用户手册

APM32E103xCxE

基于 Arm® Cortex®-M3 内核的 32 位微控制器

版本：V 1.3

© 珠海极海半导体有限公司

目录

1 简介及文档描述规则 8
1.1 简介 8
1.2 文档描述规则 8
2 系统架构 11
2.1 术语全称、缩写描述 11
2.2 系统架构框图 11
2.3 存储器映射 14
2.4 启动配置 14
3 FLASH 存储器 16
3.1 术语全称、缩写描述 16
3.2 主要特征 16
3.3 FLASH 存储器结构 16
3.4 FLASH 存储器功能说明 17
3.5 选项字节寄存器功能描述 19
3.6 FMC 寄存器地址映射 20
3.7 FMC 寄存器功能描述 21
4 外部存储器控制器（EMMC） 25
4.1 术语全称、缩写描述 25
4.2 EMMC 概述 25
4.3 SMC 简介 25
4.4 SMC 结构框图 26
4.5 SMC 功能描述 26
4.6 SMC 寄存器地址映射 31
4.7 SMC 寄存器功能描述 31
4.8 DMC 简介 40
4.9 DMC 主要特征 40
4.10 DMC 结构框图 41
4.11 DMC 功能描述 41

4.12 DMC 寄存器地址映射 44
4.13 DMC 寄存器功能描述 44
5 复位与时钟管理（RCM） 50
5.1 术语全称、缩写描述 50
5.2 复位功能描述 50
5.3 时钟管理功能描述 52
5.4 寄存器地址映射 58
5.5 寄存器功能描述 58
6 电源管理单元（PMU） 74
6.1 术语全称、缩写描述 74
6.2 简介 74
6.3 结构框图 74
6.4 功能描述 75
6.5 寄存器地址映射 79
6.6 寄存器功能描述 79
7 备份寄存器（BAKPR） 82
7.1 简介 82
7.2 主要特征 82
7.3 功能描述 82
7.4 寄存器地址映射 83
7.5 寄存器功能描述 83
8 嵌套向量中断控制器（NVIC） 86
8.1 术语全称、缩写描述 86
8.2 简介 86
8.3 主要特征 86
8.4 中断和异常向量表 86
9 外部中断/事件控制器（EINT） 90
9.1 简介 90
9.2 功能描述 90
9.3 寄存器地址映射 92

9.4 寄存器功能描述 92
10 直接存储器存取（DMA） 95
10.1 术语全称、缩写描述 95
10.2 简介 95
10.3 主要特征 95
10.4 功能描述 96
10.5 寄存器地址映射 101
10.6 寄存器功能描述 102
11 调试 MCU（DBGMCU） 106
11.1 术语全称、缩写描述 106
11.2 简介 106
11.3 主要特征 106
11.4 功能描述 107
11.5 寄存器地址映射 108
11.6 寄存器功能描述 108
12 通用输入/输出引脚（GPIO） 111
12.1 术语全称、缩写描述 111
12.2 主要特征 111
12.3 结构框图 112
12.4 功能描述 112
12.5 寄存器地址映射 115
12.6 寄存器功能描述 116
13 复用功能输入输出引脚（AFIO） 119
13.1 简介 119
13.2 功能描述 119
13.3 寄存器地址映射 123
13.4 寄存器功能描述 124
14 定时器概述 130
14.1 术语全称、缩写描述 130
14.2 定时器类别及主要差异 130

15 高级定时器（TMR1/8）.....133
15.1 简介.....133
15.2 主要特征.....133
15.3 结构框图.....134
15.4 功能描述.....134
15.5 寄存器地址映射.....150
15.6 寄存器功能描述.....151
16 通用定时器（TMR2/3/4/5）.....169
16.1 简介.....169
16.2 主要特征.....169
16.3 结构框图.....170
16.4 功能描述.....170
16.5 寄存器地址映射.....182
16.6 寄存器功能描述.....183
17 基本定时器（TMR6/7）.....198
17.1 简介.....198
17.2 主要特征.....198
17.3 结构框图.....198
17.4 功能描述.....198
17.5 寄存器地址映射.....200
17.6 寄存器功能描述.....200
18 看门狗定时器（WDT）.....204
18.1 简介.....204
18.2 独立看门狗定时器（IWDT）.....204
18.3 窗口看门狗定时器（WWDT）.....205
18.4 IWDT 寄存器地址映射.....207
18.5 IWDT 寄存器功能描述.....207
18.6 WWDT 寄存器地址映射.....208
18.7 WWDT 寄存器功能描述.....209
19 实时时钟（RTC）.....211

19.1 术语全称、缩写描述 211
19.2 主要特征 211
19.3 结构框图 211
19.4 功能描述 212
19.5 寄存器地址映射 213
19.6 寄存器功能描述 213
20 通用同步异步收发器（USART） 217
20.1 术语全称、缩写描述 217
20.2 简介 217
20.3 主要特征 217
20.4 功能描述 218
20.5 寄存器地址映射 230
20.6 寄存器功能描述 231
21 内部集成电路接口（I2C） 238
21.1 术语全称、缩写描述 238
21.2 简介 238
21.3 主要特征 238
21.4 结构框图 239
21.5 功能描述 240
21.6 寄存器地址映射 246
21.7 寄存器功能描述 246
22 内部集成电路接口（I2C3） 255
22.1 简介 255
22.2 功能描述 255
22.3 寄存器地址映射 265
22.4 寄存器功能描述 267
23 串行外设接口/片上音频接口（SPI/I2S） 287
23.1 术语全称、缩写描述 287
23.2 简介 287
23.3 主要特征 287

23.4 SPI 功能描述 288
23.5 I2S 功能描述 298
23.6 寄存器地址映射 308
23.7 寄存器功能描述 308
24 控制器局域网（CAN） 315
24.1 术语全称、缩写描述 315
24.2 简介 315
24.3 主要特性 315
24.4 功能描述 316
24.5 寄存器地址映射 323
24.6 寄存器功能描述 324
25 安全数字输入输出接口（SDIO） 338
25.1 术语全称、缩写描述 338
25.2 简介 338
25.3 主要特征 338
25.4 功能描述 338
25.5 寄存器地址映射 358
25.6 寄存器功能描述 358
26 全速 USBD 接口设备（USBD） 369
26.1 简介 369
26.2 功能描述 369
26.3 寄存器地址映射 369
26.4 寄存器功能描述 370
27 模数转换器（ADC） 379
27.1 术语全称、缩写描述 379
27.2 简介 380
27.3 主要特征 380
27.4 功能描述 381
27.5 寄存器地址映射 390
27.6 寄存器功能描述 391

28 数模转换器（DAC） 400
28.1 术语全称、缩写描述 400
28.2 简介 400
28.3 结构框图 400
28.4 功能描述 400
28.5 寄存器地址映射 403
28.6 寄存器功能描述 404
29 循环冗余校验计算单元（CRC） 409
29.1 简介 409
29.2 功能描述 409
29.3 寄存器地址映射 409
29.4 寄存器功能描述 409
30 浮点运算单元（FPU） 411
30.1 简介 411
30.2 功能描述 411
30.3 寄存器地址映射 414
30.4 寄存器功能描述 414
31 芯片电子签名 417
31.1 主存储区容量寄存器 417
31.2 96 位唯一芯片 ID 417
32 版本历史 419

1 简介及文档描述规则

1.1 简介

本用户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®-M3 内核的相关信息，请参考 Arm® Cortex®-M3 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 方式缩写及描述

R/W 方式	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，写该位无效。	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

中文全称	英文全称	英文缩写
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
周期	Period	PRD

表格 3 模块全称、简写

中文全称	英文全称	英文缩写
外部存储控制器	External Memory Controller	EMMC
静态存储控制器	Static Memory Controller	SMC
动态存储控制器	Dynamic memory Controller	DMC
复位与时钟管理单元	Reset and Clock Management Unit	RCM
电源管理单元	Power Management Unit	PMU
备份寄存器	Backup Register	BAKPR
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
四线串行外围接口	Quad Serial Peripheral Interface	QSPI
控制器局域网	Controller Area Network	CAN
安全数字输入输出	Secure Digital Input and Output	SDIO
全速 USBD 接口	Universal Serial Bus Full-Speed Device	USBD
模拟数字转换器	Analog-to-Digital Converter	ADC
数字模拟转换器	Digital-to-Analog Converter	DAC
循环冗余校验计算单元	Cyclic Redundancy Check Calculation Unit	CRC
浮点运算单元	Float Point Unit	FPU

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

主系统主要由四个驱动单元和四个被动单元构成。四个驱动单元分别是连接 Arm® Cortex®-M3 内核的 DCode 总线（D-bus）、系统总线（S-bus）、通用 DMA1 和 DMA2。四个被动单元分别是内部 SRAM、内部闪存存储器、EMMC 和 AHB 到 APB 的桥（AHB/APBx），其中，AHB/APBx 连接所有的 APB 设备。这些都是通过一个多级的 AHB 总线构架相互连接的。

由于 APM32E103 系列在封装上分为合封和非合封两种类型（尾缀有 S 的为合封产品），所以这两种型号的系统框图分开描述，如下所示：

图 1 APM32E103xCxE 系统架构框图

图 2 APM32E103VET6S 系统架构框图

注意：

（1）APM32E103xCxE 产品不同型号的模块数量请参见数据手册。

（2）实际上 APM32E103xCxE 系列有两个 USBD，但是 USBD1/2 所用的引脚、地址相同，所以两个 USBD 不能同时使用，相当于只有一个 USBD。

表格 5 总线名称

名称	说明
ICode Bus	连接 Arm® Cortex®-M3 内核的指令总线与闪存指令接口。用于预取指令。

名称	说明
DCode Bus	连接 Arm® Cortex®-M3 内核的 DCode 总线与闪存存储器的数据接口。用于加载常量和访问调试。
系统总线	连接 Arm® Cortex®-M3 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；协调 CPU 的 DCode 和 DMA 到 SRAM、Flash 和外设的访问。AHB 外设通过总线矩阵与系统总线相连，允许 DMA 访问。
AHB/APB 桥	两个桥在 AHB 和两个 APB 总线间提供同步连接。APB1，APB2 的最高操作速度不同。 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、EMMC、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。

2.3.2 位段

Arm® Cortex®-M3 存储器映射有两个位段（bit-band）区，它将每个在别名存储器区中的字映射到位段存储器的一个位。在别名存储区写入一个字，会有对位段区的目标执行读-改-写操作的相同效果。外设寄存器和 SRAM 都被映射到一个位段区里，允许执行单一的位段的写和读操作。

下面给出一份映射公式：

bit_word_addr=bit_band_base+（byte_offset×32）+（bit_number×4）

详细信息请参考《Arm® Cortex®-M3 技术参考手册》

2.4 启动配置

由于 Arm® Cortex®-M3 内核的 CPU 从 ICode Bus（指令总线）获取复位向量，导致启动只能从代码区开始，典型为 Flash 存储器启动。但是，APM32E103 微控制器系列实现了一个特殊的机制，通过配置 BOOT[1:0]引脚参数，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 F000

（4）SRAM 的起始地址为 0x2000 0000

用户可以通过设置 BOOT1 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，通过 USRAT1 或 USART2（重映射）启用启动程序，可以对 Flash 存储器进行重新编程。

3 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。

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

3.2 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 512KB
- 信息块分为系统存储区、选项字节
- 系统存储区容量大小为 2KB，存放 BootLoader 程序、96 位唯一 UID、主存储区容量信息
- 选项字节区容量大小为 16Bytes

（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

3.3 Flash 存储器结构

表格 8 APM32E103xCxE 系列产品 Flash 存储结构

块	名称	地址范围	大小（字节）
主存储块	页 0	0x0800 0000 – 0x0800 07FF	2K
主存储块	页 1	0x0800 0800 – 0x0800 0FFF	2K
主存储块	页 2	0x0800 1000 – 0x0800 17FF	2K
主存储块	页 3	0x0800 1800 – 0x0800 1FFF	2K
主存储块	…	…	…
主存储块	页 255	0x0807 F800 – 0x0807 FFFF	2K
信息块	系统存储区	0x1FFF F000 – 0x1FFF F7FF	2K
信息块	选项字节	0x1FFF F800 – 0x1FFF F80F	16

注：APM32E103xCxE 系列产品的主存储块包含页的数量和具体产品的 Flash 容量有关。

3.4 Flash 存储器功能说明

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。

3.4.1 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
- 3 个等待周期：72MHz<系统时钟≤96MHz
- 4 个等待周期：96MHz<系统时钟≤120MHz

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；只有系统时钟和 AHB 时钟一致且系统时钟小于 24MHz 时才能配置开启或关闭。

半周期访问

在需要优化功耗时可使用半周期访问；此时系统时钟和 AHB 时钟一致，且系统时钟为 8MHz 或低于 8MHz 才能使用半周期访问 Flash，否则必须处于开启状态。

3.4.2 主存储块

3.4.2.1 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）。

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。

3.4.2.2 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"）。

3.4.2.3 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 APM32E103xCxE 系列产品，读/写保护的基本单位是 2 页（即 4KBytes）。

读保护

可通过修改选项字节 READPROT 的值设置内部 Flash 保护级别。调试器一直连接 JTAG/SWD 接口设置读保护，需上电复位后生效；否则，系统电复位就能能有效。READPROT 值为除 0xA5 外的任意值时，开启读保护，无法读取主存储块的内容；READPROT 值为 0xA5 时，解除保护，可读取主存储块的内容；在解除读保护时会触发一个主存储块片擦除操作，以防止保护降级后的非法读取。

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。

3.4.2.4 解锁/锁定主存储块

复位后 FMC 的 FMC_CTRL1 会被硬件锁定，此时不能直接对 FMC_CTRL1 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 解锁一用户编程一 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。

3.4.3 选项字节

3.4.3.1 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。

3.4.3.2 写选项字节

选项字节的 8 个可配置字节均支持写入功能。

3.4.3.3 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，FMC_CTRL2 寄存器的 OBWEN 位标示允许写，清除这位将禁止写操作。

3.4.3.4 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。

3.5 选项字节寄存器功能描述

选项字节为用户提供了一些可供选择的功能，它主要由 8 个可配置的字节和对应的补码组成。在每次系统复位后，选项字节区将被重加载到 FMC_OBCS 和 FMC_WRTPROT 寄存器（选项字节只有每次被重加载到 FMC 后中才会生效）。在重加载过程中，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_OBCS 寄存器的 OBE 位置"1"），同时这个字节将会被置为"0xFF"。选项字节区 16 字节的信息见下表。

表格 9 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
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 0：进入 Standby 模式时产生复位

地址	选项字节	初始值	R/W	功能描述
				1：进入 Standby 模式时不产生复位 [3:7]：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

注意：当可配置字节和它的补码值为"0xFF"时，重加载过程中将不会再进行匹配验证

表格 10 主存储区写保护 WRPx 功能描述

产品容量	功能描述
APM32E103xCxE 系列产品 （2KB/页）	WRPx 中的每一个 bit 位控制主存储区 4KB（2 页）地址的写保护 0：开启写保护 1：未开启写保护 WRP0：第 0-15 页 WRP1：第 16-31 页 WRP2：第 32-47 页 WRP3：Bit0-Bit6 控制第 48-61 页;Bit7 控制第 62-255 页

注意：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。

3.6 FMC 寄存器地址映射

基地址：0x40022000

表格 11 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

寄存器名	描述	偏移地址
FMC_ADDR	闪存地址寄存器	0x14
FMC_OBCS	选项字节控制/状态寄存器	0x1C
FMC_WRTPROT	写保护寄存器	0x20

3.7 FMC 寄存器功能描述

3.7.1 控制寄存器 1（FMC_CTRL1）

偏移地址：0x00

复位值：0x0000 0030

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 001：1 个等待周期 010：2 个等待周期 011：3 个等待周期 100：4 个等待周期 其他：保留
3	HCAEN	R/W	使能 Flash 半周期访问（Flash Half Cycle Access Enable） 0：禁止 1：使能
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		

3.7.2 关键字寄存器（FMC_KEY）

偏移地址：0x04

复位值：0xXXXX 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。

3.7.3 选项字节关键字寄存器（FMC_OBKEY）

偏移地址：0x08

复位值：0xXXXX 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。

3.7.4 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		

3.7.5 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		

位/域	名称	R/W	描述
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		
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。
31:13	保留		

3.7.6 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。

3.7.7 选项字节控制/状态寄存器（FMC_OBCS）

偏移地址：0x1C

复位值：0x03FF FFFC

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
1	READPROT	R	读保护（Read Protect） 1：表示闪存处于读保护状态
9:2	UOB	R	用户选项字节（User Option Byte） 这里包含 OBL 加载的用户选择字节 位 2：WDTSEL 位 3：nRSTSTOP 位 4：nRSTSTDB 位[9:5]：未用
17:10	DATA0	R	Data0
25:18	DATA1	R	Data1
31:26	保留		

3.7.8 写保护寄存器（FMC_WRTPROT）

偏移地址：0x20

复位值：0xFFFF FFFF

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

4 外部存储器控制器（EMMC）

4.1 术语全称、缩写描述

表格 12 术语全称、缩写描述

中文全称	英文全称	英文缩写
静态随机存储器	Static Random Access Memory	SRAM
只读存储器	Read Only Memory	ROM
伪静态随机存储器	Pseudo Static Random Access Memory	PSRAM
随机存取存储器	Random Access Memory	RAM
同步动态随机存储器	Synchronous Dynamic Random Access Memory	SDRAM
复用	Multiplex	MUX
宽度	Width	WID
闪存	Flash Memory	FM
访问	Access	ACC
等待	Wait	W
信号	Signal	S
极性	Polarity	POL
异步	Asynchronous	ASYN
突发	Burst	BURST
时序	Timing	TIM
建立	Setup	SET
保持	Hold	HLD
空	Empty	E

4.2 EMMC 概述

EMMC 包括 SMC（静态存储控制器）、DMC（动态存储控制器）。SMC 负责控制 SRAM、PSRAM、NandFlash、NorFlash、PCCard，DMC 控制 SDRAM。

4.3 SMC 简介

SMC 是用来管理扩展静态存储器的外设；可以将 AHB 传输信号转换到适当的外部设备；内部存在四个存储块，每个存储块都对应控制不同类型的存储器，通过片选信号加以区分；任一时刻只访问一个外部设备；每个存储块都可以单独配置，时序可编程以适用外部设备。

4.4 SMC 结构框图

SMC 主要有五个部分：AHB 总线接口、配置寄存器、NORFlash 控制器、NANDFlash/PC 卡控制器和外部设备接口，具体情况如下图：

图 3 SMC 框图

4.5 SMC 功能描述

4.5.1 SMC 访问规则

SMC 是内部 CPU 通过 AHB 总线访问外部静态存储器的接口，在 AHB 总线上，32 位长度数据的操作会被转换成连续的 16 或 8 位操作。为了保证数据传输时的一致性，SMC 在对外部读写操作时需要遵从以下规则：

（1） AHB 访问外部数据宽度等于存储器数据宽度时，可正常传输，没有问题。

（2） AHB 访问外部数据宽度大于存储器数据宽度时，自动将访问操作切割成与外部数据宽度一致进行传输。

（3） AHB 访问外部数据宽度小于存储器数据宽度时，如果外部存储器具有字节选择功能，则可以通过字节通道正常传输数据；如果不具有该功能，则不允许写操作，只允许读操作。

4.5.2 外部设备地址映射

SMC 将外部设备分成多个存储块，不同的存储块对应控制不同的外部设备，具体分类见下表：

表格 13 外部设备地址映射表

起始地址	结束地址	存储块	支持存储器类型
0x60000000	0x6FFFFFFF	存储块 1（4*64MB）	NOR/PSRAM
0x70000000	0x7FFFFFFF	存储块 2（4*64MB）	NAND
0x80000000	0x8FFFFFFF	存储块 3（4*64MB）	NAND
0x90000000	0x9FFFFFFF	存储块 4（4*64MB）	PC 卡

4.5.3 NOR 闪存和 PSRAM

4.5.3.1 地址映射

存储块 1 用于控制 NOR/PSRAM 存储器，内部分成 4 块 64MB 同等大小大小的区域，各个区域的选择取决于 HADDR[27:26]的值，具体信息如下：

表格 14 存储块 1 地址映射

HADDR[27:26]	起始地址	结束地址	区域块
00	0x60000000	0x63FFFFFF	区域块 1
01	0x64000000	0x67FFFFFF	区域块 2
10	0x68000000	0x6BFFFFFF	区域块 3
11	0x6C000000	0x6FFFFFFF	区域块 4

HADDR 是需要转换到外部存储器的内部 AHB 地址线，它是字节地址，但是外部存储器有一些是不按字节访问的，所以会出现地址不一致的情况，为了避免这种情况产生的错误，SMC 会遵从下述规则进行相应的调整：

- 当外部存储器数据宽度为 8 位时，HADDR[25:0]与 SMC_A[25:0]对应连接，而 SMC_A[25:0]与外部存储器地址线相连。
- 当外部存储器数据宽度为 16 位时，HADDR[25:1]与 SMC_A[24:0]对应连接，而 SMC_A[24:0]与外部存储器地址线相连。

4.5.3.2 接口信号和控制器

存储块 1 支持 NOR Flash、PSRAM、SRAM 和 ROM 外部存储器。存储块 1 中的四个区域都存在对应的片选信号 NE[x]（x=1..4），所有其它信号都是共享的，具体接口信号及功能如下：

表格 15 NOR Flash 接口信号

SMC 信号名称	信号方向	功能
CLK	输出	同步时钟信号
NE[x]	输出	片选信号，x=1..4
NOE	输出	读使能信号
NEW	输出	写使能信号
NWAIT	输入	NOR 闪存要求 SMC 等待的信号

SMC 信号名称	信号方向	功能
A[25:0]	输出	非复用：A[25:0]全部为地址总线
	输出	复用：A[25:16]为地址总线
AD[15:0]	输入/输出	非复用：双向数据总线
	输入/输出	复用：双向地址/数据总线
NL（=NADV）	输出	地址有效信号

表格 16 PSRAM 接口信号

SMC 信号名称	信号方向	功能
CLK	输出	同步时钟信号
NE[x]	输出	片选信号，x=1..4
NOE	输出	读使能信号
NEW	输出	写使能信号
NWAIT	输入	PSRAM 要求 SMC 等待的信号
A[25:0]	输出	地址总线
D[15:0]	输入/输出	双向数据总线
NL（=NADV）	输出	地址有效信号
NBL[1]	输出	高字节使能
NBL[0]	输出	低字节使能

注意：控制器的输出信号在内部时钟的上升沿变化；在同步写模式下，输出的数据在存储器时钟的下降沿变化。

NOR Flash/PSRAM 控制器给外部存储器提供了可编程的时序参数，具体包含参数如下表：

表格 17 可编程的 NOR/PSRAM 时序参数

参数	功能	访问方式	单位	最小	最大
数据产生时间	突发模式下产生第一个数据所需的时钟数目	同步	存储器时钟周期（CLK）	2	17
时钟分频因子	存储器访问的时钟周期（CLK）与 AHB 时钟周期的比例	同步	AHB 时钟周期（HCLK）	2	16
总线恢复时间	总线恢复阶段的时间	异步或同步读		1	16
数据建立时间	数据建立阶段的时间	异步		2	256
地址保持时间	地址保持阶段的时间	异步，复用 IO		2	16
地址建立时间	地址建立阶段的时间	异步		1	16

4.5.4 NAND 闪存和 PC 卡

4.5.4.1 地址映射

存储块 2、3 和 4 是用来访问 NAND 闪存和 PC 卡，同样在每个存储块内也被分为不同的区域，不同区域对应效果不一样，具体分布如下表：

表格 18 存储块 2、3、4 地址映射

SMC 存储块	存储空间	起始地址	结束地址
存储块 2-NAND 闪存	通用	0x70000000	0x73FFFFFF
	属性	0x78000000	0x7BFFFFFF
存储块 3-NAND 闪存	通用	0x80000000	0x83FFFFFF
	属性	0x88000000	0x8BFFFFFF
存储块 4-PC 卡	通用	0x90000000	0x93FFFFFF
	属性	0x98000000	0x9BFFFFFF
	I/O	0x9C000000	0x9FFFFFFF

NAND 闪存存储块中又在低字节一部分区域划分了三个块，通过 HADDR[17:16] 来选择对不同块进行访问，三个块的具体划分及其选择如下表：

表格 19 NAND 存储块划分

HADDR[17:16]	地址范围	块名称
00	0x000000-0x00FFFF	数据块
01	0x010000-0x01FFFF	命令块
1X	0x020000-0x03FFFF	地址块

为实现对 NAND 存储器的正常读写，需要有以下步骤：

- 向存储器发送命令
- 向存储器发送需要读写的地址
- 读写数据

对应的三步操作需要操作的地址便对应存储块中所划分的三个块。向存储器发送命令便是对命令块写入相应命令值；向存储器发送地址便是向地址块发送相应地址值；读写数据便是在数据块中读写，最终写入或读出 NAND 的内部单元，对应单元的地址便是在地址块写入的地址。

4.5.4.2 接口信号和控制器

NAND/PC 卡控制器可以控制三个存储块，存储块 2、3 支持 NAND Flash，存储块 4 支持 PC 卡设备，三个存储块也都有各自的片选信号，具体接口及功能如下：

表格 20 NAND Flash 接口信号

SMC 信号名称	信号方向	功能
NCE[x]	输出	片选信号，x=2,3

SMC 信号名称	信号方向	功能
NOE（=NRE）	输出	读使能信号
NEW	输出	写使能信号
NWAIT/INT[3:2]	输入	NAND Flash 就绪/忙输入信号
A[17]	输出	NAND Flash 地址锁存信号（ALE）
A[16]	输出	NAND Flash 命令锁存信号（CLE）
D[15:0]	输入/输出	8 位复用：D[7:0]双向地址/数据总线
	输入/输出	16 位复用：D[15:0]双向地址/数据总线

表格 21 PC 卡接口信号

SMC 信号名称	信号方向	功能
NCE4_1	输出	片选信号 1
NCE4_2	输出	片选信号 2（选择 16 位或 8 位操作）
NOE	输出	读使能信号
NEW	输出	写使能信号
NWAIT	输入	PC 卡等待信号
INTR	输入	PC 卡中断信号
CD	输入	PC 卡检测信号
A[10:0]	输出	地址总线
NIOS16	输入	16 位传输 I/O 空间的数据传输宽度（须接地）
NIORD	输出	I/O 空间输出使能
NIOWR	输出	I/O 空间写使能
NREG	输出	对通用空间或者属性空间访问的选择
D[15:0]	输出/输入	双向数据总线

NAND Flash/PC 卡控制器给外部存储器提供了可编程的时序参数，具体包含参数如下表：

表格 22 可编程 NAND/PC 卡时序参数

参数	功能	操作模式	单位	最小	最大
存储器数据总线高阻时间	启动写操作后保持数据总线为高阻态的时间	写	AHB 时钟周期（HCLK）	0	255
存储器保持时间	发送命令结束后保持地址的时钟数目，写操作时也是数据的保持时间	读/写		1	254
存储器等待时间	发出的最短持续时间			2	256
存储器建立时间	发出命令之前建立地址的时钟数目			1	255

4.6 SMC 寄存器地址映射

表格 23 SMC 寄存器地址映射

寄存器名	描述	偏移地址
SMC_CSCTRL1…4	SRAM/NOR 闪存片选控制寄存器 1…4	8*(x-1),x=1…4
SMC_CSTIM1…4	SRAM/NOR 闪存片选时序寄存器 1…4	0x04 + 8*(x-1),x=1…4
SMC_WRTTIM1…4	SRAM/NOR 闪存写时序寄存器 1…4	0x104 + 8*(x-1),x=1…4
SMC_CTRL2…4	PC 卡/NAND 闪存控制寄存器 2…4	0x40 + 0x20 * (x-1),x=2…4
SMC_STSINT2…4	FIFO 状态和中断寄存器 2…4	0x44 + 0x20 * (x-1),x=2…4
SMC_CMSTIM2…4	通用存储空间时序寄存器 2…4	0x48 + 0x20 * (x-1),x=2…4
SMC_AMSTIM2…4	属性存储空间时序寄存器 2…4	0x4C + 0x20 * (x-1),x=2…4
SMC_IOSTIM4	I/O 空间时序寄存器 4	0xB0
SMC_ECCRS2/3	ECC 结果寄存器 2/3	0x54 + 0x20 * (x-1),x=2 或 3

4.7 SMC 寄存器功能描述

4.7.1 NOR 闪存和 PSRAM 控制寄存器

4.7.1.1 SRAM/NOR 闪存片选控制寄存器 1…4（SMC_CSCTRL1…4）

偏移地址：8*(x-1),x=1…4

复位值：0x0000 30DX

位/域	名称	R/W	描述
0	MBKEN	R/W	使能对应的存储器块（Enable the Corresponding Memory Bank） 0：禁止 1：使能
1	ADMUXEN	R/W	使能地址/数据复用（Address/Data Multiplexing Enable） 该位仅对 NORFlash 和 PSRAM 有效。 0：禁止 1：地址低 16 位和数据共用数据总线
3:2	MTYPECFG	R/W	配置存储器类型（Memory Type Configure） 00：SRAM、ROM（Bank2~Bank4 复位后的默认值） 01：PSRAM 10：NORFlash（Bank1 复位后的默认值） 其他保留
5:4	MDBWIDCFG	R/W	配置存储器数据总线宽度（Memory Data Bus Width Configure） 00：8 位 01：16 位 其他保留

位/域	名称	R/W	描述
6	NORFMACCEN	R/W	使能访问 NORFlash（NORFlash Memory Access Enable）） 0：禁止 1：使能
7	保留		
8	BURSTEN	R/W	使能突发模式（Burst Mode Enable） 在同步模式下，可使用突发模式访问存储器。 0：禁止 1：使能
9	WSPOLCFG	R/W	配置等待信号极性（Wait Signal Polarity Configure） 该位仅在突发模式下有效。 0：低有效 1：高有效
10	WRAPBEN	R/W	使能非对齐的突发模式（Wrapped Burst Mode Enable）） 该位仅在突发模式下有效。 0：不允许 1：允许
11	WTIMCFG	R/W	配置等待时序（Wait Timing Configure） 该位用来配置存储器是在等待状态前一个周期产生 NWAIT 信号，还是在等待期间产生；该位仅在突发模式下有效。 0：NWAIT 信号在等待前的一个数据周期有效 1：NWAIT 信号在等待期间有效
12	WREN	R/W	使能写存储器（Write Memory Enable） 该位用来使能 SMC 对存储器的写操作。 0：禁止写，否则产生一个 AHB 错误 1：允许写
13	WAITEN	R/W	等待使能（Wait Enable） 该位用来使能 NWAIT 信号插入等待状态，该位仅在突发模式下有效。 0：禁止 1：使能
14	EXTMODEEN	R/W	使能扩展模式（Extended Mode Enable） 在扩展模式下，可使用 SMC_WRTTIM 寄存器，实现读、写使用不同的时序的功能。 0：禁止 1：使能
15	WSASYNCEN	R/W	使能异步传输期间的等待信号（Wait Signal During Asynchronous Transfers Enable） 该位用来在异步协议期间 SMC 可使用 NWAIT 信号。 0：禁止 1：使能

位/域	名称	R/W	描述
18:16	CRAMPSIZECFG	R/W	配置 CRAM 页大小（CRAM Page Size Configure） 000：当跨越页面边界时没有突发分裂 001：128 字节 010：256 字节 011：512 字节 100：1024 字节 其他：保留
19	WRBURSTEN	R/W	使能突发写 PSRAM（Write PSRAM Burst Enable） 该位使能写操作的同步突发传输协议。 0：写操作为异步模式 1：写操作为同步模式
31:20	保留		

4.7.1.2 SRAM/NOR 闪存片选时序寄存器 1…4（SMC_CSTIM1…4）

偏移地址：0x04 + 8*(x-1),x=1…4

复位值：0x0FFF FFFF

位/域	名称	R/W	描述
3:0	ADDRSETCFG	R/W	配置地址建立时间（Address Setup Time Configure） 仅适用于 SRAM、ROM 和异步总线复用模式的 NOR 闪存操作。 0000：1 个 HCLK 时钟周期 0001：2 个 HCLK 时钟周期 …… 1111：16 个 HCLK 时钟周期 注：同步操作中，该参数无含义，始终为 1 个存储器时钟周期
7:4	ADDRHLDCFG	R/W	配置地址保持时间（Address-Hold Time Configure） 仅适用于 SRAM、ROM 和异步总线复用模式的 NOR 闪存操作。 0000：保留 0001：2 个 HCLK 时钟周期 …… 1111：16 个 HCLK 时钟周期 注：同步操作中，该参数无含义，始终为 1 个存储器时钟周期
15:8	DATASETCFG	R/W	配置数据建立时间（Data Setup Time Configure） 仅适用于 SRAM、ROM 和异步总线复用模式的 NOR 闪存操作。 0000 0000：保留 0000 0001：2 个 HCLK 时钟周期 0000 0010：3 个 HCLK 时钟周期 …… 1111 1111：256 个 HCLK 时钟周期
19:16	BUSTURNCFG	R/W	配置总线恢复时间（Bus Turnaround Phase Duration Configure） 这些位用来配置一次读操作后在总线上的延迟时间，仅适用于总线复用模式的 NOR 闪存操作。 0000：1 个 HCLK 时钟周期 0001：2 个 HCLK 时钟周期 …… 1111：16 个 HCLK 时钟周期

位/域	名称	R/W	描述
23:20	CLKDIVCFG	R/W	配置时钟分频系数（Clock Divide Factor Configure） CLK 来自 HCLK 的分频，这些位用来配置 CLK 时钟输出信号的频率，仅适用于在同步模式。 0000：保留 0001：2 分频 0010：3 分频 …… 1111：16 分频 注：在访问异步 NOR 闪存、SRAM 或 ROM 时，该参数无效。
27:24	DATALATCFG	R/W	配置数据保持时间（Data Latency Configure） 这些位用来配置在读取第一个数据前等待的存储器周期数目，仅适用于同步突发模式的 NOR 闪存操作。 0000：2 个 CLK 时钟周期 0001：3 个 CLK 时钟周期 …… 1111：17 个 CLK 时钟周期 注：在访问异步 NOR 闪存、SRAM 或 ROM 时，该参数无效，操作 CRAM 时，该参数为 0。
29:28	ASYNCACCCFG	R/W	配置异步访问模式（Asynchronous Access Mode Configure） 仅在 SMC_CSCTRLX 寄存器的 EXTMODEEN 位为 1 时生效。 00：访问模式 A 01：访问模式 B 10：访问模式 C 11：访问模式 D
31:30	保留		

4.7.1.3 SRAM/NOR 闪存写时序寄存器 1…4（SMC_WRTTIM1…4）

偏移地址：0x104 + 8*(x-1), x=1…4

复位值：0x0FFF FFFF

位/域	名称	R/W	描述
3:0	ADDRSETCFG	R/W	配置地址建立时间（Address Setup Time Configure） 仅适用于 SRAM、ROM 和异步总线复用模式的 NOR 闪存操作。 0000：1 个 HCLK 时钟周期 0001：2 个 HCLK 时钟周期 …… 1111：16 个 HCLK 时钟周期 注：同步操作中，该参数无含义，始终为 1 个存储器时钟周期
7:4	ADDRHLDCFG	R/W	配置地址保持时间（Address-Hold Time Configure） 仅适用于 SRAM、ROM 和异步总线复用模式的 NOR 闪存操作。 0000：保留 0001：2 个 HCLK 时钟周期 …… 1111：16 个 HCLK 时钟周期 注：同步操作中，该参数无含义，始终为 1 个存储器时钟周期

位/域	名称	R/W	描述
15:8	DATASETCFG	R/W	配置数建立时间（Data-Setup Time Configure） 仅适用于 SRAM、ROM 和异步总线复用模式的 NOR 闪存操作。 0000 0000：保留 0000 0001：2 个 HCLK 时钟周期 0000 0010：3 个 HCLK 时钟周期 …… 1111 1111：256 个 HCLK 时钟周期
19:16	BUSTURNCFG	R/W	配置总线恢复时间（Bus Turnaround Phase Duration Configure） 这些位用来配置一次读操作后在总线上的延迟时间，仅适用于总线复用模式的 NOR 闪存操作。 0000：1 个 HCLK 时钟周期 0001：2 个 HCLK 时钟周期 …… 1111：16 个 HCLK 时钟周期
23:20	CLKDIVCFG	R/W	配置 CLK 信号的时钟分频比（Clock divide ratio (for CLK signal Configure） CLK 时钟输出信号的周期，用 HCLK 周期数表示。 0000：保留 0001：CLK 周期=2×HCLK 周期 0010：CLK 周期=3×HCLK 周期 1111：CLK 周期=16×HCLK 周期（复位后的默认值） 在异步 NOR Flash、SRAM 或 ROM 访问模式下，该位无效。
27:24	DATLATCFG	R/W	配置数据延迟时间（用于同步突发 NOR Flash）（Data latency Configure） 该位用来定义获取首个数据前要发送给存储器的存储器时钟周期数(+2)。 0000：首次突发访问时，2 个 CLK 时钟周期的数据延迟 … 1111：首次突发访问时，17 个 CLK 时钟周期的数据延迟（复位后的默认值） 在异步 NOR Flash、SRAM 或 ROM 访问模式下，该位无效。
29:28	ASYNCACCCFG	R/W	配置异步访问模式（Asynchronous Access Mode Configure） 这些位用来配置异步访问模式，仅在 SMC_CSCTRLX 寄存器的 EXTMODEEN 位为 1 时生效。 00：访问模式 A 01：访问模式 B 10：访问模式 C 11：访问模式 D
31:30	保留		

4.7.2 NAND 闪存和 PC 卡控制寄存器

4.7.2.1 PC 卡/NAND 闪存控制寄存器 2…4（SMC_CTRL2…4）

偏移地址：0x40 + 0x20 * (x-1),x=2…4

复位值：0x0000 0018

位/域	名称	R/W	描述
0	保留		
1	WAITFEN	R/W	使能 PC 卡/NANDFlash 等待功能（PC Card/NANDFlash Wait Feature Enable） 0：禁止 1：使能
2	MBKEN	R/W	使能 PC 卡/NAND 存储器块（PC Card/NAND Flash Memory Bank Enable） 0：禁止 1：使能
3	MTYPECFG	R/W	配置存储器类型（Memory Type Configure） 0：PC 卡、CF 卡、CF+卡或 PCMCIA 1：NAND 闪存
5:4	DBWIDCFG	R/W	配置数据总线宽度（Databus Width Configure） 对于 PC Card，必须使用 16 位。 00：8 位 01：16 位 其他保留
6	ECCEN	R/W	使能 ECC 计算电路（ECC Computation Logic Enable） 0：禁止并复位 ECC 1：使能
8:7	保留		
12:9	C2RDCFG	R/W	配置 CLE 至 RE 的延迟（CLE To RE Delay Configure） 配置“CLE 变为低电平”至“RE 变为低电平”的时间。 0000：1 个 HCLK 周期 0000：2 个 HCLK 周期 …… 1111：16 个 HCLK 周期
16:13	A2RDCFG	R/W	配置 ALE 至 RE 的延迟（ALE To RE Delay Configure） 配置“ALE 变为低电平”至“RE 变为低电平”的时间 0000：1 个 HCLK 周期 0000：2 个 HCLK 周期 …… 1111：16 个 HCLK 周期
19:17	ECCPSCFG	R/W	配置 ECC 页面大小（ECC Page Size Configure） 000：256 字节 001：512 字节 010：1024 字节 011：2048 字节 100：4096 字节 101：8192 字节
31:20	保留		

4.7.2.2 FIFO 状态和中断寄存器 2…4（SMC_STSINT2…4）

偏移地址：0x44 + 0x20 * (x-1),x=2…4

复位值：0x0000 0040

位/域	名称	R/W	描述
0	IREFLG	R/W	产生中断上升沿标志（Interrupt Rising Edge Generate Flag） 该位由硬件置 1，软件清 0。 0：没有产生 1：产生
1	IHLFLG	R/W	产生中断高电平标志（Interrupt High-Level Generate Flag） 该位由硬件置 1，软件清 0。 0：没有产生 1：产生
2	IFEFLG	R/W	产生中断下降沿标志（Interrupt Falling Edge Generate Flag） 该位由硬件置 1，软件清 0。 0：没有产生 1：产生
3	IREDEN	R/W	使能中断上升沿检测（Interrupt Rising Edge Detection Enable） 0：禁止 1：使能
4	IHLDEN	R/W	使能中断高电平检测（Interrupt High-Level Detection Enable） 0：禁止 1：使能
5	IFEDEN	R/W	使能中断下降沿检测（Interrupt Falling Edge Detection Enable） 0：禁止 1：使能
6	FEFLG	R	FIFO 空标志（FIFO Empty Flag） 0：不空 1：空
31:7	保留		

4.7.2.3 通用存储空间时序寄存器 2…4（SMC_CMSTIM2…4）

偏移地址：0x48 + 0x20 * (x-1),x=2…4

复位值：0xFCFC FCFC

位/域	名称	R/W	描述
7:0	SETx	R/W	配置在通用空间 x 的建立时间（Common Memory x Setup Time Configure） 该位以 HCLK 为时钟周期，定义了发送命令之前建立地址的时间。 0000 0000：1 个 HCLK 周期 0000 0001：2 个 HCLK 周期 …… 1111 1110：255 个 HCLK 周期 1111 1111：保留

位/域	名称	R/W	描述
15:8	WAITx	R/W	配置在通用空间 x 的等待时间（Common Memory x Wait Time Configure） 该位以 HCLK 为时钟周期，定义了保持命令的最小时间，定义的时间结束后，若等待信号有效低则命令保持时间变长。 0000 0000：保留 0000 0001：2 个 HCLK 周期（+由 NWAIT 信号变低引入的等待周期） 0000 0010：3 个 HCLK 周期 …… 1111 1110：255 个 HCLK 周期（+由 NWAIT 信号变低引入的等待周期） 1111 1111：保留
23:16	HLDx	R/W	配置在通用空间 x 的保持时间（Common Memory x Hold Time Configure） 该位以 HCLK 为时钟周期，定义了发送命令后，地址信号保持的时间。 0000 0000：保留 0000 0001：写访问 1 个 HCLK 周期，读访问 3 个 HCLK 周期 …… 1111 1110：写访问 254 个 HCLK 周期，读访问 256 个 HCLK 周期 1111 1111：保留
31:24	HIZx	R/W	配置在通用空间 x 数据总线的高阻时间（Common Memory x Databus Hiz Time Configure） 该位以 HCLK 为时钟周期，定义了数据总线高阻态的时间，仅对写操作生效。 0000 0000：1 个 HCLK 周期 0000 0001：2 个 HCLK 周期 …… 1111 1110：255 个 HCLK 周期 1111 1111：保留

4.7.2.4 属性存储空间时序寄存器 2…4（SMC_AMSTIM2…4）

偏移地址：0x4C + 0x20 * (x-1),x=2…4

复位值：0xFCFC FCFC

位/域	名称	R/W	描述
7:0	SETx	R/W	配置在属性空间 x 的建立时间（Attribute Memory x Setup Time Configure） 该位以 HCLK 为时钟周期,定义了发送命令前建立地址信号的时间。 0000 0000：1 个 HCLK 周期 0000 0001：2 个 HCLK 周期 …… 1111 1110：255 个 HCLK 周期 1111 1111：保留

位/域	名称	R/W	描述
15:8	WAITx	R/W	配置在属性空间 x 的等待时间（Attribute Memory x Wait Time Configure） 该位以 HCLK 为时钟周期，定义了保持命令的最小时间，定义的时间结束后，若等待信号有效低则命令保持时间变长。 0000 0000：保留 0000 0001：2 个 HCLK 周期（+由 NWAIT 信号变低引入的等待周期） 0000 0010：3 个 HCLK 周期 …… 1111 1110：255 个 HCLK 周期（+由 NWAIT 信号变低引入的等待周期） 1111 1111：保留
23:16	HLDx	R/W	配置在属性空间 x 的保持时间（Attribute Memory x Hold Time Configure） 该位以 HCLK 为时钟周期，定义了发送命令后，地址信号保持的时间。 0000 0000：保留 0000 0001：写访问 1 个 HCLK 周期，读访问 3 个 HCLK 周期 …… 1111 1110：写访问 254 个 HCLK 周期，读访问 256 个 HCLK 周期 1111 1111：保留
31:24	HIZx	R/W	配置在属性空间 x 数据总线的高阻时间（Attribute Memory x Databus Hiz Time Configure） 该位以 HCLK 为时钟周期，定义了数据总线高阻态的时间，仅对写操作生效。 0000 0000：0 个 HCLK 周期 0000 0001：1 个 HCLK 周期 …… 1111 1111：255 个 HCLK 周期

4.7.2.5 I/O 空间时序寄存器 4（SMC_IOSTIM4）

偏移地址：0xB0

复位值：0xFCFC FCFC

位/域	名称	R/W	描述
7:0	SET	R/W	配置在 I/O 空间 x 的建立时间（I/O x Setup Time Configure） 该位以 HCLK 为时钟周期，定义了发送命令前建立地址信号的时间。 0000 0000：1 个 HCLK 周期 0000 0001：2 个 HCLK 周期 …… 1111 1111：256 个 HCLK 周期

位/域	名称	R/W	描述
15:8	WAIT	R/W	配置在 I/O 空间 x 的等待时间（I/O x Wait Time Configure） 该位以 HCLK 为时钟周期，定义了保持命令的最小时间，定义的时间结束后，若等待信号有效低则命令保持时间变长。 0000 0000：保留 0000 0001：2 个 HCLK 周期（+由 NWAIT 信号变低引入的等待周期） 0000 0010：3 个 HCLK 周期 …… 1111 1111：256 个 HCLK 周期（+由卡的 NWAIT 信号变低引入的等待周期）
23:16	HLD	R/W	配置在 I/O 空间 x 的保持时间（I/O x Hold Time Configure） 该位以 HCLK 为时钟周期，定义了发送命令后，地址信号保持的时间。 0000 0000：保留 0000 0001：1 个 HCLK 周期 0000 0010：2 个 HCLK 周期 …… 1111 1111：255 个 HCLK 周期
31:24	HIZ	R/W	配置在 I/O 空间 x 数据总线的高阻时间（I/O x Databus Hiz Time Configure） 该位以 HCLK 为时钟周期，定义了数据总线高阻态的时间，仅对写操作生效。 0000 0000：0 个 HCLK 周期 0000 0001：1 个 HCLK 周期 …… 1111 1111：255 个 HCLK 周期

4.7.2.6 ECC 结果寄存器 2/3（SMC_ECCRS2/3）

偏移地址：0x54 + 0x20 * (x-1),x=2 或 3

复位值：0x0000 0000

位/域	名称	R/W	描述
31:0	ECCR S	R	ECC 结果

4.8 DMC 简介

DMC 一个动态存储控制器，外接片外 SDR-SDRAM。

4.9 DMC 主要特征

- 16 位数据宽度
- 最大支持 2MB 片外 SDR-SDRAM
- SDR-SDRAM 的时序和大小可配置
- 支持 SDR-SDRAM power-down 模式
- 支持 SDR-SDRAM auto-refresh 和 self-refresh 模式

4.10 DMC 结构框图

图 4 DMC 结构框图

4.11 DMC 功能描述

4.11.1 DMC 外部存储器接口

前缀“N”的信号表示低有效信号。

表格 24DMC 引脚（适用 APM32E103xCxE 非合封产品）

信号名称	输入/输出	管脚	功能
A0	输出	PF13	地址
A1	输出	PF14	地址
A2	输出	PF15	地址
A3	输出	PG0	地址
A4	输出	PE8	地址
A5	输出	PE9	地址
A6	输出	PE10	地址
A7	输出	PE11	地址
A8	输出	PE12	地址
A9	输出	PE13	地址
A10	输出	PF12	地址

信号名称	输入/输出	管脚	功能
D0	输入/输出	PG12	双向数据
D1	输入/输出	PG13	双向数据
D2	输入/输出	PG14	双向数据
D3	输入/输出	PG15	双向数据
D4	输入/输出	PE3	双向数据
D5	输入/输出	PE5	双向数据
D6	输入/输出	PE6	双向数据
D7	输入/输出	PF0	双向数据
D8	输入/输出	PC10	双向数据
D9	输入/输出	PC11	双向数据
D10	输入/输出	PD2	双向数据
D11	输入/输出	PD3	双向数据
D12	输入/输出	PD4	双向数据
D13	输入/输出	PD5	双向数据
D14	输入/输出	PD6	双向数据
D15	输入/输出	PG9	双向数据
BA	输出	PF11	Bank 地址
CLK	输出	PE15	时钟
CKE	输出	PB11	时钟使能
LDQM	输入	PF10	16 位数据写入
UNQM	输入	PB10	16 位数据读取
NWE	输出	PF6	写使能
NCAS	输出	PF5	列地址位选通命令
NRAS	输出	PF4	行地址位选通命令
NCS	输出	PF2	片选

表格 25 DMC 引脚（适用 APM32E103VET6S）

信号名称	输入/输出	管脚	功能
A0	输出	PF13	地址
A1	输出	PF14	地址
A2	输出	PF15	地址
A3	输出	PG0	地址
A4	输出	PE8	地址

信号名称	输入/输出	管脚	功能
A5	输出	PE9	地址
A6	输出	PE10	地址
A7	输出	PE11	地址
A8	输出	PE12	地址
A9	输出	PE13	地址
A10	输出	PF12	地址
D0	输入/输出	PG12	双向数据
D1	输入/输出	PG13	双向数据
D2	输入/输出	PG14	双向数据
D3	输入/输出	PG15	双向数据
D4	输入/输出	PE3	双向数据
D5	输入/输出	PE5	双向数据
D6	输入/输出	PE6	双向数据
D7	输入/输出	PF0	双向数据
D8	输入/输出	PC10	双向数据
D9	输入/输出	PC11	双向数据
D10	输入/输出	PD2	双向数据
D11	输入/输出	PD3	双向数据
D12	输入/输出	PD4	双向数据
D13	输入/输出	PD5	双向数据
D14	输入/输出	PD6	双向数据
D15	输入/输出	PG9	双向数据
BA	输出	PF11	Bank 地址
CKE	输出	PB11	时钟使能
CLK	输出	PE15	时钟
LDQM	输入	PF10	16 位数据写入
UNQM	输入	PB10	16 位数据读取
NWE	输出	PF6	写使能
NCAS	输出	PF5	列地址位选通命令
NRAS	输出	PF4	行地址位选通命令
NCS	输出	PF2	片选

4.11.2 DMC 配置举例

配置切换寄存器 DMC_SW 置 1 使能 DMC；

SDRAM 时钟频率最大值为 50MHZ；

配置寄存器 RCM_CFG 的位[29:28]来决定 SDRAM 的时钟频率；

例：512K×2BANKS×16BIT SDRAM

- 行地址对应引脚：A0-A10 共 11bits
- 列地址对应引脚：A0-A7 共 8bits
- Bank 地址对应引脚：BA0 1bit

配置寄存器 DMC_CFG 的行地址宽度位置 1010；

配置寄存器 DMC_CFG 的列地址宽度位置 0111；

配置寄存器 DMC_CFG 的 Bank 地址宽度位置 00。

4.12 DMC 寄存器地址映射

表格 26 DMC 寄存器地址映射

寄存器名	描述	偏移地址
DMC_CFG	配置寄存器	0x00
DMC_TIM0	时序寄存器 0	0x04
DMC_TIM1	时序寄存器 1	0x08
DMC_CTRL1	控制寄存器 1	0x0C
DMC_REF	刷新寄存器	0x10
DMC_CHIPSEL	片选寄存器	0x14
DMC_MASK	掩码寄存器	0x54
DMC_SW	切换寄存器	0x0400
DMC_CTRL2	控制寄存器 2	0x0404

4.13 DMC 寄存器功能描述

4.13.1 配置寄存器（DMC_CFG）

偏移地址：0x00

复位值：0x0014 1388

位/域	名称	R/W	描述
2:0	保留		
4:3	BAWCFG	R/W	配置 bank 地址宽度（Bank Address Width Configure） 00：bank 位数是 1 01：bank 位数是 2 其他保留

位/域	名称	R/W	描述
8:5	RAWCFG	R/W	配置行地址宽度（Row Address Width Configure） 0000-1001：保留 1010：行地址位数是 11 位 1011：行地址位数是 12 位 …… 1111：行地址位数是 16 位
12:9	CAWCFG	R/W	配置列地址宽度（Column Address Width Configure） 0000-0110：保留 0111：列地址位数是 8 位 1000：列地址位数是 9 位 …… 1110：列地址位数是 15 位 1111：保留
14:13	DWCFG	R/W	配置数据宽度（Data Width Configure） 00：SDRAM 数据位宽 16bits 其他保留
31:15	保留		

4.13.2 时序寄存器 0（DMC_TIM0）

偏移地址：0x04

复位值：0x019A 5252

位/域	名称	R/W	描述
1:0	CASLSEL0	R/W	选择 CAS 等待时间（CAS Latency Select） CAS = CASLSEL0+（ECASLSEL1<< 2） 00：1 个时钟周期 01：2 个时钟周期 10：3 个时钟周期 11：4 个时钟周期 其他：保留
5:2	RASMINTSEL	R/W	选择 RAS 最小时间（RAS Minimum Time Select） 这些位用来选择行激活和预充电的之间的最小时间。 0000：1 个时钟周期 0001：2 个时钟周期 …… 1111：16 个时钟周期
8:6	DTIMSEL	R/W	选择 RAS 到 CAS 的延迟时间（RAS To CAS Delay Time Select） 000：1 个时钟周期 001：2 个时钟周期 …… 111：8 个时钟周期
11:9	PCPSEL	R/W	选择预充电周期（Precharge Period Select） 000：1 个时钟周期 001：2 个时钟周期 …… 111：8 个时钟周期

位/域	名称	R/W	描述
13:12	WRTIMSEL	R/W	对于写操作，选择最后一个数据到下一次预充电的时间（Select Time Between The Last Data And The Next Precharge For Write） 00：1 个时钟周期 01：2 个时钟周期 10：3 个时钟周期 11：4 个时钟周期
17:14	ARPSEL	R/W	选择自动刷新周期（Auto-Refresh Period Select） 这些位用来定义两次自动刷新命令间的最小时间间隔。 0000：1 个时钟周期 0001：2 个时钟周期 …… 1111：16 个时钟周期
21:18	XSR0	R/W	退出自刷新切换至激活命令或自动刷新读命令之间的最小间隔时间 XSR = XSR0 +（EXSR1 << 4）. XSR = 0 ~ 511 对应 1 ~ 512 SDCLK 时钟周期
25:22	ATACP	R/W	Active to active 命令周期（Active to Active Command Cycle） 0000：1 个时钟周期 0001：2 个时钟周期 …… 1111：16 个时钟周期
26	ECASLSEL1	R/W	扩展 CAS 等待时间（Extended CAS Latency） 见 TIM0_CASLSEL0 解释
31:27	EXSR1	R/W	退出自刷新切换至激活命令或自动刷新读命令之间的最小间隔时间 见 TIM0_XSR0 解释

4.13.3 时序寄存器 1（DMC_TIM1）

偏移地址：0x08

复位值：0x0007 4E20

位/域	名称	R/W	描述
15:0	STBTIM	R/W	稳定时间（Stable Time） SDRAM 上电后，经过多少个时钟周期后才接收其他命令。
19:16	ARNUMCFG	R/W	配置初始化过程中自动刷新数量（Number Of Auto-Refresh During Initialization Configure） 0000：1 个自动刷新 0001：2 个自动刷新 …… 1111：16 个自动刷新
31:20	保留		

