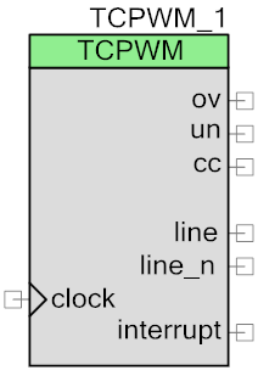
PSoC 4 定时计数器脉冲宽度调制器（TCPWM）

1.0

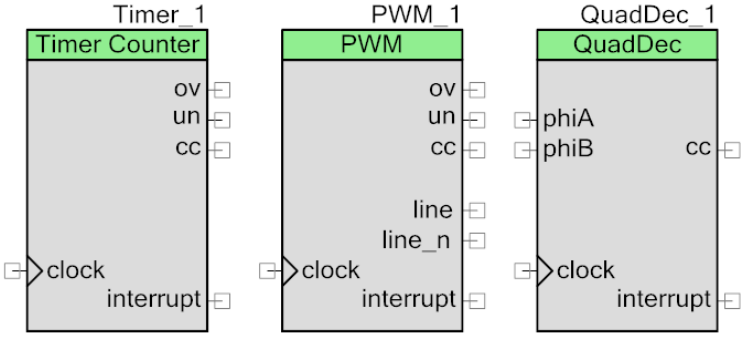
特性



- 16 位固定功能实现
- 定时器/计数器功能模式
- 正交解码器功能模式
- 脉冲宽度调制（PWM）模式
- 带可配置死区时间插入的脉冲宽度调制
- 伪随机脉冲宽度调制
- 实时动态配置

概述

TCPWM 组件是多功能组件，可使用 PSoC 4 TCPWM 模块实现多种核心微控制器功能，包括定时器/计数器、脉冲宽度调制器和正交解码器。在 PSoC Creator 组件对话框中，每项功能都可作为预配置的示意宏，被标记为“TCPWM Mode”（TCPWM 模式）。



该目录中的基本组件被设定为未配置的模式。未配置模式的组件在运行时可通过函数来完成任何模式的操作。

此组件基于硬件结构，用于在各种操作模式之间共享同一硬件。此结构使同一硬件可提供一系列灵活的功能，但几乎不会提高芯片的使用量。可在构建时定义功能以与此硬件支持的重要操作模式之一相匹配。你也可以保持 TCPWM 组件工作在未配置的模式下，而在运行时通过 API 接口函数将此组件配置为特定模式。

- TCPWM 具备 16 位计数器，其支持向上、向下和向上/向下计数模式。所有硬件输入信号上的上升沿、下降沿、双边沿的检测以及电平信号可用于产生计数器事件。有三种路由输出信号可用于说明下溢出、上溢出和计数器/比较匹配事件。
- 此组件具有双缓冲比较/捕获寄存器和周期寄存器，在运行时可通过 API 实现重新配置或寄存器切换。可通过任何 HW 输入信号衍生开始、重新加载、计数和捕获事件，还可由软件生成这些事件。
- 可将脉冲宽度调制模式设置为 PWM、带死区时间插入的脉冲宽度调制或伪随机 PWM。有两个脉冲宽度调制互补输出线路可供使用。支持 0 到 255 个计数器周期的死区时间插入。

何时使用 TCPWM

可使用定制器将 TCPWM 设置为以下任一模式：

- 带比较功能的定时器
- 带捕获功能的定时器
- PWM
- 带死区时间的 PWM
- 带伪随机序列输出的 PWM
- 正交解码器

对于任何将 PSoC Creator 作为开发环境的用户而言，使用定制器配置此组件是常见的使用案例，因为这是最简单的配置方法。也可在构建时不配置 TCPWM，而在运行时使用软件 API 对其进行配置。这种不配置的用法可用于创建多种应用的设计以及开发 Creator 硬件设计后 TCPWM 在设计中的具体用途为未知的情况。在运行时，可配置除信号连接、时钟和中断以外的所有配置设置。

输入/输出接口

本节介绍 TCPWM 组件的各种输入和输出连接。模式区域指明了可见的 I/O 所代表的功能模式。

输入	模式	说明
clock	所有器件	时钟输入定义此组件的工作频率。最高频率为48 MHz。
reload	定时器，脉冲宽度调制器	此输入使重新加载事件可初始化计数器。在脉冲宽度调制伪随机模式中，重新加载信号与开始信号的功能相同。

输入	模式	说明
index	正交解码器	此索引输入用于检测正交解码器的参考位置。此信号的事件可生成定时器/计数器事件。
count（计数信号）	定时器，脉冲宽度调制器	根据配置，计数信号可递增或递减计数器值。
phiA（相位A）	正交解码器	两个计数输入中的一个，根据其关系和模式，可控制计数值、递增和递减。
start	定时器，脉冲宽度调制器	开始信号不会初始化计数器，但会从当前计数器值继续计数。
phiB（相位B）	正交解码器	两个计数输入中的一个，根据其关系和模式，可控制计数值、递增和递减。
stop	所有器件	该停止信号用于暂停计数器。此事件不会清除当前计数器值。
capture	定时器	向这个输入上发送请求信号可触发捕获当前计数器值。
switch	PWM	此信号可在下一个定时器/计数器事件中交换周期和/或比较寄存器。

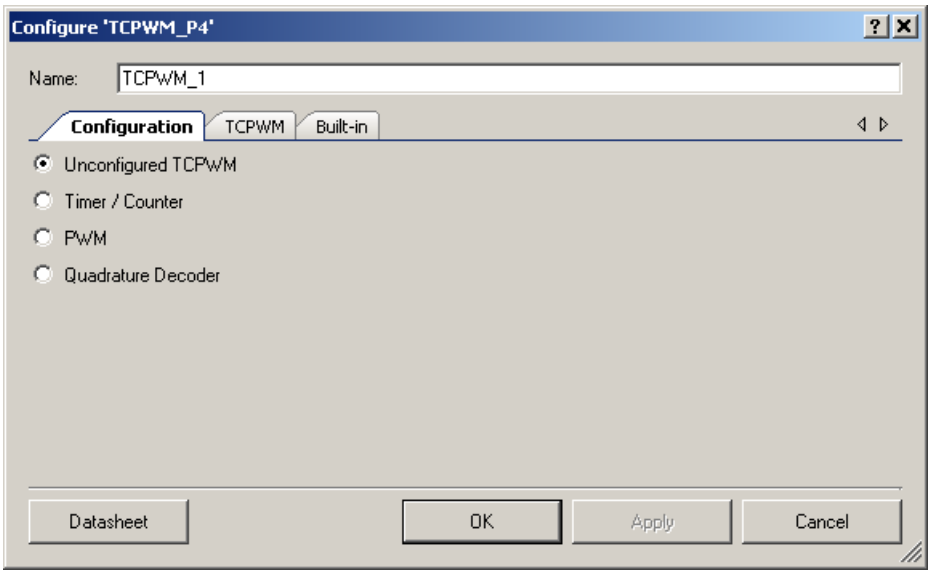
输出	模式	说明
ov	定时器，脉冲宽度调制器	此输出说明计数器上溢出的状态。当向上溢出事件发生时，它将输出高电平。不适用于伪随机脉冲宽度调制模式。
un	定时器，脉冲宽度调制器	当此输出为高电平时，它表示发生了计数器上溢出。不适用于伪随机脉冲宽度调制模式。
cc	所有器件	比较或捕获输出。
line	PWM	脉冲宽度调制输出值。
line_n	PWM	反转的脉冲宽度调制输出值。在死区时间插入模式中，line和line_n都出现了上升沿延迟，从而出现两者都为低电平的时间段。
interrupt	所有器件	中断可由以下任何源触发： • 计数达到其终端计数。 • 一个硬件捕获的执行。 • 比较信号有上升沿。

组件参数

将一个 TCPWM 组件拖放到您的设计上，并双击以打开 **Configure**（配置）对话框。此对话框包含 4 个选项卡，可引导您完成设置 TCPWM 组件的过程：

- 配置：配置 TCPWM 模式
- 定时器/计数器：提供定时器/计数器模式的配置。此选项卡只有在选择了定时器/计数器模式后可见。
- PWM：提供脉冲宽度调制模式的配置。它只有在选择了脉冲宽度调制模式后才可见。
- 正交解码器：提供正交解码器模式的配置。只有在选择了正交解码器模式后可见。

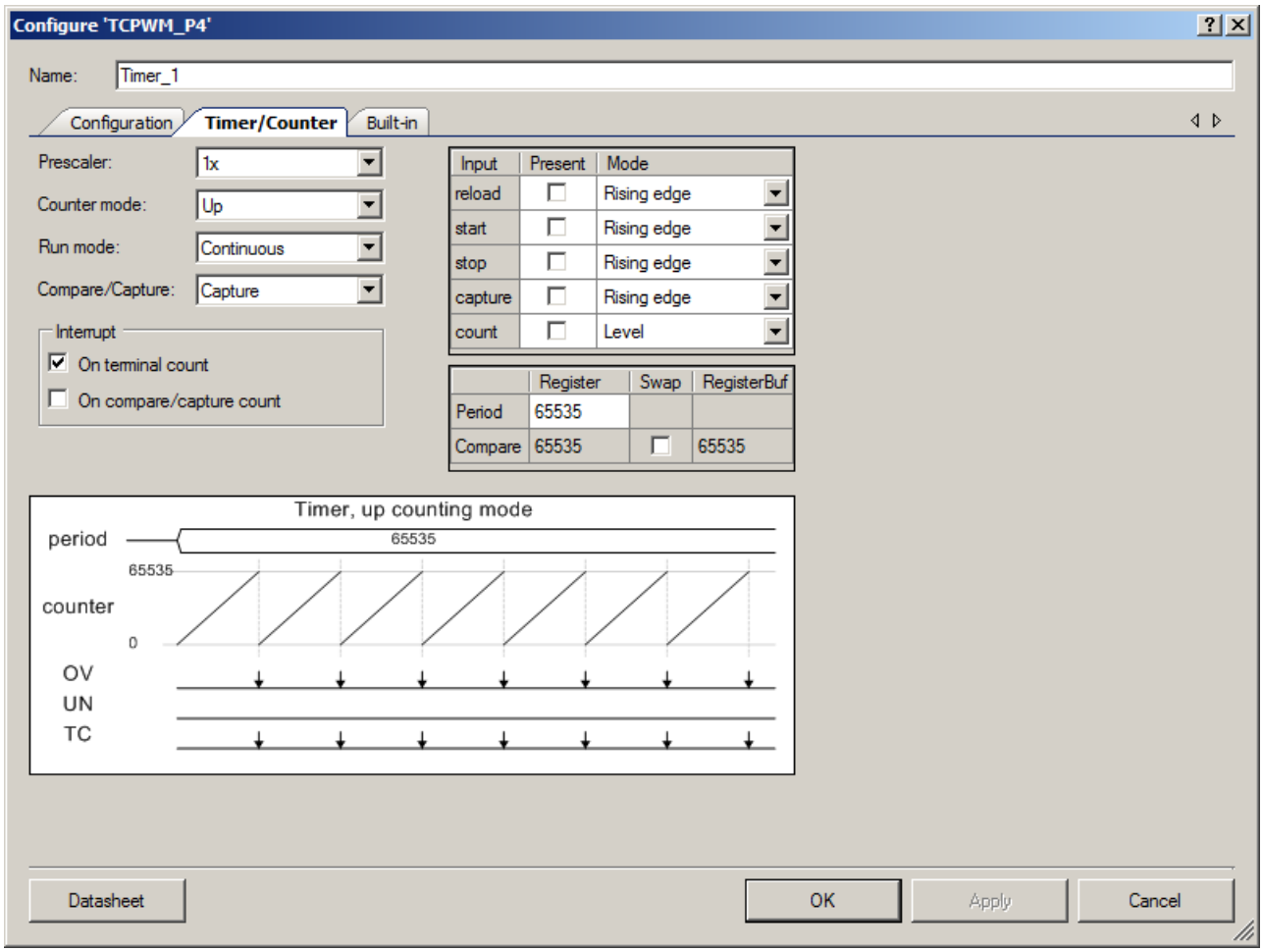
Configuration（配置）选项卡



Configuration（配置）提供用于配置 TCPWM 模式的选项。可用模式如下所示：

- **未配置 TCPWM** — 这是默认模式。配置此模式后，必须在运行时配置 TCPWM 组件。
- **定时器/计数器** — 配置为定时器/计数器模式。定时器/计数器选项卡在选择了此模式后可用。
- **PWM**（脉冲宽度调制） — 配置为脉冲宽度调制模式。脉冲宽度调制选项卡在选择了此模式后可用。
- **Quadrature Decoder**（正交解码器）— 配置为 正交解码器模式。正交解码器选项卡在选择了此模式后可用。

