



教材：《32位数字信号控制器原理及应用》

# 《DSP原理及应用》

## Lecture7 ePWM

黄灿水  
2015.3

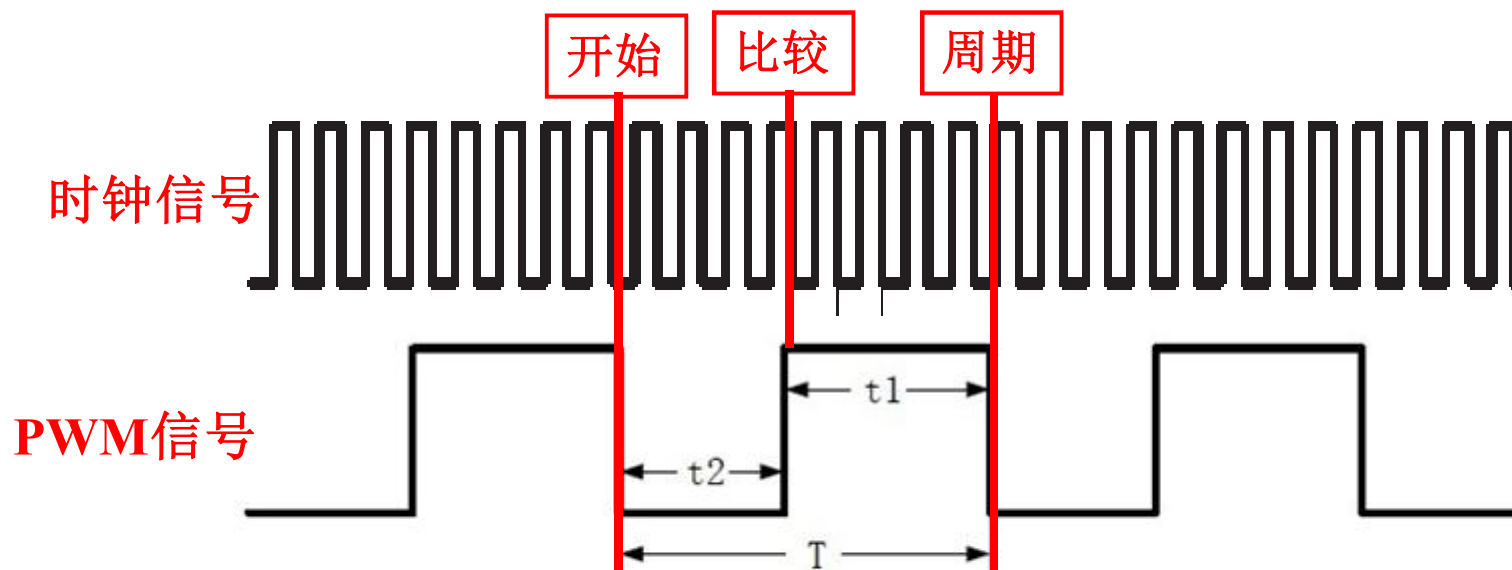




# Lecture7 ePWM

## 一、ePWM概述

**增强型脉宽调制器（ePWM）** 外设广泛用于数字电机控制系统、开关电源控制系统、不间断电源（UPS）系统、其他形式的功率控制和电源转换系统。



PWM周期： $T=t_1+t_2$

PWM频率： $F=1/T$

PWM占空比： $D=t_1/(t_1+t_2)=t_1/T$





## Lecture7 ePWM

### 一、ePWM概述

**PWM事件**就是几个特别的时刻

- (1) 0时刻
- (2) 周期时刻
- (3) 比较点时刻

**PWM动作**（类似GPIO引脚的输出）

- (1) 没反应，**Do nothing**
- (2) 置低电平，**Clear low**
- (3) 置高电平，**Set High**
- (4) 电平翻转，**Toggle**





## Lecture7 ePWM

### 一、ePWM概述

TMS320F2802x每个ePWM特性:

- 专用的**16位时基计数器**
- 具有**两路PWM**输出引脚 (EPWMxA/EPWMxB)
  - 两个独立的、单边沿操作的PWM输出
  - 两个独立的、双边沿对称操作的PWM输出
  - 一个独立的、双边沿非对称操作的PWM输出
- 通过软件**异步**控制PWM信号
- 可编程的**相位控制**，配置不同ePWM模块的相位差
- 周期性地硬件锁定 (**同步**) 相位关系







## Lecture7 ePWM

### 一、ePWM概述

TMS320F2802x每个ePWM特性:

- 具有死区发生器，带独立的上升沿和下降沿延迟控制
- 可编程触发区配置，故障时周期性触发或单次触发
- 故障时PWM输出可强制为高、低或高阻状态
- 比较器模块输出和触发区输入可以产生事件、滤波（filtered）事件或故障触发条件
- 所有事件都可以触发CPU中断和ADC开始转换（SOC）
- 事件预分频因子可编程，使得中断的CPU开销最少
- PWM被高频载波信号斩波，脉冲变压器门极驱动有用

