

FT62F08X

TIM1_CAPTURE Application note

目录

1. 高级定时器 TIM1	3
1.1. 特性	3
1.2. Timer1 相关寄存器汇总	4
1.3. 功能描述	17
2. 基本定时器 TIM4	47
2.1. 特性	47
2.2. TIM4 相关寄存器汇总	47
2.3. TIM4 时钟源	49
2.4. 预分频器	49
2.5. TIM4 中断	49
3. 应用范例	50
联系信息	63

FT62F08x TIM1_CAPTURE 应用

1. 高级定时器 TIM1

1.1. 特性

- 16bit 的向上计数、向下计数或者上/下计数器，支持自动重载；
- 支持可编程预分频的计数时钟；
- 支持 4 个独立的捕捉比较通道，通道可支持：
 - ✓ 输入捕捉
 - ✓ 输出比较
 - ✓ 边沿或中心对称 PWM
 - ✓ 单脉冲输出
 - ✓ 6 步 PWM
- PWM 互补输出和可编程死区时间；
- 可编程的重复计数器；
- 刹车功能，使输出停止在一个复位态或者一个预设状态
- 中断事件：
 - ✓ 更新事件：计数器溢出，计数器初始化
 - ✓ 触发事件：触发计数开始与停止，计数器初始化或外部触发事件
 - ✓ 输入捕捉事件
 - ✓ 输出比较事件
 - ✓ 刹车输入有效事件
- 外部时钟的触发计数
- 前沿消隐

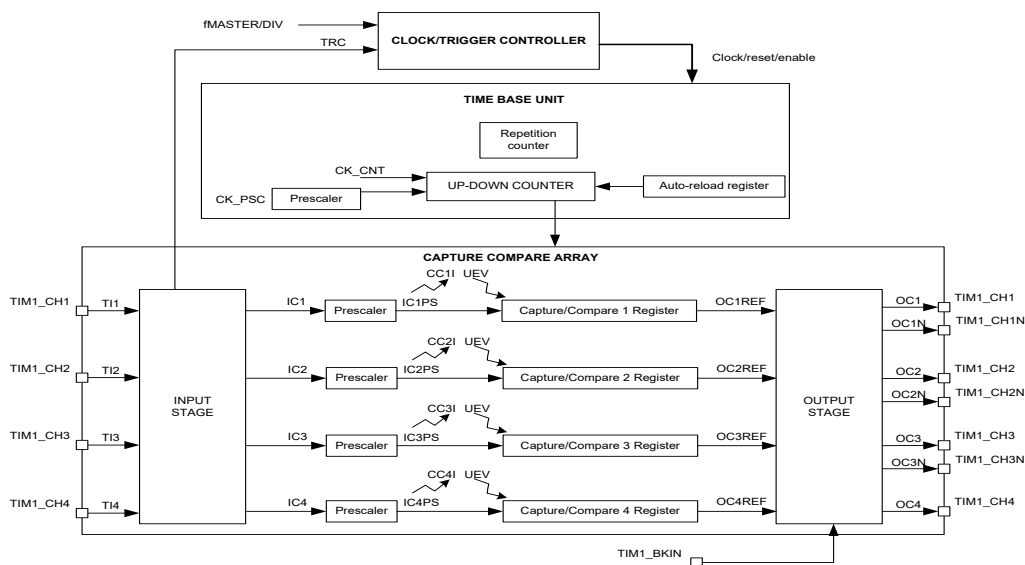


图 1-1 TIM1 原理框图

1.2. Timer1 相关寄存器汇总

名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	复位值
PCKEN	0x9A	TKEN	I2CEN	UARTEN	SPIEN	TIM4EN	TIM2EN	TIM1EN	ADCEN	0000 0000
CKOCON	0x95	SYSON	CCORDY	DTYSEL		CCOSEL[2:0]			CCOEN	0010 0000
TIM1CR1	0x211	T1ARPE	T1CMS[1:0]		T1DIR	T1OPM	T1URS	T1UDIS	T1CEN	0000 0000
TIM1SMCR	0x213	—	T1TS[2:0]			—	T1SMS[2:0]			-000 -000
TIM1IER	0x215	T1BIE	T1TIE	—	T1CC4IE	T1CC3IE	T1CC2IE	T1CC1IE	T1UIE	00-0 0000
TIM1SR1	0x216	T1BIF	T1TIF	—	T1CC4IF	T1CC3IF	T1CC2IF	T1CC1IF	T1UIF	00-0 0000
TIM1SR2	0x217	—	—	—	T1CC4OF	T1CC3OF	T1CC2OF	T1CC1OF	—	---0 000-
TIM1EGR	0x218	T1BG	—	—	T1CC4G	T1CC3G	T1CC2G	T1CC1G	—	0--0 000-
TIM1CCMR1 (output mode)	0x219	—	T1OC1M[2:0]			T1OC1PE	—	T1CC1S[1:0]		-000 0-00
TIM1CCMR1 (input mode)		T1IC1F[3:0]			T1IC1PSC[1:0]		T1CC1S[1:0]		0000 0000	
TIM1 CCMR2 (output mode)	0x21A	—	T1OC2M[2:0]			T1OC2PE	—	T1CC2S[1:0]		-000 0-00
TIM1CCMR2 (input mode)		T1IC2F[3:0]			T1IC2PSC[1:0]		T1CC2S[1:0]		0000 0000	
TIM1CCMR3 (output mode)	0x21B	—	T1OC3M[2:0]			T1OC3PE	—	T1CC3S[1:0]		-000 0-00
TIM1CCMR3 (input mode)		T1IC3F[3:0]			T1IC3PSC[1:0]		T1CC3S[1:0]		0000 0000	
TIM1CCMR4 (output mode)	0x21C	—	T1OC4M[2:0]			T1OC4PE	—	T1CC4S[1:0]		-000 0-00
TIM1CCMR4 (input mode)		T1IC4F[3:0]			T1IC4PSC[1:0]		T1CC4S[1:0]		0000 0000	
TIM1CCER1	0x21D	T1CC2NP	T1CC2NE	T1CC2P	T1CC2E	T1CC1NP	T1CC1NE	T1CC1P	T1CC1E	0000 0000
TIM1CCER2	0x21E	—	—	T1CC4P	T1CC4E	T1CC3NP	T1CC3NE	T1CC3P	T1CC3E	--00 0000
TIM1CNTRH	0x28C	T1CNT[15:8]								0000 0000
TIM1CNTRL	0x28D	T1CNT[7:0]								0000 0000
TIM1PSCRH	0x28E	T1PSC[15:8]								0000 0000
TIM1PSCRL	0x28F	T1PSC[7:0]								0000 0000
TIM1ARRH	0x290	T1ARR[15:8]								1111 1111
TIM1ARRL	0x291	T1ARR[7:0]								1111 1111
TIM1RCR	0x292	T1REP[7:0]								0000 0000
TIM1CCR1H	0x293	T1CCR1[15:8]								0000 0000
TIM1CCR1L	0x294	T1CCR1[7:0]								0000 0000
TIM1CCR2H	0x295	T1CCR2[15:8]								0000 0000
TIM1CCR2L	0x296	T1CCR2[7:0]								0000 0000
TIM1CCR3H	0x297	T1CCR3[15:8]								0000 0000
TIM1CCR3L	0x298	T1CCR3[7:0]								0000 0000
TIM1CCR4H	0x299	T1CCR4[15:8]								0000 0000
TIM1CCR4L	0x29A	T1CCR4[7:0]								0000 0000
TIM1BKR	0x29B	T1MOE	T1AOE	T1BKP	T1BKE	T1OSSR	T1OSSI	T1LOCK[1:0]		0000 0000
TIM1DTR	0x29C	T1DTG[7:0]								0000 0000
TIM1OISR	0x29D	—	T1OIS4	T1OIS3N	T1OIS3	T1OIS2N	T1OIS2	T1OIS1N	T1OIS1	-000 0000
LEBCON	0x41C	LEBEN	LEBCH[1:0]		—	EDGS	BKS[2:0]			000- 0000

表 1-1 Timer1 相关用户寄存器汇总

注意：TIM1 寄存器中的保留位必须保持为复位值，不能更改，否则可能出现预想不到的情况

名称	状态		寄存器	地址	复位值
T1CNT	TIM1 计数值	高 8 位	TIM1CNTRH[7:0]	0x28C	RW-0000 0000
		低 8 位	TIM1CNTRL[7:0]	0x28D	RW-0000 0000
T1PSC	TIM1 预分频器	高 8 位	TIM1PSCRH[7:0]	0x28E	RW-0000 0000
		低 8 位	TIM1PSCRL[7:0]	0x28F	RW-0000 0000
T1ARR	输出比较模式：PWM 周期的自动重载寄存器 (预装载值) 注：此值为 0 时，计数器不工作；	高 8 位	TIM1ARRH[7:0]	0x290	RW-1111 1111
		低 8 位	TIM1ARRL[7:0]	0x291	RW-1111 1111
T1REP	重复向下计数值		TIM1RCR[7:0]	0x292	RW-0000 0000
T1CCR1	输出比较模式：TIM1_CH1 占空比 (预装载值)	高 8 位	TIM1CCR1H[7:0]	0x293	RW-0000 0000
		低 8 位	TIM1CCR1L[7:0]	0x294	RW-0000 0000
	输入捕获模式：上一次捕获事件(IC1) 捕获的计数值	高 8 位	TIM1CCR1L[7:0]	0x293	RO-0000 0000
		低 8 位	TIM1CCR1L[7:0]	0x294	RO-0000 0000
T1CCR2	输出比较模式：TIM1_CH2 占空比 (预装载值)	高 8 位	TIM1CCR2H[7:0]	0x295	RW-0000 0000
		低 8 位	TIM1CCR2L[7:0]	0x296	RW-0000 0000
	输入捕获模式：上一次捕获事件 (IC2) 捕获的计数值	高 8 位	TIM1CCR2H[7:0]	0x295	RO-0000 0000
		低 8 位	TIM1CCR2L[7:0]	0x296	RO-0000 0000
T1CCR3	输出比较模式：TIM1_CH3 占空比 (预装载值)	高 8 位	TIM1CCR3H[7:0]	0x297	RW-0000 0000
		低 8 位	TIM1CCR3L[7:0]	0x298	RW-0000 0000
	输入捕获模式：上一次捕获事件 (IC3) 捕获的计数值	高 8 位	TIM1CCR3H[7:0]	0x297	RO-0000 0000
		低 8 位	TIM1CCR3L[7:0]	0x298	RO-0000 0000
T1CCR4	输出比较模式：TIM1_CH4 占空比 (预装载值)	高 8 位	TIM1CCR4H[7:0]	0x299	RW-0000 0000
		低 8 位	TIM1CCR4L[7:0]	0x29A	RW-0000 0000
	输入捕获模式：上一次捕获事件 (IC4) 捕获的计数值	高 8 位	TIM1CCR4H[7:0]	0x299	RO-0000 0000
		低 8 位	TIM1CCR4L[7:0]	0x29A	RO-0000 0000

表 1-2 Timer1 周期相关寄存器

名称	状态		寄存器	地址	复位值
TIM1EN	<u>TIM1 模块时钟</u>	1 = 使能 0 = 关闭	PCKEN[1]	0x9A	RW-0
SYSON	<u>睡眠模式下，系统时钟控制</u>	1 = 使能 0 = 关闭	CKOCON[7]	0x95	RW-0
T1CKSRC	<u>TIM1 时钟源 (Fmaster)</u> 000 = Sysclk 100 = 2x (XT or EC) (*) 001 = HIRC 101 = LIRC 010 = XT or EC (*) 110 = LP or EC (*) 011 = 2x HIRC 111 = 2x (LP or EC) (*) (*) FOSC 需相应配置成 LP/XT/EC 模式或 INTOSCIO 模式，否则振荡器将不会运行。		TCKSRC[2:0]	0x31F	RW-000
T1ARPE	<u>PWM 周期的自动预装载</u> 1 = 使能 (T1ARR 预装载值在更新事件到来时被加载) 0 = 禁止 (T1ARR 立即被加载)		TIM1CR1[7]	0x211	RW-0
T1CMS	<u>计数器对齐模式</u> 00 = 边沿对齐模式 (计数方向由 T1DIR 决定) 01 = 中央对齐模式 1 (向下计数时 T1CCxIF 置 1) 10 = 中央对齐模式 2 (向上计数时 T1CCxIF 置 1) 11 = 中央对齐模式 3 (向上和向下计数时 T1CCxIF 均置 1) 注： 1. 中央对齐模式为计数器交替地向上和向下计数。 2. 计数器关闭后(T1CEN=0)，才允许切换模式。		TIM1CR1[6:5]		RW-00
T1DIR	<u>计数器计数方向</u> (中央对齐模式时，该位只读) 1 = 向下 0 = 向上		TIM1CR1[4]		RW-0
T1OPM	<u>单脉冲模式</u> 1 = 使能 (下一次更新事件到来时，T1CEN 自动清零，计数器停止) 0 = 关闭 (发生更新事件时，计数器不停止)		TIM1CR1[3]		RW-0
T1URS	<u>当 T1UDIS = 0 时，更新事件中断源</u> 1 = 计数器上溢/下溢 0 = 计数器上溢/下溢，或复位触发事件		TIM1CR1[2]		RW-0
T1UDIS	<u>产生更新事件控制</u>		TIM1CR1[1]		RW-0

名称	状态	寄存器	地址	复位值
	1 = 禁止 0 = 允许			
T1CEN	<u>TIM1 计数器</u> 1 = 使能 (此时门控模式才能工作) 0 = 关闭	TIM1CR1[0]		RW-0
T1TS	<u>同步计数器的触发输入源</u> 0xx = 保留 100 = TI1 的边沿检测器 (TI1F_ED) 101 = 滤波后的定时器输入 1 (TI1FP1) 110 = 滤波后的定时器输入 2 (TI2FP2) 111 = 禁止配置 注: 1. 仅在 T1SMS = 000 时可更改这些位; 2. 通道 3/4 的输入只能用作对应通道的捕捉源, 而不能作为触发源;	TIM1SMCR[6:4]		RW-000
T1SMS	<u>触发模式</u> 000 = <u>内部时钟</u> 100 = 复位模式 (在触发输入有效沿时重新初始化计数器和预分频器) 101 = 门控模式 (计数器在触发输入有效电平期间计数, 无效电平则停止计数, 但不复位) 110 = 触发模式 (计数器在触发输入有效沿时计数, 且不复位) 其它 = 保留 注: 1. 门控模式不能选择 TI1F_ED 触发输入; 2. 触发输入有效沿的极性参阅 T1CC1P/T1CC2P;	TIM1SMCR[2:0]	0x213	RW-000

表 1-3 Timer1 相关用户控制寄存器

名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	Bit0	复位值
TIM1CCMR1	0x219	T1IC1F[3:0]				T1IC1PSC[1:0]		T1CC1S[1:0]		RW-0000 0000
TIM1CCMR2	0x21A	T1IC2F[3:0]				T1IC2PSC[1:0]		T1CC2S[1:0]		RW-0000 0000
TIM1CCMR3	0x21B	T1IC3F[3:0]				T1IC3PSC[1:0]		T1CC3S[1:0]		RW-0000 0000
TIM1CCMR4	0x21C	T1IC4F[3:0]				T1IC4PSC[1:0]		T1CC4S[1:0]		RW-0000 0000

名称	状态			寄存器	地址	复位值
T1ICxF	通道 x 输入捕获采样频率和数字滤波器长度			TIM1CCMRx[6:4] x = 1, 2, 3, 4	0x219/ 0x21A/ 0x21B/ 0x21C	RW-0000
	Value	采样频率 (f _{SAMPLING})	数字滤波器 长度 N			
	0000	Fmaster	0			
	0001	Fmaster	2			
	0010	Fmaster	4			
	0011	Fmaster	8			
	0100	Fmaster / 2	6			
	0101	Fmaster / 2	8			
	0110	Fmaster / 4	6			
	0111	Fmaster / 4	8			
	1000	Fmaster / 8	6			
	1001	Fmaster / 8	8			
	1010	Fmaster / 16	5			
	1011	Fmaster / 16	6			
	1100	Fmaster / 16	8			
	1101	Fmaster / 32	5			
	1110	Fmaster / 32	6			
1111	Fmaster / 32	8				
T1ICxPSC	通道 x 输入捕获预分频器 (几个事件触发一次捕获) 00 = 1 个 10 = 4 个 01 = 2 个 11 = 8 个 注: 当 T1CCxE = 0 时, 该预分频器复位为 00			TIM1CCMRx[3:2]		RW-00
T1CC1S ¹	通道 1 模式选择	00 = 输出 01 = 输入, 输入脚映射在 TI1FP1 10 = 输入, 输入脚映射在 TI2FP1 11 = 保留		TIM1CCMR1[1:0]	0x219	RW-00

¹ 仅在通道 x 关闭 (T1CCxE = 0 和 T1CCxNE = 0) 时可写。

T1CC2S ²	通道 2 模式选择	00 = 输出 01 = 输入, 输入脚映射在 TI2FP2 10 = 输入, 输入脚映射在 TI1FP2 11 = 保留	TIM1CCMR2[1:0]	0x21A	RW-00
T1CC3S ²	通道 3 模式选择	00 = 输出 01 = 输入, 输入脚映射在 TI3FP3 10 = 输入, 输入脚映射在 TI4FP3 11 = 保留	TIM1CCMR3[1:0]	0x21B	RW-00
T1CC4S ²	通道 4 模式选择	00 = 输出 01 = 输入, 输入脚映射在 TI3FP4 10 = 输入, 输入脚映射在 TI4FP4 11 = 保留	TIM1CCMR4[1:0]	0x21C	RW-00

表 1-4 TIM1CCMRx 作为输入配置寄存器

名称	地址	bit7	bit6	bit5	bit4	bit3	bit2	bit1	Bit0	复位值
TIM1CCMR1	0x219	-	T1OC1M[2:0]			T1OC1PE	-	T1CC1S[1:0]		RW--000 0-00
TIM1CCMR2	0x21A	-	T1OC2M[2:0]			T1OC2PE	-	T1CC2S[1:0]		RW--000 0-00
TIM1CCMR3	0x21B	-	T1OC3M[2:0]			T1OC3PE	-	T1CC3S[1:0]		RW--000 0-00
TIM1CCMR4	0x21C	-	T1OC4M[2:0]			T1OC4PE	-	T1CC4S[1:0]		RW--000 0-00

T1OCxM	通道 x 输出比较模式		OCxREF (输出参考信号)
000	冻结 (不比较)		禁止
001	当 TIM1_CNT = CCRx_SHAD 时		1
010	当 TIM1_CNT = CCRx_SHAD 时		0
011	当 TIM1_CNT = CCRx_SHAD 时		电平翻转
100	强制为无效电平		0
101	强制为有效电平		1
110	PWM1 模式	TIM1_CNT < CCRx_SHAD	1
		TIM1_CNT > CCRx_SHAD	0
111	PWM2 模式	TIM1_CNT < CCRx_SHAD	0
		TIM1_CNT > CCRx_SHAD	1

1. OCxREF 与 T1CCxP 共同决定输出引脚 OCx 的值;
2. PWM 模式下比较结果改变时, 或输出比较模式下从冻结模式切换到 PWM 模式时, OCxREF 电平才会改变;

表 1-5 T1OCxM 配置为输出比较模式

² 仅在通道 x 关闭 (T1CCxE = 0 和 T1CCxNE = 0) 时可写。

名称	状态		寄存器	地址	复位值
T1OCxPE	<u>通道 x 输出比较占空比的自动预装载</u> 1 = 使能 (T1CCRx 预装载值在更新事件到来时被加载) 0 = <u>禁止</u> (T1CCRx 立即被加载) 注: PWM 模式下必须使能, 单脉冲模式可选		TIM1CCMRx[3] x = 1, 2, 3, 4	0x219/ 0x21A/ 0x21B/ 0x21C	RW-0
T1CC1S ³	<u>通道 1 模式选择</u>	00 = <u>输出</u> 01 = 输入, 输入脚映射在 TI1FP1 10 = 输入, 输入脚映射在 TI2FP1 11 = 保留	TIM1CCMR1[1:0]	0x219	RW-00
T1CC2S ³	<u>通道 2 模式选择</u>	00 = <u>输出</u> 01 = 输入, 输入脚映射在 TI2FP2 10 = 输入, 输入脚映射在 TI1FP2 11 = 保留	TIM1CCMR2[1:0]	0x21A	RW-00
T1CC3S ³	<u>通道 3 模式选择</u>	00 = <u>输出</u> 01 = 输入, 输入脚映射在 TI3FP3 10 = 输入, 输入脚映射在 TI4FP3 11 = 保留	TIM1CCMR3[1:0]	0x21B	RW-00
T1CC4S ³	<u>通道 4 模式选择</u>	00 = <u>输出</u> 01 = 输入, 输入脚映射在 TI3FP4 10 = 输入, 输入脚映射在 TI4FP4 11 = 保留	TIM1CCMR4[1:0]	0x21C	RW-00