Timer/Counter（定时器/计数器）选项卡



Prescaler（预分频器）

Prescaler（预分频器）参数用于选择应用于时钟的预分频器值。可用值范围为 1 到 128，以 2 倍递增的方式出现。默认值为不对时钟（1x）预分频。

Counter mode（计数器模式）

Counter Mode（计数器模式）参数用于选择计数器的方向。它可设置为 Up（向上）、Down（向下）、Up/Down 0（向上/向下 0）和 Up/Down 1（向上/向下 1）计数模式。Up/Down 0（向上/向下 0）只可针对下溢出触发终端计数（TC），Up/Down 1（向上/向下 1）可针对下溢出和上溢出触发终端计数。

Capture/Compare（捕获/比较）

此参数用于在捕获或比较功能之间选择。一次只有这两种功能中的一种功能可用。

Run mode（运行模式）

可选择 **Run mode**（运行模式）是连续运行还是单触发模式。到达终端计数（TC）后，单触发模式可导致计数器停止。

Input configuration（输入配置）

Input Configuration（输入配置）参数为 5 种输入信号（重新加载、开始、停止、捕获和计数）选择可见性和模式。默认情况下，这些输入在组件符号上不可见。选中设计中需要的每个输入的 Present（显示）复选框可使得显示在组件符号上以进行连接。上述每种信号可配置为由 Rising edge（上升沿）、Falling edge（下降沿）、Either edge（任一沿）或基于信号电平进行触发。

Period（周期）

Period（周期）参数确定周期寄存器的初始值。有效的 **Period**（周期）值范围为 0 到 65535。**Period**（周期）的默认值设置为 65535。

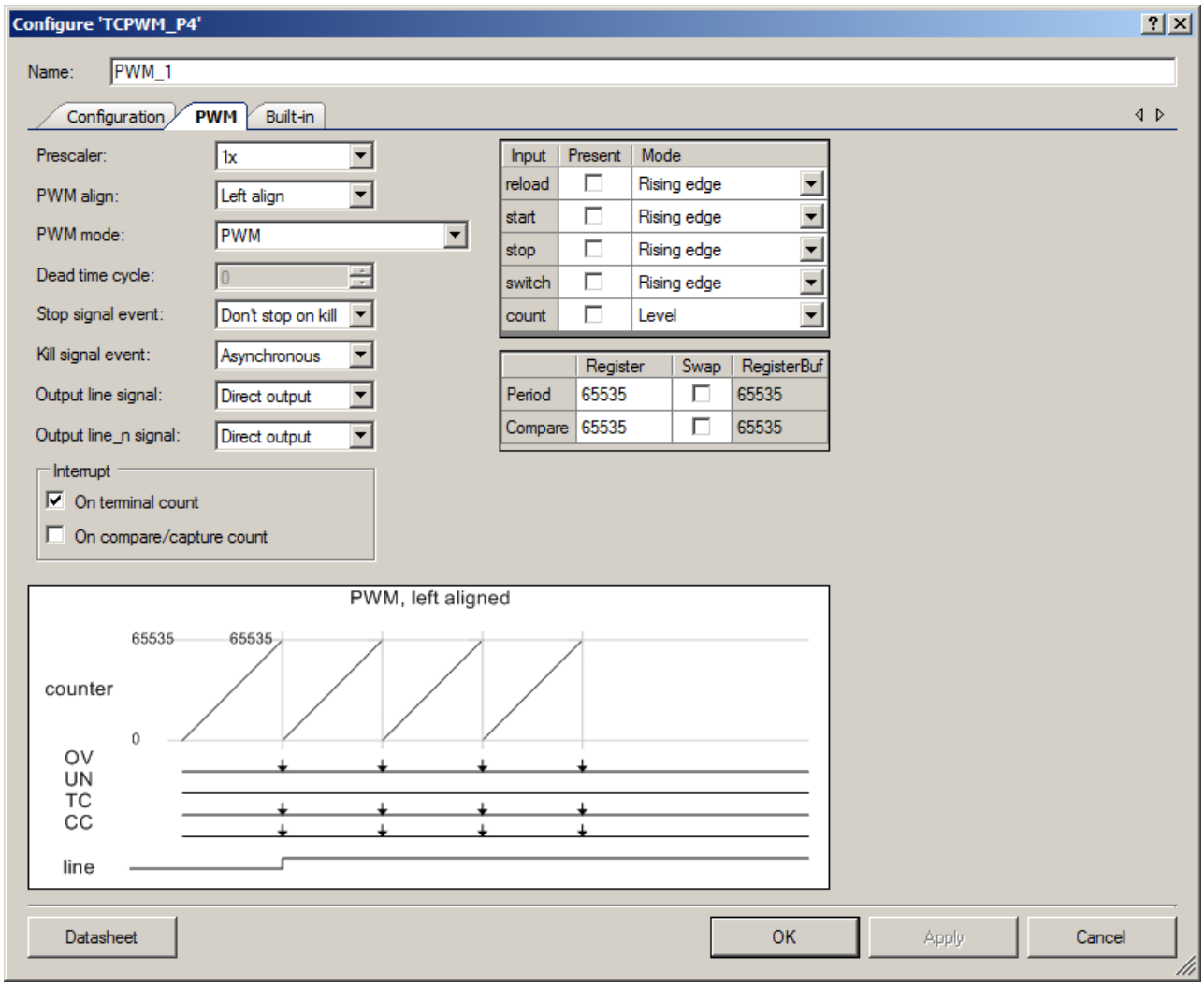
Compare（比较）

Compare（比较）参数用于设置比较寄存器的初始值，swap（切换）复选框用于选择是使用一个还是两个比较值。只有当 **Capture/Compare**（捕获/比较）模式设置为 **Compare**（比较）模式时，才可访问 Compare（比较）参数。默认情况下，不选中 swap（切换）复选框。始终显示第一个比较值，只有在选择了 swap（切换）复选框时才显示第二个比较值。选择 swap（切换）会导致两个比较值在每个 TC 事件中进行交换。有效的 **Compare**（比较）值范围为 0 到 65535。**Compare**（比较）的默认值设置为 65535。

Interrupt（中断）

Interrupt（中断）参数用于确定导致中断的事件。它可设置为在达到中断计数时或达到比较/捕获计数时进行触发。

脉冲宽度调制选项卡



Prescaler（预分频器）

Prescaler（预分频器）参数用于选择应用于时钟的预分频器值。可用值范围为 1 到 128，以 2 倍递增的方式出现。此特性在死区时间模式中不可用。

PWM align（PWM 对齐）

使用 **PWM align**（PWM 对齐）选择脉冲宽度调制波形的波形。它可设置为 Left align（左对齐）、Right align（右对齐）、Center align（中心对齐）或 Asymmetric（不对称）。

PWM mode（PWM 模式）

可将 **PWM mode**（PWM 模式）设置为 PWM、PWM with dead time insertion（带死区时间插入的 PWM）或 Pseudo random PWM（伪随机 PWM）模式。默认设置为 PWM。

Dead time cycles（死区时间周期数）

在选择了带死区时间插入的脉冲宽度调制作为 **PWM mode**（PWM 模式）后，可配置 **Dead time cycles**（死区时间周期数）。此参数用于设置死区时间插入的周期数。此值的范围为 0 到 255。默认值为 0。

Run mode（运行模式）

只有在 Pseudo Random PWM（伪随机 PWM）模式中才显示此选项。**Run mode**（运行模式）可选择为连续运行或处于单触发模式。到达终端计数（TC）后，单触发模式可导致计数器停止。

Stop signal event（停止信号事件）

Stop signal event（停止信号事件）用于确定启动停止信号后采取的操作类型。停止事件可导致 kill 操作（line 和 line_n 输出将处于非活动状态）。此选项用于确定停止事件是否同时引起 kill 操作和停止计数器计数。它可设置为 Stop on kill（在 kill 操作的时候同时停止）或 Don’t stop on kill（不在 kill 操作的时候同时停止）。默认设置为 Don’t stop on kill（不在 kill 操作的时候同时停止）。

Kill signal event（停止输出信号事件）

Kill signal event（停止输出信号事件）参数用于选择停止输出是 Synchronous（同步）还是 Asynchronous（异步）。同步停止输出将终止输出直至下一个 TC 事件再恢复。对于此模式，停止信号必须配置为沿触发信号。异步停止输出将只有在停止信号处于高电平时才终止输出。对于此模式，停止信号必须配置为电平触发信号。默认设置为 Asynchronous（异步）。

Output line signal（输出信号）

Output line signal（输出 line 信号）用于选择在输出线路信号上提供 Direct output（直接输出）还是 Inverse output（反转输出）。默认情况下，它配置为提供 Direct output（直接输出）。

Output line_n signal（输出 line_n 信号）

Output line_n 信号用于选择在 line_n 输出信号是提供 Direct output（直接输出）还是 Inverse output（反转输出）。默认情况下，它配置为提供 Direct output（直接输出）。

Input configuration（输入配置）

Input Configuration（输入配置）参数为 5 种输入信号（重新加载、开始、停止、捕获和计数）选择可见性和模式。默认情况下，这些输入在符号上不可见。选中设计中需要的每个输入的 Present（显示）复选框可使得显示在符号上以进行连接。上述每种信号可配置为由 Rising edge（上升沿）、Falling edge（下降沿）、Either edge（任一沿）或基于信号电平进行触发。

Period（周期）

Period（周期）参数确定周期寄存器的初始值。swap（切换）复选框用于选择是选择一个还是两个周期值。默认情况下，不选中 swap（切换）复选框。始终显示第一个周期值，只有在选择了 swap（切换）复选框时才显示第二个周期值。选择 swap（切换）会导致两个周期值在每个 TC 事件中进行交换。有效的 **Period**（周期）值范围为 0 到 65535。**Period**（周期）的默认值设置为 65535。

