) 00: CCR2 寄存器写入 M0CP 传出数据的 15-0 位 01: CCR2 寄存器写入 M0CP 传出数据的 31-16 位

位/域	名称	R/W	描述
			10: CCR2 寄存器写入 M0CP 传出数据的 47-32 位 11: 保留
5:4	CH3_WDATA_SEL	R/W	通道 3 数据选择寄存器(Channel 3 Data Selection Register) 00: CCR3 寄存器写入 M0CP 传出数据的 15-0 位 01: CCR3 寄存器写入 M0CP 传出数据的 31-16 位 10: CCR3 寄存器写入 M0CP 传出数据的 47-32 位 11: 保留
6	PWM_PERIOD_SEL	R/W	PWM 周期选择寄存器(PWM Cycle Selection Register) 0: 周期为 TMRx_AUTORLDR 影子寄存器的值 1: 周期为 TMRx_AUTORLDR 寄存器的值
7	CCR_NO_BUF	R/W	CCR buffer 使能寄存器(CCR Buffer Enable Register) 0: M0CP 对 CCx/CCxN 寄存器的更新在下次更新事件时生效 1: M0CP 对 CCx/CCxN 寄存器的更新会立即生效
8	PWM_PDCON_SEL	R/W	PDCON 选择寄存器(PDCON Select Register) 0: PDCON 为 0 1: PDCON 为 1
10:9	CH1N_WDATA_SEL	R/W	通道 1N 数据选择寄存器(Channel 1N Data Selection Register) 00: CCR1 寄存器写入 M0CP 传出数据的 31-16 位 01: CCR1 寄存器写入 M0CP 传出数据的 63-48 位 10: CCR1 寄存器写入 M0CP 传出数据的 95-80 位 11: 保留
12:11	CH2N_WDATA_SEL	R/W	通道 2N 数据选择寄存器(Channel 2N Data Selection Register) 00: CCR2 寄存器写入 M0CP 传出数据的 31-16 位 01: CCR2 寄存器写入 M0CP 传出数据的 63-48 位 10: CCR2 寄存器写入 M0CP 传出数据的 95-80 位 11: 保留
14:13	CH3N_WDATA_SEL	R/W	通道 3N 数据选择寄存器(Channel 3N Data Selection Register) 00: CCR3 寄存器写入 M0CP 传出数据的 31-16 位 01: CCR3 寄存器写入 M0CP 传出数据的 63-48 位 10: CCR3 寄存器写入 M0CP 传出数据的 95-80 位 11: 保留
15	SVPWM_EN	R/W	SVPWM 功能使能寄存器(SVPWM Function Enable Register) 0: M0CP 不会更新 CCx/CCxN 中的值 1: M0CP 会更新 CCx/CCxN 中的值

14.6.29 输出控制寄存器 1 (TMRx_OUTPUTCTRL1)

偏移地址: 0x70

复位值: 0x0000

位/域	名称	R/W	描述
0	CH1_FORCE_EN	R/W	通道 1 输出控制使能寄存器 (Channel 1 Outputs the Control Enable Register) 0: 通道 1 输出 PWM 波形 1: 通道 1 的输出受输出控制寄存器 2 的对应位控制
1	CH1N_FORCE_EN	R/W	通道 1 的互补通道输出控制使能寄存器 (Complementary Channel Output Control Enable Register for Channel 1) 0: 通道 1 的互补通道输出 PWM 波形

位/域	名称	R/W	描述
			1: 通道 1 的互补通道的输出受输出控制寄存器 2 的对应位控制
2	CH2_FORCE_EN	R/W	通道 2 输出控制使能寄存器 (Channel 2 Outputs the Control Enable Register) 0: 通道 2 输出 PWM 波形 1: 通道 2 的输出受输出控制寄存器 2 的对应位控制
3	CH2N_FORCE_EN	R/W	通道 2 的互补通道输出控制使能寄存器 (Complementary Channel Output Control Enable Register for Channel 2) 0: 通道 2 的互补通道输出 PWM 波形 1: 通道 2 的互补通道的输出受输出控制寄存器 2 的对应位控制
4	CH3_FORCE_EN	R/W	通道 3 输出控制使能寄存器 (Channel 3 Outputs the Control Enable Register) 0: 通道 3 输出 PWM 波形 1: 通道 3 的输出受输出控制寄存器 2 的对应位控制
5	CH3N_FORCE_EN	R/W	通道 3 的互补通道输出控制使能寄存器 (Complementary Channel Output Control Enable Register for Channel 3) 0: 通道 3 的互补通道输出 PWM 波形 1: 通道 3 的互补通道的输出受输出控制寄存器 2 的对应位控制
6	CH4_FORCE_EN	R/W	通道 4 输出控制使能寄存器 (Channel 4 Outputs the Control Enable Register) 1: 通道 4 的输出受输出控制寄存器 2 的对应位控制 0: 通道 4 输出 PWM 波形
7	CH5_FORCE_EN	R/W	通道 5 输出控制使能寄存器 (Channel 5 Outputs the Control Enable Register) 0: 通道 5 输出 PWM 波形 1: 通道 5 的输出受输出控制寄存器 2 的对应位控制
14:8	保留		
15	OUTPUTCTRL_BUF	R/W	输出控制缓冲使能(Output Control Buffering is Enable) 0: 对输出控制寄存器修改立即生效 1: 对输出控制寄存器的修改在下次更新事件时生效

注：一旦 LOCK 级别 (TMRx_BDT 寄存器中的 LOCKCFG 位) 设为 1 或 2 或 3，则不能修改这个寄存器。

14.6.30 输出控制寄存器 2 (TMRx_OUTPUTCTRL2)

偏移地址: 0x74

复位值: 0x0000

位/域	名称	R/W	描述
0	CH1_FORCE_VALUE	R/W	通道 1 输出电平寄存器(Channel 1 Output Level Register) 0: 通道 1 输出低电平 1: 通道 1 输出高电平
1	CH1N_FORCE_VALUE	R/W	通道 1 的互补通道输出电平寄存器(Complementary Channel Output Level Register for Channel 1) 0: 通道 1 的互补通道输出低电平 1: 通道 1 的互补通道输出高电平
2	CH2_FORCE_VALUE	R/W	通道 2 输出电平寄存器(Channel 2 Output Level Register) 0: 通道 2 输出低电平 1: 通道 2 输出高电平

位/域	名称	R/W	描述
3	CH2N_FORCE_VALUE	R/W	通道 2 的互补通道电平使能寄存器(Complementary Channel Output Level Register for Channel 2) 0: 通道 2 的互补通道输出低电平 1: 通道 2 的互补通道输出高电平
4	CH3_FORCE_VALUE	R/W	通道 3 输出电平寄存器(Channel 3 Output Level Register) 0: 通道 3 输出低电平 1: 通道 3 输出高电平
5	CH3N_FORCE_VALUE	R/W	通道 3 的互补通道输出电平寄存器(Complementary Channel Output Level Register for Channel 3) 0: 通道 3 的互补通道输出低电平 1: 通道 3 的互补通道输出高电平
6	CH4_FORCE_VALUE	R/W	通道 4 输出电平寄存器(Channel 4 Output Level Register) 0: 通道 4 输出低电平 1: 通道 4 输出高电平
7	CH5_FORCE_VALUE	R/W	通道 5 输出电平寄存器(Channel 5 Output Level Register) 0: 通道 5 输出低电平 1: 通道 5 输出高电平
15:8	保留		

注意：必须先配置 TMRx_OUTPUTCTRL2 再配置 TMRx_OUTPUTCTRL1。如果需要更改输出控制缓冲功能并修改 TMRx_OUTPUTCTRLx 中其他控制位的值：需要先写 TMRx_OUTPUTCTRL1 的 OUTPUTCTRL_BUF 后，再依次写 TMRx_OUTPUTCTRL2 寄存器和 TMRx_OUTPUTCTRL1 寄存器。

14.6.31 控制寄存器 4 (TMRx_CTRL4)

偏移地址：0x78

复位值：0x0000

位/域	名称	R/W	描述
0	MM2SZE	R/W	计数器归 0 时刻产生 TRGO2 信号 (When the counter returns to 0, TRGO2 signal is generated) 产生的 TRGO2 与 MM2SEL 选择的 TRGO2 信号相叠加。 只在计数器为中心对齐模式下有效，只在 MM2SEL 选择的 TRGO 为脉冲信号时有效(在使能模式、比较模式 1-5 时无效)。 0: 计数器归 0 时不产生 TRGO2 1: 计数器归 0 时产生 TRGO2
1	MM2SPE	R/W	计数器与自动重载寄存器匹配时产生 TRGO2 信号 (TRGO2 Signal is Generated when the Counter Matches the Autoreload Register) 产生的 TRGO2 与 MM2SEL 选择的 TRGO2 信号相叠加。 只在计数器为中心对齐模式下有效，只在 MM2SEL 选择的 TRGO 为脉冲信号时有效(在使能模式、比较模式 1-5 时无效)。 0: 计数器与自动重载寄存器匹配时不产生 TRGO2 1: 计数器与自动重载寄存器匹配时产生 TRGO2
3:2	保留		
7:4	MM2SEL	R/W	选择定时器主模式下用于 TRGO2 的信号 (Master Mode Signal Select) 工作在主模式的定时器的信号可用于 TRGO2，从而影响处在从模式且与主定时器级联的的定时器工作，具体影响可从模式的定时器配置有关。 0000: 复位，主模式定时器的复位信号用于 TRGO2 0001: 使能，主模式定时器的计数器使能信号用于 TRGO2 0010: 更新，主模式定时器的更新事件用于 TRGO2 0011: 比较脉冲，主模式定时器捕获/比较成功 (CCXIFLG=1) 时输出—

位/域	名称	R/W	描述
			个脉冲信号用于 TRGO2 0100: 比较模式 1, OC1REF 用于触发 TRGO2 0101: 比较模式 2, OC2REF 用于触发 TRGO2 0110: 比较模式 3, OC3REF 用于触发 TRGO2 0111: 比较模式 4, OC4REF 用于触发 TRGO2 1000: 不产生 TRGO2(只会根据 MM2SZE 和 MM2SPE 产生 TRGO2) 1001: 比较模式 5, OC5REF 用于触发 TRGO2 1010: OC4REF 上升沿和下降沿产生 TRGO2 1011: OC5REF 上升沿和下降沿产生 TRGO2 1100: OC4REF 上升沿和 OC5REF 上升沿产生 TRGO2 1101: OC4REF 下降沿和 OC5REF 下降沿产生 TRGO2 1110: OC4REF 上升沿和 OC5REF 下降沿产生 TRGO2 1111: OC4REF 下降沿和 OC5REF 上升沿产生 TRGO2
8	MM3SZE	R/W	计数器归 0 时刻产生 TRGO3 信号 (When the counter returns to 0, TRGO3 signal is generated) 参考 MM2SZE 位。 0: 计数器归 0 时不产生 TRGO3 1: 计数器归 0 时产生 TRGO3
9	MM3SPE	R/W	计数器与自动重载寄存器匹配时产生 TRGO3 信号 (TRGO3 Signal is Generated when the Counter Matches the Autoreload Register) 参考 MM2SPE 位。 0: 计数器与自动重载寄存器匹配时不产生 TRGO3 1: 计数器与自动重载寄存器匹配时产生 TRGO3
11:10	保留		
15:12	MM3SEL	R/W	选择定时器主模式下用于 TRGO3 的信号 (Master Mode Signal Select) 参考 MM2SEL 位。

15 通用定时器（TMR2/3/4）

15.1 简介

通用定时器以时基单元为核心，拥有输入捕获和输出比较等功能，可以用来测量脉冲宽度、频率和占空比，以及产生输出波形。含有一个 16 位或者 32 位的自动重载计数器（实现向上、向下和中央对齐计数）。

定时器和定时器之间是相互独立的，它们可以实现同步和级联。

15.2 主要特征

- (1) 时基单元
 - 计数器：16（TMR3/4）位或者 32（TMR2）位计数器，可以向上、向下和中央对齐计数
 - 预分频器：16 位可编程预分频器
 - 自动重载功能
- (2) 时钟源选择
 - 内部时钟
 - 外部输入
 - 外部触发
 - 内部触发
- (3) 输入功能
 - 计数功能
 - PWM 输入
 - 编码器接口模式
- (4) 输出功能
 - PWM 输出模式
 - 强制输出模式
 - 单脉冲模式
- (5) 定时器的主/从模式控制器
 - 定时器之间可以同步和级联
 - 支持多种从模式、同步信号
- (6) 中断和 DMA 请求事件
 - 更新事件（计数器上/下溢出，计数器初始化）
 - 触发事件（计数器启动、停止、内/外部触发）
 - 输入捕获
 - 输出比较

来自于外部触发接口（ETR）经过极性选择、分频、滤波以后的信号，经过触发输入选择器，连接到从模式控制器，从而控制计数器的工作。

内部触发输入

设置定时器工作于从模式，时钟源为其他定时器的输出信号，此时钟源没有滤波，可以实现定时器之间的同步或级联。主模式的定时器可以对从模式定时器执行复位、启动、停止或提供时钟。

15.4.2 时基单元

通用定时器里的时基单元包含三个寄存器

- 计数器寄存器（CNT）16/32 位
- 自动重载寄存器（AUTORLD）16/32 位
- 预分频器（PSC）16/32 位

计数器 CNT

通用定时器中的计数器中一共有三种计数模式

- 向上计数模式
- 向下计数模式
- 中央对齐模式

向上计数模式

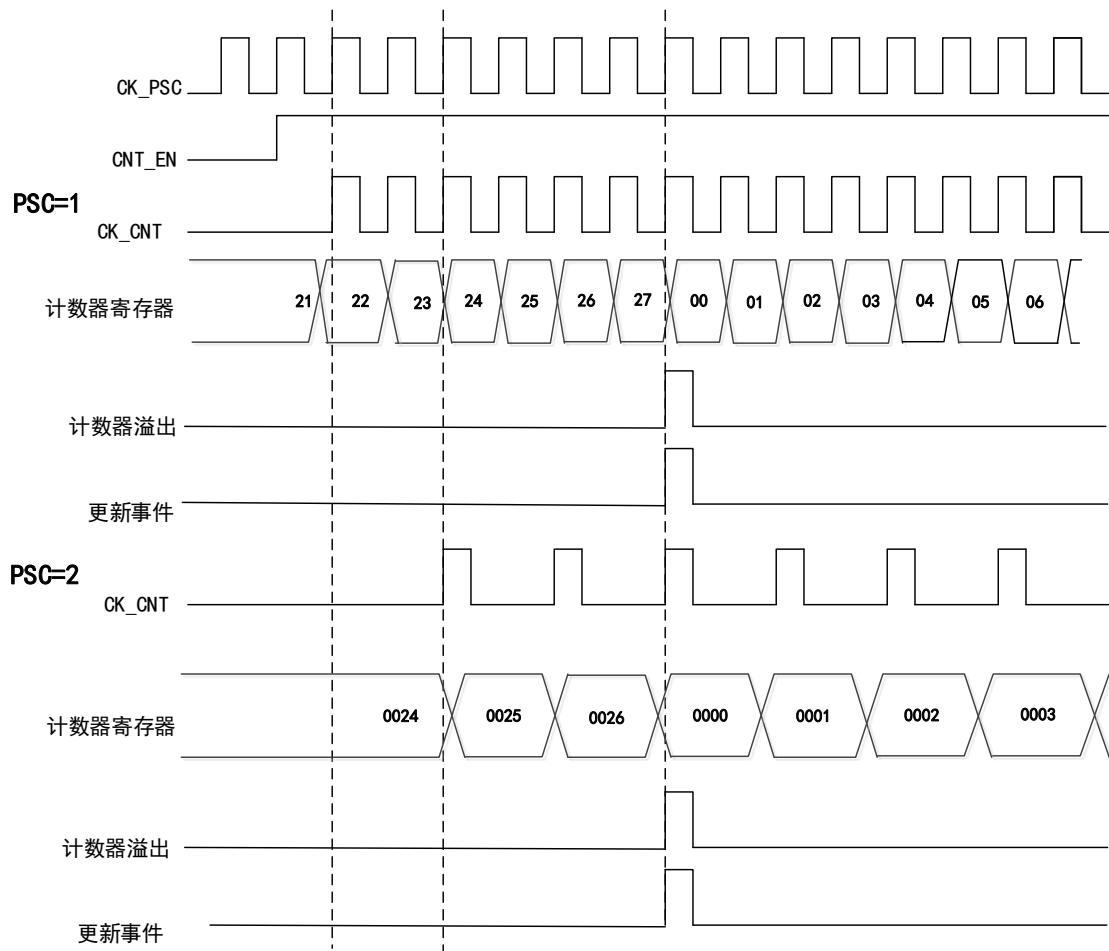
通过配置控制寄存器（TMRx_CTRL1）中的 CNTDIR 位，设置为向上计数模式。

当计数器处于向上计数模式时，计数器从 0 开始向上计数，当每来一个脉冲计数器就会增加 1，一直到计数器（TMRx_CNT）值与自动重载

（TMRx_AUTORLD）的值相等时，计数器会再次从 0 开始计数，此时产生一个计数器向上溢出事件，其中自动重载的值（TMRx_AUTORLD）是提前写入的。

当计数器溢出时，会产生更新事件，此时自动重载的影子寄存器和预分频的缓冲区都将会被更新。可以通过配置控制寄存器 TMRx_CTRL1 中的 UD 位，禁止更新事件。

图 46 向上计数模式下，分频因子为 1 或 2 的时序图



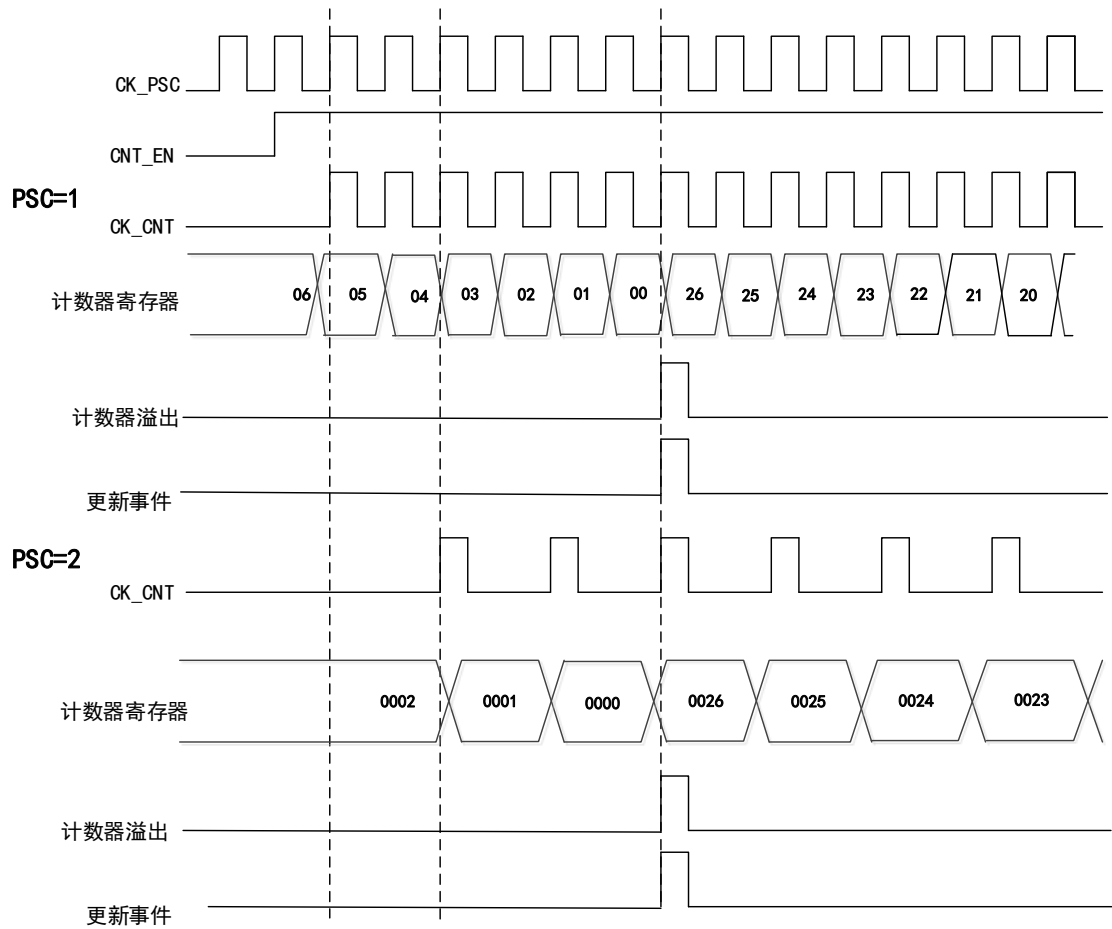
向下计数模式

通过配置控制寄存器（TMRx_CTRL1）中的 CNTDIR 位，设置为向下计数模式。

当计数器处于向下计数模式时，计数器从自动重载的值（TMRx_AUTORLD）开始向下计数，当每来一个脉冲计数器就会减 1，一直减到 0 时，计数器会重新从（TMRx_AUTORLD）开始计数，与此同时便会产生一个计数器向下溢出事件，自动重载的值（TMRx_AUTORLD）是提前写入的。

计数器溢出时，会产生更新事件，此时，自动重载的影子寄存器和预分频的缓冲区都将会被更新。可以配置 TMRx_CTRL1 寄存器中的 UD 位，禁止更新事件。

图 47 向下计数模式下，分频因子为 1 或 2 的时序图

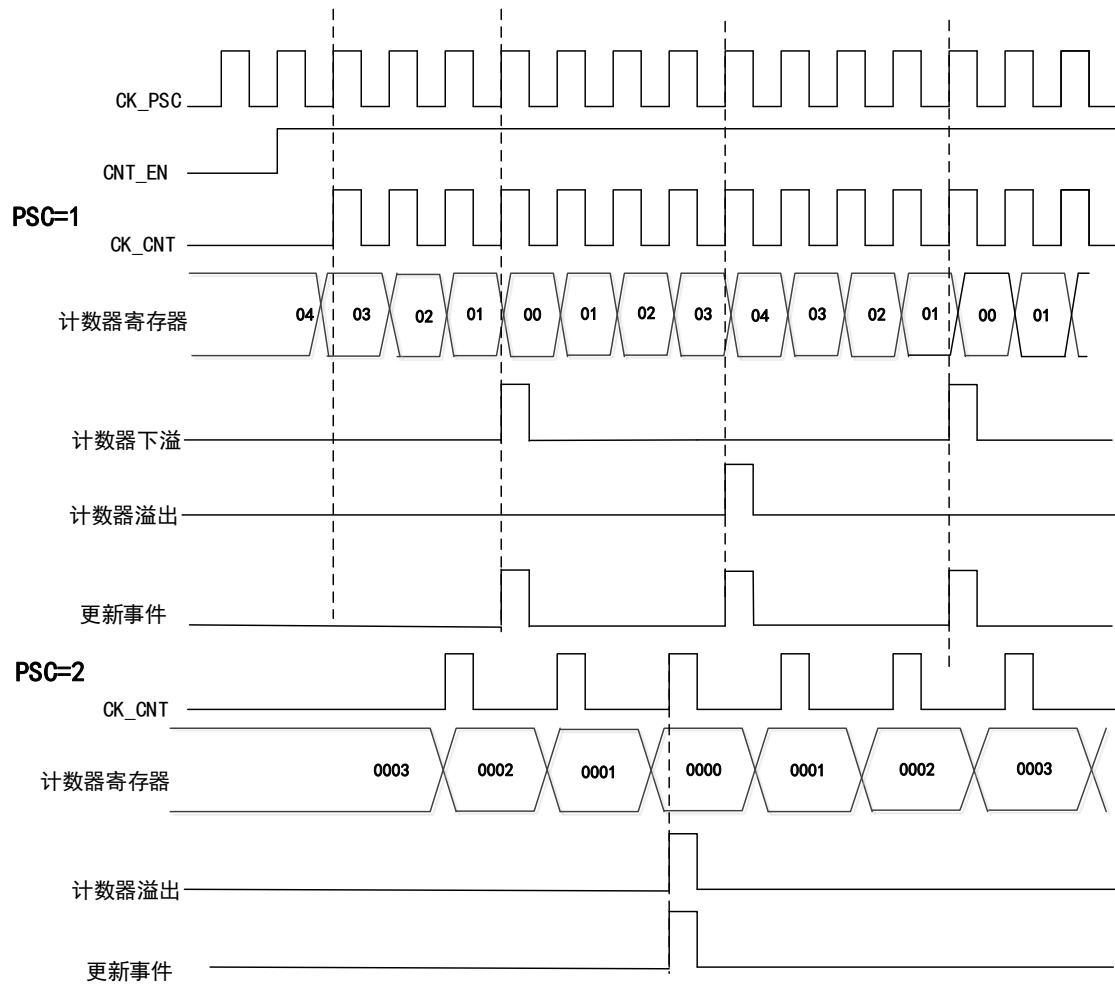


中央对齐模式

通过配置控制寄存器（TMRx_CTRL1）中的 CNTDIR 位，设置为中央对齐模式。

当计数器处于中央对齐模式时，计数器从 0 开始向上计数到自动重载的值（TMRx_AUTORLD），然后从自动重载的值（TMRx_AUTORLD）再向下计数到 0，以此往复，在向上计数时当计数器的值为（AUTORLD-1）时会产生一个计数器上溢事件，在向下计数时计数器的值为 1 时会产生一个计数器下溢事件。

图 48 中央对齐模式下，分频因子为 1 或 2 的时序图



预分频器 PSC

预分频器是 16 位的且是可编程的，它可以将计数器的时钟频率进行 1~65536 之间任意值的分频（由 TMRx_PSC 寄存器控制），经过分频后的时钟将会驱动计数器 CNT 计数。预分频器带有缓冲器，它能够在运行中被改变。

15.4.3 输入捕获

输入捕获通道

通用定时器有四个独立的捕获/比较独立通道，每一个捕获/比较的通道都围绕着一个捕获/比较寄存器。

在输入捕获中，被测量的信号会从定时器的外部引脚 T1/2/3/4 进入首先经过边沿检测器和输入滤波器，然后进入捕获通道，每个捕获通道都有相对应的捕获寄存器，当发生捕获时，计数器 CNT 的值将会被锁存在捕获寄存器 CCx 中。在进入捕获寄存器之前，信号还会经过预分频器，用于设定经过多少事件进行一次捕获。

输入捕获应用

输入捕获用来捕获外部事件，并且可以赋予时间标记表明事件的发生时刻，可以测量脉冲跳变沿事件（测量频率或者脉宽），如：在输入引脚上如果出现了被选择的边沿，TMRx_CCx 寄存器会捕获计数器当前的值，同时状态寄存器 TMRx_STS 的 CCxIFLG 位被置 1，如果 CCxIEN=1，便会产生中断。

捕获模式下可以测量一个波形的时序、频率、周期和占空比。在输入捕获模式中将边沿选择设定为上升沿检测，当捕获通道出现上升沿时，发生第一次捕获，此时计数器 CNT 的值会被锁存在捕获寄存器 CCx 中，同时会进入捕获中断，在中断服务程序中记录一次捕获，记下此时的值，当检测到下一个上升沿时，发生第二次捕获，计数器 CNT 的值会再次锁存在捕获寄存器 CCx 中，此时再次进入捕获中断，读取捕获寄存器的值，通过捕获就会得出此脉冲信号的周期。

15.4.4 输出比较

输出比较一共有八种模式：冻结、匹配时通道 x 为有效电平、匹配时通道 x 为无效电平、翻转、强制为无效、强制为有效、PWM 模式 1 和 PWM 模式 2，由 TMR_CCMx 寄存器中的 OCxMOD 位配置，在输出比较模式中可以控制输出信号的波形。

输出比较应用

输出比较模式中，定时器产生脉冲的位置、极性、频率和时间都是可以控制的。

当计数器的值和捕获/比较寄存器的值相等时，通过配置 TMRx_CCMx 寄存器中的 OCxMOD 位和输出极性 TMRx_CCEN 寄存器中的 CCxPOL 位，通道的输出可以被置高电平、低电平或者翻转。

在 TMRx_STS 寄存器中的 CCxIFLG=1 时，如果 TMRx_DIEN 寄存器中的 CCxIEN=1 产生中断；TMRx_CTRL2 寄存器中的 CCDSEL=1 产生 DMA 请求。

15.4.5 PWM 输出模式

PWM 模式是定时器对外输出可以调节的脉冲信号，其中信号的脉宽是由比较寄存器 CCx 的值决定，周期是由自动重载 AUTORLD 的值决定。

PWM 输出模式分为 PWM 模式 1 和 PWM 模式 2；PWM 模式 1 和 PWM 模式 2 分为向上计数和向下计数和边沿对齐计数；PWM 模式 1 中如果计数器 CNT 的值小于比较寄存器 CCx 的值，输出有效电平，否则反之。

设置 $CCx=5, AUTORLD=7$, 在 PWM 模式 1 下的时序图

图 49 PWM1 向上计数模式的时序图

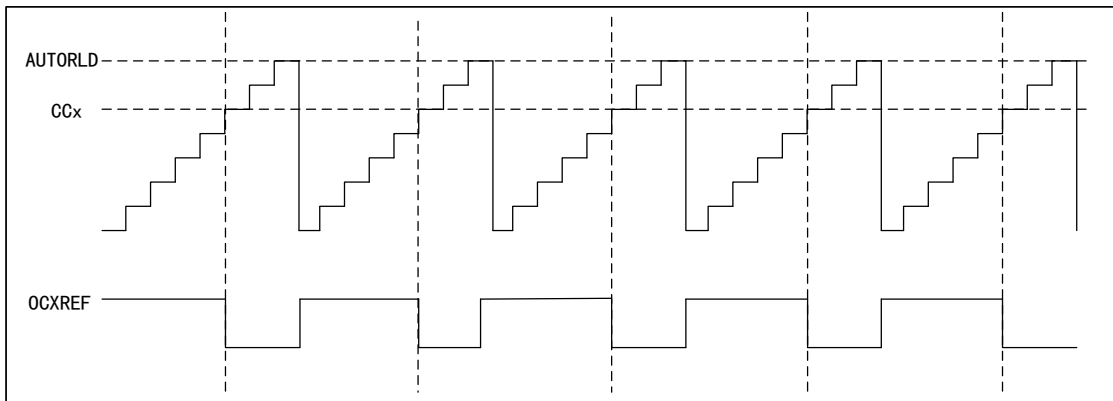


图 50 PWM1 向下计数模式的时序图

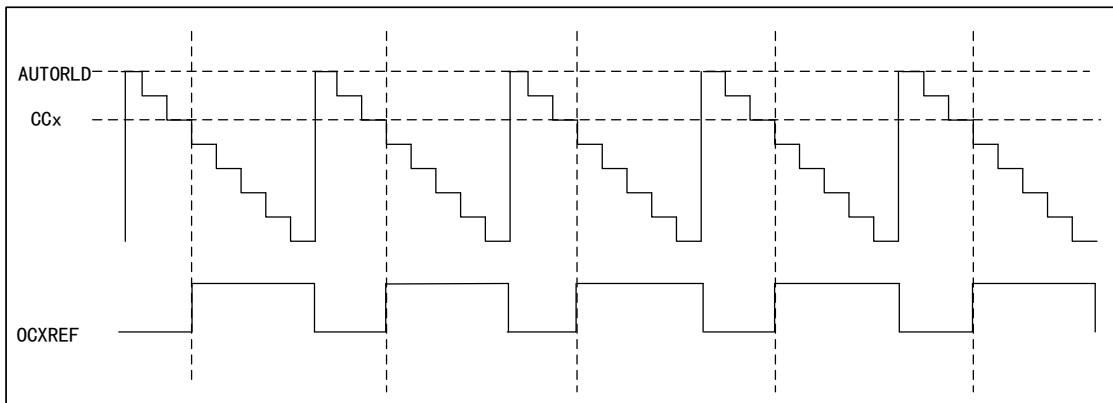
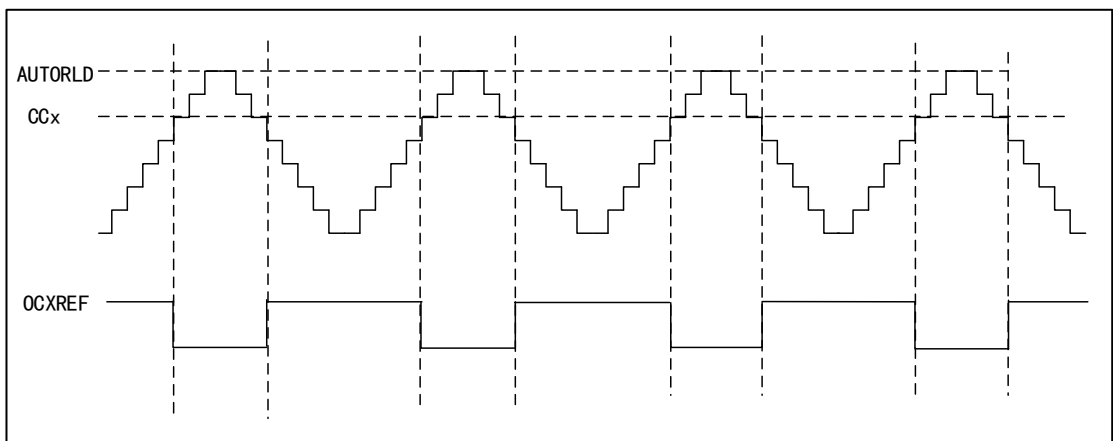


图 51 PWM1 中央对齐模式的时序图



PWM 模式 2 中如果计数器 CNT 的值小于比较寄存器 CCx 的值，输出无效电平，否则反之。

设置 $CCx=5, \text{AUTORLD}=7$, 在 PWM 模式 2 下的时序图

图 52 PWM2 向上计数模式的时序图

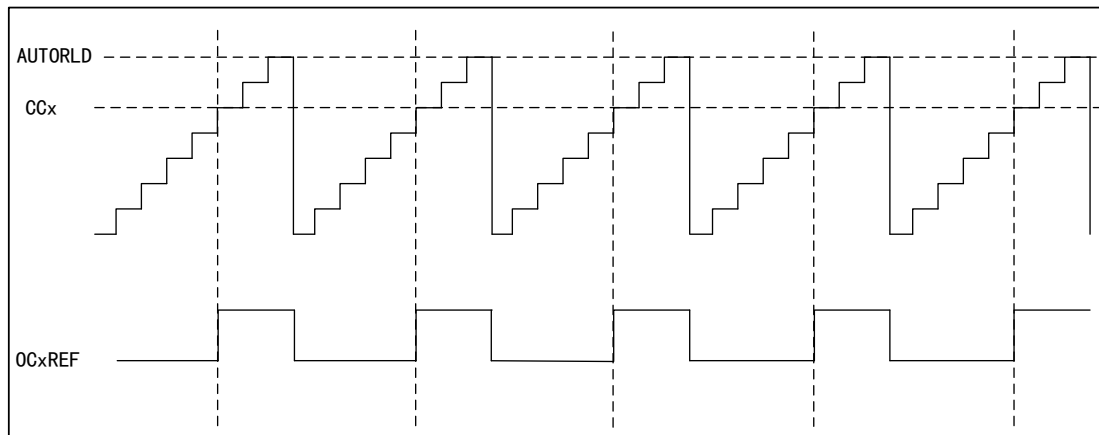


图 53 PWM2 向下计数模式的时序图

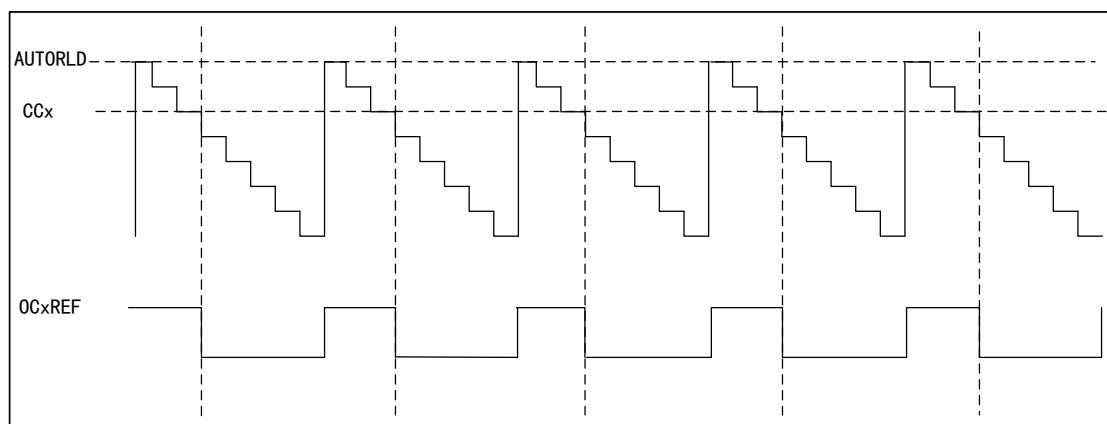
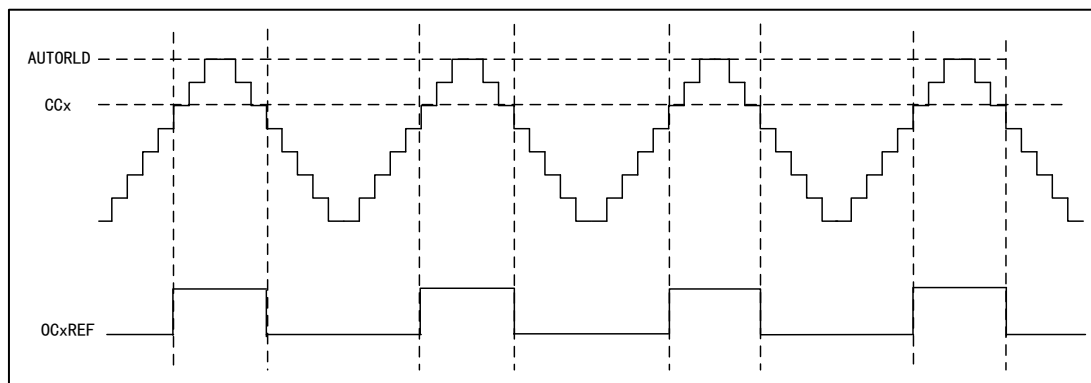


图 54 PWM2 中央对齐模式的时序图



15.4.6 PWM 输入模式

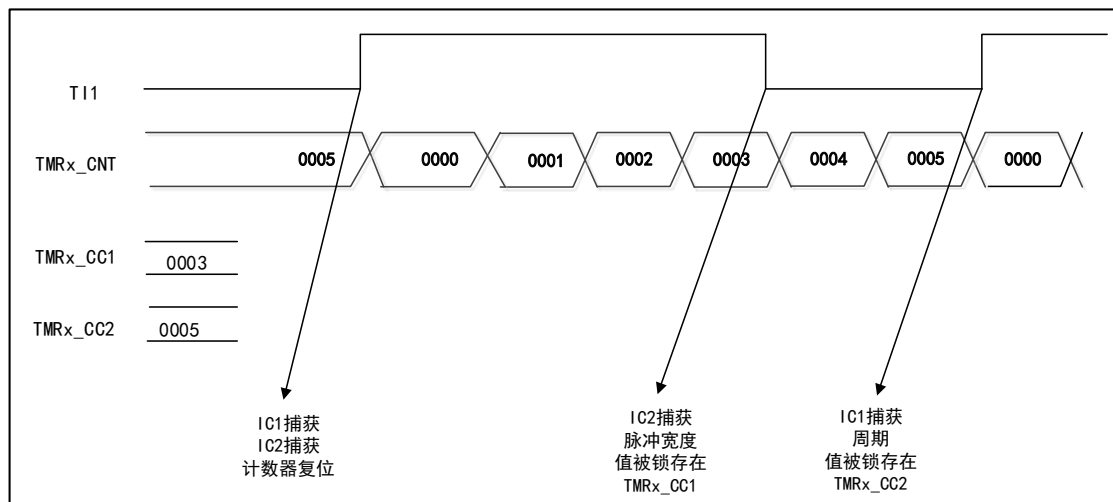
PWM 输入模式是输入捕获的一个特例。

PWM 输入模式，只有 TI1FP1、TI1FP2 连接到了从模式控制器，所以只能从通道 TMRx_CH1 和 TMRx_CH2 输入，且需要占用 CH1、CH2 的捕获寄存器。

在 PWM 输入模式中，PWM 信号从 TMRx_CH1 进入，信号会被分成两路，一路可以测量周期，一路可以测量占空比。在配置中只需设置其中一路的极性，另一路会自动配置相反的极性。

在此模式中，从模式控制器要配置成复位模式（TMRx_SMCTRL 寄存器的 SMFSEL 位）

图 55 PWM 输入模式时序图



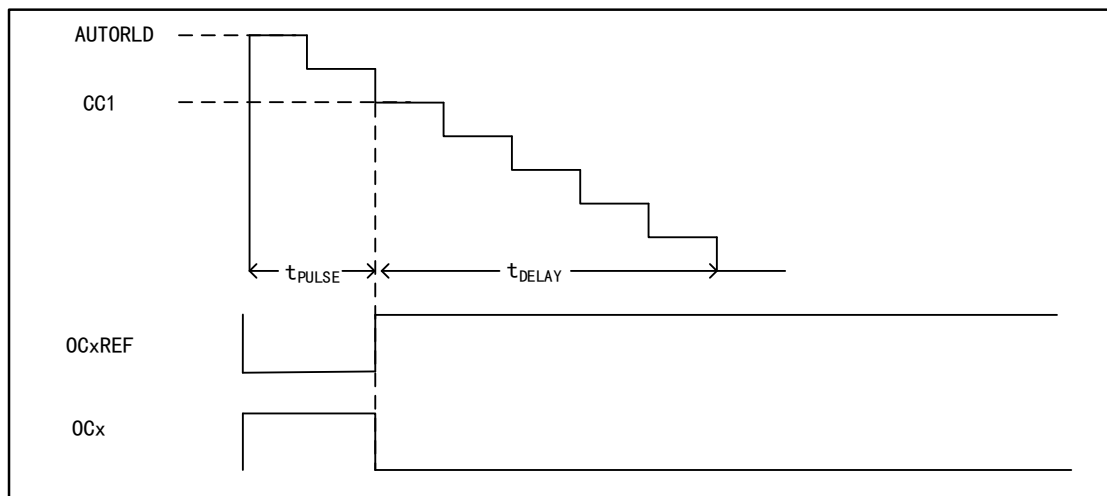
15.4.7 单脉冲模式

单脉冲模式是定时器比较输出中一种特殊情况，也是 PWM 输出模式的特例。

设置 TMRx_CTRL1 寄存器的 SPMEN 位选择单脉冲模式，计数器启动后，在未发生更新事件之前有一定个数的脉冲输出，当发生更新事件后计数器停止计数，后续不再有变化的 PWM 波形输出。

单脉冲模式通过程序在一定可控延迟后，产生一个脉宽可控的脉冲，延时时间由 TMRx_CCx 寄存器的值定义；在增计数模式下延时时间为 CCx，脉冲宽度为 AUTORLD-CCx；在减计数模式下延时时间为 AUTORLD-CCx，脉冲宽度为 CCx。

图 56 单脉冲模式下的时序图



15.4.8 强制输出模式

强制输出模式下无视比较结果，直接根据配置指令输出相应电平。

- TMRx_CCMx 寄存器的 CCxSEL=00, 设定 CCx 通道为输出
- TMRx_CCMx 寄存器的 OCxMOD=100/101, 设定强制 OCxREF 信号为无效/有效状态

在此模式中仍旧会产生相应的中断和 DMA 请求。

15.4.9 编码器接口模式

编码器接口模式相当于是一个带有方向选择的外部时钟，在编码器接口模式中，计数器的内容可一直指示编码器的位置。

选择编码器接口的方法如下：

- 通过设置 TMRx_SMCTRL 寄存器的 SMFSEL 位，可以设定计数器是在 TI1 通道/TI2 通道边沿计数，或者同时在 TI1 和 TI2 的边沿计数。
- 通过设置 TMRx_CCEN 寄存器中的 CC1POL 和 CC2POL 位，可以选择 TI1 和 TI2 的极性。
- 通过设置 TMRx_CCM1 寄存器中的 IC1F 和 IC2F 位，可以选择是否进行滤波。

两个输入 TI1 和 TI2 可以作为增量编码器的接口，计数器由在 TI1 和 TI2 经过滤波和边沿选择后的信号 TI1FP1 和 TI2FP2 的有效跳变驱动。

根据 TI1 和 TI2 的输入信号，产生计数脉冲和方向信号

- 根据输入信号的跳变顺序,计数器会向上/向下计数
- 将控制寄存器 TMRx_CTRL1 的 CNTDIR 设置为只读（任一输入端的跳变都会重新计算 CNTDIR）

计数器计数方向改变机制如下图

表格 50 计数方向与编码器的关系

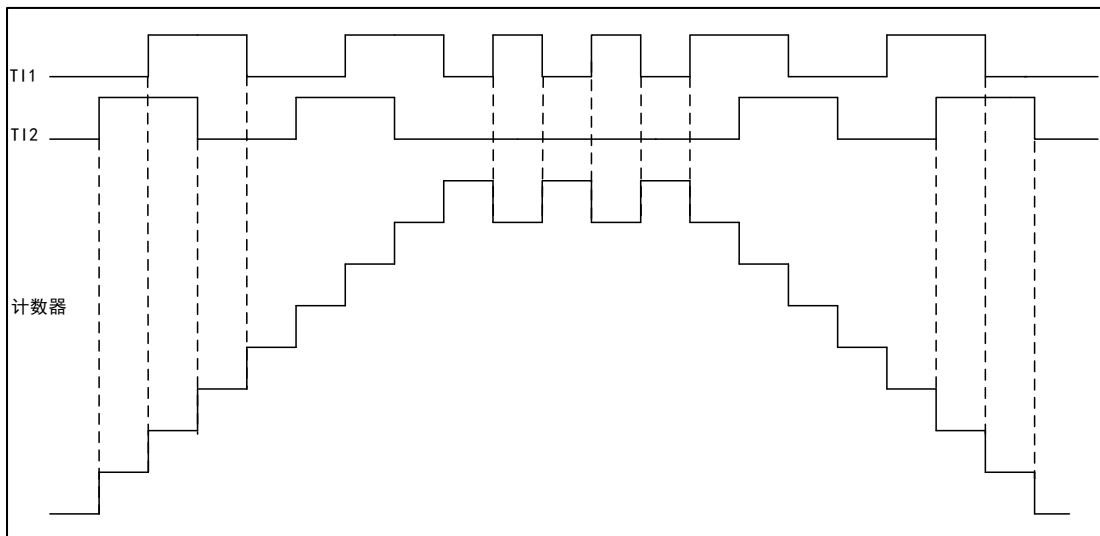
有效边沿		仅在 T11 计数		仅在 T12 计数		在 T11 和 T12 计数	
相对信号的电平		高	低	高	低	高	低
T11FP1	上升沿	—		向下计数	向上计数	向下计数	向上计数
	下降沿			向上计数	向下计数	向上计数	向下计数
T12FP2	上升沿	向上计数	向下计数	—		向上计数	向下计数
	下降沿	向下计数	向上计数			向下计数	向上计数

外部的增量编码器可以不用外部接口逻辑直接与 MCU 连接，所以使用比较器将编码器的差动输出转换到数字信号来增加抗噪声干扰。

在下图的实例中，

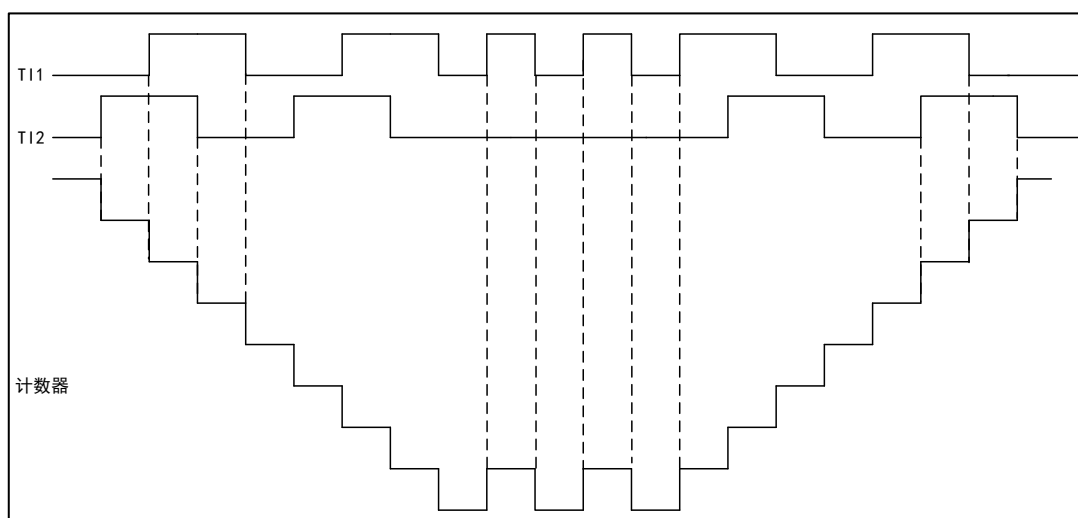
- 将 T11FP1 映射到 T11
- 将 T12FP2 映射到 T12
- T11FP1 和 T12FP2 都不反相
- 输入信号在上升沿和下降沿有效
- 使能计数器

图 57 编码器模式下的计数器操作实例



例如：当 T11 处在低电平时，如果 T12 出现上升沿状态，计数器向上计数。

图 58 TI1FP1 反相的编码器接口模式实例



例如：当 TI1 处于低电平时，TI2 发生上升沿跳变，计数器向下计数。

15.4.10 从模式

TMRx 定时器可以进行外部的触发同步

- 复位模式
- 门控模式
- 触发模式

可设置 TMRx_SMCTRL 寄存器中的 SMFSEL 位来选择是哪种模式

SMFSEL=100 设定复位模式，SMFSEL=101 设定门控模式，SMFSEL=110 设定触发模式。

复位模式下，在发生一个触发输入事件时，计数器和预分频器会被初始化，选中的触发输入（TRGI）的上升沿重新初始化计数器，并且产生一个更新寄存器的信号。

门控模式下，在置位 TMRx_CTRL1 寄存器中的 CNTEN 位后，计数器的使能依赖于选中的输入端的高电平，当触发输入为高时，计数器的时钟开启，一旦触发输入变为低，则计数器停止（但不复位），计数器的启动和停止都是受控制的。

触发模式下，计数器的使能依赖于选中的输入端上的事件，计数器在触发输入的上升沿启动（但不复位），只有计数器的启动是受控制的。

15.4.11 定时器互连

详情见 TMR1 定时器互连。

15.4.12 中断和 DMA 请求

定时器在工作时产生事件时会发生中断

- 更新事件（计数器上/下溢出，计数器初始化）
- 触发事件（计数器启动、停止、内/外部触发）

- 捕获/比较事件

其中一些内部中断事件可以产生 DMA 请求，还有专门的接口允许或者禁止触发 DMA 请求。

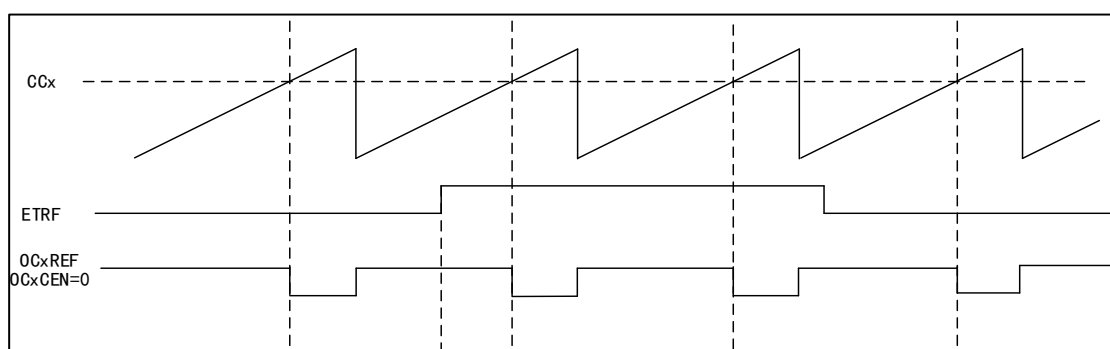
15.4.13 在外部事件时清除 OCxREF 信号

此功能是用用于输出比较和 PWM 模式。

在一个通道中，用 ETRF 输入端口的高电平将 OCxREF 的信号降为低电平，捕获/比较寄存器 TMRx_CCMx 中的 OCxCEN 的位置 1，OCxREF 信号会保持为低电平直到下一次发生更新事件。

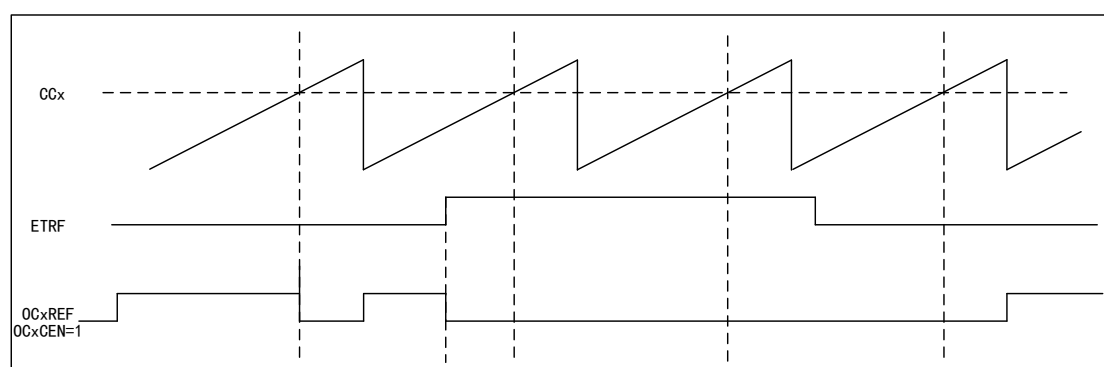
将 TMRx 置于 PWM 模式，关闭外部触发预分频器，禁止外部触发模式 2，当 ETRF 输入为高时，通过设置 OCxCEN=0，输出的 OCxREF 信号如下图。

图 59 OCxREF 时序图



将 TMRx 置于 PWM 模式，关闭外部触发预分频器，禁止外部触发模式 2，当 ETRF 输入为高时，通过设置 OCxCEN=1，输出的 OCxREF 信号如下图。

图 60 OCxREF 时序图



15.5 寄存器地址映射

下表中将 TMR2/3/4 所有寄存器映射到一个 16 位可寻址（编址）空间。

表格 51 TMR2/3/4 寄存器地址映射

寄存器名	描述	偏移地址
TMRx_CTRL1	控制寄存器 1	0x00
TMRx_CTRL2	控制寄存器 2	0x04
TMRx_SMCTRL	从模式控制寄存器	0x08
TMRx_DIEN	DMA/中断使能寄存器	0x0C
TMRx_STS	状态寄存器	0x10
TMRx_CEG	控制事件产生寄存器	0x14
TMRx_CCM1	捕获/比较模式寄存器 1	0x18
TMRx_CCM2	捕获/比较模式寄存器 2	0x1C
TMRx_CCEN	捕获/比较使能寄存器	0x20
TMRx_CNT	计数器寄存器	0x24
TMRx_PSC	预分频寄存器	0x28
TMRx_AUTORLD	自动重装寄存器	0x2C
TMRx_CC1	通道 1 捕获/比较寄存器	0x34
TMRx_CC2	通道 2 捕获/比较寄存器	0x38
TMRx_CC3	通道 3 捕获/比较寄存器	0x3C
TMRx_CC4	通道 4 捕获/比较寄存器	0x40
TMRx_DCTRL	DMA 控制寄存器	0x48
TMRx_DMADDR	连续模式的 DMA 地址寄存器	0x4C

15.6 寄存器功能描述

15.6.1 控制寄存器 1 (TMRx_CTRL1)

偏移地址: 0x00

复位值: 0x0000

位/域	名称	R/W	描述
0	CNTEN	R/W	使能计数器 (Counter Enable) 0: 禁止 1: 使能 定时器配置为外部时钟、门控模式和编码器模式时, 需要通过软件对该位写 1 启动定期工作; 配置为触发模式时, 可硬件写 1。
1	UD	R/W	禁止更新 (Update Disable) 更新事件可引起 AUTORLD、PSC、CCx 产生更新设置的数值。 0: 允许更新事件 (UEV) 更新事件可以由以下任一情况产生: 计数器溢出/下溢; 设置 UEG 位; 从模式控制器产生的更新。

位/域	名称	R/W	描述
			1: 禁止更新事件
2	URSSEL	R/W	更新请求源 (Update Request Source Select) 如果使能了中断或 DMA, 更新事件可产生更新中断或 DMA 请求, 通过该位可选择不同的更新请求源。 0: 计数器上溢或下溢 设置 UEG 位 通过从模式控制器产生的更新 1: 计数器上溢或下溢
3	SPMEN	R/W	使能单脉冲模式 (Single Pulse Mode Enable) 产生更新事件时, 可改变通道的输出电平; 在该模式下, 会清除 CNTEN 位, 停止计数器, 后续不再改变通道的输出电平。 0: 禁用 1: 使能
4	CNTDIR	R/W	配置计数器计数方向 (Counter Direction) 当计数器配置为中央对齐模式或编码器模式时, 该位为只读。 0: 向上计数 1: 向下计数
6:5	CAMSEL	R/W	选择中央对齐模式 (Center Aligned Mode Select) 中央对齐模式下, 计数器交替的向上向下计数; 否则只向上或向下计数。 不同的中央对齐模式, 影响输出输出通道的输出比较中断标志位置 1 的时机; 在计数器禁止时 (CNTEN=0) 时, 选择中央对齐模式。 00: 边沿对齐模式 01: 中心对齐模式 1 (在向下计数时, 输出通道的输出比较中断标志位置 1) 10: 中心对齐模式 2 (在向上计数时, 输出通道的输出比较中断标志位置 1) 11: 中心对齐模式 3 (在向上/下计数时, 输出通道的输出比较中断标志位置 1)
7	ARPEN	R/W	TMRx_AUTORLD 寄存器自动重装缓冲使能 (Auto-reload Preload Enable) 禁止缓存区时, 程序修改 TMRx_AUTORLD 会立刻修改装入计数器的数值; 使能缓存区时, 程序修改 TMRx_AUTORLD 会在下一个更新事件修改装入计数器的数值。 0: 禁止 1: 使能
9:8	CLKDIV	R/W	时钟分频系数 (Clock Division) 死区、数字滤波器的配置由 CK_INT 提供时钟, 通过设置改位可调整死区时间、数字滤波器的采用时钟。 00: $t_{DTS}=t_{CK_INT}$ 01: $t_{DTS}=2 \times t_{CK_INT}$ 10: $t_{DTS}=4 \times t_{CK_INT}$ 11: 保留
15:10			保留

15.6.2 控制寄存器 2 (TMRx_CTRL2)

偏移地址: 0x04

复位值: 0x0000

位/域	名称	R/W	描述
0	保留		
1	MMSEL_3	R/W	MMSEL 的补充信号 (Supplementary signal of MMSEL) 用来扩展定时器主模式下用于 TRGO 的信号。 0: TRGO 的信号由 MMSEL<2:0>决定 1: TRGO 的信号为主模式定时器的编码器时钟信号, 也即当计数器跳变时, TRGO 输出一个周期的高脉冲。建议在编码器接口模式下使用。
2	保留		
3	CCDSEL	R/W	选择发出捕获/比较的 DMA 请求 (Capture/Compare DMA Select) 0: 当发生 CCx 事件时, 送出 CCx 的 DMA 请求 1: 当发生更新事件时, 送出 CCx 的 DMA 请求
6:4	MMSEL	R/W	选择定时器主模式下用于 TRGO 的信号 (Master Mode Signal Select) 工作在主模式的定时器的信号可用于 TRGO, 从而影响处在从模式且与主定时器级联的的定时器工作, 具体影响可从模式的定时器配置有关。 000: 复位, 主模式定时器的复位信号用于 TRGO 001: 使能, 主模式定时器的计数器使能信号用于 TRGO 010: 更新, 主模式定时器的更新事件用于 TRGO 011: 比较脉冲, 主模式定时器捕获/比较成功 (CCxIFLG=1) 时输出一个脉冲信号用于 TRGO 100: 比较模式 1, OC1REF 用于触发 TRGO 101: 比较模式 2, OC2REF 用于触发 TRGO 110: 比较模式 3, OC3REF 用于触发 TRGO 111: 比较模式 4, OC4REF 用于触发 TRGO
7	TI1SEL	R/W	TI1 选择 (Timer Input 1 Select) 0: TMRx_CH1 引脚连到 TI1 输入 1: TMRx_CH1、TMRx_CH2 和 TMRx_CH3 引脚经异或后连到 TI1 输入
10:8	MM2SEL	R/W	选择定时器主模式下用于 TRGO2 的信号 (Master Mode 2 Signal Select) 工作在主模式的定时器的信号可用于 TRGO2, 从而影响处在从模式且与主定时器级联的的定时器工作, 具体影响可从模式的定时器配置有关。 000: 复位, 主模式定时器的复位信号用于 TRGO2 001: 使能, 主模式定时器的计数器使能信号用于 TRGO2 010: 更新, 主模式定时器的更新事件用于 TRGO2 011: 比较脉冲, 主模式定时器捕获/比较成功 (CCxIFLG=1) 时输出一个脉冲信号用于 TRGO2 100: 比较模式 1, OC1REF 用于触发 TRGO2 101: 比较模式 2, OC2REF 用于触发 TRGO2 110: 比较模式 3, OC3REF 用于触发 TRGO2 111: 比较模式 4, OC4REF 用于触发 TRGO2
15:11	保留		

15.6.3 从模式控制寄存器 (TMRx_SMCTRL)

偏移地址: 0x08

复位值: 0x0000

位/域	名称	R/W	描述
2:0	SMFSEL	R/W	<p>选择从模式功能 (Slave Mode Function Select)</p> <p>000: 禁止从模式, 定时器可作为主模式定时器影响从模式定时器的工 作; 如果 CTRL1_CNTEN=1, 则预分频器直接由内部时钟驱动。</p> <p>001: 编码器模式 1, 根据 TI1FP1 的电平, 计数器在 TI2FP2 的边沿计 数。</p> <p>010: 编码器模式 2, 根据 TI2FP2 的电平, 计数器在 TI1FP1 的边沿计 数。</p> <p>011: 编码器模式 3, 根据另一个信号的输入电平, 计数器在 TI1FP1、 TI2FP2 的边沿计数。</p> <p>100: 复位模式, 从模式定时器在收到 TRGI 的上升沿信号后复位计数 器, 并产生更新寄存器的信号。</p> <p>101: 门控模式, 从模式定时器在收到 TRGI 高电平信号时, 启动计数 器工作; 收到 TRGI 低电平时停止计数器工作; 再收到 TRGI 高电平信 号时, 继续工作; 整个期间不复位计数器。</p> <p>110: 触发模式, 从模式定时器在收到 TRGI 的上升沿信号后, 启动计数 器工作。</p> <p>111: 外部时钟模式 1, 选择 TRGI 的上升沿信号作为时钟源驱动计数器 工作。</p>
3	OCCSEL	R/W	<p>选择 OCREF 信号清除源 (OCREF Clear Source Select)</p> <p>该位用来选择 OCREF 清除源</p> <p>0: OCREF_CLR</p> <p>1: ETRF</p>
6:4	TRGSEL	R/W	<p>选择触发输入信号 (Trigger Input Signal Select)</p> <p>为了避免在改变该位值时产生错误的边沿检测, 须在 SMFSEL=0 时改 变。</p> <p>000: 内部触发 ITR0</p> <p>001: 内部触发 ITR1</p> <p>010: 内部触发 ITR2</p> <p>011: 内部触发 ITR3</p> <p>100: 通道 1 输入边沿检测器 TIF_ED</p> <p>101: 通道 1 滤波后定时器输入 TI1FP1</p> <p>110: 通道 2 滤波后的定时器输入 TI2FP2</p> <p>111: 外部触发输入 (ETRF)</p>
7	MSMEN	R/W	<p>使能主/从模式 (Master/slave Mode Enable)</p> <p>0: 无效</p> <p>1: 使能主/从模式</p>
11:8	ETFCFG	R/W	<p>配置外部触发滤波器 (External Trigger Filter Configure)</p> <p>0000: 禁用滤波器, 以 f_{DTs} 采样</p> <p>0001: DIV=1, N=2</p> <p>0010: DIV=1, N=4</p> <p>0011: DIV=1, N=8</p> <p>0100: DIV=2, N=6</p> <p>0101: DIV=2, N=8</p> <p>0110: DIV=4, N=6</p> <p>0111: DIV=4, N=8</p>