表 1-6 TIM1CCMRx 作为输出配置寄存器

³ 仅在通道 x 关闭 (T1CCxE = 0 和 T1CCxNE = 0) 时可写。

名称	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	地址	复位值
TIM1CCER1	T1CC2NP	T1CC2NE	T1CC2P	T1CC2E	T1CC1NP	T1CC1NE	T1CC1P	T1CC1E	0x21D	RW-0000 0000
TIM1CCER2	-	-	T1CC4P	T1CC4E	T1CC3NP	T1CC3NE	T1CC3P	T1CC3E	0x21E	RW--00 0000

名称	功能	输出比较模式	输入捕获/触发模式
T1CCxP	通道 x 引脚 输出极性选择	1 = OCx 低电平有效 0 = <u>OCx 高电平有效</u>	1 = 捕获 / 触发发生在 T1xF 低电平或下降沿 0 = <u>捕获 / 触发发生在 T1xF</u> 高电平或上升沿 注: 输入触发源仅可选通道 1 和 2
T1CCxE	通道 x 引脚 使能	1 = 使能 (OCx 输出到对应的引脚) 0 = <u>禁止</u>	1 = 使能 (捕获计数器的值到 TIM1CCRx 寄存器中) 0 = <u>禁止</u>
T1CCxNP	通道 x 互补引脚 输出极性选择	1 = OCxN 低电平有效 0 = <u>OCxN 高电平有效</u>	-
T1CCxNE	通道 x 互补引脚 使能	1 = 使能 (OCxN 输出到对应的引脚) 0 = <u>关闭</u>	-

注: 通道输出电平由 T1MOE, T1OSSI, T1OSSR, T1OISx, T1OISxN, T1CCxE 和 T1CCxNE 位的值共同决定, 参阅表 1-10。

表 1-7 Timer1 通道输出和极性选择

名称	状态	寄存器	地址	复位值
TIM1_CH1	通道 1 输出重映射	1 = PD1 0 = <u>PA0</u>	AFP0[6]	0x19E RW-0
TIM1_CH1N	通道 1 互补输出重映射	1 = PC7 0 = <u>PC0</u>	AFP0[4]	0x19E RW-0
TIM1_CH2	通道 2 输出重映射	1 = PD2 0 = <u>PA1</u>	AFP1[0]	0x19F RW-0
TIM1_CH2N	通道 2 互补输出重映射	1 = PC6 0 = <u>PA3</u>	AFP0[3]	0x19E RW-0
TIM1_CH3	通道 3 输出重映射	1 = PD3 0 = <u>PB4</u>	AFP1[1]	0x19F RW-0
TIM1_CH3N	通道 3 互补输出重映射	1 = PC5 0 = <u>PB0</u>	AFP0[2]	0x19E RW-0
TIM1_CH4	通道 4 输出重映射	1 = PD5 0 = <u>PB1</u>	AFP1[5]	0x19F RW-0
TIM1_BKIN	故障源刹车输入重映射	1 = PD4 0 = <u>PB3</u>	AFP1[3]	0x19F RW-0

表 1-8 Timer1 管脚功能重映射寄存器

名称	控制				寄存器	地址	复位值
T1MOE ⁴	主输出控制 (仅对配置为输出的通道有效) 1 = 使能 (若 T1CCxE = 1, 则使能 OC 和 OCN 输出) 0 = 禁止 (禁止 OC 和 OCN 输出或强制为空闲状态)				TIM1BKR[7]	0x29B	RW-0
T1AOE	自动输出控制 1 = T1MOE 在下一个更新事件到来时被自动置 1 (当刹车输入无效时) 或由软件置 1 0 = T1MOE 只能由软件置 1				TIM1BKR[6]		RW-0
T1BKP	故障源 TIM1_BKIN 刹车输入极性 1 = 高电平有效 0 = 低电平有效				TIM1BKR[5]		RW-0
T1BKE	刹车输入(BRK)功能 1 = 使能 0 = 禁止				TIM1BKR[4]		RW-0
T1OSSR	运行模式下(当 T1MOE = 1 时)输出“关闭状态”选择 详情请见表 1-10 Timer1 输出控制和状态”				TIM1BKR[3]		RW-0
T1OSSI	空闲模式下(当 T1MOE=0 时)输出“关闭状态”选择 详情请见表 1-10 Timer1 输出控制和状态”				TIM1BKR[2]		RW-0
T1LOCK ⁵	锁定设置 (写保护, 防止软件错误)				TIM1BKR[1:0]		RW-00
	00	01	10	11			
	关闭	锁定级别 1	锁定级别 2	锁定级别 3			
	寄存器无写保护	T1BKE, T1BKP, T1AOE, T1OISx, T1OISxN, T1DTG	包含级别 1, T1CCxP, T1CCxNP, T1OSSR, T1OSSI	包含级别 2, T1OCxM, T1OCxPE			

表 1-9 Timer1 主输出使能、刹车和锁定级别寄存器

⁴ 当刹车输入有效时, 该位将被硬件异步清 0。

⁵ 系统复位后只能写一次 LOCK 位, 一旦写入则其内容保持不变直至复位。

控制位					输出状态	
T1MOE	T1OSSI	T1OSSR	T1CCxE	T1CCxNE	OCx 输出状态	OCxN 输出状态
1	x	0	0	0	OCx = 0 (输出关闭)	OCxN = 0 (输出关闭)
		0	0	1	OCx = 0 (输出关闭)	OCxREF + 极性 OCxN = OCxREF ^ T1CCxNP
		0	1	0	OCxREF + 极性 OCx = OCxREF ^ T1CCxNP	OCxN = 0 (输出关闭)
		0	1	1	OCxREF + 极性 + 死区	OCxREF 的互补信号 + 极性 + 死区
		1	0	0	OCx = T1CCxP (输出关闭)	OCxN = T1CCxNP (输出关闭)
		1	0	1	关闭状态(运行模式下输出使能) OCx = T1CCxP	OCxREF + 极性 OCxN = OCxREF ^ T1CCxNP
		1	1	0	OCxREF + 极性 OCx = OCxREF ^ T1CCxNP	关闭状态(运行模式下输出使能) OCxN = T1CCxNP
		1	1	1	OCxREF + 极性 + 死区	OCxREF 的互补信号 + 极性 + 死区
0	0	x	0	0	OCx = T1CCxP (输出关闭)	OCxN = T1CCxNP (输出关闭)
	0		0	1	(输出关闭) OCx = T1OISx, OCxN = T1OISxN	
	0		1	0		
	0		1	1		
	1		0	0	OCx = T1CCxP (输出关闭)	OCxN = T1CCxNP (输出关闭)
	1		0	1	关闭状态(空闲模式下输出使能) OCx = T1OISx, OCxN = T1OISxN	
	1		1	0		
	1		1	1		

表 1-10 Timer1 输出控制和状态

名称	控制			寄存器	地址	复位值
T1DTG	<u>死区发生器设置</u>			TIM1DTR[7:0]	0x29C	RW-0000 0000
	T1DTG[7:5]	DT(死区持续时间)	t _{DTG}			
	0xx	T1DTG[7:0] x t _{DTG}	T _{Fmaster} (f1)			
	10x	(64+T1DTG[5:0]) x t _{DTG}	2 x T _{Fmaster} (f2)			
	110	(32+T1DTG[4:0]) x t _{DTG}	8 x T _{Fmaster} (f3)			
	111	(32+T1DTG[4:0]) x t _{DTG}	16 x T _{Fmaster} (f4)			
	* T _{Fmaster} 为 TIM1 时钟源 若 T _{Fmaster} =125 ns (8 MHz)时, 死区时间如下:					
	T1DTG[7:0]	死区时间 (μs)	步长时间			
	0 ~ 7Fh	0 ~ 15.875	125 ns (f1)			
	80h ~ BFh	16 ~ 31.75	250 ns (f2)			
C0h ~ DFh	32 ~ 63	1 μs (f3)				
E0h ~ FFh	64 ~ 126	2 μs (f4)				

表 1-11 Timer1 互补输出死区时间配置

名称	状态	寄存器	地址	复位值
T1OIS4	当 T1MOE=0 时, 通道 4(OC4) 空闲状态输出 1 = OC4 输出 1 0 = OC4 输出 0	TIM1OISR[6]	0x29D	RW-0
T1OIS3	当 T1MOE=0 时, 通道 3/2/1(OCx) 空闲状态输出	TIM1OISR[4]		RW-0
T1OIS2	1 = 死区时间后, OCx 输出 1	TIM1OISR[2]		RW-0
T1OIS1	0 = 死区时间后, OCx 输出 0	TIM1OISR[0]		RW-0
T1OIS3N	当 T1MOE=0 时, 互补通道 3/2/1(OCxN) 空闲状态输出	TIM1OISR[5]		RW-0
T1OIS2N	1 = 死区时间后, OCxN 输出 1	TIM1OISR[3]		RW-0
T1OIS1N	0 = 死区时间后, OCxN 输出 0	TIM1OISR[1]		RW-0

表 1-12 Timer1 通道输出空闲状态寄存器

名称	状态	寄存器	地址	复位值
LEBEN	<u>前沿消隐</u> 1 = 使能 0 = 关闭 注：必须在 GO/DONE=0 时进行切换，否则 ADC 结果不确定；	LEBCON[7]	0x41C	RW-0
LEBCH	<u>前沿消隐通道</u> 00 = TIM1_CH1 10 = TIM1_CH3 01 = TIM1_CH2 11 = TIM1_CH4	LEBCON[6:5]		RW-00
EDGS	<u>PWM 消隐沿</u> 1 = PWM 下降沿 0 = PWM 上升沿	LEBCON[3]		RW-0
BKS	<u>TIM1 的故障源</u> 000 = 关闭 001 = BKIN 管脚 010 = LVD 检测 100 = ADC 阈值比较	LEBCON[2:0]		RW-000

表 1-13 LEB 用户寄存器

名称	状态		寄存器	地址	复位值
GIE	<u>全局中断</u> 1 = 使能 (PEIE, T1BIE, T1BG, T1TIE, T1CCxIE, T1CCxG, T1UIE 适用) 0 = 全局关闭 (唤醒不受影响)		INTCON[7]	Bank 首地址 +0x0B	RW-0
PEIE	<u>外设总中断</u> 1 = 使能 (T1BIE, T1BG, T1TIE, T1CCxIE, T1CCxG, T1UIE 适用) 0 = 关闭 (无唤醒)		INTCON[6]		RW-0
T1BIE	刹车中断	1 = 使能	TIM1IER[7]	0x215	RW-0
T1BG ⁶	刹车软件中断	0 = 关闭	TIM1EGR[7]	0x218	WO-0
T1BIF ⁷	<u>刹车中断标志位</u> 1 = 刹车输入上检测到有效电平 0 = 无刹车事件产生		TIM1SR1[7]	0x216	R_W1C-0

⁶ 软件置 1，硬件自动清 0。

⁷ 写 1 清 0，写 0 无效。建议只使用 STR、MOVWI 指令进行写操作，而不要用 BSR 或 IOR 指令。

名称	状态		寄存器	地址	复位值
T1TIE	<u>触发中断</u> 1 = 使能 0 = <u>关闭</u>		TIM1IER[6]	0x215	RW-0
T1TIF ⁸	<u>触发中断标志位</u> 1 = 已触发 0 = <u>无触发事件产生</u>		TIM1SR1[6]	0x216	R_W1C-0
T1CC4IE	通道 4 捕获/比较中断	1 = 使能 0 = <u>关闭</u>	TIM1IER[4]	0x215	RW-0
T1CC3IE	通道 3 捕获/比较中断		TIM1IER[3]		RW-0
T1CC2IE	通道 2 捕获/比较中断		TIM1IER[2]		RW-0
T1CC1IE	通道 1 捕获/比较中断		TIM1IER[1]		RW-0
T1CC4G ⁹	通道 4 捕获/比较软件中断		TIM1EGR[4]	0x218	WO-0
T1CC3G ⁹	通道 3 捕获/比较软件中断		TIM1EGR[3]		WO-0
T1CC2G ⁹	通道 2 捕获/比较软件中断		TIM1EGR[2]		WO-0
T1CC1G ⁹	通道 1 捕获/比较软件中断		TIM1EGR[1]		WO-0
T1CC4IF ⁸	<u>通道 x 捕获/比较中断标志位</u> • 输出模式： 1 = CNT 值与 T1CCR _x 值匹配 0 = <u>不匹配</u> 注：若 T1CCR _x >T1ARR，则当 CNT 计数到 T1ARR 值时，T1CC _x IF 置 1。 • 输入模式： 1 = 计数器值已被捕获至 TIM1CCR 0 = <u>无捕获产生</u>		TIM1SR1[4]	0x216	R_W1C-0
T1CC3IF ⁸			TIM1SR1[3]		R_W1C-0
T1CC2IF ⁸			TIM1SR1[2]		R_W1C-0
T1CC1IF ⁸			TIM1SR1[1]		R_W1C-0
T1CC4OF ⁸	<u>通道 x 重复捕获标志位</u> 1 = 发生重复捕获 (计数器的值被捕获到 TIM1CCR _x 寄存器时，T1CC _x IF 的状态已经为 1) 0 = <u>无重复捕获</u> 注：仅通道配置位捕获输入时有效		TIM1SR2[4]	0x217	R_W1C-0
T1CC3OF ⁸			TIM1SR2[3]		R_W1C-0
T1CC2OF ⁸			TIM1SR2[2]		R_W1C-0
T1CC1OF ⁸			TIM1SR2[1]		R_W1C-0
T1UIE	允许更新中断	1 = 使能 0 = <u>关闭</u>	TIM1IER[0]	0x215	RW-0
T1UIF ⁸	更新中断标志位	1 = 更新事件等待响应 0 = <u>无更新事件</u>	TIM1SR1[0]	0x216	R_W1C-0

表 1-14 Timer1 中断使能和状态位

⁸ 写 1 清 0，写 0 无效。建议只使用 STR、MOVWI 指令进行写操作，而不要用 BSR 或 IOR 指令。

⁹ 软件置 1，硬件自动清 0。

1.3. 功能描述

整个 TIM1 可以分为三个大的功能部分：计数基本单元、计数控制和捕捉比较通道。计数基本单元分为向上/向下计数器、自动加载寄存器、重复计数器和预分频器；计数控制器又分为计数触发源，模式控制；捕捉比较通道分为捕捉输入通道，输出比较通道，死区产生和输出控制。

1.3.1. 计数基本单元

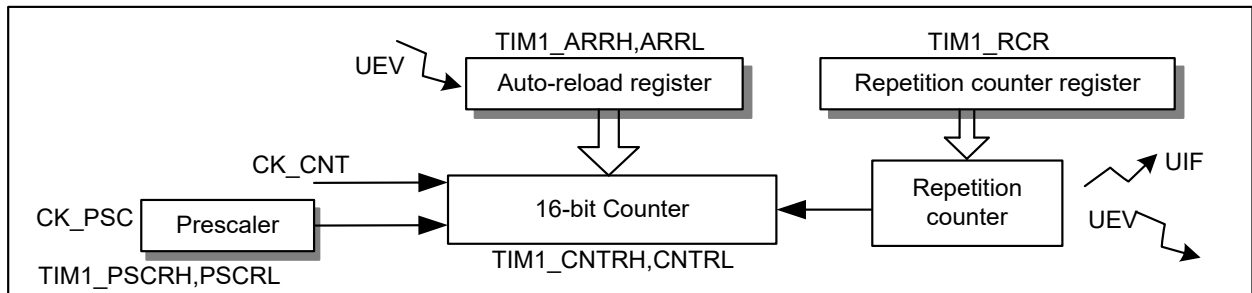


图 1-2 计数基本单元

16 位计数器，预分频器，自动重载寄存器和重复计数寄存器都能由软件进行读写。

1.3.1.1. 计数基本单元组成

1.3.1.1.1. 16 位计数器的读写

- TIM1CNTRH/L 能在任何时候进行写操作；但是建议不要在计数器运行的时候进行写操作，以免出现不正确的中间状态，
- TIM1CNTRH/L 的写操作是没有顺序限制的；可以先写高位也可以先写低位
- TIM1CNTRH/L 能在任何时候进行读操作；但是因为此设计是异步设计，所以在计数器运行期间进行读操作可能读出不正确的数值，需要读两次，比较两次数值是否一致；如果一致，则读出的数值是正确的数值；否则，读出数值是错误的。

1.3.1.1.2. 预分频器

计数时钟可以进行 16bit 的时钟预分频，分频系数为 1~65536。计算计数器时钟分频的公式，如下：

$$f_{CK_CNT} = F_{master} / (PSCR[15:0] + 1) \quad (PSCR \text{ 为实际装入预分频器影子寄存器的值})$$

预分频支持分频自动更新，即在更新事件发生后，能够自动改变预分频值。当 T1CEN 为 0 时，写入预分频寄存器的值也能直接加载实际应用的预分频寄存器中。配置步骤示例如下：

1. 使能 TIM1 模块时钟并选择 TIM1 时钟源
2. 配置计数周期
3. 配置占空比
4. 配置预分频
5. 使能计数器

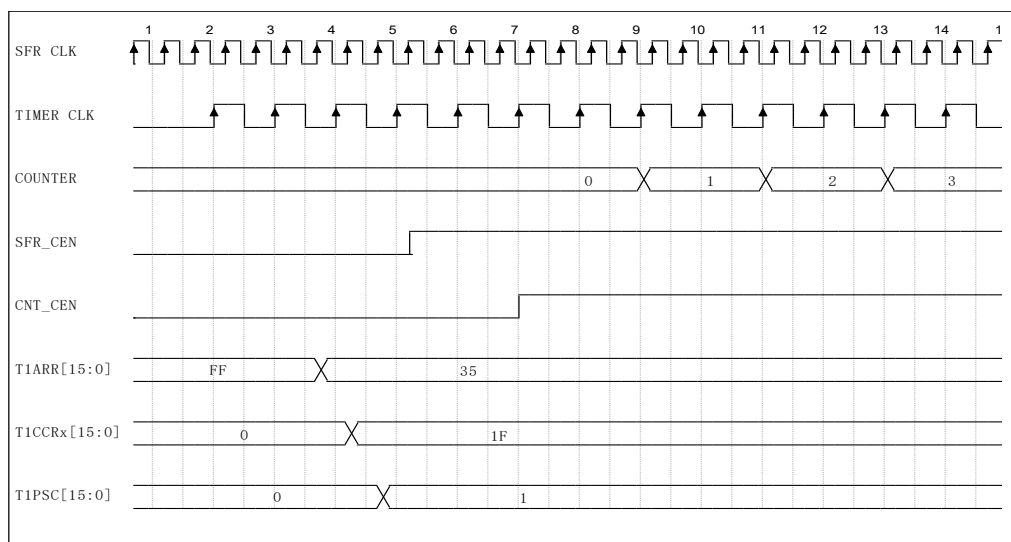


图 1-3 预分频为 1 时的计数器计数时序图

注意：

配置时，需先将周期，占空比，模式等寄存器配置完成后，且在 T1CEN 使能之前，配置预分频寄存器。

1.3.1.1.3. 自动重载寄存器

自动重载寄存器由一个预加载寄存器和一个影子寄存器组成。写自动重载寄存器的三种方式：

- 方式 1：计数器使能位打开，且周期预加载使能 (T1ARPE = 1)。在此模式下，写入自动重载寄存器的值保存在预加载寄存器中，并在下一个更新事件到来时传送到影子寄存器中进行使用。如下图所示：

