



增强型脉宽调制模块ePWM



增强型脉宽调制模块ePWM

增强型脉宽调制（ePWM）模块的作用是产生频率、相位和占空比可调的方波脉冲，是TMS320F28335的重要外设，在电机驱动控制和电力电子的设备中是必不可少的功能，可以应用于比如数字式电机控制系统、数字电源、变频器、逆变器、电动汽车充电桩、储能变流器等电力变换设备中。本章将详细介绍ePWM的结构、内部的各个子模块，并结合实际应用来介绍如何使用ePWM模块来产生所需的各种PWM波形。



12.1 PWM基础知识

PWM是Pulse Width Modulation的缩写，即脉宽调制，通俗点讲就是宽度可调节的方波脉冲，如图12-1所示。

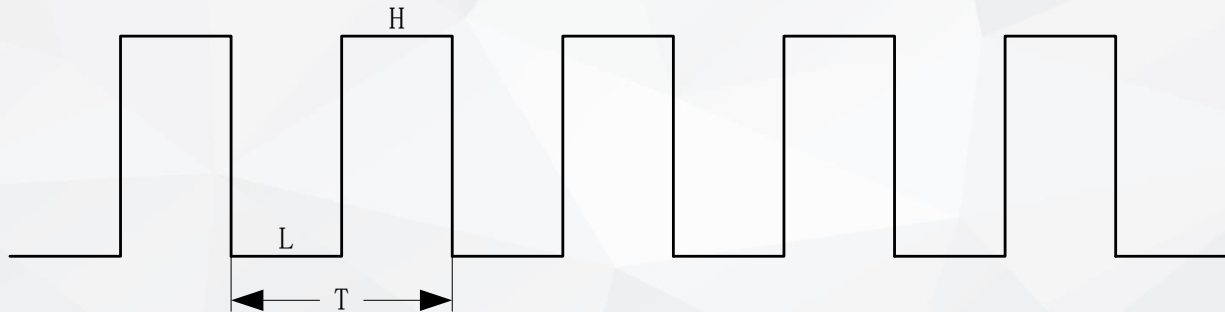


图12-1 PWM波形



12.1 PWM基础知识

在实际应用中，PWM用于驱动开关器件，PWM输出的高低电平刚好可以控制开关器件的导通或者关断，从而实现通过改变输出方波的占空比来改变等效的输出电压。图12-2是电力电子中最为常见和实用的一种拓扑结构，通过六路PWM来控制六个开关管，可以将直流电压 U_d 逆变成对称的三相交流电 U 、 V 、 W 。

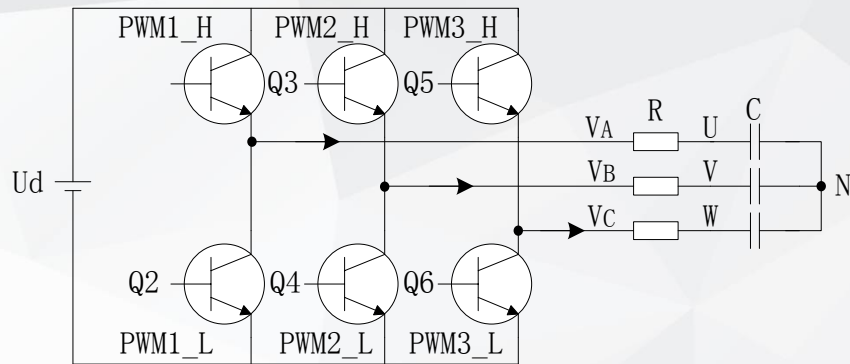


图12-2 PWM控制的三相全桥逆变电路



12.1 PWM基础知识

PWM相关的参数有频率、占空比、幅值。PWM的频率等于其周期的倒数，即：

$$f = \frac{1}{T}$$

T为PWM的周期，即每隔多长时间输出一次脉冲。占空比D为一个周期内，高电平时间与周期的比值，即高电平所占周期的比例，图12-1中的PWM波形的占空比为：

$$D = \frac{H}{T}$$

PWM的幅值是指输出波形的高电平与低电平的电压值。



12.2 ePWM模块概述

TMS320F28335有6个ePWM模块，ePWM1、ePWM2、ePWM3、ePWM4、ePWM5和ePWM6。每个ePWM模块都有相同的内部逻辑电路，因此在功能上这6个ePWM模块都是相同的。图12-3为ePWM模块内部结构框图。从图12-3可以看出，ePWM模块内部包含有7个子模块，分别是时间基准子模块TB、比较功能子模块CC，动作限定子模块AQ、死区控制子模块DB、斩波控制子模块PC、事件触发子模块ET和故障捕获子模块TZ，正是由这几个子模块的配合，才可以方便地得到所需的PWM波形。



12.2 ePWM模块概述

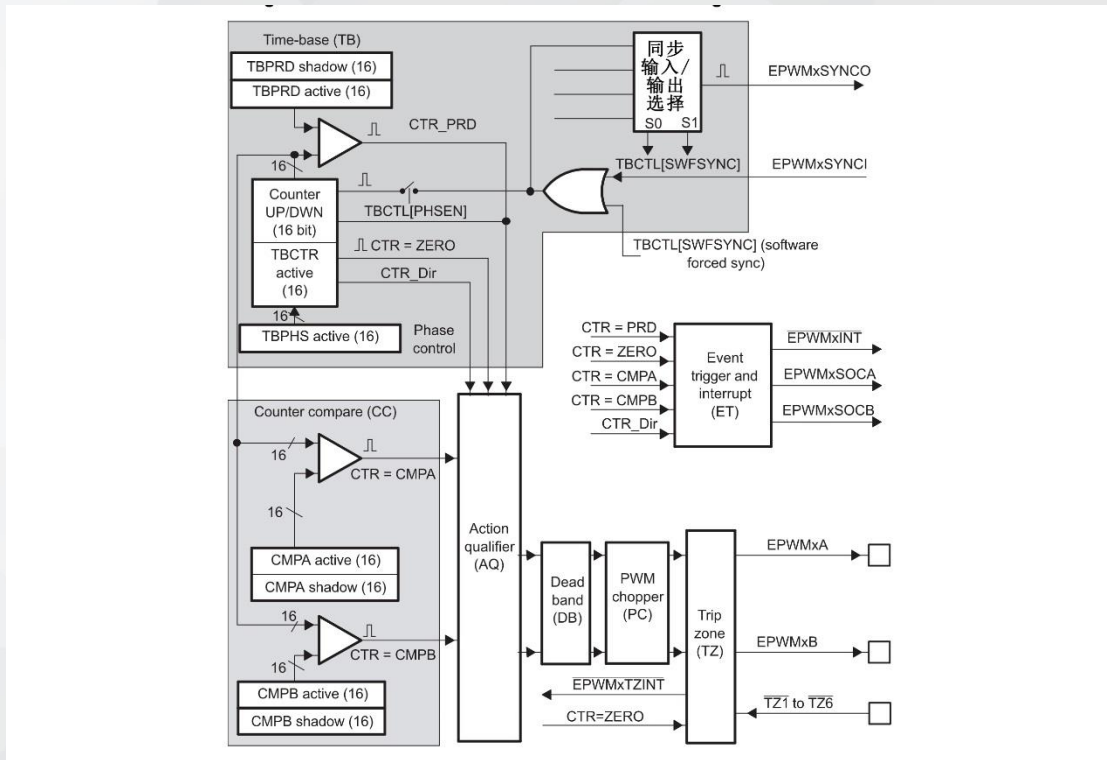


图12-3 ePWM模块内部结构框图



12.2 ePWM模块概述

每个ePWM模块都具有以下功能：

可以输出两路PWM，EPWMxA和EPWMxB；

两路PWM可以独立输出，也可以互补输出；

具有相位控制功能，可以超前或者滞后于其他ePWM模块；

具有死区控制功能，可分别对上升沿和下降沿进行延时控制；

具有故障保护功能，通过对触发条件的设置，当故障发生时，可自动将PWM输出引脚设置为低电平、高电平或高阻状态；

具有高频斩波功能，高频斩波信号对PWM进行斩波控制，用于高频变换器的门极驱动；

所有事件都可以触发中断，也都可以产生内部ADC转换的启动脉冲SOC。



12.2 ePWM模块概述

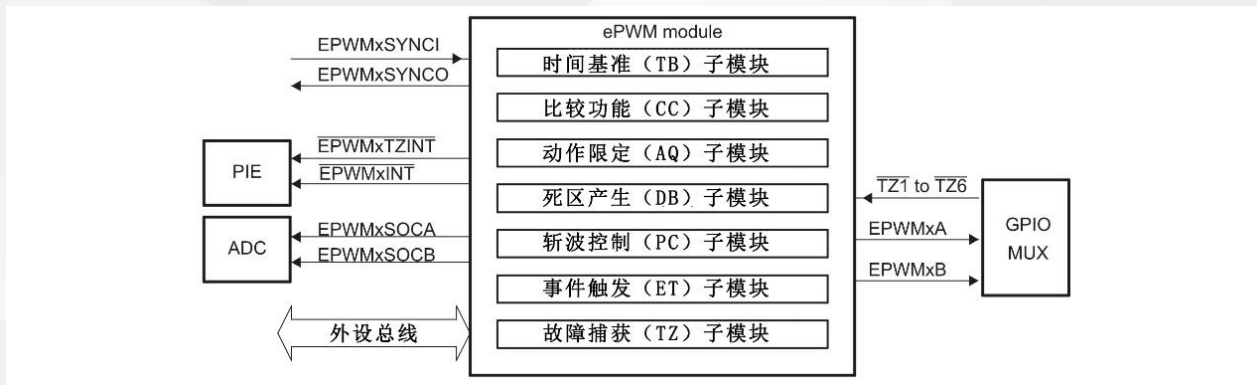


图12-4 ePWM的子模块及主要信号



12.2 ePWM模块概述

从图12-4可以看出：

PWM输出信号EPWMxA和EPWMxB，这两个信号通过GPIO口输出，从而产生PWM波；

故障触发信号 $\overline{TZ1} \sim \overline{TZ6}$ 。这些故障触发信号是用来通知ePWM模块，外部电路出现了故障，需要立即停机，从而实现保护功能。每个ePWM模块都可以使用或屏蔽掉故障触发信号。 $\overline{TZ1} \sim \overline{TZ6}$ 可以配置成同步输入模式，并从相应的GPIO口输入。

时钟基准同步信号输入EPWMxSYNCl及输出EPWMxSYNCO，同步信号可以将所有的ePWM模块连接成一个整体，当然，每个ePWM模块都可以通过设置，使用或者忽略同步信号。

ADC启动信号EPWMxSOCA和EPWMxSOCB。



12.3 ePWM的子模块

在开始介绍ePWM具体的功能前，先来介绍下ePWM产生PWM的基本原理，如图12-5所示。

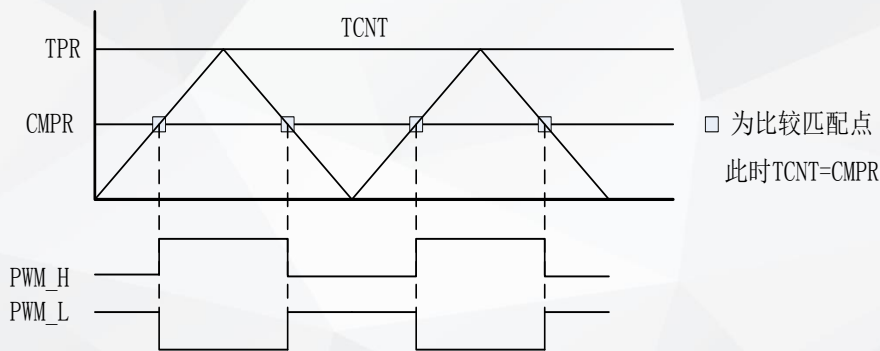


图12-5 ePWM产生PWM的基本原理



12.3 ePWM的子模块

从图12-5可以看出需要有三个寄存器，周期寄存器TPR，计数器寄存器TCNT和比较寄存器CMPR。周期寄存器TPR决定了一个周期计数的最大值，也就是决定了PWM的周期。计数器寄存器TCNT按照时钟信号来进行计数，图12-5位增减计数模式，即从0增计数到TPR，然后再从TPR减计数到0，不断重复。当计数器寄存器TCNT的值与比较寄存器的值CMPR相等时，PWM的电平发生变化，由低电平变为高电平，或者由高电平变为低电平，从而产生周期性的PWM波形。改变周期寄存器TPR，就可以改变PWM的周期，即可以改变PWM的频率；改变比较寄存器CMPR，就可以改变PWM的占空比。上述就是产生PWM波形的基本原理。下面详细介绍ePWM的各个子模块。



12.3.1 时间基准子模块TB

时间基准 (Time Base) 子模块主要有两个作用，一个是时钟信号的同步，另一个是计数。如果是单个ePWM模块，那就不存在同步的问题，自己自行根据时钟进行计数就可以了，然而实际应用中，往往需要多个ePWM模块产生的PWM同时去驱动一个电路，这就会涉及到同步问题，也就是如何让多个ePWM同时进行计数呢，时间基准子模块TB提供了同步信号来解决这个问题。图12-6为时间基准子模块TB在整个ePWM模块中的位置。