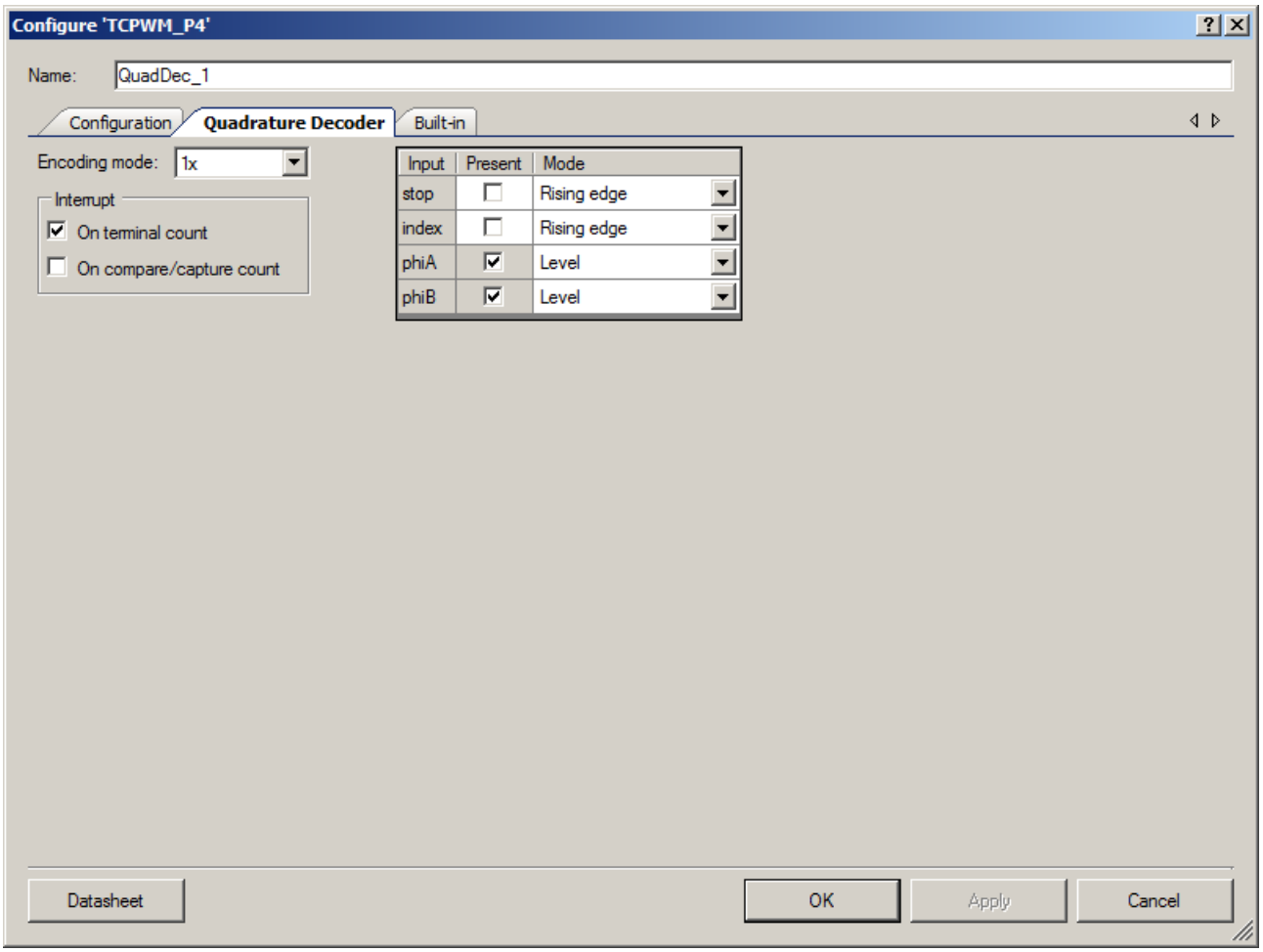
Compare（比较）

Compare（比较）参数用于设置比较寄存器的初始值，swap（切换）复选框用于选择是使用一个还是两个比较值。默认情况下，不选中 swap（切换）复选框。注意，始终显示第一个比较值，只有在选择了 swap（切换）复选框时才显示第二个比较值。选择 swap（切换）会导致两个比较值在每个 TC 事件中进行交换。有效的 **Compare**（比较）值范围为 0 到 65535。**Compare**（比较）的默认值设置为 65535。

Interrupt（中断）

Interrupt（中断）参数用于确定导致中断的事件。它可设置为在达到中断计数时或达到比较/捕获计数时进行触发。

Quadrature Decoder（正交解码器）选项卡



Encoding mode（编码模式）

正交解码器 **Encoding mode**（编码模式）可设置为以下三种模式之一：1x、2x 或 4x 它用于确定测量跃变的计数器的分辨率。分辨率越高，对位置进行编码的精度越高，但同时计数器占用资源也越多。

Input configuration（输入配置）

Input Configuration（输入配置）参数为 4 种输入信号（停止、索引、PhiA 和 PhiB）分别选择可见性和模式。默认情况下，停止输入和索引输入在组件上不可见。选中设计中需要的每个输入的 Present（显示）复选框可使得显示在组件符号上以进行连接。上述每种信号可配置为由 Rising edge（上升沿）、Falling edge（下降沿）、Either edge（任一沿）或基于信号电平进行触发。PhiA 和 PhiB 信号始终显示，且需配置为在 Level（电平）模式

Interrupt（中断）

Interrupt（中断）参数用于确定导致中断的事件。它可设置为在达到中断计数时或达到比较/捕获计数时进行触发。

时钟选择

使用时钟终端提供此时钟。此时钟必须源于全局时钟生成逻辑。时钟预分频器功能在 TCPWM 组件中可用。

放置

TCPWM 组件将放置在其中一个可用的 TCPWM 资源中。使用的特定 TCPWM 不会影响此组件的操作。

应用编程接口

通过应用编程接口（API），您可以使用软件进行配置组件。此表列出了每个函数的接口，并进行了说明。以下各节将更详细地介绍每个函数。

默认情况下，PSoC Creator 将实例名称“TCPWM_1”分配给提供的设计中的第一个组件实例。您可以将其重新命名为遵循标识符语法规则的任何唯一值。实例名称会成为每个全局函数名称、变量和常量符号的前缀。为增加可读性，下表中使用了实例名称“TCPWM”。函数设备表如下所示。

函数	说明
TCPWM_Init()	初始化/恢复默认TCPWM设置
TCPWM_Enable()	使能TCPWM。如果start输入端没有被配置成可见，将启动TCPWM
TCPWM_Start()	第一次调用时使用默认定制器值初始化TCPWM，并使能TCPWM。如果start输入端没有被配置成可见，将启动TCPWM
TCPWM_Stop()	禁用TCPWM
TCPWM_SetMode()	设置TCPWM的操作模式
TCPWM_SetPrescaler()	设置应用于时钟输入的预分频器值
TCPWM_TriggerCommand()	在指定TCPWM实例上触发指定命令
TCPWM_SetOneShot()	写入此寄存器用来控制TCPWM在计数达到周期值后是继续运行还是停止
TCPWM_SetPWMMode()	写入控制寄存器，其将确定在哪种模式中驱动脉冲宽度调制输出

函数	说明
TCPWM_SetPWMSyncKill()	通过写寄存器控制脉冲宽度调制（PWM）的停止输入信号（停止输入）将导致异步还是同步停止输出操作
TCPWM_SetPWMStopOnKill()	通过写寄存器控制脉冲宽度调制（PWM）的停止输入信号（停止输入）是否会导致脉冲宽度调制器的计数器停止
TCPWM_SetPWMDeadTime()	写入死区时间控制值
TCPWM_SetPWMInvert()	通过写入位将控制line和line_n的输出是否需要反转输出
TCPWM_SetQDMode()	将Quadrature Decoder（正交解码器）设置为3种支持模式之一
TCPWM_SetInterruptMode()	设置中断掩码以控制生成中断信号的中断请求
TCPWM_GetInterruptSourceMasked()	获取被中断掩码掩码的中断请求
TCPWM_GetInterruptSource()	获取中断请求（无掩码）
TCPWM_ClearInterrupt()	清除中断请求
TCPWM_SetInterrupt()	设置软件中断请求
TCPWM_WriteCounter()	将新的16位计数器值直接写入到计数器寄存器中
TCPWM_ReadCounter()	读取当前计数器值
TCPWM_SetCounterMode()	设置计数器模式
TCPWM_SetPeriodSwap()	通过写寄存器控制是否交换周期寄存器
TCPWM_SetCompareSwap()	写入寄存器，其将控制是否交换比较寄存器
TCPWM_ReadCapture()	读取捕获的计数器值
TCPWM_ReadCaptureBuf()	读取捕获缓冲区寄存器
TCPWM_WritePeriod()	将新的周期值写入16位周期寄存器中
TCPWM_ReadPeriod()	读取16位周期寄存器
TCPWM_WritePeriodBuf()	将新的周期值写入16位周期缓冲区寄存器中
TCPWM_ReadPeriodBuf()	读取16位周期缓冲区寄存器
TCPWM_WriteCompare()	将新的比较值写入16位比较寄存器中
TCPWM_ReadCompare()	读取比较寄存器
TCPWM_WriteCompareBuf()	将新的比较值写入16位比较缓冲区寄存器中
TCPWM_ReadCompareBuf()	读取比较缓冲区寄存器
TCPWM_SetCaptureMode()	设置捕获触发模式
TCPWM_SetReloadMode()	设置重新加载触发模式
TCPWM_SetStartMode()	设置开始触发模式

函数	说明
TCPWM_SetStopMode()	设置停止触发模式
TCPWM_SetCountMode()	设置计数触发模式
TCPWM_ReadStatus()	读取TCPWM的状态

全局变量

变量	说明
TCPWM_initVar	说明TCPWM是否已初始化。该变量初始化为0，在第一次调用TCPWM_Start()时设置为1。这允许第一次调用TCPWM_Start()子程式后器件无需重新初始化便可重新启动。 如果需要重新初始化此组件，在调用TCPWM_Start()之前调用TCPWM_Init()。也可通过调用TCPWM_Init()和TCPWM_Enable()函数重新初始化TCPWM

函数适用性

函数名称	定时器/计数器（捕获）	定时器/计数器（比较）	PWM	PWM DT	PWM PR	正交解码器
Init	+	+	+	+	+	+
Enable	+	+	+	+	+	+
Start	+	+	+	+	+	+
Stop	+	+	+	+	+	+
SetMode	+	+	+	+	+	+
SetPrescaler	+	+	+	-	+	-
TriggerCommand	+	+	+	+	+	+
SetOneShot	+	+	-	-	+	-
SetPWMMode	-	-	+	+	+	-
SetPWMSyncKill	-	-	+	+	-	-
SetPWMStopOnKill	-	-	+	+	+	-
SetPWMDeadTime	-	-	-	+	-	-
SetPWMInvert	-	-	+	+	+	-
SetQDMode	-	-	-	-	-	+
SetInterruptMode	+	+	+	+	+	+
GetInterruptSourceMasked	+	+	+	+	+	+

函数名称	定时器/计数器（捕获）	定时器/计数器（比较）	PWM	PWM DT	PWM PR	正交解码器
GetInterruptSource	+	+	+	+	+	+
ClearInterrupt	+	+	+	+	+	+
SetInterrupt	+	+	+	+	+	+
WriteCounter	+	+	+	+	+	+
ReadCounter	+	+	+	+	+	+
SetCounterMode	+	+	+	+	-	-
SetPeriodSwap	-	-	+	+	+	-
SetCompareSwap	-	+	+	+	+	-
ReadCapture	+	-	-	-	-	+
ReadCaptureBuf	+	-	-	-	-	+
WritePeriod	+	+	+	+	+	-
ReadPeriod	+	+	+	+	+	-
WritePeriodBuf	-	-	+	+	+	-
ReadPeriodBuf	-	-	+	+	+	-
WriteCompare	-	+	+	+	+	-
ReadCompare	-	+	+	+	+	-
WriteCompareBuf	-	+	+	+	+	-
ReadCompareBuf	-	+	+	+	+	-
SetCaptureMode	+	-	+（切换）	+（切换）	+（切换）	-
SetReloadMode	+	+	+	+	+	+（索引）
SetStartMode	+	+	+	+	+	+(phiB)
SetStopMode	+	+	+	+（非同步停止输入）	+	+
SetCountMode	+	+	+	+	+	+(phiA)
ReadStatus	+	+	+	+	+	+

void TCPWM_Init(void)

说明： 初始化/恢复默认TCPWM设置。

参数： 无

返回值： 无

其他影响： 无

void TCPWM_Enable(void)

说明： 使能TCPWM。如果start引脚不存在，将启动TCPWM。如果存在start引脚，则计数器将基于此信号开始。

参数： 无

返回值： 无

其他影响： 无

void TCPWM_Start(void)

说明： 第一次调用时使用默认定制器值初始化TCPWM，并使能TCPWM。对于后续调用，配置保持不变，此器件将被使能。如果start引脚不存在，将启动TCPWM。如果存在start引脚，则计数器将基于此信号开始。

参数： 无

返回值： 无

其他影响： 无

void TCPWM_Stop(void)

说明： 禁用TCPWM。

参数： 无

返回值： 无

其他影响： 无

void TCPWM_SetMode(uint32 mode)

说明： 设置TCPWM的操作模式。当此器件被设置为未配置的TCPWM时将使用此函数在运行时设定其工作模式。当此组件被禁用时，模式必须要被设置。

参数： uint32 mode：TCPWM将操作的模式。

值	说明
TCPWM_MODE_TIMER_COMPARE	带比较功能的定时器/计数器
TCPWM_MODE_TIMER_CAPTURE	带捕获功能的定时器/计数器
TCPWM_MODE_QUAD	正交解码器
TCPWM_MODE_PWM	PWM
TCPWM_MODE_PWM_DT	带死区时间的PWM
TCPWM_MODE_PWM_PR	带伪随机功能的PWM

返回值： 无

其他影响： 无

void TCPWM_SetPrescaler(uint32 prescaler)

