

14 高级控制定时器（TIM1 和 TIM8）

除非特别说明，否则本部分适用于整个 STM32F4xx 系列。

14.1 TIM1 和 TIM8 简介

高级控制定时器（TIM1 和 TIM8）包含一个 16 位自动重载计数器，该计数器由可编程预分频器驱动。

此类定时器可用于各种用途，包括测量输入信号的脉冲宽度（输入捕获），或者生成输出波形（输出比较、PWM 和带死区插入的互补 PWM）。

使用定时器预分频器和 RCC 时钟控制器预分频器，可将脉冲宽度和波形周期从几微秒调制到几毫秒。

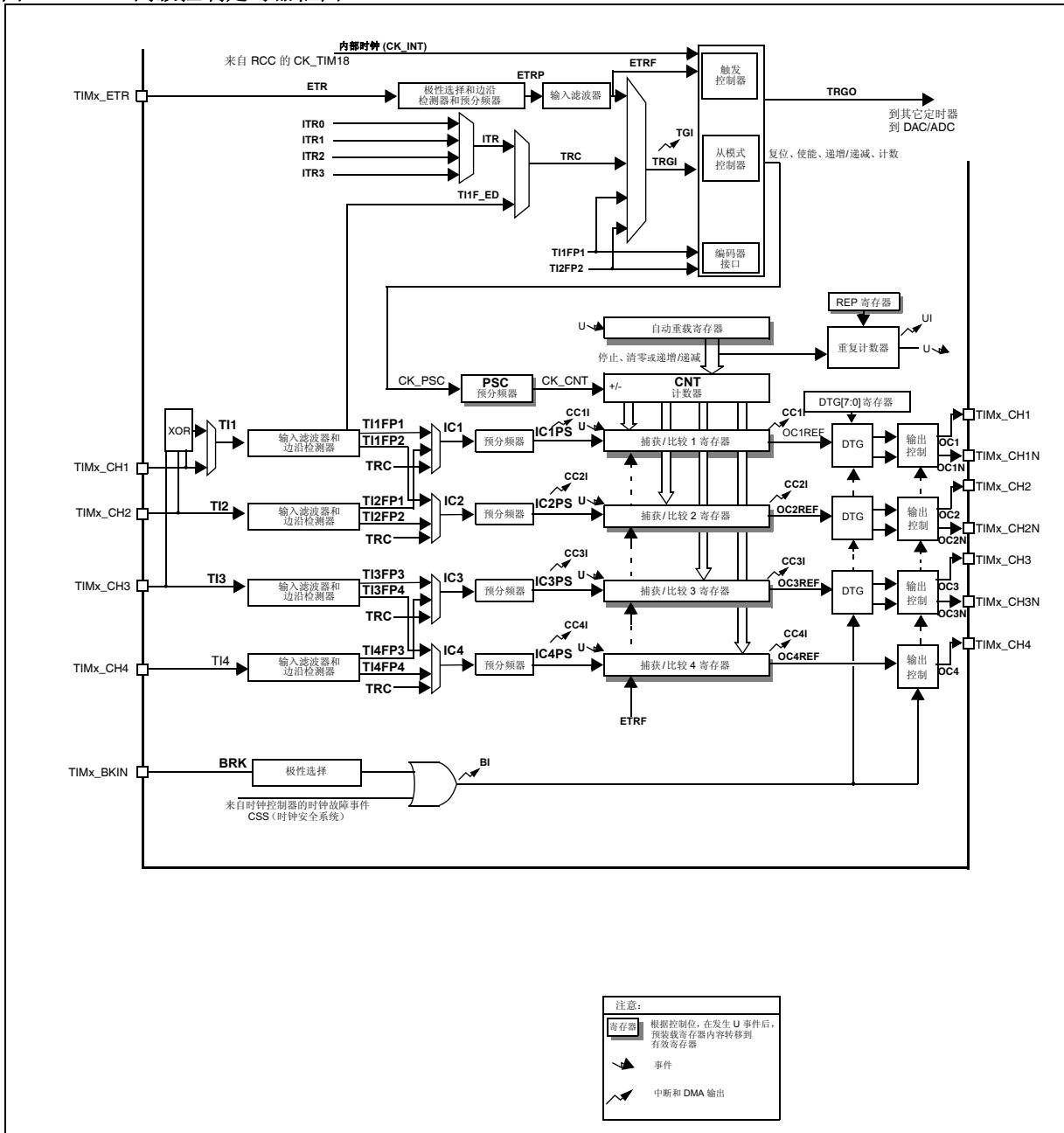
高级控制定时器（TIM1 和 TIM8）和通用（TIMx）定时器彼此完全独立，不共享任何资源。如[第 14.3.20 节](#)中所述，它们可以实现同步。

14.2 TIM1 和 TIM8 主要特性

TIM1 和 TIM8 定时器具有以下特性：

- 16 位递增、递减、递增/递减自动重载计数器。
- 16 位可编程预分频器，用于对计数器时钟频率进行分频（即运行时修改），分频系数介于 1 到 65536 之间。
- 多达 4 个独立通道，可用于：
 - 输入捕获
 - 输出比较
 - PWM 生成（边沿和中心对齐模式）
 - 单脉冲模式输出
- 带可编程死区的互补输出。
- 使用外部信号控制定时器且可实现多个定时器互连的同步电路。
- 重复计数器，用于仅在给定数目的计数器周期后更新定时器寄存器。
- 用于将定时器的输出信号置于复位状态或已知状态的断路输入。
- 发生如下事件时生成中断/DMA 请求：
 - 更新：计数器上溢/下溢、计数器初始化（通过软件或内部/外部触发）
 - 触发事件（计数器启动、停止、初始化或通过内部/外部触发计数）
 - 输入捕获
 - 输出比较
 - 断路输入
- 支持定位用增量（正交）编码器和霍尔传感器电路。
- 外部时钟触发输入或逐周期电流管理。

图 71. 高级控制定时器框图



14.3 TIM1 和 TIM8 功能说明

14.3.1 时基单元

可编程高级控制定时器的主要模块是一个 16 位计数器及其相关的自动重载寄存器。计数器可递增计数、递减计数或交替进行递增和递减计数。计数器的时钟可通过预分频器进行分频。

计数器、自动重载寄存器和预分频器寄存器可通过软件进行读写。即使在计数器运行时也可执行读写操作。

时基单元包括：

- 计数器寄存器 (TIMx_CNT)
- 预分频器寄存器 (TIMx_PSC)
- 自动重载寄存器 (TIMx_ARR)
- 重复计数器寄存器 (TIMx_RCR)

自动重载寄存器是预装载的。对自动重载寄存器执行写入或读取操作时会访问预装载寄存器。预装载寄存器的内容既可以直接传送到影子寄存器，也可以在每次发生更新事件 (UEV) 时传送到影子寄存器，这取决于 TIMx_CR1 寄存器中的自动重载预装载使能位 (ARPE)。当计数器达到上溢值（或者在递减计数时达到下溢值）并且 TIMx_CR1 寄存器中的 UDIS 位为 0 时，将发送更新事件。该更新事件也可由软件产生。下文将针对各配置的更新事件的产生进行详细介绍。

计数器由预分频器输出 CK_CNT 提供时钟，仅当 TIMx_CR1 寄存器中的计数器启动位 (CEN) 置 1 时，才会启动计数器（有关计数器使能的更多详细信息，另请参见从模式控制器的相关说明）。

注意，计数器将在 TIMx_CR1 寄存器的 CEN 位置 1 时刻的一个时钟周期后开始计数。

预分频器说明

预分频器可对计数器时钟频率进行分频，分频系数介于 1 和 65536 之间。该预分频器基于 TIMx_PSC 寄存器中的 16 位寄存器所控制的 16 位计数器。由于该控制寄存器具有缓冲功能，因此可对预分频器进行实时更改。而新的预分频比将在下一更新事件发生时被采用。

[图 72](#) 和 [图 73](#) 以一些示例说明在预分频比实时变化时计数器的行为：

图 72. 预分频器分频由 1 变为 2 时的计数器时序图

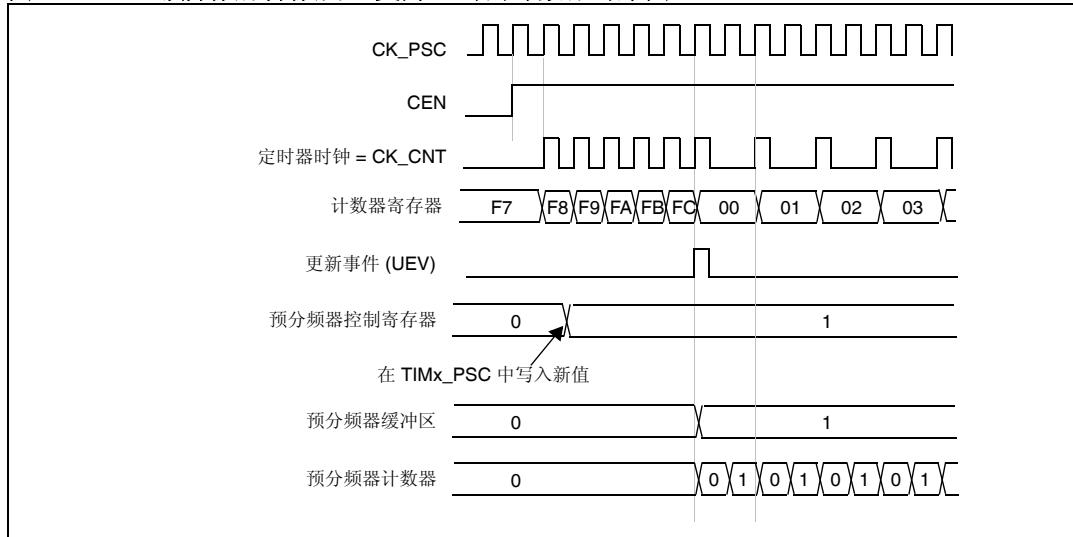
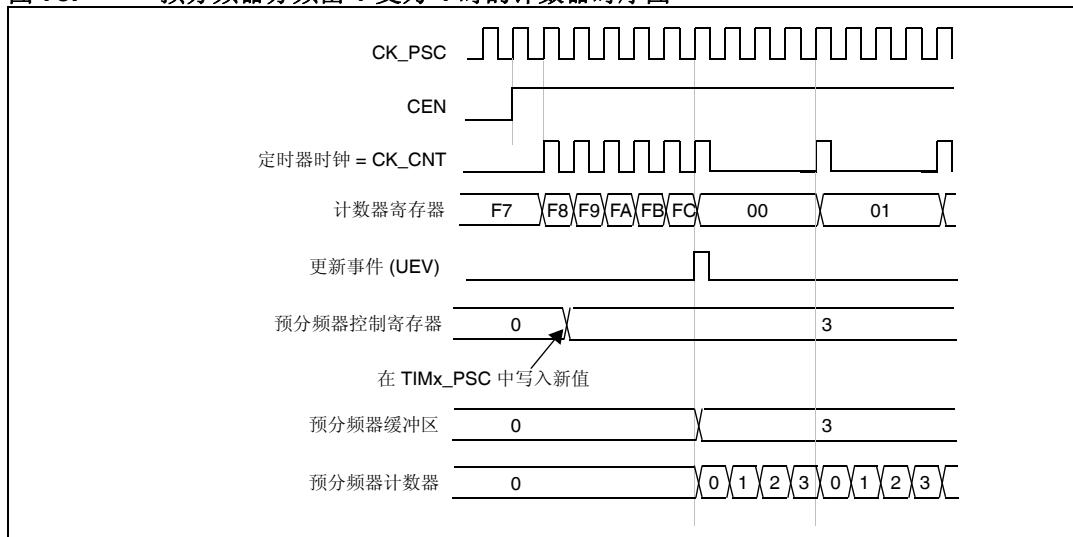


图 73. 预分频器分频由 1 变为 4 时的计数器时序图



14.3.2 计数器模式

递增计数模式

在递增计数模式下，计数器从 0 计数到自动重载值（**TIMx_ARR** 寄存器的内容），然后重新从 0 开始计数并生成计数器上溢事件。

如果使用重复计数器，则当递增计数的重复次数达到重复计数器寄存器中编程的次数加一次（**TIMx_RCR+1**）后，将生成更新事件（**UEV**）。否则，将在每次计数器上溢时产生更新事件。

将 **TIMx_EGR** 寄存器的 **UG** 位置 1（通过软件或使用从模式控制器）时，也将产生更新事件。

通过软件将 **TIMx_CR1** 寄存器中的 **UDIS** 位置 1 可禁止 **UEV** 事件。这可避免向预装载寄存器写入新值时更新影子寄存器。在 **UDIS** 位写入 0 之前不会产生任何更新事件。不过，计数器和预分频器计数器都会重新从 0 开始计数（而预分频比保持不变）。此外，如果 **TIMx_CR1** 寄存器中的 **URS** 位（更新请求选择）已置 1，则将 **UG** 位置 1 会生成更新事件 **UEV**，但不会将 **UIF** 标志置 1（因此，不会发送任何中断或 DMA 请求）。这样一来，如果在发生捕获事件时将计数器清零，将不会同时产生更新中断和捕获中断。

发生更新事件时，将更新所有寄存器且将更新标志（**TIMx_SR** 寄存器中的 **UIF** 位）置 1（取决于 **URS** 位）：

- 重复计数器中将重新装载 **TIMx_RCR** 寄存器的内容
- 自动重载影子寄存器将以预装载值（**TIMx_ARR**）进行更新
- 预分频器的缓冲区中将重新装载预装载值（**TIMx_PSC** 寄存器的内容）

以下各图以一些示例说明当 **TIMx_ARR=0x36** 时不同时钟频率下计数器的行为。

图 74. 计数器时序图, 1 分频内部时钟

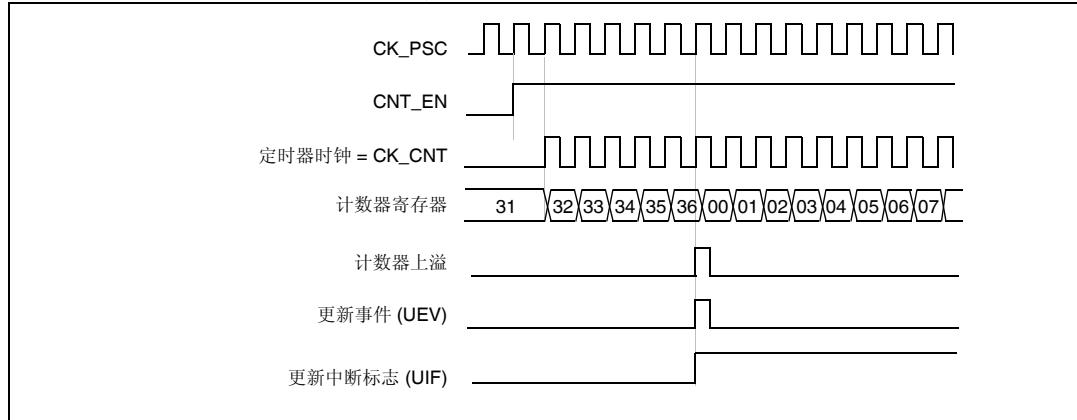


图 75. 计数器时序图, 2 分频内部时钟

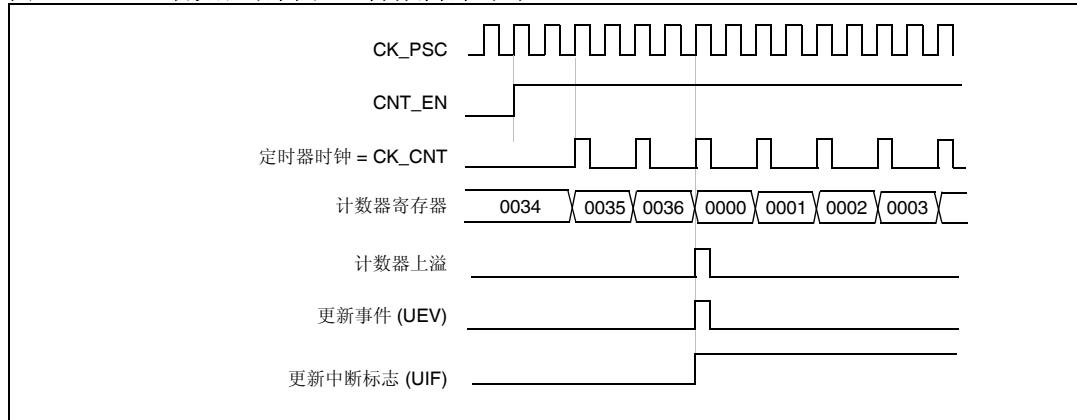


图 76. 计数器时序图，4 分频内部时钟

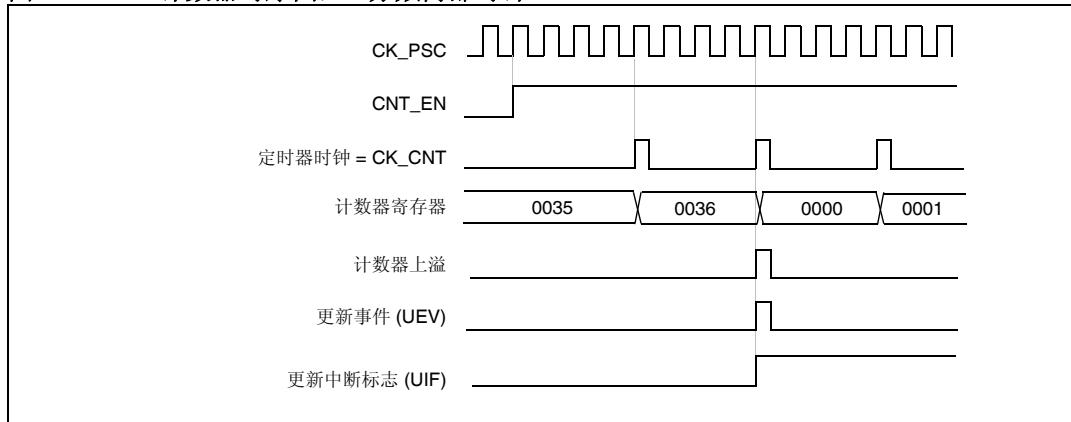


图 77. 计数器时序图，N 分频内部时钟

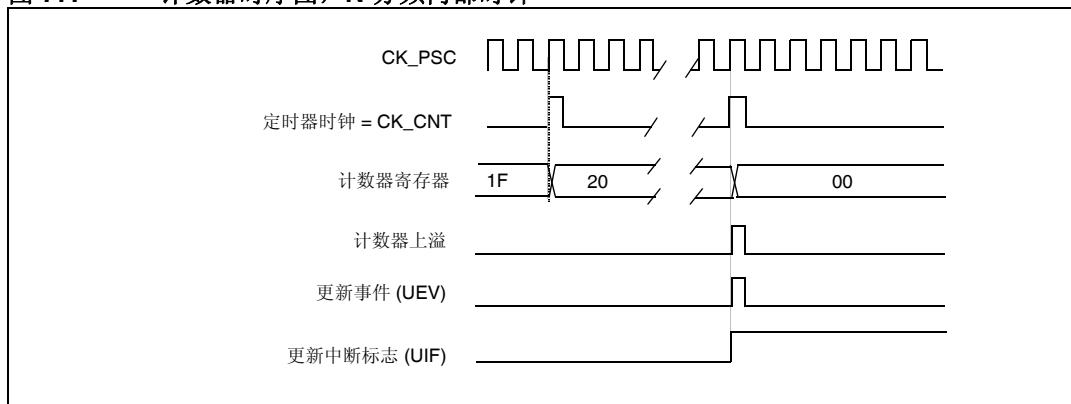


图 78. 计数器时序图，ARPE=0 时更新事件（TIMx_ARR 未预装载）

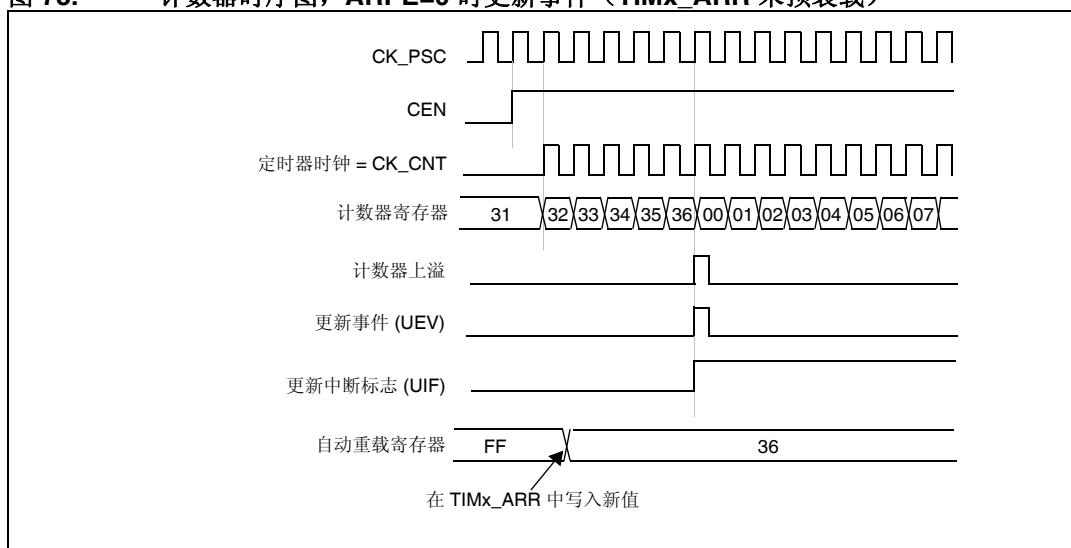
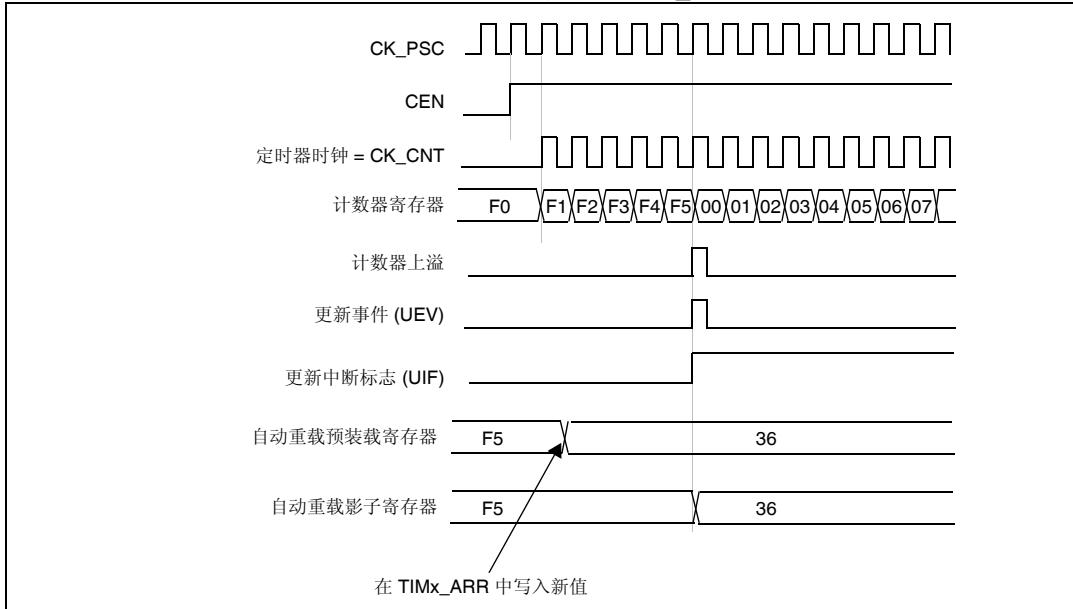


图 79. 计数器时序图, ARPE=1 时更新事件 (TIMx_ARR 预装载)



递减计数模式

在递减计数模式下，计数器从自动重载值 (TIMx_ARR 寄存器的内容) 开始递减计数到 0，然后重新从自动重载值开始计数并生成计数器下溢事件。

如果使用重复计数器，则当递减计数的重复次数达到重复计数器寄存器中编程的次数加一次 (TIMx_RCR+1) 后，将生成更新事件 (UEV)。否则，将在每次计数器下溢时产生更新事件。

将 TIMx_EGR 寄存器的 UG 位置 1 (通过软件或使用从模式控制器) 时，也将产生更新事件。

通过软件将 TIMx_CR1 寄存器中的 UDIS 位置 1 可禁止 UEV 更新事件。这可避免向预装载寄存器写入新值时更新影子寄存器。在 UDIS 位写入 0 之前不会产生任何更新事件。不过，计数器会重新从当前自动重载值开始计数，而预分频器计数器则重新从 0 开始计数 (但预分频比保持不变)。

此外，如果 TIMx_CR1 寄存器中的 URS 位 (更新请求选择) 已置 1，则将 UG 位置 1 会生成更新事件 UEV，但不会将 UIF 标志置 1 (因此，不会发送任何中断或 DMA 请求)。这样一来，如果在发生捕获事件时将计数器清零，将不会同时产生更新中断和捕获中断。

发生更新事件时，将更新所有寄存器且将更新标志 (TIMx_SR 寄存器中的 UIF 位) 置 1 (取决于 URS 位)：

- 重复计数器中将重新装载 TIMx_RCR 寄存器的内容
- 预分频器的缓冲区中将重新装载预装载值 (TIMx_PSC 寄存器的内容)
- 自动重载活动寄存器将以预装载值 (TIMx_ARR 寄存器的内容) 进行更新。注意，自动重载寄存器会在计数器重载之前得到更新，因此，下一个计数周期就是我们所希望的新周期长度