12.3.1 时间基准子模块TB

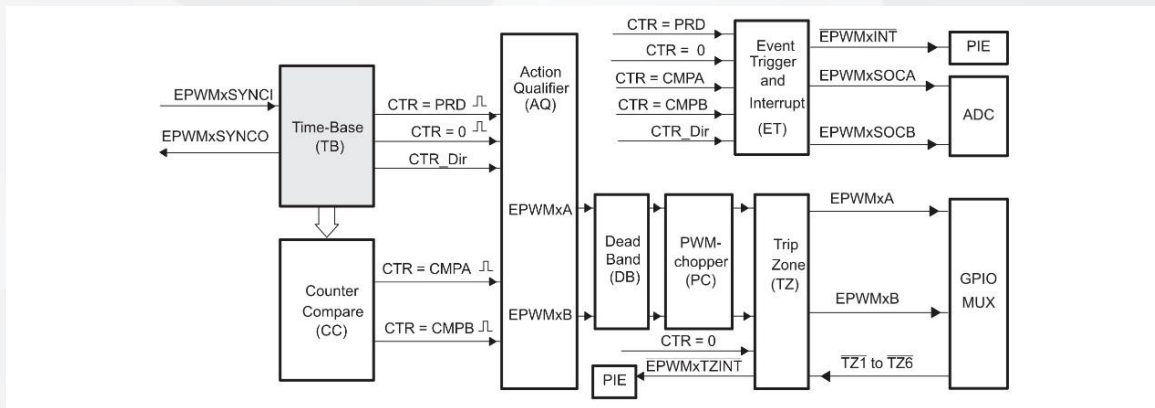


图12-6 时间基准子模块TB在ePWM中的位置



12.3.1 时间基准子模块TB

1. TB子模块内部结构

图12-7为TB子模块内部关键信号及主要寄存器。

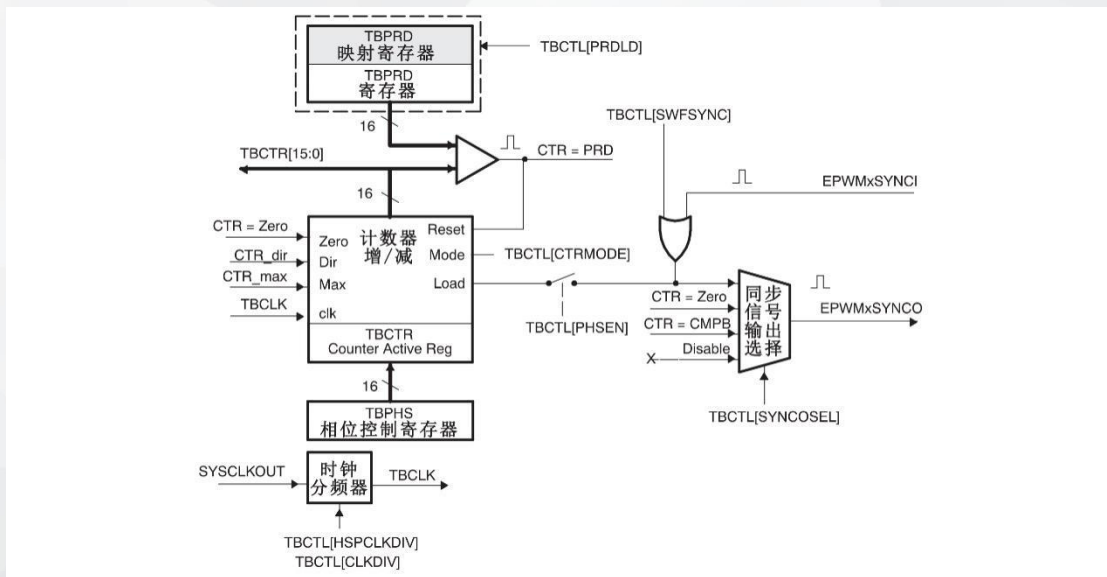


图12- 7 时间基准子模块TB



12.3.1 时间基准子模块TB

通过时间基准子模块TB可以实现计数时钟的配置、计数模式的选择、同步信号的选择、相位的控制等功能，下面会分别进行介绍。表12-1为时间基准子模块TB相关的寄存器，寄存器具体内容可以参见C2000助手软件。

寄存器名	地址偏移	是否具有映射功能	说明
TBCTL	0x0000	NO	TB控制寄存器
TBSTS	0x0001	NO	TB状态寄存器
TBPHSHR	0x0002	NO	HRPWM相位扩展寄存器
TBPHS	0x0003	NO	TB相位寄存器
TBCTR	0x0004	NO	TB计数寄存器
TBPRD	0x0005	YES	TB周期寄存器

表12-1 时间基准子模块TB的寄存器



12.3.1 时间基准子模块TB

2.计数时钟

计数器计数需要一个计数的节拍来进行计数，也就是它需要按照一定的时间来进行一次计数。时间基准子模块TB的计数时钟TBCLK是由系统时钟SYSCLKOUT分频而来，从图12-5可以看出，它和控制寄存器TBCTL的两个位有关，HSPCLKDIV和CLKDIV。如果HSPCLKDIV的值为x，则如果x为0，则分频系数为1，如果x不为0，则分频系数为2x；如果CLKDIV的值为y，则分频系数为 2^y 。计数时钟TBCLK的计算公式为：

$$TBCLK = \frac{SYSCLKOUT}{2^y} \quad (x = 0)$$

$$TBCLK = \frac{SYSCLKOUT}{2x \times 2^y} \quad (x > 0)$$

对于TMS320F23335而言，SYSCLKOUT为150MHz，如果HSPCLKDIV=0，CLKDIV=2，则TBCLK=37.5MHz。下面的例子均以这个时钟频率为例进行示范。



12.3.1 时间基准子模块TB

时间基准子模块的计数器一共有三种计数模式，增计数、减计数和增减计数。图12-8为时间基准子模块的三种计数方式。

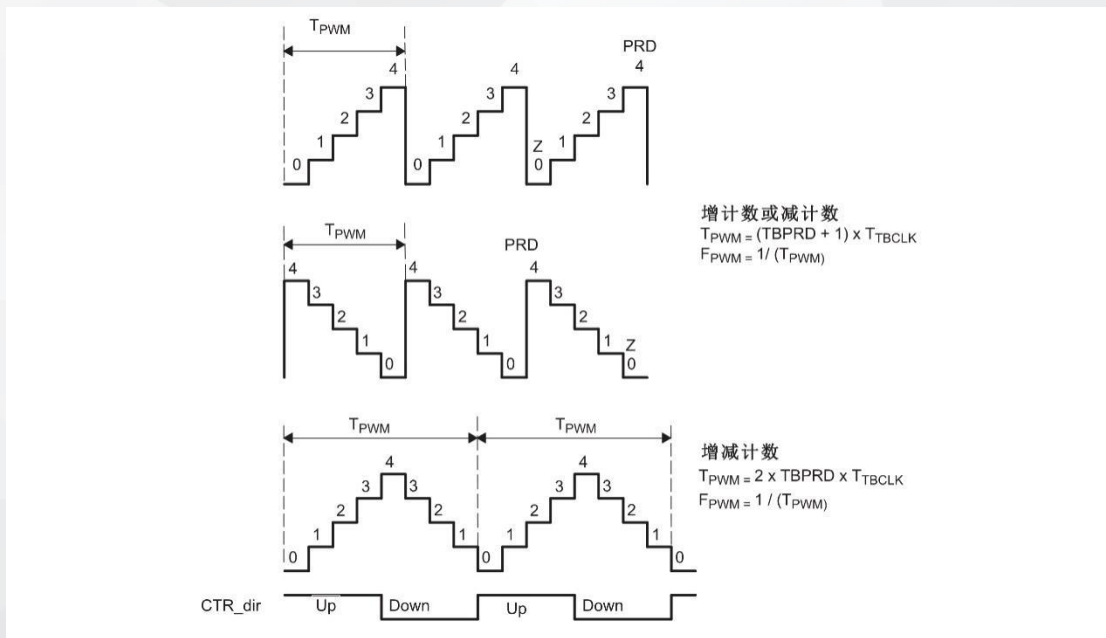


图12-8 时间基准子模块的计数方式



12.3.1 时间基准子模块TB

增计数模式：计数器TBCTR从0开始增计数，每次加1，计数到TBPRD时，TBCTR变为0，然后重新开始增计数至TBPRD，不断重复。

减计数模式：计数器TBCTR从TBPRD开始减计数，每次减1，计数到0时，TBCTR又变为TBPRD，然后重新开始减计数至0，不断重复。

增减计数模式：计数器TBCTR从0开始增计数，每次加1，计数到TBPRD时，进行减计数，每次减1，计数到0时再开始增计数，不断重复。



12.3.1 时间基准子模块TB

图12-8是以TBPRD=4为例演示的三种计数模式。由前面PWM产生的原理可以看出，周期寄存器TBPRD决定了PWM的周期，从图12-8不难得到，当计数器工作于增计数模式或者减计数模式时，PWM的计数周期为：

$$T_{PWM} = TBPRD + 1$$

当计数器工作于增减计数模式时，PWM的计数周期为：

$$T_{PWM} = 2 \times TBPRD$$



12.3.1 时间基准子模块TB

而每计一次数所需要的时间是由计数时钟来决定对的，如果计数时钟为TBCLK为X MHz，则每计一次数需要时间为：

$$t_{CLK} = \frac{1}{X \times 10^6} S$$

因此，PWM的周期为：

$$T = T_{PWM} \times t_{CLK}$$

得到PWM的周期，PWM的频率也就得到了，取倒数就可以。



12.3.1 时间基准子模块TB

3.映射寄存器

从表12-1可以看到，时间基准周期寄存器TBPRD具有一个映射寄存器，映射寄存器可以使寄存器的更新与硬件同步。ePWM所有具有映射寄存器的寄存器都会有两个寄存器，分别是当前寄存器和映射寄存器。

当前寄存器可以用来控制系统硬件的运行，并反映硬件的当前状态。映射寄存器可以用来临时存放数据，并在某个特定的时刻将数据传送给当前寄存器，可见映射寄存器对硬件没有任何直接作用。



12.3.1 时间基准子模块TB

映射寄存器和当前寄存器拥有相同的地址，TBCTL[PRDL D]位决定了是否使用TBPRD的映射寄存器功能，从而决定了CPU读写操作作用于当前寄存器还是映射寄存器。

1.TBPTD映射模式。当TBCTL[PRDL D]=0时，TBPRD使用映射模式，此时CPU读写TBPRD的地址单元将直接作用于映射寄存器。当计数器TBCTR的值等于0时，映射寄存器中的内容直接装载到当前寄存器。默认情况下TBPRD采用映射模式。

2.TBPRD立即模式。当TBCTL[PRDL D]=1时，TBPRD使用立即模式，此时CPU读写TBPRD的地址单元时将绕开映射寄存器，而直接作用于当前寄存器。



12.3.1 时间基准子模块TB

4.时钟同步和相位控制

多个ePWM模块一起工作时，往往会涉及输出PWM的同步问题，即如何让多个ePWM模块同步进行计数，换句话说通过同步可以将器件内所有的ePWM模块连在一起。TMS320F28335的每一个ePWM都有一个同步信号输入EPWMxSYNCl和一个同步信号输出EPWMxSYNCO。TMS320F28335的同步方案如图12-9所示。



12.3.1 时间基准子模块TB

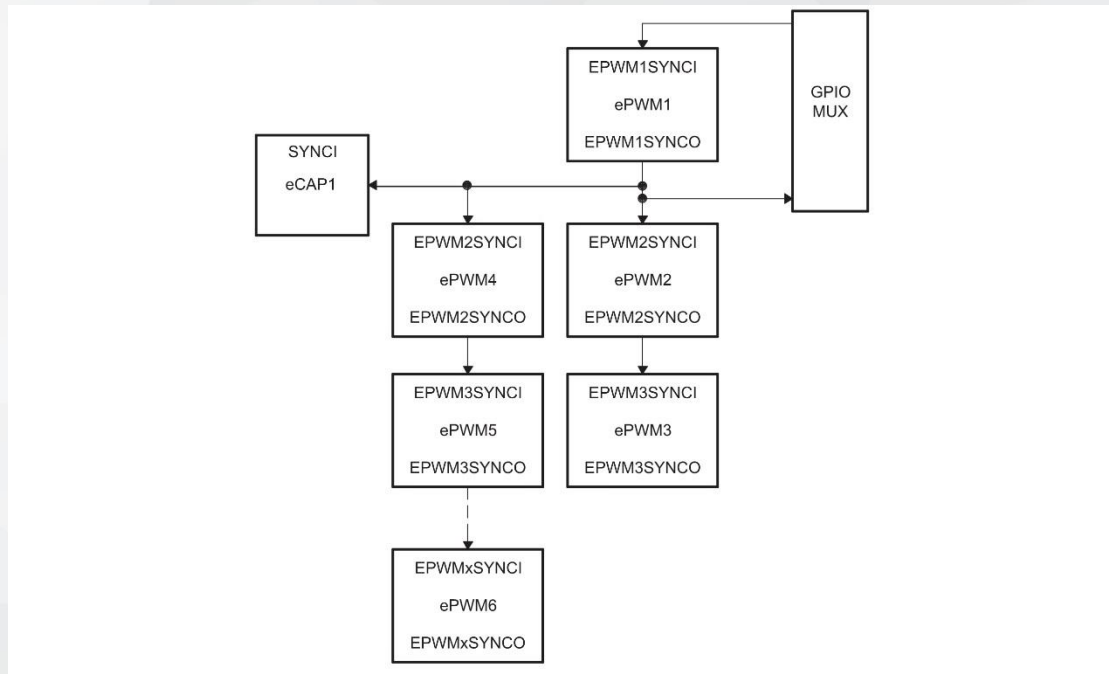


图12-9 ePWM模块的时钟同步方案



12.3.1 时间基准子模块TB

ePWM1模块的同步信号输入来自于外部引脚，然后ePWM1将同步信号输出给ePWM2和ePWM4，其他模块的同步信号输入输出关系见图12-9。每个ePWM模块都可以使用或者忽略同步信号。

在实际使用时，PWM信号之间往往会有相位的差别，ePWM的时间基准子模块可以通过TBCTL[PHSEN]位来实现相位控制功能。如果TBCTL[PHSEN]=1，那么相应的ePWM模块的时间基准计数器TBCTR将在以下情况发生时自动装载相位寄存器TBPHS中的值。



12.3.1 时间基准子模块TB

1. 同步脉冲EPWMxSYNCCI输入时，即当同步脉冲信号EPWMxSYNCCI被检测到时，相位寄存器TBPHS中的值将被装载到时间基准计数器TBCTR中，装载过程发生在下一个时间基准时钟TBCLK的上升沿。如果TBCLK=SYSCLKOUT，那么将产生两个TBCLK周期的延时；如果TBCLK≠SYSCLKOUT，那么将产生一个TBCLK周期的延时。

2. 软件强制同步脉冲产生时，即当向TBCTL[SWFSYNC]位中写1时，相当于使用软件强制的方式产生一个同步脉冲，而软件产生的同步脉冲与EPWMxSYNCCI具有相同的作用。



12.3.1 时间基准子模块TB

相位控制功能可以方便地控制各个ePWM模块所产生的PWM脉冲之间的相位关系，可控制一路PWM脉冲的相位超前、滞后或与另一路PWM脉冲同步。在增减计数模式下，TBCTL[PSHDIR]位控制同步事件发生后时间基准计数器TBCTR的计数方向，新的计数方向与同步事件之前的计数方向无关。在增计数或减计数模式下，PHSDIR位被忽略。

