User
Manual

ZSN700

智能读写卡芯片 (二次开发)

UM01010700 1.1.00 Date:2020/12/18

类别	内容
关键词	ZSN700 产品用户手册
摘要	

ZLG 致远微电子 ©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd

修订历史

版本	日期	内容
1.0.00	2020/02/18	初版;
1.1.00	2020/12/18	更新模板;

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

目录

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

1 文中缩写

1.1 寄存器描述表中使用的缩写列表

在对寄存器的描述中使用了下表中缩写：

表 3. 寄存器表中缩写

缩写	含义
read/write(rw)	软件能读写此位。
read-only(r)	软件只能读此位。
write-only(w)	软件只能写此位，读此位将返回复位值。
read/clear(rc_w1)	软件可以读此位，也可以通过写 1 清除此位，写 0 对此位无影响。
read/clear(rc_w0)	软件可以读此位，也可以通过写 0 清除此位，写 1 对此位无影响。
read/clear by read(rc_r)	软件可以读此位；读此位将自动地清除它为 0，写 0 对此位无影响。
read/set(rs)	软件可以读也可以设置此位，写 0 对此位无影响。
read-only write trigger(rt_w)	软可以读此位；写 0 或 1 触发一个事件但对此位数值没有影响。
toggle(t)	软件只能通过写 1 来翻转此位，写 0 对此位无影响。
Reserved(Res.)	保留位，必须保持默认值不变。

1.2 术语表

本节给出本文档涉及到的部分缩写词的一个简洁定义：

- SWD：为 Serial Wire Debug 的首字母缩写。其是 CPU 内核集成的一个调试口，是基于 SWD 协议的 2 线调试接口。
- Word：字，32 位长的数据或指令长度。
- Half word：半字，16 位长的数据或指令长度。
- Byte：字节，8 位数据长度。
- IAP：In-Application Programming 的首字母缩写。直译为在应用编程，即用户程序可以更新自身的程序，从而达到应用升级。
- ICP：In-Circuit Programming 的首字母缩写。直译为在电路编程，即用户可通过应用板上的 JTAG 口或 SWD 口对 MCU 的 Flash 进行编程。
- Option bytes：选项字节，保存在 Flash 中的 MCU 配置字节。
- OBL：Option Byte Loader 的首字母缩写，选项字节装载器。
- AHB：Advanced High-performance Bus 的首字母缩写，直译为先进高性能总线。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

2 存储器和总线构架

2.1 系统构架

主系统由以下部分构成：

- 2 个 AHB 总线 Master：
  - Cortex-M0+
  - DMA 控制器
- 4 个 AHB 总线 Slaves：
  - FLASH 存储器
  - SRAM 存储器
  - AHB0，AHB to APB Bridge，包含所有 APB 接口外设
  - AHB1，包含所有 AHB 接口外设

整个系统总线结构采用多层次 AHB-lite 总线互连实现：如图 1所示：

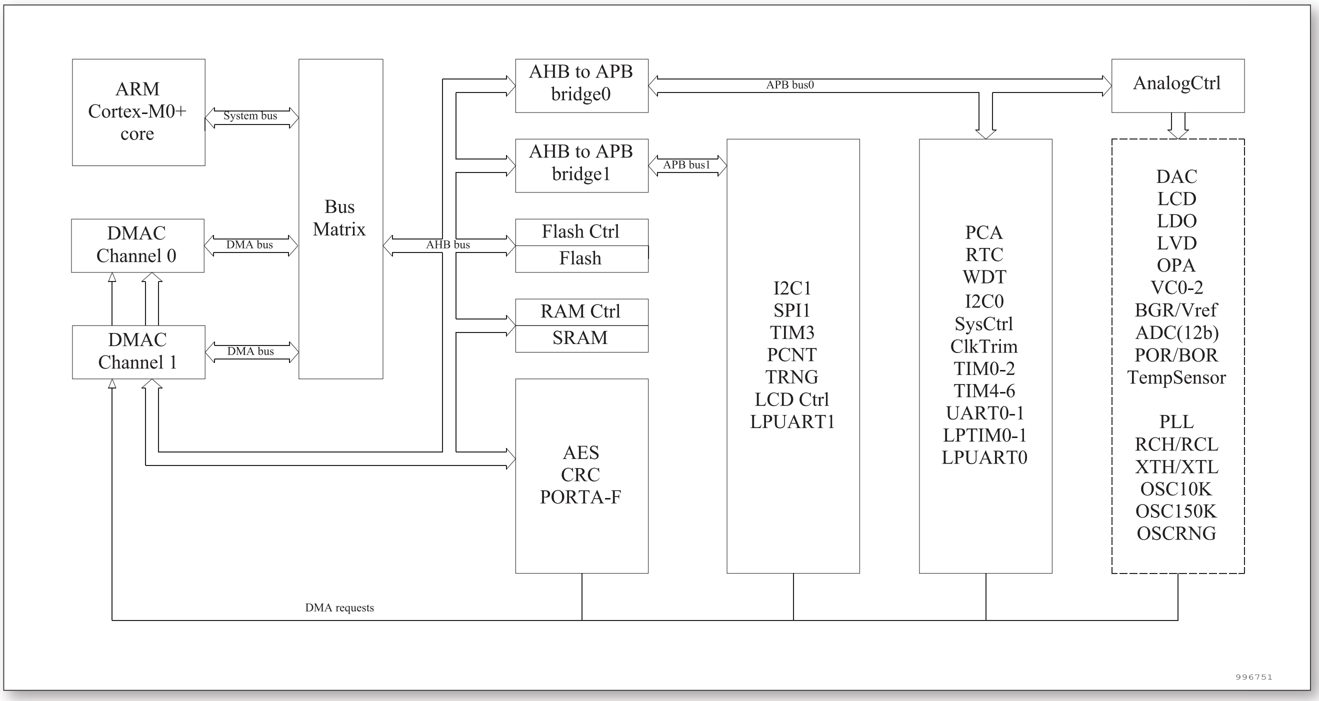

图 1. 系统架构

2.2 系统地址划分

整个系统的地址区域划分，如图 2所示。

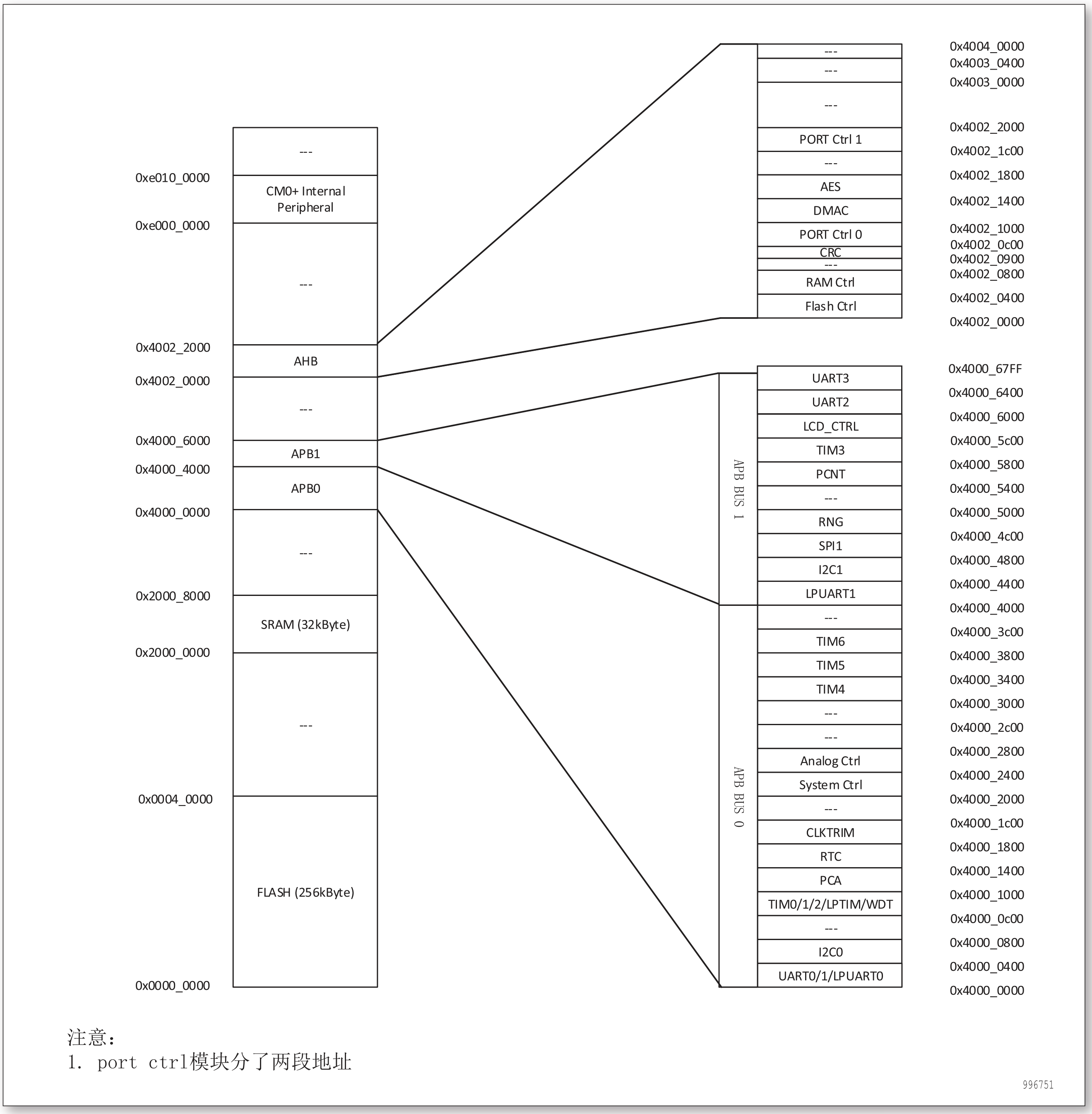

图 2. 地址区域划分示意图

2.3 存储器和模块地址分配

表 4: 存储器映像

总线	编址范围	大小	外设	备注
Flash	0x0000 0000 –0x0003 FFFF	256 KB	主闪存存储器，系统存储器	
	0x0004 0000 –0x1FFF FFFF	-	Reserved	
SRAM	0x2000 0000 –0x2000 7FFF	32 KB	SRAM	
	0x2000 8000 –0x3FFF FFFF	-	Reserved	

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

总线	编址范围	大小	外设	备注
APB0	0x4000 0000 −0x4000 00FF	256 Byte	UART0	
	0x4000 0100 −0x4000 01FF	256 Byte	UART1	
	0x4000 0200 −0x4000 02FF	256 Byte	LPUART0	
	0x4000 0300 −0x4000 03FF	-	Reserved	
	0x4000 0400 −0x4000 07FF	1 kByte	I2C0	
	0x4000 0800 −0x4000 0BFF	1 kByte	-	
	0x4000 0C00 −0x4000 0CFF	256 Byte	TIM0	
	0x4000 0D00 −0x4000 0DFF	256 Byte	TIM1	
	0x4000 0E00 −0x4000 0EFF	256 Byte	TIM2	
	0x4000 0F00 −0x4000 0F7F	128 Byte	LPTIM	
	0x4000 0F80 −0x4000 0FFF	128 Byte	WDT	
	0x4000 1000 −0x4000 13FF	1 kByte	PCA	
	0x4000 1400 −0x4000 17FF	1 kByte	RTC	
	0x4000 1800 −0x4000 1BFF	1 kByte	CLKTRIM	
	0x4000 1C00 −0x4000 1FFF	-	Reserved	
	0x4000 2000 −0x4000 23FF	1 kByte	SYSCTRL	
	0x4000 2400 −0x4000 27FF	1 kByte	ANALOGCTRL	
	0x4000 2800 −0x4000 2BFF	-	Reserved	
	0x4000 2C00 −0x4000 2FFF	-	Reserved	
	0x4000 3000 −0x4000 33FF	1 kByte	TIM4	
	0x4000 3400 −0x4000 37FF	1 kByte	TIM5	
	0x4000 3800 −0x4000 3BFF	1 kByte	TIM6	
	0x4000 3C00 −0x4000 3FFF	-	Reserved	
APB1	0x4000 4000 −0x4000 43FF	1 kByte	LPUART1	
	0x4000 4400 −0x4000 47FF	1 kByte	I2C1	
	0x4000 4800 −0x4000 4BFF	1 kByte	SPI1	
	0x4000 4C00 −0x4000 4FFF	1 kByte	TRNG	
	0x4000 5000 −0x4000 53FF	-	Reserved	
	0x4000 5400 −0x4000 57FF	1 kByte	PCNT	
	0x4000 5800 −0x4000 5BFF	1 kByte	TIM3	
	0x4000 5C00 −0x4000 5FFF	1 kByte	LCD	
	0x4000 6000 −0x4000 63FF	1 kByte	UART2	
	0x4000 6400 −0x4000 67FF	1 kByte	UART3	
	0x4000 6800 - 0x4001 FFFF	-	Reserved	
AHB	0x4002 0000 - 0x4002 03FF	1 kByte	FLASH CTRL	
	0x4002 0400 - 0x4002 07FF	1 kByte	RAM CTRL	
	0x4002 0800 - 0x4002 08FF	-	Reserved	
	0x4002 0900 - 0x4002 0BFF	768 Byte	CRC	
	0x4002 0C00 - 0x4002 0FFF	1 kByte	PORT CTRL	
	0x4002 1000 - 0x4002 13FF	1 kByte	DMAC	

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

总线	编址范围	大小	外设	备注
AHB	0x4002 1400 - 0x4002 17FF	1 kByte	AES	
	0x4002 1800 - 0x4002 1BFF	-	Reserved	
	0x4002 1C00 - 0x4002 1FFF	1 kByte	PORT CTRL1	

3 工作模式

本产品的电源管理模块负责管理本产品各种工作模式之间的切换，以及控制各工作模式下的各功能模块的工作状态。本产品的工作电压（VCC）为 1.8v ~ 3.7v。

本产品有如下几个工作模式：

- 运行模式：CPU 运行，周边功能模块运行。
- 休眠模式：CPU 停止运行，周边功能模块运行。
- 深度休眠模式：CPU 停止运行，高速时钟停止运行。

从运行模式，通过执行软件程序，可进入其他低功耗模式。从其他各种低功耗模式，通过中断触发，可回到运行模式。

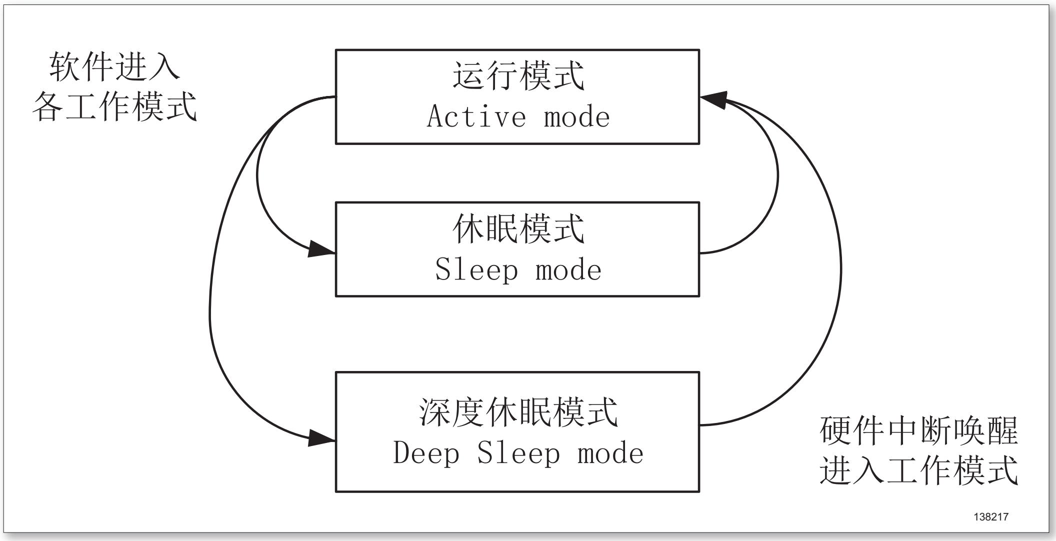

图 3. 控制模式框图

在各模式下，CPU 可响应所有的中断类型详见表 5。

表 5: CPU 可响应中断类型

标号	中断来源	运行模式	休眠模式	深度休眠模式
[0]	GPIO_PA	✓	✓	✓
[1]	GPIO_PB	✓	✓	✓
[2]	GPIO_PC/GPIO_PE	✓	✓	✓
[3]	GPIO_PD/GPIO_PF	✓	✓	✓
[4]	DMAC	✓	✓	
[5]	TIM3	✓	✓	
[6]	UART0	✓	✓	

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

标号	中断来源	运行模式	休眠模式	深度休眠模式
[7]	UART1	√	√	
[8]	LPUART0	√	√	√
[9]	LPUART1	√	√	√
[10]	-	√	√	
[11]	SPI1	√	√	
[12]	I2C0	√	√	
[13]	I2C1	√	√	
[14]	TIM0	√	√	
[15]	TIM1	√	√	
[16]	TIM2	√	√	
[17]	LPTIM0/LPTIM1	√	√	√
[18]	TIM4	√	√	
[19]	TIM5	√	√	
[20]	TIM6	√	√	
[21]	PCA	√	√	
[22]	WDT	√	√	√
[23]	RTC	√	√	√
[24]	ADC	√	√	
[25]	PCNT	√	√	√
[26]	VC0	√	√	√
[27]	VC1/VC2	√	√	√
[28]	LVD	√	√	√
[29]	LCD	√	√	√
[30]	FLASH/RAM	√	√	
[31]	CLKTRIM	√	√	√

在各模式下，本产品可响应所有的复位类型详见表 6。

表 6: 可响应复位类型

标号	复位源	运行模式	休眠模式	深度休眠模式
[0]	上电掉电复位 POR	√	√	√
[1]	外部 Reset Pin 复位	√	√	√
[2]	LVD 复位	√	√	√
[3]	WDT 复位	√	√	√
[4]	PCA 复位	√	√	
[5]	Cortex-M0+ LOCKUP 硬件复位	√		

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

标号	复位源	运行模式	休眠模式	深度休眠模式
[6]	Cortex-M0+ SYSRESETREQ 软件复位	✓		

3.1 运行模式

本产品运行模式（Active Mode）：

在系统在电源上电复位后，或从各低功耗唤醒后，微控制器 MCU 处于运行状态。当 CPU 不需继续运行时，可以利用多种低功耗模式来节能，例如等待某个外部事件时。用户需要根据最低能耗、最快速启动时间、可用的唤醒源等条件，选定一个最佳的低功耗模式。

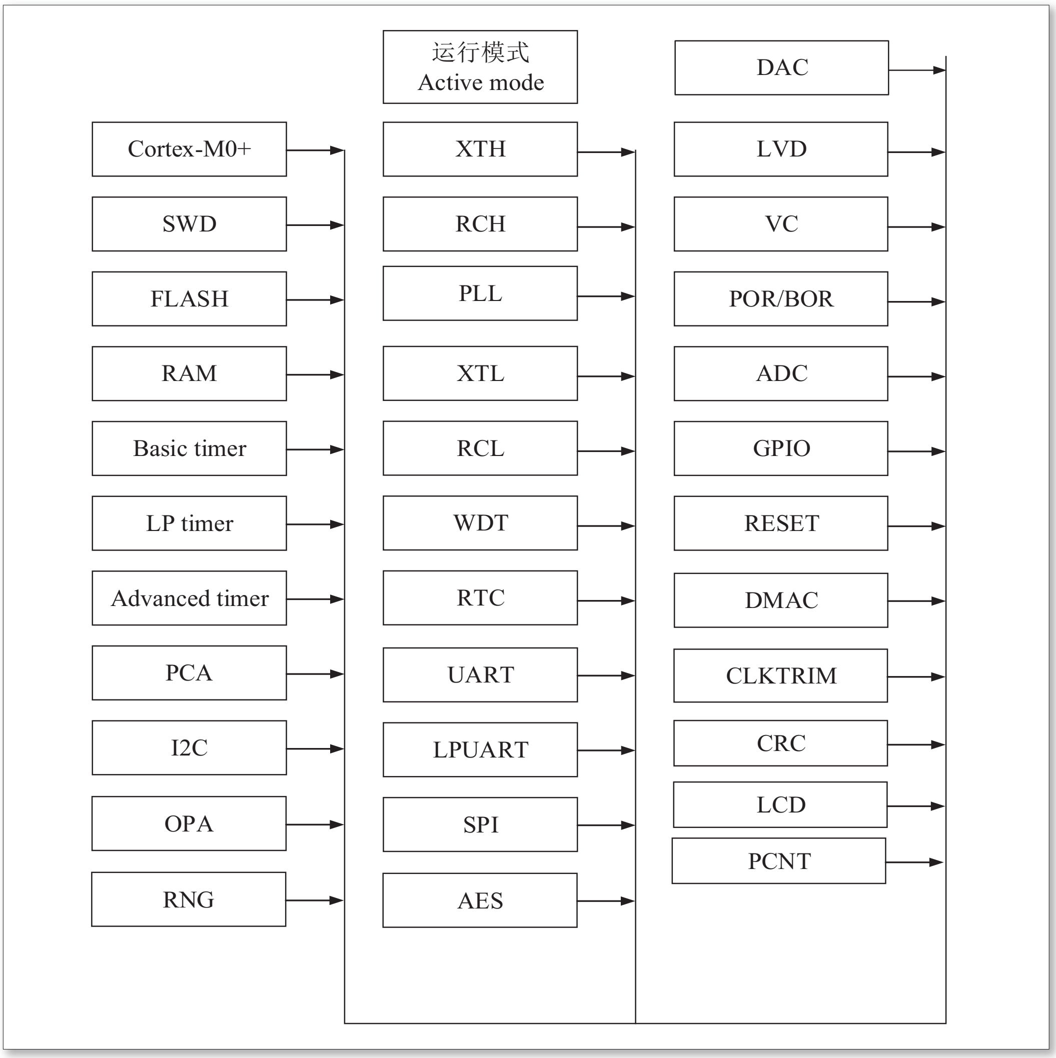

图 4. 运行模式下可运行模块图

几种降低运行模式下芯片功耗的方法：

1. 在运行模式下，通过对预分频寄存器（SYSCTRL0.HCLK_PRS, SYSCTRL0.PCLK_PRS）进行编程，可以降低任意一个系统时钟 (HCLK，PCLK) 的速度。进入睡眠模式前，也可以利用预分频器来降低外设的时钟。
2. 在运行模式下，关闭不使用外设的时钟（PERIx_CLKx）来减少功耗。

3. 在运行模式下，关闭不使用外设的时钟（PERIx_CLKx）来减少功耗，并让系统进入睡眠模式下更多地减少功耗，并在执行 WFI 指令前关闭不使用外设的时钟（PERIx_CLKEN.x）。
4. 使用低功耗模式代替休眠模式，因为本产品的唤醒时间极短（~4us），亦可满足系统的实时响应的需求。

3.2 休眠模式

本产品休眠模式（Sleep Mode）

使用 WFI 指令可以进入休眠模式，休眠模式下，CPU 停止运行，但时钟模块、系统时钟、NVIC 中断处理以及周边的功能模块仍都可以工作。

系统进入休眠状态，不会改变端口状态，在进入休眠前根据需要更改 IO 的状态为休眠下的状态。

如何进入休眠模式：

通过执行 WFI 指令进入睡眠状态。根据 Cortex-M0+ 系统控制寄存器中的 SLEEPONEXIT 位的值，有两种选项可用于选择睡眠模式进入机制：

- SLEEP-NOW：如果 SLEEPONEXIT 位被清除，当 WFI 或 WFE 被执行时，微控制器立即进入睡眠模式。
- SLEEP-ON-EXIT：如果 SLEEPONEXIT 位被置位，系统从最低优先级的中断处理程序中退出时，微控制器就立即进入睡眠模式。

如何退出休眠模式：

如果执行 WFI 指令进入睡眠模式，任意一个高优先级嵌套向量中断控制器响应的外设中断都能将系统从睡眠模式唤醒。

使用注意：

1. SLEEP-ON-EXIT 该位置 1，执行完中断自动进入 sleep，程序不需要写 __wfi()。
2. SLEEP-ON-EXIT 该位清 0，main() 执行 __wfi() 后进入 sleeping，中断触发且执行完中断程序返回 main() 后，执行 WFI 指令后进入 sleeping。等待后续中断触发。
3. SLEEP-ON-EXIT 位不影响 __wfi() 指令的执行。SLEEP-ON-EXIT =0：main() 执行 wfi() 后进入 sleeping，中断触发且执行完中断程序返回 main() 后，继续往下执行。
4. 若在中断中进入 sleep，只有优先级高于此中断的中断才能唤醒，先执行高优先级，再执行低优先级；优先级低于或等于此中断的中断不能唤醒。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

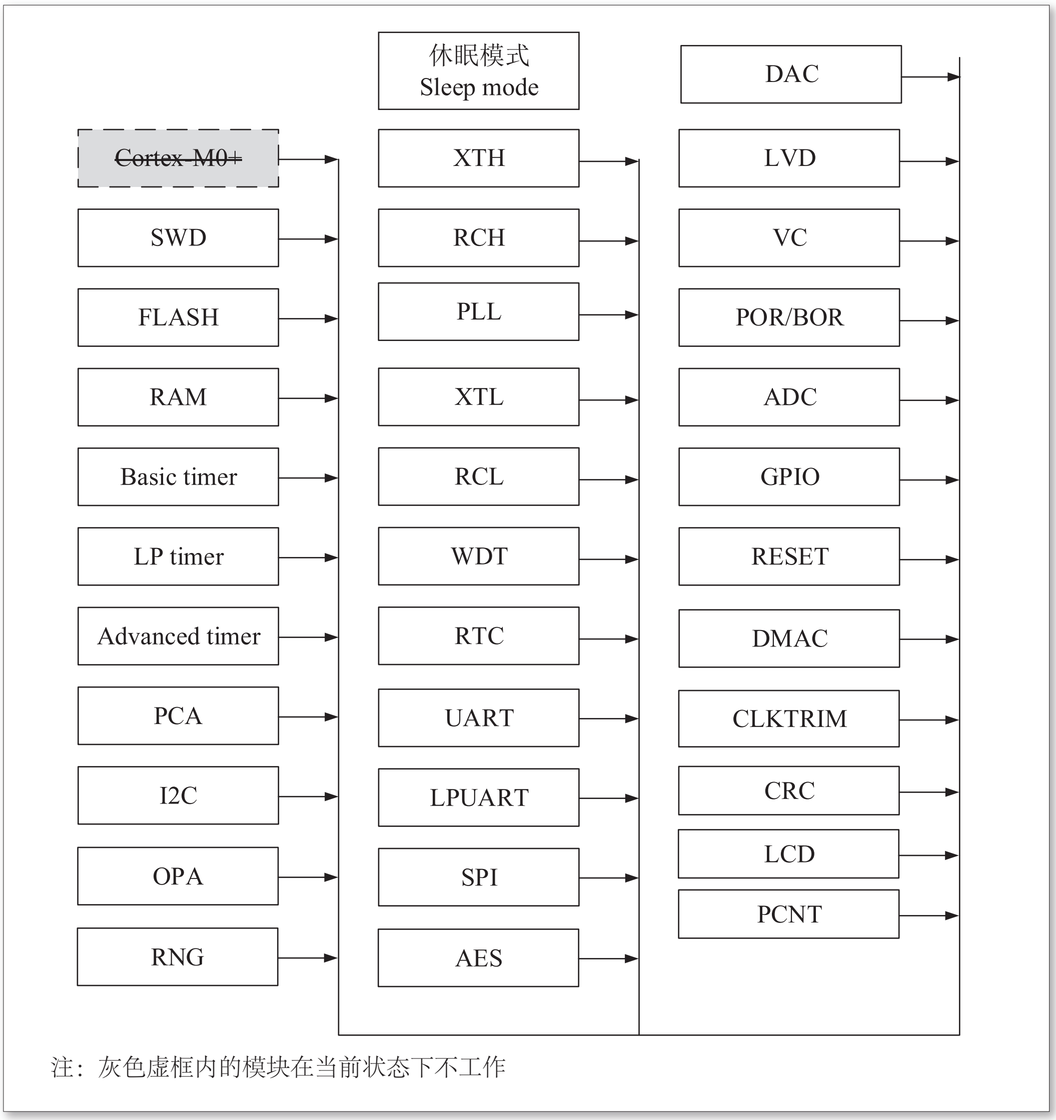

图 5. 休眠模式下可运行模块图

3.3 深度休眠模式

本产品深度休眠模式（Deep Sleep Mode）：

使用 SLEEPDEEP 配合 WFI 指令可以进入深度休眠模式，在深度休眠模式下，CPU 停止运行，高速时钟关闭，低速时钟可配置是否运行，部分低功耗的周边模块可配置是否允许，NVIC 中断处理仍可以工作。

- 系统从高速时钟进入深度休眠模式，高速时钟自动关闭，低速时钟保持进入深度睡眠前的状态。
- 系统从低速时钟进入深度休眠模式，由于低速时钟不会自动关闭，保持运行，进入休眠模式。只有 ARM Cortex-M0+ 不运行，其他模块都运行。
- 系统时钟切换时，所有时钟都不会自动关闭，需要根据功耗及系统需求软件关闭打开相应的时钟。
- 系统进入深度休眠状态，不会改变端口状态，在进入休眠前根据需要更改 IO 的状态为休眠下的状态。

如何进入深度休眠模式：

首先设置 Cortex-M0+ 系统控制寄存器中的 SLEEPDEEP 位，通过执行 WFI 指令进入睡眠状态。根据 Cortex-M0+ 系统控制寄存器中的 SLEEPONEXIT 位的值，有两种选项可用于选择深度睡眠模式进入机制：

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

- SLEEP-NOW：如果 SLEEPONEXIT 位被清除，当 WFI 或 WFE 被执行时，微控制器立即进入睡眠模式。
- SLEEP-ON-EXIT：如果 SLEEPONEXIT 位被置位，系统从最低优先级的中断处理程序中退出时，微控制器就立即进入睡眠模式。

如何退出深度休眠模式：

如果执行 WFI 指令进入睡眠模式，任意一个被嵌套向量中断控制器响应的外设中断（Deep Sleep 下可运行的周边模块中断）都能将系统从睡眠模式唤醒。

唤醒设置参考 4.4 中断唤醒控制 WIC。

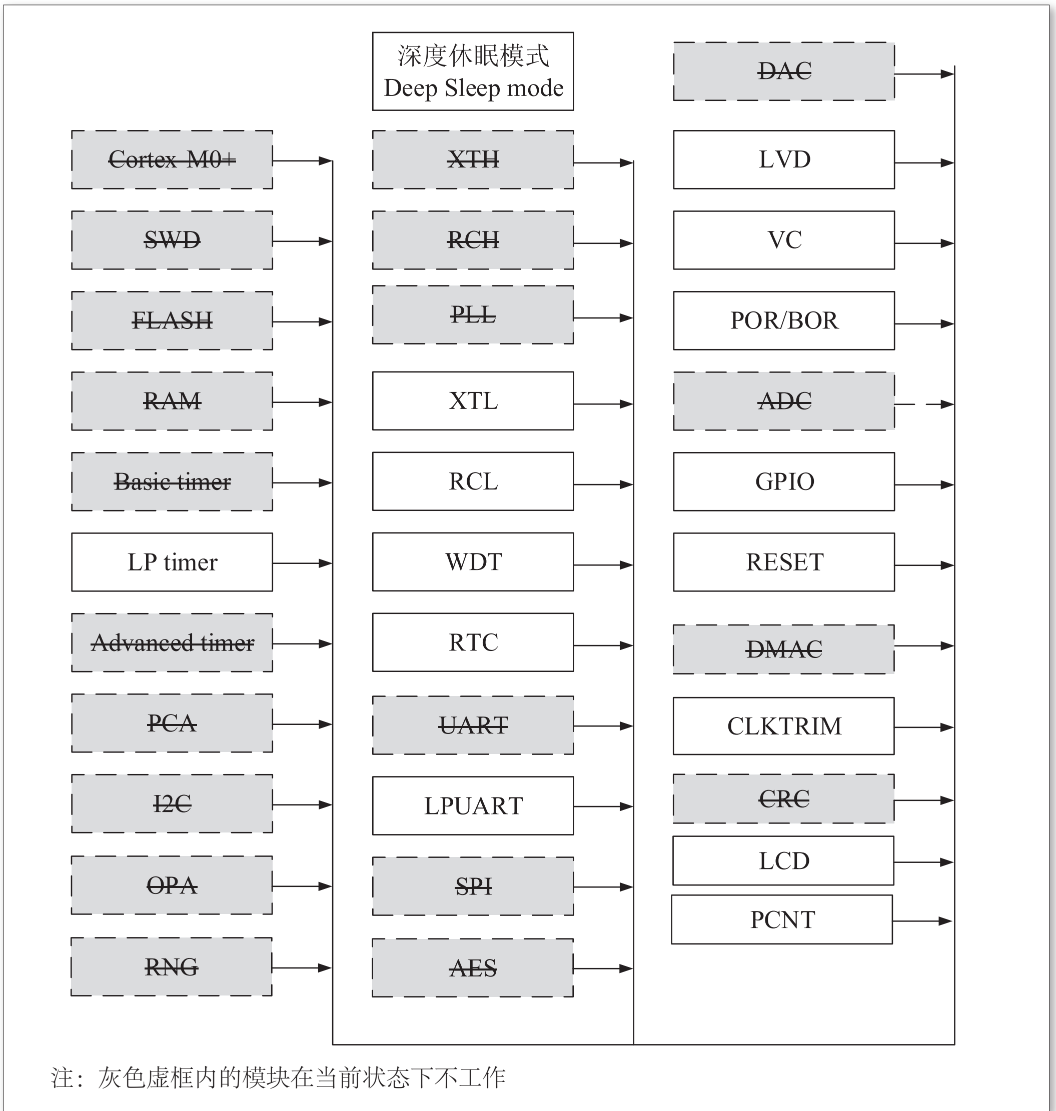

图 6. 深度休眠模式下可运行模块图

3.3.1　系统控制寄存器 (Cortex-M0+ 内核系统控制寄存器)

起始地址: 0xE000ED10

偏移地址: 0x00

复位值：0x0000 0000

Bit	Field	Type	Reset	Description
31:5	Reserved			保留。
4	SEVONPEND	rw	0x00	设置为 1 时，每次新的中断挂起都会产生一个事件，如果使用了 WFE 休眠，它可用于唤醒处理器。
3	Reserved			保留。
2	SLEEPDEEP	rw	0x00	设为 1 时，执行 WFI 进入深度休眠，本产品进入 Deep sleep 模式。 设置为 0 时，执行 WFI 进入休眠，本产品进入 sleep/Idle 模式。
1	SLEEP-ONEXIT	rw	0	设为 1 时，当退出异常处理并返回程序线程时，处理器自动进入休眠模式 (WFI)。 设置为 0 时，该特性就会被自动禁止。
0	Reserved			保留。

进入深度休眠后,唤醒后系统时钟有两种选择,默认使用进入深度休眠的时钟,配置寄存器 SYSCTRL0.wakeup-_byRCH 为 1 后不管进入深度休眠前是什么时钟，唤醒后都使用内部高速时钟 RCH。如果使用外部晶体振荡这样设置可以加速唤醒系统。

4 读卡器（Reader）

4.1 读卡器功能介绍

读卡器的工作频率是 13.56MHz，支持符合 ISO/IEC 14443 TypeA/MIFARE 协议的读写器模式，同时支持符合 ISO/IEC 14443 TypeB 的读写器模式。具有低功耗的外部卡片侦测功能，方便电池供电、需要低功耗工作、并且需要实时处理任意时刻会进入射频场的外部卡片的读写器设备。

4.2 功能特点

- 支持 ISO/IEC 14443 TypeA/MIFARE 协议；
- 支持 ISO/IEC 14443 TypeB 协议；
- 支持操作 Mifare UltraLight、NTAG、Mifare S50/S70、Desfire 等卡片；
- ISO/IEC 14443 TypeA 支持 106kbps，212kbps，424kbps，848kbps 等多种速率；
- 支持低功耗外部卡片侦测功能；
- 支持配合通道芯片拓展多路天线的应用，最多可拓展八路天线；
- 读卡距离可达 7cm（取决于天线设计）。

4.3 Reader 寄存器

表 7. Reader 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x12	TxModReg	定义发射速率和帧格式	0x00	小节 4.3.1
0x13	RxModReg	定义接收速率和帧格式	0x00	小节 4.3.2
0x14	TxControlReg	天线驱动引脚 CZ_TX1 和 CZ_TX2 控制	0x80	小节 4.3.3
0x18	RxThresholdReg	收译码器阈值选择	0x84	小节 4.3.4
0x24	ModWidthReg	调制宽度控制	0x26	小节 4.3.5
0x26	RFCfgReg	配置接收器增益和 RF 电平	0x48	小节 4.3.6
0x27	GsNOnReg	当驱动使能时，选择天线驱动 CZ_TX1 和 CZ_TX2 的 N 驱动的电导	0x88	小节 4.3.7
0x28	CWGsPReg	控制无调制时段 P 驱动的电导	0x20	小节 4.3.8
0x29	ModGsPReg	控制调制时段 P 驱动的电导	0x20	小节 4.3.9

4.3.1 TxModReg 寄存器

地址：0x12

复位值：0x00，定义发射速率和帧格式

Bit	Field	Type	Reset	Description
7	-	-	-	保留。

Bit	Field	Type	Reset	Description
6:4	TxSpeed[6:4]	rw	0x00	定义数据发射速率。 000：106kbit 001：212kbit 010：424kbit 011：848kbit 其他：预留。
3:2	-	-	-	保留。
1:0	TxFraming [1:0]	rw	0x00	定义发送数据格式。 00：ISO/IEC 14443A 11：ISO/IEC 14443B 其他：预留

4.3.2 RxModReg 寄存器

地址：0x13

复位值：0x00，定义接收速率和帧格式

Bit	Field	Type	Reset	Description
7	-	-	-	保留。
6:4	RxSpeed[6:4]	rw	0x00	定义数据接收速率。 000：106kbit 001：212kbit 010：424kbit 011：848kbit 其他：预留。
3:2	-	-	-	保留。
1:0	RxFraming [1:0]	rw	0x00	定义接收数据格式。 00：ISO/IEC 14443A 11：ISO/IEC 14443B 其他：预留

4.3.3 TxControlReg 寄存器

地址：0x14

复位值：0x80，天线驱动引脚 CZ_TX1 和 CZ_TX2 控制

Bit	Field	Type	Reset	Description
7:2	-	-	-	保留。
1	Tx2RFEn	rw	0x00	置 1，引脚 CZ_TX2 输出经由发送数据调制的能量载波信号。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
0	Tx1RFEn	rw	0x00	置 1，引脚 CZ_TX1 输出经由发送数据调制的能量载波信号。

4.3.4 RxThresholdReg 寄存器

地址：0x18

复位值：0x84，用于设置接收译码器阈值选择

Bit	Field	Type	Reset	Description
7:4	MinLevel[7:4]	rw	0x08	定义了译码器能接收的最小信号强度，如果信号强度低于这个水平，则该信号不被处理。
3	-	-	-	保留。
2:0	CollLevel[2:0]	rw	0x04	定义了输入到译码器的曼彻斯特编码的弱半位相对于强半位产生冲突位的最小信号强度。

4.3.5 ModWidthReg 寄存器

地址：0x24

复位值：0x26，用于调制宽度控制

Bit	Field	Type	Reset	Description
7:0	ModWidth[7:0]	rw	0x26	定义了主动通信模式下发起端应用的密勒编码调制宽度，宽度与载波频率关系为：（ModWidth+1）/fc。 0x26：106kbps 0x13：212kbps 0x09：424kbps 0x04：848kbps

4.3.6 RFCfgReg 寄存器

地址：0x26

复位值：0x48，用于配置接收器增益

Bit	Field	Type	Reset	Description
7	-	-	-	保留。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
6:4	RxGain[6:4]	rw	0x04	该寄存器定义了接收器信号电压增益指数。 000：18dB 001：23dB 010：18dB 011：23dB 100：33dB 101：38dB 110：43dB 111：48dB
3:0	-	-	-	保留。

4.3.7 GsNOnReg 寄存器

地址：0x27

复位值：0x88，用于当驱动使能时，选择天线驱动 CZ_TX1 和 CZ_TX2 的 N 驱动的电导

Bit	Field	Type	Reset	Description
7:4	CWGsNOn [7:4]	rw	0x08	该寄存器的值定义了无调制时段的输出 N 驱动的电导，可通过该设置控制输出功率，相应的电流功耗以及操作距离。
3:0	ModGsNOn [3:0]	rw	0x08	改该寄存器的值定义了调制时段的输出 N 驱动的电导，可用来控制调制深度。

4.3.8 CWGsPReg 寄存器

地址：0x28

复位值：0x20，用于控制无调制时段 P 驱动的电导

Bit	Field	Type	Reset	Description
7:6	-	-	-	保留。
5:0	CWGsP[5:0]	rw	0x20	改该寄存器的值定义了无调制时段的输出 P 驱动的电导，可通过该设置控制输出功率，相应的电流功耗以及操作距离。

4.3.9 ModGsPReg 寄存器

地址：0x29

复位值：0x20，用于控制调制时段 P 驱动的电导

Bit	Field	Type	Reset	Description
7:6	-	-	-	保留。

Bit	Field	Type	Reset	Description
5:0	ModGsP[5:0]	rw	0x20	改该寄存器的值定义了调制时段的输出 P 驱动的电导，可用来控制调试深度。

4.4 读卡外设操作方式

参考《ZSN700 读卡芯片 API 手册》。

5 系统控制器（SYSCTRL）

5.1 系统时钟介绍

时钟控制模块主要控制系统时钟以及外设时钟，可以配置不同的时钟源作为系统时钟、可以配置不同的系统时钟分频、可以启动或禁用外设时钟。为了确保振荡器精度，内部时钟均具有校准功能。

本产品支持以下五个不同的时钟源作为系统时钟：

1. 内部高速 RC 时钟 RCH（4M）（默认主频）；
2. 内部低速 RC 时钟 RCL（38.4K 与 32.8K 可配置）；
3. 外部高速晶振时钟 XTH；
4. 外部低速晶振时钟 XTL；
5. 锁相环时钟 PLL。

注 1：切换系统时钟的时钟源时，请严格按照操作步骤进行切换，详见 5.2 章节。

注 2：XTL 可以不接晶振，直接从 PC14 管脚输入 32.768KHz 的时钟信号。XTH 可以不接晶振，直接从 PF00 管脚输入 4~32MHz 的时钟信号。

本产品还包含以下两个辅助时钟：

1. 内部低速 10K 时钟；仅供看门狗和 CLKTRIM 模块使用。
2. 内部 150K 时钟：仅供 LVD 和 VC 模块使用。

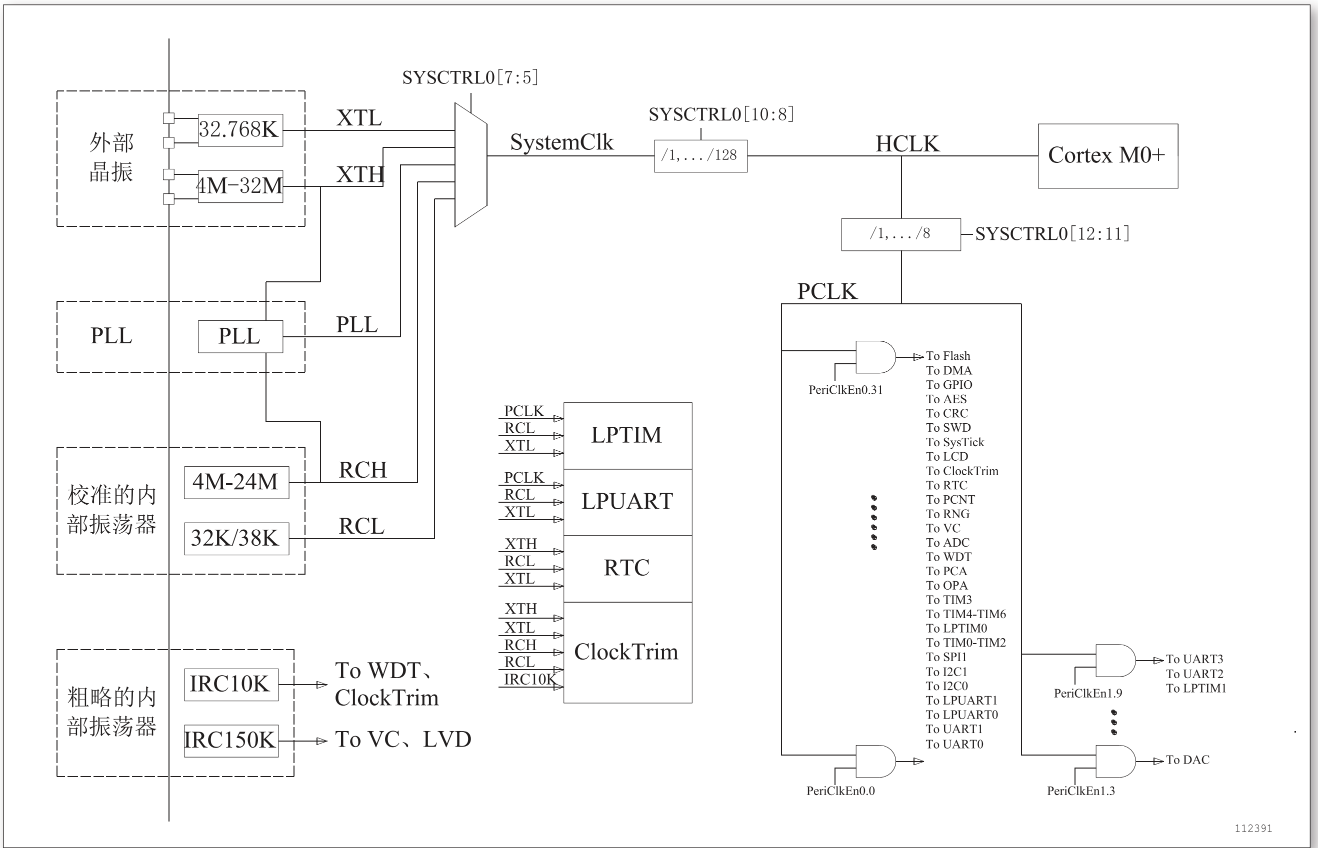

图 7. 时钟控制模块框图

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

5.1.1 内部高速 RC 时钟 RCH

芯片上电或复位后的默认时钟源为内部高速时钟，频率为 4MHz。当系统进入 DeepSleep，此高速时钟会自动关闭。内部高速时钟可通过寄存器 RCH_CR[10:0] 来选择其输出频率，可供选择的频率为 4MHz、8MHz、16MHz、22.12MHz、24MHz。

内部高速时钟从启动到稳定仅需 4us。为了在深度休眠模式下能快速响应中断，建议进入深度休眠模式前将系统时钟切换为 RCH。

5.1.2 内部低速 RC 时钟 RCL

内部低速时钟可以通过寄存器 RCL_CR[9:0] 来选择其输出频率，可供选择的频率为 38.4KHz、32.768KHz。当系统进入 DeepSleep，此低速时钟不会自动关闭，超低功耗外设模块可以选择 RCL 作为其时钟。

5.1.3 外部低速晶振时钟 XT

外部低速晶振时钟需外接一个 32.768KHz 的低功耗晶振，具有超高精度以及超低功耗。当系统进入 Deep Sleep，此低速时钟不会自动关闭。超低功耗模式下工作的外设模块可以选择 XTL 作为其时钟。

XTL 也可以不接晶振，直接从 PC14 管脚输入 32.768KHz 的时钟信号。从 PC14 输入时钟信号的方法为：配置 PC14 引脚为 GPIO 输入；设置 SYSCTRL1. EXTL_EN 为 1。

5.1.4 外部高速晶振时钟 XT

外部高速晶振时钟需外接一个 4MHz~32MHz 的高速晶振。当系统进入 Deep Sleep，此高速时钟会自动关闭。

XTH 也可以不接晶振，直接从 PF00 管脚输入 4MHz~32MHz 的时钟信号。从 PF00 输入时钟信号的方法为：配置 PF00 引脚为 GPIO 输入；设置 SYSCTRL1. EXTH_EN 为 1。

5.1.5 锁相环时钟 PLL

内建 PLL 支持 8M~48M 的时钟输出。PLL 的参考时钟源为：RCH、XTH 晶振时钟、PF00 管脚输入时钟。

5.1.6 时钟启动过程

上述五种时钟源都需要启动稳定时间，下图以外部 XTL 为例说明时钟的启动稳定过程。

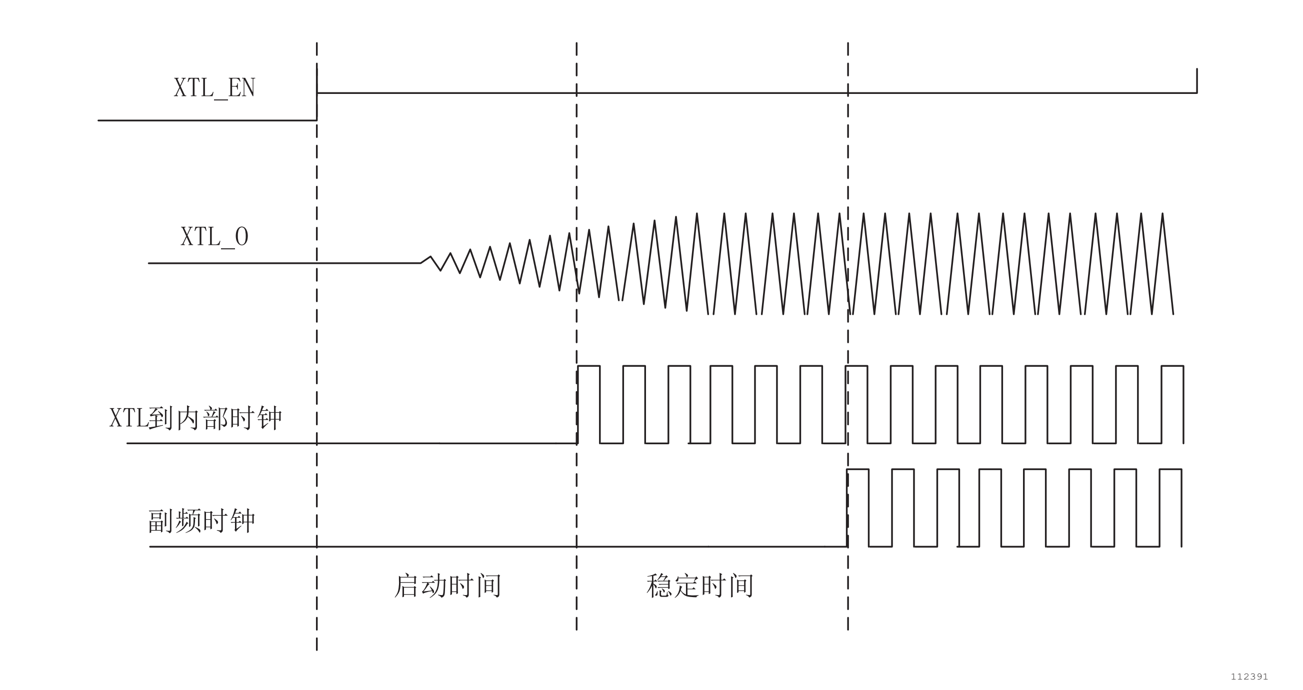

图 8. 晶振时钟启动示意图

5.2 系统时钟切换

时钟源的切换是由寄存器 SYSCTRL0[7:0] 来控制。系统时钟的时钟源可通过 SYSCTRL0[7:5] 在 RCH、RCL、XTH、XTL、PLL 之间进行切换。切换时 RCH、RCL、XTH、XTL 这四个时钟源中的任意两者可以相互切换；PLL 只可与 RCH、XTH 这两个时钟进行切换。时钟切换操作必须按照下文所描述的 7 种时钟切换流程进行，否则可能出现异常。

时钟切换时需要同步配置 FLASH_CR. WAIT。时钟频率不大于 24MHz 则应设置 FLASH_CR.WAIT 为 0；时钟频率大于 24MHz 则应设置 FLASH_CR. WAIT 为 1；时钟频率大于 48MHz 则应设置 FLASH_CR. WAIT 为 2。

注：设置 FLASH_CR. WAIT 的值，需要先向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5 然后再对 FLASH_CR. WAIT 进行赋值，详见 FLASH 控制器章节。

5.2.1 标准的时钟切换流程

操作流程如下：

Step1：如新时钟源需要外部引脚，则将该引脚设置为适当的模式。

注：接外部晶振时需要模拟引脚；接外部时钟输入时需要 GPIO 输入并使能外部时钟输入。

Step2：配置新时钟源的的振荡参数。

Step3：使能新时钟源的振荡器。

Step4：根据当前时钟源和新时钟源两者中较高的频率，按 Flash 控制器章节流程配置 FLASH_CR. WAIT。

Step5：等待新时钟源输出稳定的频率。

Step6：配置 SYSCTRL0. CLKSW，选择系统时钟的来源为新时钟源。

Step7：根据新时钟源的频率，按 Flash 控制器章节流程配置 FLASH_CR. WAIT。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Step8：关闭不再使用的时钟源。

5.2.2 RCH 不同振荡频率间切换流程

RCH 不同振荡频率间切换具有两种方案。

方案 1：

Step1：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step2：设置 SYSCTRL0.HCLK_PRS 为 0x7。

Step3：向上或向下逐级调整 RCH 的输出频率，4M -> 8M -> 16M -> 24M/22.12M 或 24M/22.12M -> 16M -> 8M -> 4M。

Step4：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step5：设置 SYSCTRL0.HCLK_PRS 为 0x0。

从 4M 切换到 24M 的示例代码如下所示：

```
M0P_SystemCtrl->SYSCTRL2 = 0X5A5A;
M0P_SystemCtrl->SYSCTRL2 = 0XA5A5;
M0P_SystemCtrl->SYSCTRL0_f.HCLK_PRS = 7;
M0P_SystemCtrl->RCH_CR = *((uint16 *)( 0X00100C08 ) ); //4M
M0P_SystemCtrl->RCH_CR = *((uint16 *)( 0X00100C06 ) ); //8M
M0P_SystemCtrl->RCH_CR = *((uint16 *)( 0X00100C04 ) ); //16M
M0P_SystemCtrl->RCH_CR = *((uint16 *)( 0X00100C00 ) ); //24M
M0P_SystemCtrl->SYSCTRL2 = 0X5A5A;
M0P_SystemCtrl->SYSCTRL2 = 0XA5A5;
M0P_SystemCtrl->SYSCTRL0_f.HCLK_PRS = 0;
```

方案 2：

Step1：将系统时钟切换为 RCL，参见 5.2.5 从其它时钟切换到 RCL 示例。

Step2：将系统时钟切换为 RCH，参见 5.2.6 从其它时钟切换到 RCH 示例。

5.2.3 从其它时钟切换到 XTL 示例

操作流程如下：

Step1：设置 PCADS. 14 及 PCADS. 15 为 1，配置 PC14/PC15 引脚为模拟端口。

Step2：根据晶振特性，配置 XTL_CR[5:0]。

Step3：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step4：设置 SYSCTRL0. XTL_EN 为 1，使能晶振振荡电路。

Step5：查询等待 XTL_CR. Stable 标志变为 1，晶振输出稳定时钟。

Step6：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step7：设置 SYSCTRL0. CLKSW 为 3，将系统时钟切换为 XTL。

Step8：按 Flash 控制器章节流程设置 FLASH_CR. WAIT 为 0。

Step9：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step10：设置 SYSCTRL0.xxx_EN 为 0，关闭原时钟。

5.2.4 从其它时钟切换到 XTH 示例

Step1：设置 PFADS. 0 及 PFADS. 1 为 1，配置 PF00/PF01 引脚为模拟端口。

Step2：根据晶振特性，配置 XTH_CR[3:0]。

Step3：设置 XTH_CR. Startup 为 3，选择最长的晶振稳定时间。

Step4：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step5：设置 SYSCTRL0. XTH_EN 为 1，使能晶振振荡电路。

Step6：根据当前时钟和 XTH 两者中较高的频率，按 Flash 控制器章节流程配置 FLASH_CR.WAIT。

Step7：查询等待 XTH_CR. Stable 标志变为 1，晶振输出稳定时钟。

Step8：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step9：设置 SYSCTRL0. CLKSW 为 1，将系统时钟切换为 XTH。

Step10：根据 XTH 的频率，按 Flash 控制器章节流程配置 FLASH_CR. WAIT。

Step11：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step12：设置 SYSCTRL0.xxx_EN 为 0，关闭原时钟。

5.2.5 从其它时钟切换到 RCL 示例

操作流程如下：

Step1：配置 RCL_CR.TRIM 及 RCL_CR. Startup。

Step2：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step3：设置 SYSCTRL0. RCL_EN 为 1，使能 RCL 振荡电路。

Step4：查询等待 RCL_CR. Stable 标志变为 1，RCL 输出稳定时钟。

Step5：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step6：设置 SYSCTRL0. CLKSW 为 2，将系统时钟切换为 RCL。

Step7：按 Flash 控制器章节流程设置 FLASH_CR. WAIT 为 0。

Step8：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step9：设置 SYSCTRL0.xxx_EN 为 0，关闭原时钟。

5.2.6 从其它时钟切换到 RCH 示例

操作流程如下：

Step1：配置 RCH_CR.TRIM。

Step2：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step3：设置 SYSCTRL0. RCH_EN 为 1，使能 RCH 振荡电路。

Step4：查询等待 RCH_CR. Stable 标志变为 1，RCH 输出稳定时钟。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Step5：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step6：设置 SYSCTRL0. CLKSW 为 0，将系统时钟切换为 RCH。

Step7：按 Flash 控制器章节流程设置 FLASH_CR. WAIT 为 0。

Step8：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step9：设置 SYSCTRL0.xxx_EN 为 0，关闭原时钟。

5.2.7 PLL 与 RCH 相互切换示例，参考时钟为 RCH

从 RCH 切换到 PLL 操作流程如下：

Step1：设置 PLL_CR.REFSEL 为 3，选择 PLL 时钟源为 RCH。

Step2：根据 RCH 频率，配置 PLL_CR.FRSEL。

Step2：根据 PLL 输出频率，配置 PLL_CR.DIVN 及 PLL_CR.FOSC。

Step3：设置 PLL_CR. Startup 为 7，选择最长的 PLL 稳定时间。

Step4：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step5：设置 SYSCTRL0.PLL_EN 为 1，使能 PLL 振荡电路。

Step6：根据 PLL 输出频率，按 Flash 控制器章节流程配置 FLASH_CR. WAIT。

Step7：查询等待 PLL_CR. Stable 标志变为 1，PLL 输出稳定时钟。

Step8：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step9：设置 SYSCTRL0. CLKSW 为 4，将系统时钟切换为 PLL。

警告：切换过程中不可以更改 RCH_CR.TRIM。

从 PLL 切换到 RCH 操作流程如下：

Step1：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step2：设置 SYSCTRL0. CLKSW 为 0，将系统时钟切换为 RCH。

Step3：按 Flash 控制器章节流程设置 FLASH_CR. WAIT 为 0。

Step4：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step5：设置 SYSCTRL0.PLL_EN 为 0，关闭 PLL。

警告：切换过程中不可以更改 RCH_CR.TRIM。

5.2.8 PLL 与 XTH 相互切换示例，参考时钟为 XTH

从 RCH 切换到 PLL 操作流程如下：

Step1：设置 PLL_CR.REFSEL 为 0，选择 PLL 时钟源为 XTH。

Step2：根据 XTH 频率，配置 PLL_CR.FRSEL。

Step2：根据 PLL 输出频率，配置 PLL_CR.DIVN 及 PLL_CR.FOSC。

Step3：设置 PLL_CR. Startup 为 7，选择最长的 PLL 稳定时间。

Step4：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Step5：设置 SYSCTRL0.PLL_EN 为 1，使能 PLL 振荡电路。

Step6：根据 PLL 输出频率，按 Flash 控制器章节流程配置 FLASH_CR. WAIT。

Step7：查询等待 PLL_CR. Stable 标志变为 1，PLL 输出稳定时钟。

Step8：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step9：设置 SYSCTRL0. CLKSW 为 4，将系统时钟切换为 PLL。

从 PLL 切换到 XTH 操作流程如下：

Step1：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step2：设置 SYSCTRL0. CLKSW 为 1，将系统时钟切换为 XTH。

Step7：根据 XTH 的频率，按 Flash 控制器章节流程配置 FLASH_CR. WAIT。

Step8：向 SYSCTRL2 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

Step9：设置 SYSCTRL0.PLL_EN 为 0，关闭 PLL。

图 9为时钟切换时序图：

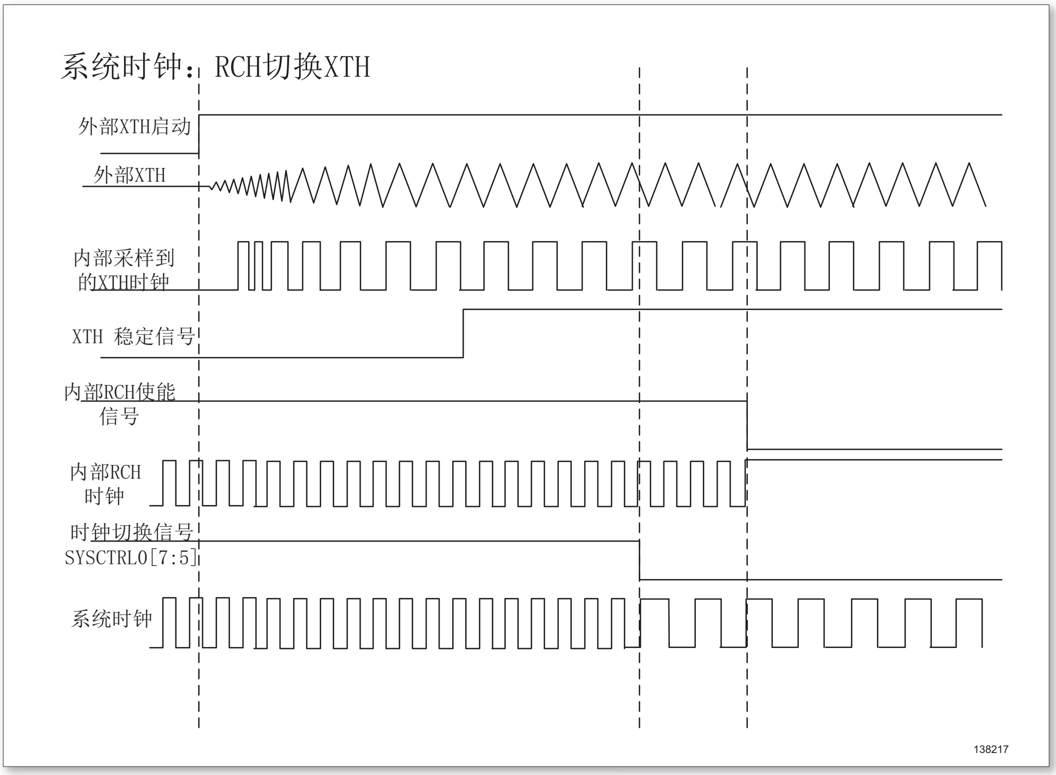

图 9. 时钟切换示意图

5.3 时钟校准模块

本产品内嵌时钟校准电路，如图 10所示，系统时钟的 5 个源均可以相互校准，当选择好参考时钟以及被校准时钟后，设置寄存器 REFCNT 值，置位 cali.start 启动时钟校准电路，此时两个 32 位计数器（递增、递减）同时工作，当递减计数器等于 0 时，cali.finish 被置位，表明校准结束，此时软件可以读取 CALCNT 值，这样就很容易得到参考时钟与被校准时钟之间的频率关系。

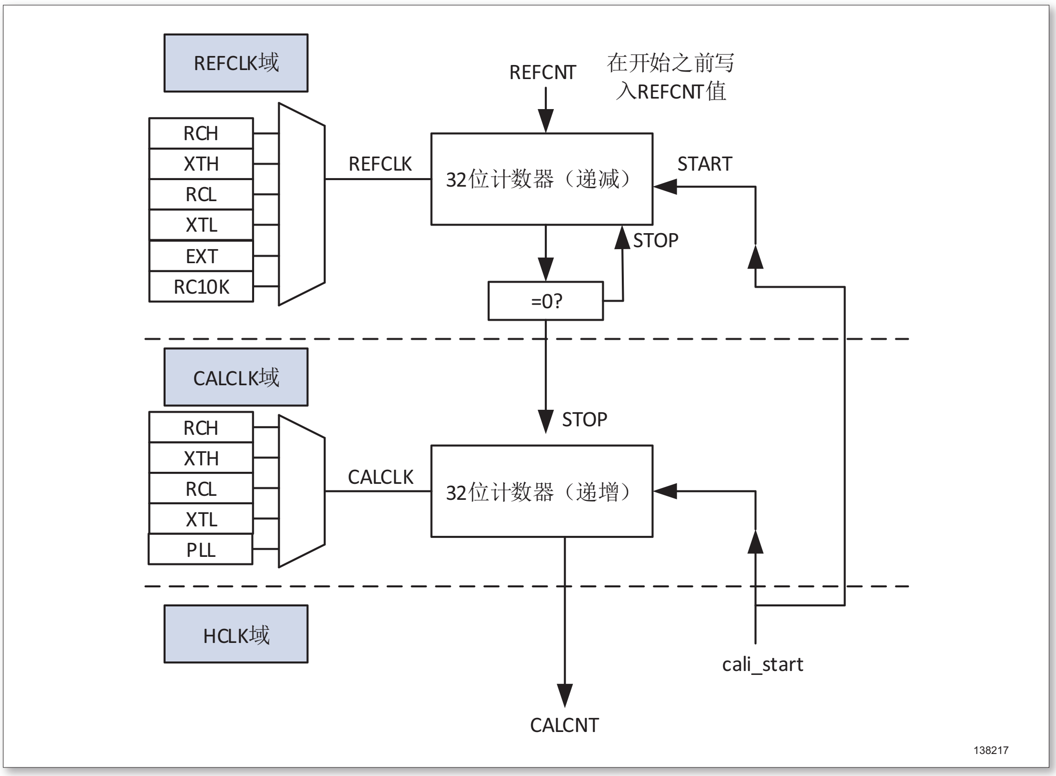

图 10. 时钟校准原理图

5.4 中断唤醒控制

当处理器执行 WFI 指令进入休眠状态时，会停止执行指令。在休眠状态下发生了中断请求（更高优先级）且需要处理时，处理器就会被唤醒。

休眠状态下的处理器收到中断请求时的行为如表 8所示：

表 8. 休眠状态下的收到中断请求时的行为

PRIMASK 状态	WFI 行为	唤醒	ISR 执行
0	IRQ 优先级 > 当前等级	Y	Y
0	IRQ 优先级 ≤ 当前等级	N	N
1	IRQ 优先级 < 当前等级	Y	N
1	IRQ 优先级 ≤ 当前等级	N	N

5.4.1 从深度休眠模式唤醒后执行中断服务程序的方法

1. 使能需要唤醒处理器的模块所对应的 NVIC。
2. 使能需要唤醒处理器的模块所对应的中断。
3. 设置 SCB->SCR.SLEEPDEEP 为 1。
4. 执行 WFI 指令以进入深度休眠模式。
5. 系统进入深度休眠模式等待中断唤醒，唤醒后执行中断服务程序。

例程：

```
SCB_SCR |= 0x00000004u;
```

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

```
while(1)
{
    __asm("WFI");
}
```

5.4.2 从深度休眠模式唤醒后不执行中断服务程序的方法

1. 使能需要唤醒处理器的模块所对应的 NVIC。
2. 使能需要唤醒处理器的模块所对应的中断。
3. 设置 PRIMASK 为 1。
4. 设置 SCB->SCR.SLEEPDEEP 为 1。
5. 执行 WFI 指令以进入深度休眠模式。
6. 系统进入深度休眠模式等待中断唤醒，唤醒后执行下一条指令。
7. 清除中断标志，清除中断挂起状态。
8. 执行用户定义的操作。

例程：

```
__asm("CPSID I"); //Set PRIMASK
SCB_SCR |= 0x00000004u;
while(1)
{
    __asm("WFI");
    BTIMERLP_REG->TFCR_f.TFC=0;              //Clear Int Flag
    NVIC_ClearPendingIRQ(TIMERLP_IRQn);      //Clear Pending Flag
    …       //执行用户定义的操作
}
```

5.4.3 使用退出休眠特性

退出休眠（sleep-on-exit）非常适合中断驱动的应用程序。当该特性使能时，只要完成异常处理并且返回到了线程模式，处理器就会进入休眠模式。利用退出休眠特性，处理器可以尽可能多的处于休眠模式。

Cortex-M0 利用退出休眠特性进入休眠，这种情况同执行完异常退出后立即执行 WFI 的效果差不多。不过，为了下次进入异常时，不用再进行压栈操作，处理器不会执行出栈的过程。

1. 使能需要唤醒处理器的模块所对应的 NVIC。
2. 使能需要唤醒处理器的模块所对应的中断。
3. 设置 SCB->SCR.SLEEPDEEP 为 1。
4. 设置 SCB—>SCR.SLEEPONEXIT 为 1。
5. 执行 WFI 指令以进入深度休眠模式。
6. 系统进入深度休眠模式等待中断唤醒，唤醒后执行中断服务子程序。
7. 退出中断服务时自动进入休眠模式。

例程：

```
SCB_SCR |= 0x00000004u;
SCB_SCR |= 0x00000002u;
while(1)
{
    __asm("WFI");
}
```

5.5 SYSCTRL 寄存器描述

表 9. SYSCTRL 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x00	SYSCTRL0	系统控制寄存器 0	0x0000 0001	小节 5.5.1
0x04	SYSCTRL1	系统控制寄存器 1	0x0000 0008	小节 5.5.2
0x08	SYSCTRL2	系统控制寄存器 2	0x0000 0000	小节 5.5.3
0x0C	RCH_CR	RCH 控制寄存器	0x0000 0126	小节 5.5.4
0x10	XTH_CR	XTH 控制寄存器	0x0000 0022	小节 5.5.5
0x14	RCL_CR	RCL 控制寄存器	0x0000 33FH	小节 5.5.6
0x18	XTL_CR	XTL 控制寄存器	0x0000 0021	小节 5.5.7
0x20	PERI_CLKEN0	外围模块时钟控制寄存器 0	0x0001 0B0F	小节 5.5.8
0x24	PERI_CLKEN1	外围模块时钟控制寄存器 1	0x8080 0000	小节 5.5.9
0x3C	PLL_CR	PLL 控制寄存器	0x0000 0000	小节 5.5.10

5.5.1 系统控制寄存器 0(SYSCTRL0)

起始地址：0x40002000

地址偏移：0x00

复位值：0x0000 0001

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Wakeup_byRCH	Reserved		PCLK_PRS		HCLK_PRS			CLKSW			PLL_EN	XTL_EN	RCL_EN	XTH_EN	RCH_EN
RW			RW		RW			RW			RW	RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
15	wakeup _byRCH	rw	0x00	1：从 Deep Sleep 唤醒后，system clock 来源为 RCH，原时钟继续使能。 0：从 Deep Sleep 唤醒后，不改变 system clock 来源。
14:13	Reserved			保留。
12:11	PCLK_PRS	rw	0x00	PCLK 时钟来源选择。 00：HCLK　01：HCLK/2 10：HCLK/4　11：HCLK/8
10:8	HCLK_PRS	rw	0x00	HCLK 时钟来源选择。 000：SystemClk　001：SystemClk/2 010：SystemClk/4　011：SystemClk/8 100：SystemClk/16　101：SystemClk/32 110：SystemClk/64　111：SystemClk/128
7:5	CLKSW	rw	0x00	SystemClk 时钟来源选择。 000：内部高速时钟 RCH 001：外部高速晶振 XTH 010：内部低速时钟 RCL 011：外部低速晶振 XTL 100：内部 PLL
4	PLL_EN	rw	0x00	PLL 使能控制。 0：关闭 1：使能
3	XTL_EN	rw	0x00	外部低速晶振 XTL 使能控制。 0：关闭 1：使能 注：需要将 PC14、PC15 设置成模拟端口。
2	RCL_EN	rw	0x00	内部低速时钟 RCL 使能控制。 0：关闭 1：使能
1	XTH_EN	rw	0x00	外部高速晶振 XTH 使能控制。 0：关闭 1：使能 注：当系统进入 DeepSleep，此高速时钟会自动关闭。
0	RCH_EN	rw	0x01	内部高速时钟 RCH 使能信号。 0：关闭 1：使能 注：当系统进入 DeepSleep，此高速时钟会自动关闭。

注意：

-每次改写 SYSCTRL0，SYSCTRL1 的值，均需要先对 SYSCTRL2 依次写入 0x5A5、0xA5A5。这样的步骤可有效防止对 SYSCTRL0，SYSCTRL1 寄存器的误操作。

5.5.2 系统控制寄存器 1(SYSCTRL1)

起始地址：0x40002000

地址偏移：0x04

复位值：0x0000 0008

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
</table>

<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="4" rowspan="2">Reserved</td><td colspan="3">RTC_FREQ_ADJUST</td><td>SWD_USE_IO</td><td rowspan="2">Res.</td><td>LOCKUP_EN</td><td>RTC_LPW</td><td rowspan="2">Res.</td><td>XTL_ALWAYS_ON</td><td>EXTL_EN</td><td>EXTH_EN</td><td rowspan="2">Res.</td></tr>
<tr><td colspan="3">RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:12	Reserved			保留。
11:9	RTC_FREQ_ADJUST	rw	0x00	RTC 高速时钟补偿时钟频率选择。 000 4M; 001 6M; 010 8M; 011 12M 100 16M; 101 20M; 110 24M; 111 32M;
8	SWD_USE_IO	rw	0x00	SWD 端口功能配置。 0：SWD 端口 1：GPIO 端口
7	Reserved			保留。
6	LOCKUP_EN	rw	0x00	Cortex-M0+ LockUp 功能配置。 0：关闭 1：使能 注：使能后，CPU 读到无效指令时会复位 MCU，可增强系统可靠性。
5	RTC_LPW	rw	0x00	RTC 模块低功耗控制。 1：低功耗模式使能 0：低功耗模式禁能 注：使能后，RTC 模块进低功耗状态，其寄存器不可以读写。
4	Reserved			保留。
3	XTL_ALWAYS_ON	rw	0x01	XTL 高级使能控制。 1：SYSCTRL0.XTL_EN 只可置位。 0：SYSCTRL0.XTL_EN 可置位可清零。
2	EXTL_EN	rw	0x00	外部 XTL 时钟输入控制。 1：XTL 输出时钟从 PC14 输入。 0：XTL 输出时钟由晶振产生。 注：使用 PC14 输入时钟时，需设置 SYSCTRL0.XTL_EN 为 1。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
1	EXTH_EN	rw	0x00	外部 XTH 输入控制。 1：XTH 输出时钟从 PF00 输入。 0：XTH 输出时钟由晶振产生。 注：使用 PF00 输入时钟时，需设置 SYSCTRL0.XTH_EN 为 1。
0	Reserved			保留。

5.5.3 系统控制寄存器 2(SYSCTRL2)

起始地址：0x40002000

地址偏移：0x08

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SYSCTRL2															
WO															

Bit	Field	Type	Reset	Description
31:12	Reserved			保留。
15:0	SYSCTRL2	w	0x00	寄存器 SYSCTL0，SYSCTRL1 保护系列控制寄存器，对 SYSCTRL2 先写 0x5A5A，再写 0xA5A5，启动对于寄存器 SYSCTL0，SYSCTRL1 的写操作，只要对寄存器 SYSCTL0，SYSCTRL1 写操作了，这个保护位自动回复保护状态，需要重新写入系列打开保护。

5.5.4 RCH 控制寄存器 (RCH_CR)

起始地址：0x40002000

地址偏移：0x0C

复位值：0x0000 0126

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				Stable	TRIM										
				RO	RW										

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。
11	stable	r	0x00	RCH 时钟稳定标志位。 1：代表 RCH 已经稳定，可以被内部电路使用。 0：代表 RCH 未稳定，不可以被内部电路使用。
10:0	TRIM	rw	0x126	时钟频率调整，Flash 中保存了 5 组频率的校准值。 将 Flash 内的校准值读出并写入 RCH_CR.TRIM 即可获得精准的频率。 24M 校准值地址：0x00100C00 - 0x00100C01 22.12M 校准值地址：0x00100C02 - 0x00100C03 16M 校准值地址：0x00100C04 - 0x00100C05 8M 校准值地址：0x00100C06 - 0x00100C07 4M 校准值地址：0x00100C08 - 0x00100C09

5.5.5 XTH 控制寄存器 (XTH_CR)

起始地址：0x40002000

地址偏移：0x10

复位值：0x0000 0022

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									Stable	Startup		xth_fsel		Driver	
									RO	RW		RW		RW	

Bit	Field	Type	Reset	Description
31:7	Reserved			保留。
6	stable	r	0x00	外部高速时钟 XTH 稳定标志位。 1：代表 XTH 已经稳定，可以被内部电路使用。 0：代表 XTH 未稳定，不可以被内部电路使用。
5:4	Startup	rw	0x10	外部高速时钟 XTH 稳定时间选择。 00: 256 个周期; 01: 1024 个周期; 10: 4096 个周期; 11: 16384 个周期; 注：强烈建议将 XTH 的稳定时间设置为 11。如果 XTH 稳定时间不足，在进行时钟切换时或从深度休眠唤醒时，系统不能稳定工作。

Bit	Field	Type	Reset	Description
3:2	xth_fsel	rw	0x00	外部晶振工作频率选择。 11: 20M~32M 10: 12M~20M 01: 6M~12M 00: 4M~6M
1:0	Driver	rw	0x10	外部晶振驱动能力选择。 11: 最强驱动能力 10: 默认驱动能力 (推荐值) 01: 弱驱动能力 00: 最弱驱动能力 注：需要根据晶振特性、负载电容以及电路板的寄生参数选择适当的驱动能力。驱动能力越大则功耗越大；驱动能力越弱则功耗越小。

5.5.6 RCL 控制寄存器 (RCL_CR)

起始地址：0x40002000

地址偏移：0x14

复位值：0x0000 033F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			Stable	Startup		TRIM									
			RO	RW		RW									

Bit	Field	Type	Reset	Description
31:13	Reserved			保留。
12	stable	r	0x00	内部低速时钟 RCL 稳定标志位。 1：代表 RCL 已经稳定，可以被内部电路使用。 0：代表 RCL 未稳定，不可以被内部电路使用。
11:10	Startup	rw	0x00	内部低速时钟 RCL 稳定时间选择。 11: 256 个周期; 10: 64 个周期; 01: 16 个周期; 00: 4 个周期;

Bit	Field	Type	Reset	Description
9:0	TRIM	rw	0x33F	内部低速时钟频率调整，Flash 中保存了 2 组频率的校准值。 将 Flash 内的校准值读出并写入 RCL_CR.TRIM 即可获得精准的频率。 38.4K 校准值地址：0x00100C20 - 0x00100C21 32.768K 校准值地址：0x00100C22 - 0x00100C23

5.5.7 XTL 控制寄存器 (XTL_CR)

起始地址：0x40002000

地址偏移：0x18

复位值：0x0000 0021

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									Stable	Startup		Amp_sel		Driver	
									RO	RW		RW		RW	

Bit	Field	Type	Reset	Description
31:7	Reserved			保留。
6	stable	r	0x00	外部低速晶振 XTL 稳定标志位。 1：代表 XTL 已经稳定，可以被内部电路使用。 0：代表 XTL 未稳定，不可以被内部电路使用。
5:4	Startup	rw	0x02	外部低速晶振 XTL 稳定时间选择。 00: 256 个周期; 01: 1024 个周期; 10: 4096 个周期; 11: 16384 个周期;
3:2	amp_sel	rw	0x00	XTL 晶振振荡幅度的调整。 11: 最大振幅 10: 较大振幅（推荐值） 01: 正常振幅 00: 最小振幅

Bit	Field	Type	Reset	Description
1:0	Driver	rw	0x01	XTL 晶振驱动能力选择。 11: 最强驱动能力 10: 较强驱动能力 01: 一般驱动能力（推荐值） 00: 最弱驱动能力 注：需要根据晶振特性、负载电容以及电路板的寄生参数选择适当的驱动能力。驱动能力越大则功耗越大；驱动能力越弱则功耗越小。

5.5.8 外围模块时钟控制寄存器 0(PERI_CLKEN0)

起始地址：0x40002000

地址偏移：0x20

复位值：0x8080 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FLASH	Res.	DMA	GPIO	AES	CRC	SWD	TICK	Res.	LCD	Trim	RTC	PCNT	RNG	VC	ADC
RW		RW	RW	RW	RW	RW	RW		RW	RW	RW	RW	RW	RW	RW

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDT	PCA	OPA	Res.	TIM3	ADVTIM	LPTIM0	BASETIM	SPI1	Res.	I2C1	I2C0	LPUART1	LPUART0	UART1	UART0
RW	RW	RW		RW	RW	RW	RW	RW		RW	RW	RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31	FLASH	rw	0x01	FLASH 控制器模块时钟使能。关闭后 FLASH 配置寄存器不可写，FLASH 中的程序仍然可以运行。 1：使能；0：关闭
30	Reserved			保留位
29	DMA	rw	0x00	DMAC 模块时钟使能。 1：使能；0：关闭
28	GPIO	rw	0x00	GPIO 模块时钟使能。 1：使能；0：关闭
27	AES	rw	0x00	AES 模块时钟使能。 1：使能；0：关闭
26	CRC	rw	0x00	CRC 模块时钟使能。 1：使能；0：关闭
25	SWD	rw	0x00	SWD 模块时钟使能。 1：使能；0：关闭

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
24	TICK	rw	0x00	SYSTICK 控制寄存器时钟使能。 1：使能；0：关闭
23	Reserved		0x01	保留位
22	LCD	rw	0x00	LCD 模块时钟使能。 1：使能；0：关闭
21	TRIM	rw	0x00	CLKTRIM 模块时钟使能。 1：使能；0：关闭
20	RTC	rw	0x00	RTC 模块时钟使能。 1：使能；0：关闭
19	PCNT	rw	0x00	PCNT 模块时钟使能。 1：使能；0：关闭
18	RNG	rw	0x00	RNG 模块时钟使能。 1：使能；0：关闭
17	VC	rw	0x00	VC、LVD，模块时钟使能。 1：使能；0：关闭
16	ADC	rw	0x00	ADC、BGR 模块时钟使能。 1：使能；0：关闭
15	WDT	rw	0x00	WDT 模块时钟使能。 1：使能；0：关闭
14	PCA	rw	0x00	PCA 模块时钟使能。 1：使能；0：关闭
13	OPA	rw	0x00	OPA 模块时钟使能。 1：使能；0：关闭
12	Reserved			保留位
11	TIM3	rw	0x00	TIM3 模块时钟使能。 1：使能；0：关闭
10	ADVTIM	rw	0x00	TIM456 模块时钟使能。 1：使能；0：关闭
9	LPTIM0	rw	0x00	LPTIM0 模块时钟使能。 1：使能；0：关闭
8	BASETIM	rw	0x00	TIM012 模块时钟使能。 1：使能；0：关闭
7	SPI1	rw	0x00	SPI1 模块时钟使能。 1：使能；0：关闭
6	Reserved		0x00	保留位
5	I2C1	rw	0x00	I2C1 模块时钟使能。 1：使能；0：关闭
4	I2C0	rw	0x00	I2C0 模块时钟使能。 1：使能；0：关闭

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
3	LPUART1	rw	0x00	LPUART1 模块时钟使能。 1：使能；0：关闭
2	LPUART0	rw	0x00	LPUART0 模块时钟使能。 1：使能；0：关闭
1	UART1	rw	0x00	UART1 模块时钟使能。 1：使能；0：关闭
0	UART0	rw	0x00	UART0 模块时钟使能。 1：使能；0：关闭

5.5.9 外围模块时钟控制寄存器 1(PERI_CLKEN0)

起始地址：0x40002000

地址偏移：0x24

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						UART3	UART2	Reserved			LPTIM1	DAC	Reserved		
						R/W	R/W				R/W	WO			

Bit	Field	Type	Reset	Description
31:10	Reserved			保留。
9	UART3	rw	0x00	UART3 模块时钟使能。 1：使能；0：关闭
8	UART2	rw	0x00	UART2 模块时钟使能。 1：使能；0：关闭
7:5	Reseved			保留。
4	LPTIM1	rw	0x00	LPTIM1 模块时钟使能。 1：使能；0：关闭
3	DAC	w	0x00	DAC 模块时钟使能。 1：使能；0：关闭 此控制为不可读，读出值始终为 0，使用读修改写其他控制位时 DAC 使能会被关闭。
2:0	Reseved			保留。

5.5.10 PLL 控制寄存器 (PLL_CR)

起始地址：0x40002000

地址偏移：0x3C

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

复位值：0x0001 0B0F

31	30	2	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved													Stable	Startup	
													RO	RW	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Stableopt	FRSEL		LFSEL		IBSEL		DVIN				FOSC			REFSEL	
RW	RW		RW		RW		RW				RW			RW	

Bit	Field	Type	Reset	Description
31:19	Reserved			保留。
18	Stable	r	0x00	PLL 稳定标志。 0: PLL 未稳定 1: PLL 已经稳定
17:15	Startup	rw	0x02	PLL 稳定时间选择。 000: 128 个 PLL 周期 001: 256 个 PLL 周期 010: 512 个 PLL 周期 011: 1024 个 PLL 周期 100: 2048 个 PLL 周期 101: 4096 个 PLL 周期 110: 8192 个 PLL 周期 111: 16384 个 PLL 周期
14:13	FRSEL	rw	0x00	PLL 输入频率选择，根据 PLL 输入时钟的频率进行如下配置。 00：4M~6M 01：6M~12M 10：12M~20M 11：20M~24M
12:11	LFSEL	rw	0x01	PLL 滤波器控制位，请保持默认值。
10:9	IBSEL	rw	0x01	PLL 偏置电流选择，请保持默认值。
8:5	DIVN	rw	0x08	PLL 输出时钟的倍频系数。 PLL 输出频率与输入频率的倍数 输出频率 = DIVN* 输入频率 DIVN 允许范围 0X2~0XC

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
4:2	FOSC	rw	0x03	PLL 输出频率范围选择，根据 PLL 输出时钟频率进行如下配置。 000：8M~12M 001：12M~18M 010：18M~24M 011：24M~36M 1xx：36M~48M
1:0	REFSEL	rw	0x03	输入时钟选择。 x0：XTH 晶振生成的时钟 01：XTH 从管脚 PF00 输入的时钟。（详见 5.1.4） 11：RCH 时钟

6 复位控制器 (RESET)

6.1 复位控制器介绍

本产品具有 7 个复位信号来源，每个复位信号都可以让 CPU 重新运行，绝大多数寄存器会被复位到复位值，程序会从复位向量处开始执行。

- 数字区域上电掉电复位 POR；
- 外部 Reset PAD，低电平为复位信号；
- WDT 复位；
- PCA 复位；
- LVD 低电压复位；
- Cortex-M0+ SYSRESETREQ 软件复位；
- Cortex-M0+ LOCKUP 硬件复位。

每个复位源由相应的复位标志进行指示。复位标志均由硬件置位，需要用户软件清零。芯片复位时，如果查询到 Reset_flag. POR15V 或 Reset_flag. POR5V 为 1 则为上电复位。上电复位时用户程序应当将寄存器 Reset_flag 清零，则下一次复位时可通过 Reset_flag 的相关比特判断复位来源。

图 11描述各区域的复位来源。

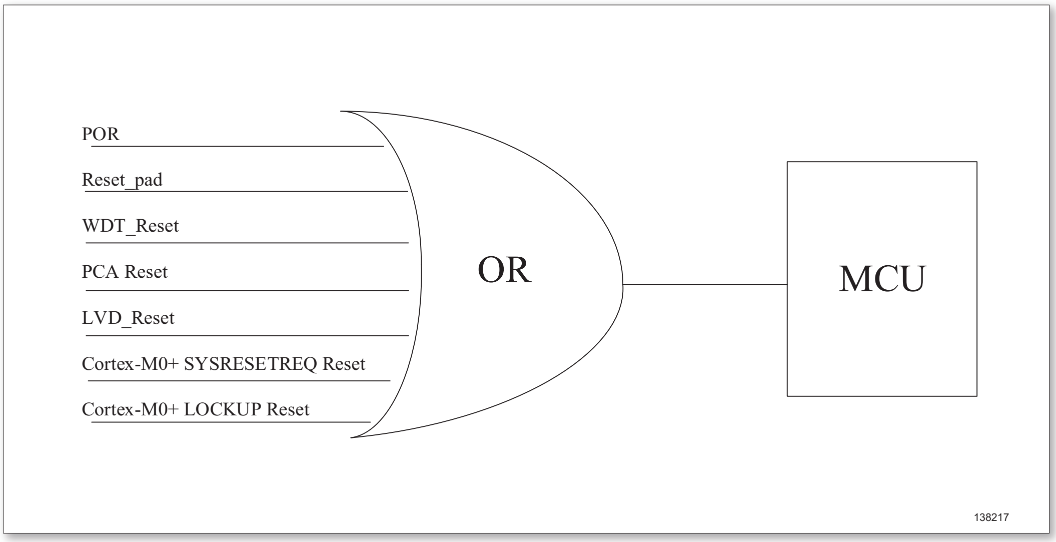

图 11. 复位来源示意图

6.1.1 上电下电复位 POR

本产品有两个供电区域：VCC 区域、Vcore 区域。所有的模拟模块及 IO 工作于 VCC 区域；其它模块工作于 Vcore 区域。

VCC 区域上电时，当 VCC 电压低于 POR 阈值电压时（典型值为 1.65V），会产生 POR5V 信号；VCC 区域下电时，当 VCC 电压低于 BOR 阈值电压时（典型值为 1.5V），会产生 POR5V 信号。

Vcore 区域上电时，当 Vcore 电压低于 POR 阈值电压时，会产生 POR15V 信号；Vcore 区域下电时，当 Vcore 电压低于 BOR 阈值电压时，会产生 POR15V 信号。

POR5V 信号和 POR15V 信号均会将芯片的寄存器复位到初始化状态。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

6.1.2　外部复位管脚复位

当外部复位引脚检测到低电平时会产生一个系统复位。该复位引脚已内置上拉电阻，并集成了一个毛刺过滤电路。毛刺过滤电路会过滤小于 20uS（典型值）的毛刺信号，因此，加到复位引脚上的低电平信号必须大于 20uS，才能确保芯片可靠复位。

6.1.3　WDT 复位

看门狗复位，请参看 WDT 一章说明。

6.1.4　PCA 复位

PCA 复位，请参看 PCA 一章说明。

6.1.5　LVD 低电压复位

LVD 复位，请参考 LVD 一章说明。

6.1.6　Cortex-M0+ SYSRESETREQ 复位

Cortex-M0+ 软件复位。

6.1.7　Cortex-M0+ LOCKUP 复位

当 Cortex-M0+ 遇到严重的异常时，它会将自己的 PC 指针停在当前地址处，并锁死自己，并在几个时钟周期延时之后复位整个 CORE 区域。

6.2　RESET 寄存器

表 10. RESET 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x1C	RESET_FLAG	复位标识寄存器	xxxxxx11b	小节 6.2.1
0x28	PERI_RESET0	外围模块复位控制寄存器 0	0x7F7F 6FFF	小节 6.2.2
0x2C	PERI_RESET1	外围模块复位控制寄存器 1	0x0000 0318	小节 6.2.3

6.2.1　复位标识寄存器 (RESET_FLAG)

起始地址：0x40002000

地址偏移：0x1C

复位值：00000000_00000000_00000000_xxxxxx11b

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								RSTB	sysreq	lockup	PCA	WDT	LVD	Por15	Por5v
								RW0	RW0	RW0	RW0	RW0	RW0	RW0	RW0

Bit	Field	Type	Reset	Description
31 :8	Reserved			保留。
7	RSTB	rc_w0	0xX	RESETB 端口复位标志，需要软件初始化及清除，上电状态不定。 1：发生端口复位 0：无端口复位发生
6	Sysreq	rc_w0	0xX	Cotrex-M0+ CPU 软件复位标志，需要软件初始化及清除，上电状态不定。 1：发生 Cotrex-M0+ CPU 软件复位 0：无 Cotrex-M0+ CPU 软件复位发生
5	Lockup	rc_w0	0xX	Cotrex-M0+ CPU Lockup 复位标志，需要软件初始化及清除，上电状态不定。 1：发生 Cotrex-M0+ CPU Lockup 复位 0：无 Cotrex-M0+ CPU Lockup 复位发生
4	PCA	rc_w0	0xX	PCA 复位标志，需要软件初始化及清除，上电状态不定。 1：发生 PCA 复位 0：无 PCA 复位发生
3	WDT	rc_w0	0xX	WDT 复位标志，需要软件初始化及清除，上电状态不定。 1：发生 WDT 复位 0：无 WDT 复位发生
2	LVD	rc_w0	0xX	LVD 复位标志，需要软件初始化及清除，上电状态不定。 1：发生 LVD 复位 0：无 LVD 复位发生
1	POR15V	rc_w0	0x1	Vcore 域复位标志。 1：Vcore 域发生复位 0：Vcore 域无复位发生
0	POR5V	rc_w0	0x1	VCC 电源域复位标志。 1：VCC 电源域发生复位 0：VCC 电源域无复位发生

6.2.2 外围模块复位控制寄存器 0(PERI_RESET0)

起始地址：0x40002000

地址偏移：0x28

复位值：0x7F7F 6FFF

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.		DMA	GPIO	AES	CRC	SWD	TICK	Res.	LCD	Trim	RTC	PCNT	RNG	VC	ADC
		RW	RW	RW	RW	RW	RW		RW	RW	RW	RW	RW	RW	RW

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res	PCA	OPA	Res.	TIM3	ADV TIM	LP TIM0	BASE TIM	SPI1	Res.	I2C1	I2C0	LPUART1	LPUART0	UART1	UART0
	RW	RW		RW	RW	RW	RW	RW		RW	RW	RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31 :30	Reserved			保留位。
29	DMA	rw	0x1	DMAC 模块复位使能。 1：正常工作；0：模块处于复位状态
28	GPIO	rw	0x1	GPIO 模块复位使能。 1：正常工作；0：模块处于复位状态
27	AES	rw	0x1	AES 模块复位使能。 1：正常工作；0：模块处于复位状态
26	CRC	rw	0x1	CRC 模块复位使能。 1：正常工作；0：模块处于复位状态
25	SWD	rw	0x1	SWD 模块复位使能。 1：正常工作；0：模块处于复位状态
24	TICK	rw	0x1	SYSTICK 模块复位使能。 1：正常工作；0：模块处于复位状态
23	Reserved			保留位。
22	LCD	rw	0x1	LCD 模块复位使能。 1：正常工作；0：模块处于复位状态
21	TRIM	rw	0x1	CLKTRIM 模块复位使能。 1：正常工作；0：模块处于复位状态
20	RTC	rw	0x1	RTC 模块复位使能。 1：正常工作；0：模块处于复位状态
19	PCNT	rw	0x1	PCNT 模块复位使能。 1：正常工作；0：模块处于复位状态
18	RNG	rw	0x1	RNG 模块复位使能。 1：正常工作；0：模块处于复位状态
17	VC	rw	0x1	VC，LVD，模块复位使能。 1：正常工作；0：模块处于复位状态
16	ADC	rw	0x1	ADC 模块复位使能。 1：正常工作；0：模块处于复位状态
15	Reserved			保留位。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
14	PCA	rw	0x1	PCA 模块复位使能。 1：正常工作；0：模块处于复位状态
13	OPA	rw	0x1	OPA 模块复位使能。 1：正常工作；0：模块处于复位状态
12	Reserved			保留位。
11	TIM3	rw	0x1	TIM3 模块复位使能。 1：正常工作；0：模块处于复位状态
10	ADVTIM	rw	0x1	TIM456 模块复位使能。 1：正常工作；0：模块处于复位状态
9	LPTIM0	rw	0x1	LPTIM0 模块复位使能。 1：正常工作；0：模块处于复位状态
8	BASETIM	rw	0x1	TIM012 模块复位使能。 1：正常工作；0：模块处于复位状态
7	SPI1	rw	0x1	SPI1 模块复位使能。 1：正常工作；0：模块处于复位状态
6	Reserved			保留位。
5	I2C1	rw	0x1	I2C1 模块复位使能。 1：正常工作；0：模块处于复位状态
4	I2C0	rw	0x1	I2C0 模块复位使能。 1：正常工作；0：模块处于复位状态
3	LPUART1	rw	0x1	LPUART1 模块复位使能。 1：正常工作；0：模块处于复位状态
2	LPUART0	rw	0x1	LPUART0 模块复位使能。 1：正常工作；0：模块处于复位状态
1	UART1	rw	0x1	UART1 模块复位使能。 1：正常工作；0：模块处于复位状态
0	UART0	rw	0x1	UART0 模块复位使能。 1：正常工作；0：模块处于复位状态

6.2.3 外围模块复位控制寄存器 1(PERI_RESET1)

起始地址：0x40002000

地址偏移：0x2C

复位值：0x0000 0318

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						UART3	UART2	Reserved			LPTIM1	DAC	Reserved		
						R/W	R/W				R/W	R/W			

Bit	Field	Type	Reset	Description
31 :10	Reserved			保留位。
9	UART3	rw	0x1	UART3 模块复位使能。 1：正常工作；0：模块处于复位状态
8	UART2	rw	0x1	UART2 模块复位使能。 1：正常工作；0：模块处于复位状态
7 :5	Reserved			保留位。
4	LPTIM1	rw	0x1	LPTIM1 模块复位使能。 1：正常工作；0：模块处于复位状态
3	DAC	rw	0x1	DAC 模块复位使能。 1：正常工作；0：模块处于复位状态 此控制为不可读，读出值始终为 0，使用读修改写其他控制位时 DAC 模块处于复位状态
2 :0	Reserved			保留位。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

7 中断控制器 (NVIC)

7.1 概述

Cortex-M0+ 处理器内置了嵌套向量中断控制器（NVIC），支持最多 32 个中断请求（IRQ）输入，以及 1 个不可屏蔽中断（NMI）输入（在本产品系统中并未使用）。另外，处理器还支持多个内部异常。

每个异常源都有一个单独的异常编号，每种异常类型都有对应的优先级，有些异常的优先级是固定的，而有些则是可编程的。具体如表 11所示：

表 11: Cortex-M0+ 处理器中断一览

异常编号	异常类型	优先级	描述
1	复位	-3（最高）	复位
2	NMI	-2	不可屏蔽中断(在本系统中没有使用)
3	硬件错误	-1	错误处理异常
4-10	保留	NA	…
11	SVC	可编程	通过 SVC 指令调用管理程序
12-13	保留	NA	…
14	PendSV	可编程	系统服务的可挂起请求
15	SysTick	可编程	SysTick 定时器
16	中断 #0	可编程	外部中断 #0
17	中断 #1	可编程	外部中断 #1
…	…	…	…
47	中断 #31	可编程	外部中断 #31

本章节只对处理器的 32 个外部中断请求（中断 #0 到中断 #31）做详细介绍，处理器内部异常的具体情况可参考其他相关文档。同时，本章节只讨论处理器内核中 NVIC 的中断处理机制，外设模块自身的中断产生机制不在这里展开讨论。

7.2 中断优先级

每一个外部中断都对应一个优先级寄存器，每个优先级都是 2 位宽，并且使用中断优先级寄存器的最高两位，每个寄存器占 1 个字节（8 位）。在这个设定下，可以使用的优先级为 0x00（最高）、0x40、0x80 和 0xc0（最低）。

表 12: Cortex-M0+ 处理器中断一览

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
已使用		未使用，读出为 0					

如果处理器已经在运行另外一个中断处理，而新中断的优先级大于正在执行的，这时就会发生抢占。正在运行的中断处理会被暂停，转而执行新的中断，这个过程通常被称为中断嵌套。新的中断执行完毕后，之前的中断处理会继续执行，并且在其结束后返回到程序线程中。

如果处理器正在运行的另外一个中断处理的优先级相同或者更高，新的中断将会等待并且进入挂起状态。挂起的中断将会一直等到当前中断等级改变，例如，当前运行的中断处理完成返回后，当前优先级降低到了比挂起

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

中断还要小。

如果两个中断同时发生，并且它们的优先级相同，中断编号较小的中断将会首先执行。例如，如果中断 #0 和中断 #1 使能且具有相同的优先级，在它们同时被触发时，中断 #0 会首先执行。

7.3 中断向量表

当 Cortex-M0+ 处理器要处理中断服务请求时，它需要首先确定异常处理的起始地址，所需的信息叫做向量表，如图 12所示。向量表存储在存储器空间的开始位置，包含了系统中可用异常（中断）的异常（中断）向量，以及主栈指针（MSP）的初始值。

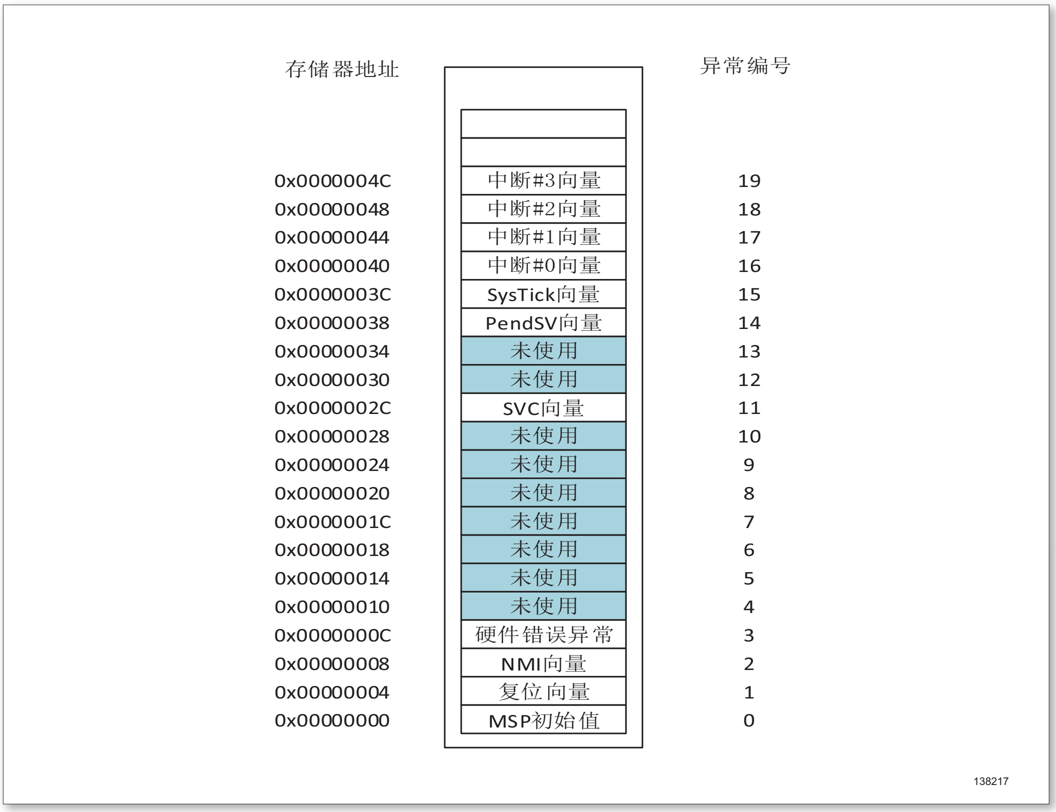

图 12. 中断向量表

其中，中断向量的存储顺序同中断编号一致，由于每个向量都是 1 个字（4 字节），中断向量的地址为中断编号乘 4，每个中断向量都是中断处理的起始地址。

7.4 中断输入和挂起行为

Cortex-M0+ 处理器的 NVIC 模块中，每一个中断输入都对应着一个挂起状态寄存器，且每个寄存器只有 1 位，用于保存中断请求，而不管这个请求有没有得到确认。当处理器开始处理这个中断时，硬件将会自动清除挂起状态位。

本系统的外设使用电平触发中断输出，当中断事件发生时，由于外设连接到了 NVIC 上，中断信号会得到确认。在处理器执行中断服务并且清除外设的中断信号以前，该信号会保持高电平。在 NVIC 内部，当检测到有中断发生时，该中断的挂起状态会被置位，当处理器接收该中断并且开始执行中断服务程序后，挂起状态就会被清除。该过程如图 13所示：

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

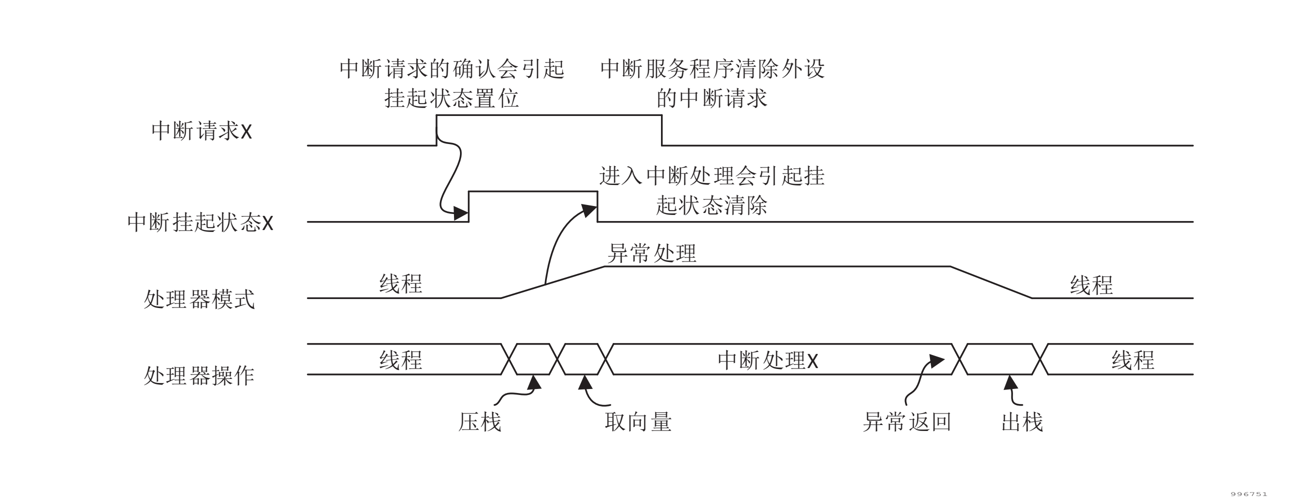

图 13. 中断激活和挂起状态

如果中断请求没有立即执行，并且在确认之前被软件清除了，这样处理器会忽略掉本次请求，并且不会执行中断处理。可以通过写 NVIC_CLRPEND 寄存器来清除中断挂起状态，这种处理在设置外设时非常有用，因为在设置以前，该外设可能已经产生了一个中断请求。

如果在软件清除挂起状态时，外设仍然保持着中断请求，挂起状态还会立即生成。该过程如图 14所示：

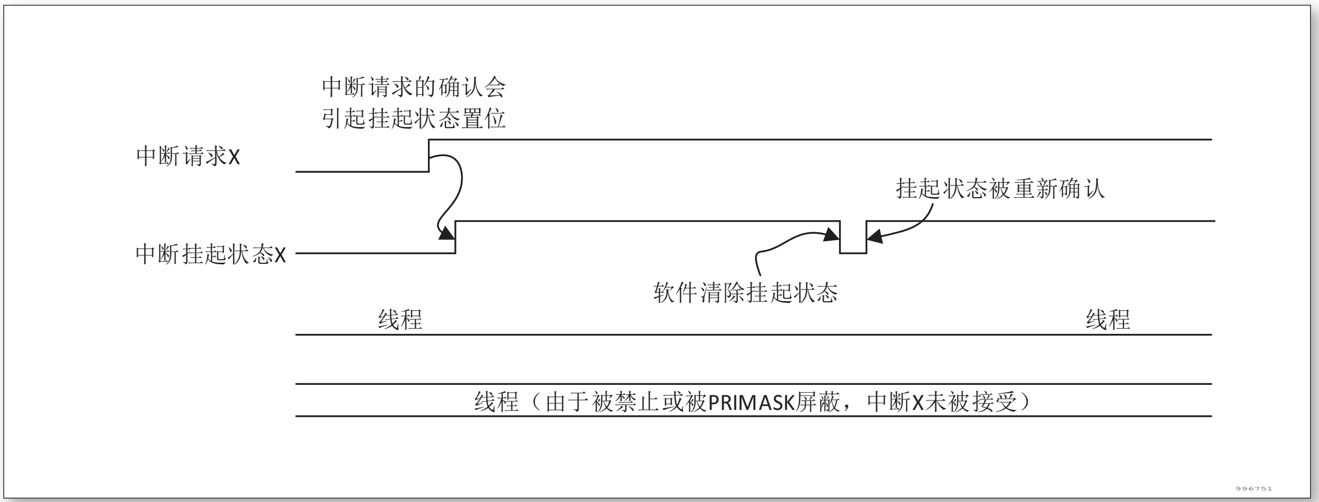

图 14. 中断挂起状态被清除然后被重新确认

如果外设产生的中断请求在异常处理时没有被清除，异常返回后挂起状态就会被又一次激活，这样中断服务程序会再次执行。该过程如图 15所示：

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

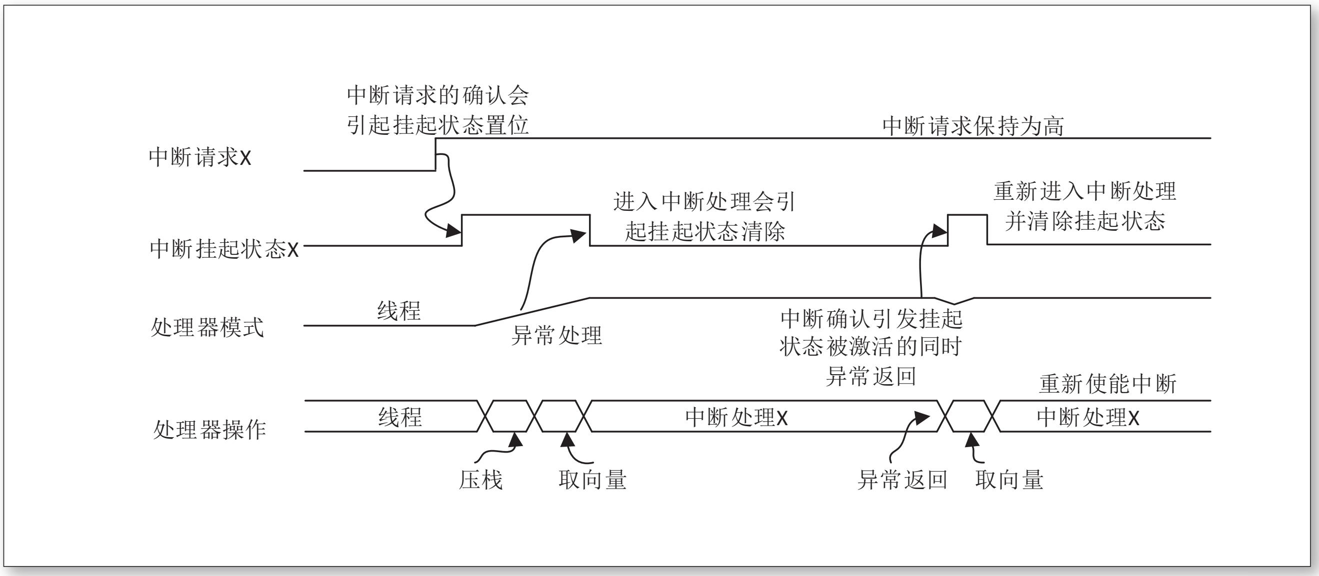

图 15. 中断退出时若中断请求保持高电平就会引起中断处理的再次执行

如果在终端服务程序执行的过程中产生外设中断请求，该请求会被当作新的中断请求，并且在本次中断退出后，还会引起中断服务程序的再次执行。该过程如图 16所示：

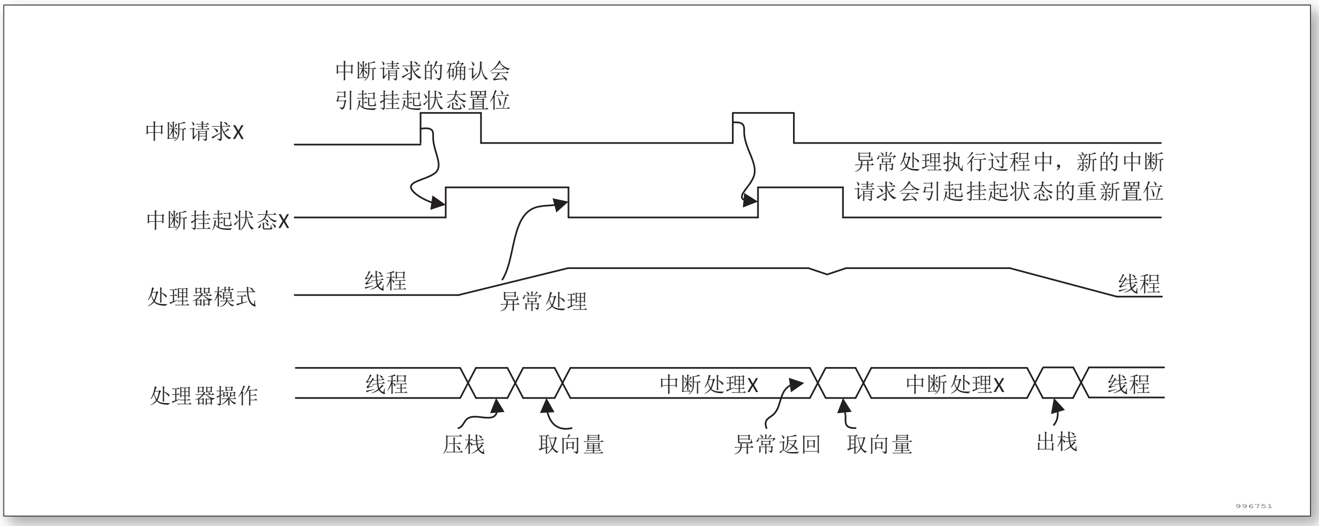

图 16. 中断处理中产生的中断挂起也可以被确认

7.5 中断等待

通常情况下，NVIC 的中断等待时间为 16 个周期。这个等待时间从中断确认的处理器时钟周期开始，一直到中断处理开始执行结束。计算中断等待需要具备以下前提：

- 该中断使能并且没有被 SCS_PRIMASK 或是其他正在执行的异常处理所屏蔽。
- 存储器系统没有任何等待状态，在中断处理、压栈、取向量或者中断处理开始时的取指都会用到总线传输，如果存储器系统需要等待，那么发生总线传输时产生的等待状态则可能使得中断延迟。

下面的几种情况可能会导致不同的中断等待：

- 中断的末尾连锁，如果中断返回时产生了另外一个中断请求，处理器就会跳过出栈和压栈过程，这样就减少了中断等待时间。

- 延迟到达，如果中断发生时，另外一个低优先级的中断正在进行压栈处理，由于延迟到达机制的存在，高优先级的中断会首先执行，这样也会导致高优先级中断的等待时间减小。

7.6 中断源

因为 Cortex-M0+ 处理器的 NVIC 支持最多 32 个外部中断，而在本系统中，外部中断源大于 32 个，所以部分外部中断复用在同一个 NVIC 中断输入上，其中 NMI（不可屏蔽中断）并没有使用。本系统所有外部中断源和 NVIC 中断输入的对应关系如表 13所示：

表 13: 外部中断与 NVIC 中断输入对应关系

NVIC 中断输入	外部中断源	Active 模式	Sleep 模式	DeepSleep 模式
中断 #0	PORTA	√	√	√
中断 #1	PORTB	√	√	√
中断 #2	PORTC/PORTE	√	√	√
中断 #3	PORTD/PORTF	√	√	√
中断 #4	DMAC	√	√	-
中断 #5	TIM3	√	√	-
中断 #6	UART0/2	√	√	-
中断 #7	UART1/3	√	√	-
中断 #8	LPUART0	√	√	√
中断 #9	LPUART1	√	√	√
中断 #10	-	-	-	-
中断 #11	SPI1	√	√	-
中断 #12	I2C0	√	√	-
中断 #13	I2C1	√	√	-
中断 #14	TIM0	√	√	-
中断 #15	TIM1	√	√	-
中断 #16	TIM2	√	√	-
中断 #17	LPTIM0/1	√	√	√
中断 #18	TIM4	√	√	-
中断 #19	TIM5	√	√	-
中断 #20	TIM6	√	√	-
中断 #21	PCA	√	√	-
中断 #22	WDT	√	√	√
中断 #23	RTC	√	√	√
中断 #24	ADC/DAC	√	√	-
中断 #25	PCNT	√	√	√
中断 #26	VC0	√	√	√
中断 #27	VC1/VC2	√	√	√
中断 #28	LVD	√	√	√
中断 #29	LCD	√	√	√

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

NVIC 中断输入	外部中断源	Active 模式	Sleep 模式	DeepSleep 模式
中断 #30	FLASH/RAM	√	√	-
中断 #31	CLKTRIM	√	√	√

注意：

-由于某些模块中断被复用于同一个 IRQ 中断源，当 CPU 进入该中断操作时，必须先判断是哪个模块产生的中断，再进行相对应的中断操作。

7.7 中断结构图

本系统的中断结构框图如图图 17。

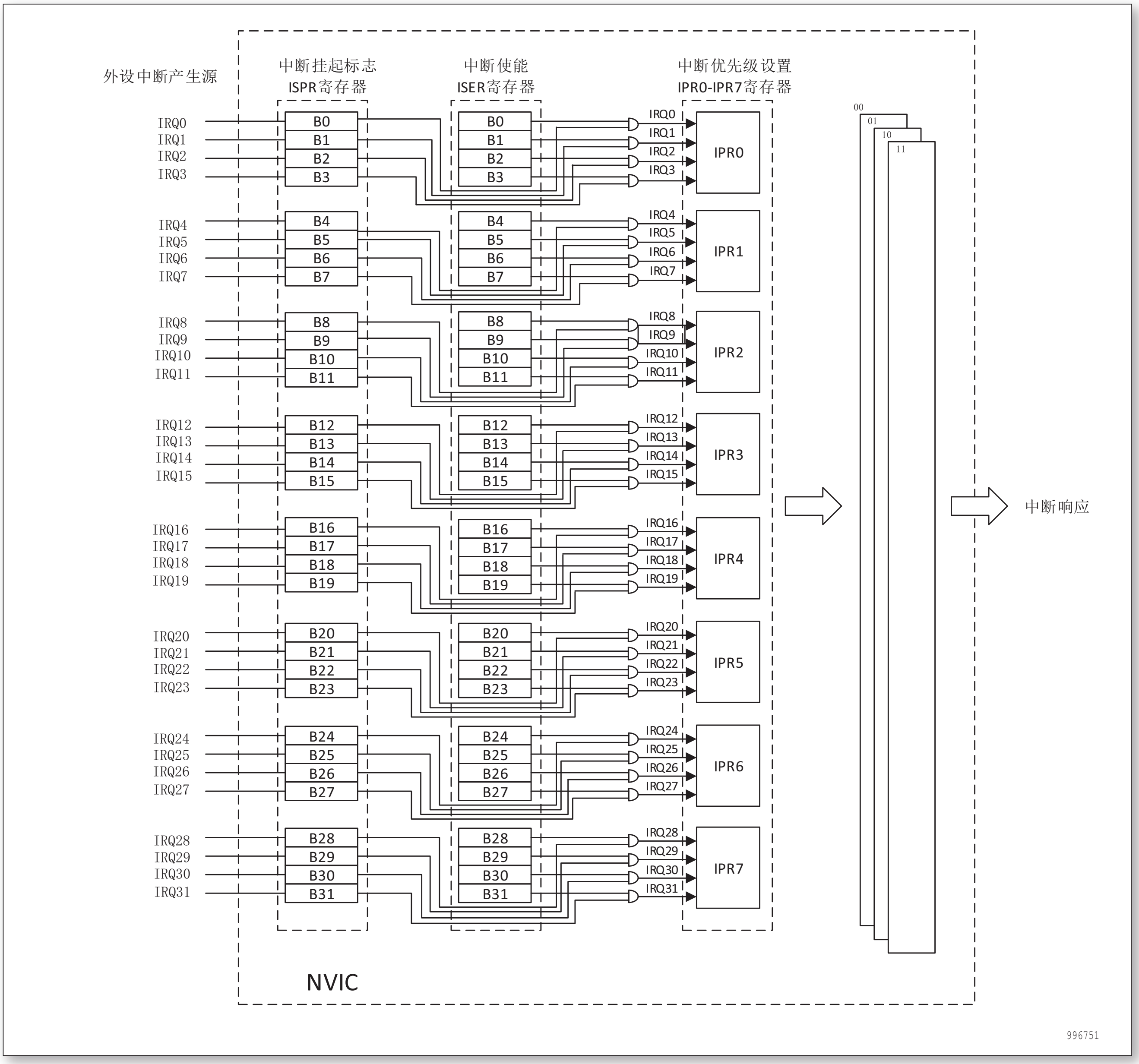

图 17. 中断结构图

本系统的中断结构框图如图 17 所示。需要注意几点：

- 外设中断源各自的中断使能没有在图中标出，这里只包括外设中断产生之后的中断信号逻辑框图。
- IRQ 有多于 1 个外设中断输入复用，必须分别读取这些外设的中断标志位来判定是哪个外设的中断。
- 如果外设中断源有高电平产生，无论 NVIC 中断使能寄存器 NVIC_ISER 置位与否，中断挂起寄存器 SCS_SEPEND 都会被置位，表示相应的外设中断源有中断产生。
- 只有中断使能寄存器 NVIC_ISER 置位，相应的中断 IRQ 才会给处理器相应，执行相应的中断程序。
- 在中断程序中必须清除外设中断源高电平中断信号，中断挂起寄存器 NVIC_ISPR 由硬件自动清除。
- 中断优先级寄存器 NVIC_IPR0- NVIC_IPR7 设置了 32 个中断源的优先级，00 优先级最高，11 优先级最低。当优先级相同时，优先级由中断编号决定，编号越小优先级越高。

7.8 NVIC 寄存器描述

表 14. NVIC 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x100	NVIC_ISER	中断请求使能寄存器	0x0000 0000	小节 7.8.1
0x180	NVIC_ICER	中断请求清除使能寄存器	0x0000 0000	小节 7.8.2
0x200	NVIC_ISPR	中断设置挂起寄存器	0x0000 0000	小节 7.8.3
0x280	NVIC_ICPR	中断清除挂起寄存器	0x0000 0000	小节 7.8.4
0x400	NVIC_IPR0	中断 #0-中断 #3 优先级寄存器	0x0000 0000	小节 7.8.5
0x404	NVIC_IPR1	中断 #4-中断 #7 优先级寄存器	0x0000 0000	小节 7.8.6
0x408	NVIC_IPR2	中断 #8-中断 #11 优先级寄存器	0x0000 0000	小节 7.8.7
0x40C	NVIC_IPR3	中断 #12-中断 #15 优先级寄存器	0x0000 0000	小节 7.8.8
0x410	NVIC_IPR4	中断 #16-中断 #19 优先级寄存器	0x0000 0000	小节 7.8.9
0x414	NVIC_IPR5	中断 #20-中断 #23 优先级寄存器	0x0000 0000	小节 7.8.10
0x418	NVIC_IPR6	中断 #24-中断 #27 优先级寄存器	0x0000 0000	小节 7.8.11
0x41C	NVIC_IPR7	中断 #28-中断 #31 优先级寄存器	0x0000 0000	小节 7.8.12
–	PRIMASK	中断屏蔽特殊寄存器	0x0000 0000	小节 7.8.13

7.8.1 中断请求使能寄存器 (NVIC_ISER)

起始地址：0xE000E000

偏移地址：0x100

复位值：0x00000000

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

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

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31: 0	SETENA[31:0]	rw	0x00	设置使能中断 #0 到中断 #31；写“1”置位，写“0”无效。 [0]:IRQ0 [1]:IRQ1 [2]:IRQ2 …… [31]:IRQ31

7.8.2 中断请求清除使能寄存器 (NVIC_ICER)

起始地址：0xE000E000

偏移地址：0x180

复位值：0x00000000

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

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

Bit	Field	Type	Reset	Description
31: 0	CLRENA	rw	0x00	清除使能中断 #0 到中断 #31；写“1”清零，写“0”无效。

7.8.3 中断设置挂起寄存器 (NVIC_ISPR)

起始地址：0xE000E000

偏移地址：0x200

复位值：0x00000000

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

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

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31: 0	SETPEND	rw	0x00	设置中断 #0 到中断 #31；写“1”置位，写“0”无效。 [0]:IRQ0 [1]:IRQ1 [2]:IRQ2 …… [31]:IRQ31

7.8.4 中断清除挂起寄存器 (NVIC_ICPR)

起始地址：0xE000E000

偏移地址：0x280

复位值：0x00000000

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

Bit	Field	Type	Reset	Description
31: 0	CLRPEND	rw	0x00	清除中断 #0 到中断 #31；写“1”清零，写“0”无效。 [0]:IRQ0 [1]:IRQ1 [2]:IRQ2 …… [31]:IRQ31

7.8.5 中断优先级寄存器 (NVIC_IPR0)

起始地址：0xE000E000

偏移地址：0x400

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IPR0[31:16]															
RW															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
IPR0[15:0]]															
RW															

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31: 0	IPR0[31:0]	rw	0x00	中断 #0 到中断 #3 的优先级。 [31:30]：中断 #3 的优先级 [23:22]：中断 #2 的优先级 [15:14]：中断 #1 的优先级 [7:6]：中断 #0 的优先级 其中，00 优先级最高，11 优先级最低。

7.8.6 中断优先级寄存器 (NVIC_IPR1)

起始地址：0xE000E000

偏移地址：0x404

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IPR1[31:16]															
RW															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IPR1[15:0]]															
RW															

Bit	Field	Type	Reset	Description
31: 0	IPR1[31:0]	rw	0x00	中断 #4 到中断 #7 的优先级。 [31:30]：中断 #7 的优先级 [23:22]：中断 #6 的优先级 [15:14]：中断 #5 的优先级 [7:6]：中断 #4 的优先级 其中，00 优先级最高，11 优先级最低。

7.8.7 中断优先级寄存器 (NVIC_IPR2)

起始地址：0xE000E000

偏移地址：0x408

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IPR2[31:16]															
RW															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IPR2[15:0]]															
RW															

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31: 0	IPR2[31:0]	rw	0x00	中断 #8 到中断 #11 的优先级。 [31:30]：中断 #11 的优先级 [23:22]：中断 #10 的优先级 [15:14]：中断 #9 的优先级 [7:6]：中断 #8 的优先级 其中，00 优先级最高，11 优先级最低。

7.8.8 中断优先级寄存器 (NVIC_IPR3)

起始地址：0xE000E000

偏移地址：0x40C

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IPR3[31:16]															
RW															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
IPR3[15:0]]															
RW															

Bit	Field	Type	Reset	Description
31: 0	IPR3[31:0]	rw	0x00	中断 #12 到中断 #15 的优先级。 [31:30]：中断 #15 的优先级 [23:22]：中断 #14 的优先级 [15:14]：中断 #13 的优先级 [7:6]：中断 #12 的优先级 其中，00 优先级最高，11 优先级最低。

7.8.9 中断优先级寄存器 (NVIC_IPR4)

起始地址：0xE000E000

偏移地址：0x410

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IPR4[31:16]															
RW															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
IPR4[15:0]]															
RW															

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31: 0	IPR4[31:0]	rw	0x00	中断 #16 到中断 #19 的优先级。 [31:30]：中断 #19 的优先级 [23:22]：中断 #18 的优先级 [15:14]：中断 #17 的优先级 [7:6]：中断 #16 的优先级 其中，00 优先级最高，11 优先级最低。

7.8.10 中断优先级寄存器 (NVIC_IPR5)

起始地址：0xE000E000

偏移地址：0x414

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IPR5[31:16]															
RW															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IPR5[15:0]]															
RW															

Bit	Field	Type	Reset	Description
31: 0	IPR5[31:0]	rw	0x00	中断 #20 到中断 #23 的优先级。 [31:30]：中断 #23 的优先级 [23:22]：中断 #22 的优先级 [15:14]：中断 #21 的优先级 [7:6]：中断 #20 的优先级 其中，00 优先级最高，11 优先级最低。

7.8.11 中断优先级寄存器 (NVIC_IPR6)

起始地址：0xE000E000

偏移地址：0x418

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IPR6[31:16]															
RW															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IPR6[15:0]]															
RW															

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31: 0	IPR6[31:0]	rw	0x00	中断 #24 到中断 #27 的优先级。 [31:30]：中断 #27 的优先级 [23:22]：中断 #26 的优先级 [15:14]：中断 #25 的优先级 [7:6]：中断 #24 的优先级 其中，00 优先级最高，11 优先级最低。

7.8.12 中断优先级寄存器 (NVIC_IPR7)

起始地址：0xE000E000

偏移地址：0x41C

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IPR7[31:16]															
RW															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
IPR7[15:0]]															
RW															

Bit	Field	Type	Reset	Description
31: 0	IPR7[31:0]	rw	0x00	中断 #28 到中断 #31 的优先级。 [31:30]：中断 #31 的优先级 [23:22]：中断 #30 的优先级 [15:14]：中断 #29 的优先级 [7:6]：中断 #28 的优先级 其中，00 优先级最高，11 优先级最低。

7.8.13 中断屏蔽特殊寄存器 (PRIMASK)

起始地址：—

偏移地址：—

复位值：0x00000000

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

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31:1	Reserved			保留。
0	PRIMASK	rw	0x00	置位后，除了 NMI 和硬件错误异常外的其他中断都会被屏蔽掉。 清零后，所有异常和中断都不会被屏蔽掉。 该特殊寄存器需要通过 MSR 和 MRS 特殊寄存器操作指令访问，也可以用改变处理器状态指令 CPS 访问。在处理对时间敏感的应用时，需要操作 PRIMASK 寄存器。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

8 端口控制器 (GPIO)

8.1 端口控制器简介

本系列有 50 个数字通用输入输出端口 PA[4:0], PA[14:8],PB[15:0], PC[15:0], PD[2:2], PF[1:0], PF[7:6],PF[11:11]。支持 50 个 GPIO，未封出的 GPIO 需要配置成输入，并使能上拉；其他封装未封出的 GPIO 也应该做相应配置。

模拟模块 ADC/VC/LVD/LCD 的输入输出信号，各功能模块（如 SPI，UART，I2C，Timer 等）的输入输出信号以及测试调试功能的输入输出信号都可以和数字通用输入输出端口复用。每个端口都可以配置成内部拉高 (pull up)/拉低 (pull down) 的输入, 高阻输入（floating input）, 推挽输出 (CMOS output)，开漏输出 (open drain output)，增强驱动能力输出。芯片复位后端口为高阻输入，目的是防止芯片被异常复位时，对外部器件产生异常动作。但为了避免高阻输入而产生的漏电，用户要在芯片启动之后对端口进行相应的配置（配置成内部拉高/拉低输入或者输出）。数字端口被配置成模拟端口后，数字功能被隔离，不能输出数字“1”和“0”，CPU 读取端口输入值寄存器的结果为“0”。

每个数字端口被配置为输入时，都可以提供外部中断，中断类型可以配置成高电平触发、低电平触发、上升沿触发、下降沿触发 4 种，查询 Px_STAT[n] 的中断标志位即可查出相应的中断触发端口。另外，每个数字端口的中断都可以把芯片从睡眠模式/深度睡眠模式唤醒到工作模式。

8.2 端口控制器主要特性

端口控制器支持以下特性：

- 端口输入值/输出值寄存器支持 FAST IO/AHB 总线读写。
- 其他寄存器支持 AHB 总线接口读写。
- 模拟功能管脚/SWD 调试管脚/数字通用管脚/数字功能管脚复用。
- 支持上拉/下拉/增强驱动/开漏输出功能选择。
- 支持工作模式/睡眠模式/深度睡眠模式下中断。
- 支持高电平/低电平/上升沿/下降沿触发中断。
- 支持位置位，位清零，位置位清零功能。

8.3 端口控制器功能描述

8.3.1 端口配置功能

每一个端口都可以根据系统需求通过配置寄存器 (PxADS) 把端口配置成模拟端口或数字端口。当配置为数字端口时，还可以配置寄存器实现以下特性。

1. 内部上拉（PxPU）/下拉（PxPU）。
2. 增强驱动输出 (PxDR)。
3. 开漏输出（PxOD）。
4. 方向选择（PxDIR）。
5. 输出高低电平选择 (PxOUT)，可通过 AHB/FASTIO 总线访问。
6. 输入电平状态（PxIN）可通过 AHB/FASTIO 总线访问。
7. 置位（PxBSET）。
8. 清零（PxBCLR）。
9. 置位清零（PxBSETCLR）。

注：以上特性在配置为模拟端口时无效。

端口的状态和寄存器配置的关系如下表：

表 15: 端口状态真值表

PxADS	PxDIR	PxOUT	PxIN	PxBSET	PxBCLR	PxBSETCLR	PxPU	PxPD	PxOD	PxDR	Px_SEL	IO 方向	IO 状态
1	W	W	0	W	W	W	W	W	W	W	W	输入/输出	模拟
0	1	W	X	W	W	W	0	0	W	W	0	输入	浮空
0	1	W	0	W	W	W	0	1	W	W	0	输入	下拉
0	1	W	1	W	W	W	1	0	W	W	0	输入	上拉
0	1	W	1	W	W	W	1	1	W	W	0	输入	上拉
0	1	W	1	W	W	W	W	W	W	W	0	输入	1
0	1	W	0	W	W	W	W	W	W	W	0	输入	0
0	0	1	1	0	0	0	W	W	0	W	0	输出	1
0	0	0	0	0	0	0	W	W	0	W	0	输出	0
0	0	W	1	1	0	0	W	W	0	W	0	输出	1
0	0	W	0	0	1	0	W	W	0	W	0	输出	0
0	0	W	(SET)1/(CLR)0	0	0	1	W	W	0	W	0	输出	(SET)1/(CLR)0
0	0	0	0	0	0	0	W	W	1	W	0	输出	0
0	0	1	X	0	0	0	0	0	1	W	0	输出	Z
0	0	1	0	0	0	0	0	1	1	W	0	输出	0
0	0	1	1	0	0	0	1	0	1	W	0	输出	1
0	0	1	1	0	0	0	1	1	1	W	0	输出	1

端口电路结构如下图所示：

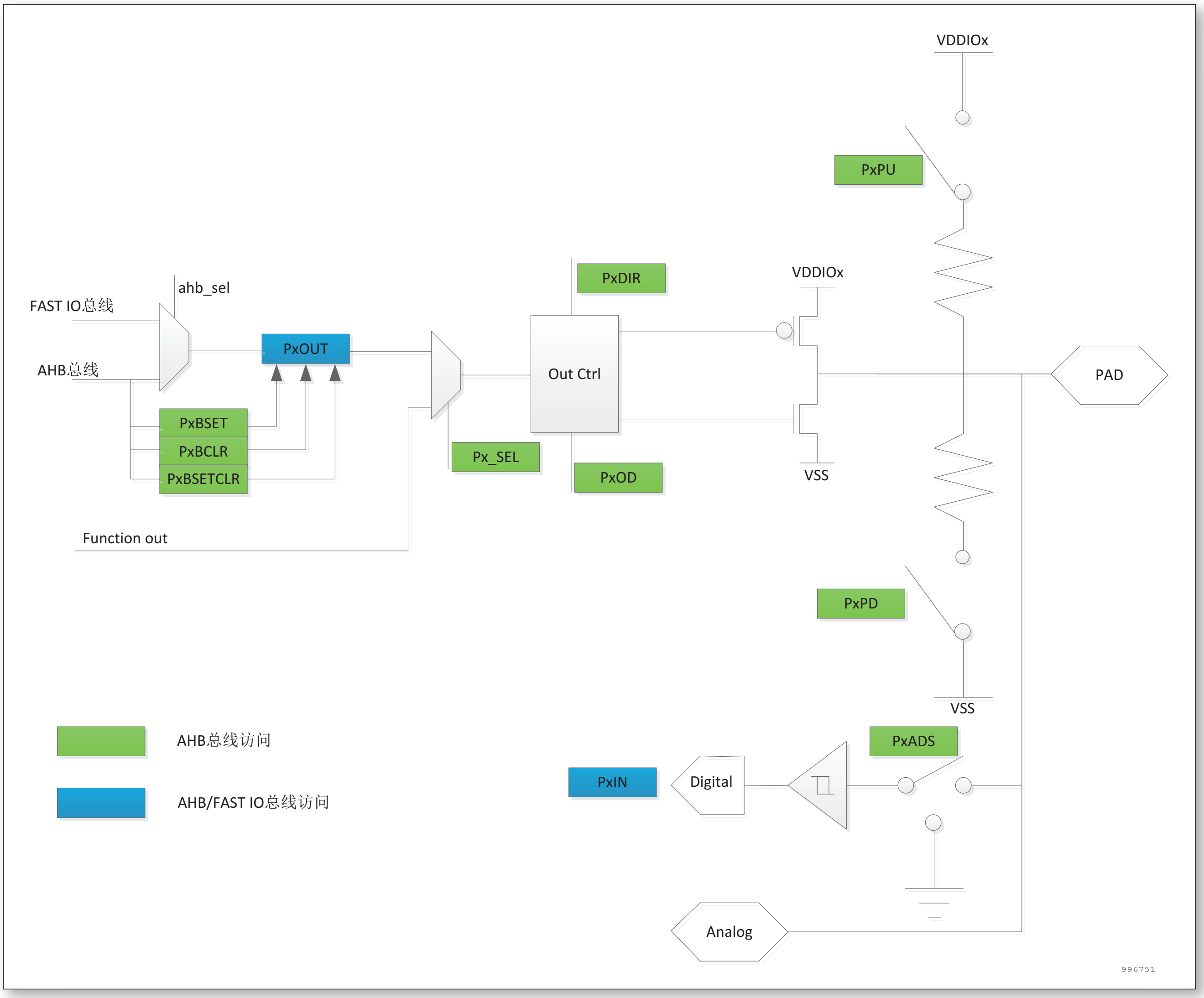

图 18. 端口电路示意图

8.3.2 端口数字复用功能

端口数字复用是端口控制器的主要功能之一。通过配置寄存器，可以灵活的把端口配置成测试调试端口/数字通用端口/数字功能端口。

如下表所示：Px_SEL 寄存器用于数字通用端口/数字功能端口切换，每一个端口都可以独立配置成系统所需的功能端口。调试端口的配置信息请参考相关章节。

表 16: 端口复用表

PSEL	1	2	3	4	5	6	7
PA0	UART1_CTS	LPUART1_TXD	TIM0_ETR	VC0_OUT	TIM1_CHA	TIM3_ETR	TIM0_CHA
PA1	UART1_RTS	LPUART1_RXD	TIM0_CHB	TIM1_ETR	TIM1_CHB	HCLK_OUT	SPI1_MOSI
PA2	UART1_TXD	TIM0_CHA	VC1_OUT	TIM1_CHA	TIM2_CHA	PCLK_OUT	SPI1_MISO
PA3	UART1_RXD	TIM0_GATE	TIM1_CHB	TIM2_CHB	SPI1_CS	TIM3_CH1A	TIM5_CHA

PSEL	1	2	3	4	5	6	7
PA4		UART1_TXD	PCA_CH4	TIM2_ETR	TIM5_CHA	LVD_OUT	TIM3_CH2B
PA8	UART0_TXD	TIM3_CH0A			TIM1_GATE	TIM4_CHA	TIM3_BK
PA9	UART0_TXD	TIM3_CH1A	TIM0_BK	I2C0_SCL		HCLK_OUT	TIM5_CHA
PA10	UART0_RXD	TIM3_CH2A	TIM2_BK	I2C0_SDA	TIM2_GATE	PCLK_OUT	TIM6_CHA
PA11	UART0_CTS	TIM3_GATE	I2C1_SCL		VC0_OUT		TIM4_CHB
PA12	UART0_RTS	TIM3_ETR	I2C1_SDA		VC1_OUT		PCNT_S0
PA13	IR_OUT	UART0_RXD	LVD_OUT	TIM3_ETR	RTC_1HZ	PCNT_S1	VC2_OUT
PA14	UART1_TXD	UART0_TXD	TIM3_CH2A	LVD_OUT	RCH_OUT	RCL_OUT	PLL_OUT
PB0	PCA_CH2	TIM3_CH1B	LPUART0_ TXD	TIM5_CHB	RCH_OUT	RCL_OUT	PLL_OUT
PB1	PCA_CH3	PCLK_OUT	TIM3_CH2B	TIM6_CHB	LPUART0_ RTS	VC2_OUT	TCLK_OUT
PB2	LPTIM_ TOG	PCA_ECI	LPUART1_ TXD	TIM4_CHA	TIM1_BK	TIM0_BK	TIM2_BK
PB3		TIM0_CHB	TIM1_GATE	TIM3_CH0A	LPTIM_ GATE	XTL_OUT	XTH_OUT
PB4		PCA_CH0	TIM2_BK	UART0_CTS	TIM2_GATE	TIM3_CH0B	LPTIM_ ETR
PB5			TIM1_BK	PCA_CH1	LPTIM_ GATE	PCNT_S0	UART0_RTS
PB6	I2C0_SCL	UART0_TXD	TIM1_CHB	TIM0_CHA	LPTIM_ ETR	TIM3_CH0A	LPTIM_ TOG
PB7	I2C0_SDA	UART0_RXD	TIM2_CHB	LPUART1_ CTS	TIM0_CHB	LPTIM_ TOGN	PCNT_S1
PB8	I2C0_SCL	TIM1_CHA		TIM2_CHA	TIM0_GATE	TIM3_CH2A	UART0_TXD
PB9	I2C0_SDA	IR_OUT	SPI1_CS	TIM2_CHA		TIM2_CHB	UART0_RXD
PB10	I2C1_SCL	SPI1_SCK	TIM1_CHA	LPUART0_ TXD	TIM3_CH1A	LPUART1_ RTS	UART1_RTS
PB11	I2C1_SDA	TIM1_CHB	LPUART0_ RXD	TIM2_GATE	TIM6_CHA	LPUART1_ CTS	UART1_CTS
PB12	SPI1_CS	TIM3_BK	LPUART0_ TXD	TIM0_BK		LPUART0_ RTS	TIM6_CHA
PB13	SPI1_SCK	I2C1_SCL	TIM3_CH0B	LPUART0_ CTS	TIM1_CHA	TIM1_GATE	TIM6_CHB
PB14	SPI1_MISO	I2C1_SDA	TIM3_CH1B	TIM0_CHA	RTC_1HZ	LPUART0_ RTS	TIM1_BK
PB15	SPI1_MOSI	TIM3_CH2B	TIM0_CHB	TIM0_GATE			LPUART1_ RXD
PC0	LPTIM_ GATE	PCNT_S0	UART1_CTS	UART2_RTS			
PC1	LPTIM_ TOG	TIM5_CHB	UART1_RTS	PCNT_S0FO		UART2_CTS	
PC2	SPI1_MISO	LPTIM_ TOGN	PCNT_S1	UART2_RXD			

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

PSEL	1	2	3	4	5	6	7
PC3	SPI1_MOSI	LPTIM_ ETR	LPTIM_ TOGN	PCNT_S1FO	UART2_TXD		
PC4	LPUART0_ TXD	TIM2_ETR	IR_OUT	VC2_OUT			
PC5	LPUART0_ RXD	TIM6_CHB	PCA_CH4				
PC6	PCA_CH0	TIM4_CHA	TIM2_CHA	LPTIM1_ GATE		UART3_RXD	
PC7	PCA_CH1	TIM5_CHA	TIM2_CHB	LPTIM1_ ETR		UART3_TXD	
PC8	PCA_CH2	TIM6_CHA	TIM2_ETR	LPTIM1_ TOG		UART3_CTS	
PC9	PCA_CH3	TIM4_CHB	TIM1_ETR	LPTIM1_ TOGN		UART3_RTS	
PC10	LPUART1_ TXD	LPUART0_ TXD	PCA_CH2				
PC11	LPUART1_ RXD	LPUART0_ RXD	PCA_CH3	PCNT_S0FO			
PC12	LPUART0_ TXD	LPUART1_ TXD	PCA_CH4	PCNT_S1FO			
PC13		RTC_1HZ	TIM3_CH1B				
PC14							
PC15							
PD2	PCA_ECI	LPUART0_ RTS	TIM1_ETR				
PF0	I2C0_SDA		UART1_TXD				
PF1	I2C0_SCL	TIM4_CHB	UART1_RXD				
PF6	I2C1_SCL	LPUART1_ CTS	UART0_CTS				
PF7	I2C1_SDA	LPUART1_ RTS	UART0_RTS				
PF11							

8.3.3 端口中断功能

每一个数字通用端口都可以由外部信号源产生中断，外部信号源可以是高电平/低电平/上升沿/下降沿 4 种类型的信号，分别对应的中断使能寄存器为高电平中断使能寄存器 (PxHIE)/低电平中断使能寄存器 (PxLIE)/上升沿中断使能寄存器 (PxRIE)/下降沿中断使能寄存器 (PxFIE)。当中断触发时，通过查询中断状态寄存器 (Px_STAT) 就可以判断是哪一个端口触发了中断，通过清零中断清除寄存器 (Px_ICLR) 就可以清除对应的中断状态标志位。

8.4 端口配置操作流程

8.4.1 端口复用配置为模拟端口操作流程

1. 设置寄存器 PxADS[n] 为 1。

8.4.2 端口复用配置为数字通用端口操作流程

1. 设置寄存器 PxADS[n] 为 0。
2. 设置寄存器 Px_SEL 为 0。
3. 设置寄存器 PxDIR[n] 为 1：端口方向为输入，CPU 可以读取端口的状态 PxIN[n]。
4. 设置寄存器 PxDIR[n] 为 0：端口方向为输出。
5. 设置寄存器 PxOUT[n] 为 1：端口输出高电平。
6. 设置寄存器 PxOUT[n] 为 0：端口输出低电平。

8.4.3 端口复用配置为数字功能端口操作流程

1. 设置寄存器 PxADS[n] 为 0。
2. 设置寄存器 Px_SEL 为 1-7 (根据系统需求，参考端口复用表)。
3. 设置寄存器 PxDIR[n](根据系统需求)。
4. 设置寄存器 PxPU[n]/PxPD[n]/PxOD[n](根据系统需求)。

8.4.4 端口复用配置为调试测试端口操作流程

参考测试调试相关章节。

8.4.5 端口复用配置为红外输出信号操作流程

端口 PA13,PB09,PC04 可以把频率为 38K 的内部时钟信号调制成红外信号输出。

1. 设置寄存器 PAADS[13]/PBADS[9]/PCADS[4] 为 0。
2. 设置寄存器 PA13_SEL = 1/PB09_SEL = 2/PC04_SEL = 3。
3. 设置寄存器 PADIR[13] = 0/PBDIR[9] = 0/PCDIR[4] = 0：端口方向为输出。
4. 设置寄存器 GPIO_CTRL1 的 bit14 选择红外信号输出极性。
5. 设置寄存器 PAOUT[13]/PBOUT[9]/PCOUT[4] 门控红外信号的输出。

8.4.6 端口高电平中断操作流

1. 设置寄存器 PxADS[n] 为 0。
2. 设置寄存器 Px_SEL 为 0。
3. 设置寄存器 PxDIR[n] 为 1。
4. 设置寄存器 Px_HIE[n] 为 1。
5. 中断触发后读取中断状态寄存器 Px_STAT[n]。
6. 设置寄存器 Px_ICLR[n] 为 0 清除中断状态寄存器 Px_STAT[n]。

8.4.7 端口低电平中断操作流程

1. 设置寄存器 PxADS[n] 为 0。
2. 设置寄存器 Px_SEL 为 0。
3. 设置寄存器 PxDIR[n] 为 1。
4. 设置寄存器 Px_LIE[n] 为 1。
5. 中断触发后读取中断状态寄存器 Px_STAT[n]。
6. 设置寄存器 Px_ICLR[n] 为 0 清除中断状态寄存器 Px_STAT[n]。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

8.4.8 端口上升沿中断操作流程

1. 设置寄存器 PxADS[n] 为 0。
2. 设置寄存器 Px_SEL 为 0。
3. 设置寄存器 PxDIR[n] 为 1。
4. 设置寄存器 Px_ RIE[n] 为 1。
5. 中断触发后读取中断状态寄存器 Px_STAT[n]。
6. 设置寄存器 Px_ICLR[n] 为 0 清除中断状态寄存器 Px_STAT[n]。

8.4.9 端口下降沿中断操作流程

1. 设置寄存器 PxADS[n] 为 0。
2. 设置寄存器 Px_SEL 为 0。
3. 设置寄存器 PxDIR[n] 为 1。
4. 设置寄存器 Px_FIE[n] 为 1。
5. 中断触发后读取中断状态寄存器 Px_STAT[n]。
6. 设置寄存器 Px_ICLR[n] 为 0 清除中断状态寄存器 Px_STAT[n]。

8.4.10 端口上拉使能配置操作流程

1. 设置寄存器 PxPU[n] 为 1。

8.4.11 端口下拉使能配置操作流程

1. 设置寄存器 PxPU[n] 为 0。
2. 设置寄存器 PxPD[n] 为 1。

注：当 PxPU[n],PxPD[n] 同时置 1 时，PxPU[n] 优先级高，PxPD[n] 无效。

8.4.12 端口增强驱动配置操作流程

1. 设置寄存器 PxDR[n] 为 0。

8.4.13 端口开漏输出配置操作流程

1. 设置寄存器 PxOD[n] 为 1。

8.4.14 端口位置位操作流程

1. 设置寄存器 PxBSET[n] 为 1。

8.4.15 端口位清零操作流程

1. 设置寄存器 PxBCLR[n] 为 1。

8.4.16 端口位置位清零操作流程

1. 设置寄存器 PxBSETCLR[n] 为 1。

8.5 端口控制器寄存器描述

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

表 17. 端口控制器寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x00	PA0_SEL	端口 PA0 功能配置寄存器	0x0000 0000	小节 8.8.1
0x04	PA1_SEL	端口 PA1 功能配置寄存器	0x0000 0000	小节 8.8.2
0x08	PA2_SEL	端口 PA2 功能配置寄存器	0x0000 0000	小节 8.8.3
0x0c	PA3_SEL	端口 PA3 功能配置寄存器	0x0000 0000	小节 8.8.4
0x10	PA4_SEL	端口 PA4 功能配置寄存器	0x0000 0000	小节 8.8.5
0x20	PA8_SEL	端口 PA8 功能配置寄存器	0x0000 0000	小节 8.8.6
0x24	PA9_SEL	端口 PA9 功能配置寄存器	0x0000 0000	小节 8.8.7
0x28	PA10_SEL	端口 PA10 功能配置寄存器	0x0000 0000	小节 8.8.8
0x2c	PA11_SEL	端口 PA11 功能配置寄存器	0x0000 0000	小节 8.8.9
0x30	PA12_SEL	端口 PA12 功能配置寄存器	0x0000 0000	小节 8.8.10
0x34	PA13_SEL	端口 PA13 功能配置寄存器	0x0000 0000	小节 8.8.11
0x38	PA14_SEL	端口 PA14 功能配置寄存器	0x0000 0000	小节 8.8.12
0x40	PB0_SEL	端口 PB0 功能配置寄存器	0x0000 0000	小节 8.9.1
0x44	PB1_SEL	端口 PB1 功能配置寄存器	0x0000 0000	小节 8.9.2
0x48	PB2_SEL	端口 PB2 功能配置寄存器	0x0000 0000	小节 8.9.3
0x4c	PB3_SEL	端口 PB3 功能配置寄存器	0x0000 0000	小节 8.9.4
0x50	PB4_SEL	端口 PB4 功能配置寄存器	0x0000 0000	小节 8.9.5
0x54	PB5_SEL	端口 PB5 功能配置寄存器	0x0000 0000	小节 8.9.6
0x58	PB6_SEL	端口 PB6 功能配置寄存器	0x0000 0000	小节 8.9.7
0x5c	PB7_SEL	端口 PB7 功能配置寄存器	0x0000 0000	小节 8.9.8
0x60	PB8_SEL	端口 PB8 功能配置寄存器	0x0000 0000	小节 8.9.9
0x64	PB9_SEL	端口 PB9 功能配置寄存器	0x0000 0000	小节 8.9.10
0x68	PB10_SEL	端口 PB10 功能配置寄存器	0x0000 0000	小节 8.9.11
0x6c	PB11_SEL	端口 PB11 功能配置寄存器	0x0000 0000	小节 8.9.12
0x70	PB12_SEL	端口 PB12 功能配置寄存器	0x0000 0000	小节 8.9.13
0x74	PB13_SEL	端口 PB13 功能配置寄存器	0x0000 0000	小节 8.9.14
0x78	PB14_SEL	端口 PB14 功能配置寄存器	0x0000 0000	小节 8.9.15
0x7c	PB15_SEL	端口 PB15 功能配置寄存器	0x0000 0000	小节 8.9.16
0x80	PC0_SEL	端口 PC0 功能配置寄存器	0x0000 0000	小节 8.10.1
0x84	PC1_SEL	端口 PC1 功能配置寄存器	0x0000 0000	小节 8.10.2
0x88	PC2_SEL	端口 PC2 功能配置寄存器	0x0000 0000	小节 8.10.3
0x8c	PC3_SEL	端口 PC3 功能配置寄存器	0x0000 0000	小节 8.10.4
0x90	PC4_SEL	端口 PC4 功能配置寄存器	0x0000 0000	小节 8.10.5
0x94	PC5_SEL	端口 PC5 功能配置寄存器	0x0000 0000	小节 8.10.6
0x98	PC6_SEL	端口 PC6 功能配置寄存器	0x0000 0000	小节 8.10.7
0x9c	PC7_SEL	端口 PC7 功能配置寄存器	0x0000 0000	小节 8.10.8

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Offset	Acronym	Register Name	Reset	Section
0xa0	PC8_SEL	端口 PC8 功能配置寄存器	0x0000 0000	小节 8.10.9
0xa4	PC9_SEL	端口 PC9 功能配置寄存器	0x0000 0000	小节 8.10.10
0xa8	PC10_SEL	端口 PC10 功能配置寄存器	0x0000 0000	小节 8.10.11
0xac	PC11_SEL	端口 PC11 功能配置寄存器	0x0000 0000	小节 8.10.12
0xb0	PC12_SEL	端口 PC12 功能配置寄存器	0x0000 0000	小节 8.10.13
0xb4	PC13_SEL	端口 PC13 功能配置寄存器	0x0000 0000	小节 8.10.14
0xb8	PC14_SEL	端口 PC14 功能配置寄存器	0x0000 0000	小节 8.10.15
0xbc	PC15_SEL	端口 PC15 功能配置寄存器	0x0000 0000	小节 8.10.16
0xc8	PD2_SEL	端口 PD2 功能配置寄存器	0x0000 0000	小节 8.11.1
0x100	PADIR	端口 PA 输入输出配置寄存器	0xffff ffff	小节 8.6.1
0x104	PAIN	端口 PA 输入值寄存器	0x—- —-	小节 8.6.2
0x108	PAOUT	端口 PA 输出值配置寄存器	0x—- —-	小节 8.6.3
0x10c	PAADS	端口 PA 数模配置寄存器	0x0000 0000	小节 8.6.4
0x110	PABSET	端口 PA 位置位寄存器	0x0000 0000	小节 8.6.5
0x114	PABCLR	端口 PA 位清零寄存器	0x0000 0000	小节 8.6.6
0x118	PABSETCLR	端口 PA 位置位清零寄存器	0x0000 0000	小节 8.6.7
0x11c	PADR	端口 PA 驱动能力配置寄存器	0x0000 0000	小节 8.6.9
0x120	PAPU	端口 PA 上拉使能配置寄存器	0x0000 0000	小节 8.6.8
0x124	PAPD	端口 PA 下拉使能配置寄存器	0x0000 0000	小节 8.6.10
0x12c	PAOD	端口 PA 开漏输出配置寄存器	0x0000 0000	小节 8.6.11
0x130	PAHIE	端口 PA 高电平中断使能配置寄存器	0x0000 0000	小节 8.6.12
0x134	PALIE	端口 PA 低电平中断使能配置寄存器	0x0000 0000	小节 8.6.13
0x138	PARIE	端口 PA 上升沿中断使能配置寄存器	0x0000 0000	小节 8.6.14
0x13c	PAFIE	端口 PA 下降沿中断使能配置寄存器	0x0000 0000	小节 8.6.15
0x200	PA_STAT	端口 PA 中断状态寄存器	0x0000 0000	小节 8.6.16
0x210	PA_ICLR	端口 PA 中断清除寄存器	0xffff ffff	小节 8.6.17
0x140	PBDIR	端口 PB 输入输出配置寄存器	0xffff ffff	小节 8.6.1
0x144	PBIN	端口 PB 输入值寄存器	0x—- —-	小节 8.6.2
0x148	PBOUT	端口 PB 输出值配置寄存器	0x—- —-	小节 8.6.3
0x14c	PBADS	端口 PB 数模配置寄存器	0x0000 0000	小节 8.6.4
0x150	PBBSET	端口 PB 位置位寄存器	0x0000 0000	小节 8.6.5
0x154	PBBCLR	端口 PB 位清零寄存器	0x0000 0000	小节 8.6.6
0x158	PBBSETCLR	端口 PB 位置位清零寄存器	0x0000 0000	小节 8.6.7

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Offset	Acronym	Register Name	Reset	Section
0x15c	PBDR	端口 PB 驱动能力配置寄存器	0x0000 0000	小节 8.6.9
0x160	PBPU	端口 PB 上拉使能配置寄存器	0x0000 0000	小节 8.6.8
0x164	PBPD	端口 PB 下拉使能配置寄存器	0x0000 0000	小节 8.6.10
0x16c	PBOD	端口 PB 开漏输出配置寄存器	0x0000 0000	小节 8.6.11
0x170	PBHIE	端口 PB 高电平中断使能配置寄存器	0x0000 0000	小节 8.6.12
0x174	PBLIE	端口 PB 低电平中断使能配置寄存器	0x0000 0000	小节 8.6.13
0x178	PBRIE	端口 PB 上升沿中断使能配置寄存器	0x0000 0000	小节 8.6.14
0x17c	PBFIE	端口 PB 下降沿中断使能配置寄存器	0x0000 0000	小节 8.6.15
0x240	PB_STAT	端口 PB 中断状态寄存器	0x0000 0000	小节 8.6.16
0x250	PB_ICLR	端口 PB 中断清除寄存器	0xffff ffff	小节 8.6.17
0x180	PCDIR	端口 PC 输入输出配置寄存器	0xffff ffff	小节 8.6.1
0x184	PCIN	端口 PC 输入值寄存器	0x—- —-	小节 8.6.2
0x188	PCOUT	端口 PC 输出值配置寄存器	0x—- —-	小节 8.6.3
0x18c	PCADS	端口 PC 数模配置寄存器	0x0000 0000	小节 8.6.4
0x190	PCBSET	端口 PC 位置位寄存器	0x0000 0000	小节 8.6.5
0x194	PCBCLR	端口 PC 位清零寄存器	0x0000 0000	小节 8.6.6
0x198	PCBSETCLR	端口 PC 位置位清零寄存器	0x0000 0000	小节 8.6.7
0x19c	PCDR	端口 PC 驱动能力配置寄存器	0x0000 0000	小节 8.6.9
0x1a0	PCPU	端口 PC 上拉使能配置寄存器	0x0000 0000	小节 8.6.8
0x1a4	PCPD	端口 PC 下拉使能配置寄存器	0x0000 0000	小节 8.6.10
0x1ac	PCOD	端口 PC 开漏输出配置寄存器	0x0000 0000	小节 8.6.11
0x1b0	PCHIE	端口 PC 高电平中断使能配置寄存器	0x0000 0000	小节 8.6.12
0x1b4	PCLIE	端口 PC 低电平中断使能配置寄存器	0x0000 0000	小节 8.6.13
0x1b8	PCRIE	端口 PC 上升沿中断使能配置寄存器	0x0000 0000	小节 8.6.14
0x1bc	PCFIE	端口 PC 下降沿中断使能配置寄存器	0x0000 0000	小节 8.6.15
0x280	PC_STAT	端口 PC 中断状态寄存器	0x0000 0000	小节 8.6.16
0x290	PC_ICLR	端口 PC 中断清除寄存器	0xffff ffff	小节 8.6.17
0x1c0	PDDIR	端口 PD 输入输出配置寄存器	0xffff ffff	小节 8.6.1
0x1c4	PDIN	端口 PD 输入值寄存器	0x—- —-	小节 8.6.2