4.13.4 控制寄存器 1（DMC_CTRL1）

偏移地址：0x0C

复位值：0x0000 3048

位/域	名称	R/W	描述
0	INIT	R/W	初始化 SDRAM（SDRAM Initialize） 0：无效 1：初始化 SDRAM，SDRAM 初始化完后，硬件自动置 0 该位。
1	SRMEN	R/W	将 SDRAM 设置为自刷新模式（Put SDRM in Self-Refresh Mode Enable） 0：无效 1：使能
2	PDMEN	R/W	使能将 SDRAM 设置为掉电模式（Put SDRM in Power-Down Mode Enable） 0：禁止 1：使能
3	PCACFG	R/W	配置预充电方式（Precharge Algorithm Configure） 0：读取操作完毕后立即行预充电。 1：读取操作完毕后延迟一段时候后才对行预充电。
4	FRBSREN	R/W	进入自刷新模式前全刷新（Full Refresh Before Entering Self-Refresh Mode） 0：进入自刷新模式前仅仅刷新一行 1：进入自刷新模式前刷新所有行
5	FRASREN	R/W	退出自刷新模式前全刷新（Full Refresh After Exit Self-Refresh Mode） 0：退出自刷新模式后仅仅刷新一行 1：退出自刷新模式后刷新所有行
8:6	RDNUMCFG	R/W	配置读取路径中插入寄存器的数量（Configure Number Of Registers Inserted In Read Data Path） 000：0 个寄存器 001：1 个寄存器 …… 111：7 个寄存器
9	MODESET	R/W	模式设置（Mode Setup） 1：更新 SDRAM 模式寄存器。完成模式寄存器更新后，硬件自动清零该位
10	保留		
11	SRMFLG	R	自刷新模式标志（Self-refresh mode flag） 1：指示当前 SDRAM 储于自刷新模式中，当 FRBSREN 位置 1 时此位有效。
15:12	BANKNUMCFG	R/W	配置打开 Bank 数量（Number Of Open Banks Configure） SDRAM 内部 Bank 被打开的数量。 0000：1 个 bank 0001：2 个 bank …… 1111：16 个 bank
31:16	保留		

4.13.5 刷新寄存器（DMC_REF）

偏移地址：0x10

复位值：0x0000 00C3

位/域	名称	R/W	描述
15:0	RCYCCFG	R/W	配置刷新周期（Refresh Cycle Configure） 指定连续两次刷新间隔多少个时钟周期。
31:16	保留		

4.13.6 片选寄存器（DMC_CHIPSEL）

偏移地址：0x14

复位值：0x0000 0000

位/域	名称	R/W	描述
15:0	保留		
31:16	BACHIPSEL	R/W	片选的基地址（Base Address For Chip Select） 默认是 0。

4.13.7 掩码寄存器（DMC_MASK）

偏移地址：0x54

复位值：0x0000 000D

位/域	名称	R/W	描述
4:0	MSIZESEL	R/W	选择 memory 空间大小（Memory Size Select） 0000：无 memory 连接 0001：64KB 0010：128KB 0011：256KB 0100：512KB 0101：1MB 0110：2MB 0111：4MB 1000：8MB 1001：16MB 1010：32MB 1011：64MB 1100：128MB 1101：256MB 其他：保留
7:5	MTYPESEL	R/W	选择存储器类型（Memory Type Select） 000：SDRAM 其他：保留
31:8	保留		

4.13.8 切换寄存器（DMC_SW）

偏移地址：0x0400

复位值：0x0000 0000

位/域	名称	R/W	描述
0	MCSW	R/W	切换存储控制器功能（Memory Controller Function Switch） 0：选择 SMC 1：选择 DMC

位/域	名称	R/W	描述
31:1	保留		

4.13.9 控制寄存器 2（DMC_CTRL2）

偏移地址：0x0404

复位值：0x0000 002E

位/域	名称	R/W	描述
0	CPHACFG	R/W	配置时钟相位（Clock Phase Configure） 0：系统时钟不反向 1：系统时钟反向
1	RDDEN	R/W	使能 RD 延时（RD Delay Function Enable） 0：使能 1：禁止
4:2	RDDCFG	R/W	配置 RD 延时时钟（RD Clock Delay Configure） 000：0 个系统时钟 001：1 个系统时钟 …… 111：7 个系统时钟
5	WPEN	R/W	使能写管道（Write Pipe Enable） 0：禁止 1：使能
6	BUFFEN	R/W	使能 DMC 加速模块（DMC Accelerate Module Enable） 通过缓冲区开启，回环突发数据传输，提高 SDRAM 读性能。
7	WRPBSEL	R/W	选择回环突发类型（WRAP Burst Type Selection） 0：回环突发长度为 4 节拍传输 1：回环突发长度为 8 节拍传输
31:8	保留		

5 复位与时钟管理（RCM）

5.1 术语全称、缩写描述

表格 27 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	HSICLK
低速内部时钟	Low Speed Internal Clock	LSICLK
锁相环	Phase Locked Loop	PLL
主要时钟输出	Main Clock Output	MCO
校准	Calibrate	CAL
调整	Trim	TRIM
唤醒	Wakeup	WUP
自动唤醒	Automatic Wakeup	AWUP
备份	Backup	BAKP
低功耗	Low Power	LPWR
时钟安全系统	Clock Security System	CSS
不可屏蔽中断	Non Maskable Interrupt	NMI

5.2 复位功能描述

支持的复位分为三类：系统复位、电源复位、备份区域复位三种复位形式。

5.2.1 系统复位

5.2.1.1 “系统复位”复位源

复位源分为外部复位源、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
- 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（控制/状态寄存器）的复位标志位和备份区域中的寄存器以外的所有寄存器复位到复位状态。

软件复位

将 Arm® Cortex®-M3 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

低功耗管理复位的产生有两种情况，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”。

5.2.1.2 “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 5 “系统复位”复位电路

5.2.2 电源复位

“电源复位”复位源

“电源复位”复位源如下：

- 上电复位（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备份区域外的所有寄存器。

5.2.3 备份域复位

“备份域复位”复位源

“备份域复位”复位源如下：

- 软件复位，设置 RCM_BDCTRL（备份域控制寄存器）中的 BDRST 位
- 在 V_{DD} 和 V_{BAT} 掉电时，V_{DD} 或 V_{BAT} 上电

以上任一事件发生时，产生备份域复位。

备份区域复位拥有两个专门的复位，它们只影响备份区域。

5.3 时钟管理功能描述

整个系统的时钟源有：HSECLK、LSECLK、HSICLK、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。

5.3.1 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 6 HSECLK/LSECLK 的时钟源硬件配置

时钟源	硬件配置
外部时钟	
晶体/陶瓷谐振器	

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。负载电容（C_{L1}、C_{L2}）的值必须根据所选择的振荡器来调整。

5.3.1.1 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 28 产生 HSECLK 的时钟源

名称	说明
外部时钟源 （HSECLK 旁路）	通过 OSC_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、其它信号发生器产生；波形可以是 50%占空比的方波、正弦波或三角波，最高频率可达 25MHz。 硬件连接上，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_CTRL 中的 HSEBCFG 和 HSEEN 位来选择这一模式。

名称	说明
外部晶体/陶瓷谐振器（HSECLK 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。 频率范围是 4-16MHz。 需要 OSC_IN、OSC_OUT 连接谐振器，可以通过设置 RCM_CTRL 中的 HSEEN 位，启动和关闭。 在时钟控制寄存器 RCM_CTRL 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。

5.3.1.2 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 29 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、其它信号发生器产生；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，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_BDCTRL 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_BDCTRL 里的 LSEEN 位启动和关闭。 在 RCM_BDCTRL 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。

5.3.2 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）。

5.3.2.1 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℃、V_{DD}=V_{DDA}=3.3V），系统复位时，工厂校准值被装载到 RCM_CTRL 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_CTRL 中的 HSICLKTRM 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 可由 RCM_CTRL 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。

5.3.2.2 LSICLK 低速内部时钟信号

LSICLK 主要特征

LSICLK 由 RC 振荡器产生。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，为独立看门狗和自动唤醒单元提供时钟。

LSICLK 可以通过 RCM_CSTS 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 里被允许，将产生 LSICLK 中断申请。

LSICLK 校准

注意：只有大容量产品有 TMR5 才能进行 LSICLK 校准

校准 LSICLK 振荡器的目的是为了补偿其频率偏移，校准后能够得到一定精度的 RTC 时钟基数以及 IWDT 的超时时间。

首先，将 TMR5 的通道 4 设置为输入捕获模式，连接 LSICLK 时钟并测量其时钟频率，然后以 HSECLK 时钟频率为精度保证，根据需要的 RTC 时间基数和 IWDT 的超过时间，设置 RTC 的 20 位预分频器。

5.3.3 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。

可选择 HSICLK/2 或 HSECLK 作为 PLL 输入时钟源，经过 PLL 倍频（可选择倍频因子）后输出 PLLCLK。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。

5.3.4 时钟树

图 7 APM32E103xCxE 时钟树

注意：

（1）HCLK 是指 AHB 时钟。

（2）PCLK1，PCLK2 分别是指连接 APB1，APB2 的外设的时钟信号。

（3）FCLK 是 Arm® Cortex®-M3 的运行时钟。

（4）可通过多个预分频器配置 AHB、APB2（高速 APB）和 APB1（低速 APB）域的频率。其中，AHB 和 APB2 域的最大频率是 120MHz，而 APB1 域的最大允许频率是 60MHz。

（5）当使用 USBD 时，PLL 必须被设置为输出 48/72/96/120MHz 时钟，经过 USBD 分频器分频，用于提供 48MHz 的 USBDCLK 时钟。

（6） 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。

（7）SysTick（系统定时器）可由 HCLK8 分频后的时钟信号提供，通过对 SysTick 控制与状态寄存器的设置，可选择不同时钟源。

（8）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。

（9）其中，TMRx（x=2,3,4,5,6,7）时钟信号通过 APB1 分频，TMRx（x=1,8）时钟信号通过 APB2 分频。

（10） 只有型号为 APM32E103VET6S 的产品才有合封 SDRAM，但没有 SDIO。

5.3.5 RTC 的时钟源选择

通过设置 RCM_BDCTRL 里的 RTCSRCSEL 位可选择 HSECLK/128、LSECLK 或 LSICLK 作为 RTCCLK 时钟源。只有备份域复位时，时钟源的选择可以改变。

由于 LSECLK 在备份域里，HSECLK、LSICLK 不在备份域内，因此，选择不同的时钟源，RTC 的工作情况存在差异，具体情况见下表：

表格 30 RTC 选择不同时钟源时 RTC 的工作情况

时钟源	工作情况
LSECLK 被选为 RTC 时钟	只要 V_{BAT} 维持供电，尽管 V_{DD} 供电被切断，RTC 仍继续工作
LSICLK 被选为自动唤醒单元时钟	如果 V_{DD} 供电被切断，AWUP 状态不能被保证。
HSECLK/128 作为 RTC 时钟	如果 V_{DD} 供电被切断或内部电压调压器被关闭（1.3V 域的供电被切断），则 RTC 状态不确定，故必须设置 PMU_CTRL（电源控制寄存器）的 BPWEN 位（取消后备区域的写保护）为“1”。

5.3.6 IWDT 的时钟源选择

当 IWDT（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 IWDT。LSICLK 在被强制打开后，将一直处于打开状态，不能被关闭。

5.3.7 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_CFG（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 MCOSEL 位说明。

5.3.8 SYSCLK 的时钟源选择

当进行系统复位后，HSICLK 振荡器被选作系统时钟，且不能被停止。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）。目标时钟源可以是 HSECLK、PLLCLK，PLLCLK 的时钟源可以是 HSECLK、HSICLK/2。

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

5.3.9 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。

注意：CSS 被软件激活并且 HSECLK 时钟出现故障时，会产生 CSS 中断和 NMI（不可屏蔽中断）。由于 CSS 中断被清除前 NMI 不断被执行，所以需要设置 RCM_INT 寄存器里的 CSSIF 位来清除中断。

5.4 寄存器地址映射

表格 31 RCM 寄存器地址映射

寄存器名	描述	偏移地址
RCM_CTRL	时钟控制寄存器	0x00
RCM_CFG	时钟配置寄存器	0x04
RCM_INT	时钟中断寄存器	0x08
RCM_APB2RST	APB2 外设复位寄存器	0x0C
RCM_APB1RST	APB1 外设复位寄存器	0x10
RCM_AHBCLKEN	AHB 外设时钟使能寄存器	0x14
RCM_APB2CLKEN	APB2 外设时钟使能寄存器	0x18
RCM_APB1CLKEN	APB1 外设时钟使能寄存器	0x1C
RCM_BDCTRL	备份域控制寄存器	0x20
RCM_CSTS	控制/状态寄存器	0x24

5.5 寄存器功能描述

5.5.1 时钟控制寄存器（RCM_CTRL）

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	使能高速内部时钟（High Speed Internal Clock Enable） 由软件置 1 或清 0。 HSICLK 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 HSICLK 作为系统时钟源或通过 PLL 提供系统时钟时，该位不能清 0。 0：关闭 HSICLK RC 振荡器 1：开启 HSICLK RC 振荡器

位/域	名称	R/W	描述
1	HSIRDYFLG	R	高速内部时钟就绪标志（High Speed Internal Clock Ready Flag） 0：HSICLK RC 振荡器未稳定 1：HSICLK RC 振荡器已稳定
2	保留		
7:3	HSITRM	R/W	调整高速内部时钟（High Speed Internal Clock Trim） 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 调整 HSICLK RC 振荡器的频率。
15:8	HSICAL	R	校准内部高速时钟（High Speed Internal Clock Calibrate） 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。
16	HSEEN	R/W	使能高速外部时钟（High Speed External Clock Enable） 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。 0：HSECLK 关闭 1：HSECLK 开启
17	HSERDYFLG	R	高速外部时钟就绪标志（High Speed External Clock Ready Flag） HSECLK 稳定后有硬件置 1，软件清 0。 0：HSECLK 未稳定 1：HSECLK 已稳定
18	HSEBCFG	R/W	配置高速外部时钟旁路模式（High Speed External Clock Bypass Configure） 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。 0：非旁路模式 1：旁路模式
19	CSSEN	R/W	使能时钟安全系统（Clock Security System Enable） 0：禁止 1：使能
23:20	保留		
24	PLLEN	R/W	使能 PLL（PLL Enable） 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。 0：PLL 关闭 1：PLL 使能
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		

5.5.2 时钟配置寄存器（RCM_CFG）

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） 选择系统时钟源。 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 HSE 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：不可用
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。 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 注意：当 AHB 时钟的预分频系数大于 1 时，必须开启预取缓冲器。
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。 警告：软件必须保证 APB1 时钟频率不超过 60MHz。 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
13:11	APB2PSC	R/W	配置 APB2 时钟预分频系数（APB1 Clock Prescaler Factor） 控制高速 APB2 时钟（PCLK2）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
15:14	ADCPSC	R/W	配置 ADC 时钟预分频系数（ADC 时钟 Prescaler Factor Configure）

位/域	名称	R/W	描述
			确定 ADC 时钟频率 00：PCLK2 2 分频后作为 ADCCLK 01：PCLK2 4 分频后作为 ADCCLK 10：PCLK2 6 分频后作为 ADCCLK 11：PCLK2 8 分频后作为 ADCCLK
16	PLLSRCSEL	R/W	选择 PLL 时钟源（PLL Clock Source Select） 选择 PLL 输入时钟源。 0：HSICLK RC 振荡器时钟经 2 分频后作为 PLL 输入时钟 1：HSECLK 作为 PLL 输入时钟 注：只能在关闭 PLL 时才能写入此位。
17	PLLHSEPSC	R/W	配置作为 PLL 时钟源的 HSECLK 分频（HSECLK Prescaler Factor for PLL Clock Source） 分频 HSECLK 后作为 PLL 输入时钟。 0：HSECLK 不分频 1：HSECLK2 分频 注：只能在关闭 PLL 时才能写入此位。
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：PLLCLK2 倍频输出 0001：PLLCLK 3 倍频输出 0010：PLLCLK 4 倍频输出 0011：PLLCLK 5 倍频输出 0100：PLLCLK 6 倍频输出 0101：PLLCLK 7 倍频输出 0110：PLLCLK 8 倍频输出 0111：PLLCLK 9 倍频输出 1000：PLLCLK 10 倍频输出 1001：PLLCLK 11 倍频输出 1010：PLLCLK 12 倍频输出 1011：PLLCLK 13 倍频输出 1100：PLLCLK 14 倍频输出 1101：PLLCLK 15 倍频输出 1110：PLLCLK 16 倍频输出 1111：PLLCLK 16 倍频输出
23:22	USBDPSC	R/W	配置 USBD 预分频系数（USBD Prescaler Factor Configure） 产生 48MHz 的 USBD 时钟。 在 RCM_APB1CLKEN 寄存器中使能 USBD 时钟之前，必须保证该位已经有效。如果 USBD 时钟被使能，该位不能被清零。 0：PLLCLK1.5 倍分频作为 USBD 时钟 1：PLLCLK 直接作为 USBD 时钟 2：PLLCLK 2 分频作为 USBD 时钟 3：PLLCLK 2.5 分频作为 USBD 时钟
26:24	MCOSEL	R/W	选择主时钟输出（Main Clock Output Select） 由软件置 1 或清零。 0xx：没有时钟输出 100：系统时钟（SYSCLK）输出 101：内部 RC 振荡器时钟（HSICLK）输出

位/域	名称	R/W	描述
			110：外部振荡器时钟（HSECLK）输出 111：PLLCLK 2 分频后输出 注意：1.该时钟输出在启动和切换 MCO 时钟源时可能会被截断。 2.在系统时钟作为输出至 MCO 引脚时，请保证输出时钟频率不超过 50MHz（I/O 口最高频率）。
27	FPUPSC	R/W	配置 FPU 预分频系数（FPU Clock Prescaler Factor Configure） 0：HCLK 作为 FPU 时钟 1：HCLK 2 分频作为 FPU 时钟
29:28	SDRAMPSC	R/W	配置 SDRAM 时钟预分频系数（SDRAM Clock Prescaler Configure） 00：DMC 时钟作为 SDRAM 时钟 01：DMC 时钟 2 分频后作为 SDRAM 时钟 其他：DMC 时钟 4 分频后作为 SDRAM 时钟 注意：SDRAM 的最高时钟频率为 50MHz。
31:30	保留		

5.5.3 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	LSICLK 就绪中断标志（LSICLK Ready Interrupt Flag） 在内部低速时钟就绪且 LSIRDYEN 位被置 1 时，由硬件置 1。 由软件通过置 1LSIRDYCLR 来清 0 该位。 0：无 LSICLK 就绪中断 1：发生 LSICLK 就绪中断
1	LSERDYFLG	R	LSECLK 就绪中断标志（LSECLK Ready Interrupt Flag） 在外部低速时钟就绪且 LSERDYEN 位被置 1 时，由硬件置 1。 由软件通过置 1LSERDYCLR 来清 0 该位。 0：无 LSECLK 就绪中断 1：发生 LSECLK 就绪中断
2	HSIRDYFLG	R	HSICLK 就绪中断标志（HSICLK Ready Interrupt Flag） 在内部高速时钟就绪且 HSIRDYEN 位被置 1 时，由硬件置 1。 由软件通过置 1HSIRDYCLR 来清 0 该位。 0：无 HSICLK 就绪中断 1：发生 HSICLK 就绪中断
3	HSERDYFLG	R	HSECLK 就绪中断标志（HSECLK Ready Interrupt Flag） 在外部低速时钟就绪且 HSERDYCLR 位被置 1 时，由硬件置 1。 由软件通过置 1 HSERDYCLR 来清 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） 在 PLL 就绪且 PLLRDYEN 位被置 1 时，由硬件置 1。 由软件通过置 1 PLLRDYCLR 来清 0 该位。 0：无 PLL 上锁产生的时钟就绪中断 1：PLL 上锁导致时钟就绪中断

位/域	名称	R/W	描述
6:5	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。 由软件通过置位 CSSCLR 位来清 0 该位。 0：无 HSECLK 失效产生的安全系统中断 1：HSECLK 失效导致了时钟安全系统中断
8	LSIRDYEN	R/W	使能 LSICLK 就绪中断（LSICLK 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
10	HSIRDYEN	R/W	HSICLK 就绪中断使能（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	HSCLKE 就绪中断使能（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	PLL 就绪中断使能（PLL Ready Interrupt Enable） 使能 PLL 就绪中断。 0：禁止 1：使能
15:13	保留		
16	LSIRDYCLR	W	清除 LSICLK 就绪中断（LSICLK Ready Interrupt Clear） 清除 LSICLK 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 LSECLK 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 HSICLK 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 HSECLK 就绪中断标志位 HSERDYFLG。 0：无作用 1：清除

位/域	名称	R/W	描述
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
22:21	保留		
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		

5.5.4 APB2 外设复位寄存器（RCM_APB2RST）

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	AFIORST	R/W	复位复用功能 IO（Alternate Function I/O Reset） 0：无作用 1：复位
1	保留		
2	PARST	R/W	复位 IO 端口 A（IO Port A Reset） 0：无作用 1：复位
3	PBRST	R/W	复位 IO 端口 B（IO Port B Reset） 0：无作用 1：复位
4	PCRST	R/W	复位 IO 端口 C（IO Port C Reset） 0：无作用 1：复位
5	PDRST	R/W	复位 IO 端口 D（IO Port D Reset） 0：无作用 1：复位
6	PERST	R/W	复位 IO 端口 E（IO Port E Reset） 0：无作用 1：复位
7	PFRST	R/W	复位 IO 端口 F（IO Port F Reset） 0：无作用 1：复位
8	PGRST	R/W	复位 IO 端口 G（IO Port G Reset） 0：无作用 1：复位

位/域	名称	R/W	描述
9	ADC1RST	R/W	复位 ADC1（ADC1 Reset） 0：无作用 1：复位 ADC1
10	ADC2RST	R/W	复位 ADC2（ADC2 Reset） 0：无作用 1：复位
11	TMR1RST	R/W	复位 TMR1 定时器（TMR1 Timer Reset） 0：无作用 1：复位
12	SPI1RST	R/W	复位 SPI1（SPI1 Reset） 0：无作用 1：复位
13	TMR8RST	R/W	复位 TMR8 定时器（TMR8 Timer Reset） 0：无作用 1：复位
14	USART1RST	R/W	复位 USART1（USART1 Rreset） 0：无作用 1：复位
15	ADC3RST	R/W	复位 ADC3（ADC3 Reset） 0：无作用 1：复位
31:16	保留		

5.5.5 APB1 外设复位寄存器（RCM_APB1RST）

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
2	TMR4RST	R/W	复位定时器 4（Timer 4Reset） 由软件置 1 或清 0 0：无作用 1：复位
3	TMR5RST	R/W	复位定时器 5（Timer5 Reset） 0：无作用 1：复位
4	TMR6RST	R/W	复位定时器 6（Timer 6 Reset） 0：无作用 1：复位

位/域	名称	R/W	描述
5	TMR7RST	R/W	复位定时器 7（Timer 7 Reset） 0：无作用 1：复位
10:6	保留		
11	WWDTRST	R/W	复位窗口看门狗（Window Watchdog Reset） 0：无作用 1：复位
13:12	保留		
14	SPI2RST	R/W	复位 SPI2（SPI2 Reset） 0：无作用 1：复位
15	SPI3RST	R/W	复位 SPI3（SPI3 Reset） 0：无作用 1：复位
16	保留		
17	USART2RST	R/W	复位 USART2（USART2 Reset） 0：无作用 1：复位
18	USART3RST	R/W	复位 USART3（USART3 Reset） 0：无作用 1：复位
19	UART4RST	R/W	复位 UART4（UART4 Reset） 0：无作用 1：复位
20	UART5RST	R/W	复位 UART5（UART5 Reset） 0：无作用 1：复位
21	I2C1RST	R/W	复位 I2C1（I2C1 Reset） 0：无作用 1：复位
22	I2C2RST	R/W	复位 I2C2（I2C2 Reset） 0：无作用 1：复位
23	USBDRST	R/W	复位 USBD1/2（USBD1/2 Reset） 0：无作用 1：复位
24	保留		
25	CAN1RST	R/W	复位 CAN1（CAN1 Reset） 0：无作用 1：复位
26	CAN2RST	R/W	复位 CAN2（CAN2 Reset） 0：无作用 1：复位

位/域	名称	R/W	描述
27	BAKPRST	R/W	复位备份接口（Backup Interface Reset） 0：无作用 1：复位
28	PMURST	R/W	复位电源接口（Power Interface Reset） 0：无作用 1：复位
29	DACRST	R/W	复位 DAC（DAC Reset） 0：无作用 1：复位
31:30	保留		

5.5.6 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	DMA1EN	R/W	使能 DMA1 时钟（DMA1 Clock Enable） 0：禁止 1：使能
1	DMA2EN	R/W	使能 DMA2 时钟（DMA2 Clock Enable） 0：禁止 1：使能
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	FPUEN	R/W	使能 FPU 时钟（FPU Clock Enable） 0：禁止 1：使能
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
7	保留		
8	EMMCEN	R/W	使能 EMMC 时钟（EMMC Clock Enable） 0：禁止 1：使能

位/域	名称	R/W	描述
9	保留		
10	SDIOEN	R/W	使能 SDIO 时钟（SDIO Clock Enable） 0：禁止 1：使能
31:11	保留		

5.5.7 APB2 外设时钟使能寄存器（RCM_APB2CLKEN）

偏移地址：0x18

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 APB2 总线上的外设被访问时，将插入等待状态直到 APB2 的外设访问结束。

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	AFIOEN	R/W	使能复用功能 IO 时钟（Alternate Function I/O Clock Enable） 0：禁止 1：使能
1	保留		
2	PAEN	R/W	使能 IO 端口 A 时钟（I/O Port A Clock Enable） 0：禁止 1：使能
3	PBEN	R/W	使能 IO 端口 B 时钟（I/O Port B Clock Enable） 0：禁止 1：使能
4	PCEN	R/W	使能 IO 端口 C 时钟（I/O Port C Clock Enable） 0：禁止 1：使能
5	PDEN	R/W	使能 IO 端口 D 时钟（I/O Port D Clock Enable） 0：禁止 1：使能
6	PEEN	R/W	使能 IO 端口 E 时钟（I/O Port E Clock Enable） 0：禁止 1：使能
7	PFEN	R/W	使能 IO 端口 F 时钟（I/O Port F Clock Enable） 0：禁止 1：使能
8	PGEN	R/W	使能 IO 端口 G 时钟（I/O Port G Clock Enable） 0：禁止 1：使能

位/域	名称	R/W	描述
9	ADC1EN	R/W	使能 ADC1 接口时钟（ADC 1 Interface Clock Enable） 0：禁止 1：使能
10	ADC2EN	R/W	使能 ADC2 接口时钟（ADC 2 Interface Clock Enable） 0：禁止 1：使能
11	TMR1EN	R/W	使能 TMR1 定时器时钟（TMR1 Timer Clock Enable） 0：禁止 1：使能
12	SPI1EN	R/W	使能 SPI1 时钟（SPI 1 Clock Enable） 0：禁止 1：使能
13	TMR8EN	R/W	使能 TMR8 定时器时钟（TMR8 Timer Clock Enable） 0：禁止 1：使能
14	USART1EN	R/W	使能 USART1 时钟（USART1 Clock Enable） 0：禁止 1：使能
15	ADC3EN	R/W	使能 ADC3 接口时钟（ADC 3 Interface Clock Enable） 0：禁止 1：使能
31:16	保留		

5.5.8 APB1 外设时钟使能寄存器（RCM_APB1CLKEN）

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 APB1 总线上的外设被访问时，将插入等待状态直到 APB1 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
2	TMR4EN	R/W	使能定时器 4 时钟（Timer 4 Clock Enable） 0：禁止 1：使能

位/域	名称	R/W	描述
3	TMR5EN	R/W	使能定时器 5 时钟（Timer 5 Clock Enable） 0：禁止 1：使能
4	TMR6EN	R/W	使能定时器 6 时钟（Timer 6 Clock Enable） 0：禁止 1：使能
5	TMR7EN	R/W	使能定时器 7 时钟（Timer74 Clock Enable） 0：禁止 1：使能
10:6	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
13:12	保留		
14	SPI2EN	R/W	使能 SPI2 时钟（SPI 2 Clock Enable） 0：禁止 1：使能
15	SPI3EN	R/W	使能 SPI3 时钟（SPI 3 Clock Enable） 0：禁止 1：使能
16	保留		
17	USART2EN	R/W	使能 USART2 时钟（USART 2 Clock Enable） 0：禁止 1：使能
18	USART3EN	R/W	使能 USART3 时钟（USART 3 Clock Enable） 0：禁止 1：使能
19	UART4EN	R/W	使能 UART4 时钟（UART 4 Clock Enable） 0：禁止 1：使能
20	UART5EN	R/W	使能 UART5 时钟（UART 5 Clock Enable） 0：禁止 1：使能
21	I2C1EN	R/W	使能 I2C1/3 时钟（I2C1/3 Clock Enable） 0：禁止 1：使能
22	I2C2EN	R/W	使能 I2C2/4 时钟（I2C2/4 Clock Enable） 0：禁止 1：使能
23	USBDEN	R/W	使能 USBD1/2 时钟（USBD1/2 Clock Enable） 0：禁止 1：使能
24	保留		

位/域	名称	R/W	描述
25	CAN1EN	R/W	使能 CAN1 时钟（CAN1 Clock Enable） 0：禁止 1：使能
26	CAN2EN	R/W	使能 CAN2 时钟（CAN2 Clock Enable） 0：禁止 1：使能
27	BAKPEN	R/W	使能备份接口时钟（Backup Interface Clock Enable） 0：禁止 1：使能
28	PMUEN	R/W	使能电源接口时钟（Power Interface Clock Enable） 0：禁止 1：使能
29	DACEN	R/W	使能 DAC 接口时钟（DAC Interface Clock Enable） 0：禁止 1：使能
31:30	保留		

5.5.9 备份域控制寄存器（RCM_BDCTRL）

偏移地址：0x20

复位值：0x0000 0000，只能由备份域复位有效复位

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 Clock Enable） 0：禁止 1：使能
1	LSERDYFLG	R	LSECLK 就绪标志（Low-Speed External Oscillator Oscillator Ready Flag） LSECLK 稳定时硬件置 1，不稳定时硬件清 0。 0：未就绪 1：外就绪
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
7:3	保留		
9:8	RTCSRCSEL	R/W	选择 RTC 时钟源（RTC Clock Source Select） 先设置 BDRST 位复位备份域，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 振荡器作为 RTC 时钟

位/域	名称	R/W	描述
			10：LSICLK 振荡器作为 RTC 时钟 11：HSECLK 振荡器在 128 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	BDRST	R/W	复位备份域软件（Backup Domain Software Reset） 由软件置 1 或清 0 0：复位未激活 1：复位整个备份域
31:17	保留		

5.5.10 控制/状态寄存器（RCM_CSTS）

偏移地址：0x24

复位值：0x0C00 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
23:2	保留		
24	RSTFLGCLR	R/W	清除复位标志（Reset Flag Clear） 由软件置 1 来清除复位标志，包括 RSTFLGCLR。 0：无作用 1：清除复位标志
25	保留		
26	NRSTFLG	R/W	发生 NRST 引脚复位标志（NRST PIN Reset Occur Flag） 在 NRST 引脚复位发生时由硬件置 1；由软件通过写 RSTFLGCLR 位清除。 0：无 NRST 引脚复位发生 1：发生 NRST 引脚复位
27	PODRSTFLG	R/W	发生上电/掉电复位标志（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

位/域	名称	R/W	描述
28	SWRSTFLG	R/W	发生软件复位标志（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/W	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
30	WWDTRSTFLG	R/W	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/W	发生低功耗复位标志（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 电源管理单元（PMU）

6.1 术语全称、缩写描述

表格 32 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
电源电压检测器	Power Voltage Detector	PVD

6.2 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 1.3V 的电源，若主电源 V_{DD} 掉电，则通过 V_{BAT} 给后备供电区域供电。

6.3 结构框图

图 8 电源控制结构框图

6.4 功能描述

6.4.1 电源域

产品的电源域有：V_{DD} 电源域、V_{DDA} 电源域、1.3V 电源域、备份电源域。

6.4.1.1 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 1.3V 电源域供电，有以下几种工作模式：

- 正常模式：此模式下 1.3V 供电区域全功率运行
- 停止模式：此模式下 1.3V 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 1.3V 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

6.4.1.2 V_{DDA} 电源域

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

独立 ADC 电源和参考电压

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
- V_{REF+}/V_{REF-}：ADC 参考电压引脚

6.4.1.3 1.3V 电源域

由电压调节器供电，给内核、Flash、SRAM、数字外设供电。

6.4.1.4 备份电源域

当 V_{DD} 存在时，后备供电区域由 V_{DD} 供电，当 V_{DD} 掉电时，后备供电区域由 V_{BAT} 供电，用来保存后备寄存器的内容和维持 RTC 功能。给 LSECLK 晶体谐振器、RTC、备份寄存器以及 RCM_BDCTRL 寄存器、PC13、PC14、PC15 供电。

6.4.2 电源管理

6.4.2.1 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 9 上电复位和掉电复位的波形图

6.4.2.2 电源电压监测器（PVD）

PVD 可设置一阈值，可监测 V_{DD}/V_{DDA} 高于还是低于该阈值。如果使能中断，可触发中断，提前处理 V_{DD}/V_{DDA} 超过阈值的情况。PVD 的使用方法如下：

（1） 配置寄存器 PMU_CTRL 的 PVDEN 位置 1 使能 PVD

（2） 配置寄存器 PMU_CTRL 的 PLSEL[2:0]位选择 PVD 的电压阈值

（3） 配置寄存器 PMU_CSTS 的 PVDOFLG 位表明 V_{DD} 的值是高于还是低于 PVD 的阈值

（4） 当检测到 V_{DD}/V_{DDA} 低于或高于 PVD 阈值时，将产生 PVD 中断

PVD 的阈值波形监下图。PVD 阈值、迟滞电压请参考“数据手册”。

图 10 PVD 阈值波形图

6.4.3 功耗控制

6.4.3.1 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 33“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 1.3V 区域时钟的影响	对 V_{DD} 区域时钟的影响
睡眠	Arm® Cortex®-M3 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 1.3V 区域的时钟	HSICLK 和 HSECLK 的振荡器关闭
待机	1.3V 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	关		

模式	说明	进入方式	唤醒方式	电压调节器	对 1.3V 区域时钟的影响	对 V_{DD} 区域时钟的影响
			的外部复位、IWDT 复位			

睡眠模式

睡眠模式特点见下表：

表格 34 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 SLEEPONEINT 置 0，且执行 WFI 或 WFE 指令，立即进入睡眠模式；当 SLEEPONEINT 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 35 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 36 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。

特性	说明
唤醒	通过 WKUP 引脚的上升沿，RTC 闹钟、唤醒、入侵事件或 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。

6.4.3.2 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。

6.5 寄存器地址映射

表格 37 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

6.6 寄存器功能描述

6.6.1 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 Deepsleep 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
4	PVDEN	R/W	使能电源电压监测器（Power Voltage Detector Enable） 0：禁止 1：使能

位/域	名称	R/W	描述
7:5	PLSEL	R/W	选择 PVD 电压阈值（PVD Level Select） 0x0：2.2V 0x1：2.3V 0x2：2.4V 0x3：2.5V 0x4：2.6V 0x5：2.7V 0x6：2.8V 0x7：2.9V 注：详细说明参见“数据手册”
8	BPWEN	R/W	使能写备份区域（Backup Domain Write Access Enable） 备份区域指 RTC、备份寄存器，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		

6.6.2 电源控制/状态寄存器（PMU_CSTS）

偏移地址：0x04

复位值：0x0000 0000（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
2	PVDOFLG	R	PVD 输出标志位（PVD Output Flag） 表示 V_{DD}/V_{DDA} 是否高于由 PLSEL[2:0]选定的 PVD 阀值 当 PVD 被 PVDEN 位使能后该位才有效。 0：V_{DD}/V_{DDA} 高于 PVD 阀值 1：V_{DD}/V_{DDA} 低于 PVD 阀值 注：复位后或者进入待机模式时（PVD 停止工作），该位为 0。
7:3	保留		
8	WKUPCFG	R/W	WKUP 引脚配置（WKUP Pin Configure） WKUP 作为普通 I/O 时，WKUP 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

位/域	名称	R/W	描述
			注：在系统复位时清除这一位
31:9	保留		

7 备份寄存器（BAKPR）

7.1 简介

备份寄存器可以用于存储 84 个字节的数据，含有 42 个 16 位的寄存器。当 V_{DD} 关闭时，备份域将由 V_{BAT} 保持通电。

唤醒待机模式下的系统，如果系统复位或者电源复位时，备份寄存器不会复位。BAKPR 控制寄存器管理侵入检测和 RTC 校验。

当 BAKP 复位后，将禁止对备份寄存器和 RTC 的访问，并保护备份域(BAKPR)免受可能的意外写访问。如果要重新启用对备份寄存器和 RTC 的访问，请按照以下步骤操作：

- 通过在 RCM_APB1CLKEN 寄存器中设置 PMU 和 BAKP 位启用电源和备用接口时钟
- 设置 PMU_CTRL 电源控制寄存器的 BPWEN 位启用对备份寄存器和 RTC 的访问

7.2 主要特征

（1） 84 字节数据寄存器

（2） 状态/控制寄存器用于管理具有中断功能的侵入检测上拉输入

（3） 时钟校准寄存器，可以存储 RTC 校准值

（4） 在侵入引脚 PC13（TAMPER）上输出 RTC 校准时钟、RTC 闹钟脉冲或秒脉冲(当该引脚不用于侵入检测时)

7.3 功能描述

7.3.1 侵入检测

可以根据 TAMPER 引脚上信号是否发生变化，判断是否产生侵入事件。侵入检测事件会重置所有数据备份寄存器。为了避免丢失侵入事件，检测信号同时检测边缘检测的信号和侵入检测使能位，就可以检测到在检测之前的侵入事件。当设置 TPALCFG 位时，如果侵入引脚在启用之前已经为有效电平，启用侵入引脚后，产生一个另外的侵入的事件。如果还设置了 BAKPR_CSTS 寄存器的 TPIEN 位，在发生侵入检测事件时会产生中断。

在检测到侵入事件并清除后，禁用侵入引脚，如果要重新启用侵入检测功能，为了防止软件写入备份数据 BAKPR_DATAx 寄存器时侵入引脚上仍有侵入检测事件，需要在写入备份数据 BAKPR_DATAx 寄存器之前设置 BAKPR_CTRL 寄存器的 TPFCFG 位（相当于侵入引脚检测）。

注：当 V_{DD} 电源关闭时，侵入检测仍然处于活动状态。侵入引脚应该外接到正确的电平防止复位数据备份寄存器重置。

7.3.2 RTC 校准

通过配置在 RTC 时钟校准 BAKPR_CLKCAL 寄存器的 CALCOEN 位来启用 RTC 校准。

RTC 时钟可以经过 64 分频输出到侵入引脚上。

7.4 寄存器地址映射

表格 38 BAKPR 寄存器地址映射

寄存器名	描述	偏移地址
BAKPR_DATAx(x=1..10)	备份数据寄存器 1	0x04+4(x-1)
BAKPR_CLKCAL	RTC 时钟校准寄存器	0x2C
BAKPR_CTRL	备份控制寄存器	0x30
BAKPR_CSTS	备份控制/状态寄存器	0x34
BAKPR_DATAx(x=11…42)	备份数据寄存器 11	0x40+4(x-1)

7.5 寄存器功能描述

外设寄存器可以通过半字(16 位)或字(32 位)访问。

7.5.1 备份数据寄存器 x（BAKPR_DATAx）(x=1…10，11…42)

偏移地址：0x04 到 0x28,0x40 到 0xBC

复位值：0x0000

位/域	名称	R/W	描述
15:0	DATA	R/W	用户备份数据（User Backup Data） 在 V_{BAT} 供电存在的情况下，无法通过电源复位、系统复位、待机模式唤醒复位对 BAKPR_DATAx 寄存器复位，只能通过复位备份域或侵入事件复位。

7.5.2 RTC 时钟校准寄存器（BAKPR_CLKCAL）

偏移地址：0x2C

复位值：0x0000 0000

位/域	名称	R/W	描述
6:0	CALVALUE	R/W	设置校准值（Calibration Value Setup） 通过跳过 RTC 的时钟脉冲数来减小 RTC 的时钟，实现校准。该值表示每 2^{20} 个时钟脉冲将被忽略的多个时钟脉冲数，可以减慢 0 到 121ppm。
7	CALCOEN	R/W	使能输出校准时钟（Calibration Clock Output Enable） 0：无输出 1：侵入引脚输出 64 分频后的 RTC 时钟，如果 LSECLK 是 32.768KHz，输出的信号频率是 512Hz。当 CALCOEN 位被设置时，需要关闭侵入检测功能来避免检测到不必要的侵入信号。 注：当 V_{DD} 电源关闭时，该位被清除。
8	ASPOEN	R/W	使能输出警报或秒脉冲（Alarm or Second Pulse Output Enable） 0：禁用 1：在侵入引脚上输出 RTC 进门警报或秒脉冲信号

位/域	名称	R/W	描述
			输出脉冲持续时间为一个 RTC 时钟周期；设置 ASPOEN 位时，需要禁用侵入检测功能。 注：此位仅通过备份域复位来清除。
9	ASPOSEL	R/W	选择输出警报或秒脉冲（Alarm or Second Pulse Output Select） 该位能够选择侵入引脚输出 RTC 秒脉冲信号或警报脉冲信号 0：输出 RTC 警报脉冲 1：输出 RTC 秒脉冲 注：此位仅通过备份域复位来清除。
31:10	保留		

7.5.3 备份控制寄存器（BAKPR_CTRL）

偏移地址：0x30

复位值：0x0000 0000

位/域	名称	R/W	描述
0	TPFCFG	R/W	配置侵入检测引脚功能（TAMPER Pin Function Configure） 0：侵入引脚作为通用 IO 口使用 1：侵入引脚复用做侵入检测
1	TPALCFG	R/W	配置侵入引脚有效电平（TAMPER Pin Active Level Configure） 选择侵入引脚检测到的有效电平复位所有的数据备份寄存器。 0：高电平 1：低电平
31:2	保留		

注：同时设置 TPALCFG 和 TPFCFG 位总是安全的。但同时清除两者会产生一个假的侵入事件。因此，推荐只在 TPFCFG 为 0 时才改变 TPALCFG 位的状态。

7.5.4 备份控制/状态寄存器（BAKPR_CSTS）

偏移地址：0x34

复位值：0x0000 0000

位/域	名称	R/W	描述
0	TECLR	W	清除侵入检测事件标志（Tamper Event Flag Clear） 该位只能写入，读出值为 0 0：无效 1：清除侵入检测事件标志，并复位侵入检测功能
1	TICLR	W	清除侵入检测中断标志（Tamper Interrupt Flag Clear） 该位只能写入，读出值为 0 0：无效 1：清除侵入检测中断和中断标志
2	TPIEN	R/W	使能侵入检测引脚中断（TAMPER Pin Interrupt Enable） 该位仅在系统复位或由待机模式唤醒后才复位，侵入中断无法在低功耗模式下将系统内核唤醒。 0：禁止 1：使能（必须设置 TPFCFG 位）
7:3	保留		

位/域	名称	R/W	描述
8	TEFLG	R	发生侵入检测事件标志（TAMPER Event Occur Flag） 该位在检测到侵入事件时由硬件置位，通过向 TECLR 位写 1 清除 0：无侵入事件 1：检测到侵入事件 注：侵入事件会复位所有的备份数据寄存器。若该位为 1，所有的备份数据寄存器将保持复位状态，无法成功对备份数据进行写操作。
9	TIFLG	R	发生侵入检测中断标志（TAMPER Interrupt Occur Flag） 该位在设置了 TPIEN 位并检测到侵入事件时由硬件置位，通过向 TICLR 位写 1 清除；该位仅在系统复位或由待机模式唤醒后才复位。 0：无侵入中断 1：发生侵入中断
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 39 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

8.2 简介

产品中的 Cortex-M3 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-M3 技术参考手册》。

8.3 主要特征

（1） 65 个可屏蔽中断通道（不包括 16 个 Arm® Cortex®-M3 中断线）

（2） 16 个可编程的优先级（使用 4 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 40 APM32E103xCxE 中断和异常向量表

异常类型	向量编号	优先级	向量地址	描述
-	-	-	0x0000_0000	保留
Reset	-	-3	0x0000_0004	复位
NMI	-	-2	0x0000_0008	不可屏蔽中断
HardFault	-	-1	0x0000_000C	各种硬件故障
MemManage	-	可设置	0x0000_0010	存储器管理
BusFault	-	可设置	0x0000_0014	-
UsageFault	-	可设置	0x0000_0018	-
-	-	-	0x0000_001C-0x0000_002B	保留
SVCall	-	可设置	0x0000_002C	SWI 指令实现系统服务调用

异常类型	向量编号	优先级	向量地址	描述
Debug Monitor	-	可设置	0x0000_0030	调试监控器
-	-	-	0x0000_0034	保留
PendSV	-	可设置	0x0000_0038	可挂起系统服务请求
SysTick	-	可设置	0x0000_003C	系统节拍定时器
WWDT	0	可设置	0x0000_0040	窗口看门狗中断
PVD	1	可设置	0x0000_0044	电源电压检测中断
TAMPER	2	可设置	0x0000_0048	侵入检测中断
RTC	3	可设置	0x0000_004C	RTC 中断
FLASH	4	可设置	0x0000_0050	闪存全局中断
RCM	5	可设置	0x0000_0054	RCM 中断
EINT0	6	可设置	0x0000_0058	EINT 线 0 中断
EINT1	7	可设置	0x0000_005C	EINT 线 1 中断
EINT2	8	可设置	0x0000_0060	EINT 线 2 中断
EINT3	9	可设置	0x0000_0064	EINT 线 3 中断
EINT4	10	可设置	0x0000_0068	EINT 线 4 中断
DMA1 通道 1	11	可设置	0x0000_006C	DMA1 通道 1 全局中断
DMA1 通道 2	12	可设置	0x0000_0070	DMA1 通道 2 全局中断
DMA1 通道 3	13	可设置	0x0000_0074	DMA1 通道 3 全局中断
DMA1 通道 4	14	可设置	0x0000_0078	DMA1 通道 4 全局中断
DMA1 通道 5	15	可设置	0x0000_007C	DMA1 通道 5 全局中断
DMA1 通道 6	16	可设置	0x0000_0080	DMA1 通道 6 全局中断
DMA1 通道 7	17	可设置	0x0000_0084	DMA1 通道 7 全局中断
ADC1/2	18	可设置	0x0000_0088	ADC1 和 ADC2 全局中断
USBD1_HP_CAN1_TX	19	可设置	0x0000_008C	USBD1 高优先级中断/CAN1 发送中断
USBD1_LP_CAN1_RX0	20	可设置	0x0000_0090	USBD1 低优先级中断/CAN1 接收 0 中断
CAN1_RX1	21	可设置	0x0000_0094	CAN1 接收 1 中断
CAN1_SCE	22	可设置	0x0000_0098	CAN1 SCE 中断
EINT9_5	23	可设置	0x0000_009C	EINT 线[9:5]中断
TMR1_BRK	24	可设置	0x0000_00A0	TMR1 刹车中断
TMR1_UP	25	可设置	0x0000_00A4	TMR1 更新中断
TMR1_TRG_COM	26	可设置	0x0000_00A8	TMR1 触发和通信中断

异常类型	向量编号	优先级	向量地址	描述
TMR1_CC	27	可设置	0x0000_00AC	TMR1 捕获比较中断
TMR2	28	可设置	0x0000_00B0	TMR2 中断
TMR3	29	可设置	0x0000_00B4	TMR3 中断
TMR4	30	可设置	0x0000_00B8	TMR4 中断
I2C1_EV	31	可设置	0x0000_00BC	I2C1 事件中断
I2C1_ER	32	可设置	0x0000_00C0	I2C1 错误中断
I2C2_EV	33	可设置	0x0000_00C4	I2C2 事件中断
I2C2_ER	34	可设置	0x0000_00C8	I2C2 错误中断
SPI1	35	可设置	0x0000_00CC	SPI1 中断
SPI2	36	可设置	0x0000_00D0	SPI2 中断
USART1	37	可设置	0x0000_00D4	USART1 中断
USART2	38	可设置	0x0000_00D8	USART2 中断
USART3	39	可设置	0x0000_00DC	USART3 中断
EINT15_10	40	可设置	0x0000_00E0	EINT 线[15:10]中断
RTC_Alarm	41	可设置	0x0000_00E4	RTC 闹钟中断
USBD_WakeUp	42	可设置	0x0000_00E8	USBD_OTG 唤醒中断
TMR8_BRK	43	可设置	0x0000_00EC	TMR8 刹车中断
TMR8_UP	44	可设置	0x0000_00F0	TMR8 更新中断
TMR8_TRG_COM	45	可设置	0x0000_00F4	TMR8 触发和通信中断
TMR8_CC	46	可设置	0x0000_00F8	TMR8 捕获比较中断
ADC3	47	可设置	0x0000_00FC	ADC3 全局中断
EMMC	48	可设置	0x0000_0100	EMMC 中断
SDIO	49	可设置	0x0000_0104	SDIO 中断
TMR5	50	可设置	0x0000_0108	TMR5 中断
SPI3	51	可设置	0x0000_010C	SPI3 中断
UART4	52	可设置	0x0000_0110	UART4 中断
UART5	53	可设置	0x0000_0114	UART5 中断
TMR6	54	可设置	0x0000_0118	TMR6 中断
TMR7	55	可设置	0x0000_011C	TMR7 中断
DMA2 通道 1	56	可设置	0x0000_0120	DMA2 通道 1 中断
DMA2 通道 2	57	可设置	0x0000_0124	DMA2 通道 2 中断
DMA2 通道 3	58	可设置	0x0000_0128	DMA2 通道 3 中断
DMA2 通道 4/5	59	可设置	0x0000_012C	DMA2 通道 4/5 中断

异常类型	向量编号	优先级	向量地址	描述
-	-	-	0x0000_0130- 0x0000_0133	保留
USBD2_HP_CAN2 _TX	61	可设置	0x0000_0134	USBD2 高优先级中断 /CAN2 发送中断
USBD2_LP_CAN2_ RX0	62	可设置	0x0000_0138	USBD2 低优先级中断 /CAN2 接收 0 中断
CAN2_RX1	63	可设置	0x0000_013C	CAN2 接收 1 中断
CAN2_SCE	64	可设置	0x0000_0140	CAN2 SCE 中断

9 外部中断/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，硬件触发可触发预先设置的工作。例如，外部事件为产生 GPIO 输出脉冲，内部事件为一个 TMR 的更新事件触发另一个 TMR 工作。

（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

9.2 功能描述

9.2.1 “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 41“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，在中断挂起寄存器对应位写 1，将清除该中断请求。

名称	触发源	配置及执行过程
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。
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，在中断挂起寄存器对应位写 1，将清除该中断请求。

9.2.2 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：

9.2.2.1 事件唤醒

触发内部中断（内部硬件事件）但不触发中断处理函数唤醒

（1） 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；
（2） 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；
（3） 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）。

通过 EINT 线事件（外部硬件事件）唤醒

（1） 配置 EINT 线为事件模式；
（2） 执行 WFE 指令使内核进入睡眠；
（3） 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。

9.2.3 外部中断与事件线映射

表格 42 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/PE0/PF0/PG0	EINT 0
PA1/PB1/PC1/PE1/PF1/PG1	EINT 1

外部中断与事件通道名称	外部中断与事件线编号
…	…
PA15/PB15/PC15/PE15/PF15/PG15	EINT 15
PVD 输出	EINT 16
RTC Alarm 事件	EINT 17
USBD 唤醒事件	EINT 18
保留	EINT 19
保留	EINT 20
保留	EINT 21
保留	EINT 22
保留	EINT 23
保留	EINT 24
保留	EINT 25
保留	EINT 26
保留	EINT 27
保留	EINT 28
保留	EINT 29
保留	EINT 30
保留	EINT 31

9.3 寄存器地址映射

表格 43 外部中断/事件控制器寄存器映射

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	使能上升沿触发选择寄存器	0x08
EINT_FTEN	使能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	中断挂起寄存器	0x14

9.4 寄存器功能描述

9.4.1 中断屏蔽寄存器（EINT_IMASK）

偏移地址：0x00

复位值：0x0000 0000

位/域	名称	R/W	描述
18:0	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
31:19	保留		

9.4.2 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
18:0	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
31:19	保留		

9.4.3 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
18:0	RTENx	R/W	使能线 x 上的上升沿触发事件和中断（Rising Trigger Event and Interrupt Enable of Line x） 0：禁止 1：使能
31:19	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。

9.4.4 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
18:0	FTENx	R/W	使能线 x 上的下降沿触发事件（Falling Trigger Event Enable of Line x） 0：禁止（中断和事件） 1：使能（中断和事件）
31:19	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。

9.4.5 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
18:0	SWINTEx	R/W	线 x 上的软件中断（Software Interrupt Event on Line x） 软件置 1，对 EINT_IPEND 的对应位写入 1 清 0。 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）。 0：无作用 1：软件产生中断（事件）
31:19	保留		

9.4.6 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
18:0	IPENDx	RC_W1	发生线 x 上的中断挂起标志（Interrupt Pending Occur of Line x Flag） 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。
31:19	保留		

10 直接存储器存取（DMA）

10.1 术语全称、缩写描述

表格 44 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。

产品一共有两个 DMA 控制器，DMA1 有 7 个通道，DMA2 有 5 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） DMA1 有 7 个通道，DMA2 有 5 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 12 个通道，DMA1 有 7 个，DMA2 有 5 个，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 45 DMA1 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_UP	TMR1_CH3	—
TMR2	TMR2_CH3	TMR2_UP	—	—	TMR2_CH1	—	TMR2_CH2 TMR2_CH4
TMR3	—	TMR3_CH3	TMR3_CH4 TMR3_UP	—	—	TMR3_CH1 TMR3_TRIG	—
TMR4	TMR4_CH1	—	—	TMR4_CH2	TMR4_CH3	—	TMR4_UP
ADC1	ADC1	—	—	—	—	—	—
SPI/I2S	—	SPI1_RX	SPI1_TX	SPI/I2S2_RX	SPI/I2S2_TX	—	—
USART	—	USART3_TX	USART3_RX	USART1_TX	USART1_RX	USART2_RX	USART2_TX
I2C	—	—	—	I2C2_TX	I2C2_RX	I2C1_TX	I2C1_RX

表格 46 DMA2 请求映射表

外设	通道 1	通道 2	通道 3	通道 4	通道 5
TMR5	TMR5_CH4 TMR5_TRIG	TMR5_CH3 TMR5_UP	—	TMR5_CH2	TMR5_CH1
TMR6/DAC 通道 1	—	—	TMR6_UP/ DAC 通道 1	—	—
TMR7/DAC 通道 2	—	—	—	TMR7_UP/ DAC 通道 2	—

外设	通道 1	通道 2	通道 3	通道 4	通道 5
TMR8	TMR8_CH3 TMR8_UP	TMR8_CH4 TMR8_TRGI TMR8_COM	TMR8_CH1	—	TMR8_CH2
ADC3	—	—	—	—	ADC3
SPI/I2S3	SPI/I2S3_RX	SPI/I2S3_TX	—	—	—
UART4	—	—	UART4_RX	—	UART4_TX
SDIO	—	—	—	SDIO	—

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 11 源为 8bits 目标为 8bits 的传输宽度

图 12 源为 8bits 目标为 16bits 的传输宽度

图 13 源为 8bits 目标为 32bits 的传输宽度

图 14 源为 32bits 目标为 8bits 的传输宽度

图 15 源为 16bits 目标为 16bits 的传输宽度

图 16 源为 16bits 目标为 32bits 的传输宽度

图 17 源为 32bits 目标为 16bits 的传输宽度

10.4.2.3 地址设置

传输地址支持两种模式：固定模式、指针增量模式。

传输地址指针增量模式

外设和存储器的指针自动增量通过配置寄存器 DMA_CHCFGx 的 PERIMODE 位和 MIMODE 位完成。下一个要传输的地址就是将前一个地址加上增量，增量值取决于所选的数据宽度。

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。

如果对存储器是执行的是写操作（目标地址），存储器包括内部 SRAM、EMMC 支持的外部 RAM（例如外部 SRAM、SDRAM）与 NORFLASH；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、EMMC 支持的 RAM、NORFLASH。

“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 47 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
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

位/域	名称	R/W	描述
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
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

位/域	名称	R/W	描述
			01：16 位 10：32 位 11：保留 注意：用户在使用 I2C3 时不能配置成 00。
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 注意：用户在使用 I2C3 时不能配置成 00。
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+20 x（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+20 x（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 48 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
串行/JTAG 调试端口	Serial Wire/JTAG Debug Port	SWJ-DP

11.2 简介

APM32E10x 微控制器系列使用 Arm® Cortex®-M3 内核，Arm® Cortex®-M3 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。

支持两种调试接口：

- 串行接口
- JTAG 调试接口

注意：Arm® Cortex®-M3 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®-M3 内核的调试功能信息，请参考 Cortex®-M3（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） 可代替内核访问 AHB 总线矩阵

（2） 灵活的调试引脚分配

（3） MCU 调试盒(支持低电源模式，控制外设时钟等)