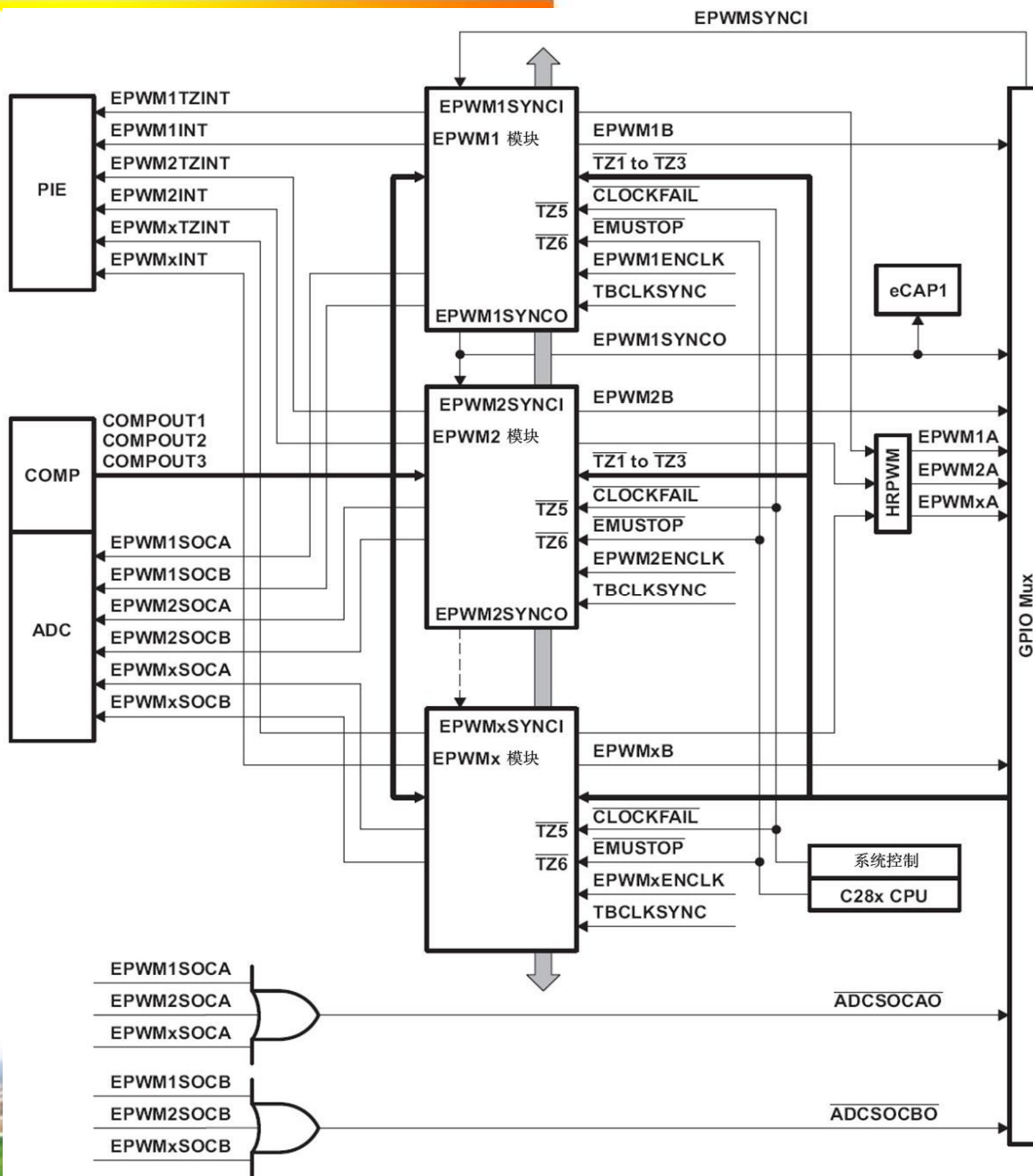




# Lecture7 ePWM

## 一、ePWM概述

ePWM 模块方框图

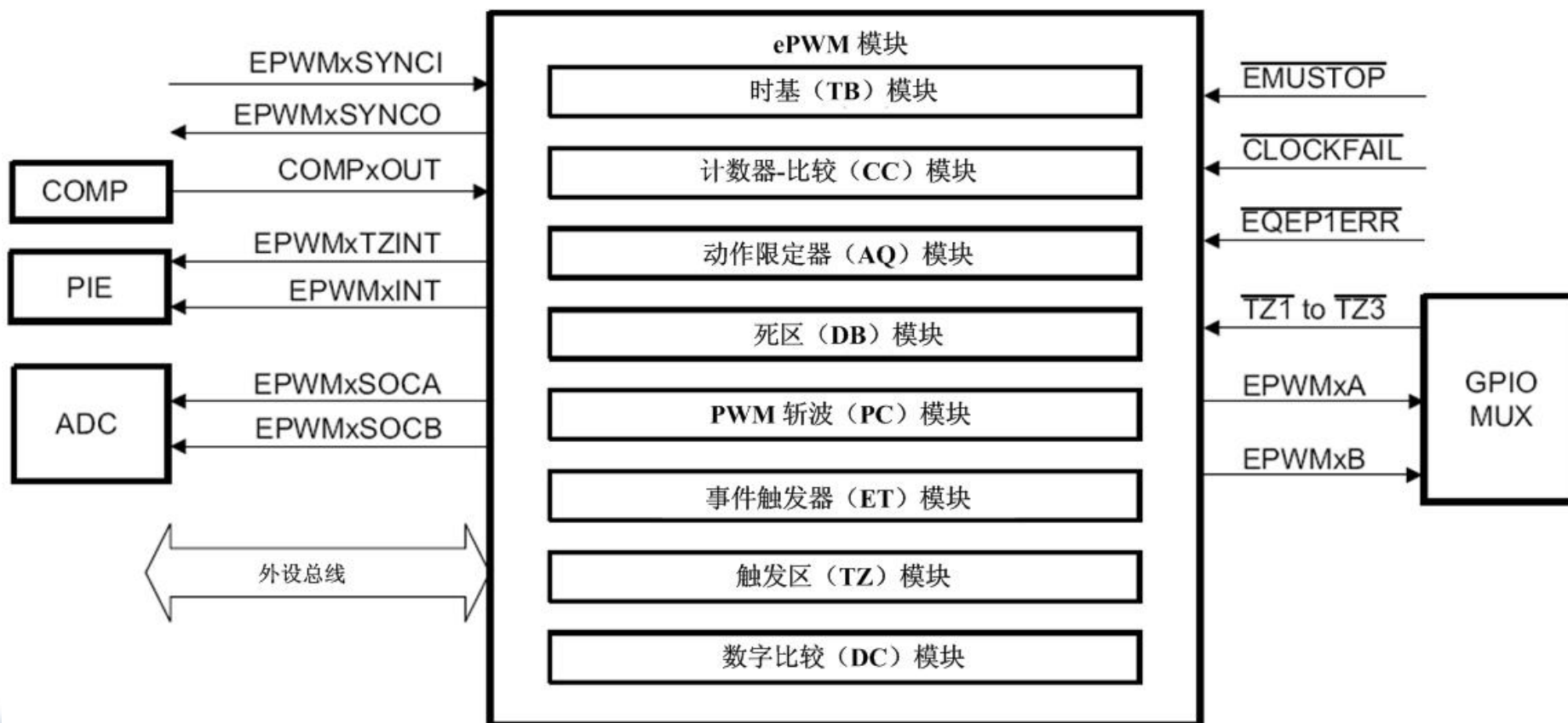




# Lecture7 ePWM

## 一、ePWM概述

ePWM模块8个子模块:





## Lecture7 ePWM

### 一、ePWM概述

#### ePWM模块信号:

- **PWM输出信号** (EPWMxA和EPWMxB)，与GPIO复用。
- **触发区信号** (TZ1到TZ6)。模块外部出现故障条件，TZ1到TZ3可配置为GPIO外设异步输入。TZ4与EQEP1模块相连。TZ5与系统时钟失效逻辑单元相连，TZ6与来自CPU的EMUSTOP输出相连。
- **时基同步输入** (EPWMxSYNCI) 和输出 (EPWMxSYNCO) 信号，可通过ePWM1管脚输出，在内部将所有ePWM模块相连。。  
EPWM1SYNCO与eCAP1的SYNCI连接。
- **ADC开始转换信号** (EPWMxSOCA和EPWMxSOCB)。
- **比较器输出信号** (COMPxOUT)，与触发区信号产生数字比较事件。
- **外设总线**，32位宽，可以对ePWM寄存器执行16位和32位写操作。