图12-10为增计数模式下相位控制的波形。当同步事件发生时，TBCTR会装载TBPHS中的值并重新开始进行增计数。



12.3.1 时间基准子模块TB

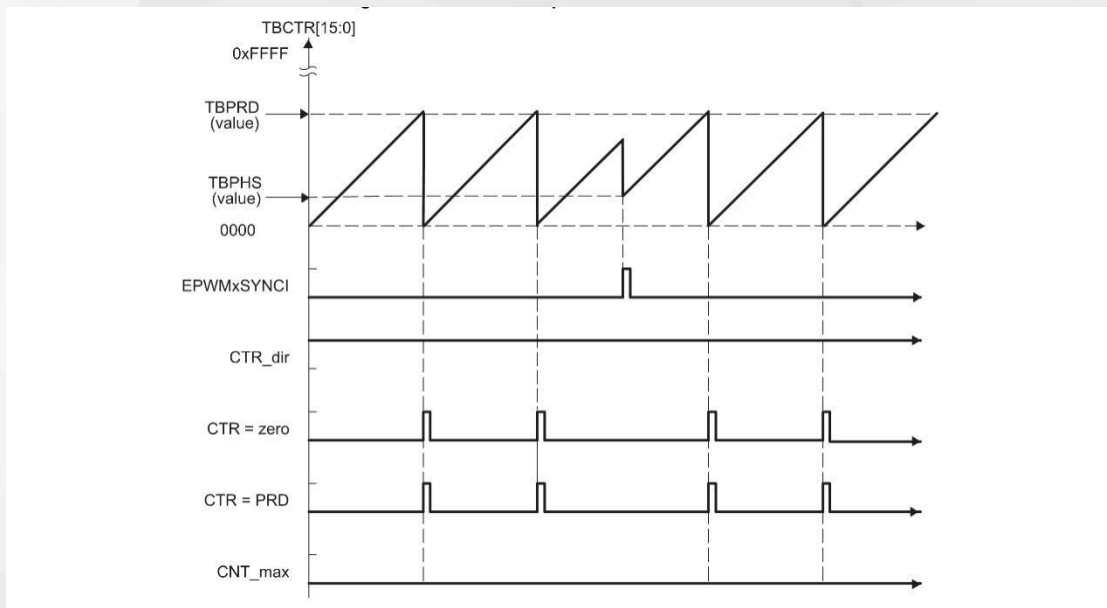


图12-10 增计数模式下相位控制的波形



12.3.1 时间基准子模块TB

图12-11为减计数模式下相位控制的波形。当同步事件发生时，TBCTR会装载TBPHS中的值并重新开始进行减计数。

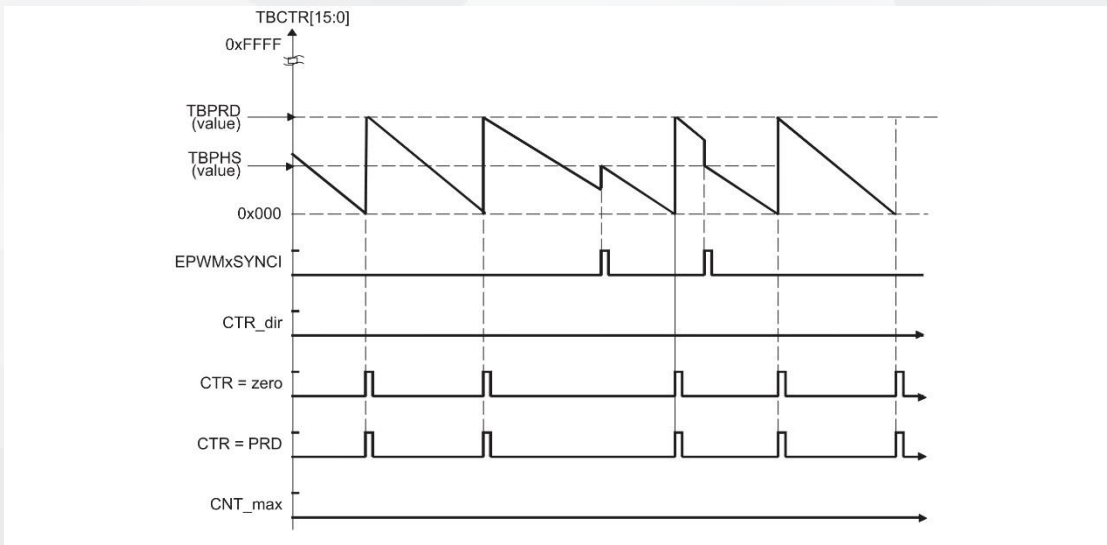


图12-11 减计数模式下相位控制的波形



12.3.1 时间基准子模块TB

图12-12为增减计数模式下，TBCTR[PHSDIR]=0时的相位控制波形。当同步事件发生时，TBCTR会装载TBPHS中的值，并进行减计数。同步事件后进行减计数，和同步事件发生前的计数方向没有关系。

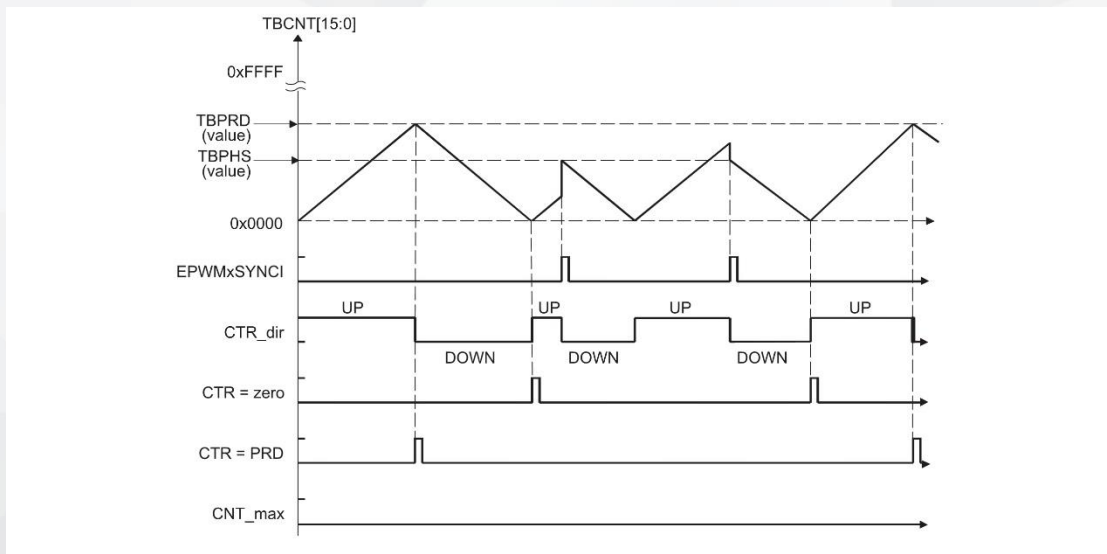


图12-12 增减计数模式下，TBCTR[PHSDIR]=0时的相位控制波形



12.3.1 时间基准子模块TB

图12-13为增减计数模式下，TBCTR[PHSDIR]=1时的相位控制波形。当同步事件发生时，TBCTR会装载TBPHS中的值，并进行增计数。同步事件后进行增计数，和同步事件发生前的计数方向没有关系。

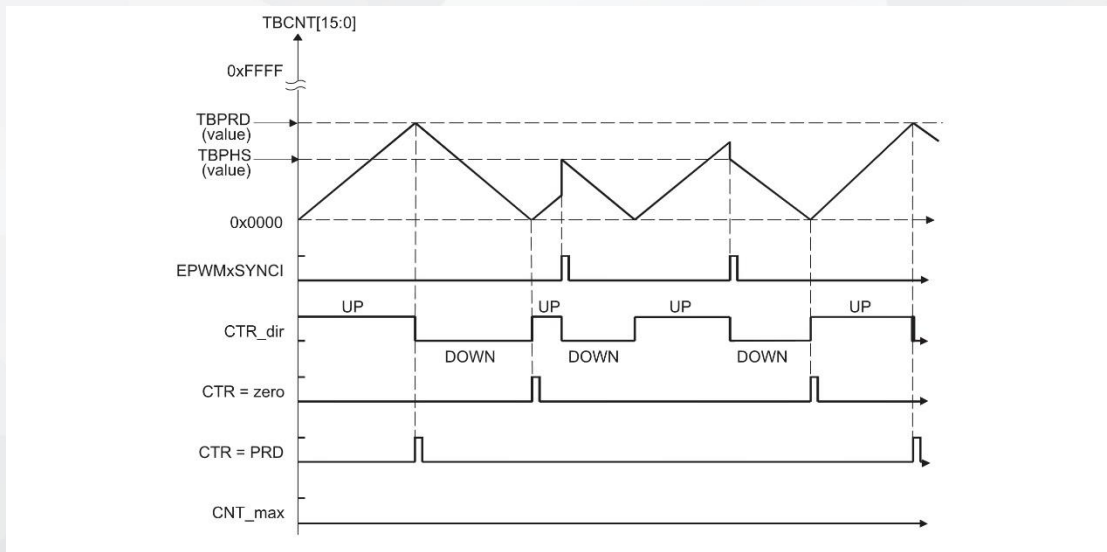


图12-13 增减计数模式下，TBCTR[PHSDIR]=1时的相位控制波形



12.3.1 时间基准子模块TB

5.软件同步多个ePWM模块的基准时钟

如果有多个ePWM模块的基准时钟被使能，那时间基准控制寄存器TBCTL[TBCLKSYNC]位可以用来同步这些基准时钟。当TBCTL[TBCLKSYNC]=0时，所有ePWM模块的时钟停止（默认）；当TBCTL[TBCLKSYNC]=1时，所有ePWM模块的时钟在TBCLK的上升沿启动。在初始化ePWM模块时，需要按照以下步骤进行操作：

- 1)使能各个ePWM模块的时钟；
- 2)将TBCLKSYNC清零，从而停止所有ePWM模块的时钟；
- 3)对ePWM模块进行配置；
- 4)将TBCLKSYNC置位，同时启动所有ePWM模块的时钟。



12.3.2 比较功能子模块CC

计数器比较功能 (Counter Compare) 子模块有两个比较寄存器 CMPA和CMPB，其功能就是将计数器寄存器TBCTR的值和这两个比较寄存器的值进行比较，由此产生比较事件，从而产生PWM波。图12-14可以看出比较功能子模块CC在整个ePWM模块中的位置。

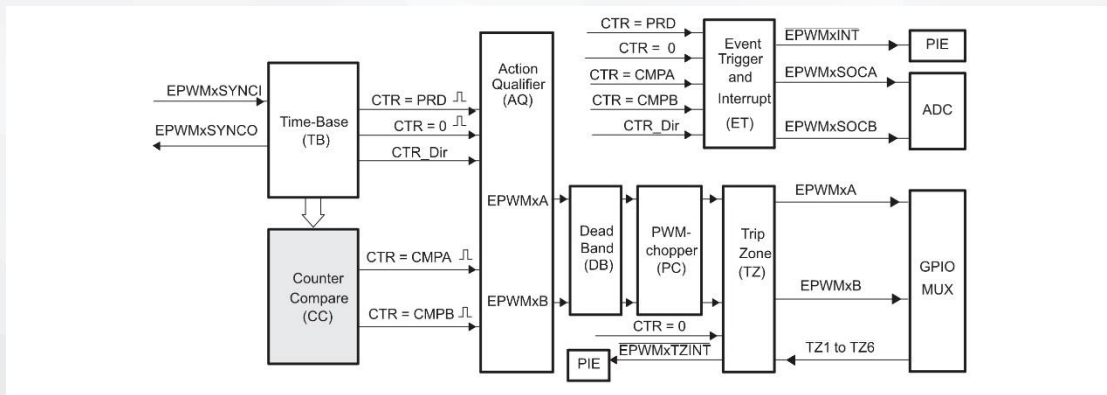


图12-14 比较功能子模块CC在ePWM中的位置



12.3.2 比较功能子模块CC

比较功能子模块相关的寄存器如表12-2所示。比较功能子模块内部信号和寄存器如图12-15所示。

寄存器名	地址偏移	是否具有映射功能	说明
CMPCTL	0x0007	NO	比较器控制寄存器
CMPAHR	0x0008	YES	HRPWM CMPA扩展寄存器
CMPA	0x0009	YES	比较寄存器A
CMPB	0x000A	YES	比较寄存器B

表12-2 比较功能子模块CC的寄存器



12.3.2 比较功能子模块CC

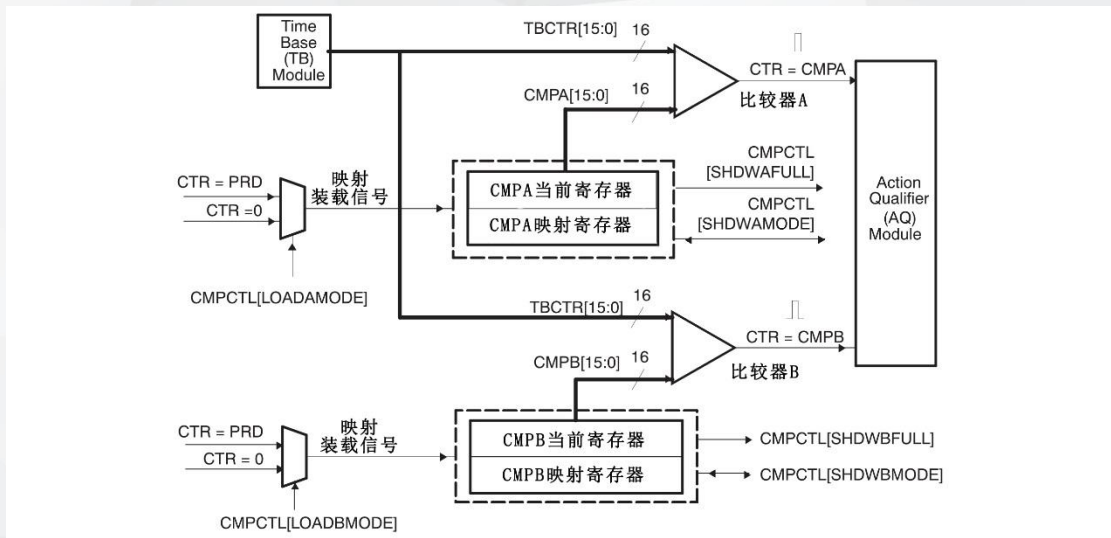


图12-15 比较功能子模块内部信号和寄存器



12.3.2 比较功能子模块CC

如图12-15所示，当计数器寄存器TBCTR的值与比较寄存器A的值相等时，会产生CTR=CMPEA事件；当计数器寄存器TBCTR的值与比较寄存器B的值相等时，会产生CTR=CMPEB事件。对于增计数和减计数模式，比较事件在一个计数周期内出现一次。对于增减计数模式，如果比较值在0~TBPRD的范围，则比较事件在一个计数周期内出现两次，如果比较值等于0或者TBPRD，则比较事件在一个计数周期内只出现一次。



12.3.2 比较功能子模块CC

比较寄存器CMPA和CMPB都有相应的映射寄存器，CMPA是否启用映射寄存器由CMPCTL[SHDWAMODE]决定，CMPB是否启用映射寄存器由CMPCTL[SHDWAMODE]决定。如果启用了映射寄存器，CMPA和CMPB工作于映射模式，可以通过CMPCTL[LOADAMODE]和CMPCTL[LOADBMODE]选择何时将映射寄存器中的内容装载进当前寄存器中，可以有三种选择：

