

概述

PT32 系列 32-bit 低功耗微控制器，整合高效能 ARM Cortex-M0 最高 48MHz 的处理效能，具有高达 32 字节的闪存和高达 4 字节的 SRAM。及多个外设模块。通过低功耗设置程序可以设置 3 种不同的低功耗模式以针对各种应用情况。广泛的增强型外设和 I/O 都可以使设备被采用到不同的应用。闪存程序存储器可以在系统通过 SWD 接口进行重新编程。

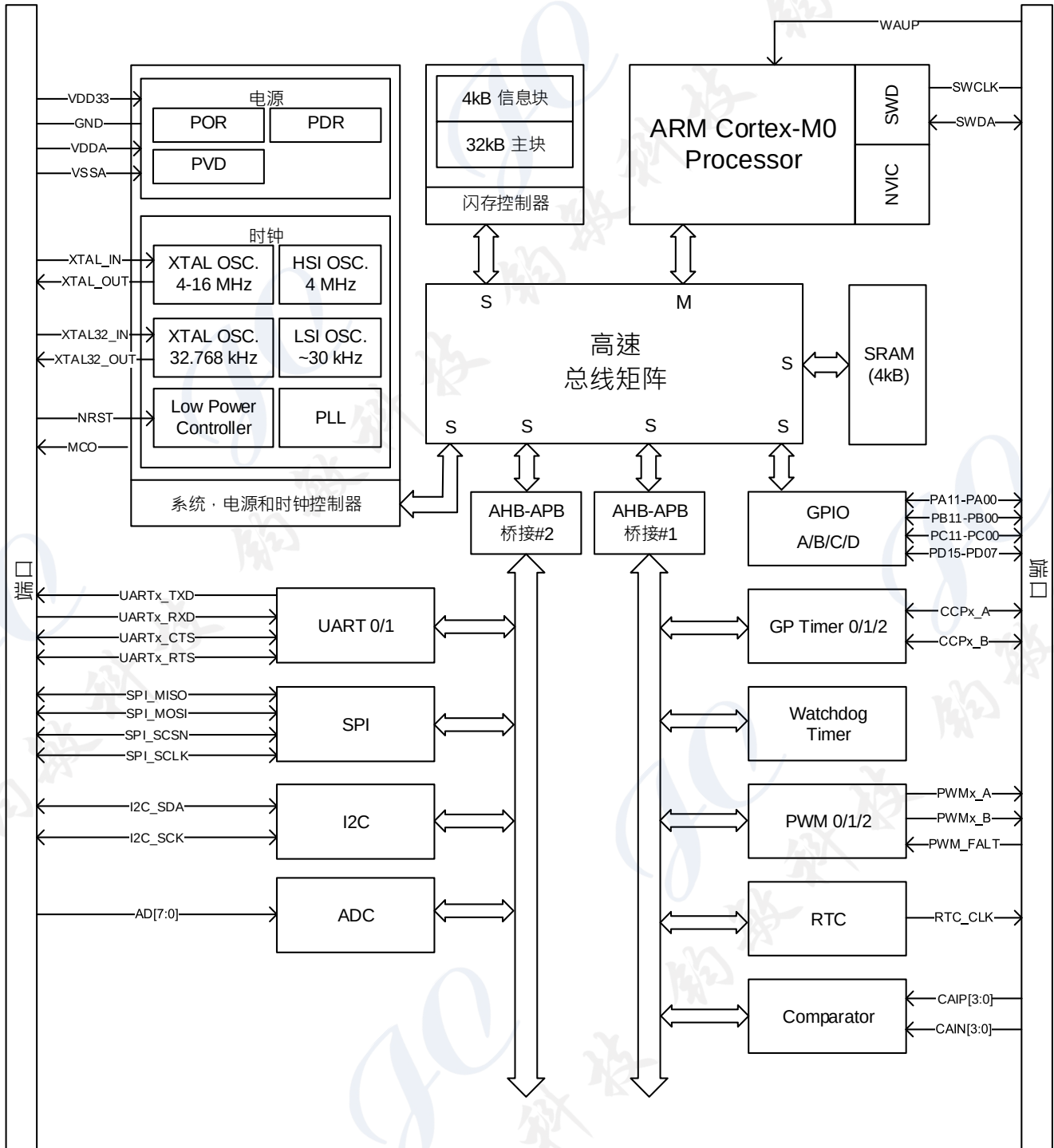
PT32U 通用 MCU 微控制器系列，给你带来任何基本功能。凭借其高度可定制的外设，它简化目标产品的设计过程。

外设集还包括一个外部总线接口，2 个 UART，1×I2C，1 个 SPI，3 个通用 16 位/32 位定时器，一个 RTC，3 个 PWM，1 个 ADC 和 4 个模拟比较器。

特性

- RM Cortex M0 处理器
- 高达 48 MHz
- 闪存 32K 字节
- SRAM 4K 字节
- 电源控制
 - POR/PDR
 - PVD
- 功耗模式
 - 睡眠模式
 - 停止模式
 - 待机模式
- 外围周边设备
- 外设
 - 多达 45 个 GPIO
 - 16/32 位定时器* 6
 - WDT/RTC
 - 8 通道 12 位 ADC
 - 4 比较器与窗口比较
 - 2 UART
 - 具有主/从模式的 I2C
 - SPI 可变位长度（4-16 位）

方框图

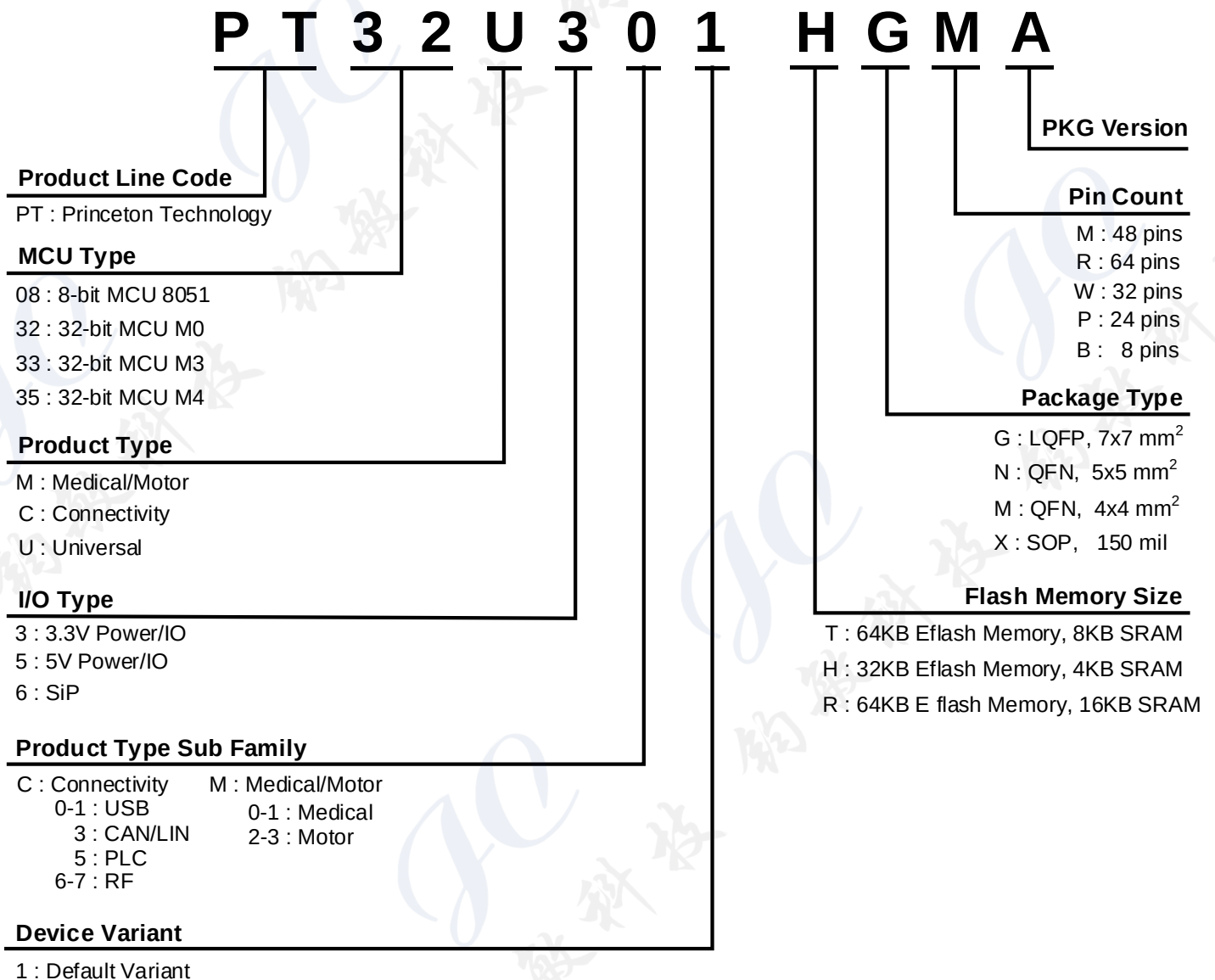


目錄

方框图.....	2
1. 订购信息	4
2. 腳位構造圖.....	5
3. 引腳描述	8
3.1 多功能引脚功能选择	10
3.2 信号线描述.....	12
4. 功能描述	14
4.1 System and Memory Overview.....	14
4.2 ARM ® Cortex™-M0 内核.....	17
4.3 CPU 系统控制	34
4.4 系统控制 (SC)	40
4.5 闪存控制器 (FC).....	71
4.6 通用输入/输出 (GPIO).....	75
4.7 通用异步收发器 (UART)	89
4.8 脉冲宽度调制 (PWM)	116
4.9 模拟/数字转换器 (ADC)	154
4.10 通用定时器 (GPTM)	188
4.11 模拟比较器 (AC)	214
4.12 看门狗定时器 (WDT)	225
4.13 实时时钟 (RTC).....	234
4.14 集成电路总线 (I2C)	255
4.15 串行外设接口 (SPI)	283
5. 电气特性	317
5.1 最大额定值.....	317
5.2 操作条件	317
5.3 I/O 管脚特征.....	317
5.4 片上低压差线性稳压器特性	318
5.5 锁相环特性.....	318
5.6 上电复位特性	319
5.7 NRST 特性	320
5.8 8 MHz 晶振特性	320
5.9 4MHz RCOSC 特性	321
5.10 32 kHz 晶振.....	321
5.11 A / D IO 特性.....	322
5.12 温度传感器特性.....	322
5.13 ADC + PGA 特性	323
5.14 比较器特性.....	324
5.15 RCOSC_32K 特性	324
5.16 功率消耗表.....	325
6. 封装信息	326

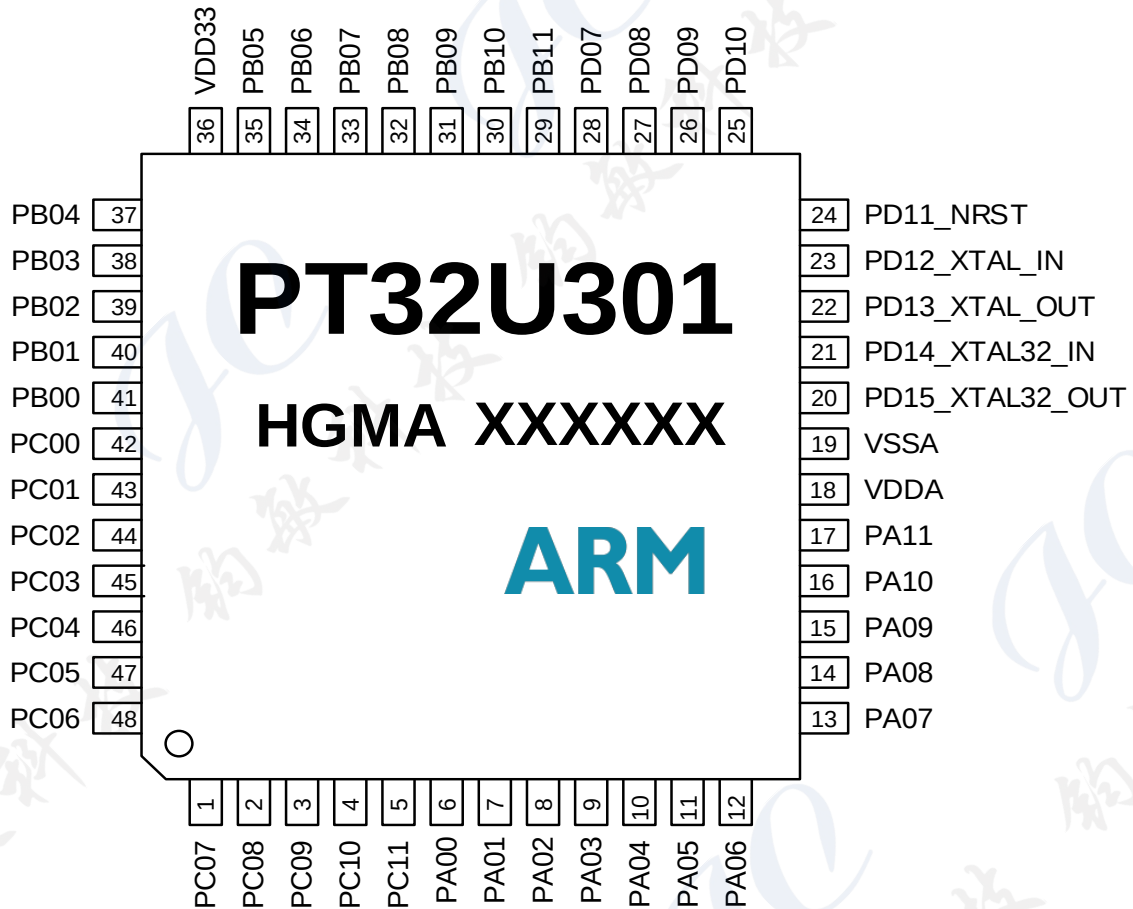
1. 订购信息

订购编号	包装类型	正印
PT32U301	LQFP 48	PT32U301
PT32U301	LQFP 32	PT32U301
PT32U301	SSOP 24	PT32U301
PT32U301	SOP 16	PT32U301
PT32U301	SOP 8	PT32U301

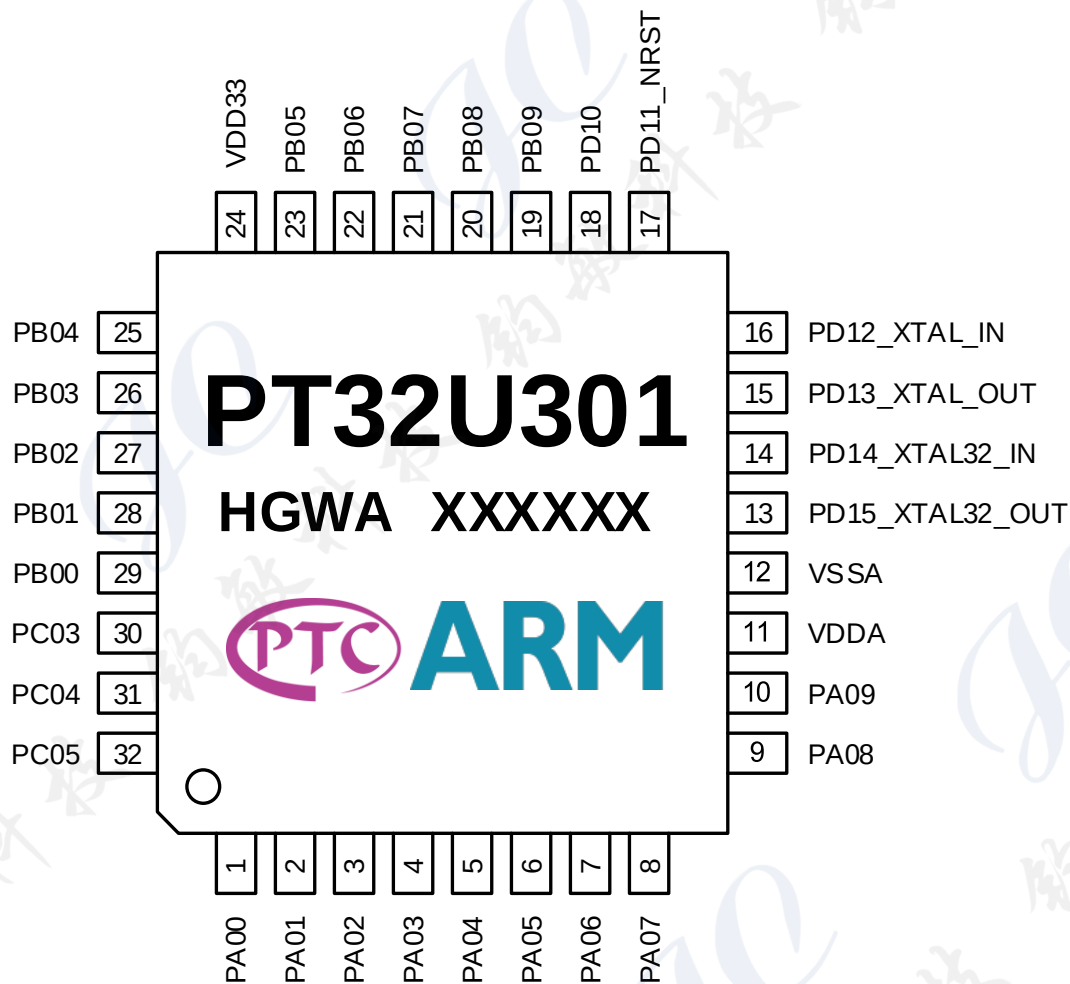


2. 腳位構造圖

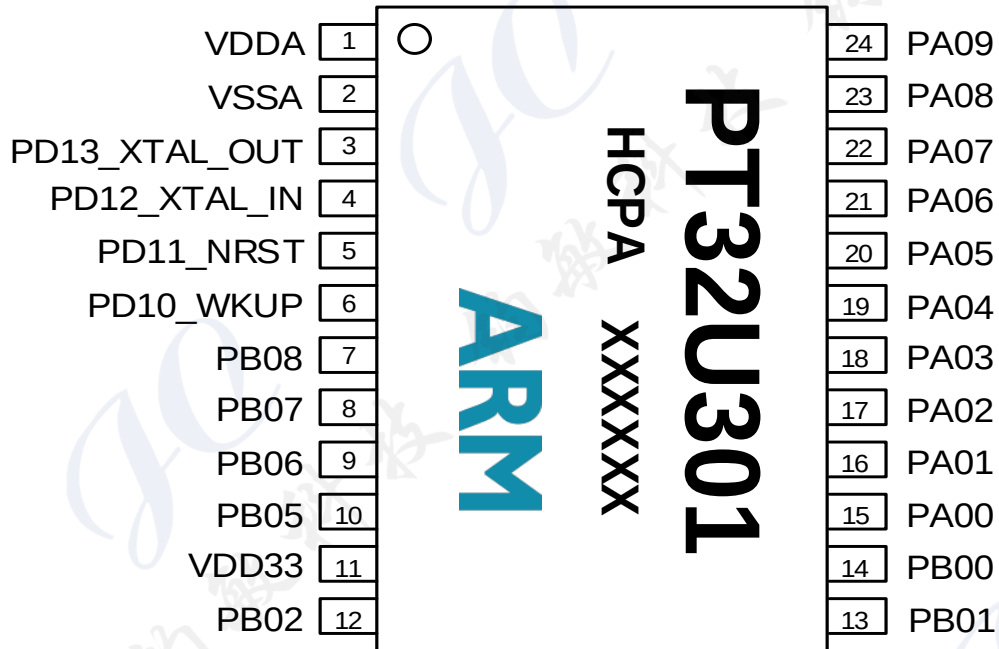
LQFP - 48



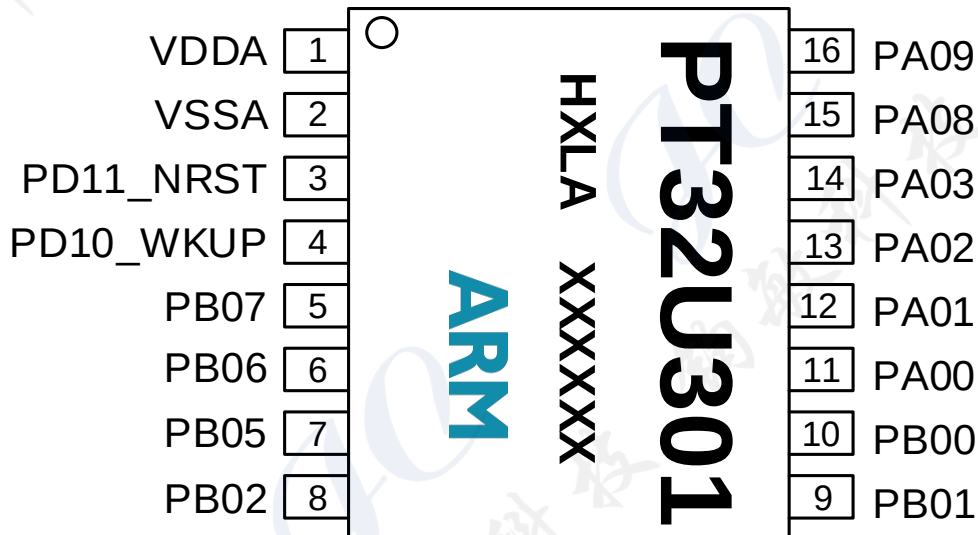
LQFP - 32



SSOP - 24



SOP - 16



3. 引腳描述

引腳名称	I/O	描述	引腳编号				
			LQFP48	LQFP32	SSOP24	SOP16	SOP8
PC07	I/O	General Purpose Digital I/O Pin	1				
PC08	I/O	General Purpose Digital I/O Pin	2				
PC09	I/O	General Purpose Digital I/O Pin	3				
PC10	I/O	General Purpose Digital I/O Pin	4				
PC11	I/O	General Purpose Digital I/O Pin	5				
PA00	I/O	General Purpose Digital I/O Pin	6	1	15	11	6
PA01	I/O	General Purpose Digital I/O Pin	7	2	16	12	7
PA02	I/O	General Purpose Digital I/O Pin	8	3	17	13	
PA03	I/O	General Purpose Digital I/O Pin	9	4	18	14	
PA04	I/O	General Purpose Digital I/O Pin	10	5	19		
PA05	I/O	General Purpose Digital I/O Pin	11	6	20		
PA06	I/O	General Purpose Digital I/O Pin	12	7	21		
PA07	I/O	General Purpose Digital I/O Pin	13	8	22	15	
PA08	I/O	General Purpose Digital I/O Pin	14	9	23	16	
PA09	I/O	General Purpose Digital I/O Pin	15	10	24		
PA10	I/O	General Purpose Digital I/O Pin	16				
PA11	I/O	General Purpose Digital I/O Pin	17				
VDDA	Supply	Analog Voltage Supply	18	11	1	1	8
VSSA	Ground	Analog Ground	19	12	2	2	1
PD15_XTAL32_OUT	I/O	General Purpose Digital I/O Pin/Crystal Output	20	13			
PD14_XTAL32_IN	I/O	General Purpose Digital I/O Pin/Crystal Input	21	14			
PD13_XTAL_OUT	I/O	General Purpose Digital I/O Pin/Crystal Output	22	15	3		
PD12_XTAL_IN	I/O	General Purpose Digital I/O Pin/Crystal input	23	16	4		
PD11_NRST	I/O	General Purpose Digital I/O Pin/Reset	24	17	5	3	2
PD10	I/O	General Purpose Digital I/O Pin	25	18	6	4	
PD09	I/O	General Purpose Digital I/O Pin	26	19			
PD08	I/O	General Purpose Digital I/O Pin	27	20			
PD07	I/O	General Purpose Digital I/O Pin	28				
PB11	I/O	General Purpose Digital I/O Pin	29				
PB10	I/O	General Purpose Digital I/O Pin	30				
PB09	I/O	General Purpose Digital I/O Pin	31				
PB08	I/O	General Purpose Digital I/O Pin	32		7		
PB07	I/O	General Purpose Digital I/O Pin	33	21	8	5	
PB06	I/O	General Purpose Digital I/O Pin	34	22	9	6	
PB05	I/O	General Purpose Digital I/O Pin	35	23	10	7	
VDD33	Supply	Voltage Supply	36	24	11		
PB04	I/O	General Purpose Digital I/O Pin	37	25			
PB03	I/O	General Purpose Digital I/O Pin	38	26			

引脚名称	I/O	描述	引脚编号				
			LQFP48	LQFP32	SSOP24	SOP16	SOP8
PB02	I/O	General Purpose Digital I/O Pin	39	27	12	8	3
PB01	I/O	General Purpose Digital I/O Pin	40	28	13	9	4
PB00	I/O	General Purpose Digital I/O Pin	41	30	14	10	5
PC00	I/O	General Purpose Digital I/O Pin	42				
PC01	I/O	General Purpose Digital I/O Pin	43				
PC02	I/O	General Purpose Digital I/O Pin	44				
PC03	I/O	General Purpose Digital I/O Pin	45	30			
PC04	I/O	General Purpose Digital I/O Pin	46	31			
PC05	I/O	General Purpose Digital I/O Pin	47	32			
PC06	I/O	General Purpose Digital I/O Pin	48				

3.1 多功能引脚功能选择

下表描述了外围信号多任务的线. GPIOx_AFRL 和 GPIOx_AFRH 为复用使能选择寄存器.

注(1): 所有的引脚默认为 AF0 模式,以下引脚除外:

(1) 石英晶体振荡器引脚: PD [15:11]这 4 个引脚默认为 AF8 模式。

(2) SWD 串行调试引脚: PB [6:5] 这 2 个引脚默认为 AF4 模式。在 TSSOP8 引脚封装, 所属 PB[3:2]被默认为 AF4 模式;

引脚名称	复用功能								
	AF0	AF1	AF0	AF3	AF0	AF5	AF0	AF7	AF0
PC07	PC07							PWM1_B	
PC08	PC08							PWM0_B	
PC09	PC09							PWM2_A	
PC10	PC10							PWM1_A	
PC11	PC11							PWM0_A	
PA00	PA00			SPI_MISO	PWM1_A	CCP1_A			AD0_CAIPO
PA01	PA01	UART0_RTS	I2C_SDA	SPI_MOSI	PWM1_B	CCP1_B	UART1_RXD		AD1_CAIN0
PA02	PA02		UART1_TXD	SPI_MISO	PWM2_A	CCP2_A			AD2_CAIPI
PA03	PA03		UART1_RXD	SPI_MOSI	PWM2_B	CCP_2B			AD3_CAIN1
PA04	PA04	UART1_TXD			PWM1_A	CCP1_A			AD4
PA05	PA05	UART1_RXD			PWM1_B	CCP1_B			AD5
PA06	PA06	UART1_CTS			PWM2_A	CCP2_A			AD6
PA07	PA07	UART1_RTS			PWM2_B	CCP2_B			AD7
PA08	PA08	UART1_TXD	I2C_SDA		PWM0_A	CCP1_A			CAIP2
PA09	PA09		UART1_RTS	SPI_SCSN	PWM0_B	CCP0_B			CAIN2
PA10	PA10								CAIP3
PA11	PA11								CAIN3
VDDA									
VSSA									
PD15 XTAL32_OUT	PD15								XTAL32_OUT(1)
PD14 XTAL32_IN	PD14								XTAL32_IN(1)
PD13 XTAL_OUT	PD13								XTAL_OUT(1)
PD12 XTAL_IN	PD12								XTAL_IN(1)
PD11_NRST	PD11			SPI_SCSN	PWM_FAULT	CCP1B	PWM1_B		NRST(1)
PD10	PD10				PWM_FAULT				WKUP_V33
PD09	PD09			SPI_SCSN					
PD08	PD08			SPI_SCKK					
PD07	PD07								
PB11	PB11								
PB10	PB10			SPI_SCKK				PWM2_B	
PB09	PB09			SPI_MISO				PWM1_B	
PB08	PB08		UART1_TXD	SPI_MOSI	PWM_FAULT			PWM0_B	
PB07	PB07	I2C_SDA	UART1_RXD	SPI_SCSN	PWM2_B			PWM2_A	

引脚名称	复用功能								
	AF0	AF1	AF0	AF3	AF0	AF5	AF0	AF7	AF0
PB06	PB06	UART0_CTS	UART1_CTS	SPI_MOSI	PWM2_A	UART0_TXD	SWCLK(2)	PWM1_A	
PB05	PB05	UART0_RTS	UART1_RTS	SPI_MISO	PWM1_B	UART0_RXD	SWDA(2)	PWM0_A	
VDD33									
PB04	PB04			SPI_MISO		UART0_TXD			
PB03	PB03			SPI_MOSI		UART0_RXD	SWCLK(2)		
PB02	PB02	UART0_CTS	I2C_SCK	SPI_SCKK	PWM1_A	CCP1_A	SWDA(2)	MCO	
PB01	PB01	UART0_TXD	I2C_SCK	SPI_SCSN	PWM0_B	CCP0_B			
PB00	PB00	UART0_RXD	I2C_SDA	SPI_MISO	PWM0_A	CCP0_A			
PC00	PC00				PWM_FALT		RTC_1HZ	PWM2_B	
PC01	PC01							PWM1_B	
PC02	PC02							PWM0_B	
PC03	PC03							PWM2_A	
PC04	PC04							PWM1_A	
PC05	PC05							PWM0_A	
PC06	PC06							PWM2_B	

3.2 信号线描述

功能名称	I/O	功能描述
电源		
VDDA	Supply	模拟域3.3 V电源
VDD33	Supply	数字域3.3V电源
VSSA	Ground	地
VSS	Ground	地
通用异步接收器/发送器(UART0, UART1), x = 0, 1		
UARTx_TXD	O	UART x 数据输出引脚
UARTx_RXD	I	UART x 数据输入引脚
UARTx_CTSn	I/O	UART x 允许发送引脚
UARTx_RTSn	I/O	UART x 请求发送引脚
串行调试 (SWD)		
SWCLK	I	SWD 时钟线
SWDA	I/O	SWD 数据线 (输入\输出)
集成总线 (I2C0, I2C1), x= 0, 1		
I2Cx_SDA	I/O	I2C 数据线
I2Cx_SCK	I/O	I2C 时钟线
同步串行 (SPI)		
SPI_MISO	I/O	SPI 主机输入从机输出
SPI_MOSI	I/O	SPI 主机输出从机输入
SPI_SCSN	I/O	SPI 片选信号
SPI_SCLK	I/O	SPI 串行时钟
通用输入输出 (GPIOA, GPIOB, GPIOC, GPIOD)		
PA11-PA00	I/O	GPIO A埠
PB11-PB00	I/O	GPIO B埠
PC11-PC00	I/O	GPIO C埠
PD15-PD04	I/O	GPIO D埠
脉冲调制器 (PWM0, PWM1, PWM2), x = 0, 1, 2		
PWMx_A	O	PWM x 信号
PWMx_B	O	PWM x信号
PWM_FALT	I	PWM 错误输入
通用定时器 (GPT0, GPT1, GPT2, GPT3, GPT4), x = 0, 1, 2, 3, 4		
CCPx_A	I/O	通用定时器x 比较捕获 A
CCPx_B	I/O	通用定时器x 比较捕获B
模拟/数字转换(ADC)		
AD[7:0]	I	ADC 信号信道输入/ * ADC 微分通道输入正或负
模拟比较 (AC0, AC1, AC2, AC3), x=0, 1, 2, 3		
CAIPx	I	比较x 正输入
CAINx	I	比较x 负输入

功能名称	I/O	功能描述
系统控制 (SC)		
XTAL32_IN	I	32.768K RTC实时时钟输入
XTAL32_OUT	O	32.768K RTC实时时钟输出
XTAL_IN	I	8MHZ 高速石英晶体输入
XTAL_OUT	O	8MHZ高速石英晶体输出
NRST	I	系统复位
WKUP	I	唤醒
MCO	O	系统时钟输出
RTC_CLK	O	RTC 时钟输出

4. 功能描述

4.1 SYSTEM AND MEMORY OVERVIEW

PT32 系列 32-bit 低功耗微控制器，整合高效能 ARM Cortex-M0 最高 48MHz 的处理效能，具有高达 32 字节的闪存和高达 4 字节的 SRAM。及多个外设模块。通过低功耗设置程序可以设置 3 种不同的低功耗模式以针对各种应用情况。广泛的增强型外设和 I/O 都可以使设备被采用到不同的应用。闪存程序存储器可以在系统通过 SWD 接口进行重新编程。

PT32U 通用 MCU 微控制器系列，给你带来任何基本功能。凭借其高度可定制的外设，它简化目标产品的設計过程。外设集还包括一个外部总线接口，2 个 UART，1×I2C，1 个 SPI，3 个通用 16 位/ 32 位定时器，一个 RTC，3 个 PWM，1 个 ADC 和 4 个模拟比较器。

此章节主要介绍 PT32U301 的特点, 系统信息和其存储结构。

4.1.1 内存映像

图 4.1-1: 内存映像

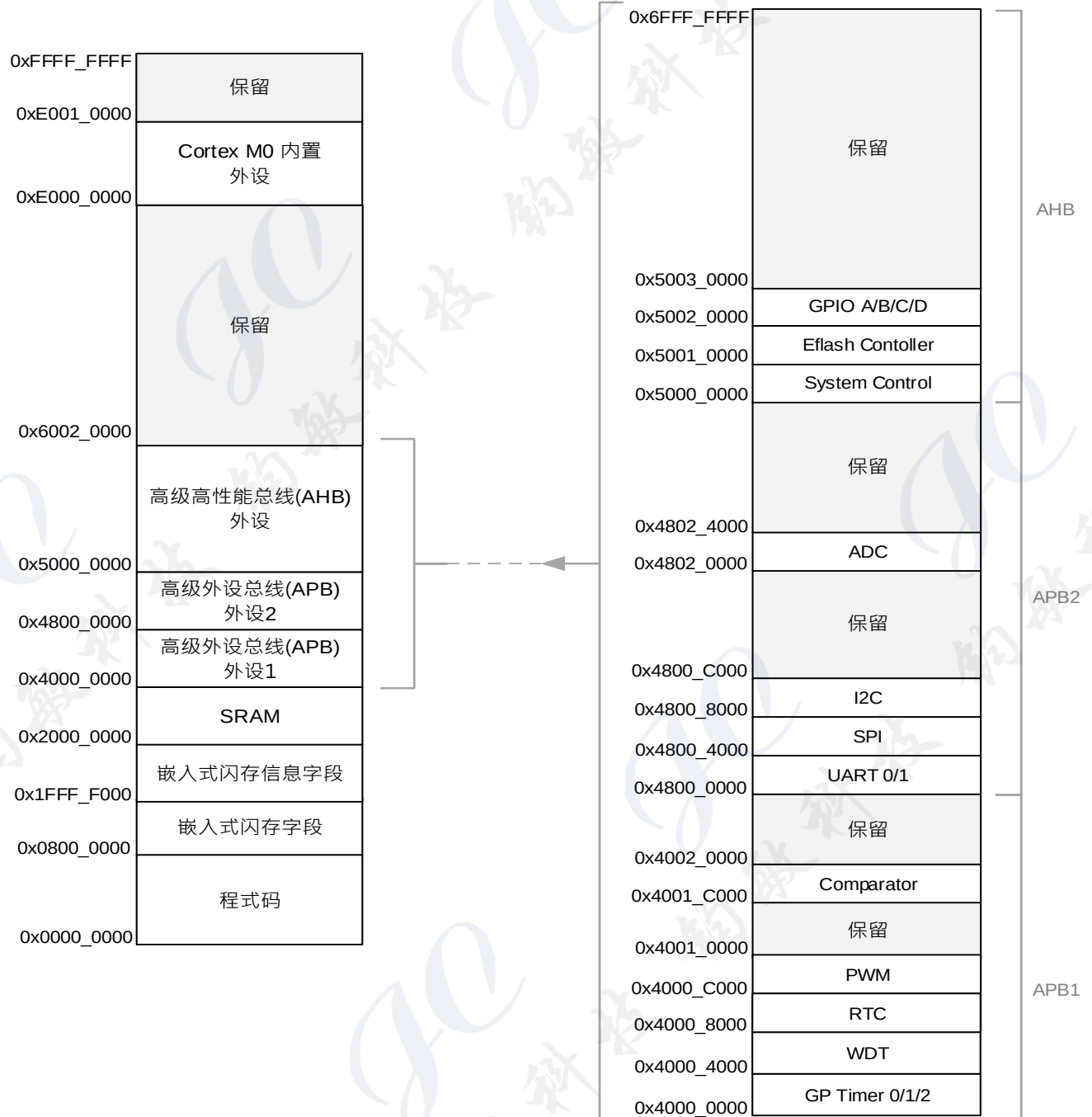


表 4.1-1: 周边外设地址

地址范围		深度 (字节)	外设描述	参考章节	总线
起始	结束				
0x0000_0000	0x1FFF_FFFF	6K	屏蔽ROM，闪存或内存主要取决于启动配置	-	AHB
0x0800_0000	0x0800_7FFF	32K	嵌入式闪存字段	-	
0x1FFF_F000	0x1FFF_FBFF	3K	嵌入式闪存信息字段	4.1.1	
0x1FFF_FC00	0x1FFF_FFFF	1K	嵌入式闪存信息字段	4.1.1	
0x2000_0000	0x2000_0FFF	4K	System RAM	4.1.1	
0x2000_1000	0x3FFF_FFFF	-	保留	-	
0x4000_0000	0x4000_3FFF		General-purpose Timer 0/1/2 Control Register	4.10.4	APB1
0x4000_4000	0x4000_7FFF		Watchdog Control Register	4.12.4	
0x4000_8000	0x4000_BFFF		RTC Control Register	4.13.3	
0x4000_C000	0x4000_FFFF		Pulse Width Modulation (PWM) Control Register	4.8.3	
0x4001_0000	0x4001_BFFF	-	保留	-	
0x4001_C000	0x4001_FFFF		Analog Comparator(AC) Control Register	4.11.3	
0x4002_0000	0x47FF_FFFF	-	保留	-	APB2
0x4800_0000	0x4800_3FFF		UART 0/1 Control Register	0	
0x4800_4000	0x4800_7FFF		SPI Control Register	4.15.4	
0x4800_8000	0x4800_BFFF		I2C Control Register	4.14.3	
0x4800_C000	0x4801_FFFF		保留	-	
0x4802_0000	0x4802_3FFF		ADC Control Register	4.9.4	
0x4802_4000	0x4FFF_FFFF	-	保留	-	AHB
0x5000_0000	0x5000_FFFF		System Control Register	4.4.2	
0x5001_0000	0x5001_FFFF		Embedded Flash Control Register	4.5.5	
0x5002_0000	0x5002_FFFF		GPIO A/B/C/D Control Register	4.6.3	
0x5003_0000	0x6FFF_FFFF		保留	-	
0x6002_0000	0xDFFF_FFF F		保留	-	
0xE000_0000	0xE00F_FFFF		ARM® Cortex™-M0 System Timer (SysTick) Control Register	4.2.2	
0xE001_0000	0xFFFF_FFFF	-	保留	-	

4.2 ARM® CORTEX™-M0 内核

ARM Cortex-M0 处理器是目前最小的 32 位可配置的多级流水线 RISC 处理器。PT32U301 处理器有以下配置特性：

- 内置的嵌套向量中断控制器(NVIC):32 个外部中断
- 系统接口支持小端 (little-endian) 数据访问
- 内置系统定时器 – SysTick
- 支持断点调试
- 快速乘法器
- 支持 SWD 模拟调试。

以下描述处理器基本特性：

- CPU 系统时钟控制(SysTick)
- CPU 嵌套向量中断控制器(NVIC)
- CPU 系统控制

更多信息请参考以下：

- ARM Cortex™-M0 技术参考手册
- ARM v6-M 体系结构参考手册

4.2.1 CPU 系统时钟控制寄存器 (SYST)

Cortex™-M0 内核内置一个系统时钟包括一个集成的系统定时器 SysTick，提供一个简单的，24 位写清除，递减，具有灵活的控制机制。计数器可以作为一个实时操作系统（RTOS）滴答定时器或简单的计数器。

当系统定时器被使能，它开始从滴答系统的当前值寄存器(SYST_CVR)中的值进行递减计数直到 0，在下一个时钟周期到来时，重装载该值到滴答系统重装载值寄存器(SYST_RVR)中，然后在接下来的时钟周期中递减。一旦该计数器减到 0，COUNTFLAG 状态标志位将会被置位。通过读可以清除 COUNTFLAG 位。

SYST_CVR 的值在复位后为不确定的值。在使能这个功能之前，软件应该对该寄存器进行写操作来清除它为 0。这将确保定时器将会从 SYST_RVR 的值开始计数，而不是当它使能时的一个随意值。

如果 SYST_RVR 为 0，在重装载这个值之后，定时器将会保持当前的值为 0。这个机制能够被用来从定时器使能位立即禁止该功能。

4.2.2 SYST 寄存器映射

基地址: 0xE000_E000

偏移地址	名称	读/写	默认值	描述	参考頁
0x0010	CSR	R/W	0x0000_0000	系统定时器控制和状态寄存器控制	18
0x0014	RVR	R/W	-	SysTick 重载值寄存器	19
0x0018	CVR	R/W	-	SysTick 当前值寄存器	19

4.2.2.1 SYST_CSR -系统定时器控制和状态寄存器控制

SYST_CSR 使能 SysTick 特征.

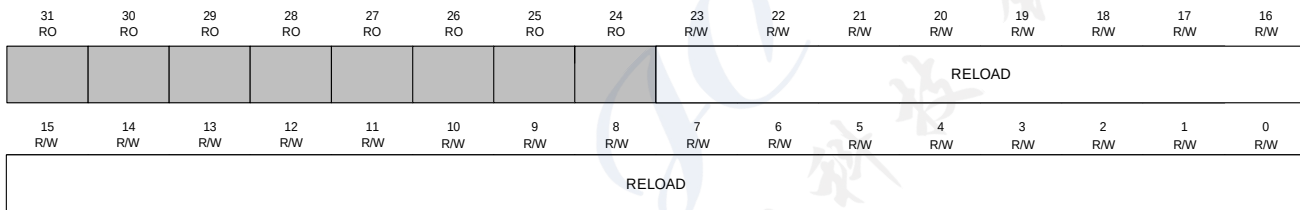
31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 R/W
															COUNT
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 R/W	1 R/W	0 R/W
													CLKSRC	TICKIE	ENABLE

偏移地址: 0x0010

位	名称	读/写	默认值	描述
31:17	保留	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
16	COUNT	R/W	0	定时器的状态 0: 系统定时器没有倒计到0 1: 系统定时器倒计到0
15:3	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
2	CLKSRC	R/W	0	选择定时器的时钟源 0: 时钟源为外部时钟. 1: 系统内部时钟.
1	TICKIE	R/W	0	定时器中断使能 0: 倒计到0不会导致定时器异常挂起.通过软件读COUNT位可以读取计数器是否递减到0。 1: 倒计到0将会导致定时器异常挂起。通过软件写操作清除滴答系统当前值寄存器，不会导致定时器挂起。
0	ENABLE	R/W	0	定时器计数使能 0: 计数禁止 1: 计数器可以在多次触发方式下操作。

4.2.2.2 SYST_RVR - SysTick 重载值寄存器

SYST_RVR 指定值加载到 SYST_CVR 寄存器中。

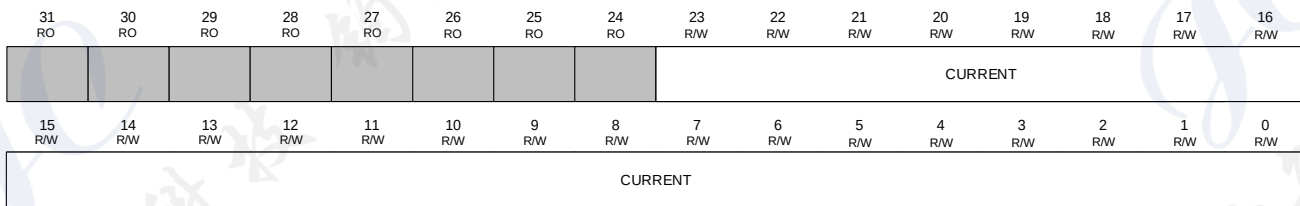


地址偏移: 0x0014

位	名称	读/写	默认值	描述
31:24	保留	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
23:0	RELOAD	R/W	0	重载值 当计数器减到0时候，装载值到定时器当前寄存器(SYST_CVR)。

4.2.2.3 SYST_CVR - SysTick 当前值寄存器

The SYST_CVR 包含系统时钟计数器当前值。



偏移地址: 0x0018

位	名称	读写	默认值	描述
31:24	保留	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
23:0	CURRENT	R/W	0	定时器时钟当前值

4.2.3 CPU 嵌模式中断向量控制器 (NVIC)

Cortex™-M0 提供中断控制器和外部异常中断模块，名称为“Nested Vectored Interrupt Controller (NVIC)”，这个是和处理器性能息息相关的并提供以下特性：

- 支持中断嵌套向量
- 处理器状态保存和恢复
- 降低中断延时
- 32 位可屏蔽中断
- 4 级优先级

NVIC 优先化和处理所有被支持的异常事件。所有异常事件在“Handler Mode”中被处理。该 NVIC 架构支持多达 32 个离散的中断请求 (IRQ [31:0]) 和 4 个优先级。所有的中断和大部分的系统异常都可以配置为不同的优先级。当一个中断发生时，NVIC 将会比较新的中断的优先级与当前运行的中断的优先级。如果新的中断的优先级高于当前运行的中断，新的中断处理程序将会覆盖当前的中断处理程序。

当一个中断被接收时，该中断服务路径(ISR)的起始地址将会从内存中的向量表中取出。不需要通过软件决定哪一个中断被接收和相关联的起始地址 ISR。当起始地址被取出，NVIC 将会自动保存处理器的状态包括将一些寄存器“PC, PSR, LR, R0~R3, R12”压入堆栈。在 ISR 的最后，该 NVIC 将会从堆栈中恢复被压栈的寄存器，并且恢复正常的执行。因此它将会采用较少的和确定的时间来处理中断请求。

NVIC 支持可以有效处理连续中断的“尾链中断”，因此减少在当前 ISR 结束时切换挂起 ISR 的延迟时间。NVIC 也支持“迟到”，它可以提高并发 ISR 的效率。在当前 ISR 开始执行之前，当一个更高优先级的中断请求发生时，NVIC 将会在没有延迟的情况下把优先权交给优先级更高的中断。因此它提高了实时性。

异常模块和系统中断映像

下表列出了异常模型。软件可以为这些异常事件设置四个级别的优先级，也可以为所有中断设置中断优先级。用户可配置的最高优先级为“0”，最低优先级为“3”。所有用户可配置中断的默认优先级为“0”。注意三个系统异常事件“复位”，“NMI”，“硬件错误”之后，优先级“0”被当作该系统的第四个优先级。

表 4.2-1: 异常模式

向量号.	异常中断名称	优先级
1	Reset	-3
2	NMI	-2
3	Hard Fault	-1
4-10	保留	保留
11	SCCall	可配置
12-13	保留	保留
14	PendSV	可配置
15	Sys Tick	可配置
16-47	Interrupt(IRQ[31:0])	可配置

表 4.2-2: 系统中断映像表

中断请求号	名称	描述
31	WAKEUP	CPU 唤醒中断
30	PVD	电压监测器中断
29-28	保留	-
27	保留	保留
26	保留	保留
25	SPI2/	SPI2 Interrupt
24	SPI1	SPI1 Interrupt
23	TIMER4	TIMER4 Interrupt
22	TIMER3	TIMER3 Interrupt
21	保留	-
20	I2C1	I2C1 Interrupt
19	保留	-
18	ADC	ADC 中断
17	COMP	模拟比较器中断
16	PWM_FAULT	PWM 错误中断
15	PWM2	PWM2中断
14	PWM1	PWM1中断
13	PWM0	PWM0 中断
12	TIMER2	TIMER2 中断
11	TIMER1	TIMER1 中断
10	TIMER0	TIMER0 中断
9	WDT	看门狗中断
8	RTC	RTC 中断
7	I2C0	I2C 中断
6	SPI0	SPI 中断
5	UART1	UART1 中断
4	UART0	UART0 中断
3	GPIO_D	GPIO D 埠中断
2	GPIO_C	GPIO C 埠中断
1	GPIO_B	GPIO B 埠中断
0	GPIO_A	GPIO A 埠中断

4.2.4 向量表

当一个中断被接收，处理器将会自动从内存中的一个向量表中读取该中断服务路劲的起始地址。对于 ARMv6-M，向量表的基地址为固定的 0x00000000。该向量表包含复位后堆栈指针的初始值，和所有异常处理程序的入口指针地址。在上一节的向量号定义了与异常处理程序项相关联的向量表中的顺序，如前一节所示。

表 4.2-3: 向量表

向量表长度	描述
0	SP_main – 主堆栈指针
向量数	当前使用异常中断个数

4.2.5 操作描述

通过写他们相对应的中断设置使能寄存器或中断清除使能寄存器，NVIC 中断能够被使能或禁止。这些寄存器使用一个写 1 使能和写 1 清除的方法，通过读这两个寄存器将会返回相对应中断的当前使能状态。当一个中断被禁止，中断请求将会导致该中断挂起，但是该中断不会被启动。如果一个中断是活动的，当它被禁止时，它仍然处于活动状态，直到通过复位被清除或异常返回。清除使能位防止相关的中断被启动。NVIC 中断可中断挂起/不挂起，使用一对互补的寄存器去使能/禁止那些中断，这些寄存器使用一个写 1 使能和写 1 清除的方法，通过读这两个寄存器将会返回相对应中断的当前使能状态。清除挂起寄存器对启动的中断的执行状态是无效。NVIC 中断优先级可以通过一个 32 位寄存器中的一个 8 位域来更新。与 NVIC 相关的通用寄存器都可以联通系统控制空间的区块，将在下一节介绍。

4.2.6 NVIC 寄存器映射

基地址 0xE000_E100

地址偏移	名称	读/写	默认值	描述	参考页
0x0000	NVIC_IUSER	R/W	0x0000_0000	IRQ使能控制寄存器	23
0x0080	NVIC_ICER	R/W	0x0000_0000	IRQ 清除中断控制寄存器	23
0x0100	NVIC_ISPR	R/W	0x0000_0000	IRQ 设置挂起控制寄存	24
0x0180	NVIC_ICPR	R/W	0x0000_0000	IRQ 清除挂起控制寄存器	24
0x0300	NVIC_IPR0	R/W	0x0000_0000	IRQ0 - IRQ3 优先级控制寄存器	25
0x0304	NVIC_IPR1	R/W	0x0000_0000	IRQ4 - IRQ7优先级控制寄存器	26
0x0308	NVIC_IPR2	R/W	0x0000_0000	IRQ8 - IRQ11优先级控制寄存器	27
0x030C	NVIC_IPR3	R/W	0x0000_0000	IRQ12 - IRQ15优先级控制寄存器	28
0x0310	NVIC_IPR4	R/W	0x0000_0000	IRQ16 - IRQ19优先级控制寄存器	29
0x0314	NVIC_IPR5	R/W	0x0000_0000	IRQ20 - IRQ23 优先级控制寄存器	30
0x0318	NVIC_IPR6	R/W	0x0000_0000	IRQ24 - IRQ27 优先级控制寄存器	31
0x031C	NVIC_IPR7	R/W	0x0000_0000	IRQ28 - IRQ31 优先级控制寄存器	32

4.2.6.1 NVIC_ISER - NVIC IRQ 使能控制寄存器

NVIC_ISER 中断使能寄存器和中断状态。

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

偏移地址: 0x0000

位	名称	读/写	默认值	描述
31:0	SETENA	R/W	0x0	中断使能 使能一个或者更多个中断. 中断数IRQ0 – IRQ31 之间 0:一个读操作表示此中断被禁止, 写操作无效 1:一个读操作表示此中断被使能, 写操作使能该中断

4.2.6.2 NVIC_ICER - NVIC IRQ 清除中断控制寄存器

The NVIC_ICER 禁止中断寄存器和中断状态.

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

偏移地址: 0x0080

位	名称	读/写	默认值	描述
31:0	CLRENA	R/W	0x0	中断禁止 关闭一个或者更多个中断. 中断数IRQ0 – IRQ31 之间 0:一个读操作表示此中断被禁止, 写操作无效 1:一个读操作表示此中断被使能, 写操作使能该中断

4.2.6.3 NVIC_ISPR - NVIC IRQ 设置挂起控制寄存器

ISPR 中断挂起状态寄存器，并可以查看到被挂起的中断位。

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

偏移地址: 0x0100

位	名称	读/写	默认值	描述
31:0	SETPEND	R/W	0x0	设置中断挂起 禁止一个或者更多个中断. 中断数IRQ0 – IRQ31 之间 0: 一个读操作表示此中断不被挂起，写操作无效 1: 一个读操作表示此中断被挂起，写操作挂起该中断

4.2.6.4 NVIC_ICPR - NVIC IRQ 清除挂起寄存器

ICPR 清除挂起中断状态寄存器，并可以查看到被挂起的中断位。

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

偏移地址: 0x0180

位	名称	读/写	默认值	描述
31:0	CLRPEND	R/W	0x0	设置中断挂起 关闭一个或者更多个中断. 中断数IRQ0 – IRQ31 之间 0: 一个读操作表示此中断不被挂起，写操作无效 1: 一个读操作表示此中断被挂起

4.2.6.5 NVIC_IPR0 - NVIC IRQ0 - IRQ3 优先级控制寄存器

IPR0 寄存器为每个中断提供一个 8 位优先级的位域，并且每个寄存器保留 4 个优先级位（IRQ[3:0]）。当设置每个优先级位域时，“0”总是被定义为最高优先级，“3”被定义为最低优先级。

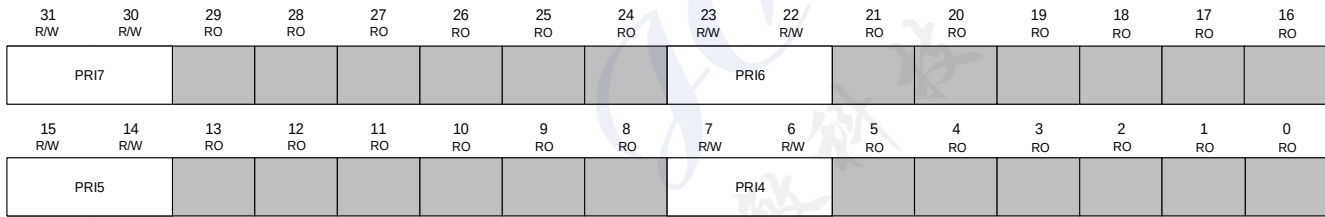
31 R/W	30 R/W	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 R/W	22 R/W	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
PRI3								PRI2							
15 R/W	14 R/W	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 R/W	6 R/W	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
PRI1								PRI1							

地址偏移: 0x0300

位	名称	读/写	默认值	描述
31:30	PRI3	R/W	0x0	IRQ3优先级
29:24	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
23:22	PRI2	R/W	0x0	IRQ2优先级
21:16	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:14	PRI1	R/W	0x0	IRQ1优先级
13:8	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
7:6	PRI1	R/W	0x0	IRQ0优先级
5:0	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。

4.2.6.6 NVIC_IPR1 - NVIC IRQ4 - IRQ7 优先级控制寄存器

IPR1 寄存器为每个中断提供一个 8 位优先级的位域，并且每个寄存器保留 4 个优先级位（IRQ[7:4]）。当设置每个优先级位域时，“0”总是被定义为最高优先级，“3”被定义为最低优先级。

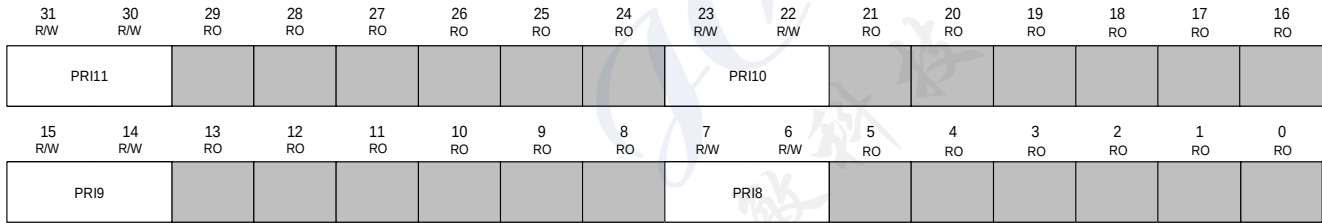


地址偏移: 0x0304

位	名称	读/写	默认值	描述
31:30	PRI7	R/W	0x0	IRQ7优先级
29:24	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
23:22	PRI6	R/W	0x0	IRQ6优先级
21:16	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:14	PRI5	R/W	0x0	IRQ5优先级
13:8	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
7:6	PRI4	R/W	0x0	IRQ4优先级
5:0	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。

4.2.6.7 NVIC_IPR2 - NVIC IRQ8 - IRQ11 优先级控制寄存器

IPR2 寄存器为每个中断提供一个 8 位优先级的位域，并且每个寄存器保留 4 个优先级位（IRQ[11:8]）。当设置每个优先级位域时，“0”总是被定义为最高优先级，“3”被定义为最低优先级。



地址偏移: 0x0308

位	名称	读/写	默认值	描述
31:30	PRI11	R/W	0x0	IRQ11优先级
29:24	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
23:22	PRI10	R/W	0x0	IRQ10优先级
21:16	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:14	PRI9	R/W	0x0	IRQ9优先级
13:8	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
7:6	PRI8	R/W	0x0	IRQ8优先级
5:0	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。

4.2.6.8 NVIC_IPR3 - NVIC IRQ12 - IRQ15 优先级控制寄存器

IPR3 寄存器为每个中断提供一个 8 位优先级的位域，并且每个寄存器保留 4 个优先级位（IRQ[15:12]）。当设置每个优先级位域时，“0”总是被定义为最高优先级，“3”被定义为最低优先级。

31 R/W	30 R/W	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 R/W	22 R/W	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
PRI15								PRI14							
15 R/W	14 R/W	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 R/W	6 R/W	5 R/W	4 RO	3 RO	2 RO	1 RO	0 RO
PRI13								PRI12							

地址偏移: 0x030C

位	名称	读/写	默认值	描述
31:30	PRI15	R/W	0x0	IRQ15优先级
29:24	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
23:22	PRI14	R/W	0x0	IRQ14优先级
21:16	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:14	PRI13	R/W	0x0	IRQ13优先级
13:8	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
7:6	PRI12	R/W	0x0	IRQ12优先级
5:0	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。

4.2.6.9 NVIC_IPR4 - NVIC IRQ16 - IRQ19 优先级控制寄存器

IPR4 寄存器为每个中断提供一个 8 位优先级的位域，并且每个寄存器保留 4 个优先级位（IRQ[19:16]）。当设置每个优先级位域时，“0”总是被定义为最高优先级，“3”被定义为最低优先级。

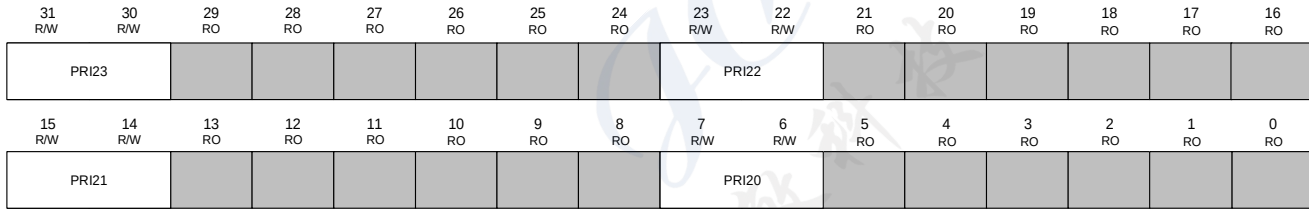
31 R/W	30 R/W	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 R/W	22 R/W	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
PRI19								PRI18							
15 R/W	14 R/W	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 R/W	6 R/W	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
PRI17								PRI16							

地址偏移: 0x0310

位	名称	读/写	默认值	描述
31:30	PRI19	R/W	0x0	IRQ19优先级
29:24	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
23:22	PRI18	R/W	0x0	IRQ18优先级
21:16	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:14	PRI17	R/W	0x0	IRQ17优先级
13:8	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
7:6	PRI16	R/W	0x0	IRQ16优先级
5:0	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。

4.2.6.10 NVIC_IPR5 - NVIC IRQ20 - IRQ23 优先级控制寄存器

IPR5 寄存器为每个中断提供一个 8 位优先级的位域，并且每个寄存器保留 4 个优先级位（IRQ[23:20]）。当设置每个优先级位域时，“0”总是被定义为最高优先级，“3”被定义为最低优先级。



地址偏移: 0x0314

位	名称	读/写	默认值	描述
31:30	PRI23	R/W	0x0	IRQ23优先级
29:24	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
23:22	PRI22	R/W	0x0	IRQ22优先级
21:16	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:14	PRI21	R/W	0x0	IRQ21优先级
13:8	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
7:6	PRI20	R/W	0x0	IRQ20优先级
5:0	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。

4.2.6.11 NVIC_IPR6 - NVIC IRQ24 - IRQ27 优先级控制寄存器

IPR6 寄存器为每个中断提供一个 8 位优先级的位域，并且每个寄存器保留 4 个优先级位（IRQ[27:24]）。当设置每个优先级位域时，“0”总是被定义为最高优先级，“3”被定义为最低优先级。

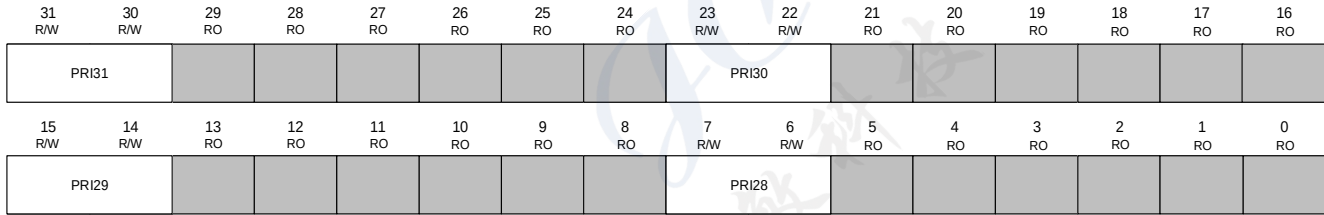
31 R/W	30 R/W	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 R/W	22 R/W	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
PRI27								PRI26							
15 R/W	14 R/W	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 R/W	6 R/W	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
PRI25								PRI24							

地址偏移: 0x0318

位	名称	读/写	默认值	描述
31:30	PRI27	R/W	0x0	IRQ27优先级
29:24	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
23:22	PRI26	R/W	0x0	IRQ26优先级
21:16	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:14	PRI25	R/W	0x0	IRQ25优先级
13:8	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
7:6	PRI24	R/W	0x0	IRQ24优先级
5:0	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。

4.2.6.12 NVIC_IPR7 - NVIC IRQ28 - IRQ31 优先级控制寄存器

IPR7 寄存器为每个中断提供一个 8 位优先级的位域，并且每个寄存器保留 4 个优先级位（IRQ[31:28]）。当设置每个优先级位域时，“0”总是被定义为最高优先级，“3”被定义为最低优先级。



地址偏移: 0x031C

位	名称	读/写	默认值	描述
31:30	PRI31	R/W	0x0	IRQ31优先级
29:24	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
23:22	PRI30	R/W	0x0	IRQ30优先级
21:16	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:14	PRI29	R/W	0x0	IRQ29优先级
13:8	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
7:6	PRI28	R/W	0x0	IRQ28优先级
5:0	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。

4.3 CPU 系统控制

Cortex™-M0 状态和工作模式控制可通过 CPU 系统控制寄存器管理。包括 CPUID，Cortex™-M0 中断优先级和 Cortex™-M0 电源管理都可以通过这些系统控制寄存器来控制。

4.3.1 CPU 系统控制寄存器映射

基地址 0xE000_ED00

地址偏移	名称	读/写	默认值	描述	参考页
0x0000	SYS_CPUID	R/W	0x410C_C200	CPU 系统ID 寄存器	34
0x0004	SYS_ICSR	R/W	0x0000_0000	系统中断控制和状态寄存器	34
0x000C	SYS_AIRCR	R/W	0xFA05_0000	系统应用中断和状态控制寄存	36
0x0010	SYS_SCR	R/W	0x0000_0000	系统控制寄存器	37
0x001C	SYS_SHPR2	R/W	0x0000_0000	系统中断优先级控制寄存器2	38
0x0020	SYS_SHPR3	R/W	0x0000_0000	系统中断优先级控制寄存器3	38

4.3.1.1 SYS_CPUID – CPU ID 寄存器

这个寄存器提供 理器的 ID 。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 R/W	18 R/W	17 R/W	16 R/W
IMPC												PART			
15 R/W	14 R/W	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 R/W	6 R/W	5 RO	4 RO	3 R/W	2 R/W	1 R/W	0 R/W
PARTNO												REV			

偏移地址: 0x0000

位	名称	读/写	默认值	描述
31:24	IMPC	RO	0x41	ARM 执行码 ARM = 0x41
23:20	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
19:16	PART	R/W	0xC	处理器构架
15:4	PARTNO	RO	0xC20	处理器号
3:0	REV	R/W	0x0	修订版本号

4.3.1.2 SYS_ICSR -中断控制和状态寄存器

这个寄存器提供中断状态信息和控制。

31 RO	30 RO	29 RO	28 R/W	27 WO	26 R/W	25 WO	24 RO	23	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
NMISP			PENDSV	PENDSV	PENDST	PENDSTC		ISRPRE	SRPEND						VTPEND
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
VTPEND												VTACT			

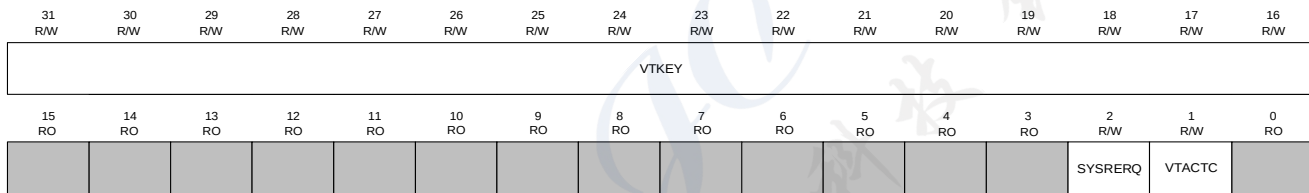
偏移地址: 0x0004

位	名称	读/写	默认值	描述
31	NMISP	RO	0	NMI (不可屏蔽中断) 设置挂起 0: 一个读操作表示此NMI异常中断不被挂起，写操作无效 1: 一个读操作表示此NMI异常中断被挂起，写会改变NMI异常状态为挂起。 因为NMI有最高中断优先权，处理器会入NMI异常处理程序一旦检测到此位被致能。当进入NMI异常处理程序，清除位。此动作保证在下一次读取这个位时，如果此位显示被致能的话，表示在处理器在进入异常处理程序时又有一NMI信号再生成。
30:29	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
28	PENDSV	R/W	0	PendSV 设置挂起 0: 一个读操作表示此PendSV异常中断不被挂起，写操作无效 1: 一个读操作表示此PendSV异常中断被挂起

27	PENDSV	WO	0	PendSV 清除挂起 0: 无效 1: 移除这个未挂起的 PendSV异常中断
26	PENDST	R/W	0	SysTick 异常设置挂起位 0: 一个读操作表示此SysTick异常中断不被挂起，写操作无效 1: 一个读操作表示此SysTick异常中断被挂起
25	PENDSTC	W	0	SysTick清除挂起 0: 无效 1: 移除这个未挂起的 SysTick异常中断
24	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
23	ISRPRE	RO	0	硬件调试中断 0: 硬件调试停止但不产生中断 1: 当外部调试退出时候产生异常中断
22	SRPEND	RO	0	中断挂起标识, NMI 异常和错误 0: 中断没被挂起 1: 中断挂起
21:18	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
17:12	VTPEND	RO	0x0	中断挂起向量数 0: 无异常挂起. 非零: 最高优先级异常挂起数量.
11:6	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
5:0	VTACT	RO	0x0	中断挂起向量数 Bit0-Bit5这几位包含异常中断数量。. 0: 线程模式. 0-64: 当前产生的异常中断数

4.3.1.3 SYS_AIRCR - 用户中断和复位控制寄存器

这个寄存器可以 置或者读取中断控制寄存器的数据。



偏移地址: 0x000C

位	名称	读/写	默认值	描述
31:16	VTKEY	R/W	0xFA05	寄存器访问密钥 对该寄存器写时，VTKEY域需要设置为0x05FA，否则写操作将被忽略。 VTKEY域用来防止意外写入该寄存器 从设置系统或清除异常状态。
15:3	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
2	SYSRERQ	R/W	0	系统复位请求 0:没有复位请求 1: 复位
1	VTACTC	R/W	0	清除 NMI / Fault 用于调试使用。当写寄存器,用户必须写0到这一点,否则行为是不可预测的。
0	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。

4.3.1.4 SYS_SCR -系统控制寄存

SCR 寄存器提供进入低功耗和退出低功耗状态。

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

偏移地址: 0x0010

位	名称	读/写	默认值	描述
31:5	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
4	EVONPEND	RO	0x0	发送事件挂起 0: 只有启用中断或事件才可以唤醒处理器。 1:使能全部中断,产生事件和禁止中断可以唤醒处理器。 当一个事件或中断进入挂起状态，事件信号从WFE中唤醒处理器。如果处理器不等待事件，该事件被注册并且影响影响下一个WFE。 处理器也唤醒一个SEV指令或一个外部事件的执行。
3	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
2	SLPDEEP	R/W	0	深度休眠 控制处理器使用深度休眠或者休眠作为低功率模式: 0: 休眠模式. 1: 深度休眠模式.
1	SLPONE	R/W	0	Sleep-On-Exit 使能 0: 当退出休眠时候返回正常工作模式 1: ISR当从深度休眠或者休眠返回线程模式 设置此位1使一个中断函数为的应用程序,以避免从休眠模式返回后跳入一个空程序
0	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。

4.3.1.5 SYS_SHPR2 - 系统处理优先级寄存器 2

31 R/W	30 R/W	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
PRISH11															
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO

偏移地址: 0x001C

位	名称	读/写	默认值	描述
31:30	PRISH11	R/W	0	系统处理优先级11 – SVCall “0”总是被定义为最高优先级，“3”被定义为最低优先级。
29:0	保留	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。

4.3.1.6 SYS_SHPR3 - 系统处理优先级寄存器 3

31 R/W	30 R/W	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 R/W	22 R/W	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
PRISH15								PRISH14							
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO

偏移地址: 0x0020

位	名称	读/写	默认值	描述
31:30	PRISH15	R/W	0x0	系统处理优先级15 – SysTick “0”总是被定义为最高优先级，“3”被定义为最低优先级。
29:24	保留	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
23:22	PRISH14	R/W	0x0	系统处理优先级 14 – PendSV “0”总是被定义为最高优先级，“3”被定义为最低优先级。
21:0	保留	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。

4.4 系统控制(SC)

系统控制配置了设备的整体操作以及提供设备相关信息。PT32U301 中的系统控制具备以下的特点：

- 设备识别
- 内存映射
- ICE 保护
- 系统管理
- 系统时钟校准
- 时钟控制
- 功耗控制

4.4.1 功能描述

4.4.1.1 设备识别

设备识别跟 **SC_PID0** 和 **SC_PID1** 寄存器有关。它们提供芯片上的相关软件信息，例如 Flash / SRAM 的存储空间，质量等级，封装以及产品的版本。此外，**SC_PID1** 寄存器内容中的封装类型 **PKG**、温度测试范围 **TEMP** 和质量等级 **QUAL** 都是由编程来设置，存放在嵌入式闪存信息(0x7F4 ~ 0x7F7)中。嵌入式闪存由 Mask ROM 启动。未来的计划中，我们将提供让用户可以通过 **SC_UID0** 和 **SC_UID1** 寄存器额外添加芯片的识别信息。

4.4.1.2 ICE 保护

用户区的闪存信息能够被保护，防止被不受信任的代码读取。在嵌入式闪存信息块中，通过指定的值分配到一个地址，将读取保护激活；上电复位(POR)后，保护也将被激活。用户可以通过寄存器(**SC_ICE**)来检查保护状态。

4.4.1.3 启动配置

PT32U301 支持 4 种不同启动模式。通过设定软件配置的内存重映像寄存器(**SC_REMAP**)的 **REMAP** 位，便可设定复位后的启动模式。

表 4.4-1: 启动模式选择

启动模式选择	启动模式*
REMAP 位	
0x0	信息闪存存储器被选为启动区域
0x1	SRAM被选为启动区域
0x2	主闪存存储器被选为启动区域
0x3	ROM被选为启动区域

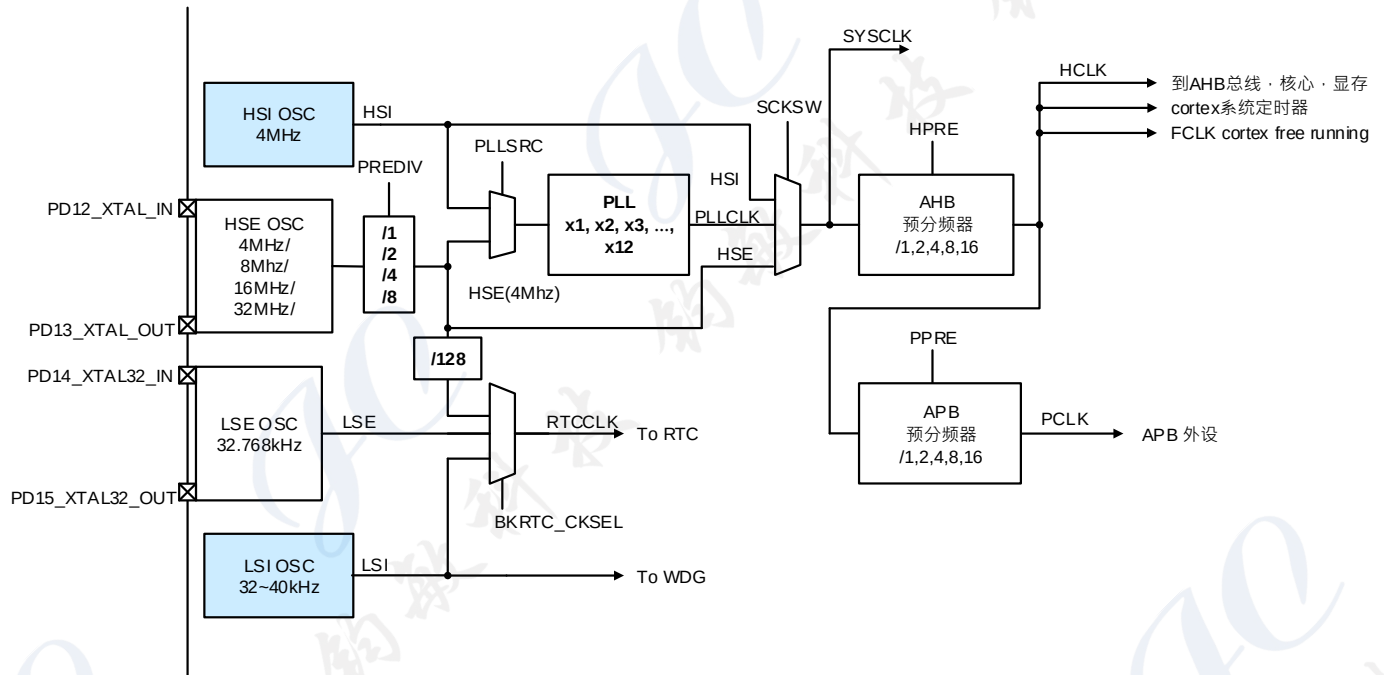
*:启动模式的选择会在上电复位或系统复位后被锁存

根据选定的启动模式，主闪存存储器、信息闪存存储器、ROM 或 SRAM 可以按照以下方式访问：

- 从信息闪存存储器启动: 信息闪存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(0x1FFF F000)访问。
- 从 SRAM 启动: SRAM 被映射到启动空间(0x0000 0000)，但仍然能够在它原有的地址(0x2000 0000)访问。
- 从主闪存存储器启动: 主闪存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(0x0800 0000)访问。
- 从 ROM 启动: 一般正常启动模式，系统只会从原有地址(0x0000 0000)访问。

4.4.1.4 时钟系统

图 4.4-1: 时钟树



4.4.1.5 时钟源

芯片有多个时钟源以供使用:

- HSI 时钟
- 高速内部时钟(HSI), RC 振荡器, 频率为 4 MHz, 可直接系统时钟来源或作为锁相环(PLL)输入的来源。
- HSE 时钟
- 高速外部时钟(HSE), 接外部时钟源。可保持运行, 用于锁相环(PLL)输入或实时时钟(RTC)的时钟源。
- LSI 时钟
- 低速内部时钟(LSI), RC 振荡器, 频率为 30 kHz。
- LSE 时钟
- 低速外部时钟(LSE), 接频率为 32.768 kHz 的石英晶体。高精度时钟源, 可作为时钟/日历或者其他定时功能的实时时钟。

4.4.1.6 时钟分布

PT32U301 的时钟系统包括:

系统时钟- SYSCCLK

所有振荡器可以用作系统时钟 (SYSCCLK) 的来源。系统时钟可以选择不同的时钟源。通过设定时钟配置寄存器 (SC_CK_CONF) 内的 SCKSW 位, 便可切换系统时钟的时钟源。可被配置的时钟源包括:

- 4 MHz HSI
- PLL 時鐘
- 4 MHz HSE

以上时钟源都可以用作驱动 SYSCCLK 的来源。在系统启动或重置后, SYSCCLK 的预设时钟来源为 HSI。请留意当当时钟源被直接或通过 PLL 间接作为系统时钟时, 它将不能被停止。只有当目标时钟源准备就绪了(经过启动稳定阶段的延迟或 PLL 稳定)。使用者可通过时钟源状态寄存器 – SC_CK_STAT 的时钟源状态表示位(就绪标志)得知时钟源是否已稳定并可被使用。唯有当就绪标志被致能时, 从一个时钟源到另一个时钟源的切换才会发生。在被选择时钟源没有就绪时, 系统时钟的切换不会发生。直至目标时钟源就绪, 才发生切换。

为安全防范不建议一直关闭 HSI 时钟, 以防外部时钟源出现故障。

实时计数器 - RTCCLK

有三个时钟源可被配置去驱动 RTCCLK。通过设定 RTC 控制寄存器(RTC_CTRL)内的 CLKSEL 位元, 便可切换实时计数器的时钟源。

- 4 MHz HSE 128 分频
- 32.768kHz LSE
- LSI

HCLK, PCLK

AHB 总线的频率(HCLK)可通过预分频器(Prescaler)去分频 SYSCCLK, 用以配置时钟给 CPU, 内存及任何配置到 AHB 总线的外围。APB 总线的频率(PCLK)亦通过预分频器去分频 HCLK, 用以配置时钟给任何配置到 APB 总线的外围。

4.4.1.7 电源

电压调节器

电压调节器嵌入在 PT32U301 中，给内部数字电源供电。设备需要 2V-3.6 V 输入电压。电压调节器总是在复位后使能。它可以在三个不同的模式下工作，具体取决于应用模式。

- 运行模式
稳压电源给内部数字模块(内核，内存核数字外设)提供充足的电源。
- 停止模式
稳压电源给数字域提供低功耗电源，保存寄存器和 SRAM 的内容。
- 待机模式

稳压电源关闭。寄存器和 SRAM 的内容会丢失除了待机电路和备份域。

复位控制

该装置具有三种方式来监测和复位器件：

电压复位

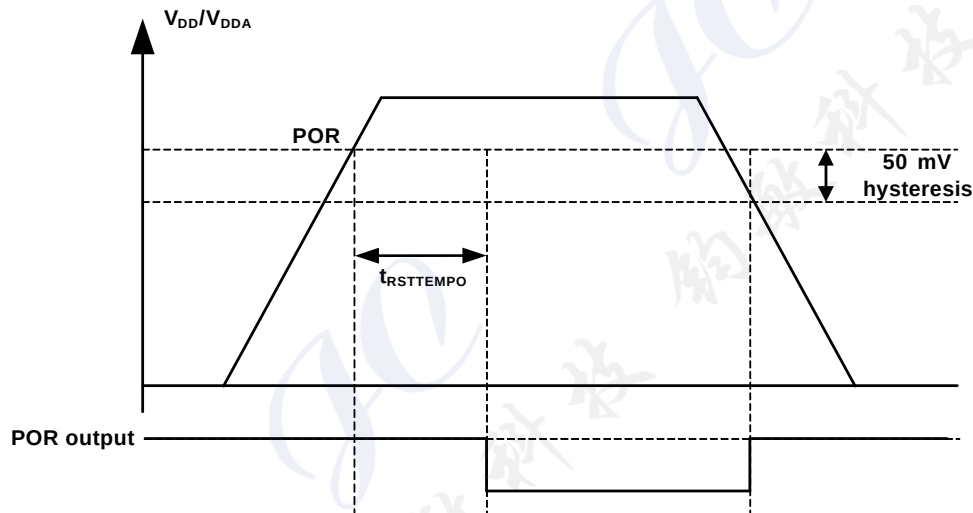
电压复位会发生在当电源降至低于装置运行所需的电压时。此装置有三种方式检测和复位设备：

- 上电复位(POR)
- 掉电复位 (PDR)
- 可编程电压触发器(PVD)

上电复位 (POR) / 掉电复位(PDR)

PT32U301 整合了 POR/PDR 电路，用来监测电力供应电压 VDD33 并确保 2V 阈值以上操作的正确性。在重置模式下 VDD/VDDA 是低于指定的阈值，VPOR/PDR, 无需外部复位电路。更多关于 POR/PDR 阈值的数据，参考上电复位特性。