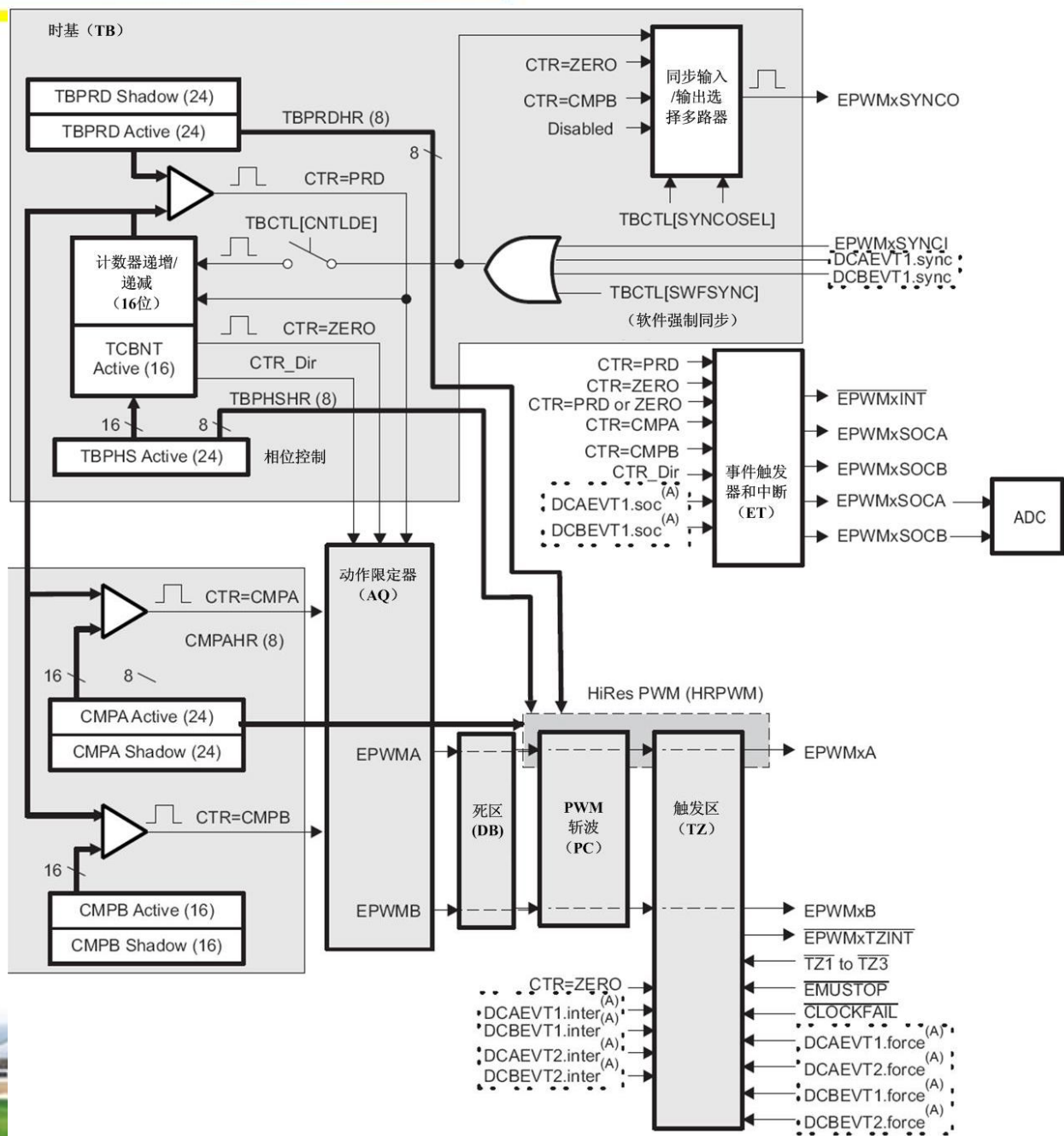




# Lecture7 ePWM

## 一、ePWM概述

8个子模块信号连接







# Lecture7 ePWM

## 二、ePWM寄存器映射

名称	大小	EALLOW	描述
时基子模块寄存器			
TBCTL	1		时基控制寄存器
TBSTS	1		时基状态寄存器
TBPHSHR*	1		时基相位高分辨率寄存器
TBPHS	1		时基相位寄存器
TBCTR	1		时基计数器寄存器
TBPRD	1		时基周期寄存器
TBPRDHR*	1		时基周期高分辨率寄存器
计数器-比较子模块寄存器			
CMPCTL	1		计数器-比较控制寄存器
CMPAHR*	1		计数器-比较A高分辨率寄存器
CMPA	1		计数器-比较A寄存器
CMPB	1		计数器-比较B寄存器





# Lecture7 ePWM

## 二、ePWM寄存器映射

(续表)

名称	大小	EALLOW	描述
事件触发器子模块寄存器			
ETSEL	1		事件触发器选择寄存器
ETPS	1		事件触发器预分频寄存器
ETFLG	1		事件触发器标志寄存器
ETCLR	1		事件触发器清零寄存器
ETFRC	1		事件触发器强制寄存器
死区发生器子模块寄存器			
DBCTL	1		死区发生器控制寄存器
DBRED	1		死区发生器上升沿延迟计数寄存器
DBFED	1		死区发生器下降沿延迟计数寄存器





## Lecture7 ePWM

### 二、ePWM寄存器映射

(续表)

名称	大小	EALLOW	描述
触发区子模块寄存器			
TZSEL	1	是	触发区选择寄存器
TZDCSEL	1	是	触发区数字比较选择寄存器
TZCTL	1	是	触发区控制寄存器
TZEINT	1	是	触发区使能中断寄存器
TZFLG	1	是	触发区标志寄存器
TZCLR	1	是	触发区清零寄存器
TZFRC	1	是	触发区强制寄存器
动作限定器子模块寄存器			
AQCTLA	1		EPWMxA动作限定器控制寄存器
AQCTLB	1		EPWMxB动作限定器控制寄存器
AQSFRC	1		动作限定器软件强制寄存器
AQCSFRC	1		动作限定器连续S/W强制寄存器组



# Lecture7 ePWM

## 二、ePWM寄存器映射

(续表)

名称	大小	EALLOW	描述
<b>PWM斩波子模块寄存器</b>			
PCCTL	1		PWM斩波控制寄存器
<b>高分辨率脉宽调制器（HRPWM）扩展寄存器</b>			
HRCNFG*	1	是	HRPWM配置寄存器
HRPWR*	1	是	HRPWM功率寄存器
HRMSTEP*	1	是	HRPWM MEP步进寄存器
HRPCTL*	1	是	高分辨率周期控制寄存器
TBPRDHRM*	1		时基周期高分辨率镜像寄存器
TBPRDM	1		时基周期镜像寄存器
CMPAHRM*	1		比较A高分辨率镜像寄存器
CMPAM	1		比较A镜像寄存器





# Lecture7 ePWM

## 二、ePWM寄存器映射

(续表)

名称	大小	EALLOW	描述
数字比较事件寄存器			
DCTRIPSEL	1	是	数字比较触发选择寄存器
DCACTL	1	是	数字比较A控制寄存器
DCBCTL	1	是	数字比较B控制寄存器
DCFCTL	1	是	数字比较滤波控制寄存器
DCCAPCTL	1	是	数字比较捕获控制寄存器
DCFOFFSET	1		数字比较滤波偏移量寄存器
DCFOFFSETCNT	1		数字比较滤波偏移量计数器寄存器
DCFWINDOW	1		数字比较滤波窗寄存器
DCFWINDOWCNT	1		数字比较滤波窗计数器寄存器
DCCAP	1		数字比较计数器捕获寄存器







## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 1、时基 (TB) 模块的功能

- 根据系统时钟 (SYSCLKOUT) 调节时基时钟 (TBCLK)
- 配置PWM时基计数器 (TBCTR) 的频率或周期
- 设置时基计数器的计数模式：
  - 递增计数模式：用于非对称PWM
  - 递减计数模式：用于非对称PWM
  - “先递增后递减”计数模式：用于对称PWM
- 配置相对于另一个ePWM模块的时基相位
- 通过硬件或软件将各模块之间的时基计数器同步





## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 1、时基 (TB) 模块的功能

- 配置发生同步事件后时基计数器的**计数方向** (递增或递减)
- 配置器件被**仿真器**异常终止时时基计数器的行为
- 指定ePWM模块**同步输出的源**:
  - 同步输入信号
  - 时基计数器的计数值等于0
  - 时基计数器的计数值等于计数器-比较B (CMPB) 的值
  - 没有产生输出同步信号