TBCTR=0时，也就是当计数器寄存器计数到0时，将映射寄存器中的值装载进当前寄存器中。

TBCTR=TBPRD时，也就是当计数器寄存器计数到TBPRD时，将映射寄存器中的值装载进当前寄存器中。

TBCTR=0或TBCTR=TBPRD时，也就是当计数器寄存器计数到0或TBPRD时，将映射寄存器中的值装载进当前寄存器中。

当然，如果选择CMPA和CMPB工作于立即模式，则只要值有更新，就会将CMPA和CMPB的值直接写进当前寄存器中，比较值立即发生更新。



12.3.2 比较功能子模块CC

计数器比较功能 (Counter Compare) 子模块有两个比较寄存器 CMPA和CMPB，其功能就是将计数器寄存器TBCTR的值和这两个比较寄存器的值进行比较，由此产生比较事件，从而产生PWM波。图12-14可以看出比较功能子模块CC在整个ePWM模块中的位置。

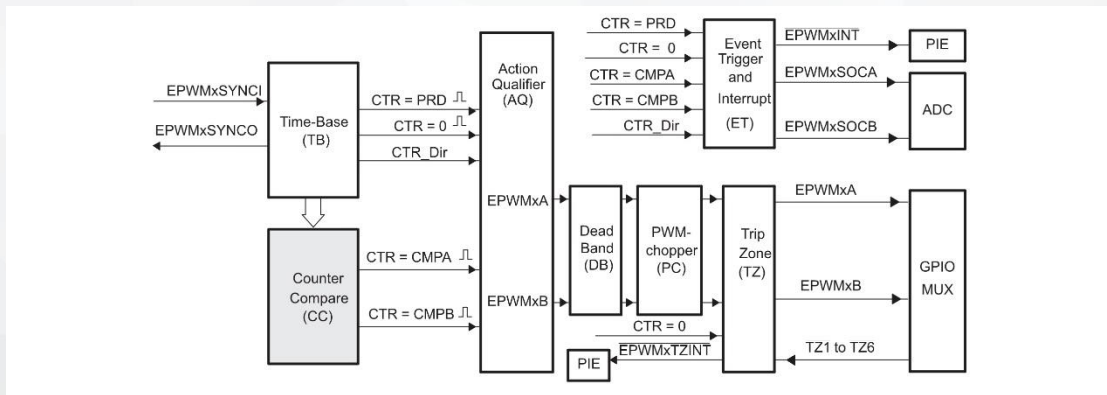


图12-14 比较功能子模块CC在ePWM中的位置



12.3.2 比较功能子模块CC

图12-16为增计数模式下比较事件产生的情况。在增计数模式下，同步信号到来，TBCTR会将相位寄存器TBPHS的值装载进来，然后从这个值开始增计数。图12-16中第二个周期的时候，由于同步信号的到来，使得原本会发生的事件 $CTR = CMPA$ 的事件丢失了， $CTR = CMPB$ 的事件提前了。

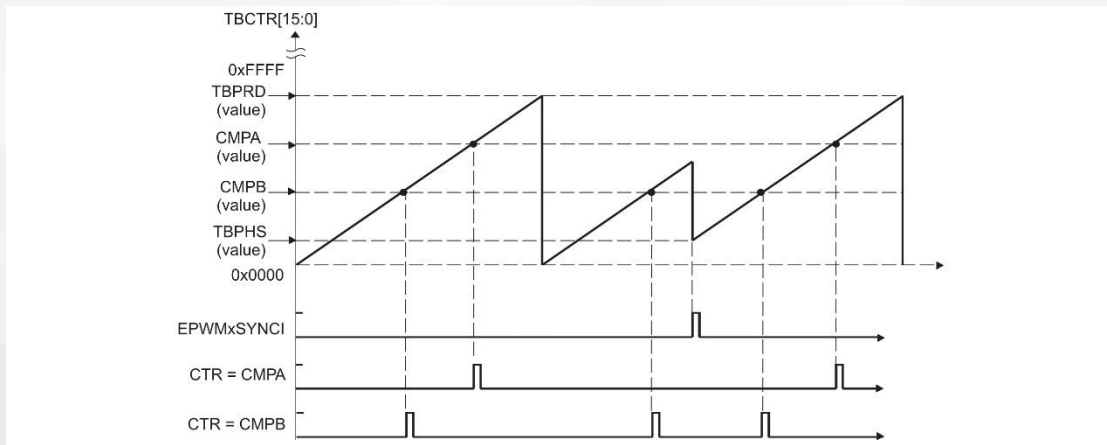


图12-16 增计数模式下比较事件产生情况



12.3.2 比较功能子模块CC

图12-17为减计数模式下比较事件产生的情况。在减计数模式下，同步信号到来，TBCTR会将相位寄存器TBPHS的值装载进来，然后从这个值开始减计数。图12-17中第三个周期的时候，由于同步信号的到来，使得原本会发生的事件 $CTR = CMPB$ 的事件丢失了， $CTR = CMPA$ 的事件提前了。

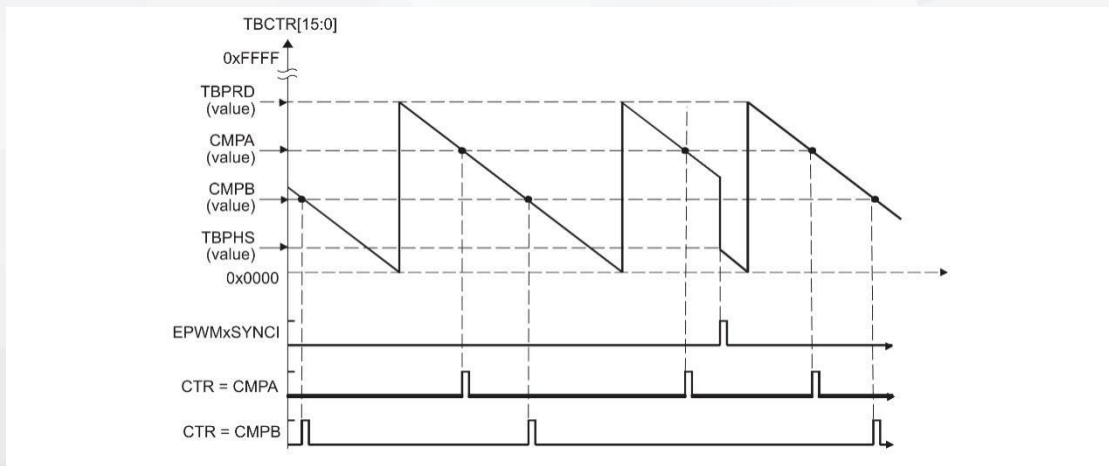


图12-17 减计数模式下比较事件产生情况



12.3.2 比较功能子模块CC

图12-18为增减计数模式，且TBCTL[PHSDIR]=0情况下比较事件产生的情况。在同步事件到来时，TBCTR将TBPHS的值装载进来，并从这个值开始减计数，而不管先前的计数方向。图12-18中，同步事件的到来并没有使得比较事件丢失，但是比较事件发生的时刻都提前了。

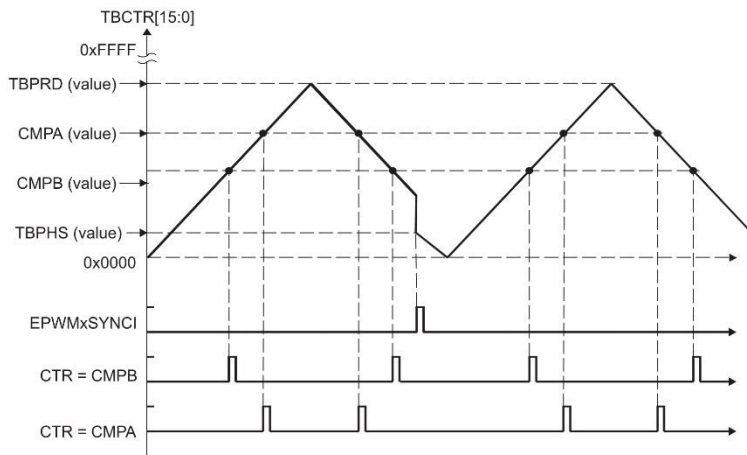


图12-18 增减计数模式下比较事件产生情况



12.3.2 比较功能子模块CC

图12-19为增减计数模式，且TBCTL[PHSDIR]=1情况下比较事件产生的情况。在同步事件到来时，TBCTR将TBPHS的值装载进来，并从这个值开始减计数，而不管先前的计数方向。图12-19中，同步事件的到来并没有使得比较事件丢失，但是比较事件发生的时刻都提前了。

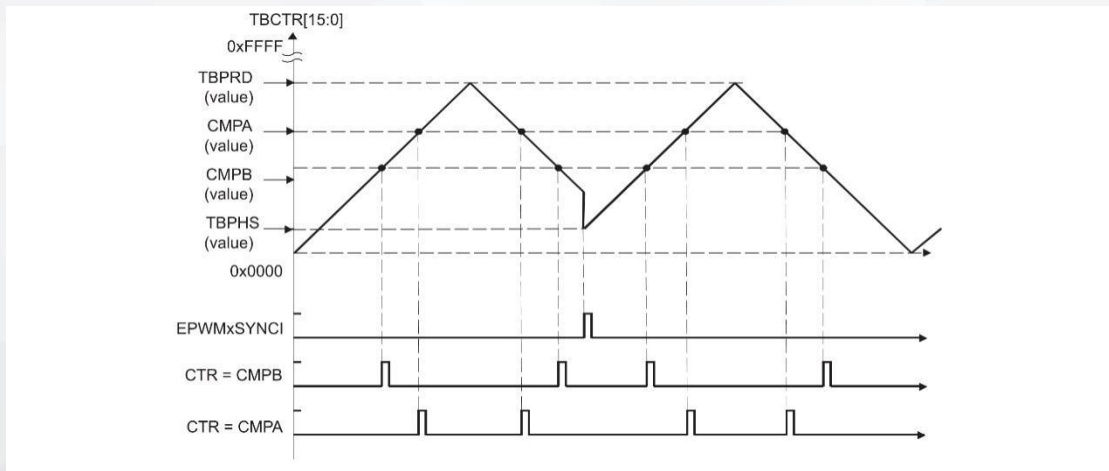


图12-19 增减计数模式下比较事件发生的情况



12.3.2 比较功能子模块CC

综上所述，ePWM的比较功能子模块的作用就是将计数器寄存器同CMPA和CMPB进行比较，以产生比较事件，然后将这些事件送入下面要介绍的动作限定子模块AQ中，驱动输出引脚产生电平的变化，从而产生PWM波。这里要提醒大家的是，不要以为EPWMxA引脚使用比较寄存器CMPA，EPWMxB引脚使用比较寄存器B，特别是对于熟悉TMS3202812的用户，因为TMS320F2812里PWM1和PWM2引脚用的是共同的比较寄存器CMPR1。在这里，EPWMxA可以使用CMPA或CMPB，EPWMxB也可以使用CMPA或CMPB，具体配置下面在AQ子模块中进行介绍。



12.3.3 动作限定子模块AQ

动作限定 (Action Qualifier) 子模块AQ在整个波形生成的环节中扮演着最重要的角色，当计数器的各种事件送入AQ后，由AQ来决定引脚应该如何动作，是变为高电平，还是变为低电平，还是没有任何动作，又或者直接翻转之前的电平，从而产生所需的PWM波形。动作限定子模块在整个ePWM中的位置如图12-20所示。

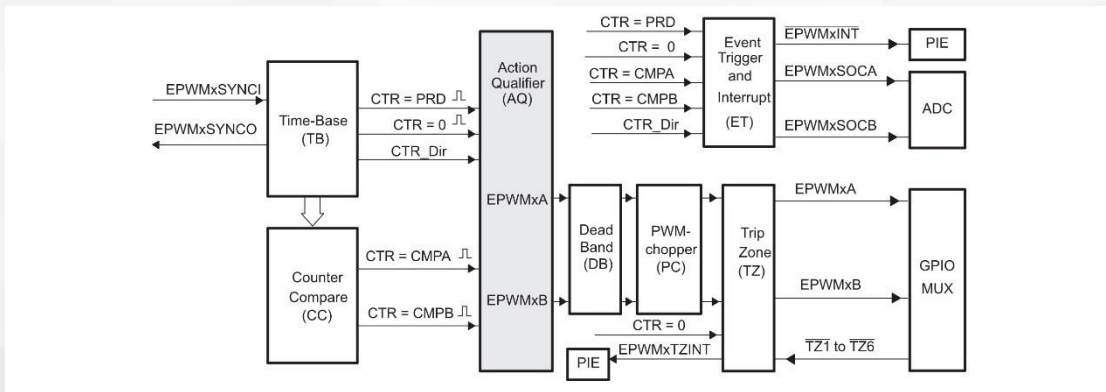


图12-20 动作限定子模块AQ在ePWM中的位置



12.3.3 动作限定子模块AQ

动作限定子模块AQ主要寄存器如表12-3所示，其内部关键信号如图12-21所示。

寄存器名	地址偏移	是否具有映射功能	说明
AQCTLA	0x000B	NO	输出A的动作限定控制寄存器
AQCTLB	0x000C	NO	输出B的动作限定控制寄存器
AQSFRC	0x000D	NO	动作限定软件强制寄存器
AQCSFRC	0x000E	YES	动作限定软件连续强制寄存器

表12-3 动作限定子模块AQ的寄存器



12.3.3 动作限定子模块AQ

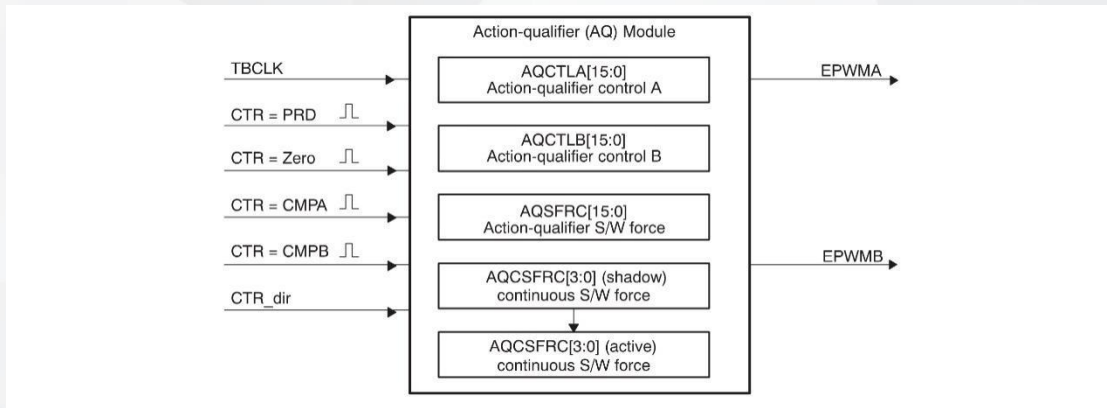


图12-21 动作限定子模块内部信号



12.3.3 动作限定子模块AQ

从图12-21可以看出，送入AQ的事件有四种，分别是：

CTR=PRD，也就是当计数器寄存器TBCTR的值与周期寄存器TBPRD相等时，通知动作限定子模块AQ；

CTR=Zero，也就是当计数器寄存器TBCTR的值等于0时，通知动作限定子模块AQ；

CTR=CMPA，也就是当计数器寄存器TBCTR的值等于比较寄存器CMPA时，通知动作限定子模块AQ；

CTR=CMPB，也就是当计数器寄存器TBCTR的值等于比较寄存器CMPB时，通知动作限定子模块AQ。



12.3.3 动作限定子模块AQ

CTR_dir为计数方向，将计数方向输入给动作限定子模块，能够让AQ对计数器的计数方向进行识别，从而使得AQ对引脚输出状态的控制变得更加灵活。加上计数方向后，能够送入动作限定子模块的事件有：