以下各图以一些示例说明当 $\text{TIMx_ARR}=0x36$ 时不同时钟频率下计数器的行为。

图 80. 计数器时序图, 1 分频内部时钟

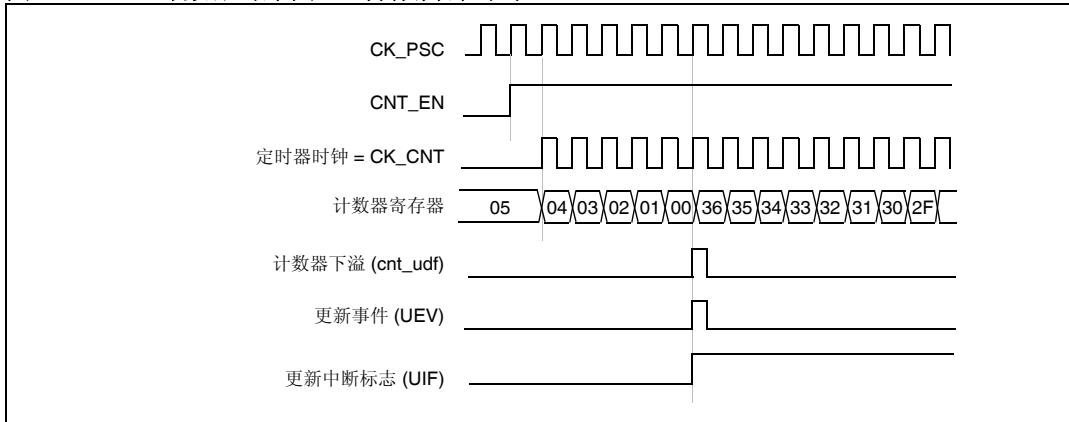


图 81. 计数器时序图, 2 分频内部时钟

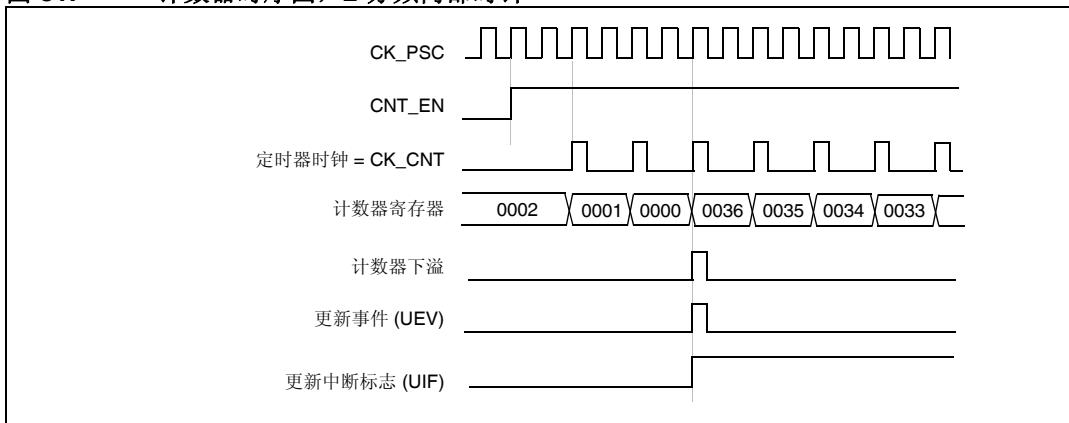


图 82. 计数器时序图, 4 分频内部时钟

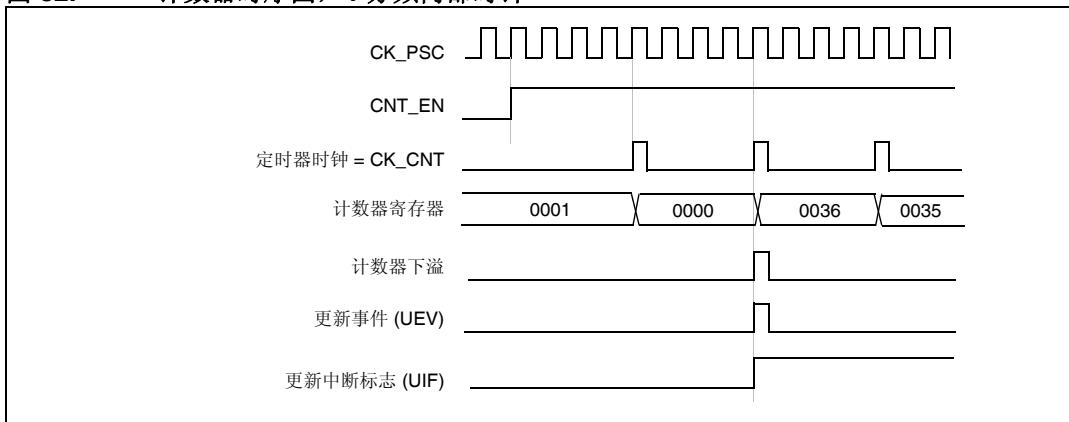


图 83. 计数器时序图，N 分频内部时钟

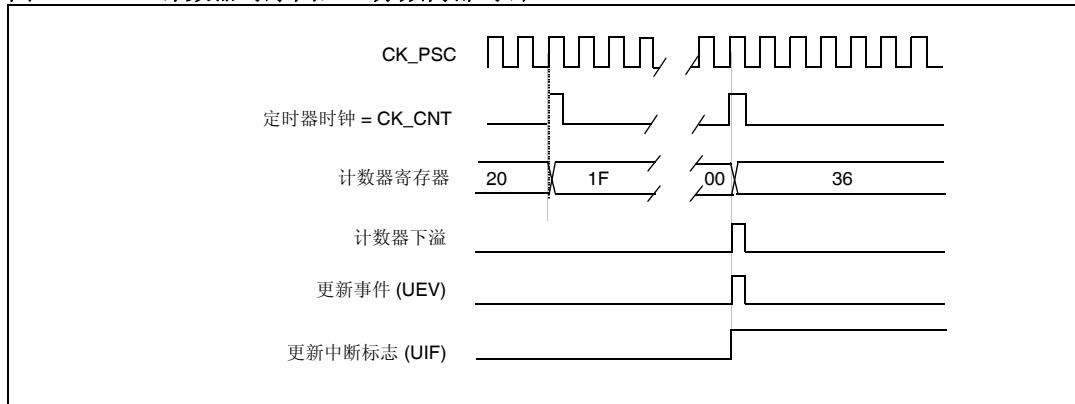
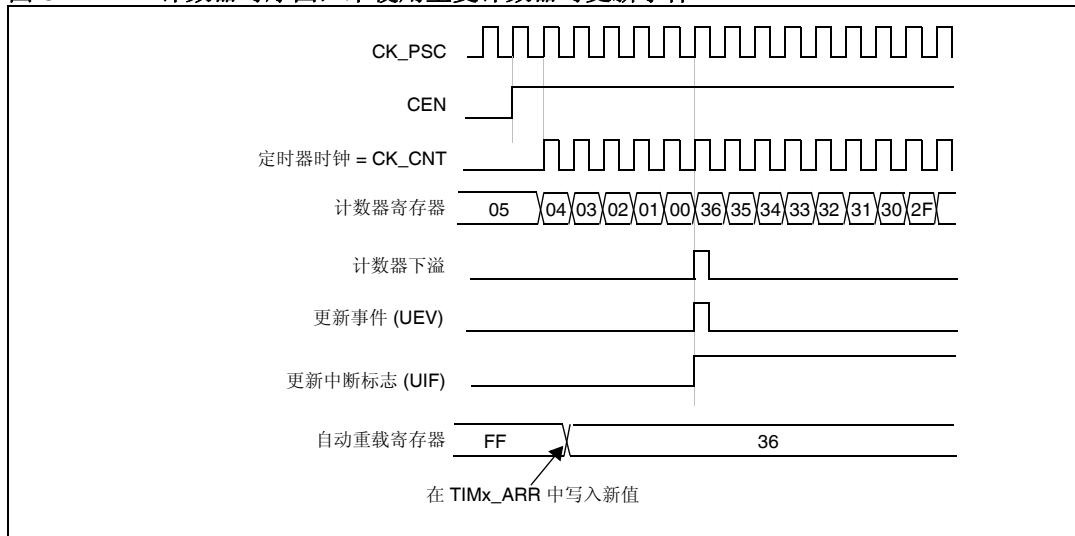


图 84. 计数器时序图，未使用重复计数器时更新事件



中心对齐模式（递增/递减计数）

在中心对齐模式下，计数器从 0 开始计数到自动重载值（**TIMx_ARR** 寄存器的内容）-1，生成计数器上溢事件；然后从自动重载值开始向下计数到 1 并生成计数器下溢事件。之后从 0 开始重新计数。

当 **TIMx_CR1** 寄存器中的 CMS 位不为“00”时，中心对齐模式有效。将通道配置为输出模式时，其输出比较中断标志将在以下模式下置 1，即：计数器递减计数（中心对齐模式 1，CMS = “01”）、计数器递增计数（中心对齐模式 2，CMS = “10”）以及计数器递增/递减计数（中心对齐模式 3，CMS = “11”）。

在此模式下，**TIMx_CR1** 寄存器的 DIR 方向位不可写入值，而是由硬件更新并指示当前计数器方向。

每次发生计数器上溢和下溢时都会生成更新事件，或将 **TIMx_EGR** 寄存器中的 UG 位置 1（通过软件或使用从模式控制器）也可以生成更新事件。这种情况下，计数器以及预分频器计数器将重新从 0 开始计数。

通过软件将 **TIMx_CR1** 寄存器中的 UDIS 位置 1 可禁止 UEV 更新事件。这可避免向预装载寄存器写入新值时更新影子寄存器。在 UDIS 位写入 0 之前不会产生任何更新事件。不过，计数器仍会根据当前自动重载值进行递增和递减计数。

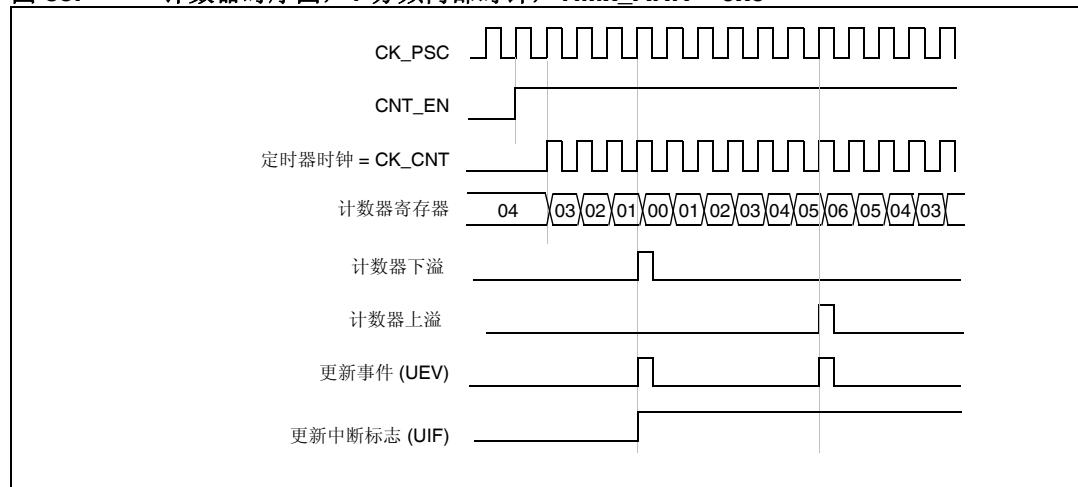
此外，如果 **TIMx_CR1** 寄存器中的 **URS** 位（更新请求选择）已置 1，则将 **UG** 位置 1 会生成 **UEV** 更新事件，但不会将 **UIF** 标志置 1（因此，不会发送任何中断或 DMA 请求）。这样一来，如果在发生捕获事件时将计数器清零，将不会同时产生更新中断和捕获中断。

发生更新事件时，将更新所有寄存器且将更新标志（**TIMx_SR** 寄存器中的 **UIF** 位）置 1（取决于 **URS** 位）：

- 重复计数器中将重新装载 **TIMx_RCR** 寄存器的内容
- 预分频器的缓冲区中将重新装载预装载值（**TIMx_PSC** 寄存器的内容）
- 自动重载活动寄存器将以预装载值（**TIMx_ARR** 寄存器的内容）进行更新。注意，如果更新操作是由计数器上溢触发的，则自动重载寄存器在重载计数器之前更新，因此，下一个计数周期就是我们所希望的新的周期长度（计数器被重载新的值）。

以下各图以一些示例说明不同时钟频率下计数器的行为。

图 85. 计数器时序图，1 分频内部时钟，TIMx_ARR = 0x6****



1. 此处使用中心对齐模式 1（有关详细信息，请参见第 365 页的第 14.4 节：**TIM1 和 TIM8 寄存器**）。

图 86. 计数器时序图，2 分频内部时钟

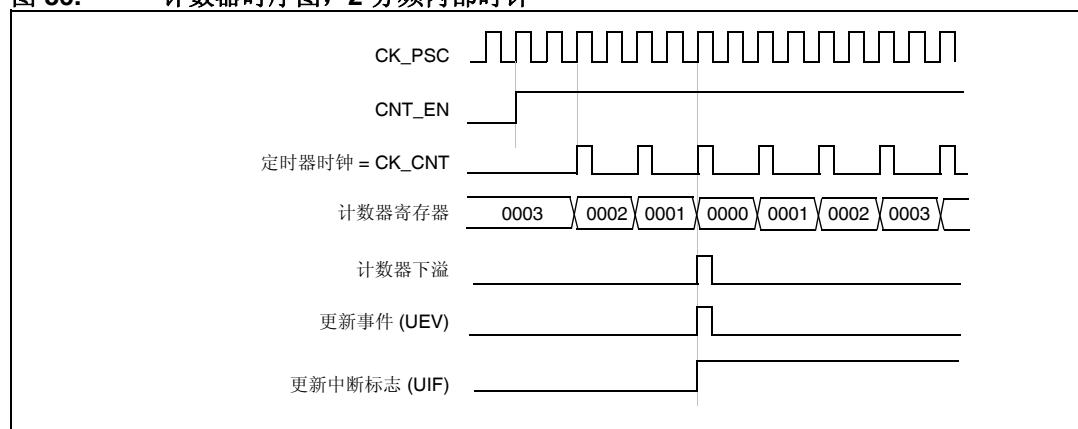
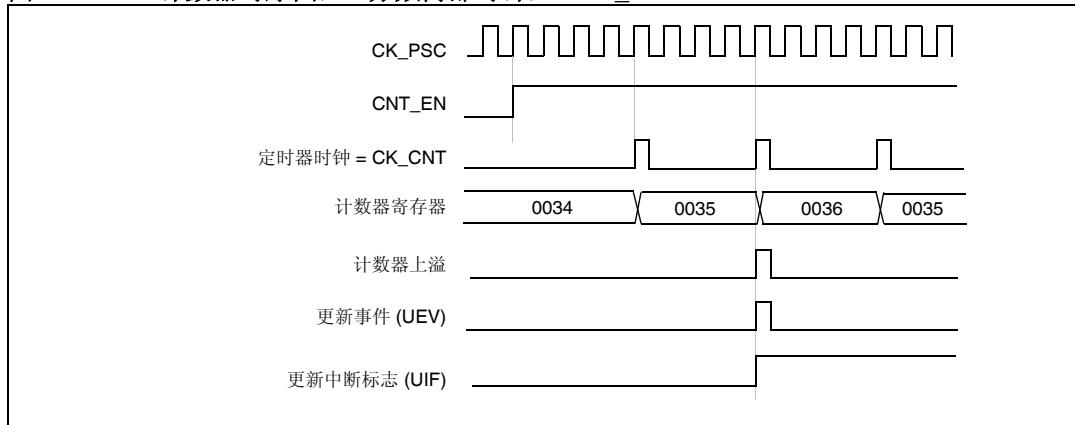


图 87. 计数器时序图，4 分频内部时钟， $\text{TIMx_ARR}=0x36$ 

1. 中心对齐模式 2 或模式 3 与上溢 UIF 结合使用。

图 88. 计数器时序图，N 分频内部时钟

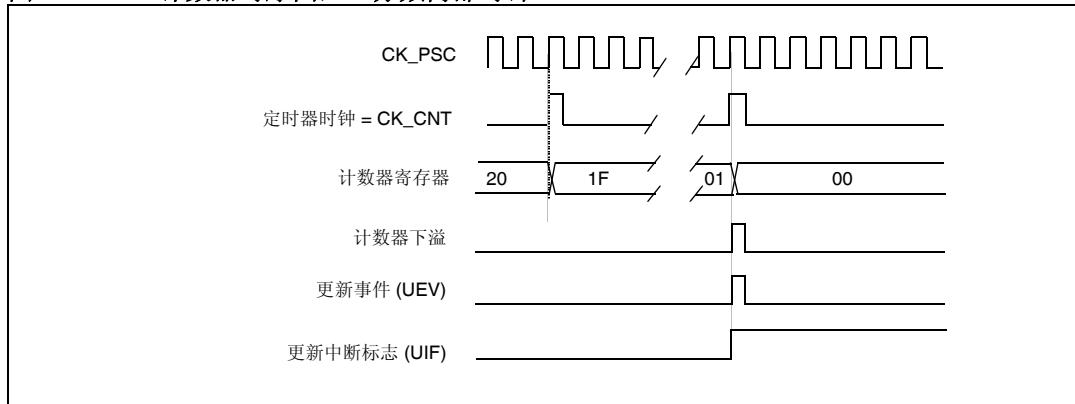


图 89. 计数器时序图，ARPE=1 时的更新事件（计数器下溢）

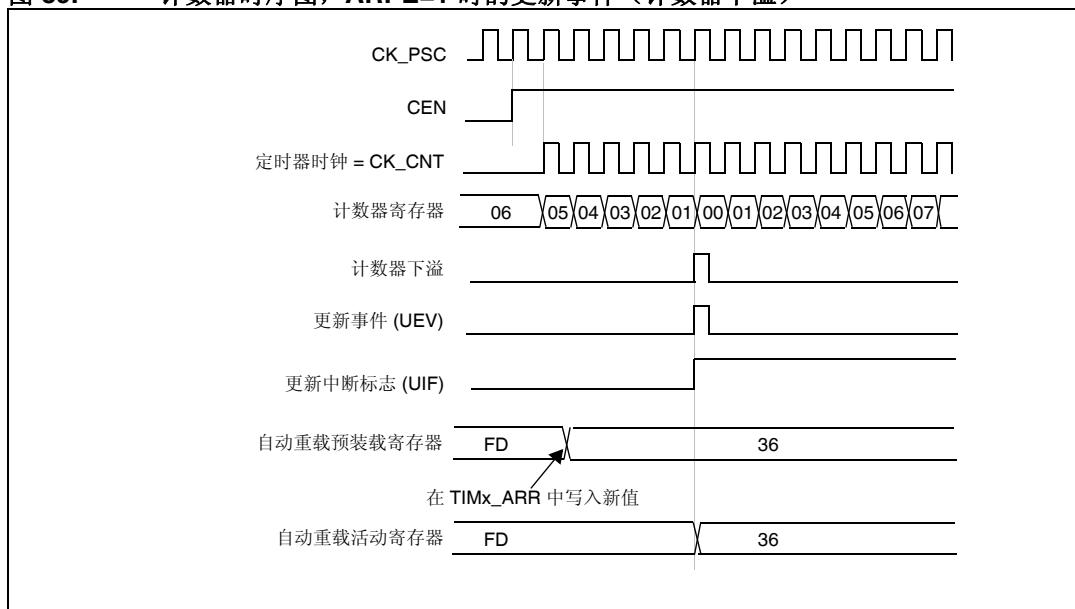
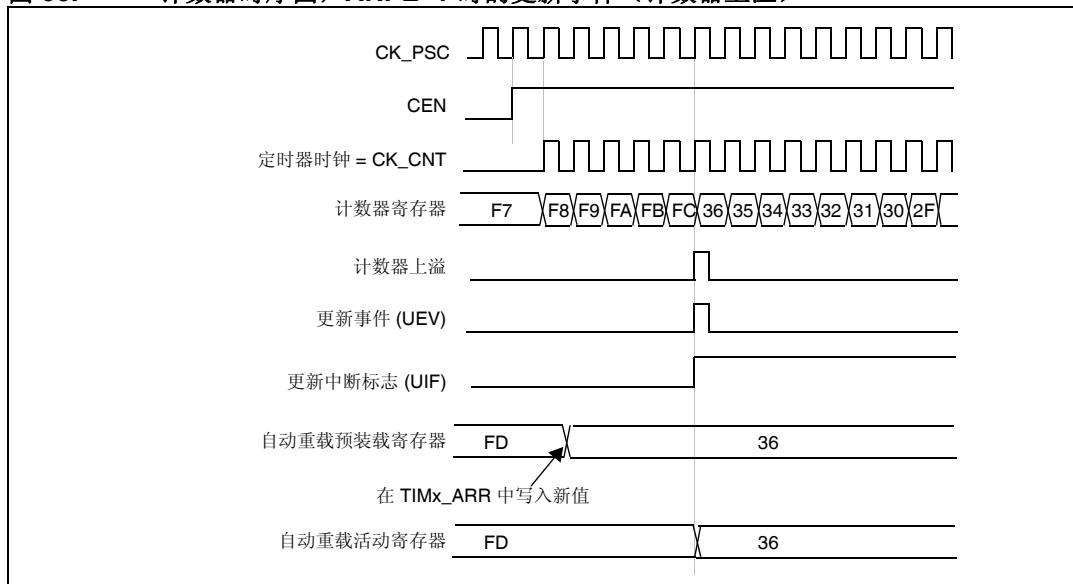


图 90. 计数器时序图, ARPE=1 时的更新事件 (计数器上溢)



14.3.3 重复计数器

[第 14.3.1 节: 时基单元](#)介绍如何因计数器上溢/下溢而生成更新事件 (UEV)。实际上，只有当重复计数器达到零时，才会生成更新事件。这在生成 PWM 信号时很有用。

这意味着，每当发生 $N+1$ 个计数器上溢或下溢（其中， N 是 **TIMx_RCR** 重复计数器寄存器中的值），数据就将从预装载寄存器转移到影子寄存器 (**TIMx_ARR** 自动重载寄存器、**TIMx_PSC** 预分频器寄存器以及比较模式下的 **TIMx_CCRx** 捕获/比较寄存器）。

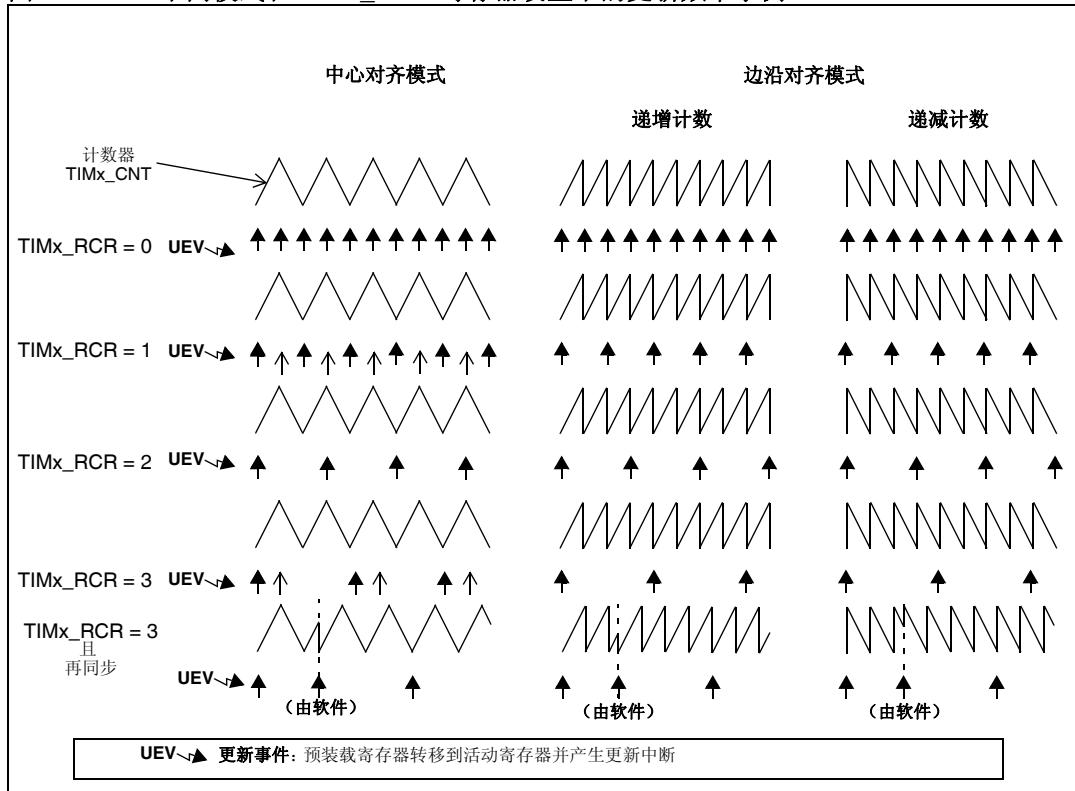
重复计数器在下列情况下递减：

- 递增计数模式下的每个计数器上溢。
- 递减计数模式下的每个计数器下溢。
- 中心对齐模式下每个计数器上溢和计数器下溢。尽管这使得最大重复次数不超过 128 个 PWM 周期，但在每个 PWM 周期内可更新占空比两次。当在中心对齐模式下，每个 PWM 周期仅刷新一次比较寄存器时，由于模式的对称性，最大分辨率为 $2 \times T_{ck}$ 。

重复计数器是自动重载类型；其重复率为 **TIMx_RCR** 寄存器所定义的值（请参见 [图 91](#)）。当更新事件由软件（通过将 **TIMx_EGR** 寄存器的 **UG** 位置 1）或硬件（通过从模式控制器）生成时，无论重复计数器的值为多少，更新事件都将立即发生，并且在重复计数器中重新装载 **TIMx_RCR** 寄存器的内容。

在中心对齐模式下，如果 **RCR** 值为奇数，更新事件将在上溢或下溢时发生，这取决于何时写入 **RCR** 寄存器以及何时启动计数器。如果在启动计数器前写入 **RCR**，则 **UEV** 在上溢时发生。如果在启动计数器后写入 **RCR**，则 **UEV** 在下溢时发生。例如，如果 **RCR** = 3，**UEV** 将在每个周期的第四个上溢或下溢事件时生成（取决于何时写入 **RCR**）。

图 91. 不同模式和 TIMx_RCR 寄存器设置下的更新频率示例



14.3.4 时钟选择

计数器时钟可由下列时钟源提供:

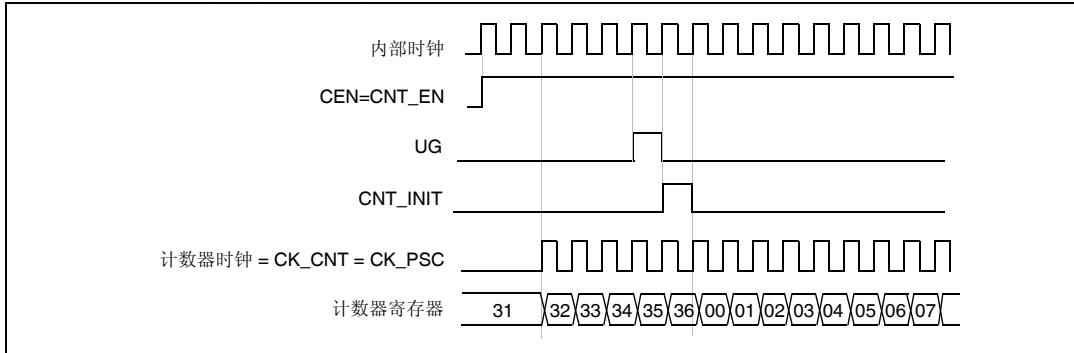
- 内部时钟 (CK_INT)
- 外部时钟模式 1: 外部输入引脚
- 外部时钟模式 2: 外部触发输入 ETR
- 外部触发输入 (ITRx): 使用一个定时器作为另一定时器的预分频器, 例如, 可将定时器 1 配置为定时器 2 的预分频器。更多详细信息, 请参见[将一个定时器用作另一个定时器的预分频器](#)。

内部时钟源 (CK_INT)

如果禁止从模式控制器 (SMS=000), 则 CEN 位、DIR 位 (TIMx_CR1 寄存器中) 和 UG 位 (TIMx_EGR 寄存器中) 为实际控制位, 并且只能通过软件进行更改 (UG 除外, 仍保持自动清零)。当对 CEN 位写入 1 时, 预分频器的时钟就由内部时钟 CK_INT 提供。

[图 92](#) 显示了正常模式下控制电路与递增计数器的行为 (没有预分频的情况下)。

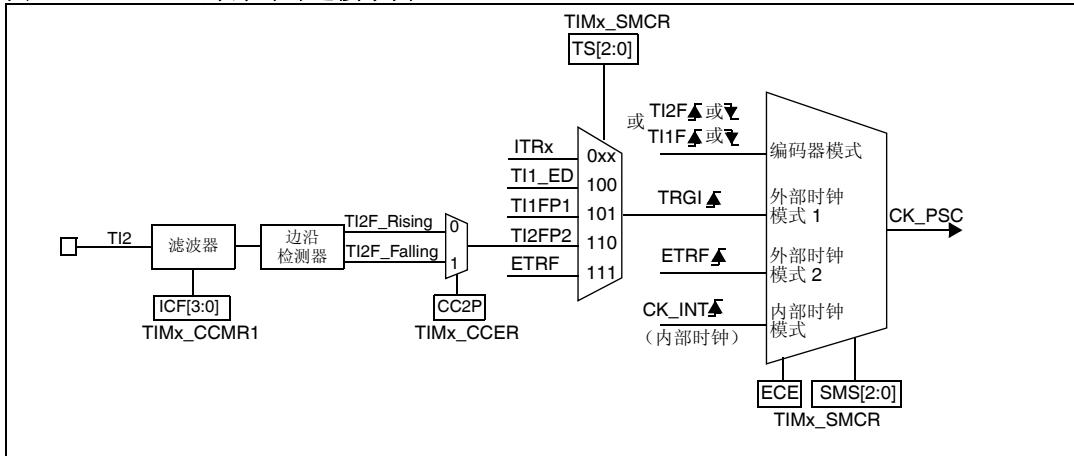
图 92. 正常模式下的控制电路, 1 分频内部时钟



外部时钟源模式 1

当 TIMx_SMCR 寄存器中的 $\text{SMS}=111$ 时, 可选择此模式。计数器可在选定的输入信号上出现上升沿或下降沿时计数。

图 93. TI2 外部时钟连接示例



例如, 要使递增计数器在 TI2 输入出现上升沿时计数, 请执行以下步骤:

1. 通过在 TIMx_CCMR1 寄存器中写入 $\text{CC2S} = "01"$ 来配置通道 2, 使其能够检测 TI2 输入的上升沿。
2. 通过在 TIMx_CCMR1 寄存器中写入 $\text{IC2F}[3:0]$ 位来配置输入滤波时间（如果不需要任何滤波, 请保持 $\text{IC2F}=0000$ ）。
3. 通过在 TIMx_CCER 寄存器中写入 $\text{CC2P}=0$ 和 $\text{CC2NP}=0$ 来选择上升沿极性。
4. 通过在 TIMx_SMCR 寄存器中写入 $\text{SMS}=111$, 使定时器在外部时钟模式 1 下工作。
5. 通过在 TIMx_SMCR 寄存器中写入 $\text{TS}=110$ 来选择 TI2 作为触发输入源。
6. 通过在 TIMx_CR1 寄存器中写入 $\text{CEN}=1$ 来使能计数器。

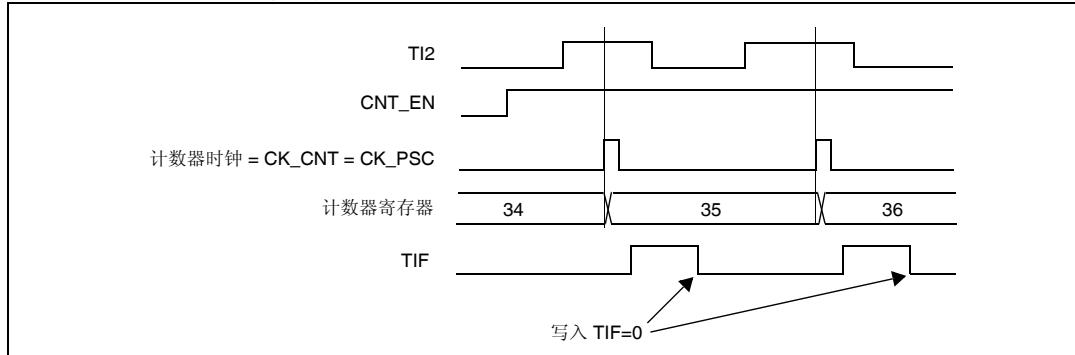
注意:

由于捕获预分频器不用于触发操作, 因此无需对其进行配置。

当 TI2 出现上升沿时, 计数器便会计数一次并且 TIF 标志置 1。

TI2 的上升沿与实际计数器时钟之间的延迟是由于 TI2 输入的重新同步电路引起的。

图 94. 外部时钟模式 1 下的控制电路



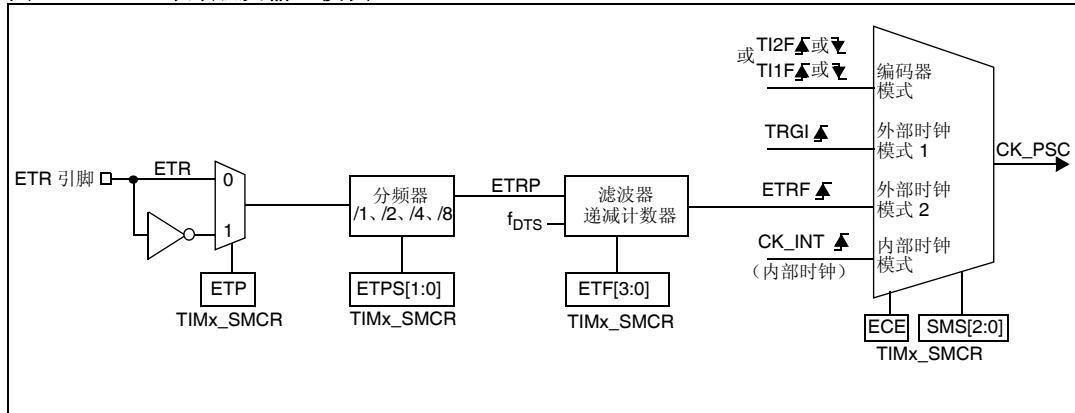
外部时钟源模式 2

通过在 **TIMx_SMCR** 寄存器中写入 **ECE=1** 可选择此模式。

计数器可在外部触发输入 **ETR** 出现上升沿或下降沿时计数。

[图 95](#) 简要介绍了外部触发输入模块。

图 95. 外部触发输入模块



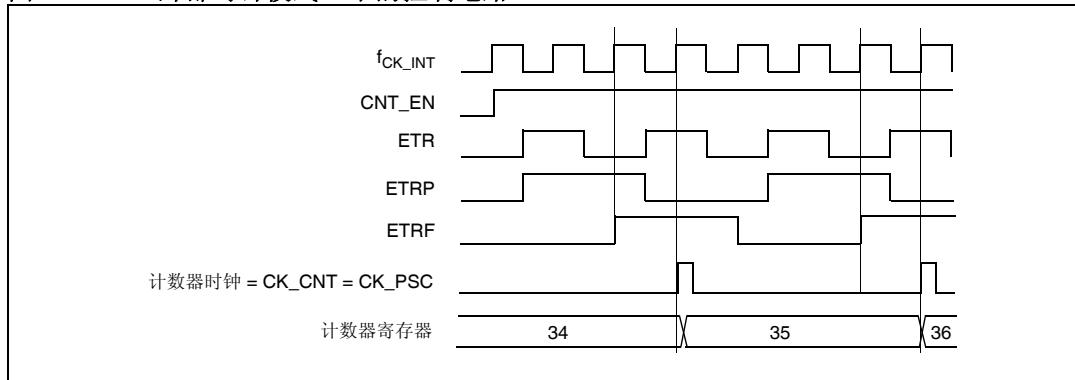
例如，要使递增计数器在 **ETR** 每出现 2 个上升沿时计数，请执行以下步骤：

1. 由于此例中不需滤波器，因此在 **TIMx_SMCR** 寄存器中写入 **ETF[3:0]=0000**。
2. 通过在 **TIMx_SMCR** 寄存器中写入 **ETPS[1:0]=01** 来设置预分频器。
3. 通过在 **TIMx_SMCR** 寄存器中写入 **ETP=0** 来选择 **ETR** 引脚的上升沿检测。
4. 通过在 **TIMx_SMCR** 寄存器中写入 **ECE=1** 来使能外部时钟模式 2。
5. 通过在 **TIMx_CR1** 寄存器中写入 **CEN=1** 来使能计数器。

ETR 每出现 2 个上升沿，计数器计数一次。

ETR 的上升沿与实际计数器时钟之间的延迟是由于 **ETRP** 信号的重新同步电路引起的。

图 96. 外部时钟模式 2 下的控制电路



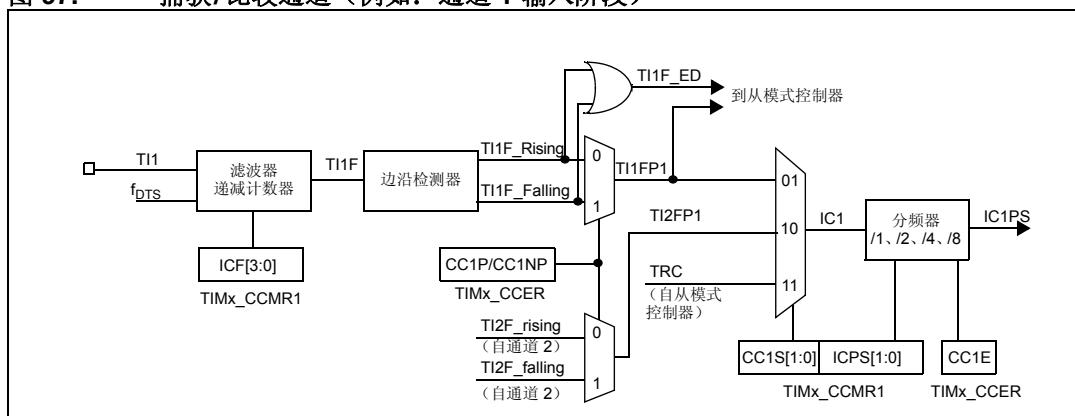
14.3.5 捕获/比较通道

每个捕获/比较通道均围绕一个捕获/比较寄存器（包括一个影子寄存器）、一个捕获输入阶段（数字滤波、多路复用和预分频器）和一个输出阶段（比较器和输出控制）构建而成。

[图 97 到 图 100](#) 概括介绍了一个捕获/比较通道。

输入阶段对相应的 TIx 输入进行采样，生成一个滤波后的信号 TIx_F 。然后，带有极性选择功能的边沿检测器生成一个信号 ($TIxFPx$)，该信号可用作从模式控制器的触发输入，也可用作捕获命令。该信号先进行预分频 (ICxPS)，而后再进入捕获寄存器。

图 97. 捕获/比较通道（例如：通道 1 输入阶段）



输出阶段生成一个中间波形作为基准：OCxRef（高电平有效）。链的末端决定最终输出信号的极性。

图 98. 捕获/比较通道 1 主电路

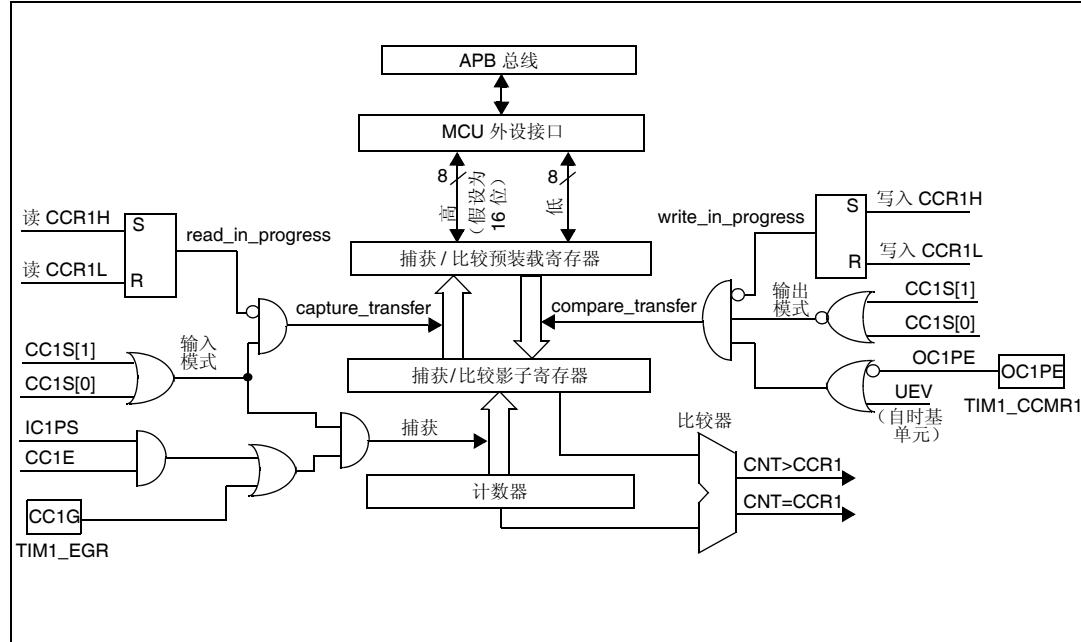


图 99. 捕获/比较通道的输出阶段（通道 1 到 3）

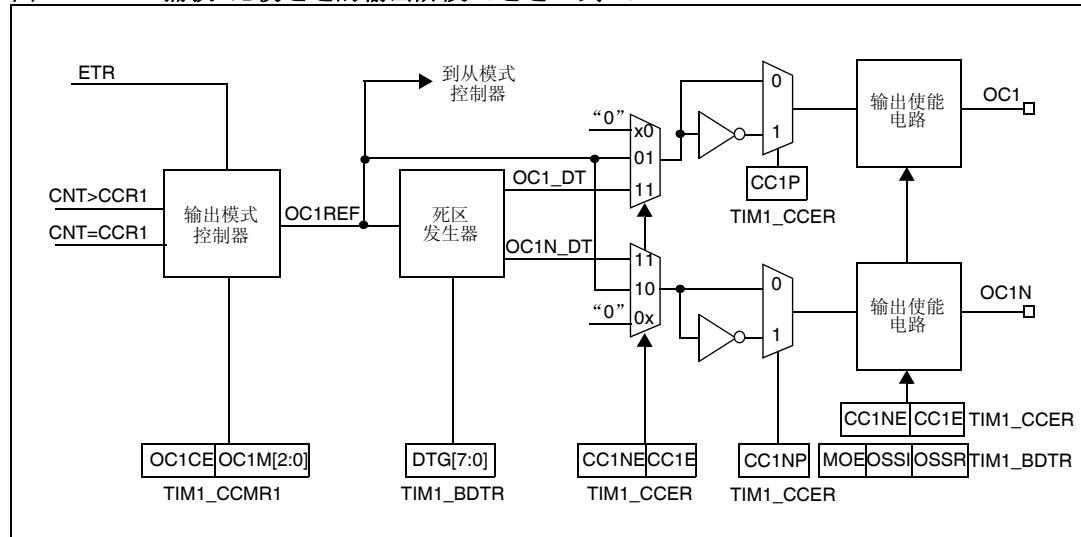
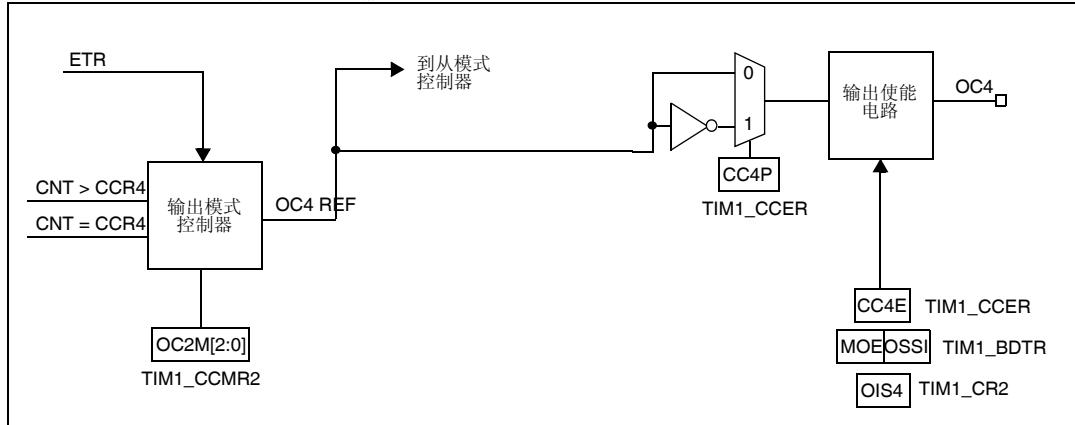


图 100. 捕获/比较通道的输出阶段 (通道 4)



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。始终可通过读写操作访问预装载寄存器。

在捕获模式下，捕获实际发生在影子寄存器中，然后将影子寄存器的内容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将被复制到影子寄存器中，然后将影子寄存器的内容与计数器进行比较。

14.3.6 输入捕获模式

在输入捕获模式下，当相应的 IC_x 信号检测到跳变沿后，将使用捕获/比较寄存器 (TIM_x_CCR_x) 来锁存计数器的值。发生捕获事件时，会将相应的 CC_xIF 标志 (TIM_x_SR 寄存器) 置 1，并可发送中断或 DMA 请求（如果已使能）。如果发生捕获事件时 CC_xIF 标志已处于高位，则会将重复捕获标志 CC_xOF (TIM_x_SR 寄存器) 置 1。可通过软件向 CC_xIF 写入 0 来给 CC_xIF 清零，或读取存储在 TIM_x_CCR_x 寄存器中的已捕获数据。向 CC_xOF 写入“0”后会将其清零。

以下示例说明了如何在 TI1 输入出现上升沿时将计数器的值捕获到 TIM_x_CCR1 中。具体操作步骤如下：

- 选择有效输入：TIM_x_CCR1 必须连接到 TI1 输入，因此向 TIM_x_CCMR1 寄存器中的 CC1S 位写入 01。只要 CC1S 不等于 00，就会将通道配置为输入模式，并且 TIM_x_CCR1 寄存器将处于只读状态。
- 根据连接到定时器的信号，对所需的输入滤波时间进行编程（如果输入为 TI_x 输入，则对 TIM_x_CCMRx 寄存器中的 ICxF 位进行编程）。假设信号边沿变化时，输入信号最多在 5 个内部时钟周期内发生抖动。因此，我们必须将滤波时间设置为大于 5 个内部时钟周期。在检测到 8 个具有新电平的连续采样（以 f_{DTS} 频率采样）后，可以确认 TI1 上的跳变沿。然后向 TIM_x_CCMR1 寄存器中的 IC1F 位写入 0011。
- 通过在 TIM_x_CCER 寄存器中将 CC1P 位和 CC1NP 位写入 0，选择 TI1 上的有效转换边沿（本例中为上升沿）。
- 对输入预分频器进行编程。在本例中，我们希望每次有效转换时都执行捕获操作，因此需要禁止预分频器（向 TIM_x_CCMR1 寄存器中的 IC1PS 位写入“00”）。
- 通过将 TIM_x_CCER 寄存器中的 CC1E 位置 1，允许将计数器的值捕获到捕获寄存器中。
- 如果需要，可通过将 TIM_x_DIER 寄存器中的 CC1IE 位置 1 来使能相关中断请求，并且/或者通过将该寄存器中的 CC1DE 位置 1 来使能 DMA 请求。

发生输入捕获时：

- 发生有效跳变沿时，**TIMx_CCR1** 寄存器会获取计数器的值。
- 将 **CC1IF** 标志置 1（中断标志）。如果至少发生了两次连续捕获，但 **CC1OF** 标志未被清零，这样 **CC1OF** 捕获溢出标志会被置 1。
- 根据 **CC1IE** 位生成中断。
- 根据 **CC1DE** 位生成 DMA 请求。

要处理重复捕获，建议在读出捕获溢出标志之前读取数据。这样可避免丢失在读取捕获溢出标志之后与读取数据之前可能出现的重复捕获信息。

注意：通过软件将 **TIMx_EGR** 寄存器中的相应 **CCxG** 位置 1 可生成 IC 中断和/或 DMA 请求。

14.3.7 PWM 输入模式

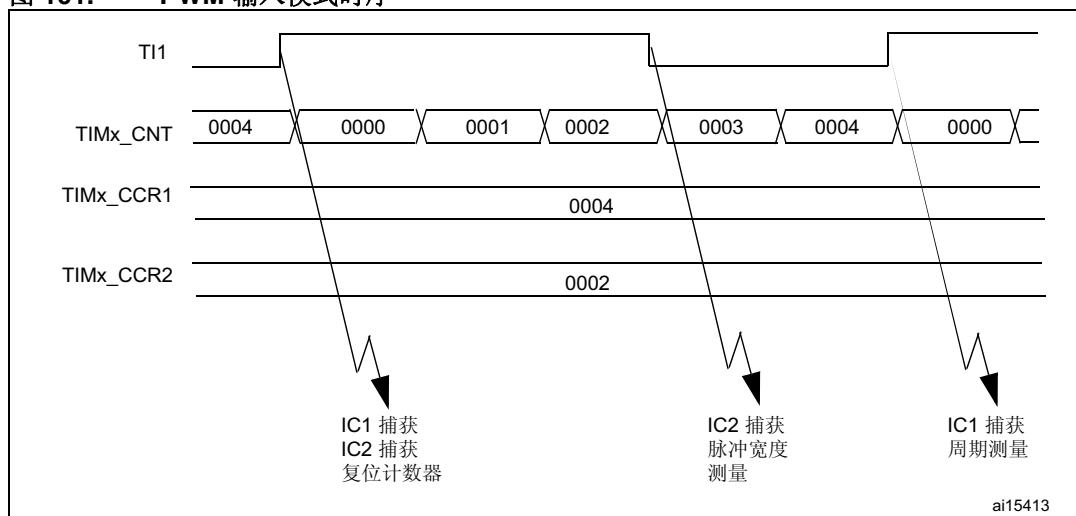
此模式是输入捕获模式的一个特例。其实现步骤与输入捕获模式基本相同，仅存在以下不同之处：

- 两个 **ICx** 信号被映射至同一个 **TIx** 输入。
- 这两个 **ICx** 信号在边沿处有效，但极性相反。
- 选择两个 **TIxFP** 信号之一作为触发输入，并将从模式控制器配置为复位模式。

例如，可通过以下步骤对应用于 **TI1** 的 PWM 的周期（位于 **TIMx_CCR1** 寄存器中）和占空比（位于 **TIMx_CCR2** 寄存器中）进行测量（取决于 **CK_INT** 频率和预分频器的值）：

- 选择 **TIMx_CCR1** 的有效输入：向 **TIMx_CCMR1** 寄存器中的 **CC1S** 位写入 01（选择 **TI1**）。
- 选择 **TI1FP1** 的有效极性（用于在 **TIMx_CCR1** 中捕获和计数器清零）：向 **CC1P** 位和 **CC1NP** 位写入“0”（上升沿有效）。
- 选择 **TIMx_CCR2** 的有效输入：向 **TIMx_CCMR1** 寄存器中的 **CC2S** 写入 10（选择 **TI1**）。
- 选择 **TI1FP2** 的有效极性（用于在 **TIMx_CCR2** 中捕获）：向 **CC2P** 位和 **CC2NP** 位写入“1”（下降沿有效）。
- 选择有效触发输入：向 **TIMx_SMCR** 寄存器中的 **TS** 位写入 101（选择 **TI1FP1**）。
- 将从模式控制器配置为复位模式：向 **TIMx_SMCR** 寄存器中的 **SMS** 位写入 100。
- 使能捕获：向 **TIMx_CCER** 寄存器中的 **CC1E** 位和 **CC2E** 位写入“1”。

图 101. PWM 输入模式时序



14.3.8 强制输出模式

在输出模式（ TIMx_CCMRx 寄存器中的 $\text{CCxS} = 00$ ）下，可直接由软件将每个输出比较信号（ OCxREF 和 OCx/OCxN ）强制设置为有效电平或无效电平，而无需考虑输出比较寄存器和计数器之间的任何比较结果。