位/域	名称	R/W	描述
			1000: DIV=8, N=6 1001: DIV=8, N=8 1010: DIV=16, N=5 1011: DIV=16, N=6 1100: DIV=16, N=8 1101: DIV=32, N=5 1110: DIV=32, N=6 1111: DIV=32, N=8 采样频率=定时器时钟频率/DIV; 滤波长度=N, 每 N 个事件产生一个跳变。
13:12	ETPCFG	R/W	配置外部触发信号预分频器 (External Trigger Prescaler Configure) ETR (外部触发输入) 的信号经过分频后为 ETRP, ETRP 的信号频率最多是 TMRxCLK 频率的 1/4; 当 ETR 频率过高时, 须经过分频降低 ETRP 的频率。 00: 禁用预分频器; 01: ETR 信号 2 分频 10: ETR 信号 4 分频 11: ETR 信号 8 分频
14	ECEN	R/W	使能外部时钟模式 2 (External Clock Enable Mode2) 0: 禁止 1: 使能 设置 ECEN 位与选择外部时钟模式 1 将 TRGI 连接到 ETRF 具有相同作用; 从模式 (复位、门控、触发) 可以与外部时钟模式 2 同时使用, 但此时 TRGI 不能连到 ETRF; 当外部时钟模式 1 和外部时钟模式 2 同时使能时, 外部时钟的输入是 ETRF。
15	ETPOL	R/W	配置外部触发极性 (External Trigger Polarity Configure) 该位决定外部触发 ETR 是否反相。 0: 外部触发 ETR 不反相, 高电平或上升沿有效 1: 外部触发 ETR 反相, 低电平或下降沿有效

表格 52 TMRx 内部触发连接

从定时器	ITR0 (TS=000)	ITR1 (TS=001)	ITR2 (TS=010)	ITR3 (TS=011)
TMR2	TMR1	TMR4	TMR3	-
TMR3	TMR1	TMR2	TMR4	-
TMR4	TMR1	TMR2	TMR3	-

15.6.4 DMA/中断使能寄存器 (TMRx_DIEN)

偏移地址: 0x0C

复位值: 0x0000

位/域	名称	R/W	描述
0	UIEN	R/W	使能更新中断 (Update Interrupt Enable) 0: 禁止 1: 使能
1	CC1IEN	R/W	使能捕获/比较通道 1 中断 (Capture/Compare Channel1 Interrupt Enable) 0: 禁止

位/域	名称	R/W	描述
			1: 使能
2	CC2IEN	R/W	使能捕获/比较通道 2 中断 (Capture/Compare Channel2 Interrupt Enable) 0: 禁止 1: 使能
3	CC3IEN	R/W	使能捕获/比较通道 3 中断 (Capture/Compare Channel3 Interrupt Enable) 0: 禁止 1: 使能
4	CC4IEN	R/W	使能捕获/比较通道 4 中断 (Capture/Compare Channel4 Interrupt Enable) 0: 禁止 1: 使能
5	保留		
6	TRGIEN	R/W	使能触发中断 (Trigger Interrupt Enable) 0: 禁止 1: 使能
7	保留		
8	UDIEN	R/W	使能更新的 DMA 请求 (Update DMA Request Enable) 0: 禁止 1: 使能
9	CC1DEN	R/W	使能捕获/比较通道 1 的 DMA 请求 (Capture/Compare Channel1 DMA Request Enable) 0: 禁止 1: 使能
10	CC2DEN	R/W	使能捕获/比较通道 2 的 DMA 请求 (Capture/Compare Channel2 DMA Request Enable) 0: 禁止 1: 使能
11	CC3DEN	R/W	使能捕获/比较通道 3 的 DMA 请求 (Capture/Compare Channel3 DMA Request Enable) 0: 禁止 1: 使能
12	CC4DEN	R/W	使能捕获/比较通道 4 的 DMA 请求 (Capture/Compare Channel4 DMA Request Enable) 0: 禁止 1: 使能
13	保留		
14	TRGDEN	R/W	使能触发 DMA 请求 (Trigger DMA Request Enable) 0: 禁止 1: 使能
15	保留		

15.6.5 状态寄存器 (TMRx_STS)

偏移地址: 0x10

复位值：0x0000

位/域	名称	R/W	描述
0	UIFLG	RC_W0	<p>产生更新事件中断标志位（Update Event Interrupt Generate Flag）</p> <p>0：没有发生更新事件中断</p> <p>1：发生更新事件中断</p> <p>计数器数值重新装载或重新初始化时，会产生更新事件，该位由硬件置 1，软件清 0；更新事件的产生的情况有以下情况：</p> <p>（1）TMRx_CTRL1 寄存器的 UD=0，重复计数器数值上/下溢时产生更新事件；</p> <p>（2）TMRx_CTRL1 寄存器的 URSSEL=0 和 UD=0，配置 TMRx_CEG 寄存器的 UEG=1 产生更新事件，需要通过软件初始化计数器；</p> <p>（3）TMRx_CTRL1 寄存器的 URSSEL=0 和 UD=0，计数器被触事件初始化时产生更新事件。</p>
1	CC1IFLG	RC_W0	<p>捕获/比较通道 1 中断标志（Captuer/Compare Channel1 Interrupt Flag）</p> <p>当捕获比较通道 1 配置为输出时：</p> <p>0：无匹配发生</p> <p>1：TMRx_CNT 的值与 TMRx_CC1 的值相匹配</p> <p>当捕获比较通道 1 配置为输入时：</p> <p>0：没有发生输入捕获</p> <p>1：发生输入捕获</p> <p>捕获事件发生时由硬件置 1，可以由软件清 0 或者读 TMRx_CC1 寄存器时清 0。</p>
2	CC2IFLG	RC_W0	<p>捕获/比较通道 2 中断标志（Captuer/Compare Channel2 Interrupt Flag）</p> <p>参考 STS_CC1IFLG</p>
3	CC3IFLG	RC_W0	<p>捕获/比较通道 3 中断标志（Captuer/Compare Channel3 Interrupt Flag）</p> <p>参考 STS_CC1IFLG</p>
4	CC4IFLG	RC_W0	<p>捕获/比较通道 4 中断标志（Captuer/Compare Channel4 Interrupt Flag）</p> <p>参考 STS_CC1IFLG</p>
5	保留		
6	TRGIFLG	RC_W0	<p>产生触发事件中断标志（Trigger Event Interrupt Generate Flag）</p> <p>0：没有发生触发事件中断</p> <p>1：发生触发事件中断</p> <p>发生触发事件时，该位由硬件置 1，软件清 0。</p>
8:7	保留		
9	CC1RCFLG	RC_W0	<p>捕获/比较通道 1 重复捕获标志（Captuer/Compare Channel1 Repetition Capture Flag）</p> <p>0：没有发生重复捕获</p> <p>1：发生重复捕获</p> <p>计数器的值被捕获到 TMRx_CCR1 寄存器中，此时 CC1IFLG=1；只有当通道被配置为输入捕获时，该位由硬件置 1，软件清 0。</p>

位/域	名称	R/W	描述
10	CC2RCFLG	RC_W0	捕获/比较通道 2 重复捕获标志 (Captuer/compare Channel2 Repetition Capture Flag) 参考 STS_CC1RCFLG
11	CC3RCFLG	RC_W0	捕获/比较通道 3 重复捕获标志 (Captuer/compare Channel3 Repetition Capture Flag) 参考 STS_CC1RCFLG
12	CC4RCFLG	RC_W0	捕获/比较通道 4 重复捕获标志 (Captuer/compare Channel4 Repetition Capture Flag) 参考 STS_CC1RCFLG
15:13	保留		

15.6.6 控制事件产生寄存器 (TMRx_CEG)

偏移地址: 0x14

复位值: 0x0000

位/域	名称	R/W	描述
0	UEG	W	产生更新事件 (Update Event Generate) 0: 无效 1: 初始化计数器, 产生更新事件 此位由软件置 1, 硬件清 0。 注意: 产生更新事件时, 预分频器的计数器会清 0, 但是预分频系数不变。 如果在向下计数模式下计数器会读取 TMRx_AUTORLD 的值; 如果在中央对齐模式下或者向上计数模式中计数器会被清 0。
1	CC1EG	W	产生捕获/比较通道 1 事件 (Capture/Compare Channel1 Event Generation) 0: 无效 1: 产生捕获/比较事件 该位由软件置 1, 硬件自动清 0。 如果通道 1 处于输出模式: 当 CC1IFLG=1 时, 如果设置了 CC1IEN 和 CC1DEN 位, 则产生相应的中断和 DMA 请求。 如果通道 1 处于输入模式: 捕获计数器的值存储在 TMRx_CC1 寄存器中; 配置 CC1IFLG=1, 如果还设置了 CC1IEN 和 CC1DEN 位, 则产生相应的中断和 DMA 请求; 如果此时 CC1IFLG=1, 则需要配置 CC1RCFLG=1。
2	CC2EG	W	产生捕获/比较通道 2 事件 (Capture/Compare Channel2 Event Generation) 参考 CC1EG 描述
3	CC3EG	W	产生捕获/比较通道 3 事件 (Capture/Compare Channel3 Event Generation) 参考 CC1EG 描述
4	CC4EG	W	产生捕获/比较通道 4 事件 (Capture/Compare Channel4 Event Generation) 参考 CC1EG 描述
5	保留		
6	TEG	W	产生触发事件 (Trigger Event Generate) 0: 无效

位/域	名称	R/W	描述
			1: 产生触发事件 该位由软件置 1，硬件自动清 0。
15:8			保留

15.6.7 捕获/比较模式寄存器 1 (TMRx_CCM1)

偏移地址: 0x18

复位值: 0x0000

可通过 CCxSEL 位配置定时器为输入 (捕获模式) 或输出 (比较模式)。该寄存器其它位的作用在输入和输出模式下不同, 同一个位在输出模式和输入模式下的功能是不同的。寄存器中的 OCxx 描述了通道在输出模式下的功能, 寄存器中的 ICxx 描述了通道在输入模式下的功能。

输出比较模式:

位/域	名称	R/W	描述
1:0	CC1SEL	R/W	选择捕获/比较通道 1 (Capture/Compare Channel1 Select) 该位定义了输入/输出的方向以及选择输入引脚。 00: CC1 通道为输出 01: CC1 通道为输入, IC1 映射在 TI1 上 10: CC1 通道为输入, IC1 映射在 TI2 上 11: CC1 通道为输入, IC1 映射在 TRC 上, 仅工作在内部触发输入 注: 该位仅在通道关闭时 (TMRx_CCEN 寄存器的 CC1EN=0 时) 可写。
2	OC1FEN	R/W	快速使能输出比较通道 1 (Output Compare Channel1 Fast Enable) 0: 禁止 1: 使能 该位用来提高捕获/比较输出对触发输入事件的响应。
3	OC1PEN	R/W	使能输出比较通道 1 预装载 (Output Compare Channel1 Preload Enable) 0: 禁止预装载功能, 通过程序写入 TMRx_CC1 寄存器的数值, 会马上起作用。 1: 启用预装载功能, 通过程序写入 TMRx_CC1 寄存器的数值, 会在产生更新事件后起作用。 注: 当保护级别为 3 级时且通道配置为输出时, 该位不能被修改。当不确定预装载寄存器情况, 仅在单脉冲模式 (SPMEN=1) 下, 可以使用 PWM 模式, 否则不确定其接下来的输出比较结果。
6:4	OC1MOD	R/W	配置输出比较通道 1 模式 (Output Compare Channel1 Mode Configure) 000: 冻结。输出比较对 OC1REF 无影响 001: 匹配时输出置为高。计数器的值和捕获比较寄存器的值 CCx 发生匹配时, 强制 OC1REF 为高电平 010: 匹配时输出置为低。计数器的值和捕获比较寄存器的值发生匹配时, 强制 OC1REF 为低电平 011: 匹配时输出翻转。计数器的值和捕获比较寄存器的值发生匹配时, 翻转 OC1REF 的电平 100: 强制输出为低。强制 OC1REF 为低电平 101: 强制输出为高。强制 OC1REF 为高电平 110: PWM 模式 1 (计数器值<输出比较值时置为高, 否则反之) 111: PWM 模式 2 (计数器值>输出比较值时置为高, 否则反之)

位/域	名称	R/W	描述
			注：当保护级别为 3 级时且通道配置为输出时，该位不能被修改。在 PWM 模式 1 和 2 中，OC1REF 电平在比较结果改变或者输出比较模式从冻结模式转换到 PWM 模式时改变。
7	OC1CEN	R/W	使能输出比较通道 1 清除（Output Compare Channel1 Clear Enable） 0: OC1REF 不受 ETRF 输入影响。 1: 检测到 ETRF 输入高电平时，OC1REF=0
9:8	CC2SEL	R/W	通道 2 模式选择（Capture/Compare Channel2 Select） 该位定义了输入/输出的方向以及选择输入引脚。 00: CC2 通道为输出 01: CC2 通道为输入，IC2 映射在 TI2 上 10: CC2 通道为输入，IC2 映射在 TI1 上 11: CC2 通道为输入，IC2 映射在 TRC 上，仅工作在内部触发输入 注意：该位仅在通道关闭时（TMRx_CCEN 寄存器的 CC2EN=0 时）可写。
10	OC2FEN	R/W	快速使能输出比较通道 2（Output Compare Channel2 Preload Enable）
11	OC2PEN	R/W	使能输出比较通道 2 缓冲（Output Compare Channel2 Buffer Enable）
14:12	OC2MOD	R/W	输出比较通道 2 模式（Output Compare Channel1 Mode）
15	OC2CEN	R/W	使能输出通道 2 比较清除（Output Compare Channel2 Clear Enable）

输入捕获模式:

位/域	名称	R/W	描述
1:0	CC1SEL	R/W	选择输入/捕获通道 1（Capture/Compare Channel1 Select） 00: CC1 通道为输出 01: CC1 通道为输入，IC1 映射在 TI1 上 10: CC1 通道为输入，IC1 映射在 TI2 上 11: CC1 通道为输入，IC1 映射在 TRC 上，仅工作在内部触发输入 注意：该位仅在通道关闭时（TMRx_CCEN 位的 CC1EN=0 时）可写。
3:2	IC1PSC	R/W	配置输入捕获通道 1 预分频因子（Input Capture Channel1 Prescaler Configure） 00: PSC=1 01: PSC=2 10: PSC=4 11: PSC=8 PSC 是预分频因子，每 PSC 个事件触发一次捕获。
7:4	IC1F	R/W	配置输入捕获通道 1 滤波器（Input Capture Channel1 Filter Configure） 0000: 禁用滤波器，以 f_{DTS} 采样 0001: DIV=1, N=2 0010: DIV=1, N=4 0011: DIV=1, N=8 0100: DIV=2, N=6 0101: DIV=2, N=8 0110: DIV=4, N=6 0111: DIV=4, N=8 1000: DIV=8, N=6 1001: DIV=8, N=8 1010: DIV=16, N=5

位/域	名称	R/W	描述
			1011: DIV=16, N=6 1100: DIV=16, N=8 1101: DIV=32, N=5 1110: DIV=32, N=6 1111: DIV=32, N=8 采样频率=定时器时钟频率/DIV; 滤波器长度=N, 表示每 N 个事件产生一个跳变。
9:8	CC2SEL	R/W	选择捕获/比较通道 2 (Capture/Compare Channel2 Select) 00: CC2 通道为输出 01: CC2 通道为输入, IC2 映射在 TI1 上 10: CC2 通道为输入, IC2 映射在 TI2 上 11: CC2 通道为输入, IC2 映射在 TRC 上, 仅工作在内部触发输入 注意: 该位仅在通道关闭时 (TMRx_CCEN 寄存器的 CC2EN=0 时) 可写。
11:10	IC2PSC	R/W	配置输入捕获通道 2 预分频因子 (Input Capture Channel2 Prescaler Configure)
15:12	IC2F	R/W	配置输入捕获通道 2 滤波器 (Input Capture Channel2 Filter Configure)

15.6.8 捕获/比较模式寄存器 2 (TMRx_CCM2)

偏移地址: 0x1C

复位值: 0x0000

参看以上 CCM1 寄存器的描述。

输出比较模式:

位/域	名称	R/W	描述
1:0	CC3SEL	R/W	选择捕获/比较通道 3 (Capture/Compare Channel1 Select) 该位定义了输入/输出的方向以及选择输入引脚。 00: CC3 通道为输出 01: CC3 通道为输入, IC3 映射在 TI3 上 10: CC3 通道为输入, IC3 映射在 TI4 上 11: CC3 通道为输入, IC3 映射在 TRC 上, 仅工作在内部触发输入 注意: 该位仅在通道关闭时 (TMRx_CCEN 寄存器的 CC3EN=0 时) 可写。
2	OC3FEN	R/W	快速使能输出比较通道 3 (Output Compare Channel3 Fast Enable) 0: 禁止 1: 使能 该位用来提高捕获/比较输出对触发输入事件的响应。
3	OC3PEN	R/W	使能输出比较通道 3 预装载 (Output Compare Channel3 Preload Enable)
6:4	OC3MOD	R/W	配置输出比较通道 3 模式 (Output Compare Channel3 Mode Configure)
7	OC3CEN	R/W	使能输出比较通道 3 清除 (Output Compare Channel3 Clear Enable) 0: OC3REF 不受 ETRF 输入影响。 1: 检测到 ETRF 输入高电平时, OC1REF=0

位/域	名称	R/W	描述
9:8	CC4SEL	R/W	选择通道 4 模式 (Capture/compare Channel4 Select) 该位定义了输入/输出的方向以及选择输入引脚。 00: CC4 通道为输出 01: CC4 通道为输入, IC4 映射在 TI4 上 10: CC4 通道为输入, IC4 映射在 TI3 上 11: CC4 通道为输入, IC4 映射在 TRC 上, 仅工作在内部触发输入 注意: 该位仅在通道关闭时 (TMRx_CCEN 寄存器的 CC4EN=0 时) 可写。
10	OC4FEN	R/W	快速使能输出比较通道 4 (Output Compare Channel4 Preload Enable)
11	OC4PEN	R/W	使能输出比较通道 4 缓冲 (Output Compare Channel4 Buffer Enable)
14:12	OC4MOD	R/W	配置输出比较通道 4 模式 (Output Compare Channel4 Mode Configure)
15	OC4CEN	R/W	使能输出通道 4 比较清除 (Output Compare Channel4 Clear Enable)

输入捕获模式:

位/域	名称	R/W	描述
1:0	CC3SEL	R/W	选择输入/捕获通道 3 (Capture/Compare Channel3 Select) 00: CC3 通道为输出 01: CC3 通道为输入, IC3 映射在 TI3 上 10: CC3 通道为输入, IC3 映射在 TI4 上 11: CC3 通道为输入, IC3 映射在 TRC 上, 仅工作在内部触发输入 注意: 该位仅在通道关闭时 (TMRx_CCEN 寄存器的 CC3EN=0 时) 可写。
3:2	IC3PSC	R/W	配置输入捕获通道 3 预分频因子 (Input Capture Channel3 Prescaler Configure) 00: PSC=1 01: PSC=2 10: PSC=4 11: PSC=8 PSC 是预分频因子, 每 PSC 个事件触发一次捕获。
7:4	IC3F	R/W	配置输入捕获通道 3 滤波器 (Input Capture Channel3 Filter Configure)
9:8	CC4SEL	R/W	选择捕获/比较通道 4 (Capture/Compare Channel4 Select) 00: CC4 通道为输出 01: CC4 通道为输入, IC4 映射在 TI4 上 10: CC4 通道为输入, IC4 映射在 TI3 上 11: CC4 通道为输入, IC4 映射在 TRC 上, 仅工作在内部触发输入 注意: 该位仅在通道关闭时 (TMRx_CCEN 寄存器的 CC4EN=0 时) 可写。
11:10	IC4PSC	R/W	配置输入捕获通道 4 预分频因子 (Input Capture Channel4 Prescaler Configure)
15:12	IC4F	R/W	配置输入捕获通道 4 滤波器 (Input Capture Channel4 Filter Configure)

15.6.9 捕获/比较使能寄存器 (TMRx_CCEN)

偏移地址: 0x20

复位值: 0x0000

位/域	名称	R/W	描述
0	CC1EN	R/W	<p>使能捕获/比较通道 1 输出 (Capture/Compare Channel1 Output Enable)</p> <p>CC1 配置为输出时:</p> <p>0: 禁止输出</p> <p>1: 开启输出</p> <p>CC1 配置为输入时:</p> <p>该位决定了计数器的值 CNT 是否能捕获进入 TMRx_CC1 寄存器中</p> <p>0: 禁止捕获</p> <p>1: 开启捕获</p>
1	CC1POL	R/W	<p>配置捕获/比较通道 1 输出极性 (Capture/Compare Channel1 Output Polarity Configure)</p> <p>CC1 通道配置为输出时:</p> <p>0: OC1 高电平有效</p> <p>1: OC1 低电平有效</p> <p>CC1 通道配置为输入时:</p> <p>CC1POL 和 CC1NPOL 同时控制触发或捕获的信号 TI1FP1 和 TI2FP1 的极性</p> <p>00: 不反相/上升沿:</p> <p>TIxFP1 不反相 (门控、编码器模式下触发), 在 TIxFP1 的上升沿捕获 (复位触发、捕获、外部时钟和触发模式)。</p> <p>01: 反相/下降沿:</p> <p>TIxFP1 反相 (门控、编码器模式下触发), 在 TIxFP1 的上升沿捕获 (复位触发、捕获、外部时钟和触发模式)。</p> <p>10: 保留</p> <p>11: 不反相/上升和下降沿:</p> <p>TIxFP1 不反相 (门控模式下触发, 不能用于编码器模式), 在 TIxFP1 的上升沿捕获 (复位触发、捕获、外部时钟和触发模式)。</p>
2	保留		
3	CC1NPOL	R/W	<p>配置捕获/比较通道 1 输出极性 Capture/Compare Channel1 Output Polarity Configure)</p> <p>CC1 通道配置为输出时:</p> <p>CC1NPOL 一直保持清除状态</p> <p>CC1 通道配置为输入时:</p> <p>该位和 CC1POL 同时控制触发或捕获的信号 TI1FP1 和 TI2FP1 的极性。</p>
4	CC2EN	R/W	<p>使能捕获/比较通道 2 输出 (Capture/Compare Channel2 Output Enable)</p> <p>参考 CCEN_CC1EN</p>
5	CC2POL	R/W	<p>配置捕获/比较通道 2 输出极性 (Capture/Compare Channel2 Output Polarity Configure)</p> <p>参考 CCEN_CC1POL</p>
6	保留		
7	CC2NPOL	R/W	<p>配置捕获/比较通道 2 输出极性 Capture/Compare Channel2 Output Polarity Configure)</p> <p>参考 CCEN_CC1NPOL</p>
8	CC3EN	R/W	<p>使能捕获/比较通道 3 输出 (Capture/Compare Channel3 Output Enable)</p>

位/域	名称	R/W	描述
			参考 CCEN_CC1EN
9	CC3POL	R/W	配置捕获/比较通道 3 输出极性 (Capture/Compare Channel3 Output Polarity Configure) 参考 CCEN_CC1POL
10	保留		
11	CC3NPOL	R/W	配置捕获/比较通道 3 输出极性 Capture/Compare Channel3 Output Polarity Configure) 参考 CCEN_CC1NPOL
12	CC4EN	R/W	使能捕获/比较通道 4 输出 (Capture/Compare Channel4 Output Enable) 参考 CCEN_CC1EN
13	CC4POL	R/W	配置捕获/比较通道 4 输出极性 (Capture/Compare Channel4 Output Polarity Configure) 参考 CCEN_CC1POL
14	保留		
15	CC4NPOL	R/W	配置捕获/比较通道 4 输出极性 (Capture/Compare Channel4 Output Polarity Configure) 参考 CCEN_CC1NPOL

表格 53 标准 OCx 通道的输出控制位

CCxEN 位	OCx 输出状态
0	禁止输出 (OCx=0, OCx_EN=0)
1	OCx=OCxREF+极性, OCx_EN=1

注：连接到标准 OCx 通道的外部 I/O 引脚状态，取决于 OCx 通道状态和 GPIO 以及 AFIO 寄存器。

15.6.10 计数器寄存器 (TMRx_CNT)

偏移地址：0x24

复位值：0x0000

位/域	名称	R/W	描述
15:0	CNT	R/W	计数器数值 (Counter Value)
31:16	CNT	R/W	计数器数值 (Counter Value) (仅 TMR2)

15.6.11 预分频寄存器 (TMRx_PSC)

偏移地址：0x28

复位值：0x0000

位/域	名称	R/W	描述
15:0	PSC	R/W	预分频器数值 (Prescaler Value) 计数器的时钟频率 (CK_CNT) = f _{CK_PSC} / (PSC+1)

15.6.12 自动重装载寄存器 (TMRx_AUTORLD)

偏移地址：0x2C

复位值：0xFFFF FFFF

位/域	名称	R/W	描述
15:0	AUTORLD	R/W	自动重载数值 (Auto Reload Value) 自动重载的值为空时, 计数器不进行计数。
31:16	AUTORLD	R/W	自动重载数值 (Auto Reload Value) (仅 TMR2)

15.6.13 通道 1 捕获/比较寄存器 (TMRx_CC1)

偏移地址: 0x34

复位值: 0x0000

位/域	名称	R/W	描述
15:0	CC1	R/W	捕获/比较通道 1 数值 (Capture/Compare Channel1 Value) 捕获/比较通道 1 配置为输入模式: CC1 包含上次输入捕获通道 1 事件传输的计数器数值。 捕获/比较通道 1 配置为输出模式: CC1 包含了当前装入捕获/比较寄存器数值 捕获比较通道 1 的值 CC1 与计数器的值 CNT 比较, 在 OC1 上产生输出信号。 当输出比较预装载禁止 (TMRx_CCM1 寄存器的 OC1PEN=0) 时, 写入的数值会立即影响输出比较结果; 当输出比较预装载使能 (TMRx_CCM1 寄存器的 OC1PEN=1) 时, 写入的值会在产生更新事件时影响输出比较结果。
31:16	CC1	R/W	捕获/比较通道 1 数值 (Capture/Compare Channel1 Value) (仅 TMR2)

15.6.14 通道 2 捕获/比较寄存器 (TMRx_CC2)

TMR2 通道 2 捕获/比较寄存器 (TMRx_CC2)

偏移地址: 0x38

复位值: 0x0000

位/域	名称	R/W	描述
31:0	CC2	R/W	捕获/比较通道 2 数值 (Capture/Compare Channel2 Value) 参考 TMRx_CC1

TMR3/4 通道 2 捕获/比较寄存器 (TMRx_CC2)

偏移地址: 0x38

复位值: 0x0000

位/域	名称	R/W	描述
15:0	CC2	R/W	捕获/比较通道 2 数值 (Capture/Compare Channel2 Value) 参考 TMRx_CC1

15.6.15 通道 3 捕获/比较寄存器 (TMRx_CC3)

TMR2 通道 3 捕获/比较寄存器 (TMRx_CC3)

偏移地址: 0x3C

复位值: 0x0000

位/域	名称	R/W	描述
31:0	CC3	R/W	捕获/比较通道 3 数值 (Capture/Compare Channel3 Value) 参考 TMRx_CC1

TMR3/4 通道 3 捕获/比较寄存器 (TMRx_CC3)

偏移地址: 0x3C

复位值：0x0000

位/域	名称	R/W	描述
15:0	CC3	R/W	捕获/比较通道 3 数值 (Capture/Compare Channel3 Value) 参考 TMRx_CC1

15.6.16 通道 4 捕获/比较寄存器 (TMRx_CC4)

TMR2 通道 4 捕获/比较寄存器 (TMRx_CC4)

偏移地址：0x40

复位值：0x0000

位/域	名称	R/W	描述
31:0	CC4	R/W	捕获/比较通道 4 数值 (Capture/Compare Channel4 Value) 参考 TMRx_CC1

TMR3/4 通道 4 捕获/比较寄存器 (TMRx_CC4)

偏移地址：0x40

复位值：0x0000

位/域	名称	R/W	描述
15:0	CC4	R/W	捕获/比较通道 4 数值 (Capture/Compare Channel4 Value) 参考 TMRx_CC1

15.6.17 DMA 控制寄存器 (TMRx_DCTRL)

偏移地址：0x48

复位值：0x0000

位/域	名称	R/W	描述
4:0	DBADDR	R/W	设置 DMA 基地址 (DMA Base Address Setup) 这些位定义了 DMA 在连续模式下的基地址 (当对 TMRx_DMADDR 寄存器进行读或写时), DBADDR 定义为从 TMRx_CTRL1 寄存器所在地址开始的偏移量: 00000: TMRx_CTRL1 00001: TMRx_CTRL2 00010: TMRx_SMCTRL
7:5	保留		
12:8	DBLEN	R/W	设置 DMA 突发传输长度 (DMA Burst Transfer Length Setup) 这些位定义 DMA 在连续模式下的传输长度、传输的次数, 其中传输的数据可以是 16 位和 8 位。 当读写 TMRx_DMADDR 寄存器时, 定时器进行一次连续传输传输: 00000: 1 次传输 00001: 2 次传输 00010: 3 次传输 10001: 18 次传输 传输的地址公式如下: 传输的地址=TMRx_CTRL1 的地址 (从地址) +DBADDR+DMA 索引; DMA 索引=DBLEN

位/域	名称	R/W	描述
			<p>例如：DBLEN=7, DBADDR=TMR2_CTRL1（从地址）表示待传输数据的地址，则 TMRx_CTRL1 的地址+DBADDR+7，表示了将要写入/读出数据的地址，</p> <p>数据的传输将发生在：TMRx_CTRL1 的地址+从 DBADDR 开始的 7 个寄存器。</p> <p>根据设置的 DMA 数据长度不同，数据传输也会发生变化：</p> <p>当传输数据设置为 16 位时，数据会传输给 7 个寄存器</p> <p>当传输数据设置为 8 位时，第一个寄存器的数据是第一个的数据的 MSB 位，第二个寄存器的数据是第一个数据的 LSB 位，数据仍然会传输给 7 个寄存器。</p>
15:13	保留		

15.6.18 连续模式的 DMA 地址寄存器（TMRx_DMADDR）

偏移地址：0x4C

复位值：0x0000

位/域	名称	R/W	描述
15:0	DMADDR	R/W	<p>DMA 突发传送寄存器（DMA Register for Burst Transfer）</p> <p>TMRx_DMADDR 寄存器的读或写操作访问会导致对以下地址所在寄存器的存取操作：</p> <p>TMRx_CTRL1 地址+（DBADDR+DMA 索引）×4</p> <p>其中：</p> <p>“TMRx_CTRL1 地址”是控制寄存器 1（TMRx_CTRL1）所在的地址；</p> <p>“DBADDR”是 TMRx_DCTRL 寄存器中定义的基地址；</p> <p>“DMA 索引”是由 DMA 自动控制的偏移量，它取决于 TMRx_DCTRL 寄存器中定义的 DBLEN。</p>

16 基本定时器（TMR6/7）

16.1 简介

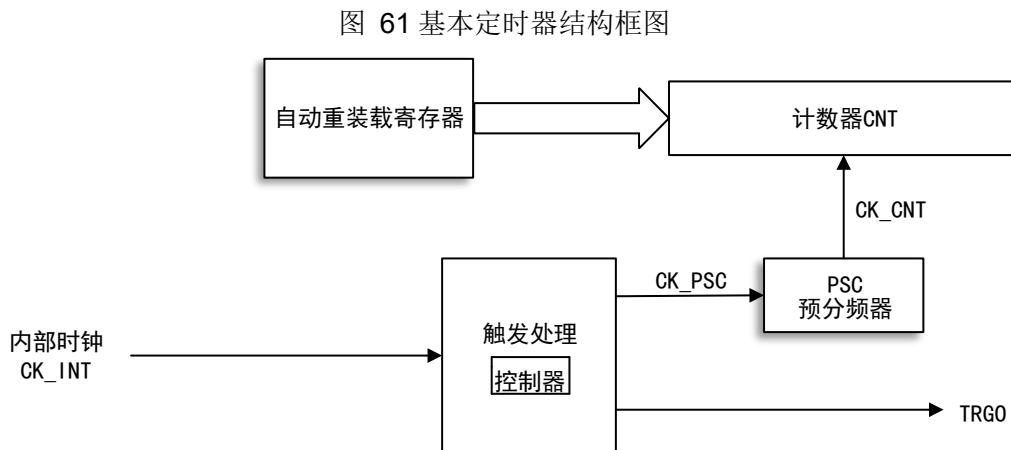
基本定时器 TMR6/TMR7 有一个无符号的 16 位计数器、自动重载寄存器、预分频器和触发控制器。

基本定时器为通用定时器提供时间基准，通过配置可以产生 DMA 请求。

16.2 主要特征

- (1) 计数器：16 位计数器，只能向上计数
- (2) 预分频器：16 位可编程预分频器
- (3) 时钟源：只有内部时钟

16.3 结构框图



16.4 功能描述

16.4.1 时钟源选择

基本定时器由内部时钟源 TMRx_CLK 驱动

配置 TMRx_CTRL1 寄存器的 CNTEN 位使能计数器，置位 CNTEN 位时，内部时钟 CK_INT 经过控制器和预分频器产生 CK_CNT 驱动计数器。

16.4.2 时基单元

基本定时器的时基单元包含三个寄存器：

- 计数器寄存器（CNT）16 位
- 自动重载寄存器（AUTORLD）16 位
- 预分频器（PSC）16 位

计数器 CNT

基本定时器中只有一种计数模式：向上计数模式

向上计数模式

当计数器处于向上计数模式时，计数器从 0 开始向上计数，当每来一个脉冲计数器就会增加 1，一直到计数器（TMRx_CNT）值与自动重载

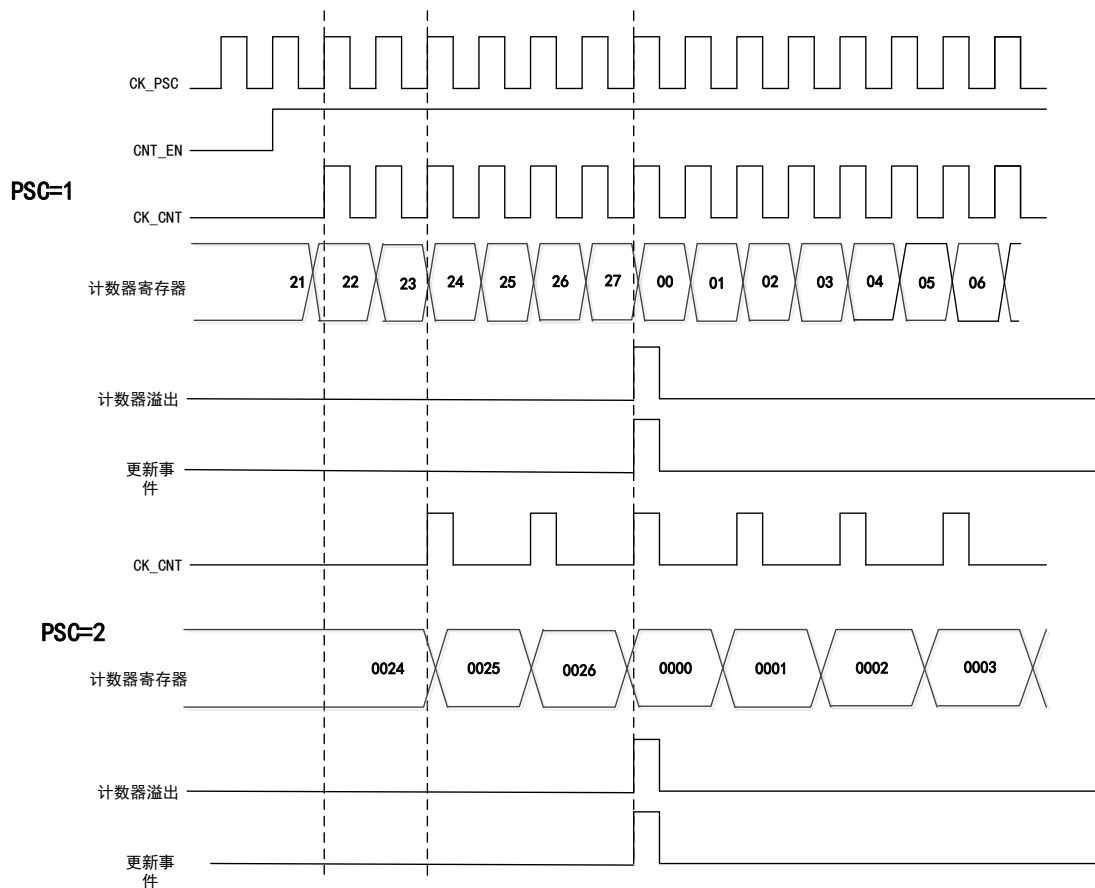
（TMRx_AUTORLD）的值相等，然后计数器会重新从 0 开始计数，产生一个计数器向上溢出事件，自动重载的值（TMRx_AUTORLD）是提前写入的。

禁止更新事件，设置 TMRx_CTRL1 寄存器中的 UD 位置 1。

产生更新中断或 DMA 请求，设置 TMRx_CTRL1 寄存器中的 URSEL 位。

在发生一个更新事件时，自动重载寄存器和预分频寄存器都将会被更新。

图 62 计数器时序图，内部时钟分频系数为 1 或 2



预分频器 PSC

预分频器是 16 位的且是可编程的，它可以将计数器的时钟频率进行 1~65536 之间任意值的分频（由 TMRx_PSC 寄存器控制），经过分频后的时钟将会驱动计数器 CNT 计数。预分频器带有缓冲器，它能够在运行中被改变。

16.5 寄存器地址映射

下表中将 TMR6/TMR7 的所有寄存器映射到一个 16 位可寻址（编址）空间。

表格 54 TMR6 和 TMR7 寄存器地址映射

寄存器名	描述	偏移地址
TMRx_CTRL1	控制寄存器 1	0x00
TMRx_CTRL2	控制寄存器 2	0x04
TMRx_DIEN	DMA/中断使能寄存器	0x0C
TMRx_STS	状态寄存器	0x10
TMRx_CEG	控制事件产生寄存器	0x14
TMRx_CNT	计数器寄存器	0x24
TMRx_PSC	预分频寄存器	0x28
TMRx_AUTORLD	自动重载寄存器	0x2C

16.6 寄存器功能描述

16.6.1 控制寄存器 1 (TMRx_CTRL1)

偏移地址：0x00

复位值：0x0000

位/域	名称	R/W	描述
0	CNTEN	R/W	使能计数器 (Counter Enable) 0: 禁止 1: 使能 定时器配置为外部时钟、门控模式和编码器模式时，需要通过软件对该位写 1 启动定期工作；配置为触发模式时，可硬件写 1。
1	UD	R/W	禁止更新 (Update Disable) 更新事件可引起 AUTORLD、PSC、CCx 产生更新设置的数值。 0: 允许更新事件 (UEV) 更新事件可以由以下任一情况产生： 计数器溢出/下溢； 设置 UEG 位； 从模式控制器产生的更新。 1: 禁止更新事件
2	URSSEL	R/W	更新请求源 (Update Request Source Select) 如果使能了中断或 DMA，更新事件可产生更新中断或 DMA 请求，通过该位可选择不同的更新请求源。 0: 计数器上溢或下溢 设置 UEG 位 通过从模式控制器产生的更新 1: 计数器上溢或下溢

位/域	名称	R/W	描述
3	SPMEN	R/W	使能单脉冲模式（Single Pulse Mode Enable） 产生更新事件时，可改变通道的输出电平；在该模式下，会清除 CNTEN 位，停止计数器，后续不再改变通道的输出电平。 0：禁止 1：使能
6:4	保留		
7	ARPEN	R/W	TMRx_AUTORLD 寄存器自动重装缓冲使能（Auto-reload Preload Enable） 禁止缓存区时，程序修改 TMRx_AUTORLD 会立刻修改装入计数器的数值；使能缓存区时，程序修改 TMRx_AUTORLD 会在下一个更新事件修改装入计数器的数值。 0：禁止 1：使能
15:8	保留		

16.6.2 控制寄存器 2（TMRx_CTRL2）

偏移地址：0x04

复位值：0x0000

位/域	名称	R/W	描述
3:0	保留		
6:4	MMSEL	R/W	选择定时器主模式下用于 TRGO 的信号（Master Mode Signal Select） 工作在主模式的定时器的信号可用于 TRGO，从而影响处在从模式且与主定时器级联的的定时器工作，具体影响可从模式的定时器配置有关。 000：复位，主模式定时器的复位信号用于 TRGO 001：使能，主模式定时器的计数器使能信号用于 TRGO 010：更新，主模式定时器的更新事件用于 TRGO
15:7	保留		

16.6.3 DMA/中断使能寄存器（TMRx_DIEN）

偏移地址：0x0C

复位值：0x0000

位/域	名称	R/W	描述
0	UIEN	R/W	使能更新中断（Update Interrupt Enable） 0：禁止 1：使能
7:1	保留		
8	UDIEN	R/W	使能更新的 DMA 请求（Update DMA Request Enable） 0：禁止 1：使能
15:9	保留		

16.6.4 状态寄存器（TMRx_STS）

偏移地址：0x10

复位值：0x0000

位/域	名称	R/W	描述
0	UIFLG	RC_W0	产生更新事件中断标志位 (Update Event Interrupt Generate Flag) 0: 没有发生更新事件中断 1: 发生更新事件中断 计数器数值重新装载或重新初始化时, 会产生更新事件, 该位由硬件置 1, 软件清 0; 更新事件的产生的情况有以下情况: (1) TMRx_CTRL1 寄存器的 UD=0, 重复计数器数值上/下溢时产生更新事件; (2) TMRx_CTRL1 寄存器的 URSSEL=0 和 UD=0, 配置 TMRx_CEG 寄存器的 UEG=1 产生更新事件, 需要通过软件初始化计数器; (3) TMRx_CTRL1 寄存器的 URSSEL=0 和 UD=0, 计数器被触事件初始化时产生更新事件。
15:1	保留		

16.6.5 控制事件产生寄存器 (TMRx_CEG)

偏移地址：0x14

复位值：0x0000

位/域	名称	R/W	描述
0	UEG	W	产生更新事件 (Update Event Generate) 0: 无效 1: 初始化计数器, 产生更新事件 此位由软件置 1, 硬件清 0。 注意: 产生更新事件时, 预分频器的计数器会清 0, 但是预分频系数不变。如果在向下计数模式下计数器会读取 TMRx_AUTORLD 的值; 如果在中央对齐模式下或者向上计数模式中计数器会被清 0。
15:1	保留		

注: 连接到标准 OCx 通道的外部 I/O 引脚状态, 取决于 OCx 通道状态和 GPIO 以及 AFIO 寄存器。

16.6.6 计数器寄存器 (TMRx_CNT)

偏移地址：0x24

复位值：0x0000

位/域	名称	R/W	描述
15:0	CNT	R/W	计数器数值 (Counter Value)

16.6.7 预分频寄存器 (TMRx_PSC)

偏移地址：0x28

复位值：0x0000

位/域	名称	R/W	描述
15:0	PSC	R/W	预分频器数值 (Prescaler Value) 计数器的时钟频率 (CK_CNT) = $f_{CK_PSC} / (PSC+1)$ 。

16.6.8 自动重装载寄存器 (TMRx_AUTORLD)

偏移地址：0x2C

复位值：0xFFFF

位/域	名称	R/W	描述
15:0	AUTORLD	R/W	自动重装数值 (Auto Reload Value) 自动重装载的值为空时，计数器不进行计数。

17 红外定时器 (IRTMR)

17.1 简介

IRTMR 是一个用于遥控的红外接口，可以使用一个红外 LED 来实现遥控功能。

17.2 功能描述

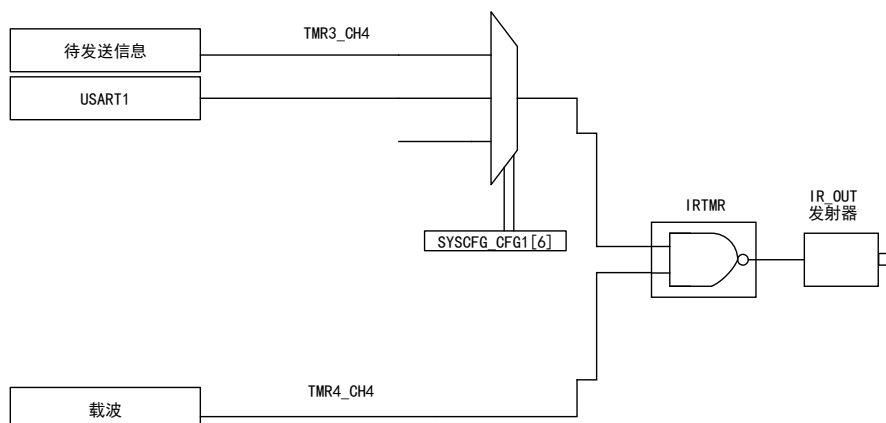
17.2.1 IRTMR 接收

红外接收可通过外部 IR 接收器模块的输出连接到控制器的 GPIO 或者定时器的输入捕获通道来实现数据的接收。

17.2.2 IRTMR 发送

IRTMR 内部连接到 TMR3 和 TMR4，具体框图如下：

图 63 IRTMR 结构框图



为了产生正确的红外遥控信号，需要正确配置 TMR4_CH4 提供一个高频载波信号，而 TMR3_CH4 提供我们发送的信息或者通过配置 SYSCFG_CFG1 寄存器的 IRSEL 位选择 USART1 来产生调制包络。

最终的调制信号通过 IR_OUT 引脚输出，通过使能 GPIOx_ALF_x 寄存器中的相关复用功能位来激活此功能。

当控制需要大电流驱动红外 LED 时，配置 SYSCFG_CFG1 寄存器中的 I2CPB9FMP，使能 PB9 的超快模式以支持大电流输出。（仅在 PB9 上使用此功能）

18 看门狗定时器 (WDT)

18.1 简介

看门狗是用来监测由软件错误引起的系统故障。片上一共有两个看门狗设备：独立看门狗和窗口看门狗，它们提高了安全性、使时间更加精确，使用变得更加灵活。

独立看门狗在计数器减到 0 时会发生复位，以及当计数器的值大于窗口值时，被重载也会产生复位。

窗口看门狗在计数器减到 0x3F 时会发生复位，当计数器计数的值在配置寄存器的窗口值之前，刷新计数器也会发生复位。

18.2 独立看门狗

18.2.1 简介

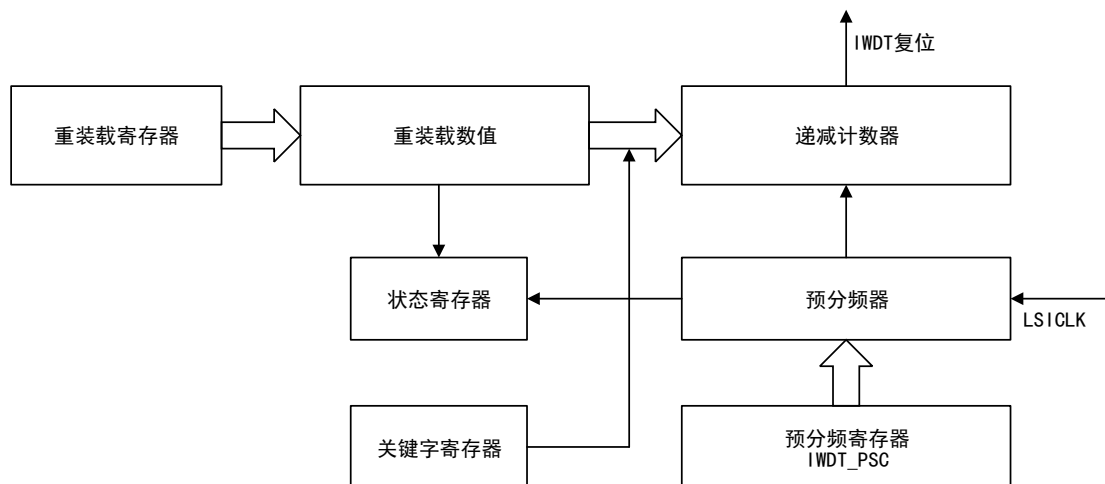
独立看门狗是由一个八位预分频器 IWDT_PSC、12 位递减计数器、12 位重载寄存器 IWDT_CNTRLD、关键字寄存器 IWDT_KEY、状态寄存器 IWDT_STS 以及窗口寄存器 IWDT_WIN 组成。

独立看门狗有独立的时钟源，即使是在主时钟失效的情况下，它仍然有效。

独立看门狗适用于需要独立的环境但对精确度要求不高的场所。

18.2.2 结构框图

图 64 独立看门狗结构框图



注：预分频器、重载数值和递减计数器在 V_{DD} 供电区内；预分频寄存器、状态寄存器、重载寄存器和关键字寄存器在 1.5V 供电区。看门狗功能处于 V_{DD} 供电区，可以在停机或者待机模式下正常工作。

18.2.3 功能描述

18.2.3.1 关键字寄存器

在关键字寄存器中写入 0xCCCC 开启独立看门狗，此时计数器由复位值 0xFFFF 开始向下计数，当计数器计数到 0x000 时，产生一次复位。

在关键字寄存器中写入 0xAAAA，重装载寄存器的值会被重新加载到计数器，以此来阻止看门狗复位。

在关键字寄存器中写入 0x5555，用来改写预分频寄存器、重装载寄存器和窗口值寄存器的值。

18.2.3.2 窗口寄存器

窗口寄存器 IWDT_WIN 的默认值为 0xFFFF，在没有更新的情况下，禁用窗口选项，当窗口值被改变时，会执行重装载操作，看门狗计数器的值会被置为 IWDT_CNTRLD 的值，可以延缓复位需要的事件周期。

独立看门狗可以工作在窗口看门狗模式下，需要设置窗口寄存器 IWDT_WIN 适当的值。

18.2.3.3 配置 IWDT

使用窗口寄存器时配置 IWDT

- 使能 IWDT（将 0xCCCC 写入关键字寄存器 IWDT_KEY）
- 打开寄存器访问许可（将 0x5555 写入关键字寄存器 IWDT_KEY）
- 配置 IWDT_PSC 预分频寄存器（将 0~7 的值写入 IWDT_PSC）
- 等待状态寄存器 IWDT_STS 的值更新为 0x00
- 配置窗口寄存器 IWDT_WIN（可以将自动重装载寄存器 IWDT_CNTRLD 的值更新到看门狗寄存器）

注意：状态寄存器 IWDT_STS 的值为 0x00 时，对窗口值进行写操作使自动重装载的值刷新计数器

禁止窗口寄存器时配置 IWDT

- 使能 IWDT（将 0xCCCC 写入关键字寄存器 IWDT_KEY）
- 打开寄存器访问许可（将 0x5555 写入关键字寄存器 IWDT_KEY）
- 配置 IWDT_PSC 预分频寄存器（将 0~7 的值写入 IWDT_PSC）
- 配置重装载寄存器 IWDT_CNTRLD
- 等待状态寄存器 IWDT_STS 的值更新为 0x00
- 使用 IWDT_CNTRLD 寄存器刷新看门狗的计数器

18.2.3.4 寄存器访问保护

预分频寄存器 IWDT_PSC、重装载寄存器 IWDT_CNTRLD 和窗口寄存器 IWDT_WIN 具有写保护功能，如果要改写这三个寄存器则需要写 0x5555 到关键字寄存器中，如果将其他的值写入关键字寄存器将会再次启动对寄存器的保护。

写 0xAAAA 到关键字寄存器中，也会启动写保护功能。

可以通过状态寄存器来观察预分频寄存器、重装载寄存器和窗口寄存器。

18.2.3.5 硬件看门狗

启动了“硬件看门狗”功能之后，当系统上电复位，看门狗会自动运行。如果没有向关键字寄存器中写入 0xAAAA，计数器计数结束后，产生复位。

18.2.3.6 调试模式

独立看门狗可以在调试模式下配置，选择停止还是继续工作。取决于 DBGMCU 模块中的 DBGMCU_APB1F 寄存器的 IWDT_STS 位。

18.3 窗口看门狗

18.3.1 简介

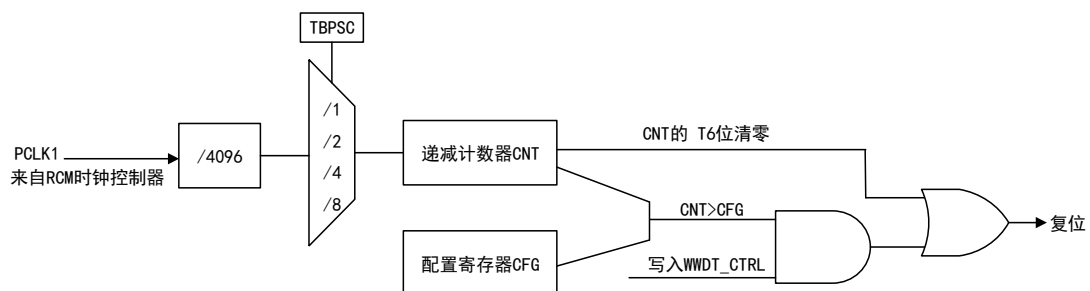
窗口看门狗含有一个 7 位自由运行递减计数器、预分频器、控制寄存器 WWDT_CTRL、配置寄存器 WWDT_CFG 和状态寄存器 WWDT_STS。

窗口看门狗时钟来自 PCLK，计数器时钟由 CK 计数器时钟经过预分频分频得到（由配置寄存器配置）。

窗口看门狗适用于需要精确计时的场合。

18.3.2 结构框图

图 65 窗口看门狗结构框图



18.3.3 功能描述

使能窗口看门狗定时器，复位条件：

- 计数器计数小于 0x40 时，产生复位。
- 计数器计数到窗口寄存器的值之前，重装载计数器产生复位。

发生复位后，看门狗总是关闭状态，需要设置 WWDT_CTRL 寄存器的 WWDTEN 位才能开启看门狗。

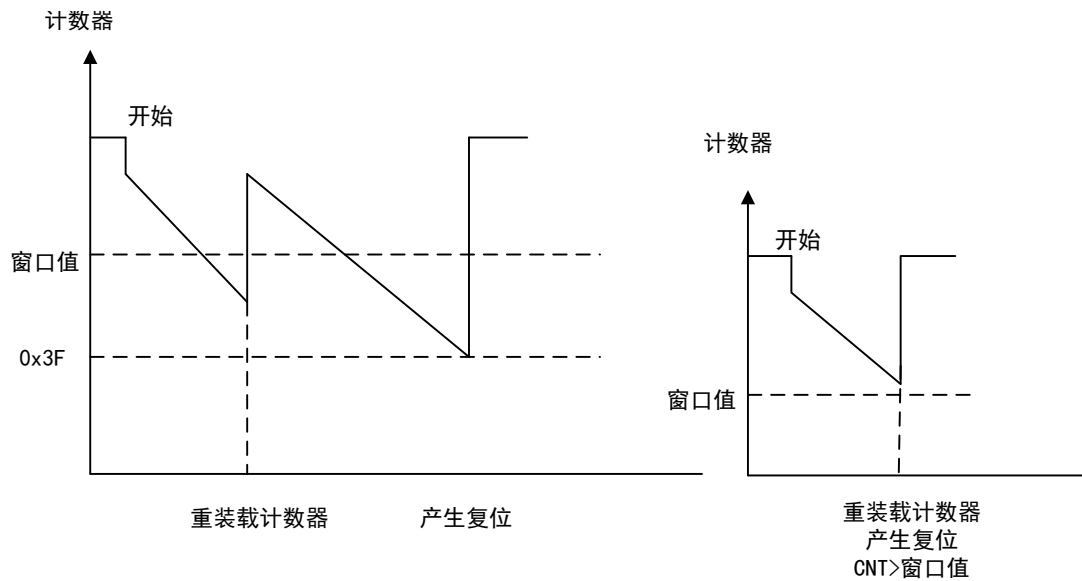
窗口看门狗的计数器是自由状态，在看门狗禁止的情况下，计数器仍然继续递减计数。计数器必须要在窗口寄存器的值和 0x40 之间重装载，来避免复位。

设置配置寄存器的 EWIEN 位可以开启提前唤醒中断，当计数到 0x40 会产生中断，进入中断服务程序（ISTS）可以用来防止窗口看门狗复位。如果要清除 EWIEN 中断需要在状态寄存器中写入 0。

窗口看门狗特有的窗口可以有效地监测程序是否故障，例如：假定一个程序段运

行的时间是 T,设定窗口寄存器的值稍稍小于 (TR-T) 如果在窗口内没有重装载寄存器,说明程序出现了故障,当计数器计数到 0x3F 时,则会产生复位。

图 66 窗口看门狗时序图



窗口看门狗定时器超时的计算公式:

$$T_{WWDt} = T_{PCLK1} \times 2^{WTB} \times (T[5:0] + 1)$$

其中:

- T_{WWDt} : WWDt 超时时间
- T_{PCLK1} : APB 以 ms 为单位的时钟周期

表格 55 在 PCLK1=36MHz 时的最小/最大超时值

WTB	最小超时值	最大超时值
0	113μs	7.28ms
1	227μs	14.56ms
2	455μs	29.12ms
3	910μs	58.25ms

18.3.3.1 调试模式

窗口看门狗可以在调试模式下配置,选择停止还是继续工作。取决于 DBGMCU 模块中的 DBGMCU_APB1F 寄存器的 WWDt_STS 位。

18.4 IWDT 寄存器地址映射

表格 56 IWDT 寄存器地址映射

寄存器名	描述	偏移地址
IWDT_KEY	关键字寄存器	0x00

寄存器名	描述	偏移地址
IWDT_PSC	预分频寄存器	0x04
IWDT_CNTRLD	计数器重载寄存器	0x08
IWDT_STS	状态寄存器	0x0C
IWDT_WIN	窗口寄存器	0x10

18.5 IWDT 寄存器功能描述

可以用半字（16 位）或字（32 位）的方式操作这些外设寄存器。

18.5.1 关键字寄存器（IWDT_KEY）

偏移地址：0x00

复位值：0x0000 0000（待机模式时复位）

位/域	名称	R/W	描述
15:0	KEY	W	允许访问 IWDT 寄存器键值（Allow Access IWDT Register Key Value） 写入 0x5555 表示允许访问 IWDT_PSC、IWDT_CNTRLD 和 IWDT_WIN 寄存器。 软件写入 0xAAAA 表示执行重载计数器，需要一定的间隔写入，防止看门狗复位。 写入 0xCCCC，启动看门狗（硬件看门狗不受此命令字限制）。 读出值为 0x0000。
31:16			保留

18.5.2 预分频寄存器（IWDT_PSC）

偏移地址：0x04

复位值：0x0000 0000

位/域	名称	R/W	描述
2:0	PSC	R/W	配置预分频系数（Prescaler Factor Configure） 有写保护功能，向 IWDT_KEY 寄存器写入 0x5555 时允许访问该寄存器；写该寄存器过程中，当 IWDT_STS 寄存器的 PSCUFLG=0 时，才能改变预分频因子；读该寄存器过程中，当 PSCUFLG=0 时，读取 PSC 寄存器的数值才有效。 000: PSC=4 001: PSC=8 010: PSC=16 011: PSC=32 100: PSC=64 101: PSC=128 110: PSC=256 111: PSC=256
31:3			保留

18.5.3 计数器重载寄存器（IWDT_CNTRLD）

偏移地址：0x08

复位值：0x0000 0FFF（待机模式时复位）

位/域	名称	R/W	描述
11:0	CNTRLD	R/W	设置看门狗计数器重载值 (Watchdog Counter Reload Value Setup) 有写保护功能, 定义 IWDT_KEY 寄存器写入 0xAAAA 时载入看门狗计数器的值; 在写该寄存器过程中, CNTUFLG=0 时, 才能对此寄存器进行修改。在读该寄存器过程中, 当 IWDT_STS 寄存器中的 CNTUFLG=0 时, 读取的数值才是有效的。 看门狗超时周期可通过此重载值和时钟预分频值来计算。
31:12	保留		

18.5.4 状态寄存器 (IWDT_STS)

偏移地址: 0x0C

复位值: 0x0000 0000 (待机模式时不复位)

位/域	名称	R/W	描述
0	PSCUFLG	R	看门狗预分频值更新标志 (Watchdog Prescaler Value Update Flag) 更新预分频器系数时, 由硬件置 1; 预分频器系数更新结束后, 由硬件清 0; 预分频器系数只有在 PSCUFLG 位被清 0 时更新。
1	CNTUFLG	R	看门狗计数器重载值更新标志 (Watchdog Counter Reload Value Update Flag) 更新计数器重载值时, 由硬件置 1; 计数器重载值更新结束后, 由硬件清 0; 计数器重载值只有在 CNTUFLG 位被清 0 时更新。
2	WINUFLG	R	看门狗计数器窗口值更新标志 (Watchdog Counter Window Value Update Flag) 更新窗口值时, 由硬件置 1。计数器的窗口值更新结束后, 该位由硬件清 0, 窗口值只有在使能 IWDT_WIN 寄存器时才有效。
31:3	保留		

18.5.5 窗口寄存器 (IWDT_WIN)

偏移地址: 0x10

复位值: 0x0000 0FFF (待机模式时复位)

位/域	名称	R/W	描述
11:0	WIN	R/W	设置看门狗计数器窗口值 (Watchdog Counter Window Value) 这些位包含窗口值和向下计数器的初始值 要在 STS_WINUFLG=0 时才能修改这些位 在计数器的值和窗口值之间对计数器重载可以防止产生复位 注: 读取此寄存器时返回 V _{DD} 供电域的值, 因此如果要读取数据要保证 STS_WINUFLG=0。
31:12	保留		

注: 如果重载设置、预分频设置和窗口值设置在运行, 如果要改变重载值、预分频值和窗口值需要先确认相关的标志位为 0。更新之后不需要再等待, 除非要进入低功耗模式。

18.6 WWDT 寄存器地址映射

表格 57 WWDT 寄存器地址映射

寄存器名	描述	偏移地址
WWDT_CTRL	控制寄存器	0x00
WWDT_CFG	配置寄存器	0x04
WWDT_STS	状态寄存器	0x08

18.7 WWDT 寄存器功能描述

可以用半字（16 位）或字（32 位）的方式操作这些外设寄存器。

18.7.1 控制寄存器（WWDT_CTRL）

偏移地址量：0x00

复位值：0x0000 007F

位/域	名称	R/W	描述
6:0	CNT	R/W	设置计数器数值（Counter Value Setup） 该计数器是 7 位，CNT6 是最高有效位 这些位用来存储看门狗的计数器值，当计数值从 0x40 减到 0x3F 时，产生 WWDT 复位。
7	WWDTEN	R/S	使能窗口看门狗（Window Watchdog Enable） 该位由软件置 1 且只能在复位后由硬件清除。当 WWDTEN=1, WWDT 可以产生复位。 0：禁止 1：使能
31:8	保留		