Offset	Acronym	Register Name	Reset	Section
0x1c8	PDOUT	端口 PD 输出值配置寄存器	0x—- —-	小节 8.6.3
0x1cc	PDADS	端口 PD 数模配置寄存器	0x0000 0000	小节 8.6.4
0x1d0	PDBSET	端口 PD 位置位寄存器	0x0000 0000	小节 8.6.5
0x1d4	PDBCLR	端口 PD 位清零寄存器	0x0000 0000	小节 8.6.6
0x1d8	PDBSETCLR	端口 PD 位置位清零寄存器	0x0000 0000	小节 8.6.7
0x1dc	PDDR	端口 PD 驱动能力配置寄存器	0x0000 0000	小节 8.6.9
0x1e0	PDPU	端口 PD 上拉使能配置寄存器	0x0000 0000	小节 8.6.8
0x1e4	PDPD	端口 PD 下拉使能配置寄存器	0x0000 0000	小节 8.6.10
0x1ec	PDOD	端口 PD 开漏输出配置寄存器	0x0000 0000	小节 8.6.11
0x1f0	PDHIE	端口 PD 高电平中断使能配置寄存器	0x0000 0000	小节 8.6.12
0x1f4	PDLIE	端口 PD 低电平中断使能配置寄存器	0x0000 0000	小节 8.6.13
0x1f8	PDRIE	端口 PD 上升沿中断使能配置寄存器	0x0000 0000	小节 8.6.14
0x1fc	PDFIE	端口 PD 下降沿中断使能配置寄存器	0x0000 0000	小节 8.6.15
0x2c0	PD_STAT	端口 PD 中断状态寄存器	0x0000 0000	小节 8.6.16
0x2d0	PD_ICLR	端口 PD 中断清除寄存器	0xffff ffff	小节 8.6.17
0x300	GPIO_CTRL0	端口辅助功能配置寄存器 0	0x0000 0001	小节 8.7.1
0x304	GPIO_CTRL1	端口辅助功能配置寄存器 1	0x0000 0000	小节 8.7.2
0x308	GPIO_CTRL2	端口辅助功能配置寄存器 2	0x0000 0000	小节 8.7.3
0x30c	GPIO_TIMGS	端口辅助功能定时器门控选择	0x0000 0000	小节 8.7.4
0x310	GPIO_TIMES	端口辅助功能定时器 ETR 选择	0x0000 0000	小节 8.7.5
0x314	GPIO_TIMCPS	端口辅助功能定时器捕获输入选择	0x0000 0000	小节 8.7.6
0x318	GPIO_PCAS	端口辅助功能 PCA 捕获选择	0x0000 0000	小节 8.7.7
0x31c	GPIO_PCNT	端口辅助功能 PCNT 输入选择	0x0000 0000	小节 8.7.8
0x1040	PF0_SEL	端口 PF0 功能配置寄存器	0x0000 0000	小节 8.12.1
0x1044	PF1_SEL	端口 PF1 功能配置寄存器	0x0000 0000	小节 8.12.2
0x1058	PF6_SEL	端口 PF6 功能配置寄存器	0x0000 0000	小节 8.12.3
0x105c	PF7_SEL	端口 PF7 功能配置寄存器	0x0000 0000	小节 8.12.4
0x106c	PF11_SEL	端口 PF11 功能配置寄存器	0x0000 0000	小节 8.12.5
0x1140	PFDIR	端口 PF 输入输出配置寄存器	0xffff ffff	小节 8.6.1
0x1144	PFIN	端口 PF 输入值寄存器	0x—- —-	小节 8.6.2
0x1148	PFOUT	端口 PF 输出值配置寄存器	0x—- —-	小节 8.6.3
0x114c	PFADS	端口 PF 数模配置寄存器	0x0000 0000	小节 8.6.4
0x1150	PFBSET	端口 PF 位置位寄存器	0x0000 0000	小节 8.6.5

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Offset	Acronym	Register Name	Reset	Section
0x1154	PFBCLR	端口 PF 位清零寄存器	0x0000 0000	小节 8.6.6
0x1158	PFBSETCLR	端口 PF 位置位清零寄存器	0x0000 0000	小节 8.6.7
0x115c	PFDR	端口 PF 驱动能力配置寄存器	0x0000 0000	小节 8.6.9
0x1160	PFPU	端口 PF 上拉使能配置寄存器	0x0000 0000	小节 8.6.8
0x1164	PFPD	端口 PF 下拉使能配置寄存器	0x0000 0000	小节 8.6.10
0x116c	PFOD	端口 PF 开漏输出配置寄存器	0x0000 0000	小节 8.6.11
0x1170	PFHIE	端口 PF 高电平中断使能配置寄存器	0x0000 0000	小节 8.6.12
0x1174	PFLIE	端口 PF 低电平中断使能配置寄存器	0x0000 0000	小节 8.6.13
0x1178	PFRIE	端口 PF 上升沿中断使能配置寄存器	0x0000 0000	小节 8.6.14
0x117c	PFFIE	端口 PF 下降沿中断使能配置寄存器	0x0000 0000	小节 8.6.15
0x1240	PF_STAT	端口 PF 中断状态寄存器	0x0000 0000	小节 8.6.16
0x1250	PF_ICLR	端口 PF 中断清除寄存器	0xffff ffff	小节 8.6.17

8.6 端口通用寄存器

8.6.1 端口 Px 输入输出配置寄存器 (PxDIR) (x = A,B,C,D,F)

起始地址：0x40020C00

地址偏移：0x100(PA),0x140(PB),0x180(PC),0x1C0(PD),0x1140(PF)

复位值：0xffff ffff

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	PxDIR	rw	0xFF	端口 Px 输入输出配置寄存器 (对应 Px15-Px00)。 1：配置成输入　　0：配置成输出 注：每个 bit 对应一个端口，例如：PxDIR[15] 对应端口 Px15

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

8.6.2 端口 Px 输入值寄存器 (PxIN) (x = A,B,C,D,F)

起始地址：0x40020C00

地址偏移：0x104(PA),0x144(PB),0x184(PC),0x1C4(PD),0x1144(PF)

复位值：NA

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	PxIN	rw	NA	端口 Px 输入值寄存器 (对应 Px15-Px00)。 1：输入为高电平 0：输入为低电平

8.6.3 端口 Px 输出值配置寄存器 (PxOUT) (x = A,B,C,D,F)

起始地址：0x40020C00

地址偏移：0x108(PA),0x148(PB),0x188(PC),0x1C8(PD),0x1148(PF)

复位值：NA

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	PxOUT	rw	NA	端口 Px 输出值配置寄存器 (对应 Px15-Px00)。 1：输出高电平。如果配置成开漏输出，则需外部上拉电阻来拉高。 0：输出低电平。

8.6.4 端口 Px 数模配置寄存器 (PxADS) (x = A,B,C,D,F)

起始地址：0x40020C00

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

地址偏移：0x10C(PA),0x14C(PB),0x18C(PC),0x1CC(PD),0x114C(PF)

复位值：0x0000 0000

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	PxADS	rw	0x00	端口 Px 数模配置寄存器 (对应 Px15-Px00)。 1：配置为模拟端口 0：配置为数字端口

8.6.5 端口 Px 位置位寄存器 (PxBSET) (x = A,B,C,D,F)

起始地址：0x40020C00

地址偏移：0x110(PA),0x150(PB),0x190(PC),0x1D0(PD),0x1150(PF)

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
PxBSET[15:0]															
W1															

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	PxBSET	rw	0x00	端口 Px 位置位寄存器 (对应 Px15-Px00)。 1：置位 0：保持

8.6.6 端口 Px 位清零寄存器 (PxBCLR) (x = A,B,C,D,F)

起始地址：0x40020C00

地址偏移：0x114(PA),0x154(PB),0x194(PC),0x1D4(PD),0x1154(PF)

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
PxBCLR[15:0]															
W1															

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	PxBCLR	rw	0x00	端口 Px 位清零寄存器 (对应 Px15-Px00)。 1：清零 0：保持

8.6.7 端口 Px 位置位清零寄存器 (PxBSETCLR) (x = A,B,C,D,F)

起始地址：0x40020C00

地址偏移：0x118(PA),0x158(PB),0x198(PC),0x1D8(PD),0x1158(PF)

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PxBSET[15:0]															
W1															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
PxBCLR[15:0]															
W1															

Bit	Field	Type	Reset	Description
31:16	PxBSET	rw	0x00	端口 Px 位置位寄存器 1：置位 0：保持
15:0	PxBCLR	rw	0x00	端口 Px 位清零寄存器 1：清零 0：保持

注意：

-PxBSET 和 PxBCLR 相同位同时置 1 时，PxBCLR 具有高优先级。即该端口被清零。

8.6.8 端口 Px 上拉使能配置寄存器 (PxPU) (x = A,B,C,D,F)

起始地址：0x40020C00

地址偏移：0x120(PA),0x160(PB),0x1A0(PC),0x1E0(PD),0x1160(PF)

复位值：0x0000 0000

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	PxPU	rw	0x00	端口 Px 上拉使能配置寄存器 1：使能 0：禁止

8.6.9 端口 Px 驱动能力配置寄存器 (PxDR) (x = A,B,C,D,F)

起始地址：0x40020C00

地址偏移：0x11C(PA),0x15C(PB),0x19C(PC),0x1DC(PD),0x115C(PF)

复位值：0x0000 0000

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	PxDR	rw	0x00	端口 Px 驱动能力配置寄存器。 1：低驱动能力 0：高驱动能力

8.6.10 端口 Px 下拉使能配置寄存器 (PxPD) (x = A,B,C,D,F)

起始地址：0x40020C00

地址偏移：0x124(PA),0x164(PB),0x1A4(PC),0x1E4(PD),0x1164(PF)

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	PxPD	rw	0x00	端口 Px 下拉使能配置寄存器。 1：使能 0：禁止

8.6.11 端口 Px 开漏输出配置寄存器 (PxOD) (x = A,B,C,D,F)

起始地址：0x40020C00

地址偏移：0x12C(PA),0x16C(PB),0x1AC(PC),0x1EC(PD),0x116C(PF)

复位值：0x0000 0000

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	PxOD	rw	0x00	端口 Px 开漏输出配置寄存器。 1：使能 0：禁止

8.6.12 端口 Px 高电平中断使能配置寄存器 (PxHIE) (x = A,B,C,D,F)

起始地址：0x40020C00

地址偏移：0x130(PA),0x170(PB),0x1B0(PC),0x1F0(PD),0x1170(PF)

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	PxHIE	rw	0x00	端口 Px 高电平中断使能配置寄存器。 1：使能 0：禁止

8.6.13 端口 Px 低电平中断使能配置寄存器 (PxLIE) (x = A,B,C,D,F)

起始地址：0x40020C00

地址偏移：0x134(PA),0x174(PB),0x1B4(PC),0x1F4(PD),0x1174(PF)

复位值：0x0000 0000

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	PxLIE	rw	0x00	端口 Px 低电平中断使能配置寄存器。 1：使能 0：禁止

8.6.14 端口 Px 上升沿中断使能配置寄存器 (PxRIE) (x = A,B,C,D,F)

起始地址：0x40020C00

地址偏移：0x138(PA),0x178(PB),0x1B8(PC),0x1F8(PD),0x1178(PF)

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	PxRIE	rw	0x00	端口 Px 上升沿中断使能配置寄存器。 1：使能 0：禁止

8.6.15 端口 Px 下降沿中断使能配置寄存器 (PxFIE) (x = A,B,C,D,F)

起始地址：0x40020C00

地址偏移：0x13C(PA),0x17C(PB),0x1BC(PC),0x1FC(PD),0x117C(PF)

复位值：0x0000 0000

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	PxFIE	rw	0x00	端口 Px 下降沿中断使能配置寄存器。 1：使能 0：禁止

8.6.16 端口 Px 中断状态寄存器 (Px_STAT) (x = A,B,C,D,F)

起始地址：0x40020C00

地址偏移：0x200(PA),0x240(PB),0x280(PC),0x2C0(PD),0x1240(PF)

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	Px_STAT	rw	0x00	端口 Px 中断状态寄存器。 1：中断触发 0：无中断触发

8.6.17 端口 Px 中断清除寄存器 (Px_ICLR) (x = A,B,C,D,F)

起始地址：0x40020C00

地址偏移：0x210(PA),0x250(PB),0x290(PC),0x2D0(PD),0x1250(PF)

复位值：0xffff ffff

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

Bit	Field	Type	Reset	Description
31:16	Reserved		0xffff	保留
15:0	Px_ICLR	rw	0xffff	端口 Px 中断清除寄存器。 1：保留中断标志位 0：清除中断标志位

8.7 端口辅助寄存器

8.7.1 端口辅助功能配置寄存器 0(GPIO_CTRL0)

起始地址：0x40020C00

地址偏移：0x300

复位值：0x0000 0001

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="15" rowspan="2">Reserved</td><td>IESEL</td></tr>
<tr><td>rw</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:1	Reserved		0x00	保留
0	IESEL	rw	0x01	端口中断模式选择寄存器。 1：DEEPSLEEP 模式 0：ACTIVE/SLEEP 模式

注意：

- 当系统处于 ACTIVE/SLEEP 模式下时，系统时钟不会被关闭，可以将 IESEL 设置为 0，此时触发端口中断的外部信号源将经过系统时钟同步之后产生中断信号，可以滤除外部信号源的毛刺。当系统即将进入 DEEPSLEEP 模式下时，系统时钟将被关闭，可以将 IESEL 设置为 1，此时触发端口中断的外部信号源直接产生中断信号，不能滤除外部信号源的毛刺。

8.7.2 端口辅助功能配置寄存器 1(GPIO_CTRL1)

起始地址：0x40020C00

地址偏移：0x304

复位值：0x0000 0000

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td rowspan="2">Res</td><td>ir_pol</td><td>hclk_en</td><td>pclk_en</td><td colspan="2">hclk_sel</td><td colspan="2">pclk_sel</td><td colspan="4" rowspan="2">Res</td><td colspan="4">ext_clk_sel</td></tr>
<tr><td>rw</td><td>rw</td><td>rw</td><td colspan="2">rw</td><td colspan="2">rw</td><td colspan="4">rw</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:15	Reserved		0x00	保留。
14	ir_pol	rw	0x00	IR 输出极性选择。 0 –正向输出 1 –反向输出

Bit	Field	Type	Reset	Description
13	hclk_en	rw	0x00	hclk 输出门控。 0 -门控 1 -输出
12	pclk_en	rw	0x00	pclk 输出门控。 0 -门控 1 -输出
11:10	hclk_sel	rw	0x00	hclk 输出分频选择。 00: hclk 01: hclk/2 10: hclk/4 11: hclk/8
9:8	pclk_sel	rw	0x00	pclk 输出分频选择。 00: pclk 01: pclk/2 10: pclk/4 11: pclk/8
7:4	Reserved		0x00	保留。
3:0	ext_clk_sel	rw	0x00	外部时钟信号来源选择。 0000: 高电平 0001: PA3 0010: PA4 0011: 保留 0100: PA8 0101: PA9 0110: PA12 0111: 保留 1000: PB1 1001: PB2 1010: PB5 1011: PB6 1100: PB9 1101: PB10 1110: PB12 1111: PB14

8.7.3 端口辅助功能配置寄存器 2(GPIO_CTRL2)

起始地址：0x40020C00

地址偏移：0x308

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ahb_sel	Res							tclk_div		tclk_sel		ssn1_sel			
rw								rw		rw		rw			

Bit	Field	Type	Reset	Description
31:16	Reserved		0x00	保留。
15	ahb_sel	rw	0x00	AHB 总线访问模式。 0 –使用 FASTIO 总线访问 PxIN,PxOUT 寄存器 1 – 使用 AHB 总线访问 PxIN,PxOUT 寄存器
14:8	Reserved		0x00	保留
7:6	tclk_div	rw	0x00	tclk 输出分频选择。 00：tclk 01：tclk/2 10：tclk/4 11：tclk/8
5:4	tclk_sel	rw	0x00	tclk 输出信号源选择。 00：- 01：RCH 10：XTH 11：PLL
3:0	ssn1_sel	rw	0x00	SPI1 SSN 信号来源选择。 0000：高电平 0001：PA3 0010：PA4 0011：保留 0100：PA8 0101：PA9 0110：PA12 0111：保留 1000：PB1 1001：PB2 1010：PB5 1011：PB6 1100：PB9 1101：PB10 1110：PB12 1111：PB14

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

8.7.4 端口辅助功能定时器门控选择 (GPIO_TIMGS)

起始地址：0x40020C00

地址偏移：0x30C

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	LPTIM_G			TIM3_G			TIM2_G			TIM1_G			TIM0_G		
	R/W			R/W			R/W			R/W			R/W		

Bit	Field	Type	Reset	Description
31:15	Reserved		0x00	保留。
14:12	LPTIM0_G	rw	0x00	LPTimer0 定时器 GATE 输入选择，选择见表18。
11:9	TIM3_G	rw	0x00	Timer3 定时器 GATE 输入选择，选择见表 18。
8:6	TIM2_G	rw	0x00	Timer2 定时器 GATE 输入选择，选择见表 18。
5:3	TIM1_G	rw	0x00	Timer1 定时器 GATE 输入选择，选择见表18。
2:0	TIM0_G	rw	0x00	Timer0 定时器 GATE 输入选择，选择见表18。

表 18: 定时器 GATE 输入选择

	TIM0_g	TIM1_g	TIM2_g	TIM3_g	LPTIM_g
000	PX_SEL	PX_SEL	PX_SEL	PX_SEL	PX_SEL
001	UART0_RXD	LPUART0_RXD	UART0_RXD	UART0_RXD	LPUART0_RXD
010	UART1_RXD	LPUART1_RXD	UART1_RXD	UART1_RXD	LPUART1_RXD
011	VC0_OUT	VC0_OUT	VC0_OUT	UART2	VC0_OUT
100	VC1_OUT	VC1_OUT	VC1_OUT	UART3	VC1_OUT
101	PA3	PA8	PA10	VC0_OUT	PB3
110	PB8	PB3	PB4	-	PB5
111	PB15	PB13	PB11	PA11	PC0

8.7.5 端口辅助功能定时器 ETR 选择 (GPIO_TIMES)

起始地址：0x40020C00

地址偏移：0x310

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	LPTIM_E			TIM3_E			TIM2_E			TIM1_E			TIM0_E		
	R/W			R/W			R/W			R/W			R/W		

Bit	Field	Type	Reset	Description
31:15	Reserved		0x00	保留。
14:12	LPTIM0_E	rw	0x00	LPTimer0 定时器 ETR 输入选择，选择见表19。
11:9	TIM3_E	rw	0x00	Timer3 定时器 ETR 输入选择，选择见表19。
8:6	TIM2_E	rw	0x00	Timer2 定时器 ETR 输入选择，选择见表19。
5:3	TIM1_E	rw	0x00	Timer1 定时器 ETR 输入选择，选择见表19。
2:0	TIM0_E	rw	0x00	Timer0 定时器 ETR 输入选择，选择见表19。

表 19: 定时器 ETR 输入选择

	TIM0_e	TIM1_e	TIM2_e	TIM3_e	LPTIM_e
000	PX_SEL	PX_SEL	PX_SEL	PX_SEL	PX_SEL
001	LPUART0_RXD	UART0_RXD	LPUART0_RXD	UART0_RXD	PCNT_S0
010	LPUART1_RXD	UART1_RXD	LPUART1_RXD	UART1_RXD	LVD_OUT
011	VC0_OUT	VC1_OUT	VC0_OUT	VC1_OUT	VC0_OUT
100	LVD_OUT	LVD_OUT	PCNT_S1	PCNT_S0	VC1_OUT
101	PA0	PA1	PA4	PA0	PB4
110	-	PC9	PC4	PA12	PB6
111	-	PD2	PC8	PA13	PC3

8.7.6 端口辅助功能定时器捕获输入选择 (GPIO_TIMCPS)

起始地址：0x40020C00

地址偏移：0x314

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TIM3_CB			TIM3_CA			TIM2_CA			TIM1_CA			TIM0_CA		
	R/W			R/W			R/W			R/W			R/W		

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31:15	Reserved		0x00	保留
14:12	TIM3_CB	rw	0x00	Timer3 定时器 CH0B 输入选择，选择见表20。
11:9	TIM3_CA	rw	0x00	Timer3 定时器 CH0A 输入选择，选择见表20。
8:6	TIM2_CA	rw	0x00	Timer2 定时器 CHA 输入选择，选择见表 20。
5:3	TIM1_CA	rw	0x00	Timer1 定时器 CHA 输入选择，选择见表 20。
2:0	TIM0_CA	rw	0x00	Timer0 定时器 CHA 输入选择，选择见表 20。

表 20: 定时器捕获输入选择

	TIM0_CHA	TIM1_CHA	TIM2_CHA	TIM3_CH0A	TIM3_CH0B
000	PX_SEL	PX_SEL	PX_SEL	PX_SEL	PX_SEL
001	UART0_RXD	UART1_RXD	LPUART0_RXD	LPUART1_RXD	UART0_RXD
010	PA0	PA0	VC0_OUT	LPUART0_RXD	UART1_RXD
011	PA2	PA2	PA2	PCNT_S0	PCNT_S1
100	PA5	-	-	VC0_OUT	VC1_OUT
101	-	PB8	PB8	PA8	-
110	PB6	PB10	PB9	PB3	PB4
111	PB14	PB13	PC6	PB6	PB13

8.7.7 端口辅助功能 PCA 捕获选择 (GPIO_PCAS)

起始地址：0x40020C00

地址偏移：0x318

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.				LPTIM1_E			LPTIM1_G			PCA_CH0			PCA_ECI		
				R/W			R/W			R/W			R/W		

Bit	Field	Type	Reset	Description
31:12	Reserved		0x00	保留
11:9	LPTIM1_E	rw	0x00	LPTimer1 定时器 ETR 输入选择，选择见表21 。
8:6	LPTIM1_G	rw	0x00	LPTimer1 定时器 GATE 输入选择，选择见表21。
5:3	PCA_ECI	rw	0x00	PCA ECI 时钟输入选择，选择见表 21 。
2:0	PCA_CH0	rw	0x00	PCA CH0 捕获口输入选择，选择见表 21 。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

表 21: PCA 捕获选择

	LPTIM_E	LPTIM_G	pca_eci	pca_ch0
000	PX_SEL	PX_SEL	PX_SEL	PX_SEL
001	UART1	UART0	PCNT_S1	PCNT_S0
010	LPUART1_RXD	LPUART0_RXD	LVD_OUT	PCNT_S1
011	VC0_OUT	VC1_OUT	VC0_OUT	LVD_OUT
100	VC2_OUT	VC2_OUT	VC1_OUT	VC1_OUT
101	LPTIM0_TOG	PD5	-	-
110	-	-	PB2	PB4
111	PC7	PC6	PD2	PC6

8.7.8 端口辅助功能 PCNT 脉冲输入选择 (GPIO_PCNT)

起始地址：0x40020C00

地址偏移：0x31C

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.												PCNT_S1		PCNT_S0	
												R/W		R/W	

Bit	Field	Type	Reset	Description
31:4	Reserved		0x00	保留
3:2	PCNT_S1	rw	0x00	PCNT_S1 输入选择，选择见表 22。
1:0	PCNT_S0	rw	0x00	PCNT_S0 输入选择，选择见表 22。

表 22: PCNT 脉冲输入选择

	PCNT_S1	PCNT_S0
00	PX_SEL	PX_SEL
01	VC0_OUT	VC0_OUT
10	VC1_OUT	VC1_OUT
11	VC2_OUT	VC2_OUT

8.8 端口 PA 功能选择寄存器

8.8.1 端口 PA00 功能配置寄存器 (PA0_SEL)

起始地址：0x40020C00

地址偏移：0x00

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PA0_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留
2:0	PA0_sel	rw	0x00	端口 PA00 功能选择。 000:GPIO　PA00 001: UART1_CTS　UART1 模块 CTS 信号 010 : LPUART1_TXD　LPUART1 模块 TXD 信号 011 : TIM0_ETR　Timer0 模块外部时钟输入信号 100 : VC0_OUT　VC0 模块输出/反向输出信号 101 : TIM1_CHA　Timer1 模块通道 A 信号 110 : TIM3_ETR　Timer3 模块外部时钟输入信号 111 : TIM0_CHA　Timer0 模块通道 A 信号

8.8.2 端口 PA01 功能配置寄存器 (PA1_SEL)

起始地址：0x40020C00

地址偏移：0x04

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PA1_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
2:0	PA1_sel	rw	0x00	端口 PA1 功能选择。 000 : GPIO　PA1 001 : UART1_RTS　UART1 模块 RTS 信号 010 : LPUART1_RXD　LPUART1 模块 RXD 信号 011 : TIM0_CHB　Timer0 模块通道 B 信号 100 : TIM1_ETR　Timer1 模块外部时钟输入信号 101 : TIM1_CHB　Timer1 模块通道 B 信号 110 : HCLK_OUT　AHB 总线时钟输出信号 111 : SPI1_MOSI　SPI1 主机输出从机输入

8.8.3 端口 PA02 功能配置寄存器（PA2_SEL）

起始地址：0x40020C00

地址偏移：0x08

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved													PA2_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PA2_sel	rw	0x00	端口 PA2 功能选择。 000:GPIO　PA2 001:UART1_TXD　UART1 模块 TXD 信号 010:TIM0_CHA　Timer0 模块通道 A 信号 011:VC1_OUT　VC1 模块输出/反向输出信号 100:TIM1_CHA　Timer1 模块通道 A 信号 101:TIM2_CHA　Timer2 模块通道 A 信号 110:PCLK_OUT　APB 总线时钟输出信号 111:SPI1_MISO　SPI1 主机输入从机输出

8.8.4 端口 PA03 功能配置寄存器（PA3_SEL）

起始地址：0x40020C00

地址偏移：0x0c

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PA3_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PA3_sel	rw	0x00	端口 PA3 功能选择。 000:GPIO　PA3 001:UART1_RXD　UART1 模块 RXD 信号 010:TIM0_GATE　Timer0 模块门控信号 011:TIM1_CHB　Timer1 模块通道 B 信号 100:TIM2_CHB　Timer2 模块通道 B 信号 101:SPI1_CS　SPI1 模块主机模式片选信号 110:TIM3_CH1A　Timer3 模块通道 1A 信号 111:TIM5_CHA　Timer5 模块通道 1A 信号

8.8.5 端口 PA04 功能配置寄存器（PA4_SEL）

起始地址：0x40020C00

地址偏移：0x10

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PA4_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。

Bit	Field	Type	Reset	Description
2:0	PA4_sel	rw	0x00	端口 PA4 功能选择。 000: GPIO PA4 001: 保留 010: UART1_TXD UART1 模块 TXD 信号 011: PCA_CH4 PCA 模块通道 4 捕获/比较信号 100: TIM2_ETR Timer2 模块外部时钟输入信号 101: TIM5_CHA Timer6 模块通道 1A 信号 110: LVD_OUT LVD 模块输出信号 111: TIM3_CH2B Timer3 模块通道 2B 信号

8.8.6 端口 PA08 功能配置寄存器（PA8_SEL）

起始地址：0x40020C00

地址偏移：0x20

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved													PA8_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PA8_sel	rw	0x00	PA8_sel: 端口 PA8 功能选择。 000:GPIO PA8 001:UART0_TXD UART0 模块 TXD 信号 010:TIM3_CH0A Timer3 模块通道 0A 信号 011: 保留 保留 100: 保留 保留 101:TIM1_GATE Timer1 模块门控信号 110:TIM4_CHA Timer6 模块通道 0A 信号 111:TIM3_BK Timer3 模块刹车信号

8.8.7 端口 PA09 功能配置寄存器（PA9_SEL）

起始地址：0x40020C00

地址偏移：0x24

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PA9_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PA9_sel	rw	0x00	端口 PA9 功能选择。 000:GPIO　PA9 001:UART0_TXD　UART0 模块 TXD 信号 010:TIM3_CH1A　Timer3 模块通道 1A 信号 011:TIM0_BK　Timer0 模块刹车信号 100:I2C0_SCL　I2C0 模块时钟信号 101: 保留　保留 110:HCLK_OUT　AHB 总线时钟输出信号 111:TIM5_CHA　Timer6 模块通道 1A 信号

8.8.8 端口 PA10 功能配置寄存器 (PA10_SEL)

起始地址：0x40020C00

地址偏移：0x28

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PA10_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
2:0	PA10_sel	rw	0x00	PA10_sel: 端口 PA10 功能选择。 000:GPIO PA10 001:UART0_RXD UART0 模块 RXD 信号 010:TIM3_CH2A Timer3 模块通道 2A 信号 011:TIM2_BK Timer2 模块刹车信号 100:I2C0_SDA I2C0 模块数据信号 101:TIM2_GATE Timer2 模块门控信号 110:PCLK_OUT APB 总线时钟输出信号 111:TIM6_CHA Timer6 模块通道 2A 信号

8.8.9 端口 PA11 功能配置寄存器（PA11_SEL）

起始地址：0x40020C00

地址偏移：0x2c

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved													PA11_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved	rw	0x00	保留。
2:0	PA11_sel	rw	0x00	PA11_sel: 端口 PA11 功能选择。 000:GPIO PA11 001:UART0_CTS UART0 模块 CTS 信号 010:TIM3_GATE Timer3 模块门控信号 011:I2C1_SCL I2C1 模块时钟信号 100:Reserved Reserved 101:VC0_OUT VC0 模块输出/反向输出信号 110:Reserved Reserved 111:TIM4_CHB Timer6 模块通道 0B 信号

8.8.10 端口 PA12 功能配置寄存器（PA12_SEL）

起始地址：0x40020C00

地址偏移：0x30

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PA12_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PA12_sel	rw	0x00	PA12_sel: 端口 PA12 功能选择。 000:GPIO PA12 001:UART0_RTS UART0 模块 RTS 信号 010:TIM3_ETR Timer3 模块外部时钟输入信号 011:I2C1_SDA I2C1 模块数据信号 100:Reserved Reserved 101:VC1_OUT VC1 模块输出/反向输出信号 110:Reserved Reserved 111:PCNT_S0 PCNT 模块输入信号 0

8.8.11 端口 PA13 功能配置寄存器（PA13_SEL）

起始地址：0x40020C00

地址偏移：0x34

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PA13_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
2:0	PA13_sel	rw	0x00	PA13_sel: 端口 PA13 功能选择。 000:GPIO PA13 001:IR_OUT 红外输出信号 010:UART0_RXD UART0 模块 RXD 信号 011:LVD_OUT LVD 模块输出信号 100:TIM3_ETR Timer3 模块外部时钟输入信号 101:RTC_1HZ RTC 模块 1Hz 输出信号 110:PCNT_S1 PCNT 模块输入信号 1 111:VC2_OUT VC2 模块输出/反向输出信号

8.8.12 端口 PA14 功能配置寄存器（PA14_SEL）

起始地址：0x40020C00

地址偏移：0x38

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PA14_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PA14_sel	rw	0x00	PA14_sel: 端口 PA14 功能选择。 000:GPIO PA14 001:UART1_TXD UART1 模块 TXD 信号 010:UART0_TXD UART0 模块 TXD 信号 011:TIM3_CH2A Timer3 模块通道 2A 信号 100:LVD_OUT LVD 模块输出信号 101:RCH_OUT 内部 24M RC 时钟输出信号 110:RCL_OUT 内部 38K RC 时钟输出信号 111:PLL_OUT 内部 PLL 时钟输出信号

8.9 端口 PB 功能选择寄存器

8.9.1 端口 PB0 功能配置寄存器（PB0_SEL）

起始地址：0x40020C00

地址偏移：0x40

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PB0_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PB0_sel	rw	0x00	端口 PB0 功能选择。 000:GPIO PB0 001:PCA_CH2 PCA 模块通道 2 捕获/比较信号 010:TIM3_CH1B Timer3 模块通道 1B 信号 011:LPUART0_TXD LPUART0 模块 TXD 信号 100:TIM5_CHB Timer6 模块通道 1B 信号 101:RCH_OUT 内部 24M RC 时钟输出信号 110:RCL_OUT 内部 38K RC 时钟输出信号 111:PLL_OUT 内部 PLL 时钟输出信号

8.9.2 端口 PB1 功能配置寄存器 (PB1_SEL)

起始地址：0x40020C00

地址偏移：0x44

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PB1_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。

Bit	Field	Type	Reset	Description
2:0	PB1_sel	rw	0x00	PB1_sel: 端口 PB1 功能选择。 000:GPIO PB1 001:PCA_CH3 PCA 模块通道 3 捕获/比较信号 010:PCLK_OUT APB 总线时钟输出信号 011:TIM3_CH2B Timer3 模块通道 2B 信号 100:TIM6_CHB Timer6 模块通道 2B 信号 101:LPUART0_RTS LPUART0 模块 RTS 信号 110:VC2_OUT VC2 模块输出/反向输出信号 111:TCLK_OUT 内部时钟输出信号

8.9.3 端口 PB2 功能配置寄存器（PB2_SEL）

起始地址：0x40020C00

地址偏移：0x48

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PB2_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PB2_sel	rw	0x00	PB2_sel: 端口 PB2 功能选择。 000:GPIO PB2 001:LPTIM_TOG 低功耗 timer 模块翻转信号 010:PCA_ECI PCA 模块外部时钟输入信号 011:LPUART1_TXD LPUART1 模块 TXD 信号 100:TIM4_CHA Timer6 模块通道 0A 信号 101:TIM1_BK Timer1 模块刹车信号 110:TIM0_BK Timer0 模块刹车信号 111:TIM2_BK Timer2 模块刹车信号

8.9.4 端口 PB3 功能配置寄存器（PB3_SEL）

起始地址：0x40020C00

地址偏移：0x4c

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved													PB3_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved	rw	0x00	保留。
2:0	PB3_sel	rw	0x00	PB3_sel: 端口 PB3 功能选择。 000:GPIO PB3 001:Reserved Reserved 010:TIM0_CHB Timer0 模块通道 B 信号 011:TIM1_GATE Timer1 模块门控信号 100:TIM3_CH0A Timer3 模块通道 0A 信号 101:LPTIM_GATE 低功耗 timer 模块门控信号 110:XTL_OUT 外部 32K 晶振输出信号 111:XTH_OUT 外部 32M 晶振输出信号

8.9.5 端口 PB4 功能配置寄存器 (PB4_SEL)

起始地址：0x40020C00

地址偏移：0x50

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved													PB4_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
2:0	PB4_sel	rw	0x00	PB4_sel: 端口 PB4 功能选择。 000:GPIO　PB4 001:Reserved　Reserved 010:PCA_CH0　PCA 模块通道 0 捕获/比较信号 011:TIM2_BK　Timer2 模块刹车信号 100:UART0_CTS　UART0 模块 CTS 信号 101:TIM2_GATE　Timer2 模块门控信号 110:TIM3_CH0B　Timer3 模块通道 0B 信号 111:LPTIM_ETR　低功耗 timer 模块外部时钟输入

8.9.6 端口 PB5 功能配置寄存器（PB5_SEL）

起始地址：0x40020C00

地址偏移：0x54

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PB5_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PB05_sel	rw	0x00	PB5_sel: 端口 PB5 功能选择。 000:GPIO　PB5 001:Reserved　Reserved 010:Reserved　Reserved 011:TIM1_BK　Timer1 模块刹车信号 100:PCA_CH1　PCA 模块通道 1 捕获/比较信号 101:LPTIM_GATE　低功耗 timer 模块门控信号 110:PCNT_S0　PCNT 模块输入信号 0 111:UART0_RTS　UART0 模块 RTS 信号

8.9.7 端口 PB6 功能配置寄存器（PB6_SEL）

起始地址：0x40020C00

地址偏移：0x58

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved													PB6_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PB6_sel	rw	0x00	PB6_sel: 端口 PB6 功能选择。 000: GPIO　PB6 001: I2C0_SCL　I2C0 模块时钟信号 010: UART0_TXD　UART0 模块 TXD 信号 011: TIM1_CHB　Timer1 模块通道 B 信号 100: TIM0_CHA　Timer0 模块通道 A 信号 101: LPTIM_ETR　低功耗 timer 模块外部时钟输入 110: TIM3_CH0A　Timer3 模块通道 0A 信号 111: LPTIM_TOG　低功耗 timer 模块翻转信号

8.9.8 端口 PB7 功能配置寄存器 (PB7_SEL)

起始地址：0x40020C00

地址偏移：0x5c

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved													PB7_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved	rw	0x00	保留。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
2:0	PB7_sel	rw	0x00	PB7_sel: 端口 PB7 功能选择。 000:GPIO PB7 001:I2C0_SDA I2C0 模块数据信号 010:UART0_RXD UART0 模块 RXD 信号 011:TIM2_CHB Timer2 模块通道 B 信号 100:LPUART1_CTS LPUART1 模块 CTS 信号 101:TIM0_CHB Timer0 模块通道 B 信号 110:LPTIM_TOGN 低功耗 timer 模块翻转反向信号 111:PCNT_S1 PCNT 模块输入信号 1

8.9.9 端口 PB8 功能配置寄存器（PB8_SEL）

起始地址：0x40020C00

地址偏移：0x60

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PB8_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PB8_sel	rw	0x00	PB8_sel: 端口 PB8 功能选择。 000:GPIO PB8 001:I2C0_SCL I2C0 模块时钟信号 010:TIM1_CHA Timer1 模块通道 A 信号 011:Reserved Reserved 100:TIM2_CHA Timer2 模块通道 A 信号 101:TIM0_GATE Timer0 模块门控信号 110:TIM3_CH2A Timer3 模块通道 2A 信号 111:UART0_TXD UART0 模块 TXD 信号

8.9.10 端口 PB9 功能配置寄存器（PB9_SEL）

起始地址：0x40020C00

地址偏移：0x64

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PB9_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PB9_sel	rw	0x00	PB9_sel: 端口 PB9 功能选择。 000:GPIO PB9 001:I2C0_SDA I2C0 模块数据信号 010:IR_OUT 红外输出信号 011:SPI1_CS SPI1 模块主机模式片选信号 100:TIM2_CHA Timer2 模块通道 A 信号 101:Reserved Reserved 110:TIM2_CHB Timer2 模块通道 B 信号 111:UART0_RXD UART0 模块 RXD 信号

8.9.11 端口 PB10 功能配置寄存器（PB10_SEL）

起始地址：0x40020C00

地址偏移：0x68

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PB10_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved	rw	0x00	保留。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
2:0	PB10_sel	rw	0x00	PB10_sel: 端口 PB10 功能选择。 000:GPIO PB10 001:I2C1_SCL I2C1 模块时钟信号 010:SPI1_SCK SPI1 模块时钟信号 011:TIM1_CHA Timer1 模块通道 A 信号 100:LPUART0_TXD LPUART0 模块 TXD 信号 101:TIM3_CH1A Timer3 模块通道 1A 信号 110:LPUART1_RTS LPUART1 模块 RTS 信号 111:UART1_RTS UART1 模块 RTS 信号

8.9.12 端口 PB11 功能配置寄存器（PB11_SEL）

起始地址：0x40020C00

地址偏移：0x6c

复位值：0x0000 0000

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="13" rowspan="2">Reserved</td><td colspan="3">PB11_sel</td></tr>
<tr><td colspan="3">rw</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PB11_sel	rw	0x00	PB11_sel: 端口 PB11 功能选择。 000:GPIO PB11 001:I2C1_SDA I2C1 模块数据信号 010:TIM1_CHB Timer1 模块通道 B 信号 011:LPUART0_RXD LPUART0 模块 RXD 信号 100:TIM2_GATE Timer2 模块门控信号 101:TIM6_CHA Timer6 模块通道 2A 信号 110:LPUART1_CTS LPUART1 模块 CTS 信号 111:UART1_CTS UART1 模块 CTS 信号

8.9.13 端口 PB12 功能配置寄存器（PB12_SEL）

起始地址：0x40020C00

地址偏移：0x70

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PB12_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PB12_sel	rw	0x00	PB12_sel: 端口 PB12 功能选择。 000:GPIO PB12 001:SPI1_CS SPI1 模块主机模式片选信号 010:TIM3_BK Timer3 模块刹车信号 011:LPUART0_TXD LPUART0 模块 TXD 信号 100:TIM0_BK Timer0 模块刹车信号 101:Reserved Reserved 110:LPUART0_RTS LPUART0 模块 RTS 信号 111:TIM6_CHA Timer6 模块通道 2A 信号

8.9.14 端口 PB13 功能配置寄存器（PB13_SEL）

起始地址：0x40020C00

地址偏移：0x74

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PB13_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
2:0	PB13_sel	rw	0x00	PB13_sel: 端口 PB13 功能选择。 000:GPIO PB13 001:SPI1_SCK SPI1 模块时钟信号 010:I2C1_SCL I2C1 模块时钟信号 011:TIM3_CH0B Timer3 模块通道 0B 信号 100:LPUART0_CTS LPUART0 模块 CTS 信号 101:TIM1_CHA Timer1 模块通道 A 信号 110:TIM1_GATE Timer1 模块门控信号 111:TIM6_CHB Timer6 模块通道 2B 信号

8.9.15 端口 PB14 功能配置寄存器（PB14_SEL）

起始地址：0x40020C00

地址偏移：0x78

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved													PB14_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PB14_sel	rw	0x00	PB14_sel: 端口 PB14 功能选择。 000:GPIO PB14 001:SPI1_MISO SPI1 主机输入从机输出 010:I2C1_SDA I2C1 模块数据信号 011:TIM3_CH1B Timer3 模块通道 1B 信号 100:TIM0_CHA Timer0 模块通道 A 信号 101:RTC_1HZ RTC 模块 1Hz 输出信号 110:LPUART0_RTS LPUART0 模块 RTS 信号 111:TIM1_BK Timer1 模块刹车信号

8.9.16 端口 PB15 功能配置寄存器（PB15_SEL）

起始地址：0x40020C00

地址偏移：0x7c

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PB15_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PB15_sel	rw	0x00	PB15_sel: 端口 PB15 功能选择。 000:GPIO PB15 001:SPI1_MOSI SPI1 主机输出从机输入 010:TIM3_CH2B Timer3 模块通道 2B 信号 011:TIM0_CHB Timer0 模块通道 B 信号 100:TIM0_GATE Timer0 模块门控信号 101:Reserved Reserved 110:Reserved Reserved 111:LPUART1_RXD LPUART1 模块 RXD 信号

8.10 端口 PC 功能选择寄存器

8.10.1 端口 PC0 功能配置寄存器（PC0_SEL）

起始地址：0x40020C00

地址偏移：0x80

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PC0_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved	rw	0x00	保留。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
2:0	PC0_sel	rw	0x00	PC0_sel: 端口 PC00 功能选择。 000:GPIO PC0 001:LPTIM_GATE 低功耗 timer 模块门控信号 010:PCNT_S0 PCNT 模块输入信号 0 011:UART1_CTS UART1 模块 CTS 信号 100:UART2_RTS UART2 模块 RTS 信号 101:I2S0_MCK I2S0 模块 MCK 信号 110:Reserved Reserved 111:Reserved Reserved

8.10.2 端口 PC1 功能配置寄存器 (PC1_SEL)

起始地址：0x40020C00

地址偏移：0x84

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PC1_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PC1_sel	rw	0x00	PC1_sel: 端口 PC1 功能选择。 000: GPIO PC1 001: LPTIM_TOG 低功耗 timer 模块翻转信号 010: TIM5_CHB Timer6 模块通道 1B 信号 011: UART1_RTS UART1 模块 RTS 信号 100: PCNT_S0FO PCNT 模块输入信号 0 滤波输出 101: Reserved Reserved 110: UART2_CTS UART2 模块 CTS 信号 111: Reserved Reserved

8.10.3 端口 PC2 功能配置寄存器 (PC2_SEL)

起始地址：0x40020C00

地址偏移：0x88

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PC2_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved	rw	0x00	保留。
2:0	PC2_sel	rw	0x00	PC2_sel: 端口 PC2 功能选择。 000:GPIO PC2 001:SPI1_MISO SPI1 主机输入从机输出 010:LPTIM_TOGN 低功耗 timer 模块翻转反向信号 011:PCNT_S1 PCNT 模块输入信号 1 100:UART2_RXD UART2 模块 RXD 信号 101:Reserved Reserved 110:Reserved Reserved 111:Reserved Reserved

8.10.4 端口 PC3 功能配置寄存器（PC3_SEL）

起始地址：0x40020C00

地址偏移：0x8c

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PC3_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
2:0	PC3_sel	rw	0x00	PC3_sel: 端口 PC3 功能选择。 000:GPIO PC3 001:SPI1_MOSI SPI1 主机输出从机输入 010:LPTIM_ETR 低功耗 timer 模块外部时钟输入 011:LPTIM_TOGN 低功耗 timer 模块翻转反向信号 100:PCNT_S1FO PCNT 模块输入信号 1 滤波输出 101:UART2_TXD UART2 模块 TXD 信号 110:Reserved Reserved 111:Reserved Reserved

8.10.5 端口 PC4 功能配置寄存器 (PC4_SEL)

起始地址：0x40020C00

地址偏移：0x90

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PC4_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PC4_sel	rw	0x00	PC4_sel: 端口 PC4 功能选择。 000:GPIO PC4 001:LPUART0_TXD LPUART0 模块 TXD 信号 010:TIM2_ETR Timer2 模块外部时钟输入信号 011:IR_OUT 红外输出信号 100:VC2_OUT VC2 模块输出/反向输出信号 101:Reserved Reserved 110:Reserved Reserved 111:Reserved Reserved

8.10.6 端口 PC5 功能配置寄存器 (PC5_SEL)

起始地址：0x40020C00

地址偏移：0x94

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PC5_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PC5_sel	rw	0x00	PC5_sel: 端口 PC5 功能选择。 000: GPIO PC5 001: LPUART0_RXD LPUART0 模块 RXD 信号 010: TIM6_CHB Timer6 模块通道 2B 信号 011: PCA_CH4 PCA 模块通道 4 捕获/比较信号 100: Reserved Reserved 101: Reserved Reserved 110: Reserved Reserved 111: Reserved Reserved

8.10.7 端口 PC6 功能配置寄存器（PC6_SEL）

起始地址：0x40020C00

地址偏移：0x98

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PC6_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved	rw	0x00	保留。

Bit	Field	Type	Reset	Description
2:0	PC6_sel	rw	0x00	PC6_sel: 端口 PC6 功能选择。 000: GPIO PC6 001: PCA_CH0 PCA 模块通道 0 捕获/比较信号 010: TIM4_CHA Timer6 模块通道 0A 信号 011: TIM2_CHA Timer2 模块通道 A 信号 100: LPTIM1_GATE 低功耗 timer1 模块门控信号 101: Reserved Reserved 110: UART3_RXD UART3 模块 RXD 信号 111: Reserved Reserved

8.10.8 端口 PC7 功能配置寄存器（PC7_SEL）

起始地址：0x40020C00

地址偏移：0x9c

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PC7_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PC7_sel	rw	0x00	PC7_sel: 端口 PC7 功能选择。 000: GPIO PC7 001: PCA_CH1 PCA 模块通道 1 捕获/比较信号 010: TIM5_CHA Timer6 模块通道 1A 信号 011: TIM2_CHB Timer2 模块通道 B 信号 100: LPTIM1_ETR 低功耗 timer1 模块外部时钟输入 101: Reserved Reserved 110: UART3_TXD UART3 模块 TXD 信号 111: Reserved Reserved

8.10.9 端口 PC8 功能配置寄存器（PC8_SEL）

起始地址：0x40020C00

地址偏移：0xa0

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved													PC8_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved	rw	0x00	保留。
2:0	PC8_sel	rw	0x00	PC8_sel: 端口 PC8 功能选择。 000: GPIO　PC8 001: PCA_CH2　PCA 模块通道 2 捕获/比较信号 010: TIM6_CHA　Timer6 模块通道 2A 信号 011: TIM2_ETR　Timer2 模块外部时钟输入信号 100: LPTIM1_TOG　低功耗 timer1 模块翻转信号 101: Reserved　Reserved 110: UART3_CTS　UART3 模块 CTS 信号 111: Reserved　Reserved

8.10.10　端口 PC9 功能配置寄存器（PC9_SEL）

起始地址：0x40020C00

地址偏移：0xa4

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved													PC9_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
2:0	PC9_sel	rw	0x00	PC9_sel: 端口 PC9 功能选择。 000: GPIO PC9 001: PCA_CH3 PCA 模块通道 3 捕获/比较信号 010: TIM4_CHB Timer6 模块通道 0B 信号 011: TIM1_ETR Timer1 模块外部时钟输入信号 100: LPTIM1_TOGN 低功耗 timer1 模块翻转反向信号 101: Reserved Reserved 110: UART3_RTS UART3 模块 RTS 信号 111: Reserved Reserved

8.10.11 端口 PC10 功能配置寄存器（PC10_SEL）

起始地址：0x40020C00

地址偏移：0xa8

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PC10_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PC10_sel	rw	0x00	PC10_sel: 端口 PC10 功能选择。 000:GPIO PC10 001:LPUART1_TXD LPUART1 模块 TXD 信号 010:LPUART0_TXD LPUART0 模块 TXD 信号 011:PCA_CH2 PCA 模块通道 2 捕获/比较信号 100:Reserved Reserved 101:Reserved Reserved 110:Reserved Reserved 111:Reserved Reserved

8.10.12 端口 PC11 功能配置寄存器（PC11_SEL）

起始地址：0x40020C00

地址偏移：0xac

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PC11_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PC11_sel	rw	0x00	PC11_sel: 端口 PC11 功能选择。 000:GPIO PC11 001:LPUART1_RXD LPUART1 模块 RXD 信号 010:LPUART0_RXD LPUART0 模块 RXD 信号 011:PCA_CH3 PCA 模块通道 3 捕获/比较信号 100:PCNT_S0FO PCNT 模块输入信号 0 滤波输出 101:Reserved Reserved 110:Reserved Reserved 111:Reserved Reserved

8.10.13 端口 PC12 功能配置寄存器（PC12_SEL）

起始地址：0x40020C00

地址偏移：0xb0

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PC12_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved	rw	0x00	保留。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
2:0	PC12_sel	rw	0x00	PC12_sel: 端口 PC12 功能选择。 000:GPIO PC12 001:LPUART0_TXD LPUART0 模块 RXD 信号 010:LPUART1_TXD LPUART1 模块 TXD 信号 011:PCA_CH4 PCA 模块通道 4 捕获/比较信号 100:PCNT_S1FO PCNT 模块输入信号 1 滤波输出 101:Reserved Reserved 110:Reserved Reserved 111:Reserved Reserved

8.10.14 端口 PC13 功能配置寄存器（PC13_SEL）

起始地址：0x40020C00

地址偏移：0xb4

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PC13_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PC13_sel	rw	0x00	PC13_sel: 端口 PC13 功能选择。 000: GPIO PC13 001: Reserved Reserved 010: RTC_1HZ RTC 模块 1Hz 输出信号 011: TIM3_CH1B Timer3 模块通道 1B 信号 100: Reserved Reserved 101: Reserved Reserved 110: Reserved Reserved 111: Reserved Reserved

8.10.15 端口 PC14 功能配置寄存器（PC14_SEL）

起始地址：0x40020C00

地址偏移：0xb8

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved													PC14_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PC14_sel	rw	0x00	PC14_sel: 端口 PC14 功能选择。 000: GPIO PC14 001: Reserved Reserved 010: Reserved Reserved 011: Reserved Reserved 100: Reserved Reserved 101: Reserved Reserved 110: Reserved Reserved 111: Reserved Reserved

8.10.16 端口 PC15 功能配置寄存器（PC15_SEL）

起始地址：0x40020C00

地址偏移：0xbc

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved													PC15_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
2:0	PC15_sel	rw	0x00	PC15_sel: 端口 PC15 功能选择。 000:GPIO PC15 001:Reserved Reserved 010:Reserved Reserved 011:Reserved Reserved 100:Reserved Reserved 101:Reserved Reserved 110:Reserved Reserved 111:Reserved Reserved

8.11 端口 PD 功能选择寄存器

8.11.1 端口 PD02 功能配置寄存器（PD2_SEL）

起始地址：0x40020C00

地址偏移：0xc8

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PD02_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PD2_sel	rw	0x00	PD2_sel: 端口 PD2 功能选择。 000: GPIO PD2 001: PCA_ECI PCA 模块外部时钟输入信号 010: LPUART0_RTS LPUART0 模块 RTS 信号 011: TIM1_ETR Timer1 模块外部时钟输入信号 100: Reserved Reserved 101: Reserved Reserved 110: Reserved Reserved 111: Reserved Reserved

8.12 端口 PF 功能选择寄存器

8.12.1 端口 PF0 功能配置寄存器 (PF0_SEL)

起始地址：0x40020C00

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

地址偏移：0x1040

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PF0_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PF0_sel	rw	0x00	端口 PF0 功能选择。 000:GPIO PF0 001:I2C0_SDA I2C0 模块数据信号 010:Reserved Reserved 011:UART1_TXD UART1 模块 TXD 信号 100:Reserved Reserved 101:Reserved Reserved 110:Reserved Reserved 111:Reserved Reserved

8.12.2 端口 PF1 功能配置寄存器 (PF1_SEL)

起始地址：0x40020C00

地址偏移：0x1044

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PF1_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved	rw	0x00	保留。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
2:0	PF1_sel	rw	0x00	PF1_sel: 端口 PF1 功能选择。 000:GPIO PF1 001:I2C0_SCL I2C0 模块时钟信号 010:TIM4_CHB Timer6 模块通道 0B 信号 011:UART1_RXD UART1 模块 RXD 信号 100:Reserved Reserved 101:Reserved Reserved 110:Reserved Reserved 111:Reserved Reserved

8.12.3 端口 PF6 功能配置寄存器 (PF6_SEL)

起始地址：0x40020C00

地址偏移：0x1058

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PF6_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved		0x00	保留。
2:0	PF6_sel	rw	0x00	PF6_sel: 端口 PF6 功能选择。 000:GPIO PF6 001:I2C1_SCL I2C1 模块时钟信号 010:LPUART1_CTS LPUART1 模块 CTS 信号 011:UART0_CTS UART0 模块 CTS 信号 100:Reserved Reserved 101:Reserved Reserved 110:Reserved Reserved 111:Reserved Reserved

8.12.4 端口 PF7 功能配置寄存器 (PF7_SEL)

起始地址：0x40020C00

地址偏移：0x105C

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PF7_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved	rw	0x00	保留。
2:0	PF7_sel	rw	0x00	PF07_sel: 端口 PF7 功能选择。 000: GPIO PF7 001: I2C1_SDA I2C1 模块数据信号 010: LPUART1_RTS LPUART1 模块 RTS 信号 011: UART0_RTS UART0 模块 RTS 信号 100: Reserved Reserved 101: Reserved Reserved 110: Reserved Reserved 111: Reserved Reserved

8.12.5 端口 PF11 功能配置寄存器 (PF11_SEL)

起始地址：0x40020C00

地址偏移：0x106C

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PF11_sel		
													rw		

Bit	Field	Type	Reset	Description
31:3	Reserved	rw	0x00	保留。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
2:0	PF11_sel	rw	0x00	PF11_sel: 端口 PF11 功能选择。 000: GPIO PF11 001: Reserved Reserved 010: Reserved Reserved 011: Reserved Reserved 100: Reserved Reserved 101: Reserved Reserved 110: Reserved Reserved 111: Reserved Reserved

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

9 I^2C 总线 (I^2C)

9.1 I^2C 简介

I^2C 是双线双向的串行总线，它为设备之间数据交换提供了一种简单高效的方法。I^2C 标准是一个具有冲突检测机制和仲裁机制的真正意义上的多主机总线。它能防止两个或者多个主机在同时请求控制总线时发生数据冲突。

I^2C 总线控制器，能满足 I^2C 总线的各种规格并支持所有与 I^2C 总线通信的传输模式。

I^2C 总线使用连接设备的"SCL"(串行时钟总线) 和"SDA"(串行数据总线) 来传送信息。

数据在主机与从机之间通过 SCL 时钟线控制在 SDA 数据线上实现一个字节一个字节的同步传输，每个字节为 8 位长度，一个 SCL 时钟脉冲传输一个数据位，数据由最高位 MSB 开始传输，每个传输字节后跟随一个应答位，每个位在 SCL 为高时采样；因此，SDA 线只有在 SCL 为低时才可以改变，在 SCL 为高时 SDA 保持稳定。当 SCL 为高时，SDA 线上的跳变视为命令中断（START 或 STOP)。I^2C 逻辑能自主地处理字节的传输。它能保持跟踪串行传送，而且还有一个状态寄存器（I^2Cx_STAT) 能反映 I^2C 总线控制器和 I^2C 总线的状态。

9.2 I^2C 主要特性

I^2C 控制器支持以下特性：

- 支持主机发送/接收，从机发送/接收四种工作模式；
- 支持标准 (100Kbps) / 快速 (400Kbps) / 高速 (1Mbps) 三种工作速率；
- 支持 7 位寻址功能；
- 支持噪声过滤功能；
- 支持广播地址；
- 支持中断状态查询功能。

9.3 I^2C 协议描述

通常标准 I^2C 传输协议包含四个部分：

1. 起始信号或重复起始信号；
2. 从机地址传输和读写位传输；
3. 数据传输；
4. 停止信号。

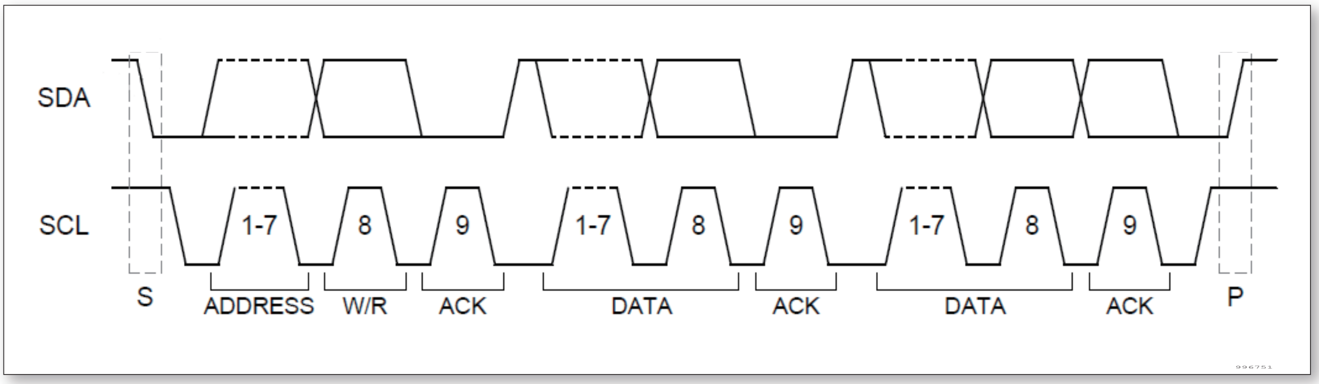

图 19. I^2C 传输协议

9.3.1 I²C 总线上数据传输

主机发出从机接收 7 位地址（一个字节）传输方向未改变。

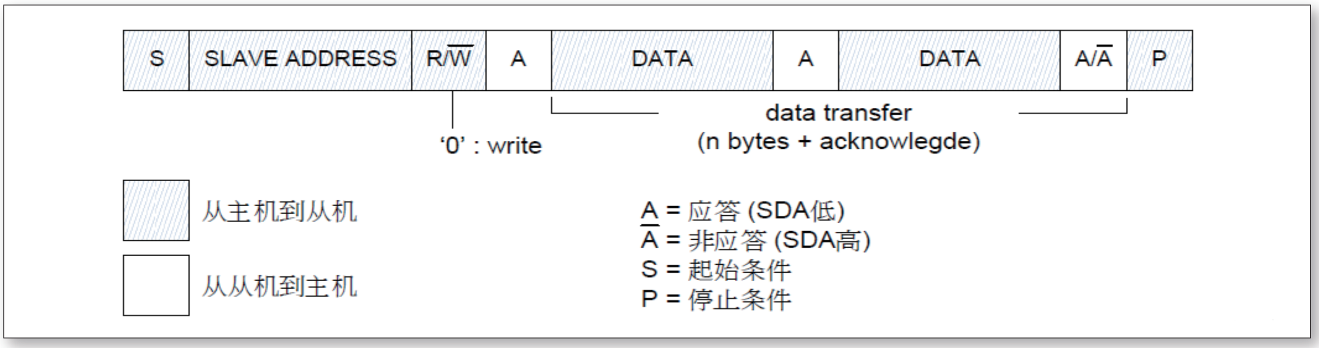

图 20. 主机向从机传输数据

第一个字节后主机紧接着由从机读取数据 (内容为从机地址) 传输方向改变。

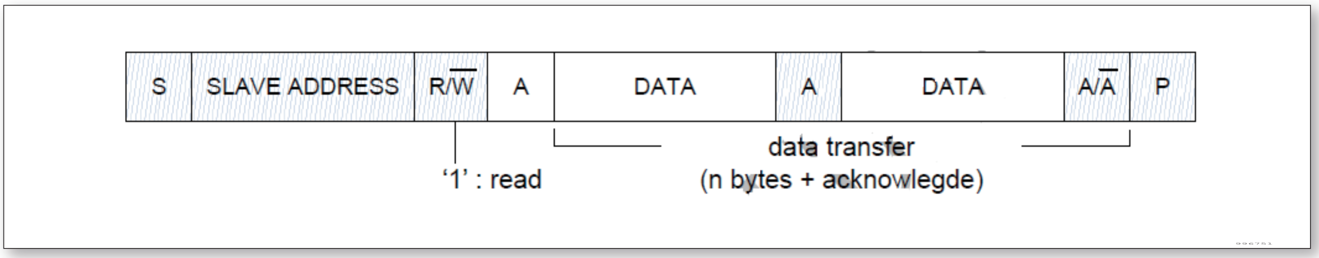

图 21. 主机由从机读取地址

9.3.2 起始位或重复起始信号

当总线处于空闲状态下，说明没有主机对总线发起传输请求（SCL 和 SDA 线同时为高），主机可以通过发送一个 START 信号来发起传输请求。

起始信号：通常表示为 S-bit。当 SCL 线为高时，SDA 线上信号由高至低，标示总线上产生起始信号，新的传输开始。

重复起始信号 (Sr)：即在两个 START 信号之间没有 STOP 信号。主机采用这种方法与另一个从机或相同的从机以不同传输方向进行通信（例如：从写入设备到从设备读出）而不释放总线。

STOP 信号：主机向总线发出停止信号结束数据传送。停止信号，通常用 P-bit 表示，当 SCL 线为高时，SDA 线上出现由低到高的信号，被定义为停止信号。

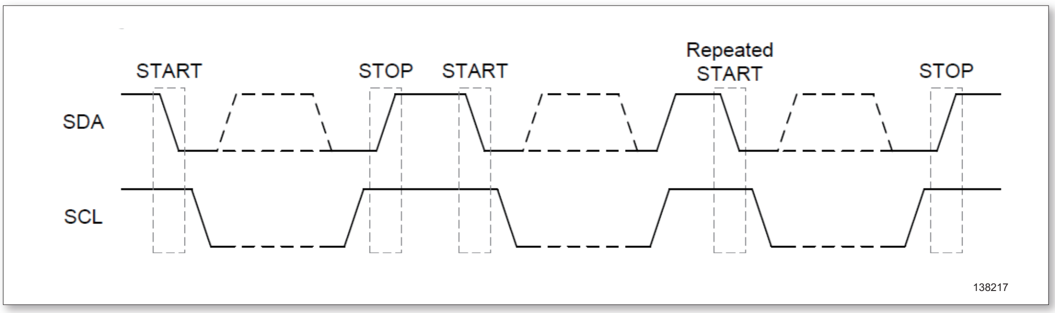

图 22. START 和 STOP 条件

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

9.3.3 从机地址传输

START 信号是从机地址时，主机立即传输数据的第一位。这是一个跟随有一个读写位的 7 位调用地址，读写位控制从机的信号传输方向。系统中没有两个从机有相同的地址，只有被主机寻址的从机会通过在第 9 个 SCL 时钟周期将 SDA 置为低电平作为应答。

9.3.4 数据传输

当从机地址被成功识别，就可以根据读写所决定的方向，开始一字节一字节的数据传输，每个传输字节最后带一个第 9 时钟周期上的响应信号，如果从机上产生无响应信号 (NACK)，主机可以产生停止信号来退出数据传输，或者产生重复起始信号开始新一轮的数据传输。

当主机作为接收器件时，发生无响应信号 (NACK)，从机释放 SDA 线，使主机产生停止信号或重复起始信号。

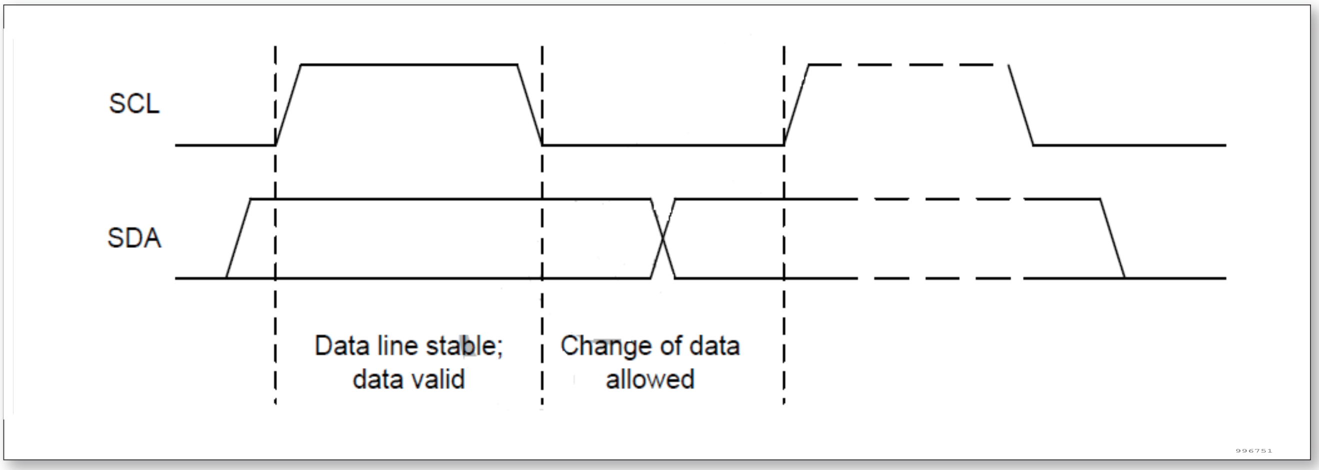

图 23. 主机由从机读取地址

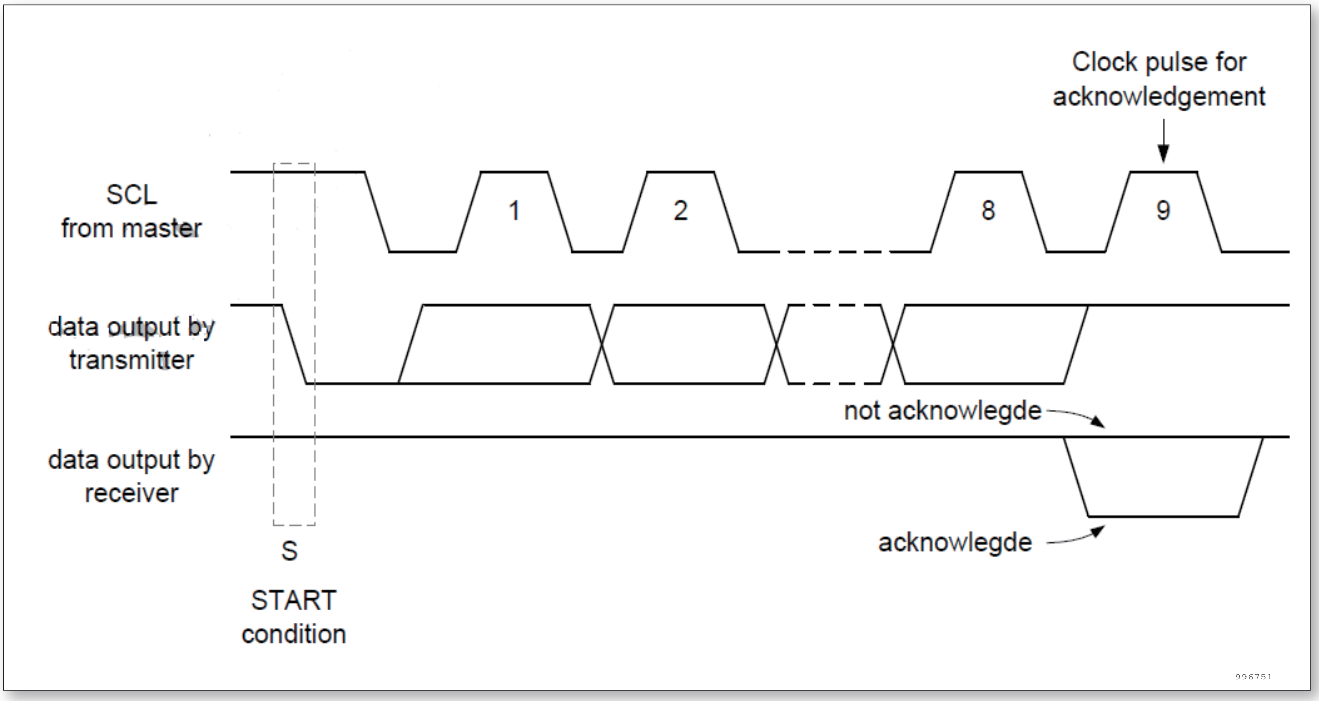

图 24. 主机由从机读取地址

9.4 I²C 功能描述

I²C 总线使用双线在连接到总线"SCL"(串行时钟线）和"SDA"(串行数据线）的设备间传送信息。由于只有无方向端口，I²C 组件需要使用到引脚的漏端开路缓冲器。每个连接到总线的设备都能使用软件通过特定地址寻址。I²C 标准是一个具有冲突检测机制和仲裁机制的真正意义上的多主机总线。它能防止两个或者多个主机在同时开始传输数据时发生数据冲突。滤波逻辑可以过滤数据总线上的毛刺来保护数据的完整性。

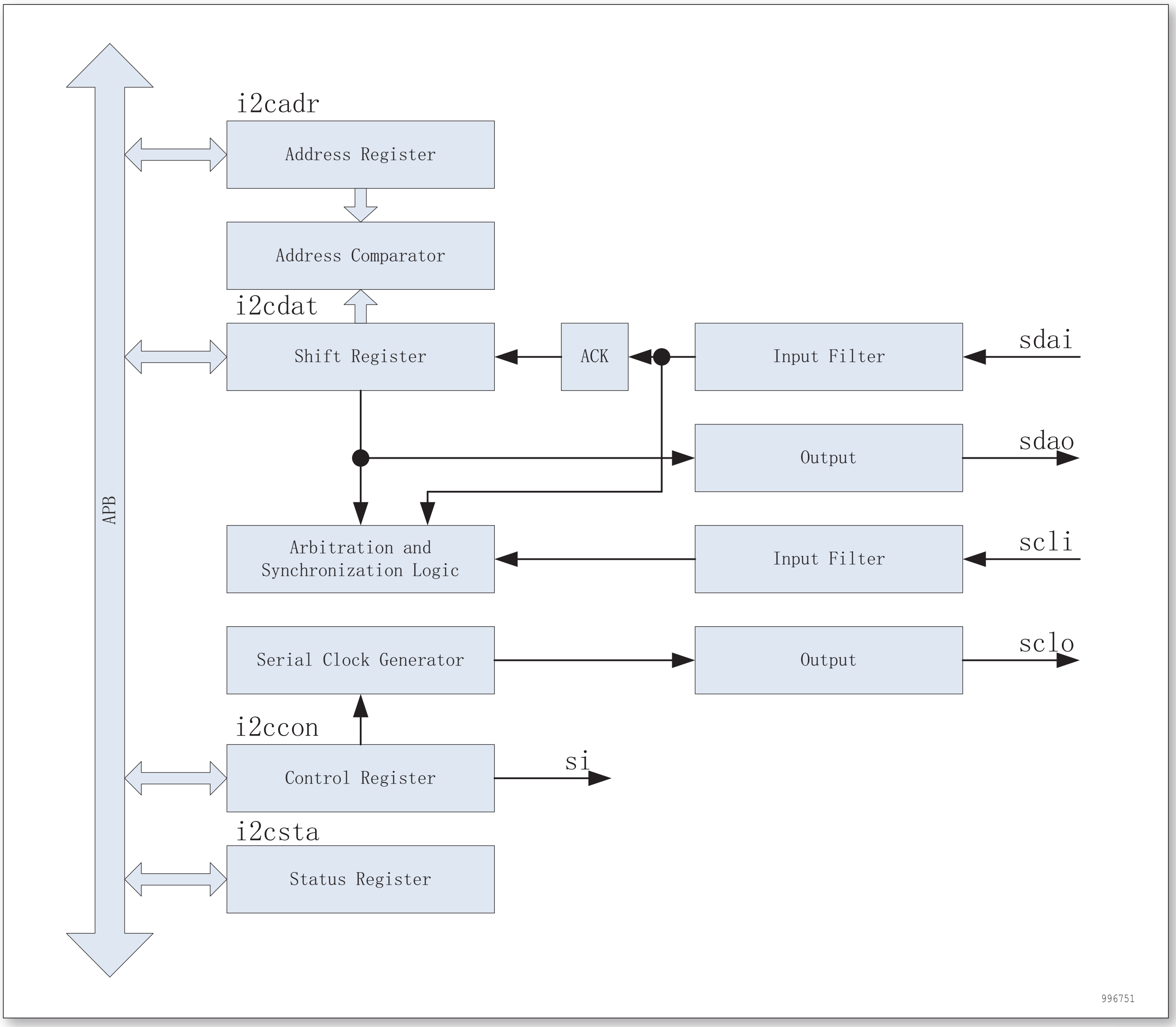

图 25. I²C 功能模块图

9.4.1 I²C 工作模式

I²C 组件可实现 8 位的双向数据传输,传输速率在标准模式下可达到 100Kbits/s 而在高速模式下可达 400Kbits/s，在超高速模式下可达 1Mbits/s，并且可以在以下四种模式下工作：

- 主机发送模式：当“SCL”输出串行时钟信号时“SDA”输出串行数据。
- 主机接收模式：当“SCL”输出串行时钟信号时串行数据通过“SDA”接收。
- 从机接收模式：串行数据和串行时钟分别通过“SDA”和“SCL”接收。
- 从机发送模式：当串行时钟从“SCL”口输入时串行数据通过“SDA”口发送。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

9.4.2 仲裁与同步逻辑

在主发送模式中，仲裁逻辑检查每个发送的逻辑 1 是否真正出现在总线上。如果总线的另一个器件撤消了一个逻辑 1 并将 SDA 线拉低，仲裁丢失，I²C 模块立刻由主发送器变为从接收器。I²C 模块将继续输出时钟脉冲（在 SCL 上），直至发送完当前的串行字节。

仲裁也可能在主接收模式中丢失。这种情况只在 I²C 模块正在向总线返回一个“非应答：（逻辑 1）”时出现。当总线的另一个器件将信号拉低时仲裁丢失。由于它只在串行字节结束时出现，因此 I²C 模块不会再产生时钟脉冲。

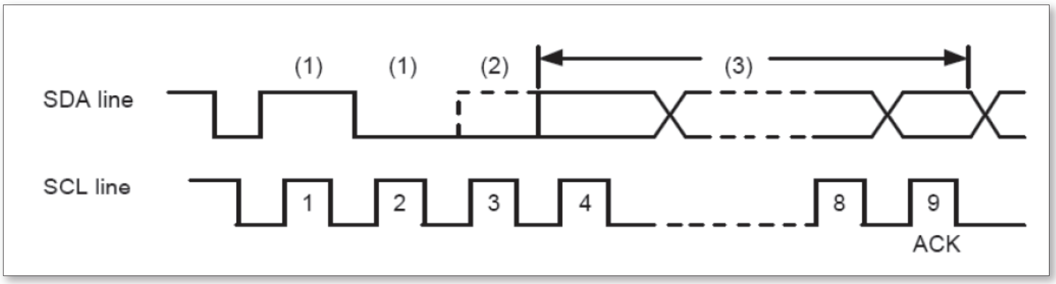

图 26. I²C 总线上的仲裁

1. 另一器件发送串行数据；
2. 另一器件通过拉低 SDA 先撤消了该 I²C 主机发送的一个逻辑 1（虚线）。仲裁丢失，I²C 进入从接收模式；
3. 此时 I²C 处于从接收模式，但仍产生时钟脉冲，直至发送完当前字节。I²C 将不为下个字节的传输产生时钟脉冲。一旦赢得仲裁，SDA 上的数据传输由新的主机来启动。

同步逻辑使得串行时钟发生器与另一个器件 SCL 线上的时钟脉冲同步。如果 2 个或更多主器件产生时钟脉冲，则高电平周期取决于产生最短高电平时间的器件；低电平周期取决于产生最长低电平时间的器件。

9.4.3 串行时钟发生器

串行时钟发生器采用一个 8 位的计数器作为波特率发生器，SCL 信号和 pclk 信号的频率关系为：

$$F_{scl} = F_{pclk}/8*(N+1)\ (N > 0)$$

下面的表格表示 pclk 为各种频率时，分频系数为 1-7 时，SCL 信号的频率值。

表 23: I²C 时钟信号波特率

频率 (Khz)	1	2	3	4	5	6	7
1000	62	41	31	25	20	17	15
2000	125	83	62	50	41	35	31
4000	250	166	125	100	83	71	62
6000	375	250	187	150	125	107	93
8000	500	333	250	200	166	142	125
10000	625	416	312	250	208	178	156
12000	750	500	375	300	250	214	187
14000	875	583	437	350	291	250	218

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

频率 (Khz)	1	2	3	4	5	6	7
16000	1000	666	500	400	333	285	250

9.4.4 输入滤波器

输入信号与时钟信号（clk）同步，低于 2 个时钟周期的尖锋脉冲信号会被滤除。每个滤波器由 3 个触发器组成。第一个触发器用来直接锁存输入信号，并将数据载入由另外两个构成的移位寄存器中。当第二和第三个触发器的状态是”11” 或”00” 时，内部的滤除信号会各自被置 1 或置 0。

9.4.5 地址比较器

I^2C 比较器将自己的从机地址与接收到的 7 位从机地址做比较。它可使用”I^2Cx_ADDR” 寄存器对自己的从机地址进行编程。并且会根据”I^2Cx_ADDR” 寄存器的”GC” 位与首次接收到的 8 位字节与通用调用地址（0x00）相比较。如果任何一者相同，”I^2Cx_CR” 寄存器的”si” 位会被置 1 并产生一个中断请求。

9.4.6 中断产生器

I^2C 模块的所有四种模式都被使用时，则有 26 种可能的总线状态。当 I^2C 进入 26 种状态的 25 种状态时，”I^2Cx_CR” 寄存器的”si” 标志位会被硬件置 1。”si” 位唯一不会被置 1 的状态是 F8h，这表明没有有效的相关状态信息。”si” 标志位必须通过软件清零。为了清除”si” 位，必须把 0 写入此位。若在”si” 里写 1 不会改变”si” 的值。为了确定中断的实际中断源，中断服务程序在清除”si” 标志位之前，会对 I^2C 状态寄存器进行查询。

9.4.7 I^2C 主机发送模式

必须将 ens 置“1”来使能 I^2C 模块。如果 aa 位复位，当另一个器件正变成总线主机时，I^2C 模块将不会应答其自身的从机地址或通用调用地址。换句话说，如果 aa 位复位，I^2C 接口就不能进入从机模式。

此时，可通过置位 sta 位进入主发送模式。一旦总线空闲，I^2C 逻辑会马上测试 I^2C 总线并产生一个起始条件。当发送起始条件时，串行中断标志（si）置位，状态寄存器（I^2Cx_STAT）中的状态代码为 0x08。中断服务程序利用该状态代码进入相应的状态服务程序，将从机地址和数据方向位（SLA+W）装入 I^2Cx_DATA。复位 I^2C_CR 的 si 位后串行传输继续进行。

当发送完从机地址和方向位且接收到一个应答位时，串行中断标志（si）再次置位，I^2Cx_STAT 中可能是一系列不同的状态代码。主机模式下为 0x18，0x20 或 0x38。每个状态代码对应的操作在表 24中详细介绍。在发送完重复起始条件（状态 0x10）后，I^2C 模块通过将 SLA+R 装入 I^2Cx_DATA 切换到主接收模式。

表 24. I^2C 主机发送模式状态表

状态代码	I^2C 总线和硬件状态	读/写 I^2Cx_DATA	写 I^2Cx_CR				I^2C 硬件执行的下一个动作
			sta	sto	si	aa	
08H	已发送起始条件	装入 SLA+W	X	0	0	X	将发送 SLA+W，接收 ACK
10H	已发送重复起始条件	装入 SLA+W	X	0	0	X	同上
		装入 SLA+R	X	0	0	X	将发送 SLA+R，I^2C 自动切换到主接收模式

状态代码	I^2C 总线和硬件状态	读/写 I^2Cx_DATA	写 I^2Cx_CR				I^2C 硬件执行的下一个动作
			sta	sto	si	aa	
18H	已发送 SLA+W 已接收 ACK	装入数据字节	0	0	0	X	将发送数据字节，将接收 ACK
		装入 0xFF 装	1	0	0	X	将发送重复起始条件
		无 I^2Cx_DATA 动作	0	1	0	X	将发送停止条件，sto 标志位复位
		无 I^2Cx_DATA 动作	1	1	0	X	将先发送停止条件，随后发送起始条件，sto 标志位复位
20H	已发送 SLA+W 已接收非 ACK	装入数据字节	0	0	0	X	将发送数据字节，将接收 ACK
		无 I^2Cx_DATA 动作	1	0	0	X	将发送重复起始条件
		无 I^2Cx_DATA 动作	0	1	0	X	将发送停止条件，sto 标志位复位
		无 I^2Cx_DATA 动作	1	1	0	X	将先发送停止条件，随后发送起始条件，sto 标志位复位
28H	已发送 I^2Cx_DATA 中的数据, 已接收 ACK	装入数据字节	0	0	0	X	将发送数据字节，将接收 ACK
		无 I^2Cx_DATA 动作	1	0	0	X	将发送重复起始条件
		无 I^2Cx_DATA 动作	0	1	0	X	将发送停止条件，sto 标志位复位
		无 I^2Cx_DATA 动作	1	1	0	X	将先发送停止条件，随后发送起始条件，sto 标志位复位
30H	已发送 I^2Cx_DATA 中的数据	装入数据字节	0	0	0	X	将发送数据字节，将接收 ACK
		无 I^2Cx_DATA 动作	1	0	0	X	将发送重复起始条件
		无 I^2Cx_DATA 动作	0	1	0	X	将发送停止条件，sto 标志位复位
		无 I^2Cx_DATA 动作	1	1	0	X	将先发送停止条件，随后发送起始条件，sto 标志位复位
38H	在 SLA+ 读写或写数据字节时丢失仲裁	无 I^2Cx_DATA 动作	0	0	0	X	I^2C 总线被释放。进入不可寻址从模式。
		无 I^2Cx_DATA 动作	0	1	0	X	当 I^2C 总线空闲时发送起始条件

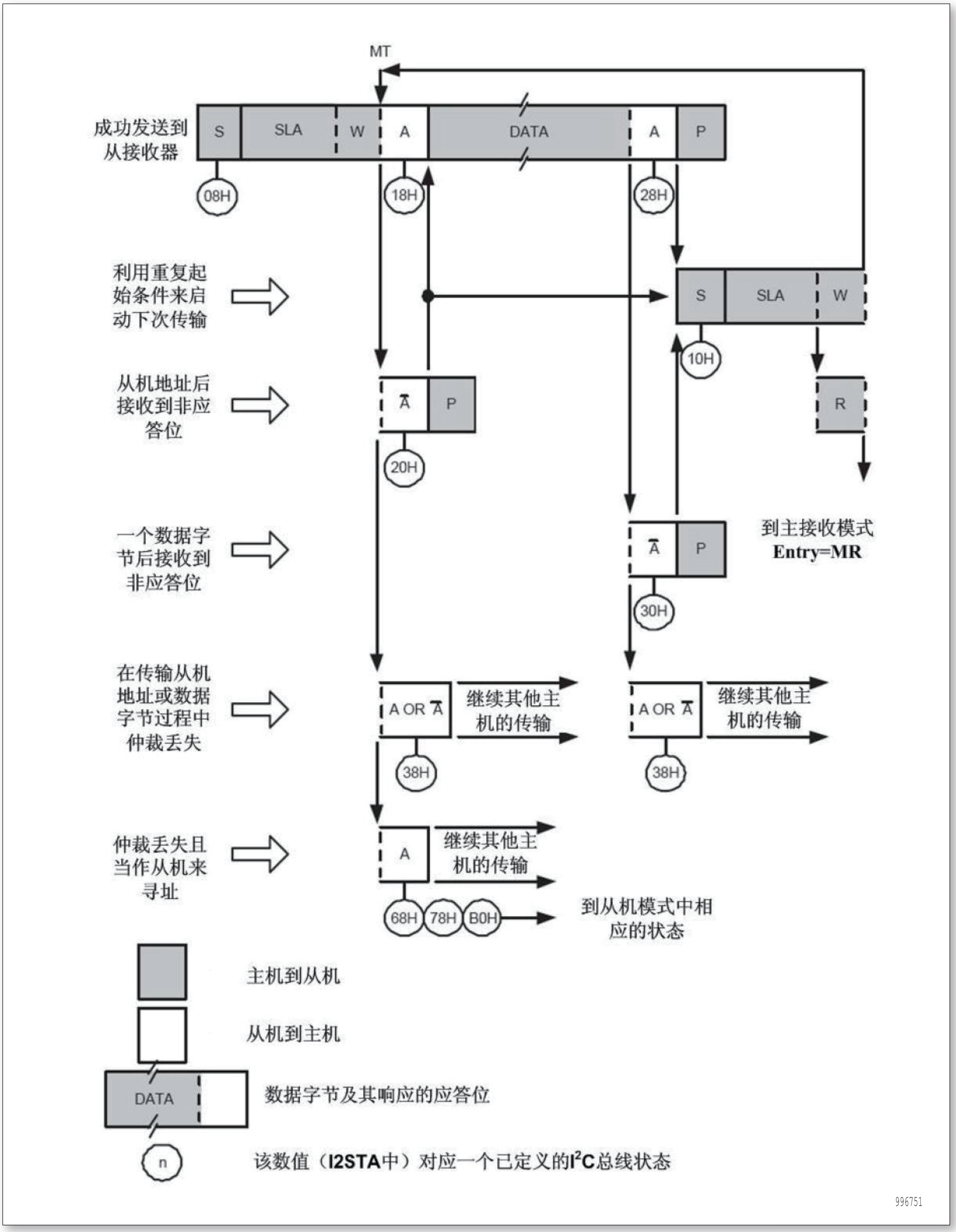

图 27. I²C 主机发送状态图

9.4.8 I²C 主机接收模式

在主接收模式中，主机所接收的数据字节来自从发送器。按主发送模式中的方法初始化传输。当发送完起始条件后，中断服务程序必须把 7 位从机地址和数据方向位（SLA+R）装入 I²Cx_DATA。清除 I²Cx_CR 中的 si 位后串行传输才会继续执行。

当发送完从机地址和数据方向位且接收到一个应答位时，串行中断标志 si 再次置位，这时，I²Cx_STAT 中可能是一系列不同的状态代码。主机模式下为 0x40，0x48 或 0x38。每个状态代码对应的操作详见表 25。在发送完重复起始条件（状态 0x10）后，I²C 模块通过将 SLA+W 装入 I²Cx_DATA 切换到主发送模式。

表 25. I²C 主机接收模式状态表

状态代码	I²C 总线和硬件状态	读/写 I²Cx_DATA	写 I²Cx_CR				I²C 硬件执行的下一个动作
			sta	sto	si	aa	
08H	已发送起始条件	装入 SLA+R	X	0	0	X	将发送 SLA+R，接收 ACK
10H	已发送重复起始条件	装入 SLA+R	X	0	0	X	同上
		装入 SLA+W	X	0	0	X	将发送 SLA+W，I²C 自动切换到主发送模式
38H	在非 ACK 中丢失仲裁	无 I²Cx_DATA 动作	0	0	0	X	I²C 总线将被释放；进入从模式
		无 I²Cx_DATA 动作	1	0	0	X	当总线空闲时发起起始条件
40H	已发送 SLA+R 已接收 ACK	无 I²Cx_DATA 动作	0	0	0	0	将接收数据字节，将返回非 ACK
		无 I²Cx_DATA 动作	0	0	0	1	将接收数据字节，将返回 ACK
48H	已发送 SLA+R 已接收非 ACK	无 I²Cx_DATA 动作	1	0	0	X	将发送重复起始条件
		无 I²Cx_DATA 动作	0	1	0	X	将发送停止条件，sto 标志位复位
		无 I²Cx_DATA 动作	1	1	0	X	将先发送停止条件，随后发送起始条件，sto 标志位复位
50H	已接收数据字节，ACK 已返回	读取数据字节	0	0	0	0	将接收数据字节，将返回非 ACK
		读取数据字节	0	0	0	1	将接收数据字节，将返回 ACK
58H	已接收数据字节，非 ACK 已返回	读取数据字节	1	0	0	X	将发送重复起始条件
		读取数据字节	0	1	0	X	将发送停止条件，sto 标志位复位

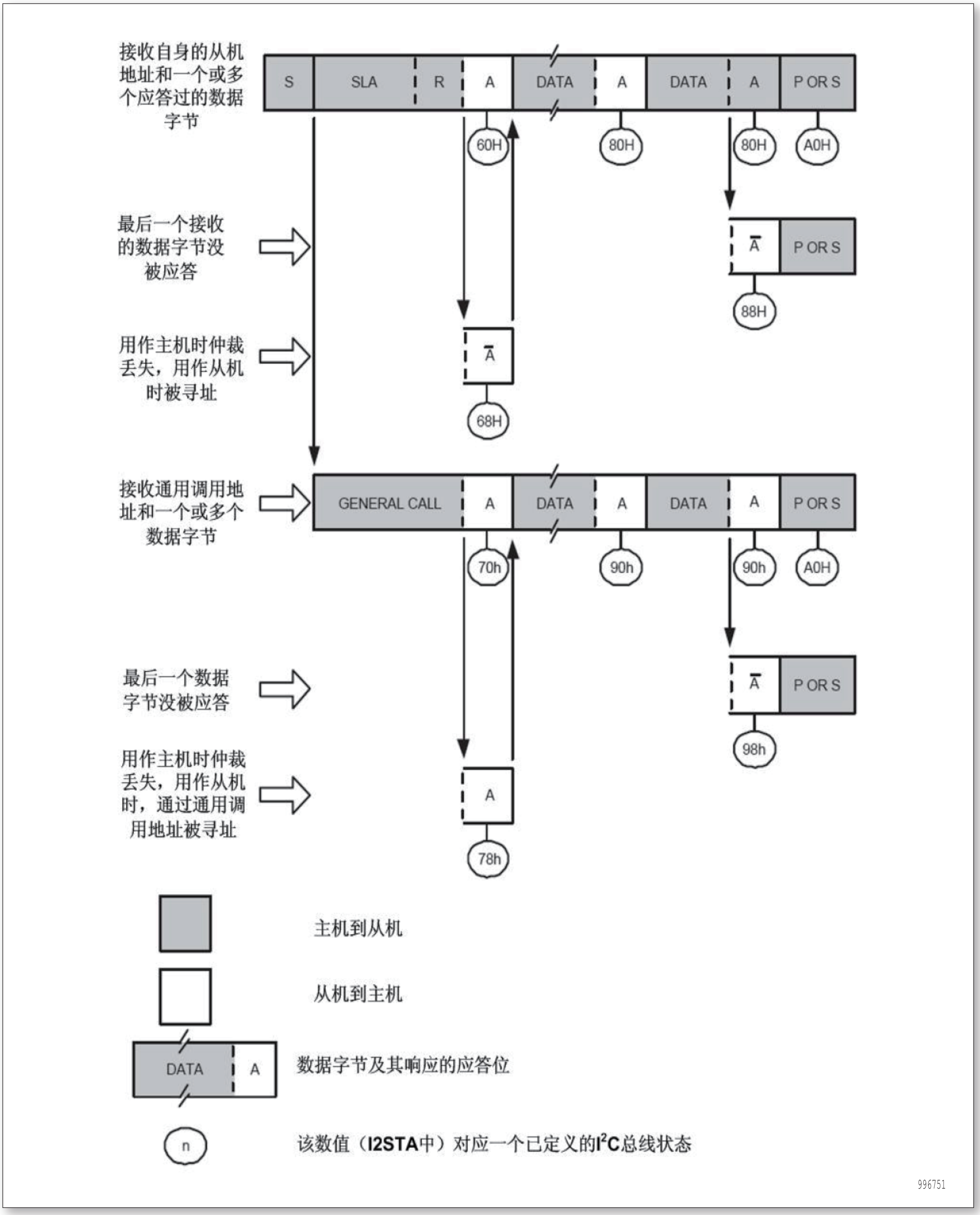

图 28. I²C 主机接收状态图

9.4.9 I^{2}C 从机接收模式

在从接收模式中，从机接收的数据字节来自主发送器。寄存器 I^{2}Cx_ADDR 的高 7 位是主机寻址时 I^{2}C 模块响应的地址。如果 LSB（GC）被置位，I^{2}C 模块将响应通用调用地址（0x00）；否则忽略通用调用地址。

I^{2}C 总线速率的设置不影响从机模式中的 I^{2}C 模块。必须置位 ena 来使能 I^{2}C 模块。aa 位必须置位以使能 I^{2}C 模块来应答其自身从机地址或通用调用地址。sta，sto 和 si 必须复位。

当 I^{2}Cx_ADDR 和 I^{2}Cx_CR 完成初始化后，I^{2}C 模块一直等待，直至被从机地址寻址，之后是数据方向位寻址，为了工作在从接收模式中，数据方向位必须为“0”（W）。接收完其自身的从机地址和 W 位后，串行中断标志（si）置位，可从 I^{2}Cx_STAT 中读出一个有效的状态代码。该状态代码用作状态服务程序的向量。每个状态代码的对应操作见表 26。如果 I^{2}C 模块在主机模式中仲裁丢失，也可进入从接收模式（请参考状态 0x68 和 0x78 的描述）。

如果 aa 位在传输过程中复位，则在接收完下一个数据字节后 I^{2}C 模块将向 SDA 返回一个非应答（逻辑 1）。当 aa 复位时，I^{2}C 模块不响应其自身的从机地址或通用调用地址。但是，I^{2}C 总线仍被监控，而且，地址识别可随时通过置位 aa 来恢复。这就意味着 aa 位可临时将 I^{2}C 模块从 I^{2}C 总线上分离出来。

表 26. I^{2}C 从机接收模式状态表

状态代码	I^{2}C 总线和硬件状态	读/写 I^{2}Cx_DATA	写 I^{2}Cx_CR				I^{2}C 硬件执行的下一个动作
			sta	sto	si	aa	
60H	已接收自身的 SLA+W；已接收 ACK;	无 I^{2}Cx_DATA 动作	X	0	0	0	将接收数据字节，将返回非 ACK
		无 I^{2}Cx_DATA 动作	X	0	0	1	将接收数据字节，将返回 ACK
68H	主控时在 SLA+ 读写丢失仲裁；已接收自身的 SLA+W；已返回 ACK;	无 I^{2}Cx_DATA 动作	X	0	0	0	将接收数据字节，将返回非 ACK
		无 I^{2}Cx_DATA 动作	X	0	0	1	将接收数据字节，将返回 ACK
70H	已接收通用调用地址 (0x00); 已返回 ACK;	无 I^{2}Cx_DATA 动作	X	0	0	0	将接收数据字节，将返回非 ACK
		无 I^{2}Cx_DATA 动作	X	0	0	1	将接收数据字节，将返回 ACK
78H	主控时在 SLA+ 读写中丢失仲裁；已接收通用调用地址；已返回 ACK;	无 I^{2}Cx_DATA 动作	X	0	0	0	将接收数据字节，将返回非 ACK
		无 I^{2}Cx_DATA 动作	X	0	0	1	将接收数据字节，将返回 ACK
80H	前一次寻址使用自身从地址；已接收数据字节；已返回 ACK;	无 I^{2}Cx_DATA 动作	X	0	0	0	将接收数据字节，将返回非 ACK
		无 I^{2}Cx_DATA 动作	X	1	0	0	将接收数据字节，将返回 ACK
88H	前一次寻址使用自身从地址；已接收数据字节；已返回 ACK;	读取数据字节	0	0	0	0	切换到不可寻址从模式；不识别自身从地址或通用地址；
		读取数据字节	0	0	0	1	
		读取数据字节	1	0	0	0	
		读取数据字节	1	0	0	1	切换到不可寻址从模式；不识别自身从地址或通用地址；当总线空闲后发送起始条件；

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

状态代码	I²C 总线和硬件状态	读/写 I²Cx_DATA	写 I²Cx_CR				I²C 硬件执行的下一个动作
			sta	sto	si	aa	
90H	前一次寻址使用通用调用地址；已接收数据字节；已返回 ACK	读取数据字节	1	0	0	X	将接收数据字节，将返回非 ACK
		读取数据字节	0	1	0	X	将接收数据字节，将返回 ACK
98H	前一次寻址使用通用调用地址；已接收数据字节；已返回非 ACK;	读取数据字节	0	0	0	0	切换到不可寻址从模式；不识别自身从地址或通用地址；
		读取数据字节	0	0	0	1	
		读取数据字节	1	0	0	0	切换到不可寻址从模式；不识别自身从地址或通用地址；当总线空闲后发送起始条件；
		读取数据字节	1	0	0	1	
A0H	当使用从接收/从发送模式中静态寻址时，接收到停止条件或重复起始条件	无 I²Cx_DATA 动作	0	0	0	0	切换到不可寻址从模式；不识别自身从地址或通用地址；
		无 I²Cx_DATA 动作	0	0	0	1	
		无 I²Cx_DATA 动作	1	0	0	0	切换到不可寻址从模式；不识别自身从地址或通用地址；当总线空闲后发送起始条件；
		无 I²Cx_DATA 动作	1	0	0	1	

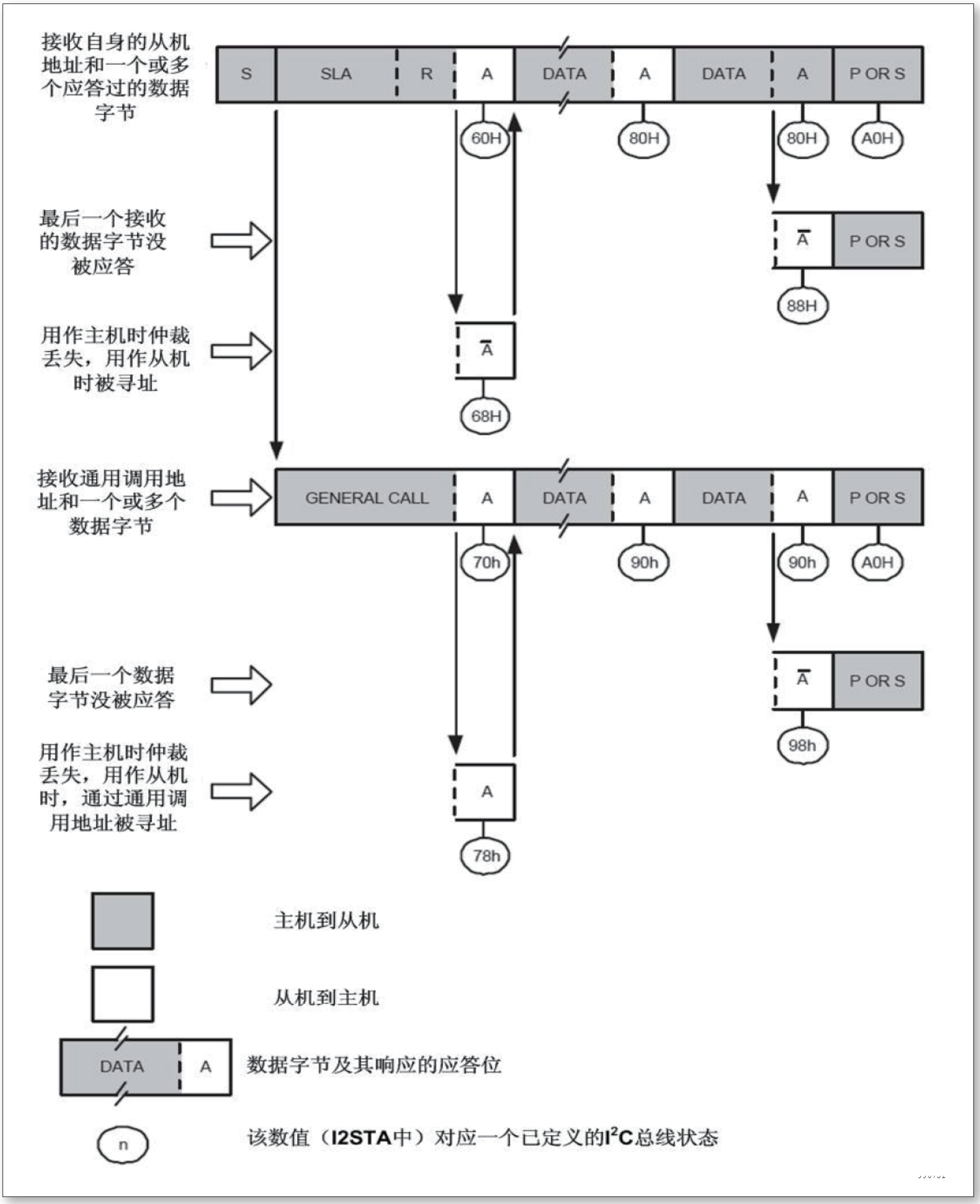

图 29. I²C 主机接收状态图

9.4.10 I^{2}C 从机发送模式

在从发送模式中，向主接收器发送数据字节。数据传输按照从接收模式中的情况初始化。当初始化 I^{2}Cx_ADDR 和 I^{2}Cx_CR 后，I^{2}C 模块一直等待，直至被自身的从机地址寻址，之后是数据方向位，该数据方向位必须为“1”（R），以便 I^{2}C 模块工作在从发送模式下。接收完其自身的从机地址和 R 位后，串行中断标志（si）置位，并且可从 I^{2}Cx_STAT 中读取一个有效的状态代码。该状态代码用作状态服务程序的向量，每个状态代码的对应操作见表 27所示。如果 I^{2}C 模块在主机模式下时仲裁丢失，则可进入从发送模式（见状态 0xB0）。

如果 aa 位在传输过程中复位，则 I^{2}C 模块将发送最后一个字节并进入状态 0xC0 或 0xC8。I^{2}C 模块切换到非寻址的从机模式，如果继续传输，它将忽略主接收器。因此主接收器接收所有 1 作为串行数据。当 aa 复位时，I^{2}C 模块不响应其自身的从机地址或通用调用地址。但是，I^{2}C 总线仍被监控，而且，地址识别可随时通过置位 aa 来恢复。这就意味着 aa 位可用来暂时将 I^{2}C 模块从 I^{2}C 总线上分离出来。

表 27. I^{2}C 从机发送模式状态表

状态代码	I^{2}C 总线和硬件状态	读/写 I^{2}Cx_DATA	写 I^{2}Cx_CR				I^{2}C 硬件执行的下一个动作
			sta	sto	si	aa	
A8H	已接收自身的 SLA+R; 已返回 ACK	装入数据字节	X	0	0	0	将发送最后一个数据字节; 将接收 ACK;
		装入数据字节	X	0	0	1	将发送一个数据字节; 将接收 ACK;
B0H	当主控时在 SLA+R 读写中丢失仲裁; 已接收自身 SLA+R; 已返回 ACK;	装入数据字节	X	0	0	0	将发送最后一个数据字节; 将接收 ACK;
		装入数据字节	X	0	0	1	将发送一个数据字节；将接收 ACK;
B8H	已发送数据; 已接收 ACK;	装入数据字节	X	0	0	0	将发送最后一个数据字节; 将接收 ACK;
		装入数据字节	X	0	0	1	将发送一个数据字节; 将接收 ACK;
C0H	已发送数据字节; 已接收非 ACK;	无 I^{2}Cx_DATA 动作	0	0	0	0	切换到不可寻址从模式; 不识别自身从地址或通用地址;
		无 I^{2}Cx_DATA 动作	0	0	0	1	
		无 I^{2}Cx_DATA 动作	1	0	0	0	
		无 I^{2}Cx_DATA 动作	1	0	0	1	切换到不可寻址从模式; 不识别自身从地址或通用地址; 当总线空闲后发送起始条件;
A0H	装入的数据字节已被发送；已接收 ACK；	无 I^{2}Cx_DATA 动作	0	0	0	0	切换到不可寻址从模式；不识别自身从地址或通用地址；
		无 I^{2}Cx_DATA 动作	0	0	0	1	
		无 I^{2}Cx_DATA 动作	1	0	0	0	切换到不可寻址从模式；不识别自身从地址或通用地址；当总线空闲后发送起始条件；
		无 I^{2}Cx_DATA 动作	1	0	0	1	

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

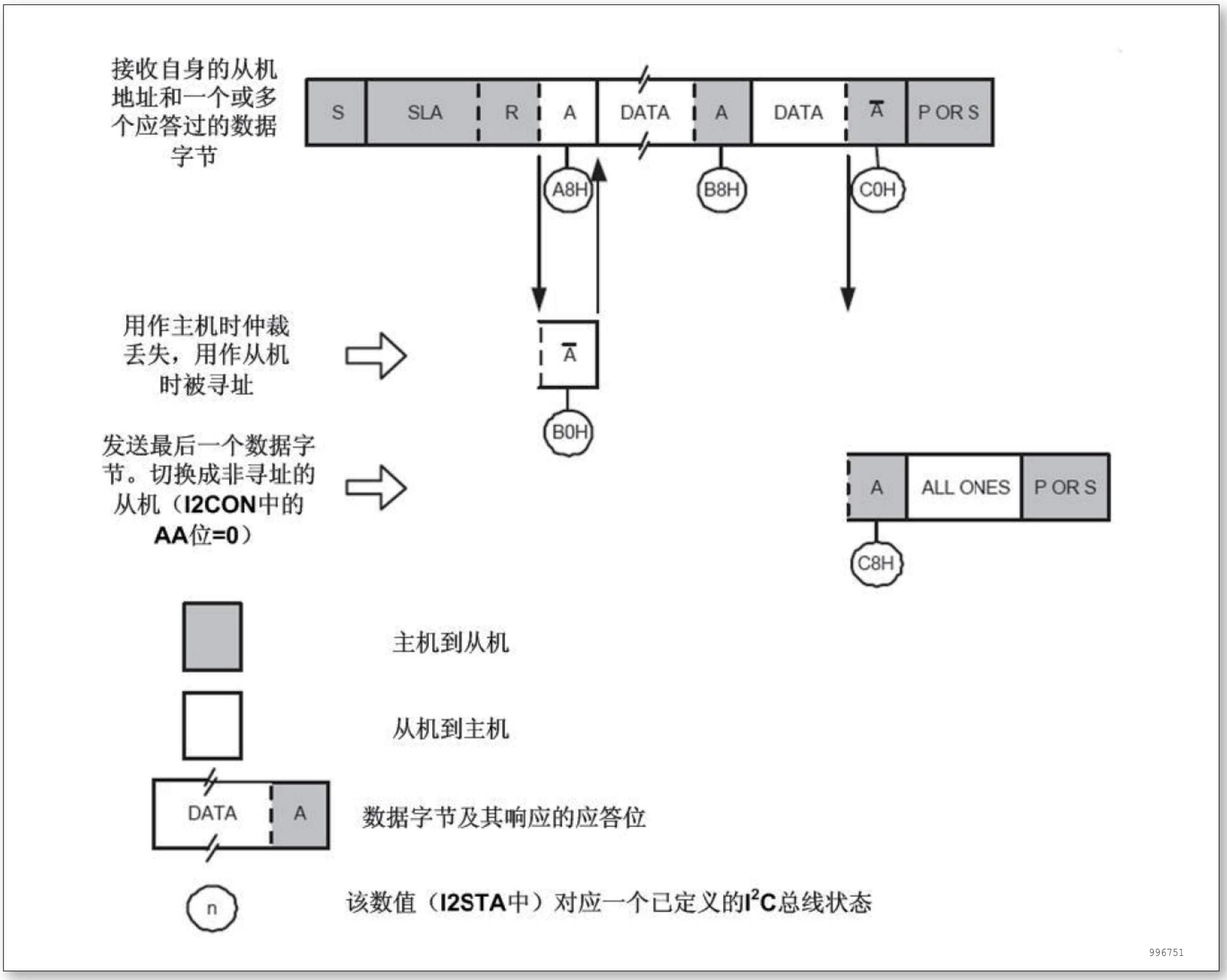

图 30. I²C 从机发送状态图

9.4.11 I²C 其他杂项状态

I²Cx_STAT= 0xF8：

这个状态码表示没有任何可用的相关信息，因为串行中断标志 si 还没有置位。这种情况在其它状态和 I²C 模块还未开始执行串行传输之间出现。

I²Cx_STAT= 0x00：

该状态代码表示在 I²C 串行传输过程中出现了总线错误。当格式帧的非法位置上出现了起始或停止条件时总线错误产生。这些非法位置是指在串行传输过程中的地址字节、数据字节或应答位。当外部干扰影响到内部 I²C 模块信号时也会产生总线错误。总线错误出现时 si 置位。要从总线错误中恢复，sto 标志必须置位，si 必须被清除。I²C 模块进入“非寻址的”从机模式（已定义的状态）并清除 sto 标志（I²Cx_CR 中的其它位不受影响）。SDA 和 SCL 线被释放（不发送停止条件）。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

表 28. 其他杂项状态表

状态代码	I^2C 总线和硬件状态	读/写 I^2Cx_DATA	写 I^2Cx_CR				I^2C 硬件执行的下一个动作
			sta	sto	si	aa	

9.5 I^2C 操作模式

9.5.1 初始化程序

将 I^2C 接口初始化用作从机或主机的例子。

1. 将自身的从机地址装入 I^2Cx_ADDR，使能通用调用识别（如果需要的话）；
2. 使能 I^2C 中断；
3. 向寄存器 I^2Cx_CR 写入 0x44 来置位 ens 和 aa 位，并使能从机功能。对于主机功能，可向寄存器 I^2Cx_CR 写入 0x40。

9.5.2 启动主机发送功能

通过建立缓冲区、指针和数据计数然后发启起始条件便可执行主发送操作。

1. 初始化主机数据计数器；
2. 建立数据将被发送到的从机地址，并且添加写位；
3. 置位 $I^2Cx\_CR.sta$；
4. 在主发送缓冲区内建立要发送的数据；
5. 初始化主机数据计数器来匹配正在发送的信息长度；
6. 退出。

9.5.3 启动主机接收功能

通过建立缓冲区、指针和数据计数然后发启起始条件便可执行主接收操作。

1. 初始化主机数据计数器；
2. 建立数据将被发送到的从机地址，并且添加读位；
3. 置位 $I^2Cx\_CR.sta$；
4. 在主接收缓冲区内建立要接收的数据；
5. 初始化主机数据计数器来匹配正在接收的信息长度；
6. 退出。

9.5.4 I^2C 中断程序

确定 I^2C 的状态和处理该状态的状态程序。

1. 从 I^2Cx_STAT 中读出 I^2C 的状态；
2. 使用状态值跳转到 26 个可能状态程序中的一个。

9.5.5 无指定模式的状态

1. 状态：0x00 总线错误。
   进入非寻址的从机模式并释放总线。
   (a) 置位 $I^2Cx\_CR.sto$ 和 $I^2Cx\_CR.aa$；

(b) 清除 I^2Cx_CR.si；

(c) 退出。

2. 主机状态

状态 08 和 10 适用于主发送模式和主接收模式。读写位决定了下一个状态是在主发送模式中还是在主接收模式中。

3. 状态：0x08

已发送起始条件。即将发送从机地址＋读写位和接收 ACK 位。

(a) 向 I^2Cx_DATA 写入从机地址和读写位；

(b) 置位 I^2Cx_CR.aa；

(c) 清除 I^2Cx_CR.si；

(d) 建立主发送模式数据缓冲区；

(e) 建立主接收模式数据缓冲区；

(f) 初始化主机数据计数器；

(g) 退出。

4. 状态：0x10

已发送重复起始条件。即将发送从机地址＋读写位和接收 ACK 位。

(a) 向 I^2Cx_DATA 写入从机地址和读写位；

(b) 置位 I^2Cx_CR.aa；

(c) 清除 I^2Cx_CR.si；

(d) 建立主发送模式数据缓冲区；

(e) 建立主接收模式数据缓冲区；

(f) 初始化主机数据计数器；

(g) 退出。

9.5.6 主发送器状态

1. 状态：0x18

之前状态为 8 或 10 表示已发送从机地址和写操作位，并接收了应答。即将发送第一个数据字节和接收 ACK 位。

(a) 将主发送缓冲区的第一个数据字节装入 I^2Cx_DATA；

(b) 清除 I^2Cx_CR.si；

(c) 主发送缓冲区指针加 1；

(d) 退出。

2. 状态：0x20

已发送从机地址和写操作位并接收了非应答。即将发送停止条件。

(a) 置位 I^2Cx_CR.sto；

(b) 清除 I^2Cx_CR.si；

(c) 退出。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

3. 状态：0x28
   已发送数据并接收了 ACK。如果发送的数据是最后一个数据字节则发送一个停止条件，否则发送下一个数据字节。
   (a) 主机数据计数器减 1，如果发送的不是最后一个数据字节就跳至第 (e) 步；
   (b) 置位 I^2Cx_CR.sto；
   (c) 清除 I^2Cx_CR.si；
   (d) 退出；
   (e) 将主发送缓冲区的下一个数据字节装入 I^2Cx_DATA；
   (f) 清除 I^2Cx_CR.si；
   (g) 主机发送缓冲区指针加 1；
   (h) 退出。
4. 状态：0x30
   已发送数据并接收到非应答。即将发送停止条件；
   (a) 置位 I^2Cx_CR.sto；
   (b) 清除 I^2Cx_CR.si；
   (c) 退出。
5. 状态：0x38
   仲裁已在发送从机地址和写操作位或数据的过程中丢失。总线已被释放且进入非寻址的从机模式。当总线再次空闲时将发送一个新的起始条件。
   (a) 置位 I^2Cx_CR.sta；
   (b) 清除 I^2Cx_CR.si；
   (c) 退出。

9.5.7 主接收状态

1. 状态：0x40
   前面的状态是 08 或 10 表示已发送从机地址和读操作位，并接收到 ACK。将接收数据和返回 ACK。
   (a) 置位 I^2Cx_CR.aa；
   (b) 清除 I^2Cx_CR.si；
   (c) 退出。
2. 状态：0x48
   已发送从机地址和读操作位，并接收到非应答。将发送停止条件。
   (a) 置位 I^2Cx_CR.sto 和 I^2Cx_CR.aa；
   (b) 清除 I^2Cx_CR.si；
   (c) 退出。
3. 状态：0x50
   已接收到数据，并返回 ACK。将从 I^2Cx_DATA 读取数据。将接收其它的数据。如果这是最后一个数据字节，则返回非应答，否则返回 ACK。

(a) 读取 I^2Cx_DATA 中的数据字节，存放到主机接收缓冲区；

(b) 主机数据计数器减 1，如果不是最后一个数据字节就跳到第 (e) 步；

(c) 清除 $I^2Cx\_CR.si$；清除 $I^2Cx\_CR.aa$；

(d) 退出；

(e) 置位 $I^2Cx\_CR.aa$；

(f) 清除 $I^2Cx\_CR.si$；

(g) 主机接收缓冲区指针加 1；

(h) 退出。

4. 状态：0x58
已接收到数据，已返回非应答。将从 I^2Cx_DATA 中读取数据和发送停止条件。

(a) 读取 I^2Cx_DATA 中的数据字节，存放到主机接收缓冲区；

(b) 置位 $I^2Cx\_CR.sto$ 和 $I^2Cx\_CR.aa$；

(c) 清除 $I^2Cx\_CR.si$；

(d) 退出。

9.5.8 从接收器状态

1. 状态：0x60
已接收到自身从机地址和写操作位，已返回 ACK。将接收数据和返回 ACK。

(a) 置位 $I^2CxCR.aa$；

(b) 清除 $I^2CxCR.si$；

(c) 建立从接收模式数据缓冲区；

(d) 初始化从机数据计数器；

(e) 退出。

2. 状态：0x68
用作总线主机时仲裁已在传输从机地址和读写位时丢失。已接收到自身从机地址和写操作位，并已返回 ACK。将接收数据和返回 ACK。当总线再次空闲后置位 sta 来重启主机模式。

(a) 置位 $I^2Cx\_CR.sta$ 和 $I^2Cx\_CR.aa$；

(b) 清除 $I^2Cx\_CR.si$；

(c) 建立从接收模式数据缓冲区；

(d) 初始化从机数据计数器；

(e) 退出。

3. 状态：0x70
已接收到通用调用和返回 ACK。将接收数据和返回 ACK。

(a) 置位 $I^2Cx\_CR.aa$；

(b) 清除 $I^2Cx\_CR.si$；

(c) 建立从接收模式数据缓冲区；

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

(d) 初始化从机数据计数器；

(e) 退出。

4. 状态：0x78

用作总线主机时仲裁已在传输从机地址和读写位时丢失。已接收到通用调用和返回 ACK。将接收数据和返回 ACK。当总线再次空闲后置位 sta 来重启主机模式。

(a) 置位 I^2Cx_CR.sta 和 I^2Cx_CR.aa；

(b) 清除 I^2Cx_CR.si；

(c) 建立从接收模式数据缓冲区；

(d) 初始化从机数据计数器；

(e) 退出。

5. 状态：0x80

之前寻址自身从机地址。已接收到数据并返回 ACK。将读取其它数据。

(a) 读取 I^2Cx_DATA 的数据字节，存放到从机接收缓冲区。

(b) 从机数据计数器减 1，如果不是最后一个数据字节就跳到第 (e) 步；

(c) 清除 I^2Cx_CR.si；清除 I^2Cx_CR.aa；

(d) 退出；

(e) 置位 I^2Cx_CR.aa；

(f) 清除 I^2Cx_CR.si；

(g) 从机接收缓冲区指针加 1；

(h) 退出。

6. 状态：0x88

之前寻址自身从机地址。已接收到数据并返回非应答。不会保存接收到的数据。进入非寻址的从机模式。

(a) 置位 I^2Cx_CR.aa；

(b) 清除 I^2Cx_CR.si；

(c) 退出。

7. 状态：0x90

之前寻址通用调用地址。已接收到数据并返回 ACK。将保存接收到的数据。只接收第一个数据字节并返回 ACK。接收其它数据字节后返回非应答。

(a) 读取 I^2Cx_DATA 的数据字节，并放入从机接收缓冲区；

(b) 清除 I^2Cx_CR.si；清除 I^2Cx_CR.aa；

(c) 退出。

8. 状态：0x98

之前寻址通用调用地址。已接收到数据并返回非应答。不会保存接收到的数据。进入非寻址的从机模式。

(a) 置位 I^2Cx_CR.aa；

(b) 清除 I^2Cx_CR.si；

(c) 退出。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

9. 状态：0xA0
   已接收停止条件或重复起始条件，但仍作为从机寻址。不保存接收到的数据。进入非寻址的从机模式。
   (a) 置位 I^2Cx_CR.aa；
   (b) 清除 I^2Cx_CR.si；
   (c) 退出。

9.5.9 从发送器状态

1. 状态：0xA8
   已接收自身从机地址和读操作位并返回 ACK。将发送数据和接收 ACK 位。
   (a) 将从机发送缓冲区的第一个数据字节装入 I^2Cx_DATA；
   (b) 置位 I^2Cx_CR.aa；
   (c) 清除 I^2Cx_CR.si；
   (d) 建立从发送模式数据缓冲区；
   (e) 从机发送缓冲区指针加 1；
   (f) 退出。
2. 状态：0xB0
   用作总线主机时，在传输从机地址和读写位时丢失仲裁。已接收自身从机地址和读操作位并返回 ACK。将发送数据和接收 ACK 位。当总线再次空闲后置位 sta 来重启主机模式。
   (a) 将从机发送缓冲区的第一个数据字节装入 I^2Cx_DATA；
   (b) 置位 I^2Cx_CR.sta 和 I^2Cx_CR.aa；
   (c) 清除 I^2Cx_CR.si；
   (d) 建立从发送模式数据缓冲区；
   (e) 从机发送缓冲区指针加 1；
   (f) 退出。
3. 状态：0xB8
   已发送数据并接收到 ACK。将发送数据和接收 ACK 位。
   (a) 将从机发送缓冲区的数据字节装入 I^2Cx_DATA；
   (b) 置位 I^2Cx_CR.aa；
   (c) 清除 I^2Cx_CR.si；
   (d) 从机发送缓冲区指针加 1；
   (e) 退出。
4. 状态：0xC0
   已发送数据并接收到非应答。进入非寻址的从机模式。
   (a) 置位 I^2Cx_CR.aa；
   (b) 清除 I^2Cx_CR.si；
   (c) 退出。

5. 状态：0xC8
   已发送最后一个数据字节并接收到 ACK。进入非寻址的从机模式。
   (a) 置位 $I^2Cx\_CR.aa$；
   (b) 清除 $I^2Cx\_CR.si$；
   (c) 退出。

9.6 I^2C 寄存器

I2C0 基地址：0x40000400

I2C1 基地址：0x40004400

表 29. I^2C 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x00	I^2Cx_TMRUN	波特率计数器使能寄存器。	0x0000 0000	小节 9.6.1
0x04	I^2Cx_TM	波特率计数器配置寄存器。	0x0000 0000	小节 9.6.2
0x08	I^2Cx_CR	配置寄存器。	0x0000 0000	小节 9.6.3
0x0c	I^2Cx_DATA	数据寄存器。	0x0000 0000	小节 9.6.4
0x10	I^2Cx_ADDR	地址寄存器。	0x0000 0000	小节 9.6.5
0x14	I^2Cx_STAT	状态寄存器。	0x0000 0000	小节 9.6.6

9.6.1 波特率计数器使能寄存器 (I2Cx_TMRUN)

起始地址：I^2C0：0x40000400　I^2C1：0x40004400

偏移地址：0x00

复位值：0x00000000

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

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

Bit	Field	Type	Reset	Description
31:1	Reserved			保留。
0	tme	rw	0x00	波特率计数器使能。0 –禁止 1 –使能

9.6.2 波特率计数器配置寄存器 (I2Cx_TM)

起始地址：I^2C0：0x40000400　I^2C1：0x40004400

偏移地址：0x04

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

复位值：0x00000000

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="8" rowspan="2">Reserved</td><td colspan="8">tm</td></tr>
<tr><td colspan="8">RW</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:8	Reserved			保留。
7:0	tm	rw	0x00	tm: 波特率计数器配置值。 Fo = Fi/8*(N+1) N = tm N >0

9.6.3 配置寄存器 (I2Cx_CR)

起始地址：I^2C0：0x40000400　I^2C1：0x40004400

偏移地址：0x08

复位值：0x00000000

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="9" rowspan="2">Reserved</td><td>ens</td><td>sta</td><td>sto</td><td>si</td><td>aa</td><td rowspan="2">Res</td><td>h1m</td></tr>
<tr><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:7	Reserved			保留。
6	ens	rw	0x00	I^2C 模块使能。0 -禁止 1 -使能
5	sta	rw	0x00	开始标志使能。0 -禁止 1 -使能
4	sto	rw	0x00	停止标志使能。0 -禁止 1 -使能
3	si	rw	0x00	I^2C 中断标志位。
2	aa	rw	0x00	应答标志使能。0 -禁止 1 -使能
1	Reserved			保留。
0	h1m	rw	0x00	I^2C 高速 1Mbps 模式使能。0 -禁止 1 -使能 当 $I^2Cx\_TM<9$ 时必须使能

9.6.4 数据寄存器 (I2Cx_DATA)

起始地址：I^{2}C0：0x40000400　I^{2}C1：0x40004400

偏移地址：0x0c

复位值：0x00000000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								i2cdat							
								RW							

Bit	Field	Type	Reset	Description
31:8	Reserved			保留。
7: 0	i2cdat	rw	0x00	I^{2}C 数据寄存器。 在 I^{2}C 发送模式下, 写发送数据到这个寄存器。 在 I^{2}C 接收模式下, 读接收数据从这个寄存器。

9.6.5 地址寄存器 (I2Cx_ADDR)

起始地址：I^{2}C0：0x40000400　I^{2}C1：0x40004400

偏移地址：0x10

复位值：0x00000000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								i2cadr							GC
								RW							RW

Bit	Field	Type	Reset	Description
31:8	Reserved			保留。
7:1	i2cadr	rw	0x00	I^{2}C 从机模式地址。
0	GC	rw	0x00	广播地址应答使能。0 -禁止 1 -使能。

9.6.6 状态寄存器 (I2Cx_STAT)

起始地址：I^{2}C0：0x40000400　I^{2}C1：0x40004400

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

偏移地址：0x14

复位值：0x00000000

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="8" rowspan="2">Reserved</td><td colspan="8">i2csta</td></tr>
<tr><td colspan="8">RO</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:1	Reserved			保留。
7:0	i2csta	r	0x00	I^2C 状态寄存器。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

10 串行外设接口 (SPI)

10.1 SPI 简述

SPI（Serial Peripheral Interface）总线是一种同步串行外设接口，它可以使 MCU 与各种外围设备以串行的方式进行信息交换。SPI 接口使用 4 条线：串行时钟线（SCK），主机输出/从机输入线（MOSI），主机输入/从机输出线（MISO），低电平有效从机选择线（SSN）。

10.2 SPI 主要特征

SPI 控制器支持以下特性：

- 通过编程可以配置为主机或者从机；
- 全双工通信；
- 四线/三线/双线模式（参见“SPI 管脚配置说明”）；
- 主机模式 7 种波特率可配置；
- 主机模式最大波特率为 1/2 系统时钟；
- 从机模式最大波特率为 1/4 系统时钟；
- 可配置的串行时钟极性和相位；
- 支持中断；
- 8 位数据传输，先传输高位后低位；
- 支持 DMA 软件/硬件访问。

10.3 SPI 功能描述

10.3.1 SPI 主机查询方式

操作流程

1. 端口配置；配置端口控制器，把 CS,SCK,MISO,MOSI 信号映射到正确的管脚；
2. SPI 波特率配置；设置 SPIx_CR.spr2，SPIx_CR.spr1，SPIx_CR.spr0；
3. 串行时钟配置；设置 SPI 时钟极性 (SPIx_CR.cpol)，时钟相位 (SPIx_CR.cpha)。详细见 SPIx_CR 寄存器；
4. 主机模式配置；SPIx_CR.mstr=1；
5. SPI 使能打开；SPIx_CR.spen=1；
6. 主机模式下由寄存器 SPIx_SSN.ssn 的值决定片选信号 SPI_CS 的输出电平；
7. 打开从机选择；配置 SPIx_SSN.ssn=0；
8. 向从机发送数据流程：查询 SPIx_STAT.txe，txe 为 1 表示发送缓冲器为空，可以把要发送的数据写到 SPI 数据寄存器 SPIx_DATA 中；txe 为 0 表示发送缓冲器已满，不能写数据到 SPIx_DATA 中。循环查询 SPIx_STAT. txe，直到所有的数据发送结束；
9. 从从机接收数据流程：查询 SPIx_STAT.rxne，rxne 为 1 表示接收缓冲器已满，可以读取 SPI 数据寄存器 SPIx_DATA，得到已经接收到的数据；rxne 为 0 表示接收缓冲器为空，读取 SPIx_DATA 只能得到上一次接收到的数据。循环查询 SPIx_STAT. rxne，直到所有的数据接收结束。

注：在多机通信中，可以用 GPIO 实现多个片选输出。

10.3.2 SPI 主机 DMA 方式

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

操作流程

1. 端口配置；配置端口控制器，把 CS,SCK,MISO,MOSI 信号映射到正确的管脚；
2. 打开系统控制器 DMAC 时钟使能寄存器；
3. 使能 NVIC DMAC 中断；
4. 根据传输需求配置 DMAC 相关寄存器；由于硬件缓冲器的大小限制，SPI 的 DMA 只支持 Hardware Block Transfer Mode；
5. 使能 SPIx_CR2 寄存器 hdma_tx 位，hdma_rx 位；
6. SPI 波特率配置；设置 SPIx_CR.spr2，SPIx_CR.spr1，SPIx_CR.spr0；
7. 串行时钟配置；设置 SPI 时钟极性 (SPIx_CR.cpol)，时钟相 (SPIx_CR.cpha)。详细见 SPIx_CR 寄存器；
8. 主机模式配置；SPIx_CR.mstr=1；
9. SPI 使能打开；SPIx_CR.spen=1；
10. 主机模式下由寄存器 SPIx_SSN.ssn 的值决定片选信号 SPI_CS 的输出电平；
11. 打开从机选择；配置 SPIx_SSN.ssn=0；
12. 向从机发送数据流程：循环等待 DMAC 中断发生。退出循环
13. 从从机接收数据流程：循环等待 DMAC 中断发生。退出循环
14. DMAC 中断服务子程序参考；
15. 读取寄存器 DMA_CABx.SS, 判断 DMAC 传输状态。SS=1,2,3,4 时表示 DMAC 传输出错，需要重新发送或者接收数据。SS=5 时表示 DMAC 传输成功结束，清零 SS 值。关闭 SPIx_CR2 寄存器 hdma_tx 位，hdma_rx 位。

注：重复步骤 5~13，可以实现 DMAC 持续传输。

10.3.3 SPI 主机数据时钟时序

使用 SPI 控制寄存器 SPIx_CR 中的时钟极性 cpol 和相位 cpha，可以将串行时钟 SCK 配置为四种模式之一。SPIx_CR.cpol 是选择 SPI 空闲时 SCK 是处于高电平还是低电平。SPIx_CR.cpha 是选择 SCK 的时钟相位（采样数据所用的边沿）。主机和从机必须配置为相同的时钟相位和极性。波特率的设置只在主机模式下有效，从机模式下的波特率设置会被忽略掉。主机模式数据/时钟时序详见图 31。

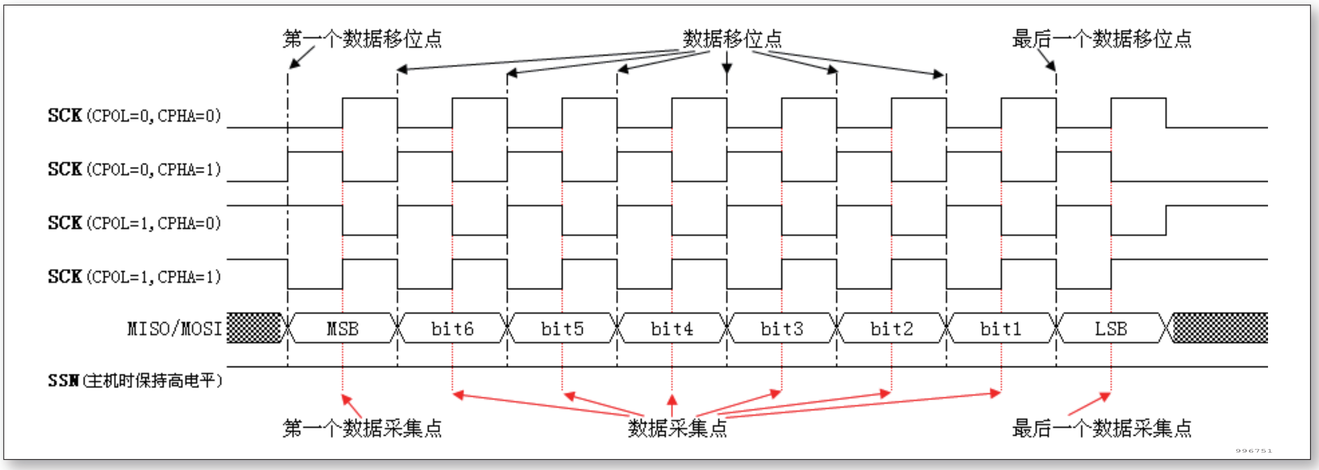

图 31. 主机模式数据/时钟时序

10.3.4 SPI 从机查询方式

SPI 工作在主机还是从机模式，由它在被使能时的主机/从机控制位 SPIx_CR.mstr 来决定。通过将主机/从机控制位 SPIx_CR.mstr 置“0”将 SPI 置于从机模式。当 SPI 处于从机模式时，由主器件控制串行时钟（SCK），从 MOSI 移入数据，SPI 逻辑中的计数器对 SCK 边沿计数，当 8 位数据传输完成后，spif 标志被置“1”，可以通过读 SPIx_DATA 得到接收到的数据。从机模式下的 SPI 作为从器件不能主动发起数据传输，只能通过写 SPIx_DATA 来预装要发送给主器件的数据，在主器件 SCK 的作用下，一位一位的移到 MISO 线上发送给主器件。

操作流程

1. 端口配置；配置端口控制器，把 SCK,MISO,MOSI 信号映射到正确的管脚；
2. 从机模式下由端口控制寄存器选择一个 GPIO 作为片选信号的来源，参见端口控制寄存器；
3. 串行时钟配置；设置 SPI 时钟极性 (SPIx_CR.cpol)，时钟相位 (SPIx_CR.cpha)。详细见 SPIx_CR 寄存器；
4. 从机模式配置；SPIx_CR.mstr=0；
5. SPI 使能打开；SPIx_CR.spen=1；
6. 向主机发送数据流程：查询 SPIx_STAT.txe，txe 为 1 表示发送缓冲器为空，可以把要发送的数据写到 SPI 数据寄存器 SPIx_DATA 中；txe 为 0 表示发送缓冲器已满，不能写数据到 SPIx_DATA 中。循环查询 SPIx_STAT.txe，直到所有的数据发送结束；
7. 从主机接收数据流程：查询 SPIx_STAT.rxne，rxne 为 1 表示接收缓冲器已满，可以读取 SPI 数据寄存器 SPIx_DATA，得到已经接收到的数据；rxne 为 0 表示接收缓冲器为空，读取 SPIx_DATA 只能得到上一次接收到的数据。循环查询 SPIx_STAT. rxne，直到所有的数据接收结束。

10.3.5 SPI 从机 DMA 方式

SPI 工作在主机还是从机模式，由它在被使能时的主机/从机控制位 SPIx_CR.mstr 来决定。通过将主机/从机控制位 SPIx_CR.mstr 置“0”将 SPI 置于从机模式。当 SPI 处于从机模式时，由主器件控制串行时钟（SCK），从 MOSI 移入数据，SPI 逻辑中的计数器对 SCK 边沿计数，当 8 位数据传输完成后，spif 标志被置“1”，可以通过读 SPIx_DATA 得到接收到的数据。从机模式下的 SPI 作为从器件不能主动发起数据传输，只能通过写 SPIx_DATA 来预装要发送给主器件的数据，在主器件 SCK 的作用下，一位一位的移到 MISO 线上发送给主器件。

操作流程

1. 端口配置；配置端口控制器，把 SCK,MISO,MOSI 信号映射到正确的管脚；
2. 从机模式下由端口控制寄存器选择一个 GPIO 作为片选信号的来源，参见端口控制寄存器；
3. 打开系统控制器 DMAC 时钟使能寄存器；
4. 使能 NVIC DMAC 中断；
5. 根据传输需求配置 DMAC 相关寄存器；由于硬件缓冲器的大小限制，SPI 的 DMA 只支持 Hardware Block Transfer Mode；
6. 使能 SPIx_CR2 寄存器 hdma_tx 位，hdma_rx 位；
7. 串行时钟配置；设置 SPI 时钟极性 (SPIx_CR.cpol)，时钟相位 (SPIx_CR.cpha)。详细见 SPIx_CR 寄存器；
8. 从机模式配置；SPIx_CR.mstr=0；
9. SPI 使能打开；SPIx_CR.spen=1；
10. 向主机发送数据流程：循环等待 DMAC 中断发生。退出循环；
11. 从主机接收数据流程：循环等待 DMAC 中断发生。退出循环；

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

12. DMAC 中断服务子程序参考；
13. 读取寄存器 DMA_CABx.SS, 判断 DMAC 传输状态。SS=1,2,3,4 时表示 DMAC 传输出错，需要重新发送或者接收数据。SS=5 时表示 DMAC 传输成功结束，清零 SS 值。关闭 SPIx_CR2 寄存器 hdma_tx 位，hdma_rx 位。

注：重复步骤 5~11，可以实现 DMAC 持续传输。

10.3.6 SPI 从机数据时钟时序

从机模式数据/时钟时序详见图 32和图 33。

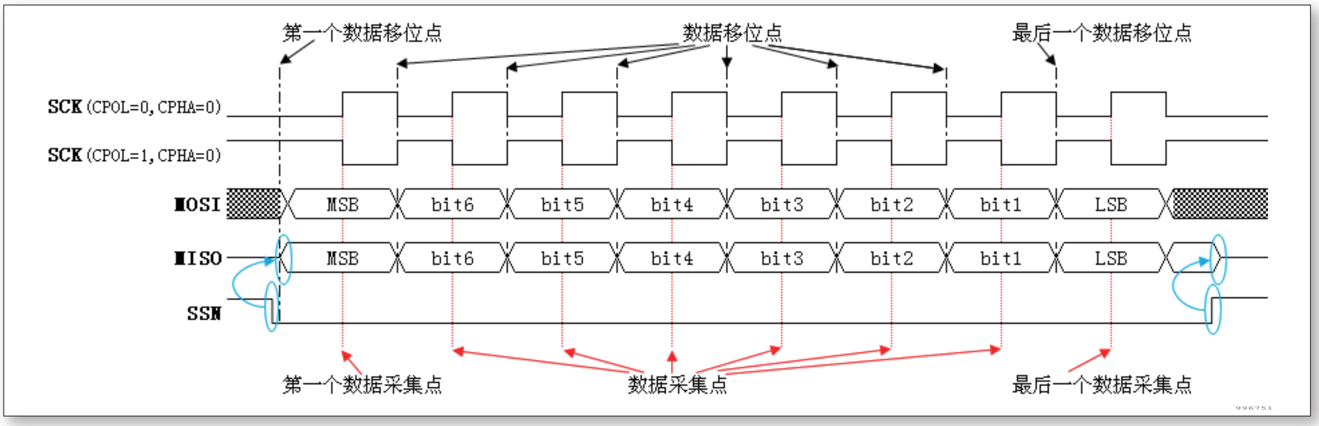

图 32. 从机模式数据/时钟时序图（cpha=0）

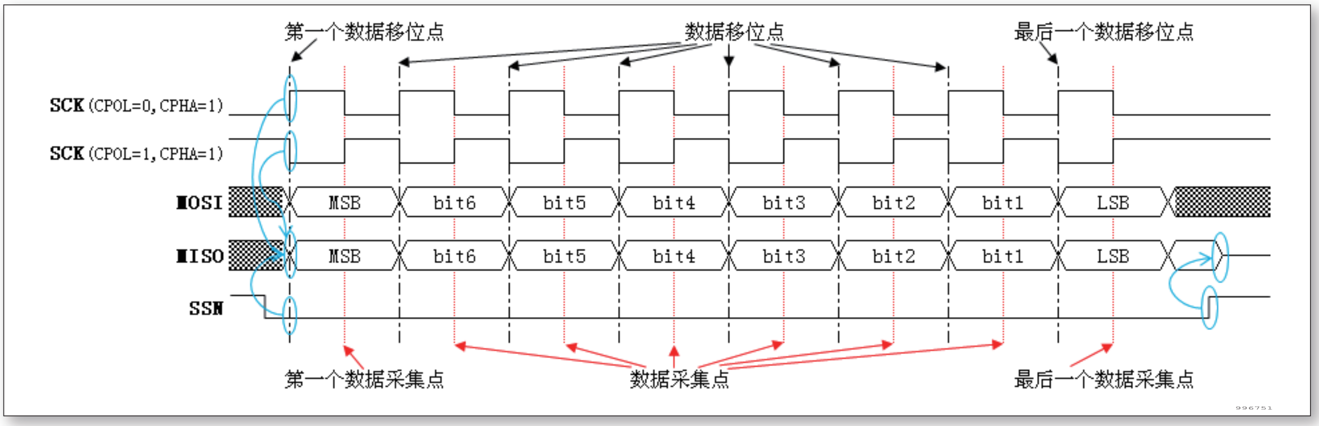

图 33. 从机模式数据/时钟时序图（cpha=1）

10.4 SPI 中断

如果 SPI 中断被允许 (SPIx_CR2.int_en 置“1”)，SPI 发送缓冲器为空标志位（SPIx_STAT.txe）被置“1”时或者 SPI 接收缓冲器非空标志位 (SPIx_STAT.rxne) 被置“1”时将产生中断；SPI 的主机模式错误中断标志位 (SPIx_STAT.mdf) 被置“1”也将产生中断。清除中断需要把 SPIx_ICLR 置“0”。

- 在每次字节传送结束，SPI 接收缓冲器非空标志位 (SPIx_STAT.rxne) 会被硬件置“1”。读 SPIx_DATA 即可清除该标志位。
- 在 SPI 发送缓冲器里的数据发送后，SPI 发送缓冲器空标志位 (SPIx_STAT.txe) 会被硬件置“1”。写 SPIx_DATA 即可清除该标志位。
- 当 SPI 被配置为主机模式时，如果外部 SSN 输入为低电平，就会和 SPI 的工作模式相冲突，SPI 的主机模式错误中断标志位 (SPIx_STAT.mdf) 被硬件置“1”。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

10.5 SPI 多主机/多从机系统的配置

本系列被应用在单主机单从机的 SPI 系统里时：如果作为主机，可以配置片选配置寄存器 SPIx_SSN.ssn，输出高低电平信号到 SPI_CS 管脚；如果作为从机，可以选择一个 GPIO 管脚作为 SPI_SSN 的来源（参见第 7 章 GPIO 端口控制器）。但是在多主机多从机系统里，必须按照相应流程配置，如下所示：

当系统为单主机多从机时，可以使用 SPI_CS 管脚作为从机 1 的片选信号，其他从机的片选信号通过 GPIO 管脚连接。当系统为多主机多从机时，所有的从机片选信号都通过 GPIO 管脚连接，主机还必须通过 GPIO 管脚和其他主机的 SPI_CS 信号相连，来监测总线是否被占用。

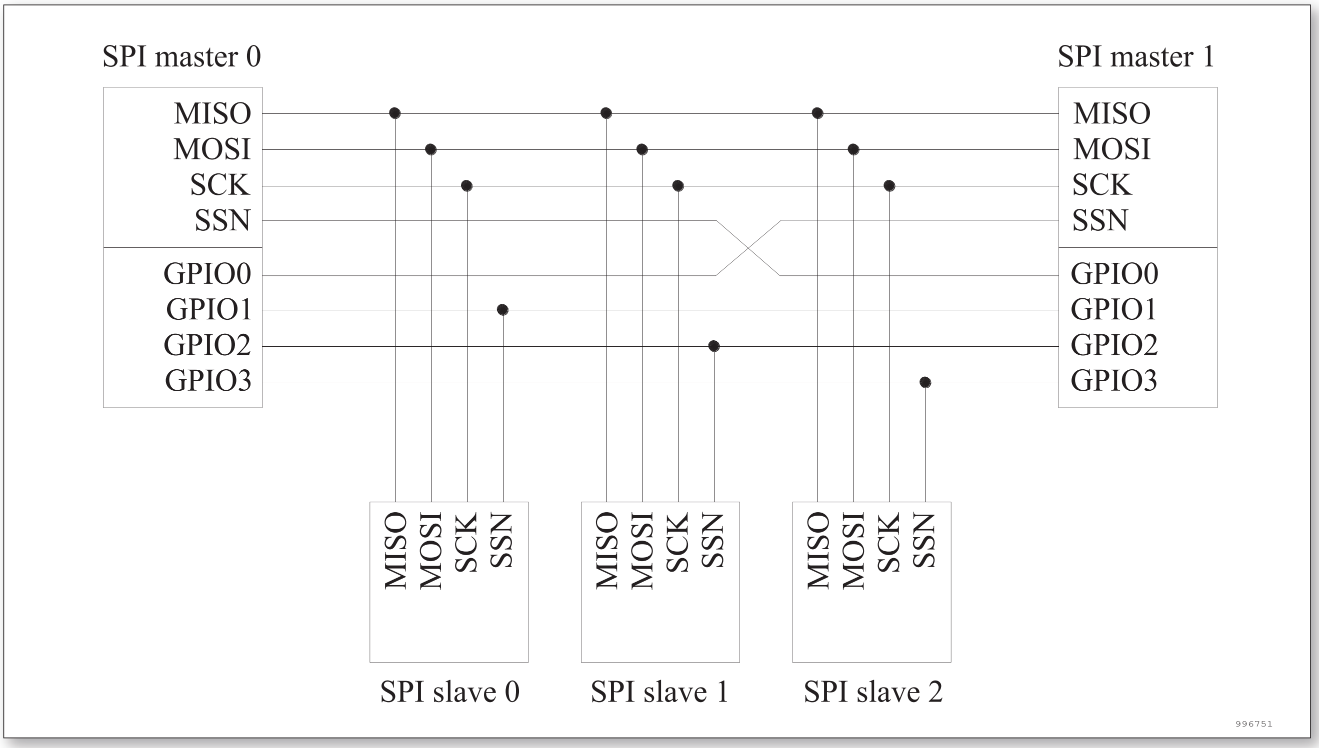

图 34. SPI 多主机/多从机系统的示意图

10.6 SPI 管脚配置说明

SPI 可以在一些特殊的管脚配置下保持部分或全部的功能。

具体情况如表 30（“✓”代表该管脚被配置并使用，空白代表该管脚未被配置）：

表 30: SPI 管脚配置说明表

	spi_cs(主)/ spi_ssn(从)	sck	mosi	miso	功能说明
主机模式	✓	✓	✓	✓	一般配置 全部主机功能正常
		✓	✓	✓	全部主机功能正常
	✓	✓	✓		全部发送功能正常
	✓	✓		✓	全部接收功能正常
		✓	✓		全部发送功能正常

	spi_cs(主)/ spi_ssn(从)	sck	mosi	miso	功能说明
		√		√	全部接收功能正常
从机模式	√	√	√	√	一般配置 全部从机功能正常
	√	√	√	√	从机接收功能正常
	√	√	√		从机发送功能正常
	固定低电平	√		√	全部从机功能正常
	固定低电平	√	√		从机接收功能正常
	固定低电平	√		√	从机发送功能正常

注：

- 表中未列举的情况暂不支持。
- 主机模式下，即使不使用 spi_cs 片选输出，也仍然要按照正常流程时的做法来操作 SPIx.SSN 寄存器——在 SPI 操作开始前必须清零，在 SPI 空闲时可以置位。
- 从机模式且片选输入固定为低电平时，为维持正常功能，必须满足 SPIx_CR.cpha=1。

10.7 SPI 寄存器描述

表 31. SPI 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x00	SPIx_CR	配置寄存器	0x0000 0014	小节 10.7.1
0x04	SPIx_SSN	片选配置寄存器	0x0000 00FF	小节 10.7.2
0x08	SPIx_STAT	状态寄存器	0x0000 0004	小节 10.7.3
0x0c	SPIx_DATA	数据寄存器	0x0000 0000	小节 10.7.4
0x10	SPIx_CR2	配置寄存器 2	0x0000 0003	小节 10.7.5
0x14	SPIx_ICLR	中断清除寄存器	0x0000 00FF	小节 10.7.6

10.7.1 配置寄存器 (SPIx_CR)

起始地址：SPI1：0x40004800

地址偏移：0x00

复位值：0x00000014

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								spr2	spen	Res	mstr	cpol	cpha	spr1	spr0
								RW	RW		RW	RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31:8	Reserved			保留。
7	spr2	rw	0x00	波特率选择位 2　参考 spr0。
6	spen	rw	0x00	模块使能寄存器。 0 -禁止 1 -使能
5	Reserved			保留。
4	mstr	rw	0x01	主机/从机模式选择。 0 -从机 1 -主机
3	cpol	rw	0x00	时钟极性选择寄存器。 0 -低 1 -高
2	cpha	rw	0x01	时钟相位选择寄存器。 0 -第一边沿 1 -第二边沿
1	spr1	rw	0x00	波特率选择位 1　参考 spr0。
0	spr0	rw	0x00	波特率选择位 0。 spr2 spr1 spr0 SCK Rate 0 0 0 $F_{sys}/2$ 0 0 1 $F_{sys}/4$ 0 1 0 $F_{sys}/8$ 0 1 1 $F_{sys}/16$ 1 0 0 $F_{sys}/32$ 1 0 1 $F_{sys}/64$ 1 1 0 $F_{sys}/128$ 1 1 1 0

10.7.2 片选配置寄存器 (SPIx_SSN)

起始地址：SPI1：0x40004800

地址偏移：0x04

复位值：0x000000FF

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

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

Bit	Field	Type	Reset	Description
31:1	Reserved			保留。
0	ssn	rw	0x01	ssn 输出值。 在主机模式下，软件配置 ssn 值控制 SPI_CS 端口电平高低

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description

10.7.3 状态寄存器 (SPIx_STAT)

起始地址：SPI1：0x40004800

地址偏移：0x08

复位值：0x00000004

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="8" rowspan="2">Reserved</td><td>spif</td><td rowspan="2">Res.</td><td>sserr</td><td>mdf</td><td>busy</td><td>txe</td><td>rxne</td><td rowspan="2">Res.</td></tr>
<tr><td>RO</td><td>RO</td><td>RO</td><td>RO</td><td>RO</td><td>RO</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:8	Reserved			保留。
7	spif	r	0x00	传输结束中断标志。
6	Reserved			保留。
5	sserr	r	0x00	从机模式 SSN 错误标志。
4	mdf	r	0x00	主机模式错误标志。
3	busy	r	0x00	SPI 总线忙标志。
2	txe	r	0x01	发送缓冲器空标志。
1	rxne	r	0x00	接收缓冲器非空标志。
0	Reserved			保留。

10.7.4 数据寄存器 (SPIx_DATA)

起始地址：SPI1：0x40004800

地址偏移：0x0c

复位值：0x00000000

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="8" rowspan="2">Reserved</td><td colspan="8">spdat</td></tr>
<tr><td colspan="8">RW</td></tr>
</table>

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31:8	Reserved			保留。
7:0	spdat	rw	0x00	数据寄存器。 在发送模式，向这个寄存器写发送值； 在接收模式，从这个寄存器读接收值；

10.7.5 配置寄存器 2(SPIx_CR2)

起始地址：SPI1：0x40004800

地址偏移：0x10

复位值：0x00000003

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									rxneie	txeie	hdma_tx	hdma_rx	int_en	Reserved	
									RW	RW	RW	RW	RW		

Bit	Field	Type	Reset	Description
31:7	Reserved			保留。
6	rxneie	rw	0x00	接收缓冲器非空中断使能。 0 -禁止 1 -使能
5	txeie	rw	0x00	发送缓冲器空中断使能。 0 -禁止 1 -使能
4	hdma_tx	rw	0x00	DMA 硬件访问发送使能。 0 -禁止 1 -使能
3	hdma_rx	rw	0x00	DMA 硬件访问接收使能。 0 -禁止 1 -使能
2	int_en	rw	0x00	SPI 中断使能。 0 -禁止 1 -使能
1:0	Reserved		0x03	为保证 SPI 正常工作，请勿修改本寄存器控制位的初值。

10.7.6 中断清除寄存器 (SPIx_ICLR)

起始地址：SPI1：0x40004800

地址偏移：0x14

复位值：0x000000FF

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved															int_clr R1W0

Bit	Field	Type	Reset	Description
31:7	Reserved			保留。
0	int_clr	rc_w0	0x01	SPI 中断清除。 0 -清除 1 -保持

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

11 时钟校准模块（CLKTRIM）

11.1 CLKTRIM 简介

CLKTRIM(Clock Trimming) 模块是一个专门用来校准/监测时钟的电路。在校准模式下选择精准的时钟源来校准不精准的时钟源，反复校准，调节不精准时钟源的参数，直到被校准时钟源的频率达到精度要求。校准模式下计数值会有一定的误差，但是在允许的精度误差范围内。在监测模式下选择稳定的时钟源来监测系统工作时钟，在设定的监测周期下，监测系统工作时钟是否有失效的情况发生并产生中断。在校准模式和监控模式下，所需的时钟源都必须初始化和使能，具体的配置过程请参考第 5 章系统控制器。

11.2 CLKTRIM 主要特性

CLKTRIM 支持以下特性：

- 校准模式；
- 监测模式；
- 32 位参考时钟计数器可加载初值；
- 32 位待校准时钟计数器可配置溢出值；
- 6 种参考时钟源；
- 5 种待校准时钟源；
- 支持中断方式。

11.3 CLKTRIM 功能描述

11.3.1 CLKTRIM 校准模式

时钟校准原理

在校准一个频率不精准，但是有参数设置可以调节频率的时钟（待校准时钟 CAL_CLK）时，需要一个精准的时钟作为参考时钟 (REF_CLK)。设定校准时间 Ttrim，待校准时钟频率 Fcal, 参考时钟频率 Fref, 使用待校准时钟和参考时钟同时计数，在校准时间结束时停止计数。读取待校准时钟计数器的值 M，参考时钟计数器的值 N，得到等式：

T_{trim} = M/F_{cal} = N/$_{Fref}$

推导出：F_{cal} = F_{ref} * M / N

由公式判断出，参考时钟的频率误差率越小，待校准时钟的误差率就越小。

计算出待校准时钟频率后，如果误差率超出系统要求的范围，可以调节待校准时钟的参数后，再次重复上面的步骤校准，直到待校准时钟频率的误差率满足系统的要求。

时钟校准模块硬件结构

时间校准模块的待校准时钟有 5 个时钟源，参考时钟有 6 个时钟源，如图 35 所示：

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

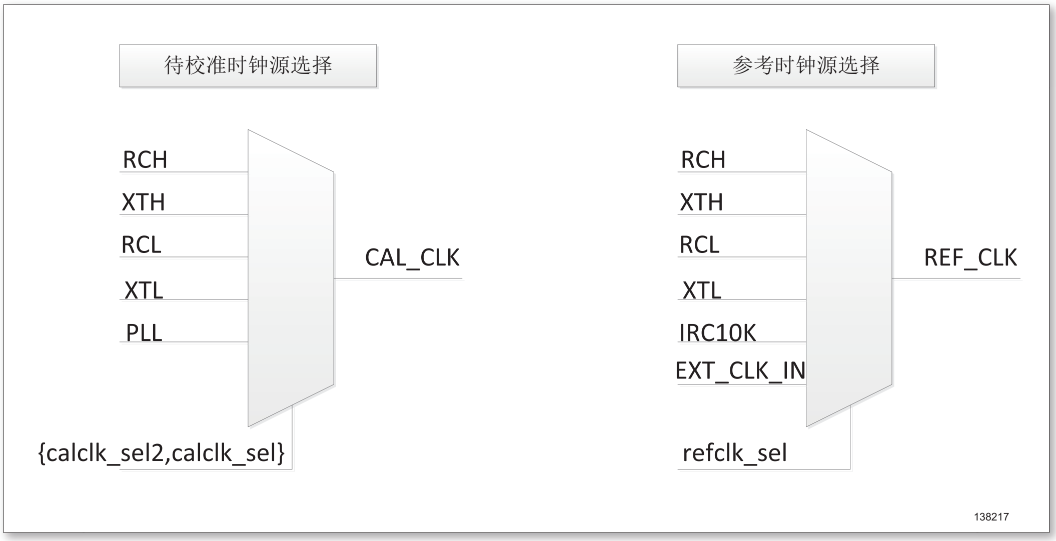

图 35. 主机模式数据/时钟时序

待校准时钟的选择由寄存器 CLKTRIM_CR.calclk_sel2,CLKTRIM_CR.calclk_sel 来配置。

参考时钟的选择由寄存器 CLKTRIM_CR.refclk_sel 来配置。

如图图 36 所示，时钟校准模块有 2 个 32 位计数器，1 个是以参考时钟为时钟，可配置初值的减计数器，初值由寄存器 CLKTRIM_REFCON .rcntval 来配置，计数器值可以从寄存器 CLKTRIM_REFCNT 读出当计数器减计数到 0 时，停止计数并产生中断。

1 个是以待校准时钟为时钟，可配置溢出值的加计数器，溢出值由寄存器 CLKTRIM_CALCON. ccntval 来配置，计数器值可以从寄存器 CLKTRIM_CALCNT 读出当计数器加计数到溢出值时，停止计数并产生中断。

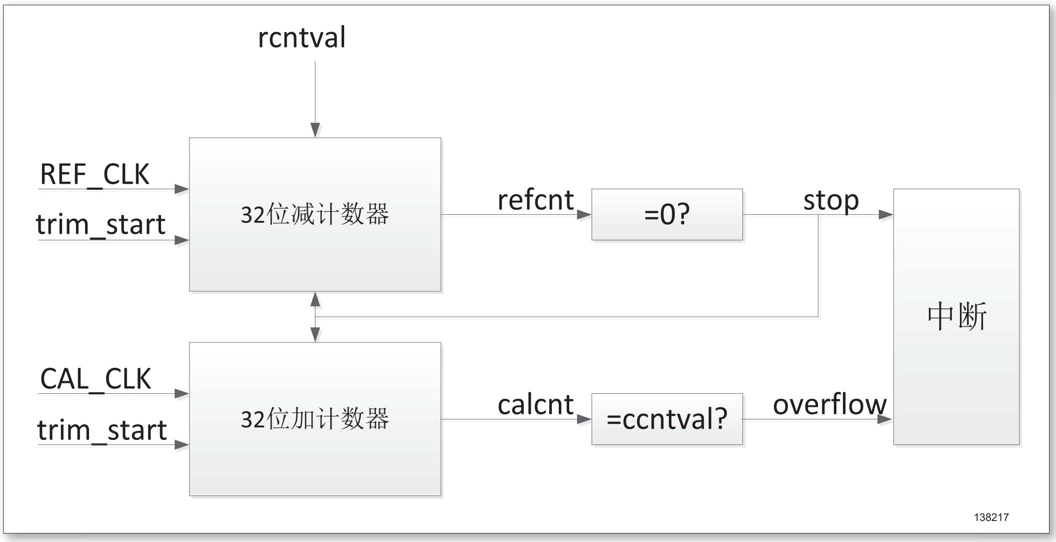

图 36. 时钟校准模块硬件示意图

时钟校准软件流程

1. 设置 CLKTRIM_CR .refclk_sel 寄存器选择参考时钟。
2. 设置 CLKTRIM_CR .calclk_sel 寄存器选择被校准时钟。
3. 设置 CLKTRIM_REFCON .rcntval 寄存器为校准时间。
4. 设置 CLKTRIM_CR .IE 寄存器使能中断。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

5. 设置 CLKTRIM_CR .trim_start 寄存器开始校准。
6. 参考时钟计数器和待校准时钟计数器开始计数。
7. 当参考时钟计数器从初始值减计数到 0 时，CLKTRIM_IFR.stop 置 1，触发中断。
8. 中断服务子程序判断 CLKTRIM_IFR.stop 为 1，读取寄存器 CLKTRIM_REFCNT 和 CLKTRIM_CALCNT 的值。
9. 清零 CLKTRIM_CR .trim_start 寄存器结束校准。
10. 根据时钟校准原理章节提到的公式计算出待校准时钟的频率。
11. 其中 M = 寄存器 CLKTRIM_CALCNT 的值，N = 寄存器 CLKTRIM_REFCON .rcntval 的值; 当待校准时钟的频率不满足误差率要求时，调整待校准时钟参数。

重新执行步骤 5$_{s}im$10，直到待校准时钟满足误差率要求。

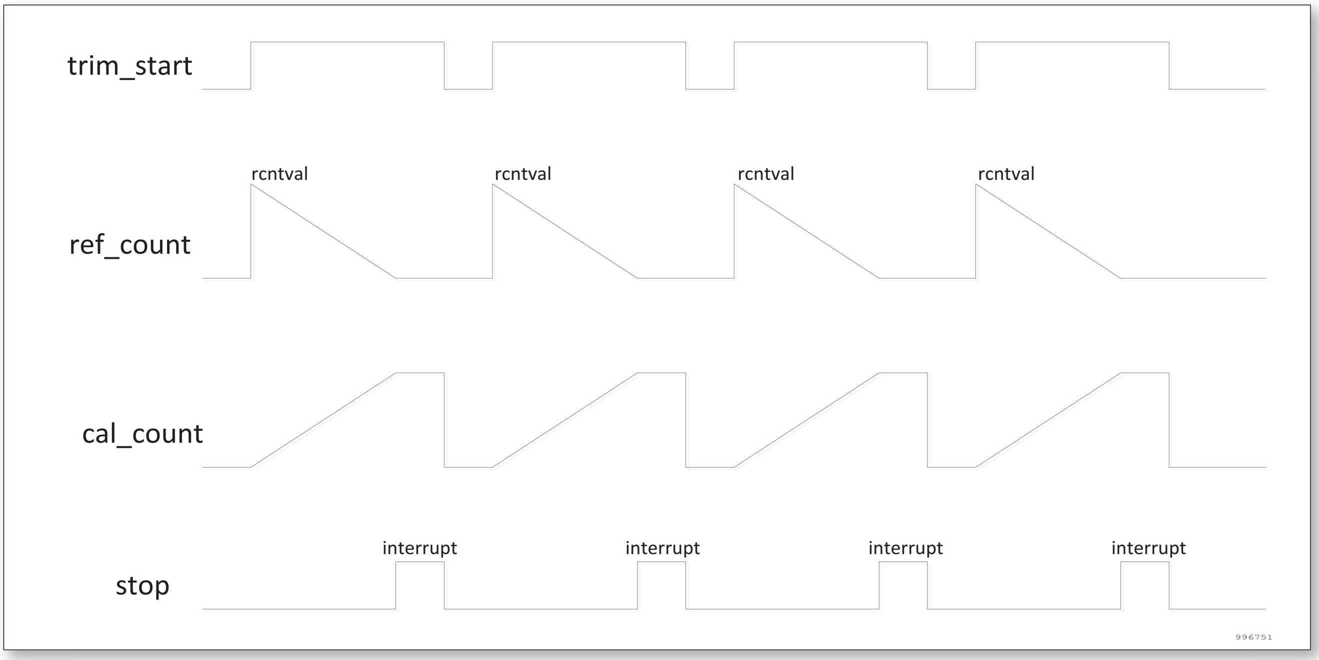

图 37. 时钟校准波形示意图

注：

- 校准模式在校准过程中有可能因为校准时间设置过长，发生待校准时钟计数器在 CLKTRIM_IFR.stop 置 1 之前溢出的情况，CLKTRIM_IFR. calcnt_of 置 1，触发中断。中断服务子程序发现 CLKTRIM_IFR. calcnt_of 置 1 时，清零 CLKTRIM_CR .trim_start 寄存器结束校准。

这种情况下校准是无法正确进行的，必须调整校准时间，重新校准。具体步骤是：

1. 设置 CLKTRIM_REFCON .rcntval 寄存器调整校准时间。
2. 设置 CLKTRIM_CR .trim_start 寄存器重新开始校准。

11.3.2 CLKTRIM 监测模式

时钟监测原理

为了监测系统工作时钟（被监测时钟 CAL_CLK）是否正常工作时，需要一个稳定的时钟作为参考时钟 (REF_CLK)。设定监测时间 Ttrim，被监测时钟频率 Fcal, 参考时钟频率 Fref, 被监测时钟计数器溢出值 CALVAL, 使用被监测时钟和参考时钟同时计数，在监测时间结束时停止计数，判断被监测时钟计数器是否处于溢出状态。如果被监测时钟计数器已溢出，表示在监测时间内被监测时钟工作正常，继续进行下一次监测。如果被监测时钟计数器没有溢出，表示在监测时间内被监测时钟工作异常，可能被监测时钟已经停止或者频率大幅变化，产生被监测时