图 18 APM32E10xxx 级别和 Arm® Cortex®-M3 级别的调试框图

11.4 功能描述

11.4.1 调试引脚功能配置

（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

表格 49 引脚功能配置

SWJ- CFG[2:0]	配置为调试专用的引脚	SWJ 接口的 I/O 口分配				
		PA13/ JTMS/ SWDIO	PA14/ JTCK/ SWCLK	PA15/ JTDI	PB3/ JTDO	PB4/ JNTRST
其它	禁止	释放				
100	JTAG-DP 接口和 SW-DP 接口都禁止					
010	JTAG-DP 接口禁止， SW-DP 接口允许	专用	专用	释放		
001	所有的 SWJ 引脚 (JTAG-DP+SW-DP) 除了 JNTRST 引脚	专用	专用	专用	专用	释放
000	所有的 SWJ 引脚 (JTAG-DP+SW-DP) 复位状态	专用	专用	专用	专用	专用

注意：对于一些运行模式下不能测试到的项目可以详细观察测试

11.4.2 ID 编码

11.4.2.1 微控制器设备 ID 编码

APM32E 微控制器系列内含一个 MCU ID 编码。使用 JTAG 或 SW 调试口或用户代码都可以进行访问。

11.4.2.2 边界扫描 TAP

JTAG ID 编码

APM32E 微控制器系列的边界扫描 TAP 集成了 JTAG ID 编码，对于 APM32E103xCxE 系列产品，其 JTAG ID 编码是 0x06414B47。

11.4.2.3 Arm® Cortex®-M3 TAP

Arm® Cortex®-M3 的 TAP 有一个 JTAG ID 编码，编码是 0x4BA00477。

11.4.2.4 Arm® Cortex®-M3 JEDEC-106 ID 代码

Arm® Cortex®-M3 有一个 JEDEC-106 ID 编码。它位于映射到内部 PPB 总线地址为 0xE00FF000_0xE00FFFFF 的 4KB ROM 表中。

11.5 寄存器地址映射

表格 50 寄存器地址映射

寄存器名	描述	地址
DBGMCU_IDCODE	设备 ID 寄存器	0xE004 2000
DBGMCU_CFG	调试 MCU 配置寄存器	0xE004 2004

11.6 寄存器功能描述

11.6.1 设备 ID 寄存器（DBGMCU_IDCODE）

地址：0xE004 2000

只支持 32 位访问

复位值：0xXXXX XXXX，X=未定义位

位/域	名称	R/W	描述
11:0	EQR	R	设备识别（Equipment Recognition） 对于 APM32E10x 微控制器系列： APM32E103xCxE 系列产品：0x414； 调试器/编程工具通过 EQR[11:0]来识别芯片。
15:12	保留		
31:16	WVR	R	晶圆版本识别（Wafer Version Recognition） 该域标识晶圆信息。

11.6.2 调试 MCU 配置寄存器（DBGMCU_CFG）

此寄存器可以在调试状态下配置 MCU。包括支持定时器和看门狗的计数器、低功耗模式、CAN 通信和分配跟踪引脚。

地址：0xE004 2004

只支持 32 位访问

复位值：0x0000 0000（不受系统复位影响，仅上电复位）

位/域	名称	R/W	描述
0	SLEEP_CLK_STS	R/W	调试睡眠模式时，配置时钟状态（Configure clock status when MCU is debugged in sleep mode） 0：FCLK 开，HCLK 关 1：FCLK 开，HCLK 开，由系统时钟提供
1	STOP_CLK_STS	R/W	调试停机模式时，配置时钟状态（Configure clock status when MCU is debugged in stop mode） 0：FCLK 关，HCLK 关 1：FCLK 开，HCLK 开，由 HSICLK 提供
2	STANDBY_CLK_STS	R/W	调试待机模式时，配置时钟状态（Configure clock status when MCU is debugged in standby mode） 0：FCLK 关，HCLK 关 1：FCLK 开，HCLK 开，由 HSICLK 提供
4:3	保留		

位/域	名称	R/W	描述
5	TRACE_IOEN	R/W	使能跟踪调试引脚（Trace Debug Pin Enable） 0：禁止跟踪调试引脚 1：使能跟踪调试引脚
7:6	TRACE_MODE	R/W	配置跟踪调试引脚模式（Trace Debug Pin Mode Configure） 仅当 TRACE_IOEN=1 时，可配置跟踪调试引脚模式： 00：异步模式 01：同步模式，数据长度为 1 10：同步模式，数据长度为 2 11：同步模式，数据长度为 4
8	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） 0：正常工作 1：停止工作
9	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） 0：正常工作 1：停止工作
13:10	TMRx_STS	R/W	内核停止时，配置定时器 x(x＝1...4)的工作状态（ConfigureTimer Work Status When Core Is in Halted） 0：正常工作 1：停止工作
14	CAN1_STS	R/W	内核停止时，配置 CAN1 的工作状态（Configure CAN1 Work Status When Core Is in Halted） 0：正常工作 1：冻结 CAN1 的接收寄存器
15	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
16	I2C2_SMBUS_TIMEOUT_STS	R/W	内核停止时，配置 I2C2_SMBUS_TIMEOUT 的工作状态（Configure I2C2_SMBUS_TIMEOUT Work Status When Core Is in Halted） 0：正常工作 1：冻结 SMBUS 的超时模式
20:17	TMRx_STS	R/W	内核停止时，配置定时器 x(x＝5...8)的工作状态（ConfigureTimer Work Status When Core Is in Halted） 0：内核停止时也可以向相关定时器的计数器提供时钟，定时器输出工作正常 1：内核停止时不向相关定时器的计数器提供时钟且禁止定时器输出
21	CAN2_STS	R/W	内核停止时，配置 CAN2 的工作状态（Configure CAN2 Work Status When Core Is in Halted） 0：正常工作 1：冻结 CAN2 的接收寄存器

位/域	名称	R/W	描述
31:22	保留		

12 通用输入/输出引脚（GPIO）

12.1 术语全称、缩写描述

表格 51 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

GPIO 端口可以通过 32 位的配置寄存器（GPIOx_CFGLOW/GPIOx_CFGHIG）和两个 32 位的数据寄存器 GPIOx_IDATA/GPIOx_ODATA）配置以下功能：

（1） 输入模式
- 模拟输入
- 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） GPIO 都可以作为外部中断/唤醒线

（5） 支持锁定 I/O 配置功能

12.3 结构框图

图 19 GPIO 结构框图

图 20 5 伏兼容 GPIO 结构框图

（1）V_{DD_FT} 和 V_{DD} 不一样，V_{DD_FT} 对 5 伏容忍 GPIO 引脚是特殊的。

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，此情况下输入模式禁用上拉/下拉电阻。复位后 JTAG 引脚被置于输入上拉

或下拉模式，具体配置如下：

- PA15：JTDI 置于上拉模式
- PA14：JTCK 置于下拉模式
- PA13：JTMS 置于上拉模式
- PB4：JNTRST 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
- 根据输入配置的不同，连接弱上拉和下拉电阻
- 输入数据寄存器 GPIOx_IDATA 在每个 APB2 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。

模拟输入模式

在模拟输入模式中

- 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
- 端口输入状态寄存器的值为 0

图 21 输入模式结构

12.4.3 输出模式

在输出模式中可以设置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
- 禁止弱上拉和下拉电阻
- 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
  - 通过输入数据寄存器 GPIOx_IDATA 读取 I/O 的实际状态

图 22 输出模式 I/O 结构

12.4.4 复用模式

在复用模式中可以设置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
- 禁止弱上拉和下拉电阻
- I/O 引脚上的数据在每个 APB2 时钟周期采样并存入端口输入状态寄存器
- 开漏模式中，通过输入数据寄存器 GPIOx_IDATA 可以读取 I/O 的实际状态
- 推挽模式中，通过输出数据寄存器 GPIOx_ODATA 读取最后写入的值

图 23 复用模式 I/O 结构

12.4.5 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。

12.4.6 位设置和位清除

在对 GPIOx_IDATA 的一些位进行编程时，软件不需要禁止中断。（可以通过对 GPIOx_BSC 和 BSC 寄存器中想要更改的位置 1 实现在 APB2 写操作中，只更改一个或者多个位的功能。

12.4.7 GPIO 锁定功能

锁定功能可以用在电源驱动模块。GPIO 的锁定机制可以保护 I/O 端口的配置。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。

12.5 寄存器地址映射

表格 52 GPIO 寄存器地址映射

寄存器名	描述	偏移地址
GPIOx_CFGLOW	端口配置低 8 位寄存器	0x00
GPIOx_CFGHIG	端口配置高 8 位寄存器	0x04
GPIOx_IDATA	端口输入数据寄存器	0x08
GPIOx_ODATA	端口输出数据寄存器	0x0C
GPIOx_BSC	端口位设置/清除寄存器	0x10
GPIOx_BC	端口位清除寄存器	0x14
GPIOx_LOCK	端口配置锁定寄存器	0x18

12.6 寄存器功能描述

必须以字（32 位）的方式操作这些外设寄存器。

12.6.1 低 8 位端口配置寄存器（GPIOx_CFGLOW）（x=A..G）

偏移地址：0x00

复位值：0x4444 4444

位/域	名称	R/W	描述
29:28，25:24 21:20，17:16 13:12，9:8 5:4 1:0	MODEy[1:0]	R/W	配置端口 x 模式（y=0…7）（Port x mode Configure） 00：输入模式（复位后的状态） 01：输出模式，最大输出速度是 10MHz 10：输出模式，最大输出速度是 2MNz 11：输出模式，最大输出速度是 50MHz 最大输出速度的定义见数据手册。
31:30，27:26 23:22，19:18 15:14 11:10 7:6 3:2	CFGy[1:0]	R/W	配置端口 x 功能（y=0…7）（Port x Function Configure） 软件通过这些位配置相应的 I/O 端口。 在输入模式（MODE[1:0]=00）时 00：模拟输入模式 01：浮空输入模式（复位后的状态） 10：上拉/下拉输入模式 11：保留 在输出模式（MODE[1:0]>00） 00：通用推挽输出模式 01：通用开漏输出模式 10：复用功能推挽输出模式 11：复用功能开漏输出模式

12.6.2 高 8 位端口配置寄存器（GPIOx_CFGHIG）（x=A..G）

偏移地址：0x04

复位值：0x4444 4444

位/域	名称	R/W	描述
29:28 25:24 21:20 17:16 13:12 9:8 5:4 1:0	MODEy[1:0]	R/W	配置端口 x 模式（y=8…15）（Port x mode Configure） 软件通过这些位配置相应的 I/O 端口。 00：输入模式（复位后的状态） 01：输出模式，最大输出速度是 10MHz 10：输出模式，最大输出速度是 2MHz 11：输出模式，最大输出速度是 50MHz 最大输出速度的定义见数据手册。
31:30 27:26 23:22 19:18 15:14 11:10 7:6	CFGy[1:0]	R/W	配置端口 x 功能（y=8…15）（Port x Function Configure） 软件通过这些位配置相应的 I/O 端口。 在输入模式（MODE[1:0]=00） 00：模拟输入模式 01：浮空输入模式（复位后的状态） 10：上拉/下拉输入模式 11：保留

位/域	名称	R/W	描述
3:2			在输出模式（MODE[1:0]>00） 00：通用推挽输出模式 01：通用开漏输出模式 10：复用功能推挽输出模式 11：复用功能开漏输出模式

12.6.3 端口输入数据寄存器（GPIOx_IDATA）（x=A...G）

偏移地址：0x08

复位值：0x0000 XXXX

位/域	名称	R/W	描述
15:0	IDATAy	R	端口输入数据（y=0…15）（Port input data） 这些位为只读并只能以字的形式读出。 0：输出信号是低电平 1：输出信号时高电平
31:16	保留		

12.6.4 端口输出数据寄存器（GPIOx_ODATA）（x=A…G）

偏移地址：0x0C

复位值：0x0000 0000

位/域	名称	R/W	描述
15:0	ODATAy	R/W	端口输出数据（y=0…15）（Port output data） 这些位可读可写并只能以字的形式操作。 0：输出低电平 1：输出高电平 注：对 GPIOx_BSC（x=A…E），可以分别地对各个 ODATAy 位进行独立的设置/清除。
31:16	保留		

12.6.5 端口位设置/清除寄存器（GPIOx_BSC）（x=A…G）

偏移地址：0x10

复位值：0x0000 0000

位/域	名称	R/W	描述
15:0	BSy	W	设置端口 x 的位 y（y=0…15）（Port x Set bit y） 这些位用来影响对应的 ODATAy 位。 0：不产生影响 1：设置对应的 ODATAy 位为 1 这些位只能进行写操作，并且只能以字的形式操作。
31:16	BCy	W	清除端口 x 的位 y（y=0…15）（Port x Clear bit y） 这些位用来影响对应的 ODATAy 位。 0：不产生影响 1：对应的 ODATAy 位清 0 注：如果同时设置了 BSy 和 BCy 的对应位，BSy 位起作用。 这些位只能进行写操作，并且只能以字的形式操作。

12.6.6 端口位清除寄存器（GPIOx_BC）（x=A…G）

偏移地址：0x14

复位值：0x0000 0000

位/域	名称	R/W	描述
15:0	BCy	W	清除端口 x 的位 y（y=0…15）（Port x Clear bit y） 这些位用来影响对应的 ODATAy 位。 0：不产生影响 1：对应的 ODATAy 位清 0 这些位只能进行写操作，并且只能以字的形式操作。
31:16	保留		

12.6.7 端口配置锁定寄存器（GPIOx_LOCK）（x=A…G）

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，需要等系统复位后才可以修改。配置 GPIO 配置时，需要对该寄存器致写入指定序列才能启动 GPIO 锁定功能。

偏移地址：0x18

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 的锁位（y=0…15）（Port x Pin y Lock bit y Configure） 这些位决定端口的配置是否锁定。 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。
16	LOCKKEY	R/W	锁键值（Lock key Value） 该位决定端口配置锁键位是否被激活 0：未激活 1：被激活，下次系统复位前 GPIOx_LOCK 寄存器被锁住 该位可随时读出，它只可通过锁键写入序列修改 锁键的写入序列： 写 1 写 0 写 1 读 0 读 1 读 1（最后一个读可省略，但可以用来确认锁键已被激活。） 注：锁键值序列中，LOCKy 的值不能改变，出现的任何错误（序列错误、读错误）都无法激活锁定保护。
31:17	保留		

13 复用功能输入输出引脚（AFIO）

13.1 简介

I/O 端口除了实现通用的输入输出功能外，还能实现作为多种外设功能的接口，为了充分利用产品的外设 I/O 引脚，产品支持复用功能。既可以在同一个引脚实现多个功能（同一时刻只能实现一个功能），也可以将某一功能重映射至其它 I/O 上（原来支持的功能不再支持）。

13.2 功能描述

13.2.1 I/O 引脚的复用功能

为了充分利用产品的外设 I/O 引脚，可以把一些复用功能重新映射到其他空闲引脚上，以达到引脚资源的利用最大化。

表格 53 复用功能相应端口配置

复用功能	对端口位配置寄存器进行配置
复用输入功能	配置成输入模式，且输入引脚必须由外部驱动
复用输出功能	配置成复用功能输出模式
双向复用功能	配置成复用功能输出功能，输入驱动器被配置成浮空输入模式

注意：

（1）通过 GPIO 控制器编程，用软件模拟复用功能输入引脚，此时端口被设置为复用功能输出模式，引脚通过 GPIO 控制器由软件驱动。

（2）对于复用功能输出时，引脚和输出寄存器断开，并和片上外设的输出信号连接，如果连接后外设并没有被激活，那么该引脚的输出将不确定。

13.2.1.1 输入模式配置

在 I/O 端口被用作复用功能输入模式时，除了弱上拉和下拉电阻被禁止，端口配置与通用输入功能时一样。

相应模式配置详情请见 GPIO 中的对应章节。

13.2.1.2 输出模式配置

在 I/O 端口被用作复用功能输出模式（推挽或开漏）时，与通用输出功能一样，在输出模式中可以设置为推挽输出和开漏输出，但此时的输出缓冲器是由内置外设的信号驱动的。

相应模式配置详情请见 GPIO 中的对应章节。

注意：当软件模拟复用功能输入引脚时，I/O 端口应该被配置为复用功能输出模式。

13.2.1.3 双向复用功能配置

使用双向复用功能时，I/O 端口必须配置为复用功能输出模式（推挽或开漏），而

输入驱动器需被配置为浮空输入模式。

相应模式配置详情请见 GPIO 中的对应章节。

13.2.2 外设引脚配置

此时由于外设引脚可能会有不同的功能，所以引脚的 I/O 端口配置不同。

表格 54 TMR 的引脚配置

TMR 引脚	配置	I/O 端口配置
TMR1/8_CHx	输入捕获通道 x	浮空输入
	输出比较通道 x	推挽复用输出
TMR1/8_CHxN	互补输出通道 x	推挽复用输出
TMR1/8_BKIN	刹车输入	浮空输入
TMR1/8_ETR	外部触发时钟输入	浮空输出
TMR2/3/4/5_CHx	输入捕获通道 x	浮空输入
	输出比较通道 x	推挽复用输出
TMR2/3/4/5_ETR	外部触发时钟输入	浮空输入

表格 55 USART 的引脚配置

USRAT 引脚	配置	I/O 端口配置
USRATx_TX	全双工模式	推挽复用输出
	半双工同步模式	推挽复用输出
USARTx_RX	全双工模式	浮空输入或带上拉输入
	半双工同步模式	未用，可作为 GPIO
USARTx_CK	同步模式	推挽复用输出
USARTx_RTS	硬件流控制	推挽复用输出
USARTx_CTS	硬件流控制	浮空输入或带上拉输入

表格 56 SPI 的引脚配置

SPI 引脚	配置	I/O 端口配置
SPIx_SCK	主模式	推挽复用输出
	从模式	浮空输入
SPIx_MOSI	全双工模式/主模式	推挽复用输出
	全双工模式/从模式	浮空输入或带上拉输入
	简单的双向数据线/主模式	推挽复用输出
	简单的双向数据线/从模式	未用，可作为 GPIO
SPIx_MISO	全双工模式/主模式	浮空输入或带上拉输入
	全双工模式/从模式	推挽复用输出

SPI 引脚	配置	I/O 端口配置
	简单的双向数据线/主模式	未用，可用作 GPIO
	简单的双向数据线/从模式	推挽复用输出
SPIx_NSS	硬件主/从模式	浮空输入或带上拉输入或带下拉输入
	硬件主模式/NSS 输出使能	推挽复用输出
	软件模式	未用，可作为 GPIO

表格 57 I2S 的引脚配置

I2S 引脚	配置	I/O 端口配置
I2Sx_WS	主模式	推挽复用输出
	从模式	浮空输入
I2Sx_CK	主模式	推挽复用输出
	从模式	浮空输入
I2Sx_SD	发送器	推挽复用输出
	接收器	浮空输入或带上拉输入或带下拉输入
I2Sx_MCK	主模式	推挽复用输出
	从模式	未用，可作为 GPIO

表格 58 I2C 的引脚配置

I2C 引脚	配置	I/O 端口配置
I2Cx_SCL	I2C 时钟	开漏复用输出
I2Cx_SDA	I2C 数据	开漏复用输出

表格 59 BxCAN 的引脚配置

CAN 引脚	I/O 端口配置
CAN_TX	推挽复用输出
CAN_RX	浮空输入或带上拉输入

表格 60 USBD 的引脚配置

USBD 引脚	I/O 端口配置
USBD_DM/USBD_DP	USBD 模块被使能后，引脚会自动连接到内部 USBD 收发器

注意：本表只适用于大容量产品。

表格 61 SDIO 的引脚配置

SDIO 引脚	I/O 配置
SDIO_CK	推挽复用输出

SDIO 引脚	I/O 配置
SDIO_CMD	推挽复用输出
SDIO[D7:D0]	推挽复用输出

表格 62 ADC/DAC 的引脚配置

ADC/DAC 引脚	I/O 端口配置
ADC/DAC	模拟输入

表格 63 SMC 的引脚配置

SMC 引脚	I/O 端口配置
SMC_A[25:0]	推挽复用输出
SMC_D[15:0]	推挽复用输出
SMC_CK	推挽复用输出
SMC_NOE	推挽复用输出
SMC_NWE	推挽复用输出
SMC_NE[4:1]	推挽复用输出
SMC_NCE[3:2]	推挽复用输出
SMC_NCE4_1	推挽复用输出
SMC_NCE4_2	推挽复用输出
SMC_NWAIT	浮空输入或带上拉输入
SMC_CD	浮空输入或带上拉输入
SMC_NIOS16	浮空输入
SMC_INTR	浮空输入
SMC_INT[3:2]	浮空输入
SMC_NL	推挽复用输出
SMC_NBL[1:0]	推挽复用输出
SMC_NIORD	推挽复用输出
SMC_NIOWR	推挽复用输出
SMC_NREG	推挽复用输出

表格 64 DMC 的引脚配置

DMC 引脚	I/O 端口配置
DMC_A[15:0]	推挽复用输出
DMC_D[12:0]	推挽复用输出
DMC_CLK	推挽复用输出
DMC_CKE	推挽复用输出

DMC 引脚	I/O 端口配置
DMC_NCS	推挽复用输出
DMC_NWE	推挽复用输出
DMC_NCAS	推挽复用输出
DMC_NRAS	推挽复用输出
DMC_UDQM	推挽复用输出
DMC_LDQM	推挽复用输出
DMC_BA0	推挽复用输出
DMC_BA1	推挽复用输出

13.2.3 重映射功能配置

一般来说，系统复位之后，引脚会被赋予一个默认的功能，此时如果用户需要复用引脚的其他功能，只要使能该外设，就可以激活复用功能。但有些外设的功能除了需要使能，还需要软件编程把信号映射至端口，即分配引脚地址，才可以在引脚使用该外设功能。

引脚的复用功能和重映射地址表见数据手册。

13.2.3.1 OSC32_IN（OUT）引脚配置为 GPIO

在没有进入待机模式或不由 V_{DD} 供电的情况下，当 LSECLK 振荡器关闭时，其引脚 OSC32_IN/OSC32_OUT 可以用作通用 I/O 口 PC14/PC15，即 LSECLK 功能优先于通用 I/O 功能。

13.2.3.2 OSC_IN（OUT）引脚配置为 GPIO

在少于 100 脚的封装产品中，用户可以通过设置 AFIO_REMAP1/2（复用重映射和调试 I/O 配置寄存器），实现用作通用 I/O 的 PD0/PD1 重映射至外部振荡器引脚 OSC_IN/OSC_OUT 引脚。此时，PD0 和 PD1 不能用来产生外部中断时间。

13.3 寄存器地址映射

表格 65 AFIO 寄存器地址映射

寄存器名	描述	偏移地址
AFIO_EVCTRL	事件控制寄存器	0x00
AFIO_REMAP1	复用重映射配置寄存器 1	0x04
AFIO_EINTSEL1	外部中断配置寄存器 1	0x08
AFIO_EINTSEL2	外部中断配置寄存器 2	0x0C
AFIO_EINTSEL3	外部中断配置寄存器 3	0x10
AFIO_EINTSEL4	外部中断配置寄存器 4	0x14
AFIO_REMAP2	复用重映射配置寄存器 2	0x18

13.4 寄存器功能描述

对寄存器 AFIO_EVCTRL，AFIO_REMAP1/2 和 AFIO_EINTSELx 进行读写操作前，应当首先打开 AFIO 的时钟。APB2 外设时钟使能寄存器（RCM_APB2CLKEN）。必须以字（32 位）的方式操作这些外设寄存器。

13.4.1 事件控制寄存器（AFIO_EVCTRL）

偏移地址：0x00

复位值：0x0000 0000

位/域	名称	R/W	描述
3:0	PINSEL	R/W	选择端口 x（x=A…E）引脚 y（y=0…15）（Portx Piny Select） 选择用于输出内核的 EVENTOUT 信号的引脚： 0000：选择 Px0 0001：选择 Px1 0010：选择 Px2 0011：选择 Px3 0100：选择 Px4 0101：选择 Px5 0110：选择 Px6 0111：选择 Px7 1000：选择 Px8 1001：选择 Px9 1010：选择 Px10 1011：选择 Px11 1100：选择 Px12 1101：选择 Px13 1110：选择 Px14 1111：选择 Px15
6:4	PORTSEL	R/W	选择端口 x（x=A…G）（Portx Select） 选择用于输出内核的 EVENTOUT 信号的端口： 000：选择 PA 001：选择 PB 010：选择 PC 011：选择 PD 100：选择 PE 101：选择 PF 110：选择 PG 111：保留
7	EVOEN	R/W	允许事件输出（Event Output Enable） 0：禁止 1：允许内核的 EVENTOUT 将连接到由 PORTSEL[2:0]和 PINSEL[3:0]选定的端口 x 引脚 y。
31:8	保留		

13.4.2 复用重映射寄存器 1（AFIO_REMAP1）

偏移地址：0x04

复位值：0x0000 0000

位/域	名称	R/W	描述
0	SPI1RMP	R/W	配置 SPI1 的 NSS、SCK、MISO 和 MOSI 重映射（SPI1 Remap Configure） 0：没有重映射 NSS—PA4，SCKPA5，MISO—PA6，MOSI—PA7 1：重映射 NSS—PA15，SCK—PB3，MISO—PB4，MOSI—PB5
1	I2C1RMP	R/W	配置 I2C1 的 SCL 和 SDA 重映射(I2C1 Remap Configure） 0：没有重映射 SCL—PB6，SDA—PB7 1：重映射 SCL—PB8，SDA—PB9
2	USART1RMP	R/W	配置 USART1 的 TX 和 RX 重映射（USART1 Remap Configure） 0：没有重映射 TX—PA9，RX—PA10 1：重映射 TX—PB6，RX—PB7
3	USART2RMP	R/W	配置 USART2 的 CTS、RTS、CK、TX 和 RX 重映射（USART2 Remap Configure） 0：没有重映射 CTS—PA0，RTS—PA1，TX—PA2，RX—PA3，CK—PA4 1：重映射 CTS—PD3，RTS—PD4，TX—PD5，RX—PD6，CK—PD7
5:4	USART3RMP	R/W	配置 USART3 的 CTS、RTS、CK、TX 和 RX 重映射（USART3 Remap Configure） 00：没有重映射 TX—PB10，RX—PB11，CK—PB12，CTS—PB13，RTS—PB14 01：部分重映射 TX—PC10，RX—PC11，CK—PC12，CTS—PB13，RTS—PB14 10：无作用 11：完全重映射 TX—PD8，RX—PD9，CK—PD10，CTS—PD11，RTS—PD12
7:6	TMR1RMP	R/W	配置定时器 1 的通道 1 至 4、1N 至 3N、外部触发（ETR）和刹车输入（BKIN）重映射（TMR1 Remap Configure） 00：没有重映射 ETR—PA12，CH1—PA8，CH2—PA9，CH3—PA10，CH4—PA11，BKIN—PB12，CH1N—PB13，CH2N—PB14，CH3N—PB15 01：部分映射 ETR—PA12，CH1—PA8，CH2—PA9，CH3—PA10，CH4—PA11，BKIN—PA6，CH1N—PA7，CH2N—PB0，CH3N—PB1 10：无作用 11：完全映射

位/域	名称	R/W	描述
			ETR—PE7，CH1—PE9，CH2—PE11，CH3—PE13，CH4—PE14，BKIN—PE15，CH1N—PE8，CH2N—PE10，CH3N—PE12
9:8	TMR2RMP	R/W	配置定时器 2 的通道 1 至 4 和外部触发（ETR）重映射（TMR2 Remap Configure） 00：没有重映射 CH1/ETR—PA0，CH2—PA1，CH3—PA2，CH4—PA3 01：部分重映射 CH1/ETR—PA15，CH2—PB3，CH3—PA2，CH4—PA3 10：部分重映射 CH1/ETR—PA0，CH2—PA1，CH3—PB10，CH4—PB11 11：完全重映射 CH1/ETR—PA15，CH2—PB3，CH3—PB10，CH4—PB11
11:10	TMR3RMP	R/W	配置定时器 3 的通道 1 至 4 重映射（TMR3 Remap Configure） 00：没有重映射 CH1—PA6，CH2—PA7，CH3—PB0，CH4—PB1 01：无作用 10：部分映射 CH1—PB4，CH2—PB5，CH3—PB0，CH4—PB1 11：完全映射 CH1—PC6，CH2—PC7，CH3—PC8，CH4—PC9 注：重映射不影响在 PD2 上的 TMR3_ETR。
12	TMR4RMP	R/W	配置定时器 4 的通道 1-4 重映射（TMR4 Remap Configure） 0：没有重映射 TMR4_CH1—PB6，TMR4_CH2—PB7，TMR4_CH3—PB8，TMR4_CH4—PB9 1：完全映射 TMR4_CH1—PD12，TMR4_CH2—PD13，TMR4_CH3—PD14，TMR4_CH4—PD15 注：重映射不影响在 PE0 上的 TMR4_ETR。
14:13	CAN1RMP	R/W	配置 CAN1 的 CAN1_RX 和 CAN1_TX 重映射（CAN1 Remap Configure） 00：CAN1_RX 映射到 PA11，CAN1_TX 映射到 PA12 01：无作用 10：CAN1_RX 映射到 PB8，CAN1_TX 映射到 PB9 11：CAN1_RX 映射到 PD0，CAN1_TX 映射到 PD1
15	PD01RMP	R/W	配置 PD0/PD1 映射到 OSC_IN/OSC_OUT（Port D0/Port D1 mapping on OSC_IN/OSC_OUT Configure） 此功能只能适用于 64 引脚的封装（PD0 和 PD1 出现在 100 脚的封装上，不必重映射）。 0：不进行 PD0 和 PD1 的重映射 1：PD0 映射到 OSC_IN，PD1 映射到 OSC_OUT 当不使用主振荡器 HSECLK 时（系统运行于内部的 8MHz 阻容振荡器），PD0 和 PD1 可以映射到 OSC_IN 和 OSC_OUT 引脚。

位/域	名称	R/W	描述
16	TMR5CH4IRMP	R/W	TMR5 通道 4 内部重映射（TMR5CH4 Interrupt Remap） 该位可由软件置 1 或清 0。它控制 TMR5 通道 4 内部映射。 0：TMR5_CH4 与 PA3 相连； 1：LIRC 内部振荡器与 TMR5_CH4 相连，目的是对 LIRC 进行校准。
17	ADC1_ETRGINJC_RMP	R/W	配置 ADC1 注入转换的外部触发重映射（ADC1 External Trigger Injected Conversion Remapping Configure） 0：ADC1 注入转换的外部触发与 EINT15 相连 1：ADC1 注入转换的外部触发与 TMR8_CH4 相连
18	ADC1_ETRGREGC_RMP	R/W	配置 ADC1 规则转换的外部触发重映射（ADC 1 External Trigger Regular Conversion Remapping Configure） 0：ADC1 规则转换的外部触发与 EINT11 相连 1：ADC1 规则转换的外部触发与 TMR8_TRGO 相连
19	ADC2_ETRGINJC_RMP	R/W	配置 ADC2 注入转换的外部触发重映射（ADC2 External Trigger Injected Conversion Remapping Configure） 0：ADC2 注入转换的外部触发与 EINT15 相连 1：ADC2 注入转换的外部触发与 TMR8_CH4 相连
20	ADC2_ETRGREGC_RMP	R/W	配置 ADC2 规则的转换外部触发重映射（ADC2 External Trigger Regular Conversion Remapping Configure） 0：ADC2 规则转换的外部触发与 EINT11 相连 1：ADC2 规则转换的外部触发与 TMR8_TRGO 相连
21	保留		
22	CAN2RMP	R/W	配置 CAN2 的 CAN2_RX 和 CAN2_TX 重映射（CAN2 Remap Configure） 0：无重映射 CAN2_RX—PB12，CAN2_TX—PB13 1：重映射 CAN2_RX—PB5，CAN2_TX—PB6
23	保留		
26:24	SWJCFG	W	配置串行线 JTAG（Serial Wire JTAG Configure） 配置 SWJ 和跟踪复用功能的 I/O 是作为调试 I/O 还是普通 I/O，适用于 GPIO 不够用的情况，这些位只可由软件写（读这些位，将返回未定义的数值）。 SWJ（串行线 JTAG）支持 JTAG 或 SWD 访问 Cortex 的调试端口，系统复位后的默认状态是启用 SWJ 但没有跟踪功能。 000：完全 SWJ（JTAG-DP+SW-DP） 001：完全 SWJ（JTAG-DP+SW-DP）但没有 NJTRST 010：关闭 JTAG-DP，启用 SW-DP 100：关闭 JTAG-DP，关闭 SW-DP 其它：保留
31:27	保留		

13.4.3 外部中断配置寄存器 1（AFIO_EINTSEL1）

偏移地址：0x08

复位值：0x0000 0000

位/域	名称	R/W	描述
15:0	EINTx[3:0]	R/W	选择外部中断线 x（EINTx）输入源（x=0…3）（EINTx Input Source Select） 0000：PA[x]引脚 0001：PB[x]引脚 0010：PC[x]引脚 0011：PD[x]引脚 0100：PE[x]引脚 0101：PF[x]引脚 0110：PG[x]引脚 其他：保留
31:16	保留		

13.4.4 外部中断配置寄存器 2（AFIO_EINTSEL2）

偏移地址：0x0C

复位值：0x0000 0000

位/域	名称	R/W	描述
15:0	EINTx[3:0]	R/W	选择外部中断线 x（EINTx）输入源（x=4…7）（EINTx Input Source Select） 0000：PA[x]引脚 0001：PB[x]引脚 0010：PC[x]引脚 0011：PD[x]引脚 0100：PE[x]引脚 0101：PF[x]引脚 0110：PG[x]引脚 其他：保留
31:16	保留		

13.4.5 外部中断配置寄存器 3（AFIO_EINTSEL3）

偏移地址：0x10

复位值：0x0000 0000

位/域	名称	R/W	描述
15:0	EINTx[3:0]	R/W	选择外部中断线 x（EINTx）输入源（x=8…11）（EINTx Input Source Select） 0000：PA[x]引脚 0001：PB[x]引脚 0010：PC[x]引脚 0011：PD[x]引脚 0100：PE[x]引脚 0101：PF[x]引脚 0110：PG[x]引脚 其他：保留
31:16	保留		

13.4.6 外部中断配置寄存器 4（AFIO_EINTSEL4）

偏移地址：0x14

复位值：0x0000 0000

位/域	名称	R/W	描述
15:0	EINTx[3:0]	R/W	选择外部中断线 x（EINTx）输入源（x=12…15）（EINTx Input Source Select） 0000：PA[x]引脚 0001：PB[x]引脚 0010：PC[x]引脚 0011：PD[x]引脚 0100：PE[x]引脚 0101：PF[x]引脚 0110：PG[x]引脚 其他：保留
31:16	保留		

13.4.7 复用重映射寄存器 2（AFIO_REMAP2）

偏移地址：0x1C

复位值：0x0000 0000

位/域	名称	R/W	描述
9:0	保留		
10	EMMCNADV	R/W	NADV 连接模式（NADV Connect/Disconnect） 位由软件控制改写。 0：NADV 标志与输出连接（出现错误） 1：NADV 标志不与输出连接，I/O 引脚可以被其他外设使用。
31:11	保留		

14 定时器概述

14.1 术语全称、缩写描述

表格 66 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
捕获	Capture	C
比较	Compare	C
长度	Length	LEN

14.2 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、系统滴答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 67 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	
名称	—	TMR1	TMR8	TMR2/3/4/5	TMR6	TMR7
时基单元	计数器	16 位		16 位	16 位	
	预分频器	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	
通道	输入通道	4		4	0	
	捕获比较通道	4		4	0	
	输出通道	7		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
功能	产生 DMA 请求	可以	可以	可以
	PWM 模式	有	有	无
	单脉冲模式	有	有	无
	强制输出模式	有	有	无
	死区插入	有	无	无

定时器术语

表格 68 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 69 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
TIxF，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

15 高级定时器（TMR1/8）

15.1 简介

高级定时器以时基单元为核心，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。高级定时器相比较其它定时器增加了互补输出、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。

15.2 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

15.3 结构框图

图 24 高级定时器结构框图

15.4 功能描述

15.4.1 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 TI1/2/3/4，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的脉冲信号进行逻辑相或以后的信号就是 TI1F_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。

15.4.2 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
- 自动重装载寄存器（AUTORLD）16 位
- 预分频器（PSC）16 位
- 重复次数寄存器（REPCNT）8 位

重复次数寄存器是高级定时器特有的。

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）值与自动重装载（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，其中自动重装载的值（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、自动重装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 25 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，计数器从自动重装载的值（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，自动重装载的值（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、自动重装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 26 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装载的值（TMRx_AUTORLD），然后从自动重装载的值（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 27 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，也就表明了在基本/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 28 向上计数模式下，设置 REPCNT=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

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。

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 TMRx_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，周期是由自动重装载 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 29 PWM1 向上计数模式的时序图

图 30 PWM1 向下计数模式的时序图

图 31 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,AUTORLD=7,在 PWM 模式 2 下的时序图

图 32 PWM2 向上计数模式的时序图

图 33 PWM2 向下计数模式的时序图

图 34 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，另一路会自动配置会相反的极性。

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 35 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 36 单脉冲模式下的时序图

15.4.8 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 37 影响输出波形的寄存器结构关系

15.4.9 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。

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 38 发生刹车事件的时序图

15.4.10 互补输出和死区插入

互补输出是高级定时器特有的输出，高级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 39 带死区插入的互补输出

15.4.11 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。

15.4.12 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，计时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 70 计数方向与编码器的关系

有效边沿		仅在 TI1 计数		仅在 TI2 计数		在 TI1 和 TI2 计数	
相对信号的电平		高	低	高	低	高	低
TI1FP1	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
TI2FP2	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 IC1FP1 映射到 TI1
- 将 IC2FP2 映射到 TI2
- IC1FP1 和 IC2FP2 都不反相
- 输入信号在上升沿和下降沿有效
- 使能计数器

图 40 编码器模式下的计数器操作实例

例如：当 TI1 处在低电平时，如果 TI2 出现上升沿状态，计数器向上计数。

图 41 IC1FP1 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。

15.4.13 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，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。

15.4.14 定时器互连

TMRx 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 42 定时器 1 主/从模式的例子

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

15.4.15 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。

15.4.16 在外部事件时清除 OCxREF 信号

此功能是用于输出比较和 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 43 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 44 OCxREF 时序图

15.5 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 71 高级定时器寄存器地址映射

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

寄存器名	描述	偏移地址
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

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。 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

位/域	名称	R/W	描述
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 寄存器自动重装载缓冲使能（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	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才起作用。 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

位/域	名称	R/W	描述
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 Selection） 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 PLOCKCFG 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		

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	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。 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。 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。
3	保留		
6:4	TRGSEL	R/W	选择触发输入信号（Trigger Input Signal Select） 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。 000：内部触发 ITR0 001：内部触发 ITR1 010：内部触发 ITR2 011：内部触发 ITR3 100：通道 1 输入边沿检测器 TIF_ED 101：通道 1 滤波后定时器输入 TI1FP1 110：通道 2 滤波后的定时器输入 TI2FP2 111：外部触发输入（ETRF）
7	MSMEN	R/W	使能主/从模式（Master/slave Mode Enable） 0：无效 1：使能主/从模式
11:8	ETFCFG	R/W	配置外部触发滤波器（External Trigger 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 1011：DIV=16，N=6 1100：DIV=16，N=8 1101：DIV=32，N=5

位/域	名称	R/W	描述
			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 Mode2 Enable） 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 72 TMRx 内部触发连接

从定时器	ITR1（TS=000）	ITR1（TS=001）	ITR2（TS=010）	ITR3（TS=011）
TMR1	TMR5	TMR2	TMR3	TMR4
TMR8	TMR1	TMR2	TMR4	TMR5

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/Campare 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/Campare 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/Campare Channel3 Interrupt Enable）

位/域	名称	R/W	描述
			0：禁止 1：使能
4	CC4IEN	R/W	使能捕获/比较通道 4 中断（Capture/Campare 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 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/Campare 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/Campare Channe2 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/Campare Channe3 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/Campare Channe4 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		

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	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。
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。

位/域	名称	R/W	描述
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
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

位/域	名称	R/W	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。
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		

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 OCx 描述了通道在输出模式下的功能，寄存器中的 ICx 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 Selection） 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。

位/域	名称	R/W	描述
			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 CNT 的值和捕获比较寄存器的值 CCx 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，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） 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 Perscaler Configure） 00：PSC=1

位/域	名称	R/W	描述
			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 Configuration） 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 Perscaler Configuration）
15:12	IC2F	R/W	配置输入捕获通道 2 滤波器（Input Capture Channel2 Filter Configuration）

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 Selection） 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

位/域	名称	R/W	描述
			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
9:8	CC4SEL	R/W	选择通道 4 模式（Capture/compare Channel4 Selection） 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 Perscaler Configuration） 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 Configuration）
9:8	CC4SEL	R/W	选择捕获/比较通道 4（Capture/Compare Channel4 Select）

位/域	名称	R/W	描述
			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 Perscaler Configuration）
15:12	IC4F	R/W	配置输入捕获通道 4 滤波器（Input Capture Channel4 Filter Configuration）

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	使能捕获/比较通道 1 输出（Capture/Compare Channel1 Output Enable） **捕获/比较通道 1 配置为输出时：** 0：禁止输出 1：开启输出 **捕获/比较通道 1 配置为输入时：** 该位决定了计数器的值 CNT 是否能捕获进入 TMRx_CC1 寄存器中 0：禁止捕获 1：开启捕获
1	CC1POL	R/W	配置捕获/比较通道 1 输出极性（Capture/Compare Channel1 Output Polarity Configure） **CC1 通道配置为输出时：** 0：OC1 高电平有效 1：OC1 低电平有效 **CC1 通道配置为输入时：** 0：不反相：在 IC1 的上升沿进行捕获，IC1 作为外部触发器时不反相。 1：反相，在 IC1 的下降沿进行捕获，IC1 作为外部触发器时反相。 注：当保护级别为 2 或者 3 时，该位不能被修改
2	CC1NEN	R/W	使能捕获/比较通道 1 互补输出（Capture/Compare Channel1 Complementary Output Enable） 0：禁止 1：开启
3	CC1NPOL	R/W	捕获/比较通道 1 互补输出极性（Capture/Compare Channel1 Complementary Output Polarity） 0：OC1N 高电平有效 1：OC1N 低电平有效 注：当保护级别为 2 或者 3 时，该位不能被修改
4	CC2EN	R/W	使能捕获/比较通道 2 输出（Capture/Compare Channel2 Output Enable） 参考 CCEN_CC1EN
5	CC2POL	R/W	配置捕获/比较通道 2 输出极性（Capture/Compare Channel2 Output Polarity Configure）

位/域	名称	R/W	描述
			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		

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）

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

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

15.6.13 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		

15.6.14 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。

15.6.15 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

15.6.16 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

15.6.17 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

15.6.18 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	设置互补输出通道的死区持续时间（Dead Time Setup） DT 为死区持续时间，DT 与寄存器 DTS 的关系如下： DTS[7:5]=0xx=>DT=DTS[7:0]×T_{DTS}，T_{DTS}=TDTS； DTS[7:5]=10x=>DT=（64+DTS[5:0]）×T_{DTS}，T_{DTS}=2×T_{DTS}； DTS[7:5]=110=>DT=（32+DTS[4:0]）×T_{DTS}，T_{DTS}=8×T_{DTS}； DTS[7:5]=111=>DT=（32+DTS[4:0]）×T_{DTS}，T_{DTS}=16×T_{DTS}； 例：假设 T_{DTS}=125ns（8MHZ），死区时间设置如下： 若步长时间为 125ns，可设置死区时间范围是 0 到 15875ns； 若步长时间为 250ns，可设置死区时间范围是 16us 到 31750ns； 若步长时间为 1us，可设置死区时间范围是 32us 到 63us； 若步长时间为 2us，可设置死区时间范围是 64us 到 126us。 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。
9:8	LOCKCFG	R/W	配置锁定写保护模式（Lock Write Protection Mode Configuration） 00：无锁定写保护，可直接写寄存器 01：锁定写保护级别 1 不能写入 TMRx_BDT 的 DTS、BRKEN、BRKPOL、AOEN 位和 TMRx_CTRL2 寄存器的 OCxOIS 和 OCxNOIS 位。 10：锁定写保护级别保护级别 2 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。 11：锁定写保护级别保护级别 3 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。 注意：在系统复位后，只能写一次锁定写保护位。
10	IMOS	R/W	配置空闲模式下的关闭状态（Idle Mode Off-state Configure） 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。 0：禁止 OCx/OCxN 输出 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
11	RMOS	R/W	配置运行模式下的关闭状态（Run Mode Off-state Configure） 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。 0：禁止 OCx/OCxN 输出

位/域	名称	R/W	描述
			1：OCx/OCxN 先输出无效电平（具体电平数值受极性配置影响）
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值置 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。

15.6.19 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 突发传送长度（DMA Burst Transfer Length Setup） 这些位定义 DMA 在连续模式下的传送长度、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送； 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：

位/域	名称	R/W	描述
			传输的地址=TMRx_CTRL1 的地址（从地址）+DBADDR+DMA 索引；DMA 索引=DBLEN 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， 数据的传输将发生在：TMRx_CTRL1 的地址+从 DBADDR 开始的 7 个寄存器。 根据设置的 DMA 数据长度不同，数据传输也会发生变化： (1) 当传输数据设置为 16 位时，数据会传输给 7 个寄存器 (2) 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。
15:13	保留		

15.6.20 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	DMA 突发传送寄存器（DMA Register for Burst Transfer） 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 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。

16 通用定时器（TMR2/3/4/5）

16.1 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 位的自动重装载计数器（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。

16.2 主要特征

（1） 时基单元
- 计数器：16 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
- 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

（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

16.3 结构框图

图 45 通用定时器结构框图

16.4 功能描述

16.4.1 时钟源选择

通用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 TI1/2/3/4，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的脉冲信号进行逻辑相或以后的信号就是 TI1F_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。

16.4.2 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 位
- 自动重装载寄存器（AUTORLD）16 位
- 预分频器（PSC）16 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）值与自动重装载（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，其中自动重装载的值（TMRx_AUTORLD）是提前写入的。

当计数器溢出时，会产生更新事件，此时自动重装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 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，计数器从自动重装载的值（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，自动重装载的值（TMRx_AUTORLD）是提前写入的。

计数器溢出时，会产生更新事件，此时，自动重装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装载的值（TMRx_AUTORLD），然后从自动重装载的值（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。

16.4.3 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。

16.4.4 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 TMRx_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。

16.4.5 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，周期是由自动重装载 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,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

16.4.6 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，另一路会自动配置会相反的极性。

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

16.4.7 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

16.4.8 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。

16.4.9 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，计时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 73 计数方向与编码器的关系

有效边沿		仅在 TI1 计数		仅在 TI2 计数		在 TI1 和 TI2 计数	
相对信号的电平		高	低	高	低	高	低
TI1FP1	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
TI2FP2	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 IC1FP1 映射到 TI1
- 将 IC2FP2 映射到 TI2
- IC1FP1 和 IC2FP2 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 TI1 处在低电平时，如果 TI2 出现上升沿状态，计数器向上计数。

图 58 IC1FP1 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。

16.4.10 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，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。

16.4.11 定时器互连

详情见 15.4.14 定时器互连

16.4.12 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。

16.4.13 在外部事件时清除 OCxREF 信号

此功能是用于输出比较和 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

16.5 寄存器地址映射

下表中将通用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 74 通用定时器寄存器地址映射

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

寄存器名	描述	偏移地址
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_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

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。

位/域	名称	R/W	描述
			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 寄存器自动重装载缓冲使能（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		

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
2:0	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，发出 CCx 的 DMA 请求 1：当发生更新事件时，发出 CCx 的 DMA 请求

位/域	名称	R/W	描述
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 Selection） 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
15:8	保留		

16.6.3 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。 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。 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。
3	保留		
6:4	TRGSEL	R/W	选择触发输入信号（Trigger Input Signal Select） 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。 000：内部触发 ITR0 001：内部触发 ITR1

位/域	名称	R/W	描述
			010：内部触发 ITR2 011：内部触发 ITR3 100：通道 1 输入边沿检测器 TIF_ED 101：通道 1 滤波后定时器输入 TI1FP1 110：通道 2 滤波后的定时器输入 TI2FP2 111：外部触发输入（ETRF）
7	MSMEN	R/W	使能主/从模式（Master/slave Mode Enable） 0：无效 1：使能主/从模式
11:8	ETFCFG	R/W	配置外部触发滤波器（External Trigger 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 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 Mode2 Enable） 0：禁止 1：使能 设置 ECEN 位与选择外部时钟模式 1 将 TRG1 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 75 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	TMR8	TMR3	TMR4
TMR3	TMR1	TMR2	TMR5	TMR4
TMR4	TMR1	TMR2	TMR3	TMR8
TMR5	TMR2	TMR3	TMR4	TMR8

16.6.4 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/Campare 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/Campare 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/Campare 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/Campare 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/Campare 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/Campare Channe2 DMA Request Enable） 0：禁止

位/域	名称	R/W	描述
			1：使能
11	CC3DEN	R/W	使能捕获/比较通道 3 的 DMA 请求（Capture/Campare Channe3 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/Campare Channe4 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		

16.6.5 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 URSEL=0 和 UD=0，配置 TMRx_CEG 寄存器的 UG=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 URSEL=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 new Interrupt Flag） 参考 STS_CC1IFLG

位/域	名称	R/W	描述
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	保留		
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。
8:7	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
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		

16.6.6 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

位/域	名称	R/W	描述
			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 1：产生触发事件 该位由软件置 1，硬件自动清 0。
15:7	保留		

16.6.7 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 OCx 描述了通道在输出模式下的功能，寄存器中的 ICx 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 Selection） 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。

位/域	名称	R/W	描述
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 CNT 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） 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

位/域	名称	R/W	描述
			注意：该位仅在通道关闭时（TMRx_CCEN 寄存器的 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 Perscaler 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 Configuration） 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 Perscaler Configuration）
15:12	IC2F	R/W	配置输入捕获通道 2 滤波器（Input Capture Channel2 Filter Configuration）

16.6.8 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 Selection） 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
9:8	CC4SEL	R/W	选择通道 4 模式（Capture/compare Channel4 Selection） 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 Perscaler Configuration） 00：PSC=1 01：PSC=2 10：PSC=4

位/域	名称	R/W	描述
			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 Configuration）
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 Perscaler Configuration）
15:12	IC4F	R/W	配置输入捕获通道 4 滤波器（Input Capture Channel4 Filter Configuration）

16.6.9 使能捕获/比较通道寄存器（TMRx_CCEN）

偏移地址：0x20

复位值：0x0000

位/域	名称	R/W	描述
0	CC1EN	R/W	使能捕获/比较通道 1 输出（Capture/Compare Channel1 Output Enable） **捕获/比较通道 1 配置为输出时：** 0：禁止输出 1：开启输出 **捕获/比较通道 1 配置为输入时：** 该位决定了计数器的值 CNT 是否能捕获进入 TMRx_CC1 寄存器中 0：禁止捕获 1：开启捕获
1	CC1POL	R/W	配置捕获/比较通道 1 输出极性（Capture/Compare Channel1 Output Polarity Configure） **CC1 通道配置为输出时：** 0：OC1 高电平有效 1：OC1 低电平有效 **CC1 通道配置为输入时：** 0：不反相：在 IC1 的上升沿进行捕获，IC1 作为外部触发器时不反相。 1：反相，在 IC1 下降沿进行捕获，IC1 作为外部触发器时反相。 注：当保护级别为 2 或者 3 时，该位不能被修改
3:2	保留		
4	CC2EN	R/W	使能捕获/比较通道 2 输出（Capture/Compare Channel2 Output Enable） 参考 CCEN_CC1EN
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
7:6	保留		

位/域	名称	R/W	描述
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
11:10	保留		
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		

表格 76 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

注意：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。

16.6.10 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）

16.6.11 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）

16.6.12 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	自动重装载数值（Auto Reload Value） 自动重装载的值为空时，计数器不进行计数。

16.6.13 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。

16.6.14 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

16.6.15 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

16.6.16 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

16.6.17 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 …….