18.7.2 配置寄存器（WWDT_CFG）

偏移地址量：0x04

复位值：0x0000 007F

位/域	名称	R/W	描述
6:0	WIN	R/W	设置窗口值（window Value Setup） 该窗口值是 7 位，用来和向下计数器作比较。
8:7	TBPSC	R/W	配置时基预分频系数（Timer Base Prescaler Factor Configure） PCLK1/4096 的基础上再分频 00：不分频 01：2 分频 10：4 分频 11：8 分频
9	EWIEN	R/S	使能提前唤醒中断（Early Wakeup Interrupt Enable） 0：无意义 1：则当计数器值达到 0x40 时，即产生中断；此中断在复位后由硬件清除。
31:10	保留		

18.7.3 状态寄存器 (WWDT_STS)

偏移地址: 0x08

复位值: 0x0000 0000

位/域	名称	R/W	描述
0	EWIFLG	RC_W0	产生提前唤醒中断标志 (Early Wakeup Interrupt Occur Flag) 0: 未发生 1: 当计数器值达到 0x40 时由硬件置 1; 若中断未被使能, 该位也会被置 1。 由软件写 0 清除。对该位写 1 无效。
31:1	保留		

19 实时时钟（RTC）

19.1 术语全称、缩写描述

表格 58 术语全称、缩写描述

中文全称	英文全称	英文缩写
秒	Second	SEC
闹钟	Alarm	ALR
预分频器	Prescaler	PSC

19.2 简介

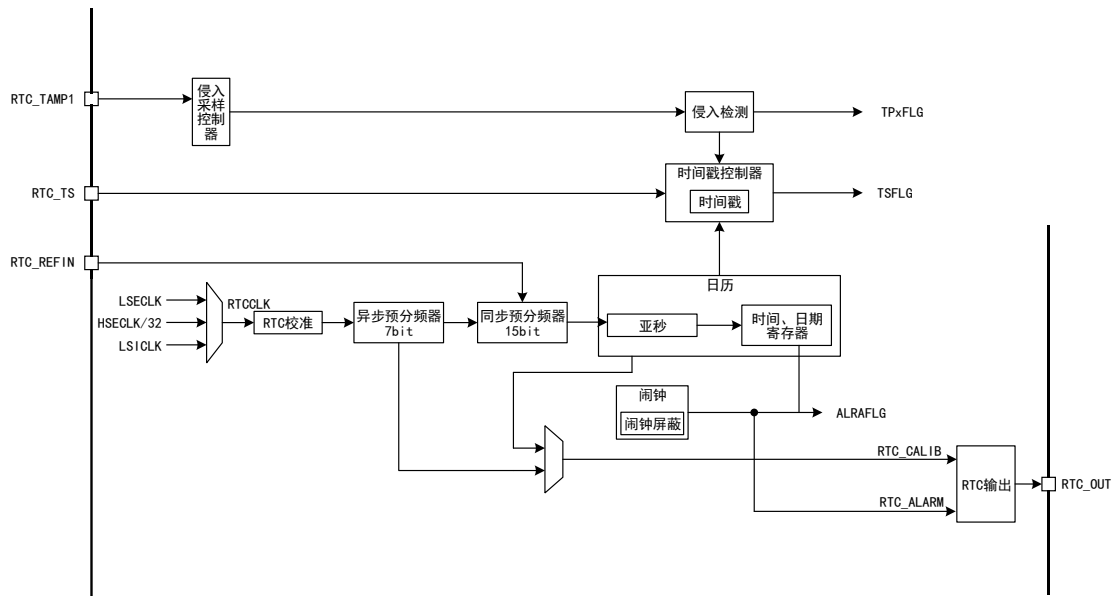
拥有采取 BCD 编码的亚秒、时间、日期寄存器，以及相应的闹钟寄存器，结合外部引脚可实现时间戳功能。带时钟校准功能及时间补偿。

19.3 主要特征

- (1) 时基单元
- (2) 时钟校准
- (3) 亚秒、时间、日期
- (4) 时间误差补偿
- (5) 闹钟（亚秒、时间、日期屏蔽）
- (6) 时间戳
- (7) 侵入检测
- (8) 3 种 RTC 输出
- (9) 多种中断控制

19.4 结构框图

图 67 RTC 结构框图



注意:

- 1) 复用功能输出: RTC_OUT 是由以下两种的其中一种形式输出
 - RTC_CALIB: 通过 RTC_CTRL 寄存器的 CALOEN 位开启此输出, 当 LSECLK 的频率是 32.768kHz 时, 时钟输出为 512Hz 或者 1Hz。
 - RTC_ALARM: 通过 RTC_CTRL 寄存器的 OUTSEL 位开启此输出, 闹钟 A。
- 2) 复用功能输入:
 - RTC_TS: 时间戳事件
 - RTC_TAMP1: 侵入事件检测 1
 - RTC_REFIN: 50 或 60Hz 参考时钟输入

19.5 功能描述

19.5.1 RTC 控制的 I/O 引脚

RTC 中的 RTC_OUT、RTC_TS 和 RTC_TAMP1 可以映射到同一个引脚 (PC13) 上。

RTC_ALARM 的输出选择通过 RTC_TACFG 寄存器配置, 其中 RTC_TACFG 寄存器的 PC13VAL 位用来选择 RTC_ALARM 配置为推挽输出还是开漏输出。

当 PC13 引脚不使用 RTC 复用功能时, 通过设置 RTC_TACFG 寄存器的 PC13EN 位将 PC13 引脚强制为推挽输出; PC13VAL 位设置 PC13 引脚输出数据的值。此时 PC13 引脚推挽输出和数据的值可以保持在待机模式下。

下表是输出机制遵循的优先级顺序:

表格 59 RTC 控制的 PC13 引脚

引脚配置和功能	RTC_ALARM 输出使能	RTC_CALIB 输出使能	RTC_TAMP1 输入使能	RTC_TS 输入使能	PC13EN	PC13VAL
RTC_ALARM 开漏输出	1	无影响	无影响	无影响	无影响	0
RTC_ALARM 推挽输出	1	无影响	无影响	无影响	无影响	1
RTC_CALIB 推挽输出	0	1	无影响	无影响	无影响	无影响
RTC_TAMP1 浮空输入	0	0	1	0	无影响	无影响
RTC_TS 和 RTC_TAMP1 浮空输入	0	0	0	1	无影响	无影响
RTC_TS 浮空输入	0	0	0	1	无影响	无影响
强制为推挽输出	0	0	0	0	1	PC13 输出数 据值
唤醒引脚或者标 准 GPIO	0	0	0	0	0	无影响

当 PC14 和 PC15 不使用 LSECLK 振荡器时，通过设置 RTC_TACFG 寄存器的 PC14EN 和 PC15EN 位将 PC14/PC15 引脚强制为推挽输出；PC14VAL 和 PC15VAL 位设置输出的数据，此时 PC14 和 PC15 推挽输出和数据的值可以保持在待机模式下。

下表是输出机制遵循的优先级顺序：

表格 60 LSECLK 控制的 PC14 引脚

引脚配置和功能	RCM_RTCCTRL 寄存器的 LSEEN 位	RCM_RTCCTRL 寄存器的 LSEBCFG 位	PC14EN	PC14VAL
LSECLK 振荡器	1	0	无影响	无影响
LSECLK 旁路	1	1	无影响	无影响
强制为推挽输出	0	无影响	1	PC14 输出数据的值
标准 GPIO	0	无影响	0	无影响

表格 61 LSECLK 控制的 PC15 引脚

引脚配置和功能	RCM_RTCCTRL 寄存器的 LSEEN 位	RCM_RTCCTRL 寄存器的 LSEBCFG 位	PC15EN	PC15VAL
LSECLK 振荡器	1	0	无影响	无影响
强制为推挽输出	1	1	1	PC15 输出数据的值
	0	无影响		
标准 GPIO	0	无影响	0	无影响

19.5.2 时基单元

时钟源

RTC 有 3 个时钟源 RTC_CLK:

- 外部 LSECLK 晶振
- 外部 HSECLK 晶振
- 内部 LSICLK

不同时钟源通过时钟控制器 RCM 外设配置。

预分频器

RTC 外设的功耗需尽量降到最低，为了兼顾功耗问题，RTC 内部采用双预分频器，7bit 的异步预分频器 APSC、15bit 的同步预分频器 SPSC。

RTC_CLK 先经过异步预分频器，分频后的时钟到达同步预分频器，合理配置两个预分频器，能产生 1Hz 的时钟供日期使用。

使用预分频器时，建议异步预分频器尽可能调高，降低功耗。

同步预分频值同时也会用作亚秒计数器重载值。

19.5.3 时钟校准

时钟同步

RTC 根据外部高精度时钟，结合寄存器 RTC_SHIFT 可实现时钟同步。主要通过前后两次获取亚秒时段的时间戳，检测到 RTC 时钟与外部时钟的偏差。由于同步预分频值用作亚秒计数器重载值，而寄存器 RTC_SHIFT 的 SFSEC 位又作用于亚秒计数器，因而调节 SFSEC 位，即可实现对 RTC 时钟的微调，人为增加、减少几个周期。

参考时钟

RTC 内部拥有参考时钟检测，可用来补偿外部 LSECLK 晶振的偏差。设置 RCLKDEN 位使能参考时钟检测，外部 50Hz 或 60Hz 的参考时钟经 RTC_REFIN 引脚与 RTC 内部 1Hz 时钟比较，通过这种机制自动补偿 LSECLK 分频后的 1Hz 时钟。

使能参考时钟检测后，时钟单元的同步、异步预分频器必须配置为默认值。

参考时钟检测不能与时钟同步同时使用，在待机模式下需要禁用参考时钟检测。

RTC 数字校准

RTC 默认采用 2^{20} 个 RTC_CLK 为一个校准周期，另外也可通过寄存器 CALW16、CALW8 设置为 2^{19} 、 2^{18} 个 RTC_CLK 为一个校准周期。当采用 LSECLK 作 RTC_CLK 时钟源，RTC 的校准周期为 32s、16s、8s。

- 16s 校准周期，硬件设定 RECALF[0]为 '0'

- 8s 校准周期，硬件设定 RECALF[1:0]为 ‘00’

以 32s 的校准周期为例，校准的机制是在校准周期内，增添或减少一部分 RTC_CLK 信号。

- 当使用 RECALF 时，每 2^{20} 个 RTC_CLK 减少 RECALF 个 RTC_CLK
- 当使用 ICALFEN，ICALFEN=1 时，每 2^{11} 个 RTC_CLK 增添 1 个 RTC_CLK
- 当使用 RECALF，ICALFEN 时，每 2^{20} 个 RTC_CLK 增添 $(512 * ICALFEN - RECALF)$ 个 RTC_CLK

19.5.4 RTC 写保护

为防止意外写入导致 RTC 计数异常，RTC 寄存器采用写保护机制，只有解除写保护，才能对写保护的寄存器进行操作。

上电后，RTC 寄存器进入写保护状态，系统复位不能解除写保护。当需要解除写保护时，通过往寄存器 RTC_WRPROT 依次写入特殊的关键字 ‘0xCA’、‘0x53’。如果写入错误的关键字，RTC 立马又开启写保护。

19.5.5 日期寄存器

RTC 拥有采用 BCD 编码的亚秒、时间、日期影子寄存器，分别为 RTC_SUBSEC、RTC_TIME、RTC_DATE。访问影子寄存器可以获得当前的日期，或者直接从日期寄存器获取日期。通过配置寄存器 RTC_CTRL 的 TIMEFCFG 位，可选取 24 小时、12 小时的时间制。

RTC 每两个 RTC_CLK 周期更新影子寄存器，并将标志位 RSFLG 置起，当从停机或待机唤醒后，一般不会更新影子寄存器，需要等待最多两个 RTC_CLK 周期。系统复位引发影子寄存器的复位。

影子寄存器与 f_{APB1} 同步。

通过配置寄存器 RTC_CTRL 的 RCMCFG 位，可选择读取日期的方式。

RCMCFG=0，从影子寄存器读取日期

该模式下，建议 f_{APB1} 大于 $7 * f_{RTC_CLK}$ ，当 f_{APB1} 过小，为确保读取日期值正常，需两次读取影子寄存器，两次获取的日期相同，则日期读取成功。

影子寄存器更新后，标志位 RSFLG 置起，软件必须等待 RSFLG 置位后读取日期，每次读取日期后，需手动清除 RSFLG 标志。

从停机或待机唤醒，由于影子寄存器未更新，因而第一时间需要清除 RSFLG 标志。

RCMCFG=1，从日期寄存器读取日期

当 f_{APB1} 小于 $7 * f_{RTC_CLK}$ ，或者从低功耗唤醒时，建议直接从日期寄存器读取日期。

读取日期时未等待 RSFLG 标志置 1，并且刚好处于日期寄存器变化的阶段，则需读取两次日期。因而也建议读取两次日期寄存器，当两次读取日期值相同时，

说明日期读取成功。

19.5.6 时间补偿

由于季节变化，有些时候需要进行时间补偿，使其更适合日常需求。RTC 内部集成时间补偿单元及其夏令时标志，用户可根据自身需求选择是否开启时间补偿。

通过设置寄存器 RTC_CTRL 的 STCCFG 位，令夏季增加 1 小时；设置寄存器 RTC_CTRL 的 WTCCFG 位，令冬季减少 1 小时。BAKP 标志用于记录夏令时是否设置。

19.5.7 可编程的闹钟

作为一个实时时钟，RTC 内部集成闹钟功能，主要通过闹钟配置寄存器、闹钟屏蔽，结合日期寄存器运作。

通过寄存器 RTC_ALRMA、RTC_ALRMAS 配置闹钟及其闹钟屏蔽，闹钟屏蔽告知 RTC 需要关注闹钟的时段。使能闹钟功能后，当关注的时段到了设定值时，闹钟才会触发，此时闹钟标志置起，如果开启了闹钟中断，则触发中断处理。

选取“秒”作为闹钟的时段，同步预分频值需要大于 3，闹钟才能正常运作。

19.5.8 时间戳

RTC 具有时间戳功能，RTC_TS 引脚结合时间戳寄存器工作。

通过寄存器 RTC_CTRL 的 TSETCFG 位设置时间戳检测极性，当 RTC_TS 引脚识别到外部时间戳边沿信号时，RTC 自动将当前日期锁存在亚秒、时间、日期时间戳寄存器中，同时时间戳标志位 TSFLG 置 1。如果开启时间戳中断，则触发时间戳中断处理。

当 TSFLG 标志位置 1 时，此时又有一个时间戳事件，将导致时间戳溢出，标志位 TSOVRFLG 置 1。如果在清除 TSFLG 标志后，又立马检测到时间戳事件，TSFLG、TSOVRFLG 标志都会置 1。

19.5.9 侵入检测

侵入检测是一种防止侵入造成数据泄露的数据自毁型保护装置，通过硬件电路设计，把侵入检测信号传送到侵入检测引脚。

侵入检测有多个侵入检测引脚，每个引脚由单独寄存器位使能。为更好地检测到真实的侵入事件，信号滤波可配置，同时每个引脚都可配置侵入检测极性。

侵入检测极性

通过寄存器 RTC_TACFG 中的 TPxALCFG 位，可选取低电平/上升沿、高电平/下降沿作为侵入检测极性。

侵入信号滤波器

寄存器 RTC_TACFG 中的 TPSFSEL 位用于配置侵入检测的采样频率，

RTC_TACFG 中的 TPFCSSEL 位用于配置连续检测多少个有效侵入信号，才产生侵入事件。

特别地，如果在使能侵入检测引脚前，侵入检测引脚上已产生过侵入信号，此时使能侵入检测引脚会立即产生一个侵入事件。

侵入时间戳

某些时刻，为更好地记录侵入检测事件，RTC 能够把当前侵入时间戳锁存，通过寄存器 RTC_TACFG 中的 TPTSEN 位即可快速使能该功能，无需额外使能时间戳功能。

19.5.10 RTC 输出

RTC 输出通过 PC13 引脚，把内部的 RTC 校准时钟、闹钟信号输送给外部。

RTC 校准时钟

校准时钟输出一般用于观察 RTC 时钟源的精度，观测值用于时钟源的校准。通过 RTC_CTRL 寄存器的 CALOSEL 位选取 512Hz、1Hz 信号输出源，RTC_CTRL 寄存器的 CALOEN 位使能校准输出。

闹钟信号

当闹钟运行的情况下，闹钟事件可作为脉冲信号对外输出。RTC_CTRL 寄存器的 OUTSEL 位选取信号输出源，POLCFG 位配置输出极性。

19.6 寄存器地址映射

表格 62 RTC 寄存器地址映射

寄存器名	描述	偏移地址
RTC_TIME	RTC 时间寄存器	0x00
RTC_DATE	RTC 日期寄存器	0x04
RTC_CTRL	RTC 控制寄存器	0x08
RTC_STS	RTC 状态寄存器	0x0C
RTC_PSC	RTC 预分频器寄存器	0x10
RTC_ALRMA	RTC alarm A 寄存器	0x1C
RTC_WRPROT	RTC 写保护寄存器	0x24
RTC_SUBSEC	RTC 亚秒寄存器	0x28
RTC_SHIFT	RTC 移位寄存器	0x2C
RTC_TSTIME	RTC 时间戳时间寄存器	0x30
RTC_TSDATE	RTC 时间戳日期寄存器	0x34
RTC_TSSUBSEC	RTC 时间戳亚秒寄存器	0x38

寄存器名	描述	偏移地址
RTC_CAL	RTC 校准寄存器	0x3C
RTC_TACFG	RTC 侵入和复用配置寄存器	0x40
RTC_ALRMAS	RTC alarm A 亚秒寄存器	0x44
RTC_BAKPx	RTC 备份寄存器	0x50-0x60

19.7 寄存器功能描述

19.7.1 RTC 时间寄存器(RTC_TIME)

RTC_TIME 是日期时间影子寄存器，该寄存器只能在初始化模式下写入，处于写保护状态。

偏移地址：0x00

上电复位值：0x0000 0000

系统复位：0xXXXX XXXX

位/域	名称	R/W	描述
3:0	SECU	R/W	设置秒个位的值以 BCD 格式存储 (Second Ones Unit in BCD Format Setup)
6:4	SECT	R/W	设置秒十位的值以 BCD 格式存储 (Second Ten's Place in BCD Format Setup)
7	保留		
11:8	MINU	R/W	设置分个位的值以 BCD 格式存储 (Minute Ones Unit in BCD Format Setup)
14:12	MINT	R/W	设置分十位的值以 BCD 格式存储 (Minute Ten's Place in BCD Format Setup)
15	保留		
19:16	HRU	R/W	设置小时个位的值以 BCD 格式存储 (Hour Ones Unit in BCD Format Setup)
21:20	HRT	R/W	设置小时十位的值以 BCD 格式存储 (Hour Ten's Place in BCD Format Setup)
22	TIMEFCFG	R/W	配置时间格式 (Time Format Configure) 0: AM 或 24 小时制 1: PM
31:23	保留		

19.7.2 RTC 日期寄存器(RTC_DATE)

RTC_DATE 是日期日期影子寄存器，该寄存器只能在初始化模式下写入，处于写保护状态。

偏移地址：0x04

复位值：0x0000 2101

位/域	名称	R/W	描述
3:0	DAYU	R/W	设置日个位的值以 BCD 格式存储 (Day Ones Unit in BCD Format Setup)
5:4	DAYT	R/W	设置日十位的值以 BCD 格式存储 (Day Ten's Place in BCD Format Setup)

位/域	名称	R/W	描述
7:6	保留		
11:8	MONU	R/W	设置月个位的值以 BCD 格式存储 (Month Ones Unit in BCD Format Setup)
12	MONT	R/W	设置月十位的值以 BCD 格式存储 (Month Ten's Place in BCD Format Setup)
15:13	WEEKSEL	R/W	选择日期单位 (Week Day Units Select) 000: 禁用 001: 星期一 ... 111: 星期日
19:16	YRU	R/W	设置年个位的值以 BCD 格式存储 (Year Ones Unit in BCD Format Setup)
23:20	YRT	R/W	设置年十位的值以 BCD 格式存储 (Year Ten's Place in BCD Format Setup)
31:24	保留		

19.7.3 RTC 控制寄存器(RTC_CTRL)

- (1) 该寄存器的位 7, 6 和 4 只能在初始化模式下写入。
- (2) 不建议在日期的小时数在增加的时候去改写此寄存器, 因为可能会屏蔽正确的小时增量。
- (3) 改写 STCCFG 和 WTCCFG 的值将在下一秒生效。
- (4) 该寄存器处于写保护下。

偏移地址: 0x08

上电复位值: 0x0000 0000

系统复位: 0xXXXX XXXX

位/域	名称	R/W	描述
2:0	保留		
3	TSETECFG	R/W	配置时间戳事件触发边沿 (Time Stamp Event Trigger Edge Configure) 该位表示 RTC_TS 是在上升沿还是下降沿生成一个时间戳事件。 0: 上升沿 1: 下降沿 该位要在 TSEN=0 时改变。
4	RCLKDEN	R/W	使能 RTC_REFIN 参考时钟检测 (RTC_REFIN reference clock detection enable) 0: 禁止 1: 使能 SPSC 必须为 0x00FF
5	RCMCFG	R/W	配置读取日期值模式 (Read Calendar Value Mode Configure) 0: 从影子寄存器读取日期值, 影子寄存器每两个 RTCCLK 周期更新一次 1: 从日期寄存器读取日期值 如果 APB1 时钟频率低于 RTCCLK 频率的 7 倍, RCMCFG 必须置 1。
6	TIMEFCFG	R/W	配置时间格式 (Time Format Configure) 0: 24 小时/天格式 1: AM/PM 时间格式
7	保留		

位/域	名称	R/W	描述
8	ALREN	R/W	使能报警功能 (Alarm A Function Enable) 0: 禁止 1: 使能
10:9	保留		
11	TSEN	R/W	使能时间戳 (Time Stamp Enable) 0: 禁止 1: 使能
12	ALRIEN	R/W	使能报警中断 (Alarm A Interrupt Enable) 0: 禁止 1: 使能
14:13	保留		
15	TSIEN	R/W	使能时间戳中断 (Time Stamp Interrupt Enable) 0: 禁止 1: 使能
16	STCCFG	R/W	配置夏季时间变化 (Summer Time Change Configure) 读过程中该位会一直保持为 0; 如果在初始化模式以外设置该位, 日期时间加 1。 0: 无效 1: 当前时间增加 1 小时, 用于校准夏季时间变化
17	WTCCFG	R/W	配置冬季时间变化 (Winter Time Change Configure) 读过程中该位会一直保持为 0; 如果在初始化模式以外设置该位, RCT_TIME 寄存器的 HRx 为 0 时, 该位无效, HRx 不为 0 时, 日期时间减 1。 0: 无效 1: 当前时间减少 1 小时, 用于校准冬季时间变化
18	BAKP	R/W	设置备份值 (Backup Value Setup) 该位表示夏季时间是否已经发生变化, 由用户写入。
19	CALOSEL	R/W	选择校准输出值 (Calibration Output Value Select) CALOEN=1 时, 该位用于选择 RTC_CALIB 的输出信号。 0: 512Hz 1: 1Hz 上述频率在 RTCCLK 为 32.768kHz 和预分频器处于默认值(APSC=127, SPSC=255)时有效。
20	POLCFG	R/W	配置输出极性 (Output Polarity Configure) 该位表示 ALRAFLG 位置 1 时 (取决于 OUTSEL 位), 引脚的电平状态。 0: 高电平 1: 低电平
22:21	OUTSEL	R/W	选择输出方式 (Output Way Select) 该位用于选择 RTC_ALARM 输出关联的标志位 00: 输出禁用 01: 闹钟 A 输出启用 10: 保留 11: 保留
23	CALOEN	R/W	使能校准输出 (Calibration Output Enable) 该位使能 RTC_CAL 输出

位/域	名称	R/W	描述
			0: 禁止 1: 使能
31:24	保留		

19.7.4 RTC 状态寄存器(RTC_STS)

该寄存器（除 RTC_STS[13:8]位外）处于写保护状态。

偏移地址：0x0C

上电复位值：0x0000 0007

系统复位：0xXXXX XXXX

位/域	名称	R/W	描述
0	ALRWFLG	R	发生闹钟 A 写入标志（Alarm A Write Occur Flag） 当 RTC_CTRL 的 ALREN=0 后，闹钟 A 的值发生变化，由硬件置 1；在初始化模式下由硬件清除。 0: 不可以更新闹钟 A 1: 可以更新闹钟 A
2:1	保留		
3	SOPFLG	R	发生移位操作挂起标志（Shift Operation Pending Occur Flag） 0: 未发生 1: 发生 当通过向 RTC_SHIFT 寄存器写入产生一个移位操作时，该位立即由硬件置 1。当相应的移位操作执行完毕后，该位由软件清 0。对 SOPFLG 写入无效。
4	INITSFLG	R	发生日期初始化状态标志（Initialization State Occur Flag） 当日期中“年”字段不为“0”时，该位由硬件置位。 0: 未发生 1: 发生
5	RSFLG	RC_WO	发生日期影子寄存器同步标志（Registers Synchronization Occur Flag） 当日期寄存器中的内容复制到影子寄存器(RTC_SUBSEC、RTC_TIME 和 RTC_DATE)中时，该位由硬件置 1；当移位操作被挂起(SOPFLG=1)或处于忽略影子寄存器模式（RCMCFG=1)时，在初始化模式下该位由硬件清 0；该位也可由软件清除。 在初始化模式下，该位由硬件/软件清除。 0: 未同步 1: 同步
6	RINITFLG	R	发生日期寄存器初始化标志（Register Initialization Occur Flag） 该位置“1”，RTC 处在初始化状态，时间、日期和预分频器寄存器可被更新。 0: 不可以初始化 1: 初始化
7	INITEN	R/W	使能初始化模式（Initialization Mode Enable） 0: 自由运行模式 1: 初始化模式，可以用来编程 RTC_TIME 和 RTC_DATE，以及 RTC_PSC。计数器停止计数，一直到 INITEN 复位之后，计数器从新值开始计数。

位/域	名称	R/W	描述
8	ALRAFLG	RC_W0	发生闹钟 A 匹配标志 (Alarm A Match Occur Flag) 当 RTC_TIME 和 RTC_DATE 与闹钟 A 寄存器 RTC_ALRMA 匹配时, 该标志由硬件置位。 该标志由软件写 0 清除。
10:9	保留		
11	TSFLG	RC_W0	发生时间戳标志 (Time Stamp Occur Flag) 当产生时间戳事件时该标志由硬件置 1; 由软件写 0 清除。
12	TSOVRFLG	RC_W0	发生时间戳溢出标志 (Time Stamp Overflow Occur Flag) 在 TSFLG=1 且产生时间戳事件时该标志位由硬件置 1; 由软件写 0 清除。 该位建议在 TSFLG 标志位清除后再清除。
13	TP1FLG	RC_W0	发生 RTC_TP1FLG 检测标志 (RTC_TP1FLG Detection Occur Flag) 在 RTC_TP1FLG 输入检测到侵入事件时该标志由硬件置 1; 由软件写 0 清除。
15:14	保留		
16	RCALPFLG	R	发生重新校准挂起标志 (Recalibration Pending Occur Flag) 当软件向 RTC_CAL 做写操作时该位自动置 1, 此时 RTC_CAL 寄存器被锁定。 当有其他新的校准设置执行时, 该位返回 0。
31:17	保留		

19.7.5 RTC 预分频寄存器(RTC_PSC)

该寄存器只能在初始化模式下写入, 初始化必须由两次独立的写访问完成, 处于写保护状态。

偏移地址: 0x10

上电复位值: 0x007F 00FF

系统复位: 0xXXXX XXXX

位/域	名称	R/W	描述
14:0	SPSC	R/W	同步预分频器系数 (Synchronous Prescaler Coefficient) $ck_spre \text{ 频率} = ck_apre \text{ 频率} / (SPSC + 1)$
15	保留		
22:16	APSC	R/W	异步预分频器系数 (Asynchronous Prescaler Coefficient) $ck_apre \text{ 频率} = RTCCLK \text{ 频率} / (APSC + 1)$
31:23	保留		

19.7.6 RTC 闹钟 A 寄存器 (RTC_ALRMA)

该寄存器只能在 RTC_STS 的 ALRWFLG 置 1 或初始化模式下写入, 处于写保护状态。

偏移地址: 0x1C

上电复位值: 0x0000 0000

系统复位: 0xXXXX XXXX

位/域	名称	R/W	描述
3:0	SECU	R/W	设置秒个位的值以 BCD 格式存储 (Second Ones Unit in BCD Format Setup)
6:4	SECT	R/W	设置秒十位的值以 BCD 格式存储 (Second Ten's Place in BCD Format Setup)
7	SECMEN	R/W	使能 Alarm A “秒” 屏蔽 (Alarm A Seconds Mask Enable) 0: 若 “秒” 匹配, 置位 Alarm A 1: 屏蔽 “秒” 的值对 Alarm A 的影响
11:8	MINU	R/W	设置分个位的值以 BCD 格式存储 (Minute Ones Unit in BCD Format Setup)
14:12	MINT	R/W	设置分十位的值以 BCD 格式存储 (Minute Ten's Place in BCD Format Setup)
15	MINMEN	R/W	使能 Alarm A “分” 屏蔽 (Alarm A Minutes Mask Enable) 0: 若 “分” 匹配, 置位 Alarm A 1: 屏蔽 “分” 的值对 Alarm A 的影响
19:16	HRU	R/W	设置小时个位的值以 BCD 格式存储 (Hour Ones Unit in BCD Format Setup)
21:20	HRT	R/W	设置小时十位的值以 BCD 格式存储 (Hour Ten's Place in BCD Format Setup)
22	TIMEFCFG	R/W	配置时间格式 (Time Format Configure) 0: AM 或 24 小时制 1: PM
23	HRMEN	R/W	使能 Alarm A “时” 屏蔽 (Alarm A Hours Mask Enable) 0: 若 “小时” 匹配, 置位 Alarm A 1: 屏蔽 “小时” 的值对 Alarm A 的影响
27:24	DAYU	R/W	设置日个位的值以 BCD 格式存储 (Day Ones Unit in BCD Format Setup)
29:28	DAYT	R/W	设置日十位的值以 BCD 格式存储 (Day Ten's Place in BCD Format Setup)
30	WEEKSEL	R/W	选择日期 (Week Day Select) 0: DAYU 表示日期 1: DAYU 表示星期的数。DAYT 无作用。
31	DATEMEN	R/W	使能 Alarm A 日期屏蔽 (Alarm A Date Mask Enable) 0: 若日期/星期匹配, 置位 Alarm A 1: 屏蔽日期/星期的值对 Alarm A 的影响

19.7.7 RTC 写保护寄存器(RTC_WRPROT)

偏移地址: 0x24

复位值: 0x0000 0000

位/域	名称	R/W	描述
7:0	KEY	W	设置写保护键值 (Write Protection Key Value Setup) 此字节由软件编写, 读取此字节始终为 0x00。
31:8	保留		

19.7.8 RTC 亚秒寄存器(RTC_SUBSEC)

偏移地址: 0x28

复位值: 0x0000 0000

位/域	名称	R/W	描述
15:0	SUBSEC	R	设置亚秒值 (Sub Second Value Setup) SUBSEC 是同步预分频器计数器中的值。由下列公式决定： 亚秒值=(SPSC-SUBSEC)/(SPSC+1) 当一个移位操作执行完毕后，SUBSEC 可能大于 SPSC。那么正确的时间/日期比 RTC_TIME/RTC_DATE 少一秒。
31:16	保留		

19.7.9 RTC 移位寄存器(RTC_SHIFT)

该寄存器处于写保护状态。

偏移地址：0x2C

复位值：0x0000 0000

位/域	名称	R/W	描述
14:0	SFSEC	W	设置减少若干分之一秒 (Subtract a Fraction of a Second Setup) 此位域只能写入，读取此字节始终为 0。当正在执行一个操作时，对此位进行写操作无效。 设置的 SFSEC 值将会被添加到同步预分频器计数器中。如果计数器是倒计时的，时钟会被延迟，延迟时间由以下公式决定： Delay (seconds)=SFSEC/(SPSC+1) 当与 ADD1SECEN 同时作用时，推进时钟将增加若干分之一秒，具体的增加值由以下公式决定： Advance(seconds)=(1-(SFSEC/(SPSC+1))) 对此位写操作可清除 RSFLG 位。软件持续运行直到 RSFLG 置 1，以此确保影子寄存器的值与移位时间同步。
30:15	保留		
31	ADD1SECEN	W	使能增加一秒 (Add One Second Enable) 0: 不增加 1: 时钟/日期增加一秒 此位只能写入，读取此字节始终为 0。当正在执行一个操作时，对此位进行写操作无效。 当与 SFSEC 同时作用时，可以增加时钟的值，增加值为若干分之一秒。

19.7.10 RTC 时间戳时间寄存器(RTC_TSTIME)

当 RTC_STS 中 TSFLG 置 1 时，该寄存器才有效。当 TSFLG 位复位时，清除该寄存器的内容。

偏移地址：0x30

上电复位值：0x0000 0000

系统复位：0xXXXX XXXX

位/域	名称	R/W	描述
3:0	SECU	R	设置秒个位的值以 BCD 格式存储 (Second Ones Unit in BCD Format Setup)
6:4	SECT	R	设置秒十位的值以 BCD 格式存储 (Second Ten's Place in BCD Format Setup)
7	保留		

位/域	名称	R/W	描述
11:8	MINU	R	设置分个位的值以 BCD 格式存储 (Minute Ones Unit in BCD Format Setup)
14:12	MINT	R	设置分十位的值以 BCD 格式存储 (Minute Ten's Place in BCD Format Setup)
15	保留		
19:16	HRU	R	设置小时个位的值以 BCD 格式存储 (Hour Ones Unit in BCD Format Setup)
21:20	HRT	R	设置小时十位的值以 BCD 格式存储 (Hour Ten's Place in BCD Format Setup)
22	TIMEFCFG	R	配置时间格式 (Time Format Configure) 0: AM 或 24 小时制 1: PM
31:23	保留		

19.7.11 RTC 时间戳日期寄存器(RTC_TSDATE)

当 RTC_STS 的 TSFLG 位置 1 时, 该寄存器才有效。当 TSFLG 位复位时清除该寄存器。

偏移地址: 0x34

上电复位值: 0x0000 0000

系统复位: 0xXXXX XXXX

位/域	名称	R/W	描述
3:0	DAYU	R	设置日个位的值以 BCD 格式存储 (Day Ones Unit in BCD Format Setup)
5:4	DAYT	R	设置日十位的值以 BCD 格式存储 (Day Ten's Place in BCD Format Setup)
7:6	保留		
11:8	MONU	R	设置月个位的值以 BCD 格式存储 (Month Ones Unit in BCD Format Setup)
12	MONT	R	设置月十位的值以 BCD 格式存储 (Month Ten's Place in BCD Format Setup)
15:13	WEEKSEL	R	选择日期单位 (Week Day Units Select) 000: 禁用 001: 星期一 ... 111: 星期日
31:16	保留		

19.7.12 RTC 时间戳亚秒寄存器(RTC_TSSUBSEC)

当 RTC_STS 寄存器的 TSFLG 位置 1 时, 该寄存器才有效。当 TSFLG 位复位时清除该寄存器的内容。

偏移地址: 0x38

上电复位值: 0x0000 0000

系统复位: 0xXXXX XXXX

位/域	名称	R/W	描述
15:0	SUBSEC	R	设置亚秒值 (Sub Second Value Setup) 当发生时间戳事件时, SUBSEC[15:0]是同步预分频器计数器中的值。
31:16	保留		

19.7.13 RTC 校准寄存器(RTC_CAL)

该寄存器处于写保护状态。

偏移地址: 0x3C

上电复位值: 0x0000 0000

系统复位: 0xXXXX XXXX

位/域	名称	R/W	描述
8:0	RECALF	R/W	减少校准频率 (Reduced Calibration Frequency) 减少日期频率: 在 2^{20} 个 RTCCLK 脉冲内(若输出频率为 32768 Hz, 则 32 秒)屏蔽 RECALF 个脉冲将减少日期的频率 (分辨率为 0.9537 ppm)。 增加日期频率: 与 ICALFEN 同时作用
12:9	保留		
13	CAL16CFG	R/W	配置 16 秒校准周期 (16 Second Calibration Cycle Period Configure) 当 CAL16CFG 置 1, 使用 16 秒校准周期, 不能和 CAL8CFG 位同时置 1。 当 CAL16CFG=1 时, RECALF[0]始终为 0。
14	CAL8CFG	R/W	配置 8 秒校准周期 (8 Second Calibration Cycle Period Configure) 当 CAL8CFG 置 1, 使用 8 秒校准周期, 不能和 CAL16CFG 位同时置 1。 当 CAL8CFG=1 时, RECALF[1:0]始终为 00。
15	ICALFEN	R/W	使能增加校准频率 (Increase Calibration Frequency Enable) 0: 不增加 RTCCLK 脉冲 1: 每 2^{11} 个脉冲增加一个 RTCCLK 脉冲 (频率增加 488.5 ppm) 与 RECALF 同时作用, 使用高分辨率时会降低日期的频率。若输入频率为 32768Hz,则在 32 秒窗口中增加 RTCCLK 脉冲的数量由下列公式决定: (512*ICALFEN) - RECALF。
31:16	保留		

19.7.14 RTC 侵入和复用配置寄存器(RTC_TACFG)

偏移地址: 0x40

上电复位值: 0x0000 0000

系统复位: 0xXXXX XXXX

位/域	名称	R/W	描述
0	TP1EN	R/W	使能 RTC_TAMP1 输入检测 (RTC_TAMP1 Input Detection Enable) 0: 禁止 1: 使能
1	TP1ALCFG	R/W	配置 RTC_TAMP1 输入的有效电平 (RTC_TAMP1 Input Active Level Configure) 当 TPFCSSEL!=00 时, 该位决定 RTC_TAMP1 输入在保持高/低电平会触发一个侵入检测事件。 0: 低电平 1: 高电平

位/域	名称	R/W	描述
			当 TPFCSSEL=00 时，该位决定 RTC_TAMP1 输入在上升/下降沿时触发一个侵入检测事件。 0: 上升沿 1: 下降沿
2	TPIEN	R/W	使能侵入中断 (Tamper Interrupt Enable) 0: 禁止 1: 使能
6:3	保留		
7	TPTSEN	R/W	使能侵入检测事件的时间戳 (Tamper Detection Event Timestamp Enable) 该位决定侵入检测事件产生的时间戳是否保存 0: 不保存 1: 保存 该位在 RTC_CTRL 寄存器 TSEN=0 时依然有效。
10:8	TPSFSEL	R/W	选择侵入采样频率 (Tamper Sampling Frequency Select) 这些位决定每个 RTC_TAMPx 输入的采样频率。 0x0: RTCCLK/32768 0x1: RTCCLK/16384 0x2: RTCCLK/8192 0x3: RTCCLK/4096 0x4: RTCCLK/2048 0x5: RTCCLK/1024 0x6: RTCCLK/512 0x7: RTCCLK/256
12:11	TPFCSSEL	R/W	选择 RTC_TAMPx 过滤器计数 (RTC_TAMPx Filter Count Select) 这些位决定在特定电平 (TAMP*TRG) 上连续几次采样后激活入侵事件。 TPFCSSEL 对每次 RTC_TAMPx 输入都有效。 0x0: 在 RTC_TAMPx 输入转换为有效电平的边沿激活入侵事件 0x1: 连续 2 次采样 0x2: 连续 4 次采样 0x3: 连续 8 次采样
14:13	TPPRDUSEL	R/W	选择 RTC_TAMPx 预充电时长 (RTC_TAMPx Precharge Duration Select) 这些位决定采样前上拉电阻启用几个 RTCCLK 周期；对每次 RTC_TAMPx 输入都有效。 0x0: 1 0x1: 2 0x2: 4 0x3: 8
15	TPPUDIS	R/W	禁止 RTC_TAMPx 上拉功能 (RTC_TAMPx Pull-up Function Disable) 该位决定是否所有 RTC_TAMPx 引脚在采样前进行预充电。 0: 使能(使能内部上拉) 1: 禁止
17:16	保留		
18	PC13VAL	R/W	配置 RTC_ALARM 输出形式/PC13 值 (RTC_ALARM Output Type/PC13 Value Configure)

位/域	名称	R/W	描述
			当 PC13 用于输出 RTC_ALARM 时,该位决定 RTC_ALARM 的输出模式: 0: 开漏输出 1: 推挽输出 当禁用所有 RTC 复用功能且 PC13EN=1 时,该位设置 PC13 输出值。
19	PC13EN	R/W	使能 PC13 模式 (PC13 Mode Enable) 0: PC13 由 GPIO 配置寄存器控制, 在待机模式下, PC13 浮空。 1: 当禁用 RTC 复用功能时, PC13 强制设置为推挽输出模式。
20	PC14VAL	R/W	设置 PC14 输出值 (PC14 Output Value Setup) 禁用 LSECLK 且 PC14EN=1, 该位设置 PC14 的输出值。
21	PC14EN	R/W	使能 PC14 模式 (PC14 Mode Enable) 0: PC14 由 GPIO 配置寄存器控制, 在待机模式下, PC14 浮空。 1: 当禁用 LSECLK 时, PC14 强制设置为推挽输出模式
22	PC15VAL	R/W	设置 PC15 输出值 (PC15 Output Value Setup) 禁用 LSECLK 且 PC15EN=1,该位设置 PC15 的输出值。
23	PC15EN	R/W	使能 PC15 模式 (PC15 Mode Enable) 0: PC15 由 GPIO 配置寄存器控制, 在待机模式下, PC15 浮空。 1: 当禁用 LSECLK 时, PC15 强制设置为推挽输出模式。
31:24	保留		

19.7.15 RTC 闹钟 A 亚秒寄存器(RTC_ALRMASS)

仅当 RTC_CTRL 寄存器的 ALREN 复位时或在初始化模式下, 该寄存器才可被写入。

该寄存器处在写保护状态。

偏移地址: 0x44

上电复位值: 0x0000 0000

系统复位: 0xXXXX XXXX

位/域	名称	R/W	描述
14:0	SUBSEC	R/W	设置亚秒值 (Sub Second Value Setup) 亚秒值与同步预分频器计数器中的值进行比较来确定是否激活闹钟 A, 仅比较位 0 到 MASKSEL-1。
23:15	保留		
27:24	MASKSEL	R/W	选择屏蔽从此位开始的最有效位 (Mask the Most-significant Bits Starting at This Bit Select) 0x0: 不对闹钟 A 进行比较。闹钟在秒单元增加 1 时设置 0x1: 在对闹钟 A 比较时, SUBSEC[14:1]不参与, 仅 SUBSEC[0]参与 0x2: 在对闹钟 A 比较时, SUBSEC[14:2]不参与, 仅 SUBSEC[1:0]参与 0x3: 在对闹钟 A 比较时, SUBSEC[14:3]不参与, 仅 SUBSEC[2:0]参与 ... 0xC: 在对闹钟 A 比较时, SUBSEC[14:12]不参与, 仅 SUBSEC[11:0]参与 0xD: 在对闹钟 A 比较时, SUBSEC[14:13]不参与, 仅 SUBSEC[12:0]参与 0xE: 在对闹钟 A 比较时, SUBSEC[14]不参与, 仅 SUBSEC[13:0] 参与 0xF: 在对闹钟 A 比较时, 15 个 SUBSEC 位都要参与, 只有全部匹配才能激活闹钟。

位/域	名称	R/W	描述
			同步计数器溢出位(位 15)从不进行比较。仅在移位操作后，该位不为 0。
31:28	保留		

19.7.16 RTC 备份寄存器 (RTC_BAKPx) (x=0-4)

偏移地址: 0x50-0x60

上电复位值: 0x0000 0000

复位值: 0xXXXX XXXX

位/域	名称	R/W	描述
31:0	BAKP	R/W	设置备份值 (Backup Value Setup) 切断 V _{DD} 电源将掉电，寄存器值无法保持；当发生侵入检测事件或禁用闪存读出保护时，此寄存器复位。 即使设备在低功耗模式下运行此位域的内容也有效。

20 控制器局域网（CAN）

20.1 术语全称、缩写描述

表格 63 术语全称、缩写描述

中文全称	英文全称	英文缩写
先进先出队列	First Input First Output	FIFO
请求	Request	REQ

20.2 简介

CAN 是控制器局域网络 Controller Area Network 的缩写，是 ISO 国际化的串行通信协议，支持 CAN 协议 2.0A 和 2.0B。在 CAN 协议中，发送者以广播形式把报文发送给所有接收者，节点在接收报文时，会经过过滤器组根据标识符决定是否需要该报文，这种设计节省了 CPU 的开销。

20.3 主要特征

- (1) 支持 CAN 协议 2.0A 和 2.0B
- (2) 通信波特率最大为 1Mbit/s
- (3) 发送功能
 - 有 3 个发送邮箱
 - 发送报文优先级可配置
 - 可记录发送时间
- (4) 接收功能
 - 有 2 个 3 级深度的接收 FIFO
 - 有 14 个过滤器组.
 - 可记录接收时间

20.4 功能描述

20.4.1 CAN 物理层特性

CAN 总线上可以有多个通讯节点，每个节点由一个 CAN 的控制器和收发器组成，控制器和收发器之间通过 CAN_TX 和 CAN_RX 连接传输逻辑信号，收发器和总线之间通过 CAN_High 和 CAN_Low 连接传输差分信号。

20.4.2 报文结构

图 68 标准数据帧

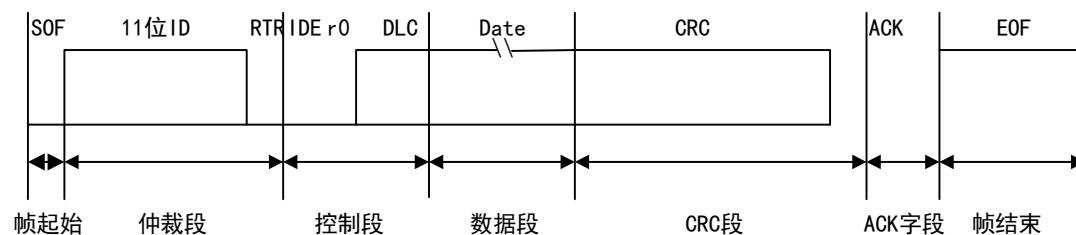
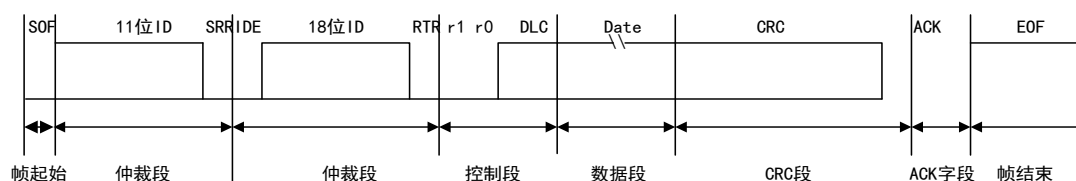


图 69 扩展数据帧



注意:

- (1) 帧起始：用来通知各个节点将有数据传输。
- (2) 仲裁段：有多个报文被发送时，用来决定哪个报文能被传输。此段主要内容是 ID 信息，标准格式下 ID 为 11 位，扩展格式下 ID 为 29 位。
- (3) 控制段：此段最主要的是数据长度码（DLC），用来表示报文中的数据段有多少个字节。数据段最多 8 个字节。
- (4) 数据段：包含节点要发送的数据信息。
- (5) CRC 段：CRC 校验码用来确保报文正确传输。
- (6) ACK 段：此段包含 ACK 槽位和 ACK 界定符位，ACK 槽中发送节点发送隐性位，接收节点将在这一位发送显性位表示应答。
- (7) 帧结束：由发送节点发送的 7 个隐性位表示结束。

20.4.3 工作模式

CAN 有三个主要的工作模式：初始化模式、正常模式和睡眠模式。

20.4.3.1 初始化模式

配置寄存器 CAN_MCTRL 的 INITREQ 位置 1 请求进入初始化模式，INITFLG 位清 0 确认进入初始化模式。

配置寄存器 CAN_MCTRL 的 INITREQ 位清 0 请求退出初始化模式，INITFLG 位清 0 确认退出初始化模式。

初始化模式下，禁止报文接收和发送。

20.4.3.2 正常模式

通过软件配置寄存器 CAN_MCTRL 的 INITREQ 位清 0 请求从初始化模式进入正

常模式，等待硬件对 INITFLG 位清 0 确认进入正常模式。

正常模式下，可以正常接收和发送报文。

20.4.3.3 睡眠模式

配置寄存器 CAN_MCTRL 的 SLEEPREQ 位置 1 请求进入睡眠模式。

睡眠模式下 CAN 的时钟停止工作，软件可以正常访问邮箱寄存器，此时 CAN 处于低功耗状态。

20.4.4 通信模式

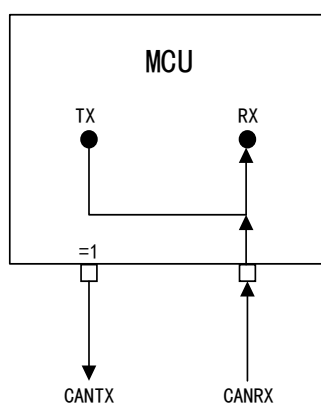
有四种通信模式分别是：静默模式、环回模式、静默环回模式和正常模式。只能在初始化模式下选择不同的通信模式。

20.4.4.1 静默模式

配置寄存器 CAN_BITTIM 的 SILMEN 位置 1，选择静默模式。

该模式下，只能向总线发送隐性位（逻辑 1），不能发送显性位（逻辑 0），可以从总线接收数据。

图 70 CAN 工作在静默模式

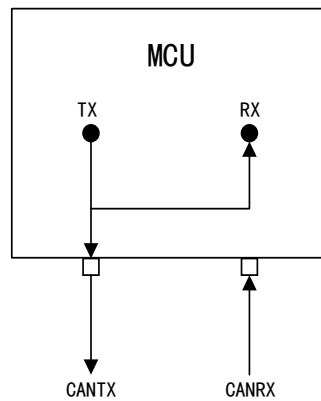


20.4.4.2 环回模式

配置寄存器 CAN_BITTIM 的 LBKMEN 位置 1，选择环回模式。

该模式下，发送的数据直接传到输入端接收，不从总线上接收数据，可以向总线发送所有数据。

图 71 CAN 工作在环回模式

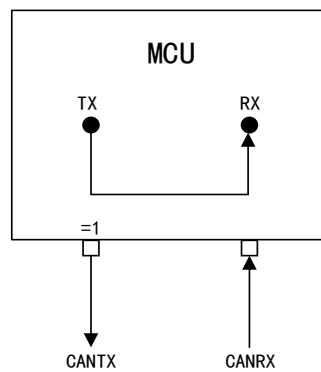


20.4.4.3 环回静默模式

配置寄存器 CAN_BITTIM 的 LBKMEN 和 SILMEN 位同时置 1，选择环回静默模式。

该模式下，发送的数据直接传到输入端接收，不从总线上接收数据，只能向总线发送隐性位（逻辑 1），不能发送显性位（逻辑 0）。

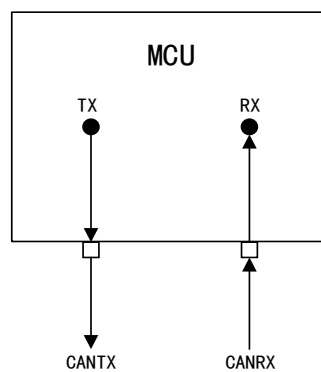
图 72 CAN 工作在静默环回模式



20.4.4.4 正常模式

该模式下，可以向总线发送数据和接收数据。

图 73 CAN 工作在正常模式



20.4.5 数据发送

20.4.5.1 发送邮箱状态转换

发送邮箱状态转换过程：

- (1) 先选择空置邮箱进行设置，通过配置寄存器 CAN_TXMIDx 的 TXMREQ 位置 1 向 CAN 总线控制器提交发送请求，然后邮箱马上进入挂号状态。
- (2) 多个邮箱在挂号状态情况下，进行优先级调度，当邮箱成为最高优先级时，进入预定状态。
- (3) 当发送邮箱中的报文向总线发送时，进入发送状态。
- (4) 报文发送成功后，邮箱重新变成空置状态。

20.4.5.2 发送优先级

当有多个报文同时等待发送时，通过配置寄存器 CAN_MCTRL 的 TXFPCFG 位决定发送顺序：

- 当 TXFPCFG 位置 0 时，优先级由报文标识符决定，标识符最小，优先级最高，标识符相等，则邮箱号小的先被发送
- 当 TXFPCFG 位置 1 时，优先级由发送请求的顺序来决定

20.4.5.3 中止

通过配置寄存器 CAN_TXSTS 的 ABREQFLG 位置 1，发送中止请求。

若邮箱处于挂号或预定状态，立马中止发送请求；若邮箱处于发送状态，两种状况：一种是邮箱发送成功，则邮箱变为空置状态，此时，CAN_TXSTS 寄存器的 TXSUSFLG 位由硬件置 1；另一种是邮箱发送失败，则邮箱变为预定状态，发送请求被中止。

20.4.5.4 禁止自动重传

一般在时间触发通信模式下，需要禁止自动重传。

在禁止自动重传模式下，报文只被发送一次，不管发送的结果如何（成功、出错或仲裁丢失），硬件不会再自动发送报文。

当发送过程结束后，CAN_TXSTS 寄存器的 REQCFGL 位置 1，发送的结果会在 TXSUSFLG、ARBLSTFLG 和 TXERRFLG 位上

20.4.6 数据接收

20.4.6.1 接收 FIFO

CAN 一共有两个接收 FIFO，每个 FIFO 有三个邮箱，寄存器 CAN_RXF 的 FMNUM[1:0]位反映当前存放的报文数目；RFOM 位置 1 来释放接收 FIFO 的输出邮箱；FFULLFLG 为满状态标志位；FOVRFLG 为溢出状态标志位。

20.4.6.2 接收 FIFO 状态转换

一开始 FIFO 处于空状态，接收到报文后变为挂号状态。

当 FIFO 处于挂号状态但 3 个邮箱已满时，接收下一个有效报文变为溢出状态，溢出状态的报文丢弃分两种情况：

- 若禁用 FIFO 锁定功能，则最后收到的报文被新报文覆盖
- 若启用 FIFO 锁定功能，则新收到的报文被丢弃

20.4.7 过滤机制

过滤器的作用：在接收节点根据报文标识符决定是否需要此报文，过滤后只接收需要的报文。CAN 控制器有 14 个过滤器组。

20.4.7.1 位宽

每个过滤器组都可以配置两种位宽。

图 74 1 个 32 位过滤器

ID	CAN_FiBANK1 [31:24]				CAN_FiBANK1 [23:16]				CAN_FiBANK1 [15:8]				CAN_FiBANK1 [7:0]				
映像	STDID [10:3]				STDID [2:0]		EXTID [17:13]		EXTID [12:5]				EXTID [4:0]		IDTYP ESEL	TXRFR EQ	0

图 75 2 个 16 位过滤器

ID	CAN_FiBANK1 [15:8]				CAN_FiBANK1 [7:0]				CAN_FiBANK2 [15:8]				CAN_FiBANK2 [7:0]				
映像	STDID [10:3]				STDID [2:0]	TXRF REQ	IDTYP ESEL	EXTID [17:15]		STDID [12:5]				STDID [2:0]	TXRF REQ	IDTYP ESEL	EXTID [17:15]

20.4.7.2 过滤模式

屏蔽位模式

该模式下，只需要把报文标识符的某几位作为列表形成掩码，报文 ID 需要与掩码相同才能被接收

表格 64 屏蔽位模式举例

ID	1	0	1	1	0	0	1	0
掩码	1	0	1	1	1	0	0	1
筛选的 ID	1	X	1	1	0	X	X	0

标识符列表模式

该模式下，报文 ID 的每一位都需要与过滤器标识符相同才能被接收。

表格 65 标识符列表模式举例

ID	1	1	1	0	1	0	0	1	1
ID	1	1	1	0	1	0	0	1	1
筛选的 ID	1	1	1	0	1	0	0	1	1

20.4.7.3 过滤器优先级

优先级规则如下：

- 位宽 32 位过滤器优先级高于位宽 16 位的
- 位宽相同情况下，标识符列表模式优先级高于屏蔽位模式
- 位宽和模式都相同情况下，过滤序号小的优先级高

20.4.8 位时序及波特率

20.4.8.1 位时序

APM32 的 CAN 外设位时序中包含三段：同步段（SYNC_SEG）、时间段 1（BS1）、时间段 2（BS2），采样点位于 BS1 及 BS2 段的交界处。

- 同步段（SYNC_SEG）：此段占用一个时间单元
- 时间段 1（BS1）：此段占用 1 到 16 个时间单元，它包含 CAN 标准里的 PROP_SEG 和 PHASE_SEG1
- 时间段 2（BS2）：此段占用 1 到 8 个时间单元，它代表 CAN 标准里的 PHASE_SEG2

20.4.8.2 波特率的计算

BS1 段时间： $T_{s1} = T_q * (TIMSEG1[3:0] + 1)$

BS2 段时间： $T_{s2} = T_q * (TIMSEG2[2:0] + 1)$

一个数据位的时间： $T_{1bit} = 1T_q + T_{s1} + T_{s2}$

波特率 = $1 / T_{1bit}$

$T_q = (BRPSC + 1) * T_{PCLK}$

20.4.9 出错管理

通过配置寄存器 CAN_ERRSTS 的 TXERRCNT 位发送错误计数器和寄存器 CAN_ERRSTS 的 RXERRCNT 位接收错误计数器来反映 CAN 总线的错误管理。

通过配置寄存器 CAN_INTEN 的 ERRIEN 位来控制错误状态下产生中断。

20.4.9.1 离线恢复

当 CAN 错误状态寄存器的 TXERRCNT 大于 255 时，CAN 总线控制器就进入离线状态，此时寄存器 CAN_ERRSTS 的 BOFLG 位置 1，该状态下 CAN 总线控制器不能接收和发送报文。

通过配置寄存器 CAN_MCTRL 的 ALBOFFM 位来决定离线恢复方式：

- 若 ALBOFFM 位置 1，一旦硬件检测到 128 次 11 位连续的隐性位，则自动退出离线状态；
- 若 ALBOFFM 位置 0，软件请求进入再退出初始化模式后，退出离线状态。

20.4.10 中断

产生发送中断的事件:

- 硬件将寄存器 CAN_TXSTS 的 REQCFLG0 位置 1, 发送邮箱 0 变为空置状态
- 硬件将寄存器 CAN_TXSTS 的 REQCFLG1 位置 1, 发送邮箱 1 变为空置状态
- 硬件将寄存器 CAN_TXSTS 的 REQCFLG2 位置 1, 发送邮箱 2 变为空置状态

产生 FIFO0 中断的事件:

- 硬件将寄存器 CAN_RXF0 的 FMNUM0[1:0]位配置为非 0 时, FIFO0 接收到一个新的保文
- 硬件将寄存器 CAN_RXF0 的 FFULLFLG0 位置 1, FIFO0 满
- 硬件将寄存器 CAN_RXF0 的 FOVRFLG0 位置 1, FIFO0 溢出

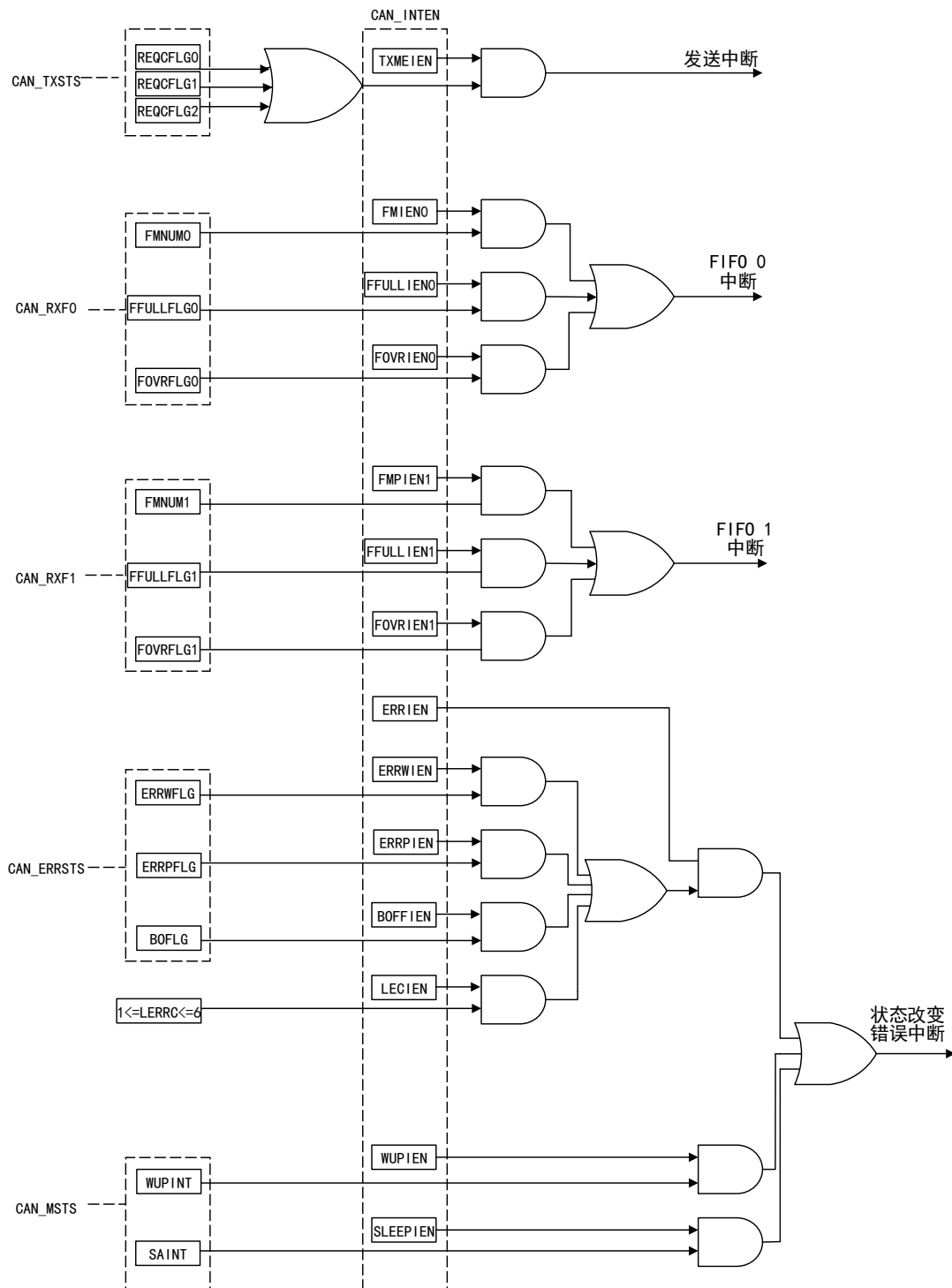
产生 FIFO1 中断的事件:

- 硬件将寄存器 CAN_RXF1 的 FMNUM1[1:0]位为非 0 时, FIFO1 接收到一个新的保文
- 硬件将寄存器 CAN_RXF1 的 FFULLFLG1 位置 1, FIFO1 满
- 硬件将寄存器 CAN_RXF1 的 FOVRFLG1 位置 1, FIFO1 溢出

产生状态改变和错误中断的事件:

- 硬件将寄存器 CAN_INTEN 的 SLEEPIEN 位置 1, 进入睡眠模式
- 硬件将寄存器 CAN_INTEN 的 WUPIEN 位置 1, 唤醒中断使能
- 硬件将寄存器 CAN_ERRSTS 的 ERRWFLG 位置 1, 表示出错次数已达到阈值
- 硬件将寄存器 CAN_ERRSTS 的 ERRPFLG 位置 1, 表示出错次数达到被动错误的阈值
- 硬件将配置寄存器 CAN_ERRSTS 的 LERRC[2:0]位, 表明上次出错的情况