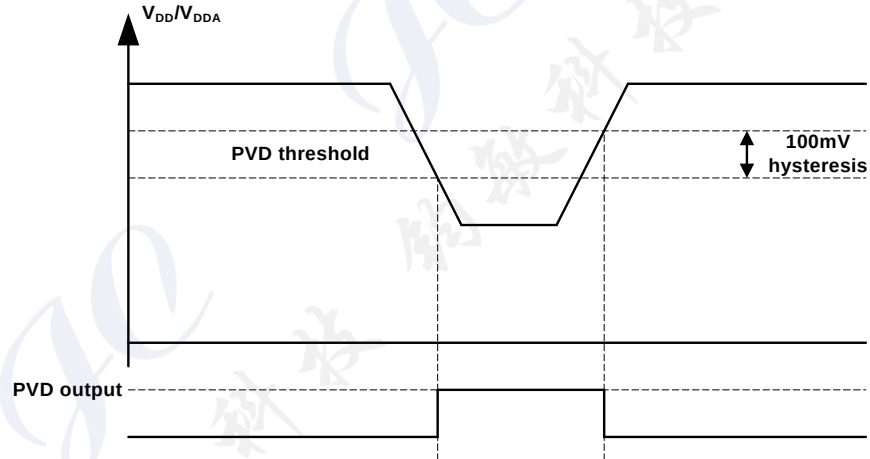
图 4.4-2: 重启和复位波形



可编程电压触发器(PVD)

PVD 可通过将 PVD 检测寄存器(SC_PVD_DET)中的 *VOLT* 位选择阈值与 *VDD/VDDA* 进行比较来检测 *VDD/VDDA*。PVD 是通过设置 SC_PVD_DET 的 *DETEN* 位来启用的。

图 4.4-3: 可编程电压触发器(PVD)临界值



低功耗模式

默认情况下，微处理器在复位后处于运行模式。当 CPU 不需要一直保持运行时，一些低功耗模式都可以节省电量，例如，当等待外部事件时。用户可选择的模式，给出了低功耗、短的启动时间和可用唤醒源之间最好的折衷办法。

该芯片具有三种低功耗模式：

- 睡眠模式 (CPU 时钟关闭，所有 Cortex®-M0 外围设备，包括像 NVIC、SysTick 等保持运行)
- 停机模式 (所有时钟停止工作)
- 待机模式 (1.8V 内电源关闭)

此外，还可以通过以下任意途径降低功耗：

- 减慢时钟系统
- 当 APB 和 AHB 外设未被使用时门控其时钟
- 闪存进入休眠模式

减慢系统时钟

在运行模式下，系统时钟 (SYSCLK，HCLK，PCLK) 的速度可以通过编程预分频寄存器来减缓。这些预分频器也可以用于进入睡眠模式前使外设减速。

外设时钟门控

在运行模式下，单个外设和内存的 AHB 时钟和 APB 时钟可以在任何时候被停止用来降低功耗。要进一步降低睡眠模式的功耗，外设时钟需在执行 WFI 前被禁用。

使能闪存睡眠模式

闪存（主块和信息块）可以进入 *sleepmode*，以减少在所有低功耗模式下的功耗。用户写 1 到闪存控制器命令寄存器—FC_CMD 位的 *SLEEP* 位以启用闪存的睡眠模式。一旦该位设置，当装置进入低功耗模式时，闪存将尽快进入 *sleepmode*。

睡眠模式

进入睡眠模式

通过执行 WFI 进入休眠模式（等待中断）指令。根据在 Cortex®-M0 系统控制寄存器中的 SLPONEXIT 位，有两个选项可供选择睡眠模式进入机制。

- Sleep-now: 如果 SLPONEXIT 位清零，MCU 只要执行 WFI 指令进入休眠模式。
- Sleep-on-exit: 如果 SLPONEXIT 位清零，单片机进入休眠模式一旦它退出最低优先级的 ISR。

在休眠模式下，所有的 I/O 引脚保持在运行模式下的相同状态。

退出睡眠模式

如果 WFI 指令用于进入睡眠模式，任意一个被嵌套向量中断控制器(NVIC)响应的外设中断都能将系统从睡眠模式唤醒。

停止模式

进入停止模式

停止模式是在 Cortex-M0 深睡眠模式下发展出的模式。在此模式下，电压调节器会被配置成低功耗模式。电压调节器可以配置在正常或低功率模式下運作。在停止模式下，所有的时钟可以在 CPU 进入深度休眠模式之前被关闭。该系统可以通过关闭 PLL/ HSI/ HSE/ ADC 在此模式下保存最大功率。并通过切换时钟源 LSI 或 LSE 降低系统时钟速度。时钟在切换到较低的频率後，该系统还可以设置 LDO 成 LDO 低功耗模式。无需切换 LDOLP 或 LDOOFF 位，停止模式与睡眠模式几乎相同。用户必须记住停止之前，关掉不必要的外围设备的 CPU。

进入停止模式

为了进一步降低停止模式下的功耗，内部稳压器可置于低功耗模式。这是由 SC_BKSLP_CTRL 的 LDOLP 位配置。该 ADC 會在停止模式下耗电，除非他们是在进入该模式前被禁用。

退出停止模式

当电压调节器处于低功耗模式下，当系统从停止模式退出时，将会有一段额外的启动延时。通过停止模式期间保持内部稳压器接通，耗电量较高但是启动时间缩短。

待机模式

进入待机模式

系统进入待机模式后会达到最低功耗。这个模式整合了 Cortex-M0 的深睡眠模式并将电压调节器关闭。稳压器电源域因此被关闭。整个 1.8V 供电区域会被断电。PLL、HSI、HSE 振荡器都会被断电。除了备份域的寄存器和待机电路会保持供电外，SRAM 及寄存器内容会丢失。

进入待机模式

为了进一步降低停止模式下的功耗，内部稳压器可以关闭。这是由 SC_BKSLP_CTRL 的 LDOOFF 位去配置。

退出待机模式

当一个 NRST 引脚上的上升或下降沿，該脚位下降沿触发或一個 RTC 唤醒事件，會喚醒 PT32U301。从待机模式唤醒后，执行程序重新启动的方式與复位后的执行程序相同。在备份 RTC 睡眠控制器寄存器（SC_BK_STATUS）的 WKUPFLAG 状态标志表明该 MCU 是处于待机模式。

4.4.2 系统控制寄存器映射

基址: 0x5000_0000

偏移地址	标识	类型	复位值	描述	参考頁
0x0000	SC_PID0	RO	0x0188_0F5C	产品ID0 号寄存器	46
0x0004	SC_PID1	R/W	0x0000_00E0	产品ID1 号寄存器	47
0x0008	SC_UID0	R/W	0x0000_0000	用户定义域ID0	48
0x000C	SC_UID1	R/W	0x0000_0000	用户定义域ID1	48
0x0010	SC_REMAP	R/W	0x0000_0F06	软件配置内部重映像	49
0x0014	SC_ICE	R/W	0x0000_0000	ICE保护	50
0x0018	SC_STCALIB	R/W	0x0100_0148	系统嘀嗒时钟校准值寄存器	50
0x001C	SC_NVM_ACR	R/W	0x0000_0040	NVM (FLASH)访问控制寄存器	51
0x0020	SC_GCK_APB	R/W	0x0000_0000	APB外设门控时钟使能寄存器	52
0x0024	SC_GCK_AHB	R/W	0x0000_0000	AHB外设门控时钟使能寄存器	53
0x0028	SC_RST_APB	R/W	0x0000_0000	APB外设复位请求寄存器	54
0x002C	SC_RST_AHB	R/W	0x0000_0000	AHB外设复位请求寄存器	55
0x0030	SC_CK_CTRL	R/W	0x0000_0001	时钟控制寄存器	56
0x0034	SC_CK_CONF	R/W	0x0000_000B	时钟配置寄存器	57
0x003C	SC_CK_STAT	RO	0x0000_0000	时钟状态寄存器	59
0x0040	SC_BK_CTRL	R/W	0x0000_0000	备份控制寄存器	60
0x0048	SC_PWR_CTRL	R/W	0x0000_0000	电源控制寄存器	60
0x004C	SC_PVD_DET	R/W	0x0000_000E	PVD检测寄存器	61
0x0050	SC_SLP_APB	R/W	0x0000_0000	睡眠模式下APB外设门控时钟使能寄存器	62
0x0054	SC_SLP_AHB	R/W	0x0000_0000	睡眠模式下AHB外设门控时钟使能寄存器	63
0x0058	SC_DSLP_APB	R/W	0x0000_0000	深度睡眠模式下APB外设门控时钟使能寄存器	64
0x005C	SC_DSLP_AHB	R/W	0x0000_0000	深度睡眠模式下AHB外设门控时钟使能寄存器	65
0x006C	SC_OSC_TRIM	R/W	0x0080_0000	环OSC 硬件校正寄存器	66
0x0080	SC_BKRTC_CTRL	R/W	0x0000_0000	备份RTC控制器寄存器	67
0x0084	SC_BK_LPCTRL	R/W	0x00C0_0000	备份RTC睡眠控制器寄存器	68
0x0088	SC_BK_STATUS	R/W	0x0000_0000	状态备份寄存器	69
0x0090	SC_BK_REG0	R/W	0x0000_0000	备份0寄存器	69
0x0094	SC_BK_REG1	R/W	0x0000_0000	备份1寄存器	69
0x0098	SC_BK_REG2	R/W	0x0000_0000	备份2寄存器	69
0x009C	SC_BK_REG3	R/W	0x0000_0000	备份3寄存器	69
0x00A0	SC_BK_REG4	R/W	0x0000_0000	备份4寄存器	69
0x00A4	SC_BK_REG5	R/W	0x0000_0000	备份5寄存器	69
0x00A8	SC_BK_REG6	R/W	0x0000_0000	备份6寄存器	69
0x00AC	SC_BK_REG7	R/W	0x0000_0000	备份7寄存器	69

4.4.2.1 SC_PID0 - 产品 ID0 号寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
PFTYPE			PRDTYPE			NVMTYPE		EFTYPE		EFSZ			SRAMSZ		
15 R/W	14 R/W	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
YEAR								WEEK				VERSION			

偏移地址: 0x0000

位	名称	读/写	默认值	描述
31:29	PFTYPE	RO	0x0	平台类型 0x0 : Cortex-M0 MCU 平台 0x1 : Cortex-M3 MCU 平台 0x2 : Cortex-M4 MCU 平台 0x3 : ASIC 0x4 : 其他
28:25	PRDTYPE	RO	0x2	产品类型 0x0 : 通用 MCU 0x1 : BLDC 应用 0x2 : USB 应用 0xE : EMC 应用 0xF : TSC 应用
24:23	NVMTYPE	RO	0x3	非易失性存储器类型 0x0 : OTP 0x1 : MTP 0x2 : EEPROM 0x3 : 嵌入式 Flash
22:21	EFTYPE	RO	0x0	嵌入式闪存类型 0: 单个嵌入式Flash存储 1: 双嵌入式Flash存储
20:18	EFSZ	RO	0x3	嵌入式闪存大小 0x0 : 8K 0x1 : 16K 0x2 : 32K 0x3 : 64K 字节 0x4 : 128 K字节(保留)
17:16	SRAMSZ	RO	0x1	SRAM 大小 4X(SRAM_SZ+1) K 字节
15:8	YEAR	RO	15	发布年份
7:2	WEEK	RO	52	发布周 e.g. 1-52 一年
1:0	VERSION	RO	0x0	主要版本 0x0: 版本A (初始版本) 0x1: 版本B (第一次修订版) 0x2: 版本C (第二次修订版) 0x3: 版本D (最新版本)

4.4.2.2 SC_PID1 - 产品 ID1 号寄存器

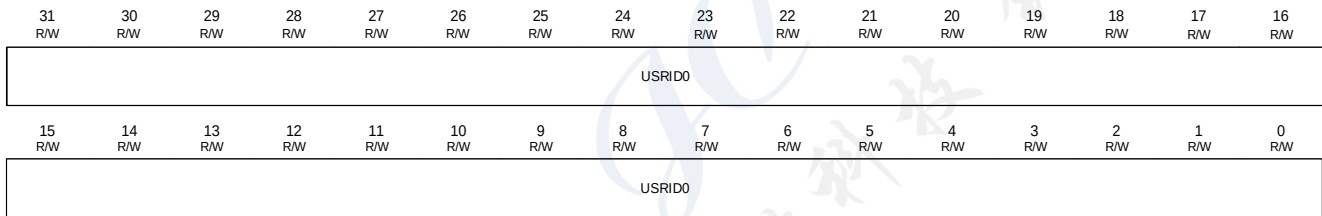
31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
PFTYPE			RCODE												
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
									PKG		BID		TEMP		QUAL

偏移地址: 0x0004

位	名称	类型	复位值	描述
31	FPT	W	0	Flash保护位 0: Flash数据被保护 1: Flash数据未被保护
30:29	保留	RO	0x0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
28:27	RCODE	R	0x0	Rom代码版本 0x0: 版本A (初始版本) 0x1: 版本B (第一次修订版) 0x2: 版本C (第二次修订版) 0x3: 版本D (最新版本)
26:8	保留	RO	0x0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
7:5	PKG	R	0x7	封装类型 0x0: 8引脚 SOP 封装 0x1: 16引脚 SSOP 封装 0x2: 20引脚 SSOP 封装 0x3: 24引脚 SSOP 封装 0x4: 32引脚 QFN 封装 0x5: 48引脚 LQFP 封装 0x6: 64引脚 LQFP 封装 0x7: 其他特殊封装类型
4	BID	R	0	边框线类型 0: 金 1: 铜
3:2	TEMP	R	0x0	温度范围 0x0: 商业温度范围 (0°C to 70°C) 0x1: 工业温度范围 (-40°C to 85°C) 0x2: 扩展温度范围 (-40°C to 125°C)
1:0	QUAL	R	0x0	质量状态 0x0: 工程样品 (不合格) 0x1: 试生产 (不合格) 0x2: 完全合格(仅 FT) 0x3: 完全合格(CP/FT)

4.4.2.3 SC_UID0 - 用户定义域 ID0 寄存器

The SC_UID0 能让用户能进一步定义产品。

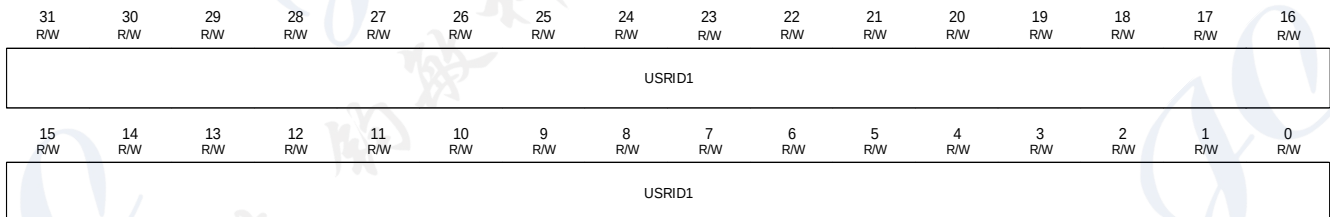


偏移地址: 0x0008

位	名称	类型	复位值	描述
31:0	USRID0	R/W	0x0	用户定义域ID0

4.4.2.4 SC_UID1 - 用户定义域 ID1 寄存器

The SC_UID0 能让用户能进一步定义产品。



偏移地址: 0x000C

位	名称	类型	复位值	描述
31:0	USRID1	R/W	0x0	用户定义域ID1

4.4.2.5 SC_REMAP - 软件配置的内存重映像

寄存器可用软件改变启动过程的位置，如 Eflash，SRAM。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 R/W	1 R/W	0 R/W
				EFSZ		HWREMAP				SWREMAP			REMAP		REMAPE

偏移地址: 0x0010

位	名称	类型	复位值	描述
31:12	保留	RO	0x0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
11:10	EFSZ	RO	0x3	最大嵌入式闪存大小 0x0 : 16K 0x1 : 32K 0x3 : 64K
9:8	HWREMAP	RO	0x3	外部硬件重映射状态 0x2 : Eflash 0x3 : Mask ROM
7:6	保留	RO	0x0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
5:4	SWREMAP	RO	0x3	内部重映射状态选择 0x0 : Eflash信息 0x1 : SRAM 0x2 : Eflash Memory 0x3 : Mask ROM
3	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
2:1	REMAP	R/W	0x3	内部重映射选择 0x0 : Eflash 信息 0x1 : SRAM 0x2 : Eflash Mem 0x3 : Mask ROM
0	REMAPE	R/W	0	内部重映射使能 1:进入重映射流 此位为自动清除

4.4.2.6 SC_ICE - ICE 保护

该寄存器将被激活，如果 Flash 信息 0x7F8：0 JTAG 保护开关被禁用。

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

偏移地址: 0x0014

位	名称	类型	复位值	描述
31:1	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
0	ICE	R/W	0	保护嵌入式和关闭JTAG功能 0: 禁用JTAG保护 1: 启用JTAG保护（JTAG禁用）

4.4.2.7 SC_STCALIB - 系统滴答定时器校准值寄存器

The SC_STCALIB 寄存器显示系统滴答校准属性

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

偏移地址: 0x0018

位	名称	类型	复位值	描述
31:26	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
25	NOREF	R/W	0	系统滴答定时器参考 0: 需要外部参考时钟 1: 表示系统滴答值只使用内核时钟去计数
24	SKEW	R/W	1	1: TENMS位域不准确
23:0	TENmS	R/W	0x148	十毫秒校准值。此为单片机设计默认值。

运行模式中，单个外设和存储器的 HCLK 和 PCLK 可以在任意时间被关闭以降低功耗。为了进一步降低睡眠模式下的功耗，可以在执行 WFI 之前禁用外设模块。

4.4.2.8 SC_NVMA CR - NVM (FLASH) 访问控制寄存器

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

偏移地址: 0x001C

位	名称	类型	复位值	描述								
31:16	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。								
15:4	ONEUS	R/W	0x4	时钟分频器将系统时钟划分成1us的脉冲时钟，例如，如果系统时钟是48MHz(20.8ns)，那么你需要将寄存器的值设置为十进制的48。								
3:1	WCYCLE	R/W	0x0	在等待周期内从嵌入式闪存读取数据。								
				<table><tr><th>HCLK 频率(<i>f</i>)</th><th>WCYCLE 值</th></tr><tr><td><i>f</i> ≤ 24MHz</td><td>0x0</td></tr><tr><td>24MHz < <i>f</i> ≤ 48 MHz</td><td>0x1</td></tr><tr><td>48 MHz < <i>f</i> ≤ 72 MHz</td><td>0x2</td></tr></table>	HCLK 频率(<i>f</i>)	WCYCLE 值	<i>f</i> ≤ 24MHz	0x0	24MHz < <i>f</i> ≤ 48 MHz	0x1	48 MHz < <i>f</i> ≤ 72 MHz	0x2
				HCLK 频率(<i>f</i>)	WCYCLE 值							
				<i>f</i> ≤ 24MHz	0x0							
				24MHz < <i>f</i> ≤ 48 MHz	0x1							
48 MHz < <i>f</i> ≤ 72 MHz	0x2											
				注意：HCLK在PT32U301的最高频率为72MHz的，因此，周期的值不应大于0X2。								
0	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。								

SC_NVMA CR. ONEUS 的作用主要用在嵌入式闪存编程或擦除。控制器根据此设置进行闪存编程或擦除。

4.4.2.9 SC_GCLK_APB - APB 外设门控时钟使能寄存器

The SC_GCLK_APB 寄存器使能 APB 独立外设模块的时钟。

31 RO	30 RO	29 RO	28 RO	27 RO	26 R/W	25 RO	24 R/W	23 RO	22 RO	21 RO	20 RO	19 RO	18 R/W	17 R/W	16 R/W
					COMP		ADC						TIM2	TIM1	TIM0
15 RO	14 RO	13 RO	12 R/W	11 R/W	10 R/W	9 RO	8 RO	7 RO	6 R/W	5 RO	4 R/W	3 RO	2 RO	1 R/W	0 R/W
			PWM	WDT	RTC				I2C		SPI			UART1	UART0

偏移地址: 0x0020

位	名称	类型	复位值	描述
31:27	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
26	COMP	R/W	0	比较器门控时钟控制 控制比较器模块的门控时钟。
25	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
24	ADC	R/W	0	ADC门控时钟控制 控制ADC模块的门控时钟。
23:19	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
18	TM2	R/W	0	定时器2门控时钟控制 控制通用定时器2模块的门控时钟。
17	TM1	R/W	0	定时器1门控时钟控制 控制通用定时器1模块的门控时钟。
16	TM0	R/W	0	定时器0门控时钟控制 控制通用定时器0模块的门控时钟。
15:13	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
12	PWM	R/W	0	PWM门控时钟控制 控制PWM模块的门控时钟。
11	WDT	R/W	0	WDT门控时钟控制 控制WDT模块的门控时钟。
10	RTC	R/W	0	RTC门控时钟控制 控制RTC模块的门控时钟。
9:7	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
6	I2C	R/W	0	I2C0门控时钟控制 控制I2C0模块的门控时钟。
5	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
4	SPI	R/W	0	SPI门控时钟控制 控制SPI模块的门控时钟。
3:2	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
1	UART1	R/W	0	UART1门控时钟控制 控制UART1模块的门控时钟。
0	UART0	R/W	0	UART0门控时钟控制 控制UART0模块的门控时钟。

4.4.2.10 SC_GCLK_AHB - AHB 外设门控时钟使能寄存器

SC_GCLK_AHB 寄存器使能与 AHB 桥连接的单独外设模块的时钟。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 R/W	2 R/W	1 R/W	0 R/W
												GPIOD	GPIOC	GPIOB	GPIOA

偏移地址: 0x0024

位	名称	类型	复位值	描述
31:4	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
3	GPIOD	R/W	0	GPIO_D门控时钟控制。 控制GPIO_D模块的门控时钟。
2	GPIOC	R/W	0	GPIO_C门控时钟控制。 控制GPIO_C模块的门控时钟。
1	GPIOB	R/W	0	GPIO_B门控时钟控制。 控制GPIO_B模块的门控时钟。
0	GPIOA	R/W	0	GPIO_A门控时钟控制。 控制GPIO_A模块的门控时钟。

4.4.2.11 SC_RST_APB - APB 外设复位请求寄存器

寄存器允许软件通过置‘1’或清‘0’来复位和不复位 APB 外设。

31 RO	30 RO	29 RO	28 RO	27 RO	26 R/W	25 RO	24 R/W	23 RO	22 RO	21 RO	20 RO	19 RO	18 R/W	17 R/W	16 R/W
					COMP		ADC						TIM2	TIM1	TIM0
15 RO	14 RO	13 RO	12 R/W	11 R/W	10 R/W	9 RO	8 RO	7 RO	6 R/W	5 RO	4 R/W	3 RO	2 RO	1 R/W	0 R/W
			PWM	WDT	RTC				I2C		SPI			UART1	UART0

偏移地址: 0x0028

位	名称	类型	复位值	描述
31:27	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
26	COMP	R/W	0	比较器复位请求 0: 设置后手动清除 1: 向比较器发送一个复位请求信号
25	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
24	ADC	R/W	0	ADC复位请求 0: 设置后手动清除 1: 向ADC发送一个复位请求信号
23:19	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
18	TIM2	R/W	0	定时器 2复位请求 0: 设置后手动清除 1: 向定时器2发送一个复位请求信号
17	TIM1	R/W	0	定时器 1复位请求 0: 设置后手动清除 1: 向定时器1发送一个复位请求信号
16	TIM0	R/W	0	定时器 0复位请求 0: 设置后手动清除 1: 向定时器0发送一个复位请求信号
15:13	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
12	PWM	R/W	0	PWM复位请求 0: 设置后手动清除 1: 向PWM发送一个复位请求信号
11	WDT	R/W	0	WDT复位请求 0: 设置后手动清除 1: 向WDT发送一个复位请求信号
10	RTC	R/W	0	RTC复位请求 0: 设置后手动清除 1: 向RTC发送一个复位请求信号
9:7	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
6	I2C	R/W	0	I2C0复位请求 0: 设置后手动清除 1: 向I2C0发送一个复位请求信号

位	名称	类型	复位值	描述
5	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
4	SPI	R/W	0	SPI复位请求。 0: 设置后手动清除 1: 向SPI发送一个复位请求信号
3:2	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
1	UART1	R/W	0	UART1 复位请求 0: 设置后手动清除 1: 向UART1发送一个复位请求信号
0	UART0	R/W	0	UART0 复位请求 0: 设置后手动清除 1: 向UART0发送一个复位请求信号。

4.4.2.12 SC_RST_AHB - AHB 外设复位请求寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 R/W	2 R/W	1 R/W	0 R/W
												GPIOD	GPIOC	GPIOB	GPIOA

偏移地址: 0x002C

位	名称	类型	复位值	描述
32:4	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
3	GPIOD	R/W	0	GPIOD复位请求 0: 设置后手动清除 1: 向GPIOD发送一个复位请求信号
2	GPIOC	R/W	0	GPIOC 复位请求。 0: 设置后手动清除 1: 向GPIOC发送一个复位请求信号
1	GPIOB	R/W	0	GPIOB复位请求 0: 设置后手动清除 1: 向GPIOB发送一个复位请求信号
0	GPIOA	R/W	0	GPIOA复位请求 0: 设置后手动清除 1: 向GPIOA发送一个复位请求信号

4.4.2.13 SC_CK_CTRL - 时钟控制寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 R/W	6 RO	5 RO	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
											PLLEN	LSEEN	LSIEN	HSEEN	HSIEN

偏移地址: 0x0030

位	名称	类型	复位值	描述
31:13	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
12:10	LSEDRV	R/W	0x1	LSE振荡器驱动能力
9	HSEFBYP	R/W	1	HSE时钟输出旁路 0: HSE 1: Bypass
8	HSIFBYP	R/W	1	HSI时钟输出滤波器/旁路 0: Filter 1: Bypass
7:5	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
4	PLLEN	R/W	0	PLL 使能 0: PLL 关闭 1: PLL 打开
3	LSEEN	R/W	0	LSE 时钟使能 0: LSE 关闭 1: LSE 打开
2	LSIEN	R/W	0	LSI 时钟使能 0: LSI 关闭 1: LSI 打开
1	HSEEN	R/W	0	HSE 时钟使能 0: HSE 关闭 1: HSE 打开
0	HSIEN	R/W	1	HSI 时钟使能 0: HSI 关闭 1: HSI 打开

4.4.2.14 SC_CK_CONF - 时钟配置寄存器

31 WO	30 R/W	29 R/W	28 R/W	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 R/W	20 R/W	19 R/W	18 R/W	17 R/W	16 R/W
CKCHG	MCOPRE				MCO					PLLMUL				PLLSRC	
15 RO	14 RO	13 R/W	12 R/W	11 RO	10 R/W	9 R/W	8 R/W	7 R/W	6 R/W	5 R/W	4 R/W	3 RO	2 RO	1 R/W	0 R/W
		HSEDIV			PPRE			HPRE						SCKSW	

偏移地址: 0x0034

位	名称	类型	复位值	描述
31	CKCHG	W	0	改变时钟配置
30:28	MCOPRE	R/W	0x0	芯片时钟输出预分频系数 当微控制器输出被禁止用来避免系统故障时，我们强烈推荐改变预分频系数。 0x0：芯片时钟预分频值为1 0x1：芯片时钟预分频值为2 0x2：芯片时钟预分频值为4 0x4：芯片时钟预分频值为8
27	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
26:24	MCO	R/W	0x0	微控制器时钟输出 0x0：没有时钟输出 0x1：内部低速振荡器时钟(LSI)输出 0x2：外部低速振荡器时钟(LSE)输出 0x3：4 MHz内部RC振荡器时钟(HSI)输出 0x4：外部4-32 MHz振荡器时钟(HSE)输出 0x5：系统时钟输出 0x6：AHB时钟输出 0x7：APB时钟输出
23:22	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
21:17	PLLMUL	R/W	0x0	PLL倍频系数 只有在PLL关闭的情况下才可被写入。PLL时钟输出频率为PLLSRC x PLLMUL[4:0] MHz 注意：PLL输出频率不能超过48 MHz。 0：4M外部/内部振荡器 (PLLSRC) 1：PLL 1倍频输入 2：PLL 2倍频输入 3：PLL 3倍频输入 ... 12：PLL 12倍频输入
16	PLLSRC	R/W	0	PLL输入时钟源 只有在PLL关闭的情况下才可被写入 0：选择HSI 作为PLL输入时钟selected as PLL input clock 1：选择HSE/PREDIV 作为PLL输入时钟
15:14	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。

位	名称	类型	复位值	描述
13:12	<i>HSEDIV</i>	R/W	0x0	PREDIV 分频系数 只有在PLL关闭的情况下才可被写入 0x0 : PREDIV 不分频 0x1 : PREDIV 2分频输入 0x2 : PREDIV 4分频输入 0x3 : PREDIV 8分频输入
11	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
10:8	<i>PPRE</i>	R/W	0x0	PCLK 预分频系数 用来控制APB时钟的预分频系数 0xX : HCLK 不分频 0x4 : HCLK 2分频 0x5 : HCLK 4分频 0x6 : HCLK 8分频 0x7 : HCLK 16分频
7:4	<i>HPRE</i>	R/W	0x0	HCLK 预分频系数 用来控制AHB时钟的预分频系数 <0x8 : SYSCLK不分频 0x8 : SYSCLK 2分频 0x9 : SYSCLK 4分频 0xA : SYSCLK 8分频 0xB : SYSCLK 16分频 >0xB : 保留
3	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
2:0	<i>SCKSW</i>	R/W	0x3	系统时钟切换 0x0 : 选择 HSI 作为系统时钟 0x1 : 选择 HSE/PREDIV 作为系统时钟 0x2 : 选择 PLL 作为系统时钟 0x3 : 保留

4.4.2.15 SC_CK_STAT - 时钟源状态寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
							SCKSRDY	CSSFLAG			PLLRDY	LSERDY	LSIRDY	HSERDY	HSIRDY

偏移地址: 0x003C

位	名称	类型	复位值	描述
31:9	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
8	SCKSRDY	RO	0	系统时钟转换准备就绪标志 0: 系统时钟转换未准备就绪 1: 系统时钟转换准备就绪
7:5	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
4	PLLRDY	RO	0	PLL 时钟就绪标志 0: PLL 时钟未就绪 1: PLL时钟准备就绪
3	LSERDY	RO	0	LSE 时钟就绪标志 0: LSE 时钟未就绪 1: LSE时钟准备就绪
2	LSIRDY	RO	0	LSI 时钟就绪标志 0: LSI 时钟未就绪 1: LSI时钟准备就绪
1	HSERDY	RO	0	HSE 时钟就绪标志 0: HSE 时钟未就绪 1: HSE 时钟准备就绪
0	HSIRDY	RO	0	HSI 时钟就绪标志 0: HSI 时钟未就绪 1: HSI时钟准备就绪

4.4.2.16 SC_BK_CTRL - 备份控制寄存器

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

偏移地址: 0x0040

位	名称	类型	复位值	描述
31:2	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
1	WKF	RO	0	低功耗唤醒标志
0	BKREGEN	RW	0	允许访问备份寄存器 1: 允许访问备份寄存器 0: 禁止访问备份寄存器

4.4.2.17 SC_PWR_CTRL - 电源控制寄存器

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

偏移地址: 0x0048

位	名称	类型	复位值	描述
31:2	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
1	NRSTPL	RW	0	NRST 下拉，高电平有效 0: 无作用 1: 下拉
0	ADCPPE	RW	0	ADC/比较器 LDO 允许 0: LDO 禁止 1: LDO 允许

4.4.2.18 SC_PVD_DET - PVD 检测寄存器

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

偏移地址: 0x004C

位	名称	类型	复位值	描述
31:8	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
7:6	DETMODE	RW	0x0	PVD检测模式设置 0: 中断 1: 门控时钟 2: 复位请求
5	DETEN	RW	0	PVD 检测允许 0: PVD 检测禁止 1: PVD 检测允许
4	INTRIE	RW	0	PVD 检测中断允许 0: PVD检测中断禁止 1: PVD检测中断允许
3:1	VOLT	RW	0x7	PVD 电压选择 0x0 : R-2.2V, F-2.1V 0x1 : R-2.3V, F-2.2V 0x2 : R-2.4V, F-2.3V 0x3 : R-2.5V, F-2.4V 0x4 : R-2.6V, F-2.5V 0x5 : R-2.7V, F-2.6V 0x6 : R-2.8V, F-2.7V 0x7 : R-2.9V, F-2.8V
0	PHYEN	RW	1	PVD 检测电流允许，高电平有效

4.4.2.19 SC_SLP_APB - 睡眠模式下 APB 外围门控时钟使能寄存器

31 RO	30 RO	29 RO	28 RO	27 R/W	26 R/W	25 RO	24 R/W	23 RO	22 RO	21 RO	20 R/W	19 R/W	18 R/W	17 R/W	16 R/W
				DAC	COMP		ADC				TIM4	TIM3	TIM2	TIM1	TIM0
15 RO	14 RO	13 RO	12 R/W	11 R/W	10 RO	9 R/W	8 R/W	7 R/W	6 R/W	5 RO	4 R/W	3 RO	2 RO	1 R/W	0 R/W
			PWM	WDT		I2S1	I2S0	I2C1	I2C0		SPI			UART1	UART0

偏移地址: 0x0050

位	名称	类型	复位值	描述
31:27	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
26	COMP	R/W	0	睡眠模式下，比较器门控时钟控制 此位用来控制在睡眠模式下的比较器的门控制时钟
25	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
24	ADC	R/W	0	睡眠模式下，ADC门控时钟控制 此位用来控制在睡眠模式下的ADC的门控制时钟
23:19	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
18	TM2	R/W	0	睡眠模式下，定时器2门控时钟控制 此位用来控制在睡眠模式下的定时器2的门控制时钟
17	TM1	R/W	0	睡眠模式下，定时器1门控时钟控制 此位用来控制在睡眠模式下的定时器1的门控制时钟
16	TM0	R/W	0	睡眠模式下，定时器0门控时钟控制 此位用来控制在睡眠模式下的定时器0的门控制时钟
15:13	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
12	PWM	R/W	0	睡眠模式下，PWM门控时钟控制 此位用来控制在睡眠模式下的PWM的门控制时钟
11	WDT	R/W	0	睡眠模式下，WDT门控时钟控制 此位用来控制在睡眠模式下的WDT的门控制时钟
10	RTC	R/W	0	睡眠模式下，RTC门控时钟控制 此位用来控制在睡眠模式下的RTC的门控制时钟
9:7	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
6	I2C	R/W	0	睡眠模式下，I2C门控时钟控制 此位用来控制在睡眠模式下的I2C的门控制时钟
5	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
4	SPI	R/W	0	睡眠模式下，SPI门控时钟控制 此位用来控制在睡眠模式下的SPI的门控制时钟
3:2	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
1	UART1	R/W	0	睡眠模式下，UART1门控时钟控制 此位用来控制在睡眠模式下的UART1的门控制时钟
0	UART0	R/W	0	睡眠模式下，UART0门控时钟控制 此位用来控制在睡眠模式下的UART0的门控制时钟

4.4.2.20 SC_SLP_AHB – 睡眠模式下 AHB 外设门控时钟使能寄存器

31 R/W	30 R/W	29 RO	28 RO	27 RO	26 RO	25 R/W	24 R/W	23 RO	22 RO	21 RO	20 R/W	19 RO	18 RO	17 RO	16 RO
RTC	EBI					USBPHY	USB				DMA				
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 R/W	2 R/W	1 R/W	0 R/W
												GPIOD	GPIOC	GPIOB	GPIOA

偏移地址: 0x0054

位	名称	类型	复位	描述
31:4	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
3	GPIOD	R/W	0	睡眠模式下，GPIO_D门控时钟控制 此位用来控制在睡眠模式下的GPIO_D的门控制时钟
2	GPIOC	R/W	0	睡眠模式下，GPIO_C门控时钟控制 此位用来控制在睡眠模式下的GPIO_C的门控制时钟
1	GPIOB	R/W	0	睡眠模式下，GPIO_B门控时钟控制 此位用来控制在睡眠模式下的GPIO_B的门控制时钟
0	GPIOA	R/W	0	睡眠模式下，GPIO_A门控时钟控制 此位用来控制在睡眠模式下的GPIO_A的门控制时钟

4.4.2.21 SC_DSLP_APB - 深度睡眠模式下 APB 外设门控时钟使能寄存器

31 RO	30 RO	29 RO	28 RO	27 R/W	26 R/W	25 RO	24 R/W	23 RO	22 RO	21 RO	20 R/W	19 R/W	18 R/W	17 R/W	16 R/W
				DAC	COMP		ADC				TM4	TM3	TM2	TM1	TM0
15 RO	14 RO	13 RO	12 R/W	11 R/W	10 RO	9 R/W	8 R/W	7 R/W	6 R/W	5 RO	4 R/W	3 RO	2 RO	1 R/W	0 R/W
			PWM	WDT		I2S1	I2S0	I2C1	I2C0		SPI			UART1	UART0

偏移地址: 0x0058

位	名称	类型	复位	描述
31:27	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
26	COMP	R/W	0	深度睡眠模式下，比较器门控时钟控制 此位用来控制在深度睡眠模式下的比较器的门控制时钟。
25	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
24	ADC	R/W	0	深度睡眠模式下，ADC控时钟控制 此位用来控制在深度睡眠模式下的ADC的门控制时钟。
23:19	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
18	TM2	R/W	0	深度睡眠模式下，定时器2门控时钟控制 此位用来控制在深度睡眠模式下的定时器2的门控制时钟。
17	TM1	R/W	0	深度睡眠模式下，定时器1门控时钟控制 此位用来控制在深度睡眠模式下的定时器1的门控制时钟。
16	TM0	R/W	0	深度睡眠模式下，定时器0门控时钟控制 此位用来控制在深度睡眠模式下的定时器0的门控制时钟。
15:13	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
12	PWM	R/W	0	深度睡眠模式下，PWM门控时钟控制 此位用来控制在深度睡眠模式下的PWM的门控制时钟。
11	WDT	R/W	0	深度睡眠模式下，WDT门控时钟控制 此位用来控制在深度睡眠模式下的WDT的门控制时钟。
10	RTC	R/W	0	深度睡眠模式下，RTC门控时钟控制 此位用来控制在深度睡眠模式下的RTC的门控制时钟。
9:7	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
6	I2C	R/W	0	深度睡眠模式下，I2C门控时钟控制 此位用来控制在深度睡眠模式下的I2C的门控制时钟。
5	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
4	SPI	R/W	0	深度睡眠模式下，SPI门控时钟控制 此位用来控制在深度睡眠模式下的SPI的门控制时钟。
3:2	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
1	UART1	R/W	0	深度睡眠模式下，UART1门控时钟控制 此位用来控制在深度睡眠模式下的UART1的门控制时钟。
0	UART0	R/W	0	深度睡眠模式下，UART0门控时钟控制 此位用来控制在深度睡眠模式下的UART0的门控制时钟。

4.4.2.22 SC_DSLP_AHB -深度睡眠模式下 AHB 外设门控时钟使能寄存器

31 R/W	30 R/W	29 RO	28 RO	27 RO	26 RO	25 R/W	24 R/W	23 RO	22 RO	21 RO	20 R/W	19 RO	18 RO	17 RO	16 RO
RTC	EBI					USBPHY	USB				DMA				
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 R/W	2 R/W	1 R/W	0 R/W
												GPIOD	GPIOC	GPIOB	GPIOA

偏移地址: 0x005C

位	名称	类型	复位	描述
31:4	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
3	GPIOD	RW	0	深度睡眠模式下，GPIO_D门控时钟控制 此位用来控制在深度睡眠模式下的GPIO_D的门控制时钟
2	GPIOC	RW	0	深度睡眠模式下，GPIO_C门控时钟控制 此位用来控制在深度睡眠模式下的GPIO_C的门控制时钟
1	GPIOB	RW	0	深度睡眠模式下，GPIO_B门控时钟控制 此位用来控制在深度睡眠模式下的GPIO_B的门控制时钟
0	GPIOA	RW	0	深度睡眠模式下，GPIO_A门控时钟控制 此位用来控制在深度睡眠模式下的GPIO_A的门控制时钟

4.4.2.23 SC_OSC_TRIM - 环形 OSC 硬件校正寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 R/W	25 WO	24 RO	23 R/W	22 R/W	21 R/W	20 R/W	19 R/W	18 R/W	17 R/W	16 R/W
					SRC	SWTRIM		CALVAL							
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 WO
CALCNT												FAIL	NOEQU	BUSY	HWTRIM

偏移地址: 0x006C

位	名称	类型	复位	描述
31:27	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
26	SRC	R/W	0	内部4M晶振校正源 0: IO端口 (UART0_RX :9600 bps) 1: 外部晶振32.768K Hz
25	SWTRIM	WO	0	HSI 手动校正流控制 此位提供用户设置的校正数值去调整HSI的频率，例如温度或电压改变带来的影响。
24	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
23:16	CALVAL	R/W	0x80	校正结果值 注意: 在系统进入待机模式时，此校正结果将不会被储存。使用者可以将此结果储存至SC_BK_SCTRL中的CALVAL位，便可保障当系统从待机模式醒来时，校正结果不会消失。
15:4	CALCNT	RO	0x0	校正计数反馈值
3	FAIL	RO	0	1: 指示标志位校正失败，电路不能发现任何参考校正源。
2	NOEQU	RO	0	1: 校正结果不等于4MHz时钟
1	BUSY	RO	0	0: HSI 完成Trimming。 1: HSI 正在Trimming 当此位显示为1时, 检查FAIL及NOEQU 确定Trimming正确
0	HWTRIM	WO	0	Trimming 开始 1: 当SRC=0，内部4M振荡器开始自动Trimming

4.4.2.24 SC_BKRTC_CTRL - 备份 RTC 控制器寄存器

31 RO	30 R/W	29 R/W	28 R/W	27 R/W	26 R/W	25 R/W	24 R/W	23 R/W	22 R/W	21 R/W	20 R/W	19 R/W	18 R/W	17 R/W	16 R/W
	CRS	CRV						CLKSCAL							
15 RO	14 R/W	13 R/W	12 R/W	11 R/W	10 R/W	9 R/W	8 R/W	7 RO	6 RO	5 RO	4 RO	3 R/W	2 R/W	1 R/W	0 R/W
	CLKPSCAL											CKSEL	RST	COUNTEN	

偏移地址: 0x0080

位	名称	类型	复位	描述
31	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
30	CRS	R/W	0	校准值控制 1: 减少校准值, CRV [5:0] 0: 增加校准值, CRV[5:0]
29:24	CRV	R/W	0	RTC时钟校准值
23:16	CLKSCAL	R/W	0xFF	RTC时钟分频器，最大分是256
15	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
14:8	CLKPSCAL	R/W	0x7F	RTC时钟分频器，最大分是128
7:4	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
3:2	CKSEL	R/W	0x0	RTC时钟源选择 00: No clock 01: LSE振荡器时钟作为RTC时钟 10: LSI 振荡器时钟作为RTC时钟 11: HSE振荡器时钟(除以128)作为RTC时钟
1	RST	R/W	0	RTC计数器复位请求 0: 复位信号不发送 1: 发送复位信号
0	COUNTEN	R/W	0	RTC计数启用 0: 禁止RTC计数 1: 允许RTC计数

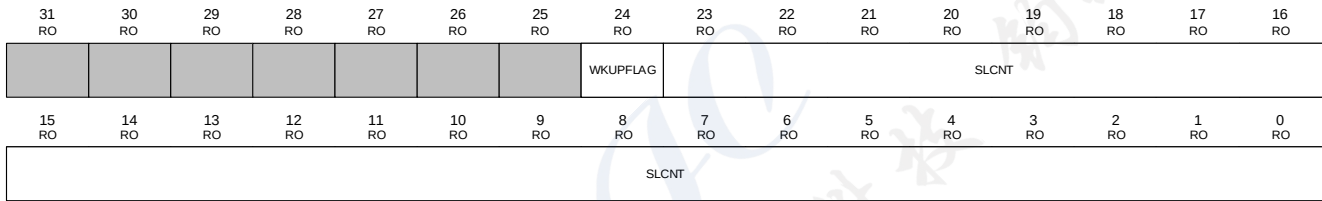
4.4.2.25 SC_BKSLP_CTRL - 备份 RTC 睡眠控制器寄存器

31 R/W	30 R/W	29 R/W	28 R/W	27 R/W	26 R/W	25 R/W	24 R/W	23 WO	22 WO	21 WO	20 WO	19 WO	18 WO	17 WO	16 WO
CWUF	CKSRCOFF	LDOLP	LDOOFF	WKUPEN	CNTREN	ALRMEN	ALRMMS	ALRMVAL							
15 WO	14 WO	13 WO	12 WO	11 WO	10 WO	9 WO	8 WO	7 WO	6 WO	5 WO	4 WO	3 WO	2 WO	1 WO	0 WO
ALRMVAL															

偏移地址: 0x0084

位	名称	类型	复位	描述
31	CWUF	R/W	0	清除唤醒标志 阅读：该位是由硬件设置以表明该设备从低功耗模式中唤醒。 写：设置该位来清除唤醒标志和SC_BK_STATUS.WKUPFLAG位
30	CKSRCOFF	R/W	0	在深度休眠模式下，关闭所有的时钟源
29	LDOLP	R/W	0	在深度休眠模式，打开1.8V LDO进入低功耗模式
28	LDOOFF	R/W	0	在深度休眠模式，关闭LDO1.8V电源
27	WKUPEN	R/W	0	启用外部引脚来唤醒系统 1：启用系统通过外部引脚被唤醒
26	CNTREN	R/W	0	备份RTC睡眠计数器启用 1：启用RTC睡眠计数器
25	ALRMEN	R/W	0	备份RTC睡眠闹钟启用 1：启用RTC睡眠闹钟
24	ALRMMS	R/W	0	备份RTC睡眠闹钟计数器设定秒钟和分钟的值
23:0	ALRMVAL	WO	0x1FFFF	备份RTC睡眠闹钟计数器设定值

4.4.2.26 SC_BK_STATUS - 状态备份寄存器

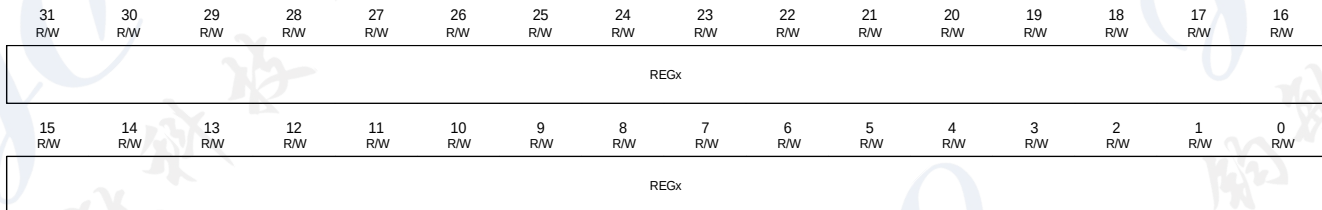


偏移地址: 0x0088

位	名称	类型	复位	描述
31:25	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
24	WKUP	RO	0	唤醒标志 此标志表明系统从RTC或唤醒事件唤醒。 0: 从RTC唤醒 1: 系统由低功耗流程唤醒
23:0	保留	RO	0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
11:0	SLCNT	RO	0x0	备份RTC睡眠计数器的值

4.4.2.27 SC_BK_REGx - 备份 X 寄存器

0x0090 to 0x00AC 之间有 8 个 32 位备份寄存器. x 的值从 0 到 7.



偏移地址: 0x0090 – 0x00AC

位	名称	类型	复位	描述
31:0	REGx	R/W	0x0	用户定义域

4.5 闪存控制器 (FC)

PT32U301 嵌入式闪存具有 4KB 的信息块和 32KB 的嵌入式闪存，可以通过 ISP 程序更新。在线系统编程(ISP) 功能允许用户更新已焊接在 PCB 上的芯片程序代码。芯片上电后 Cortex™-M0 的 CPU 读取闪存里的程序代码。

- 嵌入式 flash 可以运行在高达 24MHz 的零等待状态连续的读取访问。
- 所有的嵌入式 flash 都支持 512 字节的页擦除。
- 嵌入式 flash 支持在线应用程序编程。
- 32-位 编程

4.5.1 存储器结构

嵌入式闪存控制器是由两部分组成

- 控制段(0x50010000 - 0x5001FFFF)用来配置控制器的时钟频率，读周期延迟时间，编程周期信息等。
- 存储段(0x08000000 - 0x08007FFF / 0x1FFFF000 - 0x1FFFFFFF), 地址范围取决于映射寄存器的设置。此段常用来从 flash 宏或 flash 操作(例如信息块或者主块的读取，编程，页擦除和整体擦除)中读取数据。

4.5.2 闪存控制器时钟频率

当系统时钟发生改变时，所配置的嵌入式闪存控制器等待周期的序列不同取决于时钟是否由低到高或高到低的速度改变。低到高速度改变时，在 PLL 频率改变之前 SC_NVMACR 寄存器的等待周期首先被加载。高到低速度改变顺序颠倒，PLL 先被修改，然后设置等待周期。

当工作频率比 flash 内存的读取速度快时，都要增加等待周期。例如，对工作频率是在 24 兆赫，没有等待周期需要加入；如果当前的工作频率是介乎於 24 兆赫和 48 兆赫中间，然后需加 1 个等待周期，以满足读访问时间。并将 SC_NVMACR 寄存器的 WCYCLE 设为 1。

4.5.3 主闪存重映

當選擇了主闪存作為開機模式，使用者可更進一步設定 SC_REMAP 的 EFBASE 位去任意選擇訪問 Eflash 的指定位置。通過寫入指定的值到 EFBASE，訪問 EFlash 的位置亦有所不同：

1. 寫入 0x2 至 SC_REMAP 的 REMAP 位。
2. 決定訪問位置，在 PT32U301，訪問位置的深度基於 4 Kbyte 的倍數。

舉例說，當主闪存存储器被映射到启动空间(0x0000 0000)時：

- 寫 0x0 入 EFBASE，訪問會由 0x0000 0000 開始
- 寫 0x1 入 EFBASE，訪問會由 0x0000 1000 開始
- 寫 0x2 入 EFBASE，訪問會由 0x0000 2000 開始

如此類推...更多描述，參考 SC_REMAP 的 EFBASE 位。

所有開機程序都是只讀不寫，寫的操作基於使用者設定 FC_PADDR 寄存器。

4.5.4 闪存编程/擦除

所有擦除/编程操作由三个寄存器处理: **FC_CMD**, **FC_PDATA**, 和 **FC_PADDR**. 在闪存操作(写, 页擦除或整体擦除)时, 访问闪存将会受到抑制。指令和文字读取都保持关闭, 直到闪存操作完成。如果闪存操作期间需要执行指令, 那么正在执行的代码必须放置在 **SRAM** 中在闪存操作完成后直接执行。

- 编程步骤:
 1. 将源数据写入到 **FC_PDATA** 寄存器。
 2. 将目标地址写入到 **FC_PADDR** 寄存器。
 3. 将 **PGCMD** 位写入到 **FC_CMD** 寄存器。
 4. 查询 **FC_CMD** 寄存器直到 **PGCMD** 位自动清零。
- 页擦除步骤:
 1. 将页地址写入到 **FC_PADDR** 寄存器。
 2. 将 **PGERASE** 位写入到 **FC_CMD** 寄存器
 3. 查询 **FC_CMD** 寄存器直到 **PGERASE** 位自动清零。
- 整体擦除步骤:
 1. 将 **MASERASE** 位写到 **FC_CMD** 寄存器。
 2. 查询 **FC_CMD** 寄存器直到 **MASERASE** 位自动清零

4.5.5 闪存控制寄存器映射

基址: **0x5001_0000**

偏移地址	标识	类型	复位值	描述	参见页面
0x0004	FC_CMD	WO	0x0000_0000	Flash 命令寄存器	72
0x0008	FC_PDATA	R/W	0x0000_0000	Flash 程序数据寄存器	72
0x0010	FC_PADDR	R/W	0x0000_0000	Flash 编程地址寄存器	73

4.5.5.1 FC_CMD - 闪存控制器命令寄存器

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

偏移地址: 0x0004

位	名称	类型	复位值	描述
31:4	reserved	RO	0x0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
3	PGERASE	WO	0	页擦除周期 此位用来擦除闪存主内存页面，操作流程完成后自动清除。擦除时间最短为40ms。 1: 设置此位擦除被FC_PADDR 指定的闪存内存页面。
2	MASERASE	WO	0	整体擦除周期 此位用来对闪存主内存进行整体擦除，操作流程完成后自动清除。擦除时间最短为40ms。 1: 设置此位来擦除闪存主内存。
1	PGCMD	WO	0	编程周期。 1: 存储在FC_PDATA中的数据被写入由FC_PADDR的内容指定的位置。 需要 30-40μs.
0	SLEEP	R/W	0	闪存睡眠模式使能 1: 当设备进入低功耗模式，嵌入式闪存将进入睡眠模式。

4.5.5.2 FC_PDATA - 闪存控制器程序数据寄存器

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

偏移地址: 0x0008

位	名称	类型	复位值	描述
31:0	DATA	R/W	0x0	嵌入式程序32位数据寄存器

4.5.5.3 FC_PADDR - 闪存控制器编程地址寄存器

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

偏移地址: 0x0010

位	名称	类型	复位值	描述	
31:17	保留	RO	0x0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。	
16	INFO	R/W	0	嵌入式闪存信息块控制 1: 启用Flash信息块控制写入或擦除。	
15:0	HADDR	R/W	0x0	嵌入式编程/擦除地址寄存器。(页长度=512 字节)	
				整体擦除	HADDR[14:0] 保留
				页擦除	HADDR[8:0] 保留 HADDR[14:9] 页地址.(64 页/32K 字节)
				信息擦除	HADDR [8:0] & HADDR [14:12] 无作用. HADDR [11:9] 信息页面地址(8 页/4K 字节)
				编程	HADDR [1:0] 保留. HADDR[14:2] 双字地址

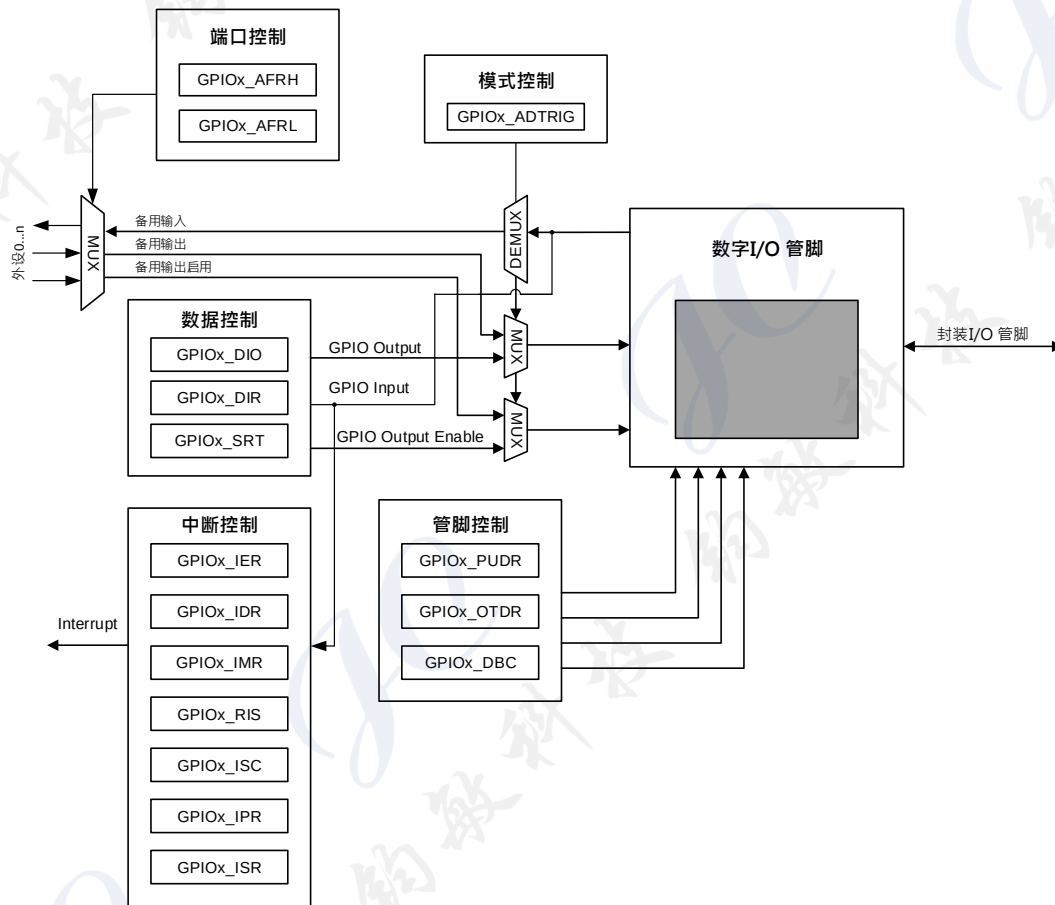
4.6 通用输入/输出 (GPIO)

PT32U301 中, GPIO 模块具有多达 45 个通用 I/O, 根据芯片配置与其它功能引脚共享使用。这 45 个引脚被分为 4 个端口, 分别为 GPIOA, GPIOB, GPIOC, and GPIOD。每个引脚都是独立的, 而且有相应的寄存器来控制引脚数据和功能模式。复位后, I/O 引脚默认浮空。

- 输入状态: 浮空, 上拉/下拉, 模拟输入
- 输出状态: 开漏输出, 推挽输出(一般用法)
- 引脚可配置为施密特触发输入
- 位电平设置和清除寄存器允许一个单一的指令集或清除端口中任一位上的数据。
- 消抖时间可编程。(1-7) * (1-256) 时钟周期。
- 可编程控制 GPIO 中断
 - 屏蔽中断触
 - 上升, 下降沿触
 - 高, 低 平触 触

4.6.1 块图

图 4.6-1: GPIO 框图



4.6.2 功能描述

4.6.2.1 数据控制

GPIO 数据处理由下列寄存器控制:

- 数据输入/ 输出
GPIO (A/B/C/D)输入和输出寄存器(**GPIOx_DIO**)包含相应 I/O 端口的输入或输出值。根据端口的设置方向, 将端口配置为只读/只写。
- 数据直接控制
GPIO (A/B/C/D)数据方向寄存器(**GPIOx_DIR**) 配置每个独立的引脚为输入或输出。每个引脚默认设置为输入。将位置‘0’可把引脚设置为输出状态, 并同时清除该 GPIO 端口相应的寄存器位。
- 数据设置和清除
GPIO (A/B/C/D)设置/复位寄存器 (**GPIOx_SRT**)可控制单独的位, 不影响其他的位。通过在寄存器中将位置‘1’可以设置或清除 **GPIOx_DIO** 寄存器中相应的位。

4.6.2.2 中断控制

GPIO 中断是由七个寄存器控制

- 中断控制 (IER, IDR, IMR)
默认情况下, 每个引脚的中断都是被禁用的, 用户可以根据自己需要配置相应的引脚为中断。如果端口(X)的数据方向寄存器被设置为输出或硬件模式, 那么其相应的中断位将被禁用。中断使能寄存器(**GPIOx_IER**)通过置‘1’来允许中断请求。同样, 中断禁用寄存器(**GPIOx_IDR**)通过置‘1’来禁止中断请求。IER 和 IDR 是控制中断屏蔽的只写寄存器, 这两个寄存器的总体结果可在中断屏蔽寄存器(**GPIOx_IMR**)中查看。IMR 是只读存储器, 用过查看状态‘1’或‘0’来判断中断请求是否被允许。
- 中断状态读取 (RIS)
中断状态读取寄存器(**GPIOx_RIS**)可查看所有模块的中断状态, 只可读。
- 清除中断 (ISC)
中断状态和中断清零寄存器(**GPIOx_ISC**)用来指示模块的非屏蔽中断状态, 因为只有已屏蔽中断会被发送到处理器进行处理。通过将寄存器位置‘1’可以清除中断状态或者通过向 IDR 置‘1’来禁用中断。
- 触发和优先级定义
中断触发寄存器(**GPIOx_ISR**)配置相应的引脚触发电平或上升, 下降沿。中断优先级寄存器(**GPIOx_IPR**)选择中断/事件信号的优先级。

4.6.2.3 模拟配置

当 I/O 端口通过 ADC 触发寄存器(**GPIOx_ADTRIG**) 被设定为模拟配置时, 输出缓冲区被禁止。施密特触发器输入被激活, 其输出是一个常数, ‘0’。

4.6.2.4 模式控制

每个 I/O 引脚通过多路复用器连接到板载外设/模块的仅允许同时只有一个外设复用功能 (AF)。默认情况下每个 I/O 引脚连接到备用功能 0 (AF0)。引脚可以工作在备用功能模式下, 即 AF0-AF8。由控制两个复用功能 寄存器: 复用功能低位寄存器(**GPIOx_AFRL**)和复用功能高位寄存器(**GPIOx_AFGH**)。

4.6.2.5 引脚控制

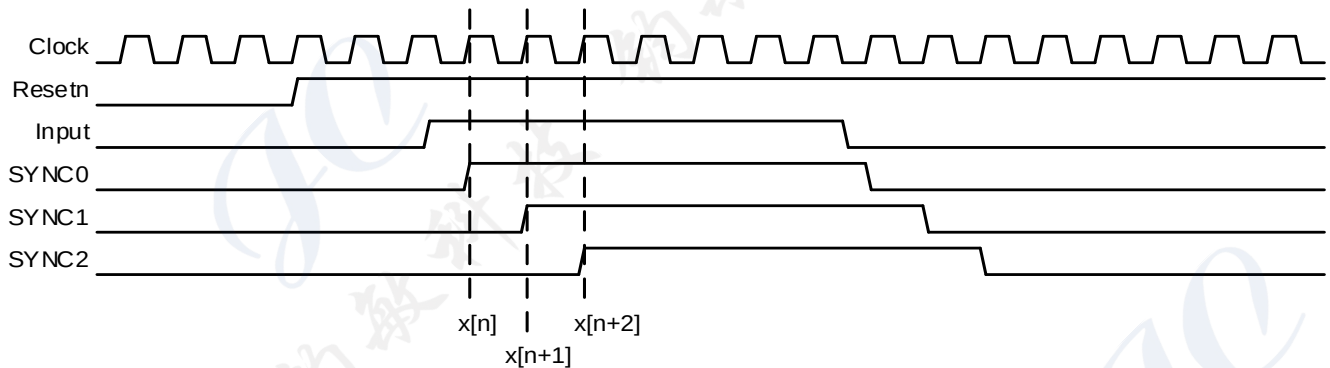
GPIO 的引脚控制寄存器包括 GPIO (A/B/C/D)端口上拉/下拉寄存器 (GPIOx_PUDR)和 GPIO (A/B/C/D) 端口输出类型和驱动寄存器(GPIOx_OTDR). 这些寄存器控制上拉和下拉的电阻，输出类型和驱动能力。

4.6.2.6 防抖功能描述

每个 GPIO 引脚都有防抖动功能。从 IO 到内核的时间图可以表示为下：

当信号进入 GPIO,通过采样电路，波形如下：

图 4.6-2: 防抖波形



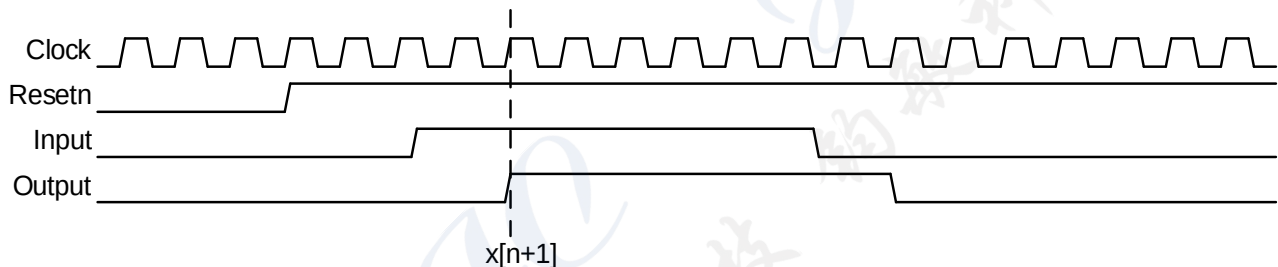
在上面的图中，SYNC0 是指输入信号由系统时钟一次采样;SYNC1 是指 SYNC0 被系统时钟一次采样； SYNC2 表示 SYNC1 由系统时钟一次采样。SYNC0、 SYNC1 和 SYNC2 可以表示成 $x[n] \times T_s$, $x[n+1] \times T_s$ 和 $x[n+2] \times T_s$ 。 T_s 是系统时钟周期， n 是采样时间。如果关闭防抖模块，时间就可以表示为 $x[n+1] \times T_s$ 。

如果打开防抖模块，SYNC2 信号将进入去抖电路，那么它将被采样，计数(用户设定样本频率和计数时间值)。时间可以表达为 $[(SPT+1) \times (FILTCNT+1)] \times T_s$ 。 SPT 是采样频率值， FILTCNT 是计数次数。当 SYNC1 和 SYNC2 不相等时,计数值将被清零。如果计数值溢满 FILTCNT 寄存器,防抖模块将输出信号。

例 1

系统频率 48MHz，防抖功能关闭。

图 4.6-3: 去抖动波形

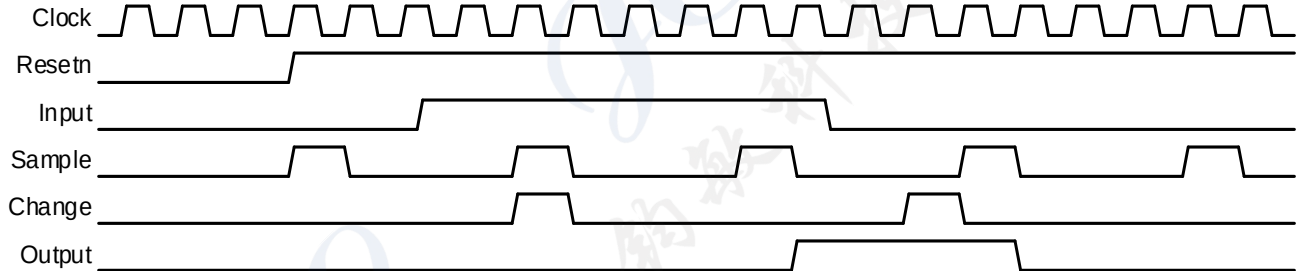


在上面的图中，防抖功能被禁用，所以输出是 $x[n+1] \times T_s$ 。

例 2

系统频率为 48MHz, 采样频率(SPT) 设置成 0x3, 计数值 (FILTCNT)是 0x0.

图 4.6-4: 去抖动波形



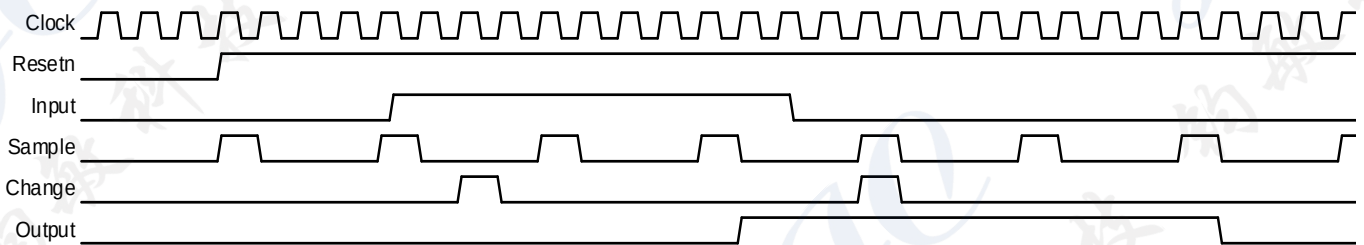
上面的图中，防抖模块启用，“Sample”是采样频率，“Change”表示 SYNC1 和 SYNC2 不相同。当“Change”时高电平，表示计数值将被清零，所以“Output” 将会在“Change”发生和 SYNC2 信号被“Sample”一次采样后设置。

去抖电路采样后，它有最大 $4 \times T_s$ 的采样错误，所以输入时间是： $[(3+1) \times (0+1)] \times T_s = 4 \times T_s$ ，然后增加采样电流 $3 \times T_s$ ，结果为 $7 \times T_s$ 。比较关系图，输入时间为 $6 \times T_s$ ，丢失的 $1 \times T_s$ 就是采样错误。

例 3

系统频率为 48MHz, 采样频率 (SPT) 是 0x3,计数值 (FILTCNT) 为 0x1。

图 4.6-5: 去抖动波形



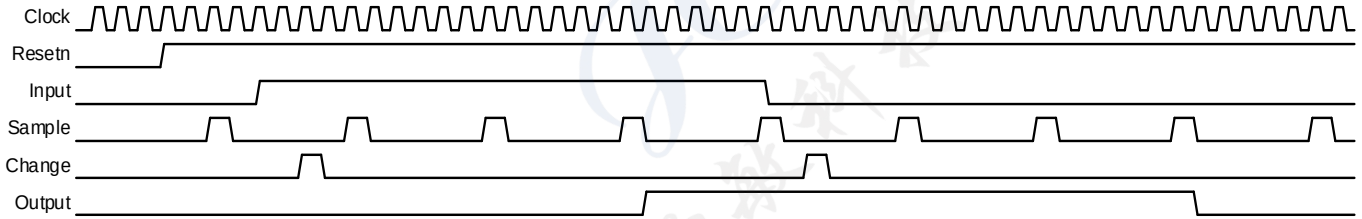
上面的图中，防抖模块启用，“Sample”是采样频率，“Change”表示 SYNC1 和 SYNC2 不相同。当“Change”时高电平，表示计数值将被清零，所以“Output” 将会在“Change”发生和 SYNC2 信号被“Sample”一次采样后设置

去抖电路采样后，它有最大 $4 \times T_s$ 的采样错误，所以输入时间是： $[(3+1) \times (1+1)] \times T_s = 8 \times T_s$ ，然后增加采样电 流 $3 \times T_s$ ，结果为 $11 \times T_s$ 。比较关系图，输入时间为 $8 \times T_s$ ，丢失的 $3 \times T_s$ 就是采样错误。

例 4

系统频率为 48MHz, 采样频率 (SPT) 是 0x5, 计数值 (FILT CNT) 为 0x2。

图 4.6-6: 去抖动波形



上面的图中，防抖模块启用，“Sample”是采样频率，“Change”表示 SYNC1 和 SYNC2 不相同。当“Change”时高电平，表示计数值将被清零，所以“Output”将会在“Change”发生和 SYNC2 信号被“Sample”一次采样后设置

去抖电路采样后，它有最大 $6 \times T_s$ 的采样错误，所以输入时间是： $[(5+1) \times (2+1)] \times T_s = 18 \times T_s$ ，然后增加采样电流 $3 \times T_s$ ，结果为 $21 \times T_s$ 。比较关系图，输入时间为 $16 \times T_s$ ，丢失的 $5 \times T_s$ 就是采样错误。

4.6.3 通用输入/输出寄存器映射

GPIO 端口基地址:

- GPIO Port A: 0x5002_0000
- GPIO Port B: 0x5002_0100
- GPIO Port C: 0x5002_0200
- GPIO Port D: 0x5002_0300

偏移地址	标识	类型	复位值	描述	参考页
0x0000	GPIOx_DIO	R/W	0x0000_FFFF	GPIOx 数据输入输出寄存器	80
0x0004	GPIOx_DIR	R/W	0x0000_FFFF	GPIOx 数据方向寄存器	80
0x0008	GPIOx_SRT	WO	0x0000_0000	GPIOx 设置/复位寄存器	81
0x000C	GPIOx_DBC	R/W	0x0000_0000	GPIOx 去抖动计数寄存器	81
0x0010	GPIOx_IPR	R/W	0x0000_0000	GPIOx 中断优先级寄存器	82
0x0014	GPIOx_ISR	R/W	0x0000_0000	GPIOx 中断触发寄存器	82
0x0018	GPIOx_IER	WO	0x0000_0000	GPIOx 中断允许寄存器	83
0x001C	GPIOx_IDR	WO	0x0000_0000	GPIOx 中断禁用寄存器	83
0x0020	GPIOx_IMS	RO	0x0000_0000	GPIOx 中断屏蔽状态寄存器	84
0x0024	GPIOx_RIS	RO	0x0000_0000	GPIOx 初始中段站状态寄存器	84
0x0028	GPIOx_ISC	RW/1C	0x0000_0000	GPIOx 中断屏蔽状态和中断清除寄存器	85
0x0030	GPIOx_PUDR	R/W	0x0000_0000	GPIOx 端口上拉/下拉寄存器	85
0x0034	GPIOx_OTSR	R/W	0x0000_0000	GPIOx 端口输出类型和驱动电流选择寄存器	86
0x0038	GPIOx_AFRL	R/W	0x0000_0000	GPIOx 低位复用功能寄存器	86
0x003C	GPIOx_AFRH	R/W	0x0000_0000	GPIOx 高位复用功能寄存器	87
0x0040	GPIOx_ADTRIG	R/W	0x0000_0000	GPIOx ACD触发寄存器	87

4.6.3.1 GPIOx_DIO - GPIOx 输入输出寄存器

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

偏移地址: 0x0000

位	名称	类型	复位值	描述
31:16	保留	RO	0x0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
15:0	DATA	R/W	Read: 0x03FF Write: 0xFFFF	GPIO 数据 当GPIO端口被设置成输出时数据输出； 当GPIO端口被设置成输入时数据输入

4.6.3.2 GPIOx_DIR - GPIOx 数据方向寄存器

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

偏移地址: 0x0004

位	名称	类型	复位值	描述
31:16	保留	RO	0x0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
15:0	DIR	R/W	0xFFFF	GPIO 方向选择 向此寄存器写入值来独立控制相应端口位的数据方向。 0: 输出 1: 输入(默认)

4.6.3.3 GPIOx_SRT - GPIOx 设置/复位寄存器

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

偏移地址: 0x0008

位	名称	类型	复位值	描述
31:16	RESET	WO	0x0	端口x复位 0: 无作用 1: 复位相应引脚输出数据 如果设置/复位被同时置'1', 强制复位。
15:0	SET	WO	0x0	端口设置位 0: 无作用 1: 设置相应引脚的输出数据

4.6.3.4 GPIOx_DBC - GPIOx 去抖计数寄存器

DBEN [x] 位用来启用每个相应的 GPIO 输入去抖功能。如果输入的脉冲宽度不能被连续的"SPT"周期采样, 输入的信号过渡看作是去抖信号, 并将不触发中断和信号状态的改变。所需要的采样周期是由" SPT "控制。例如: 如果启用去抖功能, 需设置 $SPT = 5$, $FILTCNT = 1$ 。所以, 这意味着每六个系统时钟采样一次, 如果两次采样值是一样的, 它将输出这个值。

推导公式 (系统频率/ $SPT / FILTCNT$) = 最大数据传输有效频率。例如, 如果系统时钟是 48 MHz, 需要输出的频率是 4 MHz($48/6/2 = 4$ MHz)以下。如果 SPT 和 $FILTCNT$ 值最大, 则输入数据的有效频率是 23.437 kHz ($48/256/8 = 23.437$ kHz)。

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

偏移地址: 0x000C

位	名称	类型	复位值	描述
31:16	DBEN	R/W	0x0	GPIO 防抖使能 可以单独控制每个引脚的去抖
15:8	SPT	R/W	0x0	GPIO去抖周期控制 如果启用了去抖功能, 则必须设置某个值大于0
7:3	保留	RO	0x0	保留位。考虑与其他产品的兼容性, 此值不应该被写入或读出。
2:0	FILTCNT	R/W	0x0	GPIO去抖计数

4.6.3.5 GPIOx_IPR - GPIOx 中断优先级寄存器

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

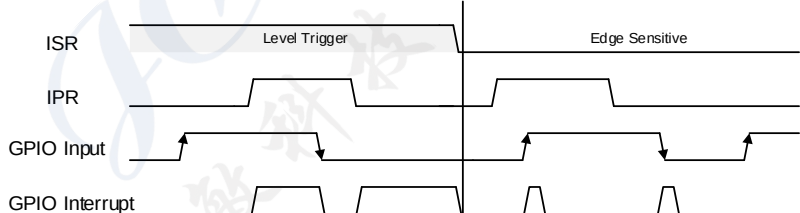
偏移地址: 0x0010

位	名称	类型	复位值	描述
31:16	IBR	R/W	0x0	0: 根据IPR位产生中断触发 1: 检测上升沿和下降沿产生中断触发
15:0	IPR	R/W	0x0	中断优先级 当输入数据是高电平或低电平时GPIOx中断被触发。 端口(X)的输入受边沿的极性或者电平高低触发。当寄存器某一位置‘0’时，相应的被配置成下降沿触发或者低电平触发；否则，为上升沿触发或高电平触发。 0: 低电平有效(默认) 1: 高电平有效

4.6.3.6 GPIOx_ISR - GPIOx 中断触发类型寄存器

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

偏移地址: 0x0014

位	名称	类型	复位	描述
31:16	保留	RO	0x0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
15:0	ISR	R/W	0x0	GPIOx 中断触发类型选择 控制端口(X)的中断触发类型。往此寄存器位写‘1’，它将被配置成电平触发，否则被配置为边沿触发。 0: 边沿触发 (默认) 1: 电平触发 

4.6.3.7 GPIOx_IER - GPIOx 中断使能寄存器

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

偏移地址: 0x0018

位	名称	类型	复位值	描述
31:16	保留	RO	0x0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
15:0	IE	WO	0x0	GPIOx中断使能 0：配置Port (X) 位作为普通GPIO (默认) 1：使能Port (X) 位作为中断

4.6.3.8 GPIOx_IDR - GPIOx 中断禁用寄存器

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

偏移地址: 0x001C

位	名称	类型	复位值	描述
31:16	保留	RO	0x0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
15:0	ID	WO	0x0	GPIOx 中断禁用 0：配置Port (X) 位作为普通GPIO (默认) 1：禁止Port (X) 位作为中断

4.6.3.9 GPIOx_IMR - GPIOx 中断屏蔽状态寄存器

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

偏移地址: 0x0020

位	名称	类型	复位值	描述
31:16	保留	RO	0x0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
15:0	IMS	RO	0x0	GPIOx 中断屏蔽状态 0: 相应引脚中断被屏蔽 1: 相应引脚中断未被屏蔽

4.6.3.10 GPIOx_RIS - GPIOx 初始中断状态寄存器

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

偏移地址: 0x0024

位	名称	类型	复位值	描述
31:16	保留	RO	0x0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
15:0	RIS	RO	0x0	GPIOx 中断初始状态 0: 允许相应引脚的中断请求 1: 禁止相应引脚的中断请求



4.6.3.11 GPIOx_ISC - GPIOx 中断状态和中断清除寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 R/W1C	14 R/W1C	13 R/W1C	12 R/W1C	11 R/W1C	10 R/W1C	9 R/W1C	8 R/W1C	7 R/W1C	6 R/W1C	5 R/W1C	4 R/W1C	3 R/W1C	2 R/W1C	1 R/W1C	0 R/W1C
ISC															

偏移地址: 0x0028

位	名称	类型	复位值	描述
31:16	保留	RO	0x0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
15:0	ISC	RO	0x0	GPIOx 中断状态和清除 0：相应引脚中断未启用 1：相应引脚中断启用 置‘1’来清除中断

4.6.3.12 GPIOx_PUDR - GPIOx 上拉/下拉寄存器

31 R/W	30 R/W	29 R/W	28 R/W	27 R/W	26 R/W	25 R/W	24 R/W	23 R/W	22 R/W	21 R/W	20 R/W	19 R/W	18 R/W	17 R/W	16 R/W
PUDR15	PUDR14	PUDR13	PUDR12	PUDR11	PUDR10	PUDR09	PUDR08								
15 R/W	14 R/W	13 R/W	12 R/W	11 R/W	10 R/W	9 R/W	8 R/W	7 R/W	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
PUDR07	PUDR06	PUDR05	PUDR04	PUDR03	PUDR02	PUDR01	PUDR00								

偏移地址: 0x0030

位	名称	类型	复位值	描述
31:30	PUDR15	R/W	0x0	GPIOx 上拉/下拉寄存器 00：浮空 01：上拉 10：下拉 11：保留浮空
29:28	PUDR14	R/W	0x0	
27:26	PUDR13	R/W	0x0	
25:24	PUDR12	R/W	0x0	
23:22	PUDR11	R/W	0x0	
21:20	PUDR10	R/W	0x0	
19:18	PUDR09	R/W	0x0	
17:16	PUDR08	R/W	0x0	
15:14	PUDR07	R/W	0x0	
13:12	PUDR06	R/W	0x0	
11:10	PUDR05	R/W	0x0	
9:8	PUDR04	R/W	0x0	
7:6	PUDR03	R/W	0x0	
5:4	PUDR02	R/W	0x0	
3:2	PUDR01	R/W	0x0	
1:0	PUDR00	R/W	0x0	

4.6.3.13 GPIOx_OTDR - GPIOx 端口输出类型和驱动电流选择寄存器

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

偏移地址: 0x0034

位	名称	类型	复位值	描述
31:16	DSR	R/W	0x0	GPIOx 输出驱动电流选择寄存器 0 : 2mA 1 : 4mA
15:0	OTR	R/W	0x0	GPIOx 输出类型选择寄存器 0 : 推挽输出 1 : 开漏输出

4.6.3.14 GPIOx_AFR1 - 复用功能低位寄存器

GPIO 提供多种 IO 功能选项，通过 GPIOx_AFR1 和 GPIOx_AFRH 寄存器来选择功能。例如，引脚 PA00 默认的模式是 GPIO，若要把它配置成 AD0_CAIP0(ADCIN0 & 比较器 0 正 入)，那么需要将寄存器 GPIOA_AFR0 的值 置为 8。

