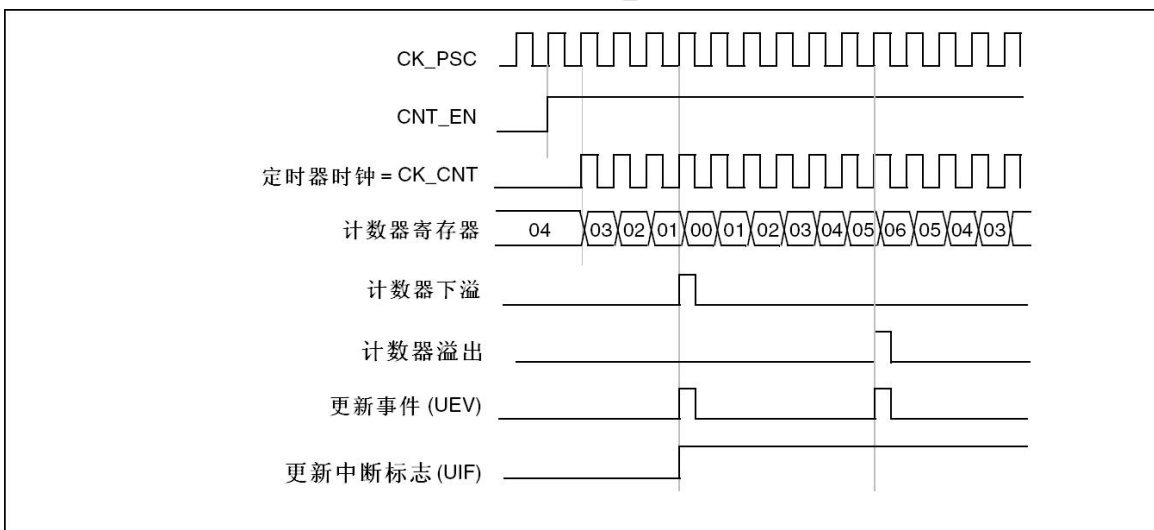


- 预分频器的缓存器被加载为预装载(TIMx_PSCRR寄存器)的值。
- 当前的自动加载寄存器被更新为预装载值(TIMx_ARLR寄存器中的内容)。注：如果因为计数器溢出而产生更新，自动重装载将在计数器重载入之前被更新，因此下一个周期将是预期的值(计数器被装载为新的值)。

以下是一些计数器在不同时钟频率下的操作的例子：

图64 计数器时序图，内部时钟分频因子为1，TIMx_ARLR=0x6



1. 这里使用了中心对齐模式1(详见13.4.1节)。

图65 计数器时序图，内部时钟分频因子为2

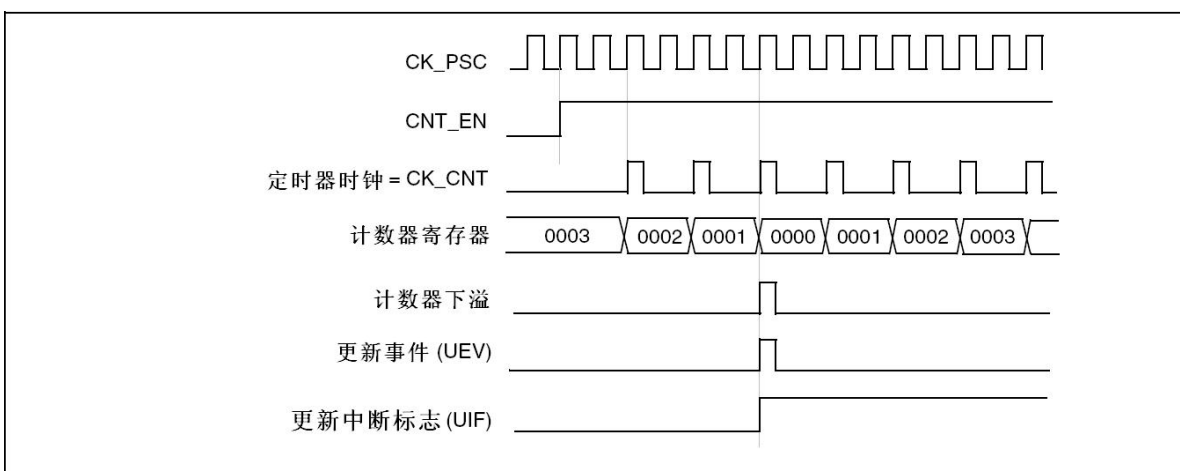


图66 计数器时序图，内部时钟分频因子为4，TIMx_ARLR=0x36

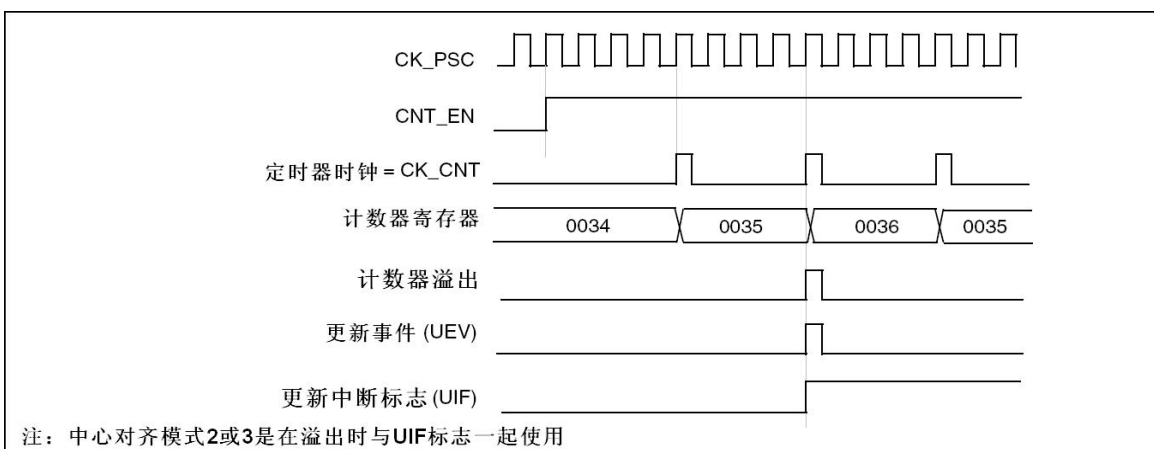


图67 计数器时序图，内部时钟分频因子为N

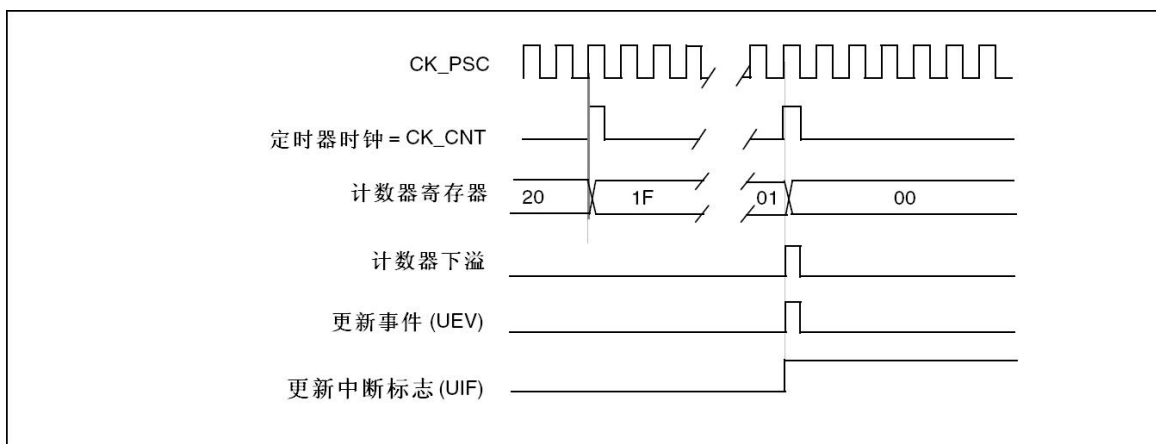


图68 计数器时序图，ARPE=1时的更新事件(计数器下溢)

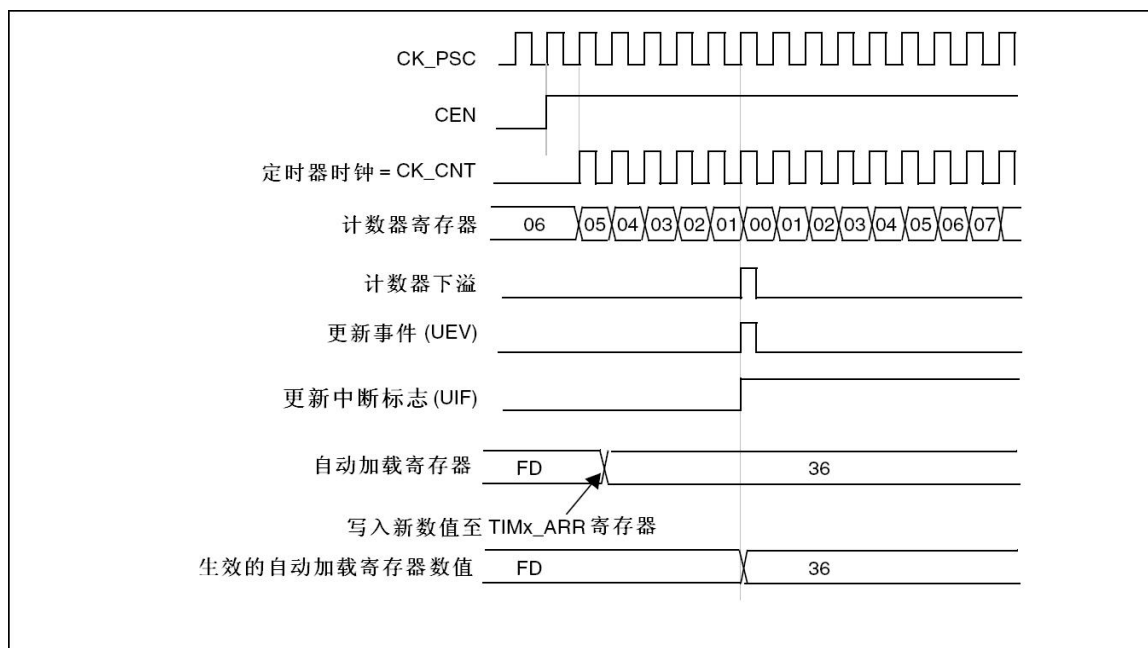
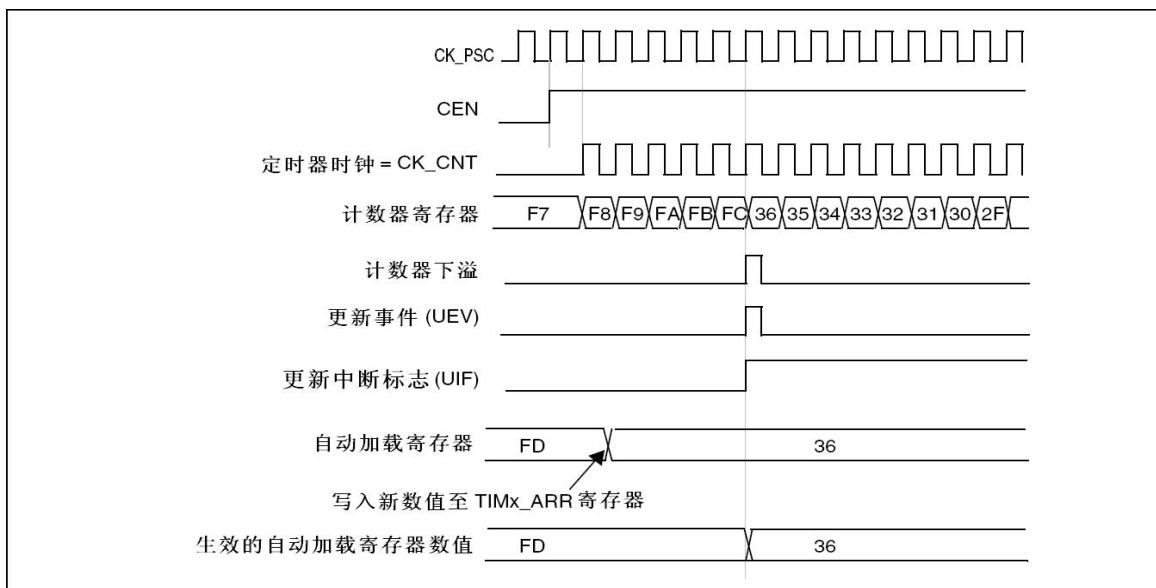


图69 计数器时序图，ARPE=1时的更新事件(计数器溢出)



11.3.3 重复计数器

13.3.1节“时基单元”解释了计数器上溢/下溢时更新事件(UEV)是如何产生的，然而事实上它只能在重复计数达到0的时候产生。这个特性对产生PWM信号非常有用。

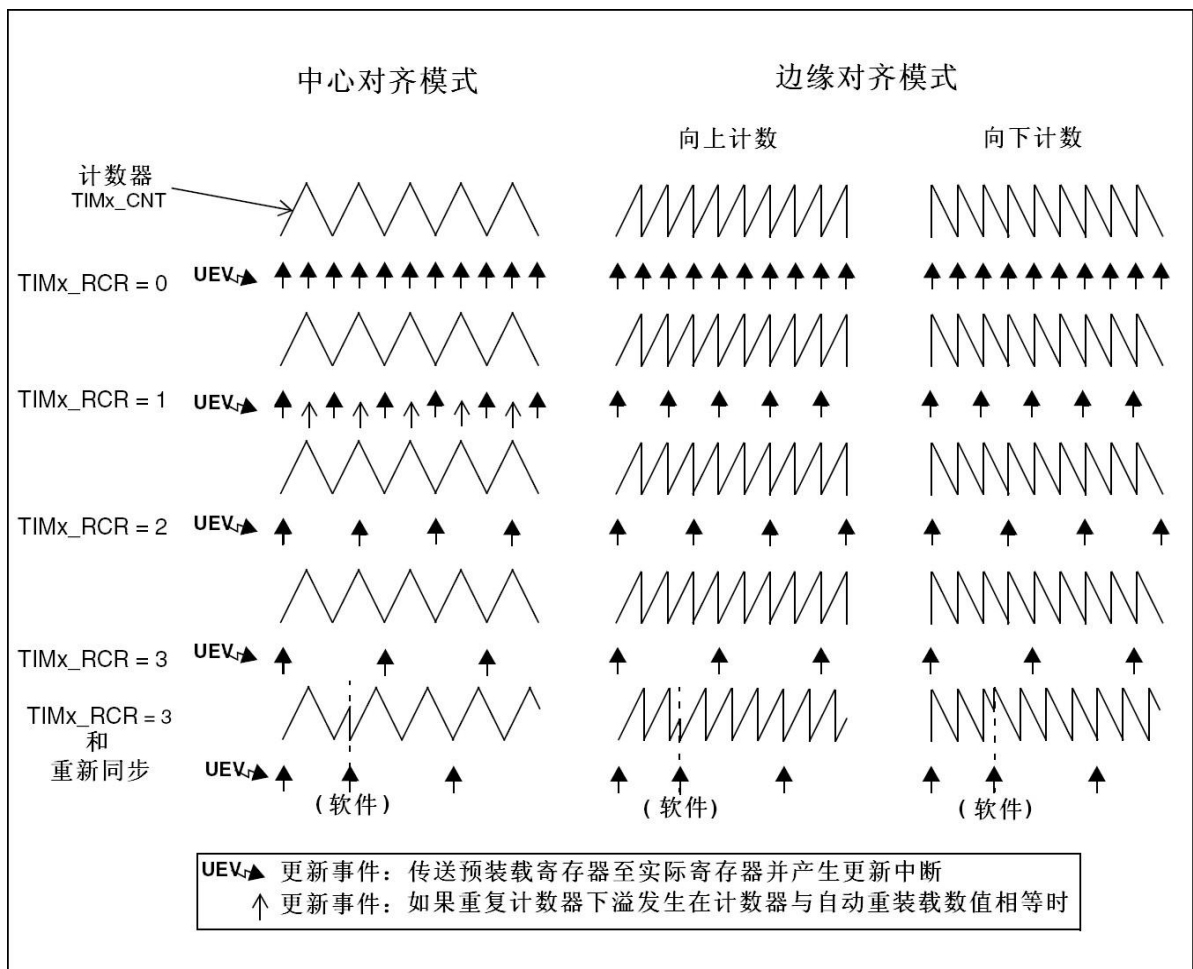
这意味着在每N次计数上溢或下溢时，数据从预装载寄存器传输到影子寄存器(TIMx_ARRL自动重载寄存器，TIMx_PSCR预装载寄存器，还有在比较模式下的捕获/比较寄存器TIMx_CCRx)，N是TIMx_RCNTR重复计数寄存器中的值。

重复计数器在下述任一条件成立时递减：

- 向上计数模式下每次计数器溢出时，
- 向下计数模式下每次计数器下溢时，
- 中央对齐模式下每次上溢和每次下溢时。虽然这样限制了PWM的最大循环周期为128，但它能够在每个PWM周期2次更新占空比。在中央对齐模式下，因为波形是对称的，如果每个PWM周期中仅刷新一次比较寄存器，则最大的分辨率为 $2 \times T_{ck}$ 。

重复计数器是自动加载的，重复速率是由TIMx_RCNTR寄存器的值定义(参看图70)。当更新事件由软件产生(通过设置TIMx_EVTGR中的UG位)或者通过硬件的从模式控制器产生，则无论重复计数器的值是多少，立即发生更新事件，并且TIMx_RCNTR寄存器中的内容被重载入到重复计数器。

图70 不同模式下更新速率的例子，及TIMx_RCNTR的寄存器设置



11.3.4 时钟选择

计数器时钟可由下列时钟源提供:

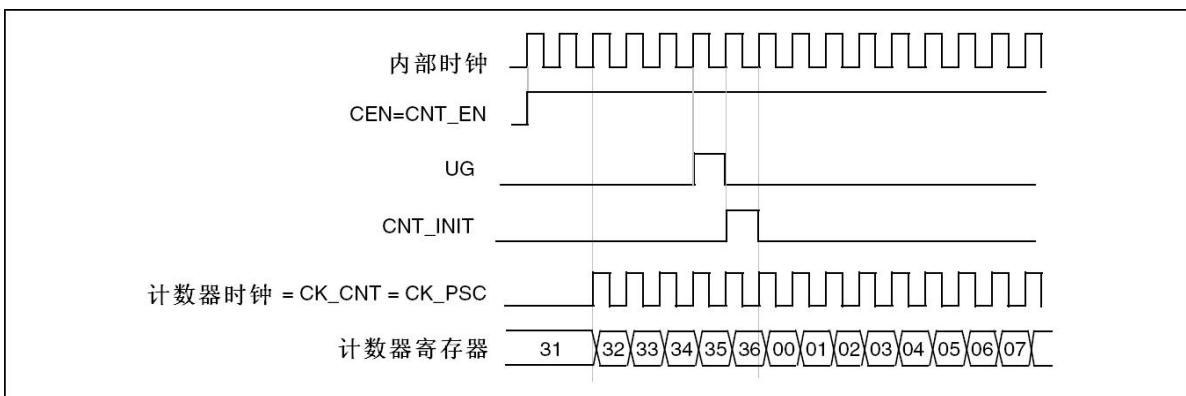
- 内部时钟(CK_INT)
- 外部时钟模式1: 外部输入引脚
- 外部时钟模式2: 外部触发输入ETR
- 内部触发输入(ITRx): 使用一个定时器作为另一个定时器的预分频器。如可以配置一个定时器Timer1而作为另一个定时器Timer2的预分频器。详见下一章。

内部时钟源(CK_INT)

如果禁止了从模式控制器(SMS=000), 则CEN、DIR(TIMx_CTLR1 寄存器) 和UG 位(TIMx_EVTGR 寄存器)是事实上的控制位, 并且只能被软件修改(UG位仍被自动清除)。只要CEN位被写成'1', 预分频器的时钟就由内部时钟CK_INT提供。

下图显示控制电路和向上计数器在一般模式下, 不带预分频器时的操作。

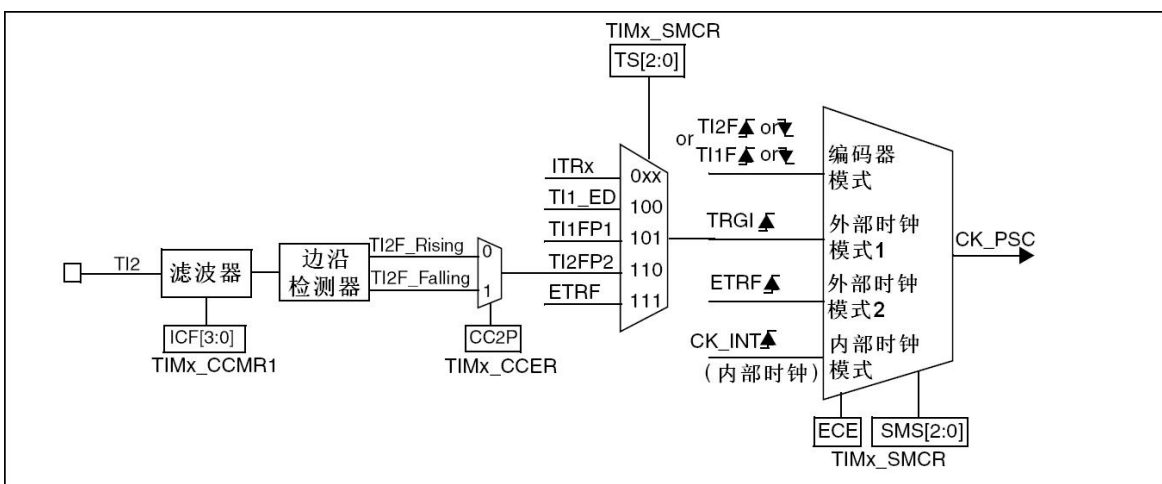
图71 一般模式下的控制电路, 内部时钟分频因子为1



外部时钟源模式1

当TIMx_SLVCTLRL寄存器的SMS=111时, 此模式被选中。计数器可以在选定输入端的每个上升沿或下降沿计数。

图72 TI2外部时钟连接例子



例如, 要配置向上计数器在TI2输入端的上升沿计数, 使用下列步骤:

配置TIMx_CCMR1寄存器CC2S=01, 配置通道2检测TI2输入的上升沿

配置TIMx_CCMR1寄存器的IC2F[3:0], 选择输入滤波器带宽(如果不需要滤波器, 保持IC2F=0000)

配置TIMx_CCER寄存器的CC2P=0，选定上升沿极性

配置TIMx_SLVCTL寄存器中的SMS=111，选择定时器外部时钟模式1

配置TIMx_SLVCTL寄存器中的TS=110，选定TI2作为触发输入源

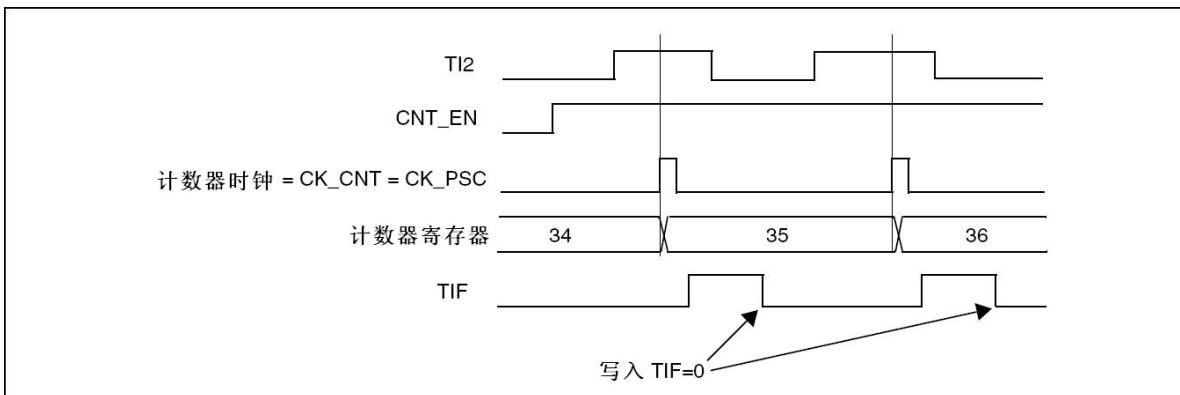
设置TIMx_CTLR1寄存器的CEN=1，启动计数器

注： 捕获预分频器不用作触发，所以不需要对它进行配置

当上升沿出现在TI2，计数器计数一次，且TIF标志被设置。

在TI2的上升沿和计数器实际时钟之间的延时，取决于在TI2输入端的重新同步电路。

图73 外部时钟模式1下的控制电路



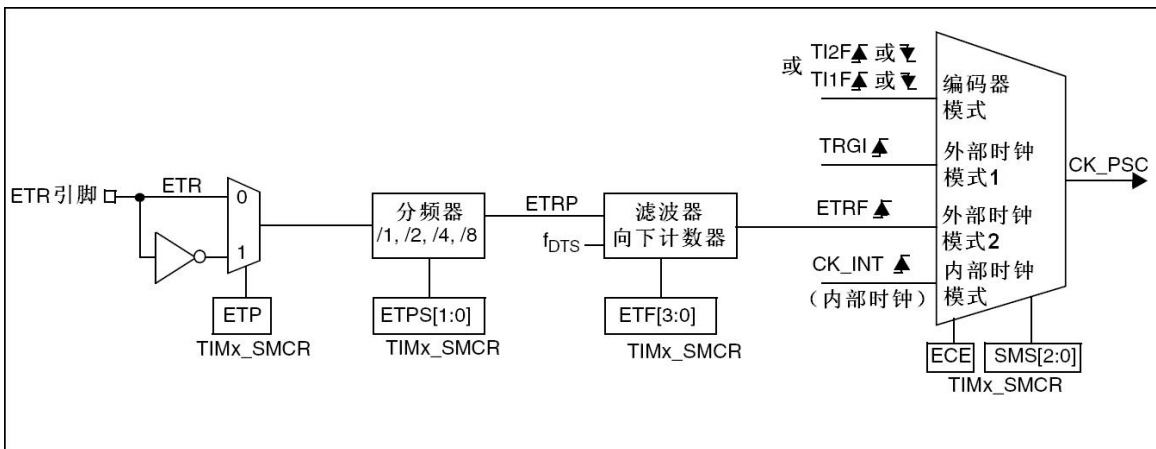
外部时钟源模式2

选定此模式的方法为：令TIMx_SLVCTL寄存器中的ECE=1

计数器能够在外部触发ETR的每一个上升沿或下降沿计数。

下图是外部触发输入的框图

图74 外部触发输入框图



例如，要配置在ETR下每2个上升沿计数一次的向上计数器，使用下列步骤：

本例中不需要滤波器，置TIMx_SLVCTL寄存器中的ETF[3:0]=0000

设置预分频器，置TIMx_SLVCTL寄存器中的ETPS[1:0]=01

选择ETR的上升沿检测，置TIMx_SLVCTL寄存器中的ETP=0

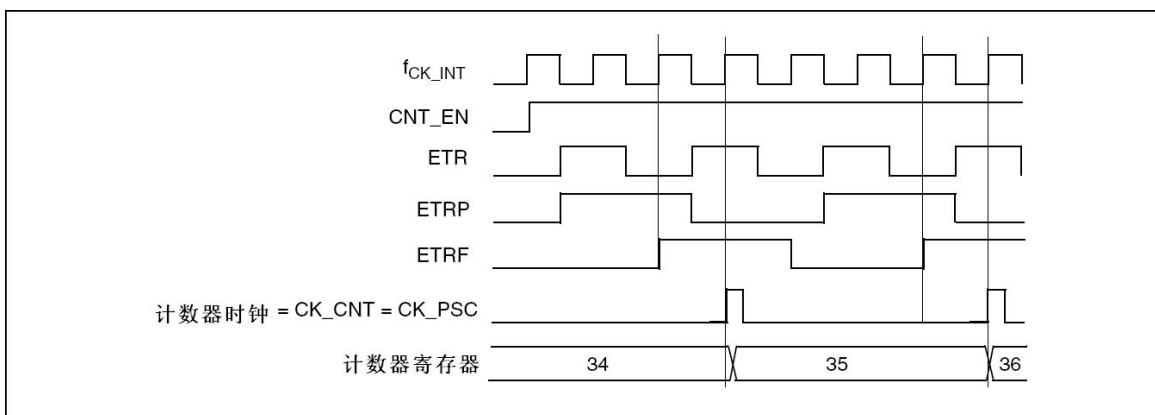
开启外部时钟模式2，写TIMx_SLVCTL寄存器中的ECE=1

启动计数器，写TIMx_CTLR1寄存器中的CEN=1

计数器在每2个ETR上升沿计数一次。

在ETR的上升沿和计数器实际时钟之间的延时取决于在ETRP信号端的重新同步电路。

图74 外部时钟模式2下的控制电路



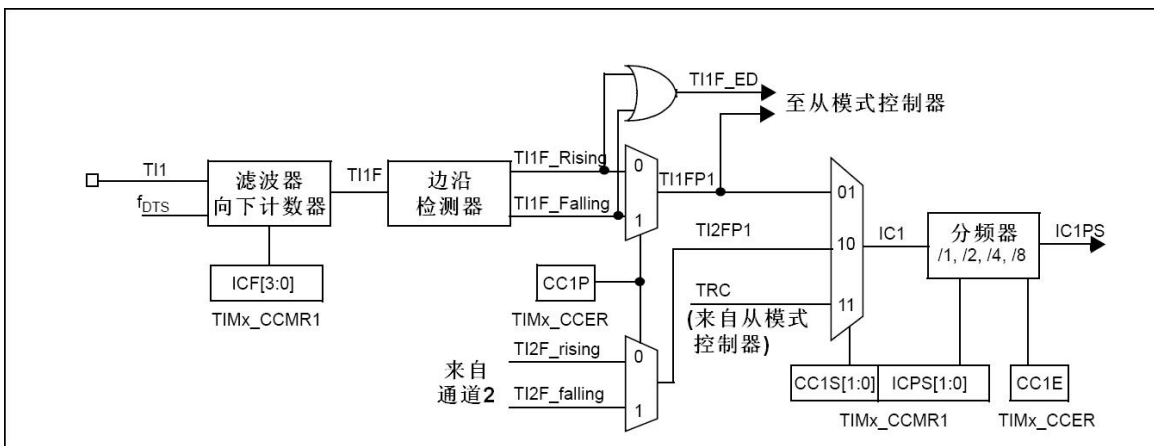
11.3.5 捕获/比较通道

每一个捕获/比较通道都是围绕着一个捕获/比较寄存器(包含影子寄存器)，包括捕获的输入部分(数字滤波、多路复用和预分频器)，和输出部分(比较器和输出控制)。

图76至图79是一个捕获/比较通道概览。

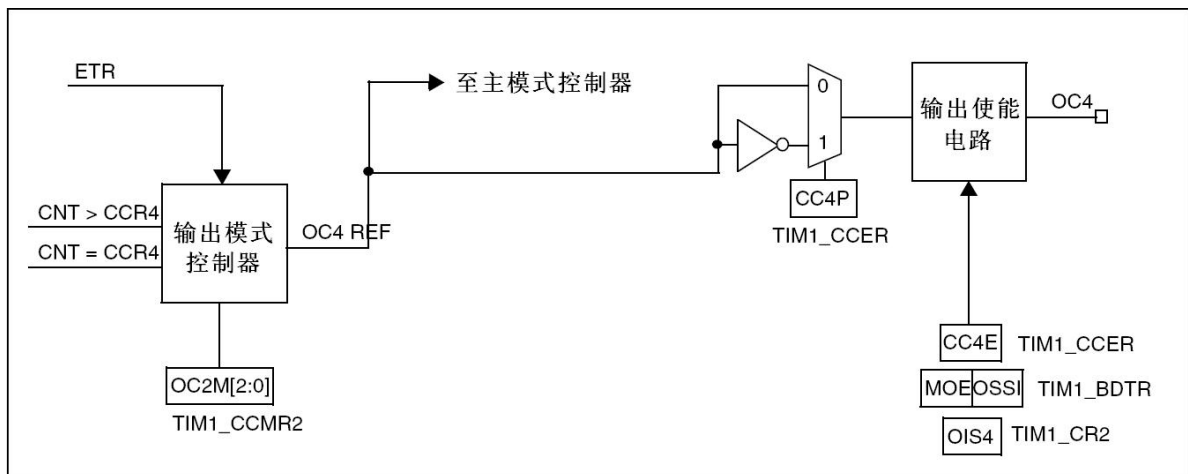
输入部分对相应的TIMx输入信号采样，并产生一个滤波后的信号TIMx_F。然后，一个带极性选择的边缘监测器产生一个信号(TIMx_FPx)，它可以作为从模式控制器的输入触发或者作为捕获控制。该信号通过预分频进入捕获寄存器(ICxPS)。