CBD：TBCTR=COMPB，且正在减计数；

CBU：TBCTR=COMPB，且正在增计数；

CAD：TBCTR=COMPB，且正在减计数；

CAU：TBCTR=COMPB，且正在增计数；

PRD：TBCTR=TBPRD；

ZRO：TBCTR=0。



12.3.3 动作限定子模块AQ

每个ePWM模块有两个输出引脚EPWMxA和EPWMxB，对这两个引脚输出动作的设定是完全独立的。EPWMxA可以通过寄存器AQCTLA进行设置，EPWMxB可以通过寄存器AQCTLB进行设置。在寄存器AQCTLA和AQCTLB中，每个事件都可以被设置为以下四种动作中的一种：

- 1.无动作，即保持EPWMxA或EPWMxB的输出状态不变，值得注意的是，虽然这种情况使得PWM引脚的输出状态不发生变化，但是这个事件仍然可以触发中断，也可以产生启动ADC转换的信号SOC；
- 2.置高，使EPWMxA或EPWMxB输出高电平；
- 3.置低，使EPWMxA或EPWMxB输出低电平；
- 4.翻转，翻转EPWMxA或EPWMxB的状态，之前是高电平的话变为低电平，之前是低电平的话变为高电平。



12.3.3 动作限定子模块AQ

在动作限定子模块中，除了计数器的各种事件能够限定PWM引脚动作外，还可以通过软件强制的功能来限定PWM引脚动作。软件强制可以通过寄存器AQSFRC和AQCSFRC来控制，比如当AQSFRC[OTSFA]=1时，就对EPWMxA引脚输出一次强制事件，此时引脚如何动作由AQSFRC[ACTSFA]来决定，可选的动作也是上面介绍的四种。AQSFRC是控制产生单次软件强制事件，而AQCSFRC是控制产生连续软件强制事件。



12.3.3 动作限定子模块AQ

为了便于介绍，采用表12-4所列的图形来表示各种动作，在默认的情况下，各个事件的动作都是“无动作”。





















软件强制	TB计数器等于				动作
	Zero	Comp A	Comp B	Period	
					无动作
					置低
					置高
					翻转

表12-4 EPWMxA和EPWMxB可能的动作



12.3.3 动作限定子模块AQ

ePWM的动作限定子模块AQ在同一时刻可以接收多个触发事件，和中断优先级类似，AQ在硬件上也设计有事件的优先级。在众多事件中，软件强制的优先级始终是最高的，因为软件强制肯定是人为干预的，所以明显要优先响应。表12-5为增减计数模式下事件的优先级。表12-6位增计数模式下事件的优先级。表12-7为减计数模式下事件的优先级。

优先级	TBCTR正在增计数TBCTR=0递增到TBCTR=TBPRD	TBCTR正在减计数TBCTR= TBPRD递增到TBCTR=0
1 (最高)	软件强制事件	软件强制事件
2	计数器的值等于CMPB (CBU)	计数器的值等于CMPB (CBD)
3	计数器的值等于CMPA (CAU)	计数器的值等于CMPA (CAD)
4	计数器等于零	计数器等于TBPRD
5	计数器的值等于CMPB (CBD)	计数器的值等于CMPB (CBU)
6 (最低)	计数器的值等于CMPA (CAD)	计数器的值等于CMPA (CAU)

表12-5 增减计数模式下的事件优先级



12.3.3 动作限定子模块AQ

优先级	事件
1（最高）	软件强制事件
2	计数器的值等于TBPRD
3	计数器的之等于CMPB（CBU）
4	计数器的之等于CMPA（CAU）
5（最低）	计数器等于零

表12-6 增计数模式下的事件优先级



12.3.3 动作限定子模块AQ

优先级	事件
1 (最高)	软件强制事件
2	计数器的值等于零
3	计数器的之等于CMPB (CBD)
4	计数器的之等于CMPA (CAD)
5 (最低)	计数器等于TBPRD

表12-7 减计数模式下的事件优先级



12.3.3 动作限定子模块AQ

通过上面的介绍可知，通过在动作限定子模块中对各种事件进行动作的限定，可以产生各种各样的波形，而在实际应用中，常用的是使用增计数模式产生不对称的PWM波形，使用减计数模式产生不对称的PWM波形，使用增减计数模式产生对称的PWM波形，PWM的占空比从0%~100%变化。



12.3.3 动作限定子模块AQ

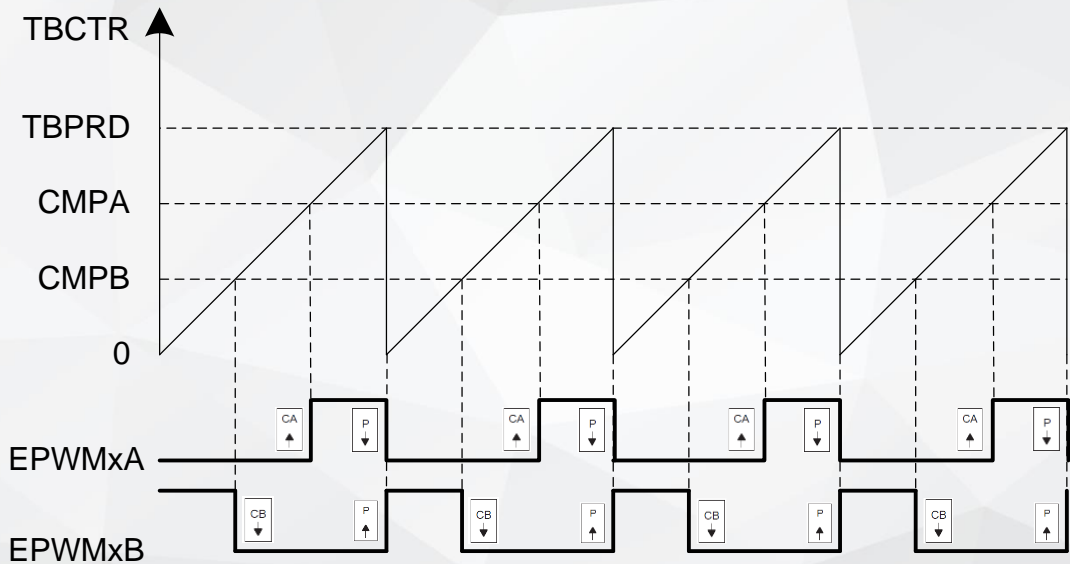


图12-22 增计数模式下产生不对称PWM波形



12.3.3 动作限定子模块AQ

EPWMxA在TBCTR=CMPA时，变为高电平，在TBCTR=TBPRD时，变为低电平。如果CMPA=0，EPWMxA则始终输出高电平，占空比为100%，如果CMPA=TBPRD，EPWMxA则始终输出低电平，占空比为0%。EPWMxA的占空比计算公式为：

$$D = \frac{TBPRD - CMPA}{TBPRD} = 1 - \frac{CMPA}{TBPRD}$$

EPWMxB在TBCTR=CMPB时，变为低电平，在TBCTR=TBPRD时变为高电平。如果CMPB=0，EPWMxB则始终输出低电平，占空比为0%，如果CMPB=TBPRD，EPWMxB则始终输出高电平，占空比为100%。EPWMxB的占空比计算公式为：

$$D = \frac{CMPB}{TBPRD}$$



12.3.3 动作限定子模块AQ

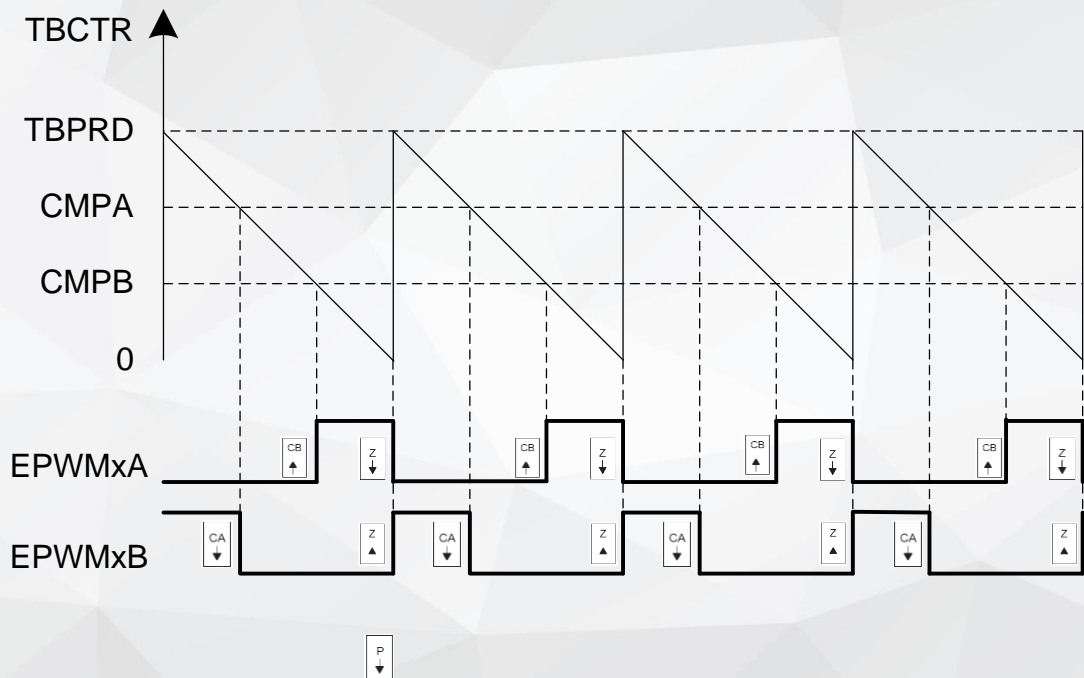


图12-23 减计数模式下产生不对称PWM波形



12.3.3 动作限定子模块AQ

EPWMxA在TBCTR=CMPB时，变为高电平，在TBCTR=TBPRD时，变为低电平。如果CMPB=0，EPWMxA则始终输出低电平，占空比为0%，如果CMPB=TBPRD，EPWMxA则始终输出高电平，占空比为100%。EPWMxA的占空比计算公式为：

$$D = \frac{CMPB}{TBPRD}$$

EPWMxB在TBCTR=CMPA时，变为低电平，在TBCTR=TBPRD时变为高电平。如果CMPA=0，EPWMxB则始终输出高电平，占空比为100%，如果CMPA=TBPRD，EPWMxB则始终输出低电平，占空比为0%。EPWMxB的占空比计算公式为：

$$D = \frac{TBPRD - CMPA}{TBPRD} = 1 - \frac{CMPA}{TBPRD}$$



12.3.3 动作限定子模块AQ

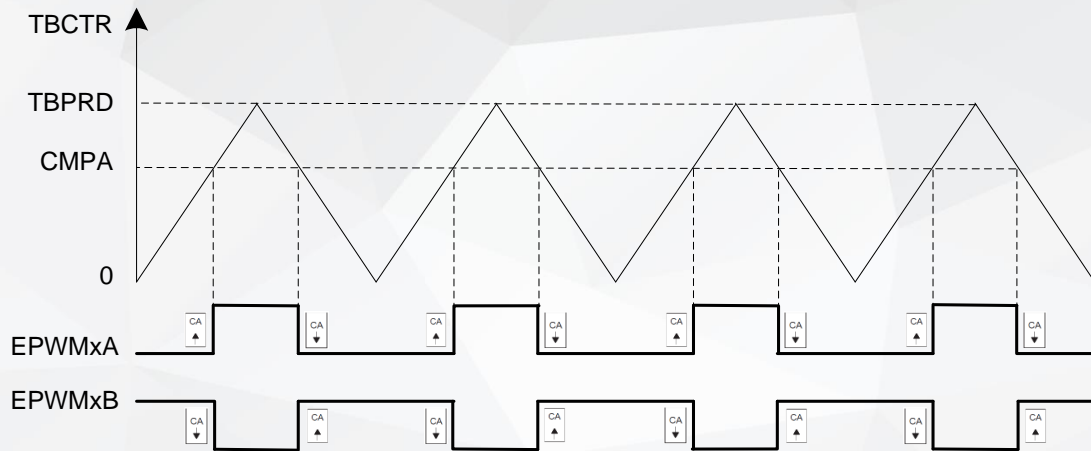


图12-24 增减计数模式下产生对称PWM波形



12.3.3 动作限定子模块AQ

EPWMxA在增计数时，如果TBCTR=CMPA，则变为高电平，在减计数时，如果TBCTR=CMPA，则变为低电平。如果CMPA=0，EPWMxA则始终输出高电平，占空比为100%，如果CMPA=TBPRD，EPWMxA则始终输出低电平，占空比为0%。EPWMxA的占空比计算公式为：

$$D = \frac{TBPRD - CMPA}{TBPRD} = 1 - \frac{CMPA}{TBPRD}$$

EPWMxB在增计数时，如果TBCTR=CMPA，则变为低电平，在减计数时，如果TBCTR=CMPA，则变为高电平。如果CMPA=0，EPWMxB则始终输出低电平，占空比为0%，如果CMPA=TBPRD，EPWMxB则始终输出高电平，占空比为100%。EPWMxB的占空比计算公式为：

$$D = \frac{CMPA}{TBPRD}$$



12.3.3 动作限定子模块AQ

简单总结下，PWM产生需要周期寄存器TBPRD，计数器寄存器TBCTR，还有比较寄存器CMPx，TBPRD决定了PWM的周期，也就是PWM的频率，计数方式、CMPx、动作限定共同决定了PWM的占空比。在使用时，改变TBPRD的值可以改变PWM的频率；改变CMPx的值可以改变PWM的占空比。