了解更多关于 PT32U301 复用功能请 考 Signal Description 和 Pin Defintionand Multiplexing。

31 R/W	30 R/W	29 R/W	28 R/W	27 R/W	26 R/W	25 R/W	24 R/W	23 R/W	22 R/W	21 R/W	20 R/W	19 R/W	18 R/W	17 R/W	16 R/W
PUDR15		PUDR14		PUDR13		PUDR12		PUDR11		PUDR10		PUDR09		PUDR08	
15 R/W	14 R/W	13 R/W	12 R/W	11 R/W	10 R/W	9 R/W	8 R/W	7 R/W	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
PUDR07		PUDR06		PUDR05		PUDR04		PUDR03		PUDR02		PUDR01		PUDR00	

偏移地址: 0x0038

位	名称	类型	复位值	描述
31:28	AFR7	R/W	0x0	端口x引脚(y = 0...7)选择复用功能 0: AF0 1: AF1 2: AF2 3: AF3 4: AF4 8: AF_ANA(模拟)
27:24	AFR6	R/W	0x0	
23:20	AFR5	R/W	0x0	
19:16	AFR4	R/W	0x0	
15:12	AFR3	R/W	0x0	
11:8	AFR2	R/W	0x0	
7:4	AFR1	R/W	0x0	
3:0	AFR0	R/W	0x0	

4.6.3.15 GPIOx_AFRH - 复用功能高位寄存器

31 R/W	30 R/W	29 R/W	28 R/W	27 R/W	26 R/W	25 R/W	24 R/W	23 R/W	22 R/W	21 R/W	20 R/W	19 R/W	18 R/W	17 R/W	16 R/W
AFR15				AFR14				AFR13				AFR12			
15 R/W	14 R/W	13 R/W	12 R/W	11 R/W	10 R/W	9 R/W	8 R/W	7 R/W	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
AFR11				AFR10				AFR9				AFR8			

偏移地址: 0x003C

位	名称	类型	复位值	描述
31:28	AFR15	R/W	0x0	端口x引脚(y = 8...15)选择复用功能 0: AF0 1: AF1 2: AF2 3: AF3 4: AF4 8: AF_ANA(模拟)
27:24	AFR14	R/W	0x0	
23:20	AFR13	R/W	0x0	
19:16	AFR12	R/W	0x0	
15:12	AFR11	R/W	0x0	
11:8	AFR10	R/W	0x0	
7:4	AFR9	R/W	0x0	
3:0	AFR8	R/W	0x0	

4.6.3.16 GPIOx_ADTRIG - GPIOx ADC 触发寄存器

为用 提供选择打开 GPIOx 端口 ADC 采 中断触 模 。

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

偏移地址: 0x0040

位	名称	类型	复位值	描述
31:16	保留	RO	0x0	保留位。考虑与其他产品的兼容性，此值不应该被写入或读出。
15:0	ADCTRIG	R/W	0x0	GPIOx ADC 触发源使能 0: 禁止端口(X) 位触发 (默认) 1: 使能端口 (X) 位为ADC触发源

4.7 通用异步收发器 (UART)

该 UART 控制器提供两通道通用异步接收器/发送器 (UART)。UART 支持流控制功能。在接收端, UART 将从外围设备接收的数据做串-并行转换;在发送端,将对 cpu 的数据做并-串转换。该 UART 控制器同时支持 IrDA SIR, 主/从模式和 RS-485 模式功能。

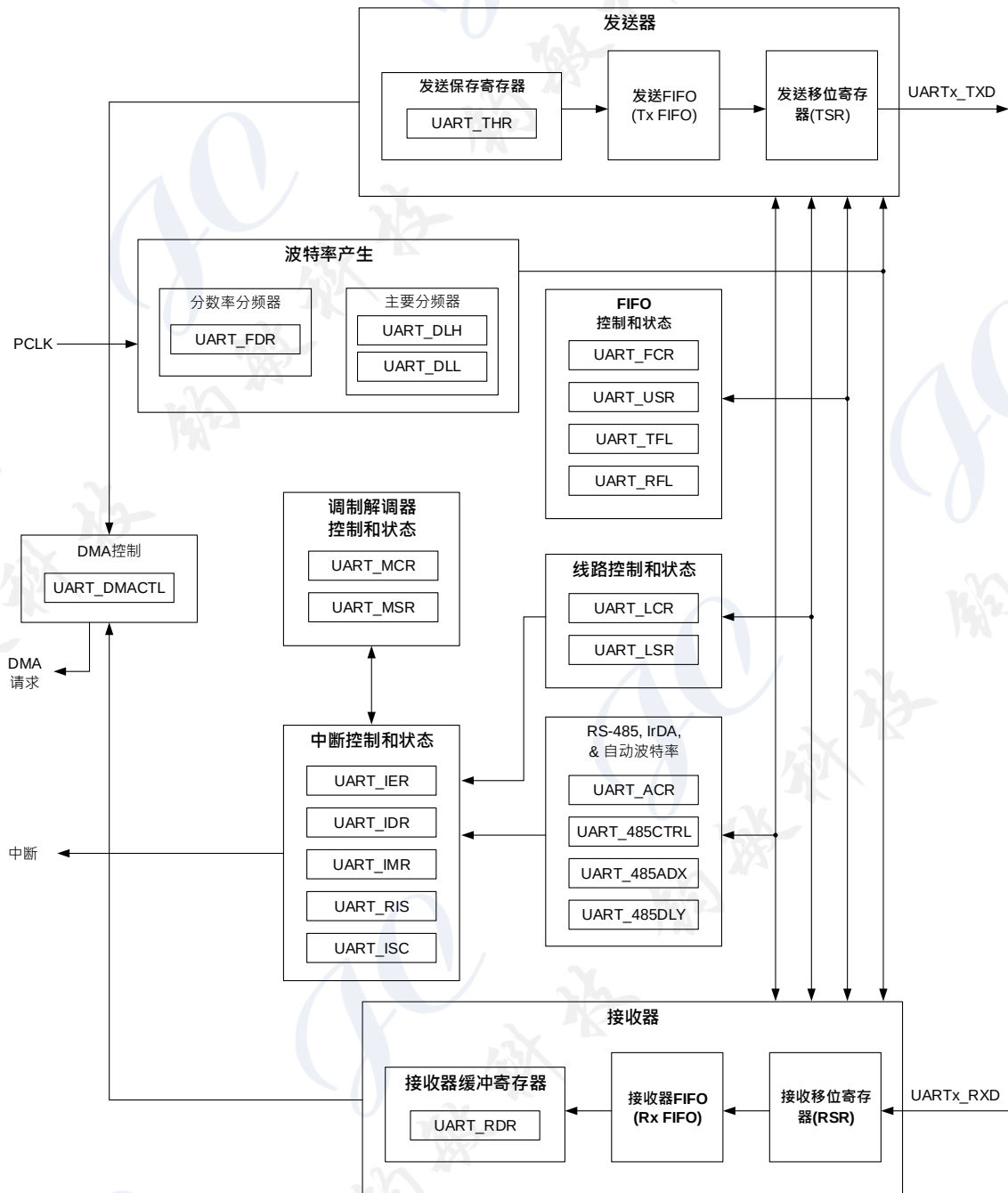
- 全双工异步通信
- 16B 接收和发送 FIFO
- 兼容 16C550 标准
- 可编程接收 Buffer 触发点
- 支持各通道独立波特率可编程
- 支持自动波特率检测
- 八个中断源
- 内置波特率发生器, 覆盖范围广的波特率不需要特定值的外部晶体
- 支持硬件自动流控制/流控制功能 (CTSn, RTSn), RTSn 控制流触发点可编程
- 支持 CTSn 唤醒功能
- 支持 IrDA SIR 模式
 - 支持 3/16 bit 周期调制
- 支持 RS-485
 - RS-485 9-bit 模式
 - 支持硬件或者软件控制 RTSn 或者软件控制 GPIO 转换检测
- 完全可编程的串行接口特性
 - 可编程数据位个数, 即 5-, 6-, 7-, 8- bit
 - 校验位, 奇、偶、无校验或者固定校验位生成和检测可编程
 - 停止位长度可编程: 1, 1.5, 2bit

4.7.1 块图

接收區塊從串行輸入線監測有效輸入。接收器移位寄存器 (RSR) 從 UARTx_RXD 接收有效字符。這些字符會在 RSR 組裝起來，並傳送到 UART 接收緩衝寄存器，等待 CPU 或主機去訪問。

傳送區塊會接收 CPU 或主機所寫的數據並緩衝到 UART 發送保存寄存器。發送器移位寄存器 (TSR)會讀存在 UART_THR 的數據，並把數據組裝後再經 UARTx_TXD 發送。

图 4.7-1: UART 框图



4.7.2 UART 功能描述

4.7.2.1 发送及接收逻辑

发送逻辑执行并行对数据从发送 FIFO 读取串口的转换。控制逻辑输出的串行比特流从一个起始位，其次是数据位（LSB 在前），奇偶校验位，并根据程序的配置在线路控制寄存器的停止位。接收逻辑后有效的启动脉冲被发现进行串并转换的受流。

4.7.2.2 自动波特率

在 UART 自动波特率检测可用于测量基于“AT”协议（Hayes 命令）进入的波特率。如果自动波特率功能启用时，控制器将测量接收数据流的位时间，并设置除数锁存注册相应的 **UART_DLL** 和 **UART_DLH**。

自动波特率检测通过设置 **UART_ACR** 开始起始位和清除该位停止。一旦自动波特率结束并且读该位将返回自动波特率的状态的 **START** 位将被清零（等待/完成）。自动波特率检测的自启动支持如果 **UART_ACR - ATSTART** 位设置启动。如果第一次检测失败，控制器会自动检测波特率一次。

这可以通过 **UART_ACR - MODE** 位选择两种自动波特率测量模式可供选择。

- 模式 0 - 波特率在 UART 的 Rx 管脚的两个连续的下降沿上测量（起始位的下降沿和最低显着位的下降沿（LSB））。
- 模式 1 - 波特率下降沿 UART 的 Rx 引脚（起始位的长度）的后续的上升沿之间测量。

UART_ACR - ATSTART 位可以使用，如果超时出现（速率测量计数器溢出）自动重新启动波特率检测。如果该位被设置，速率测量将重新启动在 UART Rx 管脚的下一个下降沿。

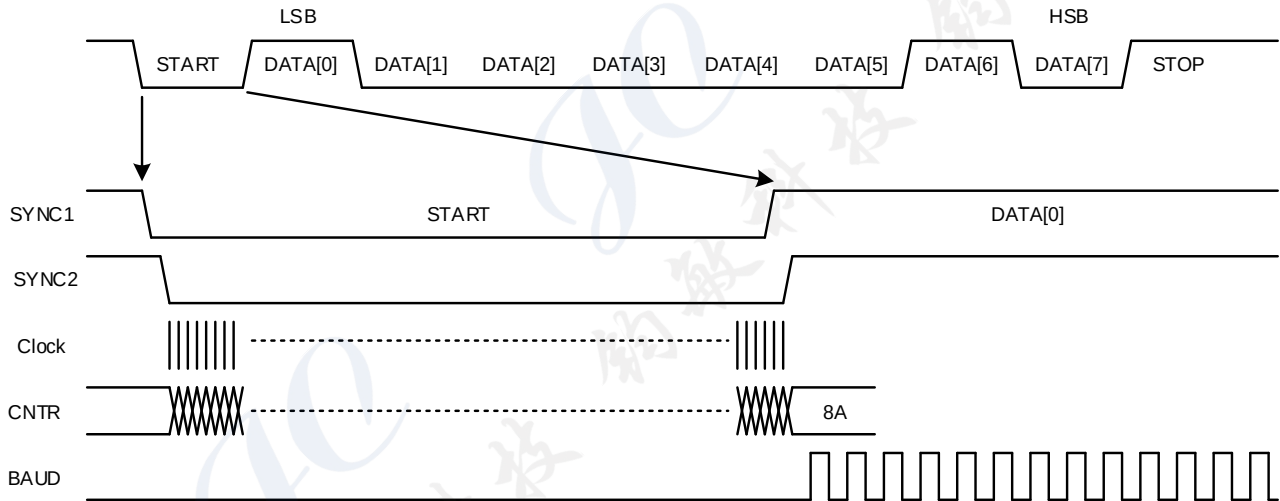
自动波特率功能可产生两个中断。

- 自动波特率检测超时中断（在 **UART_IER** 表示为 **ABTOIE**）将设置是否自动波特率测量计数器溢出）。
- 如果自动波特率成功完成自动波特率检测结束中断（在 **UART_IER** 表示为 **ABEIE**）将设置。

当程式在等待一个“AT”命令，它会将 UART 配置为符合预期字符的格式并设置 **UART_ACR - START**。不管储存在分频锁存器 **UART_DLH** 和 **UART_DLL** 的初始值。因为“A”或“a”的 ASCII 编码（“A”=0x41,“a”=0x61）时，USART0 Rx 管脚检测起始位和 LSB 的期望字符由两个下降沿去界定。当 ACR 启动位被设定时，自动波特率协议将执行以下几个阶段：

1. 当 **UART_ACR-Start bit** 被设定时，波特率测量计数器和 UART 接收缓冲寄存器(**UART_RBR**)会被重置。
2. 当 UART Rx 管脚偵查到代表 **START** 位帶來的下降沿，波特率測量計數器會以 UART 0/1 的 PCLK 為時間基底去計數波特率週期。UART PCLK 的設定請參考 **SC_GCLK_APB**。
2. 選擇模式 0 時，波特率測量計數器會在下一個下降沿(結束沿)暫停。
選擇模式 1 時，波特率測量計數器會在下一個上升沿(結束沿)暫停。
4. 當偵測到結束沿時，波特率測量計數器將計數值載入 **UART_DLH** 或 **UART_DLL** 寄存器，波特率測量計數器會轉為正常模式。當數值被載入後，如果 **ABEIE** 中斷發生，UART 便開始接收其他資料位元。

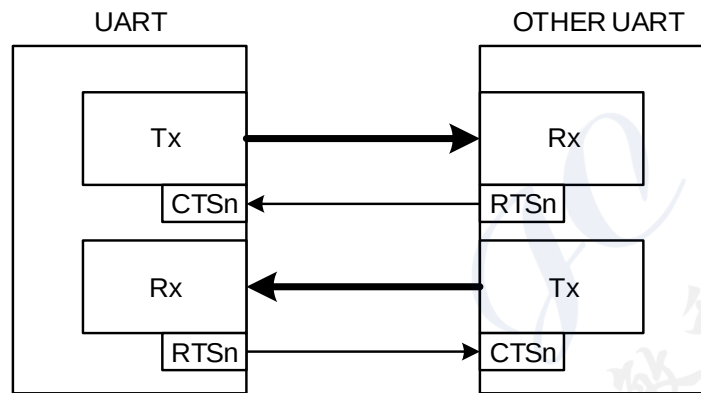
图 4.7-2：只有起始位用于自动波特率



4.7.2.3 自动流控制

如果使能自动流控制，接收 FIFO 和发送 FIFO 會通過 **UARTx_RTSn** 和 **UARTx_CTSn** 引脚去控制 UART 的接收(Rx)和发送(Tx)。

图 4.7-3：自动流量控制连接图形



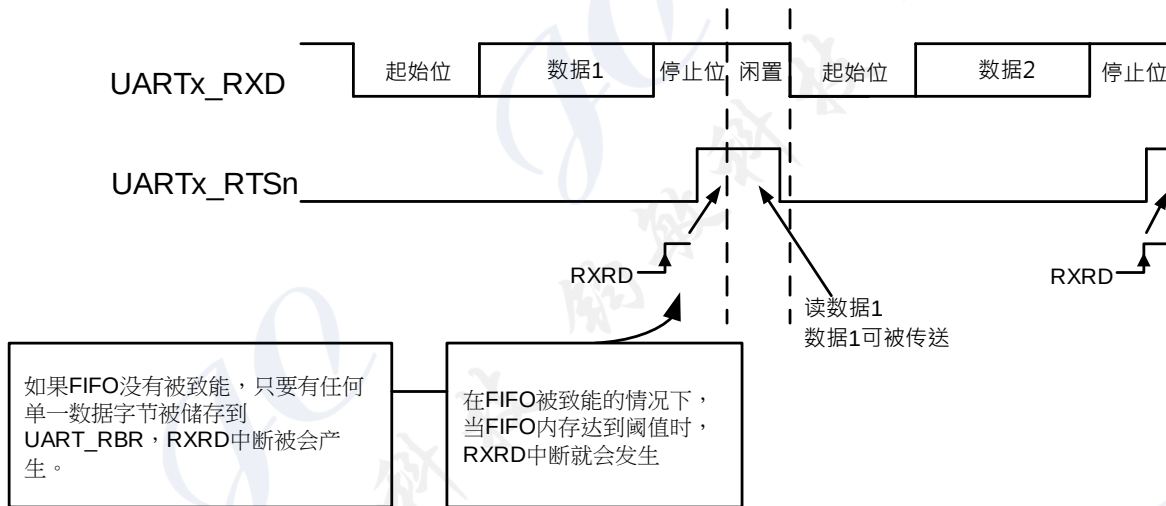
AUTO-RTS

当 Auto-RTS 被使能時，接收 FIFO 达到由 **UART_FCR** 寄存器设置的阈值，**UART_RTSn** 输出會拉為高电平。当 **UART_RTSn** 连接到另一个 UART 设备的 **UART_CTSn** 输入，另一個 UART 會停止发送串行数据，直到接收 FIFO 完全是空的。

可选择接收 FIFO 阈值的值是：1，1/4，1/2 或“2 不全”。一个额外的字符有可能會在 **UART_RTSn** 成为无效後传送到 UART（由于已经在其他 UART 进入发送器的数据块），阈值设置为“2 不全”能用一个字符的安全帶使 用 FIFO 最大。

通过读取接收缓冲寄存器 **UART_RBR**，一旦得知接收 FIFO 完全是空的時候，**UART_RTSn** 变低電平，發信通知其他 UART 继续发送数据。

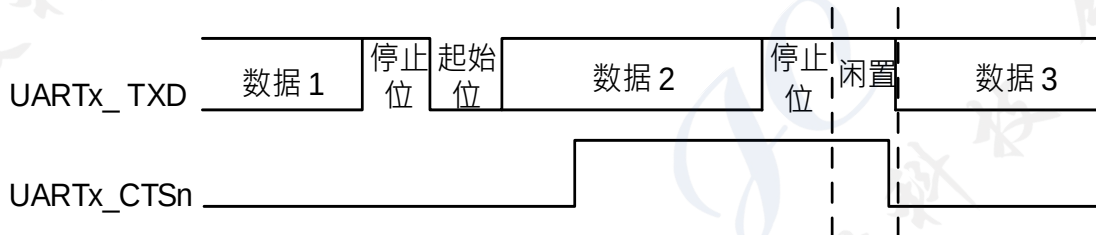
图 4.7-4：自动 RTS 定时



AUTO-CTS

当自动 CTS 启用（活动），每当 UART_CTSn 输入变高时，UART 发送器被禁用。这可以防止数据溢出接收 UART 的 FIFO。在最后一个停止位发出时，假设 UART_CTSn 亦未失活。如果 UART_CTSn 输入不，发射器会在禁用前继续传输一个字符。在发射器被禁用时，发送 FIFO 仍然可以被写入，甚至溢出。

图 4.7-5：Auto-CTS 功能



自动流控制可以减少系统中断，当自动流控制使能，CTSn 状态不会触发系统中断。因为是设备自动地控制其收发器。当没启用自动流控制，收发器将发送的信息存储到 Tx FIFO，产生接收溢出。

4.7.2.4 中斷控制

UART 满足下列条件时，能够产生中断：

- 自动波特率检测超时 - 自动波特率检测超时发生。
- 自动波特率检测结束 - 自动波特率检测完成。
- UART 忙碌 - 主机已经尝试写入线路控制，而 UART 忙。
- 字符超时 - 在过去的 4 个字符时间在 RCVR FIFO 的进出没有字符，还有在此期间它至少有 1 个字符。
- 调制解调器的状态 - 清除发送或数据集准备就绪环指标或数据载波检测。
- Rx 线状态 - 溢出/校验/帧错误或间隔中断。
- Tx 空 - 发射器保持寄存器为空或 XMIT FIFO 处于或低于阈值。
- 接收资料 - 接收器可用数据 RCVR FIFO 触发水平达到。

UART 的中断是由一组 5 个寄存器控制。

- 中断控制（IER，IDR，IMR）
UART 中断使能寄存器（**UART_IER**）通过写“1”使能中断请求线。同样，UART 中断禁止寄存器（**UART_IDR**）通过写入“1”禁止中断请求线。**IER** 和 **IDR** 是控制中断屏蔽只写寄存器。这两个寄存器的总体结果可以通过 **UART** 中断显示屏蔽寄存器（**UART_IMR**）。**IMR** 是只读使用‘1’或‘0’来指示中断请求线被使能/或禁用寄存器。
- 中断状态读取（RIS）
UART 原始中断状态（**UART RIS**）是一个只读寄存器读取所有中断模块的状态。
- 中断清除（ISC）
UART 中断状态和中断清除寄存器（**UART ISC**）用于指示模块的不可屏蔽的中断状态，因为只有现在屏蔽的中断被认定，以处理器。该寄存器写“1”到该位可以清除相应的中断状态或写 1 **IDR** 禁止中断。

4.7.2.5 FIFO 操作

UART 可被配置兩個 16 字節的 FIFO 去發送或接收數據。如果 FIFO 功能沒有被致能的話，則 UART 沒有任何 FIFO 支援而且 UART 接收緩沖寄存器(UART_RBR)和 UART 發送保存寄存器(UART_THR)只會每次接收/傳送一個字節的數據。

令 FIFO 產生中斷的觸發點是通過 UART FIFO 控制寄存器(UART_FCR)去設定的。兩個 FIFO 可分別配置為不同的觸發深度。可選的觸發深度包括 1, $\frac{1}{4}$, $\frac{1}{2}$ or “2 不全”。舉例來說，如果傳送 FIFO 選擇了 $\frac{1}{4}$ 的深度作為觸發點，則當 UART 連續傳送 4 個數據字節後便會產生一個傳送中斷。

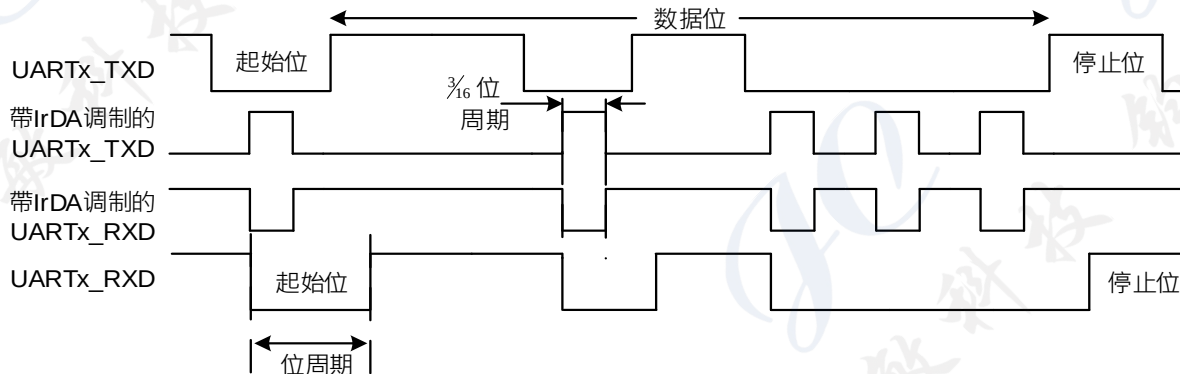
4.7.2.6 IrDA SIR 功能

IrDA SIR 模式以外線作為中介，支援與遠程設備間的雙向數據傳送。其數據格式是类似于标准的串行的数据格式。每个数据字符被串行地发送，由起始位開始，然后是 8 个位元的数据，并由至少一个停止位去結束。因此，可以发送的数据位的数目是固定的。在该模式中，不提供任何奇偶信息，只有一个停止位被使用。

试图调整发送数据的比特数或启用与 UART 线控制寄存器 (UART_LCR) 奇偶並不會起任何作用。当 UART 要被配置为支持的 IrDA1.0 SIR 時，可以通過致能 UART 模式控制寄存器 (UART_MCR) 的位 6—SIRE。当 UART 未配置为支持的 IrDA SIR 模式，其邏輯電路不會執行和其模式不能被激活，從而降低总逻辑门数。SIR 模式在啟用和執行時候，其模組會使用 UARTx_TXD 和 UARTx_RXD 引腳去執行 SIR 協定。這些信號應連接到紅外線收發器去實行 IrDA SIR 物理層連結。

发送单个红外脉冲信號標誌為逻辑零，而不发送一个脉冲表示為逻辑 1。输出脚上的逻辑 0 是一个宽度为 $\frac{3}{16}$ 位周期（按照选定的波特率）的高脉冲；逻辑 1 则以静态的低电平输出。这些电平控制红外发送器的驱动装置，逢 0 电平便发送光脉冲。在接收端，接收到的光脉冲给接收器的光敏晶体管基极加电，将其输出拉至低电平并将 UART 输入管脚置为低电平。为相比，下标时序图显示了含有 IrDA 调制和不含 IrDA 调制时的 UART 发送和接收信号。

图 4.7-6: IrDA SIR 数据格式



IrDA SIR 物理层协议制定了半双工的通信链路标准，其中规定发送和接收之间至少间隔 10ms 的延时。UART 模块本身并不提供此延时，必须由用户的应用软件予以保障。

4.7.2.7 RS-485 功能

除了標準的發送及接收數據線以外，UART 同時支援 RS-285/9 位元模式。

自動 9 位模式

UART 支援 9 位 RS-485 模式。。此功能在 UART 的多分支配置中非常有用。可被配置為可被定址的从机。单个主机可以控制多个相连的从机，每個从机都稱為可被定址的从机。在这种配置下，主机傳送一個地址字符和跟隨其後的數據字符到从机。从機會檢驗收到的數據，如果檢測到地址字符（第 9 位=校驗位=1），便會對控制器發出中斷。

每个 UART 从机都被分配了一个唯一的地址，从机可以被设置成根据地址信息手动或者自动地丢弃数据。控制器支援以下三種模式：

RS-485/EIA-485 多点通信模式(NMM)

設置 **UART_485CTRL.NMMEN** 使能此模式。在此模式下，地址的检测是接收到一个字节导致 UART 校驗错并且产生中斷。如果接收器被禁止(**UART_485CTRL.RXDIS = '1'**)，接收的数据会被忽略也不会存儲进 RX FIFO。当检测到地址字段（校驗位为 1），接收的数据会进入 RX FIFO，且 Rx Data Ready 中斷会产生。处理器会读取地址 段并决定收发器是否使能接受后续数据。当接收器被使能(**UART_485CTRL.RXDIS = '0'**)，所收到的字段不论数据还是地址都会被接受且存进 RX FIFO。当收到地址段产生校驗财务中斷，处理器会确定是否要禁止接收器。

RS-485/EIA-485 自动地址检测 (AAD)模式

当 **UART_485CTRL.NMMEN** (9-bit 模式使能) 和 **AADEN** (AAD 模式使能) 都被置位，UART 就工作在自动地址检测模式。在此模式下接收器会比较接收到的地址段（校驗位为 '1'）的信息和 **UART_485ADX** 寄存器的 8bit 值，如果接收器被禁止 (**UART_485CTRL.RXDIS = '1'**)，或者与 **UART_485ADX** 地址匹配失败，所有收到的数据都被丢弃。如果地址字段匹配成功，同时它会被存入 RX FIFO，且产生 Rx Data Ready 中斷。处理器会读取地址字段并且决定是否使能接收器接受后续数据。当接收器被使能(**UART_485CTRL.RXDIS = '0'**)，所有被接收到的数据都会被存入 RXFIFO，直到接收器收到的地址段与 **UART_485ADX** 的值失配，此时，收到的数据将被丢弃。没有匹配上的地址字段不会存入 RXFIFO。

RS-485/EIA-485 自动方向控制

RS485 / EIA-485 模式包括自动控制选中的 **RTSn/CTSn** 引腳发送输出状态指示信号。这个功能在 **UART_485CTRL.DIRCTRL** 置 "1" 时使能。当 **UART_485CTRL.OINV = 0**，则用 **RTSn**。当 **UART_485CTRL.OINV = 1**，则用 **CTSn**。当自动指示控制使能，选中的引腳在 CPU 写数据到 TXFIFO 时将被拉低。一旦最后的数据位被发送，选中的引腳将被拉高。

4.7.3 UART 寄存器映射

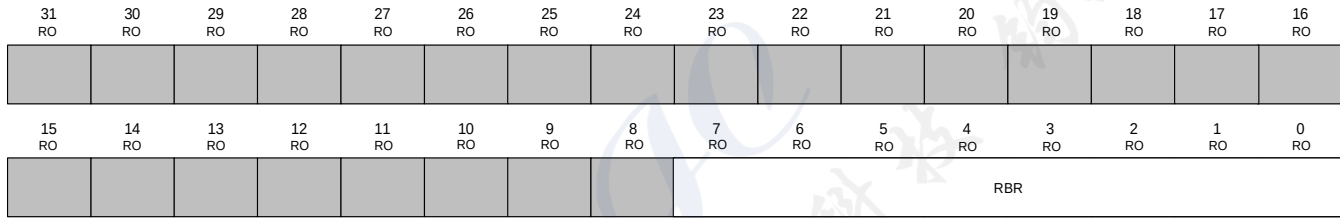
UART 基地址:

- UART 0: 0x4800_0000
- UART 1: 0x4800_0100

偏移地址	寄存器名称	读/写	默认值	描述	参考页
0x0000	UART_RBR	RO	0x0000_0000	UART接收缓冲器寄存器	97
0x0000	UART_THR	WO	0x0000_0000	UART发射保存寄存器	97
0x0000	UART_DLL	R/W	0x0000_0000	UART分频锁存器(Low)	98
0x0004	UART_DLH	R/W	0x0000_0000	UART分频锁存器(High)	98
0x0004	UART_FCR	R/W	0x0000_0000	UART FIFO控制寄存器	99
0x0008	UART_ACR	R/W	0x0000_0000	UART自动波特率控制寄存器	100
0x000C	UART_FDR	R/W	0x0000_0000	UART小数分配器	101
0x0010	UART_LCR	R/W	0x0000_0000	UART Line控制寄存器	102
0x0014	UART_MCR	R/W	0x0000_0000	UART Modem控制寄存器	103
0x0018	UART_LSR	RO	0x0000_0060	UART Line状态寄存器	104
0x001C	UART_MSR	RO	0x0000_0000	UART Modem状态寄存器	105
0x0020	UART_IER	WO	0x0000_0000	UART中断使能寄存器	106
0x0024	UART_IDR	WO	0x0000_0000	UART中断禁止寄存器	107
0x0028	UART_IMR	RO	0x0000_0000	UART中断屏蔽寄存器	108
0x002C	UART_RIS	RO	0x0000_0002	UART原始中断状态寄存器	109
0x0030	UART_ISC	R/W1C	0x0000_0000	UART中断状态和清除寄存器	110
0x0034	UART_485CTRL	R/W	0x0000_0000	UART RS485控制寄存器	111
0x0038	UART_485ADX	R/W	0x0000_0000	UART RS485地址匹配寄存器	112
0x003C	UART_485DLY	R/W	0x0000_0000	UART RS485方向延时寄存器	112
0x007C	UART_USR	RO	0x0000_0006	UART 状态寄存器	113
0x0080	UART_TFL	RO	0x0000_0000	UART发送水平寄存器	114
0x0084	UART_RFL	RO	0x0000_0000	UART接收水平寄存器	114



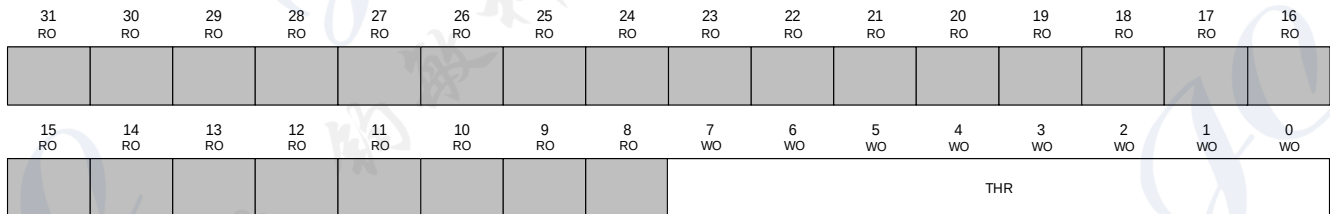
4.7.3.1 UART_RBR - UART 接收缓冲寄存器



偏移地址: 0x0000

域	名称	读/写	默认值	描述
31:8	保留	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
7:0	RBR	RO	0x0	接收器缓冲寄存器，通过阅读这个寄存器，UART将返回从RX引脚（LSB first）的8位数据接收。

4.7.3.2 UART_THR - UART 发送保存寄存器



偏移地址: 0x0000

域	名称	读/写	默认值	描述
31:8	保留	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
7:0	THR	WO	0x0	接收器缓冲寄存器 通过写这个寄存器，UART将发出来自TX引脚（LSB第一）的8位数据接收。

4.7.3.3 UART_DLL - UART 分频器

UART_DLL 和 UART_DLH 寄存器一起儲存一個 16 位除数去產生波特時鐘，這 UART_DLL 包含低 8 的除数和 UART_DLH 含有高八。如果你想訪問 UART 的除数鎖存寄存器，注意 UART_DLH 和 UART_DLL 的訪問條件是 UART_LCR.DLAB 必須是 1 和 UART 不是處於忙狀態，UART_USR.BUSY = 0。

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

偏移地址: 0x0000

域	名称	读/写	默认值	描述
31:8	保留	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
7:0	DLLSB	R/W	0x0	UART的除数锁存器LSB 与UART DLH寄存器相关决定UART波特率。

4.7.3.4 UART_DLH - UART 锁存器

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

偏移地址: 0x0004

域	名称	读/写	默认值	描述
31:8	保留	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
7:0	DLHLSB	R/W	0x0	UART除数锁存功能LSB 与UART DLL寄存器来判断UART波特率。

4.7.3.5 UART_FCR - UART FIFO 控制寄存器

這個寄存器可以致能 FIFO，清除 FIFO 和設定 FIFO 觸發等級。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 R/W	6 R/W	5 R/W	4 R/W	3 R/W	2 W1C	1 W1C	0 R/W
								RCVR		TET		PTIME	XFIFOR	XFIFOR	XFIFOR

偏移地址: 0x0004

域	名称	读/写	默认值	描述
31:8	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
7:6	<i>RCVR</i>	WO	0x0	RCVR触发 如下设置产生接收中断(RXRD)的触发深度 00：1個字符 01：1/4满 10：半满 11：2不全
5:4	<i>TET</i>	WO	0x0	UART 发送中断 FIFO 深度选择 发送中断(THRE)的触发深度设置如下。 00：FIFO空 01：2個字符 10：1/4满 11：半满
3	<i>TETES</i>	WO	0	TET有效开关 该位决定Tx空中断（TEMPLE）的产生是否将由TET位被触发。 0：TXEMPIE当FIFO无论TET值是空才会被触发 1：TXEMPIE将在响应被设置为在TET设定值
2	<i>XFIFOR</i>	W1C	0	XMIT FIFO复位 当XFIFOR设置，发送FIFO中的所有字节被清除，对待FIFO为空。注意，该位将在下一时钟周期自动清除。 0：沒有Tx重置 1：重置Tx指针
1	<i>RFIFOR</i>	W1C	0	RCVR FIFO复位 当RFIFOR设置，接收FIFO的所有字节被清除，对待FIFO为空。注意，该位将在下一时钟周期自动清除。 0：沒有Rx重置 1：重置Rx指针。
0	<i>FIFOE</i>	WO	0	FIFO 使能 0：UART FIFO禁用。 1：UART FIFO启用。

4.7.3.6 UART_ACR - UART 自动波特率控制寄存器

在串行传输率测量的用户输入时钟/数据速率，整个测量过程是由 **UART_ACR** 控制。

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

偏移地址: 0x0008

域	名称	读/写	默认值	描述
31:3	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
2	<i>ATSTART</i>	R/W	0	自动波特率检测重启（自动） 该位设置为后用波特率检测重新启动自动波特率检测的第一次失败后的一次。 0：无需重新启动检测 1：重新启动波特率检测第一次超时后（计数器是由一个下降沿重新启动）
1	<i>MODE</i>	R/W	0	自动波特率模式选择（参见自动波特率） 0：模式0 1：模式1
0	<i>START</i>	R/W	0	自动波特率检测开始 自动波特率功能后，该位被自动清零 0：禁止自动波特率检测 1：启用自动波特率检测

4.7.3.7 UART_FDR - UART 小数分频寄存器

UART 小数分频器寄存器（FDR）控制产生波特率时钟分频，并且用户可以自由地读取和写入寄存器。预分频使用 APB 时钟，并根据在十进制中规定的要求产生输出时钟。

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

偏移地址: 0x000C

域	名称	读/写	默认值	描述
31:4	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
3:0	<i>MULVAL</i>	R/W	0x0	预分频乘数值 与 UART_DLL 和 UART DLH 准值，以确定 UART 波特率的小数部分，只可以在 UART_LCR.DLAB = '1' 时被写入。

4.7.3.8 UART_LCR - UART Line 控制寄存器

此寄存器用來指定異步數據通信的格式。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 R/W	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
								DLAB	BC	RXE	PS	PE	STOP	DLS	

偏移地址: 0x0010

域	名称	读/写	默认值	描述
31:8	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
7	<i>DLAB</i>	R/W	0	除数锁存访问位 只有当UART不忙，该位才可写。用于使能除数锁存器（ UART_DLL 和 UART_DLH ）的读取和写入。 0：禁止访问除数锁存器 1：允许访问除数锁存器
6	<i>BC</i>	R/W	0	中断控制位。 这是用来使一个中断条件发送到接收设备。 1：串行数据输出（TX）是被迫的间隔状态（逻辑0）。
5	<i>RXE</i>	R/W	0	UART接收器使能 此位用于使UART接收用户与UART的设置完成之后。 1：启用UART接收器
4	<i>PS</i>	R/W	0	奇偶校验选择 0：奇校验 1：偶校验
3	<i>PE</i>	R/W	0	奇偶校验启用 0：禁止奇偶 1：启用校验
2	<i>STOP</i>	R/W	0	停止位数 这是用来选择每字符的停止比特的数量，外设发送和接收。 0：1个停止位 1：2的停止位（ LCR[1:0] = 00 ，这将是1.5个停止位）
1:0	<i>DLS</i>	R/W	0x0	数据长度选择 00：5位数据格式 01：6比特的数据格式 10：7位数据格式 11：8位数据格式

4.7.3.9 UART_MCR - UART Modem 控制寄存器

MCR 启用调制解调器回环模式，控制调制解调器输出信号。

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

偏移地址: 0x0014

域	名称	读/写	默认值	描述
31:7	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
6	<i>SIRE</i>	WO	0	SIR模式使能 0：禁止的IrDA SIR模式 1：启用红外线SIR模式，在這個模式下，UARTx_TXD和UARTx_RXD被連接到紅外線編碼器/解碼器。
5	<i>ATFLOW</i>	WO	0	自动流控制启用 當致能FIFO和這個位元後，UART便會執行自動流控制，詳見自动流控制。 0：禁止自动流量控制 1：启用自动流量控制
4	<i>LBCTRL</i>	WO	0	回环使能 这是用于将UART为测试目的的诊断模式。如果在UART模式运行，在UARTx_TXD的数据輸出為高，而串行数据环回到UARTx_RXD。在这个模式中，所有的中断都是完整運作的。如果在红外模式下运行，在經IrDA 調制的UARTx_TXD线数据就低，而串行数据輸出反向和环回到UARTx_RXD线。 0：禁用环回模式 1：启用环回模式
3:2	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
1	<i>RTSCTRL</i>	WO	0	RTSN控制 如果ATFLOW =1，则RTSn将由Rx阈值控制。 如果ATFLOW = 0，用户可以通过此位控制RTSn。
0	<i>reserved</i>	RO	0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。

4.7.3.10 UART_LSR - UART Line 状态寄存器

這個寄存器提供了 UART 與 CPU 之間數據傳送的狀態。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
								RFE	TEMT	THRE	BI	FE	PE	OE	DR

偏移地址: 0x0018

域	名称	读/写	默认值	描述
31:8	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
7	<i>RFE</i>	RO	0	接收FIFO错误 这是用来指示是否有至少一个奇偶错误，帧错误，或打破在FIFO指示。 0：RX FIFO 數據没有错误 1：RX FIFO數據發生错误
6	<i>TEMT</i>	RO	1	发送器空 UART_THR 和 TSR都是空的。 0：UART_THR或TSR有字符 1：UART_THR和TSR都是空的 在FIFO模式下，當FIFO和TSR都是空的時候，這個位就會被致能。
5	<i>THRE</i>	RO	1	发送保存寄存器空 UART_THR空。该位表明UART已經準備好接收新的字符作傳送。這個位會觸發一個THR的中斷，在該中斷被允許的情況下。 0：UART_THR 從CPU載入數據 1：有一個數據剛從UART_THR傳送數據到TSR 在FIFO模式下，這個位置會在發送FIFO是空的時候被致能; 有最少一個字符被傳送到發送FIFO時候，這個位會被清除。
4	<i>BI</i>	RO	0	暂停中断 该位用于指示在串行输入数据中断序列的检测。读UART_LSR将清除状态位。 0：间隔中断條件沒有達到 1：接收器接收到中斷信號(UARTx_RXD在一個字符帧時間內都為邏輯0)。在FIFO模式下，只有一個中止字符被載入FIFO。
3	<i>FE</i>	RO	0	帧错误 当接收到的字符停止位为0时，产生帧错误。读取该寄存器清零位。帧错误检测时间取决于UART_FCR.FIFOE。当检测到帧错误，RX将尝试与数据同步，并假设错误的停止位实际是一个超前的起始位並繼續接收其他位。在FIFO模式下，該錯誤與FIFO頂部的字符有關。 0：帧错误状态无效 1：帧错误状态有效
2	<i>PE</i>	RO	0	奇偶校验错误 当所接收的字符沒有正確的校验字节，将生成一个奇偶校验错误。在FIFO模式下，該錯誤與FIFO頂部的字符有關。 0：奇偶校验错误状态无效 1：奇偶校验错误状态有效

域	名称	读/写	默认值	描述
1	OE	RO	0	溢出错误 当收到新的数据字符，假设上一个数据字符并未被读走，便会发生溢出错误。读LSR清除这个位。在FIFO模式下，当FIFO满时，当收到新的数据字符，便会发生溢出错误。FIFO里的数据会保留，UART_RBR的数据会丢失。 0：没有溢出错误 1：有溢出错误
0	DR	RO	0	数据就绪 当RBR包含未读的字符，LSR[0]将被设置;当UART RBR FIFO为空，LSR[0]将被清除。 0：UART_RBR和接收FIFO是空的 1：有数据接收并储存在UART_RBR或是接收FIFO。

4.7.3.11 UART_MSR - UART Modem 状态寄存器

MSR 是只读寄存器，指示 Modem 输入信号的状态。读取 MSR，清除 MSR[0]。注意调制解调器信号不会对 UART 的操作有直接的影响，他们能帮组软件获取调制解调器信号的相关操作。

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

偏移地址: 0x001C

域	名称	读/写	默认值	描述
31:5	reserved	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
4	CTS	RO	0x0	清除发送。 该位是CTSN的补码。当CTS被断言这是一个迹象表明调制解调器和数据设备已准备好与UART进行数据交换。 0：输入CTSN被解除 1：输入CTSN被确认
3:1	reserved	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
0	DCTS	RO	0x0	清除发送。 这是用来指示自从上次在MSR读取的调制解调器控制线CTSN已经改变。 0：不对输入CTS检测调制解调器状态变化 1：对输入的CTS检测调制解调器状态变化

4.7.3.12 UART_IER - UART 中断使能寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 WO	6 WO	5 WO	4 WO	3 WO	2 WO	1 WO	0 WO
								ABTOIE	ABEIE	BZDIE	CHTOIE	MDSIE	RLSIE	TXEMPIE	RXRDIE

偏移地址: 0x0020

域	名称	读/写	默认值	描述
31:8	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
7	<i>ABTOIE</i>	WO	0	自动波特率检测超时中断使能 1：自动波特率检测发生时允许中断。
6	<i>ABEIE</i>	WO	0	UART自动波特率检测结束中断启用 1：中断自动波特率检测功能是结束。
5	<i>BZDIE</i>	WO	0	UART忙中断使能 1：UART中断使能忙。
4	<i>CHTOIE</i>	WO	0	UART字符超时中断使能 1：字符超时发生时允许中断。
3	<i>MDSIE</i>	WO	0	UART Modem状态中断使能 1：中断调制解调器状态启用
2	<i>RLSIE</i>	WO	0	UART Rx线状态中断使能。 1：中断使能接收器线状态
1	<i>TXEMPIE</i>	WO	0	UART TX空中断使能 1：发送保持寄存器允许中断是空的
0	<i>RXRDIE</i>	WO	0	UART接收数据可用的中断使能。 1：中断是用于接收数据可启用。

4.7.3.13 UART_IDR - UART 中断禁止寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 WO	6 WO	5 WO	4 WO	3 WO	2 WO	1 WO	0 WO
								ABTOD	ABEID	BZDID	CHTOD	MDSID	RLSID	TXEMPID	RXRDID

偏移地址: 0x0024

域	名称	读/写	默认值	描述
31:8	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
7	<i>ABTOD</i>	WO	0	自动波特率检测超时中断禁止 1：中断禁止自动波特率检测发生。
6	<i>ABEID</i>	WO	0	UART自动波特率检测结束中断禁用 1：中断自动波特率检测禁用的结束。
5	<i>BZDID</i>	WO	0	UART忙中断禁止 1：中断的UART禁用。
4	<i>CHTOD</i>	WO	0	UART字符超时中断禁止 1：中断禁止字符超时发生。
3	<i>MDSID</i>	WO	0	UART调制解调器状态中断禁用 1：中断为调制解调器状态禁用
2	<i>RLSID</i>	WO	0	UART RX线状态中断禁用。 1：中断是接收器线路状态禁用
1	<i>TXEMPID</i>	WO	0	UART TX空中断禁用 1：中断是用于发送保持寄存器禁用是空的
0	<i>RXRDID</i>	WO	0	UART接收数据可用中断禁止。 1：中断禁止用于接收数据可用。

4.7.3.14 UART_IMR - UART 中断屏蔽状态寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
								ABTOIM	ABEIM	BZDIM	CHTOIM	MDSIM	RLSIM	TXEMPIM	RXRDIM

偏移地址: 0x0028

域	名称	读/写	默认值	描述
31:8	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
7	<i>ABTOIM</i>	RO	0	自动波特率检测超时中断屏蔽状态 0：自动波特率检测超时中断将被屏蔽。 1：自动波特率检测超时中断使能。
6	<i>ABEIM</i>	RO	0	UART自动波特率检测结束中断屏蔽状态 0：自动波特率检测结束中断将被屏蔽。 1：自动波特率检测中断使能。
5	<i>BZDIM</i>	RO	0	UART忙中断屏蔽状态 0：UART忙中断将被屏蔽。 1：UART忙中断使能。
4	<i>CHTOIM</i>	RO	0	UART字符超时中断屏蔽状态 0：UART字符超时中断将被屏蔽。 1：UART字符超时中断使能。
3	<i>MDSIM</i>	RO	0	UART调制解调器状态中断屏蔽状态 0：调制解调器状态中断将被屏蔽。 1：调制解调器状态中断使能。
2	<i>RLSIM</i>	RO	0	UART RX线状态中断屏蔽状态。 0：接收线状态中断将被屏蔽。 1：接收线路状态已启用。
1	<i>TXEMPIM</i>	RO	0	UART TX空中断屏蔽状态 0：TX空中断将被屏蔽。 1：当使能的Tx空中断。
0	<i>RXRDIM</i>	RO	0	UART接收数据可用的中断屏蔽状态。 0：UART接收数据可用的中断将被屏蔽。 1：UART接收数据可用的中断使能。

4.7.3.15 UART_RIS - UART Modem 控制寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
								ABTORI	ABERI	BZDRI	CHTORI	MDSRI	RLSRI	TXEMPRI	RXRDRRI

偏移地址: 0x002C

域	名称	读/写	默认值	描述
31:8	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
7	<i>ABTORI</i>	RO	0	自动波特率检测超时原始中断状态s 0：无中断 1：自动波特率检测超时中断发生
6	<i>ABERI</i>	RO	0	UART自动波特率检测结束原始中断状态 0：无中断 1：自动波特率检测中断发生
5	<i>BZDRI</i>	RO	0	UART忙原始中断状态 0：无中断。 1：发生UART忙中断
4	<i>CHTORI</i>	RO	0	UART字符超时原始中断状态 0：无中断。 1：：发生UART字符超时中断
3	<i>MDSRI</i>	RO	0	UART Modem状态原始中断状态 0：无中断。 1：发生调制解调器状态中断。
2	<i>RLSRI</i>	RO	0	UART RX线路状态原始中断状态 0：无中断。 1：发生Rx线状态。
1	<i>TXEMPRI</i>	RO	1	UART TX空原始中断状态 0：无中断。 1：发生的Tx空中断。
0	<i>RXRDRRI</i>	RO	0	UART接收数据提供原始中断状态 0：无中断。 1：URT接收出现可用的中断数据。

4.7.3.16 UART_ISC - UART 中断状态清除寄存器

注：该寄存器读写以清除注册。“1”个别位的写操作会清除相应的中断状态。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RW1C	6 RW1C	5 RW1C	4 RW1C	3 RW1C	2 RW1C	1 RW1C	0 RW1C
								ABTOIS	ABEIS	BZDIS	CHTOIS	MDSIS	RLSIS	TXEMPIS	RXRDIS

偏移地址: 0x0030

域	名称	读/写	默认值	描述
31:8	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
7	<i>ABTOIS</i>	RW1C	0	自动波特率检测超时中断状态和清除 0：自动波特率检测有没有超时或中断被屏蔽。 1：自动波特率检测超时中断已发出信号。
6	<i>ABEIS</i>	RW1C	0	UART自动波特率检测结束中断状态和清除 0：自动波特率检测还没有结束或中断被屏蔽。 1：自动波特率检测中断已发出信号。
5	<i>BZDIS</i>	RW1C	0	UART忙中断状态和清除 0：UART不忙或中断被屏蔽。 1：UART忙中断已发出信号。
4	<i>CHTOIS</i>	RW1C	0	UART字符超时中断状态和清除 0：UART字符没有超时或中断被屏蔽。 1：UART字符超时中断已发出信号。
3	<i>MDSIS</i>	RW1C	0	UART Modem状态中断状态和清除 0：UART Modem状态没有中断或者中断被屏蔽。 1：调制解调器状态中断已发出信号。
2	<i>RLSIS</i>	RW1C	0	UART RX线状态中断状态和清除 0：UART RX线状态没有中断或者中断被屏蔽。 1：Rx线状态已发出信号。
1	<i>TXEMPIS</i>	RW1C	0	UART TX空中断状态和清除 0：UART TX不为空或中断被屏蔽。 1：TX空中断已发出信号。
0	<i>RXRDIS</i>	RW1C	0	UART接收数据可用的中断状态和清除 0：UART接收数据，不提供或中断被屏蔽。 1：UART接收可用的中断被通知的数据。

4.7.3.17 UART_485CTRL -UART RS485 控制寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 R/W	4 R/W	3 RO	2 R/W	1 R/W	0 R/W
										OINV	DCTRL		AADEN	RXDIS	NMMEN

偏移地址: 0x0034

域	名称	读/写	默认值	描述
31:6	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
5	<i>OINV</i>	R/W	0	该位保持在引脚RTS（或DTR）极性方向控制信号。 0：当发射机有数据要发送，方向控制销将被驱动为逻辑“0”。最后一个数据位被发送后，该位将被驱动为逻辑“1”。 1：当发射机有数据要发送，方向控制销将被驱动为逻辑“1”。最后一个数据位被发送后，该位将被驱动为逻辑“0”。
4	<i>DCTRL</i>	R/W	0	方向控制使能 0：禁用自动方向控制 1：启用自动方向控制
3	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
2	<i>AADEN</i>	R/W	0	自动地址检测（AAD）启用 留意AAD在RS-485 NMM模式下不可使用 0：禁用AAD 1：启用AAD
1	<i>RXDIS</i>	R/W	0	接收器使能 0：启用接收器 1：禁用接收器
0	<i>NMMEN</i>	R/W	0	RS-485 多点通信模式(NMM) 使能 0：禁用NMM 1：启用NMM

4.7.3.18 UART_485ADX - UART RS485 地址匹配寄存器

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

偏移地址: 0x0038

域	名称	读/写	默认值	描述
31:8	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出
7:0	<i>ADRMATCH</i>	R/W	0x0	包含地址匹配值

4.7.3.19 UART_RS485DLY -UART RS485 方向延时寄存器

用户可以编程与最后一个停止位离开 TxFIFO 和 RTSn(或 DTRn)的去断言之间的延迟的 8 位 UART_485DLY 寄存器。这个延迟时间是在波特率时钟的周期。从 0 到 255 比特时间的任何延迟时间可以被编程。

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

偏移地址: 0x003C

域	名称	读/写	默认值	描述
31:8	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
7:0	<i>DLY</i>	R/W	0x0	包含方向控制（RTSn或DTRn）延迟值。该寄存器协同工作以一个8位的计数器。

4.7.3.20 UART_USR -UART 状态寄存器

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

偏移地址: 0x007C

域	名称	读/写	默认值	描述
31:5	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
4	<i>RFF</i>	RO	0	接收FIFO满 当Rx FIFO不再满该位被清零。 0：接收FIFO未满 1：接收FIFO满
3	<i>RFNE</i>	RO	0	接收FIFO非空 当RX FIFO为空时，此位被清零。 0：接收FIFO空 1：接收FIFO不为空
2	<i>TFE</i>	RO	1	发送FIFO空 当发送FIFO不再是空该位被清零 0：发送FIFO不为空 1：发送FIFO空
1	<i>TFNF</i>	RO	1	发送FIFO非空 当TX FIFO满该位被清零。 0：发送FIFO满 1：发送FIFO未满
0	<i>BUSY</i>	RO	0	UART忙 0：UART空闲或处于非活动状态。 1：UART忙（串行传输过程中）

4.7.3.21 UART_TFL - UART 发送 FIFO 水平寄存器

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

偏移地址: 0x0080

域	名称	读/写	默认值	描述
31:4	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
3:0	<i>TFL</i>	RO	0x0	发送FIFO级别 这是用来表示数据条目的发送FIFO的数量。

4.7.3.22 UART_RFL - UART 接收 FIFO 水平寄存器

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

偏移地址: 0x0084

域	名称	读/写	默认值	描述
31:4	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
3:0	<i>RFL</i>	RO	0x0	接收FIFO水平 这是用来指示数据中的条目接收FIFO的数量。

4.8 脉冲宽度调制 (PWM)

脉宽调制是一种根据调制信号的信息来控制脉冲宽度的调制方式。通过改变脉冲宽度或者占空比,可以控制施加在电子器件上的有效功率。PWM 通常用于开关电源和电机控制应用。

PT32U301 PWM 模块由 3 个 PWM 发生器模块和控制模块组成。控制块决定了 PWM 信号的极性,以及将哪个信号传递到管脚。

每个 PWM 发生器产生两个 PWM 信号 (PWMx_A and PWMx_B, x=0, 1, 2) 可与彼此 (使用通用定时器频率) 独立工作, 或者它也可以是在上升沿和下降沿之间的可编程的死区延时互补信号。被驱动器件管脚之前的 PWM 输出信号首先由输出控制模块管理。输出控制模块选择哪个是 PWM 输出信号被传递到器件引脚(PWMx_A 和 PWMx_B x=0, 1, 2)和信号是否需要反相。

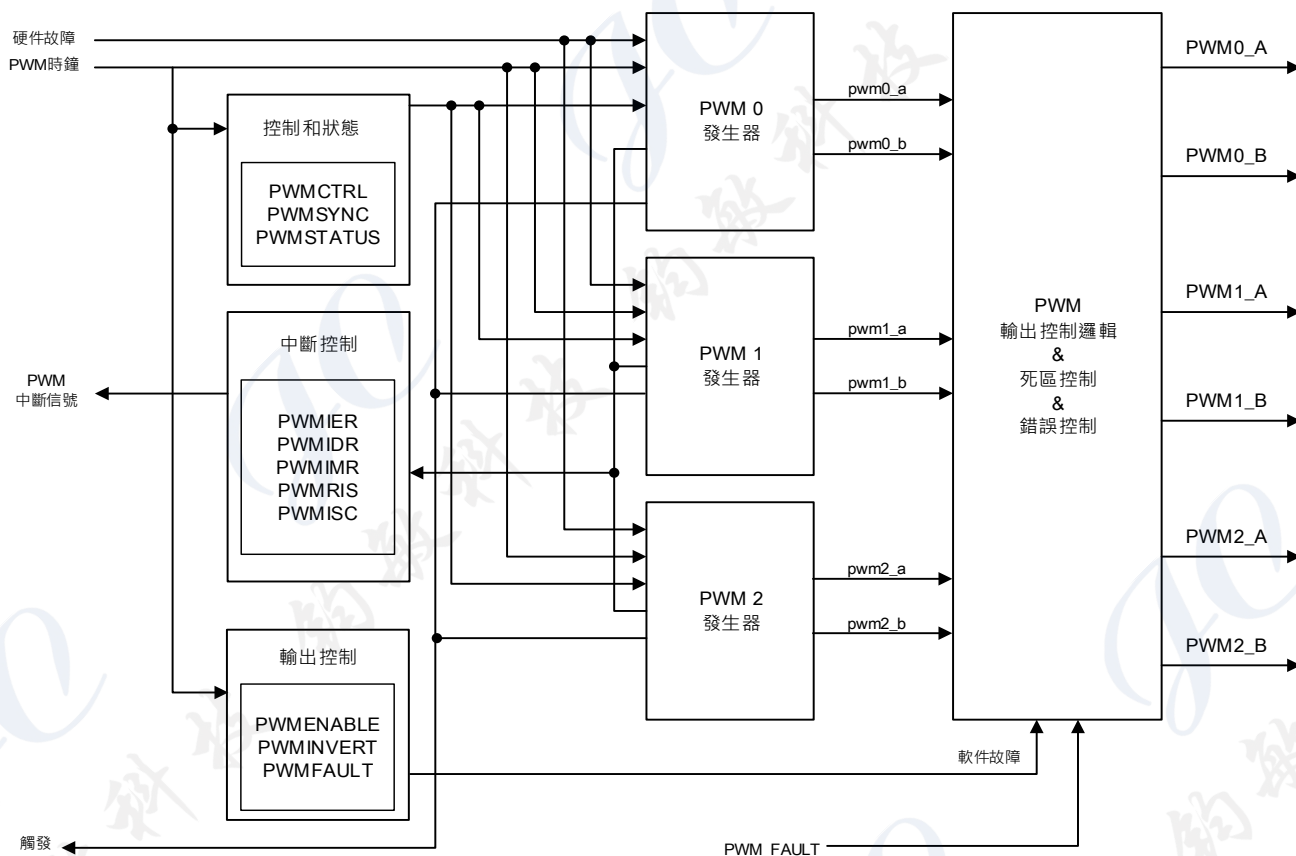
PWM 模块的操作是非常灵活的。它能产生电荷泵所需要的一个简单 PWM 信号, 或者它也可以产生与死区延时互补信号, 如在 H 桥驱动器电路常用。三个 PWM 发生器可以产生适用于电机控制的三相互补 PWM 信号。

每个 PWM 发生器有一个 16 位元 PWM 定时器, 三个用于 PWM 占空比控制的比较器, 1 个 4 位预分频器, 一个死区发生器, 一个信号发生器和一个 ADC 触发。每个 PWM 发生模块提供如下功能:

- PWM 信号发生器
 - 根据计数器和 PWM 比较器的输出信号来产生 PWM 输出信号
 - 可产生两个独立的 PWM 信号
- 一个 16 位定时器
 - 运行在: 减计数, 加计数和向上/向下计数模式。
 - 由 16 位负载值控制的输出频率控制
 - 负载值的更新可以同步
 - 在零和负载值生成输出信号
- 三个 PWM 比较器
 - 支持同步更新比较值
 - 在匹配的时候产生输出信号
- 死区发生器
 - 产生具有可编程死区延时的 PWM 信号, 用于驱动半 H 桥
 - 可以被旁路, 不修改输入的 PWM 信号
- PWM 输出可以用于触发和启动 ADC 转换。

4.8.1 块图

图 4.8-1 : PWM 框图



4.8.2 PWM 功能说明

4.8.2.1 PWM 定时器

在 PT32U301，通过计数寄存器（**PWMn_COUNT**），负载值寄存器（**PWMn_LOAD**）和比较寄存器的配置，PWM 定时器有三种计数模式：

- 递减计数模式
定时器从装载值递减计数到零，又回到了负荷值，并继续向下计数。递减计数模式一般用于右对齐 PWM 信号的产生。
- 递增计数模式
定时器从零开始计数到负载值。再回落到零，并继续计数。递增计数模式一般用于左对齐 PWM 信号的产生。
- 递增/递减计数模式
定时器从零计数到的负荷值，并计算从负载值下降到零，并一直重覆。减计数模式和递增计数模式一般用于中心对齐的 PWM 信号的产生。

右对齐 PWM 一般用于在特别情况下，需要左对齐 PWM 的相反对齐。左对齐 PWM 通用一般 PWM 用途。中心对齐 PWM 脉宽一般用于 AC 马达控制去维持相位校准。

PWM 定时器产生 3 个控制信号并被用在 PWM 产生过程中：

- 方向信号(标记为 **Direction**)用作表示定时器是处于减计数（“0”）或递增计数（“1”）方向。
- 单时钟周期宽高当计数器是零(标记为 **Zero**)。
- 单时钟周期宽度高当计数器等于负载值(标记为 **Load**)。

注：在递减模式中，零脉冲之后紧接着一个装载脉冲。在递增模式中，负载脉冲之后紧接着一个零脉冲。

4.8.2.2 PWM 比较器

每个 PWM 发生器都有三个比较器监控计数器值，当任何一个比较器与计数器的值相等时，这两个比较器输出一个宽度等于单个时钟周期的高电平脉冲。标记为“A”，“B”，“C”。

在递增/递减计数模式中，比较器在递增和递减计数时都要进行比较，因此必须通过计数器的方向信号来限定。这些限定脉冲在 PWM 信号的生成过程中使用。如果任一比较值大于计数器负载值大，那么该比较器从不输出高脉冲。下图显示了计数器处于不同计数模式时的行为以及这些脉冲之间的关系。

图 4.8-2：PWM 递减计数模式

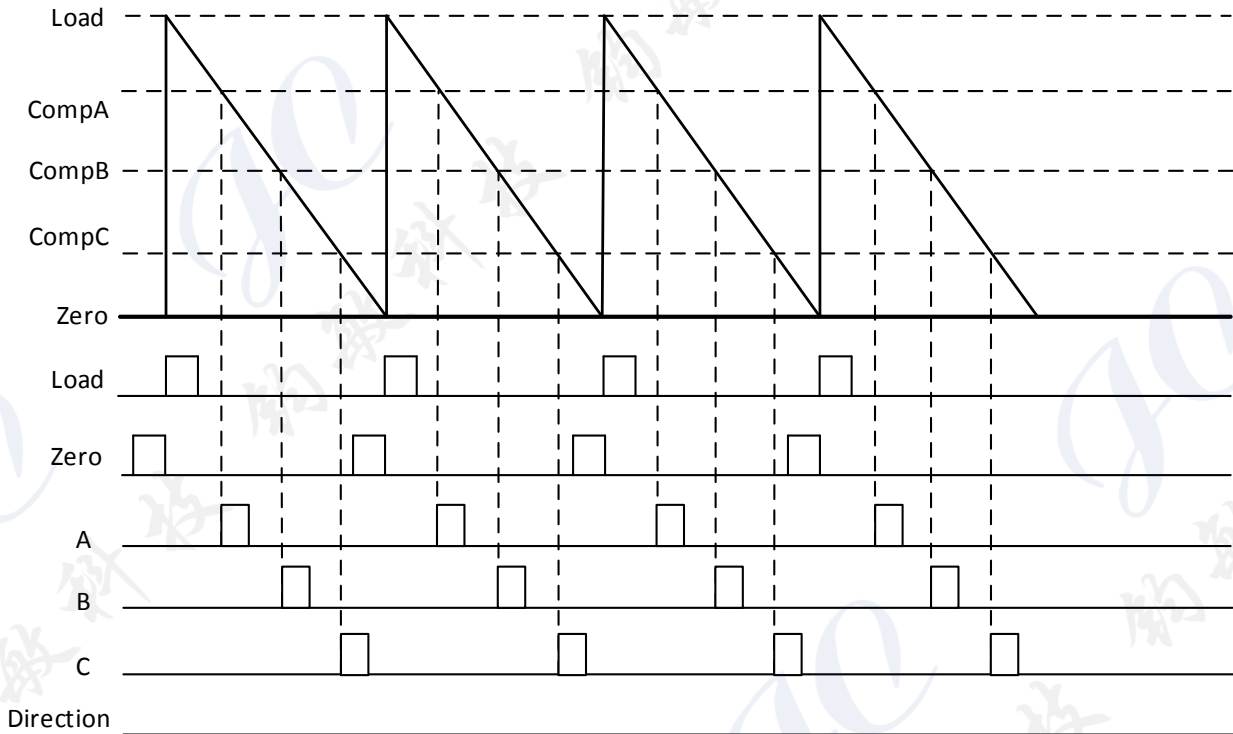


图 4.8-3 : PWM 递增计数模式

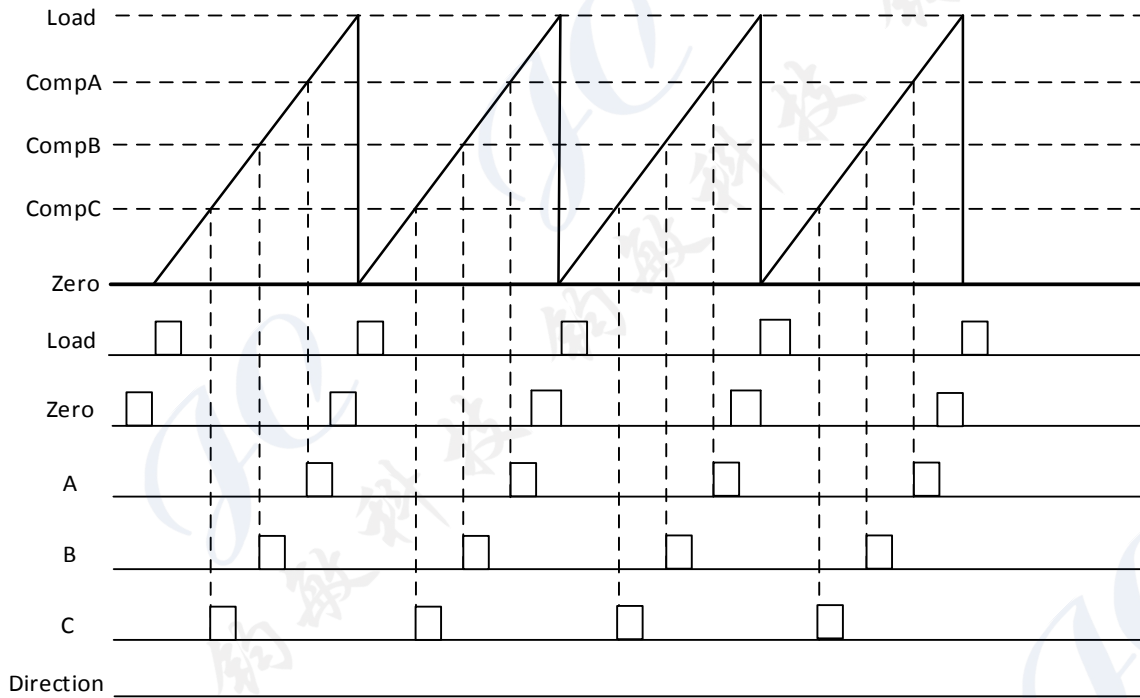
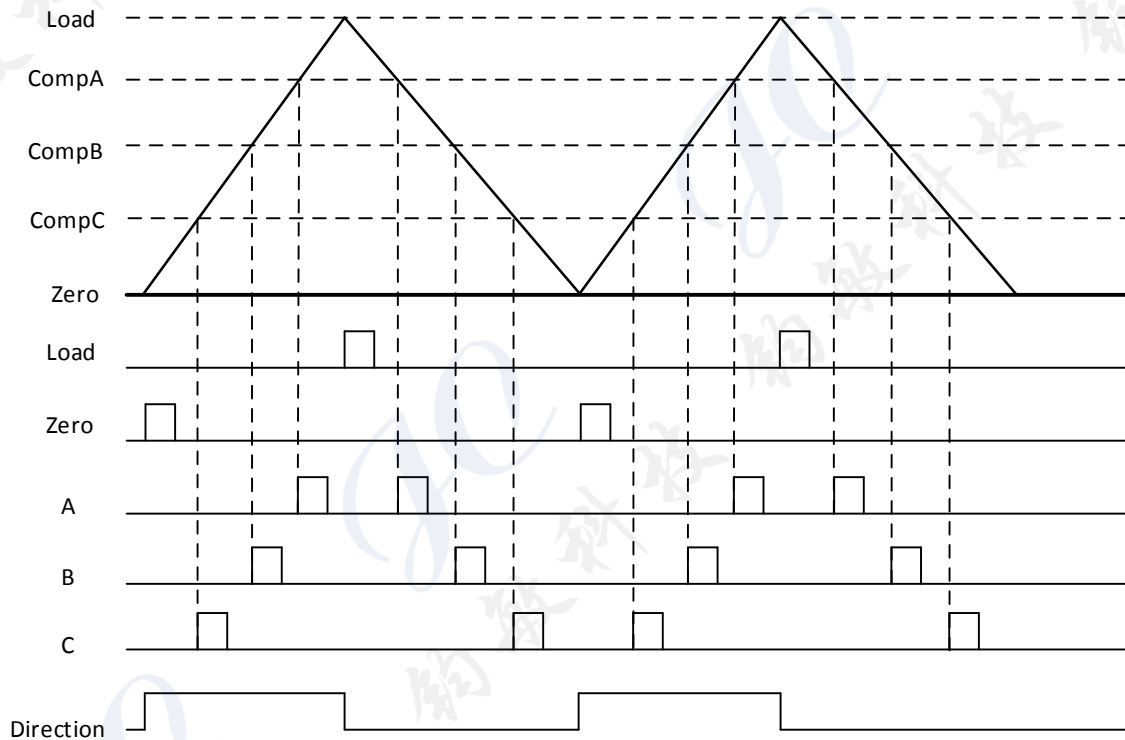


图 4.8-4: PWM 向上/向下计数模式



4.8.2.3 PWM 信号发生器

两个内部的 PWM 信号(pwm_a 及 pwm_b)都由下列匹配事件影响而独立产生：

- 匹配零
- 匹配装载
- 匹配比较器 A(匹配 A 递增/递减; 由方向信号来限定)
- 匹配比较器 B(匹配 B 递增/递减; 由方向信号来限定)
- 匹配比较器 C(匹配 C 递增/递减; 由方向信号来限定)

在递减计数模式中，能够影响这些信号的事件有：零、装载、匹配 A 递减、匹配 B 递减以及匹配 C 递减。

在递增计数模式中，能够影响这些信号的事件有：零、装载、匹配 A 递增、匹配 B 递增以及匹配 C 递增。

在递增/递减模式中，能够影响这些信号的事件有六个：零、装值、匹配 A 递增、匹配 B 递增、匹配 C 递增、匹配 A 递减、匹配 B 递减以及匹配 C 递减。

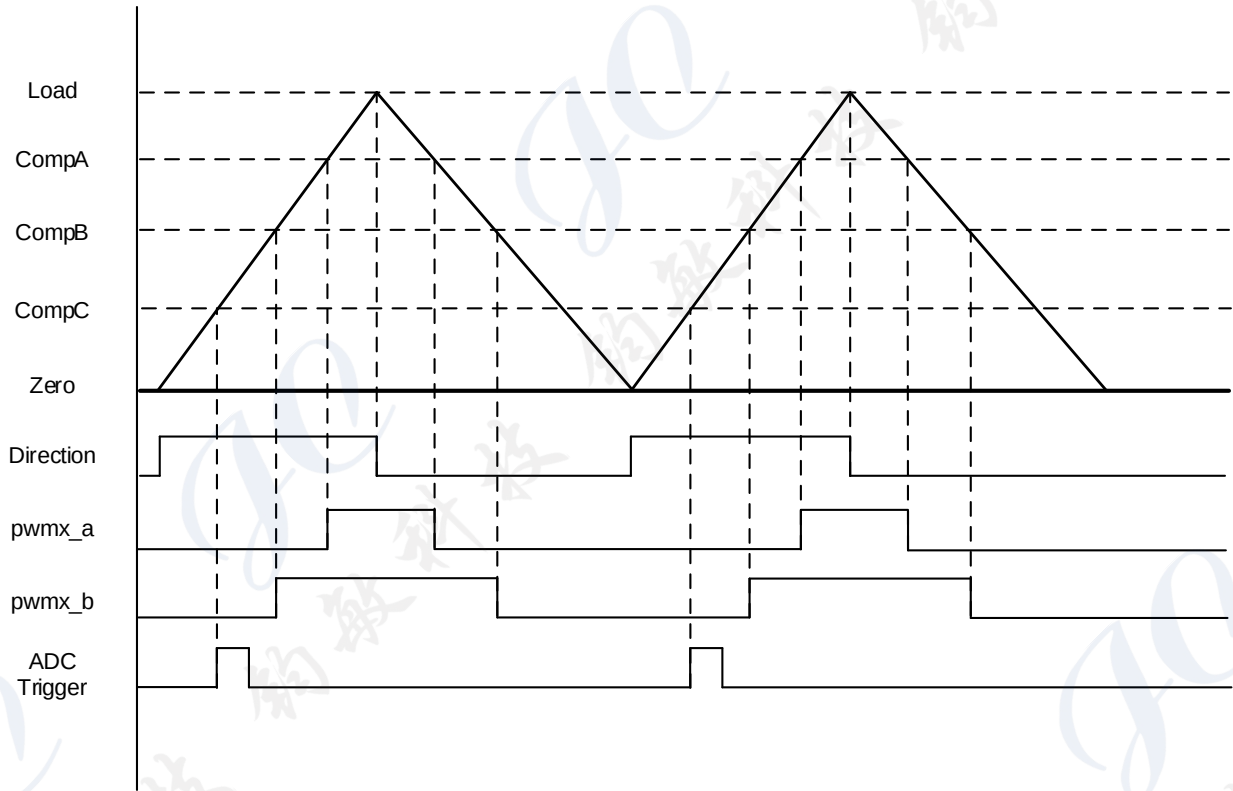
当匹配 A 或匹配 B 事件与零或装载事件重合，他们可以忽略不计。如果匹配 A 和匹配 B 与事件一致，所述第一信号 pwm_a 事件仅基于匹配所产生的，所述第二信号 pwm_b 事件仅基于匹配 B 所产生。

各个事件在 PWM 输出信号的影响上都是可编程的：

- 可以保留（忽略该事件）
- 可以翻转
- 可以驱动为低电平
- 可以驱动为高电平

这些动作可用来产生一对不同位置 and 不同占空比的 PWM 信号，这对 信号量可以重叠或不重叠。

图 4.8-5：加/减计数器模



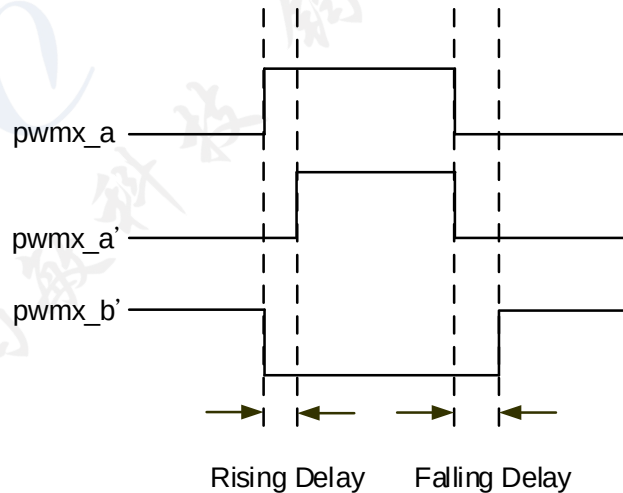
在这上面实例中，pwm_x_a 被设置为在出现匹配 A 递增事件时驱动为高电平，出现匹配 A 递减事件时驱动为低电平，而忽略其他活动。pwm_x_b 被设置为在出现匹配 B 递增事件时驱动为高电平，出现匹配 B 递减事件时驱动为低电平，而忽略其他活动。改变比较器 A 的值可改变 pwm_x_a 信号的占空比，改变比较器 B 的值可改变 pwm_x_b 信号的占空比。比较器 C 发生器被设置为在出现匹配 A 递增或递减事件时，输出一个脉冲，但 ADC 触发只有在递增时基时才会产生。

4.8.2.4 PWM 死区发生器

PWM 发生器产生的两个 PWM 信号被传递到死区发生器。如果死区发生器禁能，则 PWM 信号只简单地通过该模块，而不会发生改变。如果死区发生器使能，则丢弃第二个 PWM 信号，并在第一个 PWM 信号基础上产生两个 PWM 信号。第一个输出 PWM 信号为带上升沿延迟的输入信号，延迟时间可编程。第二个输出 PWM 信号为输入信号的反相信号，在输入信号的下降沿和这个新信号的上升沿之间增加了可编程的延迟时间。

PWMA 和 PWMB 形成一个非重叠的高电平信号，其中一个总是高的，除了过渡时间的可编程量，其中两者都是低这是死区时间，因为这两个信号是无效的。当这些信号被用于驱动 H 桥配置中，这些表示都高侧和低侧的 NMOS 晶体管截止。死区时间插入开关 NMOS，以避免贯通电流之间。下图显示了输入 PWM 信号发生器的死区的效果。

图 4.8-6：输入 PWM 信号的死区的影响



4.8.2.5 中断/ ADC-触发选择

PWM 发生器还使用相同的五（或八个）计数器事件产生中断或 ADC 触发信号。任何这些事件或一组这些事件可以被选择作为用于中断或 ADC 触发信号的源。中断或 ADC 触发器可配置为在 PWM 周期中的任何点发生。注意，当被启用死区，中断和 ADC 触发是基于 PWM 信号的原始（原）的事件，在 PWM 信号延迟边缘由死区发生器引起不考虑。

在 PWM 中断或 ADC 触发由一组 5 个寄存器控制。

- 中断控制（IER，IDR，IMR）

PWM 中断使能寄存器(PWM_IER)允许通过写'1'的中断请求线。同样，PWM 中断禁止 (PWM_IDR)通过写入“1”禁止中断请求线。IER 和 IDR 是控制中断屏蔽只写寄存器。这两个寄存器的总体结果可以通过 PWM 中断中显示屏蔽寄存器(PWM_IMR)。IMR 是只读使用'1'或'0'来指示中断请求线被使能/或禁用寄存器。

- 中断状态读取（RIS）

PWM 原始中断状态(PWM_RIS)是一个只读寄存器读取所有中断模块的状态。

- 中断清除（ISC）

PWM 中断状态和中断清除寄存器(PWM_ISC)用于指示模块的不可屏蔽的中断状态，因为只有现在屏蔽的中断被认定，以处理器。该寄存器写“1”到该位可以清除相应的中断状态或写 1 IDR 禁止中断。

4.8.2.6 同步方法

具有全局复位功能，该功能可同时复位 PWM 发生器中的任何或全部计数器。如果多个 PWM 发生器使用相同的计数器装载值来配置，那么可以保证 PWM 发生器也具有相同的计数值（这不表示 PWM 发生器必须在其同步之前被配置）。这样，通过那些信号边沿之间的已知关系可产生 2 个以上的 PWM 信号，因为计数器总是具有相同的值。

在 PWM 发生器，比较值可通过两种方式进行更新。一个是当计数器到达零时，更新比较值。此行为被很好地定义，避免过短或过长的 PWM 输出脉冲。

另一种方法是使用一个全局同步更新信号。当计数器达到零，全局同步更新信号启用，则新值被使用。这种方法允许多个 PWM 发生器可以同时具有良好定义的行为进行更新。这意味着，一切运行从旧的值，直到在这一点，他们都从新值运行。

4.8.2.7 故障状态

影响 PWM 模块的外部条件有两个；一个是故障管脚的信号输入(PWM_FAULT)，另一个是由调试器引发的控制器中止。我们可以采用两种机制来处理这些情况，一是强制将输出信号变为无效 (inactive) 状态，以及/或者让 PWM 定时器停止运行。

每个输出信号都带有一个故障位。若故障位置位，则故障输入信号将会使相应的输出信号变为无效状态。如果无效状态指的是信号能够长期停留的安全状态，那么这样可避免输出信号在故障状态下以危险的方式驱动外部电路。此外，故障条件还可以产生控制器中断。

每个 PWM 发生器还可以被配置以停止计数，每当一个失速状态为止。该计数器可以进行一旦它达到零计数停止。失速状态不会产生中断。

4.8.2.8 输出控制模块

每个 PWM 发生器产生两个原始 PWM 信号，输出控制模块在 PWM 信号进入管脚之前要对其最后的状态进行控制。通过一个 PWM 输出使能寄存器(PWM_ENABLE)就能够对实际传递到管脚的 PWM 信号进行修改。这非常适合用于控制无刷直流电机。可以利用 PWM 输出反向寄存器(PWM_INVERT)对任一 PWM 信号执行反相操作，使得默认高电平有效的信号变为低电平有效，反之亦然。