图76 捕获/比较通道(如：通道1输入部分)



输出部分产生一个中间波形OCxRef(高有效)作为基准，链的末端决定最终输出信号的极性。

图79 捕获/比较通道的输出部分(通道4)



捕获/比较模块由一个预装载寄存器和一个影子寄存器组成。读写过程仅操作预装载寄存器。

在捕获模式下，捕获发生在影子寄存器上，然后再复制到预装载寄存器中。

在比较模式下，预装载寄存器的内容被复制到影子寄存器中，然后影子寄存器的内容和计数器进行比较。

11.3.6 输入捕获模式

在输入捕获模式下，当检测到ICx信号上相应的边沿后，计数器的当前值被锁存到捕获/比较寄存器(TIMx_CCRx)中。当发生捕获事件时，相应的CCxIF标志(TIMx_SR寄存器)被置1，如果开放了中断或者DMA操作，则将产生中断或者DMA请求。如果发生捕获事件时CCxIF标志已经为高，那么重复捕获标志CCxOF(TIMx_SR寄存器)被置1。写CCxIF=0可清除CCxIF，或读取存储在TIMx_CCRx寄存器中的捕获数据也可清除CCxIF。写CCxOF=0可清除CCxOF。

以下例子说明如何在TI1输入的上升沿时捕获计数器的值到TIMx_CCR1寄存器中，步骤如下：

- 选择有效输入端：TIMx_CCR1必须连接到TI1输入，所以写入TIMx_CCR1寄存器中的CC1S=01，只要CC1S不为'00'，通道被配置为输入，并且TIMx_CCR1寄存器变为只读。
- 根据输入信号的特点，配置输入滤波器为所需的带宽(即输入为TIx时，输入滤波器控制位是TIMx_CCMRx寄存器中的ICxF位)。假设输入信号在最多5个内部时钟周期的时间内抖动，我们须配置滤波器的带宽长于5个时钟周期；因此我们可以(以f_{DTs}频率)连续采样8次，以确认在TI1上一次真实的边沿变换，即在TIMx_CCMR1寄存器中写入IC1F=0011。
- 选择TI1通道的有效转换边沿，在TIMx_CCER寄存器中写入CC1P=0(上升沿)。
- 配置输入预分频器。在本例中，我们希望捕获发生在每一个有效的电平转换时刻，因此预分频器被禁止(写TIMx_CCMR1寄存器的IC1PS=00)。
- 设置TIMx_CCER寄存器的CC1E=1，允许捕获计数器的值到捕获寄存器中。
- 如果需要，通过设置TIMx_DIER寄存器中的CC1IE位允许相关中断请求，通过设置TIMx_DIER寄存器中的CC1DE位允许

DMA请求。当发生一个输入捕获时：

- 产生有效的电平转换时，计数器的值被传送到TIMx_CCR1寄存器。
- CC1IF标志被设置(中断标志)。当发生至少2个连续的捕获时，而CC1IF未曾被清除，CC1OF也被置1。
- 如设置了CC1IE位，则会产生一个中断。
- 如设置了CC1DE位，则还会产生一个DMA请求。

为了处理捕获溢出，建议在读出捕获溢出标志之前读取数据，这是为了避免丢失在读出捕获溢出标志之后和读取数据之前可能产生的捕获溢出信息。

注：设置TIMx_EVTGR寄存器中相应的CCxG位，可以通过软件产生输入捕获中断和/或DMA请求。

11.3.7 PWM输入模式

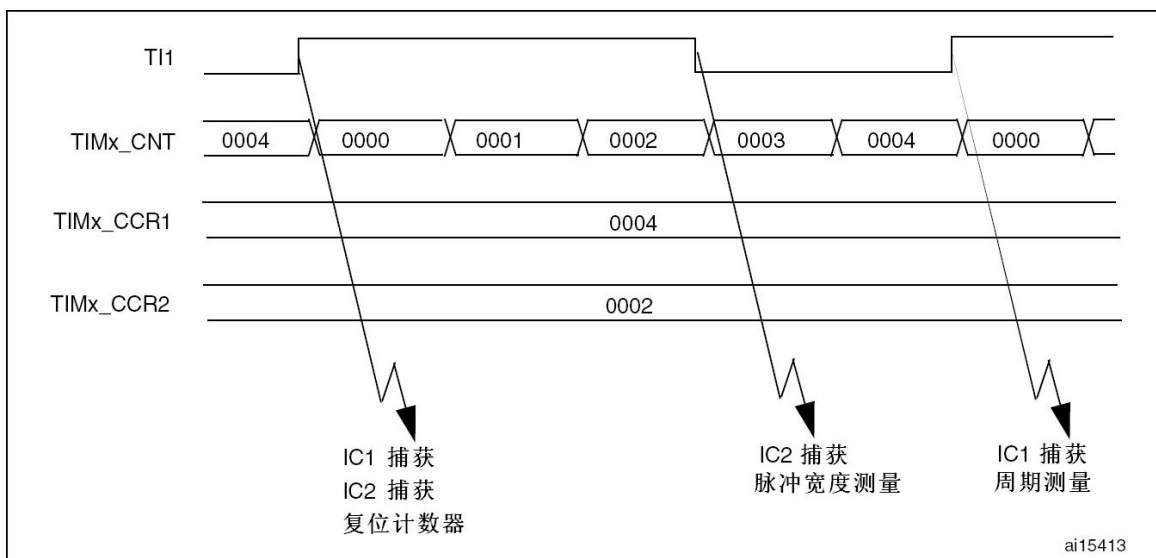
该模式是输入捕获模式的一个特例，除下列区别外，操作与输入捕获模式相同：

- 两个ICx信号被映射至同一个TIx输入。
- 这2个ICx信号为边沿有效，但是极性相反。
- 其中一个TIxFP信号被作为触发输入信号，而从模式控制器被配置成复位模式。

例如，你需要测量输入到TI1上的PWM信号的长度(TIMx_CCR1寄存器)和占空比(TIMx_CCR2寄存器)，具体步骤如下(取决于CK_INT的频率和预分频器的值)

- 选择TIMx_CCR1的有效输入：置TIMx_CCMR1寄存器的CC1S=01(选中TI1)。
- 选择TI1FP1的有效极性(用来捕获数据到TIMx_CCR1中和清除计数器)：置CC1P=0(上升沿有效)。
- 选择TIMx_CCR2的有效输入：置TIMx_CCMR1寄存器的CC2S=10(选中TI1)。
- 选择TI1FP2的有效极性(捕获数据到TIMx_CCR2)：置CC2P=1(下降沿有效)。
- 选择有效的触发输入信号：置TIMx_SLVCTL寄存器中的TS=101(选择TI1FP1)。
- 配置从模式控制器为复位模式：置TIMx_SLVCTL中的SMS=100。
- 使能捕获：置TIMx_CCER寄存器中CC1E=1且CC2E=1。

图80 PWM输入模式时序



因为只有TI1FP1和TI2FP2连到了从模式控制器，所以PWM输入模式只能使用TIMx_CH1

/TIMx_CH2信号。

11.3.8 强置输出模式

在输出模式(TIMx_CCMRx寄存器中CCxS=00)下，输出比较信号(OCxREF和相应的OCx/OCxN)能够直接由软件强置为有效或无效状态，而不依赖于输出比较寄存器和计数器间的比较结果。

置TIMx_CCMRx寄存器中相应的OCxM=101，即可强置输出比较信号(OCxREF/OCx)为有效状态。这样OCxREF被强置为高电平(OCxREF始终为高电平有效)，同时OCx得到

CCxP极性相反的信号。

例如: CCxP=0(OCx高电平有效), 则OCx被强置为高电平。

置TIMx_CCMRx寄存器中的OCxM=100, 可强置OCxREF信号为低。

该模式下, 在TIMx_CCRx影子寄存器和计数器之间的比较仍然在进行, 相应的标志也会被修改。因此仍然会产生相应的中断和DMA请求。这将会在下面的输出比较模式一节中介绍。

11.3.9 输出比较模式

此项功能是用来控制一个输出波形, 或者指示一段给定的时间已经到时。当计数器与捕获/比较寄存器的内容相同时, 输出比较功能做如下操作:

- 将输出比较模式(TIMx_CCMRx寄存器中的OCxM位)和输出极性(TIMx_CCER寄存器中的CCxP位)定义的值输出到对应的引脚上。在比较匹配时, 输出引脚可以保持它的电平(OCxM=000)、被设置成有效电平(OCxM=001)、被设置成无效电平(OCxM=010)或进行翻转(OCxM=011)。
- 设置中断状态寄存器中的标志位(TIMx_SR寄存器中的CCxIF位)。
- 若设置了相应的中断屏蔽(TIMx_DIER寄存器中的CCxIE位), 则产生一个中断。
- 若设置了相应的使能位(TIMx_DIER寄存器中的CCxDE位, TIMx_CTLR2寄存器中的CCDS位选择DMA请求功能), 则产生一个DMA请求。

TIMx_CCMRx中的OCxPE位选择TIMx_CCRx寄存器是否需要使用预装载寄存器。

在输出比较模式下, 更新事件UEV对OCxREF和OCx输出没有影响。

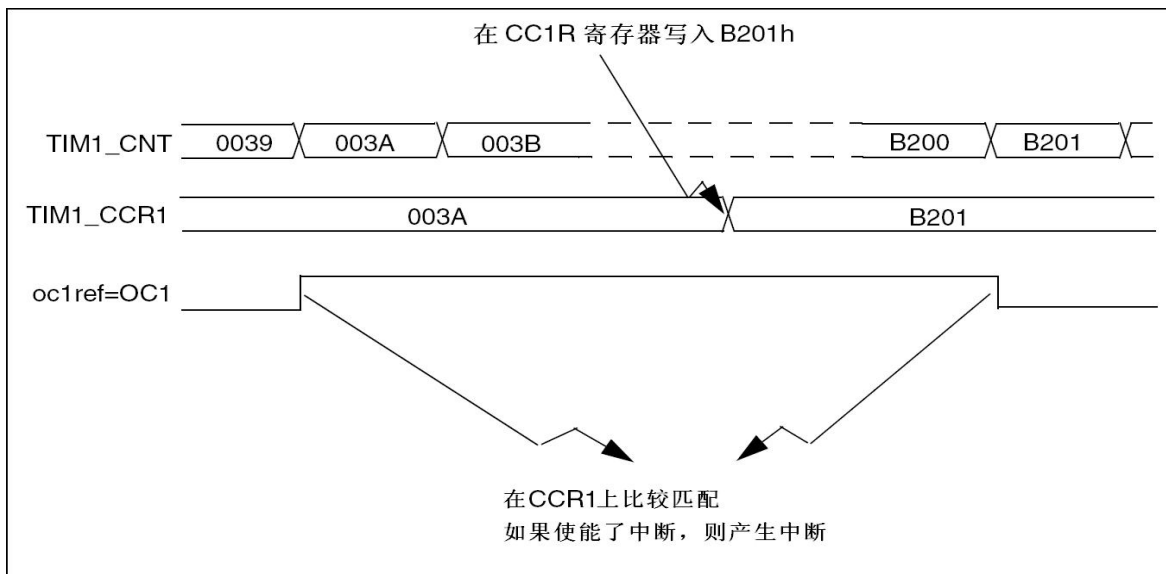
同步的精度可以达到计数器的一个计数周期。输出比较模式(在单脉冲模式下)也能用来输出一个单脉冲。

输出比较模式的配置步骤:

1. 选择计数器时钟(内部, 外部, 预分频器)。
2. 将相应的数据写入TIMx_ARRLR和TIMx_CCRx寄存器中。
3. 如果要产生一个中断请求, 设置CCxIE位。
4. 选择输出模式, 例如:
 - 要求计数器与CCRx匹配时翻转OCx的输出引脚, 设置OCxM=011
 - 置OCxPE = 0禁用预装载寄存器
 - 置CCxP = 0选择极性为高电平有效
 - 置CCxE = 1使能输出
5. 设置TIMx_CTLR1寄存器的CEN位启动计数器

TIMx_CCRx寄存器能够在任何时候通过软件进行更新以控制输出波形, 条件是未使用预装载寄存器(OCxPE='0', 否则TIMx_CCRx的影子寄存器只能在发生下一次更新事件时被更新)。下图给出了一个例子。

图81 输出比较模式，翻转OC1



11.3.10 PWM模式

脉冲宽度调制模式可以产生一个由TIMx_ARLR寄存器确定频率、由TIMx_CCRx寄存器确定占空比的信号。

在TIMx_CCMRx寄存器中的OCxM位写入'110'(PWM模式1)或'111'(PWM模式2)，能够独立地设置每个OCx输出通道产生一路PWM。必须通过设置TIMx_CCMRx寄存器的OCxPE位使能相应的预装载寄存器，最后还要设置TIMx_CTLR1寄存器的ARPE位，(在向上计数或中心对称模式中)使能自动重载的预装载寄存器。

仅当发生一个更新事件的时候，预装载寄存器才能被传送到影子寄存器，因此在计数器开始计数之前，必须通过设置TIMx_EVTGR寄存器中的UG位来初始化所有的寄存器。

OCx的极性可以通过软件在TIMx_CCER寄存器中的CCxP位设置，它可以设置为高电平有效或低电平有效。OCx的输出使能通过(TIMx_CCER和TIMx_BDTR寄存器中)CCxE、CCxNE、MOE、OSSI和OSSR位的组合控制。详见TIMx_CCER寄存器的描述。

在PWM模式(模式1或模式2)下，TIMx_CNTR和TIMx_CCRx始终在进行比较，(依据计数器的计数方向)以确定是否符合TIMx_CCRx≤TIMx_CNTR或者TIMx_CNTR≤TIMx_CCRx。

根据TIMx_CTLR1寄存器中CMS位的状态，定时器能够产生边沿对齐的PWM信号或中央对齐的PWM信号。

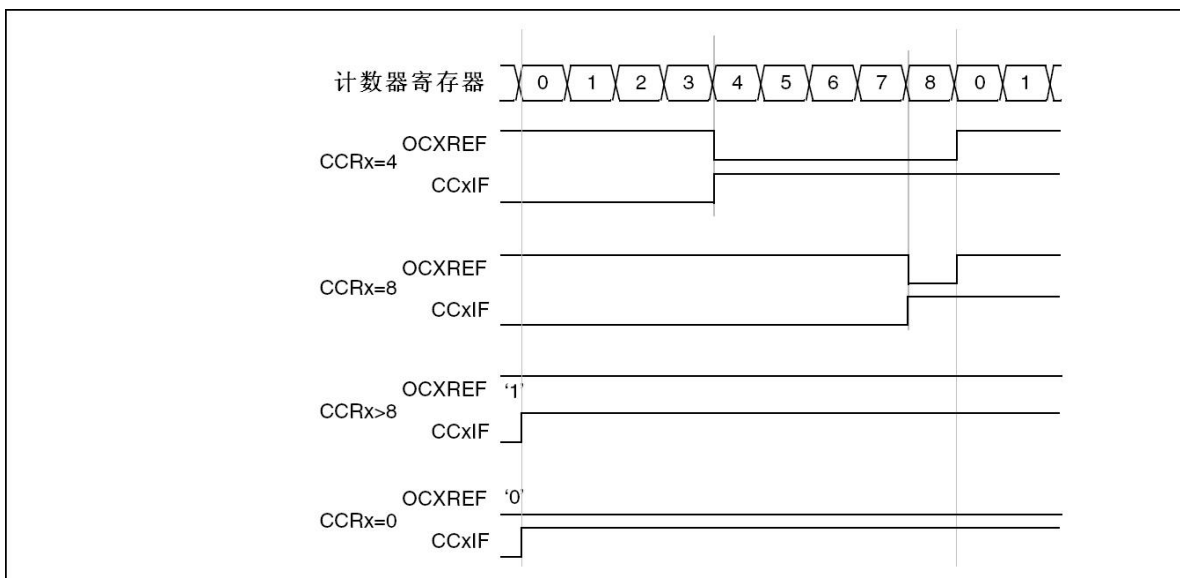
PWM 边沿对齐模式

- 向上计数配置

当TIMx_CTLR1寄存器中的DIR位为低的时候执行向上计数。参看13.3.2节。

下面是一个PWM模式1的例子。当TIMx_CNTR<TIMx_CCRx时，PWM参考信号OCxREF为高，否则为低。如果TIMx_CCRx中的比较值大于自动重载值(TIMx_ARLR)，则OCxREF保持为'1'。如果比较值为0，则OCxREF保持为'0'。下图为TIMx_ARLR=8时边沿对齐的PWM波形实例。

图82 边沿对齐的PWM波形(ARR=8)



- 向下计数的配置

当TIMx_CTLR1寄存器的DIR位为高时执行向下计数。参看13.3.2节。

在PWM模式1，当TIMx_CNTR>TIMx_CCRx时参考信号OCxREF为低，否则为高。如果TIMx_CCRx中的比较值大于TIMx_ARLR中的自动重装载值，则OCxREF保持为'1'。该模式下不能产生0%的PWM波形。

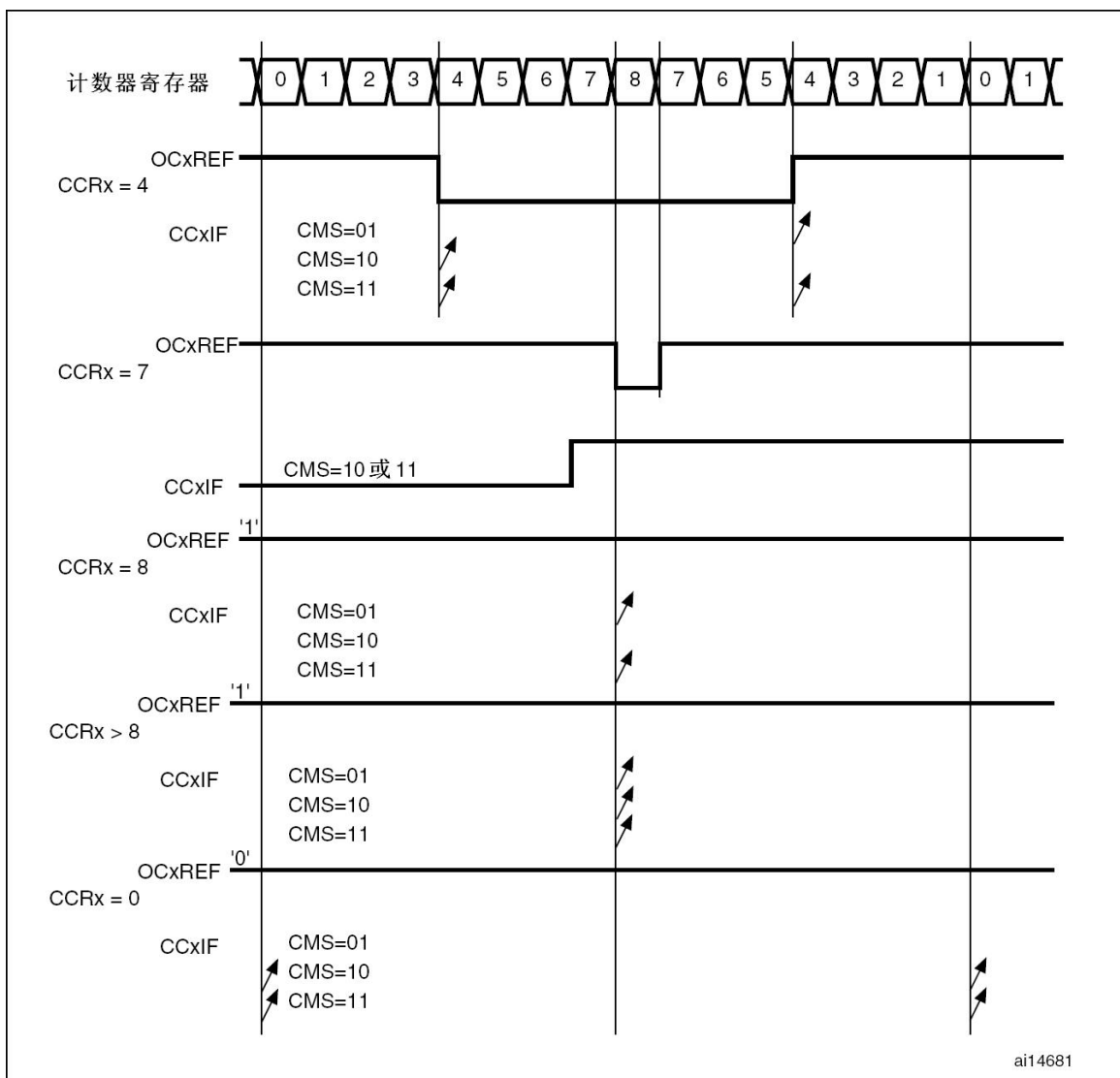
PWM 中央对齐模式

当TIMx_CTLR1寄存器中的CMS位不为'00'时为中央对齐模式(所有其他的配置对OCxREF/OCx信号都有相同的作用)。根据不同的CMS位设置，比较标志可以在计数器向上计数时被置1、在计数器向下计数时被置1、或在计数器向上和向下计数时被置1。TIMx_CTLR1寄存器中的计数方向位(DIR)由硬件更新，不要用软件修改它。参看13.3.2节的中央对齐模式。

下图给出了一些中央对齐的PWM波形的例子

- TIMx_ARLR=8
- PWM模式1
- TIMx_CTLR1寄存器的CMS=01，在中央对齐模式1下，当计数器向下计数时设置比较标志。

图83 中央对齐的PWM波形(APR=8)



使用中央对齐模式的提示:

- 进入中央对齐模式时，使用当前的向上/向下计数配置；这就意味着计数器向上还是向下计数取决于TIM_x_CTLR1寄存器中DIR位的当前值。此外，软件不能同时修改DIR和CMS位。
- 不推荐当运行在中央对齐模式时改写计数器，因为这会产生不可预知的结果。特别地：
 - 如果写入计数器的值大于自动重加载的值(TIM_x_CNTR>TIM_x_ARLR)，则方向不会被更新。例如，如果计数器正在向上计数，它就会继续向上计数。
 - 如果将0或者TIM_x_ARLR的值写入计数器，方向被更新，但不产生更新事件UEV。

11.3.11 互补输出和死区插入

高级控制定时器(TIM1和TIM8)能够输出两路互补信号，并且能够管理输出的瞬时关断和接通。这段时间通常被称为死区，用户应该根据连接的输出器件和它们的特性(电平转换的延时、电源开关的延时等)来调整死区时间。

配置TIM_x_CCER寄存器中的CCxP和CCxNP位，可以为每一个输出独立地选择极性(主输出OCx或互补输出OCxN)。

互补信号OCx和OCxN通过下列控制位的组合进行控制：TIM_x_CCER寄存器的CCxE和

CCxNE 位，TIMx_BDTR和TIMx_CTLR2寄存器中的MOE、OISx、OISxN、OSSI和OSSR位，详见表75 带刹车功能的互补输出通道OCx和OCxN的控制位。特别的是，在转换到IDLE状态时(MOE下降到0)死区被激活。

同时设置CCxE和CCxNE位将插入死区，如果存在刹车电路，则还要设置MOE位。每一个通道都有一个10位的死区发生器。参考信号OCxREF可以产生2路输出OCx和OCxN。

如果OCx和OCxN为高有效：

- OCx输出信号与参考信号相同，只是它的上升沿相对于参考信号的上升沿有一个延迟。
- OCxN输出信号与参考信号相反，只是它的上升沿相对于参考信号的下降沿有一个延迟。

如果延迟大于当前有效的输出宽度(OCx或者OCxN)，则不会产生相应的脉冲。

下列几张图显示了死区发生器的输出信号和当前参考信号OCxREF 之间的关系。(假设CCxP=0、CCxNP=0、MOE=1、CCxE=1并且CCxNE=1)