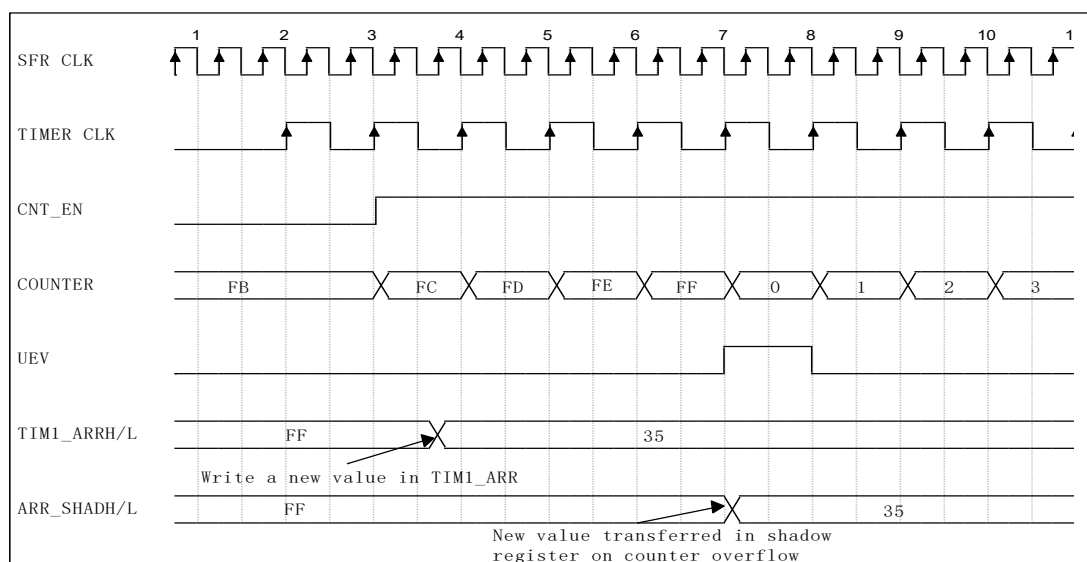


图 1-4 T1CEN = 1 且 T1ARPE = 1，周期寄存器 (T1ARR) 加载图

- 方式 2：计数器使能位打开，且周期预加载关闭 ($T1ARPE=0$)。在此模式下，写入自动重载寄存器的数值直接传送到影子寄存器中进行使用。如下图所示：

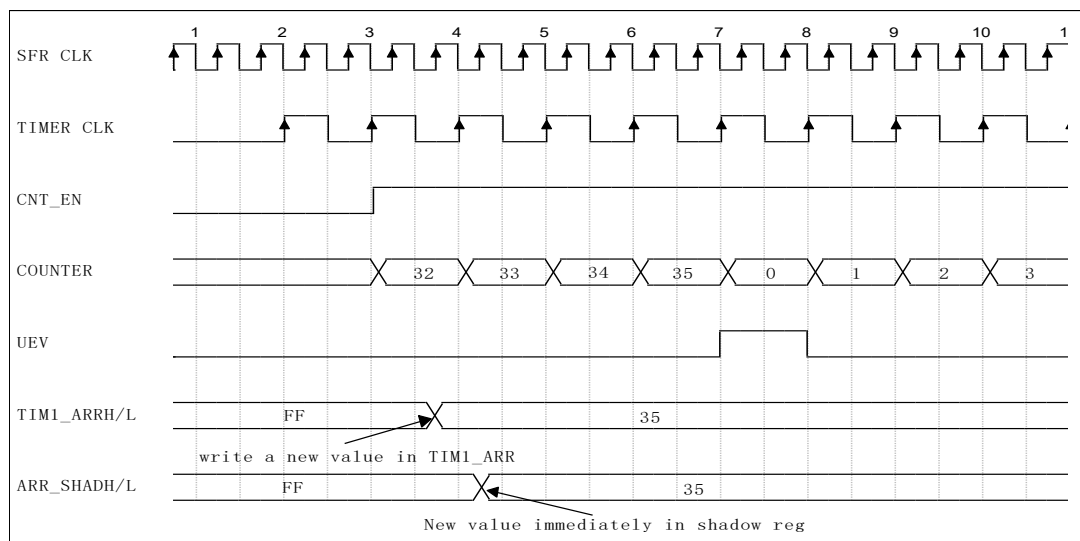


图 1-5 $T1CEN = 1$ 且 $T1ARPE = 0$ ，周期寄存器 (T1ARR) 加载图

- 方式 3：当计数器使能位 ($T1CEN$) 关闭时，不管周期预加载 ($T1ARPE$) 使能还是关闭，写入自动重载寄存器的数值直接传送到影子寄存器中进行使用。如下图所示：

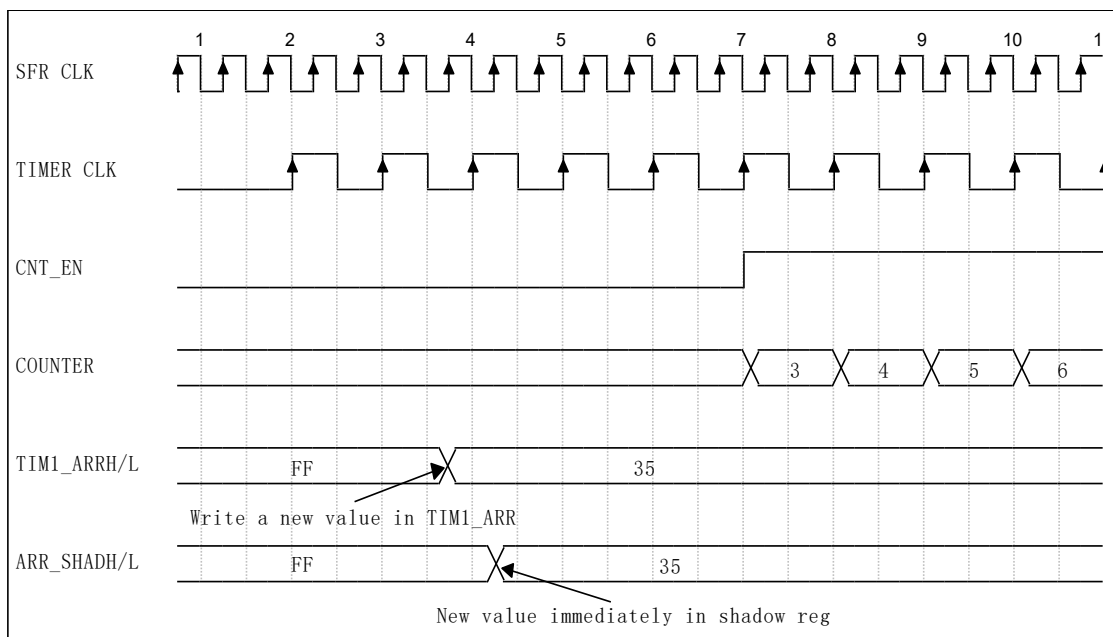


图 1-6 $T1CEN = 0$ ，周期寄存器 (T1ARR) 加载图

1.3.1.1.4. 更新事件

更新事件产生的条件：

- 计数器上溢或下溢
- 配置为复位模式(仅在输入捕捉模式下)时，触发事件的到来

更新事件的影响：

- 影响 1：某些预加载的寄存器(具体寄存器可查看寄存器表格)在预加载使能的情况下都能被更新为最新值。各类预加载寄存器的总结如下表所示：

在更新事件下，可进行预加载的寄存器	TIMARRH/L	TIM1PSCRH/L	TIM1CCRxH/L
相应的预加载使能位	T1ARPE	无使能位,预加载在计数器使能 (T1CEN=1) 时一直有效	T1OCxPE

表 1-15 更新事件相关的预加载寄存器 vs 预加载使能位

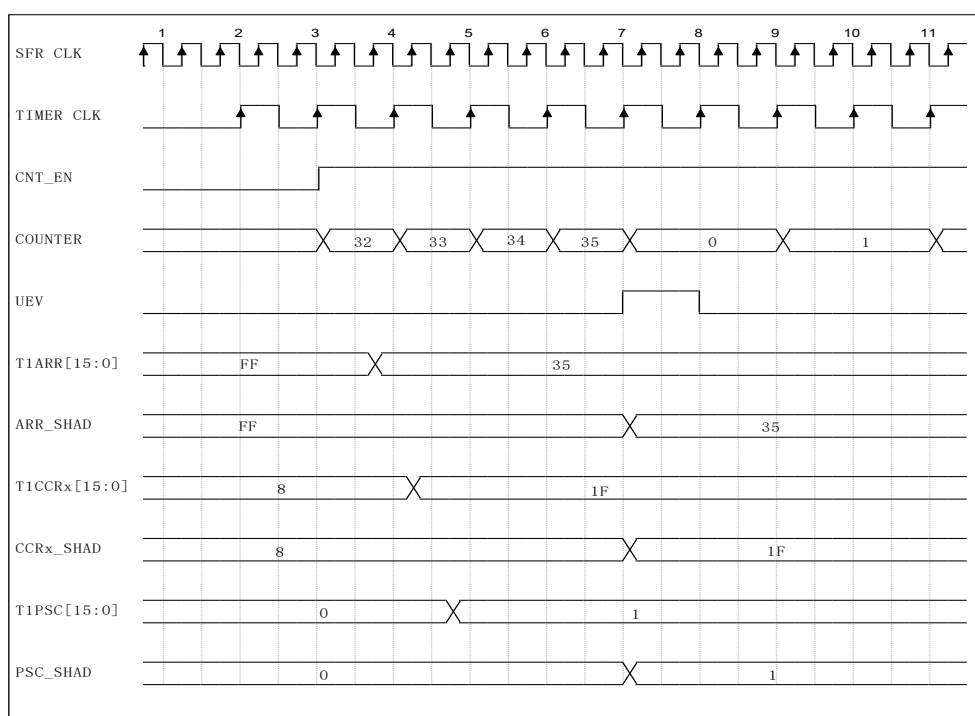


图 1-7 更新事件下，预加载寄存器的更新图

- 影响 2: 若 $T1UDIS = 0$, 当产生更新事件时, 更新标志位($T1UIF$)被置位; 反之, $T1UDIS=1$ 时, 不产生更新事件, 更新标志位 ($T1UIF$) 也不会被置位。如下图所示:

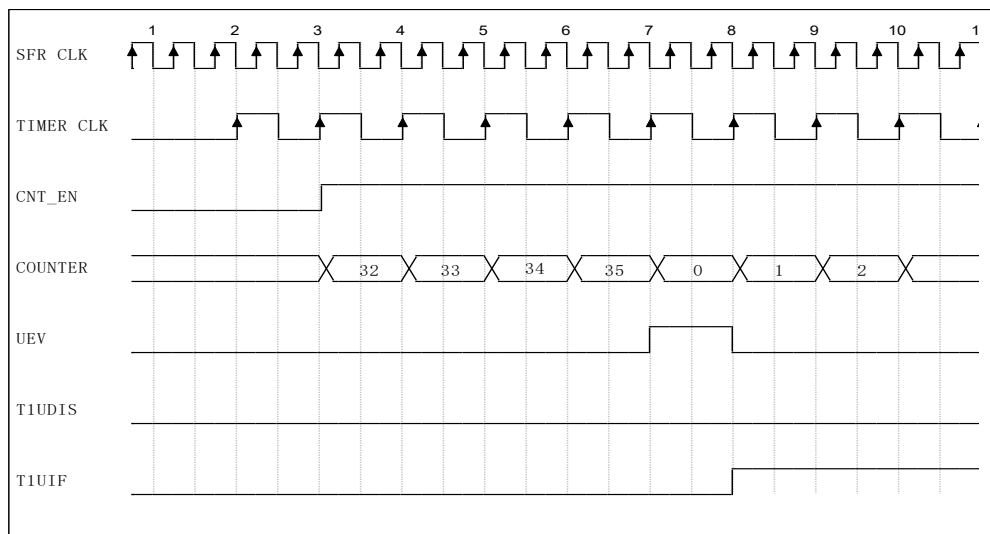


图 1-8 更新事件下且 $T1UDIS = 0$, 更新标志位变化图

- 影响 3: 单次脉冲模式下, 更新事件的到来会使计数器使能位 ($T1CEN$) 关闭, 计数器停止计数。关于单次脉冲模式的详细说明可查看[章节 1.3.3.3](#) 内容。
- 影响 4: 故障事件撤消后, 如果 $T1AOE=1$, PWM 将在更新事件后恢复正常输出。关于故障刹车事件的详细说明可查看[章节 1.3.5](#) 内容。

1.3.1.2. 向上计数模式

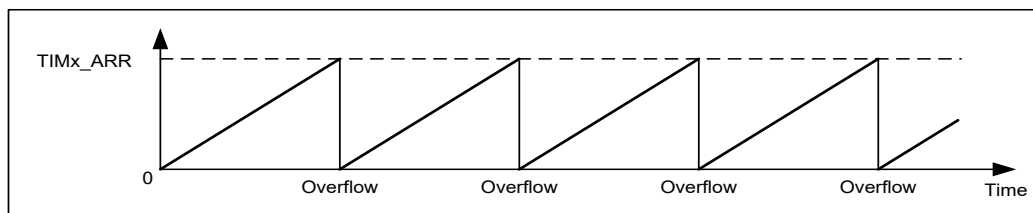


图 1-9 向上计数模式

在向上计数模式中，计数器从 0 开始计数向上计数，计到 TIM1_ARR 寄存器所设数值。然后重新从 0 开始计数并产生一个计数器上溢事件；如果 T1UDIS 设为 0，那么还会产生一个更新事件 UEV。

1.3.1.3. 向下计数模式

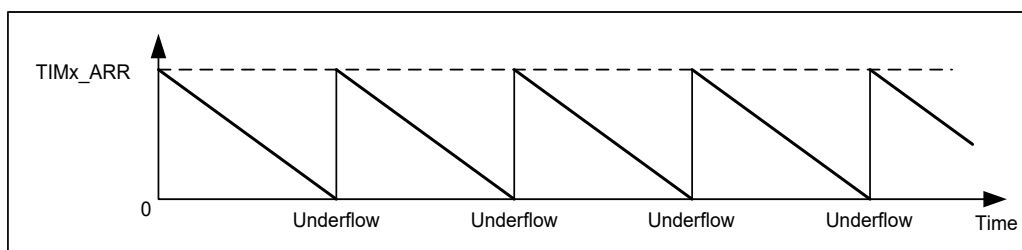


图 1-10 向下计数模式

在向下计数模式中，计数器从 TIM1_ARR 寄存器设置的自动重载值开始向下计数，直到计到 0。然后重新从自动重载值开始计数并产生一个计数器下溢事件；如果 T1UDIS 设为 0，那么还会产生一个更新事件 UEV。

1.3.1.4. 中心对齐模式

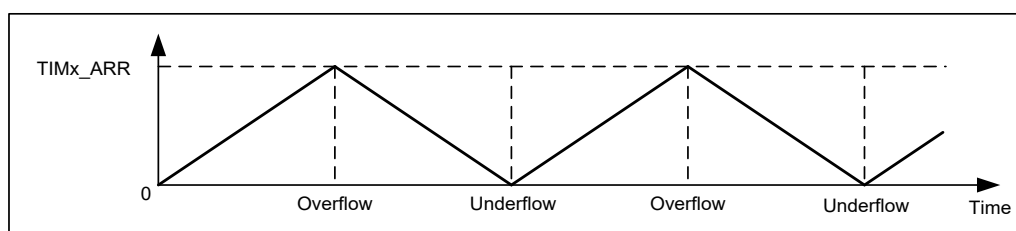


图 1-11 中心对齐模式，T1DIR 初始化为 0

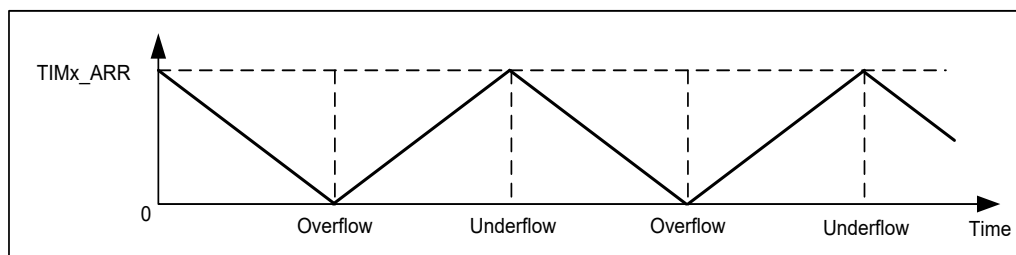


图 1-12 中心对齐模式，T1DIR 初始化为 1

在中心对齐模式中，计数器从 0 开始向上计数，计到自动重载值。此时会产生一个计数器上溢事件。然后计数器开始向下计数计到 0，产生一个下溢事件。计数器不断地重复上述的计数过程。

中心对齐模式所需注意事项：

- 方向位(T1DIR)不能进行写操作。方向位会由硬件设置成当前计数器的计数方向
- 当在中心对齐模式下开始计数时,当前的配置会被使用 – 计数开始值为写入 TIM1CNTRH/L 中的值,计数开始方向决定于写入 TIM1CR1 寄存器中的 T1DIR 位。注意 T1DIR 位和 T1CMS 值不能被软件同时改写。
- T1DIR 位在 T1CMS \neq 00 时,为只读寄存器,无法进行写操作;所以如果想要配置计数器的初始计数方向,需要先配置计数方向(T1DIR),再配置计数模式(T1CMS)。
- 运行在中心对齐模式下时,不建议写计数器值(TIM1CNTRH/L),可能产生意想不到的结果。
如果写入计数器的值大于自动加载值(TIM1_CNT > T1ARR),计数方向可能不会进行更新。
如果写入计数器的值为 0 或为 T1ARR,计数方向会进行更新但更新事件(UEV)不会产生。

配置步骤示例：

1. 使能 TIM1 模块时钟, 并选择 TIM1 时钟源
2. 配置计数周期预加载使能(T1ARPE=1)
3. 配置计数周期(T1ARR=06H), 占空比
4. 配置初始计数方向为向上计数(T1DIR=0)
5. 配置计数模式为中央对齐模式 1(T1CMS=01)
6. 配置预分频(T1PSC=0)
7. 使能计数器

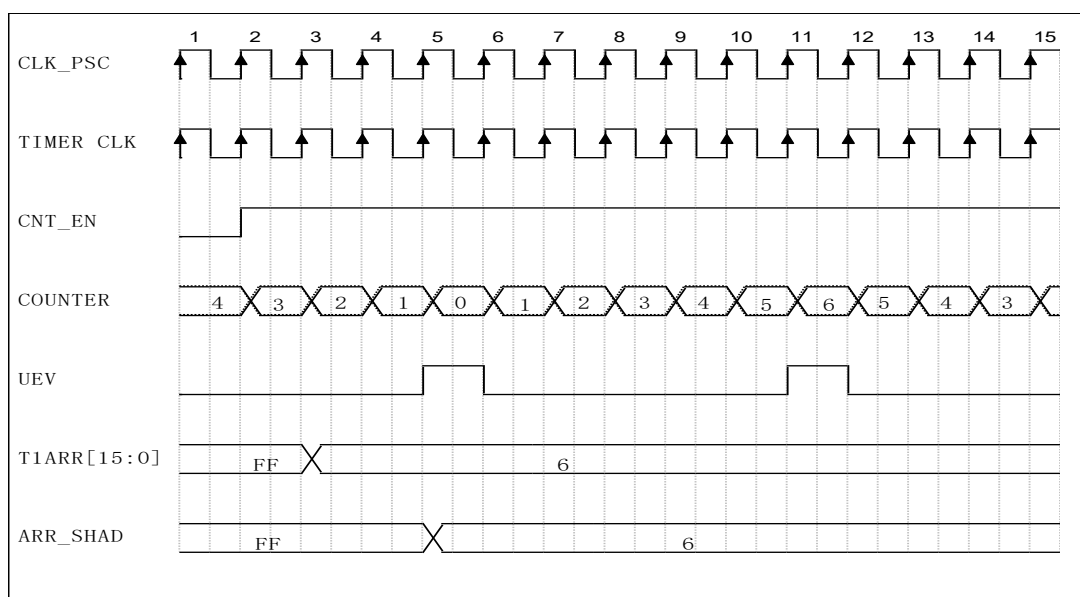


图 1-13 中心对齐模式下，计数时序图

1.3.1.5. 重复向下计数器

重复计数器是 8bit 的向下计数器，会在每次 TIMER 上溢或下溢时-1；只有当重复向下计数器减到 0 时，计数器上溢或下溢才会产生更新事件(UEV)；使用重复计数器能够设定更新事件的频率，这在产生特定数量 PWM 信号时非常有用，如下图所示。

重复向下计数器自减事件：

- 计数器向上计数模式下，每个计数上溢事件都会使重复计数器减 1。
- 计数器向下计数模式下，每个计数下溢事件都会使重复计数器减 1。
- 计数器中心对齐模式下，每个计数上溢或下溢事件都会使重复计数器减 1。

重复向下计数器是自动重载的，当发生了更新事件(UEV)时，会将 TIM1RCR 寄存器中的值自动重载到重复向下计数器中。如下图所示：

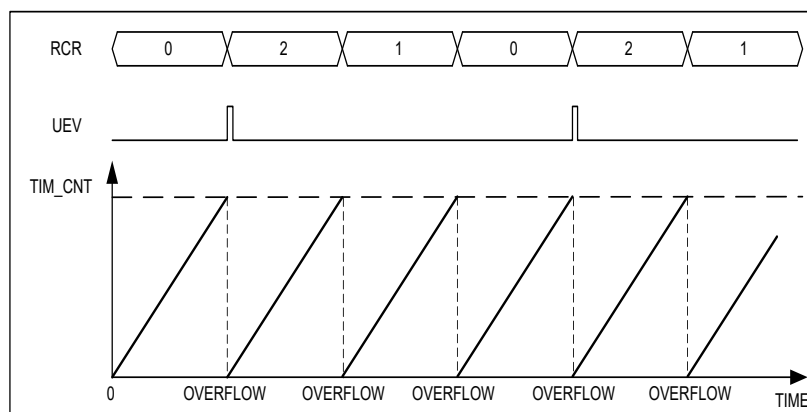


图 1-14 T1REP = 2，重复计数器计数时序图

配置产生特定个数 PWM 信号的步骤示例：

1. 使能 TIM1 模块时钟并选择 TIM1 时钟源
2. 配置 TIM1 通道相对应的端口为输出端口
3. 开启更新事件中断
4. 配置计数周期(T1ARR)，占空比(T1CCR_x)
5. 需要开启周期预加载(T1ARPE)和占空比预加载功能(T1OCxPE)
6. 配置计数方向为向上计数(T1DIR=0)
7. 配置输出比较模式(T1OCxM=3'b111)为 PWM2 输出模式，并配置通道使能
8. 打开自动主输出使能(T1AOE=1)位
9. 使能计数器
10. 在更新事件中断中，重新更改计数周期，占空比等配置