位/域	名称	R/W	描述
7:5	保留		
12:8	DBLEN	R/W	设置 DMA 突发传送长度（DMA Burst Transfer Length Setup） 这些位定义 DMA 在连续模式下的传送长度、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送； 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 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， 数据的传输将发生在：TMRx_CTRL1 的地址+从 DBADDR 开始的 7 个寄存器。 根据设置的 DMA 数据长度不同，数据传输也会发生变化： 1) 当传输数据设置为 16 位时，数据会传输给 7 个寄存器 2) 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。
15:13	保留		

16.6.18 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	DMA 突发传送寄存器（DMA Register for Burst Transfer） 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 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。

17 基本定时器（TMR6/7）

17.1 简介

基本定时器 TMR6 和 TMR7 有一个无符号的 16 位计数器、自动重装载寄存器、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，还为 DAC 提供时钟，通过配置可以产生 DMA 请求。

17.2 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

（4） 单脉冲模式

（5） 为 DAC 提供时钟

17.3 结构框图

图 61 基本定时器结构框图

17.4 功能描述

17.4.1 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_INT 驱动计数器。

17.4.2 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
- 自动重装载寄存器（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）值与自动重装载（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，自动重装载的值（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 URSSEL 位。

在发生一个更新事件时，自动重装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。

17.5 寄存器地址映射

下表中将基本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 77 基本定时器寄存器映射

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	自动重装载寄存器	0x2C

17.6 寄存器功能描述

17.6.1 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

位/域	名称	R/W	描述
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
6:4	保留		
7	ARPEN	R/W	TMRx_AUTORLD 寄存器自动重装载缓冲使能（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		

17.6.2 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
2:0	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
15:7	保留		

17.6.3 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		

17.6.4 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 URSEL=0 和 UD=0，配置 TMRx_CEG 寄存器的 UG=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 URSEL=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		

17.6.5 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。

17.6.6 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）

17.6.7 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）

17.6.8 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	自动重装载数值（Auto Reload Value） 自动重装载的值为空时，计数器不进行计数。

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、使时间更加精确以及使用变得更加灵活。

独立看门狗只有在计数器减到 0 时，会复位，在没有减到 0 之前，刷新计数器的值不会发生复位。

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 独立看门狗定时器（IWDT）

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 位重装载寄存器 IWDT_CNTRLD、键寄存器 IWDT_KEY 和状态寄存器 IWDT_STS 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 63 独立看门狗结构框图

注：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 键寄存器

在键寄存器中写入 0xCCCC 开启独立看门狗，此时计数器开始向下计数，当计数器计数到 0x000 时，产生一次复位。

在键寄存器中写入 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。

在键寄存器中写入 0x5555，用来改写预分频寄存器和重装载寄存器的值。

18.2.3.2 寄存器访问保护

预分频寄存器和重装载寄存器具有写保护功能，如果要改写这两个寄存器则需要写 0x5555 到键寄存器中，如果将其他的值写入键寄存器将会再次启动对寄存器的保护。

写 0xAAAA 到键寄存器中，也会启动写保护功能。

18.2.3.3 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。如果没有向键寄存器中写入 0xAAAA，计数器计数结束后，产生复位。

18.2.3.4 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_CFG 寄存器的 IWDT_STS 位。

18.3 窗口看门狗定时器（WWDT）

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 PCLK1，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 64 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 控制寄存器的 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 65 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}: APB1 以 ms 为单位的时钟周期

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

18.4 IWDT 寄存器地址映射

表格 78 IWDT 寄存器映射

寄存器名	描述	偏移地址
IWDT_KEY	关键字寄存器	0x00
IWDT_PSC	预分频寄存器	0x04
IWDT_CNTRLD	计数器重装载寄存器	0x08
IWDT_STS	状态寄存器	0x0C

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（在待机模式复位）

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 和 IWDT_CNTRLD 寄存器； 软件写入 0xAAAA 表示执行重装载计数器，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）； 该寄存器只写，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）

位/域	名称	R/W	描述
			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 计数器重装载寄存器（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装载值（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装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	更新看门狗预分频器系数标志（Watchdog Prescaler Factor 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	更新看门狗计数器重装载值标志（Watchdog Counter Reload Value Update Flag） 更新计数器重装载值时，由硬件置 1；计数器重装载值更新结束后，由硬件清 0；计数器重装载值只有在 CNTUFLG 位被清 0 时更新。
31:2	保留		

18.6 WWDT 寄存器地址映射

表格 79 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

寄存器名	描述	偏移地址
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。

18.7.1.1 控制寄存器（WWDT_CTRL）

偏移地址：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		

18.7.1.2 配置寄存器（WWDT_CFG）

偏移地址：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：无作用 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		

18.7.1.3 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	发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。
31:1	保留		

19 实时时钟（RTC）

19.1 术语全称、缩写描述

表格 80 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
溢出	Overflow	OVR
预分频器	Prescaler	PSC
时机时钟	Time Basic Clock	TBCLK

19.2 主要特征

实时时钟（RTC）是一个主电源掉电后自动切换备份电源，维持运行的定时器。

（1） 时基单元

（2） 可编程的 32 位计数器

（3） 多种中断控制

（4） 低功耗自动唤醒

19.3 结构框图

图 66 RTC 结构框图

19.4 功能描述

19.4.1 时基单元

时钟源

RTC 有 3 个时钟源：

- 外部 LSECLK 晶振
- 外部 HSECLK 晶振 128 分频
- 内部 LSICLK

通过配置时钟控制器 CCM 的寄存器选择不同时钟源。

预分频器

RTC 预分频器包含了一个 20 位的可编程分频器，可编程产生最长为 1 秒的 RTC 时间基准。

19.4.2 RTC 寄存器配置

为防止意外写入 RTC 寄存器导致计数异常，RTC 采用写保护机制，只有解除写保护才能对具有写保护功能的寄存器进行操作。

配置 RTC 时钟时需要先把电源控制寄存器的（PMU_CTRL）的 BPWEN 位置“1”才能进行配置，配置 RTC_CSTS 寄存器的 CFGMFLG 位，使 RTC 进入配置模式，才能配置 RTC_PSCRLD、RTC_CNT、RTC_ALR 寄存器；清除 RTC_CSTS 寄存器的 CFGMFLG 位，退出配置模式。

对 RTC 任何寄存器的写操作，需前一次写操作结束后（可通过查询 RTC_CSTS 的 OCFLG 判断）才能进行。

19.4.3 可编程的闹钟

作为一个实时时钟，RTC 内部集成闹钟功能，主要通过闹钟寄存器和计数器运作，通过寄存器 RTC_ALR 配置闹钟时间，使能闹钟功能后，当计数器值等于闹钟值时触发，闹钟标志置起，如果开启了闹钟中断，则触发中断处理，通过配置外部 17 线中断可以利用 RTC 闹钟唤醒低功耗。

19.4.4 RTC 输出

RTC 可以输出通过 PC13 引脚，把内部的 RTC 秒脉冲、闹钟信号和校准时钟输出给外部，通过配置 BAKPR_CLKCAL 寄存器可以选择输出的脉冲。

19.4.5 中断

RTC 可以产生秒中断、闹钟中断和溢出中断，当产生 20 位预分频器溢出、闹钟事件和 32 位计数器溢出时会把相应的状态标志位挂起，配置 RTC_CTRL 寄存器可产生对应的中断。

19.5 寄存器地址映射

表格 81 RTC 寄存器地址映射

寄存器名	描述	偏移地址
RTC_CTRL	RTC 控制寄存器	0x00
RTC_CSTS	RTC 控制/状态寄存器	0x04
RTC_PSCRLDH	RTC 预分频装载寄存器高位	0x08
RTC_PSCRLDL	RTC 预分频装载寄存器低位	0x0C
RTC_PSCH	RTC 预分频器余数寄存器高位	0x10
RTC_PSCL	RTC 预分频器余数寄存器低位	0x14
RTC_CNTH	RTC 计数器寄存器高位	0x18
RTC_CNTL	RTC 计数器寄存器低位	0x1C
RTC_ALRH	RTC 闹钟寄存器高位	0x20
RTC_ALRL	RTC 闹钟寄存器低位	0x24

19.6 寄存器功能描述

19.6.1 RTC 控制寄存器（RTC_CTRL）

偏移地址：0x00

复位值：0x0000

位/域	名称	R/W	描述
0	SECIEN	R/W	使能秒中断（Second Interrupt Enable） 0：禁止 1：使能
1	ALRIEN	R/W	使能闹钟中断（Alarm Interrupt Enable） 0：禁止 1：使能
2	OVRIEN	R/W	使能溢出中断（Overflow Interrupt Enable） 0：禁止 1：使能
15:3	保留		

19.6.2 RTC 控制/状态寄存器（RTC_CSTS）

偏移地址：0x04

复位值：0x0020

位/域	名称	R/W	描述
0	SECFLG	RC_W0	秒标志（Second Flag） 此标志可为 RTC 计数器提供一个周期性的信号（通常为 1 秒）。 当 32 位可编程预分频器溢出时，由硬件置 1，同时 RTC 计数器加 1；只能由软件写 0 清除。

位/域	名称	R/W	描述
			0：无秒标志 1：有秒标志
1	ALRFLG	RC_W0	发生闹钟标志（Alarm Occur Flag） 当计数器达到 RTC_ALR 数值时，由硬件置 1；只能软件写 0 清除。 0：无闹钟 1：有闹钟
2	OVRFLG	RC_W0	发生溢出标志（Overflow Occur Flag） 当计数器溢出时，由硬件置 1；只能软件写 0 清除。 0：无溢出 1：32 位可编程计数器溢出
3	RSYNCFLG	RC_W0	寄存器已被同步标志（Registers Synchronized Flag） 当 RTC_CNT、RTC_PSCRLD、RTC_ALR 寄存器已同步时，硬件置 1；由软件写 0 清除。 在 APB1 时钟复位或停止后，此位必须由软件清 0，用户程序需等待由硬件置 1，才能正确读取 RTC_CNT、RTC_PSCRLD、RTC_ALR 的数值。 0：未被同步 1：已被同步
4	CFGMFLG	R/W	使能配置模式标志（Configure Mode Enable Flag） 由软件写 1 进入配置模式，才可以对 RTC_CNT、RTC_ALR 或 RTC_PSCRLD 寄存器进行写操作；软件写 0 退出配置模式。 0：退出配置模式（开始更新 RTC 寄存器） 1：使能配置模式
5	OCFLG	R	RTC 操作完成标志（RTC Operation Complete Flag） 表示对上一次写 RTC 寄存器的状态。 0：未完成，无法执行下一次的写操作 1：已完成，可以执行下一次的写操作
15:6	保留		

19.6.3 RTC 预分频器重装载寄存器（RTC_PSCRLD）

该寄存器保存 RTC 预分频器的周期计数值，仅当 OCFLG 值为 1 时允许执行写操作。

RTC 预分频装载寄存器高位（RTC_PSCRLDH）

偏移地址：0x08

复位值：0x0000

位/域	名称	R/W	描述
3:0	PSCRLDH[19:16]	W	设置 RTC 预分频器重装载值高位（RTC Prescaler Reload Value High Setup） 根据以下公式，这些位用来定义时基时钟频率： f_{TBCLK}=f_{RTCCLK}/（RLD [19:0]+1）
15:4	保留		

RTC 预分频装载寄存器低位（RTC_PSCRLDL）

偏移地址：0x0C
复位值：0x8000

位/域	名称	R/W	描述
15:0	PSCRLDL[15:0]	W	设置 RTC 预分频器重装载值低位（RTC Prescaler Reload Value Low Setup） 根据以下公式，这些位用来定义时基时钟频率： f_{TBCLK}=f_{RTCCLK}/（RLD[19:0]+1）。

注：如果输入时钟频率是 32.768kHz（f_{RTCCLK}），这个寄存器中写入 7FFFh 可获得周期为 1 秒钟的信号。

19.6.4 RTC 预分频器寄存器（RTC_PSC）

该寄存器保存 RTC_PSCRLD 的数值，只读，在 RTC_PSCRLD 或 RTC_CNT 寄存器发生改变后，由硬件重新装载。

RTC 预分频器寄存器高位（RTC_PSCH）

偏移地址：0x10
复位值：0x0000

位/域	名称	R/W	描述
3:0	PSCH[19:16]	R	设置 RTC 时钟预分频器余数高位（RTC Clock Prescaler High Setup）
15:4	保留		

RTC 预分频器寄存器低位（RTC_PSCL）

偏移地址：0x14
复位值：0x8000

位/域	名称	R/W	描述
15:0	PSCL[15:0]	R	设置 RTC 时钟预分频器低位（RTC Clock Prescaler Low Setup）

19.6.5 RTC 计数器寄存器（RTC_CNT）

当 OCFLG 值为 1 时允许执行写操作；当进行读操作时，直接返回计数器内的计数值（系统时间）。

RTC 计数器寄存器高位（RTC_CNTH）

偏移地址：0x18
复位值：0x0000

位/域	名称	R/W	描述
15:0	CNTH[31:16]	R/W	设置 RTC 计数器高位（RTC Counter High Setup）

RTC 计数器寄存器低位（RTC_CNTL）

偏移地址：0x1C
复位值：0x0000

位/域	名称	R/W	描述
15:0	CNTL[15:0]	R/W	设置 RTC 计数器低位（RTC Counter Low Setup）

19.6.6 RTC 闹钟值寄存器（RTC_ALR）

当 OCFLG 值为 1 时允许执行写操作。

RTC 闹钟值寄存器高位（RTC_ALRH）

偏移地址：0x20

复位值：0xFFFF

位/域	名称	R/W	描述
15:0	ALRH[31:16]	W	设置 RTC 闹钟值高位（RTC Alarm Value High Setup）

RTC 闹钟值寄存器低位（RTC_ALRL）

偏移地址：0x24

复位值：0xFFFF

位/域	名称	R/W	描述
15:0	ALRL[15:0]	W	设置 RTC 闹钟值低位（RTC Alarm Value Low Setup）

20 通用同步异步收发器（USART）

20.1 术语全称、缩写描述

表格 82 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
过载	Overrun	OVR

20.2 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，也支持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。

20.3 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：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） 独立的发送器和接收器使能位

（7） 可编程的波特率发生器，波特率最高可达 4.5Mbits/s

（8） 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

（9） 同步传输模式

（10） LIN 断开帧的生成与检测

（11） 支持 ISO7816-3 标准的智能卡接口

（12） 支持 IrDA 协议

（13） 支持硬件流控制

（14） 可利用 DMA 连续通信

（15） 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

（16） 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

20.4 功能描述

表格 83 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	硬件流控制模式中请求发送
USART_nCTS	输出	硬件流控制模式中清除发送
IrDA_RDI	输入	IrDA 模式中的数据输入
IrDA_TDO	输出	IrDA 模式中的数据输出

20.4.1 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 后才可开启使能接收。
- 若总线上出现数据冲突，需要使用软件管理分配通信过程。

20.4.2 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 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 84 帧格式

DBLCFG 位	PCEN 位	USART 数据帧
0	0	起始位+8 位数据+停止位
0	1	起始位+7 位数据+奇偶检验位+停止位
1	0	起始位+9 位数据+停止位
1	1	起始位+8 位数据+奇偶检验位+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。

20.4.3 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。

20.4.3.1 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_DATA 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

- 置位 USART_CTRL1 寄存器的 UEN 位，使能 USART。
- 通过设置 USART_CTRL1 寄存器的 DBLCFG 位来决定字长。
- 通过设置 USART_CTRL2 寄存器的 STOPCFG 位来决定停止位位数。
- 若选择多缓冲器通信，需在 USART_CTRL3 寄存器中使能 DMA。
- 在 USART_BR 寄存器中设置通信的波特率。
- 使能 USART_CTRL1 寄存器的 TXEN 位，发送一个空闲帧。
- 等待 USART_STS 寄存器的 TXBEFLG 位置 1。
- 向 USART_DATA 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。

20.4.3.2 单字节通信

对 USART_DATA 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，数据发送就开始了，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 DATA 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_DATA 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。

20.4.3.3 断开帧

断开帧被视为在一个帧周期内全部收到 0。设置 USART_CTRL1 寄存器的 TXBF 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 TXBF 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 TXBF 位被复位。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：若在开始发送断开帧之前就复位了 TXBF 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 TXBF 位。

20.4.3.4 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。

20.4.4 接收器

20.4.4.1 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_DATA 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_DATA。

接收配置步骤

- 置位 USART_CTRL1 寄存器的 UEN 位，使能 USART。
- 通过设置 USART_CTRL1 寄存器的 DBLCFG 位来决定字长。
- 通过设置 USART_CTRL2 寄存器的 STOPCFG 位来决定停止位位数。
- 若选择多缓冲器通信，需在 USART_CTRL3 寄存器中使能 DMA。
- 在 USART_BR 寄存器中设置通信的波特率。
- 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_DATA 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_DATA 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。

20.4.4.2 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。

20.4.4.3 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。

20.4.4.4 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 DATA 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 DATA 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 USART_CTRL1 的 RXBNEIEN 位置 1，则会产生一个中断。
- 当 OVREFLG 位置 1 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 DATA 寄存器上，可以进行读操作。
  - 当 RXBNEFLG=0 时，DATA 寄存器中没有有效数据。
- 对 USART_STS 和 USART_DATA 寄存器依次进行读操作，可以复位 OVREFLG 位。

20.4.4.5 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 NE 标志。
- 无效数据从移位寄存器发送到 USART_DATA 寄存器。

20.4.4.6 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：

（1） 置位 USART_STS 寄存器的 FEFLG 位。

（2） 无效数据从移位寄存器发送到 USART_DATA 寄存器。

（3） 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。

20.4.5 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：

波特率＝PCLK/16×（USARTDIV）

USART2/3 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。

20.4.6 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，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:

- WUPMCFG 位被清除，总线空闲可退出静默模式。
- 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 RXMUTEEN 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 RXMUTEEN 位被硬件清 0。RXMUTEEN 还可以通过软件清 0。

图 67 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 68 地址标记退出静默模式

20.4.7 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 69 USART 同步传输的例子

图 70 USART 同步传输时序图（DBLCFG=0）

图 71 USART 同步传输时序图（DBLCFG=1）

20.4.8 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 LBDFLG 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。

图 72 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。

图 73 数据传输状态检测断开帧

20.4.9 智能卡模式

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。（为了避免在两种配置间转换，建议在发送和接收数据时使用 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 74 ISO7816-3 标准协议

20.4.10 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 75 IrDA 模式框图

20.4.11 硬件流控制

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 76 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。

20.4.12 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：

- 清零 USART_STS 寄存器的 TXCFLG 位。
- 把存放数据的 SRAM 存储器的地址设置为 DMA 源地址。
- 把 USART_DATA 寄存器的地址设置为 DMA 目的地址。
- 设置要传输的数据字节数。
- 设置通道优先级。

- 设置中断使能。
- 使能 DMA 通道。
- 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：

- 把 USART_DATA 寄存器的地址设置为 DMA 源地址。
- 把存放数据的 SRAM 存储器的地址设置为 DMA 目的地址。
- 设置要传输的数据字节数。
- 设置通道优先级。
- 设置中断使能。
- 使能 DMA 通道。

20.4.13 中断请求

表格 85 USART 中断请求

<table>
<tr><th colspan="2">中断事件</th><th>事件标志位</th><th>使能位</th></tr>
<tr><td colspan="2">接收寄存器不为空</td><td>RXBNEFLG</td><td rowspan="2">RXBNEIEN</td></tr>
<tr><td colspan="2">过载错误</td><td>OVREFLG</td></tr>
<tr><td colspan="2">检测到线路空闲</td><td>IDLEFLG</td><td>IDLEIEN</td></tr>
<tr><td colspan="2">奇偶检验错误</td><td>PEFLG</td><td>PEIEN</td></tr>
<tr><td colspan="2">LIN 断开帧标志</td><td>LBDFLG</td><td>LBDIEN</td></tr>
<tr><td rowspan="3">DMA 模式下的接收错误</td><td>噪音错误</td><td>NEFLG</td><td rowspan="3">ERRIEN</td></tr>
<tr><td>溢出错误</td><td>OVREFLG</td></tr>
<tr><td>帧错误</td><td>FEFLG</td></tr>
<tr><td colspan="2">数据发送寄存器空</td><td>TXBEFLG</td><td>TXBEIEN</td></tr>
<tr><td colspan="2">发送完成</td><td>TXCFLG</td><td>TXCIEN</td></tr>
<tr><td colspan="2">CTS 标志</td><td>CTSFLG</td><td>CTSIEN</td></tr>
</table>

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 77 USART 中断映射

20.4.14 USART 支持功能对比

表格 86 USART 支持功能对比

USART 模式	USART1	USART2	USART3	UART4	UART5
异步模式	√	√	√	√	√
硬件流控制	√	√	√	—	—
多缓存通讯（DMA）	√	√	√	√	—
多处理器通讯	√	√	√	√	√
同步	√	√	√	—	—
智能卡	√	√	√	—	—
半双工（单线模式）	√	√	√	√	√
IrDA	√	√	√	√	√
LIN	√	√	√	√	√

注意：“√”表示支持该功能，“—”表示不支持该功能。

20.5 寄存器地址映射

表格 87 USART 寄存器地址映射

寄存器名	描述	偏移地址
USART_STS	状态寄存器	0x00
USART_DATA	数据寄存器	0x04
USART_BR	波特率寄存器	0x08
USART_CTRL1	控制寄存器 1	0x0C

寄存器名	描述	偏移地址
USART_CTRL2	控制寄存器 2	0x10
USART_CTRL3	控制寄存器 3	0x14
USART_GTPSC	保护时间和预分频寄存器	0x18

20.6 寄存器功能描述

20.6.1 状态寄存器（USART_STS）

偏移地址：0x00

复位值：0x00C0

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：出现校验错误 在接收模式中，当出现奇偶校验错误时，由硬件置 1； 由软件清 0，需等 RXBNEFLG 置位后，先读取 USART_STS 寄存器，再读 USART_DATA 寄存器完成清 0。
1	FEFLG	R	发生帧错误标志（Frame Error Occur Flag） 0：无帧错误 1：出现帧错误或断开符 当出现同步错位、过多噪音或断开符时，由硬件置 1； 由软件清 0，先读取 USART_STS 寄存器，再读 USART_DATA 寄存器完成清 0。
2	NEFLG	R	发生噪音错误标志（Noise Error Occur Flag） 0：无噪音 1：出现噪音错误 当出现噪音错误时，由硬件置 1； 由软件清 0，先读取 USART_STS 寄存器，再读 USART_DATA 寄存器完成清 0。
3	OVREFLG	R	发生过载错误标志（Overrun Error Occur Flag） 0：过载出错误 1：出现过载错误 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； 由软件清 0，先读取 USART_STS 寄存器，再读 USART_DATA 寄存器完成清 0。
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； 由软件清 0，先读取 USART_STS 寄存器，再读 USART_DATA 寄存器完成清 0。

位/域	名称	R/W	描述
5	RXBNEFLG	RC_W0	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； 由软件清 0，读取 USART_DATA 成清 0，或者对该位写 0 清除。
6	TXCFLG	RC_W0	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； 由软件清 0，先读取 USART_STS 寄存器、再写 USART_DATA 寄存器完成清 0，或者对该位写 0 清除。
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； 由软件清 0，写 USART_DATA 寄存器完成清 0。
8	LBDFLG	RC_W0	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； 由软件清 0，对该位写 0 清除。
9	CTSFLG	RC_W0	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； 由软件清 0，对该位写 0 清除。
31:10	保留		

20.6.2 数据寄存器（USART_DATA）

偏移地址：0x04

复位值：0xXXXX XXXX，X=未定义位

位/域	名称	R/W	描述
8:0	DATA	R/W	数据值(Data Value) 发送数据值或接收数据值；接收数据时从中读取，发送数据时向该寄存器写入数据。 如果使能了奇偶校验位，如果是 9 个数据位，则 DATA 的第 8 位是校验位；如果是 8 个数据位，则 DATA 的第 7 位是奇偶校验位。
31:9	保留		

20.6.3 波特率寄存器（USART_BR）

偏移地址：0x08

复位值：0x0000

位/域	名称	R/W	描述
3:0	FBR[3:0]	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[15:4]	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		

20.6.4 控制寄存器 1（USART_CTRL1）

偏移地址：0x0C

复位值：0x0000

位/域	名称	R/W	描述
0	TXBF	R/W	发送断开帧（Transmit Break Frame） 0：未发送 1：将要发送 此位可由软件置位，在发送断开帧的停止位时，由硬件将其清 0。
1	RXMUTEEN	R/W	使能接收静默模式（Receive Mute Mode Enable） 0：正常工作模式 1：静默模式 此位由软件置位或清 0，或当检测到唤醒序列时，硬件将其清 0。 USART 要先接收一个数据才可置于静默模式，这样才能被空闲总线检测唤醒。 在地址标记检测唤醒中，若置位 RXBNEFLG 位，则 RXMUTEEN 位不能被软件修改。
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
3	TXEN	R/W	使能发送（Transmit Enable） 0：禁止 1：使能 除了在智能卡模式下，任何时候发送数据时，如果此位上有个 0 脉冲，那么在当前数据发送完成后，会发送一个空闲总线。 置位此位后，会延迟一个比特时间发送。
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
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
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
7	TXBEIEN	R/W	使能发送缓冲区空中断（Transmit Buffer Empty Interrupt Enable） 0：禁止产生中断 1：当 TXBEFLG 置位时，产生中断

位/域	名称	R/W	描述
8	PEIEN	R/W	使能校验错误中断（Parity Error Interrupt Enable） 0：禁止产生中断 1：当 PEFLG 置位时，产生中断
9	PCFG	R/W	配置奇偶校验位（Odd/Even Parity Configure） 0：偶校验 1：奇校验 需要等当前正在传输的字节完成后，选择才会生效。
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。
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
12	DBLCFG	R/W	配置数据位长度（Data Bits Length Configure） 0：1 个起始位，8 个数据位，n 个停止位 1：1 个起始位，9 个数据位，n 个停止位 传输数据时不能修改此位。
13	UEN	R/W	使能 USART（USART Enable） 0：禁用 USART 分频器和输出 1：使能 USART 模块
31:14	保留		

20.6.5 控制寄存器 2（USART_CTRL2）

偏移地址：0x10

复位值：0x0000 0000

位/域	名称	R/W	描述
3:0	ADDR[3:0]	R/W	设置 USART 设备节点地址（USART Device Node Address Setup） 此位只作用于多处理器通信下的静默模式，依据检测到的地址标记是否一致来决定进入静默模式还是唤醒。
4	保留		
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
6	LBDIEN	R/W	使能 LIN 断开符检测中断（LIN Break Detection Interrupt Enable） 0：禁止 1：当 LBDFLG 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；UART4 和 UART5 上不存在此位。

位/域	名称	R/W	描述
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；UART4 和 UART5 上不存在此位。
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，UART4 和 UART5 上不存在此位。
11	CLKEN	R/W	使能时钟（CK 引脚）（Clock Enable（CK pin）） 0：禁止 1：使能 UART4 和 UART5 上不存在此位。
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 UART4 和 UART5 上不存在此位。
14	LINMEN	R/W	使能 LIN 模式（LIN Mode Enable） 0：禁止 1：使能
31:15	保留		

注意：在使能发送后不能改写这三个位（CPOL、CPHA、LBCPOEN）。

20.6.6 控制寄存器 3（USART_CTRL3）

偏移地址：0x14

复位值：0x0000

位/域	名称	R/W	描述
0	ERRIEN	R/W	使能错误中断（Error Interrupt Enable） 0：禁止 1：使能，当 DMARXEN 置位并且 FEFLG、OVREFLG 或 NEFLG 其中一个置位时，产生中断。
1	IREN	R/W	使能红外功能（IrDA Function Enable） 0：禁止 1：使能
2	IRLPEN	R/W	使能红外低功耗模式（IrDA Low-power Mode Enable） 0：普通模式 1：低功耗模式
3	HDEN	R/W	使能半双工模式（Half-duplex Mode Enable） 0：禁止 1：使能

位/域	名称	R/W	描述
4	SCNACKEN	R/W	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） 0：不发送 NACK 1：发送 NACK UART4 和 UART5 上不存在此位。
5	SCEN	R/W	使能智能卡功能（Smartcard Function Enable） 0：禁止 1：使能 UART4 和 UART5 上不存在此位。
6	DMARXEN	R/W	使能 DMA 接收（DMA Receive Enable） 0：禁止 1：使能 UART4 和 UART5 上不存在此位。
7	DMATXEN	R/W	使能 DMA 发送（DMA Transmit Enable） 0：禁止 1：使能 UART4 和 UART5 上不存在此位。
8	RTSEN	R/W	使能 RTS 硬件流控制功能（RTS Hardware Flow Control Function Enable） 0：禁止 1：使能 RTS 中断 RTS：Require To Send 发送请求，为输出信号，说明已准备好接收。 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。 UART4 和 UART5 上不存在此位。
9	CTSEN	R/W	使能 CTS 硬件流控制功能（CTS Hardware Flow Control Function Enable） 0：禁止 1：使能 CTS：Clear To Send 发送清除，为输入信号 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。 UART4 和 UART5 上不存在此位。
10	CTSIEN	R/W	使能 CTS 中断（CTS Interrupt Enable） 0：禁止 1：CTSFLG 置位时产生中断 UART4 和 UART5 上不存在此位。
31:11	保留		

20.6.7 保护时间和预分频寄存器（USART_GTPSC）

偏移地址：0x18

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 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 UART4 和 UART5 上不存在此位。
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；UART4 和 UART5 上不存在此位。
31:16	保留		

21 内部集成电路接口（I2C）

21.1 术语全称、缩写描述

表格 88 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
数据包出错检测	Packet Error Checking	PEC

21.2 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

21.3 主要特征

（1） 多主机功能

（2） 主机可产生时钟、起始位和停止位

（3） 从机功能
- 可编程的 I2C 地址检测
- 双地址模式
- 检测停止位

（4） 7 位和 10 位寻址模式

（5） 响应广播

（6） 两种通信速度
- 标准模式

- 快速模式

（7） 可编程的时钟延长

（8） 状态标志
- 发送器/接收器模式标志
- 字节发送结束标志
- 总线忙标志

（9） 错误标志
- 仲裁丢失
- 应答错误
- 检测到错误的起始位或停止位

（10） 中断源
- 地址/数据通信成功
- 错误中断

（11） 支持 DMA 功能

（12） 可编程的 PEC
- 发送模式中最后传输
- 接收最后一个字节后进行 PEC 错误校验

（13） SMBus 特定功能
- 硬件 PEC
- 地址解析协议

21.4 结构框图

图 78 I2C 功能结构图

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。

21.5 功能描述

表格 89 I2C 总线专有名词说明

专有名词	说明
发送器	发送数据到总线的设备
接收器	从总线接收数据的设备
主机	初始化数据传输，产生时钟信号和结束数据传输的设备
从机	由主机寻址的设备
多主	不破坏信息的前提下同时控制总线的多个主机
同步	同步两个或更多设备之间的时钟信号的过程
仲裁	如果超过一个主机同时尝试控制总线时，只有一个主机可控制，且控制的主机信息不被破坏

21.5.1 I2C 物理层

I2C 通讯设备之间的常用连接方式如下图：

图 79 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。

21.5.2 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 总线还有起始信号,停止信号,应答信号。

- 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 80 主机写数据到从机

图 81 主机由从机读取数据

备注:

（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

（7） 0 为写入

（8） 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。

主机方向为写数据时

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（STOP），表示为传输完成。

主机方向为读数据时

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。

21.5.3 数据有效性

数据发送过程中，时钟信号 SCL 高电平期间，SDA 线上数据必须稳定，只有当 SCL 在低电平期间 SDA 的电平状态才能发生改变，每个数据比特传输都需要一个时钟脉冲。

图 82 SDA 时序图

21.5.4 起始和停止信号

所有数据传输都必定有起始信号（START）和停止信号（STOP）。

图 83 START 信号定义为：SCL 为高电平时，SDA 由高电平向低电平转换

图 84 STOP 信号定义为：SCL 为高电平时，SDA 由低电平向高电平转换

21.5.5 仲裁

仲裁也是为了解决多主机情况下的总线控制冲突。仲裁过程在主机发生，与从机无关。

只有在总线空闲时，主机才可以启动传输。两个主机可能在 START 信号的最短保持时间内在总线上产生一个有效的 START 信号。这种情况需要仲裁决定哪个主机完成传输。

仲裁是按位来进行的，在每一位仲裁期间，当 SCL 为高，每个主机都会检查 SDA 电平是否和自己发送的相同，总裁过程需要持续很多位。理论上，如果两个主机传输的内容完全相同，那他们能够成功传输而不发生仲裁失败。如果一个主机发送高电平，但是检测到 SDA 为低电平，则发生仲裁失败错误，关闭自身 SDA 输出，另一个主机完成自身传输。

图 85 SDA 时序图

注：Master1 仲裁失败

21.5.6 SMBus 特定功能

系统管理总线(System Management Bus，简称 SMBus)，结构简单的单端双线制总线，可实现轻量级的通信需求。

SMBus 常见于计算机主板，主要用于电源传输 ON/OFF 指令。SMBus 为 I2C 的衍生总线。主要用于计算机主板上的低带宽设备通信，与电源相关芯片。

地址解析协议

SMBus 规范中包含一个地址解析协议，可实现动态地址分配。动态识别硬件和软件使得总线能够支持热拔插，总线设备被自动识别并分配唯一地址。

SMBus 警报

SMBus 报警是一个带中断线的可选信号，用于那些希望扩展它们控制能力而牺牲的引脚。

21.5.7 错误标志位

表格 90 I2C 通信存在以下几种错误标志位

错误标志位	错误标志位说明
应答错误标志位(AEFLG)	没有收到应答
总线错误标志位(BERRFLG)	检测到一个外部的停止或起始条件
仲裁丢失标志位(ALFLG)	接口检测到仲裁丢失
过载/欠载错误标志位(OVRURFLG)	从模式下，接收数据未读出，下一个数据已经到达，发生过载错误。发送数据时钟已经到达，数据仍未写入 DATA 寄存器，发生欠载错误。
超时或 Tlow 错误标志位(TTEFLG)	SCL 拉低超过一定时间
PEC 比较错误标志位(PECEFLG)	CRC 值不相等

21.5.8 报文错误检验(PEC)

I2C 模块具有一个 PEC 模块，通过 CRC-8 计算器来进行对 I2C 数据的报文检

验，计算器使用的 CRC-8 多项式为：$C(x)= X^8+X^2+X+1$。

将 PECEN 位置 1，使能 PEC 功能，PEC 模块会把 I2C 总线发送的所有数据都计算，其中包括地址数据。

21.5.9 DMA 模式

根据 I2C 的软件流程，当发送寄存器为空或者接收寄存器满时，MCU 需要写入或者读取字节，此时我们通过 I2C 的 DMA 功能，能够更快速的完成该操作。

DMA 发送

设置 I2C_CTRL2 寄存器中的 DMAEN 位使能 DMA 模式，当发送寄存器为空时(TXBEFLG 置 1)，数据将通过 DMA 从存储区直接载入到 DATA 寄存器。

DMA 接收

设置 I2C_CTRL2 寄存器中的 DMAEN 位使能 DMA 模式，当接收寄存器满时（RXBNEFLG 置 1），DMA 将 DATA 寄存器数据传输到设定的存储区。

21.5.10 I2C 中断

表格 91 I2C 中断请求

中断事件	事件标志位	中断控制位
发送起始位完成	STARTFLG	EVIEN
发送完成/地址匹配地址信号	ADDRFLG	
10 位地址头段发送完成	ADDR10FLG	
接收到停止信号	STOPFLG	
数据字节传输完成	BTCFLG	
接收缓冲器非空	RXBNEFLG	EVIEN 和 BUFIEN
发送缓冲区空	TXBEFLG	
总线错误	BERRFLG	ERRIEN
仲裁丢失	ALFLG	
应答失败	AEFLG	
过载/欠载	OVRURFLG	
PEC 错误	PECEFLG	
超时或 Tlow 错误	TTEFLG	
SMBus 提醒	ALERTEN	

21.6 寄存器地址映射

表格 92 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_SADDR1	从机地址寄存器 1	0x08
I2C_SADDR2	从机地址寄存器 2	0x0C
I2C_DATA	数据寄存器	0x10
I2C_STS1	状态寄存器 1	0x14
I2C_STS2	状态寄存器 2	0x18
I2C_CLKCTRL	主机时钟控制寄存器	0x1C
I2C_RISETMAX	MRT 寄存器	0x20
I2C_SWITCH	切换寄存器	0x100

21.7 寄存器功能描述

21.7.1 控制寄存器 1（I2C_CTRL1）

偏移地址：0x00

复位值：0x0000

位/域	名称	R/W	描述
0	I2CEN	R/W	使能 I2C（I2C Enable） 0：禁止 1：使能
1	SMBEN	R/W	使能 SMBus 模式（SMBus Mode Enable） 0：I2C 模式 1：SMBus 模式
2	保留		
3	SMBTCFG	R/W	配置 SMBus 类型（SMBus Type Configure） 0：SMBus 设备 1：SMBus 主机
4	ARPEN	R/W	使能 ARP（ARP Enable） 0：禁止 1：使能 如果 SMBTCFG=0，使用 SMBus 设备的默认地址 如果 SMBTCFG=1，使用 SMBus 的主地址
5	PECEN	R/W	使能 PEC（PEC Enable） 0：禁止 1：使能

位/域	名称	R/W	描述
6	SRBEN	R/W	使能从机响应广播（Slave Responds Broadcast Enable） 0：禁止 1：使能 注意：广播地址是 0X00
7	CLKSTRETCHD	R/W	禁止从机模式时钟延长时间（Slave Mode Clock Stretching Disable） 0：使能 1：禁止 在从机模式下允许延长时钟低电平时间可避免发生过载和欠载错误。
8	START	R/W	发送起始位（Start Bit Transfer） 可软件置 1、清 0，当发送起始位或 I2CEN=0 时，由硬件清 0。 0：不发送 1：发送
9	STOP	R/W	发送停止位（Stop Bit Transfer） 可软件置 1、清 0；当发送停止位时，由硬件清 0；当检测到超时错误时，由硬件置 1。 0：不发送 1：发送
10	ACKEN	R/W	发送应答使能（Acknowledge Transfer Enable） 可软件置 1、清 0；当 I2CEN=0 时，由硬件清除。 0：不发送 1：发送
11	ACKPOS	R/W	配置接收数据应答/PEC 位置（Acknowledge /PEC Position Configure） 可软件置 1、清 0；当 I2CEN=0 时，由硬件清除。 0：接收当前字节时是否发送 NACK/ACK，PEC 是否处于移位寄存器中 1：接收下一个字节时是否发送 NACK/ACK，PEC 是否处于移位寄存器的下一字节
12	PEC	R/W	使能传输 PEC（Packet Error Check Transfer Enable） 可软件置 1、清 0；当传送 PEC 后，或发送起始位、停止位，或当 I2CEN=0 时，由硬件清 0。 0：禁止 1：使能
13	ALERTEN	R/W	使能 SMBus 提醒（SMBus Alert Enable） 可软件置 1、清 0；当 I2CEN=0 时，由硬件清 0。 0：释放 SMBAlert 引脚使其变高，提醒发送 NACK 信号后即时发送响应地址头 1：驱动 SMBAlert 引脚使其变低，提醒发送 ACKEN 信号后即时发送响应地址头
14	保留		
15	SWRST	R/W	软件配置 I2C 处于软件复位状态（Software Configure I2C under Reset State） 0：未复位 1：复位，在复位 I2C 前应确保 I2C 引脚被释放、总线是空闲状态。

21.7.2 控制寄存器 2（I2C_CTRL2）

偏移地址：0x04

复位值：0x0000

位/域	名称	R/W	描述
5:0	CLKFCFG	R/W	配置 I2C 时钟频率（I2C Clock Frequency Configure） 该时钟频率是指 I2C 模块的时钟，即从 APB 总线输入的时钟。 0：禁用 1：禁用 2：2MHz ... 50：50MHz 大于 100100：禁用。 I2C 总线最小的时钟频率：标准模式为 1MHz,快速模式为 4MHz。
7:6	保留		
8	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：SMBALTFLG、TTEFLG 、 PECEFLG 、 OVRURFLG 、 AEFLG 、 ALFLG 、 STS1_BERRFLG
9	EVIEN	R/W	使能事件中断（Event 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：STARTFLG、ADDRFLG、ADDR10FLG、STOPFLG、BTCFLG、TXBEFLG 置 1 且 BUFIEN 置 1、RXBNEFLG 置 1 且 BUFIEN 置 1。
10	BUFIEN	R/W	使能缓冲器中断（Buffer 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：TXBEFLG、RXBNEFLG
11	DMAEN	R/W	使能 DMA 请求（DMA Requests Enable） 0：禁止 1：当 TXBEFLG=1 或 RXBNEFLG=1 时，使能 DMA 请求
12	LTCFG	R/W	配置 DMA 最后一次传输（DMA Last Transfer Configure） 配置下一次 DMA 的 EOT 是否为接收的最后一次传输，只用于主机接收模式。 0：不是 1：是
15:13	保留		

21.7.3 从机模式地址寄存器 1（I2C_SADDR1）

偏移地址：0x08

复位值：0x0000

位/域	名称	R/W	描述
0	ADDR[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。

位/域	名称	R/W	描述
7:1	ADDR[7:1]	R/W	设置从机地址（Slave Address Setup） 从机地址的第 7:1 位
9:8	ADDR[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。
14:10	保留		
15	ADDRLEN	R/W	配置从机地址长度（Slave Address Length Configure） 0：7 位地址模式 1：10 位地址模式

21.7.4 从机模式地址寄存器 2（I2C_SADDR2）

偏移地址：0x0C

复位值：0x0000

位/域	名称	R/W	描述
0	ADDRNUM	R/W	配置从机地址数量（Slave Address Number Configure） 从机 7 位地址模式下，可配置为识别单地址模式、双地址模式；单地址模式下只识别 ADDR1；双地址模式下识别 ADDR1 和 ADDR2 在 7 位地址模式下可识别单个或双地址寄存器，具体情况如下： 0：识别 1 个地址（ADDR1） 1：识别 2 个地址（ADDR1 和 ADDR2）
7:1	ADDR2[7:1]	R/W	设置从机双地址模式地址（Slave Dual Address Mode Address Setup） 在双地址模式下地址的 7:1 位
15:8	保留		

21.7.5 数据寄存器（I2C_DATA）

偏移地址：0x10

复位值：0x0000

位/域	名称	R/W	描述
7:0	DATA	R/W	数据寄存器（Data Register） 在 I2C 发送模式下，将要发送的数据写到这个寄存器；在 I2C 接收模式下，从这个寄存器读取接收到的数据。
15:8	保留		

21.7.6 状态寄存器 1（I2C_STS1）

偏移地址：0x14

复位值：0x0000

位/域	名称	R/W	描述
0	STARTFLG	R	发送起始位完成标志（Start Bit Sent Finished Flag） 0：未发送 1：已发送 发出起始位时，由硬件置 1；软件先读 STS1 寄存器、再写 DATA 寄存器可清除该位；当 I2CEN=0 时由硬件清 0。

位/域	名称	R/W	描述
1	ADDRFLG	R	地址发送完成/接收匹配标志（Address Transfer Complete /Receive Match Flag） 从机模式是否接收到匹配地址： 0：未接收到 1：已接收 主机模式地址发送是否完成： 0：未完成 1：已完成 该位由硬件置 1；软件先读 STS1 寄存器、再读 STS2 寄存器可清除该位；当 I2CEN=0 时由硬件清 0。
2	BTCFLG	R	完成数据字节传输标志（Byte Transfer Complete Flag） 0：未完成 1：已完成 在接收数据时，如果未读取 DATA 寄存器中收到的数据，此时又收到一个新的数据时，由硬件置 1； 发送数据时，DATA 寄存器为空的情况下，将要发送移位寄存器中的数据时，由硬件置 1。 软件先读 STS1 寄存器、再对 DATA 寄存器执行读或写操作可清除该位；在传输中发送一个起始位、停止位，或当 I2CEN=0 时由硬件清 0。
3	ADDR10FLG	R	主机已发送 10 位地址的地址头标志（10-Bit Address Header Sent Flag） 0：未发送 1：已发送 该位由硬件置 1；软件先读 STS1 寄存器、再写 DATA 寄存器可清除该位；I2CEN=0 时由硬件清 0。
4	STOPFLG	R	停止位检测标志（Stop Bit Detection Flag） 0：未检测到 1：检测到 如果 ACKEN=1，在一个应答之后，当从机在总线上检测到停止位时，由硬件置 1； 软件读取 STS1 寄存器后、对 CTRL1 寄存器进行写操作可清除该位；当 I2CEN=0 时由硬件清 0。
5	保留		
6	RXBNEFLG	R	接收缓冲器不为空标志（Receive Buffer Not Empty Flag） 0：接收缓冲器为空 1：接收缓冲器不为空 当 DATA 寄存器有数据时，由硬件置 1； 当 BTCFLG 置 1 时，由于数据寄存器仍然为满，读取 DATA 寄存器不能清除 RXBNEFLG 位； 软件读写 DATA 寄存器可清除该位；当 I2CEN=0 时由硬件清 0。
7	TXBEFLG	R	发送缓冲器为空标志（Transmit Buffer Empty Flag） 0：发送缓冲器不为空 1：发送缓冲器为空 DATA 寄存器的内容为空时由硬件置 1； 软件写第 1 个数据到 DATA 寄存器时，会立刻将数据搬移到移位寄存器中，此时 DATA 寄存器的数据为空，无法清除该位；

位/域	名称	R/W	描述
			软件写数据到 DATA 寄存器可清除该位；发送起始位、停止位，或当 I2CEN=0 时由硬件清 0。
8	BERRFLG	RC_W0	总线错误标志（Bus Error Flag） 0：未发生总线错误 1：发生总线错误 总线错误是指起始位、停止位异常；检测到错误时由硬件置 1；软件写 0 可清除该位；I2CEN=0 时由硬件清 0。
9	ALFLG	RC_W0	主模式下的仲裁丢失标志（Master Mode Arbitration Lost Flag） 0：未发生仲裁丢失 1：发生仲裁丢失，I2C 接口自动切换回从模式 “主模式下的仲裁丢失”是指该主机失去对总线的控制；该位由硬件置 1；软件写 0 可清除该位；I2CEN=0 时由硬件清 0。
10	AEFLG	RC_W0	应答错误标志（Acknowledge Error Flag） 0：未发生应答错误 1：发生应答错误 由硬件置 1；软件写 0 可清除该位；I2CEN=0 时由硬件清 0。
11	OVRURFLG	RC_W0	发生过载或欠载标志（Overrun/Underrun Flag） 0：未发生 1：发生 CLKSTRETCHD=1、且满足以下条件之一时由硬件置 1： （1）从机接收模式下，DATA 寄存器中的数据未被读出时，又收到新的数据（该数据会丢失），此时发生过载； （2）从机发送模式中，DATA 寄存器没有写入数据，依然要发送数据（相同的数据会发送 2 次），此时发生欠载。 软件写 0 可清除该位；I2CEN=0 时由硬件清 0。
12	PECEFLG	RC_W0	接收时出现 PEC 错误标志（PEC Error in Reception Flag） 0：无 PEC 错误：在 ACKEN=1 的情况下接收到 PEC 后接收器返回 ACKEN 1：有 PEC 错误：不管 ACKEN 是什么值，只要接收到 PEC 后接收器都会返回 NACK 软件写 0 可清除该位；I2CEN=0 时由硬件清 0。
13	保留		
14	TTEFLG	RC_W0	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，从模式下，从机复位，总线被释放；主模式下，硬件发送停止位。 以下情况之一发生超时错误，由硬件置 1： （1）SCL 保持低电平大于 25ms； （2）主设备的 SCL 低电平扩展时间累计超过 10ms； （3）从设备的 SCL 低电平扩展时间累计超过 25ms。 软件写 0 可清除该位；I2CEN=0 时由硬件清 0。
15	SMBALTFLG	RC_W0	发生 SMBus 警报标志（SMBus Alert Occur Flag） 0：SMBus 主机模式，无警报； SMBus 从机模式，无警报，SMBAlert 引脚电平不变 1：SMBus 主机模式，在引脚上产生警报； SMBus 从机模式，收到警报，引起 SMBAlert 引脚电平变低 由硬件置 1；软件写 0 可清除该位；I2CEN=0 时由硬件清 0。

21.7.7 状态寄存器 2（I2C_STS2）

偏移地址：0x18

复位值：0x0000

位/域	名称	R/W	描述
0	MSFLG	R	主从模式标志（Master Slave Mode Flag） 0：从机模式 1：主机模式 配置 I2C 为主模式时，由硬件置 1； 以下条件之一硬件清 0： （1）产生停止位 （2）丢失总线仲裁 （3）I2CEN=0
1	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） SDA 或 SCL 为低电平时，由硬件置 1；产生停止位后，硬件清 0。
2	TRFLG	R	发送器模式/接收器模式标志（Transmitter / Receiver Mode Flag） 0：设备是接收器模式（读） 1：设备是发送器模式（写） 依据 R/W 位决定位的数值； 满足以下条件之一由硬件清 0： （1）产生停止位 （2）产生重复的起始位 （3）总线仲裁丢失 （4）I2CEN=0
3	保留		
4	GENCALLFLG	R	从模式接收到广播地址（0x00）标志（Slave Mode Received General Call Address Flag） 0：未收到广播地址 1：收到广播地址 由硬件置 1，满足以下条件之一由硬件清 0： （1）产生停止位 （2）产生重复的起始位 （3）I2CEN=0
5	SMBDADDRFLG	R	SMBus 设备从模式接收到默认地址标志（SMBus Device Received Default Address Flag in Slave Mode） 0：未收到默认地址 1：当 ARPEN=1 时，收到默认地址 由硬件置 1，满足以下条件之一由硬件清 0： （1）产生停止位 （2）产生重复的起始位 （3）I2CEN=0
6	SMMHADDR	R	SMBus 设备从模式收到主机头地址标志（SMBus Device Received Master Header Flag in Slave Mode） 0：未收到主机头地址 1：当同时满足 SMBTSEL=1 和 ARPEN=1 时，收到主机头地址 由硬件置 1，满足以下条件之一由硬件清 0：

位/域	名称	R/W	描述
			（1）产生停止位 （2）产生重复的起始位 （3）I2CEN=0
7	DUALADDRFLG	R	从模式接收到双地址匹配标志（Slave Mode Received Dual Address Match Flag） 0：接收到的地址与 ADDR1 寄存器的内容匹配 1：接收到的地址与 ADDR2 寄存器的内容匹配 由硬件置 1，满足以下条件之一由硬件清 0： （1）产生停止位 （2）产生重复的起始位 （3）I2CEN=0
15:8	PECVALUE	R	存储 PEC 值（Save Packet Error Checking Value） 当 PECEN=1 时，内部的 PEC 的值存放在 PECVALUE 里。

21.7.8 主机时钟控制寄存器（I2C_CLKCTRL）

偏移地址：0x1C

复位值：0x0000

位/域	名称	R/W	描述
11:0	CLKS [11:0]	R/W	设置主模式下快速/标准模式的时钟（Clock Setup in Fast/Standard Master Mode） 在 I2C 标准模式或 SMBus 模式中： T_{high}=CLKS × T_{PCLK1} T_{low}=CLKS × T_{PCLK1} I2C 快速模式中： 当 FDUTYCFG=0 时： T_{high}=CLKS×T_{PCLK1} T_{low}=2×CLKS× T_{PCLK1} 当 FDUTYCFG=1 时： T_{high}=9 × CLKS × T_{PCLK1} T_{low}=16 × CLKS × T_{PCLK1}
13:12	保留		
14	FDUTYCFG	R/W	配置快速模式下的占空比（Fast Mode Duty Cycle Configure） 此处定义占空比=t_{low}/t_{high} 0：SCLK 占空比 2 1：SCLK 占空比 16/9
15	SPEEDCFG	R/W	配置主模式速度（Master Mode Speed Configure） 0：标准模式 1：快速模式

21.7.9 最大上升时间寄存器（I2C_RISETMAX）

偏移地址：0x20

复位值：0x0002

位/域	名称	R/W	描述
5:0	RISETMAX	R/W	主模式下快速/标准模式的最大上升时间（Master Mode Maximum Rise Time in Fast/Standard Mode） 时间单位是 T_{PCLK1}，RISETMAX 为 SCL 最大上升时间单位加 1。

位/域	名称	R/W	描述
15:6	保留		

21.7.10 I2C 切换寄存器（I2C_SWITCH）

偏移地址：0x100

复位值：0x0000 0000

位/域	名称	R/W	描述
0	SWITCH	R/W	I2C 切换（I2C Switch） 0：I2C 使用 I2C1/I2C2 1：I2C 使用 I2C3
31:1	保留		

22 内部集成电路接口（I2C3）

22.1 简介

I2C 总线是一个双线串行接口，由串行数据线（SDA）和串行时钟（SCL）组成。这些电线在连接着总线的设备之间传输信息。每个设备都由一个唯一的地址识别，并且可以作为“发送器”或“接收器”工作，这取决于设备的功能。在执行数据传输时，设备也可以被视为主机或从机。主机是在总线上发起数据传送并生成时钟信号以允许该传送的设备。此时，任何被编址的设备都被认为是从机。

I2C 可以在标准模式（数据速率为 0 至 100 kb/s）、快速模式（数据速率小于或等于 400 kb/s）、快速模式+（数据速率小于或等于 1000 kb/s）、高速模式（数据速率小于或等于 3.4 Mb/s）下运行。

I2C 只能与这些模式的设备通信，只要它们连接到总线上。此外，高速模式和快速模式设备是向下兼容的。例如，在混合速度总线系统中，高速模式设备可以与快速模式和标准模式设备进行通信。在 0～100kb/s 的 I2C 总线系统中，快速模式设备可以与标准模式设备通信。

高速模式设备的一个例子是 LCD 显示器、高位计数 ADC 和大容量 EEPROM。这些设备通常需要传输大量数据。大多数维护和控制应用，即 I2C 总线的常见用途，通常以 100 kHz（在标准和快速模式下）运行。

22.2 功能描述

- 双线 I2C 串行接口-由串行数据线（SDA）和串行时钟（SCL）组成。
- 三种速度：
  - 标准模式（0 到 100 Kb/s）
  - 快速模式（≤ 400 Kb/s）或者快速模式+（≤ 1000 Kb/s）
  - 高速模式（≤ 3.4 Mb/s）
- 时钟同步
- 主或者从 I2C 操作
- 7 或 10 位编址
- 7 或 10 位合并格式传送
- 批量传输模式
- 忽略 CBUS 地址（用于共享 I2C 总线的 I2C 旧地址）
- 发送和接收缓冲器
- 处理所有总线速度下的位和字节等待
- 可编程 SDA 保持时间（$t_{HD:DAT}$）
- 总线清除特点
- 专用的发送 FIFO 和接收 FIFO，FIFO 深度为 8，FIFO 位宽为 8 位

22.2.1 I2C 协议

启动（START）和停止（STOP）条件

当总线空闲时，SCL 和 SDA 信号都通过总线上的外部上拉电阻拉高。当主机想要在总线上启动传输时，主机发出启动条件。这被定义为 SDA 信号的高-低转换，此时 SCL 为 1。当主机想要终止传输时，主机发出停止条件。这被定义为 SDA 线的低-高转换，此时 SCL 为 1。下图显示了启动和停止条件的记时。当数据在总线上传输时，当 SCL 为 1 时，SDA 线路必须稳定。

图 86 启动（START）和停止（STOP）条件

编址从机协议

两种地址格式：7 位地址格式和 10 位地址格式

7 位地址格式

在 7 位地址格式中，当位 0（R/W）设置为 0 时，主机写入从机。当位 0（R/W）设置为 1 时，主机从从机读取。

图 87 7 位地址格式

10 位地址格式

在 10 位地址格式中，需要两个字节传送 10 位地址。第一个字节的传输包含以下位定义。前五位（位 7:3）通知从机这是一个 10 位传输，随后是下两位（位 2:1）设置从机地址位 9:8，LSB 位（位 0）是 R/W 位。传输的第二个字节设置从机地址的位 7:0。

图 88 10 位地址格式

I2C 在首字节中的位定义

表格 93 I2C 位定义

从地址	R/W 位	描述
0000 000	0	广播呼叫地址. I2C 把数据放到接收缓冲区中并触发一个广播呼叫中断
0000 000	1	起始位
0000 001	X	CBUS 地址. I2C 忽略这些地址
0000 010	X	保留
0000 011	X	保留
0000 1xx	X	高速主机（主机）码
1111 1xx	X	保留
1111 0xx	X	10 位从地址.
0001 000	X	保留
0001 100	X	保留
1100 001	X	保留

发送和接收协议

主机作为主发送器或主接收器可以启动总线之间的数据传输和接收。从机响应来自主机对总线发送数据或从总线接收数据的请求。

主发送器和从接收器

所有数据以字节格式传输，每个数据传输的字节数不受限制。在主机发送地址和 R/W 位或主机向从机发送数据字节后，从机接收器必须用确认信号（ACK）响应。当从属接收器不响应 ACK 脉冲时，主机通过发出停止条件中止传输。从机必须将 SDA 线路保持在高位，以便主机可以中止传输。详细见下图。

图 89 主发送器和从接收器

主接收器和从发送器

如果主机正在接收如下图所示的数据，那么主机在接收到一个字节的数据后（除了最后一个字节），用一个确认脉冲响应从机发送器。当不发确认脉冲是主接收器通知从发送器时，这是最后一个字节的方式。从发送器在检测到无应答（NACK）后放弃 SDA 线路，以便主发送器可以发出停止（STOP）条件。

图 90 主接收器协议

从发送器在检测到无应答（NACK）后放弃 SDA 线路，以便主发送器可以发出停止条件。这与启动条件相同，只是发生在 ACK 脉冲之后。在主模式下运行时，I2C 可以使用不同方向的传输与同一从机通信。

TXFIFO 管理和启动（START）,停止（STOP）和重启（RSTAEN）

每当 TXFIFO 变空时，该组件在总线上生成一个停止（STOP）点。如果启用了重启功能，则当 TXFIFO 命令中的传输方向从读变为写时，组件将生成重新启动，反之亦然。如果未启用重新启动，则在此情况下会生成一个停止，然后再生成一个启动。

如下时序图，说明了当 TXFIFO 在作为主发送器运行时变为空时 I2C 的行为。

图 91 “TXFIFO 在作为主发送器”I2C 时序图

如下时序图，该图说明了当 TXFIFO 在作为主接收器工作时变为空时 I2C 的行为。

图 92 “TXFIFO 在作为主接收器”I2C 时序图

22.2.2 多主机仲裁

I2C 总线协议允许多个主机驻留在同一总线上。如果同一个 I2C 总线上有两个主机，如果两个主机同时试图通过同时生成启动条件来控制总线，则存在仲裁程序。

一旦一个主控器（例如微控制器）控制了总线，在第一个主控器发送停止条件并将总线置于空闲状态之前，没有其它主控器可以控制总线。

仲裁在 SDA 线上进行，而 SCL 线为 1。当另一个主服务器传输 0 时，主服务器传输 1，丢失仲裁并关闭其数据输出阶段。丢失仲裁的主机可以继续生成时钟，直到字节传输结束。如果两个主机都在处理同一个从机，仲裁可能进入数据阶段。

一旦检测到它已经失去了与另一个主机的仲裁，I2C 将停止生成 SCL（IC_CLK_oe）。

图 93 两个主控器在总线上仲裁的时序

对于高速模式，因为每个主机都用一个唯一的高速主代码编程，仲裁不能进入数据阶段。这个 8 位代码由系统设计者定义，并通过写入高速主模式代码地址寄存器 HSMC 进行设置。由于这些码是唯一的，只有一个主码才能赢得仲裁，这发生在高速主码传输结束时。总线的控制由地址或主代码以及竞争主机发送的数据决定，因此总线上既没有中央主机，也没有任何优先级顺序。

重启条件和数据位，停止条件和数据位和重启条件和停止条件这些情况之间不允许仲裁。

22.2.3 时钟同步

当两个或两个以上的主机试图同时在总线上传输信息时，它们必须仲裁并同步 SCL 时钟。数据仅在 SCL 时钟的高周期内有效。时钟同步是使用有线和连接到 SCL 信号来执行的。当主机将 SCL 时钟转换为 0 时，主机开始计数 SCL 时钟的低时间，并在下一个时钟周期开始时将 SCL 时钟信号转换为 1。但是，如果另一个主机将 SCL 线保持为 0，则主机将进入高等待状态，直到 SCL 时钟行转换为 1。

然后，所有主机都报出其高时间，高时间最短的主机将 SCL 线转换为 0。然后，主机计算出其低时间，而低时间最长的一个则迫使另一主机进入高等待状态。因此，会生成一个同步的 SCL 时钟。或者，从机可以将 SCL 线路保持在较低的位置，以减慢 I2C 总线上的计时。

22.2.4 IC_CLK 频率配置

当 I2C 配置为标准（SS）、快速（FS）/快速模式+（FM+）或高速（HS）主控时，必须在任何 I2C 总线业务发生之前设置寄存器，以确保正确的 I/O 时序。

不同模式下的寄存器如下表：

表格 94 不同模式下的寄存器

模式	寄存器
标准（SS）模式	SSCHC SSCLC
快速（FS）/快速模式加（FM+）	HSCHC HSCLC
高速（HS）	HSCHC HSCLC

SS、FS、FM+和 HS 模式下的最小高和低计数

当 I2C 作为 I2C 主机工作时，在发送和接收传输中：

（4） SSCLC 和 FSCLC 寄存器值必须大于 LSSSL+7。

（5） SSCHC 和 FSCHC 寄存器值必须大于 LSSSL+5.