图84 带死区插入的互补输出

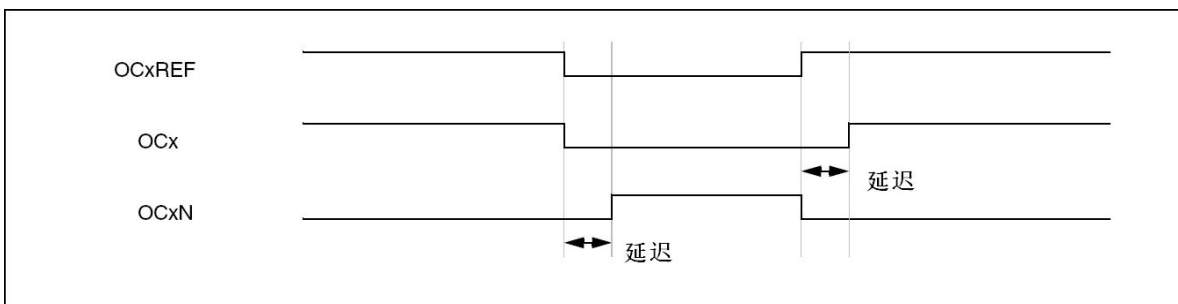


图85 死区波形延迟大于负脉冲

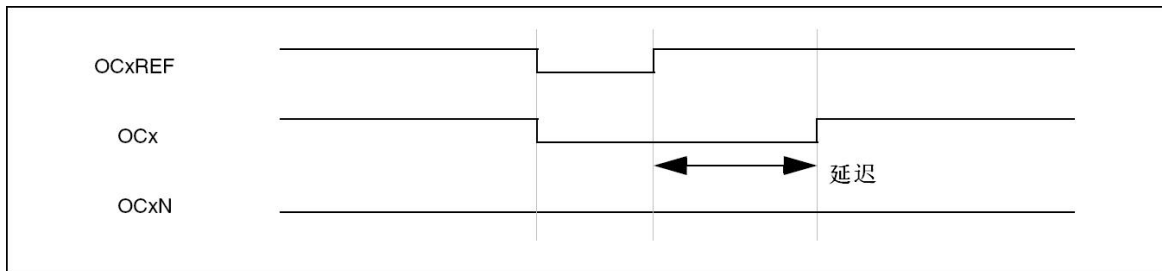
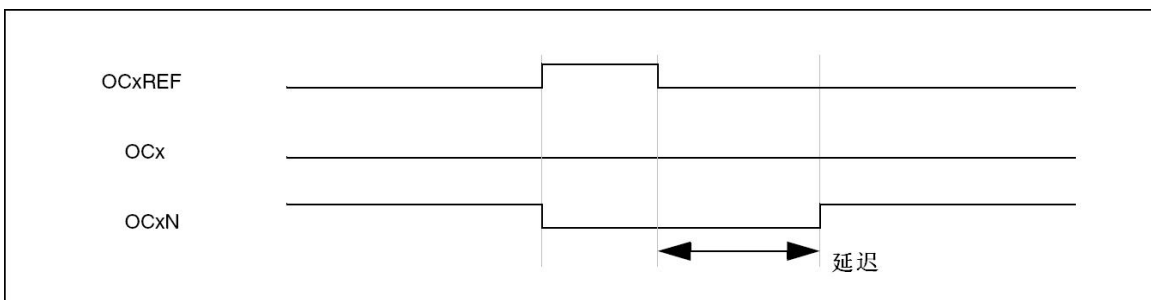


图86 死区波形延迟大于正脉冲



每一个通道的死区延时都是相同的，是由TIMx_BDTR 寄存器中的DTG 位编程配置。详见13.4.18节TIM1刹车和死区寄存器(TIMx_BDTR)中的延时计算。

重定向OCxREF到OCx或OCxN

在输出模式下(强置、输出比较或PWM)，通过配置TIMx_CCER寄存器的CCxE和CCxNE位，OCxREF可以被重定向到OCx或者OCxN的输出。

这个功能可以在互补输出处于无效电平时，在某个输出上送出一个特殊的波形(例如PWM或者静态有效电平)。另一个作用是，让两个输出同时处于无效电平，或处于有效电平和带死区的互补输出。

注: 当只使能OCxN(CCxE=0, CCxNE=1)时, 它不会反相, 当OCxREF有效时立即变高。例如, 如果CCxNP=0, 则OCxN=OCxREF。另一方面, 当OCx和OCxN都被使能时(CCxE=CCxNE=1), 当OCxREF为高时OCx有效; 而OCxN相反, 当OCxREF低时OCxN变为有效。

11.3.12 使用刹车功能

当使用刹车功能时, 依据相应的控制位(TIMx_BDTR寄存器中的MOE、OSSI和OSSR位, TIMx_CTLR2寄存器中的OISx和OISxN位), 输出使能信号和无效电平都会被修改。但无论何时, OCx和OCxN输出不能在同一时间同时处于有效电平上。详见表75带刹车功能的互补输出通道OCx和OCxN的控制位。

刹车源既可以是刹车输入引脚又可以是一个时钟失败事件。时钟失败事件由复位时钟控制器中的时钟安全系统产生, 详见6.2.7节时钟安全系统(CSS)。

系统复位后, 刹车电路被禁止, MOE位为低。设置TIMx_BDTR寄存器中的BKE位可以使能刹车功能, 刹车输入信号的极性可以通过配置同一个寄存器中的BKP位选择。BKE和BKP可以同时被修改。当写入BKE和BKP位时, 在真正写入之前会有1个APB时钟周期的延迟, 因此需要等待一个APB时钟周期之后, 才能正确地读回写入的位。

因为MOE下降沿可以是异步的, 在实际信号(作用在输出端)和同步控制位(在TIMx_BDTR寄存器中)之间设置了一个再同步电路。这个再同步电路会在异步信号和同步信号之间产生延迟。特别的, 如果当它为低时写MOE=1, 则读出它之前必须先插入一个延时(空指令)才能读到正确的值。这是因为写入的是异步信号而读的是同步信号。

当发生刹车时(在刹车输入端出现选定的电平), 有下述动作:

- MOE位被异步地清除, 将输出置于无效状态、空闲状态或者复位状态(由OSSI位选择)。这个特性在MCU的振荡器关闭时依然有效。
- 一旦MOE=0, 每一个输出通道输出由TIMx_CTLR2寄存器中的OISx位设定的电平。如果OSSI=0, 则定时器释放使能输出, 否则使能输出始终为高。
- 当使用互补输出时:
 - 输出首先被置于复位状态即无效的状态(取决于极性)。这是异步操作, 即使定时器没有时钟时, 此功能也有效。
 - 如果定时器的时钟依然存在, 死区生成器将会重新生效, 在死区之后根据OISx和OISxN位指示的电平驱动输出端口。即使在这种情况下, OCx和OCxN也不能被同时驱动到有效的电平。注, 因为重新同步MOE, 死区时间比通常情况下长一些(大约2个ck_tim的时钟周期)。
 - 如果OSSI=0, 定时器释放使能输出, 否则保持使能输出; 或一旦CCxE与CCxNE之一变高时, 使能输出变为高。
- 如果设置了TIMx_DIER寄存器中的BIE位, 当刹车状态标志(TIMx_SR寄存器中的BIF位)为'1'时, 则产生一个中断。如果设置了TIMx_DIER寄存器中的BDE位, 则产生一个DMA请求。
- 如果设置了TIMx_BDTR寄存器中的AOE位, 在下一个更新事件UEV时MOE位被自动置位; 例如, 这可以用来进行整形。否则, MOE始终保持低直到被再次置'1'; 此时, 这个特性可以被用在安全方面, 你可以把刹车输入连到电源驱动的报警输出、热敏传感器或者其他安全器件上。

注: 刹车输入为电平有效。所以, 当刹车输入有效时, 不能同时(自动地或者通过软件)设置MOE。同时, 状态标志BIF不能被清除。

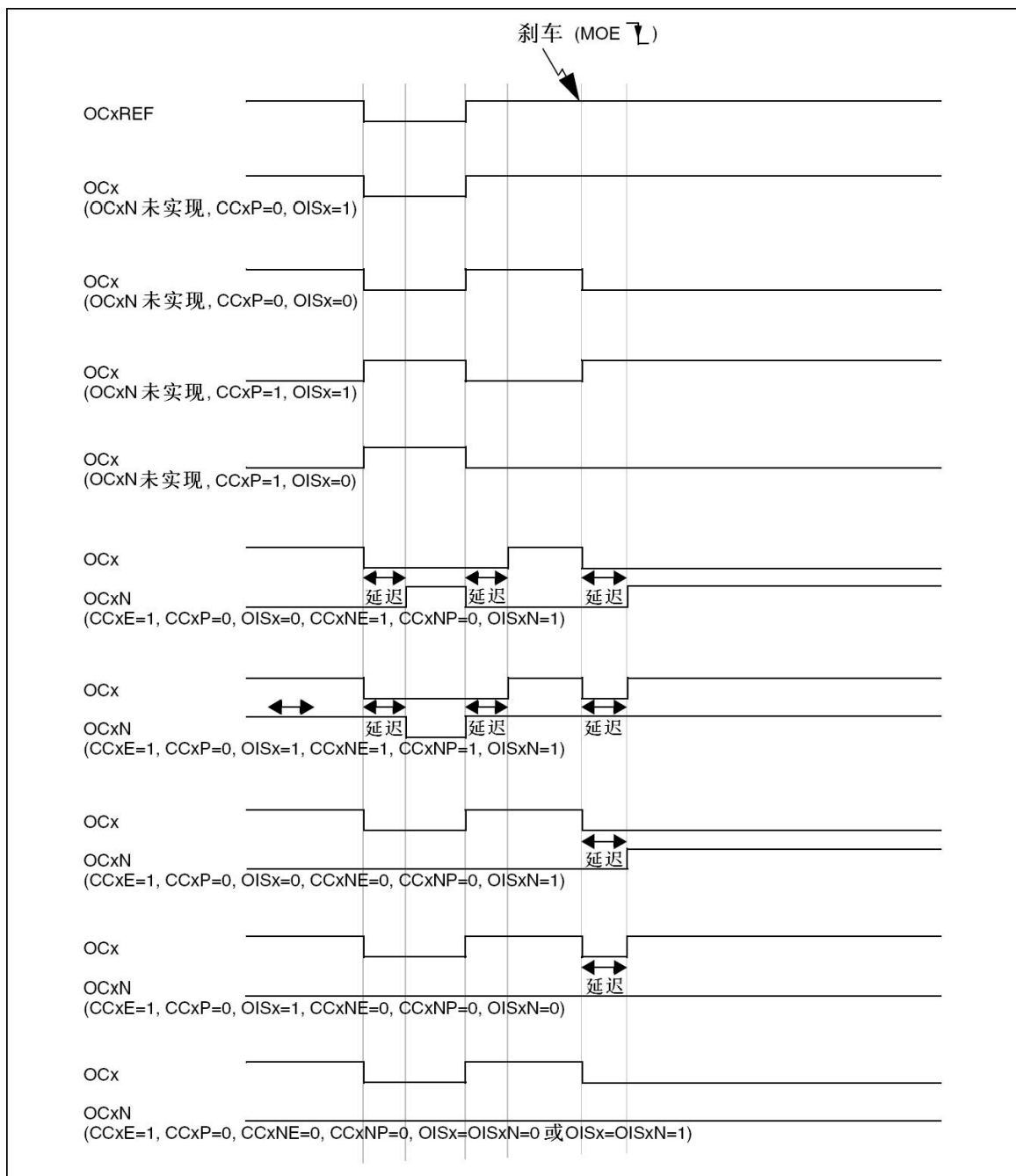
刹车由BRK输入产生, 它的有效极性是可编程的, 且由TIMx_BDTR寄存器中的BKE位开启。

除了刹车输入和输出管理, 刹车电路中还实现了写保护以保证应用程序的安全。它允许用户冻结几个配置参数(死区长度, OCx/OCxN极性和被禁止的状态, OCxM配置,

刹车使能和极性)。用户可以通过TIMx_BDTR寄存器中的LOCK位，从三级保护中选择一种，参看11.4.18节TIM1 刹车和死区寄存器(TIMx_BDTR)。在MCU复位后LOCK位只能被修改一次。

下图显示响应刹车的输出实例。

图87 响应刹车的输出



11.3.13 在外部事件时清除OCxREF信号

对于一个给定的通道，设置TIMx_CCMRx寄存器中对应的OCxCE位为'1'，能够用ETRF输入端的高电平把OCxREF信号拉低，OCxREF信号将保持为低直到发生下一次的更新事件UEV。

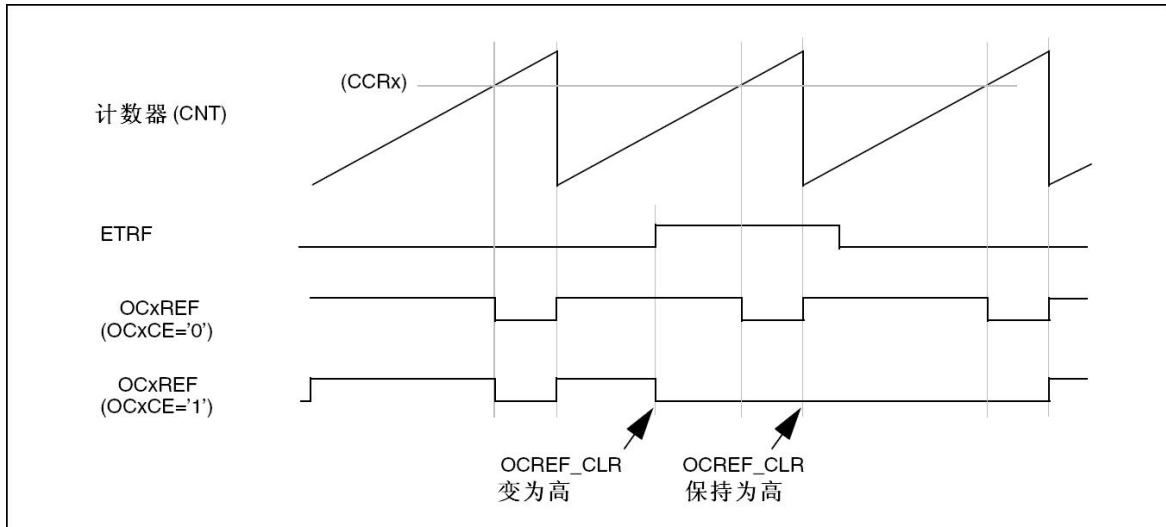
该功能只能用于输出比较和PWM模式，而不能用于强置模式。

例如，OCxREF信号可以联到一个比较器的输出，用于控制电流。这时，ETR必须配置如下：

1. 外部触发预分频器必须处于关闭: $TIMx_SLVCTL$ 寄存器中的 $ETPS[1:0]=00$ 。
2. 必须禁止外部时钟模式2: $TIMx_SLVCTL$ 寄存器中的 $ECE=0$ 。
3. 外部触发极性(ETP)和外部触发滤波器(ETF)可以根据需要配置。

下图显示了当ETRF输入变为高时, 对应不同OCxCE的值, OCxREF信号的动作。在这个例子中, 定时器TIMx被置于PWM模式。

图88 清除TIMx的OCxREF



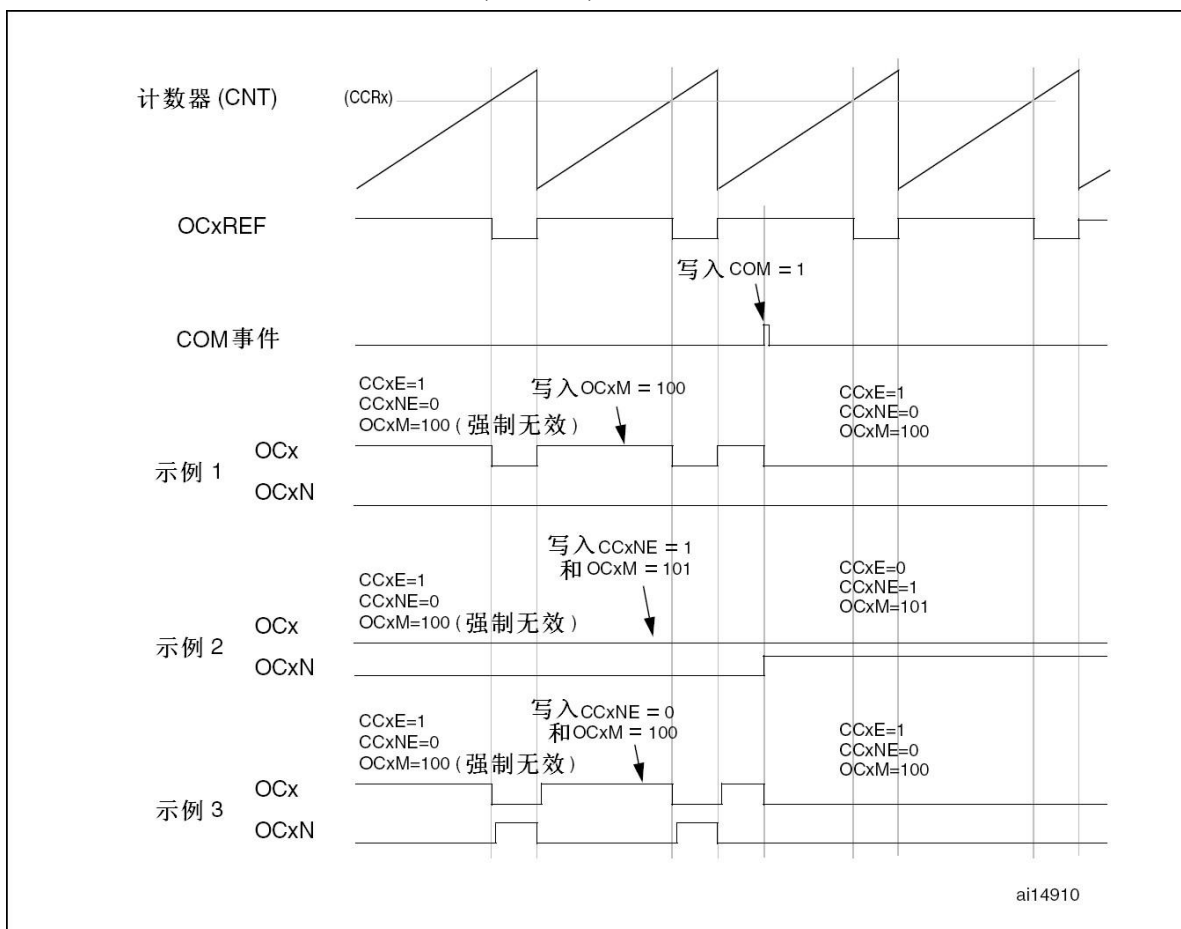
11.3.14 产生六步PWM输出

当在一个通道上需要互补输出时, 预装载位有OCxM、CCxE和CCxNE。在发生COM换相事件时, 这些预装载位被传送到影子寄存器位。这样你就可以预先设置好下一步配置, 并在同一个时刻同时修更改所有通道的配置。COM可以通过设置TIMx_EVTGR寄存器的COM位由软件产生, 或在TRGI上升沿由硬件产生。

当发生COM事件时会设置一个标志位(TIMx_SR寄存器中的COMIF位), 这时如果已设置了TIMx_DIER寄存器的COMIE位, 则产生一个中断; 如果已设置了TIMx_DIER寄存器的COMDE位, 则产生一个DMA请求。

下图显示当发生COM事件时, 三种不同配置下OCx和OCxN输出。

图89 产生六步PWM，使用COM的例子(OSSR=1)



11.3.15 单脉冲模式

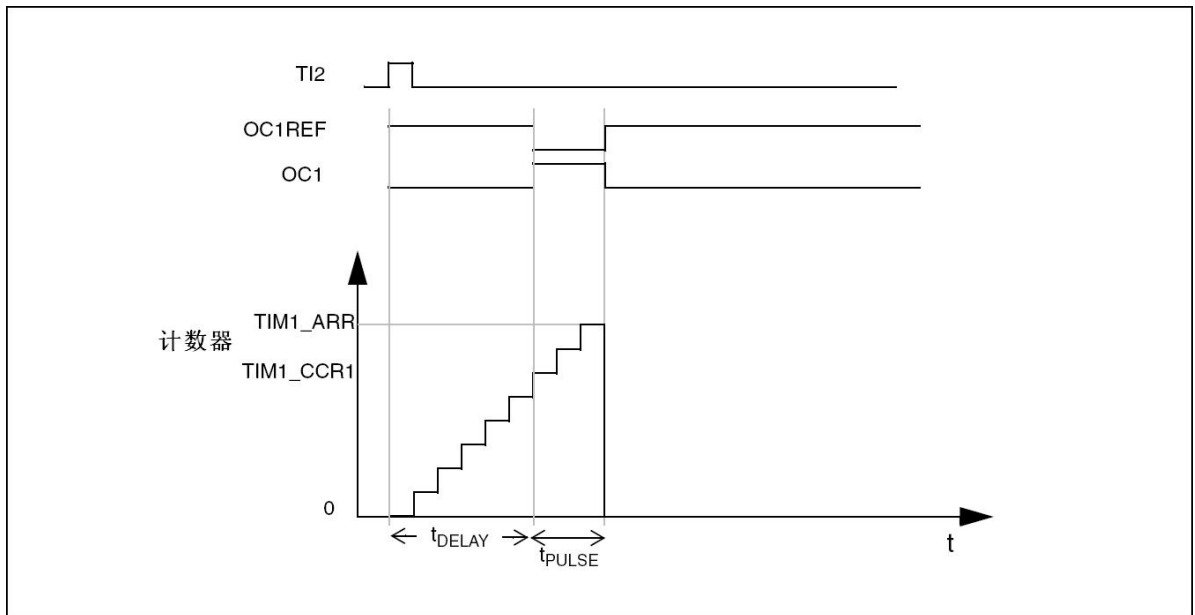
单脉冲模式(OPM)是前述众多模式的一个特例。这种模式允许计数器响应一个激励，并在一个程序可控的延时之后产生一个脉宽可程序控制的脉冲。

可以通过从模式控制器启动计数器，在输出比较模式或者PWM模式下产生波形。设置TIM_x_CTLRI寄存器中的OPM位将选择单脉冲模式，这样可以让计数器自动地在产生下一个更新事件UEV时停止。

仅当比较值与计数器的初始值不同时，才能产生一个脉冲。启动之前(当定时器正在等待触发)，必须如下配置：

- 向上计数方式：计数器CNT < CCR_x ≤ ARR (特别地, 0 < CCR_x),
- 向下计数方式：计数器CNT > CCR_x。

图90 单脉冲模式的例子



例如，你需要在从TI2输入脚上检测到一个上升沿开始，延迟 t_{DELAY} 之后，在OC1上产生一个长度为 t_{PULSE} 的正脉冲。

假定TI2FP2作为触发1:

- 置TIMx_CCMR1寄存器中的CC2S=01，把TI2FP2映像到TI2。
- 置TIMx_CCER寄存器中的CC2P=0，使TI2FP2能够检测上升沿。
- 置TIMx_SLVCTL寄存器中的TS=110，TI2FP2作为从模式控制器的触发(TRGI)。
- 置TIMx_SLVCTL寄存器中的SMS=110(触发模式)，TI2FP2被用来启动计数器。

OPM的波形由写入比较寄存器的数值决定(要考虑时钟频率和计数器预分频器)

- t_{DELAY} 由TIMx_CCR1寄存器中的值定义。
- t_{PULSE} 由自动装载值和比较值之间的差值定义($\text{TIMx_ARR} - \text{TIMx_CCR1}$)。
- 假定当发生比较匹配时要产生从0到1的波形，当计数器达到预装载值时要产生一个从1到0的波形；首先要置TIMx_CCMR1寄存器的OC1M=111，进入PWM模式2；根据需要有选择地使能预装载寄存器：置TIMx_CCMR1中的OC1PE=1和TIMx_CTLR1寄存器中的ARPE；然后在TIMx_CCR1寄存器中填写比较值，在TIMx_ARRL寄存器中填写自动装载值，设置UG位来产生一个更新事件，然后等待在TI2上的一个外部触发事件。本例中，CC1P=0。

在这个例子中，TIMx_CTLR1寄存器中的DIR和CMS位应该置低。

因为只需要一个脉冲，所以必须设置TIMx_CTLR1寄存器中的OPM=1，在下一个更新事件(当计数器从自动装载值翻转回0)时停止计数。

特殊情况：OCx快速使能：

在单脉冲模式下，在TIx输入脚的边沿检测逻辑设置CEN位以启动计数器。然后计数器和比较值间的比较操作产生了输出的转换。但是这些操作需要一定的时钟周期，因此它限制了可得到的最小延时 t_{DELAY} 。

如果要以最小延时输出波形，可以设置TIMx_CCMRx寄存器中的OCxFE位；此时OCxREF(和OCx)直接响应激励而不再依赖比较的结果，输出的波形与比较匹配时的波形一样。OCxFE只在通道配置为PWM1和PWM2模式时起作用。

11.3.16 编码器接口模式

选择编码器接口模式的方法是：如果计数器只在TI2的边沿计数，则置TIMx_SLVCTLR寄存器中的SMS=001；如果只在TI1边沿计数，则置SMS=010；如果计数器同时在TI1和TI2边沿计数，则置SMS=011。

通过设置TIMx_CCER寄存器中的CC1P和CC2P位，可以选择TI1和TI2极性；如果需要，还可以对输入滤波器编程。

两个输入TI1和TI2被用来作为增量编码器的接口。参看表73，假定计数器已经启动(TIMx_CTLR1寄存器中的CEN=1)，则计数器由每次在TI1FP1或TI2FP2上的有效跳变驱动。TI1FP1和TI2FP2是TI1和TI2在通过输入滤波器和极性控制后的信号；如果没有滤波和变相，则TI1FP1=TI1，TI2FP2=TI2。根据两个输入信号的跳变顺序，产生了计数脉冲和方向信号。依据两个输入信号的跳变顺序，计数器向上或向下计数，同时硬件对TIMx_CTLR1寄存器的DIR位进行相应的设置。不管计数器是依靠TI1计数、依靠TI2计数或者同时依靠TI1和TI2计数，在任一输入端(TI1或者TI2)的跳变都会重新计算DIR位。

编码器接口模式基本上相当于使用了一个带有方向选择的外部时钟。这意味着计数器只在0到TIMx_ARLR寄存器的自动装载值之间连续计数(根据方向，或是0到ARR计数，或是ARR到0计数)。所以在开始计数之前必须配置TIMx_ARLR；同样，捕获器、比较器、预分频器、重复计数器、触发输出特性等仍工作如常。编码器模式和外部时钟模式2不兼容，因此不能同时操作。

在这个模式下，计数器依照增量编码器的速度和方向被自动的修改，因此计数器的内容始终指示着编码器的位置。计数方向与相连的传感器旋转的方向对应。下表列出了所有可能的组合，假设TI1和TI2不同时变换。

表73 计数方向与编码器信号的关系

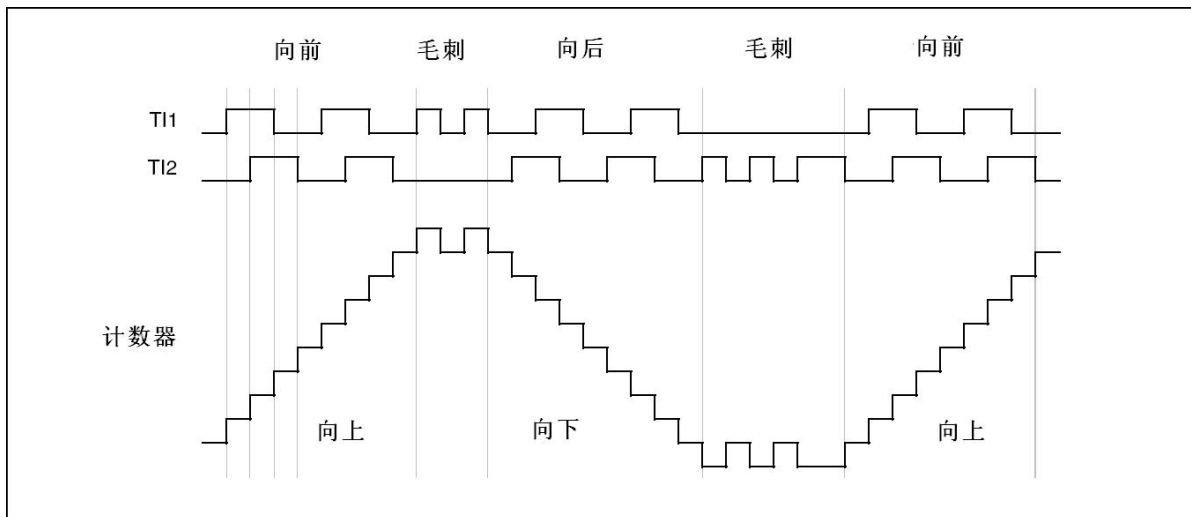
有效边沿	相对信号的电平 (TI1FP1对应TI2, TI2FP2对应TI1)	TI1FP1信号		TI2FP2信号	
		上升	下降	上升	下降
仅在TI1计数	高	向下计数	向上计数	不计数	不计数
	低	向上计数	向下计数	不计数	不计数
仅在TI2计数	高	不计数	不计数	向上计数	向下计数
	低	不计数	不计数	向下计数	向上计数
在TI1和TI2上计数	高	向下计数	向上计数	向上计数	向下计数
	低	向上计数	向下计数	向下计数	向上计数