以下是一段示例代码：

```
BANKSEL    PCKEN    ;
BSR        PCKEN,0  ; 使能 TIM1 模块时钟

BANKSEL    INTCON    ;
LDWI      H'00'      ;
```



```

STR          INTCON          ; 开启全局中断使能和外设中断使能
BANKSEL      TCKSRC          ;
LDWI         H'01'           ;
STR          TCKSRC          ; 选择 TIM1 时钟源为 HIRC
BANKSEL      TRISA           ;
LDWI         H'FE'           ;
STR          TRISA           ; 配置 PA0 为通道 1 的输出通道
BANKSEL      TIM1ARRL        ;
LDWI         H'1F'           ;
STR          TIM1ARRL        ; 将输出波形周期配置为 32
LDWI         H'10'           ;
STR          TIM1CCR1L        ; 将输出波形占空比配置为 16
LDWI         H'02'           ;
STR          TIM1RCR          ; 将重复计数器配置为 2
BSR          TIM1BKR,6       ; 打开自动主输出使能位
BANKSEL      TIM1CCMR1       ;
LDWI         H'70'           ;
STR          TIM1CCMR1       ; 配置通道 1 为 PWM2 模式输出
BSR          TIM1IER,0       ; 开启更新事件中断
LDWI         H'01'           ;
STR          TIM1CCER1        ; 使能通道 1
BANKSEL      TIM1CR1         ;
LDWI         H'81'           ; 开启计数器计数使能位
STR          TIM1CR1          ; 开启计数器计数使能位和周期预加载使能位

INT:
BANKSEL      TIM1ARRL        ;
LDWI         H'14'           ;
STR          TIM1ARRL        ; 将输出波形周期配置为 20

```

上述示例对应示意图：

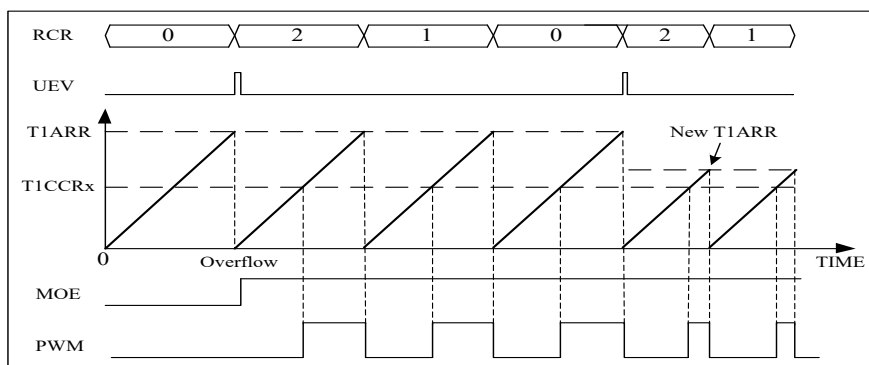


图 1-15 利用重复计数器输出 3 个特定的 PWM 的时序图

注意：

由于重复计数器只有在周期更新事件 (UEV) 发生时才重载 T1REP 值，对 TIM1_RCR 寄存器写入的新值只在下次周期更新事件发生时才起作用，所以建议当配置 T1REP 不为 0 时，在第一个更新事件 (计数器上溢或下溢) 之后再打开更新事件中断。

1.3.2. 计数控制器

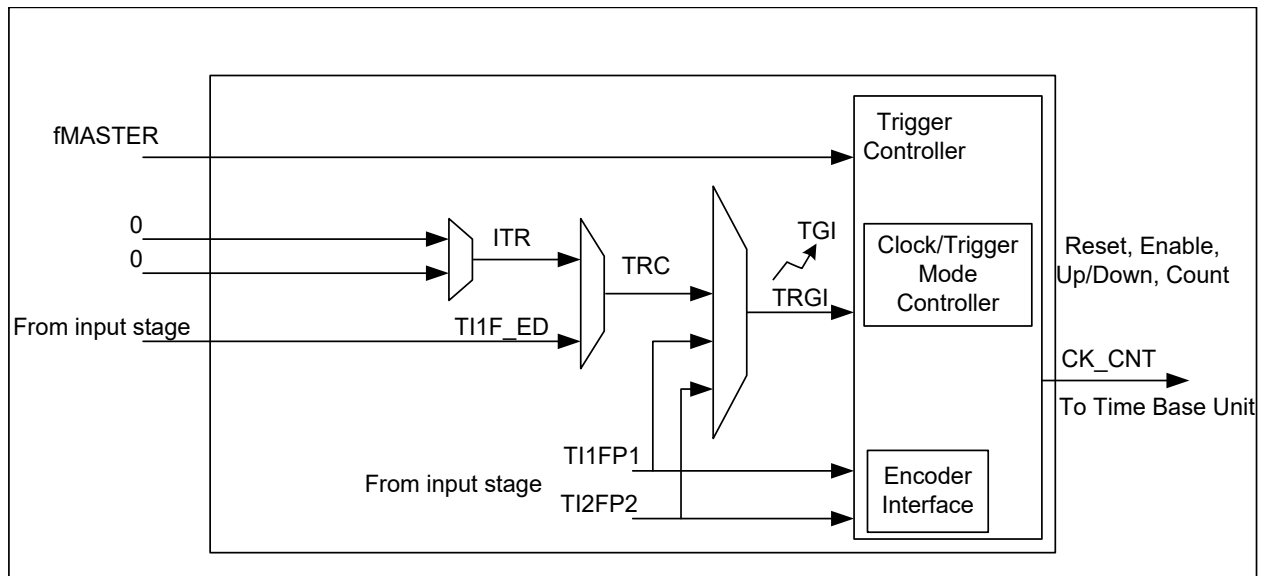


图 1-16 时钟/触发控制器框图

时钟/触发控制器允许配置各种计数器时钟源，输入触发和输出触发。

1.3.2.1. 计数器时钟源

计数器的计数时钟 (CK_CNT) 可由 TCKSRC 寄存器进行选择，总共有以下 8 种时钟源：

- 系统时钟/主时钟
- HIRC
- XT 时钟/外部时钟
- HIRC 的 2 倍频
- XT 时钟/外部时钟的 2 倍频
- LIRC
- LP 时钟/外部时钟
- LP 时钟/外部时钟的 2 倍频

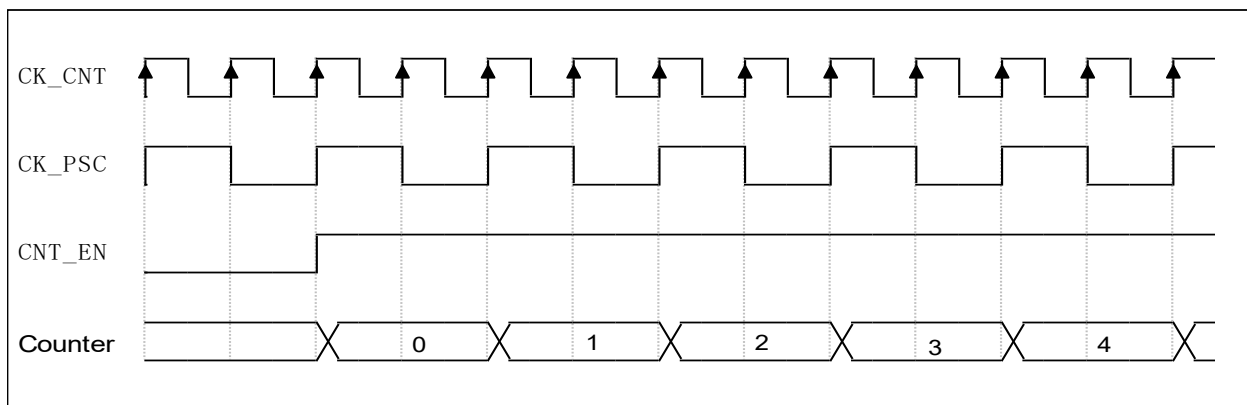


图 1-17 预分频为 1 时，计数器计数时序图

1.3.2.2. 计数触发源

计数触发源只能来自通道 1/2 对应的输入端口，来源与通道 1/2 的捕捉源的来源一致。通道 3/4 的输入只能作为对应的通道捕捉源。关于捕捉源的详细描述可查看[章节 1.3.3.1](#) 内容。

当 T1SMS = 000 时，计数由内部时钟驱动，使能 T1CEN 即可触发计数。

仅当 T1SMS = 101 / 110 时 (Slave 模式下)，计数触发源有效，T1TS[2:0] 位 (TIM1CR1 寄存器) 提供可选的 3 种触发源如下：

- 输入源 T1 的边沿检测 (TI1F_ED)；
- 滤波后的通道 1 输入 (TI1FP1)；
- 滤波后的通道 2 输入 (FI2FP2)；

1.3.2.3. 计数控制模式选择

TIM1 除了有向上计数、向下计数、中央对齐计数方式之外，还有 4 种计数控制模式，需要计数触发源的配合使用；计数模式的选择由 TIM1SMCR 寄存器中的 T1SMS[2:0] 去控制，下列是 4 种计数模式：

1. 内部时钟模式：

计数由内部时钟 (CK_CNT) 驱动。

2. 复位模式 (必须在输入捕捉模式下才能配置为此模式)：

在选中的触发输入 (TRGI) 的上升沿时重新初始化计数器，并且产生一个更新寄存器的信号。

3. 门控模式 (T1SMS=3'b101)：

当触发输入 (TRGI) 为高时，计数器的时钟开启。一旦触发输入变为低，则计数器停止(但不复位)。计数器的启动和停止都是受控的。

4. 触发模式 (T1SMS=3'b110)：

计数器在触发输入 TRGI 的上升沿启动 (但不复位)，只有计数器的启动是受控的。

- 内部时钟模式：

内部时钟 (CK_CNT) 模式下，在软件配置计数器使能 (T1CEN) 之后，计数器开始由内部时钟 (CK_CNT) 驱动下进行计数；如图 1-17 所示。

- 复位模式：

当触发输入事件到来时，计数器和计数器预分频都会被初始化。如果此时 T1URS 为 0 且 T1UDIS 也为 0，则会产生一个更新事件，同时所有的预加载寄存器都会被更新。

复位模式的步骤示例：

1. 配置输入捕捉寄存器的值 – 配置输入捕捉滤波器 T1IC1F=000；配置捕捉预分频器 T1IC1PSC=0
2. 将通道配置为输入捕捉通道 T1CC1S=01，并将 IC1 映射在 TI1FP1 上
3. 写 T1CC1P=0，选择检测触发上升沿的到来
4. 通过写 T1SMS=100，将 TIM1 配置为复位模式。同时写 T1TS=101，选择 TI1 为输入触发源
5. 置位 T1CEN，启动计数器

以下是一段示例代码：

```

BANKSEL    PCKEN
BSR        PCKEN,0      ; 使能 TIM1 模块时钟

BANKSEL    TCKSRC
LDWI      H'01'
STR       TCKSRC        ; 选择 TIM1 时钟源为 HIRC

BANKSEL    TRISA
LDWI      H'FF'
STR       TRISA         ; PA0 为通道 1 的输入通道

BANKSEL    TIM1CCMR1
LDWI      H'01'
STR       TIM1CCMR1     ; 配置通道 1 的 IC1 映射在 TI1FP1 上

LDWI      H'54'
STR       TIM1SMCR      ; 配置 TIM1 为复位模式，触发源为 TI1FP1
LDWI      H'01'
STR       TIM1CCER1     ; 使能通道 1 并且为上升沿触发

BANKSEL    TIM1CR1
BSR        TIM1CR1,0    ; 开启计数器计数使能位

```

当 TI1 的上升沿到来时，计数器被清 0 并从 0 开始重新计数。与此同时与此同时，触发标志位 (TIF) 会被置位，在触发中断使能的情况下还会产生一个中断请求。如下列示例对应时序图所示：

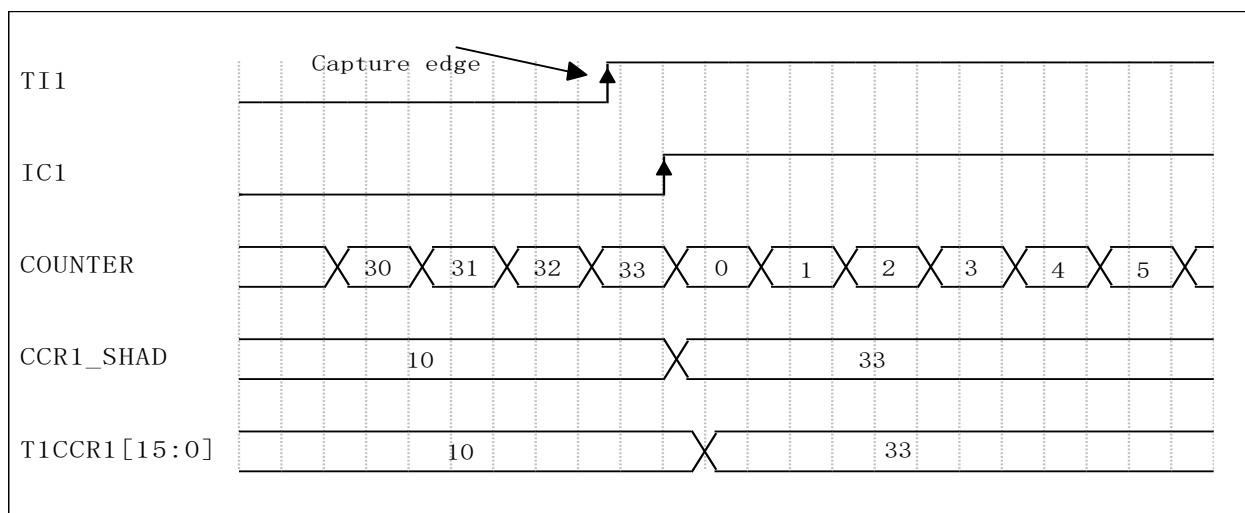


图 1-18 复位模式下，计数器计数时序图

● 门控模式：

依据选择的触发输入的电平值，计数器会被使能。此模式下，计数器的运行和停止都是受控的。

门控模式的步骤示例：

1. 使能 TIM1 模块时钟并选择 TIM1 时钟源
2. 配置输入捕捉寄存器的值 – 配置输入捕捉滤波器 T1IC1F=000；配置捕捉预分频器 T1IC1PSC=0
3. 将通道配置为输入捕捉通道 T1CC1S=01，并将 IC1 映射在 TI1FP1 上
4. 写 T1CC1P=1，选择检测输入低电平的到来
5. 通过写 T1SMS=101，将 TIM1 配置为门控模式。同时写 T1TS=101，选择 TI1 为输入源
6. 置位 T1CEN，使能计数器 (在门控模式下，需要开启 T1CEN；在此基础上，才能由输入源控制计数器的运行与停止)

以下是一段示例代码：

```

BANKSEL    PCKEN
BSR        PCKEN,0      ; 使能 TIM1 模块时钟

BANKSEL    TCKSRC
LDWI      H'01'
STR        TCKSRC      ; 选择 TIM1 时钟源为 HIRC

BANKSEL    TRISA
LDWI      H'FF'
STR        TRISA      ; PA0 为通道 1 的输入通道

BANKSEL    TIM1CCMR1
LDWI      H'01'
STR        TIM1CCMR1   ; 配置通道 1 的 IC1 映射在 TI1FP1 上

LDWI      H'55'
STR        TIM1SMCR     ; 配置 TIM1 为门控模式，触发源为 TI1FP1

LDWI      H'03'
STR        TIM1CCER1    ; 使能通道 1 并且低电平输入为有效电平
    
```

```

BANKSEL    TIM1CR1      ;
BSR        TIM1CR1,0    ; 开启计数器计数使能位

BTSS      TIM1SR1,6      ; 判断触发中断标志位是否为高
LJUMP     $-1            ;

BCR       TIM1SR1,6      ; 将触发中断标志位清零
    
```

当 TI1 为低电平时，计数器在内部时钟的驱动下进行计数；当 TI1 变为高电平时，计数器停止计数。触发标志位 (T1TIF) 会在计数器启动或停止时被置位。如下列示例对应时序图所示：

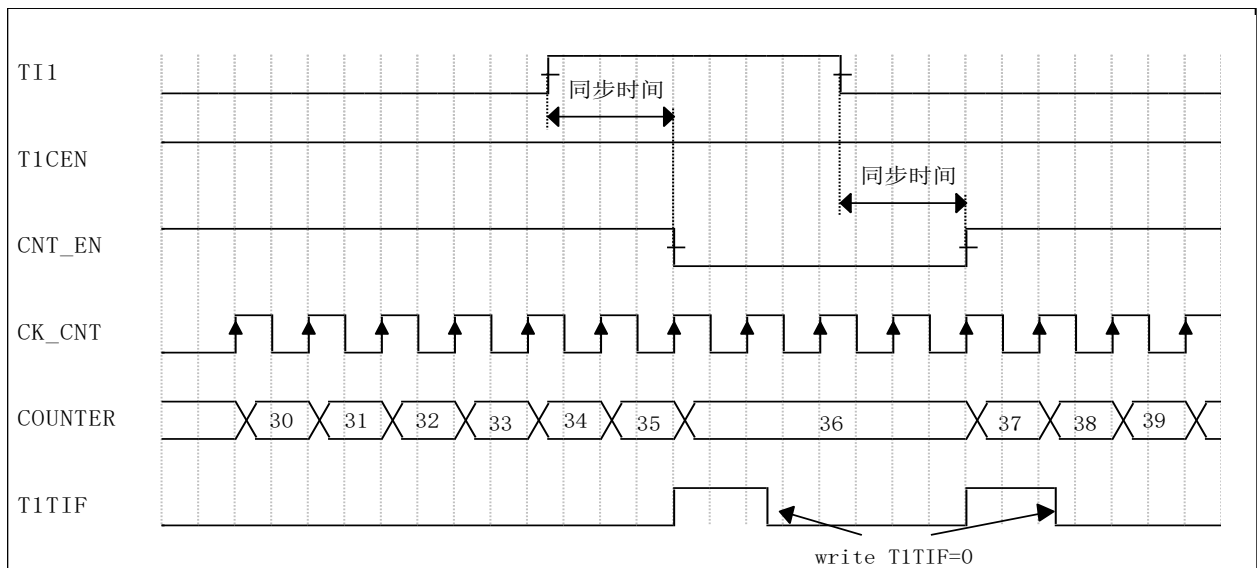


图 1-19 门控模式下，计数器计数时序图

● 触发模式：

依据选择的触发输入的电平值，计数器会被启动 (T1CEN 被置位)。

触发模式的步骤示例：

1. 使能 TIM1 模块时钟并选择 TIM1 时钟源
2. 配置输入捕捉寄存器的值 – 配置输入捕捉滤波器 T1IC2F = 000；配置捕捉预分频器 T1IC2PSC = 0
3. 将通道配置为输入捕捉通道 T1CC2S = 01，并将 IC2 映射在 TI2FP2 上
4. 写 T1CC2P = 0，选择检测触发上升沿的到来
5. 通过写 T1SMS = 110，将 TIM1 配置为触发模式。同时写 T1TS = 110，选择 TI2 为输入触发源

以下是一段示例代码：

```

BANKSEL    PCKEN      ;
BSR        PCKEN,0    ; 使能 TIM1 模块时钟

BANKSEL    TCKSRC      ;
LDWI      H'01'        ;
STR       TCKSRC      ; 选择 TIM1 时钟源为 HIRC

BANKSEL    TRISA      ;
LDWI      H'FF'        ;
STR       TRISA      ; PA1 为通道 2 的输入通道
    
```

```

BANKSEL    TIM1CCMR2    ;
LDWI       H'01'        ;
STR        TIM1CCMR2    ; 配置通道 2 的 IC2 映射在 TI2FP2 上

LDWI       H'66'        ;
STR        TIM1SMCR      ; 配置 TIM1 为触发控制模式，触发源为 TI2FP2

LDWI       H'10'        ;
STR        TIM1CCER1     ; 使能通道 2 并且为上升沿触发

BANKSEL    TIM1CR1      ;
BSR        TIM1CR1,0     ; 开启计数器计数使能位
    
```