要将输出比较信号（ OCXREF/OCx ）强制设置为有效电平，只需向相应 TIMx_CCMRx 寄存器中的 OCxM 位写入 101。 OCXREF 进而强制设置为高电平（ OCxREF 始终为高电平有效），同时 OCx 获取 CCxP 极性位的相反值。

例如： $\text{CCxP}=0$ (OCx 高电平有效) $\Rightarrow \text{OCx}$ 强制设置为高电平。

通过向 TIMx_CCMRx 寄存器中的 OCxM 位写入 100，可将 OCxREF 信号强制设置为低电平。

无论如何， TIMx_CCRx 影子寄存器与计数器之间的比较仍会执行，而且允许将标志置 1。因此可发送相应的中断和 DMA 请求。下面的输出比较模式一节对此进行了介绍。

14.3.9 输出比较模式

此功能用于控制输出波形，或指示已经过某一时间段。

当捕获/比较寄存器与计数器之间相匹配时，输出比较功能：

- 将为相应的输出引脚分配一个可编程值，该值由输出比较模式（ 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。匹配时，输出引脚既可保持其电平（ $\text{OCXM}=000$ ），也可设置为有效电平（ $\text{OCXM}=001$ ）、无效电平（ $\text{OCXM}=010$ ）或进行翻转（ $\text{OCXM}=011$ ）。
- 将中断状态寄存器中的标志置 1（ TIMx_SR 寄存器中的 CCxIF 位）。
- 如果相应中断使能位（ TIMx_DIER 寄存器中的 CCxIE 位）置 1，将生成中断。
- 如果相应 DMA 使能位（ TIMx_DIER 寄存器的 CCxDE 位， TIMx_CR2 寄存器的 CCDS 位，用来选择 DMA 请求）置 1，将发送 DMA 请求。

使用 TIMx_CCMRx 寄存器中的 OCxPE 位，可将 TIMx_CCRx 寄存器配置为带或不带预装载寄存器。

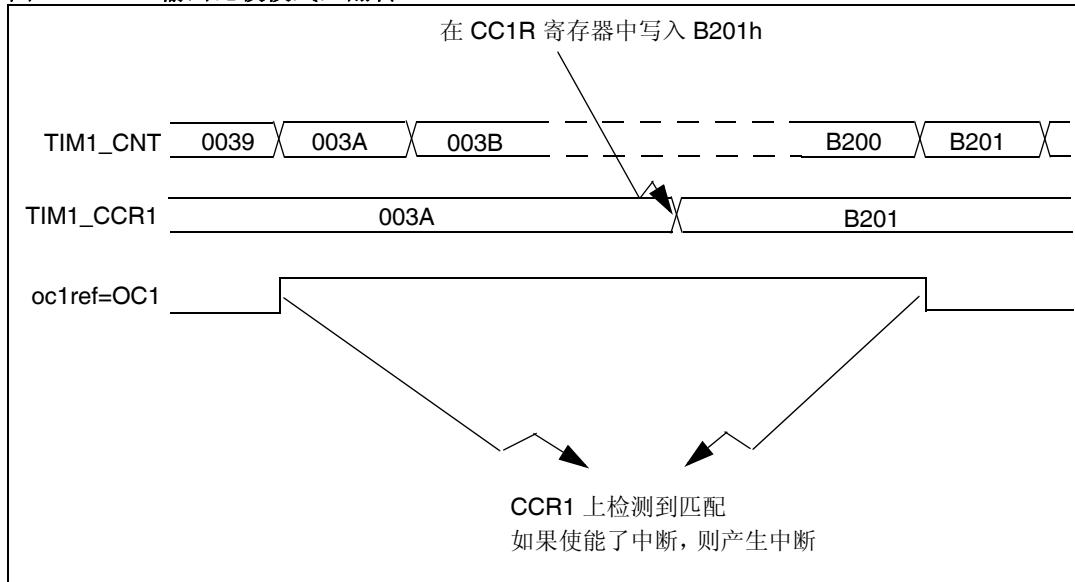
在输出比较模式下，更新事件 UEV 对 OCxREF 和 OCx 输出毫无影响。同步的精度可以达到计数器的一个计数周期。输出比较模式也可用于输出单脉冲（在单脉冲模式下）。

步骤：

1. 选择计数器时钟（内部、外部、预分频器）。
2. 在 TIMx_ARR 和 TIMx_CCRx 寄存器中写入所需数据。
3. 如果要生成中断请求，则需将 CCxIE 位置 1。
4. 选择输出模式。例如：
 - 当 CNT 与 CCRx 匹配时，写入 $\text{OCxM} = 011$ 以翻转 OCx 输出引脚
 - 写入 $\text{OCxPE} = 0$ 以禁止预装载寄存器
 - 写入 $\text{CCxP} = 0$ 以选择高电平有效极性
 - 写入 $\text{CCxE} = 1$ 以使能输出
5. 通过将 TIMx_CR1 寄存器中的 CEN 位置 1 来使能计数器。

可通过软件随时更新 TIMx_CCRx 寄存器以控制输出波形，前提是未使能预加载寄存器（ $\text{OCxPE} = "0"$ ，否则仅当发生下一个更新事件 UEV 时，才会更新 TIMx_CCRx 影子寄存器）。图 102 给出了一个示例。

图 102. 输出比较模式，翻转 OC1。



14.3.10 PWM 模式

脉冲宽度调制模式可以生成一个信号，该信号频率由 **TIMx_ARR** 寄存器值决定，其占空比则由 **TIMx_CCRx** 寄存器值决定。

通过向 **TIMx_CCMRx** 寄存器中的 **OCxM** 位写入 110 (PWM 模式 1) 或 111 (PWM 模式 2)，可以独立选择各通道（每个 **OCx** 输出对应一个 PWM）的 PWM 模式。必须通过将 **TIMx_CCMRx** 寄存器中的 **OCxPE** 位置 1 使能相应预装载寄存器，最后通过将 **TIMx_CR1** 寄存器中的 **ARPE** 位置 1 使能自动重载预装载寄存器（在递增计数或中心对齐模式下）。

由于只有在发生更新事件时预装载寄存器才会传送到影子寄存器，因此启动计数器之前，必须通过将 **TIMx_EGR** 寄存器中的 **UG** 位置 1 来初始化所有寄存器。

OCx 极性可使用 **TIMx_CCER** 寄存器的 **CCxP** 位来编程。既可以设为高电平有效，也可以设为低电平有效。通过 **CCxE**、**CCxNE**、**MOE**、**OSSI** 和 **OSSR** 位 (**TIMx_CCER** 和 **TIMx_BDTR** 寄存器) 的组合使能 **OCx** 输出。有关详细信息，请参见 **TIMx_CCER** 寄存器说明。

在 PWM 模式（1 或 2）下，**TIMx_CNT** 总是与 **TIMx_CCRx** 进行比较，以确定是 **TIMx_CCRx ≤ TIMx_CNT** 还是 **TIMx_CNT ≤ TIMx_CCRx**（取决于计数器计数方向）。

根据 **TIMx_CR1** 寄存器中的 **CMS** 位状态，定时器能够产生边沿对齐模式或中心对齐模式的 PWM 信号。

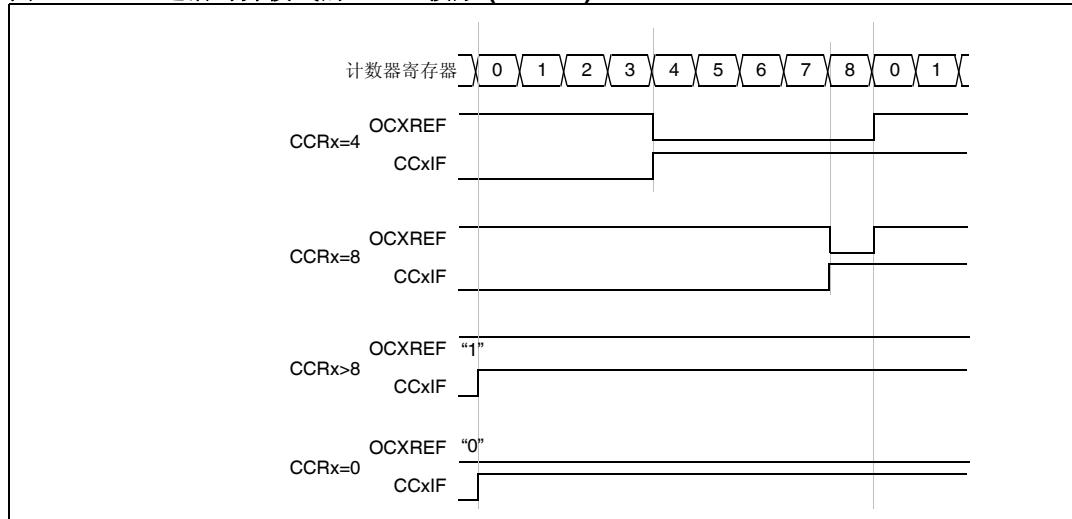
PWM 边沿对齐模式

- 递增计数配置

当 TIMx_CR1 寄存器中的 **DIR** 位为低时执行递增计数。请参见 [第 332 页的递增计数模式](#)一节。

以下以 PWM 模式 1 为例。只要 $\text{TIMx_CNT} < \text{TIMx_CCRx}$, PWM 参考信号 OCxREF 便为高电平, 否则为低电平。如果 TIMx_CCRx 中的比较值大于自动重载值 (TIMx_ARR 中), 则 OCxREF 保持为 “1”。如果比较值为 0, 则 OCxRef 保持为 “0”。[图 103](#) 举例介绍边沿对齐模式的一些 PWM 波形 ($\text{TIMx_ARR}=8$)。

图 103. 边沿对齐模式的 PWM 波形 (ARR=8)



- 递减计数配置

当 TIMx_CR1 寄存器中的 **DIR** 位为高时执行递减计数。请参见 [第 335 页的递减计数模式](#)一节。

在 PWM 模式 1 下, 只要 $\text{TIMx_CNT} > \text{TIMx_CCRx}$, 参考信号 OCxRef 即为低电平, 否则其为高电平。如果 TIMx_CCRx 中的比较值大于 TIMx_ARR 中的自动重载值, 则 OCxREF 保持为 “1”。此模式下不可能产生 0% 的 PWM 波形。

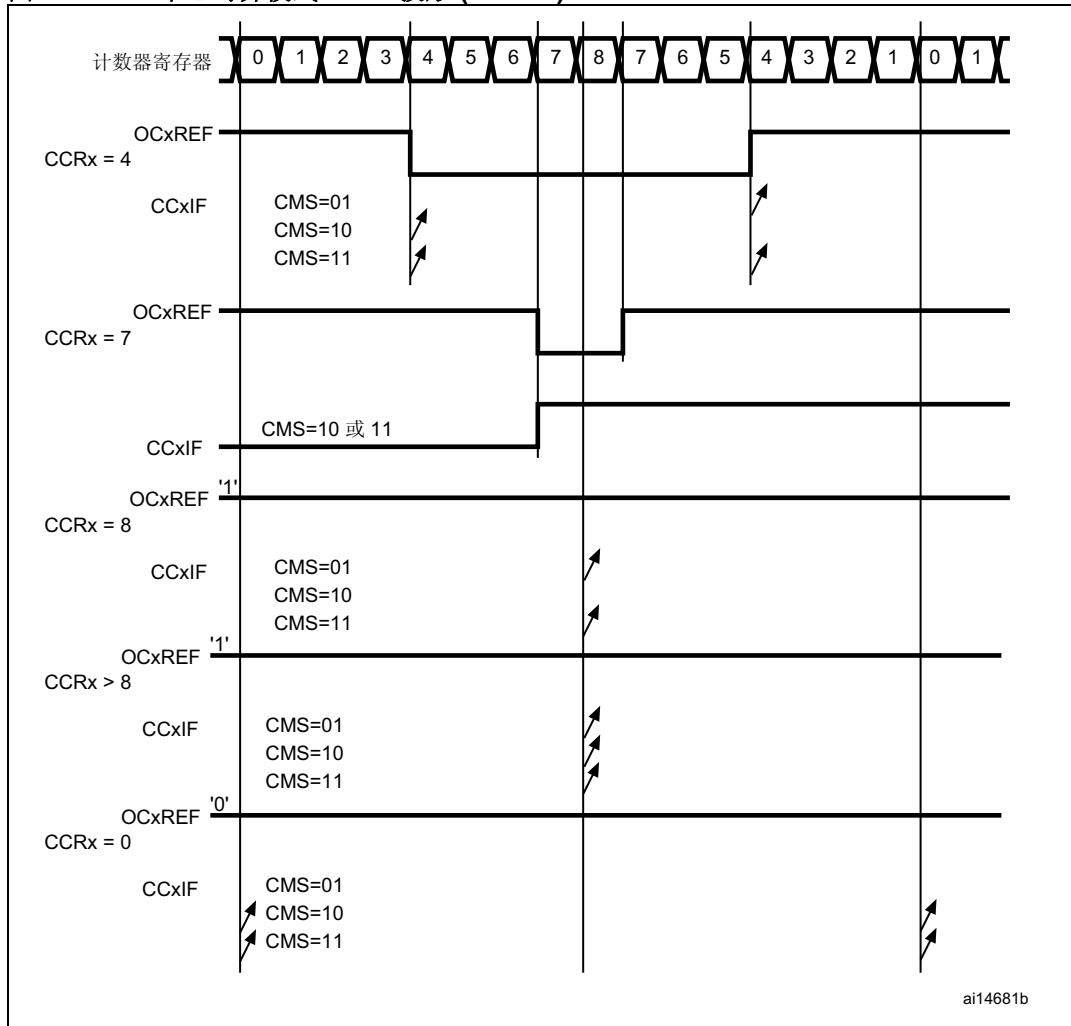
PWM 中心对齐模式

当 TIMx_CR1 寄存器中的 CMS 位不为 “00” (其余所有配置对 OCxRef/OCx 信号具有相同的作用), 中心对齐模式生效。根据 CMS 位的配置, 可以在计数器递增计数、递减计数或同时递增和递减计数时将比较标志置 1。 TIMx_CR1 寄存器中的方向位 (DIR) 由硬件更新, 不得通过软件更改。请参见 [第 337 页的中心对齐模式 \(递增/递减计数\)](#)。

[图 104](#) 显示了中心对齐模式的 PWM 波形, 在此例中:

- $\text{TIMx_ARR}=8$,
- PWM 模式为 PWM 模式 1,
- 在根据 TIMx_CR1 寄存器中 CMS=01 而选择的中心对齐模式 1 下, 当计数器递减计数时, 比较标志置 1。

图 104. 中心对齐模式 PWM 波形 (ARR=8)



中心对齐模式使用建议：

- 启动中心对齐模式时将使用当前的递增/递减计数配置。这意味着计数器将根据写入 **TIMx_CR1** 寄存器中 **DIR** 位的值进行递增或递减计数。此外，不得同时通过软件修改 **DIR** 和 **CMS** 位。
- 不建议在运行中心对齐模式时对计数器执行写操作，否则将发生意想不到的结果。尤其是：
 - 如果写入计数器中的值大于自动重载值 (**TIMx_CNT>TIMx_ARR**)，计数方向不会更新。例如，如果计数器之前递增计数，则继续递增计数。
 - 如果向计数器写入 0 或 **TIMx_ARR** 的值，计数方向会更新，但不生成更新事件 UEV。
- 使用中心对齐模式最为保险的方法是：在启动计数器前通过软件生成更新（将 **TIMx_EGR** 寄存器中的 **UG** 位置 1），并且不要在计数器运行过程中对其执行写操作。

14.3.11 互补输出和死区插入

高级控制定时器（TIM1 和 TIM8）可以输出两路互补信号，并管理输出的关断与接通瞬间。

这段时间通常称为死区，用户必须根据与输出相连接的器件及其特性（电平转换器的固有延迟、开关器件产生的延迟...）来调整死区时间

每路输出可以独立选择输出极性（主输出 OC_x 或互补输出 OC_{xN}）。可通过对 TIMx_CCER 寄存器中的 CC_{xP} 和 CC_{xNP} 位执行写操作来完成极性选择。

互补信号 OC_x 和 OC_{xN} 通过以下多个控制位的组合进行激活：TIMx_CCER 寄存器中的 CC_{xE} 和 CC_{xNE} 位以及 TIMx_BDTR 和 TIMx_CR2 寄存器中的 MOE、OIS_x、OIS_{xN}、OSSI 和 OSSR 位。更多详细信息，请参见第 382 页的表 73：具有断路功能的互补通道 OC_x 和 OC_{xN} 的输出控制位。应当注意，切换至 IDLE (MOE 下降到 0) 的时刻，死区仍然有效。

CC_{xE} 和 CC_{xNE} 位同时置 1 并且 MOE 位置 1 (如果存在断路) 时，将使能死区插入。TIMx_BDTR 寄存器中的 DTG[7:0] 位用于控制所有通道的死区生成。将基于参考波形 OC_{xREF} 生成 2 个输出 OC_x 和 OC_{xN}。如果 OC_x 和 OC_{xN} 为高电平有效：

- 输出信号 OC_x 与参考信号相同，只是其上升沿相对参考上升沿存在延迟。
- 输出信号 OC_{xN} 与参考信号相反，并且其上升沿相对参考下降沿存在延迟。

如果延迟时间大于有效输出 (OC_x 或 OC_{xN}) 的宽度，则不会产生相应的脉冲。

下图所示为死区发生器的输出信号与参考信号 OC_{xREF} 之间的关系。（在这些示例中，假定 CC_{xP}=0、CC_{xNP}=0、MOE=1、CC_{xE}=1 并且 CC_{xNE}=1）

图 105. 带死区插入的互补输出。

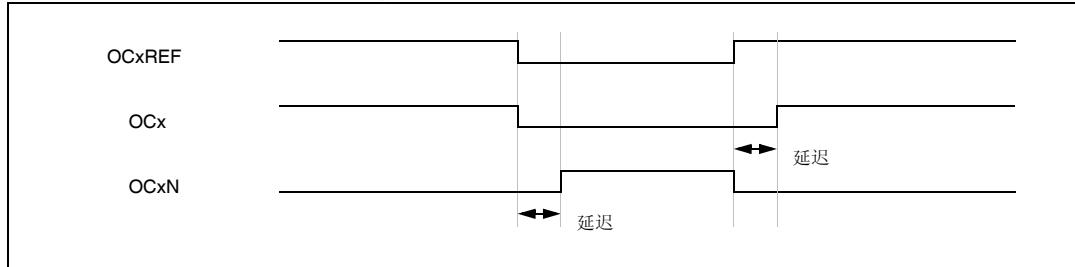


图 106. 延迟时间大于负脉冲宽度的死区波形

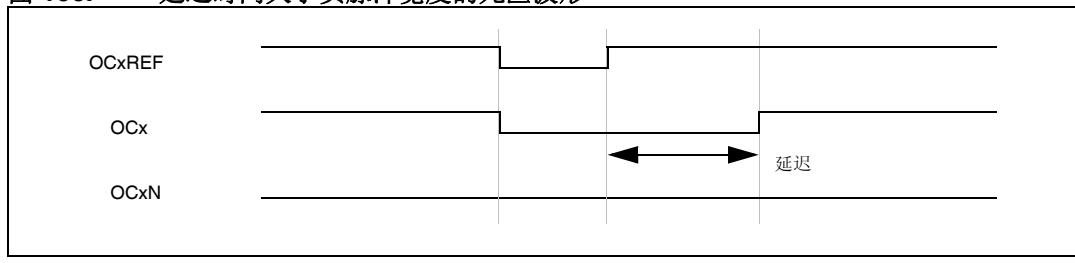
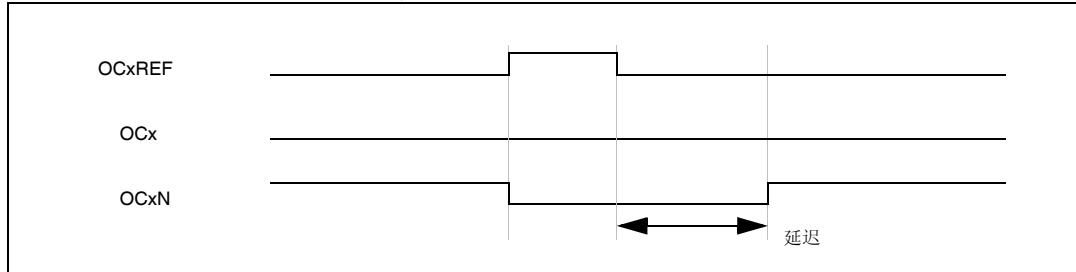


图 107. 延迟时间大于正脉冲宽度的死区波形



死区延迟对于所有通道均相同，可通过 `TIMx_BDTR` 寄存器中的 `DTG` 位进行编程。有关延迟时间计算的信息，请参见第 386 页的第 14.4.18 节：`TIM1 和 TIM8 断路和死区寄存器 (TIMx_BDTR)`。

将 `OCxREF` 重定向到 `OCx` 或 `OCxN`

在输出模式（强制输出模式、输出比较模式或 PWM 模式）下，通过配置 `TIMx_CCER` 寄存器中的 `CCxE` 和 `CCxNE` 位，可将 `OCxREF` 重定向到 `OCx` 输出或 `OCxN` 输出。

通过此功能，可以在一个输出上发送特定波形（如 PWM 或静态有效电平），而同时使互补输出保持其无效电平。或者，使两个输出同时保持无效电平，或者两个输出同时处于有效电平，两者互补并且带死区。

注意：如果仅使能 `OCxN` (`CCxE=0, CCxNE=1`)，两者不互补，一旦 `OCxREF` 为高电平，`OCxN` 即变为有效。例如，如果 `CCxNP=0`，则 `OCxN=OCxRef`。另一方面，如果同时使能 `OCx` 和 `OCxN` (`CCxE=CCxNE=1`)，`OCx` 在 `OCxREF` 为高电平时变为有效，而 `OCxN` 则与之互补，在 `OCxREF` 为低电平时变为有效。

14.3.12 使用断路功能

使用断路功能时，根据其它控制位（`TIMx_BDTR` 寄存器中的 `MOE`、`OSSI` 和 `OSSR` 位以及 `TIMx_CR2` 寄存器中的 `OISx` 和 `OISxN` 位）修改输出使能信号和无效电平。任何情况下，`OCx` 和 `OCxN` 输出都不能同时置为有效电平。更多详细信息，请参见第 382 页的表 73：具有断路功能的互补通道 `OCx` 和 `OCxN` 的输出控制位。

断路源可以是断路输入引脚，也可以是时钟故障事件，后者由复位时钟控制器中的时钟安全系统 (CSS) 生成。有关时钟安全系统的详细信息，请参见第 6.2.7 节：时钟安全系统 (CSS)。

退出复位状态后，断路功能处于禁止状态，`MOE` 位处于低电平。将 `TIMx_BDTR` 寄存器中的 `BKE` 位置 1，可使能断路功能。断路输入的极性可通过该寄存器中的 `BKP` 位来选择。`BKE` 和 `BKP` 位可同时修改。对 `BKE` 和 `BKP` 位执行写操作时，写操作会在 1 个 APB 时钟周期的延迟后生效。因此，执行写操作后，需要等待 1 个 APB 时钟周期，才能准确回读该位。

由于 `MOE` 下降沿可能是异步信号，因此在实际信号（作用于输出）与同步控制位（位于 `TIMx_BDTR` 寄存器中）之间插入了再同步电路，从而在异步信号与同步信号之间产生延迟。具体而言，如果在 `MOE` 处于低电平时向其写入 1，则必须首先插入延迟（空指令），才能准确进行读取。这是因为写入的是异步信号，而读取的却是同步信号。

发生断路（断路输入上出现所选电平）时：

- `MOE` 位异步清零，使输出处于无效状态、空闲状态或复位状态（通过 `OSSI` 位进行选择）。即使 MCU 振荡器关闭，该功能仍然有效。
- `MOE=0` 时，将以 `TIMx_CR2` 寄存器 `OISx` 位中编程的电平驱动每个输出通道。如果 `OSSI=0`，则定时器将释放使能输出，否则使能输出始终保持高电平。

- 使用互补输出时：
 - 输出首先置于复位状态或无效状态（取决于极性）。这是异步操作，因此即使没有为定时器提供时钟，该操作仍有效。
 - 如果定时器时钟仍存在，则将重新激活死区发生器，进而在死区后以 OISx 和 OISxN 位中编程的电平驱动输出。即使在这种情况下，也不能同时将 OCx 和 OCxN 驱动至其有效电平。请注意，MOE 进行再同步，因此死区的持续时间会比通常情况长一些（约 2 个 ck_tim 时钟周期）。
 - 如果 OSS1=0，则定时器会释放使能输出，否则只要 CCxE 位或 CCxNE 位处于高电平，使能输出就会保持或变为高电平。
- 将断路状态标志（TIMx_SR 寄存器中的 BIF 位）置 1。如果 TIMx_DIER 寄存器中的 BIE 位置 1，可产生中断。如果 TIMx_DIER 寄存器中的 BDE 位置 1，可发送 DMA 请求。
- 如果 TIMx_BDTR 寄存器中的 AOE 位置 1，则 MOE 位会在发生下一更新事件 (UEV) 时自动再次置 1。这一特性有许多用处，比如，可用于实现调节器的功能。否则，MOE 将始终保持低电平，直到再次向该位写入“1”。这种情况下，这一特性可用于确保安全。可以将断路输入连接到功率驱动器的警报、温度传感器或任何安全元件。