12.3.4 死区控制子模块DB

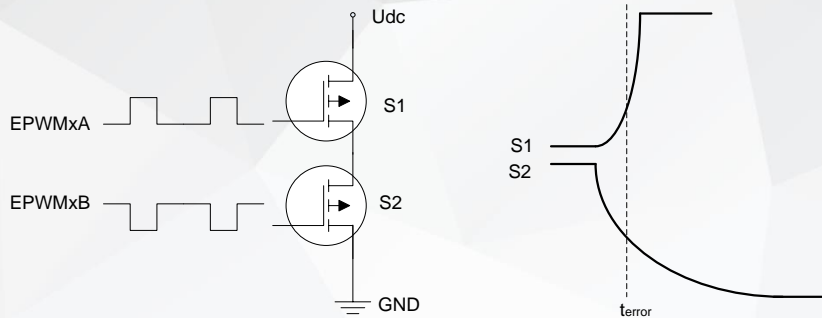


图12-25 开关管误触发引起短路



12.3.4 死区控制子模块DB

为了避免上述的现象，可以对驱动的PWM波形做一些处理，如图12-26所示，在EPWMxA和EPWMxB的上升沿与下降沿之间插入一个延时，使得同一桥臂的两个开关管导通和关断错开一定的时间，就是死区deadtime，以保证同一桥臂上的上下管子总是先关断后导通。

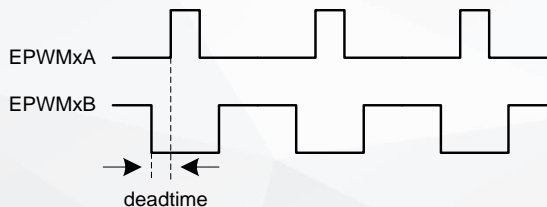


图12-26 死区概念



12.3.4 死区控制子模块DB

ePWM中的死区控制子模块DB就是用来严格地控制死区产生的边沿和极性。死区控制子模块在整个ePWM模块中的位置如图12-27所示。

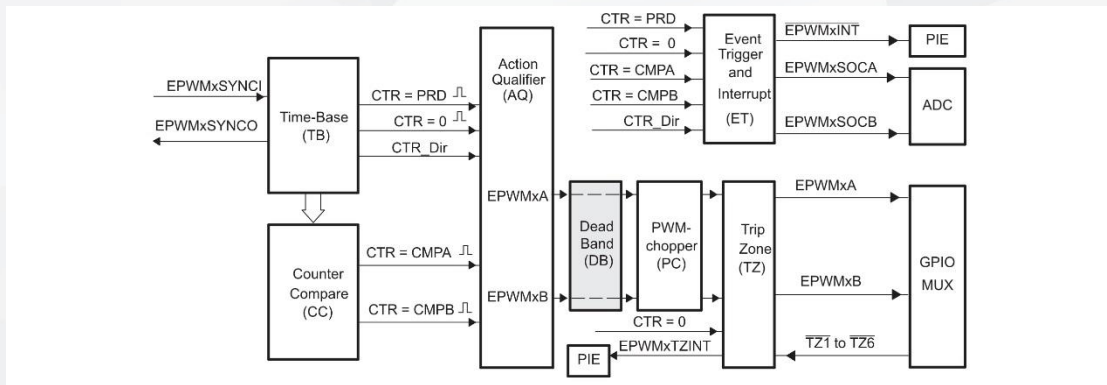


图12-27 死区控制子模块在ePWM模块中的位置



12.3.4 死区控制子模块DB

死区控制子模块相关的寄存器如表12-8所示。

寄存器名	地址偏移	是否具有映射功能	说明
DBCTL	0x000F	NO	死区控制寄存器
DBRED	0x0010	NO	死区上升沿延时寄存器
DBFED	0x0011	NO	死区下降沿延时寄存器

表12-8 死区控制子模块DB的寄存器



12.3.4 死区控制子模块DB

死区控制子模块DB的内部结构如图12-28所示，其内部主要包含了三个部分，分别是输入信号源选择、极性控制和输出模式选择，这三个部分均可以通过寄存器DBCTL来设置。

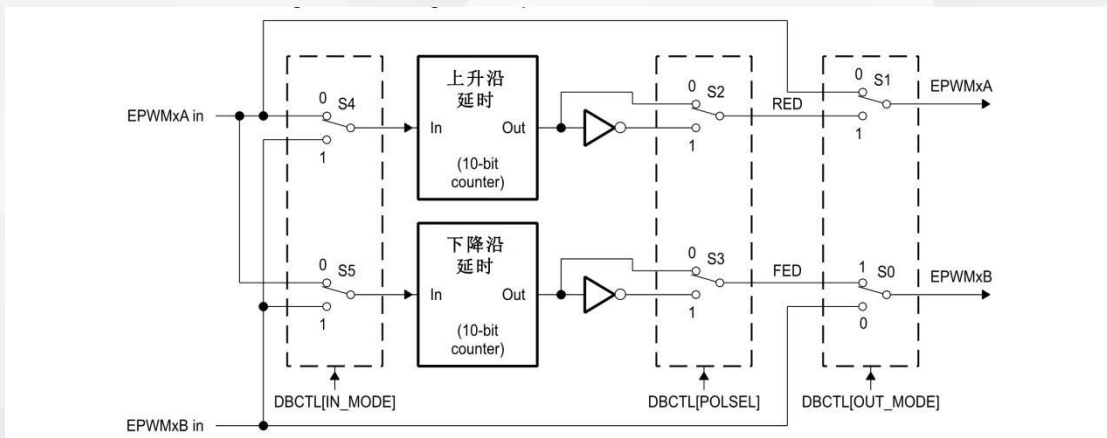


图12-28 死区控制子模块内部结构



12.3.4 死区控制子模块DB

1.输出模式选择DBCTL[OUT_MODE]，决定是否需要对输入信号进行边沿控制：

DBCTL[OUT_MODE]=0x00，禁用延时，EPWMxA和EPWMxB直接通过DB子模块；

DBCTL[OUT_MODE]=0x01，禁用上升沿延时，EPWMxA直接通过DB子模块；

DBCTL[OUT_MODE]=0x10，禁用下降沿延时，EPWMxB直接通过DB子模块；

DBCTL[OUT_MODE]=0x11，使能上升沿和下降沿延时。



12.3.4 死区控制子模块DB

2.输入信号源选择DBCTL[IN_MODE]，对需要边沿控制的信号源进行选择：

DBCTL[IN_MODE]=0x00，EPWMxA作为上升沿及下降沿延时的信号源；

DBCTL[IN_MODE]=0x01，EPWMxB作为上升沿的信号源，EPWMxA作为下降沿的信号源；

DBCTL[IN_MODE]=0x10，EPWMxA作为上升沿的信号源，EPWMxB作为下降沿的信号源；

DBCTL[IN_MODE]=0x11，EPWMxB作为上升沿及下降沿延时的信号源。



12.3.4 死区控制子模块DB

3.极性控制DBCTL[POLSEL]，决定是否在信号输出前，对经过上升沿或下降沿延时控制的信号进行取反操作：

DBCTL[POSEL]=0x00，EPWMxA和EPWMxB均不反转极性，也就是都不用取反，直接输出；

DBCTL[POSEL]=0x01，EPWMxA反转极性，EPWMxB直接输出；

DBCTL[POSEL]=0x10，EPWMxB反转极性，EPWMxA直接输出；

DBCTL[POSEL]=0x11，EPWMxA和EPWMxB均反转极性，信号取反后再输出。



12.3.4 死区控制子模块DB

下面介绍一种实际使用比较多的死区设置方案，EPWMxA和EPWMB互补输出分别驱动一个桥臂的上下管，如图12-29所示。死区控制子模块DB的输出模式设定为EPWMxA和EPWMB均需要延时，都会送进DB子模块进行延时控制。EPWMxA_in的上升沿经过延时控制后，直接输出。EPWMB_in的下降沿经过延时控制后，先取反，然后再输出。最终，得到如图12-29所示的具有死区的互补输出的一对PWM，EPWMxA和EPWMB。



12.3.4 死区控制子模块DB

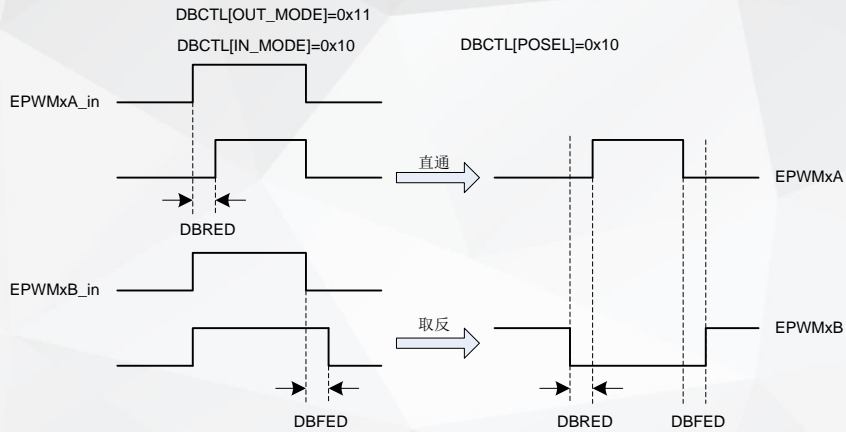


图12-29 死区控制案例



12.3.4 死区控制子模块DB

死区控制子模块DB的上升沿延时时间由寄存器DBRED决定，下降沿延时时间由DBFED决定，两个时间可以独立设置。通常，死区时间设置在几个us。DBRED和DBFED是10位寄存器，以ePWM的时钟周期TBCLK为最小的延时单位，延时时间计算式公式如下：

$$T_{RED} = DBRED \times T_{BCLK}$$

$$T_{FED} = DBFED \times T_{BCLK}$$

如果ePWM的基准时钟频率为37.5MHz，若DBRED=75，则图12-29中的EPWMxA_in的上升沿延时2us；若DBFED=75，则图12-29中的EPWMxB_in的下降沿延时2us。



12.3.5 斩波控制子模块PC

斩波控制 (PWM Chopper) 子模块PC可以通过高频载波信号对由AQ或者DB子模块输出的PWM波形进行调制，这项功能在控制高开关频率的功率器件时非常有用。斩波控制子模块PC在整个ePWM模块中的位置如图12-30所示。

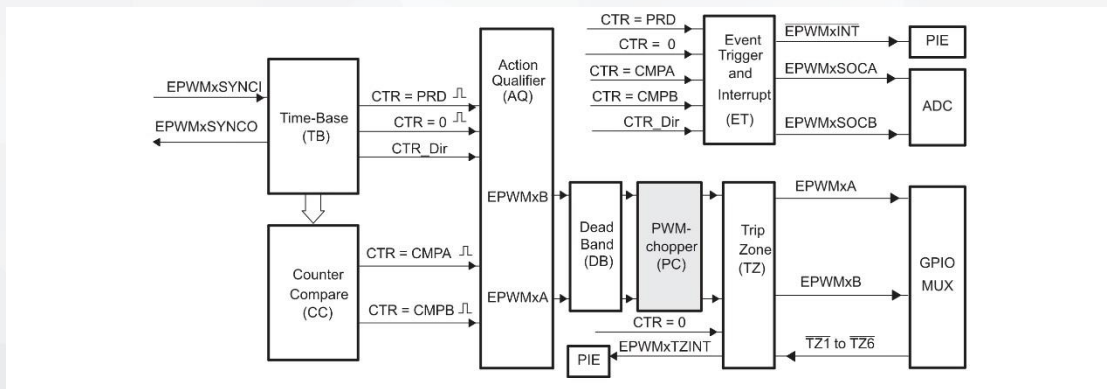


图12-30 斩波控制子模块在ePWM模块中的位置



12.3.5 斩波控制子模块PC

斩波控制子模块相关的寄存器如表12-9所示

寄存器名	地址偏移	是否具有映射功能	说明
PCCTL	0x001E	NO	PWM斩波控制寄存器

表12-9 斩波控制子模块PC的寄存器



12.3.5 斩波控制子模块PC

如果信号不需要通过斩波控制子模块而直接输出，只需将PCCTL[CHPEN]置0就可以。若将PCCTL[CHPEN]置1，则斩波功能使能，PWM信号将经过高频载波信号调制后再输出。图12-31为PC子模块的内部结构。

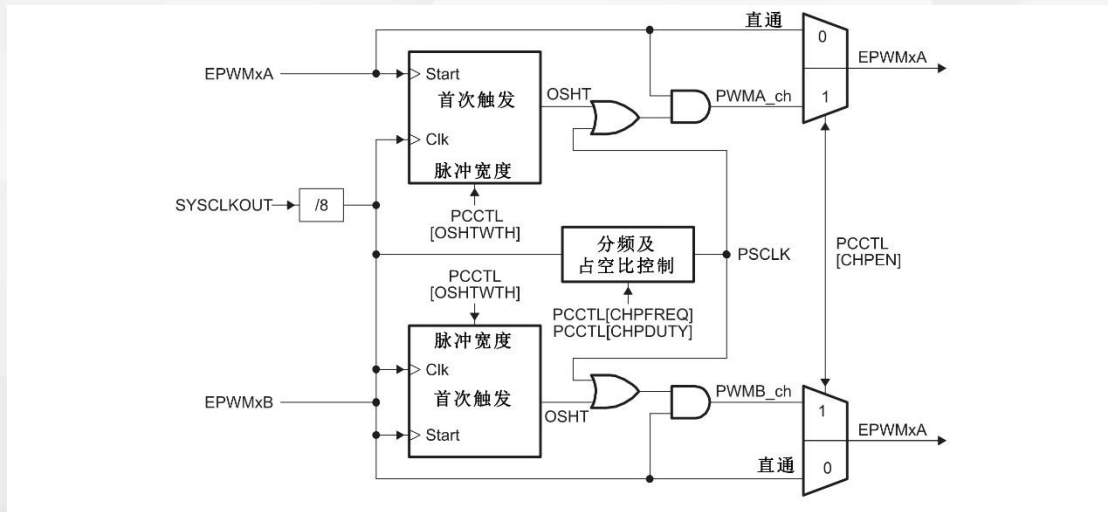


图12-31 PC子模块的内部结构



12.3.5 斩波控制子模块PC

从图12-31可以看到，高频载波信号是由系统时钟SYSCLKOUT分频而来，其频率和占空比由PCCTL[CHPFREQ]和PCCTL[CHPDUTY]控制，其频率和占空比计算公式如下：

$$f_{pwm_chopper} = \frac{SYSCLKOUT}{8 \times (CHPFREQ + 1)} \text{ Hz}$$

$$Duty_{pwm_chopper} = \frac{1 + CHPDUTY}{8} \times 100\%$$

式中，CHPFREQ的取值范围为0~7。通常，TMS320F28335的时钟频率设计为150MHz，若CHPFREQ取值为0，则载波的频率为18.75MHz；式(12-13)中CHPDUTY的取值范围为0~6。