一个外部的增量编码器可以直接与MCU连接而不需要外部接口逻辑。但是，一般会使用比较器将编码器的差动输出转换到数字信号，这大大增加了抗噪声干扰能力。编码器输出的第三个信号表示机械零点，可以把它连接到一个外部中断输入并触发一个计数器复位。

下图是一个计数器操作的实例，显示了计数信号的产生和方向控制。它还显示了当选择了双边沿时，输入抖动是如何被抑制的；抖动可能会在传感器的位置靠近一个转换点时产生。在这个例子中，我们假定配置如下：

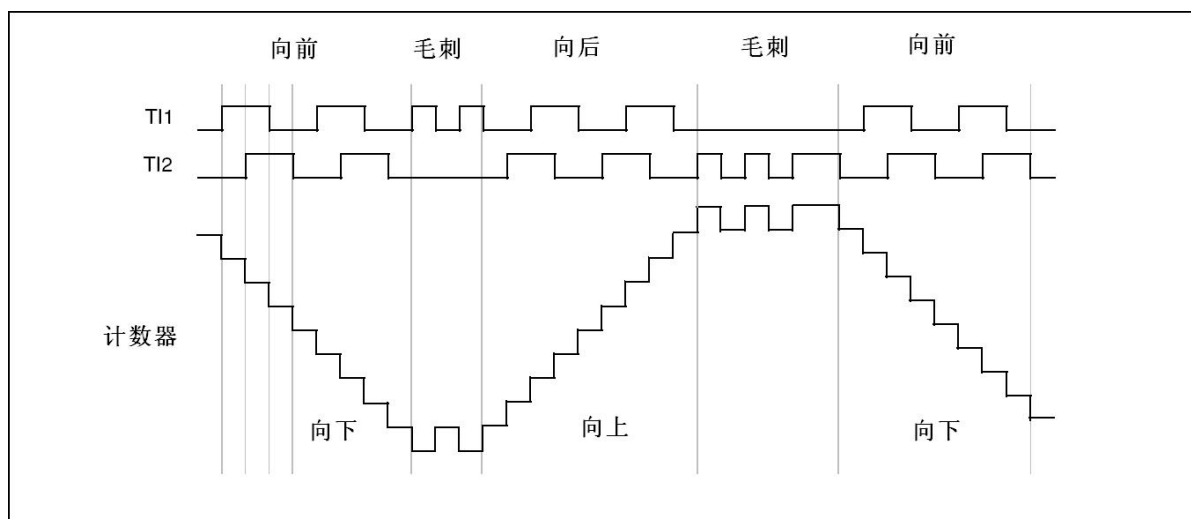
- CC1S='01' (TIMx_CCMR1寄存器，IC1FP1映射到TI1)
- CC2S='01' (TIMx_CCMR2寄存器，IC2FP2映射到TI2)
- CC1P='0' (TIMx_CCER寄存器，IC1FP1不反相，IC1FP1=TI1)
- CC2P='0' (TIMx_CCER寄存器，IC2FP2不反相，IC2FP2=TI2)
- SMS='011' (TIMx_SLVCTLR寄存器，所有的输入均在上升沿和下降沿有效)
- CEN='1' (TIMx_CTLR1寄存器，计数器使能)

图91 编码器模式下的计数器操作实例



下图为当IC1FP1极性反相时计数器的操作实例(CC1P='1', 其他配置与上例相同)

图92 IC1FP1反相的编码器接口模式实例



当定时器配置成编码器接口模式时，提供传感器当前位置的信息。使用第二个配置在捕获模式的定时器，可以测量两个编码器事件的间隔，获得动态的信息(速度，加速度，减速度)。指示机械零点的编码器输出可被用做此目的。根据两个事件间的间隔，可以按照固定的时间读出计数器。如果可能的话，你可以把计数器的值锁存到第三个输入捕获寄存器(捕获信号必须是周期的并且可以由另一个定时器产生)；也可以通过一个由实时时钟产生的DMA请求来读取它的值。

11.3.17 定时器输入异或功能

TIMx_CTLR2寄存器中的TI1S位，允许通道1的输入滤波器连接到一个异或门的输出端，异或门的3个输入端为TIMx_CH1、TIMx_CH2和TIMx_CH3。

异或输出能够被用于所有定时器的输入功能，如触发或输入捕获。下节11.3.18给出了此特性用于连接霍尔传感器的例子。

11.3.18 与霍尔传感器的接口

使用高级控制定时器(TIM1或TIM8)产生PWM信号驱动马达时，可以用另一个通用

TIMx(TIM2、TIM3、TIM4)定时器作为“接口定时器”来连接霍尔传感器，见图93，3个定时器输入脚(CC1、CC2、CC3)通过一个异或门连接到TI1输入通道(通过设置TIMx_CTLR2寄存器中的TI1S位来选择)，“接口定时器”捕获这个信号。

从模式控制器被配置于复位模式，从输入是TI1F_ED。每当3个输入之一变化时，计数器从新从0开始计数。这样产生一个由霍尔输入端的任何变化而触发的时间基准。

“接口定时器”上的捕获/比较通道1配置为捕获模式，捕获信号为TRC(见图76)。捕获值反映了两个输入变化间的时间延迟，给出了马达速度的信息。

“接口定时器”可以用来在输出模式产生一个脉冲，这个脉冲可以(通过触发一个COM事件)用于改变高级定时器TIM1各个通道的属性，而高级控制定时器产生PWM信号驱动马达。因此“接口定时器”通道必须编程为在一个指定的延时(输出比较或PWM模式)之后产生一个正脉冲，这个脉冲通过TRGO输出被送到高级控制定时器TIM1。

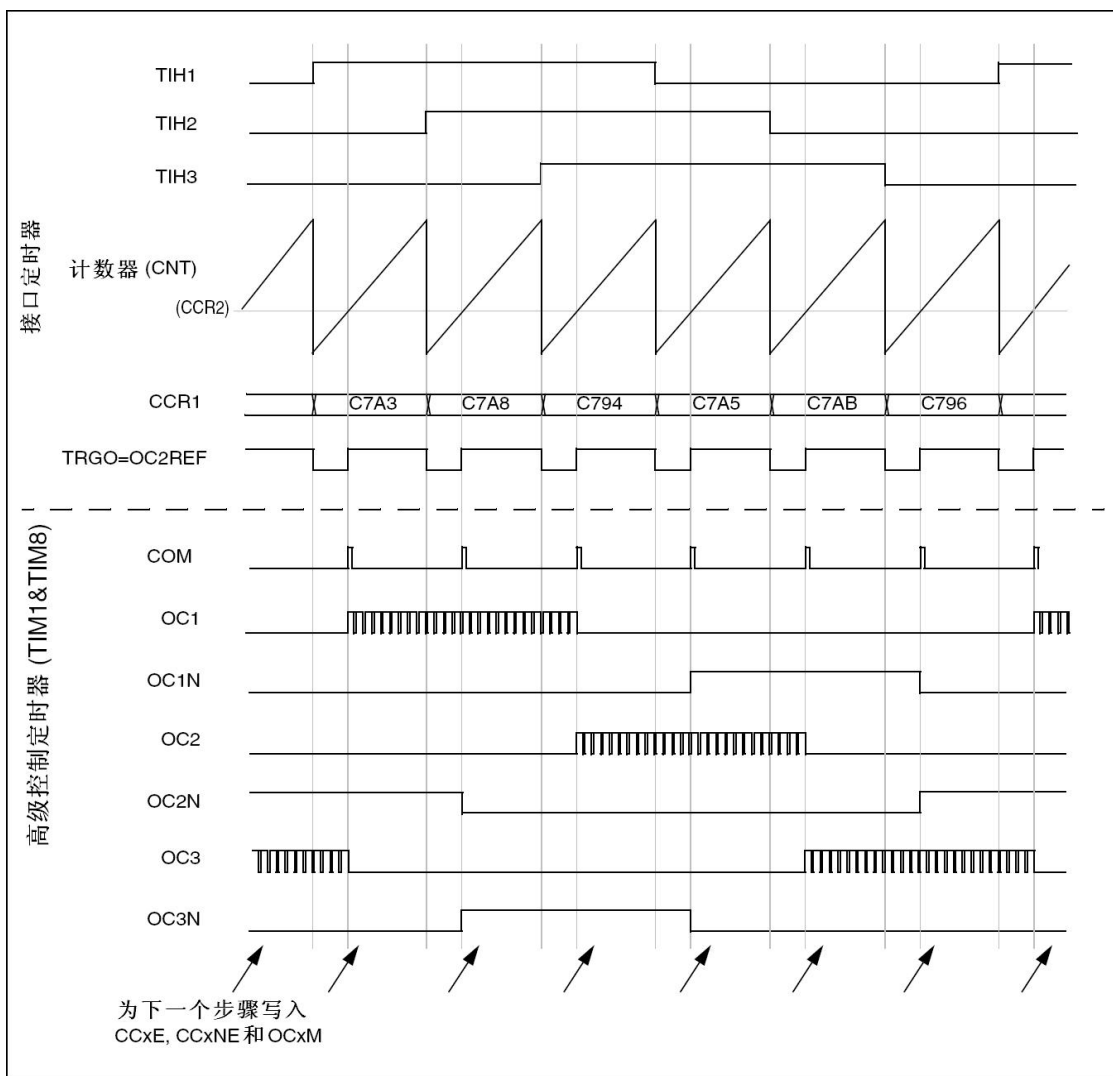
举例：霍尔输入连接到TIMx定时器，要求每次任一霍尔输入上发生变化之后的一个指定的时刻，改变高级控制定时器TIMx的PWM配置。

- 置TIMx_CTLR2寄存器的TI1S位为‘1’，配置三个定时器输入逻辑或到TI1输入，
- 时基编程：置TIMx_ARLR为其最大值(计数器必须通过TI1的变化清零)。设置预分频器得到一个最大的计数器周期，它长于传感器上的两次变化的时间间隔。
- 设置通道1为捕获模式(选中TRC)：置TIMx_CCMR1寄存器中CC1S=01，如果需要，还可以设置数字滤波器。
- 设置通道2为PWM2模式，并具有要求的延时：置TIMx_CCMR1寄存器中的OC2M=111和CC2S=00。
- 选择OC2REF作为TRGO上的触发输出：置TIMx_CTLR2寄存器中的MMS=101。

在高级控制寄存器TIM1中，正确的ITR输入必须是触发器输入，定时器被编程为产生PWM信号，捕获/比较控制信号为预装载的(TIMx_CTLR2寄存器中CCPC=1)，同时触发输入控制COM事件(TIMx_CTLR2寄存器中CCUS=1)。在一次COM事件后，写入下一步的PWM控制位(CCxE、OCxM)，这可以在处理OC2REF上升沿的中断子程序里实现。

下图显示了这个实例

图93 霍尔传感器接口的实例



11.3.19 TIMx定时器和外部触发的同步

TIMx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。

从模式：复位模式

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TIMx_CTLR1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(TIMx_ARLR, TIMx_CCRx)都被更新了。

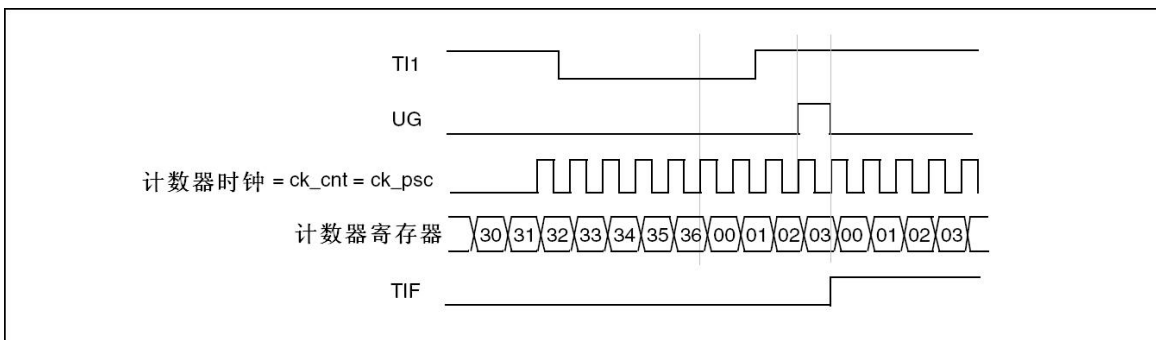
在以下的例子中，TI1输入端的上升沿导致向上计数器被清零：

- 配置通道1以检测TI1的上升沿。配置输入滤波器的带宽(在本例中，不需要任何滤波器，因此保持IC1F=0000)。触发操作中不使用捕获预分频器，所以不需要配置。CC1S位只选择输入捕获源，即TIMx_CCMR1寄存器中CC1S=01。置TIMx_CCER寄存器中CC1P=0以确定极性(只检测上升沿)。
- 置TIMx_SLVCTLRL寄存器中SMS=100，配置定时器为复位模式；置TIMx_SLVCTLRL寄存器中TS=101，选择TI1作为输入源。
- 置TIMx_CTLR1寄存器中CEN=1，启动计数器。

计数器开始依据内部时钟计数，然后正常运转直到TI1出现一个上升沿；此时，计数器被清零然后从0重新开始计数。同时，触发标志(TIMx_SR寄存器中的TIF位)被设置，根据TIMx_DIER寄存器中TIE(中断使能)位和TDE(DMA使能)位的设置，产生一个中断请求或一个DMA请求。

下图显示当自动重装载寄存器TIMx_ARRL=0x36时的动作。在TI1上升沿和计数器的实际复位之间的延时取决于TI1输入端的重同步电路。

图94 复位模式下的控制电路



从模式：门控模式

按照选中的输入端电平使能计数器。

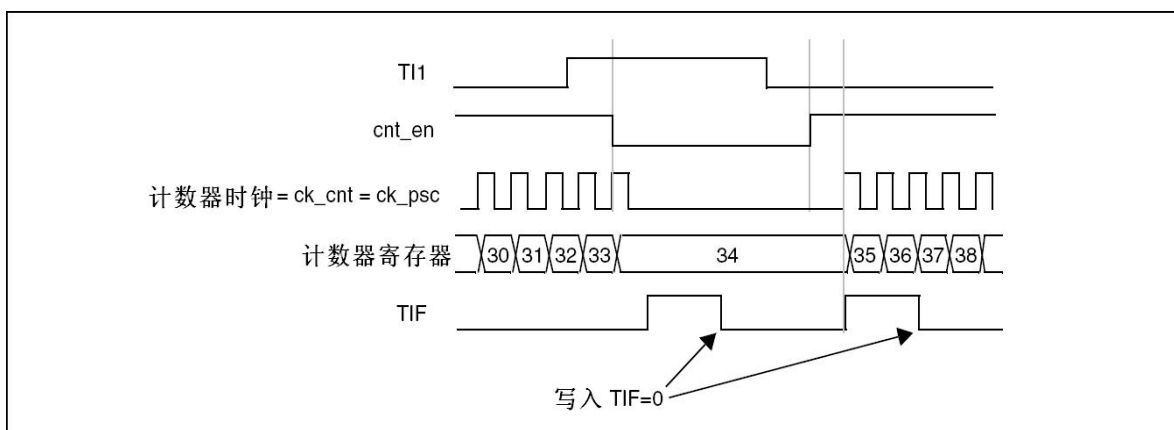
在如下的例子中，计数器只在TI1为低时向上计数：

- 配置通道1以检测TI1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IC1F=0000)。触发操作中不使用捕获预分频器，所以不需要配置。CC1S位用于选择输入捕获源，置TIMx_CCMR1寄存器中CC1S=01。置TIMx_CCER寄存器中CC1P=1以确定极性(只检测低电平)。
- 置TIMx_SLVCTL1寄存器中SMS=101，配置定时器为门控模式；置TIMx_SLVCTL1寄存器中TS=101，选择TI1作为输入源。
- 置TIMx_CTL1寄存器中CEN=1，启动计数器。在门控模式下，如果CEN=0，则计数器不能启动，不论触发输入电平如何。

只要TI1为低，计数器开始依据内部时钟计数，一旦TI1变高则停止计数。当计数器开始或停止时都设置TIMx_SR中的TIF标置。

TI1上升沿和计数器实际停止之间的延时取决于TI1输入端的重同步电路。

图95 门控模式下的控制电路



从模式：触发模式

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TI2输入的上升沿开始向上计数：

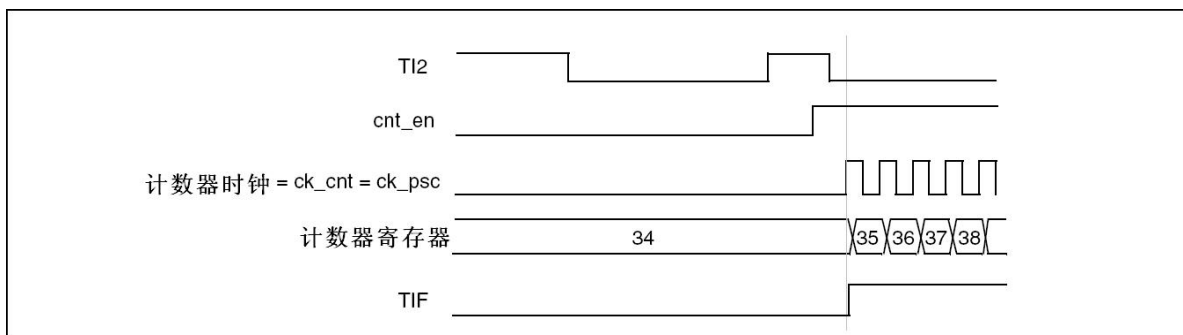
配置通道2检测TI2的上升沿。配置输入滤波器带宽(本例中不需要任何滤波器, 保持IC2F=0000)。触发操作中不使用捕获预分频器, 不需要配置。CC2S位只用于选择输入捕获源, 置TIMx_CCMR1寄存器中CC2S=01。置TIMx_CCER寄存器中CC2P=1以确定极性(只检测低电平)。

- 置TIMx_SLVCTL寄存器中SMS=110, 配置定时器为触发模式; 置TIMx_SLVCTL寄存器中TS=110, 选择TI2作为输入源。

当TI2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。

TI2上升沿和计数器启动计数之间的延时, 取决于TI2输入端的重同步电路。

图96 触发器模式下的控制电路



从模式: 外部时钟模式2 + 触发模式

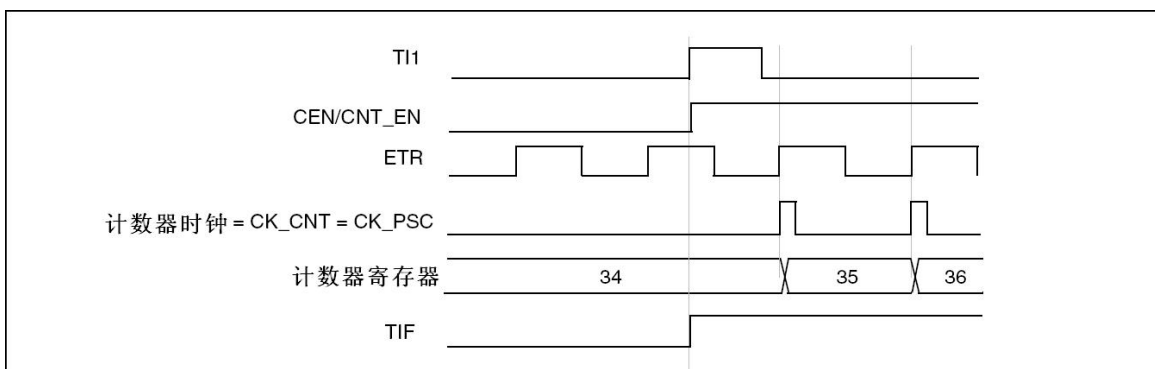
外部时钟模式2可以与另一种从模式(外部时钟模式1和编码器模式除外)一起使用。这时, ETR信号被用作外部时钟的输入, 在复位模式、门控模式或触发模式可以选择另一个输入作为触发输入。不建议使用TIMx_SLVCTL寄存器的TS位选择ETR作为TRGI。

在下面的例子中, 一旦在TI1上出现一个上升沿, 计数器即在ETR的每一个上升沿向上计数一次:

1. 通过TIMx_SLVCTL寄存器配置外部触发输入电路:
 - ETF=0000: 没有滤波
 - ETPS=00: 不用预分频器
 - ETP=0: 检测ETR的上升沿, 置ECE=1使能外部时钟模式2。
2. 按如下配置通道1, 检测TI的上升沿:
 - IC1F=0000: 没有滤波
 - 触发操作中不使用捕获预分频器, 不需要配置
 - 置TIMx_CCMR1寄存器中CC1S=01, 选择输入捕获源
 - 置TIMx_CCER寄存器中CC1P=0以确定极性(只检测上升沿)
3. 置TIMx_SLVCTL寄存器中SMS=110, 配置定时器为触发模式。置TIMx_SLVCTL寄存器中TS=101, 选择TI1作为输入源。

当TI1上出现一个上升沿时, TIF标志被设置, 计数器开始在ETR的上升沿计数。ETR信号的上升沿和计数器实际复位间的延时, 取决于ETRP输入端的重同步电路。

图97 外部时钟模式2+触发模式下的控制电路



11.3.20 定时器同步

所有TIM定时器在内部相连，用于定时器同步或链接。详见下一章14.3.15节。

11.3.21 调试模式

当微控制器进入调试模式时(内核核心停止)，根据DBG模块中DBG_TIMx_STOP的设置，TIMx计数器可以或者继续正常操作，或者停止。详见随后的29.16.2节。

11.4 TIM1 寄存器描述

关于在寄存器描述里面所用到的缩写，详见第1章。

可以用半字(16位)或字(32位)的方式操作这些外设寄存器。

11.4.1 TIM1 控制寄存器1(TIMx_CTLR1)

偏移址: 0x00

复位值: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留						CKD[1:0]	ARPE	CMS[1:0]	DIR	OPM	URS	UDIS	CEN		
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位15:10	保留，始终读为0。
位9:8	CKD[1:0]: 时钟分频因子 (Clock division) 这2位定义在定时器时钟(CK_INT)频率、死区时间和由死区发生器与数字滤波器(ETR,TIx)所用的采样时钟之间的分频比例。 00: $t_{DTS} = t_{CK_INT}$ 01: $t_{DTS} = 2 \times t_{CK_INT}$ 10: $t_{DTS} = 4 \times t_{CK_INT}$ 11: 保留，不要使用这个配置
位7	ARPE: 自动重载预装载允许位 (Auto-reload preload enable) 0: TIMx_ARLR寄存器没有缓冲; 1: TIMx_ARLR寄存器被装入缓冲器。
位6:5	CMS[1:0]: 选择中央对齐模式 (Center-aligned mode selection) 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(TIMx_CCMRx寄存器中CCxS=00)的输出比较中断标志位，只在计数器向下计数时被设置。 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(TIMx_CCMRx寄存器中CCxS=00)的输出比较中断标志位，只在计数器向上计数时被设置。 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(TIMx_CCMRx寄存器

	中CCxS=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设置。 注：在计数器开启时(CEN=1)，不允许从边沿对齐模式转换到中央对齐模式。
位4	DIR : 方向(Direction) 0: 计数器向上计数; 1: 计数器向下计数。 注：当计数器配置为中央对齐模式或编码器模式时，该位为只读。
位3	OPM : 单脉冲模式 (One pulse mode) 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。
位2	URS : 更新请求源 (Update request source) 软件通过该位选择UEV事件的源 0: 如果使能了更新中断或DMA请求，则下述任一事件产生更新中断或DMA请求: - 计数器溢出/下溢 - 设置UG位 - 从模式控制器产生的更新 1: 如果使能了更新中断或DMA请求，则只有计数器溢出/下溢才产生更新中断或DMA请求。
位1	UDIS : 禁止更新 (Update disable) 软件通过该位允许/禁止UEV事件的发生 0: 允许UEV。更新(UEV)事件由下述任一事件产生: - 计数器溢出/下溢 - 设置UG位 - 从模式控制器产生的更新 具有缓存的寄存器被装入它们的预装载值。(译注：更新影子寄存器) 1: 禁止UEV。不产生更新事件，影子寄存器(ARR、PSC、CCR _x)保持它们的值。如果设置了UG位或从模式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。
位0	CEN : 使能计数器 (Counter enable) 0: 禁止计数器; 1: 使能计数器。 注：在软件设置了CEN位后，外部时钟、门控模式和编码器模式才能工作。触发模式可以自动地通过硬件设置CEN位。

11.4.2 TIM1控制寄存器2(TIMx_CTLR2)

偏移址: 0x04

复位值: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]	CCDS	CCUS	保留	CCPC		
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw
位15	保留，始终读为0。														
位14	OIS4 : 输出空闲状态4(OC4输出)。参见OIS1位。														
位13	OIS3N : 输出空闲状态3(OC3N输出)。参见OIS1N位。														
位12	OIS3 : 输出空闲状态3(OC3输出)。参见OIS1位。														
位11	OIS2N : 输出空闲状态2(OC2N输出)。参见OIS1N位。														
位10	OIS2 : 输出空闲状态2(OC2输出)。参见OIS1位。														
位9	OIS1N : 输出空闲状态1(OC1N输出) (Output Idle state 1) 0: 当MOE=0时，死区后OC1N=0; 1: 当MOE=0时，死区后OC1N=1。														

	注：已经设置了LOCK(TIMx_BKR寄存器)级别1、2或3后，该位不能被修改。
位8	OIS1 : 输出空闲状态1(OC1输出) (Output Idle state 1) 0: 当MOE=0时，如果实现了OC1N，则死区后OC1=0; 1: 当MOE=0时，如果实现了OC1N，则死区后OC1=1。 注：已经设置了LOCK(TIMx_BKR寄存器)级别1、2或3后，该位不能被修改。
位7	TI1S : TI1 选 择 (TI1 selection) 0: TIMx_CH1引脚连到TI1输入; 1: TIMx_CH1、TIMx_CH2和TIMx_CH3引脚经异或后连到TI1输入。
位6:4	MMS[2:0] : 主模式选择 (Master mode selection) 这3位用于选择在主模式下送到从定时器的同步信息(TRGO)。可能的组合如下: 000: 复位 – TIMx_EVTGR寄存器的UG位被用于作为触发输出(TRGO)。如果是触发输入产生的复位(从模式控制器处于复位模式)，则TRGO上的信号相对实际的复位会有一个延迟。 001: 使能 – 计数器使能信号CNT_EN被用于作为触发输出(TRGO)。有时需要在同一时间启动多个定时器或控制在一段时间内使能从定时器。计数器使能信号是通过CEN控制位和门控模式下的触发输入信号的逻辑或产生。当计数器使能信号受控于触发输入时，TRGO上会有一个延迟，除非选择了主/从模式(见TIMx_SLVCTL寄存器中MSM位的描述)。 010: 更新 – 更新事件被选为触发输入(TRGO)。例如，一个主定时器的时钟可以被用作一个从定时器的预分频器。 011: 比较脉冲 – 在发生一次捕获或一次比较成功时，当要设置CC1IF标志时(即使它已经为高)，触发输出送出一个正脉冲(TRGO)。 100: 比较 – OC1REF信号被用于作为触发输出(TRGO)。 101: 比较 – OC2REF信号被用于作为触发输出(TRGO)。 110: 比较 – OC3REF信号被用于作为触发输出(TRGO)。 111: 比较 – OC4REF信号被用于作为触发输出(TRGO)。
位3	CCDS : 捕获/比较的DMA选择(Capture/compare DMA selection) 0: 当发生CCx事件时，送出CCx的DMA请求; 1: 当发生更新事件时，送出CCx的DMA请求。
位2	CCUS : 捕获/比较控制更新选择 (Capture/compare control update selection) 0: 如果捕获/比较控制位是预装载的(CCPC=1)，只能通过设置COM位更新它们; 1: 如果捕获/比较控制位是预装载的(CCPC=1)，可以通过设置COM位或TRGI上的一个上升沿更新它们。 注：该位只对具有互补输出的通道起作用。
位1	保留，始终读为0。
位0	CCPC : 捕获/比较预装载控制位 (Capture/compare preloaded control) 0: CCxE, CCxNE和OCxM位不是预装载的; 1: CCxE, CCxNE和OCxM位是预装载的; 设置该位后，它们只在设置了COM位后被更新。 注：该位只对具有互补输出的通道起作用。

11.4.3 TIM1从模式控制寄存器(TIMx_SLVCTLR)

偏移地址: 0x08

复位值: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			保留	SMS[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw

位15	ETP : 外部触发极性 (External trigger polarity) 该位选择是用ETR还是ETR的反相来作为触发操作 0: ETR不反相，高电平或上升沿有效; 1: ETR被反相，低电平或下降沿有效。
-----	---