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

钟工作异常中断，由硬件切换系统工作时钟。

时钟监测硬件结构

时间校准模块的被监测时钟有 2 个时钟源，参考时钟有 6 个时钟源，如图 38所示：

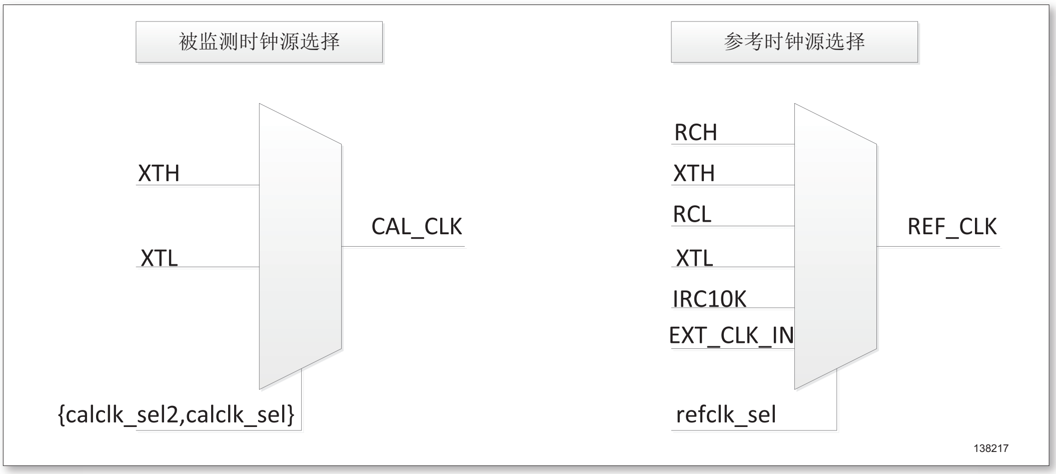

图 38. 时钟校准波形示意图

被监测时钟的选择由寄存器 CLKTRIM_CR.calclk_sel2,CLKTRIM_CR.calclk_sel 来配置，参考时钟的选择由寄存器 CLKTRIM_CR.refclk_sel 来配置。

时钟监测软件流程

1. 设置 CLKTRIM_CR .refclk_sel 寄存器选择参考时钟。
2. 设置 CLKTRIM_CR .calclk_sel 寄存器选择被监控时钟。
3. 设置 CLKTRIM_REFCON .rcntval 寄存器为监控间隔时间。
4. 设置 CLKTRIM_CALCON. ccntval 寄存器为被监控时钟计数器溢出时间。
5. 设置 CLKTRIM_CR .mon_en 寄存器使能监控功能。
6. 设置 CLKTRIM_CR .IE 寄存器使能中断。
7. 设置 CLKTRIM_CR .trim_start 寄存器开始监控。
8. 参考时钟计数器和被监控时钟计数器开始计数。
9. 当参考时钟计数器计数到达监控间隔时间时，CLKTRIM_IFR.stop 置 1，同时判断被监控时钟计数器是否溢出，即 CLKTRIM_IFR. calcnt_of 是否置 1。如果溢出，即 CLKTRIM_IFR. calcnt_of 为 1，表示被监控时钟工作正常。如果没有溢出，即 CLKTRIM_IFR. calcnt_of 为 0，表示被监控时钟失效，CLKTRIM_IFR .xtl_fault/xth_fault 置 1，触发中断。
10. 处理中断服务子程序，清除中断标志位 CLKTRIM_IFR .xtl_fault/xth_fault，清零 CLKTRIM_CR .trim_start 寄存器结束监测。

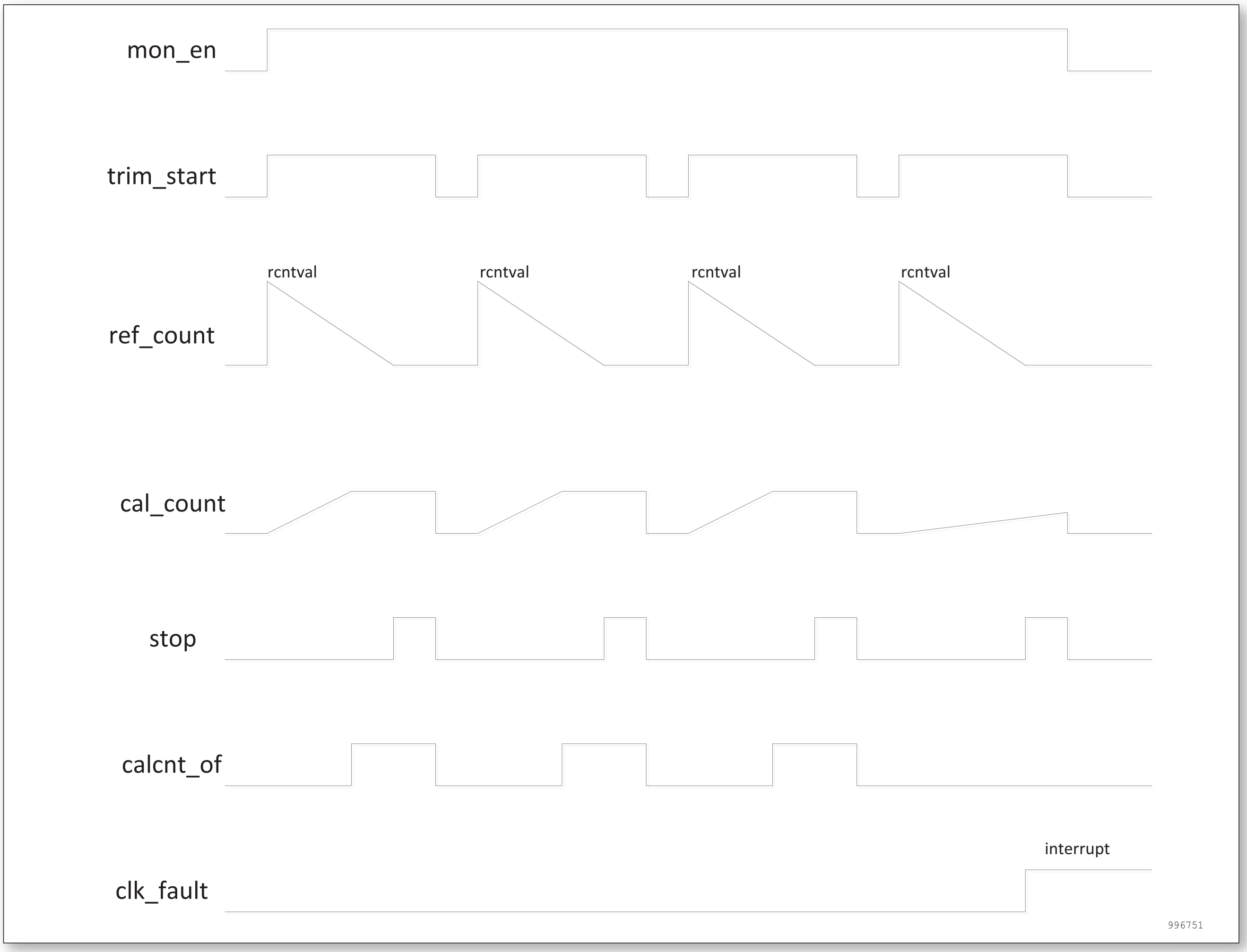

图 39. 时钟监控波形示意图

11.4 CLKTRIM 寄存器描述

表 32. CLKTRIM 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x00	CLKTRIM_CR	配置寄存器	0x0000 0000	小节 11.4.1
0x04	CLKTRIM_REFCON	参考计数器初值配置寄存器	0x0000 0000	小节 11.4.2
0x08	CLKTRIM_REFCNT	参考计数器值寄存器	0x0000 0000	小节 11.4.3
0x0c	CLKTRIM_CALCNT	校准计数器值寄存器	0x0000 0000	小节 11.4.4
0x10	CLKTRIM_IFR	中断标志位寄存器	0x0000 0000	小节 11.4.5
0x14	CLKTRIM_ICLR	中断标志位清除寄存器	0x1111 1111	小节 11.4.6
0x18	CLKTRIM_CALCON	校准计数器溢出值配置寄存器	0xFFFF FFFF	小节 11.4.7

11.4.1 配置寄存器 (CLKTRIM_CR)

起始地址：0x40001800

地址偏移：0x00

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

复位值：0x00000000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							calclk_sel2	IE	mon_en	calclk_sel		refclk_sel			trim_start
							RW	RW	RW	RW		RW			RW

Bit	Field	Type	Reset	Description
31:9	Reserved			保留。
8	calclk_sel2	rw	0x00	待校准/监测时钟选择高位寄存器。
7	IE	rw	0x00	中断使能寄存器。 0 -禁止 1 -使能
6	mon_en	rw	0x00	监测模式使能寄存器。 0 -禁止 1 -使能
5:4	calclk_sel	rw	0x00	待校准/监测时钟选择低位寄存器。 calclk_sel2，calclk_sel 000 RCH 001 XTH 010 RCL 011 XTL 100 PLL 101-111 保留
3:1	refclk_sel	rw	0x00	参考时钟选择寄存器。 000 —- RCH 001 —- XTH 010 —- RCL 011 —- XTL 100 —- IRC10K 101 —- EXT_CLK_IN
0	trim_start	rw	0x00	校准/监测开始寄存器。 0 -停止 1 -开始

11.4.2 参考计数器初值配置寄存器 (CLKTRIM_REFCON)

起始地址：0x40001800

地址偏移：0x04

复位值：0x00000000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

Bit	Field	Type	Reset	Description
31:0	rcntval	rw	0x00	参考计数器初始值。

11.4.3 参考计数器值寄存器 (CLKTRIM_REFCNT)

起始地址：0x40001800

地址偏移：0x08

复位值：0x00000000

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

Bit	Field	Type	Reset	Description
31:0	refcnt	r	0x00	参考计数器值。

11.4.4 校准计数器值寄存器 (CLKTRIM_CALCNT)

起始地址：0x40001800

地址偏移：0x0c

复位值：0x00000000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

Bit	Field	Type	Reset	Description
31:0	calcnt	r	0x00	校准计数器值。

11.4.5 中断标志位寄存器 (CLKTRIM_IFR)

起始地址：0x40001800

地址偏移：0x10

复位值：0x00000000

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

Bit	Field	Type	Reset	Description
31:5	Reserved			保留。
4	pll_fault	r	0x00	PLL 失效标志。 CLKTRIM_ICLR.pll_fault_clr 写零清除此标志位。
3	xth_fault	r	0x00	XTH 失效标志。 CLKTRIM_ICLR.xth_fault_clr 写零清除此标志位。
2	xtl_fault	r	0x00	XTL 失效标志。 CLKTRIM_ICLR.xtl_fault_clr 写零清除此标志位。
1	calcnt_of	r	0x00	校准计数器溢出标志。 CLKTRIM_CR.start 写零清除此标志位。
0	stop	r	0x00	参考计数器停止标志。 CLKTRIM_CR.start 写零清除此标志位。

11.4.6 中断标志位清除寄存器 (CLKTRIM_ICLR)

起始地址：0x40001800

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

地址偏移：0x14

复位值：0x11111111

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved											pll_fault_clr	xth_fault_clr	xtl_fault_clr	Reserved	
											R1W0	R1W0	R1W0		

Bit	Field	Type	Reset	Description
31:5	Reserved			保留。
4	pll_fault_clr	rc_w0	0x01	清除 PLL 失效标志，写零清除。
3	xth_fault_clr	rc_w0	0x00	清除 XTH 失效标志，写零清除。
2	xtl_fault_clr	rc_w0	0x00	清除 XTL 失效标志，写零清除。
1:0	Reserved			保留。

11.4.7 校准计数器溢出值配置寄存器 (CLKTRIM_CALCON)

起始地址：0x40001800

地址偏移：0x18

复位值：0xFFFFFFFF

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

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

Bit	Field	Type	Reset	Description
31:0	ccntval	rw	0xFFFFFFFF	校准计数器溢出值。

12 嵌入式闪存 (FLASH)

12.1 概述

本系统包含一块 256k 字节（Byte）容量的 FLASH 存储器，共划分为 512 个页（Sector），每个页（Sector）的容量为 512 字节（Byte）。FLASH 控制器支持对 FLASH 存储器的擦除、编程以及读操作。本控制器还支持对 FLASH 存储器擦写保护，以及控制寄存器的写保护。

12.2 FLASH 容量划分

表 33: SPI FLASH 容量划分

地址	序号		地址	序号
0x0E00 -0x0FFF	Sector7	……	0x3FE00 -0x3FFFF	Sector511
0x0C00 -0x0DFF	Sector6	……	0x3FC00 -0x3FDFF	Sector510
0x0A00 -0x0BFF	Sector5	……	0x3FA00 -0x3FBFF	Sector509
0x0800 -0x09FF	Sector4	……	0x3F800 -0x3F9FF	Sector508
0x0600 -0x07FF	Sector3	……	0x3F600 -0x3F7FF	Sector507
0x0400 -0x05FF	Sector2	……	0x3F400 -0x3F5FF	Sector506
0x0200 -0x03FF	Sector1	……	0x3F200 -0x3F3FF	Sector505
0x0000 -0x01FF	Sector0	……	0x3F000 -0x3F1FF	Sector504

12.3 功能描述

本控制器支持对 FLASH 的 Byte（8 位）、half-word（16 位）和 word（32 位）三种位宽的数据读写操作。注意，Byte 操作的目标地址必须按字节对齐，half-word 操作的目标地址必须按半字对齐（地址最低位为 1’b0），word 操作的地址必须按字对齐（地址最低两位为 2’b00）。如果目标地址没有按照位宽对齐，则该操作无效，并且 CPU 会进入 Hard Fault 出错中断。

12.3.1 页擦除（Sector Erase）

页擦除每次可以擦除用户指定的一个页（Sector）。擦除操作完成后，页（Sector）内的数据均为 0xFF。如果该擦除操作是从 FLASH 内执行，则 CPU 会停止取指，硬件自动等待该操作完成（FLASH_CR. BUSY 变为 0）；如果该擦除操作是从 RAM 内执行，则 CPU 不会停止取指，用户软件应等待该操作完成（FLASH_CR. BUSY 变为 0）。

Sector 擦除操作步骤如下：

1. 配置 FLASH 擦写参数，详见擦写时序章节。
2. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
3. 配置 FLASH_CR. OP 为 2，设置 Flash 操作模式为 Sector 擦除。
4. 检查 FLASH_CR. OP 是否为 2，如不为 2 则跳转到 Step2。
5. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
6. 设置 FLASH_SLOCK 相应的比特为 1，去除该 Sector 的擦写保护。
7. 检查 FLASH_SLOCK 相应的比特是否为相应的比特是否为 1，如不为 1 则跳转到 Step5。
8. 对待擦除的 Sector 内的任意地址进行写入任意数据，触发 Sector 擦除。

例：*((unsigned char *)0x00000200)= 0x00。

9. 等待 FLASH_CR. BUSY 变为 0，Sector 擦除操作完成。
10. 如需擦除其它 Sector，重复 Step5 -Step9。

12.3.2 全片擦除（Chip Erase）

全片擦除可以一次性擦除全部的页（Sector）。擦除操作完成后，所有页（Sector）内的数据均为 0xFF。如果该擦除操作是从 FLASH 内执行，则该操作会被禁止。因为该操作会擦除当前 PC 所在的程序段。如发生这种情况，出错标志位会被置起；如果该擦除操作是从 RAM 内执行，则 CPU 不会停止取指，用户软件应等待该操作完成（FLASH_CR. BUSY 变为 0）。

Chip 擦除操作步骤如下：

1. 配置 FLASH 擦写参数，详见 12.4 章节。
2. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
3. 配置 FLASH_CR. OP 为 3，设置 Flash 操作模式为 Chip 擦除。
4. 检查 FLASH_CR. OP 是否为 3，如不为 3 则跳转到 Step2。
5. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
6. 设置 FLASH_SLOCK 为 0xFFFF，去除所有 Sector 的擦写保护。
7. 检查 FLASH_SLOCK0 是否为 0xFFFFFFFF，如不为 0xFFFFFFFF 则跳转到 Step5。
8. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
9. 设置 FLASH_SLOCK1 为 0xFFFFFFFF，去除所有 Sector 的擦写保护。
10. 检查 FLASH_SLOCK1 是否为 0xFFFFFFFF，如不为 0xFFFFFFFF 则跳转到 Step8。
11. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
12. 设置 FLASH_SLOCK2 为 0xFFFFFFFF，去除所有 Sector 的擦写保护。
13. 检查 FLASH_SLOCK2 是否为 0xFFFFFFFF，如不为 0xFFFFFFFF 则跳转到 Step11。
14. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
15. 设置 FLASH_SLOCK3 为 0xFFFFFFFF，去除所有 Sector 的擦写保护。
16. 检查 FLASH_SLOCK3 是否为 0xFFFFFFFF，如不为 0xFFFFFFFF 则跳转到 Step14。
17. 对待擦除的 Chip 内的任意地址进行写操作，触发 Chip 擦除。
    例：*((unsigned char *)0x00000000)= 0x00。
18. 等待 FLASH_CR. BUSY 变为 0，Chip 擦除操作完成。

12.3.3 写操作（Program）

写操作只能将 FLASH 内的比特数据由 1 写成 0，故写入数据前应确保待写入的地址内的数据为 0xFF。支持写入 3 种数据长度：Byte（8bits）、Half-word（16bits）、Word（32bits）, 写入的数据以小端模式存放在 FLASH 中，即低地址存放数据的低字节。如果写操作是从 FLASH 内执行，则 CPU 会停止取指，硬件自动等待该操作完成（FLASH_CR. BUSY 变为 0）；如果该写操作是从 RAM 内执行，则 CPU 不会停止取指，用户软件应等待该操作完成（FLASH_CR. BUSY 变为 0）。

Byte 写操作步骤如下：

1. 配置 FLASH 擦写参数，详见 12.4 章节。
2. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
3. 配置 FLASH_CR. OP 为 1，设置 Flash 操作模式为写入。
4. 检查 FLASH_CR. OP 是否为 1，如不为 1 则跳转到 Step2。

5. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
6. 设置 FLASH_SLOCK 相应的比特为 1，去除擦写保护。
7. 检查 FLASH_SLOCK 相应的比特是否为相应的比特是否为 1，如不为 1 则跳转到 Step5。
8. 对待写入的目标地址进行 Byte 写操作，触发写入操作。
   例：*((unsigned char *)0x00001231)= 0x5A。
9. 等待 FLASH_CR. BUSY 变为 0，写入操作完成。
10. 如需写 Byte 到其它地址，重复 Step8 -Step9。

Half-word 写操作步骤如下：

1. 配置 FLASH 擦写时间，详见 12.4 章节。
2. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
3. 配置 FLASH_CR. OP 为 1，设置 Flash 操作模式为写入。
4. 检查 FLASH_CR. OP 是否为 1，如不为 1 则跳转到 Step2。
5. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
6. 设置 FLASH_SLOCK 相应的比特为 1，去除擦写保护。
7. 检查 FLASH_SLOCK 相应的比特是否为相应的比特是否为 1，如不为 1 则跳转到 Step5。
8. 对待写入的目标地址进行 Half-word 写操作，触发写入操作。
   例：*((unsigned char *)0x00001232)= 0xABCD。
9. 等待 FLASH_CR. BUSY 变为 0，写入操作完成。
10. 如需写 Half-word 到其它地址，重复 Step8 - Step9。

Word 写操作步骤如下：

1. 配置 FLASH 擦写参数，详见 12.4 章节。
2. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
3. 配置 FLASH_CR. OP 为 1，设置 Flash 操作模式为写入。
4. 检查 FLASH_CR. OP 是否为 1，如不为 1 则跳转到 Step2。
5. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
6. 设置 FLASH_SLOCK 相应的比特为 1，去除擦写保护。
7. 检查 FLASH_SLOCK 相应的比特是否为相应的比特是否为 1，如不为 1 则跳转到 Step5。
8. 对待写入的目标地址进行 Word 写操作，触发写入操作。
   例：*((unsigned char *)0x00001234)= 0x55667788。
9. 等待 FLASH_CR. BUSY 变为 0，写入操作完成。
10. 如需写 Word 到其它地址，重复 Step8 - Step9。

12.3.4 读操作（Read）

支持读出 3 种数据长度：Byte（8bits）、Half-word（16bits）、Word（32bits）, 读出的数据为小端模式，即低地址存放数据的低字节。读操作无需操作步骤，任何时刻都可以读出 FLASH 内的数据。

例：

Byte 读操作：	temp = *((unsigned char *)0x00001231)
Half-word 读操作	temp = *((unsigned short int *)0x00001232)
Word 读操作	temp = *((nsigned int *)0x00001234)

12.4 擦写时间

FLASH 存储器对擦除和编程操作的控制信号具有严格的时间要求，控制信号的时序不合格会造成擦除操作和编程操作失败。上电时默认装载了 HCLK 为 4MHz 时的擦写参数；如果对 Flash 进行擦写时的 HCLK 频率不为 4MHz，则用户程序应装载 HCLK 频率所对应的擦写参数。对 FLASH 进行操作时，支持的最高 HCLK 为 48MHz。

擦写时序参数相关的寄存器为：FLASH_TMRCV、FLASH_TPGS、FLASH_TPROG、FLASH_TSERASE、FLASH_TMERASE 、FLASH_TPRCV 、FLASH_TSRCV 、FLASH_TNVS。如果 HCLK 由默认 4MHz 升高为 8MHz，则上述 FLASH_Tx 寄存器的值应该设为默认值的 2 倍，即保持当前 Tsysclk*FLASH_Tx 的结果与默认值相等即可。

以下表格为不同频率下相应的 FLASH 擦写时间参数：

表 34: 不同频率下 FLASH 擦写时间参数

Parameter	4M	8M	16M	24M	32M	48M
TNVS	0x20	0x40	0x80	0xC0	0x100	0x180
TPGS	0x17	0x2E	0x5C	0x8A	0xB8	0x114
TPROG	0x1B	0x36	0x6C	0xA2	0xD8	0x144
TSERASE	0x4650	0x8CA0	0x11940	0x1A5E0	0x23280	0x34BC0
TMERASE	0x222E0	0x445C0	0x88B80	0xCD140	0x111700	0x19A280
TPRCV	0x18	0x30	0x60	0x90	0xC0	0x120
TSRCV	0xF0	0x1E0	0x3C0	0x5A0	0x780	0xB40
TMRCV	0x3E8	0x7D0	0xFA0	0x1770	0x1F40	0x2EE0

配置系统频率为 8MHz 时擦写参数的操作步骤如下：

1. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
2. 向 FLASH_TNVS 寄存器写入 0x40，如读出的该寄器值不为 0x40，则跳转到上一步。
3. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
4. 向 FLASH_TPGS 寄存器写入 0x2E，如读出的该寄器值不为 0x2E，则跳转到上一步。
5. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
6. 向 FLASH_TPROG 寄存器写入 0x36，如读出的该寄器值不为 0x36，则跳转到上一步。
7. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
8. 向 FLASH_TSERASE 寄存器写入 0x8CA0，如读出的该寄器值不为 0x8CA0，则跳转到上一步。
9. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
10. 向 FLASH_TMERASE 寄存器写入 0x445C0，如读出的该寄器值不为 0x445C0，则跳转到上一步。
11. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
12. 向 FLASH_TPRCV 寄存器写入 0x30，如读出的该寄器值不为 0x30，则跳转到上一步。
13. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
14. 向 FLASH_TSRCV 寄存器写入 0x1E0，如读出的该寄器值不为 0x1E0，则跳转到上一步。
15. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
16. 向 FLASH_TMRCV 寄存器写入 0x7D0，如读出的该寄器值不为 0x7D0，则跳转到上一步。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

12.5 读等待周期

本设备内置的 FLASH 支持的最快取指频率为 24MHz。当 HCLK 频率超过 24MHz 小于 48MHz 时，必须为 FLASH 的读取时间插入等待周期，即设置 FLASH_CR.WAIT 为 1。当插入等待周期时，FLASH 每两个周期才会完成一次读取操作。

12.6 擦写保护

12.6.1 擦写保护位

整个 256k 字节 FLASH 存储器被划分为 512 个页，每 4 个页共用一个擦写保护位。当页被保护时，对该页进行的擦写操作均无效并产生报警标志位和中断信号。当 FLASH 存储器中的任意页被保护时，对该 FLASH 的全片擦写无效，并产生报警标志位和中断信号。

12.6.2 PC 地址擦写保护

CPU 在 FLASH 中运行程序时，如果当前 PC 指针正好落在待擦写的页地址范围之内，那么该擦写操作无效并产生报警标志位和中断信号。

12.7 寄存器写保护

本模块的重要控制器屏蔽普通的写操作，必须用写序列方式才能修改。

需要通过写序列方式才能更改的寄存器如下所示：

FLASH_TNVS、FLASH_TPGS、FLASH_TPROG、FLASH_TSERASE、FLASH_CR、FLASH_TPRCV、FLASH_TSRCV、FLASH_TMRCV、FLASH_TSERASE、FLASH_SLOCK0、FLASH_SLOCK1、FLASH_SLOCK2、FLASH_SLOCK3

不需要通过写序列方式即可更改的寄存器如下所示：

FLASH_ICLR、FLASH_BYPASS。

通过写序列方式修改寄存器值的具体操作步骤如下所示：

1. 向 FLASH_BYPASS 寄存器写入 0x5A5A；
2. 向 FLASH_BYPASS 寄存器写入 0xA5A5；
3. 对待修改的寄存器写入目标值；
4. 验证待修改的寄存器当前值是否与目标相同，如不则跳转到 Step1；
5. 执行其他操作。

注：写 0x5a5a、0xa5a5、写目标寄存器，这三步操作之间不可插入本模块内的任何寄存器写操作，否则该 Bypass 序列会失效，需要重新写入 0x5a5a-0xa5a5 序列。

12.8 Flash 寄存器描述

表 35. FLASH 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x00	FLASH_TNVS	Tnvs 时间参数	0x0000 0020	小节 12.8.1
0x04	FLASH_TPGS	Tpgs 时间参数	0x0000 0017	小节 12.8.2
0x08	FLASH_TPROG	Tprog 时间参数	0x0000 001B	小节 12.8.3

Offset	Acronym	Register Name	Reset	Section
0x0C	FLASH_TSERASE	Tserase 时间参数	0x0000 4650	小节 12.8.4
0x10	FLASH_TMERASE	Tmerase 时间参数	0x0002 22E0	小节 12.8.5
0x14	FLASH_TPRCV	Tprcv 时间参数	0x0000 0018	小节 12.8.6
0x18	FLASH_TSRCV	Tsrcv 时间参数	0x0000 00F0	小节 12.8.7
0x1C	FLASH_TMRCV	Tmrcv 时间参数	0x0000 03E8	小节 12.8.8
0x20	FLASH_CR	控制寄存器	0x0000 0000	小节 12.8.9
0x24	FLASH_IFR	中断标志寄存器	0x0000 0000	小节 12.8.10
0x28	FLASH_ICLR	中断标志清除寄存器	0x0000 000F	小节 12.8.11
0x2C	FLASH_BYPASS	Bypass 序列寄存器	0x0000 0000	小节 12.8.12
0x30	FLASH_SLOCK0	Sector0-127 擦写保护寄存器	0x0000 0000	小节 12.8.13
0x34	FLASH_SLOCK1	Sector128-255 擦写保护寄存器	0x0000 0000	小节 12.8.14
0x40	FLASH_SLOCK2	Sector256-383 擦写保护寄存器	0x0000 0000	小节 12.8.15
0x44	FLASH_SLOCK3	Sector384-512 擦写保护寄存器	0x0000 0000	小节 12.8.16

12.8.1 Tnvs 时间参数 (FLASH_TNVS)

起始地址：0x40001800

地址偏移：0x00

复位值：0x00000020

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="7" rowspan="2">Reserved</td><td colspan="9">TNVS</td></tr>
<tr><td colspan="9">RW</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:9	Reserved			保留。
8: 0	TNVS	rw	0x20	计算公式：TNVS = 8*HCLK，HCLK 的单位为 MHz。 4MHz 示例：TNVS = 8*4 = 32。

12.8.2 Tpgs 时间参数 (FLASH_TPGS)

起始地址：0x40001800

地址偏移：0x04

复位值：0x00000017

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

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

Bit	Field	Type	Reset	Description
31:9	Reserved			保留。
8: 0	TPGS	rw	0x17	计算公式：TPGS = 5.75*HCLK，HCLK 的单位为 MHz。 4MHz 示例：TPGS = 5.75*4 = 23。

12.8.3 Tprog 时间参数 (FLASH_TPROG)

起始地址：0x40001800

地址偏移：0x08

复位值：0x0000001B

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

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

Bit	Field	Type	Reset	Description
31:9	Reserved			保留。
8: 0	TPROG	rw	0x1B	计算公式：TPROG = 6.75*HCLK，HCLK 的单位为 MHz。 4MHz 示例：TPROG = 6.75*4 = 27。

12.8.4 Tserase 时间参数 (FLASH_TSERASE)

起始地址：0x40001800

地址偏移：0x0C

复位值：0x00004650

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

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

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31:18	Reserved			保留。
17: 0	TSERASE	rw	0x4650	计算公式：TSERASE = 4500*HCLK，HCLK 的单位为 MHz。 4MHz 示例：TSERASE = 4500*4 = 27。

12.8.5 Tmerase 时间参数 (FLASH_TMERASE)

起始地址：0x40001800

地址偏移：0x10

复位值：0x000222E0

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

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

Bit	Field	Type	Reset	Description
31:21	Reserved			保留。
20: 0	TMERASE	rw	0x222E0	计算公式：TMERASE = 35000*HCLK，HCLK 的单位为 MHz。 4MHz 示例：TMERASE = 35000*4 = 140000。

12.8.6 Tprcv 时间参数 (FLASH_TPRCV)

起始地址：0x40001800

地址偏移：0x14

复位值：0x00000018

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

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

Bit	Field	Type	Reset	Description
31:12	Reserved			保留。
11: 0	TPRCV	rw	0x18	计算公式：TPRCV = 6*HCLK，HCLK 的单位为 MHz。 4MHz 示例：TPRCV = 6*4 = 24。

ZLG
©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

12.8.7 Tsrcv 时间参数 (FLASH_TSRCV)

起始地址：0x40001800

地址偏移：0x18

复位值：0x000000F0

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

Bit	Field	Type	Reset	Description
31:12	Reserved			保留。
11: 0	TSRCV	rw	0xF0	计算公式：TSRCV = 60*HCLK，HCLK 的单位为 MHz。 4MHz 示例：TSRCV = 60*4 = 240。

12.8.8 Tmrcv 时间参数 (FLASH_TMRCV)

起始地址：0x40001800

地址偏移：0x1C

复位值：0x000003E8

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

Bit	Field	Type	Reset	Description
31:14	Reserved			保留。
13: 0	TMRCV	rw	0x3E8	计算公式：TMRCV = 250*HCLK，HCLK 的单位为 MHz。 4MHz 示例：TMRCV = 250*4 = 1000。

12.8.9 控制寄存器 (FLASH_CR)

起始地址：0x40001800

地址偏移：0x20

复位值：0x00000000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						DPSTB_EN	Reserved		IE		BUSY	WAIT		OP	
						RW			RW		RO	RW		RW	

Bit	Field	Type	Reset	Description
31:10	Reserved			保留。
9	DPSTB_EN	rw	0x00	FLASH dpstb 使能 Mask 位； 0：当系统进入 deepsleep 模式，FLASH 不进入低功耗模式； 1：当系统进入 deepsleep 模式，FLASH 进入低功耗模式；
8:7	Reserved			保留。
6:5	IE	rw	0x00	IE[6]：FLASH 擦写被保护地址中断使能；0：不使能；1：使能 IE[5]：FLASH 擦写 PC 值中断使能；0：不使能；1：使能
4	BUSY	r	0x00	空闲/忙标志位；0：空闲状态；1：忙状态；
3:2	WAIT	rw	0x00	读 FLASH 周期； 0~24MHz: 00/11，1 个周期； 24~48MHz：01：2 个周期； 48~72MHz：10:3 个周期；（本系列产品最高时钟为 48MHz）
1:0	OP	rw	0x00	FLASH 操作； 00：读（read）；01：写（program）； 10：页擦除（sector erase）；11：全片擦除（chip erase）

12.8.10 中断标志寄存器 (FLASH_IFR)

起始地址：0x40001800

地址偏移：0x24

复位值：0x00000000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved														IF1	IF0
														RO	RO

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31:2	Reserved			保留。
1	IF1	r	0x00	擦写保护报警中断标志位。
0	IF0	r	0x00	擦写 PC 地址报警中断标志位。

12.8.11 中断标志清除寄存器 (FLASH_ICLR)

起始地址：0x40001800

地址偏移：0x28

复位值：0x0000000F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved														ICLR1	ICLR0
														R1W0	R1W0

Bit	Field	Type	Reset	Description
31:4	Reserved			保留。
3:2	Reserved			写无效，读为 0x3。
1	ICLR1	rc_w0	0x01	清除保护报警中断标志位；写 0 清除；写 1 无效；
0	ICLR0	rc_w0	0x01	清除 PC 地址报警中断标志位；写 0 清除；写 1 无效；

12.8.12 Bypass 序列寄存器 (FLASH_BYPASS)

起始地址：0x40001800

地址偏移：0x2C

复位值：0x00000000

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。
15:0	BYSEQ	w	0x00	在修改本模块寄存器之前，必须对 BYSEQ[15:0] 寄存器写入 0x5a5a-0xa5a5 序列。每次写入该 Bypass 序列后，只可以修改一次寄存器。如需再次修改寄存器，必须再次输入 0x5a5a-0xa5a5 序列。

12.8.13　擦写保护寄存器 (FLASH_SLOCK0)

起始地址：0x40001800

地址偏移：0x30

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SLOCK0[31:16]															
RW															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
SLOCK0[15:0]															
RW															

Bit	Field	Type	Reset	Description
31:0	SLOCK0	rw	0x00	Sector 擦写保护位；0：不允许擦写；1：允许擦写 Bit [0] 对应：Sector0-1-2-3 Bit [1] 对应：Sector4-5-6-7 Bit [2] 对应：Sector8-9-10-11 Bit [3] 对应：Sector12-13-14-15 …… Bit [31] 对应：Sector124-125-126-127

12.8.14　擦写保护寄存器 (FLASH_SLOCK1)

起始地址：0x40001800

地址偏移：0x34

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SLOCK1[31:16]															
RW															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
SLOCK1[15:0]															
RW															

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31:0	SLOCK1	rw	0x00	Sector 擦写保护位；0：不允许擦写；1：允许擦写 Bit [0] 对应：Sector128-129-130-131 Bit [1] 对应：Sector132-133-134-135 Bit [2] 对应：Sector136-137-138-139 Bit [3] 对应：Sector140-141-142-143 …… Bit [31] 对应：Sector251-252-253-254

12.8.15 擦写保护寄存器 (FLASH_SLOCK2)

起始地址：0x40001800

地址偏移：0x40

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SLOCK2[31:16]															
RW															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLOCK2[15:0]															
RW															

Bit	Field	Type	Reset	Description
31:0	SLOCK2	rw	0x00	Sector 擦写保护位；0：不允许擦写；1：允许擦写 Bit[0] 对应：Sector256-257-258-259 Bit [1] 对应：Sector260-261-262-263 Bit [2] 对应：Sector264-265-266-267 Bit [3] 对应：Sector268-269-270-271 …… Bit [31] 对应：Sector380-381-382-383

12.8.16 擦写保护寄存器 (FLASH_SLOCK3)

起始地址：0x40001800

地址偏移：0x44

复位值：0x00000000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SLOCK3[31:16]															
RW															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLOCK3[15:0]															
RW															

Bit	Field	Type	Reset	Description
31:0	SLOCK3	rw	0x00	Sector 擦写保护位；0：不允许擦写；1：允许擦写 Bit[0] 对应：Sector384-385-386-387 Bit [1] 对应：Sector388-389-390-391 Bit [2] 对应：Sector392-393-394-395 Bit [3] 对应：Sector396-397-398-399 …… Bit [31] 对应：Sector508-509-510-511

13 RAM 控制器 (RAM)

13.1 概述

本系统中包含一块容量为 32k 字节（Byte）的 SRAM，支持字节（8 位）、半字（16 位 bits）、字（32 位）三种读写操作。可在系统时钟频率下进行读写操作，无须等待周期。此外，本控制器还支持奇偶校验，可对每个字节（Byte）的 SRAM 数据进行奇偶校验，并产生奇偶校验出错中断。

13.2 功能描述

13.2.1 RAM 地址范围

RAM 在系统映射中的地址范围如表 36所示：

表 36: RAM 地址映射

地址范围	大小	Memory 类型
0x2000_0000 -0x2000_7FFF	32kByte	SRAM

13.2.2 读写位宽

本控制器支持字节（8 位）、半字（16 位）、字（32 位）三种位宽的读写操作。字节操作的地址必须按字节对齐，半字操作的目标地址必须按半字对齐（地址最低位为 1’b0），字操作的地址必须按字对齐（地址最低两位为 2’b00）。如果读写操作的目标地址没有按照位宽规定对齐，该操作无效，并且系统会产生 Hard Fault 出错中断。

13.2.3 奇偶校验

本控制器支持 SRAM 数据的奇偶校验。当对 SRAM 写数据时，对每个字节的数据做奇偶校验，并把 1bit 校验值和 8bits 数据一起存入 SRAM 中。当对 SRAM 读数据时，控制器会读取 8bits 数据和 1bit 校验值，并做奇偶校验，如果校验出错，则置位奇偶校验出错标志位，在中断使能情况下，会产生出错中断。

注：在奇偶校验使能时，在读取 SRAM 数据前，必须初始化 SRAM，否则可能会误触发奇偶校验报警标志位或者中断。

13.3 RAM 寄存器描述

表 37. RAM 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x00	RAM_CR	控制寄存器	0x0000 0000	小节 13.3.1
0x04	RAM_ERRADDR	出错地址寄存器	0x0000 0000	小节 13.3.2
0x08	RAM_IFR	出错中断标志寄存器	0x0000 0000	小节 13.3.3
0x0C	RAM_ICLR	出错中断标志清除寄存器	0x0000 0000	小节 13.3.4

13.3.1 控制寄存器 (RAM_CR)

起始地址：0x40020400

地址偏移：0x00

复位值：0x00000000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

Bit	Field	Type	Reset	Description
31:2	Reserved			保留。
1	IE	rw	0x00	出错报警中断使能信号；1：使能报警中断，0：关闭报警中断；
0	Reserved			保留。

13.3.2 出错地址寄存器 (RAM_ERRADDR)

起始地址：0x40020400

地址偏移：0x04

复位值：0x00000000

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

Bit	Field	Type	Reset	Description
31:15	Reserved			保留。
14:0	ERRADDR	r	0x00	奇偶校验出错 Byte 地址；中断标志位清除后，该地址同时被清除。

13.3.3 出错中断标志寄存器 (RAM_IFR)

起始地址：0x40020400

地址偏移：0x08

复位值：0x00000000

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

Bit	Field	Type	Reset	Description
31:1	Reserved			保留。
0	ERR	r	0x00	奇偶校验出错标志位。

13.3.4 出错中断标志清除寄存器 (RAM_ICLR)

起始地址：0x40020400

地址偏移：0x0C

复位值：0x00000001

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved															ERR CLR
															R1W0

Bit	Field	Type	Reset	Description
31:1	Reserved			保留。
0	ERRCLR	rc_w0	0x01	出错中断标志清除位；写 1：无效，写 0：清零

14 DMA 控制器 (DMAC)

14.1 概述

DMAC（直接内存访问控制器）功能块可以不通过 CPU 高速传输数据。使用 DMAC 能提高系统性能。

DMAC 概述：

- DMAC 配有独立的总线，所以即便是在使用 CPU 总线的同时，DMAC 也可进行传输操作。
- 由 2 条通道组成，能执行 2 种相互独立的 DMA 传输。
- 可设置传输目标地址、传输源地址、传输数据大小、传输请求源以及传输模式，并能控制各通道的传输操作启动、传输的强行终止以及传输的暂停。
- 可控制所有通道批量传输的启动、强行终止及暂停。
- 多通道同时操作时，可用固定方法或循环方法选择操作通道的优先级。
- 支持使用外设中断信号的硬件 DMA 传输。
- 遵从系统总线 (AHB)，支持 32 位地址空间 (4GB)。

各通道功能概述：

- 可递增或固定传输源和传输目标地址；
- 具有传输源和传输目标地址重装功能（即传输完成后回到原设置值的功能）；
- 传输数据的大小可以从以下三种规格中选择：
  - 传输数据宽度：（从字节/半字/字中选择）；
  - 设置块数：（从 1 至 16 中选择）；
  - 设置传输长度：（从 1 至 65536 中选择）。
- 可指定是否发出传输成功完成和传输出错中断；
- 可选择以下四种传输模式：
  - 软件-Block 传输
  - 软件-Burst 传输
  - 硬件-Block 传输
  - 硬件-Burst 传输

传输模式概述：

软件传输是通过 CPU 直接指令启动 DMAC 的方法。

硬件传输是外设发出传输请求时将外设的请求信号作为 DMAC 的传输请求信号以直接启动 DMAC 的方法。外设请求可来自于 UART0/1，LPUART0/1，SPI1，TIM0/1/2/3/4/5/6，ADC，LCD。当这些外设有数据传输请求时，数据可以通过当前通道预先的传输配置来实现传输，而不用 CPU 介入。

14.2 功能框图

DMA 功能框图详见图 40

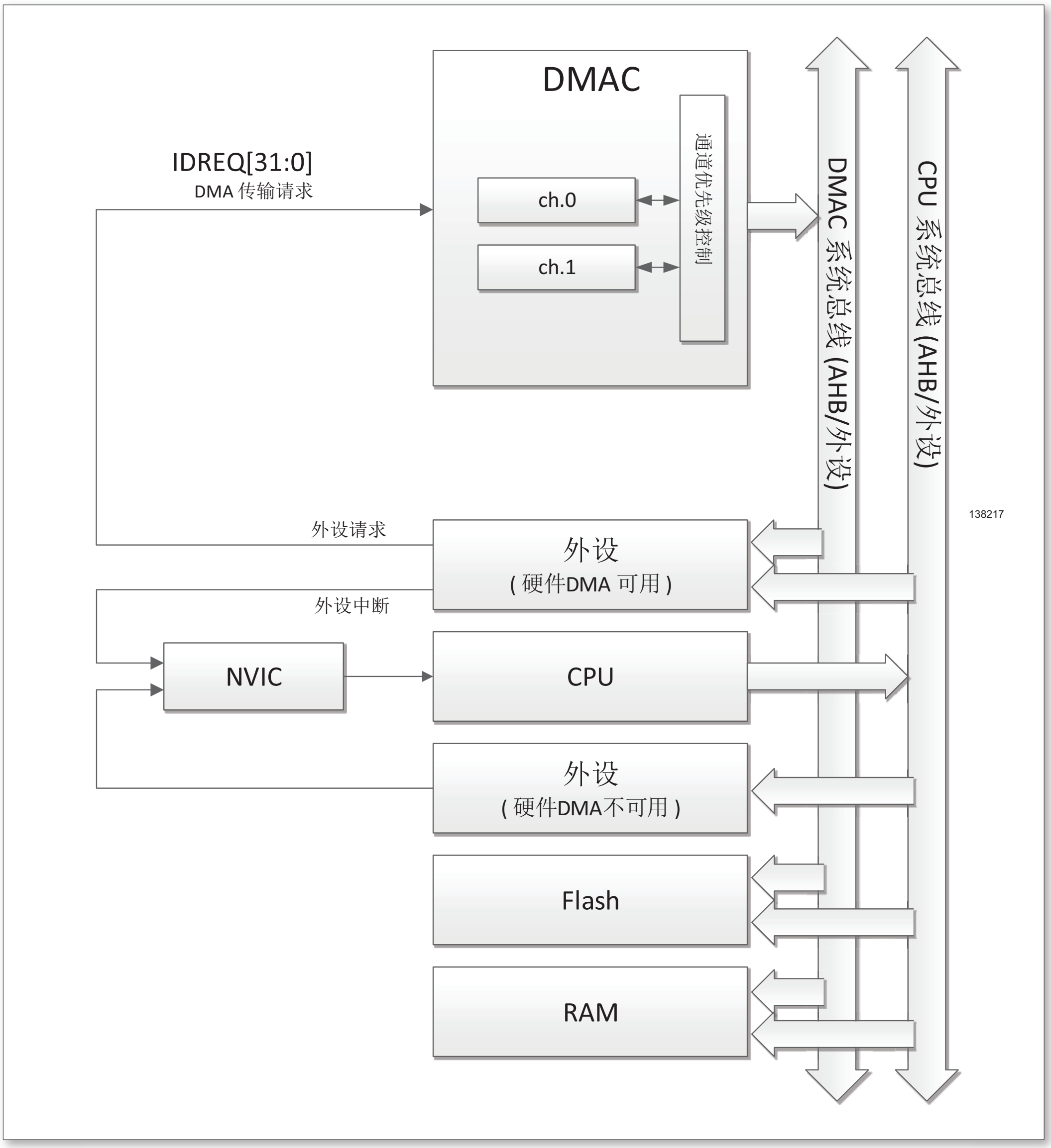

图 40. 功能框图

框图说明

- DMAC

  DMAC 最多配置 2 条通道。各通道执行独立传输。通道之间发生冲突时，优先级控制器控制这些通道的传输操作。

- 系统连接

  DMAC 通过系统总线连接至 CPU、FLASH、RAM 以及各种外设。DMAC 自身有独立于 CPU 总线的总线，可在访问 CPU 总线时执行传输操作。通过指定各通道传输目标地址和传输源地址以进行存储器与外设之间传输数据，系统中的任何地址区域均可访问。由于部分区域不能从 DMAC 访问，需检查内存映射。

- 硬件传输请求连接

  被中断控制器（指图 40 中的 DRQSEL）所选择支持硬件传输的外设中断信号，可以作为发送至 CPU 的中断信号或者同时作为发送至 DMAC 的 DMA 传输请求信号。

  根据硬件请求执行 DMA 传输时，通过设置 DRQSEL，连接各外设的中断信号，作为提前发给 DMAC

的传输请求信号。不支持硬件传输的外设中断信号不能作为 DMA 传输请求信号。

输入 DMAC 的 DMA 传输请求信号有 40 个。硬件传输时，DMAC 各通道从上述 40 条传输请求信号中选择 1 条信号执行操作。使用 CONFA: TRI_SEL 寄存器进行选择。详见表 38：

表 38: 传输请求和对应外设请求信号

IDREQ 编号	对应外设传输请求	IDREQ 编号	对应外设传输请求
0	/	18	TIM0 CHA
1	/	19	TIM0 CHB
2	SPI1 RX	20	TIM1 CHA
3	SPI1 TX	21	TIM1 CHB
4	ADC JRQ0 RX	22	TIM2 CHA
5	ADC SRQ0 RX	23	TIM2 CHB
6	LCD TX	24	TIM3 CHA
7	/	25	TIM3 CHB
8	UART0 RX	26	TIM4 CHA
9	UART0 TX	27	TIM4 CHB
10	UART1 RX	28	TIM5 CHA
11	UART1 TX	29	TIM5 CHB
12	LPUART0 RX	30	TIM6 CHA
13	LPUART0 TX	31	TIM6 CHB
14	LPUART1 RX	32	UART4 RX
15	LPUART1 TX	33	UART4 TX
16	DAC0	34	UART5 RX
17	/	35	UART5 TX

- DMAC 中断

  尽管图 40 未作说明，用于传输完成通知的中断信号连接至 NVIC。各通道有 8 个中断源，合并成 1 个输出。

14.3 基本功能

本节说明各传输模式下 DMAC 的操作。为了加速各模块间的数据搬运，PCLK 要与 HCLK 同频率下，DMA 才能发挥性能优势。

14.3.1 软件-块（Block）传输

图 41 所示为软件-块传输操作示例。示例中应用以下设置：

- 传输模式：软件请求块传输 (ST=1, TRI_SEL[5:0]=000000, MODE=00)；
- 传输源启动地址：SA(SRC_ADR=SA)；
- 传输源地址控制：可递增和重装 (FS=0, RS=1)；
- 传输目标启动地址：DA(DST_ADR=DA)；
- 传输目标地址控制：不可递增和重装 (FD=0, RD=0)；
- 传输数据大小：半字（16 位），块数 = 2，传输长度 = 3 (WIDTH=01, BC=1, TC=2)；

- BC/TC 重装：可重装 (RC=1)。

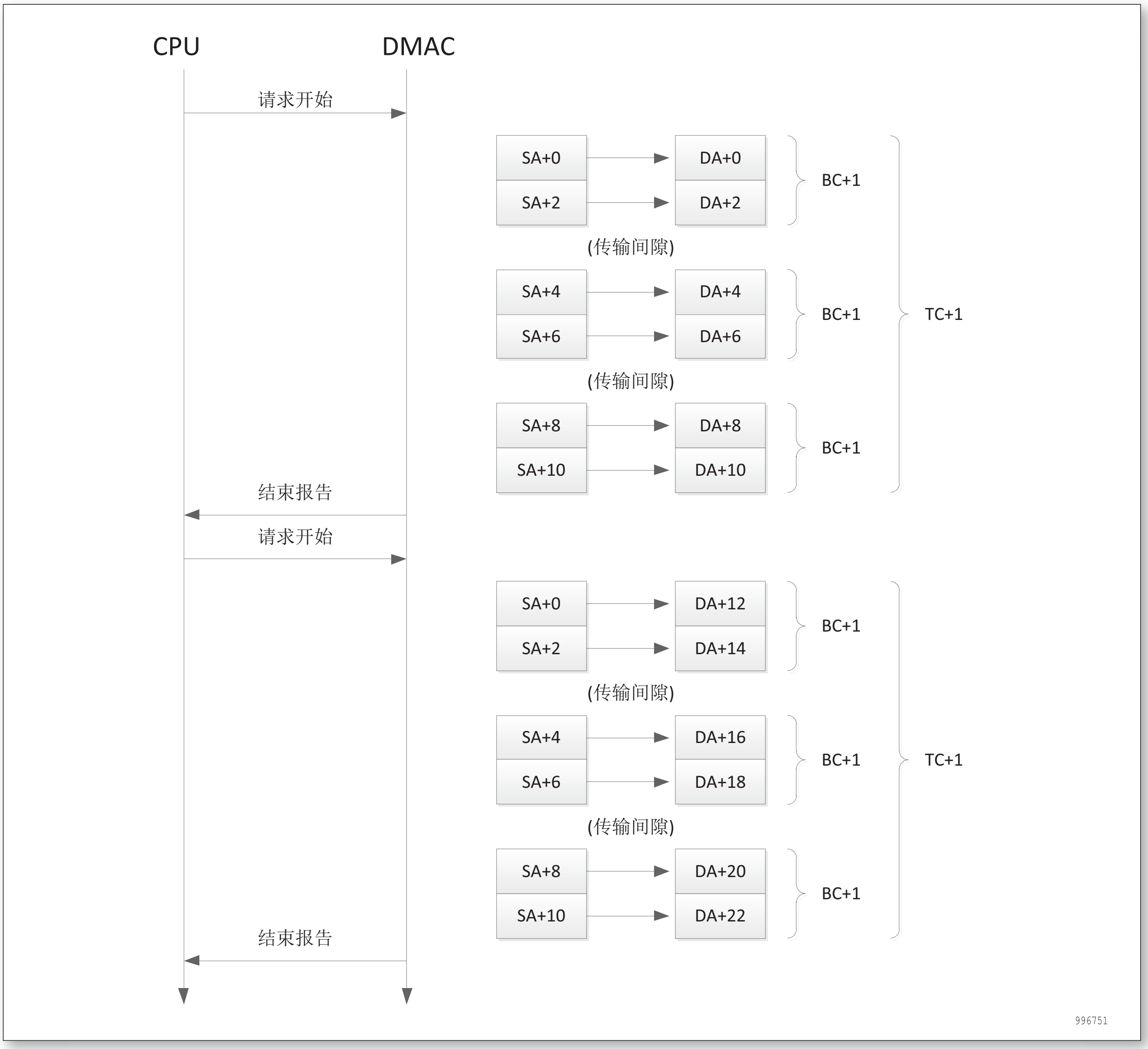

图 41. 软件-块传输操作示例

通过 CPU 设置传输内容并指示启动传输后，DMAC 执行以下操作：

- 由于指定了传输数据位宽，传输以半字（16 位）执行。
- 根据传输源及传输目标的启动地址、数据位宽以及递增/固定的规定，在源地址至目的地址区域执行块数 (=BC+1) 传输。
- 块传输时，每完成一次块传输则产生一个传输间隔。
- DMAC 根据传输数量 (=TC+1) 和传输块数 (=BC+1) 执行数据传输。根据 CPU 传输请求所传输的数据量大小等于“数据位宽 (WIDTH) × 块数 (BC+1) × 传输数量 (TC+1)”。
- 传输完成后，DMAC 通知 CPU 传输完成。
- 如传输完成后再次指示启动传输，因传输源地址已设置为重装 (RS=1)，将从之前的传输源地址重新启动传输。如果未将传输目标地址设置为重装 (RD=0)，将从先前传输结束地址的下一地址开始传输。同样，如果指定了 BC/TC 重装，下次传输将重装与上次传输相同的块数和传输数量值。

- 传输间隔期间不执行传输，插入传输间隔的目的是防止某一条 DMAC 通道占用系统总线的访问权。如多条通道均有传输请求，DMAC 切换在传输间隔计时期间执行传输操作的通道。通过调节 BC 及 TC 的设置可控制传输间隔生成的频率。

此外，传输间隔计时期间总线访问权也转至 CPU。此产品的系统总线为多层配置，带有专用于 DMA 的特殊系统总线。因此，如 CPU 与访问目标之间无冲突，CPU 操作的同时可执行传输。即使 CPU 与访问目标之间发生冲突，只要 DMAC 传输处在不同的地址段（RAM 和外设，或 FLASH 和 RAM 等），CPU 操作几乎不受影响。但如传输处在相同的地址段（RAM 和 RAM 等），CPU 性能可能会受影响，具体取决于所用块数，对此必须引起注意。

14.3.2 软件-突发（Burst）传输

图 43 所示为软件-突发传输操作示例。示例中应用以下设置：

- 传输模式：软件请求突发传输 (ST=1, TRI_SEL[5:0]=000000,MODE=01)；
- 传输源启动地址：SA(SRC_ADR=SA)；
- 传输源地址：固定和可重装 (FS=1, RS=1)；
- 传输目标启动地址：DA(DST_ADR=DA)；
- 传输目标地址：递增和不可重装 (FD=0, RD=0)；
- 传输数据大小：字（32 位），块数 =3，传输数量 =2(WIDTH=10,BC=2,TC=1)；
- 传输数量重装：重装的传输数量 (RC=1)。

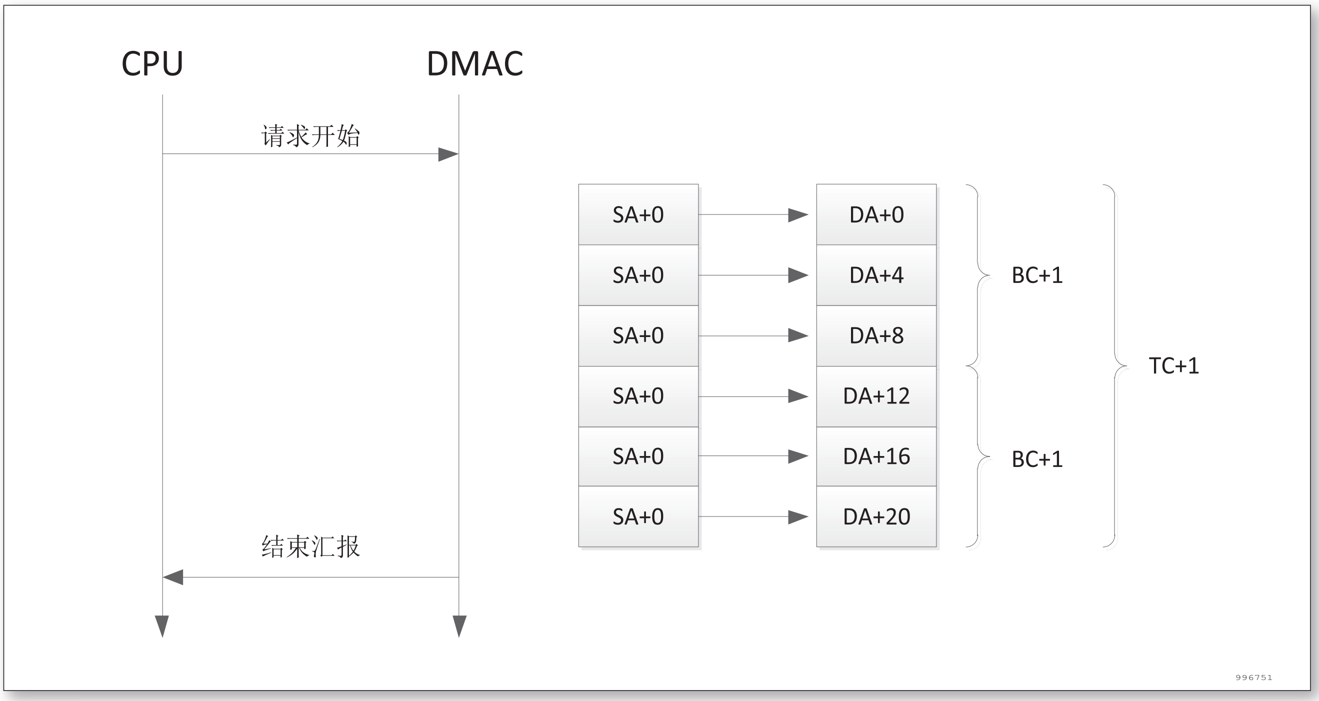

图 42. 软件-突发传输操作示例

通过 CPU 设置传输内容并指示启动传输后，DMAC 执行以下操作：

- 由于指定了传输数据位宽，传输以字（32 位）执行；
- 动地址 (SA+0) 相同；
- 突发传输时，连续执行传输，不产生传输间隔；
- DMAC 根据传输数量 (=TC+1) 执行块数 (=BC+1) 数据传输。根据 CPU 传输请求所传输的数据大小等于” 数据位宽 (WIDTH) × 块数 (BC+1) × 传输数量 (TC+1)”；

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

- 传输完成后，DMAC 通知 CPU 传输完成。

与块传输不同的是突发传输不产生传输间隔。由于受控通道占用了系统总线访问权，该通道可作为优先通道。

14.3.3 硬件-块（Block）传输

图 43 所示为硬件-块传输操作示例。示例中应用以下设置。其中省略了传输源及传输目标的地址设置以及传输数据位宽的设置。

- 传输模式：硬件-块传输（ST=0，TRI_SEL= 传输请求源的外设，MODE=00）；
- 传输数据大小：块数 = 4，传输数量 = 3 (BC=3,TC=2)。

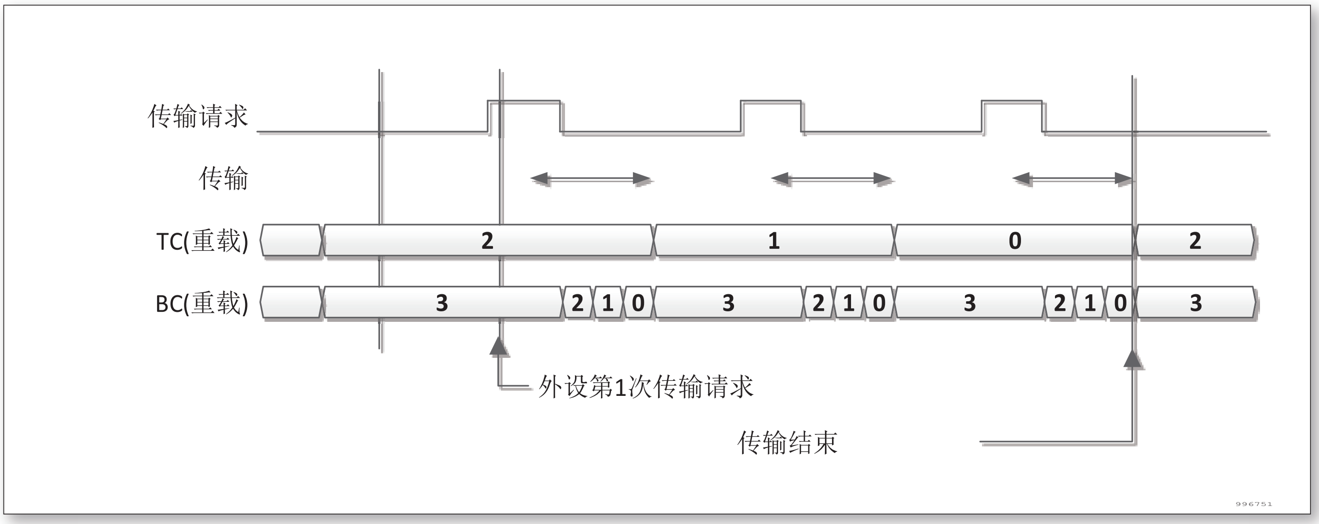

图 43. 硬件-块传输操作示例

硬件-块传输操作如下：

通过 CPU 指定传输内容，指令启动操作。DMAC 等待外设的传输请求。收到传输请求后，DMAC 执行块数 (=BC+1) 传输，然后等待下一次传输请求。等待时产生传输间隔。总共收到（TC+1）次传输请求，DMAC 执行 (TC+1) 次传输操作。执行的传输总数为 (BC+1) × (TC+1)。外设发出的传输请求必须与 DMAC 传输数量 (TC+1) 相匹配。完成所有传输后，DMAC 通知 CPU 传输完成。

14.3.4 硬件-突发（Burst）传输

图 44 所示为硬件-突发传输操作示例。示例中应用以下设置。其中省略了传输源及传输目标的地址设置以及传输数据位宽的设置。

- 传输模式：输请求源的外设，MODE=01）；
- 传输数据大小：块数 = 4，传输数量 = 5 (BC=3,TC=4)。

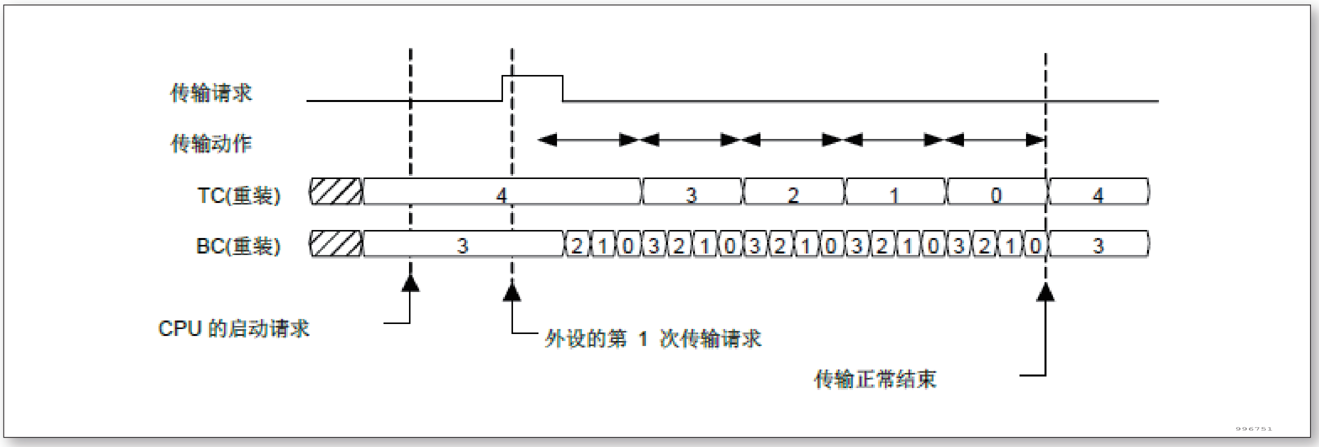

图 44. 硬件-突发传输操作示例

硬件-突发传输操作如下所述：

通过 CPU 指定传输内容，指令启动操作。DMAC 等待外设的传输请求。收到第一次传输请求后，DMAC 执行通过 (BC+1)×(TC+1) 计算出的所有次数的传输。硬件-突发传输过程不产生传输间隙。完成所有传输后，DMAC 通知 CPU 传输完成。

14.4 通道优先级控制

如 2 个通道均有传输请求，各通道传输间隔计时期间 DMAC 切换其中要传输的通道。此时根据优先控制确定执行传输的下一条通道。可通过 PRIO 从固定优先或循环优先中选择优先控制。

固定优先模式操作 (PRIO=0)

固定优先模式下，所有接到传输请求的通道中最小编号通道优先执行传输操作。(优先顺序：CH0>CH1)

首先执行最高优先级通道的传输（CH0)。最高优先级通道在传输间隔计时期间停止传输操作时，执行第二优先通道的传输操作（CH1)。因此，第一优先通道和第二优先通道交替执行传输操作。

循环优先模式操作 (PRIO=1)

循环优先模式下，所有通道均等地执行其传输操作。

14.5 DMA 寄存器

表 39. DMA 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x00	DMAC_CONF	所有通道配置寄存器	0X0000 0000	小节 14.5.1
0x10	DMAC_CONFA0	通道 0 配置 A 寄存器	0X0000 0000	小节 14.5.2
0x14	DMAC_CONFB0	通道 0 配置 B 寄存器	0X0000 0000	小节 14.5.3
0x18	DMAC_SRCADR0	通道 0 传输源地址寄存器	0X0000 0000	小节 14.5.4
0x1C	DMAC_DSTADR0	通道 0 传输目的地址寄存器	0X0000 0000	小节 14.5.5
0x20	DMAC_CONFA1	通道 1 配置 A 寄存器	0X0000 0000	小节 14.5.2
0x24	DMAC_CONFB1	通道 1 配置 B 寄存器	0X0000 0000	小节 14.5.3
0x28	DMAC_SRCADR1	通道 1 传输源地址寄存器	0X0000 0000	小节 14.5.4

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Offset	Acronym	Register Name	Reset	Section
0x2C	DMAC_DSTADR1	通道 1 传输目的地址寄存器	0X0000 0000	小节 14.5.5

14.5.1 所有通道配置寄存器 (DMAC_CONF)

起始地址：0x40021000

地址偏移：0x00

复位值：0X00000000

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td>EN</td><td>ST</td><td rowspan="2">Res</td><td>PRIO</td><td colspan="4">HALT[3:0]</td><td colspan="8" rowspan="2">Reserved</td></tr>
<tr><td>RW</td><td>RW</td><td>RW</td><td colspan="4">RW</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="16">Reserved</td></tr>
</table>

Bit	Field	Type	Reset	Description
31	EN	rw	0x00	该位控制所有通道的使能和禁止传输操作。 当该位置"1" 时，使能所有通道的操作，每个通道按照各自设置工作。 当该位清"0" 时，所有通道操作都被禁止，直到该位重新置"1"。正在传输的通道强制停止传输。
30	ST	rw	0x00	该位指示所有通道的传输状态。 如果下列任何情况在传输中发生，DMAC 将该位置" 1"。 当"0" 写入 CONF:EN 位，并且所有通道传输结束。 当除"0000" 外的值写入 CONF:HALT 位，并且所有通道传输暂停。 当 CONF:EN=1 和 CONF:HALT=0000 有效，并且所有的通道使能操作，DMAC 将本位清零。 尽管本位的属性是 RW, CPU 写"1" 不影响 DMAC 的操作。然而如果 CONF 寄存器需要更新而又不想改变当前位，应该首先读出这个位，然后重新写入同样的值。
29	Reserved			保留。
28	PRIO	rw	0x00	PRIO 控制各个通道的传输优先级。 当该位置"0"，各通道优先级固定。CH0 优先级 > CH1 优先级。 当该位置"1"，各通道优先级采用轮询方式决定。

Bit	Field	Type	Reset	Description
27:24	HALT[3:0]	rw	0x00	HALT 位控制所有通道的传输暂停或取消。 当这些位设置为"0000" 以外的值时，所有正在传输的通道进入暂停状态。当设置为"0000"，所有传输继续。 即便来自外部或者外设的传输请求有效，暂停状态的通道忽略传输请求。如果是块传输和突发传输，即便暂停清除，相关的通道也不启动传输。为了完成传输过程中暂停的传输，需要在暂停取消后发动另外的传输请求。 这些位可以用于暂停传输而不复位该通道的配置寄存器。
23:0	Reserved			保留。

14.5.2 通道 0、通道 1 配置 A 寄存器 (DMAC_CONFA0、DMAC_CONFA1)

起始地址：0x40021000

地址偏移：0x10(DMAC_CONFA0)　　0x20(DMAC_CONFA1)

复位值：0X00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ENS	PAS	ST	TRI_SEL[6:0]							Reserved		BC[3:0]			
RW	RW	RW	RW									RW			

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

Bit	Field	Type	Reset	Description
31	ENS	rw	0x00	ENS 位控制单独通道的使能和禁止功能。 当该位置"1", 相关通道使能，等待触发，开始传输功能(CONF:EN 必须置"1")。 如果 MSK 位 (CONFB[0]) 位清"0"，DMAC 在完成传输后清除本位，即"0"。 当该位为"0", 相关通道禁止，不进行数据传输。同时，如果正好在传输过程中，那么停止传输。 该位可用于强制停止正在传输操作的相关通道，同时复位配置寄存器。

Bit	Field	Type	Reset	Description
30	PAS	rw	0x00	PAS 位控制各自通道的传输暂停或取消。 当该位置"1"，相关正在传输的通道进入暂停状态。清除该位到 0，从新开始传输。 该位清除为"0" 当通道的传输结束后。 即便外部或外设传输请求有效，暂停状态通道忽略传输请求。如果是块传输和突发传输，相关通道不会开始传输，即使是清除了暂停位。为了完成暂停的传输，在清除暂停后发送另外的传输请求。 该位可用于暂停传输而不复位相关通道的配置寄存器。
29	ST	rw	0x00	ST 位用于产生单独通道的软件传输请求。 当该位置"1", 软件传输触发产生，相关通道开始传输。传输完成后，DMAC 清除该位至"0"。 当该位在传输中置"0"，传输停止。
28:22	TRI_SEL[6:0]	rw	0x00	TRI_SEL[6:0] 位选择传输请求的触发源。 设置 IS[6:0] 为： 0000000 - 传输触发设定为软件请求 1000000 - 保留　1000001 - 保留 1000010 - SPI1 RX　1000011 - SPI1 TX 1000100 - ADC JRQ0 RX　1000101 - ADC SRQ0 RX 1000110 - LCD TX　1000111 - Reserved 1001000 - UART0 RX　1001001 - UART0 TX 1001010 - UART1 RX　1001011 - UART1 TX 1001100 - LPUART0 RX　1001101 - LPUART0 TX 1001110 - LPUART1 RX　1001111 - LPUART1 TX 1010000 - DAC0　1010001 - Reserved 1010010 - TIM0A　1010011 - TIM0B 1010100 - TIM1A　1010101 - TIM1B 1010110 - TIM2A　1010111 - TIM2B 1011000 - TIM3A　1011001 - TIM3B 1011010 - TIM4A　1011011 - TIM4B 1011100 - TIM5A　1011101 - TIM5B 1011110 - TIM6A　1011111 - TIM6B 1100000 - UART4 RX　1100001 - UART4 TX 1100010 - UART5 RX　1100011 - UART5 TX
21:20	Reserved			保留。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
19:16	BC[3:0]	rw	0x00	BC[3:0] 位指定块/突发传输的个数。 设置为请求传输时，设置"0000"。 设置"BC[3:0]= 块数量 - 1"。最大允许传输的块数为 16。 这几位在传输的工程中可以去读。通常情况，当一个源地址访问，或者一个目的地址的访问成功完成，BC[3:0] 减去 1。 -在 CONFB:RC=1 的情况下： 传输开始时设置的值，在传输完成后重载。 -在 CONFB:RC=0 的情况下： 成功完成传输后，该位清零，同时在传输暂停引起的传输非成功完成时，该值保持不变。
15:0	TC[15:0]	rw	0x00	TC[15:0] 位用于指定块/突发/请求传输的传输次数。 设定值"TC = 传输数量 - 1". 最大传输数为 65536。 传输时，这些值是可以读的。正常情况，当一次块传输完成，TC 减去 1。 -在 CONFB:RC=1 的情况下 传输开始时设置的值，在传输完成后重载。 -在 CONFB:RC=0 的情况下 成功完成传输后，该位清零，同时在传输暂停引起的传输非成功完成时，该值保持不变。

14.5.3 通道 0、通道 1 配置 B 寄存器 (DMAC_CONFB0、DMAC_CONFB1)

起始地址：0x40021000

地址偏移：0x14(DMAC_CONFB0)　　0x24(DMAC_CONFB1)

复位值：0X00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved		MODE[1:0]		WIDTH[1:0]		FS	FD	RC	RS	RD	ERR_IE	FIS_IE	STAT[2:0]		
		RW		RW		RW	RW	RW	RW	RW	RW	RW	RW		

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

Bit	Field	Type	Reset	Description
31:30	Reserved			保留。

Bit	Field	Type	Reset	Description
29:28	MODE[1:0]	rw	0x00	该位控制传输模式： 00：Block 传输（初始值） 01：Burst 传输 10：Reserved 11：Reserved
27:26	WIDTH[1:0]	rw	0x00	该位指示传输数据的宽度： 00：字节（8-Bit）（初始值） 01：半字（16-Bit） 10：字（32-Bit） 11：Reserved
25	FS	rw	0x00	该位用于指定源地址自增或者固定： 0：按 CONFB:WIDTH[1:0] 的数据位宽自增传输源地址（初始值） 1：固定传输源地址
24	FD	rw	0x00	该位用于指定目的地址自增或者固定： 0：按 CONFB:WIDTH[1:0] 的数据位宽自增传输目的地址（初始值） 1：固定传输目的地址
23	RC	rw	0x00	该位控制 CONFA:BC[3:0] 和 CONFA:TC[15:0] 的重载功能。 当该位置”1”时，CONFA:BC[3:0] 和 CONFA:TC[15:0] 在传输完成后重新加载传输开始时的设定值。 0：禁止重新加载 BC/TC 值（初始值） 1：使能重新加载 BC/TC 值
22	RS	rw	0x00	该位控制传输源地址的重载功能 0：禁止重新加载传输源地址（初始值） 1：使能重新加载传输源地址
21	RD	rw	0x00	该位控制传输目的地址的重载功能 0：禁止重新加载传输目的地址（初始值） 1：使能重新加载传输目的地址
20	ERR_IE	rw	0x00	该位使能或禁止在传输未成功完成中断的产生。 0：禁止传输出错时产生中断（初始值） 1：使能传输出错时产生中断 当该位设置为”1”时，在如下情况下，传输出错中断发出。 -地址溢出 CONFB:STAT[2:0]=”001” - 由外设发出的停止传输引起的停止，或者设置 DMAC_CONF:EN、DDMAC_CONFA:ENS 引起的禁止传输 CONFB:STAT[2:0]=010 -传输源地址访问错误 CONFB:STAT[2:0]=011 -传输目的地址访问错误 CONFB:STAT[2:0]=100

Bit	Field	Type	Reset	Description
19	FIS_IE	rw	0x00	该位使能或禁止传输成功完成中断的产生。 0：禁止传输成功完成时产生中断（初始值） 1：使能传输成功完成时产生中断 当该位置”1”时，如果 CONF:STAT 在传输完成时为成功完成传输，中断发生。
18:16	STAT[2:0]	rw	0x00	该位使用 3-bit 代码指示传输的当前状态。 如果一个成功传输完成中断或者传输非成功完成中断发生，中断信号可以通过写“000”到该位来清除当前状态。 000 -初始值 001 -传输错误引起中止（地址溢出） 010 -传输停止请求引起中止 (外设停止请求引起的停止或者 EB/DE 位引起的禁止传输) 011 -传输错误引起中止 (传输源地址访问错误) 100 -传输错误引起中止 (传输目的地地址访问错误) 101 -成功传输完成 110 -保留 111 – 传输暂停
15:1	Reserved			保留。
0	MSK	rw	0x00	该位用于在传输完成时，屏蔽来自 DMAC 的 EB 位 (CONFA[31]) 清除功能。 -如果 MSK=0， DMAC 在传输完成时清除 CONFA:ENS 位。 -如果 MSK=1， DMAC 在传输完成时不清除 CONFA:ENS 位。这个功能允许连续传输而不需要 CPU 干预。 这个功能用于硬件传输。使用这个功能时，可同时使能重载功能 CONFB:RC，CONFB:RS 和 CONFB:RD 位。

14.5.4 通道 0、通道 1 传输源地址寄存器 (DMAC_SRCADR0、DMAC_SRCADR1)

起始地址：0x40021000

地址偏移：0x18(DMAC_SRCADR0)　　0x28(DMAC_SRCADR1)

复位值：0X00000000

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

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

Bit	Field	Type	Reset	Description
31:0	SRCADR[31:0]	rw	0x00	SRCADR[31:0] 位指定传输源地址。 不能设置相对于传输数据位宽 (CONFB:WIDTH[1:0]) 非对齐的地址。传输时可读取这些位的值。 -如果 CONFB:FS= “1”，传输源地址设定为固定值，并且不会改变。 -如果 CONFB:FS= “0”，且 CONFB:RS= “0”，传输时源地址根据 CONFB:WIDTH[1:0] 的数据位宽自增。如果传输成功完成，这个值为传输完成地址的下一地址。如果传输未成功完成，这个值为暂停时所设定的值。 -如果 CONFB:FS= “0” 且 CONFB:RS= “1”，传输时源地址根据 CONFB:WIDTH[1:0] 的数据位宽自增。传输完成后，传输开始时设定的源地址重新加载到本寄存器位中。

14.5.5 通道 0、通道 1 传输目的地址寄存器 (DMAC_DSTADR0、DMAC_DSTADR1)

起始地址：0x40021000

地址偏移：0x1C(DMAC_DSTADR0)　　0x2C(DMAC_DSTADR1)

复位值：0X00000000

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

Bit	Field	Type	Reset	Description
31:0	DSTADR[31:0]	rw	0x00	DSTADR[31:0] 位指定传输目的地址。 不能设置相对于传输数据位宽 (CONFB:WIDTH[1:0]) 非对齐的地址。传输时可读取这些位的值。 -如果 CONFB:FD= “1”，传输目的地址设置为固定地址，并且不改变。 -如果 CONFB:FD= “0”，且 CONFB:RD= “0” , 传输时目的地址根据 CONFB:WIDTH[1:0] 的数据位宽自增。如果传输成功完成，这个值为传输完成地址的下一地址。如果传输未成功完成，这个值为暂停时所设定的值。 -如果 CONFB:FD= “0” 且 CONFB:RD= “1” , 传输时目的地址根据 CONFB:WIDTH[1:0] 的数据位宽自增。传输完成后，传输开始时设定的目的地址重新加载到本寄存器位中。

15 通用定时器（TIM0/1/2/3）

15.1 通用定时器简介

TIM0/1/2 是分别由 1 个计数单元和 2 个比较单元组成的定时器。每个定时器支持 2 个独立 PWM 输出或 1 对互补 PWM 输出和 1 个独立 PWM 输出。支持 2 个捕获输入。TIM0/1/2 可以组成 3 对互补 PWM 输出。

TIM3 是由 1 个计数单元和 6 个比较单元组成的定时器，支持 6 个独立 PWM 输出或 3 对互补 PWM 输出。支持 6 个捕获输入。

使用定时器预分频、系统预分频器和系统时钟选择，可灵活的调整脉冲宽度和波形周期；可以方便的测量脉冲宽度。

15.1.1 基本特性 (TIM0/1/2)

- 2 路独立 PWM 输出 CHA,CHB；
- 1 路互补 PWM 输出（CHA,CHB）+1 路独立 PWM 输出（gate）；
- 1 路互补 PWM 输出（CHA,CHB）+1 路捕获功能 (gate)；
- 最多 2 路捕获输入；
- 脉冲宽度测量；
- 死区控制；
- 刹车控制；
- 边沿对齐、对称中心对齐与非对称中心对齐 PWM 输出；
- 正交编码计数功能；
- 单脉冲模式；
- 外部计数功能；
- DMA 触发。

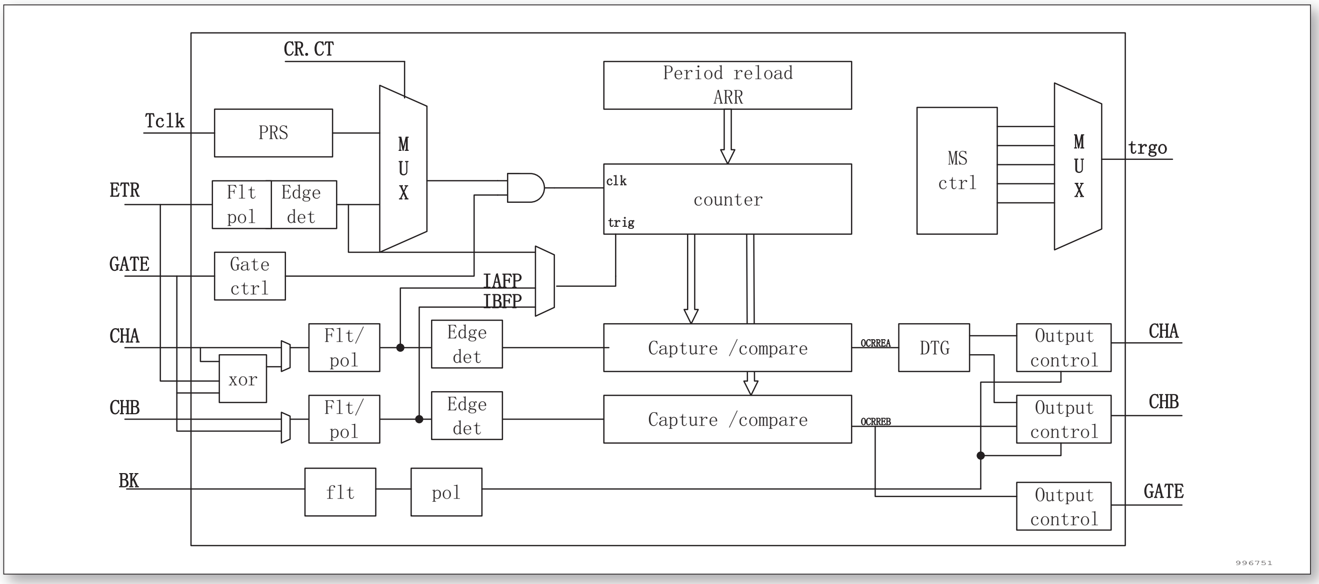

图 45. TIM0/1/2 框图

15.1.2 基本特性 (TIM3)

- 6 路独立 PWM 输出 CH0A,CH0B, CH1A,CH1B, CH2A,CH2B；

- 3 路互补 PWM 输出（CHxA,CHxB）+1 路独立 PWM 输出（gate3 路互补 PWM 输出（CHxA,CHxB）+1 路捕获功能 (gate)；
- 最多 6 路捕获输入；
- 脉冲宽度测量；
- 死区控制；
- 刹车控制；
- 边沿对齐、对称中心对齐与非对称中心对齐 PWM 输出；
- 正交编码计数功能；
- 单脉冲模式；
- 外部计数功能；
- DMA 触发。

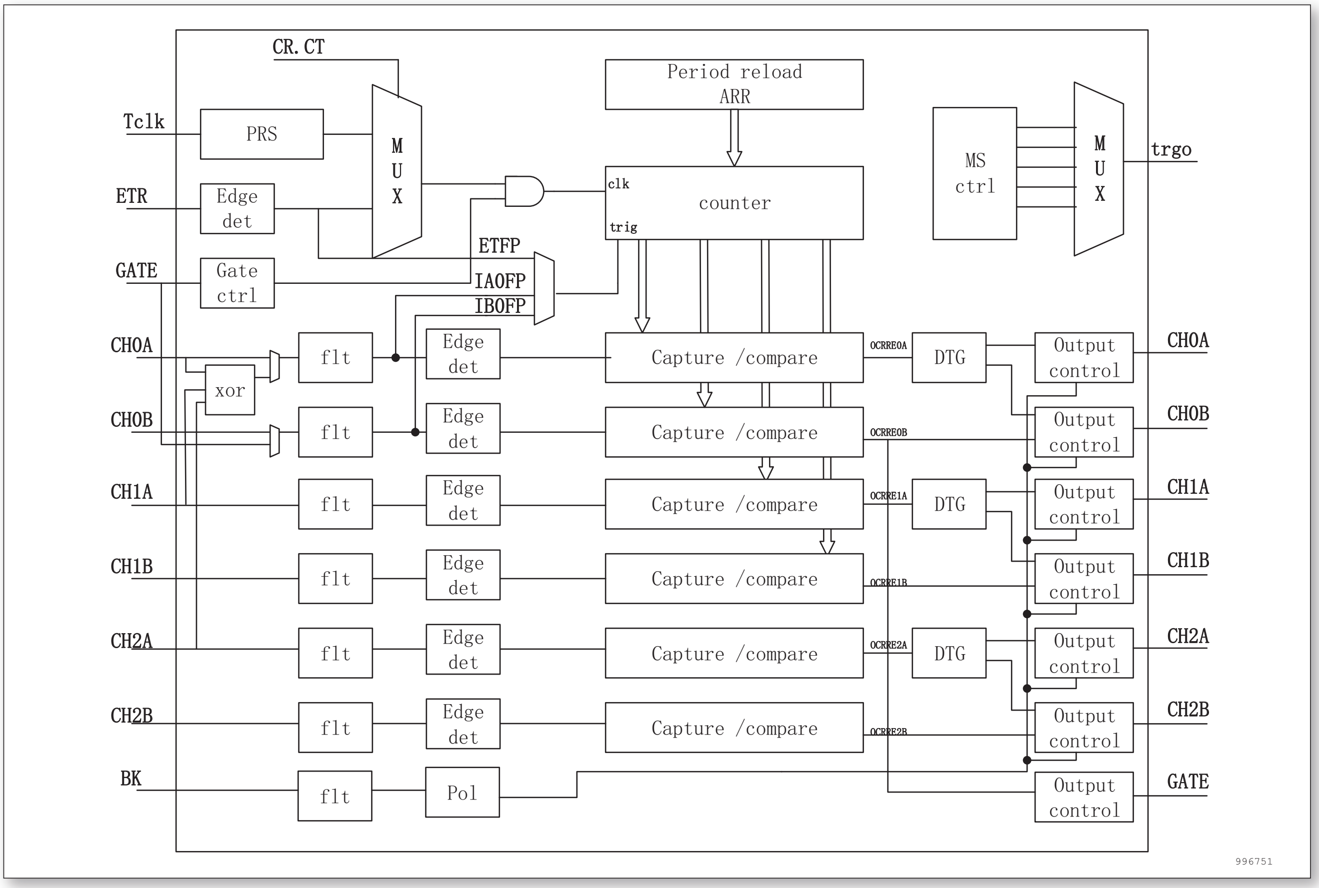

图 46. TIM3 框图

15.2 Timer 功能描述

15.2.1 定时器时钟

定时器使用 PCLK 作为定时器时钟，下文定时器时钟使用 TCLK 标示。

15.2.2 定时计数器

可编程通用定时器的主要模块是一个 16 位计数器及其相关的自动重载寄存器。计数器可递增计数、递减计数或交替进行递增和递减计数。计数器的时钟可通过预分频器进行分频。

计数器、自动重载寄存器和预分频器寄存器可通过软件进行读写。即使在计数器运行时也可执行读写操作。

15.2.3 定时器预除频

使用 TCLK 做为 Timer 定时器时钟时，可以使用预分频。预分频设置如表 40：

表 40: 预分频设置

PRS	000	001	010	011	100	101	110	111
分频比	1	2	4	8	16	32	64	256

预分频器没有预装载缓存，因此对预分频进行更改，预分频将会立刻生效。

15.2.4 模式 0 计数定时器功能

这种模式下，计数器向上计数。计数器支持两种计数模式，重载模式与自由计数模式，可以选择外部时钟 ETR 计数或系统时钟计数。门控 Gate 可以控制计数屏蔽。计数到最大值溢出产生中断。翻转输出端口 CHA,CHB，这种模式下 CHA,CHB 反相。

重载模式计数范围从 ARR 向上计数到 0xFFFF 溢出，然后从 ARR 再开始计数，计数周期为 0Xffff-ARR+1；自由计数模式从设置计数值计数到 0xFFFFFFFF 后溢出，溢出后计数值从 0x0 重新开始计数。

计数器可用通过外部门控功能控制计数器是否计数。控制关系如表 41

表 41: 计数器控制关系

M0CR.GATE	M0CR.GATEP	端口 GATE 输入	M0CR.CTEN	计数器
x	x	x	0	不计数
0	x	x	1	计数
1	0	高电平	1	计数
1	0	低电平	1	不计数
1	1	高电平	1	不计数
1	1	低电平	1	计数

功能框图

两种计数模式框图。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

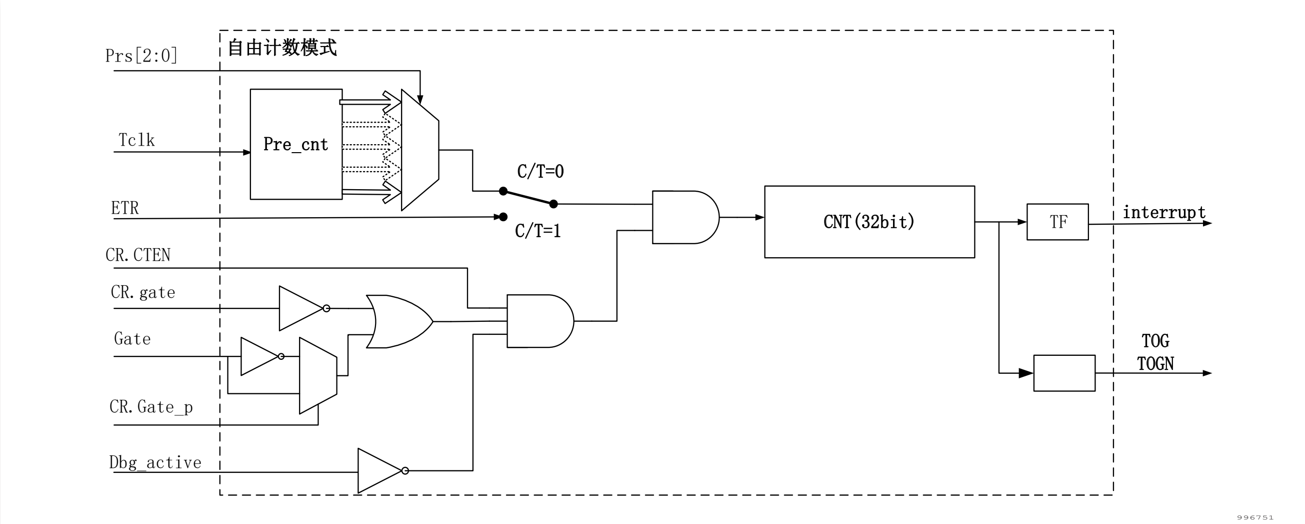

图 47. 自由计数框图

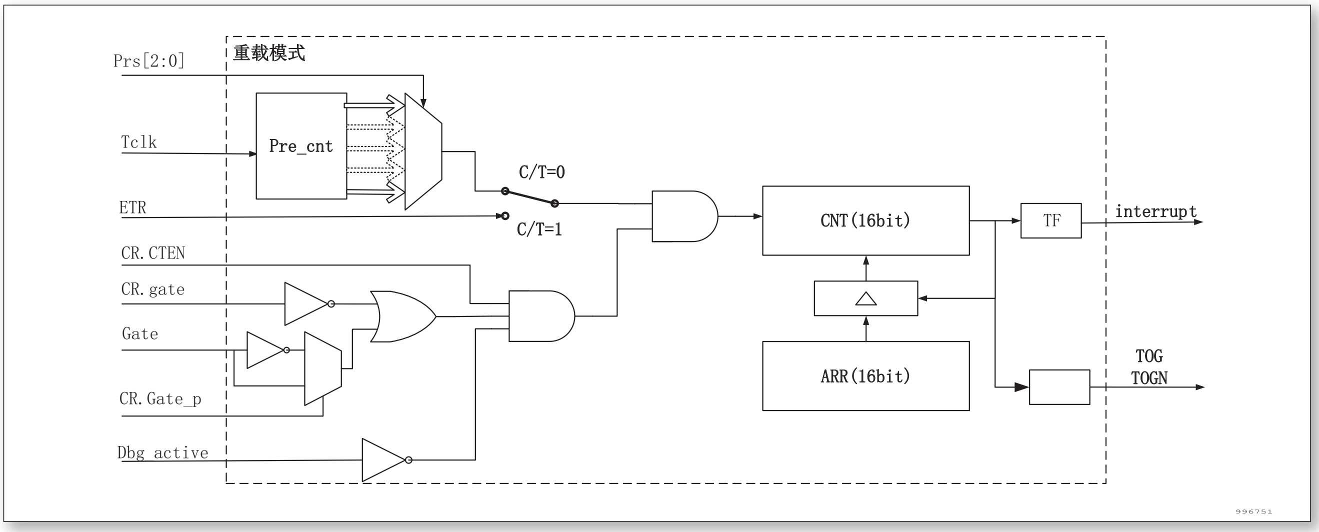

图 48. 重载计数波形

计数波形

两种计数模式波形。

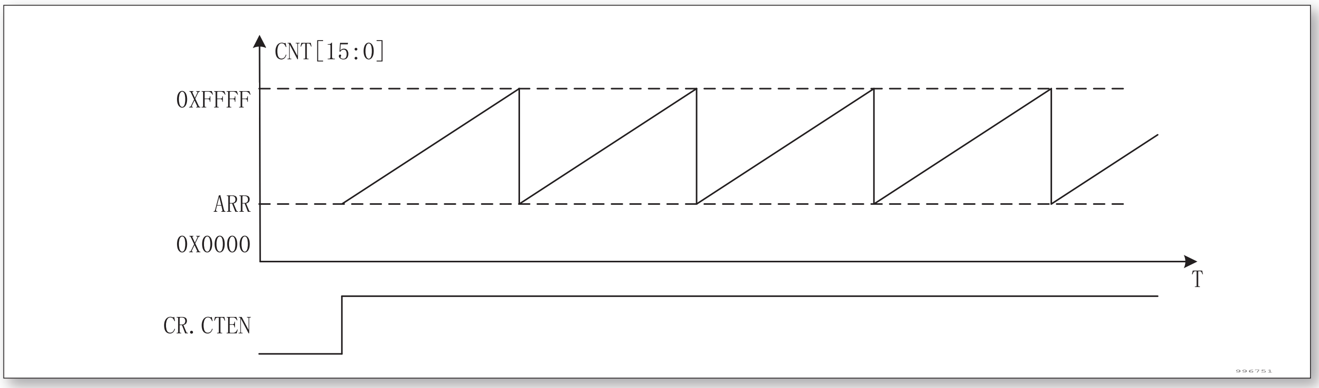

图 49. 16 位重载计数波形

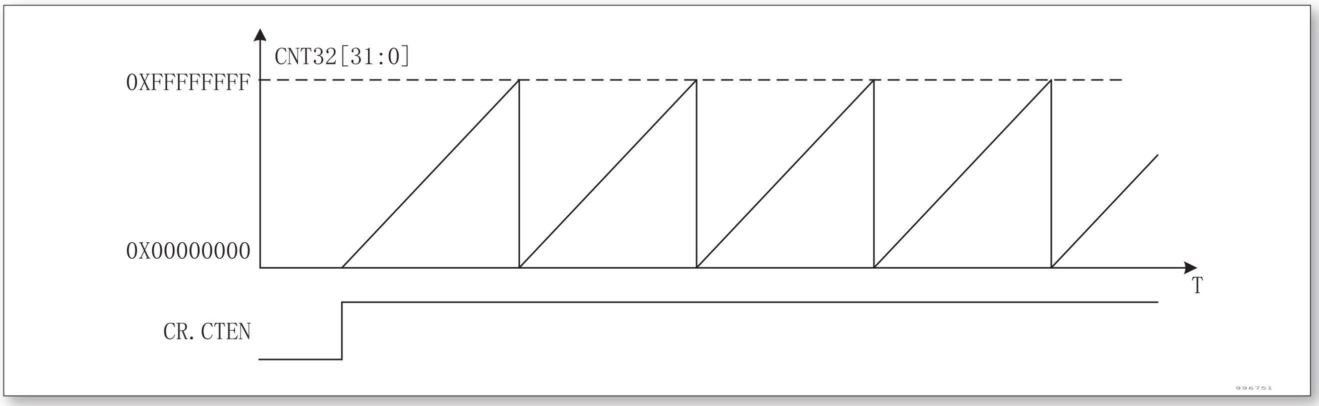

图 50. 32 位自由计数波形

计数功能

计数功能用于测定某个事件发生的次数。在计数功能中，计数器在每个相应的输入时钟的下降沿累加一次。输入信号被内部的 tclk 采样，因此外部输入时钟频率不能超过系统的 tclk 时钟。计数到最大值会溢出并且产生中断。中断标志需要软件清除。

定时功能

定时功能用于产生间隔定时。在定时功能中，定时器有预除频，定时器在每个预除频的一个时钟累加一次，计数到最大值会溢出并且产生中断。中断标志需要软件清除。

时序图

两种计数模式时序图。

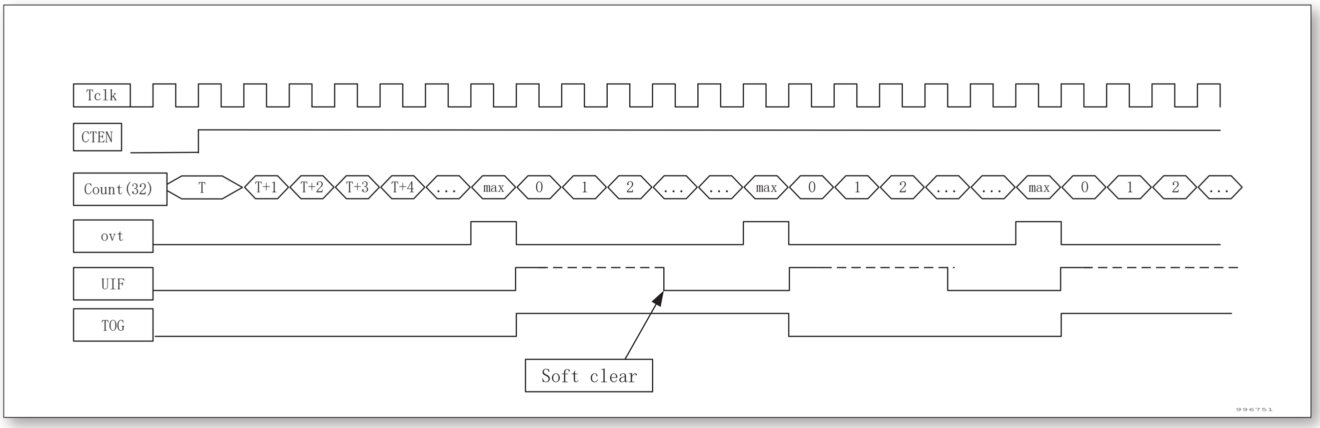

图 51. 自由计数时序图

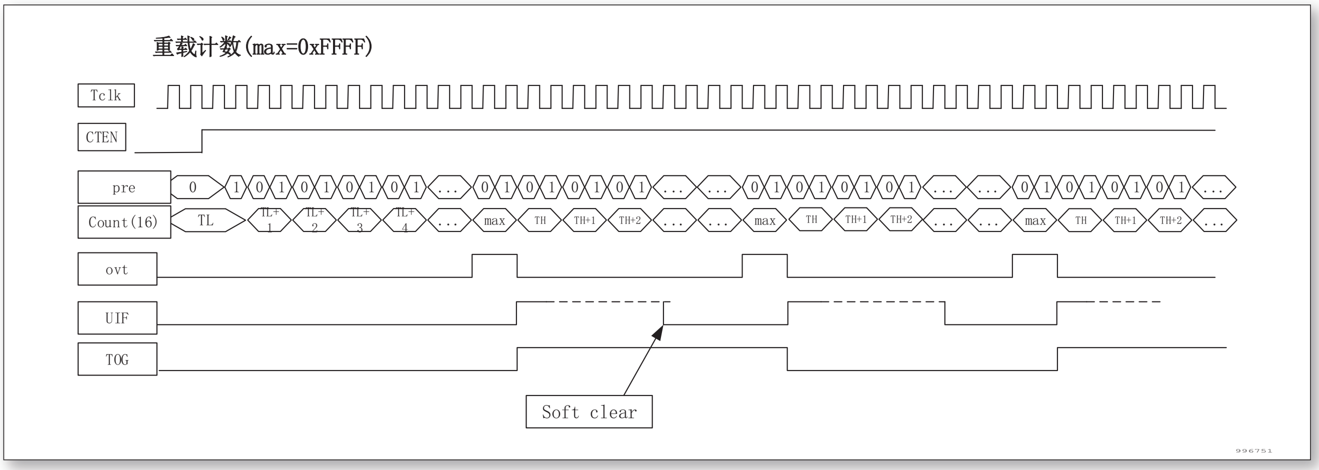

图 52. 重载计数时序图（预分频设置为 2）

Buzzer 功能

通过定时器的翻转输出功能可以实现驱动 Buzzer 的功能。使用 toggle 输出需要使能 DTR.MOE 控制位。设置 CR.TOG_EN 为 0 可以同时设置端口 CHA,CHB 输出为 0。在计数时钟为 4M 情况下 Buzzer 输出不同频率的 timer 重载模式配置如表 42：

表 42: Buzzer 频率配置

Buzzer 频率	计数器周期	计数器计数值	计数器重载值	CNTL 初始值	CNTH 重载值
1000Hz	0.5ms	2000	63536	0XF830	0XF830
2000Hz	0.25ms	1000	64536	0XFC18	0XFC18
4000Hz	0.125ms	500	65036	0XFE0C	0XFE0C

设置示例

- 重载定时器设置；
    1. 设置定时器模式 M0CR.MODE=0；
    2. 设置装载值 ARR；
    3. 设置计数器初值 CNT；
    4. 清除中断标志；
    5. 使能中断 M0CR.UIE；
    6. 使能重载模式 M0CR.MD；
    7. 开启定时器 M0CR.CTEN。
- 门控外部时钟自由计数设置；
    1. 设置定时器模式 M0CR.MODE=0；
    2. 设置计数器初值 CNT；
    3. 使能门控功能 M0CR.GATE；
    4. 选择门控有效电平 M0CR.GATEP；
    5. 清除中断标志；
    6. 使能中断 M0CR.UIE；
    7. 使能外部时钟模式 M0CR.CT；
    8. 开启定时器 M0CR.CTEN。

- BUZZER 输出控制。
    1. 根据输出频率设置合适的 ARR 值；
    2. 设置定时器为重载模式，参考重载定时器设置；
    3. 使能输出使能 DTR.MOE；
    4. 启动另外一个定时器控制 M0CR.TOGEN 实现频率的间隔输出。

15.2.5 模式 1 脉宽测量 PWC

这种模式下可以自动测量输入脉冲的高电平低电平或者周期宽度。

第一个有效边沿计数器初始化为 0x0001，第二个有效边沿将停止计数，并将当前计数值存入 CMAR，并且产生捕获中断 CAF，如果计数器发生溢出，会产生溢出标志。设置溢出中断使能会产生溢出中断。

表 43: Buzzer 频率配置

M1CR.edg1st	M1CR.edg2nd	脉宽测量
0	0	上沿 ~ 上沿周期宽度
0	1	上沿 ~ 上沿高电平宽度
1	0	下沿 ~ 上沿低电平宽度
1	1	下沿 ~ 上沿周期宽度

周期测量时，会间隔一个周期测量一个周期。

PWC 功能框图

如表 44为 PWC 测量框图。

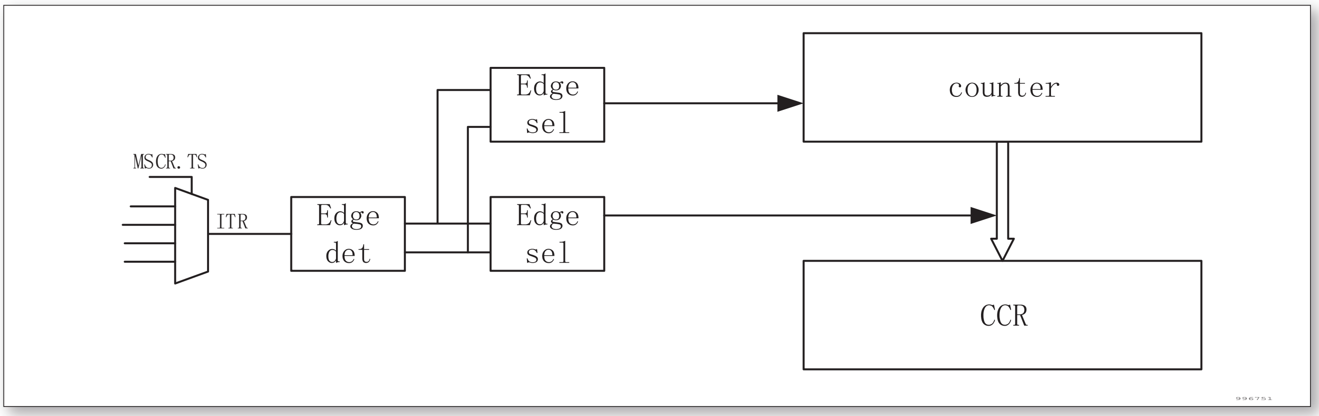

图 53. PWC 测量框图

表 44: MSCR.TS 配置

MSCR.TS	触发选择
000	端口 ETR 的滤波相位选择后的信号 ETFP
001	内部互联信号 ITR0
010	内部互联信号 ITR1
011	内部互联信号 ITR2
100	内部互联信号 ITR3

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

MSCR.TS	触发选择
101	无效
110	端口 CH0A 的滤波后的信号 IAFP（极性选择在脉宽测量模式下无效）
111	端口 CH0B 的滤波后的信号 IBFP（极性选择在脉宽测量模式下无效）

PWC 波形测量时序图

如图 54图 55图 56三种 PWC 波形测量时序图。

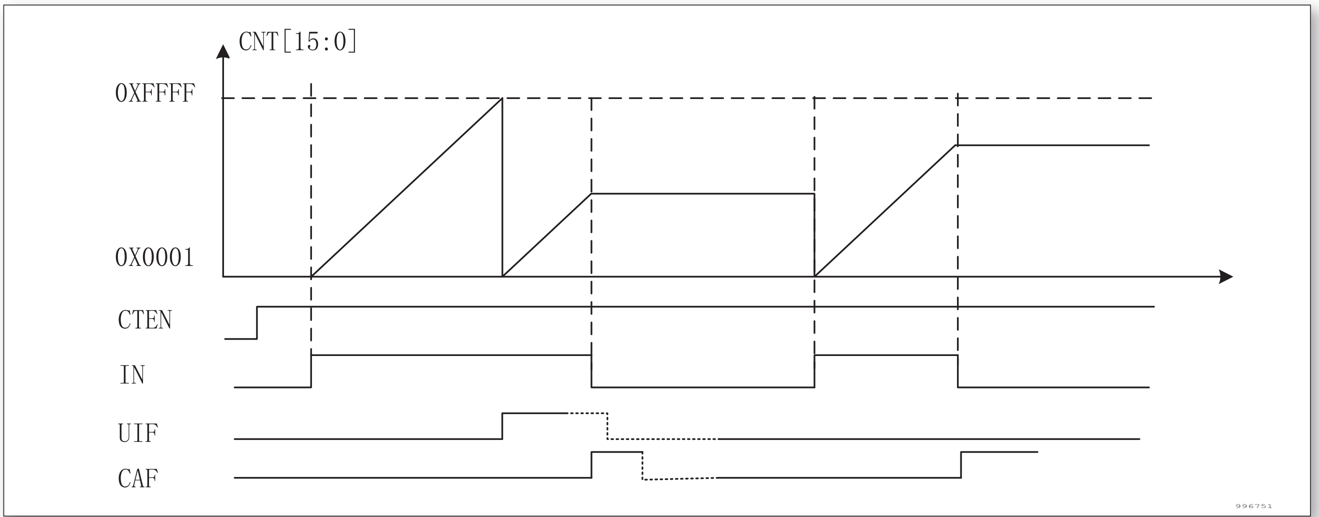

图 54. 高电平脉冲宽度测量

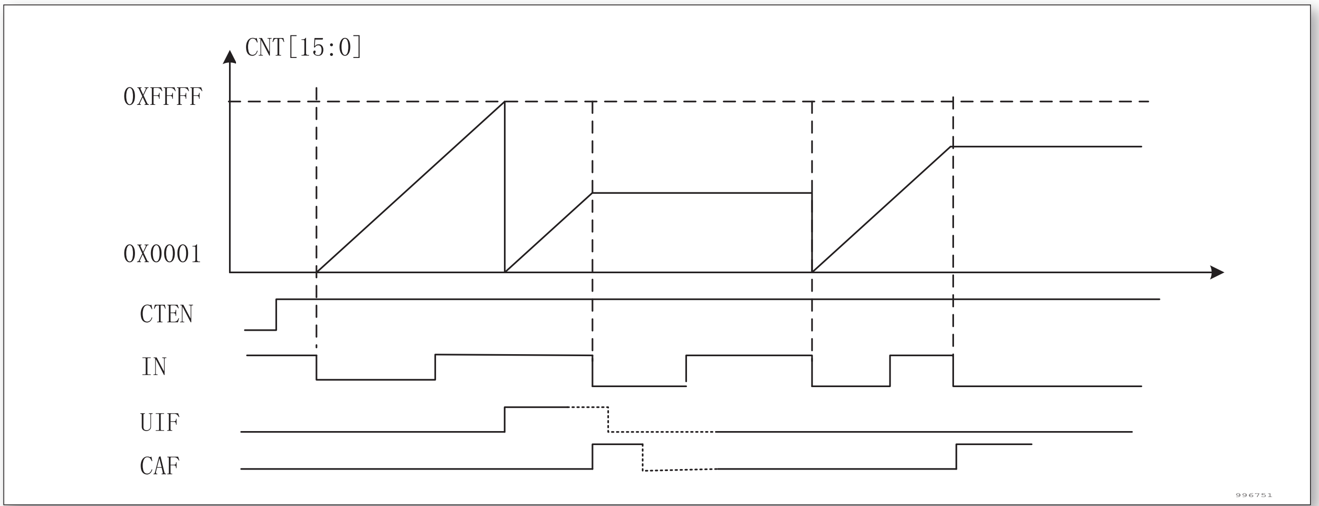

图 55. 下降沿到下降沿周期测量

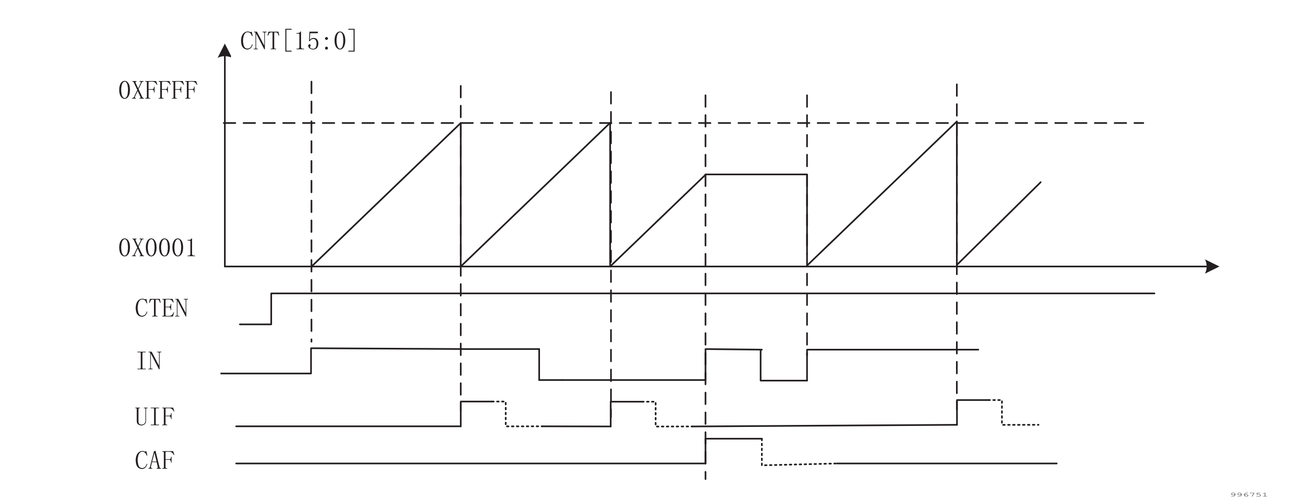

图 56. 上升沿到上升沿周期测量

通过寄存 MSCR.TS 选择测量信号源。

表 45: MSCR.TS 配置

MSCR.TS	测量信号源
000 ETFP	ETR 外部输输入滤波后的相位选择信号，可选择外部滤波与输入反向
001 ITR0	Timer 内部互联信号 0，其他 timer 的 TRGO 输出
010 ITR1	Timer 内部互联信号 1，其他 timer 的 TRGO 输出
011 ITR2	Timer 内部互联信号 2，其他 timer 的 TRGO 输出
100 ITR3	Timer 内部互联信号 3，其他 timer 的 TRGO 输出
101 IA0ED	无效
110 IAFP	CH0A 外部输输入滤波后的相位选择信号，可选择外部滤波与输入反向
111 IBFP	CH0B 外部输输入滤波后的相位选择信号，可选择外部滤波与输入反向

表 46: ITR 配置

ITRx / Timerx	ITR0	ITR1	ITR2	ITR3
Timer0	-	TIM1_TRGO	TIM2_TRGO	TIM3_TRGO
Timer1	TIM0_TRGO	-	TIM2_TRGO	TIM3_TRGO
Timer2	TIM0_TRGO	TIM1_TRGO	-	TIM3_TRGO
Timer3	TIM0_TRGO	TIM1_TRGO	TIM2_TRGO	-

注：关于 TRGO 输出参考寄存器描述。

PWC 单次触发模式

设置 M1CR.ONESHOT=1 可以设置 PWC 单次测量，测量完成后 CTEN 将被清除。

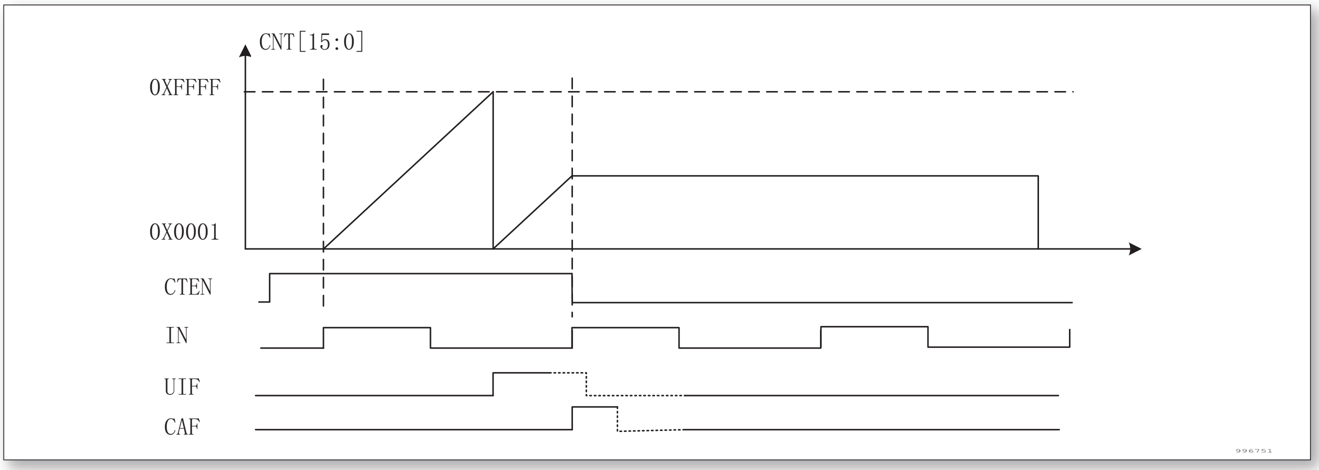

图 57. 上升沿到上升沿周期测量单次模式

设置示例

- 脉冲低电平测量设置；
  1. 设置为脉冲测量模式 M1CR.MODE=1；
  2. 设置 MSCR.TS 选择测量的信号；
  3. 设置 M1CR.edg2dn=0,M1CR.edg1st=1 选择测量低电平；
  4. 清除中断标志；
  5. 使能溢出中断 M1CR.UIE；
  6. 使能测量结束中断 CR0.CIEA；
  7. 使能定时器 M1CR.CTEN；
  8. 中断服务程序中读取 CCR0A 及溢出次数并清除中断标志；
  9. 等待下次测量。
- 脉冲高电平单次测量设置。
  1. 设置为脉冲测量模式 M1CR.MODE=1；
  2. 设置 MSCR.TS 选择测量的信号；
  3. 设置脉冲单次测量模式 M1CR.ONESHOT=1；
  4. 设置 M1CR.edg2dn=1, M1CR.edg1st=0 选择测量低电平；
  5. 清除中断标志；
  6. 使能溢出中断 M1CR.UIE；
  7. 使能测量结束中断 CR0.CIEA；
  8. 使能定时器 M1CR.CTEN；
  9. 中断服务程序中读取 CCR0A 及溢出次数并清除中断标志；
  10. 测量结束。

15.2.6 模式 2/3 比较捕获模式

计数器

如图 58为计数器框图。

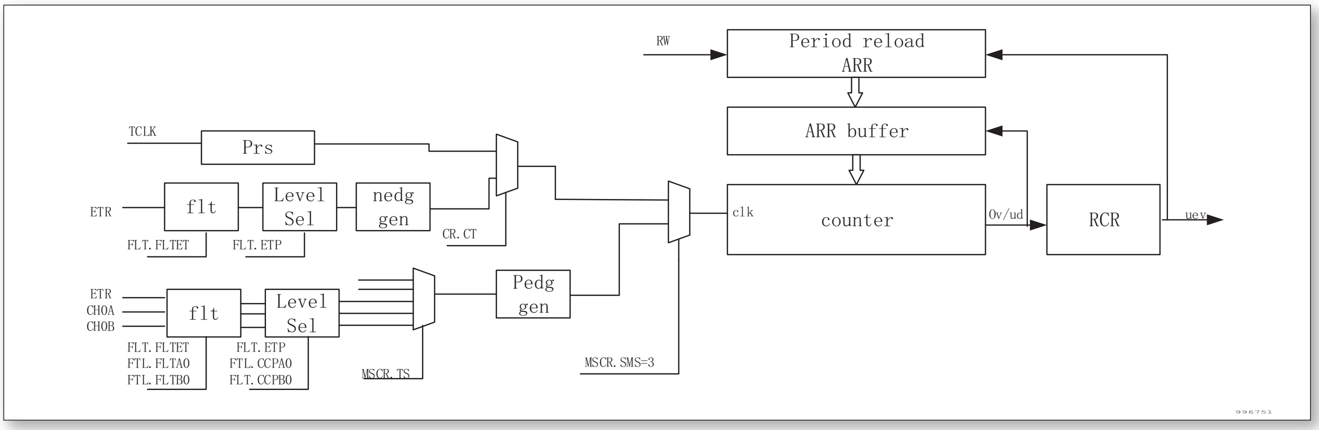

图 58. 计数器框图

计数器主要部分是一个 16 位计数器与相关的自动装载寄存器。这个计数器可以向上计数（模式 2），向下计数（模式 2）或向上向下双向计数（模式 3）。计数器的时钟可以由预分频器 PRS 分频得到，也可以选择 ETR 输入外部时钟或者通过 MSCR.TS 选择的外部输入信号和内部互联信号。

计数器基本单元包括：

- 计数器寄存器 CNT；
- 预分频寄存器 CR.PRS；
- 自动装载寄存器 ARR；
- 重复次数寄存器 RCR；
- 时钟选择控制寄存器 FLT,CR0,MSCR,CR。

自动装载寄存器具有缓存功能，计数器产生事件更新后重载值从缓存寄存器更新到计数器。当计数器停止状态或者缓存功能关闭状态，自动装载寄存器立刻更新到缓存寄存器。当定时器处于运行状态并行缓存功能有效时，写入到自动装载寄存器的值不会立刻更新到缓存寄存器，当事件更新后才有自动装载寄存器更新到缓存寄存器。

时钟选择及门控功能，触发功能，复位功能参考模式 2/3 从模式章节。

计数器波形

模式 2 为锯齿波计数波形，通过设置 CR.DIR 可以更改计数方向。

设置 CR.DIR 为 0 时，计数器为递增计数模式，这种模式下，计数器从 0 计数到自动重载值（TIMx_ARR），然后重新从 0 开始计数并生成计数器上溢事件。如果使用重复计数器，则当递增计数的重复次数达到重复计数器寄存器中编程的次数加一次 (TIMx_RCR+1) 后，将生成更新事件 (UEV)。否则，将在每次计数器上溢时产生更新事件。

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

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

- 自动重载缓存值将以 ARR 寄存器值进行更新；
- 比较缓存值将以比较寄存器 CCRxy 进行更新。

以下图示显示 ARR=0X2C 时不同计数方向的计数器波形：

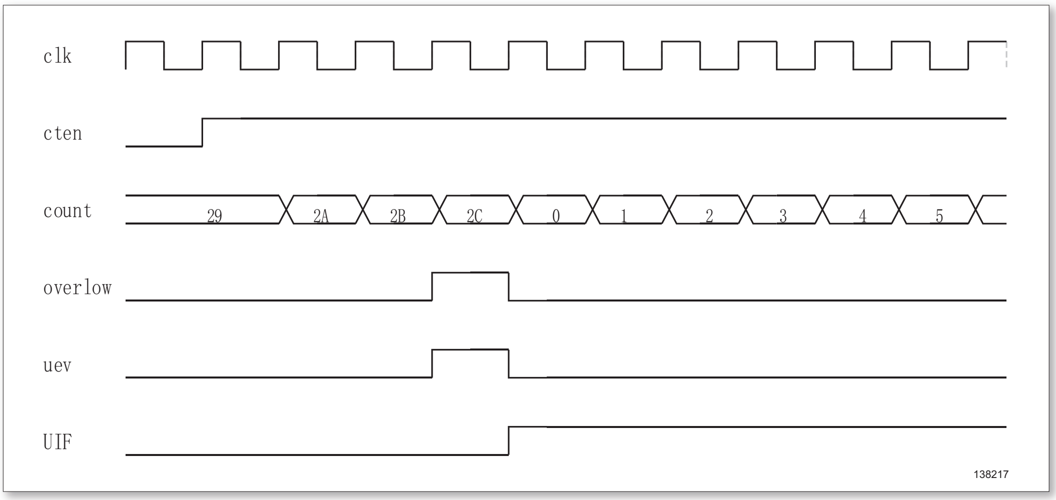

图 59. 无预分频的向上计数

计数溢出周期时钟个数为 ARR+1：

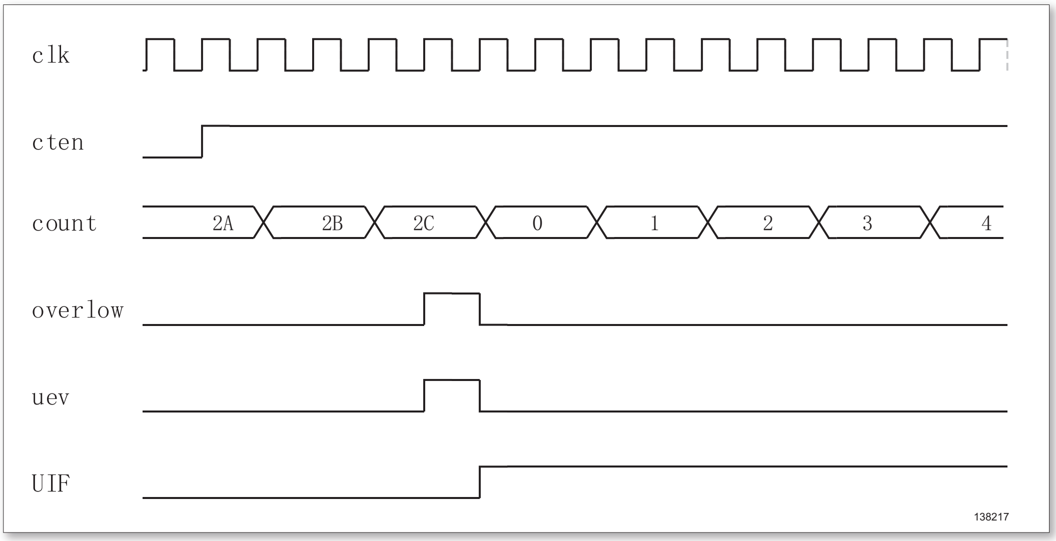

图 60. 带预分频的上计数

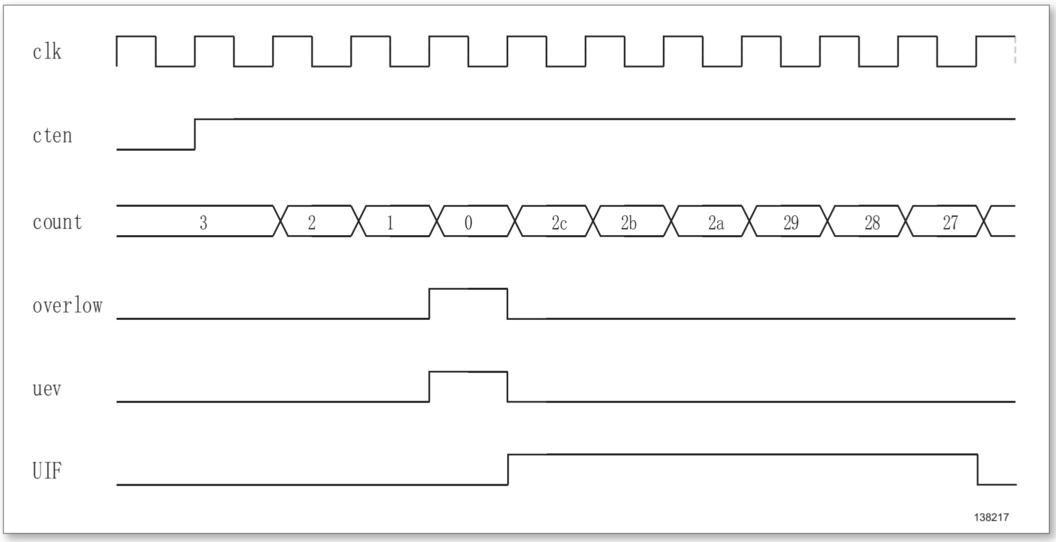

图 61. 不带预分频的下计数

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

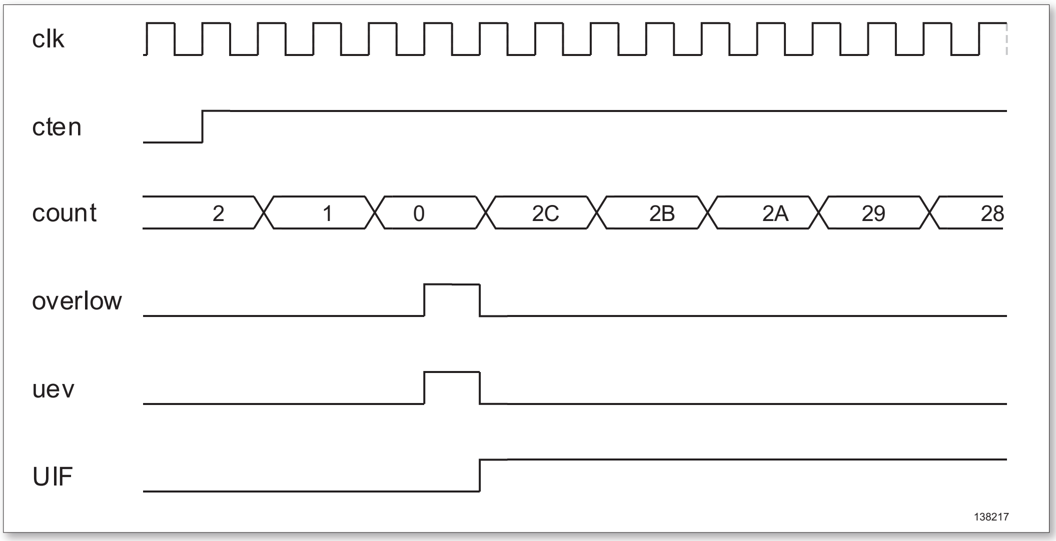

图 62. 带预分频的下计数

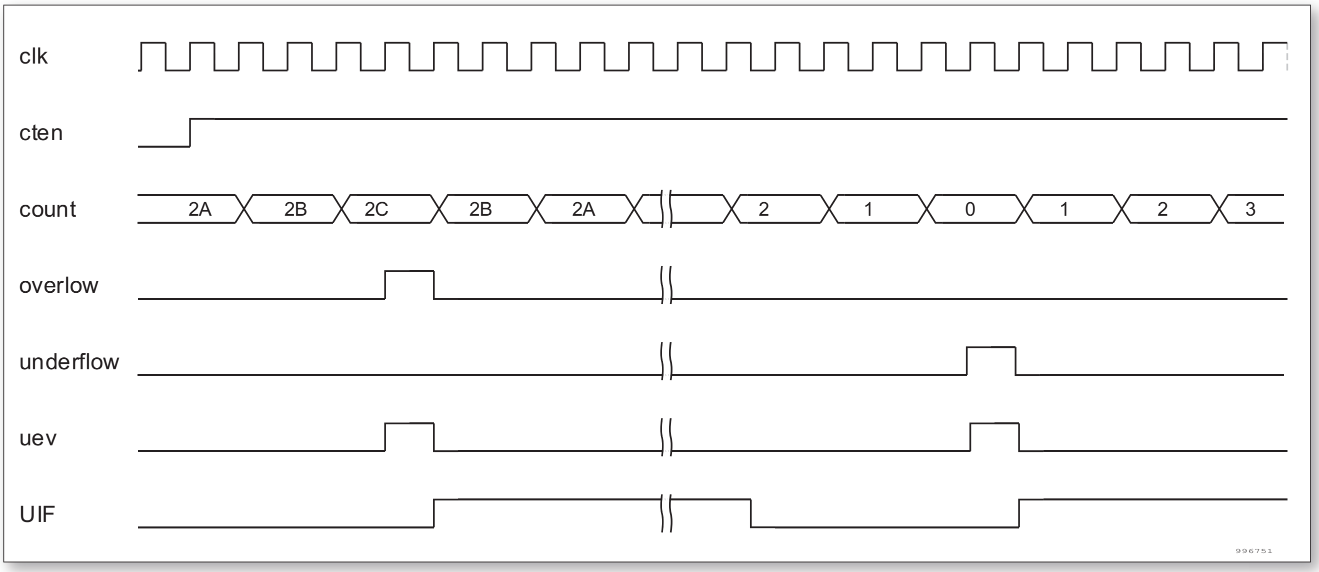

图 63. 带预分频的上下计数

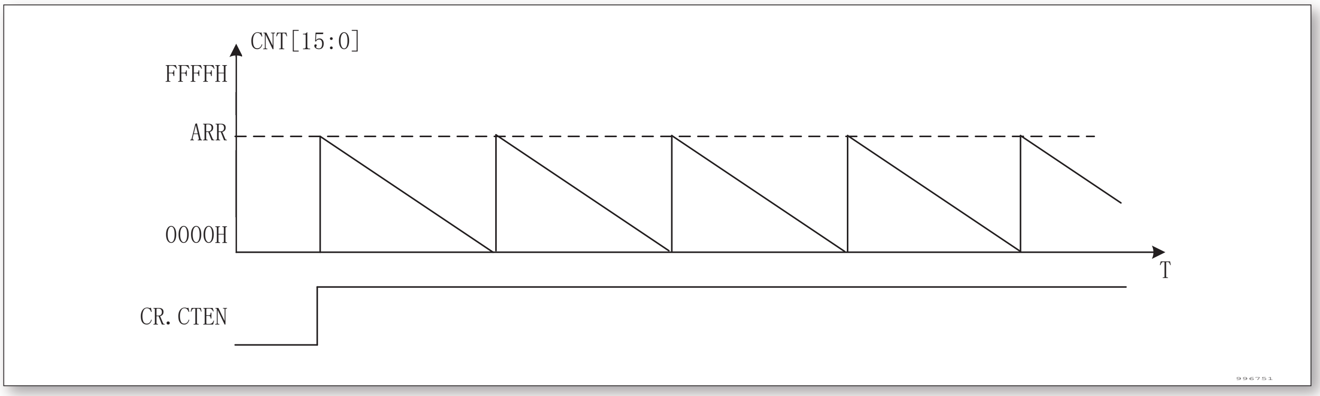

图 64. 边沿对齐计时器波形 (DIR = 1)

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

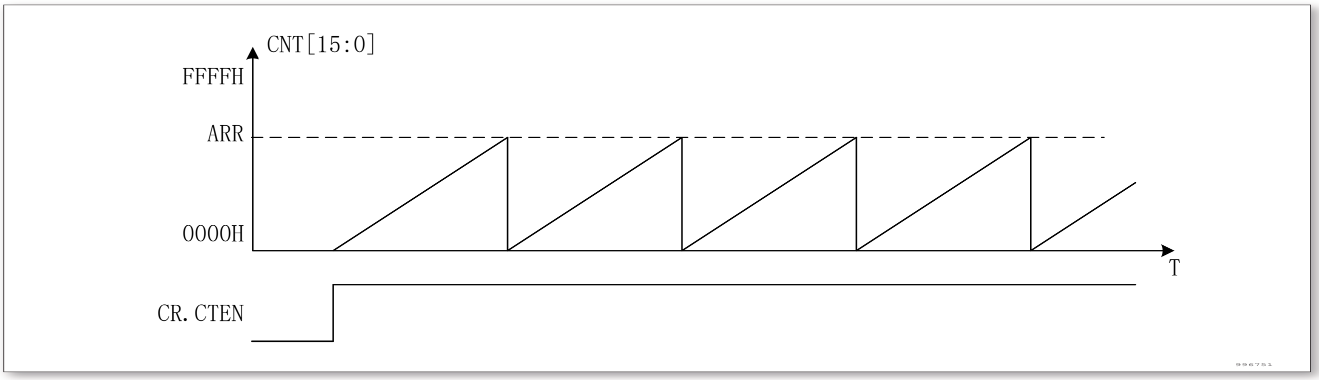

图 65. 边沿对齐计时器波形 (DIR = 0)

模式 3 为三角波计数波形，计数方向控制位只读，不可以更改计数方向。

中心对齐（三角波）模式下 CR.DIR 方向位是只读的。写值无效。从其他模式切换到中心对齐模式 DIR 自动清 0。软件事件更新和从模式外部触发复位模式 DIR 自动清零。

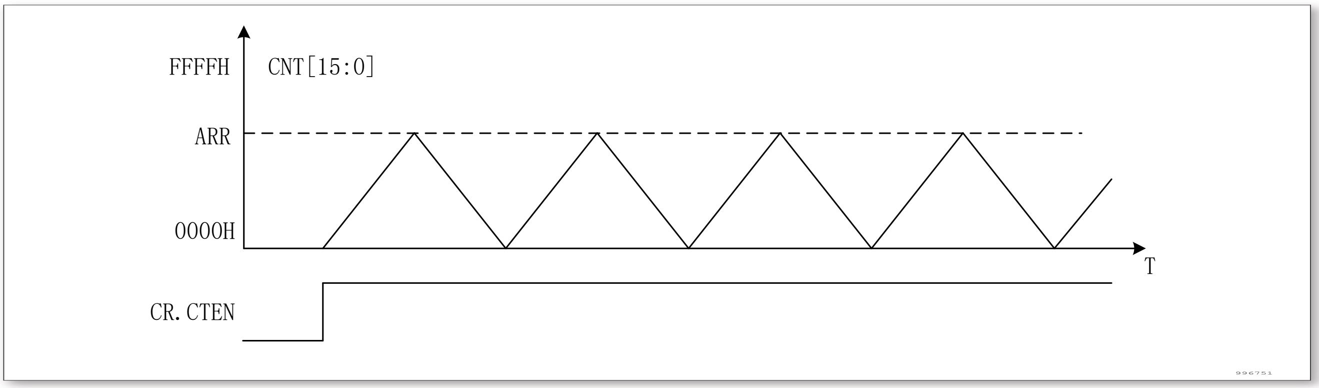

图 66. 中心对齐计数器波形 (DIR = 0)

重复计数

重复计数器使用计数器的溢出进行向下计数。计数到 0 时，即计数器发生重复寄存器设置的值加一次溢出时。当缓存寄存器使能时，周期重载寄存器更新到周期缓存寄存器。比较模式下比较寄存的值更新到比较缓存寄存中。

重复计数器在下面条件成立时递减。

- 上计数模式下每次计数器溢出时；
- 下计数模式下每次计数器下溢时；
- 三角波模式下每次上溢出和每次下溢出时。

不同计数模式更新时序如图 67。

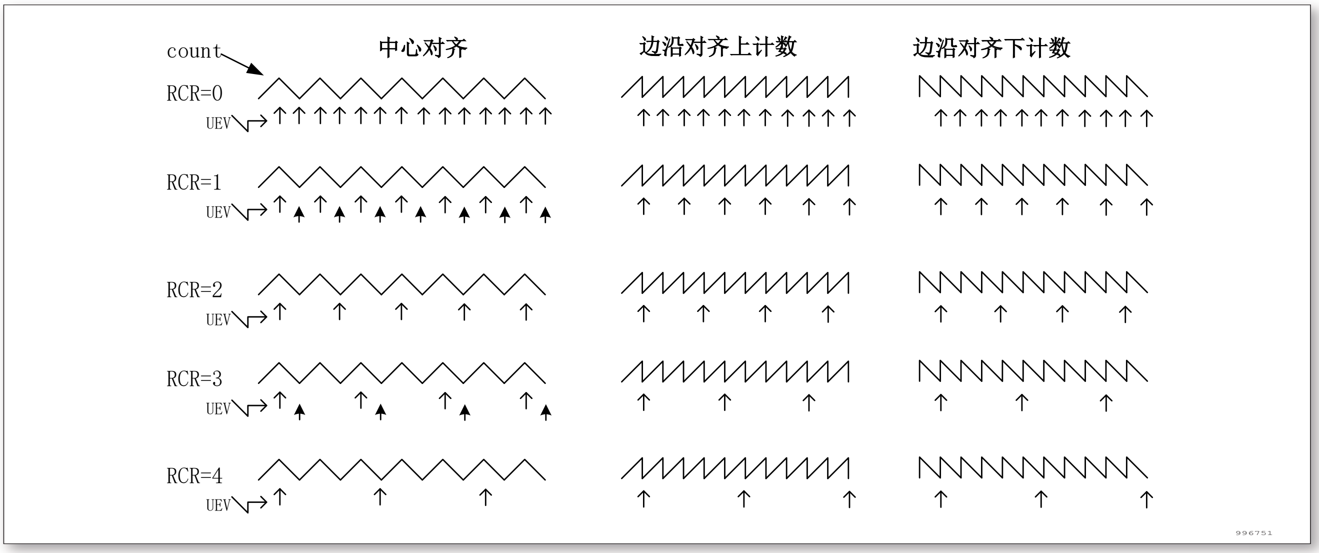

图 67. 重复计数器产生更新时序

除了上下溢出通过重复计数器可以产生事件更新 UEV 外，还可以通过写寄存器 CR.UG 产生软件及从模式复位事件更新 UEV；这时需要配置 CR.URS。

数据缓存

自动重载数据 ARR 与比较寄存器都可以配置缓存功能，当缓存功能有效时，当发生 UEV 事件更新时，写入的周期值 ARR 与比较值 CCR 才会生效。

自动重载值在不同计数模式下的更新时序图如下：

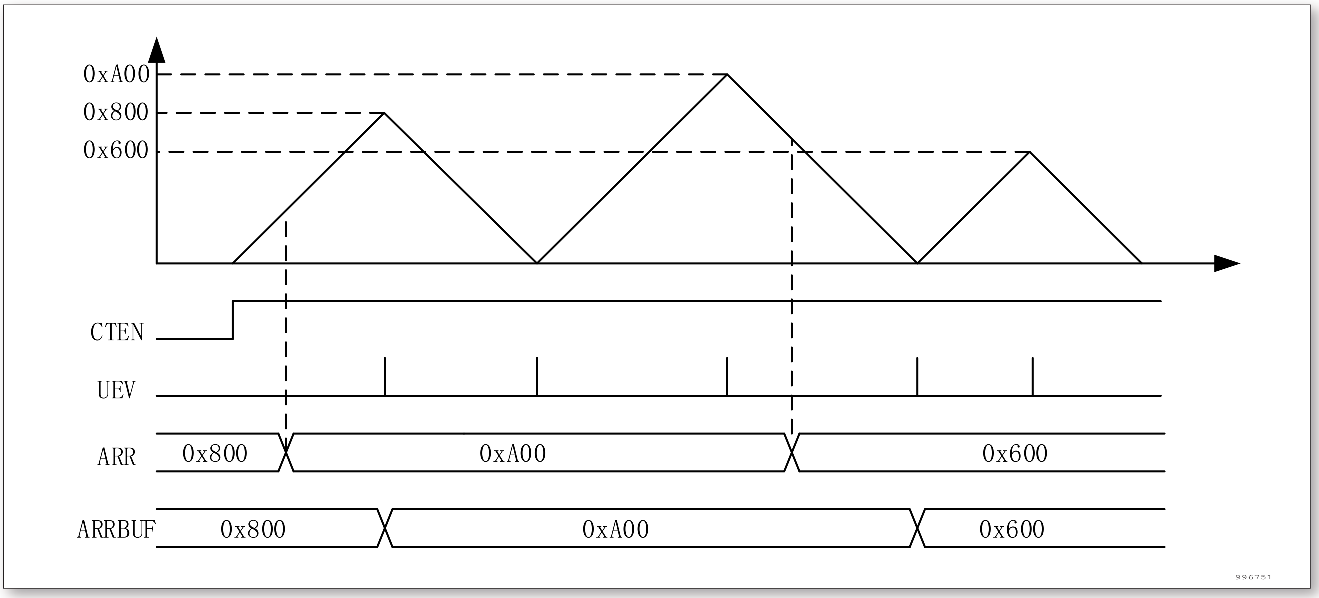

图 68. 三角波模式下缓存使能

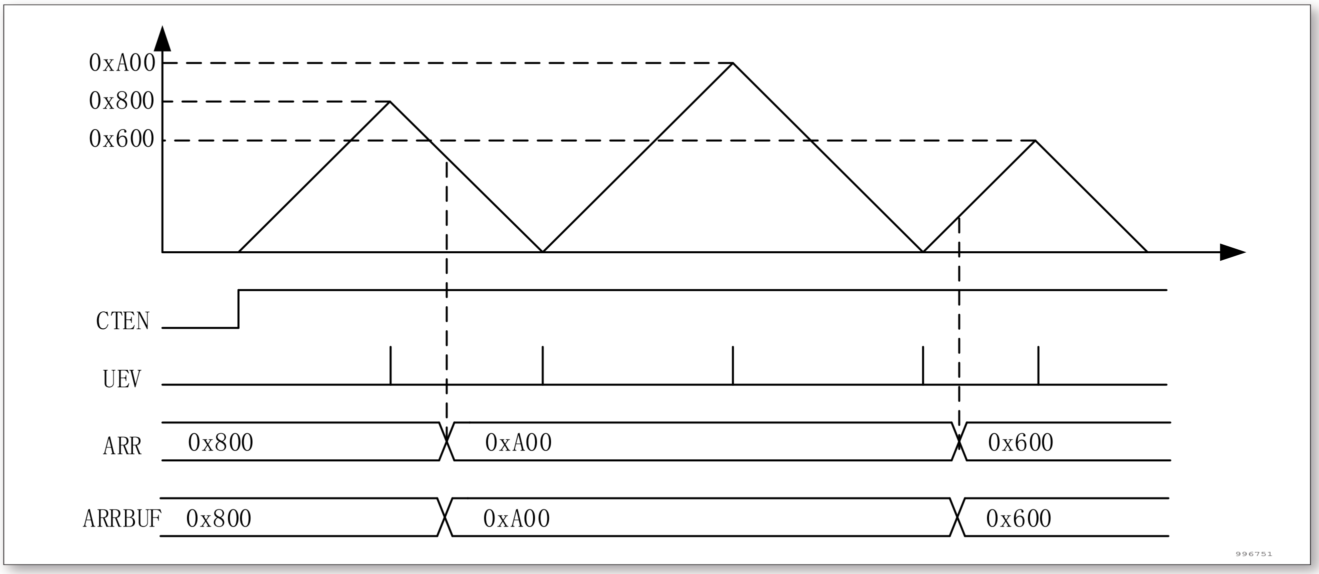

图 69. 三角波模式下缓存无效

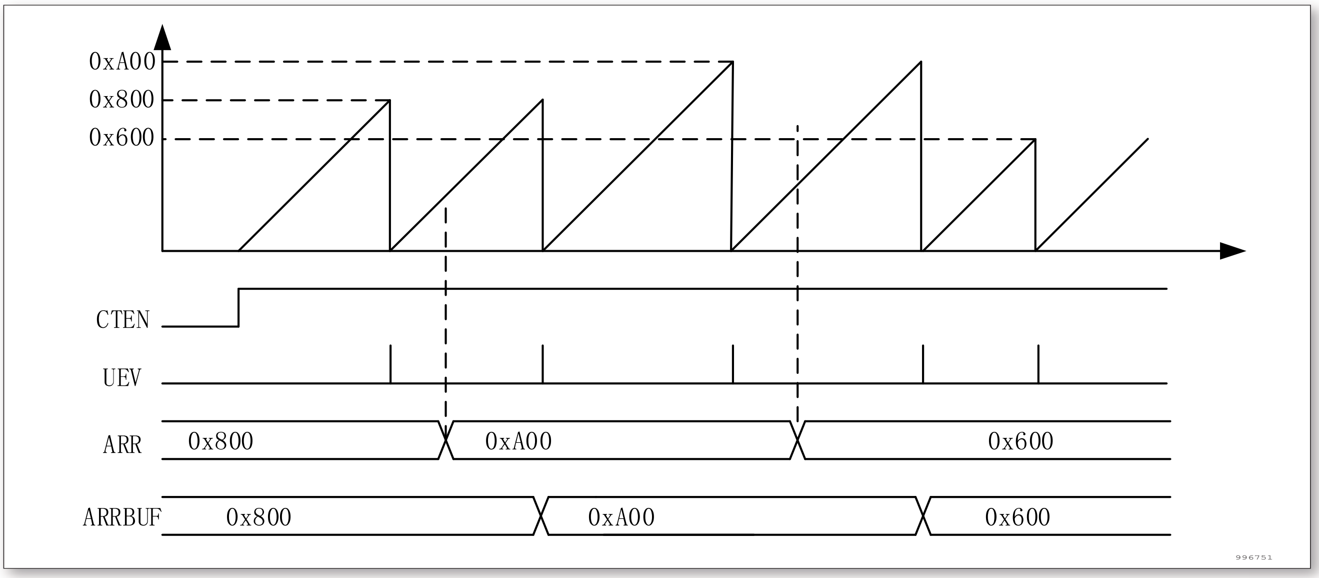

图 70. 锯齿波模式下上计数缓存使能

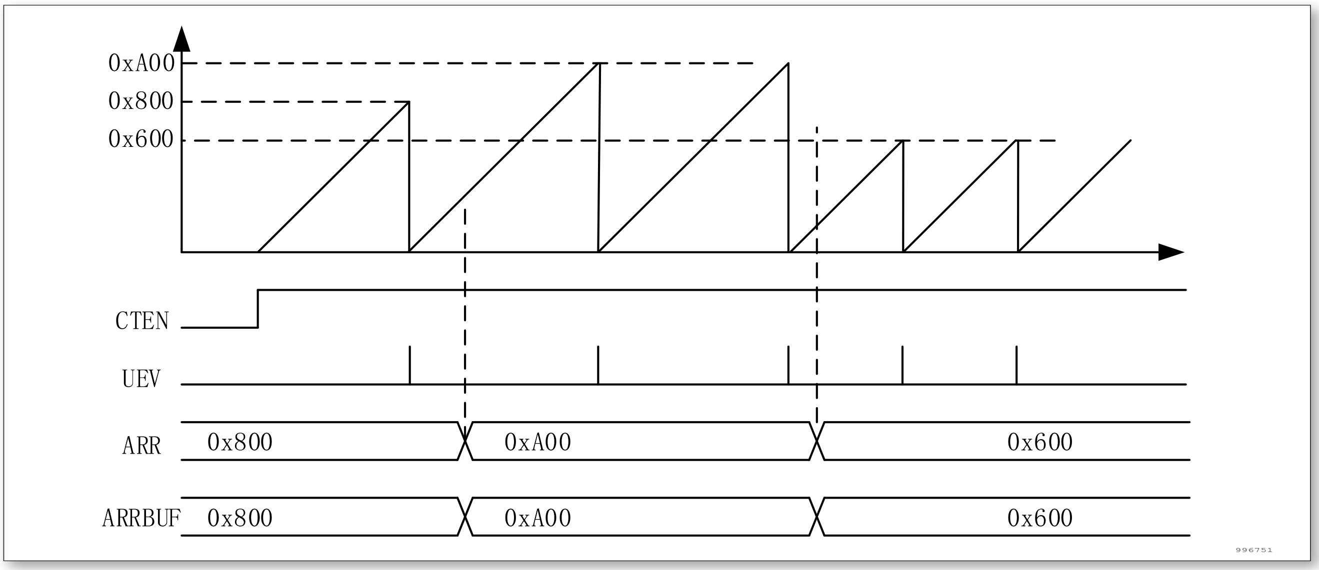

图 71. 锯齿波模式下上计数缓存无效

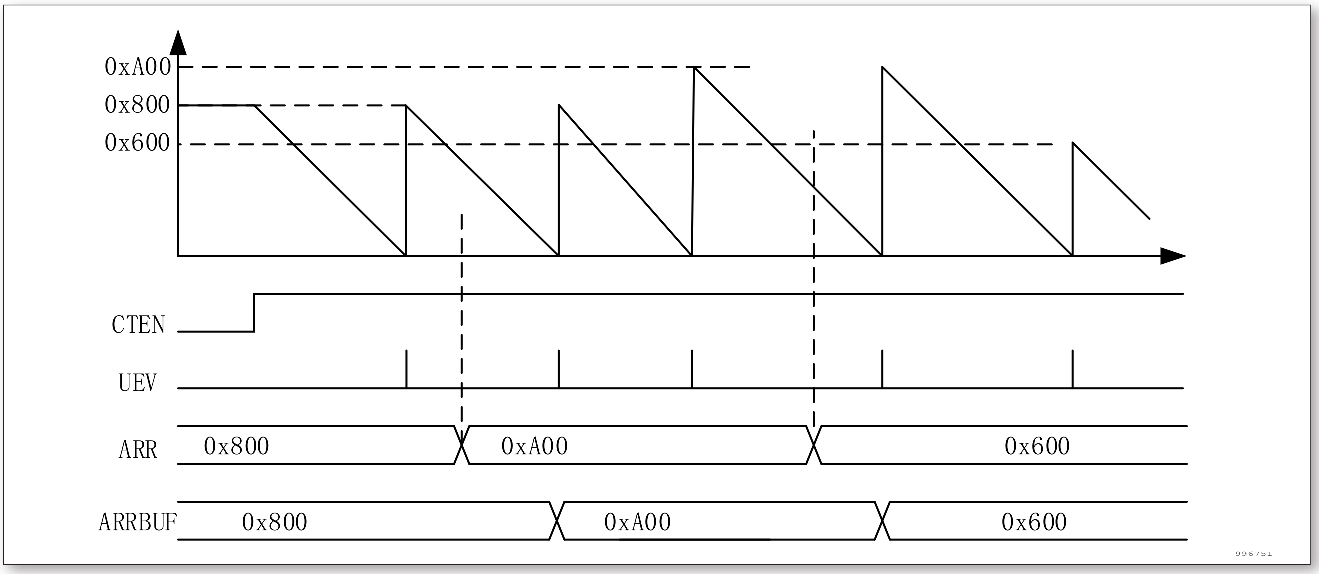

图 72. 锯齿波模式下计数缓存使能

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

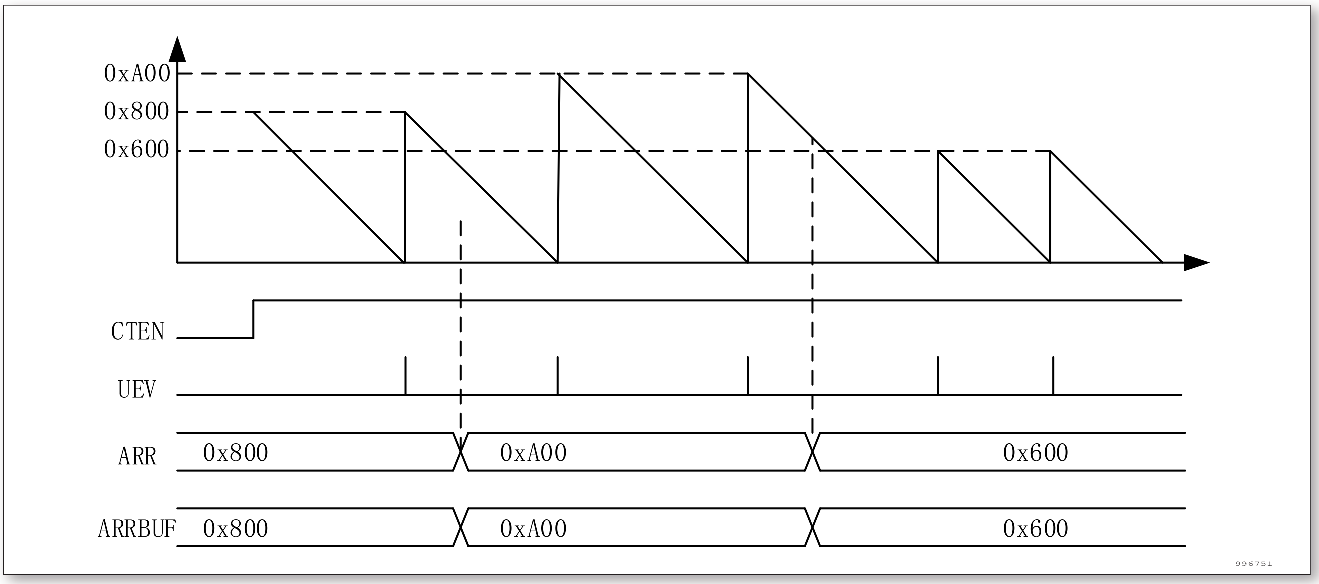

图 73. 锯齿波模式下计数缓存无效

在三角波模式与锯齿波上计数模式时，如果缓存不使能，更改的 ARR 时，当前计数器的值要小于要更改的 ARR 周期值，否则当前周期会计数到 0XFFFF。

比较缓存与周期缓存更新状态一致，这里不一一列出时序图。

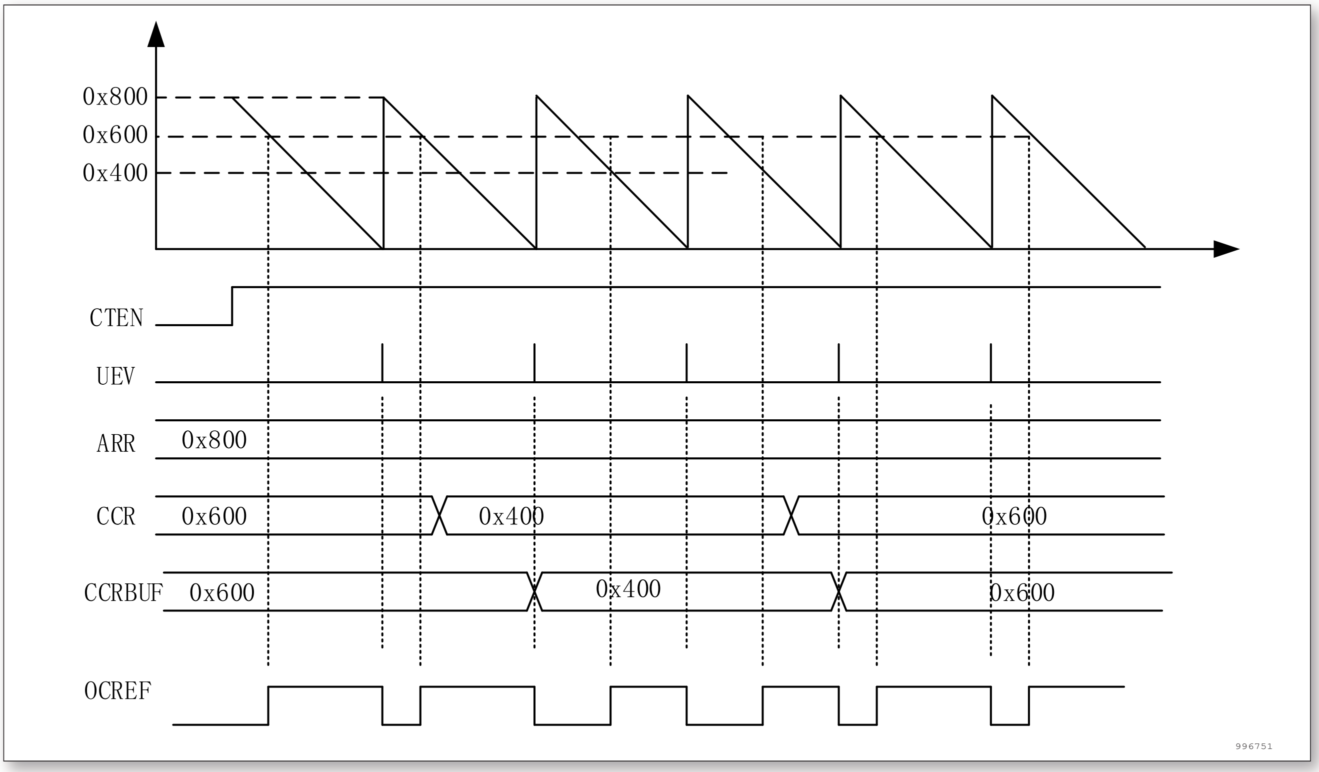

图 74. 锯齿波模式下计数比较缓存使能

比较输出 OCREF

比较输出 OCREFA 可以配置为单点比较，使用比较寄存器 CCRA 控制 OCREFA 的输出；OCREFA 的比较输出也可以配置为双点比较，使用比较寄存器 CCRA,CCRB 一起控制 OCREFA 的比较输出。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

OCREFB 的比较输出只能使用单点比较，使用比较寄存器 CCRB 控制 OCREFB 的比较输出。

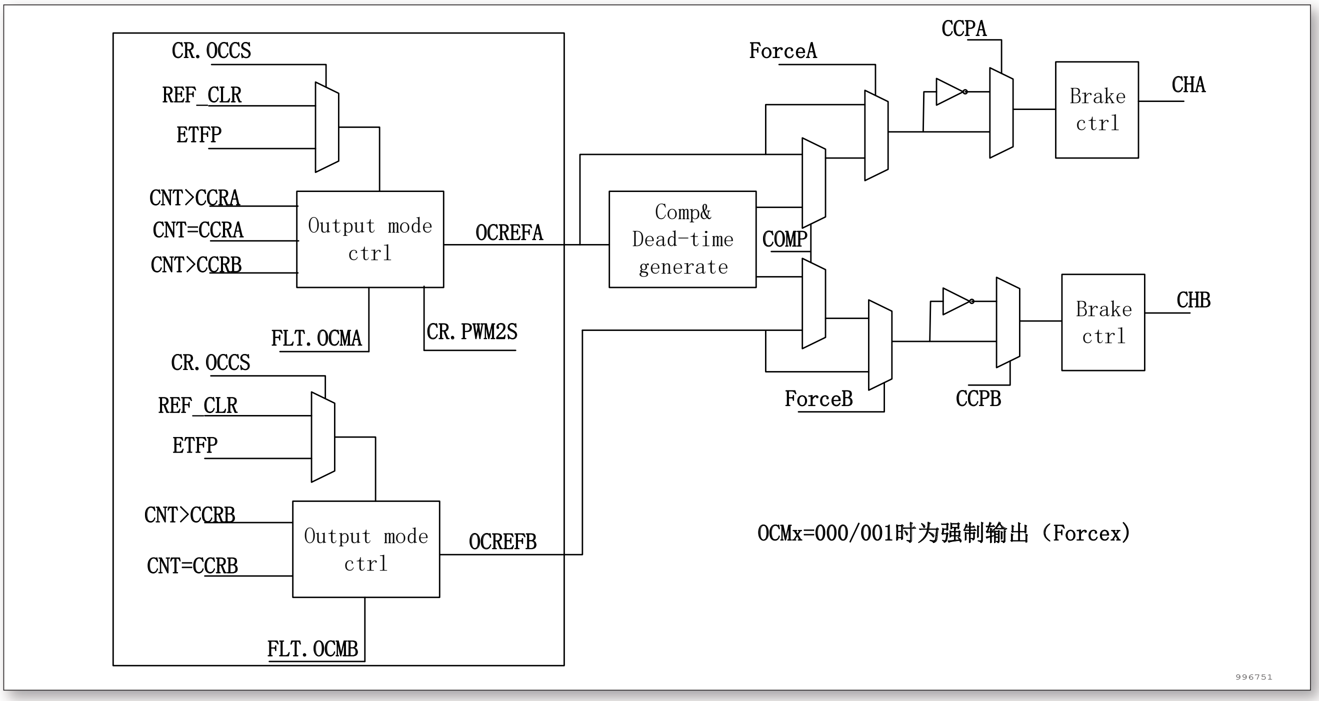

图 75. OCREF 输出框图

OCREF 输出使用 OCMx 选择。

- 000：强制为 0；
- 001：强制为 1；
- 010：比较匹配时强制为 011：比较匹配时强制为 100：比较匹配时翻转；
- 101：比较匹配时输出一个计数周期的高电平；
- 110：PWM 模式 1；