注意： 断路输入为电平有效。因此，当断路输入有效电平时，不能将 MOE 位置 1（自动或通过软件）。同时，不能将状态标志 BIF 清零。

断路可由 BRK 输入生成，该输入具有可编程极性，其使能位 BKE 位于 TIMx_BDTR 寄存器中。

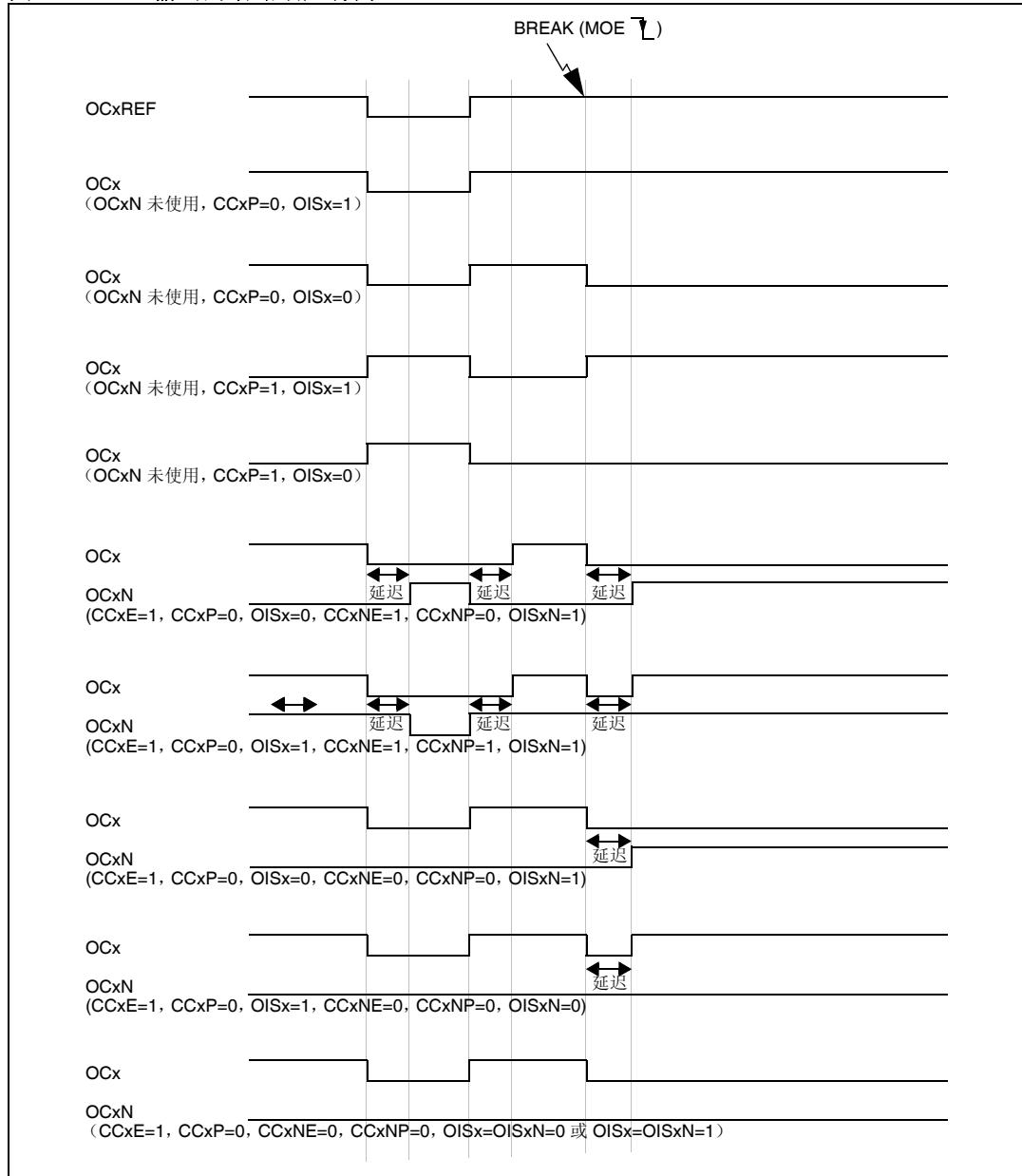
断路有以下两种生成方案：

- 使用 BRK 输入生成，该输入具有可编程极性，其使能位 BKE 位于 TIMx_BDTR 寄存器中。
- 由软件通过 TIMx_EGR 寄存器中的 BG 位生成。

除断路输入和输出管理外，断路电路内部还实施了写保护，用以保护应用的安全。通过该功能，用户可冻结多个参数配置（死区持续时间、OCx/OCxN 极性和禁止时的状态、OCxM 配置、断路使能和极性）。可以通过 TIMx_BDTR 寄存器中的 LOCK 位，从 3 种保护级别中进行选择。请参见[第 386 页的第 14.4.18 节：TIM1 和 TIM8 断路和死区寄存器 \(TIMx_BDTR\)](#)。MCU 复位后只能对 LOCK 位执行一次写操作。

图 108 所示为输出对断路响应行为的示例。

图 108. 输出的断路响应行为。



14.3.13 发生外部事件时清除 OCxREF 信号

对于给定通道，在 ETRF 输入施加高电平（相应 **TIMx_CCMRx** 寄存器中的 **OCxCE** 使能位置“1”），可使 **OCxREF** 信号变为低电平。**OCxREF** 信号将保持低电平，直到发生下一更新事件 (UEV)。

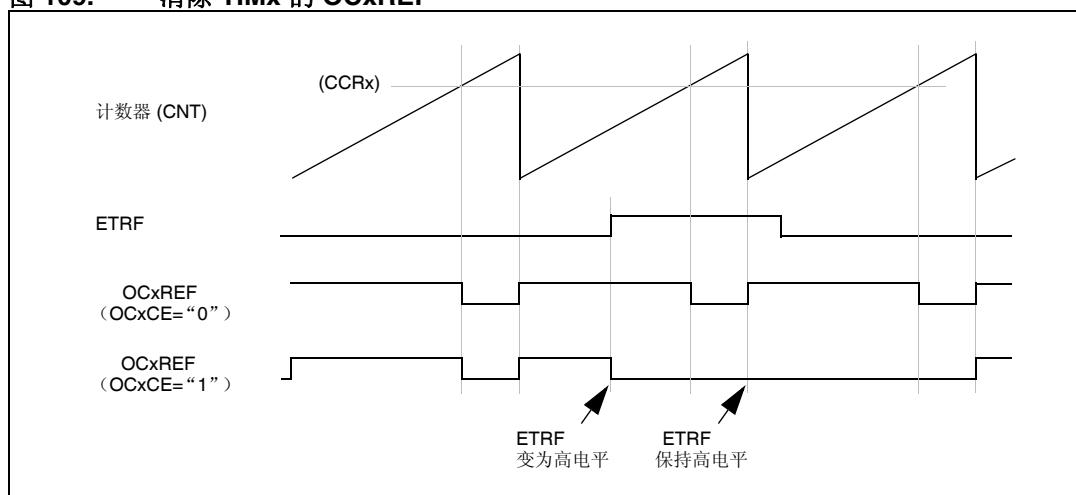
此功能仅能用于输出比较模式和 **PWM** 模式，而不适用于强制输出模式。

例如，**ETR** 信号可以连接到比较器的输出，用于控制电流。此时，**ETR** 必须如下配置：

1. 必须关闭外部触发预分频器：**TIMx_SMCR** 寄存器中的 **ETPS[1:0]** 位置“00”。
2. 必须禁止外部时钟模式 2：**TIMx_SMCR** 寄存器中的 **ECE** 位置“0”。
3. 外部触发极性 (ETP) 和外部触发滤波器 (ETF) 可根据用户需要进行配置。

[图 109](#) 对比了使能位 **OCxCE** 在不同值下的情况，显示了当 **ETRF** 输入变为高电平时 **OCxREF** 信号的行为。在本例中，定时器 **TIMx** 编程为 **PWM** 模式。

图 109. 清除 TIMx 的 OCxREF



注意：如果 **PWM** 的占空比为 100% ($CCRx > ARR$)，则下次计数器溢出时会再次使能 **OCxREF**。

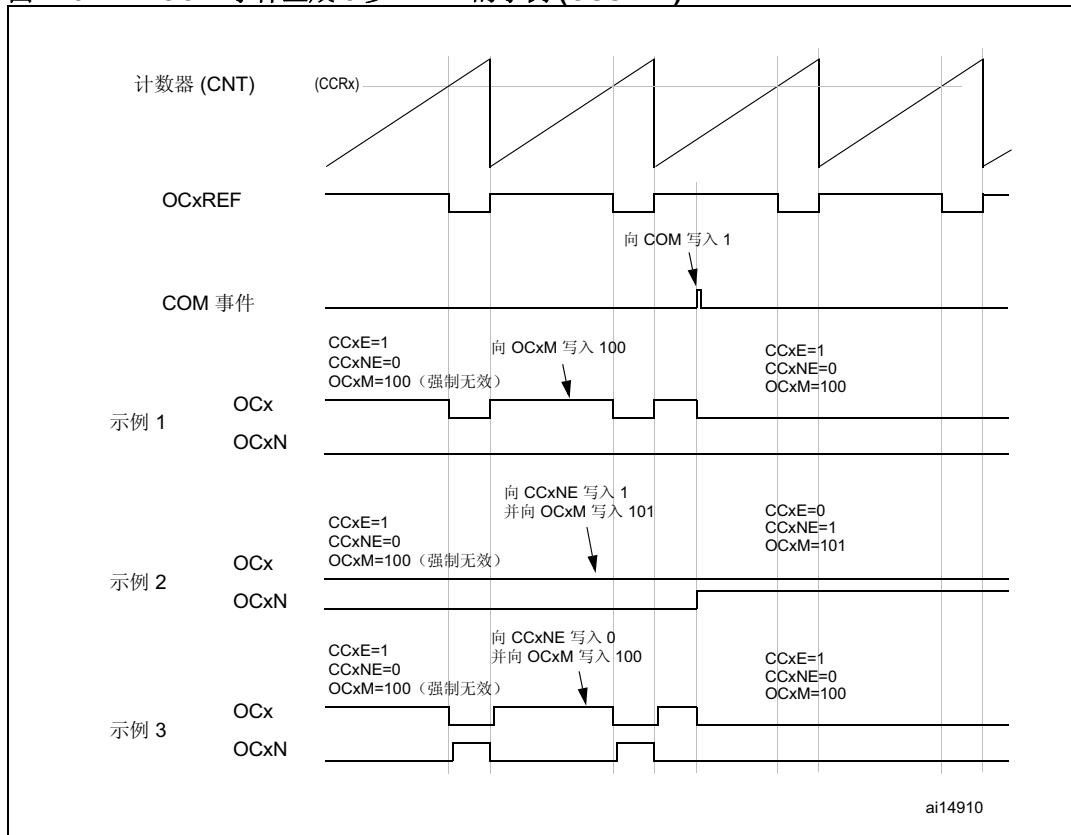
14.3.14 生成 6 步 PWM

当通道使用互补输出时，**OCxM**、**CCxE** 和 **CCxNE** 位上提供预装载位。发生 **COM** 换向事件时，这些预装载位将传输到影子位。因此，用户可以预先编程下一步骤的配置，并同时更改所有通道的配置。**COM** 可由软件通过将 **TIMx_EGR** 寄存器中的 **COM** 位置 1 而生成，也可以由硬件在 **TRGI** 上升沿生成。

发生 **COM** 事件时，某个标志位 (**TIMx_SR** 寄存器中的 **COMIF** 位) 将会置 1。这时，如果 **TIMx_DIER** 寄存器中的 **COMIE** 位置 1，将产生中断；如果 **TIMx_DIER** 寄存器中的 **COMDE** 位置 1，则将产生 DMA 请求。

图 110 以 3 种不同的编程配置为例，显示了发生 COM 事件时 OCx 和 OCxN 输出的行为。

图 110. COM 事件生成 6 步 PWM 的示例 (OSSR=1)



14.3.15 单脉冲模式

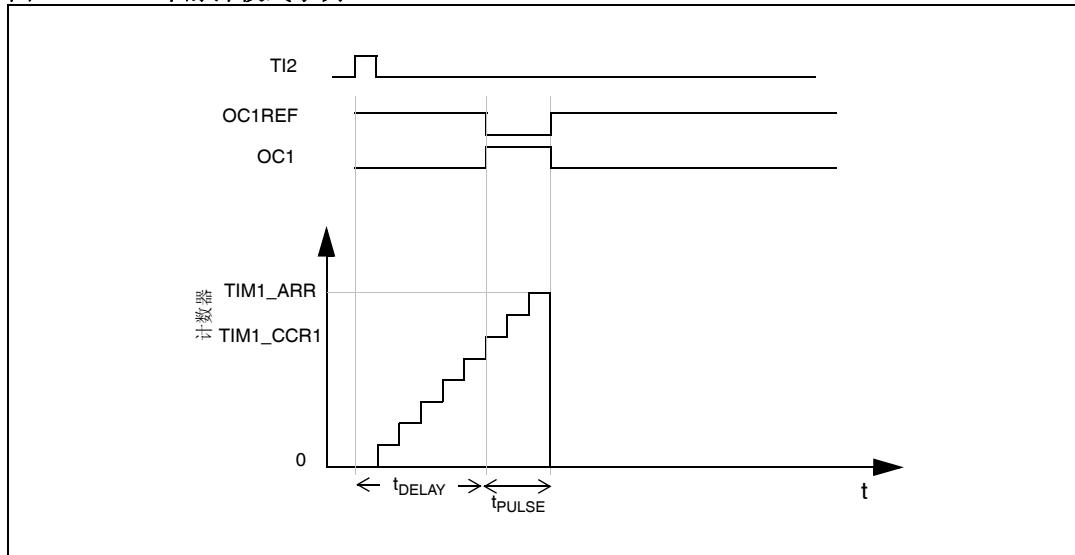
单脉冲模式 (OPM) 是上述模式的一个特例。在这种模式下，计数器可以在一个激励信号的触发下启动，并可在一段可编程的延时后产生一个脉宽可编程的脉冲。

可以通过从模式控制器启动计数器。可以在输出比较模式或 PWM 模式下生成波形。将 TIMx_CR1 寄存器中的 OPM 位置 1，即可选择单脉冲模式。这样，发生下一更新事件 UEV 时，计数器将自动停止。

只有当比较值与计数器初始值不同时，才能正确产生一个脉冲。启动前（定时器等待触发时），必须进行如下配置：

- 递增计数模式下： $CNT < CCRx \leq ARR$ (特别注意， $0 < CCRx$)
- 递减计数模式下： $CNT > CCRx$

图 111. 单脉冲模式示例:



例如，用户希望达到这样的效果：在 **TI2** 输入引脚检测到正沿时，经过 t_{DELAY} 的延迟，在 **OC1** 上产生一个长度为 t_{PULSE} 的正脉冲。

使用 **TI2FP2** 作为触发 1：

- 在 **TIMx_CCMR1** 寄存器中写入 **CC2S= “01”**，以将 **TI2FP2** 映射到 **TI2**。
- 在 **TIMx_CCER** 寄存器中写入 **CC2P= “0”** 和 **CC2NP= “0”**，使 **TI2FP2** 能够检测上升沿。
- 在 **TIMx_SMCR** 寄存器中写入 **TS= “110”**，以将 **TI2FP2** 配置为从模式控制器的触发 (**TRGI**)。
- 在 **TIMx_SMCR** 寄存器中写入 **SMS= “110”**（触发模式），以使用 **TI2FP2** 启动计数器。

OPM 波形通过对比较寄存器执行写操作来定义（考虑时钟频率和计数器预分频器）。

- t_{DELAY} 由写入 **TIMx_CCR1** 寄存器的值定义。
- t_{PULSE} 由自动重载值与比较值 (**TIMx_ARR - TIMx_CCR1**) 之差来定义。
- 假设希望产生这样的波形：信号在发生比较匹配时从“0”变为“1”，在计数器达到自动重载值时由“1”变为“0”。为此，应在 **TIMx_CCMR1** 寄存器中写入 **OC1M=111**，以使能 **PWM** 模式 2。如果需要，可选择在 **TIMx_CCMR1** 寄存器的 **OC1PE** 和 **TIMx_CR1** 寄存器的 **ARPE** 中写入“1”，以使能预装载寄存器。这种情况下，必须在 **TIMx_CCR1** 寄存器中写入比较值并在 **TIMx_ARR** 寄存器中写入自动重载值，通过将 **UG** 位置 1 来产生更新，然后等待 **TI2** 上的外部触发事件。本例中，**CC1P** 的值为“0”。

在本例中，**TIMx_CR1** 寄存器中的 **DIR** 和 **CMS** 位应为低。

由于仅需要 1 个脉冲（单脉冲模式），因此应向 **TIMx_CR1** 寄存器的 **OPM** 位写入“1”，以便在发生下一更新事件（计数器从自动重载值返回到 0）时使计数器停止计数。**TIMx_CR1** 寄存器中的 **OPM** 位置“0”时，即选择重复模式。

特例：OCx 快速使能：

在单脉冲模式下，**TIx** 输入的边沿检测会将 **CEN** 位置 1，表示使能计数器。然后，在计数器值与比较值之间发生比较时，将切换输出。但是，完成这些操作需要多个时钟周期，这会限制可能的最小延迟 (t_{DELAY} 最小值)。

如果要输出延迟时间最短的波形，可以将 **TIMx_CCMRx** 寄存器中的 **OCxFE** 位置 1。这样会强制 **OCxRef**（和 **OCx**）对激励信号做出响应，而不再考虑比较的结果。其新电平与发生比较匹配时相同。仅当通道配置为 **PWM1** 或 **PWM2** 模式时，**OCxFE** 才会起作用。

14.3.16 编码器接口模式

选择编码器接口模式时，如果计数器仅在 **TI2** 边沿处计数，在 **TIMx_SMCR** 寄存器中写入 **SMS=“001”**；如果计数器仅在 **TI1** 边沿处计数，写入 **SMS=“010”**；如果计数器在 **TI1** 和 **TI2** 边沿处均计数，则写入 **SMS=“011”**。

通过编程 **TIMx_CCER** 寄存器的 **CC1P** 和 **CC2P** 位，选择 **TI1** 和 **TI2** 极性。如果需要，还可对输入滤波器进行编程。**CC1NP** 和 **CC2NP** 必须保持低电平。

TI1 和 **TI2** 两个输入用于连接增量编码器。请参见表 71。如果使能计数器（在 **TIMx_CR1** 寄存器的 **CEN** 位中写入“1”），则计数器的时钟由 **TI1FP1** 或 **TI2FP2** 上的每次有效信号转换提供。**TI1FP1** 和 **TI2FP2** 是进行输入滤波器和极性选择后 **TI1** 和 **TI2** 的信号，如果不进行滤波和反相，则 **TI1FP1=TI1**，**TI2FP2=TI2**。将根据两个输入的信号转换序列，产生计数脉冲和方向信号。根据该信号转换序列，计数器相应递增或递减计数，同时硬件对 **TIMx_CR1** 寄存器的 **DIR** 位进行相应修改。任何输入（**TI1** 或 **TI2**）发生信号转换时，都会计算 **DIR** 位，无论计数器是仅在 **TI1** 或 **TI2** 边沿处计数，还是同时在 **TI1** 和 **TI2** 处计数。

编码器接口模式就相当于带有方向选择的外部时钟。这意味着，计数器仅在 0 到 **TIMx_ARR** 寄存器中的自动重载值之间进行连续计数（根据具体方向，从 0 递增计数到 **ARR**，或从 **ARR** 递减计数到 0）。因此，在启动前必须先配置 **TIMx_ARR**。同样，捕获、比较、预分频器、重复计数器及触发输出功能继续正常工作。编码器模式和外部时钟模式 2 不兼容，因此不能同时选择。

在此模式下，计数器会根据增量编码器的速度和方向自动进行修改，因此，其内容始终表示编码器的位置。计数方向对应于所连传感器的旋转方向。下表汇总了可能的组合（假设 **TI1** 和 **TI2** 不同时切换）。

表 71. 计数方向与编码器信号的关系

有效边沿	相反信号的电平（ 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 处计数	高	递减	递增	不计数	不计数
	低	递增	递减	不计数	不计数
仅在 TI2 处计数	高	不计数	不计数	递增	递减
	低	不计数	不计数	递减	递增
在 TI1 和 TI2 处均计数	高	递减	递增	递增	递减
	低	递增	递减	递减	递增

外部增量编码器可直接与 MCU 相连，无需外部接口逻辑。不过，通常使用比较器将编码器的差分输出转换为数字信号。这样大幅提高了抗噪声性能。用于指示机械零位的第三个编码器输出可与外部中断输入相连，用以触发计数器复位。

图 112 以计数器工作为例，说明了计数信号的生成和方向控制。同时也说明了选择双边沿时如何对输入抖动进行补偿。将传感器靠近其中一个切换点放置时可能出现这种情况。本例中假设配置如下：

- CC1S=“01”（TIMx_CCMR1 寄存器，TI1FP1 映射到 TI1 上）。
- CC2S=“01”（TIMx_CCMR2 寄存器，TI1FP2 映射到 TI2 上）。
- CC1P=“0”，CC1NP=“0”，且 IC1F=“0000”（TIMx_CCER 寄存器，TI1FP1 未反相，TI1FP1=TI1）。
- CC2P=“0”，CC2NP=“0”，且 IC2F=“0000”（TIMx_CCER 寄存器，TI1FP2 未反相，TI1FP2=TI2）。
- SMS=“011”（TIMx_SMCR 寄存器，两个输入在上升沿和下降沿均有效）。
- CEN=“1”（TIMx_CR1 寄存器，使能计数器）。

图 112. 编码器接口模式下的计数器工作示例。

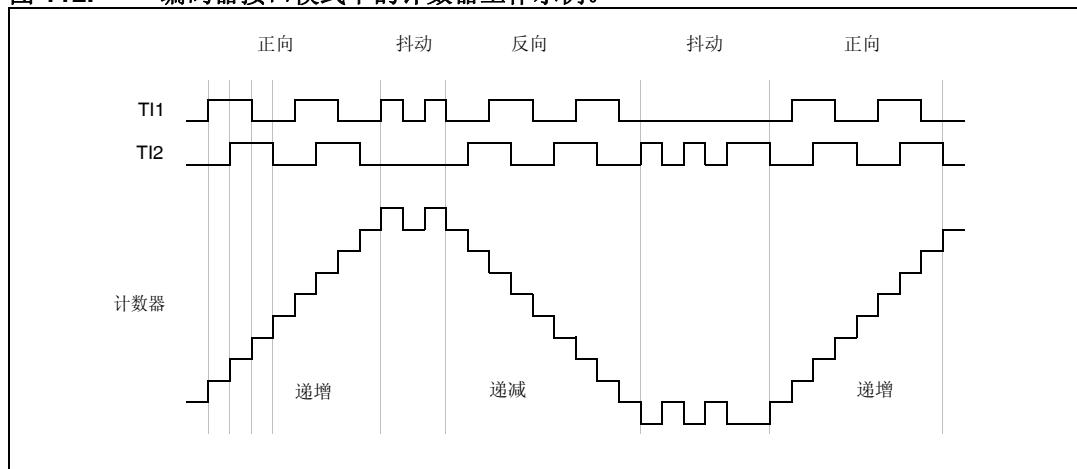
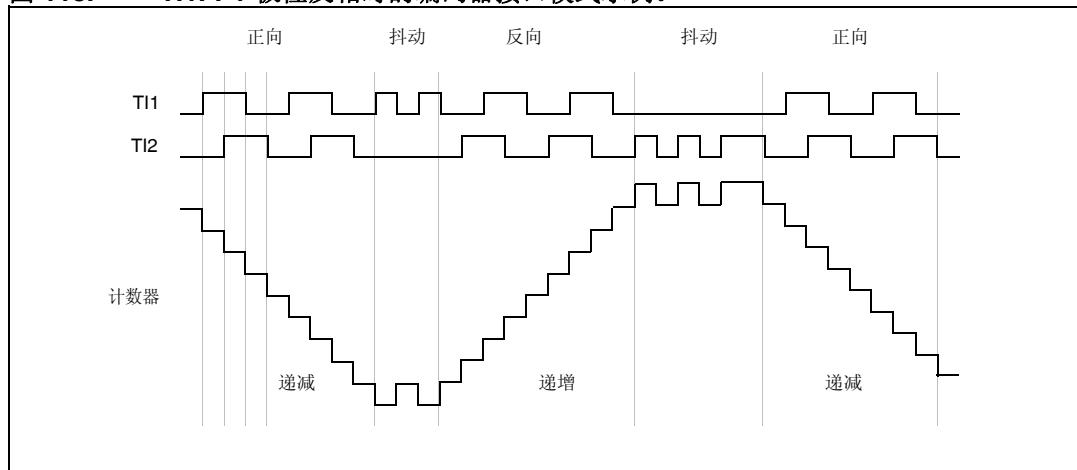


图 113 举例说明 TI1FP1 极性反相时计数器的行为（除 CC1P=“1” 外，其它配置与上例相同）。

图 113. TI1FP1 极性反相时的编码器接口模式示例。



定时器配置为编码器接口模式时，会提供传感器当前位置的相关信息。使用另一个配置为捕获模式的定时器测量两个编码器事件之间的周期，可获得动态信息（速度、加速度和减速度）。指示机械零位的编码器输出即可用于此目的。根据两个事件之间的时间间隔，还可定期读取计数器。如果可能，可以将计数器值锁存到第三个输入捕获寄存器来实现此目的（捕获信号必须为周期性信号，可以由另一个定时器产生）；还可以通过由实时时钟生成的 DMA 请求读取计数器值。

14.3.17 定时器输入异或功能

通过 **TIMx_CR2** 寄存器中的 **TI1S** 位，可将通道 1 的输入滤波器连接到异或门的输出，从而将 **TIMx_CH1** 到 **TIMx_CH3** 这三个输入引脚组合在一起。

异或输出可与触发或输入捕获等所有定时器输入功能配合使用。下面的 [第 14.3.18 节](#)以连接霍尔传感器为例介绍了此功能。

14.3.18 连接霍尔传感器

可通过用于生成电机驱动 PWM 信号的高级控制定时器（TIM1 或 TIM8）以及 [图 114](#) 中称为“接口定时器”的另一个定时器 TIMx（TIM2、TIM3、TIM4 或 TIM5），实现与霍尔传感器的连接。3 个定时器输入引脚（**TIMx_CH1**、**TIMx_CH2** 和 **TIMx_CH3**）通过异或门连接到 **TI1** 输入通道（通过将 **TIMx_CR2** 寄存器中的 **TI1S** 位置 1 来选择），并由“接口定时器”进行捕获。

从模式控制器配置为复位模式；从输入为 **TI1F_ED**。这样，每当 3 个输入中有一个输入发生切换时，计数器会从 0 开始重新计数。这样将产生由霍尔输入的任何变化而触发的时基。