4.8.3 PWM 寄存器映射

基地址: 0x4000_C000

偏移地址	寄存器名称	读/写	默认值	描述	参考页
0x0000	PWM_CTRL	R/W	0x0000_0000	PWM 主控	126
0x0004	PWM_SYNC	R/W	0x0000_0000	PWM 时序同步	127
0x0008	PWM_ENABLE	R/W	0x0000_0000	PWM 输出使能	128
0x000C	PWM_INVERT	R/W	0x0000_0000	PWM 输出反向	129
0x0010	PWM_FAULT	R/W	0x0000_0000	PWM 故障控制	130
0x0014	PWM_IER	WO	0x0000_0000	PWM 中断使能	131
0x0018	PWM_IDR	WO	0x0000_0000	PWM 中断禁止	132
0x001C	PWM_IMR	RO	0x0000_0000	PWM 中断屏蔽	133
0x0020	PWM_RIS	RO	0x0000_0000	PWM 原始中断状态	134
0x0024	PWM_ISC	R/W1C	0x0000_0000	PWM 中断状态和清零	135
0x0028	PWM_STATUS	RO	0x0000_0000	PWM 故障输入状态	136
0x0040	PWM0_CTL	R/W	0x0000_0000	PWM0 控制	137
0x0044	PWM0_IER	WO	0x0000_0000	PWM0 中断使能	138
0x0048	PWM0_IDR	WO	0x0000_0000	PWM0 中断禁止	140
0x004C	PWM0_IMR	RO	0x0000_0000	PWM0 中断屏蔽	142
0x0050	PWM0_RIS	RO	0x0000_0000	PWM0 原始中断状态	144
0x0054	PWM0_ISC	R/W1C	0x0000_0000	PWM0 中断状态和清零	145
0x0058	PWM0_LOAD	R/W	0x0000_0000	PWM0 装载值	146
0x005C	PWM0_COUNT	RO	0x0000_0000	PWM0 计数器	146
0x0060	PWM0_COMPA	R/W	0x0000_0000	PWM0 比较器A	147
0x0064	PWM0_COMPB	R/W	0x0000_0000	PWM0 比较器B	147
0x0068	PWM0_COMPC	R/W	0x0000_0000	PWM0 比较器C	148
0x006C	PWM0_GENA	R/W	0x0000_0000	PWM0 发生控制器A	149
0x0070	PWM0_GENB	R/W	0x0000_0000	PWM0 发生控制器B	150
0x0074	PWM0_DBCTL	R/W	0x0000_0000	PWM0 死区控制	151
0x0078	PWM0_DBRIS	R/W	0x0000_0000	PWM0 死区上升延迟	151
0x007C	PWM0_DBFALL	R/W	0x0000_0000	PWM0 死区下降延迟	152
0x0080	PWM1_CTL	R/W	0x0000_0000	PWM1 控制	137
0x0084	PWM1_IER	WO	0x0000_0000	PWM1 中断使能	138
0x0088	PWM1_IDR	WO	0x0000_0000	PWM1 中断禁止	140
0x008C	PWM1_IMR	RO	0x0000_0000	PWM1 中断屏蔽	142
0x0090	PWM1_RIS	RO	0x0000_0000	PWM1 原始中断状态	144
0x0094	PWM1_ISC	R/W1C	0x0000_0000	PWM1 中断状态和清零	145
0x0098	PWM1_LOAD	R/W	0x0000_0000	PWM1 装载值	146
0x009C	PWM1_COUNT	RO	0x0000_0000	PWM1 计数器	146
0x00A0	PWM1_COMPA	R/W	0x0000_0000	PWM1 比较器A	147
0x00A4	PWM1_COMPB	R/W	0x0000_0000	PWM1 比较器B	147
0x00A8	PWM1_COMPC	R/W	0x0000_0000	PWM1 比较器C	148



偏移地址	寄存器名称	读/写	默认值	描述	参考页
0x00AC	PWM1_GENA	R/W	0x0000_0000	PWM1 发生控制器A	149
0x00B0	PWM1_GENB	R/W	0x0000_0000	PWM1 发生控制器B	150
0x00B4	PWM1_DBCTL	R/W	0x0000_0000	PWM1 死区控制	151
0x00B8	PWM1_DBRSE	R/W	0x0000_0000	PWM1 死区上升延迟	151
0x00BC	PWM1_DBFALL	R/W	0x0000_0000	PWM1 死区下降延迟	152
0x00C0	PWM2_CTL	R/W	0x0000_0000	PWM2 控制	137
0x00C4	PWM2_IER	WO	0x0000_0000	PWM2 中断使能	138
0x00C8	PWM2_IDR	WO	0x0000_0000	PWM2 中断禁止	140
0x00CC	PWM2_IMR	RO	0x0000_0000	PWM2 中断屏蔽	142
0x00D0	PWM2_RIS	RO	0x0000_0000	PWM2 原始中断状态	144
0x00D4	PWM2_ISC	R/W1C	0x0000_0000	PWM2 中断状态和清零	145
0x00D8	PWM2_LOAD	R/W	0x0000_0000	PWM2 装载值	146
0x00DC	PWM2_COUNT	RO	0x0000_0000	PWM2 计数器	146
0x00E0	PWM2_COMP_A	R/W	0x0000_0000	PWM2 比较器A	147
0x00E4	PWM2_COMP_B	R/W	0x0000_0000	PWM2 比较器B	147
0x00E8	PWM2_COMP_C	R/W	0x0000_0000	PWM2 比较器C	148
0x00EC	PWM2_GENA	R/W	0x0000_0000	PWM2 发生控制器A	149
0x00F0	PWM2_GENB	R/W	0x0000_0000	PWM2 发生控制器B	150
0x00F4	PWM2_DBCTL	R/W	0x0000_0000	PWM2 死区控制	151
0x00F8	PWM2_DBRSE	R/W	0x0000_0000	PWM2 死区上升延迟	151
0x00FC	PWM2_DBFALL	R/W	0x0000_0000	PWM2 死区下降延迟	152

4.8.3.1 PWM_CTRL - PWM 主控制

生成 PWM 模块的寄存器是主要的控制。设置单独的比特引起任何排队的更新中的特定 PWM 发生器 X A 负载或比较寄存器 (X = 0, 1, 2) 的下一一次相应的计数器变为零施加。当更新已经完成，该位自动清零。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 R/W	1 R/W	0 R/W
													UPDPWM2	UPDPWM1	UPDPWM0

偏移地址: 0x0000

域	名称	读/写	默认值	描述
31:3	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
2	UPDPWM2	R/W	0	更新PWM发生器2 0: 沒有影響 1: 在下一一次对应的计数器变为 0 时，PWM 发生器 2 中已排队的更新将应用。
1	UPDPWM1	R/W	0	更新PWM发生器1 0: 沒有影響 1: 在下一一次对应的计数器变为 0 时，PWM 发生器 1 中已排队的更新将应用。
0	UPDPWM0	R/W	0	更新PWM发生器0 0: 沒有影響 1: 在下一一次对应的计数器变为 0 时，PWM 发生器 0 中已排队的更新将应用。

4.8.3.2 PWM_SYNC - PWM 时基同步

该寄存器提供一种对 PWM 发生模块中的计数器进行同步的方法。向该寄存器的某个位写 1 会让指定的计数器复位为零。如果同时向多个位写 1 会让多个计数器同时复位。复位之后，寄存器中的位自动清零，读这些位时如果返回零，则表示同步已完成。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 R/W	1 R/W	0 R/W
													SYNC2	SYNC1	SYNC0

偏移地址: 0x0004

域	名称	读/写	默认值	描述
31:3	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
2	SYNC2	R/W	0	复位发生器2计数器 0：沒有影響 1：发生器2计数器清零
1	SYNC1	R/W	0	复位发生器1计数器 0：沒有影響 1：发生器1计数器清零
0	SYNC0	R/W	0	复位发生器0计数器 0：沒有影響 1：0发电机计数器回零

4.8.3.3 PWM_ENABLE - PWM 输出使能

该寄存器提供了一个主控制，产生 PWM 信号传递到器件引脚。该 PWM 操作仍在正常运行，即使输出被禁止。设置这些位为“1”，以让相应的 PWM 信号通过输出级，它也是由 PWM_INVERT 寄存器控制。把对应的 PWM 信号的位清除替换为零值，并通过该输出级。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
									SYNCOUT	PWM2BEN	PWM2AEN	PWM1BEN	PWM1AEN	PWM0BEN	PWM0AEN

偏移地址: 0x0004

域	名称	读/写	默认值	描述
31:7	reserved	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
6	SYNCOUT	R/W	0	1：PWM输出使能只會發生在零点後
5	PWM2BEN	R/W	0	PWM2_B 输出使能 0：PWM2_B信号的值是零 1：允许产生PWM2_B信号被传递到器件的引脚。
4	PWM2AEN	R/W	0	PWM2_A输出使能 0：PWM2_A信号的值是零 1：允许生成的PWM2_A信号被传递到器件的引脚。
3	PWM1BEN	R/W	0	PWM1_B输出使能 0：PWM1_B信号的值是零 1：允许产生的PWM1_B信号被传递到器件的引脚。
2	PWM1AEN	R/W	0	PWM1_A输出使能 0：PWM1_A信号的值是零 1：允许产生PWM1_A信号被传递到器件的引脚。
1	PWM0BEN	R/W	0	PWM0_B输出启用 0：PWM0_B信号的值是零 1：允许产生PWM0_B信号被传递到器件的引脚。
0	PWM0AEN	R/W	0	PWM0_A输出使能 0：PWM0_A信号的值是零 1：允许产生PWM0_A信号被传递到器件的引脚。

4.8.3.4 PWM_INVERT -PWM 输出反相

该寄存器对器件管脚上的 PWM 信号的极性进行控制。设置相应位为“1”反转该 PWM 信号。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
										PWM2BINV	PWM2AINV	PWM1BINV	PWM1AINV	PWM0BINV	PWM0AINV

偏移地址: 0x000C

域	名称	读/写	默认值	描述
31:6	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
5	<i>PWM2BINV</i>	R/W	0	反转信号PWM2_B 0：沒有反相 1：信号PWM2_B反相
4	<i>PWM2AINV</i>	R/W	0	反转信号PWM2_A 0：沒有反相 1：信号PWM2_A反相
3	<i>PWM1BINV</i>	R/W	0	反转信号PWM1_B 0：沒有反相 1：信号PWM1_B反相
2	<i>PWM1AINV</i>	R/W	0	反转信号PWM1_A 0：沒有反相 1：信号PWM1_A反相
1	<i>PWM0BINV</i>	R/W	0	反转信号PWM0_B 0：沒有反相 1：信号PWM0_B反相
0	<i>PWM0AINV</i>	R/W	0	反转信号PWM0_A 0：沒有反相 1：信号PWM0_A反相

4.8.3.5 PWM_FAULT -PWM 控制故障

该寄存器用来控制出现故障情况时的 PWM 输出行为。故障输入和调试事件均可看作故障条件。每个 PWM 信号均可以直接通过，而无需修改或驱动为低电平。对于配置为直通的输出来说，处理相应 PWM 发生器的调试事件还决定是否继续生成 PWM 信号。故障条件在输出反相器之前发生。这样，由于故障状态驱动为低电平，在故障条件下驱动为低电平的 PWM 信号将被反相。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
									FAULT2B	FAULT2A	FAULT1B	FAULT1A	FAULT0B	FAULT0A	FAULTEN

偏移地址: 0x0010

域	名称	读/写	默认值	描述
31:7	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
6	<i>FAULT2B</i>	R/W	0	PWM2_B故障 0：产生的pwm2_b'信号被传递到PWM2_B管脚 1：PWM2_B输出信号为低的故障状态
5	<i>FAULT2A</i>	R/W	0	PWM2_A故障 0：产生的pwm2_a'信号被传递到PWM2_A管脚 1：PWM2_A输出信号为低的故障状态
4	<i>FAULT1B</i>	R/W	0	PWM1_B故障 0：产生的pwm1_b'信号被传递到PWM1_B管脚 1：PWM1_B输出信号为低的故障状态
3	<i>FAULT1A</i>	R/W	0	PWM1_A故障 0：产生的pwm1_a'信号被传递到PWM1_A管脚 1：PWM1_A输出信号为低的故障状态
2	<i>FAULT0A</i>	R/W	0	PWM0_B故障 0：产生的pwm0_b'信号被传递到PWM0_B管脚 1：PWM0_B输出信号为低的故障状态
1	<i>FAULT0B</i>	R/W	0	PWM0_A故障 0：产生的pwm0_a'信号被传递到PWM0_A管脚 1：PWM0_A输出信号为低的故障状态
0	<i>FAULTEN</i>	R/W	0	PWM故障软件启用 0：禁用故障状态控制 1：启用故障状态控制

4.8.3.6 PWM_IER - PWM 中断使能

该寄存器允许全局中断产生。中断源是从故障引脚和 PWM 发生器的各个中断。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 WO	7 RO	6 RO	5 RO	4 RO	3 RO	2 WO	1 WO	0 WO
							FAULTIE						PWM2IE	PWM1IE	PWM0IE

偏移地址: 0x0014

域	名称	读/写	默认值	描述
31:9	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
8	<i>FAULTIE</i>	WO	0	故障中断启用 1：当故障输入生效时发生中断。
7:3	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
2	<i>PWM2IE</i>	WO	0	PWM2 中断使能 1：当PWM发生器2 产生中断信号时发生中断。
1	<i>PWM1IE</i>	WO	0	PWM1 中断使能 1：当PWM发生器1 产生中断信号时发生中断。
0	<i>PWM0IE</i>	WO	0	PWM0 中断使能 1：当PWM发生器0 产生中断信号时发生中断。

4.8.3.7 PWM_IDR -PWM 中断禁止

该寄存器控制 PWM 模块的全局中断生成功能。能引起中断发生的事件是故障输入和个人从 PWM 发生器中断。使用 PWMn_IDR 控制中断禁用功能独立 PWM。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 WO	7 RO	6 RO	5 RO	4 RO	3 RO	2 WO	1 WO	0 WO
							FAULTID						PWM2ID	PWM1ID	PWM0ID

偏移地址: 0x0018

域	名称	读/写	默认值	描述
31:9	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
8	<i>FAULTID</i>	WO	0	故障中断禁用 1：当故障输入的确认不会出现中断。
7:3	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
2	<i>PWM2ID</i>	WO	0	PWM2中断禁止 1：当PWM发生器2块产生中断信号不发生会中断。
1	<i>PWM1ID</i>	WO	0	PWM1中断禁止 1：当PWM发生器1块产生中断信号不发生会中断。
0	<i>PWM0ID</i>	WO	0	PWM0中断禁止 1：当PWM发生器0模块产生中断信号不发生会中断。

4.8.3.8 PWM_IMR - PWM 中断屏蔽

该寄存器显示了全局 PWM 模块中断屏蔽状态。使用 **PWMn_IMP** 读取单个中断屏蔽状态。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
							FAULTIM						PWM2IM	PWM1IM	PWM0IM

偏移地址: 0x001C

域	名称	读/写	默认值	描述
31:9	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
8	<i>FAULTIM</i>	RO	0	故障中断屏蔽状态 0：故障输入中断将被屏蔽。 1：启用故障输入中断。
7:3	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
2	<i>PWM2IM</i>	RO	0	PWM2中断屏蔽状态 0：PWM2输入中断将被屏蔽。 1：PWM2输入中断使能。
1	<i>PWM1IM</i>	RO	0	PWM1中断屏蔽状态 0：PWM1输入中断将被屏蔽。 1：PWM1输入中断使能
0	<i>PWM0IM</i>	RO	0	PWM0中断屏蔽状态 0：PWM0输入中断将被屏蔽。 1：PWM0输入中断使能

4.8.3.9 PWM_RIS -PWM 原始中断状态

该寄存器提供的当前集，而不管它们是否产生中断由控制器来断言被断言的中断源。故障中断在检测锁存，因此必须通过 PWM_ISC 寄存器被清除。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
							FAULTRI						PWM2RI	PWM1RI	PWM0RI

偏移地址: 0x0020

域	名称	读/写	默认值	描述
31:9	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
8	<i>FAULTRI</i>	RO	0	故障原始中断状态 0：无中断 1：故障输入断言。
7:3	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
2	<i>PWM2RI</i>	RO	0	PWM2原始中断状态 0：无中断 1：PWM2是断言其中断
1	<i>PWM1RI</i>	RO	0	PWM1原始中断状态 0：无中断 1：PWM1是断言其中断
0	<i>PWM0RI</i>	RO	0	PWM0原始中断状态 0：无中断 1：PWM0是维护其中断

4.8.3.10 PWM_ISC - PWM 中断状态和清除

该寄存器汇总了单个 PWM 发生器模块的中断状态。位设置为“1”表示已经激活的中断。对应的 PWM 发生器中断状态寄存器可以咨询了解中断的具体来源。对故障中断，写“1”到该位位置清除锁存中断状态。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 R/W1C	7 RO	6 RO	5 RO	4 RO	3 RO	2 R/W1C	1 R/W1C	0 R/W1C
							FAULTIS						PWM2IS	PWM1IS	PWM0IS

偏移地址: 0x0024

域	名称	读/写	默认值	描述
31:9	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
8	<i>FAULTIS</i>	RO	0	故障中断状态和清除 0：故障输入没有断言其中断或者中断被屏蔽。 1：中断已发出信号故障输入。
7:3	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
2	<i>PWM2IS</i>	RO	0	PWM2中断状态和清除 0：PWM发生器2还没有断言其中断或中断被屏蔽。 1：PWM发生器2中断已发出信号。
1	<i>PWM1IS</i>	RO	0	PWM1中断状态和清除 0：PWM发生器1未断言其中断或中断被屏蔽。 1：PWM发生器1中断已发出信号。
0	<i>PWM0IS</i>	RO	0	PWM0中断状态和清除 0：PWM发生器0并没有断言其中断或者中断被屏蔽。 1：PWM发生器0中断已发出信号。

4.8.3.11 PWM_STATUS - PWM 故障输入状态

该寄存器用于显示故障输入信号的状态。

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

偏移地址: 0x0028

域	名称	读/写	默认值	描述
31:1	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
0	<i>FAULT</i>	R/W	0	故障中断状态 1：在读操作，指示故障输入有效。 0：在写操作，触发故障

4.8.3.12 PWMn_CTL - PWM0/1/2 控制

这些寄存器配置 PWM 信号生成。寄存器更新模式，调试模式，计数模式和 PWM 模块实现通过这些寄存器都控制。PWM 发生器产生它可以是两个独立的 PWM 信号的 PWM 信号（來自同一個計數器），或一對 PWM 信号与死区延时插入。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 R/W	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
								DEBUG	CMPCUPD	CMPBUPD	CMPAUPD	LOADUPD	MODE	ENABLE	

偏移地址：

PWM0_CTL : 0x0040

PWM1_CTL : 0x0080

PWM2_CTL : 0x00C0

域	名称	读/写	默认值	描述
31:8	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
7	<i>DEBUG</i>	R/W	0	调试模式 在调试模式下，计数器的行为。 0：计数器一直运行即使在调试模式 1：计数器达到0时停止运行时，并继续在離開调试模式时再次运行
6	<i>CMPCUPD</i>	R/W	0	比较寄存器C更新模式。 更新模式为比较的寄存器。 0：在下次计数器为 0 时，PWMn_COMPC 寄存器的更新会反映到发生器。 1：对寄存器的更新将被延迟，直到下一次通过PWM_CTRL寄存器发出同步更新后计数器为 0 时为止。
5	<i>CMPBUPD</i>	R/W	0	比较寄存器B更新模式 相同的描述在CMPCUPD但仅用于比较器B寄存器。
4	<i>CMPAUPD</i>	R/W	0	比较寄存器C更新模式。 同样的描述，在CMPCUPD但只适用于比较的寄存器。
3	<i>LOADUPD</i>	R/W	0	加载寄存器更新模式。 更新模式为负载寄存器。 0：在下次计数器为 0 时，PWMn_LOAD 寄存器的更新会反映到发生器。 1：对寄存器的更新将被延迟，直到下一次通过 PWM_CTRL 寄存器发出同步更新后计数器为 0 时为止。
2:1	<i>MODE</i>	R/W	0	计数器模式 00：递减计数模式 01：递减增计数模式 10：递增/递减计数模式 11：保留
0	<i>ENABLE</i>	R/W	0	PWM模块启用 主启用对PWM生成块。 0：禁止模块和计时。 1：模块启用，并产生的PWM信号。

4.8.3.13 PWMn_IER -PWM0/ 1/2 中断使能

这些寄存器控制 PWM 发生器的中断和 ADC 触发功能。下面列出了可产生中断或 ADC 触发事件：这些事件的任意组合可以产生中断或 ADC 的触发器。

- 计数器等于负载寄存器。
- 计数器等于零。
- 计数器等于比较器寄存器中，而计数。
- 计数器等于比较器寄存器中，而倒计时。
- 计数器等于比较寄存器 B 同时计数。
- 计数器等于比较 B 寄存器递减计数。
- 计数器等于比较 C 寄存器而计数。
- 计数器等于比较 C 寄存器递减计数。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 WO	14 WO	13 WO	12 WO	11 WO	10 WO	9 WO	8 WO	7 WO	6 WO	5 WO	4 WO	3 WO	2 WO	1 WO	0 WO
TECMPCD	TECMPCU	TECMPBD	TECMPBU	TECMPAD	TECMPAU	TECNTL	TECNTZ	IECMPCD	IECMPCU	IECMPBD	IECMPBU	IECMPAD	IECMPAU	IECNTL	IECNTZ

偏移地址：

PWM0 IER : 0x0044

PWM1 IER : 0x0084

PWM2_IER : 0x00C4

域	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
15	<i>TRECMPCD</i>	WO	0	触发使能当计数器=比较C向下 0：无触发ADC 1：当计数器倒计时模式在PWMn_COMPC值的匹配将产生ADC触发。
14	<i>TRECMPCU</i>	WO	0	触发使能当计数器=比较C最大 0：无触发ADC 1：当计数器计数模式PWMn_COMPC值的匹配将产生ADC触发。
13	<i>TREMPBD</i>	WO	0	触发使能当计数器=比较器B向下 0：无触发ADC 1：当计数器倒计时模式在PWMn_COMPB值的匹配将产生ADC触发。
12	<i>TREMPBU</i>	WO	0	触发使能当计数器=比较器B向上 0：无触发ADC 1：当计数器计数模式PWMn_COMPB值的匹配将产生ADC触发。
11	<i>TREMPAD</i>	WO	0	触发使能当计数器=比较A向下 0：无触发ADC 1：当计数器倒计时模式在PWMn_COMPA值的匹配将产生ADC触发。
10	<i>TREMPAU</i>	WO	0	触发使能当计数器=比较一上来 0：无触发ADC 1：当计数器计数模式PWMn_COMPA值的匹配将产生ADC触发。

域	名称	读/写	默认值	描述
9	TRECNTLOAD	WO	0	触发使能当计数器=装载 0：无ADC触发输出 1：ADC触发脉冲时，输出计数器在PWMn_LOAD寄存器匹配。
8	TRECNTZERO	WO	0	触发使能当计数器=零 0：无ADC触发输出 1：ADC触发脉冲时，输出计数为0。
7	IECMPCD	WO	0	中断使能计数器=比较C向下 0：无中断 1：当计数器在PWMn_CMPC寄存器值的值相同时向下计数发生原始中断。
6	IECMPCU	WO	0	中断使能计数器=比较C最大 0：无中断 1：当计数器在PWMn_CMPC寄存器值的值相同时，计数时发生的原始中断。
5	IECMPBD	WO	0	中断使能计数器=比较器B向下 0：无中断 1：当计数器在PWM CMPB寄存器值的值相同时向下计数发生原始中断。
4	IECMPBU	WO	0	中断使能计数器=比较器B向上 0：无中断 1：当计数器在PWMn_CMPB寄存器值的值相同时，计数时发生的原始中断。
3	IECMPAD	WO	0	中断使能计数器=比较A向下 0：无中断 1：当计数器在PWMn_CMPA寄存器值的值相同时向下计数发生原始中断。
2	IECMPAU	WO	0	中断使能计数器=比较A向上 0：无中断 1：当计数器在PWMn_CMPA寄存器值的值相同时，计数时发生的原始中断。
1	IECNTLOAD	WO	0	中断使能计数器=装载 0：无中断 1：当计数器在PWMn_LOAD寄存器值的数值相匹配时，发生的原始中断。
0	IECNTZERO	WO	0	中断使能计数器=0 0：无中断 1：当计数器为零时，发生的原始中断。

4.8.3.14 PWMn_IDR - PWM0/ 1/2 的中断禁用

这些寄存器控制，禁用 PWM 发生器中断和 ADC 触发信号的能力。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 WO	14 WO	13 WO	12 WO	11 WO	10 WO	9 WO	8 WO	7 WO	6 WO	5 WO	4 WO	3 WO	2 WO	1 WO	0 WO
TDCMPCD	TDCMPCU	TDCMPBD	TDCMPBU	TDCMPAD	TDCMPAU	TDCNTL	TDCNTZ	IDCMPCD	IDCMPCU	IDCMPBD	IDCMPBU	IDCMPAD	IDCMPAU	IDCNTL	IDCNTZ

偏移地址：

PWM0_IDR : 0x0048

PWM1_IDR : 0x0088

PWM2_IDR : 0x00C8

域	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
15	TRDCMPCD	WO	0	触发禁用的计数器=比较器C向下 0：无触发ADC 1：当计数器倒计时模式在PWMn_COMPC的值相匹配禁止ADC触发脉冲输出。
14	TRDCMPCU	WO	0	触发禁用的计数器=比较器C向上 0：无触发ADCer 1：当计数器倒计时模式在PWMn_COMPB的值相匹配禁止ADC触发脉冲输出。
13	TRDCMPBD	WO	0	触发禁用的计数器=比较器B向下 0：无触发ADCr 1：当计数器倒计时模式在PWMn_COMPB的值相匹配禁止ADC触发脉冲输出。
12	TRDCMPBU	WO	0	触发禁用的计数器=比较器B向上 0：无触发ADCr 1：当计数器计数模式PWMn_COMPB的值相匹配禁止ADC触发脉冲输出。
11	TRDCMPAD	WO	0	触发禁用的计数器=比较器A向下 0：无触发ADCr 1：当计数器倒计时模式在PWMn_COMPA的值相匹配禁止ADC触发脉冲输出。
10	TRDCMPAU	WO	0	触发禁用的计数器=比较器A向上 0：无触发ADC 1：当计数器计数模式PWMn_COMPA的值相匹配禁止ADC触发脉冲输出。
9	TRDCNTLOAD	WO	0	触发禁用的计数器=装载 0：无ADC触发输出 1：当计数器的PWMn_LOAD寄存器匹配禁止ADC触发脉冲输出。
8	TRDCNTZERO	WO	0	触发禁用的计数器=0 0：无ADC触发输出 1：禁止ADC触发脉冲输出当计数器为0。

域	名称	读/写	默认值	描述
7	IDCMPCD	WO	0	中断禁用的计数器=比较C向下 0：无中断 1：生中断被禁止，当该计数器在PWMn_CMPC寄存器值的值相同时倒计时。
6	IDCMPCU	WO	0	中断禁用的计数器=比较C向上 0：无中断 1：当计数器在PWMn_CMPC寄存器值的值相同时计算了原始中断被禁用。
5	IDCMPBD	WO	0	中断禁用的计数器=比较器B向下 0：无中断 1：当计数器在PWM CMPB寄存器值的值相同时，掰着指头一个原始中断被禁用。
4	IDCMPBU	WO	0	中断禁用的计数器=比较器B向上 0：无中断 1：生中断被禁止，当该计数器在PWMn_CMPB寄存器值的值相同时计数。
3	IDCMPAD	WO	0	中断禁用的计数器=比较A向下 0：无中断 1：生中断被禁止，当该计数器在PWMn_CMPA寄存器值的值相同时倒计时。
2	IDCMPAU	WO	0	中断禁用的计数器=比较A向上 0：无中断 1：生中断被禁止，当该计数器在PWMn_CMPA寄存器值的值相同时计数。
1	IDCNTLOAD	WO	0	中断禁用的计数器=装载 0：无中断 1：当计数器在PWMn_LOAD寄存器值的值相匹配的原始中断被禁用。
0	IDCNTZERO	WO	0	中断禁用的计数器=0 0：无中断 1：当计数器是零原始中断被禁用。

4.8.3.15 PWMn_IMR -PWM0/ 1/2 中断屏蔽

该寄存器显示中断屏蔽和 ADC 触发信号的状态。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
TMCMPCD	TMCMPCU	TMCMPBD	TMCMPBU	TMCMPAD	TMCMPAU	TMCNTL	TMCNTZ	IMCMPCD	IMCMPCU	IMCMPBD	IMCMPBU	IMCMPAD	IMCMPAU	IMCNTL	IMCNTZ

偏移地址：

PWM0_IMR : 0x004C

PWM1_IMR : 0x008C

PWM2_IMR : 0x00CC

域	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
15	TRMCMPCD	WO	0	触发掩蔽计数=比较器C向下 0：ADC当计数器在倒计时模式在PWMn_COMPC值的匹配触发脉冲输出的中断将被屏蔽。 1：ADC触发脉冲输出中断使能
14	TRMCMPCU	WO	0	触发掩码计数器=比较器C向上 0：ADC当计数器计数模式PWMn_COMPC值的匹配触发脉冲输出的中断将被屏蔽。 1：ADC触发脉冲输出中断使能
13	TRMCMPBD	WO	0	触发掩码计数器=比较器B向下 0：ADC当计数器在倒计时模式在PWMn_COMPB值的匹配触发脉冲输出的中断将被屏蔽。 1：ADC触发脉冲输出中断使能
12	TRMCMPBU	WO	0	触发掩码计数器=比较器B向上 0：ADC当计数器计数模式PWMn_COMPB值的匹配触发脉冲输出的中断将被屏蔽。 1：ADC触发脉冲输出中断使能
11	TRMCMPAD	WO	0	触发掩码计数器=比较器A向下 0：ADC当计数器在倒计时模式在PWMn_COMPA值的匹配触发脉冲输出的中断将被屏蔽。 1：ADC触发脉冲输出中断使能
10	TRMCMPAU	WO	0	触发掩码计数器=比较器A向上 0：ADC当计数器计数模式PWMn_COMPA值的匹配触发脉冲输出的中断将被屏蔽。 1：ADC触发脉冲输出中断使能
9	TRMCNTLOAD	WO	0	触发掩码计数器=装载 0：ADC当计数器的PWMn_LOAD寄存器匹配触发脉冲输出的中断将被屏蔽。 1：ADC触发脉冲输出中断使能
8	TRMCNTZERO	WO	0	触发掩码计数器=零 0：ADC当计数器为0触发脉冲输出的中断将被屏蔽。 1：ADC触发脉冲输出中断使能

域	名称	读/写	默认值	描述
7	IMCMPCD	WO	0	中断屏蔽位为计数器=比较C向下 0：当计数器在PWMn_CMPC寄存器值的值相同时，掰着指头一个原始的中断将被屏蔽。 1：生中断使能。
6	IMCMPCU	WO	0	中断屏蔽位为计数器=比较C向上 0：当计数器在PWMn_CMPC寄存器值的值相同时计算了原始的中断将被屏蔽。 1：生中断使能。
5	IMCMPBD	WO	0	中断屏蔽计数器=比较器B向下 0：当计数器在PWMn_CMPB寄存器值的值相同时，掰着指头一个原始的中断将被屏蔽。 1：生中断使能。
4	IMCMPBU	WO	0	中断屏蔽计数器=比较器B向上 0：当计数器在PWMn_CMPB寄存器值的值相同时计算了原始的中断将被屏蔽。 1：生中断使能。
3	IMCMPAD	WO	0	中断屏蔽位为计数器=比较A向下 0：当计数器在PWMn_CMPA寄存器值的值相同时，掰着指头一个原始的中断将被屏蔽。 1：生中断使能。
2	IMCMPAU	WO	0	中断屏蔽位为计数器=比较A向上 0：当计数器在PWMn_CMPA寄存器值的值相同时计算了原始的中断将被屏蔽。 1：生中断使能。
1	IMCNTLOAD	WO	0	中断屏蔽计数器=装载 0：当计数器在PWMn_LOAD寄存器值的值相匹配的原始中断将被屏蔽。 1：生中断使能。
0	IMCNTZERO	WO	0	中断屏蔽计数器=零 0：当计数器是零原始中断将被屏蔽。 1：生中断使能。

4.8.3.16 PWMn_RIS -PWM0/ 1/2 的原始中断状态

这些寄存器提供的当前集，而不管他们是否会引起中断被断言给控制器被断言的中断源。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
								CMPCDRI	CMPCURI	CMPBDRI	CMPBURI	CMPADRI	CMPAURI	CNTLRI	CNTZRI

偏移地址：

PWM0_RIS : 0x0050

PWM1_RIS : 0x0090

PWM2_RIS : 0x00D0

域	名称	读/写	默认值	描述
31:8	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
7	CMPCDRI	RO	0	比较器C向下的原中断状态 0：无中断 1：当倒计时计数器已PWMn_COMPC匹配的值。
6	CMPCURI	RO	0	比较器C向上的原中断状态 0：没有中断 1：当计数，计数器已PWMn_COMPC匹配的值。
5	CMPBDRI	RO	0	比较器B向下的原中断状态 0：没有中断 1：当倒计时计数器已PWMn_COMPB匹配的值。
4	CMPBURI	RO	0	比较器B向上的原中断状态 0：没有中断 1：当计数，计数器已PWMn_COMPB匹配的值。
3	CMPADRI	RO	0	比较器A向下的原中断状态 0：没有中断 1：当计数，计数器已PWMn_COMPA匹配的值。
2	CMPAURI	RO	0	比较器A向上的原中断状态 0：没有中断 1：当计数，计数器已PWMn_COMPA匹配的值。
1	CNTLOADRI	RO	0	计数器=装载原始中断状态 0：无中断 1：计数器已匹配在PWMn_LOAD的值。
0	CNTZERORI	RO	0	计数器=零原始中断状态 0：无中断 1：计数器匹配为零。

4.8.3.17 PWMn_ISC -PWM0/ 1/2 的中断状态和清除

注：该寄存器读写以清除注册。“1”个别位的写操作会清除相应的中断状态。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 R/W1C	6 R/W1C	5 R/W1C	4 R/W1C	3 R/W1C	2 R/W1C	1 R/W1C	0 R/W1C
								CMPCDIS	CMPCUIS	CMPBDRI	CMPBUIIS	CMPADIS	CMPAUIIS	CNTLIS	CNTZIS

偏移地址：

PWM0_ISC : 0x0054

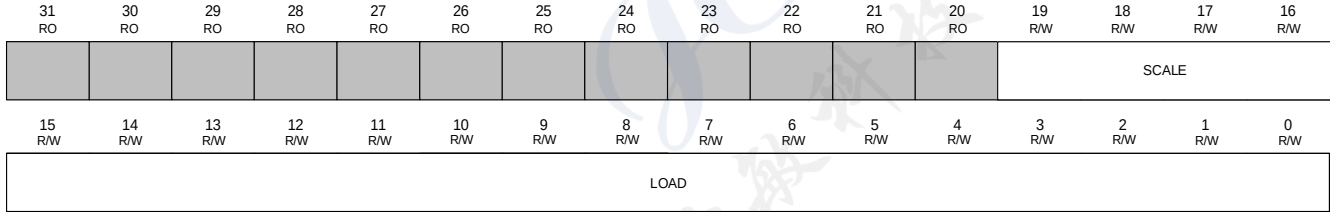
PWM1_ISC : 0x0094

PWM2_ISC : 0x00D4

域	名称	读/写	默认值	描述
31:8	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
7	CMPCDIS	R/W1C	0	比较器C向下中断状态和清除 0：无中断或中断被屏蔽。 1：比较器C向下中断已发出信号。
6	CMPCUIS	R/W1C	0	比较器C向上中断状态和清除 0：无中断或中断被屏蔽。 1：比较器C向上中断已发出信号。
5	CMPBDIS	R/W1C	0	比较器B向下中断状态和清除 0：无中断或中断被屏蔽。 1：比较器B向下中断已发出信号。
4	CMPBUIIS	R/W1C	0	比较器B向上中断状态和清除 0：无中断或中断被屏蔽。 1：比较器B向上中断已发出信号。
3	CMPADIS	R/W1C	0	比较器A向下中断状态和清除 0：无中断或中断被屏蔽。 1：比较器A向下中断已发出信号。
2	CMPAUIIS	R/W1C	0	比较器A向上中断状态和清除 0：无中断或中断被屏蔽。 1：比较器A向上中断已发出信号。
1	CNTLOADIS	R/W1C	0	计数器=负载中断状态和清除 0：无中断或中断被屏蔽。 1：计数器相匹配的负载值，中断被通知
0	CNTZEROIS	R/W1C	0	计数器=零中断状态和清除 0：无中断或中断被屏蔽。 1：计数器已匹配零，中断已发出信号。

4.8.3.18 PWMn_LOAD - PWM0/ 1/2 的装载值

这些寄存器包含了 PWM 计数器的装载值。根据不同的计数器模式，当它达到零或使用此值作为限制值后该计数器被清零或向下计数到零计数或这个值被装入计数器。如果负载更新模式是即时则该值用于下一次计数器达到零。否则，如果负载更新模式被同步然后在下一次计数器达到零和一个同步信号是活动的，那么将使用该值。



偏移地址：

PWM0_LOAD : 0x0058

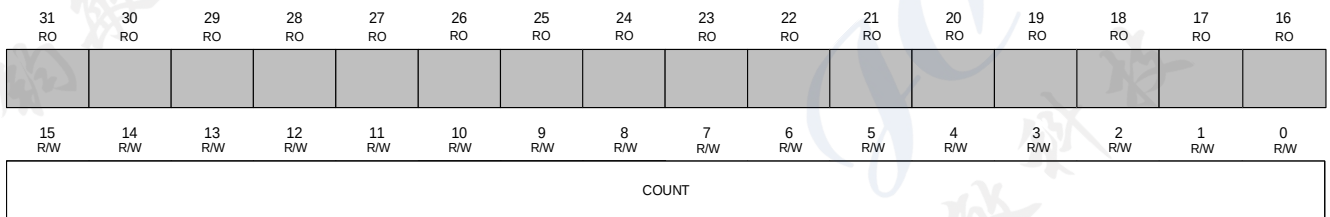
PWM0_LOAD : 0x0098

PWM0_LOAD : 0x00D8

域	名称	读/写	默认值	描述
31:20	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
19:16	<i>SCALE</i>	R/W	0x0	计数器控制规模从0到15
15:0	<i>LOAD</i>	R/W	0x0	计数器负载值。

4.8.3.19 PWMn_COUNT -PWM0/ 1/2 计数器

这些寄存器包含 PWM 计数器的当前值。当等于负载寄存器值的计数值，就会产生一个脉冲;使用代(通过 PWMn_GENA/ PWMn_GENB 寄存器)的 PWM 信号来驱动或驱动 ADC 触发(通过 PWMn_INTEN 寄存器)中断。脉冲与当该值为零相同的能力产生的。



偏移地址：

PWM0_COUNT : 0x005C

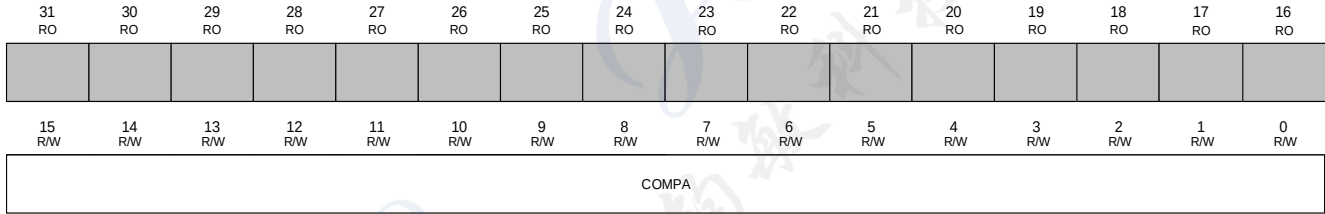
PWM0_COUNT : 0x009C

PWM0_COUNT : 0x00DC

域	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
15:0	<i>COUNT</i>	R/W	0x0	计数器值 计数器的当前值。

4.8.3.20 PWMn_COMP A -PWM0/ 1/2 比较器 A

这些寄存器包含的值与计数器进行比较。当此值的计数器相匹配时，一个脉冲是可驱动输出;一个)的 PWM 信号的生成，B) 的中断信号，C) ADC 触发信号。如果该值大于 PWMn_LOAD 寄存器大于无脉冲是有史以来输出。如果负载更新模式是即时则该值用于下一次计数器达到零。否则，如果负载更新模式被同步然后在下一次计数器达到零和一个同步信号是活动的，那么将使用该值。



偏移地址：

PWM0 COMP A : 0x0060

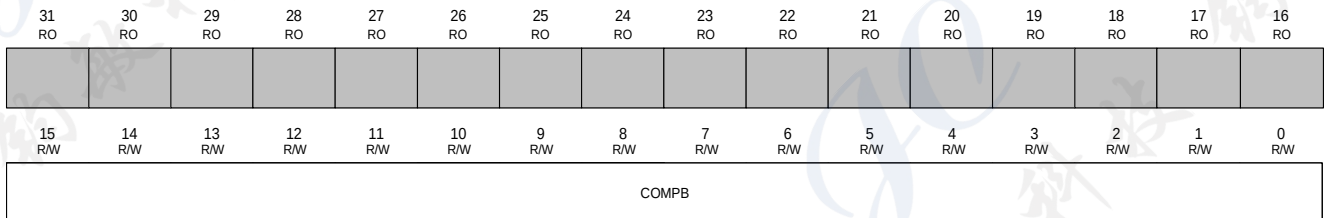
PWM0 COMP A : 0x00A0

PWM0 COMP A : 0x00E0

域	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
15:0	COMP A	R/W	0x0	比较器A的值 该值和计数器值比较

4.8.3.21 PWMn_COMP B -PWM0/ 1/2 比较器 B

该寄存器执行相同的功能如 PWMn_COMP A 但仅用于比较器 B。



偏移地址：

PWM0_COMP B : 0x0064

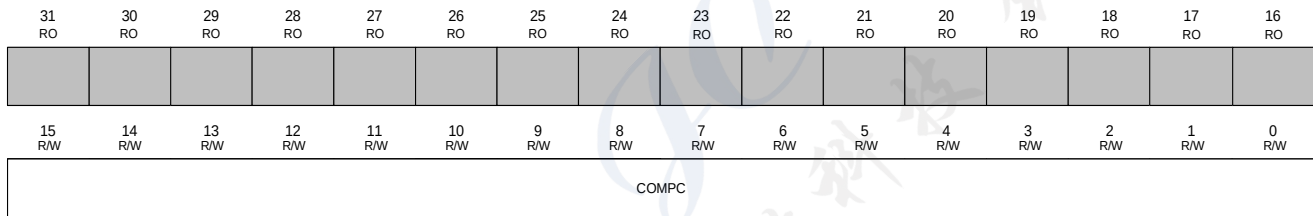
PWM0_COMP B : 0x00A4

PWM0_COMP B : 0x00E4

域	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
15:0	COMP B	R/W	0x0	比较器B的值 该值和计数器值比较

4.8.3.22 PWMn_COMPC -PWM0/ 1/2 比较器 C

该寄存器执行相同的功能如 PWMn_COMPA 但仅用于比较器 C.



偏移地址：

PWM0_COMPC : 0x0068

PWM0_COMPC : 0x00A8

PWM0_COMPC : 0x00E8

域	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
15:0	<i>COMPB</i>	R/W	0x0	比较器B的值 该值和计数器值比较

4.8.3.23 PWMn_GENA -PWM0/1/2 发生控制器 A

这些寄存器控制的基础上的负载，并从计数器输出零脉冲事件 PWMn_a 信号生成，以及来自比较器的比较 A 和比较脉冲 B 事件。对于减计数模式计数器或递增计数模式只有四个事件发生。对于加/减计数模式计数器所有六个事件发生。这些事件允许在 PWM 信号的产生更大的灵活性。如果零或装载事件与比较 A 或比较 B 事件相吻合，零或负载采取行动无视他人。如果比较 A 事件与比较 B 事件同时发生，比较一采取行动无视比较 B 的动作。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 R/W	14 R/W	13 R/W	12 R/W	11 R/W	10 R/W	9 R/W	8 R/W	7 R/W	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
ACTCMPCD		ACTCMPCU		ACTCMPBD		ACTCMPBU		ACTCMPAD		ACTCMPAU		ACTCNTL		ACTCNTZ	

偏移地址：

PWM0_GENA : 0x006C

PWM0_GENA : 0x00AC

PWM0_GENA : 0x00EC

域	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
15:14	<i>ACTCMPCD</i>	R/W	0x0	比较器C向下作用 要采取的动作当计数器比较C将只匹配而倒计时。同样的动作表作为ACT零。
13:12	<i>ACTCMPCU</i>	R/W	0x0	比较器C向上作用 要采取的动作当计数器比较C将只匹配同时计数。同样的动作表作为ACT零。
11:10	<i>ACTCMPBD</i>	R/W	0x0	比较器B向下作用 要采取的动作当计数器比较B将只匹配同时计数。同样的动作表作为ACT零。
9:8	<i>ACTCMPBU</i>	R/W	0x0	比较器B向上作用 要采取的动作当计数器比较B将只匹配同时计数。同样的动作表作为ACT零。
7:6	<i>ACTCMPAD</i>	R/W	0x0	比较器A向下作用 要采取的动作当计数器比较A将只匹配同时计数。同样的动作表作为ACT零。
5:4	<i>ACTCMPAU</i>	R/W	0x0	比较器A向上作用 要采取的动作当计数器比较A将只匹配同时计数。同样的动作表作为ACT零。
3:2	<i>ACTLOAD</i>	R/W	0x0	计数器=荷载作用 当计数器的负载值匹配时要采取的行动。同样的动作表作为ACT零。
1:0	<i>ACTZERO</i>	R/W	0x0	计数器=0 当计数器为零时将采取的行动。 动作表： 00：不执行任何 01：反转输出信号 10：输出信号设置为0。 11：输出信号设置为1。

4.8.3.24 PWMn_GENB -PWM0/ 1/2 发生控制器 B

该寄存器执行相同的功能如 PWMn_GENA 但仅用于发生控制器 B.

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 R/W	14 R/W	13 R/W	12 R/W	11 R/W	10 R/W	9 R/W	8 R/W	7 R/W	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
ACTCMPCD		ACTCMPCU		ACTCMPBD		ACTCMPBU		ACTCMPAD		ACTCMPAU		ACTCNTL		ACTCNTZ	

偏移地址：

PWM0_GENB : 0x0070

PWM0_GENB : 0x00B0

PWM0_GENB : 0x00F0

域	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
15:14	ACTCMPCD	R/W	0x0	比较器C向下作用 要采取的动作当计数器比较C将只匹配而倒计时。同样的动作表作为ACT零。
13:12	ACTCMPCU	R/W	0x0	比较器C向上作用 要采取的动作当计数器比较C将只匹配同时计数。同样的动作表作为ACT零。
11:10	ACTCMPBD	R/W	0x0	比较器B向下作用 要采取的动作当计数器比较B将只匹配同时计数。同样的动作表作为ACT零。
9:8	ACTCMPBU	R/W	0x0	比较器B向上作用 要采取的动作当计数器比较B将只匹配同时计数。同样的动作表作为ACT零。
7:6	ACTCMPAD	R/W	0x0	比较器A向下作用 要采取的动作当计数器比较A将只匹配同时计数。同样的动作表作为ACT零。
5:4	ACTCMPAU	R/W	0x0	比较器A向上作用 要采取的动作当计数器比较A将只匹配同时计数。同样的动作表作为ACT零。
3:2	ACTLOAD	R/W	0x0	计数器=荷载作用 当计数器的负载值匹配时要采取的行动。同样的动作表作为ACT零。
1:0	ACTZERO	R/W	0x0	计数器=0 当计数器为零时将要采取的行动。 动作表： 00：不执行任何 01：反转输出信号 10：输出信号设置为0。 11：输出信号设置为1。

4.8.3.25 PWMn_DBCTL -PWM0/ 1/2 死区控制

這個寄存器控制著死区发生器,在 pwmA 和 pwmB 信号的基础上产生 PWMn_A 和 PWMn_B 信号。当禁能之后,pwmn_a 直接传递为 pwmn_a'信号,pwmn_b 直接传递为 pwmn_b'信号。使能死区控制时,pwmn_b 信号被忽略,pwmn_a'信号通过对 pwmn_a 信号的上升沿延迟 PWMn_DBRIS 寄存器中指定的值来产生,pwmn_b'信号通过翻转 pwmn_a 信号并对 pwmn_a 信号的下降沿延迟 PWMn_DBFALL 寄存器中指定的值来产生。输出控制模块在 PWM0_A 引脚输出 pwm0_a'信号,并在 PWM0_B 引脚输出 pwm0_b' 信号。其他的 PWM 信号也以相同方式产生。

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

偏移地址：

PWM0_DBCTL : 0x0074

PWM0_DBCTL : 0x00B4

PWM0_DBCTL : 0x00F4

域	名称	读/写	默认值	描述
31:16	reserved	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性,在此值不应该被写入或读出。
15:0	ENABLE	R/W	0x0	死区发生器启用 0:直接传递PWM信号。 1:死区发生器插入死区到输出信号。

4.8.3.26 PWMn_DBRIS - PWM0/ 1/2 死区上升延迟

该寄存器包含时钟滴答数生成 pwmn_a'时延迟 pwmn_a 信号的上升沿。如果死区发生器被禁用该寄存器被忽略。如果该寄存器的值是比在输入 PWM 信号中的高脉冲宽度越高,上升沿延迟消耗导致对输出没有高时的信号的整个高的时间。以确保输入高时间总是长于上升沿延迟是很重要的。

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

偏移地址：

PWM0_DBRIS : 0x0078

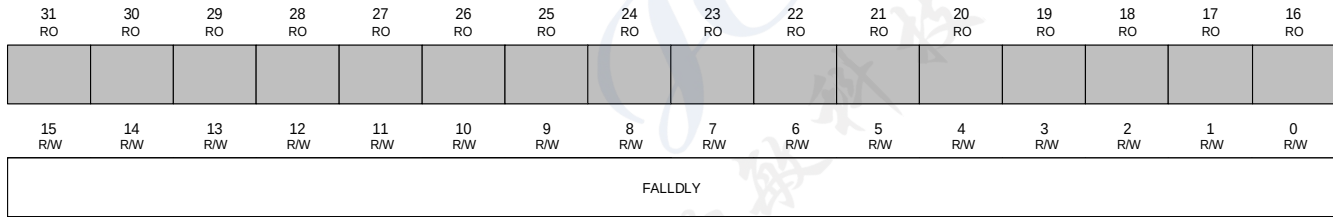
PWM0_DBRIS : 0x00B8

PWM0_DBRIS : 0x00F8

域	名称	读/写	默认值	描述
31:16	reserved	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性,在此值不应该被写入或读出。
15:0	RISEDLY	R/W	0x0	死区上升延迟 时钟周期数来延迟上升沿。

4.8.3.27 PWMn_DBFALL -PWM0/ 1/2 死区下降延迟

该寄存器包含时钟滴答数生成 `pwmn_b'` 时延迟 `pwmn_a` 信号的下降沿。如果死区发生器被禁用该寄存器被忽略。如果该寄存器的值是比在输入 PWM 信号中的低脉冲的宽度越大,下降沿延迟消耗导致对输出没有低时间的信号的整个低时间。以确保输入低时间总是长于下降沿延迟是很重要的。



偏移地址：

PWM0_DBFALL : 0x007C

PWM1_DBFALL : 0x00BC

PWM1_DBFALL : 0x00FC

域	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性,在此值不应该被写入或读出。
15:0	<i>FALLDLY</i>	R/W	0x0	死区延时 时钟滴答的数目,以延缓下降的边缘。

4.9 模拟/数字转换器 (ADC)

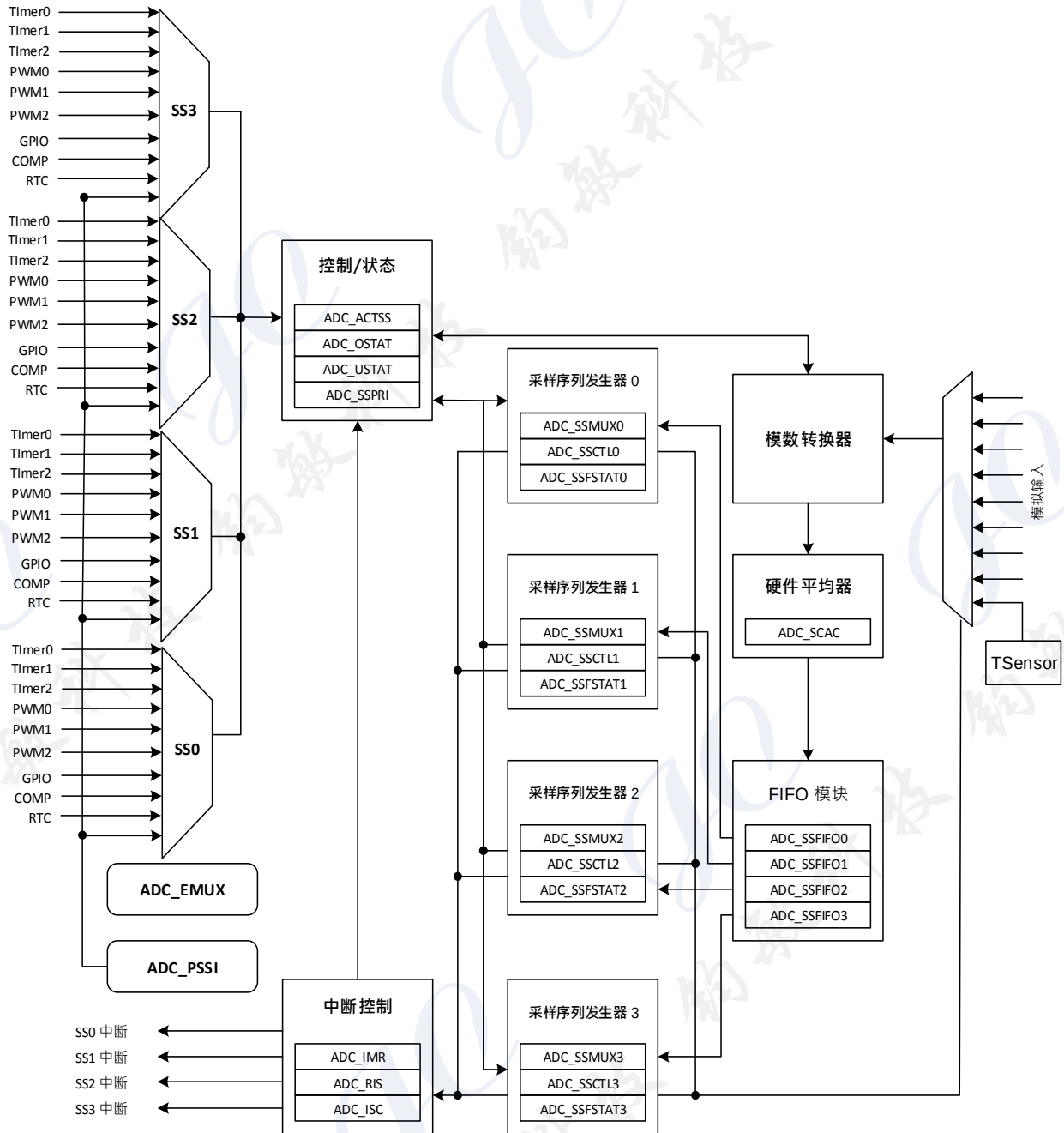
一个模拟/数字转换器 (ADC) 是转换一个连续的模拟电压的离散数字量的外设。该 PT32U301 包含一个 12 位逐次逼近型模拟数字转换器 (SAR A/D 转换器)，8 个外部输入通道，以及一个内部温度传感器。每个 ADC 模块支持四个可编程程序器，允许对多个模拟输入源无需使用控制器采样。A/D 转换器支持三种工作模式：单模，单周期扫描模式和连续扫描模式。每个采样序列完全可配置的输入源，触发事件，中断灵活的编程。

ADC 模块提供了以下功能：

- 模拟输入电压范围：0 至基准值（高达 3.3V）。
- 最多 8 个单端模拟输入通道或 4 个差分模拟输入通道。
- 支持可编程的采样时间（与 ADC_CLK 单位），通道 0 - 3 采样时间可以单独编程。
- 支持省电模式。
- 支持普通模式。
- 支持 PGA 控制模式。
- 片上内置温度传感器，支持温度感应控制模式。
- 最大 ADC 时钟频率为 48 MHz，每次转换时的采样时间决定的 20 个时钟+输入阻抗。
- 灵活的触发控制
 - 控制器（软件）
 - 定时器 0/1/2
 - 比较器
 - PWM0/1/2
 - GPIO
 - RTC
- 三种操作模式
 - 单模：A/D 转换是在一个指定的信道进行一次。
 - 单周期扫描模式：A/D 转换是在与来自最低编号的信道的最高编号的信道的序列中的所有指定的频道上执行一个周期。
 - 连续扫描模式：直到软件停止 A/D 转换 A/D 转换器连续执行单周期扫描模式。

4.9.1 块图

图 4.9-1: ADC 框图



4.9.2 功能描述

PT32U301 ADC 采集的样本数据，通过一个可编程的序列为基础的方法代替传统的单或双采样方法。每个样本序列是一个完全编程系列连续取样，使 ADC 从多个输入源不受微处理器服务收集数据。

4.9.2.1 采样序列发生器

采样控制和数据捕获由采样序列发生器进行处理所有序列，发生器的实现方法都是相同的，区别仅在于能够捕捉的采样数以及 FIFO 深度有所不同。下表显示了各序列发生器可以支持和其相应的 FIFO 深度的最大采样数。

表 4.9-1: 序列发生器的样品和 FIFO 深度

采样序列器	最大采样次数	FIFO 深度
SS0	8	8
SS1	4	4
SS2	4	4
SS3	1	1

对于一个给定的样本序列，每个样品由 ADC 采样序列输入复用选择寄存器 (ADC_SSMUXx)和 ADC 采样序列控制寄存器(ADC_SSCTLx)寄存器中的位域定义。ADC_SSMUXx 选择输入引脚，而 ADC_SSCTLx 包含样本控制位，对应的参数如温度传感器的选择、中断使能，末端序列，和差分输入方式。采样序列发生器可以通过置位 ADC 活动采样序列发生器 (ADC_ACTSS) 寄存器中相应的 SSxEN 位进行启用。软件可通过置位 ADC 处理器采样序列启动 (ADC_PSSI) 寄存器的 SSxI 位来启动采样。

当配置采样序列，允许同一序列中的多个采样动作对同一输入端进行采样。在 ADC_SSCTLx 寄存器，IEx 位可针对任意采样动作组合置位，如此可在必要时允许在采样序列的每个采样动作后产生中断。另外，ENDx 位可以在采样序列内的任何一点进行设置。例如，如果使用序列发生器 0，END 位可以在与第五样本相关联的半字节设定，从而使采样序列 0 在完成第 5 个采样动作后结束整个采样序列。

样本序列执行完成后，数据结果可以从 ADC 采样序列检索结果 FIFO 寄存器(ADC_SSFI0x)。ADC 模块的 FIFO 均为简单的环型缓冲区，反复读取同一地址 (ADC_SSFI0x) 即可依次“弹出”结果数据。为了方便软件调试，通过 ADC 采样序列 FIFO 状态 (ADC_SSSTATx) 寄存器可查询到 FIFO 头指针和尾指针的位置以及 FULL 和 EMPTY 状态标志。如果 FIFO 已满，再进行写操作时，该写操作会失败，该 FIFO 会出现上溢状况。通过 ADC_OSTAT 和 ADC_USTAT 寄存器可监控上溢和下溢状态。

4.9.2.2 模块时钟

该模块的时钟与系统时钟源相同。内部 ADC 的最大支持采样率是将输入的模拟信号转换，以 20 个 ADC 周期。ADC 的最大采样速率为 2.4MHz 使用 48MHz 的时钟源选择。ADC 可以通过设置 ADC 初步控制寄存器(ADC_INI)的 CLKDIV 领域采用较低的时钟源。

4.9.2.3 样本优先级

当同时发生多个采样事件（触发条件）时，将按照 ADC 采样序列器优先级 (**ADC_SSPRI**) 寄存器中的值对它们进行排序和依次处理。优先级的有效值为 0-3，其中 0 代表最高优先级、3 代表最低优先级。如果多个活动的采样序列具有相同的优先级，将导致转换结果数据不连续，因此软件必须确保当前活动的所有采样序列各自具有唯一的优先级。

4.9.2.4 采样事件/触发器

每个采样序列发生器的采样触发条件均通过 ADC 事件多路复用器选择 (**ADC_EMUX**) 寄存器予以定义。触发事件源包括处理器触发（默认）、模拟比较器触发、GPIO ADC 控制 (**GPIO_ADTRIG**) 寄存器指定的 GPIO 外部信号触发、通用定时器触发、PWM 发生器触发以及持续采样触发。软件可以将 ADC 处理器采样序列启动 (**ADC_PSSI**) 的 **SSxI** 位置位来启动采样序列。

配置持续采样触发条件时务必慎重。假如某个采样序列的优先级过高，可能导致其它低优先级采样序列始终无法运行。通常，要将使用连续采样模式的采样序列器设置位最低优先级。当输入接口上的电压达到了某一特定值，连续采样可以和数字比较器配合使用以产生中断。

4.9.2.5 中断控制

采样序列寄存器配置决定哪些事件会产生原始中断，但没有中断是否真正被发送到中断控制器控制。在 PT32U301，在 ADC 的中断是由一组五个寄存器控制。

- 中断控制 (**IER**, **IDR**, **IMR**)

ADC 中断使能寄存器 (**ADC_IER**) 允许通过写'1'的中断请求线。同样，ADC 中断禁用寄存器 (**ADC_IDR**) 禁用写一个'1'的中断请求线。**IER** 和 **IDR** 是控制中断屏蔽只写寄存器。这两个寄存器的总体结果可以通过 ADC 中断显示屏蔽寄存器 (**ADC_IMR**)。IMR 是只读使用'1'或'0'来指示中断请求线被使能/或禁用寄存器。该寄存器控制原始中断信号是否被提升。

- 中断状态读取 (**RIS**)

ADC 原始中断状态(**ADC_RIS**)是一个只读寄存器，显示模块的所有原始中断信号。

- 中断清除 (**ISC**)

ADC 中断状态和中断清除寄存器 (**ADC ISC**) 用于指示模块的不可屏蔽的中断状态，因为只有现在屏蔽的中断被认定，以处理器。该寄存器写“1”到该位可以清除相应的中断状态或写 1 **IDR** 禁止中断。

4.9.2.6 平均电路

更高精度的结果可以使用硬件平均电路来产生，然而，改进的结果是吞吐率将成比例地降低。硬件采样平均电路最高可将 64 次采样结果累加并计算出平均值，以平均值作为单次采样的数据写入序列发生器 FIFO 的 1 个单元中。由于是算术平均值，因此吞吐率与求平均值的采样数目成反比。例如，若取 16 次采样进行平均值计算，那么吞吐率将降为 1/16。

平均电路是默认关闭的，转换器捕捉的所有数据直接送入序列发生器的 FIFO 中。进行平均计算的硬件由 ADC 平均通道控制 (**ADC_AVGC**) 寄存器进行控制。

4.9.3 初始化和配置

4.9.3.1 模块初始化

为了要使用的 ADC 模块，必须使能 PLL 和编程。使用不支持的频率可以在 ADC 模块中导致错误操作。ADC 模块的初始化是一个简单的过程：使能 ADC 时钟，复用输入 IO 到模拟输入，和重新配置采样序列发生器的优先级。ADC 的初始化顺序如下：

1. 在系统控制寄存器中使能 ADC 时钟：致能 APB 外设门控时钟寄存器(**SC_GCLK_APB**)的 ADC。
2. 设定 **GPIOx_AFRH** 和 **GPIOx_AFRL** 启用相应 GPIO 脚位为 ADC 应用。参考多功能引脚功能选择，去了解哪些脚位需要被配置。
3. 如果应用要求，重新设置 **ADC_SSPRI 寄存器** 中的采样序列的优先级。默认配置有采样序列发生器 0 的优先级最高，采样序列发生器 3 作为优先级最低。

4.9.3.2 采样序列发生器的配置

对于每个采样序列发生器的结构如下：

1. 将对应的 **ADC_ACTSS** 寄存器的 **SSxEN** 位清零，禁用采样序列发生器。采样序列器不用使能也可以进行配置。不过如果在配置期间禁用采样序列发生器，可以有效防止在此期间因满足触发条件而造成的误执行。
2. 配置在 **ADC_EMUX** 寄存器中的采样序列的触发事件。
3. 在 **ADC_SSMUXx** 寄存器中为采样序列的每个采样配置相应的输入源。
4. 针对采样序列中的每个采样动作，对 **ADC_SSCTLx** 寄存器中相应半字节的采样控制位进行配置。在配置最后一个半字节时，应确保 **END** 位置位。如果 **END** 不置位，将导致不可预测的执行结果。
5. 如果要使用中断，在 **ADC_IER/IDR** 寄存器设置相应的屏蔽位。
6. 通过设置 **ADC_ACTSS** 寄存器相应的 **SSxEN** 位启用采样序列逻辑。

4.9.4 ADC 寄存器映射

基地址：0x4000_C000

偏移地址	寄存器名称	读/写	默认值	描述	参考页
0x0000	ADC_ACTSS	R/W	0x0000_0010	采样发生器使能寄存器	159
0x0004	ADC_IER	WO	0x0000_0000	ADC 中断使能	160
0x0008	ADC_IDR	WO	0x0000_0000	ADC中断禁止	161
0x000C	ADC_IMR	RO	0x0000_0000	ADC中断屏蔽	162
0x0010	ADC_RIS	RO	0x0000_0000	ADC原始中断状态位	163
0x0014	ADC_ISC	R/W1C	0x0000_0000	ADC中断状态和清除位	164
0x0018	ADC_OSTAT	R/W1C	0x0000_0000	ADC 上溢标志	165
0x001C	ADC_EMUX	R/W	0x0000_0000	ADC 通道选择	166
0x0020	ADC_USTAT	R/W1C	0x0000_0000	ADC 下溢状态位	167
0x0028	ADC_SSPRI	R/W	0x0000_3210	ADC 采样序列优先级控制	168
0x002C	ADC_INI	R/W	0x0000_0000	ADC初始化控制	169
0x0030	ADC_PSSI	R/W	0x0000_0000	ADC采样序列发生器启动	170
0x0034	ADC_AVGC	R/W	0x0000_0000	ADC平均信道控制	171
0x0038	ADC_PGAC	R/W	0x0000_0000	ADC PGA控制	173
0x003C	ADC_DIFC	R/W	0x0000_0000	ADC差分控制	175
0x0040	ADC_TMPC	R/W	0x0000_0000	ADC温度控制	176
0x0044	ADC_FRF	W1C	0x0000_0000	ADC FIFO刷新	176
0x0048	ADC_WAIT	R/W	0x0000_0000	ADC等待计数器	177
0x0050	ADC_SSMUX0	R/W	0x0000_0000	ADC采样序列输入多路复用器选择0	178
0x0054	ADC_SSCTL0	R/W	0x2000_0000	ADC采样序列控制0	179
0x0058	ADC_SSFIFO0	RO	0x0000_0000	ADC 采样序列结果FIFO 0	181
0x005C	ADC_SSFSTAT0	RO	0x0000_0100	ADC 采样序列FIFO0状态	182
0x0070	ADC_SSMUX1	R/W	0x0000_0000	ADC采样序列输入多路复用器选择1	183
0x0074	ADC_SSCTL1	R/W	0x0000_2000	ADC采样序列控制1	184
0x0078	ADC_SSFIFO1	RO	0x0000_0000	ADC 采样序列结果FIFO 1	181
0x007C	ADC_SSFSTAT1	RO	0x0000_0100	ADC 采样序列FIFO1状态	182
0x0090	ADC_SSMUX2	R/W	0x0000_0000	ADC采样序列输入多路复用器选择2	183
0x0094	ADC_SSCTL2	R/W	0x0000_2000	ADC采样序列控制2	184
0x0098	ADC_SSFIFO2	RO	0x0000_0000	ADC 采样序列结果FIFO 2	181
0x009C	ADC_SSFSTAT2	RO	0x0000_0100	ADC 采样序列FIFO2状态	182
0x00B0	ADC_SSMUX3	R/W	0x0000_0000	ADC采样序列输入多路复用器选择3	185
0x00B4	ADC_SSCTL3	R/W	0x0000_0002	ADC采样序列控制3	184
0x00B8	ADC_SSFIFO3	RO	0x0000_0000	ADC 采样序列结果FIFO 3	181
0x00BC	ADC_SSFSTAT3	RO	0x0000_0100	ADC 采样序列FIFO3状态	182

4.9.4.1 ADC_ACTSS - ADC 有效采样序列

该寄存器控制采样序列的激活。每个采样序列发生器可以启用或禁用独立。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 R/W	2 R/W	1 R/W	0 R/W
												IDLE	SS3EN	SS2EN	SS1EN
														SS0EN	

偏移地址: 0x0000

域	名称	读/写	默认值	描述
31:5	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
4	<i>IDLE</i>	RO	1	ADC空闲状态 0：ADC忙 1：ADC处于闲置状态
3	<i>SS3EN</i>	R/W	0	ADC采样序列发生器3（SS3）启用 0：SS3被禁用。 1：SS3启用
2	<i>SS2EN</i>	R/W	0	ADC采样序列发生器2（SS2）启用 0：SS2被禁用。 1：SS2启用
1	<i>SS1EN</i>	R/W	0	ADC采样序列发生器1（SS1）启用 0：SS1被禁用。 1：SS1启用
0	<i>SS0EN</i>	R/W	0	ADC采样序列发生器0（SS0）启用 0：SS0被禁用。 1：SS0启用

4.9.4.2 ADC_IER - ADC 中断使能

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 WO	3 WO	2 WO	1 WO	0 WO
											TOIE	SS3IE	SS2IE	SS1IE	SS0IE

偏移地址: 0x0004

域	名称	读/写	默认值	描述
31:5	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
4	<i>TOIE</i>	RO	0	超时中断使能 1：超时中断使能。
3	<i>SS3IE</i>	WO	0	ADC采样序列发生器3（SS3）启用 1：SS3中断使能
2	<i>SS2IE</i>	WO	0	ADC采样序列发生器2（SS2）启用 1：SS2中断使能
1	<i>SS1IE</i>	WO	0	ADC采样序列发生器1（SS1）启用 1：SS1中断使能
0	<i>SS0IE</i>	WO	0	ADC采样序列发生器0（SS0）启用 1：SS0中断使能

4.9.4.3 ADC_IDR - ADC 中断屏蔽

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 WO	3 WO	2 WO	1 WO	0 WO
											TOID	SS3ID	SS2ID	SS1ID	SS0ID

偏移地址: 0x0008

域	名称	读/写	默认值	描述
31:5	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
4	<i>TOID</i>	RO	0	超时中断禁用 1：超时中断禁用。
3	<i>SS3ID</i>	WO	0	ADC采样序列发生器3（SS3）禁用 1：SS3中断被禁止
2	<i>SS2ID</i>	WO	0	ADC采样序列发生器2（SS2）禁用 1：SS2中断被禁止
1	<i>SS1ID</i>	WO	0	ADC采样序列发生器1（SS1）禁用 1：SS1中断被禁止
0	<i>SS0ID</i>	WO	0	ADC采样序列发生器0（SS0）禁用 1：SS0中断被禁止

4.9.4.4 ADC_IMR - ADC 中断屏蔽状态

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
											TOIM	SS3IM	SS2IM	SS1IM	SS0IM

偏移地址: 0x000C

域	名称	读/写	默认值	描述
31:5	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
4	<i>TOIM</i>	RO	0	超时中断屏蔽状态 0：超时中断将被屏蔽。 1：超时中断使能。
3	<i>SS3IM</i>	RO	0	ADC采样序列发生器3（SS3）中断屏蔽状态 0：SS3中断将被屏蔽。 1：SS3中断使能。
2	<i>SS2IM</i>	RO	0	ADC采样序列发生器2（SS2）中断屏蔽状态 0：SS2中断将被屏蔽。 1：SS2中断使能。
1	<i>SS1IM</i>	RO	0	ADC采样序列发生器1（SS1）中断屏蔽状态 0：SS1中断将被屏蔽。 1：SS1中断使能。
0	<i>SS0IM</i>	RO	0	ADC采样序列发生器0（SS0）中断屏蔽状态 0：SS0中断将被屏蔽。 1：SS0中断使能。

4.9.4.5 ADC_RIS - ADC 原始中断状态

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
											TORI	SS3RI	SS2RI	SS1RI	SS0RI

偏移地址: 0x0010

域	名称	读/写	默认值	描述
31:5	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
4	<i>TORI</i>	RO	0	超时原始中断状态 0：无中断。 1：超时中断断言。
3	<i>SS3RI</i>	RO	0	ADC采样序列发生器3（SS3）原始中断状态 0：无中断。 1：SS3中断断言。
2	<i>SS2RI</i>	RO	0	ADC采样序列发生器2（SS2）原始中断状态 0：无中断。 1：SS2中断断言。
1	<i>SS1RI</i>	RO	0	ADC采样序列发生器1（SS1）原始中断状态 0：无中断。 1：SS1中断断言。
0	<i>SS0RI</i>	RO	0	ADC采样序列发生器0（SS0）原始中断状态 0：无中断。 1：SS0中断断言。

4.9.4.6 ADC_ISC - 中断状态和清除

注：该寄存器读写以清除注册。“1”个别位的写操作会清除相应的中断。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RW1C	3 RW1C	2 RW1C	1 RW1C	0 RW1C
											TOIS	SS3IS	SS2IS	SS1IS	SS0IS

偏移地址: 0x0014

域	名称	读/写	默认值	描述
31:5	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
4	TOIS	RW1C	0	超时中断状态和清除 0：无中断或中断已被屏蔽。 1：超时中断已发出信号。
3	SS3IS	RW1C	0	ADC采样序列发生器3（SS3）中断状态和清除 0：无中断或中断已被屏蔽。 1：SS3中断已发出信号。
2	SS2IS	RW1C	0	ADC采样序列发生器2（SS2）中断状态和清除 0：无中断或中断已被屏蔽。 1：SS2中断已发出信号。
1	SS1IS	RW1C	0	ADC采样序列发生器1（SS1）中断状态和清除 0：无中断或中断已被屏蔽。 1：SS1中断已发出信号。
0	SS0IS	RW1C	0	ADC采样序列发生器0（SS0）中断状态和清除 0：无中断或中断已被屏蔽。 1：SS0中断已发出信号。

4.9.4.7 ADC_OSTAT - ADC 溢出状态

该寄存器指示采样序列发生器 FIFO 溢出条件。一旦溢出条件已经被软件处理，病情可以通过写 1 到相应的位位置被清除。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 R/W1C	2 R/W1C	1 R/W1C	0 R/W1C
												SS3OV	SS2OV	SS1OV	SS0OV

偏移地址: 0x0018

域	名称	读/写	默认值	描述
31:4	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
3	SS3OV	R/W1C	0	ADC采样序列发生器3（SS3）FIFO溢出状态 0：FIFO没有溢出。 1：SS3 FIFO有位溢出情况下，FIFO满且要求写操作。当检测到溢出时，最近一次的写被丢弃。
2	SS2OV	R/W1C	0	ADC采样序列发生器2（SS2）FIFO溢出状态 0：FIFO没有溢出。 1：SS2 FIFO有位溢出情况下，FIFO满且要求写操作。当检测到溢出时，最近一次的写被丢弃。
1	SS1OV	R/W1C	0	ADC采样序列发生器1（SS1）FIFO溢出状态 0：FIFO没有溢出。 1：SS1 FIFO有位溢出情况下，FIFO满且要求写操作。当检测到溢出时，最近一次的写被丢弃。
0	SS0OV	R/W1C	0	ADC采样序列发生器0（SS0）FIFO溢出状态 0：FIFO没有溢出。 1：SS0 FIFO有位溢出情况下，FIFO满且要求写操作。当检测到溢出时，最近一次的写被丢弃。

4.9.4.8 ADC_EMUX - ADC 事件多路转换器选择

该 ADC MUX 选择用来启动采样每个采样序列事件（触发）。每个采样序列可以用独特的触发源进行配置。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 R/W	14 R/W	13 R/W	12 R/W	11 R/W	10 R/W	9 R/W	8 R/W	7 R/W	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
EM3				EM2				EM1				EM0			

偏移地址: 0x001C

域	名称	读/写	默认值	描述
31:4	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
3	<i>EM3</i>	R/W1C	0	SS3触发选择 此字段用于选择SS3触发源。 同样触发选择表中EM0但只为SS3
2	<i>EM2</i>	R/W1C	0	SS2触发选择 此字段用于选择SS2触发源。 同样触发选择表中EM0但只为SS2
1	<i>EM1</i>	R/W1C	0	SS1触发选择 此字段用于选择SS1触发源。 同样触发选择表中EM0但只为SS1
0	<i>EM0</i>	R/W1C	0	SS0触发选择 此字段用于选择SS0触发源。 触发选择表如下 0x0：处理器（默认） 0x1：GPIO 0x2：始终样品 0x3：RTC 0x4：定时器0 0x5：定时器1 0x6：定时器2 0x7：保留 0x8：PWM0 0x9：PWM1 0xA：PWM2 0xB：保留 0xC：模拟比较器0 0xD：模拟比较器1 0xE：模拟比较器2 0xF：模拟比较器3

4.9.4.9 ADC_USTAT - ADC 下溢状态

该寄存器指示采样序列发生器 FIFO 溢条件。相应的溢条件通过写 1 到相关的位清零。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 R/W1C	2 R/W1C	1 R/W1C	0 R/W1C
												UV3	UV2	UV1	UV0

偏移地址: 0x0020

域	名称	读/写	默认值	描述
31:4	reserved	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
3	UV3	R/W1C	0	ADC采样序列发生器3 (SS3) FIFO下溢状态 0：那么FIFO没有溢。 1：SS3 FIFO有位下溢条件的FIFO空且要求读。有问题的读操作不会移动FIFO指针，并且返回0
2	UV2	R/W1C	0	ADC采样序列发生器2 (SS2) FIFO下溢状态 0：那么FIFO没有溢。 1：SS2 FIFO有位下溢条件的FIFO空且要求读。有问题的读操作不会移动FIFO指针，并且返回0
1	UV1	R/W1C	0	ADC采样序列发生器1 (SS1) FIFO下溢状态 0：那么FIFO没有溢。 1：SS1 FIFO有位下溢条件的FIFO空且要求读。有问题的读操作不会移动FIFO指针，并且返回0
0	UV0	R/W1C	0	ADC采样序列发生器0 (SS0) FIFO下溢状态 0：那么FIFO没有溢。 1：SS0 FIFO有位下溢条件的FIFO空且要求读。有问题的读操作不会移动FIFO指针，并且返回0

4.9.4.10 ADC_SSPRI - ADC 采样序列发生器优先级

该寄存器设置优先级为每个采样序列发生器的。复位后，序列发生器 0 的优先级最高，且序列发生器 3 优先级最低。在配置序列优先级，每个序列都必须有一个唯一的优先级 ADC 的正常工作。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 R/W	12 R/W	11 RO	10 RO	9 R/W	8 R/W	7 RO	6 RO	5 R/W	4 R/W	3 RO	2 RO	1 R/W	0 R/W
		SS3				SS2				SS1				SS0	

偏移地址: 0x0028

域	名称	读/写	默认值	描述
31:14	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
13:12	SS3	R/W	0x3	ADC采样序列发生器3 (SS3) 优先 同场的描述中SS0但只为SS3
11:10	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
9:8	SS2	R/W	0x2	ADC采样序列发生器2 (SS2) 优先 同场的描述中SS0但只为SS2
7:6	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
5:4	SS1	R/W	0x1	ADC采样序列发生器1 (SS1) 优先 同场的描述中SS0但只为SS1
3:2	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
1:0	SS0	R/W	0x0	ADC采样序列发生器0 (SS0) 优先 该字段包含一个二进制编码值，指定SS0的优先级编码。分配给音序器的优先级必须被唯一地映射。 0x0：最高优先级 ... 0x3中：最低优先级

4.9.4.11 ADC_INI -ADC 初始化控制

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 R/W	14 R/W	13 R/W	12 R/W	11 RO	10 RO	9 RO	8 RO	7 R/W	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
TESTEN								CLKDIV				REFVCTRL	RSTCTRL	CLKEN	CLKEN

偏移地址: 0x002C

域	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
15:8	<i>TESTEN</i>	R/W	0x0	ADC测试模式启用宏 0XED：启用宏测试模式。
7:4	<i>CLKDIV</i>	R/W	0x0	ADC转换器的时钟分频 0至15的分频比，其中0表示没有除法实现。
3	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
2	<i>RSTCTRL</i>	R/W	0x0	ADC FIFO复位控制 0：如果它不是空的FIFO将不会被复位 1：FIFO将被重置无论是空的，而启动新的采样序列。如果满足下列条件之一时复位才会采取行动： 当FIFO溢出或当到达最后的采样序列的端点。
1	<i>CLKEN</i>	R/W	0x1	ADC时钟使能 需要注意的是修改CLKDIV，用户应该首先此位设置为0，重新启用CLKDIV修改后此位。
0	<i>ADINEN</i>	RO	0x0	ADC模拟输入使能