位14	<p>ECE: 外部时钟使能位(External clock enable)</p> <p>该位启用外部时钟模式2</p> <p>0: 禁止外部时钟模式2;</p> <p>1: 使能外部时钟模式2。计数器由ETRF信号上的任意有效边沿驱动。</p> <p>注1: 设置ECE位与选择外部时钟模式1并将TRGI连到ETRF(SMS=111和TS=111)具有相同功效。</p> <p>注2: 下述从模式可以与外部时钟模式2同时使用: 复位模式, 门控模式和触发模式; 但是, 这时TRGI不能连到ETRF(TS位不能是'111')。</p> <p>注3: 外部时钟模式1和外部时钟模式2同时被使能时, 外部时钟的输入是ETRF。</p>																
位13:12	<p>ETPS[1:0]: 外部触发预分频 (External trigger prescaler)</p> <p>外部触发信号ETRP的频率必须最多是TIMxCLK频率的1/4。当输入较快的外部时钟时, 可以使用预分频降低ETRP的频率。</p> <p>00: 关闭预分频;</p> <p>01: ETRP频率除以2;</p> <p>10: ETRP频率除以4;</p> <p>11: ETRP频率除以8。</p>																
位11:8	<p>ETF[3:0]: 外部触发滤波 (External trigger filter)</p> <p>这些位定义了对ETRP信号采样的频率和对ETRP数字滤波的带宽。实际上, 数字滤波器是一个事件计数器, 它记录到N个事件后会产生一个输出的跳变。</p> <table border="0"> <tr> <td>0000: 无滤波器, 以f_{DTS}采样</td><td>1000: 采样频率f_{SAMPLING}=f_{DTS}/8, N=6</td></tr> <tr> <td>0001: 采样频率f_{SAMPLING}=f_{CK_INT}, N=2</td><td>1001: 采样频率f_{SAMPLING}=f_{DTS}/8, N=8</td></tr> <tr> <td>0010: 采样频率f_{SAMPLING}=f_{CK_INT}, N=4</td><td>1010: 采样频率f_{SAMPLING}=f_{DTS}/16, N=5</td></tr> <tr> <td>0011: 采样频率f_{SAMPLING}=f_{CK_INT}, N=8</td><td>1011: 采样频率f_{SAMPLING}=f_{DTS}/16, N=6</td></tr> <tr> <td>0100: 采样频率f_{SAMPLING}=f_{DTS}/2, N=6</td><td>1100: 采样频率f_{SAMPLING}=f_{DTS}/16, N=8</td></tr> <tr> <td>0101: 采样频率f_{SAMPLING}=f_{DTS}/2, N=8</td><td>1101: 采样频率f_{SAMPLING}=f_{DTS}/32, N=5</td></tr> <tr> <td>0110: 采样频率f_{SAMPLING}=f_{DTS}/4, N=6</td><td>1110: 采样频率f_{SAMPLING}=f_{DTS}/32, N=6</td></tr> <tr> <td>0111: 采样频率f_{SAMPLING}=f_{DTS}/4, N=8</td><td>1111: 采样频率f_{SAMPLING}=f_{DTS}/32, N=8</td></tr> </table>	0000: 无滤波器, 以f _{DTS} 采样	1000: 采样频率f _{SAMPLING} =f _{DTS} /8, N=6	0001: 采样频率f _{SAMPLING} =f _{CK_INT} , N=2	1001: 采样频率f _{SAMPLING} =f _{DTS} /8, N=8	0010: 采样频率f _{SAMPLING} =f _{CK_INT} , N=4	1010: 采样频率f _{SAMPLING} =f _{DTS} /16, N=5	0011: 采样频率f _{SAMPLING} =f _{CK_INT} , N=8	1011: 采样频率f _{SAMPLING} =f _{DTS} /16, N=6	0100: 采样频率f _{SAMPLING} =f _{DTS} /2, N=6	1100: 采样频率f _{SAMPLING} =f _{DTS} /16, N=8	0101: 采样频率f _{SAMPLING} =f _{DTS} /2, N=8	1101: 采样频率f _{SAMPLING} =f _{DTS} /32, N=5	0110: 采样频率f _{SAMPLING} =f _{DTS} /4, N=6	1110: 采样频率f _{SAMPLING} =f _{DTS} /32, N=6	0111: 采样频率f _{SAMPLING} =f _{DTS} /4, N=8	1111: 采样频率f _{SAMPLING} =f _{DTS} /32, N=8
0000: 无滤波器, 以f _{DTS} 采样	1000: 采样频率f _{SAMPLING} =f _{DTS} /8, N=6																
0001: 采样频率f _{SAMPLING} =f _{CK_INT} , N=2	1001: 采样频率f _{SAMPLING} =f _{DTS} /8, N=8																
0010: 采样频率f _{SAMPLING} =f _{CK_INT} , N=4	1010: 采样频率f _{SAMPLING} =f _{DTS} /16, N=5																
0011: 采样频率f _{SAMPLING} =f _{CK_INT} , N=8	1011: 采样频率f _{SAMPLING} =f _{DTS} /16, N=6																
0100: 采样频率f _{SAMPLING} =f _{DTS} /2, N=6	1100: 采样频率f _{SAMPLING} =f _{DTS} /16, N=8																
0101: 采样频率f _{SAMPLING} =f _{DTS} /2, N=8	1101: 采样频率f _{SAMPLING} =f _{DTS} /32, N=5																
0110: 采样频率f _{SAMPLING} =f _{DTS} /4, N=6	1110: 采样频率f _{SAMPLING} =f _{DTS} /32, N=6																
0111: 采样频率f _{SAMPLING} =f _{DTS} /4, N=8	1111: 采样频率f _{SAMPLING} =f _{DTS} /32, N=8																
位7	<p>MSM: 主/从模式 (Master/slave mode)</p> <p>0: 无作用;</p> <p>1: 触发输入(TRGI)上的事件被延迟了, 以允许在当前定时器(通过TRGO)与它的从定时器间的完美同步。这对要求把几个定时器同步到一个单一的外部事件时是非常有用的。</p>																
位6:4	<p>TS[2:0]: 触发选择 (Trigger selection)</p> <p>这3位选择用于同步计数器的触发输入。</p> <table border="0"> <tr> <td>000: 内部触发0(ITR0)</td><td>100: TI1的边沿检测器(TI1F_ED)</td></tr> <tr> <td>001: 内部触发1(ITR1)</td><td>101: 滤波后的定时器输入1(TI1FP1)</td></tr> <tr> <td>010: 内部触发2(ITR2)</td><td>110: 滤波后的定时器输入2(TI2FP2)</td></tr> <tr> <td>011: 内部触发3(ITR3)</td><td>111: 外部触发输入(ETRF)</td></tr> </table> <p>更多有关ITRx的细节, 参见表74。</p> <p>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。</p>	000: 内部触发0(ITR0)	100: TI1的边沿检测器(TI1F_ED)	001: 内部触发1(ITR1)	101: 滤波后的定时器输入1(TI1FP1)	010: 内部触发2(ITR2)	110: 滤波后的定时器输入2(TI2FP2)	011: 内部触发3(ITR3)	111: 外部触发输入(ETRF)								
000: 内部触发0(ITR0)	100: TI1的边沿检测器(TI1F_ED)																
001: 内部触发1(ITR1)	101: 滤波后的定时器输入1(TI1FP1)																
010: 内部触发2(ITR2)	110: 滤波后的定时器输入2(TI2FP2)																
011: 内部触发3(ITR3)	111: 外部触发输入(ETRF)																
位3	保留, 始终读为0。																
位2:0	<p>SMS[2:0]: 从模式选择 (Slave mode selection)</p> <p>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)</p> <p>000: 关闭从模式 – 如果CEN=1, 则预分频器直接由内部时钟驱动。</p> <p>001: 编码器模式1 – 根据TI1FP1的电平, 计数器在TI2FP2的边沿向上/下计数。</p> <p>010: 编码器模式2 – 根据TI2FP2的电平, 计数器在TI1FP1的边沿向上/下计数。</p> <p>011: 编码器模式3 – 根据另一个信号的输入电平, 计数器在TI1FP1和TI2FP2的边沿向上/下计数。</p> <p>100: 复位模式 – 选中的触发输入(TRGI)的上升沿重新初始化计数器, 并且产生一个更新寄存器</p>																

	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_EN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变化时输出一个脉冲, 然而门控模式是要检查触发输入的电平。

表74 TIM1内部触发连接

从定时器	ITR0 (TS=000)	ITR1 (TS=001)	ITR2 (TS=010)	ITR3 (TS=011)
TIM1		TIM2	TIM3	TIM4

11.4.4 TIM1 DMA/中断使能寄存器(TIMx_DIER)

偏移地址: 0x0C

复位值: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位15	保留, 始终读为0。
位14	TDE: 允许触发DMA请求(Trigger DMA request enable) 0: 禁止触发DMA请求; 1: 允许触发DMA请求。
位13	COMDE: 允许COM的DMA请求(COM DMA request enable) 0: 禁止COM的DMA请求; 1: 允许COM的DMA请求。
位12	CC4DE: 允许捕获/比较4的DMA请求(Capture/Compare 4 DMA request enable) 0: 禁止捕获/比较4的DMA请求; 1: 允许捕获/比较4的DMA请求。
位11	CC3DE: 允许捕获/比较3的DMA请求(Capture/Compare 3 DMA request enable) 0: 禁止捕获/比较3的DMA请求; 1: 允许捕获/比较3的DMA请求。
位10	CC2DE: 允许捕获/比较2的DMA请求(Capture/Compare 2 DMA request enable) 0: 禁止捕获/比较2的DMA请求; 1: 允许捕获/比较2的DMA请求。
位9	CC1DE: 允许捕获/比较1的DMA请求(Capture/Compare 1 DMA request enable) 0: 禁止捕获/比较1的DMA请求; 1: 允许捕获/比较1的DMA请求。
位8	UDE: 允许更新的DMA请求(Update DMA request enable) 0: 禁止更新的DMA请求; 1: 允许更新的DMA请求。
位7	BIE: 允许刹车中断(Break interrupt enable) 0: 禁止刹车中断; 1: 允许刹车中断。
位6	TIE: 触发中断使能(Trigger interrupt enable) 0: 禁止触发中断; 1: 使能触发中断。

位5	COMIE : 允许COM中断(COM interrupt enable) 0: 禁止COM中断; 1: 允许COM中断。
位4	CC4IE : 允许捕获/比较4中断(Capture/Compare 4 interrupt enable) 0: 禁止捕获/比较4中断; 1: 允许捕获/比较4中断。
位3	CC3IE : 允许捕获/比较3中断(Capture/Compare 3 interrupt enable) 0: 禁止捕获/比较3中断; 1: 允许捕获/比较3中断。
位2	CC2IE : 允许捕获/比较2中断(Capture/Compare 2 interrupt enable) 0: 禁止捕获/比较2中断; 1: 允许捕获/比较2中断。
位1	CC1IE : 允许捕获/比较1中断(Capture/Compare 1 interrupt enable) 0: 禁止捕获/比较1中断; 1: 允许捕获/比较1中断。
位0	UIE : 允许更新中断(Update interrupt enable) 0: 禁止更新中断; 1: 允许更新中断。

11.4.5 TIM1 状态寄存器(TIMx_SR)

偏移地址: 0x10

复位值: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	CC4OF	CC3OF	CC2OF	CC1OF	保留	BIF	TIF	COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UIF		
	rc w0	rc w0	rc w0	rc w0		rc w0	rc w0	rc w0	rc w0	rc w0	rc w0	rc w0	rc w0		

位15:13	保留, 始终读为0。
位12	CC4OF : 捕获/比较4重复捕获标记 (Capture/Compare 4 overcapture flag) 参见CC1OF描述。
位11	CC3OF : 捕获/比较3重复捕获标记 (Capture/Compare 3 overcapture flag) 参见CC1OF描述。
位10	CC2OF : 捕获/比较2重复捕获标记 (Capture/Compare 2 overcapture flag) 参见CC1OF描述。
位9	CC1OF : 捕获/比较1重复捕获标记 (Capture/Compare 1 overcapture flag) 仅当相应的通道被配置为输入捕获时, 该标记可由硬件置1。写0可清除该位。 0: 无重复捕获产生; 1: 计数器的值被捕获到TIMx_CCR1寄存器时, CC1IF的状态已经为'1'。
位8	保留, 始终读为0。
位7	BIF : 刹车中断标记 (Break interrupt flag) 一旦刹车输入有效, 由硬件对该位置'1'。如果刹车输入无效, 则该位可由软件清'0'。 0: 无刹车事件产生; 1: 刹车输入上检测到有效电平。
位6	TIF : 触发器中断标记 (Trigger interrupt flag) 当发生触发事件(当从模式控制器处于除门控模式外的其它模式时, 在TRGI输入端检测到有效边沿, 或门控模式下的任一边沿)时由硬件对该位置'1'。它由软件清'0'。 0: 无触发器事件产生; 1: 触发中断等待响应。

位5	<p>COMIF: COM中断标记 (COM interrupt flag)</p> <p>一旦产生COM事件(当捕获/比较控制位: CCxE、CCxNE、OCxM已被更新)该位由硬件置'1'。它由软件清'0'。</p> <p>0: 无COM事件产生;</p> <p>1: COM中断等待响应。</p>
位4	<p>CC4IF: 捕获/比较4中断标记 (Capture/Compare 4 interrupt flag)</p> <p>参考CC1IF描述。</p>
位3	<p>CC3IF: 捕获/比较3中断标记 (Capture/Compare 3 interrupt flag)</p> <p>参考CC1IF描述。</p>
位2	<p>CC2IF: 捕获/比较2中断标记 (Capture/Compare 2 interrupt flag)</p> <p>参考CC1IF描述。</p>
位1	<p>CC1IF: 捕获/比较1中断标记 (Capture/Compare 1 interrupt flag)</p> <p>如果通道CC1配置为输出模式:</p> <p>当计数器值与比较值匹配时该位由硬件置1, 但在中心对称模式下除外(参考TIMx_CTLR1寄存器的CMS位)。它由软件清'0'。</p> <p>0: 无匹配发生;</p> <p>1: TIMx_CNTR的值与TIMx_CCR1的值匹配。</p> <p>当TIMx_CCR1的内容大于TIMx_APR的内容时, 在向上或向上/下计数模式时计数器溢出, 或向下计数模式时的计数器下溢条件下, CC1IF位变高</p> <p>如果通道CC1配置为输入模式:</p> <p>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TIMx_CCR1清'0'。</p> <p>0: 无输入捕获产生;</p> <p>1: 计数器值已被捕获(拷贝)至TIMx_CCR1(在IC1上检测到与所选极性相同的边沿)。</p>
位0	<p>UIF: 更新中断标记 (Update interrupt flag)</p> <p>当产生更新事件时该位由硬件置'1'。它由软件清'0'。</p> <p>0: 无更新事件产生;</p> <p>1: 更新中断等待响应。当寄存器被更新时该位由硬件置'1':</p> <ul style="list-style-type: none"> - 若TIMx_CTLR1寄存器的UDIS=0, 当重复计数器数值上溢或下溢时(重复计数器=0时产生更新事件)。 - 若TIMx_CTLR1寄存器的URS=0、UDIS=0, 当设置TIMx_EVTGR寄存器的UG=1时产生更新事件, 通过软件对计数器CNT重新初始化时。 - 若TIMx_CTLR1寄存器的URS=0、UDIS=0, 当计数器CNT被触发事件重新初始化时。(参考13.4.3: TIM1从模式控制寄存器(TIMx_SLVCTLR))。

11.4.6 TIM1 事件产生寄存器(TIMx_EVTGR)

偏移地址:0x14

复位值:0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留								BG	TG	COMG	CC4G	CC3G	CC2G	CC1G	UG
								W	W	W	W	W	W	W	W

位15:8	保留，始终读为0。
位7	BG: 产生刹车事件 (Break generation) 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，若开启对应的中断和DMA，则产生相应的中断和DMA。
位6	TG: 产生触发事件 (Trigger generation) 该位由软件置'1'，用于产生一个触发事件，由硬件自动清'0'。 0: 无动作; 1: TIMx_SR寄存器的TIF=1，若开启对应的中断和DMA，则产生相应的中断和DMA。
位5	COMG: 捕获/比较事件，产生控制更新 (Capture/Compare control update generation) 该位由软件置'1'，由硬件自动清'0'。 0: 无动作; 1: 当CCPC=1，允许更新CCxE、CCxNE、OCxM位。 注: 该位只对拥有互补输出的通道有效。
位4	CC4G: 产生捕获/比较4事件 (Capture/Compare 4 generation) 参考CC1G描述。
位3	CC3G: 产生捕获/比较3事件 (Capture/Compare 3 generation) 参考CC1G描述。
位2	CC2G: 产生捕获/比较2事件 (Capture/Compare 2 generation) 参考CC1G描述。
位1	CC1G: 产生捕获/比较1事件 (Capture/Compare 1 generation) 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，若开启对应的中断和DMA，则产生相应的中断和DMA。 若通道CC1配置为输入: 当前的计数器值被捕获至TIMx_CCR1寄存器; 设置CC1IF=1，若开启对应的中断和DMA，则产生相应的中断和DMA。若CC1IF已经为1，则设置CC1OF=1。
位0	UG: 产生更新事件 (Update generation) 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TIMx_ARLR的值。

11.4.7 TIM1 捕获/比较模式寄存器 1(TIMx_CCMR1)

偏移地址: 0x18

复位值: 0x0000

通道可用于输入(捕获模式)或输出(比较模式), 通道的方向由相应的CCxS位定义。该寄存器其它位的作用在输入和输出模式下不同。OCxx描述了通道在输出模式下的功能, ICxx描述了通道在输入模式下的功能。因此必须注意, 同一个位在输出模式和输入模式下的功能是不同的。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2CE	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
IC2F[3:0]				IC2PSC[1:0]				IC1F[3:0]			IC1PSC[1:0]				
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

输出比较模式:

位15	OC2CE : 输出比较2清0使能 (Output Compare 2 clear enable)
位14:12	OC2M[2:0] : 输出比较2模式 (Output Compare 2 mode)
位11	OC2PE : 输出比较2预装载使能 (Output Compare 2 preload enable)
位10	OC2FE : 输出比较2快速使能 (Output Compare 2 fast enable)
位9:8	CC2S[1:0] : 捕获/比较2选择。(Capture/Compare 2 selection) 该位定义通道的方向(输入/输出), 及输入脚的选择: 00: CC2通道被配置为输出; 01: CC2通道被配置为输入, IC2映射在TI2上; 10: CC2通道被配置为输入, IC2映射在TI1上; 11: CC2通道被配置为输入, IC2映射在TRC上。此模式仅工作在内部触发器输入被选中时(由TIMx_SLVCTL寄存器中的TS位选择)。 注 : CC2S仅在通道关闭时(TIMx_CCER寄存器的CC2E=0)才是可写的。
位7	OC1CE : 输出比较1清'0'使能 (Output Compare 1 clear enable) 0: OC1REF 不受ETRF输入的影响; 1: 一旦检测到ETRF输入高电平, 清除OC1REF=0。
位6:4	OC1M[2:0] : 输出比较1模式 (Output Compare 1 mode) 该3位定义了输出参考信号OC1REF的动作, 而OC1REF决定了OC1、OC1N的值。OC1REF 是高电平有效, 而OC1、OC1N的有效电平取决于CC1P、CC1NP位。 000: 冻结。输出比较寄存器TIMx_CCR1与计数器TIMx_CNTR间的比较对OC1REF不起作用; 001: 匹配时设置通道1为有效电平。当计数器TIMx_CNTR的值与捕获/比较寄存器 1 (TIMx_CCR1)相同时, 强制OC1REF为高。 010: 匹配时设置通道1为无效电平。当计数器TIMx_CNTR的值与捕获/比较寄存器 1 (TIMx_CCR1)相同时, 强制OC1REF为低。 011: 翻转。当TIMx_CCR1=TIMx_CNTR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TIMx_CNTR<TIMx_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TIMx_CNTR>TIMx_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TIMx_CNTR<TIMx_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TIMx_CNTR>TIMx_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(TIMx_BDTR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。

	<p>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。</p>
位3	<p>OC1PE: 输出比较1预装载使能 (Output Compare 1 preload enable)</p> <p>0: 禁止TIMx_CCR1寄存器的预装载功能, 可随时写入TIMx_CCR1寄存器, 并且新写入的数值立即起作用。</p> <p>1: 开启TIMx_CCR1寄存器的预装载功能, 读写操作仅对预装载寄存器操作, TIMx_CCR1的预装载值在更新事件到来时被加载至当前寄存器中。</p> <p>注1: 一旦LOCK级别设为3(TIMx_BDTR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。</p> <p>注2: 仅在单脉冲模式下(TIMx_CTLR1寄存器的OPM=1), 可以在未确认预装载寄存器情况下使用PWM模式, 否则其动作不确定。</p>
位2	<p>OC1FE: 输出比较1快速使能 (Output Compare 1 fast enable)</p> <p>该位用于加快CC输出对触发输入事件的响应。</p> <p>0: 根据计数器与CCR1的值, CC1正常操作, 即使触发器是打开的。当触发器的输入有一个有效沿时, 激活CC1输出的最小延时为5个时钟周期。</p> <p>1: 输入到触发器的有效沿的作用就象发生了一次比较匹配。因此, OC被设置为比较电平而与比较结果无关。采样触发器的有效沿和CC1输出间的延时被缩短为3个时钟周期。</p> <p>OCFE只在通道被配置成PWM1或PWM2模式时起作用。</p>
位1:0	<p>CC1S[1:0]: 捕获/比较1选择。(Capture/Compare 1 selection)</p> <p>这2位定义通道的方向(输入/输出), 及输入脚的选择:</p> <p>00: CC1通道被配置为输出;</p> <p>01: CC1通道被配置为输入, IC1映射在TI1上;</p> <p>10: CC1通道被配置为输入, IC1映射在TI2上;</p> <p>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TIMx_SLVCTL寄存器中的TS位选择)。</p> <p>注: CC1S仅在通道关闭时(TIMx_CCER寄存器的CC1E=0)才是可写的。</p>

输入捕获模式:

位15:12	IC2F[3:0]: 输入捕获2滤波器 (Input capture 2 filter)														
位11:10	IC2PSC[1:0]: 输入/捕获2预分频器 (Input capture 2 prescaler)														
位9:8	<p>CC2S[1:0]: 捕获/比较2选择 (Capture/Compare 2 selection)</p> <p>这2位定义通道的方向(输入/输出), 及输入脚的选择:</p> <p>00: CC2通道被配置为输出;</p> <p>01: CC2通道被配置为输入, IC2映射在TI2上;</p> <p>10: CC2通道被配置为输入, IC2映射在TI1上;</p> <p>11: CC2通道被配置为输入, IC2映射在TRC上。此模式仅工作在内部触发器输入被选中时(由TIMx_SLVCTL寄存器中的TS位选择)。</p> <p>注: CC2S仅在通道关闭时(TIMx_CCER寄存器的CC2E=0)才是可写的。</p>														
位7:4	<p>IC1F[3:0]: 输入捕获1滤波器 (Input capture 1 filter)</p> <p>这几位定义了TI1输入的采样频率及数字滤波器长度。数字滤波器由一个事件计数器组成, 它记录到N个事件后会产生一个输出的跳变:</p> <table border="0"> <tr> <td>0000: 无滤波器, 以f_{DTS}采样</td><td>1000: 采样频率$f_{SAMPLING}=f_{DTS}/8$, $N=6$</td></tr> <tr> <td>0001: 采样频率$f_{SAMPLING}=f_{CK_INT}$, $N=2$</td><td>1001: 采样频率$f_{SAMPLING}=f_{DTS}/8$, $N=8$</td></tr> <tr> <td>0010: 采样频率$f_{SAMPLING}=f_{CK_INT}$, $N=4$</td><td>1010: 采样频率$f_{SAMPLING}=f_{DTS}/16$, $N=5$</td></tr> <tr> <td>0011: 采样频率$f_{SAMPLING}=f_{CK_INT}$, $N=8$</td><td>1011: 采样频率$f_{SAMPLING}=f_{DTS}/16$, $N=6$</td></tr> <tr> <td>0100: 采样频率$f_{SAMPLING}=f_{DTS}/2$, $N=6$</td><td>1100: 采样频率$f_{SAMPLING}=f_{DTS}/16$, $N=8$</td></tr> <tr> <td>0101: 采样频率$f_{SAMPLING}=f_{DTS}/2$, $N=8$</td><td>1101: 采样频率$f_{SAMPLING}=f_{DTS}/32$, $N=5$</td></tr> <tr> <td>0110: 采样频率$f_{SAMPLING}=f_{DTS}/4$, $N=6$</td><td>1110: 采样频率$f_{SAMPLING}=f_{DTS}/32$, $N=6$</td></tr> </table>	0000: 无滤波器, 以 f_{DTS} 采样	1000: 采样频率 $f_{SAMPLING}=f_{DTS}/8$, $N=6$	0001: 采样频率 $f_{SAMPLING}=f_{CK_INT}$, $N=2$	1001: 采样频率 $f_{SAMPLING}=f_{DTS}/8$, $N=8$	0010: 采样频率 $f_{SAMPLING}=f_{CK_INT}$, $N=4$	1010: 采样频率 $f_{SAMPLING}=f_{DTS}/16$, $N=5$	0011: 采样频率 $f_{SAMPLING}=f_{CK_INT}$, $N=8$	1011: 采样频率 $f_{SAMPLING}=f_{DTS}/16$, $N=6$	0100: 采样频率 $f_{SAMPLING}=f_{DTS}/2$, $N=6$	1100: 采样频率 $f_{SAMPLING}=f_{DTS}/16$, $N=8$	0101: 采样频率 $f_{SAMPLING}=f_{DTS}/2$, $N=8$	1101: 采样频率 $f_{SAMPLING}=f_{DTS}/32$, $N=5$	0110: 采样频率 $f_{SAMPLING}=f_{DTS}/4$, $N=6$	1110: 采样频率 $f_{SAMPLING}=f_{DTS}/32$, $N=6$
0000: 无滤波器, 以 f_{DTS} 采样	1000: 采样频率 $f_{SAMPLING}=f_{DTS}/8$, $N=6$														
0001: 采样频率 $f_{SAMPLING}=f_{CK_INT}$, $N=2$	1001: 采样频率 $f_{SAMPLING}=f_{DTS}/8$, $N=8$														
0010: 采样频率 $f_{SAMPLING}=f_{CK_INT}$, $N=4$	1010: 采样频率 $f_{SAMPLING}=f_{DTS}/16$, $N=5$														
0011: 采样频率 $f_{SAMPLING}=f_{CK_INT}$, $N=8$	1011: 采样频率 $f_{SAMPLING}=f_{DTS}/16$, $N=6$														
0100: 采样频率 $f_{SAMPLING}=f_{DTS}/2$, $N=6$	1100: 采样频率 $f_{SAMPLING}=f_{DTS}/16$, $N=8$														
0101: 采样频率 $f_{SAMPLING}=f_{DTS}/2$, $N=8$	1101: 采样频率 $f_{SAMPLING}=f_{DTS}/32$, $N=5$														
0110: 采样频率 $f_{SAMPLING}=f_{DTS}/4$, $N=6$	1110: 采样频率 $f_{SAMPLING}=f_{DTS}/32$, $N=6$														

	0111: 采样频率 $f_{\text{SAMPLING}}=f_{\text{DTS}}/4$, $N=8$ 1111: 采样频率 $f_{\text{SAMPLING}}=f_{\text{DTS}}/32$, $N=8$
位3:2	IC1PSC[1:0]: 输入/捕获1预分频器 (Input capture 1 prescaler) 这2位定义了CC1输入(IC1)的预分频系数。 一旦CC1E=0(TIMx_CCER寄存器中), 则预分频器复位。 00: 无预分频器, 捕获输入口上检测到的每一个边沿都触发一次捕获; 01: 每2个事件触发一次捕获; 10: 每4个事件触发一次捕获; 11: 每8个事件触发一次捕获。
位1:0	CC1S[1:0]: 捕获/比较1选择 (Capture/Compare 1 Selection) 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TI1上; 10: CC1通道被配置为输入, IC1映射在TI2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TIMx_SLVCTLR寄存器的TS位选择)。 注: CC1S仅在通道关闭时(TIMx_CCER寄存器的CC1E=0)才是可写的。

11.4.8 TIM1 捕获/比较模式寄存器 2(TIMx_CCMR2)

偏移地址: 0x1C

复位值: 0x0000

参看以上CCMR1寄存器的描述

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC4CE	OC4M[4:0]			OC4PE	OC4FE	CC4S[1:0]		OC3CE	OC3M[2:0]			OC3PE	OC3FE	CC3S[1:0]	
IC4F[3:0]				IC4PSC[1:0]				IC3F[3:0]			IC3PSC[1:0]				
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