图 76 事件标志和中断产生



20.5 寄存器地址映射

CAN1 基地址: 0x4000_6400

表格 66 CAN 寄存器地址映射

寄存器名	描述	偏移地址
CAN_MCTRL	CAN 主控制寄存器	0x00
CAN_MSTS	CAN 主状态寄存器	0x04
CAN_TXSTS	CAN 发送状态寄存器	0x08
CAN_RXF0	CAN 接收 FIFO 0 寄存器	0x0C
CAN_RXF1	CAN 接收 FIFO 1 寄存器	0x10
CAN_INTEN	CAN 中断使能寄存器	0x14
CAN_ERRSTS	CAN 错误状态寄存器	0x18
CAN_BITTIM	CAN 位时序寄存器	0x1C
CAN_TXMIDx	每个邮箱包含发送邮箱标识符寄存器	0x180, 0x190, 0x1A0
CAN_TXDLENx	发送邮箱数据长度寄存器	0x184, 0x194, 0x1A4
CAN_TXMDLx	发送邮箱低字节数据寄存器	0x188, 0x198, 0x1A8
CAN_TXMDHx	发送邮箱高字节数据寄存器	0x18C, 0x19C, 0x1AC
CAN_RXMIDx	接收 FIFO 邮箱标识符寄存器	0x1B0, 0x1C0
CAN_RXDLENx	接收 FIFO 邮箱数据长度寄存器	0x1B4, 0x1C4
CAN_RXMDLx	接收 FIFO 邮箱低字节数据寄存器	0x1B8, 0x1C8
CAN_RXMDHx	接收 FIFO 邮箱高字节数据寄存器	0x1BC, 0x1CC
CAN_FCTRL	CAN 过滤器主控寄存器	0x200
CAN_FMCFG	CAN 过滤器模式寄存器	0x204
CAN_FSCFG	CAN 过滤器位宽寄存器	0x20C
CAN_FFASS	CAN 过滤器 FIFO 关联寄存器	0x214
CAN_FACT	CAN 过滤器激活寄存器	0x21C
CAN_FiBANKx	CAN 过滤器组 i 的寄存器 x	0x240..0x2AC

20.6 寄存器功能描述

20.6.1 CAN 控制和状态寄存器

20.6.1.1 CAN 主控制寄存器 (CAN_MCTRL)

偏移地址: 0x00

复位值: 0x0001 0002

位/域	名称	R/W	描述
0	INITREQ	R/W	请求进入初始化模式 (Request to Enter Initialization Mode) 0: 从初始化模式进入正常工作模式 1: 从正常工作模式进入初始化模式
1	SLEEPREQ	R/W	请求进入睡眠模式 (Request to Enter Sleep Mode)

位/域	名称	R/W	描述
			0:退出睡眠模式 1:请求进入睡眠模式。 若 AWUPCFG 位置 1, 当 RX 信号检测到 CAN 报文时, 该位由硬件清零; 复位后该位置 1, 复位后处于睡眠模式。
2	TXFPCFG	R/W	配置发送 FIFO 优先级 (Transmit FIFO Priority Configure) 该位用来决定在多个报文同时等待发送时, 由哪些参数决定发送优先级。 0: 由报文的标识符来决定 1: 由发送请求的顺序来决定
3	RXFLOCK	R/W	配置接收 FIFO 锁定模式 (Receive FIFO Locked Mode Configure) 该位用来决定接收溢出时 FIFO 是否被锁定, 且当接收 FIFO 的报文未被读出时, 下一个收到的报文如何处理。 0: 未被锁定, 当接收 FIFO 的报文未被读出, 下一个收到的报文会覆盖原有的报文 1: 被锁定, 当接收 FIFO 的报文未被读出, 下一个收到的报文会被丢弃
4	ARTXMD	R/W	禁止自动重传报文 (Automatic Retransmission Message Disable) 0: 使能自动重传, 报文将一直自动重传直到发送成功 1: 禁止自动重传, 报文只被发送一次
5	AWUPCFG	R/W	配置自动唤醒模式 (Automatic Wakeup Mode Configure) 0: 软件唤醒睡眠模式, 通过清除 CAN_MCTRL 寄存器的 SMREQ 位唤醒 1: 硬件唤醒睡眠模式, 通过检测 CAN 报文唤醒
6	ALBOFFM	R/W	自动退出离线状态条件管理 (Automatic Leaving Bus-Off Status Condition Management) 0: 软件对 CAN_MCTRL 寄存器的 INITREQ 位进行置 1 随后清 0 后, 当硬件检测到 128 次 11 位连续的隐性位, 则退出离线状态 1: 当硬件检测到 128 次 11 位连续的隐性位, 则自动退出离线状态
14:7	保留		
15	SWRST	R/S	软件复位 CAN (Software Reset CAN) 0: 正常工作 1: CAN 被强行复位, 复位后 CAN 进入睡眠模式, 硬件自动对该位清 0
16	DBGFRZE	R/W	冻结调试 (Debug Freeze) 0: 无效 1: 调试时, CAN 的接收/发送无法工作, 但仍可以正常读写和控制接收 FIFO
31:17	保留		

20.6.1.2 CAN 主状态寄存器 CAN_MSTS)

偏移地址: 0x04

复位值: 0x0000 0C02

位/域	名称	R/W	描述
0	INITFLG	R	正处于初始化模式标志 (Being Initialization Mode Flag) 该位由硬件置 1 和清 0。 0: 退出初始化模式

位/域	名称	R/W	描述
			1: 处于初始化模式, 该位是对 CAN_MCTRL 寄存器初始化请求位的确认。
1	SLEEPFLG	R	正处于睡眠模式标志 (Being Sleep Mode Flag) 该位由硬件置 1 和清 0 0: 退出睡眠模式 1: 处于睡眠模式, 该位是对 CAN_MCTRL 寄存器睡眠模式请求位的确认。
2	ERRIFLG	RC_W1	发生错误中断标志 (Error Interrupt Occur Flag) 该位由硬件置 1, 软件写 1 清 0。 0: 未发生 1: 发生
3	WUPIFLG	RC_W1	发生唤醒中断标志 (Wakeup Interrupt Occur Flag) 当进入睡眠模式且检测到 SOF 唤醒时, 该位由硬件置 1; 软件写 1 清 0。 0: 未从睡眠模式唤醒 1: 从睡眠模式唤醒
4	SLEEIFLG	RC_W1	已进入睡眠模式中中断标志 (Being Sleep Mode Interrupt Flag) 当进入睡眠模式时, 该位由硬件置 1, 触发相应的中断; 退出睡眠模式是由硬件清 0, 软件写 1 清 0。 0: 未进入睡眠模式 1: 进入睡眠模式
7:5	保留		
8	TXMFLG	R	正处于发送模式标志 (Being Transmit Mode Flag) 0: CAN 不处于发送模式 1: CAN 处于发送模式
9	RXMFLG	R	正处于接收模式标志 (Being Receive Mode Flag) 0: CAN 不处于接收模式 1: CAN 处于接收模式
10	LSAMVALUE	R	CAN 接收引脚上次采样值 (CAN Rx Pin Last Sample Value)
11	RXSIGL	R	CAN 接收引脚信号电平 (CAN Rx Pin Signal Level)
31:12	保留		

20.6.1.3 CAN 发送状态寄存器 (CAN_TXSTS)

偏移地址: 0x08

复位值: 0x1C00 0000

位/域	名称	R/W	描述
0	REQCFLG0	RC_W1	邮箱 0 请求完成标志 (Mailbox 0 Request Completed Flag) 当邮箱 0 的上次请求发送或中止完成时, 该位由硬件置 1; 当收到发送请求时, 由硬件清 0; 软件写 1 清 0。 0: 正在发送 1: 已完成发送
1	TXSUSFLG0	RC_W1	邮箱 0 发送成功标志 (Mailbox 0 Transmission Success Flag) 当邮箱 0 发送尝试成功后, 由硬件置 1; 软件写 1 清 0。 0: 上次发送尝试失败 1: 上次发送尝试成功

位/域	名称	R/W	描述
2	ARBLSTFLG0	RC_W1	邮箱 0 仲裁丢失标志 (Mailbox 0 Arbitration Lost Flag) 当邮箱 0 仲裁丢失时由硬件置 1; 软件写 1 清 0。 0: 无意义 1: 丢失
3	TXERRFLG0	RC_W1	邮箱 0 发送失败标志 (Mailbox 0 Transmission Error Flag) 当邮箱 0 发送失败时, 由硬件置 1; 软件写 1 清 0。 0: 无意义 1: 发送失败
6:4	保留		
7	ABREQFLG0	R/S	邮箱 0 放弃请求标志 (Mailbox 0 Abort Request Flag) 若邮箱 0 中没有等待发送的报文, 则该位无效。 0: 邮箱 0 的发送报文被清除, 由硬件清 0 1: 置 1 中止邮箱 0 的发送请求
8	REQCFLG1	RC_W1	邮箱 1 请求完成标志 (Mailbox 1 Request Completed Flag) 当邮箱 1 的上次请求发送或中止完成时, 该位由硬件置 1; 当收到发送请求时, 由硬件清 0; 软件写 1 清 0。 0: 正在发送 1: 已完发送
9	TXSUSFLG1	RC_W1	邮箱 1 发送成功标志 (Mailbox 1 Transmission Success Flag) 当邮箱 1 发送尝试成功后, 由硬件置 1; 软件写 1 清 0。 0: 上次发送尝试失败 1: 上次发送尝试成功
10	ARBLSTFLG1	RC_W1	邮箱 1 仲裁丢失标志 (Mailbox 1 Arbitration Lost Flag) 当邮箱 1 仲裁丢失时由硬件置 1; 软件写 1 清 0。 0: 无意义 1: 丢失
11	TXERRFLG1	RC_W1	邮箱 1 发送失败标志 (Mailbox 1 Transmission Error Flag) 当邮箱 1 发送失败时, 由硬件置 1; 软件写 1 清 0。 0: 无意义 1: 发送失败
14:12	保留		
15	ABREQFLG1	R/S	邮箱 1 放弃请求标志 (Mailbox 1 Abort Request Flag) 若邮箱 1 中没有等待发送的报文, 则该位无效。 0: 邮箱 1 的发送报文被清除, 由硬件清 0 1: 置 1 中止邮箱 1 的发送请求
16	REQCFLG2	RC_W1	邮箱 2 请求完成标志 (Mailbox 2 Request Completed Flag) 当邮箱 2 的上次请求发送或中止完成时, 该位由硬件置 1; 当收到发送请求时, 由硬件清 0; 软件写 1 清 0。 0: 正在发送 1: 已完发送
17	TXSUSFLG2	RC_W1	邮箱 2 发送成功标志 (Mailbox 2 Transmission Success Flag) 当邮箱 2 发送尝试成功后, 由硬件置 1; 软件写 1 清 0。 0: 上次发送尝试失败 1: 上次发送尝试成功

位/域	名称	R/W	描述
18	ARBLSTFLG2	RC_W1	邮箱 2 仲裁丢失标志 (Mailbox 2 Arbitration Lost Flag) 当邮箱 2 仲裁丢失时由硬件置 1; 软件写 1 清 0。 0: 无意义 1: 丢失
19	TXERRFLG2	RC_W1	邮箱 2 发送失败标志 (Mailbox 2 Transmission Error Flag) 当邮箱 2 发送失败时, 由硬件置 1; 软件写 1 清 0。 0: 无意义 1: 发送失败
22:20	保留		
23	ABREQFLG2	R/S	邮箱 2 放弃请求标志 (Mailbox 2 Abort Request Flag) 若邮箱 2 中没有等待发送的报文, 则该位无效。 0: 邮箱 2 的发送报文被清除, 由硬件清 0 1: 置 1 中止邮箱 2 的发送请求
25:24	EMNUM[1:0]	R	空邮箱编号 (Empty Mailbox Number) 该位适用于存在空邮箱的情况。当所有的发送邮箱全空时, 表示优先级最低的发送邮箱编号; 当邮箱非空但不是全空时, 表示下一个要发送的邮箱编号。
26	TXMEFLG0	R	发送邮箱 0 为空标志 (Transmit Mailbox 0 Empty Flag) 发送邮箱 0 为空时, 该位由硬件置 1。 0: 邮箱 0 中有待发送的报文 1: 邮箱 0 中没有待发送的报文
27	TXMEFLG1	R	发送邮箱 1 空标志 (Transmit Mailbox 1 Empty Flag) 发送邮箱 1 为空时, 该位由硬件置 1。 0: 邮箱 1 中有待发送的报文 1: 邮箱 1 中没有待发送的报文
28	TXMEFLG2	R	发送邮箱 2 空标志 (Transmit Mailbox 2 Empty Flag) 发送邮箱 2 为空时, 该位由硬件置 1。 0: 邮箱 2 中有待发送的报文 1: 邮箱 2 中没有待发送的报文
29	LOWESTP0	R	邮箱 0 的发送优先级最低标志 (the Lowest Transmission Priority Flag For Mailbox 0) 0: 无意义 1: 邮箱 0 的优先级在等待发送报文的邮箱中最低 注: 若只有 1 个邮箱在等待, 则 LOWESTP[2:0]被清 0。
30	LOWESTP1	R	邮箱 1 的发送优先级最低标志 (the Lowest Transmission Priority Flag For Mailbox 1) 0: 无意义 1: 邮箱 1 的优先级在等待发送报文的邮箱中最低
31	LOWESTP2	R	邮箱 2 的发送优先级最低标志 (the Lowest Transmission Priority Flag For Mailbox 2) 0: 无意义 1: 邮箱 2 的优先级在等待发送报文的邮箱中最低

20.6.1.4 CAN 接收 FIFO 0 寄存器 (CAN_RXF0)

偏移地址: 0x0C

复位值: 0x00

位/域	名称	R/W	描述
1:0	FMNUM0[1:0]	R	接收 FIFO0 中的报文数目 (the number of Message in receive FIFO0) 这些位用来反映当前接收 FIFO0 中存放的报文数目。每新接收到 1 个报文, FMNUM0 位加 1; 每释放输出邮箱的报文, FMNUM0 位减 1。
2	保留		
3	FFULLFLG0	RC_W1	接收 FIFO0 满标志 (Receive FIFO0 Full Flag) 当 FIFO0 中有 3 个报文时, 表明 FIFO0 已满; 该位由硬件置 1, 软件写 1 清 0。 0: 未满 1: 已满
4	FOVRFLG0	RC_W1	接收 FIFO0 溢出标志 (Receive FIFO 0 Overrun Flag) 当 FIFO0 中有 3 个报文时, 此时又收到新报文, 表明 FIFO0 溢出; 该位由硬件置 1, 软件写 1 清 0。 0: 未产生溢出 1: 产生溢出
5	RFOM0	R/S	释放接收 FIFO0 输出邮箱以接收报文 (Release Receive FIFO0 Output Mailbox to Receive Message) 该位由硬件置 1, 软件清 0。若 FIFO 中无报文, 则无效。当 FIFO 中有 2 个以上的报文时, 想要访问第 2 个报文必须先释放输出邮箱。 0: 无意义 1: 释放接收 FIFO0 的输出邮箱
31:6	保留		

20.6.1.5 CAN 接收 FIFO 1 寄存器 (CAN_RXF1)

偏移地址: 0x10

复位值: 0x00

位/域	名称	R/W	描述
1:0	FMNUM1[1:0]	R	接收 FIFO1 中的报文数目 (the number of Message in receive FIFO1) 这些位用来反映当前接收 FIFO1 中存放的报文数目。每新接收到 1 个报文, FMNUM1 位加 1; 每释放输出邮箱的报文, FMNUM1 位减 1。
2	保留		
3	FFULLFLG1	RC_W1	接收 FIFO1 满标志 (Receive FIFO0 Full Flag) 当 FIFO1 中有 3 个报文时, 表明 FIFO1 已满; 该位由硬件置 1, 软件写 1 清 0。 0: 未满 1: 已满
4	FOVRFLG1	RC_W1	接收 FIFO1 溢出标志 (Receive FIFO1 Overrun Flag) 当 FIFO1 中有 3 个报文时, 此时又收到新报文, 表明 FIFO1 溢出; 该位由硬件置 1, 软件写 1 清 0。 0: 未产生溢出 1: 产生溢出

位/域	名称	R/W	描述
5	RFOM1	R/S	<p>释放接收 FIFO1 输出邮箱以接收报文 (Release Receive FIFO1 Output Mailbox to Receive Message)</p> <p>该位由硬件置 1, 软件清 0。若 FIFO 中无报文, 则无效。当 FIFO 中有 2 个以上的报文时, 想要访问第 2 个报文必须先释放输出邮箱。</p> <p>0: 无意义 1: 释放接收 FIFO1 的输出邮箱</p>
31:6	保留		

20.6.1.6 CAN 中断使能寄存器 (CAN_INTEN)

偏移地址: 0x14

复位值: 0x0000 0000

位/域	名称	R/W	描述
0	TXMEIEN	R/W	<p>使能发送邮箱为空中断 (Transmit Mailbox Empty Interrupt Enable)</p> <p>当 REQFLGx 位置 1 时, 表明已完成发送, 发送邮箱为空, 若该位置 1 则产生中断。</p> <p>0: 不产生中断 1: 产生中断</p>
1	FMIEN0	R/W	<p>使能 FIFO0 中报文数量为非 0 时产生中断 (Interrupt Enable When The Number Of FIFO0 Message Is Not 0)</p> <p>当 FIFO 0 的 FMNUM0[1:0]位为非 0 时, 表明 FIFO0 的报文数量非 0, 若该位置 1 则产生中断。</p> <p>0: 不产生中断 1: 产生中断</p>
2	FFULLIEN0	R/W	<p>使能 FIFO0 满中断 (FIFO0 Full Interrupt Enable)</p> <p>当 FIFO0 的 FFULLFLG0 位置 1 时, 表明 FIFO0 的报文已满, 若该位置 1 则产生中断。</p> <p>0: 不产生中断 1: 产生中断</p>
3	FOVRIEN0	R/W	<p>使能 FIFO0 过载中断 (FIFO0 Overrun Interrupt Enable)</p> <p>当 FIFO0 的 FOVRFLG0 位置 1 时, 表明已 FIFO0 已过载, 若该位置 1 则产生中断。</p> <p>0: 不产生中断 1: 产生中断</p>
4	FMPIEN1	R/W	<p>使能 FIFO1 中报文数量为非 0 时产生中断 (Interrupt Enable when the number of FIFO1 Message is not 0)</p> <p>当 FIFO1 的 FMNUM1[1:0]位为非 0 时, 表明 FIFO1 的报文数量非 0, 若该位置 1 则产生中断。</p> <p>0: 不产生中断 1: 产生中断</p>
5	FFULLIEN1	R/W	<p>使能 FIFO1 满中断 (FIFO1 Full Interrupt Enable)</p> <p>当 FIFO1 的 FFULLFLG1 位置 1 时, 表明 FIFO1 的报文已满, 若该位置 1 则产生中断。</p> <p>0: 不产生中断 1: 产生中断</p>

位/域	名称	R/W	描述
6	FOVRIEN1	R/W	使能 FIFO 1 过载中断 (FIFO1 Overrun Interrupt Enable) 当 FIFO1 的 FOVRFLG1 位置 1 时, 表明已 FIFO1 已过载, 若该位置 1 则产生中断。 0: 不产生中断 1: 产生中断
7	保留		
8	ERRWIEN	R/W	使能错误警告中断 (Error Warning Interrupt Enable) 当 ERRWFLG 位置 1 时, 出现错误警告; 若该位置 1, 则设置 ERRIFLG, 产生警告错误中断。 0: 不设置 ERRIFLG 位 1: ERRIFLG 位置 1
9	ERRPIEN	R/W	使能被动错误中断 (Error Passive Interrupt Enable) 当 ERRPFLG 位置 1 时, 出现被动错误; 若该位置 1, 则设置 ERRIFLG, 产生被动错误中断。 0: 不设置 ERRIFLG 位 1: ERRIFLG 位置 1
10	BOFFIEN	R/W	使能离线中断 (Bus-Off Interrupt Enable) 当 BOFFFLG 位置 1 时, 出现离线; 若该位置 1, 则设置 ERRIFLG, 产生离线中断。 0: 不设置 ERRIFLG 位 1: ERRIFLG 位置 1
11	LECIEN	R/W	使能上次错误号中断 (Last Error Code Interrupt Enable) 当将测到错误、硬件设置 LERRC[2:0], 记录上次错误代码, 如果该位置 1, 则设置 ERRIFLG, 产生上次错误中断。 0: 不设置 ERRIFLG 位 1: ERRIFLG 位置 1
14:12	保留		
15	ERRIEN	R/W	使能错误中断 (Error Interrupt Enable) 当对应的错误状态寄存器 1 时, 若该位置 1, 则产生错误中断。 0: 不产生中断 1: 产生中断
16	WUPIEN	R/W	使能唤醒中断 (Wakeup Interrupt Enable) 当 WUPINT 位置 1 时, 若该位置 1, 则产生唤醒中断。 0: 不产生中断 1: 产生中断
17	SLEEPIEN	R/W	使能睡眠中断 (Sleep Interrupt Enable) 当 SLEEPFLG 位置 1 时, 若该位置 1, 则产生睡眠中断。 0: 不产生中断 1: 产生中断
31:18	保留		

20.6.1.7 CAN 错误状态寄存器 (CAN_ERRSTS)

偏移地址: 0x18

复位值: 0x0000 0000

位/域	名称	R/W	描述
0	ERRWFLG	R	出现错误警告标志 (Error Warning Occur Flag) 当接收错误计数器或发送错误计数器的值 ≥ 96 时, 该位由硬件置 1。 0: 未出现错误警告 1: 出现错误警告
1	ERRPFLG	R	出现被动错误标志 (Error Passive Occur Flag) 当接收错误计数器或发送错误计数器的值 >127 时, 该位由硬件置 1。 0: 未出现被动错误 1: 出现被动错误
2	BOFLG	R	进入离线标志 (Enter Bus-Off Flag) 当发送错误计数器 TXERRCNT 的值 >255 时, CAN 进入离线状态, 该位由硬件置 1。 0: CAN 未进入离线状态 1: CAN 进入离线状态
3	保留		
6:4	LERRC	R/W	记录上次错误代码 (Record Last Error Code) 当检测到 CAN 总线上的错误时, 依据错误类别由硬件设置; 当正确发送或接收报文时, 由硬件清 0。 000: 没有错误 001: 位填充错 010: 格式 (Form) 错 011: 确认 (ACK) 错 100: 隐性位错 101: 显性位错 110: CRC 错 111: 由软件设置
15:7	保留		
23:16	TXERRCNT	R	9 位发送错误计数器的低 8 位 (Least Significant Byte Of The 9-Bit Transmit Error Counter) 此计数器依据 CAN 协议的故障界定机制的发送部分实现。
31:24	RXERRCNT	R	接收错误计数器 (Receive Error Counter) 接收错误计数器是依据 CAN 协议的故障界定机制的接收部分实现, 当出现接收错误时, 根据出错的条件, 计数器加 1 或加 8, 接收成功后减 1, 当计数器的值 >127 , 设置计数器值为 120。

20.6.1.8 CAN 位时序寄存器 (CAN_BITTIM)

偏移地址: 0x1C

复位值: 0x0123 0000

位/域	名称	R/W	描述
9:0	BRPSC	R/W	配置波特率预分频器系数 (Baud Rate Prescaler Factor Setup) 时间单元 $t_q = (BRPSC+1) \times t_{PCLK}$
15:10	保留		
19:16	TIMSEG1	R/W	设置时间段 1 (Time Segment1 Setup) 时间段 1 占用的时间 $t_{BS1} = t_{CAN} \times (TIMSEG1+1)$ 。

22:20	TIMSEG2	R/W	设置时间段 2 (Time Segment2 Setup) 时间段 2 占用的时间 $t_{BS2} = t_{CAN} \times (TIMSEG2+1)$ 。
23	保留		
25:24	RSYNJW	R/W	重新同步跳跃宽度 (Resynchronization Jump Width) CAN 硬件在每位中可以延长或缩短多少的时间 $t_{rJW}=t_{CAN} \times (RSYNJW+1)$ 。
29:26	保留		
30	LBKMEN	R/W	使能环回模式 (Loop Back Mode Enable) 0: 禁止 1: 使能
31	SILMEN	R/W	使能静默模式 (Silent Mode Enable) 0: 正常状态 1: 静默模式

注：当 CAN 处于初始化模式时，该寄存器只能由软件访问

20.6.2 CAN 邮箱寄存器

本节描述发送和接收邮箱寄存器。

除了下述例外，发送和接收邮箱几乎一样：

- CAN_RXDLEN_x 寄存器的 FMIDX 域；
- 接收邮箱是只读的；
- 发送邮箱只有在它为空时才是可写的，CAN_TXSTS 寄存器的相应 TXMEFLG 位为 1，表示发送邮箱为空。

共有 3 个发送邮箱和 2 个接收邮箱。每个接收邮箱为 3 级深度的 FIFO，并且只能访问 FIFO 中最先收到的报文。

20.6.2.1 发送邮箱标识符寄存器 (CAN_TXMID_x) (x=0..2)

偏移地址：0x180, 0x190, 0x1A0

复位值：0xXXXX XXXX, X=未定义位 (除了第 0 位, 复位时 TXMREQ=0)

位/域	名称	R/W	描述
0	TXMREQ	R/W	请求发送邮箱数据 (Transmit Mailbox Data Request) 0: 邮箱中的数据发送完成时，邮箱为空，该位由硬件清 0 1: 软件写 1，使能请求发送邮箱数据
1	TXRFREQ	R/W	请求发送远程帧 (Transmit Remote Frame Request) 0: 数据帧 1: 远程帧
2	IDTYPESEL	R/W	选择标识符类型 (Identifier Type Select) 0: 标准标识符 1: 扩展标识符
20:3	EXTID[17:0]	R/W	设置扩展标识符 (Extended Identifier Setup) 扩展身份标识的低字节。
31:21	STDID[10:0]/EXTID[28:18]	R/W	标准标识符或扩展标识符 (Standard Identifier Or Extended Identifier) 根据 IDTYPESEL 位的内容，这些位是标准标识符 STDID[10:0]，还是扩展标识符的高字节 EXTID[28:18]。

注：1 当其所属的邮箱处在等待发送的状态时，该寄存器是写保护的

2 该寄存器实现了发送请求控制功能（第0位）—复位值为0

20.6.2.2 发送邮箱数据长度寄存器（CAN_TXDLENx）（x=0..2）

当邮箱不在空置状态时，该寄存器的所有位为写保护。

偏移地址：0x184, 0x194, 0x1A4

复位值：0xXXXX XXXX

位/域	名称	R/W	描述
3:0	DLCODE	R/W	设置发送数据长度码（Transmit Data Length Code Setup）
31:4	保留		

20.6.2.3 发送邮箱低字节数据寄存器（CAN_TXMDLx）（x=0..2）

当邮箱不在空置状态时，该寄存器的所有位为写保护，报文包含0到7个字节数据，且从字节0开始。

偏移地址：0x188, 0x198, 0x1A8

复位值：0xXXXX XXXX

位/域	名称	R/W	描述
7:0	DATABYTE0	R/W	报文数据字节0（Data Byte 0 of the Message）
15:8	DATABYTE1	R/W	报文数据字节1（Data Byte 1 of the Message）
23:16	DATABYTE2	R/W	报文数据字节2（Data Byte 2 of the Message）
31:24	DATABYTE3	R/W	报文数据字节3（Data Byte 3 of the Message）

20.6.2.4 发送邮箱高字节数据寄存器（CAN_TXMDHx）（x=0..2）

当邮箱不在空置状态时，该寄存器的所有位为写保护。

偏移地址：0x18C, 0x19C, 0x1AC

复位值：0xXXXX XXXX, X=未定义位

位/域	名称	R/W	描述
7:0	DATABYTE4	R/W	报文数据字节4（Data Byte 4 of the Message）。
15:8	DATABYTE5	R/W	报文数据字节5（Data Byte 5 of the Message）
23:16	DATABYTE6	R/W	报文数据字节6（Data Byte 6 of the Message）
31:24	DATABYTE7	R/W	报文数据字节7（Data Byte 7 of the Message）

20.6.2.5 接收 FIFO 邮箱标识符寄存器（CAN_RXMIDx）（x=0..1）

偏移地址：0x1B0, 0x1C0

复位值：0xXXXX XXXX, X=未定义位

位/域	名称	R/W	描述
0	保留		
1	RFTXREQ	R	发送请求远程帧（Remote Frame Transmission Request） 0: 数据帧 1: 远程帧

位/域	名称	R/W	描述
2	IDTYPESEL	R	选择标识符类型 (Identifier Type Select) 0: 标准标识符 1: 扩展标识符
20:3	EXTID[17:0]	R	设置扩展标识符 (Extended Identifier Setup) 扩展标识符的低字节。
31:21	STDID[10:0]/EXTID[28:18]	R	标准标识符或扩展标识符 (Standard Identifier Or Extended Identifier) 根据 IDTYPESEL 位的内容, 这些位是标准标识符 STDID[10:0], 还是扩展标识符的高字节 EXTID[28:18]。

注: 所有接收邮箱寄存器都是只读的。

20.6.2.6 接收 FIFO 邮箱数据长度寄存器 (CAN_RXDLENx) (x=0..1)

偏移地址: 0x1B4, 0x1C4

复位值: 0xXXXXX XXXX

位/域	名称	R/W	描述
3:0	DLCODE	R	设置接收数据长度码 (Receive Data Length Code Setup) 该位表示帧内的数据长度; 对于远程帧, DLCODE 恒为 0。
7:4	保留		
15:8	FMIDX	R	设置过滤器匹配序号 (Filter Match Index Setup)
31:16	保留		

注: 所有接收邮箱寄存器都是只读的。

20.6.2.7 接收 FIFO 邮箱低字节数据寄存器 (CAN_RXMDLx) (x=0..1)

偏移地址: 0x1B8, 0x1C8, 报文包含 0 到 8 个字节数据, 且从字节 0 开始。

复位值: 0xXXXXX XXXX

位/域	名称	R/W	描述
7:0	DATABYTE0	R	报文数据字节 0 (Data Byte 0 of the Message)
15:8	DATABYTE1	R	报文数据字节 1 (Data Byte 0 of the Message)
23:16	DATABYTE2	R	报文数据字节 2 (Data Byte 0 of the Message)
31:24	DATABYTE3	R	报文数据字节 3 (Data Byte 0 of the Message)

注: 所有接收邮箱寄存器都是只读的。

20.6.2.8 接收 FIFO 邮箱高字节数据寄存器 (CAN_RXMDHx) (x=0..1)

偏移地址: 0x1BC, 0x1CC

复位值: 0xXXXX XXXX, X=未定义位

位/域	名称	R/W	描述
7:0	DATABYTE4	R	报文数据字节 4 (Data Byte 0 of the Message)
15:8	DATABYTE5	R	报文数据字节 5 (Data Byte 0 of the Message)
23:16	DATABYTE6	R	报文数据字节 6 (Data Byte 0 of the Message)
31:24	DATABYTE7	R	报文数据字节 7 (Data Byte 0 of the Message)

注: 所有接收邮箱寄存器都是只读的。

20.6.3 CAN 过滤器寄存器

20.6.3.1 CAN 过滤器控制寄存器 (CAN_FCTRL)

偏移地址: 0x200

复位值: 0x2A1C 0E01

位/域	名称	R/W	描述
0	FINITEN	R/W	使能过滤器初始化模式 (Filter Init Mode Enable) 0: 正常模式 1: 初始化模式
31:1	保留		

注: 该寄存器的非保留位完全由软件控制。

20.6.3.2 CAN 过滤器模式配置寄存器 (CAN_FCFG)

偏移地址: 0x204

复位值: 0x0000 0000

位/域	名称	R/W	描述
13:0	FMCFGx	R/W	配置过滤器 x 模式 (Filter Mode Configure) x 取值为 0-13。 0: 标识符屏蔽位模式 1: 标识符列表模式
31:14	保留		

注: 只有在设置 CAN_FCTRL (FINITEN =1), 使过滤器处于初始化模式下, 才能对该寄存器写入。

20.6.3.3 CAN 过滤器位宽配置寄存器 (CAN_FSCFG)

偏移地址: 0x20C

复位值: 0x0000 0000

位/域	名称	R/W	描述
13:0	FSCFGx	R/W	配置过滤器 x 位宽 (Filterx Scale Configure) x 取值为 0-13。 0: 2 个 16 位 1: 单个 32 位
31:14	保留		

注: 只有在设置 CAN_FCTRL (FINITEN=1), 使过滤器处于初始化模式下, 才能对该寄存器写入。

20.6.3.4 CAN 过滤器 FIFO 关联寄存器 (CAN_FFASS)

偏移地址: 0x214

复位值: 0x0000 0000

位/域	名称	R/W	描述
0	FFASS0	R/W	配置过滤器 0 与 FIFO 关联 (Configure Filter0 Associated with FIFO) 0: 过滤器与 FIFO0 关联 1: 过滤器与 FIFO1 关联
1	FFASS1	R/W	配置过滤器 1 与 FIFO 关联 (Configure Filter1 Associated with FIFO) 具体描述参考 FFASS0。
2	FFASS2	R/W	配置过滤器 2 与 FIFO 关联 (Configure Filter2 Associated with FIFO) 具体描述参考 FFASS0。

位/域	名称	R/W	描述
3	FFASS3	R/W	配置过滤器 3 与 FIFO 关联 (Configure Filter3 Associated with FIFO) 具体描述参考 FFASS0。
4	FFASS4	R/W	配置过滤器 4 与 FIFO 关联 (Configure Filter4 Associated with FIFO) 具体描述参考 FFASS0。
5	FFASS5	R/W	配置过滤器 5 与 FIFO 关联 (Configure Filter5 Associated with FIFO) 具体描述参考 FFASS0。
6	FFASS6	R/W	配置过滤器 6 与 FIFO 关联 (Configure Filter6 Associated with FIFO) 具体描述参考 FFASS0。
7	FFASS7	R/W	配置过滤器 7 与 FIFO 关联 (Configure Filter7 Associated with FIFO) 具体描述参考 FFASS0。
8	FFASS8	R/W	配置过滤器 8 与 FIFO 关联 (Configure Filter8 Associated with FIFO) 具体描述参考 FFASS0。
9	FFASS9	R/W	配置过滤器 9 与 FIFO 关联 (Configure Filter9 Associated with FIFO) 具体描述参考 FFASS0。
10	FFASS10	R/W	配置过滤器 10 与 FIFO 关联 (Configure Filter10 Associated with FIFO) 具体描述参考 FFASS0。
11	FFASS11	R/W	配置过滤器 11 与 FIFO 关联 (Configure Filter11 Associated with FIFO) 具体描述参考 FFASS0。
12	FFASS12	R/W	配置过滤器 12 与 FIFO 关联 (Configure Filter12 Associated with FIFO) 具体描述参考 FFASS0。
13	FFASS13	R/W	配置过滤器 13 与 FIFO 关联 (Configure Filter13 Associated with FIFO) 具体描述参考 FFASS0。
31:14	保留		

注：只有在设置 CAN_FCTRL (FINITEN=1)，使过滤器处于初始化模式下，才能对该寄存器写入。

20.6.3.5 CAN 过滤器激活寄存器 (CAN_FACT)

偏移地址：0x21C

复位值：0x0000 0000

位/域	名称	R/W	描述
0	FACT0	R/W	激活过滤器 0 (Filter0 Active) 0: 禁用 1: 激活
1	FACT1	R/W	激活过滤器 1 (Filter1 Active) 具体描述参考 FACT0
2	FACT2	R/W	激活过滤器 2 (Filter3 Active) 具体描述参考 FACT0
3	FACT3	R/W	激活过滤器 3 (Filter3 Active) 具体描述参考 FACT0
4	FACT4	R/W	激活过滤器 4 (Filter4 Active) 具体描述参考 FACT0
5	FACT5	R/W	激活过滤器 5 (Filter5 Active) 具体描述参考 FACT0

位/域	名称	R/W	描述
6	FACT6	R/W	激活过滤器 6 (Filter6 Active) 具体描述参考 FACT0
7	FACT7	R/W	激活过滤器 7 (Filter7 Active) 具体描述参考 FACT0
8	FACT8	R/W	激活过滤器 8 (Filter8 Active) 具体描述参考 FACT0
9	FACT9	R/W	激活过滤器 9 (Filter9 Active) 具体描述参考 FACT0
10	FACT10	R/W	激活过滤器 10 (Filter10 Active) 具体描述参考 FACT0
11	FACT11	R/W	激活过滤器 11 (Filter11 Active) 具体描述参考 FACT0
12	FACT12	R/W	激活过滤器 12 (Filter12 Active) 具体描述参考 FACT0
13	FACT13	R/W	激活过滤器 13 (Filter13 Active) 具体描述参考 FACT0
31:14	保留		

20.6.3.6 CAN 过滤器组 i 的寄存器 x (CAN_FiBANKx) (i=0..13; x=1..2)

偏移地址: 0x240..0x2AC

复位值: 0xFFFF XXXX

位/域	名称	R/W	描述
31:0	FBIT[31:0]	R/W	设置过滤器位 (Filter Bits Setup) 标识符列表模式: 0: FBITx 为显性位 1: FBITx 位为隐性位 标识符屏蔽位模式: 0: FBITx 不用于比较 1: FBITx 必须匹配 说明: x 取值为 0~31, 表示 FBIT 的位编号。

注: 在产品中共有 14 组过滤器: i=0..13。每组过滤器由 2 个 32 位的寄存器, CAN_FiBANK[2:1]组成。只有在 CAN_FACT 寄存器相应的 FACTx 位清 0, 或 CAN_FCTRL 寄存器的 FINITEN 位为 1 时, 才能修改相应的过滤器寄存器。

21 通用同步异步收发器 (USART)

21.1 术语全称、缩写描述

表格 67 术语全称、缩写描述

中文全称	英文全称	英文缩写
清除发送	Clear to Send	CTS
请求发送	Request to Send	RTS
最高有效位	Most Significant Bit	MSB
最低有效位	Least Significant Bit	LSB
保护	Guard	GRD
溢出	Overrun	OVR

21.2 简介

USART (通用同步异步收发器) 是一个可以灵活地与外部设备进行全双工、半双工数据交换的串行通信设备, 且同时满足外部设备对工业标准 NRZ 异步串行数据格式的要求。USART 还提供宽范围的波特率选择, 且支持多处理器通信。

USART 不仅支持标准的异步收发模式, 还支持同步单向通信和一些其他的串行数据交换模式, 如 LIN 协议、智能卡协议、IrDA SIR ENDEC 规范和硬件流控制模式。

USART 还支持使用 DMA 功能, 以实现高速数据通信。

21.3 主要特征

- (1) 全双工异步通信
- (2) 单线半双工通信
- (3) NRZ 标准格式
- (4) 可编程的串口特性:
 - 数据位: 7 位、8 位或 9 位
 - 校验位: 偶校验、奇校验、无校验
 - 支持 0.5、1、1.5、2 个停止位
- (5) 校验控制
 - 发送校验位
 - 校验接收的数据
- (6) 选择速度和时钟容忍度采用可编程的 8 倍或 16 倍过采样率
- (7) 可编程高位优先或低位优先

- (8) 独立的发送器和接收器使能位
- (9) 独立的信号极性控制发送器和接收器
- (10) 可切换 TX/RX 引脚
- (11) 支持超时检测
- (12) 可编程的波特率发生器，波特率最高可达 6Mbits/s
- (13) 波特率自动检测
- (14) 多处理器通信：
 - 若地址不匹配，则进入静默模式
 - 通过空闲总线检测或地址标记检测，从静默模式中唤醒
- (15) 双时钟驱动
 - 停机模式下唤醒功能
 - 独立于 PCLK 的波特率选择
- (16) 同步传输模式
- (17) LIN 断开帧的生成与检测
- (18) 支持 ISO7816-3 标准的智能卡接口
- (19) 支持 IrDA 协议
- (20) 支持硬件流控制和 RS485 驱动使能
- (21) 可利用 DMA 连续通信
- (22) 支持 ModBus 通信
 - 超时检测
 - CR/LF 字符识别
- (23) 状态标志位：
 - 传输检测标志：发送寄存器为空、接收寄存器不为空、发送完成
 - 错误检测标志：溢出错误、噪音错误、奇偶校验错误、帧错误
- (24) 多个中断源：
 - 发送寄存器为空
 - 发送完成
 - CTS 改变
 - 接收寄存器不为空
 - 过载错误
 - 总线空闲
 - 奇偶校验错误
 - LIN 断开检测
 - 噪音错误
 - 溢出错误
 - 帧错误

- 地址/字符匹配
- 从停机模式下唤醒
- 未按时接收中断
- 块结束中断

21.4 功能描述

表格 68 USART 引脚描述

引脚	类型	描述
USART_RX	输入	数据接收
USART_TX	输出 I/O (单线模式/智能卡模式)	数据发送 当发送器被使能且不发送数据时, 默认为高电平
USART_CK	输出	时钟输出
USART_nRTS	输入	硬件流控制模式中发送请求
USART_nCTS	输出	硬件流控制模式中清除发送
USART_DE	输入	驱动使能激活外部发送/接收器

21.4.1 单线半双工通信

USART_CTRL3 寄存器的 HDEN 位决定是否进入单线半双工模式。

当 USART 进入单线半双工模式时:

- USART_CTRL2 寄存器的 CLKEN 位、LINMEN 位, USART_CTRL3 寄存器的 IREN 位、SCEN 位必须清 0。
- 禁止使用 RX 引脚。
- TX 脚应配置成开漏输出, 并在芯片内部与 RX 引脚相连。
- 发送数据和接收数据不可同时进行, 在数据发送前, 不可以接收数据。若需要接收数据, 必须在 USART_STS 寄存器的 TXCFLG 位置 1 后方可开启使能接收。
- 若总线上出现数据冲突, 需要使用软件管理分配通信过程。

21.4.2 帧格式

通过 USART_CTRL1 寄存器控制数据帧的帧格式

- DBLCFG 位控制字符长度, 可设置为 7 位、8 位或 9 位
- PCEN 位控制是否开启校验位
- PCFG 位控制校验位为奇校验还是偶校验

表格 69 USART 帧格式

DBLCFG 位	PCEN 位	USART 数据帧
00	0	起始位+8 位数据+停止位
00	1	起始位+7 位数据+奇偶检验位+停止位

01	0	起始位+9 位数据+停止位
01	1	起始位+8 位数据+奇偶检验位+停止位
10	0	起始位+7 位数据+停止位
10	1	起始位+6 位数据+奇偶校验位+停止位

可配置的停止位

可以通过 USART_CTRL2 寄存器的 STOPCFG 位来配置 4 种不同的停止位。

- 1 个停止位：默认的停止位
- 0.5 个停止位：智能卡模式里接收数据时使用
- 2 个停止位：常规模式、单线模式、硬件流控制模式使用
- 1.5 个停止位：智能卡模式里发送与接收数据时使用

校验位

USART_CTRL1 的 PCFG 位决定奇偶校验位，当 PCFG=0 时，为偶校验，反之为奇校验。

- 偶校验：帧数据和校验位 ‘1’ 的个数为偶数时，偶校验位为 0，否则为 1。
- 奇校验：帧数据和校验位 ‘1’ 的个数为偶数时，奇校验位为 1，否则为 0。
- 校验生成：发送数据时，置位 USART_CTRL1 寄存器的 PCEN 位，校验位会替换数据的 MSB 位被发送。
- 校验检查：
 - 若校验检查失败，USART_STS 寄存器的 PEFLG 标志位置位。
 - 如果使能校验控制，将引发相应中断。向 USART_INTFCLR 寄存器的 PECLR 位写 1，可以清除 PEFLG 标志位。

21.4.3 发送器

当寄存器 USART_CTRL1 的 TXEN 位被设置时，发送移位寄存器将通过 TX 脚输出数据，相应的时钟脉冲通过 CK 脚输出。

21.4.3.1 字符发送

USART 发送期间，数据的最低有效位会先被 TX 引脚移出。在此模式下，USART_TXDATA 寄存器有一个缓冲器，处于内部总线和发送移位寄存器之间。

一个数据帧由起始位、字符、停止位组成的，所以每个字符前面都有一个低电平的起始位；之后有一个数目可配置的高电平的停止位。

发送配置步骤

- (1) 通过设置 USART_CTRL1 寄存器的 DBLCFG 位来决定字长
- (2) 通过设置 USART_CTRL2 寄存器的 STOPCFG 位来决定停止位数
- (3) 若选择多缓冲器通信，需在 USART_CTRL3 寄存器中使能 DMA

- (4) 在 USART_BR 寄存器中设置通信的波特率
- (5) 置位 USART_CTRL1 寄存器的 UEN 位，使能 USART。等待 USART_STS 寄存器的 TXBEFLG 位置 1
- (6) 使能 USART_CTRL1 寄存器的 TXEN 位，发送一个空闲帧
- (7) 向 USART_TXDATA 寄存器写入数据（如果未使能 DMA，则每个需要发送的字节都要重复步骤 7）
- (8) 等待 USART_STS 寄存器的 TXCFLG 位置 1，表示发送完成

注意：不能在发送数据期间复位 TXEN 位，否则 TX 脚上的数据会被破坏，因为波特率发生器停止计数，正在传输的数据也将丢失。

21.4.3.2 单字节通信

对 USART_TXDATA 寄存器进行写操作可以将 TXBEFLG 位清零。当 TXBEFLG 位被硬件置位时，移位寄存器接收到由数据发送寄存器转移过来的数据，数据发送开始，此时数据发送寄存器被清空。可以在数据寄存器中写入下一个数据，且不会覆盖前面的数据。

- (1) 若 USART_CTRL1 寄存器中的 TXBEIEN 置 1，则会产生一个中断。
- (2) 若 USART 处于发送数据的状态时，对数据寄存器进行写操作，把数据存到 TXDATA 寄存器，并在当前数据发送结束时把该数据转移到移位寄存器中。
- (3) 若 USART 处于空闲状态时，对数据寄存器进行写操作，把数据放到移位寄存器中，开始发送数据，TXBEFLG 位置 1。
- (4) 当一个数据发送完成并设置了 TXBEFLG 位时，TXCFLG 位被置 1，此时若 USART_CTRL1 寄存器中的 TXCIEN 位被置 1，则会产生一个中断。
- (5) 在 USART_TXDATA 寄存器中写入最后一个数据后，在进入低功耗模式之前或关闭 USART 模块之前，必须先等待 TXCFLG 置 1。

21.4.3.3 断开帧

断开帧被视为在一个帧周期内全部收到‘0’。设置 USART_REQUEST 寄存器的 TXBFQ 位可发送一个断开帧，断开帧的长度由 USART_CTRL1 寄存器的 DBLCFG 位决定。若将 TXBFQ 位置位，则当前数据发送完成后，TX 线上将发送一个断开帧，断开帧发送完成时被复位。在断开帧结束时，发送器再插入 1 或 2 个停止位来应答起始位。

注意：若开始发送断开帧之前就复位了 TXBFQ 位，将不再发送断开帧。若要发送两个连续的断开帧，应该在前一个断开符的停止位之后置起 TXBFQ 位。

21.4.3.4 空闲帧

空闲帧被视为完全由‘1’组成的一个完整的数据帧，后面跟着包含了数据的下一帧的开始位。将 USART_CTRL1 寄存器的 TXEN 位置 1，可以在第一个数据

帧前发送一个空闲帧。

21.4.4 接收器

21.4.4.1 字符接收

USART 接收期间，RX 脚会首先引进数据的最低有效位。在此模式下，USART_RXDATA 寄存器有一个缓冲器，处于内部总线和接收移位寄存器之间。数据一位一位地送入缓冲器，接收满后，对应的接收寄存器不为空，此时用户可读取 USART_RXDATA。

接收配置步骤

- (1) 编程过采样率为 8 倍或 16 倍
- (2) 通过设置 USART_CTRL1 寄存器的 DBLCFG 位来决定字长
- (3) 通过设置 USART_CTRL2 寄存器的 STOPCFG 位来决定停止位位数
- (4) 置位 USART_CTRL1 寄存器的 UEN 位，使能 USART
- (5) 若选择多缓冲器通信，需在 USART_CTRL3 寄存器中使能 DMA
- (6) 在 USART_BR 寄存器中设置通信的波特率
- (7) 设置 USART_CTRL1 的 RXEN 位，使能接收

注意：

- (1) 不能在接收数据期间复位 RXEN 位，否则将丢失正在接收的字节。
- (2) 接收器在接收一个数据帧的过程中，若检测到溢出错误、噪音错误、帧错误等，错误标志被置 1。
- (3) 当数据从移位寄存器中被转移到 USART_RXDATA 寄存器中时，USART_STS 寄存器的 RXBNEFLG 位被硬件置位。
- (4) 若设置 RXBNEIEN 位将产生一个中断。
- (5) 单缓冲器模式中，软件读取 USART_RXDATA 寄存器能清除 RXBNEFLG 位，也可对其写 0 清除。
- (6) 多缓冲器模式中，每个字节被接收后，USART_STS 寄存器的 RXBNEFLG 位都会被置 1，且 DMA 对数据寄存器进行读操作而清零。

21.4.4.2 断开帧

当接收器接收到一个断开帧时，USART 会像接收到帧错误一样处理它。

21.4.4.3 空闲帧

当接收器接收到一个空闲帧时，USART 会像接收到普通数据帧一样处理它，如果设置 USART_CTRL1 的 IDLEIEN 位，将产生一个中断。

21.4.4.4 选择时钟源

在启用 USART 之前必须通过时钟控制系统选择时钟源

- (1) 根据传输速度和 USART 在低功耗模式下的使用可能性选择时钟源。
- (2) 时钟源频率为 f_{CK} 。
 - 通信速度范围由时钟源决定，需先使能 USART 再选择时钟源。
 - 当 USART 使用双时钟域或唤醒停机模式时，PCLK、LSECLK、HSICLK 或 SYSCLK 都可以是时钟源，否则，时钟源是 PCLK。
 - 若选择 LSECLK、LSICLK 为时钟源，可以使 USART 在低功耗模式下也能接收到数据。并根据接收到的数据和唤醒模式选择，在需要时唤醒 MCU，以便 DMA 读取接收到的数据。
 - 接收器实现用户配置的不同的过采样技术的数据恢复来区分有效的传入数据和噪音，这需要在最大通信速度和噪音/时钟不准确抗扰之间权衡。

21.4.4.5 过采样率

USART_CTRL1 寄存器的 OSMCFG 位决定过采样率。

若过采样率为波特率的 8 倍，则速度较高，但时钟容忍度较小，若为 16 倍则相反。

21.4.4.6 溢出错误

若 USART_STS 寄存器的 RXBNEFLG 位置 1，同时接收到一个新的字符，则会产生溢出错误。只有当 RXEN 复位后才能把数据从移位寄存器转移到 RXDATA 寄存器。接收到字节后 RXBNEFLG 位会置 1。该位需要在接收到下一个数据或还没服务先前的 DMA 请求前复位，否则将产生溢出错误。

当产生溢出错误时

- USART_STS 的 OVREFLG 位置 1
- 不会丢失 RXDATA 寄存器中的数据
- 覆盖之前接收到的移位寄存器中的数据，但不会保存后面接收到的数据
- 若设置 USART_CTRL1 的 RXBNEIEN 位或 ERRIEN 位，则会产生一个中断
- 当 OVREFLG 位被置位时，表明已经有数据丢失。可分为两种可能：
 - 当 RXBNEFLG=1 时，上一个有效数据还在 RXDATA 寄存器上，可以进行读操作
 - 当 RXBNEFLG=0 时，RXDATA 寄存器中没有有效数据
- 对 USART_STS 和 USART_RXDATA 寄存器依次进行读操作，可以复位 OVREFLG 位

21.4.4.7 噪音错误

当接收器在接收过程中检测到噪音时：

- 在 USART_STS 寄存器的 RXBNEFLG 位的上升沿设置 NEFLG 标志

- 无效数据从移位寄存器发送到 USART_RXDATA 寄存器
- 在单字节通信时不会产生中断，但在多缓冲区通信时置位 USART_CTRL3 寄存器的 ERRIEN 位就会产生一个中断

注意：8 倍过采样率不能在 LIN、智能卡和 IrDA 模式下使用。

21.4.4.8 帧错误

若因为噪音过大或没有同步而没有在预计的接收时间接收和识别到停止位，会检测到帧错误。

当接收器在接收过程中检测到帧错误时：

- 置位 USART_STS 寄存器的 FEFLG 位
- 无效数据从移位寄存器发送到 USART_RXDATA 寄存器
- 在单字节通信时不会产生中断，但在多缓冲区通信时置位 USART_CTRL3 寄存器的 ERRIEN 位就会产生一个中断

21.4.5 接收器对时钟的变化容忍度

只有当总时钟系统偏差小于 USART 接收器的容差时，才能使 USART 接收器正常工作。

当发生以下情况时，会造成偏差：

- (1) DTRA：因发送器误差引起的偏差
- (2) DQUANT：因接收器波特率量化引起的偏差
- (3) DREC：接收器振荡器的变化
- (4) DTCL：因传输线引起的偏差

21.4.6 波特率发生器

波特率分频系数 (USARTDIV) 是一个 16 位的数字，包含 12 位整数部分和 4 位小数部分。它与系统时钟的关系：

$$\text{波特率} = \text{PCLK} / 16 \times (\text{USARTDIV})$$

USART2 的系统时钟为 PCLK1，USART1 的系统时钟为 PCLK2。必须在时钟控制单元使能系统时钟之后再使能 USART。

21.4.7 波特率自动检测

在接收到一个字符时，USART 能够检测和自动设置 USART_BR 寄存器的值。波特率自动检测作用于在不知道系统的通信速度时、或在使用低精度时钟源时，不测量时钟偏差获取正确的比特率时。时钟源必须与预期的通信速度兼容。

必须写入一个非零波特率来初始化，确认字符内容，再打开波特率自动检测。可以设置 USART_CTRL2 寄存器的 ABRDCFG 位来选择字符内容，可能的字符内容：

- (1) 所有以 1 开头的字符，在这种情况下，对起始位的长度进行测量（下降沿到上升沿的持续时间）。

- (2) 所有以 10xx 开头的字符，在这种情况下，对起始位和首个数据位长度进行测量下降沿之间的时长，确保小信号摆率的情况下更好的准确性。

USART_CTRL2 寄存器中的 ABRDEN 位决定是否打开波特率自动检测。打开波特率自动检测后等待 RX 线上的第一个字符。检测结束后，USART_STS 寄存器中的 ABRDFLG 标志位被置位。

注意：

- (1) 如果线路噪音过大，则无法保证正确的波特率，在这种情况下，BR 值可能被损坏，ABRDFLG 标志位被置位。如果通信速度与波特率自动检测不兼容，也会发生这种情况。
- (2) 检测结束后会产生 RXBNEFLG 中断。
- (3) 在任何时候，波特率自动检测都可能通过重置 ABRDFLG 标志（写一个 0）重启。
- (4) 不能在波特率自动检测时禁用 USART，否则可能损坏 BR 的值。

21.4.8 多处理器通信

在多处理器通信中，多个 USART 连接组成一个网络。在该网络中，两个设备进行通信，其余不参与通信的设备，为减轻 USART 的负担，可启用静默模式。在静默模式中，USART_CTRL2 寄存器的 LINMEN 位、USART_CTRL3 寄存器的 SCEN 位、IREN 位和 HDEN 位清 0，且不会设置任何接收状态位，禁止所有接收中断。

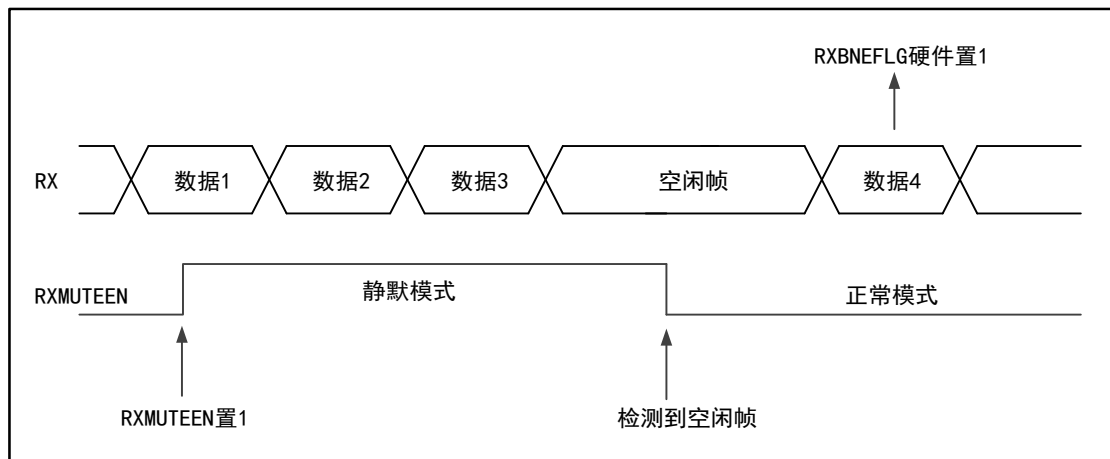
当启用静默模式后，有两种方法可退出静默模式：

- (1) WUPMCFG 位被清除，总线空闲可退出静默模式。
- (2) WUPMCFG 位被置位，收到地址标记可退出静默模式。

空闲总线检测 (WUPMCFG=0)

当 RXWFMUTE 置 1 时，USART 则进入静默模式，当检测到空闲帧时，能从静默模式中被唤醒，同时 RXWFMUTE 位被硬件清 0。RXWFMUTE 还可以通过软件清 0。

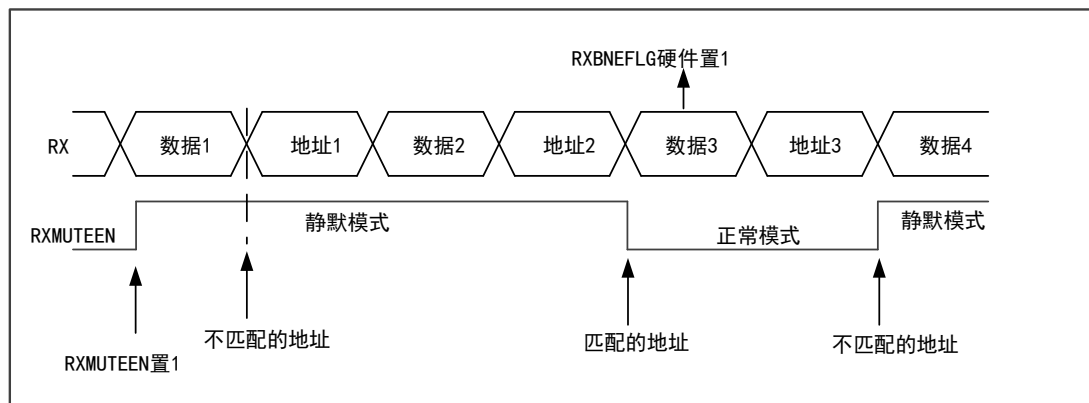
图 77 空闲总线退出静默模式



地址标记检测 (WUPMCFG=1)

如果地址标志位为 1，该字节被认为是地址。地址字节低 4 位存放地址，当接收器收到地址字节时，会先与自己的地址作比较。若地址不匹配，则进入静默模式，若地址匹配，则接收器从静默模式中被唤醒，并准备接收下一个字节。若退出静默模式后再次收到地址字节，但地址与自己的地址不匹配，则接收器会再次进入静默模式。

图 78 地址标记退出静默模式



21.4.9 从停机模式唤醒

当 USART 使用 HSICLK、LSECLK 作为时钟源时，USART_CTRL1 寄存器的 USWMEN 位决定是否从停机模式唤醒。在进入停机模式之前先要将 USART_CTRL1 寄存器的 USWMEN 位置位，每当检测到唤醒时间，WSMFLG 置 1，此时只要 WSMIEN 置位，将会产生一个中断。

停机模式下的静默模式

不能再空闲检测下退出静默模式，若使用地址匹配退出静默模式，只能地址匹配事件作为它的唤醒源。若设置起始位检测唤醒，那么 WSMFLG 置位，

RXBNEFLG 标志位不会。

21.4.10 同步模式

同步模式支持主模式下的全双工同步串行通信，比异步模式多了一个可以输出同步时钟的信号线 `USART_CK`。

`USART_CTRL2` 寄存器的 `CLKEN` 位决定是否进入同步模式。

当 `USART` 进入同步模式时：

- `USART_CTRL2` 寄存器的 `LINMEN` 位，`USART_CTRL3` 寄存器的 `IREN`，`HDEN` 和 `SCEN` 位必须清 0
- 数据帧的起始位和停止位无时钟输出
- 数据帧的最后一个数据位是否产生 `USART_CK` 时钟，由寄存器 `USART_CTRL2` 的 `LBCPOEN` 位决定
- `USART_CK` 的时钟极性由 `USART_CTRL2` 寄存器的 `CPOL` 位决定
- `USART_CK` 的相位由 `USART_CTRL2` 寄存器的 `CPHA` 位决定
- 总线空闲或出现断开帧时不能激活外部 `CK` 时钟

图 79 USART 同步传输的例子

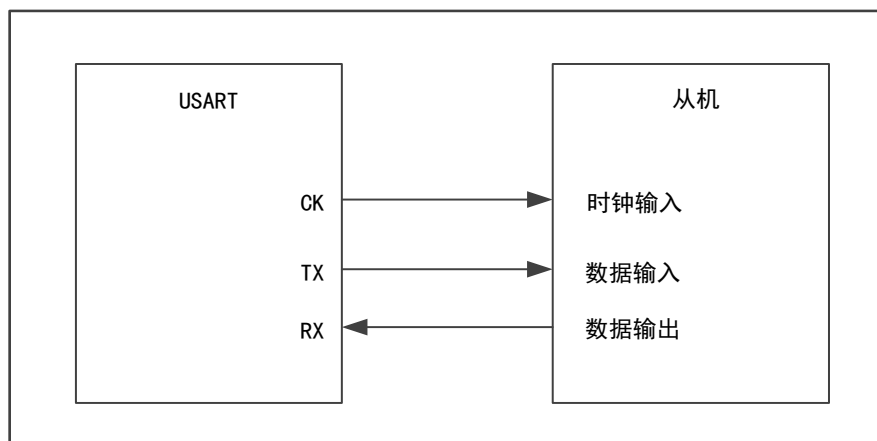


图 80 USART 同步传输时序图 (DBLCFG=10)

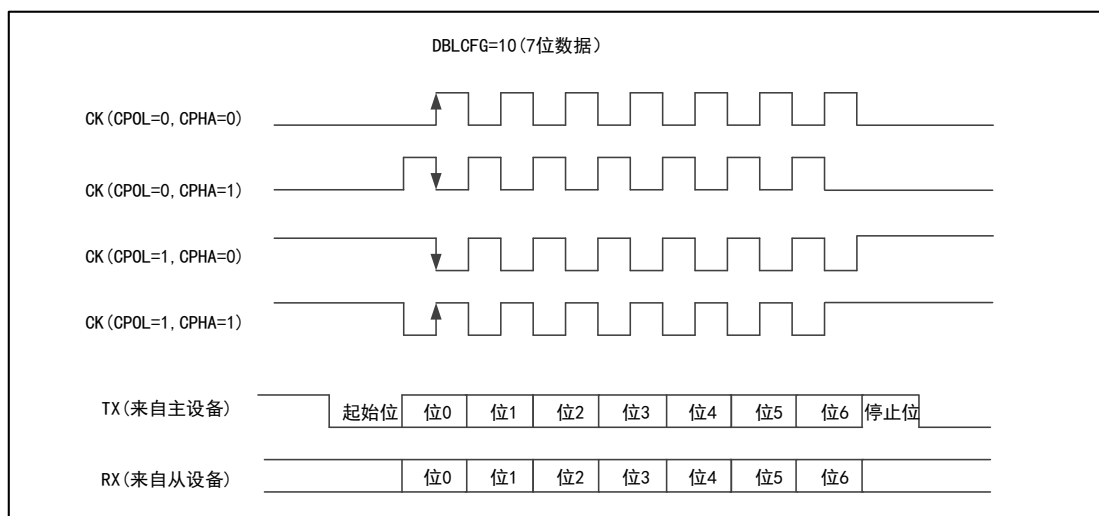


图 81 USART 同步传输时序图 (DBLCFG=00)