4.9.4.12 ADC_PSSI -ADC 处理器采样序列启动

该寄存器为应用软件提供了一种机制，以便在采样序列发生器中启动采样。采样序列发生器可以单独启动，也可以组合启动。当多个序列同时触发时，ADC_SSPRI 中的优先级编码指示执行顺序。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 R/W	6 R/W	5 R/W	4 R/W	3 WO	2 WO	1 WO	0 WO
								TRITYP3	TRITYP2	TRITYP1	TRITYP0	SS3	SS2	SS1	SS0

偏移地址: 0x0030

域	名称	读/写	默认值	描述
31:8	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
7	<i>TRITYP3</i>	R/W	0	ADC采样序列发生器3（SS3）触发类型选择 同样的描述，在TRITYP0但只为SS3
6	<i>TRITYP2</i>	R/W	0	ADC采样序列发生器2（SS2）触发类型选择 同样的描述，在TRITYP0但只为SS2
5	<i>TRITYP1</i>	R/W	0	ADC采样序列发生器1（SS1）触发类型选择 同样的描述，在TRITYP0但只为SS1
4	<i>TRITYP0</i>	R/W	0	ADC采样序列发生器0（SS0）触发类型选择 0:边沿触发 1:电平触发
3	<i>SS3I</i>	WO	0	ADC采样序列发生器3（SS3）启动 同样的描述，在SS0但只为SS3
2	<i>SS2I</i>	WO	0	ADC采样序列发生器2（SS2）启动 同样的描述，在SS0但只为SS2
1	<i>SS1I</i>	WO	0	ADC采样序列发生器1（SS1）启动 同样的描述，在SS0但只为SS1
0	<i>SS0I</i>	WO	0	ADC采样序列发生器0（SS0）启动 1：如果序列处于ADC_ACTSS启用了SS0触发采样。

4.9.4.13 ADC_AVGC -ADC 平均通道控制

该寄存器控制硬件采样序列平均值逐个通道。

如果 AVG 为 0，将样品直接通过而没有任何求平均通过。

如果 AVG= 6，则 $2^6=64$ 个连续的 ADC 采样进行平均来生成序列发生器 FIFO 一个结果。

如果 AVG=7，它提供了一个不可预知的结果。

31 RO	30 R/W	29 R/W	28 R/W	27 RO	26 R/W	25 R/W	24 R/W	23 RO	22 R/W	21 R/W	20 R/W	19 RO	18 R/W	17 R/W	16 R/W
	CH7AVG				CH6AVG				CH5AVG				CH4AVG		
15 RO	14 R/W	13 R/W	12 R/W	11 RO	10 R/W	9 R/W	8 R/W	7 RO	6 R/W	5 R/W	4 R/W	3 RO	2 R/W	1 R/W	0 R/W
	CH3AVG				CH2AVG				CH1AVG				CH0AVG		

偏移地址: 0x0034

域	名称	读/写	默认值	描述
31	reserved	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
30:28	CH7AVG	R/W	0x0	信道7平均控制 同样的描述，在CH0AVG但仅限于通道 7
27	reserved	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
26:24	CH6AVG	R/W	0x0	信道6平均控制 同样的描述，在CH0AVG但仅限于通道 6
23	reserved	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
22:20	CH5AVG	R/W	0x0	信道5平均控制 同样的描述，在CH0AVG但仅限于通道5
19	reserved	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
18:16	CH4AVG	R/W	0x0	信道4平均控制 同样的描述，在CH0AVG但仅限于通道4
15	reserved	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
14:12	CH3AVG	R/W	0x0	信道3平均控制 同样的描述，在CH0AVG但仅限于通道3
11	reserved	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
10:8	CH2AVG	R/W	0x0	信道2平均控制 同样的描述，在CH0AVG但仅限于通道2
7	reserved	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
6:4	CH1AVG	R/W	0x0	信道1平均控制 同样的描述，在CH0AVG但仅限于通道1
3	reserved	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。

域	名称	读/写	默认值	描述
2:0	CH0AVG	R/W	0X0	信道0平均控制 指定将被应用到ADC采样硬件平均量。 0x0 没有硬件过采样 0x1 2倍过采样硬件 0x2 4倍过采样硬件 0x3 8倍过采样硬件 0x4 16倍过采样硬件 0x5 32倍过采样硬件 0x6 64倍过采样硬件 0x7 保留

4.9.4.14 ADC_GAINC - ADC PGA 增益控制

该寄存器控制 PGA（可编程增益放大器）的增益通过写入对应的通道值。所述 ADC_PGA 控制寄存器用于配置芯片上的非反相运算放大器的增益。每个 ADC 通道可以具有不同的增益值被分配为显示在 PGA 控制寄存器位映射。要改变 PGA 控制寄存器的 ADC 序必须首先停止写“0”到 ADC_ACTSS。然后更新 ADC_PGA 控制寄存器。随后，通过写“1”到 ADC_ACTSS。

由 ADC 寄存器 PGAC 控制器改变在不同的 PGA 模式下的增益值

- 非反向 PGA 模式:
- 差动 PGA 模式:

图 4.9-2: PGA 框图

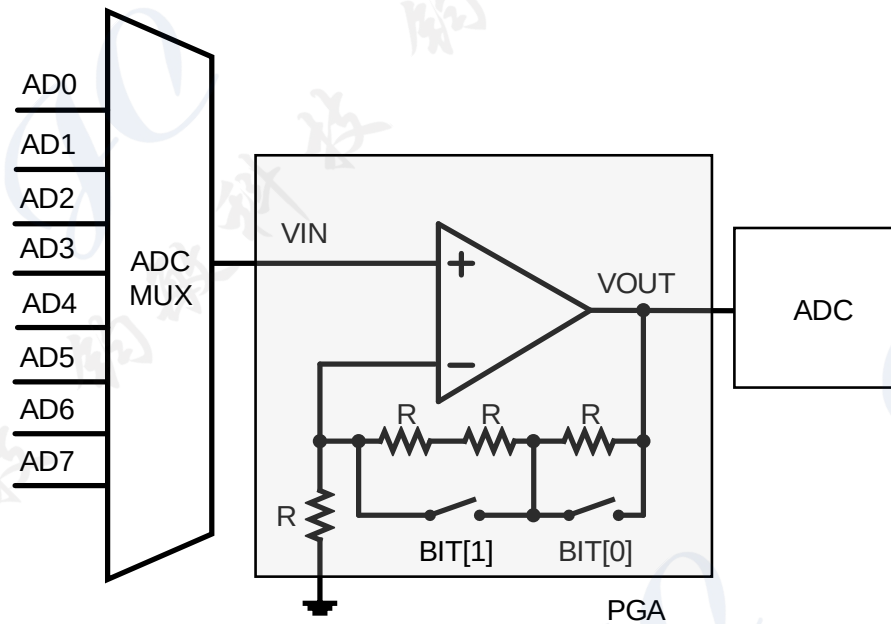


表 4.9-2: 不同的通道值的增益放大

通道值 (位[0],位[1])	增益放大
00	x1 (bypass)
01	x2
10	X3
11	X5

31 RO	30 RO	29 R/W	28 R/W	27 RO	26 RO	25 R/W	24 R/W	23 RO	22 RO	21 R/W	20 R/W	19 RO	18 RO	17 R/W	16 R/W
		CH7PGA				CH6PGA				CH5PGA				CH4PGA	
15 RO	14 RO	13 R/W	12 R/W	11 RO	10 RO	9 R/W	8 R/W	7 RO	6 RO	5 R/W	4 R/W	3 RO	2 RO	1 R/W	0 R/W
		CH3PGA				CH2PGA				CH1PGA				CH0PGA	

偏移地址: 0x0038

域	名称	读/写	默认值	描述
31:30	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
29:28	CH7PGA	R/W	0x0	通道7 PGA控制
27:26	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
25:24	CH6PGA	R/W	0x0	通道6 PGA控制
23:22	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
21:20	CH5PGA	R/W	0x0	通道5 PGA控制
19:18	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
17:16	CH4PGA	R/W	0x0	通道4 PGA控制
15:14	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
13:12	CH3PGA	R/W	0x0	通道3 PGA控制
11:10	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
9:8	CH2PGA	R/W	0x0	通道2 PGA控制
7:6	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
5:4	CH1PGA	R/W	0x0	通道1 PGA控制
3:2	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
1:0	CH0PGA	R/W	0x0	通道0 PGA控制

4.9.4.15 ADC_DIFC – ADC PGA 差分(Differential)控制

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 R/W	14 R/W	13 R/W	12 R/W	11 R/W	10 R/W	9 R/W	8 R/W	7 RO	6 RO	5 R/W	4 R/W	3 RO	2 RO	1 R/W	0 R/W
CH7INV	CH6INV	CH5INV	CH4INV	CH3INV	CH2INV	CH1INV	CH0INV			MVIP				MODE	

偏移地址: 0x003C

域	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
15	CH7INV	R/W	0x0	通道7反转控制 和CH0INV同样的描述，但只用于通道7。
14	CH6INV	R/W	0x0	通道6反转控制 和CH0INV同样的描述，但只用于通道6。
13	CH5INV	R/W	0x0	通道5反转控制 和CH0INV同样的描述，但只用于通道5。
12	CH4INV	R/W	0x0	通道4反转控制 和CH0INV同样的描述，但只用于通道4。
11	CH3INV	R/W	0x0	通道3反转控制 和CH0INV同样的描述，但只用于通道3。
10	CH2INV	R/W	0x0	通道2反转控制 和CH0INV同样的描述，但只用于通道2。
9	CH1INV	R/W	0x0	通道1反转控制 和CH0INV同样的描述，但只用于通道1。
8	CH0INV	R/W	0x0	通道0反转控制 0: 在通道0无反转 1: 通道0反转
7:6	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
5:4	MVIP	R/W	0x0	PGA MVIP 必须设置为0x0
3:2	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
1:0	MODE	R/W	0x0	PGA 模式控制 00: 非反相PGA模式 10: ADC差分模式 11: 差分PGA模式

4.9.4.16 ADC_TMC - ADC 温度控制

该寄存器控制 ADC 温度检测控制。

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

偏移地址: 0x0040

域	名称	读/写	默认值	描述
31:7	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
6:4	<i>TMPAVG</i>	R/W	0x0	温度平均控制
3:2	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
1	<i>TMPCHSW</i>	R/W	0	温度通道交换机 1：切换所有ADC通道温度输入。
0	<i>TMPEN</i>	R/W	0	温度模块启用 1：允许温度模块。

4.9.4.17 ADC_FRF - ADC FIFO 更新

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 W1C	2 W1C	1 W1C	0 W1C
												SS3RF	SS2RF	SS1RF	SS0RF

偏移地址: 0x0044

域	名称	读/写	默认值	描述
31:4	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
3	<i>SS3RF</i>	W1C	0	ADC采样序列发生器3 (SS3) FIFO刷新 1：SS3 FIFO复位
2	<i>SS2RF</i>	W1C	0	ADC采样序列发生器2 (SS2) FIFO刷新 1：SS2 FIFO复位
1	<i>SS1RF</i>	W1C	0	ADC采样序列发生器1 (SS1) FIFO刷新 1：SS1 FIFO复位
0	<i>SS0RF</i>	W1C	0	ADC采样序列发生器0 (SS0) FIFO刷新 1：SS0 FIFO复位

4.9.4.18 ADC_WAIT - ADC 等待计数器

该寄存器控制一次 ADC 转换结束到下一次 ADC 转换启动的延时时间。

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

偏移地址: 0x0044

域	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
15:8	<i>CNTINI</i>	W1C	0	ADC等待计数器初始值
7:0	<i>CNT</i>	W1C	0	ADC等待计数器值 要adc_end设置的等待周期udc_start.。

4.9.4.19 ADC_SSMUX0 - ADC 采样序列多路输入选择器 0

该寄存器定义了采样序列发生器 0 这个寄存器为 32 位宽，并且包含 8 个可能采样的信息执行序列的每个采样的模拟输入配置

31 RO	30 R/W	29 R/W	28 R/W	27 RO	26 R/W	25 R/W	24 R/W	23 RO	22 R/W	21 R/W	20 R/W	19 RO	18 R/W	17 R/W	16 R/W
	MUX7				MUX6				MUX5				MUX4		
15 RO	14 R/W	13 R/W	12 R/W	11 RO	10 R/W	9 R/W	8 R/W	7 RO	6 R/W	5 R/W	4 R/W	3 RO	2 R/W	1 R/W	0 R/W
	MUX3				MUX2				MUX1				MUX0		

偏移地址: 0x0050

域	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
30:28	<i>MUX7</i>	R/W	0x0	第8个采样输入选择 在采样序列发生器所执行序列的第8个采样过程中，此字段使用。它指定该模拟输入被用于模拟 - 数字转换采样。
27	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
26:24	<i>MUX6</i>	R/W	0x0	第7个采样输入选择 在采样序列发生器所执行序列的第8个采样过程中，此字段使用。它指定该模拟输入被用于模拟 - 数字转换采样。
23	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
22:20	<i>MUX5</i>	R/W	0x0	第6个采样输入选择 在采样序列发生器所执行序列的第8个采样过程中，此字段使用。它指定该模拟输入被用于模拟 - 数字转换采样。
19	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
18:16	<i>MUX4</i>	R/W	0x0	第5个采样输入选择 在采样序列发生器所执行序列的第8个采样过程中，此字段使用。它指定该模拟输入被用于模拟 - 数字转换采样。
15	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
14:12	<i>MUX3</i>	R/W	0x0	第4个采样输入选择 在采样序列发生器所执行序列的第8个采样过程中，此字段使用。它指定该模拟输入被用于模拟 - 数字转换采样。
11	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
10:8	<i>MUX2</i>	R/W	0x0	第3个采样输入选择 在采样序列发生器所执行序列的第8个采样过程中，此字段使用。它指定该模拟输入被用于模拟 - 数字转换采样。
7	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
6:4	<i>MUX1</i>	R/W	0x0	第2个采样输入选择 在采样序列发生器所执行序列的第8个采样过程中，此字段使用。它指定该模拟输入被用于模拟 - 数字转换采样。

域	名称	读/写	默认值	描述
3	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
2:0	<i>MUX0</i>	R/W	0x0	第1个采样输入选择 在采样序列发生器所执行序列的第8个采样过程中，此字段使用。它指定该模拟输入被用于模拟 - 数字转换采样。

4.9.4.20 ADC_SSCTL0 - ADC 采样序列控制 0

该寄存器包含每个样品用于与采样序列执行序列的配置信息。当配置一个样本序列，END 位必须在某点被设定，无论是第一样品，最后样本，或其间的任何样品之后。该寄存器为 32 位宽，并且包含 8 个可能采样的信息。

31 RO	30 R/W	29 R/W	28 RO	27 RO	26 R/W	25 R/W	24 RO	23 RO	22 R/W	21 R/W	20 RO	19 RO	18 R/W	17 R/W	16 RO
	IE7	END7			IE6	END6			IE5	END5			IE4	END4	
15 RO	14 R/W	13 R/W	12 RO	11 RO	10 R/W	9 R/W	8 RO	7 RO	6 R/W	5 R/W	4 RO	3 RO	2 R/W	1 R/W	0 R/W
	IE3	END3			IE2	END2			IE1	END1			IE0	END0	SHOTONCE

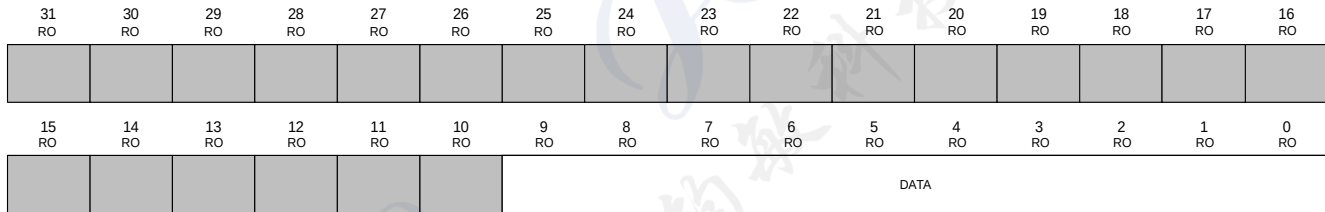
偏移地址: 0x0054

域	名称	读/写	默认值	描述
31	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
30	<i>IE7</i>	R/W	0	第8个采样中断使能 0：原始中断不被认定到中断控制器。 1：原始中断信号在第八样品转换结束时生效。
29	<i>END7</i>	R/W	0	8个采样序列结束 0：在序列中的另一个样本是最终样品。 1：第八样品是序列的最后一个样本。
28:27	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
26	<i>IE6</i>	R/W	0	第7个采样中断使能 0：原始中断不被认定到中断控制器。 1：原始中断信号在第七样品转换结束时生效。
25	<i>END6</i>	R/W	0	7个采样序列结束 0：在序列中的另一个样本是最终样品。 1：第七样品是序列的最后一个样本。
24:23	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
22	<i>IE5</i>	R/W	0	第6个采样中断使能 0：原始中断不被认定到中断控制器。 1：原始中断信号在第六样品转换结束时生效。
21	<i>END5</i>	R/W	0	6个采样序列结束 0：在序列中的另一个样本是最终样品。 1：第六样品是序列的最后一个样本。
20:19	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。

域	名称	读/写	默认值	描述
18	IE4	R/W	0	第5个采样中断使能 0：原始中断不被认定到中断控制器。 1：原始中断信号在第五样品转换结束时生效。
17	END4	R/W	0	5个采样序列结束 0：在序列中的另一个样本是最终样品。 1：第五样品是序列的最后一个样本。
16:15	reserved	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
14	IE3	R/W	0	第4个采样中断使能 0：原始中断不被认定到中断控制器。 1：原始中断信号在第四样品转换结束时生效。
13	END3	R/W	0	4个采样序列结束 0：在序列中的另一个样本是最终样品。 1：第四样品是序列的最后一个样本。
12:11	reserved	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
10	IE2	R/W	0	第3个采样中断使能 0：原始中断不被认定到中断控制器。 1：原始中断信号在第三样品转换结束时生效。
9	END2	R/W	0	3个采样序列结束 0：在序列中的另一个样本是最终样品。 1：第三样品是序列的最后一个样本。
8:7	reserved	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
6	IE1	R/W	0	第2个采样中断使能 0：原始中断不被认定到中断控制器。 1：原始中断信号在第二样品转换结束时生效。
5	END1	R/W	0	2个采样序列结束 0：在序列中的另一个样本是最终样品。 1：第二样品是序列的最后一个样本。
4:3	reserved	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
2	IE0	R/W	0	第1个采样中断使能 0：原始中断不被认定到中断控制器。 1：原始中断信号在第一样品转换结束时生效。
1	END0	R/W	0	1个采样序列结束 0：在序列中的另一个样本是最终样品。 1：第一样品是序列的最后一个样本。
0	SHOTONCE	R/W	0	单次模式 1：每个触发器将只品尝一次，在FIFO使用一个镜头模式。

4.9.4.21 ADC_SSFIFO0 -ADC 采样序列结果 FIFO 0/1/2/3

该寄存器包含与采样序列采集的样品转换结果（ADC_SSFIFO0 寄存器用于采样序列 0，序列发生器 1 ADC_SSFIFO1，ADC_SSFIFO2 序列发生器 2，和 ADC_SSFIFO3 序列发生器 3）。读取顺序样本 0 在该寄存器返回转换结果数据，样品 1，等等，直到在 FIFO 是空的。如果 FIFO 未正确通过软件，上溢和下溢情况的处理被登记在 ADC_OSTAT 和 ADC_USTAT 寄存器。



偏移地址:

ADC_SSFIFO0 : 0x0058

ADC_SSFIFO1 : 0x0078

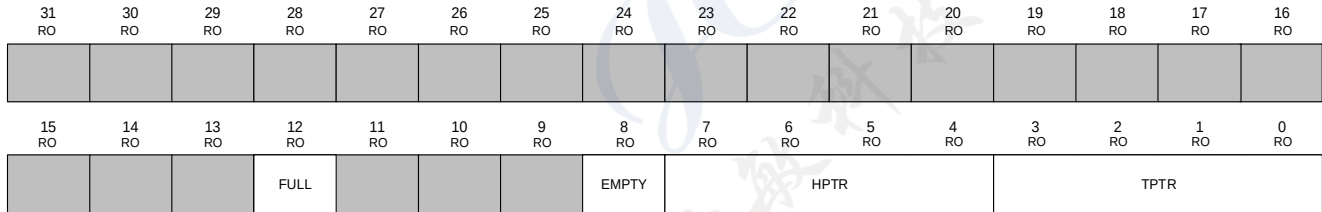
ADC_SSFIFO2 : 0x0098

ADC_SSFIFO3 : 0x00B8

域	名称	读/写	默认值	描述
31:10	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
11:0	<i>DATA</i>	RO	0x0	转换结果

4.9.4.22 ADC_SSFSTATn - 发送/接收逻辑

该寄存器提供了一个窗口，进入采样序列，提供满/空状态信息以及头和尾指针的位置。的复位值 0x100 表示 FIFO 为空。该 **ADC_SSFSTAT0** 寄存器提供 FIFO0，**ADC_SSFSTAT1** 上 FIFO1，在 FIFO2 **ADC_SSFSTAT2** 和 **ADC_SSFSTAT3** 在 FIFO3 状态。



偏移地址:

ADC_SSFIFO0 : 0x005C

ADC_SSFIFO1 : 0x007C

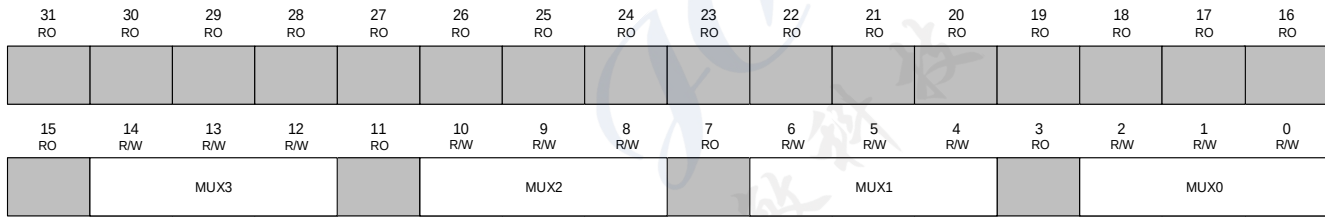
ADC_SSFIFO2 : 0x009C

ADC_SSFIFO3 : 0x00BC

域	名称	读/写	默认值	描述
31:13	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
12	<i>FULL</i>	RO	0	FIFO满 0 : The FIFO目前不充分。 1 : FIFO当前已满。
11:9	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
8	<i>EMPTY</i>	RO	1	FIFO空 0 : FIFO不空的。 1 : FIFO当前为空。
7:4	<i>HPTR</i>	RO	0x0	FIFO头指针 该字段包含当前的“头”指针指数，这是将要写入的下一个条目，用于FIFO。 0x0 – 0x7: FIFO0 0x0 – 0x3: FIFO 1 and FIFO 2 0x0: FIFO 3
3:0	<i>TPTR</i>	RO	0x0	FIFO尾指针 该字段包含当前的“尾巴”指针指数，这是要读取的下一个条目，用于FIFO。 0x0 – 0x7: FIFO 0 0x0 – 0x3: FIFO 1 and FIFO 2 0x0: FIFO 3

4.9.4.23 ADC_SSMUX1/ADC_SSMUX2 - ADC 采样序列输入多路复用选择 1/2

该寄存器定义了采样序列执行序列的每个采样的模拟输入配置 1 或 2。这些寄存器为 16 位宽，并包含 4 种可能采样的信息。该 ADC_SSMUX1 寄存器影响采样序列发生器 1 和 ADC_SSMUX2 寄存器影响采样序列发生器 2。



偏移地址:

ADC_SSMUX1 : 0x0070

ADC_SSMUX2 : 0x0090

域	名称	读/写	默认值	描述
31:15	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
14:12	<i>MUX3</i>	R/W	0x0	第4个采样输入选择 在采样序列发生器所执行序列的4个采样中此字段使用。它指定该模拟输入被用于模拟 - 数字转换采样。
11	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
10:8	<i>MUX2</i>	R/W	0x0	第3个采样输入选择 在采样序列发生器所执行序列的4个采样中此字段使用。它指定该模拟输入被用于模拟 - 数字转换采样。
7	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
6:4	<i>MUX1</i>	R/W	0x0	第2个采样输入选择 在采样序列发生器所执行序列的4个采样中此字段使用。它指定该模拟输入被用于模拟 - 数字转换采样。
3	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
2:0	<i>MUX0</i>	R/W	0x0	第1个采样输入选择 在采样序列发生器所执行序列的4个采样中此字段使用。它指定该模拟输入被用于模拟 - 数字转换采样。

4.9.4.24 ADC_SSCTL1 /ADC_SSCTL2 - ADC 采样序列控制 1/2

这些寄存器包含每个样品用于与采样序列 1 或 2 执行在配置采样序列，在 END 位必须在某些点被设定，无论是第一样品，最后一个样本，或者任何样品后的序列的配置信息在两者之间。这些寄存器为 16 位宽，并包含 4 种可能采样的信息。该 BCSSCTL1 寄存器配置采样序列发生器 1 和 ADC_SSCTL2 寄存器配置采样序列发生器 2。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 R/W	13 R/W	12 RO	11 RO	10 R/W	9 R/W	8 RO	7 RO	6 R/W	5 R/W	4 RO	3 RO	2 R/W	1 R/W	0 R/W
	IE3	END3			IE2	END2			IE1	END1			IE0	END0	SHOTONCE

偏移地址:

ADC_SSCTL1 : 0x0074

ADC_SSCTL2 : 0x0094

域	名称	读/写	默认值	描述
31:15	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
14	<i>IE3</i>	R/W	0	第4个采样中断使能 0：原始中断不被认定到中断控制器。 1：原始中断信号在第四样品转换结束时生效。
13	<i>END3</i>	R/W	0	第四个采样序列结束 0：在序列中的另一个样本是最终样品。 1：第四个样品是在序列的最后一个样本。
12:11	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
10	<i>IE2</i>	R/W	0	第3个采样中断使能 0：原始中断不被认定到中断控制器。 1：原始中断信号在第三样品转换结束时生效。
9	<i>END2</i>	R/W	0	第三个采样序列结束 0：在序列中的另一个样本是最终样品。 1：第三个样品是在序列的最后一个样本。
8:7	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
6	<i>IE1</i>	R/W	0	第2个采样中断使能 0：原始中断不被认定到中断控制器。 1：原始中断信号在第二样品转换结束时生效。
5	<i>END1</i>	R/W	0	第二个采样序列结束 0：在序列中的另一个样本是最终样品。 1：第二个样品是在序列的最后一个样本。
4:3	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
2	<i>IE0</i>	R/W	0	第1个采样中断使能 0：原始中断不被认定到中断控制器。 1：原始中断信号在第一样品转换结束时生效。

域	名称	读/写	默认值	描述
1	END0	R/W	0	第一个采样序列结束 0：在序列中的另一个样本是最终样品。 1：第一个样品是在序列的最后一个样本。
0	SHOTONCE	R/W	0	单次模式 1：每个触发器将只品尝一次，在FIFO使用一个镜头模式。

4.9.4.25 ADC_SSMUX3 - ADC 采样序列输入多路复用器选择 3

该寄存器为采样序列发生器 3. 执行样本模拟输入配置。该寄存器为 3 位宽，并包含一个可能的样品信息。见详细位说明了 ADC_SSMUX0 寄存器。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 R/W	1 R/W	0 R/W
															MUX0

偏移地址: 0x00B0

域	名称	读/写	默认值	描述
31:3	reserved	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
2:0	MUX0	R/W	0x0	第一个样品输入选择 与样品定序执行序列的第一样品中该域被使用。它指定该模拟输入被用于模拟 - 数字转换采样。

4.9.4.26 ADC_SSCTL3 - ADC 采样序列控制 3

该寄存器配置采样序列 3 取样中断引导序列和结束点时，由于采样序列 3 是只有一个采样序列中，所以结束点 **END0** 默认设置为 1。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 R/W	1 R/W	0 RO
													IE0	END0	

偏移地址: 0x00B4

域	名称	读/写	默认值	描述
31:3	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。
2	<i>IE0</i>	R/W	0	第一个样品中断使能 0：原始中断不被认定到中断控制器。 1：原始中断信号在第一采样转换结束时生效。
1	<i>END0</i>	R/W	1	第一个样品是序列结束 0：在序列中的另一个样本是最终样品。 1：第一样品是该序列的最后一个样本。
0	<i>reserved</i>	RO	0x0	软件不应该依赖保留位的值。考虑与其他产品的兼容性，在此值不应该被写入或读出。

4.10 通用定时器 (GPTM)

PT32U301 包含五個个 16 位/32 位通用定时器模块(GPTM0, GPTM1, GPTM2, GPTM 3, GPTM4)。每个模块提供 16 位/32 位定时器/计数器(简称 Timer-A 和 Timer-B)，它们能够被配置为定时器或事件计数器独立地进行操作，或将它们连接成一个 32 位定时器。

此外，定时器可以用来触发模拟-数字 (ADC)/数字-模拟 (DAC)转换。来自所有通用定时器的触发信号在到达 ADC/ DAC 模块之前先进行或逻辑操作，所以同一时间只能有一个定时器用来触发 ADC/ DAC 事件。

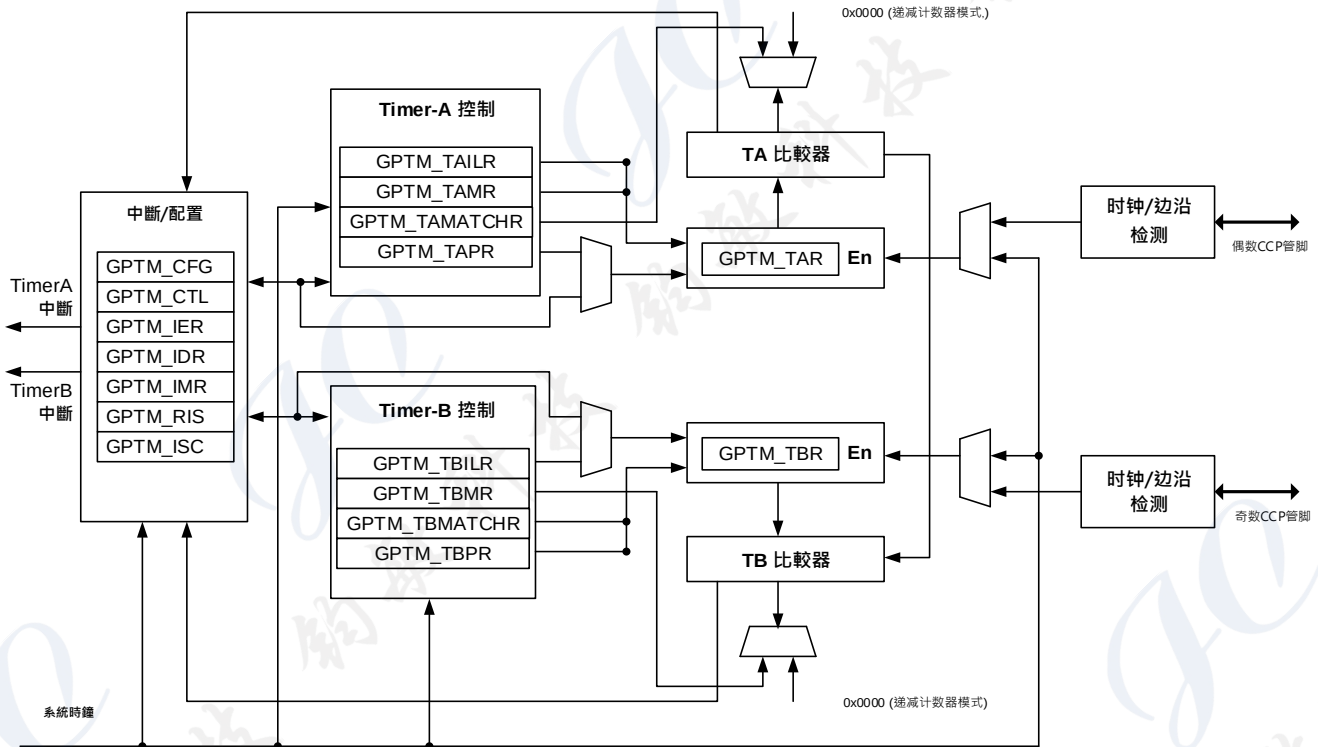
此 GPTM 在 PT32U301 上是一个可用的定时资源。其他定时器资源包括系统定时器 (SysTick)。

通用定时器模块 (GPTM) 提供以下功能选项：

- 16/32 位定时器模式
 - 带有 16 位分频器的 16 位通用定时器功能。
 - 带有 16 位分频器的 16 位输入边沿计数捕获或输入边沿定时捕获。
 - 16 位 PWM 输入边沿偵測
 - 带有 16 位分频器的 16 位 PWM 模式和软件可编程输出 PWM 信号翻转。
 - 16 位或 32 位可编程单次触发定时器。
 - 16 位或 32 位可编程周期定时器。
- ADC 事件触发。
- 触发 DAC 的同步电路
- 在调试期间当控制器产生 CPU 停止标志时，用户可使能停止计数器。

4.10.1 块图

图 4.10-1: GPTM 模块框图



4.10.2 功能描述

每个 GPTM 模块的主要部分是两个独立运行的 16 位递增/递减的计数器（简称 Timer A 和 Timer B），两个 16 位匹配寄存器，两个分频器匹配寄存器，两个 16 位装载/初始化寄存器以及和它们相关的控制功能。每个 GPTM 的具体功能可通过软件控制和通过寄存器接口配置。

4.10.2.1 GPTM 复位条件

GPTM 复位后处于激活状态，所有控制寄存器均被清零，同时进入默认状态。Timer A 和 Timer B 的装载寄存器初始化为 0xFFFF_FFFF。预分频计数器初始化为 0xFFFF。

4.10.2.2 GPTM 操作模式

各个 GPTM 模块的可用模式在下表顯示。

表 4.10-1: 可用的操作模式

模式	定时器使用	计数器大小
单次触发	独立	16-位
	连接	32-位
周期	独立	16-位
	连接	32-位
边沿计数	独立	16-位
边沿时间	独立	16-位
PWM 边沿	独立	16-位
PWM	独立	16-位

注意:

仅可以在单独使用定时器时使用预分频器
所有計時器向下數

此部分描述了各种定时器模式的运行。在连接模式中使用 Timer A 和 Timer B 时，仅必须使用 Timer A 控制和状态位；不需要使用 Timer B 控制和状态位。通过設定 GPTM 配置寄存器(GPTM_CFG)的 CFG 位可以 GPTM 配置为独立模式。在以下部分中，变量“n”用于位域和寄存器名称中，表示 Timer A 函数或 Timer B 函数

单次触发/周期定时器模式

选择单次触发模式还是周期模式由写入 GPTM Timer n 模式 (GPTM_TnMR) 寄存器中 MODE 域的值来决定。

在 32 位配置時，可选的 16 位预分频器把 32 位向下计数器的计数范围扩展到了 48 位，在 16 位配置時，预分频器将计数范围扩展到 32 位。

当软件置位了 GPTM 控制寄存器(GPTM_CTL) 寄存器中的 TnEN 位，定时器从预装载的值开始进行递减计数。一旦递减计数到 0x0 状态时，定时器将会在下一个时钟周期从 GPTM Timer n 间隔加载寄存器 (GPTM_TnILR) 寄存器重新加载其初值。。如果配置为单次触发定时器，定时器将停止计数，并且将 GPTM_CTL 寄存器中的 TnEN 位清零。如果配置为周期定时器，则定时器会在下一个时钟周期开始再次计数。

除了重装计数值外，GPTM 在到达 0x0 状态时将会产生中断并输出触发信号。GPTM 将它的原始中断状态(GPTM_RIS) 寄存器中的 TnTORI 位置位，并且保持直到通过对 GPTM 中断状态和清除(GPTM_ISC)寄存器进行写操作将其清零。如果 GPTM 中断屏蔽(GPTM_IMR)寄存器中的超时中断使能，则 GPTM 也会置位 GPTM 中断状态和清除(GPTM_ISC) 寄存器中的 TnTOISC 位。

如果计数器运行过程中软件重装 GPTM_TnILR 或 GPTM_TnPR 寄存器，则在下一个时钟周期装载新值并从新值开始继续计数。如果设置 GPTM_CTL 寄存器中的 TnSTALL 位，则在调试模式下被处理器停止时，定时器停止计数。直到处理器恢复执行定时器才恢复计数。

下面的表格显示了在使用预分频器时 16 或 32 位自由运行的定时器的各种配置。所有的值都是以时钟频率 16-MHz（时钟周期 Tc=62.5 ns）作为标准进行计算。

表 4.10-2: 带预分频器的 16 位定时器配置

Prescale	Clock	Max Time	Units
00000000	1	4.096	ms
00000001	2	8.192	ms
00000010	3	12.288	ms
--	--	--	--
11111101	254	1040.384	ms
11111110	255	1044.48	ms
11111111	256	1048.576	ms

表 4.10-3: 带预分频器的 32 位定时器配置

Prescale	Clock	Max Time	Units
00000000	1	268.435	s
00000001	2	536.871	s
00000010	3	805.306	s
--	--	--	--
11111101	254	0.682	10 ⁵ s
11111110	255	0.685	10 ⁵ s
11111111	256	0.687	10 ⁵ s

输入边沿计数模式

注意：在上升沿检测中，上升沿到来后必须保持至少两个系统时钟周期的高电平。同样在下降沿检测中，下降沿到来之后必须保持至少两个系统时钟周期的低电平。因此在边沿检测中输入信号的最大频率为 1/4 个系统时钟。

注意：在此模式下与分频器无效。

在边沿计数模式中，计数器被配置成能够捕获 3 种事件类型的递减计数器：上升沿，下降沿，或上升下降沿。把 **GPTM_TnMR.MODE** 位设为 0x3 而 **GPTM_TnMR.CM** 須为零进入边沿计数器模式。计数器计数所采用的边沿类型由 **GPTM_CTL** 中的 **TnEVENT** 域来决定。

在初始化过程中，需要对 **GPTM** 定时器匹配(**GPTM_TnMATCHR**)寄存器进行配置，使得在 **GPTM_TnILR** 寄存器和 **GPTM_TnMATCHR** 寄存器中的差值等于边沿事件的计数值。

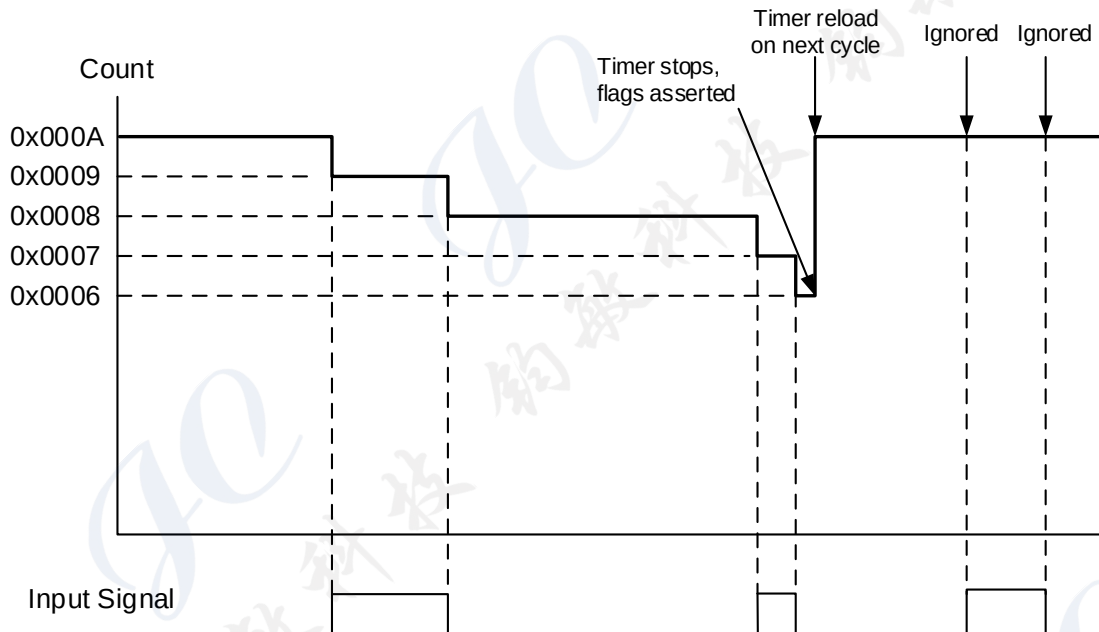
当软件对 **GPTM** 控制(**GPTM_CTL**)寄存器中的 **TnEN** 位执行写操作时，定时器被使能捕获事件功能。在 **CCP** 输入引脚上每输入一个事件，计数器的值就减 1 直到事件计数器的值匹配 **GPTM_TnMATCHR** 寄存器的值，此时 **GPTM** 的 **GPTM_RIS** 寄存器中的 **CnMRI** 位会被置 1（如果中断没被屏蔽，**CnMISC** 位也会被置 1）。

然后计数器使用 **GPTM_TnILR** 中的值进行重载，**GPTM** 自动地清除 **GPTM_CTL** 寄存器中的 **TnEN** 位，因此计数器停止。一旦事件计数器计满，接下来的所有事件被忽略，直到通过软件重新设置 **TnEN** 位。

下图显示了边沿计数模式的工作过程，在这种情况下，定时器起始值通过 **GPTM_TnILR = 0x000A** 被设置，同时匹配值通过 **GPTM_TnMATCHR = 0x0006** 被设置，因此 3 个边沿事件被计数。该计数器配置为检测输入信号的上升沿和下降沿。

注意：由于在当前计数值与 **GPTM_TnMATCHR** 寄存器中的值相匹配后，定时器自动清除 **TnEN** 位，所以最后的两个边沿被忽略。

图 4.10-2: 16 位输入边沿计数模式范例



输入边沿定时模式

注意：在上升沿检测中，上升沿来到后输入信号必须保持至少两个系统时钟周期的高电平，同样在下降沿检测中，下降沿到来后它也必须保持至少两个系统时钟周期的低电平。基于这个标准，在边沿检测中输入信号的最大频率为 $1/4$ 个系统频率。

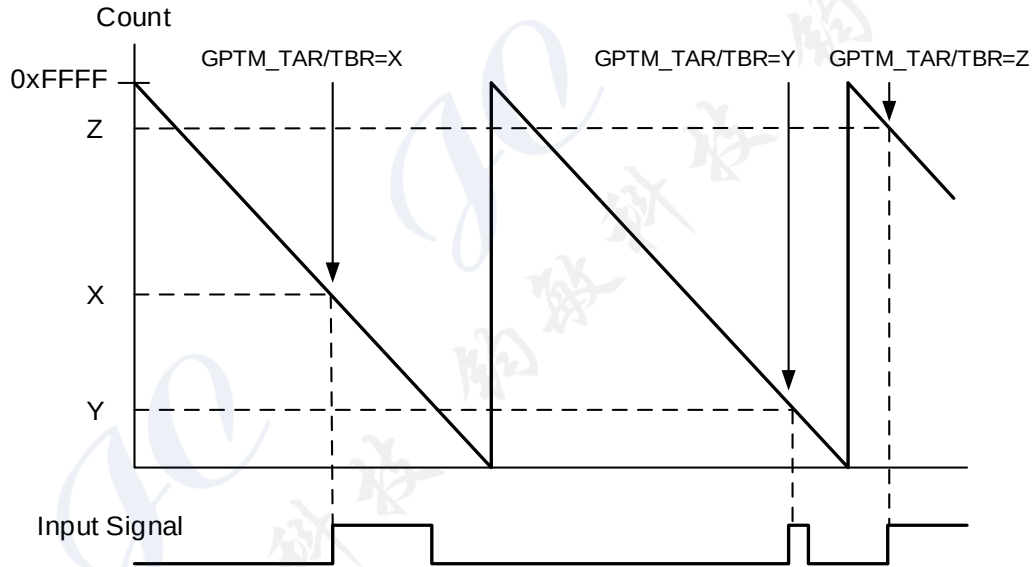
注意：在此模式下预分频器无效。

在边沿定时模式下，定时器被配置为独立的递减计数，初值装载到 **GPTM_TnILR 寄存器**。该定时器可以捕获 3 种类型的事件：上升沿，下降沿，或上升下降沿。把 **GPTM_TnMR.MODE** 位设置进入边沿定时模式。定时器捕获事件的边沿类型由 **GPTM_CTL** 寄存器中的 **TnEVENT** 位域来决定。

当 **GPTM_CTL.TnEN** 被使能时，定时器使能事件捕获功能。当检测到被选择的事件时，当前计数器的值在 **GPTM_TnR 寄存器** 中被捕获，并且该值可以通过处理器读取。此时 **CnERI** 位被置 1（如果中断没有被屏蔽，**CnEISC** 位也会被置 1）。当捕获到一个事件后，定时器不会停止运行。它会持续计数直到 **TnEN** bit 位被清除。当定时器递减到 0x0000 时，初值被重装载到 **GPTM_TnILR** 寄存器中。

下图显示了边沿定时模式的工作过程。在下图中假定定时器的初值为 0xFFFF，且定时器配置为上升沿事件捕获，每一个上升沿事件到来，定时器的当前计数值都被锁存到 **GPTM_TnR 寄存器** 中，并保持该值直到下一个上升沿事件到来（关键是新的计数值被装载到 **GPTM_TnR**）。

图 4.10-3: 16 位输入边沿定时模式范例



输入 PWM 边沿检测模式

注意：在上升沿检测中，上升沿来到后输入信号必须保持至少两个系统时钟周期的高电平，同样在下降沿检测中，下降沿到来后它也必须保持至少两个系统时钟周期的低电平。基于这个标准，在边沿检测中输入信号的最大频率为 $1/4$ 个系统频率。

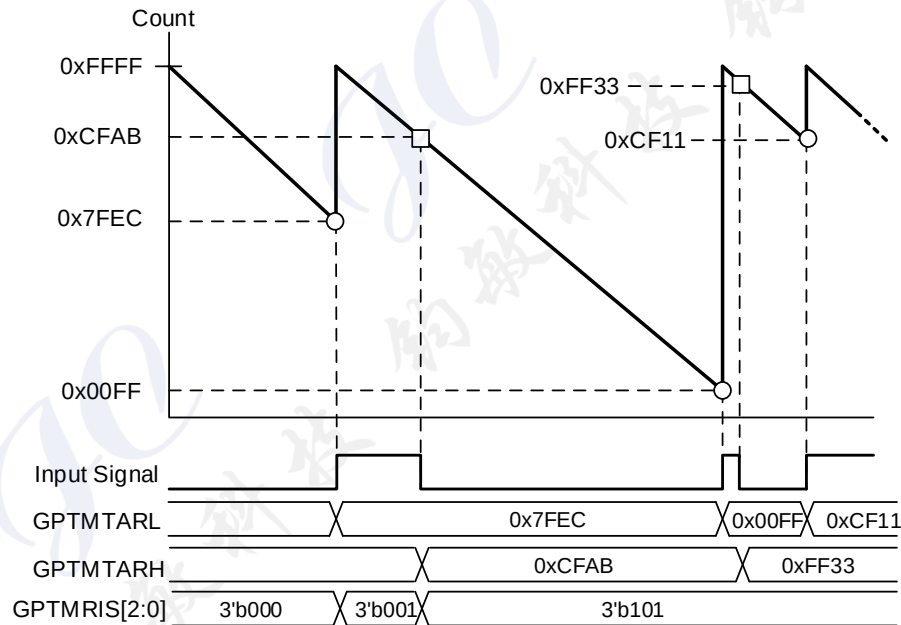
注意：在此模式下预分频器无效。

在 PWM 边沿检测模式中，定时器配置为一个独立的递减计数器，初值由 **GPTM_TnILR** 寄存器定义。定时器捕获模式必须设置为上升和下降沿模式。通过设置 **GPTM_TnMR.MODE** 位可以使定时器进入 PWM 边沿检测模式，而 **GPTM_CTL.TnEVENT** 必需设置为 0x3。

当软件对 **GPTM_CTL** 寄存器中的 **TnEN** 位写时，定时器使能事件捕获功能。当检测到上升沿时，当前计数值被锁存到 **GPTM_TnR** 寄存器的低 15 位，该值可以通过处理器进行读取。当下降沿到来时，当前计数值被锁存到 **GPTM_TnR** 寄存器的高 15 位，该值可以通过处理器进行读取。然后 **CnERI** 位会被置 1（如果中断没有被屏蔽，**CnEISC** 位也会被置 1）。如果检测到上升沿，**TnTORI** 位将会被置 1。一个上升沿到来或定时器递减到 0 后，**GPTM_TnILR** 寄存器的值会被重载。当下降沿到来时，定时器不会停止计数 直到检测到上升沿或 **TnEN** 位被清除。

下图显示输入 PWM 边沿检测模式的工作原理。在图中假设定时器的初始值默认为 0xFFFF，同时定时器配置为上升沿和下降沿事件捕获。

图 4.10-4: 16 位输入边沿检测模式范例



PWM 模式

注意：在此模式下预分频器无效。GPTM 支持一个简单地 PWM 产生模式。

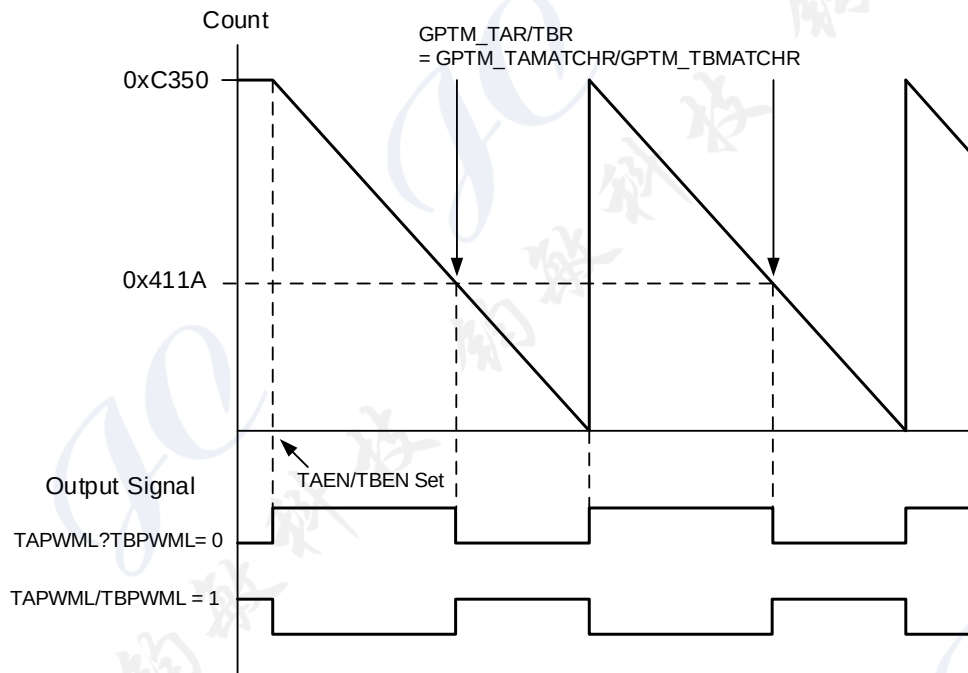
在 PWM 模式中，定时器被配置为一个递减计数器，它的初值可以通过 **GPTM_TnILR** 来确定。在这种模式下，PWM 的频率和周期是同步事件，因此可以保证无毛刺。通过设置 **GPTM_TnMR** 寄存器中的 **AMS** 位为 0x1，**CM** 位为 0x0，**MODE** 域为 0x0 或 0x2，可以使能 PWM 模式。

当软件对 **GPTM_CTL** 寄存器中的 **TnEN** 位写 1 时，该计数器开始递减计数直到它的值递减到 0x0000。在下一个计数周期，该计数器从 **GPTM_TnILR** 中重新装载它的初值，并且持续计数直到软件清除 **GPTM_CTL** 寄存器中的 **TnEN** 位。在 PWM 模式下，没有中断和状态位会被置位。

计数器的值等于 **GPTM_TnILR** 寄存器的值时，PWM 输出 1，计数器的值等于 GPTM 匹配寄存器(**GPTM_TnMATCHR**)的值时，它输出 0。软件通过 **GPTM_CTL** 寄存器中的 **TnPWML** 位可以实现 PWM 信号的反向输出。

下图显示怎样产生一个周期为 1ms，占空比为 66% 的 PWM 波形，假设输入时钟为 50MHz，同时 GPTM 匹配 **TnPWML=0**（对 **TnPWML=1** 的配置，占空比为 33%）。针对该实例，初始值为 **GPTM_TnILR=0xC350**，匹配值为 **GPTM_TnMATCHR=0x411A**。

图 4.10-5: 16 位 PWM 检测模式范例



中断控制

在捕获事件，捕获匹配和定时器超时中产生中断。通过 5 个寄存器来控制定时器的中断产生。

- 中断控制 (IER, IDR, IMR)

GPTM 中断使能寄存器(**GPTM_IER**)通过简单地写“1”来使能中断请求线。类似地，GPTM 中断禁止寄存器(**GPTM_IDR**)通过写“1”禁止中断请求线。IER 和 IDR 是控制中断屏蔽的只写寄存器。这两个寄存器的总体结果可以通过 GPTM 中断屏蔽寄存器(**GPTM_IMR**)来显示。IMR 是一个只读寄存器，使用“1”或“0”表示是否中断请求线被使能还是被禁止。

- 中断状态读(RIS)

GPTM 原始中断状态(**GPTM_RIS**)是一个只读寄存器，可以从该模块中读取中断状态。

- 中断清除(ISC)

GPTM 中断状态&中断清除寄存器(**GPTM_ISC**)是用来指示该模块中未被屏蔽的中断，因为只有未被屏蔽的中断才能向处理器发出中断请求。对该寄存器中的某一位写入“1”可以清除相对应的中断状态或通过写“1”到 IDR 禁止该中断。

4.10.3 GPTM 初始化和配置

本节介绍 GPTM 各模式的配置和初始化，并举例说明。

4.10.3.1 32 位单次触发/周期定时模式

GPTM 按照下面的顺序配置 32 位单次触发和周期定时模式：

1. 在配置定时器之前确保定时器已经被停止（**GPTM_CTL 寄存器**中的 *TnEN* 位被清除）。
2. 把 GPTM 配置寄存器(**GPTM_CFG**)清零。
3. 设置 GPTM Timer-A/B 模式寄存器(**GPTM_TnMR**)中的 *MODE* 位域：
 - a. 写入 0x1 进入单次触发模式。
 - b. 写入 0x2 进入周期模式。
4. 如果需要使用预分频器，分频系数需要写入 GPTM 定时器预分频寄存器(**GPTM_TnPR**)。
5. 将定时器的初值装载到 GPTM Timer-A/B 间隔装载寄存器(**GPTM_TnILR**)。
6. 如果需要产生中断，需要设置 GPTM 中断使能寄存器(**GPTM_IER**)中的 *TnTOIE* 位。
7. 把 **GPTM_CTL 寄存器**中的 *TnEN* 位置 1 使能定时器并开始计数。
8. 轮询 **GPTM_RIS 寄存器**中的 *TnTORIS* 位或等待中断请求产生（如果中断被使能）。向 GPTM 中断状态和清除寄存器(**GPTM_ISC**)中的 *TnTOISC* 位写 1 可以清除状态标志。

在单次触发模式下，在步骤 8 后定时器停止计数。如果要重新启动定时器，重复上面的步骤。在周期模式下即使定时器溢出后，它也不会停止计数。

4.10.3.2 16 位单次触发/周期定时模式

GPTM 按照下面的顺序配置 16 位单次触发和周期定时模式：

1. 在配置定时器之前确保定时器已经被停止（*TnEN* 位被清除）。
2. 设置 GPTM 配置寄存器(**GPTM_CFG**)：
 - a. 写入 0x4 进入单次触发模式。
 - b. 写入 0x5 进入周期模式。
3. 设置 GPTM 定时器模式寄存器(**GPTM_TnMR**)中的 *MODE* 位域：
 - a. 写入 0x1 进入单次触发模式。
 - b. 写入 0x2 进入周期模式。
4. 如果需要使用预分频器，分频系数需要写入 GPTM 定时器预分频寄存器(**GPTM_TnPR**)。
5. 将定时器的初值装载到 GPTM 定时器间隔装载寄存器(**GPTM_TnILR**)。
6. 如果需要产生中断，需要设置 GPTM 中断使能寄存器(**GPTM_IER**)中的 *TnTOIE* 位。
7. 把 **GPTM_CTL 寄存器**中的 *TnEN* 位置 1 使能定时器并开始计数。
8. 轮询 **GPTM_RIS 寄存器**中的 *TnTORI* 位或等待中断请求产生（如果中断被使能）。向 GPTM 中断状态和清除寄存器(**GPTM_ISC**)中的 *TnTOISC* 位写 1 可以清除状态标志。

在单次触发模式下，在步骤 8 后定时器停止计数。如果要重新启动定时器，重复上面的步骤。在周期模式下即使定时器溢出后，它也不会停止计数。

4.10.3.3 16 位输入边沿计数模式

GPTM 按照下面的顺序配置 16 位输入边沿计数模式：

1. 在配置定时器之前确保定时器已经被停止（*TnEN* 位被清除）。
2. 向 GPTM 配置(GPTM_CFG)寄存器写入 0x6。
3. 向 GPTM 定时器模式(GPTM_TnMR)寄存器中的 *CM* 域写入 0x0，*MODE* 域写入 0x3。
4. 通过对 GPTM 控制(GPTM_CTL)寄存器中的 *TnEVENT* 域进行写操作可以配置定时器捕获事件的类型。
 - a. *TnEVENT*=0x0，上升沿捕获。
 - b. *TnEVENT*=0x1，下降沿捕获。
 - c. *TnEVENT*=0x3，上升沿/下降沿捕获。
5. 将定时器的初值装载到 GPTM 定时器间隔装载(GPTM_TnILR)寄存器。
6. 装载希望的事件计数值到 GPTM 定时器匹配(GPTM_TnMATCHR)寄存器。
7. 如果需要去抖动，应该设置 GPTM 定时器去抖动(GPTM_DBC)寄存器。
8. 如果需要产生中断请求，设置 GPTM 中断使能(GPTM_IER)寄存器中的 *CnMIE* 位。
9. 置位 GPTM_CTL 寄存器中的 *TnEN* 位使能定时器和开始等待边沿事件。
10. 轮询 GPTM_RIS 寄存器中的 *CnMRI* 位或等待中断请求产生（如果中断使能）。向 GPTM 中断状态和清除寄存器(GPTM_ISC)中的 *CnMISC* 位写 1 可以清除状态标志。

在输入边沿计数模式下，在边沿事件总数到达设置的总事件数后，定时器停止计数。如果想要重启该定时器，需要确保 *TnEN* 位被清除，并且重复步骤 4 到 10。

4.10.3.4 16 位输入边沿定时模式

GPTM 按照下面的顺序配置 16 位输入边沿计数模式：

1. 在配置定时器之前确保定时器已经被停止（*TnEN* 位被清除）。
2. 向 GPTM 配置(GPTM_CFG)寄存器写入 0x6。
3. 向 GPTM 定时器模式(GPTM_TnMR)寄存器中的 *CM* 域写入 0x1，*MODE* 域写入 0x3。
4. 通过对 GPTM 控制(GPTM_CTL)寄存器中的 *TnEVENT* 域进行写操作可以配置定时器捕获事件的类型。
 - a. *TnEVENT*=0x0，上升沿捕获。
 - b. *TnEVENT*=0x1，下降沿捕获。
 - c. *TnEVENT*=0x3，上升沿/下降沿捕获。
5. 将定时器的初值装载到 GPTM 定时器间隔装载(GPTM_TnILR)寄存器。
6. 如果需要去抖动，应该设置 GPTM 定时器去抖动(GPTM_DBC)寄存器。
7. 如果需要产生中断请求，设置 GPTM 中断使能(GPTM_IER)寄存器中的 *CnEIE* 位。
8. 置位 GPTM_CTL 寄存器中的 *TnEN* 位使能定时器和开始计时。
9. 轮询 GPTM_RIS 寄存器中的 *CnERI* 位或等待中断请求产生（如果中断使能）。向 GPTM 中断状态和清除寄存器(GPTM_ISC)中的 *CnEISC* 位写 1 可以清除状态标志。

在输入边沿定时模式下，定时器在检测到边沿事件之后继续运行，但通过写 GPTM_TnILR 寄存器可在任何时候改变定时器间隔。此改变在写操作的下一个周期生效。

4.10.3.5 16 位输入 PWM 边沿检测模式

GPTM 按照下面的顺序配置 16 位输入 PWM 边沿检测模式：

1. 在配置定时器之前确保定时器已经被停止（*TnEN* 位被清除）。
2. 向 GPTM 配置(GPTM_CFG)寄存器写入 0x6。
3. 向 GPTM 定时器模式(GPTM_TnMR)寄存器中的 *CM* 域写入 0x1，*MODE* 域写入 0x3。
4. 通过对 GPTM 控制(GPTM_CTL)寄存器中的 *TnEVENT* 域写 0x3，它表示选择上升沿/下降沿捕获。
5. 将定时器的初值装载到 GPTM 定时器间隔装载(GPTM_TnILR)寄存器。
6. 如果需要去抖动，应该设置 GPTM 定时器去抖动(GPTM_DBC)寄存器。
7. 如果需要产生中断请求，设置 GPTM 中断使能(GPTM_IER)寄存器中的 *CnEIE* 位和 *TnTOIE* 位。
8. 置位 GPTM_CTL 寄存器中的 *TnEN* 位使能定时器和开始等待边沿事件。
9. 轮询 GPTM_RIS 寄存器中的 *CnERI* 位或等待中断请求产生（如果中断使能）。向 GPTM 中断状态和清除寄存器 (GPTM_ISC)中的 *CnEISC* 位写 1 可以清除状态标志。该事件发生的时间可以通过读 GPTM 定时器模式 (GPTM_TnR)寄存器来获得。当 *TnTOIE* 位被置 1 时，它意味着 GPTM_TnR 的低 15 位是一个捕获值用来计算 PWM 占空比(GPTM_TnILR - GPTM_TnR 低 15 位)；当 *CnEIE* 位被置 1 时，它意味着 GPTM_TnR 的高 15 位是一个捕获值用来计算 PWM 正脉宽(GPTM_TnILR - GPTM_TnR 高 15 位)。16 位 PWM 模式

4.10.3.6 16 位 PWM 模式

GPTM 按照下面的顺序配置 16 位 PWM 模式：

1. 在配置定时器之前确保定时器已经被停止（*TnEN* 位被清除）。
2. 向 GPTM 配置(GPTM_CFG)寄存器写入 0x7。
3. 向 GPTM 定时器模式(GPTM_TnMR)寄存器中的 *AMS* 位写入 0x1，*CM* 域写入 0x0，*MODE* 域的配置如下：
 - a. *MODE*=0x0，三角 PWM 模式。
 - b. *MODE*=0x2，简单 PWM 模式。
4. 通过配置 GPTM 控制(GPTM_CTL)寄存器中的 *TnPWML* 域可以选择 PWM 信号的输出状态。
5. 将定时器的初值装载到 GPTM 定时器间隔装载(GPTM_TnILR)寄存器。
6. 装载希望的事件计数值到 GPTM 定时器匹配(GPTM_TnMATCHR)寄存器。
7. 置位 GPTM 控制 (GPTM_CTL) 寄存器中的 *TnEN* 位使能定时器和开始产生 PWM 输出信号。

在 PWM 定时模式下，PWM 信号产生后该定时器任然会继续运行。PWM 周期可以在任何时刻通过对 GPTM_TnILR 和 GPTM_TnMATCHR 寄存器执行写操作来进行调整，对该寄存器写之后将会在下一个时钟周期生效。

4.10.4 通用定时器寄存器映射

GPTM 基地址:

- GPTM0: 0x4000_0000
- GPTM1: 0x4000_0100
- GPTM2: 0x4000_0200

偏移地址	寄存器名称	读/写	默认值	描述	参考页
0x0000	GPTM_CFG	R/W	0x0000_0000	GPTM配置寄存器	199
0x0004	GPTM_TAMR	R/W	0x0000_0000	GPTM Timer-A模式寄存器	200
0x0008	GPTM_TBMR	R/W	0x0000_0000	GPTM Timer-B模式寄存器	201
0x000C	GPTM_CTL	R/W	0x0000_0000	GPTM控制寄存器	202
0x0010	GPTM_IER	WO	0x0000_0000	GPTM中断使能寄存器	203
0x0014	GPTM_IDR	WO	0x0000_0000	GPTM中断禁止寄存器	204
0x0018	GPTM_IMR	RO	0x0000_0000	GPTM中断屏蔽状态寄存器	205
0x001C	GPTM_RIS	RO	0x0000_0000	GPTM原始中断状态寄存器	206
0x0020	GPTM_ISC	R/W1C	0x0000_0000	GPTM中断状态和清除寄存器	207
0x0028	GPTM_TAILR	R/W	0xFFFF_FFFF	GPTM Timer-A间隔时间装载寄存器	208
0x002C	GPTM_TBILR	R/W	0xFFFF_FFFF	GPTM Timer-B间隔时间装载寄存器	208
0x0030	GPTM_TAMATCHR	R/W	0x0000_FFFF	GPTM Timer-A匹配值寄存器	209
0x0034	GPTM_TBMATCHR	R/W	0x0000_FFFF	GPTM Timer-B匹配值寄存器	209
0x0038	GPTM_TAPR	R/W	0x0000_FFFF	GPTM Timer-A预分频寄存器	210
0x003C	GPTM_TBPR	R/W	0x0000_FFFF	GPTM Timer-B预分频寄存器	210
0x0040	GPTM_DBC	R/W	0x0000_0000	GPTM去抖动寄存器	211
0x0048	GPTM_TAR	RO	0xFFFF_FFFF	GPTM Timer-A寄存器	212
0x004C	GPTM_TBR	RO	0xFFFF_FFFF	GPTM Timer-B寄存器	212



4.10.4.1 GPTM_CFG - GPTM 配置寄存器

该寄存器是 GPTM 的全局配置寄存器。写入该寄存器的值用来决定它是 32 位还是 16 位模式。

注意: 该寄存器中的位应仅在 GPTM_CTL 寄存器中的 *TnEN* 位清零时发生更改。

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

偏移地址：0x0000

位	名称	读/写	默认值	描述
31:3	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
2:0	<i>CFG</i>	R/W	0x0	GPTM配置 0x0：选择32位定时器配置。 0x0-0x3：保留。 0x4：选择16位定时器单次触发模式。 0x5：选择16位定时器周期模式。 0x6：选择16位定时器捕获模式。 0x7：选择16位定时器PWM模式。

4.10.4.2 GPTM_TAMR - GPTM Timer-A 模式寄存器

该寄存器的配置基于在 **GPTM_CFG** 寄存器。如果要选择 16 位 PWM 模式，需要设置 **AMS** 位为 0x1，**CM** 位为 0x0，并且 **MODE** 域为 0x0 或 0x2。

注意：该寄存器中的位域仅能够在 **GPTM_CTL** 寄存器中的 **TAEN** 位被清零的前提下才能被改变。

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

偏移地址：0x0004

位	名称	读/写	默认值	描述
31: 4	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
3	<i>AMS</i>	R/W	0	GPTM Timer-A复用模式选择 0：捕获模式使能。 1：PWM模式使能。
2	<i>CM</i>	R/W	0	GPTM Timer-A捕获模式 0：边沿计数模式。 1：边沿定时模式。
1:0	<i>MODE</i>	R/W	0x0	GPTM Timer-A模式 0x0：PWM U/D模式。 0x1：单次触发定时器模式。 0x2：周期定时器模式。 0x3：捕获模式。

4.10.4.3 GPTM_TBMR - GPTM Timer-B 模式寄存器

该寄存器的配置基于在 **GPTM_CFG** 寄存器。如果要选择 16 位 PWM 模式，需要设置 **AMS** 位为 0x1，**CM** 位为 0x0，并且 **MODE** 域为 0x0 或 0x2。

注意：该寄存器中的位域仅能够在 **GPTM_CTL** 寄存器中的 **TBEN** 位被清零的前提下才能被改变。

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

偏移地址：0x0008

位	名称	读/写	默认值	描述
31: 4	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
3	<i>AMS</i>	R/W	0	GPTM Timer-B复用模式选择 0：捕获模式使能。 1：PWM模式使能。
2	<i>CM</i>	R/W	0	GPTM Timer-B捕获模式 0：边沿计数模式。 1：边沿定时模式。
1:0	<i>MODE</i>	R/W	0x0	GPTM Timer-B模式 0x0：PWM U/D模式。 0x1：单次触发定时器模式。 0x2：周期定时器模式。 0x3：捕获模式。

4.10.4.4 GPTM_CTL - GPTM 控制寄存器

该寄存器与 **GPTM_CFG 寄存器** 和 **GPTM_TAMR/TBMR 寄存器** 一起用来精确调整定时器的配置，也用来使能其它的一些功能，例如定时器停止和输出触发。输出触发可以用来初始化 **ADC 模块** 的传输。

注意：该寄存器中的位域仅能够在 **GPTM_CTL** 寄存器中的 **TnEN** 位被清零的前提下才能被改变。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 R/W	13 R/W	12 RO	11 R/W	10 R/W	9 R/W	8 R/W	7 R/W	6 R/W	5 R/W	4 RO	3 R/W	2 R/W	1 R/W	0 R/W
	TBPWML	TBOTE		TBEVENT	TBSTALL	TBEN	TAPNTBPE	TAPWML	TAOTE			TAEVENT	TASTALL	TAEN	

偏移地址：0x000C

位	名称	读/写	默认值	描述
31:15	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
14	<i>TBPWML</i>	R/W	0	GPTM Timer-B PWM输出 0：原始输出。 1：反转输出。
13	<i>TBOTE</i>	R/W	0	GPTM Timer-B输出触发使能 0：输出Timer-B ADC触发禁止。 0：输出Timer-B ADC触发使能。
12	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
11:10	<i>TBEVENT</i>	R/W	0x0	GPTM Timer-B事件模式 0x0：上升沿。 0x1：下降沿。 0x2：保留。 0x3：上升沿下降沿都有效。
9	<i>TBSTALL</i>	R/W	0	GPTM Timer-B停止使能 0：Timer-B持续计数。 1：Timer-B停止计数。
8	<i>TBEN</i>	R/W	0	GPTM Timer-B使能 0：Timer-B禁止。 1：Timer-B使能并且开始计数。
7	<i>TAPNTBPE</i>	R/W	1	GPTM Timer-A PWM输出和GPTM Timer-B PWM输出互补 0x0：GPTM Timer-A PWM输出和GPTM Timer-B PWM输出互补禁止。 0x1：GPTM Timer-A PWM输出和GPTM Timer-B PWM输出互补使能。
6	<i>TAPWML</i>	R/W	0	GPTM Timer-A PWM输出 0：原始输出。 1：反转输出。
5	<i>TAOTE</i>	R/W	0	GPTM Timer-A输出触发使能 0：输出Timer-A ADC触发禁止。 0：输出Timer-A ADC触发使能。
4	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。

位	名称	读/写	默认值	描述
3:2	TAEVENT	RW	0x0	GPTM Timer-A事件模式 0x0：上升沿。 0x1：下降沿。 0x2：保留。 0x3：上升沿下降沿都有效。
1	TASTALL	RW	0	GPTM Timer-A停止使能 0：Timer-A持续计数。 1：Timer-A停止计数。
0	TAEN	RW	0	GPTM Timer-A使能 0：Timer-A禁止。 1：Timer-A使能并且开始计数。

4.10.4.5 GPTM_IER - GPTM 中断使能寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 WO	9 WO	8 WO	7 RO	6 RO	5 RO	4 RO	3 RO	2 WO	1 WO	0 WO
					CBEIE	CBMIE	TBTOIE						CAEIE	CAMIE	TATOIE

偏移地址：0x0010

位	名称	读/写	默认值	描述
31:11	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
10	CBEIE	WO	0	GPTM Timer-B捕获模式事件中断使能 1：中断使能。
9	CBMIE	WO	0	GPTM Timer-B捕获模式匹配中断使能 1：中断使能。
8	TBTOIE	WO	0	GPTM Timer-B超时中断使能 1：中断使能。
7:3	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
2	CAEIE	WO	0	GPTM Timer-A捕获模式事件中断使能 1：中断使能。
1	CAMIE	WO	0	GPTM Timer-A捕获模式匹配中断使能 1：中断使能。
0	TATOIE	WO	0	GPTM Timer-A超时中断使能 1：中断使能。

4.10.4.6 GPTM_IDR - GPTM 中断禁止寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 WO	9 WO	8 WO	7 RO	6 RO	5 RO	4 RO	3 RO	2 WO	1 WO	0 WO
					CBEID	CBMID	TBTOID						CAEID	CAMID	TATOID

偏移地址：0x0014

位	名称	读/写	默认值	描述
31:11	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
10	<i>CBEID</i>	WO	0	GPTM Timer-B捕获模式事件中断禁止 1：中断禁止。
9	<i>CBMID</i>	WO	0	GPTM Timer-B捕获模式匹配中断禁止 1：中断禁止。
8	<i>TBTOID</i>	WO	0	GPTM Timer-B超时中断禁止 1：中断禁止。
7:3	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
2	<i>CAEID</i>	WO	0	GPTM Timer-A捕获模式事件中断禁止 1：中断禁止。
1	<i>CAMID</i>	WO	0	GPTM Timer-A捕获模式匹配中断禁止 1：中断禁止。
0	<i>TATOID</i>	WO	0	GPTM Timer-A超时中断禁止 1：中断禁止。

4.10.4.7 GPTM_IMR - GPTM 中断屏蔽状态寄存器

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

偏移地址：0x0018

位	名称	读/写	默认值	描述
31:11	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
10	<i>CBEIM</i>	RO	0	GPTM Timer-B捕获模式事件中断屏蔽 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
9	<i>CBMIM</i>	RO	0	GPTM Timer-B捕获模式匹配中断屏蔽 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
8	<i>TBTOIM</i>	RO	0	GPTM Timer-B超时中断屏蔽 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
7:3	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
2	<i>CAEIM</i>	RO	0	GPTM Timer-A捕获模式事件中断屏蔽 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
1	<i>CAMIM</i>	RO	0	GPTM Timer-A捕获模式匹配中断屏蔽 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
0	<i>TATOIM</i>	RO	0	GPTM Timer-A超时中断屏蔽 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。



4.10.4.8 GPTM_RIS - GPTM 原始中断状态寄存器

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

偏移地址：0x001C

位	名称	读/写	默认值	描述
31:11	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
10	<i>CBERI</i>	RO	0	GPTM Timer-B捕获模式事件原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
9	<i>CBMRI</i>	RO	0	GPTM Timer-B捕获模式匹配原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
8	<i>TBTORI</i>	RO	0	GPTM Timer-B超时原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
7:3	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
2	<i>CAERI</i>	RO	0	GPTM Timer-A捕获模式事件原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
1	<i>CAMRI</i>	RO	0	GPTM Timer-A捕获模式匹配原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
0	<i>TATORI</i>	RO	0	GPTM Timer-A超时原始中断状态 0：没有中断产生。 1：该中断正在发出请求。

4.10.4.9 GPTM_ISC - GPTM 原始中断状态和清除寄存器

注意：该寄存器是一个读和写清除寄存器。对其中的某一位写“1”将会清除相对应位的中断状态。

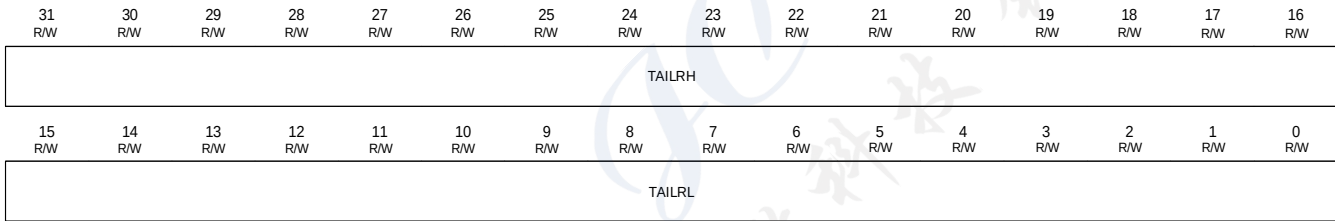
31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RW1C	9 RW1C	8 RW1C	7 RO	6 RO	5 RO	4 RO	3 RO	2 RW1C	1 RW1C	0 RW1C
					CBEISC	CBMISC	TBTOISC						CAEISC	CAMISC	TATOISC

偏移地址：0x0020

位	名称	读/写	默认值	描述
31:11	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
10	<i>CBEISC</i>	RW1C	0	GPTM Timer-B捕获模式事件中断状态和清除 0：没有中断产生或被屏蔽。 1：该中断已发出请求。
9	<i>CBMISC</i>	RW1C	0	GPTM Timer-B捕获模式匹配中断状态和清除 0：没有中断产生或被屏蔽。 1：该中断已发出请求。
8	<i>TBTOISC</i>	RW1C	0	GPTM Timer-B超时中断状态和清除 0：没有中断产生或被屏蔽。 1：该中断已发出请求。
7:3	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
2	<i>CAEISC</i>	RW1C	0	GPTM Timer-A捕获模式事件中断状态和清除 0：没有中断产生或被屏蔽。 1：该中断已发出请求。
1	<i>CAMISC</i>	RW1C	0	GPTM Timer-A捕获模式匹配中断状态和清除 0：没有中断产生或被屏蔽。 1：该中断已发出请求。
0	<i>TATOISC</i>	RW1C	0	GPTM Timer-A超时中断状态和清除 0：没有中断产生或被屏蔽。 1：该中断已发出请求。

4.10.4.10 GPTM_TAILR -GPTM Timer-A 间隔时间装载寄存器

该寄存器用来装载初始计数值到 Timer-A。

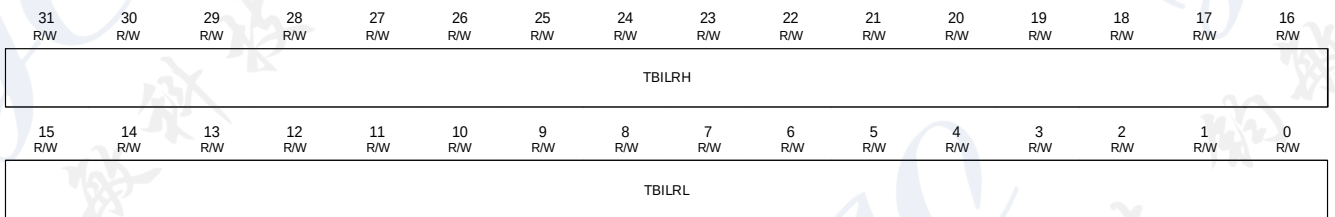


偏移地址：0x0028

位	名称	读/写	默认值	描述
31:16	TAILRH	R/W	0xFFFF	GPTM Timer-A间隔时间装载寄存器高位 32位模式下，通过对该域进行写可以将计数值的高位装载到Timer-A的计数器。读操作将会返回GPTM_TAILR的当前值。在16位模式下，读操作将会返回0。
15:0	TAILRL	R/W	0xFFFF	GPTM Timer-A间隔时间装载寄存器低位 16位或32位模式下，通过对该域进行写可以将计数值的低位装载到Timer-A的计数器。读操作将会返回GPTM_TAILR的当前值。

4.10.4.11 GPTM_TBILR -GPTM Timer-B 间隔时间装载寄存器

该寄存器用来装载初始计数值到 Timer-B。



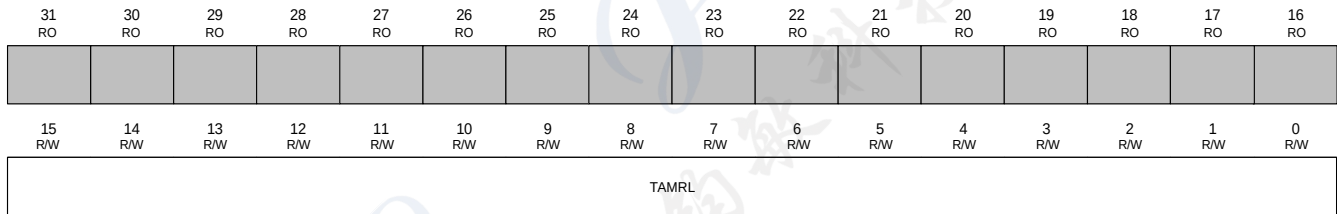
偏移地址：0x002C

位	名称	读/写	默认值	描述
31:16	TBILRH	R/W	0xFFFF	GPTM Timer-B间隔时间装载寄存器高位 32位模式下，通过对该域进行写可以将计数值的高位装载到Timer-B的计数器。读操作将会返回GPTM_TAILR的当前值。在16位模式下，读操作将会返回0。
15:0	TBILRL	R/W	0xFFFF	GPTM Timer-B间隔时间装载寄存器低位 16位或32位模式下，通过对该域进行写可以将计数值的低位装载到Timer-B的计数器。读操作将会返回GPTM_TAILR的当前值。

4.10.4.12 GPTM_TAMATCHR -GPTM Timer-A 匹配值寄存器

该寄存器加载一个匹配值。在单次触发或周期模式中，如果定时器的值等于该寄存器的值则引发一个中断。在边沿计数模式中，该寄存器和 **GPTM_TAILR** 寄存器一起记录共有多少个边沿事件发生。检测到的边沿数等于 **GPTM_TAILR** 的值减去该寄存器里的值。