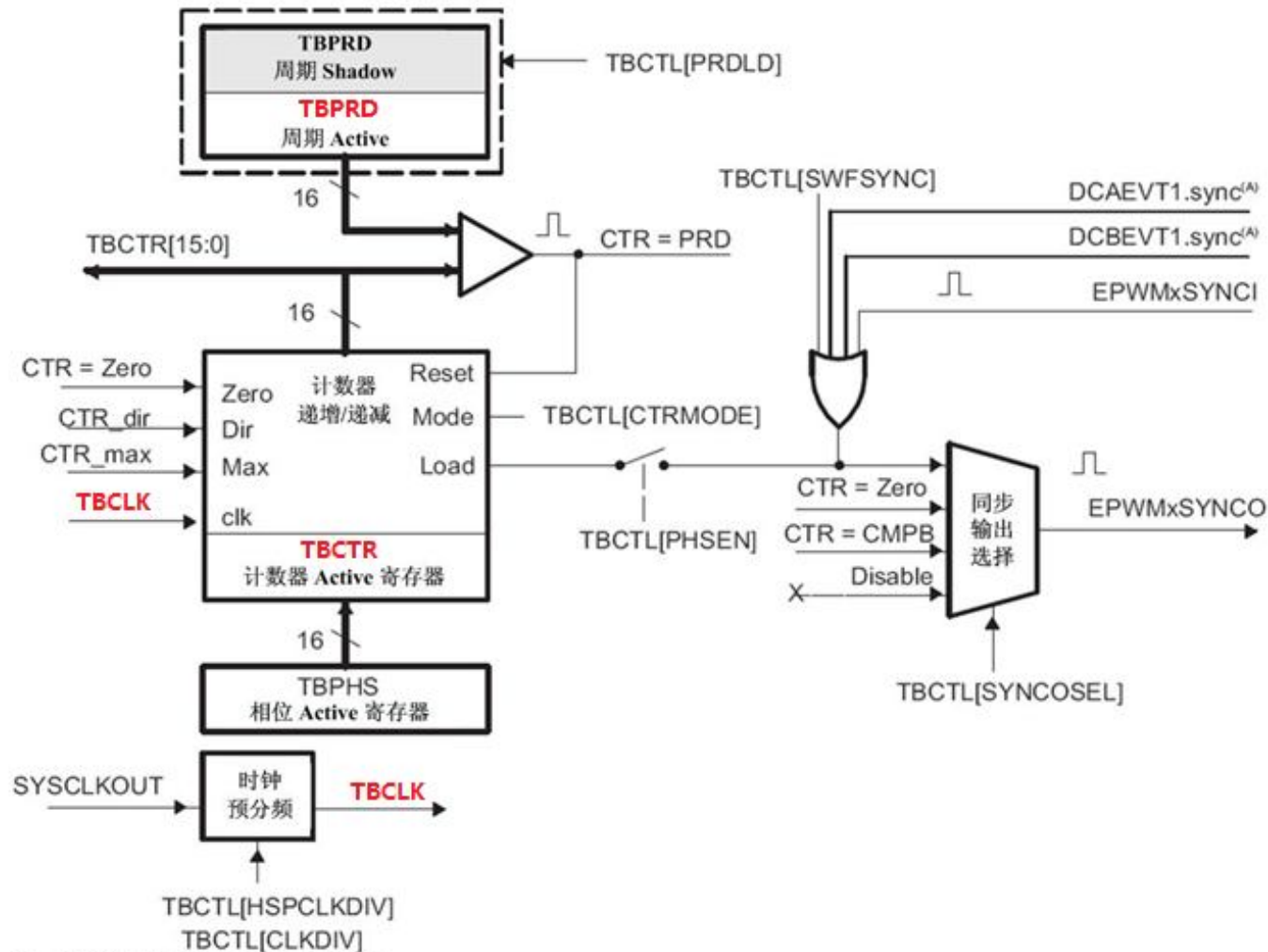




# Lecture7 ePWM

## 三、ePWM子模块功能和配置 -- 时基 (TB)

### 2、时基 (TB) 模块的组成



注A: 这些信号由数字比较 (DC) 子模块产生





## Lecture7 ePWM

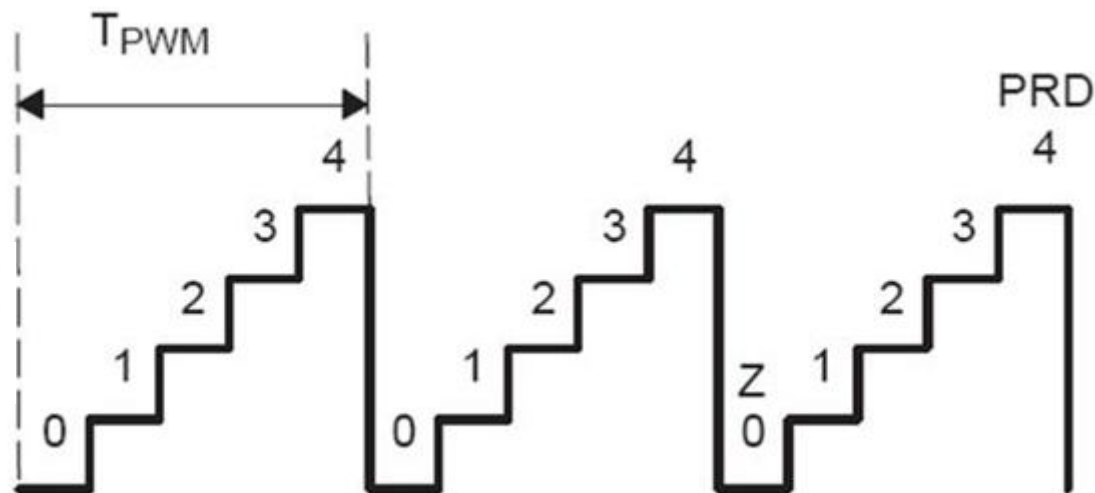
### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 3、计算PWM的周期或频率

PWM的周期或频率由时基周期 (TBPRD) 寄存器和时基计数器的计数模式控制。

递增计数模式:

$$T_{PWM} = (TBPRD + 1) \times T_{TBCLK}$$
$$F_{PWM} = 1 / (T_{PWM})$$





# Lecture7 ePWM

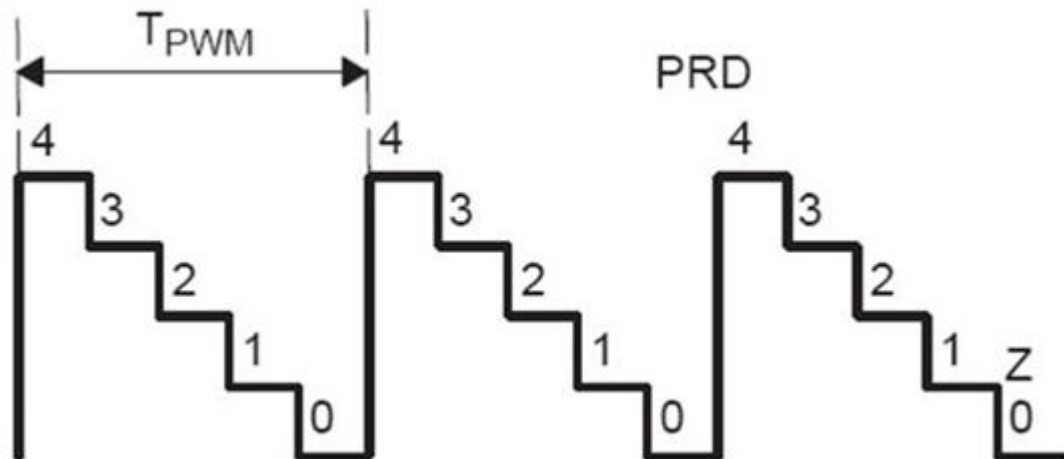
## 三、ePWM子模块功能和配置 -- 时基 (TB)

### 3、计算PWM的周期或频率

递减计数模式:

$$T_{PWM} = (TBPRD + 1) \times T_{TBCLK}$$

$$F_{PWM} = 1 / (T_{PWM})$$

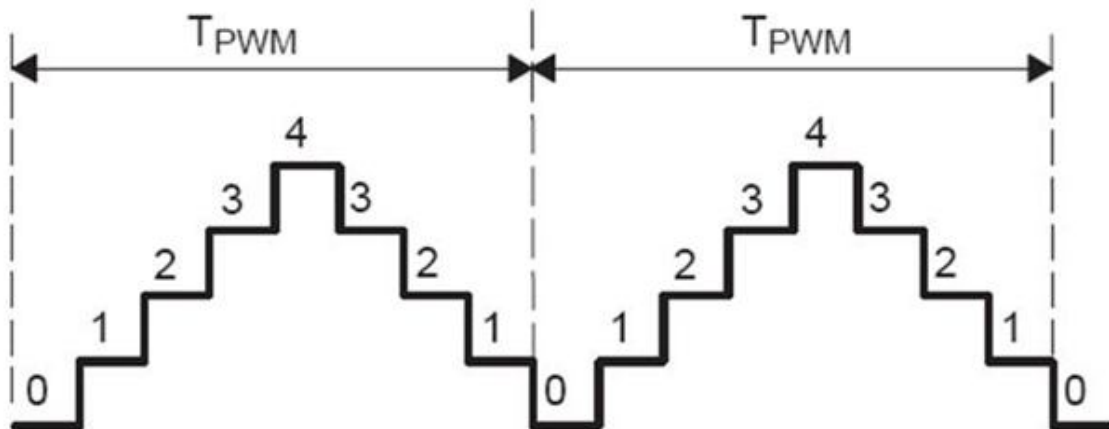


先递增后递减

计数模式:

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

$$F_{PWM} = 1 / (T_{PWM})$$







## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 4、时基 (TB) 周期寄存器 (TBPRD)

shadow周期寄存器的存储器地址与active寄存器相同。哪个寄存器被写或被读由TBCTL. PRDL位决定



TBCTL. PRDL=0: TBPRD指向shadow寄存器。当时基计数器

TBCTR=0时, shadow寄存器内容送到active寄存器。

TBCTL. PRDL=1: TBPRD指向active寄存器。





# Lecture7 ePWM

## 三、ePWM子模块功能和配置 -- 时基 (TB)

### 5、时基 (TB) 模块的同步机制

适用于:

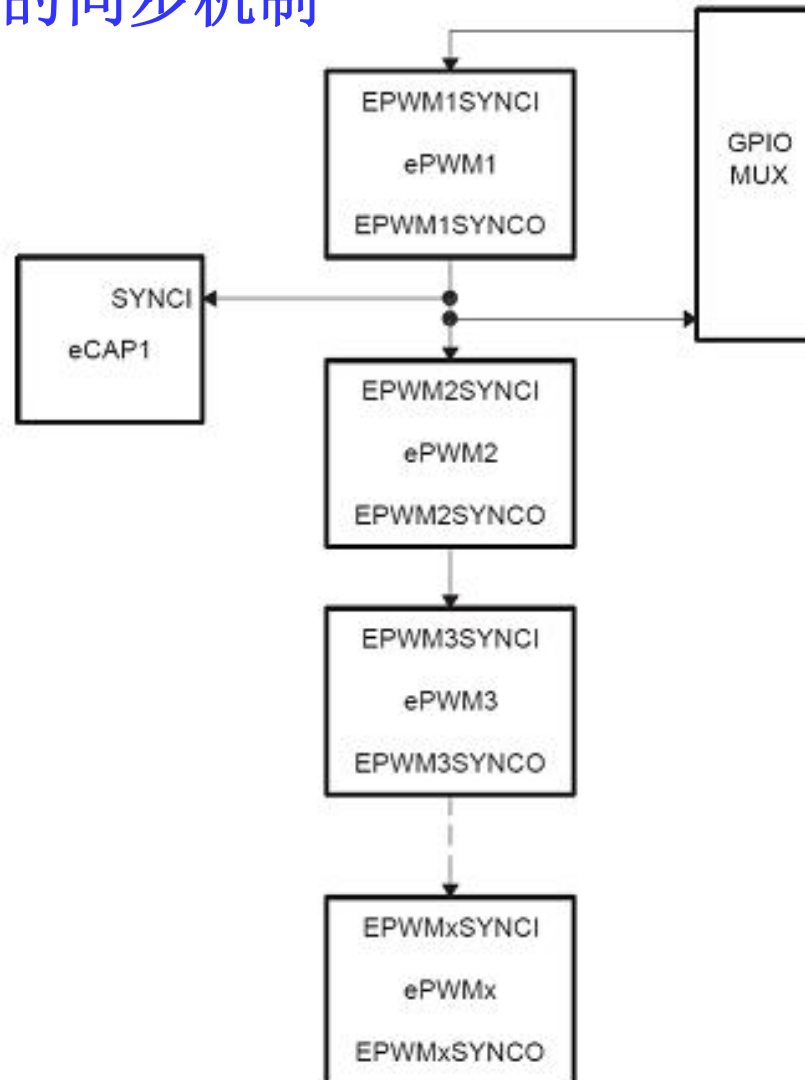
**280x**

**2801x**

**2802x**

**2803x**

**2804x兼容模式**





## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 5、时基 (TB) 模块的同步机制

若**TBCTL.PHSEN=1**，出现以下脉冲时，**TBCTR=TBPHS**：

- **EPWMxSYNCl**：同步输入脉冲：
- 软件强制同步脉冲：向**TBCTRL.SWFSYNC**位写1
- 数字比较事件同步脉冲：**DCAEVT1**和**DCBEVT1**数字比较事件。

若**TBCTL.PHSEN=0**，忽略同步信号。但是**EPWMxSYNCO**可以流过，用于同步其他**ePWM**模块。





## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 6、多个ePWM模块时基时钟同步步骤

- (1) 使能单独的ePWM模块时钟。PCLKCR1的3-0位
- (2) 设置TBCLKSYNC=0 (PCLKCR0.2位)，所有被使能ePWM模块的时基时钟都将停止运行。
- (3) 配置预分频器的值以及期望的ePWM模式。要得到完全同步的TBCLK，使能ePWM模块预分频必须相同。
- (4) 设置TBCLKSYNC=1。同时启动使能的ePWM时钟。