说明： 设置应用于时钟输入的预分频器值。它不适用于带死区时间的脉冲宽度调制或正交解码器模式。

参数： uint32 prescaler：预分频器分频器值。

值	说明
TCPWM_PRESCALE_DIVBY1	1分频（无预分频）
TCPWM_PRESCALE_DIVBY2	2分频
TCPWM_PRESCALE_DIVBY4	4分频
TCPWM_PRESCALE_DIVBY8	8分频
TCPWM_PRESCALE_DIVBY16	16分频
TCPWM_PRESCALE_DIVBY32	32分频
TCPWM_PRESCALE_DIVBY64	64分频
TCPWM_PRESCALE_DIVBY128	128分频

返回值： 无

其他影响： 无

void TCPWM_TriggerCommand(uint32 mask, uint32 command)

说明： 在指定TCPWM实例上触发指定命令。使用掩码将此命令应用于多个实例。从而可实现多个TCPWM实例保持同步。

参数： uint32 mask：此命令应应用于的每个TCPWM实例的掩码位组合。一个实例的函数可用于将命令应用于设计中的任何实例。特定实例的掩码值适用于该实例的TCPWM_MASK定义。

uint32 command：已枚举的命令值。

值	说明
TCPWM_CMD_CAPTURE	触发捕获命令
TCPWM_CMD_RELOAD	触发重新加载命令
TCPWM_CMD_STOP	触发停止命令
TCPWM_CMD_START	触发开始命令

返回值： 无

其他影响： 无

void TCPWM_SetPWMMode(uint32 modeMask)

说明： 写入控制寄存器，其将确定在哪种模式中驱动脉冲宽度调制输出线路。存在针对比较匹配（CC_MATCH）、上溢出（OVERFLOW）和下溢出（UNDERFLOW）的操作的设置。上述3个值都必须执行“或”运算以构成对应的模式。

参数： uint32 modeMask：3个模式设置的组合。 掩码必须包括3个模式设置的值，或使用一个预配置的脉冲宽度调制设置。

值	说明
TCPWM_CC_MATCH_SET	对于比较值匹配时拉高信号
TCPWM_CC_MATCH_CLEAR	对于比较值匹配时拉低信号
TCPWM_CC_MATCH_INVERT	对于比较值匹配时反转信号
TCPWM_CC_MATCH_NO_CHANGE	对于比较值匹配时不做处理
TCPWM_OVERLOW_SET	向上溢出时拉高信号
TCPWM_OVERLOW_CLEAR	向上溢出时拉低信号
TCPWM_OVERLOW_INVERT	向上溢出时反转信号
TCPWM_OVERLOW_NO_CHANGE	向上溢出时不做处理
TCPWM_UNDERFLOW_SET	向下溢出时拉高信号
TCPWM_UNDERFLOW_CLEAR	向下溢出时拉低信号
TCPWM_UNDERFLOW_INVERT	向下溢出时反转信号
TCPWM_UNDERFLOW_NO_CHANGE	向下溢出时不做处理
TCPWM_PWM_MODE_LEFT	左对齐脉冲宽度调制的设置。应结合向上计数模式
TCPWM_PWM_MODE_RIGHT	右对齐脉冲宽度调制的设置。应结合向下计数模式
TCPWM_PWM_MODE_CENTER	中心对齐脉冲宽度调制的设置。应结合向上/向下0模式。
TCPWM_PWM_MODE_ASYM	异步脉冲宽度调制的设置。应结合向上/向下1模式。

返回值： 无

其他影响： 无

void TCPWM_SetOneShot(uint32 oneShotEnable)

说明： 写入寄存器，其将控制TCPWM在达到终端计数后是连续运行还是停止。默认情况下，TCPWM在连续运行模式下运行。它适用于定时器/计数器或伪随机脉冲管宽度调制。

参数： uint32 oneShotEnable：0 = Continuous（连续）；1 = Enable One Shot（使能单触发）。

返回值： 无

其他影响： 无

void TCPWM_SetPWMSyncKill(uint32 syncKillEnable)

说明： 通过写寄存器控制脉冲宽度调制停止输入信号（停止输入）将导致异步还是同步停止输出操作。对于同步模式，停止信号将导致停止line和line_n信号输出直至下一个终端计数值才恢复。此模式需要停止输入处于上沿模式。对于异步模式，当存在停止输入信号时，将禁用line和line_n信号。此模式应只能在停止信号（停止输入）设置位于通过模式（电平敏感信号）时可用。默认情况下，kill操作是异步的。此功能仅适用于脉冲宽度调制以及带死区时间的脉冲宽度调制模式。

参数： uint32 syncKillEnable：0 = Asynchronous（异步）；1 = Synchronous（同步）。

返回值： 无

其他影响： 无

void TCPWM_SetPWMStopOnKill(uint32 stopOnKillEnable)

说明： 通过写寄存器控制脉冲宽度调制器非同步停止输入信号（停止输入）是否导致脉冲宽度调制计数器停止。默认情况下，kill操作不会停止此计数器。此功能仅适用于三种脉冲宽度调制模式。

参数： uint32 stopOnKillEnable：0 = Don’t stop（不停止）；1 = Stop（停止）。

返回值： 无

其他影响： 无

void TCPWM_SetPWMDeadTime(uint32 deadTime)

说明： 写入死区时间控制值。该值会延迟line信号和line_n信号的上升沿（在直接信号模式下）或下降沿（在反转信号模式下）的到来；该延迟的时长等于所指定的时钟周期数。这样会使两种信号在多个周期内一直处于非活动状态。此功能仅适用于带死区时间的脉冲宽度调制模式。

参数： uint32 deadTime：要插入的死区时间。范围为0到255。

返回值： 无

其他影响： 无

void TCPWM_SetPWMInvert(uint32 mask)

说明： 通过位操作控制line和 line_n 输出是否经过反转后再输出。此功能仅适用于三种脉冲宽度调制模式。

参数： uint32 mask：要反转的输出的掩码。掩码中未包括的输出被设置为正常非反转操作。

值	说明
TCPWM_INVERT_LINE	反转line输出
TCPWM_INVERT_LINE_N	反转line_n输出

返回值： 无

其他影响： 无

void TCPWM_SetQDMode(uint32 qdMode)

说明： 将Quadrature Decoder（正交解码器）设置为3种支持模式之一。此功能仅适用于正交解码器操作。

参数： uint32 qdMode：正交解码器模式。

值	说明
TCPWM_MODE_X1	phi A上沿的计数
TCPWM_MODE_X2	phiA双沿的计数（快2倍）
TCPWM_MODE_X4	phiA和phiB的双沿的计数（快4倍）

返回值： 无

其他影响： 无

void TCPWM_SetInterruptMode(uint32 interruptMask)

说明： 设置中断掩码以控制生成中断信号的中断请求。

参数： uint32 interruptMask：要使能的位的掩码。

值	说明
TCPWM_INTR_MASK_TC	终端计数掩码
TCPWM_INTR_MASK_CC_MATCH	比较计数/捕获掩码

返回值： 无

其他影响： 无

uint32 TCPWM_GetInterruptSourceMasked(void)

说明： 获取被中断掩码掩码的中断请求。

参数： 无

返回值： uint32 已掩码中断源。

值	说明
TCPWM_INTR_MASK_TC	终端计数掩码
TCPWM_INTR_MASK_CC_MATCH	比较计数/捕获掩码

其他影响： 无

uint32 TCPWM_GetInterruptSource(void)

说明： 获取中断请求（无掩码）。

参数： 无

返回值： uint32 中断请求值。

值	说明
TCPWM_INTR_MASK_TC	终端计数掩码
TCPWM_INTR_MASK_CC_MATCH	比较计数/捕获掩码

其他影响： 无

void TCPWM_ClearInterrupt(uint32 interruptMask)

说明： 清除中断请求。

参数： uint32 interruptMask：需要清除的中断的掩码。

值	说明
TCPWM_INTR_MASK_TC	终端计数掩码
TCPWM_INTR_MASK_CC_MATCH	比较计数/捕获掩码

返回值： 无

其他影响： 无

void TCPWM_SetInterrupt(uint32 interruptMask)

说明： 设置软件中断请求。

参数： uint32 interruptMask：需要设置的中断的掩码。

值	说明
TCPWM_INTR_MASK_TC	终端计数掩码
TCPWM_INTR_MASK_CC_MATCH	比较计数/捕获掩码

返回值： 无

其他影响： 无

void TCPWM_WriteCounter(uint32 count)

说明： 将新的16位计数器值直接写入计数器寄存器中，从而将计数器（而非周期）设置为写入的值。不建议在计数器正在运行时写入到此字段中。

参数： uint32 count：要写入的值。

返回值： 无

其他影响： 无

uint32 TCPWM_ReadCounter(void)

说明： 读取当前计数器值。

参数： 无

返回值： uint32当前计数器值。

其他影响： 无

void TCPWM_SetCounterMode(uint32 counterMode)

说明： 设置计数器模式。它适用于所有模式，除了正交解码器和带伪随机输出的脉冲宽度调制器。

参数： uint32 counterMode：已枚举计数器类型值。

值	说明
TCPWM_COUNT_UP	向上计数
TCPWM_COUNT_DOWN	向下计数
TCPWM_COUNT_UPDOWN0	向上和向下计数。当计数器达到0时，生成终端计数
TCPWM_COUNT_UPDOWN1	向上和向下计数。当计数器达到0时以及完成周期时，生成终端计数

返回值： 无

其他影响： 无

void TCPWM_SetPeriodSwap(uint32 swapEnable)

说明： 通过写寄存器控制是否交换周期寄存器。在脉冲宽度调制模式中启用时，在硬件切换事件后的下一个TC事件中将发生交换。不适用于正交解码器和定时器/计数器模式。

参数： uint32 swapEnable：0 = Disable swap（禁用交换）；1 = Enable swap（使能交换）。

返回值： 无

其他影响： 无

void TCPWM_SetCompareSwap(uint32 swapEnable)

说明： 通过写寄存器控制是否交换比较寄存器。在定时器/计数器（无捕获）模式中使能时，TC事件中将发生交换。在脉冲宽度调制模式中，在硬件切换事件后的下一个TC事件中将发生交换。不适用于带捕获的定时器/计数器或正交解码器模式。

参数： uint32 swapEnable：0 = Disable swap（禁用交换）；1 = Enable swap（使能交换）。

返回值： 无

其他影响： 无

uint32 TCPWM_ReadCapture(void)

说明： 读取捕获的计数器值。此API不适用于带捕获的定时器/计数器和正交解码器模式。

参数： 无

返回值： uint32已捕获值。

其他影响： 无

uint32 TCPWM_ReadCaptureBuf(void)

说明： 读取捕获缓冲区寄存器。此API不适用于带捕获的定时器/计数器和正交解码器模式。

参数： 无

返回值： uint32捕获缓冲区值

其他影响： 无

void TCPWM_WritePeriod(uint32 period)

说明： 将新的周期值写入16位周期寄存器中。要使计数器为计数N个周期，此寄存器应写入N-1（计数为0到周期数，包含周期数）。此API不适用于正交解码器模式。

参数： uint32 period：周期值。

返回值： 无

其他影响： 在PWM模式下，当**PWM align**参数被设置为Left align（左对齐）时，计数器值将从零递增计数到周期值。然后，该计数器将复位为0，并重新开始计数。然而，如果新的周期值被设为低于当前的计数器值，那么PWM将不会正常工作。复位为0并连续正常工作前，该计数器会计数到65535。为了避免这种状况，当周期值在左对齐模式下被设为较小的值时，请调用TCPWM_WriteCounter(0)。

uint32 TCPWM_ReadPeriod(void)

说明： 读取16位周期寄存器。

参数： 无

返回值： uint32周期值。

其他影响： 无

void TCPWM_WritePeriodBuf(uint32 periodBuf)

说明： 将新的周期值写入16位周期缓冲区寄存器中。此API不适用于所有脉冲宽度调制模式。

参数： uint32 periodBuf：周期值。

返回值： 无

其他影响： 无

uint32 TCPWM_ReadPeriodBuf(void)

说明： 读取16位周期缓冲区寄存器。此API不适用于所有脉冲宽度调制模式。

参数： 无

返回值： uint32周期缓冲区值。

其他影响： 无

void TCPWM_WriteCompare(uint32 compare)