当 TI2 的上升沿到来时，计数器在内部时钟的驱动下启动计数，并且触发标志位 (T1TIF) 被置位。如下列示例对应时序图所示：

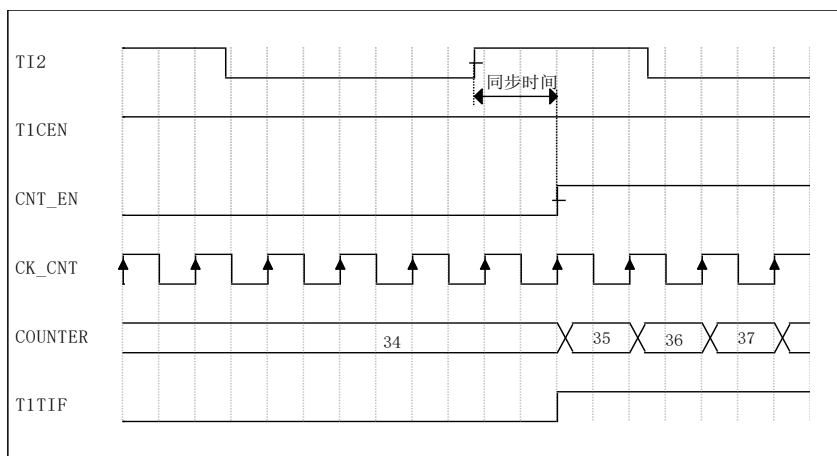


图 1-20 触发模式下，计数器计数时序图

1.3.3. 捕捉比较通道

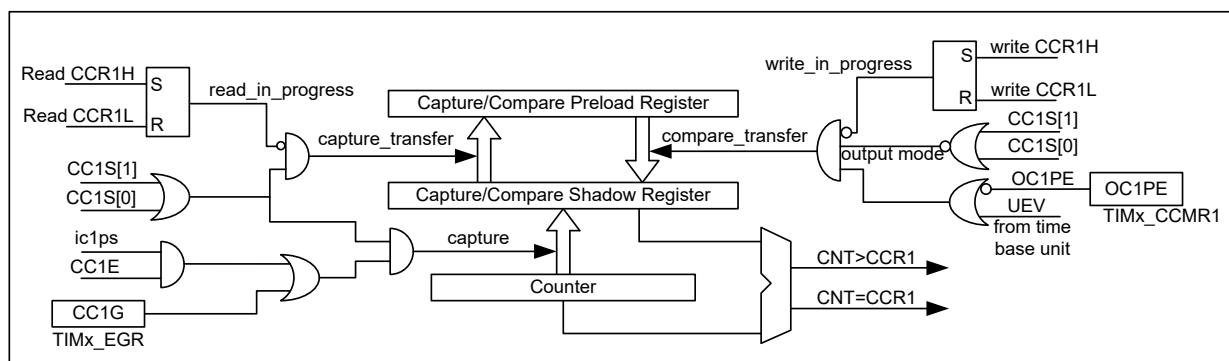


图 1-21 捕捉/比较通道 1 框图

TIMER 的 I/O 口能被配置为输入捕捉或输出比较功能。这个配置由 T1CCxS 通道选择位进行设定；对单个通道而言，输入捕捉功能和输出比较功能是互斥的两个功能。但每个通道都有独立的配置寄存器，所以可以将某些通道配置为输入捕捉功能，另一些通道配置为输出比较功能；例如：配置 T1CC1S=2'b00，T1CC2S=2'b00，T1CC3S=2'b01，T1CC3S=2'b10，这样通道 1 和通道 2 为输出比较通道，可在输出波形，而通道 3 和通道 4 为输入捕捉通道，可进行捕捉功能。

TIM1CCRxH/L 寄存器的读写：

TIM1CCRxH/L 寄存器由一个预加载寄存器和一个影子寄存器组成。

- 在输出比较模式下：

TIM1CCRxH/L 寄存器的访问没有任何限制，可读可写。

读 TIM1CCRxH/L：读出的值来自 CCRx 预加载寄存器的值，跟先前写入 TIM1CCRxH/L 寄存器的值保持一致。

写 TIM1CCRxH/L：有预加载使能位 (T1OCxPE)；如果预加载使能 (T1OCxPE=0) 关闭，则写入 TIM1CCRxH/L 寄存器的值直接由 CCRx 预加载寄存器传递到 CCRx 影子寄存器。反之，写入 TIM1CCRxH/L 寄存器的值在下次更新事件发生时才会从 CCRx 预加载寄存器传递到 CCRx 影子寄存器。

- 在输入捕捉模式下：

TIM1CCRxH/L 寄存器为只读寄存器。在捕捉事件发生时，计数器值会被写入到 CCRx 影子寄存器中，而后再写回到 CCRx 预加载寄存器中。

读 TIM1CCRxH/L 寄存器时，必须先读高 8 位，再读低 8 位。读高 8 位时，CCRx 预加载寄存器被冻结，此时计数器值无法写回到 CCRx 预加载寄存器中；只有按顺序读完低 8 位后，CCRx 预加载寄存器才能更新为最新一次捕捉值。

TIM1CCMRx 寄存器为复用寄存器。详见[章节 1.2](#)

1.3.3.1. 捕捉输入通道

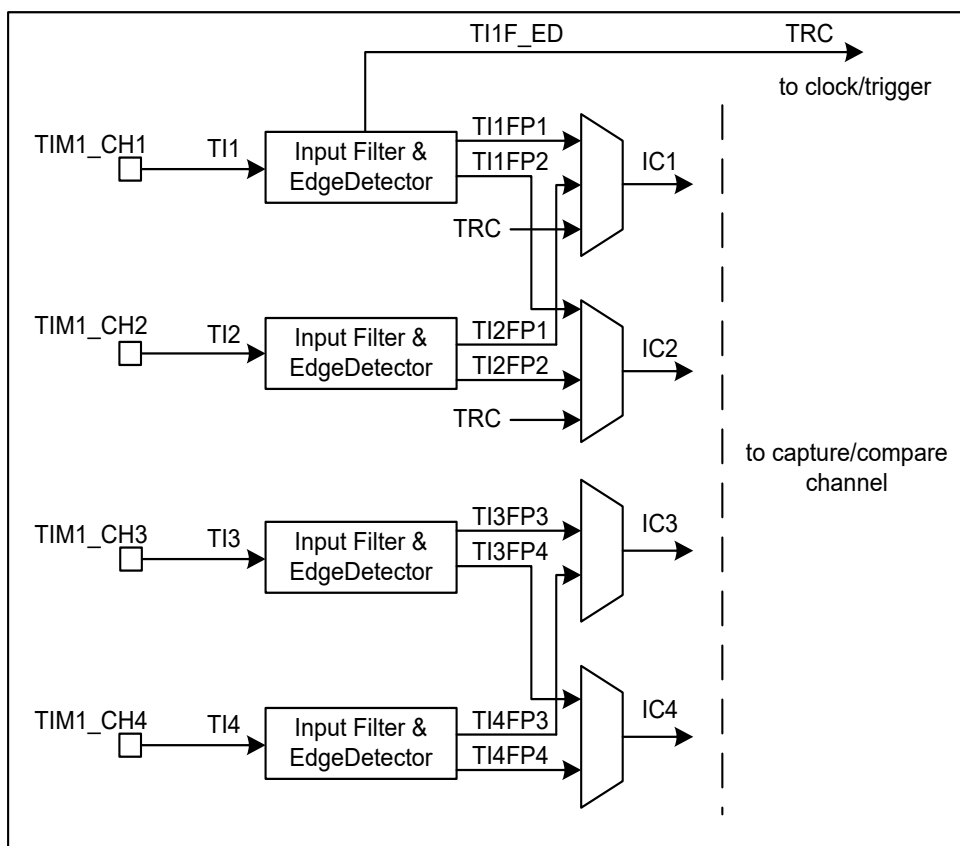


图 1-22 通道输入框图

信号名称	详细说明
TIM1_CH1/2/3/4	通道 1/2/3/4 对应 I/O 口的输入
IC1/2/3/4	通过选择后的真正的通道捕捉源
TI1FP1	来自通道 1 对应 I/O 的输入捕捉信号，作为通道 1 的捕捉源之一
TI1FP2	来自通道 1 对应 I/O 的输入捕捉信号，作为通道 2 的捕捉源之一
TI2FP2	来自通道 2 对应 I/O 的输入捕捉信号，作为通道 2 的捕捉源之一
TI2FP1	来自通道 2 对应 I/O 的输入捕捉信号，作为通道 1 的捕捉源之一
TI3FP3	来自通道 3 对应 I/O 的输入捕捉信号，作为通道 3 的捕捉源之一
TI3FP4	来自通道 3 对应 I/O 的输入捕捉信号，作为通道 4 的捕捉源之一
TI4FP4	来自通道 4 对应 I/O 的输入捕捉信号，作为通道 4 的捕捉源之一
TI4FP3	来自通道 4 对应 I/O 的输入捕捉信号，作为通道 3 的捕捉源之一
TRC	来自通道 1 对应 I/O 的输入双沿捕捉信号，作为通道 1 和通道 2 的捕捉源之一

表 1-16 信号说明列表

当一个通道被配置成输入捕捉通道并且输入捕捉事件有效时，可以将当前的计数值保存在 TIM1CCR_x 寄存器。每个通道都有一个数字滤波单元，可配置采样频率 (T1ICxF[3:0])，捕捉预分频 (T1IC1PSC[1:0])，捕捉极性选择 (T1CCxP) 和捕捉触发源 (T1CCxS)。每个通道都有各自的捕捉源，如下表所示：

T1CCxS (捕捉源选择)	通道 1	通道 2	通道 3	通道 4
2'b00	TI1FP1	TI2FP2	TI3FP3	TI4FP4
2'b01	TI2FP1	TI1FP2	TI4FP3	TI3FP4
2'b10	TRC	TRC	—	—

表 1-17 各通道输入捕捉源列表

当一个输入捕捉发生时：

- TIM1CCR1H/L 寄存器得到捕捉发生时计数器的值。
- 输入捕捉标志位 (T1CCxIF) 被置位。如果当 T1CCxIF 保持为 1 时，又一次发生了输入捕捉事件，那么溢出捕捉标志位 (T1CCxOF) 也会被置位。
- 如果 T1CCxIE 为 1，那么捕捉将产生一个中断事件。

配置为输入捕捉通道的示例步骤：

1. 使能 TIM1 模块时钟并选择 TIM1 时钟源
2. 将通道相应的端口配置为输入端口
3. 选择输入触发源 (T1CCxS)
4. 配置采样频率 (T1ICxF[3:0])，捕捉预分频 (T1IC1PSC[1:0])
5. 配置捕捉源的捕捉极性 (T1CCxP)
6. 使能捕捉通道 (T1CCxE)
7. 使能计数器 (T1CEN)

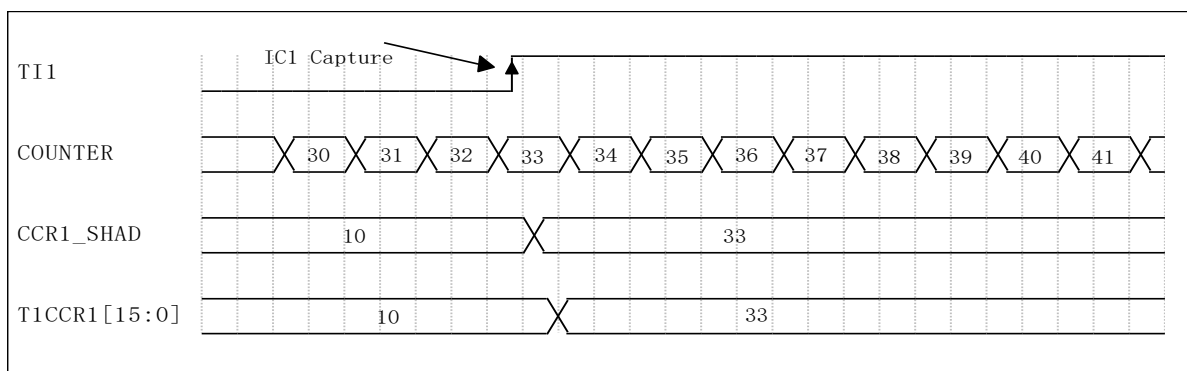


图 1-23 简单输入捕捉时序图

PWM 输入信号测量的应用：

利用捕捉输入模式和复位模式，并且将两个通道的输入捕捉源都选择为同一个通道的 PWM 信号输入；这样就可以测量从通道输入的 PWM 信号的周期以及占空比。

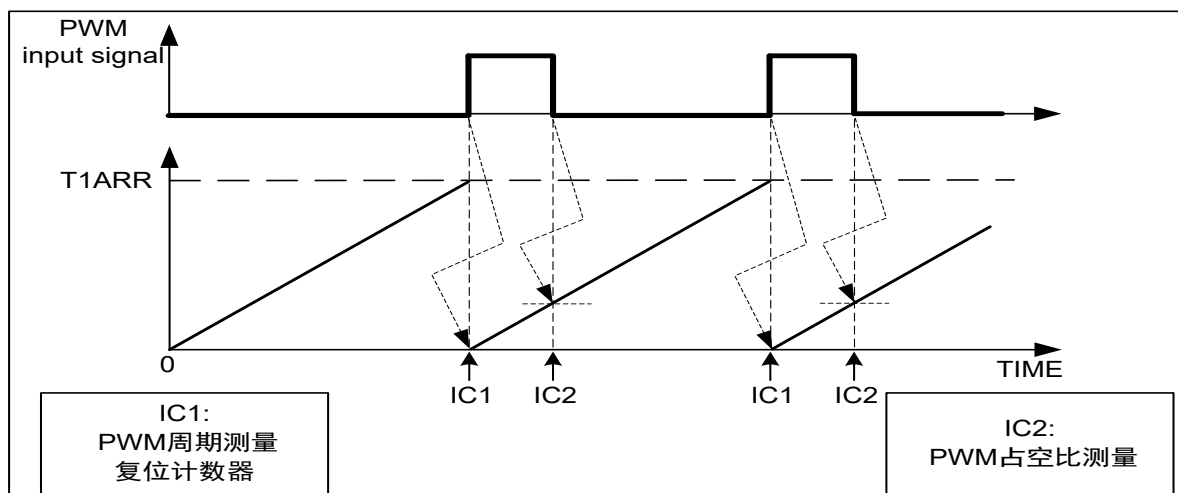


图 1-24 测量 PWM 信号的示意图

具体测量 PWM 的配置步骤如下：

1. 使能 TIM1 模块时钟并选择 TIM1 时钟源
2. 将通道 1/2 相应的端口配置为输入端口
3. 通道 1 配置将 IC1 映射在 TI1FP1 上；通道 2 配置将 IC2 映射在 TI2FP1 上
4. 配置通道 1 为上升沿捕捉 (T1CC1P=0)；通道 2 为下降沿捕捉 (T1CC2P=1)
5. 配置采样频率 (T1ICxF[3:0]=4'b0000)，捕捉预分频 (T1IC1PSC[1:0]=2'b00)
6. 将计数控制模式配置为复位模式 (T1SMS=101)，计数触发源配置为 TI1FP1 (T1TS=101)
7. 使能计数器 (T1CEN)
8. 开启通道 1 和通道 2 的输入捕捉功能 (T1CC1E=1 且 T1CC2E=1)

注意：

因为捕捉沿先于复位触发源两个计数时钟周期，所以需要软件进行一下操作才能等到准确测量值：

- 当预分频为 0 时，PWM 周期 = $T1CCR1H/L+2$ ，占空比 = $T1CCR2H/L+2$
- 当预分频为 1 时，PWM 周期 = $T1CCR1H/L+1$ ，占空比 = $T1CCR2H/L+1$
- 当预分频大于 1 时，PWM 周期 = $T1CCR1H/L$ ，占空比 = $T1CCR2H/L$

以下是一段示例代码：

```

BANKSEL    PCKEN
BSR        PCKEN,0
; 使能 TIM1 模块时钟

BANKSEL    TCKSRC
LDWI       H'01'
STR        TCKSRC
; 选择 TIM1 时钟源为 HIRC

BANKSEL    TRISA
LDWI       H'FF'
STR        TRISA
; 配置 PA0 为通道 1 的输入通道，PA1 为通道 2 的输入通道

BANKSEL    TIM1CCMR1
LDWI       H'01'

```

```

STR      TIM1CCMR1      ; 配置通道 1 为 IC1 映射在 TI1FP1 上
LDWI     H'02'          ;
STR      TIM1CCMR2      ; 配置通道 2 的 IC2 映射在 TI1FP2 上
LDWI     H'54'          ;
STR      TIM1SMCR        ; 配置 TIM1 为复位控制模式，触发源为 TI1FP1
LDWI     H'31'          ;
STR      TIM1CCER1      ; 使能通道 1/2，通道 1 为上升沿捕捉，通道 2 为下降沿捕捉
BANKSEL  TIM1CR1        ;
BSR      TIM1CR1,0      ; 开启计数器计数使能位
    
```

上述示例代码对应波形图：

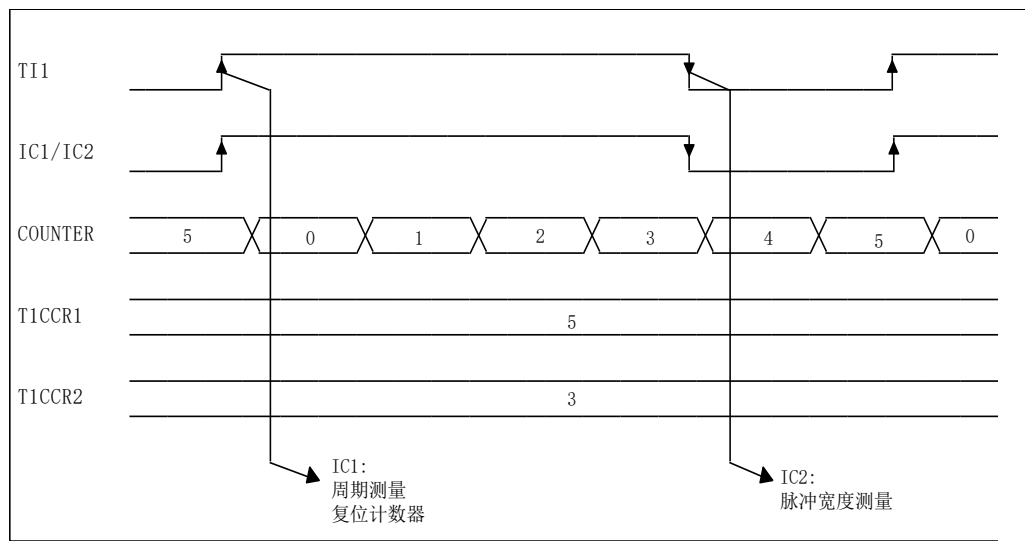


图 1-25 测量 PWM 信号的时序图

1.3.3.2. 输出比较通道

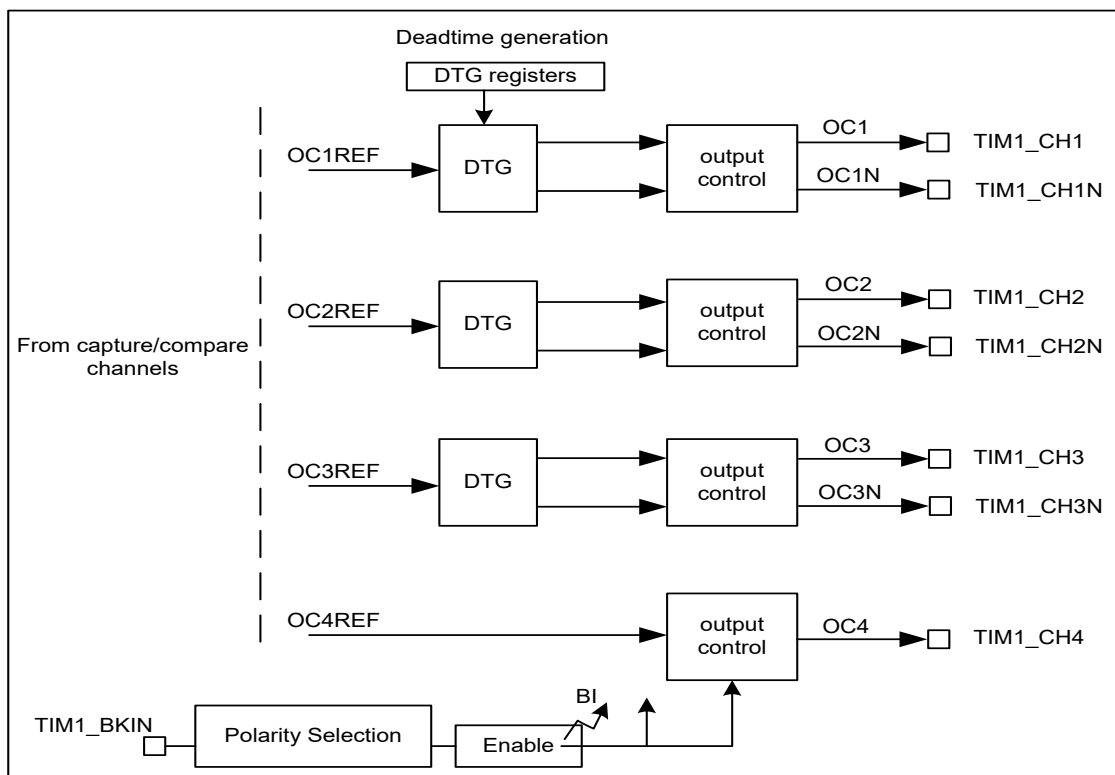


图 1-26 通道输出框图

输出阶段产生立即响应的波形，用来作为参考波形，叫做 OCxREF 信号（高有效）。刹车功能，极性选择和其他输出控制位都在参考波形之后去做控制。

输出比较通道根据计数值与比较值 CCRx，产生 OCxREF 输出并送到死区产生模块，经过死区产生模块后再经过其他输出控制位的控制将波形输出到端口。

具体的输出控制位以及可达到的输出效果可以查看[章节 1.3.3.5](#) 内容。

输出比较模式下，可选择不同的输出模式去输出 PWM 波形；输出模式由 T1OCxM[3:0]选择，总共有以下 8 种不同的输出模式(最终的输出还需要取决于极性选择 (T1CCxP)):