（6） 如果这个组件被编程以支持 HS，HSCLC 寄存器值必须大于 HSSSL+7。

（7） 如果这个组件被编程以支持 HS，HSCHC 寄存器值必须大于 HSSSL+5。

有关 I2C 高和低计数的详细信息如下：

（8） *_LCNT 寄存器的 IC_*_SPKLEN+7 最小值是由于 I2C 在 SCL 的负边缘之后驱动 SDA 所需的时间。

（9） *_HCNT 寄存器的 IC_*_SPKLEN+5 最小值是由于 I2C 在 SCL 的高周期时期采集 SDA 样本所需的时间。

（10） I2C 将一个循环添加到编程的*_LCNT 值，以生成 SCL 时钟的低周期；这是由于 SCL 低计数的计数逻辑到（*_LCNT+1）。

（11） I2C 将 IC_*_SPKLEN+7 循环添加到编程的*_HCNT 值，以生成 SCL 时钟的高周期；这是由于以下因素。

- SCL 高计数的计数逻辑到（*_HCNT+1）。
- 应用于 SCL 线路的数字滤波会导致 spklen+2IC_clk 周期的延迟，
- 该滤波包括亚稳态去除和 SDA 和 SCL 边缘的可编程峰值抑制。
- 每当 I2C 驱动 SCL 从 1 到 0 时，即完成 SCL 高时间，会产生三个 IC_CLK 周期的内部逻辑延迟。因此，I2C 能够达到的最小 SCL 低时间为 9 个 IC_CLK 周期（7+1+1），而最小 SCL 高时间为 13 个 IC_CLK 周期（6+1+3+3）。

注：I2C 主机生成的 SCL 的总高时间和低时间也受 SCL 线的上升时间和下降时间的影响，如图和方程式所示。应注意，SCL 上升和下降时间参数随外部因素而变化，如：

- IO 驱动器特性
- 上拉电阻值
- SCL 线上的总电容，等等

图 94 上升和下降时间参数配置图

22.2.5 操作模式

22.2.5.1 从机模式操作

初始配置

要将 I2C 用作从属设备，请执行以下步骤：

（1） 通过将“0”写入 CTRL1 寄存器的位 0 来禁用 I2C

（2） 写入 SLAADDR 寄存器（位 9:0）以设置从机地址。这是 I2C 响应的地址

（3） 写入 CTRL0 寄存器，以指定支持哪种类型的寻址（7 位或 10 位，通过第 3 位设置）通过将“0”写入位 6（IC_SLAVE_DISABLE）和将“0”写入位 0（MST），在仅从模式下启用 I2C。

（4） 通过在 CTRL1 寄存器的位 0 中写入“1”来启用 I2C。

单字节的从发送器操作

当总线上的另一个 I2C 主机寻址 I2C 并请求数据时，I2C 充当从属发送器，并发生以下步骤：

（1） 另一个 I2C 主机启动 I2C 传输，其地址与 I2C 的 SLAADDR 寄存器中的从机地址匹配。

（2） I2C 确认发送地址并识别传输方向，以表明其作为从属发送器。

（3） I2C 启用 RRIF 中断（RIS 寄存器的第 5 位），并保持 SCL 线低。它处于等待状态，直到软件响应。如果由于 INTEN [5]寄存器（RRIE 位字段）设置为 0，已屏蔽了 RRIF 中断，则建议使用硬件和/或软件定时程序指示 CPU 执行 RIS 寄存器的定期读取。

- 必须将指示 RIS [5]（RRIF 位字段）设置为 1 的读操作视为启用的 RRIF 中断的等效操作。
- 然后软件必须运作以满足 I2C 传输。
- 所用的定时间隔应为 I2C 能处理的最快 SCL 时钟周期的 10 倍。例如，对于 400 kb/s，计时间隔为 25μs。

（4） 如果在接收到读取请求之前，发送 FIFO 中还有任何数据，则 I2C 启用 Tx ABRT 中断（RIS 寄存器的位 6）以从发送 FIFO 中刷新旧数据。如果 Tx-ABRT 中断被屏蔽，由于 INTEN[6]寄存器（TAIE 位字段）被设置为 0，则建议重新使用计时程序（如上一步所描述）或类似程序读取 RIS 寄存器。

- 指示位 6（TAIF）被设置为 1 的读取必须被视为等同于所启用的 TAIF 中断。
- 软件无需进一步操作。
- 所用的定时间隔应类似于上一步骤中对 RIS[5]寄存器所述的时间间隔。

（5） 软件将数据写入 DATA 寄存器（通过将‘0’写入位 8）。

（6） 在程序进行前，软件必须清除 RIS 寄存器的 RRIF 和 TAIF 中断（分别为第 5 位和第 6 位）。如果 RRIF 和/或 TAIF 中断被屏蔽，那么当 RRIF 或 TAIF 位被读取为 1 时，RIS 寄存器的清除将已经执行。

（7） I2C 释放 SCL 并传输字节。

（8） 主机可以通过发出重新启动（RSTAEN）条件保留 I2C 主线，或发出停止（STOP）条件释放主线。

单字节从机接收器操作

当主线上的另一个 I2C 主机编址 I2C 并发送数据时，I2C 充当从机接收器，执行以下步骤：

（1） 另一个 I2C 主机用 SLAADDR 寄存器中匹配 I2C 从编址的编址启动一个 I2C 传输。

（2） I2C 确认发送的地址并识别传输的方向，以表明 I2C 充当从机接收器。

（3） I2C 接收传输的字节并将其放置在接收缓冲区。

（4） I2C 启用 RFFIF 中断（RIS[2]寄存器）由于将 INTEN[2]寄存器设置为 0 或将 TFT 设置为大于 0 的值，如果 RFFIF 中断已被屏蔽，那么建议为 STS1 寄存器的定期读取实行一个定时程序。位 3（RFNEF）设置为 1 的 STS1 寄存器的读取操作必须被软件视作与所启用的 RFFIF 中断的等效操作。

（5） 软件可以从 DATA 寄存器读取字节（位 7:0）。

（6） 另一个主机可以通过发送重新启动（RSTAEN）条件保留 I2C 主线，或发送停止（STOP）条件释放主线。

批量传输的从机传输操作

在标准的 I2C 协议中，所有传输都是单字节传输的，通过将一个字节写入从机的 FIFO 来响应远程主机读取请求。当从机（从发送器）响应来自远程主机（主接收器）的读请求时，从发送器的 TXFIFO 中应该要放置至少一个条目。I2C 被用来处理 TXFIFO 中更多的数据，以便后续的读请求可以在不引发中断的情况下获取更多的数据。

最终，这消除了在每次 TXFIFO 中仅放置一个条目时，在引发数据中断之间发生重大延迟的可能性。

这种模式只在 I2C 充当从发送器时才会发生。如果远程主机确认从机发送的数据，而从机的 TXFIFO 中没有数据，I2C 在发出读请求中断（RRIF）的同时将行保持在低位，并在数据发送到远程主机之前等待数据写入 TXFIFO。

如果 RRIF 中断被屏蔽，由于 ISTS 寄存器的位 5（RRIE）被设置为 0，那么建议使用一个定时程序来激活 RIS 寄存器的定期读取。将位 5（RRIF）恢复设置为 1 的 RIS 的读操作，必须将其视为与本节提到的 RRIF 中断的等效读操作。

RRIF 中断是在读取请求时引发的，与中断一样，在退出中断服务处理例程（ISR）时必须清除。ISR 允许您将一个字节或多个字节写入 TXFIFO。在向主机传输这些字节期间，如果主机确认最后一个字节，从机必须再次发出 RRIF，因为主机正在请求更多的数据。

如果事先知道远程主机正在请求一个 n 字节的数据包，那么当另一个主机编址 I2C 并请求数据时，可以用 n 个字节编写 TXFIFO，远程主机将它作为连续的数据流接收。例如，只要远程主机在确认发送的数据并且 TXFIFO 中有可用的数据，I2C 从机就会继续向远程主机发送数据。不需要将 SCL 行保持低位或再次发出 RRIF。

如果远程主机要从 I2C 接收 n 个字节，但是将大于 n 的字节数写入了 TXFIFO，当从机完成发送请求的 n 个字节时，主机将清除 TXFIFO 并忽略任何多余的字节。

I2C 生成一个传输中止（TAIF）事件，以指示 TXFIFO 在本例中的清除。在需要 ACK/NACK 时，如果接收到 NACK，远程主机就拥有全部所需数据。此时，从机的状态机发出标志，清除 TXFIFO 中的剩余数据。此标志被传输到 FIFO 所在的处理器总线时钟域，此时 TXFIFO 的内容被清除。

22.2.5.2 主模式操作

初始配置

作为 I2C 的主机，目标编址和编址格式在无需禁用 I2C 的情况下可以不断更改。而从机要求在对编址进行任何更改之前禁用组件。

（1） 在 CTRL1 寄存器的位 0 写入 0 禁用 I2C。

（2） 当设备为从机（位 3）时，写入 CTRL0 寄存器以设置从操作支持的最大速度模式（位 2:1），并指定 I2C 是否以 7/10 位寻址模式开始传输。

（3） 将要寻址的 I2C 设备的地址写入 TARADDR 寄存器。它还指示 I2C 将执行常规调用（General Call）还是起始字节（START BYTE）命令。I2C 主控发起的传输（7 位或 10 位寻址）所需的速度由 MAM 位字段（位 12）控制。

（4） 仅适用于高速模式传输。将 I2C 所需的主代码写入 HSMC 寄存器。主代码由程序员定义。

（5） 将 1 写入 CTRL1 寄存器的位 0 启用 I2C。

（6） 现在写入要发送到 DATA 寄存器的传输方向和数据。如果 DATA 寄存器是在 I2C 启用前编写的，则在 I2C 不启用时，数据和命令会随着缓冲区的清除而丢失。

22.2.6 尖峰抑制

I2C 包含可编程的峰值抑制逻辑，符合 I2C 总线规范对 SS/FS、HS 模式的要求。

此逻辑基于监视输入信号（SCL 和 SDA）的计数器。每个信号都有一个单独的计数器（SCL 和 SDA）。IC_CLK 周期的数量可以由用户编程，并且在计算时应该考虑 IC_CLK 的频率和相关的峰值长度规范。

每当输入信号改变其值时，每个计数器就会启动。根据输入信号的行为，会出现以下其中一种情景：

输入信号保持不变，直到计数器达到计数限值。当发生这种情况时，将用输入值更新信号的内部版本，并重置和停止计数器。在检测到输入信号的新更改之前，计数器不会重新启动。

在计数器达到计数限值之前，输入信号再次发生变化。当发生这种情况时，计数器被重置并停止，但是信号的内部版本没有更新。计数器保持停止状态，直到检测到输入信号上的新更改。

这些寄存器的宽度为 8 位，可以通过 APB 接口进行读写；但是，只有在禁用 I2C 时才能写入它们。可编程到这些寄存器中的最小值是 1；试图编写小于 1 的值会导致写入值 1。

22.3 寄存器地址映射

表格 95 I2C3 寄存器地址映射

寄存器名	描述	偏移地址
CTRL0	控制寄存器	0x00
TARADDR	目标地址寄存器	0x04
SLAADDR	从地址寄存器	0x08
HSMC	主地址寄存器	0x0C
DATA	数据命令寄存器	0x10

寄存器名	描述	偏移地址
SSCHC	标准速度 I2C 时钟的 SCL 高位计数寄存器	0x14
SSCLC	标准速度 I2C 时钟的 SCL 低位计数寄存器	0x18
FSCHC	快速模式或快速+模式的 I2C 时钟的 SCL 高位计数寄存器	0x1C
FSCLC	快速模式或快速+模式的 I2C 时钟的 SCL 低位计数寄存器	0x20
HSCHC	高速模式 I2C 时钟的 SCL 高位计数寄存器	0x24
HSCLC	高速模式 I2C 时钟的 SCL 低位计数寄存器	0x28
ISTS	中断状态寄存器	0x2C
INTEN	中断使能寄存器	0x30
RIS	原始中断状态寄存器	0x34
RFT	发送 FIFO 阈值寄存器	0x38
TFT	接收 FIFO 阈值寄存器	0x3C
ICF	清除组合和单独中断寄存器	0x40
RFUIC	清除 RFUIF 中断寄存器	0x44
RFOIC	清除 RFOIF 中断寄存器	0x48
TFOIC	清除 TFOIF 中断寄存器	0x4C
RRIC	清除 RRIF 中断寄存器	0x50
TAIC	清除 TAIF 中断寄存器	0x54
RDIC	清除 RDIF 中断寄存器	0x58
AIC	清除 ACTIF 中断寄存器	0x5C
STPDIC	清除 STPDIF 中断寄存器	0x60
STADIC	清除 STADIF 中断寄存器	0x64
GCIC	清除 GCIF 中断寄存器	0x68
CTRL1	启动寄存器	0x6C
STS1	状态寄存器	0x70
TFL	发送 FIFO 存量寄存器	0x74
RFL	接收 FIFO 存量计数器	0x78
SDAHOLD	SDA 维持时长寄存器	0x7C
TAS	发送中止源头寄存器	0x80
SDNO	生成从机数据 NACK 寄存器	0x84
DMACTRL	DMA 控制寄存器	0x88
DTDL	DMA 发送数据电平寄存器	0x8C

寄存器名	描述	偏移地址
DRDL	DMA 接收数据电平寄存器	0x90
SDADLY	SDA 设置寄存器	0x94
GCA	ACK 通用呼叫寄存器	0x98
STS2	启用状态寄存器	0x9C
LSSSL	SS、FS 或者 FM+尖峰抑制限制寄存器	0xA0
HSSSL	HS 尖峰抑制限制寄存器	0xA4

22.4 寄存器功能描述

22.4.1 控制寄存器（CTRL0）

偏移地址：0x00

复位值：0x3E

位/域	名称	R/W	描述
0	MST	R/W	主模式控制器 这个位是否控制 I2C 主机 1：启用主模式 0：禁用主模式
2:1	SPEED	R/W	I2C 操作速度 这些位控制 I2C 操作的速度;只有在主模式下操作 I2C 时才能设置。硬件可以防止由软件编写的非法值。这些位必须适当地为从模式编程，因为它是用于根据速度模式捕获正确的窄带滤波器值。该寄存器的编程值只能在 1 到 3 之间;否则，硬件将此寄存器的值更新为 3。 01：标准模式（100kbit/s） 10：快速模式（<=400kbit/s）或快速模式+（<=1000Kbit/s） 11：高速模式（3.4Mbit/s）
3	SAM	R/W	选择寻址模式 0：7 位寻址模式 1：10 位寻址模式
4	保留		
5	RSTAEN	R/W	允许发送重复起始位 0：不支持重复起始位发送 1：支持重复起始位发送 确定是否可以在充当主机时发送 RSTAEN。一些较老的从机不支持处理 RSTAEN;但是，在一些 I2C 操作中使用了 RSTAEN。禁用重新启动时，禁止主程序执行以下功能： 发送一个起始字节（START BYTE） 执行任何高速模式操作 高速模式操作 以组合形式模式执行方向更改 使用 10 位地址执行读取操作 通过用在停止和后跟 START 替换 RSTAEN，操作被分解为多个 I2C 传输。如果执行上述操作，将导致设置 RIS 寄存器的位 6（TAIF）。

位/域	名称	R/W	描述
6	SLADIS	R/W	使能从机模式 0：从模式被启用 1：从模式被禁用 这个位控制 I2C 是否禁用了它的从机。在应用重置之后，您可以选择启用或禁用从机，这意味着软件不必配置从机。默认情况下，从机总是启用的（也处于重置状态）。如果您需要在重置后禁用它，请将此位设置为 1。如果设置了这个位（从机禁用），I2C 只作为主机运行，不执行任何需要从机的操作。
7	DSA	R/W	在从模式下： 1：（启用 ENABLED）：只要被编址，从机即发出 STPDIF 中断 0：（禁用 DISABLED）：从机总是发出 STPDIF 中断 注意:在常规调用寻址期间，如果 DSA=1b1，则此从机不会发出 STPDIF 中断，即使该从机通过生成 ACK 来响应常规调用寻址。STPDIF 中断仅在传输地址与从地址（SAR）匹配时生成。
8	TFEIC	R/W	这个位控制 TFEIF 中断的生成，如 RIS 寄存器中所述。 1（启用 ENABLED）：TFEIF 中断的控制生成 0（禁用 DISABLED）：TFEIF 中断的默认行为
9	RFFIE	R	该位控制当 RXFIFO 已满时 I2C 是否保持总线 1：（启用 ENABLED）：当 RX_FIFO 满时保持总线 0：（禁用 DISABLED）：当 RX_FIFO 满时溢出
10	DSMA	R	在主模式下： 1：（启用 ENABLED）：只有当主机运作时启用 STPDIF 中断 0：（禁用 DISABLED）：不论主机是否运作，启用 STPDIF 中断
31:11	保留		

22.4.2 目标地址寄存器（TARADDR）

偏移地址：0x4

复位值：0x1055

位/域	名称	R/W	描述
8:0	ADDR	R/W	这是任何主机事务的目标地址。 在 GENERAL_CALL 时，这些位会被忽略。如果要生成 START_BYTE，CPU 只需要对这些位写入一次。如果 ADDR 和 SAR 相同，则存在自己向自己发送，但 FIFOs 在主从机之间共享，不允许自己向自己发送。只支持单工，不支持双工模式。主机不能向自己传送信息;它只能传输给一个从机。
10:9	STA	R/W	如果将 GCEN 位设置为 1，则该位指示 I2C 是执行广播还是起始字节命令。 1：START_BYTE：起始字节传输 0：GENERAL_CALL：广播字节传输
11	GCEN	R/W	这个位指示软件是否执行设备 ID，或常规调用（General Call），或起始字节（START BYTE）命令。 1：（启用 ENABLED）：启用 GENERAL_CALL 或 START_BYTE 传输的编程 0：（禁用 DISABLED）：禁用 GENERAL_CALL 或 START_BYTE 传输的编程

位/域	名称	R/W	描述
12	MAM	R/W	当 I2C 作为主机时，这个位配置是以 7 位或 10 位寻址模式开始它的传输。 1：（ADDR_10BITS）：寻址 10 位传输格式 0：（ADDR_7BITS）：寻址 7 位传输格式
31:13	保留		

22.4.3 从地址寄存器（SLAADDR）

偏移地址：0x8

复位值：0x055

位/域	名称	R/W	描述
9:0	SAR	R/W	当 I2C 作为从机运行时，SLAADDR 保存从机地址。对于 7 位寻址，只使用 SLAADDR[6:0]。此寄存器只能在禁用 I2C 接口时编写，该接口对应于设置为 0 的 CTRL1[0]寄存器。其它情况下写入无效。
31:10	保留		

22.4.4 主地址寄存器（HSMC）

偏移地址：0xC

复位值：0x7

位/域	名称	R/W	描述
2:0	HSMC	R/W	此位字段保存 I2C HS 模式主代码的值。HS 模式主码是保留的 8 位码（00001xxx），不用于从寻址或其它目的。每个主机都有其惟一的主代码;在同一个 I2C 总线系统上，可以提供多达 8 个高速模式主机。有效值从 0 到 7。此寄存器消失并返回 0，变为只读。此寄存器只能在禁用 I2C 接口时编写，该接口对应于设置为 0 的 CTRL1[0]寄存器。其它情况下写入无效。
31:3	保留		

22.4.5 数据命令寄存器（DATA）

偏移地址：0x10

复位值：0x00

位/域	名称	R/W	描述
7:0	DATA	R/W	这个寄存器包含要在 I2C 总线上传输或接收的数据。在写入此位时读取该位无效。读取该位时，这些位返回的值是 I2C 接口上接收到的数据。
8	CMD	W	该位控制是执行读操作还是执行写操作。 当 I2C 充当从机时，此位不会控制方向。当它作为主机时，控制方向。 当在 TXFIFO 中输入一个命令时，这个位将区分写和读命令。在从接收模式中，这个位是无用。在从发送模式下，“0”表示要传输 DATA 中的数据。在编写这个位时，您应该记住以下内容:在发送常规调用命令之后试图执行一个读操作，会导致产生 TAIF 中断（RIS 寄存器的位 6），除非 TARADDR 寄存器的位 11（GCEN）已被清除。如果在接收到 RRIF 中断后将“1”写入该位，则会发生 TAIF 中断。 0：序列数据字节接收，主写入命令 1：主读取命令
9	STOP	R/W	该位控制数据停止传输或接收。
31:10	保留		

22.4.6 标准速度 I2C 时钟的 SCL 高位计数寄存器（SSCHC）

偏移地址：0x14

复位值：0x190

位/域	名称	R/W	描述
13:0	CNT	R/W	在进行任何 I2C 总线事务之前，必须设置此寄存器，以确保正确的输入/输出（I/O）时间。此寄存器设置标准速度下 SCL 时钟计数器的高 16 位。有关更多信息，请参阅“IC_CLK 频率配置”。 此寄存器只能在禁用 I2C 接口时编写（CTRL1[0]寄存器为 0）。其它情况下写入无效。最小有效值为 6； 硬件可以防止写入小于此值的值，如果试图设置值，则结果设置值为 6。对于 APB_DATA_WIDTH=16 的设计，编程顺序对于确保 I2C 的正确操作非常重要。必须先编程较低的字节。然后对上位字节进行编程。
31:14	保留		

22.4.7 标准速度 I2C 时钟 的 SCL 低位计数寄存器（SSCLC）

偏移地址：0x18

复位值：0x1D6

位/域	名称	R/W	描述
13:0	CNT	R/W	在进行任何 I2C 总线事务之前，必须设置此寄存器，以确保正确的输入/输出（I/O）时间。此寄存器设置标准速度下 SCL 时钟计数器的低 16 位。有关更多信息，请参阅“IC_CLK 频率配置”，此寄存器只能在禁用 I2C 接口时编写（CTRL1[0]寄存器为 0）。其它情况下写入无效。最小有效值为 8;硬件可以防止写入的值小于此值，如果试图设置值，则结果设置值为 8。对于 APB_DATA_WIDTH=16 的设计，编程顺序对于确保 I2C 的正确操作非常重要。必须先对下字节进行编程，然后对上字节进行编程。
31:14	保留		

22.4.8 快速模式或快速+模式的 I2C 时钟的 SCL 高位计数寄存器（FSCHC）

偏移地址：0x1C

复位值：0x3C

位/域	名称	R/W	描述
13:0	CNT	R/W	在进行任何 I2C 总线事务之前，必须设置此寄存器，以确保正确的输入/输出（I/O）时间。此寄存器设置快速模式或快速+模式下 SCL 时钟计数器的高 16 位。它在快速模式下用于发送主代码和 START BYTE 或 General CALL。有关更多信息，请参阅“IC_CLK 频率配置”。此寄存器复位时为 0，变为只读。此寄存器只能在禁用 I2C 接口时编写（CTRL1[0]寄存器为 0）。最小有效值为 6;硬件可以防止写入小于此值的值，如果试图设置值，则设置为 6。对于 APB_DATA_WIDTH=16 的设计，编程顺序对于确保 I2C 的正确操作非常重要。必须先编程较低的字节。然后对上位字节进行编程。
31:14	保留		

22.4.9 快速模式或快速+模式的 I2C 时钟的 SCL 低位计数寄存器(FSCLC)

偏移地址：0x20

复位值：0x82

位/域	名称	R/W	描述
13:0	CNT	R/W	在进行任何 I2C 总线事务之前，必须设置此寄存器，以确保正确的输入/输出（I/O）时间。此寄存器设置快速模式或快速+模式下 SCL 时钟计数器的高 16 位。它在快速模式下用于发送主代码和 START BYTE 或 General CALL。有关更多信息，请参阅“IC_CLK 频率配置”。此寄存器复位时为 0，变为只读。此寄存器只能在禁用 I2C 接口时编写（CTRL1[0]寄存器为 0），其它情况下写入无效。最小有效值为 8;硬件可以防止写入小于此值的值，如果试图设置值，则结果设置值为 8。对于 APB_DATA_WIDTH=16 的设计，编程顺序对于确保 I2C 的正确操作非常重要。必须先编程较低的字节。然后对上位字节进行编程。如果值小于 8，则计数值将更改为 8。
31:14	保留		

22.4.10 高速模式 I2C 时钟的 SCL 高位计数寄存器（HSCHC）

偏移地址：0x24

复位值：0x6

位/域	名称	R/W	描述
13:0	CNT	R/W	在进行任何 I2C 总线事务之前，必须设置此寄存器，以确保正确的输入/输出（I/O）时间。此寄存器设置高速模式下 SCL 时钟计数器的高 16 位。请参阅“IC_CLK 频率配置”。此寄存器设置取决于总线。100pF 加载时为 60ns; 400pF 加载时为 120ns。此寄存器复位时为 0，变为只读。此寄存器只能在禁用 I2C 接口时编写（CTRL1[0]寄存器为 0），其它情况下写入无效。最小有效值为 6;硬件可以防止写入小于此值的值，如果试图设置值，则最终设置值为 6。对于 APB_DATA_WIDTH=16 的设计，编程顺序对于确保 I2C 的正确操作非常重要。必须先编程较低的字节。然后对上位字节进行编程。
31:14	保留		

22.4.11 高速模式 I2C 时钟的 SCL 低位计数寄存器（HSCLC）

偏移地址：0x28

复位值：0x10

位/域	名称	R/W	描述
13:0	CNT	R/W	在进行任何 I2C 总线事务之前，必须设置此寄存器，以确保正确的输入/输出（I/O）时间。此寄存器设置高速模式下 SCL 时钟计数器的低 16 位。有关更多信息，请参阅 “IC_CLK 频率配置”。 此寄存器设置取决于总线，100pF 加载时为 160ns;对于 400pF 加载时为 320ns。此寄存器复位时为 0，变为只读。此寄存器只能在禁用 I2C 接口时编写（CTRL1[0]寄存器为 0），其它情况下写入无效。最小有效值为 8;硬件可以防止写入小于此值的值，如果试图设置值，则最终设置值为 8。对于 APB_DATA_WIDTH=16 的设计，编程顺序对于确保 I2C 的正确操作非常重要。必须先编程较低的字节。然后对上位字节进行编程。如果值小于 8，则计数值将更改为 8。
31:14	保留		

22.4.12 中断状态寄存器（INTSTS）

偏移地址：0x2C

复位值：0x00

位/域	名称	R/W	描述
0	RFUIF	R	有关 RFUIF 位的详细描述，请参见 RIS。 1：RFUIF 中断处于活动状态 0：RFUIF 中断处于不活动状态
1	RFOIF	R	有关 RFOIF 位的详细描述，请参见 RIS。 1：RFOIF 中断处于活动状态 0：RFOIF 中断处于不活动状态
2	RFFIF	R	有关 RFFIF 位的详细描述，请参见 RIS。 1：RFFIF 中断处于活动状态 0：RFFIF 中断处于不活动状态
3	TFOIF	R	有关 TFOIF 位的详细描述，请参见 RIS。 1：TFOIF 中断处于活动状态 0：TFOIF 中断处于不活动状态
4	TFEIF	R	有关 TFEIF 位的详细描述，请参见 RIS。 1：TFEIF 中断处于活动状态 0：TFEIF 中断处于不活动状态
5	RRIF	R	有关 RRIF 位的详细描述，请参见 RIS。 1：RRIF 中断处于活动状态 0：RRIF 中断处于不活动状态
6	TAIF	R	有关 TAIF 位的详细描述，请参见 RIS。 1：TAIF 中断处于活动状态 0：TAIF 中断处于不活动状态
7	RDIF	R	有关 RDIF 位的详细描述，请参见 RIS。 1：RDIF 中断处于活动状态 0：RDIF 中断处于不活动状态
8	ACTIF	R	有关 R_ACTIF 位的详细描述，请参见 RIS。 1：R_ACTIF 中断处于活动状态 0：R_ACTIF 中断处于不活动状态
9	STPDIF	R	有关 STPDIF 位的详细描述，请参见 RIS。 1：STPDIF 中断处于活动状态 0：STPDIF 中断处于不活动状态
10	STADIF	R	有关 STADIF 位的详细描述，请参见 RIS。 1：STADIF 中断处于活动状态 0：STADIF 中断处于不活动状态
11	GCIF	R	有关 GCIF 位的详细描述，请参见 RIS。 1：GCIF 中断处于活动状态 0：GCIF 中断处于不活动状态
12	RSTADIF	R	有关 RSTADIF 位的详细描述，请参见 RIS。 1：RSTADIF 中断处于活动状态 0：RSTADIF 中断处于不活动状态
13	MOHIF	R	有关 MOHIF 位的详细描述，请参见 RIS。 1：MOHIF 中断处于活动状态 0：MOHIF 中断处于不活动状态
31:14	保留		

22.4.13 中断使能寄存器（INTEN）

偏移地址：0x30

复位值：0x8FF

位/域	名称	R/W	描述
0	RFUIE	R/W	这个位在 ISTS 寄存器中使能 RFUIF 中断。 1：使能 0：禁止
1	RFOIE	R/W	这个位在 ISTS 寄存器中使能 RFOIF 中断。 1：使能 0：禁止
2	RFFIE	R/W	这个位在 ISTS 寄存器中使能 RFFIF 中断。 1：使能 0：禁止
3	TFOIE	R/W	这个位在 ISTS 寄存器中使能 TFOIF 中断。 1：使能 0：禁止
4	TFEIE	R/W	这个位在 ISTS 寄存器中使能 TFEIF 中断。 1：使能 0：禁止
5	RRIE	R/W	这个位在 ISTS 寄存器中使能 RRIF 中断。 1：使能 0：禁止
6	TAIE	R/W	这个位在 ISTS 寄存器中使能 TAIF 中断。 1：使能 0：禁止
7	RDIE	R/W	这个位在 ISTS 寄存器中使能 RDIF 中断。 1：使能 0：禁止
8	ACTIE	R/W	这个位在 ISTS 寄存器中使能 R_ACTIF 中断。 1：使能 0：禁止
9	STPDIE	R/W	这个位在 ISTS 寄存器中使能 STPDIF 中断。 1：使能 0：禁止
10	STADIE	R/W	这个位在 ISTS 寄存器中使能 STADIF 中断。 1：使能 0：禁止
11	GCIE	R/W	这个位在 ISTS 寄存器中使能 GCIF 中断。 1：使能 0：禁止
31:12	保留		

22.4.14 原始中断状态寄存器（RIS）

偏移地址：0x34

复位值：0x00

位/域	名称	R/W	描述
0	RFUIF	R	在接收缓冲区为空，处理器试图从 DATA 寄存器读取接收缓冲区时设置。如果模块被禁用（CTRL1[0]=0），这个位保持它的级别，直到主状态机或从状态机进入空闲状态，当 I2CEN 变为 0 时，此中断被清除。 1：RFUIF 中断处于活动状态 0：RFUIF 中断处于不活动状态
1	RFOIF	R	如果接收缓冲区已完全填满，则设置到 IC_RX_BUFFER_DEPTH，并从外部 I2C 设备接收另一个字节。I2C 确认这一点，但 FIFO 满后接收的任何数据字节都将丢失。如果模块被禁用（CTRL1[0]=0），这个位保持它的级别，直到主状态机或从状态机进入空闲状态，当 I2CEN 变为 0 时，这个中断被清除。 1：RFOIF 中断处于活动状态 0：RFOIF 中断处于不活动状态
2	RFFIF	R	在接收缓冲区达到或超过 RFT 寄存器中的 RFT 阈值时设置。当缓冲区级别低于阈值时，硬件会自动清除它。如果模块被禁用（CTRL1[0]=0），则刷新 RX FIFO 并保存在重新设定中；因此，RX FIFO 未满。一旦 CTRL1 位 0 被编程为 0，这个位将被清除，而不管继续的活动是什么。 1：RFFIF 中断处于活动状态 0：RFFIF 中断处于不活动状态
3	TFOIF	R	如果传输缓冲区被填满到 16，并且处理器试图通过写入 DATA 寄存器来发出另一个 I2C 命令，则在传输期间设置。当模块被禁用时，这个位保持它的级别，直到主或从状态机进入闲置状态，当 I2CEN 变为 0 时，此中断被清除。 1：TFOIF 中断处于活动状态 0：TFOIF 中断处于不活动状态
4	TFEIF	R	根据 CTRL0 寄存器中的 TFEIC 选择，TFEIF 中断状态的行为是不同的。 当 TFEIC=0:当传输缓冲区位于或低于 TFT 寄存器中设置的阈值时，该位设置为 1。 当 TFEIC=1:当传输缓冲区位于或低于 TFT 寄存器中设置的阈值时，该位被设置为 1，并且完成了最近弹出命令的来自内部移位寄存器地址/数据的传输。 当缓冲区级别超过阈值时，硬件会自动清除它。 当 CTRL1[0]设置为 0 时，TXFIFO 被刷新并保存在重新设定（reset）中。这里的 TXFIFO 看起来没有数据，因此这个位被设置为 1，前提是在主状态机或从状态机中活动。当不再有任何活动，I2CEN=0 时，这个位被设置为 0。 1：TFEIF 中断处于活动状态 0：TFEIF 中断处于不活动状态
5	RRIF	R	当 I2C 充当从服务器而另一个 I2C 主机试图从 I2C 读取数据时，该位被设置为 1。I2C 将 I2C 总线保持在等待状态（SCL=0），直到这个中断得到服务，这意味着从机已经由请求传输数据的远程主机寻址。处理器必须响应这个中断，然后将请求的数据写入 DATA 寄存器。在处理器读取 IC_CLRRIF 寄存器之后，这个位被设置为 0。 1：RRIF 中断处于活动状态 0：RRIF 中断处于不活动状态

位/域	名称	R/W	描述
6	TAIF	R	这个位表示如果 I2C 作为一个 I2C 发送器，不能完成对发送 FIFO 内容的预期操作。这种情况既可以发生在 I2C 主机上，也可以发生在 I2C 从机上，称为“传输中止”。 当这个位设置为 1 时，TAS 寄存器指示传输中止发生的原因。 注意：当 TAS 寄存器跟踪的任何事件导致传输中止时，I2C 只刷新/重置/清空 TX_FIFO。在读取寄存器 IC_CLTAIF 之前，TXFIFO 保持这种刷新状态。执行此读取之后，TXFIFO 就可以接受来自 APB 接口的更多数据字节。 1：TAIF 中断处于活动状态 0：TAIF 中断处于不活动状态
7	RDIF	R	这个位掩码 ISTS 寄存器中的 RDIF 中断。 1：RDIF 中断被解除屏蔽 0：RDIF 中断被屏蔽
8	ACTIF	R	这个位捕获 I2C 活动并保持设置，直到它被清除。有四种方法可以清除它： 禁用 I2C； 读取 AIC 寄存器； 读取 ICF 寄存器； 系统复位。 一旦这个位被设置，它就会保持设置，除非使用这四个方法之一来清除它。即使 I2C 模块是闲置的，这个位直到清除为止仍然保持设置，这表明总线上有活动。 1：RAW_INTR_ACTIF 中断处于活动状态 0：RAW_INTR_ACTIF 中断处于不活动状态 当 I2C 充当从发送器时，如果主机不确认传输的字节，则将该位设置为 1。这发生在传输的最后一个字节，表示传输已经完成。 1：RDIF 中断处于活动状态 0：RDIF 中断处于不活动状态
9	STPDIF	R	指示是否在 I2C 接口上发生了停止（STOP）条件，不管 I2C 是在从模式还是主模式下运行。 在从模式下： 如果 CTRL0 寄存器的 DSA=1b1，那么 STPDIF 中断只有在从机被寻址时发出。 注意：在常规调用寻址期间，如果 DSA=1b1，则此从机不会发出 STPDIF 中断，即使该从机通过生成 ACK 来响应常规调用寻址。 STPDIF 中断仅在传输地址与从地址（SAR）匹配时生成。 如果 CTRL0 寄存器的 DSA=1b0，则发出 STOP DET 中断，而不管它是否被寻址。 在主模式下： 如果 CTRL0 寄存器的 DSMA=1b1，STPDIF 中断只会在主机活动时发出。 如果 CTRL0 寄存器的 DSMA=1b0，则无论主机是否活动，都会发出 STPDIF 中断。 1：STPDIF 中断处于活动状态 0：STPDIF 中断处于不活动状态

位/域	名称	R/W	描述
10	STADIF	R	指示 I2C 接口上是否发生了启动（START）或重新启动（RSTAEN）条件，不管 I2C 是在从模式还是主模式下运行。 1：STADIF 中断处于活动状态 0：STADIF 中断处于不活动状态
11	GCIF	R	只在接收到常规调用寻址并确认该地址时设置。直到通过禁用 I2C 或当 CPU 读取 GCIC 寄存器的位 0 时清除它为止，都保持设置状态。I2C 将接收到的数据存储在 Rx 缓冲区中。 1：GCIF 中断处于活动状态 0：GCIF 中断处于不活动状态
12	RSTADIF	R	指示当 I2C 在从模式下操作且正在寻址从机时，是否会在 I2C 接口上发生了重新启动（RSTAEN）条件。仅当 IC_SLV_RSTADIF_EN=1 时启用。注意:但是，在高速模式下或在起始字节传输期间，根据 I2C 协议，在地址字段之前重新启动。在这种情况下，当重新启动发出时，从机不是寻址的从机，因此 I2C 不会生成 RSTADIF 中断。 1：RSTADIF 中断处于活动状态 0：RSTADIF 中断处于不活动状态
13	MOHIF	R	指示 master 是否持有总线，TXFIFO 是否为空。仅当 I2C_DYNAMIC_TAR_UPDATE=1 和 IC_EMPTYFIFO_HOLD_MASTER_EN=1 时启用。 1：MOHIF 中断是活动的 0：MOHIF 中断不活动
31:14	保留		

22.4.15 接收 FIFO 阈值寄存器（RFT）

偏移地址：0x38

复位值：0x00

位/域	名称	R/W	描述
7:0	RFT	R/W	接收 FIFO 阈值控制触发 RFFIF 中断（RIS 寄存器中的位 2）的数据位数，大于此值则触发中断。有效范围是 0-255，但是硬件不允许将该值设置为大于缓冲区深度的值。如果尝试这样做，实际的值将是缓冲区的最大深度。值 0 设置 1 个位的阈值，值 255 设置 256 个位的阈值。
31:8	保留		

22.4.16 发送 FIFO 阈值寄存器（TFT）

偏移地址：0x3C

复位值：0x00

位/域	名称	R/W	描述
7:0	TFT	R/W	发送 FIFO 阈值控制触发 TFEIF 中断（RIS 寄存器中的位 4）小于此值触发中断。有效的范围是 0-255，附加的限制是它不能设置为大于缓冲区深度的值。如果尝试这样做，实际的值将是缓冲区的最大深度。值 0 设置 0 个位的阈值，值 255 设置 255 位的阈值。
31:8	保留		

22.4.17 清除组合和单独中断寄存器（ICF）

偏移地址：0x40

复位值：0x00

位/域	名称	R/W	描述
0	ICF	R	读取此寄存器以清除组合中断、所有单独中断和 TAS 寄存器。这个位不清除硬件可清除的中断，但是清除软件可清除的中断。 有关清除 TAS 的异常，参考 TAS 寄存器的位 9。
31:1	保留		

22.4.18 清除 RFUIF 中断寄存器（RFUIC）

偏移地址：0x44

复位值：0x00

位/域	名称	R/W	描述
0	RFUIC	R	读取此寄存器以清除 RIS 寄存器的 RFUIF 中断（位 0）。
31:1	保留		

22.4.19 清除 RFOIF 中断寄存器（RFOIC）

偏移地址：0x48

复位值：0x00

位/域	名称	R/W	描述
0	RFOIC	R	读取此寄存器以清除 RIS 寄存器的 RFOIF 中断（位 1）。
31:1	保留		

22.4.20 清除 TFOIF 中断寄存器（TFOIC）

偏移地址：0x4C

复位值：0x00

位/域	名称	R/W	描述
0	TFOIC	R	读取此寄存器以清除 RIS 寄存器的 TFOIF 中断（位 3）。
31:1	保留		

22.4.21 清除 RRIF 中断寄存器（RRIC）

偏移地址：0x50

复位值：0x00

位/域	名称	R/W	描述
0	RRIC	R	读取此寄存器以清除 RIS 寄存器的 RRIF 中断（位 5）
31:1	保留		

22.4.22 清除 TAIF 中断寄存器（TAIC）

偏移地址：0x54

复位值：0x00

位/域	名称	R/W	描述
0	TAIC	R	读取此寄存器以清除 RIS 寄存器和 TAS 寄存器的 TAIF 中断（位 6）。这也会将 TXFIFO 从刷新/重置状态释放出来，从而允许对 TXFIFO 进行更多的写入。有关清除 TAS 的异常，请参阅 TAS 寄存器的 MSTDIS 位。
31:1	保留		

22.4.23 清除 RDIF 中断寄存器（RDIC）

偏移地址：0x58

复位值：0x00

位/域	名称	R/W	描述
0	RDIC	R	读取此寄存器以清除 RIS 寄存器的 RDIF 中断（位 7）。
31:1	保留		

22.4.24 清除 ACTIF 中断寄存器（AIC）

偏移地址：0x5C

复位值：0x00

位/域	名称	R/W	描述
0	AIC	R	如果 I2C 不再活动，则读取此寄存器清除 ACTIF 中断。如果 I2C 模块在总线上仍然处于活动状态，则继续设置 ACTIF 中断位。如果模块被禁用，并且总线上没有其它活动，硬件将自动清除该中断位。读取该寄存器用以获取 RIS 寄存器的 ACTIF 中断状态（位 8）的值。
31:1	保留		

22.4.25 清除 STPDIF 中断寄存器（STPDIC）

偏移地址：0x60

复位值：0x00

位/域	名称	R/W	描述
0	STPDIC	R	读取此寄存器以清除 RIS 寄存器的 STPDIF 中断（位 9）。
31:1	保留		

22.4.26 清除 STADIF 中断寄存器（STADIC）

偏移地址：0x64

复位值：0x00

位/域	名称	R/W	描述
0	STADIC	R	读取此寄存器以清除 RIS 寄存器的 STADIF 中断（位 10）。
31:1	保留		

22.4.27 清除 GCIF 中断寄存器（GCIC）

偏移地址：0x68

复位值：0x00

位/域	名称	R/W	描述
0	GCIC	R	读取此寄存器以清除 RIS 寄存器的 GCIF 中断（位 11）。
31:1	保留		

22.4.28 启动寄存器（CTRL1）

偏移地址：0x6c

复位值：0x00

位/域	名称	R/W	描述
0	I2CEN	R/W	控制是否启用 I2C 0：禁用 I2C（TX 和 RX FIFO 保留在已檫除状态下） 1：启用 I2C 当 I2C 处于活动状态时，软件可以禁用它。但是，必须注意确保正确禁用 I2C。在“禁用 I2C”描述推荐了一个程序。禁用 I2C 时，会发生以下情况： TXFIFO 和 RXFIFO 刷新 在 I2C 进入空闲（IDLE）状态之前，ISTS 寄存器中的状态位仍然是活动的。 如果模块正在传输，在当前传输完成后，它将停止并删除传输缓冲区的内容。如果模块正在接收，I2C 将在当前字节的末尾停止当前传输，并且不确认传输。
1	ABR	R/W	设置好后，控制器启动传输中止。 0：中止未启动或中止已完成 1：中止操作正在进行 通过设置此位，软件可以在主模式下中止 I2C 传输。软件只能在已设置启用（I2CEN）时设置此位;否则，控制器将忽略中止（ABR）位的任何写入。一旦设定软件无法清除中止位。为了响应中止，在完成当前传输后，控制器发出一个 STOP 并刷新 TXFIFO，然后在中止操作之后设置 TX_ABORT 中断。中止位在中止操作后将被自动清除。
2	TCB	R/W	1：即使 TXFIFO 有数据要传输，也会阻止 I2C 总线上数据的传输。 0：当 TXFIFO 中的第一个数据可用时，数据在 I2C 总线上自动开始传输。 注意:在 TXFIFO 为空（STS1[2]==1）且主机处于空闲状态（STS1[5]==0）时设置 TCB 位，会停止主机执行命令。在 TCB 位未设置之前，不执行任何放在 TXFIFO 中的命令。
31:3	保留		

22.4.29 状态寄存器（STS1）

偏移地址：0x70

复位值：0x06

这是一个只读寄存器，用于指示当前传输状态和 FIFO 状态。状态寄存器可以在任何时候读取。此寄存器中没有任何一个字段请求中断。当通过在 CTRL1 寄存器的位 0 中写入 0 禁用 I2C 时：位 1 和位 2 被设置为 0，位 3 和位 10 被设置为 0。当主状态机或从状态机空闲，I2CEN=0 时：位 5 和位 6 被设置为 0

位/域	名称	R/W	描述
0	ACTF	R	I2C 活动状态 1：I2C 处于活动状态 0：I2C 处于空闲状态
1	TFNFF	R	发送 FIFO 未满。当发送 FIFO 包含一个或多个空位置时设置，FIFO 满时清除。 1：TXFIFO 未满 0：TXFIFO 满
2	TFEF	R	发送 FIFO 完全为空。当发送 FIFO 完全为空时，设置此位。当它包含一个或多个有效条目时，清除此字段。这个位字段不请求中断。 1：TXFIFO 空 0：TXFIFO 不空
3	RFNEF	R	接收 FIFO 不空。当接收 FIFO 包含一个或多个条目时设置此位;当接收 FIFO 为空时，它将被清除。 1：Rx FIFO 不空 0：Rx FIFO 空
4	RFFF	R	接收 FIFO 完全满。当 Receive FIFO 完全满时，设置此字段。当接收 FIFO 包含一个或多个空位置时，清除此字段。 1：RXFIFO 满 0：RXFIFO 未满
5	MAF	R	主机活动状态。当主机不处于空闲状态时，设置此位。 1：主机不空闲 0：主机空闲
6	SAF	R	从机活动状态。当从机不处于空闲（IDLE）状态时，设置此位。 1：从机不空闲 0：从机空闲
31:7	保留		

22.4.30 发送 FIFO 存量寄存器（TFL）

偏移地址：0x74

复位值：0x00

位/域	名称	R/W	描述
3:0	TFL	R	发送 FIFO 内数据量 包含发送 FIFO 中有效数据的数量。
31:4	保留		

22.4.31 接收 FIFO 存量寄存器（RFL）

偏移地址:0x78

复位值：0x00

位/域	名称	R/W	描述
3:0	RFL	R	接收 FIFO 内数据量 包含接收 FIFO 中有效数据的数量。
31:4	保留		

22.4.32 SDA 维持时长寄存器（SDAHOLD）

偏移地址：0x7C

复位值：0x01

位/域	名称	R/W	描述
15:0	TXHOLD	R/W	当 I2C 作为发送器时，以 IC_CLK 周期为单位设置所需的 SDA 保持时间。
23:16	RXHOLD	R/W	当 I2C 作为接收器时，以 IC_CLK 周期为单位设置所需的 SDA 保持时间。
31:24	保留		

22.4.33 发送中止源头寄存器（TAS）

偏移地址：0x80

复位值：0x00

位/域	名称	R/W	描述
0	AD7NA	R	此字段表示 Master 处于 7 位寻址模式，发送的地址没有被任何从机确认。 1：此中止因 7 位地址的 NOACK 生成 0：未生成此中止
1	AD10NA1	R	此字段指示 Master 处于 10 位寻址模式，并且无任何从机确认 10 位地址的首字节字节。 1：无任何从机 ACKed 10 位地址的首字节 0：有从机响应
2	AD10NA2	R	此字段表示 Master 处于 10 位寻址模式，并且该 10 位寻址的第二个地址字节没有被任何从机确认。 1：无任何从机 ACKed 10 位地址的字节 2 0：有响应。
3	TDNA	R	此字段指示仅主模式字段。当主机接收到地址确认，而当它发送地址后面的数据字节时，表示没有接收到来自远程从机的确认。 1：被寻址从机未 ACKed 传输数据 0：被寻址从机 ACKed 传输数据
4	GCNA	R	此字段表示主模式下的 I2C 已发送了一个 General Call，总线上的任何从机都未确认该 General Call。 1：无任何从机告知以 ACKed 响应 0：有从机以 ACKed 响应
5	GCR	R	此字段表示主模式下的 I2C 发送了一个 General Call，但是用户将 General Call 之后的字节编程为总线读取（DATA[9]设置为 1）。 1：GCALL 后接总线读取 0：GCALL 后未接总线读取
6	HSAD	R	此字段表示主机处于高速模式，并且已确认高速主代码（错误行为）。 1：在 HS 模式下 HS 主代码告知收到（ACKed） 0：在 HS 模式下 HS 主代码未告知收到（ACKed）-场景不存在
7	SNR	R	此字段表示主机发送了一个开始（START）字节，且该开始字节已被确认（错误行为）。 1：检测到起始字节的 ACK 0：未检测到开始字节的 ACK

位/域	名称	R/W	描述
8	RNR10B	R	此字段指示禁用重新启动（IC_RESTART_EN 位（CTRL0[5]）=0），并且主机以 10 位寻址模式发送一个读指令。 1：当重新启动禁用时，主机试图在 10 位寻址模式下读取 0：重新启动禁用时，主机不试图在 10 位寻址模式下读取
9	MSTDIS	R	此字段表示用户试图在主模式禁用下启动主机操作。 1：用户在 MASTER 禁用时启动主机操作 0：用户未在 MASTER 禁用时启动主机操作
10	ARBLOST	R	此字段指定主机丢失仲裁，如果也设置了 TAS[14]，则从发送器丢失仲裁。 1：主或从传输器丢失仲裁 0：主或从传输器未丢失仲裁出现异常
11	LFTF	R	这个字段指定从机接收到一个读指令，TXFIFO 中存在一些数据，因此从机发出一个 TAIF 中断来刷新 TXFIFO 中的旧数据。 1：从机获取读指令后刷新 TX-FIFO 中的现存数据 0：从机获取读指令后刷新 TX-FIFO 中的现存数据
12	SAL	R	此字段指定从机丢失仲裁 1：从机丢失仲裁 0：从机未丢失仲裁
13	SRI	R	此字段表示从机读取完成 1：从机读取完成 0：从机未读取完成
14	USRARB	R	此字段表示从机在向远程主机传输数据时丢失了总线。同时设置 TAS[12]。 1：从机丢失总线 0：从机未丢失总线
15	FLUCNT	R	当处理器端响应从模式将数据传输到远程主机的请求，用户在 DATA 寄存器的 CMD（位 8）中写入 1 时。 1：从机试图在读取模式下传输到远程主机 0：从机未试图在读取模式下传输到远程主机
31:16	保留		

22.4.34 生成从机数据 NACK 寄存器（SDNO）

偏移地址：0x84

复位值：0x00

位/域	名称	R/W	描述
0	NACK	R/W	生成 NACK. 这种 NACK 生成只发生在 I2C 是从接收器的情况下。如果此寄存器被设置为 1，它只能在接收到一个数据字节后生成一个 NACK;因此，数据传输被中止，接收到的数据不会被推送到接收缓冲区。当寄存器被设置为 0 时，它会在正常条件下生成 NACK/ACK。 0x1（ENABLED）：从接收器接收数据到时只生成 NACK 0x0（DISABLED）：从接收器收数据后正常响应

位/域	名称	R/W	描述
31:1	保留		

22.4.35 DMA 控制寄存器（DMACTRL）

偏移地址：0x88

复位值：0x00

位/域	名称	R/W	描述
0	RXEN	R/W	接收 DMA 启用：此位启用/禁用接收 FIFO DMA 通道。 1：启用接收 FIFO DMA 通道 0：禁用接收 FIFO DMA 通道 注意：用户配置 DMA 数据宽度只能为 16 位和 32 位；
1	TXEN	R/W	发送 DMA 启用：此位启用/禁用发送 FIFO DMA 通道. 1：启用 FIFO DMA 通道 0：禁用 FIFO DMA 通道. 注意：用户配置 DMA 数据宽度只能为 16 位和 32 位；
31:2	保留		

22.4.36 DMA 发送数据电平寄存器（DTDL）

偏移地址：0x8C

复位值：0x00

位/域	名称	R/W	描述
2:0	DTDL	R/W	发送 FIFO 内的字节数小于等于这个位字段指示的值时发送数据电平，这个位字段控制由传输逻辑发出 DMA 请求的电平。The watermark level=DRDL。
31:3	保留		

22.4.37 DMA 接收数据电平寄存器（DRDL）

偏移地址：0x90

复位值：0x00

位/域	名称	R/W	描述
2:0	DRDL	R/W	接收 FIFO 内的字节数大于这个位字段指示的值时数据电平，这个字段控制接收逻辑发出 DMA 请求的电平，The watermark level=DRDL+1。
31:3	保留		

22.4.38 SDA 设置寄存器（(SDADLY）

偏移地址：0x94

复位值：0x64

位/域	名称	R/W	描述
8:0	SDADLY	R/W	SDA 设置 建议如果所需的延迟为 1000ns，那么对于 10 MHz 的 IC_CLK 频率，SDADLY 应该被编程为值 11。SDADLY 最小可输入值是 2。
31:9	保留		

22.4.39 ACK 通用呼叫寄存器（GCA）

偏移地址：0x98

复位值：0x01

位/域	名称	R/W	描述
0	GCA	R/W	ACK 通用呼叫 当设置为 1 时，I2C 在收到 General Call 时以 ACK 进行响应。否则，I2C 将以 NACK 响应。 1：为 General Call 生成 ACK 0：为 General Call 生成 NACK
31:1	保留		

22.4.40 启用状态寄存器（STS2）

偏移地址：0x9C

复位值：0x00

位/域	名称	R/W	描述
0	I2CEN	R	I2CEN Status。这个位总是反映输出端口 I2CEN 上驱动的值。 1：启用 I2C 0：禁用 I2C
1	SDWB	R	繁忙时禁用从机 此位指示是否由于 CTRL1 寄存器的位 0 从 1 设置为 0，而中止了潜在的或活动的从机操作。当 CPU 向 CTRL1 寄存器写入 0 时设置此位，同时： （a） I2C 从远程主机接收从发送器操作的地址字节。 （b） 来自远程主机的从接收器操作的地址和数据字节。当读取为 1 时，I2C 被视为在 I2C 传输的任一环节中强制执行了一个 NACK，无论 I2C 地址是否与 I2C（SLAADDR 寄存器）中的从地址匹配，或者是否在 CTRL1 设置为 0 之前完成了传输但尚未生效。 注意：如果远程 I2C 主机在 I2C 总线上有机会发出（NACK）之前终止传输，并且 CTRL1[0]设置为 0，那么这个位也设置为 1。当读取为 0，主机活动或 I2C 总线空闲时，I2C 被视为禁用。 注意:当 I2CEN（位 0）被读取为 0 时，CPU 可以安全读取这个位。 1（ACTIVE）：从机在活动时禁用 0（INACTIVE）：从机在空闲时禁用
2	SRDL	R	从机接收数据丢失（Slave Received Data Lost） 该位指示由于 CTRL1 的位 0 从 1 设置为 0，从 I2C 传输接收中止操作。当读取为 1 时，I2C 被视为积极参与了一个中止的 I2C 传输（具有匹配的地址），且 I2C 传输的数据已被输入，即使数据字节已被 NACK 响应。 注意:如果远程 I2C 主机在 I2C 有机会 NACK 传输之前以停止（STOP）条件终止传输，并且 CTRL1[0]设置为 0，那么这个位也设置为 1。当读取为 0 时，I2C 被视为禁用，没有积极参与从接收器传输的数据阶段。 注意:当 I2CEN（位 0）被读取为 0 时，CPU 可以安全读取这个位。 1：从 RX 数据丢失 0：从 RX 数据不会丢失
31:3	保留		