说明： 将新的比较值写入16位比较寄存器中。不适用于带捕获的定时器/计数器或正交解码器模式。

参数： uint32 compare：比较值。

返回值： 无

其他影响： 无

uint32 TCPWM_ReadCompare(void)

说明： 读取比较寄存器。不适用于带捕获的定时器/计数器或正交解码器模式。

参数： 无

返回值： uint32比较值。

其他影响： 无

void TCPWM_WriteCompareBuf(uint32 compareBuf)

说明： 将新的比较值写入16位比较缓冲区寄存器中。不适用于带捕获的定时器/计数器或正交解码器模式。

参数： uint32 compareBuf：比较值。

返回值： 无

其他影响： 无

uint32 TCPWM_ReadCompareBuf(void)

说明： 读取比较缓冲区寄存器。不适用于带捕获的定时器/计数器或正交解码器模式。

参数： 无

返回值： uint32比较缓冲区值。

其他影响： 无

void TCPWM_SetCaptureMode(uint32 triggerMode)

说明： 设置捕获触发模式。对于脉冲宽度调制模式，它为切换输入。此输入不适用于无捕获的定时器/计数器。对于脉冲宽度调制模式，它为切换输入。

参数： uint32 triggerMode：已枚举触发模式值。

值	说明
TCPWM_TRIG_LEVEL	电平
TCPWM_TRIG_RISING	上升沿
TCPWM_TRIG_FALLING	下降沿
TCPWM_TRIG_BOTH	上升沿和下降沿

返回值： 无

其他影响： 无

void TCPWM_SetReloadMode(uint32 triggerMode)

说明： 设置重新加载触发模式。对于正交解码器模式，它为索引输入。

参数： uint32 triggerMode：已枚举触发模式值。

值	说明
TCPWM_TRIG_LEVEL	电平
TCPWM_TRIG_RISING	上升沿
TCPWM_TRIG_FALLING	下降沿
TCPWM_TRIG_BOTH	上升沿和下降沿

返回值： 无

其他影响： 无

void TCPWM_SetStartMode(uint32 triggerMode)

说明： 设置开始触发模式。对于正交解码器模式，它为phiB输入。

参数： uint32 triggerMode：已枚举触发模式值。

值	说明
TCPWM_TRIG_LEVEL	电平
TCPWM_TRIG_RISING	上升沿
TCPWM_TRIG_FALLING	下降沿
TCPWM_TRIG_BOTH	上升沿和下降沿

返回值： 无

其他影响： 无

void TCPWM_SetStopMode(uint32 triggerMode)

说明： 设置停止触发模式。对于脉冲宽度调制模式，它为停止输入。

参数： uint32 triggerMode：已枚举触发模式值。

值	说明
TCPWM_TRIG_LEVEL	电平
TCPWM_TRIG_RISING	上升沿
TCPWM_TRIG_FALLING	下降沿
TCPWM_TRIG_BOTH	上升沿和下降沿

返回值： 无

其他影响： 无

void TCPWM_SetCountMode(uint32 triggerMode)

说明： 设置计数触发模式。对于正交解码器模式，它为phiA输入。

参数： uint32 triggerMode：已枚举触发模式值。

值	说明
TCPWM_TRIG_LEVEL	电平
TCPWM_TRIG_RISING	上升沿
TCPWM_TRIG_FALLING	下降沿
TCPWM_TRIG_BOTH	上升沿和下降沿

返回值： 无

其他影响： 无

uint32 TCPWM_ReadStatus(void)

说明： 读取TCPWM的状态。

参数： 无

返回值： uint32状态。

值	说明
STATUS_DOWN	设置是否向下计数
STATUS_RUNNING	设置计数器是否正在运行

其他影响： 无

MISRA 合规性

本节介绍了 MISRA-C:2004 合规性和本器件的偏差。定义了两种类型的偏差：项目偏差 — 适用于所有 PSoC Creator 组件的偏差；特定偏差 — 仅适用于该组件的偏差。本节提供了有关组件特定偏差的信息。*《系统参考指南》*的 MISRA 合规性章节中介绍了项目偏差以及有关 MISRA 合规性验证环境的信息。

TCPWM 组件没有任何特定偏差。

示例固件源代码

在 Find Example Project 对话框中，PSoC Creator 提供了大量的示例项目，包括原理图和示例代码。要获取组件示例，请打开组件目录中的对话框或原理图中的组件实例。要查看通用示例，请打开 Start Page 或 **File** 菜单中的对话框。根据要求，可以通过使用对话框中的 **Filter Options** 项来限定可选的项目列表。

有关更多信息，请参考 PSoC Creator 帮助中的“Find Example Project”（查找示例项目）主题。

功能说明

特定模式中重新定义了一些事件功能：

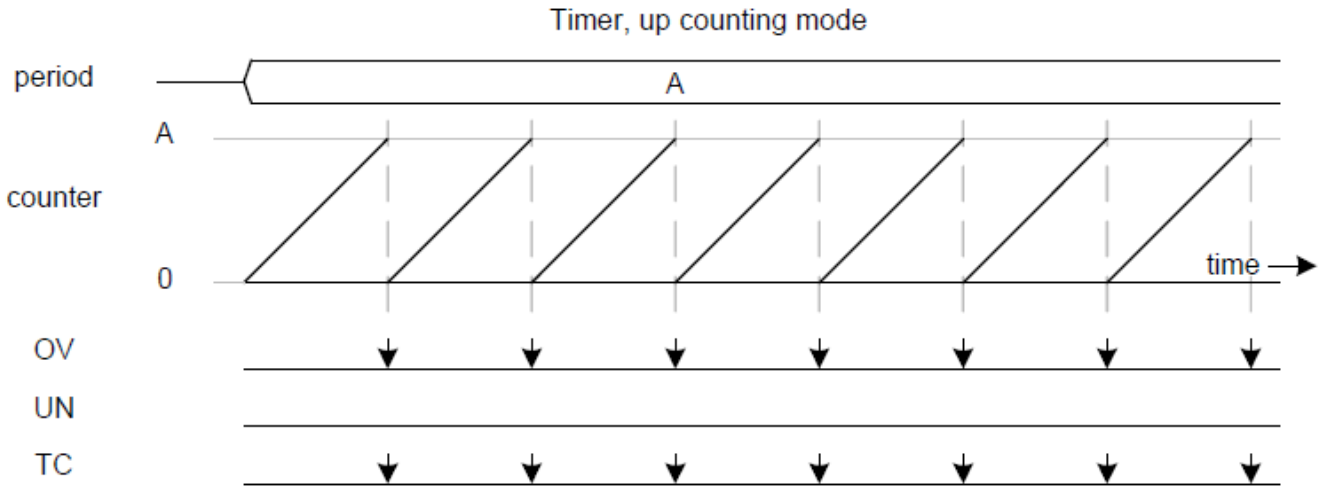
- 在正交模式中，重新加载事件作为正交索引事件。索引/重新加载指示已完成旋转。
- 在正交模式下，计数事件作为正交相位 phiA。
- 在正交模式下，开始事件作为正交相位 phiB。
- 在脉冲宽度调制模式中，停止事件作为禁止输出事件。停止事件或禁止事件将近禁能脉冲宽度调制信号输出。
- 在脉冲宽度调制模式中，捕获事件作为切换事件。它在比较寄存器值和缓冲的比较寄存器值之间切换。可使用此功能调制脉冲宽度。

定时器/计数器

以下三个图显示的是四种计数模式下定时器的行为：向上、向下、向上/向下 0、向上/向下 1。这些图显示了周期寄存器（包含最大计数器值）、与时间相关的计数器值，以及“ov”和“un”引脚的触发时间。

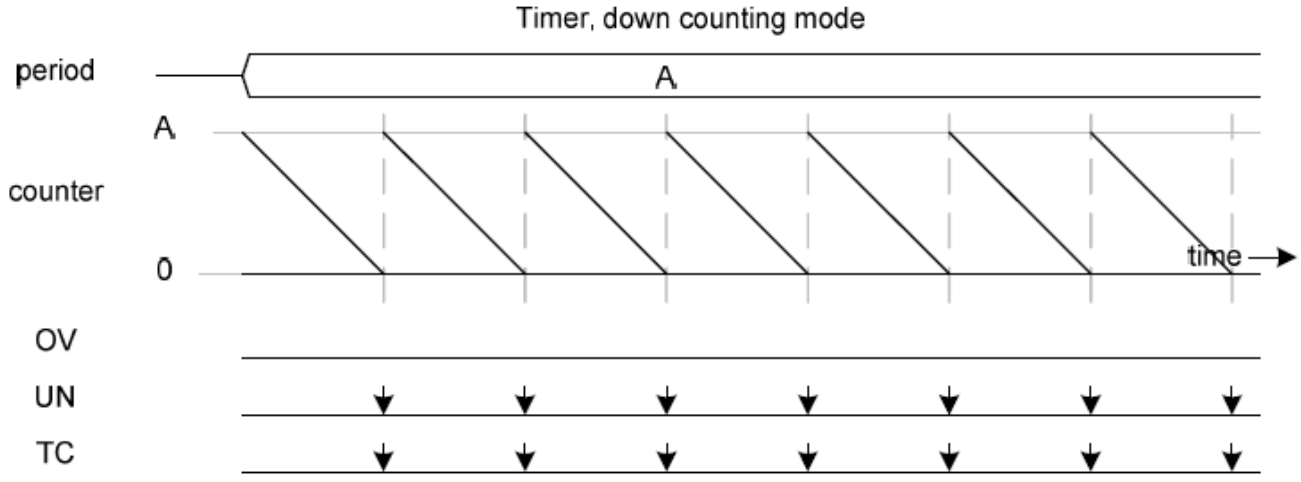
向上模式

在 Up（向上）计数模式中，当周期寄存器值为 A 时，计数器从 0 开始向上计数至 A，将导致 A+1 个计数器值。在下一个循环中，计数器返回到 0 并再次向上计数至 A。当计数器达到此点时，将触发上溢出信号（ov）。



向下模式

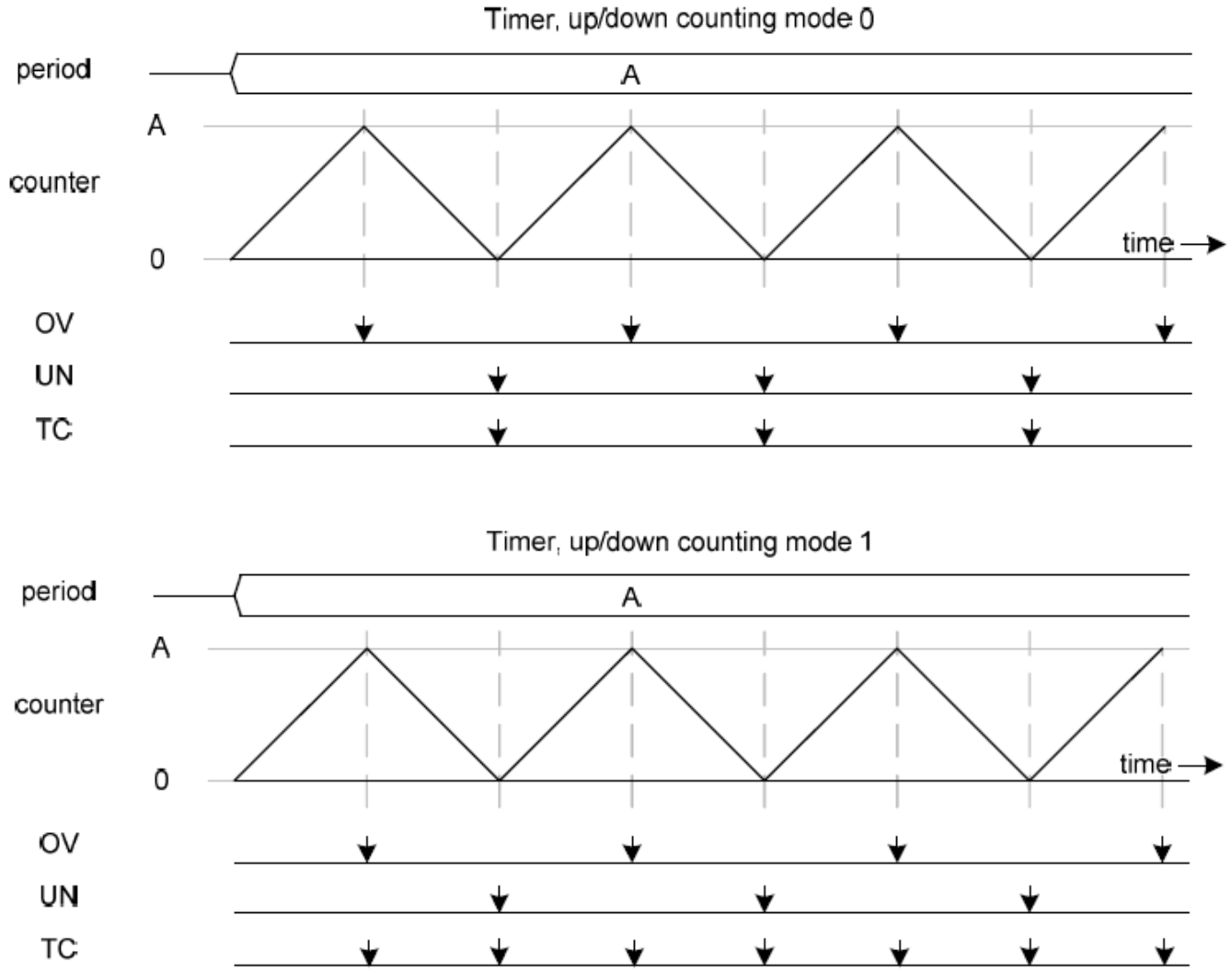
在 Down（向下）计数模式中，当周期寄存器值为 A 时，计数器从 A 开始向下计数至 0，将导致共 A+1 个计数器值。然后计数器将重新复位为 A，并触发下溢出信号“un”。