在 PWM 模式中，该寄存器与 **GPTM_TAILR** 寄存器一起决定输出信号的占空比。



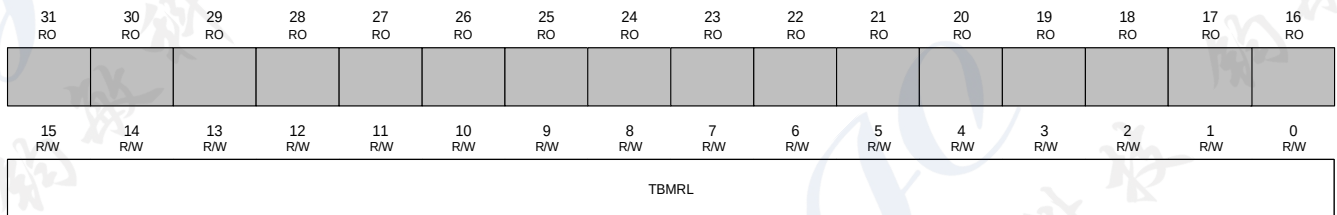
偏移地址：0x0030

位	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:0	<i>TAMRL</i>	R/W	0xFFFF	GPTM Timer-A匹配值寄存器低位

4.10.4.13 GPTM_TBMATCHR -GPTM Timer-B 匹配值寄存器

该寄存器加载一个匹配值。在单次触发或周期模式中，如果定时器的值等于该寄存器的值则引发一个中断。在边沿计数模式中，该寄存器和 **GPTM_TBILR** 寄存器一起记录共有多少个边沿事件发生。检测到的边沿数等于 **GPTM_TBILR** 的值减去该寄存器里的值。

在 PWM 模式中，该寄存器与 **GPTM_TBILR** 寄存器一起决定输出信号的占空比。

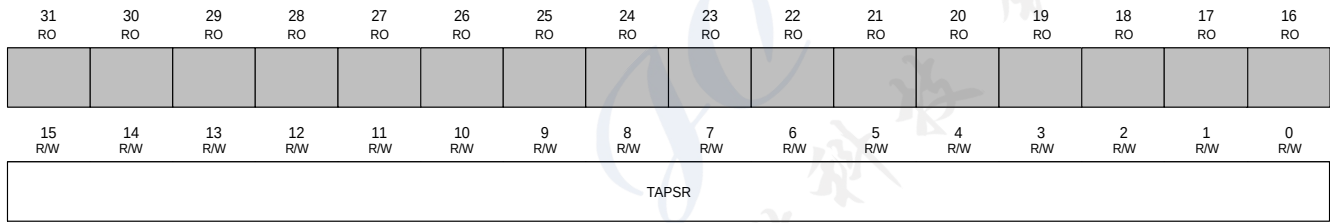


偏移地址：0x0034

位	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:0	<i>TBMRL</i>	R/W	0xFFFF	GPTM Timer-B匹配值寄存器低位

4.10.4.14 GPTM_TAPR -GPTM Timer-A 预分频寄存器

当操作在单次触发或周期模式时，该寄存器允许软件扩展 16 位定时器的定时范围。

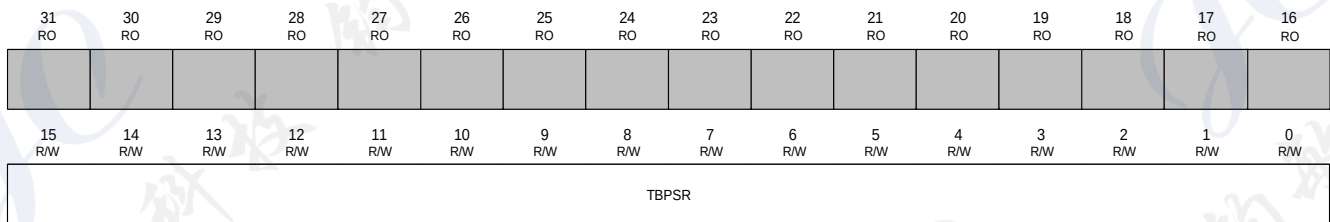


偏移地址：0x0038

位	名称	读/写	默认值	描述
31:16	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:0	TAPSR	R/W	0xFFFF	GPTM Timer-A预分频 对该寄存器写可以装载预分频值。读会返回该寄存器的当前值。

4.10.4.15 GPTM_TBPR -GPTM Timer-B 预分频寄存器

当操作在单次触发或周期模式时，该寄存器允许软件扩展 16 位定时器的定时范围。



偏移地址：0x003C

位	名称	读/写	默认值	描述
31:16	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:0	TBPSR	R/W	0xFFFF	GPTM Timer-B预分频 对该寄存器写可以装载预分频值。读会返回该寄存器的当前值。

4.10.4.16 GPTM_DBC -GPTM 定时器去抖动寄存器

该寄存器仅用于捕捉模式，同时输入信号被去抖动。例如，假设系统时钟为 16MHz，并设置 *DBCNTn* 为 0x03。当输入信号来自 Even ccp 时，4MHz 以上的信号会被忽略。如果系统时钟频率为 16MHz 时，RMS *DBCNTn* 的配置如下：

DBCNTn: "0x0000_0000"信号直接输入。

DBCNTn: "0x0000_0001"8MHz 以上的信号被忽略。

DBCNTn: "0x0000_0011"4MHz 以上的信号被忽略。

DBCNTn: "0x0000_0111"2MHz 以上的信号被忽略。

DBCNTn: "0x0000_1111"1MHz 以上的信号被忽略。

DBCNTn: "0x0001_1111"0.5MHz 以上的信号被忽略。

DBCNTn: "0x0011_1111"250KHz 以上的信号被忽略。

DBCNTn: "0x0111_1111"125KHz 以上的信号被忽略。

DBCNTn: "0x1111_1111"62.5KHz 以上的信号被忽略。

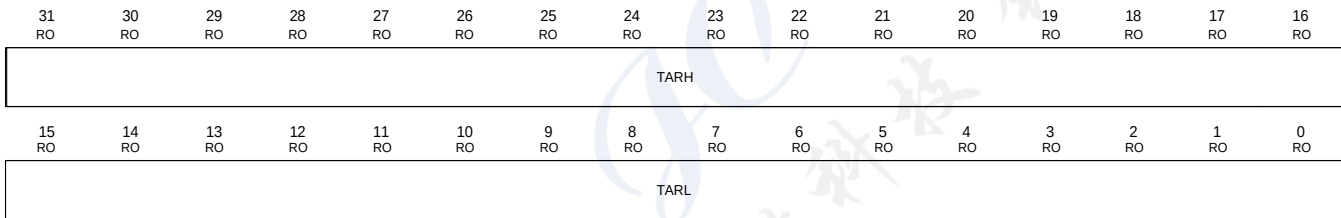
31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 R/W	14 R/W	13 R/W	12 R/W	11 R/W	10 R/W	9 R/W	8 R/W	7 R/W	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
DBCNTB								DBCNTB							

偏移地址：0x0040

位	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:8	<i>DBCNTB</i>	R/W	0x0	Timer-B ccp的去抖动输入
7:0	<i>DBCNTA</i>	R/W	0x0	Timer-A ccp的去抖动输入

4.10.4.17 GPTM_TAR -GPTM Timer-A 寄存器

该寄存器显示 Timer-A 计数器的当前值。

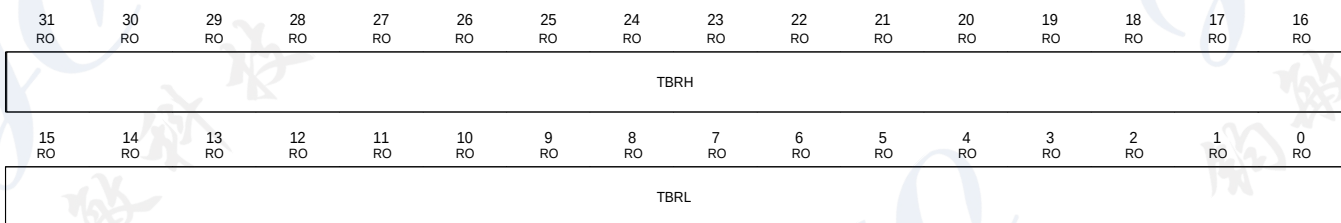


偏移地址：0x0048

位	名称	读/写	默认值	描述
31:16	TARH	RO	0xFFFF	GPTM Timer-A寄存器高位 如果CFG位设置为32位模式，该域表示Timer-A的值的高位。 如果CFG位设置为16位模式，对该域读将会返回无意义的数
15:0	TARL	RO	0xFFFF	GPTM Timer-A寄存器低位 对该域读将会返回在所有情况下Timer-A的当前值，但除输入边沿模式以外，因为它将会返回被计数的边沿数量。

4.10.4.18 GPTM_TBR -GPTM Timer-B 寄存器

该寄存器显示 Timer-B 计数器的当前值。



偏移地址：0x004C

位	名称	读/写	默认值	描述
31:16	TBRH	RO	0xFFFF	GPTM Timer-B寄存器高位 如果CFG位设置为32位模式，该域表示Timer-B的值的高位。 如果CFG位设置为16位模式，对该域读将会返回无意义的数
15:0	TBRL	RO	0xFFFF	GPTM Timer-B寄存器低位 对该域读将会返回在所有情况下Timer-B的当前值，但除输入边沿模式以外，因为它将会返回被计数的边沿数量。

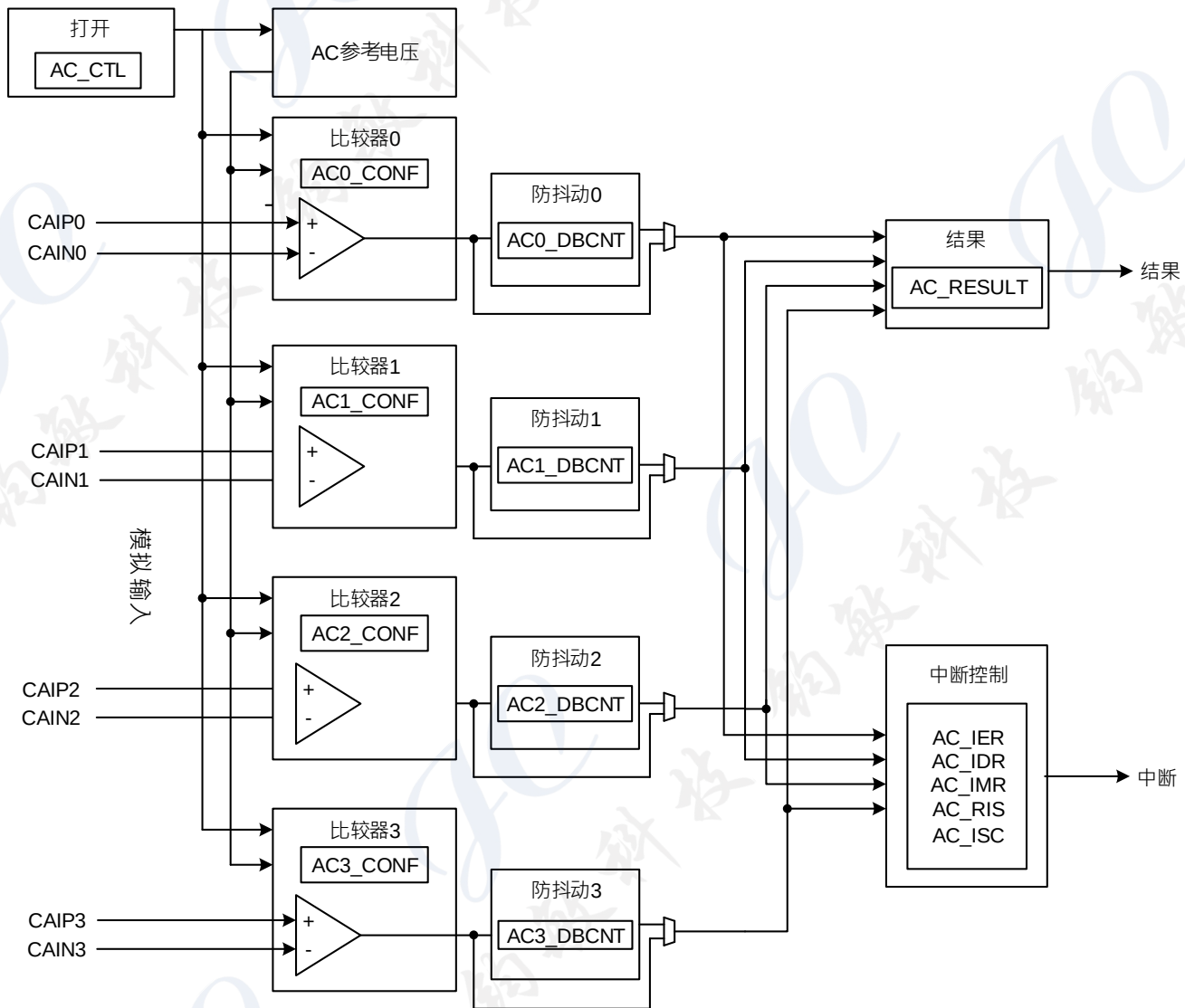
4.11 模拟比较器 (AC)

模拟比较器是一个能够比较两个模拟电压值的外设，并且把比较的结果通过逻辑的形式输出。该模拟比较器支持四个独立的比较器（AC0、AC1、AC2、AC3），它们每个都可以提供输出到器件的引脚并且代替一个模拟比较器。AC 能够通过一个中断来触发 ADC 或通知应用程序使其启动捕获采样序列。独立的外部参考电压。

- 4 个独立的模拟比较器。
- 一个通用外部参考电压。
- 共享的内部参考电压。
- 可编程的去抖动计数器。

4.11.1 块图

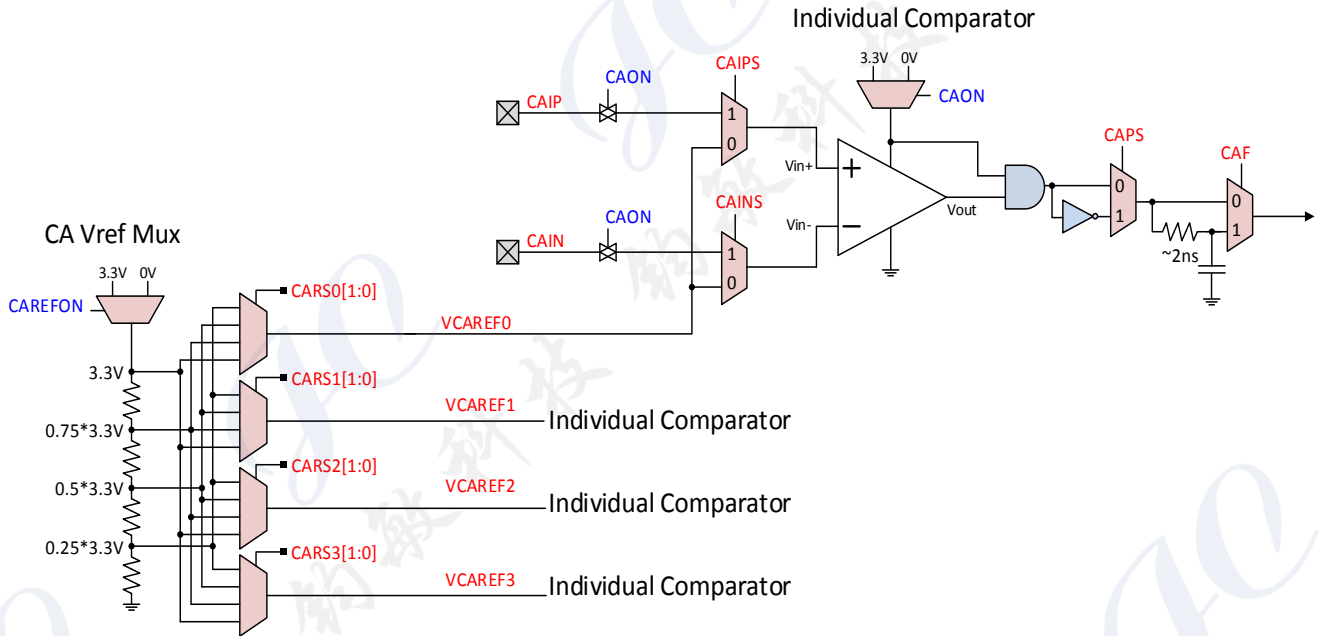
图 4.11-1: 模拟比较器框图



4.11.2 功能描述

比较器比较 CAIP 和 CAIN 的输入，它们的电压源可配置用来产生一个输出。下图显示比较器单元的结构。

图 4.11-2: 比较器单元



4.11.2.1 中断控制

当一个比较器的输出发生变化（例如：Vout 发生变化）时，将会产生中断。AC 的这些中断通过 5 个寄存器被控制。

- 中断控制 (IER, IDR, IMR)

AC 中断使能寄存器(AC_IER)通过简单地写“1”来使能中断请求线。类似地，AC 中断禁止寄存器(AC_IDR)通过写“1”禁止中断请求线。IER 和 IDR 是控制中断屏蔽的只写寄存器。这两个寄存器的总体结果可以通过 AC 中断屏蔽寄存器(AC_IMR)来显示。IMR 是一个只读寄存器，使用“1”或“0”表示是否中断请求线被使能还是被禁止。

- 中断状态读(RIS)

AC 原始中断状态(WDT_RIS)是一个只读寄存器，可以从该模块中读取中断状态。

- 中断清除(ISC)

AC 中断状态&中断清除寄存器(AC_ISC)是用来读取该模块中未被屏蔽的中断，因为只有未被屏蔽的中断才能向处理器发出中断请求。对该寄存器中的某一位写入“1”可以清除相对应的中断状态或通过写“1”到 IDR 禁止该中断。

4.11.3 比较器寄存器映射

基地址: 0x4001_C000

偏移地址	寄存器名称	读/写	默认值	描述	参考页
0x0000	AC_CTRL	R/W	0x0000_0000	AC控制寄存器	216
0x0004	AC0_CONF	R/W	0x0000_0000	AC0配置寄存器	217
0x0008	AC0_BCNT	R/W	0x0000_0000	AC0去抖动寄存器	218
0x000C	AC1_CONF	R/W	0x0000_0000	AC1配置寄存器	217
0x0010	AC1_BCNT	R/W	0x0000_0000	AC1去抖动寄存器	218
0x0014	AC2_CONF	R/W	0x0000_0000	AC2配置寄存器	217
0x0018	AC2_BCNT	R/W	0x0000_0000	AC2去抖动寄存器	218
0x001C	AC3_CONF	R/W	0x0000_0000	AC3配置寄存器	217
0x0020	AC3_BCNT	R/W	0x0000_0000	AC3去抖动寄存器	218
0x0024	AC_RESULT	RO	0x0000_0000	AC输出结果	219
0x0028	AC_IER	WO	0x0000_0000	AC中断使能寄存器	220
0x002C	AC_IDR	WO	0x0000_0000	AC中断禁止寄存器	220
0x0030	AC_IMR	RO	0x0000_0000	AC中断屏蔽状态	221
0x0034	AC_RIS	RO	0x0000_0000	AC原始中断状态	222
0x0038	AC_ISC	R/W1C	0x0000_0000	AC中断状态和清除	223

4.11.3.1 AC_CTL - AC 控制寄存器

此寄存器控制比较器是有效的。每个比较器可以单独使能/禁止。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
											REFEN	AC3EN	AC2EN	AC1EN	AC0EN

偏移地址：0x0000

位	名称	读/写	默认值	描述
31:5	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
4	REFEN	R/W	0	比较器参考电压使能 0：参考电压关闭。 1：参考电压有效。
3	AC3EN	R/W	0	比较器3使能 0：比较器3关闭。 1：比较器3有效。
2	AC2EN	R/W	0	比较器2使能 0：比较器2关闭。 1：比较器2有效。
1	AC1EN	R/W	0	比较器1使能 0：比较器1关闭。 1：比较器1有效。
0	AC0ED	R/W	0	比较器0使能 0：比较器0关闭。 1：比较器0有效。

4.11.3.2 ACn_CONF - AC 0/1/2/3 配置寄存器

此寄存器控制比较器 x 的输出，这里 x 是比较器的代码。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
									ADCTRI	ACRVS	ACF	ACPS	CAINS	CAIPS	

偏移地址：

- CA0_CONF: 0x0004
- CA1_CONF: 0x000C
- CA2_CONF: 0x0014
- CA3_CONF: 0x001C

位	名称	读/写	默认值	描述
31:7	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
6	ADCTRI	R/W	0	ADC触发使能 0：ADC触发关闭。 1：ADC触发有效。
5:4	ACRVS	R/W	0x0	模拟比较器参考电压选择 00：0.25 * 1.8V。 01：0.5 * 1.8V。 10：0.75 * 1.8V。 11：1.8V。
3	ACF	R/W	0	模拟比较器滤波电路使能 0：AC滤波电路关闭。 1：AC滤波电路有效。
2	ACPS	R/W	0	模拟比较器反向电路使能 0：AC反向电路关闭。 1：AC反向电路有效。
1	CAINS	R/W	0	比较器负端输入选择 0：选择参考电压。 1：选择输入电压。
0	CAIPS	R/W	0	比较器正端输入选择 0：选择参考电压。 1：选择输入电压。

4.11.3.3 ACn_DBCNT - AC 0/1/2/3 去抖动寄存器

这个寄存器控制比较器 n 的去抖动计数器的值，这里 n 是比较器的代码。例如，假设系统时钟频率为 16MHz，并设置 DBCNT 为 0x03，当输入信号来自 CAO 时，它将会忽略上述 4MHz 的信号。如果系统时钟频率为 16MHz，RMS DBCNT 如下：

DBCNT: “0x0000_0000”信号直接输入。

DBCNT: “0x0000_0001”8MHz 以上的信号被忽略，并且检测输入。

DBCNT: “0x0000_0011”4MHz 以上的信号被忽略，并且检测输入。

DBCNT: “0x0000_0111”2MHz 以上的信号被忽略，并且检测输入。

DBCNT: “0x0000_1111”1MHz 以上的信号被忽略，并且检测输入。

DBCNT: “0x0001_1111”0.5MHz 以上的信号被忽略，并且检测输入。

DBCNT: “0x0011_1111”250KHz 以上的信号被忽略，并且检测输入。

DBCNT: “0x0111_1111”125KHz 以上的信号被忽略，并且检测输入。

DBCNT: “0x1111_1111”62.5Hz 以上的信号被忽略，并且检测输入。

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

偏移地址：

- AC0_CONF: 0x0008
- AC1_CONF: 0x0010
- AC2_CONF: 0x0018
- AC3_CONF: 0x0020

位	名称	读/写	默认值	描述
31:8	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
7:0	DBCNT	R/W	0x0	比较器去抖动计数器值

4.11.3.4 AC_RESULT - AC 输出结果

读这个寄存器将会得到比较器的输出值。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
												ACO3	ACO2	ACO1	ACO0

偏移地址：0x0024

位	名称	读/写	默认值	描述
31:5	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
4	ACO3	RO	0	Vin +和Vin-是比较器3的正负输入电压（通过AC3_CONF中的CAIPS和CAINS位来定义）。 0：Vin + Vin -。 1：Vin + Vin -。
3	ACO2	RO	0	Vin +和Vin-是比较器2的正负输入电压（通过AC2_CONF中的CAIPS和CAINS位来定义）。 0：Vin + Vin -。 1：Vin + Vin -。
2	ACO1	RO	0	Vin +和Vin-是比较器1的正负输入电压（通过AC1_CONF中的CAIPS和CAINS位来定义）。 0：Vin + Vin -。 1：Vin + Vin -。
0	ACO0	RO	0	Vin +和Vin-是比较器1的正负输入电压（通过AC0_CONF中的CAIPS和CAINS位来定义）。 0：Vin + Vin -。 1：Vin + Vin -。

4.11.3.5 AC_IER - AC 中断使能寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 WO	2 WO	1 WO	0 WO
												AC3IE	AC2IE	AC1IE	AC0IE

偏移地址：0x0028

位	名称	读/写	默认值	描述
31:4	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
3	AC3IE	WO	0	比较器3中断使能 1：中断使能。
2	AC2IE	WO	0	比较器2中断使能 1：中断使能。
1	AC1IE	WO	0	比较器1中断使能 1：中断使能。
0	AC0IE	WO	0	比较器0中断使能 1：中断使能。

4.11.3.6 AC_IDR - AC 中断禁止寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 WO	2 WO	1 WO	0 WO
												AC3ID	AC2ID	AC1ID	AC0ID

偏移地址：0x002C

位	名称	读/写	默认值	描述
31:4	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
3	AC3ID	WO	0	比较器3中断禁止 1：中断禁止。
2	AC2ID	WO	0	比较器2中断禁止 1：中断禁止。
1	AC1ID	WO	0	比较器1中断禁止 1：中断禁止。
0	AC0ID	WO	0	比较器0中断禁止 1：中断禁止。

4.11.3.7 AC_IMR - AC 中断屏蔽状态

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
												AC3IM	AC2IM	AC1IM	AC0IM

偏移地址：0x0030

位	名称	读/写	默认值	描述
31:4	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
3	AC3IM	RO	0	比较器3中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
2	AC2IM	RO	0	比较器2中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
1	AC1IM	RO	0	比较器1中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
0	AC0IM	RO	0	比较器0中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。

4.11.3.8 AC_RIS - AC 原始中断状态

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
												AC3RI	AC2RI	AC1RI	AC0RI

偏移地址：0x0034

位	名称	读/写	默认值	描述
31:4	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
3	AC3RI	RO	0	比较器3原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
2	AC2RI	RO	0	比较器2原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
1	AC1RI	RO	0	比较器1原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
0	AC0RI	RO	0	比较器0原始中断状态 0：没有中断产生。 1：该中断正在发出请求。

4.11.3.9 AC_ISC - AC 中断状态和清除寄存器

注：该寄存器是读和写清除寄存器。对单独的某一位写“1”将会清除与之相对应位的中断状态。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 R/W1C	2 R/W1C	1 R/W1C	0 R/W1C
												AC3IS	AC2IS	AC1IS	AC0IS

偏移地址：0x0038

位	名称	读/写	默认值	描述
31:4	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
3	AC3RI	RO	0	比较器3原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
2	AC2RI	RO	0	比较器2原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
1	AC1RI	RO	0	比较器1原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
0	AC0RI	RO	0	比较器0原始中断状态 0：没有中断产生。 1：该中断正在发出请求。

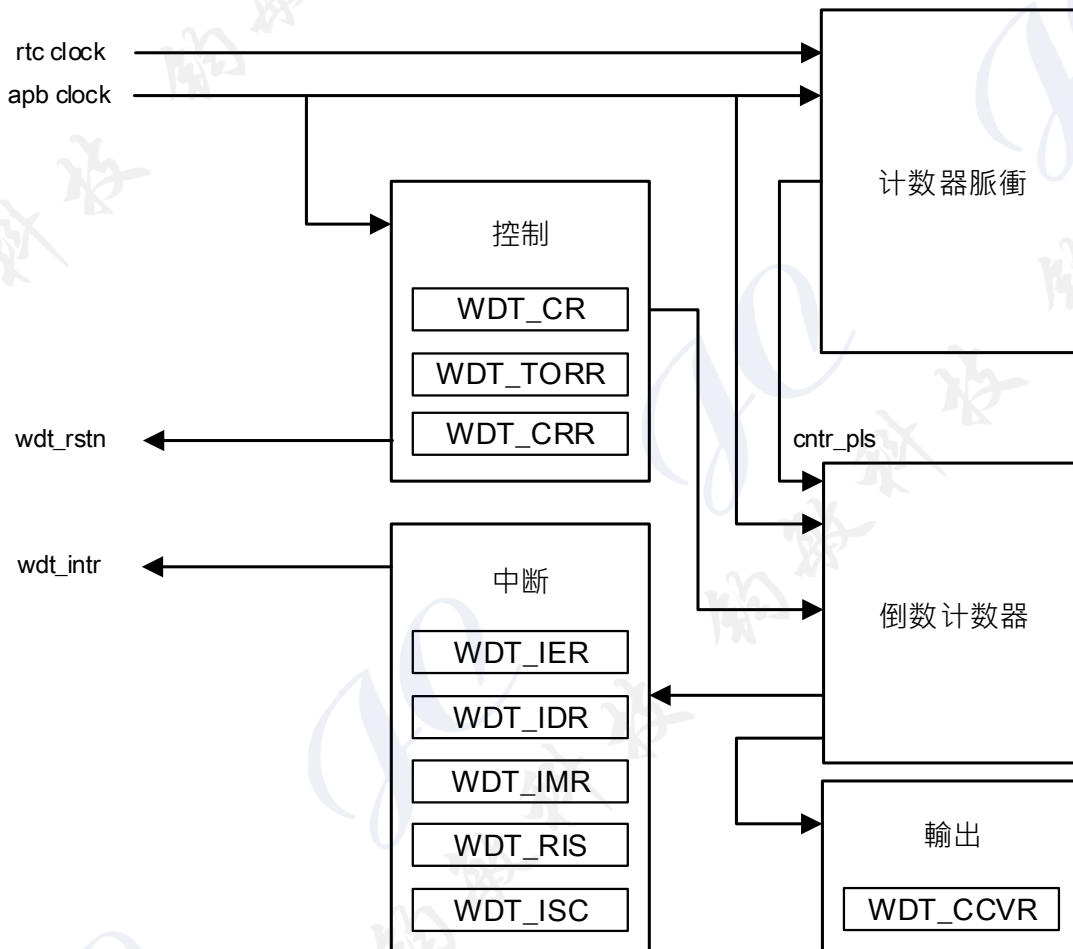
4.12 看门狗定时器 (WDT)

看门狗定时器的目的是在软件故障或外部设备故障后进行系统复位。这样可以防止系统在一个无限的时间内挂起。WDT 被配置为一个预定义的超时时间，并在使能后不断地运行。如果 WDT 没有在超时期限内被清除，它会发出一个系统复位。此外，这个看门狗定时器支持从掉电模式中唤醒 CPU 的功能。它包括一个 32 位的独立运行的计数器以及可编程的时间间隔。

- 可防止系统死锁的两阶段模式
 - 一阶段模式：WDT 从一个预设值（超时）开始计数直到该值减到零。当该计数器达到零时，它可以产生中断或简单地复位系统，同时它替换到超时值并且继续递减。
 - 二阶段模式：当首次超时发生时，WDT 可以编程产生一个中断。如果它没有在第二次超时发生的时刻被清除时，然后它将会产生一个系统复位。如果复位和看门狗计数器到达零在相同的时间发生时，不会产生中断。
- 为看门狗定时器超时时间间隔提供一个 32 位独立运行的 WDT 计数器。
- 可选的超时时间间隔（ 2^4 到 2^{18} ），超时时间间隔为 104 ms ~ 26.316s（如果 WDT_CLK = 10kHz）。
- 复位周期 = $(1 / 10\text{kHz}) * 63$ ，如果 WDT_CLK = 10kHz。

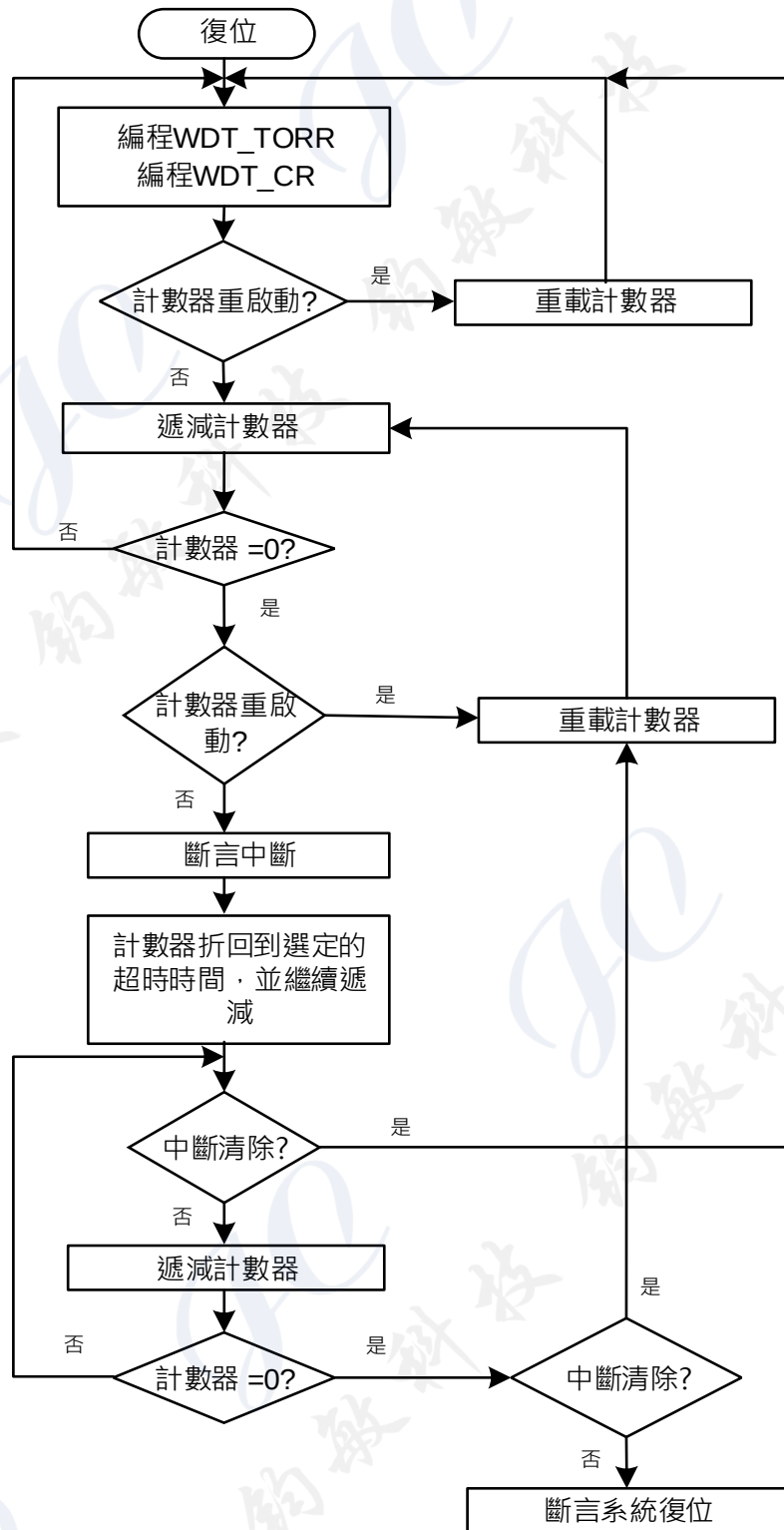
4.12.1 块图

图 4.12-1: 看门狗定时器框图



4.12.2 操作流程图

图 4.12-2: WDT 操作流程图



4.12.3 功能描述

4.12.3.1 中断控制

当超时发生时，WDT 可以编程产生一个中断（一个系统复位）。WDT 的中断可以通过 5 个寄存器来控制。

- 中断控制（IER, IDR, IMR）

WDT 中断使能寄存器(WDT_IER)通过简单地写‘1’去使能中断请求线。类似地，WDT 中断屏蔽寄存器(WDT_IDR)通过简单地写‘1’去禁止中断请求线。IER 和 IDR 是只写寄存器，上述寄存器的总体结果可以通过 WDT 中断屏蔽寄存器(WDT_IMR)被显示。IMR 是一个只读寄存器，使用“1”或“0”表示是否中断请求线被使能还是被禁止。

- 中断状态读(RIS)

WDT 原始中断状态(WDT_RIS)是一个只读寄存器，可以从该模块中读取中断状态。

- 中断清除(ISC)

WDT 中断状态&中断清除寄存器(WDT_ISC)是用来读取该模块中被屏蔽的中断，显示哪一个中断未被屏蔽。在 ISC 中的每一位是 RIS 和 IMR 中的相对应位的逻辑与。对该寄存器中的某一位写入“1”可以清除相对应的中断状态，但不是中断本身。

4.12.4 WDT 寄存器映射

基地址: 0x4000_4000

偏移地址	寄存器名称	读/写	默认值	描述	参考页
0x0000	WDT_CTRL	R/W	0x0000_0000	WDT控制寄存器	228
0x0004	WDT_TORR	R/W	0x0000_0000	WDT超时范围寄存器	229
0x0008	WDT_CCVR	RO	0x0000_FFFF	WDT当前计数器值寄存器	230
0x000C	WDT_CRR	WO	0x0000_0000	WDT计数器重启寄存器	230
0x0010	WDT_IER	WO	0x0000_0000	WDT中断使能寄存器	230
0x0014	WDT_IDR	WO	0x0000_0000	WDT中断禁止寄存器	231
0x0018	WDT_IMR	RO	0x0000_0000	WDT中断屏蔽寄存器	231
0x001C	WDT_RIS	RO	0x0000_0000	WDT原始中断状态寄存器	232
0x0020	WDT_ISC	R/W1C	0x0000_0000	WDT中断状态和清除寄存器	232

4.12.4.1 WDT_CTRL - WDT 控制寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
									CLKSEL		RSTWTH		MODEL	ENABLE	

偏移地址：0x0000

位	名称	读/写	默认值	描写
31:6	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
5	<i>CLKSEL</i>	R/W	0	WDT时钟源选择 0：APB时钟(PCLK)。 1：RTC时钟(RTCCLK)。
4:2	<i>RSTWTH</i>	R/W	0	复位信号脉冲宽度 该域是用来为系统复位停留所选择pclk周期的数量。该值的可用范围为2到256个pclk周期。 000：2个pclk周期。 001：4个pclk周期。 010：8个pclk周期。 011：16个pclk周期。 100：32个pclk周期。 101：64个pclk周期。 110：128个pclk周期。 111：256个pclk周期。
1	<i>MODEL</i>	R/W	0	WDT二阶段模式选择 0：执行二阶段模式。 1：执行一阶段模式。
0	<i>ENABLE</i>	R/W	0	WDT使能 0：WDT禁止，没有中断或系统复位将会被产生。 1：WDT使能。一旦此位被使能，它仅能够通过系统复位才被清除。

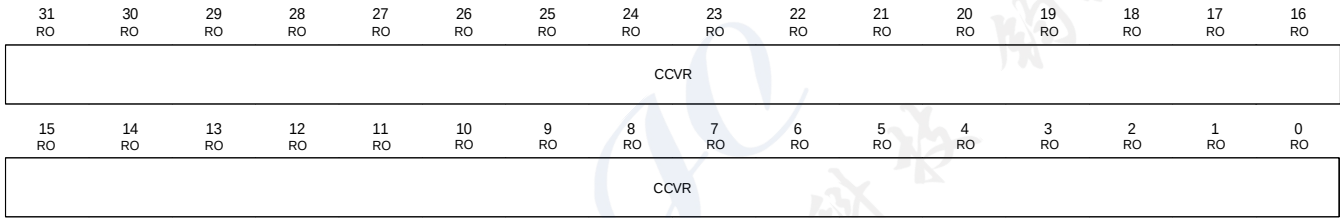
4.12.4.2 WDT_TORR - WDT 超时范围寄存器

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

偏移地址：0x0004

位	名称	读/写	默认值	描述
31:8	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
7:4	<i>SNDINIT</i>	R/W	0	第二次超时宽度周期初始化 此超时范围在一阶段模式下无效。此域用来选择看门狗计数器为第一次计数器重启而重启的超时时间。 0x0 : 0xFFFF 0x0 : 0xFFFFFFFF 0x1 : 0x1FFFF 0x1 : 0x1FFFFFFF 0x2 : 0x3FFFF 0x2 : 0x3FFFFFFF 0x3 : 0x7FFFF 0x3 : 0x7FFFFFFF 0x4 : 0xFFFFF 0x4 : 0xFFFFFFFF 0x5 : 0x1FFFF 0x5 : 0x1FFFFFFF 0x6 : 0x3FFFFF 0x6 : 0x3FFFFFFF 0x7 : 0x7FFFFF 0x7 : 0x7FFFFFFF
3:0	<i>FSTINIT</i>	R/W	0	首次超时宽度周期初始化 此域用来选择从看门狗计数器重启的超时时间。 0x0 : 0xFFFF 0x0 : 0xFFFFFFFF 0x1 : 0x1FFFF 0x1 : 0x1FFFFFFF 0x2 : 0x3FFFF 0x2 : 0x3FFFFFFF 0x3 : 0x7FFFF 0x3 : 0x7FFFFFFF 0x4 : 0xFFFFF 0x4 : 0xFFFFFFFF 0x5 : 0x1FFFF 0x5 : 0x1FFFFFFF 0x6 : 0x3FFFFF 0x6 : 0x3FFFFFFF 0x7 : 0x7FFFFF 0x7 : 0x7FFFFFFF

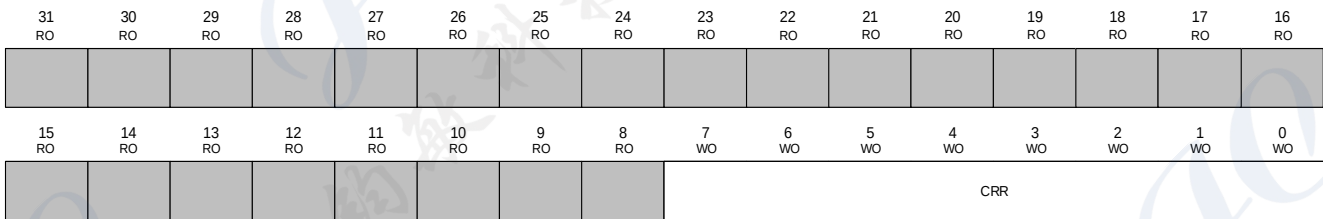
4.12.4.3 WDT_CCVR - WDT 当前计数器值寄存器



偏移地址：0x0008

位	名称	读/写	默认值	描述
31:0	CCVR	RO	0xFFFF	WDT当前计数器值 内部计数器的当前值。

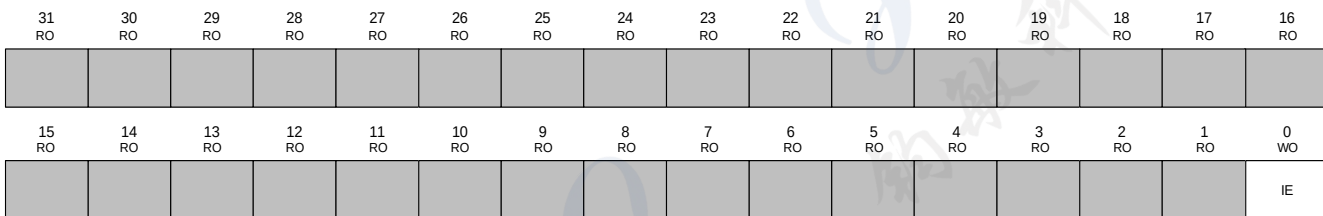
4.12.4.4 WDT_CRR- WDT 计数器重启寄存器



偏移地址：0x000C

位	名称	读/写	默认值	描述
31:8	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
7:0	CRR	WO	0x0	WDT计数器重启 该寄存器用来重启WDT计数器。

4.12.4.5 WDT_IER- WDT 中断使能寄存器



偏移地址：0x0010

位	名称	读/写	默认值	描述
31:1	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
0	IER	WO	0	WDT中断使能 1：该中断使能。

4.12.4.6 WDT_IDR - WDT 中断禁止寄存器

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

偏移地址：0x0014

位	名称	读/写	默认值	描述
31:1	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
0	<i>IDR</i>	WO	0	WDT中断禁止 1：该中断禁止。

4.12.4.7 WDT_IMR- WDT 中断屏蔽寄存器

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

偏移地址：0x0018

位	名称	读/写	默认值	描述
31:1	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
0	<i>IMR</i>	WO	0	WDT中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。

4.12.4.8 WDT_RIS- WDT 原始中断状态

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

偏移地址：0x001C

位	名称	读/写	默认值	描述
31:1	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
0	<i>RIS</i>	RO	0	WDT原始中断状态 0：没有中断产生。 1：该中断正在发出请求。

4.12.4.9 WDT_ISC- WDT 中断状态和清除

注：该寄存器是读和写清除寄存器。对其中的某一位写“1”将会清除相对应位的中断状态。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 R/W1C
															ISC

偏移地址：0x0020

位	名称	读/写	默认值	描述
31:1	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
0	<i>ISC</i>	R/W1C	0	WDT中断状态和清除 0：没有中断产生或中断被屏蔽。 1：该中断已经发出请求。

4.13 实时时钟 (RTC)

实时时钟提供给用户实际时间和日历信息。RTC 可以从低功耗模式中唤醒装置。

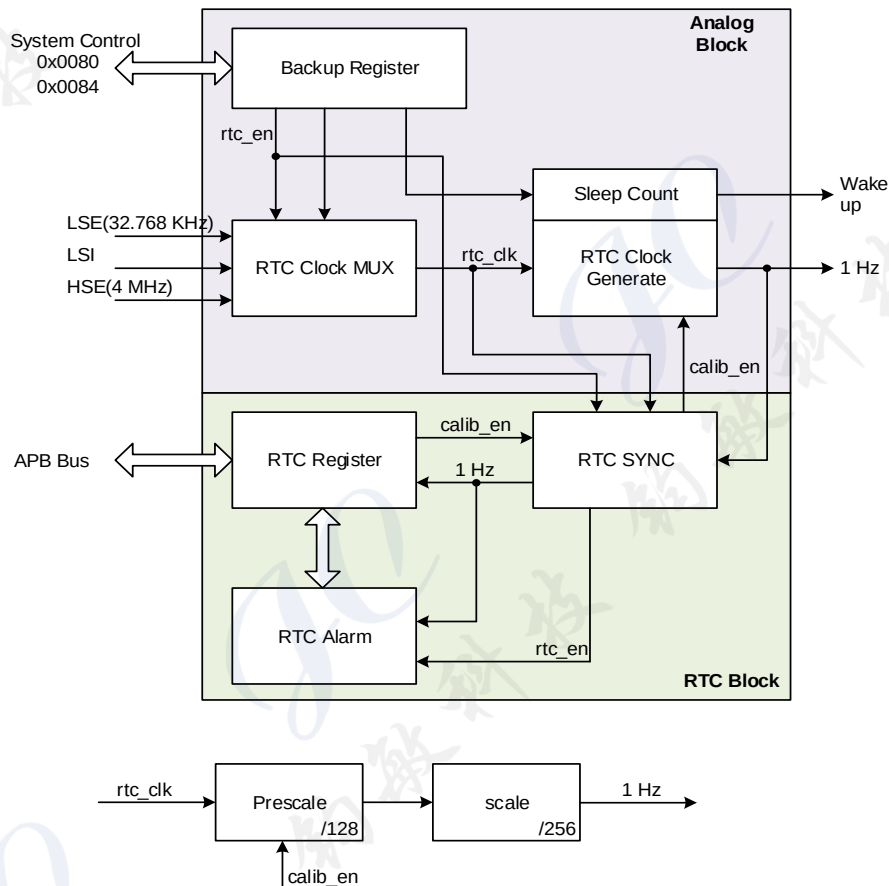
RTC 的时钟源可以来自一个 32.768kHz 的低速外部晶体或陶瓷谐振器，一个 LSI RC 和一个 PLL 除以 128。参考时钟分布。

RTC 控制器提供日历信息（世纪，年，月，周以及日），也提供时间信息（时，分，秒）。所有的数据信息皆为 BCD 格式。RTC 也提供闹钟功能，用户可以分别预先设置闹钟时间和闹钟日历。

- 日历：世纪，年，月，周和日。
- 时间：时，分，秒。
- 可编程闹钟的中断功能，它能够通过任意组合被触发：世纪，年，月，天，星期，时，分和秒。
- 自动的闰年补偿。
- 触发 ADC 或 DAC
- 所有的时间和日历信息都以 BCD 码格式表示。
- 翻转中断：世纪、年、月、日、星期、小时、分、秒。
- 频率补偿：时钟偏差可以通过校准计数器进行补偿。
- 支持备份寄存器。

4.13.1 块图

图 4.13-1: RTC 框图



4.13.2 功能描述

4.13.2.1 RTC 使能

当用户想要使能 RTC 功能时，需要设置系统备份 RTC 控制寄存器(SC_BKRTC_CTRL)。SC_BKRTC_CTRL 可以配置 RTC 使能，预分频数值，分频值，校准值和校准的减少/增加。

例如：若设置 $COUNTEN = 1$ ， $CLKPSCAL = 0x7F$ 和 $CLKSCAL = 0xFF$ 。在此配置下，定时器将会计数 32768 次，然后才使得秒加 1。

4.13.2.2 RTC 时间和日历信息的初始化和配置

当用户想要改变 RTC 的时间和日历信息时，需要先设置 RTC_CTRL，这意味着开始写入数据到定时器，然后设置数据到 RTC_TIME 和 RTC_CAL 寄存器。当设置完成时，清除 RTC_CTRL，表示 RTC 使能。

4.13.2.3 频率补偿

RTC 的频率补偿机制讓軟體有機會為一個太快或太慢的 RTC 時鐘源作出數位补偿。以下展示了兩個針對於過快或過慢的時鐘源作出的补偿例子。

計算校準值的公式如下：

1. 測量值整數 – 預期值整數 = 校驗值整數，正值表示時鐘源太快，負值表示時鐘源太慢。
2. 測量值分數 $\times 60$ = 校驗值分數

表 4.13-1: 补偿例子

校準一個過於快速的振盪源	校準一個過於慢的振盪源
頻率計數器測量: 32773.65 Hz 校驗值_{整數} = 32773 – 32768 = +5 校驗值_{分數} = $0.65 \times 60 = +39$ 在這種情況下，每1秒鐘補償值為+5，對於每60秒，補償值為+39 寄存器配置: 1. 寫 1 到 CALIBEN 和 CALIBCYC 在 RTC_CALIB 寄存器. 2. 配置 SC_BKRTC_CTRL 寄存器: a. 寫 0x1 到 COUNTEN 去致能 RTC 的計數; b. 寫 0x1 到 CKSEL 去選擇 RTC 的時鐘源; 寫 0x7F 及 0xFF 到 CLKPSCAL 和 CLKSCAL 去讓 RTC 計數 32768 次; c. 寫 0x0 及 0x5 到 CRS 和 CRV，每 1 秒鐘，RTC 數 32773(32768+5)。每 60 秒鐘，改變 CRV 值從 0x5 到 0x2C (+5+39=44)。在其餘時間，CRV 值為 0x5. 此操作用于补偿小数部分错误。	頻率計數器測量: 32765.27 Hz 校驗值_{整數} = 32765 – 32768 = -3 校驗值_{分數} = $0.27 \times 60 = +16.2$ (i.e. 16) 在這種情況下，每 1 秒鐘補償值為-3，對於每 60 秒，補償值為+16 寄存器配置: 1. 寫 1 到 CALIBEN 和 CALIBCYC 在 RTC_CALIB 寄存器. 2. 配置 SC_BKRTC_CTRL 寄存器: a. 寫 0x1 到 COUNTEN 去致能 RTC 的計數; b. 寫 0x1 到 CKSEL 去選擇 RTC 的時鐘源; c. 寫 0x7F 及 0xFF 到 CLKPSCAL 和 CLKSCAL 去讓 RTC 計數 32768 次; d. 寫 0x1 & 0x3 到 CRS 和 CRV，每 1 秒鐘，RTC 數 32765 (32768-3)。每 60 秒鐘，改變 CRV 值從 0x3 到 0xD (-3+39=44)。在其餘時間，CRV 值為 0x3. 此操作用于补偿小数部分错误。

4.13.2.4 闹钟中断

当闹钟使能并且在 **RTC_TIME** 和 **RTC_CAL** 中的 RTC 时间等于在 **RTC_ARTIME** 和 **RTC_ARCAL** 中被设置的闹钟时间时，闹钟中断标志(**RTC_RIS** [7:0])会被置位，并且如果闹钟中断被使能时，它会向 CPU 发出中断请求。

周期闹钟中断：

如果启用在 **RTC_ALMEN** 中的任何一个闹钟，并且使能它在 **RTC_IER** 中的中断。在每当时间等于闹钟设置的时间时，会发出中断请求。例如：若配置闹钟使能寄存器 **RTC_ALMEN** 为 0x2,在 **RTC_ALMTIME** 中配置分钟数据为 0x50，并且在 **RTC_IER** 中使能分钟中断，则每当时间到达 50 分钟时，会发出中断请求。

所有闹钟中断：

如果开启在 **RTC_ALMEN** 中的所有闹钟，并且在 **RTC_IER** 中使能所有的中断，则仅当在 **RTC_TIME** 和 **RTC_CAL** 中的全部时间等于在 **RTC_ALMTIME** & **RTC_ALMCAL** 中被设置的全部时间时，才会发出中断请求。例如：若配置时钟使能寄存器 **RTC_AREN** 为 0xFF，日历闹钟寄存器 **RTC_ALMCAL** 为 0x20151224，时间闹钟寄存器 **RTC_ALMTIME** 为 0x06173000，并且在 **RTC_IER** 中使能所有闹钟中断，当时间为 2015/12/24，星期六，17:30:00 时，中断请求会被发出。

4.13.3 RTC 寄存器映射

基地址: 0x4000_8000

偏移地址	寄存器名称	读/写	默认值	描述	参考页
0x0000	RTC_CTRL	R/W	0x00FF_7F00	RTC 控制寄存器	237
0x0004	RTC_TIME	R/W	0x0000_0000	RTC 日历时间寄存器	238
0x0008	RTC_CAL	R/W	0x0000_0000	RTC 日历设置寄存器	239
0x000C	RTC_ALMTIME	R/W	0x0000_0000	RTC 闹钟时间寄存器	240
0x0010	RTC_ALMCAL	R/W	0x0000_0000	RTC 闹钟日历寄存器	241
0x0018	RTC_ALMEN	R/W	0x0000_0000	RTC 闹钟使能寄存器	242
0x001C	RTC_CALIB	R/W	0x0000_0000	RTC 校准设置寄存器	243
0x0020	RTC_TRIG	R/W	0x0000_0000	RTC 触发选择寄存器	243
0x0024	RTC_IER	WO	0x0000_0000	RTC 中断使能寄存器	245
0x0028	RTC_IDR	WO	0x0000_0000	RTC 中断禁止寄存器	246
0x002C	RTC_IMR	RO	0x0000_0000	RTC 中断屏蔽寄存器	248
0x0030	RTC_RIS	RO	0x0000_0000	RTC 原始中断状态寄存器	250
0x0034	RTC_ISC	R/W1C	0x0000_0000	RTC 中断状态和清除寄存器	252

4.13.3.1 RTC_CTRL - RTC 控制寄存器

当在 RTC 中写寄存器时，这个寄存器用来暂时地禁止在 RTC 定时器和日历寄存器中的任何更新。

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

偏移地址：0x0000

位	名称	读/写	默认值	描述
31:1	保留	RO	0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
0	RTCEN	R/W	0	RTC初始化 0: RTC计数器被禁用 1: RTC计数器启用。

4.13.3.2 RTC_TIME - RTC 时间设置寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 R/W	25 R/W	24 R/W	23 RO	22 RO	21 R/W	20 R/W	19 R/W	18 R/W	17 R/W	16 R/W
					WEEK					HRTEN		HRUNIT			
15 RO	14 R/W	13 R/W	12 R/W	11 R/W	10 R/W	9 R/W	8 R/W	7 RO	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 RO
	MINTEN			MINUNIT					SECTEN		SECUNIT				

偏移地址：0x0004

位	名称	读/写	默认值	描述
31:27	保留	RO	0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
26:24	WEEK	R/W	0x0	星期值 使用BCD格式。
23:22	保留	RO	0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
21:20	HRTEN	R/W	0x0	时钟十位值 使用BCD格式。
19:16	HRUNIT	R/W	0x0	时钟个位值 使用BCD格式。
15	保留	RO	0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
14:12	MINTEN	R/W	0x0	分钟十位值 使用BCD格式。
11:8	MINUNIT	R/W	0x0	分钟个位值 使用BCD格式。
7	保留	RO	0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
6:4	SECTEN	R/W	0x0	秒钟十位值 使用BCD格式。
3:0	SECUNIT	R/W	0x0	秒钟个位值 使用BCD格式。

4.13.3.3 RTC_CAL - RTC 日历设置寄存器

31 R/W	30 R/W	29 R/W	28 R/W	27 R/W	26 R/W	25 R/W	24 R/O	23 R/W	22 R/W	21 R/W	20 R/W	19 R/W	18 R/W	17 R/W	16 R/W
CENTEN				CENUNIT				YRTEN				YRUNIT			
15 R/O	14 R/O	13 R/O	12 R/W	11 R/W	10 R/W	9 R/W	8 R/W	7 R/O	6 R/O	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
			MHTEN	MTHUNIT						DATETEN		DATEUNIT			

偏移地址：0x0008

位	名称	读/写	默认值	描述
31:28	CENTEN	R/W	0x0	世纪十位值 使用BCD格式
27:24	CENUNIT	R/W	0x0	世纪个位值 使用BCD格式
23:20	YRTEN	R/W	0x0	年数十位值 使用BCD格式。
19:16	YRUNIT	R/W	0x0	年数个位值 使用BCD格式。
15:13	reserved	R/O	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
12	MHTEN	R/W	0x0	月数十位值 使用BCD格式。
11:8	MTHUNIT	R/W	0x0	月数个位值 使用BCD格式。
7:6	reserved	R/O	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
5:4	DATETEN	R/W	0x0	天数十位值 使用BCD格式。
3:0	DATEUNIT	R/W	0x0	天数个位值 使用BCD格式。

4.13.3.4 RTC_ALMTIME - RTC 闹钟时间设置寄存器

这个寄存器是和 RTC_ARCAL 一起用来设置 RTC 闹钟。

31 RO	30 RO	29 RO	28 RO	27 RO	26 R/W	25 R/W	24 R/W	23 RO	22 RO	21 R/W	20 R/W	19 R/W	18 R/W	17 R/W	16 R/W
					WEEK					HRTEN		HRUNIT			
15 RO	14 R/W	13 R/W	12 R/W	11 R/W	10 R/W	9 R/W	8 R/W	7 RO	6 RO	5 RO	4 RO	3 R/W	2 R/W	1 R/W	0 R/W
	MINTEN				MINUNIT				SECTEN			SECUNIT			

偏移地址：0x000C

位	名称	读/写	默认值	描述
31:27	保留	RO	0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
26:24	WEEK	R/W	0x0	闹钟星期值 使用BCD格式。
23:22	保留	RO	0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
21:20	HRTEN	R/W	0x0	闹钟时钟十位值 使用BCD格式。
19:16	HRUNIT	R/W	0x0	闹钟时钟个位值 使用BCD格式。
15	保留	RO	0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
14:12	MINTEN	R/W	0x0	闹钟分钟十位值 使用BCD格式。
11:8	MINUNIT	R/W	0x0	闹钟分钟个位值 使用BCD格式。
7	保留	RO	0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
6:4	SECTEN	R/W	0x0	闹钟秒钟十位值 使用BCD格式。
3:0	SECUNIT	R/W	0x0	闹钟秒钟个位值 使用BCD格式。

4.13.3.5 RTC_ALMCAL - RTC 闹钟日历设置寄存器

这个寄存器是和 **RTC_ARTIME** 一起用来设置 RTC 闹钟。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 R/W	22 RO	21 R/W	20 R/W	19 R/W	18 R/W	17 R/W	16 R/W
								YRTEN				YRUNIT			
15 RO	14 RO	13 RO	12 R/W	11 R/W	10 R/W	9 R/W	8 R/W	7 RO	6 RO	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
			MHTTEN	MTHUNIT						DATETEN		DATEUNIT			

偏移地址：0x0010

位	名称	读/写	默认值	描述
31: 28	CENTEN	R/W	0x0	闹钟世纪十位值 使用BCD格式。
27:24	CENUNIT	R/W	0x0	闹钟世纪个位值 使用BCD格式。
23:20	YRTEN	R/W	0x0	闹钟年数十位值 使用BCD格式。
19:16	YRUNIT	R/W	0x0	闹钟年数个位值 使用BCD格式。
15:13	保留	RO	0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
12	MHTTEN	R/W	0x0	闹钟月数十位值 使用BCD格式。
11:8	MTHUNIT	R/W	0x0	闹钟月数个位值 使用BCD格式。
7:6	保留	RO	0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
5:4	DATETEN	R/W	0x0	闹钟天数十位值 使用BCD格式。
3:0	DATEUNIT	R/W	0x0	闹钟天数个位值 使用BCD格式。

4.13.3.6 RTC_ALMEN - RTC 闹钟使能设置寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
									YREN	MTHEN	DATEEN	WKEN	HREN	MINEN	SECEN

偏移地址：0x0018

位	名称	读/写	默认值	描述
31: 8	保留	RO	0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
7	CENEN	R/W	0	世纪闹钟功能使能 0：世纪闹钟功能禁止。 1：世纪闹钟功能使能。
6	YREN	R/W	0	年份闹钟功能使能 0：年份闹钟功能禁止。 1：年份闹钟功能使能。
5	MTHEN	R/W	0	月份闹钟功能使能 0：月份闹钟功能禁止。 1：月份闹钟功能使能。
4	DATEEN	R/W	0	日期闹钟功能使能 0：日期闹钟功能禁止。 1：日期闹钟功能使能。
3	WKEN	R/W	0	星期闹钟功能使能 0：星期闹钟功能禁止。 1：星期闹钟功能使能。
2	HREN	R/W	0	小时闹钟功能使能 0：小时闹钟功能禁止。 1：小时闹钟功能使能。
1	MINEN	R/W	0	分钟闹钟功能使能 0：分钟闹钟功能禁止。 1：分钟闹钟功能使能。
0	SECEN	R/W	0	秒钟闹钟功能使能 0：秒钟闹钟功能禁止。 1：秒钟闹钟功能使能。

4.13.3.7 RTC_CALIB - RTC 校准设置寄存器

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

偏移地址：0x001C

位	名称	读/写	默认值	描述
31: 2	保留	RO	0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
1	CYCSEL	R/W	0	校准周期选择 0：当CALIBEN为1时，校准周期为每60秒一次。 1：当CALIBEN为1时，校准周期为每秒一次。
0	CALIBEN	R/W	0	校准功能使能 0：RTC校准禁止。 1：RTC校准使能。

4.13.3.8 RTC_TRIG - RTC 触发选择寄存器

RTC_TRIG 寄存器被用来使能触发 ADC 的相应位。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 R/W
															WKUPTM
15 R/W	14 R/W	13 R/W	12 R/W	11 R/W	10 R/W	9 R/W	8 R/W	7 R/W	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 RO	0 R/W
1HZFLAG	RYR	RMTH	RDATE	RWK	RHR	RMIN	RSEC	AALL	AYR	AMTH	ADATE	AWK	AHR	AMIN	ASEC

偏移地址：0x0020

位	名称	读/写	默认值	描述
31: 18	保留	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
17	1HZFLAG	R/W	0	RTC 1 Hz 触发标志 此位與RTC_CLK信號有關。 1: 對RTC_CLK 輸出產生的ADC觸發致能。對於每一秒，RTC都會產生一個ADC觸發。
16	RCEN	R/W	0	滚动世纪使能 1：世纪滚动触发ADC使能。
15	RYR	R/W	0	滚动年份使能 1：年份滚动触发ADC使能。
14	RMTH	R/W	0	滚动月份使能 1：月份滚动触发ADC使能。
13	RDATE	R/W	0	滚动日期使能 1：日期滚动触发ADC使能。

位	名称	读/写	默认值	描述
12	<i>RWK</i>	R/W	0	滚动星期使能 1：星期滚动触发ADC使能。
11	<i>RHR</i>	R/W	0	滚动小时使能 1：小时滚动触发ADC使能。
10	<i>RMIN</i>	R/W	0	滚动分钟使能 1：分钟滚动触发ADC使能。
9	<i>RSEC</i>	R/W	0	滚动秒钟使能 1：秒钟滚动触发ADC使能。
8	<i>AALL</i>	R/W	0	所有闹钟使能 1：当RTC_TIME和RTC_CAL寄存器与RTC_ARTIME和RTC_ARCAL寄存器中被设置的值相互匹配时，产生ADC触发。
7	<i>ACEN</i>	R/W	0	世纪闹钟使能 1：当世纪闹钟的值相互匹配时，产生ADC触发
6	<i>AYR</i>	R/W	0	年份闹钟使能 1：当年份闹钟的值相互匹配时，产生ADC触发。
5	<i>AMTH</i>	R/W	0	月份闹钟使能 1：当月份闹钟的值相互匹配时，产生ADC触发。
4	<i>ADATE</i>	R/W	0	日期闹钟使能 1：当日期闹钟的值相互匹配时，产生ADC触发。
3	<i>AWK</i>	R/W	0	星期闹钟使能 1：当星期闹钟的值相互匹配时，产生ADC触发。
2	<i>AHR</i>	R/W	0	小时闹钟使能 1：当小时闹钟的值相互匹配时，产生ADC触发。
1	<i>AMIN</i>	R/W	0	分钟闹钟使能 1：当分钟闹钟的值相互匹配时，产生ADC触发。
0	<i>ASEC</i>	R/W	0	秒钟闹钟使能 1：当秒钟闹钟的值相互匹配时，产生ADC触发。

4.13.3.9 RTC_IER - RTC 中断使能寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 WO
															WKUPTME
15 WO	14 WO	13 WO	12 WO	11 WO	10 WO	9 WO	8 WO	7 WO	6 WO	5 WO	4 WO	3 WO	2 WO	1 WO	0 WO
1HZFLAGIE	ROLYRIE	ROLMTHIE	ROLDATEIE	ROLWKIE	ROLHRIE	ROLMINIE	ROLSECIE	AALLIE	AYRIE	AMTHIE	ADATEIE	AWKIE	AHRIE	AMINIE	ASECIE

偏移地址：0x0024

位	名称	读/写	默认值	描述
31: 18	保留	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
17	1HZFLAGIE	WO	0	RTC 1Hz 触发标志中断使能 此位與RTC_CLK信號有關。 1:每一秒產生的中断使能。
16	ROLCENIE	WO	0	滚动世纪中断使能 1: 世纪滚动中断使能。
15	ROLYRIE	WO	0	滚动年份中断使能 1:年份滚动中断使能。
14	ROLMTHIE	WO	0	滚动月份中断使能 1:月份滚动中断使能。
13	ROLDATEIE	WO	0	滚动日期中断使能 1:日期滚动中断使能。
12	ROLWKIE	WO	0	滚动星期中断使能 1:星期滚动中断使能。
11	ROLHRIE	WO	0	滚动小时中断使能 1:小时滚动中断使能。
10	ROLMINIE	WO	0	滚动分钟中断使能 1:分钟滚动中断使能。
9	ROLSECIE	WO	0	滚动秒钟中断使能 1:秒钟滚动中断使能。
8	AALLIE	WO	0	所有闹钟中断使能 1：当RTC_TIME和RTC_CAL寄存器与RTC_ARTIME和RTC_ARCAL寄存器中被设置的值相互匹配时，该中断使能。
7	ACENIE	WO	0	世纪闹钟中断使能 1：当世纪闹钟的值相互匹配时，该中断使能。
6	AYRIE	WO	0	年份闹钟中断使能 1：当年份闹钟的值相互匹配时，该中断使能。
5	AMTHIE	WO	0	月份闹钟中断使能 1：当月份闹钟的值相互匹配时，该中断使能。
4	ADATEIE	WO	0	日期闹钟中断使能 1：当日期闹钟的值相互匹配时，该中断使能。
3	AWKIE	WO	0	星期闹钟中断使能 1：当星期闹钟的值相互匹配时，该中断使能。



位	名称	读/写	默认值	描述
2	AHRIE	WO	0	小时闹钟中断使能 1：当小时闹钟的值相互匹配时，该中断使能。
1	AMINIE	WO	0	分钟闹钟中断使能 1：当分钟闹钟的值相互匹配时，该中断使能。
0	ASECIE	WO	0	秒钟闹钟中断使能 1：当秒钟闹钟的值相互匹配时，该中断使能。

4.13.3.10 RTC_IDR - RTC 中断禁止寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 WO
															WKUPTMD
15 WO	14 WO	13 WO	12 WO	11 WO	10 WO	9 WO	8 WO	7 WO	6 WO	5 WO	4 WO	3 WO	2 WO	1 WO	0 WO
1HZFLAGID	ROLYRID	ROLMTHID	ROLDATEID	ROLWKID	ROLHRID	ROLMINID	ROLSECID	AALLID	AYRID	AMTHID	ADATEID	AWKID	AHRID	AMINID	ASECID

偏移地址：0x0028

位	名称	读/写	默认值	描述
31: 18	保留	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
17	1HZFLAGID	WO	0	RTC 1Hz 触发标志中断禁止 此位與RTC_CLK信號有關。 1:每一秒產生的中断禁止。
16	ROLCENID	WO	0	滚动世纪中断禁止 1: 世纪滚动中断禁止。
15	ROLYRID	WO	0	滚动年份中断禁止 1:年份滚动中断禁止。
14	ROLMTHID	WO	0	滚动月份中断禁止 1:月份滚动中断禁止。
13	ROLDATEID	WO	0	滚动日期中断禁止 1:日期滚动中断禁止。
12	ROLWKID	WO	0	滚动星期中断禁止 1:星期滚动中断禁止。
11	ROLHRID	WO	0	滚动小时中断禁止 1:小时滚动中断禁止。
10	ROLMINID	WO	0	滚动分钟中断禁止 1:分钟滚动中断禁止。
9	ROLSECID	WO	0	滚动秒钟中断禁止 1:秒钟滚动中断禁止。
8	AALLID	WO	0	所有闹钟中断禁止 1：当RTC_TIME和RTC_CAL寄存器与RTC_ARTIME和RTC_ARCAL寄存器中被设置的值相互匹配时，该中断禁止。
7	ACENID	WO	0	世纪闹钟中断禁止 1：当世纪闹钟的值相互匹配时，该中断禁止。

位	名称	读/写	默认值	描述
6	AYRID	WO	0	年份闹钟中断禁止 1：当年份闹钟的值相互匹配时，该中断禁止。
5	AMTHID	WO	0	月份闹钟中断禁止 1：当月份闹钟的值相互匹配时，该中断禁止。
4	ADATEID	WO	0	日期闹钟中断禁止 1：当日期闹钟的值相互匹配时，该中断禁止。
3	AWKID	WO	0	星期闹钟中断禁止 1：当星期闹钟的值相互匹配时，该中断禁止。
2	AHRID	WO	0	小时闹钟中断禁止 1：当小时闹钟的值相互匹配时，该中断禁止。
1	AMINID	WO	0	分钟闹钟中断禁止 1：当分钟闹钟的值相互匹配时，该中断禁止。
0	ASECID	WO	0	秒钟闹钟中断禁止 1：当秒钟闹钟的值相互匹配时，该中断禁止。



4.13.3.11 RTC_IMR - RTC 中断屏蔽寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
															WKUPTMIM
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
1HZFLAGIM	ROLYRIM	ROLMTHIM	ROLDATEIM	ROLWKIM	ROLHRIM	ROLMINIM	ROLSECIM	AALLIM	AYRIM	AMTHIM	ADATEIM	AWKIM	AHRIM	AMINIM	ASECIM

偏移地址：0x002C

位	名称	读/写	默认值	描述
31: 18	保留	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
17	1HZFLAGIM	RO	0	RTC 1Hz触发标志中断屏蔽状态 此位與RTC_CLK信號有關。 0：该中断将会被屏蔽。 1：中断不会被屏蔽。
16	RCENIM	RO	0	滚动世纪中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
15	ROLYRIM	RO	0	滚动年份中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
14	ROLMTHIM	RO	0	滚动月份中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
13	ROLDATEIM	RO	0	滚动日期中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
12	ROLWKIM	RO	0	滚动星期中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
11	ROLHRIM	RO	0	滚动小时中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
10	ROLMINIM	RO	0	滚动分钟中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
9	ROLSECIM	RO	0	滚动秒钟中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
8	AALLIM	RO	0	所有闹钟中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
7	ACENIM	RO	0	世纪闹钟中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。

位	名称	读/写	默认值	描述
6	AYRIM	RO	0	年份闹钟中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
5	AMTHIM	RO	0	月份闹钟中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
4	ADATEIM	RO	0	日期闹钟中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
3	AWKID	RO	0	星期闹钟中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
2	AHRID	RO	0	小时闹钟中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
1	AMINID	RO	0	分钟闹钟中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
0	ASECID	RO	0	秒钟闹钟中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。

4.13.3.12 RTC_RIS - RTC 原始中断状态寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
															WKUPTMRI
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
1HZFLAGRI	ROLYRRI	ROLMTHRI	ROLDATERI	ROLWKRI	ROLHRRI	ROLMINRI	ROLSECR	AALLRI	AYRRI	AMTHRI	ADATERI	AWKRI	AHRRI	AMINRI	ASECRI

偏移地址：0x0030

位	名称	读/写	默认值	描述
31: 18	保留	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
17	1HZFLAGRI	RO	0	RTC 1Hz触发标志原始中断状态 此位與RTC_CLK信號有關。 0：没有中断产生。 1：该中断正在发出请求。
16	ROLCENRI	RO	0	滚动世纪原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
15	ROLYRRI	RO	0	滚动年份原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
14	ROLMTHRI	RO	0	滚动月份原始中断状态 0：没有中断产生。 1:该中断正在发出请求。
13	ROLDATERI	RO	0	滚动日期原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
12	ROLWKRI	RO	0	滚动星期原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
11	ROLHRRI	RO	0	滚动小时原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
10	ROLMINRI	RO	0	滚动分钟原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
9	ROLSECR	RO	0	滚动秒钟原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
8	AALLRI	RO	0	所有闹钟原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
7	ACENRI	RO	0	世纪闹钟原始中断状态 0：没有中断产生。 1：该中断正在发出请求。

位	名称	读/写	默认值	描述
6	AYRRI	RO	0	年份闹钟原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
5	AMTHRI	RO	0	月份闹钟原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
4	ADATERI	RO	0	日期闹钟原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
3	AWKRI	RO	0	星期闹钟原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
2	AHRRI	RO	0	小时闹钟原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
1	AMINRI	RO	0	分钟闹钟原始中断状态 0：没有中断产生。 1：该中断正在发出请求。
0	ASECRI	RO	0	秒钟闹钟原始中断状态 0：没有中断产生。 1：该中断正在发出请求。

4.13.3.13 RTC_ISC- RTC 中断状态和清除寄存器

注：该寄存器是读和写清除寄存器。写‘1’到相应的位将会清除该中断的状态。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 R/W1C
															WKUPTMS
15 R/W1C	14 R/W1C	13 R/W1C	12 R/W1C	11 R/W1C	10 R/W1C	9 R/W1C	8 R/W1C	7 R/W1C	6 R/W1C	5 R/W1C	4 R/W1C	3 R/W1C	2 R/W1C	1 R/W1C	0 R/W1C
1HZFLAGIS	ROLYRIS	ROLMTHIS	ROLDATEIS	ROLWKIS	ROLHRIS	ROLMINIS	ROLSECIS	AALLIS	AYRIS	AMTHIS	ADATEIS	AWKIS	AHRIS	AMINIS	ASECIS

偏移地址：0x0034

位	名称	读/写	默认值	描述
31: 18	保留	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
17	1HZFLAGIS	R/W1C	0	RTC 1Hz触发标志中断状态和清除 0：没有中断产生或被屏蔽。 1:该中断已经发出请求。
16	ROLCENIS	R/W1C	0	滚动世纪中断状态和清除 0：没有中断产生或被屏蔽。 1:该中断已经发出请求。
15	RYRIS	R/W1C	0	滚动年份中断状态和清除 0：没有中断产生或被屏蔽。 1:该中断已经发出请求。
14	RMTHIS	R/W1C	0	滚动月份中断状态和清除 0：没有中断产生或被屏蔽。 1:该中断已经发出请求。
13	RDATEIS	R/W1C	0	滚动日期中断状态和清除 0：没有中断产生或被屏蔽。 1:该中断已经发出请求。
12	RWKIS	R/W1C	0	滚动星期中断状态和清除 0：没有中断产生或被屏蔽。 1:该中断已经发出请求。
11	RHRIS	R/W1C	0	滚动小时中断状态和清除 0：没有中断产生或被屏蔽。 1:该中断已经发出请求。
10	RMINIS	R/W1C	0	滚动分钟中断状态和清除 0：没有中断产生或被屏蔽。 1:该中断已经发出请求。
9	RSECIS	R/W1C	0	滚动秒钟中断状态和清除 0：没有中断产生或被屏蔽。 1:该中断已经发出请求。
8	AALLIS	R/W1C	0	所有闹钟中断状态和清除 0：没有中断产生或被屏蔽。 1:该中断已经发出请求。
7	ACENIS	R/W1C	0	世纪闹钟中断状态和清除 0：没有中断产生或被屏蔽。 1:该中断已经发出请求。

位	名称	读/写	默认值	描述
6	AYRIS	RW1C	0	年份闹钟中断状态和清除 0：没有中断产生或被屏蔽。 1:该中断已经发出请求。
5	AMTHIS	RW1C	0	月份闹钟中断状态和清除 0：没有中断产生或被屏蔽。 1:该中断已经发出请求。
4	ADATEIS	RW1C	0	日期闹钟中断状态和清除 0：没有中断产生或被屏蔽。 1:该中断已经发出请求。
3	AWKIS	RW1C	0	星期闹钟中断状态和清除 0：没有中断产生或被屏蔽。 1:该中断已经发出请求。
2	AHRIS	RW1C	0	小时闹钟中断状态和清除 0：没有中断产生或被屏蔽。 1:该中断已经发出请求。
1	AMINIS	RW1C	0	分钟闹钟中断状态和清除 0：没有中断产生或被屏蔽。 1:该中断已经发出请求。
0	ASECIS	RW1C	0	秒钟闹钟中断状态和清除 0：没有中断产生或被屏蔽。 1:该中断已经发出请求。

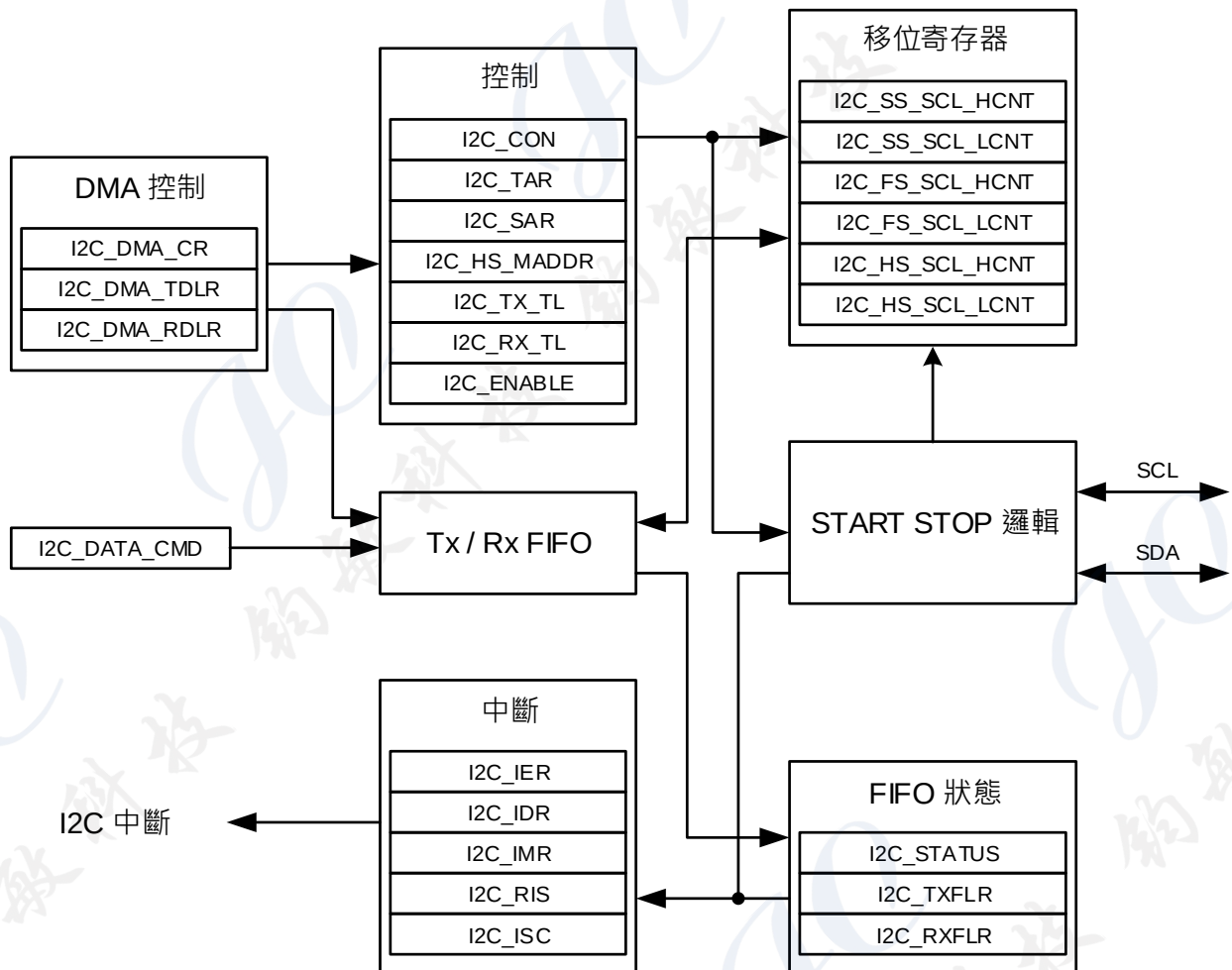
4.14 集成电路总线 (I2C)

I2C 是两线双向串行通信总线，提供了一种简单而有效的方法来实现设备之间的数据交换。I2C 标准是一个多主机总线包括冲突检测和仲裁，如果两个或两个以上的主机试图同时控制总线时，其中仲裁可以防止数据损坏。数据在主机和从机之间被同步到串行时钟(SCK)然后在串行数据(SDA)线上逐字节地被传输。每个数据字节是 8 位的。

- 两线式串行通信接口。
- 支持不同的通信速度：
 - 标准模式：100Kb/s
 - 快速模式：400Kb/s
 - 高速模式：3.4Mb/s
- 主/从设备可配置。
- 支持重发功能。
- 支持多主机操作。
- 支持 7 位或 10 位的地址。
- 支持 7 位或 10 位的传输格式。
- 支持广播呼叫。