22.4.41 SS、FS 或者 FM+尖峰抑制限制寄存器（LSSSL）

偏移地址：0xA0

复位值：0x05

位/域	名称	R/W	描述
7:0	LSSSL	R/W	在进行任何 I2C 总线事务之前，必须设置此寄存器，以确保稳定运行。此寄存器设置 SCL 或 SDA 行中由尖峰抑制逻辑，过滤掉的最长尖峰的持续时间（以 IC_CLK 周期度量）。该寄存器只能在 I2C 接口被禁用时编写，对应于设置为 0 的 CTRL1[0]寄存器。其它时间写入无效。最小有效值为 1;硬件防止写入的值小于此值，如试图篡改，结果设置为 1。
31:8	保留		

22.4.42 HS 尖峰抑制限制寄存器（HSSSL）

偏移地址：0xA4

复位值：0x01

位/域	名称	R/W	描述
7:0	HSSSL	R/W	此寄存器必须在进行任何 I2C 总线事务时设置，以确保稳定的操作。该寄存器设置在 SCL 或 SDA 线中由尖峰抑制逻辑过滤掉的最长尖峰的持续时间（以 IC_CLK 周期度量）；此寄存器只能在 I2C 接口禁用时编写，对应于设置为 0 的 CTRL1[0]寄存器。其它时间写入无效。最小有效值为 1；硬件防止写入的值小于此值，如试图篡改，结果设置为 1。
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 96 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、同步和串行方式通信，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 主要特征

23.3.1 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 8 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） 主从模式的快速通信，最高可达 18MHz

（8） 时钟的极性和相位可编程

（9） 数据顺序可编程，选择 MSB 或者 LSB 在前

（10） 主模式故障、过载以及 CRC 错误标志可触发中断

（11） 具有 DMA 传输和接收缓冲器

（12） 通过硬件 CRC 进行计算、发送和校验

23.3.2 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

23.4 SPI 功能描述

23.4.1 SPI 信号线描述

表格 97 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

引脚名称	描述
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。

23.4.2 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 98 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
2	1	0	偶数边沿	低电平
3	1	1	偶数边沿	高电平

23.4.3 数据帧格式

通过配置 SPI_CTRL1 寄存器中的 LSBSEL 位，设定是 MSB 在先还是 LSB 在先。
通过配置 SPI_CTRL1 寄存器的 DFLSEL 位，选择是 8/16 位数据帧格式进行发送/接收。

23.4.4 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。

23.4.5 SPI 模式

23.4.5.1 SPI 主模式

在主模式中，在 SCK 引脚产生串行时钟

主模式配置

- 配置 SPI_CTRL1 寄存器中的 MSMSEL=1
- 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 DFLSEL 位选择 8/16 位数据帧格式
- 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 SPIEN 位，使能 SPI

在主模式中：MOSI 引脚是数据输出，MISO 是数据输入

23.4.5.2 SPI 从模式

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 MSMSEL=0
- 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 DFLSEL 位选择 8/16 位数据帧格式
- 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 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

23.4.5.3 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 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

23.4.5.4 SPI 的单工通信

一条时钟线和一条单向数据线

此模式中，SPI 模块只接收或者只发送

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：

- 主模式下，使能 SPI 启动通信，清除 SPI_CTRL1 寄存器的 SPEN 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。

只接收模式中可以通过设置 SPI_CTRL1 寄存器中 RXOMEN 位关闭 SPI 输出功能，此时，释放发送引脚（主模式下是 MOSI，从模式下是 MISO），可以作为其它功能使用。

23.4.6 SPI 不同模式下数据发送和接收过程

表格 99 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 95 全双工模式连接

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）

图 98 双向线连接

23.4.6.1 处理数据的发送与接收

数据发送

完成模式配置之后，SPI 模块使能保持在空闲状态。

主模式下：软件写入一个数据帧到发送缓冲器中，发送过程开始

从模式下：SCK 引脚上的 SCK 信号开始跳变，与此同时 NSS 引脚电平位低，发送过程开始（数据开始发送前，确保数据已提前写入发送缓冲区）。

SPI 发送一个数据帧时，会将数据帧从数据缓冲区加载到移位寄存器中，之后开始发送数据。数据帧发送一位后，TXBEFLG 置 1，如果需要连续发送数据，软件需要等到 TXBEFLG=1 向 SPI_DATA 寄存器中写入数据。（TXBEFLG 标志是由硬件置 1，软件清零）。

数据接收

在数据接收过程中 BSYFLG 标志会一直置 1。

在采样时钟的最后一个边沿，接收到的数据从移位寄存器传送到接收缓冲区，设置 RXBNEFLG 标志，软件通过读取数据寄存器（SPI_DATA）中的数据获取接收缓冲区中的内容，如果设置了 SPI_CTRL2 寄存器的 RXBNEIEN 位，此时会产生中断，在读取数据之后会自动清除 BSYFLG 标志。

23.4.6.2 主/从设备下的全双工发送和接收模式

主设备下的全双工模式

- 写数据到 SPI_DATA 寄存器（发送缓冲器）后，开始数据传输。
- SPI 发送第一位数据时，数据从发送缓冲器传送到移位寄存器中，按照顺序被串行的传送到 MOSI 引脚中。
- MISO 引脚上接收的数据，按照顺序被串行的传送到 SPI_DATA 寄存器（接收缓冲器）中。

发送和接收是同时进行的。

从设备下的全双工模式

- 从设备接收到时钟信号，MOSI 引脚出现第一个数据位时，开始数据传输，之后的数据位会被依次传送到移位寄存器中。
- SPI 发送第一位数据时，数据从发送缓冲器传送到移位寄存器中，按照顺序被串行的传送到 MISO 引脚中。
- 软件必须保证在 SPI 主设备开始传输数据之前确保已写入要发送的数据。

发送和接收是同步进行的

主/从设备下的全双工发送和接收过程

（1） 使能 SPI 模块：配置 SPI_CTRL1 寄存器的 SPIEN=1。

（2） 将第一个要发送的数据写入 SPI_DATA 寄存器中，会清除 TXBEFLG 标志。

（3） 等待 TXBEFLG 标志位置 1（硬件控制），写入需要发送的第二个数据位。

（4） 等待 RXBNEFLG 标志位值置 1（硬件控制）读出 SPI_DATA 寄存器中的第一个接收的数据，与此同时清除 RXBNEFLG 标志（软件清零）。重复操作，同时发送和接收数据。

（5） 等待 RXBNEFLG=1，接收最后一个数据。

（6） 等待 TXBEFLG=1，在 BSYFLG=0 之后关闭 SPI 模块。

23.4.6.3 主/从设备下的双向发送模式

主设备下的双向发送

- 数据写入到 SPI_DATA 寄存器后，开始传输
- 发送缓冲器中的数据并行的传送到移位寄存器中，之后按顺序串行传送到 MOSI 引脚上。

从设备下的双向发送

- 从设备接收到时钟信号，MISO 引脚出现第一个数据位时，开始数据传输。
- 与此同时，发送缓冲器需要发送的数据并行的传送到移位寄存器中，随后被串行发送到 MISO 引脚上，（数据开始发送前，确保数据已提前写入发送缓冲区）。

主/从设备下的双向发送过程

（1） 使能 SPI 模块：配置 SPI_CTRL1 寄存器的 SPIEN=1。

（2） 将第一个要发送的数据写入 SPI_DATA 寄存器中，会清除 TXBEFLG 标志。

（3） 等待 TXBEFLG=1，写入第二个数据，重复操作，发送后面的数据

（4） 写入最后一个数据后，等待 TXBEFLG=1、BSYFLG=0，发送完成

23.4.6.4 主/从设备下的单向/双向接收模式

（1） 使能 SPI 模块：配置 SPI_CTRL1 寄存器的 SPIEN=1。

（2） 主设备中：立即产生 SCK 时钟，SPI 关闭之前，不断接收数据。

（3） 从设备中：SPI 主设备拉低 NSS 且产生时钟时，接收数据。

（4） 等到 RXBNEFLG 标志置 1，通过 SPI_DATA 读取数据，重复操作接收数据。

23.4.7 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。

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）。

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；与此同时比较 CRC 和 SPI_RXCRC 的值，如果出现了不匹配的情况，需要设置 SPI_STS 寄存器的 CRCEFLG 位，在设置了 SPI_CTRL2 寄存器的 ERRIEN 位时，会发生中断。

注意：

（1）当 SPI 处于从设备下并且使用了 CRC 功能，在 NSS 引脚时高电平时 CRC 计算仍会继续。例如：当主设备与多个从设备交替通信时，就会出现上述情况，则需要避免 CRC 的错误操作。

（2）一个从设备从未选中（NSS 为高电平）到选中（NSS 为低电平 0）过程中，需要清除主从两端的 CRC 数值，来保持主从设备端下次 CRC 计算结果的同步。

（3）当 SPI 处于从设备模式时，要在时钟稳定之后再使能 CRC 计算。

（4）当 SPI 时钟频率过高时，CPU 的操作会影响 SPI 的带宽，建议采用 DMA 模式避免 SPI 速度的降低。

（5）当 SPI 时钟频率过高时，在 CRC 传输期间内，减少 CPU 的使用频率，在发送 CRC 过程中禁止函数调用来避免接收最后的数据和 CRC 时出错。

（6）在从模式下使用了 NSS 硬件模式，NSS 引脚应该在数据传输和 CRC 传输期间保持为低。

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）

23.4.8 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 寄存器中，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。

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。

23.4.9 关闭 SPI

数据传输结束后，通过关闭 SPI 模块结束通讯。在一些配置中，如果在数据还没有传输完成下关闭了 SPI，可能会造成数据传输错误。不同的运行模式需要不同的方法关闭 SPI

主/从设备下的全双工模式

（1） 等待 RXBNEFLG 标志位置 1，接收最后一个数据

（2） 等待 TXBEFLG 标志位置 1

（3） 等待 BSYFLG 标志位清零

（4） 关闭 SPI（设置 SPI_CTRL1 寄存器的 SPIEN=0）

主/从设备下的单向/双向只接收模式

（1） 等待第 n-1 个 RXBNEFLG 标志位置 1

（2） 在关闭 SPI（设置 SPI_CTRL1 寄存器的 SPIEN=0）之前等待一个 SPI 时钟周期

（3） 在进入停机模式之前等待最后一个 RXBNEFLG 标志位置 1

从模式下的只接收/双向的接收模式

在任意时刻都可以关闭 SPI（设置 SPI_CTRL1 寄存器的 SPIEN=0），会在传输结束后关闭。如果要进入停机模式，需要等到 BSYFLG 标志位清零。

23.4.10 SPI 中断

23.4.10.1 状态标志位

完全监控 SPI 总线的状态有三个标志位

发送缓冲器空标志 TXBEFLG

TXBEFLG=1 说明发送缓冲器位空，可以写入下一个待发送的数据，当数据写入 SPI_DATA 寄存器时，清除 TXBEFLG 标志位。

接收缓冲器非空标志 RXBNEFLG

RXBNEFLG=1 说明接收缓冲器中含有有效数据，可以通过 SPI_DATA 寄存器来读取数据，清除 RXBNEFLG 标志

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，但是在主设备下的双线接收模式，在接收数据期间 BSYFLG=0。

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。

BSYFLG 标志位可以用于在多主模式下避免写入数据的冲突。

当出现传输结束（主模式的连续通信除外）、SPI 关闭以及主模式失效的情况，BSYFLG 标志将会清零。

通讯不连续时，在数据项和数据项之间，BSYFLG=0。

通讯连续时：

- 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

注意：最好使用 TXBEFLG 和 RXBNEFLG 标志来处理每一个数据项的发送和接收。

23.4.10.2 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）；MSMSEL 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 MSMSEL 位

溢出错误 OVRFLG

溢出错误：当主设备发送了数据之后，RXBNEFLG 标志位还为 1，说明此时发生为溢出错误。此时 OVRFLG 位置 1，如果还设置了 ERRIEN 位，将会产生中断。

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，此时读取 SPI_DATA 寄存器的值是之前没有读取的数据，而在之后的发送的数据不会被读取。

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 CRCEFLG

通过设置 SPI_CTRL1 寄存器的 CRCEN 位，启动 CRC 计算，CRC 错误标志，可以核对接收的数据是否有效。

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。

表格 100 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
MEFLG	主模式错误标志	ERRIEN	读/写 SPI_STS 寄存器，然后写 SPI_CTRL1 寄存器
OVRFLG	溢出错误标志		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

23.5 I2S 功能描述

通过设置 SPI_I2SCFG 的 I2SMOD 位，使能 I2S 功能。

I2S 与 SPI 共用三个引脚：

- 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。

23.5.1 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。

23.5.1.1 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 99 I2S 飞利浦协议波形（16/32 位）

图 100 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 101 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。

23.5.1.2 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 102 MSB 对齐标准波形（16/32 位数据）

图 103 MSB 对齐标准波形（24 位数据）

图 104 MSB 对齐标准波形（16 位扩展至 32 位）

23.5.1.3 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 105 LSB 对齐标准波形（16/32 位数据）

图 106 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 107 LSB 对齐标准下（16 位扩展至 32 位）

23.5.1.4 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 108 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 109 PCM 标准波形

23.5.2 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 101 音频采样频率（Fs）公式

MCOEN	CHLEN	音频采样频率（Fs）
1	0	I2SxCLK/[（16*2）*（(2*I2SPSC）+ODDPSC）*8]
1	1	I2SxCLK/[（32*2）*（(2*I2SPSC）+ODDPSC）*4]
0	0	I2SxCLK/[（16*2）*（(2*I2SPSC）+ODDPSC）]
0	1	I2SxCLK/[（32*2）*（(2*I2SPSC）+ODDPSC）]

23.5.3 I2S 模式

表格 102 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

23.5.3.1 I2S 主模式配置流程

（1） 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。

（2） 配置 SPI_I2SCFG 寄存器的 CPOL 位，定义 SPI 在空闲状态的时钟极性。

（3） 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。

（4） 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）。

（5） 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。

（6） 通过配置 SPI_I2SCFG 的 I2SMOD 的位，设置 I2S 的运行模式。

（7） 将 SPI_I2SCFG 寄存器的 I2SEN 位置 1。

23.5.3.2 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。

23.5.3.3 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入下：

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

23.5.3.4 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。

（8） 配置 SPI_I2SCFG 寄存器的 I2SMOD 位激活 I2S 功能。

（9） 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。

（10） 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）。

（11） 将 SPI_I2SCFG 寄存器的 I2SEN 位置 1。

23.5.3.5 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。

23.5.3.6 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。

23.5.4 I2S 中断

23.5.4.1 状态标志位

在 I2S 中有三个状态标志位监控 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 标志位表明了此时 SD 引脚上发送的数据处在左声道还是右声道，此标志位在 TXBEFLG=1 时刷新。

在从模式的发送流程中如果发生了下溢错误，SCHDIR 标志位的值失效，如果需要重新开始通讯，需要把 I2S 功能关闭再打开。

在接收模式中，SCHDIR 标志位表明了此时接收的数据是来自左声道还是右声道。此标志位在 SPI_DATA 寄存器接收到数据时刷新。，

在接收模式中如果发生上溢错误，SCHDIR 标志位失效，如果需要重新开始通讯，需要把 I2S 功能关闭在打开。

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=1 时，会产生中断。通过读取 SPI_STS 寄存器的值可以清除中断标志。

23.5.4.2 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。

表格 103 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

中断标志	中断事件	使能控制位	清除方法
UDRFLG	上溢标志位		读 SPI_STS 寄存器 再读 SPI_DATA 寄存器

23.5.4.3 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。

23.6 寄存器地址映射

表格 104 SPI 和 I2S 寄存器映射

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

23.7 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。

23.7.1 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。
1	CPOL	R/W	配置时钟极性（Clock Polarity Configure） 当 SPI 处于空闲状态时，SCK 保持的电平状态。 0：低电平 1：高电平 注：通信进行时，不能修改该位

位/域	名称	R/W	描述
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
5:3	BRSEL	R/W	选择波特率分频系数（Baud Rate Divider Factor Selectl） 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=F_{PCLK}/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） 当 CTRL1_SSEN=1 时（软件 NSS 模式），可配置该位选择内部 NSS 电平 0：内部 NSS 为低电平 1：内部 NSS 为高电平
9	SSEN	R/W	使能软件从设备（Software Slave Device Enable） 0：禁止软件 NSS 模式，内部 NSS 电平由外部 NSS 引脚决定 1：启用软件 NSS 模式，内部 NSS 电平由外部 ISSEL 引脚决定
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	DFLSEL	R/W	选择数据帧长度（Data Frame Length Format Select） 0：8 位数据帧格式 1：16 位数据帧格式 只有在 SPIEN=0 时，才能写入该位，改变数据帧长度。
12	CRCNXT	R/W	使能下一个传输数据是 CRC（CRC Transfer Next Enable） 0：下一个传输数据来自发送缓冲区 1：下一个传输数据来自 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。

位/域	名称	R/W	描述
13	CRCEN	R/W	使能 CRC 校验（CRC Calculate Enable） 0：禁止 1：使能 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。
14	BMOEN	R/W	使能双向模式的输出（Bidirectional Mode Output Enable） 0：禁止，即仅接收模式 1：使能，即仅发送模式 在 BMEN=1 即单线双线模式下，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：双线单向模式 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

23.7.2 SPI 控制寄存器 2（SPI_CTRL2）

偏移地址：0x04

复位值：0x0000

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 模式下不使用。
4:3	保留		
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。
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

位/域	名称	R/W	描述
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
15:8	保留		

23.7.3 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	接收缓冲非空标志（Receive Buffer Not Empty Flag） 0：空 1：非空
1	TXBEFLG	R	发送缓冲器为空标志（Transmit Buffer Empty Flag） 0：非空 1：空
2	SCHDIR	R	声道方向标志（Sound Channel Direction Flag） 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 注：在 SPI 模式下不使用，在 PCM 模式下无左右声道。
3	UDRFLG	R	发生欠载标志（Underrun Occur Flag） 0：未发生 1：发生 该标志位由硬件置位，软件对该位写 0 清除。 在 SPI 模式下不使用
4	CRCEFLG	RC_W0	发生 CRC 错误标志（CRC Error Occur Flag） 该位表示接收的 CRC 值和 RXCRC 寄存器的值是否匹配 0：匹配 1：不匹配 该位由硬件置位，软件对该位写 0 清除；在 I2S 模式下不使用。
5	MEFLG	R	发生模式错误标志（Mode Error Occur Flag） 0：未发生 1：发生 由硬件置位，软件对该位写 0 清除；在 I2S 模式下不使用。
6	OVRFLG	R	发生过载标志（Overrun Occur Flag） 0：未发生 1：发生 由硬件置位，软件对该位写 0 清除。
7	BSYFLG	R	SPI 忙标志（SPI Busy Flag） 0：SPI 空闲 1：SPI 正在通信 由硬件置位或者清除。
15:8	保留		

23.7.4 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） 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、接收数据时只会用到 DATA[7:0]，DATA[15:8]是无效；对于 16 位数据，发送、接收时会用到 DATA[15:0]。

23.7.5 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） 该寄存器是包含了于 CRC 计算的 CRC 多项式，可修改，复位值是 0x0007。

23.7.6 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） 硬件计算接收数据的 CRC 数据存储在改寄存器；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 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。

23.7.7 SPI 发送 CRC 寄存器（SPI_TXCRC）

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） 硬件计算发送数据的 CRC 数据存储在改寄存器；位数和数据帧的长度一致，即如果发送数据是 8 位的，CRC 计算按 CRC8 方式；如果发送数据是 16 位的，CRC 计算按 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。

23.7.8 SPI_I2S 配置寄存器（SPI_I2SCFG）

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 声道长度大于数据长度时，才能成功配置声道长度，否则硬件会自动调整声道长度；当 I2SEN=0 时才能配置该位，在 SPI 模式下不使用。
2:1	DATALEN	R/W	配置待传输数据长度（Configure the Length of the sData to Be Transferred） 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 当 I2SEN=0 时才能配置该位，在 SPI 模式下不使用。
3	CPOL	R/W	配置空闲状态时钟极性（Idle State Clock Polarity Configure） 0：低电平 1：高电平 当 I2SEN=0 时才能配置该位，在 SPI 模式下不使用。
5:4	I2SSSEL	R/W	选择 I2S 标准（I2S Standard Selection） 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 当 I2SEN=0 时才能配置该位，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 只适用 PCM 标准（I2SSSEL=11）；当 I2SEN=0 时才能配置该位，在 SPI 模式下不使用。
9:8	I2SMOD	R/W	配置 I2 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：主设备接受 当 I2SEN=0 时才能配置该位，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		

23.7.9 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） I2SPSC 不能设置为 0 和 1；当 I2SEN=0 时才能配置该位，在 SPI 模式下不使用。
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 当 I2SEN=0 时才能配置该位，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 当 I2SEN=0 时才能配置该位，在 SPI 模式下不使用。
15:10	保留		

24 控制器局域网（CAN）

24.1 术语全称、缩写描述

表格 105 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

24.1.1 数据接收

24.1.1.1 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。

24.1.1.2 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

24.2 简介

CAN 是控制器局域网络 Controller Area Network 的缩写，是 ISO 国际标准化的串行通信协议，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。

24.3 主要特性

（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 28 个过滤器组

- 可记录接收时间

（5） 存储器

- CAN1 与 USBD1 共用 512-Byte SRAM
- CAN2 与 USBD2 共用 512-Byte SRAM
- CAN1 和 USBD2 可同时使用
- CAN2 和 USBD1 可同时使用
- CAN1 和 CAN2 可同时使用

24.4 功能描述

24.4.1 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。

24.4.2 报文结构

图 110 标准数据帧

图 111 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。

24.4.3 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。

24.4.3.1 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。

24.4.3.2 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。

24.4.3.3 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。

24.4.4 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。

24.4.4.1 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 112 CAN 工作在静默模式

24.4.4.2 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 113 CAN 工作在环回模式

24.4.4.3 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 114 CAN 工作在静默环回模式

24.4.4.4 正常模式

该模式下，可以向总线发送数据和接收数据。

图 115 CAN 工作在正常模式

24.4.5 数据发送

24.4.5.1 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。

24.4.5.2 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

24.4.5.3 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。

24.4.5.4 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 REQCFLG 位置 1，发送的结果会在 TXSUSFLG、ARBLSTFLG 和 TXERRFLG 位上

24.4.6 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 28 个过滤器组。

24.4.6.1 位宽

每个过滤器组都可以配置两种位宽。

图 116 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]	IDTYPESEL	TXRFREQ	0

图 117 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]

24.4.6.2 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 106 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 107 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

24.4.6.3 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

24.4.7 位时序及波特率

24.4.7.1 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

24.4.7.2 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：$T1bit=1T_q+T_{s1}+T_{s2}$

波特率=1/ T1bit

T_q =（BRPSC+1）* T_{PCLK}

24.4.8 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。

24.4.8.1 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。

24.4.9 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：

- 硬件将配置寄存器 CAN_RXF0 的 FMNUM0[1:0]位为非 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：

- 硬件将配置寄存器 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 118 事件标志和中断产生

24.5 寄存器地址映射

CAN1 基地址：0x4000_6400

CAN2 基地址：0x4000_6800

注：CAN1 和 CAN2 除基地址外寄存器及偏移地址完全相同。

表格 108 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	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

24.6 寄存器功能描述

24.6.1 CAN 控制和状态寄存器

24.6.1.1 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		

24.6.1.2 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 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。

位/域	名称	R/W	描述
1	SLEEPFLG	R	正处于睡眠模式标志（Being Sleep Mode Fiag） 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	SLEEPIFLG	RC_W1	已进入睡眠模式中断标志（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		

24.6.1.3 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

24.6.1.4 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 FIFO 0 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 Massage） 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		

24.6.1.5 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
5	RFOM1	R/S	释放接收 FIFO1 输出邮箱以接收报文（Release Receive FIFO1 Output Mailbox to Receive Massage）

位/域	名称	R/W	描述
			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 FIFO1 的输出邮箱
31:6	保留		

24.6.1.6 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	使能发送邮箱为空中断（Transmit Mailbox Empty Interrupt Enable） 当 REQCFLGx 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。 0：不产生中断 1：产生中断
1	FMIEN0	R/W	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） 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。 0：不产生中断 1：产生中断
2	FFULLIEN0	R/W	使能 FIFO0 满中断（FIFO0 Full Interrupt Enable） 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。 0：不产生中断 1：产生中断
3	FOVRIEN0	R/W	使能 FIFO0 过载中断（FIFO0 Overrun Interrupt Enable） 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。 0：不产生中断 1：产生中断
4	FMPIEN1	R/W	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） 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。 0：不产生中断 1：产生中断
5	FFULLIEN1	R/W	使能 FIFO1 满中断（FIFO1 Full Interrupt Enable） 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。 0：不产生中断 1：产生中断

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置 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 SLEEPIFLG 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		

24.6.1.7 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。

24.6.1.8 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） × 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} x（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 $t_{BS2} = t_{CAN}$ x（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 $t_{RJW}=t_{CAN}$ x（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

24.6.2 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_RXDLENx 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。

24.6.2.1 发送邮箱标识符寄存器（CAN_TXMIDx）（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）一复位值为 0

24.6.2.2 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		

24.6.2.3 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 2of the Message）
31:24	DATABYTE3	R/W	报文数据字节 3（Data Byte 3 of the Message）

24.6.2.4 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）

24.6.2.5 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
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

位/域	名称	R/W	描述
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。

24.6.2.6 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。

24.6.2.7 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 1 of the Message）
23:16	DATABYTE2	R	报文数据字节 2（Data Byte 2 of the Message）
31:24	DATABYTE3	R	报文数据字节 3（Data Byte 3 of the Message）

注： 所有接收邮箱寄存器都是只读的。

24.6.2.8 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 4 of the Message）
15:8	DATABYTE5	R	报文数据字节 5（Data Byte 5 of the Message）
23:16	DATABYTE6	R	报文数据字节 6（Data Byte 6 of the Message）
31:24	DATABYTE7	R	报文数据字节 7（Data Byte 7 of the Message）

注： 所有接收邮箱寄存器都是只读的。

24.6.3 CAN 过滤器寄存器

24.6.3.1 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
7:1	保留		
13:8	CAN2SB	R/W	CAN2 起始存储区（CAN2 Start Bank） 该位用来定义 CAN2 的起始存储区，范围：0-27 注：当 CAN2SB=28d 时，CAN1 的所有过滤器都可以使用；当 CAN2SB=0 时，CAN2 的所有过滤器都可以使用。
31:14	保留		

注： 该寄存器的非保留位完全由软件控制。

24.6.3.2 CAN 过滤器模式配置寄存器（CAN_FMCFG）

偏移地址：0x204

复位值：0x0000 0000

位/域	名称	R/W	描述
27:0	FMCFGx	R/W	配置过滤器 x 模式（Filter Mode Configure） x 取值为 0-27。 0：标识符屏蔽位模式 1：标识符列表模式
31:28	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。

24.6.3.3 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
27:0	FSCFGx	R/W	配置过滤器 x 位宽（Filterx Scale Configure） x 取值为 0-27。 0：2 个 16 位 1：单个 32 位
31:28	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。

24.6.3.4 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
27:0	FFASSx	R/W	配置过滤器 x 与 FIFO 关联（Configure Filterx Associated with FIFO） x 取值为 0-27。 0：过滤器与 FIFO0 关联 1：过滤器与 FIFO1 关联
31:28	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。

24.6.3.5 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
27:0	FACTx	R/W	激活过滤器 x（Filterx Active） x 取值为 0-27。 0：禁用 1：激活
31:28	保留		

24.6.3.6 CAN 过滤器组 i 的寄存器 x（CAN_FiBANKx）（i＝0..27；x=1..2）

偏移地址：0x240..0x31C

复位值：0xXXXX 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 28 组过滤器：i=0..27。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。

25 安全数字输入输出接口（SDIO）

25.1 术语全称、缩写描述

表格 109 术语全称、缩写描述

中文全称	英文全称	英文缩写
先入先出队列	First Input First Output	FIFO
命令状态机	Command Path State Machine	CPSM
数据状态机	Data Path State Machine	DPSM

25.2 简介

安全数字输入/输出接口能够连接 SD 卡，SD I/O 卡，多媒体卡（MMC）和 CE-ATA 卡主机接口,提供 AHB 系统总线与 SD 存储卡、SD I/O 卡、MMC 和 CE-ATA 设备之间的数据传输。

25.3 主要特征

（1） SD 卡：与 SD 存储卡规格版本 2.0 全兼容

（2） SD I/O 卡：与 SD I/O 卡规格版本 2.0 全兼容:支持两种不同的总线模式:1 位（默认）和 4 位。

（3） MMC：与多媒体卡系统规格书 4.2 及之前版本全兼容。有三种不同的数据总线模式：1 位（默认）、4 位和 8 位。

（4） CE-ATA：与 CE-ATA 数字协议版本 1.1 全兼容。

（5） 8 位总线模式下数据传输速率达 48MHz

（6） 中断和 DMA 请求

（7） 数据和命令输出使能信号，用于控制外部双向驱动器。

注：当前版本的 SDIO 在同一时间里只支持一个 SD/SD IO/MMC 4.2 卡，但可支持多个 MMC4.1 或以前版本的卡。

25.4 功能描述

SDIO 结构主要有两大部分：

SDIO 适配器：实现 MMC/SD/SD I/O 卡的相关功能，由控制单元、数据单元、命令单元组成。控制单元管理时钟信号，数据单元管理数据传输，命令单元管理命令传输。

AHB 总线接口：操作 SDIO 适配器中的寄存器，用于数据传输的 FIFO 单元，产生中断和 DMA 请求信号。

图 119 SDIO 结构框图如图

表格 110 SDIO 引脚定义

引脚	方向	说明
SDIO_CK	输出	MMC/SD/SD I/O 卡时钟，从主机到卡的时钟线
SDIO_CMD	双向	MMC/SD/SD I/O 卡命令，双向命令信号
SDIO_D[7:0]	双向	MMC/SD/SD I/O 卡数据，双向数据总线

25.4.1 SDIO 总线拓扑

上电复位后，主机必须通过特殊的基于消息的总线协议初始化设备。

每个消息有以下部分中的一个来表示：

- 命令：命令是启动一个操作的令牌，主机到卡，命令串行传输到 CMD 线上。
- 响应：从卡到主机，作为先前收到命令的应答，响应串行传输在 CMD 上。
- 数据：可以从主机到卡或者从卡到主机。通过数据线传送。用于数据传输的数据线数目可以是 1（D0）、4（D0-D3）、8（D0-D7）。

在多媒体卡/SD/SD I/O 总线上的基本操作是命令/响应结构。

在 SD/SD I/O 存储器卡上传送的数据时以数据块的形式传输；在 MMC 上传送的数据时以数据块或数据流的形式传输；在 CE-ATA 设备上传送的数据也是以数据块的形式传输。

图 120 SDIO “无响应”和“无数据”操作

图 121 SDIO（多）数据块读操作

图 122 SDIO（多）数据块写操作

:繁忙

图 123 SDIO 数据流读操作

图 124 SDIO 数据流写操作

25.4.2 SDIO 适配器

SDIO 产生的信号有以下几种：

- SDIO_CLK：SDIO 控制器提供给卡的时钟。每个时钟周期在命令线（SDIO_CMD）和所有的数据线（SDIO_D）上直接发送一位命令或数据。对于 MMC 卡 V3.31 版本，SDIO_CLK 频率在 0-20MHz，对于 MMC 卡 V4.2 版本在 0-48MHz 之间，对于 SD 或 SD I/O 卡在 0-25MHz 之间。
- SDIO_CMD：该信号是双向命令通道，用于卡的初始化和命令传输。命令从 SDIO 控制器到卡，响应从卡发送到主机。CMD 信号有两种操作模式：初始化的开漏模式（用于 MMC 卡 V3.31 及以前版本）和命令发送的推挽模式（SD、SD I/O、MMC 卡 4.2 版本初始化也是推挽模式）。
- SDIO_D[7:0]：信号线均为双向数据通道，处于推挽模式。默认情况下，上电或复位后，只有 D0 用于数据传输，SDIO 适配器可配置更宽的数据总线用于数据传输，使用 D0-D3、D0-D7（仅使用于 MMC V4.2）。

控制单元

控制单元包括电源管理和时钟管理功能。

命令单元

命令单元实现向卡发送命令与接收卡发送的命令，数据传输由命令状态机（CPSM）控制。

命令状态机

CS_Idle　复位后准备发送命令		
1.CPSM被使能并且WAITDEND使能	→	CS_Pend
2.CPSM被使能并且WAITDEND失能	→	CS_Send
3.CPSM被关闭	→	CS_Idle
注：命令状态机空闲状态至少保持8个SDIO_CLK周期，以满足Ncc和Nrc时序限制。 Ncc：两个主机命令之间最小时间间隔 Nrc：主机命令与卡响应的最小时间间隔		

CS_Pend　等待数据传输结束		
1.数据传送完成	→	CS_Send
2.CPSM被关闭	→	CS_Idle

CS_Send　发送命令		
1.发送命令后有响应	→	CS_Wait
2.发送命令后无响应	→	CS_Idle
3.CPSM被关闭	→	CS_Idle

CS_Wait 等待响应起始位		
1.接收到响应(检测到起始位)	→	CS_Wait
2.接收响应超时	→	CS_Idle
3.CPSM被关闭	→	CS_Idle
注意:命令超时时间固定为64个SDIO_CLK时钟周期		

CS_Receive 接收响应并检测CRC		
1.在CE-ATA模式下收到响应，失能CE-ATA中断并且等待CE-ATA设备命令完成信号使能	→	CS_Waitcompl
2.在CE-ATA模式下收到响应，失能CE-ATA中断并且等待CE-ATA设备命令完成信号失能	→	CS_Pend
3.CPSM被关闭	→	CS_Idle
4.收到响应	→	CS_Idle
5.命令CRC检测失败	→	CS_Idle

CS_Waitcompl 等待CE-ATA设备命令完成信号		
1.收到CE-ATA命令完成信号	→	CS_Idle
2.CPSM被关闭	→	CS_Idle
3.命令CRC检测失败	→	CS_Idle

数据单元

数据单元实现主机与卡之间的数据传输。当数据宽度为 8 位,数据传输使用 SDIO_D[7:0]信号线。当数据宽度为 4 位，数据传输使用 SDIO_D[3:0]信号线。当数据宽度为 1 位，数据传输使用 SDIO_D[0]信号线。

数据传输流由数据状态机（DPSM）控制。

数据状态机

DS_Idle 数据单元空闲，等待发送和接收数据		
1.DPSM使能并且数据传输从主机到卡	→	DS_WaitS
2.DPSM使能并且数据传输从卡到主机	→	DS_WaitR
3.DPSM使能并且读等待已经开始并且使能SDIO模式	→	DS_Readwait

DS_WaitS 等待数据FIFO为空标志无效或者数据传输结束		
1.数据传输结束	→	DS_Idle
2.DPSM被关闭	→	DS_Idle
3.数据FIFO为空标志无效	→	DS_Send

DS_Send 发送数据到卡		
1.数据已发送	→	DS_Busy
2.DPSM被关闭	→	DS_Idle
3.数据FIFO下溢错误发送	→	DS_Idle
4.内部CRC错误	→	DS_Idle

DS_Busy 等待CRC状态标志		
1.数据已发送	→	DS_Busy
2.DPSM被关闭	→	DS_Idle
3.数据FIFO下溢错误发送	→	DS_Idle
4.内部CRC错误	→	DS_Idle

DS_WaitR 等待接收数据的起始位		
1.数据接收结束	→	DS_Idle
2.DPSM被关闭	→	DS_Idle
3.数据超时	→	DS_Idle
4.在超时前收到起始位	→	DS_Receive

DS_Receive 接收卡的数据并将其写入数据FIFO		
1.数据块已接收	→	DS_WaitR
2.数据传输结束	→	DS_WaitR
3.数据FIFO下溢发送	→	DS_Idle
4.数据已经接收并且读等待开始并使能SDI/O模式	→	DS_Readwait
5.DPSM被关闭或CRC错误	→	DS_Idle

DS_Readwait 等待“读等待停止”指令		
1.“读等待停止”使能	→	DS_WaitR
2.DPSM被关闭	→	DS_Idle

25.4.3 AHB 接口

AHB 接口能够访问 SDIO 寄存器和数据 FIFO，也能产生中断和 DMA 请求。它包括数据 FIFO 单元、寄存器单元和中断/DMA 请求控制逻辑。

SDIO 中断

当至少一个已经被选中的状态标志为高时，中断逻辑产生中断请求。中断使能允许寄存器允许中断逻辑产生相应中断。

数据 FIFO

数据 FIFO 单元有一个数据缓冲区，用于接收与发送 FIFO。FIFO 包括一个每个字的宽度为 32 位，深度为 32 字的数据缓冲区。发送 FIFO 被用在当需要写数据到卡上，待传输的数据通过 AHB 总线写入发送 FIFO，SDIO 适配器中的数据单元从发送 FIFO 读取数据，再发送到卡。接收 FIFO 用于从卡读取数据时，从卡读取数据，然后将待传送数据写入接收 FIFO。

寄存器单元

寄存器单元包括所有的系统寄存器，生成信号用于控制卡与控制器之间的通信。

25.4.4 卡功能说明

25.4.4.1 卡寄存器

卡内部定义了接口寄存器：OCR、CID、CSD、EXT_CSD、RCA、DSR、SCR。这些寄存器只能通过相应命令访问。OCR、CID、CSD 和 SCR 寄存器包括卡的特定信息，RCA 和 DSR 寄存器时配置寄存器，存储实际的配置参数。EXT_CSD 寄存器同时包括卡的特定信息和实际的结构参数。

OCR 寄存器：32 位操作条件寄存器存储卡的 V_{DD} 电压描述和存储模式指示（MMC）。另外，该寄存器包括一个状态信息位。如果卡上电过程已经完成该状态位置位。该寄存器在 MMC 和 SD 卡之间有点不同。主机可以使用 CMD1（MMC）,ACMD41（SD 存储卡），CMD5（SD I/O）来获取该寄存器的内容。

CID 寄存器：卡识别寄存器（CID）是 128 位宽。它包含在卡识别阶段使用的卡识别信息。每个读/写（RW）卡应具有唯一的标识号。

CSD 寄存器：卡特定数据寄存器提供访问卡中的内容信息。CSD 定义数据格式、错误校正类型、最大数据访问时间、数据传输速度等。

扩展 CSD 寄存器：只有 MMC4.2 有该寄存器。扩展 CSD 寄存器定义卡属性和选择模式。它的长度为 512 字节。最高 320 字节为属性段，定义了卡的功能，并且不能由主机修改。最低 192 字节是模式段，定义了卡工作在哪种配置下。

RCA 寄存器：可写的 16 位相对卡地址寄存器存放卡地址，该地址在卡的初始化期间由卡向外发布。这个地址用于卡识别过程之后，所寻址的主机和卡通信。

DSR 寄存器：16 位驱动阶段寄存器，可用于在扩展操作条件中提高总线性能（取决于总线长度、传输速率等参数）。

SCR 寄存器：仅 SD/SD I/O 有这个寄存器。

25.4.4.2 命令

控制卡的命令有以下四种不同的类型：

命令类型	意义
广播命令（BC）	发送到所有卡，没有响应返回
带响应的广播命令（BCR）	发送到所有卡，同时收到从所有卡返回的响应
寻址（点对点）命令（AC）	发送到寻址卡上，SDIO_D 线上没有数据传输
寻址（点对点）的数据传输命令（ADTC）	发送到寻址卡上，SDIO_D 线上有数据传输

命令格式

所有命令格式都是 48 位的固定码长，需要 1.92us（25MHz）0.96us（50MHz）和 0.92us（52MHz）的发送时间。

表格 111 表命令格式

位	47	46	[45:40]	[39:8]	[7:1]	0
宽度	1	1	6	32	7	1
数值	0	1	-	-	-	1
说明	开始位	传输位	命令索引	参数	CRC7	结束位

SD I/O 支持两种响应类型，两种类型都有 CRC 错误检测。

- 48 位短响应
- 136 位长响应

表格 112 短响应格式

位	47	46	[45:40]	[39:8]	[7:1]	0
宽度	1	1	6	32	7	1
数值	0	0	-	-	-	1
说明	开始位	传输位	命令索引	参数	CRC7 或（1111111）	结束位

表格 113 长响应格式

位	135	134	[133:128]	[127:1]	0
宽度	1	1	6	127	1
数值	0	0	111111	-	1
说明	开始位	传输位	保留	CID 或 CSD	结束位

命令描述

表格 114 基本命令

命令索引	类型	参数	响应格式	简称	说明
CMD0	bc	[31:0]填充位	-	GO_IDLE_STATE	复位所有卡到空闲状态。

命令索引	类型	参数	响应格式	简称	说明
CMD1	bc	[31:0]OCR	R3	SEND_OP_COND	在空闲状态，请求卡通过 CMD 线发送响应（包含操作条件寄存器的内容）
CMD2	bcr	[31:0]填充位	R2	ALL_SEND_CID	请求任何卡通过 CMD 线发送 CID 数据（所有连接到主机的卡都会响应）
CMD3	bcr	[31:0]填充位	R6	SEND_RELATIVE_ADDR	请求卡发布新的相对卡地址（RCA）
CMD4	bc	[31:16]DSR [15:0]填充位	-	SET_DSR	设置所有卡的 DSR 寄存器。
CMD5	bcr	[31:25]保留位 [24]S18R [23:0]I/O OCR	R4	IO_SEND_OP_COND	仅适用于 I/O 卡。查询所有 IO 卡的电压范围。
CMD6	ac	[31:26]设为 0 [25:24]访问 [23:16]索引 [15:8]值 [7:3]设为 0 [2:0]命令集	R1b	SWITCH	仅适用于 MMC 卡。切换所选卡操作模式或修改 EXT_CSD 寄存器。
CMD7	ac	[31:16]RCA [15:0]填充位	R1b	SELECT/DESELECT_CARD	用于卡在状态上的切换。
CMD8	bcr	[31:12]保留位 [11:8]工作电压（VHS） [7:0]检查模式	R7	SEND_IF_COND	向 SD 卡发送接口条件，包括主机供电电压信息和询问卡是否支持电压。保留位应设为 0。
CMD9	ac	[31:16]RCA [15:0]填充位	R2	SEND_CSD	被选定的卡通过 CMD 发送到他的卡特定数据（CSD）
CMD10	ac	[31:16]RCA [15:0]填充位	R2	SEND_CID	被选定的卡通过 CMD 发送到他的卡标识（CID）

命令索引	类型	参数	响应格式	简称	说明
CMD12	ac	[31:0]填充位	R1b	STOP_TRANSMISSION	强制卡停止传输。
CMD13	ac	[31:16]RCA [15:0]填充位	R1	SEND_STATUS	被选定的卡发送它的状态寄存器。
CMD14	adtc	[31:0]填充位	R1	BUSTEST_R	主机从卡中读取反向的总线测试数据模式。
CMD15	ac	[31:16]RCA [15:0]填充位	-	GO_INACTIVE_STATE	将被选定的卡转换到非激活（Inactive）状态。
CMD19	adtc	[31:0]填充位	R1	BUSTEST_W	主机向卡发送总线测试模式。

表格 115 面向块的写命令

命令索引	类型	参数	响应格式	简称	说明
CMD23	ac	[31:16]设为 0 [15:0]块数目	R1	SET_BLOCK_COUNT	定义在后续的多个块或写命令中传输块的数目。
CMD24	adtc	[31:0]数据地址	R1	WRITE_BLOCK	按照 SET_BLOCKLEN 命令选择的长度写一个块。
CMD25	adtc	[31:0]数据地址	R1	WRITE_MULTIPLE_BLOCK	连续写入数据块，直到收到 STOP_TRANSMISSION 命令或达到了指定的块数目。
CMD26	adtc	[31:0]填充位	R1	PROGAM_CID	对卡识别寄存器编程。对每个卡只能发送一次该命令。该编程涉及到硬件更改，防止首次编程后以后的操作。通过该命令保留给生产厂商。
CMD27	adtc	[31:0]填充位	R1	PROGAM_CSD	对卡的 CSD 中可编程的位编程。
CMD28	ac	[31:0]数据地址	R1b	SET_WRITE_PROT	如果卡有写保护功能，该命令设置指定组的写保护位。写保护特性设置在卡的特殊数据区（WP_GRP_SIZE）

命令索引	类型	参数	响应格式	简称	说明
CMD29	ac	[31:0]数据地址	R1b	CLR_WRITE_PROT	如果卡有写保护功能，该命令将清除寻址组的写保护位。
CMD30	adtc	[31:0]写保护数据地址	R1	SEND_WRITE_PROT	如果卡有写保护功能，该命令请求卡发送写保护位状态。

25.4.5 特定操作

25.4.5.1 SD I/O 卡操作

SD I/O 卡（包括 IO 卡与组合卡）支持以下特定操作：

- 读等待操作
- 暂停/恢复操作
- 中断

只有设置 SDIO_DCTRL 的 SDIOEN 位时，SDIO 才支持这些操作；但读暂停除外，它不需要特殊的硬件操作。

SD I/O 读等待操作

读等待（RW）仅用于 SD I/O 的 1 位和 4 位模式。读等待操作允许一个主机给卡执行一个读多个块操作时发信号，以停止数据传输，同时允许主机发送命令到 SD I/O 卡内任何功能函数。

在收到第一个数据块之前即可以开始读等待过程，使能数据通道（DTEN 位置 1）、使能 SDIO 特定操作（SDIOEN 位置 1）、开始读等待,同时数据传输方式为从卡到 SDIO 主机，DPSM 将直接从空闲进入读等待状态。

在读等待状态时，2 个 SDIO_CK 时钟周期后，DPSM 驱动 SDIO_D2 为 0，在此状态，如果设置了 RWSTOP 位，则 DPSM 会在等待状态多停留 2 个 SDIO_CK 时钟周期，（根据 SDIO 规范）并在一个时钟周期中驱动 SDIO_D2 为 1。然后 DPSM 开始等待从卡里接收数据。在接收数据块时，即使设置了开始等待，DPSM 也不会进入读等待。

SDIO 卡不能以上读等待操作，SDIO 可以停止 SDIO_CK 进入读等待，在接收大当前数据块结束后 2 个 SDIO_CK 周期后，DPSM 停止时钟，在设置读等待开始位后恢复时钟。

图 125 使用 SDIO_D2 信号线的读等待操作

图 126 通过停止 SDIO_CLK 的读等待操作

SDIO 中断

当设置了 SDIOEN 位，SDIO 主机在 SDIO_D1 信号线上检测 SDIO 中断。

25.4.5.2 CE-ATA 特定操作

CE-ATA 设备有以下特定操作：

- 接收命令完成信号
- 发送命令完成关闭信号

只有在设置了 SDIO_CMD 的 ATACMD 位时，SDIO 才支持这些操作。

命令完成信号

CE-ATA 定义了命令完成信号，设备使用该信号通知主机 ATA 命令完成或者遇到错误，ATA 命令终止

命令完成关闭信号

主机可通过发送命令完成关闭信号取消设备返回命令完成信号的功能。

如果未设置 ATACMD 的“允许 CMD 结束位”，设置了 INTDIS 中的“非终端使能位”，则在收到一个短响应后的八个位周期后，发送命令完成关闭信号。

25.4.6 卡状态寄存器功能描述

响应格式 R1 包含了一个 32 位的卡状态域，这个域是用于向卡主机发送卡的状态信息（这些信息有可能存在本地的状态寄存器中）。除非特别说明，卡返回的状态始终是与之前的命令相关的。

下表定义了不同的状态信息：

表格 116 不同的状态信息类型

缩写	定义
E	错误位
S	状态位
R	检测位，并依据实际的命令响应而设置
X	检测位，在命令的执行中设置。 SDIO 卡主机通过发送状态命令读出这些位而查询卡的状态。

表格 117 清除条件

缩写	定义
A	依据卡的当前状态
B	始终与之前的命令相关。接收到正确的命令即可清除（具有一个命令的延迟）。
C	读即可清除

表格 118 卡状态

位	名称	类型	数值	说明	清除条件
1:0	保留给生产厂家的测试模式。				
2	保留给与应用相关的命令。				
3	AKE_SEQ_ERROR	ER	0=无错误 1=错误	验证的顺序有错误。	C
4	保留给 SD I/O 卡				
5	APP_CMD	SR	0=不允许 1=允许	卡期望 ACMD，或指示命令已经被解释为 ACMD 命令。	C
6	保留				
7	SWITCH_ERROR	EX	0=无错误 1=转换错	卡没有按照 SWITCH 命令的要求转换到希望的模式。	B
8	READY_FOR_DATA	SR	0=未就绪 1=就绪	与总线上的缓冲器空的信号相对应。	

位	名称	类型	数值	说明	清除条件
12:9	CURRENT_STATE	SR	0=空闲 1=就绪 2=识别 3=待机 4=发送 5=数据 6=接收 7=编程 8=断开 9=忙测试 10~15=保留	当收到命令时卡内状态机的状态。如果命令的执行导致状态的变化，这个变化将会在下个命令的响应中反映出来。这四个位按十进制数 0 至 15 解释。	B
13	ERASE_RESET		0=清除 1=设置	因为收到一个擦除顺序之外的命令（非 CMD35 、 CMD36、 CMD38 或 CMD13 命令），进入擦除过程的序列被中止。	C
14	CARD_ECC_DISABLED	SX	0=允许 1=不允许	执行命令时没有使用内部的 ECC。	A
15	WP_ERASE_SKIP	EX	0=未保护 1=已保护	遇到已经存在的写保护数据块，仅有部分地址空间被擦除。	C
16	CID/CSD_OVERWRITE	EX	0=无错误 1=错误	可以是任何一个下述的错误： 已经写入了 CID 寄存器，不能覆盖 CSD 的只读部分与卡的内容不匹配 试图进行拷贝或永久写保护的反向操作，即恢复原状或解除写保护。	C
17	保留				
18	保留				
19	ERROR	EX	0=无错误 1=错误	产生了与执行上一个主机命令相关的（标准中未定义） 卡内部的错误（例如：读或写错误）。	C
20	CC_ERROR	ER	0=无错误 1=错误	（标准中未定义）卡内部发生错误，与主机的命令无关。	C
21	CARD_ECC_FAILED	EX	0=成功 1=失败	卡的内部实施了 ECC 校验，但在更正数据时失败。	C
22	ILLEGAL_COMMAND	ER	0=无错误 1=错误	对于当前的卡状态，命令非法。	B

位	名称	类型	数值	说明	清除条件
23	COM_CRC_ERROR	ER	0=无错误 1=错误	之前的命令中 CRC 校验错误。	B
24	LOCK_UNLOCK_FAILED	EX	0=无错误 1=错误	在上锁/解锁中有命令的顺序错误或检测到密码错误。	C
25	CARD_IS_LOCKED	SR	0=卡未锁 1=卡已锁	当设置了该位，表示卡已经被锁住。	A
26	WP_VIOLATION	EX	0=无错误 1=错误	试图对一个写保护的数据块编程。	C
27	ERASE_PARAM	EX	0=无错误 1=错误	擦除时选择了非法的擦除组。	C
28	ERASE_SEQ_ERROR		0=无错误 1=错误	发送擦除命令的顺序错误。	C
29	BLOCK_LEN_ERROR		0=无错误 1=错误	SET_BLOCKLEN 命令的参数超出了卡的最大允许范 围，或先前定义的数据块长度对于当前命令来说是非法的（例如：主机发出一个写命令，当前的块长度小于卡所允许的最小长度，同时又不允许写入部分数据块）。	C
30	ADDRESS_MISALIGN		0=无错误 1=错误	命令中的地址参数（与当前的数据块长度对照）定义的第一个数据块未与卡的物理块对齐。 一个多数据块或数据流读/写操作（即使从一个合法的地址开始）试图读或写未与物理块对齐的数据块。	C
31	ADDRESS_OUT_OF_RANGE	ERX	0=无错误 1=错误	命令中的地址参数超出了卡的允许范围。 一个多数据块或数据流读/写操作（即使从一个合法的地址开始）试图读或写超出卡的容量的部分。	C

25.4.6.1 SD 状态寄存器功能描述

SD 状态包含与 SD 存储器卡特定功能相关的状态位和一些与未来应用相关的状态位， SD 状态的长度是一个 512 位的数据块。收到 ACMD13 命令（CMD55，然后是 CMD13）后，这个寄存器的内容被传送到 SDIO 卡主机。只有卡处于传输状态时（卡已被选择）才能发送 ACMD13 命令。

下表定义了不同的 SD 状态寄存器信息。

表格 119 SD 状态寄存器信息

缩写	定义
E	错误位
S	状态位
R	检测位，并依据实际的命令响应而设置
X	检测位，在命令的执行中设置。 SDIO 卡主机通过发送状态命令读出这些位而查询卡的状态。

注：表中有关类型和清除条件域的缩写定义见。

表格 120 清除条件

缩写	定义
A	依据卡的当前状态
B	始终与之前的命令相关。接收到正确的命令即可清除（具有一个命令的延迟）。
C	读即可清除置

表格 121 SD 状态

位	名称	类型	数值	说明	清除条件
311:0	保留给生产厂商				
399:312	保留				
401:400	ERASE_OFFSET				
407:402	ERASE_TIMEOUT	S R	在擦除时增加的固定偏移数值	（见以下说明）	A
423:408	ERASE_SIZE	S R	擦除 UNIT_OF_ERASE_AU 指定的范围的超时数值	（见以下说明）	A
427:424	保留	S R	一次可以擦除的 AU 数目	（见以下说明）	A
431:428	AU_SIZE	S R	AU 的大小（见以下说明）	（见以下说明）	A
439:432	PERFORMANCE_MOVE	S R	以 1MB/秒为单位的传输性能（见以下说明）	（见以下说明）	A
447:440	SPEED_CLASS	S R	卡的速度类型（见以下说明）	（见以下说明）	A
479:448	SIZE_OF_PROTECTED_AREA	S R	受保护的区域大小（见以下说明）	（见以下说明）	A

位	名称	类型	数值	说明	清除条件
495:480	SD_CARD_TYPE	S R	‘00xxh’= 在物理规范版本 1.01 ~2.00 的 SD 存储器卡（’x’表示任意值）。已定义的卡有： ‘0000’= 通用 SD 读写卡 ‘0001’= SD ROM 卡	这个域的低 8 位可以在未来定义 SD 存储卡的不同变种（每个位可以用于定义不同的 SD 类型）。高 8 位可以用于定义那些不遵守当前的 SD 物理层规范的 SD 卡。	A
508:496	保留				
509	SECURED_MODE	S R	0= 未处于保密模式 1= 处于保密模式	卡处于保密操作模式（详见“SD 保密规范 “）。	A
511:510	DAT_BUS_WIDTH	S R	00=1（默认） 01=保留 10= 4 位宽 11=保留	由 SET_BUS_WIDTH 命令定义的当前数据总线宽度。	A

SIZE_OF_PROTECTED_AREA

标准容量卡和高容量卡设置该位的方式不同：

- 对于标准容量卡，受保护区域的容量由下式计算：
  受保护区域=SIZE_OF_PROTECTED_AREA * MULT * BLOCK_LEN
  SIZE_OF_PROTECTED_AREA 的单位是 MULT * BLOCK_LEN。
- 对于高容量卡，受保护区域的容量由下式计算：
  受保护区域=SIZE_OF_PROTECTED_AREA
  SIZE_OF_PROTECTED_AREA 的单位是字节。

SPEED_CLASS

这 8 位指示速度的类型和可以通过计算 PW/2 的数值（PW 是写的性能）。

表格 122 速度类型代码

SPEED_CLASS	数值定义
00h	类型 0
01h	类型 2
02h	类型 4
03h	类型 6
04h~FFh	保留

PERFORMANCE_MOVE

这 8 位以 1MB/秒为单位指示移动性能（Pm）。如果卡不用 RU（纪录单位）移动数据，应该认为 Pm 是无穷大。设置这个域为 FFh 表示无穷大。

表格 123 移动性能代码

PERFORMANCE_MOVE	数值定义
00h	未定义
01h	1MB/秒
02h	2MB/秒
……	……
Feh	254MB/秒
FFh	无穷大

AU_SIZE

这 4 位指示 AU 的长度，数值是 16K 字节为单位 2 的幂次的倍数。

表格 124 AU_SIZE 代码

AU_SIZE	数值定义
00h	未定义
01h	16KB
02h	32KB
03h	64KB
04h	128KB
05h	256KB
06h	512KB
07h	1MB
08h	2MB
09h	4MB
Ah~Fh	保留