输出比较模式:

位15	OC4CE: 输出比较4清0使能 (Output compare 4 clear enable)
位14:12	OC4M[2:0]: 输出比较4模式 (Output compare 4 mode)
位11	OC4PE: 输出比较4预装载使能 (Output compare 4 preload enable)
位10	OC4FE: 输出比较4快速使能 (Output compare 4 fast enable)
位9:8	CC4S[1:0]: 捕获/比较4选择 (Capture/Compare 4 selection) 该2位定义通道的方向(输入/输出), 及输入脚的选择: 00: CC4通道被配置为输出; 01: CC4通道被配置为输入, IC4映射在TI4上; 10: CC4通道被配置为输入, IC4映射在TI3上; 11: CC4通道被配置为输入, IC4映射在TRC上。此模式仅工作在内部触发器输入被选中时(由TIMx_SLVCTLR寄存器的TS位选择)。 注: CC4S仅在通道关闭时(TIMx_CCER寄存器的CC4E=0)才是可写的。
位7	OC3CE: 输出比较3清0使能 (Output compare 3 clear enable)
位6:4	OC3M[2:0]: 输出比较3模式 (Output compare 3 mode)
位3	OC3PE: 输出比较3预装载使能 (Output compare 3 preload enable)
位2	OC3FE: 输出比较3快速使能 (Output compare 3 fast enable)

位1:0	CC3S[1:0]: 捕获/比较3选择 (Capture/Compare 3 selection) 这2位定义通道的方向(输入/输出), 及输入脚的选择: 00: CC3通道被配置为输出; 01: CC3通道被配置为输入, IC3映射在TI3上; 10: CC3通道被配置为输入, IC3映射在TI4上; 11: CC3通道被配置为输入, IC3映射在TRC上。此模式仅工作在内部触发器输入被选中时(由TIMx_SLVCTL寄存器TS位选择)。 注: CC3S仅在通道关闭时(TIMx_CCER寄存器的CC3E=0)才是可写的。
------	---

输入捕获模式:

位15:12	IC4F[3:0]: 输入捕获4滤波器 (Input capture 4 filter)
位11:10	IC4PSC[1:0]: 输入/捕获4预分频器 (Input capture 4 prescaler)
位9:8	CC4S[1:0]: 捕获/比较4选择 (Capture/Compare 4 selection) 这2位定义通道的方向(输入/输出), 及输入脚的选择: 00: CC4通道被配置为输出; 01: CC4通道被配置为输入, IC4映射在TI4上; 10: CC4通道被配置为输入, IC4映射在TI3上; 11: CC4通道被配置为输入, IC4映射在TRC上。此模式仅工作在内部触发器输入被选中时(由TIMx_SLVCTL寄存器TS位选择)。 注: CC4S仅在通道关闭时(TIMx_CCER寄存器的CC4E=0)才是可写的。
位7:4	IC3F[3:0]: 输入捕获3滤波器 (Input capture 3 filter)
位3:2	IC3PSC[1:0]: 输入/捕获3预分频器 (Input capture 3 prescaler)
位1:0	CC3S[1:0]: 捕获/比较3选择 (Capture/compare 3 selection) 这2位定义通道的方向(输入/输出), 及输入脚的选择: 00: CC3通道被配置为输出; 01: CC3通道被配置为输入, IC3映射在TI3上; 10: CC3通道被配置为输入, IC3映射在TI4上; 11: CC3通道被配置为输入, IC3映射在TRC上。此模式仅工作在内部触发器输入被选中时(由TIMx_SLVCTL寄存器TS位选择)。 注: CC3S仅在通道关闭时(TIMx_CCER寄存器的CC3E=0)才是可写的。

11.4.9 TIM1 捕获/比较使能寄存器(TIMx_CCER)

偏移地址: 0x20

复位值: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E	
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
位15:14	保留, 始终读为0。														
位13	CC4P: 输入/捕获4输出极性 (Capture/Compare 4 output polarity) 参考CC1P的描述。														
位12	CC4E: 输入/捕获4输出使能 (Capture/Compare 4 output enable) 参考CC1E的描述。														
位11	CC3NP: 输入/捕获3互补输出极性 (Capture/Compare 3 complementary output polarity) 参考CC1NP的描述。														
位10	CC3NE: 输入/捕获3互补输出使能 (Capture/Compare 3 complementary output enable) 参考CC1NE的描述。														

位9	CC3P: 输入/捕获3输出极性 (Capture/Compare 3 output polarity) 参考CC1P的描述。
位8	CC3E: 输入/捕获3输出使能 (Capture/Compare 3 output enable) 参考CC1E 的描述。
位7	CC2NP: 输入/捕获2互补输出极性 (Capture/Compare 2 complementary output polarity) 参考CC1NP的描述。
位6	CC2NE: 输入/捕获2互补输出使能 (Capture/Compare 2 complementary output enable) 参考CC1NE的描述。
位5	CC2P: 输入/捕获2输出极性 (Capture/Compare 2 output polarity) 参考CC1P的描述。
位4	CC2E: 输入/捕获2输出使能 (Capture/Compare 2 output enable) 参考CC1E的描述。
位3	CC1NP: 输入/捕获1互补输出极性 (Capture/Compare 1 complementary output polarity) 0: OC1N高电平有效; 1: OC1N低电平有效。 注: 一旦LOCK级别(TIMx_BDTR寄存器中的LOCK位)设为3或2且CC1S=00(通道配置为输出) 则该位不能被修改。
位2	CC1NE: 输入/捕获1互补输出使能 (Capture/Compare 1 complementary output enable) 0: 关闭— OC1N禁止输出, 因此OC1N的电平依赖于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。
位1	CC1P: 输入/捕获1输出极性 (Capture/Compare 1 output polarity) CC1通道配置为输出: 0: OC1高电平有效; 1: OC1低电平有效。 CC1通道配置为输入: 该位选择是IC1还是IC1的反相信号作为触发或捕获信号。 0: 不反相: 捕获发生在IC1的上升沿; 当用作外部触发器时, IC1不反相。 1: 反相: 捕获发生在IC1的下降沿; 当用作外部触发器时, IC1反相。 注: 一旦LOCK级别(TIMx_BDTR寄存器中的LOCK位)设为3或2, 则该位不能被修改。
位0	CC1E: 输入/捕获1输出使能 (Capture/Compare 1 output enable) 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TIMx_CCR1寄存器。 0: 捕获禁止; 1: 捕获使能。

表75 带刹车功能的互补输出通道OCx和OCxN的控制位

控制位					输出状态 ⁽¹⁾	
MOE 位	OSSI 位	OSSR 位	CCxE 位	CCxNE 位	OCx 输出状态	OCxN 输出状态
1	X	0	0	0	输出禁止(与定时器断开) OCx=0, OCx_EN=0	输出禁止(与定时器断开) OCxN=0, OCxN_EN=0
		0	0	1	输出禁止(与定时器断开) OCx=0, OCx_EN=0	OCxREF + 极性, OCxN= OCxREF xor CCxNP, OCxN_EN=1
		0	1	0	OCxREF + 极性, OCx= OCxREF xor CCxP, OCx_EN=1	输出禁止(与定时器断开) OCxN=0, OCxN_EN=0
		0	1	1	OCxREF + 极性 + 死区, OCx_EN=1	OCxREF反相 + 极性 + 死区, OCxN_EN=1
		1	0	0	输出禁止(与定时器断开) OCx=CCxP, OCx_EN=0	输出禁止(与定时器断开) OCxN=CCxNP, OCxN_EN=0
		1	0	1	关闭状态(输出使能且为无效电 平) OCx=CCxP, OCx_EN=1	OCxREF + 极性, OCxN= OCxREF xor CCxNP, OCxN_EN=1
		1	1	0	OCxREF + 极性, OCx= OCxREF xor CCxP, OCx_EN=1	关闭状态(输出使能且为无效电平) OCxN=CCxNP, OCxN_EN=1
		1	1	1	OCxREF + 极性 + 死区, OCx_EN=1	OCxREF反相 + 极性 + 死区, OCxN_EN=1
0	0	X	0	0	输出禁止(与定时器断开) 异步地: OCx=CCxP, OCx_EN=0, OCxN=CCxNP, OCxN_EN=0; 若时钟存 在: 经过 一个死区 时间后 OCx=OISx, OCxN=OISxN, 假设OISx与OISxN并不都对应OCx和OCxN的有效电平。	
	0		0	1		
	0		1	0		
	0		1	1		
	1		0	0	关闭状态(输出使能且为无效电平) 异步地: OCx=CCxP, OCx_EN=1, OCxN=CCxNP, OCxN_EN=1; 若时钟存 在: 经过 一个死区 时间后 OCx=OISx, OCxN=OISxN, 假设OISx与OISxN并不都对应OCx和OCxN的有效电平。	
	1		0	1		
	1		1	0		
	1		1	1		

1. 如果一个通道的2个输出都没有使用(CCxE = CCxNE = 0), 那么OISx, OISxN, CCxP和CCxNP都必须清零。

注: 引脚连接到互补的OCx和OCxN通道的外部I/O引脚的状态, 取决于OCx和OCxN通道状态和GPIO以及AFIO寄存器。

11.4.10 计数器(TIMx_CNTR)

偏移地址: 0x24

复位值: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
位15:0		CNT[15:0]: 计数器的值 (Counter value)													

11.4.11 TIM1预分频器(TIMx_PSCR)

偏移地址: 0x28

复位值: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW
位15:0	PSC[15:0]: 预分频器的值 (Prescaler value) 计数器的时钟频率(CK_CNT)等于 $f_{CK_PSC}/(PSC[15:0]+1)$ 。 PSC包含了每次当更新事件产生时, 装入当前预分频器寄存器的值; 更新事件包括计数器被TIM_EGR的UG位清'0'或被工作在复位模式的从控制器清'0'。														

11.4.12 TIM1自动重装载寄存器(TIMx_ARLR)

偏移地址: 0x2C

复位值: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW
位15:0	ARR[15:0]: 自动重装载的值 (Prescaler value) ARR包含了将要装载入实际的自动重装载寄存器的值。 详细参考13.3.1节: 有关ARR的更新和动作。 当自动重装载的值为空时, 计数器不工作。														

11.4.13 TIM1重复计数寄存器(TIMx_RCNTR)

偏移地址: 0x30

复位值: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留								REP[7:0]							
								rW	rW	rW	rW	rW	rW	rW	rW
位15:8	保留, 始终读为0。														
位7:0	REP[7:0]: 重复计数器的值 (Repetition counter value) 开启了预装载功能后, 这些位允许用户设置比较寄存器的更新速率(即周期性地从预装载寄存器传输到当前寄存器); 如果允许产生更新中断, 则会同时影响产生更新中断的速率。 每次向下计数器REP_CNT达到0, 会产生一个更新事件并且计数器REP_CNT重新从REP值开始计数。由于REP_CNT只有在周期更新事件U_RC发生时才重载REP值, 因此对TIMx_RCNTR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。 这意味着在PWM模式中, (REP+1)对应着: <ul style="list-style-type: none"> 在边沿对齐模式下, PWM周期的数目; 在中心对称模式下, PWM半周期的数目; 														

11.4.14 TIM1捕获/比较寄存器 1(TIMx_CCR1)

偏移地址: 0x34

复位值: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1[15:0]															
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW
位15:0	CCR1[15:0]: 捕获/比较通道1的值 (Capture/Compare 1 value) 若CC1通道配置为输出: CCR1包含了装入当前捕获/比较1寄存器的值(预装载值)。 如果在TIMx_CCMR1寄存器(OC1PE位)中未选择预装载功能, 写入的数值会立即传输至当前寄存器中。否则只有当更新事件发生时, 此预装载值才传输至当前捕获/比较1寄存器中。 当前捕获/比较寄存器参与同计数器TIMx_CNTR的比较, 并在OC1端口上产生输出信号。 若CC1通道配置为输入: CCR1包含了由上一次输入捕获1事件(IC1)传输的计数器值。														

11.4.15 TIM1捕获/比较寄存器 2(TIMx_CCR2)

偏移地址: 0x38

复位值: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2[15:0]															
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW
位15:0	CCR2[15:0]: 捕获/比较通道2的值 (Capture/Compare 2 value) 若CC2通道配置为输出: CCR2包含了装入当前捕获/比较2寄存器的值(预装载值)。 如果在TIMx_CCMR2寄存器(OC2PE位)中未选择预装载特性, 写入的数值会立即传输至当前寄存器中。否则只有当更新事件发生时, 此预装载值才传输至当前捕获/比较2寄存器中。 当前捕获/比较寄存器参与同计数器TIMx_CNTR的比较, 并在OC2端口上产生输出信号。 若CC2通道配置为输入: CCR2包含了由上一次输入捕获2事件(IC2)传输的计数器值。														

11.4.16 TIM1捕获/比较寄存器 3(TIMx_CCR3)

偏移地址: 0x3C

复位值: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR3[15:0]															
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW
位15:0	CCR3[15:0]: 捕获/比较通道3的值 (Capture/Compare 3 value) 若CC3通道配置为输出: CCR3包含了装入当前捕获/比较3寄存器的值(预装载值)。 如果在TIMx_CCMR3寄存器(OC3PE位)中未选择预装载特性, 写入的数值会立即传输至当前寄存器中。否则只有当更新事件发生时, 此预装载值才传输至当前捕获/比较3寄存器中。 当前捕获/比较寄存器参与同计数器TIMx_CNTR的比较, 并在OC3端口上产生输出信号。 若CC3通道配置为输入: CCR3包含了由上一次输入捕获3事件(IC3)传输的计数器值。														

11.4.17 TIM1捕获/比较寄存器 4(TIMx_CCR4)

偏移地址: 0x40

复位值: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR4[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
位15:0	CCR4[15:0]: 捕获/比较通道4的值 (Capture/Compare 4 value) 若CC4通道配置为输出: CCR4包含了装入当前捕获/比较4寄存器的值(预装载值)。 如果在TIMx_CCMR4寄存器(OC4PE位)中未选择预装载特性, 写入的数值会立即传输至当前寄存器中。否则只有当更新事件发生时, 此预装载值才传输至当前捕获/比较4寄存器中。 当前捕获/比较寄存器参与同计数器TIMx_CNTR的比较, 并在OC4端口上产生输出信号。 若CC4通道配置为输入: CCR4包含了由上一次输入捕获4事件(IC4)传输的计数器值。														

11.4.18 TIM1刹车和死区寄存器(TIMx_BDTR)

偏移地址: 0x44

复位值: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DTG[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

注释: 根据锁定设置, AOE、BKP、BKE、OSSI、OSSR和DTG[7:0]位均可被写保护, 有必要在第一次写入TIMx_BDTR寄存器时对它们进行配置。

位15	MOE: 主输出使能 (Main output enable) 一旦刹车输入有效, 该位被硬件异步清'0'。根据AOE位的设置值, 该位可以由软件清'0'或被自动置1。它仅对配置为输出的通道有效。 0: 禁止OC和OCN输出或强制为空闲状态; 1: 如果设置了相应的使能位(TIMx_CCER寄存器的CCxE、CCxNE位), 则开启OC和OCN输出。 有关OC/OCN使能的细节, 参见13.4.9节, TIM1和TIM8捕获/比较使能寄存器(TIMx_CCER)。
位14	AOE: 自动输出使能 (Automatic output enable) 0: MOE只能被软件置'1'; 1: MOE能被软件置'1'或在下一个更新事件被自动置'1'(如果刹车输入无效)。 注: 一旦LOCK级别(TIMx_BDTR寄存器中的LOCK位)设为'1', 则该位不能被修改。
位13	BKP: 刹车输入极性 (Break polarity) 0: 刹车输入低电平有效; 1: 刹车输入高电平有效。 注: 一旦LOCK级别(TIMx_BDTR寄存器中的LOCK位)设为'1', 则该位不能被修改。注: 任何对该位的写操作都需要一个APB时钟的延迟以后才能起作用。
位12	BKE: 刹车功能使能 (Break enable) 0: 禁止刹车输入(BRK及CCS时钟失效事件); 1: 开启刹车输入(BRK及CCS时钟失效事件)。 注: 当设置了LOCK级别1时(TIMx_BDTR寄存器中的LOCK位), 该位不能被修改。 注: 任何对该位的写操作都需要一个APB时钟的延迟以后才能起作用。
位11	OSSR: 运行模式下“关闭状态”选择 (Off-state selection for Run mode) 该位用于当MOE=1且通道为互补输出时。没有互补输出的定时器中不存在OSSR位。

	<p>参考OC/OCN使能的详细说明(13.4.9节, TIM1和TIM8捕获/比较使能寄存器(TIMx_CCER))。</p> <p>0: 当定时器不工作时, 禁止OC/OCN输出(OC/OCN使能输出信号=0);</p> <p>1: 当定时器不工作时, 一旦CCxE=1或CCxNE=1, 首先开启OC/OCN并输出无效电平, 然后置OC/OCN使能输出信号=1。</p> <p>注: 一旦LOCK级别(TIMx_BDTR寄存器中的LOCK位)设为2, 则该位不能被修改。</p>
位10	<p>OSSI: 空闲模式下“关闭状态”选择 (Off-state selection for Idle mode)</p> <p>该位用于当MOE=0且通道设为输出时。</p> <p>参考OC/OCN使能的详细说明(13.4.9节, TIM1和TIM8捕获/比较使能寄存器(TIMx_CCER))。</p> <p>0: 当定时器不工作时, 禁止OC/OCN输出(OC/OCN使能输出信号=0);</p> <p>1: 当定时器不工作时, 一旦CCxE=1或CCxNE=1, OC/OCN首先输出其空闲电平, 然后OC/OCN使能输出信号=1。</p> <p>注: 一旦LOCK级别(TIMx_BDTR寄存器中的LOCK位)设为2, 则该位不能被修改。</p>
位9:8	<p>LOOK[1:0]: 锁定设置 (Lock configuration)</p> <p>该位为防止软件错误而提供写保护。</p> <p>00: 锁定关闭, 寄存器无写保护;</p> <p>01: 锁定级别1, 不能写入TIMx_BDTR寄存器的DTG、BKE、BKP、AOE位和TIMx_CTLR2寄存器的OISx/OISxN位;</p> <p>10: 锁定级别2, 不能写入锁定级别1中的各位, 也不能写入CC极性位(一旦相关通道通过CCxS位设为输出, CC极性位是TIMx_CCER寄存器的CCxP/CCNxP位)以及OSSR/OSSI位;</p> <p>11: 锁定级别3, 不能写入锁定级别2中的各位, 也不能写入CC控制位(一旦相关通道通过CCxS位设为输出, CC控制位是TIMx_CCMRx寄存器的OCxM/OCxPE位);</p> <p>注: 在系统复位后, 只能写一次LOCK位, 一旦写入TIMx_BDTR寄存器, 则其内容冻结直至复位。</p>
位7:0	<p>DTG[7:0]: 死区发生器设置 (Dead-time generator setup)</p> <p>这些位定义了插入互补输出之间的死区持续时间。假设DT表示其持续时间:</p> <p>$DTG[7:5]=0xx \Rightarrow DT=DTG[7:0] \times T_{dtg}, T_{dtg} = T_{DTS};$</p> <p>$DTG[7:5]=10x \Rightarrow DT=(64+DTG[5:0]) \times T_{dtg}, T_{dtg} = 2 \times T_{DTS};$</p> <p>$DTG[7:5]=110 \Rightarrow DT=(32+DTG[4:0]) \times T_{dtg}, T_{dtg} = 8 \times T_{DTS};$</p> <p>$DTG[7:5]=111 \Rightarrow DT=(32+DTG[4:0]) \times T_{dtg}, T_{dtg} = 16 \times T_{DTS};$</p> <p>例: 若$T_{DTS} = 125ns(8MHZ)$, 可能的死区时间为:</p> <p>0到15875ns, 若步长时间为125ns;</p> <p>16us到31750ns, 若步长时间为250ns;</p> <p>32us到63us, 若步长时间为1us;</p> <p>64us到126us, 若步长时间为2us;</p> <p>注: 一旦LOCK级别(TIMx_BDTR寄存器中的LOCK位)设为1、2或3, 则不能修改这些位。</p>

11.4.19 TIM1 DMA控制寄存器(TIMx_DCTLR)

偏移地址: 0x48

复位值: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留			DBL[4:0]					保留			DBA[4:0]				
rw			rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw
位15:13			保留，始终读为0。												
位12:8			<div>DBL[4:0]: DMA连续传送长度 (DMA burst length)</div> <div>这些位定义了DMA在连续模式下的传送长度(当对TIMx_DMAR寄存器进行读或写时，定时器则进行一次连续传送)，即：定义传输的次数，传输可以是半字(双字节)或字节：</div> <div>00000：1次传输00001：2次传输</div>												

	<p>00010: 3次传输 10001: 18次传输</p> <p>例：我们考虑这样的传输：DBL=7，DBA=TIM2_CR1</p> <p>- 如果DBL=7，DBA=TIM2_CR1表示待传输数据的地址，那么传输的地址由下式给出： (TIMx_CTLR1的地址) + DBA + (DMA索引), 其中 DMA索引 = DBL</p> <p>其中(TIMx_CTLR1的地址) + DBA再加上7，给出了将要写入或者读出数据的地址，这样数据的传输将发生在从地址(TIMx_CTLR1的地址) + DBA开始的7个寄存器。</p> <p>根据DMA数据长度的设置，可能发生以下情况：</p> <ul style="list-style-type: none"> - 如果设置数据为半字(16位)，那么数据就会传输给全部7个寄存器。 - 如果设置数据为字节，数据仍然会传输给全部7个寄存器：第一个寄存器包含第一个MSB字节，第二个寄存器包含第一个LSB字节，以此类推。因此对于定时器，用户必须指定由DMA传输的数据宽度。
位7:5	保留，始终读为0。
位4:0	<p>DBA[4:0]: DMA基地址 (DMA base address)</p> <p>这些位定义了DMA在连续模式下的基地址(当对TIMx_DMAR寄存器进行读或写时)，DBA定义为从TIMx_CTLR1寄存器所在地址开始的偏移量：</p> <p>00000: TIMx_CTLR1， 00001: TIMx_CTLR2， 00010: TIMx_SLVCTLR，</p>

11.4.20 TIM1连续模式的DMA地址(TIMx DAR)

偏移地址: 0x4C

复位值: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAB[15:0]															
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
位15:0	DMAB[15:0]: DMA连续传送寄存器 (DMA register for burst accesses) 对TIMx_DMAR寄存器的读或写会导致对以下地址所在寄存器的存取操作： TIMx_CTLR1 地 址 + DBA + DMA 索 引 ， 其 中 ： “TIMx_CTLR1地址”是控制寄存器1(TIMx_CTLR1)所在的地 址； “DBA”是TIMx_DCR寄存器中定义的基地址； “DMA索引”是由DMA自动控制的偏移量，它取决于TIMx_DCR寄存器中定义的DBL。														

11.4.21 TIM1寄存器图

下表中将TIM1的所有寄存器映射到一个16位可寻址(编址)空间。

表76 TIM1-寄存器图和复位值

偏移	寄存器	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
000h	TIMx_CTLR1	保留																						CKD [1:0]		ARPE	CMS [1:0]		DIR	OPM	URS	UDIS	CEN			
	复位值																							0	0	0	0	0	0	0	0	0				
004h	TIMx_CTLR2	保留																		OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TIMS		MMS[2:0]		CCDS	CCUS	保留	CCPC		
	复位值																			0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0
008h	TIMx_SLVCTLR	保留																		ETP	ECE	ETPS [1:0]		ETF[3:0]		MSM	TS[2:0]		保留	SMS[2:0]						
	复位值																			0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0
00Ch	TIMx_DIER	保留																		TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UIE		
	复位值																			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010h	TIMx_SR	保留																						CC4OF	CC3OF	CC2OF	CC1OF	保留	BIF	TIF	COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UIF
	复位值																							0	0	0	0		0	0	0	0	0	0	0	0
014h	TIMx_EVTGR	保留																						BG	TG	COMG	CC4G	CC3G	CC2G	CC1G	UG					
	复位值																							0	0	0	0	0	0	0	0	0	0			
018h	TIMx_CCMR1 输出比较模式	保留																		OC2CE	OC2M [2:0]		OC2PE	OC2FE	CC2S [1:0]		OC1CE	OC1M [2:0]		OC1PE	OC1FE	CC1S [1:0]				
	复位值																			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIMx_CCMR1 输入比较模式	保留																		IC2F[3:0]		IC2 PSC [1:0]	CC2S [1:0]		IC1F[3:0]		IC1PS C[1:0]		CC1S [1:0]							
	复位值																			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
01Ch	TIMx_CCMR2 输出比较模式	保留																		OC4CE	OC4M [2:0]		OC4PE	OC4FE	CC4S [1:0]		OC3CE	OC3M [2:0]		OC3PE	OC3FE	CC3S [1:0]				
	复位值																			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIMx_CCMR2 输入比较模式	保留																		IC4F[3:0]		IC4 PSC [1:0]	CC4S [1:0]		IC3F[3:0]		IC3PS C[1:0]		CC3S [1:0]							
	复位值																			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
020h	TIMx_CCER	保留																		CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E			
	复位值																			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
024h	TIMx_CNTR	保留																		CNT[15:0]																
	复位值																			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
028h	TIMx_PSCR	保留																		PSC[15:0]																
	复位值																			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

偏移	寄存器	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
02Ch	TIMx_ARLR	保留																ARR[15:0]															
	复位值																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
030h	TIMx_RCNTR	保留																						REP[7:0]									
	复位值																							0	0	0	0	0	0	0	0		
034h	TIMx_CCR1	保留																CCR1[15:0]															
	复位值																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
038h	TIMx_CCR2	保留																CCR2[15:0]															
	复位值																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
03Ch	TIMx_CCR3	保留																CCR3[15:0]															
	复位值																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
040h	TIMx_CCR4	保留																CCR4[15:0]															
	复位值																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
044h	TIMx_BDTR	保留																MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK [1:0]	DTG[7:0]								
	0																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
048h	TIMx_DAR	保留																				DBL[4:0]				保留				DBA[4:0]			
	0																					0	0	0	0					0	0	0	0
04Ch	TIMx_DAR	保留																DMAB[15:0]															
	复位值																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

关于寄存器的起始地址，参见表1

12 通用定时器(TIMx)

12.1 TIMx简介

通用定时器是一个通过可编程预分频器驱动的16位自动装载计数器构成。

它适用于多种场合，包括测量输入信号的脉冲长度(输入捕获)或者产生输出波形(输出比较和PWM)。

使用定时器预分频器和RCC时钟控制器预分频器，脉冲长度和波形周期可以在几个微秒到几个毫秒间调整。

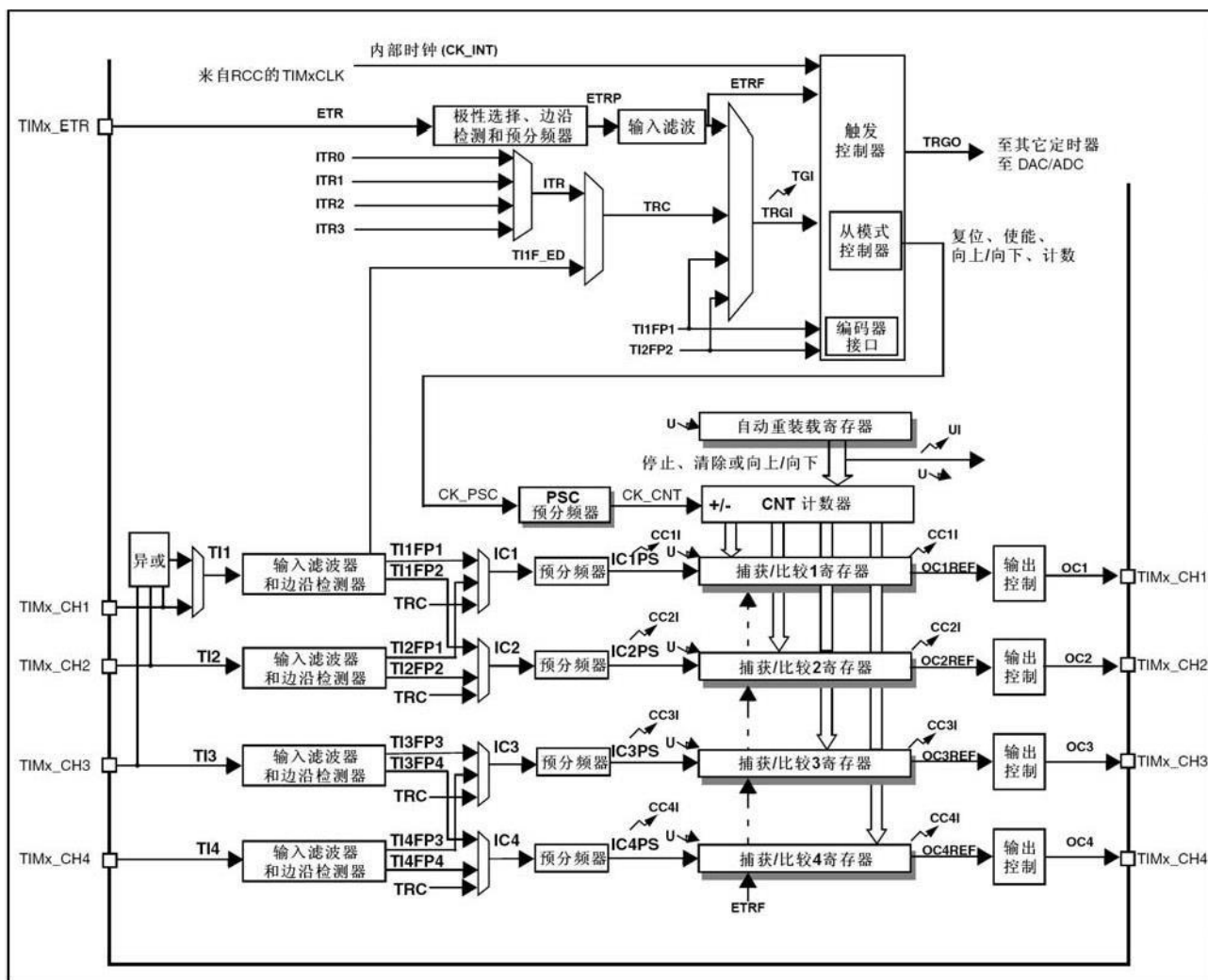
每个定时器都是完全独立的，没有互相共享任何资源。它们可以一起同步操作，参见12.3.15节。