12.3.5 斩波控制子模块PC

图12-32为PWM波形经过PC子模块高频载波调制输出的原理。从图12-32不难看出，从逻辑上分析的话，经过PC子模块调制后输出的波形其实是将PWM波形同高频载波信号做逻辑与的运算。原来是低电平的地方还是低电平，原来是高电平的地方变为高频载波。

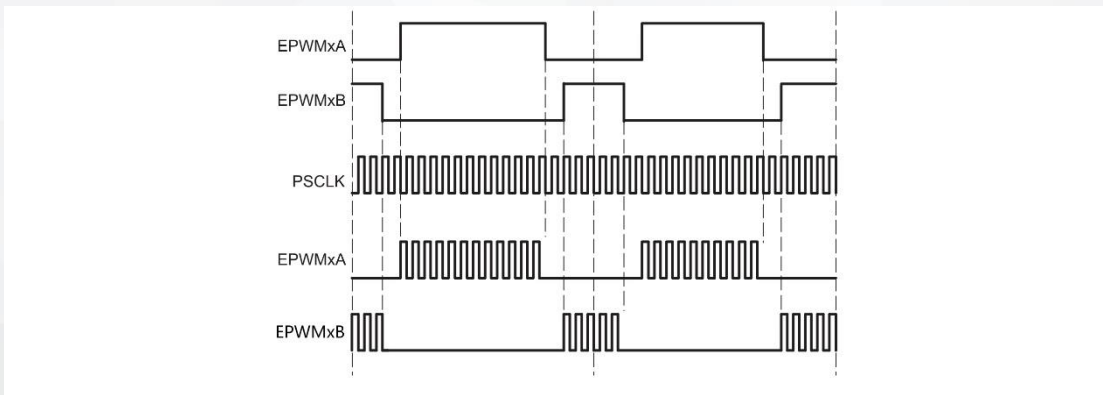


图12-32 PC子模块高频载波调制原理



12.3.5 斩波控制子模块PC

图12-32中，原先PWM高电平的地方变成了高频载波信号，把每一个周期内的第一个载波脉冲称为首次脉冲（one shot）。首次脉冲的宽度是可编程的，可以使得第一个脉冲携带较大的能量，从而保证功率器件能够可靠开通，而其余脉冲用来维持功率器件的持续开通与关断。

首次脉冲宽度可以通过PCCTL[OSHTWTH]来设置，取值范围为0~15，首次脉冲宽度计算公式如下：

$$T_{first_pulse} = T_{SYSCLKOUT} \times 8 \times (1 + OSHTWTH)$$

式中，TSYSCLKOUT为系统时钟SYSCLKOUT的周期，TM320F28335就是6.67ns。若PCCTL[OSHTWTH]=0，则首次脉冲宽度为53.36ns。



12.3.5 斩波控制子模块PC

首次脉冲及维持脉冲波形如图12-33所示。

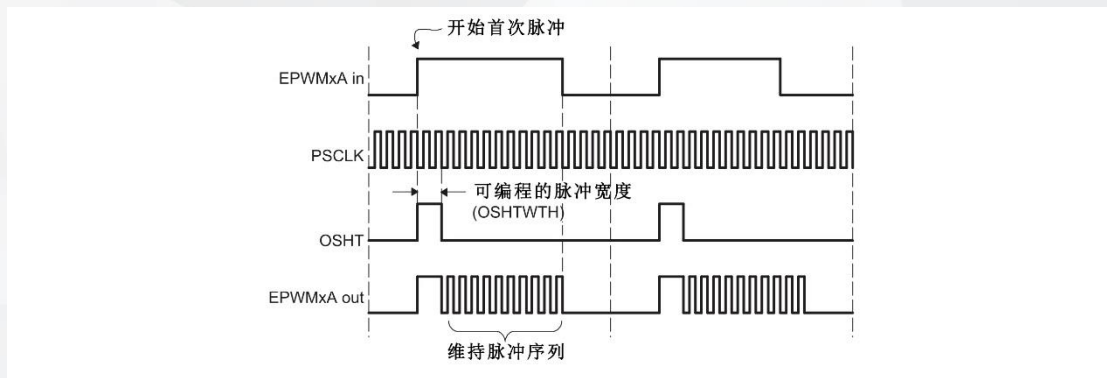


图12-33 首次脉冲及维持脉冲波形



12.3.5 斩波控制子模块PC

首个脉冲后面的维持脉冲的频率受PCCTL[CHPFREQ]控制，占空比受PCCTL[CHDUTY]控制，占空比控制如图12-34所示。。

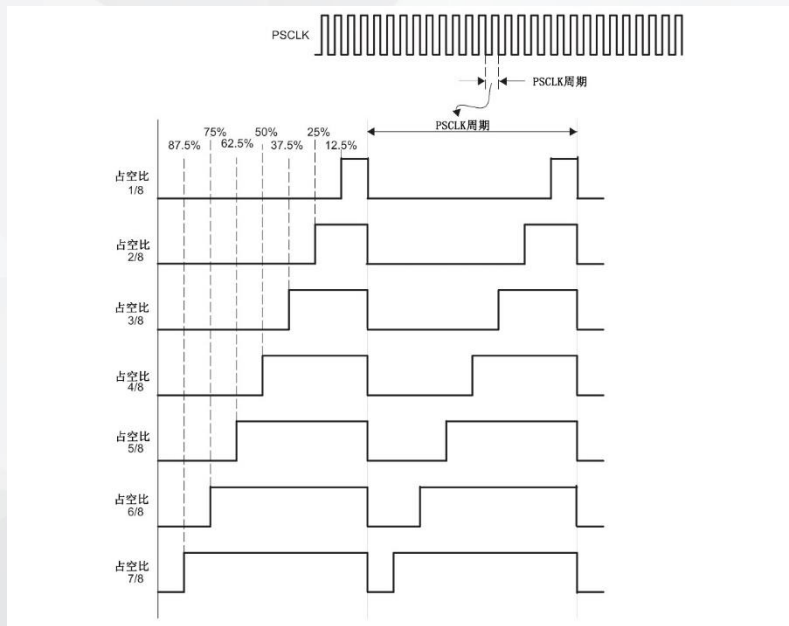


图12-34 占空比控制



12.3.6 故障捕获子模块TZ

故障捕获 (trip zone) 子模块TZ是起到故障保护的作用的，它有6个输入引脚 \overline{TZn} ，外部信号可以通过这几个引脚接入故障捕获子模块，用来表示发生了外部故障或者其他事件，从而ePWM模块对此作出相应的动作。比如将所有的PWM信号置为低电平。故障捕获子模块TZ在整个ePWM模块中的位置如图12-35所示。

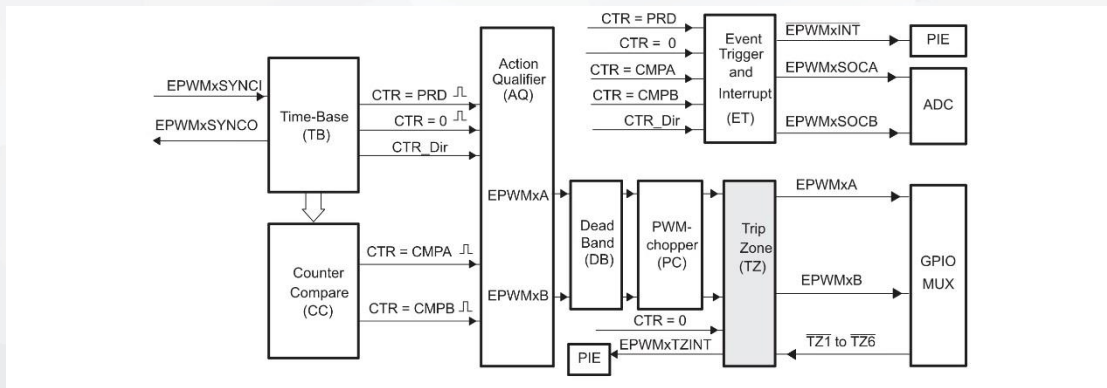


图12-35 故障捕获子模块在ePWM模块中的位置



12.3.6 故障捕获子模块TZ

从图12-35可以看出，PWM信号经过重重关卡，终于来到了最后一关，TZ关，如果外部一切正常，则TZ模块放PWM信号过关，然后通过GPIO引脚输出。一旦外部出现了故障，TZ模块接到了引脚变为低电平的信息，则立马根据相关寄存器的设置对PWM信号进行强制的处理，从而起到保护电路的作用。



12.3.6 故障捕获子模块TZ

故障故障保护子模块相关的寄存器如表12-10所示。

寄存器名	地址偏移	是否具有映射功能	说明
TZSEL	0x0012	NO	TZ选择寄存器
TZCTL	0x0014	NO	TZ控制寄存器
TZEINT	0x0015	NO	TZ中断使能寄存器
TZFLG	0x0016	NO	TZ标志寄存器
TZCLR	0x0017	NO	TZ标志清除寄存器
TZFRC	0x0018	NO	TZ强制触发寄存器

表12-10 故障捕获子模块TZ的寄存器



12.3.6 故障捕获子模块TZ

通过选择寄存器TZSEL，每个ePWM模块都可以使用或者忽略6路故障触发信号中的任何一路，如果某个ePWM不使用故障保护的功能，也就是忽略了所有的故障触发信号，那么这个ePWM模块的PWM信号不受故障保护，将直接输出。如果为某个ePWM模块选择了故障触发信号输入引脚，该引脚平时是高电平状态，如果通过电路的设计，外部一旦出现故障，将该故障触发信号输入引脚的电平置为低电平，则故障捕获子模块捕获到故障信号，然后根据控制寄存器TZCTL的设置来完成相应的动作，可以将相应的EPWMxA引脚和EPWMxB引脚强制为低电平、高电平或者高阻态输出。



12.3.6 故障捕获子模块TZ

故障捕获子模块内部逻辑电路如图12-36所示。

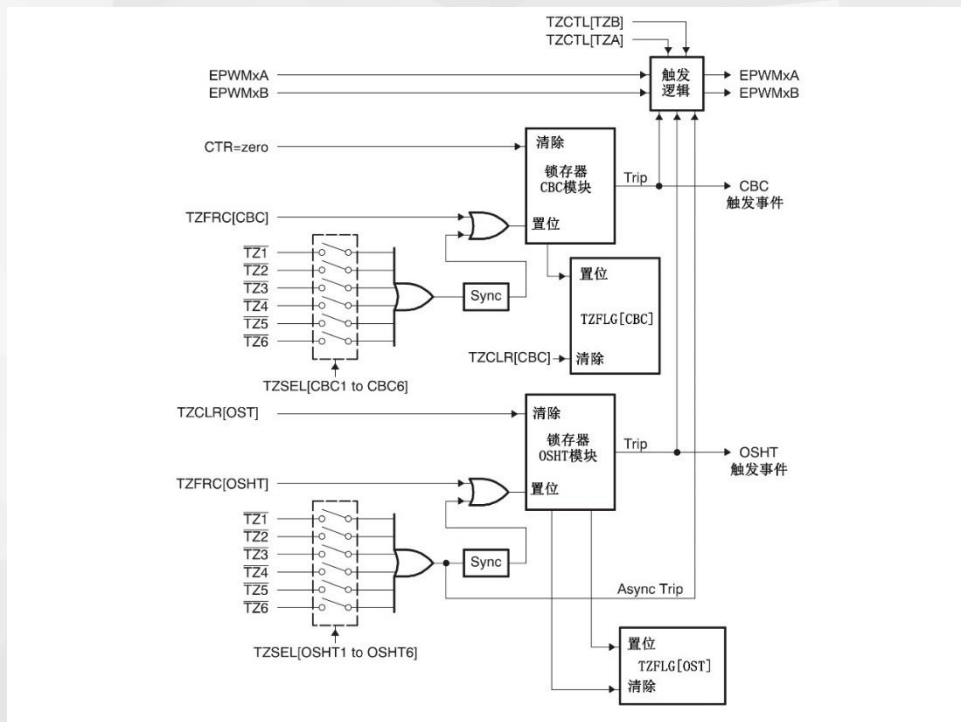


图12-36 故障捕获子模块内部逻辑电路



12.3.6 故障捕获子模块TZ

每个 \overline{TZn} 输入可以配置成单次触发 (one shot trip) 或周期性触发 (cycle-by-cycle trip) ，这由寄存器TZSEL[OSHTn]位和TZSEL[CBCn]位决定。这里需要来讲下“单次触发”和“周期性触发”的含义，如果不加以思索的话，这两个名词很容易被误解为是“外部故障单次触发故障保护功能”和“外部故障周期性触发故障保护功能”，但这样理解肯定是不正确的，外部故障是根据实际情况自动生成的，不能讲它单次或者周期性地触发故障保护，只有出现故障的时候才会产生故障信号，才回去触发故障保护。



12.3.6 故障捕获子模块TZ

1. 单次触发

单次触发是指故障捕获模块一旦被故障信号触发，就会根据TZCTL寄存器里设定的情形来强制EPWMxA和EPWMxB的输出，这种输出状态会一直保持下去，除非人为清除故障信号并复位ePWM。

另外，单次触发事件标志位TZFLG[OST]置位。如果通过TZEINT寄存器使能了外设中断和相应的PIE中断，将产生EPWMx_TZINT中断。TZFLG[OST]标志位必须通过写TZCLR[OST]位手动清除。



12.3.6 故障捕获子模块TZ

2. 周期性触发

周期性触发是以计数器TBCTR的计数周期为单位的，在每一个周期内，如果捕获到故障信号，则EPWMxA和EPWMxB的输出立即由TZCTL寄存器中所设定的状态决定，但是当PWM模块的计数器寄存器TBCTR计数到0时并且故障信号已经不存在的话，EPWMxA和EPWMxB的强制状态就会被清除。因此，在该模式下触发事件在每个ePWM周期内被清除。

另外，周期性故障触发事件标志位TZFLG[CBC]置位。如果通过TZEINT寄存器使能了外设中断和相应的PIE中断，将产生EPWMx_TZINT中断。TZFLG[CBC]标志位将一直保持不变，直到通过写TZCLR[CBC]位可将其清零。如果周期性触发事件仍然存在的话，即使手动清除TZFLG[CBC]，也会立即再次被置位。