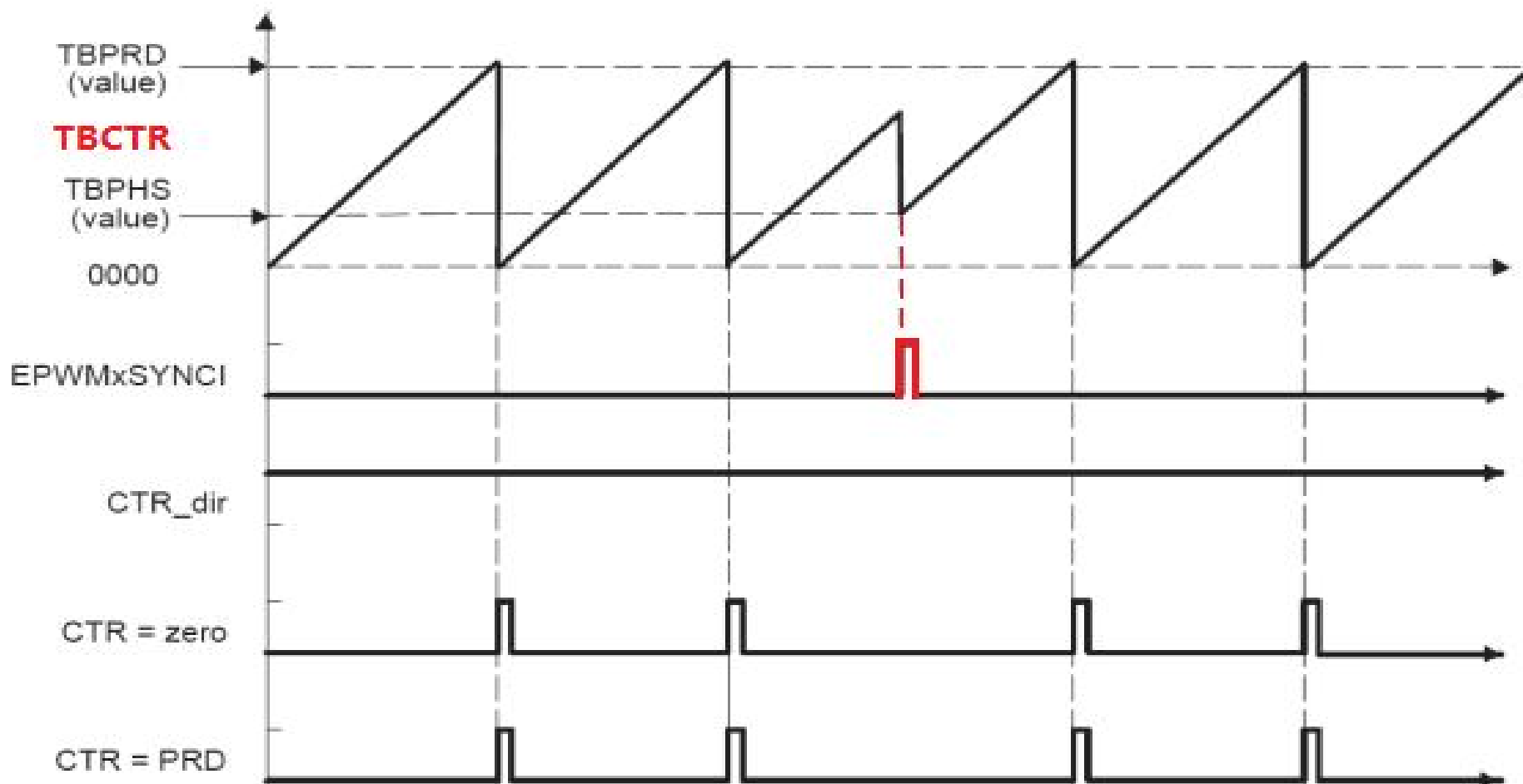




# Lecture7 ePWM

## 三、ePWM子模块功能和配置 -- 时基 (TB)

### 7、时基 (TB) 模块的时序波形 - 递增



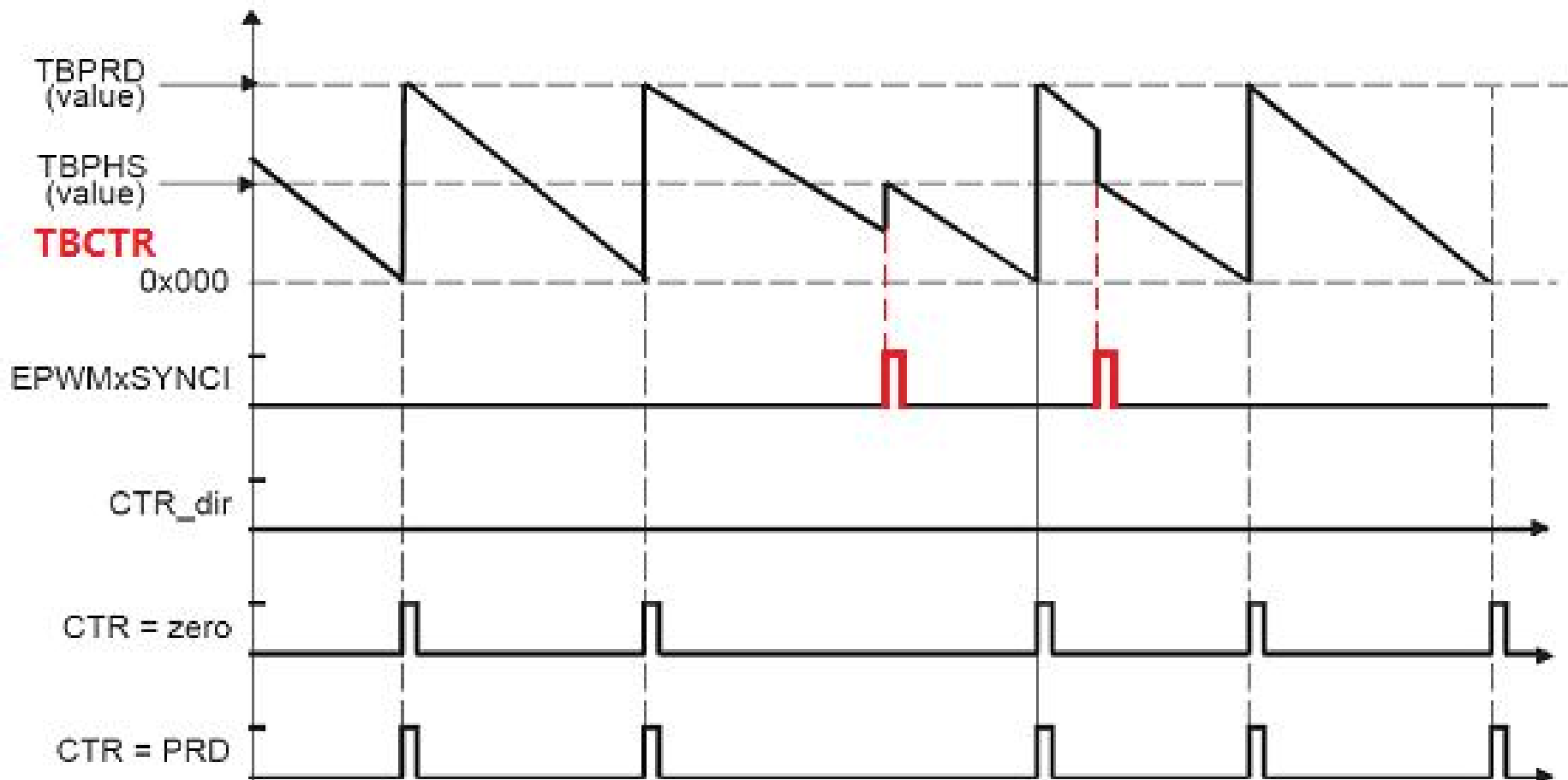




## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 7、时基 (TB) 模块的时序波形 - 递减



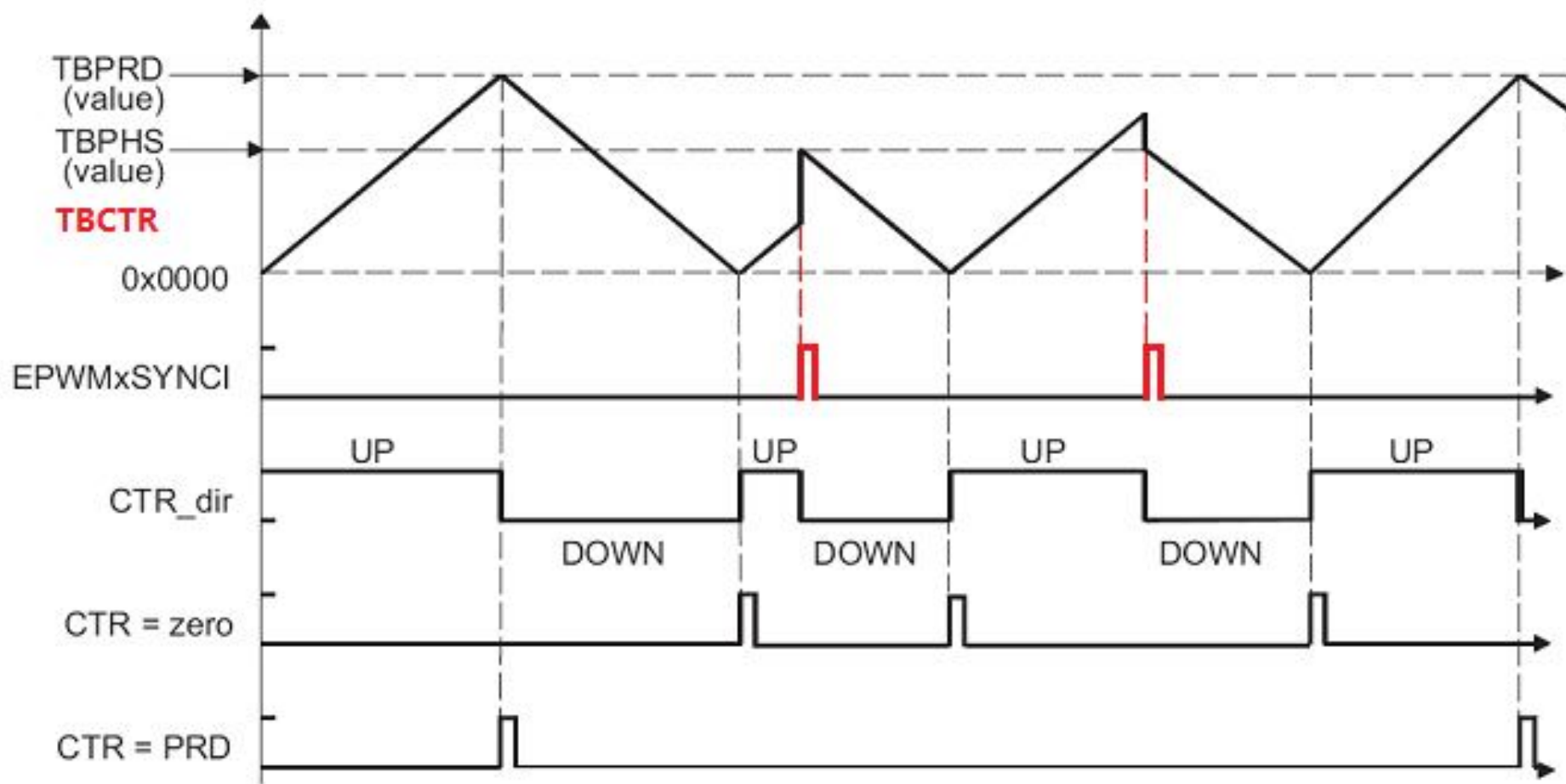


## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 7、时基 (TB) 模块的时序波形 - 先递增后递减

TBCTL[PHSDIR = 0]



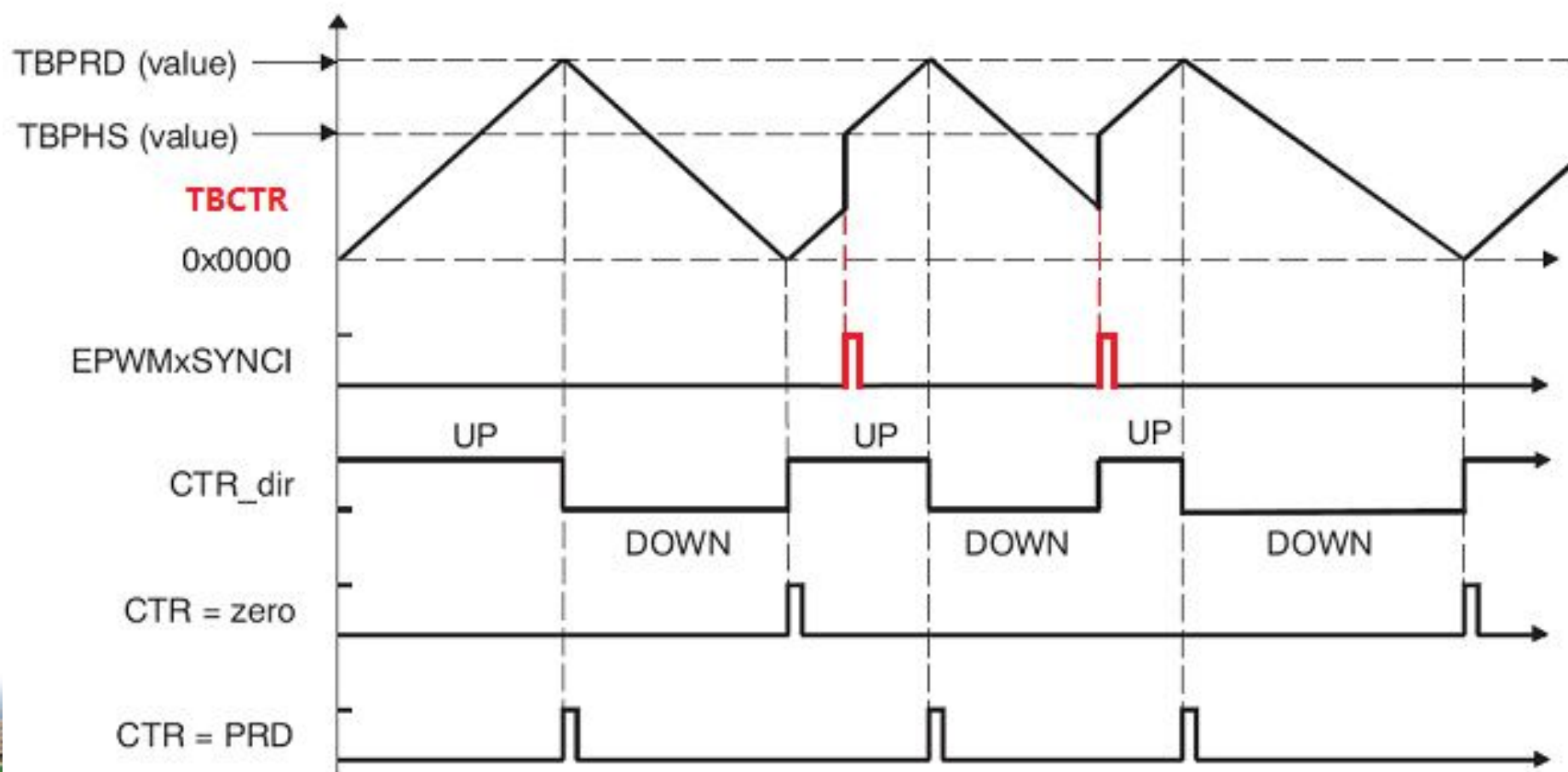


## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 7、时基 (TB) 模块的时序波形 - 先递增后递减

TBCTL[PHSDIR = 1]





## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 8、时基 (TB) 模块的寄存器 -- TBPRD



➤ BIT15-0: 时基周期值。0000-FFFFh

若TBCTL.PRDL D = 0, TBPRD=shadowREG

若TBCTL.PRDL D = 1, TBPRD=activeREG

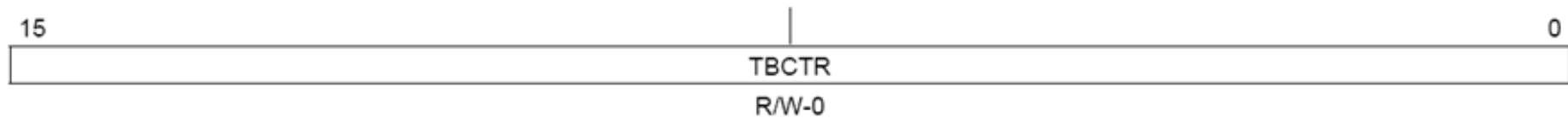




## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 8、时基 (TB) 模块的寄存器 -- TBCTR



- BIT15-0: 时基计数器的当前值。0000 – FFFFh



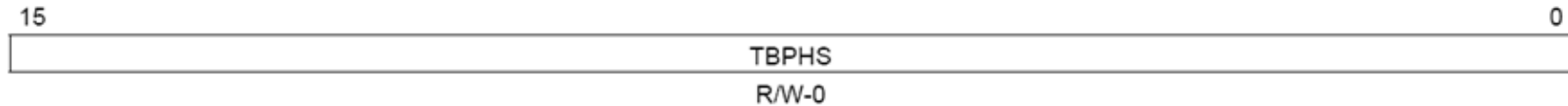




## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 8、时基 (TB) 模块的寄存器 -- TBPHS



➤ BIT15-0: 时基相位值。0000 – FFFFh





## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 8、时基 (TB) 模块的寄存器 -- TBCTL

15	14	13	12	10	9	8	
FREE, SOFT		PHSDIR		CLKDIV	HSPCLKDIV		
R/W-0		R/W-0		R/W-0	R/W-0,0,1		
7	6	5	4	3	2	1	0
HSPCLKDIV	SWFSYNC	SYNCSEL		PRDLD	PHSEN	CTRMODE	
R/W-0,0,1	R/W-0	R/W-0		R/W-0	R/W-0	R/W-11	

➤ **BIT15-14: 仿真模式位。**

**00** - 在下一次时基计数器递增或递减之后停止。

**01** - 在计数器完成一个完整周期后停止。

**1x** - 自由运行。





## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 8、时基 (TB) 模块的寄存器 -- TBCTL