在“接口定时器”上，捕获/比较通道 1 配置为捕获模式，捕获信号为 **TRC**（请参见 [第 344 页的图 97：捕获/比较通道（例如：通道 1 输入阶段）](#)）。捕获值对应于输入上两次变化的间隔时间，可提供与电机转速相关的信息。

“接口定时器”可用于在输出模式下产生脉冲，以通过触发 **COM** 事件更改高级控制定时器（TIM1 或 TIM8）各个通道的配置。TIM1 定时器用于生成电机驱动 PWM 信号。为此，必须对接口定时器通道进行编程，以便在编程的延迟过后产生正脉冲（在输出比较或 PWM 模式中）。该脉冲通过 **TRGO** 输出发送到高级控制定时器（TIM1 或 TIM8）。

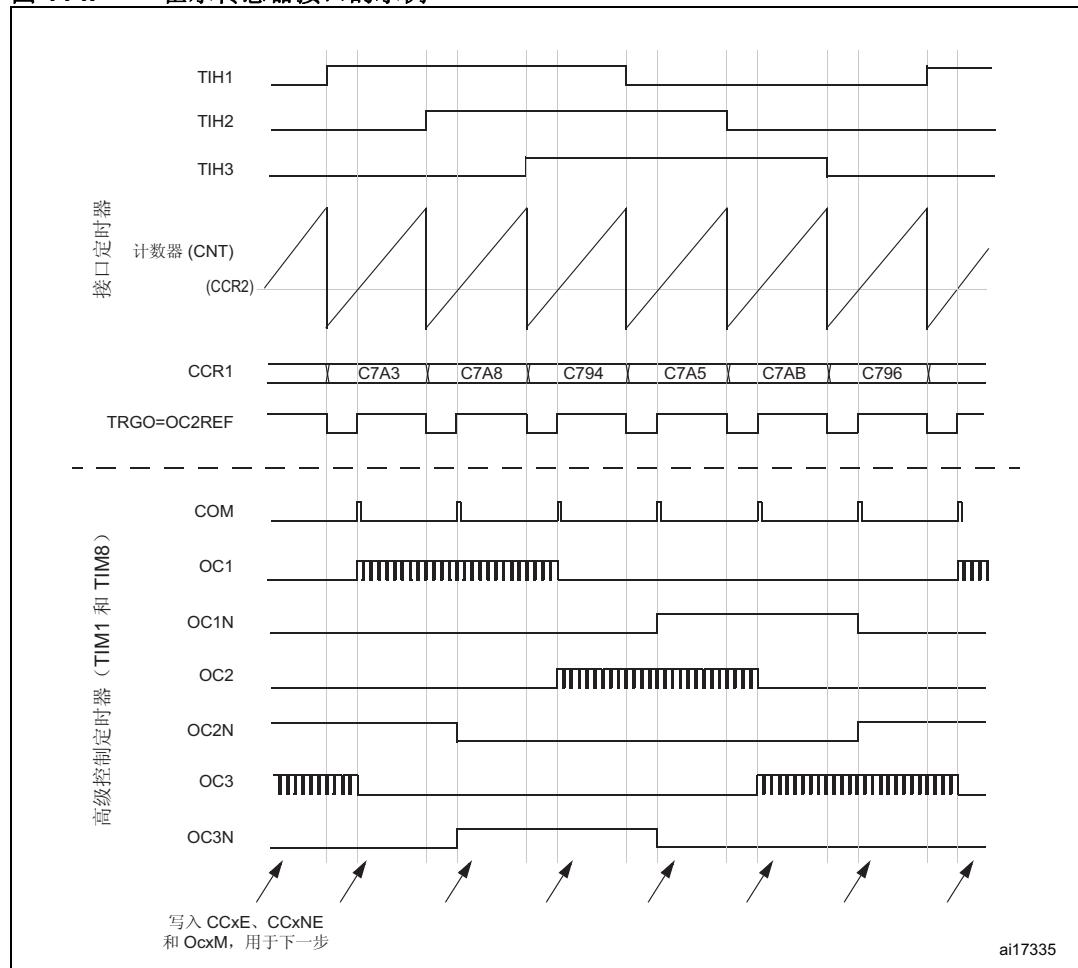
示例：霍尔输入与一个 TIMx 定时器相连接，每当霍尔输入发生更改，需要在所编程的延迟过后更改高级控制定时器 TIM1 的 PWM 配置。

- 向 **TIMx_CR2** 寄存器的 **TI1S** 位写入“1”，使 3 个定时器输入经过异或运算后进入 **TI1** 输入通道。
- 时基编程：向 **TIMx_ARR** 写入其最大值（计数器必须通过 **TI1** 的变化清零）。设置预分频器，以得到最大计数器周期，该周期长于传感器上两次变化的间隔时间。
- 将通道 1 编程为捕获模式（选择 **TRC**）：向 **TIMx_CCMR1** 寄存器的 **CC1S** 位写入“11”。如果需要，还可以编程数字滤波器。
- 将通道 2 编程为 PWM 2 模式，并具有所需延迟：向 **TIMx_CCMR1** 寄存器的 **OC2M** 位写入“111”，**CC2S** 位写入“00”。
- 选择 **OC2REF** 作为 **TRGO** 上的触发输出：向 **TIMx_CR2** 寄存器的 **MMS** 位写入“101”。

在高级控制定时器 TIM1 中，必须选择正确的 ITR 输入作为触发输入，定时器编程为可产生 PWM 信号，捕获/比较控制信号进行预装载 (TIMx_CR2 寄存器的 CCPC=1)，并且 COM 事件由触发输入控制 (TIMx_CR2 寄存器中 CCUS=1)。发生 COM 事件后，在 PWM 控制位 (CCxE、OCxM) 中写入下一步的配置，此操作可在由 OC2REF 上升沿产生的中断子程序中完成。

[图 114](#) 为本示例的示意图。

图 114. 霍尔传感器接口的示例



14.3.19 TIMx 与外部触发同步

TIMx 定时器可与外部触发以下列模式实现同步：复位模式、门控模式和触发模式。

从模式：复位模式

当触发输入信号产生变化时，计数器及其预分频器可重新初始化。此外，如果 **TIMx_CR1** 寄存器中的 **URS** 位处于低电平，则会生成更新事件 **UEV**。然后，所有预装载寄存器 (**TIMx_ARR** 和 **TIMx_CCRx**) 都将更新。

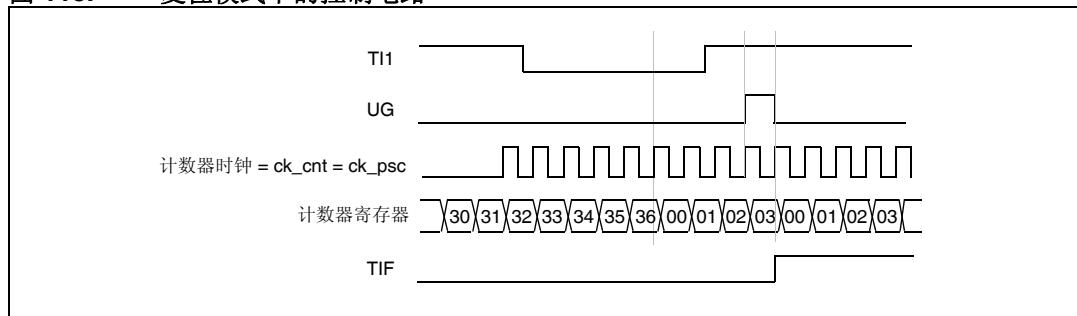
在以下示例中，**TI1** 输入上出现上升沿时，递增计数器清零：

- 将通道 1 配置为检测 **TI1** 的上升沿。配置输入滤波时间（本例中不需要任何滤波，因此保持 **IC1F=0000**）。由于捕获预分频器不用于触发操作，因此无需对其进行配置。**CC1S** 位只选择输入捕获源，即 **TIMx_CCMR1** 寄存器中的 **CC1S = 01**。在 **TIMx_CCER** 寄存器中写入 **CC1P=0** 和 **CC1NP='0'**，验证极性（仅检测上升沿）。
- 在 **TIMx_SMCR** 寄存器中写入 **SMS=100**，将定时器配置为复位模式。在 **TIMx_SMCR** 寄存器中写入 **TS=101**，选择 **TI1** 作为输入源。
- 在 **TIMx_CR1** 寄存器中写入 **CEN=1**，启动计数器。

计数器使用内部时钟计数，然后正常运转，直到出现 **TI1** 上升沿。当 **TI1** 出现上升沿时，计数器清零，然后重新从 0 开始计数。同时，触发标志 (**TIMx_SR** 寄存器中的 **TIF** 位) 置 1，使能中断或 DMA 后，还可发送中断或 DMA 请求（取决于 **TIMx_DIER** 寄存器中的 **TIE** 和 **TDE** 位）。

下图显示了自动重载寄存器 **TIMx_ARR=0x36** 时的相关行为。**TI1** 的上升沿与实际计数器复位之间的延迟是由于 **TI1** 输入的重新同步电路引起的。

图 115. 复位模式下的控制电路



从模式：门控模式

输入信号的电平可用来使能计数器。

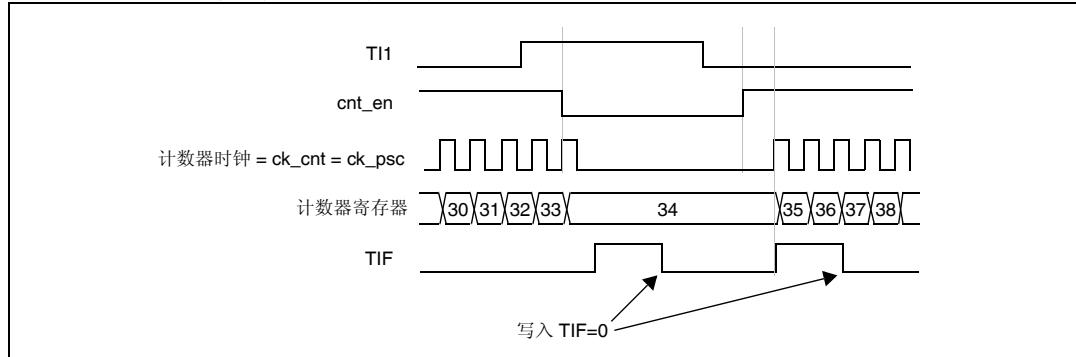
在以下示例中，递增计数器仅在 **TI1** 输入为低电平时计数：

- 将通道 1 配置为检测 **TI1** 上的低电平。配置输入滤波时间（本例中不需要任何滤波，因此保持 **IC1F=0000**）。由于捕获预分频器不用于触发操作，因此无需对其进行配置。**CC1S** 位只选择输入捕获源，即 **TIMx_CCMR1** 寄存器中的 **CC1S=01**。在 **TIMx_CCER** 寄存器中写入 **CC1P=1** 和 **CC1NP=“0”**，以确定极性（仅检测低电平）。
- 在 **TIMx_SMCR** 寄存器中写入 **SMS=101**，将定时器配置为门控模式。在 **TIMx_SMCR** 寄存器中写入 **TS=101**，选择 **TI1** 作为输入源。
- 在 **TIMx_CR1** 寄存器中写入 **CEN=1**，使能计数器（在门控模式下，如果 **CEN=0**，则无论触发输入电平如何，计数器都不启动）。

只要 TI1 为低电平，计数器就开始根据内部时钟计数，直到 TI1 变为高电平时停止计数。计数器启动或停止时，TIMx_SR 寄存器中的 TIF 标志都会置 1。

TI1 的上升沿与实际计数器停止之间的延迟是由于 TI1 输入的重新同步电路引起的。

图 116. 门控模式下的控制电路



从模式：触发模式

所选输入上发生某一事件时可以启动计数器。

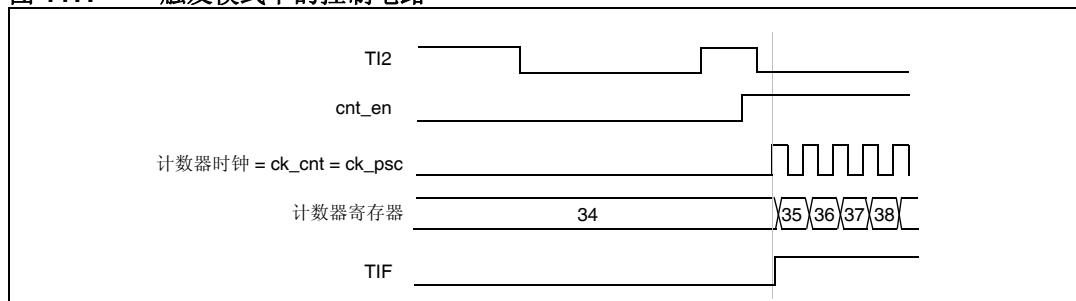
在以下示例中，TI2 输入上出现上升沿时，递增计数器启动：

- 将通道 2 配置为检测 TI2 上的上升沿。配置输入滤波时间（本例中不需要任何滤波，因此保持 IC2F=0000）。由于捕获预分频器不用于触发操作，因此无需对其进行配置。CC2S 位只选择输入捕获源，即 TIMx_CCMR1 寄存器中的 CC2S=01。在 TIMx_CCER 寄存器中写入 CC2P=1 和 CC2NP=0，以确定极性（仅检测低电平）。
- 在 TIMx_SMCR 寄存器中写入 SMS=110，将定时器配置为触发模式。在 TIMx_SMCR 寄存器中写入 TS=110，选择 TI2 作为输入源。

当 TI2 出现上升沿时，计数器开始根据内部时钟计数，并且 TIF 标志置 1。

TI2 的上升沿与实际计数器启动之间的延迟是由于 TI2 输入的重新同步电路引起的。

图 117. 触发模式下的控制电路



从模式：外部时钟模式 2 + 触发模式

外部时钟模式 2 可与另一种从模式（外部时钟模式 1 和编码器模式除外）结合使用。这种情况下，ETR 信号用作外部时钟输入，在复位模式、门控模式或触发模式下工作时，可选择另一个输入作为触发输入。不建议通过 TIMx_SMCR 寄存器中的 TS 位来选择 ETR 作为 TRGI。

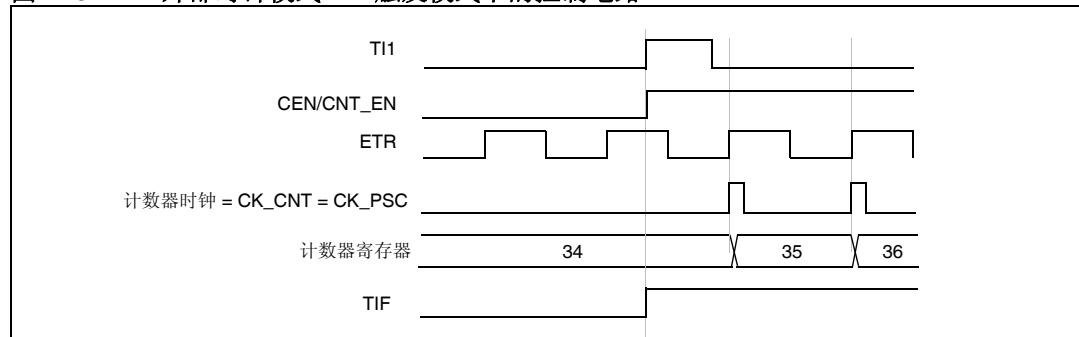
在以下示例中，只要 TI1 出现上升沿，递增计数器即会在 ETR 信号的每个上升沿处递增：

1. 通过对 TIMx_SMCR 寄存器进行如下编程，配置外部触发输入电路：
 - ETF = 0000：无滤波器
 - ETPS = 00：禁止预分频器
 - ETP = 0：检测 ETR 的上升沿，并写入 ECE=1，以便能外部时钟模式 2。
2. 如下配置通道 1，以检测 TI 的上升沿：
 - IC1F=0000：无滤波器。
 - 由于捕获预分频器不用于触发操作，因此无需对其进行配置。
 - TIMx_CCMR1 寄存器中 CC1S=01，只选择输入捕获源。
 - TIMx_CCER 寄存器中 CC1P=0 且 CC1NP=“0”，以确定极性（仅检测上升沿）。
3. 在 TIMx_SMCR 寄存器中写入 SMS=110，将定时器配置为触发模式。在 TIMx_SMCR 寄存器中写入 TS=101，选择 TI1 作为输入源。

TI1 出现上升沿时将使能计数器并且 TIF 标志置 1。然后计数器在 ETR 出现上升沿时计数。

ETR 信号的上升沿与实际计数器复位之间的延迟是由于 ETRP 输入的重新同步电路引起的。

图 118. 外部时钟模式 2 + 触发模式下的控制电路



14.3.20 定时器同步

TIM 定时器从内部链接在一起，以实现定时器同步或级联。有关详细信息，请参见[第 419 页的第 15.3.15 节：定时器同步](#)。

14.3.21 调试模式

当微控制器进入调试模式（Cortex™-M4F 内核停止）时，TIMx 计数器会根据 DBG 模块中的 DBG_TIMx_STOP 配置位选择继续正常工作或者停止工作。有关详细信息，请参见[第 33.16.2 节：对定时器、看门狗、bxCAN 和 PC 的调试支持](#)。

14.4 TIM1 和 TIM8 寄存器

有关寄存器说明中使用的缩写，请参见[第 47 页的第 1.1 节](#)。

外设寄存器必须按半字（16 位）或字（32 位）进行写访问。而读访问则可按字节（8 位）、半字（16 位）或字（32 位）进行。

14.4.1 TIM1 和 TIM8 控制寄存器 1 (TIMx_CR1)

TIM1&TIM8 control register 1

偏移地址: 0x00

复位值: 0x0000

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

位 15:10 保留，必须保持复位值。

位 9:8 **CKD[1:0]**: 时钟分频 (Clock division)

此位域指示定时器时钟 (CK_INT) 频率与死区发生器以及数字滤波器 (ETR、Tlx) 所使用的死区及采样时钟 (t_{DTS}) 之间的分频比，

- 00: $t_{DTS} = t_{CK_INT}$
- 01: $t_{DTS} = 2 * t_{CK_INT}$
- 10: $t_{DTS} = 4 * t_{CK_INT}$
- 11: 保留，不要设置成此值

位 7 **ARPE**: 自动重载预装载使能 (Auto-reload preload enable)

- 0: TIMx_ARR 寄存器不进行缓冲
- 1: TIMx_ARR 寄存器进行缓冲

位 6:5 **CMS[1:0]**: 中心对齐模式选择 (Center-aligned mode selection)

- 00: 边沿对齐模式。计数器根据方向位 (DIR) 递增计数或递减计数。
- 01: 中心对齐模式 1。计数器交替进行递增计数和递减计数。仅当计数器递减计数时，配置为输出的通道 (TIMx_CCMRx 寄存器中的 CxS=00) 的输出比较中断标志才置 1。
- 10: 中心对齐模式 2。计数器交替进行递增计数和递减计数。仅当计数器递增计数时，配置为输出的通道 (TIMx_CCMRx 寄存器中的 CxS=00) 的输出比较中断标志才置 1。
- 11: 中心对齐模式 3。计数器交替进行递增计数和递减计数。当计数器递增计数或递减计数时，配置为输出的通道 (TIMx_CCMRx 寄存器中的 CxS=00) 的输出比较中断标志都会置 1。

注意：只要计数器处于使能状态 (CEN=1)，就不得从边沿对齐模式切换为中心对齐模式。

位 4 **DIR**: 方向 (Direction)

- 0: 计数器递增计数
- 1: 计数器递减计数

注意：当定时器配置为中心对齐模式或编码器模式时，该位为只读状态。

位 3 **OPM**: 单脉冲模式 (One pulse mode)

- 0: 计数器在发生更新事件时不会停止计数
- 1: 计数器在发生下一更新事件时停止计数 (将 CEN 位清零)

位 2 **URS**: 更新请求源 (Update request source)

此位由软件置 1 和清零，用以选择 UEV 事件源。

- 0: 使能时，所有以下事件都会生成更新中断或 DMA 请求。此类事件包括：

- 计数器上溢/下溢
- 将 UG 位置 1
- 通过从模式控制器生成的更新事件

- 1: 使能时，只有计数器上溢/下溢会生成更新中断或 DMA 请求。

位 1 UDIS: 更新禁止 (Update disable)

此位由软件置 1 和清零，用以使能/禁止 UEV 事件生成。

0: 使能 UEV。更新 (UEV) 事件可通过以下事件之一生成：

- 计数器上溢/下溢
- 将 UG 位置 1
- 通过从模式控制器生成的更新事件

然后更新影子寄存器的值。

1: 禁止 UEV。不会生成更新事件，各影子寄存器的值 (ARR、PSC 和 CCRx) 保持不变。但如果将 UG 位置 1，或者从模式控制器接收到硬件复位，则会重新初始化计数器和预分频器。

位 0 CEN: 计数器使能 (Counter enable)

0: 禁止计数器

1: 使能计数器

注意：只有事先通过软件将 CEN 位置 1，才可以使用外部时钟、门控模式和编码器模式。而触发模式可通过硬件自动将 CEN 位置 1。

14.4.2 TIM1 和 TIM8 控制寄存器 2 (TIMx_CR2)

TIM1&TIM8 control register 2

偏移地址：0x04

复位值：0x0000

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]	CCDS	CCUS	Res.	CCPC			
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位 15 保留，必须保持复位值。

位 14 OIS4: 输出空闲状态 4 (OC4 输出) (Output Idle state 4 (OC4 output))

参见 OIS1 位

位 13 OIS3N: 输出空闲状态 3 (OC3N 输出) (Output Idle state 3 (OC3N output))

参见 OIS1N 位

位 12 OIS3: 输出空闲状态 3 (OC3 输出) (Output Idle state 3 (OC3 output))

参见 OIS1 位

位 11 OIS2N: 输出空闲状态 2 (OC2N 输出) (Output Idle state 2 (OC2N output))

参见 OIS1N 位

位 10 OIS2: 输出空闲状态 2 (OC2 输出) (Output Idle state 2 (OC2 output))

参见 OIS1 位

位 9 OIS1N: 输出空闲状态 1 (OC1N 输出) (Output Idle state 1 (OC1N output))

0: 当 MOE=0 时，经过死区时间后 OC1N=0

1: 当 MOE=0 时，经过死区时间后 OC1N=1

注意：只要编程了 LOCK (TIMx_BDTR 寄存器中的 LOCK 位) 级别 1、2 或 3，此位即无法修改。

位 8 **OIS1**: 输出空闲状态 1 (OC1 输出) (Output Idle state 1 (OC1 output))

- 0: 当 MOE=0 时, (如果 OC1N 有效, 则经过死区时间之后) OC1=0
- 1: 当 MOE=0 时, (如果 OC1N 有效, 则经过死区时间之后) OC1=1

注意: 只要编程了 *LOCK* (*TIMx_BDTR* 寄存器中的 *LOCK* 位) 级别 1、2 或 3, 此位即无法修改。

位 7 **TI1S**: TI1 选择 (TI1 selection)

- 0: *TIMx_CH1* 引脚连接到 TI1 输入
- 1: *TIMx_CH1*、*CH2* 和 *CH3* 引脚连接到 TI1 输入 (异或组合)

位 6:4 **MMS[1:0]**: 主模式选择 (Master mode selection)

这些位可选择主模式下将要发送到从定时器以实现同步的信息 (TRGO)。这些位的组合如下:

000: 复位——*TIMx_EGR* 寄存器中的 *UG* 位用作触发输出 (TRGO)。如果复位由触发输入生成 (从模式控制器配置为复位模式), 则 TRGO 上的信号相比实际复位会有延迟。

001: 使能——计数器使能信号 *CNT_EN* 用作触发输出 (TRGO)。该触发输出可用于同时启动多个定时器, 或者控制在一段时间内使能从定时器。计数器使能信号可由 *CEN* 控制位产生。当配置为门控模式时, 也可由触发输入产生。当计数器使能信号由触发输入控制时, TRGO 上会存在延迟, 选择主/从模式时除外 (请参见 *TIMx_SMCR* 寄存器中 *MSM* 位的说明)。

010: 更新——选择更新事件作为触发输出 (TRGO)。例如, 主定时器可用作从定时器的预分频器。

011: 比较脉冲——一旦发生输入捕获或比较匹配事件, 当 *CC1IF* 被置 1 时 (即使已为高电平), 触发输出都会发送一个正脉冲。(TRGO)。

100: 比较——*OC1REF* 信号用作触发输出 (TRGO)

101: 比较——*OC2REF* 信号用作触发输出 (TRGO)

110: 比较——*OC3REF* 信号用作触发输出 (TRGO)

111: 比较——*OC4REF* 信号用作触发输出 (TRGO)

位 3 **CCDS**: 捕获/比较 DMA 选择 (Capture/compare DMA selection)

- 0: 发生 *CCx* 事件时发送 *CCx DMA* 请求
- 1: 发生更新事件时发送 *CCx 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 (CCPC=1) 进行预装载, 仅通过将 *COMG* 位置 1 来对这些位进行更新

1: 如果捕获/比较控制位 (CCPC=1) 进行预装载, 可通过将 *COMG* 位置 1 或 *TRGI* 的上升沿对这些位进行更新。

注意: 此位仅对具有互补输出的通道有效。

位 1 保留, 必须保持复位值。

位 0 **CCPC**: 捕获/比较预装载控制 (Capture/compare preloaded control)

0: *CCxE*、*CCxNE* 和 *OCxM* 位未进行预装载

1: *CCxE*、*CCxNE* 和 *OCxM* 位进行了预装载, 写入这些位后, 仅当发生换向事件 (*COM*) (*COMG* 位置 1 或在 *TRGI* 上检测到上升沿, 取决于 *CCUS* 位) 时才会对这些位进行更新。

注意: 此位仅对具有互补输出的通道有效。

14.4.3 TIM1 和 TIM8 从模式控制寄存器 (*TIMx_SMCR*)

TIM1&TIM8 slave mode control register

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]		Res.	SMS[2:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	Res.	rw	rw	rw

位 15 **ETP**: 外部触发极性 (External trigger polarity)

此位可选择将 ETR 还是 \overline{ETR} 用于触发操作

0: ETR 未反相, 高电平或上升沿有效。

1: ETR 反相, 低电平或下降沿有效。

位 14 **ECE**: 外部时钟使能 (External clock enable)