12.3.6 故障捕获子模块TZ

图12-37为故障捕获子模块中断逻辑。

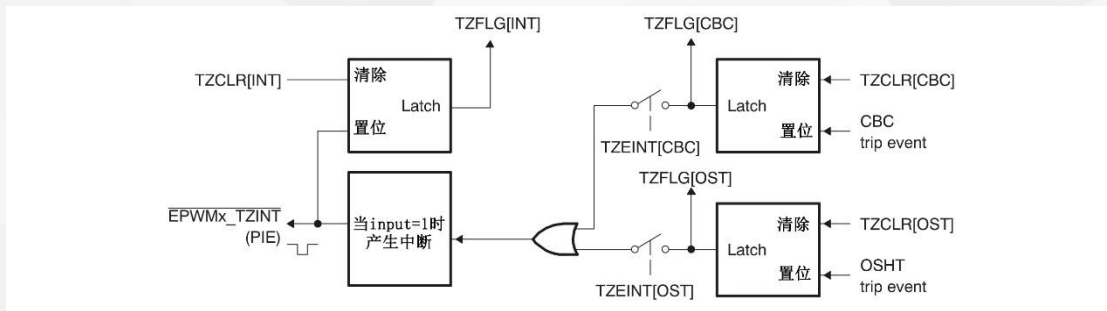


图12-37 故障捕获子模块中断逻辑



12.3.7 事件触发子模块ET

事件触发 (event trigger) 子模块ET用来处理时间基准计数器、比较功能子模块所产生的各种事件，然后向CPU发出中断请求或产生ADC启动信号SOCA或SOCB。事件触发子模块在整个ePWM模块中的位置如图12-38所示。

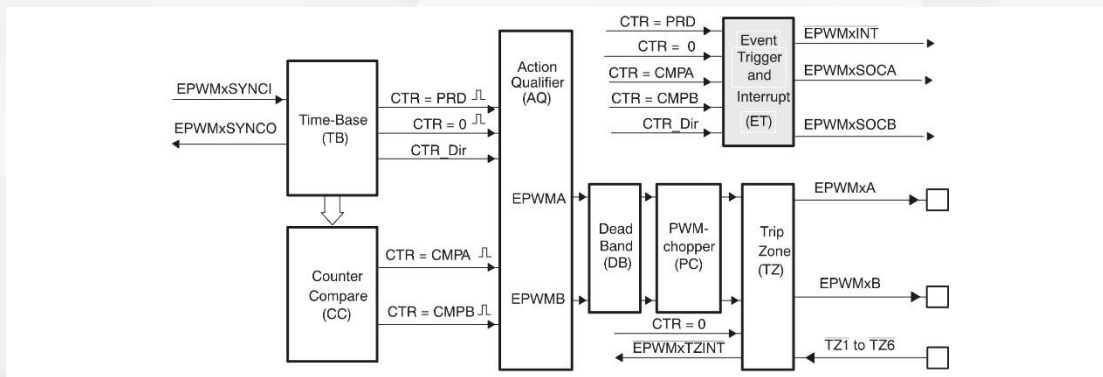


图12-38 事件触发子模块在ePWM模块中的位置



12.3.7 事件触发子模块ET

每个ePWM模块都有一条连接到PIE上的中断请求信号线和连接到ADC模块上的两路ADC启动信号SOCA及SOCB。如图12-39所示，所有ADC启动信号都通过“或门”连接到了一起，如果同时有两路ADC启动信号出现，则只有一路启动信号能被识别。

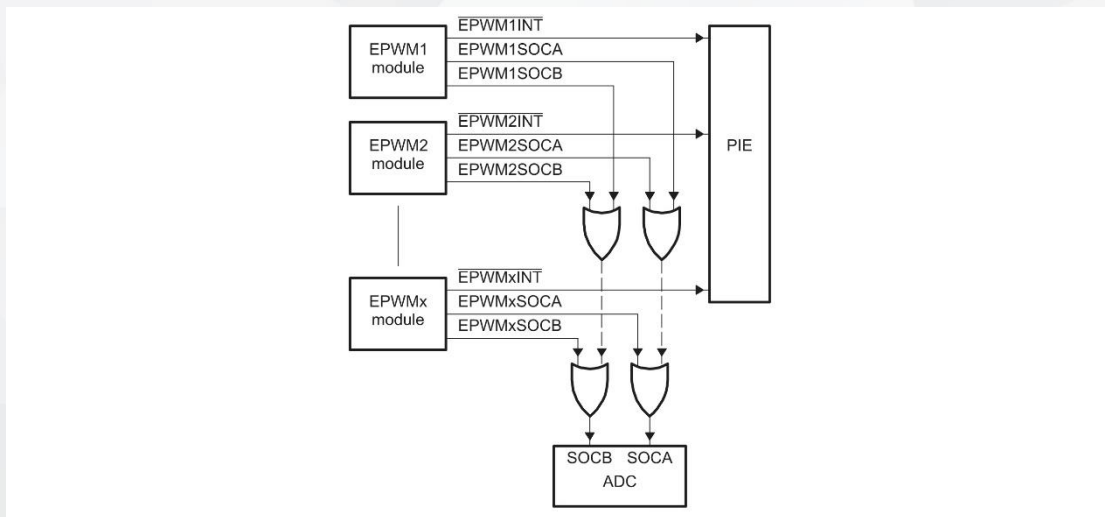


图12-39 ADC启动信号



12.3.7 事件触发子模块ET

事件触发子模块的内部信号如图12-40所示，相关的寄存器如表12-11所示。

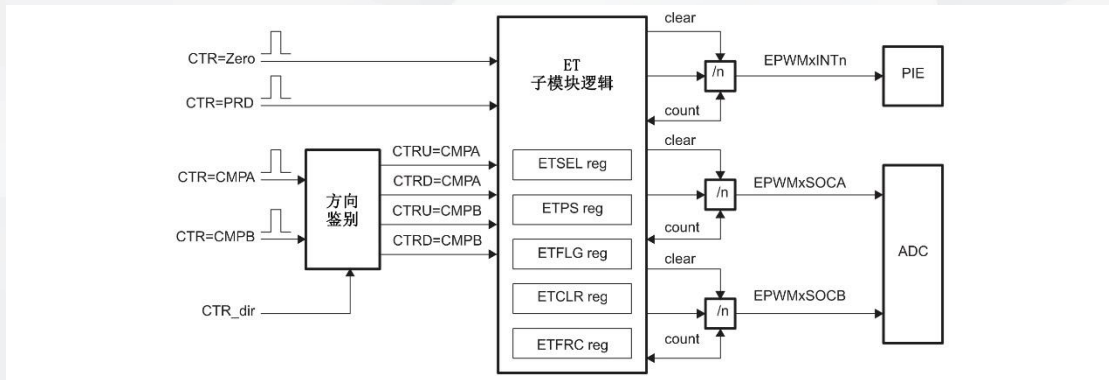


图12-40 事件触发子模块内部信号



12.3.7 事件触发子模块ET

寄存器名	地址偏移	是否具有映射功能	说明
ETSEL	0x0019	NO	ET选择寄存器
ETPS	0x001A	NO	ET预分频寄存器
ETFLG	0x001B	NO	ET标志寄存器
ETCLR	0x001C	NO	ET标志清除寄存器
ETFRC	0x001D	NO	ET强制触发寄存器

表12-11 事件触发子模块ET的寄存器



12.3.7 事件触发子模块ET

从图12-40可以看到，时间基准子模块和比较功能子模块产生的事件有：

计数器寄存器的值为0，TBCTR=0；

计数器寄存器的值为PRD，TBCTR=PRD；

当计数器增计数时，计数器寄存器的值等于CMPA的值，TBCTR=CMPA，CTR_dir=1；

当计数器减计数时，计数器寄存器的值等于CMPA的值，TBCTR=CMPA，CTR_dir=0；

当计数器增计数时，计数器寄存器的值等于CMPB的值，TBCTR=CMPB，CTR_dir=1；

当计数器减计数时，计数器寄存器的值等于CMPB的值，TBCTR=CMPB，CTR_dir=0；



12.3.7 事件触发子模块ET

上述这些事件中的任何一个都可以产生中断，也都可以产生ADC的启动信号，究竟是哪种事件可以产生中断或者ADC启动信号则可以通过ETSEL寄存器进行设置。从事件产生的结果来看就事件触发子模块就只有三种情况：触发中断、ADCSOCA、ADCSOB，这三种情况寄存器设置的内容是相同的。比如ETSEL寄存器，每种情况都有一个位用来使能或者禁止改信号，还有三位来选择具体的事件触发源。下面按中断功能和产生ADC启动信号来分别进行介绍。



12.3.7 事件触发子模块ET

1. 中断控制功能

事件触发子模块ET的中断产生逻辑如图12-41所示。

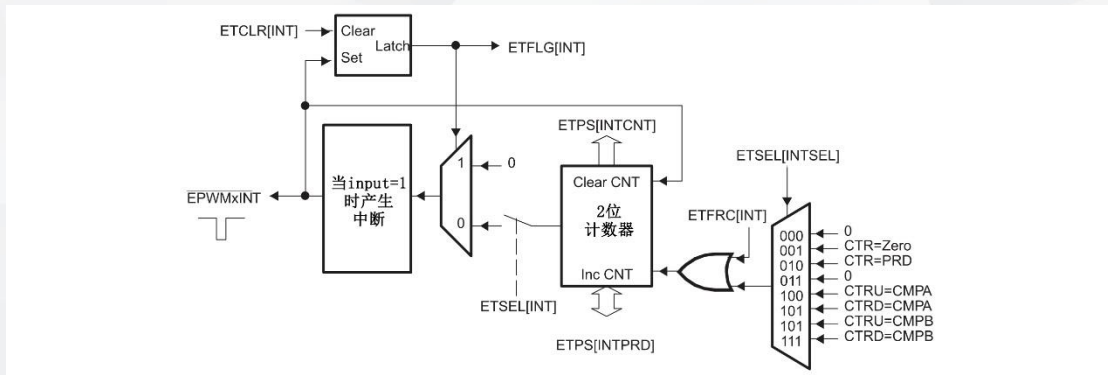


图12-41 中断产生逻辑



12.3.7 事件触发子模块ET

通常中断事件一发生，如果中断被使能的话就会产生中断。在这里会稍微有些差别。ET的中断逻辑里有一个计数器ETPS[INTCNT]，它是用来统计中断事件发生的次数的，当寄存器ETSEL[INTSEL]中设定的中断事件发生时，计数器ETPS[INTCNT]会加1，此时是不会产生中断请求的，只有当ETPS[INTCNT]=ETPS[INTPRD]时，ET才会向PIE发出中断请求。ETPS[INTPRD]用来表明每发生多少次中断事件，会产生中断信号EPWMx_INT。



12.3.7 事件触发子模块ET

当ETPS[INTCNT]=ETPS[INTPRD]时，计数器停止计数，接下来可能发生的情况有下面三种：

如果外设中断没有被使能ETSEL[INTEN]=0，或中断标志位已经被置位ETFLG[INT]=1，则不会产生中断请求，中断事件计数器ETPS[INTCNT]停止计数，保持当前值不变。

如果外设中断被使能ETSEL[INTEN]=1，且中断标志位尚未置位ETFLG[INT]=0，则会将中断标志位置位，即ETFLG[INT]=1，还会产生中断请求，当中断请求送达PIE后，计数器ETPS[INTCNT]被清零并重新开始计数。

如果外设中断被使能ETSEL[INTEN]=1，且中断标志位已经被置位ETFLG[INT]=1，也就是说前面已经产生了中断而且中断还没有被响应，则这个状态会保持，然后等CPU响应中断，等到ENTFLG[INT]被清零，计数器重新开始计数。



12.3.7 事件触发子模块ET

向ETPS[INTPRD]中写数据将直接对ETPS[INTCNT]清零，并将ETPS[INTCNT]的输出信号复位，但不产生中断请求。每次向强制中断寄存器ETFRC[INT]中写1，会使ETFLG[INTCNT]增加1，直到ETPS[INTCNT]=ETPS[INTPRD]。如果ETPS[INTPRD]=0，则中断事件计数器被禁止，不检测任何中断事件，ETFRC[INT]也被忽略，这时候也不会产生中断请求。



12.3.7 事件触发子模块ET

2.产生ADC启动信号

图12-42为ET子模块产生ADC启动信号ADCSOCA的原理图，由于ADCSOCA和ADCSOCB是相同的，因此以ADCSOCA为例来进行讲解。

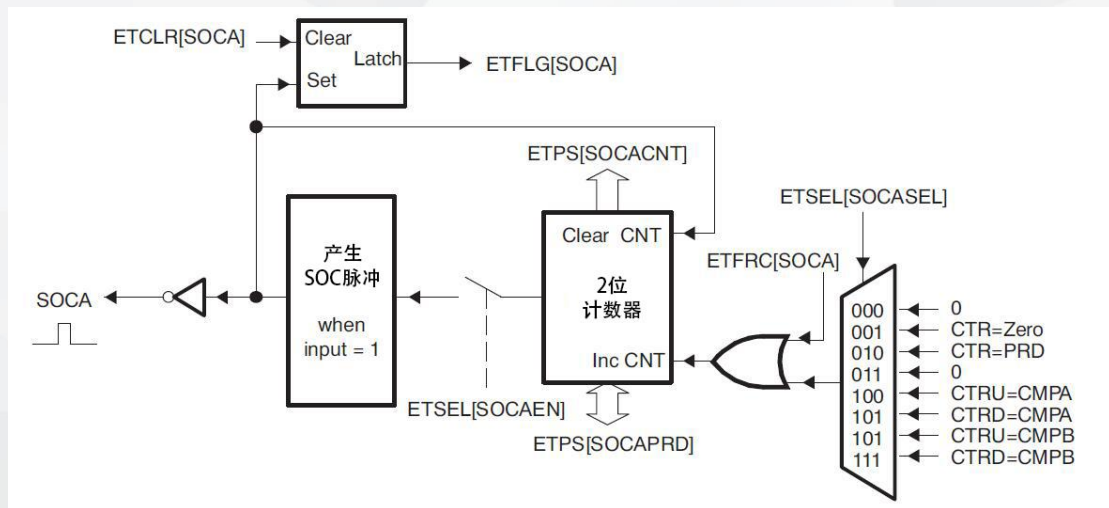


图12-42 事件触发子模块SOCA产生原理



12.3.7 事件触发子模块ET

产生ADC启动信号和产生中断的方式是类似的，也有个事件计数器ETPS[SOCACNT]用来统计事件发生的数量，然后有个ETPS[SOCAPRD]来表明每发生多少次事件产生ADC启动信号。如果ETPS[SOCAPRD]=0，则禁止事件计数器工作，也就不会产生ADC启动信号。和产生中断不同的是，启动信号ADCSOCA是连续的脉冲信号，也就是说，即使ETFLG[SOCA]被置位，也不会影响接下来脉冲的产生。



12.3.7 事件触发子模块ET

通过寄存器ETSEL[SOCA]和ETSEL[SOCB]可以分别独立设置ADC启动信号ADCSOCA和ADCSOCB的触发事件。倘若禁止ETSEL[SOCAEN]或者ETSEL[SOCBEN]，则可立即停止启动信号的产生，但是事件计数器仍然计数，直到计数器的值等于其周期寄存器的值。