  单点比较：

  上计数时 CNT<CCRxy 输出高，下计数时 CNT>CCRxy 输出为低电平

  双点比较：

  1. 锯齿波上计数 CCRxA<CNT≤CCRxB 输出为低电平；
  2. 锯齿波下计数 CCRxA<CNT≤CCRxB 输出为高电平；
  3. 三角波上计数 CNT<CCRxA 输出高，下计数 CNT>CCRxB 为低电平。
- 111：PWM 模式 2。

  单点比较：

  上计数时 CNT<CCRxy 输出低，下计数时 CNT>CCRxy 输出为高电平

  双点比较：

  1. 锯齿波上计数 CCRxA≤CNT<CCRxB 输出为高电平；
  2. 锯齿波下计数 CCRxA≤CNT<CCRxB 输出为低电平；
  3. 三角波上计数 CNT<CCRxA 输出低，下计数 CNT>CCRxB 为高电平。

注：强制输出有高优先级，当强制输出有效时，互补输出控制无效。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

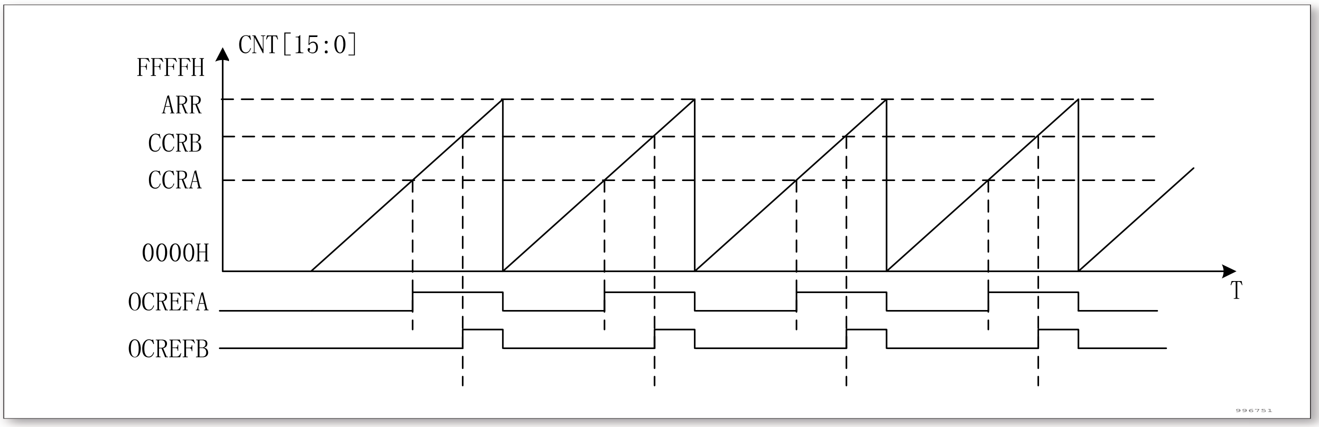

图 76. 锯齿波计数单点比较 OCREF 输出波形 (OCMx=111)

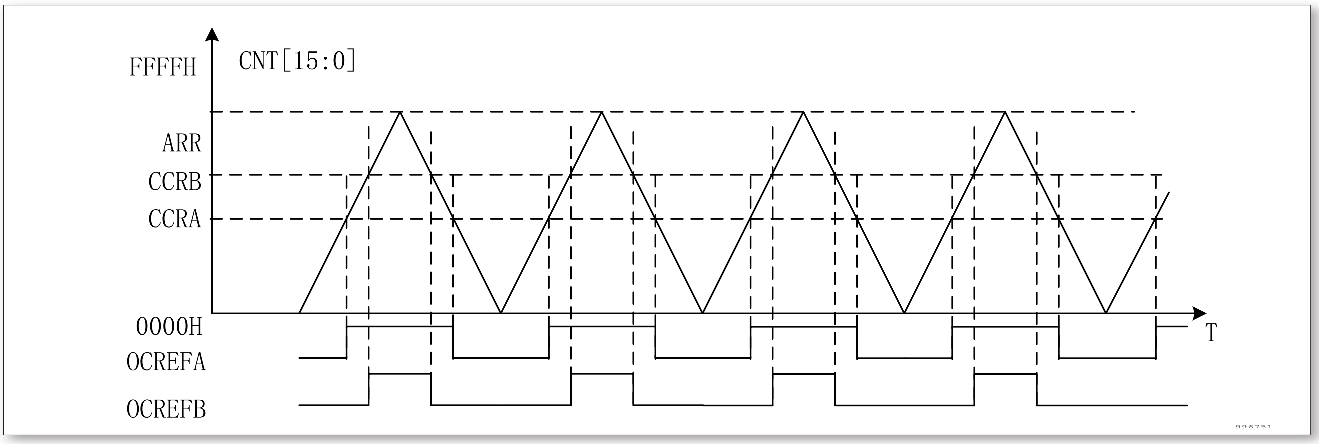

图 77. 三角波计数单点比较 OCREF 输出波形 (OCMx=111)

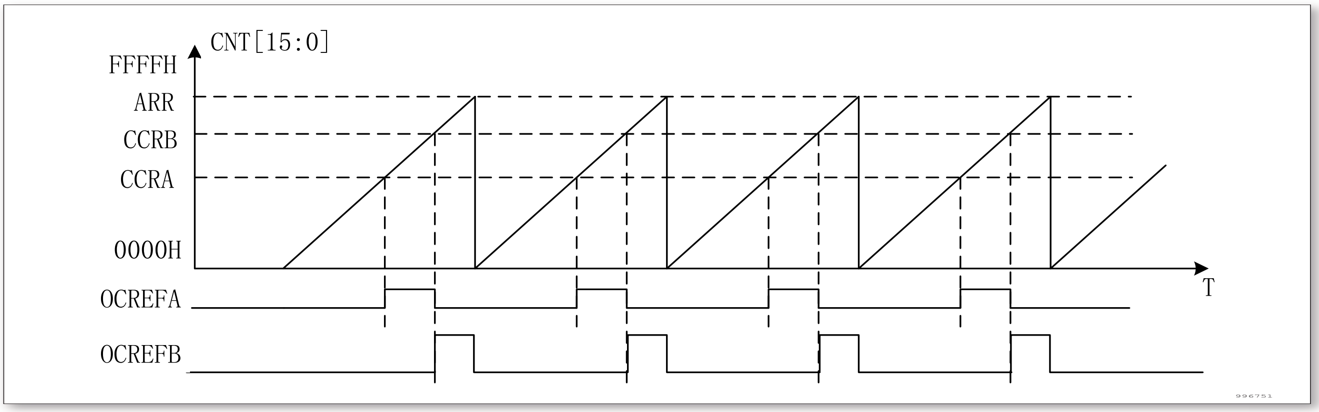

图 78. 锯齿波计数双点比较 OCREF 输出 (OCMx=111)

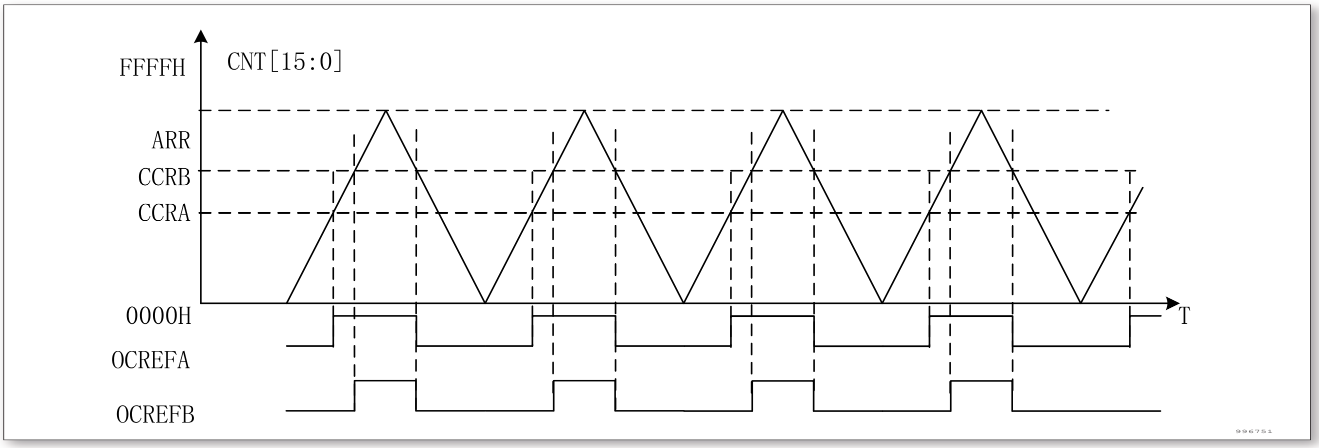

图 79. 三角波计数双点比较 OCREF 输出 (OCMx=111)

独立 PWM 输出

由 OCREFA 控制 CHA 的输出，OCREFB 控制 CHB 的输出。通过 CRCHx.CCPA, CRCHx.CCPB 可以控制 CHA，CHB 输出的反向。

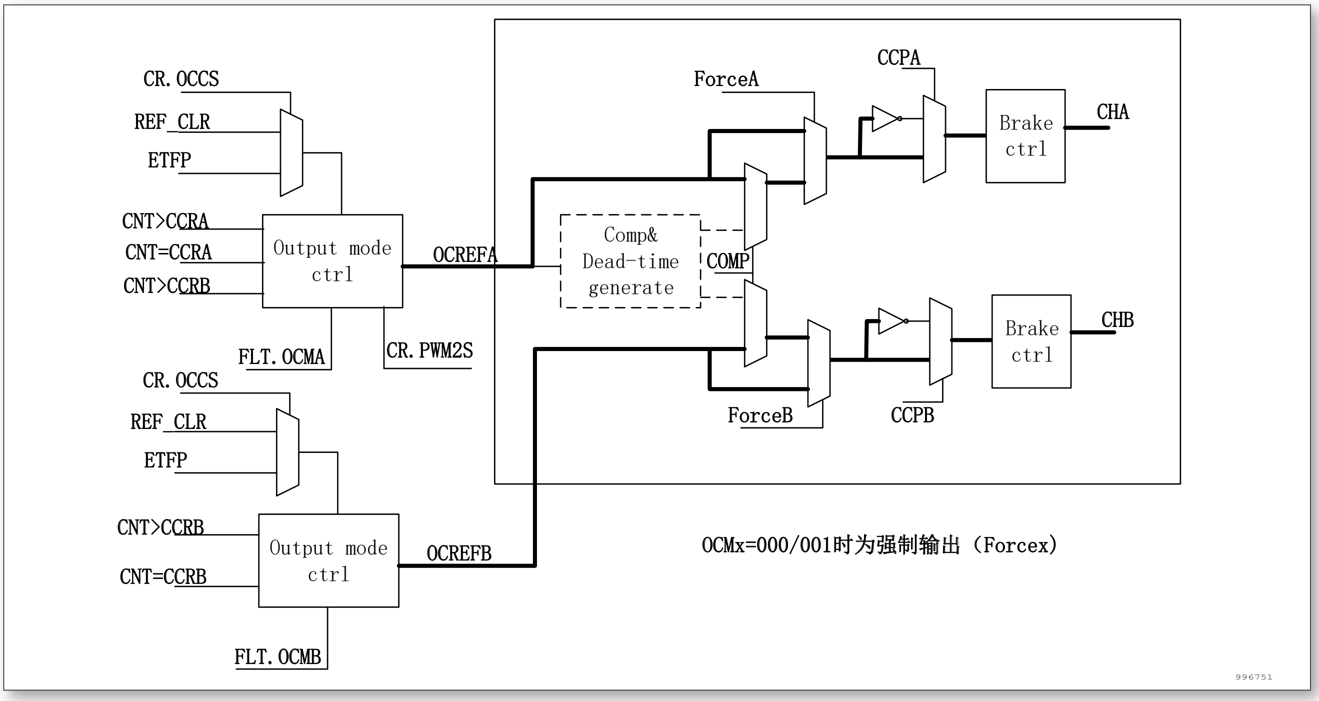

图 80. 独立 PWM 输出框图

PWM 输出与 OCREF 关系

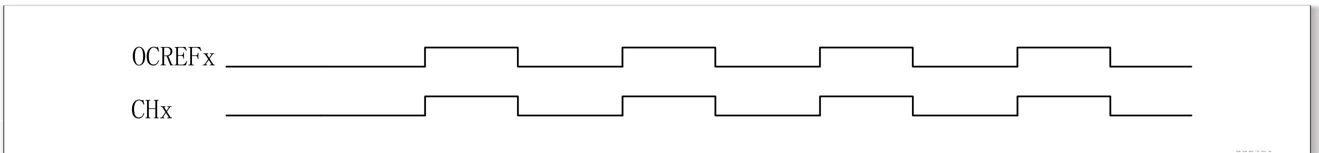

图 81. CCPx=0 时 PWM 输出波形

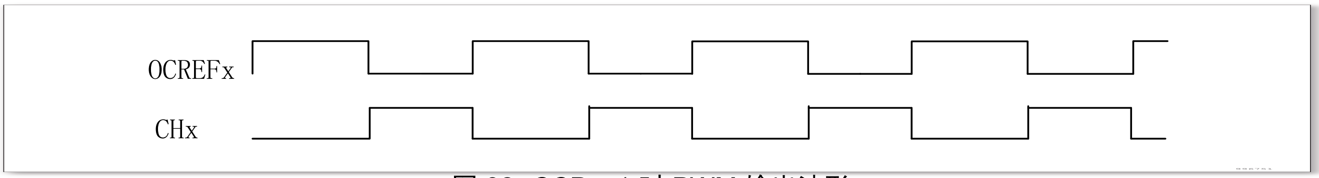

图 82. CCPx=1 时 PWM 输出波形

互补 PWM 输出

由 OCREFA 控制 CHA 的输出，OCREFA 同时控制 CHB 的输出。比较寄存器 CCRxB 可以作为专用比较控制 ADC 触发。

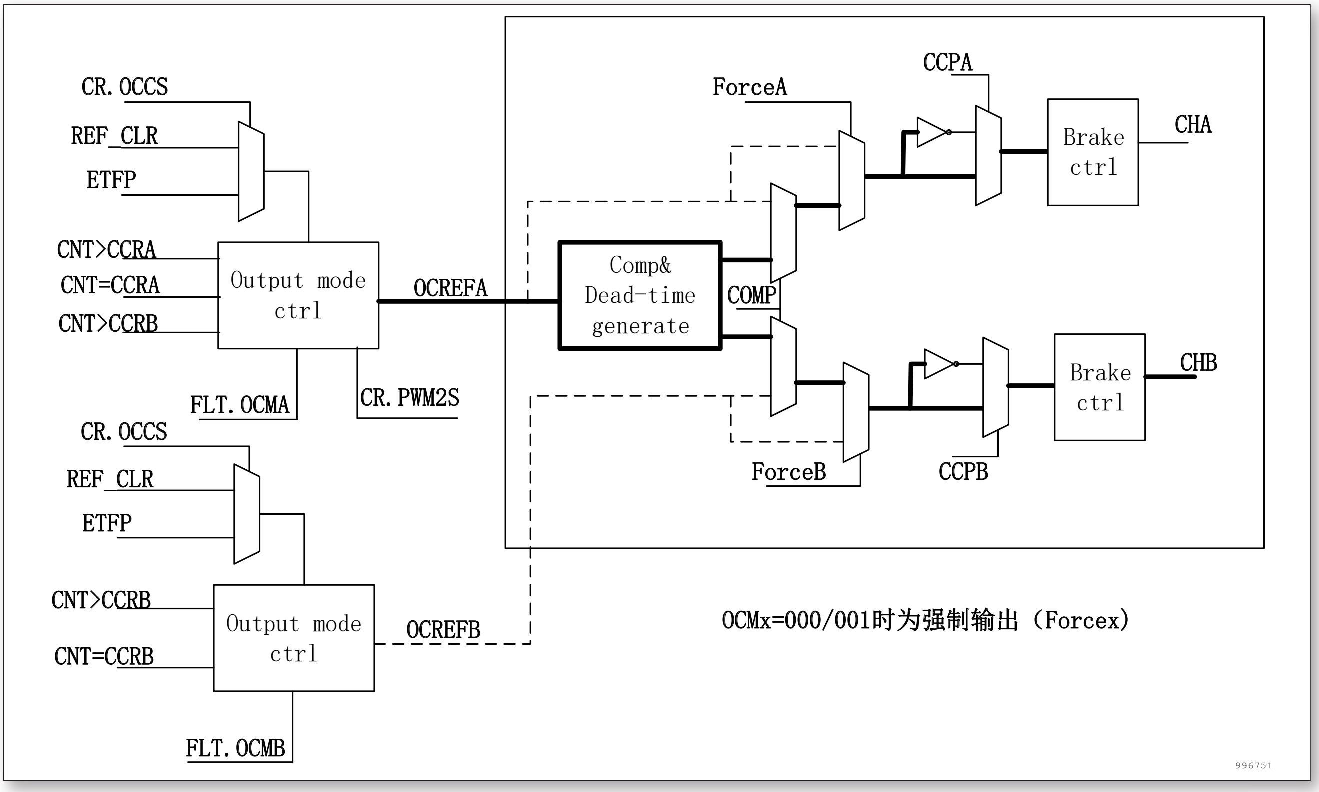

图 83. 互补 PWM 输出框图

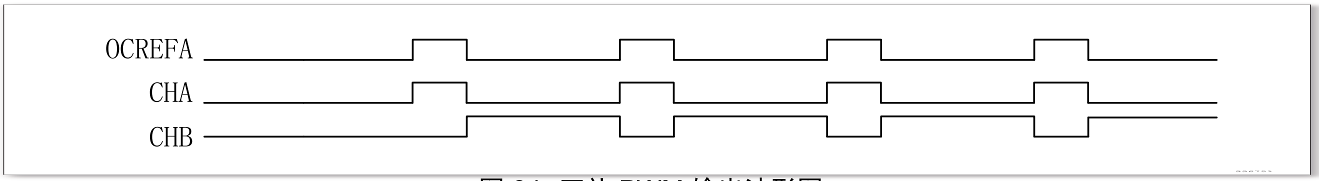

图 84. 互补 PWM 输出波形图

有死区的 PWM 输出

在互补 PWM 输出模式下可以设置死区功能。

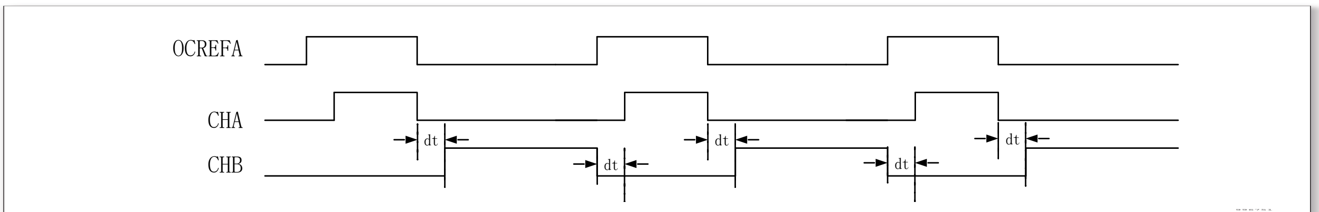

图 85. 互补 PWM 输出波形图

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

死区时间使用 8 位 DTR 控制，死区时间 dt 与 DTR 的关系如下

DTR[7] =0	T=DTR[6:0]+2	2-129	step=1
DTR[7:6] =10	T=DTR[5:0]+64*2 +2	130-256	step=2
DTR[7:5] =110	T=DTR[4:0]+32*8 +2	258-506	step=8
DTR[7:5] =111	T=DTR[4:0]+32*16 +2	514-1010	step=16

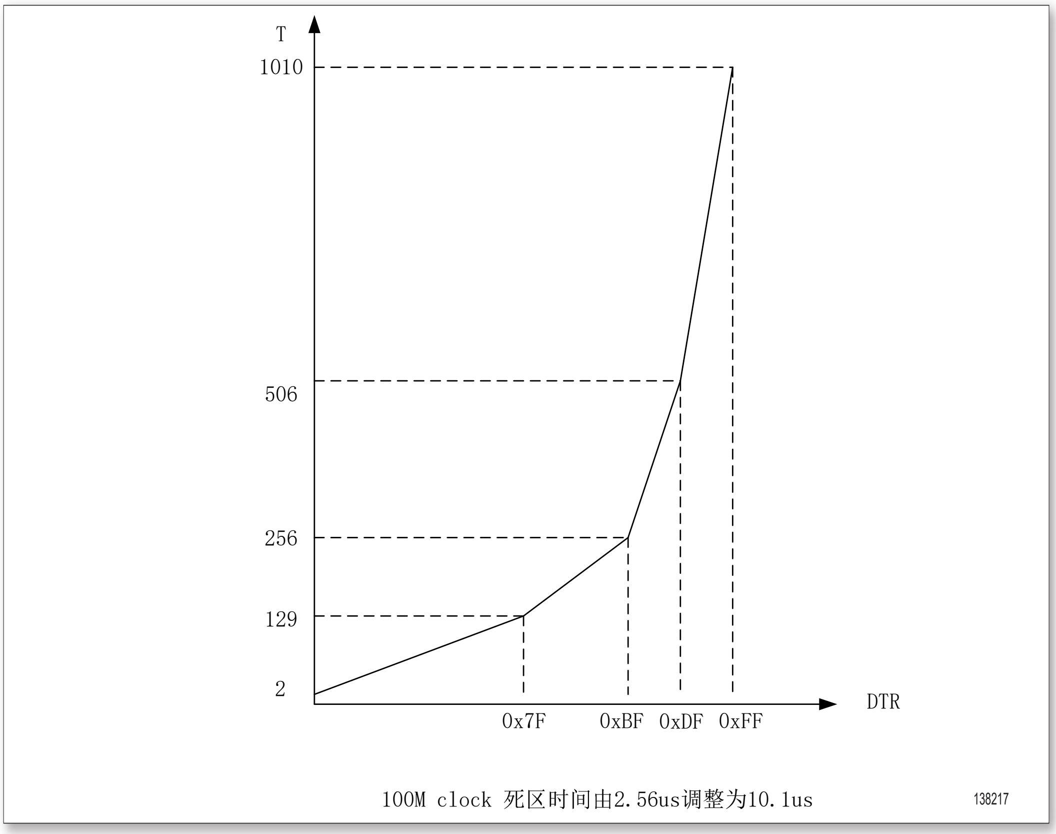

图 86. 死区时间

单脉冲输出

单脉冲模式 (ONE SHOT) 是上述 PWM 模式的一个特例。在这种模式下，计数器可以在一个激励信号的触发下启动，并可在一段可编程的延时后产生一个脉宽可编程的脉冲。

可以通过从模式控制器启动计数器。可以在输出比较模式或 PWM 模式下生成波形。

将 TIMx_M23CR 寄存器中的 ONESHOT 位置 1，即可选择单脉冲模式。这样，发生下一更新事件 UEV 时，计数器将自动停止。

单脉冲模式在锯齿波下计数模式计数器初值不要设置为 0, 上计数模式计数值不要设置到大于等于 ARR。

只有当比较值与计数器初始值不同时，才能正确产生输出脉冲。启动前（定时器等待触发时），必须进行如下配置：

- 递增计数时：CNT<CCRxy□ ARR（特别注意，0<CCRxy）；
- 递减计数时：CNT>CCRxy。

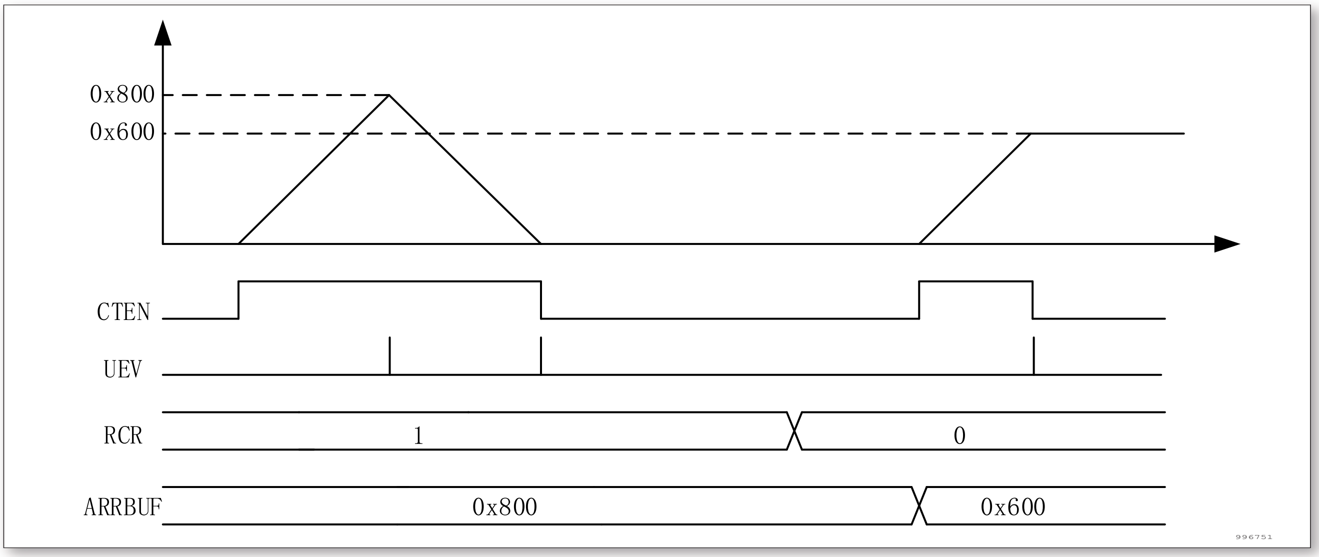

图 87. 三角波模式单脉冲计数

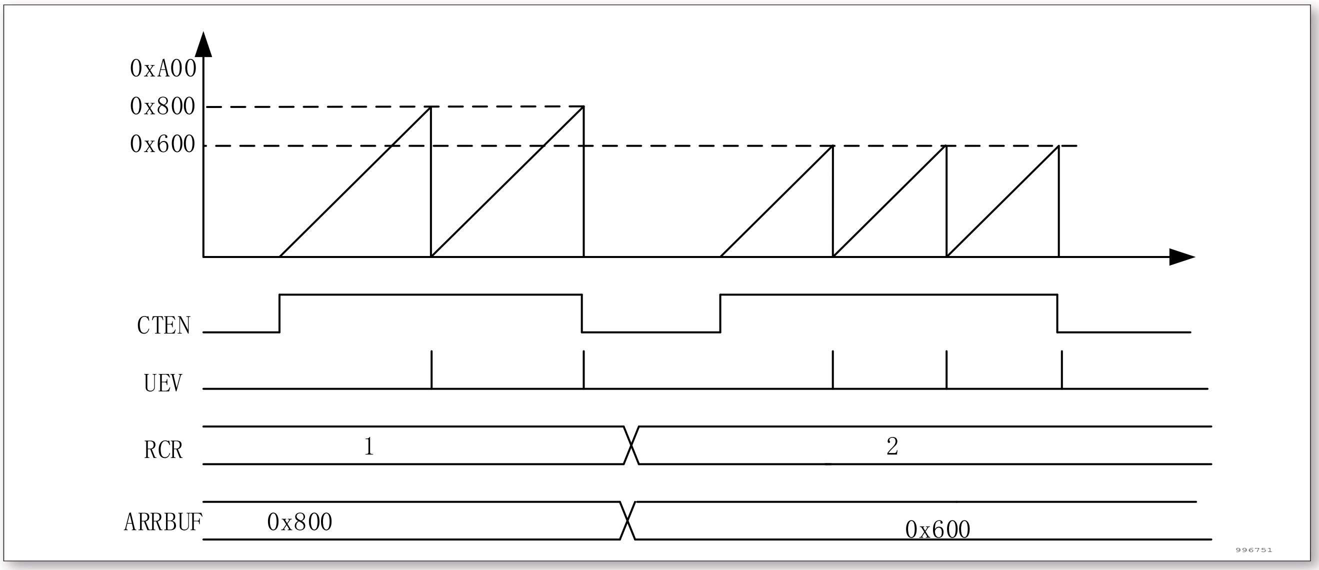

图 88. 锯齿波上计数单脉冲模式

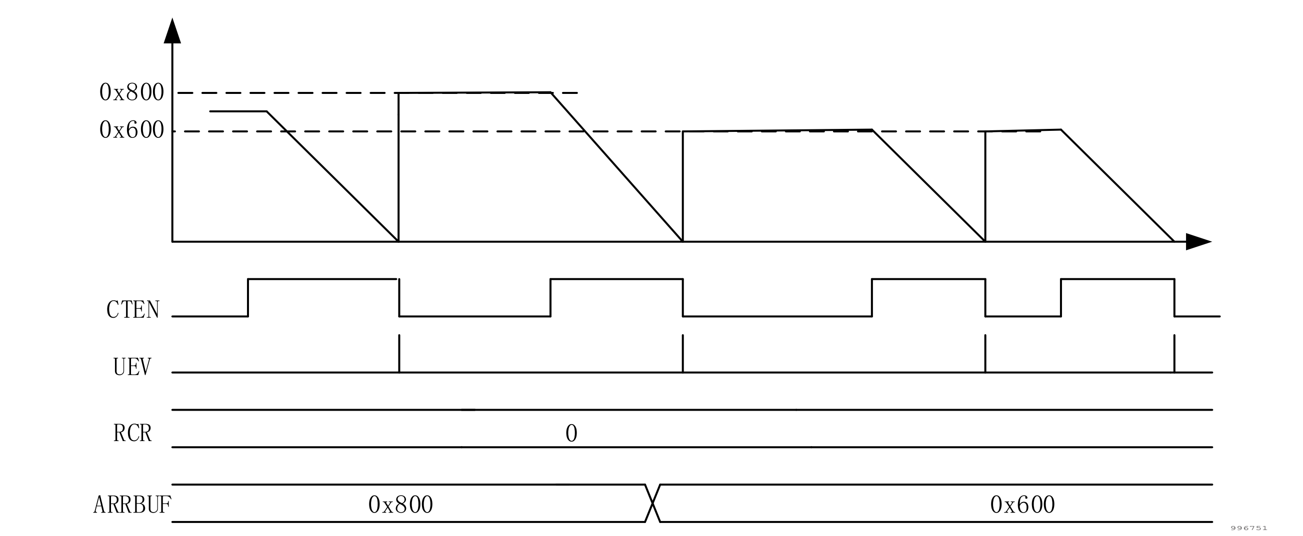

图 89. 锯齿波下计数单脉冲模式

比较中断

锯齿波比较匹配会将 IFR 寄存器相应的标志置 1，如果中断使能 CRCHx.CIEy (x=0,1,2;y=A,B) 将会触发中断。

三角波比较匹配可单独选择上升计数比较匹配，下降计数比较匹配或者两者比较都匹配。

比较 A 比较匹配当计数值与 CCRxA 相等时，统一使用 CR.CIS 控制。

- CR.CIS=2’b00 时比较匹配无输出；
- CR.CIS=2’b01 时上计数时比较匹配；
- CR.CIS=2’b10 时下计数时比较匹配；
- CR.CIS=2’b11 时上下计数时比较都匹配。

比较 B 比较匹配当计数值与 CCRxB 相等时，不同通道可以使用 CRCHx.CISB 单独控制。

- CRCHx.CISB=2’b00 时通道 x 比较匹配无输出；
- CRCHx.CISB=2’b01 时通道 x 上计数时比较匹配；
- CRCHx.CISB=2’b10 时通道 x 下计数时比较匹配；
- CRCHx.CISB=2’b11 时通道 x 上下计数时比较都匹配。

B 通道比较匹配单独控制为了更灵活的触发 ADC。详见 Timer 触发 ADC 章节。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

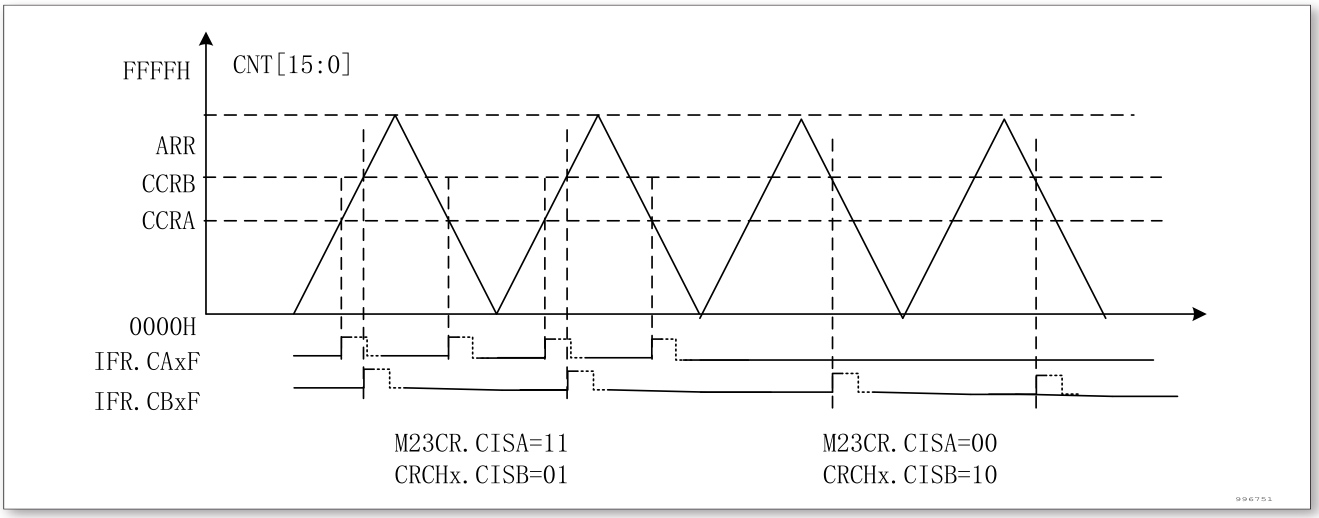

图 90. 中断示意图

捕获输入

CR.MODE=2/3

在三角波计数或者锯齿波计数模式下都可以设置捕获功能，可以设置捕获的电平边沿，当发生捕获时捕获的值存入比较捕获寄存器并产出捕获中断。

每个通道的比较捕获功能可以单独设置，通过寄存器 CRCHx.CSA/CSB 选择。

每个通道的捕获边沿可以单独设置，通过寄存器 CRCHx.CFy/CRy(x=0/1/2;y=A/B) 选择捕获触发的边沿。

当捕获发生后捕获标志未清除前再次发生捕获动作，会产生捕获数据覆盖标志。

定时器未启动时如果有效的捕获边沿也会产生捕获标志及捕获动作。

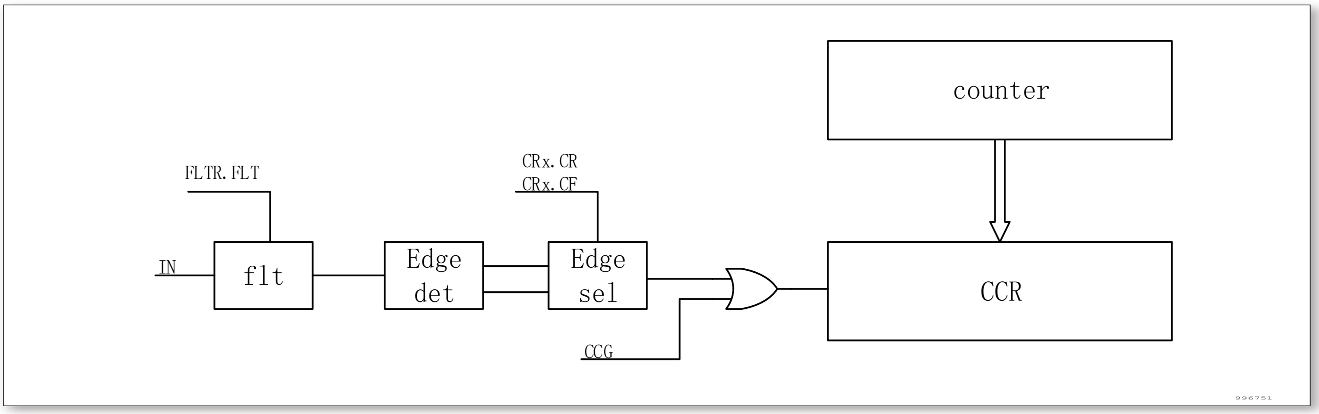

图 91. 捕获功能框图

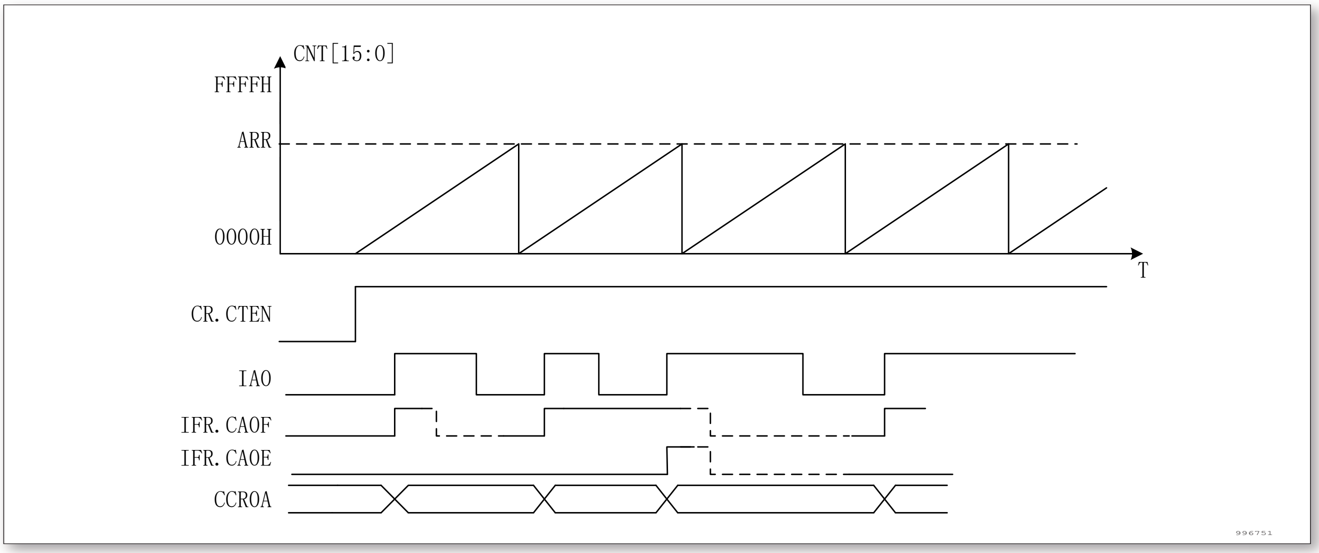

图 92. 捕获时序图

CH0A 的捕获输入通过 MSCR.IA0S 选择 CH0A 输入还是 CH0A，CH1A，CH2A 的异或输入。

表 48: MSCR.IA0S 配置

IA0S	0	1
Timer0/1/2	CHA0	CHA0 ETR GATE 异或输入
Timer3	CHA0	CHA0 CHA CHA2 异或输入

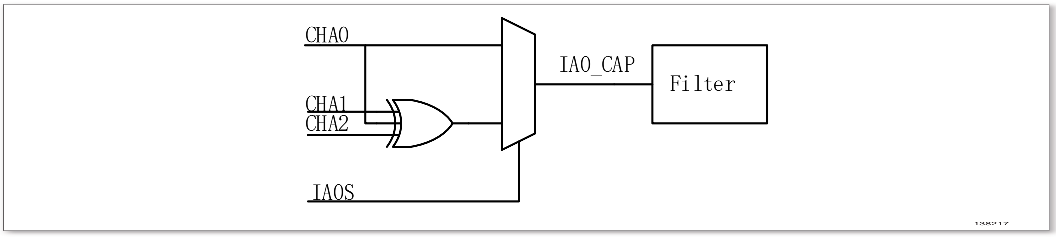

图 93. CHA 端口选择

CH0B 的捕获输入通过 MSCR. IB0S 选择 CH0B 输入还是内容 MSCR.TS 选择的信号。

表 49: MSCR.IB0S 配置

CH0B	0	1
Timer0/1/2/3	CHB0	内部触发 MSCR.TS 选择信号, 详见表 50

表 50: MCSCR.TS 配置

MCSCR.TS	通道 B 捕获输入 0
000	ETR 滤波相位选择输出信号，滤波相位选择可配
001	内部互联 ITR0
010	内部互联 ITR1
011	内部互联 ITR2

MCSCR.TS	通道 B 捕获输入 0
100	内部互联 ITR3
101	CH0A 边沿
110	CH0A 滤波输出信号，滤波功能可配
111	CH0B 滤波输出信号，滤波功能可配

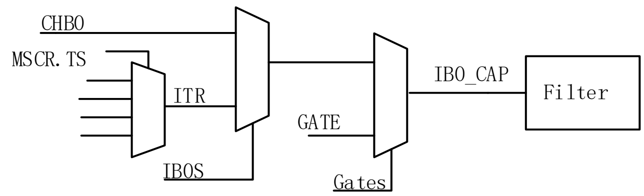

图 94. CHB 端口选择

Gates 信号为选择 PWM 互补输出功能生成的信号，当 PWM 互补输出时，比较捕获寄存器 CCR0B 没有使用，自动切换到使用 GATE 作为捕获/比较输出。

设置示例

- 边沿对齐独立 PWM 输出设置；
  1. 设置模式 CR.MODE=2；
  2. 设置锯齿波计数方向 CR.DIR；
  3. 设置 PWM 周期值 ARR；
  4. 设置计数器初值（初值必须小于周期值）；
  5. 设置 PWM 比较值 CCRxA,CCRxB；
  6. 设置 PWM 输出比较模式 FLT.OCMA FLT.OCMB 为 6 或 7；
  7. 清除相关中断标志；
  8. 使能相应的中断；
  9. 设置输出极性 FLT.CCPAx FLT.CCPBx；
  10. 使能输出 DTR.MOE；
  11. 使能定时器 CR.CTEN。
- 中心对齐互补 PWM 输出设置；
  1. 设置模式 CR.MODE=3；
  2. 设置互补输出 CR.COMP；
  3. 设置 PWM 周期值 ARR；
  4. 设置计数器初值（初值必须小于周期值）；
  5. 设置 PWM 比较值 CCRxA(CCRxB 不需要设置，PWM 输出与这个寄存器无关)；
  6. 设置 PWM 输出比较模式 FLT.OCMA FLT.OCMB 为 6 或 7；
  7. 清除相关中断标志；
  8. 使能相应的中断；
  9. 设置输出极性 FLT.CCPAx FLT.CCPBx；
  10. 使能输出 DTR.MOE；
  11. 使能定时器 CR.CTEN。
- 三角波非中心对齐带死区互补 PWM 输出设置；
  1. 设置模式 CR.MODE=3；

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

2. 设置互补输出 CR.COMP；
3. 设置两个比较使能 CR.PWM2S；
4. 设置 PWM 周期值 ARR；
5. 设置计数器初值（初值必须小于周期值）；
6. 设置 PWM 比较值 CCRxA,CCRxB, 上计数比较点为 CCRxA, 下计数比较点为 CCRxB；
7. 设置 PWM 输出比较模式 FLT.OCMA FLT.OCMB 为 6 或 7；
8. 清除相关中断标志；
9. 使能相应的中断；
10. 设置输出极性 FLT.CCPAx FLT.CCPBx；
11. 设置死区使能 DTR.DTEN；
12. 设置死区时间 DTR.DT；
13. 使能输出 DTR.MOE；
14. 使能定时器 CR.CTEN。

- 捕获功能设置；

  设置 CH0B 为上升沿捕获功能

  1. 选择计数器计数方式，设置 mode=2；
  2. 选择 CH0B 为捕获模式 CRCHx.CSB=1；
  3. 根据需要选择输入滤波 FLT.FLTB0；
  4. 选择捕获的边沿 CRCHx.CRB=1；
  5. 设置 ARR 改变周期值；
  6. 启动定时器 CR.CTEN；
  7. 清除相关中断标志；
  8. 清除捕获标志，使能相应中断；
  9. 查询到捕获标志后，读取 CCR0B 获得捕获值。

注：定时器未启动时如果有效的捕获边沿也会产生捕获标志及捕获动作。

- 互补 PWM 输出时使用 GATE 作为 PWM 输出功能；
  1. 设置为 PWM 互补输出模式，参考“中心对齐互补 PWM 输出设置”；
  2. 设置 CCR0B 设置 GATE PWM 比较值；
  3. 设置 FLT.OCMB0=6/7 设置 PWM 输出模式；
  4. 使能 PWM 输出 DTR.MOE。
- 互补 PWM 输出时使用 GATE 下降沿作为捕获输入功能。
  1. 设置为 PWM 互补输出模式，参考“中心对齐互补 PWM 输出设置”；
  2. 设置 GATE 作为捕获输入 CR.CSG；
  3. 选择 GATE 捕获输入边沿设置 CR.CFG(GATE 作为捕获输入 B 通道滤波设置无效)；
  4. 启动定时器 CR.CTEN；
  5. 清除相关中断标志；
  6. 清除捕获标志，使能相应中断；
  7. 查询到捕获标志后，读取 CCR0B 获得捕获值。

15.2.7 模式 2/3 从模式

定时器能够在多种模式下和一个外部的触发同步：复位模式、门控模式和触发模式。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

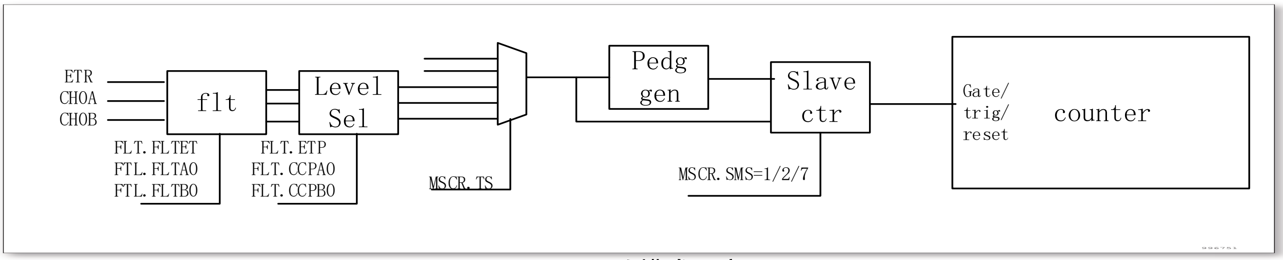

图 95. 从模式示意图

从模式功能选择通过 MSCR.SMS 选择；

表 51: MSCR.SMS 配置

MSCR.SMS	从模式功能选择
001	复位功能
010	触发模式
111	门控功能

表 52: MSCR.TS 配置

MSCR.TS	触发选择
000	端口 ETR 的滤波相位选择后的信号 ETFP，滤波功能可配置
001	内部互联信号 ITR0
010	内部互联信号 ITR1;
011	内部互联信号 ITR2;
100	内部互联信号 ITR3;
101	端口 CH0A 的边沿信号;
110	端口 CH0A 的滤波相位选择后的信号 IAFP，滤波功能可配置
111	端口 CH0B 的滤波相位选择后的信号 IBFP，滤波功能可配置

门控计数

按照选中的输入端电平使能计数器。在如下的例子中，计数器只在 CH0A 为低电平时向上计数：

- 配置通道 CH0A 以检测 CH0A 上的低电平。配置输入滤波器带宽 (本例中，不需要滤波，所以保持 FLT.FLTA0=000)。置 FLTR 寄存器中 CCPA0=1 以确定极性 (只检测低电平)。
- 置 SMCR 寄存器中 SMS=111，配置定时器为门控模式；置 SMCR 寄存器中 TS=110，选择 CH0A 作为输入源。
- 置 CR 寄存器中 CTEN=1，启动计数器。在门控模式下，如果 CTEN=0，则计数器不能启动，不论触发输入电平如何。只要 CH0A 为低，计数器开始依据内部时钟计数，一旦 CH0A 变高则停止计数。

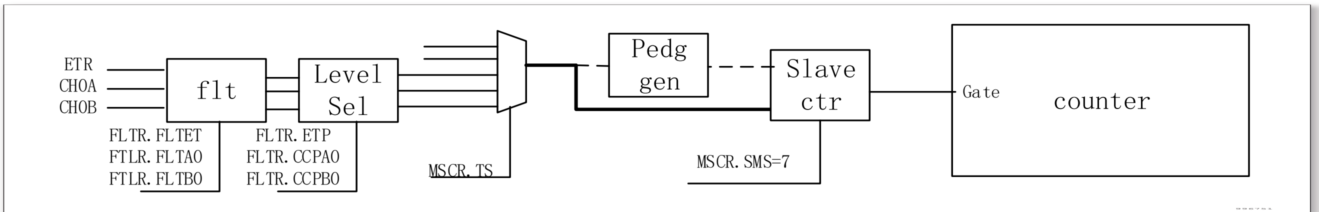

图 96. 门控功能

触发功能

使用外部触发（CH0A、CH0B、ETR）可以使定时器同步启动，使用定时器内部互联信号结合 MSCR.MSM 也可以配置定时器同步启动。触发信号为输入信号的上升沿。也可以使用软件写 CR.TG 启动软件触发功能。

输入端上选中的事件使能计数器。在下面的例子中，计数器在 CH0B 输入的上升沿开始向上计数：

- 配置通道 CH0B 检测 CH0B 的上升沿。配置输入滤波器带宽 (本例中，不需要任何滤波器，保持 FLT.FLTB0=000)。设 FLTR 寄存器中 CCPB0=0 以确定极性 (不反向)。
- 置 SMCR 寄存器中 SMS=010，配置定时器为触发模式；置 SMCR 寄存器中 TS=111，选择 CH0B 作为输入源。当 CH0B 出现一个上升沿时，计数器开始在内部时钟驱动下计数，同时设置 TIF 标志。CH0B 上升沿和计数器启动计数之间的延时，取决于 CH0B 输入端的重同步电路。

注意：如果使用下降沿触发，先选择触发极性，然后再选择模式，否则会产生误触发。

复位计数

在发生一个触发输入事件时，计数器和它的预分频器能够重新被初始化；同时，如果 CR 寄存器的 URS 位为低，还产生一个更新事件 UEV；然后所有的预装载寄存器 (ARR，CCRx) 都被更新了。

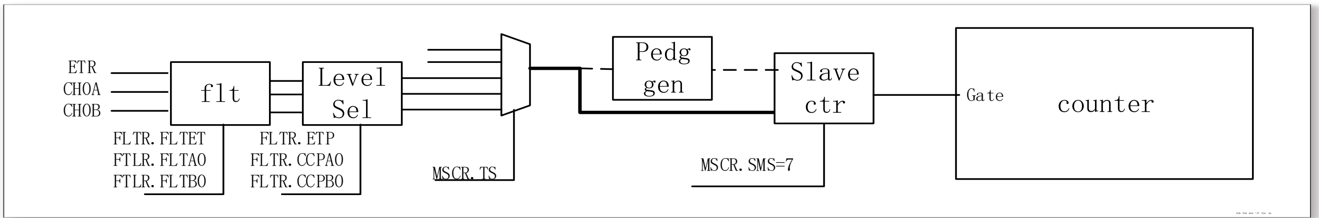

图 97. 门控功能

15.2.8 正交编码计数功能

MSCR.SMS=4/5/6，对应正交编码模式的模式 1/2/3。这时计数器根据 IAFP, IBFP 的相位关系进行编码计数。IAFP,IBFP 为端口输入 CH0A，CH0B 的滤波相位选择的信号。模式 1 使用 CH0A 的边沿计数。模式 2 使用 CH0B 的边沿计数。模式 3 使用 CH0A,CH0B 的边沿都计数。

为了保证计数相位的正确性，需要保证 A/B 输入的相位大于一个脉冲宽度的相位差，A/B 输入脉冲宽度需要大于两个脉冲宽度。

表 53: 三种模式的区别

			IAFP		IBFP	
	IBFP	IAFP	Rising	falling	Rising	falling
MOD1	High		Down	Up	-	-
	Low		Up	Down	-	-
MOD2		High	-	-	up	Down
		Low	-	-	Down	Up
MOD3	High	High	Down	Up	Up	Down
	Low	Low	up	Down	Down	Up

CHAF/CHBF 为端口 CH0A/CH0B 滤波的信号，IAFP/IBFP 为端口滤波相位选择后的信号。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

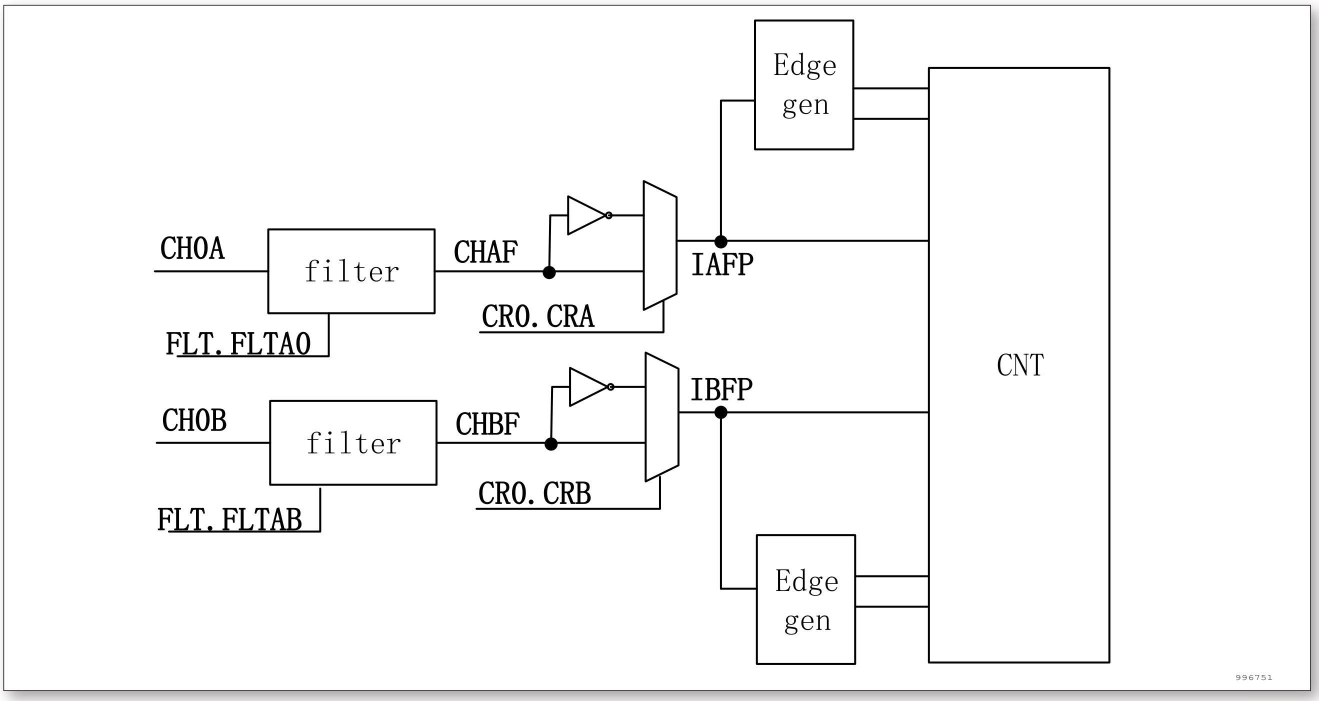

图 98. 正交编码计数器

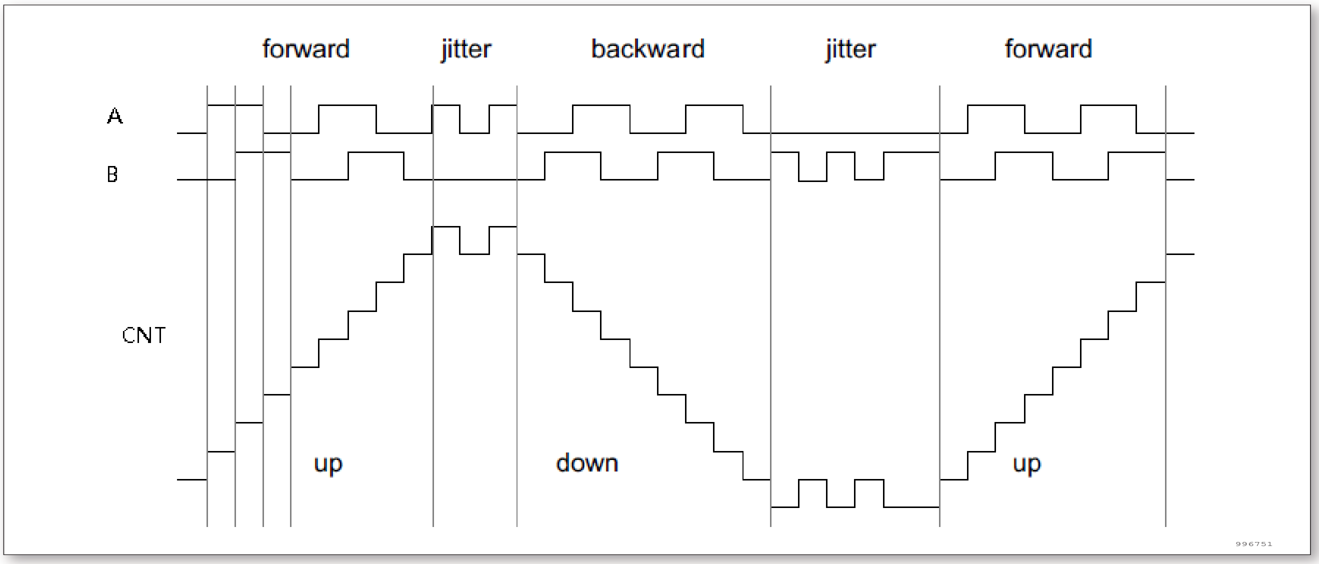

图 99. 编码计数

15.2.9 Timer 触发 ADC

CCMR0A,CCMR1A,CCMR2A 比较匹配可以配置触发 ADC，中心对齐 PWM 时只能通过控制寄存器 CR.CIS 可以选择在上升匹配触发或者下降匹配触发。

CCMR0B,CCMR1B,CCMR2B 比较匹配可以配置触发 ADC，中心对齐 PWM 时，可以通过寄存器 CRx.CISB 分别控制三个匹配触发点（上升，下降，上升下降）。

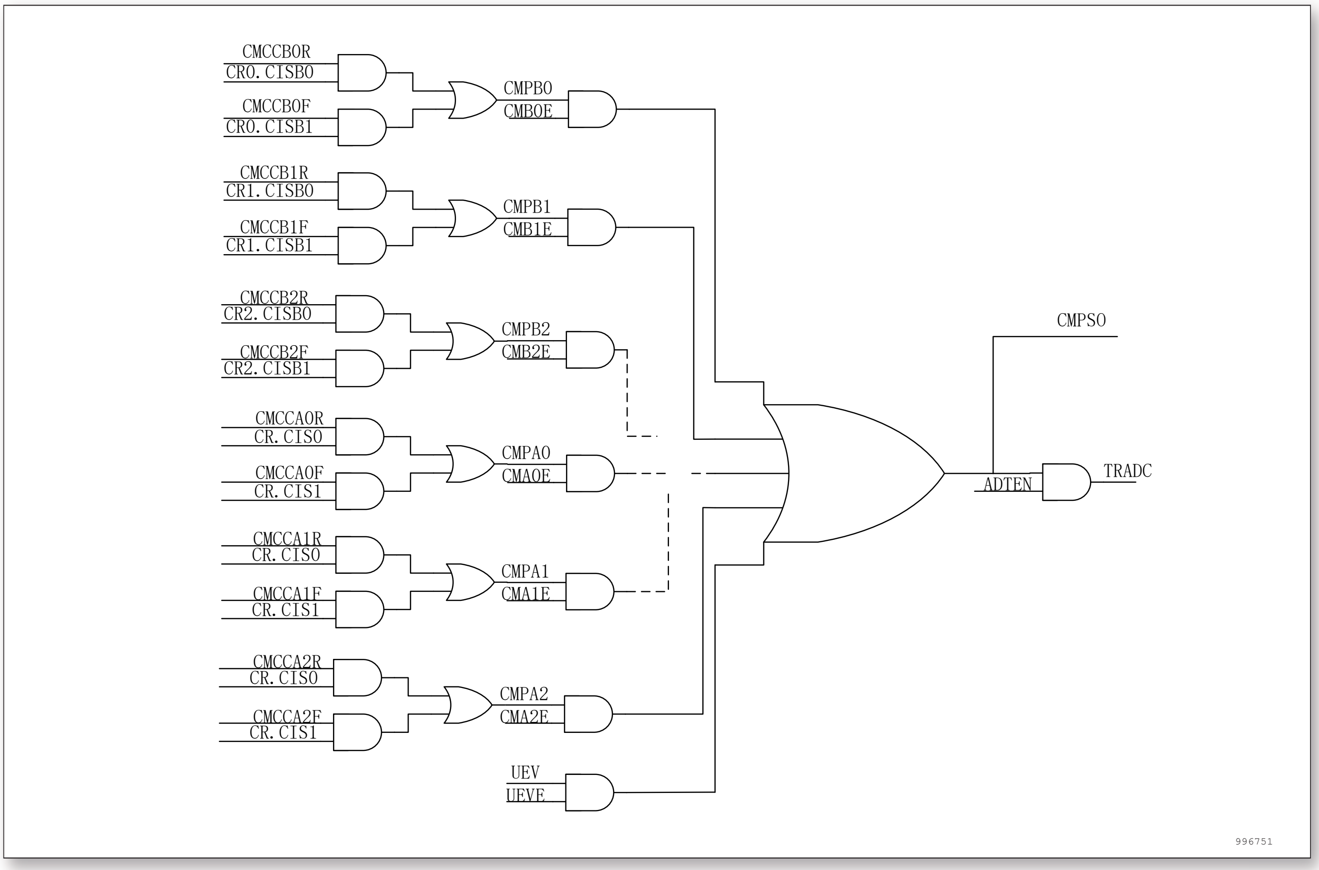

图 100. ADC 触发

互补 PWM 输出时 B 比较作为 ADC 触发功能

1. 设置为 PWM 互补输出模式，参考“中心对齐互补 PWM 输出设置”；
2. 设置 CCRxB 设置 ADC 触发比较点；
3. 设置 CRCHx.CISB 选择上计数下计数比较匹配（锯齿波不需要选择）；
4. 选择 ADC 触发比较的源 ADTR；
5. 使能 ADC 触发 ADTR.ADTE。

15.2.10 刹车控制

VC 比较输出可以控制刹车功能，外部 BK 端口输入可控制刹车，系统 fail 可以控制刹车功能。通过 CR.BG 可以实现软件刹车功能，控制输出端口到设定的状态。

两通道的输出的 Tim0/1/2 可以选择使用 Tim0 的刹车端口控制 TIM1/2 的刹车功能，也可以使用各自的刹车输入控制各自的刹车功能。当 DTR.BKSEL 设置为 1 时，TIM1/2 共用 TIM0 的刹车输入。

15.2.11 Timer 互联

TRGO 输出信号可以连接到其他定时器的 ITR 信号。连接关系如表 54：

表 54: TRGO 与 ITR 连接关系

ITRx / Timerx	ITR0	ITR1	ITR2	ITR3
Timer0	-	TIM1_TRGO	TIM2_TRGO	TIM3_TRGO
Timer1	TIM0_TRGO	-	TIM2_TRGO	TIM3_TRGO
Timer2	TIM0_TRGO	TIM1_TRGO	-	TIM3_TRGO
Timer3	TIM0_TRGO	TIM1_TRGO	TIM2_TRGO	-

- 定时器 TIM0/1/2 同时启动 PWM 输出设置（使用 TIM0 的 CTEN 触发 TIM1/2）示例；
  1. 参考 PWM 输出设置，设置定时器 0/1/2 的脉冲调整输出；
  2. 设置 TIM0 为主模式 MSCR.MSM=1；
  3. 设置 TIM0 的 CTEN 触发另外两个定时器 MSCR.MMS=1；
  4. 保持 TIM1/2 的 MSCR.MSM=0；
  5. 设置 TIM1/2 为触发模式 MSCR.SMS=2；
  6. 选择 TIM1/2 的触发源为 TIM0 的 TRGO, MSCR.TS= 最后使能定时器 TIM0，启动定时器 CR.CTEN=1。
- 定时器 TIM0/1/2 同时启动 PWM 输出设置（使用 TIM1 的 UG 触发 TIM0/2）示例。
  1. 参考 PWM 输出设置，设置定时器 0/1/2 的脉冲调整输出；
  2. 设置 TIM1 为触发主模式 MSCR.MSM=1；
  3. 设置 TIM1 的 UG 触发另外两个定时器 MSCR.MMS=0；
  4. 保持 TIM0/2 的 MSCR.MSM=0；
  5. 设置 TIM0/2 为触发模式 MSCR.SMS=2；
  6. 选择 TIM0/2 的触发源为 TIM0 的 TRGO, MSCR.TS=2；
  7. 最后使能定时器 TIM1，启动定时器 CR.UG=1。

15.2.12 GATE 输入互联

GATE 输入可以从端口通过 PX_SEL 选择直接输入，也可通过端口功能寄存器 GPIO_TIMGS 选择可以连通到其他模块或端口。

当 TIMx_G=0x0 时，门控 GATE 输入为 PX_SEL 选择的端口输入，当 TIMx_G=0x1~0x7 时，连接其他模块的输入或输出。

表 55: GATE 输入下的 TIMx_G 配置

	TIM0_g	TIM1_g	TIM2_g	TIM3_g
000	PX_SEL	PX_SEL	PX_SEL	PX_SEL
001	UART0_RXD	LPUART0_RXD	UART0_RXD	UART0_RXD
010	UART1_RXD	LPUART1_RXD	UART1_RXD	UART1_RXD
011	VC0_OUT	VC0_OUT	VC0_OUT	LPUART0
100	VC1_OUT	VC1_OUT	VC1_OUT	LPUART1
101	PA03	PA08	PA10	VC0_OUT
110	PB08	PB03	PB04	PA06
111	PB15	PB13	PB11	PA11

15.2.13 ETR 输入互联

ETR 输入可以从端口直接输入，也可通过端口功能寄存器 GPIO_TIMES 选择可以连通到其他模块或端口。

当 TIMx_E=0x0 时，外部时钟 EXT 输入为 PX_SEL 选择的端口输入，当 TIMx_E=0x1~0x7 时，连接其他模块的输入或输出。

表 56: ETR 输入下的 TIMx_G 配置

	TIM0_e	TIM1_e	TIM2_e	TIM3_e
000	PX_SEL	PX_SEL	PX_SEL	PX_SEL
001	LPUART0_RXD	UART0_RXD	LPUART0_RXD	UART0_RXD
010	LPUART1_RXD	UART1_RXD	LPUART1_RXD	UART1_RXD
011	VC0_OUT	VC1_OUT	VC0_OUT	VC1_OUT
100	LVD_OUT	LVD_OUT	PCNT_S1	PCNT_S0
101	PA00	PA01	PA04	PA00
110	PA05	PC09	PC04	PA12
111	PA15	PD02	PC08	PA13

15.2.14 CHx 捕获输入互联

Timer0/1/2 的 CHA,Timer3 的 CH0A/CH0B 输入可以从端口直接输入,也可通过端口功能寄存器 GPIO_TIMCPS 选择可以连通到其他模块或端口。

当 TIMx_CHy=0x0 时，捕获输入是 PX_SEL 选择的端口输入，当 TIMx_CHy=0x1~0x7 时，连接其他模块的输入或输出。

表 57: CHx 捕获输入下的 TIMx_G 配置

	TIM0_CHA	TIM1_CHA	TIM2_CHA	TIM3_CH0A	TIM3_CH0B
000	PX_SEL	PX_SEL	PX_SEL	PX_SEL	PX_SEL
001	UART0_RXD	UART1_RXD	LPUART0_RXD	LPUART1_RXD	UART0_RXD
010	PA00	PA00	VC0_OUT	LPUART0_RXD	UART1_RXD
011	PA02	PA02	PA02	PCNT_S0	PCNT_S1
100	PA05	PA06	PA07	VC0_OUT	VC1_OUT
101	PA15	PB08	PB08	PA08	PA07
110	PB06	PB10	PB09	PB03	PB04
111	PB14	PB13	PC06	PB06	PB13

15.2.15 DMA

Timer 支持软件和硬件触发 DMA 进行数据传输。支持数据从其他位置写入定时器，或从定时器读出写入其他位置。可应用于数据捕获后数据的自动搬运和更改周期值或占空比的脉冲宽度的自动调整。每个定时器有两个 DMA 请求，A 请求触发源可选择比较捕获 A，外部触发，B 请求触发源可选择比较捕获 B，事件更新。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

表 58: 触发源

IDREQ	Interrupt Signal of Peripheral
18	TIM0A
19	TIM0B
20	TIM1A
21	TIM1B
22	TIM2A
23	TIM2B
24	TIM3A
25	TIM3B

- TIMxA 的触发源可选择比较捕获 A,ETRIG；
- TIMxB 的触发源可选择比较捕获 B,UEV。

设置示例

- 捕获数据 DMA 数据传输；
  1. 使能 DMA；
  2. 使能 DMA 通道使能；
  3. 选择定时器 DMA 的通道；
  4. 设置传输类型、传输长度、传输方式；
  5. 设置源起始地址，目标起始地址；
  6. 设置源地址、目标地址的递增方式；
  7. 根据需要使能 DMA 中断；
  8. 参考数据捕获流程设置数据捕获；
  9. 使能 CRCHx.CDy 使能捕获数据触发 DAM 传输。

  Note：源地址设置为捕获通道寄存器，源地址固定，目标地址加。
- 脉冲调整 DMA 数据传输。
  1. 使能 DMA；
  2. 使能 DMA 通道使能；
  3. 选择定时器 DMA 的通道；
  4. 设置传输类型、传输长度、传输方式；
  5. 设置源起始地址，目标起始地址；
  6. 设置源地址、目标地址的递增方式；
  7. 根据需要使能 DMA 中断；
  8. 参考脉冲调整输出设置脉冲调制输出；
  9. 根据需要选择硬件触发 DMA 的条件（UEV 触发或者比较匹配触发）。

  Note：目标地址设置为捕获通道寄存器，源地址变化，目标地址固定。

15.3 TIMx 寄存器描述

表 59. TIMx 寄存器概览

Offset	Acronym	Register Name	Reset	Section
模式 0 定时器寄存器				
0X00	TIMx_ARR	Timer 重载寄存器/周期	0x0000 0000	小节 15.4.1
0X04	TIMx_CNT	Timer 16 位模式计数寄存器	0x0000 0000	小节 15.4.2
0X08	TIMx_CNT32	Timer 32 位模式计数寄存器	0x0000 0000	小节 15.4.3
0X0C	TIMx_M0CR	Timer 模式 0 控制寄存器（按不同模式进行描述）	0x0060 0008	小节 15.4.4
0X10	TIMx_IFR	Timer 中断标志	0x0000 0000	小节 15.4.5
0X14	TIMx_ICLR	Timer 中断清除寄存器	0x0000 FFFF	小节 15.4.6
0X30	TIMx_DTR	死区寄存器	0x0000 0000	小节 15.4.7
脉冲宽度测量 PWC 寄存器				
0X04	TIMx_CNT	Timer 16 位模式计数寄存器	0x0000 0000	小节 15.5.1
0X0C	TIMx_M1CR	Timer 模式 1 控制寄存器（按不同模式进行描述）	0x0060 0008	小节 15.5.2
0X10	TIMx_IFR	Timer 中断标志	0x0000 0000	小节 15.5.3
0X14	TIMx_ICLR	Timer 中断清除寄存器	0x0000 FFFF	小节 15.5.4
0X18	TIMx_MSCR	主从模式控制	0x0000 0000	小节 15.5.5
0X1c	TIMx_FLTR	滤波控制	0x0000 0000	小节 15.5.6
0x24	TIMx_CR0	控制寄存器	0x0000 3000	小节 15.5.7
0X3C	TIMx_CCR0A	比较 0A 寄存器	0x0000 0000	小节 15.5.8
模式 2,3 寄存器模式 2,3 寄存器				
0X38	TIMx_ARRDM	Timer 重载寄存器/周期映射地址	0x0000 0000	小节 15.6.1
0X04	TIMx_CNT	Timer 16 位模式计数寄存器	0x0000 0000	小节 15.6.2
0X0C	TIMx_M23CR	Timer 模式 23 控制寄存器（按不同模式进行描述）	0x0060 0008	小节 15.6.3
0X10	TIMx_IFR	Timer 中断标志	0x0000 0000	小节 15.6.4
0X14	TIMx_ICLR	Timer 中断清除寄存器	0x0000 FFFF	小节 15.6.5
0X18	TIMx_MSCR	主从模式控制	0x0000 0000	小节 15.6.6
0X1c	TIMx_FLTR	滤波控制	0x0000 0000	小节 15.6.7
0X20	TIMx_ADTR	ADC 触发控制	0x0000 0000	小节 15.6.8
0x24	TIMx_CRCH0	比较单元 0 控制寄存器	0x0000 3000	小节 15.6.9
0X28	TIMx_CRCH1	比较单元 1 控制寄存器	0x0000 3000	小节 15.6.10
0x2C	TIMx_CRCH2	比较单元 2 控制寄存器	0x0000 3000	小节 15.6.10
0X30	TIMx_DTR	死区寄存器	0x0000 0000	小节 15.6.11
0X34	TIMx_RCR	重复计数寄存器	0x0000 0000	小节 15.6.12
0X3C	TIMx_CCR0A	比较 0A 寄存器	0x0000 0000	小节 15.6.13
0x40	TIMx_CCR0B	比较 0B 寄存器	0x0000 0000	小节 15.6.13

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Offset	Acronym	Register Name	Reset	Section
0X44	TIMx_CCR1A	比较 1A 寄存器	0x0000 0000	小节 15.6.14
0X48	TIMx_CCR1B	比较 1B 寄存器	0x0000 0000	小节 15.6.14
0X4C	TIMx_CCR2A	比较 2A 寄存器	0x0000 0000	小节 15.6.14
0X50	TIMx_CCR2B	比较 2B 寄存器	0x0000 0000	小节 15.6.14

注：TIMx_CNT32 是 TIMx_ARR 与 TIMx_CNT 的重映射寄存器。为了方便 32 位定时器写操作。

15.4 模式 0 定时器寄存器描述

15.4.1 Timer 重载寄存器/周期 (TIMx_ARR)

起始地址：Timer0:0x40000C00 Timer1:0x40000D00 Timer2:0x40000E00 Timer3:0X40005800

地址偏移：0X00

复位值：0x00000000

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。
15:0	ARR	rw	0x00	16 位重载定时器重载值寄存器

15.4.2 Timer16 位模式计数寄存器 (TIMx_CNT)

起始地址：Timer0:0x40000C00 Timer1:0x40000D00 Timer2:0x40000E00 Timer3:0X40005800

地址偏移：0X04

复位值：0x00000000

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。
15:0	CNT	rw	0x00	16 位重载定时器计数值寄存器。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

15.4.3 Timer32 位模式计数寄存器 (TIMx_CNT32)

起始地址：Timer0:0x40000C00 Timer1:0x40000D00 Timer2:0x40000E00 Timer3:0X40005800

地址偏移：0X08

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CNT32[31:16]															
RW															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT32[15:0]															
RW															

Bit	Field	Type	Reset	Description
31:0	CNT32	rw	0x00	32 位定时器计数值寄存器。 注：仅在模式 0 32 位定时器自由计数模式下有效，其他模式禁止写该寄存器。

15.4.4 Timer 模式 0 控制寄存器 (TIMx_M0CR)

起始地址：Timer0:0x40000C00 Timer1:0x40000D00 Timer2:0x40000E00 Timer3:0X40005800

地址偏移：0X0C

复位值：0x00600008

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

15	14	13	12	11	10	9	8
Resvered		Mode		Resvered	UIE	GATEP	GATE
		RW			RW	RW	RW

7	6	5	4	3	2	1	0
Resvered	PRS			TOGEN	CT	MD	CTEN
	RW			RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31:14	Reserved			保留。
13:12	MODE	rw	0x00	工作模式。 00 定时器模式 0; 01 PWC 模式 10 锯齿波模式; 11 三角波模式
11	Reserved			保留。
10	UIE	rw	0x00	中断使能控制，写 1 后使能中断。

Bit	Field	Type	Reset	Description
9	GATEP	rw	0x00	端口 GATE 极性控制。 0：端口 GATE 高电平有效 1：端口 GATE 低电平有效
8	GATE	rw	0x00	定时器门控使能。 0：无门控，CTEN=1 时定时器工作 1：门控使能，端口 GATE 有效并且 CTEN=1 时定时器才工作
7	Reserved			保留。
6:4	PRS	rw	0x00	内部时钟分频选择。 000：1；001：2；010：4；011：8 100：16；101：32；110：64；111：256
3	TOGEN	rw	0x01	模式 0 下翻转输出使能。 1：翻转输出使能 0：翻转输出关闭 CHA,CHB 输出为低电平
2	CT	rw	0x00	计数时钟选择。 0：内部计数时钟 TCLK 1：外部计数时钟 ETR
1	MD	rw	0x00	模式选择 32 定时/16 定时模式选择。 0：32 位自由计数 1：16 位重载计数
0	CTEN	rw	0x00	定时器使能。 0：定时器停止 1：定时器使能

15.4.5 Timer 中断标志 (TIMx_IFR)

起始地址：Timer0:0x40000C00　Timer1:0x40000D00　Timer2:0x40000E00　Timer3:0X40005800

地址偏移：0X10

复位值：0x00000000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved															UIF R1W0

Bit	Field	Type	Reset	Description
31:1	Reserved			保留。
0	UIF	r	0x00	溢出中断。

15.4.6 Timer 中断清除寄存器 (TIMx_ICLR)

起始地址：Timer0:0x40000C00　Timer1:0x40000D00　Timer2:0x40000E00　Timer3:0X40005800

地址偏移：0X14

复位值：0x0000FFFF

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

<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="15" rowspan="2">Reserved</td><td>UIF</td></tr>
<tr><td>R1W0</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:1	Reserved			保留。
0	UIF	r	0x01	溢出中断清除，写 0 清除。

15.4.7 死区寄存器 (TIMx_DTR)

起始地址：Timer0:0x40000C00　Timer1:0x40000D00　Timer2:0x40000E00　Timer3:0X40005800

地址偏移：0X30

复位值：0x00000000

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

<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="3" rowspan="2">Reserved</td><td>MOE</td><td colspan="12" rowspan="2">Resvered</td></tr>
<tr><td>RW</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:13	Reserved			保留。
12	MOE	r	0x00	翻转输出使能。 0：翻转输出为输入状态 1：翻转端口为输出状态
11:0	Reserved			保留。

15.5 脉冲宽度测量 PWC 寄存器描述

15.5.1 Timer16 位模式计数寄存器 (TIMx_CNT)

起始地址：Timer0:0x40000C00　Timer1:0x40000D00　Timer2:0x40000E00　Timer3:0X40005800

地址偏移：0X04

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

复位值：0x00000000

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
</table>

<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="16">CNT</td></tr>
<tr><td colspan="16">RW</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。
15:0	CNT	rw	0x00	16 位重载定时器计数值寄存器。

15.5.2 Timer 模式 1 控制寄存器（按不同模式进行描述）(TIMx_M1CR)

起始地址：Timer0:0x40000C00 Timer1:0x40000D00 Timer2:0x40000E00 Timer3:0X40005800

地址偏移：0X0C

复位值：0x00600008

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
</table>

<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr>
<tr><td rowspan="2">Resvered</td><td>Oneshot</td><td colspan="2">Mode</td><td rowspan="2">Resvered</td><td>UIE</td><td>Edg2nd</td><td>Edg1st</td></tr>
<tr><td>RW</td><td colspan="2">RW</td><td>RW</td><td>RW</td><td>RW</td></tr>
</table>

<table>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td rowspan="2">Resvered</td><td colspan="3">PRS</td><td rowspan="2">Resvered</td><td>CT</td><td rowspan="2">Resvered</td><td>CTEN</td></tr>
<tr><td colspan="3">RW</td><td>RW</td><td>RW</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:15	Reserved			保留。
14	Oneshot	rw	0x00	单次触发模式选择。 1：完成一次脉冲测量自动结束，再次测量需要重新使能 CTEN 0：循环测量
13:12	MODE	rw	0x00	工作模式。 00 定时器模式 0; 01 PWC 模式 10 锯齿波模式; 11 三角波模式
11	Reserved			保留。
10	UIE	rw	0x00	中断使能控制，写 1 后使能中断。 计数到 0XFFFF 才会溢出并产生溢出标志
9	Edg2nd	rw	0x00	脉冲宽度测量结束边沿选择。

Bit	Field	Type	Reset	Description
8	Edg1st	rw	0x00	脉冲宽度测量开始边沿选择。 Edg2nd Edg1st: 00 / 01 测量: 上沿-上沿周期 / 低电平宽度 Edg2nd Edg1st: 10 / 11 测量: 高电平宽度 / 下沿-下沿周期
7	Reserved			保留。
6:4	PRS	rw	0x00	内部时钟分频选择。 000：1； 001：2； 010：4； 011：8； 100：16； 101：32； 110：64； 111：256；
3	Reserved			保留。
2	CT	rw	0x00	计数时钟选择。 0：内部计数时钟 TCLK 1：外部计数时钟 ETR
1	Reserved			保留。
0	CTEN	rw	0x00	脉冲宽度测量使能。 0：禁止 1：使能

15.5.3 Timer 中断标志 (TIMx_IFR)

起始地址：Timer0:0x40000C00 Timer1:0x40000D00 Timer2:0x40000E00 Timer3:0X40005800

地址偏移：0X10

复位值：0x00000000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													CA0F	Res.	UIF
													RO		RO

Bit	Field	Type	Reset	Description
31:3	Reserved			保留。
2	CA0F	r	0x00	脉冲宽度测量中断标志。
1	Reserved			保留。
0	UIF	r	0x00	溢出中断标志。

15.5.4 Timer 中断清除寄存器 (TIMx_ICLR)

起始地址：Timer0:0x40000C00 Timer1:0x40000D00 Timer2:0x40000E00 Timer3:0X40005800

地址偏移：0X14

复位值：0x0000FFFF

31:16
Reserved

15:3	2	1	0
Reserved	CA0F	Res.	UIF
	R1W0		R1W0

Bit	Field	Type	Reset	Description
31:3	Reserved			保留。
2	CA0F	rc_w0	0x01	脉冲宽度测量中断标志清除，写 0 清除。
1	Reserved			保留。
0	UIF	rc_w0	0x01	溢出中断清除，写 0 清除。

15.5.5 主从模式控制 (TIMx_MSCR)

起始地址：Timer0:0x40000C00 Timer1:0x40000D00 Timer2:0x40000E00 Timer3:0X40005800

地址偏移：0X18

复位值：0x00000000

31:16
Reserved

15:13	12	11	10:8	7:5	4:0
Resverved	IB0S	IA0S	Resvered	TS	Resvered
	RW	RW		RW	

Bit	Field	Type	Reset	Description
31:13	Reserved			保留。
12	IB0S	rw	0x00	CH0B 输入选择。 0：CH0B; 1：内部触发 TS 选择信号 注：当 PWM 互补输出时自动选择 GATE 端口作为 CH0B 的输入。
11	IA0S	rw	0x00	IA0 输入选择。 0：CH0A; 1：CH0A CH1A CH2A XOR(TIM3) 0：CH0A; 1：CH0A ETR GATE XOR(TIM0) 注：设置为 1 后端口的任意一个端口变化将导致输入变化。
10:8	Reserved			保留。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
7:5	TS	rw	0x00	触发选择。 000：端口 ETR 的滤波相位选择后的信号 ETFP 001：内部互联信号 ITR0 010：内部互联信号 ITR1 011：内部互联信号 ITR2 100：内部互联信号 ITR3 101：端口 CH0A 的边沿信号 110：端口 CH0A 的滤波相位选择后的信号 IAFP 111：端口 CH0B 的滤波相位选择后的信号 IBFP
4:0	Reserved			保留。

15.5.6 滤波控制 (TIMx_FLTR)

起始地址：Timer0:0x40000C00 Timer1:0x40000D00 Timer2:0x40000E00 Timer3:0X40005800

地址偏移：0X1c

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ETP	FLTET			Resvered											
RW	RW														

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Resvered									FLTB0			Resvered	FLTA0		
									RW				RW		

Bit	Field	Type	Reset	Description
31	ETP	rw	0x00	ETR 输入相位选择。 0：同相位 1：反向输入
30:28	FLTET	rw	0x00	ETR 滤波控制。 滤波设置 0xx：滤波无效 100：pclk 3 个连续有效 101：pclk/4 3 个连续有效 110：pclk/16 3 个连续有效 111：pclk/64 3 个连续有效
27:7	Reserved			保留。

Bit	Field	Type	Reset	Description
6:4	FLTB0	rw	0x00	CHB 输入滤波控制。 滤波设置 0xx：滤波无效 100：pclk 3 个连续有效 101：pclk/4 3 个连续有效 110：pclk/16 3 个连续有效 111：pclk/64 3 个连续有效
3	Reserved			保留。
2:0	FLTA0	rw	0x00	CHA 输入滤波控制。 滤波设置 0xx：滤波无效 100：pclk 3 个连续有效 101：pclk/4 3 个连续有效 110：pclk/16 3 个连续有效 111：pclk/64 3 个连续有效

15.5.7 比较单元 0 控制寄存器 (TIMx_CR0)

起始地址：Timer0:0x40000C00 Timer1:0x40000D00 Timer2:0x40000E00 Timer3:0X40005800

地址偏移：0x24

复位值：0x00003000

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

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

Bit	Field	Type	Reset	Description
31:9	Reserved			保留。
8	CIEA	rw	0x00	CIEA 脉冲宽度测量完成中断使能。 0：禁止 1：使能
7:0	Reserved			保留。

15.5.8 比较捕获寄存 (TIMx_CCR0A)

起始地址：Timer0:0x40000C00 Timer1:0x40000D00 Timer2:0x40000E00 Timer3:0X40005800

地址偏移：0X3C

复位值：0x00000000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR0A															
R															

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。
15:0	CCR0A	r	0x00	CIEA 脉冲宽度测量完成中断使能。

15.6 模式 2,3 寄存器描述

15.6.1 16 位模式重载寄存器 (TIMx_ARR)

起始地址：Timer0:0x40000C00　Timer1:0x40000D00　Timer2:0x40000E00　Timer3:0X40005800

地址偏移：0x00(0X38 dummy address)

复位值：0x00000000

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。
15:0	ARR	rw	0x00	重载寄存器/周期寄存器，具有缓存功能。 在计数器未使能或者缓存没有使能时，缓存寄存器可以立刻更新。

15.6.2 Timer16 位模式计数寄存器 (TIMx_CNT)

起始地址：Timer0:0x40000C00　Timer1:0x40000D00　Timer2:0x40000E00　Timer3:0X40005800

地址偏移：0X04

复位值：0x00000000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。
15:0	CNT	rw	0x00	16 位重载定时器计数值寄存器。

注：使用 PWM 比较输出时，初始化 CNT 值需要小于 ARR 的值。

15.6.3 Timer 模式 23 控制寄存器（按不同模式进行描述）(TIMx_M23CR)

起始地址：Timer0:0x40000C00 Timer1:0x40000D00 Timer2:0x40000E00 Timer3:0X40005800

地址偏移：0X0C

复位值：0x00600008

31	30	29	28	27	26	25	24
Resvered				DIR	BG	UG	TG
				RW/RO	W1	W1	W1

23	22	21	20	19	18	17	16
OCCE	CIS		BIE	TIE	TDE	URS	OCCS
RW	RW		RW	RW	RW	RW	RW

15	14	13	12	11	10	9	8
CSG	Oneshot	Mode		UDE	UIE	CFG	CRG
RW	RW	RW		RW	RW	RW	RW

7	6	5	4	3	2	1	0
BUFPEN	PRS			Pwm2s	CT	Comp	CTEN
RW	RW			RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31:28	Reserved			保留。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
27	DIR	rw/r	0x00	计数方向，只有在锯齿波模式下可以写。其他模式下只读，写无效。 0：上计数 1：下计数 注：从其他模式切换到中心对齐模式 DIR 自动清 0。软件事件更新和从模式外部触发复位模式 DIR 自动清零。
26	BG	rc_w1	0x00	软件刹车，自动清零。 写 1 产生软件刹车； 写 0 无效
25	UG	rc_w1	0x00	软件更新，自动清零。 写 1 产生软件更新； 写 0 无效 初始化计数器并更新缓存寄存器到相应寄存器（缓存使能），预除频计数器也会被清零。
24	TG	rc_w1	0x00	软件触发，自动清零，需要在触发模式 SMS=2 且 mode=2/3 下都可以触发。 写 1 产生软件触发； 写 0 无效
23	OCCE	rw	0x00	OCREF 清除使能。 1：OCREF_CLR 信号可以清除 OCREF 输出 0：OCREF 输出不受 OCREF_CLR 影响
22:21	CIS	rw	0x03	中心对齐 A 比较中断模式 (B 比较中断单独控制在 CRx 寄存器中 CISB00：无中断。 01：上沿中断， 10：下沿中断， 11：上下沿都中断
20	BIE	rw	0x00	刹车中断使能。 1：中断使能 0：中断禁止
19	TIE	rw	0x00	触发中断使能。 1：中断使能 0：中断禁止
18	TD	rw	0x00	E 触发 DMA 使能。 1：中断使能 0：中断禁止
17	URS	rw	0x00	更新源。 0：上溢出/下溢出/软件更新 UG/从模式复位; 1：上溢出/下溢出

Bit	Field	Type	Reset	Description
16	OCCS	rw	0x00	OCREF 清除源选择。 0：电压比较器 VC 输出，VC 选择在 VCx_OUTCFG 寄存器设置 1：ETR 端口滤波相位选择后的信号 当 OCCE 有效时，OC_clr 可以清除 OCREF 的比较输出信号为零，（当 OCMx>1 时有效），下一个 uev 事件后继续比较输出
15	CSG	rw	0x00	GATE 在 PWM 互补模式下捕获/比较选择。（只有在 PWM 互补输出时有效） 使用 CCR0B 作为 GATE 的比较或捕获通道 1：捕获； 0：比较
14	Oneshot	rw	0x00	单次触发模式选择。 1：发生事件更新后定时器停止。 0：循环计数
13:12	MODE	rw	0x00	工作模式。 00 定时器模式 0; 01 PWC 模式 10 锯齿波模式; 11 三角波模式
11	UDE	rw	0x00	更新 DMA 使能。 1：使能更新触发 DMA 0：禁止更新触发 DMA
10	UIE	rw	0x00	UIE 更新中断使能。 1：使能更新中断 0：禁止更新中断
9	CFG	rw	0x00	GATE 作为捕获输入时，下沿捕获有效控制。（只有在 PWM 互补输出时有效） 1：下沿捕获有效 0：下沿捕获无效
8	CRG	rw	0x00	GATE 作为捕获输入时，上沿捕获有效控制。（只有在 PWM 互补输出时有效） 1：上沿捕获有效 0：上沿捕获无效
7	BUFPEN	rw	0x00	重载缓存使能。 1：周期缓存使能，写入后下个事件更新后才影响到周期值。 0：周期缓存无效，写入后立刻影响周期值
6:4	PRS	rw	0x00	内部时钟分频选择。 000：1　001：2　010：4　011：8 100：16　101：32　110：64　111：256

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
3	PWM2S	rw	0x01	OCREFA 双点比较选择 (缺省值为 1)。 0: 双点比较使能，使用 CCRA,CCRB 比较控制 OCREFA 输出 1：单点比较使能，只使用 CCRA 比较控制 OCREFA 输出 注：OCREFB 不受影响，仍然使用 CCRB 控制 OCREFB 输出
2	CT	rw	0x00	计数时钟选择。 0：内部计数时钟 TCLK 1：外部计数时钟 ETR;
1	Comp	rw	0x00	PWM 互补输出模式选择。 0：独立 PWM 输出 1：互补 PWM 输出
0	CTEN	rw	0x00	定时器使能。 0：禁止； 1：使能 可以外部触发使能，在 oneshot 模式下结束时该位自动清零

15.6.4 Timer 中断标志 (TIMx_IFR)

起始地址：Timer0:0x40000C00 Timer1:0x40000D00 Timer2:0x40000E00 Timer3:0X40005800

地址偏移：0X10

复位值：0x00000000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIF	BIF	CB2E	CB1E	CB0E	CA2E	CA1A	CA0E	CB2F	CB1F	CB0F	CA2F	CA1F	CA0F	Res.	UIF
RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO		RO

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。
15	TIF	r	0x00	触发中断标志。
14	BIF	r	0x00	刹车中断标志。
13	CB2E	r	0x00	通道 CH2B 捕获数据丢失标志。 0：无数据丢失；1：数据丢失（仅 TIM3 存在）
12	CB1E	r	0x00	通道 CH1B 捕获数据丢失标志。 0：无数据丢失；1：数据丢失（仅 TIM3 存在）
11	CB0E	r	0x00	通道 CH0B 捕获数据丢失标志。 0：无数据丢失；1：数据丢失

Bit	Field	Type	Reset	Description
10	CA2E	r	0x00	通道 CH2A 捕获数据丢失标志。 0：无数据丢失；1：数据丢失（仅 TIM3 存在）
9	CA1E	r	0x00	通道 CH1A 捕获数据丢失标志。 0：无数据丢失；1：数据丢失（仅 TIM3 存在）
8	CA0E	r	0x00	通道 CH0A 捕获数据丢失标志。 0：无数据丢失；1：数据丢失
7	CB2F	r	0x00	通道 CH2B 发生捕获/比较匹配标志。 0：没有发生 1：发生（仅 TIM3 存在）
6	CB1F	r	0x00	通道 CH1B 发生捕获/比较匹配标志。 0：没有发生 1：发生（仅 TIM3 存在）
5	CB0F	r	0x00	通道 CH0B 发生捕获/比较匹配标志。 0：没有发生 1：发生
4	CA2F	r	0x00	通道 CH2A 发生捕获/比较匹配标志。 0：没有发生 1：发生（仅 TIM3 存在）
3	CA1F	r	0x00	通道 CH1A 发生捕获/比较匹配标志。 0：没有发生 1：发生（仅 TIM3 存在）
2	CA0F	r	0x00	通道 CH0A 发生捕获/比较匹配标志。 0：没有发生 1：发生
1	Reserved			保留。
0	UIF	r	0x00	事件更新中断标志。 0：没有发生 1：发生

15.6.5 Timer 中断清除寄存器 (TIMx_ICLR)

起始地址：Timer0:0x40000C00　Timer1:0x40000D00　Timer2:0x40000E00　Timer3:0X40005800

地址偏移：0X14

复位值：0x0000FFFF

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIF	BIF	CB2E	CB1E	CB0E	CA2E	CA1A	CA0E	CB2F	CB1F	CB0F	CA2F	CA1F	CA0F	Res.	UIF
R1W0	R1W0	R1W0	R1W0	R1W0	R1W0	R1W0	R1W0	R1W0	R1W0	R1W0	R1W0	R1W0	R1W0		R1W0

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。
15	TIF	rc_w0	0x00	触发中断标志清除，写 0 清除。
14	BIF	rc_w0	0x00	刹车中断标志清除，写 0 清除。
13	CB2E	rc_w0	0x00	通道 CH2B 捕获数据丢失标志清除，写 0 清除（仅 TIM3 存在）。

Bit	Field	Type	Reset	Description
12	CB1E	rc_w0	0x00	通道 CH1B 捕获数据丢失标志清除，写 0 清除（仅 TIM3 存在）。
11	CB0E	rc_w0	0x00	通道 CH0B 捕获数据丢失标志清除，写 0 清除。
10	CA2E	rc_w0	0x00	通道 CH2A 捕获数据丢失标志清除，写 0 清除（仅 TIM3 存在）。
9	CA1E	rc_w0	0x00	通道 CH1A 捕获数据丢失标志清除，写 0 清除（仅 TIM3 存在）。
8	CA0E	rc_w0	0x00	通道 CH0A 捕获数据丢失标志清除，写 0 清除。
7	CB2F	rc_w0	0x00	通道 CH2B 捕获/比较匹配标志清除，写 0 清除（仅 TIM3 存在）。
6	CB1F	rc_w0	0x00	通道 CH1B 捕获/比较匹配标志清除，写 0 清除（仅 TIM3 存在）。
5	CB0F	rc_w0	0x00	通道 CH0B 捕获/比较匹配标志清除，写 0 清除。
4	CA2F	rc_w0	0x00	通道 CH2A 捕获/比较匹配标志清除，写 0 清除（仅 TIM3 存在）。
3	CA1F	rc_w0	0x00	通道 CH1A 捕获/比较匹配标志清除，写 0 清除（仅 TIM3 存在）。
2	CA0F	rc_w0	0x00	通道 CH0A 捕获/比较匹配标志清除，写 0 清除。
1	Reserved			保留。
0	UIF	rc_w0	0x00	事件更新中断清除，写 0 清除。

15.6.6 主从模式控制 (TIMx_MSCR)

起始地址：Timer0:0x40000C00 Timer1:0x40000D00 Timer2:0x40000E00 Timer3:0X40005800

地址偏移：0X18

复位值：0x00000000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Resverved			IB0S	IA0S	SMS			TS			***MSM***	***CCDS***	MMS		
			RW	RW	RW			RW			RW	RW	RW		

Bit	Field	Type	Reset	Description
31:13	Reserved			保留。
12	IB0S	rw	0x00	CH0B 输入选择。 0：CH0B; 1：内部触发 TS 选择信号； 注：当 PWM 互补输出时自动选择 GATE 端口作为 CH0B 的输入

Bit	Field	Type	Reset	Description
11	IA0S	rw	0x00	IA0 输入选择。 0：CH0A; 1：CH0A CH1A CH2A XOR(TIM3) 0：CH0A; 1：CH0A ETR GATE XOR(TIM0) 注：设置为 1 后端口的任意一个端口变化将导致输入变化
10:8	SMS	rw	0x00	从模式功能选择。 000：使用内部时钟; 001：复位功能; 010：触发模式; 011：外部时钟模式 100：正交编码计数模式 1 ; 101：正交编码计数模式 2; 110：正交编码计数模式 3; 111：门控功能
7:5	TS	rw	0x00	触发选择。 000：端口 ETR 的滤波相位选择后的信号 ETFP ; 001：内部互联信号 ITR0 010：内部互联信号 ITR1; 011：内部互联信号 ITR2; 100：内部互联信号 ITR3; 101：端口 CH0A 的边沿信号; 110：端口 CH0A 的滤波相位选择后的信号 IAFP 111：端口 CH0B 的滤波相位选择后的信号 IBFP;
4	MSM	rw	0x00	主从选择。 0：无延时 1：延时使能，使主送计数器同时启动. 注：使用触发模式时，从模式设置为 0，主模式设置为 1，可以使主从计数同时启动
3	CCDS	rw	0x00	比较模式下 DMA 比较触发选择。 0 ：比较匹配触发 DMA; 1：比较匹配不触发 DMA，事件更新代替比较匹配触发 DMA

Bit	Field	Type	Reset	Description
2:0	MMS	rw	0x00	主模式输出选择，用于内部互联，连接到其他定时器的 ITR。 定时器 0/1/2 000：软件更新 UG，写 CR.UG 001：定时器使能 CTEN 010：定时器事件更新 UEV; 011：比较匹配选择输出 CMPSO; 100：定时器比较参数输出 OCREF0A 101：定时器比较参数输出 OCREF0B 110：定时器比较参数输出 OCREF0B 111：定时器比较参数输出 OCREF0B 定时器 3 000：软件更新 UG，写 CR.UG 001：定时器使能 CTEN 010：定时器事件更新 UEV; 011：比较匹配选择输出 CMPSO; 100：定时器比较参数输出 OCREF0A 101：定时器比较参数输出 OCREF1A 110：定时器比较参数输出 OCREF2A 111：定时器比较参数输出 OCREF0B

15.6.7 输出控制/输入滤波 (TIMx_FLTR)

起始地址：Timer0:0x40000C00　Timer1:0x40000D00　Timer2:0x40000E00　Timer3:0X40005800

地址偏移：0X1c

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ETP	FLTET			BKP	FLTBK			CCPB2	OCMB2			CCPA2	OCMA2		
-	-			-	-			-	FLTB2			-	FLTA2		
RW	RW			RW	RW			RW	RW			RW	RW		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCPB1	OCMB1			CCPA1	OCMA1			CCPB0	OCMB0			CCPA0	OCMA0		
-	FLTB1			-	FLTA1			CCPB0	FLTB0			CCPA0	FLTA0		
RW	RW			RW	RW			RW	RW			RW	RW		

Bit	Field	Type	Reset	Description
31	ETP	rw	0x00	ETR 输入相位选择。 0：同相位 1：反向输入

Bit	Field	Type	Reset	Description
30:28	FLTET	rw	0x00	ETR 滤波控制。 滤波设置 0xx：滤波无效 100：pclk 3 个连续有效 101：pclk/4 3 个连续有效 110：pclk/16 3 个连续有效 111：pclk/64 3 个连续有效
27	BKP	rw	0x00	刹车 BK 输入相位选择。 0：同相位 1：反向输入；
26:24	FLTBK	rw	0x00	刹车输入滤波控制。 滤波设置 0xx：滤波无效 100：pclk 3 个连续有效; 101：pclk/4 3 个连续有效 110：pclk/16 3 个连续有效; 111：pclk/64 3 个连续有效 注：为了保证 OCMB0 的 PWM 输出设置，GATE 在 PWM 互补模式下作为捕获输入，滤波设置无效。
23	CCPB2	rw	0x00	比较功能：CH2B 通道比较输出相位控制。 0：正常输出 1：反向输出
22:20	OCMB2 FLTB2	rw	0x00	比较功能：CH2B 通道比较控制; 参考 OCMB0。 捕获功能：CH2B 输入通道滤波设置，参考 FLTBK
19	CCPA2	rw	0x00	比较功能：CH2A 通道比较输出相位控制。 0：正常输出 1：反向输出
18:16	OCMA2 FLTA2	rw	0x00	比较功能：CH2A 通道比较控制; 参考 OCMB0。 捕获功能：CH2A 输入通道滤波设置，参考 FLTBK。
15	CCPB1	rw	0x00	比较功能：CH1B 通道比较输出相位控制。 0：正常输出 1：反向输出
14：12	OCMB1 FLTB1	rw	0x00	比较功能：CH1B 通道比较控制; 参考 OCMB0。 捕获功能：CH1B 输入通道滤波设置，参考 FLTBK。
11	CCPA1	rw	0x00	比较功能：CH1A 通道比较输出相位控制。 0：正常输出 1：反向输出
10:8	OCMA1 FLTA1	rw	0x00	比较功能：CH1A 通道比较控制; 参考 OCMB0 捕获功能：CH1A 输入通道滤波设置，参考 FLTBK

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
7	CCPB0	rw	0x00	比较功能：输出比较模式 CCPBx 比较输出 CHBx 端口极性控制 0：正常输出 1：反向输出 编码计数与从模式门控功能：输入相位控制 CCPB0 从模式门控，复位，外部触发，外部时钟使用 CH0B 端口输入极性控 0：正常输入； 1：反向输入
6:4	OCMB0 FLTB0	rw	0x00	比较功能：CH0B 通道比较控制 000：强制为 0 001：强制为 1 010：比较匹配时强制为 0 011：比较匹配时强制为 1 100：比较匹配时翻转 101：比较匹配时输出一个计数周期的高电平 110：PWM 模式 1 ● 单点比较： – 上计数时 CNT<CCRxy 输出高，下计数时 CNT>CCRxy 输出为低电平。 ● 双点比较： – 锯齿波上计数 CCRxA<CNT≤CCRxB 输出为低电平； – 锯齿波下计数 CCRxA<CNT≤CCRxB 输出为高电平； – 三角波上计数 CNT<CCRxA 输出高，下计数 CNT>CCRxB 为低电平。 111 PWM 模式 2 ● 单点比较： – 上计数时 CNT<CCRxy 输出低，下计数时 CNT>CCRxy 输出为高电平。 ● 双点比较： – 锯齿波上计数 CCRxA≤CNT<CCRxB 输出为高电平； – 锯齿波下计数 CCRxA≤CNT<CCRxB 输出为低电平； – 三角波上计数 CNT<CCRxA 输出低，下计数 CNT>CCRxB 为高电平。 捕获功能：CH0B 输入通道滤波设置，参考 FLTBK

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
3	CCPA0	rw	0x00	比较功能：CCPAx 比较输出 CHAx 端口极性控制 0：正常输出 1：反向输出 编码计数与从模式门控功能：输入相位控制 CCPA0 从模式门控，复位，外部触发，外部时钟使用 CH0A 端口输入极性控制 0：正常输入 1：反向输入
2:0	OCMA0 FLTA0	rw	0x00	比较功能：A 通道比较控制；参考 OCMB0 捕获功能：CH0A 输入通道滤波设置，参考 FLTBK

15.6.8 ADC 触发控制 (TIMx_ADTR)

起始地址：Timer0:0x40000C00 Timer1:0x40000D00 Timer2:0x40000E00 Timer3:0X40005800

地址偏移：0X20

复位值：0x00000000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								ADTE	CMB2E	CMB1E	CMB0E	CMA2E	CMA1E	CMA0E	UEVE
								RW	RW	RW	RW	RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31:8	Reserved			保留。
7	ADTE	rw	0x00	使能 ADC 触发全局控制。 1：使能 0：禁止
6	CMB2E	rw	0x00	通道 2B 比较匹配触发 ADC 使能，只有 TIM3 存在。 1：使能 0：禁止
5	CMB1E	rw	0x00	通道 1B 比较匹配触发 ADC 使能，只有 TIM3 存在。 1：使能 0：禁止
4	CMB0E	rw	0x00	通道 0B 比较匹配触发 ADC 使能。 1：使能 0：禁止
3	CMA2E	rw	0x00	通道 2A 比较匹配触发 ADC 使能，只有 TIM3 存在。 1：使能 0：禁止

Bit	Field	Type	Reset	Description
2	CMA1E	rw	0x00	通道 1A 比较匹配触发 ADC 使能，只有 TIM3 存在。 1：使能 0：禁止
1	CMA0E	rw	0x00	通道 0A 比较匹配触发 ADC 使能。 1：使能 0：禁止
0	UEVE	rw	0x00	事件更新触发 ADC 使能。 1：使能 0：禁止

15.6.9 通道 0 控制寄存器 (TIMx_CRCH0)

起始地址：Timer0:0x40000C00 Timer1:0x40000D00 Timer2:0x40000E00 Timer3:0X40005800

地址偏移：0x24

复位值：0x00003000

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
</table>

<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>CCGB</td><td>CCGA</td><td colspan="2">CISB</td><td>CDEB</td><td>CDEA</td><td>CIEB</td><td>CIEA</td><td>-</td><td>-</td><td>CSB</td><td>CSA</td><td>CFB</td><td>CRB</td><td>CFA</td><td>CRA</td></tr>
<tr><td>CCGB</td><td>CCGA</td><td colspan="2">CISB</td><td>CDEB</td><td>CDEA</td><td>CIEB</td><td>CIEA</td><td>BUFEB</td><td>BUFEA</td><td>CSB</td><td>CSA</td><td colspan="2">bksb</td><td colspan="2">bksa</td></tr>
<tr><td>RW</td><td>RW</td><td colspan="2">RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td colspan="2">RW</td><td colspan="2">RW</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。
15	CCGB	rw	0x00	捕获比较 B 软件触发，硬件自动清零。 在比较模式下只产生中断； 在捕获模式下产生中断并且捕获计数器的值到捕获寄存器中。
14	CCGA	rw	0x00	捕获比较 A 软件触发，硬件自动清零。 在比较模式下只产生中断； 在捕获模式下产生中断并且捕获计数器的值到捕获寄存器中。
13:12	CISB	rw	0x00	B 通道比较匹配设置。 00 无匹配；01 上升匹配；10 下降匹配；11 双匹配
11	CDEB	rw	0x00	B 捕获比较触发 DMA 使能。 0：禁止 1：使能
10	CDEA	rw	0x00	A 捕获比较触发 DMA 使能。 0：禁止 1：使能

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
9	CIEB	rw	0x00	B 捕获比较触发中断使能。 0：禁止 1：使能
8	CIEA	rw	0x00	A 捕获比较触发中断使能。 0：禁止 1：使能
7	BUFEB	rw	0x00	比较功能：B 比较缓存使能控制。 0：禁止 1：使能
6	BUFEA	rw	0x00	比较功能：A 比较缓存使能控制。 0：禁止 1：使能
5	CSB	rw	0x00	B 通道捕获/比较功能选择。 0：比较模式 1：捕获模式
4	CSA	rw	0x00	A 通道捕获/比较功能选择。 0：比较模式 1：捕获模式
3	CFB	rw	0x00	B 通道下降沿捕获使能。 0：禁止 1：使能
2	CRB	rw	0x00	B 通道上升沿捕获使能。 0：禁止 1：使能
3:2	BKSB	rw	0x00	B 通道比较功能输出刹车电平控制。 00：高阻输出 01：对输出无影响 10：强制输出低电平 11：强制输出高电平;
1	CFA	rw	0x00	A 通道下降沿捕获使能。 0：禁止 1：使能
0	CRA	rw	0x00	A 通道上升沿捕获使能。 0：禁止 1：使能
1:0	BKSA	rw	0x00	A 通道比较功能输出刹车电平控制。 00：高阻输出 01：对输出无影响 10：强制输出低电平 11：强制输出高电平;

15.6.10 通道 1/2 控制寄存器 (TIM3_CRCH1/2)(仅 TIM3 存在)

起始地址：Timer3:0X40005800

地址偏移：TIM3_CRCH1：0x028 TIM3_CRCH2：0x02C

复位值：0x00003000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCGB	CCGA	CISB		CDEB	CDEA	CIEB	CIEA	-	-	CSB	CSA	CFB	CRB	CFA	CRA
CCGB	CCGA	CISB		CDEB	CDEA	CIEB	CIEA	BUFEB	BUFEA	CSB	CSA	bksb		bksa	
RW	RW	RW		RW	RW	RW	RW	RW	RW	RW	RW	RW		RW	

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。
15	CCGB	rw	0x00	捕获比较 B 软件触发，硬件自动清零。 在比较模式下只产生中断。 在捕获模式下产生中断并且捕获计数器的值到捕获寄存器中。
14	CCGA	rw	0x00	捕获比较 A 软件触发，硬件自动清零。 在比较模式下只产生中断。 在捕获模式下产生中断并且捕获计数器的值到捕获寄存器中。
13:12	CISB	rw	0x00	B 通道比较匹配设置。 00 无匹配；01 上升匹配；10 下降匹配；11 双匹配
11	CDEB	rw	0x00	B 捕获比较触发 DMA 使能。 0：禁止 1：使能
10	CDEA	rw	0x00	A 捕获比较触发 DMA 使能。 0：禁止 1：使能
9	CIEB	rw	0x00	B 捕获比较触发中断使能。 0：禁止 1：使能
8	CIEA	rw	0x00	A 捕获比较触发中断使能。 0：禁止 1：使能
7	BUFEB	rw	0x00	比较功能：B 比较缓存使能控制。 0：禁止 1：使能
6	BUFEA	rw	0x00	比较功能：A 比较缓存使能控制。 0：禁止 1：使能
5	CSB	rw	0x00	B 通道捕获/比较功能选择。 0：比较模式 1：捕获模式
4	CSA	rw	0x00	A 通道捕获/比较功能选择。 0：比较模式 1：捕获模式
3	CFB	rw	0x00	B 通道下降沿捕获使能。 0：禁止 1：使能
2	CRB	rw	0x00	B 通道上升沿捕获使能。 0：禁止 1：使能

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
3:2	BKSB	rw	0x00	B 通道比较功能输出刹车电平控制。 00：高阻输出 01：保持之前输出 I 10：强制输出低电平 11：强制输出高电平
1	CFA	rw	0x00	A 通道下降沿捕获使能。 0：禁止 1：使能
0	CRA	rw	0x00	A 通道上升沿捕获使能。 0：禁止 1：使能
1:0	BKSA	rw	0x00	A 通道比较功能输出刹车电平控制。 00：高阻输出 01：保持之前输出 I 10：强制输出低电平 11：强制输出高电平

15.6.11 死区寄存器 (TIMx_DTR)

起始地址：Timer0:0x40000C00 Timer1:0x40000D00 Timer2:0x40000E00 Timer3:0X40005800

地址偏移：0X30

复位值：0x00000000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	VCE	Safeen	MOE	AOE	BKE	DTEN	Bksel	DTR							
	RW	RW	RW	RW	RW	RW	RW	RW							

Bit	Field	Type	Reset	Description
31:15	Reserved			保留。
14	VCE	rw	0x00	VC Brake enable。 0：禁止 1：使能
13	Safeen	rw	0x00	Safety 刹车使能（osc fail,brown down,lockup）。 0：禁止 1：使能
12	MOE	rw	0x00	PWM 输出使能。 0：禁止 1：使能
11	AOE	rw	0x00	PWM 输出自动使能。 0：禁止 1：使能
10	BKE	rw	0x00	刹车使能。 0：禁止 1：使能

Bit	Field	Type	Reset	Description
9	DTEN	rw	0x00	死区控制使能。 0：禁止 1：使能
8	bksel	rw	0x00	刹车选择。 0：使用自己的刹车控制； 1：TIM1/2 使用 TIM0 的的刹车控制 注：TIM0/TIM3 选择无效。
7:0	DTR	rw	0x00	死区时间寄存器。 DTR[7:5]　T　取值范围　step 0xx　DTR[6:0]+2　2-129　1 10x　DTR[5:0]+64*2+2　130-256　2 110　DTR[4:0]+32*8+2　258-506　8 111　DTR[4:0]+32*16+2　514-1010　16

15.6.12 重复计数寄存器 (TIMx_RCR)

起始地址：Timer0:0x40000C00　Timer1:0x40000D00　Timer2:0x40000E00　Timer3:0X40005800

地址偏移：0X34

复位值：0x00000000

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

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

Bit	Field	Type	Reset	Description
31:8	Reserved			保留。
7:0	RCR	rw	0x00	重复周期计数值。 设置 RCR+1 个周期个上溢出下溢出后产生事件更新，计数器上溢出或下溢出时内部 RCR_CNT 减 1，当计数到零后 RCR_CNT 重载 RCR 的值，并且产生事件更新 UEV 信号。

15.6.13 通道 0 比较捕获寄存器 (TIMx_CCR0A/B)

起始地址：Timer0:0x40000C00　Timer1:0x40000D00　Timer2:0x40000E00　Timer3:0X40005800

地址偏移：TIMx_CCR0A：0x03C　　TIMx_CCR0B：0x040

复位值：0x00000000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR0y															
RW															

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。
15:0	CCR0y	rw	0x00	比较捕获寄存器，比较具有缓存功能 (y=A,B)。

15.6.14 通道 1/2 比较捕获寄存器 (TIM3_CCR1/2 A/B)(仅 TIM3 存在)

起始地址：Timer0:0x40000C00 Timer1:0x40000D00 Timer2:0x40000E00 Timer3:0X40005800

地址偏移：TIM3_CCR1A：0x044 TIM3_CCR1B：0x048 TIM3_CCR2A：0x04C TIM3_CCR2B：0x050

复位值：0x00000000

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。
15:0	CCRxy	rw	0x00	比较捕获寄存器，比较具有缓存功能 (x=1,2；y=A,B)。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

16 低功耗定时器（LPTIM）

16.1 LPTimer 简介

LPTimer 是异步 16 位定时/计数器，在系统时钟关闭后仍然可以通过内部低速 RC 或者外部低速晶体振荡计时/计数。通过中断在低功耗模式下唤醒系统。

LPTimer 支持最大 256 的预分频，系统内有两个低功耗定时器，可以通过内部进行级联。

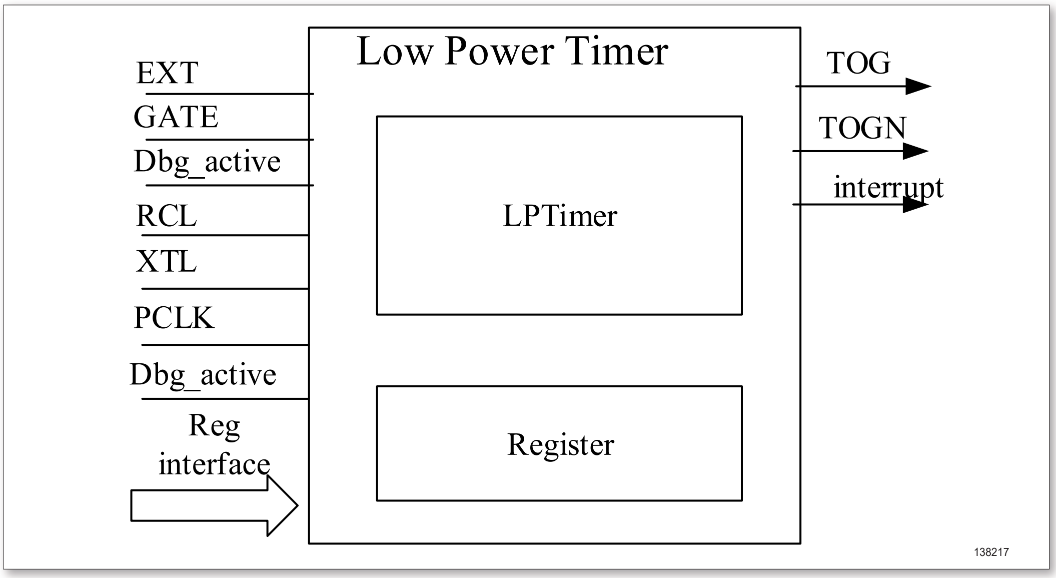

图 101. LPTimer 框图

16.2 LPTimer 功能描述

LPTimer 支持 2 种工作模式，每个定时器/计数器都有独立的控制启动信号，以及外部输入时钟，门控信号。

LPTimer 使用 EXT，GATE 来进行计数功能，EXT 用于计数器的外部输入时钟信号，Gate 用于有效电平计数使能信号。

LPTimer 的定时器支持两种工作模式，通过设置定时器控制寄存器（CR）中 MD 选择工作模式。模式 1 为 16 位自由计数模式。模式 2 是 16 位重载模式。LPTimer 启动时会自动装载重载寄存器 ARR 的值到计数器中。

LPTimer 可选三种时钟作为定时器时钟，通过控制寄存器 CR.TCK_SEL 选择。默认选择 PCLK。时钟选择如表：

表 61: TCK_SEL 时钟选择

TCK_SEL	00	01	10	11
定时器时钟	PCLK	PCLK	XTL	RCL
读定时器计数值	读经过同步	无同步	读经过同步	读经过同步

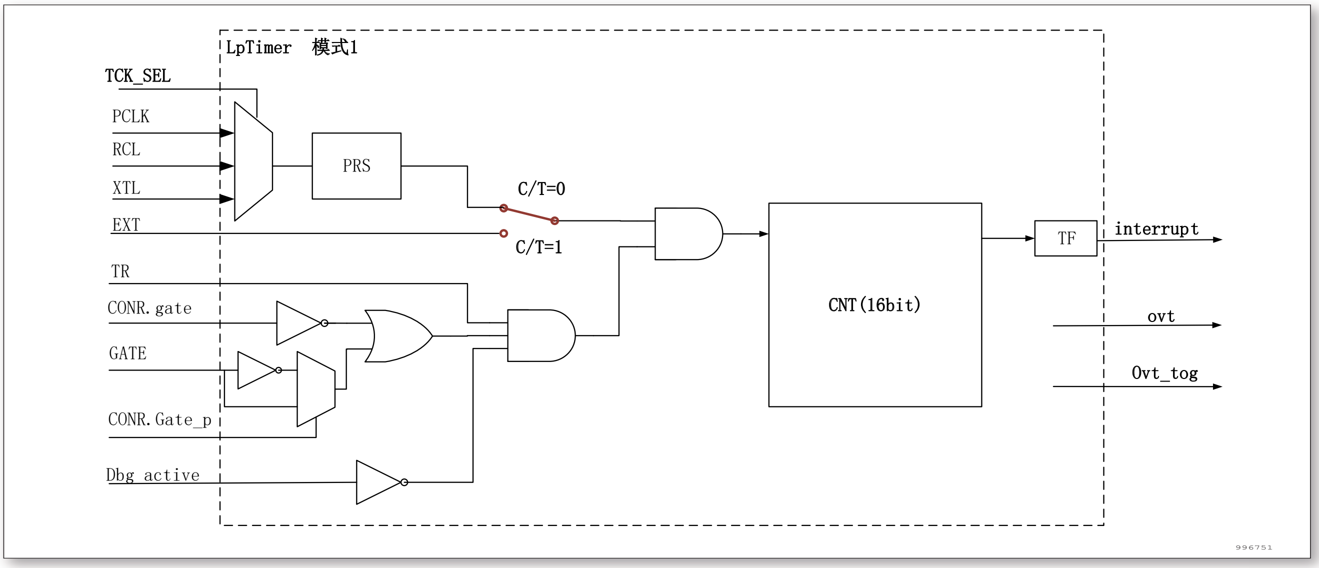

图 102. LPTimer 模式 1

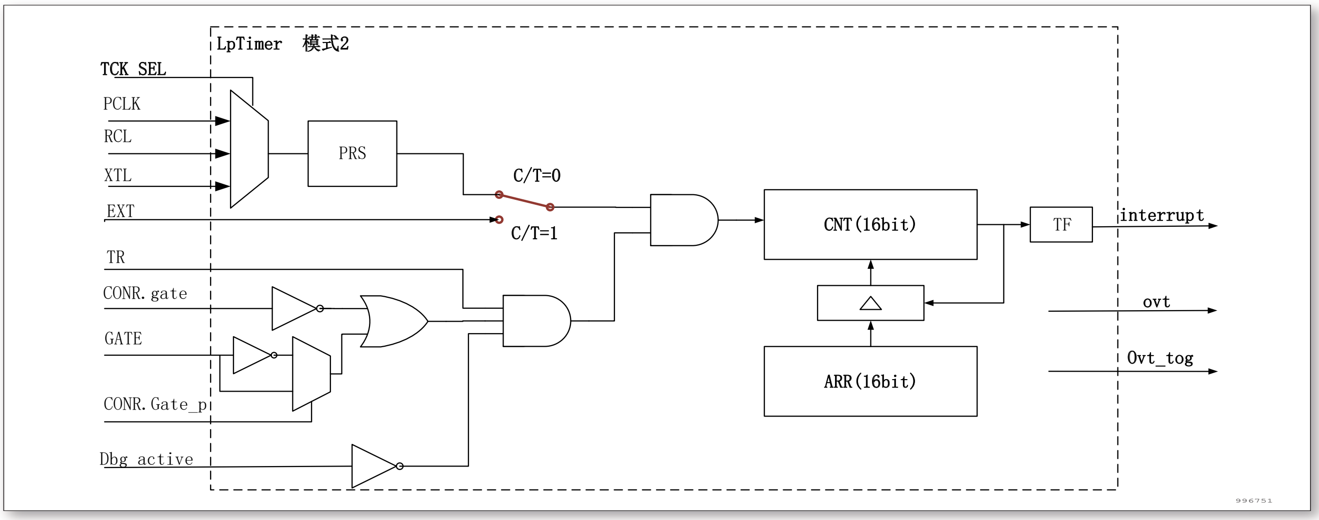

图 103. LPTimer 模式 2

当设置对应定时器 TR 为 1 后定时器开始运行。计数器从设定的值开始计数，计数到最大 0XFFFF 后产生溢出中断。模式 1 产生中断后计数器清零继续向上计数，模式 2 产生中断后重载寄存器 ARR 的值到计数器中，向上计数。计数器 CNT 的初值在启动定时器时由 ARR 自动装载。

16.2.1 计数功能

计数功能用于测定某个事件发生的次数。在计数功能中，计数器在每个相应的输入时钟的下降沿累加一次。输入信号被内部的计数时钟采样，因此外部输入时钟频率不能超过系统的计数时钟。计数到最大值会溢出并且产生中断。

16.2.2 定时功能

定时功能用于产生间隔定时。在定时功能中，定时器一个时钟累加一次，计数到最大值会溢出并且产生中断。

定时时钟周期数为 (0XFFFF-ARR+0X0001)

16.3 LPTimer 互连

16.3.1 GATE 互联

GATE 输入可以从端口通过 PX_SEL 选择直接输入，也可通过端口功能寄存器 GPIO_TIMGS 选择可以连通到其他模块或端口。

当 LPTIM_Gx=0x0 时，门控 GATE 输入为 PX_SEL 选择的端口输入，当 LPTIM_Gx=0x1~0x7 时，连接其他模块的输入或输出。

表 62: LPTIM_Gx 说明

	LPTIM0_G		LPTIM1_G
000	PX_SEL	000	PX_SEL
001	LPUART0_RXD	001	UART0_RXD
010	LPUART1_RXD	010	LPUART0_RXD
011	VC0_OUT	011	VC1_OUT
100	VC1_OUT	100	VC2_OUT
101	PB03	101	PD05
110	PB05	110	PE07
111	PC00	111	PC06

16.3.2 EXT 互联

EXT 输入可以从端口直接输入，也可通过端口功能寄存器 GPIO_TIMES 选择可以连通到其他模块或端口。

当 LPTIM_Ex=0x0 时，外部时钟 EXT 输入为 PX_SEL 选择的端口输入，当 LPTIM_Ex=0x1~0x7 时，连接其他模块的输入或输出。

表 63: LPTIM_Ex 说明

	LPTIM0_E		LPTIM1_E
000	PX_SEL	000	PX_SEL
001	PCNT_S0	001	UART1_RXD
010	LVD_OUT	010	LPUART1_RXD
011	VC0_OUT	011	VC0_OUT
100	VC1_OUT	100	VC2_OUT
101	PB04	101	LPTIM0_TOG
110	PB06	110	PE08
111	PC03	111	PC07

设置 LPTIM1 的定时器的外部输入 LPTIM1_E 为 0x5, 选择 LPTIM0 的翻转输出 LPTIM0_TOG 为外部时钟输入，LPTIM1 设置为外部计数模式。这样两个定时器就可以级联使用。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

16.3.3 Toggle 输出互联

LPTimer 的翻转输出到端口上，可以驱动 Buzzer 实现蜂鸣器的控制。

16.4 LPTimer 寄存器描述

表 64. LPTimer 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0X00	LPTIMx_CNT	LPTimer 计数值只读寄存器	0x00000000	小节 16.4.1
0X04	LPTIMx_ARR	LPTimer 重载寄存器	0x00000000	小节 16.4.2
0X0C	LPTIMx_CR	LPTimer 控制寄存器	0x00000008	小节 16.4.3
0X10	LPTIMx_IFR	LPTimer 中断标志	0x00000000	小节 16.4.4
0X14	LPTIMx_ICLR	LPTimer 中断清除寄存器	0x00000001	小节 16.4.5

16.4.1 LPTimer 计数值只读寄存器 (LPTIMx_CNT)

起始地址：LPTIMER0:0X40000F00 LPTIMER1:0X40000F40

地址偏移：0X00

复位值：0x00000000

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。
15:0	CNT	r	0x00	低功耗定时器计数值只读寄存器

16.4.2 LPTimer 重载寄存器 (LPTIMx_ARR)

起始地址：LPTIMER0:0X40000F00 LPTIMER1:0X40000F40

地址偏移：0X04

复位值：0x00000000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。
15:0	ARR	rw	0x00	低功耗定时器重载寄存器。 写 ARR 前需要读取 CR.WT_FLAG，当且仅当 WT_FLAG 为 1 时，写入时才能写入数据。写 ARR 寄存器后 WT_FLAG 会变低。 注：在定时器启动时或者重载时这个值会装载到定时器计数器中。定时器计数器值不可软件更改。

16.4.3 LPTimer 控制寄存器 (LPTIMx_CR)

起始地址：LPTIMER0:0X40000F00　LPTIMER1:0X40000F40

地址偏移：0X0C

复位值：0x00000008

31:11	10	9	8	7	6	4:5	3	2	1	0
Reserved	IE	GATE_P	GATE	WT_FLAG	Reserved	TCK_SEL	TOG_EN	CT	MD	TR
	RW	RW	RW	R		RW	RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31:14	Reserved			保留。
13:11	PRS	rw	0x00	预除频（在定时器运行时不可以更改）。 000 : 1　001 : 2 010 : 4　011 : 8 100 : 16　101 : 32 110 : 64　111 : 256
10	IE	rw	0x00	中断使能控制，写 1 后使能中断。
9	GATE_P	rw	0x00	GATE 极性控制，默认高电平 gate 有效，设置为 1 后低电平有效。
8	GATE	rw	0x00	定时器门控。 0：无门控，TR=1 时定时器工作； 1: 只有门控为 1 并且 TR=1 时才工作；

Bit	Field	Type	Reset	Description
7	WT_FLAG	r	0x00	WT, 写同步标志。 0：正在同步，此时写 ARR 无效 1：同步完成，此时可更改 ARR
6	Reserved			保留。
5:4	TCK_SEL	rw	0x00	LPTimer 时钟选择 00 PCLK; 10:XTL; 11 ,RCL。
3	TOG_EN	rw	0x01	TOG 输出使能。 0：TOG,TOGN 同时输出 0 1：TOG,TOGN 输出相位相反的信号。可供 buzzer 使用。
2	CT	rw	0x00	计数器/定时器功能选择。 0：定时器功能，定时器使用 TCK_SEL 选择的时钟进行计数。 1：计数器功能，计数器使用外部输入的下降沿进行计数。采样时钟使用 TCK_SEL 选择的时钟，外部输入时钟要低于 1/2 采样时钟。
1	MD	rw	0x00	定时器工作模式。 0：模式 1 无重载模式 16 位计数器/定时器 1：模式 2 自动重装载 16 位计数器/定时器
0	TR	rw	0x00	定时器运行控制位。 0：定时器停止 1：定时器运行

16.4.4 LPTimer 中断标志 (LPTIMx_IFR)

起始地址：LPTIMER0:0X40000F00 LPTIMER1:0X40000F40

地址偏移：0X10

复位值：0x00000000

31:8	7	6	5	4	3	2	1	0
Reserved								TF
								RO

Bit	Field	Type	Reset	Description
31:1	Reserved			保留。
0	TF	r	0x00	中断标志, 硬件置位。写清除寄存器清零。

16.4.5 LPTimer 中断清除寄存器 (LPTIMx_ICLR)

起始地址：LPTIMER0:0X40000F00 LPTIMER1:0X40000F40

地址偏移：0X14

复位值：0x00000001

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

<table>
<tr><th>31:8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td colspan="8" rowspan="2">Reserved</td><td>TFC</td></tr>
<tr><td>R1W0</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:1	Reserved			保留。
0	rc_w0	r	0x01	中断标志清除，写 0 清除，写 1 无效。

17 可编程计数阵列（PCA）

17.1 PCA 简介

PCA(可编程计数器阵列 Programmable Counter Array) 支持最多 5 个 16 位的捕获/比较模块。该定时/计数器可用作为一个通用的时钟计数/事件计数器的捕获/比较功能。PCA 的每个模块都可以进行独立编程，以提供输入捕捉，输出比较或脉冲宽度调制。另外模块 4 有额外的看门狗定时器模式。

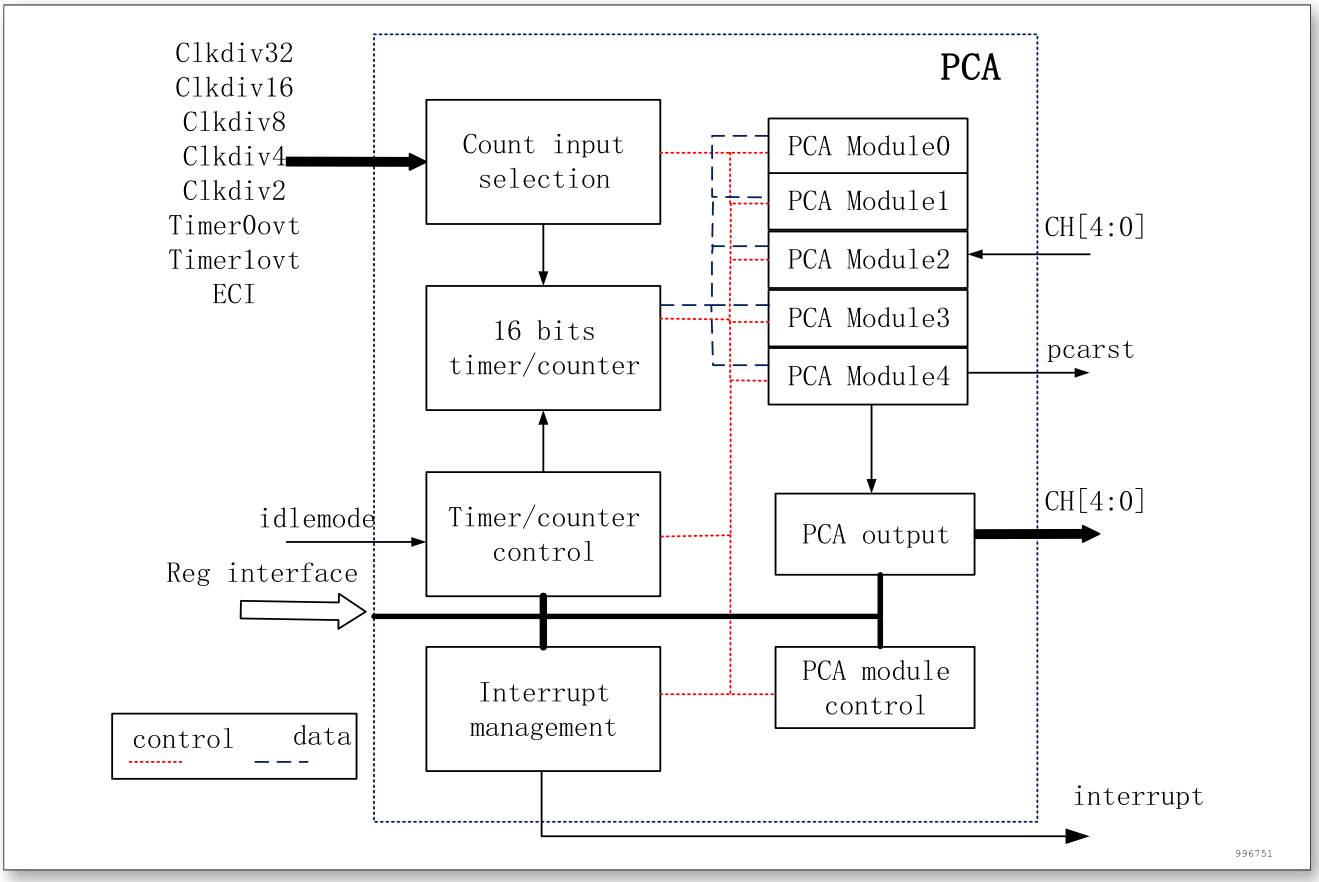

图 104. PCA 整体框图

17.2 PCA 功能描述

五个模块都可被配置为独立工作，有三种工作方式：边沿触发捕捉、输出比较、8/16 位脉宽调制。每个模块在系统控制器中都有属于自己的功能寄存器，这些寄存器用于配置模块的工作方式和与模块交换数据。

每组比较/捕获模块是由一个比较/捕获寄存器组 (CCAPx)，以及 1 个 16 位比较器，和各种逻辑门控制组成。寄存器组用来存储时间或次数，针对外部触发捕获条件，或内部触发比较条件。在 8 位 PWM 模式下，寄存器 (CCAPxL) 用来控制输出波形的占空比，CCAPxH 为 8 位比较缓存。16 位 PWM 模式下 CARR 用于控制 PWM 输出的周期，CCAPx 寄存器控制占空比。

每个模块都可以独立编程的操作在任何以下模式：

- 16 位捕获模式的上升沿，下降沿或任意沿触发。
- 比较模式：16 位高速输出，16 位看门狗定时器（模块 4）或 16/8 位脉冲宽度调制。
- 关闭状态。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

比较/捕获模块模式寄存器（CCAPMx）确定相应的工作模式。对于比较/捕获模块进行编程时，他们是基于共同的时间计数。定时器/计数器打开和关闭通过 CCON.CR 位即可控制 PCA 定时/计数器的运行。在一个比较/捕获模块捕获，软件定时器，高速输出，设置模块的比较/捕获标志（CCON.CCFx），并产生 PCA 中断请求，如果相应的使能位在 CCAPMx 寄存器设置。CPU 可以在任何时候读写 CCAPx 寄存器。

17.2.1 PCA 定时/计数器

CNT 的这组特殊功能寄存器可用作为一个 16 位定时器/计数器。这是一个 16 位向上计数的计数器。如果 CMOD.CFIE 位被置“1”时，当 CNT 溢出时硬件自动设置 PCA 溢出标志（CCON.CF）并产生 PCA 中断请求。CMOD.CPS[2：0] 三位选择八个信号输入到定时器/计数器。

- 系统时钟 PCLK 的 32 分频。
- 系统时钟 PCLK 的 16 分频。
- 系统时钟 PCLK 的 8 分频。
- 系统时钟 PCLK 的 4 分频。
- 系统时钟 PCLK 的 2 分频。
- 定时器 0 的溢出。每次定时器 0 计数溢出后，CNT 就递增，这样提供了 PCA 的可变编程频率输入。
- 定时器 1 的溢出。每次定时器 1 计数溢出后，CNT 就递增，这样提供了 PCA 的可变编程频率输入。
- ECI。CPU 每过 4 个 PCLK 时钟周期就对 PCAECI 进行采样，当每次采样结果从高变低时，CL 自动加 1，因此最高的 ECI 输入频率不能高于系统时钟 PCLK 的 1/8, 以满足采样需求。

设置运行控制器（CCON.CR）启动 PCA 定时/计数器。当 CMOD.CIDL 置“1”后，PCA 定时器/计数器可以继续运行在空闲模式下。CPU 可以随时读取 CNT 的数值，但当计数启动后（CCON.CR=1）时，为了防止计数错误，CNT 是禁止写入的。

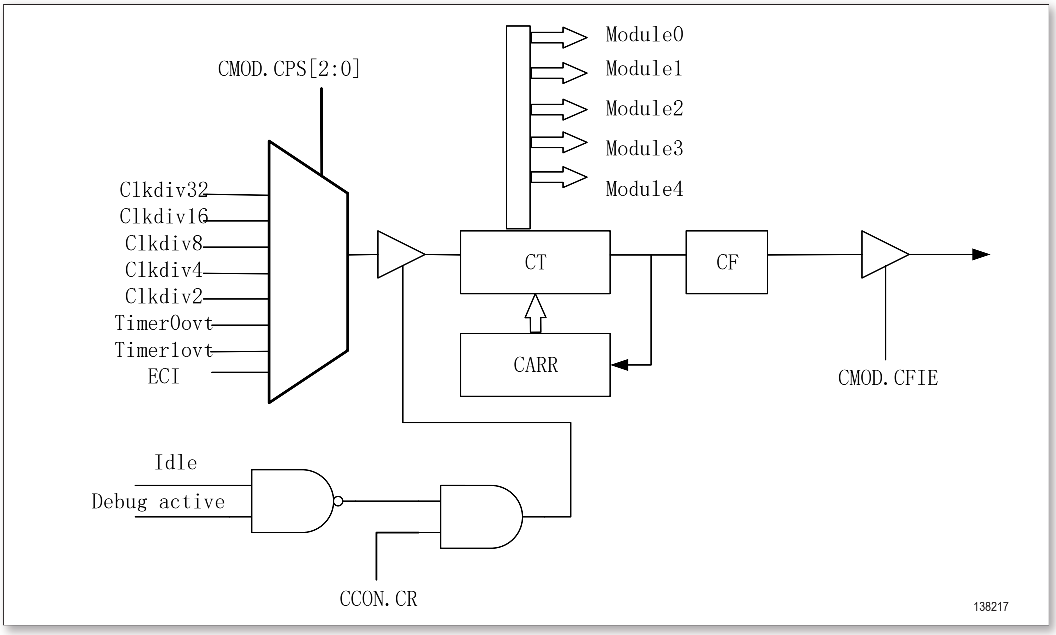

图 105. PCA 计数器框图

16 位自由计数模式

计数器计数到最大值 0XFFFF 溢出后计数器的值变为 0，重新开始向上计数，如果需要更改计数周期值可以停止 PCA 更改计数器的初值然后继续计数。可用于 PCA 捕获模式，8 位 PWM 模式。

- 设置流程
  1. 保持 PCA_EPWM 的 EPMW 为 0。
  2. 设置 PCA_CMOD.CPS 选择计数时钟。
  3. 根据需要设置 PCA_CMOD.CFIE 设置计数溢出中断使能。
  4. 时钟 PCA_CCON.CR 启动 PCA 计数器。
  5. 溢出更改计数器初值需要停止 PCA 计数器。

16 位重载计数模式

计数器计数到与寄存器 CARR 的值相同时溢出后计数器的值变为 0，继续开始向上计数，可用于 PCA 捕获模式，16 位 PWM 模式。

- 设置流程
  1. 设置 PCA_EPWM 的 EPMW 为 1。
  2. 设置 PCA_CARR 设置计数周期值。
  3. 设置 PCA_CMOD.CPS 选择计数时钟。
  4. 根据需要设置 PCA_CMOD.CFIE 设置计数溢出中断使能。
  5. 时钟 PCA_CCON.CR 启动 PCA 计数器。

17.3 PCA 捕获功能

PCA 捕获模式提供了 5 路 PCA 测量脉冲周期，脉冲宽度，占空比和相位差的功能。引脚上出现的电平跳变导致 PCA 捕捉 PCA 计数器/定时器的值并将其装入到对应模块的 16 位捕捉/比较寄存器（CCAPx）。CCAPMx.CAPP 以及 CCAPMx.CAPN 位用于选择触发捕捉的电平变化类型：低电平到高电平（正沿）、高电平到低电平（负沿）或任何变化（正沿或负沿）。当捕捉发生时，CCON 中的捕捉/比较标志（CCFn）被置为逻辑‘1’并产生一个中断请求（如果 CCF 中断被允许）。当 CPU 转向中断服务程序时，CCFn 位不能被硬件自动清除，用户软件写 INTCL 寄存器清除此标志位。如果 CCPMx.CAPP 以及 CCAPMx.CAPN 位都被设置为逻辑‘1’，可以通过直接读对应端口引脚的状态来确定本次捕捉是由上升沿触发还是由下降沿触发。

分辨率等于定时器/计数器的时钟。输入信号必须在高电平或低电平期间至少保持 2 个时钟周期，以保证输入信号能够被硬件识别。

CPU 可以在任何时候读取或写入 CCAPx 的寄存器。

捕获设置：

- 当需要在外部上升沿进行捕获，CCPMx.CAPP = “1” 以及 CCAPMx.CAPN = “0”。
- 当需要在外部下降沿进行捕获，CCPMx.CAPP = “0” 以及 CCAPMx.CAPN = “1”。
- 当需要在外部上升、下降沿进行捕获，CCPMx.CAPP = “1” 以及 CCAPMx.CAPN = “1”。
- 根据需要配置捕获中断及中断处理程序。
- 按照定时/计数器配置 PCA 计数器启动。

注意：

随后由同一模块的捕获值会覆盖现有捕获的值。为了保持捕获的值，在中断服务程序中将它保存在 RAM 里面，这个操作必须在下一次事件出现之前完成，否则就会丢失前面一次捕获采样值。

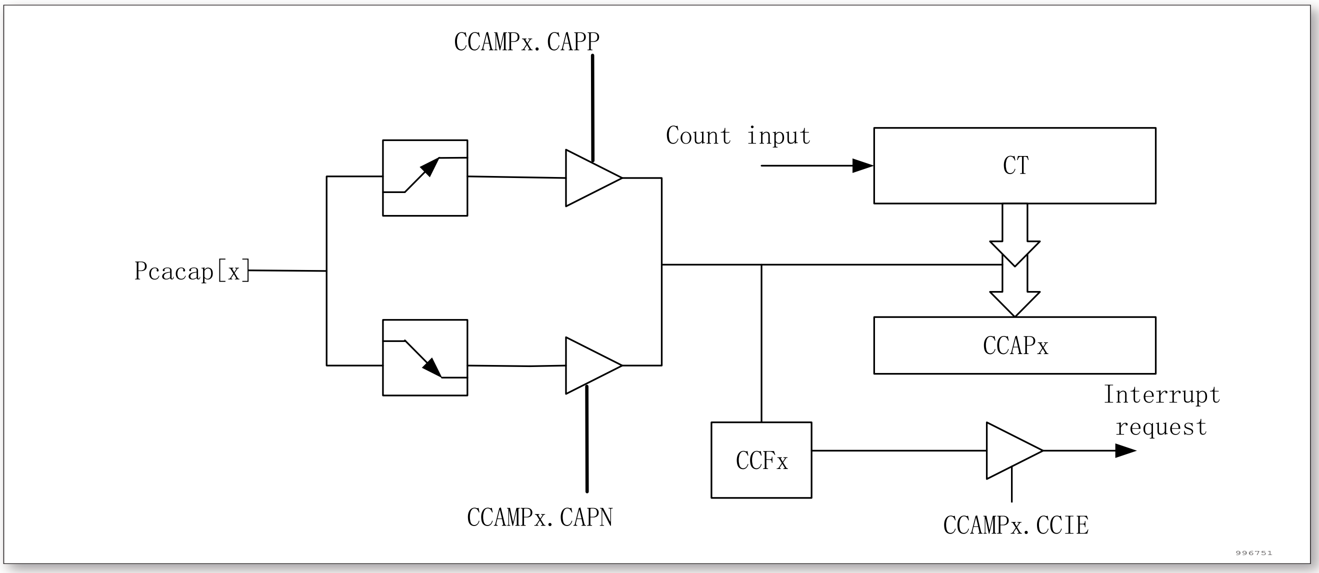

图 106. PCA 捕获功能框图

17.3.1 PCA 比较功能

比较功能提供如下功能，高速输出模式，WDT 模式，16 为 PWM 模式和 8 位 PWM 模式。在前三个功能中，比较/捕获模块比较 16 位 PCA 定时器/计数器的值与预先加载到该模块的 CCAPx 寄存器中的 16 位值。在 8 位 PWM 模式下，PCA 模块不断地将 PCA 定时器/计数器低字节寄存器（CNT）与一个在 CCAPxL 模块寄存器 8 位的值进行比较。每 4 个时钟周期比较一次，即与最快的 PCA 定时器/计数器的时钟速率相匹配。设置 CCAPMx.ECOM 位选择该模块的比较功能。

若要正确使用在比较模式下的模块，请遵守以下的一般程序：

- 选择 PCA 模块的操作模式。
- 选择 PCA 定时器/计数器的输入信号。
- 比较值加载到模块的比较/捕获寄存器对。
- 设置 PCA 定时器/计数器运行控制位。
- 匹配后产生中断，清除模块的比较/捕获标志。

比较翻转输出模式

在比较翻转输出模式，每当 PCA 计数器内的值与模块的 16 位捕捉/比较寄存器（CCAPx）发生匹配时，模块 PCA 的 CH[x] 引脚上的逻辑电平将发生变化。这可以提供比切换 IO 输出有更高精度，因为这个输出不会被中断响应而影响输出频率，靠 CPU 来切换 IO 输出的话，功耗，精度都有所欠缺。

要设定一个比较/捕获模块的比较翻转输出模式，设置 CCAPMx.ECOM，CCAPMx.MAT 和 CCAPMx.TOG 位。PCA 定时器/计数器和比较/捕获的寄存器（CCAPx）之间的匹配切换 PCA 的 CH[x] 信号，并设置模块的比较/捕获标志（CCON.CCFx）。通过软件设置或清除 PCA 的 CH[x] 信号，用户可以选择匹配切换信号从低到高或高到低。

用户也可以选择产生一个中断请求，通过设置相应的中断使能位（CCAPMx.CCIE）当匹配发生时，即可产生中断请求。由于硬件无法清除的比较/捕获标志中断，用户必须在软件中清除这个标志位。如果用户在中断程序中不去改变比较/捕获寄存器，PCA 并重新计数比较值，如相匹配则发生下一次翻转。在中断服务程序中，一个新的 16 位比较值可以被写入比较/捕获的寄存器（CCAPx）。

注意:

为了防止无效的匹配，而更新这些寄存器，用户软件应该写 CCAPxL 的首先然后 CCAPxH。写到 CCAPxL 清除禁用比较功能 ECOM 位，而写到 CCAPxH 设置的 ECOM 位，重新启用比较功能。

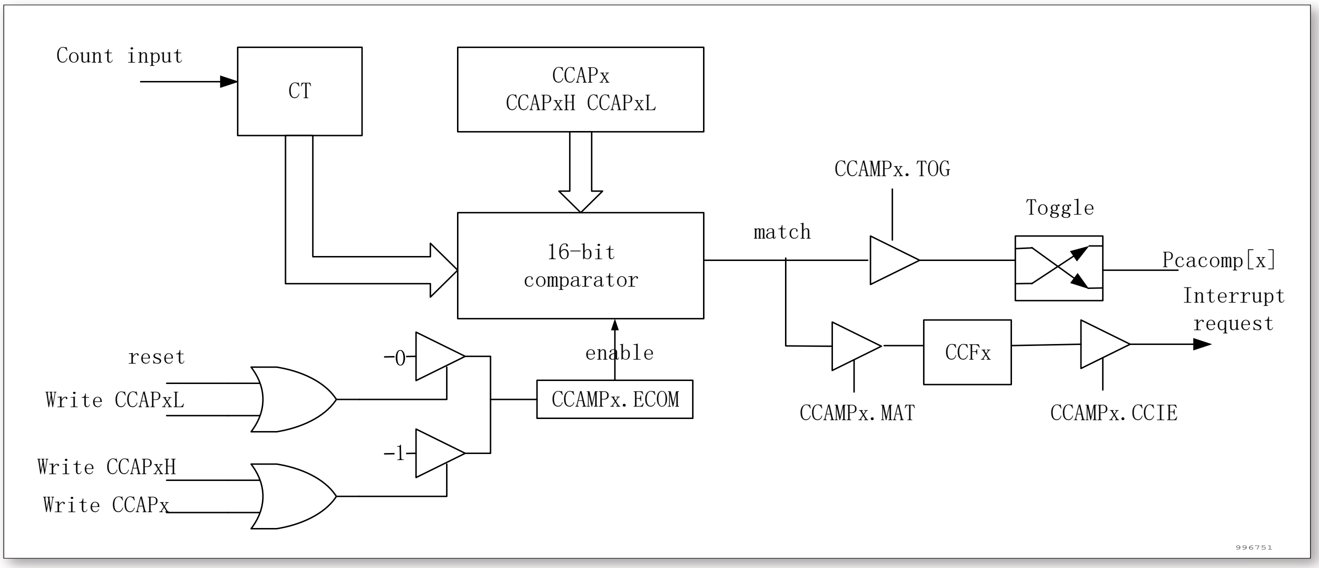

图 107. PCA 比较功能框图

PCA 16 位 PWM 功能 在比较输出 PWM 模式,每当 PCA 计数器内的值与模块的 16 位捕捉/比较寄存器（CCAPx）发生匹配时，模块 PCA 的 CH[x] 引脚上的逻辑电平将发生变化。计数器溢出时，CH[x] 引脚上的逻辑电平将清零。这可以提供一组 16 位的 PWM 输出。

要设定一个比较/捕获模块的 PWM 模式，设置 CCAPMx.ECOM，CCAPMx.MAT 和 CCAPMx.TOG 位及 EPWM 寄存器。PCA 定时器/计数器和比较/捕获的寄存器（CCAPx）之间的匹配切换 PCA 的 CH[x] 信号，并设置模块的比较/捕获标志（CCON.CCFx）。用户也可以选择产生一个中断请求，通过设置相应的中断使能位（CCAPMx.CCIE）当匹配发生时，即可产生中断请求。由于硬件无法清除的比较/捕获标志中断，用户必须在软件中清除这个标志位。

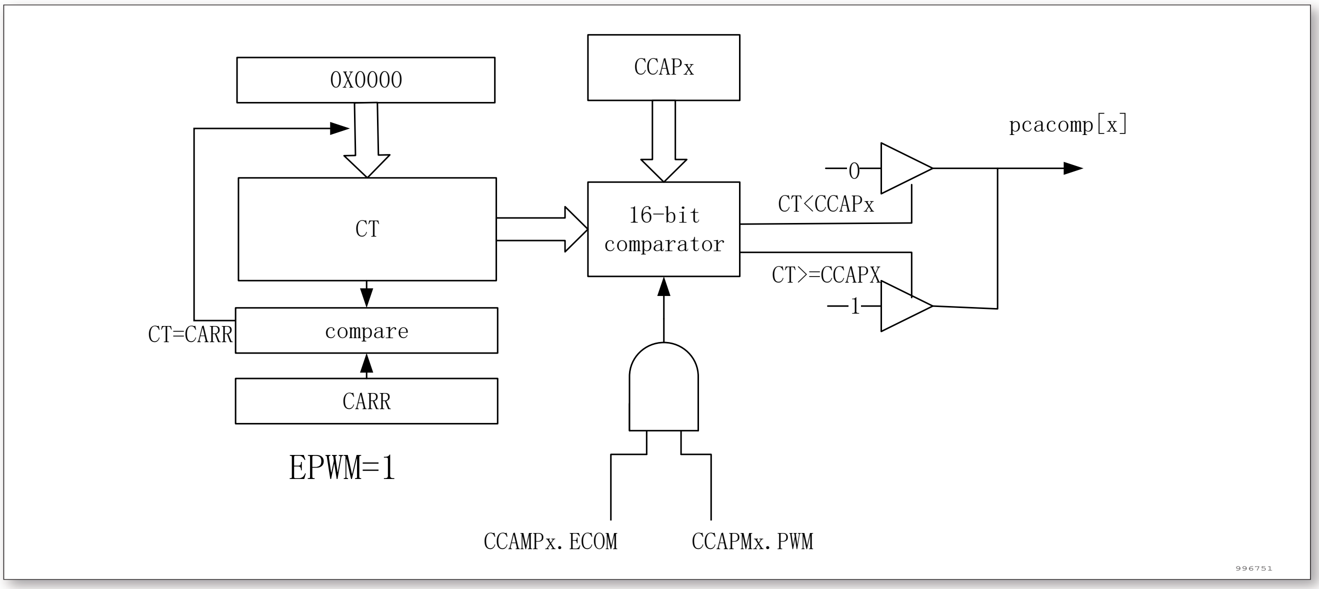

图 108. PCA 16 位 PWM 功能框图

PCA 模块 4 的 WDT 功能

本系列除了一个 WDT 硬件模块，PCA 的模块 4，还提供一个可编程频率的 16 位 WDT。当 PCA 定时器/计数器的计数值与模块 4 中存储的值比较/捕获寄存器（CCAP4）相匹配时，这种模式产生复位信号。PCA 的 WDT 复位信号做为一个独立的复位信号。与外部复位（RST），硬件看门狗复位（WDTRST）和 LVD 低电压复位，POR 上电下电复位相结合。用户可以自由结合或单独使用它们。模块 4 是具有 WDT 模式唯一 PCA 的模块。当不设置为 WDT 时，它可以在其它模式中独立使用。

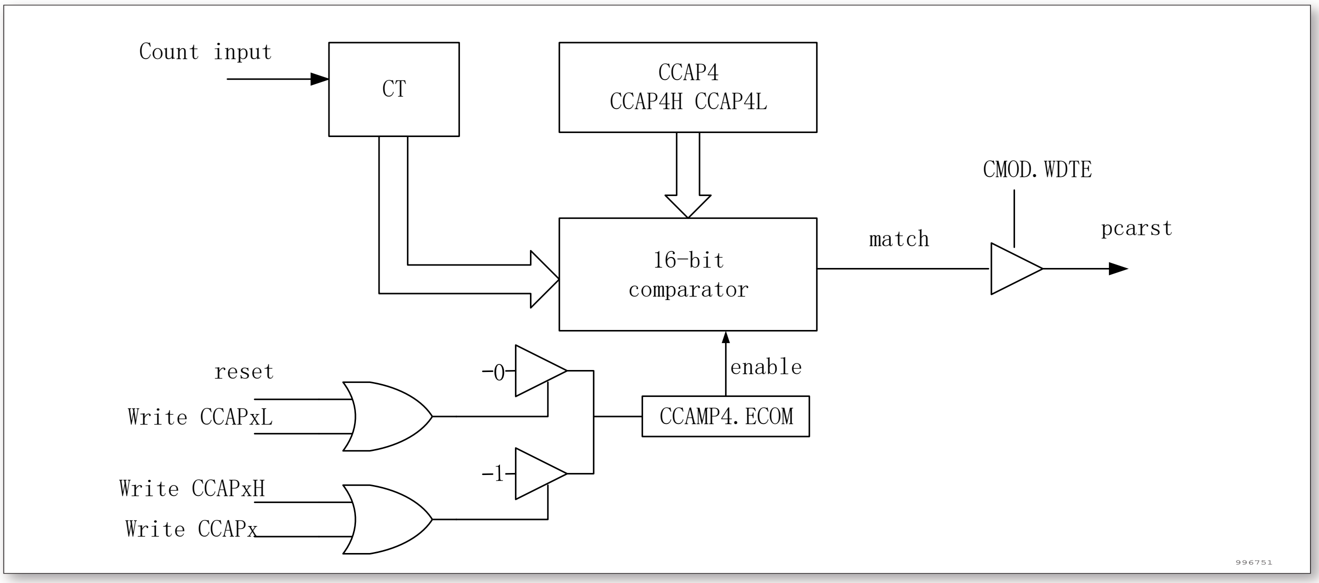

图 109. PCA WDT 功能框图

当把 PCA 模块 4 当 WDT 使用时，必须设置 CCAPM4.ECOM4，CCAPM4.MAT4 以及 CMOD.WDTE。另外 PCA 定时器/计数器可以设置 CMOD.CPS 来选择不同的输入计数频率。

在比较/捕获寄存器（CCAP4）输入一个 16 位的比较值。在 PCA 定时器/计数器（CNT），输入一个 16 位的初始值或使用复位值（0000h）。这些值乘以的 PCA 输入脉冲率之间的差额确定的 WDT 匹配运行时间。设置定时器/计数器运行控制位（CCON.CR）启动 PCA WDT。每次匹配时，PCA 的 WDT 产生复位信号。要防止一个 PCA WDT 复位，用户有三种选择。

- 定期的比较值 CCAP4 的改变，所以匹配永远不会发生。
- 定期更改 PCA 定时器/计数器值（CNT）所以匹配永远不会发生。
- 通过在匹配前清除的 CMOD.WDTE 位来禁用模块复位输出信号，后来重新启用它。

前两个选项是更可靠的，因为 WDT 在第三个选项没有被禁用。

第二个选项是不推荐，如果其他 PCA 模块都在使用，因为五个模块共享一个共同的时间基。因此，在大多数应用中的第一个选项是最好的。

PCA WDT 配置流程

1. 配置 WDT 比较/捕获寄存器 PCA_CCAP4；
2. 配置 PCA 计数寄存器 PCA_CNT；
3. 配置 PCA_CCAMP4 选择比较匹配功能；
4. 配置 PCA_CMOD 选择输入时钟，使能 WDT 功能；
5. 启动 PCA；
6. 选择清除 PCA WDT 清除方式在 PCA WDT 复位前清除 PCA WDT。

PCA 8 位 PWM 功能

脉宽调制是一种使用程序来控制波形占空比，周期，相位的技术。5 个 PCA 模块都可以被独立地用于在对应 PCA 的 CH[x] 引脚产生脉宽调制（PWM）输出，脉冲宽度为 8 位分辨率。PWM 输出的频率取决于 PCA 计数器/定时器的时基。使用模块的捕捉/比较寄存器 CCAPxL 来改变 PWM 输出信号的占空比。当 PCA 计数器/定时器的低字节（CL）与 CCAPxL 中的值相等时，PCA 的 CH[x] 引脚上的输出被置“1”；当 CL 中的计数值溢出时，PCA 的 CH[x] 输出被复位“0”。当计数器/定时器的低字节 CL 溢出时（从 0xFF 到 0x00），保存在 CCAPxH 中的值被自动装入到 CCAPxL，不需软件干预。

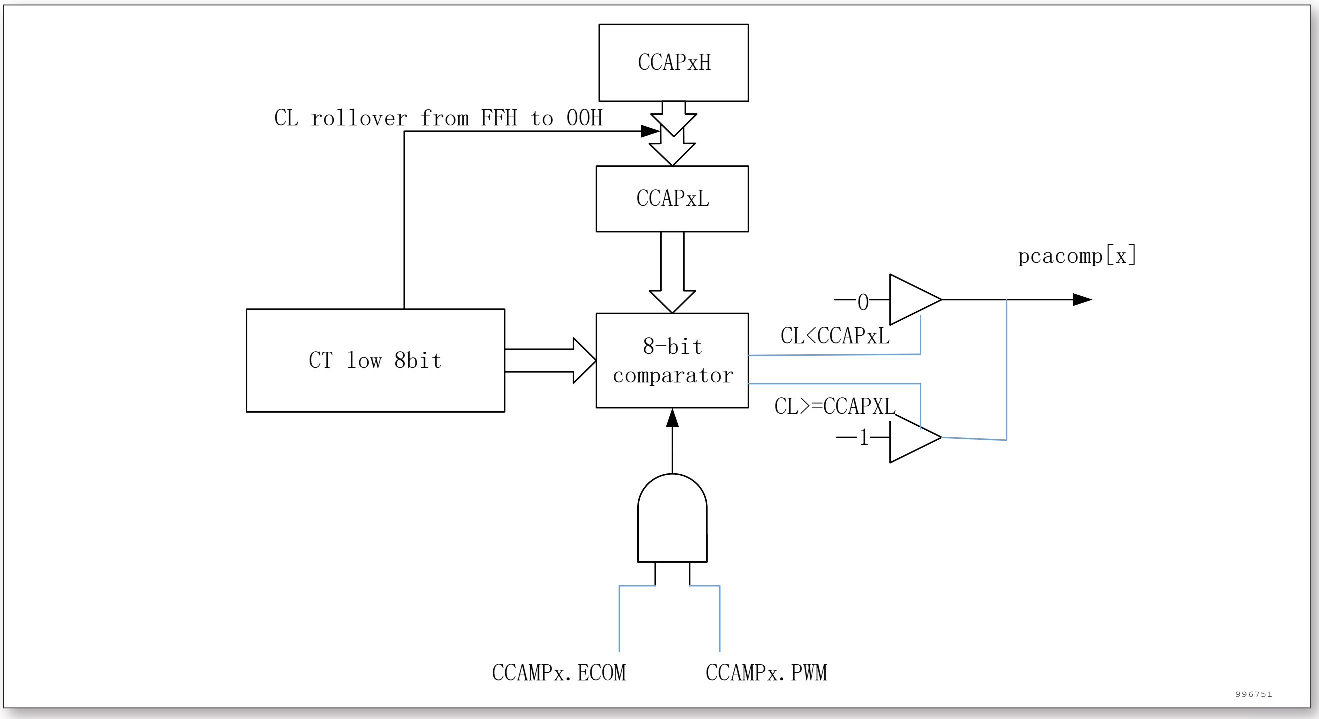

图 110. PCA PWM 功能框图

在这种模式下，PCA 定时器/计数器（CL）的低字节中的值是不断在低字节比较/捕获寄存器（CCAPxL）的值相比。当 CL < CCAPxL，输出波形为低。当两者匹配时（CL= CCAPxL），输出波形去到高，直到 CL 溢出从 FFH 到 00H，结束期间仍然很高。在溢出时，在 CCAPxH 的值自动装载到 CCAPxL 内，一个新的周期的开始。