向上/向下模式

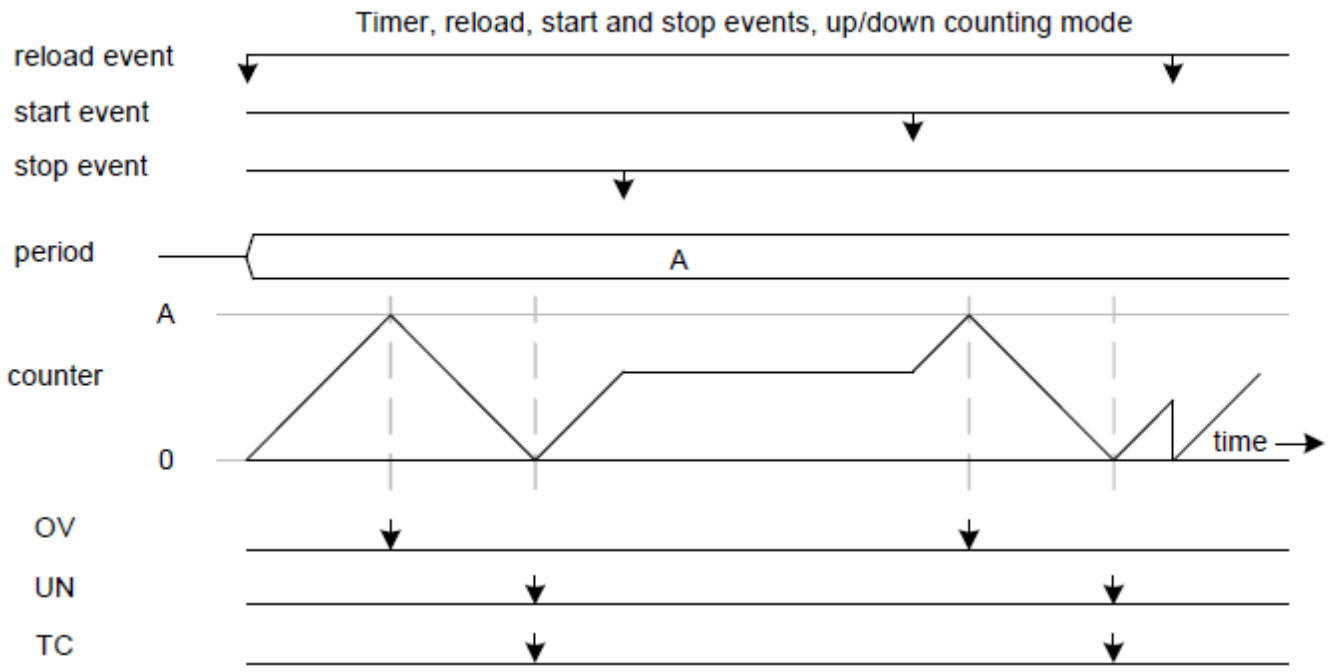
在 Up/Down（向上/向下）计数模式 0 和 1 中，当周期寄存器值为 A 时，从 0 开始向上计数至 A，然后向下计数至 0。因此，整个周期将观察到 2*A 计数器值。当计数器达到 A 时将触发“ov”信号，当计数器达到 0 时将触发“un”信号。

在向上/向下计数模式 0 中，当计数器达到“0”时，将生成终端计数（TC）条件；当计数器值达到周期值时，将不生成 TC 条件。在向上/向下计数模式 1 中，当计数器达到“0”和周期值时，都将生成终端计数（TC）条件。

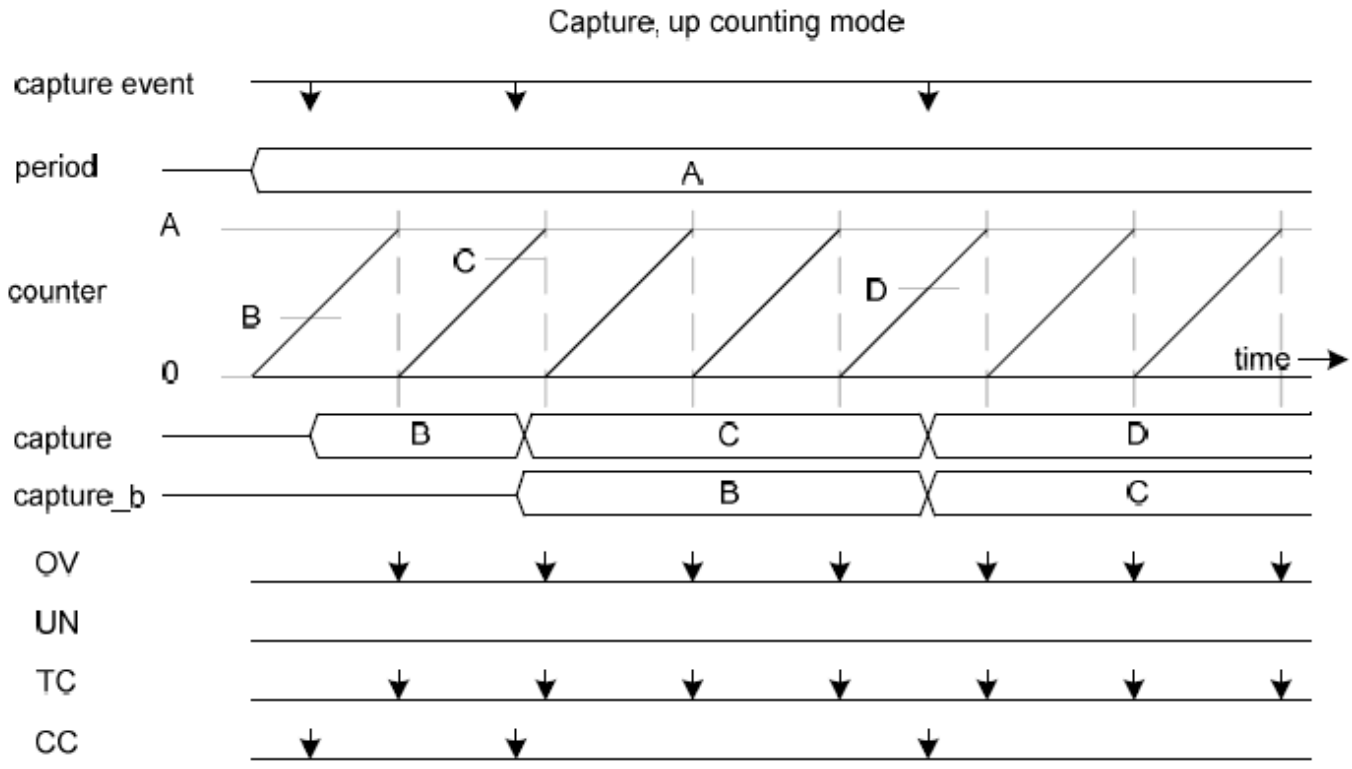


操作

此第一个示例显示如何使用重新加载事件、启动事件和停止事件控制计数器。考虑定时器处于向上/向下计数模式 0 中。启动重新加载事件会将计数器初始化为 0。启动事件不会初始化计数器，但会从当前计数器值继续计数。停止事件停止计数并保持当前计数器值不受影响。这些操作以图形方式显示如下。



第二个示例显示定时器的捕获行为。考虑在 Up（向上）计数模式中配置定时器。当发生捕获事件时，计数器值将被复制到捕获寄存器中。捕获值将被复制到缓冲的捕获寄存器中。捕获到计数器值时，将生成 CC 条件。此条件可用于生成中断。

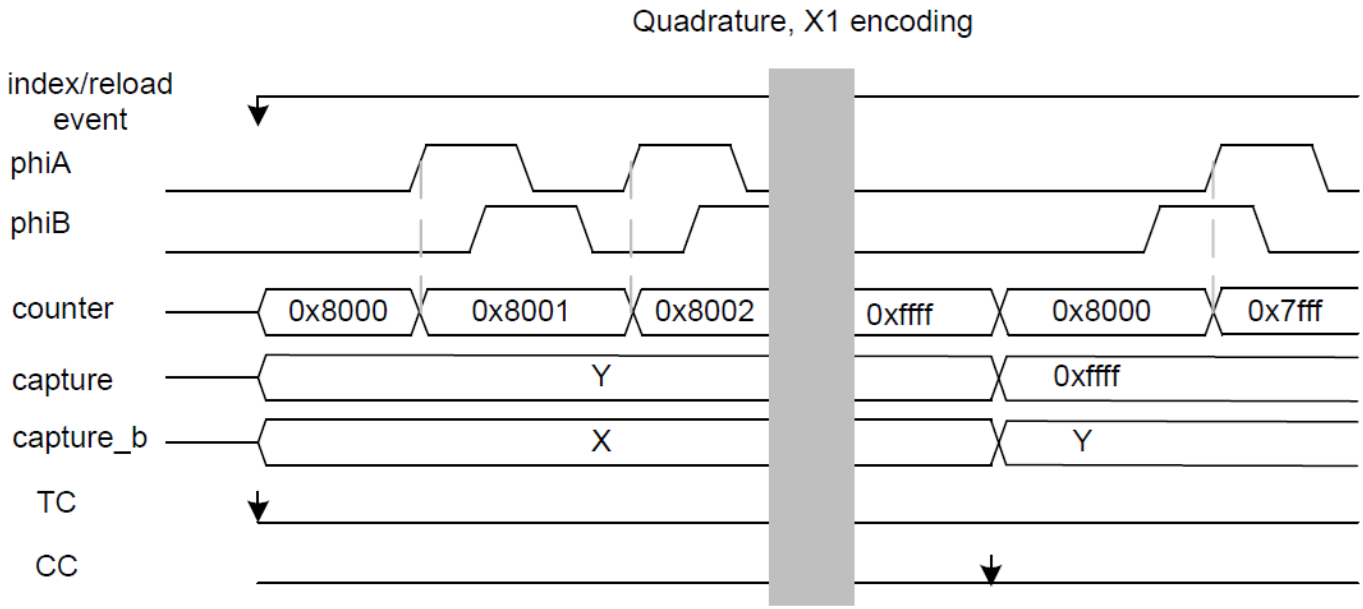


对计数器进行的递增和递减都受计数事件和计数器时钟使能信号的控制。通常的操作使用 1 作为计数事件，并使用 TCPWM 时钟作为计数器时钟，而无需进行任何时钟预调节。但高级操作可能使用计数事件配置。例如，可配置计数事件以从 DSI 中对触发输入的上升沿进行计数。