12.2 TIMx主要功能

通用TIMx (TIM2、TIM3和TIM4)定时器功能包括：

- 16位向上、向下、向上/向下自动装载计数器
- 16位可编程(可以实时修改)预分频器，计数器时钟频率的分频系数为1~65536之间的任意数值
- 4个独立通道：
 - 输入捕获
 - 输出比较
 - PWM生成(边缘或中间对齐模式)
 - 单脉冲模式输出
- 使用外部信号控制定时器和定时器互连的同步电路
- 如下事件发生时产生中断/DMA：
 - 更新：计数器向上溢出/向下溢出，计数器初始化(通过软件或者内部/外部触发)
 - 触发事件(计数器启动、停止、初始化或者由内部/外部触发计数)
 - 输入捕获
 - 输出比较
- 支持针对定位的增量(正交)编码器和霍尔传感器电路
- 触发输入作为外部时钟或者按周期的电流管理

图98 通用定时器框图



注：  根据控制位的设定，在U事件时传送预加载寄存器的内容至工作寄存器事件

 中断和DMA输出

12.3 TIMx功能描述

12.3.1 时基单元

可编程通用定时器的主要部分是一个16位计数器和与其相关的自动装载寄存器。这个计数器可以向上计数、向下计数或者向上向下双向计数。此计数器时钟由预分频器分频得到。

计数器、自动装载寄存器和预分频器寄存器可以由软件读写，在计数器运行时仍可以读写。时基单元包含：

- 计数器寄存器(TIMx_CNTR)
- 预分频器寄存器(TIMx_PSCR)
- 自动装载寄存器(TIMx_ARLR)

自动装载寄存器是预先装载的，写或读自动重载寄存器将访问预装载寄存器。根据在TIMx_CTLR1寄存器中的自动装载预装载使能位(ARPE)的设置，预装载寄存器的内容被立即或在每次的更新事件UEV时传送到影子寄存器。当计数器达到溢出条件(向下计数时的下溢条件)并当TIMx_CTLR1寄存器中的UDIS位等于'0'时，产生更新事件。更

新事件也可以由软件产生。随后会详细描述每一种配置下更新事件的产生。

计数器由预分频器的时钟输出CK_CNT驱动，仅当设置了计数器TIMx_CTLR1寄存器中的计数器使能位(CEN)时，CK_CNT才有效。(有关计数器使能的细节，请参见控制器的从模式描述)。

注：真正的计数器使能信号CNT_EN是在CEN的一个时钟周期后被设置。

预分频器描述

预分频器可以将计数器的时钟频率按1到65536之间的任意值分频。它是基于一个(在TIMx_PSCR寄存器中的)16位寄存器控制的16位计数器。这个控制寄存器带有缓冲器，它能够在工作时被改变。新的预分频器参数在下一次更新事件到来时被采用。

图99和图100给出了在预分频器运行时，更改计数器参数的例子。

图99 当预分频器的参数从1变到2时，计数器的时序图

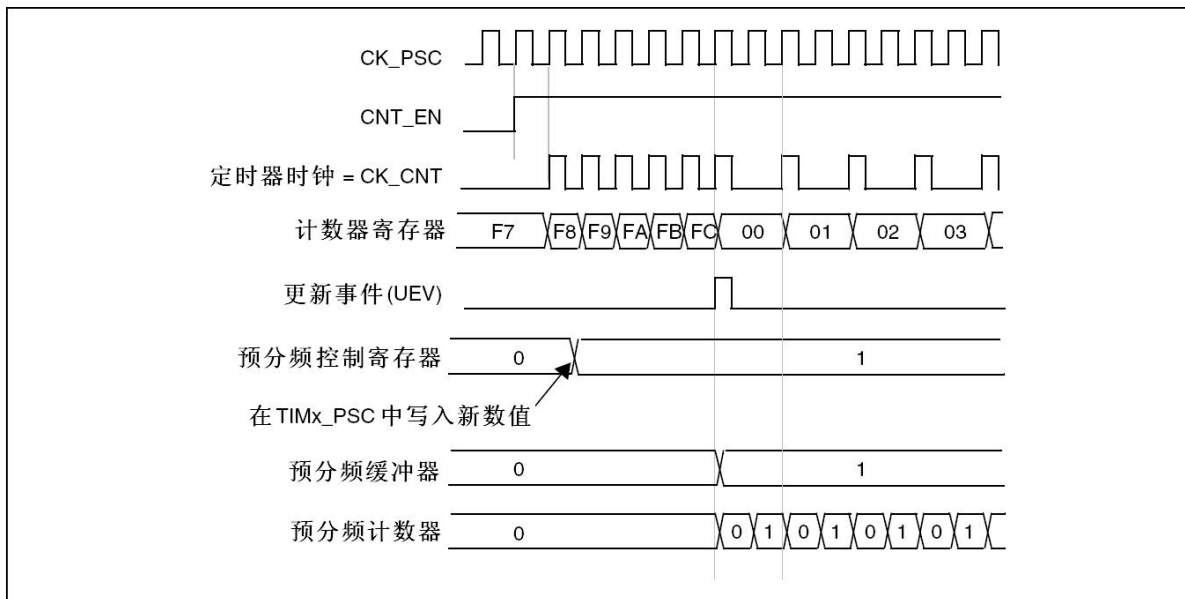
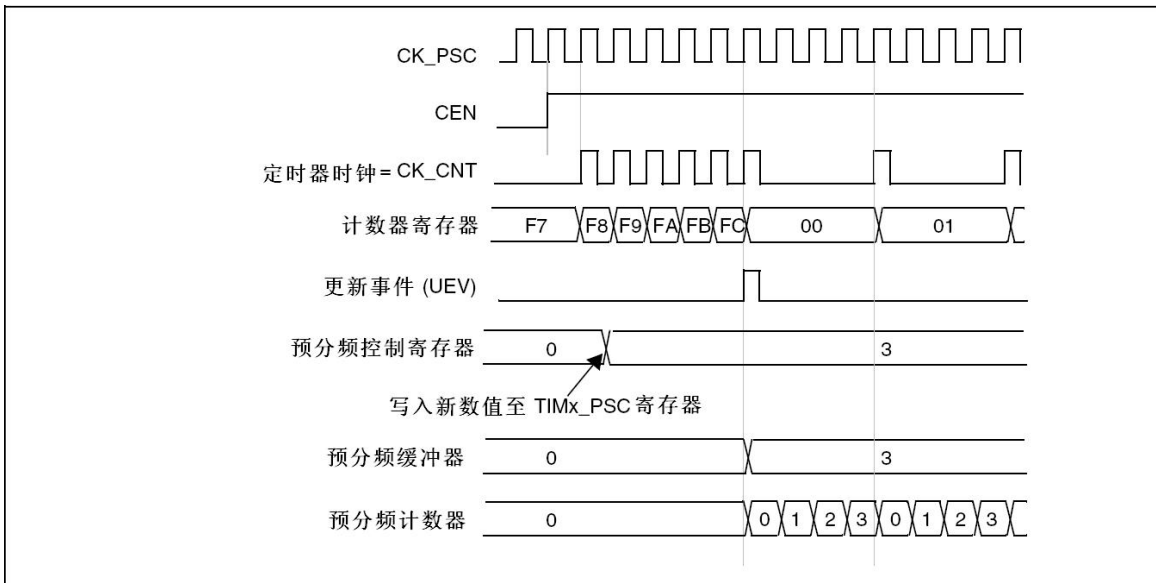


图100 当预分频器的参数从1变到4时，计数器的时序图



12.3.2 计数器模式

向上计数模式

在向上计数模式中，计数器从0计数到自动加载值(TIMx_ARRL寄存器内容)，然后重新从0开始计数并且产生一个计数器溢出事件。

每次计数器溢出时可以产生更新事件，在TIMx_EVTGR寄存器中(通过软件方式或者使用从模式控制器)设置UG位也同样可以产生一个更新事件。

设置TIMx_CTLR1寄存器中的UDIS位，可以禁止更新事件；这样可以避免在向预装载寄存器中写入新值时更新影子寄存器。在UDIS位被清'0'之前，将不产生更新事件。但是在应该产生更新事件时，计数器仍会被清'0'，同时预分频器的计数也被清0(但预分频系数不变)。此外，如果设置了TIMx_CTLR1寄存器中的URS位(选择更新请求)，设置UG位将产生一个更新事件UEV，但硬件不设置UIF标志(即不产生中断或DMA请求)；这是为了避免在捕获模式下清除计数器时，同时产生更新和捕获中断。

当发生一个更新事件时，所有的寄存器都被更新，硬件同时(依据URS位)设置更新标志位(TIMx_SR寄存器中的UIF位)。

- 预分频器的缓冲区被置入预装载寄存器的值(TIMx_PSCR寄存器的内容)。
- 自动装载影子寄存器被重新置入预装载寄存器的值(TIMx_ARRL)。

下图给出一些例子，当TIMx_ARRL=0x36时计数器在不同时钟频率下的动作。

图101 计数器时序图，内部时钟分频因子为1

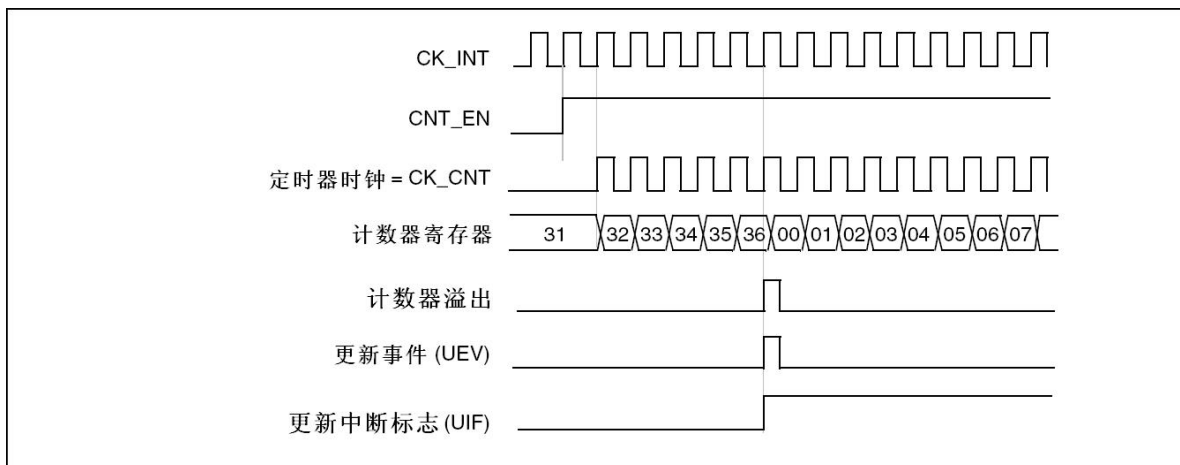


图102 计数器时序图，内部时钟

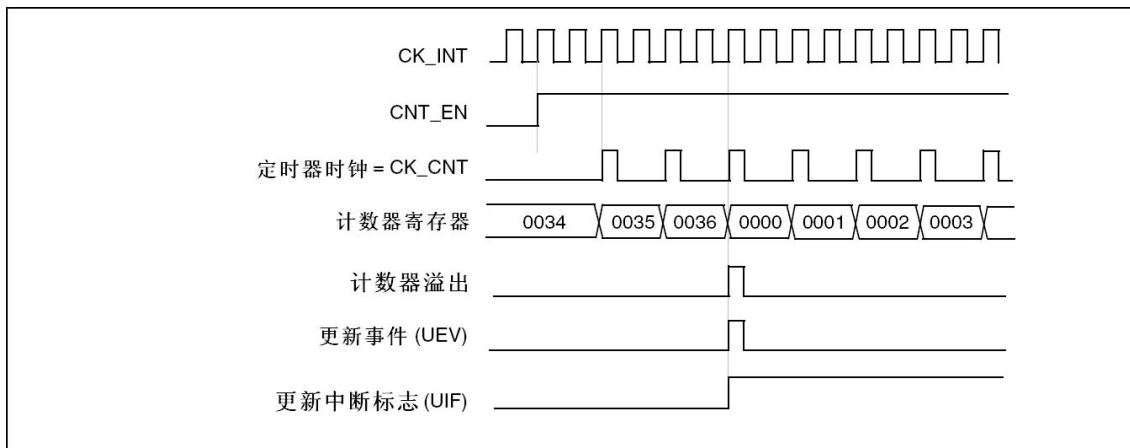


图103 计数器时序图，内部时钟分频因子为4

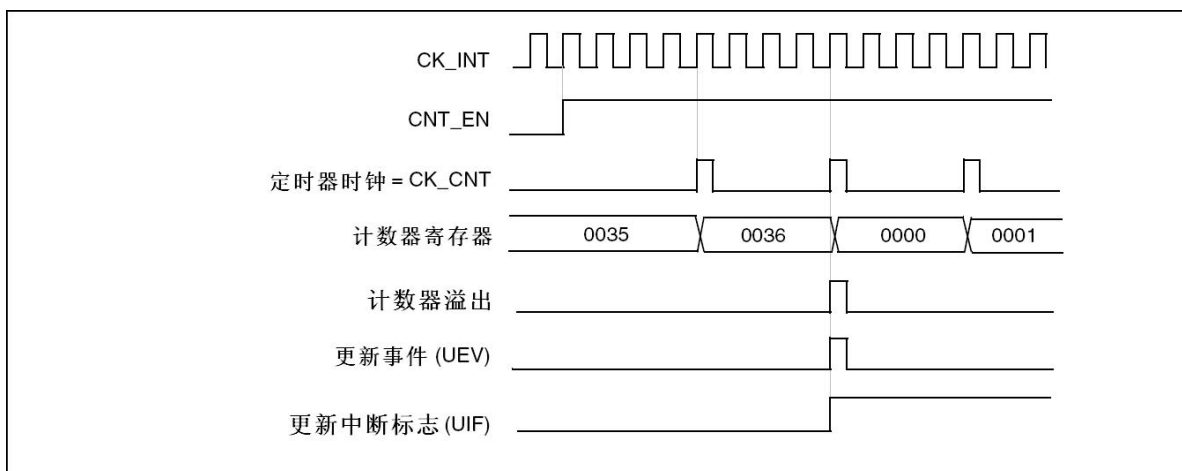


图104 计数器时序图，内部时钟分频因子为N

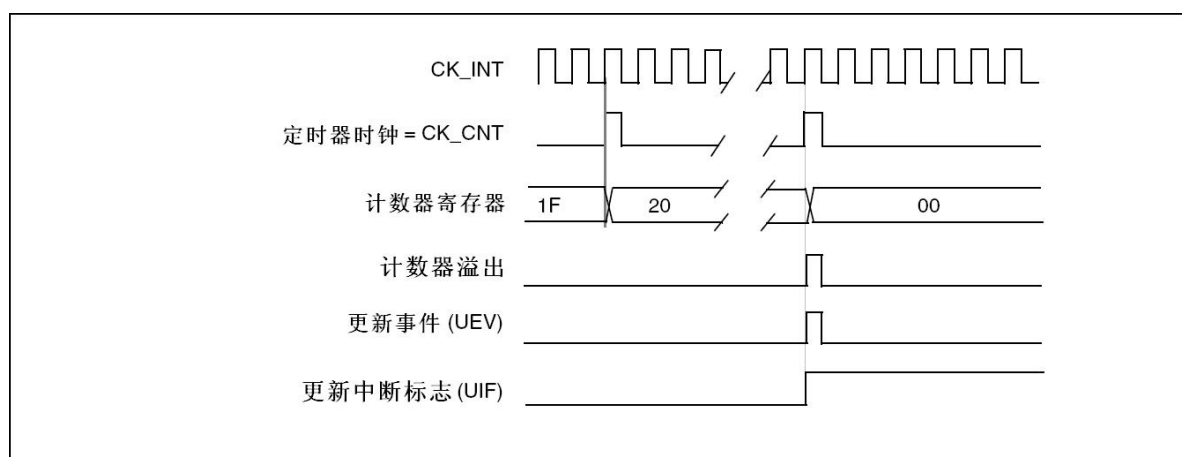
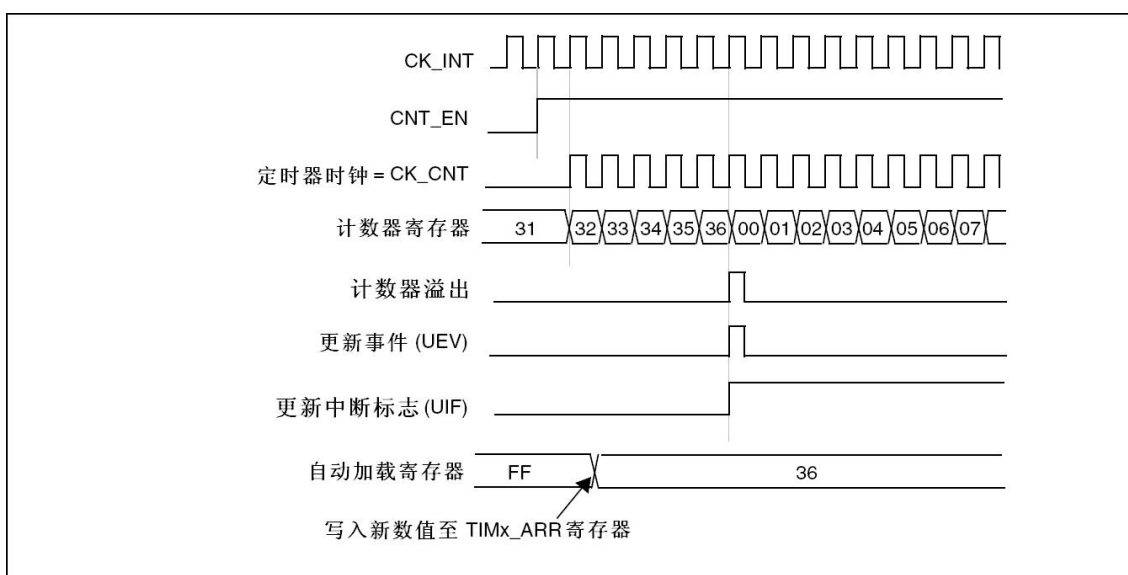
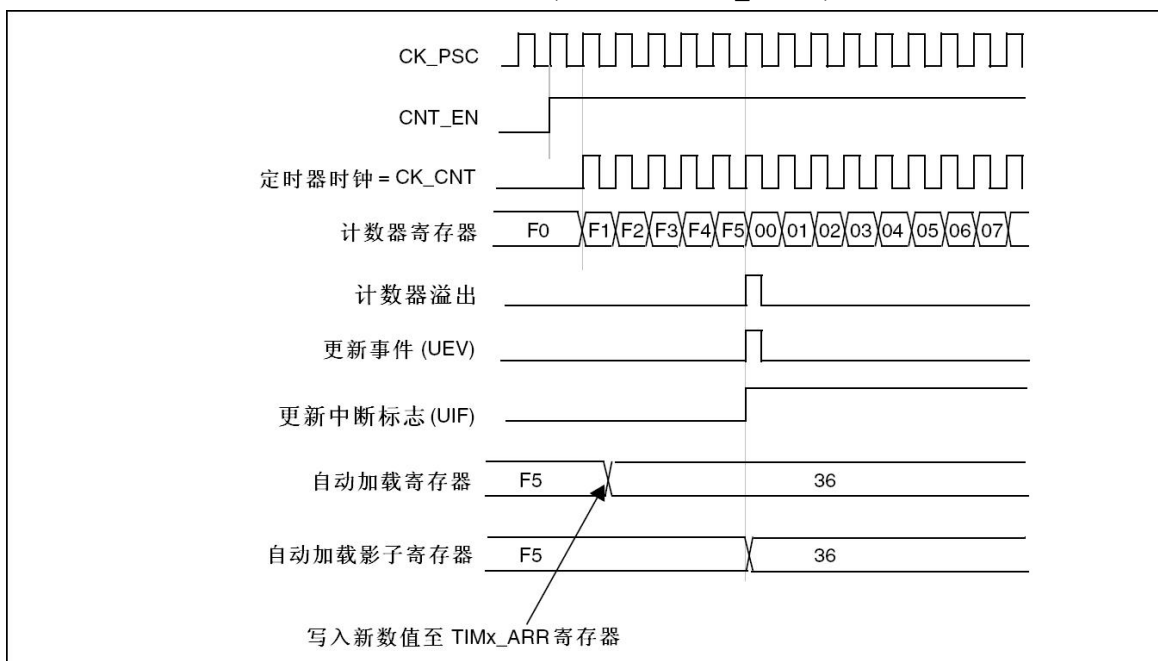


图105 计数器时序图，当ARPE=0时的更新事



106 计数器时序图, 当ARPE=1时的更新事件(预装入了TIMx_ARRL)



向下计数模式

在向下模式中, 计数器从自动装入的值(TIMx_ARRL计数器的值)开始向下计数到0, 然后从自动装入的值重新开始并且产生一个计数器向下溢出事件。

每次计数器溢出时可以产生更新事件, 在TIMx_EVTGR寄存器中(通过软件方式或者使用从模式控制器)设置UG位, 也同样可以产生一个更新事件。

设置TIMx_CTLR1寄存器的UDIS位可以禁止UEV事件。这样可以避免向预装载寄存器中写入新值时更新影子寄存器。因此UDIS位被清为'0'之前不会产生更新事件。然而, 计数器仍会从当前自动加载值重新开始计数, 同时预分频器的计数器重新从0开始(但预分频系数不变)。

此外, 如果设置了TIMx_CTLR1寄存器中的URS位(选择更新请求), 设置UG位将产生一个更新事件UEV但不设置UIF标志(因此不产生中断和DMA请求), 这是为了避免在发生捕获事件并清除计数器时, 同时产生更新和捕获中断。

当发生更新事件时, 所有的寄存器都被更新, 并且(根据URS位的设置)更新标志位(TIMx_SR寄存器中的UIF位)也被设置。

- 预分频器的缓存器被置入预装载寄存器的值(TIMx_PSCR寄存器的值)。
- 当前的自动加载寄存器被更新为预装载值(TIMx_ARRL寄存器中的内容)。注: 自动装载在计数器重载入之前被更新, 因此下一个周期将是预期的值。