在 CCAPxL 的值决定当前波形的占空比。在 CCAPxH 的值确定下一个波形的占空比。改变 CCAPxL 中的值即可更改的脉冲宽度调制。正如图所示，8 位值在 CCAPxL 可以从 0（100％占空比），到 255（0.4％占空比）。要改变 CCAPxL 值而不会产生毛刺，需要在高字节寄存器（CCAPxH）写入一个新值。当 CL 超过 FFH 滚动到 00h，这个值是由硬件自动加载到 CCAPxL。

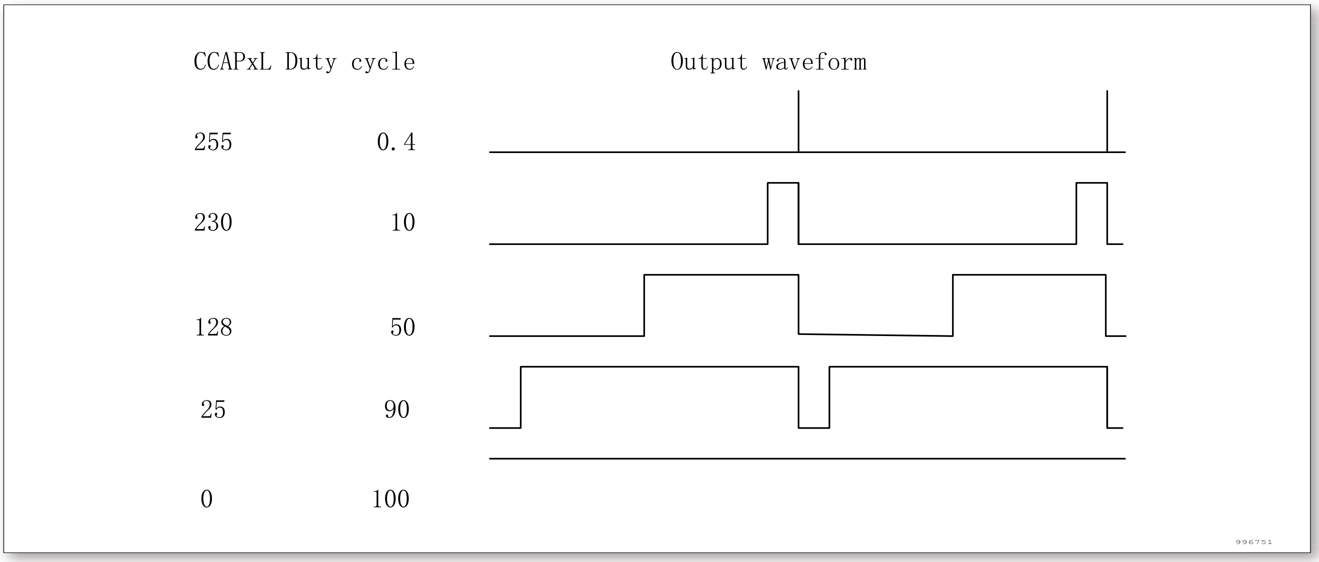

图 111. PCA PWM 输出波形

要设定一个比较 / 捕获模块在 PWM 模式下，需要设置 CCAPMx.ECOM 和 CCAPMx.PWM 位。另外 PCA 定时器/计数器由编程 CMOD.CSP[2:0] 可以选择输入计数信号频率。在 CCAPxL 输入一个 8 位的值指定第一个 PWM 波形的占空比。在 CCAPxH 输入一个 8 位的值会指定第二个 PWM 波形的占空比。设置定时器/计数器运行控制位（CCON.CR）启动 PCA 定时器/计数器。

表 65: CCAPMx 配置

ECOM	CAPP	CAPN	MAT	TOG	PWM	ECCF	EPWM	工作方式
X	1	0	0	0	0	X	0	用正沿触发捕获
X	0	1	0	0	0	X	0	用负沿触发捕获
X	1	1	0	0	0	X	0	用跳变沿触发捕获
1	0	0	1	0	0	X	0	软件定时器
1	0	0	1	1	0	X	0	高速输出
1	0	0	0	0	1	X	0	8 位脉冲宽度调制器
1	0	0	1	1	0	X	1	16 位脉冲宽度调制器

17.4 PCA 模块与其他模块互连及控制

PCA 通过端口功能选择可以连通到其他模块或端口。

表 66: PCA 端口功能选择

		0	1	2	3	4	5	6	7
GPIO_PCAS[2:0]	PCA_ECI	PX_SEL	PCNT1	LVD	VC0	VC1	PA05	PB02	PD02
GPIO_PCAS[5:3]	PCA_CH0	PX_SEL	PCNT0	PCNT1	LVD	VC1	PA06	PB04	PC06

当 GPIO_PCAS[2:0]=0x0 时，PCA_ECI 输入为 PX_SEL 选择的端口输入，当 GPIO_PCAS[2:0]=0x1~0X7 时，连接其他模块的输入或输出。

当 GPIO_PCAS[5:3]=0x0 时，PCA_CH0 捕获输入为 PX_SEL 选择的端口输入，当 GPIO_PCAS[5:3]=0x1~0X7

时，连接其他模块的输入或输出。

17.5 PCA 寄存器描述

表 67. PCA 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0X00	PCA_CCON	PCA 控制寄存器	0x0000 0000	小节 17.5.1
0X04	PCA_CMOD	PCA 模式寄存器	0x0000 0000	小节 17.5.2
0X08	PCA_CNT	PCA 计数寄存器	0x0000 0000	小节 17.5.3
0X0C	PCA_ICLR	PCA 中断清除寄存器	0x0000 009F	小节 17.5.4
0x10	PCA_CCAPM0	PCA 比较/捕获模块 0 模式寄存器	0x0000 0000	小节 17.5.5
0x14	PCA_CCAPM1	PCA 比较/捕获模块 1 模式寄存器	0x0000 0000	小节 17.5.5
0x18	PCA_CCAPM2	PCA 比较/捕获模块 2 模式寄存器	0x0000 0000	小节 17.5.5
0x1C	PCA_CCAPM3	PCA 比较/捕获模块 3 模式寄存器	0x0000 0000	小节 17.5.5
0x20	PCA_CCAPM4	PCA 比较/捕获模块 4 模式寄存器	0x0000 0000	小节 17.5.5
0X24	PCA_CCAP0H	PCA 比较/捕获模块 0 高 8 位寄存器	0x0000 0000	小节 17.5.6
0X28	PCA_CCAP0L	PCA 比较/捕获模块 0 低 8 位寄存器	0x0000 0000	小节 17.5.7
0X2C	PCA_CCAP1H	PCA 比较/捕获模块 1 高 8 位寄存器	0x0000 0000	小节 17.5.6
0X30	PCA_CCAP1L	PCA 比较/捕获模块 1 低 8 位寄存器	0x0000 0000	小节 17.5.7
0X34	PCA_CCAP2H	PCA 比较/捕获模块 2 高 8 位寄存器	0x0000 0000	小节 17.5.6
0X38	PCA_CCAP2L	PCA 比较/捕获模块 2 低 8 位寄存器	0x0000 0000	小节 17.5.7
0X3C	PCA_CCAP3H	PCA 比较/捕获模块 3 高 8 位寄存器	0x0000 0000	小节 17.5.6
0X40	PCA_CCAP3L	PCA 比较/捕获模块 3 低 8 位寄存器	0x0000 0000	小节 17.5.7
0X44	PCA_CCAP4H	PCA 比较/捕获模块 4 高 8 位寄存器	0x0000 0000	小节 17.5.6
0X48	PCA_CCAP4L	PCA 比较/捕获模块 4 低 8 位寄存器	0x0000 0000	小节 17.5.7
0X4C	PCA_CCAPO	PCA PWM 与高速输出标志寄存器	0x0000 0000	小节 17.5.8
0X50	PCA_CCAP0	PCA 比较/捕获模块 0 的 16 位寄存器	0x0000 0000	小节 17.5.9
0X54	PCA_CCAP1	PCA 比较/捕获模块 1 的 16 位寄存器	0x0000 0000	小节 17.5.9
0X58	PCA_CCAP2	PCA 比较/捕获模块 2 的 16 位寄存器	0x0000 0000	小节 17.5.9
0X5C	PCA_CCAP3	PCA 比较/捕获模块 3 的 16 位寄存器	0x0000 0000	小节 17.5.9
0X60	PCA_CCAP4	PCA 比较/捕获模块 4 的 16 位寄存器	0x0000 0000	小节 17.5.9
0X64	PCA_CARR	PCA 周期装载寄存器	0x0000 0000	小节 17.5.10
0X68	PCA_EPWM	PCA PWM 增强寄存器	0x0000 0000	小节 17.5.11

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

17.5.1 PCA 控制寄存器 (PCA_CCON)

起始地址：0X40001000

地址偏移：0X00

复位值：0x00000000

31-8	7	6	5	4	3	2	1	0
Reserved	CF	CR	Reserved	CCF4	CCF3	CCF2	CCF1	CCF0
	RO	RW		RO	RO	RO	RO	RO

Bit	Field	Type	Reset	Description
31:8	Reserved			保留。
7	CF	r	0x00	PCA 计数器溢出标志（写无效） 当 PCA 计数溢出时，CF 由硬件置位，如果 CMOD 寄存器的 CFIE 位为 1，则 CF 标志可以产生中断 1：发生计数器溢出；0：无溢出；
6	CR	rw	0x00	PCA 计数器运行控制位 1：启动 PCA 计数器计数 0：关闭 PCA 计数器计数
5	Reserved			保留。
4	CCF4	r	0x00	PCA 计数器模块 4 比较/捕获标志位。 当出现匹配或捕获时，该位由硬件置位。（写无效） 当 CCAPM4.CCIE 置位时，这个标志位会产生一个 PCA 中断
3	CCF3	r	0x00	PCA 计数器模块 3 比较/捕获标志位。 当出现匹配或捕获时，该位由硬件置位。（写无效） 当 CCAPM3.CCIE 置位时，这个标志位会产生一个 PCA 中断
2	CCF2	r	0x00	PCA 计数器模块 2 比较/捕获标志位。 当出现匹配或捕获时，该位由硬件置位。（写无效） 当 CCAPM2.CCIE 置位时，这个标志位会产生一个 PCA 中断
1	CCF1	r	0x00	PCA 计数器模块 1 比较/捕获标志位。 当出现匹配或捕获时，该位由硬件置位。（写无效） 当 CCAPM1.CCIE 置位时，这个标志位会产生一个 PCA 中断
0	CCF0	r	0x00	PCA 计数器模块 0 比较/捕获标志位。 当出现匹配或捕获时，该位由硬件置位。（写无效） 当 CCAPM0.CCIE 置位时，这个标志位会产生一个 PCA 中断

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

17.5.2　PCA 模式寄存器 (PCA_CMOD)

起始地址：0X40001000

地址偏移：0X04

复位值：0x00000000

31-8	7	6	5	4	3	2	1	0
Reserved	CIDL	WDTE	Reserved		CPS			CFIE
	RW	RW			RW			RW

Bit	Field	Type	Reset	Description
31:8	Reserved			保留。
7	CIDL	rw	0x00	空闲模式 IDLE 下，PCA 是否停止工作。 1：休眠模式（sleep）下，PCA 停止工作 0：休眠模式（sleep）下，PCA 继续工作
6	WDTE	rw	0x00	PCA WDT 功能使能控制位。 1：启动 PCA 模块 4 WDT 功能 0：关闭 PCA 模块 4 WDT 功能
5:4	Reserved			保留。
3:1	CPS[2:0]	rw	0x00	时钟分频选择及时钟源选择。 000：PCLK/32 001：PCLK/16 010：PCLK/8 011：PCLK/4 100：PCLK/2 101：timer0 overflow 110：timer1 overflow 111：ECI 外部时钟，时钟 PCLK 四分频采样
0	CFIE	rw	0x00	PCA 计数器中断使能控制信号。 1：使能中断 0：关闭中断

17.5.3　PCA 计数寄存器 (PCA_CNT)

起始地址：0X40001000

地址偏移：0X08

复位值：0x00000000

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。
15:0	CNT	rw	0x00	定时器计数器的值。 只有在 PCA 停止状态 CNT 才可以写入。否则写入无效

17.5.4 PCA 中断清除寄存器 (PCA_ICLR)

起始地址：0X40001000

地址偏移：0X0C

复位值：0x0000009F

31-8	7	6	5	4	3	2	1	0
Reserved	CF	Reserved		CCF4	CCF3	CCF2	CCF1	CCF0
	R1W0			R1W0	R1W0	R1W0	R1W0	R1W0

Bit	Field	Type	Reset	Description
31:8	Reserved			保留。
7	CF	rc_w0	0x01	PCA 计数器溢出标志清除（软件写 0 清零，写 1 无效），读出值为 1
6:5	Reserved			保留。
4	CCF4	rc_w0	0x01	PCA 计数器模块 4 比较/捕获标志位清除（软件写 0 清零，写 1 无效），读出值为 1。
3	CCF3	rc_w0	0x01	PCA 计数器模块 3 比较/捕获标志位清除（软件写 0 清零，写 1 无效），读出值为 1。
2	CCF2	rc_w0	0x01	PCA 计数器模块 2 比较/捕获标志位清除（软件写 0 清零，写 1 无效），读出值为 1。
1	CCF1	rc_w0	0x01	PCA 计数器模块 1 比较/捕获标志位清除（软件写 0 清零，写 1 无效），读出值为 1。
0	CCF0	rc_w0	0x01	PCA 计数器模块 0 比较/捕获标志位清除（软件写 0 清零，写 1 无效），读出值为 1。

17.5.5 比较捕获模式寄存器 (PCA_CCAPM0~4)

起始地址：0X40001000

地址偏移：CCAPM0：0x010　CCAPM1：0x014　CCAPM2：0x018　CCAPM3：0x01C　CCAPM4：0x020

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

复位值：0x00000000

31-8	7	6	5	4	3	2	1	0
Reserved		ECOM	CAPP	CAPN	MAT	TOG	PWM	CCIE
		RW	RW	RW	RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31:7	Reserved			保留。
6	ECOM	rw	0x00	允许比较器功能控制位 1：允许比较器功能；0：禁止比较强功能； 当 PCA 用于软件计数器，高速输出，PWM 模式，WDT 模式，要置位 ECOM。 写 CCAMPHx 或 CCAMPx 寄存器会自动置位 ECOM；写 CCAMPLx 寄存器会自动清除 ECOM 位。
5	CAPP	rw	0x00	正沿捕获控制位。 1：允许上升沿捕获；0：禁止上升沿捕获
4	CAPN	rw	0x00	负沿捕获控制位。 1：允许下降沿捕获；0：禁止下降沿捕获
3	MAT	rw	0x00	允许匹配控制位。 1：PCA 计数值与模块的比较/捕获寄存器的值一旦匹配，将置位 CCON 寄存的中断标志 CCFx(x=0-4) 0：禁止匹配功能
2	TOG	rw	0x00	翻转控制位。 1：工作在 PCA 高速输出模式，PCA 计数器的值与模块的比较/捕获寄存器的值一旦匹配，CCPx 引脚翻转 0：禁止翻转功能
1	PWM	rw	0x00	脉宽调制控制位。 1：允许 CCPx 引脚作为 PWM 输出 0：禁止 PWM 脉宽调制功能 只有 CCAPMx[6:0]=100_0010 时，PWM 功能才有效。
0	CCIE	rw	0x00	PCA 使能中断。 1：使能比较/捕获中断 0：PCA 比较/捕获功能中断禁止

17.5.6 比较捕获数据寄存器高 8 位 (PCA_CCAP0~4H)

起始地址：0X40001000

地址偏移：CCAP0H：0x024　CCAP1H：0x02C　CCAP2H：0x034　CCAP3H：0x03C　CCAP4H：0x044

复位值：0x00000000

31:8	7	6	5	4	3	2	1	0
Reserved	CCAPx[15：8]							
	RW							

Bit	Field	Type	Reset	Description
31:8	Reserved			保留。
7:0	CCAPx[15:8]	rw	0x00	比较/捕获模式高 8 位寄存器。 当 PCA 模式用于比较/捕获模式时，用于保存 16 位捕获计数值的高 8 位；写 CCAPxH 寄存器会自动置位寄存器 CCAPMx 的 ECOM 位。 当 PCA 模式用于 PWM 模式时，用于控制输出占空比装载寄存器，在计数器低 8 位溢出时，装载寄存器会自动更新到 PWM 比较寄存器。

17.5.7 比较捕获数据寄存器低 8 位 (PCA_CCAP0~4L)

起始地址：0X40001000

地址偏移：CCAP0L：0x028　CCAP1L：0x030　CCAP2L：0x038　CCAP3L：0x040　CCAP4L：0x048

复位值：0x00000000

31:8	7	6	5	4	3	2	1	0
Reserved	CCAPx[7：0]							
	RW							

Bit	Field	Type	Reset	Description
31:8	Reserved			保留。
7:0	CCAPx[7:0]	rw	0x00	比较/捕获模式低 8 位寄存器 当 PCA 模式用于比较/捕获模式时，用于保存 16 位捕获计数值的低 8 位；写 CCAPxL 寄存器会自动清除寄存器 CCAPMx 的 ECOM 位。 当 PCA 模式用于 PWM 模式时，用于控制输出占空比比较寄存器，在 PWM 模式，计数器的低 8 位的值小于 CCAPx[7:0] 的值 PWM 输出低电平，否则 PWM 输出高电平。

17.5.8 比较高速输出标志寄存器 (PCA_CCAPO)

起始地址：0X40001000

地址偏移：0x4C

复位值：0x00000000

31:8	7	6	5	4	3	2	1	0
Reserved				CCAPO4	CCAPO3	CCAPO2	CCAPO1	CCAPO0
				RW	RW	RW	RW	RW

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31:5	Reserved			保留。
4	CCAPO4	rw	0x00	比较模块 4 的输出值。
3	CCAPO3	rw	0x00	比较模块 3 的输出值。
2	CCAPO2	rw	0x00	比较模块 2 的输出值。
1	CCAPO1	rw	0x00	比较模块 1 的输出值。
0	CCAPO0	rw	0x00	比较模块 0 的输出值。

17.5.9 比较捕获 16 位寄存器 (PCA_CCAP0~4)

起始地址：0X40001000

地址偏移：CCAP0：0x050 CCAP1：0x054 CCAP2：0x058 CCAP3：0x05C CCAP4：0x060

复位值：0x00000000

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。
15:0	CCAPx	rw	0x00	比较/捕获模式 16 位寄存器 当 PCA 模式用于比较/捕获模式时，用于保存 16 位捕获计数值；写 CCAPx 寄存器会置位寄存器 CCAPMx 的 ECOM 位。 写 CCAPX 寄存器相当于写 CCAPxL 及 CCAPxH 这两个 8 位寄存器。在比较/捕获模式下可以直接读写这个寄存器，在 PWM 模式下，使用 CCAPxL 及 CCAPxH 寄存器。

17.5.10 周期装载寄存器 (PCA_CARR)

起始地址：0X40001000

地址偏移：0X64

复位值：0x00000000

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留。
15:0	CARR	rw	0x00	计数周期重载寄存器。

17.5.11 增强 PWM 控制 (PCA_EPWM)

起始地址：0X40001000

地址偏移：0X68

复位值：0x00000000

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

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

Bit	Field	Type	Reset	Description
31:1	Reserved			保留。
0	EPWM	rw	0x00	16 bit PWM 使能。

18 高级控制定时器 (TIM4/5/6)

18.1 Advanced Timer 简介

Advanced Timer 是一个包含三个定时器 Timer4/5/6。Timer4/5/6 功能相同的高性能计数器，可用于计数产生不同形式的时钟波形，1 个定时器可以产生互补的一对 PWM 或者独立的 2 路 PWM 输出，可以捕获外界输入进行脉冲宽度或周期测量。

Advanced Timer 基本的功能及特性如表所示。

表 68: Advanced Timer 基本特性

波形模式	锯齿波、三角波
基本功能	递加、递减计数方向
	软件同步
	硬件同步
	缓存功能
	正交编码计数
	通用 PWM 输出
	保护机制
	AOS 关联动作
中断类型	计数比较匹配中断
	计数周期匹配中断
	死区时间错误中断

表 69: Advanced Timer 端口列表

端口名	方向	功能
TIMx_CHA	输入/输出	正交编码计数时钟输入端口或 捕获输入端口或比较输出端口（x=4~6）
TIMx_CHB		
TIMTRIA	输入	硬件计数时钟输入端口或捕获输入端口 硬件启动、停止、清零条件输入端口
TIMTRIB		
TIMTRIC		
TIMTRID		

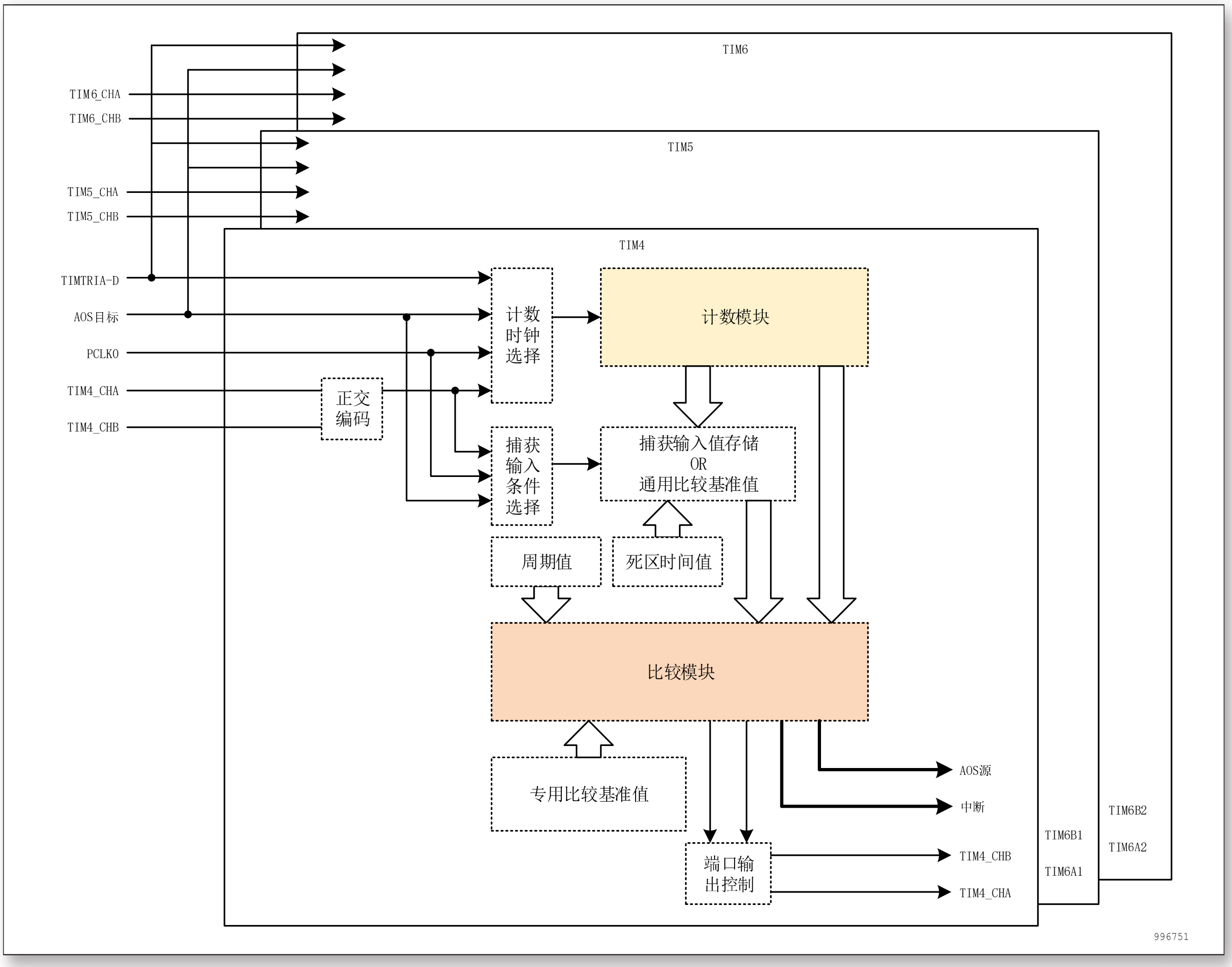

图 112. Advanced Timer 框图

18.2 Advanced Timer 功能描述

18.2.1 基本动作

基本波形模式

Timer4/5/6 有 2 种基本计数波形模式，锯齿波模式和三角波模式。波形模式又由于不同的内部计数动作有所细分，三角波模式分为三角波 A 模式、三角波 B 模式。锯齿波和三角波的基本波形如图 18-2 图 18-3 所示。三角波 A 模式与三角波 B 模式区别在于缓存传送有差别，三角波 A 模式一个周期只发生一次缓存传送（谷点），而三角波 B 模式一个周期发生两次缓存传送（峰点和谷点）。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

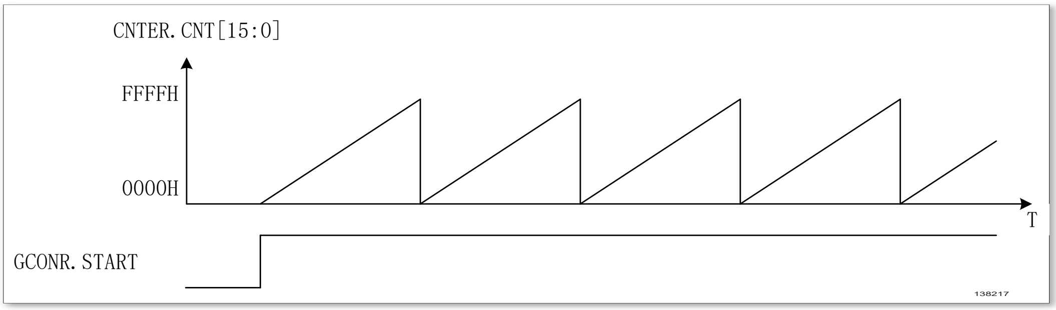

图 113. 锯齿波波形 (递加计数)

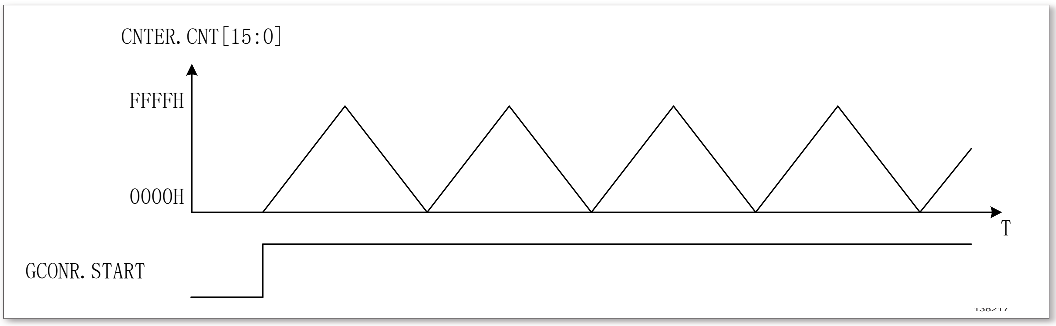

图 114. 三角波波形

比较输出

Timer4/5/6 一个定时器有 2 个比较输出端口（CHxA、CHxB），可在计数值与计数基准值比较匹配时输出指定的电平。GCMAR、GCMBR 寄存器分别对应了 CHxA、CHxB 的计数比较基准值。当计数器的计数值和 GCMAR 相等时，CHxA 端口输出指定的电平；当计数器的计数值和 GCMBR 相等时，CHxB 端口输出指定电平。

CHxA、CHxB 端口的计数起始电平、停止电平、计数比较匹配时的电平等，可由端口控制寄存器（PCONR）的 PCONR.STACA、PCONR.STPCA、PCONR.STASTPSA、PCONR.CMPCA[1:0]、PCONR.PERCA[1:0] 和 PCONR.STACB、PCONR.STPCB、PCONR.STASTPSB、PCONR.CMPCB[1:0]、PCONR.PERCB[1:0] 位设定。图 18-4 为比较输出的动作例。

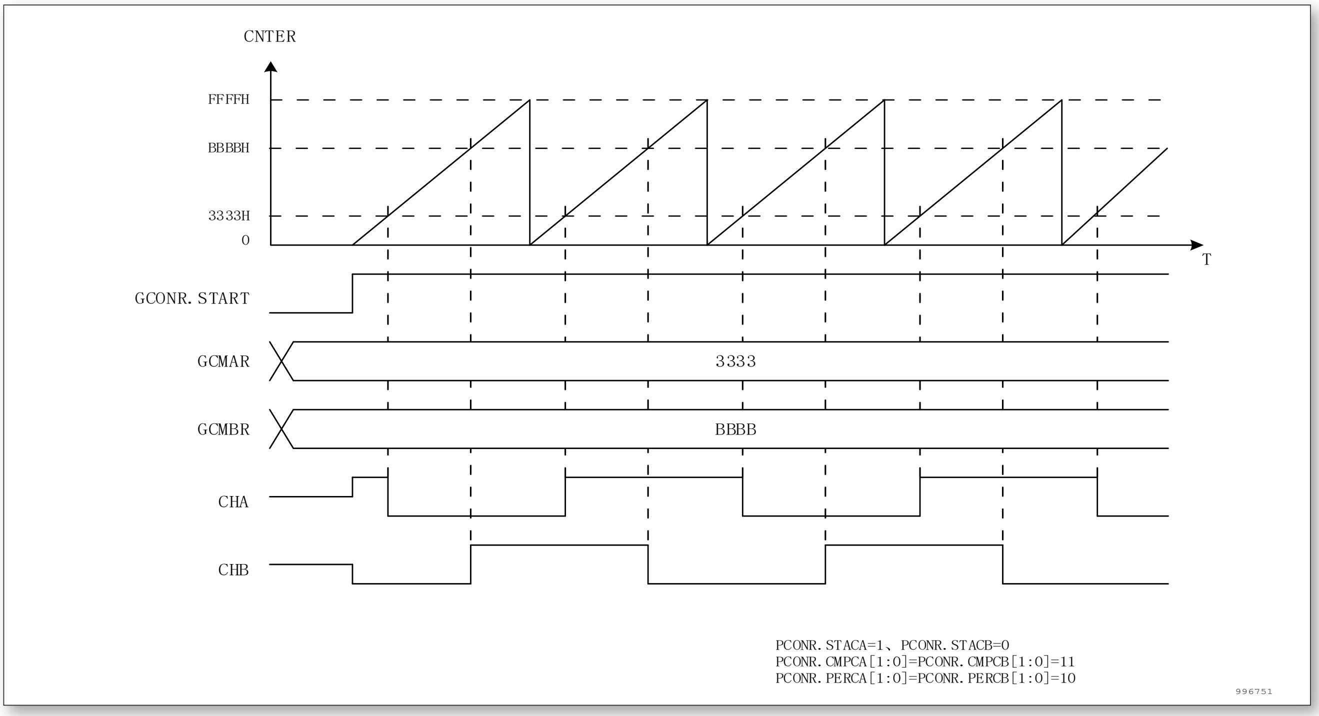

图 115. 比较输出动作

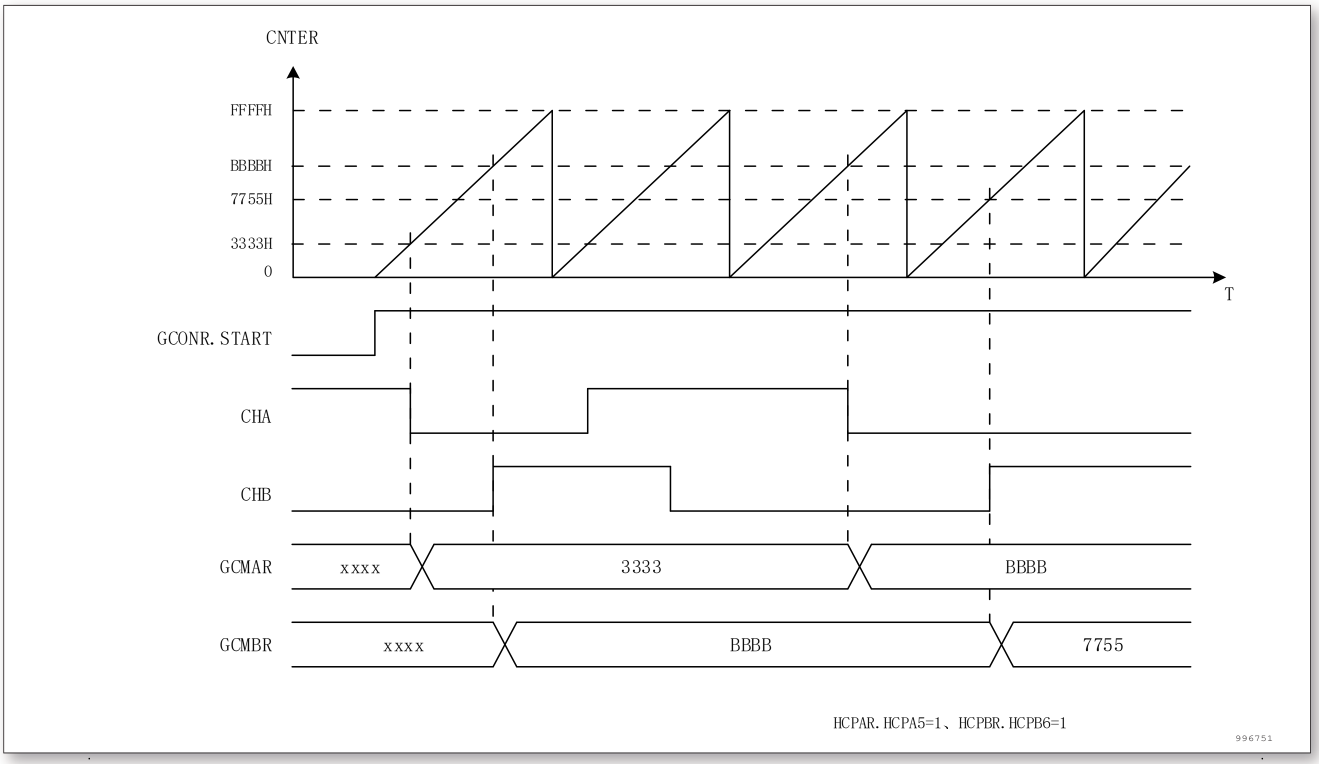

图 116. 捕获输入动作

Timer4/5/6 都具有捕获输入功能，具备 2 组捕获输入寄存器（GCMAR、GCMBR），用于保存捕获到的计数值。设定端口控制寄存器（PCONR）的 PCONR.CAPCA、PCONR.CAPCB 位为 1，捕获输入功能就有效了。当设定了对应的捕获输入条件且该条件有效时，当前的计数值就被保存到相应的寄存器（GCMAR、GCMBR）中。

每组捕获输入的条件可以是 AOS 事件触发、TIMTRIA-TIMTRID 输入、CHxA 或 CHxB 的输入等，具体的条

件选择可通过硬件捕获事件选择寄存器（HCPAR、HCPBR）来设定。图 116为捕获输入的动作例。

18.2.2 时钟源选择

Timer4/5/6 的计数时钟可以有以下几种选择：

1. PCLK 及 PCLK 的 2、4、8、16、64、256、1024 分频 (GCONR.CKDIV[2:0] 设定)。
2. AOS 事件触发输入（HCUPR.HCUP [19:16] 或 HCDOR.HCDO [19:16] 设定）。
3. CHxA 和 CHxB 的正交编码输入（HCUPR.HCUP[7:0] 或 HCDOR.HCDO [7:0] 设定）。
4. TIMTRIA-TIMTRID 的端口输入（HCUPR.HCUP [15:8] 或 HCDOR.HCDO [15:8] 设定）。

从上述描述可以看到，2、3、4 时钟互相独立，可分别设定有效或无效，并且当选择 2、3、4 时钟时，1 时钟自动无效

18.2.3 计数方向

Timer4/5/6 的计数器计数方向可通过软件方式改变。不同波形模式时，改变计数方向的方法略有不同。

锯齿波计数方向

锯齿波模式时，计数方向可在计数器计数中或停止时设定。

在递加计数中时，设定 GCONR.DIR=0（递减计数），则计数器计数到上溢后变为递减计数模式；在递减计数中时，设定 GCONR.DIR=1（递加计数），则计数器计数到下溢后变为递加计数模式。

在计数停止时，设定 GCONR.DIR 位。则计数开始后直至上溢或下溢时，GCONR.DIR 的设定才会反映到计数中

三角波计数方向

三角波模式时，计数方向只能在计数器停止时设定。在计数中设定计数方向无效。在计数停止时，设定 GCONR.DIR 位。则计数开始后直至上溢或下溢时，GCONR.DIR 的设定才会反映到计数中

18.2.4 数字滤波

Timer4/5/6 的 CHxA、CHxB、TIMTRIA~D 端口输入都有数字滤波功能。可通过设定滤波控制寄存器（FCONR）的相关使能位开启对应端口的滤波功能。滤波用的基准时钟也通过滤波控制寄存器（FCONR）设定。

在滤波采样基准时钟采样到端口上 3 次一致的电平时，该电平被当作有效电平传送到模块内部；小于 3 次一致的电平会被当作外部干扰滤掉，不传送到模块内部。其动作例如图 117 所示。

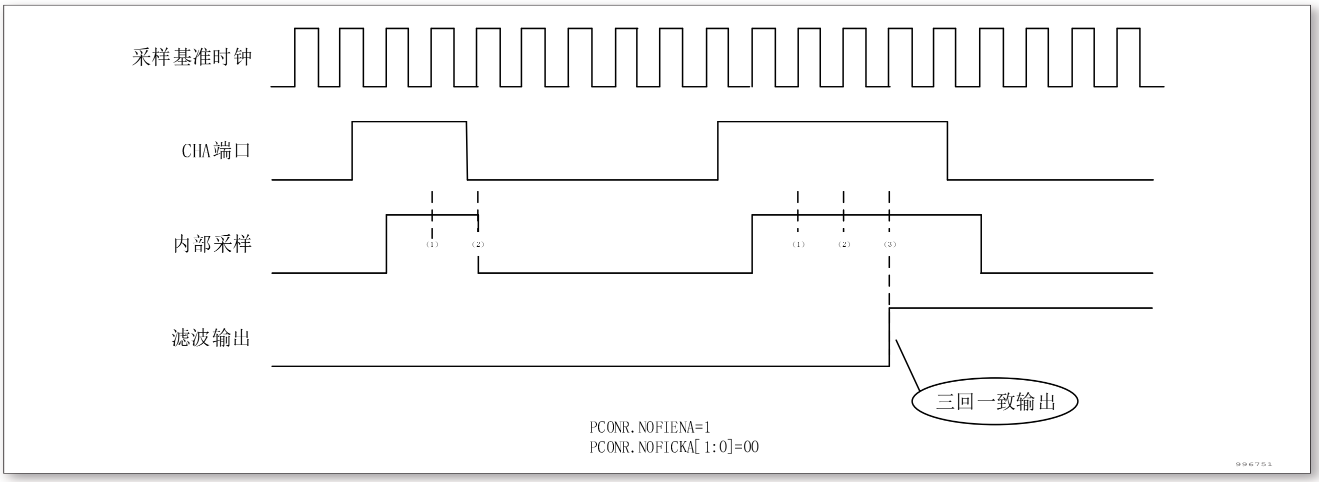

图 117. 捕获输入端口的滤波功能

TIMTRIA~D 端口是一组 Timer4/5/6 间共用的端口，该组端口的数字滤波功能在只在 Timer4 实现，其他定时器 Timer5/6 对该组端口的数字滤波功能设定无效。

18.2.5 软件同步

软件同步启动

Timer4/5/6 可通过设定软件同步启动寄存器（SSTAR）的相关位，实现目标 Timer4/5/6 的同步启动。

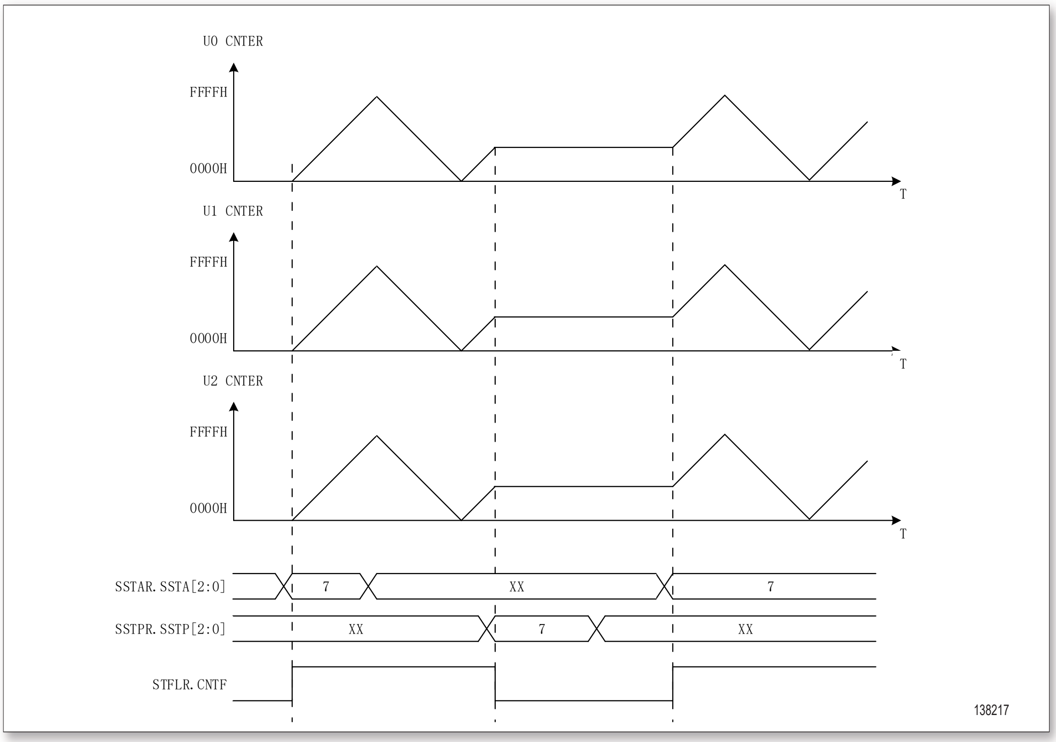

图 118. 软件同步动作

软件同步停止

Timer4/5/6 可通过设定软件同步停止寄存器（SSTPR）的相关位，实现目标 Timer4/5/6 的同步停止。

软件同步清零

Timer4/5/6 可通过设定软件同步清零寄存器（SCLRR）的相关位，实现目标 Timer4/5/6 的同步清零。

如图 18-7 所示、若设定 Timer4 的 SSTAR.SSTA0=SSTAR.SSTA1=SSTAR.SSTA2, 即可实现 Timer4/5/6 的软件同步启动。

软件同步动作相关寄存器（SSTAR、SSTPR、SCLRR）是一组独立于 Timer4/5/6 外、各个 TIM 间共用的寄存器，这组寄存器的各个位只在写 1 时有效，写 0 无效。在读取 SSTAR 寄存器时，会读出各个定时器的计数器状态，在读取 SSTPR 或 SCLRR 时，会读出 0。

18.2.6 硬件同步

每个定时器除独立拥有 2 个通用输入端口（CHxA、CHxB）外，还共同拥有 4 个外部通用输入端口（TIMTRIA、TIMTRIB、TIMTRIC、TIMTRID）及 4 个 AOS 目标，可实现定时器间的硬件同步动作。

硬件同步启动

各 Timer4/5/6 均可选择用硬件方式启动计数器，选择相同硬件启动条件的定时器即可在启动条件有效时实现

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

同步启动。具体的硬件启动条件由硬件启动事件选择寄存器（HSTAR）的设定来决定。

硬件同步停止

各 Timer4/5/6 均可选择用硬件方式停止计数器，选择相同硬件停止条件的定时器即可在停止条件有效时实现同步停止。具体的硬件停止条件由硬件停止事件选择寄存器（HSTPR）的设定来决定。

硬件同步清零

各 Timer4/5/6 均可选择用硬件方式清零计数器，选择相同硬件清零条件的定时器即可在清零条件有效时实现同步清零。具体的硬件清零条件由硬件清零事件选择寄存器（HCLRR）的设定来决定。

硬件同步捕获输入

各 Timer4/5/6 均可选择用硬件方式实现捕获输入功能，选择相同捕获输入功能条件的定时器即可在捕获输入功能条件有效时实现同步捕获输入。具体的硬件捕获输入功能条件由硬件捕获事件选择寄存器（HCPAR、HCPBR）的设定来决定。

硬件同步计数

Timer4/5/6 均可选择用硬件输入作为 CLOCK 进行计数，选择相同硬件计数条件的定时器即可在硬件计数 CLOCK 有效时实现同步计数。具体的硬件计数条件由硬件递加事件选择寄存器（HCUPR）和硬件递减事件选择寄存器（HCDOR）的设定来决定。选择硬件同步计数功能时，只是选择了外部输入时钟源，不影响计数器的启动、停止、清零动作。计数器的启动、停止、清零等还需要单独设定。

图 119 所示、Timer4/5/6 的硬件同步动作例。

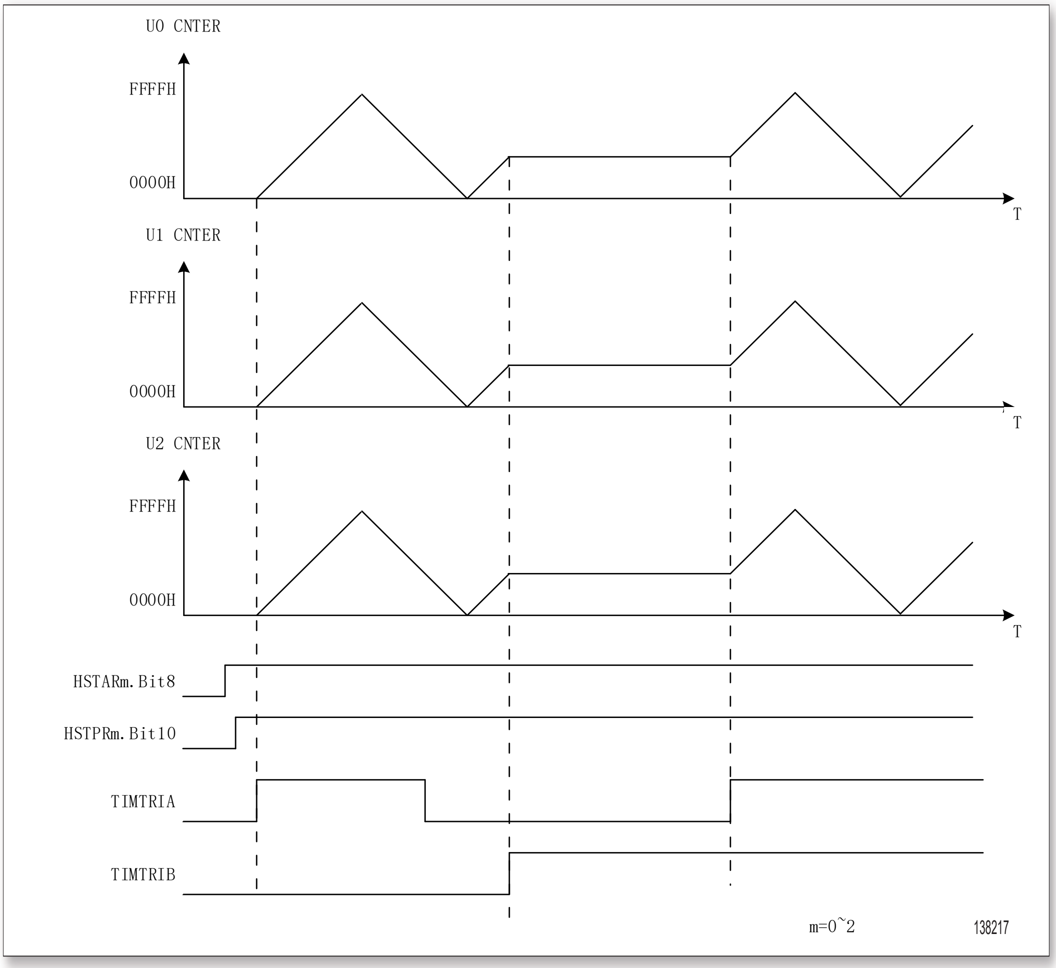

图 119. 硬件同步动作

18.2.7 缓存功能

缓存动作是指通过设定缓存控制寄存器（BCONR），在缓存传送时间点，选择发生以下事件：

- 通用周期基准值缓存寄存器（PERBR）的值自动传送到通用周期基准值寄存器（PERAR）中。
- 通用比较基准值缓存寄存器（GCMCR、GCMDR）的值自动传送到通用比较基准值寄存器（GCMAR、GCMBR）中（比较输出时）。
- 通用比较基准值寄存器（GCMAR、GCMBR）的值自动传送到通用比较基准值缓存寄存器（GCMCR、GCMDR）中（捕获输入时）。

如图 120 所示，是比较输出动作时、通用比较基准值寄存器的单缓存方式的时序图。从图中可以看到，在计数期间改变通用比较基准值寄存器（GCMAR）的值可以调整输出占空比，改变通用周期基准值寄存器（PERAR）的值可以调整输出周期。

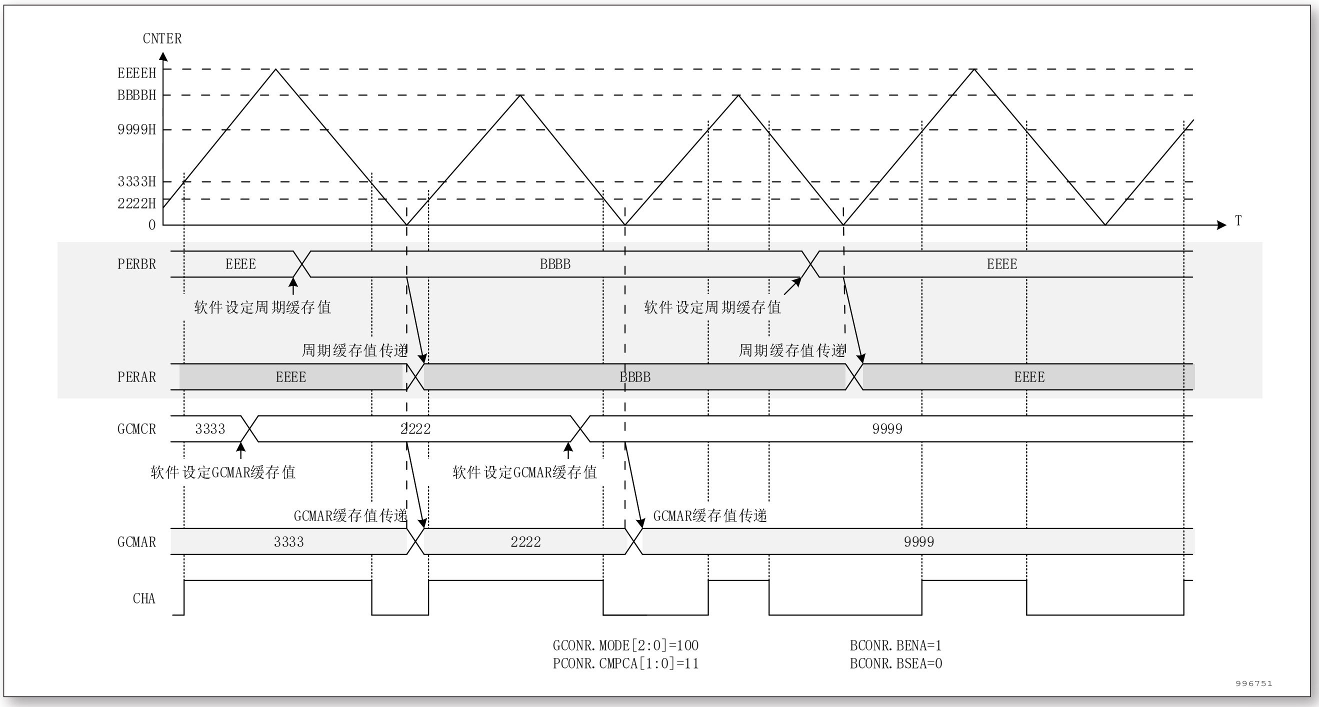

图 120. 单缓存方式比较输出时序

缓存传送时间点

通用周期基准值缓存传送时间点

周期基准值缓存传送时间点为锯齿波时递加计数上溢点或递减计数下溢点、三角波时计数谷点。

通用比较基准值缓存传送时间点

锯齿波 A 模式时，设定 BCONR.BENA=1 或 BCONR.BNEB=1，缓存动作有效。缓存传送发生在上溢点或下溢点。

三角波 A 模式时，设定 BCONR.BENA=1 或 BCONR.BNEB=1，缓存动作有效。缓存传送发生在计数谷点。

三角波 B 模式时，设定 BCONR.BENA=1 或 BCONR.BNEB=1，缓存动作有效。缓存传送发生在计数谷点和计数峰点。

捕获输入值缓存传送时间点

捕获输入动作缓存传送时间点为捕获输入动作时。

清零动作时缓存传送

在锯齿波计数模式或硬件计数模式时，正常的比较输出动作期间若有清零动作产生，通用周期基准值、通用比较基准值、等会根据相应的缓存动作设定状况发生一次缓存传送。

18.2.8 通用 PWM 输出

PWM 展频输出

为了降低 PWM 输出对外部的干扰，在 PWM 输出级有展频配置。每个 PWM 输出周期会微调 PWM 输出的相位。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

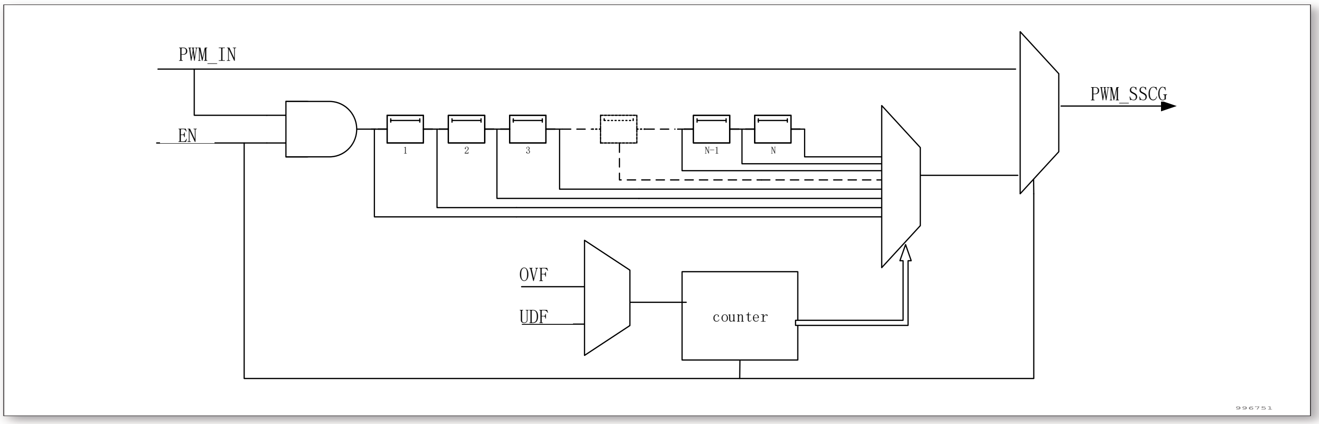

图 121. PWM 展频输出示意图

独立 PWM 输出

每个定时器的 2 个端口 CHxA、CHxB 能独立的输出 PWM 波。如 图 122，定时器 Timer6 的 CHA 端口输出 PWM 波。

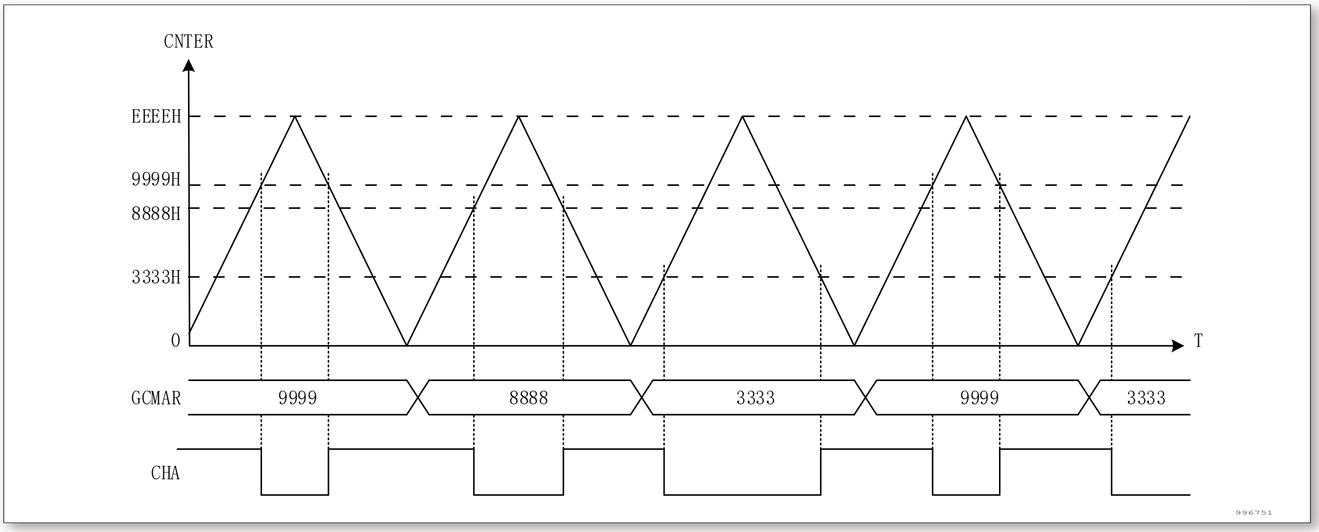

图 122. CHA 输出 PWM 波

互补 PWM 输出

CHxA 端口和 CHxB 端口，在不同的模式下可组合输出互补 PWM 波形。

- 软件设定 GCMBR 互补 PWM 输出；

软件设定 GCMBR 互补 PWM 输出是指在锯齿波模式和三角波 A 模式、三角波 B 模式下，用于 CHxB 端口波形输出的通用比较基准值寄存器（GCMBR）的值由寄存器直接设定，与通用比较基准值寄存器（GCMAR）的值没有直接关系。

图 123 为软件设定 GCMBR 互补 PWM 波的输出例。

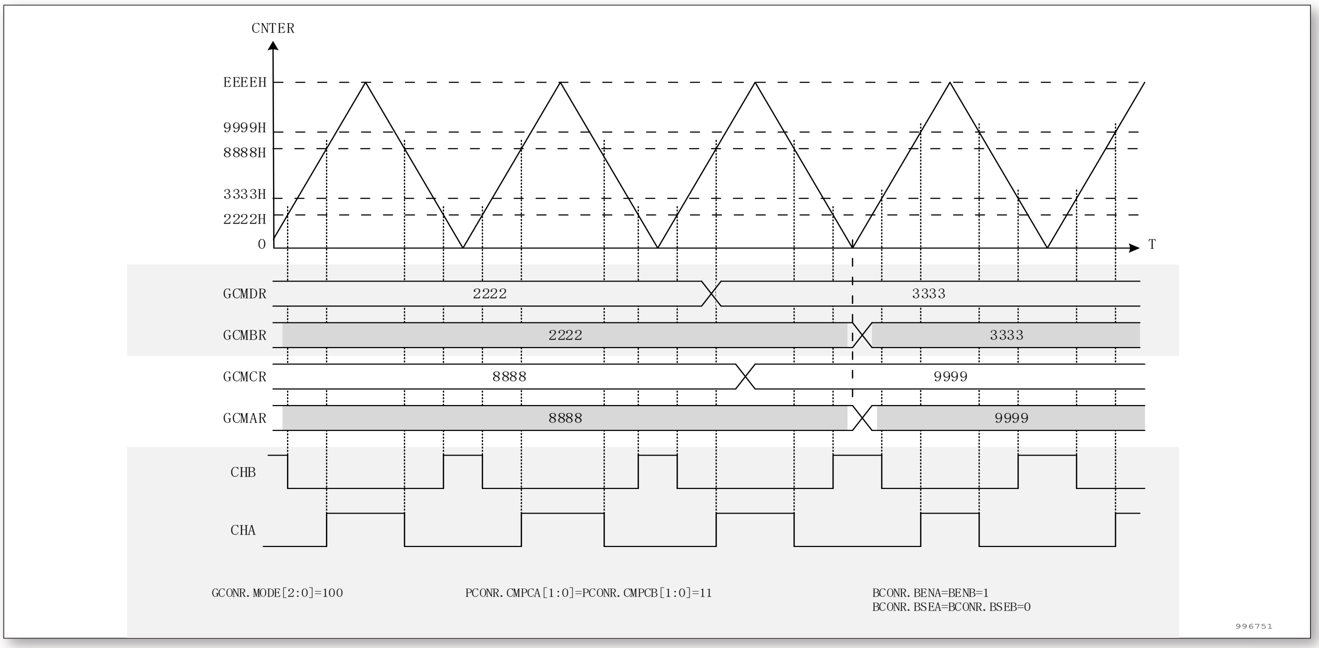

图 123. 三角波 A 模式时软件设定 GCMBR 互补 PWM 波输出

- 硬件设定 GCMBR 互补 PWM 输出；

硬件设定 GCMBR 互补 PWM 输出是指在三角波 A 模式、三角波 B 模式下，用于 CHxB 端口波形输出的通用比较基准值寄存器（GCMBR）的值由通用比较基准值寄存器（GCMAR）和死区时间基准值寄存器（DTUAR、DTDAR）的值运算决定。

图 124 为硬件设定 GCMBR 互补 PWM 波输出例。

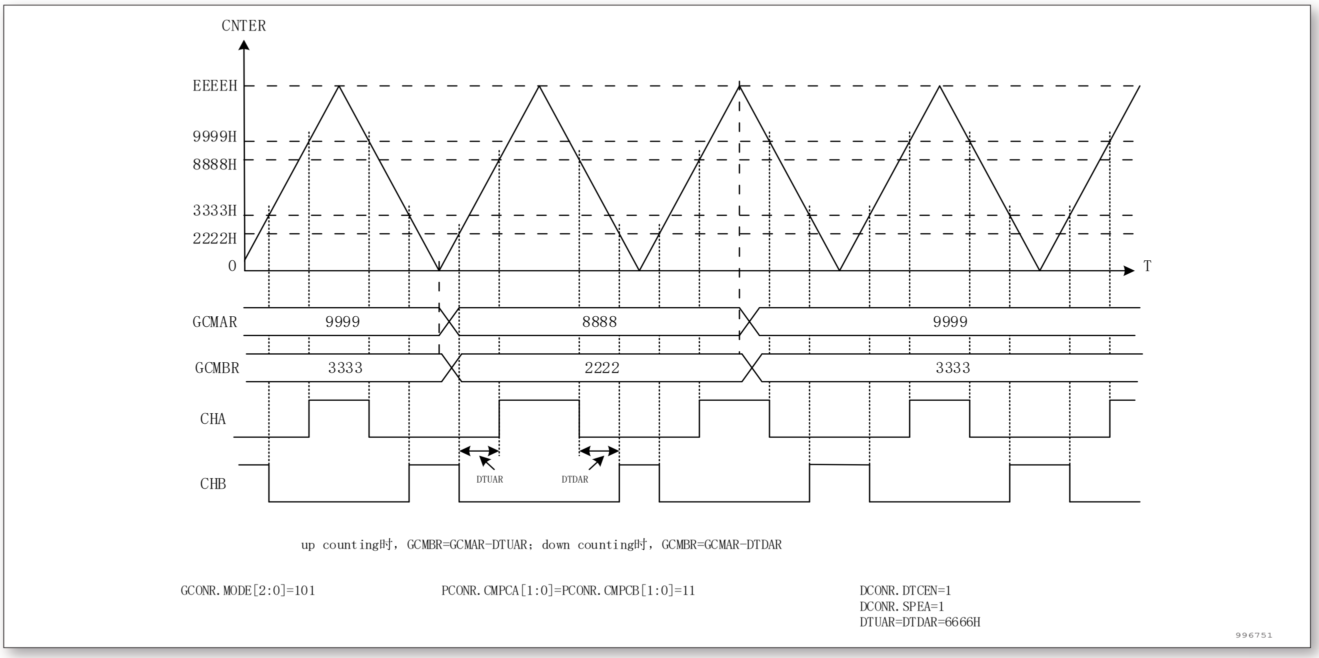

图 124. 三角波 B 模式时硬件设定 GCMBR 互补 PWM 波输出（对称死区）

多相 PWM 输出

每个定时器的 CHxA、CHxB 端口都能输出 2 相独立的 PWM 波或一组互补 PWM 波，多个定时器间组合，同

时结合软件、硬件同步动作就可实现多相 PWM 波输出。如 图 125，Timer4，Timer5，Timer6 组合输出 6 相 PWM 波；如图 126，Timer4，Timer5，Timer6 组合输出 3 相互补 PWM 波。

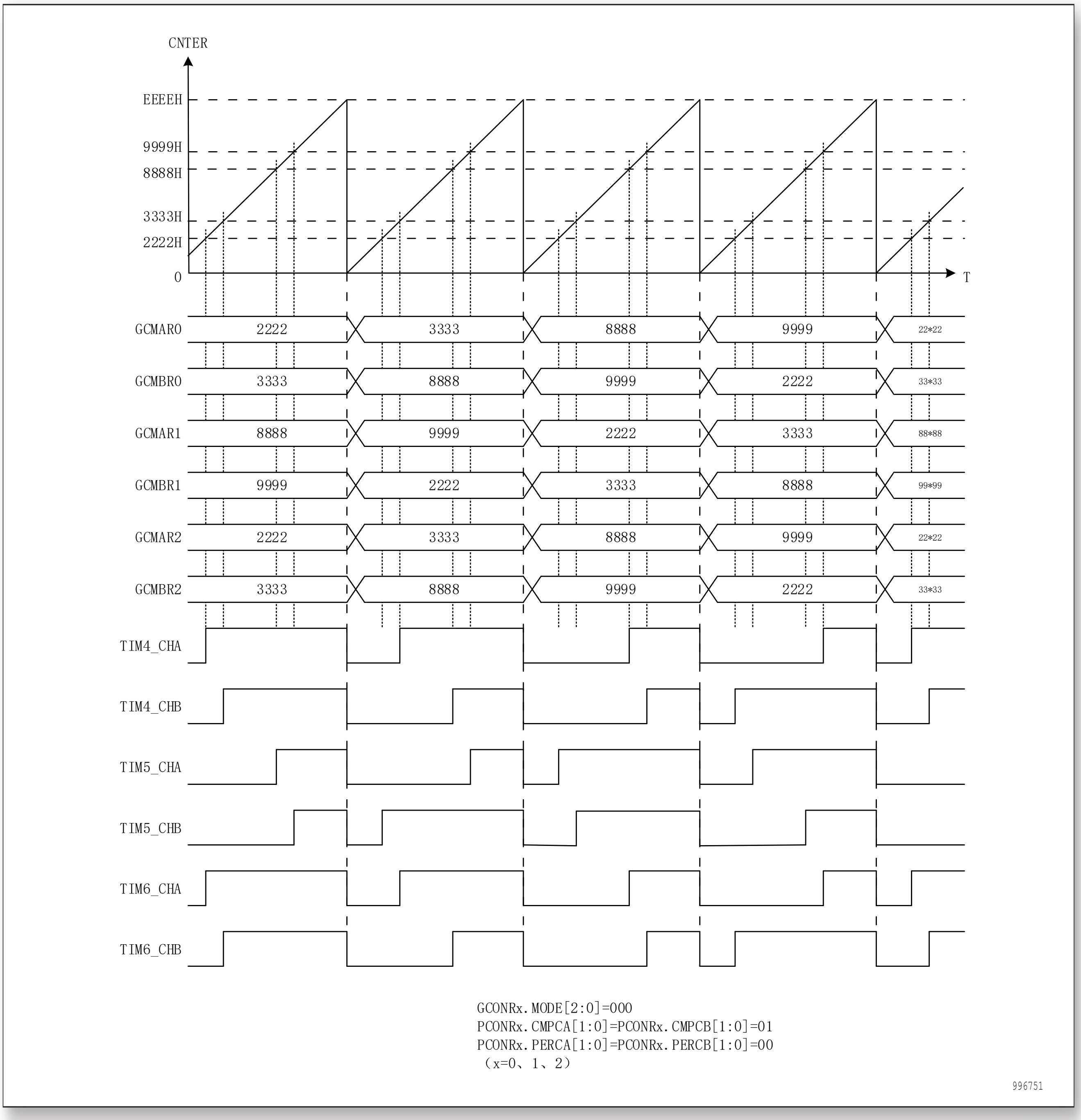

图 125. 6 相 PWM 波

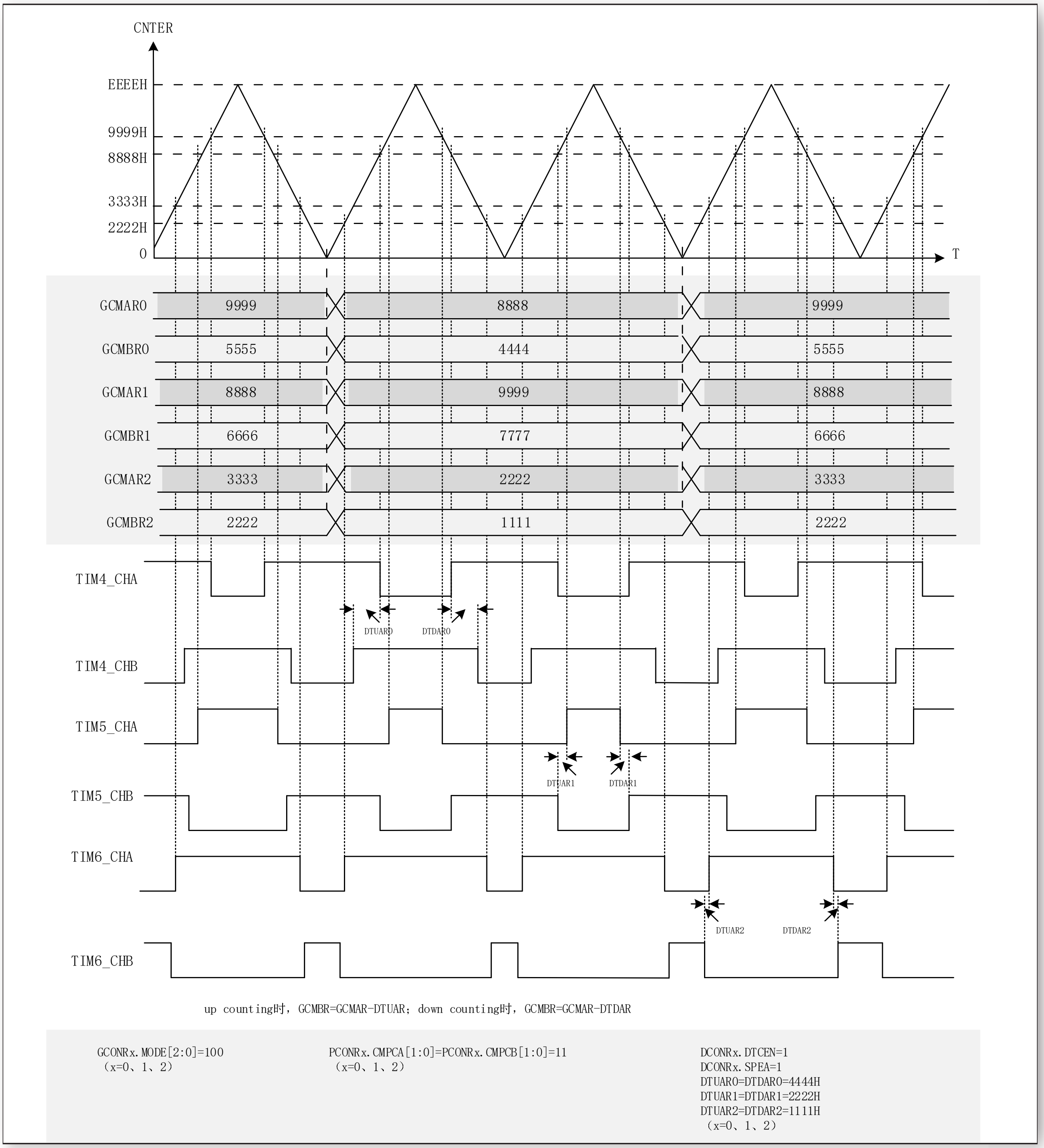

图 126. 三角波 A 模式时带死区时间三相互补 PWM 波输出

18.2.9 正交编码计数

将 CHxA 输入看作 AIN 输入、CHxB 输入看作 BIN 输入、TIMTRIA-D 中的任意一个输入看作 ZIN 输入，Advanced Timer 就可以实现三路输入的正交编码计数。

一个定时器的 AIN、BIN 单独动作可以实现位置计数模式；两个定时器的 AIN、BIN、ZIN 组合动作可以实现公转计数模式，一个定时器用于位置计数，一个定时器用于公转计数。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

公转计数模式时，每两个定时器组合（定时器 4、5 组合，定时器 4 作为位置计数单元，定时器 5 作为公转计数单元）分别实现位置计数和公转计数。

AIN 和 BIN 的计数条件通过设定硬件递加事件选择寄存器（HCUPR）和硬件递减事件选择寄存器（HCDOR）中 CHxA 和 CHxB 的正交关系实现；ZIN 的输入动作通过设定位置单元的硬件清零事件选择寄存器（HCLRR）实现位置计数单元的位置计数器清零、通过设定公转单元的硬件递加事件选择寄存器（HCUPR）实现公转计数单元的公转计数器计数。

位置计数模式

正交编码位置模式，是指根据 AIN、BIN 的输入实现基本计数功能、相位差计数功能和方向计数功能。

- 基本计数；

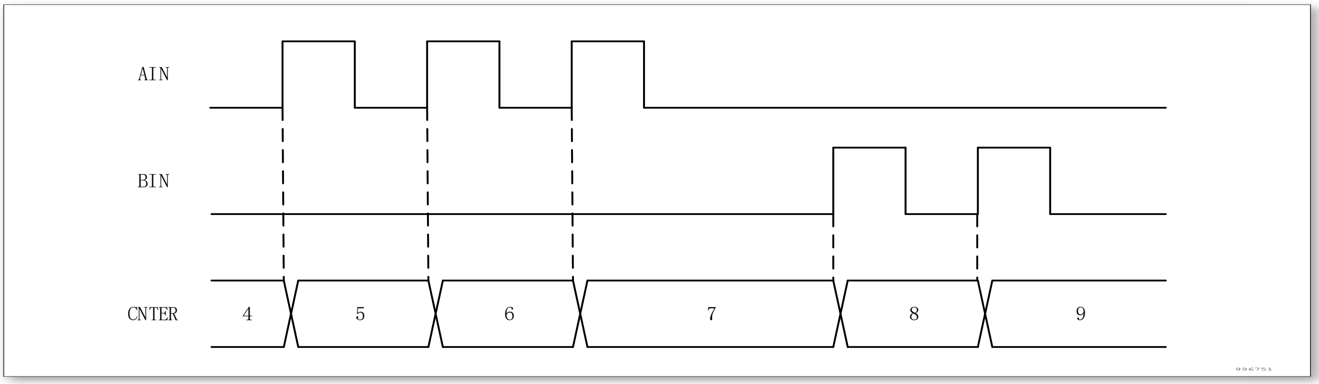

图 127. 位置模式时基本计数动作

通过设定 HCUPR 与 HCDOR 寄存器，可以灵活的实现各种方式的相位差计数。

- 相位差计数；

相位差计数是指根据 AIN 和 BIN 的相位关系进行计数。根据设定的不同，可以实现 1 倍计数、2 倍计数、4 倍计数等，如图 128~图 130所示。

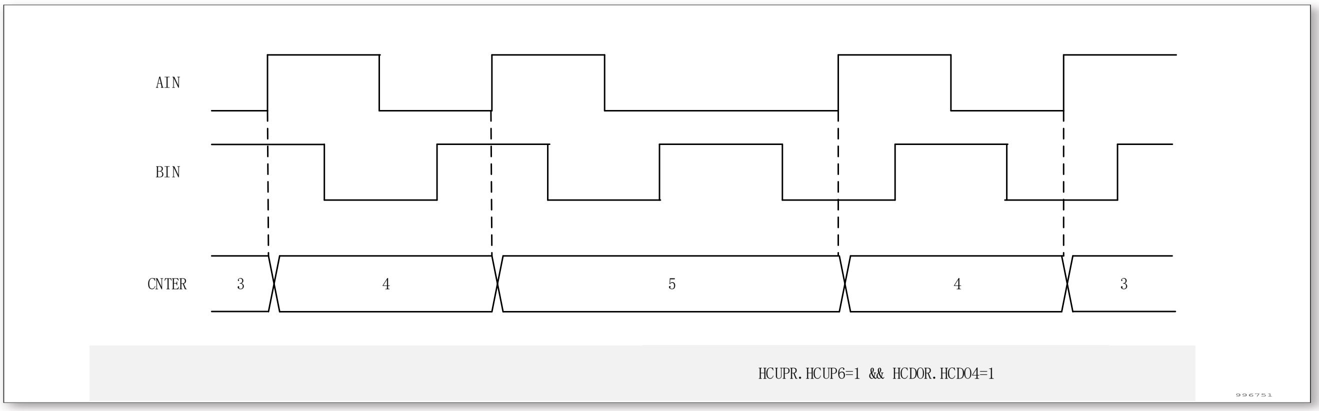

图 128. 位置模式时相位差计数动作设定 (1 倍)

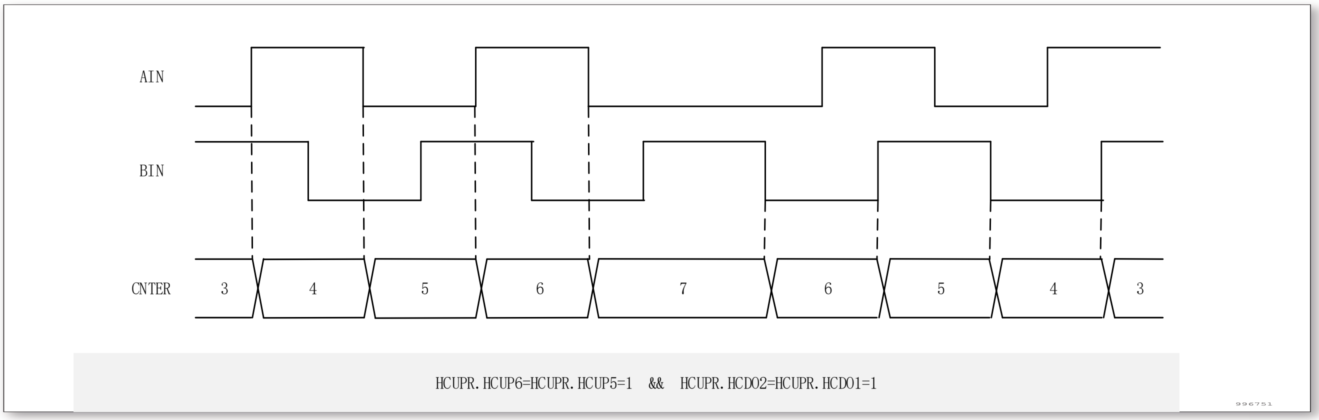

图 129. 位置模式时相位差计数动作设定 (2 倍)

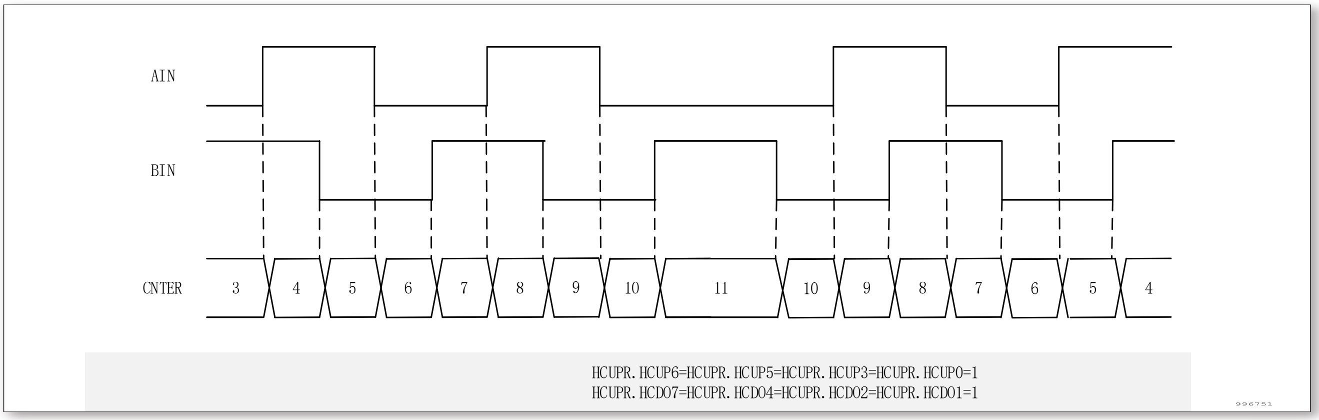

图 130. 位置模式时相位差计数动作设定 (4 倍)

- 方向计数；

方向计数是指将 AIN 的输入状态设定为方向控制，将 BIN 的输入作为时钟计数，如 图 131 所示。

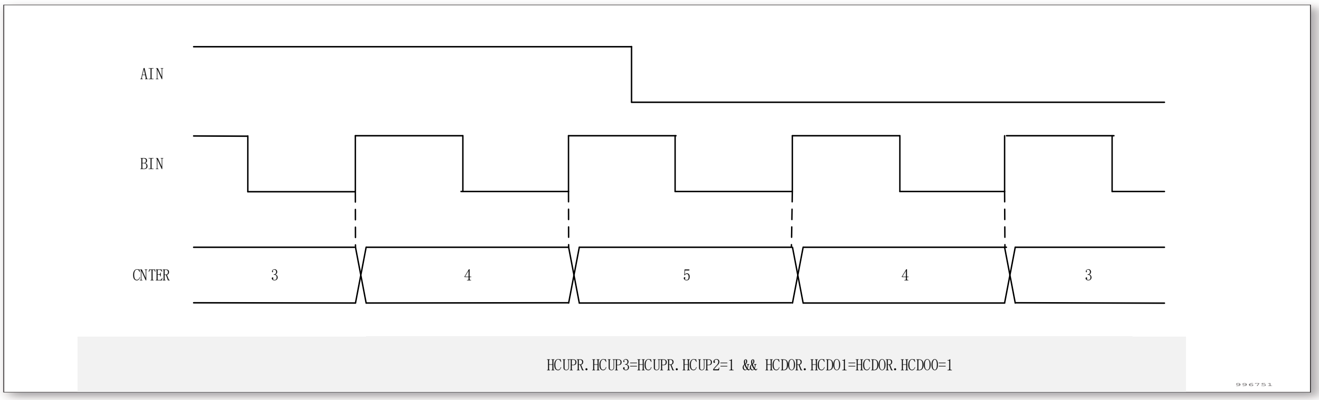

图 131. 位置模式时方向计数动作

公转模式

正交编码公转模式，是指在 AIN、BIN 计数的基础上，加入 ZIN 的输入事件以实现对公转圈数等的判断。公转模式时根据公转计数器的计数方式，可实现 Z 相计数功能、位置计数器输出计数功能和 Z 相计数与位置计数器输出混合计数功能。即使用两个 Advanced Timer 实现此功能。

- Z 相计数；

Z 相计数是指根据 ZIN 的输入，公转计数单元进行计数，同时将位置计数单元清零的计数动作。

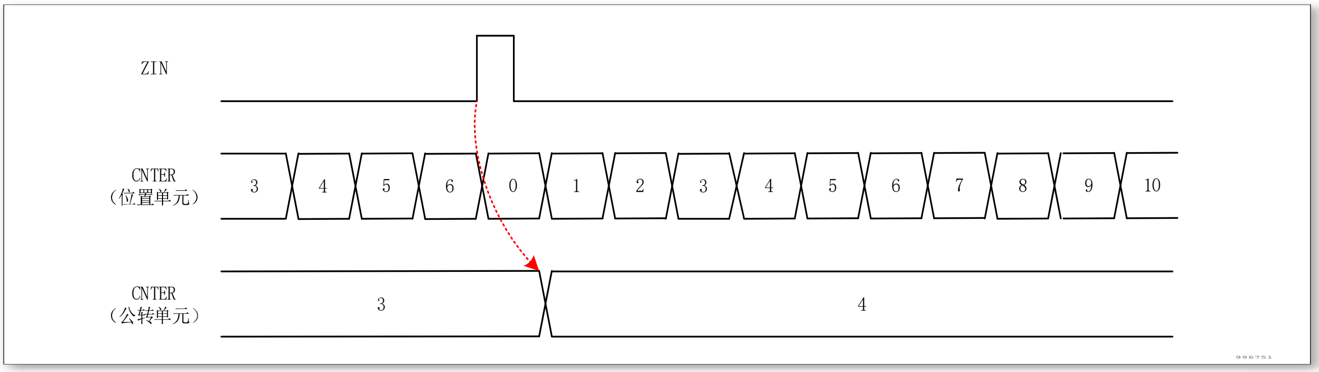

图 132. 公转模式时 Z 相计数动作

- 位置溢出计数；

位置溢出计数是指位置计数单元计数发生上溢或下溢时，产生一个溢出事件，从而触发公转计数单元的计数器进行一次计数（在该计数方式时 ZIN 的输入不进行公转计数单元的计数动作和位置计数单元的清零动作）。

位置计数单元的溢出事件通过 AOS 模块的联动选通实现公转计数单元计数，即可实现位置溢出计数。公转计数单元的硬件递加（递减）事件选择寄存器（HCUPR 或 HCDOR）的递加（递减）事件选择 Bit16:Bit19 中的 1 位，同时 AOS 模块设定对应的递加（递减）事件的事件源为位置计数单元的计数溢出事件，具体请参考 AOS 章节。如图 133 所示。

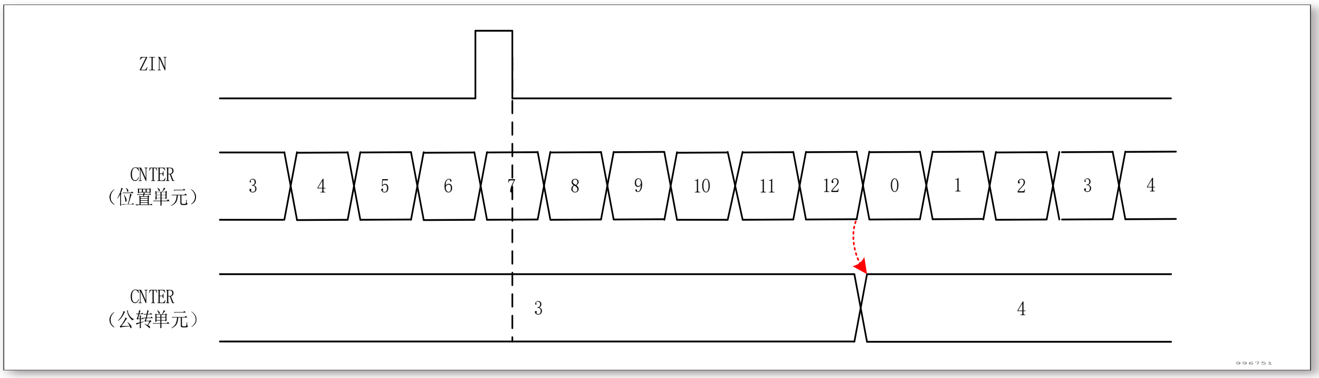

图 133. 公转模式时位置计数器输出计数动作

- 混合计数；

混合计数是指上述 Z 相计数和位置溢出计数两种计数方式合并起来的计数动作，其实现方式也是上述两种计数方式的组合。如图 134 所示。

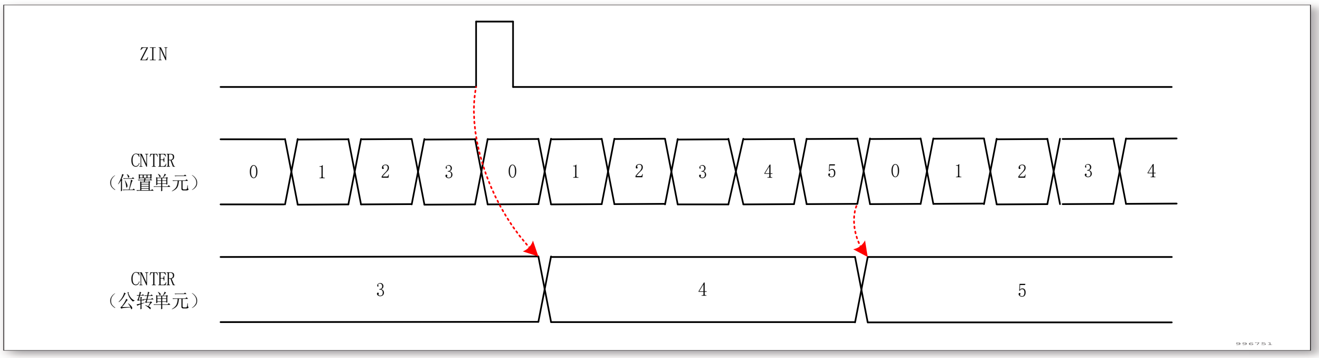

图 134. 公转模式时 Z 相计数和位置计数器输出混合计数动作

- Z 相动作屏蔽。

在公转计数模式的 Z 相计数功能或混合计数功能时，可以设定在位置计数器的上溢点或下溢点后的几个周期内（GCONR.ZMSK[0:1] 设定），将 ZIN 的有效输入屏蔽，不进行公转计数单元的计数和位置计数单元的清零。

位置计数单元的通用控制寄存器（GCONR）的 GCONR.ZMSKPOS 为 1 时，位置计数单元的 Z 相屏蔽功能使能，Z 相屏蔽的周期数由 GCONR.ZMSK 设定；公转计数单元的通用控制寄存器（GCONR）的 GCONR.ZMSKREV 为 1 时，公转计数单元的 Z 相屏蔽功能使能。

图 135 是公转计数模式混合计数时，在位置计数单元计数上溢后的 4 个计数周期内有 ZIN 相输入时，ZIN 相输入的动作无效，即公转计数单元不计数、位置计数单元不清零；之后再来的 ZIN 相输入正常动作。

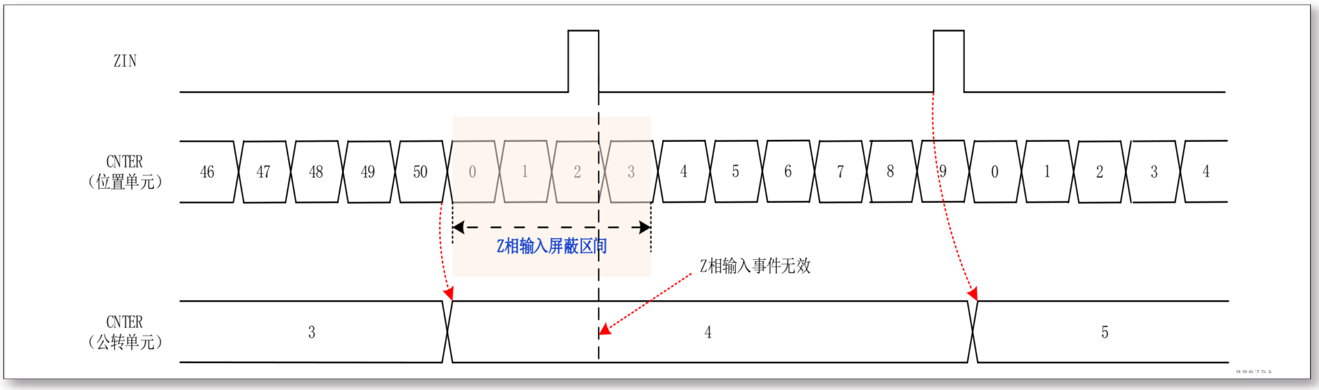

图 135. 公转计数模式-混合计数 Z 相屏蔽动作例 1

图 136 是公转计数模式混合计数时，在位置计数单元计数上溢后的第 3 个周期，计数方向发生变化，此时设定的 4 个周期的屏蔽周期变为无效（实际 ZIN 相屏蔽功能维持了 3 个周期），开始向下计数。在位置计数单元发生计数下溢后，ZIN 相屏蔽功能重新开启，维持 4 个周期后变为无效。在 ZIN 相屏蔽期间，ZIN 相的输入功能无效，即公转计数单元不计数、位置计数单元不清零；之后再来的 ZIN 相输入正常动作。

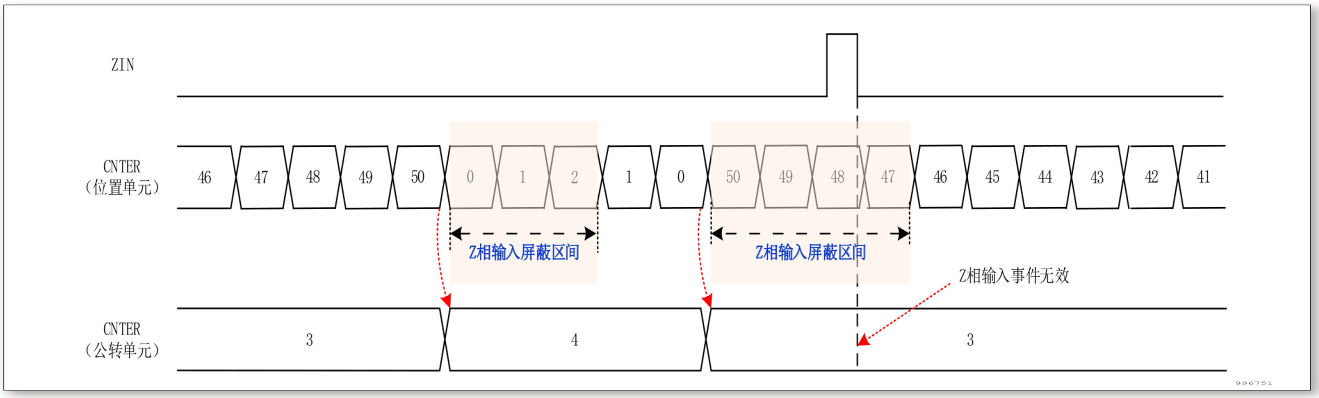

图 136. 公转计数模式-混合计数 Z 相屏蔽动作例 2

18.2.10 周期间隔响应

Timer4/5/6 的通用比较基准值寄存器（GCMAR~GCMDR），在计数比较匹配时可分别产生专用有效请求信号，送到 AOS 模块中用于和其它模块关联动作。

该请求信号可以每间隔几个周期后产生一次有效的请求信号。通过设定有效周期寄存器（VPERR）的 VPERR.PCNTS 位来指定每隔多少个周期请求信号有效一次，其它周期内即使计数值和比较基准值寄存器 GCMAR 或 GCMBR 的值相等，也不会输出有效的请求信号。图 137 所示是周期间隔有效请求信号的动作例。

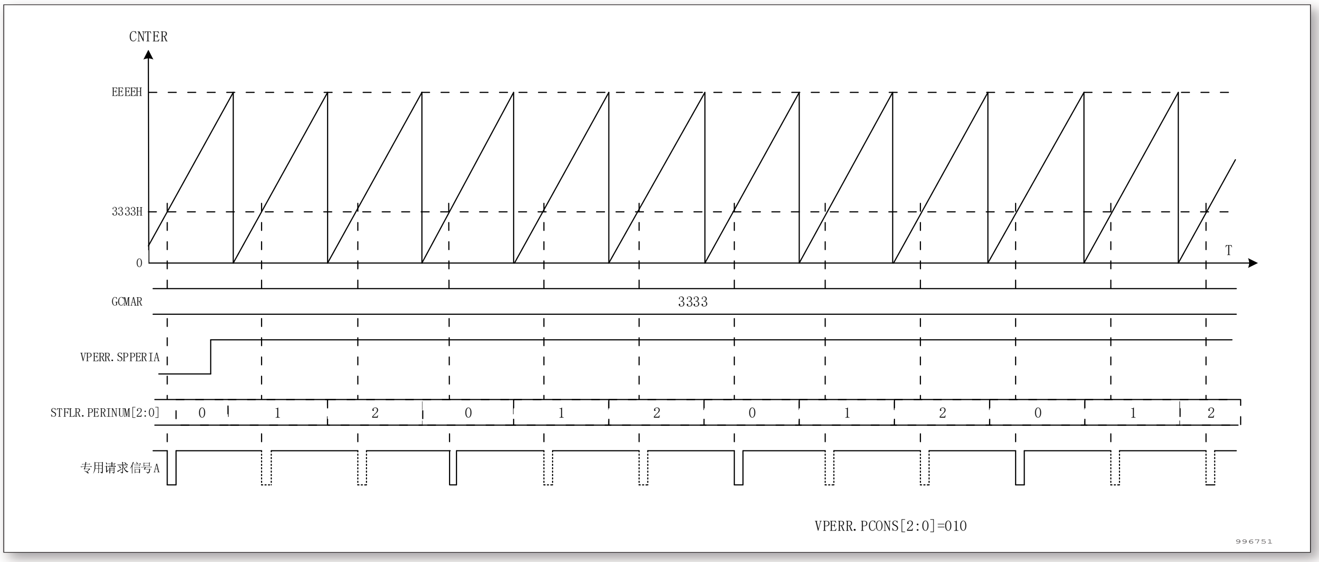

图 137. 周期间隔有效请求信号动作

18.2.11　保护机制

Advanced Timer 可以对端口的输出状态进行保护控制。

Advanced Timer 有 4 个共用的端口输出无效事件接口，这 4 个接口连接刹车控制模块输出的 4 组刹车事件。每个接口上选通的异常状况事件可从刹车控制设定，当这些接口上监测到异常状况时，可以实现对通用 PWM 输出的控制。

端口在正常输出期间，若监测到从刹车控制过来的刹车事件，则端口的输出状态可变为预先设定好的状态。通用 PWM 输出端口在刹车控制异常事件发生时，端口状态可以变为输出高阻态、输出低电平或输出高电平（PCONR.DISVALA、PCONR.DISVALB 的设定决定）。

例如，若 PCONR.DISSELA[1:0]=01&PCONR.DISVALA=01 设定时，则在 CHxA 端口正常输出期间，若输出无效条件 1 上产生刹车事件，则 CHxA 端口上输出变为高阻态。

18.2.12　中断说明

Timer4/5/6 各含有 3 类共计 9 个中断。分别是 4 个通用计数比较匹配中断（含 2 个捕获输入中断）、2 个计数周期匹配中断、1 个死区时间错误中断。

计数比较匹配中断

通用比较基准值寄存器（GCMAR-GCMDR）共计 4 个，可分别与计数值比较产生比较匹配有效信号。计数比较匹配时，状态标志寄存器（STFLR ）中的 STFLR.CMAF~STFLR.CMDF 位分别会被置为 1。此时若设定中断控制寄存器（ICONR）的 ICONR.INTENA~ICONR.INTEND 中相应位为 1 使能中断，则对应的中断请求也会被触发。

在硬件捕获事件选择寄存器（HCPAR、HCPBR）选择的捕获输入有效条件产生时，捕获输入动作发生。此时若设置中断控制寄存器（ICONR）的 ICONR.INTENA 或 ICONR.INTENB 位为 1 使能中断，则对应的中断请求被触发。

计数周期匹配中断

锯齿波递加计数至上溢点、锯齿波递减计数至下溢点、三角波计数至谷点或三角波计数至峰点时，状态标志

寄存器（STFLR）的 STFLR.OVFF 或 STFLR.UDFF 位会被置为 1。此时若设置中断控制寄存器（ICONR）的 ICONR.INTENOVF 位与 ICONR.INTENUDF 位使能中断，则在对应的时间点可触发计数周期匹配中断。

死区时间错误中断

将死区时间基准值寄存器（DTUAR、DTDAR）的值加载到通用比较基准值寄存器（GCMBR）中时，若超过周期限制，则会产生死区时间错误，状态标志寄存器（STFLR）的 STFLR.DTEF 位会被置为 1。此时若设置中断控制寄存器（ICONR）的 ICONR.INTENDE 位使能中断，则会在该时刻触发死区时间错误中断。

18.2.13 DMA

Timer 支持软件和硬件触发 DMA 进行数据传输。支持数据从其他位置写入定时器，或从定时器读出写入其他位置。可应用于数据捕获后数据的自动搬运和更改周期值或占空比的脉冲宽度的自动调整。每个定时器有两个 DMA 请求,A 请求触发源可选择通用比较捕获 A,C, 专用比较 A, 计数上溢出，B 请求触发源可选择通用比较捕获 B、D, 专用比较 B，计数下溢出。

表 70: 外设的中断信号

IDREQ	Interrupt Signal of Peripheral
26	TIM4A
27	TIM4B
28	TIM5A
29	TIM5B
30	TIM6A
31	TIM6B

TIMxA 的触发源可选择比较捕获 A，C，计数上溢出，专用比较 A。

TIMxB 的触发源可选择比较捕获 B，D，计数下溢出，专用比较 B。

18.2.14 刹车保护

当可以设定无效条件 0~3，配置 PCONR.DISVALA, PCONR.DISVALB。无效条件有效时硬件自动将端口状态改变为预设状态（高电平，低电平，高阻态，保持正常输出）。

端口刹车与软件刹车

端口经过极性选择控制，有效使能后，经过数字滤波，同步，产生端口刹车标志；端口刹车标志做为 Advanced Timer 的无效条件 3。端口刹车标志需要软件清除。

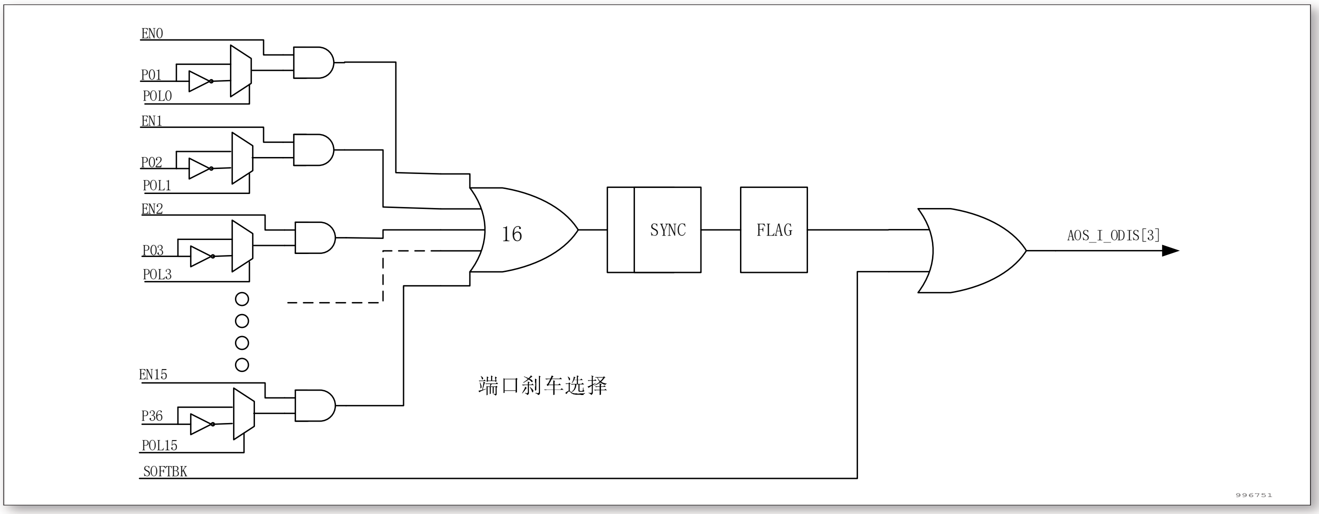

图 138. 端口刹车与软件刹车示意图

低功耗模式自动刹车

系统进入低功耗模式，时钟停止后 PWM 将不能正常工作。低功耗模式作为 Advanced Timer 的无效条件 2 控制 PWM 刹车。

输出电平同高同低刹车

输出电平经过电平监测，有效使能后，经过同步，产生同高同低刹车标志；端口刹车标志做为 Advanced Timer 的无效条件 1。同高同低刹车标志需要软件清除。

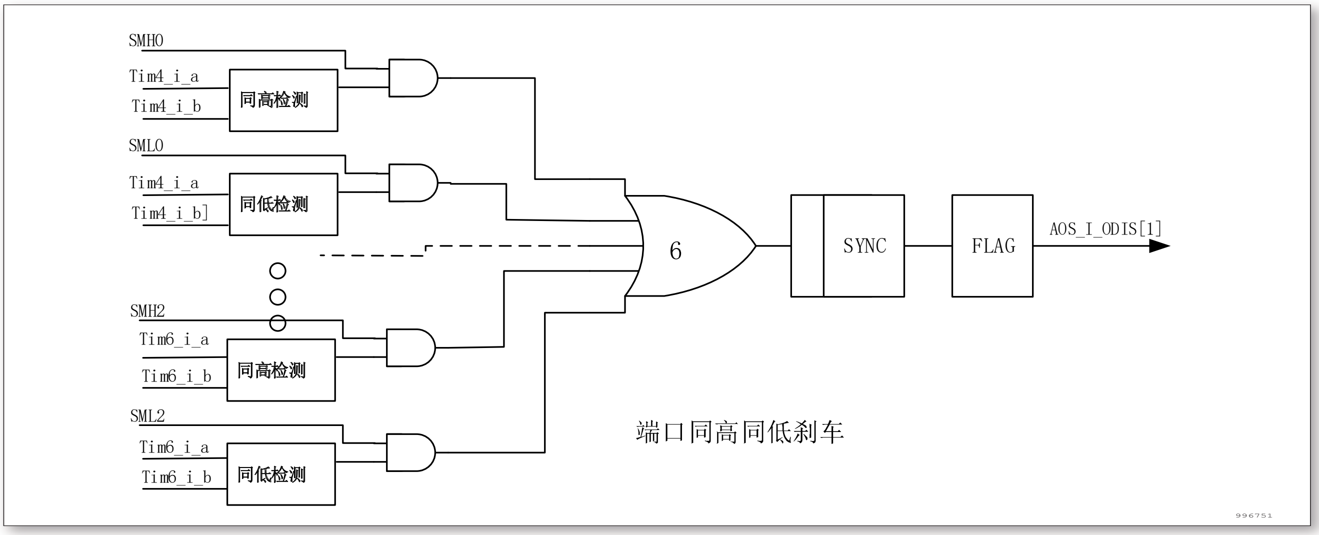

图 139. 输出同高同低刹车示意图

VC 刹车

VC0，VC1，VC2 中断标志经过使能后作为 Advanced Timer 的无效条件 0。

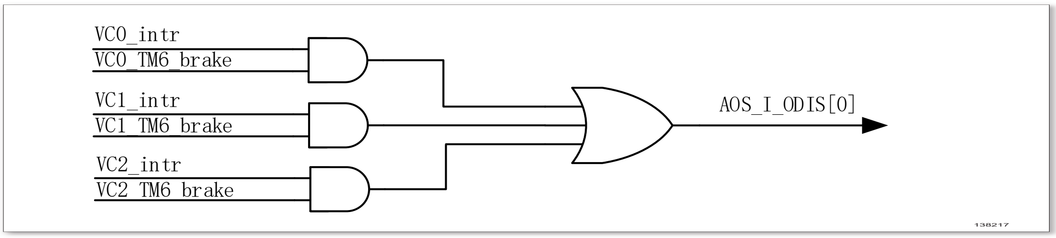

图 140. VC 刹车控制示意图

18.2.15 内部互连

中断触发输出

由于 Timer4/5/6 的一个中断包含多个中断源。控制触发 ADC 与控制 AOS 的中断信号有单独控制可以选择不同的源，可以选择上溢出、下溢出，4 个比较匹配共 6 个 TIMx 的中断源的任意中断源作为触发条件。

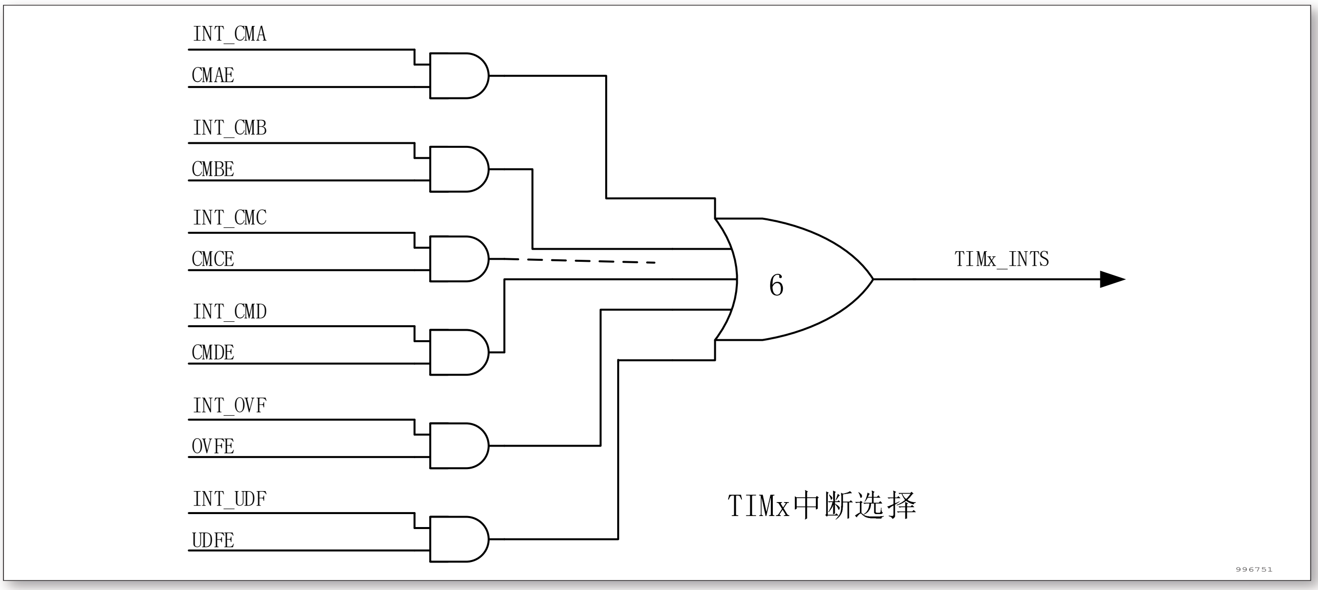

图 141. Timer4/5/6 中断选择

AOS 触发

AOS 是系统的内部信号，通过选择控制后可以触发 Advanced Timer 的计数器的开始，停止，清零，加 1，减 1 等功能。Advanced Timer 有 4 路 AOS 触发，每路触发可以选择不同模块的中断源。选择后的信号产生单脉冲触发输入到 Advanced Timer，控制 Advanced Timer 的计数器的开始，停止，清零。

Timer4/5/6 内部使用寄存器选择不同的 AOS_I_TRIG 作为自己的触发信号。如可使用 HSTAR 寄存器可以使用中断触发相应定时器的硬件启动。

表 71: AOS 源选择

	AOS_i_trig0	AOS_i_trig1	AOS_i_trig2	AOS_i_trig3
选择控制信号	ITRIG.IAOS0S	ITRIG.IAOS1S	ITRIG.IAOS2S	ITRIG.IAOS3S
0000	TIM0_INT	TIM0_INT	TIM0_INT	TIM0_INT
0001	TIM1_INT	TIM1_INT	TIM1_INT	TIM1_INT
0010	TIM2_INT	TIM2_INT	TIM2_INT	TIM2_INT
0011	LPTIMER_INT	LPTIMER_INT	LPTIMER_INT	LPTIMER_INT

	AOS_i_trig0	AOS_i_trig1	AOS_i_trig2	AOS_i_trig3
选择控制信号	ITRIG.IAOS0S	ITRIG.IAOS1S	ITRIG.IAOS2S	ITRIG.IAOS3S
0100	TIM4_INTS	TIM4_INTS	TIM4_INTS	TIM4_INTS
0101	TIM5_INTS	TIM5_INTS	TIM5_INTS	TIM5_INTS
0110	TIM6_INTS	TIM6_INTS	TIM6_INTS	TIM6_INTS
0111	UART0_INT	UART0_INT	UART0_INT	UART0_INT
1000	UART1_INT	UART1_INT	UART1_INT	UART1_INT
1001	LPUART0_INT	LPUART0_INT	LPUART0_INT	LPUART0_INT
1010	VC0_INT	VC0_INT	VC0_INT	VC0_INT
1011	VC1_INT	VC1_INT	VC1_INT	VC1_INT
1100	RTC_INT	RTC_INT	RTC_INT	RTC_INT
1101	PCA_INT	PCA_INT	PCA_INT	PCA_INT
1110	SPI_INT	SPI_INT	SPI_INT	SPI_INT
1111	ADC_INT	ADC_INT	ADC_INT	ADC_INT

端口触发 TRIGA-TRIGD

端口触发可以控制 Advanced Timer 的硬件启动、停止、清零、捕获、计数器加减计数等功能，有数字滤波功能可选，端口可以配置为芯片的任意一个端口上。

表 72: 端口触发选择

选择控制信号分别独立控制	TRIGA	TRIGB	TRIGC	TRIGD
0000	PA3	PA3	PA3	PA3
0001	PB3	PB3	PB3	PB3
0010	PC3	PC3	PC3	PC3
0011	PD3	PD3	PD3	PD3
0100	PA7	PA7	PA7	PA7
0101	PB7	PB7	PB7	PB7
0110	PC7	PC7	PC7	PC7
0111	PD7	PD7	PD7	PD7
1000	PA11	PA11	PA11	PA11
1001	PB11	PB11	PB11	PB11
1010	PC11	PC11	PC11	PC11
1011	PD1	PD1	PD1	PD1
1100	PA15	PA15	PA15	PA15
1101	PB15	PB15	PB15	PB15
1110	PC5	PC5	PC5	PC5
1111	PD5	PD5	PD5	PD5

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

比较输出 VC 与 Advanced Timer 互连

VC 内部可以互连到 Advanced Timer 的捕获输入端，可以对 VC 输出的边沿进行捕获；VC0 与 CHA 连接，VC1 与 CHB 连接；控制在 VC 控制寄存器。

18.3 TIM_Adv 寄存器描述

表 73. TIM_Adv 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x000	TIMx_CNTER	通用计数基准值寄存器	0x0000 0000	小节 18.3.1
0x004	TIMx_PERAR	通用周期基准值寄存器	0x0000 FFFF	小节 18.3.2
0x008	TIMx_PERBR	通用周期基准值缓存寄存器	0x0000 FFFF	小节 18.3.3
0x010	TIMx_GCMAR	通用比较 A 基准值寄存器	0x0000 FFFF	小节 18.3.4
0x014	TIMx_GCMBR	通用比较 B 基准值寄存器	0x0000 FFFF	小节 18.3.4
0x018	TIMx_GCMCR	通用比较 C 基准值寄存器	0x0000 FFFF	小节 18.3.4
0x01C	TIMx_GCMDR	通用比较 D 基准值寄存器	0x0000 FFFF	小节 18.3.4
0x028	TIMx_SCMAR	专用比较 A 基准值寄存器	0x0000 FFFF	小节 18.3.5
0x02C	TIMx_SCMBR	专用比较 B 基准值寄存器	0x0000 FFFF	小节 18.3.5
0x040	TIMx_DTUAR	死区时间基准值寄存器	0x0000 FFFF	小节 18.3.6
0x044	TIMx_DTDAR	死区时间基准值寄存器	0x0000 FFFF	小节 18.3.6
0x050	TIMx_GCONR	通用控制寄存器	0x0000 0100	小节 18.3.7
0x054	TIMx_ICONR	中断控制寄存器	0x0000 0000	小节 18.3.8
0x058	TIMx_PCONR	端口控制寄存器	0x0000 0000	小节 18.3.9
0x05C	TIMx_BCONR	缓存控制寄存器	0x0000 0000	小节 18.3.10
0x060	TIMx_DCONR	死区控制寄存器	0x0000 0000	小节 18.3.11
0x068	TIMx_FCONR	滤波控制寄存器	0x0000 0000	小节 18.3.12
0x06C	TIMx_VPERR	有效周期寄存器	0x0000 0000	小节 18.3.13
0x070	TIMx_STFLR	状态标志寄存器	0x0000 0000	小节 18.3.14
0x074	TIMx_HSTAR	硬件启动事件选择寄存器	0x0000 0000	小节 18.3.15
0x078	TIMx_HSTPR	硬件停止事件选择寄存器	0x0000 0000	小节 18.3.16
0x07C	TIMx_HCLRR	硬件清零事件选择寄存器	0x0000 0000	小节 18.3.17
0x080	TIMx_HCPAR	硬件捕获事件选择寄存器	0x0000 0000	小节 18.3.18
0x084	TIMx_HCPBR	硬件捕获事件选择寄存器	0x0000 0000	小节 18.3.19
0x088	TIMx_HCUPR	硬件递加事件选择寄存器	0x0000 0000	小节 18.3.20
0x08C	TIMx_HCDOR	硬件递减事件选择寄存器	0x0000 0000	小节 18.3.21
0x100	TIMx_IFR	中断标志寄存器	0x0000 0000	小节 18.3.25
0x104	TIMx_ICLR	中断清除寄存器	0x0000 0000	小节 18.3.26
0x108	TIMx_CR	展频及中断触发选择寄存器	0x0000 0300	小节 18.3.27
0x110	TIMx_AOSSR	AOS 选择寄存器，三个通道共用	0x0000 0000	小节 18.3.28

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Offset	Acronym	Register Name	Reset	Section
0x114	TIMx_AOSCL	AOS 刹车标志清除寄存器，三个通道共用	0x0000 0000	小节 18.3.29
0x118	TIMx_PTBKS	端口刹车控制寄存器, 三个通道共用	0x0000 0000	小节 18.3.30
0x11C	TIMx_TTRIG	端口触发控制寄存器, 三个通道共用	0x0000 0000	小节 18.3.31
0x120	TIMx_ITRIG	AOS 触发控制寄存器，三个通道共用	0x0000 0000	小节 18.3.32
0x124	TIMx_PTBKP	端口刹车极性控制寄存器, 三个通道共用	0x0000 0000	小节 18.3.33
0x3F4	TIMx_SSTAR	软件同步启动寄存器	0x0000 0000	小节 18.3.22
0x3F8	TIMx_SSTPR	软件同步停止寄存器	0x0000 0000	小节 18.3.23
0x3FC	TIMx_SCLRR	软件同步清零寄存器	0x0000 0000	小节 18.3.24

18.3.1 通用计数基准值寄存器 (TIMx_CNTER)

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x000

复位值：0x0000 0000

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	CNT[15:0]	rw	0x00	当前计数器的计数值

18.3.2 通用周期基准值寄存器 (TIMx_PERAR)

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x004

复位值：0x0000 FFFF

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	PERA[15:0]	rw	0xFFFF	计数周期值，设定每轮计数的计数周期值

18.3.3 通用周期基准值缓存寄存器（TIMx_PERBR）

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x008

复位值：0x0000 FFFF

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	PERB[15:0]	rw	0xFFFF	缓存计数周期值，计数周期的缓存值

18.3.4 通用比较基准值寄存器（TIMx_GCMAR-GCMDR)

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x0010,0x0014,0x0018,0x001C

复位值：0x0000 FFFF

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	GCMA-D[15:0]	rw	0xFFFF	计数比较基准值，比较基准值设定，与计数值相等时匹配信号有效

18.3.5 专用比较基准值寄存器 (TIMx_SCMAR-SCMBR)

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x0028,0x002C

复位值：0x0000 FFFF

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	SCMA-B [15:0]	rw	0x00	计数比较基准值，比较基准值设定，与计数值相等时匹配信号有效

18.3.6 死区时间基准值寄存器 (TIMx_DTUAR-DTDAR)

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x040,0x044

复位值：0x0000 FFFF

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DTUA/DTDA [15:0]															
RW															

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	DTUA/DA [15:0]	rw	0xFFFF	死区时间值，死区时间设定值

18.3.7 通用控制寄存器（TIMx_GCONR）

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x050

复位值：0x0000 0100

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved												ZMSK[1:0]		ZMSK POS	ZMSK REV
												RW		RW	RW

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							DIR	Res.	CKDIV[2:0]			MODE[2:0]			START
							RW		RW			RW			RW

Bit	Field	Type	Reset	Description
31:20	Reserved			保留
19:18	ZMSK[1:0]	rw	0x00	Z 相输入屏蔽周期数 正交编码 Z 相输入被屏蔽的计数周期值 00：Z 相输入屏蔽功能无效 01：位置计数上溢后或下溢后的 4 个计数周期内的 Z 相输入被屏蔽 10：位置计数上溢后或下溢后的 8 个计数周期内的 Z 相输入被屏蔽 11：位置计数上溢后或下溢后的 16 个计数周期内的 Z 相输入被屏蔽

Bit	Field	Type	Reset	Description
17	ZMSKPOS	rw	0x00	Z 相输入位置计数器选择 0：Z 相输入时该定时器作为位置计数器，在屏蔽周期期间内位置计数器清零功能正常动作 1：Z 相输入时该定时器作为位置计数器，在屏蔽周期期间内位置计数器清零功能被屏蔽
16	ZMSKREV	rw	0x00	Z 相输入公转计数器选择 0：Z 相输入时该定时器作为公转计数器，在屏蔽周期期间内公转计数器计数功能正常动作 1：Z 相输入时该定时器作为公转计数器，在屏蔽周期期间内公转计数器计数功能被屏蔽
15:9	Reserved			保留
8	DIR	rw	0x01	计数方向 0：递减计数；1：递加计数
7	Reserved			保留
6:4	CKDIV[2:0]	rw	0x00	计数时钟选择 000：PCLK0 001：PCLK0/2 010：PCLK0/4 011：PCLK0/8 100：PCLK0/16 101：PCLK0/64 110：PCLK0/256 111：PCLK0/1024
3:1	MODE[2:0]	rw	0x00	计数模式 000：锯齿波 A 模式 100：三角波 A 模式 101：三角波 B 模式 请不要设定其它值
0	START	rw	0x00	计数器启动 0：计数器关闭；1：计数器启动 注：该位在软件停止条件或硬件停止条件有效时，会自动变为 0

18.3.8 中断控制寄存器（TIMx_ICONR）

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x054

复位值：0x0000 0000

31 30 29 28 27 26 25 24 23 22 21 20	19	18	17	16
Reserved	INTEN SBD	INTEN SBU	INTEN SAD	INTEN SAU
	RW	RW	RW	RW

15	14	13 12 11 10 9	8	7	6	5 4	3	2	1	0
INTEN SAMH	INTEN SAML	Reserved	INTEN DE	INTEN UDF	INTEN OVF	Reserved	INTEN D	INTEN C	INTEN B	INTEN A
RW	RW		RW	RW	RW		RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31:20	Reserved			保留
19	INTENSBD	rw	0x00	专用向下计数触发 ADC 使能 B
18	INTENSBU	rw	0x00	专用向上计数触发 ADC 使能 B
17	INTENSAD	rw	0x00	专用向下计数触发 ADC 使能 A
16	INTENSAU	rw	0x00	专用向上计数触发 ADC 使能 A
15	INTENSAMH	rw	0x00	同高中断使能
14	INTENSAML	rw	0x00	同低中断使能
13:9	Reserved			保留
8	INTENDE	rw	0x00	死区时间错误中断使能 0：死区时间错误时，该中断无效 1：死区时间错误时，该中断使能
7	INTENUDF	rw	0x00	下溢中断使能 0：锯齿波时下溢发生或三角波时计数到谷点，该中断无效 1：锯齿波时下溢发生或三角波时计数到谷点，该中断使能
6	INTENOVF	rw	0x00	上溢中断使能 0：锯齿波时上溢发生或三角波时计数到峰点，该中断无效 1：锯齿波时上溢发生或三角波时计数到峰点，该中断使能
5:4	Reserved			保留
3	INTEND	rw	0x00	计数匹配中断使能 D 0：GCMDR 寄存器与计数值相等时，该中断无效 1：GCMDR 寄存器与计数值相等时，该中断使能
2	INTENC	rw	0x00	计数匹配中断使能 C 0：GCMCR 寄存器与计数值相等时，该中断无效 1：GCMCR 寄存器与计数值相等时，该中断使能

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
1	INTENB	rw	0x00	计数匹配中断使能 B 0：GCMBR 寄存器与计数值相等时，或者发生捕获输入事件时，该中断无效 1：GCMBR 寄存器与计数值相等时，或者发生捕获输入事件时，该中断使能
0	INTENA	rw	0x00	计数匹配中断使能 A 0：GCMAR 寄存器与计数值相等时，或者发生捕获输入事件时，该中断无效 1：GCMAR 寄存器与计数值相等时，或者发生捕获输入事件时，该中断使能

18.3.9 端口控制寄存器（TIMx_PCONR）

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x058

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved			DISVALB		DISSELB		OUTENB	PERCB		CMPCB		STASTPSB	STPCB	STACB	CAPCB
			RW		RW		RW	RW		RW		RW	RW	RW	RW

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			DISVALA		DISSELA		OUTENA	PERCA		CMPCA		STASTPSA	STPCA	STACA	CAPCA
			RW		RW		RW	RW		RW		RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31:29	Reserved			保留
28:27	DISVALB	rw	0x00	CHxB 输出状态控制 00：强制输出无效条件 0~3 中被选择的条件成立时，CHxB 端口正常输出 01：强制输出无效条件 0~3 中被选择的条件成立时，CHxB 端口输出高阻态 10：强制输出无效条件 0~3 中被选择的条件成立时，CHxB 端口输出低电平 11：强制输出无效条件 0~3 中被选择的条件成立时，CHxB 端口输出高电平

Bit	Field	Type	Reset	Description
26:25	DISSELB	rw	0x00	强制输出无效条件选择 B 00：选择强制输出无效条件 0；01：选择强制输出无效条件 1 10：选择强制输出无效条件 2；11：选择强制输出无效条件 3
24	OUTENB	rw	0x00	输出使能 B 0：Advanced Timer 功能时的 CHxB 端口输出无效 1：Advanced Timer 功能时的 CHxB 端口输出有效
23:22	PERCB	rw	0x00	周期值匹配时端口状态设定 B 00：计数器计数值与周期值相等时，CHxB 端口输出保持为低电平 01：计数器计数值与周期值相等时，CHxB 端口输出设定为高电平 10：计数器计数值与周期值相等时，CHxB 端口输出设定为先前状态 11：计数器计数值与周期值相等时，CHxB 端口输出设定为反转电平
21:20	CMPCB	rw	0x00	比较值匹配时端口状态设定 B 00：计数器计数值与 GCMBR 相等时，CHxB 端口输出保持为低电平 01：计数器计数值与 GCMBR 相等时，CHxB 端口输出设定为高电平 10：计数器计数值与 GCMBR 相等时，CHxB 端口输出设定为先前状态 11：计数器计数值与 GCMBR 相等时，CHxB 端口输出设定为反转电平
19	STASTPSB	rw	0x00	计数开始停止端口状态选择 B 0:计数开始或停止时,CHxB 端口输出由 STACB、STPCB 决定 1：计数开始或停止时，CHxB 端口输出设定为先前状态 注：此处的计数开始是指初始计数开始或停止再开始；计数停止是指初始时停止或计数开始后再停止
18	STPCB	rw	0x00	计数停止端口状态设定 B 0：计数停止时，CHxB 端口输出设定为低电平 1：计数停止时，CHxB 端口输出设定为高电平
17	STACB	rw	0x00	计数开始端口状态设定 B 0：计数开始时，CHxB 端口输出设定为低电平 1：计数开始时，CHxB 端口输出设定为高电平
16	CAPCB	rw	0x00	功能模式选择 B 0：比较输出功能；1：捕获输入功能
15:13	Reserved			保留

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
12:11	DISVALA	rw	0x00	CHxA 输出状态控制 00：强制输出无效条件 0~3 中被选择的条件成立时，CHxA 端口正常输出 01：强制输出无效条件 0~3 中被选择的条件成立时，CHxA 端口输出高阻态 10：强制输出无效条件 0~3 中被选择的条件成立时，CHxA 端口输出低电平 11：强制输出无效条件 0~3 中被选择的条件成立时，CHxA 端口输出高电平
10:9	DISSELA	rw	0x00	强制输出无效条件选择 A 00：选择强制输出无效条件 0；01：选择强制输出无效条件 1 10：选择强制输出无效条件 2；11：选择强制输出无效条件 3
8	OUTENA	rw	0x00	输出使能 A 0：Advanced Timer 功能时的 CHxA 端口输出无效 1：Advanced Timer 功能时的 CHxA 端口输出有效
7:6	PERCA	rw	0x00	周期值匹配时端口状态设定 A 00：计数器计数值与周期值相等时，CHxA 端口输出保持为低电平 01：计数器计数值与周期值相等时，CHxA 端口输出设定为高电平 10：计数器计数值与周期值相等时，CHxA 端口输出设定为先前状态 11：计数器计数值与周期值相等时，CHxA 端口输出设定为反转电平
5:4	CMPCA	rw	0x00	比较值匹配时端口状态设定 A 00：计数器计数值与 GCMAR 相等时，CHxA 端口输出保持为低电平 01：计数器计数值与 GCMAR 相等时，CHxA 端口输出设定为高电平 10：计数器计数值与 GCMAR 相等时，CHxA 端口输出设定为先前状态 11：计数器计数值与 GCMAR 相等时，CHxA 端口输出设定为反转电平
3	STASTPSA	rw	0x00	计数开始停止端口状态选择 A 0:计数开始或停止时,CHxA 端口输出由 STACA、STPCA 决定 1：计数开始或停止时，CHxA 端口输出设定为先前状态 注：此处的计数开始是指初始计数开始或停止再开始；计数停止是指初始时停止或计数开始后再停止

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
2	STPCA	rw	0x00	计数停止端口状态设定 A 0：计数停止时，CHxA 端口输出设定为低电平； 1：计数停止时，CHxA 端口输出设定为高电平
1	STACA	rw	0x00	计数开始端口状态设定 A 0：计数开始时，CHxA 端口输出设定为低电平 1：计数开始时，CHxA 端口输出设定为高电平
0	CAPCA	rw	0x00	功能模式选择 A 0：比较输出功能；1：捕获输入功能

18.3.10 缓存控制寄存器（TIMx_BCONR）

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x05C

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							BENP	Reserved					BENB	Res.	BENA
							RW						RW		RW

Bit	Field	Type	Reset	Description
31:9	Reserved			保留
8	BENP	rw	0x00	周期值缓存传送 0：缓存传送无效 1：缓存传送使能（PERBR->PERAR）
7:3	Reserved			保留
2	BENB	rw	0x00	通用比较值缓存传送 B 0：缓存传送无效 1：缓存传送使能 比较输出功能时：（GCMDR->GCMBR）； 捕获输入功能时：（GCMBR->GCMDR）
1	Reserved			保留
0	BENA	rw	0x00	通用比较值缓存传送 A 0：缓存传送无效 1：缓存传送使能 比较输出功能时：（GCMCR->GCMAR）； 捕获输入功能时：（GCMAR->GCMCR）

Bit	Field	Type	Reset	Description

18.3.11 死区控制寄存器（TIMx_DCONR）

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x060

复位值：0x0000 0000

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

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

Bit	Field	Type	Reset	Description
31:9	Reserved			保留
8	SEPA	rw	0x00	分离设定 0：DTUAR 和 DTDAR 分别设定 1：DTDAR 的值和 DTUAR 的值自动相等
7:1	Reserved			保留
0	DTCEN	rw	0x00	死区功能 0：死区功能无效 1：死区功能有效

18.3.12 滤波控制寄存器（TIMx_FCONR）

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x068

复位值：0x0000 0000

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

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

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31	Reserved			保留
30:29	NOFICKTD	rw	0x00	TRID 端口滤波采样基准时钟选择 00：PCLK0 01：PCLK0/4 10：PCLK0/16 11：PCLK0/64
28	NOFIENTD	rw	0x00	TRID 端口捕获输入滤波使能，0 无效；1 使能
27	Reserved			保留
26:25	NOFICKTC	rw	0x00	TRIC 端口滤波采样基准时钟选择 00：PCLK0 01：PCLK0/4 10：PCLK0/16 11：PCLK0/64
24	NOFIENTC	rw	0x00	TRIC 端口捕获输入滤波使能，0 无效；1 使能
23	Reserved			保留
22:21	NOFICKTB	rw	0x00	TRIB 端口滤波采样基准时钟选择 00：PCLK0 01：PCLK0/4 10：PCLK0/16 11：PCLK0/64
20	NOFIENTB	rw	0x00	TRIB 端口捕获输入滤波使能，0 无效；1 使能
19	Reserved			保留
18:17	NOFICKTA	rw	0x00	TRIA 端口滤波采样基准时钟选择 00：PCLK0 01：PCLK0/4 10：PCLK0/16 11：PCLK0/64
16	NOFIENTA	rw	0x00	TRIA 端口捕获输入滤波使能，0 无效；1 使能
15:7	Reserved			保留
6:5	NOFICKGB	rw	0x00	CHxIB 端口滤波采样基准时钟选择 00：PCLK0 01：PCLK0/4 10：PCLK0/16 11：PCLK0/64
4	NOFIENGB	rw	0x00	CHxIB 端口捕获输入滤波使能，0 无效；1 使能
3	Reserved			保留
2:1	NOFICKGA	rw	0x00	CHxIA 端口滤波采样基准时钟选择 00：PCLK0 01：PCLK0/4 10：PCLK0/16 11：PCLK0/64
0	NOFIENGA	rw	0x00	CHxIA 端口捕获输入滤波使能，0 无效；1 使能

注意：

-TRIGA-D 滤波设置只有在 TIM4 中设置有效，在 Timer5/6 设置无效。

18.3.13 有效周期寄存器 (TIMx_VPERR)

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x06C

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31:21	20:19		17:16
Reserved	PCNTS		PCNTE
	RW		RW

15:4	3	2	1	0
Reserved	GEPERID	GEPERIC	GEPERIB	GEPERIA
	RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31:21	Reserved			保留
20:18	PCNTS	rw	0x00	有效周期选择 000：有效周期选择功能无效 001：每隔 1 个周期有效一次 010：每隔 2 个周期有效一次 011：每隔 3 个周期有效一次 100：每隔 4 个周期有效一次 101：每隔 5 个周期有效一次 110：每隔 6 个周期有效一次 111：每隔 7 个周期有效一次
17:16	PCNTE	rw	0x00	有效周期计数条件选择 00：有效周期选择功能无效 01：锯齿波计数上、下溢点或三角波波谷做为计数条件 10：锯齿波计数上、下溢点或三角波波峰做为计数条件 11：锯齿波计数上、下溢点或三角波波谷、波峰做为计数条件
15:4	Reserved			保留
3	GEPERID	rw	0x00	通用信号有效周期选择 D 0：有效周期选择功能无效；1：有效周期选择功能使能
2	GEPERIC	rw	0x00	通用信号有效周期选择 C 0：有效周期选择功能无效；1：有效周期选择功能使能
1	GEPERIB	rw	0x00	通用信号有效周期选择 B 0：有效周期选择功能无效；1：有效周期选择功能使能
0	GEPERIA	rw	0x00	通用信号有效周期选择 A 0：有效周期选择功能无效；1：有效周期选择功能使能

18.3.14 状态标志寄存器（TIMx_STFLR）

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x070

复位值：0x0000 0000

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td>DIRF</td><td colspan="7" rowspan="2">Reserved</td><td colspan="3">VPERNUM</td><td colspan="5" rowspan="2">Reserved</td></tr>
<tr><td>R</td><td colspan="3">R</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="3" rowspan="2">Reserved</td><td>CMS
BDF</td><td>CMS
BUF</td><td>CMS
ADF</td><td>CMS
AUF</td><td>DTEF</td><td>UDFF</td><td>OVFF</td><td colspan="2" rowspan="2">Reserved</td><td>CMD
F</td><td>CMC
F</td><td>CMB
F</td><td>CMA
F</td></tr>
<tr><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td></tr>
</table>

Bit	Field	Type	Reset	Description
31	DIRF	rw	0x00	计数方向 0：递减计数 1：递加计数
30:24	Reserved			保留
23:21	VPERNUM	rw	0x00	周期次数 有效周期选择功能使能时，计数后的周期次数
20:13	Reserved			保留
12	CMSBDF	rw	0x00	向下计数专用比较基准值匹配 B
11	CMSBUF	rw	0x00	向上计数专用比较基准值匹配 B
10	CMSADF	rw	0x00	向下计数专用比较基准值匹配 A
9	CMSAUF	rw	0x00	向上计数专用比较基准值匹配 A
8	DTEF	rw	0x00	死区时间错误 0：未发生死区时间错误；1：发生死区时间错误
7	UDFF	rw	0x00	下溢匹配 0：未发生锯齿波下溢或三角波计数到谷点 1：发生锯齿波下溢或三角波计数到谷点
6	OVFF	rw	0x00	上溢匹配 0：未发生锯齿波上溢或三角波计数到峰点 1：发生锯齿波上溢或三角波计数到峰点
5:4	Reserved			保留
3	CMDF	rw	0x00	计数匹配 D 0：GCMDR 寄存器的值与计数值不相等 1：GCMDR 寄存器的值与计数值相等
2	CMCF	rw	0x00	计数匹配 C 0：GCMCR 寄存器的值与计数值不相等 1：GCMCR 寄存器的值与计数值相等

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
1	CMBF	rw	0x00	计数匹配 B 0：GCMBR 寄存器的值与计数值不相等，且未发生 CHxB 捕获完成动作 1：GCMBR 寄存器的值与计数值相等，或发生 CHxB 捕获完成动作
0	CMAF	rw	0x00	计数匹配 A 0：GCMAR 寄存器的值与计数值不相等，且未发生 CHxA 捕获完成动作 1：GCMAR 寄存器的值与计数值相等，或发生 CHxA 捕获完成动作

18.3.15 硬件启动事件选择寄存器（TIMx_HSTAR）

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x074

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DIRF	Reserved							VPERNUM			Reserved				
R								R							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			CMSBDF	CMSBUF	CMSADF	CMSAUF	DTEF	UDFF	OVFF	Reserved		CMDF	CMCF	CMBF	CMAF
			RW	RW	RW	RW	RW	RW	RW			RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31	STARTS	rw	0x00	硬件启动使能 0：硬件启动无效 1：硬件启动有效 注：硬件启动有效时，SSTAR 的设定无效
30:16	Reserved			保留
15	HSTA15	rw	0x00	硬件启动条件 15：TIMTRID 端口上采样到下降沿 0：条件匹配时，硬件启动无效 1：条件匹配时，硬件启动有效
14	HSTA14	rw	0x00	硬件启动条件 14：TIMTRID 端口上采样到上升沿 0：条件匹配时，硬件启动无效 1：条件匹配时，硬件启动有效

Bit	Field	Type	Reset	Description
13	HSTA13	rw	0x00	硬件启动条件 13：TIMTRIC 端口上采样到下降沿 0：条件匹配时，硬件启动无效 1：条件匹配时，硬件启动有效
12	HSTA12	rw	0x00	硬件启动条件 12：TIMTRIC 端口上采样到上升沿 0：条件匹配时，硬件启动无效 1：条件匹配时，硬件启动有效
11	HSTA11	rw	0x00	硬件启动条件 11：TIMTRIB 端口上采样到下降沿 0：条件匹配时，硬件启动无效 1：条件匹配时，硬件启动有效
10	HSTA10	rw	0x00	硬件启动条件 10：TIMTRIB 端口上采样到上升沿 0：条件匹配时，硬件启动无效 1：条件匹配时，硬件启动有效
9	HSTA9	rw	0x00	硬件启动条件 9：TIMTRIA 端口上采样到下降沿 0：条件匹配时，硬件启动无效 1：条件匹配时，硬件启动有效
8	HSTA8	rw	0x00	硬件启动条件 8：TIMTRIA 端口上采样到上升沿 0：条件匹配时，硬件启动无效 1：条件匹配时，硬件启动有效
7	HSTA7	rw	0x00	硬件启动条件 7：CHxB 端口上采样到下降沿 0：条件匹配时，硬件启动无效 1：条件匹配时，硬件启动有效
6	HSTA6	rw	0x00	硬件启动条件 6：CHxB 端口上采样到上升沿 0：条件匹配时，硬件启动无效 1：条件匹配时，硬件启动有效
5	HSTA5	rw	0x00	硬件启动条件 5：CHxA 端口上采样到下降沿 0：条件匹配时，硬件启动无效 1：条件匹配时，硬件启动有效
4	HSTA4	rw	0x00	硬件启动条件 4：CHxA 端口上采样到上升沿 0：条件匹配时，硬件启动无效 1：条件匹配时，硬件启动有效
3	HSTA3	rw	0x00	硬件启动条件 3：从 AOS 来的事件触发 3 有效 0：条件匹配时，硬件启动无效 1：条件匹配时，硬件启动有效
2	HSTA2	rw	0x00	硬件启动条件 2：从 AOS 来的事件触发 2 有效 0：条件匹配时，硬件启动无效 1：条件匹配时，硬件启动有效
1	HSTA1	rw	0x00	硬件启动条件 1：从 AOS 来的事件触发 1 有效 0：条件匹配时，硬件启动无效 1：条件匹配时，硬件启动有效

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
0	HSTA0	rw	0x00	硬件启动条件 0：从 AOS 来的事件触发 0 有效 0：条件匹配时，硬件启动无效 1：条件匹配时，硬件启动有效

18.3.16 硬件停止事件选择寄存器（TIMx_HSTPR）

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x078

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HSTP15	HSTA14	HSTP13	HSTP12	HSTP11	HSTP10	HSTP9	HSTP8	HSTP7	HSTP6	HSTP5	HSTP4	HSTP3	HSTP2	HSTP1	HSTP0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31	STOPS	rw	0x00	硬件停止使能 0：硬件停止无效 1：硬件停止有效 注：硬件停止有效时，软件停止的设定无效
30:16	Reserved			保留
15	HSTP15	rw	0x00	硬件停止条件 15：TIMTRID 端口上采样到下降沿 0：条件匹配时，硬件停止无效 1：条件匹配时，硬件停止有效
14	HSTP14	rw	0x00	硬件停止条件 14：TIMTRID 端口上采样到上升沿 0：条件匹配时，硬件停止无效 1：条件匹配时，硬件停止有效
13	HSTP13	rw	0x00	硬件停止条件 13：TIMTRIC 端口上采样到下降沿 0：条件匹配时，硬件停止无效 1：条件匹配时，硬件停止有效
12	HSTP12	rw	0x00	硬件停止条件 12：TIMTRIC 端口上采样到上升沿 0：条件匹配时，硬件停止无效 1：条件匹配时，硬件停止有效

Bit	Field	Type	Reset	Description
11	HSTP11	rw	0x00	硬件停止条件 11：TIMTRIB 端口上采样到下降沿 0：条件匹配时，硬件停止无效 1：条件匹配时，硬件停止有效
10	HSTP10	rw	0x00	硬件停止条件 10：TIMTRIB 端口上采样到上升沿 0：条件匹配时，硬件停止无效 1：条件匹配时，硬件停止有效
9	HSTP9	rw	0x00	硬件停止条件 9：TIMTRIA 端口上采样到下降沿 0：条件匹配时，硬件停止无效 1：条件匹配时，硬件停止有效
8	HSTP8	rw	0x00	硬件停止条件 8：TIMTRIA 端口上采样到上升沿 0：条件匹配时，硬件停止无效 1：条件匹配时，硬件停止有效
7	HSTP7	rw	0x00	硬件停止条件 7：CHxB 端口上采样到下降沿 0：条件匹配时，硬件停止无效 1：条件匹配时，硬件停止有效
6	HSTP6	rw	0x00	硬件停止条件 6：CHxB 端口上采样到上升沿 0：条件匹配时，硬件停止无效 1：条件匹配时，硬件停止有效
5	HSTP5	rw	0x00	硬件停止条件 5：CHxA 端口上采样到下降沿 0：条件匹配时，硬件停止无效 1：条件匹配时，硬件停止有效
4	HSTP4	rw	0x00	硬件停止条件 4：CHxA 端口上采样到上升沿 0：条件匹配时，硬件停止无效 1：条件匹配时，硬件停止有效
3	HSTP3	rw	0x00	硬件停止条件 3：从 AOS 来的事件触发 3 有效 0：条件匹配时，硬件停止无效 1：条件匹配时，硬件停止有效
2	HSTP2	rw	0x00	硬件停止条件 2：从 AOS 来的事件触发 2 有效 0：条件匹配时，硬件停止无效 1：条件匹配时，硬件停止有效
1	HSTP1	rw	0x00	硬件停止条件 1：从 AOS 来的事件触发 1 有效 0：条件匹配时，硬件停止无效 1：条件匹配时，硬件停止有效
0	HSTP0	rw	0x00	硬件停止条件 0：从 AOS 来的事件触发 0 有效 0：条件匹配时，硬件停止无效 1：条件匹配时，硬件停止有效

18.3.17 硬件清零事件选择寄存器 (TIMx_HCLRR)

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x07C

复位值：0x0000 0000

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

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

Bit	Field	Type	Reset	Description
31	CLEARS	rw	0x00	硬件清零使能 0：硬件清零无效 1：硬件清零有效 注：硬件清零有效时，软件清零的设定无效
30:16	Reserved			保留
15	HCEL15	rw	0x00	硬件清零条件 15：TIMTRID 端口上采样到下降沿 0：条件匹配时，硬件清零无效 1：条件匹配时，硬件清零有效
14	HCEL14	rw	0x00	硬件清零条件 14：TIMTRID 端口上采样到上升沿 0：条件匹配时，硬件清零无效 1：条件匹配时，硬件清零有效
13	HCEL13	rw	0x00	硬件清零条件 13：TIMTRIC 端口上采样到下降沿 0：条件匹配时，硬件清零无效 1：条件匹配时，硬件清零有效
12	HCEL12	rw	0x00	硬件清零条件 12：TIMTRIC 端口上采样到上升沿 0：条件匹配时，硬件清零无效 1：条件匹配时，硬件清零有效
11	HCEL11	rw	0x00	硬件清零条件 11：TIMTRIB 端口上采样到下降沿 0：条件匹配时，硬件清零无效 1：条件匹配时，硬件清零有效
10	HCEL10	rw	0x00	硬件清零条件 10：TIMTRIB 端口上采样到上升沿 0：条件匹配时，硬件清零无效 1：条件匹配时，硬件清零有效
9	HCEL9	rw	0x00	硬件清零条件 9：TIMTRIA 端口上采样到下降沿 0：条件匹配时，硬件清零无效 1：条件匹配时，硬件清零有效

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
8	HCEL8	rw	0x00	硬件清零条件 8：TIMTRIA 端口上采样到上升沿 0：条件匹配时，硬件清零无效 1：条件匹配时，硬件清零有效
7	HCEL7	rw	0x00	硬件清零条件 7：CHxB 端口上采样到下降沿 0：条件匹配时，硬件清零无效 1：条件匹配时，硬件清零有效
6	HCEL6	rw	0x00	硬件清零条件 6：CHxB 端口上采样到上升沿 0：条件匹配时，硬件清零无效 1：条件匹配时，硬件清零有效
5	HCEL5	rw	0x00	硬件清零条件 5：CHxA 端口上采样到下降沿 0：条件匹配时，硬件清零无效 1：条件匹配时，硬件清零有效
4	HCEL4	rw	0x00	硬件清零条件 4：CHxA 端口上采样到上升沿 0：条件匹配时，硬件清零无效 1：条件匹配时，硬件清零有效
3	HCEL3	rw	0x00	硬件清零条件 3：从 AOS 来的事件触发 3 有效 0：条件匹配时，硬件清零无效 1：条件匹配时，硬件清零有效
2	HCEL2	rw	0x00	硬件清零条件 2：从 AOS 来的事件触发 2 有效 0：条件匹配时，硬件清零无效 1：条件匹配时，硬件清零有效
1	HCEL1	rw	0x00	硬件清零条件 1：从 AOS 来的事件触发 1 有效 0：条件匹配时，硬件清零无效 1：条件匹配时，硬件清零有效
0	HCEL0	rw	0x00	硬件清零条件 0：从 AOS 来的事件触发 0 有效 0：条件匹配时，硬件清零无效 1：条件匹配时，硬件清零有效

18.3.18 硬件捕获 A 事件选择寄存器 (TIMx_HCPAR)

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x080

复位值：0x0000 0000

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15	HCPA15	rw	0x00	硬件捕获 A 条件 15：TIMTRID 端口上采样到下降沿 0：条件匹配时，硬件捕获 A 无效 1：条件匹配时，硬件捕获 A 有效
14	HCPA14	rw	0x00	硬件捕获 A 条件 14：TIMTRID 端口上采样到上升沿 0：条件匹配时，硬件捕获 A 无效 1：条件匹配时，硬件捕获 A 有效
13	HCPA13	rw	0x00	硬件捕获 A 条件 13：TIMTRIC 端口上采样到下降沿 0：条件匹配时，硬件捕获 A 无效 1：条件匹配时，硬件捕获 A 有效
12	HCPA12	rw	0x00	硬件捕获 A 条件 12：TIMTRIC 端口上采样到上升沿 0：条件匹配时，硬件捕获 A 无效 1：条件匹配时，硬件捕获 A 有效
11	HCPA11	rw	0x00	硬件捕获 A 条件 11：TIMTRIB 端口上采样到下降沿 0：条件匹配时，硬件捕获 A 无效 1：条件匹配时，硬件捕获 A 有效
10	HCPA10	rw	0x00	硬件捕获 A 条件 10：TIMTRIB 端口上采样到上升沿 0：条件匹配时，硬件捕获 A 无效 1：条件匹配时，硬件捕获 A 有效
9	HCPA9	rw	0x00	硬件捕获 A 条件 9：TIMTRIA 端口上采样到下降沿 0：条件匹配时，硬件捕获 A 无效 1：条件匹配时，硬件捕获 A 有效
8	HCPA8	rw	0x00	硬件捕获 A 条件 8：TIMTRIA 端口上采样到上升沿 0：条件匹配时，硬件捕获 A 无效 1：条件匹配时，硬件捕获 A 有效
7	HCPA7	rw	0x00	硬件捕获 A 条件 7：CHxB 端口上采样到下降沿 0：条件匹配时，硬件捕获 A 无效 1：条件匹配时，硬件捕获 A 有效
6	HCPA6	rw	0x00	硬件捕获 A 条件 6：CHxB 端口上采样到上升沿 0：条件匹配时，硬件捕获 A 无效 1：条件匹配时，硬件捕获 A 有效

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
5	HCPA5	rw	0x00	硬件捕获 A 条件 5：CHxA 端口上采样到下降沿 0：条件匹配时，硬件捕获 A 无效 1：条件匹配时，硬件捕获 A 有效
4	HCPA4	rw	0x00	硬件捕获 A 条件 4：CHxA 端口上采样到上升沿 0：条件匹配时，硬件捕获 A 无效 1：条件匹配时，硬件捕获 A 有效
3	HCPA3	rw	0x00	硬件捕获 A 条件 3：从 AOS 来的事件触发 3 有效 0：条件匹配时，硬件捕获 A 无效 1：条件匹配时，硬件捕获 A 有效
2	HCPA2	rw	0x00	硬件捕获 A 条件 2：从 AOS 来的事件触发 2 有效 0：条件匹配时，硬件捕获 A 无效 1：条件匹配时，硬件捕获 A 有效
1	HCPA1	rw	0x00	硬件捕获 A 条件 1：从 AOS 来的事件触发 1 有效 0：条件匹配时，硬件捕获 A 无效 1：条件匹配时，硬件捕获 A 有效
0	HCPA0	rw	0x00	硬件捕获 A 条件 0：从 AOS 来的事件触发 0 有效 0：条件匹配时，硬件捕获 A 无效 1：条件匹配时，硬件捕获 A 有效

18.3.19 硬件捕获 B 事件选择寄存器 (TIMx_HCPBR)

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x084

复位值：0x0000 0000

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15	HCPB15	rw	0x00	硬件捕获 B 条件 15：TIMTRID 端口上采样到下降沿 0：条件匹配时，硬件捕获 B 无效 1：条件匹配时，硬件捕获 B 有效

Bit	Field	Type	Reset	Description
14	HCPB14	rw	0x00	硬件捕获 B 条件 14：TIMTRID 端口上采样到上升沿 0：条件匹配时，硬件捕获 B 无效 1：条件匹配时，硬件捕获 B 有效
13	HCPB13	rw	0x00	硬件捕获 B 条件 13：TIMTRIC 端口上采样到下降沿 0：条件匹配时，硬件捕获 B 无效 1：条件匹配时，硬件捕获 B 有效
12	HCPB12	rw	0x00	硬件捕获 B 条件 12：TIMTRIC 端口上采样到上升沿 0：条件匹配时，硬件捕获 B 无效 1：条件匹配时，硬件捕获 B 有效
11	HCPB11	rw	0x00	硬件捕获 B 条件 11：TIMTRIB 端口上采样到下降沿 0：条件匹配时，硬件捕获 B 无效 1：条件匹配时，硬件捕获 B 有效
10	HCPB10	rw	0x00	硬件捕获 B 条件 10：TIMTRIB 端口上采样到上升沿 0：条件匹配时，硬件捕获 B 无效 1：条件匹配时，硬件捕获 B 有效
9	HCPB9	rw	0x00	硬件捕获 B 条件 9：TIMTRIA 端口上采样到下降沿 0：条件匹配时，硬件捕获 B 无效 1：条件匹配时，硬件捕获 B 有效
8	HCPB8	rw	0x00	硬件捕获 B 条件 8：TIMTRIA 端口上采样到上升沿 0：条件匹配时，硬件捕获 B 无效 1：条件匹配时，硬件捕获 B 有效
7	HCPB7	rw	0x00	硬件捕获 B 条件 7：CHxB 端口上采样到下降沿 0：条件匹配时，硬件捕获 B 无效 1：条件匹配时，硬件捕获 B 有效
6	HCPB6	rw	0x00	硬件捕获 B 条件 6：CHxB 端口上采样到上升沿 0：条件匹配时，硬件捕获 B 无效 1：条件匹配时，硬件捕获 B 有效
5	HCPB5	rw	0x00	硬件捕获 B 条件 5：CHxA 端口上采样到下降沿 0：条件匹配时，硬件捕获 B 无效 1：条件匹配时，硬件捕获 B 有效
4	HCPB4	rw	0x00	硬件捕获 B 条件 4：CHxA 端口上采样到上升沿 0：条件匹配时，硬件捕获 B 无效 1：条件匹配时，硬件捕获 B 有效
3	HCPB3	rw	0x00	硬件捕获 B 条件 3：从 AOS 来的事件触发 3 有效 0：条件匹配时，硬件捕获 B 无效 1：条件匹配时，硬件捕获 B 有效
2	HCPB2	rw	0x00	硬件捕获 B 条件 2：从 AOS 来的事件触发 2 有效 0：条件匹配时，硬件捕获 B 无效 1：条件匹配时，硬件捕获 B 有效

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
1	HCPB1	rw	0x00	硬件捕获 B 条件 1：从 AOS 来的事件触发 1 有效 0：条件匹配时，硬件捕获 B 无效 1：条件匹配时，硬件捕获 B 有效
0	HCPB0	rw	0x00	硬件捕获 B 条件 0：从 AOS 来的事件触发 0 有效 0：条件匹配时，硬件捕获 B 无效 1：条件匹配时，硬件捕获 B 有效

18.3.20 硬件递加事件选择寄存器 (TIMx_HCUPR)

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x088

复位值：0x0000 0000

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

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

Bit	Field	Type	Reset	Description
31:20	Reserved			保留
19	HCUP19	rw	0x00	硬件递加条件：从 AOS 来的事件触发 3 有效 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效
18	HCUP18	rw	0x00	硬件递加条件：从 AOS 来的事件触发 2 有效 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效
17	HCUP17	rw	0x00	硬件递加条件：从 AOS 来的事件触发 1 有效 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效
16	HCUP16	rw	0x00	硬件递加条件：从 AOS 来的事件触发 0 有效 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
15	HCUP15	rw	0x00	硬件递加条件：TIMTRID 端口上采样到下降沿 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效
14	HCUP14	rw	0x00	硬件递加条件：TIMTRID 端口上采样到上升沿 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效
13	HCUP13	rw	0x00	硬件递加条件：TIMTRIC 端口上采样到下降沿 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效
12	HCUP12	rw	0x00	硬件递加条件：TIMTRIC 端口上采样到上升沿 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效
11	HCUP11	rw	0x00	硬件递加条件：TIMTRIB 端口上采样到下降沿 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效
10	HCUP10	rw	0x00	硬件递加条件：TIMTRIB 端口上采样到上升沿 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效
9	HCUP9	rw	0x00	硬件递加条件：TIMTRIA 端口上采样到下降沿 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效
8	HCUP8	rw	0x00	硬件递加条件：TIMTRIA 端口上采样到上升沿 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效
7	HCUP7	rw	0x00	硬件递加条件：CHxB 端口为高电平时，CHxA 端口上采样到下降沿 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效
6	HCUP6	rw	0x00	硬件递加条件：CHxB 端口为高电平时，CHxA 端口上采样到上升沿 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效
5	HCUP5	rw	0x00	硬件递加条件：CHxB 端口为低电平时，CHxA 端口上采样到下降沿 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效
4	HCUP4	rw	0x00	硬件递加条件：CHxB 端口为低电平时，CHxA 端口上采样到上升沿 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
3	HCUP3	rw	0x00	硬件递加条件：CHxA 端口为高电平时，CHxB 端口上采样到下降沿 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效
2	HCUP2	rw	0x00	硬件递加条件：CHxA 端口为高电平时，CHxB 端口上采样到上升沿 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效
1	HCUP1	rw	0x00	硬件递加条件：CHxA 端口为低电平时，CHxB 端口上采样到下降沿 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效
0	HCUP0	rw	0x00	硬件递加条件：CHxA 端口为低电平时，CHxB 端口上采样到上升沿 0：条件匹配时，硬件递加无效 1：条件匹配时，硬件递加有效

18.3.21 硬件递减事件选择寄存器 (TIMx_HCDOR)

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x08C

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved												HCDO19	HCDO18	HCDO17	HCDO16
												RW	RW	RW	RW

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HCDO15	HCDO14	HCDO13	HCDO12	HCDO11	HCDO10	HCDO9	HCDO8	HCDO7	HCDO6	HCDO5	HCDO4	HCDO3	HCDO2	HCDO1	HCDO0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31:20	Reserved			保留
19	HCDO19	rw	0x00	硬件递减条件：从 AOS 来的事件触发 3 有效 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效

Bit	Field	Type	Reset	Description
18	HCDO18	rw	0x00	硬件递减条件：从 AOS 来的事件触发 2 有效 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效
17	HCDO17	rw	0x00	硬件递减条件：从 AOS 来的事件触发 1 有效 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效
16	HCDO16	rw	0x00	硬件递减条件：从 AOS 来的事件触发 0 有效 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效
15	HCDO15	rw	0x00	硬件递减条件：TIMTRID 端口上采样到下降沿 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效
14	HCDO14	rw	0x00	硬件递减条件：TIMTRID 端口上采样到上升沿 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效
13	HCDO13	rw	0x00	硬件递减条件：TIMTRIC 端口上采样到下降沿 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效
12	HCDO12	rw	0x00	硬件递减条件：TIMTRIC 端口上采样到上升沿 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效
11	HCDO11	rw	0x00	硬件递减条件：TIMTRIB 端口上采样到下降沿 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效
10	HCDO10	rw	0x00	硬件递减条件：TIMTRIB 端口上采样到上升沿 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效
9	HCDO9	rw	0x00	硬件递减条件：TIMTRIA 端口上采样到下降沿 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效
8	HCDO8	rw	0x00	硬件递减条件：TIMTRIA 端口上采样到上升沿 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效
7	HCDO7	rw	0x00	硬件递减条件：CHxB 端口为高电平时，CHxA 端口上采样到下降沿 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效
6	HCDO6	rw	0x00	硬件递减条件：CHxB 端口为高电平时，CHxA 端口上采样到上升沿 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效

Bit	Field	Type	Reset	Description
5	HCDO5	rw	0x00	硬件递减条件：CHxB 端口为低电平时，CHxA 端口上采样到下降沿 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效
4	HCDO4	rw	0x00	硬件递减条件：CHxB 端口为低电平时，CHxA 端口上采样到上升沿 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效
3	HCDO3	rw	0x00	硬件递减条件：CHxA 端口为高电平时，CHxB 端口上采样到下降沿 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效
2	HCDO2	rw	0x00	硬件递减条件：CHxA 端口为高电平时，CHxB 端口上采样到上升沿 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效
1	HCDO1	rw	0x00	硬件递减条件：CHxA 端口为低电平时，CHxB 端口上采样到下降沿 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效
0	HCDO0	rw	0x00	硬件递减条件：CHxA 端口为低电平时，CHxB 端口上采样到上升沿 0：条件匹配时，硬件递减无效 1：条件匹配时，硬件递减有效

18.3.22 软件同步启动寄存器（TIMx_SSTAR）

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x3F4

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													SSTA2	SSTA1	SSTA0
													RW	RW	RW