事件对计数器的影响与计数器的时钟频率有关。事件的检测在计数器频率上进行的。当事件频率超出计数器时钟频率时，可能会丢失事件。对于高频率事件（在计数器频率域内）和使用预调节计数器时钟的定时器配置，这种情况更容易发生。

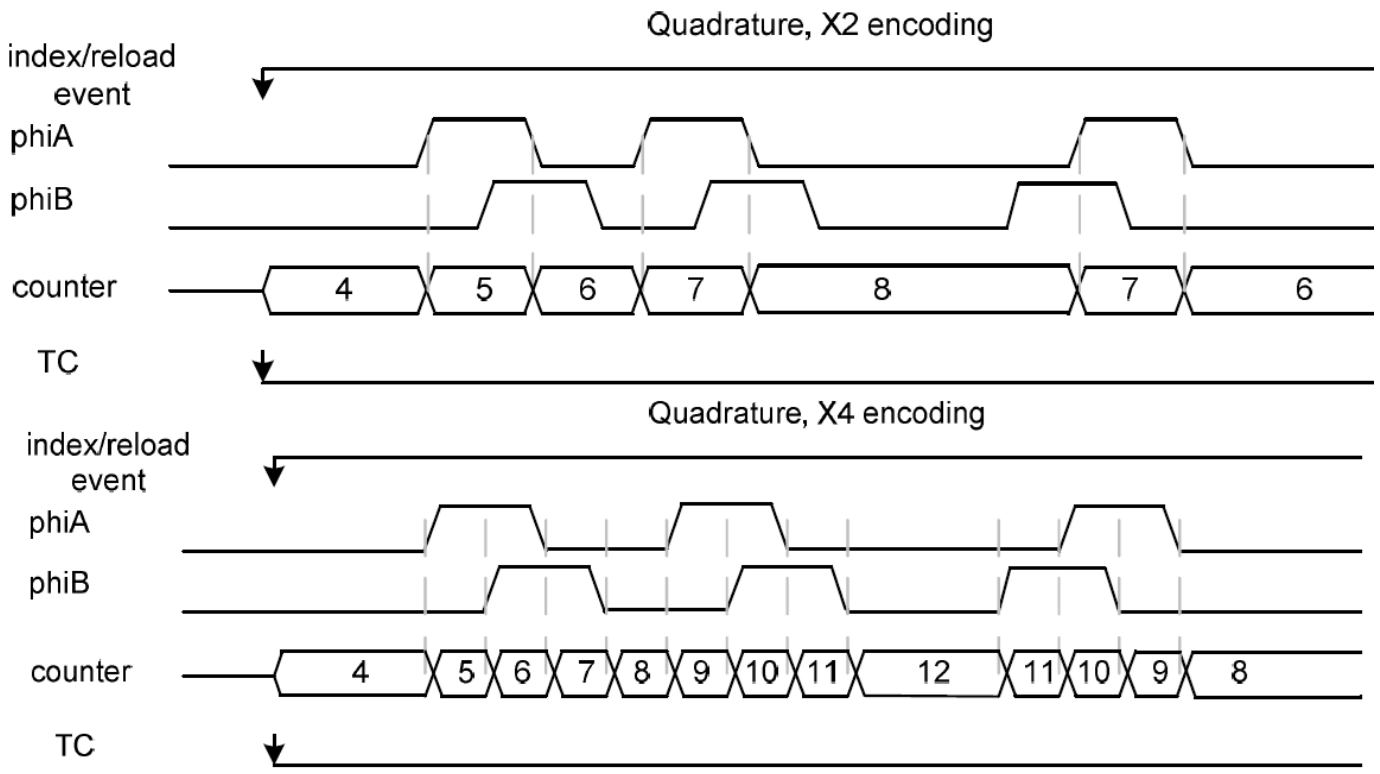
正交解码器

下图显示的是正交解码器在 1x 编码模式中的行为：当 phiB 为 0 时，phiA 上的正向沿对计数器进行递增，当 phiB 为 1 时，phiA 上的正向沿对计数器进行递减。将针对索引事件初始化计数器，且中点计数器值为“0x8000”。初始化计数器时，还将生成终端计数（TC）条件。此条件可用于生成中断。



当计数器达到“0xFFFF”（最大计数器值）时，此计数器值将被复制到捕获寄存器中，且计数器将初始化为“0x8000”（中点计数器值）。复制此计数器值时，将生成 CC 条件。此条件可用于生成中断。

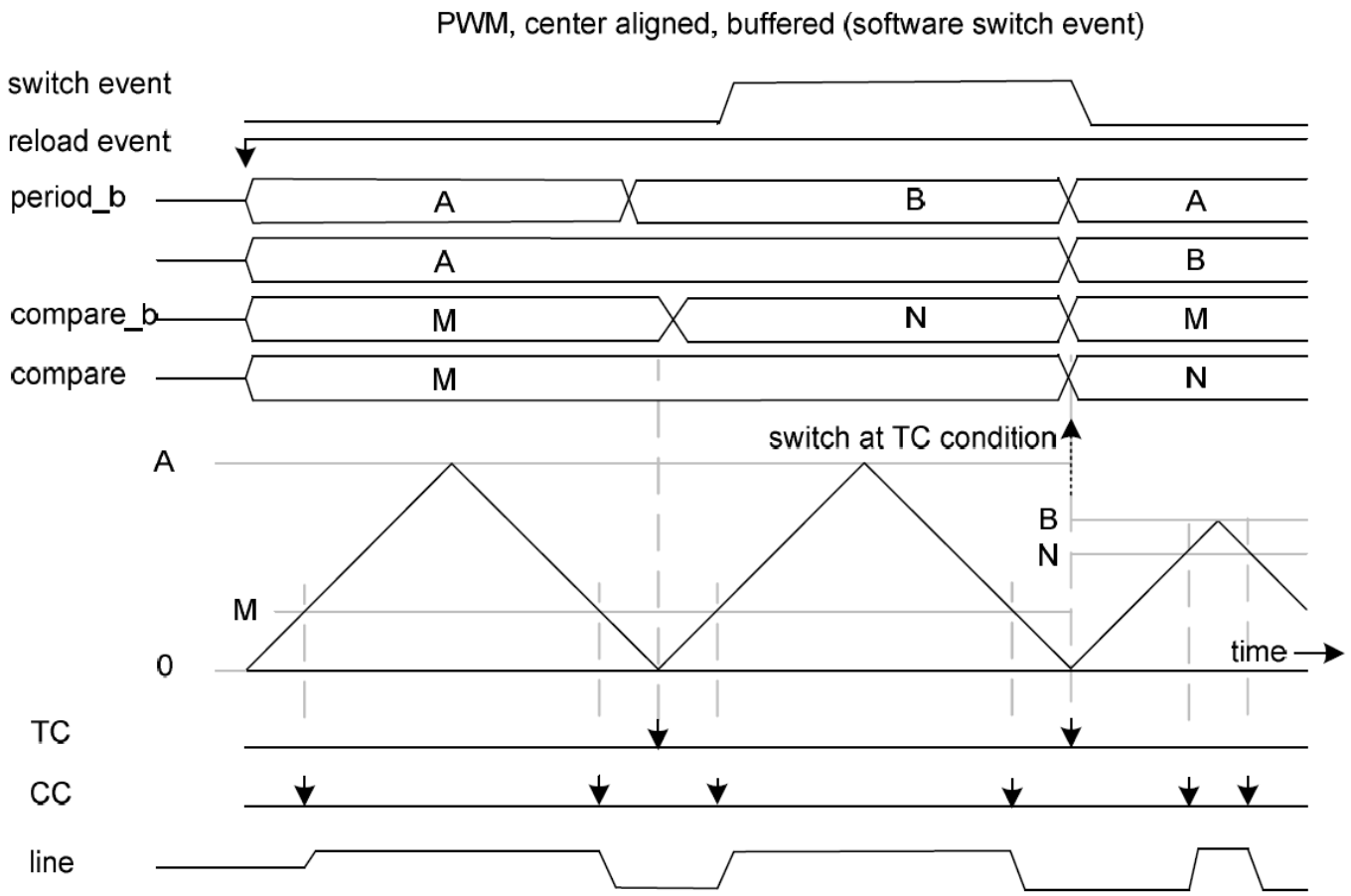
2x 和 4x 正交编码模式的计数速度分别是 1x 正交编码模式的 2 倍和 4 倍。以下两个图显示了正交解码器在 2x 和 4x 编码模式下的操作。注意，使用这些模式时，将实现较高的分辨率。



PWM

脉冲宽度调制的模式可设为脉冲宽度调制、带死区时间插入的脉冲宽度调制，或伪随机脉冲宽度调制模式。默认设置为左对齐并提供直接输出的脉冲宽度调制模式。

当定制器选择了 swap（交换）复选框并触发了切换事件时，脉冲宽度调制的周期和比较寄存器可针对终端计数（TC）事件自动进行交换。可用硬件信号或软件切换事件命令来触发切换事件。在上述各个情况下，实际的交换操作将发生在 TC 事件中。下图显示了中心对齐的脉冲宽度调制，包括软件生成的切换事件。仅在更新周期缓冲区和比较缓冲区寄存器之后，软件才生成切换事件。由于第二个脉冲宽度调制脉冲的更新还未完成（在 TC 条件之后），因此将重复第一个脉冲宽度调制脉冲。完成交换后，硬件在 TC 条件下自动清除切换事件。



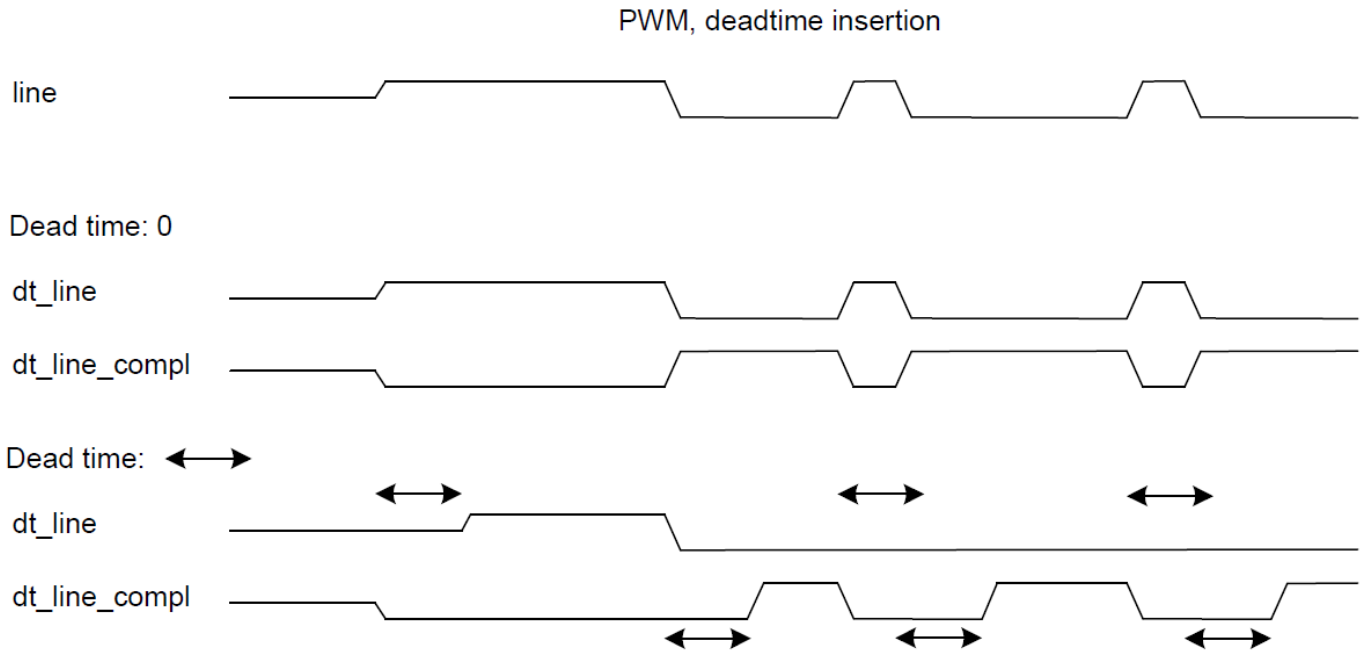
切换/捕获事件功能可用于实现脉冲宽度调制高相位的 DSI 受控调制，使脉冲宽度调制高相位的精度超过单个脉冲宽度调制周期的精度。将通过在多个脉冲宽度调制周期中在两个比较值之间进行调制，实现更高的精度。下例中显示了此目标功能。