此位可使能外部时钟模式 2。

0: 禁止外部时钟模式 2

1: 使能外部时钟模式 2。计数器时钟由 ETRF 信号的任意有效边沿提供。

注意: 1: 将 ECE 位置 1 与选择外部时钟模式 1 并将 TRGI 连接到 ETRF (SMS=111 且 TS=111) 具有相同效果。

2: 外部时钟模式 2 可以和以下从模式同时使用: 复位模式、门控模式和触发模式。不过此类情况下 TRGI 不得连接 ETRF (TS 位不得为 111)。

3: 如果同时使能外部时钟模式 1 和外部时钟模式 2, 则外部时钟输入为 ETRF。

位 13:12 **ETPS[1:0]**: 外部触发预分频器 (External trigger prescaler)

外部触发信号 ETRP 频率不得超过 TIMxCLK 频率的 1/4。可通过使能预分频器来降低 ETRP 频率。这种方法在输入快速外部时钟时非常有用。

00: 预分频器关闭

01: 2 分频 ETRP 频率

10: 4 分频 ETRP 频率

11: 8 分频 ETRP 频率

位 11:8 **ETF[3:0]**: 外部触发滤波器 (External trigger filter)

此位域可定义 ETRP 信号的采样频率和适用于 ETRP 的数字滤波时间。数字滤波器由事件计数器组成, 每 N 个事件才视为一个有效边沿:

0000: 无滤波器, 按 f_{DTS} 频率进行采样

0001: $f_{SAMPLING}=f_{CK_INT}$, N=2

0010: $f_{SAMPLING}=f_{CK_INT}$, N=4

0011: $f_{SAMPLING}=f_{CK_INT}$, N=8

0100: $f_{SAMPLING}=f_{DTS}/2$, N=6

0101: $f_{SAMPLING}=f_{DTS}/2$, N=8

0110: $f_{SAMPLING}=f_{DTS}/4$, N=6

0111: $f_{SAMPLING}=f_{DTS}/4$, N=8

1000: $f_{SAMPLING}=f_{DTS}/8$, N=6

1001: $f_{SAMPLING}=f_{DTS}/8$, N=8

1010: $f_{SAMPLING}=f_{DTS}/16$, N=5

1011: $f_{SAMPLING}=f_{DTS}/16$, N=6

1100: $f_{SAMPLING}=f_{DTS}/16$, N=8

1101: $f_{SAMPLING}=f_{DTS}/32$, N=5

1110: $f_{SAMPLING}=f_{DTS}/32$, N=6

1111: $f_{SAMPLING}=f_{DTS}/32$, N=8

位 7 **MSM:** 主/从模式 (Master/slave mode)

- 0: 不执行任何操作
- 1: 当前定时器的触发输入事件 (TRGI) 的动作被推迟, 以使当前定时器与其从定时器实现完美同步 (通过 TRGO)。此设置适用于由单个外部事件对多个定时器进行同步的情况。

位 6:4 **TS[2:0]:** 触发选择 (Trigger selection)

此位域可选择将要用于同步计数器的触发输入。

- 000: 内部触发 0 (ITR0)
- 001: 内部触发 1 (ITR1)
- 010: 内部触发 2 (ITR2)
- 011: 内部触发 3 (ITR3)
- 100: TI1 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)

有关各定时器 ITRx 含义的详细信息, 请参见第 370 页的表 72: TIMx 内部触发连接。

注意: 这些位只能在未使用的情况下 (例如, SMS=000 时) 进行更改, 以避免转换时出现错误的边沿检测。

位 3 保留, 必须保持复位值。

位 2:0 **SMS:** 从模式选择 (Slave mode selection)

选择外部信号时, 触发信号 (TRGI) 的有效边沿与外部输入上所选的极性相关 (请参见输入控制寄存器和控制寄存器说明)。

- 000: 禁止从模式——如果 CEN = “1”, 预分频器时钟直接由内部时钟提供。
- 001: 编码器模式 1——计数器根据 TI1FP1 电平在 TI2FP2 边沿递增/递减计数。
- 010: 编码器模式 2——计数器根据 TI2FP2 电平在 TI1FP1 边沿递增/递减计数。
- 011: 编码器模式 3——计数器在 TI1FP1 和 TI2FP2 的边沿计数, 计数的方向取决于另外一个信号的电平。
- 100: 复位模式——在出现所选触发输入 (TRGI) 上升沿时, 重新初始化计数器并生成一个寄存器更新事件。
- 101: 门控模式——触发输入 (TRGI) 为高电平时使能计数器时钟。只要触发输入变为低电平, 计数器立即停止计数 (但不复位)。计数器的启动和停止都是受控的。
- 110: 触发模式——触发信号 TRGI 出现上升沿时启动计数器 (但不复位)。只控制计数器的启动。
- 111: 外部时钟模式 1——由所选触发信号 (TRGI) 的上升沿提供计数器时钟。

注意: 如果将 TI1F_ED 选作触发输入 (TS=“100”), 则不得使用门控模式。实际上, TI1F 每次转换时, TI1F_ED 都输出 1 个脉冲, 而门控模式检查的是触发信号的电平。

表 72. TIMx 内部触发连接

从 TIM	ITR0 (TS = 000)	ITR1 (TS = 001)	ITR2 (TS = 010)	ITR3 (TS = 011)
TIM1	TIM5	TIM2	TIM3	TIM4
TIM8	TIM1	TIM2	TIM4	TIM5

14.4.4 TIM1 和 TIM8 DMA/中断使能寄存器 (TIMx_DIER)

TIM1&TIM8 DMA/interrupt enable register

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
Res.		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	rw

位 15 保留，必须保持复位值。

位 14 **TDE**: 触发 DMA 请求使能 (Trigger DMA request enable)

- 0: 禁止 DMA 请求
- 1: 使能 DMA 请求

位 13 **COMDE**: COM DMA 请求使能 (COM DMA request enable)

- 0: 禁止 COM DMA 请求
- 1: 使能 COM DMA 请求

位 12 **CC4DE**: 捕获/比较 4 DMA 请求使能 (Capture/Compare 4 DMA request enable)

- 0: 禁止 CC4 DMA 请求
- 1: 使能 CC4 DMA 请求

位 11 **CC3DE**: 捕获/比较 3 DMA 请求使能 (Capture/Compare 3 DMA request enable)

- 0: 禁止 CC3 DMA 请求
- 1: 使能 CC3 DMA 请求

位 10 **CC2DE**: 捕获/比较 2 DMA 请求使能 (Capture/Compare 2 DMA request enable)

- 0: 禁止 CC2 DMA 请求
- 1: 使能 CC2 DMA 请求

位 9 **CC1DE**: 捕获/比较 1 DMA 请求使能 (Capture/Compare 1 DMA request enable)

- 0: 禁止 CC1 DMA 请求
- 1: 使能 CC1 DMA 请求

位 8 **UDE**: 更新 DMA 请求使能 (Update DMA request enable)

- 0: 禁止更新 DMA 请求
- 1: 使能更新 DMA 请求

位 7 **BIE**: 断路中断使能 (Break interrupt enable)

- 0: 禁止断路中断
- 1: 使能断路中断

位 6 **TIE**: 触发信号 (TGRI) 中断使能 (Trigger interrupt enable)

- 0: 禁止触发信号 (TGRI) 中断
- 1: 使能触发信号 (TGRI) 中断

位 5 **COMIE**: COM 中断使能 (COM interrupt enable)

- 0: 禁止 COM 中断
- 1: 使能 COM 中断

位 4 **CC4IE**: 捕获/比较 4 中断使能 (Capture/Compare 4 interrupt enable)

- 0: 禁止 CC4 中断
- 1: 使能 CC4 中断

位 3 **CC3IE**: 捕获/比较 3 中断使能 (Capture/Compare 3 interrupt enable)

- 0: 禁止 CC3 中断
- 1: 使能 CC3 中断

位 2 **CC2IE**: 捕获/比较 2 中断使能 (Capture/Compare 2 interrupt enable)

- 0: 禁止 CC2 中断
- 1: 使能 CC2 中断

位 1 **CC1IE**: 捕获/比较 1 中断使能 (Capture/Compare 1 interrupt enable)

- 0: 禁止 CC1 中断
- 1: 使能 CC1 中断

位 0 **UIE**: 更新中断使能 (Update interrupt enable)

- 0: 禁止更新中断
- 1: 使能更新中断

14.4.5 TIM1 和 TIM8 状态寄存器 (TIMx_SR)

TIM1&TIM8 status register

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
Reserved	CC4OF	CC3OF	CC2OF	CC1OF	Res.	BIF	TIF	COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UIF			
	rc_w0	rc_w0	rc_w0	rc_w0	Res.	rc_w0										

位 15:13 保留, 必须保持复位值。

位 12 **CC4OF**: 捕获/比较 4 重复捕获标志 (Capture/Compare 4 overcapture flag)

请参见 CC1OF 说明

位 11 **CC3OF**: 捕获/比较 3 重复捕获标志 (Capture/Compare 3 overcapture flag)

请参见 CC1OF 说明

位 10 **CC2OF**: 捕获/比较 2 重复捕获标志 (Capture/Compare 2 overcapture flag)

请参见 CC1OF 说明

位 9 **CC1OF**: 捕获/比较 1 重复捕获标志 (Capture/Compare 1 overcapture flag)

仅当将相应通道配置为输入捕获模式时, 此标志位才会由硬件置 1。通过软件写入“0”可将该位清零。

0: 未检测到重复捕获。

1: TIMx_CCR1 寄存器中已捕获到计数器值且 CC1IF 标志已置 1。

位 8 保留, 必须保持复位值。

位 7 **BIF**: 断路中断标志 (Break interrupt flag)

只要断路输入变为有效状态, 此标志便由硬件置 1。断路输入无效后可通过软件对其进行清零。

0: 未发生断路事件。

1: 在断路输入上检测到有效电平。

位 6 **TIF**: 触发中断标志 (Trigger interrupt flag)

在除门控模式以外的所有模式下，当使能从模式控制器后在 TRGI 输入上检测到有效边沿时，该标志将由硬件置 1。选择门控模式时，该标志将在计数器启动或停止时置 1。但需要通过软件清零。

0: 未发生触发事件。

1: 触发中断挂起。

位 5 **COMIF**: COM 中断标志 (COM interrupt flag)

此标志在发生 COM 事件时（捕获/比较控制位 CCxE、CCxNE 和 OCxM 已更新时）由硬件置 1。但需要通过软件清零。

0: 未发生 COM 事件。

1: COM 中断挂起。

位 4 **CC4IF**: 捕获/比较 4 中断标志 (Capture/Compare 4 interrupt flag)

请参见 CC1IF 说明

位 3 **CC3IF**: 捕获/比较 3 中断标志 (Capture/Compare 3 interrupt flag)

请参见 CC1IF 说明

位 2 **CC2IF**: 捕获/比较 2 中断标志 (Capture/Compare 2 interrupt flag)

请参见 CC1IF 说明

位 1 **CC1IF**: 捕获/比较 1 中断标志 (Capture/Compare 1 interrupt flag)

如果通道 CC1 配置为输出：

当计数器与比较值匹配时，此标志由硬件置 1，中心对齐模式下除外（请参见 TIMx_CR1 寄存器中的 CMS 位说明）。但需要通过软件清零。

0: 不匹配。

1: TIMx_CNT 计数器的值与 TIMx_CCR1 寄存器的值匹配。当 TIMx_CCR1 的值大于 TIMx_ARR 的值时，CC1IF 位将在计数器发生上溢（递增计数模式和增减计数模式下）或下溢（递减计数模式下）时变为高电平。

如果通道 CC1 配置为输入：

此位将在发生捕获事件时由硬件置 1。通过软件或读取 TIMx_CCR1 寄存器将该位清零。

0: 未发生输入捕获事件

1: TIMx_CCR1 寄存器中已捕获到计数器值 (IC1 上已检测到与所选极性匹配的边沿)

位 0 **UIF**: 更新中断标志 (Update interrupt flag)

该位在发生更新事件时通过硬件置 1。但需要通过软件清零。

0: 未发生更新。

1: 更新中断挂起。该位在以下情况下更新寄存器时由硬件置 1:

- TIMx_CR1 寄存器中的 UDIS=0，并且重复计数器值上溢或下溢时（重复计数器 = 0 时更新）。

- TIMx_CR1 寄存器中的 URS = 0 且 UDIS = 0，并且由软件使用 TIMx_EGR 寄存器中的 UG 位重新初始化 CNT 时。

- TIMx_CR1 寄存器中的 URS=0 且 UDIS=0，并且 CNT 由触发事件重新初始化时（请参见第 14.4.3 节：TIM1 和 TIM8 从模式控制寄存器 (TIMx_SMCR)）。

14.4.6 TIM1 和 TIM8 事件生成寄存器 (TIMx_EGR)

TIM1&TIM8 event generation register

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
								BG	TG	COMG	CC4G	CC3G	CC2G	CC1G	UG
Reserved								w	w	w	w	w	w	w	w

位 15:8 保留，必须保持复位值。

位 7 **BG**: 断路生成 (Break generation)

此位由软件置 1 以生成事件，并由硬件自动清零。

0: 不执行任何操作

1: 生成断路事件。MOE 位清零且 BIF 标志置 1。使能后可发生相关中断或 DMA 传输事件。

位 6 **TG**: 生成触发信号 (Trigger generation)

此位由软件置 1 以生成事件，并由硬件自动清零。

0: 不执行任何操作

1: TIMx_SR 寄存器中的 TIF 标志置 1。使能后可发生相关中断或 DMA 传输事件。

位 5 **COMG**: 捕获/比较控制更新生成 (Capture/Compare control update generation)

该位可通过软件置 1，并由硬件自动清零

0: 不执行任何操作

1: CCPC 位置 1 时，可更新 CCxE、CCxNE 和 OCxM 位

注意：此位仅对具有互补输出的通道有效。

位 4 **CC4G**: 捕获/比较 4 生成 (Capture/Compare 4 generation)

请参见 CC1G 说明

位 3 **CC3G**: 捕获/比较 3 生成 (Capture/Compare 3 generation)

请参见 CC1G 说明

位 2 **CC2G**: 捕获/比较 2 生成 (Capture/Compare 2 generation)

请参见 CC1G 说明

位 1 **CC1G**: 捕获/比较 1 生成 (Capture/Compare 1 generation)

此位由软件置 1 以生成事件，并由硬件自动清零。

0: 不执行任何操作

1: 通道 1 上生成捕获/比较事件：

如果通道 CC1 配置为输出：

使能时，CC1IF 标志置 1 并发送相应的中断或 DMA 请求。

如果通道 CC1 配置为输入：

TIMx_CCR1 寄存器中将捕获到计数器当前值。使能时，CC1IF 标志置 1 并发送相应的中断或 DMA 请求。如果 CC1IF 标志已为高电平，CC1OF 标志将置 1。

位 0 **UG**: 更新生成 (Update generation)

该位可通过软件置 1，并由硬件自动清零。

0: 不执行任何操作

1: 重新初始化计数器并生成一个寄存器更新事件。请注意，预分频器计数器也将清零（但预分频比不受影响）。如果选择中心对齐模式或 DIR=0（递增计数），计数器将清零；如果 DIR=1（递减计数），计数器将使用自动重载值 (TIMx_ARR)。

14.4.7 TIM1 和 TIM8 捕获/比较模式寄存器 1 (TIMx_CCMR1)

TIM1&TIM8 capture/compare mode register 1

偏移地址: 0x18

复位值: 0x0000

这些通道可用于输入（捕获模式）或输出（比较模式）模式。通道方向通过配置相应的 CC_xS 位进行定义。此寄存器的所有其它位在输入模式和输出模式下的功能均不同。对于任一给定位，OC_{xx} 用于说明通道配置为输出时该位对应的功能，IC_{xx} 则用于说明通道配置为输入时该位对应的功能。因此，必须注意同一个位在输入阶段和输出阶段具有不同的含义。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
OC2 CE	OC2M[2:0]			OC2 PE	OC2 FE	CC2S[1:0]	OC1 CE	OC1M[2:0]			OC1 PE	OC1 FE	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	

输出比较模式:

位 15 **OC2CE**: 输出比较 2 清零使能 (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)

此位域定义通道方向（输入/输出）以及所使用的输入。

00: CC2 通道配置为输出

01: CC2 通道配置为输入，IC2 映射到 TI2 上

10: CC2 通道配置为输入，IC2 映射到 TI1 上

11: CC2 通道配置为输入，IC2 映射到 TRC 上。此模式仅在通过 TS 位 (TIMx_SMCR 寄存器) 选择内部触发输入时有效

注意: 仅当通道关闭时 (TIMx_CCER 中的 CC2E = "0")，才可向 CC2S 位写入数据。

位 7 **OC1CE**: 输出比较 1 清零使能 (Output Compare 1 clear enable)

OC1CE: 输出比较 1 清零使能 (Output Compare 1 Clear Enable)

0: OC1Ref 不受 ETRF 输入影响

1: ETRF 输入上检测到高电平时，OC1Ref 立即清零。

位 6:4 OC1M: 输出比较 1 模式 (Output Compare 1 mode)

这些位定义提供 OC1 和 OC1N 的输出参考信号 OC1REF 的行为。OC1REF 为高电平有效，而 OC1 和 OC1N 的有效电平则取决于 CC1P 位和 CC1NP 位。

000: 冻结——输出比较寄存器 TIMx_CCR1 与计数器 TIMx_CNT 进行比较不会对输出造成任何影响。（该模式用于生成时基）。

001: 将通道 1 设置为匹配时输出有效电平。当计数器 TIMx_CNT 与捕获/比较寄存器 1 (TIMx_CCR1) 匹配时，OC1REF 信号强制变为高电平。

010: 将通道 1 设置为匹配时输出无效电平。当计数器 TIMx_CNT 与捕获/比较寄存器 1 (TIMx_CCR1) 匹配时，OC1REF 信号强制变为低电平。

011: 翻转——TIMx_CNT=TIMx_CCR1 时，OC1REF 发生翻转。

100: 强制变为无效电平——OC1REF 强制变为低电平。

101: 强制变为有效电平——OC1REF 强制变为高电平。

110: PWM 模式 1——在递增计数模式下，只要 TIMx_CNT<TIMx_CCR1，通道 1 便为有效状态，否则为无效状态。在递减计数模式下，只要 TIMx_CNT>TIMx_CCR1，通道 1 便为无效状态 (OC1REF=“0”），否则为有效状态 (OC1REF=“1”）。

111: PWM 模式 2——在递增计数模式下，只要 TIMx_CNT<TIMx_CCR1，通道 1 便为无效状态，否则为有效状态。在递减计数模式下，只要 TIMx_CNT>TIMx_CCR1，通道 1 便为有效状态，否则为无效状态。

注意: 1: 只要编程了 LOCK (TIMx_BDTR 寄存器中的 LOCK 位) 级别 3 且 CC1S=“00” (通道配置为输出)，这些位即无法修改。

2: 在 PWM 模式 1 或 PWM 模式 2 下，仅当比较结果发生改变或输出比较模式由“冻结”模式切换到“PWM”模式时，OCREF 电平才会发生更改。

3: 此位域将在具有互补输出的通道上进行预装载。如果 TIMx_CR2 寄存器中的 CCPC 位置 1，则仅当生成 COM 事件时，OC1M 有效位才会从预装载位获取新值。

位 3 OC1PE: 输出比较 1 预装载使能 (Output Compare 1 preload enable)

0: 禁止与 TIMx_CCR1 相关的预装载寄存器。可随时向 TIMx_CCR1 写入数据，写入后将立即使用新值。

1: 使能与 TIMx_CCR1 相关的预装载寄存器。可读/写访问预装载寄存器。TIMx_CCR1 预装载值在每次生成更新事件时都会装载到活动寄存器中。

注意: 1: 只要编程了 LOCK (TIMx_BDTR 寄存器中的 LOCK 位) 级别 3 且 CC1S=“00” (通道配置为输出)，这些位即无法修改。

2: 只有单脉冲模式下才可在未验证预装载寄存器的情况下使用 PWM 模式 (TIMx_CR1 寄存器中的 OPM 位置 1)。其它情况下则无法保证该行为。

位 2 OC1FE: 输出比较 1 快速使能 (Output Compare 1 fast enable)

此位用于加快触发输入事件对 CC 输出的影响。

0: 即使触发开启，CC1 也将根据计数器和 CCR1 值正常工作。触发输入出现边沿时，激活 CC1 输出的最短延迟时间为 5 个时钟周期。

1: 触发输入上出现有效边沿相当于 CC1 输出上的比较匹配。随后，无论比较结果如何，OC 都设置为比较电平。采样触发输入和激活 CC1 输出的延迟时间缩短为 3 个时钟周期。仅当通道配置为 PWM1 或 PWM2 模式时，OCFE 才会起作用。

位 1:0 CC1S: 捕获/比较 1 选择 (Capture/Compare 1 selection)

此位域定义通道方向（输入/输出）以及所使用的输入。

00: CC1 通道配置为输出

01: CC1 通道配置为输入，IC1 映射到 TI1 上

10: CC1 通道配置为输入，IC1 映射到 TI2 上

11: CC1 通道配置为输入，IC1 映射到 TRC 上。此模式仅在通过 TS 位 (TIMx_SMCR 寄存器) 选择内部触发输入时有效

注意: 仅当通道关闭时 (TIMx_CCER 中的 CC1E = “0”），才可向 CC1S 位写入数据。

输入捕获模式

位 15:12 **IC2F**: 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 **CC2S**: 捕获/比较 2 选择 (Capture/Compare 2 selection)

此位域定义通道方向（输入/输出）以及所使用的输入。

00: CC2 通道配置为输出

01: CC2 通道配置为输入, IC2 映射到 TI2 上

10: CC2 通道配置为输入, IC2 映射到 TI1 上

11: CC2 通道配置为输入, IC2 映射到 TRC 上。此模式仅在通过 TS 位 (TIMx_SMCR 寄存器) 选择内部触发输入时有效

注意: 仅当通道关闭时 (TIMx_CCER 中的 CC2E = “0”) , 才可向 CC2S 位写入数据。

位 7:4 **IC1F[3:0]**: 输入捕获 1 滤波器 (Input capture 1 filter)

此位域可定义 TI1 输入的采样频率和适用于 TI1 的数字滤波器带宽。数字滤波器由事件计数器组成，每 N 个事件才视为一个有效边沿：

0000: 无滤波器, 按 f_{DTS} 频率进行采样

0001: $f_{SAMPLING}=f_{CK_INT}$, N=2

0010: $f_{SAMPLING}=f_{CK_INT}$, N=4

0011: $f_{SAMPLING}=f_{CK_INT}$, N=8

0100: $f_{SAMPLING}=f_{DTS}/2$, N=6

0101: $f_{SAMPLING}=f_{DTS}/2$, N=8

0110: $f_{SAMPLING}=f_{DTS}/4$, N=6

0111: $f_{SAMPLING}=f_{DTS}/4$, N=8

1000: $f_{SAMPLING}=f_{DTS}/8$, N=6

1001: $f_{SAMPLING}=f_{DTS}/8$, N=8

1010: $f_{SAMPLING}=f_{DTS}/16$, N=5

1011: $f_{SAMPLING}=f_{DTS}/16$, N=6

1100: $f_{SAMPLING}=f_{DTS}/16$, N=8

1101: $f_{SAMPLING}=f_{DTS}/32$, N=5

1110: $f_{SAMPLING}=f_{DTS}/32$, N=6

1111: $f_{SAMPLING}=f_{DTS}/32$, N=8

位 3:2 **IC1PSC**: 输入捕获 1 预分频器 (Input capture 1 prescaler)

此位域定义 CC1 输入 (IC1) 的预分频比。

只要 CC1E=“0” (TIMx_CCER 寄存器) , 预分频器便立即复位。

00: 无预分频器, 捕获输入上每检测到一个边沿便执行捕获

01: 每发生 2 个事件便执行一次捕获

10: 每发生 4 个事件便执行一次捕获

11: 每发生 8 个事件便执行一次捕获

位 1:0 **CC1S**: 捕获/比较 1 选择 (Capture/Compare 1 Selection)

此位域定义通道方向（输入/输出）以及所使用的输入。

00: CC1 通道配置为输出

01: CC1 通道配置为输入, IC1 映射到 TI1 上

10: CC1 通道配置为输入, IC1 映射到 TI2 上

11: CC1 通道配置为输入, IC1 映射到 TRC 上。此模式仅在通过 TS 位 (TIMx_SMCR 寄存器) 选择内部触发输入时有效

注意: 仅当通道关闭时 (TIMx_CCER 中的 CC1E = “0”) , 才可向 CC1S 位写入数据。

14.4.8 TIM1 和 TIM8 捕获/比较模式寄存器 2 (TIMx_CCMR2)

TIM1&TIM8 capture/compare mode register 2

偏移地址: 0x1C

复位值: 0x0000

请参见上述 CCMR1 寄存器说明。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC4 CE	OC4M[2:0]			OC4 PE	OC4 FE	CC4S[1:0]	OC3 CE.	OC3M[2:0]			OC3 PE	OC3 FE	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 清零使能 (Output compare 4 clear enable)

位 14:12 **OC4M**: 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**: 捕获/比较 4 选择 (Capture/Compare 4 selection)

此位域定义通道方向 (输入/输出) 以及所使用的输入。

00: CC4 通道配置为输出

01: CC4 通道配置为输入, IC4 映射到 TI4 上

10: CC4 通道配置为输入, IC4 映射到 TI3 上

11: CC4 通道配置为输入, IC4 映射到 TRC 上。此模式仅在通过 TS 位 (TIMx_SMCR 寄存器) 选择内部触发输入时有效

注意: 仅当通道关闭时 (TIMx_CCER 中的 CC4E = “0”), 才可向 CC4S 位写入数据。

位 7 **OC3CE**: 输出比较 3 清零使能 (Output compare 3 clear enable)

位 6:4 **OC3M**: 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**: 捕获/比较 3 选择 (Capture/Compare 3 selection)