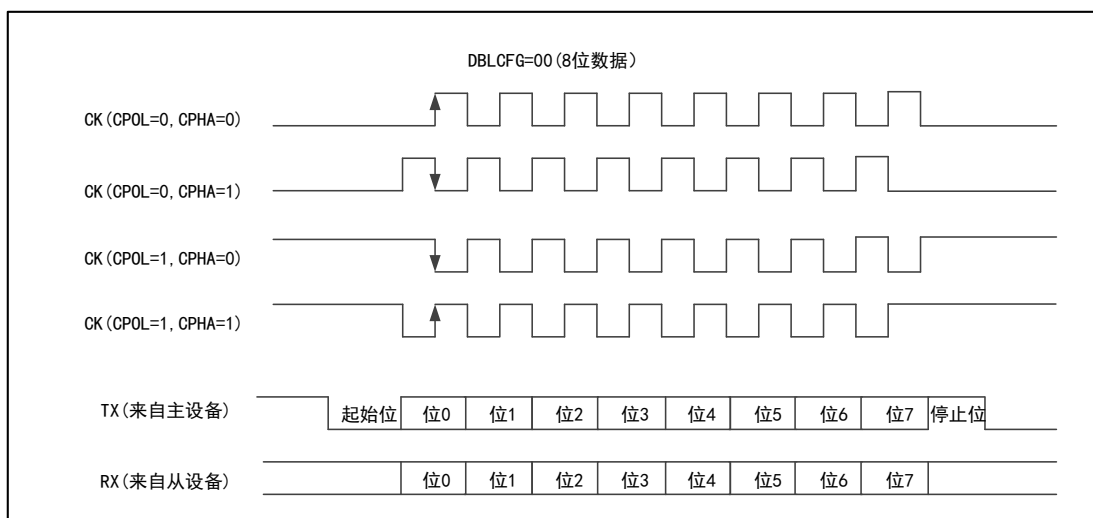
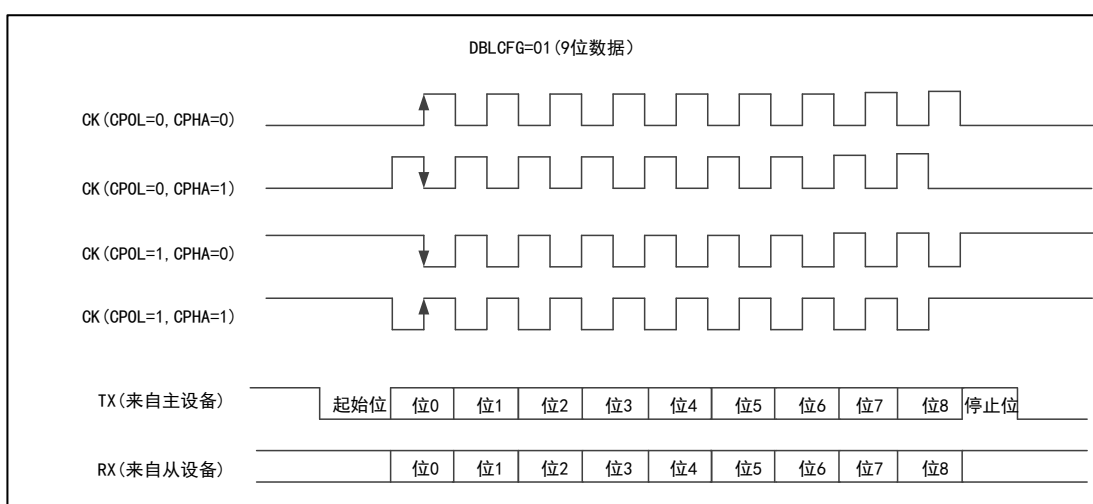


图 82 USART 同步传输时序图 (DBLCFG=01)



21.4.11 LIN 模式

USART_CTRL2 寄存器的 LINMEN 位决定是否进入 LIN 模式。

当进入 LIN 模式时：

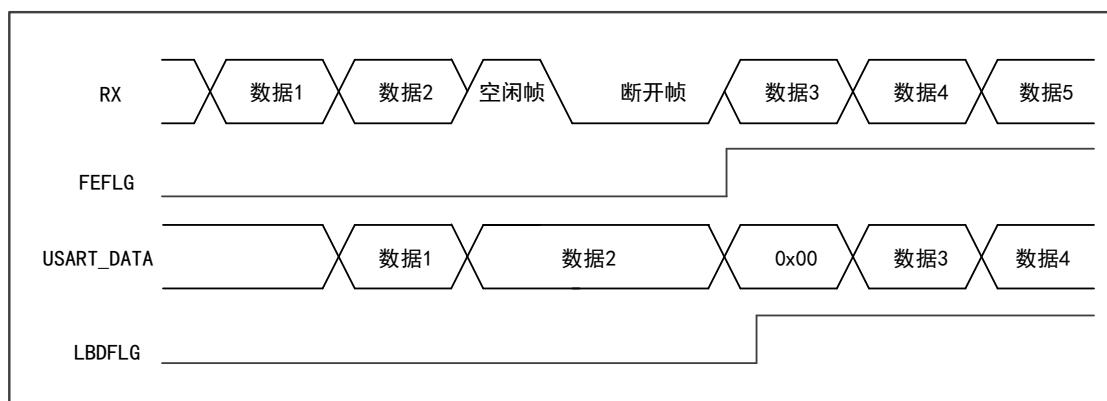
- 数据帧都为 8 位数据位和 1 位停止位
- USART_CTRL2 寄存器的 CLKEN 位、STOPCFG 位，
USART_CTRL3 寄存器的 IREN 位、HDEN 位和 SCEN 位都需要清 0

USART 在 LIN 主模式下能生成断开帧，断开帧检测长度可通过 USART_CTRL2 的 LBDLCFG 位设置为 10 位、11 位。断开帧检测电路独立于 USART 接收器，无论是空闲状态，还是数据传输状态，RX 脚都会检测到断开帧，且 USART_STS 寄存器的 LBDLFG 位置 1，若此时使能 USART_CTRL2 的 LBDIEN 位，则会产生中断。

空闲状态检测断开帧

在空闲状态下，若 RX 脚上检测到断开帧，接收器会接收到一个为 0 的数据帧并会产生 FEFLG 错误。

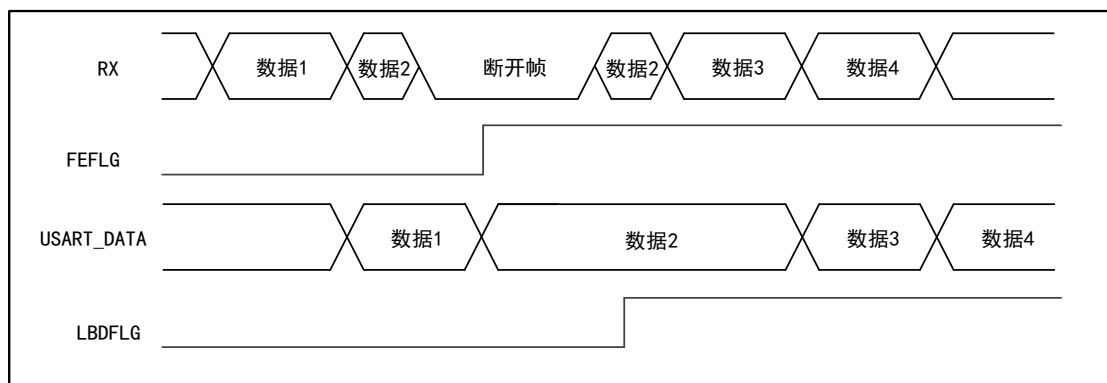
图 83 空闲状态检测断开帧



数据传输状态检测断开帧

在数据传输的过程中，若 RX 脚检测到断开帧，则当前传输的数据帧会产生 FEFLG 错误。

图 84 数据传输状态检测断开帧



21.4.12 智能卡模式

注：仅 USART1 有此功能。

智能卡模式是一种采用单线半双工通信的模式，该接口支持 ISO7816-3 标准协议，能控制读写符合该标准协议的智能卡。

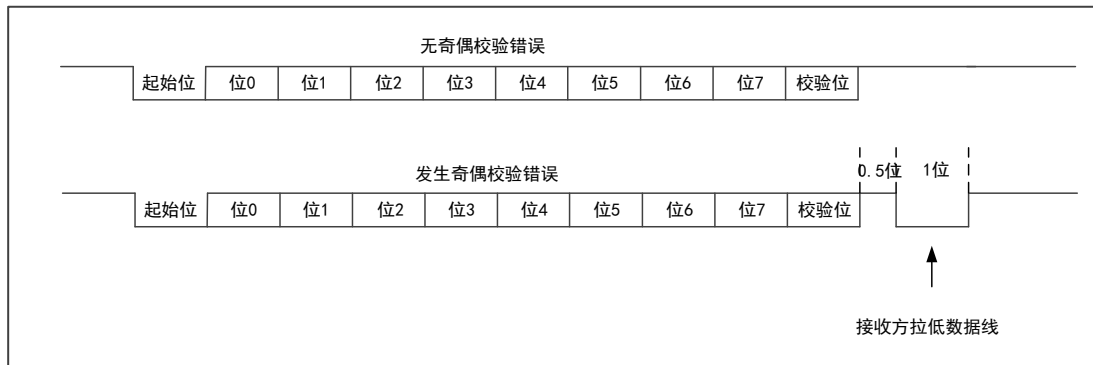
USART_CTRL3 寄存器的 SCEN 位决定是否进入智能卡模式。

当 USART 进入智能卡模式时：

- USART_CTRL2 寄存器的 LINMEN 位，USART_CTRL3 寄存器的 IREN 位、HDEN 位必须清 0
- 数据帧格式为 8 个数据位加 1 个校验位，使用 0.5 或 1.5 个停止位
- 可以置位 USART_CTRL2 寄存器的 CLKEN 位，为智能卡提供时钟

- 在通信期间，当接收方检测到奇偶校验错误时，为了通知发送方未成功接收数据，数据线会在半个波特率时钟后被拉低，并保持拉低 1 个波特率时钟
- 断开帧在智能卡模式里没有意义，一个带帧错误的 00h 数据将被当成数据而不是断开符号

图 85 ISO7816-3 标准协议



21.4.13 红外 (IrDA SIR) 功能模式

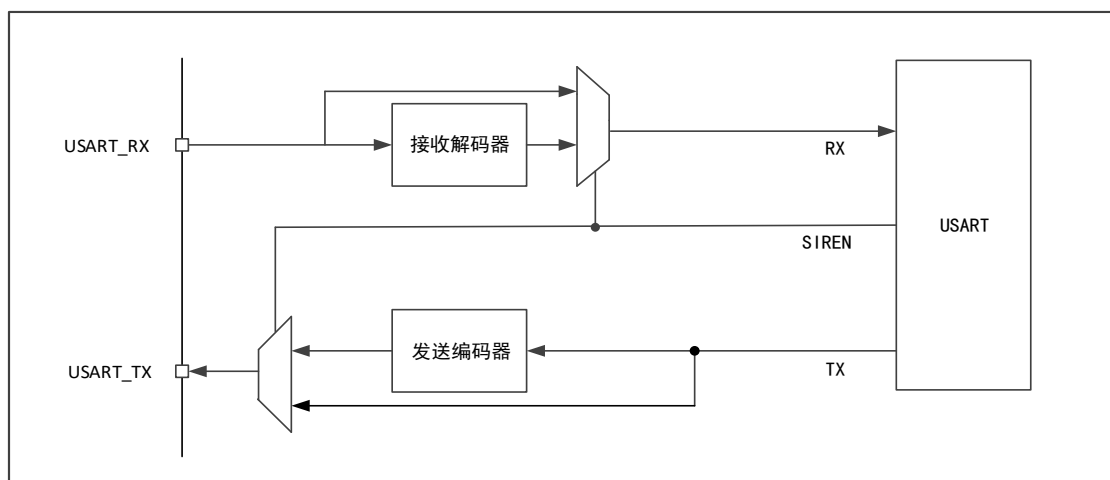
IrDA 模式是一个半双工协议，发送数据和接收数据不能同时进行，且数据发送与接收之间需要延时 10ms 以上。

USART_CTRL3 寄存器的 IREN 位决定是否进入 IrDA 模式。

当 USART 进入 IrDA 模式时：

- USART_CTRL2 寄存器的 CLKEN 位、STOPCFG 位和 LINMEN 位，USART_CTRL3 寄存器的 HDEN 位、SCEN 位都必须清 0。
- 数据帧使用 1 个停止位，波特率小于 115200Hz。
- 使用红外光脉冲 (RZI) 表示逻辑 '0'，所以在正常模式下，他的脉宽为 3/16 个波特率周期。IrDA 低功耗模式下，为确保 IrDA 正常检测到该脉冲，建议脉宽大于 3 个 DIV 分频时钟。

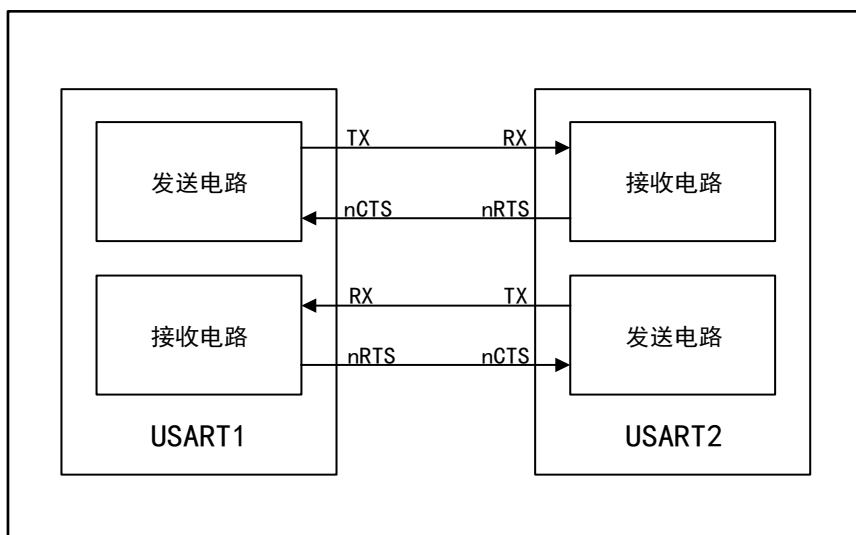
图 86 IrDA 模式框图



21.4.14 硬件流控制和 RS485 驱动使能

硬件流控制功能是通过 nCTS 引脚和 nRTS 引脚来控制两个设备间的串行数据流。

图 87 两个 USART 之间的硬件流控制



CTS 流控制

USART_CTRL3 寄存器的 CTSEN 位决定是否使能 CTS 流控制，若使能 CTS 流控制，发送器会检测 nCTS 引脚的数据帧是否可以被发送。若 USART_STS 寄存器的 TXBEFLG 位=0 且 nCTS 被拉成低电平，则数据帧可以被发送。若 nCTS 在传输期间变为高电平，那么在当前的数据帧发送完成后，发送器将停止发送。

RTS 流控制

USART_CTRL3 寄存器的 RTSEN 位决定是否使能 RTS 流控制，若使能 RTS 流控制，当接收器接收数据时，nRTS 被拉成低电平，当一个数据帧接收完成时，nRTS 变为高电平以通知发送器停止发送数据帧。

RS485 驱动使能

USART_CTRL3 寄存器的 DEN 位决定是否打开驱动使能功能，这个功能能够允许 DE 信号打开外部收发器的控制端。

提前时间：驱动使能信号和第一个字节的起始位之间的时间间隔。由 USART_CTRL1 寄存器的 DLTEN[4:0]控制。

滞后时间：最后一个字节的停止位和释放 DE 信号之间的时间间隔。由 USART_CTRL1 寄存器的 DDLTEN[4:0]控制。

21.4.15 DMA 多缓冲器通信

为了减轻处理器的负担，USART 可以用 DMA 方式访问数据缓冲区。

使用 DMA 方式发送

USART_CTRL3 寄存器上的 DMATXEN 位决定是否使用 DMA 方式发送。当用 DMA 发送时，在指定的 SRAM 区的数据会被 DMA 传输到发送缓冲区。

用 DMA 方式发送的配置步骤：

- (1) 清零 USART_STS 寄存器的 TXCFLG 标志位
- (2) 把存放数据的 SRAM 存储器的地址设置为 DMA 源地址
- (3) 把 USART_TXDATA 寄存器的地址设置为 DMA 目的地址
- (4) 设置要传输的数据字节数
- (5) 设置通道优先级
- (6) 设置中断使能
- (7) 使能 DMA 通道
- (8) 等待 USART_STS 寄存器的 TXCFLG 位置 1，表示发送完成

使用 DMA 方式接收

USART_CTRL3 寄存器的 DMARXEN 位决定是否使用 DMA 方式接收，当用 DMA 接收时，每收到一个字节，接收缓冲区的数据会被 DMA 传输到指定的 SRAM 区。

用 DMA 方式接收的配置步骤：

- (1) 把 USART_RXDATA 寄存器的地址设置为 DMA 源地址
- (2) 把存放数据的 SRAM 存储器的地址设置为 DMA 目的地址
- (3) 设置要传输的数据字节数
- (4) 设置通道优先级
- (5) 设置中断使能
- (6) 使能 DMA 通道

21.4.16 ModBus 通信

USART 支持对 ModBus/RTU 和 ModBus/ASCII 协议，ModBus/RTU 是一个半双工的块式传输协议。在软件中才能实现协议的控制部分。USART 支持结束块检测，不需要软件或其他条件。

21.4.16.1 ModBus/RTU

这个功能通过可编程的超时功能来实现。在这个模式下，一个块的结束被视为一个超过 2 个字符长度的空闲线。USART_CTRL1 寄存器的 RXTOIEN 位和 USART_CTRL2 寄存器中的 RXTODEN 位控制超时功能和相应的中断。往 USART_RXTO 寄存器中写入一个超时数字，当接收线空闲状态达到这个长度时

产生一个中断，表示块接收完毕。

21.4.16.2 Modbus/ASCII

在这个模式下，一个块的结束由一个特定的(CR/LF)字符序列识别。USART 使用字符匹配函数来管理这个机制。通过在 ADDR[7:0]字段中编程 LF 的 ASCII 码并激活该字符匹配中断(CMIEN=1)。那么当收到一个 LF 字符时，软件会被告知检查 DMA 缓冲区中的 CR/LF。

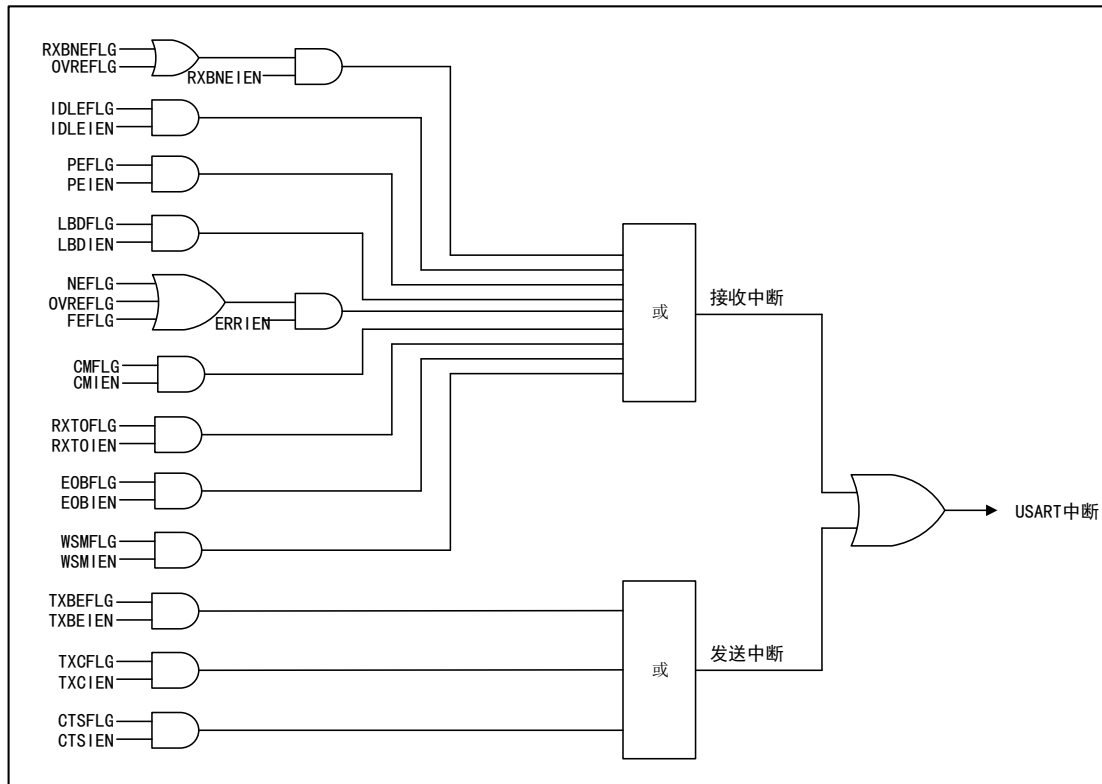
21.4.17 中断请求

表格 70 USART 中断请求

中断事件		事件标志位	使能位
接收寄存器不为空		RXBNEFLG	RXBNEIEN
过载错误		OVREFLG	
检测到线路空闲		IDLEFLG	IDLEIEN
奇偶检验错误		PEFLG	PEIEN
LIN 断开错误		LBDFLG	LBDIEN
DMA 模式下的接收错误	噪音错误	NEFLG	ERRIEN
	溢出错误	OVREFLG	
	帧错误	FEFLG	
匹配字符		CMFLG	CMIEN
未按时接收错误		RXTOFLG	RXTOIEN
检测到块结束		EOBFLG	EOBIEN
停机模式	从停机模式唤醒	WSMFLG	WSMIEN
数据发送寄存器空		TXBEFLG	TXBEIEN
发送完成		TXCFLG	TXCIEN
CTS 标志		CTSFLG	CTSIEN

USART 的中断请求都连接在同一个中断控制器上，中断请求在发送到中断控制器之前都是逻辑或关系。

图 88 USART 中断映射



21.4.18 USART 支持功能对比

表格 71 USART 支持功能对比

USART 模式	USART1	USART2
半双工（单线模式）	√	√
波特率自动检测	√	—
多处理器通信	√	√
双时钟域和从停机模式唤醒	√	—
同步	√	—
LIN	√	—
智能卡	√	—
IrDA	√	—
硬件流控制	√	√
RS485 驱动使能	√	√
多缓冲器通信（DMA）	√	√
接收超时中断	√	—
ModBus 通信	√	—

注意：

(1) “√”表示支持该功能，“—”表示不支持该功能。

21.5 寄存器地址映射

表格 72 USART 寄存器地址映射

寄存器名	描述	偏移地址
USART_CTRL1	控制寄存器 1	0x00
USART_CTRL2	控制寄存器 2	0x04
USART_CTRL3	控制寄存器 3	0x08
USART_BR	波特率寄存器	0x0C
USART_GTPSC	保护时间和预分频器寄存器	0x10
USART_RXTO	接收超时寄存器	0x14
USART_REQUEST	请求寄存器	0x18
USART_STS	中断和状态寄存器	0x1C
USART_INTFCLR	中断标志清除寄存器	0x20
USART_RXDATA	数据接收寄存器	0x24
USART_TXDATA	数据发送寄存器	0x28

21.6 寄存器功能描述

21.6.1 控制寄存器 1 (USART_CTRL1)

偏移地址: 0x00

复位值: 0x0000

位/域	名称	R/W	描述
0	UEN	R/W	使能 USART (USART Enable) 0: 禁用 USART 分频器和输出 1: 使能 USART 模块 由软件置 1 或清 0; 清除此位将取消当前的操作且 USART 的预分频器和输出立即停止工作。不会复位对 USART 的设置, 但会复位 USART_STS 中的状态标志。
1	USWMEN	R/W	使能 USART 在停机模式下唤醒 MCU (USART in Stop Mode Wake Up MCU Enable) 0: 禁止 1: 使能 由软件置 1 和清 0; 置位此位需要选择 HSICLK 或 LSECLK 作为 USART 的时钟源 (参见 RCM 章节)
2	RXEN	R/W	使能接收 (Receive Enable) 0: 禁止 1: 使能, 并开始检测 RX 引脚上的起始位 由软件置 1 或清 0。
3	TXEN	R/W	使能发送 (Transmitte Enable) 0: 禁止 1: 使能 由软件置 1 或清 0。

位/域	名称	R/W	描述
4	IDLEIEN	R/W	使能 IDLE 中断 (IDLE Interrupt Enable) 0: 禁止 1: 当 IDLEFLG 置位时产生中断 由软件置 1 或清 0。
5	RXBNEIEN	R/W	使能接收缓冲区非空中断 (Receive Buffer Not Empty Interrupt Enable) 0: 禁止 1: 当 OVREFLG 或 RXBNEFLG 置位时产生中断 由软件置 1 或清 0。
6	TXCIEN	R/W	使能发送完成中断 (Transmit Complete Interrupt Enable) 0: 禁止 1: 当 TXCFLG 置位时产生中断 由软件置 1 或清 0。
7	TXBEIEN	R/W	使能发送缓冲区空中断 (Transmit Buffer Empty Interrupt Enable) 0: 禁止 1: 当 TXBEFLG 置位时产生中断 由软件置 1 或清 0。
8	PEIEN	R/W	使能校验错误中断 (Parity Error Interrupt Enable) 0: 禁止 1: 当 PEFLG 置位时产生中断 由软件置 1 或清 0。
9	PCFG	R/W	配置奇偶校验位 (Odd/Even Parity Configure) 0: 偶校验 1: 奇校验 由软件置 1 或清 0。 需要等当前正在传输的字节完成后, 选择才会生效。 只有在未使能 USART 时才能置位此位。
10	PCEN	R/W	使能检验控制 (Parity Control Enable) 0: 禁止 1: 使能 若置位此位, 发送数据时在最高位插入一个校验位; 接收数据时, 检查接收的数据的校验位是否正确。 需要等当前正在传输的字节完成后, 校验控制才会生效。 只有在未使能 USART 时才能置位此位。
11	WUPMCFG	R/W	配置唤醒方式 (Wakeup Method Configure) 0: 空闲总线唤醒 1: 地址标记唤醒 由软件置 1 或清 0。 只有在未使能 USART 时才能置位此位。
12	DBLCFG0	R/W	配置数据位长度 (Data Bits Length Configure) 此位和 DBLCFG1 位共同决定数据位长度。 由软件置 1 或清 0。 传输数据时不能修改此位。

位/域	名称	R/W	描述
13	RXMUTEEN	R/W	使能接收静默模式 (Receive Mute Mode Enable) 0: 正常工作模式 1: 可以在正常模式和静默模式间切换 由软件置 1 或清 0。
14	CMIEN	R/W	使能字符匹配中断 (Character Match Interrupt Enable) 0: 禁止 1: 当 CMFLG 置位时产生中断 由软件置 1 或清 0。
15	OSMCFG	R/W	配置过采样模式 (Oversampling Mode Configure) 0: 16 倍过采样 1: 8 倍过采样 只有在未使能 USART 时才能置位此位。
20:16	DDLLEN[4:0]	R/W	使能驱动程序滞后时间 (Driver De-lead Time Enable) 此位域是传输时最后一个停止位和 DE 信号之间的时间间隔。它的单位是采样时间，由过采样率决定。 如果在 DDLLEN 时间内对 USART_TXDATA 寄存器进行写操作，在 DDLLEN 和 DLLEN 时间过去之后才会发送刚写入的数据。 只有在未使能 USART 时才能设置此位域。
25:21	DLLEN[4:0]	R/W	使能驱动程序提前时间 (Driver Lead Time Enable) 此位域是传输时 DE 信号和第一个起始位的时间间隔。它的单位是采样时间，由过采样率决定。 只有在未使能 USART 时才能设置此位域。
26	RXTOIEN	R/W	使能接收超时中断 (Receiver Timeout Interrupt Enable) 0: 禁止 1: 当 RXTOFLG 置位时产生中断 由软件置位或清 0。
27	EOBIEN	R/W	使能块结束中断 (End of Block Interrupt Enable) 由软件置 1 和清零。 0: 禁止 1: 当 EOBFLG 置位时，产生中断 由软件置位或清 0。
28	DBLCFG1	R/W	配置数据位长度 (Data Bits Length Configure) 此位和 DBLCFG0 位共同决定数据位长度。 DBLCFG[1:0]=00: 1 个起始位，8 个数据位，n 个停止位 DBLCFG[1:0]=01: 1 个起始位，9 个数据位，n 个停止位 DBLCFG[1:0]=10: 1 个起始位，7 个数据位，n 个停止位 由软件置 1 或清 0。 传输数据时不能修改此位。
31:29	保留		

21.6.2 控制寄存器 2 (USART_CTRL2)

偏移地址: 0x04

复位值: 0x0000

位/域	名称	R/W	描述
3:0	保留		
4	ADDRLLEN	R/W	配置地址长度 (Slave Address Length Configure) 0: 4 位地址

位/域	名称	R/W	描述
			1: 7 位地址 只有在未使能 USART 时才能设置此位域。
5	LBDLCFG	R/W	配置 LIN 断开符检测长度 (LIN Break Detection Length Configure) 0: 10 位 1: 11 位 只有在未使能 USART 时才能置位此位。
6	LBDIEN	R/W	使能 LIN 断开符检测中断 (LIN Break Detection Interrupt Enable) 0: 禁止 1: 当 LBDIFLG 位置位, 产生中断。
7	保留		
8	LBCPOEN	R/W	使能输出最后一位时钟脉冲 (Last Bit Clock Pulse Output Enable) 0: 不从 CK 输出 1: 从 CK 输出 此位只作用于同步模式。 只有在未使能 USART 时才能置位此位。
9	CPHA	R/W	配置时钟相位 (Clock Phase Configure) 此位表明在第几个时钟边沿进行采样 0: 第一个 1: 第二个 此位只作用于同步模式。 只有在未使能 USART 时才能置位此位。
10	CPOL	R/W	配置时钟极性 (Clock Polarity Configure) 当 USART 处于空闲状态时, CK 引脚的状态 0: 低电平 1: 高电平 此位只作用于同步模式。 只有在未使能 USART 时才能置位此位。
11	CLKEN	R/W	使能时钟 (CK 引脚) (Clock Enable (CK pin)) 0: 禁止 1: 使能 只有在未使能 USART 时才能置位此位。
13:12	STOPCFG	R/W	配置停止位 (STOP Bit Configure) 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 只有在未使能 USART 时才能置位此位。
14	LINMEN	R/W	使能 LIN 模式 (LIN Mode Enable) 0: 禁止 1: 使能 由软件置位或清 0。 在 LIN 模式下, 可以置位 TXBFQ 位发送和检测 LIN 同步断开符。 只有在未使能 USART 时才能置位此位。
15	SWAPEN	R/W	使能交换 TX/RX 引脚功能 (Swap TX/RX Pins Function Enable) 0: 按标准分配使用 1: TX 和 RX 的引脚功能交换使用, 与其它 USART 进行交叉互联时

位/域	名称	R/W	描述
			起作用。 由软件置位或清 0。 只有在未使能 USART 时才能置位此位。
16	RXINVEN	R/W	使能 RX 脚有效电平反向 (RX Pin Active Level Inversion Enable) 0: 标准逻辑电平(VDD =1/IDLE, Gnd=0/mark) 1: 反向(VDD =0/mark, Gnd=1/IDLE), RX 线上带有外部反相器时起作用。 由软件置位或清 0。 只有在未使能 USART 时才能置位此位。
17	TXINVEN	R/W	使能 TX 脚有效电平反向 (TX Pin Active Level Inversion Enable) 0: 标准逻辑电平(VDD =1/IDLE, Gnd=0/mark) 1: 反向(VDD =0/mark, Gnd=1/IDLE), TX 线上带有外部反相器时起作用。由软件置位或清 0。 只有在未使能 USART 时才能置位此位。
18	BINVEN	R/W	使能二进制数反向 (Binary Data Inversion Enable) 0: 正/直接逻辑(0=L,1=H) 1: 负/反向逻辑(0=H,1=L) 由软件置位或清 0。 只有在未使能 USART 时才能置位此位。 置位此位时校验位也反向。
19	MSBFEN	R/W	使能最高有效位首位传输 (Most Significant Bit First Enable) 0: 第 0 位数据紧跟着起始位 1: 最高位数据紧跟着起始位 由软件置位或清 0。 只有在未使能 USART 时才能置位此位。
20	ABRDEN	R/W	使能自动波特率检测 (Auto Baud Rate Detection Enable) 0: 禁止 1: 使能 由软件置位或清 0。
22:21	ABRDCFG	R/W	配置自动波特率检测模式 (Auto Baud Rate Detection Mode Configure) 00: 测量起始位 01: 测量下降沿 10: 0x7F 帧检测 11: 0x55 帧检测 由软件置位或清 0。
23	RXTODEN	R/W	使能接收超时检测功能 (Receive Timeout Detection Function Enable) 0: 禁止 1: 使能 由软件置位或清 0。 置位此位, 当 RX 线检测到空闲达到 RXTO 寄存器配置的时间长度后, RXTOFLG 位被硬件置位。
27:24	ADDRL	R/W	设置 USART 设备节点地址低位 (USART Device Node Address Low Setup) 此位域作用于多机通讯且进入静默状态或者停机模式时的 7 位地址标记唤醒检测。 只有在关闭接收器或未使能 USART 时才能置位此位。

位/域	名称	R/W	描述
31:28	ADDRH	R/W	<p>设置 USART 设备节点地址高位（USART Device Node Address High Setup）</p> <p>此位域不仅作用于多机通讯且进入静默状态或者停机模式时的 7 位地址标记唤醒检测。（发送器的字符最高位应该为 1）也用于正常接收过程中的字符检测。（此时禁用静默状态）此时，若接收到的 8 位字节与 ADDRH 匹配，则 CMFLG 位被置位。</p> <p>只有在关闭接收器或未使能 USART 时才能置位此位。</p>

21.6.3 控制寄存器 3（USART_CTRL3）

偏移地址：0x08

复位值：0x0000

位/域	名称	R/W	描述
0	ERRIEN	R/W	<p>使能错误中断（Error Interrupt Enable）</p> <p>0：禁止</p> <p>1：使能，当 FEFLG、OVREFLAG 或 NEFLAG 其中一个置位时，产生中断。</p>
1	IREN	R/W	<p>使能红外功能（IrDA Function Enable）</p> <p>0：禁止</p> <p>1：使能</p> <p>由软件置位或清 0。</p> <p>只有在未使能 USART 时才能置位此位。</p>
2	IRLPEN	R/W	<p>使能红外低功耗模式（IrDA Low-power Mode Enable）</p> <p>0：普通模式</p> <p>1：低功耗模式</p> <p>只有在未使能 USART 时才能置位此位。</p>
3	HDEN	R/W	<p>使能半双工模式（Half-duplex Mode Enable）</p> <p>0：禁止</p> <p>1：使能</p> <p>只有在未使能 USART 时才能置位此位。</p>
4	SCNACKEN	R/W	<p>在智能卡功能下，使能出现校验错误时传输 NACK（NACK Transmit Enable during parity error in Smartcard Function）</p> <p>0：不发送 NACK</p> <p>1：发送 NACK</p> <p>只有在未使能 USART 时才能置位此位。</p>
5	SCEN	R/W	<p>使能智能卡功能（Smartcard Function Enable）</p> <p>0：禁止</p> <p>1：使能</p> <p>只有在未使能 USART 时才能置位此位。</p>
6	DMARXEN	R/W	<p>使能 DMA 接收（DMA Receive Enable）</p> <p>0：禁止</p> <p>1：使能</p> <p>由软件置位或清 0。</p>
7	DMATXEN	R/W	<p>使能 DMA 发送（DMA Transmit Enable）</p> <p>0：禁止</p> <p>1：使能</p> <p>由软件置位或清 0。</p>

位/域	名称	R/W	描述
8	RTSEN	R/W	<p>使能 RTS 功能（RTS Function Enable）</p> <p>0: 禁止</p> <p>1: 使能 RTS 中断</p> <p>RTS: Require To Send 发送请求，为输出信号，说明已准备好接收。</p> <p>当接收缓冲区有空间时才请求接收数据；当可以接收数据时，RTS 输出被拉至低电平。</p> <p>只有在未使能 USART 时才能置位此位。</p>
9	CTSEN	R/W	<p>使能 CTS 功能（CTS Function Enable）</p> <p>0: 禁止</p> <p>1: 使能</p> <p>CTS: Clear To Send 发送清除，为输入信号</p> <p>当 CTS 输入信号为低电平时，才能发送数据，否则无法发送数据；若在传输数据时，CTS 信号被拉高，那么当这个数据发送完成后，将停止数据传输；如果在 CTS 为高电平时对数据寄存器进行写操作，则要等 CTS 有效时才会将这个数据发送出去。</p> <p>只有在未使能 USART 时才能置位此位。</p>
10	CTSIEN	R/W	<p>使能 CTS 中断（CTS Interrupt Enable）</p> <p>0: 禁止</p> <p>1: 当 CTSFLG 置位时产生中断</p>
11	SAMCFG	R/W	<p>配置采样方法（Sample Method Configure）</p> <p>0: 三次采样</p> <p>1: 单次采样，禁止噪音检测标志</p> <p>只有在未使能 USART 时才能置位此位。</p>
12	OVRDEDIS	R/W	<p>禁止溢出检测（Overrun Detection Disable）</p> <p>0: 使能。当 RXBNEFLG 位被置位且接收到新数据时，OVREFLG 位被置位。</p> <p>1: 禁止。若接收到新数据时 RXBNEFLG 仍然置位，但 OVREFLG 没有被置位，那么还未被读出的数据会被新数据覆盖。</p> <p>只有在未使能 USART 时才能置位此位。</p>
13	DDISRXEEN	R/W	<p>使能在接收错误时禁用 DMA（DMA Disable on Receive Error Enable）</p> <p>0: 不禁用 DMA。对应的错误标志位被置位，但为了避免数据溢出覆盖，RXBNEFLG 不会被置位。在智能卡模式下，作为结果，不会发出 DMA 请求，所以不会传输错误的的数据，但会传输下一个正确的数据。</p> <p>1: 禁用 DMA。若 RXBNEFLG 置位，那么对应的错误标志位也会被置位。直到对应的错误标志位清零，DMA 请求才会解除屏蔽。所以需要先禁止 DMA 请求或在先清除 RXBNEFLG 标志再清除错误标志。</p> <p>只有在未使能 USART 时才能置位此位。</p>
14	DEN	R/W	<p>使能驱动器（Driver Enable）</p> <p>允许用户通过 DE 信号来激活外部收发器的控制端。</p> <p>0: 禁止 DE 功能</p> <p>1: 使能 DE 功能，在 RTS 脚输出 DE 信号</p> <p>只有在未使能 USART 时才能置位此位。</p>
15	DPCFG	R/W	<p>配置驱动输出脚极性（Driver Polarity Configure）</p> <p>0: DE 信号高电平有效</p> <p>1: DE 信号低电平有效</p> <p>只有在未使能 USART 时才能置位此位。</p>

位/域	名称	R/W	描述
16	保留		
19:17	SCARCCFG	R/W	配置智能卡模式自动重试计数器 (Smartcard Mode Auto-retry Count Configure) 0x0: 关闭重发功能, 在发送模式下不自动重发。 0x1~0x7: 自动重试次数 发送模式: 在产生发送错误前自动重发数据的次数。 接收模式: 在产生接收错误前自动重试错误接收的次数。 只有在未使能 USART 时才能置位此位。
21:20	WSIFLGSEL	R/W	选择从停机模式唤醒中断标志的方法 (Wakeup From Stop Mode Interrupt Flag Select) 00: 地址匹配时 01: 保留 10: 检测到起始位时 11: 接收数据寄存器非空时 只有在未使能 USART 时才能置位此位。
22	WSMIEN	R/W	使能从停机模式唤醒中断 (Wakeup from Stop mode interrupt enable) 0: 禁止 1: 当 WSMFLG 置位时产生中断 由软件置位或清 0。
31:23	保留		

21.6.4 波特率寄存器 (USART_BR)

只有在未使能 USART 时才能设置此寄存器。在自动波特率检测时, 可能会被硬件置位。

偏移地址: 0x0C

复位值: 0x0000

位/域	名称	R/W	描述
3:0	FBR	R/W	USART 波特率分频系数的小数部分 (Fraction of USART Baud Rate Divider factor) USART 波特率分频系数的小数部分由这 4 位决定。
15:4	IBR	R/W	USART 波特率分频系数的整数部分 (Integer of USART Baud Rate Divider factor) USART 波特率分频系数的整数部分由这 12 位决定。
31:16	保留		

21.6.5 保护时间和预分频器寄存器 (USART_GTPSC)

偏移地址: 0x10

复位值: 0x0000

位/域	名称	R/W	描述
7:0	PSC	R/W	设置预分频系数 (Prescaler Factor Setup) 对系统时钟进行分频提供时钟; 在不同的工作模式下, PSC 的有效位存在差异, 具体如下: 在红外低功耗模式下: PSC[7:0]有效。 00000000: 保留

位/域	名称	R/W	描述
			00000001: 1 分频 00000010: 2 分频 11111111: 255 分频 在红外的正常模式下: PSC 只能设置为 00000001 在智能卡模式下: PSC[7:5]无效, PSC[4:0]有效 00000: 保留 00001: 2 分频 00010: 4 分频 00011: 6 分频 11111: 62 分频 智能卡模式下不使用位[7:5]。 只有在未使能 USART 时才能置位此位。
15:8	GRDT	R/W	设置保护时间值 (Guard Time Value Setup) 在发送数据完成后, 需等待保护时间才将 TXCFLG 置位; 时间单位是波特时钟; 可应用于智能卡模式。 只有在未使能 USART 时才能置位此位。
31:16			保留

21.6.6 接收超时寄存器 (USART_RXTO)

偏移地址: 0x14

复位值: 0x0000

位/域	名称	R/W	描述
23:0	RXTO	R/W	设置接收超时值 (Receiver Timeout Value Setup) 此位域规定了以波特时钟为单位的接收超时值。 在标准模式下, 接收到最后一个字节后, 若在 RXTO 值的时长内没有检测到新的起始位时, RXTOFLG 被硬件置位。 在智能卡模式下, 这个值用来实现 CWT 和 BWT。在此模式下, 从最后一个字节的起始位开始超时测量。
31:24	BLEN[7:0]	R/W	设置块长度 (Block Length Setup) 此位域规定了智能卡模式 T=1 接收时的块长度, 这个值是信息块字符数+结束部分 (1-LEC/2-CRC) -1。 如: BLEN =0->0 个信息字符+LEC BLEN =1->0 个信息字符+CRC BLEN =255-> 254 个信息字符+CRC 智能卡模式下, 当 TXBEFLG=0 会清除块长度计数器。 块长度计数器在 RXEN=0 时或在 EOBCFLG 位被置位时清零。

21.6.7 请求寄存器 (USART_REQUEST)

偏移地址: 0x18

复位值: 0x0000

位/域	名称	R/W	描述
0	ABRDQ	W	请求自动波特率检测 (Auto Baud Rate Detection Request) 置位此位会清除 ABRDFLG 标志, 并在下一次接收数据时进行一次自动波特率检测。
1	TXBFQ	W	请求发送断开帧 (Transmit Break Frame Request) 置位此位会置位 TXBFFLG 标志, 并在使能发送状态机后发送一个断开帧。
2	MUTEQ	W	请求进入静默模式 (Mute Mode Request) 置位此位会进入静默模式, 并清除 RXWFMUTE 标志。
3	RXDFQ	W	请求接收数据清空 (Receive Data Flush Request) 置位此位会清除 RXBNEFLG 标志。 可以丢弃接收寄存器内还没有被读出的数据, 以免产生溢出错误
4	TXDFQ	W	请求发送数据清空 (Transmit Data Flush Request) 置位此位会置位 TX 标志。 可以取消数据发送。此位作用于当因为错误中断还未发送数据, FEFLG 标志置位时。 此位只作用于智能卡模式
31:5	保留		

21.6.8 中断和状态寄存器 (USART_STS)

偏移地址: 0x1C

复位值: 0x0200 00C0

位/域	名称	R/W	描述
0	PEFLG	R	发生校验错误标志 (Parity Error Occur Flag) 0: 无错误 1: 检测到校验错误 在接收模式中, 当出现奇偶校验错误时, 由硬件置 1; 置位 PECLR 可清除此位。
1	FEFLG	R	发生帧错误标志 (Frame Error Occur Flag) 0: 无帧错误 1: 检测到帧错误或断开符 当出现同步错位、过多噪音或断开符时, 由硬件置 1; 置位 FECLR 可清除此位。
2	NEFLG	R	发生噪音错误标志 (Noise Error Occur Flag) 0: 无噪音 1: 检测到噪音 当出现噪音错误时, 由硬件置 1; 置位 NFCLR 可清除此位。
3	OVREFLG	R	发生溢出错误标志 (Overrun Error Occur Flag) 0: 无溢出错误 1: 检测到溢出错误 当 RXBNEFLG 位被置位, 且移位寄存器中的数据要传输到接收寄存器时, 由硬件置 1; 置位 OVRECLR 可清除此位。
4	IDLEFLG	R	检测到空闲总线标志 (IDLE Line Detected Flag) 0: 未检测到空闲总线 1: 检测到空闲总线 当检测到空闲总线时, 由硬件置 1; 置位 IDLECLR 可清除此位。

位/域	名称	R/W	描述
5	RXBNEFLG	R	接收数据缓冲器不为空标志 (Receive Data Buffer Not Empty Flag) 0: 接收数据缓冲器为空 1: 接收数据缓冲器不为空 当数据寄存器接收到接收移位寄存器传输的数据时, 由硬件置 1; 对 TXDATA 寄存器执行读操作或置位 RXDFQ 可清除此位。
6	TXCFLG	R	发送数据完成标志 (Transmit Data Complete Flag) 0: 发送数据未完成 1: 发送数据完成 当数据的最后一帧发送完成且 TXBEFLG 置位时, 由硬件置 1; 对 TXDATA 寄存器进行写操作或置位 TXCCLR 可清除此位。
7	TXBEFLG	R	发送数据缓冲器为空标志 (Transmit Data Buffer Empty Flag) 0: 发送数据缓冲器不为空 1: 发送数据缓冲器为空 当移位寄存器接收到发送数据寄存器传输的数据时, 由硬件置 1; 对 TXDATA 寄存器进行写操作可清除此位。
8	LBDFLG	R	检测到 LIN 断开标志 (LIN Break Detected Flag) 0: 没有检测到 LIN 断开 1: 检测到 LIN 断开 当检测到 LIN 断开时, 由硬件置 1; 置位 LBDCLR 可清除此位。 若 USART_CTRL2 中的 LBDIEN 置位, 则产生一个中断。
9	CTSFLG	R	CTS 变化标志 (CTS Change Flag) 0: nCTS 状态线上没有变化 1: nCTS 状态线上发生变化 若 CTSEN 位置位, 当 nCTS 输入切换时, 由硬件置 1; 置位 CTSCCLR 可清除此位。
10	CTSCFG	R	配置 CTS 状态 (CTS Status Configure) 0: 设置 nCTS 线 1: 复位 nCTS 线 由硬件置 1 和清 0。 此位设置了 nCTS 输入脚的反向状态。
11	RXTOFLG	R	接收超时标志 (Receiver Timeout Flag) 0: 未超时 1: 已超时 若在 RXTO 位设置的时长内没有检测到起始位, 由硬件置 1; 置位 RXTOCLR 位可清除此位。
12	EOBFLG	R	块结束标志 (End of Block Flag) 0: 未到块结束 1: 到达块结束 当接收到一个完整的块时由硬件置 1; 置位 EOBCLR 位可清除此位。 当接收到的字节数达到 BLEN+4 时完成检测。 若置位 EOBIEN 位, 则产生一个中断。
13	保留		
14	ABRDEFLG	R	自动波特率检测错误标志 (Auto Baud Rate Detection Error Flag)

位/域	名称	R/W	描述
			在波特率检测失败时，由硬件置 1；置位 ABRDQ 位可清除此位。
15	ABRDFLG	R	自动波特率检测标志（Auto Baud Rate Detection Flag） 当打开自动波特率功能或当自动波特率操作中中断时，由硬件置 1；重新开始波特率检测时清除此位。
16	BSYFLG	R	忙碌标志（Busy Flag） 0：空闲状态 1：正在接收数据 当检测到起始位时，由硬件置 1，接收结束后被清除。 由硬件置 1 和清 0。
17	CMFLG	R	字符匹配标志（Character Match Flag） 0：无字符匹配 1：有字符匹配 当接收到的字符和 ADDR[7:0]设置的值匹配时，由硬件置 1；置位 CMCLR 位可清除此位。
18	TXBFFLG	R	发送断开帧标志（Transmit Break Frame Flag） 0：未发送 1：将要发送 若置位 TXBFQ 位，由软件置 1；在发送断开帧的停止位时，由硬件清 0。
19	RXWFMUTE	R	从静默模式唤醒接收器（Receiver Wakeup From Mute Mode） 0：正常模式 1：静默模式 当切换唤醒模式和静默模式时，由硬件置 1 和清 0；如果由空闲信号唤醒，向 USART_REQUEST 寄存器进行写操作可以将此位置 1。 WUPMCFG 位决定静默模式控制顺序。
20	WSMFLG	R	从停机模式唤醒标志（Wakeup From Stop Mode Flag） 0：未检测到 1：检测到 置位 PECLR 位可清除此位。 若置位 WSMFLG 位，则产生一个中断。
21	TXENACKFLG	R	发送使能应答标志（Transmit Enable Acknowledge Flag） 当读到发送使能信号时，由硬件置 1。 当 TXEN=0 产生空闲帧请求，为了确保 TXEN=0 的最小周期，会紧接着将 TXEN 置位。
22	RXENACKFLG	R	接收使能应答标志（Receive Enable Acknowledge Flag） 当读到接收使能信号时，由硬件置 1。 此位用来确认 USART 在进入停机模式前，是否已准备好接收数据。
31:23	保留		

21.6.9 中断标志清除寄存器（USART_INTFCLR）

偏移地址：0x20

复位值：0x0000

位/域	名称	R/W	描述
0	PECLR	RC_W1	清除校验错误标志 (Parity Error Flag Clear) 置位此位可以清除 USART_STS 寄存器的 PEFLG 标志位。
1	FECLR	RC_W1	清除帧错误标志 (Framing Error Flag Clear) 置位此位可以清除 USART_STS 寄存器的 FEFLG 标志位。
2	NECLR	RC_W1	清除噪音检测标志 (Noise Detected Flag Clear) 置位此位可以清除 USART_STS 寄存器的 NEFLG 标志位。
3	OVRECLR	RC_W1	清除溢出错误标志 (Overrun Error Flag Clear) 置位此位可以清除 USART_STS 寄存器的 OVREFLG 标志位。
4	IDLECLR	RC_W1	清除空闲总线检测标志 (IDLE Line Detected Clear Flag) 置位此位可以清除 USART_STS 寄存器的 IDLEFLG 标志位。
5	保留		
6	TXCCLR	RC_W1	清除发送数据完成标志 (Transmission Data Complete Flag Clear) 置位此位可以清除 USART_STS 寄存器的 TXCFLG 标志位。
7	保留		
8	LBDCLR	RC_W1	清除 LIN 断开检测标志 (LIN Break Detection Flag Clear) 置位此位可以清除 USART_STS 寄存器的 LBDFLG 标志位。
9	CTSCLR	RC_W1	清除 CTS 变化标志 (CTS Flag Clear) 置位此位可以清除 USART_STS 寄存器的 CTSFLG 标志位。
10	保留		
11	RXTOCLR	RC_W1	清除接收超时标志 (Receiver Timeout Flag Clear) 置位此位可以清除 USART_STS 寄存器的 RXTOFLG 标志位。
12	EOBCLR	RC_W1	清除块结束标志 (End of Block Flag Clear) 置位此位可以清除 USART_STS 寄存器的 EOBFLG 标志位。
16:13	保留		
17	CMCLR	RC_W1	清除字符匹配标志 (Character Match Flag Clear) 置位此位可以清除 USART_STS 寄存器的 CMFLG 标志位。
19:18	保留		
20	WSMCLR	RC_W1	清除从停机模式唤醒标志 (Wakeup From Stop Mode Flag Clear) 置位此位可以清除 USART_STS 寄存器的 WSMFLG 标志位。
31:21	保留		

21.6.10 数据接收寄存器 (USART_RXDATA)

偏移地址: 0x24

复位值: 0xFFFF

位/域	名称	R/W	描述
8:0	RXDATA	R	设置接收数据的值 (Receive Data Value Setup) 包含接收到的数据字节。 提供输入移位寄存器和内部总线之间的并行接口。 若在接收数据时打开校验位, 对此寄存器进行读操作, 最高位是校验位。
31:9	保留		

21.6.11 数据发送寄存器 (USART_TXDATA)

偏移地址: 0x28

复位值: 0xXXXX

位/域	名称	R/W	描述
8:0	TXDATA	R/W	<p>设置发送数据的值 (Transmit Data Value Setup)</p> <p>包含要发送的数据字节。</p> <p>提供发送移位寄存器和内部总线之间的并行接口。</p> <p>若在发送数据时打开校验位, 那么向最高位进行写操作无效, 因为它会被校验位取代之后再发送。</p>
31:9	保留		

22 内部集成电路接口 (I2C)

22.1 术语全称、缩写描述

表格 73 术语全称、缩写描述

中文全称	英文全称	英文缩写
串行数据线	Serial Data	SDA
串行时钟线	Serial Clock	SCL
系统管理总线	System Management Bus	SMBus
时钟	Clock	CLK
高速串行时钟	Serial Clock High	SCLH
低速串行时钟	Serial Clock Low	SCLL
地址解析协议	Address Resolution Protocol	ARP
否认应答	Negative Acknowledgement	NACK
应答	Acknowledgement	ACK
数据包出错检测	Packet Error Checking	PEC

22.2 简介

I2C 是一种短距离总线通信协议，物理实现上，I2C 总线由两根信号线（SDA 与 SCL）和一个地线组成，两根信号线为双向传输的。

- 两根信号线,SCL 时钟线、SDA 数据线。由 SCL 为 SDA 提供时序，SDA 串行发送/接收数据
- SCL、SDA 这两根信号线均为双向
- 两个系统使用 I2C 总线通信时共地

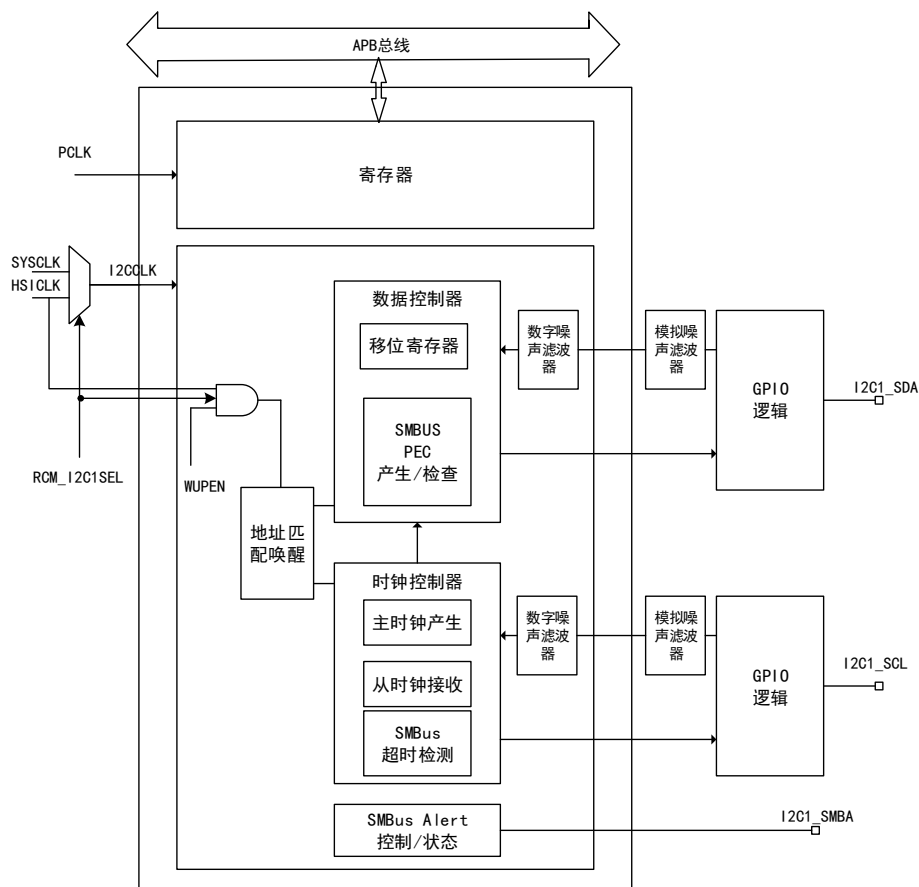
22.3 主要特征

- (1) 可选择主机或从机模式
- (2) 多主机功能
- (3) 7 位和 10 位寻址模式
- (4) 响应广播
- (5) 多个 7 位从地址
- (6) 三种模式
 - 标准模式
 - 快速模式
 - 超快速模式

- (7) 可编程的时钟延长
- (8) 软件复位
- (9) 可编程的开始时间和保持时间
- (10) DMA 功能
- (11) 可编程的噪音滤波器
- (12) SMBus 特定功能
 - 硬件 PEC
 - 命令接收与数据应答控制
 - 地址解析协议
 - HOST 通知协议
 - SMBus 报警
 - SMBus 超时管理
- (13) 可选择一个独立的时钟源
- (14) 从停机模式唤醒

22.4 结构框图

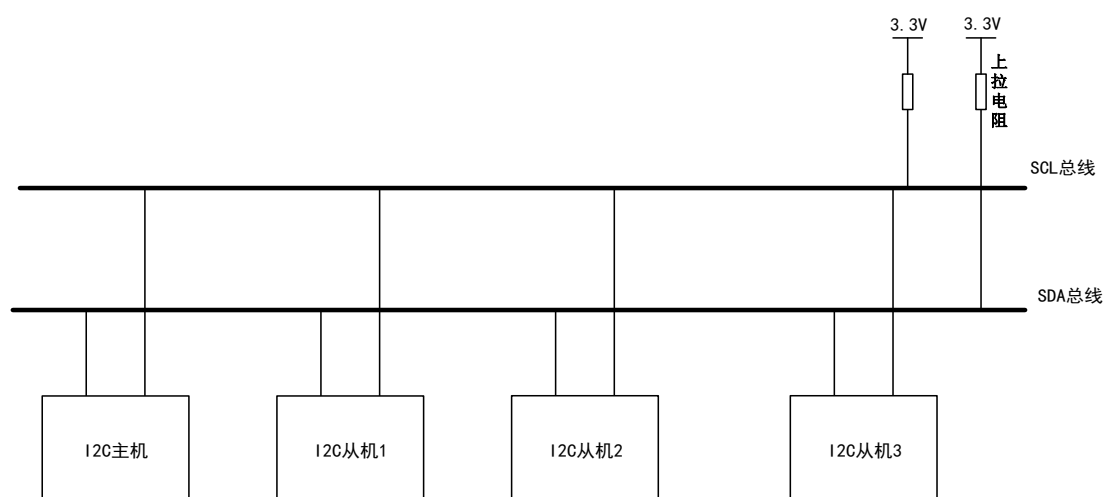
图 89 I2C1 功能结构图



22.5 功能描述

22.5.1 I2C 物理层

图 90 常见 I2C 通信连接图



物理层特点

- (1) 支持多设备的总线（多个设备共用的信号线），在 I2C 通信总线中，可连接多个通信主机与通信从机。
- (2) 一个 I2C 总线只使用两条总线线路,一条双向串行数据线（SDA），一条串行时钟线（SCL）。数据线用于传输数据，时钟线用于数据收发同步。
- (3) 每一个连接在总线上的设备都具有独立的地址（七位或十位），主机根据设备地址寻址访问从设备。
- (4) 总线需要接上拉电阻到电源，I2C 总线空闲状态下，输出为高阻态，所有设备空闲时，都输出高阻态，上拉电阻把总线拉成高电平。
- (5) 三种通信模式：标准模式（高达 100KHz）、快速模式（高达 400KHz）、超快速模式（高达 1MHz）。
- (6) 多主机同时使用总线时，防止数据冲突，采取总线仲裁方式决定哪个设备占用总线。
- (7) 可编程建立和保持时间,可对 I2C 中 SCL 的高电平时间与低电平时间进行编程。

22.5.2 I2C 协议层

协议层特点

- (1) 数据以帧的形式传输，每一帧中由 1 个字节（8 位）组成。

- (2) 在 SCL 的上升沿阶段，SDA 需要保持稳定，SDA 在 SCL 为低期间作出改变。
- (3) 除了数据帧，I2C 总线还有起始位,停止位,应答位。
- 起始位：在 SCL 为稳定的高电平期间，SDA 的一个下降沿启动传输。
 - 停止位：在 SCL 为稳定的高电平期间，SDA 的一个上升沿停止传输。
 - 应答位：用于表示一个字节传输成功。总线发送器（无论主机还是从机），在发送 8 个位的数据后，SDA 将释放（由输出变为输入），在第九个时钟脉冲期间，接收器将 SDA 拉低，来应答接收到了数据。

I2C 通信读写过程

图 91 主机写数据到从机

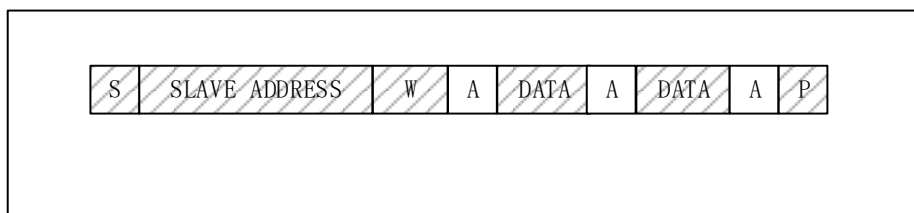
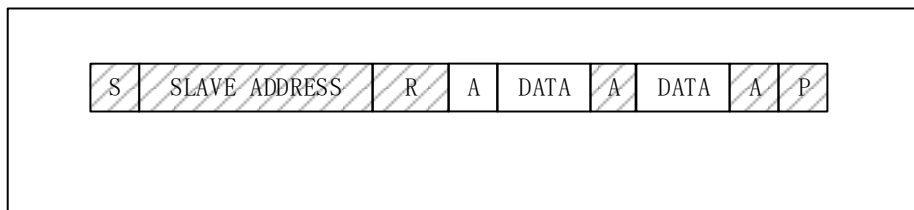




图 92 主机由从机读取数据



注：

- (1) ：此数据由主机传输到从机
- (2) S：起始信号
- (3) SLAVE ADDRESS：从机地址
- (4) ：此数据由从机传输到主机
- (5) R/W：传输方向选择位
- (6) 1 为读取 0 为写入
- (7) P：停止信号

起始信号产生后，所有从机都将等待主机发送的从机地址信号，I2C 总线中，每个设备的地址都是唯一的，当地址信号与设备地址匹配后,从机将被选中，没被选中的从机将忽略以后的数据信号。

传输方向为写数据时

广播完地址后，接收到应答信号，主机向从机发传输数据，数据长度为一个字节，主机每次发完一个字节数据后，都需等待从机发送的应答信号，当传输的所有字节完成后，主机向从机发送一个停止信号，表示为传输完成。

传输方向为读数据时

广播完地址后，接收到应答信号，从机开始向主机传输数据,数据包的大小为 8 位，从机每发送完一个字节数据,都要等待主机的应答信号，当主机想停止接收数据时，需要向从机返回一个非应答信号，则从机自动停止数据传输。

22.5.3 I2C 时钟介绍

22.5.3.1 I2C 时钟源

I2C 由一个独立的时钟源驱动,可让 I2C1 独立于 PCLK 频率独立运作。

I2C 时钟源可选择 HSICLK 或 SYSCLK。

22.5.3.2 I2C 时钟要求

- (1) $t_{I2C_CLK} < (t_{low} - t_{filters}) / 4$ 并且 $t_{I2C_CLK} < t_{HIGH}$
- (2) t_{low} : SCL 低电平时间
- (3) t_{HIGH} : SCL 高电平时间
- (4) $t_{filters}$: I2C 启动时，模拟滤波器与数字滤波器所带来的滞后总和