依据卡的容量，最大的 AU 长度由下表定义。卡可以在 RU 长度和最大的 AU 长度之间设置任意的 AU 长度。

表格 125 最大的 AU 长度

容量	最大的 AU 长度
16MB~64MB	512KB
128MB~256MB	1MB
512MB	2MB
1GB~32GB	4MB

ERASE_SIZE

这个 16 位域给出了 N_{ERASE} 的值，当 N_{ERASE} 个 AU 被擦除时，ERASE_TIMEOUT 定义超时时间。

如果主机能够确定某次擦除中 N_{ERASE} 的值，那么就可以显示擦除的进度。

表格 126 ERASE_SIZE 代码

ERASE_SIZE	数值定义
0000h	不支持擦除的超时计算
0001h	1 个 AU
0002h	2 个 AU
0003h	3 个 AU
……	……
FFFFh	65535 个 AU

ERASE_TIMEOUT

这 6 位给出了 T_{ERASE}，当多个 AU 被擦除时，T_{ERASE} 给出了从偏移量算起的擦除超时。

表格 127 擦除超时代码

ERASE_TIMEOUT	数值定义
00	不支持擦除的超时计算
01	1 秒
02	2 秒
03	3 秒
……	……
63	63 秒

ERASE_OFFSET

这 2 位给出了 TOFFSET，当 ERASE_SIZE 和 ERASE_TIMEOUT 同为 0 时这个数值没有意义。

表格 128 擦除偏移代码

ERASE_OFFSET	数值定义
0	0 秒
1	1 秒
2	2 秒
3	3 秒

25.5 寄存器地址映射

表格 129 SDIO 寄存器地址映射

寄存器名	描述	偏移地址
SDIO_PWRCTRL	SDIO 电源控制寄存器	0x00
SDIO_CLKCTRL	SDIO 时钟控制寄存器	0x04
SDIO_ARG	SDIO 参数寄存器	0x08
SDIO_CMD	SDIO 命令寄存器	0x0C
SDIO_CMDRES	SDIO 命令响应寄存器	0x10
SDIO_RESx	SDIO 响应 x 寄存器	0x14 + 4*（x-1），其中 x=1 to 4
SDIO_DATATIME	SDIO 数据定时器寄存器	0x24
SDIO_DATALEN	SDIO 数据长度寄存器	0x28
SDIO_DCTRL	SDIO 数据控制寄存器	0x2C
SDIO_DCNT	SDIO 数据计数器寄存器	0x30
SDIO_STS	SDIO 状态寄存器	0x34
SDIO_ICF	SDIO 清除中断寄存器	0x38
SDIO_MASK	SDIO 中断屏蔽寄存器	0x3C
SDIO_FIFOCNT	SDIO 计数器寄存器	0x48
SDIO_FIFODATA	SDIO 数据 FIFO 寄存器	0x80

25.6 寄存器功能描述

设备通过可以在 AHB 上操作的 32 位控制寄存器与系统通信，必须以字（32 位）的方式操作这些外设寄存器。

25.6.1 SDIO 电源控制寄存器（SDIO_PWRCTRL）

偏移地址：0x00

复位值：0x0000 0000

位/域	名称	R/W	描述
1:0	PWRCTRL	R/W	控制电源（Power Supply Control） 选择卡时钟的当前功能状态。 00：电源关闭，卡的时钟停止。 01：保留。 10：保留的上电状态。 11：上电状态，卡的时钟开启。
31:2	保留		

注：写数据后的 7 个 HCLK 时钟周期内，不能写入这个寄存器。

25.6.2 SDIO 时钟控制寄存器（SDIO_CLKCTRL）

偏移地址：0x04

复位值：0x0000 0000

SDIO_CLKCTRL 寄存器控制 SDIO_CLK 输出时钟。

位/域	名称	R/W	描述
7:0	CLKDIV	R/W	时钟分频系数（Clock Divide Factor） 这个域定义了输入时钟（SDIOCLK）与输出时钟（SDIO_CLK）间的分频系数： SDIO_CLK 频率=SDIOCLK/[CLKDIV + 2]。
8	CLKEN	R/W	使能时钟（Clock Enable） 0：禁止 1：使能
9	PWRSAV	R/W	配置省电模式（Power Saving Mode Configuration） 通过禁止 SDIO_CLK 在总线活动外输出来减少功耗。 0：允许 1：禁止
10	BYPASSEN	R/W	使能旁路时钟分频器（Clock Divider Bypass Enable） 驱动 SDIO_CLK 输出信号前，需要对 SDIOCLK 分频，但如果旁路了分频器，那么 SDIOCLK 直接驱动 SDIO_CLK 输出信号。 0：禁止 1：使能
12:11	WBSEL	R/W	选择总线模式宽度（Wide Bus Mode Select） 选择不同位的总线模式，对应不同 SDIO_D 位。 00：默认，使用 SDIO_D0 01：4 位，使用 SDIO_D[3:0] 10：8 位，使用 SDIO_D[7:0] 11：保留
13	DEPSEL	R/W	选择 SDIO_CLK 相位（SDIO_CLK Dephasing Select） 选择 SDIOCLK 上升沿或者下降沿产生 SDIO_CLK。 0：上升沿 1：下降沿
14	HFCEN	R/W	使能硬件流控制（HW Flow Control Enable） 0：禁止 1：使能
31:15	保留		

注：

（1）当 SD/SD I/O 卡或多媒体卡处于识别模式， SDIO_CLK 的频率必须低于 400kHz。

（2）当所有卡都被赋予了相应的地址后，时钟频率可以改变到卡总线允许的最大频率。

（3）写数据后的 7 个 HCLK 时钟周期内不能写入这个寄存器。对于 SD I/O 卡，在读等待期间可以停止 SDIO_CLK，此时 SDIO_CLKCTRL 寄存器不控制 SDIO_CLK。

25.6.3 SDIO 参数寄存器（SDIO_ARG）

偏移地址：0x08

复位值：0x0000 0000

命令参数也是命令的一部分，SDIO_ARG 寄存器包含 32 位命令参数并且和命令一起发送到卡中。

位/域	名称	R/W	描述
31:0	CMDARG	R/W	命令参数（Command Argument） 存储命令参数。

25.6.4 SDIO 命令寄存器（SDIO_CMD）

偏移地址：0x0C

复位值：0x0000 0000

SDIO_CMD 寄存器包含命令索引和命令类型位。

位/域	名称	R/W	描述
5:0	CMDINDEX	R/W	命令索引（Command Index） 命令索引作为命令的一部分与其一起发送到卡中。
7:6	WAITRES	R/W	等待响应（Wait for Response） 指示 CPSM 是否需要等待响应，如果需要等待响应，则指示响应类型。 00：无响应，期待 CMDSENT 标志 01：短响应，期待 CMDREND 或 CCRCFAIL 标志 10：无响应，期待 CMDSENT 标志 11：长响应，期待 CMDREND 或 CCRCFAIL 标志
8	WAITINT	R/W	CPSM 等待中断请求（CPSM Waits for Interrupt Request） CPSM 开启或关闭命令超时控制并等待中断请求。 0：开启 1：关闭
9	WENDDATA	R/W	CPSM 等待数据传输结束（CmdPend 内部信号）（CPSM Waits for Ends of Data Transfer（CmdPend Internal Signal）） 0：无效 1：CPSM 在开始发送一个命令之前等待数据传输结束。
10	CPSMEN	R/W	使能命令通道状态机（CPSM）（Command Path State Machine（CPSM）Enable） 使能 CPSM。 0：禁止 1：使能
11	SDIOSC	R/W	SD I/O 暂停命令（SD I/O Suspend Command） 0：无效 1：将要发送的命令是一个暂停命令（只能用于 SD I/O 卡）。
12	CMDCPEN	R/W	使能 CMD 完成（Enable CMD Completion） 使能命令完成信号。 0：禁止 1：使能
13	INTEN	R/W	使能中断（Interrupt Enable） 0：使能 1：禁止
14	ATACMD	R/W	CE-ATA 命令（CE-ATA Command） 0：无效 1：CPSM 转至 CMD61
31:15	保留		

注：

（1）写数据后的 7 个 HCLK 时钟周期内不能写入这个寄存器。

（2）多媒体卡可以发送 2 种响应：48 位长的短响应，或 136 位长的长响应。SD 卡和 SD I/O 卡只能发送短响应，参数可以根据响应的类型而变化，软件将根据发送的命令区分响应的类型。CE-ATA 设备只发送短响应。

25.6.5 SDIO 命令响应寄存器（SDIO_CMDRES）

偏移地址：0x10

复位值：0x0000 0000

位/域	名称	R/W	描述
5:0	CMDRES	R	响应的命令索引（Response Command Index） 存储最后收到的命令响应中的命令索引。
31:6	保留		

25.6.6 SDIO 响应 x 寄存器（SDIO_RESx）

偏移地址：0x14 + 4*（x-1），其中 x=1…4

复位值：0x0000 0000

SDIO_RES1/2/3/4 寄存器包含卡的状态，即收到响应的部分信息。

位/域	名称	R/W	描述
31:0	CARDSTSx	R	见下表。

根据响应状态，卡的状态长度是 32 位或 127 位。

表格 130 响应类型和 SDIO_RESx 寄存器

寄存器	短响应	长响应
SDIO_RES1	卡状态[31:0]	卡状态[127:96]
SDIO_RES2	不用	卡状态[95:64]
SDIO_RES3	不用	卡状态[63:32]
SDIO_RES4	不用	卡状态[31:1]

总是先收到卡状态的最高位，SDIO_RES3 寄存器的最低位始终为 0。

25.6.7 SDIO 数据定时器寄存器（SDIO_DATATIME）

偏移地址：0x24

复位值：0x0000 0000

位/域	名称	R/W	描述
31:0	DATATIME	R/W	数据超时时间（Data Timeout Period） 记录以卡总线时钟周期为单位的数据超时时间。

注:在写入数据控制寄存器进行数据传输之前，必须先写入数据定时器寄存器和数据长度寄存器。

25.6.8 SDIO 数据长度寄存器（SDIO_DATALEN）

偏移地址：0x28

复位值：0x0000 0000

位/域	名称	R/W	描述
24:0	DATALEN	R/W	数据长度（Data Length） 要传输的数据字节长度。
31:25	保留		

注:对于块数据传输，SDIO_DATALEN 中的数值必须是数据块长度的倍数。在写入 SDIO_DCTRL 进行数据传输之前，必须先写入 SDIO_DATATIME 和 SDIO_DATALEN。

25.6.9 SDIO 数据控制寄存器（SDIO_DCTRL）

偏移地址：0x2C

复位值：0x0000 0000

SDIO_DCTRL 寄存器控制数据通道状态机（DPSM）。

位/域	名称	R/W	描述
0	DTEN	R/W	使能数据传输（Data Transfer Enabled） 0：禁止 1：使能
1	DTDRCFG	R/W	配置数据传输方向（Data Transfer Direction Configuration） 0：控制器至卡 1：卡至控制器
2	DTSEL	R/W	选择数据传输模式（Data Transfer Mode Select） 0：块数据传输 1：流数据传输
3	DMAEN	R/W	使能 DMA（DMA Enable） 0：禁止 1：使能
7:4	DBSIZE	R/W	数据块长度（Data Block Size） 定义数据块长度： 0000：块长度=20=1 字节 0001：块长度=21=2 字节 0010：块长度=22=4 字节 0011：块长度=23=8 字节 0100：块长度=24=16 字节 0101：块长度=25=32 字节 0110：块长度=26=64 字节 0111：块长度=27=128 字节 1000：块长度=28=256 字 1001：块长度=29=512 字 1010：块长度=210=1024 字节 1011：块长度=211=2048 字节 1100：块长度=212=4096 字节 1101：块长度=213=8192 字节 1110：块长度=214=16384 字节 1111：保留
8	RWSTR	R/W	读等待开始（Read Wait Start） 0：无效 1：开始读等待操作

位/域	名称	R/W	描述
9	RWSTOP	R/W	使能读等待停止（Read Wait Stop Enable） 如果设置了 RWSTR，可使能停止读操作。 0：禁止 1：使能
10	RDWAIT	R/W	读等待模式（Read Wait Mode） 0：控制停止 SDIO_D2 1：控制使用 SDIO_CK
11	SDIOF	R/W	使能 SD I/O 功能（SD I/O Enable Functions） 如果设置了该位，则 DPSM 执行 SD I/O 卡特定的操作。
31:12	保留		

注:写数据后的 7 个 HCLK 时钟周期内不能写入这个寄存器。

25.6.10 SDIO 数据计数器寄存器（SDIO_DCNT）

偏移地址：0x30

复位值：0x0000 0000

当 DPSM 从空闲状态进入 Wait_R 或 Wait_S 状态时，SDIO_DCNT 从 SDIO_DATALEN 加载数值，数据传输过程中，该计数器向下计数至 0，然后 DPSM 进入空闲状态并设置 DATAEND 标志。

位/域	名称	R/W	描述
24:0	DATACNT	R	数据计数数值（Data Count Number） 存储待传输的数据字节数。
31:25	保留		

注:只能在数据传输结束时读这个寄存器。

25.6.11 SDIO 状态寄存器（SDIO_STS）

偏移地址：0x34

复位值：0x0000 0000

SDIO_STS 是一个只读寄存器，它包含两类标志：

（1）静态标志（位[23:22、10:0]）：写入 SDIO 中断清除寄存器，可以清除这些位。

（2）动态标志（位[21:11]）：这些位的状态变化根据它们对应的那部分逻辑而变化。

位/域	名称	R/W	描述
0	COMRESP	R	已收到命令响应（CRC 检测失败）（Command Response Received）
1	DBDR	R	已发送/接收数据块（CRC 检测失败）（Data Block Sent/Received）
2	CMDRESTO	R	命令响应超时（Command Response Timeout） 命令超时时间为 64 个 SDIO_CLK 时钟周期。
3	DATATO	R	数据超时（Data Timeout）
4	TXUDRER	R	发送 FIFO 下溢错误（Transmit FIFO Underrun Error）
5	RXOVRER	R	接收 FIFO 上溢错误（Received FIFO Overrun Error）

位/域	名称	R/W	描述
6	CMDRES	R	已接收到响应（CRC 检测成功）（Command Response）
7	CMDSENT	R	命令已发送（不需要响应）（Command Sent（No Response Required））
8	DATAEND	R	数据结束（数据计数器，SDIO_DCNT=0）
9	SBE	R	在总线模式宽度下，没有在所有数据信号上检测到起始位（Start Bit Not Detected On All Data Signals In Wide Bus Mode）
10	DBCP	R	已发送/接收数据块（Data Block Sent/Received）
11	CMDACT	R	正在传输命令（Command Transfer In Progress）
12	TXACT	R	正在发送数据（Data Transmit In Progress）
13	RXACT	R	正在接收数据（Data Receive In Progress）
14	TXFHF	R	发送 FIFO 半空（Transmit FIFO Half Empty）： FIFO 中至少还可以写入 8 个字。
15	RXFHF	R	接收 FIFO 半满（Receive FIFO Half Full） FIFO 中至少还有 8 个字。
16	TXFF	R	发送 FIFO 满（Transmit FIFO Full）
17	RXFF	R	接收 FIFO 满（Receive FIFO Full） 若使用了硬件流控制，当 FIFO 还差 2 个字满时， RXFF 信号变为有效。
18	TXFE	R	发送 FIFO 空（Transmit FIFO Empty） 若使用了硬件流控制，当 FIFO 包含 2 个字时， TXFE 信号变为有效。
19	RXFE	R	接收 FIFO 空（Receive FIFO Empty）
20	TXDA	R	在发送 FIFO 中的数据可用（Data Available In Transmit FIFO）
21	RXDA	R	在接收 FIFO 中的数据可用（Data Available In Receive FIFO）
22	SDIOINT	R	收到 SDIO 中断（SDIO Interrupt Received）
23	ATAEND	R	在 CMD61 接收到 CE-ATA 命令完成信号（CE-ATA Command Completion Signal Received For CMD61）
31:24	保留		

25.6.12 SDIO 清除中断寄存器（SDIO_ICF）

偏移地址：0x38

复位值：0x0000 0000

SDIO_ICF 是一个只写寄存器，在对应寄存器位置位将清除 SDIO_STS 状态寄存器中的对应位。

位/域	名称	R/W	描述
0	DBCE	R/W	DBCE 标志清除位（DBCE Flag Clear） 清除 DBCE 标志。 0：无效 1：清除
1	CRCE	R/W	CRCE 标志清除位（CRCE Flag Clear） 清除 CRCE 标志。 0：无效 1：清除

位/域	名称	R/W	描述
2	CRTO	R/W	CRTO 标志清除位（CRTO Flag Clear） 清除 CRTO 标志。 0：无效 1：清除
3	DTO	R/W	DTO 标志清除位（DTO Flag Clear） 清除 DTO 标志。 0：无效 1：清除
4	TXFUE	R/W	TXFUE 标志清除（TXFUE Flag Clear） 清除 TXFUE 标志。 0：无效 1：清除
5	RXFOE	R/W	RXFOE 标志清除位（RXFOE Flag Clear） 清除 RXFOE 标志。 0：无效 1：清除
6	CMDRES	R/W	CMDRES 标志清除位（CMDRES Flag Clear） 清除 CMDRES 标志。 0：无效 1：清除
7	CMDSENT	R/W	CMDSENT 标志清除位（CMDSENT Flag Clear） 清除 CMDSENT 标志。 0：无效 1：清除
8	DATAEND	R/W	DATAEND 标志清除位（DATAEND Flag clear） 清除 DATAEND 标志。 0：无效 1：清除
9	SBE	R/W	SBE 标志清除位（SBE Flag Clear） 清除 SBE 标志。 0：无效 1：清除
10	DBCP	R/W	DBCP 标志清除位（DBCP Flag Clear） 清除 DBCP 标志。 0：无效 1：清除
21:11	保留		
22	SDIOIT	R/W	SDIOIT 标志清除位（SDIOIT flag clear bit） 清除 SDIOIT 标志。 0：无效 1：清除
23	ATAEND	R/W	ATAEND 标志清除位（ATAEND flag clear bit） 清除 ATAEND 标志。 0：无效 1：清除

位/域	名称	R/W	描述
31:24	保留		

25.6.13 SDIO 中断屏蔽寄存器（SDIO_MASK）

偏移地址：0x3C

复位值：0x0000 0000

由软件置位或清零。

在对应位置 1，SDIO_MASK 中断屏蔽寄存器决定哪一个状态位产生中断。

位/域	名称	R/W	描述
0	CCRCFAIL	R/W	使能命令块 CRC 检测失败产生中断（Command CRC Fail Interrupt Enable） 允许/关闭命令块 CRC 检测失败中断。 0：禁止 1：使能
1	DCRCFAIL	R/W	使能数据块 CRC 检测失败产生中断（Data CRC Fail Interrupt Enable） 允许/关闭数据块 CRC 检测失败中断。 0：禁止 1：使能
2	CMDTO	R/W	使能命令超时产生中断（Command Timeout Interrupt Enable） 允许/关闭命令超时中断。 0：禁止 1：使能
3	DATATO	R/W	使能数据超时产生中断（Data timeout interrupt enable） 允许/关闭数据超时中断。 0：禁止 1：使能
4	TXURER	R/W	发送 FIFO 下溢错误产生中断（Tx FIFO Underrun Error Interrupt Enable） 允许/关闭发送 FIFO 下溢错误中断。 0：禁止 1：使能
5	RXORER	R/W	使能接收 FIFO 上溢错误产生中断（Rx FIFO Overrun Error Interrupt Enable） 允许/关闭接收 FIFO 上溢错误中断。 0：禁止 1：使能
6	CMDRESRC	R/W	使能接收到响应产生中断（Command Response Received Interrupt Enable） 允许/关闭接收到响应中断。 0：禁止 1：使能
7	CMDSENT	R/W	使能命令已发送产生中断（Command Sent Interrupt Enable） 由软件设置/清除该位，允许/关闭命令已发送中断。 0：禁止 1：使能

位/域	名称	R/W	描述
8	DATAEND	R/W	使能数据传输结束产生中断（Data End Interrupt Enable） 允许/关闭数据传输结束中断。 0：禁止 1：使能
9	STRTER	R/W	使能起始位错误产生中断（Start Bit Error Interrupt Enable） 允许/关闭起始位错误中断。 0：禁止 1：使能
10	DBEND	R/W	使能数据块传输结束产生中断（Data Block End Interrupt Enable） 允许/关闭数据块传输结束中断。 0：禁止 1：使能
11	CMDACT	R/W	使能正在传输命令产生中断（Command Acting Interrupt Enable） 允许/关闭正在传输命令中断。 0：禁止 1：使能
12	TXACT	R/W	使能正在发送数据产生中断（Data Transmit Acting Interrupt Enable） 允许/关闭正在发送数据中断。 0：禁止 1：使能
13	RXACT	R/W	正在接收数据产生中断（Data receive acting interrupt enable） 允许/关闭正在接收数据中断。 0：禁止 1：使能
14	TXHFERT	R/W	使能发送 FIFO 半空产生中断（Tx FIFO Half Empty Interrupt Enable） 允许/关闭发送 FIFO 半空中断。 0：禁止 1：使能
15	RXHFFUL	R/W	使能接收 FIFO 半满产生中断（Rx FIFO Half Full Interrupt Enable） 允许/关闭接收 FIFO 半满中断。 0：禁止 1：使能
16	TXFUL	R/W	使能发送 FIFO 满产生中断（Tx FIFO full Interrupt Enable） 允许/关闭发送 FIFO 满中断。 0：禁止 1：使能
17	RXFUL	R/W	使能接收 FIFO 满产生中断（Rx FIFO full interrupt enable） 允许/关闭接收 FIFO 满中断。 0：禁止 1：使能
18	TXEPT	R/W	使能发送 FIFO 空产生中断（Tx FIFO Empty Interrupt Enable） 允许/关闭发送 FIFO 空中断。 0：禁止 1：使能

位/域	名称	R/W	描述
19	RXFEIE	R/W	使能接收 FIFO 空产生中断（Rx FIFO Empty Interrupt Enable） 允许/关闭接收 FIFO 空中断。 0：禁止 1：使能
20	TXDAVB	R/W	使能发送 FIFO 中的数据有效产生中断（Data Available in Tx FIFO Interrupt Enable） 允许/关闭发送 FIFO 中的数据有效中断。 0：禁止 1：使能
21	RXDAVB	R/W	使能接收 FIFO 中的数据有效产生中断（Data Available in Rx FIFO Interrupt Enable） 允许/关闭接收 FIFO 中的数据有效中断。 0：禁止 1：使能
22	SDIOINTREC	R/W	使能允许 SDIO 模式中断已接收中断（SDIO Mode Interrupt Received Interrupt Enable） 允许/关闭 SDIO 模式中断已接收中断功能。 0：禁止 1：使能
23	ATACLPREC	R/W	使能允许接收到 CE-ATA 命令完成信号产生中断（CE-ATA Command Completion Signal Received Interrupt Enable） 允许/关闭在收到 CE-ATA 命令完成信号产生中断功能。 0：禁止 1：使能
31:24	保留		

25.6.14 SDIO FIFO 计数器寄存器（SDIO_FIFOCNT）

偏移地址：0x48

复位值：0x0000 0000

位/域	名称	R/W	描述
23:0	FIFOCNT	R	接收和发送 FIFO 数目（Receive And Transmit FIFO Number） 将要写入 FIFO 或将要从 FIFO 读出数据字的数目。 注：如果数据长度不是 4 的倍数，则最后剩下的字节可以当成一个字处理。
31:24	保留		

25.6.15 SDIO 数据 FIFO 寄存器（SDIO_FIFODATA）

偏移地址：0x80

复位值：0x0000 0000

位/域	名称	R/W	描述
31:0	DATA	R/W	接收或发送 FIFO 数据（Receive And Transmit FIFO Data） 将要写入 FIFO 或将要从 FIFO 读出数据。

26 全速 USBD 接口设备（USBD）

26.1 简介

该外设符合全速 USB2.0 接口设备的协议标准及技术规范，实现了 USBD 总线与 APB1 总线之间接口对接，同时该外设还还具备最多 8 个端点可配置、同步传输和端点的双缓冲机制，以及 CRC 校验、USBD 挂起/恢复操作和帧锁定时钟脉冲生成等功能。

26.2 功能描述

26.2.1 主要功能

USBD 模块实现了微控制器与 PC 之间符合 USB2.0 接口协议的通信接口功能。该功能实现过程中用到了一个可被 USBD 外设直接访问的共享专用数据缓冲区，这个数据缓冲区的大小由配置的端点数目和不同端点的最大数据分组大小决定，最多可使用 8 个双向或 16 各单向端点，每个端点最多可以使用 512 字节，USBD 和 PC 通信的数据帧格式由硬件完成。

不同的端点需有一个独立的缓冲区描述块，用于描述该端点的缓冲区地址、大小、以及传输数据的字节长度。芯片内核通过响应 USBD 中断，在中断中通过判断所设定的端点状态寄存器的置位情况，进入不同的中断处理子函数，处理端点需要得到的服务，进行相应类型的数据传输。

26.2.2 USBD 与 CAN 同时使用的说明

虽然实际上有 2 个一模一样的 USBD（引脚也相同），但是它们不能同时使用，所以相当于只有 1 个。用户可以通过重映射（引脚的复用功能）实现“同时使用”，具体使用说明如下：

- USBD1 与 CAN2 可以同时使用；USBD2 与 CAN1 可以同时使用
- USBD1 与 USBD2 不可以同时使用
- CAN1 与 CAN2 可以同时使用

26.3 寄存器地址映射

表格 131 USBD 寄存器地址映射

寄存器名	描述	偏移地址
USBD_CTRL	USBD 控制寄存器	0x40
USBD_INTSTS	USBD 中断状态寄存器	0x44
USBD_FRANUM	USBD 帧状态寄存器	0x48
USBD_ADDR	USBD 设备地址寄存器	0x4C
USBD_BUFFTB	USBD 分组缓冲区描述表地址寄存器	0x50
USBD_SWITCH	USBD 切换寄存器	0x100

寄存器名	描述	偏移地址
USBD_EPn(n=[0…7])	USBD 端点 n 寄存器	0x00 至 0x1C
USBD_TXADDRn	发送缓冲区地址寄存器 n	[USBD_ BUFFTB]+n×16
USBD_TXCNTn	发送数据字节数寄存器 n	[USBD_ BUFFTB]+n×16+4
USBD_RXADDRn	接收缓冲区地址寄存器 n	[USBD_ BUFFTB]+n×16+8
USBD_RXCNTn	接收数据字节数寄存器 n	[USBD_ BUFFTB]+n×16+12

26.4 寄存器功能描述

USBD 模块的寄存器有以下三类:

（1） 通用类寄存器：中断寄存器和控制寄存器；

（2） 端点类寄存器：端点配置寄存器和状态寄存器；

（3） 缓冲区描述表类寄存器：用来确定数据分组存放地址的寄存器。

缓冲区描述表类寄存器的基地址由 USBD_ BUFFTB 寄存器指定，所有其它寄存器的基地址为 USBD 模块的基地址 0x4000 5C00。由于 APB1 总线按 32 位寻址，因此所有的 16 位寄存器的地址都是按 32 位字对齐的。同样的地址对齐方式也用于从 0x4000 6000 开始的分组缓冲存储区。

26.4.1 通用寄存器

用于定义 USBD 模块的工作模式，中断的处理、设备的地址和读取当前帧的编号。

26.4.1.1 USBD 控制寄存器（USBD_CTRL）

偏移地址：0x40

复位值：0x0003

位/域	名称	R/W	描述
0	FORRST	R/W	强制复位 USBD（Force USBD Reset） 强制复位 USBD，在软件清 0 前，USBD 一直保持复位状态。 0：清除 1：强制复位
1	PWRDOWN	R/W	关闭 USBD 模块（Power Down） 此模式用于彻底关闭 USBD 模块。当此位被置 1 时，不能使用 USBD 模块。 0：打开 1：关闭
2	LPWREN	R/W	使能低功耗模式（Lowpower Mode Enable） 开启此模式后，在 USBD 挂起时，系统时钟将会停止或者降低到一定的频率，且除了外接上拉电阻的供电，其它的静态功耗都被关闭。 USBD 的任一唤醒事件都可以将此位清 0。 0：正常模式 1：低功耗模式

位/域	名称	R/W	描述
3	FORSUS	R/W	强制挂起（Force Suspend） 当 USBD 总线上 3ms 没有数据通信时，将触发 SUSP 中断，此时软件需将此位置 1。 0：无效 1：强制挂起 USBD
4	WUPREQ	R/W	唤醒请求（Wakeup Request） 向 PC 主机发送唤醒请求，如果此位在 1ms 到 15ms 内保持有效，主机将对 USBD 实行唤醒操作。 0：无效 1：有效
7:5	保留		
8	ESOFIEN	R/W	使能帧首丢失中断（Expected Start of Frame Interrupt Enable） 0：禁止 1：使能
9	SOFIEN	R/W	使能帧首中断（Start of Frame Interrupt Enable） 0：禁止 1：使能
10	RSTIEN	R/W	使能 USBD 复位中断（USBD Reset Interrupt Enable） 0：禁止 1：使能
11	SUSIEN	R/W	使能挂起中断（Suspend Mode Interrupt Enable） 0：禁止 1：使能
12	WUPIEN	R/W	使能唤醒中断（Wakeup interrupt Enable） 0：禁止 1：使能
13	ERRIEN	R/W	使能出错中断（Error interrupt Enable） 0：禁止 1：使能
14	PMAOUIEN	R/W	使能分组缓冲区溢出中断（Packet Memory Area Over / Underrun Interrupt Enable） 0：禁止 1：使能
15	CTRIEN	R/W	使能正确传输（CTR）中断（Correct Transfer Interrupt Enable） 0：禁止 1：使能

26.4.1.2 USBD 中断状态寄存器（USBD_INTSTS）

偏移地址：0x44

复位值：0x0000

应用程序通过读取此寄存器的中断源状态信息确认产生中断请求事件。

当中断事件发生时，相应位会被硬件置位，如果 USBD_CTRL 中相应位也被置位，则会产生中断。中断执行后将清除对应的状态位。需要注意的是，当多个中断标志被置位时只产生一个中断。

USBD 模块有高优先级和低优先级两路中断请求源，前者的中断请求不能被屏蔽，但后者有可屏蔽中断请求。识别出中断请求由哪个端点后，调用相应中断服务程序。

下面详细描述每个位：

位/域	名称	R/W	描述
3:0	EPID[3:0]	R	端点 ID（Endpoint Identifier） 应用程序通过软件读取中断请求的处理顺序，处理顺序由端点号（端点号越小，优先级越高）决定，优先级由硬件写入。
4	DOT	R	传输方向（Direction of Transaction） 应用程序通过软件读取此位了解挂起中断传输方向，以此访问 USBD_EPn 中 CTFT 或 CTFR 位对应操作，此位由硬件根据传输方向写入。 其中 IN 分组指数据从 USBD 模块传输到 PC 主机，OUT 分组则相反。分组挂起表示数据传输完成。 0：相应端点的 CTFT 位被置位，表示一个 IN 分组挂起。 1：相应端点的 CTFR 位被置位，表示一个 OUT 分组挂起。如果 CTFT 位同时也被置位，就表示同时存在挂起的 OUT 分组和 IN 分组。
7:5	保留		
8	ESOFFLG	RC_W0	期望帧首标志（Expected Start of Frame Flag） USBD 模块未收到期望的 SOF 分组时由硬件置位。 0：清除 1：未收到期望的 SOF 分组
9	SOFFLG	RC_W0	帧首标志（Start Of Frame Flag） 表示一个新的 USBD 帧的开始，USBD 模块检测到总线上的 SOF 分组时由硬件置位。 0：清除 1：检测到 SOF 分组
10	RSTREQ	RC_W0	复位请求（USBD Reset Request） 应用程序可读可写，只在写 0 时有效，由硬件置位。 0：清除 1：检测到复位信号
11	SUSREQ	RC_W0	模块挂起请求（Suspend Mode Request） 表示一个来自 USBD 总线的挂起请求，在 USBD 线上超过 3ms 没有信号传输时由硬件置位。 0：清除 1：发生挂起中断。
12	WUPREQ	RC_W0	唤醒请求（Wakeup Request） 如果 USBD 模块处于挂起状态并检测到唤醒信号，由硬件置位。 0：清除 1：检测到唤醒信号
13	ERRFLG	RC_W0	传输错误中断（Failure Of Transfer Flag） 当发生 NANS（主机应答超时）/CRC/BST（位填充错误）/FVIO（帧格式错误）时，由硬件置位。 0：清除 1：发生传输错误

位/域	名称	R/W	描述
14	PMOFLG	RC_W0	分组缓冲区溢出标志（Packet Memory Overflow Flag） 当访问 USBD 分组缓冲区请求发出但 MCU 长时间没有响应时，由硬件置位。 0：清除 1：发生分组缓冲区溢出
15	CTFLG	R	正确传输标志（Correct Transfer Flag） 在端点正确完成一次数据传输后由硬件置位。 0：无意义 1：完成一次正确数据传输

26.4.1.3 USBD 帧状态寄存器（USBD_FRANUM）

偏移地址：0x48

复位值：0x0XXX，X 代表未定义数值

位/域	名称	R/W	描述
10:0	FRANUM[10:0]	R	帧序号（Frame Sequence Number） 记录了最新 SOF 分组的 11 位帧编码，序号随着主机发送帧的数量增加而自加。
12:11	LSOFNUM[1:0]	R	连续帧首丢失数目（Continuous Lost SOF Number） 记录了连续帧首丢失的 SOF 分组数目，当再次收到 SOF 分组时清除此位。
13	LOCK	R	锁定位（Lock） 如果 USBD 连续检测到 2 个及以上 SOF 分组，则由硬件置位，那么帧计数器将停止计数，直到 USBD 复位或总线挂起时恢复计数。
14	RXDMSTS	R	D-状态（Receive Data- line Status） 表示 USBD D-（Data minus）数据线的状态，可检测挂起状态下是否出现唤醒条件。
15	RXDPSTS	R	D+状态（Receive Data+ Line Status） 表示 USBD D+（Data plus）数据线的状态，可检测挂起状态下是否出现唤醒条件。

26.4.1.4 USBD 设备地址寄存器（USBD_ADDR）

偏移地址：0x4C

复位值：0x0000

位/域	名称	R/W	描述
6:0	ADDR[6:0]	R/W	设备地址（Device Address） 记录了 USBD 主机在枚举过程中为设备分配的地址值。
7	USBDEN	R/W	使能 USBD 模块（USBD Enable） 使能 USBD。 0：禁止 1：使能
15:8	保留		

26.4.1.5 USBD 分组缓冲区描述表地址寄存器（USBD_BUFFTB）

偏移地址：0x50

复位值：0x0000

位/域	名称	R/W	描述
2:0	保留		
15:3	BUFFTB	R/W	缓冲表（Buffer Table） 记录分组缓冲区描述表的起始地址。
31:16	保留		

26.4.1.6 USBD 切换寄存器（USBD_SWITCH）

偏移地址：0x100

复位值：0x0000_0000

位/域	名称	R/W	描述
0	SWITCH[0]	R/W	USBD 切换位（ USBD Switch） 默认为 0。 0：使用 USBD1 1：使用 USBD2，此时可以和 CAN 同时使用（引脚 PA11 和 PA12 给 USBD 使用）。
15:1	保留		

26.4.2 端点寄存器

每个端点都有与之对应的 USBD_EPn（n=0…7）寄存器，用于存储该端点的各种状态信息。

26.4.2.1 USBD 端点 n 寄存器（USBD_EPn）（n=0 to 7）

偏移地址：0x00 至 0x1C

复位值：0x0000

端点寄存器在收到 USBD 总线复位或 USBD_CTRL 的 FORRST 被置位时，除了 CTFT 和 CTFR 位保持不变用以数据传输外，其他位被复位。

下面详细描述每个位：

位/域	名称	R/W	描述
3:0	ADDR[3:0]	R/W	端点地址（Endpoint Address） 在使能一个端点前，应用程序必须将此 4 位设置为一个地址。
5:4	TXSTS	T	发送数据状态（Status Bits for Transmission Transfers） 此位用于标识端点的当前状态，发送状态编码表标出了所有的状态。应用程序可读可写，写 0 无效，写 1 翻转此位。
6	TXDTOG	T	发送数据翻转位（Data Toggle for Transmission Transfers） （1）对非同步端点，用于指示下一个要传输的数据分组的 Toggle 位： 0：DATA0 1：DATA1 （2）对于双缓冲端点，可用于分组缓冲区交换。 （3）对于同步端点，由于只传送 DATA0，因此该位只用于支持分组缓冲区交换。硬件在接收到数据分组后就设置该位。 应用程序可读可写，写 0 无效，写 1 翻转此位。
7	CTFT	RC_W0	发送正确标志位（Correct Transfer Flag for Transmission） 在一个正确的 IN 分组传输完成后，由硬件置位。 0：清除此位，读出无意义 1：IN 分组正确传输完成。写入无效

位/域	名称	R/W	描述
			应用程序可读可写，写 0 有效，写 1 无效。
8	KIND	R/W	端点特殊类型位（Endpoint Kind） 具体的定义请参考端点特殊类型定义表。
10:9	TYPE	R/W	端点类型位（Endpoint Type） 具体的定义请参考端点类型定义表。
11	SETUP	R	分组（Setup Transaction Completed）（只读） 表示当前传输完成的是否为 SETUP 分组。在 USBD 模块收到一个正确的 SETUP 分组后由硬件置位。 0：否 1：是 注：CTFR 为 0 时，此位才可以被修改，CTFR 为 1 时不能修改。
13:12	RXSTS	T	数据接收状态（Status Bits for Reception Transfers） 此位用于标识端点的当前状态，接收状态编码表标出了所有的状态。 应用程序可读可写，写 0 无效，写 1 翻转此位。
14	RXDTOG	T	用于数据接收的数据翻转位（Data Toggle for Reception Transfers） （1）对于非同步端点，标记希望接收的下一个数据分组的 Toggle 位，由硬件设置： 0：DATA0 1：DATA1 （2）对于控制端点，在收到 SETUP 分组后，由硬件清除。 （3）对于双缓冲端点，用于双缓冲区的交换。 （4）对于同步端点，仅用于双缓冲区的交换，而不需进行翻转。 应用程序可读可写，写 0 无效，写 1 翻转此位。
15	CTFR	RC_W0	正确接收标志位（Correct Transfer Flag for Reception） 在正确接收到 OUT 或 SETUP 分组时，由硬件置位。 0：清除 1：正确接受到分组 应用程序可读可写，写 0 有效，写 1 无效。

表格 132 接收状态编码

RXSTS	描述
00	DISABLED：端点忽略所有的接收请求。
01	STALL：端点以 STALL 分组响应所有的接收请求。
10	NAK：端点以 NAK 分组响应所有的接收请求。
11	VALID：端点可用于接收。

表格 133 端点类型编码

TYPE	描述
00	BULK：批量端点
01	CONTROL：控制端点
10	ISO：同步端点
11	INTERRUPT：中断端点

表格 134 端点特殊类型定义

TYPE 的值	KIND 意义
00（BULK）	DBL_BUF：双缓冲端点
01（CONTROL）	STATUS_OUT
其他	未使用

表格 135 发送状态编码

TXSTS	描述
00	DISABLED：端点忽略所有的发送请求。
01	STALL：端点以 STALL 分组响应所有的发送请求。
10	NAK：端点以 NAK 分组响应所有的发送请求。
11	VALID：端点可用于发送。

26.4.3 缓冲区寄存器

缓冲区描述表位于分组缓冲区内，用以配置 USBD 模块和微控制器内核共享的分组缓冲区的地址和大小。

26.4.3.1 发送缓冲区地址寄存器 n（USBD_TXADDRn）（n=0 to 7）

偏移地址：[USBD_BUFFTB]+n×16

USBD 本地地址：[USBD_BUFFTB]+n×8

位/域	名称	R/W	描述
0	保留，必须为 0。		
15:1	ADDR	R/W	发送缓冲区地址（Transmission Buffer Address） 记录收到下一个 IN 分组时，需要发送的数据所在的缓冲区起始地址。

26.4.3.2 发送数据字节数寄存器 n（USBD_TXCNTn）（n=0 to 7）

偏移地址：[USBD_BUFFTB]+n×16+4

USBD 本地地址：[USBD_BUFFTB]+n×8+2

位/域	名称	R/W	描述
9:0	CNT	R/W	发送数据字节数（Transmission Byte Count） 此位记录了收到下一个 IN 分组时要传输的数据字节数。
15:10	保留		

注：双缓冲区和同步 IN 端点有两个 USBD_TXCNT 寄存器：分别为 USBD_TXCNT_1 和 USBD_TXCNT_0，内容如下：

位/域	名称	R/W	描述
9:0	CNT_0	R/W	发送数据字节数（Transmission Byte Count0） 记录收到下一个 IN 分组时要传输的数据字节数。
15:10	保留		
25:16	CNT_1	R/W	发送数据字节数（Transmission Byte Count1） 记录收到下一个 IN 分组时要传输的数据字节数。
31:26	保留		

26.4.3.3 接收缓冲区地址寄存器 n（USBD_RXADDRn）（n=0 to 7）

偏移地址：[USBD_BUFFTB]+n×16+8

USBD 本地地址：[USBD_BUFFTB]+n×8+4

位/域	名称	R/W	描述
0	保留，必须为 0。		
15:1	ADDR	R/W	接收缓冲区地址（Reception Buffer Address） 记录收到下一个 OUT 或者 SETUP 分组时，用于保存数据的缓冲区起始地址。

26.4.3.4 接收数据字节数寄存器 n（USBD_RXCNTn）（n=0 to 7）

偏移地址：[USBD_BUFFTB]+n×16+12

USBD 本地地址：[USBD_BUFFTB]+n×8+6

用于存放接收分组时需要使用到的两个参数：接收分组缓冲区大小和 USBD 实际接收的字节数。

位/域	名称	R/W	描述
9:0	CNT	R	接收到的字节数（Reception Byte Count） 由 USBD 模块写入，用以记录端点收到的最新的 OUT 或 SETUP 分组的实际字节数。
14:10	BLKNUM	R/W	存储区块的数目（Number of Blocks） 记录分配的存储区块的数目，存储区块数目决定分组缓冲区大小。
15	BLKSIZE	R/W	存储区块的大小（Block Size） 决定存储区块的大小，它决定缓冲区大小。 0：2 字节 1：32 字节

注：双缓冲区和同步 IN 端点有两个 USBD_RXCNT 寄存器：分别为 USBD_RXCNT_1 和 USBD_RXCNT_0，内容如下：

位/域	名称	R/W	描述
9:0	CNT_0[9:0]	R	接收到的字节数（Reception Byte Count） 记录端点收到的最新的 OUT 或 SETUP 分组的实际字节数。
14:10	BLKNUM_0	R/W	存储区块的数目（Number of Blocks） 记录分配的存储区块的数目，它决定分组缓冲区的大小。
15	BLKSIZE_0	R/W	存储区块的大小（Block Size） 决定存储区块的大小。 0：2 字节 1：32 字节
25:16	CNT_1	R	接收到的字节数（Reception Byte Count） 记录端点收到的最新的 OUT 或 SETUP 分组的实际字节数。
30:26	BLKNUM_1	R/W	存储区块的数目（Number of Blocks） 记录分配的存储区块的数目。
31	BLKSIZE_1	R/W	存储区块的大小（Block Size） 决定存储区块的大小。 0：2 字节 1：32 字节

表格 136 分组缓冲区大小的定义

BLKNUM[4:0]的值	BLKSIZE=0 时的 分组缓冲区大小	当 BLKSIZE=1 时的分组缓冲区大小
00000	不允许使用	32 字节
00001	2 字节	64 字节
00010	4 字节	96 字节
00011	6 字节	128 字节
…	…	…
01111	30 字节	512 字节
10000	32 字节	保留
10001	34 字节	保留
10010	36 字节	保留
…	…	…
11110	60 字节	保留
11111	62 字节	保留

27 模数转换器（ADC）

27.1 术语全称、缩写描述

表格 137 ADC 术语全称、缩写描述

中文全称	英文全称	英文缩写
模拟看门狗	Analog watchdog	AWD
转换	Conversion	C
注入	Injected	INJ
规则	Regular	REG
开始	Start	S
扫描	Scan	SCAN
单个	Single	SINGLE
自动	Automatic	A
组	Group	G
间断	Discontinuous	DISC
计数	Count	CNT
双数	Dual	DUAL
连续	Continuous	C
校准	Calibration	CAL
复位	Reset	RST
对齐	Alignment	ALIGN
外部	External	EXT
事件	Event	E
触发	Trigger	TRG
温度	Temperature	T
传感器	Sensor	S
时间	Time	TIM
采样	Sample	SMP
偏移	Offset	OF
高	High	H
低	Low	L
阈值	Threshold	T
序列	Sequence	SEQ

中文全称	英文全称	英文缩写
长度	Length	LEN
-		
规则通道	Regular Channels	REG
注入通道	Injected Channel	INJ
注入组	Injected Group	INJG
自动	Automatic	A
转换	Conversion	C
模拟看门狗	Analog Watchdog	AWD
间断模式	Discontinuous Mode	DISC
扫描模式	Scan Mode	SCAN
-		
连续转换	Continuous Conversion	CONTC
单次转换	Single Conversion	SINGLEC
外部	External	EXT
外部触发	External Trigger	EXTTRG
采样时间	Sample Time	SMPTIM
序列	Sequence	SEQ
数目	Number	NUM

27.2 简介

系列产品有 3 个 ADC，精度为 12 位，每个 ADC 最多有 16 个外部通道和 2 个内部通道。其中 ADC1 和 ADC2 都有 16 个外部通道，ADC3 一般有 8 个外部通道，各通道 A/D 转换模式有单次、连续、扫描或间断，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。

27.3 主要特征

（1） ADC 供电要求：2.4V 到 3.6V，一般电源电压为 3.3V。

（2） ADC 输入范围：$V_{REF-} \leq V_{IN} \leq V_{REF+}$。

（3） 12 位分辨率

（4） ADC 转换时间

- 公式：TCONV=采样时间+12.5 个周期
- 采样时间由 SMPCYCCFGx[2:0]位控制，最小采样周期为 1.5 个，当 ADCCLK=14MHz，采样时间为 1.5 周期：TCONV=1.5 周期+12.5 周期=14 周期=1 μs。

（5） 模式输入通道类别

- 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

（6） 通道转换模式

- 单个通道转换模式：单次转换模式、连续转换模式
- 输入通道分类：规则通道、注入通道
- 一组通道转换模式：扫描模式、间断模式、注入通道管理
- ADC 模式：独立 ADC 模式、双重 ADC 模式

（7） 触发方式

- 片上定时器信号触发
- 外部引脚信号触发
- 软件触发

（8） 数据寄存器

- 规则数据寄存器
- 注入数据寄存器

（9） 中断

- 转换结束中断
- 模拟看门狗中断

（10） 支持规则数据转换的 DMA 请求

（11） 数据对齐

- 数据寄存器 ADC_CTRL2 的的 DALIGNCFG 位可配置数据对齐方式为左对齐或右对齐。

（12） 自校准

- 通过设置 ADC_CTRL2 寄存器的 CAL 位启动校准。在校准期间 CAL 位置 1，校准结束硬件清零，每次上电前都重新执行一次校准操作。

27.4 功能描述

27.4.1 ADC 引脚

表格 138 ADC 引脚

名称	说明	信号类型
V_{REF+}	ADC 使用的高端/正极参考电压，$2.4V \leqslant V_{REF+} \leqslant V_{DDA}$	输入，模拟参考正极
V_{DDA}(1)	等效于 V_{DD} 的模拟电源且：$2.4V \leqslant V_{DDA} \leqslant V_{DD}(3.6V)$	输入，模拟电源
V_{REF-}	ADC 使用的低端/负极参考电压，$V_{REF-} = V_{SSA}$	输入，模拟参考负极
V_{SSA}(1)	等效于 V_{SS} 的模拟电源地	输入，模拟电源地

名称	说明	信号类型
ADCx_IN[15:0]	16 个模拟输入通道	模拟输入信号

注：1.V_{DDA} 和 V_{SSA} 应该分别连接到 V_{DD} 和 V_{SS}。

27.4.2 ADC 转换模式

产品内置多个 ADC、多个通道（具体数量参考数据手册），可以组合出多种转换模式。

内置多个 ADC，依据 ADC 的数量，转换模式可以分为独立 ADC 模式、双重 ADC 模式；内置多个通道，可以将通道 2 类组别，即规则通道、注入通道，在每组内部转换模式分为扫描模式、间断模式；对于每个组内部的通道，转换模式分为单次转换模式、连续转换模式。

在应用中，依据实际应用需求，可结合 ADC 的数量、转换的通道数、每个通道的转换方式设计出满足需求的 ADC 转换方式。

27.4.2.1 单个 ADC 工作、单个通道的转换方式

单 ADC 单通道

单 ADC 单通道不使用外部触发软件开启，转换模式为单次和连续同时禁止扫描，数据转换的结果为右对齐，单 ADC 转换完成后触发中断，在中断服务函数中读取数据，不使用 DMA 传输。

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。

该模式通过配置寄存器 ADC_CTRL2 的 ADCEN 位启动或外部触发启动。

规则通道一次转换结束，转换数据将存储到 16 位 ADC_REGDATA 寄存器，EOCFLG 位置 1，若配置 EOCIEN 位置 1，则产生中断。

注入通道一次转换结束，转换数据将存储到 16 位 ADC_INJDATA1 寄存器，INJEOCFLG 位置 1，若配置 INJEOCIEN 位置 1，则产生中断。

图 127 单次转换模式时序图

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。

该模式通过配置寄存器 ADC_CTRL2 的 ADCEN 位启动或外部触发启动。

一个规则通道转换结束，转换数据将存储到 16 位 ADC_REGDATA 寄存器，EOCFLG 位置 1，若配置 EOCIEN 位置 1，则产生中断。

一个注入通道转换结束，转换数据将存储到 16 位 ADC_INJDATA1 寄存器，INJEOCFLG 位置 1，若配置 INJEOCIEN 位置 1，则产生中断。

图 128 连续转换模式时序图

27.4.2.2 单个 ADC、一组通道转换方式

单 ADC 多通道

单 ADC 多通道下开启扫描模式，不使用外部触发转换，软件触发，数据转换结果右对齐，ADC 转换结果数据使用 DMA 传输到存储器。

模拟输入通道分类

规则通道组

- 规则组由 16 个通道组成
- 规则通道转换顺序通过配置寄存器 ADC_REGSEQx 决定
- 规则组转换通道的总数通过配置寄存器 ADC_REGSEQ1 的 REGSEQLEN[3:0]位决定

注入通道组

- 注入组由 4 个通道组成
- 注入通道转换顺序通过配置寄存器在 ADC_INJSEQ 决定
- 注入组转换通道的总数通过配置寄存器在 ADC_INJSEQ 的 INJSEQLEN[1:0]位决定

内部输入通道

温度传感器：

- 温度传感器用来测量芯片内部的温度
- 温度传感器选择 ADC1_IN16 输入通道

- 通过配置寄存器 ADC_CTRL2 的 TSVREFEN 位启动

内部参考电压 V_{REFINT}：

- 内部参考电压用来提供一个稳定的电压输出给 ADC
- 内部参考电压 V_{REFINT} 选择 ADC1_IN17 输入通道

通道转换顺序

规则序列寄存器配置：

- 配置寄存器 ADC_REGSEQ3 的 REGSEQC1[4:0]~REGSEQC6[4:0]位设置第 1~6 个转换通道
- 配置寄存器 ADC_REGSEQ2 的 REGSEQC7[4:0]~REGSEQC12[4:0]位设置第 7~12 个转换通道
- 配置寄存器 ADC_REGSEQ1 的 REGSEQC13[4:0]~REGSEQC16[4:0]位设置第 13~16 个转换通道
- 配置寄存器 ADC_REGSEQ1 的 REGSEQLEN[3:0]设置需要转换的通道数目

注入序列寄存器配置：

- 配置寄存器 ADC_INJSEQ 的 INJSEQC1[4:0]~INJSEQC4[4:0]位设置第 1~4 个转换通道
- 配置寄存器 ADC_INJSEQ 的 INJSEQLEN[1:0]设置需要转换的通道数目
- 若 INJSEQLEN 的值小于 4，则转换顺序不同，转换顺序从（4-INJSEQLEN）开始。

通道转换模式

扫描模式

该模式适用于一组通道，相当于在一组通道的每个通道上进行单次转换。

该模式通过配置寄存器 ADC_CTRL1 的 SCANEN 位启动，启动后 ADC 扫描所有的按序列寄存器 ADC_REGSEQ 或 ADC_INJSEQ 排好顺序的通道，每个通道转换完成后将自动转换该组下一个通道。

若配置 CONTCEN 位置 1，当该组最后一个通道转换完成后将继续从该组第一个通道开始转换。

若配置 DMAEN 位置 1，每次通道转换完成后，DMA 控制器将规则通道的转换数据传输到 SRAM。

图 129 扫描模式时序图

间断模式

该模式适用于一组通道，相当于对一组通道里的多个通道进行连续转换。

对于规则组，该模式通过配置寄存器 ADC_CTRL1 的 REGDISCEN 位启动，启动后进行一次 n 个通道的短序列转换（n<=8），n 通过配置寄存器 ADC_CTRL1 的 DISCNUMCFG[2:0]决定；通过软件控制或外部触发源可以启动下一轮 n 个通道的转换，直到该组通道全部转换完成，EOCFLG 位置 1。

对于注入组，该模式通过配置寄存器 ADC_CTRL1 的 INJDISCEN 位启动，启动后按序列寄存器配置顺序进行一个通道的转换，通过软件控制或外部触发源可以按顺序启动下一个通道的转换，直到该组通道全部转换完成，EOCFLG 位和 INJEOCFLG 位置 1。

图 130 间断模式时序图

注入通道管理

触发注入：通过清除寄存器 ADC_CTRL1 的 INJGACEN 位并配置 SCANEN 位启动，在规则组通道转换期间若产生一个软件触发或外部触发将启动触发注入转换，此时规则通道转换停止，注入通道序列开始转换，在注入组通道转换完成后恢复规则组通道转换。

图 131 触发注入时序图

自动注入：需要配置寄存器 ADC_CTRL1 的 INJGACEN 位启动，在规则组通道转换完成后，注入组通道自动开始转换，自动注入模式下，注入组通道的外部触发必须禁止；若还配置了寄存器 ADC_CTRL2 的 CONTCEN 位，规则组和注入组的所有通道按序列进行连续转换。

图 132 自动注入时序图

27.4.2.3 双重 ADC 模式、一组通道转换模式

两个或两个以上 ADC 模块的产品，使用双重 ADC 模式，ADC1 默认为主 ADC，其他默认为从 ADC，通过配置 ADC1_CTRL1 寄存器中 DUALMCFG[2:0] 位设置双 ADC 模式。

当配置由外部事件触发时，需要设置为仅触发主 ADC，然后配置从 ADC 为软件触发。（主和从的外部触发必须同时发生）

双重 ADC 有以下 8 种可能的模式：

（1） 同时规则模式

同时规则模式即 ADC1 和 ADC2 同时转换一个规则通道组。两个 ADC 不能同时转换一个通道。

外部触发事件由寄存器 ADC1_CTRL2 的 REGEXTTRGSEL[2:0]决定。

ADC 转换结束产生 DMA 传输请求，ADC1 转换数据放在寄存器 ADC1_REGDATA 的低 16 位，ADC2 转换数据放在寄存器 ADC1_REGDATA 的高 16 位。

所有 ADC 规则通道转换完后，产生 EOCFLG 中断。

（2） **同时注入模式**

同时注入模式即 ADC1 和 ADC2 同时转换一个注入通道组。两个 ADC 不能同时转换一个通道。

外部触发事件由寄存器 ADC1_CTRL2 的 INJGEXTTRGSEL[2:0]决定。

ADC 转换结束，转换数据存储在寄存器 ADC_INJDATAx。

所有 ADC 注入通道转换完成后，产生 INJEOCFLG 中断。

（3） **快速交叉模式**

快速交叉模式即 ADC1 和 ADC2 交替采集一个规则通道组，间隔时间短。

外部触发事件由寄存器 ADC_CTRL2 的 REGEXTTRGSEL[2:0]决定，触发产生后 ADC2 启动，ADC1 延时 7 个 ADC 时钟周期后启动。

采样时间要小于 7 个 ADC 时钟周期。

（4） **慢速交叉模式**

快速慢速交叉模式即 ADC1 和 ADC2 交替采集一个规则通道组，间隔时间长。

外部触发事件由寄存器 ADC_CTRL2 的 REGEXTTRGSEL[2:0]决定，触发产生后 ADC2 启动，ADC1 延时 14 个 ADC 时钟周期后启动。