以下是一些当TIMx_ARRL=0x36时, 计数器在不同时钟频率下的操作例子。

图107 计数器时序图，内部时钟分频因子为1

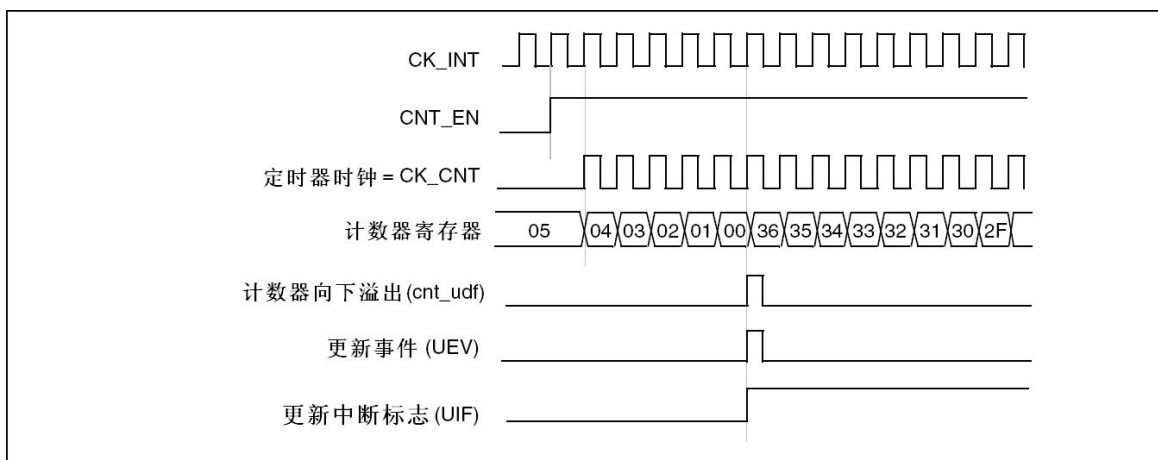


图108 计数器时序图，内部时钟分频因子为2

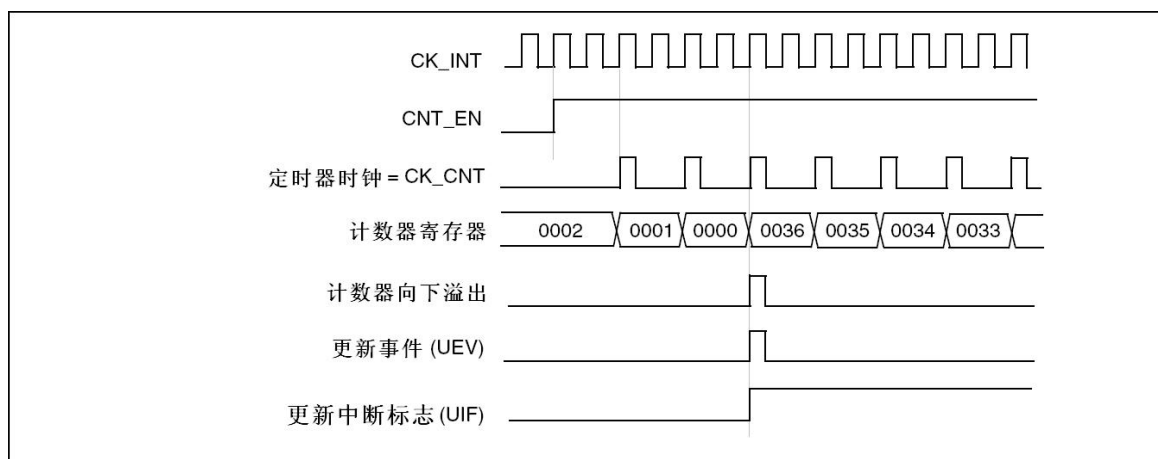


图109 计数器时序图，内部时钟分频因子为4

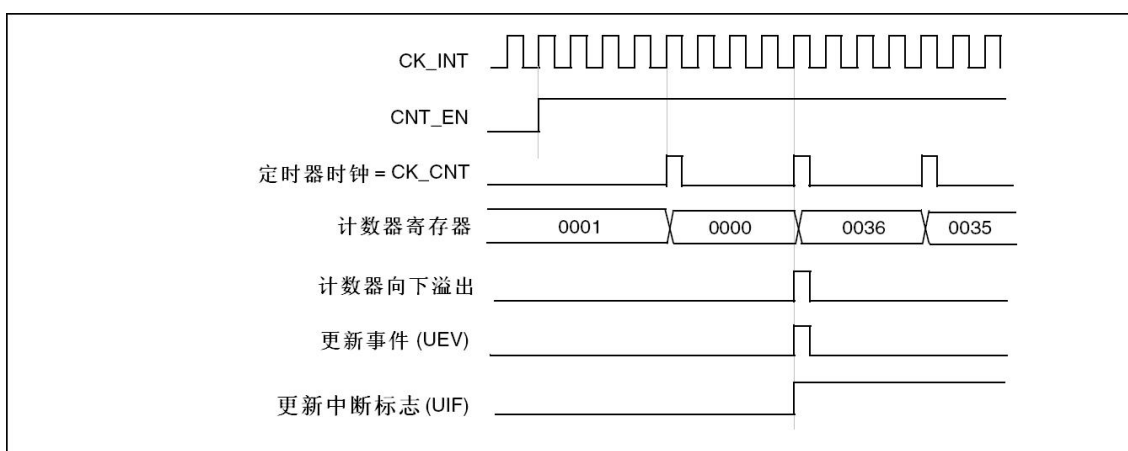


图110 计数器时序图，内部时钟分频因子为N

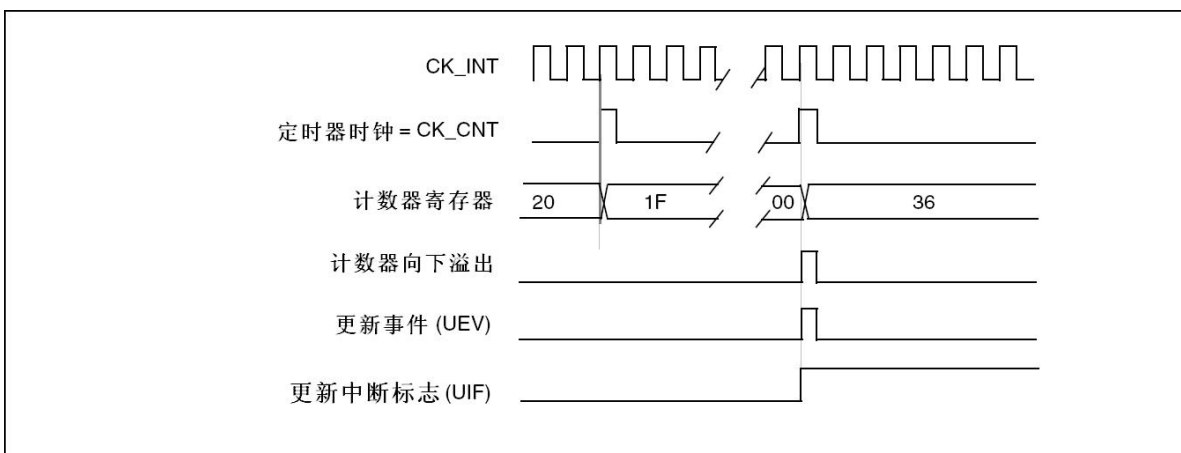
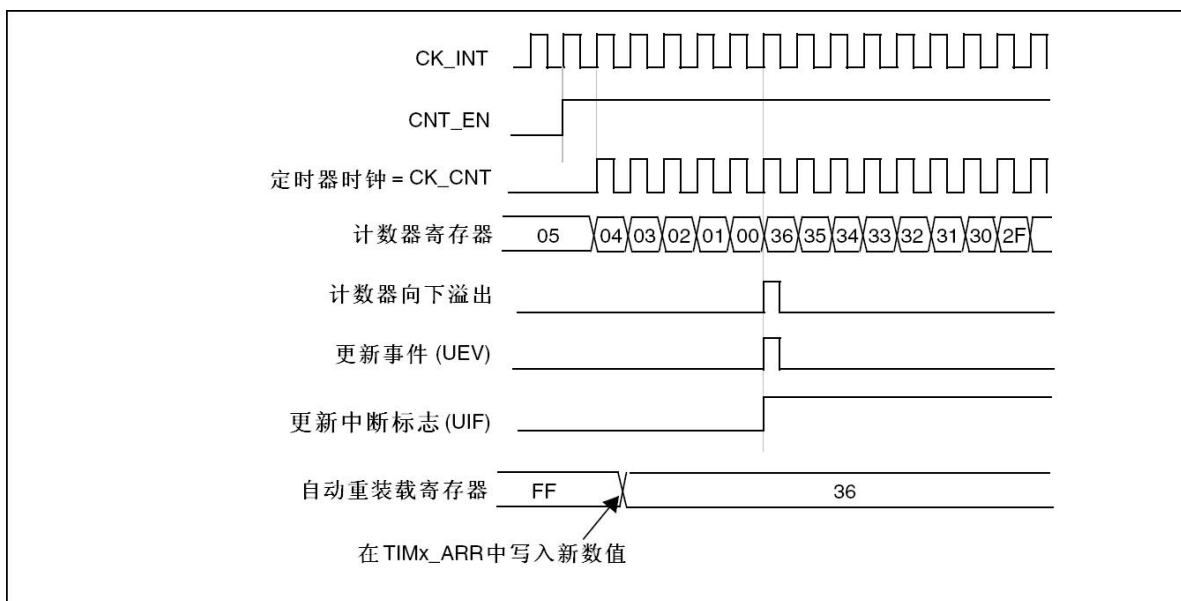


图111 计数器时序图，当没有使用重复计数器时的更新事件



中央对齐模式(向上/向下计数)

在中央对齐模式，计数器从0开始计数到自动加载的值(TIMx_ARRL寄存器)-1，产生一个计数器溢出事件，然后向下计数到1并且产生一个计数器下溢事件；然后再从0开始重新计数。

在这个模式，不能写入TIMx_CTLR1中的DIR方向位。它由硬件更新并指示当前的计数方向。可以在每次计数上溢和每次计数下溢时产生更新事件；也可以通过(软件或者使用从模式控制器) 设置TIMx_EVTGR寄存器中的UG位产生更新事件。然后，计数器重新从0开始计数，预分频器也重新从0开始计数。

设置TIMx_CTLR1寄存器中的UDIS位可以禁止UEV事件。这样可以避免在向预装载寄存器中写入新值时更新影子寄存器。因此UDIS位被清为'0'之前不会产生更新事件。然而，计数器仍会根据当前自动重加载的值，继续向上或向下计数。

此外，如果设置了TIMx_CTLR1寄存器中的URS位(选择更新请求)，设置UG位将产生一个更新事件UEV但不设置UIF标志(因此不产生中断和DMA请求)，这是为了避免在发生捕获事件并清除计数器时，同时产生更新和捕获中断。

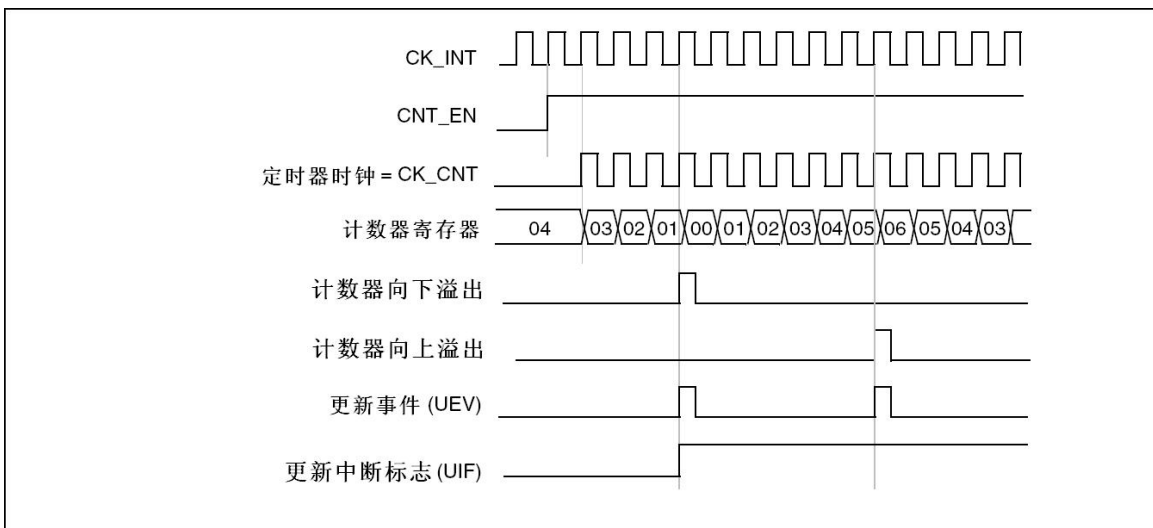
当发生更新事件时，所有的寄存器都被更新，并且(根据URS位的设置)更新标志位(TIMx_SR寄存器中的UIF位)也被设置。

- 预分频器的缓存器被加载为预装载(TIMx_PSCRR寄存器)的值。

当前的自动加载寄存器被更新为预装载值(TIMx_ARLR寄存器中的内容)。注：如果因为计数器溢出而产生更新，自动重装载将在计数器重载入之前被更新，因此下一个周期将是预期的值(计数器被装载为新的值)。

以下是一些计数器在不同时钟频率下的操作的例子：

图112 计数器时序图，内部时钟分频因子为1，TIMx_ARLR=0x6



1. 这里使用了中心对齐模式1(详见14.4.1节)。

图113 计数器时序图，内部时钟分频因子为2

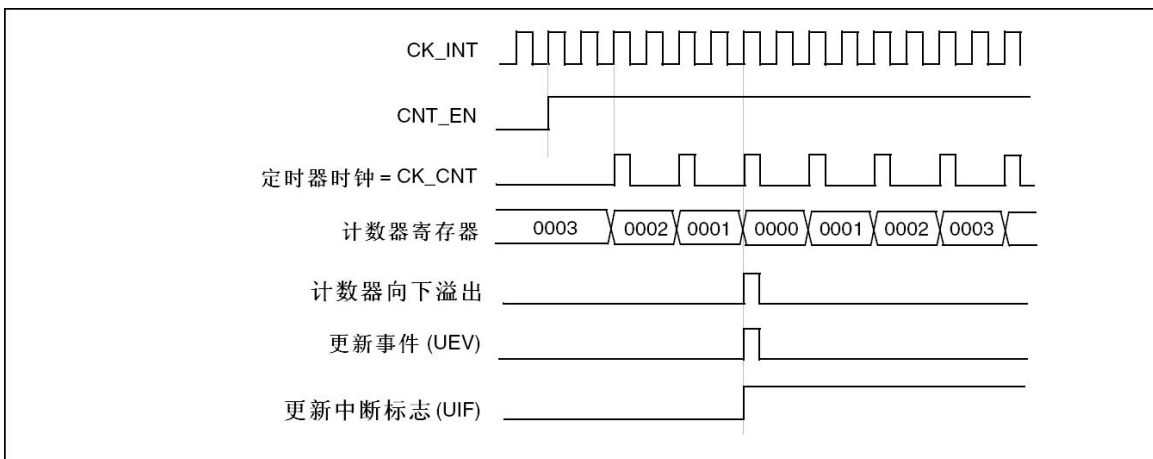


图114 计数器时序图，内部时钟分频因子为4，TIMx_ARLR=0x36

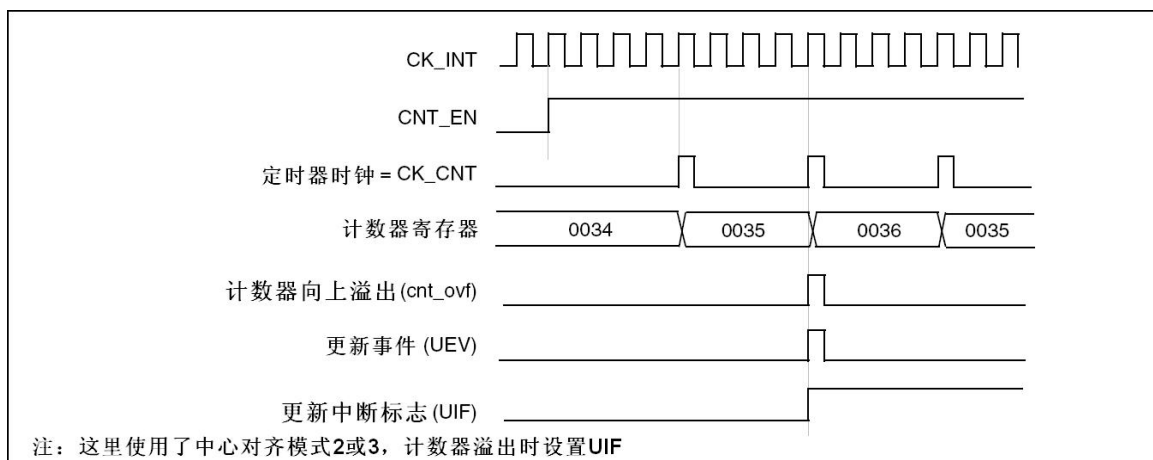


图115 计数器时序图，内部时钟分频因子为N

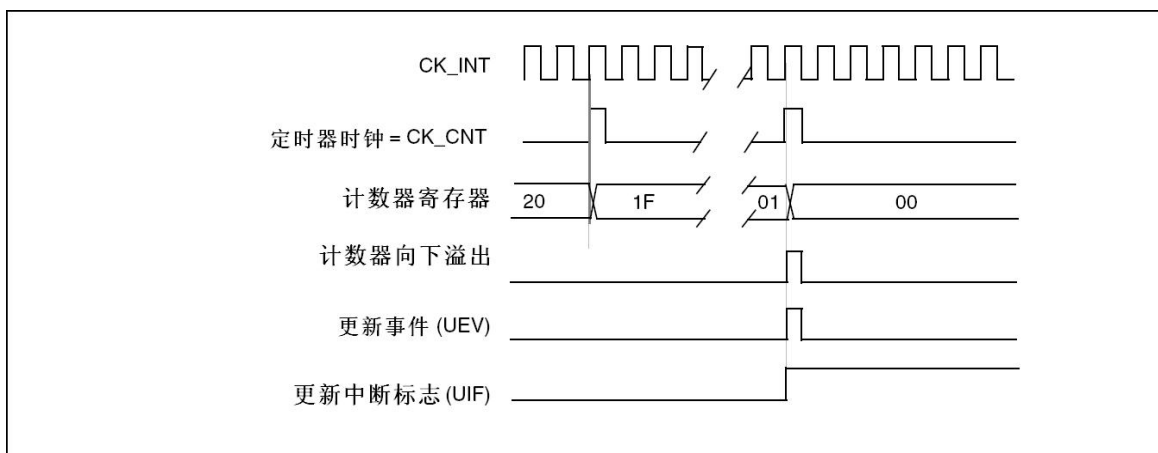


图116 计数器时序图，ARPE=1时的更新事件(计数器下溢)

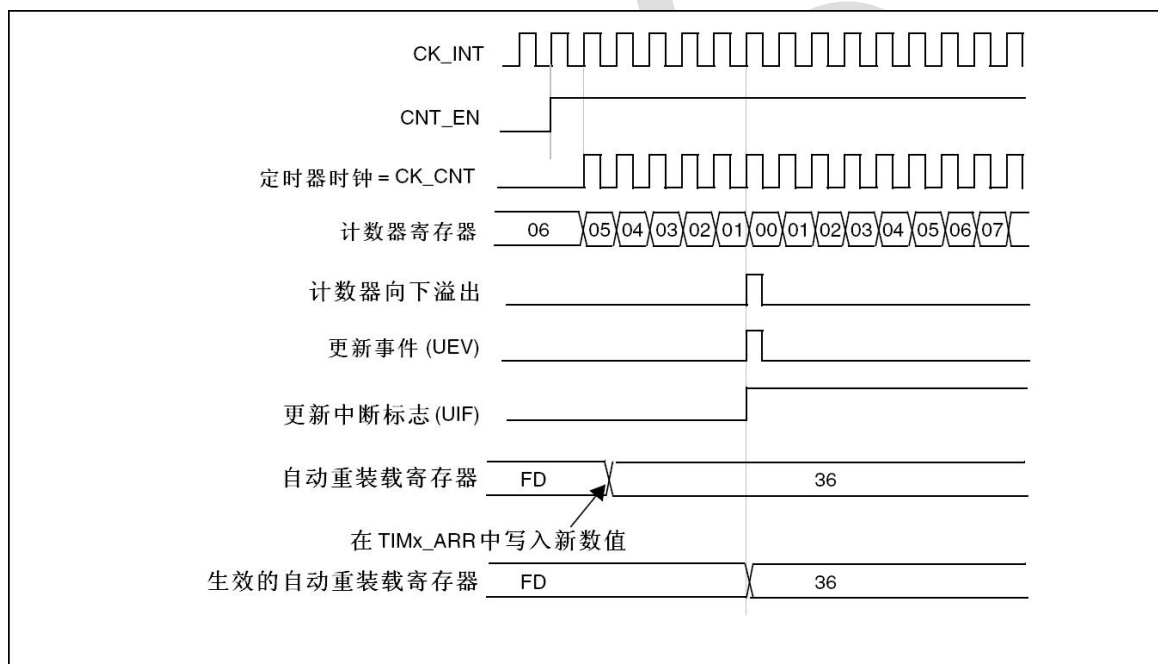
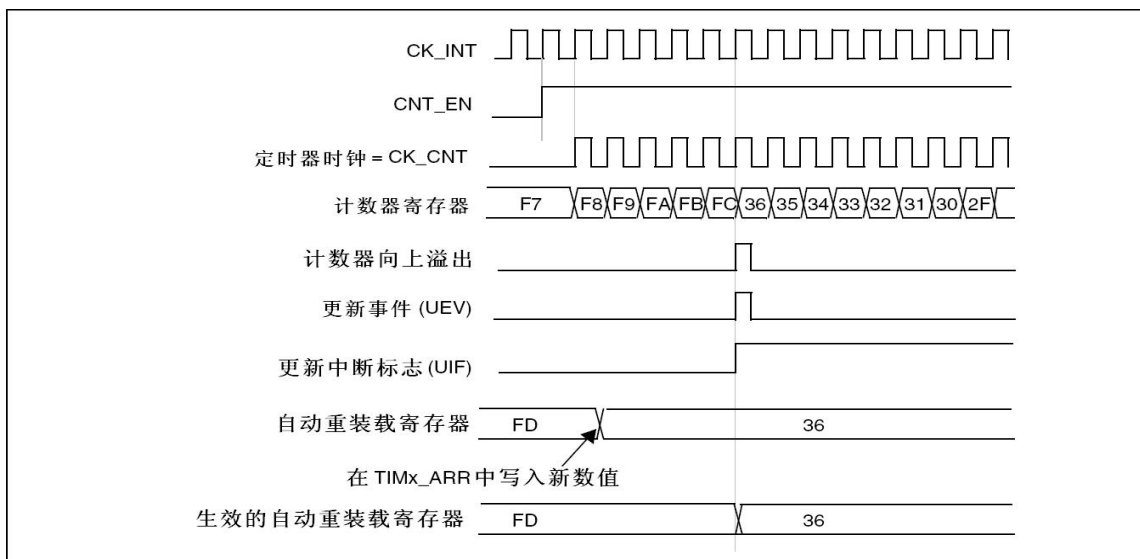


图117 计数器时序图，ARPE=1时的更新事件(计数器溢出)



12.3.3 时钟选择

计数器时钟可由下列时钟源提供:

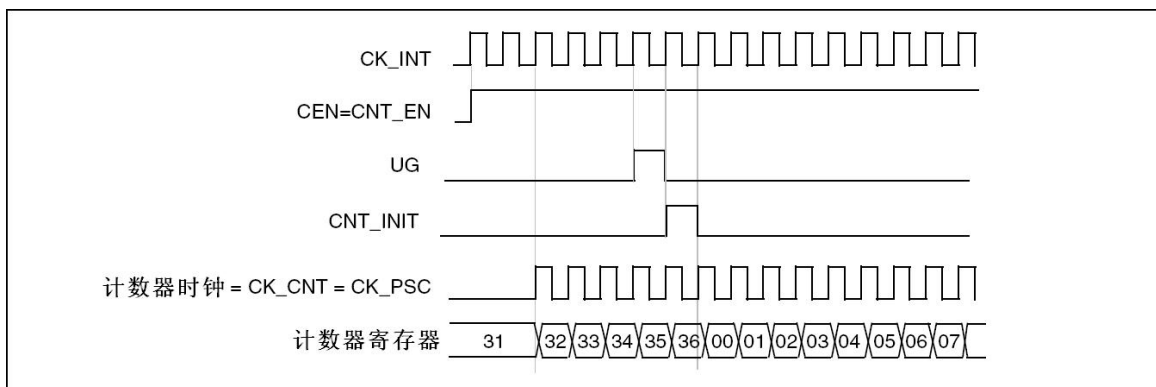
- 内部时钟(CK_INT)
- 外部时钟模式1: 外部输入脚(TIx)
- 外部时钟模式2: 外部触发输入(ETR)
- 内部触发输入(ITRx): 使用一个定时器作为另一个定时器的预分频器, 如可以配置一个定时器Timer1而作为另一个定时器Timer2的预分频器。参见12.3.15。

内部时钟源(CK_INT)

如果禁止了从模式控制器 (TIMx_SLVCTL 寄存器的 SMS=000), 则 CEN、DIR(TIMx_CTLR1寄存器) 和UG位(TIMx_EVTGR寄存器)是事实上的控制位, 并且只能被软件修改(UG位仍被自动清除)。只要CEN位被写成'1', 预分频器的时钟就由内部时钟CK_INT提供。

下图显示了控制电路和向上计数器在一般模式下, 不带预分频器时的操作。

图118 一般模式下的控制电路, 内部时钟分频因子为1

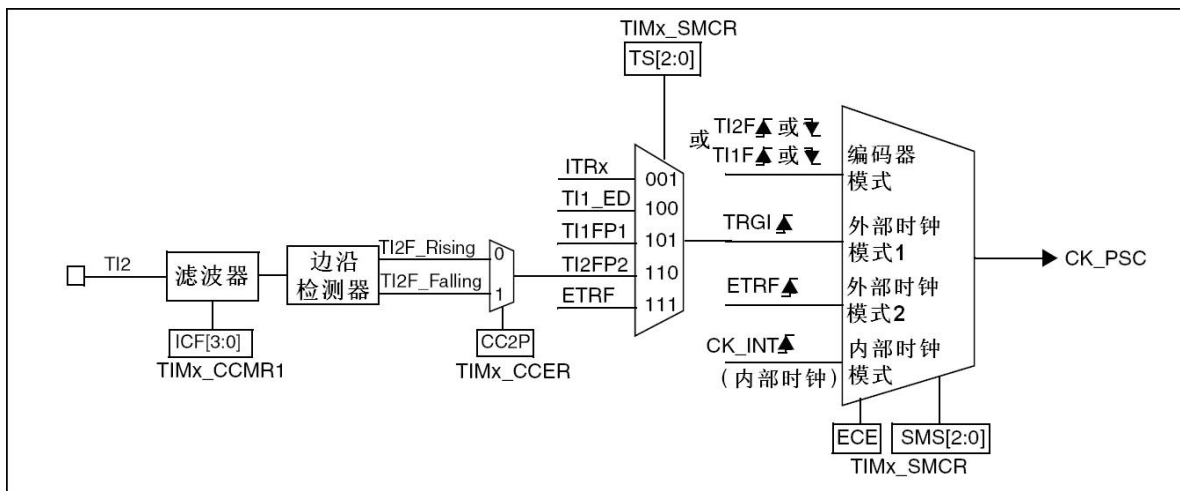


外部时钟源模式1

当TIMx_SLVCTL寄存器的SMS=111时, 此模式被选中。

计数器可以在选定输入端的每个上升沿或下降沿计数。

图119 TI2外部时钟连接例子



例如, 要配置向上计数器在TI2输入端的上升沿计数, 使用下列步骤:

1. 配置TIMx_CCMR1寄存器CC2S='01', 配置通道2检测TI2输入的上升沿
2. 配置TIMx_CCMR1寄存器的IC2F[3:0], 选择输入滤波器带宽(如果不需要滤波器, 保持

IC2F=0000)

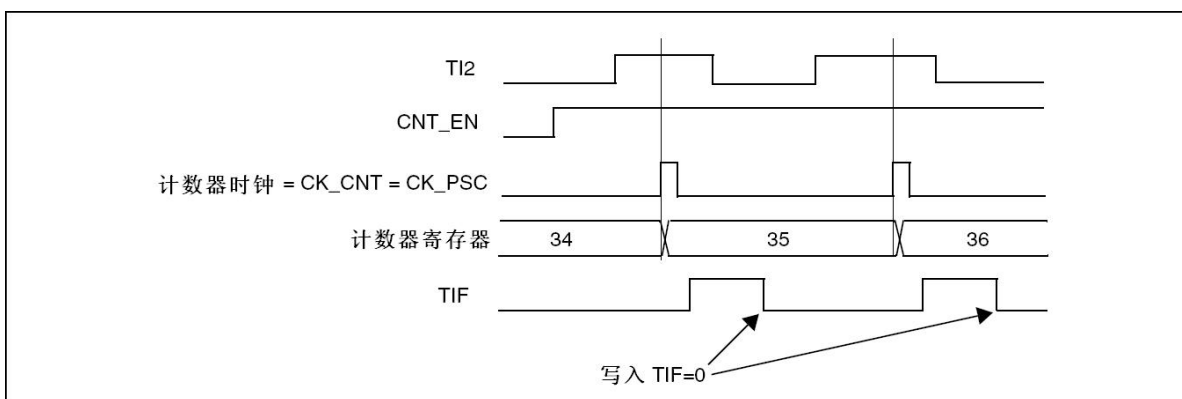
注: 捕获预分频器不用作触发, 所以不需要对它进行配置

3. 配置TIMx_CCER寄存器的CC2P='0', 选定上升沿极性
4. 配置TIMx_SLVCTLRL寄存器的SMS='111', 选择定时器外部时钟模式1
5. 配置TIMx_SLVCTLRL寄存器中的TS='110', 选定TI2作为触发输入源
6. 设置TIMx_CTLRL寄存器的CEN='1', 启动计数器

当上升沿出现在TI2, 计数器计数一次, 且TIF标志被设置。

在TI2的上升沿和计数器实际时钟之间的延时, 取决于在TI2输入端的重新同步电路。

图120 外部时钟模式1下的控制电路



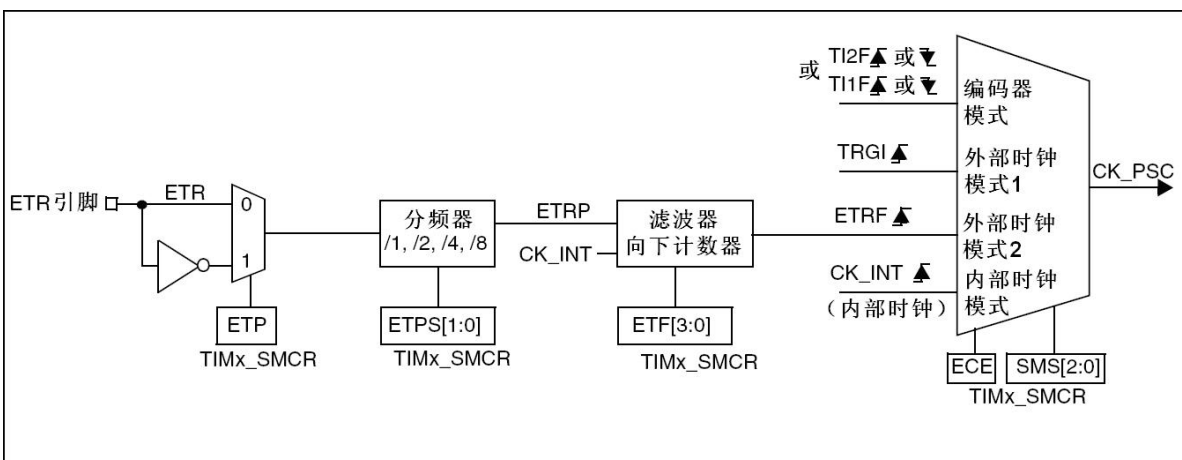
外部时钟源模式2

选定此模式的方法为: 令TIMx_SLVCTLRL寄存器中的ECE=1

计数器能够在外部触发ETR的每一个上升沿或下降沿计数。

下图是外部触发输入的框图

图121 外部触发输入框图



例如, 要配置在ETR下每2个上升沿计数一次的向上计数器, 使用下列步骤:

1. 本例中不需要滤波器, 置TIMx_SLVCTLRL寄存器中的ETF[3:0]=0000
2. 设置预分频器, 置TIMx_SLVCTLRL寄存器中的ETPS[1:0]=01
3. 设置在ETR的上升沿检测, 置TIMx_SLVCTLRL寄存器中的ETP=0
4. 开启外部时钟模式2, 置TIMx_SLVCTLRL寄存器中的ECE=1
5. 启动计数器, 置TIMx_CTLRL寄存器中的CEN=1 计数器在每2个ETR上升沿计数一次。

在ETR的上升沿和计数器实际时钟之间的延时取决于在ETRP信号端的重新同步电路。