I2C 时钟配置

启动外设前，需要配置 I2C_TIMING 寄存器中的 SCLH 与 SCLL 位来配置 I2C 时钟。

其中可实现时钟同步机制，支持多主机环境与从机时钟延长。

$$t_{SCL} = t_{SYNC1} + t_{SYNC2} + \{ ((SCLH+1) + (SCLL+1)) * (TIMINGPSC+1) * t_{I2C_CLK} \}$$

t_{SYNC1} 取决于：

- SCL 下降斜率
- 模拟滤波器的输入延迟
- 数字滤波器的输入延迟
- SCL 的同步 I2C_CLK 时钟带来的延迟

t_{SYNC2} 取决于：

- SCL 上升斜率
- 模拟滤波器的输入延迟
- 数字滤波器的输入延迟
- SCL 的同步 I2C_CLK 时钟带来的延迟

为了满足 I2C 与 SMBus 模式兼容，时钟时序要求如下表：

表格 74 时钟时序要求

符号	参数	标准模式		快速模式		超快速模式		SMBus		单位
		最小值	最大值	最小值	最大值	最小值	最大值	最小值	最大值	
f _{SCL}	SCL 时钟频率	-	100	-	400	-	1000	-	100	KHz
t _{HD:STA}	START 信号保持时间	4	-	0.6	-	0.26	-	4.0	-	μs
t _{SU:STA}	START 信号建立时间	5	-	0.6	-	0.26	-	4.7	-	μs
t _{SU:STO}	STOP 信号建立时间	4	-	0.6	-	0.26	-	4.7	-	μs
t _{BUF}	在 STOP 与 START 信号之间总线空闲时间	5	-	1.3	-	0.50	-	4.0	-	μs
t _{LOW}	SCL 时钟低电平时间	8	-	1.3	-	0.50	-	4.7	-	μs
t _{HIGH}	SCL 时钟高电平时间	4	-	0.6	-	0.26	-	4.0	50	μs
t _r	SDA 与 SCL 信号的上升沿时间	-	1000	-	300	-	120	-	1000	ns
t _f	SDA 与 SCL 信号的下降沿时间	-	300	-	300	--	120	-	300	ns

22.5.3.3 I2C_TIMING 寄存器配置

表格 75 I2C_TIMING 寄存器配置

f _{I2C_CLK} =48MHz				
参数	标准模式		快速模式	超快速模式
	10 KHz	100 KHz	400 KHz	1 MHz
TIMINGPSC	0xB	0xB	5	5
SCLL	0xC7	0x13	0x9	0x3
t _{SCLL}	200x250ns=50μs	20x250ns= 5.0μs	10x125ns = 1250 ns	4x125 ns =500 ns
SCLH	0xC3	0xF	0x3	0x1
t _{SCLH}	196x250 ns = 49μs	16x250 ns = 4.0μs	4x125 ns = 500 ns	2x125 ns = 250 ns
t _{SCL}	100μs	10μs	2500ns	875ns
DATAHT	0x2	0x2	0x3	0
t _{DATAHT}	2x250 ns = 500 ns	2x250 ns = 500 ns	3x125 ns = 375 ns	0ns
DATAT	0x4	0x4	0x3	0x1
t _{DATAT}	5x250 ns = 1250 ns	5 x250 ns = 1250 ns	4x125 ns = 500 ns	2x125 ns = 250 ns

注:

$$t_{I2C_CLK} = 1 / f_{I2C_CLK}$$

$$t_{TIMINGPSC} = (TIMINGPSC+1) \times t_{I2C_CLK}$$

$$t_{\text{DATAT}} = (\text{DATAT} + 1) \times t_{\text{TIMINGPSC}}$$

$$t_{\text{DATAHT}} = \text{DATAHT} \times t_{\text{TIMINGPSC}}$$

$$t_{\text{SCLH}} = (\text{SCLH} + 1) \times t_{\text{TIMINGPSC}}$$

$$t_{\text{SCLL}} = (\text{SCLL} + 1) \times t_{\text{TIMINGPSC}}$$

22.5.4 I2C 功能配置模式

接口可配置为以下模式:

- 从机发送
- 从机接收
- 主机发送
- 主机接收

当 I2C 接口初始状态下, 工作模式为从机模式。当 I2C 接口发送起始信号后, 将自动从从机模式转换为主机模式。

22.5.4.1 软件复位

软件复位是通过设置 I2C_CTRL1 寄存器的 SWRST 位来实现的。此时释放 SCL 和 SDA 线, 内部状态机复位, 所有通信控制位和状态位回到其复位值, 配置寄存器不受影响。

22.5.4.2 从机模式

从模式发送

由主机发送起始信号与地址后, 寻址成功, 清除 ADDRMLG 位, 发送器将把待发送数据从 I2C_TXDATA 寄存器由内部移位寄存器发送到 SDA 线上。

从机每发送一字节, 都会等待主机的应答信号 (ACK), 重复这个过程, 直到主机希望停止接收数据时, 向从机返回一个非应答信号 (NACK), 此时从机将停止数据传输。

从模式接收

接收到主机的地址后, 清除 ADDRMLG 位, 从机通过内部移位寄存器从 SDA 线接收到的数据存储到 I2C_RXDATA 寄存器中。

从机接收到一个字节后, 都会向主机发送一个应答信号 (ACK), 直至主机发送停止信号后, 传输结束。

从时钟延长

在默认模式下, I2C 从机在下列情况下会拉低 SCL 时钟:

- 接收到的地址与启用的从机地址匹配, 拉低 SCL 时钟直至 ADDRMLG 标志被软件清零后释放。清除 ADDRMLG 标志位方法是 ADDRMLR 位置 1。

- 在发送时，如果之前的数据已被传输完毕，然后没有新的数据写入到 I2C_TXDATA 寄存器中，或者 ADDR_MFLG 标志被清除，I2C_TXDATA 寄存器没有写入字节，将拉低 SCL 时钟，直至 I2C_TXDATA 寄存器由数据写入，释放 SCL 时钟。
- 在接收时如果 I2C_RXDATA 寄存器内容未被读走，又有新的数据收到，拉低 SCL 时钟，直到 I2C_RXDATA 寄存器被读取后，释放 SCL 时钟。

22.5.4.3 主机模式

主机发送

I2C 接口发送起始信号并通过内部移位寄存器将地址送到 SDA 线上，传输方向为写，等待从机应答，当从机应答后，主机通过内部移位寄存器将字节从 I2C_TXDATA 寄存器发送到 SDA 线上，并等待从机发送的应答信号（ACK），如此反复，当 I2C_TXDATA 寄存器写入最后一个字节后，将设置 STOP 位产生一个停止信号。

主机接收

I2C 接口发送起始信号并通过内部移位寄存器将地址送到 SDA 线上，传输方向为读，从机应答后，主机进入接收模式，主机通过内部移位寄存器接收 SDA 线上的数据送至 I2C_RXDATA 寄存器中，当主机每接收一个数据后，都将返回一个应答信号（ACK），如此反复，直到主机需要停止读取数据时，发送一个非应答信号（NACK），将停止读取数据。

22.5.4.4 SMBus 特定功能

系统管理总线（SMBus）是一种两线接口，它基于 I2C 总线原理。

系统管理总线规范指的是三种类型的设备

从机：接收或相应命令的设备。

主机：下达命令，产生时钟与终止传输的设备。

HOST：一种特殊主机，向系统 CPU 提供接口，HOST 必须具有主机和从机双重功能，并且支持 SMBus HOST 通知协议，一个系统只有一个 HOST。

总线协议

对于任何给定的设备有 11 个可能的命令协议，一个设备可以使用任何或者全部的 11 个协议进行通信。

地址解析协议（ARP）

SMBus 从机地址冲突问题可通过为从设备标定一个新的独特的地址方式解决。为了分配地址，需要可区分每种设备的机制，每个设备都拥有一个唯一的设备标识符。这个 128 位的标识符是由软件实现的。

本设备支持地址解析协议（ARP）。将 I2C_CTRL1 寄存器中的 DEADDREN 位置 1,将启用 SMBus 设备的默认地址（0b1100001）。ARP 命令由用户软件实现。

ARP 支持的仲裁也在从机模式下完成。

命令接收与数据应答控制

SMBus 接收器对每个接收到的命令与数据都会回应 NACK。启动从机模式下的 ACK 控制，需将 I2C_CTRL1 寄存器的 SBCEN 位置 1，来启动从机字节控制模式。

HOST 通知协议

设置 I2C_CTRL1 寄存器的 HADDREN 位，使得本外设支持 HOST 通知协议，这种情况下，HOST 会应答 SMBus 主机地址（0b0001000）。

使用该协议，该设备作为主机，HOST 作为从机。

SMBus 报警

本外设可选 SMBus 提醒信号支持。一个仅作为从机的设备在想要发起通信时，可以通过 SMBALERT 引脚通知 HOST。HOST 会处理这个中断，并且随即通过提醒响应地址（0b0001100）来访问所有 SMBALERT 设备。只有 SMBALERT 引脚被拉低的设备会回应提醒响应地址。

SMBus 超时管理

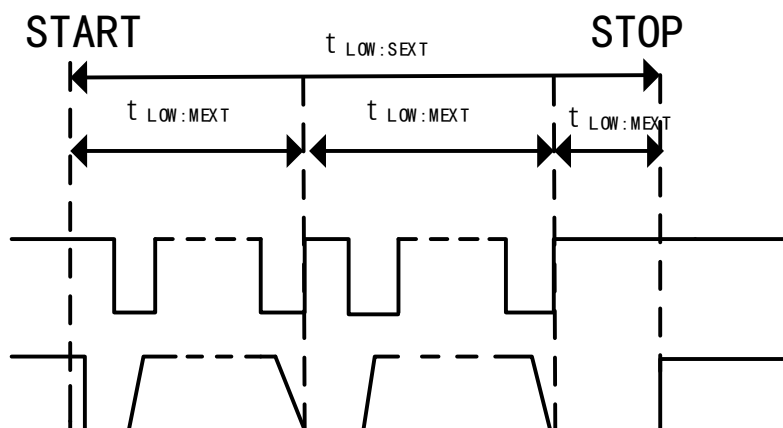
表格 76 SMBus 超时规范

符号	参数	范围		单位
		最小值	最大值	
t _{TIMEOUT}	检测时钟低超时	25	35	ms
t _{LOW:SEXT}	从机累积时钟低延长时间		25	ms
t _{LOW:MEXT}	主机累积时钟低延长时间		10	ms

t_{LOW:SEXT} 是一个从机设备从 START 到 STOP 给定的可延长时钟周期累计。当一个从机设备或主机设备对时钟进行占用导致总时钟低的时间大于 t_{LOW:SEXT}。因此这个参数测试条件是从机作为一个全速主机的唯一通信目标。

t_{LOW:MEXT} 是一个主机设备按照 START 到 ACK, ACK 到 ACK, ACK 到 STOP 的方式发送一个字节所允许的时钟周期累计，当另一个从机设备或主机占用时钟时也可能导致总时钟占用时间大于 t_{LOW:MEXT}。因此，这个参数测量条件是只有一个全速从机作为唯一通信目标。

图 93 $t_{LOW:SEXT}$ 与 $t_{LOW:MEXT}$ 时间



总线空闲检测

若主机检测到时钟信号为高的时间比 $t_{HIGH,MAX}$ 大，此时总线处于空闲状态。

22.5.4.5 停机模式下唤醒

当 I2C_CTRL1 寄存器的 WUPEN 位置 1 时，启动从停机模式唤醒功能。

只有选择 HSICLK 振荡器作为 I2C_CLK 的时钟源才能从停机模式下唤醒。

在停机模式下，HSICLK 处于关闭状态，当检测到起始位后，I2C 接口打开 HSICLK，此时 SCL 拉低直到 HSICLK 启动完成。当地址匹配时，在 MCU 唤醒期间 I2C 将持续拉低 SCL。只有在软件清除了 ADDRFLG 标志后，才会释放 SCL，传输进入正常状态。若地址不匹配，HSICLK 将关闭，MCU 保持停机模式。

注意：

- (1) 如果选择 I2C_CLK 为系统时钟或 WUPEN=0，HSICLK 振荡器收到起始位也不会被打开。
- (2) 只有 ADDR 中断可以唤醒 MCU，所以 I2C 做主机传输数据或从机被寻址到时，不要进入停机模式。
- (3) 数字滤波器和从停机模式唤醒功能不兼容。所以若 DNFCFG 位不等于 0，设置 WUPEN 位则没有作用。
- (4) 需要启用时钟延长功能。

22.5.4.6 错误标志位

I2C 通信存在以下几种导致通信失败的错误标志位。

总线错误标志位 (BERRFLG)

当在 9 倍 SCL 时钟脉冲信号以外检测到一个 START 或者 STOP 条件，会发生一个总线错误。当 SCL 为高的时候，SDA 出现上升沿或下降沿，则会检测为 START 或 STOP 信号。

只有在 I2C 通信传输数据情况下才会发生总线错误（作为主机发送数据或作为从

机已地址匹配后), 在从机模式地址匹配阶段不会发生这种错误。

当检测到总线错误时, I2C_STS 寄存器中的 BERRFLG 标志位将被硬件置 1, 若 I2C_CTRL1 寄存器中的 ERRIEN 位置 1 的话, 则产生错误中断。

仲裁丢失标志位 (ALFLG)

SDA 线上发送高电平时, 在 SCL 上升沿却从 SDA 采样得到低电平时, 就会检测为仲裁丢失错误。

- 主机模式下, 仲裁丢失在地址阶段、数据阶段及数据确认阶段检测。在这种情况下, SDA 和 SCL 线被释放, START 控制位硬件清除, 主机模式自动切换为从机模式。
- 从机模式下, 仲裁丢失在数据阶段和数据确认阶段进行检测, 在这种情况下, 传输被终止, SCL 和 SDA 线被释放。

当检测到仲裁丢失错误时, I2C_STS 寄存器中的 ALFLG 标志位将被硬件置 1, 若 I2C_CTRL1 寄存器中的 ERRIEN 位置 1 的话, 则产生错误中断。

过载/欠载错误标志位 (OVRURFLG)

当时钟延长被禁止时, 从机模式下有下列条件发生时, 检测到欠载或过载错误:

- 接收时, RXDATA 寄存器还未被读取, 但是新发送的字节已经接收到。
- 发送时, 当应该发送第一个数据字节但 STOPFLG=1。如果 TXBEFLG=0, 那么 I2C_TXDATA 寄存器的值发送出去, 如果不是 0, 那么发送 0xFF。
- 发送时, 在一个新的字节应该被写入到 I2C_TXDATA 寄存器, 但却没有写, 那就会将 0xFF 发送出去。

当检测到过载/欠载错误, I2C_STS 寄存器的 OVRURFLG 标志硬件置 1, 如果 I2C_CTRL1 寄存器 ERRIEN 位置 1, 则产生中断。

包错误检查错误标志位 (PECEFLG)

该错误条件仅针对 SMBus 功能部分。在收到 PEC 字节与 I2C_PEC 寄存器的内容不匹配时, 会检测到 PEC 错误, 错误的 PEC 接收后, 会自动返回一个 NACK。当检测到 PEC 错误, I2C_STS 寄存器的 PECEFLG 标志硬件置 1, 如果 I2C_CTRL1 寄存器 ERRIEN 位置 1, 则产生中断。

超时错误标志位 (TTEFLG)

该错误条件仅针对 SMBus 功能部分。

当以下条件发生时超时错误将会产生:

- (1) 检测 SMBus 超时
 - IDLECLKTO=0 并且 SCL 保持低的时间达到了 TIMEOUTA[11:0]位域定义的时间

- IDLECLKTO=1 并且 SDA 与 SCL 高电平时间超过了 TIMEOUTA[11:0] 位域定义的时间

(2) 检测 SMBus 空闲超时

- 主机时钟低延长累计时间达到了 TIMEOUTB[11:0]位域定义的时间 (t_{LOW:MEXT})
- 从机时钟低延长累计时间达到了 TIMEOUTB[11:0]位域定义的时间 (t_{LOW:SEXT})

当检测到 TIMEOUT 错误, I2C_STS 寄存器的 TTEFLG 标志硬件置 1, 如果 I2C_CTRL1 寄存器 ERRIEN 位置 1, 则产生中断。

22.5.4.7 DMA 请求

通过设置 I2C_CTRL1 寄存器 DMATXEN 位,可以启用 DMA 发送。数据被预先放到 DMA 外设所设定的 SRAM 区域发送到 I2C_TXDATA 寄存器 (无需考虑 TXINTFLG 位状态)。

只使用 DMA 传输字节:

- 主机模式下: 初始化, 从机地址, 方向, 字节数和起始位由软件设置 (当从机地址已被发送后, 无法使用 DMA 传输)。当所有数据都使用 DMA 传输时, 必须在 START 位置 1 前, 初始化 DMA。
- 从机模式下: DMA 必须在地址匹配事件之前初始化。

22.5.5 I2C 中断

表格 77 中断请求表

中断事件	事件标志位	事件标志位清除方法	中断使能控制位
接收的字符非空	RXBNEFLG	读 I2C_RXDATA 寄存器	RXIEN
发送中断状态	TXINTFLG	写 I2C_TXDATA 寄存器	TXIEN
停止信号检测标志	STOPFLG	写 STOPCLR=1	STOPIEN
传输完成重装载	TXCRFLG	写 I2C_CTRL2 且 NUMBYT[7:0]不等于 0	TXCIEN
传输完成	TXCFLG	写 START=1 或 STOP=1	
地址匹配	ADDRMFLG	写 ADDRMCCLR=1	SADDRMIEN
收到 NACK 标志位	NACKFLG	写 NACKCLR=1	NACKRXIEN
总线错误	BERRFLG	写 BERRCLR=1	ERRIEN
仲裁丢失	ALFLG	写 ALCLR=1	
过载/欠载错误	OVRURFLG	写 OVRURCLR=1	
PEC 错误	PECEFLG	写 PECECLR=1	
时钟超时	TTEFLG	写 TTECLR=1	
SMBus 提醒	SMBALTFLG	写 SMBALTCLR=1	

使能 I2C 中断, 需要:

- 在 NVIC 中配置与启动 I2C 通道
- 配置 I2C 中断使能位

22.6 寄存器地址映射

表格 78 I2C 寄存器地址映射

寄存器名	描述	偏移地址
I2C_CTRL1	控制寄存器 1	0x00
I2C_CTRL2	控制寄存器 2	0x04
I2C_ADDR1	主机地址寄存器 1	0x08
I2C_ADDR2	主机地址寄存器 2	0x0C
I2C_TIMING	时序寄存器	0x10
I2C_TIMEOUT	超时寄存器	0x14
I2C_STS	状态寄存器	0x18
I2C_INTFCLR	中断标志清除寄存器	0x1C
I2C_PEC	PEC 寄存器	0x20
I2C_RXDATA	接收数据寄存器	0x24
I2C_TXDATA	发送数据寄存器	0x28

22.7 寄存器功能描述

22.7.1 控制寄存器 1 (I2C_CTRL1)

偏移地址: 0x00

复位值: 0x0000 0000

位/域	名称	R/W	描述
0	I2CEN	R/W	使能 I2C (I2C Enable) 0: 禁止 1: 使能
1	TXIEN	R/W	使能发送中断 (Transmit Interrupt Enable) 0: 禁止 1: 使能
2	RXIEN	R/W	使能接收中断 (RX Interrupt Enable) 0: 禁止 1: 使能
3	SADDRMIEN	R/W	使能从机地址匹配中断 (Slave Address Match Interrupt Enable) 0: 禁止 1: 使能
4	NACKRXIEN	R/W	使能 NACK 接收中断 (NACK Received Interrupt Enable) 0: 禁止 1: 使能
5	STOPIEN	R/W	使能 STOP 检测中断 (STOP Detection Interrupt Enable) 0: 禁止 1: 使能

位/域	名称	R/W	描述
6	TXCIEN	R/W	使能发送完成中断 (Transmit Complete Interrupt Enable) 0: 禁止 1: 使能
7	ERRIEN	R/W	使能出错中断 (Error Interrupt Enable) 0: 禁止 1: 使能以下任何状态寄存器中的位置 1 时, 将产生该中断: SMBALFLG、TTEFLG、PECEFLG、OVRURFLG、ALFLG、STS1_BERRFLG
11:8	DNFCFG	R/W	配置数字噪音滤波器 (Digital Noise Filter Configure) SDA 和 SCL 的数字噪音滤波器通过这个位域来配置。数字滤波器的长度为 DNFCFG[3:0]*tI2C_CLK。 0000: 禁止 0001: 使能, 1 个 tI2C_CLK 1111: 使能, 15 个 tI2C_CLK 若同时使能模拟滤波器, 那么数字滤波器加在模拟滤波器上; 此位只能在未置位 I2CEN 时设置。
12	ANFD	R/W	禁止模拟噪音滤波器 (Analog Noise Filter Disable) 0: 使能 1: 禁止 此位只能在未置位 I2CEN 时设置。
13	SWRST	W	软件复位 (Software Reset) 当此位置 1 时, SCL 和 SDA 线被释放, 内部状态机、通信控制位和状态位回到其复位值, 除此以外的控制位被保留。 此位只能写 1, 写 0 无效。读此位将返回 0。
14	DMATXEN	R/W	使能 DMA 发送 (DMA Transmit Enable) 0: 禁止 1: 使能
15	DMARXEN	R/W	使能 DMA 接收 (DMA Receive Enable) 0: 禁止 1: 使能
16	SBCEN	R/W	使能从机字节控制 (Slave Byte Control Enable) 0: 禁止 1: 使能
17	CLKSTRECHD	R/W	禁止时钟延长时间 (Clock Stretching Time Disable) 0: 使能 1: 禁止 此位只能在未置位 I2CEN 时设置且只适用于从机模式。
18	WUPEN	R/W	使能从停机模式唤醒 (Wakeup From Stop Mode Enable) 0: 禁止 1: 使能
19	RBEN	R/W	使能响应广播 (Responds Broadcast Enable) 响应广播地址是 0x00。 0: 禁止 1: 使能

位/域	名称	R/W	描述
20	HADDREN	R/W	使能 SMBus HOST 地址 (SMBus Host Address Enable) HOST 地址是 0x10/0x11。 0: 禁止 1: 使能 若不支持 SMBus 模式, 保留此位并强制为 0。
21	DEADDREN	R/W	使能 SMBus 设备的默认地址 (SMBus Device Default Address Enable) 默认地址是 0xC2/0xC3。 0: 禁止 1: 使能 若不支持 SMBus 模式, 保留此位并强制为 0。
22	ALTEN	R/W	使能 SMBus 通知功能 (SMBus Alert Function Enable) 设备模式 (HADDREN=0): 0: 释放 SMBALERT 引脚, 禁止 NACK 之后的通知响应地址头。 1: 拉低 SMBALERT 引脚, 使能 ACK 之后的通知响应地址头。 HOST 模式 (HADDREN=1): 0: 不支持 1: 支持 若 ALTEN=0, SMBALERT 引脚可以当作一个 GPIO; 若不支持 SMBus 模式, 保留此位并强制为 0。
23	PECEN	R/W	使能 PEC (PEC Enable) 0: 禁止 1: 使能 若不支持 SMBus 模式, 保留此位并强制为 0。
31:24	保留		

22.7.2 控制寄存器 2 (I2C_CTRL2)

偏移地址: 0x04

复位值: 0x0000 0000

位/域	名称	R/W	描述
0	SADDR[0]	R/W	设置从机地址 (Slave Address Setup) 地址模式为 7 位时, 该位无效; 地址模式为 10 位时, 该位是地址的第 0 位。
7:1	SADDR[7:1]	R/W	设置从机地址 (Slave Address Setup) 从机地址的第 7:1 位。
9:8	SADDR[9:8]	R/W	设置从机地址 (Slave Address Setup) 地址模式为 7 位时, 该位无效; 地址模式为 10 位时, 该位是地址的第 9:8 位。
10	TXDIR	R/W	设置主机模式的传输方向 (Master Mode Transfer Direction Setup) 0: 写传输 1: 读传输
11	SADDRLEN	R/W	配置从机地址长度 (Slave Address Length Configure) 0: 7 位寻址模式 1: 10 位寻址模式

位/域	名称	R/W	描述
12	ADDR10	R/W	配置主机发送 10 位地址头 (Master Transmit 10-Bit Address Header Configure) 0: 发送 10 位从机地址读序列: 起始位+2 字节 10 位写方向地址+重新起始+10 位读方向地址中的前 7 位。 1: 发送 10 位从机地址读序列的前 7 位+读方向。
13	START	R/W	发送起始位 (Start Bit Transmit) 可软件置 1、清 0, 在发送起始位和地址序列后、发生仲裁丢失、超时错误或未设置 I2CEN 位时由硬件清 0, 也可以置位 I2C_INTFCLR 寄存器的 ADDRMCCLR 位清 0。 在主模式下: 0: 不发送 1: 重复发送 在从模式下: 0: 不发送 1: 总线空闲时发送 对此位写 0 无意义; 置位 RELOAD 位此位不起作用。
14	STOP	R/W	发送停止位 (Stop Bit Transmit) 可软件置 1、清 0; 当发送停止位时或未设置 I2CEN 位时由硬件清 0。 在主模式下: 0: 不发送 1: 发送 对此位写 0 无意义。
15	NACKEN	R/W	使能发送 NACK (Transmit NACK Enable) 可软件置 1、清 0; 当发送停止位、NACK、收到地址匹配事件后或未设置 I2CEN 位时由硬件清 0。 0: 发送 ACK 1: 发送 NACK 对此位写 0 无意义且只适用于从模式。 在主机接收模式中, 在发送最后一个字节后和发送停止位或 RESTART 位之间自动发送。 在从机接收模式中, 发生过载时, 会自动发送 NACK。这时 NACKEN 位不起作用; 使能硬件 PEC 检查后, PEC 的确认值也不取决于 NACK 位的值。
23:16	NUMBYT	R/W	设置字节数 (Number of Bytes Setup) 此位决定要传输的字节数。在从机模式且 SBCEN=0 时, 此位无意义。 此位只能在未设置 START 位时置位。
24	RELOADEN	R/W	使能 NUMBYT 重载模式 (NUMBYT Reload Mode Enable) 由软件置 1、清 0。 0: 传输完 NUMBYT 个字节后传输结束 1: 传输完 NUMBYT 个字节后重载 NUMBYT。当传输完 NUMBYT 个数据后, TXCFLG 标志位置位, SCL 被拉低。
25	ENDCFG	R/W	配置结束模式 (End Mode Configure) 由软件置 1、清 0。 0: 软件结束模式: 当传输完 NUMBYT 个数据后, TXCFLG 标志位置位, SCL 被拉低。

位/域	名称	R/W	描述
			1: 自动结束模式: 当传输完 NUMBYT 个数据后, 自动发送一个停止位。 此位在从机模式或 RELOADEN 位置位时无效。
26	PEC	R/W	使能传输包错误检查字节 (Transfer Packet Error Checking Byte Enable) 可软件置 1, 清 0; 在 PEC 传输完成、收到停止位、收到地址匹配事件后或未设置 I2CEN 位时由硬件清 0。 0: 禁止 1: 使能 对此位写 0 无意义。 置位 RELOADEN 位或在从机模式下清除 SBCEN 位, 此位不起作用; 若不支持 SMBus 模式, 保留此位并强制为 0。
31:27	保留		

22.7.3 主机地址寄存器 1 (I2C_ADDR1)

偏移地址: 0x08

复位值: 0x0000 0000

位/域	名称	R/W	描述
0	ADDR1[0]	R/W	设置主机地址 (Master Address Setup) 地址模式为 7 位时, 该位无效; 地址模式为 10 位时, 该位是地址的第 0 位。
7:1	ADDR1[7:1]	R/W	设置主机地址 (Master Address Setup) 主机地址的第 7:1 位
9:8	ADDR1[9:8]	R/W	设置主机地址 (Master Address Setup) 地址模式为 7 位时, 该位无效; 地址模式为 10 位时, 该位是地址的第 9:8 位。
10	ADDR1LEN	R/W	配置主机地址长度 (Master Address Length Configure) 0: 7 位寻址模式 1: 10 位寻址模式
14:11	保留		
15	ADDR1EN	R/W	使能主机地址 1 (Master Address 1 Enable) 0: 禁止。收到从机地址 ADDR 后发送 NACK 1: 使能。收到从机地址 ADDR 后发送 ACK
31:16	保留		

22.7.4 主机地址寄存器 2 (I2C_ADDR2)

偏移地址: 0x0C

复位值: 0x0000 0000

位/域	名称	R/W	描述
0	保留		
7:1	ADDR2[7:1]	R/W	设置主机地址 (Master Address Setup) 主机地址的第 7:1 位
10:8	ADDR2MSK	R/W	选择屏蔽主机地址 2 (Masks Master Address 2 Select)

位/域	名称	R/W	描述
			000: 无屏蔽 001: 屏蔽 ADDR2[1], ADDR2[7:2]进行比较。 010: 屏蔽 ADDR2[2:1], ADDR2[7:3]进行比较。 011: 屏蔽 ADDR2[3:1], ADDR2[7:4]进行比较。 100: 屏蔽 ADDR2[4:1], ADDR2[7:5]进行比较。 101: 屏蔽 ADDR2[5:1], ADDR2[7:6]进行比较。 110: 屏蔽 ADDR2[6:1], ADDR2[7]进行比较。 111: 屏蔽 ADDR2[7:1], 没有比较, 所有接收到的 7 位地址都会发送 ACK。 此位只能在未置位 ADDR2EN 位时设置; 若 ADDR2MSK 不等于 0, 保留的 I2C 地址也没有被响应, 匹配也无意义。
14:11	保留		
15	ADDR2EN	R/W	使能主机地址 2 (Master Address 2 Enable) 0: 禁止。收到从机地址 ADDR2 后发送 NACK。 1: 使能。收到从机地址 ADDR2 后发送 ACK。
31:16	保留		

22.7.5 时序寄存器 (I2C_TIMING)

偏移地址: 0x10

复位值: 0x0000 0000

位/域	名称	R/W	描述
7:0	SCLL	R/W	设置 SCL 低电平时间 (SCL Low Level Time Setup) $t_{SCLL} = (SCLL + 1) \times t_{TIMINGPSC}$ SCLL 决定 tBUF 和 tSU:STA 时序。
15:8	SCLH	R/W	设置 SCL 高电平时间 (SCL High Level Time Setup) $t_{SCLH} = (SCLH + 1) \times t_{TIMINGPSC}$ SCLH 决定 tSU:STO 和 tHD:STA 时序。
19:16	DATAHT	R/W	设置数据延长时间 (Data Hold Time Setup) 此位域决定发送模式中 SCL 的下降沿和 SDA 的沿之间的延迟 tDATAHT。 $t_{DATAHT} = DATAHT \times t_{TIMINGPSC}$ DATAHT 决定 tHD:DAT 时序。
23:20	DATAT	R/W	设置数据时间 (Data Time Setup) 此位域决定发送模式中 SDA 的沿和 SCL 上升沿之间的延迟 tDATAT。 $t_{DATAT} = (DATAT + 1) \times t_{TIMINGPSC}$ tDATAT 决定 tSU:DAT 时序。
27:24	保留		
31:28	TIMINGPSC	R/W	设置时序预分频器 (Timing Prescaler Setup) 此位域分频 I2C_CLK, 为数据的建立、保持时间计数器以及 SCL 高低电平计数器提供时钟周期 tTIMINGPSC。 $t_{TIMINGPSC} = (TIMINGPSC + 1) \times t_{I2C_CLK}$ 。

注: 此寄存器只能在未设置 I2CEN 位时置位。

22.7.6 超时寄存器 (I2C_TIMEOUT)

偏移地址: 0x14

复位值: 0x0000 0000

位/域	名称	R/W	描述
11:0	TIMEOUTA	R/W	设置总线超时 A (Bus Timeout A Setup) 当 IDLECLKTO=0 时, SCL 低超时: $t_{TIMEOUT}=(TIMEOUTA+1) \times 2048 \times t_{I2C_CLK}$ 当 IDLECLKTO=1 时, 总线空闲时: $t_{IDLE}=(TIMEOUTA+1) \times 4 \times t_{I2C_CLK}$ 此位只能在未设置 CLKTOEN 位时置位。
12	IDLECLKTO	R/W	配置空闲时钟超时检测 (Idle Clock Timeout Detection Configure) 0: 检测 SCL 低电平超时 1: 检测 SCL 和 SDA 高电平超时 (总线空闲) 此位只能在未设置 CLKTOEN 位时置位。
14:13	保留		
15	CLKTOEN	R/W	使能时钟超时检测 (Clock Timeout Enable) 0: 禁止 1: 使能。当 SCL 保持低的时间超过 $t_{TIMEOUT}$ 或保持高的时间超过 t_{IDLE} , 检测到超时错误。
27:16	TIMEOUTB	R/W	设置总线超时 B (Bus Timeout B Setup) 主机模式下, 要检测的累积主机时钟低延长时间($t_{LOW:MEXT}$)。 从机模式下, 要检测的累积从机时钟低延长时间($t_{LOW:SEXT}$)。 $t_{LOW:EXT}=(TIMEOUTB+1) \times 2048 \times t_{I2C_CLK}$ 此位域只能在未设置 EXCLKTOEN 位时设置。
30:28	保留		
31	EXCLKTOEN	R/W	使能延长时钟超时 (Extended Clock Timeout Enable) 0: 禁止 1: 使能。当 SCL 保持低的时间达到 $t_{LOW:EXT}$, 检测到超时错误。

22.7.7 状态寄存器 (I2C_STS)

偏移地址: 0x18

复位值: 0x0000 0001

位/域	名称	R/W	描述
0	TXBEFLG	R/S	发送数据缓冲器为空标志 (Transmit Data Buffer Empty Flag) 0: 发送缓冲器不为空 1: 发送缓冲器为空 I2C_TXDATA 寄存器的内容为空时由硬件置 1; 当把要发送的数据写到 I2C_TXDATA 寄存器时清除此位。 可软件置 1, 以清空 I2C_TXDATA 寄存器; I2CEN=0 时由硬件清 0。
1	TXINTFLG	R/S	发送中断状态标志 (Transmit Interrupt State Flag) 0: 未发送 1: 发送 I2C_TXDATA 寄存器为空的时候由硬件置 1, 此时把要发送的数据写到 I2C_TXDATA 寄存器。下一个要发送的数据写到 I2C_TXDATA 寄存器时清除此位。 可在 CLKSTRETCHD=1 时由软件置 1, 以产生 TXINTFLG 标志位; I2CEN=0 时由硬件清 0。
2	RXBNEFLG	R	接收数据缓冲器不为空标志 (Receive Data Buffer Not Empty Flag)

位/域	名称	R/W	描述
			0: 接收缓冲器为空 1: 接收缓冲器不为空 当 RXDATA 寄存器有数据时, 由硬件置 1; 读 I2C_RXDATA 可清除此位; I2CEN=0 时由硬件清 0。
3	ADDRMFLG	R	从机地址匹配标志 (Slave Address Match Flag) 0: 从机地址不匹配 1: 从机地址匹配 当接收到的从机地址与其中任意一个有效的从机地址匹配时, 此位由硬件置 1。将 ADDRMCCLR 位置 1, 由软件清零; I2CEN=0 时由硬件清 0。
4	NACKFLG	R	接收到 NACK 标志 (Receive Not Acknowledge Flag) 0: 未接收到 NACK 标志 1: 接收到 NACK 标志 当一个字节传输后收到 NACK 时由硬件置 1。将 NACKCLR 位置 1, 由软件清 0; I2CEN=0 时由硬件清 0。
5	STOPFLG	R	停止位检测标志 (Stop Bit Detection Flag) 0: 未检测到停止位 1: 检测到停止位 当外设参与传输时, 在总线上检测到停止位时由硬件置 1。 若在作为主机时由外设发送停止位, 或在作为从机时外设在本次传输前被正确寻址, STOPCLR=1 时, 由软件清 0; I2CEN=0 时由硬件清 0。
6	TXCFLG	R	发送数据完成标志 (Transmit Data Complete Flag) 0: 发送数据未完成 1: 发送数据完成 在 RELOADEN=0, ENDCFG=0 且发送 NUMBYT 个数据完毕后由硬件置 1; START=1 或 STOP=1 时清 0; I2CEN=0 时由硬件清 0。
7	TXCRFLG	R	传输完成重加载标志 (Transfer Complete Reload Flag) 0: 传输完成结束 1: 传输完成重加载 在 RELOADEN=1 且发送 NUMBYT 个数据完毕后由硬件置 1; 向 NUMBYT 写入是一个非零的值时由软件清 0; I2CEN=0 时, 由硬件清 0。此位只作用于主机模式, 或从机模式中 SBCEN=1 时。
8	BERRFLG	R	总线错误标志 (Bus Error Flag) 0: 未发生总线错误 1: 发生总线错误 在检测到错误的起始位或停止位时由硬件置 1; 设置 BERRCLR 位由软件清 0; I2CEN=0 时由硬件清 0。
9	ALFLG	R	仲裁丢失标志 (Arbitration Lost Flag) 0: 未发生仲裁丢失 1: 发生仲裁丢失 在总线仲裁丢失时由硬件置 1; 设置 ALCLR 位由软件清 0; I2CEN=0 时由硬件清 0。
10	OVRURFLG	R	发生过载/欠载标志 (Overrun/Underrun Flag) 0: 未发生过载/欠载 1: 发生过载/欠载

位/域	名称	R/W	描述
			在从机模式下 CLKSTRETCHD=1 时，发生过载/欠载错误时由硬件置 1；设置 OVRURCLR 位由软件清 0；I2CEN=0 时由硬件清 0。
11	PECEFLG	R	接收时出现 PEC 错误标志（PEC Error in Reception Flag） 0：未出现 PEC 错误 1：出现 PEC 错误 在收到的 PEC 值和 PEC 寄存器的值不匹配时由硬件置 1。收到错误的 PEC 时，自动发送一个 NACK。设置 PECECLR 位由软件清 0；I2CEN=0 时由硬件清 0。若不支持 SMBus 模式，保留此位并由硬件强制为 0。
12	TTEFLG	R	超时或 Tlow 错误标志（Timeout or Tlow Error Flag） 0：无超时错误 1：发生超时错误 发生超时或外部时钟超时时由硬件置 1；设置 TTECLR 位由软件清 0；I2CEN=0 时由硬件清 0。若不支持 SMBus 模式，保留此位并由硬件强制为 0。
13	SMBALTF LG	R	发生 SMBus 警报标志（SMBus Alert Occur Flag） 0：未发生 SMBus 警报 1：发生 SMBus 警报 若 HADDREN=1（SMBus HOST 配置）和 ALTEN=1，在 SMBALERT 脚上检测到 SMBALERT 下降沿时由硬件置 1；设置 SMBALTCLR 位由软件清 0；I2CEN=0 时由硬件清 0。若不支持 SMBus 模式，保留此位并由硬件强制为 0。
14	保留		
15	BUSBSYFLG	R	总线忙碌标志（Bus Busy Flag） 0：总线空闲（无通信） 1：总线忙（正在通信） 检测到一个起始位时由硬件置 1；检测到一个停止位时由硬件清 0；I2CEN=0 时清 0。
16	TXDIRFLG	R	传输方向标志（Transfer Direction Flag） 发生地址匹配事件时更新。 0：写传输，从机进入接收模式 1：读传输，从机进入发送模式
23:17	ADDRCMFLG	R	地址码匹配标志（Address Code Match Flag） 由地址匹配事件发生时所接收到的地址更新。 0：地址码不匹配 1：地址码匹配 在 10 位地址时，ADDRCMFLG 提供 10 位地址的头 2 位以后的地址。
31:24	保留		

22.7.8 中断标志清除寄存器（I2C_INTFCLR）

偏移地址：0x1C

复位值：0x0000 0000

位/域	名称	R/W	描述
2:0	保留		

位/域	名称	R/W	描述
3	ADDRMCLR	W	清除从机地址匹配标志（Slave Address Match Flag Clear） 设置此位会清除 I2C_STS 寄存器的 ADDRMLG 标志位和 I2C_CTRL2 寄存器的 START 位。
4	NACKCLR	W	清除接收到 NACK 标志（Receive Not Acknowledge Flag Clear） 设置此位会清除 I2C_STS 寄存器的 NACKFLG 标志位。
5	STOPCLR	W	清除停止位检测标志（Stop Bit Detection Flag Clear） 设置此位会清除 I2C_STS 寄存器的 STOPFLG 标志位。
7:6	保留		
8	BERRCLR	W	清除总线错误标志（Bus Error Flag Clear） 设置此位会清除 I2C_STS 寄存器的 BERRFLG 标志位。
9	ALCLR	W	清除仲裁丢失标志（Arbitration Lost Flag Clear） 设置此位会清除 I2C_STS 寄存器的 ALFLG 标志位。
10	OVRURCLR	W	清除过载/欠载标志（Overrun/Underrun Flag Clear） 设置此位会清除 I2C_STS 寄存器的 OVRURFLG 标志位。
11	PECECLR	W	清除 PEC 错误标志（PEC Error in Reception Flag Clear） 设置此位会清除 I2C_STS 寄存器的 PECEFLG 标志位。 若不支持 SMBus 模式，保留此位并由硬件强制为 0。
12	TTECLR	W	超时或 Tlow 错误标志（Timeout or Tlow Error Flag Clear） 设置此位会清除 I2C_STS 寄存器中的 TTEFLG 标志位。 若不支持 SMBus 模式，保留此位并由硬件强制为 0。
13	SMBALTCLR	W	清除发生 SMBus 警报标志（SMBus Alert Occur Flag Clear） 设置此位会清除 I2C_STS 寄存器中的 SMBALTFLG 标志位。 若不支持 SMBus 模式，保留此位并由硬件强制为 0。
31:14	保留		

22.7.9 PEC 寄存器（I2C_PEC）

偏移地址：0x20

复位值：0x0000 0000

位/域	名称	R/W	描述
7:0	PEC	R	设置 PEC 值（PEC Value Setup） 当 PECEN=1 时，此位域表示内部 PEC 值。 I2CEN=0 时由硬件清 0。
31:8	保留		

22.7.10 接收数据寄存器（I2C_RXDATA）

偏移地址：0x24

复位值：0x0000 0000

位/域	名称	R/W	描述
7:0	RXDATA	R	8 位接收数据字节（8-Bit Receive Data Byte） 从 I2C 总线接收的数据字节。
31:8	保留		

22.7.11 发送数据寄存器 (I2C_TXDATA)

偏移地址: 0x28

复位值: 0x0000 0000

位/域	名称	R/W	描述
7:0	TXDATA	R/W	8 位发送数据字节 (8-Bit Transmit Data Byte) 要发送到 I2C 总线上的数据字节。 此位域位只能在 TXBEFLG=1 时设置。
31:8	保留		

23 串行外设接口/片上音频接口（SPI/I2S）

23.1 术语全称、缩写描述

表格 79 SPI 术语全称、缩写描述

中文全称	英文全称	英文缩写
最高有效位	Most Significant Bit	MSB
最低有效位	Least Significant Bit	LSB
主机输出从机输入	Master Out Slave In	MOSI
主机输入从机输出	Master In Slave Out	MISO
串行时钟线	Serial Clock	SCK
串行数据	Serial Data	SD
主时钟	Master Clock	MCK
字选	Word Select	WS
脉冲编制调制	Pulse-code Modulation	PCM
片间传递音讯	Inter-IC Sound	I2S
发送	Transmit	TX
接收	Receive	RX
忙	Busy	BSY

23.2 简介

SPI 接口可以配置为支持 SPI 协议和 I2S 音频协议，默认工作在 SPI 模式，I2S 模式可以通过软件将功能切换。

串行外设接口（SPI）提供了基于 SPI 协议的数据发送和数据接收功能,允许芯片与外部设备以半双工、全双工、同行和串行方式通信，可以工作于主机或者从机模式。

片上音频接口（I2S）支持四种音频标准：飞利浦 I2S 标准、MSB 对齐标准、LSB 对齐标准以及 PCM 标准。在半双工通讯中，可以工作在主从模式下。

23.3 SPI 主要特征

- (1) 具有 3 线全双工同步传输接收的主从操作
- (2) 2 线可实现（第三根双向数据线可选带/不带）单工同步传输
- (3) 选择 4 位或者 16 位传输帧格式
- (4) 具有多主设备模式

- (5) 具有专用的发送和接收标志，可触发中断
- (6) 具有 SPI 总线忙状态标志
- (7) SPI 支持摩托罗拉方式
- (8) 主从模式的快速通信，最高可达 18MHz
- (9) 时钟的极性和相位可编程
- (10) 数据顺序可编程，选择 MSB 或者 LSB 在前
- (11) 主模式故障、过载以及 CRC 错误标志可触发中断
- (12) 具有 DMA 传输和接收缓冲器
- (13) 通过硬件 CRC 进行计算、发送和校验
- (14) CRC 错误标志
- (15) 两个 32 位嵌入式 RXFIFO 和 TXFIFO 带有 DMA 功能

23.4 I2S 主要特征

- (1) 具有单工通信（仅发送/接收）的主从模式
- (2) 四种音频标准
 - I2S 飞利浦标准
 - MSB 对齐标准
 - LSB 对齐标准
 - PCM 标准
- (3) 可选 16/24/32 位数据长度
- (4) 16 位或 32 位的通道长度
- (5) 时钟极性可编程
- (6) 16 位数据寄存器用于发送和接收
- (7) 数据方向一直都是 MSB 在先
- (8) 发送和接收支持 DMA 功能
- (9) 主时钟可以向一个外部音频组件输出

23.5 SPI 功能描述

23.5.1 SPI 信号线描述

表格 80 SPI 信号线描述

引脚名称	描述
SCK	主设备: SPI 时钟输出 从设备: SPI 时钟输入
MISO	主设备: 输入引脚, 接收数据 从设备: 输出引脚, 发送数据 数据方向: 从设备到主设备
MOSI	主设备: 输出引脚, 发送数据 从设备: 输入引脚, 接收数据 数据方向: 主设备到从设备
NSS	软件 NSS 模式: NSS 引脚可以它用。 主设备硬件 NSS 模式: 为 NSS 输出, 单主模式, NSS 关闭输出: 允许操作多主环境, 从设备硬件 NSS 模式: NSS 信号设置为低电平, 作为从机的片选信号

23.5.2 通信格式

SPI 通信中, 接收数据和发送数据可以同时进行, SCK 同步发送并且对数据线上的数据进行采样。通信格式取决于时钟相位、时钟极性和数据帧格式, 如果要正常通信, 主设备和从设备必须是相同的通信格式。

23.5.2.1 时钟信号的相位和极性

时钟极性和时钟相位是 SPI_CTRL1 寄存器的 CPOL 和 CPHA 位。

时钟极性 CPOL 是指 SPI 处于空闲状态时, SCK 信号线的电平信号。

- CPOL=0 时, SCK 信号线在空闲状态为低电平
- CPOL=1 时, SCK 信号线在空闲状态为高电平

时钟相位 CPHA 是指数据的采样时刻

- CPHA=0 时, MOSI 或 MISO 数据线上的信号将会在 SCK 时钟线的“奇数边沿”被采样。
- CPHA=1 时, MOSI 或 MISO 数据线上的信号将会在 SCK 时钟线的“偶数边沿”被采样。

根据时钟相位 CPHA 和时钟极性 CPOL 的不同状态, 可以将 SPI 分成四种模式。

表格 81 SPI 的四种模式

SPI 模式	CPHA	CPOL	采样时刻	空闲时 SCK 时钟
0	0	0	奇数边沿	低电平
1	0	1	奇数边沿	高电平

SPI 模式	CPHA	CPOL	采样时刻	空闲时 SCK 时钟
2	1	0	偶数边沿	低电平
3	1	1	偶数边沿	高电平

注意:

- (1) 如果要改变 CPOL 和 CPHA 位, SPI 必须通过 SPIEN 位清零关闭。
- (2) SCK 在空闲状态下, 如果 CPOL=1 上拉 SCK, 如果 CPOL=0 下拉 SCK。

23.5.2.2 数据帧格式

可以通过配置 SPI_CTRL1 寄存器的 LSBSEL 位选择是 LSB 还是 MSB 先行。配置 SPI_CTRL2 寄存器的 DSCFG 位选择数据字长, 无论是选择了何种, 当对 FIFO 进行读访问时必须与 FRTCFCFG 对齐。当访问 SPI_DATA 寄存器时, 数据帧总是右对齐。在通信过程中, 只有数据字长范围内的位会随着时钟输出。

23.5.3 NSS 模式

软件 NSS 模式: 通过配置 SPI_CTRL1 寄存器的 SSEN 位来选择启动或者禁止此模式, 内部 NSS 信号电平由 SPI_CTRL1 寄存器的 ISSEL 位驱动。

硬件 NSS 模式:

- 启动 NSS 输出: SPI 处于主模式时, 使能 SSOEN 位, NSS 引脚被拉至低电平, SPI 将自动进入从机模式。
- 关闭 NSS 输出: 允许操作在多主环境。

23.5.4 SPI 模式

23.5.4.1 SPI 主模式初始化

主模式中, 在 SCK 引脚产生串行时钟。

配置主模式

- 配置 SPI_CTRL1 寄存器中的 MSMCFG=1, 设置为主模式
- 通过配置 SPI_CTRL1 寄存器中的 BRSEL 位选择串行时钟波特率
- 通过配置 SPI_CTRL1 寄存器中的 CPOL 和 CPHA 位, 选择极性和相位
- 通过配置 SPI_CTRL1 寄存器中的 RXOMEN、BMOEN 和 BMEN 位选择传输模式
- 通过配置 SPI_CTRL2 寄存器中的 DSCFG 位选择数据位宽
- 通过配置 SPI_CTRL2 寄存器中的 NSSPEN 位打开 NSS 脉冲模式 (配置该位时, CPHA 位必须置 1)
- 通过配置 SPI_CTRL2 寄存器中的 FRTCFCFG 位, 设置触发 RXBNEFLG 事件的 RXFIFO 的阈值
- 如果使用 DMA 功能则需要配置 SPI_CTRL2 寄存器中的 LDTX 和 LDRX 位
- 如果使用 CRC, 需要将 CRC 多项式设置为输入同时设置 CRCEN 位

- 通过配置 SPI_CTRL1 寄存器中 LSBSEL 选择是 LSB 先行还是 MSB 先行
- NSS 配置：
 - NSS 引脚工作在输入模式：硬件模式下，需要在整个数据帧传输期间把 NSS 引脚连接在高电平；软件模式下，需要设置 SPI_CTRL1 寄存器中的 SSEN 位和 ISSEL 位
 - NSS 工作在输出模式中，需要配置 SPI_CTRL2 寄存器的 SSOEN 位
- 配置 SPI_CTRL1 寄存器中的 SPIEN 位，使能 SPI

在主模式中：MOSI 引脚是数据输出，MISO 是数据输入。

23.5.4.2 SPI 从模式初始化

在从模式中，SCK 引脚接收主设备传来的的串行时钟。

从模式配置

- 配置 SPI_CTRL1 寄存器中的 MSMCFG=0 设置为从模式
- 通过配置 SPI_CTRL1 寄存器中的 CPOL 和 CPHA 位，选择极性和相位
- 通过配置 SPI_CTRL1 寄存器中的 RXOMEN、BMOEN 和 BMEN 位选择传输模式
- 通过配置 SPI_CTRL2 寄存器中的 DSCFG 位选择数据位宽
- 通过配置 SPI_CTRL2 寄存器中的 NSSPEN 位打开 NSS 脉冲模式（配置该位时，CPHA 位必须置 1）
- 通过配置 SPI_CTRL2 寄存器中的 FRTCFCFG 位，设置触发 RXBNEFLG 事件的 RXFIFO 的阈值
- 如果使用 DMA 功能则需要配置 SPI_CTRL2 寄存器中的 LDTX 和 LDRX 位
- 如果使用 CRC，需要将 CRC 多项式设置为输入同时设置 CRCEN 位
- 通过配置 SPI_CTRL1 寄存器中 LSBSEL 选择是 LSB 先行还是 MSB 先行
- NSS 配置：
 - 硬件模式下：在完整的数据帧传输过程中 NSS 引脚必须为低电平
 - 软件模式下：设置 SPI_CTRL1 寄存器中的 SSEN 位并清除 ISSEL 位
- 配置 SPI_CTRL1 寄存器中的 SPIEN 位，使能 SPI

在从模式中：MOSI 引脚是数据输入，MISO 引脚是数据输出。

23.5.4.3 SPI 的全双工通信

通常情况下，SPI 被配置为全双工通信，其中，主机和从机的移位寄存器通过两个单向线 MOSI 和 MISO 进行连接。在 SPI 通信时，根据 SCK 时钟沿进行同步数据传输。主机的数据通过 MOSI 引脚发送到从机，从机的数据通过 MISO 引脚发送到主机，数据传输完成时，说明信息交换成功。

23.5.4.4 SPI 的半双工通信

一条时钟线加一条双向数据线

- 通过设置 SPI_CTRL1 寄存器的 BMEN 位启动此模式
- 通过设置 SPI_CTRL1 寄存器的 BMOEN 位控制数据线是输入还是输出
- SCK 引脚作为时钟，主设备中使用 MOSI 引脚传输数据，从设备中使用 MISO 引脚传输数据

23.5.4.5 SPI 的单工通信

一条时钟线和一条单向数据线（只接受或只发送）

此模式中，SPI 模块作为只接收或者只发送。

只发送模式：

- 数据在发送引脚上传输（主模式下是 MOSI、从模式下是 MISO）
- 此时接收引脚可以作为通用 I/O 使用（主模式下是 MISO，从模式下是 MOSI）

只接收模式：

- 通过设置 SPI_CTRL1 寄存器中 RXOMEN 位关闭 SPI 输出功能
- 释放发送引脚（主模式下是 MOSI，从模式下是 MISO）
- 主模式下，使能 SPI 启动通信，清除 SPI_CTRL1 寄存器的 SPIEN 位可以立即停止接收数据，不需要读取 BSYFLG 标志（一直为 1）
- 从模式下：NSS 拉至低电平，只要 SCK 由时钟脉冲，SPI 会一直接收

23.5.4.6 SPI 的多从设备通信

SPI 可以由多个从设备工作，主设备使用 GPIO 引脚管理从设备的片选线，可以控制两个以及两个以上的独立从设备。

主设备通过拉低从设备的 NSS 引脚，来决定与哪个从设备进行数据传输。

23.5.5 SPI 不同模式下数据发送和接收过程

表格 82 SPI 的运行模式

模式	配置	数据引脚
主设备的全双工模式	BMEN=0, RXOMEN=0	MOSI 发送; MISO 接收
主设备的单向接收模式	BMEN=0, RXOMEN=1	MOSI 不使用; MISO 接收
主设备的双向发送模式	BMEN=1, BMOEN=1	MOSI 发送; MISO 不使用
主设备的双向接收模式	BMEN=1, BMOEN=0	MOSI 不使用; MISO 接收
从设备的全双工模式	BMEN=0, RXOMEN=0	MOSI 接收; MISO 发送
从设备的单向接收模式	BMEN=0, RXOMEN=1	MOSI 接收; MISO 不使用
从设备的双向发送模式	BMEN=1, BMOEN=1	MOSI 不使用; MISO 发送
从设备的双向接收模式	BMEN=1, BMOEN=0	MOSI 接收; MISO 不使用

图 94 全双工模式连接

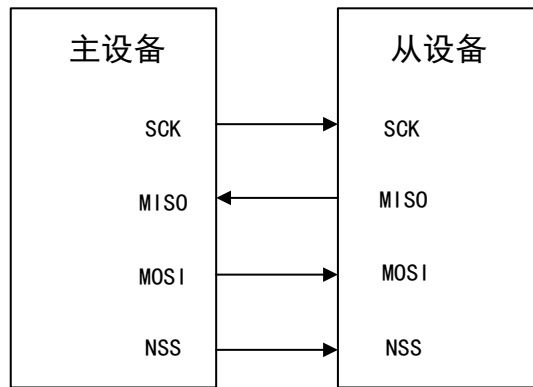


图 95 半双工模式连接

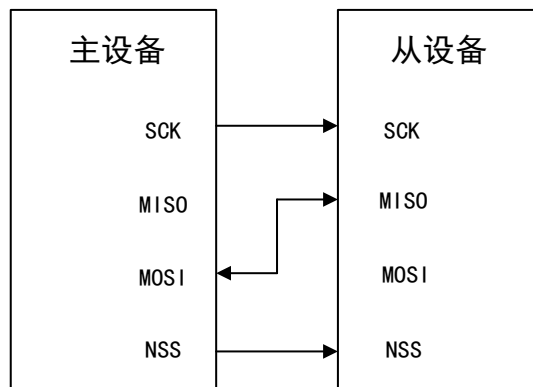


图 96 单工模式连接（主机用来接收，从机用来发送）

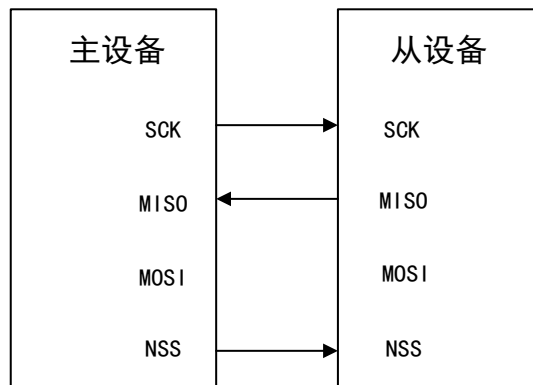
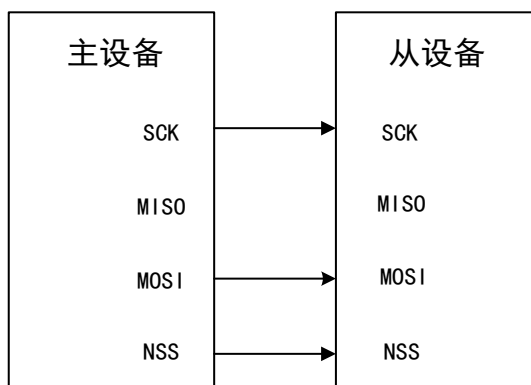


图 97 单工模式连接（主机只发送，从机接收）



23.5.5.1 数据的发送与接收

为了防止数据帧较短时溢出，保证 SPI 可以连续工作，所有的 SPI 数据都需要通过 32 位的嵌入式 FIFO。每个方向都会有各自的 FIFO，为 TXFIFO 和 RXFIFO。

根据 SPI 单工、双工模式，数据帧格式、FIFO 数据寄存器上执行的访问大小以及访问 FIFO 时是否使用数据打包对 FIFO 进行处理。

对 SPI_DATA 寄存器的读访问将返回存储在 RXFIFO 中尚未读取的最早的值。对 SPI_DATA 的写访问将写入数据存储在发送队列末尾的 TXFIFO 中。读访问必须始终与 SPI_CTRL2 寄存器中 FRTCFG 位配置的 RXFIFO 阈值对齐。FTLSEL 和 FRLSEL 位表示两个 FIFO 的当前占用水平。

对 SPI_DATA 寄存器的读访问必须由 RXBNEFLG 事件管理，当数据存储在 RXFIFO 中并达到阈值(由前位定义)时触发此事件；当 RXBNEFLG 被清除时，RXFIFO 被认为是空的，以相似的方式，对要传输的数据帧的写访问由 TXBEFLG 事件管理。当 TXFIFO 小于或等于其容量的一半时触发 RXBNEFLG 事件，否则 TXBEFLG 被清除，同时认为 TXFIFO 存有数据，因此，当数据帧格式小于或者等于一个字节时，RXFIFO 最多可以存储 4 个数据帧，TXFIFO 可以存储 3 个数据帧。当软件试图以 16 位模式向 TXFIFO 写入更多数据时，这一差异可以防止已经存储在 TXFIFO 中的 3 个 8 位数据帧损坏。TXBEFLG 和 RXBNEFLG 事件都可以轮询或由中断处理。

23.5.5.2 序列处理

在发送数据中，可以将多个数据按照顺序组成一个序列，当开始发送时，TXFIFO 会按照顺序连续发送完成。

在单接收模式中，半双工或者单工模式下，使能 SPI，主设备会马上开始接收序列，直到 SPI 关闭或者单接收模式关闭。当数据帧开始传输时，从设备无法控制数据序列，因此从机必须在传输开始前准备数据，即保证 TXFIFO 中有待发送的数据。

在有多个从设备时，每个序列都需要对应到不同的从设备中，所以需要 NSS 脉冲分隔序列保证正确。

注意:

- (1) 结合 FTLSEL 位和 BSYFLG 位检查数据是否传输完成，当传输完成时钟输出停止。
- (1) 在数据包模式中，数据传输是奇数时要特别注意空字节。
- (2) 主设备在单接收模式中，需要禁用 SPI 或者单接收模式，停止时钟输出。
- (3) 掌握正确接收的时间，保证数据传输正确
- (4) 关闭的动作要在首位采样时间和下一个字节的首位之间。

23.5.5.3 数据打包

当数据帧的大小小于或者等于一个字节时，对 SPI_DATA 寄存器执行 16 位的读写访问时，数据都会自动打包在一起，此时可以并行处理双数据。当对 SPI_DATA 写访问后，会发送 2 个字节的数据；如果 RXFIFO 的阈值设置为 16 位，会生成一个接受 RXBNEFLG 事件。

对单个 RXBNEFLG 事件，数据接收器要对 SPI_DATA 进行一次读操作，才能将数据全部取到。

注：RXFIFO 的阈值要和跟进的数据访问的位宽保持一致。

23.5.6 NSS 脉冲模式

通过配置 SPI_CTRL1 寄存器的 NSSPEN 位，设置 NSS 脉冲模式，此模式只有在 SPI 配置为摩托罗拉主模式且捕获第一个边沿时才有效。此模式的发送中，NSS 脉冲在两个连续的数据帧之间产生，NSS 至少会在高电平保持一个周期。NSS 脉冲模式允许从机锁存数据。

23.5.7 TI 模式

TI 协议主模式

通过配置 SPI_CTRL2 寄存器的 FRFCFG 位令 SPI 接口兼容 TI 协议主模式。

在 TI 协议主模式中，不受 SPI_CTRL1 寄存器设置的影响，时钟极性、相位和 NSS 管理都会保持在符合 TI 协议的要求。在从模式中，SPI 波特率分频器用来控制 MISO 引脚，使 MISO 引脚为高阻态，任何波特率都可以使用，保证了最佳灵活性。

一般情况下会设置波特率为外部主时钟的波特率，MISO 信号成为高阻态的延迟取决于内部在同步和通过 SPI_CTRL1 寄存器中的 BRSEL 位设置波特率。它的公式为：

$$T_{\text{baud_rate}/2+4} \times t_{\text{pclk}} < t_{\text{release}} < t_{\text{baud_rate}+6} \times t_{\text{pclk}}$$

注：此功能不适用于摩托罗拉 SPI 通信模式（FRFCFG 位设置为 0）

23.5.8 CRC 功能

SPI 模块含有两个 CRC 计算单元分别用来数据接收和数据发送。

CRC 计算单元是在 SPI_CRCPOLY 寄存器来定义多项式（应该是奇数，不支持偶数）。

通过配置 SPI_CTRL1 寄存器中的 CRCEN 位使能 CRC 计算；同时复位 CRC 寄存器（SPI_RXCRC 和 SPI_TXCRC）。

由 CPU 管理传输时的 CRC

为了得到传输计算的 CRC 值，当最后一个数据写入发送缓冲器后，需要设置 SPI_CTRL1 的 CRCNXT 位；指示硬件在发送最后一个数据之后，发送 CRC 数值，CRCNXT 位被清除；在 CRC 数据传输期间，CRC 计算会被冻结。