采样时间要小于 14 个 ADC 时钟周期。

（5） **交替触发模式**

交替触发模式即 ADC1 和 ADC2 轮流采集注入通道组。

外部触发事件由寄存器 ADC1_CTRL2 的 INJGEXTTRGSEL[2:0]决定，触发产生后 ADC1 开始转换直到所有通道都转换完成 ADC2 才开始转换。

若 ADC1 和 ADC2 使能间断模式，触发产生后 ADC1 开始转换第一个注入通道，第二个触发产生后 ADC2 开始转换第一个注入通道，以此类推。

（6） **混合的同时规则/注入模式**

混合同时规则/注入模式即同时规则模式被中断后启动同时注入模式。

该模式中，必须转换相同长度的序列或设置触发间隔时间能够转换完成较长序列。

（7） **混合的同时规则+交替触发模式**

混合的同时规则+交替触发模式即同时规则模式被中断后启动交替触发模式。

该模式中，必须转换相同长度的序列或设置触发间隔时间能够转换完成较长序列。

（8） **混合同时注入+交叉模式**

混合同时注入+交叉模式即交叉模式被中断后启动同时注入模式模式。

27.4.3 外部触发

外部触发的寄存器配置如下：

- 规则组通道的外部事件触发通过配置寄存器 ADC_CTRL2 的 REGEXTTRGSEL[2:0]位启动
- 注入组通道的外部事件触发通过配置寄存器 ADC_CTRL2 的 INJGEXTTRGSEL[2:0]位启动

表格 139 ADC1 和 ADC2 的规则通道外部触发

触发源	REGEXTTRGSEL[2:0]	触发类型
TMR1_CC1	000	来自片上定时器的内部信号
TMR1_CC2	001	
TMR1_CC3	010	
TMR2_CC2	011	
TMR3_TRGO	100	
TMR4_CC4	101	
EINT 线 11/TMR8_TRGO	110	外部引脚/来自片上定时器的内部信号
REGSWSC	111	软件控制位

表格 140 ADC1 和 ADC2 的注入通道外部触发

触发源	INJGEXTTRGSEL[2:0]	触发类型
TMR1_TRGO	000	来自片上定时器的内部信号
TMR1_CC4	001	
TMR2_TRGO	010	
TMR2_CC1	011	
TMR3_CC4	100	
TMR4_TRGO	101	
EINT 线 15/TMR8_CC4	110	外部引脚/来自片上定时器的内部信号
INJSWSC	111	软件控制位

表格 141 ADC3 的规则通道外部触发

触发源	REGEXTTRGSEL[2:0]	触发类型
TMR3_CC1	000	来自片上定时器的内部信号
TMR2_CC3	001	
TMR1_CC3	010	
TMR8_CC1	011	
TMR8_TRGO	100	
TMR5_CC1	101	
TMR5_CC3	110	
REGSWSC	111	软件控制位

表格 142 ADC3 的注入通道外部触发

触发源	REGEXTTRGSEL[2:0]	触发类型
TMR1_TRGO	000	来自片上定时器的内部信号
TMR1_CC4	001	
TMR4_CC3	010	
TMR8_CC2	011	
TMR8_CC4	100	
TMR5_TRGO	101	
TMR5_CC4	110	
INJSWSC	111	软件控制位

27.4.4 数据寄存器

27.4.4.1 规则数据寄存器

ADC_REGDATA 是一个 32 位的 ADC 规则数据寄存器，单 ADC 模式下只使用低 16 位存储转换数据，双 ADC 模式下低 16 位存储 ADC1 的转换数据，高 16 位存储 ADC2 的转换数据，数据对齐方式为左对齐和右对齐。

通过配置寄存器 ADC_CTRL2 的 DALIGNCFG 位决定是否使用 DMA 传输，其中规则通道最多有 16 个，但规则数据寄存器只有 1 个，所以在多通道转换时将产生数据覆盖问题，此时就需要使用 DMA 传输。

27.4.4.2 注入数据存储器

ADC_INJDATAx（x=1,2,3,4）是 ADC 注入数据寄存器，有四个 32 位的寄存器，其低 16 位有效，高 16 位保留。注入通道最多有 4 个，注入数据寄存器也有四个，在多通道转换时不会发生数据覆盖问题，数据对齐方式为左对齐和右对齐。

27.4.5 中断

27.4.5.1 转换结束中断

规则组通道转换结束中断

规则通道转换结束后产生中断，在中断函数中读取规则数据寄存器的值。

通过配置寄存器 ADC_STS 的 EOCFLG 位决定。

注入组通道转换结束中断

注入通道转换结束后产生中断，在中断函数中读取规则数据寄存器的值。

通过配置寄存器 ADC_STS 的 INJEOCFLG 位决定。

27.4.5.2 模拟看门狗中断

若输入的模拟电压不在阈值范围内将产生模拟看门狗中断。

通过配置存器 ADC_STS 的 AWDFLG 位决定。

27.4.6 DMA

规则通道转换结束后产生 DMA 请求，可以将转换的数据结果从 ADC_REGDATA 寄存器传输到存储器中。

只有 ADC1 和 ADC3 可以产生 DMA 请求，ADC2 的转换结果通过 ADC1 的 DMA 功能传输。

27.5 寄存器地址映射

表格 143 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_CTRL1	ADC 控制寄存器 1	0x04
ADC_CTRL2	ADC 控制寄存器 2	0x08
ADC_SMPTIM1	ADC 采样时间寄存器 1	0x0C
ADC_SMPTIM2	ADC 采样时间寄存器 2	0x10
ADC_INJDOFx	ADC 注入通道数据偏移寄存器 x	0x14-0x20
ADC_AWDHT	模拟看门狗高阀值寄存器	0x24
ADC_AWDLT	模拟看门狗低阀值寄存器	0x28
ADC_REGSEQ1	ADC 规则序列寄存器 1	0x2C
ADC_REGSEQ2	ADC 规则序列寄存器 2	0x30
ADC_REGSEQ3	ADC 规则序列寄存器 3	0x34

寄存器名	描述	偏移地址
ADC_INJSEQ	ADC 注入序列寄存器	0x38
ADC_INJDATAx	ADC 注入数据寄存器 x	0x3C–0x48
ADC_REGDATA	ADC 规则数据寄存器	0x4C

27.6 寄存器功能描述

27.6.1 ADC 状态寄存器（ADC_STS）

偏移地址：0x00

复位值：0x0000 000

位/域	名称	R/W	描述
0	AWDFLG	RC_W0	发生模拟看门狗事件标志位（Analog Watchdog Occur 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
1	EOCFLG	RC_W0	转换结束标志（End Of Conversion Flag） 0：未完成 1：完成
2	INJEOCFLG	RC_W0	注入通道转换结束标志（Injected Channel End Of Conversion Flag） 0：未完成 1：完成
3	INJCSFLG	RC_W0	注入通道转换开始标志（Injected Channel Conversion Start Flag） 0：未开始 1：开始
4	REGCSFLG	RC_W0	规则通道转换开始标志（Regular Channel Conversion Start Flag） 0：未开始 1：开始
31:5	保留		

27.6.2 ADC 控制寄存器 1（ADC_CTRL1）

偏移地址：0x04

复位值：0x0000 0000

位/域	名称	R/W	描述
4:0	AWDCHSEL	R/W	选择模拟看门狗通道（Analog Watchdog Channel Select） 00000：ADC 模拟输入通道 0 00001：ADC 模拟输入通道 1 …… 01111：ADC 模拟输入通道 15 10000：ADC 模拟输入通道 16 10001：ADC 模拟输入通道 17 其它值：保留 关于该寄存器，注意如下：

位/域	名称	R/W	描述
			（1）ADC1 的模拟输入通道 16 和通道 17 在芯片内部分别连到了温度传感器和 V_{REFINT} （2）ADC2 的模拟输入通道 16 和通道 17 在芯片内部连到了 V_{SS}； （3）ADC3 模拟输入通道 9、14、15、16、17 与 V_{ss} 相连。
5	EOCIEN	R/W	使能 EOC 中断（EOC Interrupt Enable） 用于使能转换结束后产生中断。 0：禁止 1：使能
6	AWDIEN	R/W	使能模拟看门狗中断（Analog Watchdog Interrupt Enable） 若该位被设置且处于扫描模式下，当看门狗检测到数值超过阈值时，产生中断并中止扫描。 0：禁止 1：使能
7	INJEOCIEN	R/W	使能注入通道转换结束中断（Interrupt Enable For Injected Channels End Of Conversion Flag） 0：禁止 1：使能
8	SCANEN	R/W	使能扫描模式（Scan Mode Enable） 在扫描模式中，转换由 ADC_REGSEQX 或 ADC_INJSEQX 寄存器选中的通道。 0：禁止 1：使能 注：如果分别设置了 EOCINTEN 或 INJEOCINTEN 位，只在最后一个通道转换完毕后才会产生 EOC 或 INJEOC 中断。
9	AWDSGLEN	R/W	扫描模式下，在单一通道上使能看门狗（Enable The Watchdog On A Single Channel In Scan Mode） 该通道由 AWDCHSEL[4:0]位指定。 0：在所有的通道上使能 1：在单一通道上使能
10	INJGACEN	R/W	使能注入通道组自动转换（Automatic Injected Group Conversion Enable） 用于使能规则通道组转换结束后注入通道组自动转换。 0：禁止 1：使能
11	REGDISCEN	R/W	使能规则通道上的间断模式（Discontinuous Mode On Regular Channels Enable） 0：禁止 1：使能
12	INJDISCEN	R/W	使能注入通道上的间断模式（Discontinuous Mode On Injected Channels Enable） 0：禁止 1：使能
15:13	DISCNUMCFG	R/W	配置间断模式通道数据（Discontinuous Mode Channel Number Configure） 000：1 个通道 001：2 个通道 ……

位/域	名称	R/W	描述
			111：8 个通道
19:16	DUALMCFG	R/W	配置双 ADC 模式（Dual ADC Mode Configure） 0000：独立模式 0001：混合的同步规则+注入同步模式 0010：混合的同步规则+交替触发模式 0011：混合同步注入+快速交叉模式 0100：混合同步注入+慢速交叉模式 0101：注入同步模式 0110：规则同步模式 0111：快速交叉模式 1000：慢速交叉模式 1001：交替触发模式 其它：保留 在 ADC2 和 ADC3 中这些位为保留位；在双 ADC 模式中，改变通道的配置会产生一个重新开始的条件，这将导致同步丢失，建议在进行任何配置改变前关闭双 ADC 模式（即配置为独立模式）。
21:20	保留		
22	INJAWDEN	R/W	使能注入通道的模拟看门狗功能（Enable the Analog Watchdog Function On the Injected Channels） 0：禁止 1：使能
23	REGAWDEN	R/W	使能规则通道上的模拟看门狗功能（Enable the Analog Watchdog Function On the Regular Channels） 0：禁止 1：使能
31:24	保留		

27.6.3 ADC 控制寄存器 2（ADC_CTRL2）

偏移地址：0x08

复位值：0x0000 0000

位/域	名称	R/W	描述
0	ADCEN	R/W	使能 ADC（ADC Enable） （1）该位为 0 时，写为 1 将给 ADC 上电，并启 ADC 动转换 （2）该位为 1 时，写 1 将启动转换 0：禁止进行 ADC 转换和校准，进入掉电模式 1：启用 ADC 并且开始转换 注：为防止触发错误的转换，若此寄存器中有其他位与该位都被改变，则转换不会被触发。
1	CONTCEN	R/W	使能连续转换模式（Continuous Conversion Mode Enable） 0：单次转换模式 1：连续转换模式
2	CAL	R/W	A/D 校准（A/D Calibration） 该位由软件置 1 开始校准，校准完成时硬件清 0。 0：校准完成 1：开始校准

位/域	名称	R/W	描述
3	CALRST	R/W	复位校准（Calibration Reset） 该位有软件置 1，校准寄存器复位完成后由硬件清 0。 0：表明复位校准寄存器已完成 1：复位校准寄存器
7:4	保留		
8	DMAEN	R/W	使能 DMA 模式（DMA Mode Enable） 0：禁止 1：使能 注：只有 ADC1 和 ADC3 能产生 DMA 请求。
10:9	保留		
11	DALIGNCFG	R/W	配置数据对齐方式（Data Alignment Mode Configure） 0：右对齐 1：左对齐
14:12	INJGEXTTRGSEL	R/W	选择启动注入组转换的外部事件（Select the External Trigger Event to Start the Injected Group Conversion） ADC1 和 ADC2 的触发配置如下： 000：定时器 1 的 TRGO 事件 001：定时器 1 的 CC4 事件 010：定时器 2 的 TRGO 事件 011：定时器 2 的 CC1 事件 100：定时器 3 的 CC4 事件 101：定时器 4 的 TRGO 事件 110：EINT 线 15 111：INJSWSC ADC3 的触发配置如下： 000：定时器 1 的 TRGO 事件 001：定时器 1 的 CC4 事件 010：定时器 4 的 CC3 事件 011：定时器 8 的 CC2 事件 100：定时器 8 的 CC4 事件 101：定时器 5 的 TRGO 事件 110：定时器 5 的 CC4 事件 111：INJSWSC
15	INJEXTTRGEN	R/W	使能注入通道的外部触发转换模式（Enable the External Trigger Conversion Mode of the Injected Channels） 0：禁止 1：使能
16	保留		
19:17	REGEXTTRGSEL	R/W	选择启动规则组转换的外部事件（Select the External Trigger Event to Start the Regular Group Conversion） ADC1 和 ADC2 的触发配置如下： 000：定时器 1 的 CC1 事件 001：定时器 1 的 CC2 事件 010：定时器 1 的 CC3 事件 011：定时器 2 的 CC2 事件 100：定时器 3 的 TRGO 事件 101：定时器 4 的 CC4 事件 110：EINT 线 11 111：REGSWSC

位/域	名称	R/W	描述
			ADC3 的触发配置如下： 000：定时器 3 的 CC1 事件 001：定时器 2 的 CC3 事件 010：定时器 1 的 CC3 事件 011：定时器 8 的 CC1 事件 100：定时器 8 的 TRGO 事件 101：定时器 5 的 CC1 事件 110：定时器 5 的 CC3 事件 111：REGSWSC
20	REGEXTTRGEN	R/W	使能规则通道的外部触发转换模式（Enable the External Trigger Conversion Mode of the Regular Channels） 0：禁止 1：使能
21	INJSWSC	R/W	软件启动转换注入通道（Software Start Conversion Injected Channels） 如果在 INJEXESEL[2:0]位中选择 INJSWSC 作为触发事件，该位用于启动一组注入通道的转换；该位软件可置 1、清 0；启动转换后硬件清 0。 0：复位状态 1：启动转换注入通道
22	REGSWSC	R/W	软件启动转换规则通道（Software Start Conversion Regular Channels） 如果在 REGEXTSEL[2:0]位中选择了 REGSWSC 为触发事件，该位用于启动一组规则通道的转换；该位软件可置 1、清 0；启动转换后硬件清 0。 0：复位状态 1：开始转换规则通道
23	TSVREFEN	R/W	使能温度传感器和 V_{REFINT} 通道（Temperature Sensor And V_{REFINT} Channel Enable） 该位仅在 ADC1 中有效。 该位由软件置 1 和清 0，在多个 ADC 的器件中，该位仅出现在 ADC1 中。 0：禁止 1：使能
31:24	保留		

27.6.4 ADC 采样时间寄存器 1（ADC_SMPTIM1）

偏移地址：0x0C

复位值：0x0000 0000

位/域	名称	R/W	描述
2:0	SMPCYCCFG10	R/W	配置通道 10 的采样周期（Channel 10 Sample Cycles Configure） 000：1.5 周期 001：7.5 周期 010：13.5 周期 011：28.5 周期 100：41.5 周期 101：55.5 周期 110：71.5 周期

位/域	名称	R/W	描述
			111：239.5 周期
5:3	SMPCYCCFG11	R/W	配置通道 11 的采样周期（Channel 11 Sample Cycles Configure） 参考 SMPCYCCFG10 的描述。
8:6	SMPCYCCFG12	R/W	配置通道 12 的采样周期（Channel 12 Sample Cycles Configure） 参考 SMPCYCCFG10 的描述。
11:9	SMPCYCCFG13	R/W	配置通道 13 的采样周期（Channel 13 Sample Cycles Configure） 参考 SMPCYCCFG10 的描述。
14:12	SMPCYCCFG14	R/W	配置通道 14 的采样周期（Channel 14 Sample Cycles Configure） 参考 SMPCYCCFG10 的描述。
17:15	SMPCYCCFG15	R/W	配置通道 15 的采样周期（Channel 15 Sample Cycles Configure） 参考 SMPCYCCFG10 的描述。
20:18	SMPCYCCFG16	R/W	配置通道 16 的采样周期（Channel 16 Sample Cycles Configure） 参考 SMPCYCCFG10 的描述。
23:21	SMPCYCCFG17	R/W	配置通道 17 的采样周期（Channel 17 Sample Cycles Configure） 参考 SMPCYCCFG10 的描述。
31:24	保留		

27.6.5 ADC 采样时间寄存器 2（ADC_SMPTIM2）

偏移地址：0x10

复位值：0x0000 0000

位/域	名称	R/W	描述
2:0	SMPCYCCFG0	R/W	配置通道 0 的采样周期（Channel 0 Sample Cycles Configure） 参考 SMPCYCCFG10 的描述。
5:3	SMPCYCCFG1	R/W	配置通道 1 的采样周期（Channel 1 Sample Cycles Configure） 参考 SMPCYCCFG10 的描述。
8:6	SMPCYCCFG2	R/W	配置通道 2 的采样周期（Channel 2 Sample Cycles Configure） 参考 SMPCYCCFG10 的描述。
11:9	SMPCYCCFG3	R/W	配置通道 3 的采样周期（Channel 3 Sample Cycles Configure） 参考 SMPCYCCFG10 的描述。
14:12	SMPCYCCFG4	R/W	配置通道 4 的采样周期（Channel 4 Sample Cycles Configure） 参考 SMPCYCCFG10 的描述。
17:15	SMPCYCCFG5	R/W	配置通道 5 的采样周期（Channel 5 Sample Cycles Configure） 参考 SMPCYCCFG10 的描述。
20:18	SMPCYCCFG6	R/W	配置通道 6 的采样周期（Channel 6 Sample Cycles Configure） 参考 SMPCYCCFG10 的描述。
23:21	SMPCYCCFG7	R/W	配置通道 7 的采样周期（Channel 7 Sample Cycles Configure） 参考 SMPCYCCFG10 的描述。
26:24	SMPCYCCFG8	R/W	配置通道 8 的采样周期（Channel 8 Sample Cycles Configure） 参考 SMPCYCCFG10 的描述。
29:27	SMPCYCCFG9	R/W	配置通道 9 的采样周期（Channel 9 Sample Cycles Configure） 参考 SMPCYCCFG10 的描述。
31:30	保留		

27.6.6 ADC 注入通道数据偏移寄存器 x（ADC_INJDOFx）（x=1..4）

偏移地址：0x14-0x20

复位值：0x0000 0000

位/域	名称	R/W	描述
11:0	INJDOFx	R/W	注入通道 x 的数据偏移（Data Offset For Injected Channel x） 当转换注入通道时，这些位定义了用于从原始转换数据中减去的数值，转换的结果可以在 ADC_INJDATAx 寄存器中读出。
31:12	保留		

27.6.7 模拟看门狗高阀值寄存器（ADC_AWDHT）

偏移地址：0x24

复位值：0x0000 0FFF

位/域	名称	R/W	描述
11:0	AWDHT[11:0]	R/W	模拟看门狗高阀值（Analog Watchdog High Threshold）
31:12	保留		

27.6.8 模拟看门狗低阀值寄存器（ADC_AWDLT）

偏移地址：0x28

复位值：0x0000 0000

位/域	名称	R/W	描述
11:10	AWDLT[11:0]	R/W	模拟看门狗低阀值（Analog Watchdog Low Threshold）
31:12	保留		

27.6.9 ADC 规则序列寄存器 1（ADC_REGSEQ1）

偏移地址：0x2C

复位值：0x0000 0000

位/域	名称	R/W	描述
4:0	REGSEQC13	R/W	规则序列中的第 13 个转换（13th Conversion In Regular Sequence） 定义规则序列中第 13 个转换的通道编号（0~17）
9:5	REGSEQC14	R/W	规则序列中的第 14 个转换（14th Conversion In Regular Sequence） 参考 REGSEQC13 的描述
14:10	REGSEQC15	R/W	规则序列中的第 15 个转换（15th Conversion In Regular Sequence） 参考 REGSEQC13 的描述
19:15	REGSEQC16	R/W	规则序列中的第 16 个转换（16th Conversion In Regular Sequence） 参考 REGSEQC13 的描述
23:20	REGSEQLEN	R/W	规则通道序列长度（Regular Channel Sequence Length） 这些位由软件定义在规则通道转换序列中的通道数目。 0000：1 个转换 0001：2 个转换 …… 1111：16 个转换
31:24	保留		

27.6.10 ADC 规则序列寄存器 2（ADC_REGSEQ2）

偏移地址：0x30

复位值：0x0000 000

位/域	名称	R/W	描述
4:0	REGSEQC7	R/W	规则序列中的第 7 个转换（7th Conversion In Regular Sequence） 参考 REGSEQC13 的描述
9:5	REGSEQC8	R/W	规则序列中的第 8 个转换（8th Conversion In Regular Sequence） 参考 REGSEQC13 的描述
14:10	REGSEQC9	R/W	规则序列中的第 9 个转换（9th Conversion In Regular Sequence） 参考 REGSEQC13 的描述
19:15	REGSEQC10	R/W	规则序列中的第 10 个转换（10th Conversion In Regular Sequence） 参考 REGSEQC13 的描述
24:20	REGSEQC11	R/W	规则序列中的第 11 个转换（11th Conversion In Regular Sequence） 参考 REGSEQC13 的描述
29:25	REGSEQC12	R/W	规则序列中的第 12 个转换（12th Conversion In Regular Sequence） 参考 REGSEQC13 的描述
31:30	保留		

27.6.11 ADC 规则序列寄存器 3（ADC_REGSEQ3）

偏移地址：0x34

复位值：0x0000 0000

位/域	名称	R/W	描述
4:0	REGSEQC1	R/W	规则序列中的第 1 个转换（1st Conversion In Regular Sequence） 参考 REGSEQC13 的描述
9:5	REGSEQC2	R/W	规则序列中的第 2 个转换（2nd Conversion In Regular Sequence） 参考 REGSEQC13 的描述
14:10	REGSEQC3	R/W	规则序列中的第 3 个转换（3rd Conversion In Regular Sequence） 参考 REGSEQC13 的描述
19:15	REGSEQC4	R/W	规则序列中的第 4 个转换（4th Conversion In Regular Sequence） 参考 REGSEQC13 的描述
24:20	REGSEQC5	R/W	规则序列中的第 5 个转换（5th Conversion In Regular Sequence） 参考 REGSEQC13 的描述
29:25	REGSEQC6	R/W	规则序列中的第 6 个转换（6th Conversion In Regular Sequence） 参考 REGSEQC13 的描述
31:30	保留		

27.6.12 ADC 注入序列寄存器（ADC_INJSEQ）

偏移地址：0x38

复位值：0x0000 0000

位/域	名称	R/W	描述
4:0	INJSEQC1	R/W	注入序列中的第 1 个转换（1st Conversion In Injected Sequence） 定义注入序列中第 1 个转换的通道编号（0~17）
9:5	INJSEQC2	R/W	注入序列中的第 2 个转换（2nd Conversion In Injected Sequence）

位/域	名称	R/W	描述
14:10	INJSEQC3	R/W	注入序列中的第 3 个转换（3rd Conversion In Injected Sequence）
19:15	INJSEQC4	R/W	注入序列中的第 4 个转换（4th Conversion In Injected Sequence）
21:20	INJSEQLEN	R/W	注入通道序列长度（Injected Channel Sequence Length） 这些位由软件定义在注入通道转换序列中的通道数目，其转换序列是： INJSEQC(4-INJSEQLEN) →INJSEQ (5-INJSEQLEN) →INJSEQC(6-INJSEQLEN) →INJSEQC(7-INJSEQLEN)，具体情况如下： 00：1 个转换，只转换 INJSEQC4 01：2 个转换，转换顺序是 INJSEQC3→INJSEQC4 10：3 个转换，转换序列是 INJSEQC2→INJSEQC3→INJSEQC4 11：4 个转换，转换序列是 INJSEQC1→INJSEQC2→INJSEQC3→INJSEQC4
31:22	保留		

27.6.13 ADC 注入数据寄存器 x（ADC_INJDATAx）（x= 1..4）

偏移地址：0x3C–0x48

复位值：0x0000 0000

位/域	名称	R/W	描述
15:0	INJDATA	R	注入转换的数据（Injected Conversion Data） 注入通道的转换结果，只读。
31:16	保留		

27.6.14 ADC 规则数据寄存器（ADC_REGDATA）

偏移地址：0x4C

复位值：0x0000 0000

位/域	名称	R/W	描述
15:0	REGDATA	R	规则转换的数据（Regular Conversion Data） 规则通道的转换结果，只读。
31:16	ADC2DATA	R	ADC2 转换的数据（ADC2 Conversion Data） （1）在 ADC1 中该位有效，表示双 ADC 模式下，ADC2 规则通道转换的结果； （2）ADC2 和 ADC3 中，该位保留。

28 数模转换器（DAC）

28.1 术语全称、缩写描述

表格 144 DAC 术语全称、缩写描述

中文全称	英文全称	英文缩写
线性反馈移位寄存器	Linear Feedback Shift Register	LFSR

28.2 简介

DAC 是一个可配置为输入 8 位或 12 位数据，输出电压的数字/模拟转换器，在输入 12 位数据时，可以将数据设置为左对齐或右对齐。具有两路 DAC 输出通道，两路通道互不影响；每路通道有多个触发源可触发转换，可单路触发转换输出，也可以两路同时触发转换输出。两路通道都可独立生成噪声波形和三角波形。

28.3 结构框图

图 133 DAC 结构框图

28.4 功能描述

28.4.1 DAC 转换及触发源

DAC 的输出是通过对 DAC_DATAOCHx 寄存器中的数据进行计算得到相应电压值，但是无法对 DAC_DATAOCHx 寄存器直接写入数据，需要写入 DAC_DHx 寄存器之后通过相应的触发从而使 DAC_DHx 中的数据加载到 DAC_DATAOCHx 中。

当关闭通道触发时（寄存器 DAC_CTRL 中的 TRGENCHx 位置 0），写入 DAC_DHx 寄存器中的值会在一个 APB1 时钟周期后自动传入 DAC_DATAOCHx

中。

当打开通道触发时（寄存器 DAC_CTRL 中的 TRGENCHx 位置 1），写入 DAC_DHx 寄存器中的值会根据选择的触发源不同而经过不同的时钟周期后传入 DAC_DATAOCHx 中。可供选择的触发源有三类：

- 定时器的更新事件
- 外部中断触发
- 软件触发

当选择定时器的更新事件和外部中断作为触发源时，会经过三个 APB1 时钟周期后完成传输；当选择软件触发时，经过一个 APB1 时钟周期便完成传输。

当数据传输到 DAC_DATAOCHx 寄存器中时，经过一段时间之后，数字量便被线性的转换成为模拟电压输出。中间的转换时间会根据电源电压和模拟输出负载的不同而不同。

28.4.2 DAC 参考电压及输出

DAC 使用 V_{REF} 作为参考电压，将 V_{SSA} 接地，可得到 DAC 的输出电压范围为：0-V_{REF}。

DAC 输出计算公式为：DAC 输出=V_{REF} * （DATAOCHx/4095）。

28.4.3 DAC 数据格式

单通道 DAC

三种模式下对应写入的寄存器分别如下

- 8 位数据右对齐：DAC_DH8Rx[7:0]
- 12 位数据左对齐：DAC_DH12Lx[15:4]
- 12 位数据右对齐：DAC_DH12Rx[11:0]

双通道 DAC

三种模式下对应写入的寄存器分别如下

- 8 位数据右对齐：DAC_DH8RD[15:0]
- 12 位数据左对齐：DAC_DH12LDUAL[15:4]、DAC_DH12LDUAL[31:20]
- 12 位数据右对齐：DAC_DH12RDUAL[11:0]、DAC_DH12RDUAL[27:16]

28.4.4 DAC 波形生成

DAC 每个通道都可以单独生成噪声和三角波。

28.4.5 DAC 双通道转换

当两路通道同时工作时，为了有效的利用总线带宽，写入的数据可以写入到共用寄存器：DH8RDUAL、DH12RDUAL 和 DH12LDUAL。

对于双通道转换，可以分为独立转换和同步转换两大类，具体配置及描述如下。

28.4.5.1 独立触发

禁用波形发生器

（1） 使能两个通道的触发模式；

（2） 配置两个通道使用不同的触发源。

使用相同 LFSR

（1） 使能两个通道的触发模式；

（2） 配置两个通道使用不同的触发源；

（3） 使能两个通道的噪声生成功能，设置相同的 LFSR 屏蔽值。

使用不同 LFSR

（1） 使能两个通道的触发模式；

（2） 配置两个通道使用不同的触发源；

（3） 使能两个通道的噪声生成功能，设置不同的 LFSR 屏蔽值。

产生相同三角波

（1） 使能两个通道的触发模式；

（2） 配置两个通道使用不同的触发源；

（3） 使能两个通道的三角波生成功能，设置相同的三角波幅值。

产生不同三角波

（1） 使能两个通道的触发模式；

（2） 配置两个通道使用不同的触发源；

（3） 使能两个通道的三角波生成功能，设置不同的三角波幅值。

28.4.5.2 同步触发

同步软件启动

失能两个通道的触发模式，写入数据后等待一个 APB1 时钟周期之后分别同时传入 DAC_DATAOCH1 和 DAC_DATAOCH2 寄存器。

禁用波形发生器

（1） 使能两个通道的触发模式；

（2） 配置两个通道使用相同的触发源。

使用相同 LFSR

（1） 使能两个通道的触发模式；

（2） 配置两个通道使用相同的触发源；

（3） 使能两个通道的噪声生成功能，设置相同的 LFSR 屏蔽值。

使用不同 LFSR

（1） 使能两个通道的触发模式；

（2） 配置两个通道使用相同的触发源；

（3） 使能两个通道的噪声生成功能，设置不同的 LFSR 屏蔽值。

产生相同三角波

（1） 使能两个通道的触发模式；

（2） 配置两个通道使用相同的触发源；

（3） 使能两个通道的三角波生成功能，设置相同的三角波幅值。

使用不同三角波

（1） 使能两个通道的触发模式；

（2） 配置两个通道使用相同的触发源；

（3） 使能两个通道的三角波生成功能，设置不同的三角波幅值。

28.5 寄存器地址映射

表格 145 DAC 寄存器地址映射

寄存器名	描述	偏移地址
DAC_CTRL	DAC 控制寄存器	0x00
DAC_SWTRG	DAC 软件触发寄存器	0x04
DAC_DH12R1	DAC 通道 1 的 12 位右对齐数据保持寄存器	0x08
DAC_DH12L1	DAC 通道 1 的 12 位左对齐数据保持寄存器	0x0C
DAC_DH8R1	DAC 通道 1 的 8 位右对齐数据保持寄存器	0x10
DAC_DH12R2	DAC 通道 2 的 12 位右对齐数据保持寄存器	0x14
DAC_DH12L2	DAC 通道 2 的 12 位左对齐数据保持寄存器	0x18
DAC_DH8R2	DAC 通道 2 的 8 位右对齐数据保持寄存器	0x1C
DAC_DH12RDUAL	双 DAC 的 12 位右对齐数据保持寄存器	0x20
DAC_DH12LDUAL	双 DAC 的 12 位左对齐数据保持寄存器	0x24

寄存器名	描述	偏移地址
DAC_DH8RDUAL	双 DAC 的 8 位右对齐数据保持寄存器	0x28
DAC_DATAOCH1	DAC 通道 1 数据输出寄存器	0x2C
DAC_DATAOCH2	DAC 通道 2 数据输出寄存器	0x30

28.6 寄存器功能描述

28.6.1 DAC 控制寄存器（DAC_CTRL）

偏移地址：0x00

复位值：0x0000 0000

位/域	名称	R/W	描述
0	ENCH1	R/W	使能 DAC 通道 1（DAC Channel1 Enable） 0：禁止 1：使能
1	BUFFDCH1	R/W	禁止 DAC 通道 1 输出缓存（DAC Channel1 Output Buffer Disable） 0：使能 1：禁止
2	TRGENCH1	R/W	使能 DAC 通道 1 触发（DAC Channel1 Trigger Enable） 0：禁止 1：使能
5:3	TRGSELCH1	R/W	选择 DAC 通道 1 触发源（DAC Channel1 Trigger Source Select） 在通道 1 触发使能（TRGENCH1=1）的情况下，可通过该寄存器选择触发源。 000：TMR6 TRGO 事件 001：TMR8 TRGO 事件 010：TMR7 TRGO 事件 011：TMR5 TRGO 事件 100：TMR2TRGO 事件 101：TMR4 TRGO 事件 110：外部中断线 9 111：软件触发
7:6	WAVENCH1	R/W	使能 DAC 通道 1 产成噪声/三角波（DAC Channel1 Noise/Triangle Wave Generation Enable） 00：不产生波形 01：产生噪声波形 1x：产生三角波波形
11:8	MAMPSELCH1	R/W	选择 DAC 通道 1 屏蔽的 LFSR 位/三角波幅值（Select DAC Channel2 LFSR Bit Mask/Triangle Wave Amplitude Selector） 在“产生 LFSR 噪声”模式下，通过该位选择屏蔽 LFSR 的位； 在“产生三角波”模式下通过该位选择三角波的幅值。 0000：不屏蔽 LFSR 位 0/ 三角波幅值等于 1 0001：不屏蔽 LFSR 位[1:0] / 三角波幅值等于 3 0010：不屏蔽 LFSR 位[2:0] / 三角波幅值等于 7 0011：不屏蔽 LFSR 位[3:0] / 三角波幅值等于 15 0100：不屏蔽 LFSR 位[4:0] / 三角波幅值等于 31

位/域	名称	R/W	描述
			0101：不屏蔽 LFSR 位[5:0]/ 三角波幅值等于 63 0110：不屏蔽 LFSR 位[6:0] / 三角波幅值等于 127 0111：不屏蔽 LFSR 位[7:0] / 三角波幅值等于 255 1000：不屏蔽 LFSR 位[8:0] / 三角波幅值等于 511 1001：不屏蔽 LFSR 位[9:0] / 三角波幅值等于 1023 1010：不屏蔽 LFSR 位[10:0]/ 三角波幅值等于 2047 ≥1011：不屏蔽 LFSR 位[11:0] / 三角波幅值等于 4095
12	DMAENCH1	R/W	使能 DAC 通道 1DMA（DAC Channel1 DMA Enable） 0：禁止 1：使能
15:13	保留		
16	ENCH2	R/W	使能 DAC 通道 2（DAC Channel2 Enable） 0：禁止 1：使能
17	BUFFDCH2	R/W	禁止 DAC 通道 2 输出缓存（DAC Channel2 Output Buffer Disable） 0：使能 1：禁止
18	TRGENCH2	R/W	使能 DAC 通道 2 触发（DAC Channel2 Trigger Enable） 0：禁止 1：使能
21:19	TRGSELCH2	R/W	选择 DAC 通道 2 触发源（DAC Channel2 Trigger Source Select） 在通道 2 触发使能（TRGENCH2=1）的情况下，可通过该寄存器选择触发源 000：TMR6 TRGO 事件 001：TMR8 TRGO 事件 010：TMR7 TRGO 事件 011：TMR5 TRGO 事件 100：TMR2TRGO 事件 101：TMR4 TRGO 事件 110：外部中断线 9 111：软件触发
23:22	WAVENCH2	R/W	使能 DAC 通道 2 噪声/三角波生成（DAC Channel2 Noise/Triangle Wave Generation Enable） 00：不产生波形 01：产生噪声波形 1x：产生三角波波形
27:24	MAMPSELCH2	R/W	选择 DAC 通道 2 屏蔽的 LFSR 位/三角波幅值（Select DAC Channel2 LFSR Bit Mask/Triangle Wave Amplitude Selector） 在“产生 LFSR 噪声”模式下，通过该位选择屏蔽 LFSR 的位； 在“产生三角波”模式下通过该位选择三角波的幅值。 0000：不屏蔽 LFSR 位 0/ 三角波幅值等于 1 0001：不屏蔽 LFSR 位[1:0] / 三角波幅值等于 3 0010：不屏蔽 LFSR 位[2:0] / 三角波幅值等于 7 0011：不屏蔽 LFSR 位[3:0] / 三角波幅值等于 15 0100：不屏蔽 LFSR 位[4:0] / 三角波幅值等于 31 0101：不屏蔽 LFSR 位[5:0]/ 三角波幅值等于 63

位/域	名称	R/W	描述
			0110：不屏蔽 LFSR 位[6:0] / 三角波幅值等于 127 0111：不屏蔽 LFSR 位[7:0] / 三角波幅值等于 255 1000：不屏蔽 LFSR 位[8:0] / 三角波幅值等于 511 1001：不屏蔽 LFSR 位[9:0] / 三角波幅值等于 1023 1010：不屏蔽 LFSR 位[10:0]/ 三角波幅值等于 2047 ≥1011：不屏蔽 LFSR 位[11:0] / 三角波幅值等于 4095
28	DMAENCH2	R/W	使能 DAC 通道 2DMA（DAC channel2 DMA enable） 0：禁止 1：使能
31:29	保留		

28.6.2 DAC 软件触发寄存器（DAC_SWTRG）

偏移地址：0x04

复位值：0x0000 0000

位/域	名称	R/W	描述
0	SWTRG1	W	使能 DAC 通道 1 软件触发（DAC Channel1 Software Trigger Enable） 该位可由软件置 1、清 0；一旦寄存器 DAC_DH1 的数据传入寄存器 DAC_DATAOCH1，该位由硬件清 0。 0：禁止 1：使能
1	SWTRG2	W	使能 DAC 通道 2 软件触发（DAC Channel2 Software Trigger Enable） 该位可由软件置 1、清 0；一旦寄存器 DAC_DH2 的数据传入寄存器 DAC_DATAOCH2，该位由硬件清 0。 0：禁止 1：使能
31:2	保留		

28.6.3 DAC 通道 1 的 12 位右对齐数据保持寄存器（DAC_DH12R1）

偏移地址：0x08

复位值：0x0000 0000

位/域	名称	R/W	描述
11:0	DATA	R/W	DAC 通道 1 的 12 位右对齐数据（DAC Channel1 12-Bit Right-Aligned Data） 该位由软件写入，表示 12 位 DAC 通道 1 的数据。
31:12	保留		

28.6.4 DAC 通道 1 的 12 位左对齐数据保持寄存器（DAC_DH12L1）

偏移地址：0x0C

复位值：0x0000 0000

位/域	名称	R/W	描述
3:0	保留		
15:4	DATA	R/W	DAC 通道 1 的 12 位左对齐数据（DAC Channel1 12-Bit Left-Aligned Data）
31:16	保留		

28.6.5 DAC 通道 1 的 8 位右对齐数据保持寄存器（DAC_DH8R1）

偏移地址：0x10

复位值：0x0000 0000

位/域	名称	R/W	描述
7:0	DATA	R/W	DAC 通道 1 的 8 位右对齐数据（DAC Channel1 8-bit Right-Aligned Data）
31:8	保留		

28.6.6 DAC 通道 2 的 12 位右对齐数据保持寄存器（DAC_DH12R2）

偏移地址：0x14

复位值：0x0000 0000

位/域	名称	R/W	描述
11:0	DATA	R/W	DAC 通道 2 的 12 位右对齐数据（DAC Channel2 12-bit Right-Aligned Data）
31:12	保留		

28.6.7 DAC 通道 2 的 12 位左对齐数据保持寄存器（DAC_DH12L2）

偏移地址：0x18

复位值：0x0000 0000

位/域	名称	R/W	描述
3:0	保留		
15:4	DATA	R/W	DAC 通道 2 的 12 位左对齐数据（DAC Channel2 12-Bit Left-Aligned Data）
31:16	保留		

28.6.8 DAC 通道 2 的 8 位右对齐数据保持寄存器（DAC_DH8R2）

偏移地址：0x1C

复位值：0x0000 0000

位/域	名称	R/W	描述
7:0	DATA	R/W	DAC 通道 2 的 8 位右对齐数据（DAC Channel2 8-bit Right-Aligned Data）
31:8	保留		

28.6.9 双 DAC 的 12 位右对齐数据保持寄存器（DAC_DH12RDUAL）

偏移地址：0x20

复位值：0x0000 0000

位/域	名称	R/W	描述
11:0	DATACH1	R/W	DAC 通道 1 的 12 位右对齐数据（DAC Channel1 12-Bit Right-Aligned Data）
15:12	保留		
27:16	DATACH2	R/W	DAC 通道 2 的 12 位右对齐数据（DAC Channel2 12-Bit Right-Aligned Data）
31:28	保留		

28.6.10 双 DAC 的 12 位左对齐数据保持寄存器（DAC_DH12LDUAL）

偏移地址：0x24

复位值：0x0000 0000

位/域	名称	R/W	描述
3:0	保留		
15:4	DATACH1	R/W	DAC 通道 1 的 12 位左对齐数据（DAC Channel1 12-Bit Left-Aligned Data）
19:16	保留		
31:20	DATACH2	R/W	DAC 通道 2 的 12 位左对齐数据（DAC Channel212-Bit Left-Aligned Data）

28.6.11 双 DAC 的 8 位右对齐数据保持寄存器（DAC_DH8RDUAL）

偏移地址：0x28

复位值：0x0000 0000

位/域	名称	R/W	描述
7:0	DATACH1	R/W	DAC 通道 1 的 8 位右对齐数据（DAC Channel1 8-Bit Right-Aligned Data）
15:8	DATACH2	R/W	DAC 通道 2 的 8 位右对齐数据（DAC Channel2 8-Bit Right-Aligned Data）
31:16	保留		

28.6.12 DAC 通道 1 数据输出寄存器（DAC_DATAOCH1）

偏移地址：0x2C

复位值：0x0000 0000

位/域	名称	R/W	描述
11:0	DATA	R	DAC 通道 1 输出数据（DAC Channel1 Data Output）
31:12	保留		

28.6.13 DAC 通道 2 数据输出寄存器（DAC_DATAOCH2）

偏移地址：0x30

复位值：0x0000 0000

位/域	名称	R/W	描述
11:0	DATA	R	DAC 通道 2 输出数据（DAC Channel2 Data Output）
31:12	保留		

29 循环冗余校验计算单元（CRC）

29.1 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。

29.2 功能描述

29.2.1 计算方式

使用 CRC-32（以太网）多项式：0x4C11DB7

$$（X^{32}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^{8}+X^{7}+X^{5}+X^{4}+X^{2}+X+1）$$

29.2.2 计算时间

其计算时间为 4 个 AHB 时钟周期。

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）。在计算期间，会暂停 CPU 的写操作，因此可以对寄存器 CRC_DATA 进行“背靠背”写入或连续地“读-写”操作。

29.3 寄存器地址映射

表格 146 CRC 计算单元寄存器地址映射

寄存器名	描述	偏移地址
DATA	数据寄存器	0x00
INDATA	独立数据寄存器	0x04
CTRL	控制寄存器	0x08

29.4 寄存器功能描述

CRC 计算单元包括 2 个数据寄存器和 1 个控制寄存器。

29.4.1 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 位数据寄存器（32bit Data Register） 执行写操作时，作为 CRC 计算的新数据输入口；执行读操作时，返回 CRC 计算的结果。

29.4.2 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 位数据寄存器（Independent 8bit Data Register） 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。

29.4.3 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	W	复位 CRC 计算单元（Reset CRC Calculation Unit） 复位后设置数据寄存器为 0xFFFF FFFF，对该位只能软件写 1、硬件自动清 0。
31:1	保留		

30 浮点运算单元（FPU）

30.1 简介

FPU（浮点运算单元）完全支持单精度加法、减法、乘法、除法、乘法和累加以及平方根的运算。同时，它还提供定点和浮点数据格式之间的转换，以及浮点常数指令。FPU 提供浮点估算功能，符合 ANSI（美国国家标准学会）/IEEE（电气和电子工程师协会） 754-2008 标准和 IEEE 二进制浮点算法标准，称为 IEEE 754 标准。

注：仅适用于 APM32E103xCxE 产品。

30.2 功能描述

30.2.1 支持算法

FPU 仅支持单精度，支持 IEEE754 合规性，包含的算法：CMP、SUM、SUB、PRDCT、MAC、DIV、INVSQRT、SUMSQ、DOT、浮点到整数转换和整数到浮点转换。

表格 147 FPU 支持的算法

术语	定义
I2FP	整数到浮点转换
FP2I	浮点到整数转换
CMP（p，q）	p 和 q 比较
SUM（p，n）	$SUM(p,n)=\sum_{(i=0,\ n)} pi = p0+p1+..+pi+..+pn$，0≤n<16
SUB（p，n）	$SUB(p,n)=p0-\sum_{(i=0,\ n)} pi = p0-p1-..-pi-..-pn$，0≤n<16
PRDCT（p，n）	Product：$PRDCT(p.n)=\prod_{(i=0,n)} pi = p0*p1*..*pi*..*pn$，0≤n<16
DIV（p，q）	Division:$DIV(p,q)=\frac{p}{q}$
INVSQRT（p）	Square Root：INVSQRT（p）=$\frac{1}{\sqrt{p}}$
MAC（p，q，s）	Floating-Point Multiply-Add：MAC（p，q，s） $=p*q+s$
SUMSQ（p，n）	∑（i=1,n）pi2，其中 p 是长度 n 的矢量，0≤n<16
DOT（p，q，n）	∑（i=1,n）（pi * qi），其中 p 和 q 是长度 n 的矢量，0≤n<16
SIN（x）	$x\in(-\infty,+\infty)$,x 以弧度表示
COS（x）	$x\in(-\infty,+\infty)$，x 以弧度表示
TAN（x）	$x\in(-\infty,+\infty)$，x 以弧度表示
ASIN（x）	$x\in[-1,+1]$，输出值以弧度表示，输出值在区间$[-\pi/2,+\pi/2]$内
ACOS（x）	$x\in[-1,+1]$，输出值以弧度表示，输出值在区间 $[0,\pi]$内
ATAN（x）	$x\in[-\infty,+\infty]$，输出值以弧度表示，输出值在区间$[-\pi/2,+\pi/2]$内

术语	定义
ATAN2（y，x）	$x, y \in [-\infty, +\infty]$，输出值以弧度表示，输出值在区间$[-\pi, +\pi]$内

30.2.2 软件编程

30.2.2.1 编程流程

推荐使用我们提供的浮动库连接用户程序，而不是在寄存器基础上编程。

30.2.2.2 单操作数操作

对于 FP2I、I2FP、INVSQRT、SIN、COS、TAN、ASIN、ACOS 和 ATAN，建议配置如下：

（1） 配置 CTRL 寄存器。选择所需要的操作模式，设置起始（STR）位。如果是 INVSQRT 模式，设置循环次数以提高 CTRL[7:6]的精确度。

（2） 操作数写入 FPU_FPI1

（3） 等待中断完成或者检查 FPU_ISTS 寄存器的完成（DONE）标志

（4） 从 FPU_FPO 读取结果

30.2.2.3 两个操作数操作

对于 DIV，建议配置如下：

（1） 配置 FPU_CTRL 寄存器，选择 DIV 模式和设置起始（STR）位。

（2） 第一个操作数写入 FPU_FPI1

（3） 第二个操作数写入 FPU_FPI2

（4） 等待中断完成或者检查 FPU_ISTS 寄存器的完成（DONE）标志

（5） 从 FPU_FPO 读取结果

对于 CMP，建议配置如下：

（1） 配置 FPU_CTRL 寄存器，选择 CMP 模式和设置起始（STR）位。

（2） 第一个操作数写入 FPU_FPI1

（3） 第二个操作数写入 FPU_FPI2

（4） 等待中断完成或者检查 FPU_ISTS 寄存器的完成（DONE）标志

（5） 从 FPU_ISTS[3:2]读取结果 CMP 结果

对于 ATAN2，建议配置如下：

（1） 配置 FPU_CTRL 寄存器，选择 ATAN2 模式和设置起始（STR）位。

（2） 第一个操作数写入 FPU_FPI1

（3） 第二个操作数写入 FPU_FPI2

（4） 等待中断完成或者检查 FPU_ISTS 寄存器的完成（DONE）标志

（5） 从 FPU_FPO 读取结果

30.2.2.4 三个操作数操作

对于 MAC，建议配置如下：

（1） 配置 FPU_CTRL 寄存器，选择 MAC 模式和设置起始（STR）位。

（2） 第一个操作数写入 FPU_FPI1

（3） 第二个操作数写入 FPU_FPI2

（4） 第三个操作数写入 FPU_FPI2

（5） 等待中断完成或者检查 FPU_ISTS 寄存器的完成（DONE）标志

（6） 从 FPU_FPO 读取结果

30.2.2.5 n 个操作数操作

对于 SUM、SUB、PRDCT、SUMSQ，建议配置如下：

（1） 配置 FPU_CTRL 寄存器，选择需要的模式，在 nLEN 字段录入 n 的值，设置起始（STR）位

（2） P0 操作数写入 FPU_FPI1

（3） P1..n-1 的操作数依次写入 FPU_FPI2

（4） 等待中断完成或者检查 FPU_ISTS 寄存器的完成（DONE）标志

（5） 从 FPU_FPO 读取结果

对于 DOT，建议配置如下：

（1） 配置 FPU_CTRL 寄存器，选择 DOT 模式，在 nLEN 字段录入 n 的值，设置起始（STR）位

（2） P0 操作数写入 FPU_FPI1

（3） Q0 操作数依次写入 FPU_FPI2

（4） P1 操作数写入 FPU_FPI1

（5） Q1 操作数依次写入 FPU_FPI2

（6） …

（7） Pn-1 操作数写入 FPU_FPI1

（8） Qn-1 操作数依次写入 FPU_FPI2

（9） 等待中断完成或者检查 FPU_ISTS 寄存器的完成（DONE）标志

（10） 从 FPU_FPO 读取结果

30.3 寄存器地址映射

表格 148 FPU 寄存器地址映射

寄存器名	描述	偏移地址
FPU_CTRL	控制寄存器	0x00
FPU_ISTS	中断状态寄存器	0x04
FPU_FPO	浮点输出值寄存器	0x08
FPU_FPI1	浮点输入值寄存器 1	0x0C
FPU_FPI2	浮点输入值寄存器 2	0x10
FPU_IM	中断屏蔽寄存器	0x14

30.4 寄存器功能描述

30.4.1 控制寄存器（FPU_CTRL）

偏移地址：0x0

复位值：0x0000 0000

位/域	名称	R/W	描述
0	STR	R/W	启动 FPU 0：无影响 1：启动 FPU。启动后由硬件自动清零。
5:1	OP	R/W	FPU 操作（Operation）方式 当位域为不同的值（十进制）时，代表的 FPU 操作方式： 0：ASIN 1：ACOS 2：ATAN 3：ATAN2 4：SIN 5：COS 6：TAN 7~15：Reserved 16：SUM（p，n），Summation 17：SUMSQ（p，n），Sum of squares 18：SUB（p，n），Subtraction 19：PRDCT（p，n），Production 20：DOT（p，q，n），Dot Production 21：MAC（p，q，s），Floating-Point Multiply-Add 22：DIV（p，q），Division 23：INVSQRT（p），Square Root 24：CMP（p,q） 25：I2FP，整数到浮点转换

位/域	名称	R/W	描述
			26：FP2I，浮点到整数转换 27~31：保留
7:6	ITERTION	R/W	提高精确度算法 INVSQRT 的重复时间
11:8	nLEN	R/W	算式 SUM（p，n），SUB（p，n），SUMSQR（p，n），PRDCT（p，n） 和 DOT（p，q，n） 的长度。
31:12	保留		

30.4.2 中断状态寄存器（FPU_ISTS）

偏移地址：0x4

复位值：0x0000 0000

位/域	名称	R/W	描述
0	DONE	R/W	FPU 计算完成标志 0：FPU 计算未完成 1：FPU 计算完成
1	BUSY	R/W	FPU 忙标志 0：FPU 处于空闲状态。 1：FPU 处于忙状态。FPU 运行后硬件置位，FPU 完成计算后硬件自动清除。
3:2	CMPR[3:2]	R/W	比较算法结果寄存器 00：a = b。 01：a < b。 10：a > b。 11：其中一个输入值是非数（NaN）时，是无序的
4	ZERO	R/W	零标志 0：无意义。 1：计算结果为零或四舍五入后结果为零
5	INF	R/W	无穷大 0：无意义 1：计算结果无穷大或四舍五入后结果无穷大
6	INVALID	R/W	浮点操作无效 0：无意义 1：非法的输入操作
7	TINY	R/W	计算结果极小 0：无意义。 1：经过四舍五入之后，浮点是一个小于最小正常化数的量，但不是精确零。
8	HUGE	R/W	计算结果极大。 0：无意义。 1：经过四舍五入之后，有限的浮点结果是一个大于最大正常化数的量。
9	INEXACT	R/W	计算结果不精确 0：无意义。 1：整数或者浮点输出值不等于无穷大的精确结果
10	HUGEINT	R/W	极大整数补数 0：无意义

位/域	名称	R/W	描述
			1：经过四舍五入之后，整数结果是一个大于最大可表示的两个相同符号的补数的量
11	DIVBY0	R/W	被零除 0：无意义 1：运行了被零除操作
31:12	保留		

30.4.3 浮点输出值寄存器（FPU_FPO）

偏移地址：0x8

复位值：0x0000 0000

位/域	名称	R/W	描述
31:0	FPO	R	浮点输出值

30.4.4 浮点输入值寄存器 1（FPU_FPI1）

偏移地址：0xc

复位值：0x0000 0000

位/域	名称	R/W	描述
31:0	FPI1	R/W	浮点输入值 1

30.4.5 浮点输入值寄存器 2（FPU_FPI2）

偏移地址：0x10

复位值：0x0000 0000

位/域	名称	R/W	描述
31:0	FPI2	R	浮点输入值 2

30.4.6 中断屏蔽寄存器（FPU_IM）

偏移地址：0x14

复位值：0x0000 0000

位/域	名称	R/W	描述
0	DONE	R/W	操作完成中断屏蔽 0：操作完成中断使能 1：操作完成中断屏蔽
1	INVALID	R/W	操作中断屏蔽无效 0：无效操作中断使能 1：无效操作中断屏蔽
31:2	保留		

31 芯片电子签名

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。

31.1 主存储区容量寄存器

31.1.1 闪存容量寄存器（16 位）

基地址：0x1FFF F7E0

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：0x0080=128 K 字节

31.2 96 位唯一芯片 ID

唯一 ID 的用途可以有：

- 作为序列号（例如 USBD 字符序列号或其他的终端应用）
- 作为密码，在编写闪存时，将此唯一标识与软件加解密算法结合使用，提高代码在闪存存储器内的安全性
- 用于激活带安全机制的启动过程
- 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。

基地址：0x1FFF F7E8

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	U_ID[15:0]	R	唯一身份标志 15:0 位

偏移地址：0x02

只读，其值在出厂时编写

位/域	名称	R/W	描述
15:0	U_ID[31:16]	R	唯一身份标志 31:16 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

32 版本历史

表格 149 文件版本历史

日期	版本	变更历史
2021.7.23	V1.0	新建
2022.3.15	V1.1	（1）修改 SDIO 模块内容 （2）修改 DMC_CTRL2 寄存器 （3）修改 CAN 通信模式部分的内容 （4）修改 Flash 的选项字节的 WDTSEL 位 （5）修改 CAN 模块过滤寄存器 （6）将“SCLKSW”改为“SCLKSEL”，“SCLKSWSTS”改为“SCLKSELSTS”
2022.6.21	V1.2	（1）修改 DMC 模块配置举例和偏移地址 （2）修改 SMC 模块有关时序参数的内容和偏移地址 （3）修改 Arm 商标 （4）增加版权说明 （5）修改 USBD 寄存器描述
2022.8.22	V1.3	（1）修改 RCM 模块的复位寄存器、使能寄存器 （2）删除 FPU 中的 RGSQT 描述

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。

©2021-2022 珠海极海半导体有限公司 – 保留所有权利