此位域定义通道方向 (输入/输出) 以及所使用的输入。

00: CC3 通道配置为输出

01: CC3 通道配置为输入, IC3 映射到 TI3 上

10: CC3 通道配置为输入, IC3 映射到 TI4 上

11: CC3 通道配置为输入, IC3 映射到 TRC 上。此模式仅在通过 TS 位 (TIMx_SMCR 寄存器) 选择内部触发输入时有效

注意: 仅当通道关闭时 (TIMx_CCER 中的 CC3E = “0”), 才可向 CC3S 位写入数据。

输入捕获模式

位 15:12 **IC4F**: 输入捕获 4 滤波器 (Input capture 4 filter)

位 11:10 **IC4PSC**: 输入捕获 4 预分频器 (Input capture 4 prescaler)

位 9:8 **CC4S**: 捕获/比较 4 选择 (Capture/Compare 4 selection)

此位域定义通道方向（输入/输出）以及所使用的输入。

00: CC4 通道配置为输出

01: CC4 通道配置为输入, IC4 映射到 TI4 上

10: CC4 通道配置为输入, IC4 映射到 TI3 上

11: CC4 通道配置为输入, IC4 映射到 TRC 上。此模式仅在通过 TS 位 (TIMx_SMCR 寄存器) 选择内部触发输入时有效

注意: 仅当通道关闭时 (TIMx_CCER 中的 CC4E = "0")，才可向 CC4S 位写入数据。

位 7:4 **IC3F**: 输入捕获 3 滤波器 (Input capture 3 filter)

位 3:2 **IC3PSC**: 输入捕获 3 预分频器 (Input capture 3 prescaler)

位 1:0 **CC3S**: 捕获/比较 3 选择 (Capture/compare 3 selection)

此位域定义通道方向（输入/输出）以及所使用的输入。

00: CC3 通道配置为输出

01: CC3 通道配置为输入, IC3 映射到 TI3 上

10: CC3 通道配置为输入, IC3 映射到 TI4 上

11: CC3 通道配置为输入, IC3 映射到 TRC 上。此模式仅在通过 TS 位 (TIMx_SMCR 寄存器) 选择内部触发输入时有效

注意: 仅当通道关闭时 (TIMx_CCER 中的 CC3E = "0")，才可向 CC3S 位写入数据。

14.4.9 TIM1 和 TIM8 捕获/比较使能寄存器 (TIMx_CCER)

TIM1&TIM8 capture/compare enable register

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
Reserved			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

位 15:14 保留, 必须保持复位值。

位 13 **CC4P**: 捕获/比较 4 输出极性 (Capture/Compare 4 output polarity)

请参见 CC1P 说明

位 12 **CC4E**: 捕获/比较 4 输出使能 (Capture/Compare 4 output enable)

请参见 CC1E 说明

位 11 **CC3NP**: 捕获/比较 3 互补输出极性 (Capture/Compare 3 complementary output polarity)

请参见 CC1NP 说明

位 10 **CC3NE**: 捕获/比较 3 互补输出使能 (Capture/Compare 3 complementary output enable)

请参见 CC1NE 说明

位 9 **CC3P**: 捕获/比较 3 输出极性 (Capture/Compare 3 output polarity)

请参见 CC1P 说明

位 8 **CC3E**: 捕获/比较 3 输出使能 (Capture/Compare 3 output enable)

请参见 CC1E 说明

位 7 **CC2NP**: 捕获/比较 2 互补输出极性 (Capture/Compare 2 complementary output polarity)

请参见 CC1NP 说明

位 6 **CC2NE**: 捕获/比较 2 互补输出使能 (Capture/Compare 2 complementary output enable)

请参见 CC1NE 说明

位 5 **CC2P**: 捕获/比较 2 输出极性 (Capture/Compare 2 output polarity)

请参见 CC1P 说明

位 4 **CC2E**: 捕获/比较 2 输出使能 (Capture/Compare 2 output enable)

请参见 CC1E 说明

位 3 **CC1NP**: 捕获/比较 1 互补输出极性 (Capture/Compare 1 complementary output polarity)

CC1 通道配置为输出:

0: OC1N 高电平有效。

1: OC1N 低电平有效。

CC1 通道配置为输入:

此位与 CC1P 配合使用, 用以定义 TI1FP1 和 TI2FP1 的极性。请参见 CC1P 说明。

注意: 此位将在具有互补输出的通道上进行预装载。如果 TIMx_CR2 寄存器中的 CCPC 位置 1, 则仅当生成换向事件时, CC1NP 有效位才会从预装载位获取新值。

注意: 只要编程了 LOCK (TIMx_BDTR 寄存器中的 LOCK 位) 级别 2 或 3 且 CC1S=“00” (通道配置为输出), 此位立即变为不可写状态。

位 2 **CC1NE**: 捕获 / 比较 1 互补输出使能 (Capture/Compare 1 complementary output enable)

0: 关闭 ——OC1N 未激活。OC1N 电平是 MOE、OSSI、OSSR、OIS1、OIS1N 和 CC1E 位的函数。

1: 开启 ——在相应输出引脚上输出 OC1N 信号, 具体取决于 MOE、OSSI、OSSR、OIS1、OIS1N 和 CC1E 位。

注意: 此位将在具有互补输出的通道上进行预装载。如果 TIMx_CR2 寄存器中的 CCPC 位置 1, 则仅当生成换向事件时, CC1NE 有效位才会从预装载位获取新值。

位 1 **CC1P**: 捕获/比较 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:

CC1NP/CC1P 位可针对触发或捕获操作选择 TI1FP1 和 TI2FP1 的有效极性。

00: 非反相/上升沿触发

电路对 TIxFP1 上升沿敏感（在复位模式、外部时钟模式或触发模式下执行捕获或触发操作），TIxFP1 未反相（在门控模式或编码器模式下执行触发操作）。

01: 反相/下降沿触发

电路对 TIxFP1 下降沿敏感（在复位模式、外部时钟模式或触发模式下执行捕获或触发操作），TIxFP1 反相（在门控模式或编码器模式下执行触发操作）。

10: 保留，不使用此配置。

11: 未反相/边沿触发。

电路对 TIxFP1 上升沿和下降沿都敏感（在复位模式、外部时钟模式或触发模式下执行捕获或触发操作），TIxFP1 未反相（在门控模式下执行触发操作）。编码器模式下不得使用此配置。

注意: 此位将在具有互补输出的通道上进行预装载。如果 TIMx_CR2 寄存器中的 CCPC 位置 1，则仅当生成换向事件时，CC1P 有效位才会从预装载位获取新值。

注意: 只要编程了 LOCK (TIMx_BDTR 寄存器中的 LOCK 位) 级别 2 或 3，此位立即变为不可写状态。

位 0 **CC1E**: 捕获 / 比较 1 输出使能 (Capture/Compare 1 output enable)

CC1 通道配置为输出:

0: 关闭——OC1 未激活。OC1 电平是 MOE、OSSI、OSSR、OIS1、OIS1N 和 CC1NE 位的函数。

1: 开启——OC1 信号是相应输出引脚上的输出，具体取决于 MOE、OSSI、OSSR、OIS1、OIS1N 和 CC1NE 位。

CC1 通道配置为输入:

此位决定了是否可以实际将计数器值捕获到输入捕获 / 比较寄存器 1 (TIMx_CCR1) 中。

0: 禁止捕获。

1: 使能捕获。

注意: 此位将在具有互补输出的通道上进行预装载。如果 TIMx_CR2 寄存器中的 CCPC 位置 1，则仅当生成换向事件时，CC1E 有效位才会从预装载位获取新值。

表 73. 具有断路功能的互补通道 OCx 和 OCxN 的输出控制位

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	禁止输出 (不由定时器驱动) OCx=0、OCx_EN=0	禁止输出 (不由定时器驱动) OCxN=0、OCxN_EN=0
		0	0	1	禁止输出 (不由定时器驱动) OCx=0、OCx_EN=0	OCxREF + 极性 OCxN=OCxREF 异或 CCxNP、OCxN_EN=1
		0	1	0	OCxREF + 极性 OCx=OCxREF 异或 CCxP、 OCx_EN=1	禁止输出 (不由定时器驱动) OCxN=0、OCxN_EN=0
		0	1	1	OCREF + 极性 + 死区 OCx_EN=1	OCREF 互补项 (而非 OCREF) + 极性 + 死区 OCxN_EN=1
		1	0	0	禁止输出 (不由定时器驱动) OCx=CCxP、OCx_EN=0	禁止输出 (不由定时器驱动) OCxN=CCxNP、OCxN_EN=0
		1	0	1	关闭状态 (输出使能为无效状态) OCx=CCxP、OCx_EN=1	OCxREF + 极性 OCxN=OCxREF 异或 CCxNP、 OCxN_EN=1
		1	1	0	OCxREF + 极性 OCx=OCxREF 异或 CCxP、 OCx_EN=1	关闭状态 (输出使能为无效状态) OCxN=CCxNP、OCxN_EN=1
		1	1	1	OCREF + 极性 + 死区 OCx_EN=1	OCREF 互补项 (而非 OCREF) + 极性 + 死区 OCxN_EN=1
0	X	0	0	0	禁止输出 (不由定时器驱动) OCx=CCxP、OCx_EN=0	禁止输出 (不由定时器驱动) OCxN=CCxNP、OCxN_EN=0
		0	0	1	禁止输出 (不由定时器驱动)	
		0	1	0	异步: OCx=CCxP、OCx_EN=0、OCxN=CCxNP、OCxN_EN=0	
		0	1	1	如果存在时钟: 在死区后 OCx=OISx 且 OCxN=OISxN, 从而假定 OISx 和 OISxN 在有效状态下与 OCx 和 OCxN 不对应。	
		1	0	0	禁止输出 (不由定时器驱动) OCx=CCxP、OCx_EN=0	禁止输出 (不由定时器驱动) OCxN=CCxNP、OCxN_EN=0
		1	0	1	关闭状态 (输出使能为无效状态)	
		1	1	0	异步: OCx=CCxP、OCx_EN=1、OCxN=CCxNP、OCxN_EN=1	
		1	1	1	如果存在时钟: 在死区后 OCx=OISx 且 OCxN=OISxN, 从而假定 OISx 和 OISxN 在有效状态下与 OCx 和 OCxN 不对应	

1. 如果一个通道的两个输出均未使用 (CCxE = CCxNE = 0), 则 OISx、OISxN、CCxP 和 CCxNP 位必须保持清零状态。

注意: 与互补通道 OCx 和 OCxN 相连的外部 I/O 引脚的状态取决于通道 OCx 和 OCxN 的状态以及 GPIO 寄存器。

14.4.10 TIM1 和 TIM8 计数器 (TIMx_CNT)

TIM1&TIM8 counter

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]**: 计数器值 (Counter value)

14.4.11 TIM1 和 TIM8 预分频器 (TIMx_PSC)

TIM1&TIM8 prescaler

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]**: 预分频器值 (Prescaler value)

计数器时钟频率 (CK_{CNT}) 等于 $f_{CK_{PSC}} / (PSC[15:0] + 1)$ 。

PSC 包含每次发生更新事件（包括计数器通过 TIMx_EGR 寄存器中的 UG 位清零时，或在配置为“复位模式”时通过触发控制器清零时）时要装载到活动预分频器寄存器的值。

14.4.12 TIM1 和 TIM8 自动重载寄存器 (TIMx_ARR)

TIM1&TIM8 auto-reload register

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]**: 自动重载值 (Auto-reload value)

ARR 为要装载到实际自动重载寄存器的值。

有关 ARR 更新和行为的详细信息，请参见 [第 331 页的第 14.3.1 节：时基单元](#)。

当自动重载值为空时，计数器不工作。

14.4.13 TIM1 和 TIM8 重复计数器寄存器 (TIMx_RCR)

TIM1&TIM8 repetition counter register

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
Reserved								REP[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

位 15:8 保留，必须保持复位值。

位 7:0 REP[7:0]: 重复计数器值 (Repetition counter value)

使能预装载寄存器时，用户可通过这些位设置比较寄存器的更新频率（即，从预装载寄存器向活动寄存器周期性传输数据）；使能更新中断时，也可设置更新中断的生成速率。

与 REP_CNT 相关的减计数器每次计数到 0 时，都将生成一个更新事件并且计数器从 REP 值重新开始计数。由于只有生成重复更新事件 U_RC 时，REP_CNT 才会重载 REP 值，因此在生成下一重复更新事件之前，无论向 TIMx_RCR 寄存器写入何值都无影响。

这意味着 PWM 模式下 (REP+1) 相当于：

- 边沿对齐模式下的 PWM 周期数
- 中心对齐模式下的 PWM 半周期数

14.4.14 TIM1 和 TIM8 捕获/比较寄存器 1 (TIMx_CCR1)

TIM1&TIM8 capture/compare register 1

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 1 值 (Capture/Compare 1 value)

如果通道 CC1 配置为输出：

CCR1 是捕获/比较寄存器 1 的预装载值。

如果没有通过 TIMx_CCMR 寄存器中的 OC1PE 位来使能预装载功能，写入的数值会被直接传输至当前寄存器中。否则只在发生更新事件时生效（拷贝到实际起作用的捕获/比较寄存器 1）。实际捕获/比较寄存器中包含要与计数器 TIMx_CNT 进行比较并在 OC1 输出上发出信号的值。

如果通道 CC1 配置为输入：

CCR1 为上一个输入捕获 1 事件 (IC1) 发生时的计数器值。

14.4.15 TIM1 和 TIM8 捕获/比较寄存器 2 (TIMx_CCR2)

TIM1&TIM8 capture/compare register 2

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 2 值 (Capture/Compare 2 value)

如果通道 CC2 配置为输出:

CCR2 为要装载到实际捕获/比较 2 寄存器的值 (预装载值)。

如果没有通过 TIMx_CCMR2 寄存器中的 OC2PE 位来使能预装载功能, 写入的数值会被直接传输至当前寄存器中。否则只有发生更新事件时, 预装载值才会复制到活动捕获/比较 2 寄存器中。实际捕获/比较寄存器中包含要与计数器 TIMx_CNT 进行比较并在 OC2 输出上发出信号的值。

如果通道 CC2 配置为输入:

CCR2 为上一个输入捕获 2 事件 (IC2) 发生时的计数器值。

14.4.16 TIM1 和 TIM8 捕获/比较寄存器 3 (TIMx_CCR3)

TIM1&TIM8 capture/compare register 3

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/比较值 (Capture/Compare value)

如果通道 CC3 配置为输出:

CCR3 为要装载到实际捕获/比较 3 寄存器的值 (预装载值)。

如果没有通过 TIMx_CCMR3 寄存器中的 OC3PE 位来使能预装载功能, 写入的数值会被直接传输至当前寄存器中。否则只有发生更新事件时, 预装载值才会复制到活动捕获/比较 3 寄存器中。实际捕获/比较寄存器中包含要与计数器 TIMx_CNT 进行比较并在 OC3 输出上发出信号的值。

如果通道 CC3 配置为输入:

CCR3 为上一个输入捕获 3 事件 (IC3) 发生时的计数器值。

14.4.17 TIM1 和 TIM8 捕获/比较寄存器 4 (TIMx_CCR4)

TIM1&TIM8 capture/compare register 4

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/比较值 (Capture/Compare value)

如果通道 CC4 配置为输出:

CCR4 为要装载到实际捕获/比较 4 寄存器的值 (预装载值)。

如果没有通过 TIMx_CCMR4 寄存器中的 OC4PE 位来使能预装载功能, 写入的数值会被直接传输至当前寄存器中。否则只有发生更新事件时, 预装载值才会复制到活动捕获/比较 4 寄存器中。实际捕获/比较寄存器中包含要与计数器 TIMx_CNT 进行比较并在 OC4 输出上发出信号的值。

如果通道 CC4 配置为输入:

CCR4 为上一个输入捕获 4 事件 (IC4) 发生时的计数器值。

14.4.18 TIM1 和 TIM8 断路和死区寄存器 (TIMx_BDTR)

TIM1&TIM8 break and dead-time register

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

注意: 由于可以根据 LOCK 配置锁定位 AOE、BKP、BKE、OSSI、OSSR 和 DTG[7:0] 的写操作, 因此必须在第一次对 TIMx_BDTR 寄存器执行写访问时对这些位进行配置。

位 15 MOE: 主输出使能 (Main output enable)

只要断路输入变为有效状态, 此位便由硬件异步清零。此位由软件置 1, 也可根据 AOE 位状态自动置 1。此位仅对配置为输出的通道有效。

0: OC 和 OCN 输出禁止或被强制为空闲状态。

1: 如果 OC 和 OCN 输出的相应使能位 (TIMx_CCER 寄存器中的 CCxE 和 CCxNE 位) 均置 1, 则使能 OC 和 OCN 输出。

有关详细信息, 请参见 OC/OCN 使能说明 (第 379 页的第 14.4.9 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 只能由软件置 1

1: MOE 可由软件置 1, 也可在发生下一更新事件时自动置 1(如果断路输入无效)

注意: 只要编程了 LOCK (TIMx_BDTR 寄存器中的 LOCK 位) 级别 1, 此位即无法修改。

位 13 **BKP:** 断路极性 (Break polarity)

- 0: 断路输入 BRK 为低电平有效
- 1: 断路输入 BRK 为高电平有效

注意: 只要编程了 *LOCK* (*TIMx_BDTR* 寄存器中的 *LOCK* 位) 级别 1, 此位即无法修改。

注意: 对该位执行任何写操作后, 都需要经过 1 个 APB 时钟周期的延迟才生效。

位 12 **BKE:** 断路使能 (Break enable)

- 0: 禁止断路输入 (BRK 和 CSS 时钟故障事件)
- 1: 使能断路输入 (BRK 和 CSS 时钟故障事件)

注意: 编程了 *LOCK* (*TIMx_BDTR* 寄存器中的 *LOCK* 位) 级别 1 后, 此位即无法修改。

注意: 对该位执行任何写操作后, 都需要经过 1 个 APB 时钟周期的延迟才生效。

位 11 **OSSR:** 运行模式下的关闭状态选择 (Off-state selection for Run mode)

此位在 **MOE=1** 时作用于配置为输出模式且具有互补输出的通道。如果定时器中没有互补输出, 则不存在 OSSR。

有关详细信息, 请参见 OC/OCN 使能说明 ([第 379 页的第 14.4.9 节: TIM1 和 TIM8 捕获/比较使能寄存器 \(TIMx_CCER\)](#))。

0: 处于无效状态时, 禁止 OC/OCN 输出 (OC/OCN 使能输出信号=0)。

1: 处于无效状态时, 一旦 CCxE=1 或 CCxNE=1, 便使能 OC/OCN 输出并将其设为无效电平。然后设置 OC/OCN 使能输出信号=1

注意: 编程了 *LOCK* (*TIMx_BDTR* 寄存器中的 *LOCK* 位) 级别 2 后, 此位即无法修改。

位 10 **OSSI:** 空闲模式下的关闭状态选择 (Off-state selection for Idle mode)

此位在 **MOE=0** 时作用于配置为输出的通道。

有关详细信息, 请参见 OC/OCN 使能说明 ([第 379 页的第 14.4.9 节: TIM1 和 TIM8 捕获/比较使能寄存器 \(TIMx_CCER\)](#))。

0: 处于无效状态时, 禁止 OC/OCN 输出 (OC/OCN 使能输出信号=0)。

1: 处于无效状态时, 一旦 CCxE=1 或 CCxNE=1, 便将 OC/OCN 输出首先强制为其空闲电平。然后设置 OC/OCN 使能输出信号=1

注意: 编程了 *LOCK* (*TIMx_BDTR* 寄存器中的 *LOCK* 位) 级别 2 后, 此位即无法修改。

位 9:8 **LOCK[1:0]:** 锁定配置 (Lock configuration)

这些位用于针对软件错误提供写保护。

00: 关闭锁定——不对任何位提供写保护。

01: 锁定级别 1, 此时无法对 *TIMx_BDTR* 寄存器中的 DTG 位、*TIMx_CR2* 寄存器中的 OISx 和 OISxN 位以及 *TIMx_BDTR* 寄存器中的 BKE/BKP/AOE 位执行写操作。

10: 锁定级别 2, 此时无法对锁定级别 1 中适用的各位、CC 极性位 (*TIMx_CCER* 寄存器中的 CCxP/CCxNP 位, 只要通过 CCxS 位将相关通道配置为输出) 以及 OSSR 和 OSSI 位执行写操作。

11: 锁定级别 3, 此时无法对锁定级别 2 中适用的各位、CC 控制位 (*TIMx_CCMRx* 寄存器中的 OCxM 和 OCxPE 位, 只要通过 CCxS 位将相关通道配置为输出) 执行写操作。

注意: 复位后只能对 *LOCK* 位执行一次写操作。对 *TIMx_BDTR* 寄存器执行写操作后其中的内容将冻结, 直到下一次复位。

位 7:0 DTG[7:0]: 配置死区发生器 (Dead-time generator setup)

此位域定义插入到互补输出之间的死区持续时间。DT 与该持续时间相对应。

$DTG[7:5]=0xx \Rightarrow DT=DTG[7:0] \times t_{dtg}$, 其中 $t_{dtg}=t_{DTS}$ 。

$DTG[7:5]=10x \Rightarrow DT=(64+DTG[5:0]) \times t_{dtg}$, 其中 $T_{dtg}=2 \times t_{DTS}$ 。

$DTG[7:5]=110 \Rightarrow DT=(32+DTG[4:0]) \times t_{dtg}$, 其中 $T_{dtg}=8 \times t_{DTS}$ 。

$DTG[7:5]=111 \Rightarrow DT=(32+DTG[4:0]) \times t_{dtg}$, 其中 $T_{dtg}=16 \times t_{DTS}$ 。

示例：如果 $T_{DTS}=125\text{ns}$ (8MHz), 则可能的死区值为：

0 到 15875 ns (步长为 125 ns) ,

16 us 到 31750 ns (步长为 250 ns) ,

32 us 到 63us (步长为 1 us) ,

64 us 到 126 us (步长为 2 us)

注意：只要编程了 *LOCK* (*TIMx_BDTR* 寄存器中的 *LOCK* 位) 级别 1、2 或 3, 此位域即无法修改。

14.4.19 TIM1 和 TIM8 DMA 控制寄存器 (TIMx_DCR)**TIM1&TIM8 DMA control register**

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	
Reserved					DBL[4:0]					Reserved			DBA[4:0]			
					rw	rw	rw	rw	rw				rw	rw	rw	rw

位 15:13 保留，必须保持复位值。

位 12:8 DBL[4:0]: DMA 连续传送长度 (DMA burst length)

该 5 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）。

TIMx_DMAR 地址）

00000: 1 次传输

00001: 2 次传输

00010: 3 次传输

...

10001: 18 次传输

位 7:5 保留，必须保持复位值。

位 4:0 DBA[4:0]: DMA 基址 (DMA base address)

该 5 位向量定义 DMA 传输的基址（通过 *TIMx_DMAR* 地址进行读/写访问时）。DBA 定义为从 *TIMx_CR1* 寄存器地址开始计算的偏移量。

示例：

00000: *TIMx_CR1*,

00001: *TIMx_CR2*,

00010: *TIMx_SMCR*,

...

示例：考虑以下传输：DBL = 7 次传输，DBA = *TIMx_CR1*。这种情况下将向/从自 *TIMx_CR1* 地址开始的 7 个寄存器传输数据。

14.4.20 TIM1 和 TIM8 全传输 DMA 地址 (TIMx_DMAR)

TIM1&TIM8 DMA address for full transfer

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 DMAR 寄存器执行读或写操作将访问位于如下地址的寄存器

(TIMx_CR1 地址) + (DBA + DMA 索引) × 4

其中 TIMx_CR1 地址为控制寄存器 1 的地址, DBA 为 TIMx_DCR 寄存器中配置的 DMA 基址, DMA 索引由 DMA 传输自动控制, 其范围介于 0 到 DBL (TIMx_DCR 寄存器中配置的 DBL) 之间。

DMA 连续传送功能使用方法示例

本例中, 定时器 DMA 连续传送功能用于将 CCRx 寄存器 ($x = 2, 3, 4$) 的内容更新为通过 DMA 传输到 CCRx 寄存器中的多个半字。

具体操作步骤如下:

1. 将相应的 DMA 通道配置如下:
 - DMA 通道外设地址为 DMAR 寄存器地址
 - DMA 通道存储器地址为包含要通过 DMA 传输到 CCRx 寄存器的数据的 RAM 缓冲区地址。
 - 要传输的数据量 = 3 (参见下文注释)。
 - 禁止循环模式。
2. 通过将 DBA 和 DBL 位域配置如下来配置 DCR 寄存器:
DBL = 3 次传输, DBA = 0xE。
3. 使能 TIMx 更新 DMA 请求 (DIER 寄存器中的 UDE 位置 1)。
4. 使能 TIMx
5. 使能 DMA 通道

注意:

本例适用于每个 CCRx 寄存器只更新一次的情况。如果每个 CCRx 寄存器要更新两次, 则要传输的数据量应为 6。下面以包含 data1、data2、data3、data4、data5 和 data6 的 RAM 缓冲区为例。数据将按照如下方式传输到 CCRx 寄存器: 在第一个更新 DMA 请求期间, data1 传输到 CCR2, data2 传输到 CCR3, data3 传输到 CCR4; 在第二个更新 DMA 请求期间, data4 传输到 CCR2, data5 传输到 CCR3, data6 传输到 CCR4。

14.4.21 TIM1 和 TIM8 寄存器映射

TIM1 和 **TIM8** 寄存器可映射为 16 位可寻址寄存器，如下表所述：

表 74. TIM1 和 TIM8 寄存器映射和复位值

表 74. TIM1 和 TIM8 寄存器映射和复位值（续）

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
0x34	TIMx_CCR1	Reserved												CCR1[15:0]																			
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
0x38	TIMx_CCR2	Reserved												CCR2[15:0]																			
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
0x3C	TIMx_CCR3	Reserved												CCR3[15:0]																			
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
0x40	TIMx_CCR4	Reserved												CCR4[15:0]																			
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
0x44	TIMx_BDTR	Reserved												MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK [1:0]	DT[7:0]												
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
0x48	TIMx_DCR	Reserved												DBL[4:0]				Reserved			DBA[4:0]												
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
0x4C	TIMx_DMAR	Reserved												DMAB[15:0]																			
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					

有关寄存器边界地址的信息，请参见第 52 页的表 2。