接收到的 CRC 数据会被存储在 RXFIFO 中，一个 CRC 格式的事务通常在数据序列的末尾需要多一个数据帧来通信，然而当设置有个由 16 位 CRC 校检的 8 位数据帧时，需要两个数据帧来发送完整的 CRC。当接收到最后一个 CRC 数据时，对接收的值和 SPI_RXCRC 寄存器的值进行比较。通过检查 SPI_STS 寄存器中的 CRCEFLG 标志位来判断在传输过程中数据是否被破坏，可以通过写 0 来清除 CRCEFLG 位。通过读取 SPI_DATA 寄存器清除 RXBNEFLG 位。

清除 CRC 数值顺序

- (1) 关闭 SPI (SPIEN=0)
- (2) 将 CRCEN 位清零
- (3) 将 CRCEN 位置 1
- (4) 使能 SPI (SPIEN=1)

注意：当 SPI 工作在从模式中时，软件必须在时钟稳定的情况下启用 CRC 运算。并且在数据阶段和 CRC 阶段时间，NSS 信号需要拉低并一直保持。

23.5.9 DMA 功能

SPI 中的请求/应答 DMA 机制，方便了高速率的数据传输，提高了系统效率，能够及时向 SPI 发送缓冲器传送数据，同时接收缓冲器也能及时读取数据防止溢出。

SPI 只发送数据时，只需要使能 DMA 的发送通道。

SPI 只接收数据时，只需要使能 DMA 的接收通道。

通过配置 SPI_CTRL2 寄存器的 TXDEN 和 RXDEN 位，使能 SPI 模式的 DMA 功能。

- 发送时：TXBEFLG 标志位置 1 时发出 DMA 请求，DMA 控制器写数据到 SPI_DATA 中，此时 TXBEFLG 标志位被清除。
- 接收时：RXBNEFLG 标志位置 1 时发出 DMA 请求，DMA 控制器从 SPI_DATA 寄存器中读取数据，此时 RXBNEFLG 标志位被清除。

通过监视 BSYFLG 标志位确认在发送模式下 DMA 传送所有需要发送的数据后 SPI 通信是否结束，可以避免破坏最后一个数据的传输。

带 CRC 的 DMA 功能

结束通信时，如果 SPI 同时使能 CRC 运算和 DMA 功能，CRC 字节的发送和接收会自动完成。CRCNXT 位不收软件控制，SPI 发送 DMA 通道计数器必须设置为不包含 CRC 数据的数量，但是接收时 DMA 通道计数器必须要多包含一个 CRC 数据的长度。

CRC 校验环节读取 CRC 数据后，SPI_TXCRC 和 SPI_RXCRC 的值会自动清零。此时，可以使用 DMA 循环模式（除了在单接受模式下）实现连续传输。

在数据和 CRC 传输结束时，如果 SPI_STS 寄存器中的 CRCEFLG 标志位置 1，说明传输期间发生了错误。

23.5.10 关闭 SPI

数据传输结束后，通过关闭 SPI 模块结束通信。

当数据还在传输时，或者 TXFIFO 中存在数据时，不可以操作 SPI_CTRL1 寄存器中的 SPIEN 位来关闭 SPI。如果设置了 SPIEN=0，时钟信号会持续发送，一直到重新启用外设。为了预防以上情况的发生需要一定的步骤来关闭 SPI。

关闭 SPI 步骤

- (1) 等待 FTLSEL 清零
- (2) 等待 BSYFLG 标志位清零
- (3) 等待 FRLSEL 清零
- (4) 关闭 SPI (SPIEN=0)

在一些单接收模式中关闭 SPI 步骤

- (1) 等待 RXOMEN 清零或者 BMOEN 置 1
- (2) 等待 BSYFLG 标志位清零
- (3) 等待 FRLSEL 清零
- (4) 关闭 SPI (SPIEN=0)

23.5.11 SPI 中断

在 SPI 运行期间，以下事件的发生可以出发中断：

- TXFIFO 准备装载
- RXFIFO 接收到数据
- 主模式错误
- CRC 错误
- TI 帧格式错误

23.5.11.1 状态标志位

完全监控 SPI 总线的状态有三个标志位

发送缓冲器空闲标志 TXBEFLG

TXBEFLG=1 说明 TXFIFO 有空间存储发送数据，TXBEFLG 标志位是连接到 TXFIFO 位上的，在存入数据过程时，如果 TXFIFO 的存储内容小于或者等于 FIFO/2 时，TXBEFLG 标志位保持为高。当 TXFIFO 的存储内容大于 FIFO/2，TXBEFLG 标志位会清零。其中如果设置了 SPI_CTRL2 寄存器中的 TXBEIEN 位，会产生中断。

接收缓冲器非空闲标志 RXBNEFLG

RXBNEFLG 标志位取决于 SPI_CTRL2 寄存器中的 FRTCFCG 位的值：

- 如果 FRTCFCG=1，当 RXFIFO 的存储内容大于或者等于 8 位时，RXBNEFLG=1
- 如果 FRTCFCG=1，当 RXFIFO 的存储内容大于或者等于 16 位时，RXBNEFLG=1

非上述情况，RXBNEFLG 标志位会自动清零。

当 SPI_CTRL2 寄存器中的 RXBNEIEN=1 时，会产生一个中断。

忙标志 BSYFLG

BSYFLG 标志由硬件设置和清除，可以表明 SPI 通信层的状态，BSYFLG=1 时，表明 SPI 正在通信。使用 BSYFLG 标志可以检测传输是否结束，避免破坏最后一次传输数据。

在以下情况中 BSYFLG 标志将会清零

- 主模式下，结束传输
- 主模式故障
- 从模式下，两次数据传输之间间隔至少一个 SPI 周期
- 关闭 SPI

通信连续时：

- 主模式中：传输的整个过程，BSYFLG=1
- 从模式中：每个数据的传输之间，BSYFLG 在一个 SCK 时钟周期保持为低

注：最好使用 TXBEFLG 和 RXBNEFLG 标志来处理每一个数据项的发送和接收

23.5.11.2 错误标志位

主模式错误 MEFLG

MEFLG 是一个错误标志位。主模式错误发生在：在硬件 NSS 模式中，主设备的 NSS 引脚被拉低；在软件 NSS 模式中，ISSEL 位被清零时；MEFLG 位被自动置位。

主模式失效的影响：MEFLG 置 1，在设置了 ERRIEN 的情况下，产生 SPI 中断；SPIEN 被清零，（输出停止，关闭 SPI 接口）；MSMCFG 被清零，设备强制

进入从模式。

清除 MEFLG 标志位的操作：MEFLG 标志位置 1 时，需要对 SPI_STS 寄存器进行读或者写操作，然后写入 SPI_CTRL1 寄存器。

MEFLG 标志位为 1 时，不允许设置 SPIEN 和 MSMCFG 位。

溢出错误 OVRFLG

发生以下事件会产生溢出错误

- 当主设备发送了数据之后，RXBNEFLG 标志位还为 1 时
- 当接收数据时 RXFIFO 中的空间不能存储要接收的数据时
- 当软件或者 DMA 没有能够及时的读取 RXFIFO 中的数据时
- 当 CRC 只在接收模式下启用，RXFIFO 不可用，接收缓冲区被限制在单个数据帧缓冲区中

发生溢出错误时：OVRFLG 位被置 1，如果还设置了 ERRIEN 位，会产生中断。

发生溢出错误后，接受缓冲器的数据不是主设备发送的数据，通过读取 SPI_DATA 值是之前没有读取的数据，之后的数据会被丢弃。

根据顺序读出 SPI_DATA 寄存器和 SPI_STS 寄存器可以清除 OVRFLG 标志。

CRC 错误标志位 CRCEFLG

通过设置 SPI_CTRL1 寄存器的 CRCEN 位，使能 CRC 运算，CRC 错误标志可以核对接收数据的是否有效。

当 SPI_TXCRC 寄存器发送的值与 SPI_RXCRC 寄存器中的数值不匹配时，产生 CRC 错误，此时 SPI_STS 寄存器中的 CRCEFLG 标志位置 1。

对 SPI_STS 寄存器的 CRCEFLG 位写 0，清除 CRCEFLG。

TI 模式帧格式错误 (FREFLG)

从设备下并且符合 TI 模式协议时，在数据通信期间，NSS 出现一个脉冲时，会导致 TI 模式帧格式错误。当出现 TI 模式帧格式错误时，SPI_STS 寄存器 FREFLG 标志位会置 1，不会禁用 SPI，忽略 NSS 脉冲，SPI 会在重新传输前等待下一个 NSS 脉冲。由于错误检测有可能导致两个数据字节的丢失，数据可能已经损坏。

可以通过读取 SPI_STS 寄存器，清除 FREFLG 标志，如果设置了 ERRIEN 位，就在 NSS 错误时产生一个中断，此时，禁用 SPI，因为不能保证数据的一致性，当 SPI 再次启用时，需要重新初始化主服务器。

表格 83 SPI 中断请求

中断标志	中断事件	使能控制位	清除方式
TXBEFLG	发送缓冲器空标志	TXBEIEN	写 SPI_DATA 寄存器
RXBNEFLG	接收缓冲器非空标志	RXBNEIEN	读 SPI_DATA 寄存器

中断标志	中断事件	使能控制位	清除方式
MEFLG	主模式失效事件	ERRIEN	读/写 SPI_STS 寄存器，然后写 SPI_CTRL1 寄存器
OVRFLG	溢出错误		读 SPI_DATA 寄存器，然后读 SPI_STS 寄存器
CRCEFLG	CRC 错误标志		写 0 到 CRCEFLG 位
FREFLG	TI 模式帧格式错误		读取 SPI_STS 寄存器

23.6 I2S 功能描述

通过设置 SPI_I2SCFG 的 I2SMOD 位，使能 I2S 功能。

I2S 与 SPI 共用四个引脚：

- SD：串行数据，发送和接收 2 路时分复用通道的数据
- WS：片选，切换左右声道的数据
- CK：串行时钟，主模式下时钟信号输出，从模式下时钟信号输入
- MCK：主时钟，在主模式下时并且 SPI_I2SPSC 寄存器的 MCOEN 位置 1 时，可以作为输出额外的时钟信号引脚使用。

23.6.1 I2S 音频标准

I2S 的音频标准是通过设置 SPI_I2SCFG 寄存器的 I2SSSEL 位和 PFSSEL 位来选择，一共可以选择四种音频标准：I2S 飞利浦标准，MSB 对齐标准，LSB 对齐标准和 PCM 标准。其中除了 PCM 标准其他的音频标准都是两个通道：左声道和右声道。

数据长度和通道长度可以通过 SPI_I2SCFG 寄存器中 DATALEN 和 CHLEN 位来配置。其中通道长度必须大于或者等于数据长度，有四种数据格式发送数据：16 位数据打包进 16 位帧，16 位数据打包进 32 位帧，24 位数据打包进 32 位帧，32 位数据打包进 32 位帧。

当 16 位的数据扩展到 32 位时，前 16 位的数据是有效数据，后 16 位强制为 0，此过程中不需要任何外部干预。

因为用于发送和接收的数据缓冲器都是 16 位，因此当 24 位和 32 位数据传输时，SPI_DATA 需要进行两次读/写操作，如果使用了 DMA 则需要两次 DMA 传输。

对于所有的通讯标准和数据格式，数据的最高位总是最先发送。

对于时分复用，总是先发送左通道，然后发送右通道。

23.6.1.1 I2S 飞利浦标准

在 I2S 飞利浦标准下，引脚 WS 可以说明正在发送的数据来自左通道还是右通道。

在 I2S 飞利浦标准中，WS 和 SD 都是在 CK 时钟信号的下降沿变化。

发送方会在时钟信号 CK 的下降沿改变数据，接收方会在时钟信号 CK 的上升沿改变数据。

图 98 I2S 飞利浦协议波形（16/32 位）

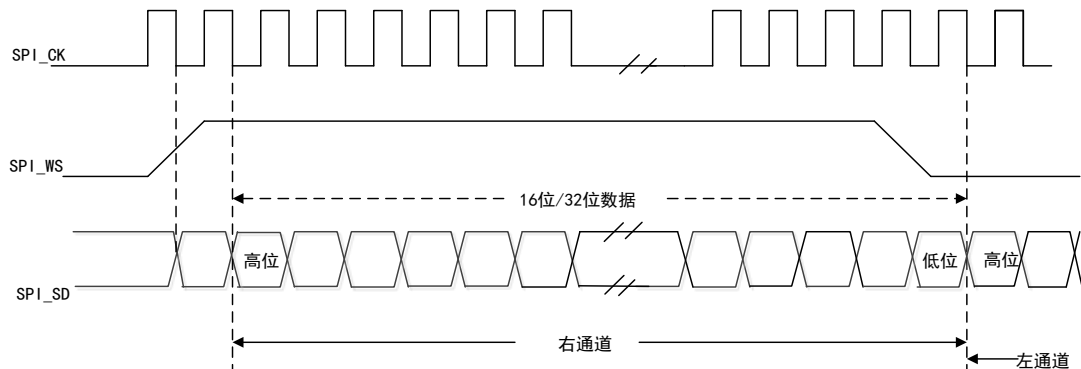
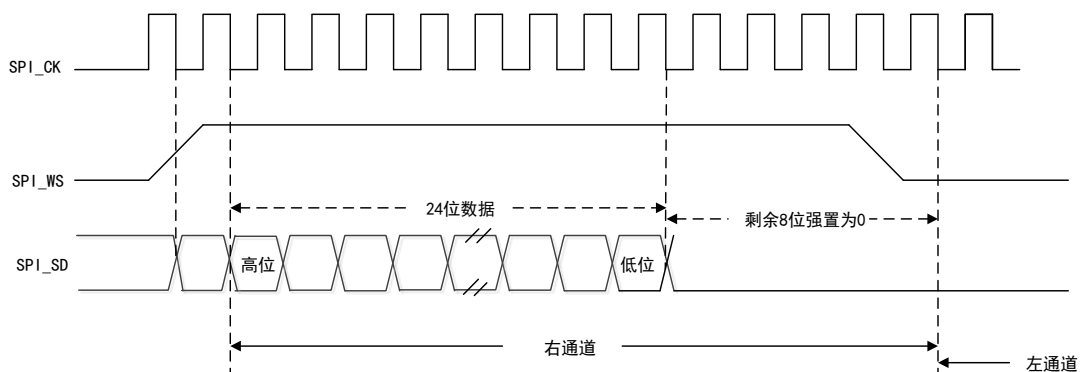


图 99 I2S 飞利浦协议波形（24 位）



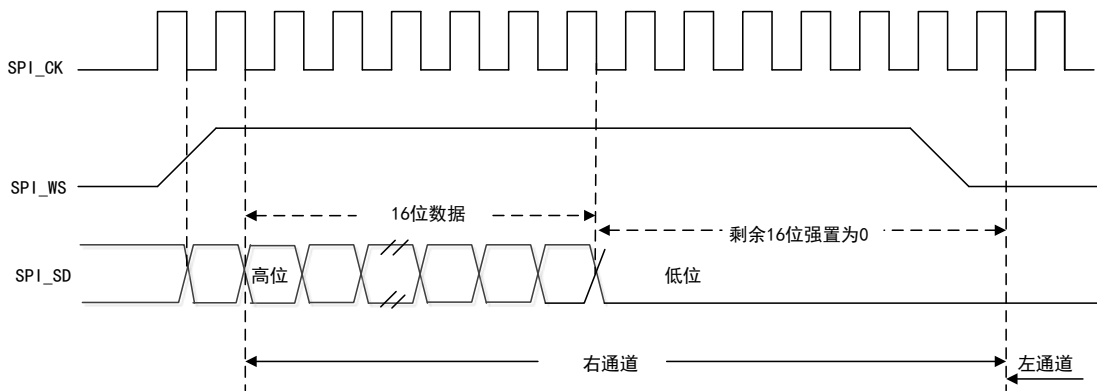
在 I2S 飞利浦标准下，如果要发送/接收 24 位和 32 位的数据需要对 SPI_DATA 寄存器进行两次读/写操作；例如：

- 需要发送 0x9FBB88（24 位数据时），第一次将 0x9FBB 写入 SPI_DATA 寄存器，第二次将 0x88XX 写入寄存器。
- 需要接收 0x9FBB88（24 位数据时），第一次从 SPI_DATA 寄存器中读出 0x9FBB，第二次从寄存器中读出 0x8800。

在 I2S 配置时，选择将 16 位数据扩展到 32 位数据帧的帧格式时，需要访问一次 SPI_DATA 寄存器，剩下的 16 位数据会被硬件强制置位 0x0000；例如：

- 需要接收或者发送的数据是 0x62D8 扩展到 32 位是 0x62D80000，需要将 0x62D8 写入 SPI_DATA 寄存器或者从 SPI_DATA 寄存器中读出。

图 100 I2S 飞利浦协议波形（16 位扩展至 32 位）



在发送过程中，要将 MSB 写入寄存器 SPI_DATA，在 TXBEFLG 标志位置 1 时可以写入新数据；如果有相应的中断，可以产生中断。

在接收过程中，每收到 MSB，RXBNEFLG 标志位置 1；如果还有相应的中断，可以产生中断。

23.6.1.2 MSB 对齐标准

在 MSB 标准中，WS 信号和第一个数据位同时产生

在发送过程中，在时钟信号的下降沿改变数据；在接收过程中，在时钟信号的上升沿读取数据。

图 101 MSB 对齐标准波形（16/32 位数据）

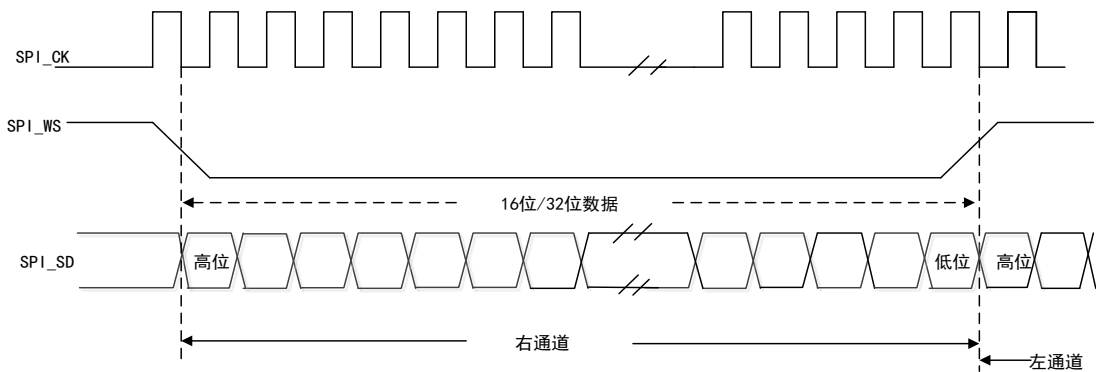


图 102 MSB 对齐标准波形（24 位数据）

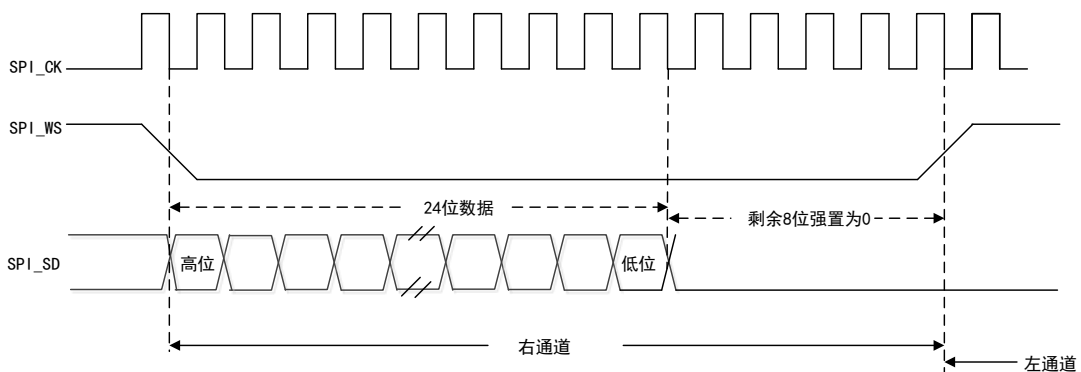
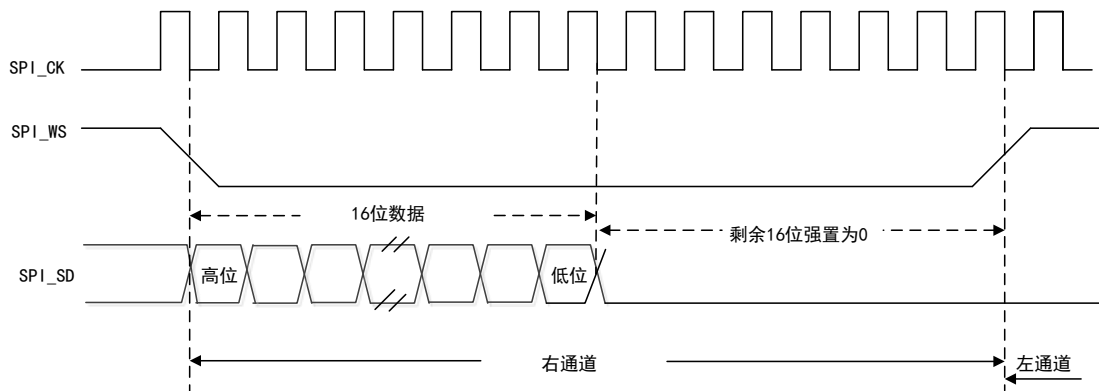


图 103 MSB 对齐标准波形（16 位扩展至 32 位）



23.6.1.3 LSB 对齐标准

LSB 对齐标准的发送过程中，在时钟信号的下降沿改变数据；在接收过程中，在时钟信号的上升沿读取数据。当通道长度和数据长度一样时，LSB 对齐标准和 MSB 对齐标准相同，如果通道长度大于数据长度，LSB 对齐标准的有效数据与最低位对齐。

图 104 LSB 对齐标准波形（16/32 位数据）

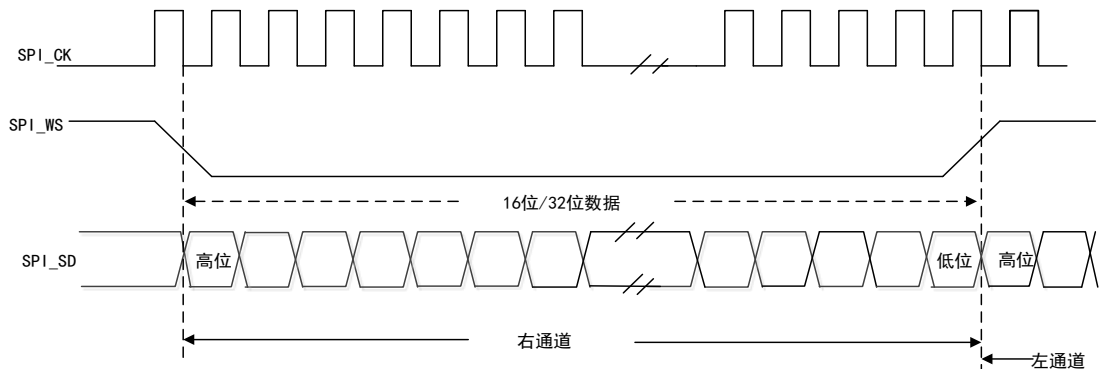
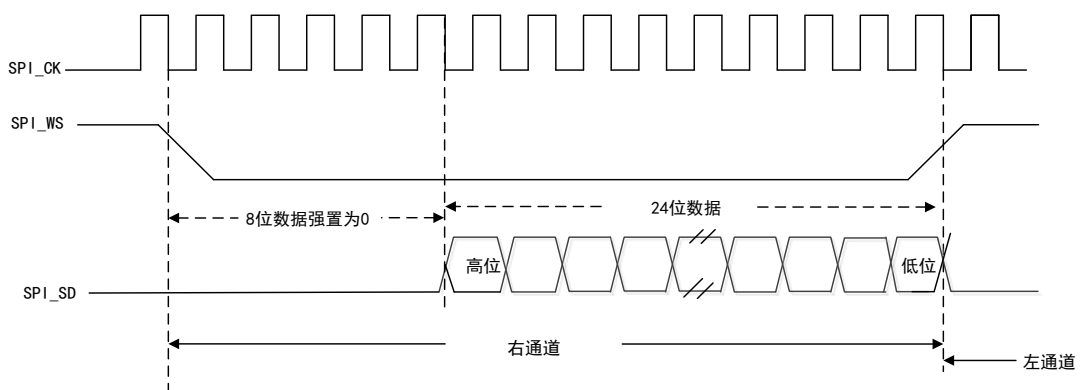


图 105 LSB 对齐标准波形（24 位数据）



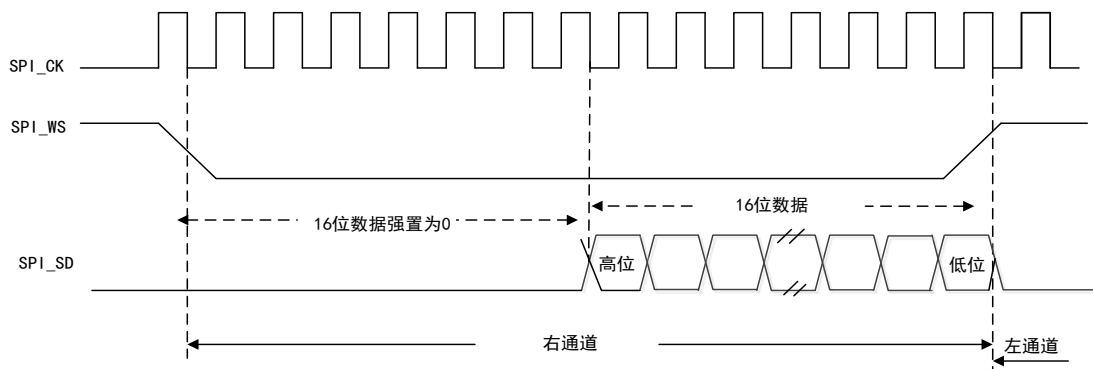
在发送过程中，如果要发送/接收 24 位数据时，需要对 SPI_DATA 寄存器进行 2 次读/写操作；例如：

- 需要发送 0x56EA98 时，第一次将 0xXX56 写进 SPI_DATA 寄存器，第二次将 0xEA98 写进 SPI_DATA 寄存器。
- 需要接收 0x56EA98 时，第一次将 0x0056 从 SPI_DATA 寄存器中读出，第二次将 0xEA98 从 SPI_DATA 寄存器中读出。

在 I2S 配置时，选择将 16 位数据扩展到 32 位数据帧的帧格式时，需要访问一次 SPI_DATA 寄存器，高 16 位数据会被硬件强制置位 0x0000；例如：

- 需要接收或者发送的数据是 0x98A5 扩展到 32 位是 0x000098A5，需要将 0x98A5 写入 SPI_DATA 寄存器或者从 SPI_DATA 寄存器中读出。

图 106 LSB 对齐标准下（16 位扩展至 32 位）

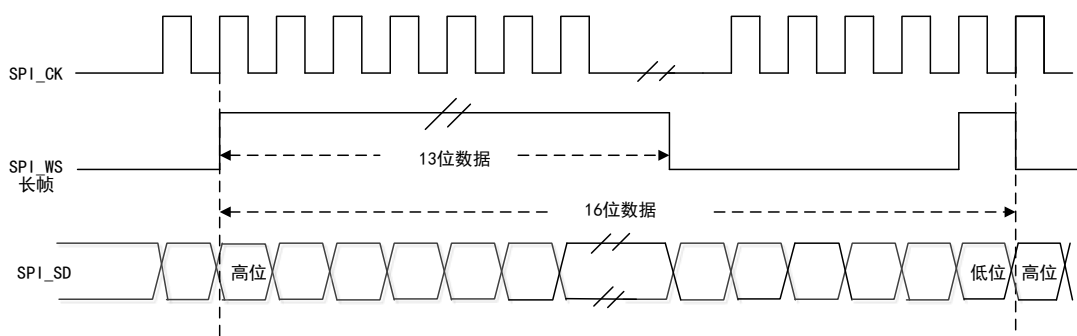


23.6.1.4 PCM 标准

在 PCM 标准中，没有声道选择。通过配置 SPI_I2SCFG 寄存器中的 PFSSEL 位来选择 PCM 标准的短帧和长帧。

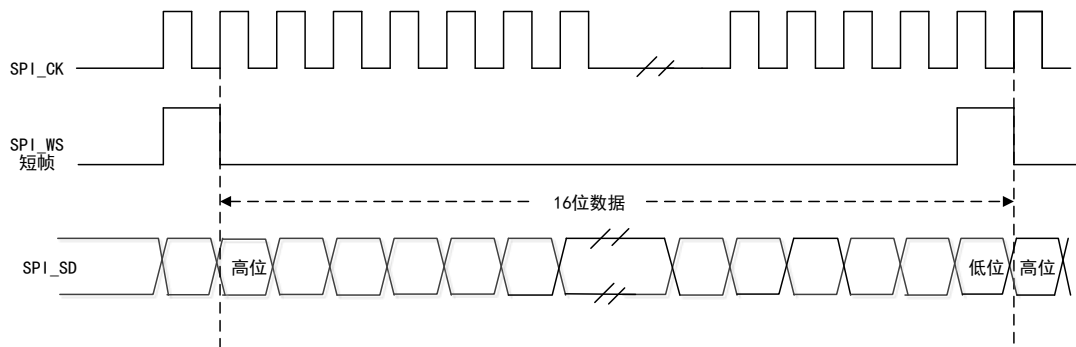
长帧结构在主模式中，同步的 WS 信号有效的时间为 13 位。

图 107 PCM 标准波形



短帧结构在主模式中，同步的 WS 信号长度有 1 位。

图 108 PCM 标准波形



23.6.2 I2S 时钟

I2SxCLK 的时钟来源为系统时钟（AHB 时钟的 HSICLK、HSECLK 或者 PLL）

I2S 的比特率确定了 I2S 数据线上的数据流和 I2S 的时钟信号频率。

- I2S 比特率=每个声道的比特数×声道数目×音频采样频率
- 有左右两声道是 16 位音频信号：I2S 比特率=16×2×Fs

音频的采样频率（Fs）和 I2S 比特率（I2S）的关系由以下公式定义：

表格 84 音频采样频率（Fs）公式

MCOEN	CHLEN	音频采样频率（Fs）
1	0	$I2SxCLK / [(16 \times 2) * ((2 * I2SPSC) + ODDPSC) * 8]$
1	1	$I2SxCLK / [(32 \times 2) * ((2 * I2SPSC) + ODDPSC) * 4]$
0	0	$I2SxCLK / [(16 \times 2) * ((2 * I2SPSC) + ODDPSC)]$
0	1	$I2SxCLK / [(32 \times 2) * ((2 * I2SPSC) + ODDPSC)]$

23.6.3 I2S 模式

表格 85 I2S 运行模式

运行模式	SD	WS	CK	MCK
主机发送	输出	输出	输出	输出/不使用
主机接收	输入	输出	输出	输出/不使用
从机发送	输出	输入	输入	输出/不使用
从机接收	输入	输入	输入	输出/不使用

23.6.3.1 I2S 主模式配置流程

- 配置 SPI_I2SPSC 寄存器的 I2SPSC 位和 ODDPSC 位，定义与音频采样频率相符的串行时钟波特率和实际分频系数。
- 配置 SPI_I2SCFG 寄存器的 CPOL 位，定义 SPI 在空闲状态的时钟极性。
- 配置 SPI_I2SCFG 寄存器的 I2SMOD 位激活 I2S 功能，配置 SPI_I2SCFG 寄存器的 I2SMOD 和 PFSSEL 位，选择 I2S 标准，配置

SPI_I2SCFG 寄存器的 DATALEN 位选择声道的数据位数，进而 I2SMOD 位选择 I2S 主模式以及是发送端/接收端。

- 配置 SPI_CTRL2 寄存器选择是否开启中断和 DMA 功能（可以选择是否需要）。
- 将 WS 引脚和 CK 引脚配置成输出模式，当 SPI_I2SPSC 的 MCOEN 位为 1 时，MCK 引脚也要配置成输出模式。
- 通过配置 SPI_I2SCFG 的 I2SMOD 的位，设置 I2S 的运行模式。
- 将 SPI_I2SCFG 寄存器的 I2SEN 位置 1。

23.6.3.2 I2S 主模式发送流程

当数据写入发送缓冲器时，此时开始发送，数据会从发送缓冲器传送到移位寄存器，此时 TXBEFLG 标志位置 1，其中 SCHDIR 标志位提示当前传输的数据所对应的声道。并且 SCHDIR 标志位的值会在 TXBEFLG 标志位为 1 时更新。

在发送第一位数据时，16 位的数据被并行的传送到 16 位移位寄存器中，之后串行的从引脚 MISO/SD 发出。下一个数据需要在 TXBEFLG 标志位为 1 时写入 SPI_DATA 寄存器，如果 SPI_CTRL2 的 TXBEIEN 的位为 1，则产生中断。

在当前数据传输完成之前，写入下一个待传输的数据，保证音频数据连续传输。

关闭 I2S 时，I2SEN 要等待标志位 TXBEFLG 为 1 且 BSYFLG 为 0 后清零。

23.6.3.3 I2S 主模式接收流程

RXBNEFLG 标志用来控制接收序列，RXBNEFLG 标志表示接收缓冲器是否为空，当填满接收缓存器后，RXBNEFLG 标志位置 1，如果配置了 SPI_CTRL2 寄存器的 RXBNEIEN 位，则发生中断，用户将数据从 SPI_DATA 寄存器中读出后，RXBNEFLG 标志位清零。一定要在读操作结束后，接收新数据，否则会发生上溢，OVRFLG 标志位被置 1。

在接收数据以后要立即更新 SCHDIR 的值，它取决于 I2S 产生的 WS 信号。

其中不管是哪种数据和声道长度，音频数据总是以 16 位的形式接收。根据配置的数据和声道长度，数据需要通过一次或者两次把数据传送到接收缓冲器中。

关闭 I2S 功能，不同的音频协议，数据长度和通道长度的操作步骤如下：

数据长度为 16 位，32 位通道长度（DATALEN=00, CHLEN=1, I2SSSEL=10）在 LSB 对齐模式下

- 等到倒数第二个 RXBNEFLG 置 1
- 等到 17 个 I2S 时钟周期（软件延迟）
- I2SEN 标志位清零

数据长度 16 位，32 位通道长度（DATALEN=00, CHLEN=1, I2SSSEL=10）在 MSB 对齐模式下

- 等到最后一个 RXBNEFLG 置 1
- 等到 1 个 I2S 时钟周期（软件延迟）
- I2SEN 标志位清零

其他的所有情况

- 等到倒数第二个 RXBNEFLG 置 1
- 等到 1 个 I2S 时钟周期（软件延迟）
- I2SEN 标志位清零

在数据传输期间 BSYFLG 标志时钟为低

23.6.3.4 I2S 从模式配置流程

从模式的配置方法和主模式的配置方法基本一样，在从模式中，不需要 I2S 提供时钟，是由外部 I2S 设备提供时钟信号和 WS 信号。

- 配置 SPI_I2SCFG 寄存器的 I2SMOD 位激活 I2S 功能。
- 配置 SPI_I2SCFG 寄存器的 I2SSSEL 位选择使用的 I2S 标准；配置 SPI_I2SCFG 寄存器的 DATALEN[1:0]位选择数据的比特数，配置 SPI_I2SCFG 寄存器的 CHLEN 位选择每个声道的数据位数，配置 SPI_I2SCFG 寄存器的 I2SMOD 位选择 I2S 从模式是发送端/接收端。
- 配置 SPI_CTRL2 寄存器选择是否开启中断和 DMA 功能（可以选择是否需要）。
- 将 SPI_I2SCFG 寄存器的 I2SEN 位置 1。

23.6.3.5 I2S 从模式发送流程

使能从设备，将数据写入 I2S 数据寄存器中，外部主设备开始通信，外部主设备发送时钟信号，当数据传输开始时，发送流程开始。

发送第一位数据时，16 位的数据并行的传送到 16 位移位寄存器中，然后串行的从引脚 MOSI/SD 发出，当数据从数据寄存器传送到移位寄存器中时，TXBEFLG 标志位置 1，如果此时设置了 SPI_CTRL2 寄存器的 TXBEIEN 位，则会产生中断。为了保证数据传输的连续性，要在数据完成传输之前，向 SPI_DATA 寄存器写入下一个数据，否则会发生“下溢”，UDRFLG 标志位置 1。

SPI_STS 寄存器的 SCHDIR 位表明了传输的数据所对应的声道，在从模式中 SCHDIR 位是由外部主设备 WS 信号决定的。

I2S 的 MSB 和 LSB 对齐模式，第一个写进数据寄存器的数据对应左声道的数据。

关闭 I2S，需要等待 TXBEFLG 标志位置 1，BSYFLG 标志位清零。

23.6.3.6 I2S 从模式接收流程

RXBNEFLG 标志用来控制接收序列，RXBNEFLG 标志表示接收缓冲器是否为空，当填满接收缓存器后，RXBNEFLG 标志位置 1，如果配置了 SPI_CTRL2 寄存器的 RXBNEIEN 位，则发生中断，将数据从 SPI_DATA 寄存器中读出后，RXBNEFLG 标志位清零，一定要在读操作结束后，接收新数据，否则会发生“上溢”，OVRFLG 标志位被置 1。

在接收数据以后要立即更新 SCHDIR 的值，它取决于 I2S 产生的 WS 信号。

其中不管是哪种数据和声道长度，音频数据总是以 16 位的形式接收。根据配置的数据和声道长度，数据需要通过一次或者两次把数据传送到接收缓冲器中。

关闭 I2S，在接收最后一个 RXBNEFLG 置 1 时，将 I2SEN 标志位清零。

23.6.4 I2S 中断

23.6.4.1 状态标志位

在 I2S 中有四个状态标志位监控 I2S 总线的状态。

发送缓存器空标志位 TXBEFLG

当 TXBEFLG 标志位为 1 时表明发送缓冲器为空，可以向发送缓冲器写入待发送数据，当写入数据后 TXBEFLG 标志位清 0。（关闭 I2S 时，TXBEFLG 标志位为 0）。

接收缓冲器非空标志位 RXBNEFLG

当 RXBNEFLG 标志位为 1 时表明接收缓冲器此时有待接受的数据，当对 SPI_DATA 寄存器进行读操作后，RXBNEFLG 标志位清零。

忙标志位 BSYFLG

BSYFLG 标志位为 1 时表明此时 I2S 正处于通讯状态（由硬件设置和清除），但是在主接收模式中，接收期间 BSYFLG 标志始终为 0。

当关闭 I2S 和数据传输结束时，BSYFLG 标志位清 0。

连续通信时

- 主发送模式时，传输期间，BSYFLG 标志位始终为高。
- 从模式时，每个数据项传输之间，BSYFLG 标志位在 1 个 I2S 时钟周期内置 0。

声道标志位 SCHDIR

发送模式中，SCHDIR 标志位在 TXBEFLG 标志位为高时刷新，表明在此时 SD 引脚上发送的数据所在的声道。如果在从模式的发送流程中发生了下溢错误，SCHDIR 标志位的值无效，需要把 I2S 功能关闭在打开之后开始通讯。

在接收模式中，SCHDIR 标志位在 SPI_DATA 寄存器接收到数据时刷新，表明接收的数据所在的声道，如果发生上溢错误，SCHDIR 标志位无效，需要把 I2S 功能关闭在打开之后开始通讯。

在 PCM 标准中，因为没有声道的选择，所以 SCHDIR 标志位无意义。

当 SPI_STS 寄存器的 OVRFLG 和 UDRFLG 标志位为 1 时，且 SPI_CTRL2 的 ERRIEN 位为 1，会产生中断。通过读取 SPI_STS 寄存器的值清除标志。

23.6.4.2 错误标志位

I2S 中含有两个错误标志位

下溢标志位 UDRFLG

在发送模式中，如果没有在数据发送之前，向 SPI_DATA 寄存器写入新的待发送数据，UDRFLG 标志位会置 1；此时如果 SPI_CTRL2 寄存器的 ERRIEN 位置 1，会产生中断。

此标志位在 SPI_I2SCFG 的 I2SMOD 位置 1 后，才会生效。

通过读 SPI_STS 寄存器清除 UDRFLG 标志位。

上溢标志位 OVRFLG

在接收模式中，如果在没有读出数据之前又接收到新的数据，OVRFLG 标志位会置 1。此时如果 SPI_CTRL2 寄存器的 ERRIEN 位置 1，会产生中断表明发生了错误。

读取 SPI_DATA 寄存器返回最后一个正确接收的数据，其他所有新接收的数据都会丢失。

可以通过先读 SPI_STS 寄存器然后再读 SPI_DATA 寄存器清除 OVRFLG 标志。

帧错误标志 FREFLG

仅当 I2S 配置为从模式下时，该位才可被硬件置 1。如果在传输数据帧的过程中，WS 线信号被改变，则该标志为被置 1。此时如果 SPI_CTRL2 寄存器的 ERRIEN 位置 1，会产生中断。

通过读 SPI_STS 寄存器清除 FRECFG 标志位。

表格 86 I2S 中断请求

中断标志	中断事件	使能控制位	清除方法
TXBEFLG	发送缓冲器空标志	TXBEIEN	写 SPI_DATA 寄存器
RXBNEFLG	接收缓冲器非空标志	RXBNEIEN	读 SPI_DATA 寄存器
OVRFLG	下溢标志位	ERRIEN	读 SPI_STS 寄存器
UDRFLG	上溢标志位		读 SPI_STS 寄存器 再读 SPI_DATA 寄存器
FREFLG	帧格式错误标志位		读 SPI_STS 寄存器

23.6.4.3 DMA 功能

在 I2S 模式中 DMA 的工作方式除了 CRC 功能没有之外和 SPI 相同。

23.7 寄存器地址映射

表格 87 SPI 和 I2S 寄存器地址映射

寄存器名	描述	偏移地址
SPI_CTRL1	SPI 控制寄存器 1	0x00
SPI_CTRL2	SPI 控制寄存器 2	0x04
SPI_STS	SPI 状态寄存器	0x08
SPI_DATA	SPI 数据寄存器	0x0C
SPI_CRCPOLY	SPI CRC 多项式寄存器	0x10
SPI_RXCRC	SPI 接收 CRC 寄存器	0x14
SPI_TXCRC	SPI 发送 CRC 寄存器	0x18
SPI_I2S_CFG	SPI_I2S 配置寄存器	0x1C
SPI_I2SPSC	SPI_I2S 预分频寄存器	0x20

23.8 寄存器功能描述

可以用半字（16 位）或字（32 位）的方式操作这些外设寄存器。

23.8.1 SPI 控制寄存器 1（SPI_CTRL1）（I2S 模式下不使用）

偏移地址：0x00

复位值：0x0000

位/域	名称	R/W	描述
0	CPHA	R/W	配置时钟相位（Clock Phase Configure） 该位表明在几个时钟边沿开始采样 0：在第 1 个时钟边沿 1：在第 2 个时钟边沿 注：通信进行时，不能修改该位。除了在 TI 模式下使用 CRC 功能，该位在 I2S 模式和 SPI TI 模式不使用。
1	CPOL	R/W	配置时钟极性（Clock Polarity Configure） 当 SPI 处于空闲状态时，SCK 保持的状态。 0：SCK 低电平 1：SCK 高电平 注：通信进行时，不能修改该位。除了在 TI 模式下使用 CRC 功能，该位在 I2S 模式和 SPI TI 模式不使用。
2	MSMCFG	R/W	配置主/从模式（Master/Salve Mode Configure） 0：配置为从模式 1：配置为主模式 注：通信进行时，不能修改该位。

位/域	名称	R/W	描述
5:3	BRSEL	R/W	选择波特率分频系数 (Baud Rate Divider Factor Select) 000: DIV=2 001: DIV=4 010: DIV=8 011: DIV=16 100: DIV=32 101: DIV=64 110: DIV=128 111: DIV=256 波特率=Fmaster/DIV 注: 通信进行时, 不能修改该位
6	SPIEN	R/W	使能 SPI 设备 (SPI Device Enable) 0: 禁用 1: 使能 注: 当关闭 SPI 设备时, 请按照关闭 SPI 的过程操作。
7	LSBSEL	R/W	选择 LSB 首位传输 (LSB First Transfer Select) 0: 先发送最高有效位 (MSB) 1: 先发送最低有效位 (LSB)
8	ISSEL	R/W	选择内部从设备 (Internal Slave Device Select) 决定 NSS 引脚上的电平 当 CTRL1_SSEN=1 时设置该位才有效。
9	SSEN	R/W	使能软件从设备 (Software Slave Device Enable) 0: 禁止 1: 启用 当 SSEN 置位时, NSS 引脚的电平由 SSEN 决定。
10	RXOMEN	R/W	使能仅接收模式 (Receive Only Mode Enable) 0: 同时发送和接收 1: 仅接收模式 RXOMEN 位和 BMEN 位一起决定了双线双向模式下的传输方向, 在多个从设备的配置中, 为了避免数据传输冲突的发生, 需要在未被访问的从设备上使 RXOMEN 位置 1。
11	CRCLSEL	R/W	选择 CRC 长度 (CRC Length Select) 0: 使用 8 位 CRC 1: 使用 16 位 CRC 注: 只有在 SPIEN=0 时, 才能写该位, 否则出错。
12	CRCNXT	R/W	使能下一个传输值是 CRC (CRC Transfer Next Enable) 0: 下一个传输的值来自发送缓冲区 1: 下一个传输的值来自发送 CRC 寄存器 注: 在 SPI_DATA 寄存器写入最后一个数据后, 马上设置 CRCNXT 位。
13	CRGEN	R/W	使能 CRC 校验 (CRC Calculate Enable) 0: 禁用 CRC 校验 1: 使能 CRC 校验 CRC 校验功能仅应用于全双工模式; 只有在 SPIEN=0 时, 才能改变该位。

位/域	名称	R/W	描述
14	BMOEN	R/W	使能双向模式的输出 (Bidirectional Mode Output Enable) 0: 禁止 (仅接收模式) 1: 使能 (仅发送模式) BMEN=1 即单线双向模式下, 该位决定传输线的传输方向。
15	BMEN	R/W	使能双向模式 (Bidirectional Mode Enable) 0: 双线双向模式 1: 单线双向模式 单线双向传输指: 数据主机的 MOSI 引脚和从机的 MISO 引脚之间传输。

23.8.2 SPI 控制寄存器 2 (SPI_CTRL2)

偏移地址: 0x04

复位值: 0x0700

位/域	名称	R/W	描述
0	RXDEN	R/W	使能接收缓冲区 DMA (Receive Buffer DMA Enable) 当 RXDEN=1 时, RXBNEFLG 标志一旦被置位就发出 DMA 请求。 0: 禁止 1: 启动
1	TXDEN	R/W	使能发送缓冲区 DMA (Transmit Buffer DMA Enable) 当该位被设置时, TXBEFLG 标志一旦被置位就发出 DMA 请求。 0: 禁止 1: 启动
2	SSOEN	R/W	使能 SS 输出 (SS Output Enable) SS 输出在主模式下 0: 禁止 SS 输出, 可以工作在多主机模式。 1: 开启 SS 输出, 不能工作在多主机模式。 注: I2S 和 SPI 的 TI 模式下不可用。
3	NSSPEN	R/W	使能 NSS 脉冲管理 (NSS Pulse Management Enable) 0: 禁止 1: 使能 注: 在连续传输时, 允许两个数据传输之间产生 NSS 脉冲。 当处于单数据传输时, 传输结束后 NSS 引脚强制拉高。 当 CPHA=1 或者 FRFCFG=1 时, 该位无效。 只有在 SPIEN=0 时, 才能写该位。 I2S 和 SPI 的 TI 模式下不可用。
4	FRFCFG	R/W	配置帧格式 (Frame Format Configure) 0: SPI 摩托罗拉模式 1: SPI TI 模式 注: 只有在 SPIEN=0 时, 才能写该位。 I2S 模式下不可用。
5	ERRIEN	R/W	使能错误中断 (Error Interrupt Enable) 0: 禁止 1: 使能 产生错误时, ERRIEN 位控制是否产生中断。

位/域	名称	R/W	描述
6	RXBNEIEN	R/W	使能接收缓冲区非空中断（Receive Buffer Not Empty Interrupt Enable） 0：禁止 1：允许 当 RXBNEFLG 标志位置 1 时产生中断请求
7	TXBEIEN	R/W	使能发送缓冲区空中断（Transmit Buffer Empty Interrupt Enable） 0：禁止 1：使能 当 TXBEFLG 标志位置 1 时产生中断请求
11:8	DSCFG	R/W	配置数据位宽（Data Size Configure） 配置 SPI 传输数据的位宽： 0000：保留 0001：保留 0010：保留 0011：4 位 0100：5 位 0101：6 位 0110：7 位 0111：8 位 1000：9 位 1001：10 位 1010：11 位 1011：12 位 1100：13 位 1101：14 位 1110：15 位 1111：16 位 注： 当软件写入了保留位，此时的值会强制位 0111（8 位） I2S 模式下不使用
12	FRTCFCG	R/W	配置 FIFO 接收阈值（FIFO Reception Threshold Configure） 配置 FIFO 阈值，当超过此阈值时，产生 RXBNEFLG 0：16 位 1：8 位 注：I2S 模式下不可用
13	LDRX	R/W	最后一次 DMA 接收（Last DMA Receive） 这些位在数据打包模式中使用，用来定义 DMA 接收的总数是奇数还是偶数。 0：偶数 1：奇数 注： 只有当设置了 SPI_CTRL2 寄存器的 RXDEN 位，并且开启了打包模式这些位才有意义。 只有在 SPIEN=0 时，才能写该位。 需要按照“23.5.10”步骤，关闭 SPI。 I2S 模式下不使用。
14	LDTX	R/W	最后一次 DMA 发送（Last DMA Transmit）

位/域	名称	R/W	描述
			<p>这些位在数据打包模式中使用，用来定义 DMA 发送的总数是奇数还是偶数。</p> <p>0: 偶数 1: 奇数</p> <p>注： 只有当设置了 SPI_CTRL2 寄存器的 RXDEN 位，并且开启了打包模式这些位才有意义。 只有在 SPIEN=0 时，才能写该位。 需要按照“23.5.10”步骤，关闭 SPI。 I2S 模式下不使用。</p>
15			保留

23.8.3 SPI 状态寄存器 (SPI_STS)

偏移地址: 0x08

复位值: 0x0002

位/域	名称	R/W	描述
0	RXBNEFLG	R	<p>接收缓冲非空标志 (Receive Buffer Not Empty Flag)</p> <p>该位表示接收缓冲器是否为空</p> <p>0: 空 1: 非空</p>
1	TXBEFLG	R	<p>发送缓冲器为空标志 (Transmit Buffer Empty Flag)</p> <p>该位表示发送缓冲器是否为空</p> <p>0: 非空 1: 空</p>
2	SCHDIR	R	<p>声道方向标志 (Sound Channel Direction Flag)</p> <p>0: 表明左声道正在传输或者接收需要的数据 1: 表明右声道正在传输或者接收需要的数据</p> <p>注: 在 SPI 模式下不使用，在 PCM 模式下无左右声道。</p>
3	UDRFLG	R	<p>发生欠载标志 (Underrun Occur Flag)</p> <p>该位表示是否发生下溢</p> <p>0: 未发生 1: 发生</p> <p>该标志位由硬件置位，软件复位。 在 SPI 模式下不使用</p>
4	CRCEFLG	RC_W0	<p>发生 CRC 错误标志 (CRC Error Occur Flag)</p> <p>该位表示接收的 CRC 值和 RXCRC 寄存器的值是否匹配</p> <p>0: 匹配 1: 不匹配</p> <p>该位由硬件置位，软件复位。 在 I2S 模式下不使用</p>
5	MEFLG	R	<p>发生模式错误标志 (Mode Error Occur Flag)</p> <p>该位表示是否发生模式错误</p> <p>0: 未发生 1: 发生</p> <p>由硬件置位，软件复位。 在 I2S 模式下不使用</p>

位/域	名称	R/W	描述
6	OVRFLG	R	发生过载标志 (Overrun Occur Flag) 该位表示是否发生溢出 0: 未发生 1: 发生 由硬件置位, 软件复位。
7	BSYFLG	R	忙标志 (Busy Flag) 该位表示 SPI 的工作状态 0: SPI 空闲 1: SPI 正在通信 由硬件置位或者复位
8	FREFLG	R	帧格式错误标志 (Frame Format Error Flag) 0: 未发生 1: 发生 注: 该位由硬件置 1, 读取 SPI_STS 寄存器时清 0。
10:9	FRLSEL	R	选择 FIFO 接收水平 (FIFO Receive Level Select) 00: FIFO 空 01: FIFO/4 10: FIFO/2 11: FIFO 满 注: 该位由硬件置 1 和清 0。I2S 模式和有 CRC 校验的 SPI 单接收模式下不使用。
12:11	FTLSEL	R	选择 FIFO 发送水平 (FIFO Transmit Level Select) 00: FIFO 空 01: FIFO/4 10: FIFO/2 11: FIFO 满 (FIFO 阈值大于 1/2 时认为是满) 注: 该位由硬件置 1 和清 0。I2S 模式下不使用
15:13	保留		

23.8.4 SPI 数据寄存器 (SPI_DATA)

偏移地址: 0x0C

复位值: 0x0000

位/域	名称	R/W	描述
15:0	DATA	R/W	发送接收数据寄存器 (Transmit Receive Data Register) 存储待发送或者接收到的数据。 写该寄存器时, 会将数据写入发送缓冲区; 读该寄存器时, 是读取接收缓冲区的数据。 缓冲区的大小和数据帧的长度保持一致, 即对于 8 位数据, 发送、接收数据时会用到 DATA[7:0], DATA[15:8]无效; 对于 16 位数据, 发送、接收时会用到 DATA[15:0]。

23.8.5 SPI CRC 多项式寄存器 (SPI_CRCPOLY) (I2S 模式下不使用)

偏移地址: 0x10

复位值: 0x0007

位/域	名称	R/W	描述
15:0	CRCPOLY	R/W	设置 CRC 多项式数值 (CRC Polynomial Value Setup) 该寄存器包含了 CRC 计算 CRC 的多项式, 可以修改, 其复位值为 0x0007。

23.8.6 SPI 接收 CRC 寄存器 (SPI_RXCRC) (I2S 模式下不使用)

偏移地址: 0x14

复位值: 0x0000

位/域	名称	R/W	描述
15:0	RXCRC	R	接收数据的 CRC 数值 (Receive Data CRC Value) 硬件计算接收字节的 CRC 数据存储在寄存器; 位数和数据帧的长度一致, 即如果接收数据是 8 位的, CRC 计算按 CRC8 方式; 如果接收数据是 16 位的, CRC 计算按照 CRC16 计算。 当 CRCEN 置位时, 硬件清除该寄存器。 注意: 当 BSYFLG 位置 1 时, 读取 RXCRC 寄存器数值有可能错误。

23.8.7 SPI 发送 CRC 寄存器 (SPI_TXCRC) (I2S 模式下不使用)

偏移地址: 0x18

复位值: 0x0000

位/域	名称	R/W	描述
15:0	TXCRC	R	发送数据的 CRC 数值 (Transmit Data CRC Value) 硬件计算发送字节的 CRC 数据存储在 TXCRC 中; 尾位数和数据帧的长度一致, 即如果发送数据是 8 位的, CRC 按照 CRC8 方式; 如果发送数据是 16 位的, CRC 计算按 CRC16 计算。 注意: 当 BSYFLG 位置 1 时, 读取 TXCRC 寄存器数值有可能错误。

23.8.8 SPI_I2S 配置寄存器 (SPI_I2SCFG) (在 SPI 模式下不可用)

偏移地址: 0x1C

复位值: 0x0000

位/域	名称	R/W	描述
0	CHLEN	R/W	配置声道长度 (Channel Length Configure) 声道长度是指每个音频声道的数据位数 0: 16 位宽 1: 32 位宽 该位在 DATALEN=00 时, 写操作才有意义, 否则声道长度由硬件固定为 32 位。 注: 该位在关闭了 I2S 时才能被设置。 在 SPI 模式下不使用。
2:1	DATALEN	R/W	配置待传输数据长度 (Length of Data To Be Transferred Configure) 00: 16 位数据长度 01: 24 位数据长度 10: 32 位数据长度 11: 不允许 注: 该位在关闭了 I2S 时才能被设置。 在 SPI 模式下不使用。

位/域	名称	R/W	描述
3	CPOL	R/W	选择静止态时钟极性 (Steady State Clock Polarity Select) I2S 时钟静止态时的电平状态 0: 低电平 1: 高电平 注: 该位在关闭了 I2S 时才能被设置。 在 SPI 模式下不使用。
5:4	I2SSSEL	R/W	配置 I2S 标准 (I2S Standard Select) 00: I2S 飞利浦标准 01: 高字节对齐标准 (左对齐) 10: 低字节对齐标准 (右对齐) 11: PCM 标准 注: 该位在关闭了 I2S 时才能被设置。 在 SPI 模式下不使用。
6	保留		
7	PFSSEL	R/W	选择 PCM 帧同步模式 (PCM Frame Synchronization Mode Select) 0: 短帧同步 1: 长帧同步 注: 该位只在 I2SSSEL=11 时有意义。 在 SPI 模式下不使用。
9:8	I2SMOD	R/W	配置 I2S 主/从、发送/接收模式 (I2S Master/Slave Transmit/Receive Mode Configure) 00: 从设备发送 01: 从设备接收 10: 主设备发送 11: 主设备接收 注: 该位在关闭了 I2S 时才能被设置。 在 SPI 模式下不使用。
10	I2SEN	R/W	使能 I2S (I2S Enable) 0: 关闭 I2S 1: 使能 I2S 注: 在 SPI 模式下不使用。
11	MODESEL	R/W	选择 SPI/I2S 模式 (SPI/I2S Mode Select) 0: 选择 SPI 模式 1: 选择 I2S 模式 注: 该位只有在关闭了 SPI 或者 I2S 时才能设置。
15:12	保留		

23.8.9 SPI_I2S 预分频寄存器 (SPI_I2SPSC) (在 SPI 模式下不使用)

偏移地址: 0x20

复位值: 0x0002

位/域	名称	R/W	描述
7:0	I2SPSC	R/W	配置 I2S 线性预分频系数 (I2S linear prescaler Factor Configure) 禁止设置 I2SPSC [7:0]=0 或者 I2SPSC [7:0]=1 注: 该位在关闭了 I2S 时才能被设置。该位只在 I2S 主设备模式下使用。 在 SPI 模式下不使用。

位/域	名称	R/W	描述
8	ODDPSC	R/W	配置预分频系数为奇数 (Configure the prescaler factor to be odd) 0: 实际分频系数=I2SPSC*2 1: 实际分频系数= (I2SPSC*2) +1 注: 该位在关闭了 I2S 时才能被设置。该位只在 I2S 主设备模式下使用。 在 SPI 模式下不使用。
9	MCOEN	R/W	使能主设备时钟输出 (Master Device clock output enable) 0: 关闭 1: 使能 注: 该位在关闭了 I2S 时才能被设置。该位只在 I2S 主设备模式下使用。 在 SPI 模式下不使用。
15:10	保留		

24 模拟数字转换器（ADC）

24.1 简介

12 位精度的 ADC，共 19 个通道，16 个外部通道和 3 个内部通道，各通道 A/D 转换模式有单次、连续和断续，ADC 转换结果可以左对齐或右对齐存储在 16 位数据寄存器中。

24.2 主要特征

- (1) ADC 供电要求：2.4V 到 3.6V
- (2) ADC 输入范围： $V_{SSA} \leq V_{IN} \leq V_{DDA}$
- (3) 转换模式
 - 单次转换模式
 - 连续转换模式
 - 断续模式
- (4) 模拟输入通道类别
 - 外部 GPIO 输入通道
 - 1 个内部温度传感（ V_{SENSE} ）输入通道
 - 1 个内部参考电压（ V_{REFINT} ）输入通道
 - 1 个（ $1/2 \cdot V_{DD}$ ）引脚电压输入通道
- (5) 高性能
 - 12 位，10 位，8 位或 6 位可配置分辨率。
 - 自校准
 - 可编程采样时间
 - 数据对齐
 - DMA 支持
- (6) 低功耗
 - 低功耗运行降低了 PCLK 频率同时保持最佳 ADC 性能
 - 自动延时模式：运行在 PCLK 低速下，防止 ADC 超限
 - 自动关闭模式：ADC 除了转换期间，其他时间自动断电
- (7) 中断
 - 转换结束中断
 - 序列转换结束中断
 - 采样阶段结束中断
 - ADC 准备好中断
 - 过载中断
 - 模拟看门狗状态复位中断
- (8) 触发方式

- 外部引脚信号触发
 - 片内定时器产生的内部信号触发
- (9) 数据寄存器
- 19 个数据寄存器，用来存储序列分段采样的结果

24.3 功能描述

24.3.1 ADC 引脚和内部信号

表格 88 ADC 内部信号

名称	说明	信号类型
TMRx_TRG	从定时器来的内部信息	输入
VSENSE	内部温度传感器输出电压	输入
VREFINT	内部参考电压的输出	输入
1/2*VDD	1/2*VDD 引脚输入电压	输入

表格 89 ADC 引脚

名称	说明	信号类型
VDDA	模拟供电电源，ADC 参考电压的正极， $V_{DDA} \geq V_{DD}$	输入，模拟供电电源
VSSA	模拟地， $V_{SSA} = V_{SS}$	输入，模拟电源地
ADC_IN[15:0]	16 路模拟输入	模拟输入信号

24.3.2 校准

校准的作用是消除各芯片 A/D 转换的偏移误差，所以应在 A/D 转换前进行校准，在校准期间应用不能使用 ADC 模块。

校准配置过程：

- 配置寄存器 ADC_CTRL 的 ADCEN 位为 0，禁用 ADC
- 配置寄存器 ADC_CTRL 的 CAL 位置 1，启动校准
- 校准完成后，CAL 位由硬件自动清零
- 校准因子在寄存器 ADC_DATA 的 CDATA[6:0]位中读取

24.3.3 ADC 转换模式

24.3.3.1 单次转换模式

该模式对单个通道而言之只对此通道进行一次转换，对多通道而言只对此组通道进行一次转换。

配置寄存器 ADC_CFG1 的 CMODESEL 位为 0 时，ADC 设置为单次转换模式；通过软件配置寄存器 ADC_CTRL 的 STARTCEN 位置 1 或硬件触发事件开启 ADC 转换。

每个通道转换结束后，转换数据将存储到 16 位 ADC_DATA 寄存器，EOCFLG 位置 1，若 EOCIEEN 位置 1，则产生中断。通道序列转换结束后，EOSEQFLG 位置 1，若 EOSEQIEN 位置 1，则产生中断。

24.3.3.2 连续转换模式

该模式对单通道而言之对此通道进行连续转换，对多通道而言只对此组通道进行连续转换。

配置寄存器 ADC_CFG1 的 CMODESEL 位置 1 时，ADC 设置为连续转换模式：通过软件配置寄存器 ADC_CTRL 的 STARTCEN 位置 1 或硬件触发事件开启 ADC 转换。

每个通道转换结束后，转换数据将存储到 16 位 ADC_DATA 寄存器，EOCFLG 位置 1，若 EOCIEN 位置 1，则产生中断。通道序列转换结束后，EOSEQFLG 位置 1，若 EOSEQIEN 位置 1，则产生中断。

24.3.3.3 断续模式

配置寄存器 ADC_CFG1 的 DISCEN 位置 1，ADC 设置为断续模式：通过软件或硬件的触发事件开启 ADC 转换。该模式下每次只转换一个序列中的一个通道，若 DISCEN 位清零，则每次转换一个序列的所有通道。

例如：

- DISCEN 位置 1，通道序列为 0,1,5
 - 1st 触发，通道 0 被转换并且产生一个 EOCFLG 事件
 - 2nd 触发，通道 1 被转换并且产生一个 EOCFLG 事件
 - 3rd 触发，通道 5 被转换并且产生一个 EOCFLG 事件
- DISCEN 位为 0，通道序列为 0,1,5
 - 1st 触发，通道 0,1,5 按顺序依次被转换，每个通道转换完产生一个 EOCFLG 事件，整个序列转换完成后产生一个 EOSEQFLG 事件