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31:3	Reserved			保留
2	SSTA2	rw	0x00	Timer6 软件启动 0：软件启动无效 1：软件启动使能
1	SSTA1	rw	0x00	Timer5 软件启动 0：软件启动无效 1：软件启动使能
0	SSTA0	rw	0x00	Timer4 软件启动 0：软件启动无效 1：软件启动使能

18.3.23 软件同步停止寄存器 (TIMx_SSTPR)

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x3F8

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													SSTP2	SSTP1	SSTP0
													RW	RW	RW

Bit	Field	Type	Reset	Description
31:3	Reserved			保留
2	SSTP2	rw	0x00	Timer6 软件停止 0：软件停止无效 1：软件停止使能
1	SSTP1	rw	0x00	Timer5 软件停止 0：软件停止无效 1：软件停止使能
0	SSTP0	rw	0x00	Timer4 软件停止 0：软件停止无效 1：软件停止使能

18.3.24 软件同步清零寄存器 (TIMx_SCLRR)

CH0 基地址: 0x40003000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x3FC

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													SCLR2	SCLR1	SCLR0
													RW	RW	RW

Bit	Field	Type	Reset	Description
31:3	Reserved			保留
2	SCLR2	rw	0x00	Timer6 软件清零 0：软件清零无效 1：软件清零使能
1	SCLR1	rw	0x00	Timer5 软件清零 0：软件清零无效 1：软件清零使能
0	SCLR0	rw	0x00	Timer4 软件清零 0：软件清零无效 1：软件清零使能

18.3.25 中断标志寄存器 (TIMx_IFR)

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x100

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													SCLR2	SCLR1	SCLR0
													RW	RW	RW

Bit	Field	Type	Reset	Description
31:16	Reserved			保留

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
15	SAMHF	rw	0x00	CHxA/B 端口高状态中断标志 0：CHxA 和 CHxB 端口上没有同时出现高电平 1：CHxA 和 CHxB 端口上同时出现高电平
14	SAMLF	rw	0x00	CHxA/B 端口低状态中断标志 0：CHxA 和 CHxB 端口上没有同时出现低电平 1：CHxA 和 CHxB 端口上同时出现低电平
13:9	Reserved			保留
8	DTEF	rw	0x00	死区时间错误中断标志 0：未发生死区时间错误；1：发生死区时间错误
7	UDFF	rw	0x00	下溢匹配中断标志 0：未发生锯齿波下溢或三角波计数到谷点 1：发生锯齿波下溢或三角波计数到谷点
6	OVFF	rw	0x00	上溢匹配中断标志 0：未发生锯齿波上溢或三角波计数到峰点 1：发生锯齿波上溢或三角波计数到峰点
5:4	Reserved			保留
3	CMDF	rw	0x00	计数匹配 D 中断标志 0：GCMDR 寄存器的值与计数值不相等 1：GCMDR 寄存器的值与计数值相等
2	CMCF	rw	0x00	计数匹配 C 中断标志 0：GCMCR 寄存器的值与计数值不相等 1：GCMCR 寄存器的值与计数值相等
1	CMBF	rw	0x00	计数匹配 B 中断标志 0：GCMBR 寄存器的值与计数值不相等，且未发生 CHxB 捕获完成动作 1：GCMBR 寄存器的值与计数值相等，或发生 CHxB 捕获完成动作
0	CMAF	rw	0x00	计数匹配 A 中断标志 0：GCMAR 寄存器的值与计数值不相等，且未发生 CHxA 捕获完成动作 1：GCMAR 寄存器的值与计数值相等，或发生 CHxA 捕获完成动作

18.3.26 中断清除寄存器（TIMx_ICLR）

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x104

复位值：0x0000 0000

31:16
Reserved

15	14	13:9	8	7	6	5:4	3	2	1	0
SAMHC	SAMLC	Reserved	DTEC	UDFC	OVFC	Reserved	CMDC	CMCC	CMBC	CMAC
R1W0	R1W0		R1W0	R1W0	R1W0		R1W0	R1W0	R1W0	R1W0

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15	SAMHC	rw	0x00	CHxA/B 端口高状态中断标志清除，写 1 无效，写 0 清除对应中断
14	SAMLC	rw	0x00	CHxA/B 端口低状态中断标志清除，写 1 无效，写 0 清除对应中断
13:9	Reserved			保留
8	DTEC	rw	0x00	死区时间错误中断标志清除，写 1 无效，写 0 清除对应中断
7	UDFC	rw	0x00	下溢匹配中断标志清除，写 1 无效，写 0 清除对应中断
6	OVFC	rw	0x00	上溢匹配中断标志清除，写 1 无效，写 0 清除对应中断
5:4	Reserved			保留
3	CMDC	rw	0x00	计数匹配 D 中断标志清除，写 1 无效，写 0 清除对应中断
2	CMCC	rw	0x00	计数匹配 C 中断标志清除，写 1 无效，写 0 清除对应中断
1	CMBC	rw	0x00	计数匹配 B 中断标志清除，写 1 无效，写 0 清除对应中断
0	CMAC	rw	0x00	计数匹配 A 中断标志清除，写 1 无效，写 0 清除对应中断

18.3.27 展频及中断触发选择寄存器（TIMx_CR）

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x108

复位值：0x0000 0300

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved									Dma_s_cmb	Dma_s_cma	Dma_g_udf	Dma_g_ovf	Reserved		Dma_g_cmd
									RW	RW	RW	RW			RW

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Dma_g_cmc	Dma_g_cmb	Dma_g_cma	CMSBE	CMSAE	DITENS	DITENB	DITNA	UDFE	OVFE	Reserved		CMDE	CMCE	CMBE	CMAE
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW			RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31:23	Reserved			保留
22	Dma_s_cmb	rw	0x00	专用比较 DMA 使能
21	Dma_s_cma	rw	0x00	专用比较 DMA 使能
20	Dma_g_udf	rw	0x00	下溢出 DMA 使能
19	Dma_g_ovf	rw	0x00	上溢出 DMA 使能
18:17	Reserved			保留
16	Dma_g_cmd	rw	0x00	通用比较 DMA 使能
15	Dma_g_cmc	rw	0x00	通用比较 DMA 使能
14	Dma_g_cmb	rw	0x00	通用比较 DMA 使能
13	Dma_g_cma	rw	0x00	通用比较 DMA 使能
12	CMSBE	rw	0x00	专用比较基准值匹配 B 使能触发 ADC
11	CMSAE	rw	0x00	专用比较基准值匹配 A 使能触发 ADC
10	DITENS	rw	0x00	PWM 展频计数选择 0: 选择下溢出，1：选择上溢出
9	DITENB	rw	0x00	PWM 通道 B 展频使能 0：使能无效，1：使能有效，每个周期改变 PWM 的输出延时
8	DITENA	rw	0x00	PWM 通道 A 展频使能 0：使能无效，1：使能有效，每个周期改变 PWM 的输出延时
7	UDFE	rw	0x00	下溢匹配使能触发 ADC 0：使能无效，1：使能有效，这个匹配可以控制 ADC/AOS_i_tirg
6	OVFE	rw	0x00	上溢匹配使能触发 ADC 0：使能无效，1：使能有效，这个匹配可以控制 ADC/AOS_i_tirg
5:4	Reserved			保留

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

18.3.28 AOS 选择寄存器, 三个通道共用 (TIMx_AOSSR)

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x110

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		SMH2	SMH1	SMH0	SML2	SML1	SML0	SOFTBK	Reserved		BFILTEN	BFILTS		FSAME	FBRAKE
		RW	RW	RW	RW	RW	RW	RW			RW	RW		R	R

Bit	Field	Type	Reset	Description
31:14	Reserved			保留
13	SMH2	rw	0x00	通道 2 同高选择 0：选择无效，1：选择有效，出现同高时 AOS_i_odis[1]
12	SMH1	rw	0x00	通道 1 同高选择 0：选择无效，1：选择有效，出现同高时 AOS_i_odis[1]
11	SMH0	rw	0x00	通道 0 同高选择 0：选择无效，1：选择有效，出现同高时 AOS_i_odis[1]
10	SML2	rw	0x00	通道 2 同低选择 0：选择无效，1：选择有效，出现同低时 AOS_i_odis[1]
9	SML1	rw	0x00	通道 1 同低选择 0：选择无效，1：选择有效，出现同低时 AOS_i_odis[1]
8	SML0	rw	0x00	通道 0 同低选择 0：选择无效，1：选择有效，出现同低时 AOS_i_odis[1]
7	SOFTBK	rw	0x00	软件刹车：写 1 实现软件刹车
6:5	Reserved			保留
4	BFILTEN	rw	0x00	端口刹车滤波使能
3:2	BFILTS	rw	0x00	端口刹车滤波时钟选择
1	FSAME	rw	0x00	同高同低刹车标志，只读
0	FBRAKE	rw	0x00	端口刹车标志，只读

18.3.29 AOS 刹车标志清除寄存器, 三个通道共用 (TIMx_AOSCL)

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x114

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

复位值：0x0000 0000

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

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

Bit	Field	Type	Reset	Description
31:2	Reserved			保留
1	FSAME	ro	0x00	同高同低刹车标志清除，写 0 清除，写 1 无效，读恒为 1
0	FBRAKE	ro	0x00	端口刹车标志清除，写 0 清除，写 1 无效，读恒为 1

18.3.30 端口刹车控制寄存器, 三个通道共用（TIMx_PTBKS）

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x118

复位值：0x0000 0000

Timer4/5/6 使用同一个实体寄存器，任意一个定时器其更改后，在另外两个定时器的值会同时更改。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EN15	EN14	EN13	EN12	EN11	EN10	EN9	EN8	EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15	EN15	rw	0x00	PD5 刹车端口使能：1 选择，0 无效
14	EN14	rw	0x00	PC5 刹车端口使能：1 选择，0 无效
13	EN13	rw	0x00	PB15 刹车端口使能：1 选择，0 无效
12	EN12	rw	0x00	PA15 刹车端口使能：1 选择，0 无效
11	EN11	rw	0x00	PD1 刹车端口使能：1 选择，0 无效
10	EN10	rw	0x00	PC11 刹车端口使能：1 选择，0 无效
9	EN9	rw	0x00	PB11 刹车端口使能：1 选择，0 无效
8	EN8	rw	0x00	PA11 刹车端口使能：1 选择，0 无效
7	EN7	rw	0x00	PD7 刹车端口使能：1 选择，0 无效

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
6	EN6	rw	0x00	PC7 刹车端口使能：1 选择，0 无效
5	EN5	rw	0x00	PB7 刹车端口使能：1 选择，0 无效
4	EN4	rw	0x00	PA7 刹车端口使能：1 选择，0 无效
3	EN3	rw	0x00	PD3 刹车端口使能：1 选择，0 无效
2	EN2	rw	0x00	PC3 刹车端口使能：1 选择，0 无效
1	EN1	rw	0x00	PB3 刹车端口使能：1 选择，0 无效
0	EN0	rw	0x00	PA3 刹车端口使能：1 选择，0 无效

18.3.31 端口触发控制寄存器, 三个通道共用 (TIMx_TTRIG)

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x11C

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRIGDS				TRIGCS				TRIGBS				TRIGAS			
RW				RW				RW				RW			

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:12	TRIGDS	rw	0x00	TIMx 触发 D 端口选择
11:8	TRIGCS	rw	0x00	TIMx 触发 C 端口选择
7:4	TRIGBS	rw	0x00	TIMx 触发 B 端口选择
3:0	TRIGAS	rw	0x00	TIMx 触发 A 端口选择

表 74: 控制信号与端口选择

0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
PA3	PB3	PC3	PD3	PA7	PB7	PC7	PD7	PA11	PB11	PC11	PD1	PA15	PB15	PC15	PD5

18.3.32 AOS 触发控制寄存器, 三个通道共用 (TIMx_ITRIG)

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

地址偏移：0x120

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IAOS3S				IAOS2S				IAOS1S				IAOS0S			
RW				RW				RW				RW			

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:12	IAOS3S	rw	0x00	TIMx AOS3 触发源选择
11:8	IAOS2S	rw	0x00	TIMx AOS2 触发源选择
7:4	IAOS1S	rw	0x00	TIMx AOS1 触发源选择
3:0	IAOS0S	rw	0x00	TIMx AOS0 触发源选择

表 75: 控制信号（IAOSxS）与中断源选择 (x=0,1,2,3)

0000	0001	0010	0011	0100	0101	0110	0111
TIM0_INT	TIM1_INT	TIM2_INT	LPTIMER_INT	TIM4_INTS	TIM5_INTS	TIM6_INTS	UART0_INT
1000	1001	1010	1011	1100	1101	1110	1111
UART1_INT	LPUART0_INT	VC0_INT	VC1_INT	RTC_INT	PCA_INT	SPI_INT	ADC_INT

18.3.33 端口刹车极性控制寄存器, 三个通道共用（TIMx_PTBKP）

CH0 基地址: 0x40003000

CH1 基地址: 0x40003400

CH2 基地址: 0x40003800

地址偏移：0x124

复位值：0x0000 0000

Timer4/5/6 使用同一个实体寄存器，任意一个定时器其更改后，在另外两个定时器的值会同时更改。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
POL15	POL14	POL13	POL12	POL11	POL10	POL9	POL8	POL7	POL6	POL5	POL4	POL3	POL2	POL1	POL0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15	POL15	rw	0x00	PD5 刹车端口极性选择：1 低电平有效，0 高电平有效
14	POL14	rw	0x00	PC5 刹车端口极性选择：1 低电平有效，0 高电平有效
13	POL13	rw	0x00	PB15 刹车端口极性选择：1 低电平有效，0 高电平有效
12	POL12	rw	0x00	PA15 刹车端口极性选择：1 低电平有效，0 高电平有效
11	POL11	rw	0x00	PD1 刹车端口极性选择：1 低电平有效，0 高电平有效
10	POL10	rw	0x00	PC11 刹车端口极性选择：1 低电平有效，0 高电平有效
9	POL9	rw	0x00	PB11 刹车端口极性选择：1 低电平有效，0 高电平有效
8	POL8	rw	0x00	PA11 刹车端口极性选择：1 低电平有效，0 高电平有效
7	POL7	rw	0x00	PD7 刹车端口极性选择：1 低电平有效，0 高电平有效
6	POL6	rw	0x00	PC7 刹车端口极性选择：1 低电平有效，0 高电平有效
5	POL5	rw	0x00	PB7 刹车端口极性选择：1 低电平有效，0 高电平有效
4	POL4	rw	0x00	PA7 刹车端口极性选择：1 低电平有效，0 高电平有效
3	POL3	rw	0x00	PD3 刹车端口极性选择：1 低电平有效，0 高电平有效
2	POL2	rw	0x00	PC3 刹车端口极性选择：1 低电平有效，0 高电平有效
1	POL1	rw	0x00	PB3 刹车端口极性选择：1 低电平有效，0 高电平有效
0	POL0	rw	0x00	PA3 刹车端口极性选择：1 低电平有效，0 高电平有效

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

19 实时时钟（RTC）

19.1 实时时钟简介

实时时钟/日历提供秒、分、时、日、周、月、年的信息，每月的天数和闰年的天数可自动调整。时钟操作可通过 AM/PM 寄存器位，决定采用 24 或 12 小时格式。表 76 所示是其基本特性。具有最小周期为 0.5S 的周期中断，如果需要更短的中断周期请使用 LPTIM 产生中断。

表 76: RTC 的基本特性

时钟源	片外低速晶振 XTL（32.768kHz） 片内低速振荡器 RCL(32kHz，1% 精度) 片外高速晶振 XTH
基本功能	可计算 00~99 年之间的秒、分、时、日、周、月、年
	可自动进行闰年调整
	可配置为 24 或 12 小时格式
	可程序控制启动或停止
	具有闹钟功能
	具有高精度 1Hz 方波信号输出
中断	具有最小周期为 0.5S 的周期中断
	具有精确到秒的闹钟中断

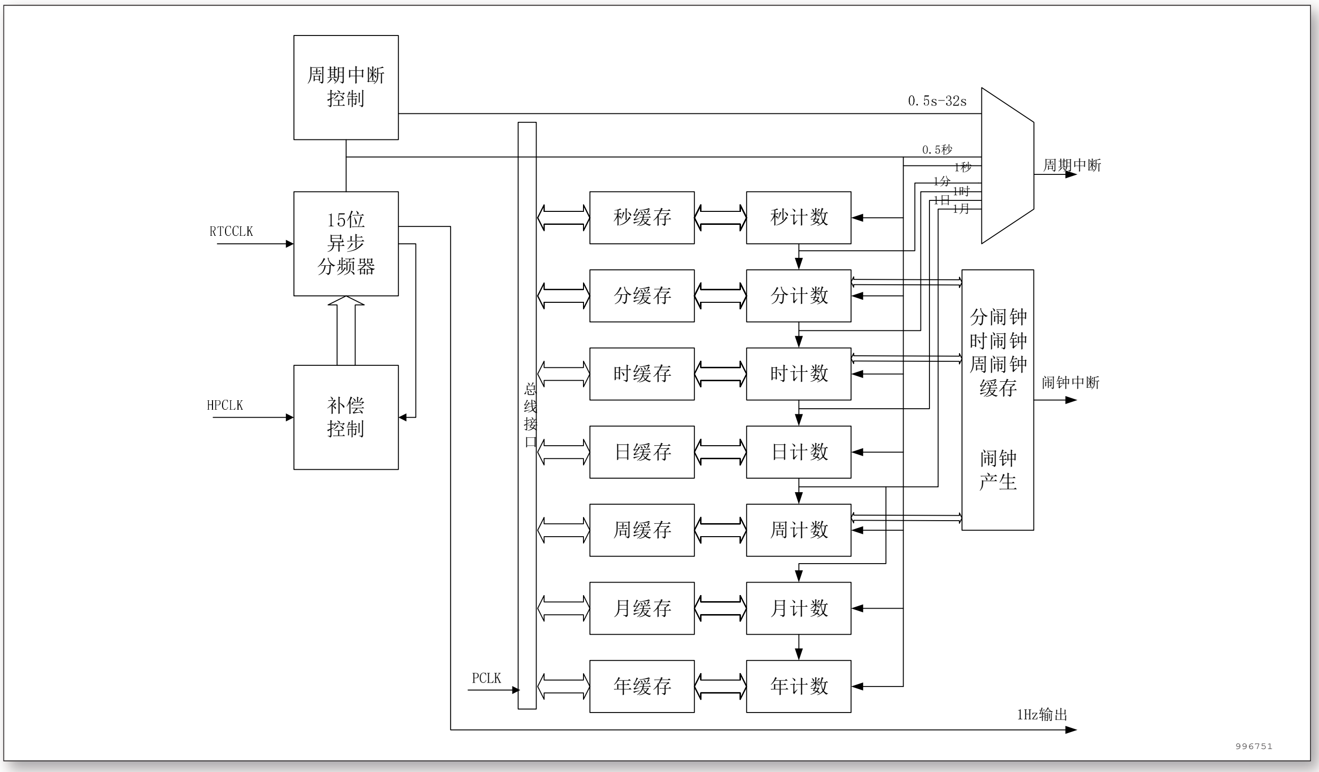

图 142. RTC 框图

19.2 实时时钟功能描述

实时时钟的时钟源可配置为外部低速晶振，外部高速晶振，内部低速 RC；默认使用外部低速晶振。控制寄存器 RTC_CR0、RTC_CR1 与 RTC_COMPEN 只受上电复位控制，其他复位源不能复位这三个控制寄存器。其他数据寄存器上电状态不定，上电后需要初始化，不受任何复位影响。

所有软件写入和读取的日期时间值都为 BCD 码，无须十六进制转换为十进制。

任何无效的日期时间将无法写入，比如 32 日，25 时，70 秒，B 月等。

19.2.1 上电设定

RTC 在上电之后复位一次，在系统不掉电的情况下，外部各种复位请求都不能复位 RTC，RTC 会一直处于计数状态。在上电之后，设定日历初始值、闹钟设置、误差补偿、中断等之后，启动 RTC。

19.2.2 RTC 计数开始设定

1. 设定 RTC_CR0.START=0，计数停止；
2. 设定 RTC_CR0.AMPM 和 RTC_CR0.PRDS，RTC_CR0.PRDX 设定时制和中断周期；
3. 设定 RTC_CR1.CKSEL 选择 RTC 的计时时钟；
4. 设定秒，分，时，周，日，月，年的日历计数寄存器；
5. 需要进行时钟误差补偿时，设定计数时钟误差补偿寄存器 RTC_COMPEN；
6. 清除中断标志位 RTC_CR1.ALMF，RTC_CR1.PRDF，并使能中断；
7. 设定 RTC_CR0.START=1，计数开始。

19.2.3 系统低功耗模式切换

在 RTC 计数开始后，系统如果立即切换为低功耗模式时，请执行下列任意一种确认后再进行模式切换。

控制寄存器在系统控制寄存器 SYSCTRL1.RTC_LPW。

在 RTC_CR0.START=1 设定后，经过 2 个以上的 RTC 计数时钟后再进行模式切换。

在 RTC_CR0.START=1 设定后，设定 RTC_CR1.WAIT=1，查询 RTC_CR1.WAITF=1。再设定 RTC_CR1.WAIT=0，查询 RTC_CR1.WAITF=0 后再进行模式切换。

在 RTC 低功耗模式下，RTC 寄存器不能读写。在低功耗模式下，RTC 消耗更少的电流。RTC 运行时切换低功耗模式不需要等待。

19.2.4 读出计数寄存器

有三种读取计数寄存器的方式：

方式 1：任意时刻式读取方式 1

1. 设定 RTC_CR1.WAIT=1，停止日历寄存器计数，进入读写模式；
2. 查询直到 RTC_CR1.WAITF=1；
3. 读出秒，分，时，周，日，月，年计数寄存器值；
4. 设定 RTC_CR1.WAIT=0，计数器计数；
5. 查询直到 RTC_CR1.WAITF=0。

方式 2：任意式时刻读取方式 2

1. 读出分，时，周，日，月，年计数寄存器值；

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

2. 读出秒计数寄存器值；
3. 再次读出秒计数寄存器值；
4. 判断两次秒的读出值是否相同，不同重新从第一步开始，相同读取结束。

方式 3：中断

在 RTC 周期中断服务中读取秒，分，时，周，日，月，年计数寄存器值。因为中断发生后到下次数据改变至少 0.5s 的时间。

19.2.5 写入计数寄存器

1. 设定 RTC_CR1.WAIT=1，停止日历寄存器计数，进入读写模式；
2. 查询直到 RTC_CR1.WAITF=1；
3. 写入秒，分，时，周，日，月，年计数寄存器值；
4. 设定 RTC_CR1.WAIT=0，计数器重新开始计数。注意：须在 1 秒内完成所有写操作；
5. 查询直到 RTC_CR1.WAITF=0。

在 RTC 未启动模式下写秒，分，时，周，日，月，年计数寄存器不需要等 WAIT。

注意：

-在计数模式下更改秒寄存器会复位秒计数，写分，时，周，日，月，年计数寄存器值不会影响 RTC 计数。

19.2.6 闹钟设定

1. 设定 RTC_CR1.ALMEN=0，闹钟禁止；
2. 设定 RTC_CR1.ALMIE=1，闹钟中断许可；
3. 秒闹钟 RTC_ ALMSEC，分闹钟 RTC_ALMMIN，时闹钟 RTC_ALMHOUR，周闹钟 RTC_ALMWEEK 设定；
4. 设定 RTC_CR1.ALMEN=1，闹钟许可；
5. 等待发生中断；
6. 由于闹钟中断和定周期中断共用中断请求信号，则当 RTC_CR1.ALMF=1 时，进入闹钟中断处理；否则进入定周期中断处理。

19.2.7 1Hz 输出

RTC 可选择输出一般精度，较高精度和高精度 3 种 1Hz 时钟。当时钟误差补偿功能有效时输出较高精度的 1Hz 时钟；当使用不同频率的 PCLK 时输出高精度的 1Hz 时钟。需要根据 PCLK 频率配置系统控制寄存器，其中：

一般精度的 1Hz 输出设定如下：（无时钟补偿）

1. 设定 RTC_CR0.START=0，计数停止；
2. RTC 输出引脚设定；
3. RTC_CR0.1HZOE=1，时钟输出许可；
4. 设定 RTC_CR0.START=1，计数开始；
5. 等待 2 个计数周期以上；
6. 1Hz 输出开始。

较高精度的 1Hz 输出设定如下：（低速补偿）

1. 设定 RTC_CR0.START=0，计数停止；

2. RTC 输出引脚设定；
3. RTC_CR0.1HZOE=1，时钟输出许可；
4. 时钟误差补偿寄存器 RTC_COMPEN.CR 补偿数设定；
5. 时钟误差补偿寄存器 RTC_COMPEN.EN=1，误差补偿有效；
6. 设定 RTC_CR0.START=1，计数开始；
7. 等待 2 个计数周期以上；
8. 1Hz 输出开始。

当高精度的 1Hz 输出时，需要在较高精度输出的基础上，为 RTC 提供 4M,6M,8M,12M,16M,20M, 24M,32MHz 的高速 PCLK 时钟，输出设定如下：

1. 设定 RTC_CR0.START=0，计数停止；
2. RTC 输出引脚设定；
3. RTC_CR0.1HZOE=1，时钟输出许可；
4. RTC_CR0.1HZSEL=1，选择输出高精度 1Hz 时钟；
5. 配置高速时钟补偿时钟 SYSCTRL1.RTC_FREQ_ADJUST；
6. 时钟误差补偿寄存器 RTC_COMPEN.CR[8:0] 补偿数设定；
7. 时钟误差补偿寄存器 RTC_COMPEN.EN=1，精度补偿有效；
8. 设定 RTC_CR0.START=1，计数开始；
9. 等待 2 个计数周期以上；
10. 1Hz 输出开始。

19.2.8 时钟误差补偿

由于外部晶振存在误差，在需要得到高精度的计数结果时，需要对该误差进行补偿。补偿方法分为两种：第一种，基于自身时钟的误差补偿；第二种，基于高速时钟的误差补偿。

基于自身时钟的误差补偿原理与计算：

由于计数器采用 32.768KHz 的时钟计数，如果需要对每秒精度进行补偿时，只能按照 32.768KHz 的整数周期补偿，则每秒补偿的最小单位为（1/32768）*10 6 =30.5ppm，无法满足高精度的要求。

那么要在 32.768KHz 的计数时钟下实现精度较高的时钟补偿时，需要在算法上做调整，将最大补偿周期扩大 32 倍。则在只能补偿的最小单位为 30.5ppm 的情况下，平均到每秒的补偿单位变为 30.5ppm/32=0.96ppm。满足了精度较高的时钟补偿要求。而且补偿发生在每 32 秒内比较均匀的范围内。所以，该寄存器中引入了 5 位小数的设定。

例 1：

当默认状态下直接输出 1Hz 时钟，通过测定该时钟的精度，计算补偿目标值。

假设实际测定值为 0.9999888Hz，则：

实际发振频率 = 32768 X 0.9999888 ≈ 2767.63

补偿目标值 = （实际发振频率-目标频率）/目标频率 X 10^6

= （32767.96 -32768）/32768 X 10^6

= -11.29ppm

根据

$$CR[8:0] = (\frac{A1X2^{15}}{10^6})_{A2} + 0001.00000B$$

如果补偿目标值为 -11.29ppm，计算相应的寄存器值如下：

$CR[8:0] = (-11.29X215/106)_{A2}$ + 0001.00000B

=$(-0.37)_{A2}$ + 0001.00000B

= 1111.10101B + 0001.00000B

= 0000.10101B

其中：

A1 为补偿目标值 [ppm]；

A2 为取 2 的补码.

基于高速 24MHz 时钟的误差补偿原理与计算：

该方式的计算方法与基于自身时钟的误差补偿相同。由于引入了 4M-32MHz 高速时钟，本来需要在最多 32 秒内累计的 1/32768 秒误差可分散到每 1 秒，针对每 1 秒进行最小 0.96ppm（23 个 24MHz 时钟周期）的补偿，实现平均的每秒高精度 1Hz 时钟输出。

19.3 RTC 中断

RTC 支持两种中断类型。闹钟中断、定周期中断。闹钟中断与定周期中断共用一个中断信号。

19.3.1 RTC 闹钟中断

当 RTC_CR1.ALMIE=1 时，若当前日历时间与秒闹钟寄存器（RTC_ALMSEC）、分闹钟寄存器（RTC_ALMMIN）、时闹钟寄存器（RTC_ALMHOUR）、周闹钟寄存器（RTC_ALMWEEK）相等时，触发闹钟中断。

19.3.2 RTC 周期中断

控制寄存器 1（RTC_CR1）的 ALMIE=1 时，选择的周期发生后，触发定周期唤醒中断，由于闹钟和定周期共用中断，通过标志寄存器位来区分。

19.4 RTC 寄存器描述

表 77. RTC 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0X000	RTC_CR0	控制寄存器 0	0x0000 0000	小节 19.4.1
0X004	RTC_CR1	控制寄存器 1	0x0000 0000	小节 19.4.2
0X008	RTC_SEC	秒计数寄存器	0xXXXX XXXX	小节 19.4.3
0X00C	RTC_MIN	分计数寄存器	0xXXXX XXXX	小节 19.4.4
0X010	RTC_HOUR	时计数寄存器	0xXXXX XXXX	小节 19.4.5
0X014	RTC_WEEK	周计数寄存器	0xXXXX XXXX	小节 19.4.7
0X018	RTC_DAY	日计数寄存器	0xXXXX XXXX	小节 19.4.6
0X01C	RTC_MON	月计数寄存器	0xXXXX XXXX	小节 19.4.8
0X020	RTC_YEAR	年计数寄存器	0xXXXX XXXX	小节 19.4.9
0X024	RTC_ALMMIN	分闹钟寄存器	0xXXXX XXXX	小节 19.4.11
0X028	RTC_ALMHOUR	时闹钟寄存器	0xXXXX XXXX	小节 19.4.12

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Offset	Acronym	Register Name	Reset	Section
0X02C	RTC_ALMWEEK	周闹钟寄存器	0xXXXX XXXX	小节 19.4.13
0X030	RTC_COMPEN	时钟误差补偿寄存器	0x0000 0020	小节 19.4.14
0X034	RTC_ALMSEC	秒闹钟寄存器	0xXXXX XXXX	小节 19.4.10

19.4.1 控制寄存器 0 (RTC_CR0)

起始地址：0x40001400

地址偏移：0X000

复位值：0x0000 0000 (只有上电对该寄存器复位有效)

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

15	14	13:8	7	6	5	4	3	2:0
Res.	PRDSEL	PRDX	START	HZ1SEL	HZ1OE	Res.	AMPM	PRDS
	R/W	R/W	R/W	R/W	R/W		R/W	R/W

Bit	Field	Type	Reset	Description
31:15	Reserved			保留
14	PRDSEL	rw	0x00	0：使用 PRDS 所设定的周期中断时间间隔 1：使用 PRDX 所设定的周期中断时间间隔
13:8	PRDX	rw	0x00	设置产生周期中断的时间间隔，可设定的范围为 0.5 秒到 32 秒，步进为 0.5 秒。 000000: 0.5 秒 000001: 1 秒 …… 111110: 31.5 秒 111111: 32 秒
7	START	rw	0x00	0：停止 RTC 计数器 1：使能 RTC 计数器
6	HZ1SEL	rw	0x00	0：普通精度 1Hz 输出 1：高精度 1Hz 输出
5	HZ1OE	rw	0x00	0：禁止 1Hz 输出 1：使能 1Hz 输出
4	Reserved			保留
3	AMPM	rw	0x00	0：12 小时制 1：24 小时制

Bit	Field	Type	Reset	Description
2:0	PRDS	rw	0x00	设置产生中断的时间间隔： 000：不产生周期中断 001：0.5 秒 010：1 秒 011：1 分钟 100：1 小时 101：1 天 11x：1 月 注意：如需要在 START=1 时写入更改周期中断的时间间隔操作步骤如下： step1，在 NVIC 中关闭 RTC 中断； step2，更改周期中断的时间间隔； step3，清除 RTC 中断标志； step4，使能 RTC 中断。

19.4.2 控制寄存器 1 (RTC_CR1)

起始地址：0x40001400

地址偏移：0X004

复位值：0x0000 0000 (只有上电对该寄存器复位有效)

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

15:11	10:8	7	6	5	4	3	2	1	0
Reserved	CKSEL	ALMEN	ALMIE	Res.	ALMF	PRDF	Res.	WAITF	WAIT
	R/W	R/W	R/W		R/W	R/W		R/W	R/W

Bit	Field	Type	Reset	Description
31:11	Reserved			保留
10:8	CKSEL	rw	0x00	RTC 时钟选择 00x: XTL 32.768k 01x: RCL 32k 100: XTH/128 (晶振为 4M 时选择此项) 101: XTH/256 (晶振为 8M 时选择此项) 110: XTH/512 (晶振为 16M 时选择此项) 111: XTH/1024 (晶振为 32M 时选择此项)

Bit	Field	Type	Reset	Description
7	ALMEN	rw	0x00	0：禁止闹钟 1：使能闹钟 注意：在 START=1 日历计数过程中并且 ALMIE=1 中断许可的情况下使能 ALMEN 时，为防止误动作请将系统中断关闭。使能后请将 ALMF 标志位清除。
6	ALMIE	rw	0x00	0：禁止闹钟中断 1：使能闹钟中断
5	Reserved			保留
4	ALMF	rw	0x00	0：未发生闹钟中断 1：已发生闹钟中断 注意：该位仅在 ALMEN=1 时有效。闹钟匹配时，32.7689KHz 一个时钟后置 1。写 0 时清除标志，写 1 无效。
3	PRDF	rw	0x00	0：未发生周期中断 1：已发生周期中断 注意：发生周期中断后，该位置 1。写 0 时清除该标志，写 1 无效。
2	Reserved			保留
1	WAITF	rw	0x00	0：非写入/读出状态 1：写入/读出状态 注意：WAIT 位设定是否有效标志。在写入/读出前请确认该位是否为 "1 "。计数过程中，在 WAIT 位清" 0 "后等待写入完成后该位才清" 0 "。
0	WAIT	rw	0x00	0：正常计数模式 1：写入/读出模式 注意：在写入/读出时请将该位置 "1"，由于计数器在连续计数，请在 1 秒的时间内完成写入/读出操作并将该位清 "0 "。

19.4.3 秒计数寄存器（RTC_SEC）

起始地址：0x40001400

地址偏移：0X008

复位值：0xXXXX XXXX

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									SECH			SECL			
									R/W			R/W			

Bit	Field	Type	Reset	Description
31:7	Reserved	rw	0xXX	保留
6:4	SECH	rw	0xXX	秒计数十位值
3:0	SECL	rw	0xXX	秒计数个位值

表示 0-59 秒，采用十进制计数。请写入十进制 0-59 的 BCD 码，写入错误值时，写入值将被忽略。

19.4.4 分计数寄存器 (RTC_MIN)

起始地址：0x40001400

地址偏移：0X00C

复位值：0xXXXX XXXX

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									MINH			MINL			
									R/W			R/W			

Bit	Field	Type	Reset	Description
31:7	Reserved			保留
6:4	MINH	rw	0xXX	分计数十位值
3:0	MINL	rw	0xXX	分计数个位值

表示 0-59 分，采用十进制计数。请写入十进制 0-59 的 BCD 码，写入错误值时，写入值将被忽略。

19.4.5 时计数寄存器 (RTC_HOUR)

起始地址：0x40001400

地址偏移：0X010

复位值：0xXXXX XXXX

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										HOURH		HOURL			
										R/W		R/W			

Bit	Field	Type	Reset	Description
31:6	Reserved			保留
5:4	HOURH	rw	0xXX	时计数十位值

Bit	Field	Type	Reset	Description
3:0	HOURL	rw	0xXX	时计数个位值

24 小时时制时，表示 0-23 小时。12 小时时制时，b5=0 表示 AM，则 01:12 表示上午；b5=1 表示 PM，则 21:32 表示下午。

请根据控制为 AMPM 的值，设定正确十进制的 0:23 或者 01:12,21:32 的 BCD 码。

写入超出范围的值将被忽略。

具体时间表示参考表 78：

表 78: 具体时间

24 小时时制	AMPM=1	12 小时时制	AMPM=0
时间	寄存器表示	时间	寄存器表示
00 时	00H	AM 12 时	12H
01 时	01H	AM 01 时	01H
02 时	02H	AM 02 时	02H
03 时	03H	AM 03 时	03H
04 时	04H	AM 04 时	04H
05 时	05H	AM 05 时	05H
06 时	06H	AM 06 时	06H
07 时	07H	AM 07 时	07H
08 时	08H	AM 08 时	08H
09 时	09H	AM 09 时	09H
10 时	10H	AM 10 时	10H
11 时	11H	AM 11 时	11H
12 时	12H	PM 12 时	32H
13 时	13H	PM 01 时	21H
14 时	14H	PM 02 时	22H
15 时	15H	PM 03 时	23H
16 时	16H	PM 04 时	24H
17 时	17H	PM 05 时	25H
18 时	18H	PM 06 时	26H
19 时	19H	PM 07 时	27H
20 时	20H	PM 08 时	28H
21 时	21H	PM 09 时	29H
22 时	22H	PM 10 时	30H
23 时	23H	PM 11 时	31H

19.4.6 日计数寄存器 (RTC_DAY)

起始地址：0x40001400

地址偏移：0X018

复位值：0xXXXX XXXX

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										DAYH		DAYL			
										R/W		R/W			

Bit	Field	Type	Reset	Description
31:6	Reserved			保留
5:4	DAYH	rw	0xXX	日计数十位值
3:0	DAYL	rw	0xXX	日计数个位值

十进制表示 1:31 日，自动计算闰年和月份。具体表示如表 79：

表 79: 自动计算闰年和月份

月份	日计数表示
2 月（普通年）	01:28
2 月（闰年）	01:29
4、6、9、11 月	01:30
1、3、5、7、8、10、12 月	01:31

19.4.7 周计数寄存器 (RTC_WEEK)

起始地址：0x40001400

地址偏移：0X014

复位值：0xXXXX XXXX

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													WEEK		
													R/W		

Bit	Field	Type	Reset	Description
31:3	Reserved			保留
2:0	WEEK	rw	0xXX	周计数值

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

十进制 0:6 表示周日: 周六。请写入正确的十进制 0:6 的 BCD 码，写入其他值，将被忽略。周计数值对应关系如表 80：

表 80: 周计数值

周	周计数表示
周日	00H
周一	01H
周二	02H
周三	03H
周四	04H
周五	05H
周六	06H

19.4.8 月计数寄存器 (RTC_MON)

起始地址：0x40001400

地址偏移：0X01C

复位值：0xXXXX XXXX

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
</table>

<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="11" rowspan="2">Reserved</td><td colspan="5">MON</td></tr>
<tr><td colspan="5">R/W</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:5	Reserved			保留
4:0	MON	rw	0xXX	月计数值

十进制 1:12 表示 1:12 月。请写入正确的十进制 1:12 的 BCD 码，写入其他值，将被忽略。

19.4.9 年计数寄存器 (RTC_YEAR)

起始地址：0x40001400

地址偏移：0X020

复位值：0xXXXX XXXX

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								YEARH				YEARL			
								R/W				R/W			

Bit	Field	Type	Reset	Description
31:8	Reserved			保留
7:4	YEARH	rw	0xXX	年计数十位值
3:0	YEARL	rw	0xXX	年个位计数值

十进制 0:99 表示 0:99 年。根据月进位计数。自动计算闰年如：00、04、08、…、92、96 等。请写入正确的十进制年计数值，写入错误值将被忽略。

19.4.10 秒闹钟寄存器 (RTC_ALMSEC)

起始地址：0x40001400

地址偏移：0X034

复位值：0xXXXX XXXX

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									ALMSECH			ALMSECL			
									R/W			R/W			

Bit	Field	Type	Reset	Description
31:7	Reserved			保留
6:4	ALMSECH	rw	0xXX	秒闹钟匹配值十位
3:0	ALMSECL	rw	0xXX	秒闹钟匹配值个位

请设定十进制 0:59 的 BCD 码。写入其他值，不会发生闹钟匹配。

19.4.11 分闹钟寄存器 (RTC_ALMMIN)

起始地址：0x40001400

地址偏移：0X024

复位值：0xXXXX XXXX

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									ALMMINH			ALMMINL			
									R/W			R/W			

Bit	Field	Type	Reset	Description
31:7	Reserved			保留
6:4	ALMMINH	rw	0xXX	分闹钟匹配值十位
3:0	ALMMINL	rw	0xXX	分闹钟匹配值个位

请设定十进制 0:59 的 BCD 码。写入其他值，不会发生闹钟匹配。

19.4.12 时闹钟寄存器 (RTC_ALMHOUR)

起始地址：0x40001400

地址偏移：0X028

复位值：0xXXXX XXXX

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										ALMHOURH		ALMHOURL			
										R/W		R/W			

Bit	Field	Type	Reset	Description
31:6	Reserved			保留
5:4	ALMHOURH	rw	0xXX	时闹钟十位匹配值
3:0	ALMHOURL	rw	0xXX	时闹钟个位匹配值

请根据时制设定正确的闹钟匹配值，否则不会发生时闹钟匹配。

19.4.13 周闹钟寄存器 (RTC_ALMWEEK)

起始地址：0x40001400

地址偏移：0X02C

复位值：0xXXXX XXXX

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									ALMWEEK						
									R/W						

Bit	Field	Type	Reset	Description
31:7	Reserved	rw	0xXX	保留
6:0	ALMWEEK	rw	0xXX	周闹钟匹配值。 b0:b6 分别对应周日: 周六，对应为置 “1 “时，代表每周该日闹钟有效。 如，b0=1，b5=1 代表周日和周五闹钟设定有效。

请根据时制设定正确的闹钟匹配值，否则不会发生时闹钟匹配。

19.4.14 时钟误差补偿寄存器 (RTC_COMPEN)

起始地址：0x40001400

地址偏移：0X030

复位值：0x0000 0020

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EN	Reserved						CR								
R/W							R/W								

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15	EN	rw	0x00	补偿使能 0：禁止时钟误差补偿 1：使能时钟误差补偿
14:9	Reserved			保留
8:0	CR	rw	0x20	补偿值 通过补偿值设定，可针对每秒进行 +/-0.96ppm 的精度补偿。补偿值为 9 位带小数点的 2 的补码，后 5 位为小数部分。可补偿范围 274.6ppm:212.6ppm。最小微分误差 +/-0.48ppm。最小分辨率 0.96ppm。具体补偿精度请参考表 81。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

表 81: 补偿精度

补偿值设定										补偿数
EN	CR[8:0]									
1	1	0	0	0	0	0	0	0	0	-274.6ppm
	1	0	0	0	0	0	0	0	1	-273.7ppm
	:	:	:	:	:	:	:	:	:	:
	0	0	0	0	1	1	1	1	1	-0.95ppm
	0	0	0	1	0	0	0	0	0	0ppm
	:	:	:	:	:	:	:	:	:	:
	0	1	1	1	1	1	1	1	0	+211.7ppm
	0	1	1	1	1	1	1	1	1	+212.6ppm
0	X	X	X	X	X	X	X	X	X	无补偿

补偿原理说明与计算：

由于计数器采用 32.768KHz 的时钟计数，如果需要对每秒精度进行补偿时，只能按照 32.768KHz 的整数周期补偿，则每秒补偿的最小单位为（1/32768）*10 6 =30.5ppm，无法满足高精度的要求。

那么要在 32.768KHz 的计数时钟下实现精度较高的时钟补偿时，需要在算法上做调整，将最大补偿周期扩大 32 倍。则在只能补偿的最小单位为 30.5ppm 的情况下，平均每秒的补偿单位变为为 30.5ppm/32=0.96ppm。满足了精度较高的时钟补偿要求。而且补偿发生在每 32 秒内比较均匀的范围内。所以，该寄存器中引入了 5 位小数的设定。

设定值计算如下：

$$CR[8:0] = (\frac{A1X2^{15}}{10^6})_{A2} + 0001.00000B$$

其中：

A1 为补偿目标值 [ppm]；

A2 为取 2 的补码。

如果补偿目标值为 +20.6ppm，计算相应的寄存器值如下：

$CR[8:0] = (20.3X2^{15}/10^6)_{A2} + 0001.00000B$

= $(0.6651904)_{A2}$ + 0001.00000B

= 0000.10101B + 0001.00000B

= 0001.10101B

如果补偿目标值为-20.6ppm，计算相应的寄存器值如下：

$CR[8:0] = (-20.3X2^{15}/10^6)_{A2} + 0001.00000B$

= $(-0.6651904)_{A2}$ + 0001.00000B

= 1111.01011B + 0001.00000B

= 0000.01011B

20 看门狗定时器（WDT）

20.1 WDT 简介

WDT 可用来检测和解决由软件错误引起的故障。当 WDT 计数器达到设定的溢出时间后，会触发中断或产生系统复位。WDT 由专用的 10KHz 片内振荡器驱动。

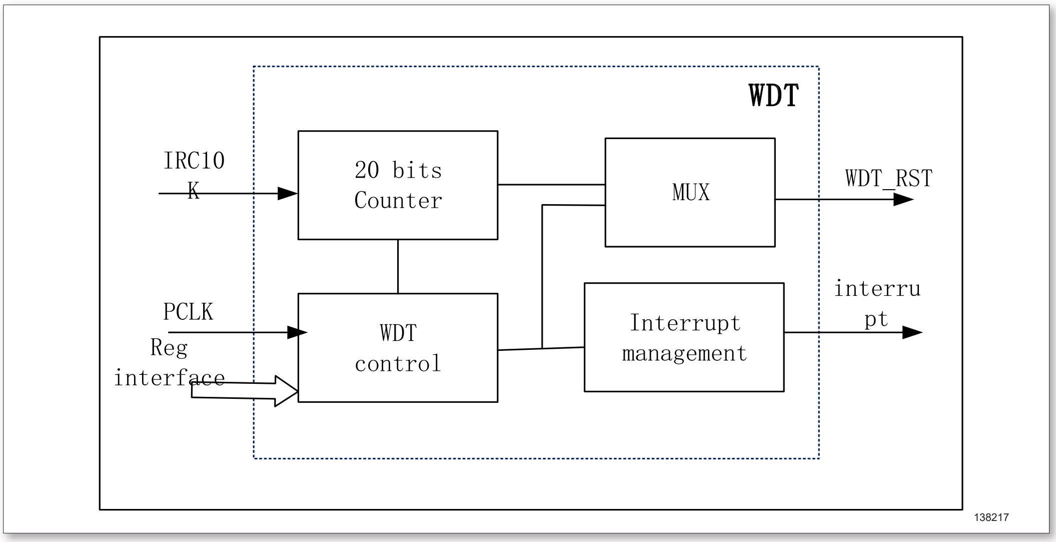

图 143. WDT 整体框图

20.2 WDT 功能描述

- 20Bit 自由运行的递增计数器，溢出时间可配置为 1.6ms ~ 50s。
- 溢出后的动作可配置为中断或复位。
- WDT 时钟由独立的 RC 振荡器提供，可在 Sleep 和 DeepSleep 模式下工作。
- WDTCON 寄存器只有在 WDT 未被启动时才能修改，以防止启动后无意之间修改 WDT 的配置。

20.2.1 WDT 溢出后产生中断

在本模式下，WDT 将按所设定的时间周期性地产生中断。在中断服务程序中需要清除 WDT 中断标志。

配置方法如下所示：

Step1：配置 WDT_CON. WOV，选择 WDT 计时溢出时间。

Step2：设置 WDT_CON. WINT_EN 为 1，选择 WDT 溢出后产生中断。

Step3：使能 NVIC 中断向量表中的 WDT 中断。

Step4：向 WDT_RST 寄存器依次写入 0x1E、0xE1，启动 WDT 定时器。

Step5：在中断服务程序中向 WDT_RST 寄存器依次写入 0x1E、0xE1 以清除中断标志。

20.2.2 WDT 溢出后产生复位

在本模式下，WDT 计数器溢出后会产生 Reset 信号，该信号会复位 MCU。用户程序需要在 WDT 溢出前清零 WDT 计数器，从而避免产生 WDT 复位。

配置方法如下所示：

Step1：配置 WDT_CON. WOV，选择 WDT 计数器溢出时间。

Step2：设置 WDT_CON. WINT_EN 为 0，选择 WDT 溢出后产生复位。

Step3：向 WDT_RST 寄存器依次写入 0x1E、0xE1，启动 WDT 定时器。

Step4：在 WDT 溢出前向 WDT_RST 寄存器依次写入 0x1E、0xE1 以清零 WDT 计数器。

注意：

-由于 WDT 振荡器是低精度的 RC 振荡器，强烈建议在 WDT 计数器计数值到达溢出值的一半之前对 WDT 进行清零。

20.3 WDT 寄存器描述

表 82. WDT 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0X080	WDT_RST	WDT 清除控制寄存器	0x0000 0000	小节 20.3.1
0X084	WDT_CON	WDT 控制寄存器	0x0000 000F	小节 20.3.2

20.3.1 WDT 清除控制寄存器 (WDT_RST)

起始地址：0x40000F00

地址偏移：0X080

复位值：0x0000 0000

<table>
<tr><td>31:8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td rowspan="2">Reserved</td><td colspan="8">WDTRST</td></tr>
<tr><td colspan="8">WO</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:8	Reserved			保留位，读为 0
7:0	WDTRST	wo	0x00	看门狗启动/清零控制 当看门狗未启动时，向该寄存器依次写入 0x1E、0xE1，启动 WDT 定时器。 当看门狗已启动时，向该寄存器依次写入 0x1E、0xE1，清零 WDT 定时器及中断标志。

20.3.2 WDT 控制寄存器 (WDT_CON)

起始地址：0x40000F00

地址偏移：0X084

复位值：0x0000 000F

<table>
<tr><td>31:16</td><td>15:8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td rowspan="2">Reserved</td><td>WCNTL</td><td>WDTINT</td><td rowspan="2">Res.</td><td>WINT_EN</td><td>WDTR</td><td colspan="4">WOV</td></tr>
<tr><td>RO</td><td>RO</td><td>RW</td><td>RO</td><td colspan="4">RW</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:16	Reserved			保留位，读为 0
15:8	WCNTL	ro	0x00	WDT 计数器低 8 位
7	WDTINT	ro	0x00	WDT 中断标志 1：已发生 WDT 中断，向 WDT_RST 寄存器依次写入 0x1E、0xE1 以清除该中断标志。 0：未发生 WDT 中断。
6	Reserved			保留位，读为 0
5	WINT_EN	rw	0x00	WDT 溢出后的动作配置 1：WDT 溢出后产生中断。 0：WDT 溢出后产生复位。
4	WDTR	ro	0x00	WDT 运行标志 1：WDT 正在运行 0：WDT 停止
3:0	WOV[3:0]	rw	0x0F	WDT 计时溢出时间配置 0000：1.6ms 1000：500ms 0001：3.2ms 1001：820ms 0010：6.4ms 1010：1.64s 0011：13ms 1011：3.28s 0100：26ms 1100：6.55s 0101：51ms 1101：13.1s 0110：102ms 1110：26.2s 0111：205ms 1111：52.4s

21 脉冲计数器（PCNT）

21.1 脉冲计数器简介

脉冲计数器（PCNT）模块可以对输入的脉冲进行计数，支持三种脉冲模式：单通道脉冲、双通道正交脉冲、双通道非交脉冲。无需软件参与，可在低功耗模式下正确计数。

21.2 脉冲计数器主要特性

脉冲计数器支持以下特性：

- 支持重载功能的 16 bit 计数器；
- 单通道脉冲计数；
- 双通道非交脉冲计数；
- 双通道正交脉冲计数，不失码；
- 加/减计数溢出中断；
- 脉冲超时中断；
- 4 种解码错误中断，非交脉冲模式；
- 1 种方向改变中断，正交脉冲模式；
- 多级脉冲宽度滤波；
- 输入脉冲极性可配置；
- 支持低功耗模式计数；
- 支持唤醒低功耗模式下 MCU；
- 支持任意脉冲沿间距不小于 1 个计数时钟周期；
- 支持 IO 输入或者 VC 输出作为脉冲输入信号
- 支持双路脉冲信号整形之后通过 IO 观测输出 PCNT_S0FO/PCNT_S1FO。

21.3 脉冲计数器功能描述

21.3.1 整体框图

脉冲计数器整体框图如图 144：

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

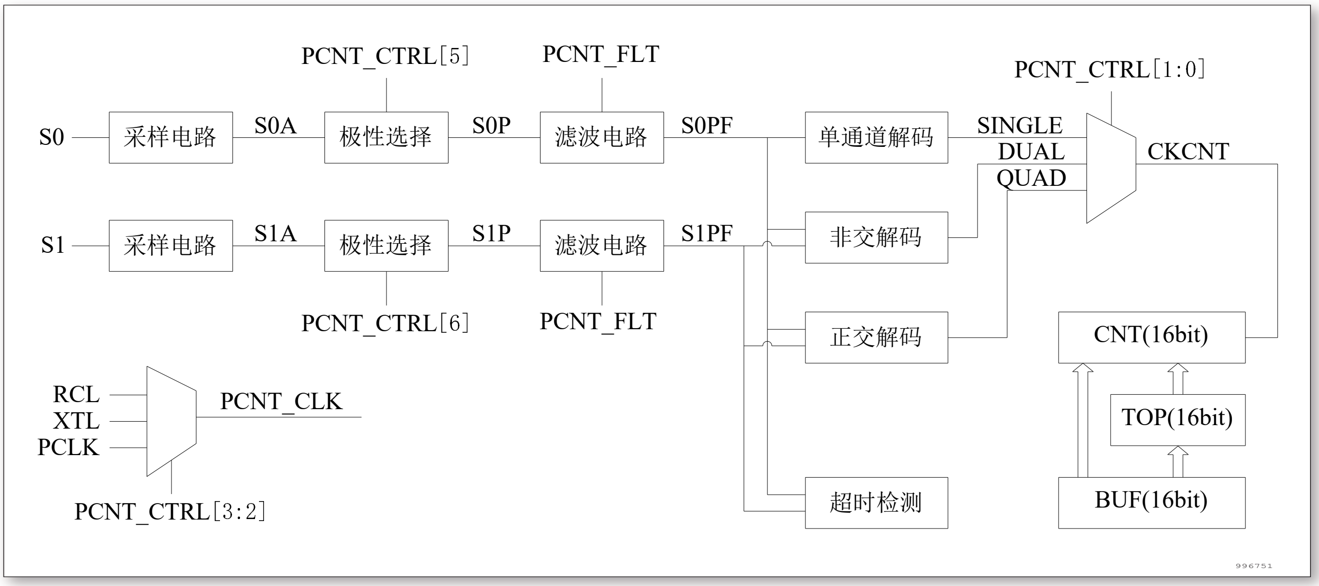

图 144. 整体框图

21.3.2 信号说明

以下以输入信号 S0 为例说明信号在计数器模块中的处理流程。

- S0 信号经采样模块后输出 S0A 信号，采样时钟为 PCNT_CLK。
- S0A 信号经极性选择模块后输出 S0P 信号。
- S0P 信号经脉冲宽度滤波模块后输出 S0PF 信号，滤波模块的时钟为分频后的 PCNT_CLK。
- S0PF 信号经单通道解码模块、双通道非交解码模块、双通道正交解码模块解码后输出相应的 SIGNAL、DUAL、QUAD 信号。
- SIGNAL、DUAL、QUAD 经多路选择器后输出 CKCNT 信号以驱动 CNT 模块进行计数。

21.3.3 计数模式

本模块支持 3 种计数模式：单通道脉冲、双通道正交脉冲、双通道非交脉冲。

单通道脉冲计数模式（Single Mode ）

当配置 PCNT_CTRL.Mode 为 0x00 或 0x01 时，脉冲计数器工作于单通道脉冲计数模式。该模式下解码模块只对 S0 脉冲信号进行计数。当 PCNT_CLK 时钟采样到 S0PF 信号的下降沿时，计数器按 PCNT_CTRL.DIR 的配置进行一次递增或递减操作。

计数器的计数范围为 0x00 至计数上阈值 (PCNT_TOP)。

在加计数状态，当计数值等于 PCNT_TOP 且采样到 S0PF 信号的下降沿时，计数器发生上溢出，计数值回到 0 并置位 PCNT_IFR.OF，该中断标志需要使用软件进行清零。图 145为 PCNT 加计数的时序图，图中 PCNT_TOP 的值为 99。

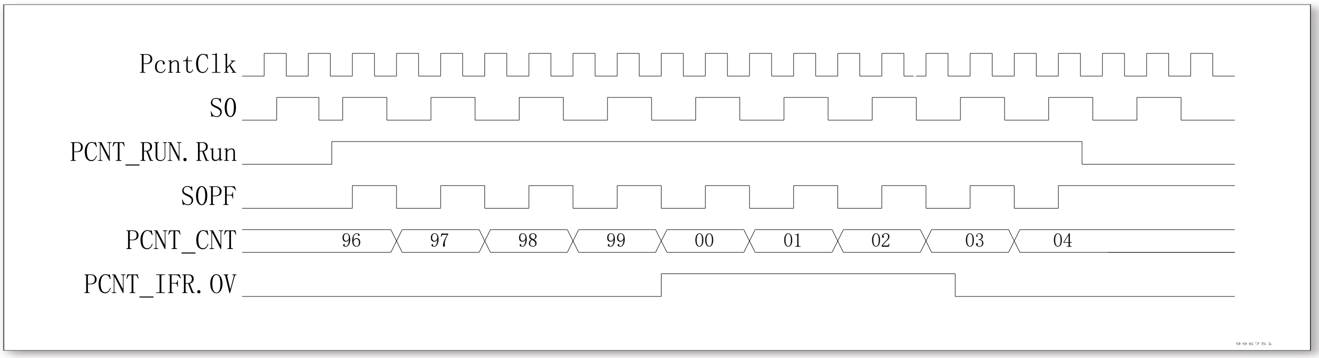

图 145. 单通道脉冲记数模式加计数波形

在减计数状态，当计数值等于 0x00 且采样到 S0PF 信号的下降沿时，计数器发生下溢出，计数值回到 PCNT_TOP 并置位 PCNT_IFR.UF，该中断标志需要使用软件进行清零。图 146为 PCNT 减计数的时序图，图中 PCNT_TOP 的值为 99。

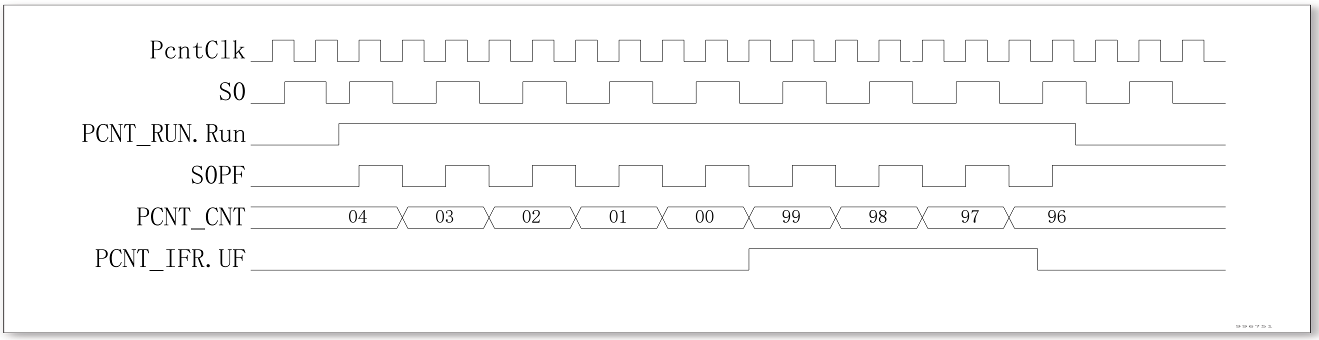

图 146. 单通道脉冲计数模式减计数波形

双通道非交脉冲计数模式（Dual Mode ）

当配置 PCNT_CTRL.Mode 为 0x02 时，脉冲计数器工作于双通道非交脉冲计数模式。该模式下解码模块对 S0 和 S1 脉冲信号进行解码计数。当 S0PF 和 S1PF 依次出现两个正脉冲，则计数器按 PCNT_CTRL.DIR 的配置进行一次递增或递减操作。

解码模块能正确计数的两种波形如图 147所示，S0PF 与 S1PF 没有同时为高的情况。

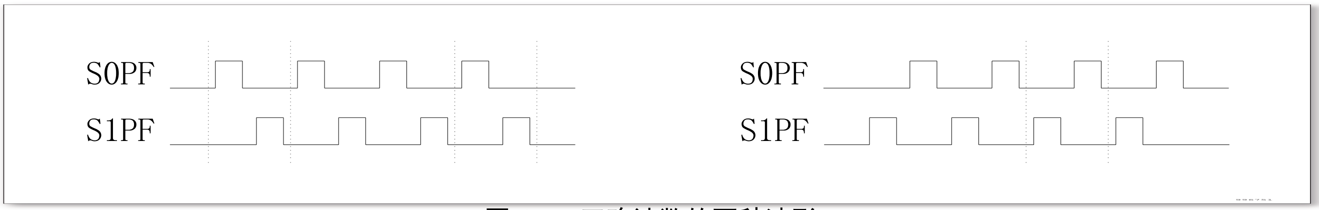

图 147. 正确计数的两种波形

解码模块不能正确计数的几种波形如图 148所示，S0PF 与 S1PF 出现同时为高的情况。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

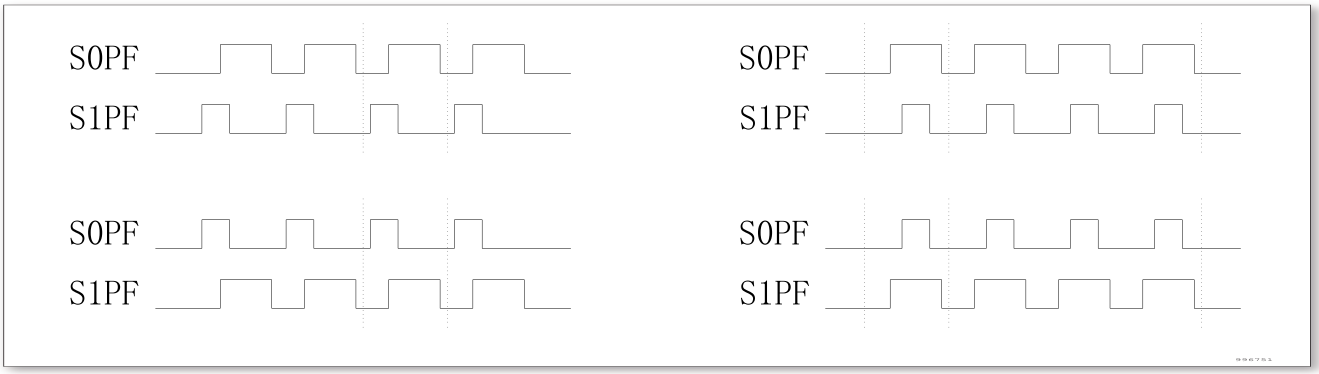

图 148. 不能正确计数的几种波形

计数器的计数范围为 0x00 至计数上阈值 (PCNT_TOP)。

在加计数状态，当计数值等于 PCNT_TOP 且 S0PF、S1PF 依次出现两个正脉冲时，计数器发生上溢出，计数值回到 0 并置位 PCNT_IFR.OF，该中断标志需要使用软件进行清零。图 149为 PCNT 加计数的时序图，图中 PCNT_TOP 的值为 99。

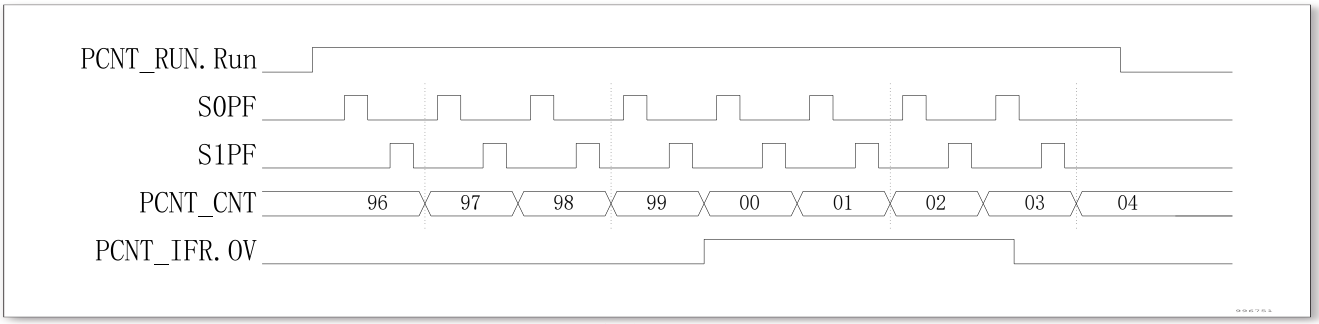

图 149. 双通道非交脉冲计数模式加计数波形

在减计数状态，当计数值等于 0x00 且 S0PF、S1PF 依次出现两个正脉冲时，计数器发生下溢出，计数值回到 PCNT_TOP 并置位 PCNT_IFR.UF，该中断标志需要使用软件进行清零。图 150为 PCNT 减计数的时序图，图中 PCNT_TOP 的值为 99。

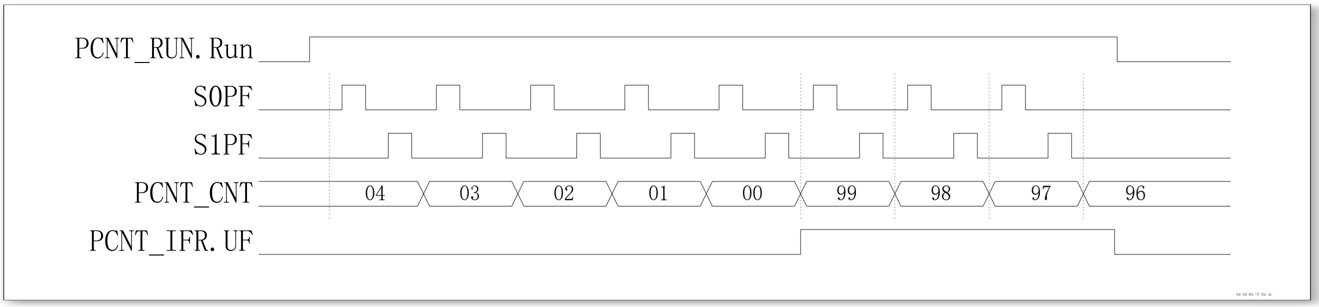

图 150. 双通道非交脉冲计数模式减计数波形

双通道正交脉冲记数模式（Quad Mode ）

当配置 PCNT_CTRL.Mode 为 0x03 时，脉冲计数器工作于双通道正交脉冲计数模式。

该模式下解码模块需要 S0 和 S1 两路脉冲信号进行解码计数。计数器根据 S0PF 和 S1PF 脉冲的相位关系自动判断计数的方向，物理设备任意正转、反转均能正确计数。当 S0PF 和 S1PF 完成一个周期的变化时，计数器按计数方向进行一次递增或递减操作。

解码模块判断计数方向的波形如图 151所示：

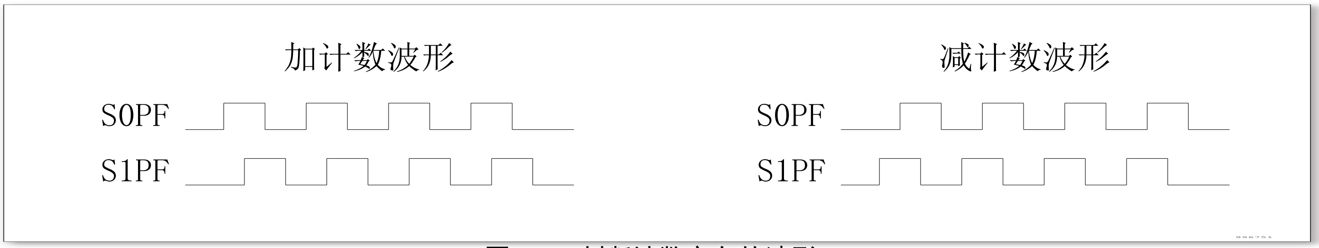

图 151. 判断计数方向的波形

计数器的计数范围为 0x00 至计数上阈值 (PCNT_TOP)。

在加计数状态，当计数值等于 PCNT_TOP 且 S0PF 和 S1PF 完成一个周期的变化时，计数器发生上溢出，计数值回到 0 并置位 PCNT_IFR.OF，该中断标志需要使用软件进行清零。图 152为 PCNT 加计数的时序图，图中 PCNT_TOP 的值为 99。

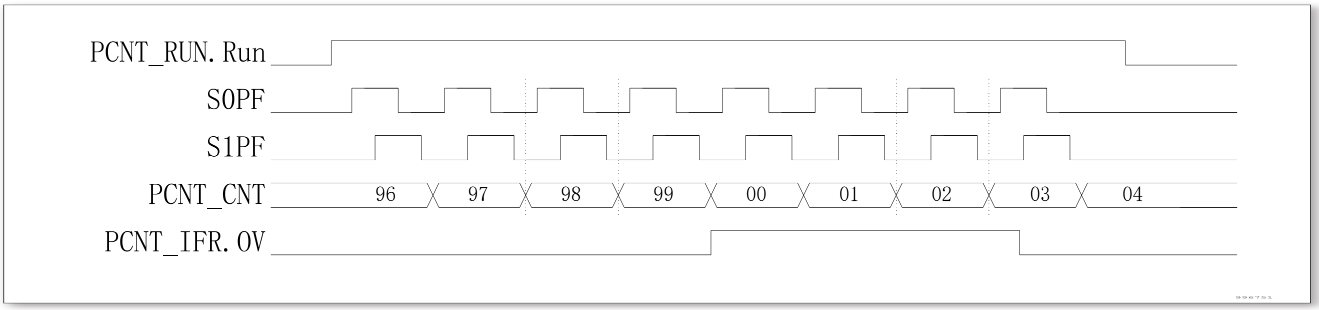

图 152. 双通道正交脉冲计数模式加计数波形

在减计数状态，当计数值等于 0x00 且 S0PF 和 S1PF 完成一个周期的变化时，计数器发生下溢出，计数值回到 PCNT_TOP 并置位 PCNT_IFR.UF，该中断标志需要使用软件进行清零。图 153为 PCNT 减计数的时序图，图中 PCNT_TOP 的值为 99。

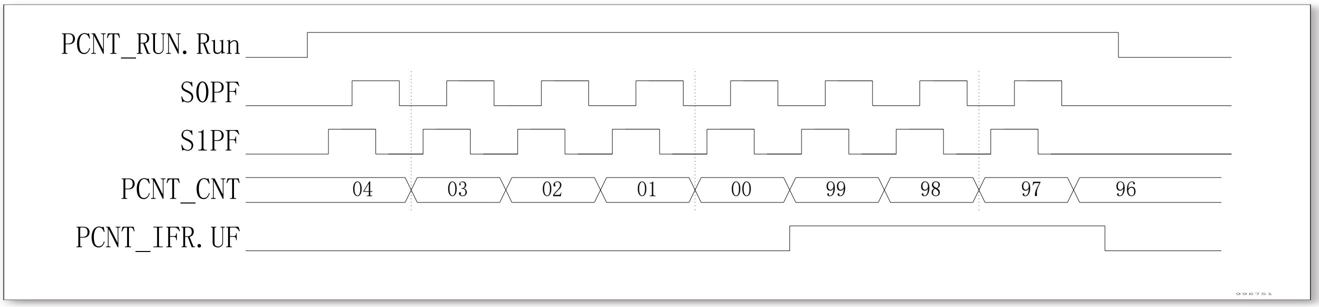

图 153. 双通道正交脉冲计数模式减计数波形

21.3.4 脉冲宽度滤波

该脉冲宽度滤波器使用计数时钟分频作为滤波时钟，并以此计数从而实现脉冲去抖动目的。其中滤波时钟的分频系数从 1~8096 倍配置寄存器可选（FLT.ClkDiv），保持电平超过（FLT.DebTop）（1~7）个滤波时钟视为非抖动，可通过滤波器。

该滤波功能可由配置寄存器（FLT.EN）打开或关闭。

同时仅当启动 PCNT 之后（Run=1），滤波计数器才启动。

21.3.5 超时

打开超时功能（TOCR.EN）将使用滤波时钟对脉冲高电平时间计数，采样到低电平将使计数器回到 0，若一个高电平计数值达到（TOCR.TH）将产生中断标志位 IF.TO。同时仅当启动 PCNT 之后（Run=1），超时计数器

才启动。

21.3.6 脉冲输入选择

相较于 HC32L130 / HC32L136 系列在 IO_MUX 中选择 IO 的输入作为 PCNT 的脉冲输入，在本系列中增加了 VC 的输出作为 PCNT 的脉冲输入，通过 IO_CTRL7 寄存器来进行选择，详见 IO_CTRL 模块的寄存器描述。

21.3.7 Debug 观测输出

为确认进入脉冲解码模块的信号符合预期,可以通过寄存器选择三组整形后的脉冲通过 IO 输出作为 PCNT_S0FO、PCNT_S1FO 观测信号，分别是：

- S0A/S1A
- S0P/S1P
- S0PF/S1PF

通过观测 PCNT_S0FO/PCNT_S1FO 输出脉冲信号，可以更直观的调整极性选择和脉冲宽度滤波器的配置，避免解码电路的输入信号不符合预期。

21.4 PCNT 寄存器描述

表 83. PCNT 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x00	PCNT_RUN	PCNT 启动寄存器	0x0000 0000	小节 21.4.1
0x04	PCNT_CTRL	PCNT 控制寄存器	0x0000 0000	小节 21.4.2
0x08	PCNT_FLT	PCNT 滤波控制寄存器	0x0000 2000	小节 21.4.3
0x0c	PCNT_TOCR	PCNT 超时控制寄存器	0x0000 0FFF	小节 21.4.4
0x10	PCNT_CMD	PCNT 命令寄存器	0x0000 0000	小节 21.4.5
0x14	PCNT_SR1	PCNT 状态寄存器 1	0x0000 0000	小节 21.4.6
0x18	PCNT_CNT	PCNT 计数寄存器	0x0000 0000	小节 21.4.7
0x1c	PCNT_TOP	PCNT 计数溢出寄存器	0x0000 00FF	小节 21.4.8
0x20	PCNT_BUF	PCNT 计数溢出缓存寄存器	0x0000 00FF	小节 21.4.9
0x24	PCNT_IFR	PCNT 中断标识寄存器	0x0000 0000	小节 21.4.10
0x28	PCNT_ICR	PCNT 中断清除寄存器	0x0000 00FF	小节 21.4.11
0x2c	PCNT_IEN	PCNT 中断使能寄存器	0x0000 0000	小节 21.4.12
0x30	PCNT_SR2	PCNT 状态寄存器 2	0x0000 0000	小节 21.4.13
0x34	PCNT_DBG	PCNT PCNT_S0FO/PCNT_S1FO 输出选择寄存器	0x0000 0000	小节 21.4.14

21.4.1 PCNT 启动寄存器 (PCNT_RUN)

起始地址：0x40005400

地址偏移：0x00

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

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

Bit	Field	Type	Reset	Description
31:1	Reserved			保留
0	Run	rw	0x00	PCNT 启动/停止控制。 0: 停止 1: 启动

注意：

-当前模块将 pclk 时钟域的配置寄存器视作长信号，未进行到 pcnt_clk 时钟域的同步，所以当 Run=1 启动 PCNT 后，配置寄存器不允许进行修改以避免发生未知错误。所有配置寄存器的修改务必在 PCNT 停止的状态（run=0）下进行。

-对 Run 位写入一个值后，需要经过一个同步的过程才可启动/停止 PCNT 计数，在同步未完成之前读取 Run 位将返回写入之前的 Run 状态值。所以如果要停止 PCNT 之后再启动 PCNT，需对 Run 位写入 0，查询到 Run 位 =0 之后才可继续后续其他操作（包括对 Run 位写入 1 以重新启动 PCNT）。

21.4.2 PCNT 控制寄存器 (PCNT_CTRL)

起始地址：0x40005400

地址偏移：0x04

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									S1P	S0P	DIR	ClkSel		Mode	
									rw	rw	rw	rw		rw	

Bit	Field	Type	Reset	Description
31:7	Reserved			保留
6	S1P	rw	0x00	S1 通道输入极性选择 0: 不取反 1: 取反

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
5	S0P	rw	0x00	S0 通道输入极性选择 0: 不取反 1: 取反
4	DIR	rw	0x00	计数方向选择 0: 加计数 1: 减计数 注：仅对单通道模式和双通道非交模式有效。
3:2	ClkSel	rw	0x00	PCNT_CLK 计数时钟选择 0：PCLK 1：PCLK 2：XTL 3：RCL
1:0	Mode	rw	0x00	脉冲计数模式选择 0: 单通道脉冲计数模式 1: 单通道脉冲计数模式 2: 双通道非交脉冲计数模式 3: 双通道正交脉冲计数模式

21.4.3 PCNT 滤波控制寄存器（PCNT_FLT）

起始地址：0x40005400

地址偏移：0x08

复位值：0x0000 2000

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="15" rowspan="2">Reserved</td><td>EN</td></tr>
<tr><td>rw</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="3">DebTop</td><td colspan="13">ClkDiv</td></tr>
<tr><td colspan="3">rw</td><td colspan="13">rw</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:17	Reserved			保留
16	EN	rw	0x00	脉冲宽度滤波器使能控制。 0: 不使能 1: 使能

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
15:13	DebTop	rw	0x01	计数器阈值 使用滤波时钟对输入脉冲连续采样，当连续电平采样计数值达到阈值则视为正常脉冲，可通过滤波器。 0： 为非法值
12:0	ClkDiv	rw	0x00	滤波时钟分频系数 系数 = ClkDiv + 1

21.4.4 PCNT 超时控制寄存器 (PCNT_TOCR)

起始地址：0x40005400

地址偏移：0x0c

复位值：0x0000 0FFF

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

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

Bit	Field	Type	Reset	Description
31:17	Reserved	rw	0x00	保留
16	EN	rw	0x00	超时功能使能控制 0: 不使能 1: 使能
15:12	Reserved			保留
11:0	TH	rw	0x00	超时阈值 当脉冲高电平被滤波时钟连续采样计数达到阈值，产生超时中断标识，计数器重新开始计数。

21.4.5 PCNT 命令寄存器 (PCNT_CMD)

起始地址：0x40005400

地址偏移：0x10

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved													B2C	B2T	T2C
													wo1	wo1	wo1

Bit	Field	Type	Reset	Description
31:3	Reserved			保留
2	B2C	wo1	0x00	写 1，立即将 BUF 中的值同步到 CNT。 同步过程中 PCNT_SR2 的相应位为 1，此时不应对 BUF 进行写操作，不应对 CNT 进行读操作。
1	B2T	wo1	0x00	写 1，立即将 BUF 中的值同步到 TOP。 同步过程中 PCNT_SR2 的相应位为 1，此时不应对 BUF 进行写操作，不应对 TOP 进行读操作。
0	T2C	wo1	0x00	写 1，立即将 TOP 中的值同步到 CNT。 同步过程中 PCNT_SR2 的相应位为 1，此时不应对 B2T 写 1，不应对 CNT 进行读操作。

21.4.6 PCNT 状态寄存器 1 (PCNT_SR1)

起始地址：0x40005400

地址偏移：0x14

复位值：0x0000 0000

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

Bit	Field	Type	Reset	Description
31:1	Reserved			保留
0	DIR	ro	0x00	双通道正交脉冲计数方向指示 0: 加计数 1: 减计数

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

21.4.7 PCNT 计数寄存器（PCNT_CNT）

起始地址：0x40005400

地址偏移：0x18

复位值：0x0000 0000

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	CNT	ro	0x00	计数器计数值

21.4.8 PCNT 计数溢出寄存器（PCNT_TOP）

起始地址：0x40005400

地址偏移：0x1c

复位值：0x0000 00FF

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	TOP	ro	0xFF	计数器上溢出阈值 计数器下溢出重载值

21.4.9 PCNT 计数溢出缓存寄存器（PCNT_BUF）

起始地址：0x40005400

地址偏移：0x20

复位值：0x0000 00FF

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	BUF	rw	0xFF	CNT 寄存器、TOP 寄存器的缓存

21.4.10 PCNT 中断标识寄存器 (PCNT_IFR)

起始地址：0x40005400

地址偏移：0x24

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								S1E	S0E	BB	FE	DIR	TO	OV	UF
								ro	ro	ro	ro	ro	ro	ro	ro

Bit	Field	Type	Reset	Description
31:8	Reserved			保留
7	S1E	ro	0x00	脉冲解码错误中断标识。 在一个非交编码完整采样周期内，S1 通道未发生变化。 仅在双通道非交编码模式有效。
6	S0E	ro	0x00	脉冲解码错误中断标识。 在一个非交编码完整采样周期内，S0 通道未发生变化。 仅在双通道非交编码模式有效。
5	BB	ro	0x00	脉冲解码错误中断标识。 双通道脉冲分别背靠背采样为高，中间并无低状态。 仅在双通道非交编码模式有效。
4	FE	ro	0x00	脉冲解码错误中断标识。 在一个非交编码完整采样周期内，采样到的不是一个正确的非交编码帧。 仅在双通道非交编码模式有效。

Bit	Field	Type	Reset	Description
3	DIR	ro	0x00	正交脉冲方向改变中断标识。 正交脉冲被解码器判定为计数方向发生改变。 仅在双通道正交脉冲有效。
2	TO	ro	0x00	超时中断标识。 在 3 种模式均有效。
1	OV	ro	0x00	上溢出中断标识。 在 3 种模式均有效。
0	UF	ro	0x00	下溢出中断标识。 在 3 种模式均有效。

21.4.11 PCNT 中断清除寄存器 (PCNT_ICR)

起始地址：0x40005400

地址偏移：0x28

复位值：0x0000 00FF

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								S1E	S0E	BB	FE	DIR	TO	OV	UF
								wo0	wo0	wo0	wo0	wo0	wo0	wo0	wo0

对每个 bit 写 0 可对 小节 21.4.10中 PCNT_IFR 各对应中断标识清除。

每 bit 详细解释参见 小节 21.4.10。

21.4.12 PCNT 中断使能寄存器 (PCNT_IEN)

起始地址：0x40005400

地址偏移：0x2c

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								S1E	S0E	BB	FE	DIR	TO	OV	UF
								rw	rw	rw	rw	rw	rw	rw	rw

对每个 bit 对应小节 21.4.10中 PCNT_IFR 各对应中断的输出使能位。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

1：使能。 0：不使能。

每 bit 详细解释参见小节 21.4.10。

21.4.13 PCNT 状态寄存器 2 (PCNT_SR2)

起始地址：0x40005400

地址偏移：0x30

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													B2C	B2T	T2C
													ro	ro	ro

Bit	Field	Type	Reset	Description
31:3	Reserved			保留
2	B2C	ro	0x00	将 BUF 中数值同步到 CNT 时的同步状态位。 1：正在进行同步，不可读取 CNT 寄存器 0：同步已完成
1	B2T	ro	0x00	将 BUF 中数值同步到 TOP 时的同步状态位。 1：正在进行同步，不可读取 TOP 寄存器 0：同步已完成
0	T2C	ro	0x00	将 TOP 中数值同步到 CTN 时的同步状态位。 1：正在进行同步，不可读取 CNT 寄存器 0：同步已完成

21.4.14 PCNT PCNT_S0FO/PCNT_S1FO 输出选择寄存器 (PCNT_DBG)

起始地址：0x40005400

地址偏移：0x34

复位值：0x0000 0000

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

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

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31:2	Reserved			保留
1:0	DBG	rw	0x00	选择脉冲 PCNT_S0FO/PCNT_S1FO 观测输出的来源。 0：固定为 0 1：脉冲同步之后的输出（S0A/S1A） 2：脉冲极性选择之后的输出（S0P/S1P） 3：脉冲滤波之后的输出（S0PF/S1PF）

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

22 通用同步异步收发器 (UART)

22.1 概述

通用 UART 模块（UART0/1/2/3）支持以下基本功能：

- 半双工和全双工传输
- 单线半双工传输
- 8/9-Bit 传输数据长度
- 硬件奇偶校验
- 1/1.5/2-Bit 停止位
- 四种不同传输模式
- 16-Bit 波特率计数器
- 多机通讯
- 硬件地址识别
- DMAC 硬件传输握手
- 硬件流控
- 支持单线模式

22.2 功能框图

图 154为 UART 结构框图。

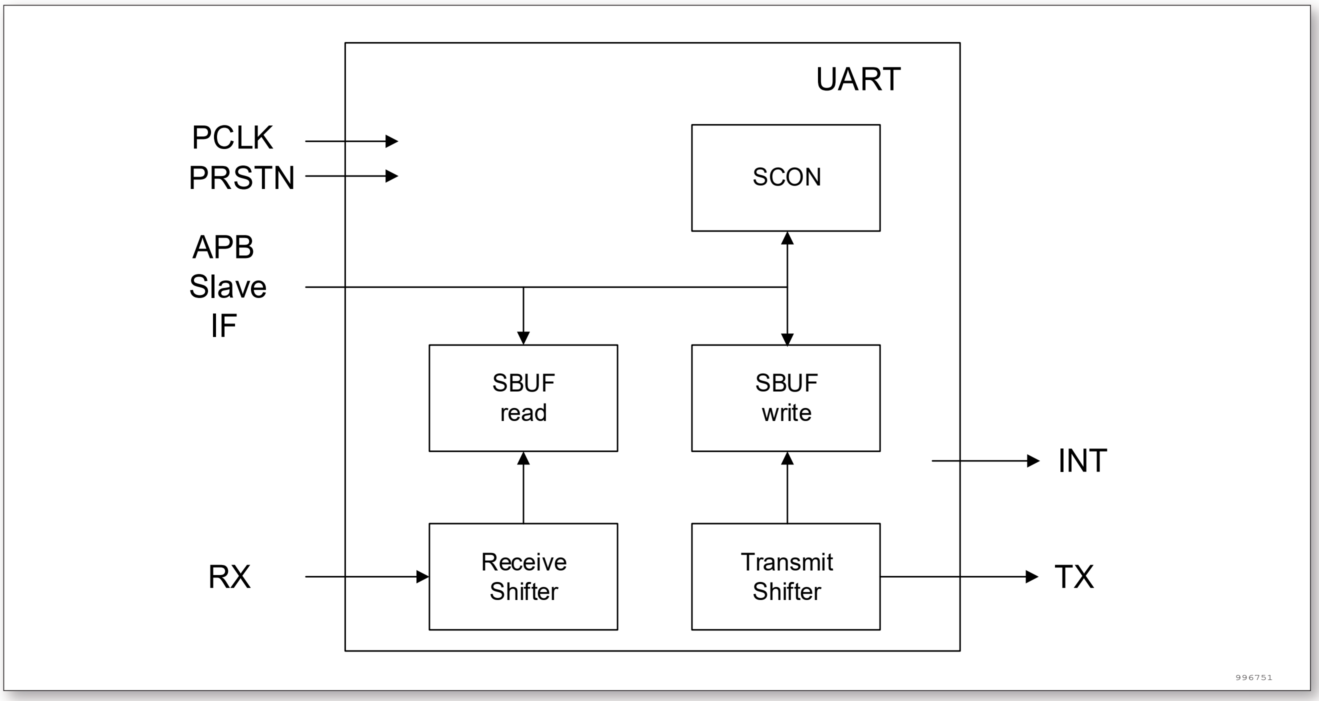

图 154. 结构框图

22.3 工作模式

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

22.3.1 Mode0（同步模式，半双工）

当工作在 Mode0 时，UART 为同步传输模式，其波特率固定为 PCLK 时钟频率的 1/12。UART 收发数据都由 RXD 输入输出，RXD 此时为双向端口。同步移位时钟由 TXD 输出，TXD 此时为输出端口。注意，本模式只能作为主机发送 UART 同步移位时钟，不可以作为从机接收外部输入的 UART 同步移位时钟。Mode0 时，传输数据位宽为 8- Bit，没有起始位和结束位。

当 UARTx_SCON.SM[1:0]=2'b00，可进入 Mode0 工作模式。

发送数据

发送数据时，清除 UARTx_SCON.REN 位，并将数据写入 UARTx_SBUF 寄存器中。此时，发送数据将从 RXD 输出（低位在先，高位在后），同步移位时钟从 TXD 输出。

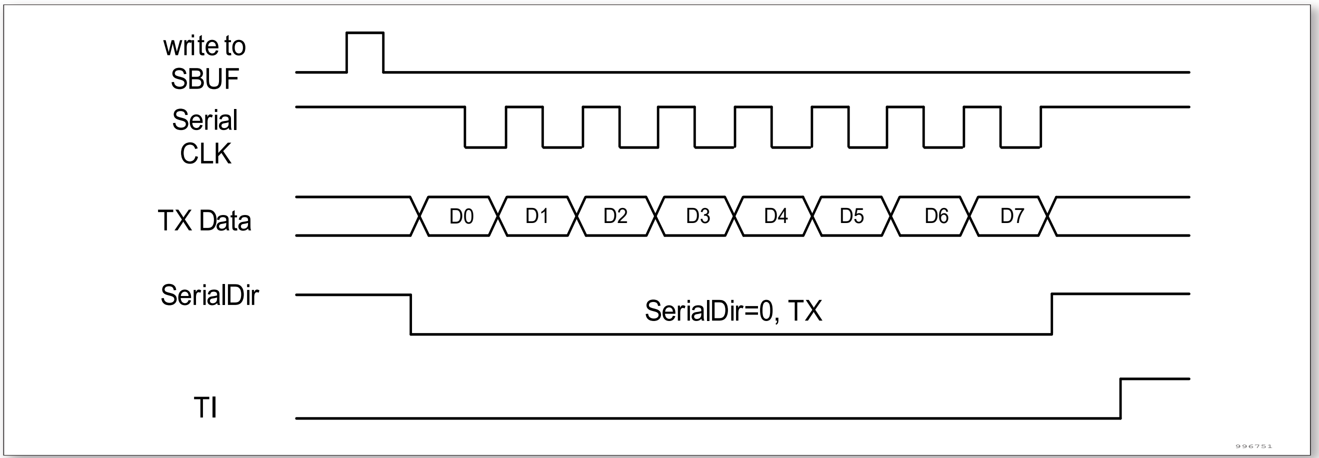

图 155. Mode0 发送数据

接收数据

接收数据时，将 UARTx_SCON.REN 位置 1，并将 UARTx_ISR.RC 位清零。当接收结束，数据可从 UARTx_SBUF 寄存器读取。此时，接收数据从 RXD 输入（低位在先，高位在后），同步移位时钟从 TXD 输出。

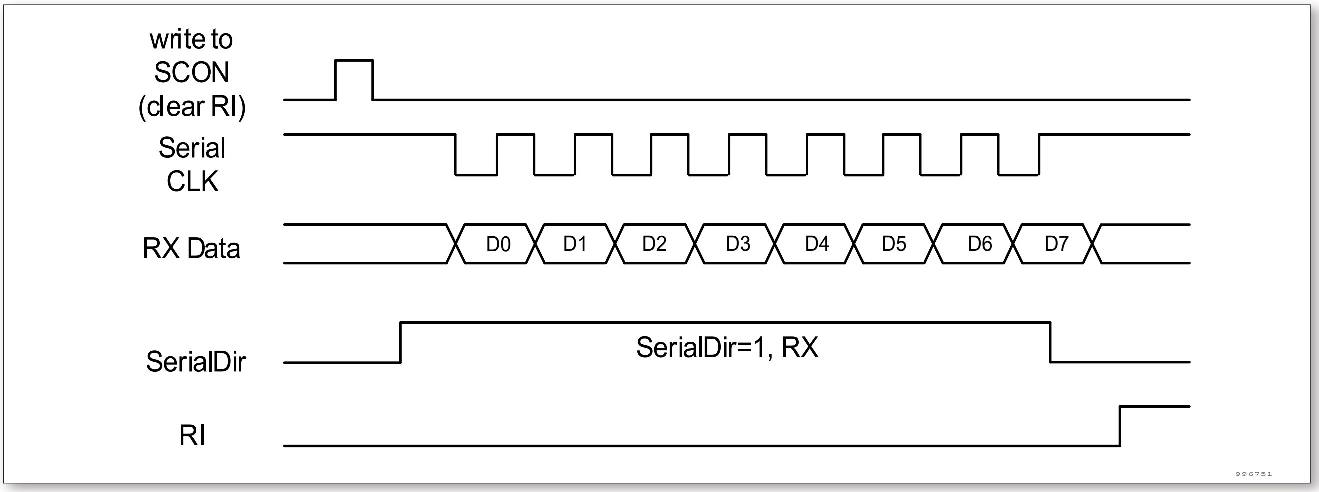

图 156. Mode0 接收数据

22.3.2 Mode1（异步模式，全双工）

当工作在 Mode1 时，发送数据通过 TXD 发送，接收数据通过 RXD 接收。该数据帧由以下比特位组成：1-Bit 起始位“0”开始，紧接着 8-Bit 数据位（低位在先，高位在后），最后是 1/1.5/2-Bit 结束位“1”。结束位长度由

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

UARTx_SCON.STOPBIT[1:0] 选择。该模式下，波特率由内部 16-Bit 可编程波特率发生器产生。波特率计算方式请见后面波特率编程章节。

当 UARTx_SCON.SM[1:0]=2'b01 时，可进入 Mode1 工作模式。

发送数据

发送数据时，与 UARTx_SCON.REN 的值无关，将所发送数据写入 UARTx_SBUF 寄存器中，数据就会从 TXD 移出（低位在先，高位在后）。

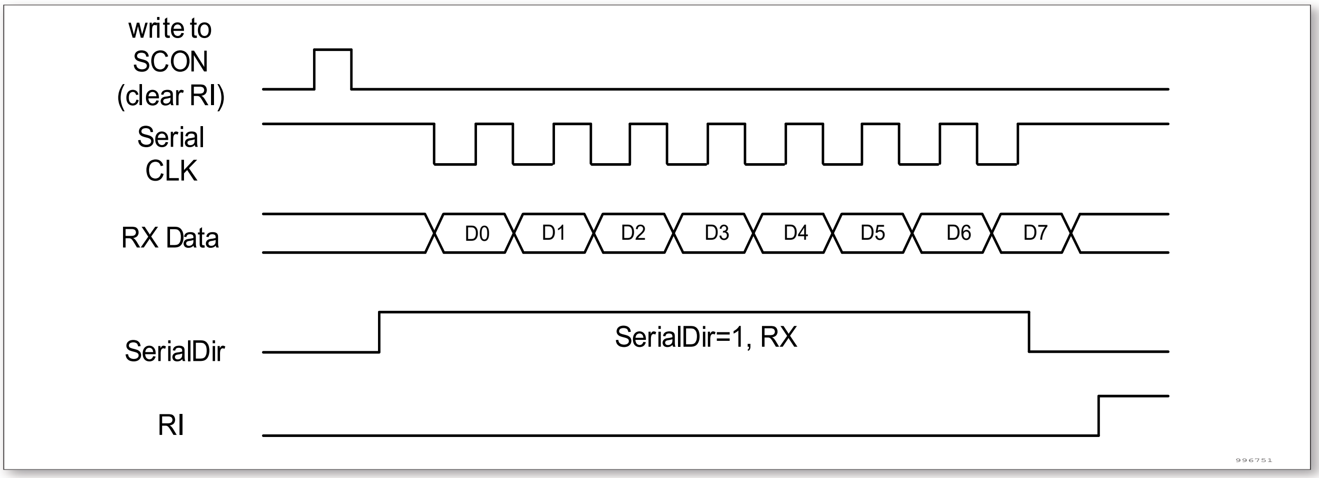

图 157. Mode1 发送数据

接收数据

接收数据时，将 UARTx_SCON.REN 位置 1，并将 UARTx_ISR.RC 位清 0。开始接收 RXD 上数据（低位在先，高位在后），当接收完毕，可以从 UARTx_SBUF 寄存器读出。

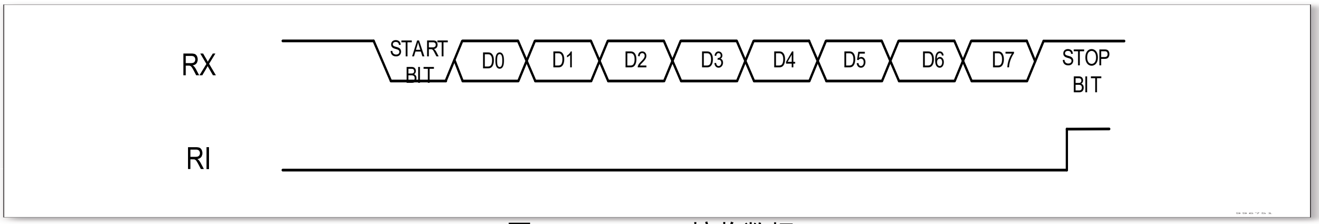

图 158. Mode1 接收数据

22.3.3 Mode2（异步模式，全双工）

当工作在 Mode2 时，发送数据通过 TXD 发送，接收数据通过 RXD 接收。该数据由以下比特位组成：1-Bit 起始位"0"开始，接着是 8-Bit 数据位，1-Bit TB8 位和 1/1.5/2-Bit 结束位。结束位长度由 UARTx_SCON.STOPBIT[1:0] 选择。额外的 TB8 位用于在多机通讯环境下使用，当 TB8=1，表明所接收的是地址帧；当 TB8=0，表明所接收的是数据帧。当不需要多机通讯时，此位也可以作为奇偶校验位或者数据位来使用。该模式下，波特率为 PCLK 时钟的固定分频，波特率计算方式请见后面波特率编程章节。

当 UARTx_SCON.SM[1:0]=2'b10，可进入 Mode2 工作模式。

发送数据

发送数据时，与 UARTx_SCON.REN 的值无关，将所发送数据写入 UARTx_SBUF 寄存器中，数据就会从 TXD 移出（低位在先，高位在后）。

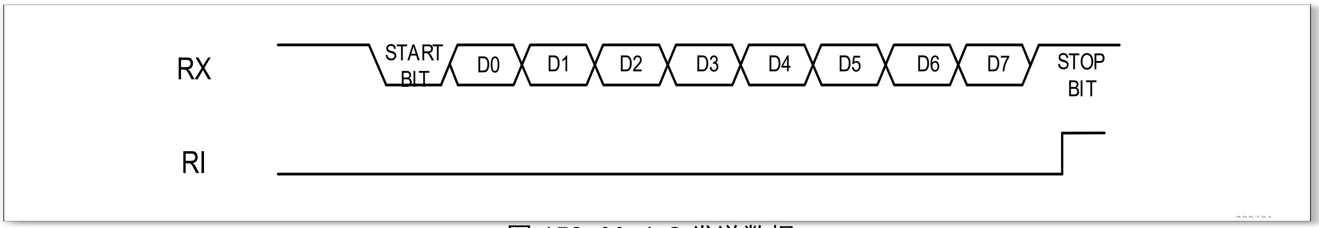

图 159. Mode2 发送数据

接收数据

接收数据时，需将 UARTx_SCON.REN 位置 1，并将 UARTx_ISR.RC 位清 0。开始接收 RXD 上的数据（低位在先，高位在后），当接收完毕，可以从 UARTx_SBUF 寄存器读取。

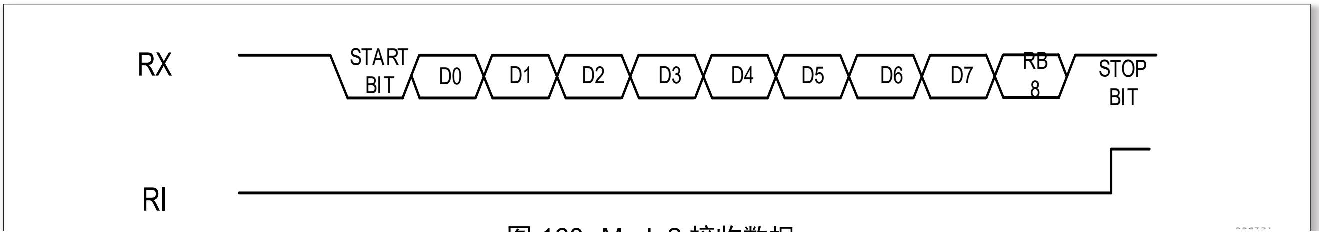

图 160. Mode2 接收数据

22.3.4 Mode3（异步模式，全双工）

当工作在 Mode3 时，发送数据通过 TXD 发送，接收数据通过 RXD 接收。该数据由以下比特位组成组成：1-Bit 起始位“0”开始，接着是 8-Bit 数据位，1-Bit TB8 位和 1/1.5/2- Bit 结束位。结束位长度由 UARTx_SCON.STOPBIT[1:0]选择。额外的 TB8 位用于在多机通讯环境下使用，当 TB8=1，表明所接收的是地址帧；当 TB8=0，表明所接收的是数据帧。当不需要多机通讯时，此位也可以作为奇偶校验位或者数据位来使用。该模式下，波特率由内部 16-Bit 可编程波特率发生器产生。波特率计算方式请见后面波特率编程章节。

当 UARTx_SCON.SM[1:0]=2’b11，可进入 Mode3 工作模式。

发送数据

发送数据时，与 UARTx_SCON.REN 的值无关，将所发送数据写入 UARTx_SBUF 寄存器中，数据就会从 TXD 移出（低位在先，高位在后）。

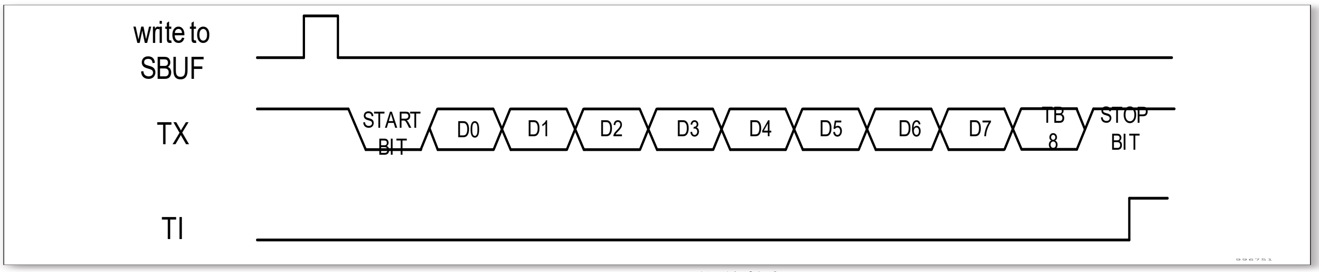

图 161. Mode3 发送数据

接收数据

接收数据时，将 UARTx_SCON.REN 位置 1，并将 UARTx_ISR.RC 位清 0。开始接收 RXD 上数据（低位在先，高位在后），当接收完毕，可以从 UARTx_SBUF 寄存器读出。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

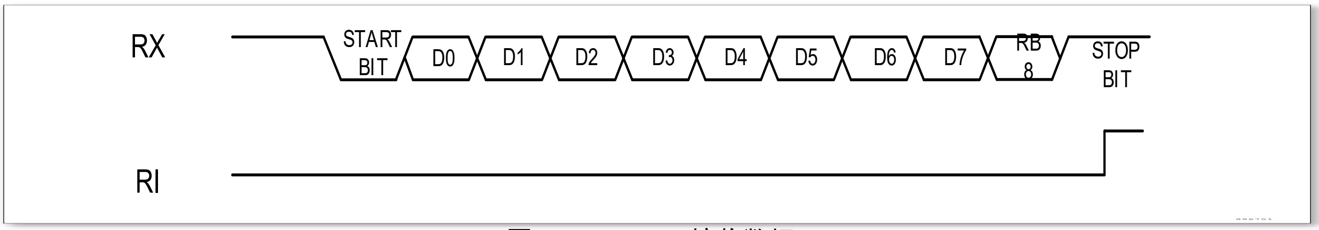

图 162. Mode3 接收数据

22.3.5 Single-wire Half-duplex（单线半双工）

当 UARTx_SCON.HDSEL=1’b1，可进入单线半双工传输模式。在该模式下，TX 与 RX 信号线在模块内部相连。RX 信号不再使用，接收发送的数据都经由 TX 信号完成。TX 信号的方向控制由硬件逻辑完成，无需软件控制。当发送缓存为空时，TX 信号始终为输入（接收状态）。当向发送缓存填入一个数据，TX 信号变为输出（发送状态）。当发送完成，发送缓存变为空，TX 信号又回到输入（接收状态）。

注意：

1. 即使模块处于接收过程中，只要发送缓存被填入数据，TX 信号立即变为输出（发送状态）。接收过程将被打断，并得到一个错误的接收数据。
2. 单线半双工模式并不是 Mode0 模式，虽然两者都是用一根信号分时复用地完成了数据的收发功能。但是 Mode0 为同步传输，有同步时钟。而单线半双工模式为异步传输，所有的波特率计算还是遵照 Mode1-2-3 的波特率公式，仅仅是收发功能复用到了 TX 这一根信号上面。

22.4 波特率编程

22.4.1 Mode0

当工作在 Mode0 时，UART 同步移位时钟频率固定为 PCLK 的 1/12。具体公式如下所示：

$$BaudRate = \frac{Freq}{12}$$

其中，Freq 为 PCLK 的时钟频率。

22.4.2 Mode1/3

当工作在 Mode1 或者 Mode3 时，波特率由可编程波特率发生器产生。具体公式如下所示：

$$BaudRate = \frac{Freq}{OVER * SCNT}$$

其中，Freq 为 PCLK 的时钟频率；OVER 可以为 16 或者 8，由 UARTx_SCON.OVER 的值决定；SCNT 为 16-Bit 波特率计数器 UARTx_SCNT 的计数值。表 84为常用系统主频和波特率的对照表。该表格同样适用于 LPUART0/1 的波特率计算。

注意：LPUART0/1 支持 OVER4/OVER8/OVER16。UART0/1 仅支持 OVER8/OVER16。

表 84: PCLK=4MHz 波特率计算表

波特率	OVER4(仅 LPUART0/1 支持)			OVER8			OVER16		
	CNT	实际波特率	误差%	CNT	实际波特率	误差%	CNT	实际波特率	误差%
2400	417	2398.08	-0.08%	208	2403.85	0.16%	104	2403.85	0.16%
4800	208	4807.69	0.16%	104	4807.69	0.16%	52	4807.69	0.16%
9600	104	9615.38	0.16%	52	9615.38	0.16%	26	9615.38	0.16%

波特率	OVER4(仅 LPUART0/1 支持)			OVER8			OVER16		
	CNT	实际波特率	误差%	CNT	实际波特率	误差%	CNT	实际波特率	误差%
19200	52	19230.77	0.16%	26	19230.77	0.16%	13	19230.77	0.16%
38400	26	38461.54	0.16%	13	38461.54	0.16%	7	35714.29	-6.99%
57600	17	58823.53	2.12%	9	55555.56	-3.55%	4	62500.00	8.51%
76800	13	76923.08	0.16%	7	71428.57	-6.99%	3	83333.33	8.51%
115200	9	111111.11	-3.55%	4	125000.00	8.51%	2	125000.00	8.51%

表 85: PCLK=8MHz 波特率计算表

波特率	OVER4(仅 LPUART0/1 支持)			OVER8			OVER16		
	CNT	实际波特率	误差%	CNT	实际波特率	误差%	CNT	实际波特率	误差%
2400	833	2400.96	0.04 %	417	2398.08	-0.08%	208	2403.85	0.16%
4800	417	4796.16	-0.08%	208	4807.69	0.16%	104	4807.69	0.16%
9600	208	9615.38	0.16 %	104	9615.38	0.16%	52	9615.38	0.16%
19200	104	19230.77	0.16 %	52	19230.77	0.16%	26	19230.77	0.16%
38400	52	38461.54	0.16 %	26	38461.54	0.16%	13	38461.54	0.16%
57600	35	57142.86	-0.79%	17	58823.53	2.12%	9	55555.56	-3.55%
76800	26	76923.08	0.16 %	13	76923.08	0.16%	7	71428.57	-6.99%
115200	17	117647.06	2.12 %	9	111111.11	-3.55%	4	125000.00	8.51%

表 86: PCLK=16MHz 波特率计算表

波特率	OVER4(仅 LPUART0/1 支持)			OVER8			OVER16		
	CNT	实际波特率	误差%	CNT	实际波特率	误差%	CNT	实际波特率	误差%
2400	1667	2399.52	-0.02%	833	2400.96	0.04%	417	2398.08	-0.08%
4800	833	4801.92	0.04%	417	4796.16	-0.08%	208	4807.69	0.16%
9600	417	9592.33	-0.08%	208	9615.38	0.16%	104	9615.38	0.16%
19200	208	19230.77	0.16%	104	19230.77	0.16%	52	19230.77	0.16%
38400	104	38461.54	0.16%	52	38461.54	0.16%	26	38461.54	0.16%
57600	69	57971.01	0.64%	35	57142.86	-0.79%	17	58823.53	2.12%
76800	52	76923.08	0.16%	26	76923.08	0.16%	13	76923.08	0.16%
115200	35	114285.71	-0.79%	17	117647.06	2.12%	9	111111.11	-3.55%
128000	31	129032.26	0.81%	16	125000.00	-2.34%	8	125000.00	-2.34%
256000	16	250000.00	-2.34%	8	250000.00	-2.34%	4	250000.00	-2.34%

表 87: PCLK=24MHz 波特率计算表

波特率	OVER4(仅 LPUART0/1 支持)			OVER8			OVER16		
	CNT	实际波特率	误差%	CNT	实际波特率	误差%	CNT	实际波特率	误差%
2400	2500	2400.00	0.00%	1250	2400.00	0.00%	625	2400.00	0.00%
4800	1250	4800.00	0.00%	625	4800.00	0.00%	313	4792.33	-0.16%

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

波特率	OVER4(仅 LPUART0/1 支持)			OVER8			OVER16		
	CNT	实际波特率	误差%	CNT	实际波特率	误差%	CNT	实际波特率	误差%
9600	625	9600.00	0.00%	313	9584.66	-0.16%	156	9615.38	0.16%
19200	313	19169.33	-0.16%	156	19230.77	0.16%	78	19230.77	0.16%
38400	156	38461.54	0.16%	78	38461.54	0.16%	39	38461.54	0.16%
57600	104	57692.31	0.16%	52	57692.31	0.16%	26	57692.31	0.16%
76800	78	76923.08	0.16%	39	76923.08	0.16%	20	75000.00	-2.34%
115200	52	115384.62	0.16%	26	115384.62	0.16%	13	115384.62	0.16%
128000	47	127659.57	-0.27%	23	130434.78	1.90%	12	125000.00	-2.34%
256000	23	260869.57	1.90%	12	250000.00	-2.34%	6	250000.00	-2.34%

表 88: PCLK=32MHz 波特率计算表

波特率	OVER4(仅 LPUART0/1 支持)			OVER8			OVER16		
	CNT	实际波特率	误差%	CNT	实际波特率	误差%	CNT	实际波特率	误差%
2400	3333	2400.24	0.01%	1667	2399.52	-0.02%	833	2400.96	0.04%
4800	1667	4799.04	-0.02%	833	4801.92	0.04%	417	4796.16	-0.08%
9600	833	9603.84	0.04%	417	9592.33	-0.08%	208	9615.38	0.16%
19200	417	19184.65	-0.08%	208	19230.77	0.16%	104	19230.77	0.16%
38400	208	38461.54	0.16%	104	38461.54	0.16%	52	38461.54	0.16%
57600	139	57553.96	-0.08%	69	57971.01	0.64%	35	57142.86	-0.79%
76800	104	76923.08	0.16%	52	76923.08	0.16%	26	76923.08	0.16%
115200	69	115942.03	0.64%	35	114285.71	-0.79%	17	117647.06	2.12%

表 89: PCLK=48MHz 波特率计算表

波特率	OVER4(仅 LPUART0/1 支持)			OVER8			OVER16		
	CNT	实际波特率	误差%	CNT	实际波特率	误差%	CNT	实际波特率	误差%
2400	5000	2400.00	0.00%	2500	2400.00	0.00%	1250	2400.00	0.00%
4800	2500	4800.00	0.00%	1250	4800.00	0.00%	625	4800.00	0.00%
9600	1250	9600.00	0.00%	625	9600.00	0.00%	313	9584.66	-0.16%
19200	625	19200.00	0.00%	313	19169.33	-0.16%	156	19230.77	0.16%
38400	313	38338.66	-0.16%	156	38461.54	0.16%	78	38461.54	0.16%
57600	208	57692.31	0.16%	104	57692.31	0.16%	52	57692.31	0.16%
76800	156	76923.08	0.16%	78	76923.08	0.16%	39	76923.08	0.16%
115200	104	115384.62	0.16%	52	115384.62	0.16%	26	115384.62	0.16%

22.4.3 Mode2

当工作在 Mode2 时，波特率为固定值，具体公式如下所示：

$$BaudRate = \frac{Freq}{OVER}$$

其中，Freq 为 PCLK 的时钟频率；OVER 可以为 32 或者 16，由 UARTx_SCON[9] 的值决定。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

22.5 传输数据结构

在不同工作模式下，传输数据的组成有所不同，请参考表 93：

表 90: 传输数据组成结构

工作模式	传输位宽	数据组成
Mode0	8-Bit	8-Bit Data
Mode1	10-Bit	1-Bit Start Bit + 8-Bit Data + (1/1.5/2)-Bit Stop Bit
Mode2	11-Bit	1-Bit Start Bit + 8-Bit Data + 1-Bit B8 + (1/1.5/2)-Bit Stop Bit
Mode3	11-Bit	1-Bit Start Bit + 8-Bit Data + 1-Bit B8 + (1/1.5/2)-Bit Stop Bit

B8 数据位比较特殊，在不同数据位比较特殊，在不同应用下具有不同的含义，请参考表 91：

表 91: B8 数据含义

应用场景	UARTx_SCON.ADRDET	UARTx_SCON.B8CONT[1:0]	B8 数据含义
奇偶校验	–	01/10	接收时，B8 是所收到的 8-Bit 数据的奇偶校验位； 发送时，B8 是所发送的 8-Bit 数据的奇偶校验位；
多机通讯	1	–	B8=1，代表当前是地址帧；B8=0，代表当前是数据帧；
其他	0	00/11	接收/发送的 DATA[8]

注意：

-当开启多机通讯模式，接收数据奇偶校验自动关闭；发送数据奇偶校验仍受 B8CONT 控制；

22.6 帧错误检测

当工作在 Mode1/2/3 时，UART 具有帧错误检测功能，硬件会自动检测接收到的帧数据是否带有效的 Stop 位。如果没有收到有效 Stop 位，则 UARTx_ISR.FE 置 1。UARTx_ISR.FE 位由硬件置 1，软件清 0，如果软件未及时清 0，则后续收到数据即使带有效 Stop 位，也不会把 UARTx_ISR.FE 标志清 0。

22.7 多机通讯

当工作在 Mode2/3 时，可以打开多机通讯功能，为此在其帧格式中增加了 1 位 TB8/RB8。

将 UARTx_SCON.ADRDET 位置“1”，可开启多机通讯功能。

当开启多机通讯后，主机可以通过 UARTx_SBUF[8] 来区分当前的发送帧是地址帧（UARTx_SBUF[8]=1）还是数据帧（UARTx_SBUF[8]=0）。从机会忽略 RB8 位（接收到的 Bit8 数据位）为“0”的接收帧，该帧数据不会存入到从机的 UARTx_SBUF 寄存器中，从机也不会产生接收中断当接收帧的 RB8 位（接收到的 Bit8 数据位）为“1”表明其是地址帧，从机会继续判断接收到的地址与其自身地址是否相等。如果匹配，则从机会对 UARTx_SBUF[8] 置“1”，同时将该地址帧存入到 UARTx_SBUF 寄存器中。并对 UARTx_ISR.RC 置“1”，以表明该帧为地址帧并且地址已经匹配。从机软件看到 UARTx_SBUF[8]=1 并且 UARTx_ISR.RC=1 后，先把 UARTx_SCON.ADEDET 位清“0”，然后准备接受给它的数据帧。如果地址不等，表明主机并不是寻址该从机，从机硬件保持 UARTx_SBUF[8]

和 UARTx_ISR.RC 为“0”，软件保持 UARTx_SCON.ADRDET 位为“1”，从机继续处于地址监听状态。

22.7.1 自动地址识别

当开启多机通讯后（UARTx_SCON.ADRDET 位置“1”），自动地址识别功能也将开启。该功能由硬件实现，使得从机可以检测接收到每个地址帧，如果该地址与从机地址匹配，接收从机会给出 UARTx_ISR.RC 接收标志。如果地址不匹配，则接收从机不会给出任何接收标志。

如果有需要，也可以在 Mode1 下开启多机通讯位，此时 TB8 位由 Stop 位代替。当从机接收到匹配的地址帧和有效的 Stop 位时，UARTx_ISR.RC 会被置“1”。

22.7.2 给定地址

UART 设备的 UARTx_SADDR 寄存器用来表示自己的设备给定地址，UARTx_SADEN 寄存器是地址掩码，可以用来定义地址中的无关位。当 UARTx_SADEN 的某一位为“0”，表示该位地址为无关位，也就是说在地址匹配过程中，该位地址不参与地址匹配。这些无关位增加了寻址的灵活性，使得主机可以同时寻址一个或者多个从机设备。注意，如果需要给出唯一匹配地址，UARTx_SADEN 寄存器必须设为 8’hFF。给定地址公式如下所示：

GivenAddr = SADEN & SADDR

22.7.3 广播地址

广播地址是用来同时寻址所有从机设备的，一般广播地址为 8’hFF。

BroadCastA = SADEN | SADDR ddr

22.7.4 举例

假设某从机的 UARTx_SADDR 和 UARTx_SADEN 配置如下：

SADDR：8’b01101001

SADEN：8’b11111011

那么其给定地址和广播地址如下：

Given：8’b01101x01

Broadcast: 8’b11111x11

可见，主机可以用四个地址寻址到本从机，分别是：

8’b01101001 和 8’b01101101 (given address)

8’b11111011 和 8’b11111111 (broadcast address)

22.8 DMAC 硬件握手

UART 模块支持 DMAC 的硬件握手逻辑。

将 UARTx_SCON.DMACTXEN 设置为 1，可以打开 UART TX 的 DMAC 硬件握手逻辑。当发送缓存为空时，UART 会向 DMAC 发出数据搬运请求 TX REQ。DMAC 收到该 TX REQ 信号，则从 DMAC 源地址搬运一个帧的发送数据到 UARTx_SBUF 中。上述步骤重复发生，直到 DMAC 中所配置的数据长度全部搬运完毕。将 UARTx_SCON.DMACRXEN 设置为 1，可以打开 UART RX 的 DMAC 硬件握手逻辑。当一帧接收完成，UART 会向 DMAC 发出数据搬运请求 RX REQ。DMAC 收到该 RX REQ 信号，则从 UARTx_SBUF 中把接收数据搬运至 DMAC 的目标地址中。上述步骤重复发生，直到 DMAC 中所配置的数据长度全部搬运完毕。

22.9 硬件流控

通过增加 nCTS 和 nRTS 信号可以实现 UART 硬件流控的功能，即 UART 硬件模块根据 nCTS 和 nRTS 的高低电平自动控制数据的收发，而无需通过软件来判断。两个 UART 模块之间的硬件流控示意图如下所示：

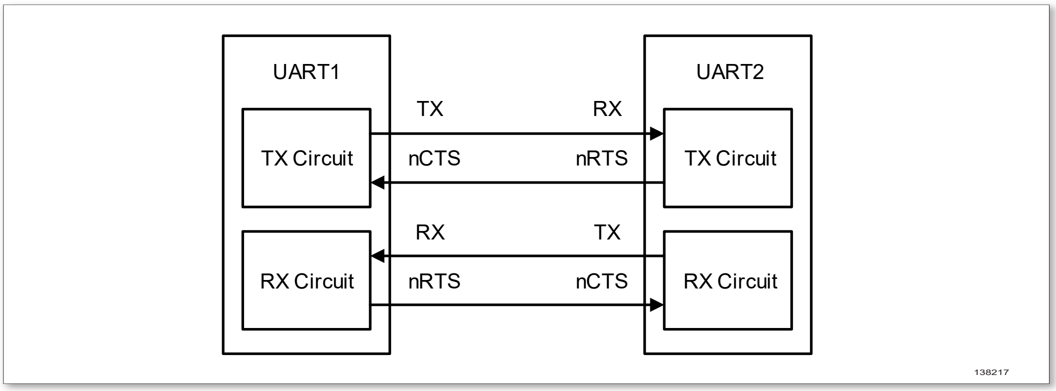

图 163. UART 硬件流控

22.9.1 nRTS 流控

当 nRTS 流控使能时（UARTx_SCON.RTSEN 设置为 1），当 UART 接收缓存空时，nRTS 信号拉低（低表示可以接收数据），通知对端设备可以发送下一帧数据。当接收缓存满时，nRTS 信号拉高（高表示不可以接收数据），通知对端设备在当前发送帧传输完成后，暂停发送下一帧数据。

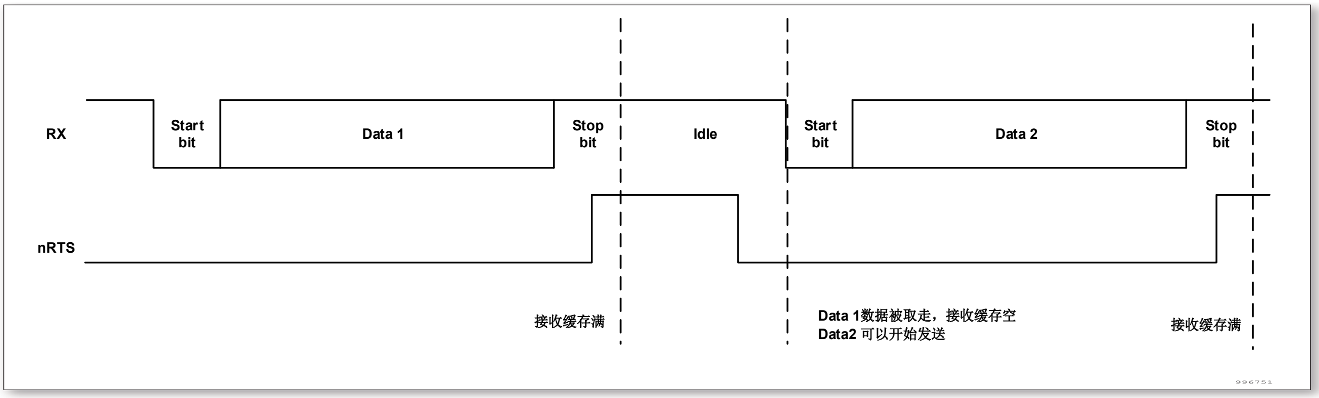

图 164. nRTS 硬件流控信号

22.9.2 CTS 流控

当 nCTS 流控使能时（UARTx_SCON.CTSEN 设置为 1），在 UART 发送下一帧数据之前，先判断 nCTS 的高低电平。如果 nCTS 为低电平（低电平表示对端设备可以接收下一帧数据），则 UART 发送下一帧数据。如果 nCTS 为高电平（高电平表示对端设备不可以接收下一帧数据），则 UART 暂停发送下一帧数据。nCTS 拉高不会影响当前发送帧的传输，只会暂停下一帧的发送，直到 nCTS 重新变低为止。

当 nCTS 流控使能时，一旦 nCTS 信号发生翻转，会把 UARTx_SFLAG.CTSIF 置 1。如果使能 CTSIE 中断（UARTx_SCON.CTSIE 设置为 1），则会产生中断。同时，nCTS 信号的高低电平会记录在 UARTx_SFLAG.CTS 标志位中。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

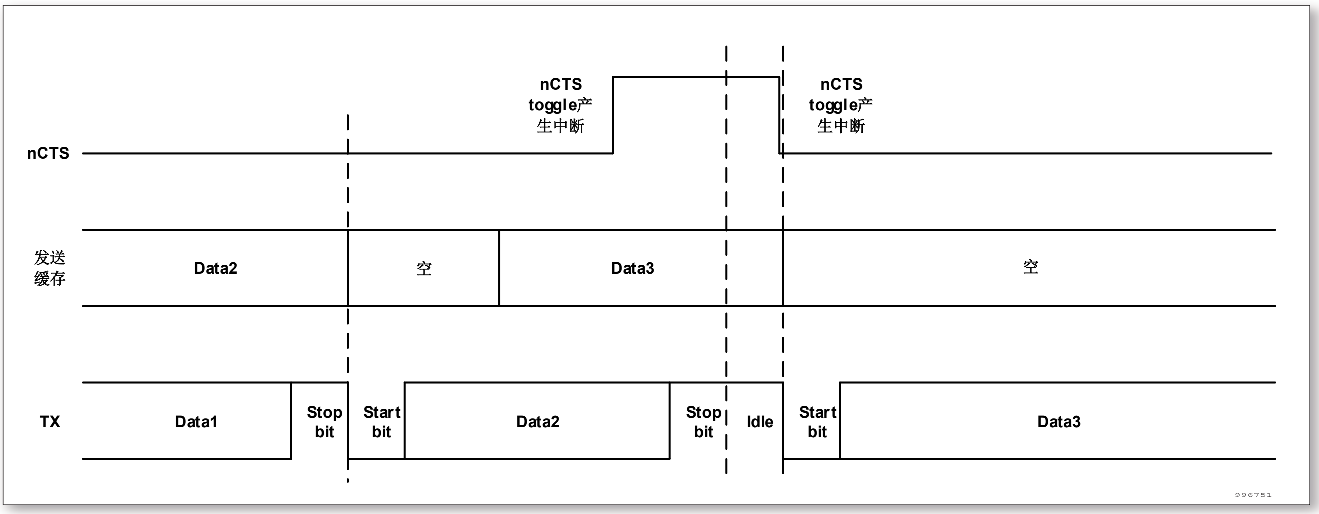

图 165. nCTS 硬件流控信号

22.10 收发端缓存

22.10.1 接收缓存

UART 模块接收端有一个帧（8/9-Bit）的接收缓存，也就是说当一帧数据接收完毕后，接收缓存中的数据会被一直保持，直到下一帧数据的 Stop 位接收完毕后，接收缓存才会更新到这一帧所收到的数据。

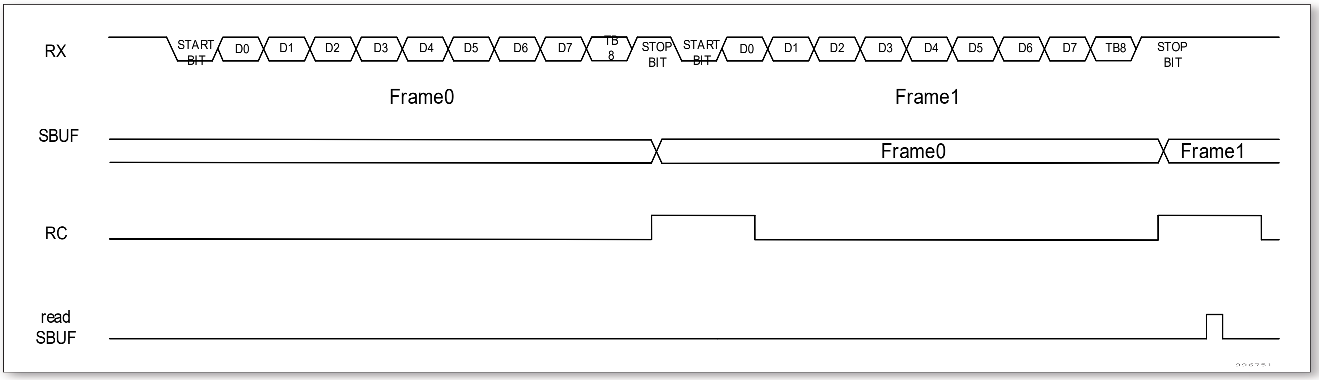

图 166. 接收缓存

22.10.2 发送缓存

UART 模块发送端有一个帧 (8/9-Bit) 的发送缓存，当 UART 在发送当前帧时，软件可以对 UARTx_SBUF 写入下一个发送数据。UARTx_ISR.TXE 位表示了当前发送缓存的空满状态，当 UARTx_ISR.TXE=0 时，表明当前发送缓存满，软件不能对 UARTx_SBUF 写入下一个发送数据。如果软件继续写入 UARTx_SBUF，则该笔写入数据会硬件丢弃。当 UARTx_ISR.TXE=1 时，表明当前发送缓存空，软件可以写入下一个需要发送的数据帧。该笔发送数据会保存在发送缓存中，直到当前发送数据完成，硬件自动把发送缓存中的数据装载入移位寄存器中发送出去。

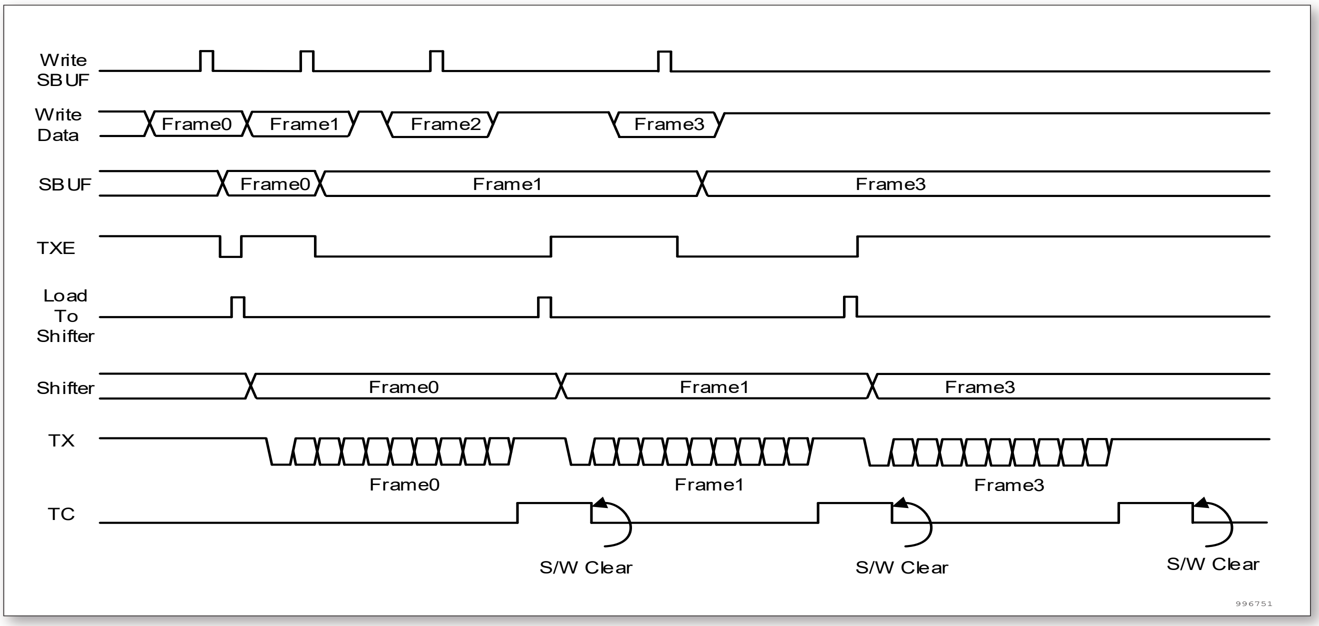

图 167. 发送缓存

22.11 UART 寄存器

表 92. UART 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x00	UARTx_SBUF	数据寄存器	0x0000 0000	小节 22.11.1
0x04	UARTx_SCON	控制寄存器	0x0000 0000	小节 22.11.2
0x08	UARTx_SADDR	地址寄存器	0x0000 0000	小节 22.11.3
0x0C	UARTx_SADEN	地址掩码寄存器	0x0000 0000	小节 22.11.4
0x10	UARTx_ISR	中断标志位寄存器	0x0000 0008	小节 22.11.5
0x14	UARTx_ICR	中断标志位清除寄存器	0x0000 0000	小节 22.11.6
0x18	UARTx_SCNT	波特率寄存器	0x0000 0000	小节 22.11.7

22.11.1 数据寄存器 (UARTx_SBUF)

起始地址： UART0:0x40000000 UART1:0x40000100 UART2:0x40006000 UART3:0x40006400

地址偏移：0x00

复位值：0x00000000

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

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

Bit	Field	Type	Reset	Description
31:9	Reserved			保留
8	DATA[8]	rw	0x00	在 Mode0/1 下，读取该位为 0，写入该位无效； 在 Mode2/3 下，该位表示 Bit8 数据位，分以下两种情况： （1）当硬件奇偶校验位开启时，接收时该位为接收数据奇偶校验位，校验由硬件进行，如校验出错，校验错误标志位 PE 置 1；发送时该位无效，发送数据奇偶校验位由硬件计算并发送； （2）当硬件奇偶校验位关闭时，接收时该位为接收数据 Bit8；发送时该位为发送数据 Bit8； 注意：当开启多机通讯模式，接收数据奇偶校验自动关闭；发送数据奇偶校验仍受 B8CONT 控制；
7:0	DATA[7:0]	rw	0x00	发送数据时，将发送数据写入该寄存器；接收数据时，数据接收完毕后，从该寄存器中读取。

22.11.2 控制寄存器 (UARTx_SCON)

起始地址： UART0:0x40000000 UART1:0x40000100 UART2:0x40006000 UART3:0x40006400

地址偏移：0x04

复位值：0x00000000

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="9" rowspan="2">Reserved</td><td>HDSEL</td><td>FEIE</td><td>CTSIE</td><td>CTSEN</td><td>RTSEN</td><td>DMATXEN</td><td>DMARXEN</td></tr>
<tr><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="2">STOPBIT</td><td>PEIE</td><td colspan="3" rowspan="2">Reserved</td><td>OVER</td><td>TXEIE</td><td colspan="2">SM</td><td>ADRDET</td><td>REN</td><td colspan="2">B8CONT</td><td>TCIE</td><td>RCIE</td></tr>
<tr><td colspan="2">RW</td><td>RW</td><td>RW</td><td>RW</td><td colspan="2">RW</td><td>RW</td><td>RW</td><td colspan="2">RW</td><td>RW</td><td>RW</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:23	Reserved			保留
22	HDSEL	rw	0x00	单线半双工模式是能；0: 关闭单线半双工模式；1：打开单线半双工模式；
21	FEIE	rw	0x00	帧错误中断使能；0：关闭中断；1：打开中断；
20	CTSIE	rw	0x00	CTS 信号翻转中断使能；0：关闭中断；1：打开中断；
19	CTSEN	rw	0x00	硬件 flow control 信号使能位；0：关闭 flow control 信号；1：打开 flow control 信号；
18	RTSEN	rw	0x00	硬件 flow control 信号使能位；0：关闭 flow control 信号；1：打开 flow control 信号；
17	DMATXEN	rw	0x00	TX DMAC 的硬件握手信号使能位；0：关闭硬件握手信号；1：打开硬件握手信号；

Bit	Field	Type	Reset	Description
16	DMARXEN	rw	0x00	RX DMAC 的硬件握手信号使能位；0：关闭硬件握手信号；1：打开硬件握手信号；
15:14	STOPBIT	rw	0x00	stop bit 长度选择；00:1-bit；01:1.5-bit；10:2-bit；11：reserved； 注意：Mode0 时虽然没有 Stop Bit，但仍需把 STOPBIT[1:0] 保持为 2’b00；
13	PEIE	rw	0x00	奇偶校验错误中断使能位；0：奇偶校验错误中断关闭；1：奇偶校验错误中断打开； 数据接收标志产生于数据停止位接收完成，硬件奇偶校验与数据接收中断由于停止位设置不同，奇偶校验错误中断会提前于数据接收中断。使用此中断使能时请注意。建议如下： 方法 1：关闭 PEIE 中断使能，接收数据中断后软件判断 ISR.PE 奇偶校验是否正确 。方法 2：关闭 PEIE 中断使能，接收数据中断通过接收 SBUF.BIT8 软件判断奇偶校验是否正确。
12:10	Reserved			保留
9	OVER	rw	0x00	Mode0：无效； Mode1/3：0：16 采样分频；1：8 采样分频； Mode2: 0：32 采样分频；1：16 采样分频；
8	TXEIE	rw	0x00	TX 空中断使能位；0：TX Buffer 空中断关闭；1：TX Buffer 空中断打开；
7:6	SM	rw	0x00	工作模式；00：mode0；01：mode1；10：mode2；11：mode3；
5	ADRDET	rw	0x00	多机通讯地址自动识别使能位；0：关闭；1：打开；
4	REN	rw	0x00	Mode0：0：发送；1：接收； Mode1/2/3：0：发送；1：接收/发送；
3:2	B8CONT	rw	0x00	Bit8 数据控制位； 00：由软件读写 SBUF[8] 来决定；01：硬件偶校验； 10：硬件奇校验；11：保留；
1	TCIE	rw	0x00	发送中断使能位；0：发送中断关闭；1：发送中断打开；
0	RCIE	rw	0x00	接收中断使能位；0：接收中断关闭；1：接收中断打开；

22.11.3 地址寄存器 (UARTx_SADDR)

起始地址： UART0:0x40000000 UART1:0x40000100 UART2:0x40006000 UART3:0x40006400

地址偏移：0x08

复位值：0x00000000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

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

Bit	Field	Type	Reset	Description
31:8	Reserved			保留
7:0	SADDR	rw	0x00	从机设备地址寄存器

22.11.4 地址掩码寄存器 (UARTx_SADEN)

起始地址： UART0:0x40000000 UART1:0x40000100 UART2:0x40006000 UART3:0x40006400

地址偏移：0x0C

复位值：0x00000000

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

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

Bit	Field	Type	Reset	Description
31:8	Reserved			保留
7:0	SADEN	rw	0x00	从机设备地址掩码寄存器

22.11.5 中断标志位寄存器 (UARTx_ISR)

起始地址： UART0:0x40000000 UART1:0x40000100 UART2:0x40006000 UART3:0x40006400

地址偏移：0x10

复位值：0x00000008

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									CTS	CTSIF	PE	TXE	FE	TC	RC
									RO	RO	RO	RO	RO	RO	RO

Bit	Field	Type	Reset	Description
31:7	Reserved			保留
6	CTS	ro	0x00	CTS 信号标志位；硬件置 1；硬件清零；0：CTS 信号为低电平；1：CTS 信号为高电平；
5	CTSIF	ro	0x00	CTS 中断标志位；硬件置 1；软件清零；0：CTS 信号没有发生反转；1：CTS 信号发生反转；
4	PE	ro	0x00	奇偶校验错误标志位；硬件置 1；软件清零；0：无奇偶校验错误；1：奇偶校验错误；
3	TXE	ro	0x00	Tx Buffer 空标志位；硬件置 1；硬件清零；0：Tx Buffer 非空；1：Tx Buffer 空
2	FE	ro	0x00	帧错误标志位；0：硬件置 1；软件清零；
1	TC	ro	0x00	发送完毕中断标志位；硬件置 1；软件清零；0：发送未完成；1：发送完毕；
0	RC	ro	0x00	接收完毕中断标志位；硬件置 1；软件清零；0：接收未完成；1：接收完成；

22.11.6 中断标志位清除寄存器 (UARTx_ICR)

起始地址： UART0:0x40000000 UART1:0x40000100 UART2:0x40006000 UART3:0x40006400

地址偏移：0x14

复位值：0x00000000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										CTSIFCF	PECF	Res.	FECF	TCCF	RCCF
										R1W0	R1W0		R1W0	R1W0	R1W0

Bit	Field	Type	Reset	Description
31:6	Reserved			保留
5	CTSIFCF	r1w0	0x00	CTSIF 标志清楚位；写 0 清除；写 1 无效；
4	PECF	r1w0	0x00	PE 标志清楚位；写 0 清除；写 1 无效；
3	Reserved			保留
2	FECF	r1w0	0x00	FE 标志清除位；写 0 清零；写 1 无效；
1	TCCF	r1w0	0x00	TC 标志清除位；写 0 清零；写 1 无效；
0	RCCF	r1w0	0x00	RC 标志清除位；写 0 清零；写 1 无效；

22.11.7 波特率寄存器 (UARTx_SCNT)

起始地址： UART0:0x40000000 UART1:0x40000100 UART2:0x40006000 UART3:0x40006400

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

地址偏移：0x18

复位值：0x00000000

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

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

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	SCNT	rw	0x00	波特率计数器

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

23 低功耗同步异步收发器 (LPUART)

23.1 概述

LPUART 模块（LPUART0/1）支持以下基本功能：

- 配置时钟 PCLK
- 传输时钟 SCLK（SCLK 可选择 XTL、RCL 以及 PCLK）
- 系统低功耗模式下收发数据
- 半双工和全双工传输
- 单线半双工传输
- 8/9-Bit 传输数据长度
- 硬件奇偶校验
- 1/1.5/2-Bit 停止位
- 四种不同传输模式
- 16-Bit 波特率计数器
- 多机通讯
- 硬件地址识别
- DMAC 硬件传输握手
- 硬件流控
- 支持单线模式

23.2 功能框图

图 168为 LPUART 结构框图：

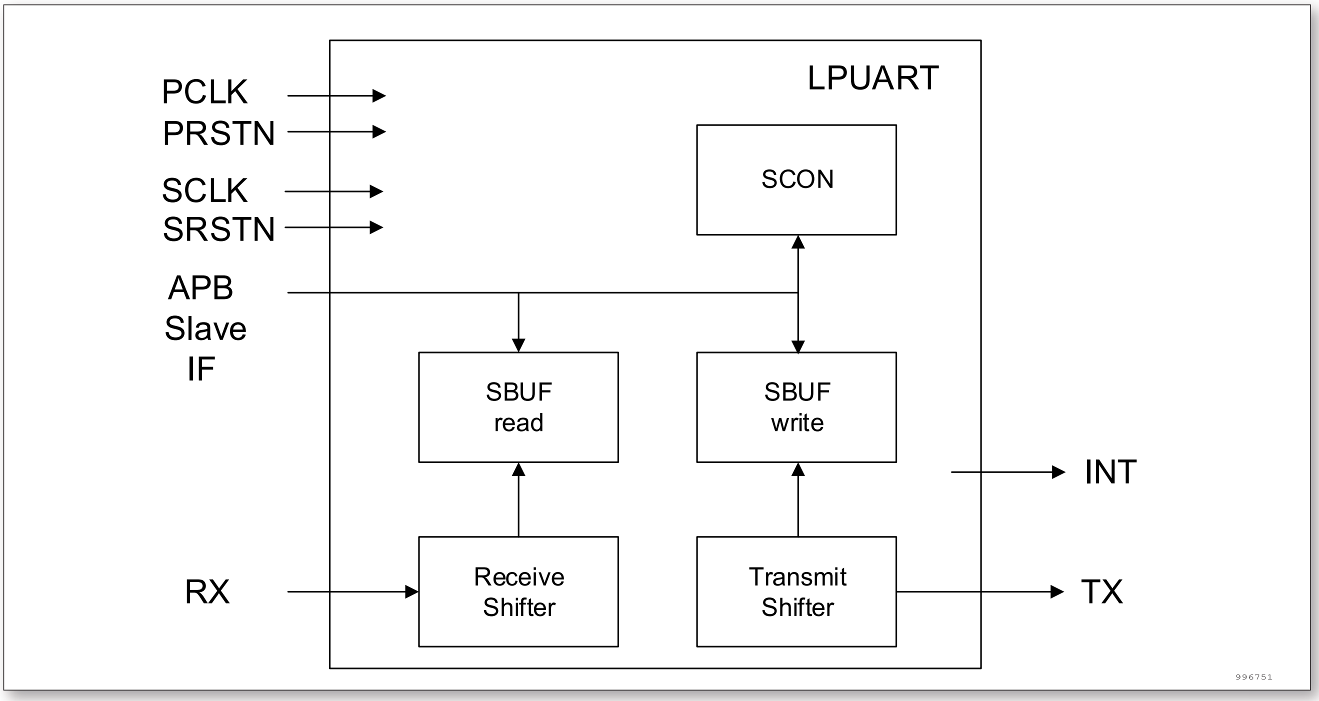

图 168. 结构框图

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

23.3 配置时钟和传输时钟

LPUART 模块有两个时钟：配置时钟 PCLK 和传输时钟 SCLK。

23.3.1 配置时钟

配置时钟（PCLK）用于系统 APB 总线对 LPUART 模块进行寄存器配置，配置时钟固定为 APB 总线时钟 PCLK。当系统进入深度睡眠（DeepSleep）模式，PCLK 时钟将会停止。

23.3.2 传输时钟

传输时钟（SCLK）用于 LPUART 数据收发逻辑工作，可选择外部低速晶振时钟（XTL）、内部低速 RC 时钟（RCL）以及 PCLK 时钟。

当系统进入深度睡眠（DeepSleep）模式，如果 SCLK 选择外部低速晶振时钟（XTL）或者内部低速 RC 时钟（RCL）。LPUART 仍旧可以进行正常的数据收发，而不受系统深度睡眠（DeepSleep）模式的影响。

23.4 工作模式

23.4.1 Mode0(同步模式，半双工)

当工作在 Mode0 时，LPUART 为同步传输模式，其波特率固定为 SCLK 时钟频率的 1/12。LPUART 收发数据都由 RXD 输入输出，RXD 此时为双向端口。同步移位时钟由 TXD 输出，TXD 此时为输出端口。注意，本模式只能作为主机发送 LPUART 同步移位时钟，不可以作为从机接收外部输入的 LPUART 同步移位时钟。Mode0 时，传输数据位宽为 8-Bit，没有起始位和结束位。

当 LPUARTx_SCON.SM[1:0]=2’b00，可进入 Mode0 工作模式。

发送数据

发送数据时，清除 LPUARTx_SCON.REN 位，并将数据写入 LPUARTx_SBUF 寄存器中。此时，发送数据将从 RXD 输出（低位在先，高位在后），同步移位时钟从 TXD 输出。

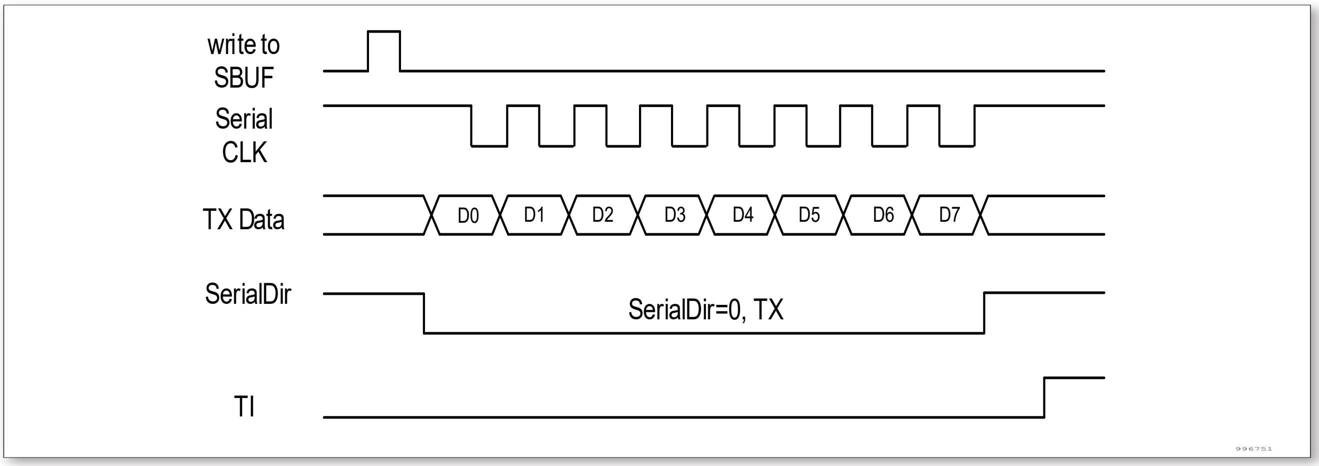

图 169. Mode0 发送数据

接收数据

接收数据时，将 LPUARTx_SCON.REN 位置 1，并将 LPUARTx_ISR.RC 位清零。当接收结束，数据可从 LPUARTx_SBUF 寄存器读取。此时，接收数据从 RXD 输入（低位在先，高位在后），同步移位时钟从 TXD 输出。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

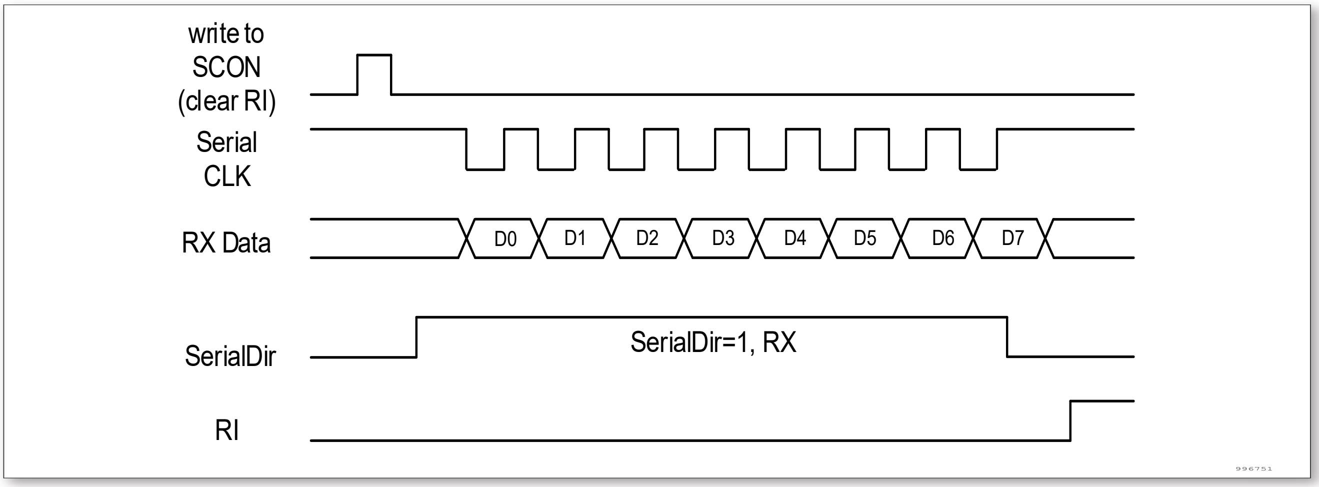

图 170. Mode0 接收数据

23.4.2 Mode1 （异步模式，全双工）

当工作在 Mode1 时，发送数据通过 TXD 发送，接收数据通过 RXD 接收。该数据帧由以下比特位组成：1-Bit 起始位“0”开始，紧接着 8-Bit 数据位（低位在先，高位在后），最后是 1/1.5/2-Bit 结束位“1”。结束位长度由 LPUARTx_SCON.STOPBIT[1:0] 选择。该模式下，波特率由内部 16-Bit 可编程波特率发生器产生。波特率计算方式请见后面波特率编程章节。

当 LPUARTx_SCON.SM[1:0]=2’b01 时，可进入 Mode1 工作模式。

发送数据

发送数据时，与 LPUARTx_SCON.REN 的值无关，将所发送数据写入 LPUARTx_SBUF 寄存器中，数据就会从 TXD 移出（低位在先，高位在后）。

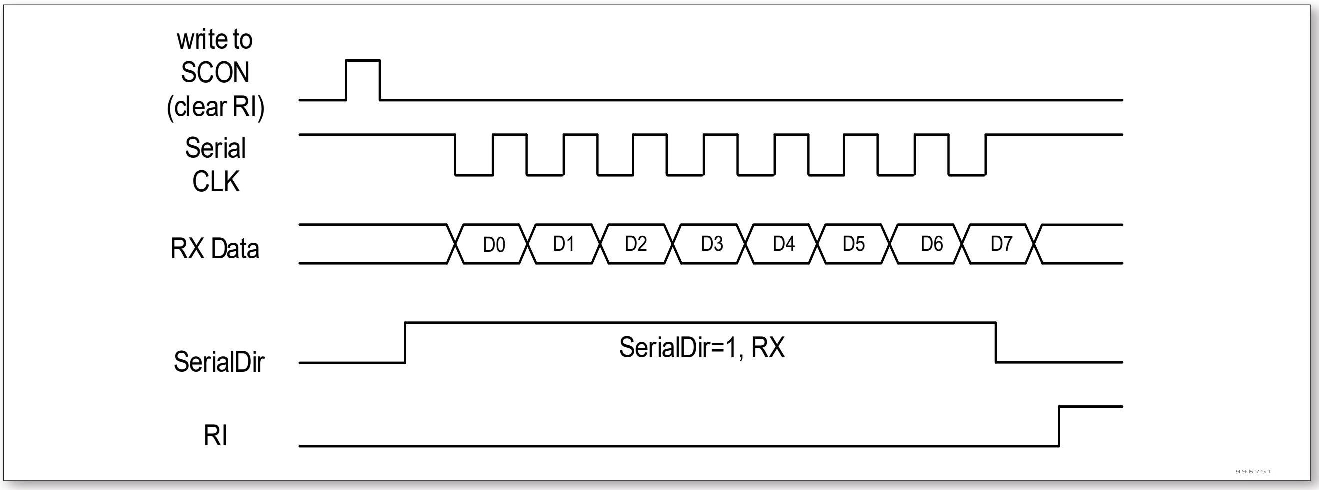

图 171. Mode1 发送数据

接收数据

接收数据时，将 LPUARTx_SCON.REN 位置 1，并将 LPUARTx_ISR.RC 位清 0。开始接收 RXD 上数据（低位在先，高位在后），当接收完毕，可以从 LPUARTx_SBUF 寄存器读出。

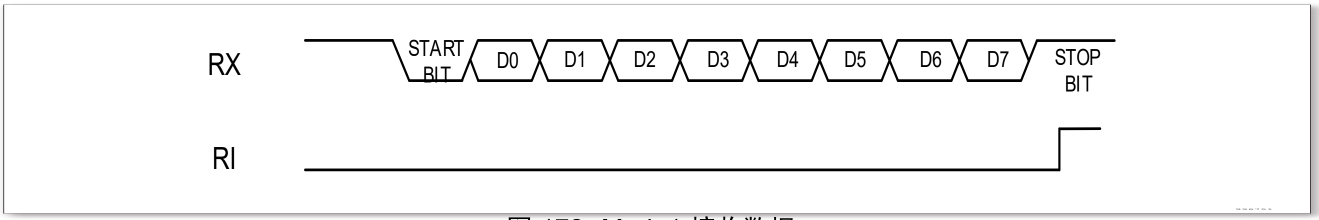

图 172. Mode1 接收数据

23.4.3 Mode2 （异步模式，全双工）

当工作在 Mode2 时，发送数据通过 TXD 发送，接收数据通过 RXD 接收。该数据由以下比特位组成：1-Bit 起始位“0”开始，接着是 8-Bit 数据位，1-Bit TB8 位和 1/1.5/2-Bit 结束位。结束位长度由 LPUARTx_SCON.STOPBIT[1:0]选择。额外的 TB8 位用于在多机通讯环境下使用，当 TB8=1，表明所接收的是地址帧；当 TB8=0，表明所接收的是数据帧。当不需要多机通讯时，此位也可以作为奇偶校验位或者数据位来使用。该模式下，波特率为传输时钟 SCLK 的固定分频，波特率计算方式请见后面波特率编程章节。

当 LPUARTx_SCON.SM[1:0]=2’b10，可进入 Mode2 工作模式。

发送数据

发送数据时，与 LPUARTx_SCON.REN 的值无关，将所发送数据写入 LPUARTx_SBUF 寄存器中，数据就会从 TXD 移出（低位在先，高位在后）。

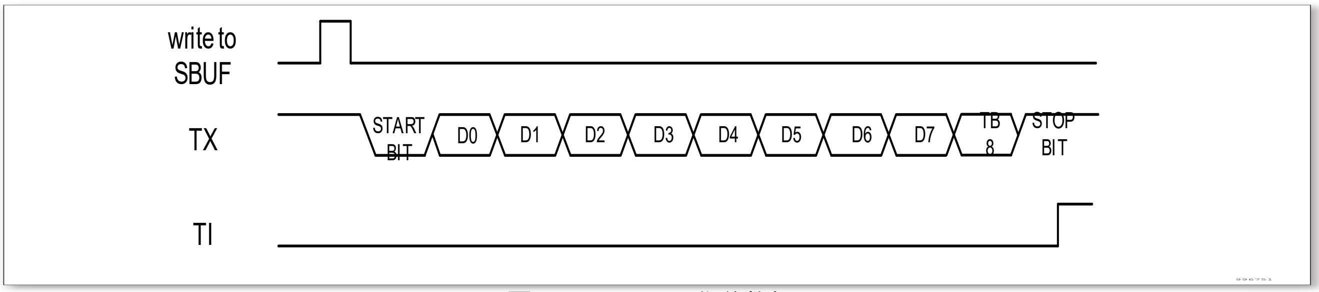

图 173. Mode2 发送数据

接收数据

接收数据时，需将 LPUARTx_SCON.REN 位置 1，并将 LPUARTx_ISR.RC 位清 0。开始接收 RXD 上的数据（低位在先，高位在后），当接收完毕，可以从 LPUARTx_SBUF 寄存器读取。

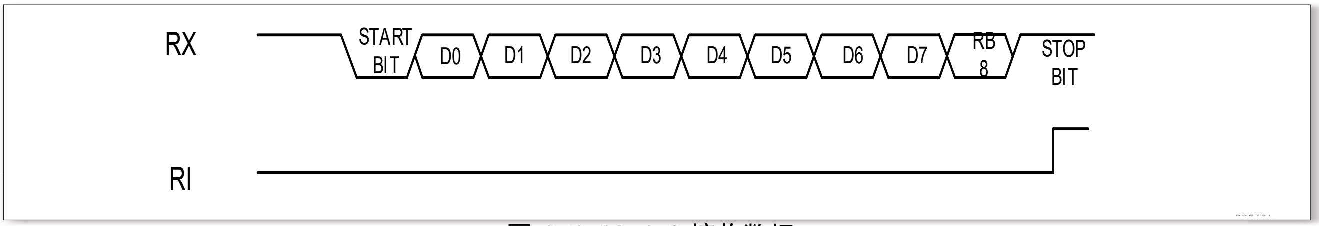

图 174. Mode2 接收数据

23.4.4 Mode3（异步模式，全双工）

当工作在 Mode3 时，发送数据通过 TXD 发送，接收数据通过 RXD 接收。该数据由以下比特位组成：1-Bit 起始位“0”开始，接着是 8-Bit 数据位，1-Bit TB8 位和 1/1.5/2-Bit 结束位。结束位长度由 LPUARTx_SCON.STOPBIT[1:0]选择。额外的 TB8 位用于在多机通讯环境下使用，当 TB8=1，表明所接收的是地址帧；当 TB8=0，表明所接收的是数据帧。当不需要多机通讯时，此位也可以作为奇偶校验位或者数据位来使用。该模式下，波特率由内部 16-Bit 可编程波特率发生器产生。波特率计算方式请见后面波特率编程章节。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

当 LPUARTx_SCON.SM[1:0]=2’b11，可进入 Mode3 工作模式。

发送数据

发送数据时，与 LPUARTx_SCON.REN 的值无关，将所发送数据写入 LPUARTx_SBUF 寄存器中，数据就会从 TXD 移出（低位在先，高位在后）。

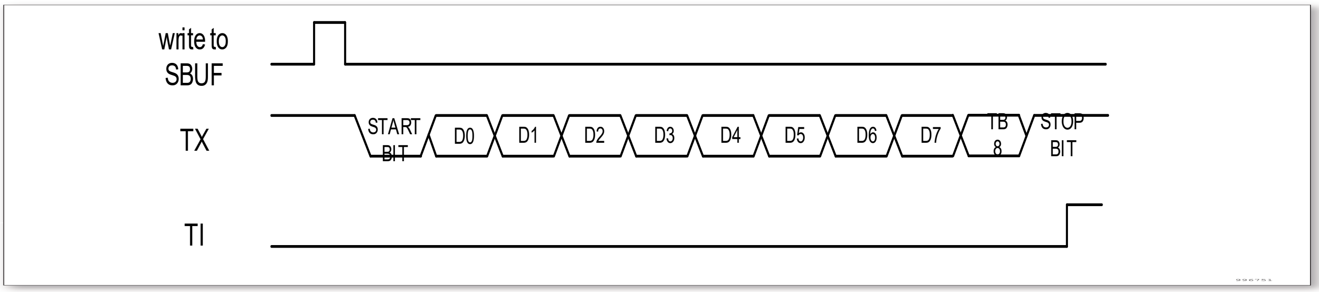

图 175. Mode2 发送数据

接收数据

接收数据时，将 LPUARTx_SCON.REN 位置 1，并将 LPUARTx_ISR.RC 位清 0。开始接收 RXD 上数据（低位在先，高位在后），当接收完毕，可以从 LPUARTx_SBUF 寄存器读出。

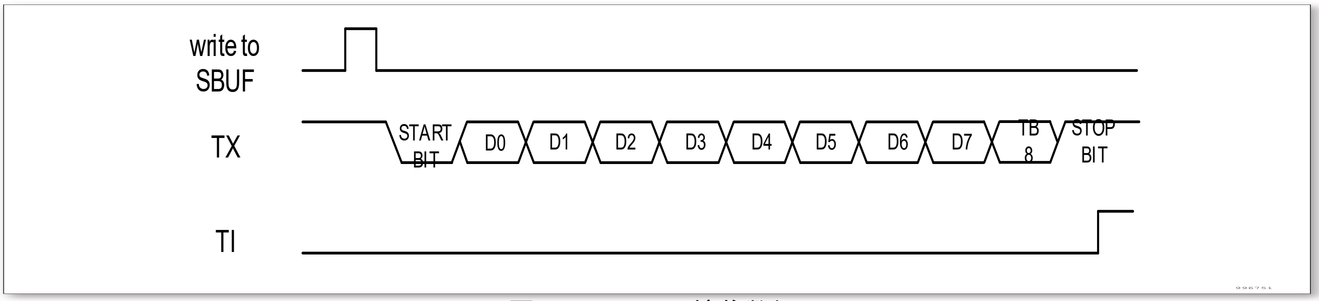

图 176. Mode3 接收数据

23.4.5 Single-wire Half-duplex（单线半双工）

当 LPUARTx_SCON.HDSEL=1’b1，可进入单线半双工传输模式。在该模式下，TX 与 RX 信号线在模块内部相连。RX 信号不再使用，接收发送的数据都经由 TX 信号完成。TX 信号的方向控制由硬件逻辑完成，无需软件控制。当发送缓存为空时，TX 信号始终为输入（接收状态）。当向发送缓存填入一个数据，TX 信号变为输出（发送状态）。当发送完成，发送缓存变为空，TX 信号又回到输入（接收状态）。