- (1) 冻结模式：输出冻结，输出实际比较值 (CCRx_SHAD) 与计数器 TIM1_CNT 间的比较对 OC1REF 不起作用
- (2) 匹配有效：当计数值 CNT 与实际比较值 (CCRx_SHAD) 匹配时，OCxREF 为高电平；
- (3) 匹配无效：当计数值 CNT 与实际比较值 (CCRx_SHAD) 匹配时，OCxREF 为低电平；
- (4) 翻转：当计数值 CNT 与实际比较值 (CCRx_SHAD) 匹配时，输出翻转；
- (5) 强制无效：OCxREF 强制为低电平；
- (6) 强制有效：OCxREF 强制为高电平；
- (7) PWM1：向上计数时，当 CNT < 实际比较值 (CCRx_SHAD) 时，OCxREF 有效；
向下计数时，CNT > 实际比较值 (CCRx_SHAD) 时，OCxREF 无效；
- (8) PWM2：向上计数时，当 CNT < 实际比较值 (CCRx_SHAD) 时，OCxREF 无效；
向下计数时，CNT > 实际比较值 (CCRx_SHAD) 时，OCxREF 有效；

配置为输出比较通道的示例步骤和示例代码如下：

1. 使能 TIM1 模块时钟并选择 TIM1 时钟源

2. 将通道相应的端口配置为输出端口
3. 配置输出波形的周期 (T1ARR) 和占空比 (T1CCRx)
4. 配置输出比较模式 (T1OCxM) 和输出极性 (T1CCxP)
5. 使能比较输出通道 (T1CCxE)
6. 打开主输出自动使能位 (T1AOE), 在更新事件发生时硬件会自动使能主输出 (T1MOE)
7. 使能计数器 (T1CEN)

```

BANKSEL    PCKEN          ;
BSR        PCKEN,0        ; 使能 TIM1 模块时钟

BANKSEL    TCKSRC         ;
LDWI       H'01'          ;
STR        TCKSRC         ; 选择 TIM1 时钟源为 HIRC

BANKSEL    TRISA          ;
LDWI       H'FE'          ;
STR        TRISA          ; 配置 PA0 为通道 1 的输出通道

BANKSEL    TIM1ARRL       ;
LDWI       H'05'          ;
STR        TIM1ARRL       ; 将输出波形周期配置为 6

LDWI       H'03'          ;
STR        TIM1CCR1L       ; 将输出波形占空比配置为 3

BANKSEL    TIM1CCMR1      ;
LDWI       H'10'          ;
STR        TIM1CCMR1      ; 配置通道 1 为匹配有效模式输出

LDWI       H'01'          ;
STR        TIM1CCER1      ; 使能通道 1

BANKSEL    TIM1BKR        ;
BSR        TIM1BKR,6      ; 打开主输出自动使能位 T1AOE

BANKSEL    TIM1CR1        ;
BSR        TIM1CR1,0      ; 开启计数器计数使能位
    
```

上述示例代码对应波形图:

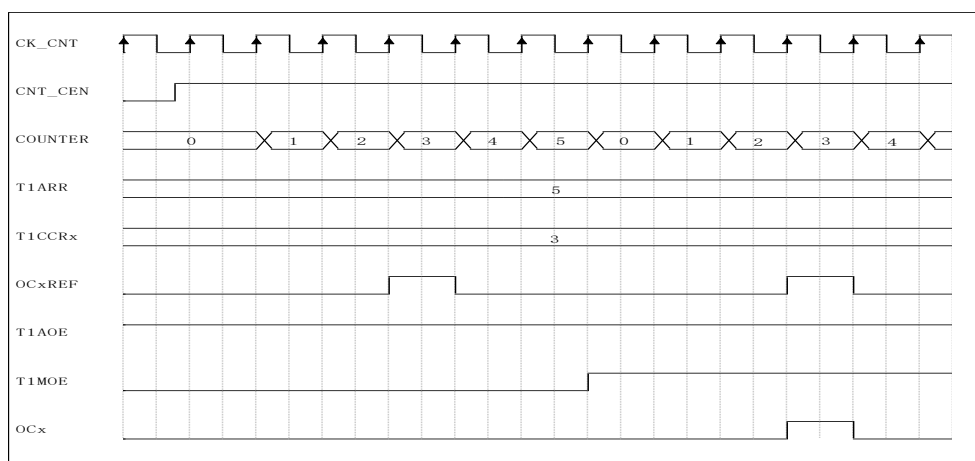


图 1-27 T1OCxM 为匹配有效模式下的输出时序图

示例代码：

```

BANKSEL PCKEN      ;
BSR     PCKEN,0    ; 使能 TIM1 模块时钟

BANKSEL TCKSRC      ;
LDWI    H'01'       ;
STR     TCKSRC      ; 选择 TIM1 时钟源为 HIRC

BANKSEL TRISA       ;
LDWI    H'FE'       ;
STR     TRISA        ; 配置 PA0 为通道 1 的输出通道

BANKSEL TIM1ARRL    ;
LDWI    H'05'       ;
STR     TIM1ARRL     ; 将输出波形周期配置为 6

LDWI    H'03'       ;
STR     TIM1CCR1L    ; 将输出波形占空比配置为 3

BANKSEL TIM1CCMR1   ;
LDWI    H'30'       ;
STR     TIM1CCMR1    ; 配置通道 1 为翻转模式输出

LDWI    H'01'       ;
STR     TIM1CCER1    ; 使能通道 1

BANKSEL TIM1BKR     ;
BSR     TIM1BKR,6    ; 打开主输出自动使能位 T1AOE

BANKSEL TIM1CR1     ;
BSR     TIM1CR1,0    ; 开启计数器计数使能位
    
```

上述示例代码对应波形图：

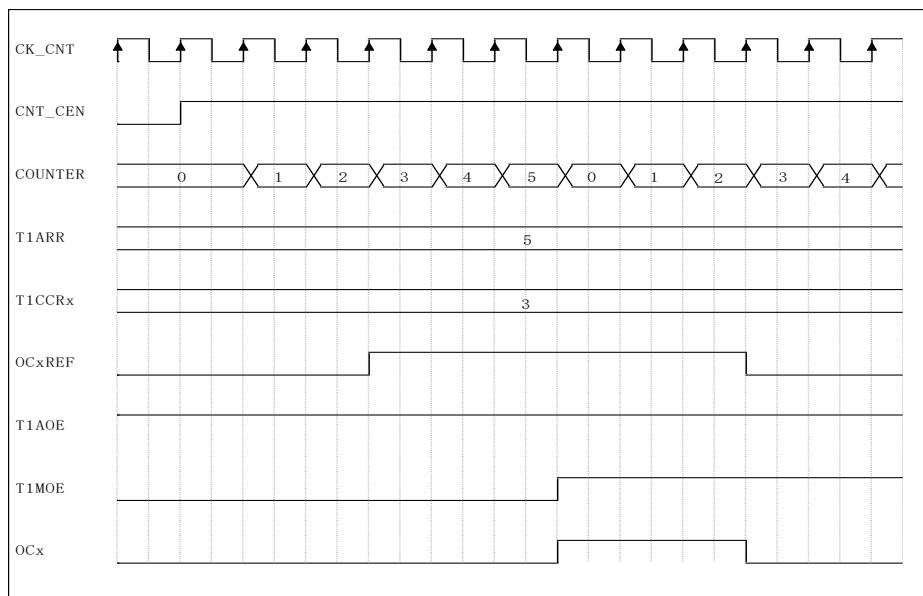


图 1-28 T1OCxM 为翻转模式下的输出时序图

示例代码：

```

BANKSEL PCKEN      ;
BSR     PCKEN,0    ; 使能 TIM1 模块时钟

BANKSEL TCKSRC      ;
LDWI    H'01'       ;
STR     TCKSRC      ; 选择 TIM1 时钟源为 HIRC
    
```

```

BANKSEL    TRISA      ;
LDWI       H'FE'      ;
STR        TRISA      ; 配置 PA0 为通道 1 的输出通道

BANKSEL    TIM1ARRL    ;
LDWI       H'05'      ;
STR        TIM1ARRL    ; 将输出波形周期配置为 6

LDWI       H'03'      ;
STR        TIM1CCR1L    ; 将输出波形占空比配置为 3

BANKSEL    TIM1CCMR1    ;
LDWI       H'70'      ;
STR        TIM1CCMR1    ; 配置通道 1 为 PWM2 模式输出

LDWI       H'01'      ;
STR        TIM1CCER1    ; 使能通道 1

BANKSEL    TIM1BKR      ;
BSR        TIM1BKR,6    ; 打开主输出自动使能位 T1AOE

BANKSEL    TIM1CR1      ;
BSR        TIM1CR1,0    ; 开启计数器计数使能位
    
```

上述示例代码对应波形图：

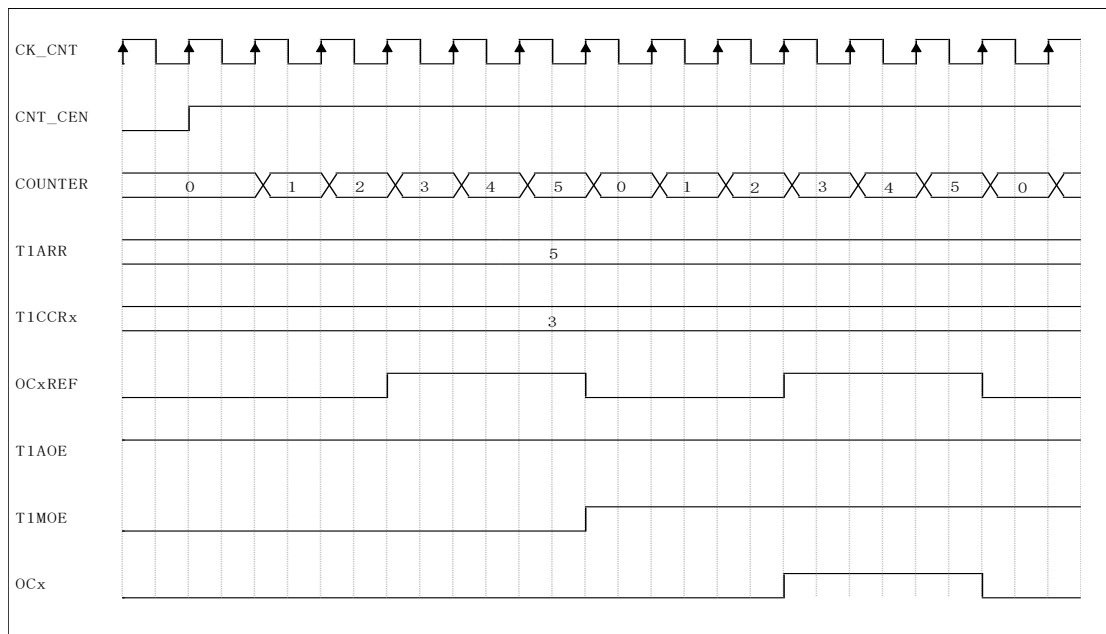


图 1-29 T1OCxM 为 PWM2 模式下的输出时序图

注意：

- 刹车事件的到来只会影响输出，并不影响 TIMER 本身的计数。
- 为了避免在计数器还未开启时，就开始输出脉冲；需要在计数器使能之后再开启 T1CCxE 和 T1CCxNE

1.3.3.3. 单次脉冲模式

开启单次脉冲模式和不开单次脉冲模式的区别在于：开启单次脉冲模式 (T1OPM=1) 并且下一次更新事件到来时，硬件会自动关闭计数器使能位 (T1CEN)，计数器停止计数。

想要产生一个正确的脉冲，比较值 (T1CCR_x) 必须与计数器初始值 (T1ARR) 不同；所以在开始计数之前必须满足以下配置：

- 在向上计数模式下: $COUNTER < T1CCR_x < T1ARR$
- 在向下计数模式下: $COUNTER > T1CCR_x$

想要产生一个正确的脉冲, 还必须满足以下配置:

- 在输出模式为 PWM1 模式 ($T1OC_xM=110$) 下: $T1CC_xP$ 必须为 1
如果输出模式为 PWM1 模式并且 $T1CC_xP$ 为 0, 更新事件之后 PWM 输出会一直为有效值
- 在输出模式为 PWM2 模式 ($T1OC_xM=111$) 下: $T1CC_xP$ 必须为 0
如果输出模式为 PWM2 模式并且 $T1CC_xP$ 为 1, 更新事件之后 PWM 输出会一直为有效值

单次脉冲模式可以配合触发模式在特定的时间点产生单个特定的 PWM 输出; 配置步骤如下所示:

1. 使能 TIM1 模块时钟并选择 TIM1 时钟源
2. 将通道 2 相应的端口配置为输入端口, 通道 1 相应的端口配置为输出端口
3. 通道 2 配置 $T1CC2S$ 为 01, IC2 映射在 $TI2FP2$ 上; 并配置通道 2 为上升沿捕捉 ($T1CC2P=0$)
4. 将计数控制模式配置为触发模式 ($T1SMS=110$), 计数触发源配置为 $TI2FP2$ ($T1TS=110$)
5. 通道 1 配置为输出通道 ($T1CC1S=00$)
6. 通道 1 的比较输出模式配置为 PWM2 模式 ($T1OC1M=111$), 输出极性配置为高电平有效 ($T1CC1P=0$)
7. 打开主输出使能 ($T1MOE$) 并且使能计数器 ($T1CEN$)
8. 开启通道 2 的输入捕捉功能 ($T1CC2E=1$) 和通道 1 的输出比较功能 ($T1CC1E$)

以下是一段示例代码:

```
BANKSEL    PCKEN          ;
BSR        PCKEN,0       ; 使能 TIM1 模块时钟

BANKSEL    TCKSRC         ;
LDWI       H'01'          ;
STR        TCKSRC         ; 选择 TIM1 时钟源为 HIRC

BANKSEL    TRISA          ;
LDWI       H'FE'          ;
STR        TRISA          ; 配置 PA0 为通道 1 的输出通道, PA1 为通道 2 的输入通道

BANKSEL    TIM1CCMR1      ;
LDWI       H'70'          ;
STR        TIM1CCMR1      ; 配置通道 1 为 PWM2 模式输出

LDWI       H'01'          ;
STR        TIM1CCMR2      ; 配置通道 2 的 IC2 映射在 TI2FP2 上

LDWI       H'66'          ;
STR        TIM1SMCR        ; 配置 TIM1 为触发控制模式, 触发源为 TI2FP2

LDWI       H'11'          ;
STR        TIM1CCER1      ; 使能通道 1 和通道 2

BANKSEL    TIM1BKR        ;
BSR        TIM1BKR,7      ; 打开主输出使能 T1MOE

BANKSEL    TIM1CR1        ;
BSR        TIM1CR1,0      ; 开启计数器计数使能位
```

上述示例代码对应示意图：

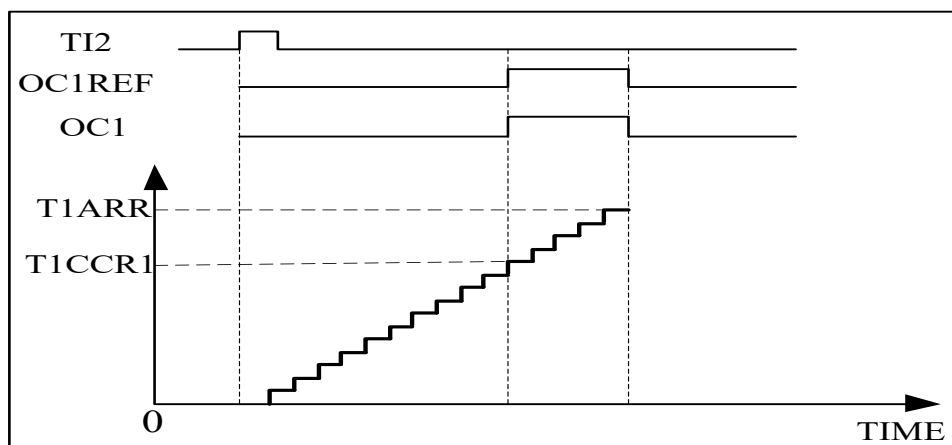


图 1-30 单次脉冲应用示意图

1.3.3.4. 死区产生

当把通道的互补输出使能时，就自动使能死区功能。每当一个输出信号（正向输出信号或互补输出信号）出现下降沿时，就会将另一个信号的上升沿后延一个死区时间长度。如图 1-31 和图 1-32 所示：

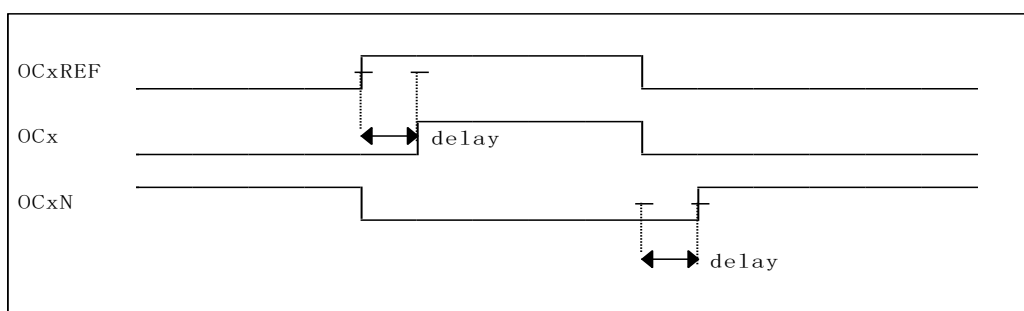


图 1-31 正向输出插入死区时序图

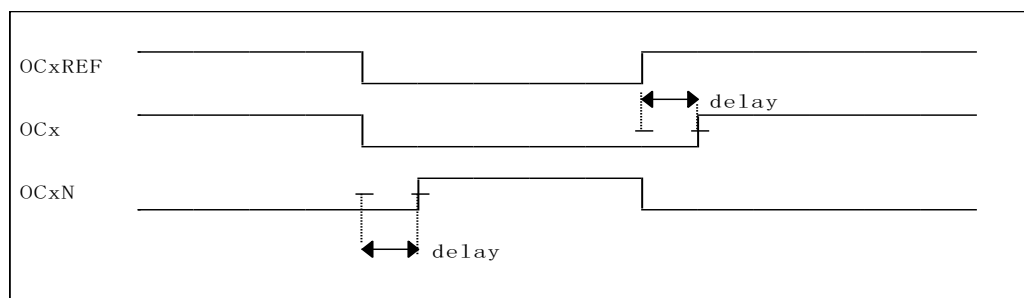


图 1-32 互补输出插入死区时序图

死区时间可以编程:根据寄存器位 T1DTG[7:0]，可以配置死区时间长度，具体参考寄存器描述 TIM1DTR 的 T1DTG[7:0]。

有的 OCXREF 输出的脉冲时间很短（小于死区时间），有可能某一脉冲信号（正向输出信号或反向输出信号）会被死区覆盖，导致输出不变化。如图 1-29 和图 1-30 所示：