24.3.4 ADC 通道分类

24.3.4.1 GPIO 引脚引入的模拟输入通道

共 16 个通道连接到 ADC_IN0...ADC_IN15。

24.3.4.2 内部模拟输入通道

内部参考电压 VREFINT

- (1) 内部参考电压用来提供一个稳定的电压输出给 ADC
- (2) 内部参考电压 V_{REFINT} 选择 ADC1_IN17 输入通道

1/2*VDD 引脚电压

1/2*VDD 引脚输入电压选择 ADC1_IN18 输入通道。

温度传感器

- (1) 温度传感器用来测量芯片内部的温度
- (2) 温度传感器选择 ADC1_IN16 输入通道
- (3) 通过配置寄存器 ADC_CCFG 的 TSEN 位启动

- (4) 可选择采样时间
- (5) 支持的温度范围：-40~105℃

温度传感器用于测量产品的接点温度，传感器将电压值通过 ADC_IN16 传输到 ADC，通过 ADC 转换成数值。

配置过程

- 使能 ADC，选择输入通道
- 选择采样时间（17.1us）
- 通过配置 ADC_CCFG 寄存器的 TSEN 位使能温度传感器
- 通过配置 ADC_CTRL 寄存器的 STARTCEN 位开启转换
- 读取 ADC_DATA 寄存器中的转换结果
- 使用公式将结果转换为温度 T

$$T(^{\circ}\text{C}) = 25 + \frac{V_{25} - V_{\text{sensor}}}{\text{Slope}}$$

注：（1）V₂₅： 25℃下 V_{sensor} 的值，详见数据手册；

（2）Slope： V_{sensor} 和温度的平均斜率值（单位：mV/℃），详见数据手册。

（3）配置启动时间时，需要在同时配置 ADCEN 位 和 TSEN 位。

24.3.5 外部触发和触发极性

- （1） 序列分段采样功能第一段的外部触发以及其他功能的外部触发事件可通过配置寄存器 ADC_CFG1 的 EXTTRGSEL1 位选择。

表格 90 第一段外部触发

触发源	EXTTRGSEL	触发类型
TMR1_TRGO	0000	片内定时器产生的内部信号
TMR1_CC4	0001	
TMR2_TRGO	0010	
TMR3_TRGO	0011	
TMR4_TRGO	0100	
TMR1_TRGO2	0101	
TMR1_TRGO3	0110	
TMR2_TRGO2	0111	
TMR3_TRGO2	1000	
TMR4_TRGO2	1001	
保留	1010/1100/1101/1110/1111	

当寄存器 ADC_CFG1 的位 EXTPOLSEL1 ≠ “0b00”，外部事件可在其选择的极性上触发转换。

表格 91 配置触发极性

EXTPOLSEL	源
00	禁止触发检测
01	在上升沿时检测
10	在下降沿时检测

EXTPOLSEL	源
11	在上升沿和下降沿都检测

(2) 序列分段采样功能第二段的外部触发以及其他功能的外部触发事件可通过配置寄存器 ADC_CFG3 的 EXTTRGSEL2 位选择。

表格 92 第二段外部触发

触发源	EXTTRGSEL2	触发类型
TMR1_TRGO	0000	片内定时器产生的内部信号
TMR1_CC4	0001	
TMR2_TRGO	0010	
TMR3_TRGO	0011	
TMR4_TRGO	0100	
TMR1_TRGO2	0101	
TMR1_TRGO3	0110	
TMR2_TRGO2	0111	
TMR3_TRGO2	1000	
TMR4_TRGO2	1001	
保留	1010/1100/1101/1110/1111	外部引脚

当寄存器 ADC_CFG3 的位 EXTPOLSEL2 ≠ “0b00”，外部事件可在其选择的极性上触发转换。

表格 93 配置触发极性

EXTPOLSEL2	源
00	禁止触发检测
01	在上升沿时检测
10	在下降沿时检测
11	在上升沿和下降沿都检测

(3) 序列分段采样功能第三段的外部触发以及其他功能的外部触发事件可通过配置寄存器 ADC_CFG3 的 EXTTRGSEL3 位选择。

表格 94 第三段外部触发

触发源	EXTTRGSEL3	触发类型
TMR1_TRGO	0000	片内定时器产生的内部信号
TMR1_CC4	0001	
TMR2_TRGO	0010	
TMR3_TRGO	0011	
TMR4_TRGO	0100	
TMR1_TRGO2	0101	
TMR1_TRGO3	0110	
TMR2_TRGO2	0111	

触发源	EXTTRGSEL3	触发类型
TMR3_TRGO2	1000	
TMR4_TRGO2	1001	
保留	1010/1100/1101/1110/1111	外部引脚

当寄存器 ADC_CFG3 的位 EXTPOLSEL3 ≠ “0b00”，外部事件可在其选择的极性上触发转换。

表格 95 配置触发极性

EXTPOLSEL3	源
00	禁止触发检测
01	在上升沿时检测
10	在下降沿时检测
11	在上升沿和下降沿都检测

24.3.6 数据寄存器

数据对齐方式为左对齐或右对齐，配置寄存器 ADC_CFG1 的 DALIGCFG 位决定，DALIGCFG 为 0 右对齐，DALIGCFG 为 1 左对齐。ADC 转换结果可以左对齐或右对齐存储在 16 位数据寄存器中。

序列分段采样模式的数据存储在 ADC_DATA(0-18)的寄存器，可左右对齐。

例如：配置 3 段序列采样，第一段通道总数为 1（采样通道 0），第二段通道总数为 2（采样通道 1，0），第三段通道总数为 3（采样通道 2，3，1），（配置 ADC_SEQ_NUM = 0x20000820；ADC_CHANNEL1.CH0=0；ADC_CHANNEL1.CH1=1，ADC_CHANNEL1.CH2=0；ADC_CHANNEL1.CH3=2，ADC_CHANNEL1.CH4=3，ADC_CHANNEL1.CH5=1），则第一段的数据结果将出现在 ADC_DATA0，第二段的数据结果将出现在 ADC_DATA1-2，第三段的数据结果将出现在 ADC_DATA3-5，虽然 ADC 分段采样模式支持乱序（即第二、三段采样可先于第一段），但每段对应的结果寄存器的对应关系是固定的（例如，配置完成后开启 ADC，若 ADC 首先接收第三段采样对应的硬件触发，则该段采样结束后模拟通道 2 的结果将出现在 ADC_DATA3，模拟通道 3 的结果将出现在 ADC_DATA3，模拟通道 1 的结果将出现在 ADC_DATA5）。

24.3.7 可编程转换分辨率

降低分辨率可提高转换时间，通过配置寄存器 ADC_CFG1 的 DATARESCFG 位来选择 12、10、8 或 6 位模式。

表格 96 t_{SAR} 与转换分辨率有关的转换时间

DATARESCFG 位	t _{SAR}	t _{SAR} (ns)@f _{ADC} =14MHz	t _{SAMPL} (min)	t _{ADC}	t _{ADC} (μs)@f _{ADC} =14MHz
6	7.5	535ns	1.5	9	643ns
8	9.5	678ns	1.5	11	785ns

DATARESCFG 位	t _{SAR}	t _{SAR} (ns)@f _{ADC} =14MHz	t _{SAMPL} (min)	t _{ADC}	t _{ADC} (μs)@f _{ADC} =14MHz
10	11.5	821ns	1.5	13	928ns
12	12.5	893ns	1.5	14	1000ns

24.3.8 中断

表格 97 ADC 中断

中断事件	事件标志	使能控制
转换结束	EOCFLG	EOCIEN
序列转换结束	EOSEQFLG	EOSEQIEN
采样阶段结束	EOSMPFLG	EOSMPIEN
ADC 准备好	ADCRDYFLG	ADCRDYIEN
过载	OVREFLG	OVR IEN
模拟看门狗状态复位	AWDFLG	AWDIEN
序列分段采样第三段结束	SEQ_NUM3_FIN	SEQ_NUM3_FINIEN
序列分段采样第二段结束	SEQ_NUM2_FIN	SEQ_NUM2_FINIEN
序列分段采样第一段结束	SEQ_NUM1_FIN	SEQ_NUM1_FINIEN

24.3.9 ADC 过载

ADC 过载是指当转换好的数据未及时被 DMA 或 CPU 读取时，另一个转换数据开始有效。

在 EOCFLG 位为 1，但另一个新的转换完成时，产生过载事件，寄存器 ADC_STS 的 OVREFLG 位被置 1，若 OVR IEN 位置 1，则产生一个过载中断。

通过配置寄存器 ADC_CFG1 的 OVRMAG 位来决定发生过载事件时 ADC 数据寄存器的数据是被保持还是覆盖：

- OVRMAG 为 0：当检测到过载事件时，ADC_DATA 寄存器保持为老数据
- OVRMAG 置 1：当检测到过载事件时，ADC_DATA 寄存器用最后一次的转换数据覆盖

24.3.10 转换数据管理

24.3.10.1 无 DMA 参与管理转换数据

软件控制数据转换，每次转换完成后，EOCFLG 位置 1，从 ADC_DATA 寄存器读取转换结果。此时 ADC_CFG1 寄存器中的 OVRMAG 位应为 0。

24.3.10.2 无 DMA 且无过载参与管理转换数据

转换一个或多个通道并且每次转换结果不需要都读取时，OVRMAG 位置 1，过载事件不能阻止 ADC 转换且寄存器 ADC_DATA 只保存最后一次转换数据。

24.3.10.3 DMA 管理转换数据

使用 DMA 传输可以及时将转换结果从数据寄存器传输到存储器，以防 ADC_DATA 寄存器中的转换结果丢失。

配置寄存器 ADC_CFG1 的 DMAEN 位置 1 使能 DMA，每次转换结束后会产生一个 DMA 请求，将数据寄存器的转换数据传输到存储器中。

当 DMA 不能及时响应 DMA 请求时，将产生过载事件，OVREFLG 位置 1，之后 ADC 将不会产生 DMA 请求，DMA 不再传输新的转换结果。直到 OVREFLG 位清零，DMA 重新开始工作。

注：序列分段采样模式不支持 DMA 功能。

配置寄存器 ADC_CFG1 的 DMACFG 位选择 DMA 模式：

- DMACFG 为 0，DMA 单次模式
 - DMA 编程用来传输固定长度数据
 - 此模式下，ADC 每次转换数据有效产生一次 DMA 请求，当 ADC 转换重新启动，ADC 停止产生 DMA 请求
 - 当 ADC 转换次数到达 DMA 变成的长度时，需要使用软件配置 STOP 位来停止 ADC
- DMACFG 置 1，DMA 循环模式
 - DMA 编程为循环模式或双缓冲模式
 - 此模式下，ADC 转换重新启动且转换的数据有效时产生一次 DMA 请求

24.3.11 低功耗特性

24.3.11.1 自动延迟转换模式

此模式可以很大的降低应用功耗，适用于相对较少转换或转换请求时间间隔很长的应用。自动关断模式可与自动延迟转换模式联合使用于低频应用中。

配置寄存器 ADC_CFG1 的 AOEN 位置 1 时开启自动关断模式。AOEN 位置 1，ADC 无转换时将会自动断电，开始转换时 ADC 自动唤醒。

注：序列分段采样模式不支持。

24.3.11.2 自动关断模式

此模式可以很大的降低应用功耗，适用于相对较少转换或转换请求时间间隔很长的应用。自动关断模式可与自动延迟转换模式联合使用于低频应用中。

配置寄存器 ADC_CFG1 的 AOEN 位置 1 时开启自动关断模式。AOEN 位置 1，ADC 无转换时将会自动断电，开始转换时 ADC 自动唤醒。

注：序列分段采样模式不支持。

24.4 寄存器地址映射

表格 98 ADC 寄存器地址映射

寄存器名	描述	偏移地址
ADC_STS	ADC 状态寄存器	0x00
ADC_IEN	ADC 中断使能寄存器	0x04
ADC_CTRL	ADC 控制寄存器	0x08
ADC_CFG1	ADC 配置寄存器 1	0x0C
ADC_CFG2	ADC 配置寄存器 2	0x10
ADC_SMPTIM	ADC 采样时间寄存器	0x14
ADC_CFG3	ADC 配置寄存器 3	0x18
ADC_AWDT	ADC 看门狗阈值寄存器	0x20
ADC_CHSEL	ADC 通道选择寄存器	0x28
ADC_DATA	ADC 数据寄存器	0x40
ADC_DATAx (x=0~18)	ADC 序列分段采样数据寄存器	0x44+4x
ADC_SEQ_NUM	ADC 序列分段采样控制寄存器	0x100
ADC_CHANNEL1	ADC1-6 次传输的通道寄存器	0x104
ADC_CHANNEL2	ADC7-12 次传输的通道寄存器	0x108
ADC_CHANNEL3	ADC13-18 次传输的通道寄存器	0x10C
ADC_CHANNEL4	ADC 第 19 次传输的通道寄存器	0x110
ADC_CCFG	ADC 通用配置寄存器	0x308

注：上面的次数不是实际传输的次数，实际的次数是根据 ADC_SEQ_NUM 寄存器配置得到的。

24.5 寄存器功能描述

24.5.1 ADC 状态寄存器 (ADC_STS)

偏移地址：0x00

复位值：0x0000 0000

位/域	名称	R/W	描述
0	ADCRDYFLG	RC_W1	ADC 准备好标志 (ADC Ready Flag) 0: ADC 未准备好 1: ADC 已准备好开始转换
1	EOSMPFLG	RC_W1	采样结束标志 (End of Sampling Flag) 该位由硬件置 1, 软件清 0 0: 不在采样结束阶段 1: 达到采样阶段结束条件

位/域	名称	R/W	描述
2	EOCFLG	RC_W1	转换结束标志 (End of Conversion Flag) 该位由硬件置 1, 软件清 0 0: 转换未结束 1: 转换结束
3	EOSEQFLG	RC_W1	序列转换结束标志 (End of Sequence Flag) 该位由硬件置 1, 软件清 0 0: 序列转换未完成 1: 序列转换完成
4	OVREFLAG	RC_W1	产生 ADC 过载事件标志 (ADC Overrun Event Flag) 该位由硬件置 1, 软件清 0 0: 无过载事件产生 1: 产生了过载事件
6:5	保留		
7	AWDFLG	RC_W1	发生模拟看门狗标志 (Analog Watchdog Flag) 该位由硬件置 1, 软件清 0, 表示是否发生模拟看门狗事件。 0: 没有发生 1: 发生
8	SEQ_NUM3_FIN	RC_W1	序列分段采样第三段结束 (NUM.3 of Sequential Section Sampling Finish) 0: 没有发生 1: 发生
9	SEQ_NUM1_FIN	RC_W1	序列分段采样第一段结束 (NUM.1 of Sequential Section Sampling Finish) 0: 没有发生 1: 发生
10	SEQ_NUM2_FIN	RC_W1	序列分段采样第二段结束 (NUM.2 of Sequential Section Sampling Finish) 0: 没有发生 1: 发生
31:11	保留		

24.5.2 ADC 中断使能寄存器 (ADC_IEN)

偏移地址: 0x04

复位值: 0x0000 0000

位/域	名称	R/W	描述
0	ADCRDYIEN	R/W	使能 ADC 准备好中断 (ADC Ready Interrupt Enable) 0: 禁止 1: 使能
1	EOSMPIEN	R/W	使能采样结束中断 (End of Sampling Flag Interrupt Enable) 0: 禁止 1: 使能
2	EOCIEN	R/W	使能转换结束中断 (End of Conversion Interrupt Enable) 0: 禁止 1: 使能

位/域	名称	R/W	描述
3	EOSEQIEN	R/W	使能序列转换结束中断 (End of Conversion Sequence Interrupt Enable) 0: 禁止 1: 使能
4	OVRIEN	R/W	使能过载中断 (Overrun Interrupt Enable) 0: 禁止 1: 使能
6:5	保留		
7	AWDIEN	R/W	使能模拟看门狗中断 (Analog Watchdog Interrupt Enable) 0: 禁止 1: 使能
8	SEQ_NUM3_FINIEN	R/W	序列分段采样第三段结束中断 (NUM.3 of Sequential Section Sampling Finish Interrupt Enable) 0: 禁止 1: 使能
9	SEQ_NUM1_FINIEN	R/W	序列分段采样第一段结束中断 (NUM.1 of Sequential Section Sampling Finish Interrupt Enable) 0: 禁止 1: 使能
10	SEQ_NUM2_FINIEN	R/W	序列分段采样第二段结束中断 (NUM.2 of Sequential Section Sampling Finish Interrupt Enable) 0: 禁止 1: 使能
31:11	保留		

注：只有当 STARTCEN=0 时才允许改写这些位。

24.5.3 ADC 控制寄存器 (ADC_CTRL)

偏移地址：0x08

复位值：0x0000 0000

位/域	名称	R/W	描述
0	ADCEN	R/S	使能 ADC (ADC Enable) 该位由软件置 1，硬件清 0。 0: 禁用 ADC 1: 使能 ADC 注：只有在 ADC_CTRL 寄存器所有位为 0 的情况下，软件才能设置 ADCEN 位。
1	ADCD	R/S	禁止 ADC (ADC Disable) 该位由软件置 1，硬件清 0。 0: 无效 1: 禁止 ADC，进入掉电模式 注：只有当 ADCEN=1 和 STARTCEN=0 时才允许软件设置 ADCD 位。

位/域	名称	R/W	描述
2	STARTCEN	R/S	使能 ADC 开始转换 (ADC Start Conversion Enable) 该位由软件置 1, 硬件清 0。 0: 禁止 ADC 转换 1: 开始 ADC 转换 注: 只有当 ADCEN=1 和 ADCD=0 时才允许软件设置 STARTCEN 位。
3	保留		
4	STOPCEN	R/S	使能 ADC 停止转换 (ADC Stop Conversion Enable) 该位由软件置 1, 硬件清 0。 0: 无效 1: 停止 ADC 转换 注: 只有当 STARTCEN=1 和 ADCD=0 时软件才能对该位进行设置。
30:5	保留		
31	CAL	R/S	ADC 校准 (ADC Calibrate) 该位软件置 1, 硬件清 0。 0: 校准完成 1: 开始校准 注: 只有在 ADC 禁止的情况下才允许软件设置 CAL 位。

24.5.4 ADC 配置寄存器 1 (ADC_CFG1)

偏移地址: 0x0C

复位值: 0x0000 0000

位/域	名称	R/W	描述
0	DMAEN	R/W	使能 DMA (DMA Enable) 0: 禁止 DMA 1: 使能 DMA
1	DMACFG	R/W	配置 DMA 模式 (DMA Mode Configure) 该位仅在 DMAEN=1 时生效。 0: DMA 单次模式 1: DMA 循环模式
2	SCANSEQDIR	R/W	配置扫描序列方向 (Scan Sequence Direction Configure) 0: 向前扫描(从 CHSEL0 到 CHSEL16) 1: 向后扫描(从 CHSEL16 到 CHSEL0)
4:3	DATARESCFG	R/W	配置数据分辨率 (Data Resolution Configure) 00: 12 位 01: 10 位 10: 8 位 11: 6 位
5	DALIGCFG	R/W	配置数据对齐方式 (Data Alignment Configure) 0: 右对齐 1: 左对齐
9:6	EXTTRGSEL1	R/W	选择外部触发事件 1 (External Trigger Event Select) 这些位用于选择触发 ADC 转换的外部事件。 0000: 事件 0 0001: 事件 1

位/域	名称	R/W	描述
			0010: 事件 2 0011: 事件 3 0100: 事件 4 0101: 事件 5 0110: 事件 6 0111: 事件 7 1000: 事件 8 1001: 事件 9 1010: 事件 10 1011: 事件 11 1100: 事件 12 1101: 事件 13: 1110: 事件 14 1111: 事件 15
11:10	EXTPOLSEL1	R/W	使能外部触发和极性选择 1 (External Trigger Enable and Polarity Select) 00: 硬件触发检测关闭(可由软件启动转换) (分段采样功能中, 该位不允许为 00) 01: 在上升沿进行硬件触发检测 10: 在下降沿进行硬件触发检测 11: 在上升和下降沿进行硬件触发检测
12	OVRMAG	R/W	过载管理模式 (Overrun Management Mode) 0: 当检测到过载事件时, ADC_DATA 寄存器保存以前的数据 1: 当检测到过载事件时, ADC_DATA 寄存器保存最后一次转换的数据
13	CMODESEL	R/W	选择单次/连续转换模式 (Select Single/Continuous Conversion Mode) 0: 单次转换模式 1: 连续转换模式
14	WAITCEN	R/W	使能自动延迟转换模式 (Wait Conversion Mode Enable) 0: 禁止 1: 开启
15	AOEN	R/W	使能自动关断模式 (Auto-Off Mode Enable) 0: 禁止 1: 开启
16	DISCEN	R/W	使能断续模式 (Discontinuous Mode Enable) 0: 禁止 1: 开启
21:17	保留		
22	AWDCHEN	R/W	在单一通道或所有通道使能看门狗 (Enable The Watchdog On A Single Channel or on All Channels) 0: 在所有通道上使能模拟看门狗 1: 在单一通道上使能模拟看门狗
23	AWDEN	R/W	使能模拟看门狗 (Analog Watchdog Enable)、 0: 禁止 1: 开启

位/域	名称	R/W	描述
25:24	保留		
30:26	AWDCHSEL	R/W	<p>选择模拟看门狗通道 (Analog Watchdog Channel Selecte) 这些位用来配置模拟看门狗监视 ADC 的输入通道。 00000: 通道 0 00001: 通道 1 10010: 通道 18 其它值: 保留, 不会被使用 注: 被 AWDCHSEL 位所选择的通道必须同样写入 CHSELR 寄存器。 序列分段采样模式, 用来检测那一个数据寄存器存储的数据 00000: ADC_DATA0 里的数据 00001: ADC_DATA1 里的数据 10010: ADC_DATA18 里的数据 其他: 保留</p>
31	保留		

注: 只有当 STARTCEN=0 时(确定无进行中的转换)才允许改写这些位。

24.5.5 ADC 配置寄存器 2 (ADC_CFG2)

偏移地址: 0x10

复位值: 0x0000 0000

位/域	名称	R/W	描述
0	SEQEN	R/E	<p>使能序列分段采样 (Sequential Section Sampling Enable) 0: 禁止 1: 使能</p>
7:1	保留		
22:8	TGAP	R/W	<p>配置序列段中转换的时间间隔 (Time of The Gap between Conversions in Sequential Section Configure) 此位配置序列中每一段中上一次转换完成到下一次开始采样的时间间隔 (以 PCLK 为单位)。 0x0: 无间隔 0x1: 1 个 PCLK 时钟周期间隔 0x2: 2 个 PCLK 时钟周期间隔 0x7FFF: 32767 个 PCLK 时钟周期间隔</p>
25:23	保留		
26	CLKSEL	R/W	<p>选择 ADC 的转换与模拟部分的时钟 (ADC Conversion and Analog Clock Select) 当 CLKCFG=00 时: 0: 使用 ADCCLK (异步时钟) 1: 使用 PCLK 或 PCLK 的分频时钟, 分频系数参考[29:27]位</p>

位/域	名称	R/W	描述
29:27	PCLKDIV	R/W	配置 PCLK 的分频系数 (PCLK Division Factor Configure) 当 CLKSEL=1 时, 此位配置 PCLK 将会经过多少分频再作为 ADC 模拟部分的时钟。 000: 不分频 001: 2 分频 010: 3 分频 011: 4 分频 100: 5 分频 101: 6 分频 110: 7 分频 111: 8 分频
31:30	CLKCFG	R/W	配置 ADC 时钟模式 (ADC Clock Mode Configure) 00: ADCCLK(异步时钟模式) 或 PCLK 或 PCLK 的分频时钟 01: PCLK/2(同步时钟模式) 10: PCLK/4(同步时钟模式) 11: 保留 注:软件只允许在 ADC 被禁用时写入这些位。

注: 只有当 ADC 被禁止时 (STARTCEN=0, DISCEN=0, CAL=0, STOPCEN=0, ADCEN=0) 才允许改写该寄存器。

24.5.6 ADC 采样时间寄存器 (ADC_SMPTIM)

偏移地址: 0x14

复位值: 0x0000 0000

位/域	名称	R/W	描述
2:0	SMPCYCSEL	R/W	选择采样时间 (Sampling Cycles Selecte) 000: 1.5ADC 时钟周期 001: 7.5ADC 时钟周期 010: 13.5ADC 时钟周期 011: 28.5ADC 时钟周期 100: 41.5ADC 时钟周期 101: 55.5ADC 时钟周期 110: 71.5ADC 时钟周期 111: 239.5ADC 时钟周期 注: 只有当 STARTCEN=0 时才允许改写这些位。
31:3	保留		

24.5.7 ADC 配置寄存器 3 (ADC_CFG3)

偏移地址: 0x18

复位值: 0x0000 0000

如果不使用序列分段采样功能, 则无需配置此寄存器。

位/域	名称	R/W	描述
1:0	EXTPOLSEL2	R/W	使能外部触发和极性选择 2 (External Trigger Enable and Polarity Select) 00: 无效 (分段采样功能中, 该位不允许为 00) 01: 在上升沿进行硬件触发检测 10: 在下降沿进行硬件触发检测 11: 在上升和下降沿进行硬件触发检测
5:2	EXTTRGSEL2	R/W	选择外部触发事件 2 (External Trigger Event Select) 这些位用于选择触发 ADC 转换的外部事件。 0000: 事件 0 0001: 事件 1 0010: 事件 2 0011: 事件 3 0100: 事件 4 0101: 事件 5 0110: 事件 6 0111: 事件 7 1000: 事件 8 1001: 事件 9 1010: 事件 10 1011: 事件 11 1100: 事件 12 1101: 事件 13: 1110: 事件 14 1111: 事件 15
7:6	保留		
9:8	EXTPOLSEL3	R/W	使能外部触发和极性选择 3 (External Trigger Enable and Polarity Select) 00: 无效 (分段采样功能中, 此位不能为 00) 01: 在上升沿进行硬件触发检测 10: 在下降沿进行硬件触发检测 11: 在上升和下降沿进行硬件触发检测

位/域	名称	R/W	描述
13:10	EXTTRGSEL3	R/W	选择外部触发事件 3 (External Trigger Event Select) 这些位用于选择触发 ADC 转换的外部事件。 0000: 事件 0 0001: 事件 1 0010: 事件 2 0011: 事件 3 0100: 事件 4 0101: 事件 5 0110: 事件 6 0111: 事件 7 1000: 事件 8 1001: 事件 9 1010: 事件 10 1011: 事件 11 1100: 事件 12 1101: 事件 13: 1110: 事件 14 1111: 事件 15
31:14	保留		

注: 只有当 STARTCEN=0 时(确定无进行中的转换)才允许改写这些位。

24.5.8 ADC 看门狗阈值寄存器(ADC_AWDT)

偏移地址: 0x20

复位值: 0x0FFF 0000

位/域	名称	R/W	描述
11:0	AWDLT[11:0]	R/W	模拟看门狗的低阈值 (Analog Watchdog Low Threshold)
15:12	保留		
27:16	AWDHT[11:0]	R/W	模拟看门狗的高阈值 (Analog Watchdog High Threshold)
31:28	保留		

注: 只有当 STARTCEN=0 时才允许改写这些位。

24.5.9 ADC 通道选择寄存器(ADC_CHSEL)

偏移地址: 0x28

复位值: 0x0000 0000

位/域	名称	R/W	描述
17:0	CHxSEL	R/W	选择通道 (Channel-x Select) 0: 输入通道 x 不被选为转换通道 1: 输入通道 x 被选为转换通道
31:18	保留		

注: 只有当 STARTCEN=0 时才允许改写这些位。

24.5.10 ADC 数据寄存器(ADC_DATA)

偏移地址: 0x40

复位值: 0x0000 0000

位/域	名称	R/W	描述
15:0	CDATA[15:0]	R	转换数据 (Converted Data) 这些位只读。其包含最后转换通道的转换结果值。 仅在校准完成时, CDATA[6:0]值为校准因子。
31:16	保留		

24.5.11 ADC 序列分段采样数据寄存器 (ADC_DATAx) (x=0...18)

偏移地址: 0x44+4x (x=0...18)

复位值: 0x0000 0000

位/域	名称	R/W	描述
15:0	ADC_DATAx	R/W	序列分段采样结果数据 x (x=0...18) (Sequential Section Sampling Data) 序列分段采样的结果将依照固有规则放入该 19 个数据寄存器中。例如, 配置 3 段序列采样, 第一段通道总数为 1, 第二段通道总数为 2, 第三段通道总数为 3 (配置 ADC_SEQ_NUM = 0x20000820) 则第一段的数据结果将出现在 ADC_DATA0, 第二段的数据结果将出现在 ADC_DATA1-2, 第二段的数据结果将出现在 ADC_DATA3-5, 虽然 ADC 分段采样模式支持乱序 (即第二、三段采样可先于第一段), 但每段对应的结果寄存器固定。
31:16	保留		

24.5.12 ADC 序列分段采样控制寄存器 (ADC_SEQ_NUM)

偏移地址: 0x100

复位值: 0x0000 0000

位/域	名称	R/W	描述
4:0	SEQ_NUM1	R/W	设置序列分段采样第一段传输次数 (Section1 Transmission Time Setup) 00000: 传输 1 次 00001: 传输 2 次 10000: 传输 17 次 10001: 传输 18 次 10010: 传输 19 次 其他: 无效
9:5	SEQ_NUM2	R/W	设置序列分段采样第二段传输次数 (Section2 Transmission Time Setup) 00000: 传输 1 次 00001: 传输 2 次 10000: 传输 17 次 10001: 传输 18 次 其他: 无效

位/域	名称	R/W	描述
14:10	SEQ_NUM3	R/W	设置序列分段采样第三段传输次数 (Section3 Transmission Time Setup) 00000: 传输 1 次 00001: 传输 2 次 10000: 传输 17 次 其他: 无效
27:15	保留		
29:28	SG_NUM	R/W	序列的段数 (The Number of Sequential Section) 00: 1 段 01: 2 段 10: 3 段 11: 无效
31:30	保留		

24.5.13 ADC1-6 次传输的通道寄存器 (ADC_CHANNEL1)

偏移地址: 0x104

复位值: 0x0000 0000

位/域	名称	R/W	描述
4:0	CH0	R/W	第一次转换的通道数 (Number of Channels Converted for The First Time)
9:5	CH1	R/W	第二次转换的通道数 (Number of Channels Converted for The Second Time)
14:10	CH2	R/W	第三次转换的通道数 (Number of Channels Converted for The Third Time)
19:15	CH3	R/W	第四次转换的通道数 (Number of Channels Converted for The Forth Time)
24:20	CH4	R/W	第五次转换的通道数 (Number of Channels Converted for The Fifth Time)
29:25	CH5	R/W	第六次转换的通道数 (Number of Channels Converted for The Sixth Time)
31:30	保留		

24.5.14 ADC7-12 次传输的通道寄存器 (ADC_CHANNEL2)

偏移地址: 0x108

复位值: 0x0000 0000

位/域	名称	R/W	描述
4:0	CH6	R/W	第 7 次转换的通道数 (Number of Channels Converted for The 7th Time)
9:5	CH7	R/W	第 8 次转换的通道数 (Number of Channels Converted for The 8th Time)
14:10	CH8	R/W	第 9 次转换的通道数 (Number of Channels Converted for The 9th Time)
19:15	CH9	R/W	第 10 次转换的通道数 (Number of Channels Converted for The 10th Time)

位/域	名称	R/W	描述
24:20	CH10	R/W	第 11 次转换的通道数 (Number of Channels Converted for The 11th Time)
29:25	CH11	R/W	第 12 次转换的通道数 (Number of Channels Converted for The 12th Time)
31:30	保留		

24.5.15 ADC13-18 次传输的通道寄存器 (ADC_CHANNEL3)

偏移地址: 0x10C

复位值: 0x0000 0000

位/域	名称	R/W	描述
4:0	CH12	R/W	第 13 次转换的通道数 (Number of Channels Converted for The 13th Time)
9:5	CH13	R/W	第 14 次转换的通道数 (Number of Channels Converted for The 14th Time)
14:10	CH14	R/W	第 15 次转换的通道数 (Number of Channels Converted for The 15th Time)
19:15	CH15	R/W	第 16 次转换的通道数 (Number of Channels Converted for The 16th Time)
24:20	CH16	R/W	第 17 次转换的通道数 (Number of Channels Converted for The 17th Time)
29:25	CH17	R/W	第 18 次转换的通道数 (Number of Channels Converted for The 18th Time)
31:30	保留		

24.5.16 ADC19 次传输的通道寄存器 (ADC_CHANNEL4)

偏移地址: 0x110

复位值: 0x0000 0000

位/域	名称	R/W	描述
4:0	CH18	R/W	第 19 次转换的通道数 (Number of Channels Converted for The 19th Time)
31:5	保留		

24.5.17 ADC 通用配置寄存器(ADC_CCFG)

偏移地址: 0x308

复位值: 0x0000 0000

位/域	名称	R/W	描述
21:0	保留		
22	VREFEN	R/W	使能 VREFINT (VREFINT Enable) 0: 禁止 1: 使能
23	TSEN	R/W	使能温度传感 (Temperature Sensor Enable) 0: 禁止 1: 使能

位/域	名称	R/W	描述
24	HLAF_VDDEN	R/W	使能 1/2*VDD (1/2*VDD Enable) 0: 禁止 1: 使能
31:25	保留		

注：只有当 STARTCEN=0 时才允许改写这些位

25 比较器 (COMP)

25.1 术语全称、缩写描述

表格 99 术语全称、缩写描述

中文全称	英文全称	英文缩写
比较器	Comparator	COMP
反转	Invert	INV
迟滞	Hysteresis	HYS
正输入	Input Plus	INP
负输入	Input Minus	INM

25.2 简介

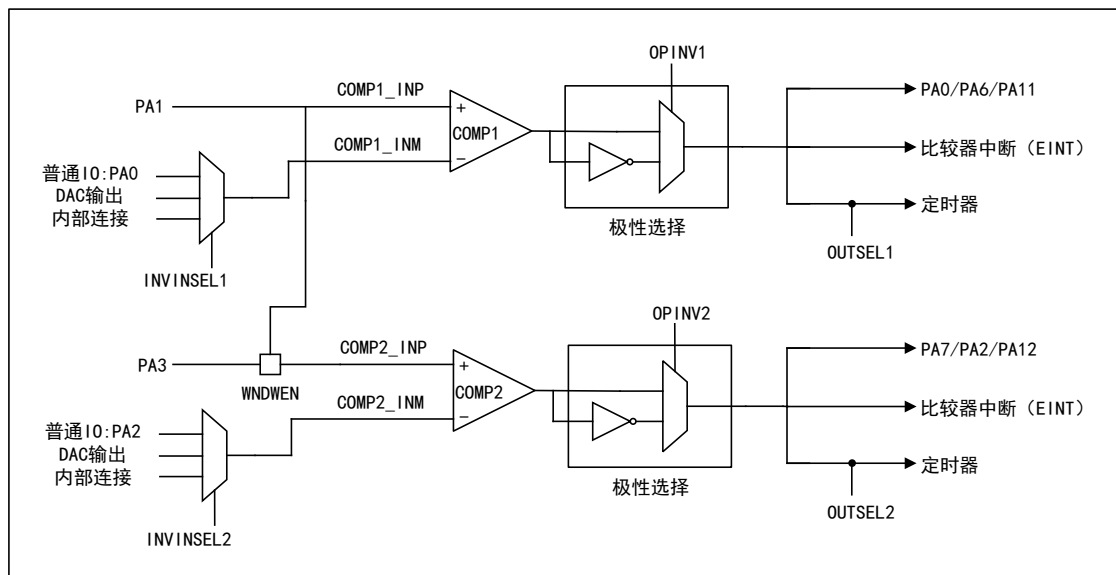
MCU 内嵌两个可独立使用的通用比较器 (COMP1 和 COMP2)，它们可以和定时器结合使用。

25.3 主要特征

- 轨对轨输入
- 两个比较器可以组合成一个窗口比较器使用
- 迟滞、速率和损耗都可编程
- 能够产生中断
- 能从睡眠模式和停止模式唤醒 (通过 EINT)
- 能在停止模式下工作

25.4 结构框图

图 109 COMP 结构框图



25.5 功能描述

25.5.1 COMP 时钟

COMP 没有单独的时钟使能控制位，工作时独立于 PCLK 时钟，但它的时钟与 PCLK 同步。

COMP 只能通过系统复位来达到模块的复位。

25.5.2 COMP 输入

GPIO 作为比较器输入时需要配置为模拟模式。

COMP 输入分为同相输入与反相输入。同相输入端都连接到外部 IO；反相输入端可编程进行选择，外部连接有 IO 引脚、DAC 输出引脚；内部连接有内部参考电压 (V_{REFINT})、内部参考电压的 1/4 或 1/2 或 3/4。

25.5.3 COMP 输出

比较器的输出可以连接到外部 IO 口；

同样也可以连接到内部定时器以下信号：

- PWM 的刹车信号
- OCREF_CLR 的输入信号
- 定时器的输入捕获通道

输出极性可通过编程 COMP_CSTS 寄存器中 OPINVx 位来进行修改

25.5.4 COMP 模式

比较器的速率和损耗都是可编程的。综合实际应用的场合，我们可以通过编程寄

寄存器中 MODx 位来达到最合适的状态。

比较器拥有可编程迟滞功能，选择适当的迟滞时间可以避免由于噪声而产生的无效输出。

窗口比较器模式

COMP1 和 COMP2 的同相输入连接，同时断开与 PA3 的连接。

25.5.5 COMP 中断

比较器输出内部连接到了外部中断和事件控制器，正确配置外部中断便可通过 DAC 输出产生中断或者唤醒进入 sleep、stop 模式的 MCU。

25.6 寄存器地址映射

表格 100 寄存器地址映射

寄存器名	描述	偏移地址
COMP_CSTS	COMP 控制状态寄存器	0x1C

25.7 寄存器功能描述

25.7.1 COMP 控制状态寄存器 (COMP_CSTS)

地址偏移: 0x1C

复位值: 0x0000 0000

COMP 的控制位可以有两种 R/W 状态，RW/R 指可读可写或只读状态。

位/域	名称	R/W	描述
0	EN1	RW/R	使能比较器 1 (COMP1 Enable) 0: 禁止 1: 使能
1	SW1	RW/R	比较器 COMP1 同相输入 DAC 开关 (COMP1 Non Inverting Input on PA1 And PA4 Switch) 当该位被设置时,关闭 PA1 上通用比较器 COMP1 的同相输入端和 PA4 (DAC) 的 I/O 之间的开关。 0: 断开 1: 闭合
3:2	MOD1	RW/R	比较器 1 模式 (COMP1 Mode) 控制通用比较器 COMP1 的工作模式, 即工作速率和损耗。 00: 高速率/全功率 01: 中等速率/中等功率 10: 低速率/低功率 11: 极低速率/极低功率
6:4	INVINSEL1	RW/R	选择比较器 COMP1 反相输入 (COMP1 Inverting Input Select) 选择连接到比较器 COMP1 的反相输入信号源。 000: V _{REFINT} 的 1/4 001: V _{REFINT} 的 1/2 010: V _{REFINT} 的 3/4

位/域	名称	R/W	描述
			011: V _{REFINT} (内部参考电压) 100: COMP1_INM4 (PA4, 即 DAC_OUT1) 101: COMP1_INM5 (PA5) 110: COMP1_INM6 (PA0) 111 保留。
7	保留		
10:8	OUTSEL1	RW/R	选择比较器 COMP1 输出 (COMP1 Output Select) 这些位用来选择比较器 COMP1 的输出方向。 000: 无选择 001: TMR1 中断输入 010: TMR1 输入捕捉 1 011: TMR 1OCrefclear 输入 100: TMR 2 输入捕捉 4 101: TMR 2OCrefclear 输入 110: TMR 3 输入捕捉 1 111: TMR 3OCrefclear 输入
11	OPINV1	RW/R	反转比较器 1 输出极性 (COMP1 Output Polarity Invert) 反转比较器 COMP1 的输出极性 0: 同相输出 1: 反相输出。
13:12	HYSCFG1	RW/R	配置比较器 1 迟滞等级 (COMP1 Hysteresis Level Configure) 配置 COMP1 的迟滞等级 00: 无迟滞 01: 低程度迟滞 10: 中程度迟滞 11: 高程度迟滞
14	OUTSTS1	R	比较器 COMP1 输出状态 (COMP1 Output State) 0: 低输出 同相输出情况下: 同相输入低于反相输入, 输出低电平 反相输出情况下: 同相输入高于反相输入, 输出低电平 1: 高输出 同相输出情况下: 同相输入高于反相输入, 输出高电平 反相输出情况下: 同相输入低于反相输入, 输出高电平
15	LOCK1	R/S	锁定比较器 1 (COMP1 Lock) 该位只写一次, 由软件置位, 只能通过系统重置来清除。 锁定时, COMP1 的所有控制位变为只读。 0: COMP1 控制位可读可写 1: COMP1 控制位只读
16	EN2	RW/R	使能比较器 COMP2 (COMP2 Enable) 0: 禁止 1: 使能
17	保留		
19:18	MOD2	RW/R	比较器 COMP2 模式 (COMP2 Mode) 控制通用比较器 COMP2 的工作模式, 即工作速率和损耗。 00: 高速率/全功率

位/域	名称	R/W	描述
			01: 中等速率/中等功率 10: 低速率/低功率 11: 极低速率/极低功率
22:20	INVINSEL2	RW/R	选择比较器 COMP2 反相输入 (COMP2 Inverting Input Select) 选择连接到比较器 COMP2 的反相输入信号源。 000: V _{REFINT} 的 1/4 001: V _{REFINT} 的 1/2 010: V _{REFINT} 的 3/4 011: V _{REFINT} (内部参考电压) 100: COMP2_INM4 (PA4, 即 DAC_OUT1) 101: COMP2_INM5 (PA5) 110: COMP2_INM6 (PA0) 111: 保留
23	WMODEN	RW/R	使能窗口模式 (Window Mode Enable) 两个通用比较器能够连接成为窗口比较器模式。 0: 禁止 1: 使能
26:24	OUTSEL2	RW/R	选择比较器 COMP2 输出 (COMP2 Output Select) 选择比较器 COMP2 的输出方向。 000: 无选择 001: 定时器 1 中断输入 010: 定时器 1 输入捕捉 1 011: 定时器 1OCrefclear 输入 100: 定时器 2 输入捕捉 4 101: 定时器 2OCrefclear 输入 110: 定时器 3 输入捕捉 1 111: 定时器 3OCrefclear 输入
27	OPINV2	RW/R	反转比较器 COMP2 输出极性 (COMP2 Output Polarity Invert) 反转比较器 COMP2 的输出极性 0: 同相输出 1: 反相输出
29:28	HYSCFG2	RW/R	配置比较器 COMP2 迟滞等级 (COMP2 Hysteresis Level Configure) 00: 无迟滞 01: 低程度迟滞 10: 中程度迟滞 11: 高程度迟滞
30	OUTSTS2	R	比较器 COMP2 输出状态 (COMP2 Output State) 0: 低输出 同相输出情况下: 同相输入低于反相输入, 输出低电平 反相输出情况下: 同相输入高于反相输入, 输出低电平 1: 高输出 同相输出情况下: 同相输入高于反相输入, 输出高电平 反相输出情况下: 同相输入低于反相输入, 输出高电平
31	LOCK2	R/S	锁定比较器 COMP2 (COMP2 Lock) 该位只写一次, 由软件置位, 只能通过系统重置来清除。 锁定时, COMP2 的所有控制位变为只读。

位/域	名称	R/W	描述
			0: COMP2 控制位可读可写 1: COMP2 控制位只读

26 运算放大器（OPA）

26.1 简介

MCU 内嵌四个可独立使用的运算放大器（OPA1、OPA2、OPA3 和 OPA4），它们可以和 COMP、ADC 结合使用。

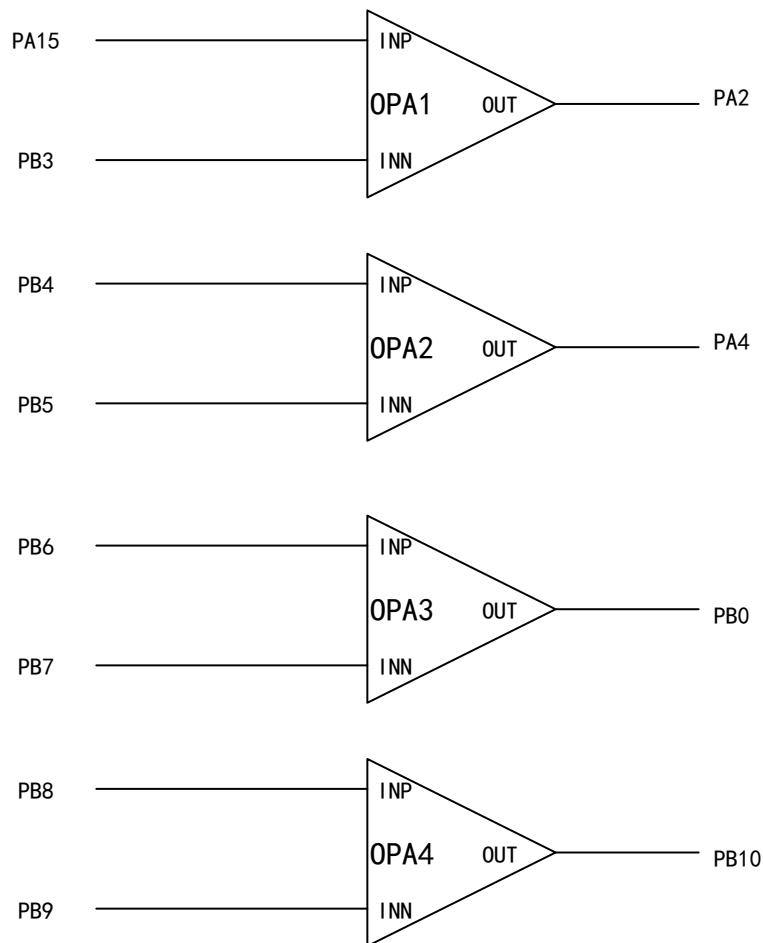
26.2 主要特征

- (4) 运算放大器的输入输出端都开放，可通过外界电阻调整运放增益，也可以通过设置选择内部预设的增益档位。
- (5) 放大器输出可作为 ADC 转换模块的输入。

26.3 结构框图

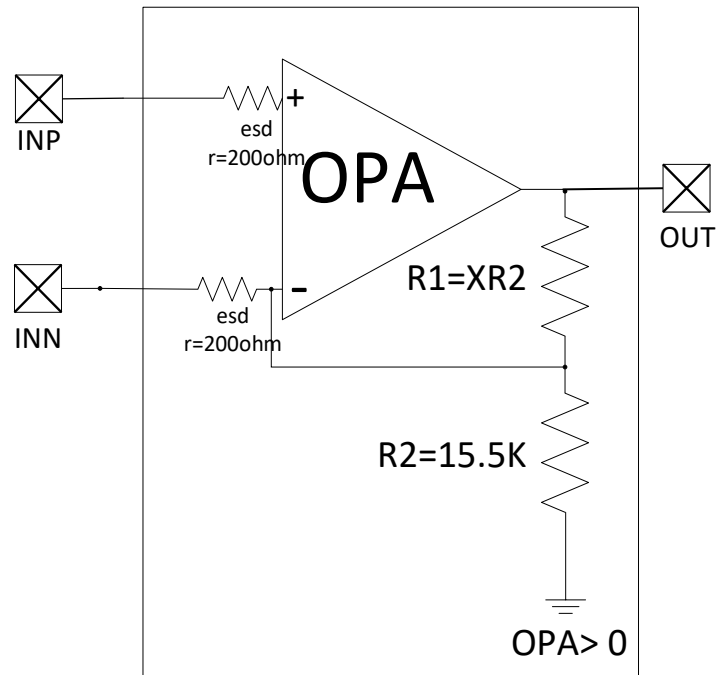
26.3.1 结构框图

图 110 OPA 结构框图



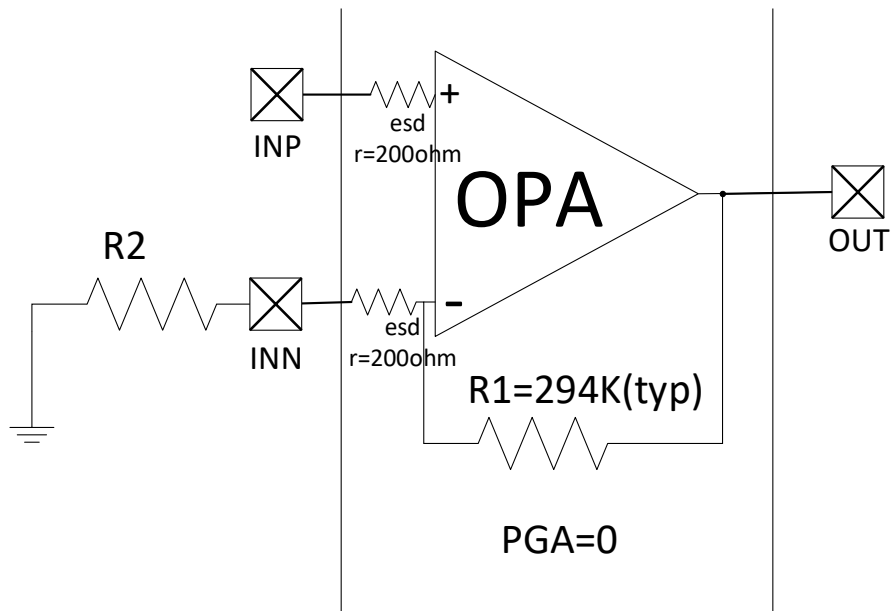
26.3.2 内部增益应用示意图

图 111 内部增益应用示意图



26.3.3 外部增益应用示意图

图 112 外部增益应用示意图



26.4 寄存器地址映射

表格 101 OPA 寄存器地址映射

寄存器名	描述	偏移地址
OPA_CFG	OPA 配置寄存器	0x34

26.5 寄存器功能描述

26.5.1 运放配置寄存器 (OPA_CFG)

偏移地址: 0x34

复位值: 0x0000 0000

位/域	名称	R/W	描述
0	OPA1EN	R/W	使能 OPA1 (OPA1 Enable) 0: 禁止 1: 使能
1	保留		
4:2	OPA1G	R/W	OPA1 的增益系数 (OPA1 Gain Factor) 000: 外部电阻网络确定, OP1N 为运放反相输入脚, OP1OUT 为运放输出脚 001: 由内部增益系数确定 X1, OP1N/OP1OUT 为普通 IO 010: 由内部增益系数确定 X2, OP1N/OP1OUT 为普通 IO 011: 由内部增益系数确定 X4, OP1N/OP1OUT 为普通 IO 100: 由内部增益系数确定 X8, OP1N/OP1OUT 为普通 IO 101: 由内部增益系数确定 X10, OP1N/OP1OUT 为普通 IO 110: 由内部增益系数确定 X16, OP1N/OP1OUT 为普通 IO 111: 由内部增益系数确定 X20, OP1N/OP1OUT 为普通 IO
7:5	保留		
8	OPA2EN	R/W	使能 OPA2 (OPA2 Enable) 0: 禁止 1: 使能
9	保留		
12:10	OPA2G	R/W	OPA2 的增益系数 (OPA2 Gain Factor) 000: 外部电阻网络确定, OP2N 为运放反相输入脚, OP2OUT 为运放输出脚 001: 由内部增益系数确定 X1, OP2N/OP2OUT 为普通 IO 010: 由内部增益系数确定 X2, OP2N/OP2OUT 为普通 IO 011: 由内部增益系数确定 X4, OP2N/OP2OUT 为普通 IO 100: 由内部增益系数确定 X8, OP2N/OP2OUT 为普通 IO 101: 由内部增益系数确定 X10, OP2N/OP2OUT 为普通 IO 110: 由内部增益系数确定 X16, OP2N/OP2OUT 为普通 IO 111: 由内部增益系数确定 X20, OP2N/OP2OUT 为普通 IO
15:13	保留		

位/域	名称	R/W	描述
16	OPA3EN	R/W	使能 OPA3 (OPA3 Enable) 0: 禁止 1: 使能
17	保留		
20:18	OPA3G	R/W	OPA3 的增益系数 (OPA3 Gain Factor) 000: 外部电阻网络确定, OP3N 为运放反相输入脚, OP3OUT 为运放输出脚 001: 由内部增益系数确定 X1, OP3N/OP3OUT 为普通 IO 010: 由内部增益系数确定 X2, OP3N/OP3OUT 为普通 IO 011: 由内部增益系数确定 X4, OP3N/OP3OUT 为普通 IO 100: 由内部增益系数确定 X8, OP3N/OP3OUT 为普通 IO 101: 由内部增益系数确定 X10, OP3N/OP3OUT 为普通 IO 110: 由内部增益系数确定 X16, OP3N/OP3OUT 为普通 IO 111: 由内部增益系数确定 X20, OP3N/OP3OUT 为普通 IO
23:21	保留		
24	OPA4EN	R/W	使能 OPA4 (OPA4 Enable) 0: 禁止 1: 使能
25	保留		
28:26	OPA4G	R/W	OPA4 的增益系数 (OPA4 Gain Factor) 000: 外部电阻网络确定, OP4N 为运放反相输入脚, OP4OUT 为运放输出脚 001: 由内部增益系数确定 X1, OP4N/OP4OUT 为普通 IO 010: 由内部增益系数确定 X2, OP4N/OP4OUT 为普通 IO 011: 由内部增益系数确定 X4, OP4N/OP4OUT 为普通 IO 100: 由内部增益系数确定 X8, OP4N/OP4OUT 为普通 IO 101: 由内部增益系数确定 X10, OP4N/OP4OUT 为普通 IO 110: 由内部增益系数确定 X16, OP4N/OP4OUT 为普通 IO 111: 由内部增益系数确定 X20, OP4N/OP4OUT 为普通 IO
31:29	保留		

27 循环冗余校验计算单元 (CRC)

27.1 简介

循环冗余校验 (CRC) 计算单元可将输入数据经过固定的生成多项式计算得到 8/16/32 位的 CRC 计算结果, 主要用来检测或校验数据传输或者保存后的正确性与完整性。

27.2 功能描述

27.2.1 计算方式

使用 CRC-32 (以太网) 多项式: 0x4C11DB7

$$(X^{32}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^8+X^7+X^5+X^4+X^2+X+1)$$

27.2.2 计算时间

- 处理 32 位数据时, 其计算时间为 4 个 AHB 时钟周期
- 处理 16 位数据时, 其计算时间为 2 个 AHB 时钟周期
- 处理 8 位数据时, 其计算时间为 1 个 AHB 时钟周期

27.2.3 功能特性

- 处理 8 位、16 位、32 位数据
- 可以使用位数可编程 (7 位、8 位、16 位或 32 位) 的完全可编程多项式
- 可编程 CRC 初始值
- 独立 32 位输入/输出寄存器
- 可被用于临时存储的通用 8 位寄存器
- I/O 数据的可逆性选项
- 数据宽度可动态调整以减少计算写入次数
- 输入数据高低位可以颠倒, 以适应不同数据存放方式 (字节、半字或字, 大小端体系)
- 可以进行字计算或字节计算, 这由写入的不同数据格式决定
- 拥有输入缓冲区以减少等待周期和避免总线阻塞

CRC 单元含有一个 32 位读/写寄存器 CRC_DATA, 用来写入新数据和给出 CRC 计算结果, 每写入一次新数据, 其结果是上一次的计算结果和新的计算结果的组合。(对整个字进行运算)。对 CRC_DATA 能够进行字或右对齐半字或右对齐字节的存取操作, 而其他寄存器只能进行 32 位的存取。

可编程多项式

在应用程序复位后或者对 CRC_DATA 读操作后, 才可以通过设置 CRC_CTRL 寄存器中 POLSEL 位来选择或更改多项式的大小, 即在 CRC 计算时, 不能更改多项式的值或大小。

27.3 寄存器地址映射

表格 102 CRC 寄存器地址映射

寄存器名	描述	偏移地址
CRC_DATA	数据寄存器	0x00
CRC_INDATA	独立数据寄存器	0x04
CRC_CTRL	控制寄存器	0x08
CRC_INITVAL	CRC 初始值寄存器	0x10
CRC_POL	CRC 多项式寄存器	0x14

27.4 寄存器功能描述

27.4.1 数据寄存器 (CRC_DATA)

偏移地址: 0x00

复位值: 0xFFFF FFFF

位/域	名称	R/W	描述
31:0	DATA	R/W	32 位数据 (32bit Data) 作为输入寄存器: 写入时存储 CRC 计算器的新数据。 作为输出寄存器: 读取时返回 CRC 计算的结果。

27.4.2 独立数据寄存器 (CRC_INDATA)

偏移地址: 0x04

复位值: 0x0000 0000

位/域	名称	R/W	描述
7:0	INDATA	R/W	独立 8 位数据 (Independent 8bit Data) 可用于临时存放 1 字节的数据。 寄存器 CRC_CTRL 的 RST 位产生的 CRC 复位对本寄存器没有影响。
31:8			保留。

注: 此寄存器不参与 CRC 计算, 可以存放任何数据。

27.4.3 控制寄存器 (CRC_CTRL)

偏移地址: 0x08

复位值: 0x0000 0000

位/域	名称	R/W	描述
0	RST	R/S	复位 CRC 计算单元 (Reset CRC Calculation Unit) 设置数据寄存器为 0xFFFF FFFF。只能对该位置位, 它由硬件自动清 0。
2:1			保留

位/域	名称	R/W	描述
4:3	POLSEL	R/W	选择可编程多项式大小 (Programmable Polynomial Size Select) 00: 32 位 01: 16 位 10: 8 位 11: 7 位
6:5	REVI	R/W	翻转输入数据 (Input Data Reverse) 对输入数据以不同单位进行翻转。 00: 不翻转 01: 以字节为单位 10: 以半字为单位 11: 以字为单位
7	REVO	R/W	翻转输出数据 (Output Data Reverse) 0: 不翻转 1: 翻转
31:8	保留		

27.4.4 CRC 初始值寄存器 (CRC_INITVAL)

偏移地址: 0x10

复位值: 0xFFFF FFFF

位/域	名称	R/W	描述
31:0	VALUE	R/W	CRC 的初始值 (Initial CRC Value) CRC 的初始值是可编程的, 此位用于设置 CRC 的初值。

27.4.5 CRC 多项式寄存器 (CRC_POL)

偏移地址: 0x14

复位值: 0x04C11DB7

位/域	名称	R/W	描述
31:0	PPOL	R/W	可编程多项式 (Programmable Polynomial) 可写入可编程多项式系数, 如果系数小于 32 位, 则必须使用最低有效位编程正确值。

28 芯片电子签名（Chip Electronic Signature）

28.1 简介

芯片电子签名包括主存储器 Flash 容量信息、96 位唯一芯片 ID，出厂前写入芯片的系统存储区，用户只读且不可修改。

28.2 功能描述

96 位芯片的 ID 主要用处：

- 作为序列号
- 作为密码，在编写闪存时，代码和密码通过算法结合，可以提高代码在 Flash 内的安全性
- 用于启动配置
- 身份标识所提供的参考号码对任一 MCU 微控制器系列都是唯一的，无论在什么情况下，用户都不能改变这唯一的身份标识。根据不同用法，用户可以选择以字节、半字或全字为单位读取身份标识。

28.3 寄存器功能描述

28.3.1 96 位唯一芯片 ID

基地址：0x1FFF F7AC

偏移地址：0x00

位/域	名称	R/W	描述
31:0	U_ID[31:0]	R	唯一身份标志 31:0 位

偏移地址：0x04

只读，其值在出厂时编写

位/域	名称	R/W	描述
31:0	U_ID[63:32]	R	唯一身份标志 63:32 位

偏移地址：0x08

只读，其值在出厂时编写

位/域	名称	R/W	描述
31:0	U_ID[95:64]	R	唯一身份标志 95:64 位

28.3.2 闪存容量寄存器（16 位）

基地址：0x1FFF F7CC

偏移地址：0x00

只读，其值在出厂时编写

位/域	名称	R/W	描述
15:0	F_SIZE	R	闪存存储器容量 (Flash Size) 表示产品的主存储区的容量 (以 K 字节为单位)。 例: 0x0040=64 K 字节

29 版本历史

表格 103 文档版本历史

日期	版本	变更历史
2023.3	V0.1	新建
2023.7	V0.2	修改格式
2023.11	V0.3	(1) 修改 GPIO 章节的 I/O 结构图

声明

本手册由珠海极海半导体有限公司（以下简称“极海”）制订并发布，所列内容均受商标、著作权、软件著作权相关法律法规保护，极海保留随时更正、修改本手册的权利。使用极海产品前请仔细阅读本手册，一旦使用产品则表明您（以下称“用户”）已知悉并接受本手册的所有内容。用户必须按照相关法律法规和本手册的要求使用极海产品。

1、权利所有

本手册仅应当被用于与极海所提供的对应型号的芯片产品、软件产品搭配使用，未经极海许可，任何单位或个人均不得以任何理由或方式对本手册的全部或部分内容进行复制、抄录、修改、编辑或传播。

本手册中所列带有“®”或“™”的“极海”或“Geehy”字样或图形均为极海的商标，其他在极海产品上显示的产品或服务名称均为其各自所有者的财产。

2、无知识产权许可

极海拥有本手册所涉及的全部权利、所有权及知识产权。

极海不应因销售、分发极海产品及本手册而被视为将任何知识产权的许可或权利明示或默示地授予用户。

如果本手册中涉及任何第三方的产品、服务或知识产权，不应被视为极海授权用户使用前述第三方产品、服务或知识产权，除非在极海销售订单或销售合同中另有约定。

3、版本更新

用户在下单购买极海产品时可获取相应产品的最新版的手册。

如果本手册中所述的内容与极海产品不一致的，应以极海销售订单或销售合同中的约定为准。

4、信息可靠性

本手册相关数据经极海实验室或合作的第三方测试机构批量测试获得，但本手册相关数据难免会出现校正笔误或因测试环境差异所导致的误差，因此用户应当理解，极海对本手册中可能出现的该等错误无需承担任何责任。本手册相关数据仅用于指导用户作为性能参数参照，不构成极海对任何产品性能方面的保证。

用户应根据自身需求选择合适的极海产品，并对极海产品的应用适用性进行有效验证和测试，以确认极海产品满足用户自身的需求、相应标准、安全或其它可靠性要求；若因用户

未充分对极海产品进行有效验证和测试而致使用户损失的，极海不承担任何责任。

5、合规要求

用户在使用本手册及所搭配的极海产品时，应遵守当地所适用的所有法律法规。用户应了解产品可能受到产品供应商、极海、极海经销商及用户所在地等各国有关出口、再出口或其它法律的限制，用户（代表其本身、子公司及关联企业）应同意并保证遵守所有关于取得极海产品及 / 或技术与直接产品的出口和再出口适用法律与法规。

6、免责声明

本手册由极海“按原样”（as is）提供，在适用法律所允许的范围内，极海不提供任何形式的明示或暗示担保，包括但不限于对产品适销性和特定用途适用性的担保。

对于用户后续在针对极海产品进行设计、使用的过程中所引起的任何纠纷，极海概不承担责任。

7、责任限制

在任何情况下，除非适用法律要求或书面同意，否则极海和/或以“按原样”形式提供本手册的任何第三方均不承担损害赔偿责任，包括任何一般、特殊因使用或无法使用本手册相关信息而产生的直接、间接或附带损害（包括但不限于数据丢失或数据不准确，或用户或第三方遭受的损失）。

8、适用范围

本手册的信息用以取代本手册所有早期版本所提供的信息。

©2023 珠海极海半导体有限公司 – 保留所有权利