注意：

- 即使模块处于接收过程中，只要发送缓存被填入数据，TX 信号立即变为输出（发送状态）。接收过程将被打断，并得到一个错误的接收数据。

单线半双工模式并不是 Mode0 模式，虽然两者都是用一根信号分时复用地完成了数据的收发功能。但是 Mode0 为同步传输，有同步时钟。而单线半双工模式为异步传输，所有的波特率计算还是遵照 Mode1-2-3 的波特率公式，仅仅是收发功能复用到了 TX 这一根信号上面。

23.5 波特率编程

23.5.1 Mode0

当工作在 Mode0 时，LPUART 同步移位时钟频率固定为 SCLK 的 1/12。具体公式如下所示：

$$BaudRate = \frac{Freq}{12}$$

其中，Freq 为 SCLK 的时钟频率。

23.5.2 Mode1/3

当工作在 Mode1 或者 Mode3 时，波特率由可编程波特率发生器产生。具体公式如下所示：

$$BaudRate = \frac{Freq}{OVER * SCNT}$$

其中，Freq 为 SCLK 的时钟频率；OVER 可以为 16、8 或者 4，由 LPUARTx_SCON.OVER 的值决定；SCNT 为 16-Bit 波特率计数器 LPUARTx_SCNT 的计数值。

23.5.3 Mode2

当工作在 Mode2 时，波特率为固定值，具体公式如下所示：

$$BaudRate = \frac{Freq}{OVER}$$

其中，Freq 为 SCLK 的时钟频率；OVER 可以为 32、16 或者 8，由 LPUARTx_SCON.OVER 的值决定。

23.6 传输数据结构

在不同工作模式下，传输数据的组成有所不同，请参考表 93：

表 93: 传输数据组成结构

工作模式	传输位宽	数据组成
Mode0	8-Bit	8-Bit Data
Mode1	10-Bit	1-Bit Start Bit + 8-Bit Data + (1/1.5/2)-Bit Stop Bit
Mode2	11-Bit	1-Bit Start Bit + 8-Bit Data + 1-Bit B8 + (1/1.5/2)-Bit Stop Bit
Mode3	11-Bit	1-Bit Start Bit + 8-Bit Data + 1-Bit B8 + (1/1.5/2)-Bit Stop Bit

B8 数据位比较特殊，在不同应用下具有不同的含义，请参考表 94：

表 94: B8 数据含义

应用场景	LPUARTx_SCON.ADRDET	LPUARTx_SCON.B8CONT[1:0]	B8 数据含义
奇偶校验	–	01/10	接收时，B8 是所收到的 8-Bit 数据的奇偶校验位； 发送时，B8 是所发送的 8-Bit 数据的奇偶校验位；
多机通讯	1	–	B8=1，代表当前是地址帧；B8=0，代表当前是数据帧；
其他	0	00/11	接收/发送的 DATA[8]

注意：

-当开启多机通讯模式，接收数据奇偶校验自动关闭；发送数据奇偶校验仍受 B8CONT 控制。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

23.7 帧错误检测

当工作在 Mode1/2/3 时，LPUART 具有帧错误检测功能，硬件会自动检测接收到的帧数据是否带有效的 Stop 位。如果没有收到有效 Stop 位，则 LPUARTx_ISR.FE 置 1。LPUARTx_ISR.FE 位由硬件置 1，软件清 0，如果软件未及时清 0，则后续收到数据即使带有效 Stop 位，也不会把 LPUARTx_ISR.FE 标志清 0。

23.8 多机通讯

当工作在 Mode2/3 时，可以打开多机通讯功能，为此在其帧格式中增加了 1 位 TB8/RB8。将 LPUARTx_SCON.ADRDET 位置“1”，可开启多机通讯功能。

当开启多机通讯后，主机可以通过 LPUARTx_SBUF[8] 来区分当前的发送帧是地址帧（LPUARTx_SBUF[8]=1）还是数据帧（LPUARTx_SBUF[8]=0）。从机会忽略 RB8 位（接收到的 Bit8 数据位）为“0”的接收帧，该帧数据不会存入到从机的 LPUARTx_SBUF 寄存器中，从机也不会产生接收中断。当接收帧的 RB8 位（接收到的 Bit8 数据位）为“1”表明其是地址帧，从机会继续判断接收到的地址与其自身地址是否相等。如果匹配，则从机会对 LPUARTx_SBUF[8] 置“1”，同时将该地址帧存入到 LPUARTx_SBUF 寄存器中。并对 LPUARTx_ISR.RC 置“1”，以表明该帧为地址帧并且地址已经匹配。从机软件看到 LPUARTx_SBUF[8]=1 并且 LPUARTx_ISR.RC=1 后，先把 LPUARTx_SCON.ADEDET 位清“0”，然后准备接受给它的数据帧。如果地址不等，表明主机并不是寻址该从机，从机硬件保持 LPUARTx_SBUF[8] 和 LPUARTx_ISR.RC 为“0”，软件保持 LPUARTx_SCON.ADRDET 位为“1”，从机继续处于地址监听状态。

23.8.1 自动地址识别

当开启多机通讯后（LPUARTx_SCON.ADRDET 位置“1”），自动地址识别功能也将开启。该功能由硬件实现，使得从机可以检测接收到每个地址帧，如果该地址与从机地址匹配，接收从机会给出 LPUARTx_ISR.RC 接收标志。如果地址不匹配，则接收从机不会给出任何接收标志。

如果有需要，也可以在 Mode1 下开启多机通讯位，此时 TB8 位由 Stop 位代替。当从机接收到匹配的地址帧和有效的 Stop 位时，LPUARTx_ISR.RC 会被置“1”。

23.8.2 给定地址

LPUART 设备的 LPUARTx_SADDR 寄存器用来表示自己的设备给定地址，LPUARTx_SADEN 寄存器是地址掩码，可以用来定义地址中的无关位。当 LPUARTx_SADEN 的某一位为“0”，表示该位地址为无关位，也就是说在地址匹配过程中，该位地址不参与地址匹配。这些无关位增加了寻址的灵活性，使得主机可以同时寻址一个或者多个从机设备。注意，如果需要给出唯一匹配地址，LPUARTx_SADEN 寄存器必须设为 8’hFF。给定地址公式如下所示：

GivenAddr = SADDR & SADEN

23.8.3 广播地址

广播地址是用来同时寻址所有从机设备的，一般广播地址为 8’hFF。

BroadCastAddr = SADDR | SADEN

23.8.4 举例

假设某从机的 LPUARTx_SADDR 和 LPUARTx_SADEN 配置如下：

SADDR：8’b01101001

SADEN：8’b11111011

那么其给定地址和广播地址如下：

Given：8’b01101x01

Broadcast：8’b11111x11

可见，主机可以用四个地址寻址到本从机，分别是：

8’b01101001 和 8’b01101101 (given address)

8’b11111011 和 8’b11111111 (broadcast address)。

23.9 DMAC 硬件握手

LPUART 模块支持 DMAC 的硬件握手逻辑。

将 LPUARTx_SCON.DMACTXEN 设置为 1，可以打开 LPUART TX 的 DMAC 硬件握手逻辑。当发送缓存为空时，LPUART 会向 DMAC 发出数据搬运请求 TX REQ。DMAC 收到该 TX REQ 信号，则从 DMAC 源地址搬运一个帧的发送数据到 LPUARTx_SBUF 中。上述步骤重复发生，直到 DMAC 中所配置的数据长度全部搬运完毕。

将 LPUARTx_SCON.DMACRXEN 设置为 1，可以打开 LPUART RX 的 DMAC 硬件握手逻辑。当一帧接收完成，LPUART 会向 DMAC 发出数据搬运请求 RX REQ。DMAC 收到该 RX REQ 信号，则从 LPUARTx_SBUF 中把接收数据搬运至 DMAC 的目标地址中。上述步骤重复发生，直到 DMAC 中所配置的数据长度全部搬运完毕。

23.10 硬件流控

通过增加 nCTS 和 nRTS 信号可以实现 LPUART 硬件流控的功能，即 LPUART 硬件模块根据 nCTS 和 nRTS 的高低电平自动控制数据的收发，而无需通过软件来判断。两个 LPUART 模块之间的硬件流控示意图如下所示：

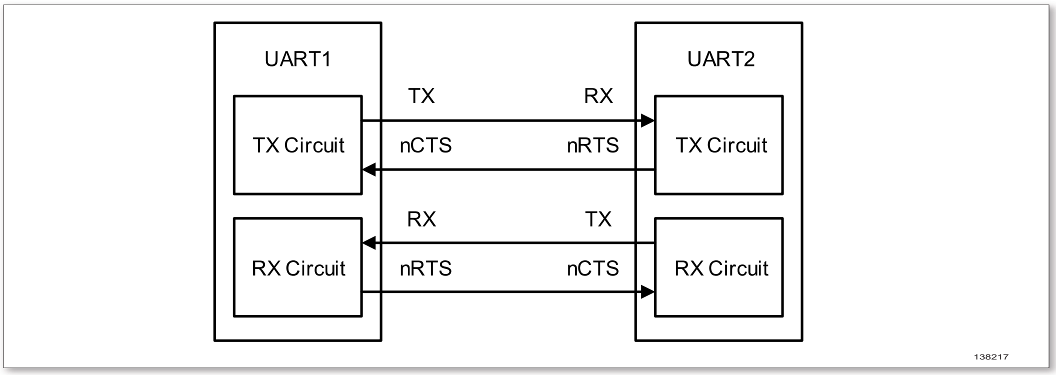

图 177. LPUART 硬件流控

23.10.1 nRTS 流控

当 nRTS 流控使能时（LPUARTx_SCON.RTSEN 设置为 1），当 LPUART 接收缓存空时，nRTS 信号拉低（低表示可以接收数据），通知对端设备可以发送下一帧数据。当接收缓存满时，nRTS 信号拉高（高表示不可以接收数据），通知对端设备在当前发送帧传输完成后，暂停发送下一帧数据。

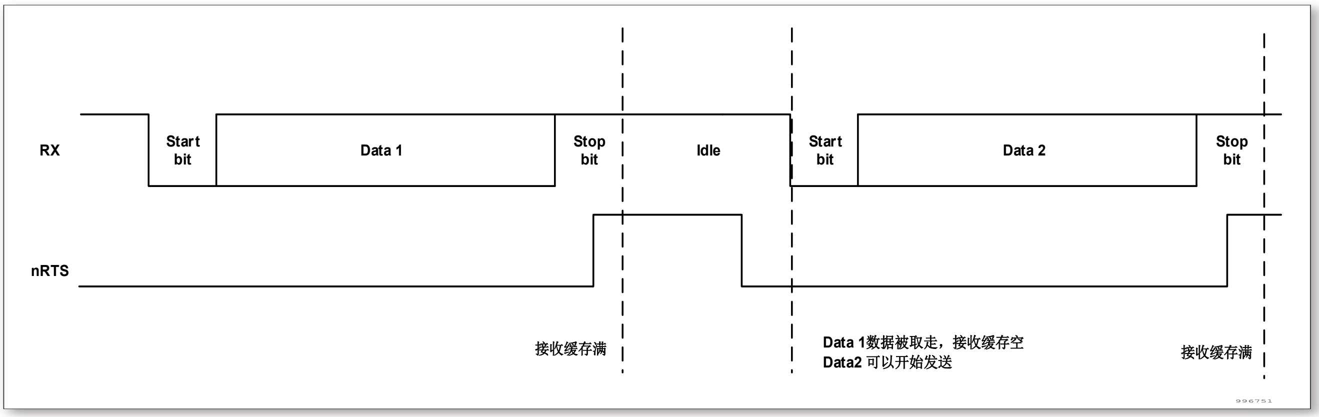

图 178. nRTS 硬件流控信号

23.10.2 CTS 流控

当 nCTS 流控使能时（LPUARTx_SCON.CTSEN 设置为 1），在 LPUART 发送下一帧数据之前，先判断 nCTS 的高低电平。如果 nCTS 为低电平（低电平表示对端设备可以接收下一帧数据），则 LPUART 发送下一帧数据。如果 nCTS 为高电平（高电平表示对端设备不可以接收下一帧数据），则 LPUART 暂停发送下一帧数据。nCTS 拉高不会影响当前发送帧的传输，只会暂停下一帧的发送，直到 nCTS 重新变低为止。当 nCTS 流控使能时，一旦 nCTS 信号发生翻转，会把 LPUARTx_SFLAG.CTSIF 置 1。如果使能 CTSIE 中断（LPUARTx_SCON.CTSIE 设置为 1），则会产生中断。同时，nCTS 信号的高低电平会记录在 LPUARTx_SFLAG.CTS 标志位中。

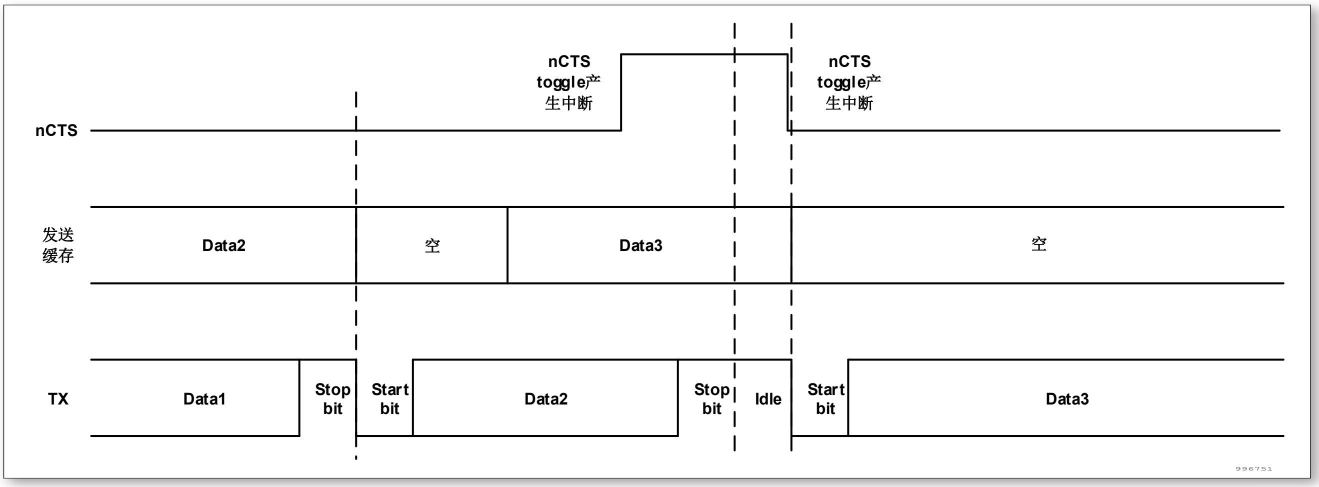

图 179. nCTS 硬件流控信号

23.11 收发端缓存

23.11.1 接收缓存

LPUART 模块接收端有一个帧（8/9-Bit）的接收缓存，也就是说当一帧数据接收完毕后，接收缓存中的数据会被一直保持，直到下一帧数据的 Stop 位接收完毕后，接收缓存才会更新到这一帧所收到的数据。

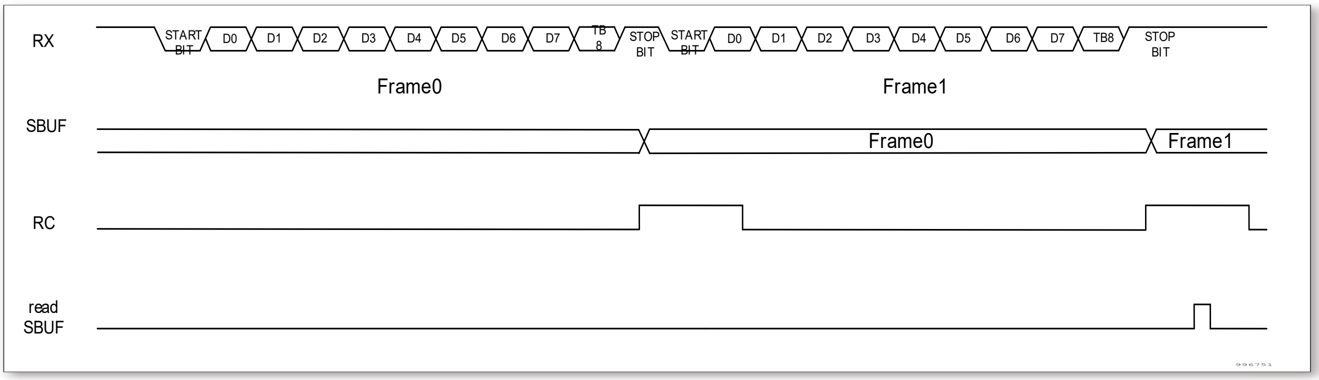

图 180. 接收缓存

23.11.2 发送缓存

LPUART 模块发送端有一个帧 (8/9-Bit) 的发送缓存，当 LPUART 在发送当前帧时，软件可以对 LPUARTx_SBUF 写入下一个发送数据。LPUARTx_ISR.TXE 位表示了当前发送缓存的空满状态，当 LPUARTx_ISR.TXE=0 时，表明当前发送缓存满，软件不能对 LPUARTx_SBUF 写入下一个发送数据。如果软件继续写入 LPUARTx_SBUF，则该笔写入数据会硬件丢弃。当 LPUARTx_ISR.TXE=1 时，表明当前发送缓存空，软件可以写入下一个需要发送的数据帧。该笔发送数据会保存在发送缓存中，直到当前发送数据完成，硬件自动把发送缓存中的数据装载入移位寄存器中发送出去。

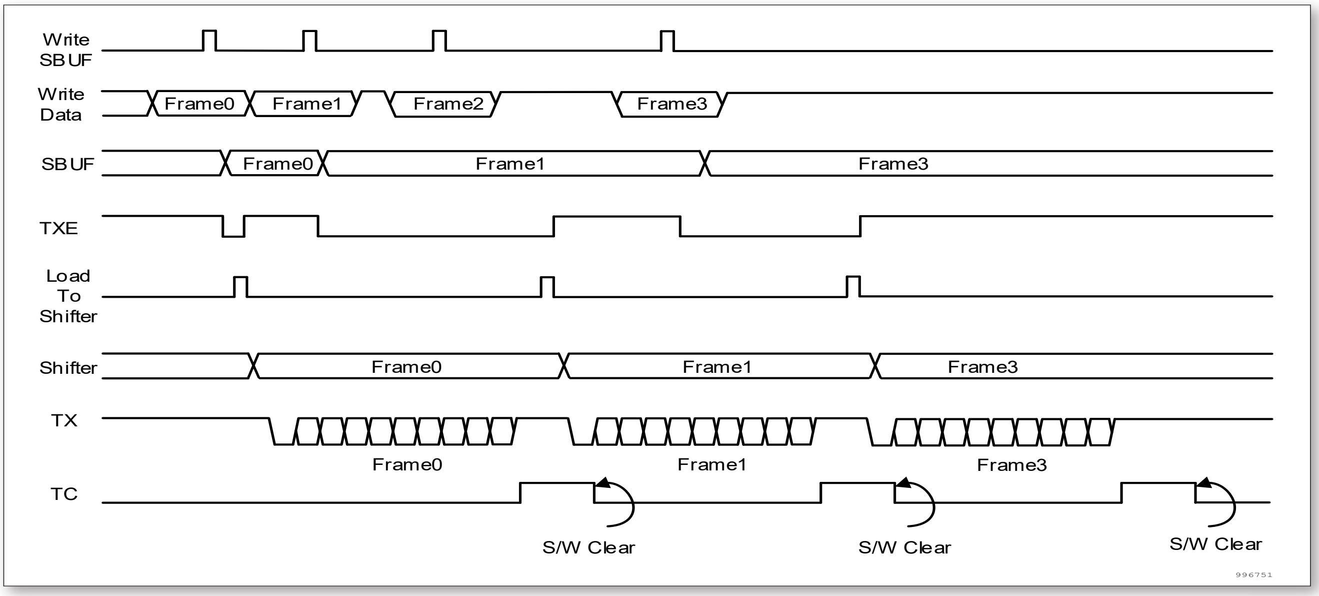

图 181. 发送缓存

23.12 LPUART 寄存器

表 95. LPUART 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x00	LPUARTx_SBUF	数据寄存器	0x0000 0000	小节 23.12.1
0x04	LPUARTx_SCON	控制寄存器	0x0000 0000	小节 23.12.2
0x08	LPUARTx_SADDR	地址寄存器	0x0000 0000	小节 23.12.3

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Offset	Acronym	Register Name	Reset	Section
0x0C	LPUARTx_SADEN	地址掩码寄存器	0x0000 0000	小节 23.12.4
0x10	LPUARTx_ISR	中断标志位寄存器	0x0000 0008	小节 23.12.5
0x14	LPUARTx_ICR	中断标志位清除寄存器	0x0000 0000	小节 23.12.6
0x18	LPUARTx_SCNT	波特率寄存器	0x0000 0000	小节 23.12.7

23.12.1 数据寄存器 (LPUARTx_SBUF)

起始地址：LPUART0：0x40000200 LPUART1：0x40004000

地址偏移：0x00

复位值：0x0000 0000

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

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

Bit	Field	Type	Reset	Description
31:9	Reserved			保留
8	DATA[8]		0x00	在 Mode0/1 下，读取该位为 0，写入该位无效； 在 Mode2/3 下，该位表示 Bit8 数据位，分以下两种情况： （1）当硬件奇偶校验位开启时，接收时该位为接收数据奇偶校验位，校验由硬件进行，如校验出错，校验错误标志位 PE 置 1；发送时该位无效，发送数据奇偶校验位由硬件计算并发送； （2）当硬件奇偶校验位关闭时，接收时该位为接收数据 Bit8；发送时该位为发送数据 Bit8； 注意：当开启多机通讯模式，接收数据奇偶校验自动关闭；发送数据奇偶校验仍受 B8CONT 控制；
7:0	DATA[7:0]	rw	0x00	发送数据时，将发送数据写入该寄存器；接收数据时，数据接收完毕后，从该寄存器中读取。

23.12.2 控制寄存器 (LPUARTx_SCON)

起始地址：LPUART0：0x40000200 LPUART1：0x40004000

地址偏移：0x04

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved									HSEL	FEIE	CTSIE	CTSEN	RTSEN	DMATXEN	DMARXEN
									RW	RW	RW	RW	RW	RW	RW

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
STOPBIT		PEIE	SCLKSEL			OVER	TXEIE	SM		ADRDET	REN	B8CONT		TCIE	RCIE
RW		RW	RW			RW	RW	RW		RW	RW	RW		RW	RW

Bit	Field	Type	Reset	Description
31:23	Reserved			保留
22	HDSEL	rw	0x00	单线半双工模式是能；0: 关闭单线半双工模式；1：打开单线半双工模式；
21	FEIE	rw	0x00	帧错误中断使能；0：关闭中断；1：打开中断；
20	CTSIE	rw	0x00	CTS 信号翻转中断使能；0：关闭中断；1：打开中断；
19	CTSEN	rw	0x00	硬件 flow control 信号使能位；0：关闭 flow control 信号；1：打开 flow control 信号；
18	RTSEN	rw	0x00	硬件 flow control 信号使能位；0：关闭 flow control 信号；1：打开 flow control 信号；
17	DMATXEN	rw	0x00	TX DMAC 的硬件握手信号使能位；0：关闭硬件握手信号；1：打开硬件握手信号；
16	DMARXEN	rw	0x00	RX DMAC 的硬件握手信号使能位；0：关闭硬件握手信号；1：打开硬件握手信号；
15:14	STOPBIT	rw	0x00	stop bit 长度选择；00:1-bit；01:1.5-bit；10:2-bit；11：reserved； 注意：Mode0 时虽然没有 Stop Bit，但仍需把 STOPBIT[1:0] 保持为 2’b00；
13	PEIE	rw	0x00	奇偶校验错误中断使能位；0：奇偶校验错误中断关闭；1：奇偶校验错误中断打开； 在系统时钟使用高速时钟与 LPUART 使用低速时钟情况下，由于需要不同时钟域的同步，硬件奇偶校验与数据接收中断由于停止位设置不同，奇偶校验错误中断会提前或落后于数据接收中断。使用奇偶校验中断使能时请注意。 建议如下： 关闭 PEIE 中断使能，接收数据中断后通过接收的数据位 SBUF.BIT8 软件判断是否出现奇偶校验错误。
12:11	SCLKSEL	rw	0x00	传输时钟选择位：00，01：PCLK；10：XTL；11：RCL；

ZLG
©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
10:9	OVER	rw	0x00	Mode0：无效； Mode1/3：00：16 采样分频；01：8 采样分频；10：4 采样分频；11：保留； Mode2：00：32 采样分频；01：16 采样分频；10：8 采样分频；11：保留；
8	TXEIE	rw	0x00	TX 空中断使能位；0：TX Buffer 空中断关闭；1：TX Buffer 空中断打开；
7:6	SM	rw	0x00	工作模式；00：mode0；01：mode1；10：mode2；11：mode3；
5	ADRDET	rw	0x00	多机通讯地址自动识别使能位；0：关闭；1：打开；
4	REN	rw	0x00	Mode0：0：发送；1：接收； Mode1/2/3：0：发送；1：接收/发送；
3:2	B8CONT	rw	0x00	Bit8 数据控制位； 00：由软件读写 SBUF[8] 来决定；01：硬件偶校验； 10：硬件奇校验；11：保留；
1	TCIE	rw	0x00	发送中断使能位；0：发送中断关闭；1：发送中断打开；
0	RCIE	rw	0x00	接收中断使能位；0：接收中断关闭；1：接收中断打开；

23.12.3 地址寄存器（LPUARTx_SADDR）

起始地址：LPUART0：0x40000200　LPUART1：0x40004000

地址偏移：0x08

复位值：0x0000 0000

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

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

Bit	Field	Type	Reset	Description
31:8	Reserved			保留
7:0	SADDR	rw	0x00	从机设备地址寄存器

23.12.4 地址掩码寄存器（LPUARTx_SADEN）

起始地址：LPUART0：0x40000200　LPUART1：0x40004000

地址偏移：0x0C

复位值：0x0000 0000

ZLG
©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

Bit	Field	Type	Reset	Description
31:8	Reserved			保留
7:0	SADEN	rw	0x00	从机设备地址掩码寄存器

23.12.5 中断标志位寄存器（LPUARTx_ISR）

起始地址：LPUART0：0x40000200　LPUART1：0x40004000

地址偏移：0x10

复位值：0x0000 0008

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved									CTS	CTSIF	PE	TXE	FE	TC	RC
									RO	RO	RO	RO	RO	RO	RO

Bit	Field	Type	Reset	Description
31:7	Reserved	ro	0x00	保留
6	CTS	ro	0x00	CTS 信号标志位；硬件置 1；硬件清零；0：CTS 信号为低电平；1：CTS 信号为高电平；
5	CTSIF	ro	0x00	CTS 中断标志位；硬件置 1；软件清零；0：CTS 信号没有发生反转；1：CTS 信号发生反转；
4	PE	ro	0x00	奇偶校验错误标志位；硬件置 1；软件清零；0：无奇偶校验错误；1：奇偶校验错误；
3	TXE	ro	0x01	Tx Buffer 空标志位；硬件置 1；硬件清零；0：Tx Buffer 非空；1：Tx Buffer 空
2	FE	ro	0x00	帧错误标志位；0：硬件置 1；软件清零；
1	TC	ro	0x00	发送完毕中断标志位；硬件置 1；软件清零；0：发送未完成；1：发送完毕；
0	RC	ro	0x00	接收完毕中断标志位；硬件置 1；软件清零；0：接收未完成；1：接收完成；

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

23.12.6 中断标志位清除寄存器 (LPUARTx_ICR)

起始地址：LPUART0：0x40000200 LPUART1：0x40004000

地址偏移：0x14

复位值：0x0000 0000

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="10" rowspan="2">Reserved</td><td>CTSI FCF</td><td>PECF</td><td rowspan="2">Res.</td><td>FECF</td><td>TCCF</td><td>RCCF</td></tr>
<tr><td>R1W0</td><td>R1W0</td><td>R1W0</td><td>R1W0</td><td>R1W0</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:6	Reserved			保留
5	CTSIFCF	rw	0x00	CTSIF 标志清楚位；写 0 清除；写 1 无效；
4	PECF	rw	0x00	PE 标志清楚位；写 0 清除；写 1 无效；
3	Reserved			保留
2	FECF	rw	0x00	FE 标志清除位；写 0 清零；写 1 无效；
1	TCCF	rw	0x00	TC 标志清除位；写 0 清零；写 1 无效；
0	RCCF	rw	0x00	RC 标志清除位；写 0 清零；写 1 无效；

23.12.7 波特率寄存器 (LPUARTx_SCNT)

起始地址：LPUART0：0x40000200 LPUART1：0x40004000

地址偏移：0x18

复位值：0x0000 0000

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="16">SCNT</td></tr>
<tr><td colspan="16">RW</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:0	SCNT	rw	0x00	波特率计数器

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

24 循环冗余校验 (CRC)

24.1 概述

在许多应用中，都需要用 CRC 算法来校验数据的完整性和正确性。尤其是在数据传输中，CRC 校验更是被广泛运用。本模块实现 CRC16 和 CRC32 两种算法对数据进行运算和校验。

24.2 功能框图

图 182示意了 CRC 算法在数据传输中的一个最典型的应用：

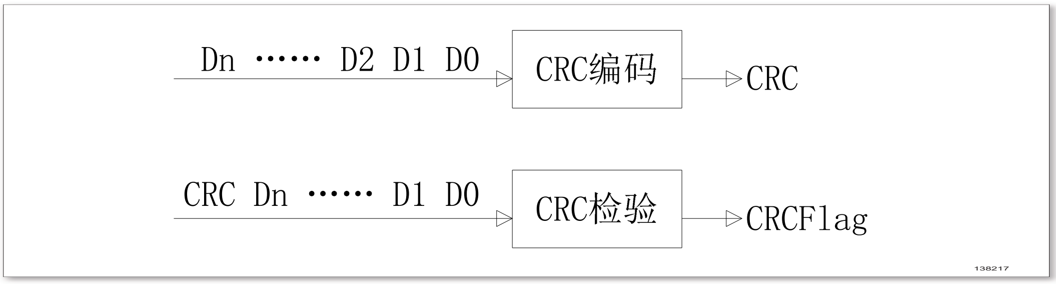

图 182. CRC 应用示意图

24.3 功能描述

本模块 CRC 算法遵从 ISO/IEC13239 的定义，分别采用 32 位和 16 位的 CRC。其中，CRC32 的生成多项式为 $x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^{8}+x^{7}+x^{5}+x^{4}+x^{2}+x+1$,32 位初值为“0xFFFFFFFF”。CRC16 的生成多项式为 $x^{16}+x^{12}+x^{5}+1$,16 位初值为“0xFFFF”。

本模块功能包括：

- CRC 编码和 CRC 校验；
- 3 种位宽访问方式 8 位、16 位、32 位。

  8 位位宽下输入数据示例为 0x00, 0x11, 0x22 , 0x33, 0x44, 0x55, 0x66, 0x77；

  16 位位宽下输入数据示例为 0x1100, 0x3322, 0x5544, 0x7766；

  32 位位宽下输入数据示例为 0x33221100, 0x77665544。par

24.4 CRC 寄存器

表 96. CRC 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x00	CRC_CR	CRC 控制寄存器	0x0000 0001	小节 24.4.1
0x04	CRC_RESULT	CRC 结果寄存器	0x0000 0000	小节 24.4.2
0x80-0xFF	CRC_DATA	CRC 数据寄存器	0x0000 0000	小节 24.4.3

24.4.1 CRC 控制寄存器 (CRC_CR)

起始地址：0x40020900

地址偏移：0x00

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

复位值：0x0000 0001

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

Bit	Field	Type	Reset	Description
31:2	Reserved			保留
1	FLAG	ro	0x00	CRC 校验结果标志位；0：当前校验错误；1：当前校验正确；
0	CR	rw	0x01	CRC 功能选择；0：CRC16；1：CRC32

24.4.2 CRC 结果寄存器 (CRC_RESULT)

起始地址：0x40020900

地址偏移：0x04

复位值：0x0000 0000

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

Bit	Field	Type	Reset	Description
31:0	RESULT	rw	0x00	CRC 结果寄存器；选择 CRC16 时，取 RESULT[15:0]；选择 CRC32 时，取 RESULT[31:0]；

24.4.3 CRC 数据寄存器 (CRC_DATA)

起始地址：0x40020900

地址偏移：0x80-0xFF

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

Bit	Field	Type	Reset	Description
31:0	DATA	wo	0x00	本寄存器用于输入需要运算的数据;本寄存器的地址是一个范围（0x80-0xFF），对该地址范围内的任何地址进行操作，都会认为对本寄存器进行操作。这样定义的目的就是方便软件可以用 STM 指令对本寄存器进行连续的 32 位数据写入操作，以加快运算速度。同时本寄存器也支持 8/16 位的输入方式。 需要特别注意的是，无论是计算还是检验，对本寄存器进行的数据写入操作，必须符合“位宽一致、先低后高”的原则，即“每次写入数据，都必须写入到与本次有效数据位宽相等的寄存器中，并且较低位数据的写入先于较高位数据”。举例如下： 如果待写入数据为 0xFEDCBA9876543210，那么 按字节（单次 8 位）写入时： *((unsigned char *) 0x40020980) = 0x10 ; *((unsigned char *) 0x40020981) = 0x32 ; …… *((unsigned char *) 0x40020986) = 0xDC ; *((unsigned char *) 0x40020987) = 0xFE ; 按半字（单次 16 位）写入时： *((unsigned short *) 0x40020980) = 0x3210 ; *((unsigned short *) 0x40020982) = 0x7654 ; *((unsigned short *) 0x40020984) = 0xBA98 ; *((unsigned short *) 0x40020986) = 0xFEDC ; 按全字（单次 32 位）写入时： *((unsigned int *) 0x40020980) = 0x76543210 ; *((unsigned int *) 0x40020984) = 0xFEDCBA98 ;

24.5 软件基本操作

24.5.1 CRC16 编码模式

编码模式可以对原始数据编码以计算其 CRC 值，操作流程如下所示：

Step 1：向 CRC_CR.CR 写入 1’b0，选择 CRC16。

Step 2：向 CRC_RESULT[15:0] 写入 0xFFFF，初始化 CRC 计算。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Step 3：将待编码的原始数据按 8 位/16 位/32 位的组织方式，依次写入 CRC_DATA 寄存器。（注：数据写入的具体方式参见 CRC_DATA 寄存器的说明）

Step 4：读取 CRC_RESULT[15:0]，即为 CRC 值。

Step 5：如果需要计算其他数据，重复 step1-step4。

24.5.2 CRC16 检验模式

检验模式可以检验已编码的数据是否被篡改，操作流程如下所示：

Step 1：向 CRC_CR.CR 写入 1’b0，选择 CRC16。

Step 2：向 CRC_RESULT[15:0] 写入 0xFFFF，初始化 CRC 计算。

Step 3：将已编码的数据按 8 位/16 位/32 位的组织方式，依次写入 CRC_DATA 寄存器。（注：数据写入的具体方式参见 CRC_DATA 寄存器的说明）

Step 4：读取 CRC_CR.FLAG，以判定编码的数据是否被篡改。当 CRC_FLAG=0，表示校验失败；当 CRC_FLAG=1，表示校验成功。

Step 5：如果需要检验其他数据，重复 step1-step4。

24.5.3 CRC32 编码模式

编码模式可以对原始数据编码以计算其 CRC 值，操作流程如下所示：

Step 1：向 CRC_CR.CR 写入 1，选择 CRC32。

Step 2：向 CRC_RESULT[31:0] 写入 0xFFFF FFFF，初始化 CRC 计算。

Step 3：将待编码的原始数据按 8 位/16 位/32 位的组织方式，依次写入 CRC_DATA 寄存器。（注：数据写入的具体方式参见 CRC_DATA 寄存器的说明）

Step 4：读取 CRC_RESULT[31:0]，即为 CRC 值。

Step 5：如果需要计算其他数据，重复 step1-step4。

24.5.4 CRC32 检验模式

检验模式可以检验已编码的数据是否被篡改，操作流程如下所示：

Step 1：向 CRC_CR.CR 写入 1，选择 CRC32。

Step 2：向 CRC_RESULT[31:0] 写入 0xFFFF FFFF，初始化 CRC 计算。

Step 3：将已编码的数据按 8 位/16 位/32 位的组织方式，依次写入 CRC_DATA 寄存器。（注：数据写入的具体方式参见 CRC_DATA 寄存器的说明）

Step 4：读取 CRC_CR.FLAG，以判定编码的数据是否被篡改。当 CRC_CR.FLAG=0，表示校验失败；当 CRC_CR.FLAG=1，表示校验成功。

Step 5：如果需要检验其他数据，重复 step1-step4。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

25 真随机数发生器 (TRNG)

25.1 概述

真随机数模块产生 64 位真随机数。

25.2 功能框图

图 183示意了 TRNG 模块的数据流：

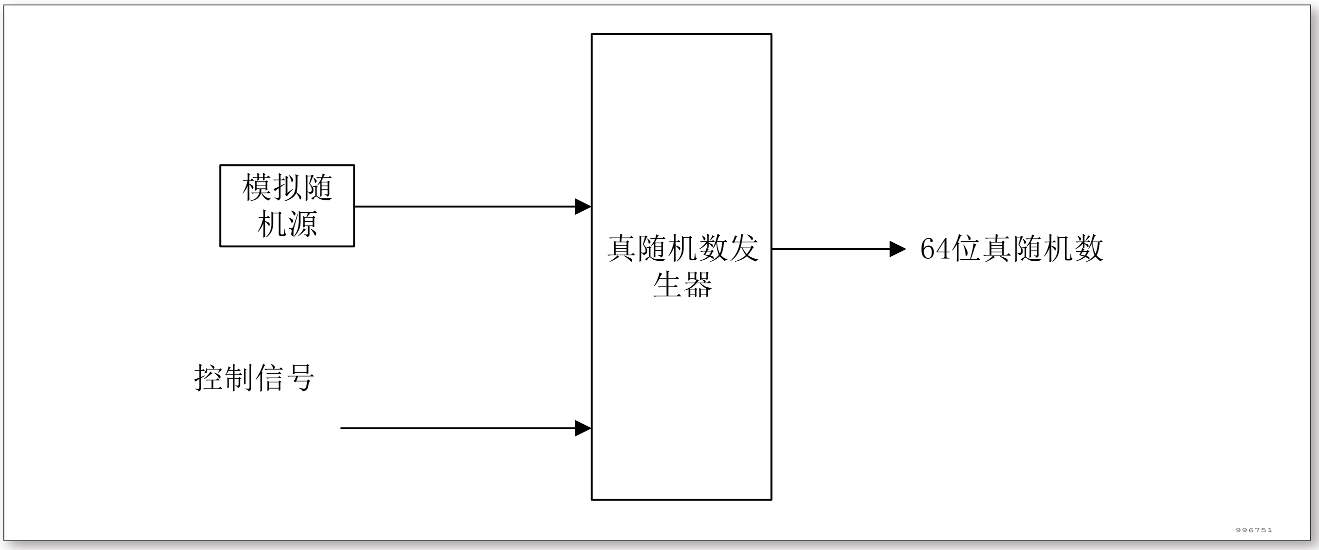

图 183. TRNG 数据流

25.3 功能描述

本模块采用内部的模拟随机源，每次启动都可以产生 64bits 真随机数。此外，还可以对真随机数生成的方式进行软件配置，详细内容可查看寄存器描述章节。生成的 64 位真随机数分别存放在 DATA0 和 DATA1 寄存器中。

25.4 TRNG 寄存器

表 97. TRNG 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x00	TRNG_CR	控制寄存器	0x0000 0000	小节 25.4.1
0x04	TRNG_MODE	模式寄存器	0x0000 0010	小节 25.4.2
0x0C	TRNG_DATA0	数据寄存器 0	0xXXXX XXXX	小节 25.4.3
0x10	TRNG_DATA1	数据寄存器 1	0xXXXX XXXX	小节 25.4.4

25.4.1 控制寄存器 (TRNG_CR)

起始地址：0x40004C00

地址偏移：0x00

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

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

Bit	Field	Type	Reset	Description
31:2	Reserved			保留
1	RNG_RUN	rw	0x00	软件写入“1”，开始产生新的 64bits 随机数；运行完毕后，硬件清零； 0：随机数产生完成； 1：写 1 启动随机数产生，读 1 表示随机数正在产生；
0	RNGcir_EN	rw	0x00	随机源电路使能位： 0：关闭随机源； 1：打开随机源；

25.4.2 模式寄存器 (TRNG_MODE)

起始地址：0x40004C00

地址偏移：0x04

复位值：0x0000 0010

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved											RNG_CNT			RNG_FDBK	RNG_LOAD
											RW			RW	RW

Bit	Field	Type	Reset	Description
31:5	Reserved			保留

Bit	Field	Type	Reset	Description
4:2	RNG_CNT	rw	0x04	64bits RNG 的反馈移位次数 3’b000: 移位 0 次（即输出随机源的采样值） 3’b001: 移位 8 次 3’b010: 移位 16 次 3’b011: 移位 32 次 3’b100: 移位 64 次 3’b101: 移位 128 次 3’b110: 移位 256 次 3’b111:Reserved
1	RNG_FDBK	rw	0x00	在移位操作时，64bits RNG 的反馈信号是否与随机源进行异或操作 0：不进行异或操作； 1：进行异或操作；
0	RNG_LOAD	rw	0x00	在产生新的随机数时，64bits RNG 是否从随机源获得新的初始值 0：不装载新的初始值（产生伪随机数）； 1：装载新的初始值（产生真随机数）；

25.4.3 数据寄存器 0 (TRNG_DATA0)

起始地址：0x40004C00

地址偏移：0x0C

复位值：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA0[31:16]															
RO															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA0[15:0]															
RO															

Bit	Field	Type	Reset	Description
31:0	DATA0	ro	0xXX	软件对本寄存器读取将得到低 32 位的随机数

25.4.4 数据寄存器 1 (TRNG_DATA1)

起始地址：0x40004C00

地址偏移：0x10

复位值：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA1[31:16]															
RO															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA1[15:0]															
RO															

Bit	Field	Type	Reset	Description
31:0	DATA1	ro	0xXX	软件对本寄存器读取将得到高 32 位的随机数

25.5 软件基本操作

25.5.1 生成 64bits 真随机数的操作流程（上电第一次）

上电第一次生成 64bits 真随机数，需要经过以下操作：

Step1：打开随机源电路：对真随机数控制寄存器的 Bit0（RNG_CR.RNGcir_En）写入“1”，启动随机源电路，开始输出串行真随机数。

Step2：选择重新装载初始值：将真随机数模式寄存器的 Bit0 （RNG_MODE. RNG_LOAD）设置为“1”，使新产生的真随机数的初始值从随机源获得。

Step3：选择 PRNG64 的直接反馈的方式：将真随机数模式寄存器的 Bit1（RNG_MODE. RNG_FDBK）设置为“1”，将反馈信号与随机源异或后输入 PRNG 中。

Step4：选择 PRNG64 的移位次数：设置真随机数模式寄存器的 Bit4 — Bit2 （RNG_MODE. RNG_CNT）为“110”，选择移位 256 次。

Step5：生成真随机数：软件将“1”写入真随机数控制寄存器的 Bit1（RNG_CR. RNG_RUN），硬件根据真随机数生成配置进行操作，在操作完成后，硬件自动将 Bit1 清为“0”。

Step6：选择不重新装载初始值：将真随机数模式寄存器的 Bit0（RNGModeReg. RNG_Load）设置为“0”。

Step7：选择 PRNG64 的直接反馈的方式：将真随机数模式寄存器的 Bit1（RNG_MODE. RNG_FDBK）设置为“0”，将反馈信号直接输入 PRNG 中。

Step8：选择 PRNG64 的移位次数：设置真随机数模式寄存器的 Bit4 — Bit2 （RNG_MODE.RNG_CNT）为“100”，选择移位 64 次。

Step9 ：生成真随机数：软件将“1 ”写入真随机数控制寄存器的 Bit1 （RNG_CR.RNG_RUN），硬件根据真随机数生成配置进行操作，在操作完成后，硬件自动将 Bit1 清为“0”。

Step10：读取真随机数：软件在查询到真随机数控制寄存器的 Bit1（RNG_CR.RNG_RUN）变为“0”后，通过读取真随机数数据寄存器 0(RNG_DATA0) 和真随机数数据寄存器 1(RNG_DATA1)，得到 64Bits 真随机数。

Step11:完成真随机数的生成后,推荐选择关闭随机源电路,节省功耗:对真随机数控制寄存器的 Bit0（RNG_CR. RNGcir_En）写入“0”，关闭随机源电路。

25.5.2 生成 64bits 真随机数的操作流程（非上电第一次生成）

非上电第一次生成 64bits 真随机数，需要经过以下操作：

Step1：打开随机源电路：对真随机数控制寄存器的 Bit0（RNG_CR.RNGcir_En）写入“1”，启动随机源电路，开始输出串行真随机数。

Step2 ：选择不重新装载初始值：将真随机数模式寄存器的 Bit0（RNG_MODE.RNG_LOAD）设置为“0”。

Step3：选择 PRNG64 的直接反馈的方式：将真随机数模式寄存器的 Bit1（RNG_MODE.RNG_FDBK）设置为“1”，将反馈信号与随机源异或后输入 PRNG 中。

Step4：选择 PRNG64 的移位次数：设置真随机数模式寄存器的 Bit4 － Bit2（RNG_MODE.RNG_CNT）为”110”，选择移位 256 次。

Step5 ：生成真随机数：软件将“1 ”写入真随机数控制寄存器的 Bit1 （RNG_CR.RNG_RUN），硬件根据真随机数生成配置进行操作，在操作完成后，硬件自动将 Bit1 清为“0”。

Step6: 选择 PRNG64 的直接反馈的方式：将真随机数模式寄存器的 Bit1 （RNG_MODE.RNG_FDBK）设置为“0”，将反馈信号直接输入 PRNG 中。

Step7：选择 PRNG64 的移位次数：设置真随机数模式寄存器的 Bit4 － Bit2（RNG_MODE.RNG_CNT）为”100”，选择移位 64 次。

Step8：读取真随机数：软件在查询到真随机数控制寄存器的 Bit1 （RNG_CR.RNG_RUN）变为“0”后，通过读取真随机数数据寄存器 0（RNG_Data0）和真随机数数据寄存器 1（RNG_Data1），得到 64Bits 真随机数。如果需要继续生成新的真随机数，那么回到 Step2，直到满足要求。

Step9:完成真随机数的生成后,推荐选择关闭随机源电路,节省功耗:对真随机数控制寄存器的 Bit0（RNG_CR.RNGcir_En）写入“0”，关闭随机源电路。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

26 高级加密标准模块 (AES)

26.1 功能定义

26.1.1 AES 算法简述

AES（The Advanced Encryption Standard）是美国国家标准技术研究所（NIST）在 2000 年 10 月 2 日正式宣布的新的数据加密标准。

AES 的分组长度固定为 128 位，而密钥长度支持 128、192 和 256 位。对于加密来说，其输入是一个明文分组和一个密钥，输出是一个密文分组；对解密而言，输入是一个密文分组和一个密钥，而输出是一个明文分组。此过程如图 184所示：

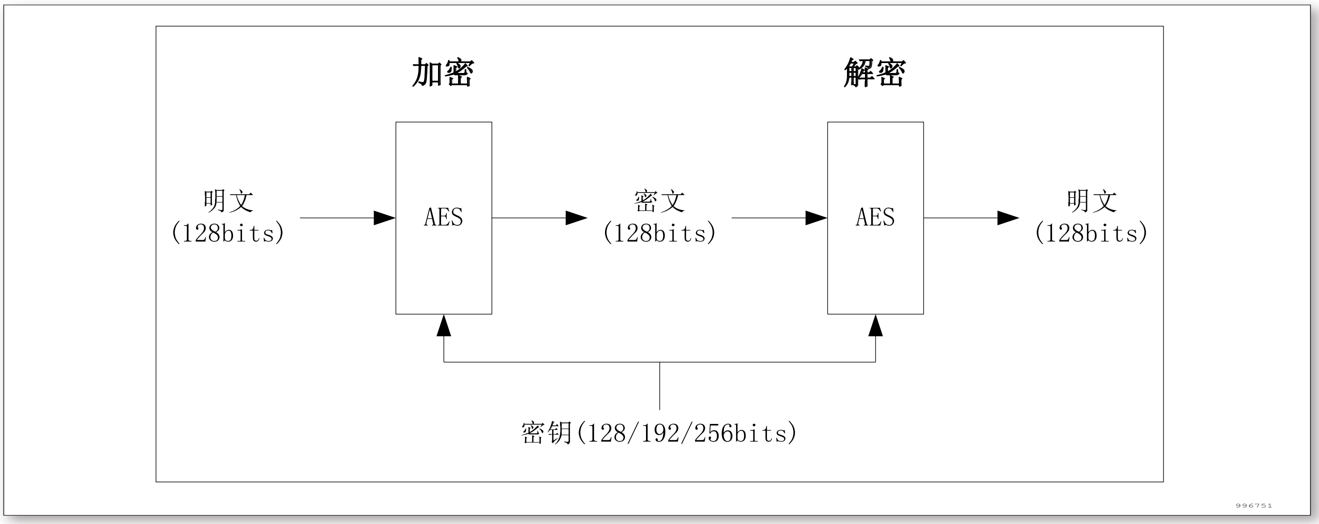

图 184. AES 的加解密示意图

AES 算法处理的基本单位是字节，128 位信息被分成 16 个字节，按顺序复制到一个 4×4 的矩阵中，称为状态（state），AES 的所有变换都是基于状态矩阵的变换，该矩阵上保存着计算的中间结果。AES 是一个密钥迭代分组密码，包含了轮变换对状态的重复作用。AES 的轮变换由四个操作组成：SubBytes、ShiftRows、MixColumns、AddRoundKey。其中，SubBytes 包括求每个字节在 GF(2^8) 中的模逆元和一个仿射变换；ShiftRows 是一个字节换位，它将状态中的行按照不同的偏移量进行循环移位；MixColumns 对状态各列进行线性变换；AddRoundKey，状态中的各字节与轮密钥进行逐位异或操作。AES 的加密流程如图 185所示：

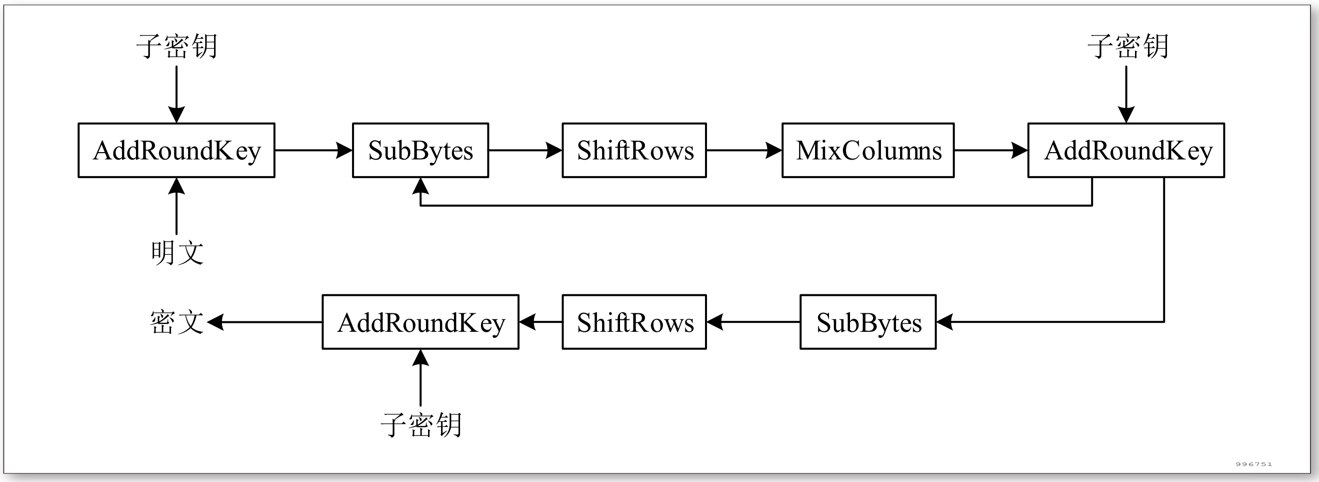

图 185. AES 的加密流程图

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

图中所用子密钥需要由初始密钥扩展而来，且密钥的扩展过程和加密过程是同步进行的。

由于明文固定为 128 位，加密过程运行的轮数就取决于密钥的长度。比如，密钥为 128 位时，运行轮数为 10 轮；密钥为 192 位时，运行轮数为 12 轮；密钥为 256 位时，运行轮数为 14 轮。除了最后一轮缺少 MixColumns 变换，其余各轮均进行完整的轮变换操作。

解密流程与加密流程有所区别，首先必须完成所有密钥的扩展，解密过程从扩展的最后一轮子密钥往回使用；然后是轮变换的四个操作变成了相应的逆运算：InvSubBytes、InvShiftRows、InvMixColumns、AddRoundKey。InvSubBytes 中的模逆运算仍然保持，但仿射变换改为逆变换；InvShiftRows 和 InvMixColumns 变成相应的逆变换；AddRoundKey 保持不变。

直接解密流程的轮变换对四个操作的调用顺序为:InvShiftRows、InvSubBytes、AddRoundKey、InvMixColumns，与加密流程的调用顺序不一致，但使用的密钥与加密流程一致；等价解密流程的轮变换对四个操作的调用顺序为：InvSubBytes、InvShiftRows、InvMixColumns、AddRoundKey，与加密流程的调用顺序完全一致，只是每一轮的子密钥需要进行 InvMixColumns 运算。

关于详细的算法表述，可以参见标准《FIPS PUB 197》

26.1.2 AES 模块功能描述

- 执行 AES 算法标准的加密流程和解密流程，其执行结果完全符合《FIPS PUB 197》对算法原理的描述；
- 支持 128、192 和 256 位密钥。

26.2 AES 寄存器

表 98. AES 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x00 或 0x40	AES_CR	控制寄存器	0x0000 0000	小节 26.2.1
0x10-0x1C	AES_Data	数据寄存器	0x0000 0000	小节 26.2.2
0x20-0x3C	AES_Key	密钥寄存器	0x0000 0000	小节 26.2.3

26.2.1 控制寄存器（AES_CR）

起始地址：0x40021400

地址偏移：0x00 或 0x40

复位值：0x0000 0000

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
<tr><td colspan="16"></td></tr>
</table>

<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="11">Reserved</td><td colspan="2">Keysize</td><td>Reserved</td><td>Mode</td><td>Start</td></tr>
<tr><td colspan="11"></td><td colspan="2">RW</td><td></td><td>RW</td><td>RW</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:5	Reserved			保留位，读为 0
4:3	Keysize	rw	0x00	2'b00：密钥长度为 128 位 2'b01：密钥长度为 192 位 2'b10：密钥长度为 256 位 2'b11：密钥长度为 128 位
2	Reserved			保留位，读为 0
1	Mode	rw	0x00	0：加密运算 1：解密运算
0	Start	rw	0x00	0：本模块运算结束或未被启动 1：启动本模块进行运算

说明：

1. AES_CR.Start 位的操作方法是：软件对本位写入 1 后，本模块将启动运行，本次运行结束后本模块硬件会自动将本位清 0，软件查询到本位为 0 即表示本次运行完成。
2. 对本寄存器的写入操作只能在本模块不处于运算状态时（即 AES_CR.Start = 0 时）才能进行，否则硬件将自动忽略写操作。读操作则不受此限制。

26.2.2 数据寄存器 (AES_Data)

起始地址：0x40021400

地址偏移：0x10-0x1C

复位值：0x0000 0000

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

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

Bit	Field	Type	Reset	Description
31:0	Data	rw	0x00	存放 AES 算法的 128 比特明文/密文

说明：

1. 数据寄存器由四个 32 位的寄存器组成 128 位数据，用于在模块运算前存放需要被加密的明文或者需要被解密的密文，并且运算完成后存放加密后的密文或者解密后的明文。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

表 99

加密运算		解密运算	
运算前	运算后	运算前	运算后
128 位明文	128 位密文	128 位密文	128 位明文

四个 32 位寄存器连接在一起组成一个 128 位的数据，读写操作时需要分别对四个寄存器进行操作。数据寄存器对应的操作顺序如下：

数据举例：0xFFEEDDCCBBAA99887766554433221100

表 100

偏移地址	寄存器名称	填入数据
0x10	AES_Data0	0x33221100
0x14	AES_Data1	0x77665544
0x18	AES_Data2	0xBBAA9988
0x1C	AES_Data3	0xFFEEDDCC

2. 对于本寄存器的写入只能在本模块没有处于运算状态时（即 AES_CR.Start = 0 时）才能进行，否则硬件将自动忽略对本寄存器的写操作。
3. 对于本寄存器的读取只能在本模块没有处于运算状态时（即 AES_CR.Start = 0 时）才能进行，否则对本寄存器的读取将得到全 0。

26.2.3 密钥寄存器 (AES_Key)

起始地址：0x40021400

地址偏移：0x20-0x3C

复位值：0x0000 0000

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

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

Bit	Field	Type	Reset	Description
31:0	Key	rw	0x00	存放 AES 算法的 128、192、256 比特密钥

1. 密钥寄存器由 8 个 32 位的寄存器组成，存放输入的初始密钥。写操作时需要分别对 8 个 32 位的寄存器进行操作。对应的操作顺序如下：

   128 位密钥举例：0x0F0E0D0C_0B0A0908_07060504_03020100

表 101

偏移地址	寄存器名称	填入数据
0x20	AES_Key0	0x03020100
0x24	AES_Key1	0x07060504
0x28	AES_Key2	0x0B0A0908
0x2C	AES_Key3	0x0F0E0D0C

192 位密钥举例：0x17161514_13121110_0F0E0D0C_0B0A0908_07060504_03020100

表 102

偏移地址	寄存器名称	填入数据
0x20	AES_Key0	0x03020100
0x24	AES_Key1	0x07060504
0x28	AES_Key2	0x0B0A0908
0x2C	AES_Key3	0x0F0E0D0C
0x30	AES_Key4	0x13121110
0x34	AES_Key5	0x17161514

256 位密钥举例:0x1F1E1D1C_1B1A1918_17161514_13121110_0F0E0D0C_0B0A0908_07060504_03020100

表 103

偏移地址	寄存器名称	填入数据
0x20	AES_Key0	0x03020100
0x24	AES_Key1	0x07060504
0x28	AES_Key2	0x0B0A0908
0x2C	AES_Key3	0x0F0E0D0C
0x30	AES_Key4	0x13121110
0x34	AES_Key5	0x17161514
0x38	AES_Key6	0x1B1A1918
0x3C	AES_Key7	0x1F1E1D1C

2. 对于本寄存器的写入只能在本模块没有处于运算状态时（即 AES_CR.Start = 0 时）才能进行，否则硬件将自动忽略对本寄存器的写操作。
3. 对于本寄存器的读取只能在本模块没有处于运算状态时（即 AES_CR.Start = 0 时）才能进行，否则对本寄存器的读取将得到全 0。

26.3 异常机制

- 只支持 32 位访问，其它位宽的访问会导致系统异常，进入硬件异常中断。
- 访问 AES 模块的偏移地址大于等于 0x80 的地址，会导致系统异常，进入硬件异常中断。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

26.4 本模块操作说明

本模块共有两个功能：加密、解密。对两个功能的操作有一些共同的特点，下面先介绍其共同点，再分别介绍每个功能的标准操作流程。

26.4.1 IP 操作的共同点

1. 在 AES 加解密过程中，数据寄存器会改变，如果下次运算的被操作数据就是本次运算的结果，那么就无需重新写入数据了。
2. 支持 128、192 和 256 位密钥，128 位密钥写入偏移地址 0x20~0x2C，192 位密钥写入偏移地址 0x20~0x34，256 位密钥写入偏移地址 0x20~0x3C。
3. 判断模块运算结束的方法：不断读取 AES_CR.Start，如果其值变为 0，则表示运算结束。

26.4.2 加密操作流程

Step 1：将待加密的 128 位数据写入数据寄存器（AES_DATA）中。

Step 2：将加密密钥写入密钥寄存器（AES_KEY）中。

Step 3：根据密钥长度设置 AES_CR.Keysize。

Step 4：将 AES_CR.Mode 设置为 0，启动加密模式。

Step 5：向控制寄存器中的 AES_CR.Start 写入 1，启动模块进行运算。Step 3、Step 4 和 Step 5 可同时进行。

Step 6：等待 AES_CR.Start 的值恢复为 0，模块运算结束。

Step 7：读取数据寄存器（AES_DATA），获得 128 位密文。

26.4.3 解密操作流程

Step 1：将待解密的 128 位数据写入数据寄存器（AES_DATA）中。

Step 2：将解密密钥写入密钥寄存器（AES_KEY）中。

Step 3：根据密钥长度设置 AES_CR.Keysize。

Step 4：将 AES_CR.Mode 设置为 1，启动解密模式。

Step 5：向控制寄存器中的 AES_CR.Start 写入 1，启动模块进行运算。Step 3、Step 4 和 Step 5 可同时进行。

Step 6：等待 AES_CR.Start 的值恢复为 0，模块运算结束。

Step 7：读取数据寄存器（AES_DATA），获得 128 位明文。

26.4.4 数据示例

128 位明文：0xFFEEDDCCBBAA99887766554433221100

128 位密钥：0x0F0E0D0C0B0A09080706050403020100

128 位密文：0x5AC5B47080B7CDD830047B6AD8E0C469

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

表 104: 128 位操作寄存器示例-加密前

寄存器	值（密钥）	寄存器	值（明文）
Key0	0x03020100	Data0	0x33221100
Key1	0x07060504	Data1	0x77665544
Key2	0x0B0A0908	Data2	0xBBAA9988
Key3	0x0F0E0D0C	Data3	0xFFEEDDCC

表 105: 128 位操作寄存器示例-加密后

寄存器	值（密钥）	寄存器	值（密文）
Key0	0x03020100	Data0	0xD8E0C469
Key1	0x07060504	Data1	0x30047B6A
Key2	0x0B0A0908	Data2	0x80B7CDD8
Key3	0x0F0E0D0C	Data3	0x5AC5B470

128 位明文：0xFFEEDDCCBBAA99887766554433221100

192 位密钥：0x1716151413121110F0E0D0C0B0A0908070605040302010 0

128 位密文：0x5AC5B47080B7CDD830047B6AD8E0C469

表 106: 192 位操作寄存器示例-加密前

寄存器	值（密钥）	寄存器	值（明文）
Key0	0x03020100	Data0	0x33221100
Key1	0x07060504	Data1	0x77665544
Key2	0x0B0A0908	Data2	0xBBAA9988
Key3	0x0F0E0D0C	Data3	0xFFEEDDCC
Key4	0x13121110		
Key5	0x17161514		

表 107: 192 位操作寄存器示例-加密后

寄存器	值（密钥）	寄存器	值（密文）
Key0	0x03020100	Data0	0xA47CA9DD
Key1	0x07060504	Data1	0xE0DF4C86
Key2	0x0B0A0908	Data2	0xA070AF6E
Key3	0x0F0E0D0C	Data3	0x91710DEC
Key4	0x13121110		
Key5	0x17161514		

128 位明文：0xFFEEDDCCBBAA99887766554433221100

256 位密钥：0x1F1E1D1C1B1A191817161514131211100F0E0D0C0B0A09080706050403020100

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

128 位密文：0x5AC5B47080B7CDD830047B6AD8E0C469

表 108: 256 位操作寄存器示例-加密前

寄存器	值（密钥）	寄存器	值（明文）
Key0	0x03020100	Data0	0x33221100
Key1	0x07060504	Data1	0x77665544
Key2	0x0B0A0908	Data2	0xBBAA9988
Key3	0x0F0E0D0C	Data3	0xFFEEDDCC
Key4	0x13121110		
Key5	0x17161514		
Key6	0x1B1A1918		
Key7	0x1F1E1D1C		

表 109: 256 位操作寄存器示例-加密后

寄存器	值（密钥）	寄存器	值（密文）
Key0	0x03020100	Data0	0xCAB7A28E
Key1	0x07060504	Data1	0xBF456751
Key2	0x0B0A0908	Data2	0x9049FCAE
Key3	0x0F0E0D0C	Data3	0x8960494B
Key4	0x13121110		
Key5	0x17161514		
Key6	0x1B1A1918		
Key7	0x1F1E1D1C		

26.5 运行时间说明

本模块从启动一次运算（AES_CR.Start 写入 1）到该次运算结束（AES_CR.Start 恢复到 0）所需时间如表 110所示：

表 110: AES 加解密运行时间

	128 位密钥	192 位密钥	256 位密钥
加密	220 cycles	260 cycles	300 cycles
解密	290 cycles	332 cycles	398 cycles

27 液晶控制器 (LCD)

27.1 LCD 简介

LCD 控制器是一款适用于单色无源液晶显示器 (LCD) 的数字控制器/驱动器，最多具有 8 个公用端子（COM）和 48 个区段端子（SEG），用以驱动 208 (4x52) 或 384 (8x48) 个 LCD 图像元素。端子的确切数量取决于数据手册中所述的器件引脚。

LCD 由若干区段（像素或完整符号）组成，这些区段均可点亮或熄灭。每个区段都包含一层在两根电极之间对齐的液晶分子。当向液晶施加高于阈值电压的电压时，相应的区段可见。区段电压必须为交流，以避免液晶中出现电泳效应（这将影响显示效果）。之后，必须在区段两端生成波形以避免出现直流。

词汇表

- 液晶（LCD）：无源显示面板，带有直接引向区段的端子。
- 公用（COM）：连接到多个区段的电气连接端子。
- 偏置（BIAS）：驱动 LCD 时使用的电压等级，定义为 1/(驱动 LCD 显示的电压等级数-1)。
- 区段（SEG）：最小可视单元（LCD 显示器上的最小组成元素，线条或点）。
- 占空比（DUTY）：定义为 1/(LCD 显示器上的公用端子数) 的数字。
- 帧：写入区段的波形的一个周期。
- 帧速率：每秒帧数，即每秒激励 LCD 区段的次数。

27.2 LCD 主要特性

- 高度灵活的帧速率控制。
- 支持静态、1/2、1/3、1/4、1/6 和 1/8 占空比。
- 支持 1/2、1/3 偏置。
- 多达 16 个寄存器的 LCD 数据 RAM。
- 可通过软件配置 LCD 的对比度。
- 3 种驱动波形生成方式。

  -内部电阻分压、外部电阻分压，外部电容分压方式

  -可通过软件配置内部电阻分压方式的功耗，从而匹配 LCD 面板所需的电容电荷
- 支持低功耗模式：LCD 控制器可在 Active、Sleep、DeepSleep 模式下进行显示。
- 可配置帧中断。
- 支持 LCD 闪烁功能且可配置多种闪烁频率。
- 未使用的 LCD 区段和公共引脚可配置为数字或模拟功能。

27.3 LCD 框图

LCD 框图如图 186所示：

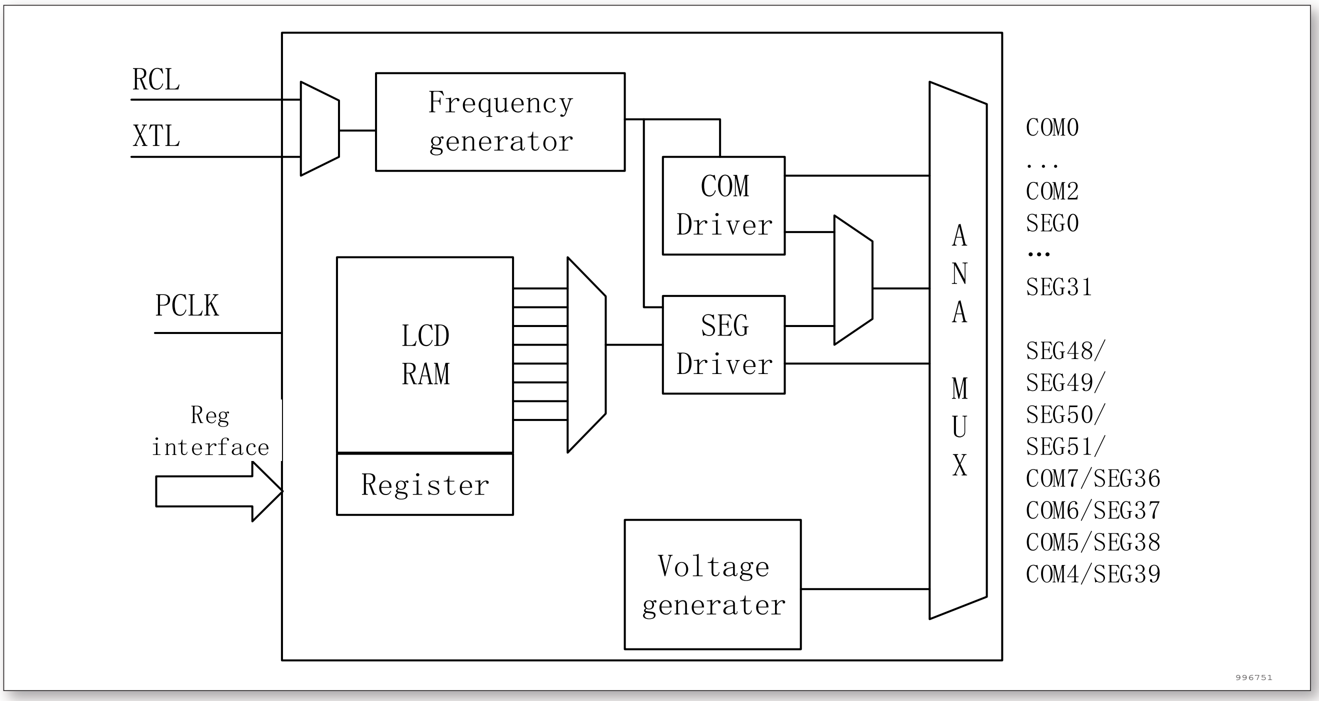

图 186. LCD 框图

27.4 LCD 驱动波形

LCD 支持 5 种占空比（Duty）的驱动波形：静态、1/2、1/3、1/4、1/6 和 1/8，由 LCD_CR0.Duty 进行设置。LCD 支持 2 种偏置（Bias）的驱动波形：1/2、1/3，由 LCD_CR0.Bias 进行设置。建议的组合方式如表 111所示：

表 111: LCD 驱动模式

	1/2 Duty	1/3 Duty	1/4 Duty	1/6 Duty	1/8 Duty
1/2 Bias	√	√	不推荐	不推荐	不推荐
1/3 Bias	不推荐	不推荐	√	√	√

各模式下的驱动波形如下方所示：

27.4.1 静态驱动波形

静态驱动波形如图 187所示：

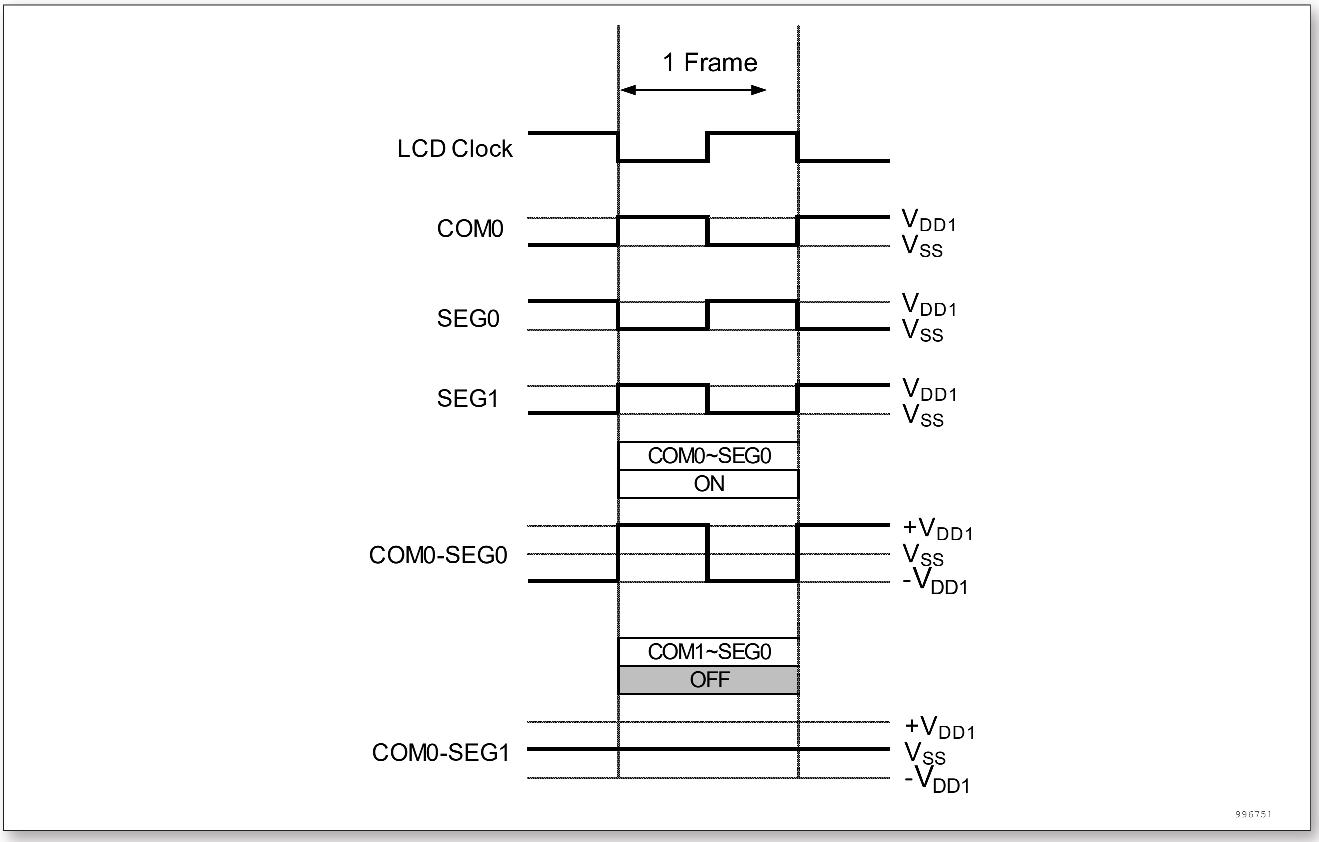

图 187. 静态驱动波形

27.4.2 1/2Duty 1/2Bias 驱动波形

1/2Duty 1/2Bias 驱动波形如图 188所示：

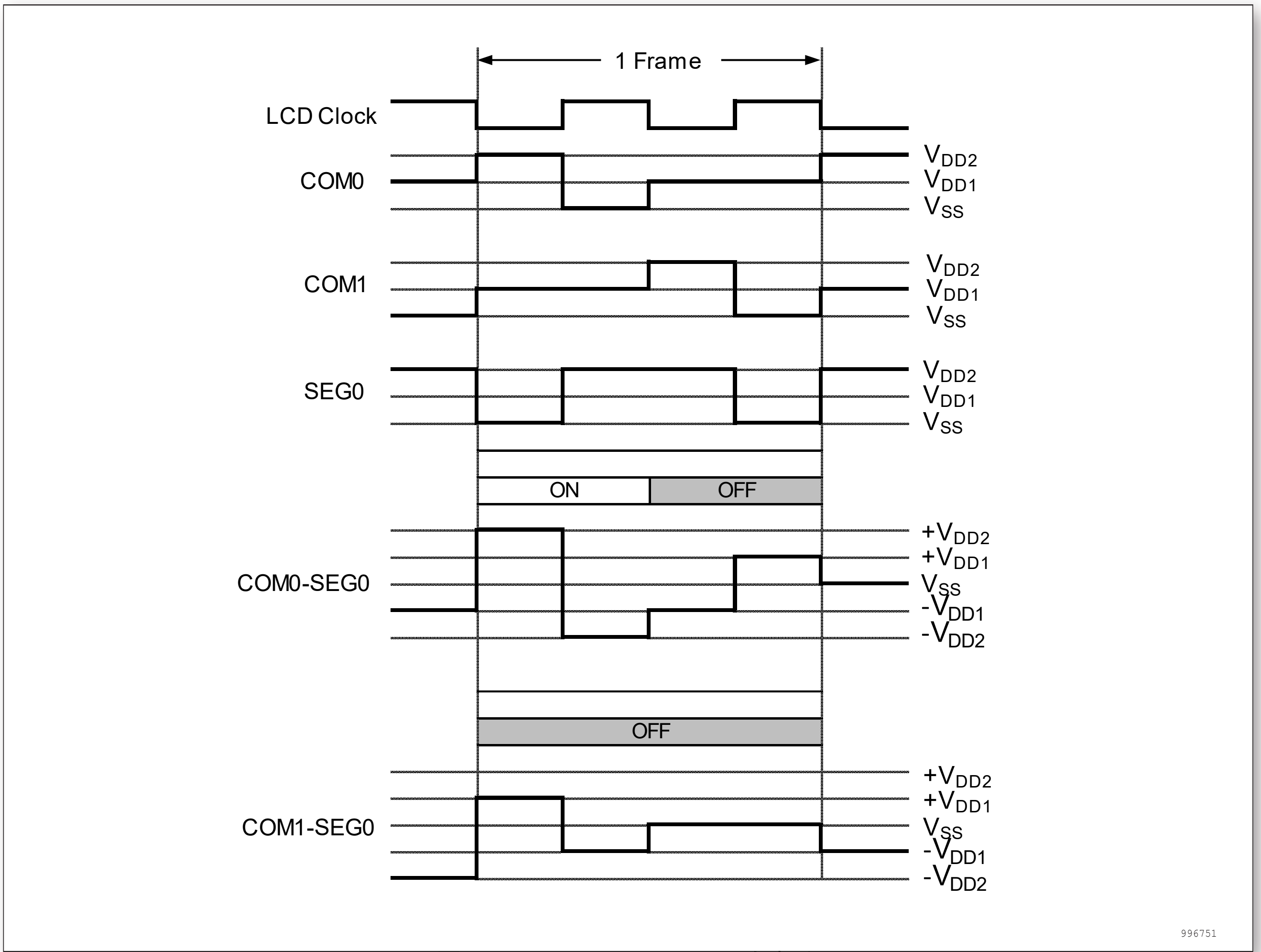

图 188. 1/2Duty 1/2Bias 驱动波形

27.4.3 1/2Duty 1/3Bias 驱动波形

1/2Duty 1/3Bias 驱动波形如图 189所示：

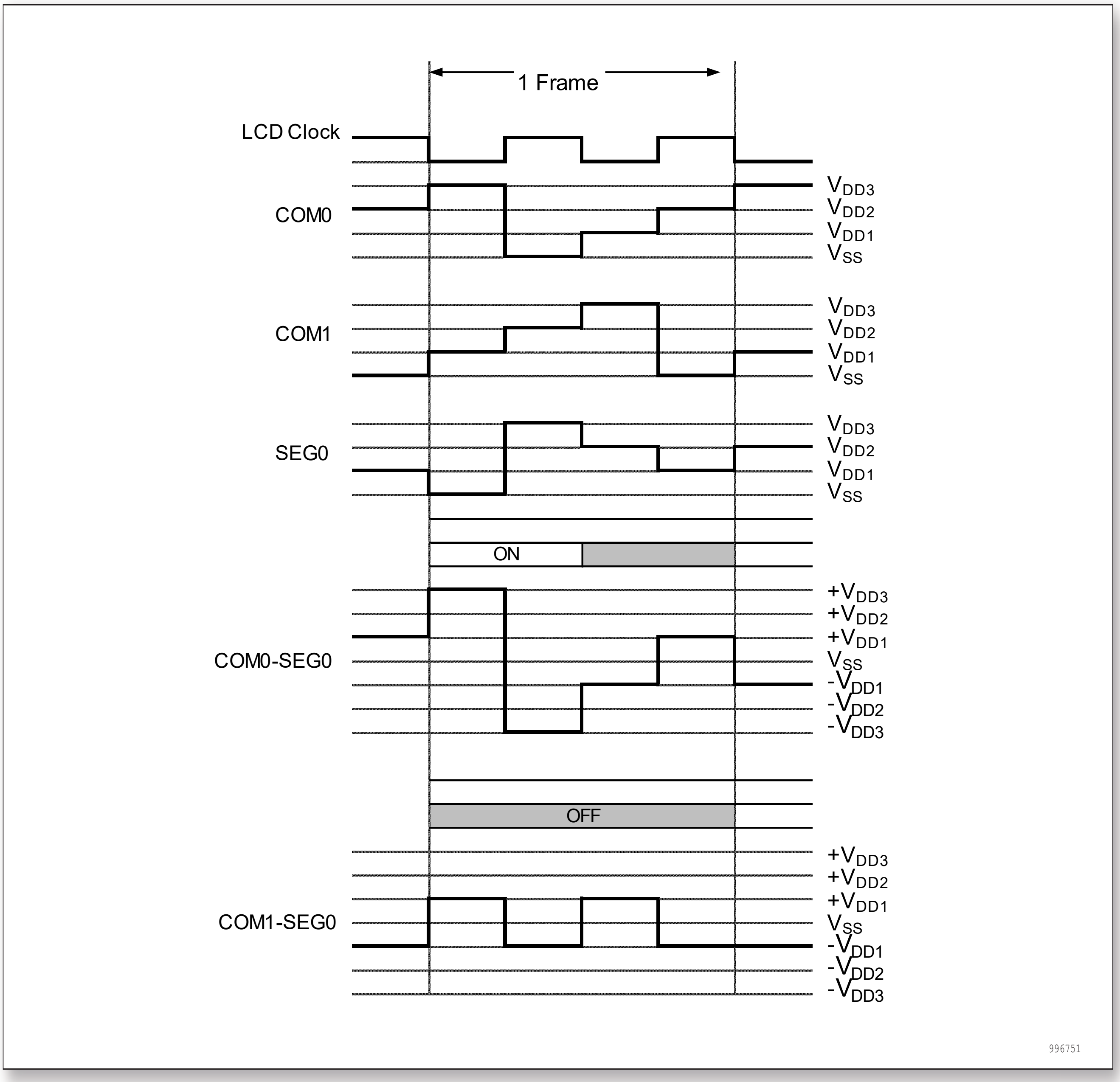

图 189. 1/2Duty 1/3Bias 驱动波形

27.4.4 1/3Duty 1/2Bias 驱动波形

1/3Duty 1/2Bias 驱动波形如图 190所示：

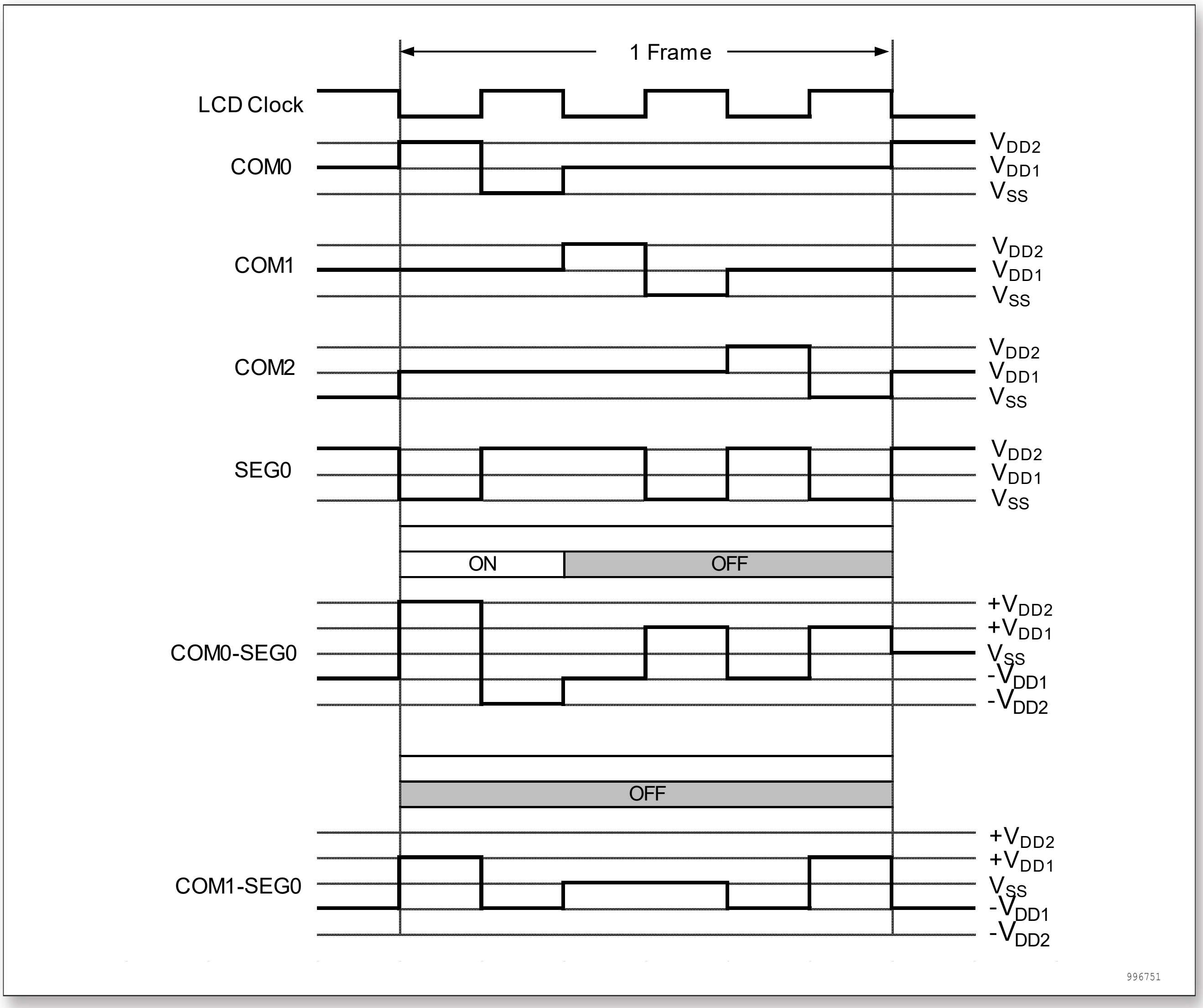

图 190. 1/3Duty 1/2Bias 驱动波形

27.4.5 1/3Buty 1/3Bias 驱动波形

1/3Buty 1/3Bias 驱动波形如图 191所示：

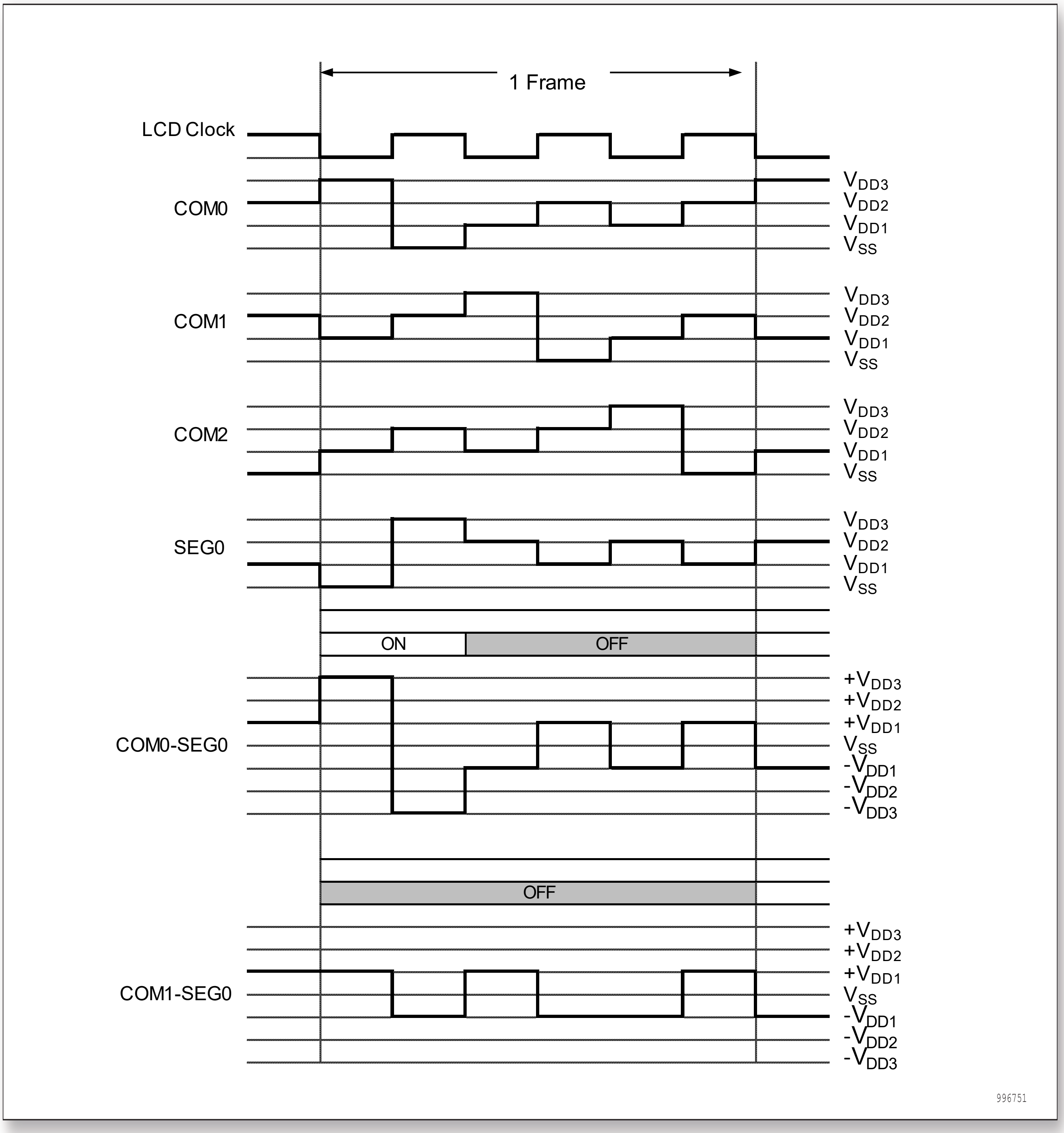

图 191. 1/3Buty 1/3Bias 驱动波形

27.4.6 1/4Duty 1/2Bias 驱动波形

1/4Duty 1/2Bias 驱动波形如图 192所示：

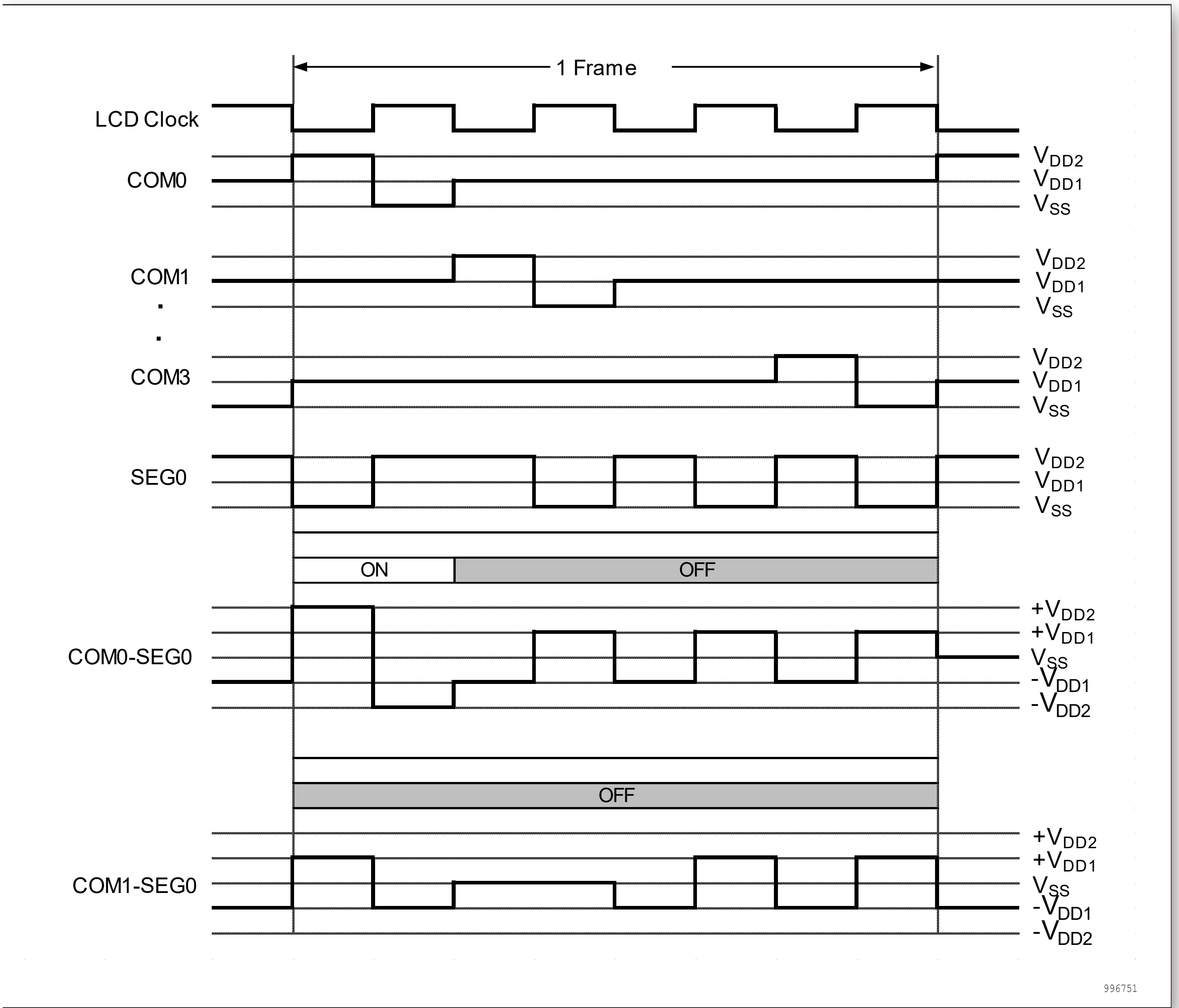

图 192. 1/4Duty 1/2Bias 驱动波形

27.4.7 1/4Duty 1/3Bias 驱动波形

1/4Duty 1/3Bias 驱动波形如图 193所示：

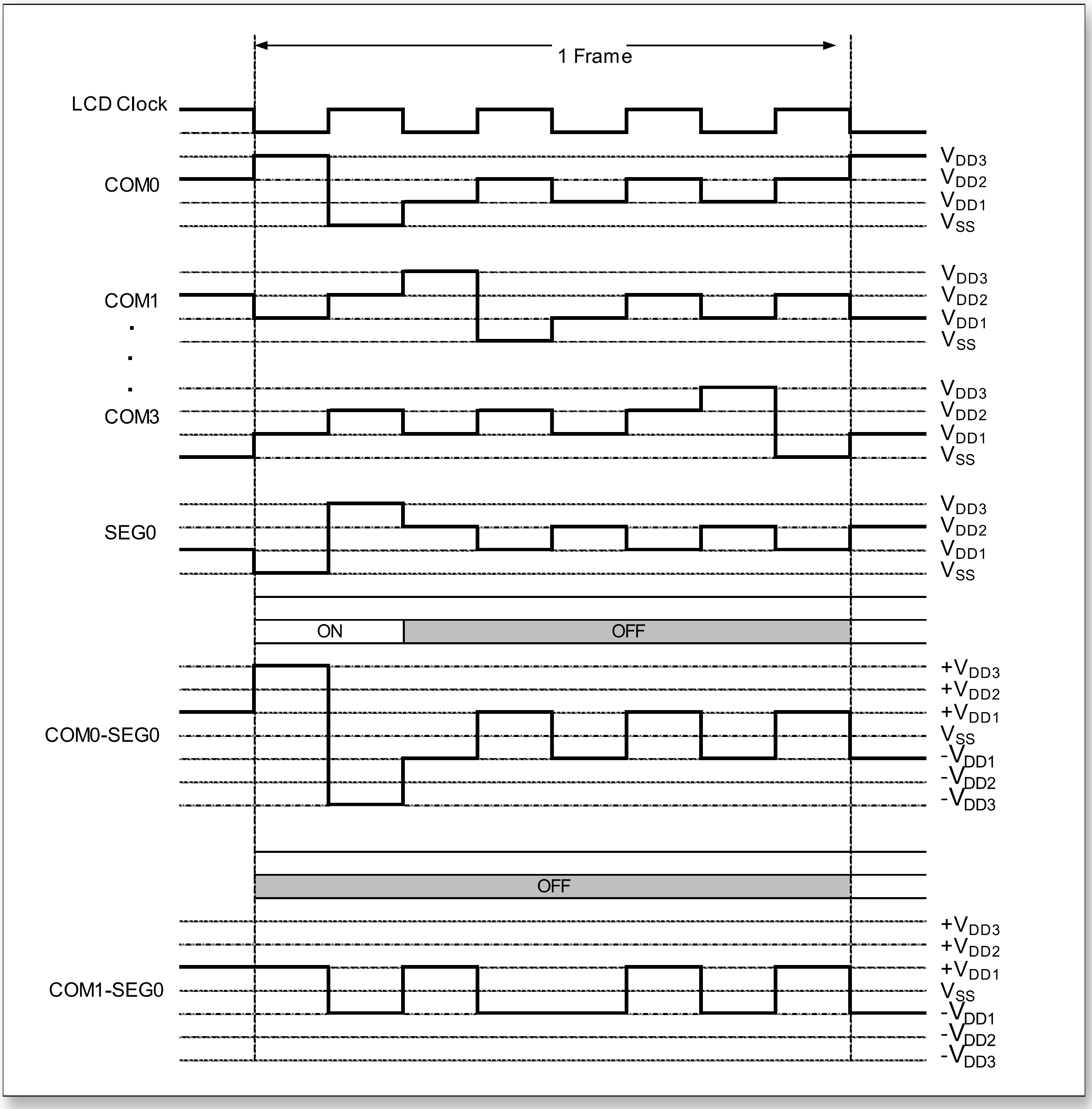

图 193. 1/4Duty 1/3Bias 驱动波形

27.4.8 1/6Duty 1/3Bias 驱动波形

1/6Duty 1/3Bias 驱动波形如图 194所示：

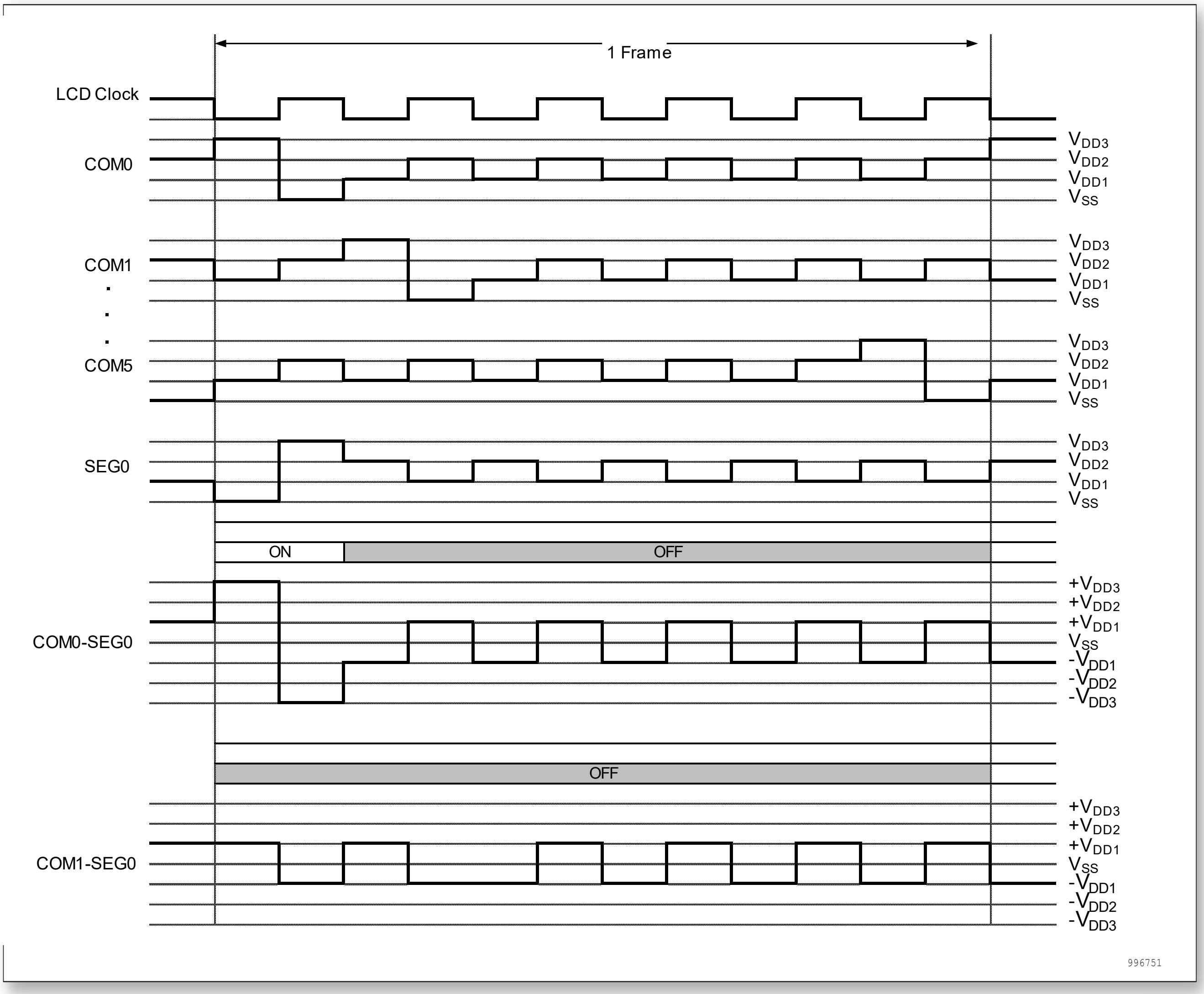

图 194. 1/6Duty 1/3Bias 驱动波形

27.4.9 1/8Duty 1/3Bias 驱动波形

1/8Duty 1/3Bias 驱动波形如图 195所示：

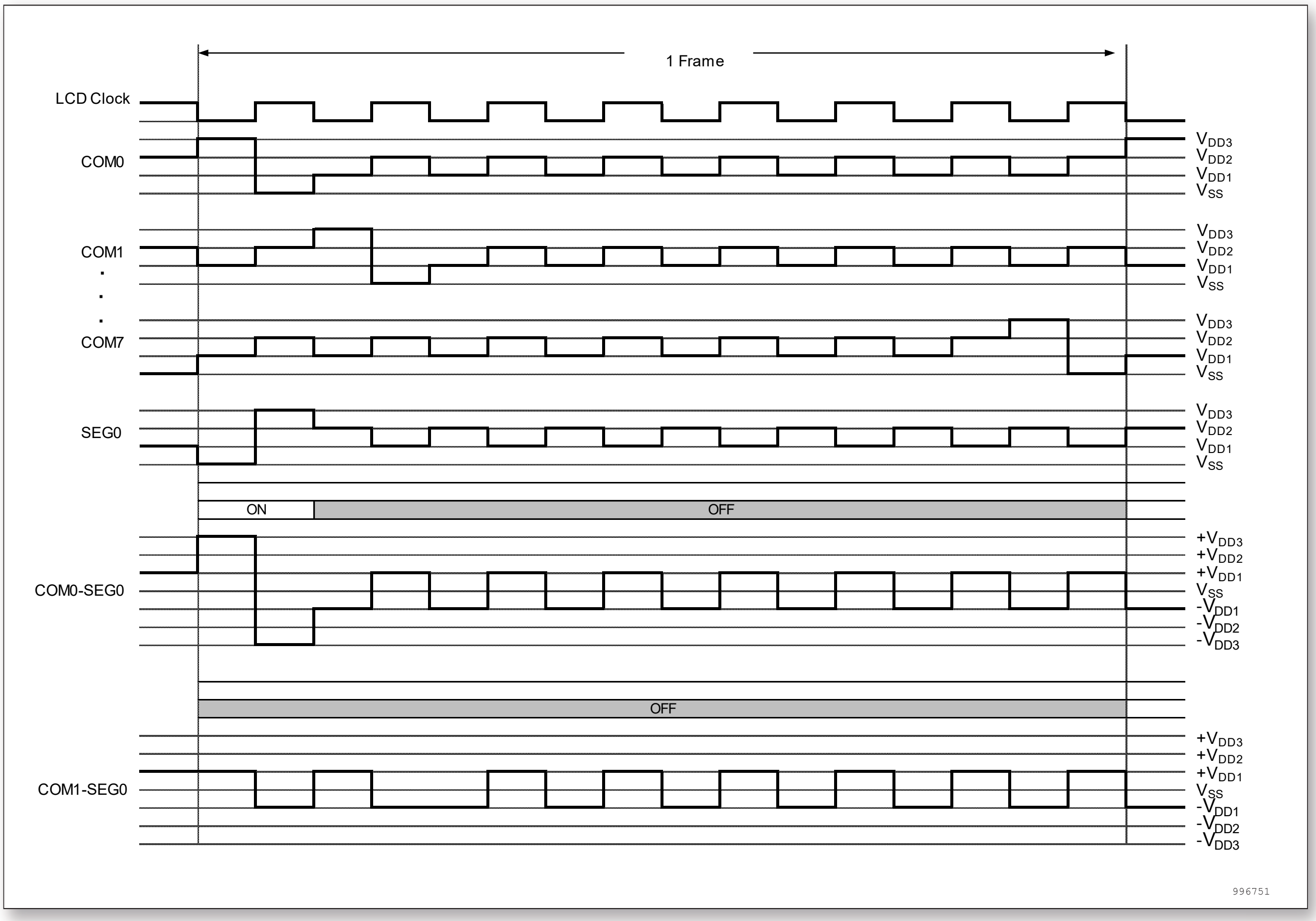

图 195. 1/8Duty 1/3Bias 驱动波形

27.5 LCD Bias 产生电路

LCD 的 Bias 电压具有 3 种来源：内部电阻分压、外部电阻分压、外部电容分压。当选择内部电阻分压时，芯片会自动切换内部的电路以产生符合 Bias 和 Duty 的电压。当选择外部电阻分压或外部电容分压时，需要用户在芯片的外围引脚搭建相关电路。

27.5.1 内部电阻模式

内部电阻模式 VLCDH,VLCD1~VLCD3 可以作为 LCD SEG 输出或者 IO 端口使用。内部电阻模式，根据 AVCC 的范围选择 CR0.Contrast 来调整 VLCD，计算公式如下：

1/3bias（op=CR0.Contrast）：

$$VLCD = \frac{AVCC * 18 * R}{(op<3>*7.5+op<2>*4+op<1>*2+op<0>*1+18)*R}$$

1/2bias（op=CR0.Contrast）：

$$VLCD = \frac{AVCC * 12 * R}{(op<3>*7.5+op<2>*4+op<1>*2+op<0>*1+12)*R}$$

表 112为 VCC 与 VLCD 的关系

表 112: VCC 与 VLCD 的关系

CR0.Contrast	VLCD(1/3bias)	VLCD(1/2bias)
0	1.00 * VCC	1.00 * VCC
1	0.94 * VCC	0.92 * VCC
2	0.9 * VCC	0.85 * VCC
3	0.85 * VCC	0.8 * VCC
4	0.81 * VCC	0.75 * VCC
5	0.78 * VCC	0.70 * VCC
6	0.75 * VCC	0.66 * VCC
7	0.72 * VCC	0.63 * VCC
8	0.70 * VCC	0.61 * VCC
9	0.67 * VCC	0.58 * VCC
10	0.65 * VCC	0.55 * VCC
11	0.63 * VCC	0.53 * VCC
12	0.61 * VCC	0.51 * VCC
13	0.59 * VCC	0.48 * VCC
14	0.57 * VCC	0.47 * VCC
15	0.55 * VCC	0.45 * VCC

27.5.2 外部电容模式

外部电容模式如图 196所示：

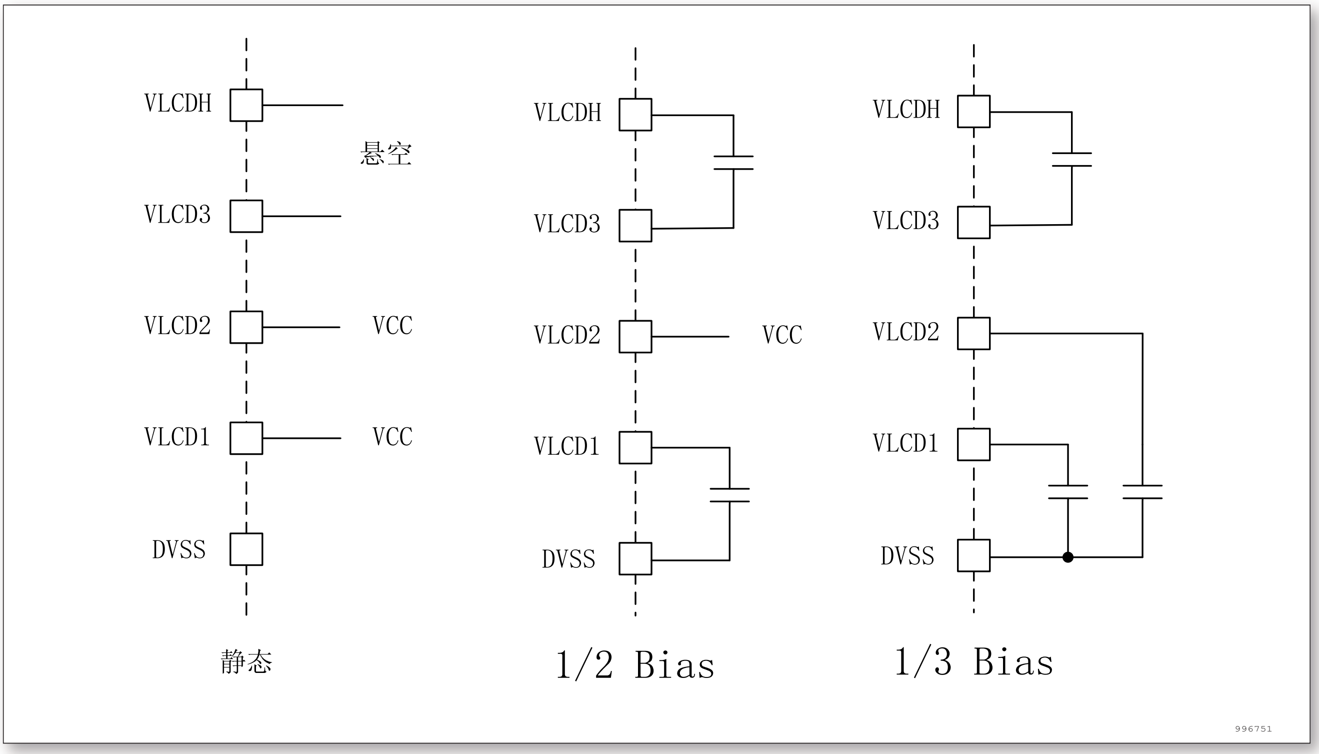

图 196. 外部电容模式

27.5.3 外部电阻模式

外部电阻模式图 197所示：

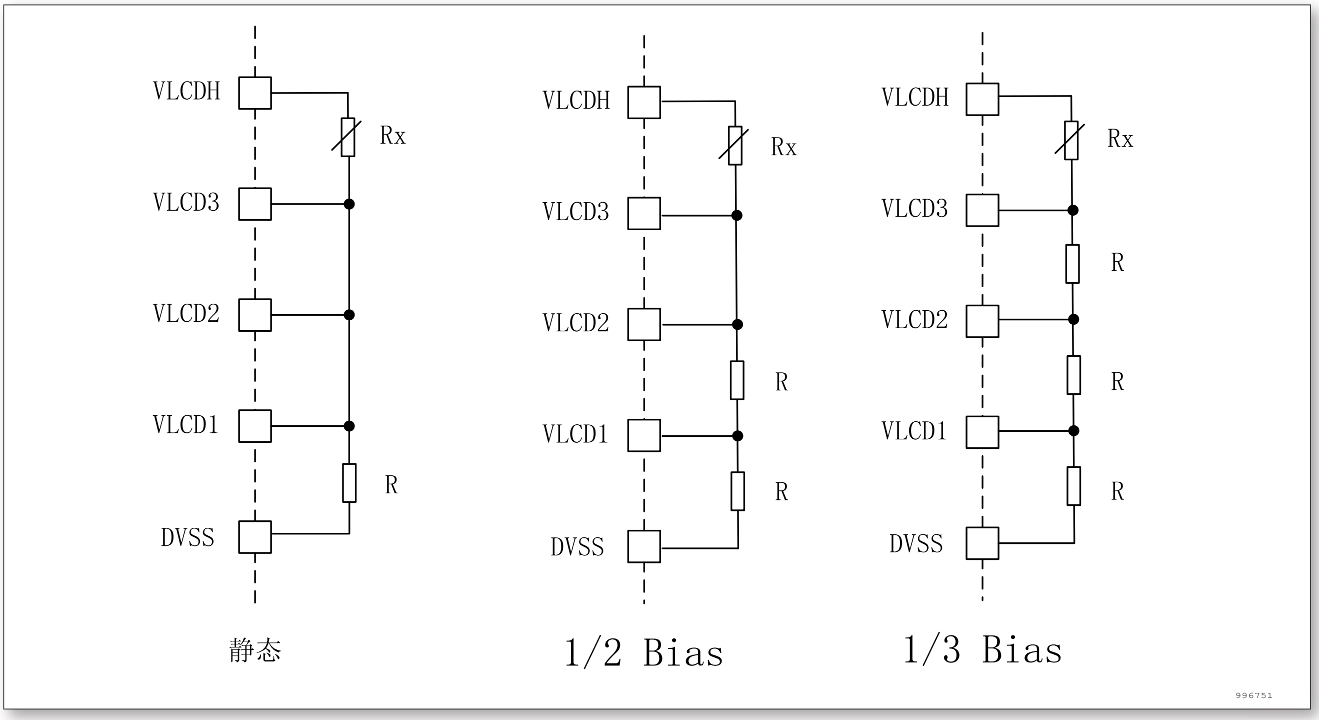

图 197. 外部电阻模式容模式

注意：

- Rx 为可调电阻，用于调节 LCD 显示对比度。
- 根据使用 LCD 屏幕选择合适的电阻 R。

27.6 DMA

LCD 支持软件和硬件触发 DMA 数据传输，可以将需要显示的内容从 RAM 或 ROM 中自动搬到 LCD 显示 RAM 中。硬件触发使用的是帧中断信号。LCD 使用的 DMA 通道号为 [6]。

DMA 数据传送配置流程：

1. 使能 DMA；
2. 使能 DMA 通道使能；
3. 选择 LCD DMA；
4. 设置传输类型、传输长度、传输方式；
5. 设置源起始地址，目标起始地址；
6. 设置源地址、目标地址的递增方式；
7. 根据需要使能 DMA 中断；
8. 使能 LCD DMA 触发。

27.7 中断

当 LCD 设置有效时，LCD 中断可以配置为帧数产生中断。

27.8 LCD 显示模式

LCD 支持两种显示模式。一种以 COM 为显示单元，同一个 SEG 的所有 COM 段在同一字节中（模式 0）。另外一种为同一个 COM 的不同 SEG 在同一个字节中（模式 1）。根据 LCD 面板选择合适的显示方式可以简化程序操作。

27.8.1 LCD 显示模式 1 (MODE = 1)

1/8 Duty图 198所示：

	Bit0	Bit1	Bit2	Bit3	Bit4	Bit5	Bit6	Bit7	Bit8	Bit9	Bit10	Bit11	Bit12	Bit13	Bit14	Bit15	Bit16	Bit17	Bit18	Bit19	Bit20	Bit21	Bit22	Bit23	Bit24	Bit25	Bit26	Bit27	Bit28	Bit29	Bit30	Bit31	
	SEG0	SEG1	SEG2	SEG3	SEG4	SEG5	SEG6	SEG7	SEG8	SEG9	SEG10	SEG11	SEG12	SEG13	SEG14	SEG15	SEG16	SEG17	SEG18	SEG19	SEG20	SEG21	SEG22	SEG23	SEG24	SEG25	SEG26	SEG27	SEG28	SEG29	SEG30	SEG31	
COM0																																	LCDRAM0
COM1																																	LCDRAM1
COM2																																	LCDRAM2
COM3																																	LCDRAM3
COM4																																	LCDRAM4
COM5																																	LCDRAM5
COM6																																	LCDRAM6
COM7																																	LCDRAM7
	SEG32	SEG33	SEG34	SEG35					SEG40	SEG41	SEG42	SEG43	SEG44	SEG45	SEG46	SEG47																	
COM0																																	LCDRAM8
COM1																																	LCDRAM9
COM2																																	LCDRAMA
COM3																																	LCDRAMB
COM4																																	LCDRAMC
COM5																																	LCDRAMD
COM6																																	LCDRAME
COM7																																	LCDRAMF

图 198. 1/8 Duty

1/6 Duty图 199所示：

	Bit0	Bit1	Bit2	Bit3	Bit4	Bit5	Bit6	Bit7	Bit8	Bit9	Bit10	Bit11	Bit12	Bit13	Bit14	Bit15	Bit16	Bit17	Bit18	Bit19	Bit20	Bit21	Bit22	Bit23	Bit24	Bit25	Bit26	Bit27	Bit28	Bit29	Bit30	Bit31	
	SEG0	SEG1	SEG2	SEG3	SEG4	SEG5	SEG6	SEG7	SEG8	SEG9	SEG10	SEG11	SEG12	SEG13	SEG14	SEG15	SEG16	SEG17	SEG18	SEG19	SEG20	SEG21	SEG22	SEG23	SEG24	SEG25	SEG26	SEG27	SEG28	SEG29	SEG30	SEG31	
COM0																																	LCDRAM0
COM1																																	LCDRAM1
COM2																																	LCDRAM2
COM3																																	LCDRAM3
COM4																																	LCDRAM4
COM5																																	LCDRAM5
																																	LCDRAM6
																																	LCDRAM7
	SEG32	SEG33	SEG34	SEG35	SEG36	SEG37			SEG40	SEG41	SEG42	SEG43	SEG44	SEG45	SEG46	SEG47	SEG48	SEG49	SEG50	SEG51													
COM0																																	LCDRAM8
COM1																																	LCDRAM9
COM2																																	LCDRAMA
COM3																																	LCDRAMB
COM4																																	LCDRAMC
COM5																																	LCDRAMD
																																	LCDRAME
																																	LCDRAMF

图 199. 1/6 Duty

1/4 Duty图 200所示：

	Bit0	Bit1	Bit2	Bit3	Bit4	Bit5	Bit6	Bit7	Bit8	Bit9	Bit10	Bit11	Bit12	Bit13	Bit14	Bit15	Bit16	Bit17	Bit18	Bit19	Bit20	Bit21	Bit22	Bit23	Bit24	Bit25	Bit26	Bit27	Bit28	Bit29	Bit30	Bit31	
	SEG0	SEG1	SEG2	SEG3	SEG4	SEG5	SEG6	SEG7	SEG8	SEG9	SEG10	SEG11	SEG12	SEG13	SEG14	SEG15	SEG16	SEG17	SEG18	SEG19	SEG20	SEG21	SEG22	SEG23	SEG24	SEG25	SEG26	SEG27	SEG28	SEG29	SEG30	SEG31	
COM0																																	LCDRAM0
COM1																																	LCDRAM1
COM2																																	LCDRAM2
COM3																																	LCDRAM3
																																	LCDRAM4
																																	LCDRAM5
																																	LCDRAM6
																																	LCDRAM7
	SEG32	SEG33	SEG34	SEG35	SEG36	SEG37	SEG38	SEG39	SEG40	SEG41	SEG42	SEG43	SEG44	SEG45	SEG46	SEG47	SEG48	SEG49	SEG50	SEG51													
COM0																																	LCDRAM8
COM1																																	LCDRAM9
COM2																																	LCDRAMA
COM3																																	LCDRAMB
																																	LCDRAMC
																																	LCDRAMD
																																	LCDRAME
																																	LCDRAMF

图 200. 1/4 Duty

1/3 Duty 1/2 Duty图 201所示：

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

	Bit0	Bit1	Bit2	Bit3	Bit4	Bit5	Bit6	Bit7	Bit8	Bit9	Bit10	Bit11	Bit12	Bit13	Bit14	Bit15	Bit16	Bit17	Bit18	Bit19	Bit20	Bit21	Bit22	Bit23	Bit24	Bit25	Bit26	Bit27	Bit28	Bit29	Bit30	Bit31	
	SEG0	SEG1	SEG2	SEG3	SEG4	SEG5	SEG6	SEG7	SEG8	SEG9	SEG10	SEG11	SEG12	SEG13	SEG14	SEG15	SEG16	SEG17	SEG18	SEG19	SEG20	SEG21	SEG22	SEG23	SEG24	SEG25	SEG26	SEG27	SEG28	SEG29	SEG30	SEG31	
COM0																																	LCDRAM0
COM1																																	LCDRAM1
COM2																																	LCDRAM2
																																	LCDRAM3
																																	LCDRAM4
																																	LCDRAM5
																																	LCDRAM6
																																	LCDRAM7
	SEG32	SEG33	SEG34	SEG35	SEG36	SEG37	SEG38	SEG39	SEG40	SEG41	SEG42	SEG43	SEG44	SEG45	SEG46	SEG47	SEG48	SEG49	SEG50	SEG51													
COM0																																	LCDRAM8
COM1																																	LCDRAM9
COM2																																	LCDRAMA
																																	LCDRAMB
																																	LCDRAMC
																																	LCDRAMD
																																	LCDRAME
																																	LCDRAMF

图 201. 1/3 Duty 1/2 Duty

27.8.2 LCD 显示模式 0 (MODE = 0)

1/8 Duty图 202所示：

Bit0	Bit1	Bit2	Bit3	Bit4	Bit5	Bit6	Bit7	Bit8	Bit9	Bit10	Bit11	Bit12	Bit13	Bit14	Bit15	Bit16	Bit17	Bit18	Bit19	Bit20	Bit21	Bit22	Bit23	Bit24	Bit25	Bit26	Bit27	Bit28	Bit29	Bit30	Bit31	
COM0	COM1	COM2	COM3	COM4	COM5	COM6	COM7	COM0	COM1	COM2	COM3	COM4	COM5	COM6	COM7	COM0	COM1	COM2	COM3	COM4	COM5	COM6	COM7	COM0	COM1	COM2	COM3	COM4	COM5	COM6	COM7	
SEG0								SEG1								SEG2								SEG3								LCDRAM0
SEG4								SEG5								SEG6								SEG7								LCDRAM1
SEG8								SEG9								SEG10								SEG11								LCDRAM2
SEG12								SEG13								SEG14								SEG15								LCDRAM3
SEG16								SEG17								SEG18								SEG19								LCDRAM4
SEG20								SEG21								SEG22								SEG23								LCDRAM5
SEG24								SEG25								SEG26								SEG29								LCDRAM6
SEG28								SEG29								SEG30								SEG31								LCDRAM7
SEG32								SEG40								SEG48																LCDRAM8
SEG33								SEG41								SEG49																LCDRAM9
SEG34								SEG42								SEG50																LCDRAMA
SEG35								SEG43								SEG51																LCDRAMB
								SEG44																								LCDRAMC
								SEG45																								LCDRAMD
								SEG46																								LCDRAME
								SEG47																								LCDRAMF

图 202. 1/8 Duty

1/6 Duty图 203所示：

Bit0	Bit1	Bit2	Bit3	Bit4	Bit5	Bit6	Bit7	Bit8	Bit9	Bit10	Bit11	Bit12	Bit13	Bit14	Bit15	Bit16	Bit17	Bit18	Bit19	Bit20	Bit21	Bit22	Bit23	Bit24	Bit25	Bit26	Bit27	Bit28	Bit29	Bit30	Bit31	
COM0	COM1	COM2	COM3	COM4	COM5			COM0	COM1	COM2	COM3	COM4	COM5			COM0	COM1	COM2	COM3	COM4	COM5			COM0	COM1	COM2	COM3	COM4	COM5	COM6	COM7	
SEG0								SEG1								SEG2								SEG3								LCDRAM0
SEG4								SEG5								SEG6								SEG7								LCDRAM1
SEG8								SEG9								SEG10								SEG11								LCDRAM2
SEG12								SEG13								SEG14								SEG15								LCDRAM3
SEG16								SEG17								SEG18								SEG19								LCDRAM4
SEG20								SEG21								SEG22								SEG23								LCDRAM5
SEG24								SEG25								SEG26								SEG29								LCDRAM6
SEG28								SEG29								SEG30								SEG31								LCDRAM7
SEG32								SEG40								SEG48																LCDRAM8
SEG33								SEG41								SEG49																LCDRAM9
SEG34								SEG42								SEG50																LCDRAMA
SEG35								SEG43								SEG51																LCDRAMB
SEG36								SEG44																								LCDRAMC
SEG37								SEG45																								LCDRAMD
								SEG46																								LCDRAME
								SEG47																								LCDRAMF

图 203. 1/6 Duty

1/4 Duty图 204所示：

Bit0	Bit1	Bit2	Bit3	Bit4	Bit5	Bit6	Bit7	Bit8	Bit9	Bit10	Bit11	Bit12	Bit13	Bit14	Bit15	Bit16	Bit17	Bit18	Bit19	Bit20	Bit21	Bit22	Bit23	Bit24	Bit25	Bit26	Bit27	Bit28	Bit29	Bit30	Bit31	
COM0	COM1	COM2	COM3					COM0	COM1	COM2	COM3					COM0	COM1	COM2	COM3					COM0	COM1	COM2	COM3					
SEG0								SEG1								SEG2								SEG3								LCDRAM0
SEG4								SEG5								SEG6								SEG7								LCDRAM1
SEG8								SEG9								SEG10								SEG11								LCDRAM2
SEG12								SEG13								SEG14								SEG15								LCDRAM3
SEG16								SEG17								SEG18								SEG19								LCDRAM4
SEG20								SEG21								SEG22								SEG23								LCDRAM5
SEG24								SEG25								SEG26								SEG29								LCDRAM6
SEG28								SEG29								SEG30								SEG31								LCDRAM7
SEG32								SEG40								SEG48																LCDRAM8
SEG33								SEG41								SEG49																LCDRAM9
SEG34								SEG42								SEG50																LCDRAMA
SEG35								SEG43								SEG51																LCDRAMB
SEG36								SEG44																								LCDRAMC
SEG37								SEG45																								LCDRAMD
SEG38								SEG46																								LCDRAME
SEG39								SEG47																								LCDRAMF

图 204. 1/4 Duty

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

1/3 Duty 1/2 Duty图 205所示：

Bit0	Bit1	Bit2	Bit3	Bit4	Bit5	Bit6	Bit7	Bit8	Bit9	Bit10	Bit11	Bit12	Bit13	Bit14	Bit15	Bit16	Bit17	Bit18	Bit19	Bit20	Bit21	Bit22	Bit23	Bit24	Bit25	Bit26	Bit27	Bit28	Bit29	Bit30	Bit31	
COM0	COM1	COM2						COM0	COM1	COM2						COM0	COM1	COM2						COM0	COM1	COM2						
SEG0								SEG1								SEG2								SEG3								LCDRAM0
SEG4								SEG5								SEG6								SEG7								LCDRAM1
SEG8								SEG9								SEG10								SEG11								LCDRAM2
SEG12								SEG13								SEG14								SEG15								LCDRAM3
SEG16								SEG17								SEG18								SEG19								LCDRAM4
SEG20								SEG21								SEG22								SEG23								LCDRAM5
SEG24								SEG25								SEG26								SEG29								LCDRAM6
SEG28								SEG29								SEG30								SEG31								LCDRAM7
SEG32								SEG40								SEG48																LCDRAM8
SEG33								SEG41								SEG49																LCDRAM9
SEG34								SEG42								SEG50																LCDRAMA
SEG35								SEG43								SEG51																LCDRAMB
SEG36								SEG44																								LCDRAMC
SEG37								SEG45																								LCDRAMD
SEG38								SEG46																								LCDRAME
SEG39								SEG47																								LCDRAMF

图 205. 1/3 Duty 1/2 Duty

27.9 LCD 寄存器

表 113. LCD 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x000	LCD_CR0	LCD 配置寄存器 0	0x0000 00DA	小节 27.9.1
0x004	LCD_CR1	LCD 配置寄存器 1	0x0000 0000	小节 27.9.2
0x008	LCD_INTCLR	LCD 中断清除寄存器	0x0000 0000	小节 27.9.3
0x00C	LCD_POEN0	LCD 输出配置寄存器	0xFFFF FFFF	小节 27.9.4
0x010	LCD_POEN1	LCD 输出配置寄存器	0x01FF FFFF	小节 27.9.5
0x040	LCD_RAM0	LCD RAM0	0xXXXX XXXX	小节 27.9.6
0x044	LCD_RAM1	LCD RAM1	0xXXXX XXXX	小节 27.9.6
0x048	LCD_RAM2	LCD RAM2	0xXXXX XXXX	小节 27.9.6
0x04C	LCD_RAM3	LCD RAM3	0xXXXX XXXX	小节 27.9.6
0x050	LCD_RAM4	LCD RAM4	0xXXXX XXXX	小节 27.9.6
0x054	LCD_RAM5	LCD RAM5	0xXXXX XXXX	小节 27.9.6
0x058	LCD_RAM6	LCD RAM6	0xXXXX XXXX	小节 27.9.6
0x05C	LCD_RAM7	LCD RAM7	0xXXXX XXXX	小节 27.9.6
0x060	LCD_RAM8	LCD RAM8	0xXXXX XXXX	小节 27.9.7
0x064	LCD_RAM9	LCD RAM9	0xXXXX XXXX	小节 27.9.7
0x068	LCD_RAMA	LCD RAM10	0xXXXX XXXX	小节 27.9.7
0x06C	LCD_RAMB	LCD RAM11	0xXXXX XXXX	小节 27.9.7
0x070	LCD_RAMC	LCD RAM12	0xXXXX XXXX	小节 27.9.7
0x074	LCD_RAMD	LCD RAM13	0xXXXX XXXX	小节 27.9.7
0x078	LCD_RAME	LCD RAM14	0xXXXX XXXX	小节 27.9.7
0x07C	LCD_RAMF	LCD RAM15	0xXXXX XXXX	小节 27.9.7

27.9.1 LCD 配置寄存器 0 (LCD_CR0)

起始地址：0x40005C00

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

地址偏移：0x000

复位值：0x0000 00DA

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Reserved															

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Contrast				BSEL			Duty			Bias	CpClk		LcdClk		EN
RW				RW			RW			RW	RW		RW		RW

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:12	Contrast	rw	0x00	LCD 对比度调整 注：仅当 Bias 电压来源选择内部电阻分压时有效。 Constrast 值越大，LCD 波形的幅度越小。 0X0 时，LCD 波形幅度最大，对比度最大； …… 0XF 时，LCD 波形幅度最小，对比度最小；
11:9	BSEL	rw	0x00	Bias 电压来源选择 111：Reserved 110：内部电阻分压，大功耗模式 101：Reserved 100：内部电阻分压，小功耗模式 011：Reserved 010：内部电阻分压，中功耗模式 001：电容分压模式，需要外部电路配合 000：外部电阻模式，需要外部电路配合

<table>
<tr><th>Bit</th><th>Field</th><th>Type</th><th>Reset</th><th>Description</th></tr>
<tr><td>8:6</td><td>Duty</td><td>rw</td><td>0x03</td><td>LCD duty 配置
000: 静态
001: 1/2 duty
010: 1/3 duty
011: 1/4 duty
100: Reserved
101: 1/6 duty
110: Reserved
111: 1/8 duty
<table>
<tr><th>duty</th><th>PAD</th><th>FUNCTION</th></tr>
<tr><td rowspan="4">1/4 duty</td><td>COM4SEG39PAD</td><td>SEG39</td></tr>
<tr><td>COM5SEG38PAD</td><td>SEG38</td></tr>
<tr><td>COM6SEG37PAD</td><td>SEG37</td></tr>
<tr><td>COM7SEG36PAD</td><td>SEG36</td></tr>
<tr><td rowspan="4">1/6 duty</td><td>COM4SEG39PAD</td><td>COM4</td></tr>
<tr><td>COM5SEG38PAD</td><td>COM5</td></tr>
<tr><td>COM6SEG37PAD</td><td>SEG37</td></tr>
<tr><td>COM7SEG36PAD</td><td>SEG36</td></tr>
<tr><td rowspan="4">1/8 duty</td><td>COM4SEG39PAD</td><td>COM4</td></tr>
<tr><td>COM5SEG38PAD</td><td>COM5</td></tr>
<tr><td>COM6SEG37PAD</td><td>COM6</td></tr>
<tr><td>COM7SEG36PAD</td><td>COM7</td></tr>
</table></td></tr>
<tr><td>5</td><td>Bias</td><td>rw</td><td>0x00</td><td>LCD Bias 配置
0: 1/3 bias
1: 1/2 bias</td></tr>
<tr><td>4:3</td><td>CpClk</td><td>rw</td><td>0x03</td><td>电压泵时钟频率选择
00: 2k Hz
01: 4k Hz
10: 8k Hz
11: 16k Hz</td></tr>
<tr><td>2:1</td><td>LcdClk</td><td>rw</td><td>0x01</td><td>LCD 扫描频率选择
00: 64Hz
01: 128Hz
10: 256Hz
11: 512Hz
注：LCD 帧频率 = LCD 扫描频率 ×Duty</td></tr>
<tr><td>0</td><td>EN</td><td>rw</td><td>0x00</td><td>LCD 使能控制
1：使能
0：禁止</td></tr>
</table>

27.9.2 LCD 配置寄存器 1 (LCD_CR1)

起始地址：0x40005C00

地址偏移：0x004

复位值：0x0000 0000

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Reserved															

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Reserved				INTF	DmaEn	IE	Mode	ClkSrc	BlinkE	BlinkCnt					
				RO	RW	RW	RW	RW	RW	RW					

Bit	Field	Type	Reset	Description
31:11	Reserved			保留
11	INTF	ro	0x00	中断标志 1：中断 0：无中断
10	DmaEn	rw	0x00	DMA 硬件触发使能 1：使能 LCD 中断触发 DMA，0：禁止 LCD 中断触发 DMA
9	IE	rw	0x00	中断使能 1：使能 0：禁止
8	Mode	rw	0x00	LCD RAM 显示模式选择 0 模式 0 1 模式 1
7	ClkSrc	rw	0x00	LCD 时钟源选择 1：XTL 0：RCL
8	BlinkEn	rw	0x00	LCD 闪屏配置 1：使能 0：禁止
5:0	BlinkCnt	rw	0x00	闪屏频率与 LCD 中断间隔设置 注：LCD 闪烁频率 = LCD 帧频率 / (BlinkCnt+1) LCD 中断间隔 = (BlinkCnt+1)*(1/LCD 帧频率)

27.9.3 LCD 中断清除寄存器 (LCD_INTCLR)

起始地址：0x40005C00

地址偏移：0x008

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Reserved															

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Reserved					INTF	Reserved									
					R0W0										

Bit	Field	Type	Reset	Description
31:11	Reserved		0x00	保留
10	INTF	R0W0	0x00	中断标志清除，写 0 清除，写 1 无效
9:0	Reserved		0x00	保留

27.9.4 LCD 输出配置寄存器 (LCD_POEN0)

起始地址：0x40005C00

地址偏移：0x00C

复位值：0xFFFF FFFF

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
S31	S30	S29	S28	S27	S26	S25	S24	S23	S22	S21	S20	S19	S18	S17	S16
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
S15	S14	S13	S12	S11	S10	S9	S8	S7	S6	S5	S4	S3	S2	S1	S0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31:0	Sx	rw	0xFF	Segx 输出控制位 0: SEG 输出使能 1: SEG 输出关闭，可以使用其他功能，如 IO, 模拟输入输出

27.9.5 LCD 输出配置寄存器 (LCD_POEN1)

起始地址：0x40005C00

地址偏移：0x010

复位值：0x01FF FFFF

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Reserved							C3	C2	C1	C0	S51	S50	S49	S48	S47
							RW	RW	RW	RW	RW	RW	RW	RW	RW

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
S46	S45	S44	MUX	S43	S42	S41	S40	S39 C4	S38 C5	S37 C6	S36 C7	S35	S34	S33	S32
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Field	Type	Reset	Description
3:0	Sx	rw	0x0F	Segx 输出控制位 0: SEG 输出使能 1: SEG 输出关闭，可以使用其他功能，如 IO, 模拟输入输出
7:4	SxCy	rw	0x0F	Segx/COMy 输出控制位 0: SEG/COM 输出使能 1: SEG/COM 输出关闭，可以使用其他功能，如 IO, 模拟输入输出 SEG COM 管脚功能选择由 CR0.DUTY 决定
11:8	Sx	rw	0x0F	Segx 输出控制位 0: SEG 输出使能 1: SEG 输出关闭，可以使用其他功能，如 IO, 模拟输入输出
12	MUX	rw	0x01	SEG32-SEG35 端口功能选择，详细参考表 114
20:13	Sx	rw	0xFF	SEG40-SEG51 端口功能选择
24:21	Cx	rw	0x0F	COMx 输出控制位 0: COM 输出使能 1: COM 输出关闭，可以使用其他功能，如 IO, 模拟输入输出

表 114: SEG32-SEG35 端口功能选择

VLCDXSEGXPAD 状态描述		**寄存器如何配置**				
		MUX	**S<35：32>**	**BSEL**		
VLCDXSEGXPAD 选择 GPIO，LCD disable(LCD_ON=0) 时	VLCDHSEG35=IO	1	1111	X	X	X
	VLCD3SEG34=IO					
	VLCD2SEG33=IO					
	VLCD1SEG32=IO					

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

VLCDXSEGXPAD 状态描述			寄存器如何配置 MUX	S<35：32>	BSEL		
VLCDXSEGXPAD 选择 IO，LCD enable(LCD_ON=1)时，这时只能选择内部电阻工作模式	小电阻（大电流）模式	VLCDHSEG35=IO	1	1111	1	1	0
		VLCD3SEG34=IO					
		VLCD2SEG33=IO					
		VLCD1SEG32=IO					
	中电阻（中电流）模式	VLCDHSEG35=IO	1	1111	0	1	0
		VLCD3SEG34=IO					
		VLCD2SEG33=IO					
		VLCD1SEG32=IO					
	大电阻（小电流）模式	VLCDHSEG35=IO	1	1111	1	0	0
		VLCD3SEG34=IO					
		VLCD2SEG33=IO					
		VLCD1SEG32=IO					
选择外部电阻工作模式，内部电阻断路		VLCDHSEG35=VOUT	0	0000	0	0	0
		VLCD3SEG34=VLCD3					
		VLCD2SEG33=VLCD2					
		VLCD1SEG32=VLCD1					
选择内部电阻工作模式	小电阻（大电流）模式	VLCDHSEG35=SEG35/IO	0	XXXX	1	1	0
		VLCD3SEG34=SEG34/IO					
		VLCD2SEG33=SEG33/IO					
		VLCD1SEG32=SEG32/IO					
	中电阻（中电流）模式	VLCDHSEG35=SEG35/IO	0	XXXX	0	1	0
		VLCD3SEG34=SEG34/IO					
		VLCD2SEG33=SEG33/IO					
		VLCD1SEG32=SEG32/IO					
	大电阻（小电流）模式	VLCDHSEG35=SEG35/IO	0	XXXX	1	0	0
		VLCD3SEG34=SEG34/IO					
		VLCD2SEG33=SEG33/IO					
		VLCD1SEG32=SEG32/IO					
选择外部电阻工作模式，内部电阻断路		VLCDHSEG35=DH1	0	0000	0	0	1
		VLCD3SEG34=DH2					
		VLCD2SEG33=VLCD2					
		VLCD1SEG32=VLCD1					

27.9.6 LCD RAM0-7

起始地址：0x40005C00

地址偏移：0x040 - 0x5c

复位值：0xXXXX XXXX

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
D31	D30	D29	D28	D27	D26	D25	D24	D23	D22	D21	D20	D19	D18	D17	D16
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31:0	Dx	rw	0xXX	LCD 点输出，显示参考 LCD 显示模式 0 对应的 SEG COM 交叉点不亮；1 对应的 SEG COM 交叉点亮；

27.9.7 LCD RAM8-F

起始地址：0x40005C00

地址偏移：0x060 - 0x7C

复位值：0xXXXX XXXX

b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Reserved								D23	D22	D21	D20	D19	D18	D17	D16
								RW	RW	RW	RW	RW	RW	RW	RW

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31:24	Reserved			保留
23:0	Dx	rw	0xXX	LCD 点输出，显示参考 LCD 显示模式 0 对应的 SEG COM 交叉点不亮；1 对应的 SEG COM 交叉点亮；

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

28 模数转换器（ADC）

28.1 模块简介

外部的模拟信号需要转变成数字信号才能由 MCU 进一步处理。内部集成了一个 12 位高精度、高转换速率的逐次逼近型模数转换器 (SAR ADC) 模块。具有以下特性：

- 12 位转换精度；
- 1M SPS 转换速度；
- 24 路输入通道，包括 20 路外部管脚输入、1 路内部温度传感器电压、1 路 1/3 AVCC 电压、1 路内建 BGR 1.2V 电压、1 路 DAC 内部输出；
- 4 种参考源：AVCC 电压、ExRef 引脚、内置 1.5v 参考电压、内置 2.5v 参考电压；
- ADC 的电压输入范围：0~Vref；
- 4 种转换模式：单次转换、顺序扫描连续转换、插队扫描连续转换、连续转换累加；
- 输入通道电压阈值监测；
- 软件可配置 ADC 的转换速率；
- 内置信号跟随器，可转换高阻信号；
- 支持片内外设自动触发 ADC 转换，有效降低芯片功耗并提高转换的实时性。

28.2 ADC 框图

图 206为 ADC 示意框图。

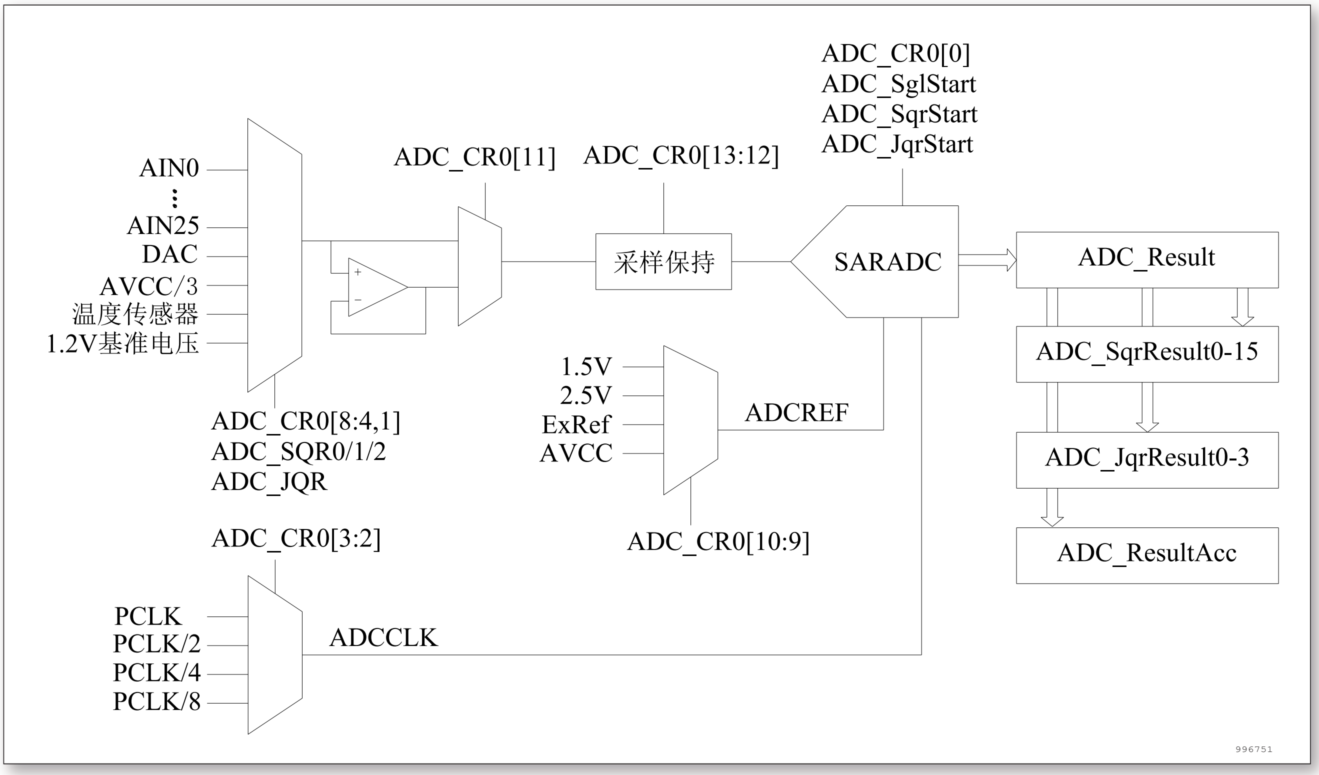

图 206. ADC 示意框图

28.3 转换时序及转换速度

ADC 的转换时序如图 207所示：一次完整的 ADC 转换由转换过程及逐次比较过程组成。其中转换过程需要 4~12 个 ADCCLK，由 ADC_CR0.SAM 配置；逐次比较过程需要 16 个 ADCCLK。所以，一次 ADC 转换共需要 20~28 个 ADCCLK。

ADC 转换速度的单位为 SPS，即每秒进行多少次 ADC 转换。ADC 转换速度的计算方法为：ADCCLK 的频率 / 一次 ADC 转换所需要的 ADCCLK 的个数。

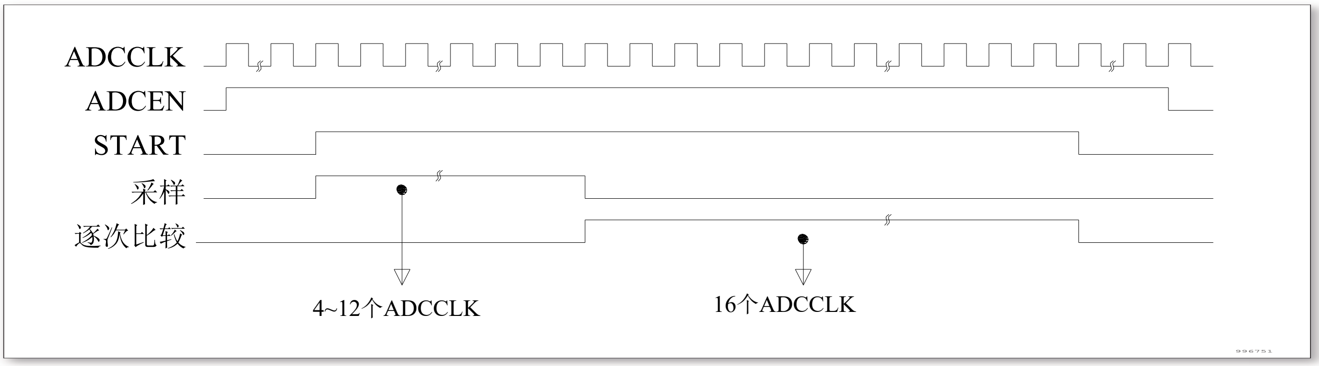

图 207. ADC 转换时序图

ADC 转换速度与 ADC 参考电压及 AVCC 电压相关，最高转换速度如表 115所示：

表 115: ADC 最高转换速度

ADC 参考电压	AVCC 电压	最高转换速度	最大 ADCCLK 频率
内部 1.5V	2.8V~3.6V	200K SPS	4MHz
内部 2.5V	2.8V~3.6V	200K SPS	4MHz
AVCC / ExRef	2.8V~3.6V	1M SPS	24MHz

28.4 单次转换模式

在单次转换模式下，ADC 启动后只执行一次转换，可对所有的 24 路 ADC 通道进行转换。该模式既可通过设置 ADC_SglStart.Start 位启动也可通过设置 ADC_ExtTrigger0 的外部触发启动。一旦选定通道的 ADC 转换完成，ADC_IFR.SGLIF 位会自动置 1，转换结果保存在 ADC_Result 寄存器中。

通过 ADC_SglStart.Start 位启动 ADC 单次转换操作流程：

Step1：配置 PAADS~PEADS 相应的位，将待转换的 ADC 通道配置为模拟端口。

Step2：设置 PBADS.bit1 为 1，将 ADC 外部参考电压引脚配置为模拟端口。

注：如果 ADC 参考电压不选择外部参考电压引脚，则可以略过本步骤。

Step3：设置 BGR_CR.BGR_EN 为 1，使能 BGR 模块。

Step4：设置 ADC_CR0.En 为 1，使能 ADC 模块。

Step5：延时 20us，等待 ADC 及 BGR 模块启动完成。

Step6：设置 ADC_CR1.Mode 为 0，选择单次转换模式。

Step7：配置 ADC_CR0.Ref，选择 ADC 的参考电压。

Step8：设置 ADC_CR0.InRefEn 为 1，使能 ADC 内部参考电压。

注：如果 ADC 参考电压不选择内部参考电压，则可以略过本步骤。

Step9：配置 ADC_CR0.SAM 及 ADC_CR0.CkDiv，设置 ADC 的转换速度。

Step10：配置 ADC_CR0.SGLMux，选择待转换的通道。

Step11：设置 ADC_ICR.SGLIC 为 0，清除 ADC_IFR.SGLIF 标志。

Step12：设置 ADC_SglStart.Start 为 1，启动 ADC 单次转换。

Step13：等待 ADC_IFR. SGLIF 变为 1，读取 ADC_Result 寄存器以获取 ADC 转换结果。

Step14：如需对其它通道进行转换，重复执行 Step10~Step13。

Step15：设置 ADC_CR0.En 及 BGR_CR.BGR_EN 为 0，关闭 ADC 模块、BGR 模块。

通过外部触发启动 ADC 单次转换操作流程：

Step1：配置 PAADS~PEADS 相应的位，将待转换的 ADC 通道配置为模拟端口。

Step2：设置 PBADS.bit1 为 1，将 ADC 外部参考电压引脚配置为模拟端口。

注：如果 ADC 参考电压不选择外部参考电压引脚，则可以略过本步骤。

Step3：设置 BGR_CR.BGR_EN 为 1，使能 BGR 模块。

Step4：设置 ADC_CR0.En 为 1，使能 ADC 模块。

Step5：延时 20us，等待 ADC 及 BGR 模块启动完成。

Step6：设置 ADC_CR1.Mode 为 0，选择单次转换模式。

Step7：设置 ADC_CR0.IE 为 1，使能 ADC 中断。

Step8：使能 NVIC 中断向量表中的 ADC 中断。

Step9：配置 ADC_CR0.Ref，选择 ADC 的参考电压。

Step10：设置 ADC_CR0.InRefEn 为 1，使能 ADC 内部参考电压。

注：如果 ADC 参考电压不选择内部参考电压，则可以略过本步骤。

Step11：配置 ADC_CR0.SAM 及 ADC_CR0.CkDiv，设置 ADC 的转换速度。

Step12：配置 ADC_CR0.SGLMux，选择待转换的通道。

Step13：设置 ADC_IFR 为 0x0，清除 ADC 中断标志。

Step14：配置 ADC_ExtTrigger0，选择外部触发条件。

Step15：当外部触发条件触发 ADC 完成转换时，ADC 模块会产生中断。用户可在 ADC 中断服务程序中读取 ADC_Result 寄存器以获取 ADC 转换结果。

Step16：如需对其它通道进行转换，重复执行 Step12~Step15。

Step17：设置 ADC_CR0.En 及 BGR_CR.BGR_EN 为 0，关闭 ADC 模块、BGR 模块。

28.5 扫描转换模式

扫描转换模式分为顺序扫描转换和插队扫描转换两种模式。两种模式各自单独工作时，均可连续对多个通道进行多次转换；当两种模式同时工作时，则优先对插队扫描配置的通道进行转换。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

28.5.1 顺序扫描转换模式

顺序扫描转换模式最多可进行 16 次连续转换，转换的总次数由 ADC_SQR2.CNT 进行配置；可配置所有 24 个通道进行转换，待转换通道由 ADC_SQRx.CHxMux 进行配置。该模式既可通过设置 ADC_SqrStart.Start 位启动也可通过设置 ADC_ExtTrigger0 的外部触发启动。启动转换后，ADC 模块依次转换 CHxMux~CH0Mux 中配置的通道直到总转换次数完成。ADC 模块完成总转换次数后，ADC_IFR.SQRIF 位会自动置 1，转换结果保存在转换通道所对应的 ADC_SqrResultx~ADC_SqrResult0 寄存器中。图 208演示了对 AIN0、AIN1、AIN5 进行 8 次转换的顺序扫描转换过程。其中顺序扫描转换通道 7，4，1 配置为 AIN1，转换通道 6，3，0 配置为 AIN0，转换通道 5，2 配置为 AIN5。ADC_SqrStart.Start 置 1 后，ADC 模块会依次对顺序扫描转换通道 7~0 进行转换。

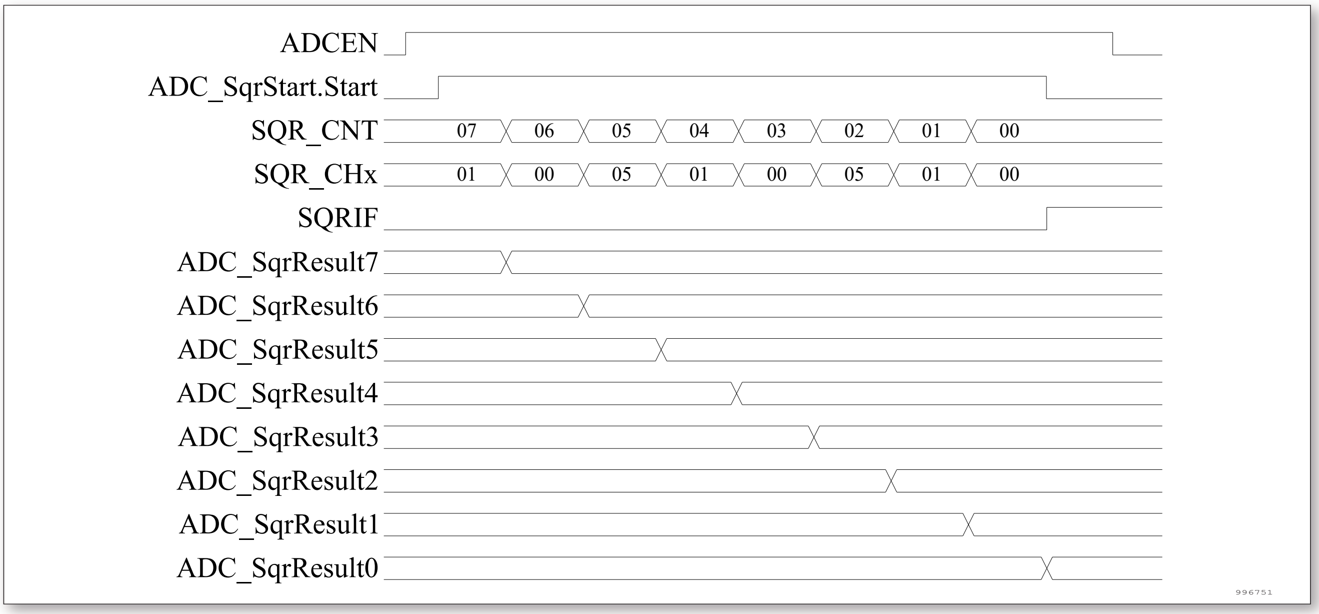

图 208. ADC 顺序扫描转换过程示例

通过 ADC_SqrStart.Start 位启动 ADC 顺序扫描转换操作流程：

Step1：配置 PAADS~PEADS 相应的位，将待转换的 ADC 通道配置为模拟端口。

Step2：设置 PBADS.bit1 为 1，将 ADC 外部参考电压引脚配置为模拟端口。

注：如果 ADC 参考电压不选择外部参考电压引脚，则可以略过本步骤。

Step3：设置 BGR_CR.BGR_EN 为 1，使能 BGR 模块。

Step4：设置 ADC_CR0.En 为 1，使能 ADC 模块。

Step5：延时 20us，等待 ADC 及 BGR 模块启动完成。

Step6：设置 ADC_CR1.Mode 为 1，选择扫描转换模式。

Step7：配置 ADC_CR0.Ref，选择 ADC 的参考电压。

Step8：设置 ADC_CR0.InRefEn 为 1，使能 ADC 内部参考电压。

注：如果 ADC 参考电压不选择内部参考电压，则可以略过本步骤。

Step9：配置 ADC_CR0.SAM 及 ADC_CR0.CkDiv，设置 ADC 的转换速度。

Step10：配置 ADC_SQRx.CHxMux，选择顺序扫描转换通道。

Step11：配置 ADC_SQR2.CNT，选择顺序扫描转换的总转换次数。

注：总转换次数需与 Step9 中配置的转换通道数保持一致。

Step12：设置 ADC_ICR.SQRIC 为 0，清除 ADC_IFR.SQRIF 标志。

Step13：设置 ADC_SqrStart.Start 为 1，启动 ADC 顺序扫描转换。

Step14：等待 ADC_IFR.SQRIF 变为 1，读取 ADC_SqrResultx ~ ADC_SqrResult0 寄存器以获取相应通道的转换结果。

Step15：如需对其它通道进行转换，重复执行 Step10~Step14。

Step16：设置 ADC_CR0.En 及 BGR_CR.BGR_EN 为 0，关闭 ADC 模块、BGR 模块。

通过外部触发启动 ADC 顺序扫描转换操作流程：

Step1：配置 PAADS~PEADS 相应的位，将待转换的 ADC 通道配置为模拟端口。

Step2：设置 PBADS.bit1 为 1，将 ADC 外部参考电压引脚配置为模拟端口。

注：如果 ADC 参考电压不选择外部参考电压引脚，则可以略过本步骤。

Step3：设置 BGR_CR.BGR_EN 为 1，使能 BGR 模块。

Step4：设置 ADC_CR0.En 为 1，使能 ADC 模块。

Step5：延时 20us，等待 ADC 及 BGR 模块启动完成。

Step6：设置 ADC_CR1.Mode 为 1，选择扫描转换模式。

Step7：设置 ADC_CR0.IE 为 1，使能 ADC 中断。

Step8：使能 NVIC 中断向量表中的 ADC 中断。

Step9：配置 ADC_CR0.Ref，选择 ADC 的参考电压。

Step10：设置 ADC_CR0.InRefEn 为 1，使能 ADC 内部参考电压。

注：如果 ADC 参考电压不选择内部参考电压，则可以略过本步骤。

Step11：配置 ADC_CR0.SAM 及 ADC_CR0.CkDiv，设置 ADC 的转换速度。

Step12：配置 ADC_SQRx.CHxMux，选择顺序扫描转换通道。

Step13：配置 ADC_SQR2.CNT，选择顺序扫描转换的总转换次数。

注：总转换次数需与 Step11 中配置的转换通道数保持一致。

Step14：设置 ADC_IFR 为 0x0，清除 ADC 中断标志。

Step15：配置 ADC_ExtTrigger0，选择外部触发条件。

Step16：当外部触发条件触发 ADC 完成转换时，ADC 模块会产生中断。用户可在 ADC 中断服务程序中读取 ADC_SqrResultx ~ ADC_SqrResult0 寄存器以获取相应通道的转换结果。

Step17：如需对其它通道进行转换，重复执行 Step12~Step16。

Step18：设置 ADC_CR0.En 及 BGR_CR.BGR_EN 为 0，关闭 ADC 模块、BGR 模块。

28.5.2 插队扫描转换模式

插队扫描转换模式最多可进行 4 次连续转换，转换的总次数由 ADC_JQR.CNT 进行配置；可配置所有 24 个通道进行转换，待转换通道由 ADC_JQR.CHxMux 进行配置。该模式既可通过设置 ADC_JqrStart.Start 位启动也可通过设置 ADC_ExtTrigger1 的外部触发启动。启动转换后，ADC 模块依次转换 CHxMux~CH0Mux 中配置的通道直到总转换次数完成。ADC 模块完成总转换次数后，ADC_IFR.JQRIF 位会自动置 1，转换结果保存在转换通道所对应的 ADC_JqrResultx~ ADC_JqrResult0 寄存器中。图 209演示了对 AIN0、AIN1、AIN5 进行 4 次转换的插

队扫描转换的过程。其中插队扫描转换通道 3，2，1，0 分别设置为 AIN5，AIN0，AIN1，AIN5。ADC_JqrStart.Start 置 1 后，ADC 模块会依次对插队扫描转换通道 3~0 进行转换。