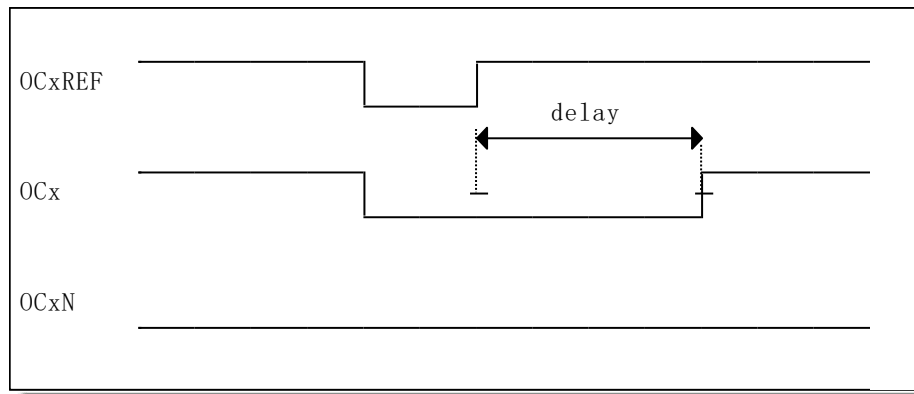


图 1-33 正向输出被死区覆盖时序图

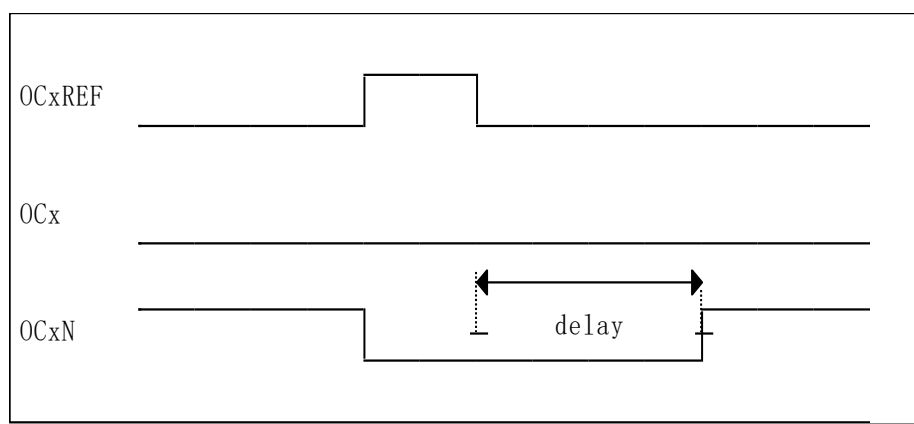


图 1-34 互补输出被死区覆盖时序图

1.3.3.5. 输出控制

参考信号 OCxREF 产生后不直接输出到端口，而是先经过死区控制模块和极性选择，再由以下 5 位控制信号 (T1MOE、T1OSSI、T1OSSR、T1CCxNE 和 CCxE) 进行组合控制之后送到端口上。具体的组合控制内容请见表 1-10 Timer1 输出控制和状态”

输出的状态转换实际是刹车事件异步将 T1MOE 清零实现的。T1MOE 异步清零，但要注意刹车事件撤销时需要同步 2 个 CK_CNT 时钟 (如果有时钟的情况下)。

1.3.4. TIM1 中断

TIM1 有以下 7 个中断请求源：

- 刹车中断
- 触发中断
- 捕捉/比较 4 中断
- 捕捉/比较 3 中断
- 捕捉/比较 2 中断
- 捕捉/比较 1 中断

- 更新中断 (例如: 上溢、下溢、计数初始化)

在用这些中断之前需要提前打开 TIM1IER 寄存器中的中断使能位 (T1BIE、T1TIE、T1CCxIE 和 T1UIE)。

不同的中断源还可以配置通过 TIM1EGR 寄存器来产生 (软件产生中断)。

1.3.5. 故障刹车源

TIM1 有以下 3 种刹车事件:

- BKIN 管脚事件
- LVD 事件
- ADC 比较事件

当故障事件有效且被选择为刹车源 (由 BKS0~2 决定), 如果 BKE 位为 1, PWM 输出管脚将被置于预设的状态, 预设状态由寄存器 TIM1OISR 决定。

当一个刹车事件发生时:

- T1MOE 位会被异步清 0, 强制输出进入无效状态, 空闲状态或复位状态。甚至在 MCU 振荡器关闭的情况下, T1MOE 也会被刹车事件清 0。
- 在 T1MOE=0 之后, 每个输出通道都会先将输出值置为无效值, 等死区时间到之后变成提前设置好的 T1OISx 位的值。如果 T1OSSI=0, TIMER 会将输出关闭。
- 当互补输出使能时:

输出首先会设置为无效值 (根据极性选择位)。此操作是异步清 0 的, 所以即使 TIM1 没有时钟驱动也能进行。

如果 TIM1 是有时钟进行驱动的, 那么死区时间到来之后就会进入由 T1OISx 和 T1OISxN 提前设定的预设状态。(由于 T1MOE 的同步, 所以此情况下真正的死区时间会比死区设置值长 2 个 CK_CNT 时钟)

- 刹车状态标志位 (T1BIF) 被置位。如果 T1BIE 位为 1, 那么将会产生一个中断事件。
- 如果 T1AOE 位配置为 1, 那么 T1MOE 位在下次更新事件 (UEV) 到来时, 将会由硬件自动置位。如果 T1AOE 位为 0, 那么只能由软件将 T1MOE 位重新置位。

当故障事件有效时, T1MOE 清 0, PWM 输出将一直置于预设状态;

故障事件撤消后, 如果 T1AOE=1, PWM 将在下一次更新事件后恢复正常输出, 否则, 软件需要自动打开 T1MOE。



1.3.6. 前沿消隐

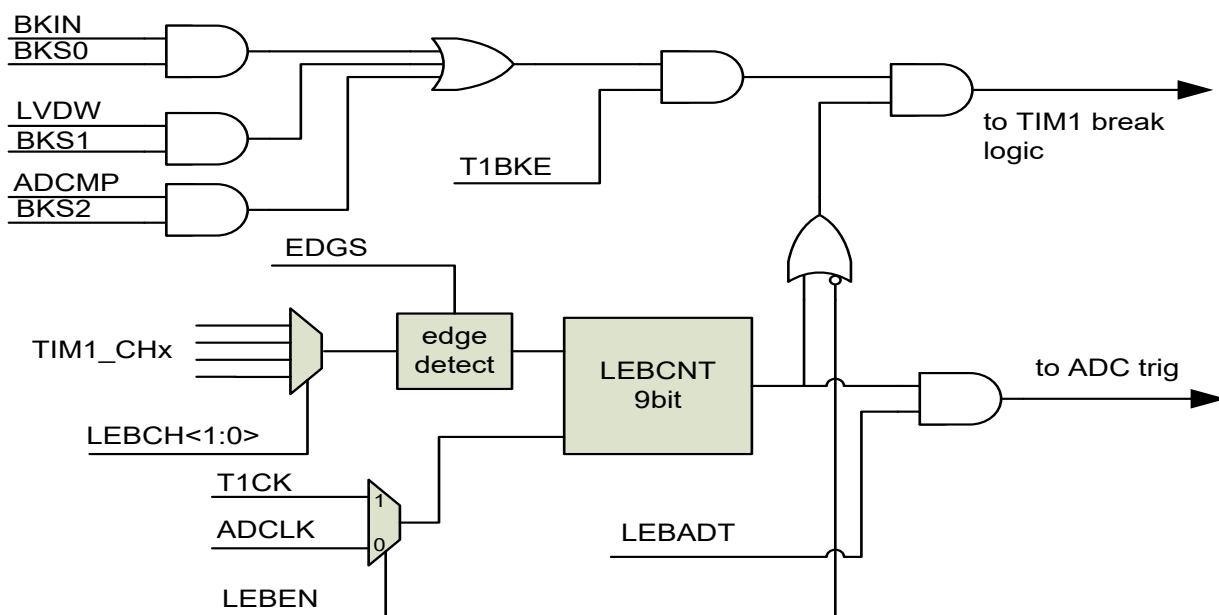


图 1-37 前沿消隐原理框图

在高速开关应用中，开关通常会产极大的瞬变，这些瞬变可能会导致测量误差。利用前沿消隐 (LEB) 功能，应用程序可以忽略 PWM 输出边沿附近发生的瞬变。

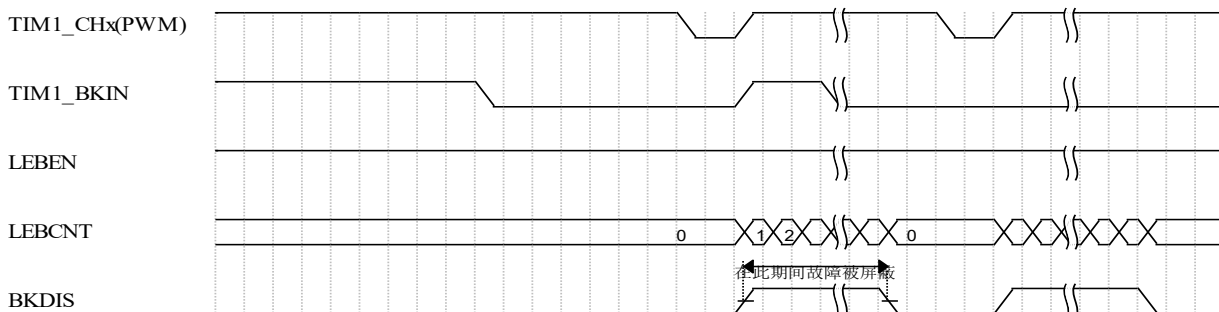


图 1-38 前沿消隐时序示意图

LEBCH 用于选择被消隐 TIM1 的 PWM 通道，EDGS 选择边沿类型。当 LEBEN 为 1，PWM 边沿将触发 LEB 定时器计数，时钟源为 TIM1 时钟，直到计数值等于 LEBPR，LEB 定时器停止计数，这段时间为消隐周期，期间所发生的刹车事件将被忽略；在消隐周期内如果再次发生有效的 PWM 边沿，则 LEB 定时器将清 0，重新开始计数。

注意：

- (1) LEB 定时器和 ADC 延时定时器复用了同一个 9bit 计数器，当 LEBEN 为 1 时，原 ADC 的延时触发功能被禁止，但如果 LEBADT 为 1，LEB 定时器溢出将触发一次 AD 转换。
- (2) 寄存器 ADCON3 中 ADCMPEN 位的关闭能将 ADCMP 产生的刹车事件清除。

2. 基本定时器 TIM4

2.1. 特性

- 8bit 自动重载向上计数器
- 计数时钟可编程预分频
- 计数器溢出中断

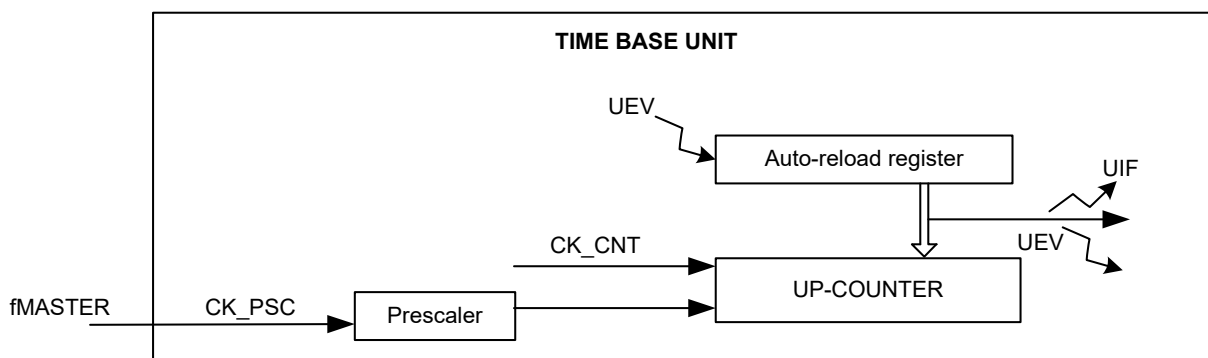


图 2-39 TIM4 原理框图

2.2. TIM4 相关寄存器汇总

[illegible]

表 2-18 Timer4 相关用户寄存器汇总

名称	状态		寄存器	地址	复位值
TIM4EN	<u>TIM4 模块时钟</u>	1 = 使能 0 = 关闭	PCKEN[3]	0x9A	RW-0
SYSON	睡眠模式下，系统时钟控制	1 = 使能 0 = 关闭	CKOCON[7]	0x95	RW-0
T4ARPE	<u>周期的自动预装载</u> 1 = 使能 (T4ARR 预装载值在更新事件到来时被加载) 0 = 禁止 (T4ARR 立即被加载)		TIM4CR1[7]	0x111	RW-0
T4CKS	<u>Timer4 时钟源</u> 00 = Sysclk 10 = LP ^(*) 01 = HIRC 11 = XT ^(*) ^(*) FOSC 应相应配置成 LP/XT 或选择 INTOSCIO 模式，否则振荡器将不会运行		TIM4CR1[5:4]		RW-00
T4OPM	<u>单脉冲模式</u> 1 = 使能 (下一次更新事件到来时，计数器停止) 0 = 关闭 (发生更新事件时，计数器不停止)		TIM4CR1[3]		RW-0
T4URS	<u>当 T4UDIS = 0 时，更新事件中断源</u> 1 = 计数器上溢 0 = 软件设置 T4UG 位或计数器上溢		TIM4CR1[2]		RW-0
T4UDIS	<u>产生更新事件控制</u> 1 = 禁止 0 = 允许		TIM4CR1[1]		RW-0
T4CEN	TIM4 计数器	1 = 使能 0 = 关闭	TIM4CR1[0]		RW-0
T4PSC	<u>Timer4 预分频器</u> 000 = 1 100 = 16 001 = 2 101 = 32 010 = 4 110 = 64 011 = 8 111 = 128 注：必须产生更新事件或 T4CEN=0，更新的预分频值才生效		TIM4PSCR[2:0]	0x116	RW-000
T4CNT	Timer4 计数值		TIM4CNTR[7:0]	0x115	RW-0000 0000
T4ARR	<u>周期的自动重载寄存器(预装载值)</u> 注：此值为 0 时，计数器不工作		TIM4ARR[7:0]	0x117	RW-1111 1111

表 2-19 Timer4 相关用户控制寄存器

名称	状态		寄存器	地址	复位值
GIE	全局中断 1 = 使能 (PEIE, T4UIE, T4UG 适用) 0 = 全局关闭 (唤醒不受影响)		INTCON[7]	Bank 首地址 +0x0B	RW-0
PEIE	外设总中断	1 = 使能 (T4UIE, T4UG 适用) 0 = 关闭 (无唤醒)	INTCON[6]		RW-0
T4UIE	允许更新中断	1 = 使能	TIM4IER[0]	0x112	RW-0
T4UG ¹⁰	允许更新软件中断	0 = 关闭	TIM4EGR[0]	0x114	WO-0
T4UIF	更新中断标志位	1 = 更新事件等待响应 0 = 无更新事件	TIM4SR[0]	0x113	R_W1C-0

表 2-20 Timer4 中断使能和状态位

2.3. TIM4 时钟源

TIM4 有 4 种时钟源可选，由寄存器位 T4CKS 设置。在 TIM4 的被使能 (PCKEN.TIM4EN=1) 的情况下，所选择的时钟源被自动使能。

注意：

1. 如果要选择 LP 晶体时钟，系统时钟配置寄存器位 FOSC 必须选择 LP 模式，否则对应的时钟源将不被使能；
2. 同理，如果要选择 XT 晶体时钟，系统时钟配置寄存器位 FOSC 必须选择 XT 模式，否则对应的时钟源将不被使能；

SLEEP 模式下，如果 SYSON 为 1，且 TIM4EN=1，则所选择的时钟源将保持振荡，TIM4 将继续工作；否则，所选的时钟源取决于其他模块的设置情况。

2.4. 预分频器

计数时钟可以进行 3bit 的时钟预分频：

$$f_{CK_CNT} = F_{master} / 2^{(PSCR[2:0])}$$

预分频支持分频自动更新，即在更新事件发生后，能够自动改变预分频值。当 T4CEN 为 0 时，写入预分频寄存器的值也能直接加载实际应用的预分频寄存器中。

2.5. TIM4 中断

TIM4 只有一个中断请求源：

- 更新中断 (计数器上溢或计数器初始化)

在用这些中断之前需要提前打开 TIM4IER 寄存器中的中断使能位 (T4UIE)。不同的中断源还可以配置通过 TIM4EGR 寄存器来产生 (软件产生中断 T4UG)。

¹⁰ 软件置 1，硬件自动清 0。

3. 应用范例

```
//*****
/* 文件名: TEST_62F08x_TIM1_CAPTURE.c
* 功能:    FT62F08x_TIM1_CAPTURE 功能演示
* IC:      FT62F088 LQFP32
* 内部:    16M
* empno:   500
* 说明:    例程通过 TIM1 的捕获通道 1 测量 TIM4 输出到 PB3 的波形的周期 (4K)
*          TIM1 计数器捕获的数据放在 testdata 数组里, 一共 20 个。
*          程序只用到一个数据(testdata[1])来计算周期。
*
* 参考原理图 TEST_62F08x_sch.pdf
*/
//*****
#include "SYSCFG.h"
//*****宏定义*****
#define unchar unsigned char
#define uint unsigned int

#define DemoPortOut PB3
#define DemoPortOut1 PB4

volatile uint testdata[20];
volatile unchar y=0;
volatile unchar dataH;
volatile unchar dataL;
volatile uint period=0;
/*-----
* 函数名: interrupt ISR
* 功能:   定时器 4 的中断处理
* 输入:   无
* 输出:   无
-----*/
void interrupt ISR(void)
{
    if(T4UIE && T4UIF)
    {
        T4UIF = 1;                //写 1 清零标志位
        DemoPortOut = ~DemoPortOut; //翻转电平
    }
    if(T1CC1IE && T1CC1IF)        //捕获中断
    {
        T1CC1IF=1;
    }
}
```

```

    T1CC1OF=0;
    DemoPortOut1 = ~DemoPortOut1;    //捕获中断调试查看
    dataH = TIM1CCR1H;
    dataL = TIM1CCR1L;
    testdata[y++]=(uint)(dataH<<8 | dataL);
    NOP();

    if(y>=20)
    {
        y=0;

        //计算周期(μs)
        period = testdata[1]/32;    //周期约等于 250μs
    }
}
/*-----
* 函数名: POWER_INITIAL
* 功能:   上电系统初始化
* 输入:   无
* 输出:   无
-----*/
void POWER_INITIAL (void)
{

    OSCCON = 0B01110001;    //IRCF=111=16MHz 1:1
    INTCON = 0;    //暂禁止所有中断

    PORTA = 0B00000000;
    TRISA = 0B00000001;    //PA 输入输出 0-输出 1-输入  PA0-IN
    PORTB = 0B00000000;
    TRISB = 0B00000000;    //PB 输入输出 0-输出 1-输入  PB3,PB4-OUT

    PORTC = 0B00000000;
    TRISC = 0B00000000;    //PC 输入输出 0-输出 1-输入
    PORTD = 0B00000000;
    TRISD = 0B00000000;    //PD 输入输出 0-输出 1-输入

    WPUA = 0B00000000;    //PA 端口上拉控制 1-开上拉 0-关上拉
    WPUB = 0B00000000;    //PB 端口上拉控制 1-开上拉 0-关上拉
    WPUC = 0B00000000;    //PC 端口上拉控制 1-开上拉 0-关上拉
    WPUD = 0B00000000;    //PD 端口上拉控制 1-开上拉 0-关上拉

    WPDA = 0B00000000;    //PA 端口下拉控制 1-开下拉 0-关下拉

```

```
WPDB = 0B00000000;           //PB 端口下拉控制 1-开下拉 0-关下拉
WPDC = 0B00000000;           //PC 端口下拉控制 1-开下拉 0-关下拉
WPDD = 0B00000000;           //PD 端口下拉控制 1-开下拉 0-关下拉

PSRC0 = 0B11111111;          //PORTA,PORTB 源电流设置最大
PSRC1 = 0B11111111;          //PORTC,PORTD 源电流设置最大

PSINK0 = 0B11111111;          //PORTA 灌电流设置最大 0:最小, 1:最大
PSINK1 = 0B11111111;          //PORTB 灌电流设置最大 0:最小, 1:最大
PSINK2 = 0B11111111;          //PORTC 灌电流设置最大 0:最小, 1:最大
PSINK3 = 0B11111111;          //PORTD 灌电流设置最大 0:最小, 1:最大

ANSELA = 0B00000000;          //全为数字管脚
}
```

```
/*-----
 * 函数名: Time4Initial
 * 功能:   初始化定时器 4
 * 输入:   无
 * 输出:   无
-----*/
```

```
void Time4Initial(void)
```

```
{
    PCKEN |= 0B00001000;           //TIME4 模块时钟使能

    TIM4CR1 = 0B00000001;
    //Bit7: 0: TIM1_ARR 寄存器没有缓冲, 它可以被直接写入; 1: TIM1_ARR 寄存器由预装载缓冲器
缓冲。
    //Bit6: 保留
    //Bit[5:4]: timer4 时钟选择位。
    //00: 系统时钟/主时钟
    //01: 内部快时钟 HIRC
    //10: LP 时钟, 只有当 FOSC 选择 LP 模式时才有意义
    //11: XT 时钟, 只有当 FOSC 选择 XT 模式时才有意义

    //Bit3:单脉冲模式
    //0: 在发生更新事件时, 计数器不停止;
    //1: 在发生下一次更新事件(清除 CEN 位)时, 计数器停止。

    //Bit2:更新请求源
    //0: 如果 UDIS 允许产生更新事件, 则下述任一事件产生一个更新中断:
    //寄存器被更新(计数器上溢/下溢)
    //软件设置 UG 位
    //时钟/触发控制器产生的更新
    //1: 如果 UDIS 允许产生更新事件, 则只有当下列事件发生时才产生更新中断, 并 UIF 置 1:
    //寄存器被更新(计数器上溢/下溢)

    //Bit1:禁止更新
    //0: 一旦下列事件发生, 产生更新(UEV)事件:
    //计数器溢出/下溢
    //产生软件更新事件
    //时钟/触发模式控制器产生的硬件复位被缓存的寄存器被装入它们的预装载值。
    //1: 不产生更新事件, 影子寄存器(ARR、PSC、CCRx)保持它们的值。如果设置了 UG 位或时钟/
触发控制器发出了一个硬件复位, 则计数器和预分频器被重新初始化。

    //Bit0: 0: 禁止计数器; 1: 使能计数器。

    TIM4IER=0B00000001;
```