4.14.1 块图

图 4.14-1: I2C 块图



4.14.2 功能描述

4.14.2.1 I2C 總線協定

I2C 开始(START)和停止(STOP)条件协议

I2C 规范将 SCK 线位高电平时，SDA 线从高电平到低电平状态的过渡被定义为一个起始标志。一个起始标志必须由主机产生，它表示总线从空闲状态过渡到活动状态。每一位数据对应一个时钟脉冲，并且数据从最高位开始发送。当 SCK 处在高电平时，在 SDA 线上的一个变化被认为是一个命令（START 或 STOP）。每一位在 SCK 的高电平期间被采样；因此，SDA 线的状态仅能在 SCK 的低电平期间被改变，并且必须在 SCK 的高电平期间保持稳定。

当总线处于空闲状态时 SCK 和 SDA 信号通过外部上拉电阻都被拉为高电平。当主机需要在总线上开始一个传输时，它需要发起一个 START 条件，即在 SCL 为 1 时 SDA 信号由高电平向低电平切换。当主机需要终止一个传输时，它需要发出一个 STOP 条件，即在 SCL 为 1 时 SDA 信号由低电平向高电平切换。下图为 START 和 STOP 条件的时序图。当数据开始传输时，SDA 线必须在 SCK 为高电平期间保持稳定。

图 4.14-2: 启动和停止条件



I2C 从机地址协议

I2C 有两个地址格式：7 位地址格式和 10 位的地址格式。在 7 位地址格式中，第一个字节的头 7 位组成了从机地址，最低位（LSB）是第 8 位（R/W），它决定了报文的方向。第一字节的最低位是 0，表示主机会写数据到被选中的从机，为 1 表示主机会向从机读数据。数据传输是从高位（MSB）开始。在 10 位地址格式中，用两个字节表示从机地址。第一个字节的传输包含以下位的定义。第一个字节的高 5 位（位 7:3）告诉从机这是一个 10 位地址的传输，紧接着的两位（位 2:1）表示从机地址的高两位（addr9:8），最低位（位 8）即是读写位（R/W）。第二个字节是从机地址的低 8 位（addr7:0）。

图 4.14-3: 7 位地址格式

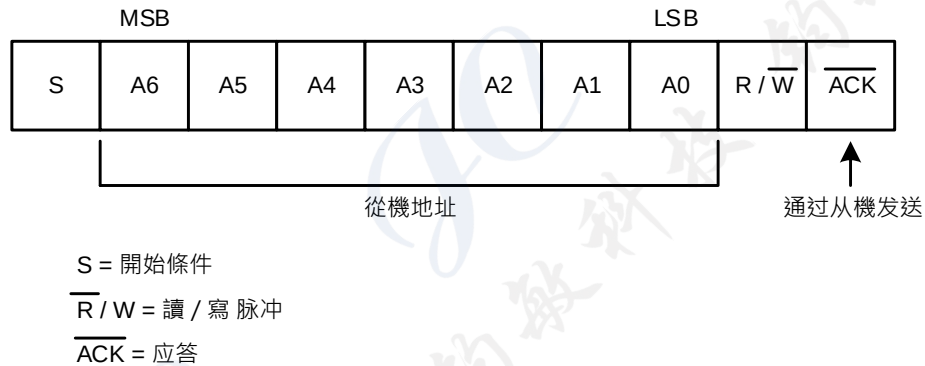


图 4.14-4: 10 位地址格式

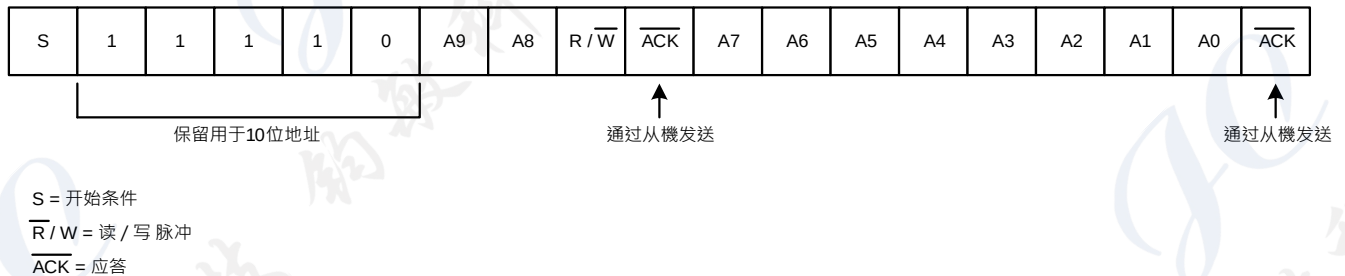


表 4.14-1: 第一个字节的定义

从机地址	读/写位	描述
0000 000	0	通用广播呼叫地址
0000 000	1	START字节
0000 001	X	CBUS 地址
1111 0XX	X	10位从机地址

I2C 发送和接收协议

所有数据都以字节传输，每一次传输的字节数没有限制。主机发送地址和读/写位或每发一个字节数据给从机后，从机必须返回一个 **ACK** 响应给主机。当从机没有发 **ACK** 响应主机时，主机必须发一个 **STOP** 条件终止传输，此时从机必须释放 **SDA** 线以便主机能发 **STOP** 条件。

如果主机的发送器正在发送数据，每一字节数据接收到后从机接收器需要发送一个 **ACK** 给主机的发送器。

对于7位地址

S	从機地址	R / $\overline{W}$	A	数据	A	数据	A / $\overline{A}$	P
---	------	--------------------	---	----	---	----	--------------------	---

↑
“0” (写)

对于10位地址

S	从機地址首7位	R / $\overline{W}$	A	从機地址 第二个字节	A	数据	A / $\overline{A}$	P
---	---------	--------------------	---	---------------	---	----	--------------------	---

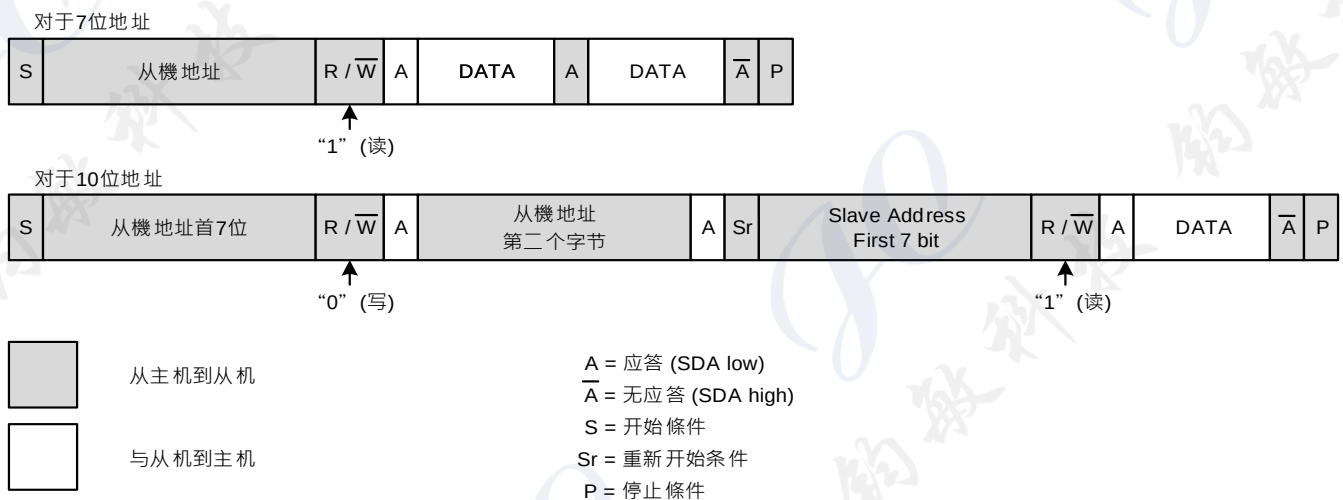
↑
“0” (写)

从主机到从机

与从机到主机

A = 应答 (SDA low)
 $\overline{A}$ = 无应答 (SDA high)
 S = 开始条件
 P = 停止条件

图 4.14-6: 主机接收协议



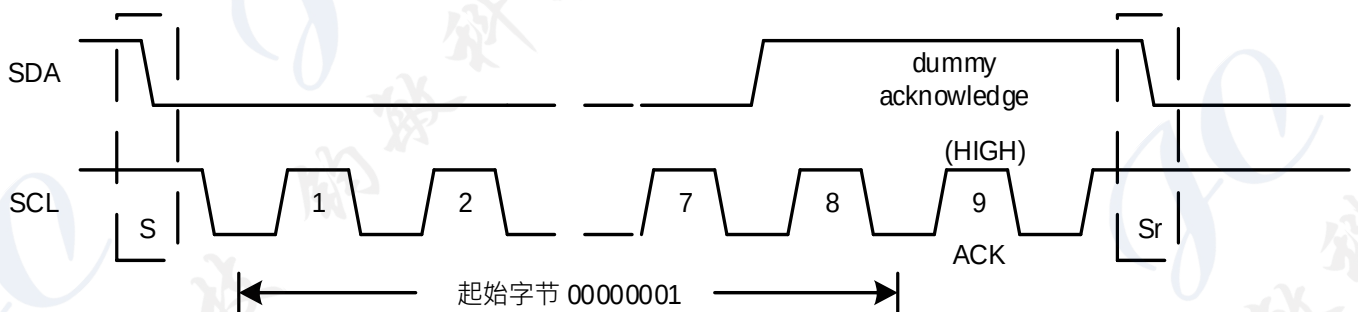
I2C START BYTE 传输协议

START BYTE 传输协议是为没有片上硬件 I2C 模块的系统而建立。当 I2C 被确定为一个从机时，它总是在被支持的最高速度下采样 I2C 总线，因此它不需要一个 START BYTE 传输。然而，当 I2C 被用作一个主机时，一旦有从机设备请求它，它可以在每一个传输的开始产生 START BYTE 传输。START BYTE 协议由“00000001”组成，如下图所示。这使得正在轮询总线的处理器可以欠采样地址阶段直到检测到 0。一旦 微控制器检测到 0，它将切换欠采样率到正确的主机采样率。

START BYTE 过程如下：

1. 主机产生一个 START 条件。
2. 主机发送 START 字节（0001 0000）。
3. 主机发送 ACK 时钟脉冲。
4. 没有从机返回 ACK。
5. 主机重新发送 START（Sr）条件。

图 4.14-7: 起始字节传输



I2C 传输速度算法介绍

在 I2C 传输之前，SCK 时钟占空比周期必须通过下面的寄存器被设置。

- 标准模式（100Kbps）:SCK 时钟的最小高电平时间为 4000ns,最小低电平时间为 4700ns。

SCK 的波形周期通过下面的寄存器被设置：

I2C_SS_SCK_HCNT

I2C_SS_SCK_LCNT

- 快速模式（400Kbps）:SCK 时钟的最小高电平时间为 600ns,最小低电平时间为 1300ns。

SCK 的波形周期通过下面的寄存器被设置：

I2C_FS_SCK_HCNT

I2C_FS_SCK_LCNT

- 高速模式（3.4Mbps）：

负载低于 100pF 时: SCK 时钟的最小高电平时间为 60ns,最小低电平时间为 160ns。

负载低于 400pF 时: SCK 时钟的最小高电平时间为 120ns,最小低电平时间为 320ns。

SCK 的波形周期通过下面的寄存器被设置：

I2C_HS_SCK_HCNT

I2C_HS_SCK_LCNT

注意：当使用高速模式时，由于传送主机编号是使用的快速模式，因此快速模式下的寄存器(I2C_FS_SCK_HCNT & I2C_FS_SCK_LCNT)也需要被设置。

算法如下：

例如：在系统时钟为 16MHz 时，I2C 被确定为快速模式。

SCK 时钟的高电平时间：

$600\text{ns} = 62.5\text{ns}(16\text{MHz}) * \text{I2C_FS_SCK_HCNT} \Rightarrow \text{I2C_FS_SCK_HCNT}$ 设为 10(0xA)。

SCK 时钟的低电平时间：

$1300\text{ns} = 62.5\text{ns}(16\text{MHz}) * \text{I2C_FS_SCK_LCNT} \Rightarrow \text{I2C_FS_SCK_LCNT}$ 设为 21(0x15)。

4.14.2.2 中斷

I2C 的中斷可以通过 5 个寄存器来控制。

- 中斷控制 (IER, IDR, IMR)

I2C 中斷使能寄存器(I2C_IER)通过简单地写‘1’去使能中斷请求线。类似地，I2C 中斷屏蔽寄存器(I2C_IDR)通过简单地写‘1’去禁止中斷请求线。IER 和 IDR 是只写寄存器，上述寄存器的总体结果可以通过 WDT 中斷屏蔽寄存器(I2C_IMR)被显示。IMR 是一个只读寄存器，使用“1”或“0”表示是否中斷请求线被使能还是被禁止。

- 中斷状态读(RIS)

I2C 原始中斷状态(I2C_RIS)是一个只读寄存器，可以从该模块中读取中斷状态。在觀察到以下情况時，I2C 可以產生中斷：

- 發生超時
- 廣播呼叫地址已收到和被應答
- 总线检测到開始條件
- 总线检测到停止條件
- 捕獲到 I2C 活動
- 从机传输
- 發送中止
- 从机数据请求
- 發送緩衝區空
- 發送緩衝區溢出
- 接收緩衝滿
- 接收緩衝區溢出
- 接收緩衝器下溢

- 中斷清除(ISC)

I2C 中斷状态&中斷清除寄存器(I2C_ISC)是用来读取该模块中被屏蔽的中斷，显示哪一个中斷未被屏蔽。在 ISC 中的每一位是 RIS 和 IMR 中的相对应位的逻辑与。对该寄存器中的某一位写入“1”可以清除相对应的中斷。

4.14.2.3 操作模式

I2C 作為主機或從機，可以操作在四種不同的模式：從發送，從接收，主發送器或主接收器以支持數據傳輸。主機啟動數據傳輸並產生時鐘信號 SCL。主機尋址的任何設備被認為是一個從機。本節介紹這些模式和他們的編程模型：

注意：I2C 應設置僅作為 I2C 主站或 I2C 從器件進行操作，永遠不會設置兩者同時進行。

从器件模式操作

初始配置

要将 I2C 控制器作为从器件，请执行以下步骤：

1. 通过将 0 写入 I2C_ENABLE.ENABLE 来禁用 I2C 控制器。
2. 写入 I2C_SAR 来设置从器件地址。这是 I2C 控制器做出响应的地址。
3. 写入 I2C_CTRL 寄存器以指定哪种类型的寻址被支持(7- 或 10-位)。以及使能仅作为从器件模式中的 I2C 控制器。
4. 通过将 1 写入 I2C_ENABLE.ENABLE 来使能 I2C 控制器。

从器件 - 发送器操作

在這種模式下，I2C 是從器件发送数据到主机。當從主器件收到的地址與从器件地址它自己的地址是相同，而且 $R/\overline{W} = 1$ 時，便會進入此模式。从发送器与由主装置产生的时钟脉冲移位串行数据到 SDA 輸出。从器件不产生时钟。

当 I2C 是作为从发送器，会出现以下步骤：

1. 另一个 I2C 主器件使用一个地址(匹配 I2C 控制器的 I2C_SAR 寄存器中的从器件地址)发起 I2C 传输。
2. I2C 控制器确认发送地址并且识别传输的方向以便表示它作为从器件- 发送器。
3. I2C 控制器置位 RDREQRI 中断(I2C_RIS 寄存器) 并且等待软件响应。
4. 如果接收读请求之前，TX FIFO 中有剩余的数据，那么 I2C 控制器置位一个 TXABRTRI 中断(I2C_RIS 寄存器) 以便从 TX FIFO 中刷新旧数据。
5. 软件将要写入的数据写入 I2C_DATA_CMD.DAT 位并且将 0 写入 I2C_DATA_CMD.CMD。
6. 软件在继续操作之前必须清除 RDREQRI 和 TXABRTRI 中断，通过写 1 到相应的位—在 I2C_ISC 寄存器的 RDREQIS 和 TXABRTIS。
7. I2C 控制器发送字节。
8. 主器件可能通过发出 RESTART 条件保持 I2C 总线或通过发出 STOP 条件释放总线。

从器件 接收器操作

在這種模式下，I2C 是從器件发送数据到主机。當從主器件收到的地址與从器件地址它自己的地址是相同，而且 $R/\overline{W} = 0$ 時，便會進入此模式。从发送器移位從 SDA 輸入的串行数据根據由主装置产生的时钟脉冲。从器件不产生时钟。I2C 控制器作为一个从器件- 接收器并且会执行以下步骤：

1. 另一个 I2C 主器件使用一个地址(匹配 I2C_SAR 寄存器中的 I2C 控制器的从地址) 发起 I2C 传输。
2. I2C 控制器确认发送的地址并且识别传输的方向以表明 I2C 控制器作为一个从器件- 接收器。
3. I2C 控制器接收到发送的字节并且将其放入接收缓冲器。
4. 状态和中断相应的接收缓冲区位被更新。
5. 软件必须读取来自 I2C_DATA_CMD 寄存器中的 DAT 域。
6. 主器件可能通过发出 RESTART 条件保持 I2C 总线或通过发出 STOP 条件释放总线。

主器件模式操作

初始配置

要将 I2C 控制器作为主器件，请执行以下步骤：

1. 通过将 0 写入 I2C_ENABLE.ENABLE 来禁用 I2C 控制器。
2. 写入 I2C_CTRL 寄存器以便设置从器件操作的最大速度模式并指定当器件是从器件时，I2C 控制器是否在 7/10 位寻址模式中开始传输。
3. 将 I2C 器件被寻址的地址写入 I2C_TAR 寄存器。它也会表明 General Call 或 START BYTE 命令是否将会由 I2C 执行。
4. 对 IC_HS_MADDR 寄存器写入所需的主码。请注意，这一步只适用于高速模式传输。
5. 通过将 1 写入 I2C_ENABLE.ENABLE 来使能 I2C 控制器。
6. 现在将要发送的传输方向和数据写入 I2C_DATA_CMD 寄存器。如果 I2C_DATA_CMD 寄存器在使能 I2C 控制器之前被写入，那么使能 I2C 控制器时，由于缓冲器被清除，数据和命令丢失。

主器件发送和主器件接收

I2C 控制器支持动态地读和写之间的来回切换。要发送数据，将要写的数据写入 I2C_DATA_CMD.DAT 字节。对于 I2C 写操作，I2C_DATA_CMD.CMD 位应该被置 0。随后，读命令可能通过将 "don't cares" 写入 I2C_DATA_CMD 寄存器的 DAT 而被发出，并且 CMD 位应该置 1。

4.14.3 I2C 寄存器映射

基地址: 0x4800_8000

偏移地址	寄存器名称	读/写	默认值	描述	参考页
0x0000	I2C_CTRL	R/W	0x0000_0025	I2C控制寄存器	265
0x0004	I2C_TAR	R/W	0x0000_0055	I2C目标地址寄存器	266
0x0008	I2C_SAR	R/W	0x0000_0055	I2C从机地址寄存器	266
0x000C	I2C_HS_MADDR	R/W	0x0000_0001	I2C主机编号地址寄存器	267
0x0010	I2C_DATA_CMD	R/W	0x0000_0000	I2C Tx/Rx数据寄存器	267
0x0014	I2C_SS_SCK_HCNT	R/W	0x0000_0040	I2C标准模式SCK高电平寄存器	268
0x0018	I2C_SS_SCK_LCNT	R/W	0x0000_004B	I2C标准模式SCK低电平寄存器	268
0x001C	I2C_FS_SCK_HCNT	R/W	0x0000_000A	I2C快速模式SCK高电平寄存器	269
0x0020	I2C_FS_SCK_LCNT	R/W	0x0000_0015	I2C快速模式SCK低电平寄存器	269
0x0024	I2C_HS_SCK_HCNT	R/W	0x0000_0001	I2C高速模式SCK高电平寄存器	270
0x0028	I2C_HS_SCK_LCNT	R/W	0x0000_0007	I2C高速模式SCK低电平寄存器	270
0x002C	I2C_IER	WO	0x0000_0000	I2C中断使能寄存器	271
0x0030	I2C_IDR	WO	0x0000_0000	I2C中断禁止寄存器	272
0x0034	I2C_IMR	RO	0x0000_0000	I2C中断屏蔽状态寄存器	273
0x0038	I2C_RIS	RO	0x0000_0000	I2C原始中断状态寄存器	274
0x003C	I2C_ISC	R/W1C	0x0000_0000	I2C中断状态和清除寄存器	276
0x0040	I2C_RX_TL	R/W	0x0000_0000	I2C接收FIFO阈值寄存器	277
0x0044	I2C_TX_TL	R/W	0x0000_0000	I2C发送FIFO阈值寄存器	277
0x006C	I2C_ENABLE	R/W	0x0000_0000	I2C使能寄存器	278
0x0070	I2C_STATUS	RO	0x0000_0006	I2C状态寄存器	278
0x0074	I2C_TXFLR	RO	0x0000_0000	I2C发送FIFO深度寄存器	279
0x0078	I2C_RXFLR	RO	0x0000_0000	I2C接收FIFO深度寄存器	279
0x007C	I2C_TIMEOUT	R/W	0x0000_0000	I2C超时使能寄存器	280
0x0080	I2C_TX_ABRT	R	0x0000_0000	I2C发送终止状态寄存器	280

4.14.3.1 I2C_CTRL - I2C 控制寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
									SLAVE	RESTART	10ADDRM	10ADDRS	SPEED	MASTER	

偏移地址：0x0000

位	名称	读/写	默认值	描述
31: 7	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
6	SLAVE	R/W	0	I2C从机禁止 此位控制复位后I2C是否作为从机。 0：从机使能。 1：从机禁止。
5	RESTART	R/W	1	I2C重发功能使能 这位确定当它作为主机时是否发送重发条件。注意如果它的重发功能被禁止，主机不能执行下面的功能。 - 每个传输（拆分）发送多个字节 - 在一个传输（拆分）内改变方向 - 发送一个起始字节 - 执行任何高速模式操作 - 在10位地址模式下进行组合格式传输 - 只在10位地址下进行读操作 0：重发功能禁止。 1：重发功能使能。
4	10BADDRM	R/W	0	10位地址主机模式 此位控制是否I2C在作为主机时通过10位地址模式开始传输。 0：7位地址模式 1：10位地址模式
3	10BADDRS	R/W	0	10位地址从机模式 此位控制是否I2C在作为从机时通过10位地址模式开始传输。 0：7位地址模式 1：10位地址模式
2:1	SPEED	R/W	0x2	I2C速度选择 0：保留 1：标准模式(100 kbit/s) 2：快速模式(400 kbit/s) 3：高速模式(3.4 Mbit/s)
0	MASTER	R/W	1	I2C主机禁止 这位控制I2C主机模式是否使能。 0：主机模式禁止。 1：主机模式使能。

4.14.3.2 I2C_TAR- I2C 目标地址寄存器

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

偏移地址：0x0004

位	名称	读/写	默认值	描述
31:12	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
11	SPECIAL	R/W	0	特殊命令 这位表时软件是否想要执行一个通用广播呼叫或起始字节（START BYTE）的I2C命令。 0：忽略位10 GCORST并且正常使用I2C_TAR。 1：执行在GCORST位被指定的特殊I2C命令。
10	GCORST	R/W	0	通用广播呼叫或起始字节（START BYTE）命令 如果SPECIAL设为1时，则该位指示当在发出广播呼叫后，由I2C或广播呼叫地址进行广播呼叫或START命令字节，只有写操作可被执行。I2C保持在广播呼叫模式，直至SPECIAL位被清除。 0：通用广播呼叫地址 1：START BYTE
9:0	TADDR	R/W	0x55	I2C目标地址寄存器 任何一个主机传输时，此位是目标的地址。

4.14.3.3 I2C_SAR- I2C 从机地址寄存器

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

偏移地址：0x0008

位	名称	读/写	默认值	描述
31:10	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
9:0	SADDR	R/W	0x55	I2C从地址寄存器 当I2C被用作一个从机时此域保持该从机的地址。对于7位寻址，只有位[6：0]从机地址寄存器使用。该寄存器只能在I2C接口已禁用时写入

4.14.3.4 I2C_HS_MADDR - I2C 高速主机编号地址寄存器

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

偏移地址：0x000C

位	名称	读/写	默认值	描述
31:3	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
2:0	HSMADDR	R/W	0x1	I2C高速主机编号地址寄存器 此域持有在高速模式下I2C主机编号的值。

4.14.3.5 I2C_DATA_CMD - I2C Tx/Rx 数据和命令寄存器

当 CPU 要填充 TX FIFO 时，写入数据到这个寄存器。读取该寄存器会从 RX FIFO 回送字节。

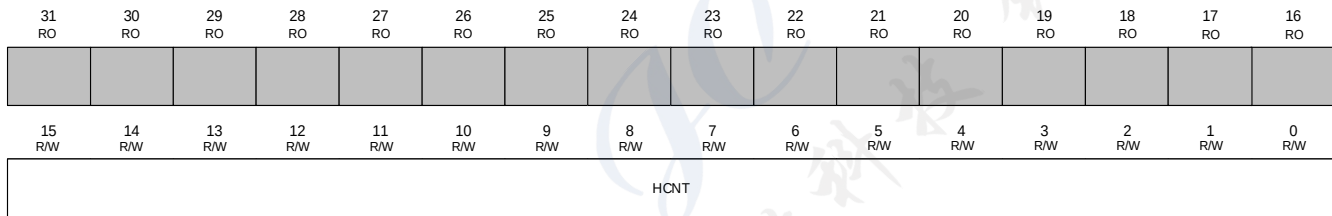
31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 R/W	7 R/W	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
							CMD		DAT						

偏移地址：0x000C

位	名称	读/写	默认值	描述
31:9	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
8	CMD	R/W	0	Rx/Tx 命令 此位控制是否执行一个读或写。该位控制是否进行读或写操作。当I2C作为从机时，该位不控制方向。只有当I2C作为一个主机时，它在可控制方向。 在从机接收模式下，该位是“无关”，因为写入该寄存器不是必需的。在从机发送模式下，一个“0”表示CPU的数据要被传送。 0：主机读。 1：主机写。
7:0	DAT	R/W	0x0	Rx/Tx 数据缓冲区 这个寄存器容纳在I2C总线上需要发送或接收的数据。读这些位将会返回在I2C接口上接收到的数据。写这些位将会把数据从I2C接口发出去。

4.14.3.6 I2C_SS_SCK_HCNT- I2C 标准模式 SCK 高电平寄存器

此寄存器设置标准速度的 SCL 时钟高周期计数。该寄存器必须在任何 I2C 总线交易之前设定，以确保适当的 I/O 时序。

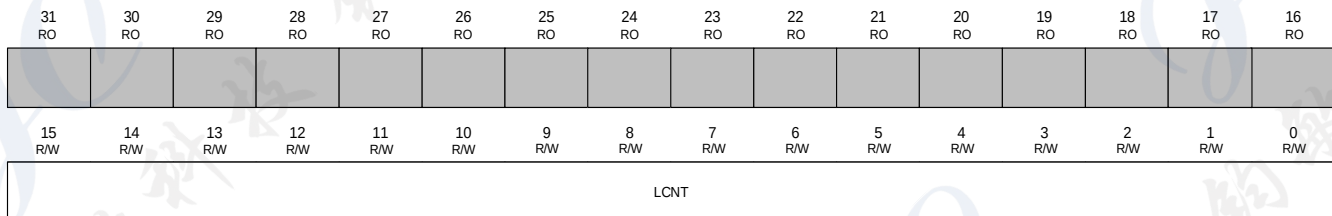


偏移地址：0x0014

位	名称	读/写	默认值	描述
31:16	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:0	HCNT	R/W	0x40	标准速度下I2C时钟SCK高电平计数值 此域是为标准速度模式设置SCK时钟的高电平计数值。该位只能在I2C接口被禁用时写入。写在其他时间没有任何效果。

4.14.3.7 I2C_SS_SCK_LCNT- I2C 标准模式 SCK 低电平寄存器

此寄存器设置标准速度的 SCL 时钟低周期计数。

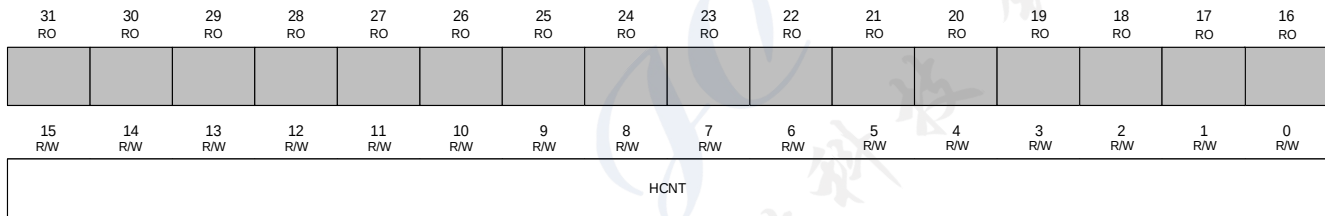


偏移地址：0x0018

位	名称	读/写	默认值	描述
31:16	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:0	LCNT	R/W	0x4B	标准速度下I2C时钟SCK低电平计数值 此域是为标准速度模式设置SCK时钟的低电平计数值。该位只能在I2C接口被禁用时写入。写在其他时间没有任何效果。

4.14.3.8 I2C_FS_SCK_HCNT- I2C 快速模式 SCK 高电平寄存器

此寄存器设置快速 SCL 时钟高周期计数。该寄存器必须在任何 I2C 总线交易之前设定，以确保适当的 I/O 时序。

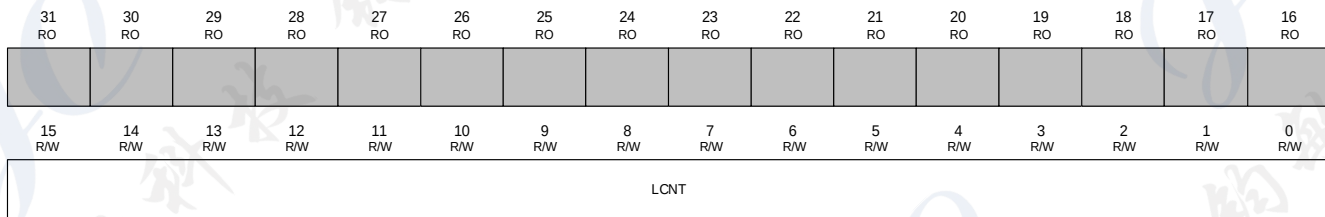


偏移地址：0x001C

位	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:0	<i>HCNT</i>	R/W	0x0A	快速模式下I2C时钟SCK高电平计数值 此域是为快速模式设置SCK时钟的高电平计数值。该位只能在I2C接口被禁用时写入。写在其他时间没有任何效果。

4.14.3.9 I2C_FS_SCK_LCNT- I2C 快速模式 SCK 低电平寄存器

此寄存器设置快速 SCL 时钟低周期计数。



偏移地址：0x0020

位	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:0	<i>LCNT</i>	R/W	0x15	快速模式下I2C时钟SCK低电平计数值 此域是为快速模式设置SCK时钟的低电平计数值。该位只能在I2C接口被禁用时写入。写在其他时间没有任何效果。

4.14.3.10 I2C_HS_SCK_HCNT- I2C 高速模式 SCK 高电平寄存器

此寄存器设置高速 SCL 时钟高周期计数。该寄存器必须在任何 I2C 总线交易之前设定，以确保适当的 I/O 时序。

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

偏移地址：0x0024

位	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:0	<i>HCNT</i>	R/W	0x01	高速模式下I2C时钟SCK高电平计数值 此域是为高速模式设置SCK时钟的高电平计数值。

4.14.3.11 I2C_HS_SCK_LCNT- I2C 高速模式 SCK 低电平寄存器

此寄存器设置高速 SCL 时钟低周期计数。

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

偏移地址：0x0028

位	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:0	<i>LCNT</i>	R/W	0x07	高速模式下I2C时钟SCK低电平计数值 此域是为高速模式设置SCK时钟的低电平计数值。

4.14.3.12 I2C_IER- I2C 中断使能寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 WO	11 WO	10 WO	9 WO	8 WO	7 WO	6 WO	5 WO	4 WO	3 WO	2 WO	1 WO	0 WO
			TOIE	GCIE	STDETIE	STPDETIE	ACTIE	RXDNEIE	TXABRTIE	RDREQIE	TXEPTIE	TXOVIE	RXFULLIE	RXOVERIE	RXUNDIE

偏移地址：0x002C

位	名称	读/写	默认值	描述
31:13	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
12	TOIE	WO	0	超时中断使能 1：该中断使能。
11	GCIE	WO	0	通用广播呼叫中断使能 1：改中断使能。
10	STDETIE	WO	0	START条件检测中断使能 1：该中断使能。
9	STPDETIE	WO	0	STOP条件检测中断使能 1：该中断使能。
8	ACTIE	WO	0	I2C有活动检测中断使能 1：该中断使能。
7	RXDONEIE	WO	0	接收（从机）传输完成中断使能 1：该中断使能。
6	TXABRTIE	WO	0	发送终止中断使能 1：该中断使能。
5	RDREQIE	WO	0	接收（从机）请求数据中断使能 1：该中断使能。
4	TXEPTIE	WO	0	发送缓冲区空中断使能 1：该中断使能。
3	TXOVIE	WO	0	发送溢出中断使能 1：该中断使能。
2	RXFULLIE	WO	0	接收缓冲区满中断使能 1：该中断使能。
1	RXOVERIE	WO	0	接收上溢中断使能 1：该中断使能。
0	RXUNDIE	WO	0	接收下溢中断使能 1：该中断使能。

4.14.3.13 I2C_IDR- I2C 中断禁止寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 WO	11 WO	10 WO	9 WO	8 WO	7 WO	6 WO	5 WO	4 WO	3 WO	2 WO	1 WO	0 WO
			TOID	GCID	STADETID	STPDETID	ACTID	RXDNDID	TXABRTID	RDREQID	TXEPTID	TXOVID	RXFULLID	RXOVERID	RXUNDID

偏移地址：0x0030

位	名称	读/写	默认值	描述
31:13	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
12	TOID	WO	0	超时中断禁止 1：该中断禁止。
11	GCID	WO	0	通用广播呼叫中断禁止 1：改中断禁止。
10	STDETID	WO	0	START条件检测中断禁止 1：该中断禁止。
9	STPDETID	WO	0	STOP条件检测中断禁止 1：该中断禁止。
8	ACTID	WO	0	I2C有活动检测中断禁止 1：该中断禁止。
7	RXDONEID	WO	0	接收（从机）传输完成中断禁止 1：该中断禁止。
6	TXABRTID	WO	0	发送终止中断禁止 1：该中断禁止。
5	RDREQID	WO	0	接收（从机）请求数据中断禁止 1：该中断禁止。
4	TXEPTID	WO	0	发送缓冲区空中断禁止 1：该中断禁止。
3	TXOVID	WO	0	发送溢出中断禁止 1：该中断禁止。
2	RXFULLID	WO	0	接收缓冲区满中断禁止 1：该中断禁止。
1	RXOVERID	WO	0	接收上溢中断禁止 1：该中断禁止。
0	RXUNDID	WO	0	接收下溢中断禁止 1：该中断禁止。

4.14.3.14 I2C_IMR- I2C 中断屏蔽状态寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
			TOIM	GCIM	STADETIM	STPDETIM	ACTIM	RXDNIIM	TXABRTIM	RDREQIM	TXEPTIM	TXOVIM	RXFULLIM	RXOVERIM	RXUNDIM

偏移地址：0x0034

位	名称	读/写	默认值	描述
31:13	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
12	<i>TOIM</i>	RO	0	超时中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
11	<i>GCIM</i>	RO	0	通用广播呼叫中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
10	<i>STDETIM</i>	RO	0	START条件检测中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
9	<i>STPDETIM</i>	RO	0	STOP条件检测中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
8	<i>ACTIM</i>	RO	0	I2C有活动检测中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
7	<i>RXDONEIM</i>	RO	0	接收（从机）传输完成中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
6	<i>TXABRTIM</i>	RO	0	发送终止中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
5	<i>RDREQIM</i>	RO	0	接收（从机）请求数据中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
4	<i>TXEPTIM</i>	RO	0	发送缓冲区空中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
3	<i>TXOVIM</i>	RO	0	发送溢出中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
2	<i>RXFULLIM</i>	RO	0	接收缓冲区满中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。

位	名称	读/写	默认值	描述
1	RXOVERIM	RO	0	接收上溢中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
0	RXUNDIM	RO	0	接收下溢中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。

4.14.3.15 I2C_RIS - I2C 原始中断状态寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
			TORI	GCRI	STADETRI	STPDETRI	ACTRI	RXDNR	TXABRTRI	RDREQRI	TXEPTRI	TXOVRI	RXFULLRI	RXOVERRI	RXUNDRI

偏移地址：0x0038

位	名称	读/写	默认值	描述
31:13	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
12	TORI	RO	0	超时原始中断状态 总线没有闲置，而且SCL信号保持低电平比I2C_TIMEOUT寄存器中配置的时间长时，此中断被设定。 0：该中断没有发出请求。 1：该中断正在发出请求。
11	GCRI	RO	0	通用广播呼叫原始中断状态 当接收和应答到一个广播地址，此中断被设定。 0：该中断没有发出请求。 1：该中断正在发出请求。
10	STDETRI	RO	0	START条件检测原始中断状态 指示是否I2C接口上发生了START或RESTART条件，无论I2C是否在从机或主机模式下工作。 0：该中断没有发出请求。 1：该中断正在发出请求。
9	STPDETRI	RO	0	STOP条件检测原始中断状态 指示是否I2C接口上发生了STOP条件，无论I2C是否在从机或主机模式下工作。 0：该中断没有发出请求。 1：该中断正在发出请求。
8	ACTRI	RO	0	I2C有活动检测原始中断状态 该位捕获I2C活动，并保持设置，直到它被清除。 0：该中断没有发出请求。 1：该中断正在发出请求。

位	名称	读/写	默认值	描述
7	<i>RXDONERI</i>	RO	0	接收（从机）传输完成原始中断状态 当I2C充当从机发送器，如果主机不应答字节的发送，该位被设置为1。 这发生在传输的最后一个字节，指示该传输已经完成。 0：该中断没有发出请求。 1：该中断正在发出请求。
6	<i>TXABTRRI</i>	RO	0	发送终止原始中断状态 该位指示I2C，作为I2C发送器，无法完成对发送FIFO的内容的预期操作。这种情况在I2C作为主机或从机时都可以发生。 0：该中断没有发出请求。 1：该中断正在发出请求。
5	<i>RDREQRI</i>	RO	0	接收（从机）请求数据原始中断状态 当I2C充当从机而另一个I2C主机正试图从I2C读取数据，此位设置。 0：该中断没有发出请求。 1：该中断正在发出请求。
4	<i>TXEPTRI</i>	RO	0	发送缓冲区空原始中断状态 当发送缓冲器处于或低于在I2C_TX_TL寄存器中设置的阈值，该位被置位。 0：该中断没有发出请求。 1：该中断正在发出请求。
3	<i>TXOVRI</i>	RO	0	发送溢出原始中断状态 如果发送缓冲区已满，在处理器试图通过写入I2C_DATA_CMD寄存器发出另一条I2C命令，该位被发送过程中设置。 0：该中断没有发出请求。 1：该中断正在发出请求。
2	<i>RXFULLRI</i>	RO	0	接收缓冲区满原始中断状态 当接收缓冲区达到或高于在I2C_RX_TL寄存器中的RX_TL的阈值时设置 0：该中断没有发出请求。 1：该中断正在发出请求。
1	<i>RXOVERRI</i>	RO	0	接收上溢原始中断状态 如果接收缓冲区满和一个额外的字节从外部I2C设备接收，这个位被设置。I2C应答这额外的字节，但FIFO在满后，任何接收数据都会丢失。 0：该中断没有发出请求。 1：该中断正在发出请求。
0	<i>RXUNDRI</i>	RO	0	接收下溢原始中断状态 如果处理器尝试从I2C_DATA_CMD读接收缓冲器，此位设置。 0：该中断没有发出请求。 1：该中断正在发出请求。

4.14.3.16 I2C_ISC - I2C 中断状态和清除寄存器

注：该寄存器是读和写清除寄存器。写“1”到单独的位可以清除各自的中断。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RW1C	11 RW1C	10 RW1C	9 RW1C	8 RW1C	7 RW1C	6 RW1C	5 RW1C	4 RW1C	3 RW1C	2 RW1C	1 RW1C	0 RW1C
			TOIS	GCIS	STADETIS	STPDETIS	ACTIS	RXDNIS	TXABRTIS	RDREQIS	TXEPTIS	TXOVIS	RXFULLIS	RXOVERIS	RXUNDIS

偏移地址：0x003C

位	名称	读/写	默认值	描述
31:13	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
12	TOIS	RW1C	0	超时中断状态和清除 0：该中断没有发出请求或被屏蔽。 1：该中断已经发出请求。
11	GCIS	RW1C	0	通用广播呼叫中状态和清除 0：该中断没有发出请求或被屏蔽。 1：该中断已经发出请求。
10	STDETIS	RW1C	0	START条件检测中断状态和清除 0：该中断没有发出请求或被屏蔽。 1：该中断已经发出请求。
9	STPDETIS	RW1C	0	STOP条件检测中断状态和清除 0：该中断没有发出请求或被屏蔽。 1：该中断已经发出请求。
8	ACTIS	RW1C	0	I2C有活动检测中断状态和清除 0：该中断没有发出请求或被屏蔽。 1：该中断已经发出请求。
7	RXDONEIS	RW1C	0	接收（从机）传输完成中断状态和清除 0：该中断没有发出请求或被屏蔽。 1：该中断已经发出请求。
6	TXABRTIS	RW1C	0	发送终止中断状态和清除 0：该中断没有发出请求或被屏蔽。 1：该中断已经发出请求。
5	RDREQIS	RW1C	0	接收（从机）请求数据中断状态和清除 0：该中断没有发出请求或被屏蔽。 1：该中断已经发出请求。
4	TXEPTIS	RW1C	0	发送缓冲区空中断状态和清除 0：该中断没有发出请求或被屏蔽。 1：该中断已经发出请求。
3	TXOVIS	RW1C	0	发送溢出中断状态和清除 0：该中断没有发出请求或被屏蔽。 1：该中断已经发出请求。
2	RXFULLIS	RW1C	0	接收缓冲区满中断状态和清除 0：该中断没有发出请求或被屏蔽。 1：该中断已经发出请求。

位	名称	读/写	默认值	描述
1	<i>RXOVERIS</i>	RW1C	0	接收上溢中断状态和清除 0：该中断没有发出请求或被屏蔽。 1：该中断已经发出请求。
0	<i>RXUNDIS</i>	RW1C	0	接收下溢中断状态和清除 0：该中断没有发出请求或被屏蔽。 1：该中断已经发出请求。

4.14.3.17 I2C_RX_TL- I2C 接收 FIFO 阈值寄存器

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

偏移地址：0x0040

位	名称	读/写	默认值	描述
31:8	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
7:0	<i>RXTL</i>	RW	0x0	接收FIFO阈值深度 此域控制触发RXFULL中断的数据进入量。

4.14.3.18 I2C_TX_TL- I2C 发送 FIFO 阈值寄存器

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

偏移地址：0x004C

位	名称	读/写	默认值	描述
31:8	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
7:0	<i>TXTL</i>	RW	0x0	发送FIFO阈值深度 此域控制触发TXEMPTY中断的数据进入量。

4.14.3.19 I2C_ENABLE- I2C 使能寄存器

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

偏移地址：0x006C

位	名称	读/写	默认值	描述
31:8	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
7:0	<i>ENABLE</i>	R/W	0x0	I2C使能 0：I2C禁止。I2C被禁用时，出现以下情况：TX FIFO和RX FIFO被刷新 1：I2C使能。

4.14.3.20 I2C_STATUS- I2C 状态寄存器

这是一个只读寄存器，用于表示当前的传输状态和 FIFO 状态。

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

偏移地址：0x0070

位	名称	读/写	默认值	描述
31:5	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
4	<i>RFF</i>	RO	0	接收FIFO完全满 0：接收FIFO没有满。 1：接收FIFO已满。
3	<i>RFNE</i>	RO	0	接收FIFO没有空 0：接收FIFO已空。 1：接收FIFO没有空。
2	<i>TFE</i>	RO	1	发送FIFO完全空 0：发送FIFO没有空。 1：发送FIFO已空。
1	<i>TFNF</i>	RO	1	发送FIFO没有满 0：发送FIFO已满。 1：发送FIFO没有满。
0	<i>ACTIVITY</i>	RO	0	I2C活动状态 0：I2C处于空闲状态。 1：I2C处于活动状态。



4.14.3.21 I2C_TXFLR- I2C 发送 FIFO 深度寄存器

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

偏移地址：0x0074

位	名称	读/写	默认值	描述
31:3	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
2:0	TXFLR	RO	0x0	发送FIFO深度 此域包含进入发送FIFO中的有效数据的数量。

4.14.3.22 I2C_RXFLR- I2C 接收 FIFO 深度寄存器

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

偏移地址：0x0078

位	名称	读/写	默认值	描述
31:3	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
2:0	RXFLR	RO	0x0	接收FIFO深度 此域包含进入接收FIFO中的有效数据的数量。

4.14.3.23 I2C_TIMEOUT- I2C 超时使能寄存器

此寄存器包含超时功能使能和超时计数值。

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

偏移地址：0x007C

位	名称	读/写	默认值	描述
31:12	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
11:4	TOVAL	RO	0x0	超时计数值
3:1	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
0	TOEN	R/W	0x0	超时使能 这一功能只针对主机而言。当超时功能使能后，如果从机还没有就绪，主机能够等待256计数周期并且复位自己。 0：超时功能禁止。 1：超时功能使能。

4.14.3.24 I2C_TX_ABRT- I2C 发送终止状态寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 R/W	14 R/W	13 R/W	12 R/W	11 R/W	10 R/W	9 R/W	8 R/W	7 R/W	6 R/W	5 R/W	4 R/W	3 R/W	2 R/W	1 R/W	0 R/W
SREQINTX	SARBLOST	SFLUSHTX	ARBLOST	ARBMDIS	10RNR	SBNR	HSNR	SBACK	HSACK	GCR	GCNACK	TXDNACK	10ADR2NACK	10ADR1NACK	7ADRNACK

偏移地址：0x0080

位	名称	读/写	默认值	描述
31:16	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15	SREQINTX	R/W	0	从机请求发送数据 从机请求主机发送数据，但是主机在发送FIFO中却被配置成读命令。
14	SARBLOST	R/W	0	从机丢失仲裁 1：当从机正在发送数据到远程主机时，它却丢失了总线。
13	SFLUSHTX	R/W	0	从机刷新发送FIFO 1：从机接收了一个读命令，但是有数据已经放到了发送FIFO中，因此它发布一个发送仲裁去冲掉在发送FIFO中的旧数据。
12	ARBLOST	R/W	0	仲裁丢失 1：主机丢失了仲裁，或如果此时SARLO位也被置1，表示从机发送器丢失了仲裁。

位	名称	读/写	默认值	描述
11	ARBMDIS	R/W	0	主机仲裁禁止 1：用户试图使用被禁止的主机。
10	10RNR	R/W	0	10位地址读并且无重发功能 1：重发功能被禁止的情况下，主机却在10位地址模式下发出一个读命令。
9	SBNR	R/W	0	起始字节（START BYTE）并且无重发功能 1：重发功能被禁止的情况下，主机却发出一个起始字节（START BYTE）命令。
8	HSNR	R/W	0	高速模式并且无重发功能 1：重发功能被禁止的情况下，用户却试图使用主机在高速模式中发送数据。
7	SBACK	R/W	0	起始字节（START BYTE）ACK应答 1：主机发送了一个起始字节（START BYTE）的同时收到了一个它的ACK应答（错误行为）。
6	HSACK	R/W	0	高速模式ACK应答 1：主机处在高速模式中发送主机编号的同时收到了它的ACK应答（错误行为）。
5	GCR	R/W	0	通用广播呼叫读 1：主机发起一个通用广播传输，但是用户却配置读写位为读。
4	GCNACK	R/W	0	通用广播呼叫无ACK应答 1：主机发起一个通用广播呼叫，但在总线上没有从机发送ACK来响应此呼叫。
3	TXDNACK	R/W	0	发送数据无ACK应答 1：主机发送地址时收到了从机的ACK应答，随后主机发送数据到从机时却没有收到该从机的ACK应答。
2	10ADR2NACK	R/W	0	10位地址的第二字节地址没有ACK应答 1：在10位地址模式下主机发送的第二个字节地址没有收到ACK应答。
1	10ADR1NACK	R/W	0	10位地址的第一字节地址没有ACK应答 1：在10位地址模式下主机发送的第一个字节地址没有收到ACK应答。
0	7ADRNACK	R/W	0	7位地址没有ACK应答 1：在7位地址模式下主机发送的地址没有收到ACK应答。

4.15 串行外设接口 (SPI)

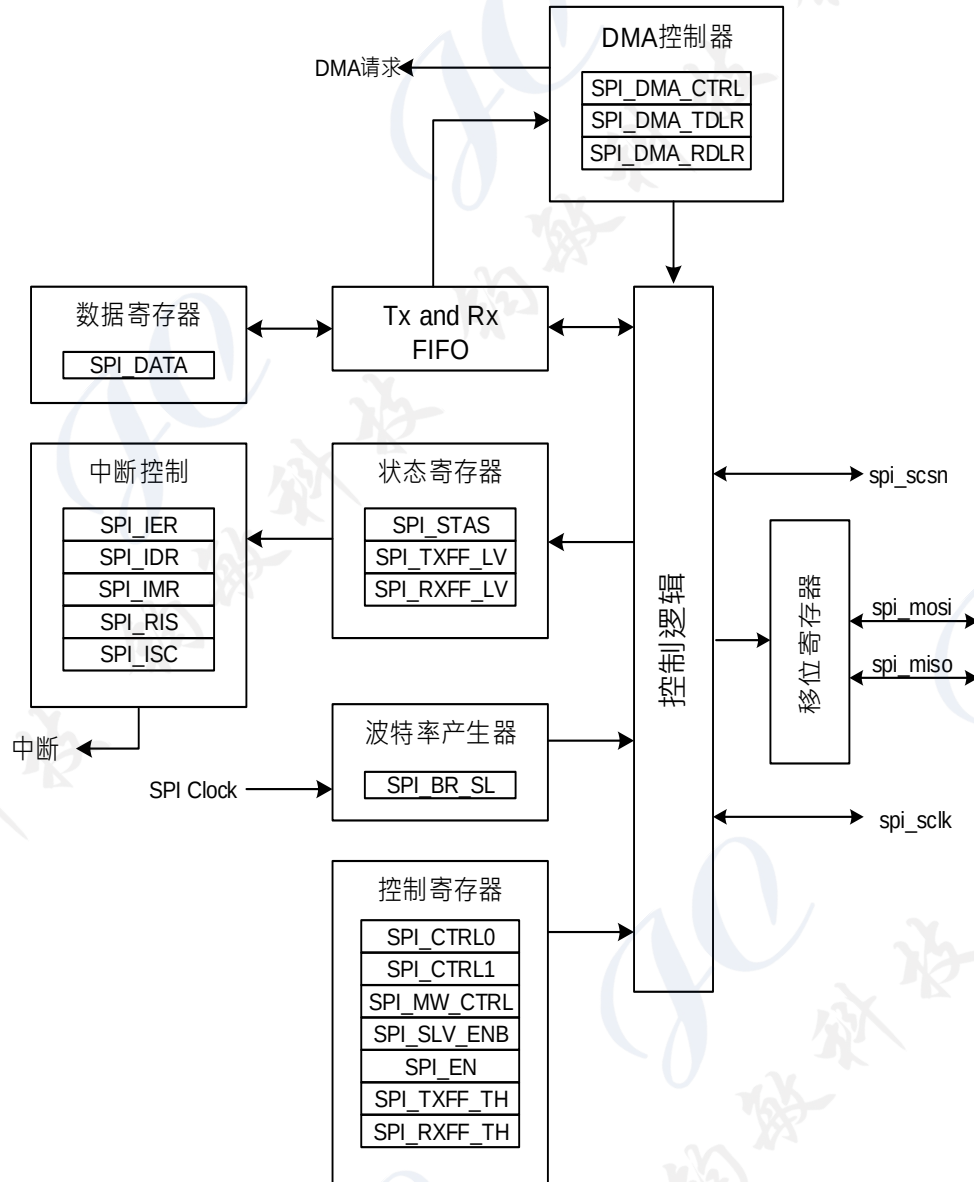
同步串行控制器可以控制 SPI 操作。串行外设接口支持与外部设备以半双工，全双工和单同步的串行方式通信。在总线上可以有多个主机或从机存在，但是每一次数据传输只允许一个主机和从机通信。在全双工数据传输模式中，4 位到 16 位的数据通过主机被发送到从机或从从机发送到主机。

SPI 主要特点

- 支持主/从操作。
- 串行主和串行从控制器– 通过串行- 主或串行- 从外设器件使能串行通信。
- 串行接口操作– 从可编程的以下协议中进行选择
 - 摩托罗拉 SPI 协议
 - 德州仪器同步串行协议(SSP)
 - 美国国家半导体 Microwire 总线。
- 支持同步串行通信。
- 单数据方向操作允许在 MISO 或 MOSI 引脚上实现交替功能。
- 每帧数据支援 4~16 位。

4.15.1 块图

图 4.15-1: I2C 块图



- **SPIx_MOSI**：主机数据输出/从机数据输入。一般情况下，该引脚用于在从机模式下接收数据和在主机模式下发送数据。
- **SPIx_MISO**：主机数据输入/从机数据输出。一般情况下，该引脚用于在从机模式下发送数据和在主机模式下接收数据。
- **SPIx_SCLK**：SPI 主机/从机的串行时钟输出/输入引脚。
- **SPIx_SCSN**：从机选择引脚。根据 **SPI** 和 **SCSN** 的设置，这个引脚可以被用作下面任何一个：
 - 选择一个从机设备进行通信。
 - 同步数据帧。
 - 检测多主机之间的冲突。

4.15.2 SPI 功能描述

4.15.2.1 串行比特率时钟

SPI 串行比特率时钟 (SPI_SCLK) 的最大频率为 SPI 主时钟 (SPI 时钟) 的频率的四分之一。

SPI 的 SCLK 的频率由以下公式定义：

$$F_{SPI_SCLK} = F_{SPI\ Clock} / SCKDV$$

SCKDV 是寄存器 I2C_BR_SL 一个位域，保存一个在 65534 至 4 范围内的一偶数值。如果 SCKDV 等于或小于 3，则 SPI_SCLK 被禁用。

4.15.2.2 传输模式

当传输串行总线上的数据时，SPI 控制器在该部分讨论的模式中操作。

传输模式(TMOD) 通过写入 SPI_CTRL0.TMOD 位而设置。

注意: 传输模式设置不影响串行传输的双工。微总线传输忽略 SPI_CTRL0.TMOD，它由 SPI_MW_CTRL 寄存器控制。

注意：在主模式下，SPI MOSI 线表示为 txd 线; SPI_MISO 行表示为 rxd 线。在从模式下，SPI MOSI 线表示为 rxd 线; SPI_MISO 行表示为 txd 线。

发送和接收

当 SPI_CTRL0.TMOD = 0 时，发送和接收逻辑均有效。根据所选的帧格式(串行协议)，数据传输正常进行。发送数据从发送 FIFO 缓冲器移除并且通过 txd 线发送到目标器件，该目标器件会返回 rxd 线的数据。目标器件的接收数据在每个数据帧结束时从接收移位寄存器移到接收 FIFO 缓冲器。

只发送

当 SPI_CTRL0.TMOD = 1 时，所有接收数据均被忽略。根据所选的帧格式(串行协议)，数据传输正常进行。发送数据从发送 FIFO 缓冲器移除并且通过 SPI_MOSI 线发送到目标器件，该目标器件会返回 rxd 线的数据。数据帧结束时，接收移位寄存器不将最新接收的数据加载到接收 FIFO 缓冲器。接收移位寄存器中的数据由下一个传输所覆盖。当进入该模式时，应该屏蔽来自接收逻辑的中断。

只接收

当 SPI_CTRL0.TMOD = 2 时，发送数据无效。SPI 从器件的情况中，发送 FIFO 缓冲器在只接收模式中从不被移除。传输期间，txd 输出保持在恒定逻辑电平。根据所选帧格式(串行协议)，数据传输正常进行。每个数据帧结束时，目标器件的接收数据从接收移位寄存器移到接收 FIFO 缓冲器。当进入该模式时，应该屏蔽来自发送逻辑的中断。

EEPROM 读

注意: 该传输模式只对串行主器件有效。

当 SPI_CTRL0.TMOD = 3 时，发送数据是用于发送一个 opcode 和/或一个地址到 EEPROM 器件。通常该行为使用 3 个数据帧(8-位高位地址和 8-位 低位地址紧跟 8-位 opcode)。opcode 和地址的传输期间，接收逻辑没有采集数据(只要 SPI 主器件发送 txd 线的数据，rxd 线的数据就会被忽略)。SPI 主器件继续发送数据直到发送 FIFO 缓冲器为空。因此，发送 FIFO 缓冲器中只能有足够的数据帧以对 EEPROM 提供 opcode 和地址。如果发送 FIFO 缓冲器中的数据帧多于所需的，那么读数据丢失。当发送 FIFO 缓冲器为空时(所有控制信息已经被发送)，那么接收线(rxd) 的数据有效并且被存储在接收 FIFO 缓冲器中；txd 输出保持在恒定逻辑电平。串行传输继续进行直到 SPI 主器件接收的数据帧数匹配 SPI_CTRL1.NDF 域的值加 1

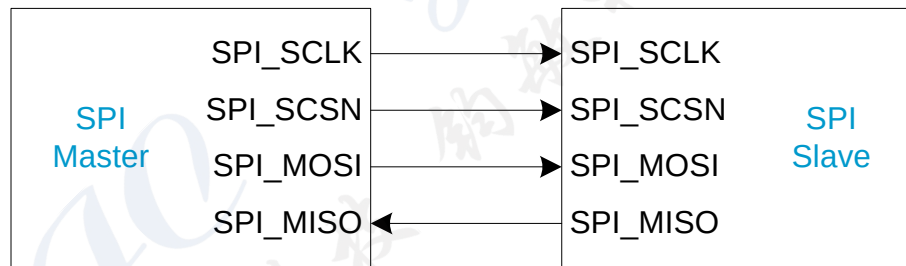
4.15.2.3 SPI 主器件與从器件模式

当传输串行总线上的数据时，SPI 控制器在该部分讨论的模式中操作。

I2C 可以被配置在下面的两个基本模式：

- SPI 主模式
- SPI 從屬模式

图 4.15-2: SPI 主模式與從屬模式連結



SPI 主器件

SPI 主器件发起并控制与串行从外围设备的所有串行传输。串行比特率时钟由 SPI 生成和控制，驱动于 SPI_SCLK 线。当 SPI 被禁止，就不会发生串行传输和 SPI_SCLK 处于“无效”状态中。SPI_SCSN 引脚在完整的数据传输过程中被激活。数据流将会在移位寄存器被发送或被接收到 SPI_MOSI 和 SPI_MISO 引脚上。

数据传输

当符合以下所有条件时，SPI 主器件开始数据传输：

- SPI 主器件被使能
- 发送 FIFO 缓冲器中至少有一个有效入口
- 一个从器件被选择

当有效地传输数据时，状态寄存器(SPI_STAS) 中的忙碌标记(BUSY) 被设置。尝试新的串行传输之前，必须等待直到忙碌标记被清零。

注意: 当数据被写入发送 FIFO 缓冲器时，BUSY 状态不被设置。只有当目标从器件已经被选择并且传输在进行时，该位才被设置。将数据写入发送 FIFO 缓冲器后，直到 SPI_SCLK 信号的正边沿出现移位逻辑才开始串行传输。等待该正边沿的延迟取决于串行传输的波特速率。

主器件 SPI 和 SSP 串行传输

有关 SPI 和 SSP 协议的详细说明，请参阅摩托罗拉 SPI 和德州仪器 SSP。

当传输模式是“发送和接收”或“只发送”(分别为 **SPI_CTRL0.TMOD = 0** 或 **SPI_CTRL0.TMOD = 1**) 时，当发送 FIFO 缓冲器为空时，传输由移位控制逻辑中止。要进行连续地数据传输，必须确保所有数据被发送之前发送 FIFO 缓冲器不为空。发送 FIFO 阈值水平(**SPI_TXFF_TH**) 可用于提前中断(**SPI_RIS.TXERI**) 处理器，从而表明发送 FIFO 缓冲器几乎为空。

当传输模式是“只接收”(**SPI_CTRL0.TMOD = 2**) 时，选择了串行从器件时，串行传输通过将一个“哑”数据字写入发送 FIFO 缓冲器而开始。串行传输期间，SPI 控制器的 **txd** 输出保持在持续逻辑电平。串行传输期间，发送 FIFO 缓冲器仅在开始时被移除并且会保持为空。串行传输结束时由控制寄存器 1(**I2C_CTRL1**) 中“数据帧数”(NDF) 域控制。

当传输模式是“EEPROM read”(**SPI_CTRL0.TMOD = 3**) 时，当一个串行从器件(EEPROM) 被选择时，串行传输通过将 **opcode** 和/ 或地址写入发送 FIFO 缓冲器而开始。**opcode** 和地址被发送到 EEPROM 器件，之后读数据从 EEPROM 器件接收并且存储到接收 FIFO 缓冲器。串行传输结束时由控制寄存器 1(**I2C_CTRL1**) 中的 NDF 域控制。

主器件 Microwire 串行传输

有关与 Microwire 协议的详细说明，请参考美国国家半导体 Microwire 总线部分。

SPI 串行主器件的 Microwire 串行传输由 Microwire 控制寄存器(**SPI_MW_CTRL**) 控制。**SPI_MW_CTRL.MHS** 位域使能和禁用 Microwire 握手接口。**SPI_MW_CTRL.MDD** 位域控制数据帧的方向(控制帧总是由主器件发送以及由从器件接收)。**SPI_MW_CTRL.MWMOD** 位域定义传输是有序的还是无序的。

当发送 FIFO 缓冲器中至少有 1 个控制字并且从器件使能时，所有 Microwire 传输通过 SPI 串行主器件而开始。当 SPI 主器件发送数据帧(**SPI_MW_CTRL.MDD = 1**) 时，发送 FIFO 缓冲器为空时，传输由移位逻辑中止。当 SPI 主器件接收数据帧(**SPI_MW_CTRL.MDD = 0**) 时，传输的中止取决于 **MWMOD** 位域的设置。如果传输是无序的(**SPI_MW_CTRL.MWMOD = 0**)，从从器件移入数据帧后，当发送 FIFO 变空时，传输会被终止。当传输有序时(**SPI_MW_CTRL.MWMOD = 1**)，接收的数据帧数等于 **SPI_CTRL1** 寄存器中的值加 1 时，它被移位逻辑中止。

当 SPI 主器件的握手接口使能(**SPI_MW_CTRL.MHS = 1**) 时，目标从器件的状态在传输后被轮询。只有当从器件报告一个 **ready** 状态时，SPI 主器件才完成传输并且清除 **BUSY** 状态。如果传输是连续的，那么下一个控制/ 数据帧直到从器件返回一个 **ready** 状态才被发送。

SPI 从器件

SPI 从器件处理传输发起的串行通信和受控於串行主机外围设备。

SPI_SCLK 引脚作为输入引脚和串行时钟将从外部主控设备导出。SPI_SCSN 也作为输入引脚。数据流将会在移位寄存器被发送或被接收到 SPI_MOSI 和 SPI_MISO 引脚上。

从器件 SPI 和 SSP 串行传输

有关 SPI 和 SSP 协议的详细说明，请参阅摩托罗拉 SPI 和德州仪器 SSP。

如果 SPI 从器件发送数据到主器件，那么必须确保传输由串行-主器件发起前数据出现在发送 FIFO 缓冲器。当没有数据出现在发送 FIFO 缓冲器时，如果主器件发起到 SPI 从器件的传输，那么错误标记(SPI_STAS.TXE) 在 SPI 状态寄存器中设置，并且之前发送的数据帧在 txd 上重新发送。为了实现连续地数据传输，必须确保发送 FIFO 缓冲器在所有数据被发送前不为空。发送 FIFO 阈值水平寄存器(SPI_TXFF_TH)可用于提前中断(发送 FIFO 空中断 - SPI_RIS.TXERI) 处理器，从而表明发送 FIFO 缓冲器几乎为空。

如果 SPI 从器件是只接收，那么发送 FIFO 缓冲器不需要包含有效数据，因为每次从器件被选择时，发送移位寄存器中的当前数据会被重新发送。

從器件 Microwire 串行传输

有关与 Microwire 协议的详细说明，请参考美国国家半导体 Microwire 总线部分。微总线协议的操作和 SPI 协议大致相同。不存在由 SPI 从器件解码的控制帧。

4.15.2.4 伙伴连接接口

SPI 控制器可以使用以下任何协议连接其它任何 SPI 器件：

- Motorola SPI 协议
- Texas Instrument 串行协议(SSP)
- National Semiconductor Microwire 协议

SPI 控制器支持的串行协议可实现通过使用硬件对串行从器件选择或寻址。串行从器件在专用硬件选择线的控制下被选择。串行主器件生成的选择线数等于总线上出现的串行从器件数。数据传输开始之前，串行-主器件置位目标串行从器件的选择线。

SPI 引脚描述

此表描述在不同的接口下 SPI 引脚的复用功能（SPI_SCK, SPI_SCSN, SPI_MISO 和 SPI_MOSI）。

表 4.15-1: 在不同接口中 SPI 引脚描述

名称	接口名称/功能		描述
	SSP	MicroWire	
SCLK	CLK	SK	串行时钟 SCLK 是用于同步数据传输的时钟信号。使用 SPI 接口时，可将时钟程序设计为高电平有效或低电平有效，否则，它将一直是高电平有效。SCLK 只有在数据传输过程中才跳变。
SCSN	FS	CS	帧同步/从机选择。 当 SPI 是主机时，它在串行数据发起前将该信号驱动到有效状态，再在发送数据后将信号释放到无效状态。该信号为高电平有效还是低电平有效取决于所选择的总线模式。当 SPI 接口为总线从机时，该信号从主机发出。 当只有一个总线主机和一个总线从机时，来自主机的帧同步或从机选择信号可以直接连接到从机的相应输入。当总线上有多个从机时，就有必要进一步限制其帧选择/从机选择信号，以避免多个从机对传输作出响应。
MISO	DR(M) DX(S)	SI (M) SO (S)	主机输入从机输出 MISO 将串行数据由从机传输到主机。当 SPI 是从机时，从该信号上输出串行数据。当 SPI 为主机时，它记录从该信号发出的串行数据。当 SPI 为从机且不被 FS/SSEL 选中时，它不驱动该信号（使其处于高阻状态）。
MOSI	DX(M) DR(S)	SO(M) SI(S)	主机输出从机输入 MOSI 信号将串行数据从主机传输到从机。当 SPI 为主机时，串行数据从该引脚输出。当 SPI 为从机时，该引脚接收从主机输入的数据。

摩托罗拉 SPI

SPI 是 4 线接口，SCSN 为片选信号。SPI 格式的主要特点是 SCK 信号极性和相位可以由 SPI_CTRL0 寄存器的 SCPOL 位和 SCPH 位控制。

时钟极性（SCPOL）和相位（SCPH）控制

SCPOL 控制时钟极性。当设备处于空闲状态时，如果 SCPOL 为低，它将会在 SCLK 引脚产生一个稳定的低电平；如果 SCPOL 为高，它将会在 SCLK 引脚产生一个稳定的高电平。

SCPH 控制位选择捕获数据和允许数据改变时钟沿的状态。第一个数据捕获边缘之前允许或不允许时钟转变对第一次数据传输有很大的影响。当 SCPH 为低，数据将会在时钟的第一个边沿转变被捕获。当 SCPH 为高，数据将会在时钟的第二个边沿转变被捕获。

模式 1 : SCPOL = 0 和 SCPH = 0

当 SCPOL = 0 和 SCPH = 0 时 :

图 4.15-3: 模式 1 下的 SPI 串行单帧格式

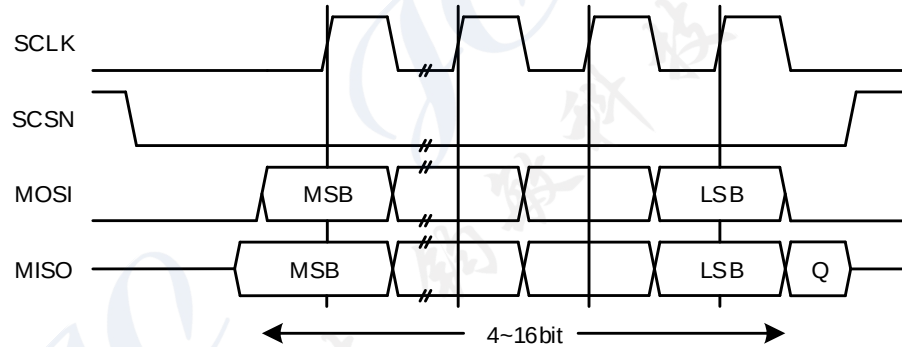
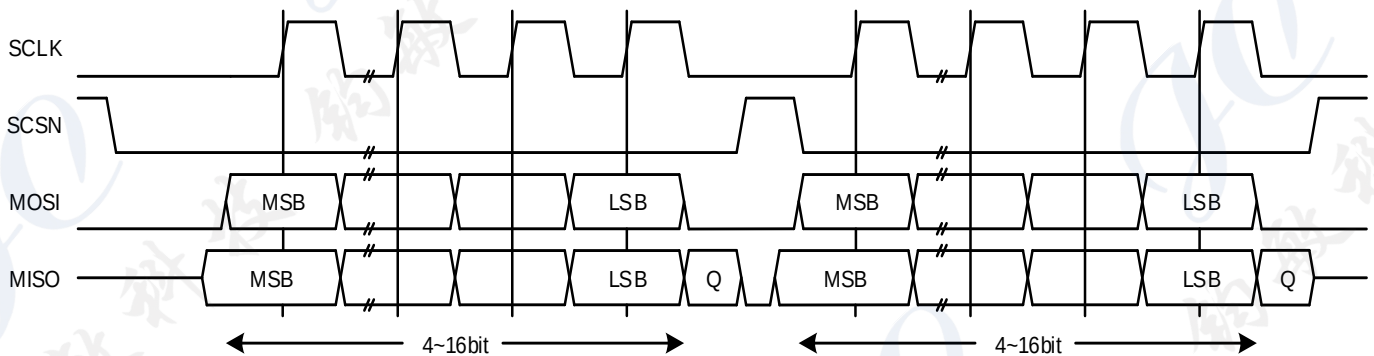


图 4.15-4: 模式 1 下的 SPI 串行连续传输格式



空闲期间的配置 :

- CLK 信号被强制置低。
- SSEL 信号被强制置高。
- MOSI/MISO 处于高阻状态。

当配置参数 SCPH=0 时, 数据传输在从机选择信号的下降沿开始。第一位数据通过主机和外围从机在串行时钟的第一个边沿被捕获; 因此, 有效数据必须在第一个串行时钟边沿之前出现在 tx_d 和 rx_d 线上。和单帧传输一样, 当 SCPH=0 时, 数据传输开始于从机选择信号的下降沿, 连续数据帧传输要求当每帧数据发送结束后从机选择信号都必须置高, 同时在下一帧数据发送之前需要将它置低。

模式 2：SCPOL = 0 和 SCPH = 1

当 SCPOL = 0 和 SCPH = 1 时：

图 4.15-5: 模式 2 下的 SPI 串行单帧格式

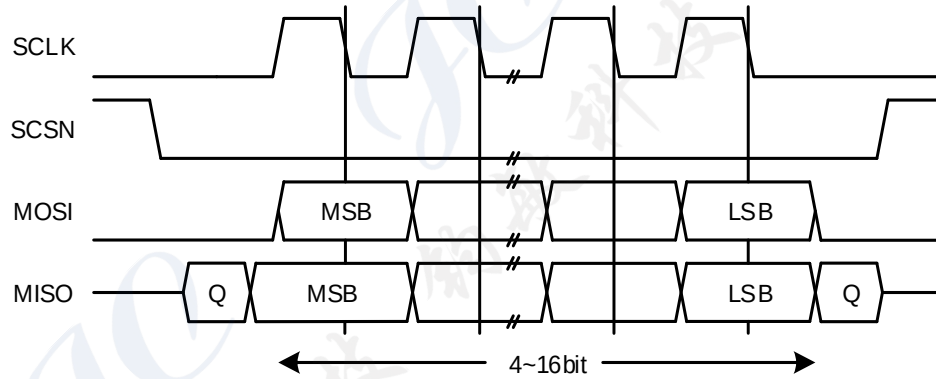
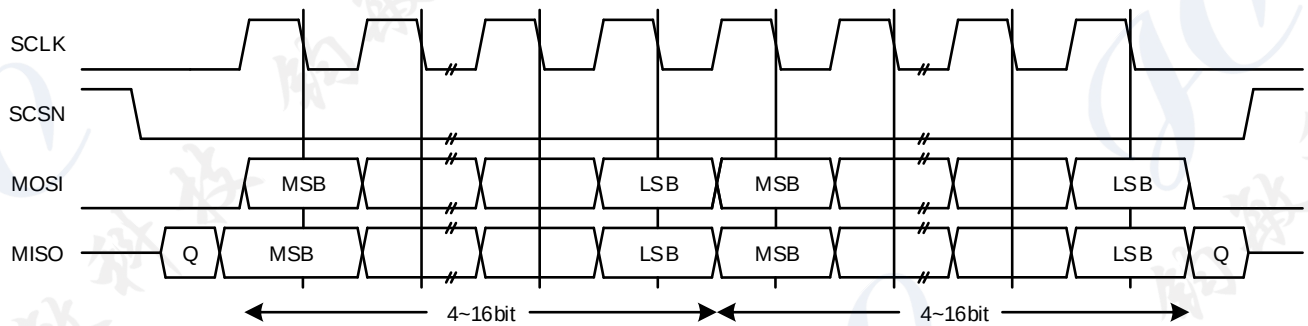


图 4.15-6: 模式 2 下的 SPI 串行连续传输格式



空闲期间的配置：

- CLK 信号被强制置低。
- SSEL 信号被强制置高。
- MOSI/MISO 处于高阻状态。

当配置参数 SCPH=1 时，在从机选择线被激活后主机和外设从机在第一个串行时钟边沿开始传输数据。第一个数据位在第二个串行时钟边沿被捕获。数据通过主机和外设从机在串行时钟的第一个边沿被传播。在连续数据帧传输期间，从机选择线保持为低电平有效直到最后一帧数据的最后一位被捕获。连续数据帧传输和单帧传输方式一样，只是当前帧的最后一位的后面直接更随下一帧的最高位。从机选择信号在传输期间需要保持为有效状态。

模式 3 : SCPOL = 1 和 SCPH = 0

当 SCPOL = 1 和 SCPH = 0 时 :

图 4.15-7: 模式 3 下的 SPI 串行单帧格式

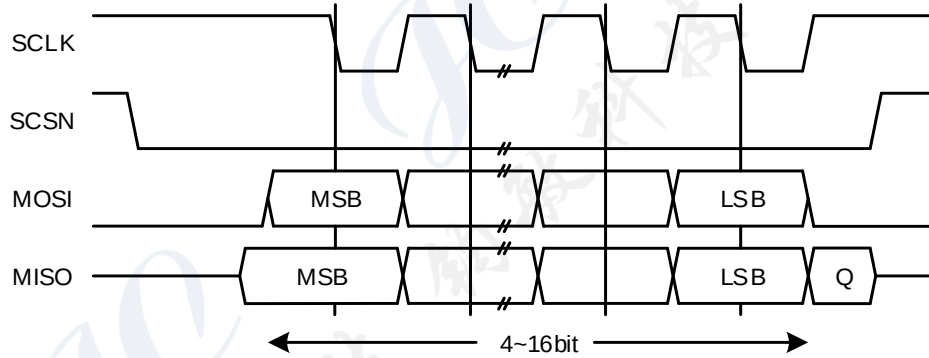
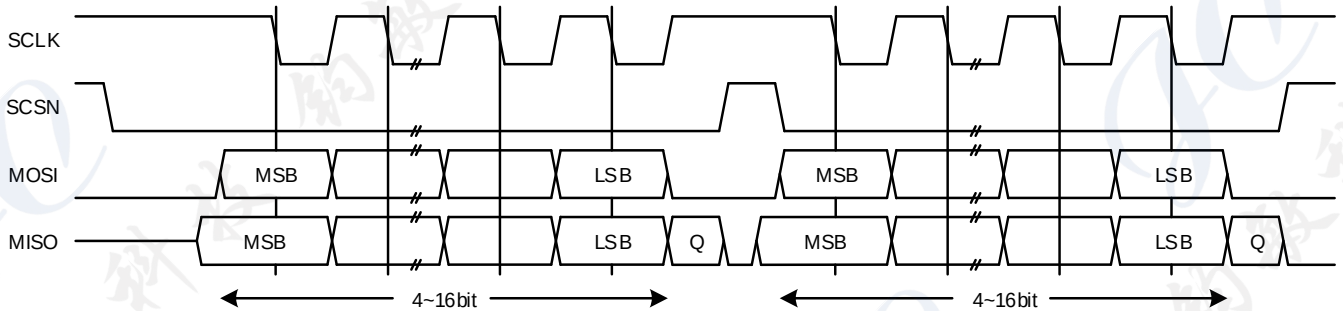


图 4.15-8: 模式 3 下的 SPI 串行连续传输格式



空闲期间的配置:

- CLK 信号被强制置高。
- SSEL 信号被强制置高。
- MOSI/MISO 处于高阻状态。

当配置参数 SCPH=0 时, 数据传输在从机选择信号的下降沿开始。第一位数据通过主机和外围从机在串行时钟的第一个边沿被捕获; 因此, 有效数据必须在第一个串行时钟边沿之前出现在 txd 和 rxd 线上。和单帧传输一样, 当 SCPH=0 时, 数据传输开始于从机选择信号的下降沿, 连续数据帧传输要求当每帧数据发送结束后从机选择信号都必须置高, 同时在下一帧数据发送之前需要将它置低。

模式 4 : SCPOL = 1 和 SCPH = 1

当 SCPOL = 1 和 SCPH = 1 时 :

图 4.15-9: 模式 4 下的 SPI 串行单帧格式

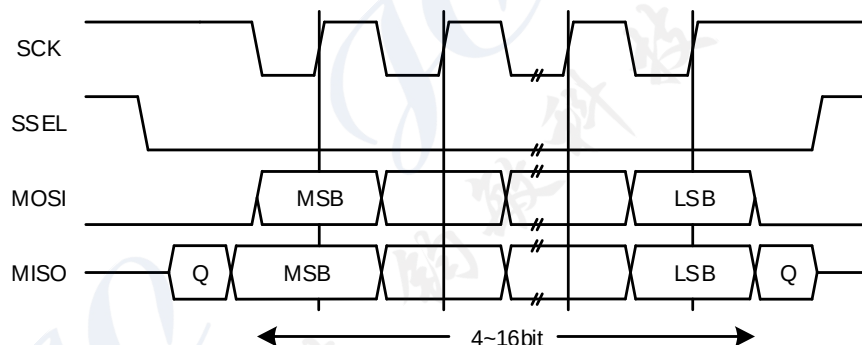
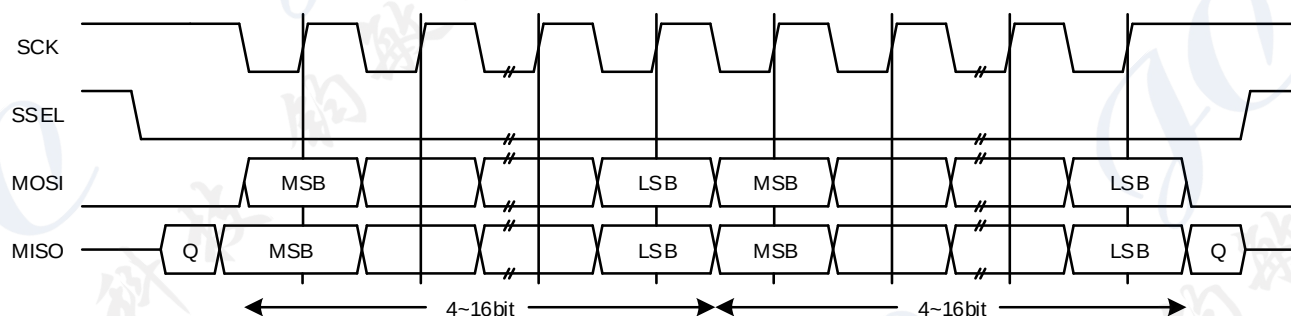


图 4.15-10: 模式 4 下的 SPI 串行连续传输格式



空闲期间的配置 :

- CLK 信号被强制置高。
- SSEL 信号被强制置高。
- MOSI/MISO 处于高阻状态。

当配置参数 SCPH=1 时, 在从机选择线被激活后主机和外设从机在第一个串行时钟边沿开始传输数据。第一个数据位在第二个串行时钟边沿被捕获。数据通过主机和外设从机在串行时钟的第一个边沿被传播。在连数据帧传输期间, 从机选择线保持为低电平有效直到最后一帧数据的最后一位被捕获。连续数据帧传输和单帧传输方式一样, 只是当前帧的最后一位的后面直接更随下一帧的最高位。从机选择信号在传输期间需要保持为有效状态。