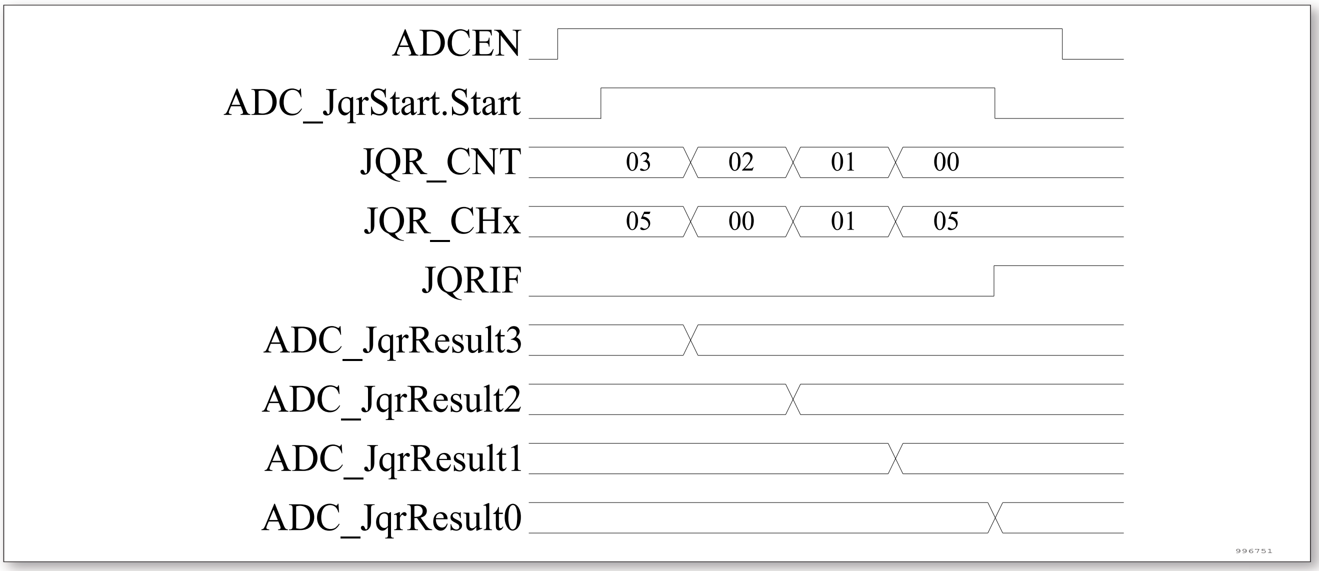

图 209. ADC 插队扫描转换过程示例

通过 ADC_JqrStart.Start 位启动 ADC 插队扫描转换操作流程：

Step1：配置 PAADS~PEADS 相应的位，将待转换的 ADC 通道配置为模拟端口。

Step2：设置 PBADS.bit1 为 1，将 ADC 外部参考电压引脚配置为模拟端口。

注：如果 ADC 参考电压不选择外部参考电压引脚，则可以略过本步骤。

Step3：设置 BGR_CR.BGR_EN 为 1，使能 BGR 模块。

Step4：设置 ADC_CR0.En 为 1，使能 ADC 模块。

Step5：延时 20us，等待 ADC 及 BGR 模块启动完成。

Step6：设置 ADC_CR1.Mode 为 1，选择扫描转换模式。

Step7：配置 ADC_CR0.Ref，选择 ADC 的参考电压。

Step8：设置 ADC_CR0.InRefEn 为 1，使能 ADC 内部参考电压。

注：如果 ADC 参考电压不选择内部参考电压，则可以略过本步骤。

Step9：配置 ADC_CR0.SAM 及 ADC_CR0.CkDiv，设置 ADC 的转换速度。

Step10：配置 ADC_JQR.CHxMux，选择插队扫描转换通道。

Step11：配置 ADC_JQR.CNT，选择插队扫描转换的总转换次数。

注：总转换次数需与 Step9 中配置的转换通道数保持一致。

Step12：设置 ADC_ICR.JQRIC 为 0，清除 ADC_IFR.JQRIF 标志。

Step13：设置 ADC_JqrStart.Start 为 1，启动 ADC 插队扫描转换。

Step14：等待 ADC_IFR.JQRIF 变为 1，读取 ADC_JqrResultx~ ADC_JqrResult0 寄存器以获取相应通道的转换结果。

Step15：如需对其它通道进行转换，重复执行 Step10~Step14。

Step16：设置 ADC_CR0.En 及 BGR_CR.BGR_EN 为 0，关闭 ADC 模块、BGR 模块。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

通过外部触发启动 ADC 插队扫描转换操作流程：

Step1：配置 PAADS~PEADS 相应的位，将待转换的 ADC 通道配置为模拟端口。

Step2：设置 PBADS.bit1 为 1，将 ADC 外部参考电压引脚配置为模拟端口。

注：如果 ADC 参考电压不选择外部参考电压引脚，则可以略过本步骤。

Step3：设置 BGR_CR.BGR_EN 为 1，使能 BGR 模块。

Step4：设置 ADC_CR0.En 为 1，使能 ADC 模块。

Step5：延时 20us，等待 ADC 及 BGR 模块启动完成。

Step6：设置 ADC_CR1.Mode 为 1，选择扫描转换模式。

Step7：设置 ADC_CR0.IE 为 1，使能 ADC 中断。

Step8：使能 NVIC 中断向量表中的 ADC 中断。

Step9：配置 ADC_CR0.Ref，选择 ADC 的参考电压。

Step10：设置 ADC_CR0.InRefEn 为 1，使能 ADC 内部参考电压。

注：如果 ADC 参考电压不选择内部参考电压，则可以略过本步骤。

Step11：配置 ADC_CR0.SAM 及 ADC_CR0.CkDiv，设置 ADC 的转换速度。

Step12：配置 ADC_JQR.CHxMux，选择插队扫描转换通道。

Step13：配置 ADC_JQR.CNT，选择插队扫描转换的总转换次数。

注：总转换次数需与 Step11 中配置的转换通道数保持一致。

Step14：设置 ADC_IFR 为 0x0，清除 ADC 中断标志。

Step15：配置 ADC_ExtTrigger1，选择外部触发条件。

Step16：当外部触发条件触发 ADC 完成转换时，ADC 模块会产生中断。用户可在 ADC 中断服务程序中读取 ADC_JqrResultx~ ADC_JqrResult0 寄存器以获取相应通道的转换结果。

Step17：如需对其它通道进行转换，重复执行 Step12~Step16。

Step18：设置 ADC_CR0.En 及 BGR_CR.BGR_EN 为 0，关闭 ADC 模块、BGR 模块。

插队扫描转换的执行优先级高于顺序扫描转换，当顺序扫描转换正在进行时，如果插队扫描转换启动，则顺序扫描在完成当前通道的转换后暂停，等到插队扫描转换完成后，再继续执行剩下的通道转换。图 210演示了对 AIN0，AIN1，AIN5，AIN8 进行顺序扫描转换时，启动对 AIN2，AIN6 进行插队扫描转换的过程。其中顺序扫描转换通道 3，2，1，0 分别设置为 AIN1，AIN0，AIN5，AIN8，插队扫描转换通道 1，0 分别设置为 AIN6，AIN2。在顺序扫描进行 AIN0 转换的过程中，启动了插队扫描，顺序扫描会将当前 AIN0 的转换完成然后暂停，等到插队扫描完成 AIN6 和 AIN2 的转换后，顺序扫描再进行 AIN5 和 AIN8 的转换。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

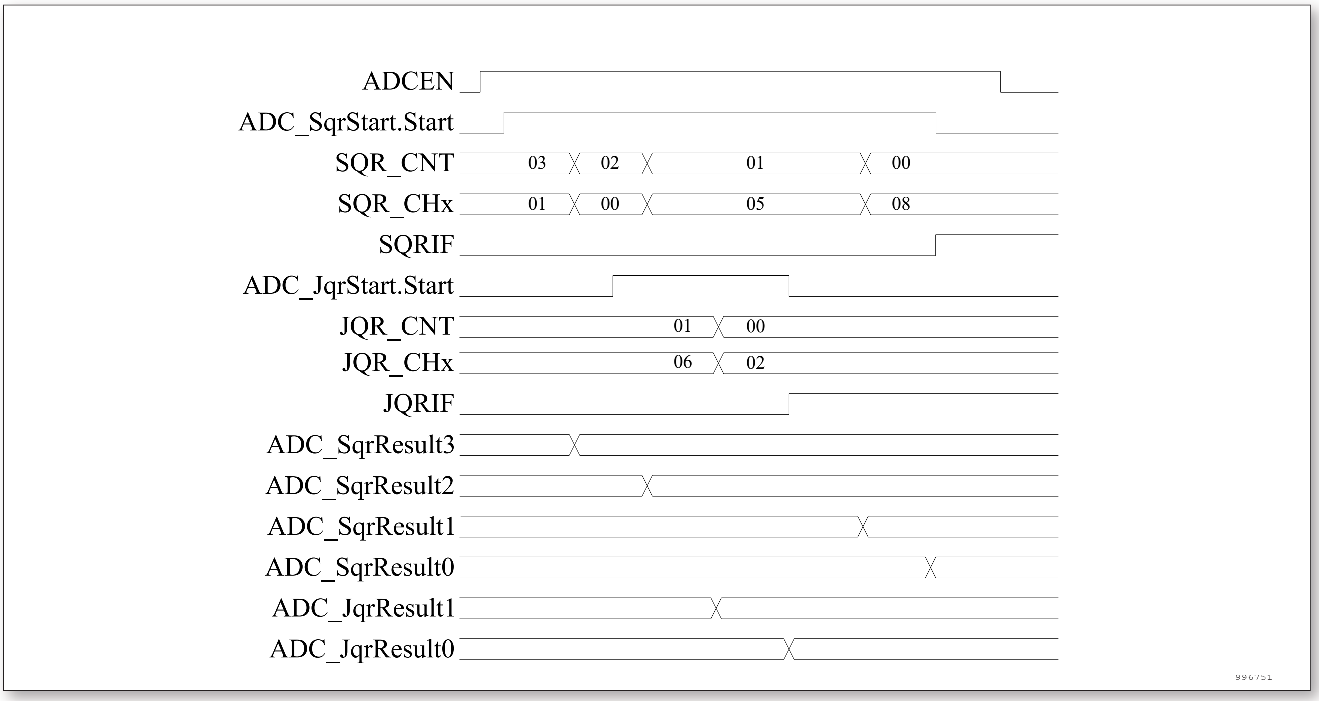

图 210. ADC 顺序扫描过程中进行插队扫描示例

28.5.3 扫描转换触发 DMA 读取

顺序扫描和插队扫描转换完成后，可自动触发 DMA 读取转换结果。顺序扫描模式下，通过配置 ADC_CR1. DmaSqr 为 1 使能该功能；插队扫描模式下，通过配置 ADC_CR1. DmaJqr 为 1 使能该功能。

28.6 连续转换累加模式

在连续转换累加模式下，启动一次 ADC 可对多个通道进行多次转换并对每次转换的结果进行累加；可配置所有 24 个通道进行转换。ADC 转换的总次数由 ADC_SQR2.CNT 进行配置；待转换通道由 ADC_SQRx.CHxMux 进行配置。该模式既可通过设置 ADC_SqrStart.Start 位启动也可通过设置 ADC_ExtTrigger0 的外部触发启动。启动连续转换后，ADC 模块依次转换 CHxMux~CH0Mux 中配置的通道直到总转换次数完成。ADC 模块完成总转换次数后，ADC_IFR.SQRIF 位会自动置 1，转换结果的累加值保存在 ADC_ResultAcc 寄存器中。

图 211演示了对 AIN0、AIN1、AIN5 进行 10 次连续转换累加的过程。顺序扫描转换通道 9，6，3，0 配置为 AIN0，转换通道 8，5，2 配置为 AIN1，转换通道 7，4，1 配置为 AIN5。ADC_SqrStart.Start 置 1 后，ADC 模块会按照顺序扫描转换通道配置依次进行转换，直到 SQR_CNT 的计数值变为 0。每次转换完成时，ADC_ResultAcc 寄存器都会自动进行累加。图中给定的 AIN0、AIN1、AIN5 的转换结果依次为 0x010、0x020、0x040。

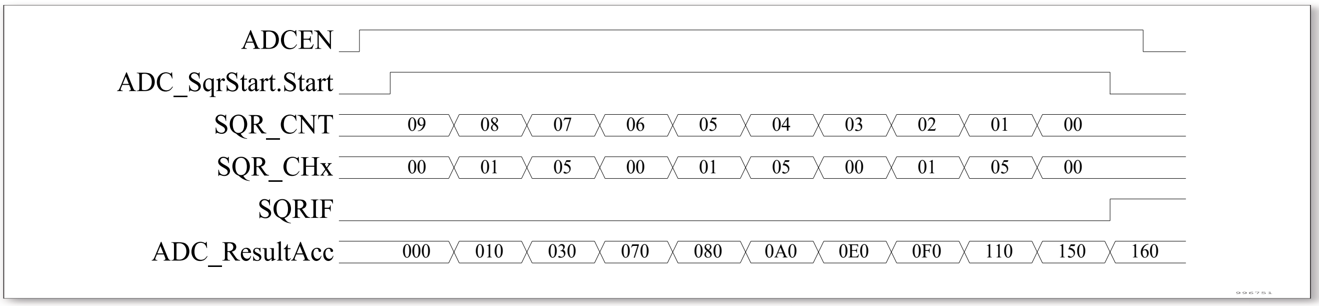

图 211. ADC 连续转换累加过程示例

通过 ADC_SqrStart.Start 位启动 ADC 连续转换累加操作流程：

Step1：配置 PAADS~PEADS 相应的位，将待转换的 ADC 通道配置为模拟端口。

Step2：设置 PBADS.bit1 为 1，将 ADC 外部参考电压引脚配置为模拟端口。

注：如果 ADC 参考电压不选择外部参考电压引脚，则可以略过本步骤。

Step3：设置 BGR_CR.BGR_EN 为 1，使能 BGR 模块。

Step4：设置 ADC_CR0.En 为 1，使能 ADC 模块。

Step5：延时 20us，等待 ADC 及 BGR 模块启动完成。

Step6：设置 ADC_CR1.Mode 为 1，选择扫描转换模式。

Step7：设置 ADC_CR1.RAccEn 为 1，使能 ADC 转换自动累加功能。

Step8：配置 ADC_CR0.Ref，选择 ADC 的参考电压。

Step9：设置 ADC_CR0.InRefEn 为 1，使能 ADC 内部参考电压。

注：如果 ADC 参考电压不选择内部参考电压，则可以略过本步骤。

Step10：配置 ADC_CR0.SAM 及 ADC_CR0.CkDiv，设置 ADC 的转换速度。

Step11：配置 ADC_SQRx.CHxMux，选择顺序扫描转换通道。

Step12：配置 ADC_SQR2.CNT，选择顺序扫描转换的总转换次数。

注：总转换次数需与 Step10 中配置的转换通道数保持一致。

Step13：设置 ADC_ICR.SQRIC 为 0，清除 ADC_IFR.SQRIF 标志。

Step14：设置 ADC_CR1.RAccClr 为 0，清零 ADC_ResultAcc 寄存器。

Step15：设置 ADC_SqrStart.Start 为 1，启动 ADC 顺序扫描转换。

Step16：等待 ADC_IFR.SQRIF 变为 1，读取 ADC_ResultAcc 寄存器以获取转换结果累加值。

Step17：如需对其它通道进行转换，重复执行 Step11~Step16。

Step18：设置 ADC_CR0.En 及 BGR_CR.BGR_EN 为 0，关闭 ADC 模块、BGR 模块。

通过外部触发启动 ADC 连续转换累加操作流程：

Step1：配置 PAADS~PEADS 相应的位，将待转换的 ADC 通道配置为模拟端口。

Step2：设置 PBADS.bit1 为 1，将 ADC 外部参考电压引脚配置为模拟端口。

注：如果 ADC 参考电压不选择外部参考电压引脚，则可以略过本步骤。

Step3：设置 BGR_CR.BGR_EN 为 1，使能 BGR 模块。

Step4：设置 ADC_CR0.En 为 1，使能 ADC 模块。

Step5：延时 20us，等待 ADC 及 BGR 模块启动完成。

Step6：设置 ADC_CR1.Mode 为 1，选择扫描转换模式。

Step7：设置 ADC_CR0.IE 为 1，使能 ADC 中断。

Step8：使能 NVIC 中断向量表中的 ADC 中断。

Step9：设置 ADC_CR1.RAccEn 为 1，使能 ADC 转换自动累加功能。

Step10：配置 ADC_CR0.Ref，选择 ADC 的参考电压。

Step11：设置 ADC_CR0.InRefEn 为 1，使能 ADC 内部参考电压。

注：如果 ADC 参考电压不选择内部参考电压，则可以略过本步骤。

Step12：配置 ADC_CR0.SAM 及 ADC_CR0.CkDiv，设置 ADC 的转换速度。

Step13：配置 ADC_SQRx.CHxMux，选择顺序扫描转换通道。

Step14：配置 ADC_SQR2.CNT，选择顺序扫描转换的总转换次数。

注：总转换次数需与 Step12 中配置的转换通道数保持一致。

Step15：设置 ADC_IFR 为 0x0，清除 ADC 中断标志。

Step16：设置 ADC_CR1.RAccClr 为 0，清零 ADC_ResultAcc 寄存器。

Step17：配置 ADC_ExtTrigger0，选择外部触发条件。

Step18：当外部触发条件触发 ADC 完成转换时，ADC 模块会产生中断。用户可在 ADC 中断服务程序中读取 ADC_ResultAcc 寄存器以获取转换结果累加值。

Step19：如需对其它通道进行转换，重复执行 Step13~Step18。

Step20：设置 ADC_CR0.En 及 BGR_CR.BGR_EN 为 0，关闭 ADC 模块、BGR 模块。

28.7 ADC 转换外部触发源

ADC 转换既可通过软件配置启动，也可通过外部触发启动。

配置 ADC_ExtTrigger0 寄存器可设置 ADC 单次转换或顺序扫描转换的外部触发源。

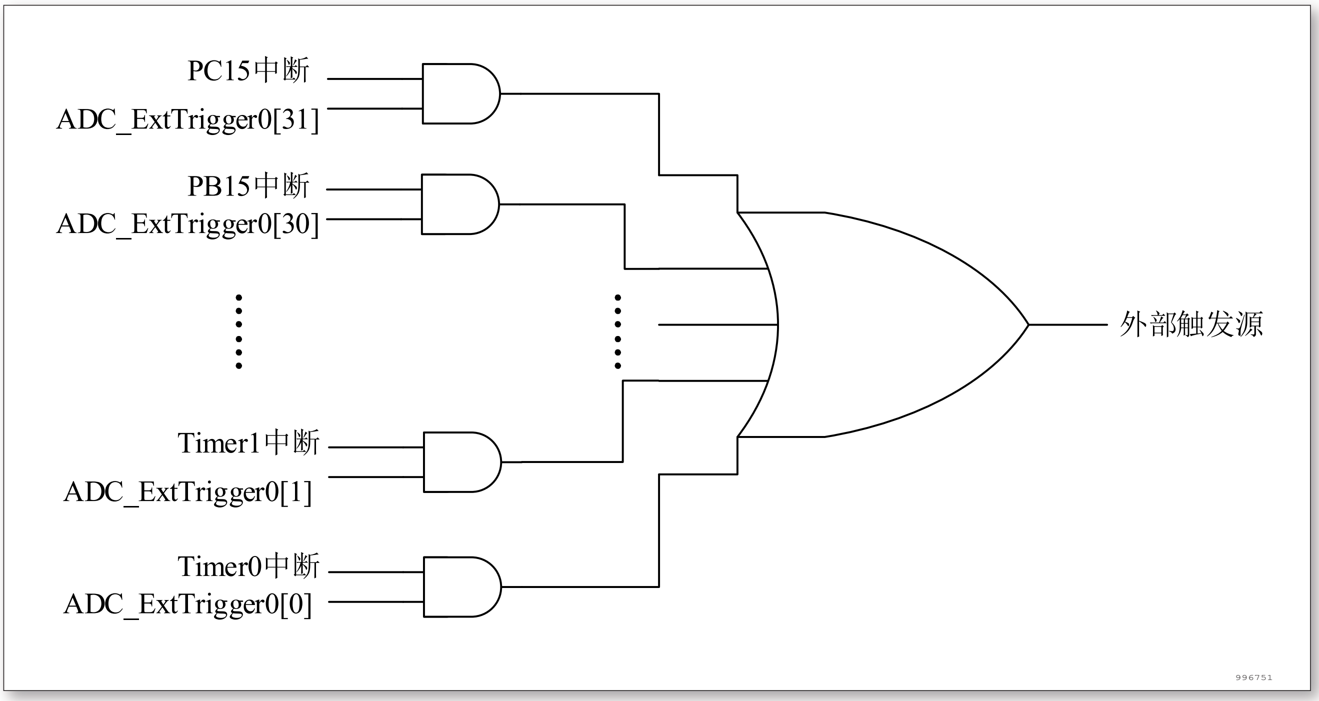

图 212. ADC 单次转换或顺序扫描转换外部触发源示意图

配置 ADC_ExtTrigger1 寄存器可设置 ADC 插队扫描转换的外部触发源。

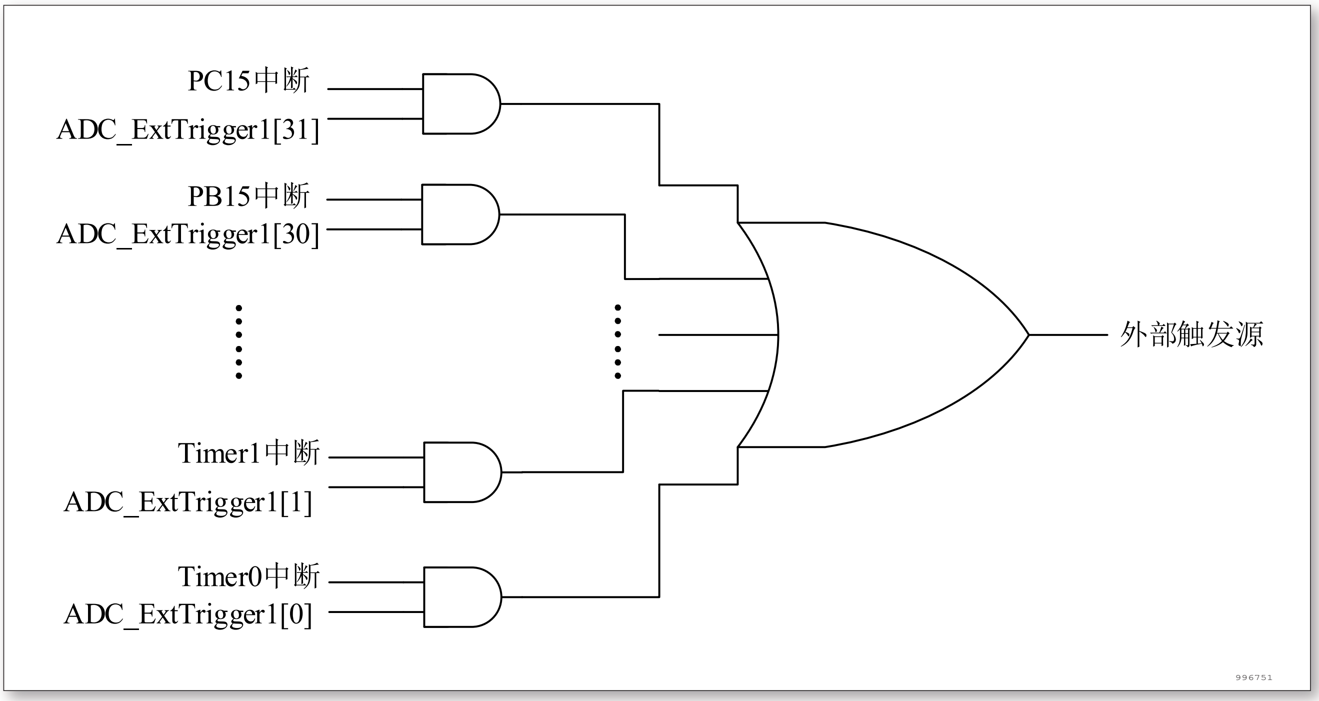

图 213. ADC 插队扫描转换外部触发源示意图

28.8 ADC 转换结果比较

ADC 转换完成时，ADC 转换结果可以与用户设定的阈值进行比较，支持上阈值比较、下阈值比较、区间值比较。该功能需要将相应的控制位 HtCmp、LtCmp、RegCmp 置 1。该功能可实现对模拟量的自动监测，直到 ADC 转换结果符合用户预期时才产生中断申请用户程序界入。监测通道选择通过 ADC_CR1.ThCh 进行配置。

上阈值比较：当 ADC 转换结果位于 [ADC_HT，4095] 区间内则 ADC_IFR.HTIF 置 1；向 ADC_ICR.HTIC 写入 0 则清零 ADC_IFR.HTIF。

下阈值比较：当 ADC 转换结果位于 [0,ADC_LT) 区间内则 ADC_IFR.LTIF 置 1；向 ADC_ICR.LTIC 写入 0 则清零 ADC_IFR.LTIF。

区间值比较：当 ADC 转换结果位于 [ADC_LT,ADC_HT) 区间内则 ADC_IFR.REGIF 置 1;向 ADC_ICR.REGIC 写入 0 则清零 ADC_IFR.REGIF。

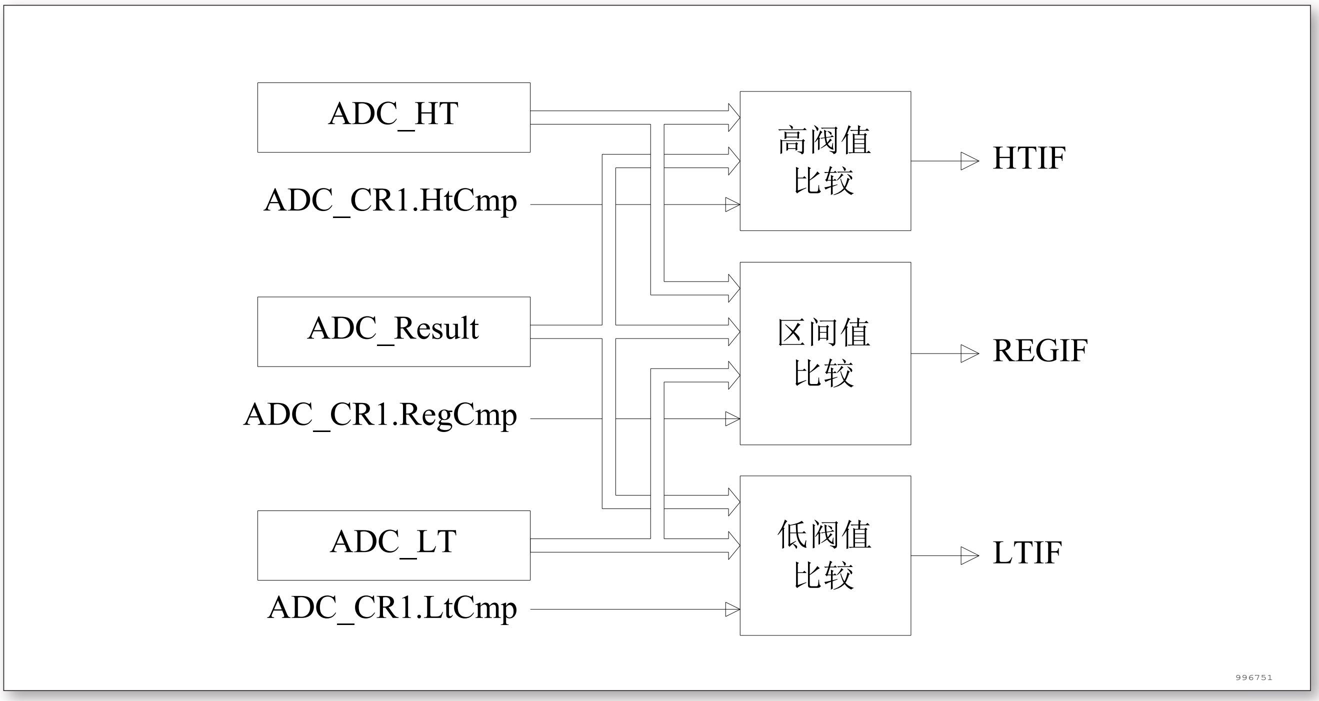

图 214. ADC 转换结果比较

28.9 ADC 中断

ADC 中断请求如表所示：

表 116: ADC 中断请求

中断源	中断标志	中断使能
ADC 插队扫描转换完成	ADC_IFR.JQRIF	ADC_CR0.IE
ADC 顺序扫描转换完成	ADC_IFR.SQRIF	
ADC 转换结果位于区间值区域（大于等于下阈值小于上阈值）	ADC_IFR.REGIF	
ADC 转换结果位于上阈值区域 (大于等于上阈值)	ADC_IFR.HTIF	
ADC 转换结果比较下阈值区域（小于下阈值）	ADC_IFR.LTIF	
ADC 单次转换完成	ADC_IFR.SGLIF	

28.10 使用温度传感器测量环境温度

温度传感器的输出电压会随环境温度的改变而变化，故根据温度传感器的输出电压即可计算出相应的环境温度。当 ADC 模块的测量通道选择温度传感器的输出电压时，即可测量环境温度。

计算公式如下：

环境温度 = 25 + 0.0839 × Vref × (AdcValue –Trim)

其中：Vref 为当前 ADC 模块的参考电压，取值为 1.5 或 2.5。

AdcValue 为 ADC 模块测量温度传感器输出电压的结果，取值为 0~4095。

Trim 为 16Bits 的校准值，计算时需要从 Flash 存储器中读出，其存放地址详见表 117。

表 117: 16Bits 的校准值

ADC 参考电压	校准值存放地址	校准值精度
内部 1.5V	0x00100C34	±3 °C
内部 2.5V	0x00100C36	±3 °C

计算示例如下：

条件 1：Vref=2.5、AdcValue=0x7E5、Trim=0x76C：

温度 1：25 + 0.0839 × 2.5 × (0x7E5 -0x76C) = 50°C。

条件 2：Vref=1.5、AdcValue=0x72D、Trim=0x76C：

温度 3：25 + 0.0839 × 1.5 × (0x72D -0x76C) = 17°C。

通过 ADC 测量环境温度操作流程：

Step1：设置 BGR_CR 为 3，使能 BGR 模块和温度传感器模块。

Step2：设置 ADC_CR0.En 为 1，使能 ADC 模块。

Step3：延时 20us，等待 ADC 及 BGR 模块启动完成。

Step4：设置 ADC_CR1.Mode 为 0，选择单次转换模式。

Step5: 设置 ADC_CR0.InRefEn 为 1，使能 ADC 内部参考电压。

Step6：配置 ADC_CR0.Ref，选择 ADC 的参考电压为内部 1.5V 或内部 2.5V。

Step7：配置 ADC_CR0.SAM 及 ADC_CR0.CkDiv，设置 ADC 的转换速度。

Step8：设置 ADC_CR0.SGLMux 为 0x1C，选择待转换的通道为温度传感器的输出。

Step9：设置 ADC_CR0.Buf 为 1，使能输入信号放大器。

Step10：设置 ADC_ICR.SGLIC 为 0，清除 ADC_IFR.SGLIF 标志。

Step11：设置 ADC_SglStart.Start 为 1，启动 ADC 单次转换。

Step12：等待 ADC_IFR.SGLIF 变为 1，读取 ADC_Result 寄存器以获取 ADC 转换结果。

Step13：设置 ADC_CR0.En 及 BGR_CR 为 0，关闭 ADC 模块、BGR 模块、温度传感器模块。

Step14：读取温度传感器校准值，根据公式计算当前的环境温度。

28.11 ADC 模块寄存器

表 118. ADC 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x004	ADC_CR0	ADC 基本配置寄存器 0	0x0000 67F0	小节 28.11.1
0x008	ADC_CR1	ADC 基本配置寄存器 1	0x0000 8000	小节 28.11.2
0x040	ADC_SQR0	ADC 顺序扫描转换通道配置寄存器 0	0x0000 0000	小节 28.11.3

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Offset	Acronym	Register Name	Reset	Section
0x044	ADC_SQR1	ADC 顺序扫描转换通道配置寄存器 1	0x0000 0000	小节 28.11.4
0x048	ADC_SQR2	ADC 顺序扫描转换通道配置寄存器 2	0x0000 0000	小节 28.11.5
0x04C	ADC_JQR	ADC 插队扫描转换通道配置寄存器	0x0000 0000	小节 28.11.6
0x050	ADC_SqrResult0	ADC 顺序扫描转换通道 0 转换结果	0x0000 0000	小节 28.11.7
0x054	ADC_SqrResult1	ADC 顺序扫描转换通道 1 转换结果	0x0000 0000	小节 28.11.7
0x058	ADC_SqrResult2	ADC 顺序扫描转换通道 2 转换结果	0x0000 0000	小节 28.11.7
0x05C	ADC_SqrResult3	ADC 顺序扫描转换通道 3 转换结果	0x0000 0000	小节 28.11.7
0x060	ADC_SqrResult4	ADC 顺序扫描转换通道 4 转换结果	0x0000 0000	小节 28.11.7
0x064	ADC_SqrResult5	ADC 顺序扫描转换通道 5 转换结果	0x0000 0000	小节 28.11.7
0x068	ADC_SqrResult6	ADC 顺序扫描转换通道 6 转换结果	0x0000 0000	小节 28.11.7
0x06C	ADC_SqrResult7	ADC 顺序扫描转换通道 7 转换结果	0x0000 0000	小节 28.11.7
0x070	ADC_SqrResult8	ADC 顺序扫描转换通道 8 转换结果	0x0000 0000	小节 28.11.7
0x074	ADC_SqrResult9	ADC 顺序扫描转换通道 9 转换结果	0x0000 0000	小节 28.11.7
0x078	ADC_SqrResult10	ADC 顺序扫描转换通道 10 转换结果	0x0000 0000	小节 28.11.7
0x07C	ADC_SqrResult11	ADC 顺序扫描转换通道 11 转换结果	0x0000 0000	小节 28.11.7
0x080	ADC_SqrResult12	ADC 顺序扫描转换通道 12 转换结果	0x0000 0000	小节 28.11.7
0x084	ADC_SqrResult13	ADC 顺序扫描转换通道 13 转换结果	0x0000 0000	小节 28.11.7
0x088	ADC_SqrResult14	ADC 顺序扫描转换通道 14 转换结果	0x0000 0000	小节 28.11.7
0x08C	ADC_SqrResult15	ADC 顺序扫描转换通道 15 转换结果	0x0000 0000	小节 28.11.7
0x090	ADC_JqrResult0	ADC 插队扫描转换通道 0 转换结果	0x0000 0000	小节 28.11.8
0x094	ADC_JqrResult1	ADC 插队扫描转换通道 1 转换结果	0x0000 0000	小节 28.11.8
0x098	ADC_JqrResult2	ADC 插队扫描转换通道 2 转换结果	0x0000 0000	小节 28.11.8
0x09C	ADC_JqrResult3	ADC 插队扫描转换通道 3 转换结果	0x0000 0000	小节 28.11.8
0x0A0	ADC_Result	ADC 转换结果	0x0000 0000	小节 28.11.9
0x0A4	ADC_ResultAcc	ADC 转换结果累加值	0x0000 0000	小节 28.11.10
0x0A8	ADC_HT	ADC 比较上阈值	0x0000 0FFF	小节 28.11.11
0x0AC	ADC_LT	ADC 比较下阈值	0x0000 0000	小节 28.11.12
0x0B0	ADC_IFR	ADC 中断标志寄存器	0x0000 0000	小节 28.11.13
0x0B4	ADC_ICR	ADC 中断清除寄存器	0x0000 003F	小节 28.11.14

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Offset	Acronym	Register Name	Reset	Section
0x0B8	ADC_ExtTrigger0	ADC 单次转换或顺序扫描转换外部中断触发源配置寄存器	0x0000 0000	小节 28.11.15
0x0BC	ADC_ExtTrigger1	ADC 插队扫描转换外部中断触发源配置寄存器	0x0000 0000	小节 28.11.16
0x0C0	ADC_SglStart	ADC 单次转换启动控制寄存器	0x0000 0000	小节 28.11.17
0x0C4	ADC_SqrStart	ADC 顺序扫描转换启动控制寄存器	0x0000 0000	小节 28.11.18
0x0C8	ADC_JqrStart	ADC 插队扫描转换启动控制寄存器	0x0000 0000	小节 28.11.19
0x0CC	ADC_allstart	ADC 一直转换启动停止控制寄存器	0x0000 0000	小节 28.11.20

28.11.1 ADC 基本配置寄存器 0 (ADC_CR0)

起始地址：0x40002400

地址偏移：0x004

复位值：0x0000 67F0

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IE	InRefEn	SAM		Buf	Ref		SGLMux					CkDiv		RES.	En
RW	RW	RW		RW	RW		RW					RW			RW

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15	IE	rw	0x00	ADC 中断控制 1：使能中断 0：禁止中断
14	InRefEn	rw	0x01	ADC 内部参考电压使能 1：使能内部参考电压 0：禁止内部参考电压
13:12	SAM	rw	0x02	ADC 采样周期选择 00：4 个转换周期 01：6 个转换周期 10：8 个转换周期 11：12 个转换周期

Bit	Field	Type	Reset	Description
11	Buf	rw	0x00	ADC 输入信号放大器控制 0：关闭放大器, 外部输入信号与 ADC 直接相连。 1：打开放大器, 外部输入信号通过放大器放大后与 ADC 相连，用于高阻信号。 以下几种情况需要打开 BUF 功能，使用 BUF 功能时，最大速率 200k bps 1）外部驱动很弱信号 2）测量 1/3AVCC 3）测量 temp sensor 4）测量 VREF1P2
10:9	Ref	rw	0x03	ADC 参考电压选择 00：内部 1.5V 01：内部 2.5V 10：外部参考电压 ExRef（PB01） 11：AVCC 电压

Bit	Field	Type	Reset	Description
8:4	SGLMux	rw	0x1F	单次转换模式转换通道选择 00000：AIN0（PA00） 00001：AIN1（PA01） 00010：AIN2（PA02） 00011：AIN3（PA03） 00100：AIN4（PA04） 00101：AIN5（PA05） 00110：AIN6（PA06） 00111：AIN7（PA07） 01000：AIN8（PB00） 01001：AIN9（PB01） 01010：AIN10（PC00） 01011：AIN11（PC01） 01100：AIN12（PC02） 01101：AIN13（PC03） 01110：AIN14（PC04） 01111：AIN15（PC05） 10000：AIN16（PB02） 10001：AIN17（PB10） 10010：AIN18（PB11） 10011：AIN19（PB12） 10100：AIN20（PB13） 10101：AIN21（PB14） 10110：AIN22（PB15） 10111：AIN23（PE15） 11000：AIN24（PE14） 11001：AIN25（PE13）
3:2	CkDiv	rw	0x00	11010：DAC 输出　　注：ADC_CR0.Buf 必须为 1 11011：1/3VCC　　注：ADC_CR0.Buf 必须为 1 11100：内置温度传感器输出电压　　注：ADC_CR0.Buf 必须为 1 11101：内部基准 1.2V 输出电压　　注：ADC_CR0.Buf 必须为 1
1	Reserved			保留
0	En	rw	0x00	ADC 使能控制 1：使能 ADC 0：禁止 ADC

28.11.2　ADC 基本配置寄存器 1 (ADC_CR1)

起始地址：0x40002400

地址偏移：0x008

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

复位值：0x0000 8000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RAccClr	RegCmp	HtCmp	LtCmp	RAccEn	Mode	DmaJqr	DmaSqr	ThCh					Align	Reserved	
WO	RW	RW	RW	RW	RW	RW	RW	RW					RW		

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15	RAccClr	wo	0x01	ADC 转换结果累加寄存器清零 1：无作用； 0：ADC 转换结果累加寄存器（ADC_ResultAcc）清零。
14	RegCmp	rw	0x00	ADC 区间比较控制 1：使能区间比较 0：禁止区间比较
13	HtCmp	rw	0x00	ADC 高阈值比较控制 1：使能高阈值比较 0：禁止高阈值比较
12	LtCmp	rw	0x00	ADC 低阈值比较控制 1：使能低阈值比较 0：禁止低阈值比较
11	RAccEn	rw	0x00	ADC 转换结果自动累加控制 1：使能 ADC 转换结果自动累加功能 0：禁止 ADC 转换结果自动累加功能
10	Mode	rw	0x00	ADC 转换模式选择 1：扫描转换模式 0：单次转换模式
9	DmaJqr	rw	0x00	插队扫描触发 DMA 读取控制 1：使能插队扫描转换触发 DMA 读取 0：禁止插队扫描转换触发 DMA 读取
8	DmaSqr	rw	0x00	顺序扫描触发 DMA 读取控制 1：使能顺序扫描转换触发 DMA 读取 0：禁止顺序扫描转换触发 DMA 读取

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
7:3	ThCh	rw	0x00	阈值比较通道选择 00000：选择通道 0 进行阈值比较 00001：选择通道 1 进行阈值比较 00010：选择通道 2 进行阈值比较 …… 11101：选择通道 29 进行阈值比较
2	Align	rw	0x00	转换结果对齐控制 1：转换结果 16Bits 左对齐存储 0：转换结果 16Bits 右对齐存储
1:0	Reserved			保留

28.11.3 ADC 顺序扫描转换通道配置寄存器 0 (ADC_SQR0)

起始地址：0x40002400

地址偏移：0x040

复位值：0x0000 0000

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="2" rowspan="2">Reserved</td><td colspan="5">CH5Mux</td><td colspan="5">CH4Mux</td><td colspan="4">CH3Mux</td></tr>
<tr><td colspan="5">RW</td><td colspan="5">RW</td><td colspan="4">RW</td></tr>
</table>

<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td></td><td colspan="5">CH2Mux</td><td colspan="5">CH1Mux</td><td colspan="5">CH0Mux</td></tr>
<tr><td></td><td colspan="5">RW</td><td colspan="5">RW</td><td colspan="5">RW</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:30	Reserved			保留
29:25	CH5Mux	rw	0x00	顺序扫描转换通道 5 选择，设置参见 ADC_CR0.SGLMux
24:20	CH4Mux	rw	0x00	顺序扫描转换通道 4 选择，设置参见 ADC_CR0.SGLMux
19:15	CH3Mux	rw	0x00	顺序扫描转换通道 3 选择，设置参见 ADC_CR0.SGLMux
14:10	CH2Mux	rw	0x00	顺序扫描转换通道 2 选择，设置参见 ADC_CR0.SGLMux
9:5	CH1Mux	rw	0x00	顺序扫描转换通道 1 选择，设置参见 ADC_CR0.SGLMux
4:0	CH0Mux	rw	0x00	顺序扫描转换通道 0 选择，设置参见 ADC_CR0.SGLMux

28.11.4 ADC 顺序扫描转换通道配置寄存器 1 (ADC_SQR1)

起始地址：0x40002400

地址偏移：0x044

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved		CH11Mux					CH10Mux					CH9Mux			
		RW					RW					RW			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH8Mux					CH7Mux					CH6Mux				
	RW					RW					RW				

Bit	Field	Type	Reset	Description
31:30	Reserved			保留
29:25	CH11Mux	rw	0x00	顺序扫描转换通道 11 选择，设置参见 ADC_CR0.SGLMux
24:20	CH10Mux	rw	0x00	顺序扫描转换通道 10 选择，设置参见 ADC_CR0.SGLMux
19:15	CH9Mux	rw	0x00	顺序扫描转换通道 9 选择，设置参见 ADC_CR0.SGLMux
14:10	CH8Mux	rw	0x00	顺序扫描转换通道 8 选择，设置参见 ADC_CR0.SGLMux
9:5	CH7Mux	rw	0x00	顺序扫描转换通道 7 选择，设置参见 ADC_CR0.SGLMux
4:0	CH6Mux	rw	0x00	顺序扫描转换通道 6 选择，设置参见 ADC_CR0.SGLMux

28.11.5 ADC 顺序扫描转换通道配置寄存器 2 (ADC_SQR2)

起始地址：0x40002400

地址偏移：0x048

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved								CNT				CH15Mux			
								RW				RW			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH14Mux					CH13Mux					CH12Mux				
	RW					RW					RW				

Bit	Field	Type	Reset	Description
31:24	Reserved			保留
23:20	CNT	rw	0x00	顺序扫描转换次数
19:15	CH15Mux	rw	0x00	顺序扫描转换通道 15 选择，设置参见 ADC_CR0.SGLMux
14:10	CH14Mux	rw	0x00	顺序扫描转换通道 14 选择，设置参见 ADC_CR0.SGLMux

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
9:5	CH13Mux	rw	0x00	顺序扫描转换通道 13 选择，设置参见 ADC_CR0.SGL Mux
4:0	CH12Mux	rw	0x00	顺序扫描转换通道 12 选择，设置参见 ADC_CR0.SGL Mux

28.11.6 ADC 插队扫描转换通道配置寄存器 (ADC_JQR)

起始地址：0x40002400

地址偏移：0x04C

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved										CNT		CH3Mux			
										RW		RW			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH2Mux					CH1Mux					CH0Mux				
	RW					RW					RW				

Bit	Field	Type	Reset	Description
31:24	Reserved			保留
21:20	CNT	rw	0x00	插队扫描转换次数
19:15	CH3Mux	rw	0x00	插队扫描转换通道 3 选择，设置参见 ADC_CR0.SGLMux
14:10	CH2Mux	rw	0x00	插队扫描转换通道 2 选择，设置参见 ADC_CR0.SGLMux
9:5	CH1Mux	rw	0x00	插队扫描转换通道 1 选择，设置参见 ADC_CR0.SGLMux
4:0	CH0Mux	rw	0x00	插队扫描转换通道 0 选择，设置参见 ADC_CR0.SGLMux

28.11.7 ADC 顺序扫描转换通道 0 转换结果 (ADC_SqrResult0-15)

起始地址：0x40002400

地址偏移：0x050

复位值：0x0000 0000

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

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

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31:12	Reserved			保留
11:0	Result	ro	0x00	ADC 顺序扫描转换通道 x 转换结果

28.11.8 ADC 插队扫描转换通道 0 转换结果（ADC_JqrResult0-3）

起始地址：0x40002400

地址偏移：0x090

复位值：0x0000 0000

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

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

Bit	Field	Type	Reset	Description
31:12	Reserved			保留
11:0	Result	ro	0x00	ADC 插队扫描转换通道 x 转换结果

28.11.9 ADC 转换结果（ADC_Result）

起始地址：0x40002400

地址偏移：0x0A0

复位值：0x0000 0000

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

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

Bit	Field	Type	Reset	Description
31:12	Reserved			保留
11:0	Result	ro	0x00	ADC 转换结果

28.11.10 ADC 转换结果累加值（ADC_ResultAcc）

起始地址：0x40002400

地址偏移：0x0A4

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

复位值：0x0000 0000

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

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

Bit	Field	Type	Reset	Description
31:20	Reserved			保留
19:0	ResultAcc	ro	0x00	ADC 转换累加值

28.11.11 ADC 比较上阈值（ADC_HT）

起始地址：0x40002400

地址偏移：0x0A8

复位值：0x0000 0FFF

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

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

Bit	Field	Type	Reset	Description
31:12	Reserved			保留
11:0	HT	rw	0x00	ADC 转换结果比较上阈值

28.11.12 ADC 比较下阈值（ADC_LT）

起始地址：0x40002400

地址偏移：0x0AC

复位值：0x0000 0000

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

b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Reserved				LT											
				RW											

Bit	Field	Type	Reset	Description
31:12	Reserved			保留
11:0	LT	rw	0x00	ADC 转换结果比较下阈值

28.11.13 ADC 中断标志寄存器 (ADC_IFR)

起始地址：0x40002400

地址偏移：0x0B0

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										JQRIF	SQRIF	REGIF	HHT_INTF	LLT_INTF	S_INTF
										RO	RO	RO	RO	RO	RO

Bit	Field	Type	Reset	Description
31:6	Reserved			保留
5	JQRIF	ro	0x00	ADC 插队扫描转换完成标志 1：ADC 插队扫描转换完成 0：ADC 插队扫描转换未完成
4	SQRIF	ro	0x00	ADC 顺序扫描转换完成标志 1：ADC 顺序扫描转换完成 0：ADC 顺序扫描转换未完成
3	REGIF	ro	0x00	ADC 转换结果比较区间标志 1：ADC 转换结果位于 [ADC_LT， ADC_HT) 区间内 0：ADC 转换结果位于 [ADC_LT， ADC_HT) 区间外
2	HTIF	ro	0x00	ADC 转换结果比较上阈值标志 1：ADC 转换结果位于 [ADC_HT，4095] 区间内 0：ADC 转换结果位于 [ADC_HT，4095] 区间外

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
1	LTIF	ro	0x00	ADC 转换结果比较下阈值标志 1：ADC 转换结果位于 [0，ADC_LT) 区间内 0：ADC 转换结果位于 [0，ADC_LT) 区间外
0	SGLIF	ro	0x00	ADC 单次转换完成标志 1：ADC 单次转换完成 0：ADC 单次转换未完成

28.11.14 ADC 中断清除寄存器 (ADC_ICR)

起始地址：0x40002400

地址偏移：0x0B4

复位值：0x0000 003F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										JQRIC	SQRIC	REGIC	HHT_INTC	LLT_INTC	S_INTC
										R1W0	R1W0	R1W0	R1W0	R1W0	R1W0

Bit	Field	Type	Reset	Description
31:6	Reserved			保留
5	JQRIC	r1w0	0x01	写 0 清除 ADC 插队扫描转换完成标志 写 1 无作用
4	SQRIC	r1w0	0x01	写 0 清除 ADC 顺序扫描转换完成标志 写 1 无作用
3	REGIC	r1w0	0x01	写 0 清除 ADC 转换结果比较区间标志 写 1 无作用
2	HTIC	r1w0	0x01	写 0 清除 ADC 转换结果比较上阈值 写 1 无作用
1	LTIC	r1w0	0x01	写 0 清除 ADC 转换结果比较下阈值标志 写 1 无作用
0	SGLIC	r1w0	0x01	写 0 清除 ADC 单次转换完成标志 写 1 无作用

28.11.15 ADC 单次转换或顺序扫描转换外部中断触发源配置寄存器 (ADC_ExtTrigger0)

起始地址：0x40002400

地址偏移：0x0B8

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										JQRIC	SQRIC	REGIC	HHT_INTC	LLT_INTC	S_INTC
										R1W0	R1W0	R1W0	R1W0	R1W0	R1W0

Bit	Field	Type	Reset	Description
31	PC15	rw	0x00	PC15 中断触发 ADC 转换
30	PB15	rw	0x00	PB15 中断触发 ADC 转换
29	PA15	rw	0x00	PA15 中断触发 ADC 转换
28	PC11	rw	0x00	PC11 中断触发 ADC 转换
27	PB11	rw	0x00	PB11 中断触发 ADC 转换
26	PA11	rw	0x00	PA11 中断触发 ADC 转换
25	PD07	rw	0x00	PD07 中断触发 ADC 转换
24	PC07	rw	0x00	PC07 中断触发 ADC 转换
23	PB07	rw	0x00	PB07 中断触发 ADC 转换
22	PA07	rw	0x00	PA07 中断触发 ADC 转换
21	PD03	rw	0x00	PD03 中断触发 ADC 转换
20	PC03	rw	0x00	PC03 中断触发 ADC 转换
19	PB03	rw	0x00	PB03 中断触发 ADC 转换
18	PA03	rw	0x00	PA03 中断触发 ADC 转换
17	DMA	rw	0x00	DMA 中断触发 ADC 转换
16	SPI1	rw	0x00	SPI1 中断触发 ADC 转换
15	-	-	-	保留
14	PCA	rw	0x00	PCA 中断触发 ADC 转换
13	RTC	rw	0x00	RTC 中断触发 ADC 转换
12	VC1	rw	0x00	VC1 中断触发 ADC 转换
11	VC0	rw	0x00	VC0 中断触发 ADC 转换
10	LPUART1	rw	0x00	LPUART1 中断触发 ADC 转换
9	LPUART0	rw	0x00	LPUART0 中断触发 ADC 转换
8	UART1	rw	0x00	UART1 中断触发 ADC 转换
7	UART0	rw	0x00	UART0 中断触发 ADC 转换
6	TIM6	rw	0x00	Timer6 中断触发 ADC 转换
5	TIM5	rw	0x00	Timer5 中断触发 ADC 转换
4	TIM4	rw	0x00	Timer4 中断触发 ADC 转换
3	TIM3	rw	0x00	Timer3 中断触发 ADC 转换
2	TIM2	rw	0x00	Timer2 中断触发 ADC 转换
1	TIM1	rw	0x00	Timer1 中断触发 ADC 转换

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
0	TIM0	rw	0x00	Timer0 中断触发 ADC 转换

Notes:

1) TIM4/5/6 中断触发 ADC 自动转换，除了需要使能 TIM4/5/6 的相应中断外，还需要配置 Advanced Timer 的展频及中断触发选择寄存器 TIMX_CR 选择可以触发 ADC 的中断源。

2) 触发 ADC 使用的是各中断标志位的上升沿。如果需要重复触发，需要清除中断标志。如果不需要进入中断服务程序，请不要使能 NVIC 的中断使能。

28.11.16 ADC 插队扫描转换外部中断触发源配置寄存器（ADC_ExtTrigger1）

起始地址：0x40002400

地址偏移：0x0BC

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PC15	PB15	PA15	PC11	PB11	PA11	PD07	PC07	PB07	PA07	PD03	PC03	PB03	PA03	DMA	SPI1
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI0	PCA	RTC	VC1	VC0	LPUAR	LPUAR	UART1	UART0	TIM6	TIM5	TIM4	TIM3	TIM2	TIM1	TIM0
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31	PC15	rw	0x00	PC15 中断触发 ADC 转换
30	PB15	rw	0x00	PB15 中断触发 ADC 转换
29	PA15	rw	0x00	PA15 中断触发 ADC 转换
28	PC11	rw	0x00	PC11 中断触发 ADC 转换
27	PB11	rw	0x00	PB11 中断触发 ADC 转换
26	PA11	rw	0x00	PA11 中断触发 ADC 转换
25	PD07	rw	0x00	PD07 中断触发 ADC 转换
24	PC07	rw	0x00	PC07 中断触发 ADC 转换
23	PB07	rw	0x00	PB07 中断触发 ADC 转换
22	PA07	rw	0x00	PA07 中断触发 ADC 转换
21	PD03	rw	0x00	PD03 中断触发 ADC 转换
20	PC03	rw	0x00	PC03 中断触发 ADC 转换
19	PB03	rw	0x00	PB03 中断触发 ADC 转换
18	PA03	rw	0x00	PA03 中断触发 ADC 转换
17	DMA	rw	0x00	DMA 中断触发 ADC 转换

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
16	SPI1	rw	0x00	SPI1 中断触发 ADC 转换
15	-	-	-	保留
14	PCA	rw	0x00	PCA 中断触发 ADC 转换
13	RTC	rw	0x00	RTC 中断触发 ADC 转换
12	VC1	rw	0x00	VC1 中断触发 ADC 转换
11	VC0	rw	0x00	VC0 中断触发 ADC 转换
10	LPUART1	rw	0x00	LPUART1 中断触发 ADC 转换
9	LPUART0	rw	0x00	LPUART0 中断触发 ADC 转换
8	UART1	rw	0x00	UART1 中断触发 ADC 转换
7	UART0	rw	0x00	UART0 中断触发 ADC 转换
6	TIM6	rw	0x00	Timer6 中断触发 ADC 转换
5	TIM5	rw	0x00	Timer5 中断触发 ADC 转换
4	TIM4	rw	0x00	Timer4 中断触发 ADC 转换
3	TIM3	rw	0x00	Timer3 中断触发 ADC 转换
2	TIM2	rw	0x00	Timer2 中断触发 ADC 转换
1	TIM1	rw	0x00	Timer1 中断触发 ADC 转换
0	TIM0	rw	0x00	Timer0 中断触发 ADC 转换

Notes:

1) TIM4/5/6 中断触发 ADC 自动转换，除了需要使能 TIM4/5/6 的相应中断外，还需要配置 Advanced Timer 的展频及中断触发选择寄存器 TIMX_CR 选择可以触发 ADC 的中断源。

2) 触发 ADC 使用的是各中断标志位的上升沿。如果需要重复触发，需要清除中断标志。如果不需要进入中断服务程序，请不要使能 NVIC 的中断使能。

28.11.17 ADC 单次转换启动控制寄存器 (ADC_SglStart)

起始地址：0x40002400

地址偏移：0x0C0

复位值：0x0000 0000

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

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

Bit	Field	Type	Reset	Description
31:1	Reserved			保留

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
0	Start	rw	0x00	ADC 单次转换启动控制 1：启动 ADC 单次转换 0：停止 ADC 单次转换

28.11.18 ADC 顺序扫描转换启动控制寄存器（ADC_SqrStart）

起始地址：0x40002400

地址偏移：0x0C4

复位值：0x0000 0000

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

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

Bit	Field	Type	Reset	Description
31:1	Reserved			保留
0	Start	rw	0x00	ADC 顺序扫描转换启动控制 1：启动 ADC 顺序扫描转换 0：停止 ADC 顺序扫描转换

28.11.19 ADC 插队扫描转换启动控制寄存器（ADC_JqrStart）

起始地址：0x40002400

地址偏移：0x0C8

复位值：0x0000 0000

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

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

Bit	Field	Type	Reset	Description
31:1	Reserved			保留
0	Start	rw	0x00	ADC 插队扫描转换启动控制 1：启动 ADC 插队扫描转换 0：停止 ADC 插队扫描转换

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

28.11.20 ADC 一直转换启动停止控制寄存器（ADC_allstart）

起始地址：0x40002400

地址偏移：0x0CC

复位值：0x0000 0000

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

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

Bit	Field	Type	Reset	Description
31:1	Reserved			保留
0	Start	rw	0x00	ADC 插队扫描转换启动控制 1：启动 ADC 一直转换 0：停止 ADC 一直转换，需要等待到当次转换结束 读 0 无转换；1 正在转换

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

29 数模转换器（DAC ）

DAC 模块是 12 位电压输出数模转换器。DAC 可以按 8 位或 12 位模式进行配置，并且可与 DMA 控制器配合使用。在 12 位模式下，数据可以采用左对齐或右对齐。可使用输入参考电压 VREF+（与 ADC 共享）。可以有选择地缓冲输出以实现更高的电流驱动。

29.1 DAC 特性

器件集成了 1 个 DAC 转换器，具有一个输出通道：DAC_OUT

DAC 主要特性如下：

- 一个数据保持寄存器；
- 12 位模式下数据采用左对齐或右对齐；
- 同步更新功能；
- 生成噪声波；
- 生成三角波；
- DMA 功能（包括下溢检测）；
- 通过外部触发信号进行转换；
- 4 种参考源：AVCC 电压、ExRef 引脚、内置 1.5v 参考电压、内置 2.5v 参考电压。

29.2 功能说明

29.2.1 框图

如图 215和图 216是 DAC 示意框图。

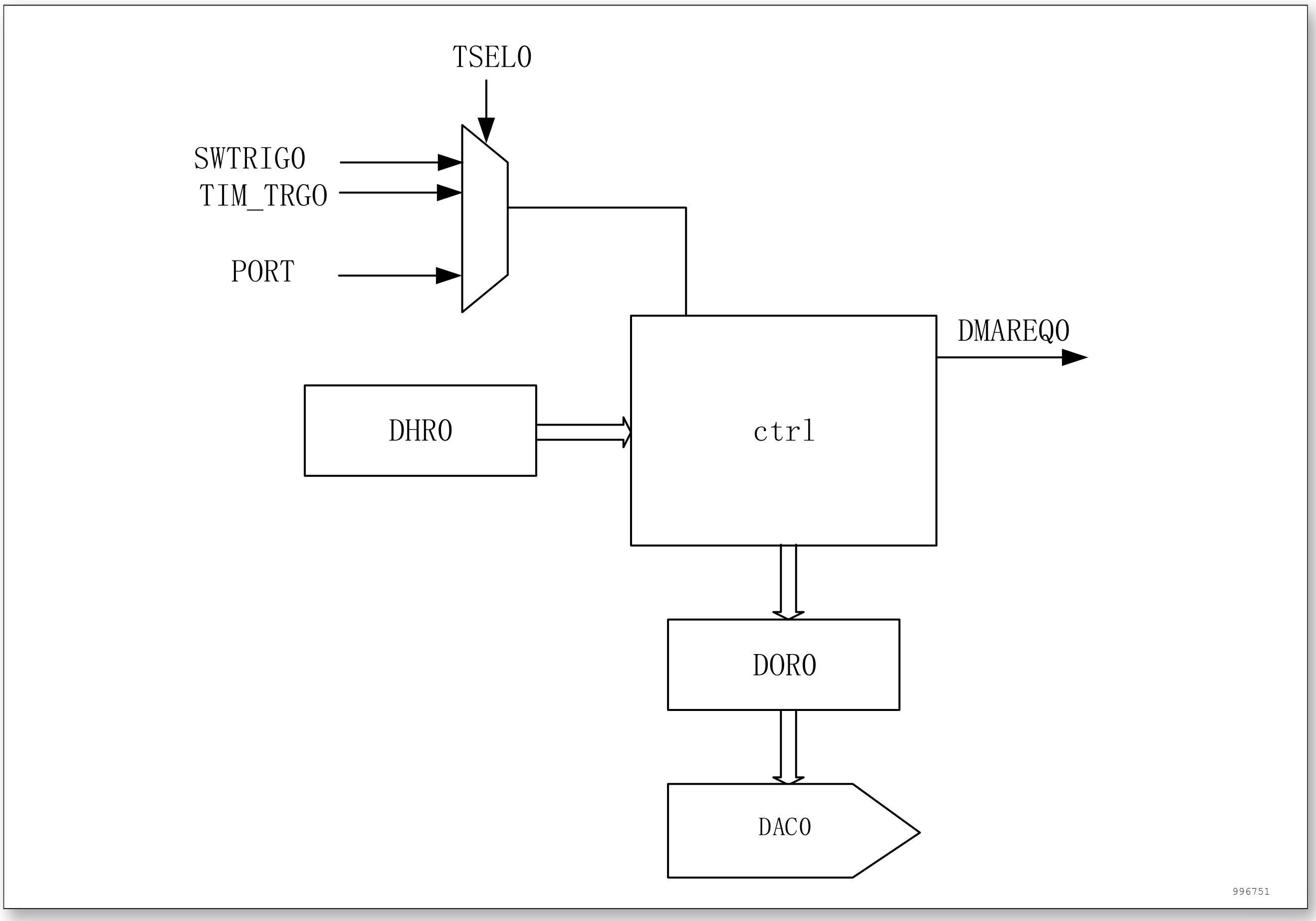

图 215. DAC 框图

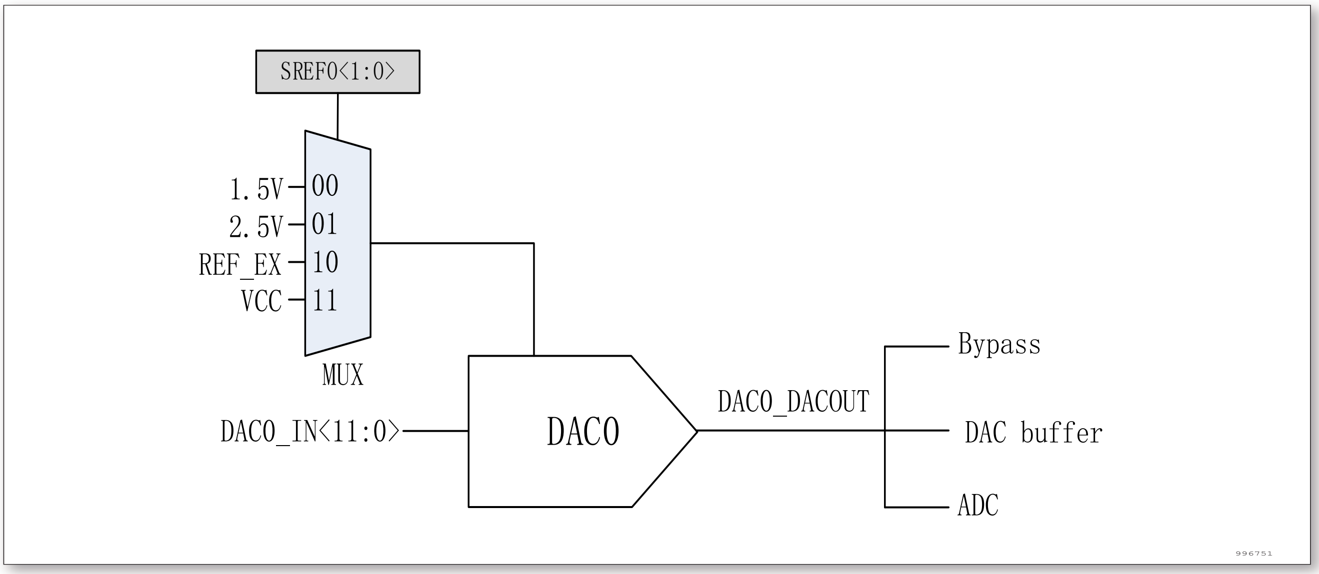

图 216. DAC 框图

29.2.2 DAC 输出缓冲器使能

DAC 可以使用输出缓冲器，可用来降低输出阻抗并在不增加外部运算放大器的情况下直接驱动外部负载。关于输出缓存器的设置参考 < 运算放大器 OPA> 章节。

通过 DAC_CR 寄存器中的 BOFF1 位，可使能或禁止 DAC 通道输出缓冲器。

29.2.3 DAC 通道使能

将 DAC_CR 寄存器中的相应 EN0 位置 1，即可使能对应 DAC 通道。经过一段启动时间 t WAKEUP 后，各 DAC 通道被真正使能。

EN0 位只会使能模拟 DAC 通道。即使 EN0 位复位，DAC 通道数字接口仍处于使能状态。

29.2.4 DAC 输出电压

经过线性转换后，数字输入会转换为 0 到 VREF+ 之间的输出电压。各 DAC 通道引脚的模拟输出电压通过以下公式确定：

$$DAC_{out} = V_{ref} * DOR/4096$$

29.2.5 DAC 触发选择

如果 TEN0 控制位置 1，可通过外部事件（定时计数器、外部中断线）或软件触发转换。TSEL0[2:0] 控制位将决定哪个可能事件将触发转换，如表 119所示。

表 119: DAC 触发选择

TSEL	000	001	010	011	100	101	110	111
触发源	TIM0_TRADC	TIM1_TRADC	TIM2_TRADC	TIM3_TRADC	TIM4_TRADC	TIM5_TRADC	SW TRIG	EXTI

每当 DAC 所选定时器 TRGO 输出或所选外部端口触发上检测到上升沿时，DAC_DHR0 寄存器中存储的最后一个数据即会转移到 DAC_DOR0 寄存器中。发生触发后再经过三个 APB1 周期，DAC_DOR0 寄存器将会得到更新。

如果选择软件触发，一旦 SWTRIG 位置 1，转换即会开始。DAC_DHR0 寄存器内容加载到 DAC_DOR0 寄存器中后，SWTRIG 即由硬件复位。

注意：

- EN0 位置 1 时，无法更改 TSEL0[2:0] 位。如果选择软件触发，DAC_DHR0 寄存器的内容只需一个 APB1 时钟周期即可转移到 DAC_DOR0 寄存器。

29.2.6 功能说明

DAC 数据格式

根据所选配置模式，数据必须按如下方式写入指定寄存器：

存在三种可能：

- 8 位右对齐：软件必须将数据加载到 DAC_DHR8R0 [7:0] 位（存储到 DHR0[11:4] 位）。
- 12 位左对齐：软件必须将数据加载到 DAC_DHR12L0 [15:4] 位（存储到 DHR0[11:0] 位）。
- 12 位右对齐：软件必须将数据加载到 DAC_DHR12R0 [11:0] 位（存储到 DHR0[11:0] 位）。

根据加载的 DAC_DHR0 寄存器，用户写入的数据将移位并存储到相应的 DHR0（数据保持寄存器，即内部非存储器映射寄存器）。之后，DHR0 寄存器将被自动加载，或者通过软件或外部事件触发加载到 DOR0 寄存器。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

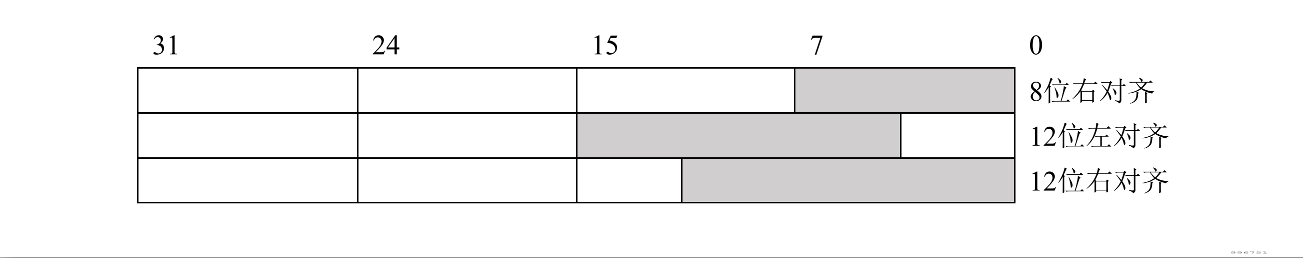

图 217. DAC 数据格式

DAC 转换

DAC_DOR0 无法直接写入，任何数据都必须通过加载 DAC_DHR0 寄存器（写入 DAC_DHR8R0、DAC_DHR12L0、DAC_DHR12R0）才能传输到 DAC 通道。

如果未选择触发（DAC_CR 寄存器中的 TEN0 位复位），那么经过一个 APB1 时钟周期后，DAC_DHR0 寄存器中存储的数据将自动转移到 DAC_DOR0 寄存器。但是，如果选择硬件触发（置位 DAC_CR 寄存器中的 TEN0 位）且触发条件到来，将在三个 PCLK1 时钟周期后进行转移。

当 DAC_DOR0 加载了 DAC_DHR0 内容时，模拟输出电压将在一段时间 tSETTLING 后可用，具体时间取决于电源电压和模拟输出负载。

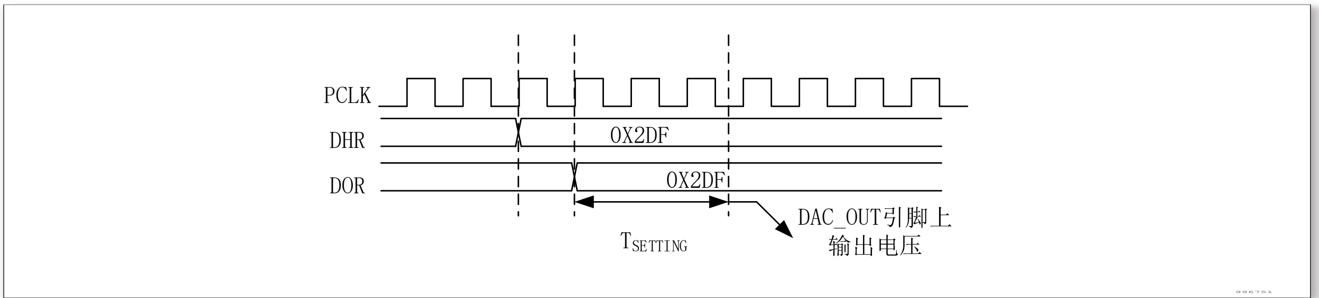

图 218. 触发不使能时 DAC 转换时序图

生成单个 LFSR（线性反馈移位寄存器）的独立触发

要将 DAC 配置为此转换模式（请参见第 33.3.8 节：生成噪声），需要遵循以下顺序：

1. 将 DAC 通道触发使能位 TEN0 置 1。
2. 通过设置 TSEL0[2:0] 位配置触发源。
3. 将 DAC 通道的 WAVE0[1:0] 位配置为“01”，并在 MAMP0[3:0] 位中配置相同的 LFSR 掩码值。
4. 将 DAC 通道数据加载到所需 DAC_DHR0 寄存器（DHR12RD、DHR12LD 或 DHR8RD）。

DAC 通道触发信号到达时，LFSR0 计数器内容（使用相同的掩码）与 DHR0 寄存器内容相加，所得总和转移到 DAC_DOR0 中（三个 APB 时钟周期之后）。LFSR0 计数器随即更新。

生成单个三角波的独立触发

要将 DAC 配置为此转换模式（请参见第 33.2.9 节：生成三角波），需要遵循以下顺序：

1. 将 DAC 通道触发使能位 TEN0 置 1。
2. 通过设置 TSEL0[2:0] 位配置触发源。
3. 将 DAC 通道的 WAVE0[1:0] 位配置为“1x”，并在 MAMP0[3:0] 位中配置相同的最大振幅值。
4. 将 DAC 通道数据加载到所需 DAC_DHR0 寄存器（DHR12RD、DHR12LD 或 DHR8RD）。

DAC 通道触发信号到达时，DAC 通道三角波计数器内容（使用相同的三角波振幅）与 DHR0 寄存器内容相

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

加，所得总和转移到 DAC_DOR0 中（三个 APB 时钟周期之后）。

DAC 通道三角波计数器随即更新。

29.2.7 噪音生成

为了生成可变振幅的伪噪声，可使用 LFSR（线性反馈移位寄存器）。将 WAVE0[1:0] 置为“01”即可选择生成噪声。LFSR 中的预加载值为 0xAAA。在每次发生触发事件后，经过三个 APB 时钟周期，该寄存器会依照特定的计算算法完成更新。

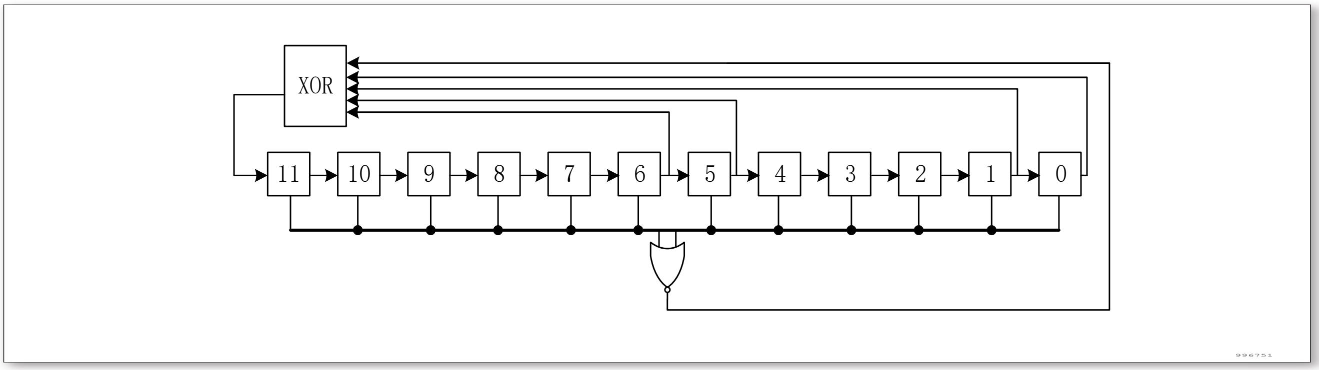

图 219. LFSR（线性反馈移位寄存器）

LFSR 值可以通过 DAC_CR 寄存器中的 MAMP0[3:0] 位来部分或完全屏蔽，在不发生溢出的情况下，该值将与 DAC_DHR0 的内容相加，然后存储到 DAC_DOR0 寄存器中。

如果 LFSR 为 0x0000，将向其注入“1”（防锁定机制）。

可以通过复位 WAVE0[1:0] 位来将 LFSR 波形产生功能关闭。

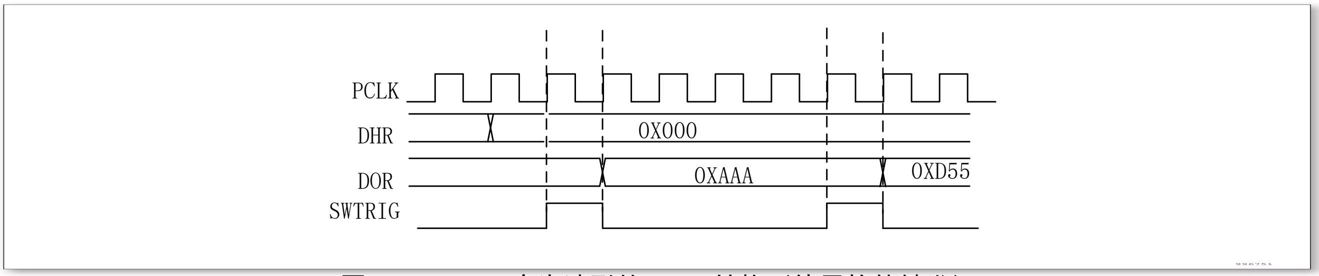

图 220. LFSR 产生波形的 DAC 转换（使用软件触发）

29.2.8 三角波生成

可以在直流电流或慢变信号上叠加一个小幅三角波。将 WAVE0[1:0] 置为“10”即可选择 DAC 生成三角波。振幅通过 DAC_CR 寄存器中的 MAMP0[3:0] 位进行配置。每次发生触发事件后，经过三个 APB 时钟周期，内部三角波计数器将会递增。在不发生溢出的情况下，该计数器的值将与 DAC_DHR0 寄存器内容相加，所得总和将存储到 DAC_DOR0 寄存器中。只要小于 MAMP0[3:0] 位定义的最大振幅，三角波计数器就会一直递增。一旦达到配置的振幅，计数器将递减至零，然后再递增，以此类推。可以通过复位 WAVE0[1:0] 位来将三角波产生功能关闭。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

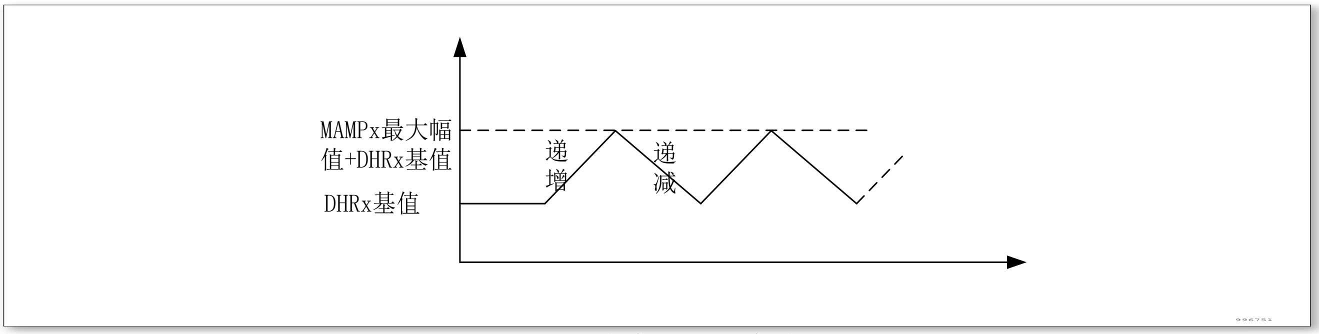

图 221. 生成 DAC 三角波

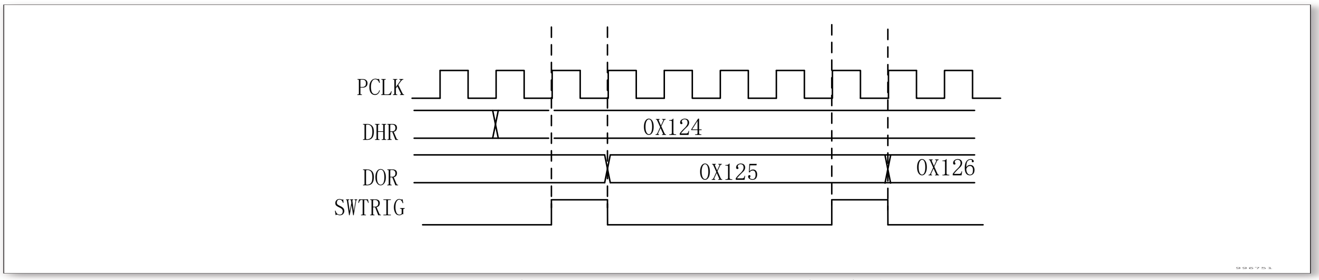

图 222. 生成三角波波形的 DAC 转换（使能软件触发）

29.2.9 DMA 请求

DAC 通道都具有 DMA 功能。DMA 通道用于处理 DAC 通道的 DMA 请求。

当 DMAEN0 位置 1 时，如果发生触发，则将产生 DAC DMA 请求。DAC_DHR0 寄存器的值随后转移到 DAC_DOR0 寄存器。

在双通道模式下，如果两个 DMAEN0 位均置 1，则将产生两个 DMA 请求。如果只需要一个 DMA 请求，用户应仅将相应 DMAEN0 位置 1。这样，应用程序可以在双通道模式下通过一个 DMA 请求和一个特定 DMA 通道来管理两个 DAC 通道。

DMA 下溢

DAC DMA 请求没有缓冲队列。这样，如果第二个外部触发到达时尚未收到第一个外部触发的应答，将不会发出新的请求，并且 DAC_SR 寄存器中的 DAM 通道下溢标志 DMAUDR0 将置 1，以报告这一错误状况。DMA 数据传输随即禁止，并且不再处理其他 DMA 请求。DAC 通道仍将继续转换旧有数据。

软件应通过写入“1”来将 DMAUDR0 标志清零，将所用 DMA 数据流的 DMAEN 位清零，并重新初始化 DMA 和 DAC 通道，以便正确地重新开始 DMA 传输。软件应修改 DAC 触发转换频率或减轻 DMA 工作负载，以避免再次发生 DMA 下溢。最后，可通过使能 DMA 数据传输和转换触发来继续完成 DAC 转换。

对于各 DAC 通道，如果使能 DAC_CR 寄存器中相应的 DMAUDRIE0 位，还将产生中断。

29.3 DAC 寄存器

PERI_RESET1.DAC 与 PERI_RESET1.DAC 控制不可读，读出的值恒为 0 ，使用读修改写更改这两个寄掉存器的其他控制位时会修改掉 DAC 控制位的值成 0 ， 需注意！

表 120. DAC 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x000	DAC_CR	控制寄存器	0x0000 C000	小节 29.3.1
0x004	DAC_SWTR	软件触发寄存器	0x0000 0000	小节 29.3.2
0x008	DAC_DHR12R0	DAC0 12 位右对齐数据寄存器	0x0000 0000	小节 29.3.3
0x00C	DAC_DHR12L0	DAC0 12 位左对齐数据寄存器	0x0000 0000	小节 29.3.4
0X010	DAC_DHR8R0	DAC0 8 位右对齐数据寄存器	0x0000 0000	小节 29.3.5
0x02C	DAC_DOR0	DAC0 数据输出寄存器	0x0000 0000	小节 29.3.6
0X034	DAC_SR	状态寄存器	0x0000 4000	小节 29.3.7
0X038	DAC_ETRS	DAC 测试寄存器/外部触发选择寄存器	0x0000 0000	小节 29.3.8

29.3.1 控制寄存器（DAC_CR）

起始地址：0x40002500

地址偏移：0x000

复位值：0x0000 C000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SREF0		DMAUDRIE0	DMAEN0	MAMP0				WAVE0		TSEL0			TEN0	BOFF0	EN0
RW		RW	RW	RW				RW		RW			RW	RW	RW

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:14	SREF0	rw	0x03	DAC0 参考电压选择 00：内部 1.5V 01：内部 2.5V 10：外部参考电压 ExRef（PB01） 11：AVCC 电压
13	DMAUDRIE0	rw	0x00	DAC0 DMA 下溢中断使能 0: 禁止 DAC0 通道 DMA 下溢中断 1: 使能 DAC0 通道 DMA 下溢中断
12	DMAEN0	rw	0x00	DAC0 通道 DMA 使能 0: 禁止 DAC0 通道 DMA 模式 1: 使能 DAC0 通道 DMA 模式

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
11:8	MAMP0	rw	0x00	DAC0 通道掩码/振幅选择器 0000：不屏蔽 LFSR 的位 0/三角波振幅等于 1 0001：不屏蔽 LFSR 的位 [1:0]/三角波振幅等于 3 0010：不屏蔽 LFSR 的位 [2:0]/三角波振幅等于 7 0011：不屏蔽 LFSR 的位 [3:0]/三角波振幅等于 15 0100：不屏蔽 LFSR 的位 [4:0]/三角波振幅等于 31 0101：不屏蔽 LFSR 的位 [5:0]/三角波振幅等于 63 0110：不屏蔽 LFSR 的位 [6:0]/三角波振幅等于 127 0111：不屏蔽 LFSR 的位 [7:0]/三角波振幅等于 255 1000：不屏蔽 LFSR 的位 [8:0]/三角波振幅等于 511 1001：不屏蔽 LFSR 的位 [9:0]/三角波振幅等于 1023 1010：不屏蔽 LFSR 的位 [10:0]/三角波振幅等于 2047 ≥1011：不屏蔽 LFSR 的位 [11:0]/三角波振幅等于 4095
7:6	WAVE0	rw	0x00	DAC0 通道噪声/三角波生成使能 00: 禁止生成波 01: 使能生成噪声波 1x 使能生成三角波
5:3	TSEL0	rw	0x00	DAC0 通道触发器选择 000 TIM0 TRADC 触发 001 TIM1 TRADC 触发 010 TIM2 TRADC 触发 011 TIM3 TRADC 触发 100 TIM4 TRADC 触发 101 TIM5 TRADC 触发 110 软件触发 111 外部端口触发
2	TEN0	rw	0x00	DAC0 通道触发使能 0：禁止 DAC0 通道触发，写入 DAC_DHR0 寄存器的数据在一个 APB1 时钟周期之后转移到 DAC_DOR0 寄存器 1：使能 DAC0 通道触发，DAC_DHR0 寄存器的数据在三个 APB1 时钟周期之后转移到 DAC_DOR0 寄存器
1	BOFF0	rw	0x00	DAC0 通道输出缓冲器禁止 0：使能 DAC0 通道输出缓冲器 1：禁止 DAC0 通道输出缓冲器
0	EN0	rw	0x00	DAC0 通道使能 0：禁止 DAC0 通道 1：使能 DAC0 通道

29.3.2 软件触发寄存器（DAC_SWTR）

起始地址：0x40002500

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

地址偏移：0x004

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved															SWTRIG0
															W1

Bit	Field	Type	Reset	Description
31:1	Reserved			保留
0	SWTRIG0	w1	0x00	DAC0 软件触发 写 1 软件触发，自动清零，写 0 无动作

29.3.3 DAC0 12 位右对齐数据寄存器 (DAC_DHR12R0)

起始地址：0x40002500

地址偏移：0x008

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				DHR0											
				RW											

Bit	Field	Type	Reset	Description
31:12	Reserved			保留
11:0	DHR0	rw	0x00	DAC 通道 0 12 位右对齐数据 这些位由软件写入，用于为 DAC 通道 0 指定 12 位数据。

29.3.4 DAC0 12 位左对齐数据寄存器 (DAC_DHR12L0)

起始地址：0x40002500

地址偏移：0x00C

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DHR0												Reserved			
RW															

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15:4	DHR0	rw	0x00	DAC 通道 0 12 位左对齐数据 这些位由软件写入，用于为 DAC 通道 0 指定 12 位数据。
3:0	Reserved			保留

29.3.5 DAC0 8 位右对齐数据寄存器 (DAC_DHR8R0)

起始地址：0x40002500

地址偏移：0X010

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								DHR0							
								RW							

Bit	Field	Type	Reset	Description
31:8	Reserved			保留
7:0	DHR0	rw	0x00	DAC 通道 0 8 位右对齐数据 这些位由软件写入，用于为 DAC 通道 0 指定 8 位数据。 写入的为 DHR 的高八位。

29.3.6 DAC0 数据输出寄存器 (DAC_DOR0)

起始地址：0x40002500

地址偏移：0x02C

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				DOR0											
				RO											

Bit	Field	Type	Reset	Description
31:12	Reserved			保留
11:0	DOR0	ro	0x00	DAC 0 通道数据输出 DAC 0 通道的数据输出

29.3.7 状态寄存器(DAC_SR)

起始地址：0x40002500

地址偏移：0X034

复位值：0x0000 4000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		DMAUDR0	Reserved												
		RW													

Bit	Field	Type	Reset	Description
31:14	Reserved			保留
13	DMAUDR0	rw	0x00	DAC 0 通道 DMA 下溢标志 0：DAC0 通道未发生 DMA 下溢错误状况 1：DAC 0 通道发生 DMA 下溢错误状况（当前所选触发源以高于 DMA 服务能力的频率驱动 DAC 0 通道转换）
12:0	Reserved			保留

29.3.8 DAC 测试寄存器/外部触发选择寄存器(DAC_ETRS)

起始地址：0x40002500

地址偏移：0X038

复位值：0x0000 0000

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

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

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

Bit	Field	Type	Reset	Description
31:7	Reserved			保留
6:4	PTRIGSEL	rw	0x00	端口触发选择 000 PA9 001 PB9 010 PC9 011 PD9 100 PE9 101 PF9 11X 保留
3:0	Reserved			保留

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

30 模拟比较器 (VC)

30.1 模拟电压比较器 VC 简介

模拟电压比较器 VC 用于比较两个输入模拟电压的大小，并根据比较结果输出高/低电平。当“＋”输入端电压高于“－”输入端电压时，电压比较器输出为高电平；当“＋”输入端电压低于“－”输入端电压时，电压比较器输出为低电平。本产品内部集成的模拟电压比较器 VC 具有以下特性：

- 支持电压比较功能；
- 支持内部 64 阶 VCC 分压（使用分压来源电压需要大于 1.8V）；
- 支持 16 个外部输入端口和片内 BGR 输出的参考电压作为电压比较器的输入；
- 支持三种软件可配置的中断触发方式：高电平触发/上升沿触发/下降沿触发；
- 电压比较器的输出可以作为通用定时器和低功耗定时器控制输入；
- 电压比较器的输出可以作为高级定时器和通用定时器的刹车输入或者捕获输入；
- 支持在超低功耗模式下工作，电压比较器的中断输出可以将芯片从超低功耗模式下唤醒；
- 提供软件可配置的滤波时间以增强芯片的抗干扰能力。

注意：使用 VC 需要使能 BGR, 参考 BGR 寄存器描述

30.2 电压比较器框架图

图 223是电压比较器框架图。

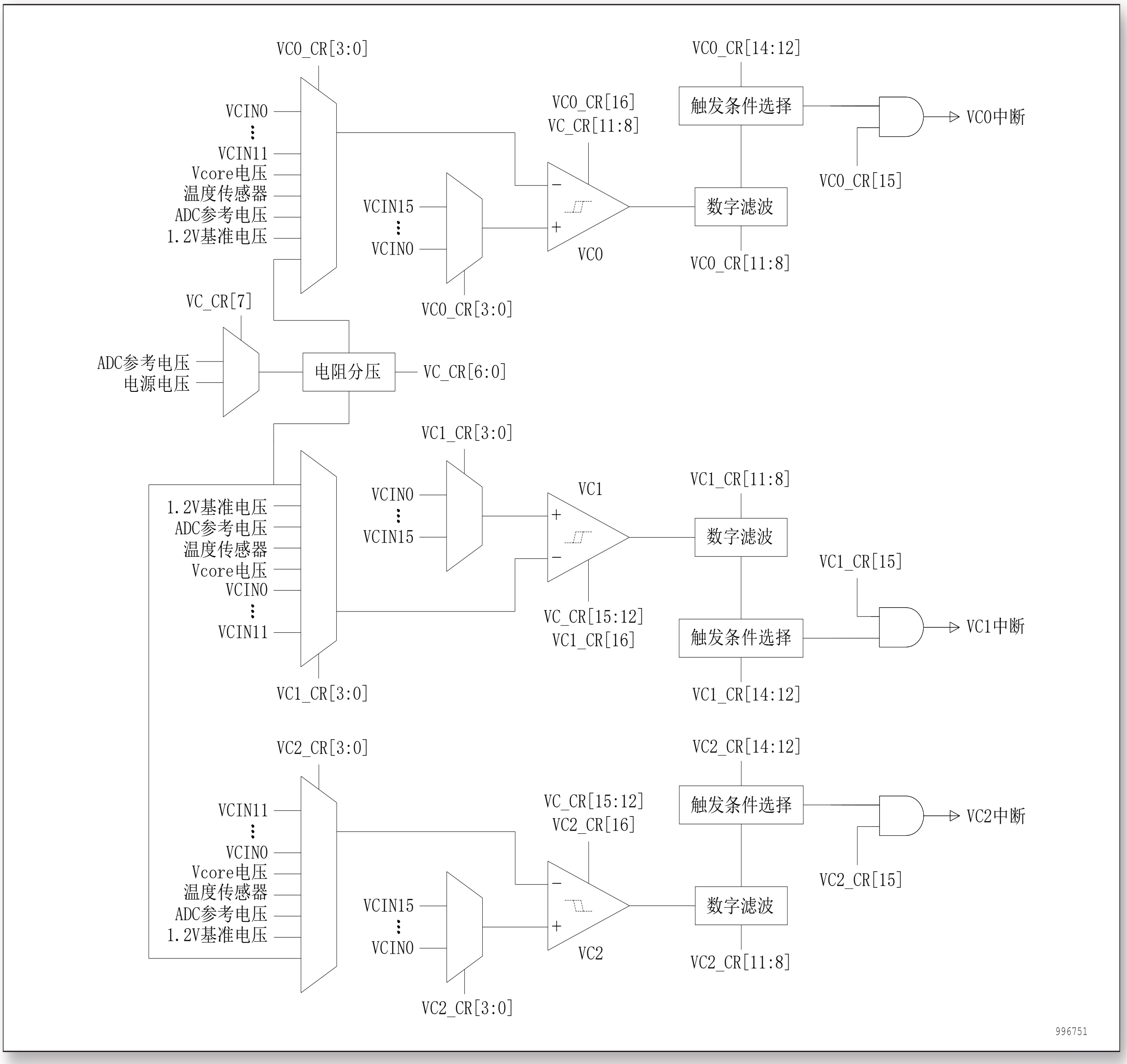

图 223. VC 框架图

30.3 建立/响应时间

当使用电压比较器时，从 VC 使能或者 VC 的两端输入电压变化到输出正确的结果的时间由 VC 控制寄存器（VC_CR）中的 BIAS_SEL 控制位决定，电流越大 VC 响应越快，典型值从 200nS 到 20uS 四档可调。

如果需要内部 BGR 提供的参考电压（1.5V 或 2.5V）作为比较基准电压。内部 BGR 的启动时间大约为 20us，电压比较器需要等待内部 BGR 稳定后才能正常输出。

30.4 滤波时间

在电压比较器固有的建立/响应时间之外，用户可以设置更长的滤波时间来过滤掉系统噪声，比如马达停止时的大电流噪声。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

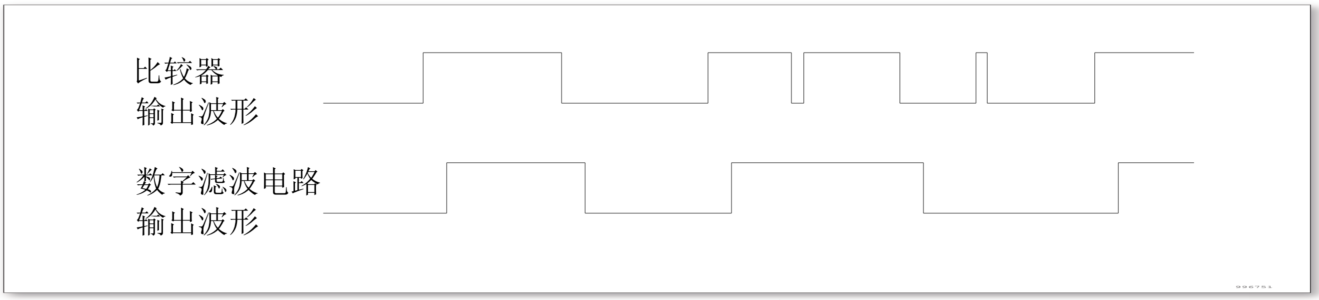

图 224. VC 滤波响应时间

30.5 迟滞功能

电压比较器可以选择迟滞功能，迟滞功能使能后的图示如图 225：

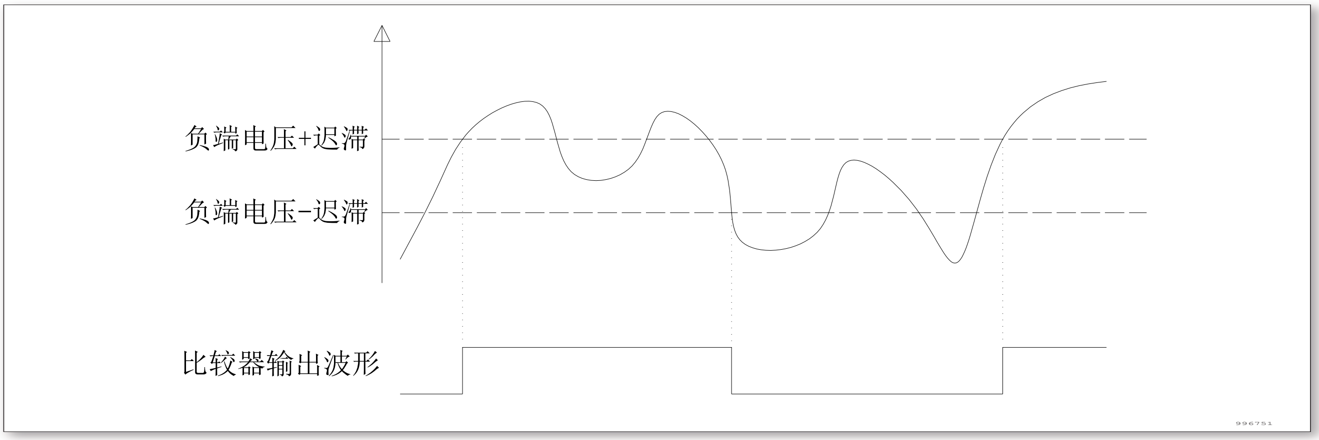

图 225. VC 迟滞功能

30.6 VC 寄存器

表 121. VC 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x010	VC_CR	VC0/1 配置寄存器 0	0x0000 0020	小节 30.6.1
0x014	VC0_CR	VC0 配置寄存器	0x0000 0000	小节 30.6.2
0x018	VC1_CR	VC1 配置寄存器	0x0000 0000	小节 30.6.3
0x01C	VC0_OUT_CFG	VC0 输出配置寄存器	0x0000 0000	小节 30.6.4
0x020	VC1_OUT_CFG	VC1 输出配置寄存器	0x0000 0000	小节 30.6.5
0x024	VC_INT	VC 中断寄存器	0x0000 0000	小节 30.6.6
0x150	VC2_CR	VC2 配置寄存器	0x0000 0000	小节 30.6.7
0x154	VC2_OUT_CFG	VC2 输出配置寄存器	0x0000 0000	小节 30.6.8

30.6.1 VC0/1 配置寄存器 0 (VC_CR)

起始地址：0x40002400

地址偏移：0x010

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved												VC2_HYS_SEL		VC2_BIAS_SEL	
												R/W		R/W	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VC1_HYS_SEL		VC1_BIAS_SEL		VC0_HYS_SEL		VC0_BIAS_SEL		VC_REF2P5_SEL	VC_DIV_EN	VC_DIV					
R/W		R/W		R/W		R/W		R/W	R/W	R/W					

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
19:18	VC2_HYS_SEL	rw	0x00	VC2 迟滞选择: 00: 没有迟滞 01: 迟滞电压大约 10mV 10: 迟滞电压大约 20mV 11: 迟滞电压大约 30mV
17:16	VC2_BIAS_SEL	rw	0x00	VC2 功耗选择（功耗越大，响应速度越快） 00:300nA 01:1.2uA 10:10uA(需要开启 BGR，BGR 启动时间大约 20us) 11:20uA(需要开启 BGR，BGR 启动时间大约 20us)
15:14	VC1_HYS_SEL	rw	0x00	VC1 迟滞选择: 00: 没有迟滞 01: 迟滞电压大约 10mV 10: 迟滞电压大约 20mV 11: 迟滞电压大约 30mV
13:12	VC1_BIAS_SEL	rw	0x00	VC1 功耗选择（功耗越大，响应速度越快） 00:300nA 01:1.2uA 10:10uA(需要开启 BGR，BGR 启动时间大约 20us) 11:20uA(需要开启 BGR，BGR 启动时间大约 20us)
11:10	VC0_HYS_SEL	rw	0x00	VC0 迟滞选择: 00: 没有迟滞 01: 迟滞电压大约 10mV 10: 迟滞电压大约 20mV 11: 迟滞电压大约 30mV

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
9:8	VC0_BIAS_SEL	rw	0x00	VC0 功耗选择（功耗越大，响应速度越快） 00:300nA 01:1.2uA 10:10uA(需要开启 BGR，BGR 启动时间大约 20us) 11:20uA(需要开启 BGR，BGR 启动时间大约 20us)
7	VC_REF2P5_SEL	rw	0x00	VC_DIV 参考电压 Vref 选择 0:VCC 1:ADC_CR0.SREF 所选择的参考电压
6	VC_DIV_EN	rw	0x00	6 位 DAC 使能 1：使能
5:0	VC_DIV	rw	0x00	6 位 DAC 配置 000000: 1/64 Vref 000001:2/64 Vref 000010:3/64 Vref 000011:4/64 Vref … 111110:63/64 Vref 111111: Vref

30.6.2 VC0 配置寄存器 (VC0_CR)

起始地址：0x40002400

地址偏移：0x014

复位值：0x0000 0000

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="15" rowspan="2">Reserved</td><td>EN</td></tr>
<tr><td>RW</td></tr>
</table>

<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>IE</td><td>level</td><td>rising</td><td>falling</td><td colspan="3">debounce_time</td><td>FLTEN</td><td colspan="4">n_sel</td><td colspan="4">p_sel</td></tr>
<tr><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td colspan="3">RW</td><td>RW</td><td colspan="4">RW</td><td colspan="4">RW</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:17	Reserved			保留
16	EN	rw	0x00	电压比较器使能 1：使能电压比较器 0：关闭电压比较器
15	IE	rw	0x00	VC 中断使能 1：使能；0：禁止

Bit	Field	Type	Reset	Description
14	level	rw	0x00	VC 输出信号触发中断选择 1：使能高电平触发 INT flag 0：禁止高电平触发 INT flag
13	rising	rw	0x00	VC 输出信号触发中断选择 1：使能上升沿触发 INT flag 0：禁止上升沿触发 INT flag
12	falling	rw	0x00	VC 输出信号触发中断选择 1：使能下降沿触发 INT flag 0：禁止下降沿触发 INT flag
11:9	debounce_time	rw	0x00	VC 输出滤波时间配置 111：滤波时间大约为 28.8ms 110：滤波时间大约为 7.2ms 101：滤波时间大约为 1.8ms。 100：滤波时间大约为 450us 011：滤波时间大约为 112us 010：滤波时间大约为 28us 001：滤波时间大约为 14us 000：滤波时间大约为 7us 注意：滤波时间的配置只有在 FLTEN=1 时才有效。
8	FLTEN	rw	0x00	1：启动 VC 滤波 0：VC 无滤波
7:4	N_SEL	rw	0x00	电压比较器“-”端输入选择 0000: select channel 0 input PA0 0001: select channel 1 input PA1 0010: select channel 2 input PA2 0011: select channel 3 input PA3 0100: select channel 4 input PA4 0101: select channel 5 input PA5 0110: select channel 6 input PA6 0111: select channel 7 input PA7 1000: select channel 8 input PC4 1001: select channel 9 input PC5 1010:DAC0 1011: 电阻分压输出电压 1100: 内置温度传感器输出电压 1101: 内部基准 1.2V 输出电压 1110:ADC 的 REF (使用需要使能 ADC) 1111:LDO 的输出电压

Bit	Field	Type	Reset	Description
3:0	P_SEL	rw	0x00	电压比较器“+”端输入选择 0000: select channel 0 input PC0 0001: select channel 1 input PC1 0010: select channel 2 input PC2 0011: select channel 3 input PC3 0100: select channel 4 input PA0 0101: select channel 5 input PA1 0110: select channel 6 input PA2 0111: select channel 7 input PA3 1000: select channel 8 input PA4 1001: select channel 9 input PA5 1010: select channel 10 input PA6 1011: select channel 11 input PA7 1100: select channel 12 input PB4 1101: select channel 13 input PB5 1110: select channel 14 input PB6 1111: DAC0

30.6.3 VC1 配置寄存器（VC1_CR）

起始地址：0x40002400

地址偏移：0x018

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IE	level	rising	falling	debounce_time			FLTEN	n_sel				p_sel			
RW	RW	RW	RW	RW			RW	RW				RW			

Bit	Field	Type	Reset	Description
31:17	Reserved			保留
16	EN	rw	0x00	电压比较器使能 1：使能电压比较器 0：关闭电压比较器
15	IE	rw	0x00	VC 中断使能 1：使能；0：禁止

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
14	level	rw	0x00	VC 输出信号触发中断选择 1：使能高电平触发 INT flag 0：禁止高电平触发 INT flag
13	rising	rw	0x00	VC 输出信号触发中断选择 1：使能上升沿触发 INT flag 0：禁止上升沿触发 INT flag
12	falling	rw	0x00	VC 输出信号触发中断选择 1：使能下降沿触发 INT flag 0：禁止下降沿触发 INT flag
11:9	debounce_time	rw	0x00	VC 输出滤波时间配置 111：滤波时间大约为 28.8ms 110：滤波时间大约为 7.2ms 101：滤波时间大约为 1.8ms。 100：滤波时间大约为 450us 011：滤波时间大约为 112us 010：滤波时间大约为 28us 001：滤波时间大约为 14us 000：滤波时间大约为 7us 注意：滤波时间的配置只有在 FLTEN=1 时才有效。
8	FLTEN	rw	0x00	1：启动 VC 滤波 0：VC 无滤波
7:4	N_SEL	rw	0x00	电压比较器“-”端输入选择 0000: select channel 0 input PC0 0001: select channel 1 input PC1 0010: select channel 2 input PC2 0011: select channel 3 input PC3 0100: select channel 4 input PA0 0101: select channel 5 input PA1 0110: select channel 6 input PB0 0111: select channel 7 input PB1 1000: select channel 8 input PB2 1001: select channel 9 input PB3 1010: DAC1 1011: 电阻分压输出电压 1100: 内置温度传感器输出电压 1101: 内部基准 1.2V 输出电压 1110:ADC 的 REF (使用需要使能 ADC) 1111:LDO 的输出电压

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
3:0	P_SEL	rw	0x00	电压比较器“+”端输入选择 0000: select channel 0 input PA0 0001: select channel 1 input PA1 0010: select channel 2 input PA2 0011: select channel 3 input PA3 0100: select channel 4 input PA4 0101: select channel 5 input PA5 0110: select channel 6 input PB1 0111: select channel 7 input PB2 1000: select channel 8 input PB10 1001: select channel 9 input PB12 1010: select channel 10 input PB13 1011: select channel 11 input PB14 1100: select channel 12 input PB4 1101: DAC0 1110: select channel 14 input PB6 1111: select channel 15 input PB7

30.6.4 VC0 输出配置寄存器 (VC0_OUT_CFG)

起始地址：0x40002400

地址偏移：0x01C

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
brake	TIM6	INV_TIM6	TIM5	INV_TIM5	TIM4	INV_TIM4	Reserved			TIMBK	TIM3RCLR	TIM2RCLR	TIM1RCLR	TIM0RCLR	INV_Timer
RW	RW	RW	RW	RW	RW	RW				RW	RW	RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15	brake	rw	0x00	VC0 作为 Advanced Timer 刹车控制 1：使能；0：禁止。
14	TIM6	rw	0x00	VC0 filter 结果输出到 TIM6 捕获输入 CHA 使能 1：使能；0：禁止。
13	INV_TIM6	rw	0x00	VC0 filter 结果输出到 TIM6 反向使能 1：使能反向；0：禁止反向，输入与 VC 输出同向。

Bit	Field	Type	Reset	Description
12	TIM5	rw	0x00	VC0 filter 结果输出到 TIM5 捕获输入 CHA 使能 1：使能；0：禁止。
11	INV_TIM5	rw	0x00	VC0 filter 结果输出到 TIM5 反向使能 1：使能反向；0：禁止反向，输入与 VC 输出同向。
10	TIM4	rw	0x00	VC0 filter 结果输出到 TIM4 捕获输入 CHA 使能 1：使能；0：禁止。
9	INV_TIM4	rw	0x00	VC0 filter 结果输出到 TIM4 反向使能 1：使能反向；0：禁止反向，输入与 VC 输出同向。
8:6	Res.			保留
5	TIMBK	rw	0x00	VC0 filter 结果输出到 Timer0/1/2/3 刹车控制 1：使能；0：禁止。
4	TIM3RCLR	rw	0x00	VC0 filter 结果输出到 TIM3 REFCLR 使能控制 1：使能；0：禁止。
3	TIM2RCLR	rw	0x00	VC0 filter 结果输出到 TIM2 REFCLR 使能控制 1：使能；0：禁止。
2	TIM1RCLR	rw	0x00	VC0 filter 结果输出到 TIM1 REFCLR 使能控制 1：使能；0：禁止。
1	TIM0RCLR	rw	0x00	VC0 filter 结果输出到 TIM0 REFCLR 使能控制 1：使能；0：禁止。
0	INV_Timer	rw	0x00	VC0 filter 结果输出反向到各 TIM0/1/2/3/REFCLR 1：使能反向；0：禁止反向，输入与 VC 输出同向。

30.6.5 VC1 输出配置寄存器（VC1_OUT_CFG）

起始地址：0x40002400

地址偏移：0x020

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
brake	TIM 6	INV_ TIM6	TIM 5	INV_ TIM5	TIM 4	INV_ TIM4	Reserved			TIMB K	TIM3 RCLR	TIM2 RCLR	TIM1 RCLR	TIM0 RCLR	INV_ Timer
RW	RW	RW	RW	RW	RW	RW				RW	RW	RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15	brake	rw	0x00	VC1 作为 Advanced Timer 刹车控制 1：使能；0：禁止。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
14	TIM6	rw	0x00	VC1 filter 结果输出到 TIM6 捕获输入 CHB 使能 1：使能；0：禁止。
13	INV_TIM6	rw	0x00	VC1 filter 结果输出到 TIM6 反向使能 1：使能反向；0：禁止反向，输入与 VC 输出同向。
12	TIM5	rw	0x00	VC1 filter 结果输出到 TIM5 捕获输入 CHB 使能 1：使能；0：禁止。
11	INV_TIM5	rw	0x00	VC1 filter 结果输出到 TIM5 反向使能 1：使能反向；0：禁止反向，输入与 VC 输出同向。
10	TIM4	rw	0x00	VC1 filter 结果输出到 TIM4 捕获输入 CHB 使能 1：使能；0：禁止。
9	INV_TIM4	rw	0x00	VC1 filter 结果输出到 TIM4 反向使能 1：使能反向；0：禁止反向，输入与 VC 输出同向。
8:6	Res.	rw	0x00	保留
5	TIMBK	rw	0x00	VC1 filter 结果输出到 Timer0/1/2/3 刹车控制 1：使能；0：禁止。
4	TIM3RCLR	rw	0x00	VC1 filter 结果输出到 TIM3 REFCLR 使能控制 1：使能；0：禁止。
3	TIM2RCLR	rw	0x00	VC1 filter 结果输出到 TIM2 REFCLR 使能控制 1：使能；0：禁止。
2	TIM1RCLR	rw	0x00	VC1 filter 结果输出到 TIM1 REFCLR 使能控制 1：使能；0：禁止。
1	TIM0RCLR	rw	0x00	VC1 filter 结果输出到 TIM0 REFCLR 使能控制 1：使能；0：禁止。
0	INV_Timer	rw	0x00	VC1 filter 结果输出反向到各 TIM0/1/2/3/REFCLR 1：使能反向；0：禁止反向，输入与 VC 输出同向。

30.6.6 VC 中断寄存器（VC_INT）

起始地址：0x40002400

地址偏移：0x024

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										VC2_ Filter	VC2_ INTF	VC1_ Filter	VC0_ Filter	VC1_ INTF	VC0_ INTF
										RO	RW0	RO	RO	RW0	RW0

Bit	Field	Type	Reset	Description
31:4	Reserved			保留
5	VC2_Filter	ro	0x00	VC2 Filter 后的状态
4	VC2_INTF	rw0	0x00	VC2 中断标志，1 发生 VC2 中断；0 未发生中断；写 0 清除中断标志，写 1 无效
3	VC1_Filter	ro	0x00	VC1 Filter 后的状态
2	VC0_Filter	ro	0x00	VC0 Filter 后的状态
1	VC1_INTF	rw0	0x00	VC1 中断标志，1 发生 VC1 中断；0 未发生中断；写 0 清除中断标志，写 1 无效
0	VC0_INTF	rw0	0x00	VC0 中断标志，1 发生 VC0 中断；0 未发生中断；写 0 清除中断标志，写 1 无效

30.6.7 VC2 配置寄存器 (VC2_CR)

起始地址：0x40002400

地址偏移：0x150

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															EN
															R/W

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IE	level	rising	fallin	debounce_time			FLTEN	n_sel				p_sel			
R/W	R/W	R/W	R/W	R/W			R/W	R/W				R/W			

Bit	Field	Type	Reset	Description
31:17	Reserved			保留
16	EN	rw	0x00	电压比较器使能 1：使能电压比较器 0：关闭电压比较器
15	IE	rw	0x00	VC 中断使能 1：使能；0：禁止
14	level	rw	0x00	VC 输出信号触发中断选择 1：使能高电平触发 INT flag 0：禁止高电平触发 INT flag
13	rising	rw	0x00	VC 输出信号触发中断选择 1：使能上升沿触发 INT flag 0：禁止上升沿触发 INT flag
12	falling	rw	0x00	VC 输出信号触发中断选择 1：使能下降沿触发 INT flag 0：禁止下降沿触发 INT flag

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
11:9	debounce _time	rw	0x00	VC 输出滤波时间配置 111：滤波时间大约为 28.8ms 110：滤波时间大约为 7.2ms 101：滤波时间大约为 1.8ms 100：滤波时间大约为 450us 011：滤波时间大约为 112us 010：滤波时间大约为 28us 001：滤波时间大约为 14us 000：滤波时间大约为 7us 注意：滤波时间的配置只有在 FLTEN=1 时才有效。
8	FLTEN	rw	0x00	1：启动 VC 滤波 0：VC 无滤波
7:4	N_SEL	rw	0x00	电压比较器“-”端输入选择 0000: select channel 0 input PA5 0001: select channel 1 input PB1 0010: select channel 2 input PE11 0011: select channel 3 input PE15 0100: select channel 4 input PB11 0101: select channel 5 input PB14 0110: select channel 6 input PD10 0111: select channel 7 input PD11 1000: select channel 8 input PC7 1001: DAC0 1010: DAC0 1011:NC 1100: 内置温度传感器输出电压 1101: 内部基准 1.2V 输出电压 1110:ADC 的 REF (使用需要使能 ADC) 1111:LDO 的输出电压

Bit	Field	Type	Reset	Description
3:0	P_SEL	rw	0x00	电压比较器“+”端输入选择 0000: select channel 0 input PA5 0001: select channel 1 input PB1 0010: select channel 2 input PE9 0011: select channel 3 input PE10 0100: select channel 4 input PE11 0101: select channel 5 input PE13 0110: select channel 6 input PE14 0111: select channel 7 input PE15 1000: select channel 8 input PB11 1001: select channel 9 input PB14 1010: select channel 10 input PD9 1011: select channel 11 input PD10 1100: select channel 12 input PD11 1101: select channel 13 input PC7 1110: DAC0 1111: DAC0

30.6.8 VC2 输出配置寄存器 (VC2_OUT_CFG)

起始地址：0x40002400

地址偏移：0x154

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
brake	TIM6	INV_Timer6	TIM5	INV_Timer5	TIM4	INV_Timer4	Reserved			TIMBK	TIM3RCLR	TIM2RCLR	TIM1RCLR	TIM0RCLR	INV_Timer
R/W	R/W	R/W	R/W	R/W	R/W	R/W				R/W	R/W	R/W	R/W	R/W	R/W

Bit	Field	Type	Reset	Description
31:16	Reserved			保留
15	brake	rw	0x00	VC2 作为 Advanced Timer 刹车控制 1：使能；0：禁止。
14	TIM6	rw	0x00	VC2 filter 结果输出到 TIM6 捕获输入 CHB 使能 1：使能；0：禁止。
13	INV_TIM6	rw	0x00	VC2 filter 结果输出到 TIM6 反向使能 1：使能反向；0：禁止反向，输入与 VC 输出同向。

Bit	Field	Type	Reset	Description
12	TIM5	rw	0x00	VC2 filter 结果输出到 TIM5 捕获输入 CHB 使能 1：使能；0：禁止。
11	INV_TIM5	rw	0x00	VC2 filter 结果输出到 TIM5 反向使能 1：使能反向；0：禁止反向，输入与 VC 输出同向。
10	TIM4	rw	0x00	VC2 filter 结果输出到 TIM4 捕获输入 CHB 使能 1：使能；0：禁止。
9	INV_TIM4	rw	0x00	VC2 filter 结果输出到 TIM4 反向使能 1：使能反向；0：禁止反向，输入与 VC 输出同向。
8:6	Res.			保留
5	TIMBK	rw	0x00	VC2 filter 结果输出到 Timer0/1/2/3 刹车控制 1：使能；0：禁止。
4	TIM3RCLR	rw	0x00	VC2 filter 结果输出到 TIM3 REFCLR 使能控制 1：使能；0：禁止。
3	TIM2RCLR	rw	0x00	VC2 filter 结果输出到 TIM2 REFCLR 使能控制 1：使能；0：禁止。
2	TIM1RCLR	rw	0x00	VC2 filter 结果输出到 TIM1 REFCLR 使能控制 1：使能；0：禁止。
1	TIM0RCLR	rw	0x00	VC2 filter 结果输出到 TIM0 REFCLR 使能控制 1：使能；0：禁止。
0	INV_Timer	rw	0x00	VC2 filter 结果输出反向到各 TIM0/1/2/3/REFCLR 1：使能反向；0：禁止反向，输入与 VC 输出同向。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31 低电压检测器 (LVD)

31.1 LVD 简介

LVD 可用于监测 VCC 及芯片管脚的电压。当被监测电压与 LVD 阈值的比较结果满足触发条件时，LVD 会产生中断或复位信号，用户可根据该信号执行一些紧急任务。

LVD 具有以下特性：

- 4 路监测源，AVCC、PC13、PB08、PB07；
- 16 阶阈值电压，1.8V~3.3V 可选；
- 8 种触发条件，高电平、上升沿、下降沿组合；
- 2 种触发结果，复位、中断；
- 8 阶滤波配置，防止误触发；
- 具备迟滞功能，强力抗干扰。

31.2 LVD 框图

LVD 框图如图 226所示：

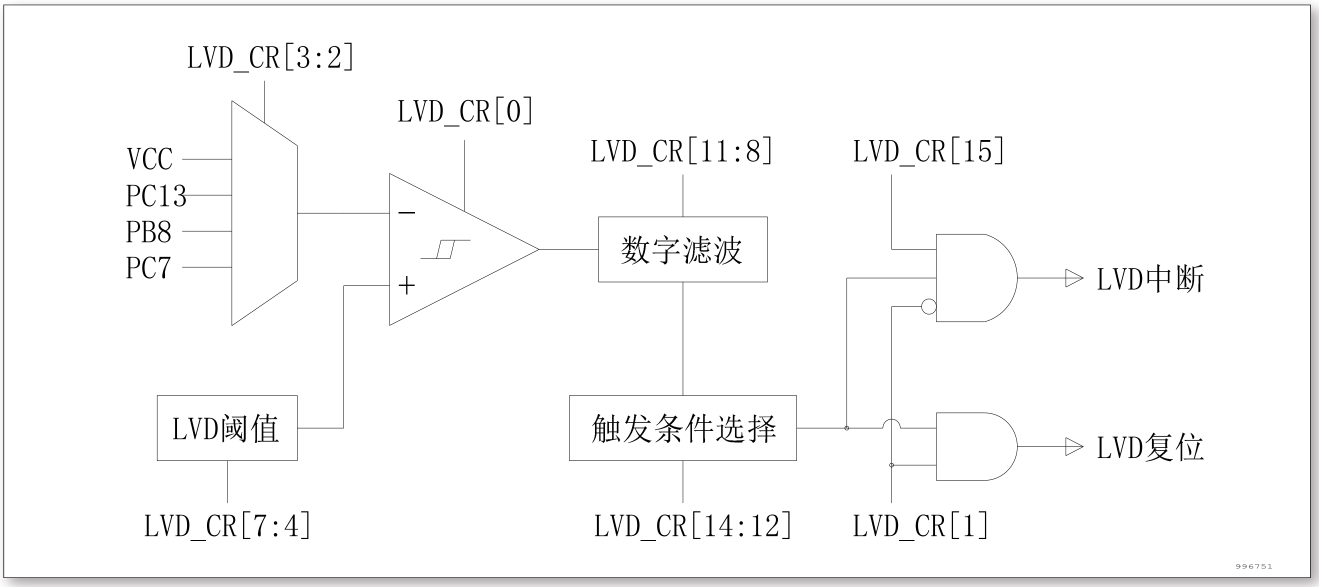

图 226. LVD 框图

31.3 迟滞功能

LVD 内置的电压比较器具有迟滞功能，可以增强芯片的抗干扰能力。LVD 的输出信号会等到输入信号高于或低于阈值电压 20mV 后才发生翻转，其输入输出信号示意如图 227：

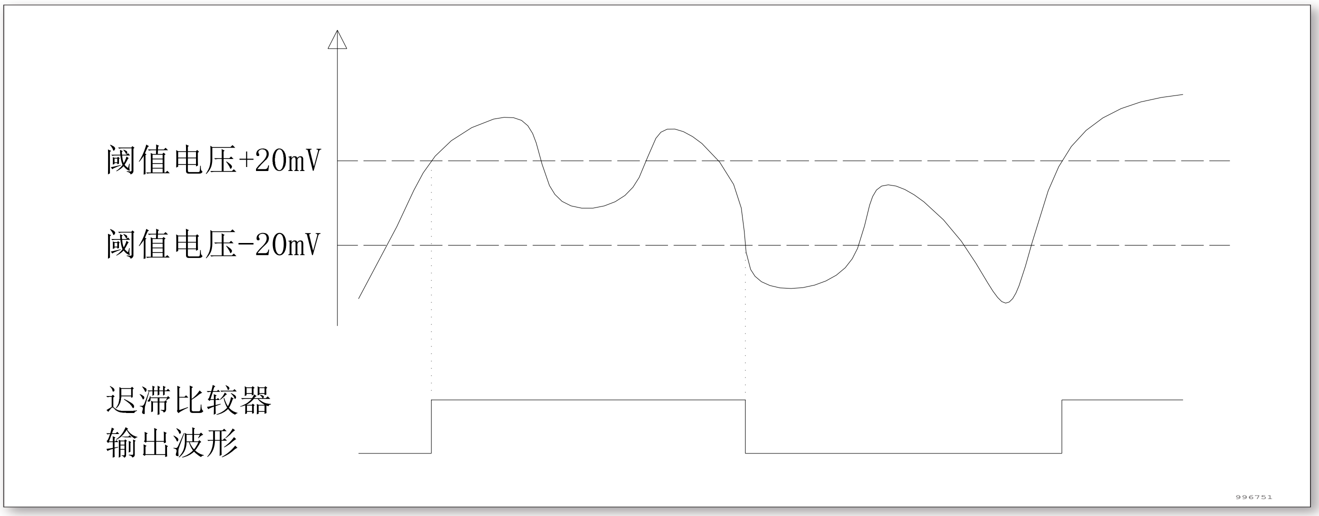

图 227. LVD 迟滞响应

31.4 数字滤波

如果芯片的工作环境恶劣，迟滞比较器的输出会出现噪声信号。使能数字滤波模块，则迟滞比较器的输出波形中脉宽小于 LVD_CR.Debounce_time 的噪声信号都可以被滤除。禁止数字滤波模块，则数字滤波模块的输入输出信号相同。使能数字滤波模块，滤波示意如图 228所示：

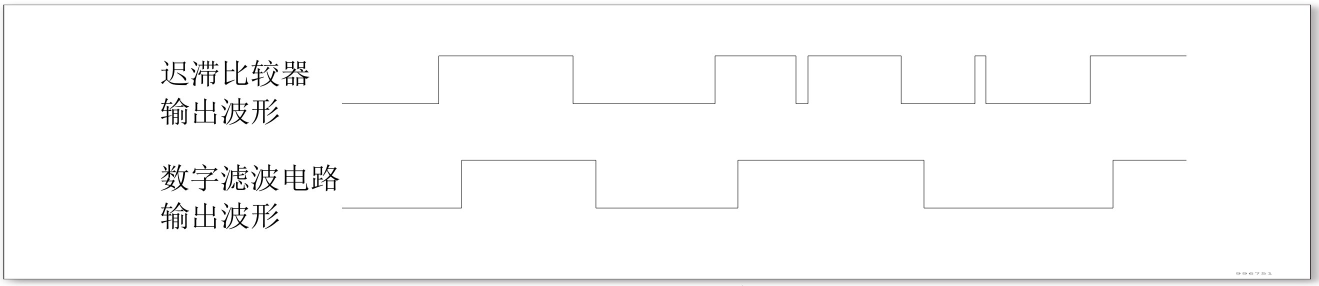

图 228. LVD 滤波输出

31.5 配置示例

31.5.1 LVD 配置为低电压复位

在本模式下，监测电压低于阈值电压时复位 MCU。

配置方法如下所示：

Step1：配置 LVD_CR.Source_sel，选择待监测的电压来源。

Step2：配置 LVD_CR. VTDS，选择 LVD 的阈值电压。

Step3：配置 LVD_CR. Debounce_time，选择 LVD 滤波时间。

Step4：配置 LVD_CR. FLTEN，使能 LVD 滤波。

Step5：设置 LVD_CR. HTEN 为 1，选择高电平触发 LVD 动作。

Step6：设置 LVD_CR.ACT 为 1，选择 LVD 动作为复位。

Step7：设置 LVD_CR.LVDEN 为 1，使能 LVD。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

31.5.2 LVD 配置为电压变化中断

在本模式下，监测电压高于或低于阈值电压时产生中断。

配置方法如下所示：

Step1：配置 LVD_CR.Source_sel，选择待监测的电压来源。

Step2：配置 LVD_CR. VTDS，选择 LVD 的阈值电压。

Step3：配置 LVD_CR. Debounce_time，选择 LVD 滤波时间。

Step4：配置 LVD_CR. FLTEN，使能 LVD 滤波。

Step5：设置 LVD_CR. RTEN 和 LVD_CR.FTEN 为 1，选择电平变化触发 LVD 动作。

Step6：设置 LVD_CR.ACT 为 0，选择 LVD 动作为中断。

Step7：设置 LVD_CR.IE 为 1，使能 LVD 中断。

Step8：使能 NVIC 中断向量表中的 LVD 中断。

Step9：设置 LVD_CR.LVDEN 为 1，使能 LVD。

Step10：在 LVD 的中断服务程序中执行用户需要进行的操作；退出中断服务程序前向 LVD_IFR 写入 0x00 以清除中断标志。

31.6 LVD 寄存器

表 122. LVD 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x028	LVD_CR	LVD 配置寄存器	0x0000 0000	小节 31.6.1
0x02C	LVD_IFR	LVD 中断标志寄存器	0x0000 0000	小节 31.6.2

31.6.1 LVD 配置寄存器（LVD_CR）

起始地址：0x40002400

地址偏移：0x028

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IE	HTEN	RTEN	FTEN	Debounce_time			FLTEN	VTDS				Source_sel		ACT	LVDEN
RW	RW	RW	RW	RW			RW	RW				RW		RW	RW

Bit	Field	Type	Reset	Description
31:16	Reserved			保留

Bit	Field	Type	Reset	Description
15	IE	rw	0x00	LVD 中断使能 1：使能； 0：禁止。
14	HTEN	rw	0x00	高电平触发使能 (被监测电压低于阈值电压) 1：使能； 0：禁止。
13	RTEN	rw	0x00	上升沿触发使能 (被监测电压从高于阈值电压变为低于阈值电压) 1：使能； 0：禁止。
12	FTEN	rw	0x00	下降沿触发使能 (被监测电压从低于阈值电压变为高于阈值电压) 1：使能； 0：禁止。
11:9	Debounce_time	rw	0x00	数字滤波时间配置 111：滤波时间大约为 28.8ms 110：滤波时间大约为 7.2ms 101：滤波时间大约为 1.8ms 100：滤波时间大约为 450us 011：滤波时间大约为 112us 010：滤波时间大约为 28us 001：滤波时间大约为 14us 000：滤波时间大约为 7us 注：滤波时间仅在 FLTEN 为 1 时有效。
8	FLTEN	rw	0x00	数字滤波使能配置 1: 使能数字滤波 0：禁止数字滤波

Bit	Field	Type	Reset	Description
7:4	VTDS	rw	0x00	LVD 阈值电压选择 1111：3.3v 1110：3.2v 1101：3.1v 1100：3.0v 1011：2.9v 1010：2.8v 1001：2.7v 1000：2.6v 0111：2.5v 0110：2.4v 0101：2.3v 0100：2.2v 0011：2.1v 0010：2.0v 0001：1.9v 0000：1.8v
3:2	Source_sel	rw	0x00	LVD 监测来源选择 11：PB07 端口输入电压 10：PB08 端口输入电压 01：PC13 端口输入电压 00：AVCC 电源电压
1	ACT	rw	0x00	LVD 触发动作选择 1：系统复位 0：NVIC 中断
0	LVDEN	rw	0x00	LVD 使能控制 1：使能 LVD 0：禁止 LVD

31.6.2 LVD 中断标志寄存器 (LVD_IFR)

起始地址：0x40002400

地址偏移：0x02C

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved														LVD_Filter	INTF
														RO	RW0

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
31:2	Reserved			保留
1	LVD_Filter	ro	0x00	LVD Filter 后的状态
0	INTF	ro	0x00	LVD 中断标志： 1：发生 LVD 中断； 0：未发生中断； 写 0 清除中断标志，写 1 无效。

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

32 运算放大器 (OPA)

OPA 模块适用于简易放大器和电压跟随器应用。OPA 的输入范围是 0V 到 AVCC，输出范围是 0.1V 到 AVCC-0.2V。

32.1 OPA 特性

- 1 个独立配置运放；
- OPA 的输入范围是 0 到 AVCC，输出范围是 0.1V 到 AVCC-0.2V 可编程增益；
- 可配置为以下模式。

  -通用运放模式（general purpose OPA）；

  -DAC 电压跟随器。

32.2 OPA 功能描述

OPA 可以通过使用外部元件组成放大器来放大小信号模拟输入信号，输出为放大后的信号。

32.2.1 OPA (复用为 DAC buffer)

OPA 的示意图如图 229：

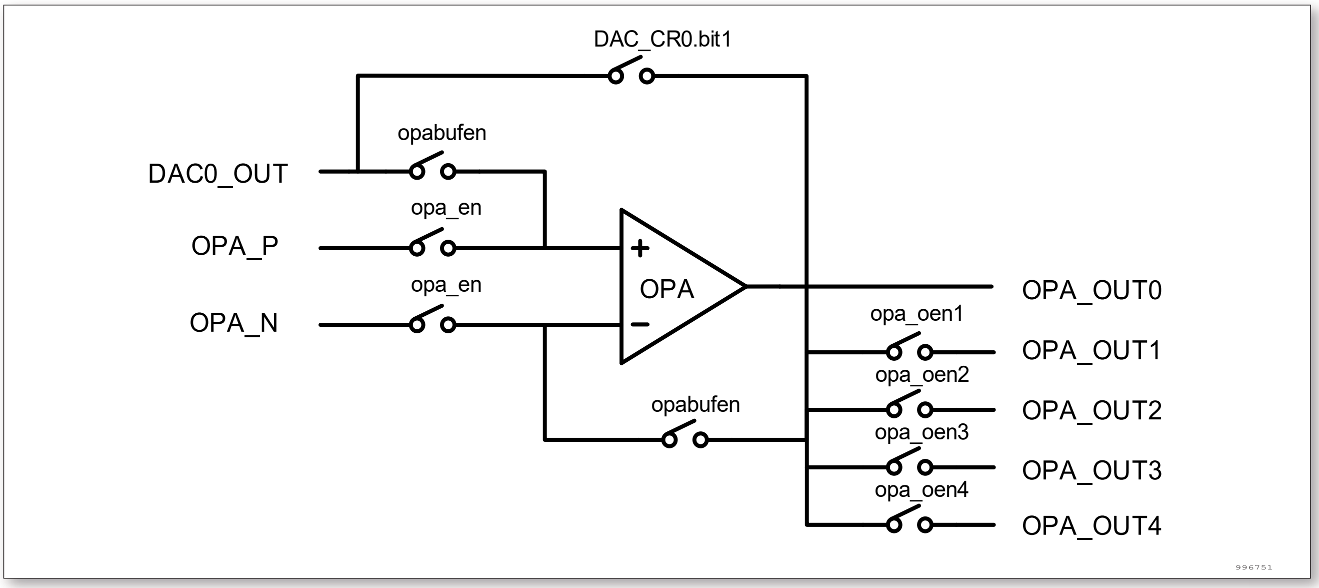

图 229. OPA 示意图

32.3 OPA 校零

OPA 支持不同模式的自动校零功能，可以软件自己控制，也可以硬件控制。高精度 OPA 应用，在每次使用 OPA 前，需要 OPA 校零。校零建议时间 10us 以上。校零后请立即采样。

32.3.1 软件控制

- 使能需要校零的 OPA 的 OPA_CRx.AZEN 为 1；
- 设置 OPA_CR. CLK_SW_SET 为 1；
- 软件写 OPA_CR. AZ_PULSE 为 1；

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

- 设定校零时间，等待校零时间结束，清除 OPA_CR. AZ_PULSE 为 0。

32.3.2 软件触发控制

- 使能需要校零的 OPA 的 OPA_CRx.AZEN 为 1；
- 选择需要校零的时间参数 OPA_CR.CLK_SEL；
- 设置软件触发 OPA_CR. TRIGGER。

32.3.3 ADC 触发控制

- 使能需要校零的 OPA 的 OPA_CRx.AZEN 为 1；
- 选择需要校零的时间参数 OPA_CR.CLK_SEL；
- 使能 ADC 触发 ADCTR_EN。

32.4 OPA 寄存器

表 123. OPA 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x030	OPA_CR0	OPA 控制寄存器 0	0x0000 0180	小节 32.4.1
0x03C	OPA_CR1	OPA 控制寄存器 1	0x0000 0000	小节 32.4.2

32.4.1 OPA 控制寄存器 0 (OPA_CR0)

起始地址：0x40002400

地址偏移：0x030

复位值：0x0000 0180

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						BIASSEL			OPA OEN3	OPA OEN3	OPA OEN2	OPAO EN1	OPA BUF EN	AZEN	EN
						RW			RW	RW	RW	RW	RW	RW	RW

Bit	Field	Type	Reset	Description
31:10	Reserved			保留
9:7	BIASSEL	rw	0x03	OPA 偏置电流选择 011 不需要更改配置，使用缺省值就可以
6	OPA_OEN4	rw	0x00	OP3 输出 4 使能
5	OPA_OEN3	rw	0x00	OP3 输出 3 使能
4	OPA_OEN2	rw	0x00	OP3 输出 2 使能
3	OPA_OEN1	rw	0x00	OP3 输出 1 使能

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
2	OPABUFEN	rw	0x00	DAC 使用 OP3 单位增加缓存使能，与 OPA3 使能不能同时使能
1	AZEN	rw	0x00	自动校零使能
0	OPA_EN	rw	0x00	OPA 使能

32.4.2 OPA 控制寄存器 1 (OPA_CR1)

起始地址：0x40002400

地址偏移：0x03C

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								CLK_SEL				CLK_SW_SET	AZ_PULSE	TRIGGER	ADCTR_EN
								RW				RW	RW	W1	RW

Bit	Field	Type	Reset	Description
31:8	Reserved			保留
7:4	CLK_SEL	rw	0x00	自动校准脉冲宽度设置 0000:1 个 PCLK 周期 0001:2 个 PCLK 周期 0010:4 个 PCLK 周期 0011:8 个 PCLK 周期 0100:16 个 PCLK 周期 0101:32 个 PCLK 周期 0110:64 个 PCLK 周期 0111:128 个 PCLK 周期 1000:256 个 PCLK 周期 1001:512 个 PCLK 周期 1010:1024 个 PCLK 周期 1011:2048 个 PCLK 周期 11xx:4096 个 PCLK 周期 建议自动校准时间为 10us
3	CLK_SW_SET	rw	0x00	自动校零选择 1：软件校准使能 0：软件校准禁止，软件触发校准使能

Bit	Field	Type	Reset	Description
2	AZ_PULSE	rw	0x00	软件校准 1：校准 0：无效
1	TRIGGER	w1	0x00	软件触发校准设置，自动清零 1：软件触发校准 0：无效
0	ADCTR_EN	rw	0x00	ADC 启动触发 OPA 自动校准使能 1：使能 0：禁止 高精度 OPA 应用，在每次使用 OPA 前, 需要 OPA 校零。校零建议时间 10us 以上。校零后请立即采样。

33 模拟其它寄存器

33.1 模拟其它寄存器寄存器

表 124. 模拟其它寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x000	BGR_CR	BGR 控制寄存器	0x0000 0000	小节 33.1.1

33.1.1 BGR 控制寄存器 (BGR_CR)

起始地址：0x40002400

地址偏移：0x000

复位值：0x0000 0000

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
</table>

<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="14" rowspan="2">Reserved</td><td>TS_EN</td><td>BGR_EN</td></tr>
<tr><td>RW</td><td>RW</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:2	Reserved			保留
1	TS_EN	rw	0x00	内置温度传感器使能控制 1：使能内部温度传感器, 必须使能 BGR 0：禁止内部温度传感器 注意：温度传感器使能 20us 后，才能输出稳定信号。
0	BGR_EN	rw	0x00	BGR 使能控制 1：使能 BGR 0：禁止 BGR 注意： 1) PERI0_CLKEN.ADC 为 1 时，才可以操作此寄存器。 2) BGR 使能 20us 后，才能输出稳定的高精度参考电压。BGR 稳定后才可以被其它模块使用，故用户操作中应加入等待 BGR 稳定的步骤。 3) 当使用 ADC,VC,OPA 时，必须使能 BGR

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

34 SWD 调试接口

本系列使用 ARM Cortex-M0+ 内核，该内核具有硬件调试模块 SWD，支持复杂的调试操作。硬件调试模块允许内核在取指 (指令断点) 或访问数据 (数据断点) 时停止。内核停止时，内核的内部状态和系统的外部状态都可以在 IDE 中进行查询。完成查询后，内核和外设可以被复原，程序将继续执行。当本系列微控制器连接到调试器并开始调试时，调试器将使用内核的硬件调试模块进行调试操作。

注意：

-SWD 能把 DeepSleep 唤醒，请注意不要以 SWD 接口断开作为 DeepSleep 的状态。

34.1 SWD 调试附加功能

本产品使用了 ARM Cortex-M0+ CPU，该内核包含用于高级调试功能的硬件扩展，因此本产品所拥有的调试功能与 Cortex-M0+ 一致。调试扩展允许内核可以在取指（指令断点）或取访问数据（数据断点）时停止内核。内核停止时，可以查询内核的内部状态和系统的外部状态。查询完成后，将恢复内核和系统并恢复程序执行。

当调试主机与 MCU 相连并进行调试时，将使用调试功能。

Cortex®-M0+ 内核中内置的调试功能是 ARM® CoreSight 设计套件的一部分。ARM® Cortex®-M0+ 内核提供集成片上调试支持。它包括：

- SW-DP：串行线
- BPU：断点单元
- DWT：数据观察点触发

注：

-有关 ARM® Cortex®-M0+ 内核支持的调试功能的详细信息，请参见 Cortex®- M0+ 技术参考手册。

34.2 ARM ® 参考文档

- Cortex®-M0+ 技术参考手册 (TRM)；
- 可从 www.infocenter.arm.com 获取；
- ARM® 调试接口 V5；
- ARM® CoreSight 设计套件版本 r1p1 技术参考手册。

34.3 调试端口引脚

34.3.1 SWD 端口引脚

本系列的 SWD 接口，需要用到 2 个引脚，如表 125所示。

表 125: SWD 端口引脚

SWD 端口名称	调试功能	引脚分配
SWCLK	串行时钟	PA14
SWDIO	串行数据输入/输出	PA13

34.3.2 SW-DP 引脚分配

如果烧录程序时使能了【加密芯片】选项，则上电后 SWD 调试功能被禁止。如果烧录程序时没有使能【加密芯片】选项，则上电后 PA13/PA14 引脚均被初始化为可被调试器使用的专用引脚。用户可设置 SYSCTRL1 .SWD_USE_IO 寄存器来禁止 SWD 引脚的调试功能，SWD 引脚将被释放以用作普通 GPIO。SWD 引脚的配置与功能小结如表 126所示：

表 126: SWD 端口引脚

【加密芯片】选项	SWD_USE_IO 配置	PA13/PA14 功能
加密	0	NA
加密	1	GPIO
不加密	0	SWD
不加密	1	GPIO

34.3.3 SWD 引脚上的内部上拉

用户软件释放 SW I/O 后，GPIO 控制器便会控制这些引脚。GPIO 控制寄存器的复位状态会将 I/O 置于等效的状态：

- SWDIO：输入上拉
- SWCLK：输入上拉

由于内置上拉和下拉电阻，因此无需添加外部电阻。

34.4 SWD 端口

34.4.1 SWD 协议简介

此同步串行协议使用两个引脚：

- SWCLK：从主机到目标的时钟
- SWDIO：双向

利用该协议，可以同时读取和写入两组寄存器组（DPACC 寄存器组和 APACC 寄存器组）。传输数据时，LSB 在前。

对于 SWDIO 双向管理，必须在电路板上对线路进行上拉（ARM® 建议采用 100 k）。这些上拉电阻可在内部配置。无需外部上拉电阻。

每次在协议中更改 SWDIO 的方向时，都会插入转换时间，此时线路即不受主机驱动也不受目标驱动。默认情况下，此转换时间为一位时间，但可以通过配置 SWCLK 频率来调整。

34.4.2 SWD 协议序列

每个序列包括三个阶段：

1. 主机发送的数据包请求（8 位）；
2. 目标发送的确认响应（3 位）；
3. 主机或目标发送的数据传输阶段（33 位）。

表 127: 主机发送的数据包请求

位	名称	说明
0	启动	必须为 1
1	APnDP	0：DP 访问；1：AP 访问
2	RnW 0：	写请求；1：读请求
4:3	A[3:2]	DP 或 AP 寄存器的地址字段
5	奇偶校验	前面几位的单位奇偶校验
6	停止	0
7	驻留	不受主机驱动。由于存在上拉，因此必须由目标读为 1

有关 DPACC 和 APACC 寄存器的详细说明，请参见 Cortex®-M0+ TRM。

数据包请求后面始终为转换时间（默认 1 位），此时主机和目标都不会驱动。

表 128: 目标发送的确认响应

位	名称	说明
0	ACK	001：FAULT 010：WAIT 100：OK

仅当发生 READ 事务或者接收到 WAIT 或 FAULT 确认时，ACK 响应后才必须是转换时间。

表 129: 主机或目标发送的数据传输

位	名称	说明
0:31	WDATA 或 RDATA	写入或读取数据
32	奇偶校验	32 个数据位的单奇偶校验

仅当发生 READ 事务时，DATA 传输后才必须是转换时间。

34.4.3 SW-DP 状态机（复位、空闲状态、ID 代码）

SW-DP 的状态机有一个用于标识 SW-DP 的内部 ID 代码。该代码符合 JEP-106 标准。此 ID 代码是默认的 ARM® 代码，设置为 0x0BB11477（相当于 Cortex®-M0+）。

注意：

-在目标读取此 ID 代码前，SW-DP 状态机是不工作的。

- 在上电复位后或者线路处于高电平超过 50 个周期后，SW-DP 状态机处于复位状态。
- 如果在复位状态后线路处于低电平至少两个周期，SW-DP 状态机处于空闲状态。
- 复位状态后，该状态机必须首先进入空闲状态，然后对 DP-SW ID CODE 寄存器执行读访问。否则，目标将在另一个事务上发出 FAULT 确认响应。

34.4.4 DP 和 AP 读/写访问

- 不延迟对 DP 的读访问：可以立即发送目标响应（如果 ACK=OK），也可以延迟发送目标响应（如果 ACK=WAIT）。
- 延迟对 AP 的读访问。这意味着会在下次传输时返回访问结果。如果要执行的下次访问不是 AP 访问，则必须读取 DP-RDBUFF 寄存器来获取结果。
- 每次进行 AP 读访问或 RDBUFF 读请求时都会更新 DP-CTRL/STAT 寄存器的 READOK 标志，以便了解 AP 读访问是否成功。
- SW-DP 有写缓冲区（用于 DP 或 AP 写入），这样即使在其它操作仍未完成时，也可以接受写入操作。如果写缓冲区已满，则目标确认响应为 WAIT。但 IDCODE 读取、CTRL/STAT 读取或 ABORT 写入除外，这几项操作在写缓冲区已满时也会被接受。
- 由于存在异步时钟域 SWCLK 和 HCLK，因此写操作后（奇偶校验位后）还需要两个额外的 SWCLK 周期，以使写入操作在内部生效。应在将线路驱动为低电平时（空闲状态）应用这些周期。

在写 CTRL/STAT 寄存器以提出一个上电请求时，这一点特别重要。否则下一个操作（在内核上电后才有效的操作）会立即执行，这将导致失败。

34.4.5 SW-DP 寄存器

当 APnDP=0 时能够访问表 130中的寄存器：

表 130: SW-DP 寄存器

A[3:2]	RW	SELECT 寄存器的 CTRLSEL 位	寄存器	注释
00	读取		IDCODE	制造商代码设置为 Cortex®-M0+ 的默认 ARM® 代码。 0x0BB11477（标识 SW-DP）
00	写		ABORT	
01	读 / 写	0	DP-CTRL/STAT	目的： -请求系统或调试上电 -配置 AP 访问的传输操作 -控制比较和验证操作 -读取一些状态标志（上溢和上电确认）。
01	读 / 写	1	WIRE CONTROL	用于配置物理串行端口协议（如转换时间的持续时间）。
10	读取		READ RESEND	允许从已损坏的调试软件传输中恢复读取数据，无需重复执行原始 AP 传输。
10	写		SELECT	用于选择当前访问端口和活动的 4 字寄存器窗口。
11	读 / 写		READ BUFFER	由于已发出 AP 访问，因此该读缓冲区非常有用（在执行下个 AP 事务时提供读取 AP 请求的结果）。此读取缓冲区捕获 AP 中的数据，显示为前一次读取的结果，无需启动新操作。

34.4.6 SW-AP 寄存器

当 APnDP=1 时能够访问表 131中的寄存器：有多个 AP 寄存器，这些寄存器按以下组合进行寻址：

- 移位值 A[3:2]；
- DP SELECT 寄存器的当前值。

表 131: SW-AP 寄存器

地址	A[3:2] 值	说明
0x0	00	保留，必须保持复位值。
0x4	01	DP CTRL/STAT 寄存器。用于： -请求系统或调试上电； -配置 AP 访问的传输操作； -控制比较和验证操作； -读取一些状态标志（上溢和上电确认）。
0x8	10	DP SELECT 寄存器：用于选择当前访问端口和活动的 4 字寄存器窗口。 -位 31:24：APSEL：选择当前 AP (select the current AP)； -位 23:8：保留； -位 7:4：APBANKSEL：在当前 AP 上选择活动的 4 字寄存器窗口； -位 3:0：保留。
0xC	11	DP RDBUFF 寄存器：用于通过调试器在执行一系列操作后获取最后结果。

34.5 内核调试

通过内核调试寄存器调试内核。通过调试访问端口调试访问这些寄存器。它由四个寄存器组成，如表 132所示：

表 132: 内核调试寄存器

寄存器	说明
DHCSR	32 位调试停止控制和状态寄存器： 此寄存器提供有关处理器状态的信息，能够使内核进入调试停止状态并提供处理器步进功能。
DCRSR	17 位调试内核寄存器选择器寄存器： 此寄存器选择需要进行读写操作的处理器寄存器。
DCRDR	32 位调试内核寄存器数据寄存器： 此寄存器保存在寄存器与 DCRSR（选择器）寄存器选择的处理器之间读取和写入的数据。
DEMCR	32 位调试异常和监视控制寄存器： 此寄存器提供向量捕获和调试监视控制。

这些寄存器在系统复位时不复位。它们只能通过上电复位来复位。有关更多详细信息，请参见 Cortex®-M0+

TRM。

为了在复位后立即使内核进入调试停止状态，必须：

- 使能调试和异常监视控制寄存器的位 0 (VC_CORRESET)。
- 使能调试停止控制和状态寄存器的位 0 (C_DEBUGEN)。

34.6 BPU（断点单元）

Cortex®-M0+ BPU 实现提供四个断点寄存器。

34.6.1 BPU 功能

处理器断点实现了基于 PC 的断点功能。

有关 BPU CoreSight 标识寄存器及其地址和访问类型的更多信息，请参见 ARMv6-M ARM® 和 ARM® CoreSight 组件技术参考手册。

34.7 DWT（(数据观察点）

Cortex®-M0+ DWT 实现提供了两个观察点寄存器组。

34.7.1 DWT 功能

处理器观察点实现了数据地址和基于 PC 的观察点功能（即 PC 采样寄存器），并支持比较器地址掩码，如 ARMv6-M ARM® 中所述。

34.7.2 DWT 程序计数器采样寄存器

实现数据观察点单元的处理器还实现了 ARMv6-M 可选 DWT 程序计数器采样寄存器 (DWT_PCSR)。此寄存器允许调试程序定期采样 PC，无需停止处理器。这可提供粗略分析。有关更多信息，请参见 ARMv6-M ARM®。

Cortex®-M0+ DWT_PCSR 记录通过条件代码和指令以及未通过条件代码的指令。

34.8 MCU 调试组件 (DBG)

MCU 调试组件帮助调试器为以下各项提供支持：

- 低功耗模式；
- 断点期间的定时器、看门狗的时钟控制。

34.8.1 对低功耗模式的调试支持

要进入低功耗模式，必须执行指令 WFI 或 WFE。

MCU 支持多个低功耗模式，这些模式可以禁止 CPU 时钟或降低 CPU 功耗。

内核不允许在调试会话期间关闭 FCLK 或 HCLK。由于调试期间需要使用它们进行调试连接，因此其必须保持激活状态。MCU 集成了特殊方法，允许用户在低功耗模式下调试软件。

34.8.2 对定时器、看门狗的调试支持

断点期间，必须选择定时器和看门狗的计数器的行为方式：

- 在产生断点时，计数器继续计数。例如，当 PWM 控制电机时，通常需要这种方式。

- 在产生断点时，计数器停止计数。用于看门狗时需要这种方式。

34.9 调试模式寄存器

表 133. 调试模式寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x00	DEBUG_ACTIVE	调试模式模块工作状态控制	0x0000 0FFF	小节 34.9.1

34.9.1 调试模式模块工作状态控制 (DEBUG_ACTIVE)

起始地址：0x40002038

地址偏移：0x00

复位值：0x0000 0FFF (仅在 SWD 调试模式下，此寄存器设置才起作用)

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="16">Reserved</td></tr>
</table>

<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="3" rowspan="2">Reserved</td><td>LPTIM1</td><td>TIM3</td><td rowspan="2">Res.</td><td>RTC</td><td>WDT</td><td>PCA</td><td>TIM6</td><td>TIM5</td><td>TIM4</td><td>LPTIM</td><td>TIM2</td><td>TIM1</td><td>TIM0</td></tr>
<tr><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td><td>RW</td></tr>
</table>

Bit	Field	Type	Reset	Description
31:13	Reserved			保留
12	LPTIM1	rw	0x00	调试时，LpTimer1 计数功能配置 1：在 SWD 调试界面下，暂停 LPTimer1 计数功能 0：在 SWD 调试界面下，LPTimer1 正常计数功能
11	TIM3	rw	0x01	调试时，Timer3 计数功能配置 1：在 SWD 调试界面下，暂停 Timer3 计数功能 0：在 SWD 调试界面下，Timer3 正常计数功能
10	Reserved			保留
9	RTC	rw	0x01	调试时，RTC 计数功能配置 1：在 SWD 调试界面下，暂停 RTC 计数功能 0：在 SWD 调试界面下，RTC 正常计数功能
8	WDT	rw	0x01	调试时，WDT 计数功能配置 1：在 SWD 调试界面下，暂停 WDT 计数功能 0：在 SWD 调试界面下，WDT 正常计数功能
7	PCA	rw	0x01	调试时，PCA 计数功能配置 1：在 SWD 调试界面下，暂停 PCA 计数功能 0：在 SWD 调试界面下，PCA 正常计数功能
6	TIM6	rw	0x01	调试时，Timer6 计数功能配置 1：在 SWD 调试界面下，暂停 Timer 计数功能 0：在 SWD 调试界面下，Timer 正常计数功能

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
5	TIM5	rw	0x01	调试时，Timer5 计数功能配置 1：在 SWD 调试界面下，暂停 Timer 计数功能 0：在 SWD 调试界面下，Timer 正常计数功能
4	TIM4	rw	0x01	调试时，Timer4 计数功能配置 1：在 SWD 调试界面下，暂停 Timer 计数功能 0：在 SWD 调试界面下，Time r 正常计数功能
3	LPTIM	rw	0x01	调试时，LpTimer0 计数功能配置 1：在 SWD 调试界面下，暂停 LPTimer0 计数功能 0：在 SWD 调试界面下，LPTimer0 正常计数功能
2	TIM2	rw	0x01	调试时，Timer2 计数功能配置 1：在 SWD 调试界面下，暂停 Timer 计数功能 0：在 SWD 调试界面下，Timer 正常计数功能
1	TIM1	rw	0x01	调试时，Timer1 计数功能配置 1：在 SWD 调试界面下，暂停 Timer 计数功能 0：在 SWD 调试界面下，Timer 正常计数功能
0	TIM0	rw	0x01	调试时，Timer0 计数功能配置 1：在 SWD 调试界面下，暂停 Timer 计数功能 0：在 SWD 调试界面下，Timer 正常计数功能

35 唯一设备 ID 寄存器（80 位）

唯一设备标识符最适合：

- 用作序列号；
- 在对内部 Flash 进行编程前将唯一 ID 与软件加密原语和协议结合使用时用作安全密钥以提高 Flash 中代码的安全性；
- 激活安全自举过程等。

80 位的唯一设备标识符提供了一个对于任何设备和任何上下文都唯一的参考号码。用户永远不能改变这些位。

80 位的唯一设备标识符也可以以单字节/半字/字等不同方式读取，然后使用自定义算法连接起来。

基址：0x0010 0E74

表 134: UID 规则

<table>
<tr><th rowspan="2">偏移地址</th><th rowspan="2">描述</th><th colspan="8">UID Bits（80 bits）</th></tr>
<tr><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td>0</td><td rowspan="6">Lot Number</td><td colspan="8">UID[7:0]</td></tr>
<tr><td>1</td><td colspan="8">UID[15:8]</td></tr>
<tr><td>2</td><td colspan="8">UID[23:16]</td></tr>
<tr><td>3</td><td colspan="8">UID[31:24]</td></tr>
<tr><td>4</td><td colspan="8">UID[39:32]</td></tr>
<tr><td>5</td><td colspan="8">UID[47:40]</td></tr>
<tr><td>6</td><td>X Coordinate on the wafer</td><td colspan="8">UID[55:48]</td></tr>
<tr><td>7</td><td>Y Coordinate on the wafer</td><td colspan="8">UID[63:56]</td></tr>
<tr><td>8</td><td>Wafer Number</td><td colspan="8">UID[71:64]</td></tr>
<tr><td>9</td><td>Rev ID</td><td colspan="8">UID[79:72]</td></tr>
</table>

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

36 附录 A SysTick 定时器

36.1 SysTick 定时器简介

OS 要想支持多任务，就需要周期执行上下文切换，这样就需要有定时器之类的硬件资源打断程序执行。当定时器中断产生时，处理器就会在异常处理中进行 OS 任务调度，同时还会进行 OS 维护的工作。Cortex-M0 处理器中有一个称为 SysTick 的简单定时器，用于产生周期性的中断请求。

SysTick 为 24 位的定时器，并且向下计数。定时器的计数减到 0 后，就会重新装载一个可编程的数值，并且同时产生 SysTick 异常（异常编号为 15），该异常事件会引起 SysTick 异常处理的执行，这个过程是 OS 的一部分。

对于不需要 OS 的系统，SysTick 定时器也可以用作其他用途，比如定时、计时或者为需要周期执行的任务提供中断源。SysTick 异常的产生是可控的，如果异常被禁止，仍然可以用轮询的方法使用 SysTick 定时器，比如检查当前的计数值或者轮询计数标志。

36.2 设置 SysTick

由于 SysTick 定时器的重新装载值和当前值在复位时都是未定义的，为了防止产生异想不到的结果，SysTick 的设置代码需要遵循一定的流程。

启动 → 禁止 SysTick→ 设置重载值寄存器 → 清除当前值寄存器 → 使能 SysTick→ 完成

SysTick 有一个校准值寄存器，它提供的信息有助于所需重载值的计算。如果微处理器中有这个定时的校准，那么这个寄存器的 TENMS 域中就会包含 10 毫秒所对应的计数值。

36.3 SysTick 寄存器

表 135. SysTick 寄存器概览

Offset	Acronym	Register Name	Reset	Section
0x00	SysTick_CTRL	SysTick 控制和状态寄存器	0x0000 0000	小节 36.3.1
0x04	SysTick_LOAD	SysTick 重装载寄存器	0xXXXX XXXX	小节 36.3.2
0x08	SysTick_VAL	SysTick 当前值寄存器	0xXXXX XXXX	小节 36.3.3
0x0C	SysTick_CALIB	SysTick 校准值寄存器	0xXXXX XXXX	小节 36.3.4

36.3.1 SysTick 控制和状态寄存器 (SysTick_CTRL)

起始地址：0XE000E010

地址偏移：0x00

复位值：0x0000 0000

Bit	Field	Type	Reset	Description
31:17	Reserved			保留
16	COUNTFLAG	r0	0x00	当定时器到 0 时，该位置 1，读取寄存器会被清零
15:3	Reserved			保留

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

Bit	Field	Type	Reset	Description
2	CLKSOURCE	rw	0x00	SysTick 时钟源选择 1: 使用内核时钟 0: 使用参考时钟
1	TICKINT	rw	0x00	SysTick 中断使能 1：中断使能 0：中断禁止
0	ENABLE	rw	0x00	置 1 时 SysTick 定时器使能，要不然计数会被禁止

36.3.2 SysTick 重装载寄存器（SysTick_LOAD）

起始地址：0XE000E010

地址偏移：0x04

复位值：0xXXXX XXXX

Bit	Field	Type	Reset	Description
31:24	Reserved			保留
23:0	RELOAD	rw	0xXX	SysTick 定时器重载值

36.3.3 SysTick 当前值寄存器（SysTick_VAL）

起始地址：0XE000E010

地址偏移：0x08

复位值：0xXXXX XXXX

Bit	Field	Type	Reset	Description
31:24	Reserved			保留
23:0	CURRENT	rw	0xXX	读出值为 SysTick 定时器的当前数值，写入任何值都会清除寄存器，COUNTFLAG 也会清零（不会引起 SysTick 异常）

36.3.4 SysTick 校准值寄存器（SysTick_CALIB）

起始地址：0XE000E010

地址偏移：0x0C

复位值：0xXXXX XXXX

Bit	Field	Type	Reset	Description
31	NOREF	ro	0xXX	如果读出值为 1，表示没有使用外部参考时钟，SysTick 使用内核时钟；如果是 0，则使用外部参考时钟（外部低速晶振 XTL）
30	SKEW	ro	0xXX	如果设为 1，则 TENMS 域不准确

Bit	Field	Type	Reset	Description
29:24	Reserved			保留
23:0	TENMS	ro	0xXX	10 毫秒校准值

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

37 型号命名

该芯片的命名规则如下图所示。

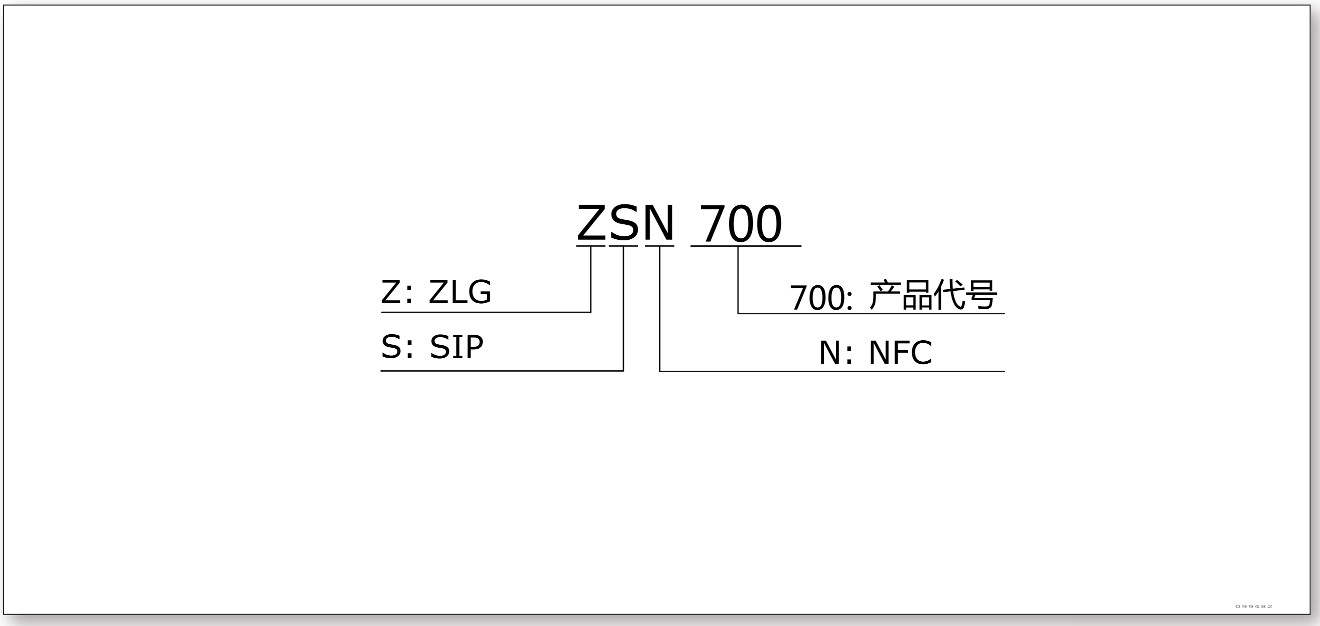

图 230. ZSN700 型号命名

38 表格

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

39 图片

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.

40 免责声明

应用信息

本应用信息适用于 ZSN700 的开发设计。客户在开发产品前，必须根据其产品特性给予修改并验证。

修改文档的权利

本着为用户提供更好服务的原则，广州致远微电子有限公司（下称“致远微电子”）在本手册中将尽可能地为用户呈现详实、准确的产品信息。但介于本手册的内容具有一定的时效性，致远微电子不能完全保证该文档在任何时段的时效性与适用性。致远微电子有权在没有通知的情况下对本手册上的内容进行更新，恕不另行通知。为了得到最新版本的信息，请尊敬的用户定时访问立功科技官方网站或者与致远微电子工作人员联系。感谢您的包容与支持！

专业 . 专注成就梦想

Dreams come true with professionalism and dedication.

广州致远微电子有限公司

更多详情请访问
ww.zlgmcu.com

欢迎拨打全国服务热线
400-888-2705

©2020 Guangzhou ZHIYUAN Micro Electronics Co., Ltd.