//Bit0: 0: 禁止更新中断; 1: 允许更新中断。

TIM4SR = 0B00000000;

//Bit0: 当产生更新事件时该位由硬件置 1。它由软件写 1 清 0

//0: 无更新事件产生;

//1: 更新事件等待响应。当寄存器被更新时该位由硬件置 1:

//若 TIM4_CR1 寄存器的 UDIS=0, 当计数器上溢或下溢时;

//若 TIM4_CR1 寄存器的 UDIS=0、URS=0, 当设置 TIM4_EGR 寄存器的 UG 位软件对计数器

//CNT 重新初始化时;

//若 TIM4_CR1 寄存器的 UDIS=0、URS=0, 当计数器 CNT 被触发事件重新初始化时。

TIM4EGR = 0B00000000;

//Bit0: 该位由软件置 1, 由硬件自动清 0。

//0: 无动作;

//1: 重新初始化计数器, 并产生一个更新事件。注意预分频器的计数器也被清 0(但是预分频系数不变)。若在中心对称模式下或 DIR=0(向上计数)则计数器被清 0; 若 DIR=1(向下计数)则计数器取 TIM1_ARR 的值。

TIM4CNTR=0;

//TIM4 8 位计数器

TIM4PSCR=0B00000100;

//预分频器对输入的 CK_PSC 时钟进行分频。

//计数器的时钟频率 fCK_CNT 等于 fCK_PSC/2(PSC[2:0])。PSC[7:3]由硬件清 0。

//PSCR 包含了当更新事件产生时装入当前预分频器寄存器的值(包括由于清除 TIMx_EGR 寄存器的 UG 位产生的计数器清除事件),

//这意味着如要新的预分频值生效, 必须产生更新事件或者 CEN=0。

TIM4ARR = 124;

//ARR 包含了将要装载入实际的自动重装载寄存器的值。

//当自动重装载的值为空时, 计数器不工作。

INTCON |= 0B11000000;

//开总中断和外设中断

}

/*-----*/

* 函数名: Time1Initial

* 功能: 初始化计时器 1

* 输入: 无

* 输出: 无

-----*/

void Time1_Capture_Initial(void)

{

PCKEN |= 0B00000010;

//使能 timer1 时钟模块

CKOCON = 0B00100000;

TCKSRC = 0B00000011;

//TIM1 时钟为 HIRC 的 2 倍频

//Bit7 低频内振模式：1=256K 振荡频率模式,0 = 32K 振荡频率模式

//Bit[6:4]:TIM2 时钟源选择位

//值 时钟源

//0 系统时钟/主时钟

//1 HIRC

//2 XT 时钟/外部时钟

//3 HIRC 的 2 倍频

//4 XT 时钟/外部时钟的 2 倍频

//5 LIRC

//6 LP 时钟/外部时钟

//7 LP 时钟/外部时钟的 2 位频

//Bit3:保留位

//Bit[2:1]:TIM1 时钟源选择位

//值 时钟源

//0 系统时钟/主时钟

//1 HIRC

//2 XT 时钟/外部时钟

//3 HIRC 的 2 倍频

//4 XT 时钟/外部时钟的 2 倍频

//5 LIRC

//6 LP 时钟/外部时钟

//7 LP 时钟/外部时钟的 2 位频

TIM1CR1 = 0B00000110; //预载允许，边沿对齐向上计数器，计数器使能

//Bit7:自动预装载允许位

//0: TIM1_ARR 寄存器没有缓冲，它可以被直接写入；

//1: TIM1_ARR 寄存器由预装载缓冲器缓冲。

//Bit[6:5]:选择对齐模式

//00: 边沿对齐模式。计数器依据方向位(DIR)向上或向下计数。

//01: 中央对齐模式 1。计数器交替地向上和向下计数。配置为输出的通道(TIM1_CCMRx 寄存器中 CciS=00)的输出比较中断标志位，只在计数器向下计数时被置 1。

//10: 中央对齐模式 2。计数器交替地向上和向下计数。配置为输出的通道(TIM1_CCMRx 寄存器中 CciS=00)的输出比较中断标志位，只在计数器向上计数时被置 1。

//11: 中央对齐模式 3。计数器交替地向上和向下计数。配置为输出的通道(TIM1_CCMRx 寄存器中 CciS=00)的输出比较中断标志位，在计数器向上和向下计数时均被置 1。

//Bit4:方向

//0: 计数器向上计数；

//1: 计数器向下计数。

//Bit3:单脉冲模式

//0: 在发生更新事件时, 计数器不停止;

//1: 在发生下一次更新事件(清除 CEN 位)时, 计数器停止。

//Bit2:更新请求源

//0: 如果 UDIS 允许产生更新事件, 则下述任一事件产生一个更新中断:

//寄存器被更新(计数器上溢/下溢)

//软件设置 UG 位

//时钟/触发控制器产生的更新

//1: 如果 UDIS 允许产生更新事件, 则只有当下列事件发生时才产生更新中断, 并 UIF 置 1:

//寄存器被更新(计数器上溢/下溢)

//Bit1: 禁止更新

//0: 一旦下列事件发生, 产生更新(UEV)事件:

//计数器溢出/下溢

//产生软件更新事件

//时钟/触发模式控制器产生的硬件复位被缓存的寄存器被装入它们的预装载值。

//1: 不产生更新事件, 影子寄存器(ARR、PSC、CCRx)保持它们的值。如果设置了 UG 位或时钟/触发控制器发出了一个硬件复位, 则计数器和预分频器被重新初始化。

//Bit0: 允许计数器

//0: 禁止计数器;

//1: 使能计数器。

TIM1SMCR = 0B01010000;

TIM1SMCR |= 0B00000100;

//Bit7: 保留位

//Bit[6:4]: 触发选择,这 3 位选择用于选择同步计数器的触发输入。

//000: 内部触发 ITR0 连接到 TIM6 TRGO (此设计没有 TIM6, 所以固定接 0)

//001: 保留

//010: 内部触发 ITR2 连接到 TIM5 TRGO(此设计没有 TIM5, 所以固定接 0)

//011: 保留

//100: TI1 的边沿检测器(TI1F_ED)

//101: 滤波后的定时器输入 1(TI1FP1)

//110: 滤波后的定时器输入 2(TI2FP2)

//111: 外部触发输入(ETRF)

//注: 这些位只能在未用到(如 SMS=000)时被改变, 以避免在改变时产生错误的边沿检测。

//Bit3:保留

//Bit[2:0]: 时钟/触发/从模式选择,当选择了外部信号, 触发信号(TRGI)的有效边沿与选中的外部输入极性相关(见输入控制寄存器和控制寄存器的说明)

//000: 时钟/触发控制器禁止 - 如果 CEN=1, 则预分频器直接由内部时钟驱动。

//001: 编码器模式 1 - 根据 TI1FP1 的电平, 计数器在 TI2FP2 的边沿向上/下计数。
 //010: 编码器模式 2 - 根据 TI2FP2 的电平, 计数器在 TI1FP1 的边沿向上/下计数。
 //011: 编码器模式 3 - 根据另一个输入的电平, 计数器在 TI1FP1 和 TI2FP2 的边沿向上/下计数。
 //100: 复位模式 - 在选中的触发输入(TRGI)的上升沿时重新初始化计数器, 并且产生一个更新寄存器的信号。

//101: 门控模式 - 当触发输入(TRGI)为高时, 计数器的时钟开启。一旦触发输入变为低, 则计数器停止(但不复位)。计数器的启动和停止都是受控的。

//110: 触发模式 - 计数器在触发输入 TRGI 的上升沿启动(但不复位), 只有计数器的启动是受控的。

//111: 外部时钟模式 1 - 选中的触发输入(TRGI)的上升沿驱动计数器。

//注: 如果 TI1F_ED 被选为触发输入(TS=100)时, 不要使用门控模式。这是因为 TI1F_ED 在每次 //TI1F 变化时只是输出一个脉冲, 然而门控模式是要检查触发输入的电平。

TIM1IER = 0B00000010; //捕获比较 1 中断使能

//Bit7: 允许刹车中断

//0: 禁止刹车中断;

//1: 允许刹车中断。

//Bit6: 触发中断使能

//0: 禁止触发中断;

//1: 使能触发中断。

//Bit5: 保留位

//Bit4: 允许捕获/比较 4 中断

//0: 禁止捕获/比较 4 中断;

//1: 允许捕获/比较 4 中断。

//Bit3: 允许捕获/比较 3 中断

//0: 禁止捕获/比较 3 中断;

//1: 允许捕获/比较 3 中断。

//Bit2: 允许捕获/比较 2 中断

//0: 禁止捕获/比较 2 中断;

//1: 允许捕获/比较 2 中断。

//Bit1: 允许捕获/比较 1 中断

//0: 禁止捕获/比较 1 中断;

//1: 允许捕获/比较 1 中断。

//Bit0: 允许更新中断

//0: 禁止更新中断;

//1: 允许更新中断。

TIM1SR1 = 0B00000000; //T1CC1IF

//Bit7: 刹车中断标记(写 1 清 0, 写 0 无效)

//0: 无刹车事件产生;

//1: 刹车输入上检测到有效电平。

//Bit6: 触发器中断标记(写 1 清 0, 写 0 无效)

//0: 无触发器事件产生;

//1: 触发中断等待响应。

//Bit5:保留位

//Bit4: 捕获/比较 4 中断标记

//参考 CC1IF 描述。

//Bit3: 捕获/比较 3 中断标记

//参考 CC1IF 描述。

//Bit2: 捕获/比较 2 中断标记

//参考 CC1IF 描述。

//Bit1: 捕获/比较 1 中断标记 如果通道 CC1 配置为输出模式: (写 1 清 0, 写 0 无效)

//0: 无匹配发生;

//1: TIMx_CNT 的值与 TIMx_CCR1 的值匹配。

//注: 在中心对称模式下, 当计数器值为 0 时, 向上计数, 当计数器值为 ARR 时, 向下计数 (它从 0 向上计数到 ARR-1, 再由 ARR 向下计数到 1)。因此, 对所有的 SMS 位值, 这两个值都不置标记。但是, 如果 CCR1>ARR, 则当 CNT 达到 ARR 值时, CC1IF 置 1。

//如果通道 CC1 配置为输入模式: 当捕获事件发生时该位由硬件置 1, 它由软件清 0 或通过读 TIM1_CCR1L 清 0。

//0: 无输入捕获产生;

//1: 计数器值已被捕获(拷贝)至 TIM1_CCR1(在 IC1 上检测到与所选极性相同的边沿)。

//Bit0: 更新中断标记,当产生更新事件时该位由硬件置 1。它由软件清 0。

//0: 无更新事件产生;

//1: 更新事件等待响应。当寄存器被更新时该位由硬件置 1:

//若 TIM1_CR1 寄存器的 UDIS=0, 当计数器上溢或下溢时;

//若 TIM1_CR1 寄存器的 UDIS=0、URS=0, 当设置 TIM1_EGR 寄存器的 UG 位软件对计数器

//CNT 重新初始化时;

//若 TIM1_CR1 寄存器的 UDIS=0、URS=0, 当计数器 CNT 被触发事件重新初始化时 (参考 0

//从模式控制寄存器 TIM1_SMCR)。

TIM1SR2 = 0B00000000;

TIM1EGR = 0B00000000;

//Bit7: 产生刹车事件,该位由软件置 1, 用于产生一个刹车事件, 由硬件自动清 0。

//0: 无动作;

//1: 产生一个刹车事件。此时 MOE=0、BIF=1, 若开启对应的中断(BIE=1), 则产生相应的中断。

//Bit[6:5]:保留位

//Bit4: 产生捕获/比较 4 事件

//参考 CC1G 描述。

//Bit3: 产生捕获/比较 3 事件

//参考 CC1G 描述。

//Bit2: 产生捕获/比较 2 事件

//参考 CC1G 描述。

//Bit1: 产生捕获/比较 1 事件

//该位由软件置 1, 用于产生一个捕获/比较事件, 由硬件自动清 0。

//0: 无动作;

//1: 在通道 CC1 上产生一个捕获/比较事件: 若通道 CC1 配置为输出:

//设置 CC1IF=1, 若开启对应的中断, 则产生相应的中断。若通道 CC1 配置为输入:

//当前的计数器值被捕获至 TIM1_CCR1 寄存器, 设置 CC1IF=1, 若开启对应的中断, 则产生相应的中断。若 CC1IF 已经为 1, 则设置 CC1OF=1。

//Bit0: 保留位

TIM1CCMR1 = 0B00000001; //CC1 通道被配置为输入

//Bit7: 保留位

//Bit[6:4]: 输出比较 1 模式, 该 3 位定义了输出参考信号 OC1REF 的动作, 而 OC1REF 决定了 OC1 的值。OC1REF 是高电平有效, 而 OC1 的有效电平取决于 CC1P 位。

//000: 冻结。输出比较寄存器 TIM1_CCR1 与计数器 TIM1_CNT 间的比较对 OC1REF 不起作用;

//001: 匹配时设置通道 1 的输出为有效电平。当计数器 TIM1_CNT 的值与捕获/比较寄存器 1(TIM1_CCR1)相同时, 强制 OC1REF 为高。

//010: 匹配时设置通道 1 的输出为无效电平。当计数器 TIM1_CNT 的值与捕获/比较寄存器 1(TIM1_CCR1)相同时, 强制 OC1REF 为低。

//011: 翻转。当 TIM1_CCR1=TIM1_CNT 时, 翻转 OC1REF 的电平。

//100: 强制为无效电平。强制 OC1REF 为低。

//101: 强制为有效电平。强制 OC1REF 为高。

//110: PWM 模式 1— 在向上计数时, 一旦 TIM1_CNT<TIM1_CCR1 时通道 1 为有效电平, 否则为无效电平; 在向下计数时, 一旦 TIM1_CNT>TIM1_CCR1 时通道 1 为无效电平(OC1REF=0), 否则为有效电平(OC1REF=1)。

//111: PWM 模式 2— 在向上计数时, 一旦 TIM1_CNT<TIM1_CCR1 时通道 1 为无效电平, 否则为有效电平; 在向下计数时, 一旦 TIM1_CNT>TIM1_CCR1 时通道 1 为有效电平, 否则为无效电平。

//注 1: 一旦 LOCK 级别设为 3(TIM1_BKR 寄存器中的 LOCK 位)并且 CC1S=00(该通道配置成输出)则该位不能被修改。

//注 2: 在 PWM 模式 1 或 PWM 模式 2 中, 只有当比较结果改变了或在输出比较模式中从冻结模式

切换到 PWM 模式时, OC1REF 电平才改变。(参考 17.5.7PWM 模式)

//注 3: 在有互补输出的通道上, 这些位是预装载的。如果 TIM1_CR2 寄存器的 CCPC=1, OCM 位只有在 COM 事件发生时, 才从预装载位取新值。

//Bit3: 输出比较 1 预装载使能

//0: 禁止 TIM1_CCR1 寄存器的预装载功能, 可随时写入 TIM1_CCR1 寄存器, 并且新写入的数值立即起作用。

//1: 开启 TIM1_CCR1 寄存器的预装载功能, 读写操作仅对预装载寄存器操作, TIM1_CCR1 的预装载值在更新事件到来时被加载至当前寄存器中。

//注 1: 一旦 LOCK 级别设为 3(TIM1_BKR 寄存器中的 LOCK 位)并且 CC1S=00(该通道配置成输出)则该位不能被修改。

//注 2: 为了操作正确, 在 PWM 模式下必须使能预装载功能。但在单脉冲模式下(TIM1_CR1 寄存器的 OPM=1), 它不是必须的?

//Bit2: 保留位

//Bit[1:0]:捕获/比较 1 选择。这 2 位定义通道的方向(输入/输出), 及输入脚的选择:

//00: CC1 通道被配置为输出;

//01: CC1 通道被配置为输入, IC1 映射在 TI1FP1 上;

//10: CC1 通道被配置为输入, IC1 映射在 TI2FP1 上;

//11: CC1 通道被配置为输入, IC1 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时(由 TIM1_SMCR 寄存器的 TS 位选择)。

//注: CC1S 仅在通道关闭时(TIM1_CCER1 寄存器的 CC1E=0)才是可写的。

TIM1CCER1 =0B00000001; //上升沿触发, 触发使能

//Bit7: 输入捕获/比较 2 互补输出极性。参考 CC1NP 的描述。

//Bit6: 输入捕获/比较 2 互补输出使能。参考 CC1NE 的描述。

//Bit5: 输入捕获/比较 2 输出极性。参考 CC1P 的描述。

//Bit4: 输入捕获/比较 2 输出使能。参考 CC1E 的描述。

//Bit3: 输入捕获/比较 1 互补输出极性

//0: OC1N 高电平有效;

//1: OC1N 低电平有效。

//注 1: 一旦 LOCK 级别(TIM1_BKR 寄存器中的 LCKC 位)设为 3 或 2 且 CC1S=00(通道配置为输出)则该位不能被修改。

//Bit2: 输入捕获/比较 1 互补输出使能

//0: 关闭— OC1N 禁止输出, 因此 OC1N 的输出电平依赖于 MOE、OSSI、OSSR、OIS1、OIS1N 和 CC1E 位的值。

//1: 开启— OC1N 信号输出到对应的输出引脚, 其输出电平依赖于 MOE、OSSI、OSSR、OIS1、OIS1N 和 CC1E 位的值。

```
//Bit1: 输入捕获/比较 1 输出极性 CC1 通道配置为输出:
//0: OC1 高电平有效;
//1: OC1 低电平有效。
//CC1 通道配置为触发(参考图 61):
//0: 触发发生在 TI1F 的高电平或上升沿;
//1: 触发发生在 TI1F 的低电平或下降沿。
//CC1 通道配置为输入(参考图 61):
//0: 捕捉发生在 TI1F 的高电平或上升沿;
//1: 捕捉发生在 TI1F 的低电平或下降沿。
//注 1: 一旦 LOCK 级别(TIM1_BKR 寄存器中的 LCCK 位)设为 3 或 2, 则该位不能被修改。
//COM 事件发生时, CC1P 位才从预装载位中取新值。
```

```
//Bit0: 输入捕获/比较 1 输出使能
//CC1 通道配置为输出:
//0: 关闭— OC1 禁止输出, 因此 OC1 的输出电平依赖于 MOE、OSSI、OSSR、OIS1、OIS1N
和 CC1NE 位的值。
//1: 开启— OC1 信号输出到对应的输出引脚, 其输出电平依赖于 MOE、OSSI、OSSR、OIS1、
OIS1N 和 CC1NE 位的值。CC1 通道配置为输入
//该位决定了计数器的值是否能捕获入 TIM1_CCR1 寄存器。
//0: 捕获禁止;
//1: 捕获使能。
```

```
TIM1CCER2 = 0B00000000;
```

```
TIM1CNTRH = 0B00000000; //TIM1 计数器
TIM1CNTRL = 0B00000000;
```

```
TIM1PSCRH = 0B00000000; //计数时钟不分频
TIM1PSCRL = 0B00000000;
```

```
        T1CEN=1;
    }
/*-----
* 函数名: main
* 功能: 主函数
* 输入: 无
* 输出: 无
-----*/
void main(void)
{
    POWER_INITIAL(); //系统初始化
    Time1_Capture_Initial();
    Time4Initial();
```

```
while(1)
{
    NOP();
}
}
```

联系信息

Fremont Micro Devices Corporation

#5-8, 10/F, Changhong Building
Ke-Ji Nan 12 Road, Nanshan District,
Shenzhen, Guangdong, PRC 518057

Tel: (+86 755) 8611 7811

Fax: (+86 755) 8611 7810

Fremont Micro Devices (HK) Limited

#16, 16/F, Block B, Veristrong Industrial Centre,
34-36 Au Pui Wan Street, Fotan, Shatin, Hong Kong SAR

Tel: (+852) 2781 1186

Fax: (+852) 2781 1144

<http://www.fremontmicro.com>

* Information furnished is believed to be accurate and reliable. However, Fremont Micro Devices Corporation assumes no responsibility for the consequences of use of such information or for any infringement of patents of other rights of third parties, which may result from its use. No license is granted by implication or otherwise under any patent rights of Fremont Micro Devices Corporation. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. Fremont Micro Devices Corporation products are not authorized for use as critical components in life support devices or systems without express written approval of Fremont Micro Devices Corporation. The FMD logo is a registered trademark of Fremont Micro Devices Corporation. All other names are the property of their respective owners.