德州仪器 SSP

德州仪器串行协议是一个 4 线全双工串行协议。

图 4.15-11：SSP 串行单帧格式

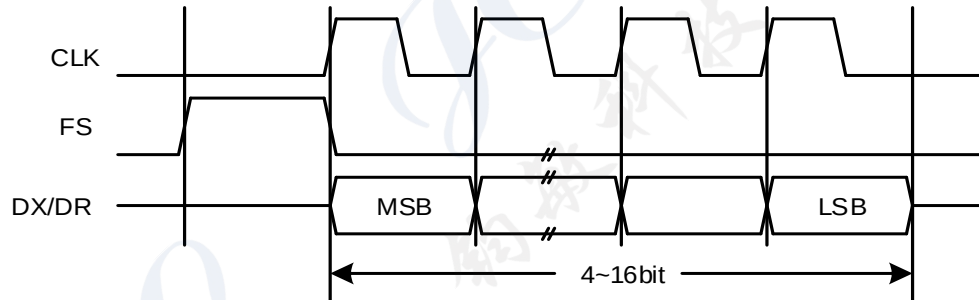
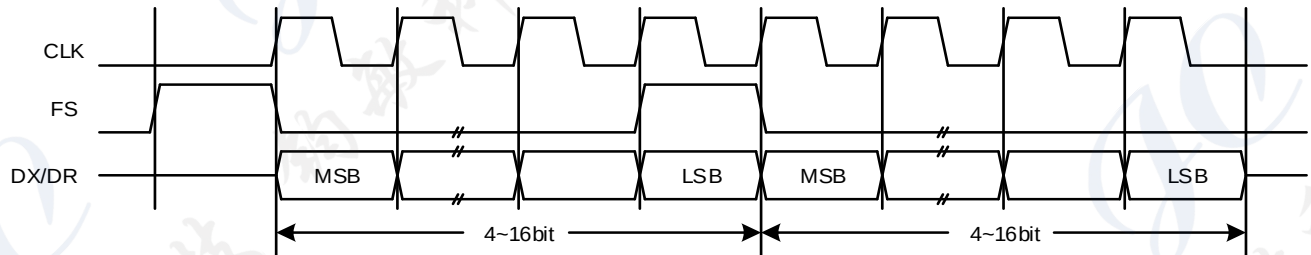


图 4.15-12: SSP 串行连续传输格式

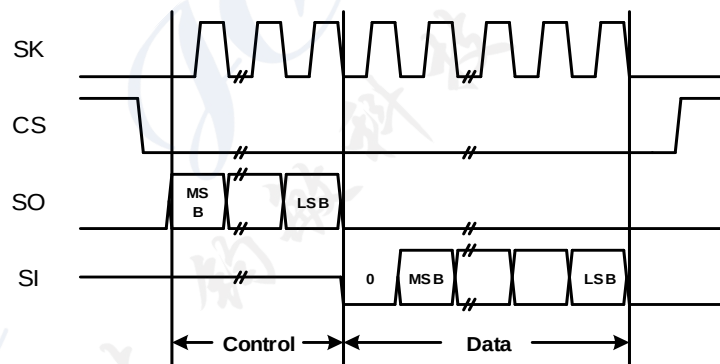


数据传输通过发起一个串行时钟周期的帧指示符（FS）而开始。一个串行时钟周期后需要被发送的数据将被驱动到 DX 线上；同样来自从机的数据会被驱动到 DR 线上。数据在串行时钟的上升沿被传播和在下降沿被捕获。数据帧的长度范围从 4 到 16 位。连续的数据帧传输和单帧的传输方式一样。在相同的时钟周期期间帧指示符维持一个时钟周期用来作为当前传输的最后一位，表示接下来将会是另一个数据帧。

美国国家半导体 Microwire 总线

当串行 SPI 配置为主机时，数据传输随从机选择信号（CS）的下降沿而开始。在半个串行时钟（SK）周期之后，将第一个控制字发送到 SO 线上。通过配置 SPI_CTRL0 寄存器的 CFS（15:12 位）位可以设置控制字的长度，它的范围为 1 到 16 位。剩下的控制字通过 SPI 主机发送出去（在 sclk_out 的下降沿被传播）。在没 有数 据传输期间 SPI 的 SI 线为高阻抗。

图 4.15-13: Microwire 串行格式



数据字的传输方向由 Microwire 控制寄存器（SPI_MW_CTRL）的 MDD 位（位 1）来控制。当 MDD=0 时，表示 SPI 串行主机从外部串行从机接收数据。在一个时钟周期之后控制字的最低位发出，外设从机响应一个虚拟的 0 位，紧接着是长度为 4 到 16 位的数据帧，数据在串行时钟的下降沿被传播和在上升沿被捕获。

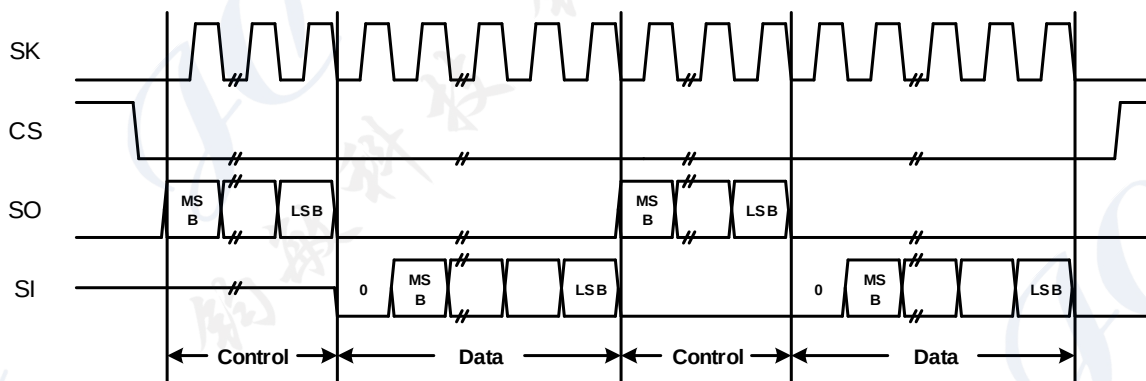
在传输期间从机选择信号为低电平有效，并且数据传输后再过半个时钟周期被终止。

在 Microwire 模式下，用户可以设置控制寄存器来改变传输格式，例如打开传输模式（MDD），或连续数据传输模式（MWMOD），或开启握手功能（MHS）等等，如下面几种模式设置：

模式 1：MHS = 0, MDD = 0 和 MWMOD = 0

当握手功能（MHS）是禁止的，传输模式（MDD）选择为接收，并且连续数据传输模式被禁止时，Microwire 的格式如下图：

图 4.15-14：模式 1 下的 Microwire 串行格式

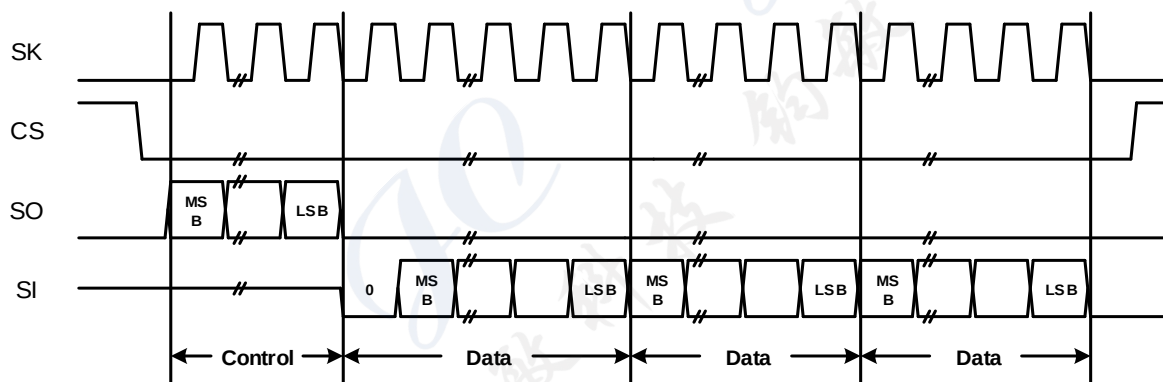


在连续传输配置为该模式下，它与单次传输格式相同，一帧数据接收完成后，紧接着下一帧控制字将会立即被发送出去。一帧数据的最后一个字节的最低位被锁存到 SPI 移位寄存器后，接收到的全部数据在 SK 的下降沿将会被传输到接收 FIFO 中。

模式 2：MHS = 0, MDD = 0 和 MWMOD = 1

当握手功能（MHS）是禁止的，传输模式（MDD）选择为接收，并且连续数据传输模式被使能时，Microwire 的格式如下图：

图 4.15-15: 模式 2 下的 Microwire 串行格式

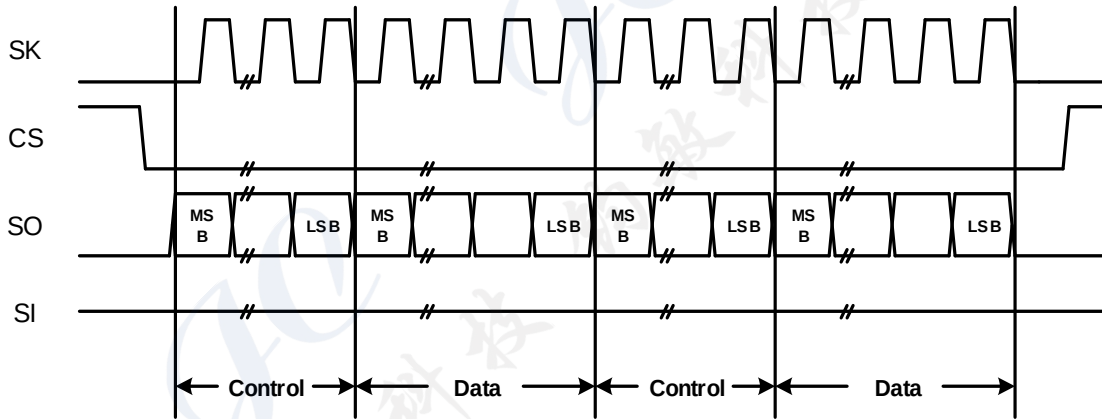


在连续传输配置为该模式下，一帧数据接收完成后，紧接着需要立即接收下一帧数据。接收数据字节的大小可以通过 SPI_CTRL1 寄存器来设置。一帧数据的最后一个字节的最低位被锁存到 SPI 移位寄存器，然后每帧接收到的数据在 SK 的下降沿被传输到接收 FIFO 中。

模式 3：MHS = 0, MDD = 1 和 MWMOD = 0

当握手功能（MHS）是禁止的，传输模式（MDD）选择为发送，并且连续数据传输模式被禁止时，Microwire 的格式如下图：

图 4.15-16: 模式 3 下的 Microwire 串行格式

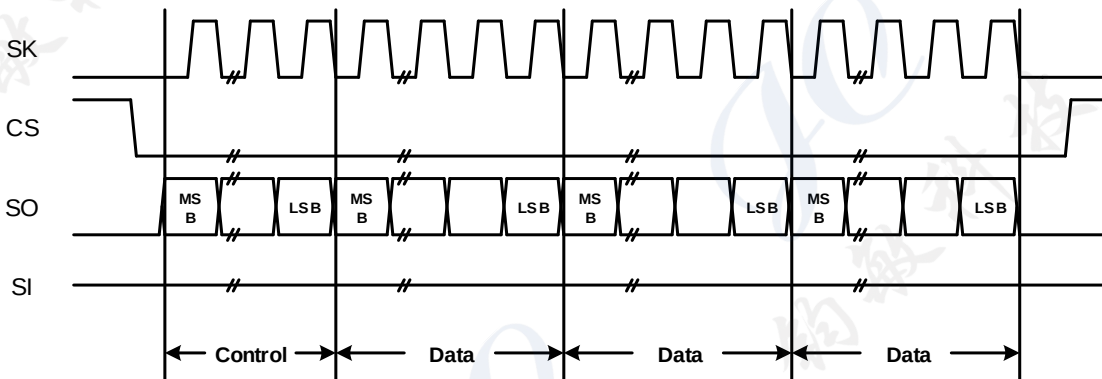


在连续传输配置为该模式下，在当前数据源帧的最低位被发送后，紧接着下一字节的控制字的最高位将要被发送出去。在这种情况下不能接收。

模式 4：MHS = 0, MDD = 1 和 MWMOD = 1

当握手功能（MHS）是禁止的，传输模式（MDD）选择为发送，并且连续数据传输模式被使能时，Microwire 的格式如下图：

图 4.15-17: 模式 4 下的 Microwire 串行格式



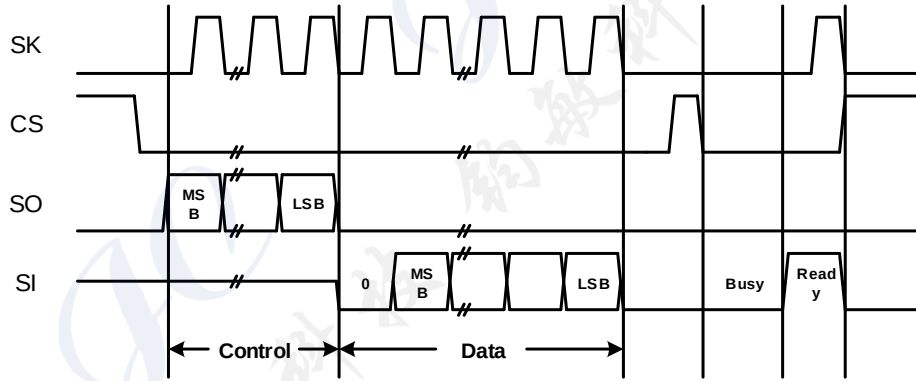
在连续传输配置为该模式下，一帧数据发送完成后，紧接需要立即发送下一帧数据。在这种情况下不能接收。

模式 5 & 6 : MHS = 1, MDD = 0 和 MWMOD = x

当握手功能 (MHS) 被使能, 传输模式 (MDD) 选择为接收, 并且连续数据传输模式被使能/禁止时, Microwire 的格式如下图:

- 仅主机模式有效。

图 4.15-18: 模式 5&6 下的 Microwire 串行格式



在连续传输配置为该模式下, 它与单次传输格式相同。当前数据最后一字节的最低位接收到之后, 将 CS 置高再置低, 然后开始等待外部从机的应答信号。收到应答信号后, 这次传输结束。

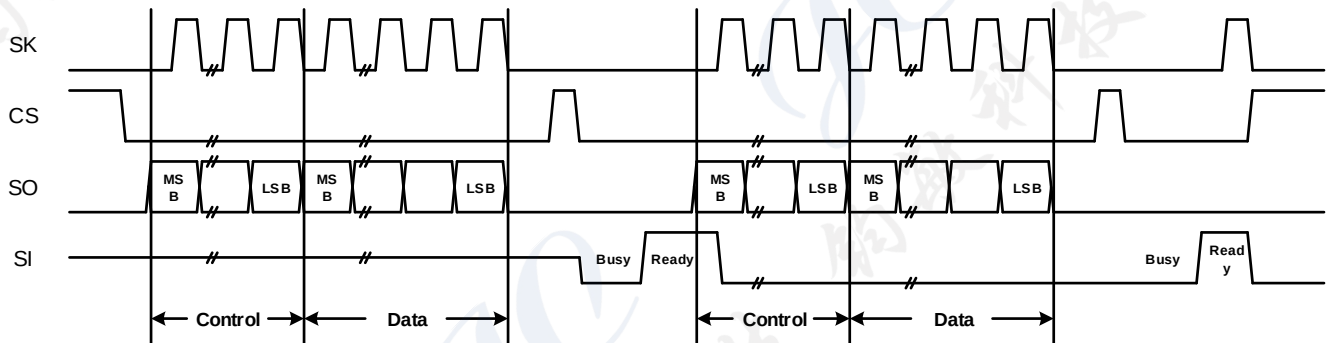
注意: 如果设置 MWMOD = 1, 数据全部接收到后握手信号才发出后 (在 SPI_CTRL1 设置数据长度)。

模式 7 : MHS = 1, MDD = 1 和 MWMOD = 0

当握手功能 (MHS) 被使能, 传输模式 (MDD) 选择为发送, 并且连续数据传输模式被禁止时, Microwire 的格式如下图:

- 仅主机模式有效。

图 4.15-19: 模式 7 下的 Microwire 串行格式



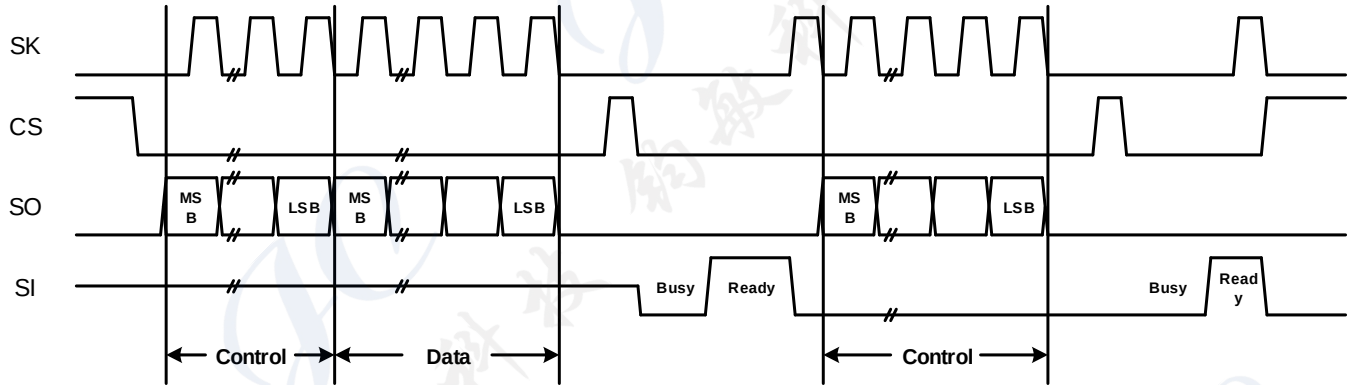
在连续传输配置为该模式下, 当前数据最后一字节的最低位接收之后, 将 CS 置高再置低, 然后开始等待外部从机的应答信号。收到应答信号后, 继续发送控制字和数据字节直到所有的数据都被发送出去。同时, 此次传输结束。

模式 8：MHS = 1, MDD = 1 和 MWMOD = 1

当握手功能（MHS）被使能，传输模式（MDD）选择为发送，并且连续数据传输模式被使能时，Microwire 的格式如下图：

- 仅主机模式有效。

图 4.15-20: 模式 8 下的 Microwire 串行格式



在连续传输配置为该模式下，当前数据最后一字节的最低位接收之后，将 CS 置高再置低，然后开始等待外部从机的应答信号。收到应答信号后，继续发送下一数据字节的最高位，每次发送一个数据握手信号直到所有的数据被发送出去，并且此次传输结束。

4.15.3 SPI 编程模型

本节介绍了 SPI 编程模型:

- 主器件 SPI 和 SSP 串行传输
- 主器件 Microwire 串行传输
- 从器件 SPI 和 SSP 串行传输
- 从器件 Microwire 串行传输

4.15.3.1 主器件 SPI 和 SSP 串行传输

要使用 SPI 主机进行 SPI 和 SSP 串行传输，请执行以下步骤：

1. 通过将 0 写入 SPI 使能寄存器 - **SPI_ENABLE** 的 **SPIEN** 将 SPI 禁用。
2. 对传输设置 SPI 主器件控制寄存器 - **SPI_CTRL0** 和 **SPI_CTRL1**；您可以用任何顺序设置这些寄存器。
 - a. 写入控制寄存器 0 - **SPI_CTRL0**。对于 SPI 传输，必须设置串行时钟极性和串行时钟相位参数等同于目标从器件。
 - b. 如果传输模式是只接收，那么使用传输中的帧数减 1 写入控制寄存器 1 - **SPI_CTRL1**。例如，如果想要接收 5 个数据帧，那么将 4 写入该寄存器。
 - c. 对于传输，写入 Baud 速率选择寄存器 - **SPI_BR_SL** 以设置 baud 速率。
 - d. 写入发送和接收 FIFO 阈值水平寄存器 - **SPI_TXFF_TH** 和 **SPI_RXFF_TH** 以设置 FIFO 缓冲器阈值水平。
 - e. 写中断允许和禁用注册 - **SPI_IER** 和 **SPI_IDR** 控制中断屏蔽。
 - f. 写入从器件使能寄存器(**SPI_SLV_ENB**) 寄存器以便选择性地使能目标从器件。如果从器件使能，那么一个有效数据输入一出现在发送 FIFO 缓冲器，传输就会开始。如果写入数据寄存器(**SPI_DATA**) 之前没有使能从器件，那么传输直到从器件使能才开始。
3. 使能 SPI 主器件。
4. 将传输到目标从器件的数据写入发送 FIFO 缓冲器(写 **SPI_DATA**)。如果此时没有从器件在 **SPI_SLV_ENB** 寄存器使能，那么将其使能以开始传输。
5. 轮询 **BUSY** 状态以等待传输完成。如果进行了发送 FIFO 空中断请求，那么写入发送 FIFO 缓冲器(写 **SPI_DATA**)。如果进行了接收 FIFO 满中断请求，那么读取接收 FIFO 缓冲器(读 **SPI_DATA**)。
6. 当发送 FIFO 缓冲器为空时，移位控制逻辑停止传输。如果传输模式是只接收(**SPI_CTRL0.TMOD** = 2)，那么当接收到指定的帧数时，移位控制逻辑停止传输。当传输完成时，**BUSY** 状态被复位至 0。
7. 如果传输模式不是只发送(**SPI_CTRL0.TMOD** ≠ 1)，那么读取接收 FIFO 缓冲器直到它为空。

4.15.3.2 主器件 Microwire 串行传输

要使用 SPI 主机进行 Microwire 串行传输，请执行以下步骤：

1. 通过将 0 写入 SPI 使能寄存器 - **SPI_ENABLE** 的 **SPIEN** 将 SPI 禁用。
2. 对传输设置 SPI 主器件控制寄存器 - **SPI_CTRL0** 和 **SPI_CTRL1**；您可以用任何顺序设置这些寄存器：
 - a. 写入 **SPI_CTRL0** 以设置传输参数。如果传输是有序的并且 SPI 主器件接收到数据，那么使用传输中的帧数减 1 写入 **SPI_CTRL1**。例如，如果想要接收 5 个数据帧，那么将 4 写入该寄存器。
 - b. 写入 Baud 速率选择寄存器 - **SPI_BR_SL** 以设置 baud 速率。
 - c. 写入 **SPI_TXFF_TH** 和 **SPI_RXFF_TH** 以设置 FIFO 缓冲器阈值水平。
 - d. 写中断允许和禁用注册 - **SPI_IER** 和 **SPI_IDR** 控制中断屏蔽。
 - e. 写入从器件使能寄存器(**SPI_SLV_ENB**) 寄存器以便选择性地使能目标从器件。如果从器件使能，那么一个有效数据输入一出现在发送 FIFO 缓冲器，传输就会开始。如果写入数据寄存器(**SPI_DATA**) 之前没有使能从器件，那么传输直到从器件使能才开始。
 - f. 配置 **SPI_MW_CTRL** 寄存器
3. 使能 SPI 主器件。
4. 将传输到目标从器件的数据写入发送 FIFO 缓冲器(写 **SPI_DATA**)。如果此时没有从器件在 **SPI_SLV_ENB** 寄存器使能，那么将其使能以开始传输。
5. 轮询 **BUSY** 状态以等待传输完成。如果进行了发送 FIFO 空中断请求，那么写入发送 FIFO 缓冲器(写 **SPI_DATA**)。如果进行了接收 FIFO 满中断请求，那么读取接收 FIFO 缓冲器(读 **SPI_DATA**)。
6. 当发送 FIFO 缓冲器为空时，移位控制逻辑停止传输。如果传输模式是有序的 - **SPI_MW_CTRL.MWMOD = 1** 并且 SPI 主器件接收到数据，那么当接收到指定的数据帧数时，移位控制逻辑停止传输。当传输完成时，**BUSY** 状态被复位至 0。
7. 如果 SPI 主器件接收到数据，那么读取接收 FIFO 缓冲器直到它为空。

4.15.3.3 从器件 SPI 和 SSP 串行传输

要使用 SPI 为从器件执行 SPI 和 SSP 串行传输，请执行以下步骤：

1. 通过将 0 写入 SPI 使能寄存器 - **SPI_ENABLE** 的 **SPIEN** 将 SPI 禁用。
2. 对传输设置 SPI 主器件控制寄存器 - **SPI_CTRL0** 和 **SPI_CTRL1**；您可以用任何顺序设置这些寄存器：
 - a. 写入控制寄存器 0 - **SPI_CTRL0**。对于 SPI 传输，必须设置串行时钟极性和串行时钟相位参数等同于目标从器件。
 - b. 写入 **SPI_TXFF_TH** 和 **SPI_RXFF_TH** 以设置 FIFO 缓冲器阈值水平。
 - c. 写中断允许和禁用注册 - **SPI_IER** 和 **SPI_IDR** 控制中断屏蔽。
3. 使能 SPI 主器件。
4. 如果传输模式是发送和接收或是只发送，那么将传输到主器件的数据写入发送 FIFO 缓冲器(写 **SPI_DATA**)。如果传输模式是只接收，那么不需要将数据写入发送 FIFO 缓冲器。发送移位寄存器中的当前值被重新发送。
5. SPI 从器件现在可以开始串行传输。当串行- 主器件选择 SPI 从器件时传输开始。
6. 当传输进行时，**BUSY** 状态可以被轮询以返回传输状态。如果进行了发送 FIFO 空中断请求，那么写入发送 FIFO 缓冲器(写 **SPI_DATA**)。如果进行了接收 FIFO 满中断请求，那么读取接收 FIFO 缓冲器(读 **SPI_DATA**)。
7. 当串行主器件移除 SPI 从器件的选择输入时，传输结束。当完成传输时，**BUSY** 状态被复位至 0。
8. 如果传输模式不是只发送(**SPI_CTRL0.TMOD** ≠ 1)，那么读取接收 FIFO 缓冲器直到它为空。

4.15.3.4 从器件 Microwire 串行传输

对于 SPI 从器件，Microwire 协议的操作方式和 SPI 协议基本相同。不存在 SPI 从器件解码的控制帧。

4.15.4 SPI 寄存器映射

基地址: 0x4800_4000

偏移地址	寄存器名称	读/写	默认值	描述	参考页
0x0000	SPI_CTRL0	R/W	0x0000_0007	SPI控制寄存器0	302
0x0004	SPI_CTRL1	R/W	0x0000_0000	SPI控制寄存器1	304
0x0008	SPI_ENABLE	R/W	0x0000_0000	SPI使能寄存器	304
0x000C	SPI_MW_CTRL	R/W	0x0000_0000	Microwire控制寄存器	305
0x0010	SPI_SLV_ENB	R/W	0x0000_0000	SPI从机使能寄存器	306
0x0014	SPI_BR_SL	R/W	0x0000_0000	SPI波特率选择	306
0x0018	SPI_TXFF_TH	R/W	0x0000_0000	SPI发送FIFO阈值深度	307
0x001C	SPI_RXFF_TH	R/W	0x0000_0000	SPI接收FIFO阈值深度	307
0x0020	SPI_TXFF_LV	RO	0x0000_0000	SPI发送FIFO深度寄存器	308
0x0024	SPI_RXFF_LV	RO	0x0000_0000	SPI接收FIFO深度寄存器	308
0x0028	SPI_STAS	RO	0x0000_0006	SPI状态寄存器	309
0x002C	SPI_IER	WO	0x0000_0000	SPI中断使能寄存器	310
0x0030	SPI_IDR	WO	0x0000_0000	SPI中断禁止寄存器	311
0x0034	SPI_IMR	RO	0x0000_0000	SPI中断屏蔽状态寄存器	312
0x0038	SPI_RIS	RO	0x0000_0000	SPI原始中断状态寄存器	313
0x003C	SPI_ISC	R/W1C	0x0000_0000	SPI中断状态和清除寄存器	314
0x0060	SPI_DATA	R/W	0x0000_0000	SPI数据寄存器	315

4.15.4.1 SPI_CTRL0 - SPI 控制寄存器 0

这个寄存器控制串行数据传输。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RW
															SLAVEEN
15 RW	14 RW	13 RW	12 RW	11 RW	10 RW	9 RW	8 RW	7 RW	6 RW	5 RW	4 RW	3 RW	2 RW	1 RW	0 RW
CFS				SRL	SLVOE	TMOD		SCPOL	SCPH	FRF		DFS			

偏移地址：0x0000

位	名称	读/写	默认值	描述
31:17	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
16	SLAVEEN	RW	0	从机模式使能 0：主机模式。 1：从机模式。
15:12	CFS	RW	0x0	控制字框大小 这个域用来为Microwire框格式选择控制字的长度。 0x0: 1-位控制字 0x1: 2-位控制字 0x2: 3-位控制字 0x3: 4-位控制字 0x4: 5-位控制字 0x5: 6-位控制字 0x6: 7-位控制字 0x7: 8-位控制字 0x8: 9-位控制字 0x9: 10-位控制字 0xA: 11-位控制字 0xB: 12-位控制字 0xC: 13-位控制字 0xD: 14-位控制字 0xE: 15-位控制字 0xF: 16-位控制字
11	SRL	RW	0	移位寄存器循环 这一位仅用于测试。 0：正常模式操作。 1：测试模式操作
10	SLVOE	RW	0	从机输出使能 这一位仅当SPI被配置成串行从机设备时才有效。 0：从机发送数据信号（MISO）使能。 1：从机发送数据信号（MISO）禁止。



位	名称	读/写	默认值	描述
9:8	TMOD	R/W	0x0	传输模式 该域用来选择串行通信的传输模式。 00：发送&接收。 01：仅发送。 10：只接收。 11：读取EEPROM。
7	SCPOL	R/W	0	串行时钟极性 当帧格式（FRF）设置为摩托罗拉SPI时，该位才有效。 0：空闲时，SCK保持为低。 1：空闲时，SCK保持为高。
6	SCPH	R/W	0	串行时钟相位 当帧格式（FRF）设置为摩托罗拉SPI时，该位才有效。 0：串行时钟在第一个数据位的中间跳变。 1：串行时钟在第一个数据位的起始跳变。
5:4	FRF	R/W	0x0	帧格式 选择哪一个串行协议传输数据。 00：摩托罗拉SPI。 01：德州仪器SSP。 10：美国国家半导体Microwire。 11：保留。
3:0	DFS	R/W	0x7	数据帧大小 选择数据帧的长度。不支持使用值的范围是0x0 - 0x2。长度（DFS + 1）位 0x0 - 0x2: 保留。 0x3: 4-位串行数据传输 0x4: 5-位串行数据传输 0x5: 6-位串行数据传输 0x6: 7-位串行数据传输 0x7: 8-位串行数据传输 0x8: 9-位串行数据传输 0x9: 10-位串行数据传输 0xA: 11-位串行数据传输 0xB: 12-位串行数据传输 0xC: 13-位串行数据传输 0xD: 14-位串行数据传输 0xE: 15-位串行数据传输 0xF: 16-位串行数据传输。

4.15.4.2 SPI_CTRL1 - SPI 控制寄存器 1

当只在接收模式下，此寄存器控制串行传输的结束。

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

偏移地址：0x0004

位	名称	读/写	默认值	描述
31:17	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
16	CSP	R/W	0	芯片选择脉冲管理 该位仅在主控模式下使用。它不允许SPI在做连续传输时对连续两个数据产生SCSN脉冲。在单次数据传输的情况下，它迫使转印后SCSN引脚高电平。它具有如果SPI_CTRL0.SCPH='1'，且仅在SPI_CTRL0.FRF='00'使用没有任何意义。
15:0	NDF	R/W	0	数据帧的数量 当TMOD = 10，该寄存器设置通过SPI连续接收数据帧的数量。SPI连续接收串行数据直到数据帧的数量等于该寄存器的值加1，这使用户可以在连续传输模式下接收多达64KB的数据。 当SPI被用作串行从机时，只要它被选择传输就会持续。因此，当SPI被配置为串行从机时，这个寄存器是没有意义的。

4.15.4.3 SPI_ENABLE - SPI 使能寄存器

此寄存器用来使能 SPI。

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

偏移地址：0x0008

位	名称	读/写	默认值	描述
31:1	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
0	SPIEN	R/W	0	使能SPI 使能和禁止所有SPI的操作。当它被禁止时，所有串行传输会被立即停止。当设备被禁止时，发送和接收FIFO缓冲区将会被清除，同时也不可能对SPI的控制寄存器实行编程操作。

4.15.4.4 SPI_MW_CTRL - Microwire 控制寄存器

此寄存器用来为半双工 Microwire 串行协议控制数据字的方向。

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

偏移地址：0x000C

位	名称	读/写	默认值	描述
31:3	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
2	<i>MHS</i>	R/W	0	Microwire握手 这一位仅在SPI被配置为串行主机设备时才有效。用来为Microwire协议使能和禁止“busy/ready”握手接口。当它被使能时，最后一位数据/控制字传输后，在SR寄存器中清除BUSY状态之前，SPI检测来至目标从机的就绪状态。 0：握手接口禁止。 1：握手接口使能
1	<i>MDD</i>	R/W	0	Microwire控制 当Microwire串行协议被使用时，它定义数据字的方向。当这位被设置成0时，数据字通过SPI MacroCell从外部串行设备被接收。当这位被设置成1时，数据字从SPI MacroCell被发送到外部串行设备。
0	<i>MWMOD</i>	R/W	0	Microwire传输模式 它定义Microwire传输是连续的还是非连续的。当使用连续模式时，只需要一个控制字来发送或接收一个数据字块。当使用非连续模式时，必须为每一个被发送或接收的数据字提供一个控制字。 0：非连续传输。 1：连续传输。

4.15.4.5 SPI_SLV_ENB - SPI 从机使能寄存器

只有当 SPI 被配置为主机设备，该寄存器有效。寄存器允许从 SPI 主器件个别从器件选择输出线。

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

偏移地址：0x0010

位	名称	读/写	默认值	描述
31:1	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
0	SER	RW	0	从机选择使能标志 当该位被置位，同时一个串行传输开始时，来自主机相对应的从机选择线被激活。应该注意的是在相对应的从机选择输出直到一个传输被开始，设置或清除该寄存器的某一位是无效的。开始一个传输之前，用户应该使能该寄存器中和主机想要通信的从机的相对应的位。当它处在广播模式下时，该域只有一位应该被置位。 1：选择相对应的从机。 0：未选择相对应的从机。

4.15.4.6 SPI_BR_SL - SPI 波特率选择

只有当 SPI 被配置为主机设备，该寄存器有效。该寄存器导出串行时钟的频率调节数据传送。

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

偏移地址：0x0014

位	名称	读/写	默认值	描述
31:16	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
15:0	SCKDV	RW	0x0	SPI时钟分频器 此域的最低位始终被设置为0，并且对该位进行写操作时无意义的，这可以确保在这个寄存器的值是偶数。如果这个值是低於3，串行输出时钟（SPI_SCLK）将会被禁止。SPI_SCLK的频率从下面的等式获得： $F_{SPI_SCLK} = F_{SPI\ Clock} / SCKDV$ 这里SCKDV是4到65534中的任何一个偶数值。 例如： $F_{SPI\ Clock} = 16\ MHz, \text{ 此时 } SCKDV = 4, F_{SPI_SCLK} = 16 / 4 = 4\ MHz$

4.15.4.7 SPI_TXFF_TH - SPI 发送 FIFO 阈值深度

这个寄存器控制发送 FIFO 存储器的阈值。

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

偏移地址：0x0018

位	名称	读/写	默认值	描述
31:8	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
7:0	TFT	R/W	0x0	发送FIFO阈值 控制发送FIFO控制器触发中断的数据进入量。FIFO深度可在2~8之间进行配置；这个寄存器的大小为需要访问FIFO的地址位的数量。如果用户试图设置这个值大于或等于这个FIFO的深度，这个域将不能被写并且将保持它自己的当前值。当发送FIFO的数据进入量少于或等于这个值时，发送FIFO空中断将会被触发。 000：当发送FIFO中没有数据时触发中断。 001：当发送FIFO中的数据小于等于1时触发中断。 010：当发送FIFO中的数据小于等于2时触发中断。 011：当发送FIFO中的数据小于等于3时触发中断。 100：当发送FIFO中的数据小于等于4时触发中断。 101：当发送FIFO中的数据小于等于5时触发中断。 110：当发送FIFO中的数据小于等于6时触发中断。 111：当发送FIFO中的数据小于等于7时触发中断。

4.15.4.8 SPI_RXFF_TH - SPI 接收 FIFO 阈值深度

这个寄存器控制接收 FIFO 存储器的阈值。

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

偏移地址：0x001C

位	名称	读/写	默认值	描述
31:17	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
16	SLAVEEN	R/W	0	从机模式使能 0：主机模式。 1：从机模式。

4.15.4.9 SPI_TXFF_LV - SPI 发送 FIFO 深度寄存器

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

偏移地址：0x0020

位	名称	读/写	默认值	描述
31:4	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
3:0	TXTLF	RO	0x0	发送FIFO深度 包含在发送FIFO中有效数据进入量。

4.15.4.10 SPI_RXFF_LV - SPI 接收 FIFO 深度寄存器

这个寄存器包含在接收 FIFO 空间中有效数据进入量。

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

偏移地址：0x0024

位	名称	读/写	默认值	描述
31:4	reserved	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
3:0	RXTFL	RO	0x0	接收FIFO深度 包含在接收FIFO中有效数据进入量。

4.15.4.11 SPI_STAS - SPI 状态寄存器

这是一个只读寄存器，指示当前的传送状态，FIFO 状态，以及可能发生的任何传输/接收差错。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 RO	4 RO	3 RO	2 RO	1 RO	0 RO
									DCOL	TXE	RFF	RFNE	TFE	TFNF	BUSY

偏移地址：0x0028

位	名称	读/写	默认值	描述
31:7	<i>reserved</i>	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
6	<i>DCOL</i>	RO	0	数据冲突错误 仅当SPI配置为一个主机设备时才有意义。如果SPI主机正在传输数据时，同时另一个主机试图选择这个设备作为一个从机时，这一位将会被置位。这将告诉处理器在传输完成之前最后的数据传输已经被停止。读操作将会清除这一位。 0：没有错误。 1：传输数据冲突错误。
5	<i>TXE</i>	RO	0	传输错误 在开始一个传输的同时，如果发送FIFO为空，这一位会被置位。这一位仅当SPI被配置成一个从机时才有意义。之前传输的数据在txd线上被重新发送。读操作将会清除这一位。 0：没有错误。 1：传输错误。
4	<i>RFF</i>	RO	0	接收FIFO满 当接收FIFO完全的满，这一位会被置位。当接收FIFO包含一个或多个空位置时，这一位会被清零。 0：接收FIFO没有满。 1：接收FIFO已满。
3	<i>RFNE</i>	RO	0	接收FIFO不为空 当接收FIFO包含一个或多个数据时，这位会被置位；当它为空时，这位会被清零。这位可以通过软件轮询接收FIFO来确定是否完全的空。 0：接收FIFO为空。 1：接收FIFO不为空。
2	<i>TFE</i>	RO	1	发送FIFO空 当发送FIFO完全空时，这位会被置位；当它包含一个或多个有效数据时，这位会被清零。这个位不会产生中断请求。 0：发送FIFO不为空。 1：发送FIFO为空。
1	<i>TFNF</i>	RO	1	发送FIFO未满 当发送FIFO包含一个或多个空位置时，这位会被置位；当它满时，这位会被清零。 0：发送FIFO满。 1：发送FIFO未满。



位	名称	读/写	默认值	描述
0	BUSY	RO	0	SPI忙标志 当这位被置位时，表示一个串行传输正在进行；当它被清零时，表示SPI处于空闲或禁止状态。 0：SPI闲置或禁止。 1：SPI正在传输数据。

4.15.4.12 SPI_IER - SPI 中断使能寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 WO	4 WO	3 WO	2 WO	1 WO	0 WO
										MSTIE	RXFIE	RXOIE	RXUIE	TXOIE	TXEIE

偏移地址：0x002C

位	名称	读/写	默认值	描述
31:6	保留	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
5	MSTIE	WO	0	多主机冲突中断使能 1：该中断被使能。
4	RXFIE	WO	0	接收FIFO满中断使能 1：该中断被使能。
3	RXOIE	WO	0	接收FIFO上溢中断使能 1：该中断被使能。
2	RXUIE	WO	0	接收FIFO下溢中断使能 1：该中断被使能。
1	TXOIE	WO	0	发送FIFO上溢中断使能 1：该中断被使能。
0	TXEIE	WO	0	发送FIFO空中断使能 1：该中断被使能。

4.15.4.13 SPI_IDR - SPI 中断禁止寄存器

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 WO	4 WO	3 WO	2 WO	1 WO	0 WO
										MSTID	RXFID	RXOID	RXUID	TXOID	TXEID

偏移地址：0x0030

位	名称	读/写	默认值	描述
31:6	保留	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
5	MSTID	WO	0	多主机冲突中断禁止 1：该中断被禁止。
4	RXFID	WO	0	接收FIFO满中断禁止 1：该中断被禁止。
3	RXOID	WO	0	接收FIFO上溢中断禁止 1：该中断被禁止。
2	RXUID	WO	0	接收FIFO下溢中断禁止 1：该中断被禁止。
1	TXOID	WO	0	发送FIFO上溢中断禁止 1：该中断被禁止。
0	TXEID	WO	0	发送FIFO空中断禁止 1：该中断被禁止。

4.15.4.14 SPI_IMR - SPI 中断屏蔽状态寄存器

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

偏移地址：0x0034

位	名称	读/写	默认值	描述
31:6	保留	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
5	MSTIM	RO	0	多主机冲突中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
4	RXFIM	RO	0	接收FIFO满中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
3	RXOIM	RO	0	接收FIFO上溢中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
2	RXUIM	RO	0	接收FIFO下溢中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
1	TXOIM	RO	0	发送FIFO上溢中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。
0	TXEIM	RO	0	发送FIFO空中断屏蔽状态 0：该中断将会被屏蔽。 1：该中断不会被屏蔽。

4.15.4.15 SPI_RIS - SPI 原始中断状态寄存器

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

偏移地址：0x0038

位	名称	读/写	默认值	描述
31:6	保留	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
5	MSTRI	RO	0	多主机冲突原始中断状态 0：该中断没有发出请求。 1：该中断正在发出请求。
4	RXFRI	RO	0	接收FIFO满原始中断状态 0：该中断没有发出请求。 1：该中断正在发出请求。
3	RXORI	RO	0	接收FIFO上溢原始中断状态 0：该中断没有发出请求。 1：该中断正在发出请求。
2	RXURI	RO	0	接收FIFO下溢原始中断状态 0：该中断没有发出请求。 1：该中断正在发出请求。
1	TXORI	RO	0	发送FIFO上溢原始中断状态 0：该中断没有发出请求。 1：该中断正在发出请求。
0	TXERI	RO	0	发送FIFO空原始中断状态 0：该中断没有发出请求。 1：该中断正在发出请求。

4.15.4.16 SPI_ISC - SPI 中断状态和清除寄存器

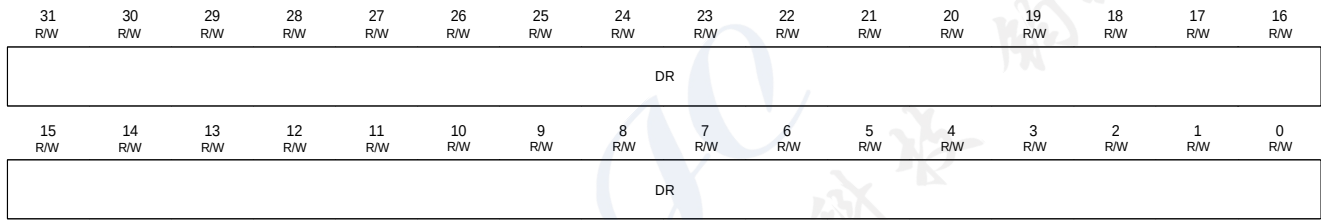
注意：这个寄存器是读和写清除寄存器。写‘1’到某一位将会清除相对应位的中断。

31 RO	30 RO	29 RO	28 RO	27 RO	26 RO	25 RO	24 RO	23 RO	22 RO	21 RO	20 RO	19 RO	18 RO	17 RO	16 RO
15 RO	14 RO	13 RO	12 RO	11 RO	10 RO	9 RO	8 RO	7 RO	6 RO	5 R/W1C	4 R/W1C	3 R/W1C	2 R/W1C	1 R/W1C	0 R/W1C
										MSTIS	RXFIS	RXOIS	RXUIS	TXOIS	TXEIS

偏移地址：0x003C

位	名称	读/写	默认值	描述
31:6	保留	RO	0x0	软件不应该操作此保留位的值。考虑到与其他产品的兼容性，这位的值不应该写或读。
5	MSTIS	R/W1C	0	多主机冲突中断状态和清除 0：没有中断产生或中断被屏蔽。 1：已产生中断。
4	RXFIS	R/W1C	0	接收FIFO满中断状态和清除 0：没有中断产生或中断被屏蔽。 1：已产生中断。
3	RXOIS	R/W1C	0	接收FIFO上溢中断状态和清除 0：没有中断产生或中断被屏蔽。 1：已产生中断。
2	RXUIS	R/W1C	0	接收FIFO下溢中断状态和清除 0：没有中断产生或中断被屏蔽。 1：已产生中断。
1	TXOIS	R/W1C	0	发送FIFO上溢中断状态和清除 0：没有中断产生或中断被屏蔽。 1：已产生中断。
0	TXEIS	R/W1C	0	发送FIFO空中断状态和清除 0：没有中断产生或中断被屏蔽。 1：已产生中断。

4.15.4.17 SPI_DATA- SPI 数据寄存器



偏移地址：0x0060

位	名称	读/写	默认值	描述
31:0	DR	R/W	0x0	数据寄存器 在该寄存器写数据时，必需右对齐数据，而读数据将会自动地右对齐。 读操作：接收FIFO缓冲区。 写操作：发送FIFO缓冲区。

5. 电气特性

5.1 最大额定值

最大额定值是为该设备不被永久性损坏而提出的限制。通过超出绝对最大额定值来延长可能会导致设备的可靠性受到不利影响。

符号	参数	最小值	最大值	单位
VDD -VSS	VDD 电源电压	2.0	3.6	V
VI/VO	输入电压		3.6	V
TA	操作环境温度	-40	+125	°C
TSTG	存储温度	-40	+125	°C

5.2 操作条件

所有的电气特性是适用于下列条件，除非另有说明：

- 操作温度范围：TA = - 40°C 至 125°C 并且结点温度达到 TJ = 100°C。
- 典型值是基于 TA = 25°C 和 VI = 3.3 V 除非另有规定。

5.3 I/O 管脚特征

符号	参数	条件	最小值	典型值	最大值	单位
RPU	上拉电阻		34K	49K	74K	Ω
RPD	下拉电阻		30K	47K	86K	Ω
ILKG	输入漏极电流				±1μ	A
VOL	输出低电平电压				0.4	V
VOH	输出高电平电压		2.4			V
VIL	输入低电平电压		-0.3		0.8	V
VIH	输入高电平电压		2		3.6	V
VT+	施密特触发器低到高的临界点		1.53		1.66	V
VT-	施密特触发器高到低的临界点		1.13		1.27	V
IOL	低电平输出电流	VOL(Max)	Output Drive: 2mA	2.4	3.8	mA
			Output Drive: 4mA	4.7	7.6	
IOH	高电平输出电流	VOH(Min)	Output Drive: 2mA	3.4	7.0	mA
			Output Drive: 4mA	6.9	14.0	

5.4 片上低压差线性稳压器特性

符号	参数	条件	最小值	典型值	最大值	单位
电源范围	V_{DD}		2.0	3.3	3.6	V
带隙基准	V_{REF}		1.19	1.22	1.26	V
输出电流	I_{out}			200		mA
功率消耗	Current	正常模式	13	16.3	26.2	μ A
		Lpr模式	2.5	3.3	4.9	
输出电压	V_{out}	角变化, 18Ω 的负载 $V_{ref}=1.2V$	1.76	1.82	1.87	V
温度系数	T_c	$-40^{\circ}C \sim 125^{\circ}C$	233	396	578	ppm
线性调整率		18Ω 的负载		2.7		%/V
		1800Ω 的负载		0.41		
负载调整率		1mA ~ 100mA		0.0019		%/mA
输出稳定时间	T_s	99%	4	6.9	9.6	ms
阈值电荷泵	V_{thl}	V_{out} 变化		1.776		V
过电压保护阈值	V_{thh}	V_{out} 变化		1.85		V

5.5 锁相环特性

符号	参数	条件	最小值	典型值	最大值	单位
电源范围	V_{DD}		2.0	3.3		MHz
PLL输入时钟	F_{pll_in}			4		
PLL输出时钟	F_{pll_out}		4		72	MHz
PLL锁定时间	T_{lock}				400	μ s
PLL频率转换时间	T_{conv}				100	μ s

5.6 上电复位特性

符号	参数	条件	最小值	典型值	最大值	单位
电源范围	V _{DD}		1.8	3.3	3.6	V
可编程电压检测器 (PVD) 等级选择	VPVD	PLS[2:0]=000(上升沿)	2.15	2.2	2.25	V
		PLS[2:0]=000(下降沿)	2.05	2.1	2.15	V
		PLS[2:0]=001(上升沿)	2.24	2.3	2.36	V
		PLS[2:0]=001(下降沿)	2.15	2.2	2.25	V
		PLS[2:0]=010(上升沿)	2.34	2.4	2.46	V
		PLS[2:0]=010(下降沿)	2.24	2.3	2.36	V
		PLS[2:0]=011(上升沿)	2.44	2.5	2.56	V
		PLS[2:0]=011(下降沿)	2.34	2.4	2.46	V
		PLS[2:0]=100(上升沿)	2.54	2.6	2.66	V
		PLS[2:0]=100(下降沿)	2.44	2.5	2.56	V
		PLS[2:0]=101(上升沿)	2.63	2.7	2.77	V
		PLS[2:0]=101(下降沿)	2.54	2.6	2.66	V
		PLS[2:0]=110(上升沿)	2.73	2.8	2.87	V
		PLS[2:0]=110(下降沿)	2.63	2.7	2.77	V
		PLS[2:0]=111(上升沿)	2.83	2.9	2.97	V
		PLS[2:0]=111(下降沿)	2.73	2.8	2.87	V
PVD迟滞	VPVDhyst			100		mV
上电/掉电复位阈值	VPDR	下降沿	1.85	1.89	1.94	V
		上升沿	1.89	1.93	1.98	V
PDR迟滞	VPDRhyst			50		mV
复位拖延	Trsttempo		1.5	2.2	4.7	mS

5.7 NRST 特性

符号	参数	条件	最小值	典型值	最大值	单位
电源范围	V_{DD}		2.0	3.3	3.6	V
NRST输入低电平电压	$V_{IL(NRST)}^{(1)}$		-0.5		0.8	V
NRST输入高电平电压	$V_{IH(NRST)}^{(1)}$		2		$V_{DD} + 0.5$	
NRST施密特触发电压迟滞	$V_{hys(NRST)}^{(1)}$			200		mV
弱上拉等效电阻 (2)	R_{PU}	$V_{IN} = V_{SS}$	30	40	50	k Ω
NRST输入滤波脉冲	$V_F(NRST)^{(1)}$				100	ns
NRST输入未滤波脉冲	$V_{NF(NRST)}^{(1)}$		300			ns
外部负载电容	CL			0.1		μ F
本机复位时间	TR	$C_L = 0.1\mu$		2.7		ms

5.8 8 MHz 晶振特性

符号	参数	条件	最小值	典型值	最大值	单位
电源范围	V_{DD}		2.0	3.3	3.6	V
振荡器频率	f_{OSC_IN}			8		MHz
反馈电阻	RF			200		k Ω
推荐负载电容与等效串联晶体电阻(RS)	C	$R_s = 30 \Omega$			30	pF
振荡器驱动电流	I_{VDD}	$V_{DD} = 3.3 V$			1	mA
振荡器跨导	g_m	启动	25			mA/V
启动时间	t_{SU}	VDD保持稳定		2		ms
用户外部时钟源频率	f_{HSE_ext}		1	8	25	MHz
OSC_IN输入引脚的高电平电压	V_{HSEH}		$0.7V_{DD}$		V_{DD}	V
OSC_IN输入引脚的低电平电压	V_{HSEL}		V_{SS}		$0.3V_{DD}$	
OSC_IN高电平或低电平时间	$t_{w(HSE)} t_{w(HSE)}$		16			ns
OSC_IN上升或下降时间	$t_{r(HSE)} t_{f(HSE)}$				20	
OSC_IN输入电容	$C_{in(HSE)}$			5		pF
占空比	$DuCy_{(HSE)}$		45		55	%
OSC_IN输入漏电流	I_L	$V_{SS} \leq V_{IN} \leq V_{DD}$			± 1	μ A

5.9 4MHZ RCOSC 特性

符号	参数	条件	最小值	典型值	最大值	单位
电源范围	VDD		2.0	3.3	3.6	V
功率消耗	Current	OSC_4M_A[7:0]= 80h	45	65	90	μA
输出时钟频率	Fout	OSC_4M_A[7:0]= 80h	3.13	4	6.2	MHz
频率调整步进	Fstep			14		KHz
频率调整范围	Frang		-40%		37.5	%
频率温度系数	Fppm	-40°C~125°C		151		ppm/°C
输出启动时间	Tsu				10	CLK
调整稳定时间	Tst				5	CLK

5.10 32 KHZ 晶振

符号	参数	条件	最小值	典型值	最大值	单位
电源范围	VDD		2	3.3	3.6	V
推荐负载电容与等效串联晶体电阻(RS)	CL	RS = 30 kΩ		10		pF
振荡器驱动电流	Ivdd	C3,C2,C1=1,1,0,		400		nA
启动时间	tsu	VDD保持稳定			3	s
用户外部时钟源频率	fLSE_ext	EN=1		32.768		kHz
占空比	DuCy(LSE)		30		70	%
OSC32_IN输入漏电流	IL				±1	μA

5.11 A/D IO 特性

符号	参数	条件	最小值	典型值	最大值	单位
输入低电压	V_{IL}		-0.3		0.8	V
输入高电压	V_{IH}		2		3.6	V
阈值点	V_T		1.33	1.4	1.47	V
施密特触发器从低到高阈值点	V_{T+}		1.53	1.6	1.66	V
施密特触发器高到低阈值点	V_{T-}		1.13	1.2	1.27	V
输入漏电流	I_I	$V_I=3.3V$ or $0V$			$\pm 1\mu$	A
三态输出漏电流	I_{OZ}	$V_O=3.3V$ or $0V$			$\pm 1\mu$	A
上拉电阻	R_{PU}		34K	49K	74K	Ω
下拉电阻	R_{PD}		30K	47K	86K	Ω
输出低电压	V_{OL}				0.4	V
输出高电压	V_{OH}		2.4			V
低电平输出电流	I_{OL}	$V_{OL(max)}, 02mA$	2.4	3.8	5.3	mA
		$V_{OL(max)}, 04mA$	4.7	7.6	10.6	
高电平输出电流	I_{OH}	$V_{OH(min)}, 02mA$	3.4	7.0	11.6	mA
		$V_{OH(min)}, 04mA$	6.9	14.0	23.2	

5.12 温度传感器特性

符号	参数	条件	最小值	典型值	最大值	单位
电源范围	V_{DD}		2	3.3	3.63	V
电流消耗	I_{DD}		70	100	160	μA
待机电流	I_{STBY}	EN = LOW			1	μA
温度范围	T_A		-40		125	$^{\circ}C$
输出电压	V_O	$T_A = -40^{\circ}C$	0.85	0.86	0.87	V
		$T_A = +125^{\circ}C$	1.44	1.45	1.46	V
启动时间	t_{start}				50	μs

5.13 ADC + PGA 特性

5.13.1 推荐操作条件

符号	参数	条件	最小值	典型值	最大值	单位
电源电压	VDD		2.4	3.3	3.63	V
电源电流 (ADC + PGA)	Idd			3.5		mA
时钟频率	Fclk			20	48	MHz
操作温度	Ta		-40		125	°C
模拟条件	Tj		-40		125	°C

5.13.2 直流精度

符号	参数	条件	最小值	典型值	最大值	单位
分辨率	Res				12	Bit
ENOB	ENOB		9.2	10.5	11.8	
积分非线性	INL				+/-5	LSB
偏移误差	Eoff				5	%
增益误差	Egain				5	%

5.13.3 时序规范 (参考时序图)

符号	参数	条件	最小值	典型值	最大值	单位
时钟占空比	CLK_DUTY		45		55	%
ADCEN设置时间到 START"L"→"H"	t1		200			ns
ADSEL设置时间到 START"L"→"H"	t2		20			ns
ADCOUT保持时间到 START"L"→"H"	t3		20			ns
转换时间 (注释1)	Tc		1.12			μs
PGA稳定时间	Tsettleing	Gain=4		3		μs

5.13.4 PGA 特性

符号	参数	条件	最小值	典型值	最大值	单位
电压摆动(Vpp)	Vswin		0.3		3	V
PGA频宽	BW1	GAIN=8, -3dB Non-inverting		12		MHz
	BW2	GAIN=7, -3dB Inverting		8.5		
总谐波失真	Thd	Fin=100khz		0.1		%

5.14 比较器特性

符号	参数	条件	最小值	典型值	最大值	单位
电源范围	VDDH		2.0	3.3	3.6	V
电流消耗	IDD	CAON<3:0> = 1111		200		μA
待机电流	ISTBY	CAON<3:0> = 0000			1	μA
传播延迟	T _{PGD}			10		ns
开始时间*	TS				5	μs

5.15 RCOSC_32K 特性

符号	参数	条件	最小值	典型值	最大值	单位
电源范围	V _{DD}		2.0	3.3	3.6	V
功率消耗	Current	Corner Change	0.30	0.55	0.89	uA
输出时钟频率	F _{out}	Corner Change	28	32	53	kHz
输出启动时间	T _{su}	Corner Change		10		CLK
输出频率随机误差	σ(F _{out})		1.04		2.05	%
频率温度系数	Fppm	-40°C~125°C		2097		ppm/°C

5.16 功率消耗表

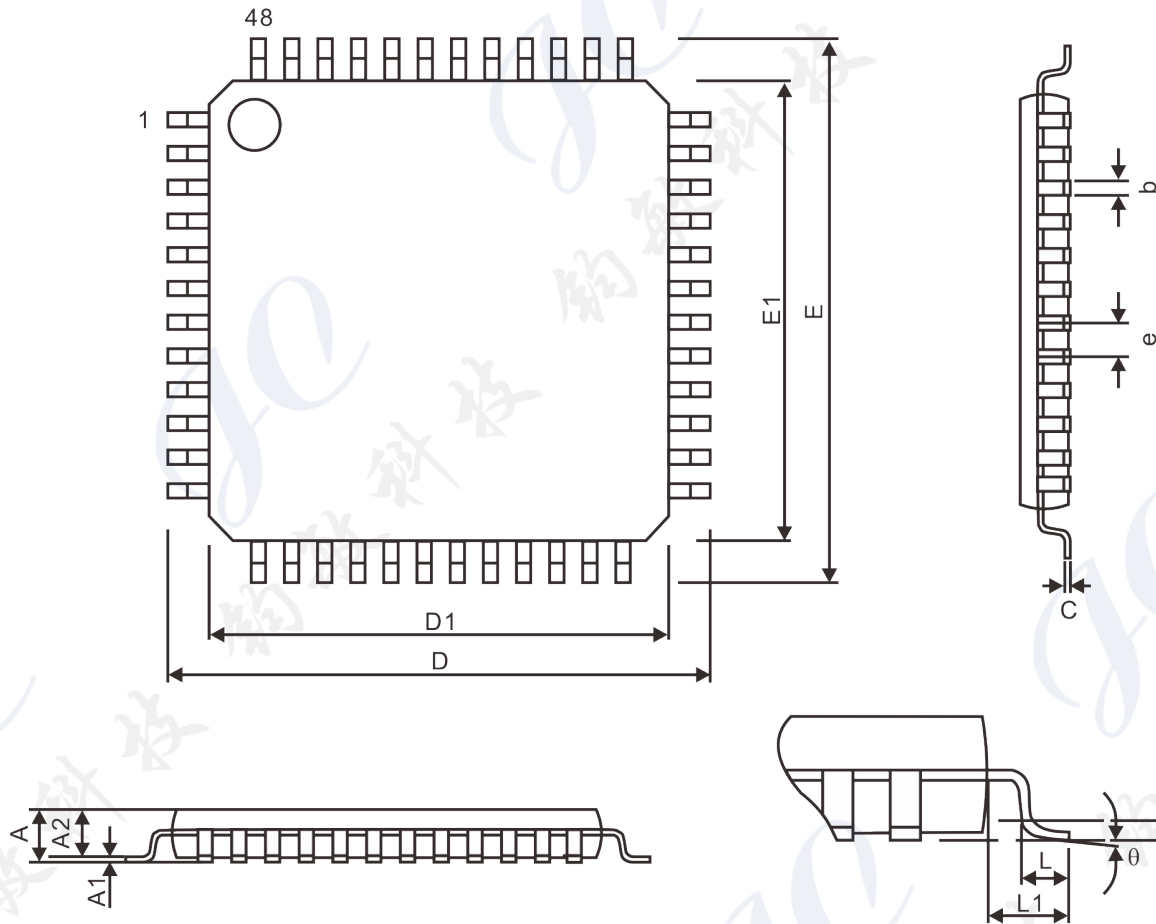
该表展示了三种不同模式下的功耗数据：待机模式、睡眠模式和正常模式。

Clock Enable	Frequency	Normal		Sleep Mode, LDO ON		Unit
		All IP Clock ON	All IP Clock OFF	All IP Clock ON	All IP Clock OFF	
LSI+HSI		1.84 1.82	1.285 1.276	1.313 1.309	0.813 0.81	mA
LSE+HSI		1.83 1.81	1.288 1.276	1.312 1.309	0.815 0.813	mA
HSI		1.82 1.80	1.288 1.277	1.302 1.309	0.815 0.812	mA
HSE+HSI		2.22 2.19	1.775 1.756	1.793 1.791	1.2 1.199	mA
PLL+HSI	4 MHz	2.43 2.40	0.1763 0.1958	1.932 1.928	1.44 1.43	mA
	8 MHz	2.48 2.47	2.456 2.448	2.525 2.522	1.519 1.516	mA
	16 MHz	5.65 5.63	3.585 3.576	3.696 3.681	1.697 1.694	mA
	32 MHz	8.87 8.83	4.824 4.807	6.037 6.014	2.301 2.027	mA
	48 MHz	11.87 11.81	5.736 5.695	8.256 8.218	2.265 2.263	mA
PLL+HSE	4 MHz	2.76 2.73	2.328 2.308	2.338 2.334	1.735 1.734	mA
	8 MHz	3.845 3.833	2.916 2.903	2.853 2.848	1.83 1.826	mA
	16 MHz	6.043 6.032	4.032 4.026	4.056 4.047	2.007 2.003	mA
	32 MHz	9.3829 9.3759	5.322 5.303	6.446 6.428	2.372 2.361	mA
	48 MHz	12.345 12.296	6.309 6.289	8.706 8.695	2.624 2.62	mA

Clock enable	Standby Mode, LDO OFF			unit
	RTC enable	RTC disable	All clock off	
LSI	3.2 3.1	3.1 3.0	2.5 2.4 (LSI and LSE disable)	uA
LSE	4.5 4.4	4.5 4.4		uA

6. 封裝信息

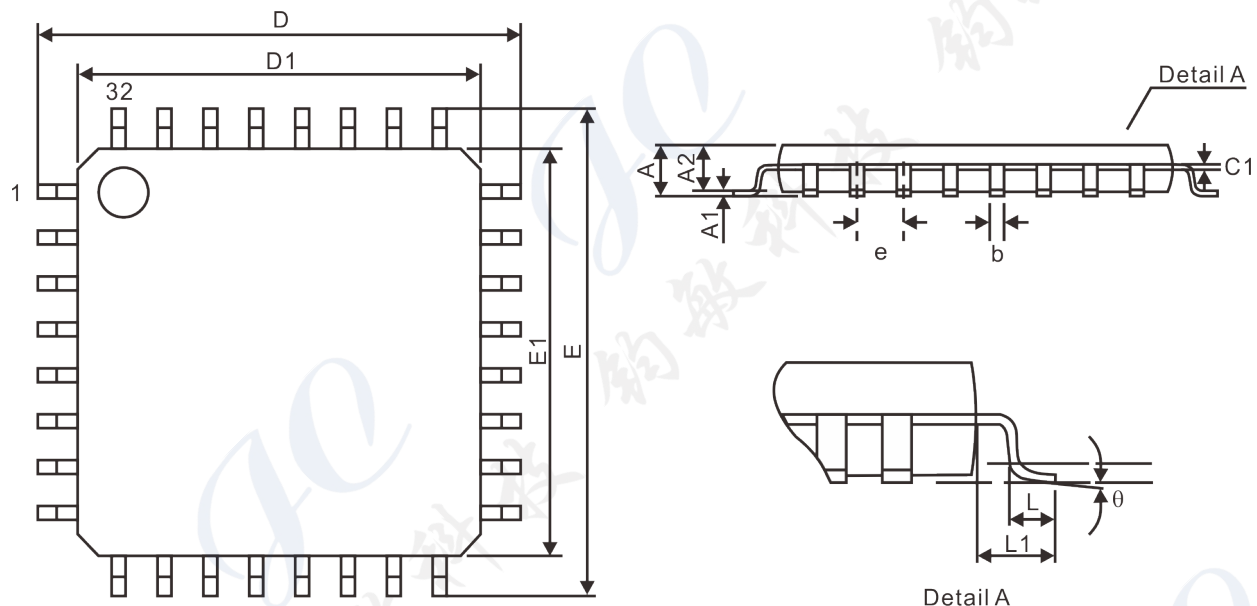
48-PIN, LQFP, 7X7



Symbol	Dimensions (mm)		
	Min.	Nom.	Max.
A	-	-	1.60
A1	0.05	-	0.15
A2	1.35	1.40	1.45
b	0.17	0.22	0.27
c	0.09	-	0.20
D	9.00 BSC		
D1	7.00 BSC		
E	9.00 BSC		
E1	7.00 BSC		
e	0.50 BSC		
L	0.45	0.60	0.75
L1	1.00 REF		
θ	0°	3.5°	7°

Note: Refer to JEDEC MS-026 BBC

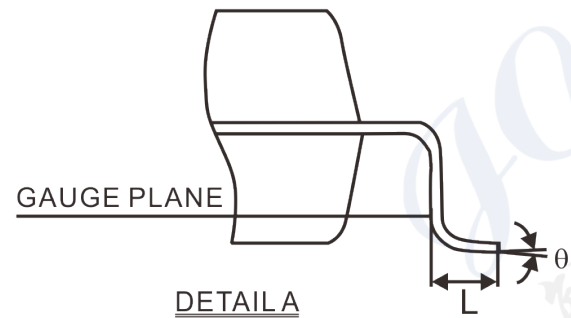
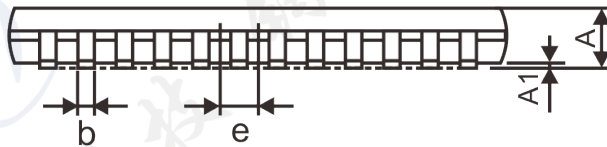
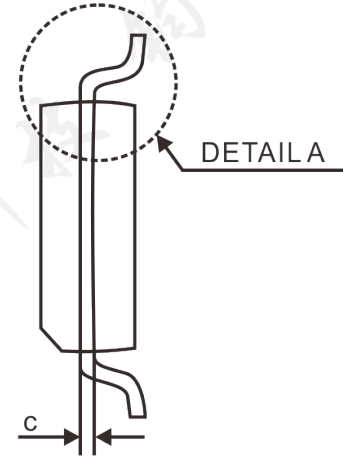
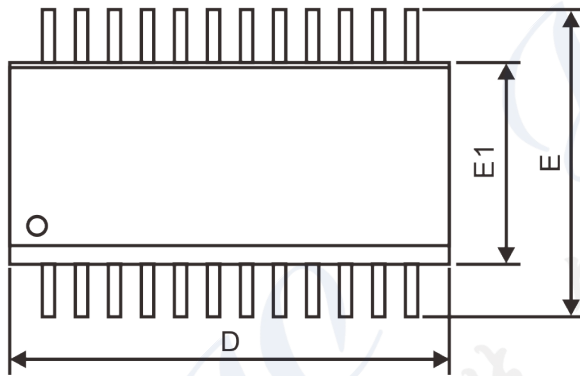
32-PIN, LQFP, 7X7



Symbol	Dimensions (mm)		
	Min.	Nom.	Max.
A	-	-	1.60
A1	0.05	-	0.15
A2	1.35	1.40	1.45
b	0.30	0.37	0.45
c1	0.09	-	0.20
D	9.00 BSC		
D1	7.00 BSC		
E	9.00 BSC		
E1	7.00 BSC		
e	0.80 BSC		
L	0.45	0.60	0.75
L1	1.00 REF		
θ	0°	3.5°	7°

Note: Refer to JEDEC MS-026 BBC

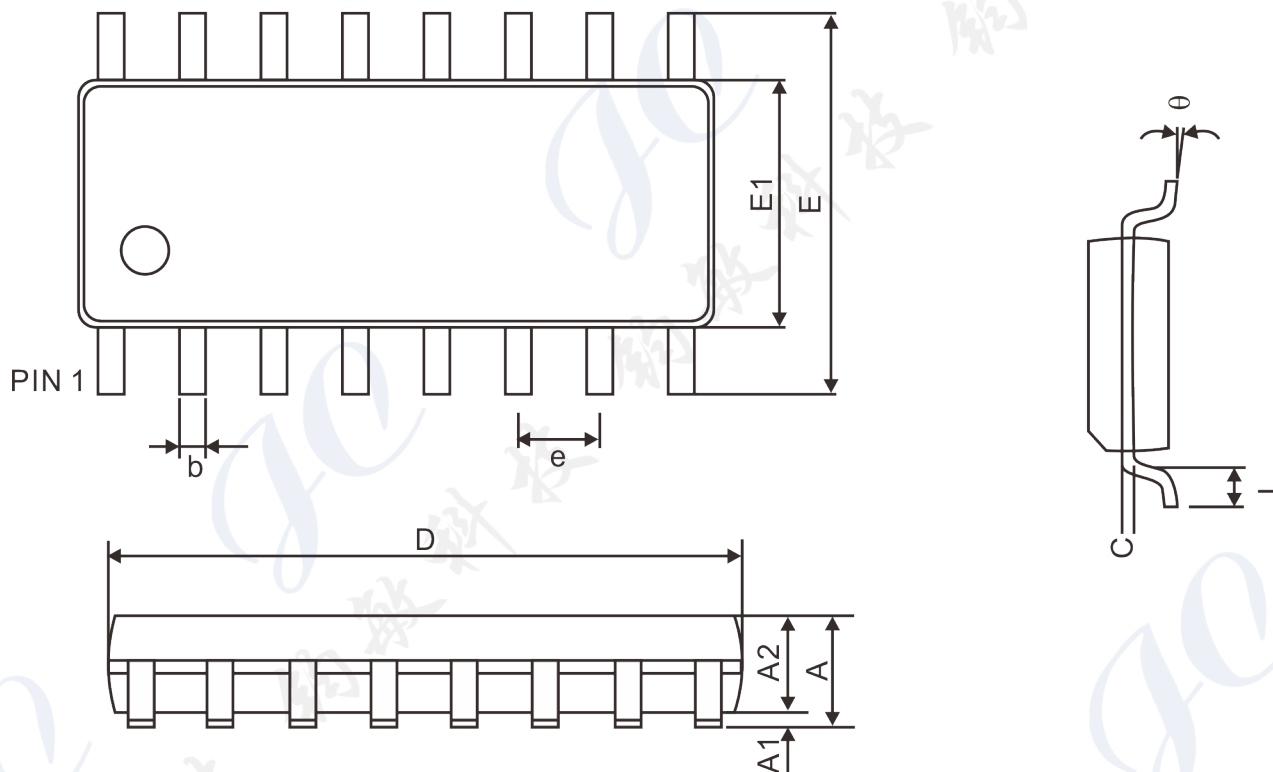
24-PIN, SSOP



Symbol	Dimensions (mm)		
	Min.	Nom.	Max.
A	-	-	1.750
A1	0.100	-	0.250
b	0.200	-	0.300
c	0.100	-	0.250
e	0.635 BSC		
D	8.66 BSC		
E	5.99 BSC		
E1	3.91 BSC		
L	0.410	-	1.270
theta	0°	-	8°

Note: Refer to JEDEC MO-137 AE

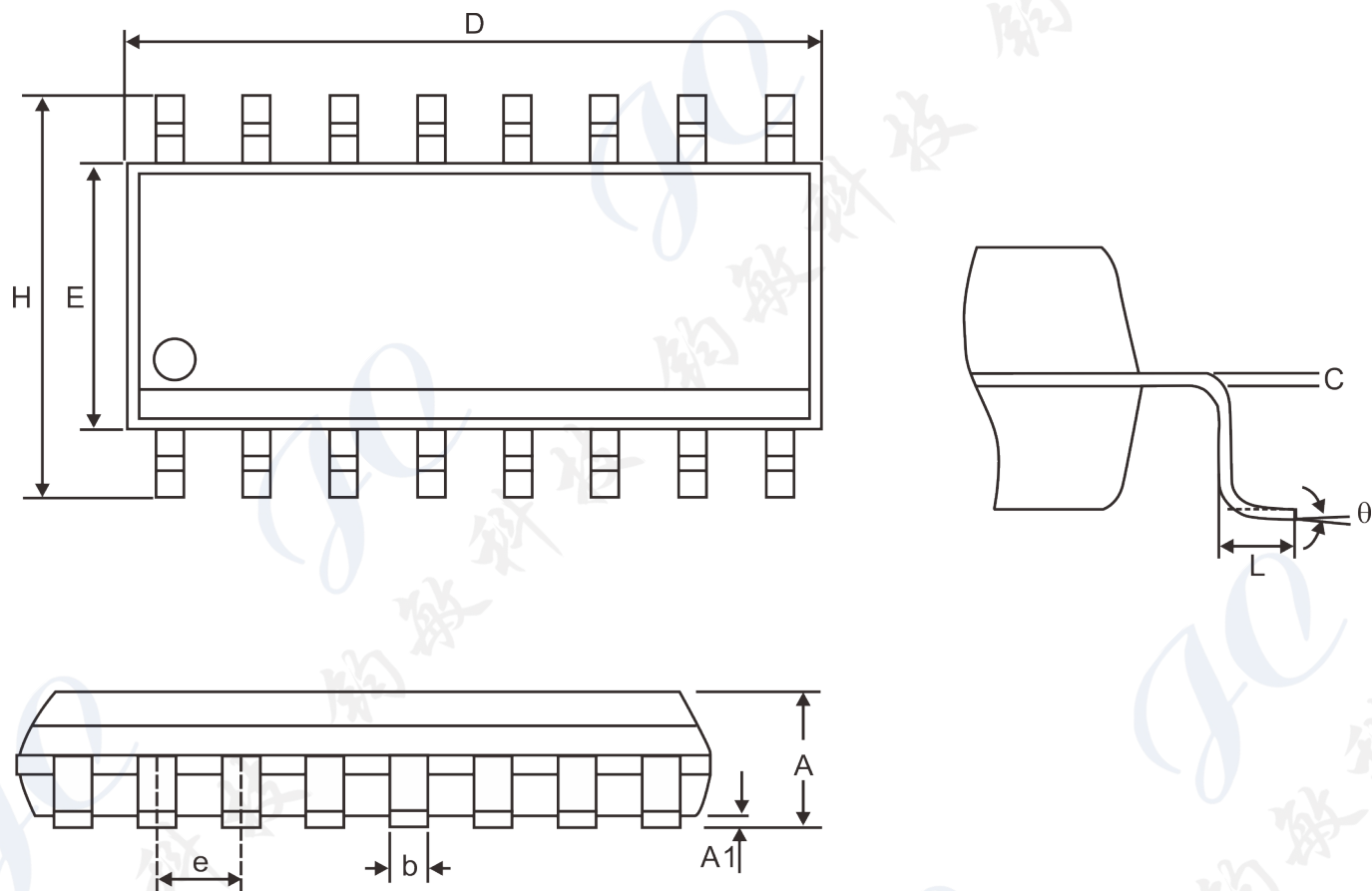
16-PIN, SOP, 300 MIL



Symbol	Dimensions (mm)		
	Min.	Nom.	Max.
A	-	-	2.65
A1	0.10	-	0.30
A2	2.05	-	-
b	0.31	-	0.51
c	0.18	-	0.33
e	1.27 BSC		
D	10.3 BSC		
E	10.3 BSC		
E1	7.5BSC		
L	0.3	-	1.27
θ	0°	-	8°

Note: Refer to JEDEC MS-013 AA

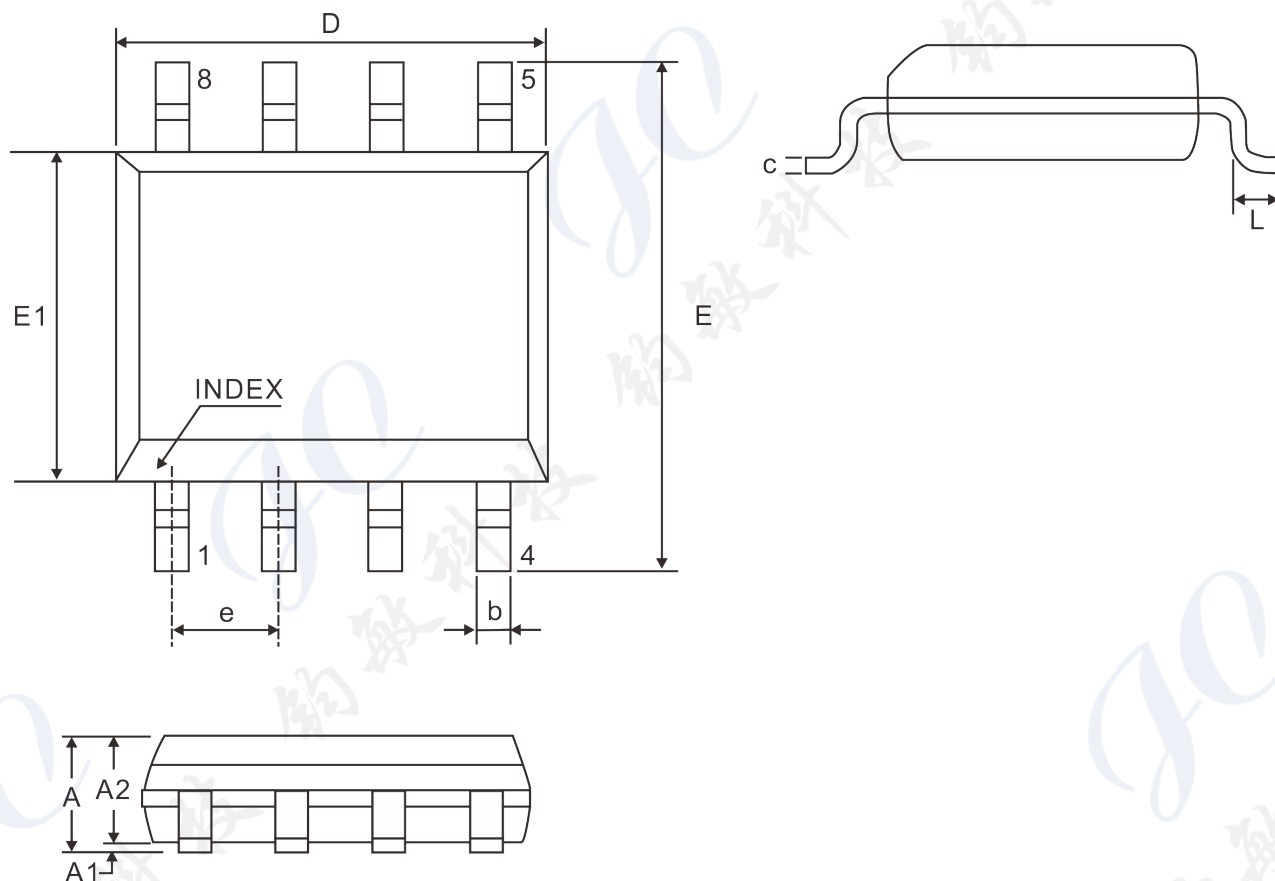
16-PIN, SOP, 150 MIL



Symbol	Dimensions (mm)		
	Min.	Nom.	Max.
A	1.35	-	1.75
A1	0.10	-	0.25
b	0.31	-	0.51
c	0.10	-	0.25
e	1.27 BSC		
D	9.90 BSC		
H	6.00 BSC		
E	3.90BSC		
L	0.45	-	1.27
θ	0°	-	8°

Note: Refer to JEDEC MS-012 AC

8-PIN, SOP, 150 MIL



Symbol	Dimensions (mm)		
	Min.	Nom.	Max.
A	-	-	1.75
A1	0.10	-	0.25
A2	1.25	-	-
b	0.31	-	0.51
c	0.10	-	0.25
D	4.90 BSC		
E	6.00 BSC		
E1	3.90 BSC		
e	1.27 BSC		
L	0.40	-	1.27
θ	0°	-	8°

Note: Refer to JEDEC MS-012 AA

IMPORTANT NOTICE

Princeton Technology Corporation (PTC) reserves the right to make corrections, modifications, enhancements, improvements, and other changes to its products and to discontinue any product without notice at any time.

PTC cannot assume responsibility for use of any circuitry other than circuitry entirely embodied in a PTC product. No circuit patent licenses are implied.

Princeton Technology Corp.
2F, 233-1, Baociao Road,
Sindian Dist., New Taipei City 23145, Taiwan
Tel : 886-2-66296288
Fax: 886-2-29174598
<http://www.princeton.com.tw>