模块示例：假定 100 个循环的脉冲宽度调制周期。周期寄存器和缓冲周期寄存器都设置为 99，以实现此 100 循环的脉冲宽度调制周期。在未经调制的情况下，比较值 50 将生成 50 个循环的脉冲宽度调制高相位。100 个循环的脉冲宽度调制周期可实现 1%的脉冲精度。要用相同的 100 循环脉冲宽度调制周期实现更高的精度，需支持两个比较值之间的调制。在一个脉冲宽度调制周期内，精度仍为 1%。但是，在多个脉冲宽度调制周期内，可实现较高的精度。要实现 50.5 循环的脉冲宽度调制高相位（0.5%的精度），应将比较寄存器值设置为 50，将缓冲比较寄存器值设置为 51。在切换/捕获事件中（可能由 DSI 生成），将在比较值和缓冲比较值之间进行切换。当两个比较值都处于活动状态时且为相同的时间量时，可在多个 100 个循环的脉冲宽度调制周期中实现 50.5 个循环的脉冲宽度调制高相位。

带死区时间插入的脉冲宽度调制（PWM_DT）

死区时间插入模式可将死区时间插入到脉冲宽度调制器。下图显示了如何根据脉冲宽度调制输出线路“line”生成互补输出线路“dt_line”和“dt_line_compl”。顶部示例显示不带死区时间的输

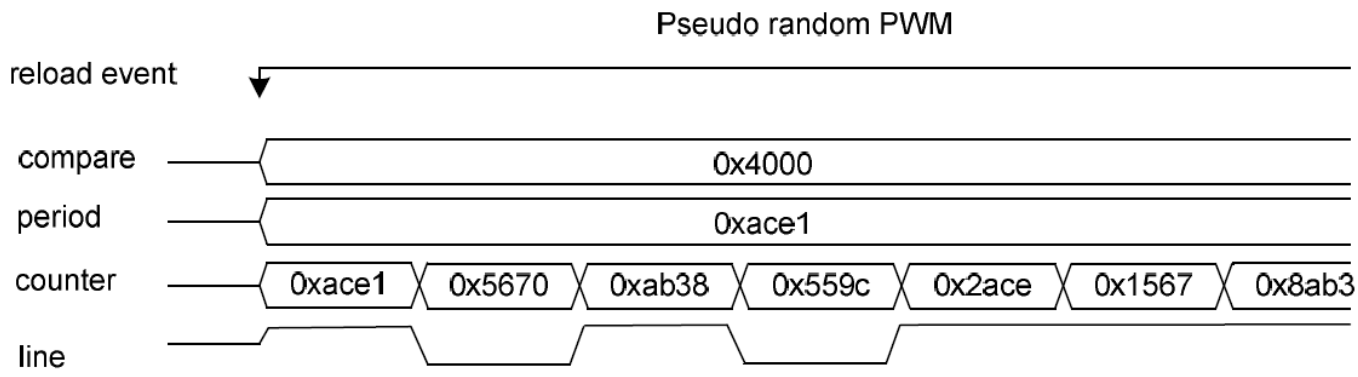
出线路。底部示例显示带较短死区时间的输出。在直接输出信号模式下，上升沿被死区时间延迟，下降沿不会被其延迟。在反转输出信号模式下，下降沿被死区时间延迟，上升沿不会被其延迟。



要根据 DSI 输入信号的值实施“kill”（立即禁用两条输出线路的能力），可选择指定 DSI 输入信号的值作为停止事件。在通过模式下，不会对触发信号执行边沿检测。生成的停止事件用于禁用两条输出线路。

脉冲宽度调制伪随机模式（PWM_PR）

在伪随机模式中，输出线路可在计数器的伪随机值超过比较寄存器中指定的值时进行切换。计数器值在伪随机序列中会出现更改。将计数器值的低 15 位与比较寄存器值相比较，以确定 line 的值。当计数器值的低 15 位小于比较值时，此线路处于高电平，否则处于低电平。下图显示了伪随机噪声行为在比较值为 0x4000 时产生约 50%的占空比。由于 16 位计数器值仅较低的 15 位用于与比较寄存器值进行比较，因此这种情况可能出现。



寄存器

有关寄存器的详细信息，请参见芯片的《技术参考手册》（TRM）。

API 存储器的使用情况

根据不同编译程序、器件、所使用的 API 数量以及组件的配置情况，组件所用的存储空间大小也不一样。下表显示了在指定组件配置中所有可用的 API 的存储器使用情况。

已利用“释放”模式中配置的相关编译器进行了测量，大小采用了优化设定。有关特定的设计，可分析编译器所生成的映射文件以确定内存的使用情况。

配置	PSoC 4 （GCC）	
	闪存（字节）	SRAM（字节）
未配置	1036	5
定时器/计数器	748	5
正交解码器	568	5
PWM	1012	5

直流和交流电的电气特性

除非另有说明，否则这些规范的适用条件是：-40°C ≤ T_A ≤ 85 °C 且 T_J ≤ 100 °C。除非另有说明，否则这些规范的适用范围为 1.71 V 到 5.5 V。

定时器直流规范

参数	说明	最小值	典型值	最大值	单位	条件
I_{TIM1}	3 MHz时的模块电流消耗	–	–	19	µA	16位定时器
I_{TIM2}	12 MHz时的模块电流消耗	–	–	66	µA	16位定时器
I_{TIM3}	48 MHz时的模块电流消耗	–	–	285	µA	16位定时器

定时器交流规范

参数	说明	最小值	典型值	最大值	单位	条件
$T_{TIMFREQ}$	工作频率	–	–	48	MHz	
$T_{CAPWINT}$	捕获脉冲宽度（内部）	42	–	–	ns	
$T_{CAPWEXT}$	捕获脉冲宽度（外部）	42	–	–	ns	
T_{TIMRES}	定时器分辨率	21	–	–	ns	
$T_{TENWIDINT}$	使能脉冲宽度（内部）	42	–	–	ns	
$T_{TENWIDEXT}$	使能脉冲宽度（外部）	42	–	–	ns	
$T_{TIMRESWINT}$	复位脉冲宽度（内部）	42	–	–	ns	
$T_{TIMRESEXT}$	复位脉冲宽度（外部）	42	–	–	ns	

计数器直流规范

参数	说明	最小值	典型值	最大值	单位	条件
I_{CTR1}	3 MHz时的模块电流消耗	–	–	19	µA	16位计数器
I_{CTR2}	12 MHz时的模块电流消耗	–	–	66	µA	16位计数器
I_{CTR3}	48 MHz时的模块电流消耗	–	–	285	µA	16位计数器

计数器交流规范

参数	说明	最小值	典型值	最大值	单位	条件
$T_{CTRFREQ}$	工作频率	–	–	48	MHz	
$T_{CTRPWINT}$	捕获脉冲宽度（内部）	42	–	–	ns	
$T_{CTRPWEXT}$	捕获脉冲宽度（外部）	42	–	–	ns	
T_{CTRES}	计数器分辨率	21	–	–	ns	
$T_{CENWIDINT}$	使能脉冲宽度（内部）	42	–	–	ns	
$T_{CENWIDEXT}$	使能脉冲宽度（外部）	42	–	–	ns	
$T_{CTRRESWINT}$	复位脉冲宽度（内部）	42	–	–	ns	
$T_{CTRRESWEXT}$	复位脉冲宽度（外部）	42	–	–	ns	

PWM 直流电规范

参数	说明	最小值	典型值	最大值	单位	条件
I_{PWM1}	3 MHz时的模块电流消耗	–	–	19	µA	16位PWM
I_{PWM2}	12 MHz时的模块电流消耗	–	–	66	µA	16位PWM
I_{PWM3}	48 MHz时的模块电流消耗	–	–	285	µA	16位PWM

PWM 交流电规范

参数	说明	最小值	典型值	最大值	单位	条件
$T_{PWMFREQ}$	工作频率	–	–	48	MHz	
$T_{PWMPWINT}$	脉冲宽度（内部）	42	–	–	ns	
T_{PWMEXT}	脉冲宽度（外部）	42	–	–	ns	
$T_{PWMKILLINT}$	停止信号脉冲宽度（内部）	42	–	–	ns	
$T_{PWMKILLEXT}$	停止信号（kill）脉冲宽度（外部）	42	–	–	ns	
$T_{PWMEINT}$	使能脉冲宽度（内部）	42	–	–	ns	
$T_{PWMENEXT}$	使能脉冲宽度（外部）	42	–	–	ns	
$T_{PWMRESWINT}$	复位脉冲宽度（内部）	42	–	–	ns	
$T_{PWMRESWEXT}$	复位脉冲宽度（外部）	42	–	–	ns	

组件更改

本节列出了各版本中主要组件的更改内容。

版本	更改说明	更改/影响原因
1.0.a	编辑了数据手册。	更新了用于直接和反转输出信号模式的PWM死区时间功能的说明。 说明了TCPWM_WritePeriod()函数的功能。
1.0	新组件	

赛普拉斯半导体公司，2013-2016 年。本文件是赛普拉斯半导体公司及其子公司，包括 Spansion LLC（“赛普拉斯”）的财产。本文件，包括其包含或引用的任何软件或固件（“软件”），根据全球范围内的知识产权法律以及美国与其他国家签署条约由赛普拉斯所有。除非在本款中另有明确规定，赛普拉斯保留在该等法律和条约下的所有权利，且未就其专利、版权、商标或其他知识产权授予任何许可。如果软件并不附随有一份许可协议且贵方未以其他方式与赛普拉斯签署关于使用软件的书面协议，赛普拉斯特此授予贵方属人性质的、非独家且不可转让的如下许可（无再许可权）：（1）在赛普拉斯特软件著作权项下的下列许可权（一）对以源代码形式提供的软件，仅出于在赛普拉斯硬件产品上使用之目的且仅在贵方集团内部修改和复制软件，和（二）仅限于在有关赛普拉斯硬件产品上使用之目的将软件以二进制代码形式的向外部最终用户提供（无论直接提供或通过经销商和分销商间接提供），和（2）在被软件（由赛普拉斯公司提供，且未经修改）侵犯的赛普拉斯专利的权利主张项下，仅出于在赛普拉斯硬件产品上使用之目的制造、使用、提供和进口软件的许可。禁止对软件的任何其他使用、复制、修改、翻译或汇编。

在适用法律允许的限度内，赛普拉斯未对本文件或任何软件作出任何明示或暗示的担保，包括但不限于关于适销性和特定用途的默示保证。赛普拉斯保留更改本文件的权利，届时将不另行通知。在适用法律允许的限度内，赛普拉斯不对因应用或使用本文件所述任何产品或电路引起的任何后果负责。本文件，包括任何样本设计信息或程序代码信息，仅为供参考之目的提供。文件使用人应负责正确设计、计划和测试信息应用和由此生产的任何产品的功能和安全性。赛普拉斯产品不应被设计为、设定为或授权用作武器操作、武器系统、核设施、生命支持设备或系统、其他医疗设备或系统（包括急救设备和手术植入物）、污染控制或有害物质管理系统中的关键部件，或产品植入之设备或系统故障可能导致人身伤害、死亡或财产损失其他用途（“非预期用途”）。关键部件指，若该部件发生故障，经合理预期会导致设备或系统故障或会影响设备或系统安全性和有效性的部件。针对由赛普拉斯产品非预期用途产生或相关的任何主张、费用、损失和其他责任，赛普拉斯不承担全部或部分责任且贵方不应追究赛普拉斯之责任。贵方应赔偿赛普拉斯因赛普拉斯产品任何非预期用途产生或相关的所有索赔、费用、损失和其他责任，包括因人身伤害或死亡引起的主张，并使之免受损失。

赛普拉斯、赛普拉斯徽标、Spansion、Spansion 徽标，及上述项目的组合，及 PSoC、CapSense、EZ-USB、F-RAM 和 Traveo 应视为赛普拉斯在美国和其他国家的商标或注册商标。请访问 cypress.com 获取赛普拉斯商标的完整列表。其他名称和品牌可能由其各自所有者主张为该方财产。