15	14	13	12	10	9	8	
FREE, SOFT		PHSDIR		CLKDIV	HSPCLKDIV		
R/W-0		R/W-0		R/W-0	R/W-0,0,1		
7	6	5	4	3	2	1	0
HSPCLKDIV	SWFSYNC	SYNCOSEL		PRDLD	PHSEN	CTRMODE	
R/W-0,0,1	R/W-0	R/W-0		R/W-0	R/W-0	R/W-11	

➤ **BIT13: 相位方向位**。该位仅在“先递增后递减”模式有效。

0 -在同步事件后递减计数。

1 -在同步事件后递增计数。





## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 8、时基 (TB) 模块的寄存器 -- TBCTL

15	14	13	12	10	9	8	
FREE, SOFT		PHSDIR		CLKDIV	HSPCLKDIV		
R/W-0		R/W-0		R/W-0	R/W-0,0,1		
7	6	5	4	3	2	1	0
HSPCLKDIV	SWFSYNC	SYNCOSEL		PRDLD	PHSEN	CTRMODE	
R/W-0,0,1	R/W-0	R/W-0		R/W-0	R/W-0	R/W-11	

➤ **BIT12-10: 时基时钟预分频位。**

**000 – CLKDIV=1 (默认)      001 - CLKDIV=2**

**010 – CLKDIV=4                011 - CLKDIV=8**

**100 – CLKDIV=16              101 - CLKDIV=32**

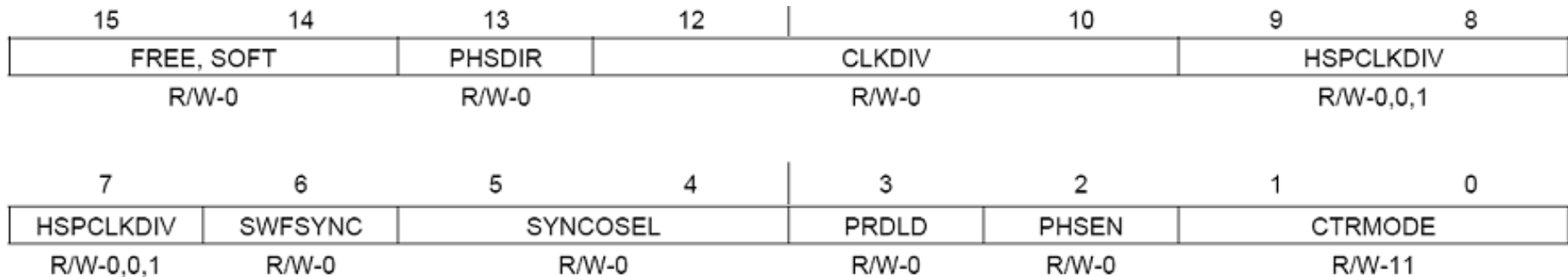
**110 – CLKDIV=64               111 - CLKDIV=128**



## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 8、时基 (TB) 模块的寄存器 -- TBCTL



➤ **BIT9-7: 高速时基时钟预分频位。**

**000 – HSPCLKDIV=1**

**001 - HSPCLKDIV=2 (默认)**

**010 – HSPCLKDIV=4**

**011 - HSPCLKDIV=6**

**100 – HSPCLKDIV=8**

**101 - HSPCLKDIV=10**

**110 – HSPCLKDIV=12**

**111 - HSPCLKDIV=14**





## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 8、时基 (TB) 模块的寄存器 -- TBCTL

15	14	13	12	10	9	8	
FREE, SOFT		PHSDIR		CLKDIV	HSPCLKDIV		
R/W-0		R/W-0		R/W-0	R/W-0,0,1		
7	6	5	4	3	2	1	0
HSPCLKDIV	SWFSYNC	SYNCOSEL		PRDLD	PHSEN	CTRMODE	
R/W-0,0,1	R/W-0	R/W-0		R/W-0	R/W-0	R/W-11	

$$\text{TBCLK} = \text{SYSCLKOUT} / (\text{HSPCLKDIV} \times \text{CLKDIV})$$





## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 8、时基 (TB) 模块的寄存器 -- TBCTL

15	14	13	12	10	9	8	
FREE, SOFT		PHSDIR		CLKDIV	HSPCLKDIV		
R/W-0		R/W-0		R/W-0	R/W-0,0,1		
7	6	5	4	3	2	1	0
HSPCLKDIV	SWFSYNC	SYNCOSSEL		PRDLD	PHSEN	CTRMODE	
R/W-0,0,1	R/W-0	R/W-0		R/W-0	R/W-0	R/W-11	

- **BIT6: 软件强制同步脉冲位。** 写0无影响，读取时返回0  
写1强制产生一个一次 (**one-time**) 同步脉冲  
该位仅在**EPWMxSYNCI**被**SYNCOSSEL = 00**选中时有效





## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 8、时基 (TB) 模块的寄存器 -- TBCTL

15	14	13	12	10	9	8	
FREE, SOFT		PHSDIR	CLKDIV		HSPCLKDIV		
R/W-0		R/W-0	R/W-0		R/W-0,0,1		
7	6	5	4	3	2	1	0
HSPCLKDIV	SWFSYNC	SYNCOSEL		PRDLD	PHSEN	CTRMODE	
R/W-0,0,1	R/W-0	R/W-0		R/W-0	R/W-0	R/W-11	

➤ **BIT5-4: 同步输出选择位。** 选择EPWMxSYNCO信号的源。

**00 – EPWMxSYNC**

**01 – CTR = 0: TBCTR = 0x0000**

**10 – CTR = CMPB: TBCTR = CMPB**

**11 – 禁用EPWMxSYNCO信号**



## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 8、时基 (TB) 模块的寄存器 -- TBCTL

15	14	13	12	10	9	8	
FREE, SOFT		PHSDIR		CLKDIV	HSPCLKDIV		
R/W-0		R/W-0		R/W-0	R/W-0,0,1		
7	6	5	4	3	2	1	0
HSPCLKDIV	SWFSYNC	SYNCOSEL		PRDL	PHSEN	CTRMODE	
R/W-0,0,1	R/W-0	R/W-0		R/W-0	R/W-0	R/W-11	

➤ **BIT3: 周期寄存器加载位。**

写0 – shadow寄存器有效。 (TBPRD=shadowREG)

写1 – shadow寄存器无效。 (TBPRD=activeREG)





## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 8、时基 (TB) 模块的寄存器 -- TBCTL

15	14	13	12	10	9	8	
FREE, SOFT		PHSDIR		CLKDIV	HSPCLKDIV		
R/W-0		R/W-0		R/W-0	R/W-0,0,1		
7	6	5	4	3	2	1	0
HSPCLKDIV	SWFSYNC	SYNCOSEL		PRDLD	PHSEN	CTRMODE	
R/W-0,0,1	R/W-0	R/W-0		R/W-0	R/W-0	R/W-11	

➤ **BIT2: 相位控制使能位。**

写**0** – 禁止相位控制，同步功能忽略。

写**1** – 使能相位控制。有同步信号是**TBCTR=TBPHS**。







## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 8、时基 (TB) 模块的寄存器 -- TBCTL

15	14	13	12	10	9	8	
FREE, SOFT		PHSDIR		CLKDIV	HSPCLKDIV		
R/W-0		R/W-0		R/W-0	R/W-0,0,1		
7	6	5	4	3	2	1	0
HSPCLKDIV	SWFSYNC	SYNCOSEL		PRDLD	PHSEN	CTRMODE	
R/W-0,0,1	R/W-0	R/W-0		R/W-0	R/W-0	R/W-11	

➤ **BIT1-0: 计数模式选择位。**

**00** – 递增计数模式。

**01** – 递减计数模式。

**10** – 先递增后递减。

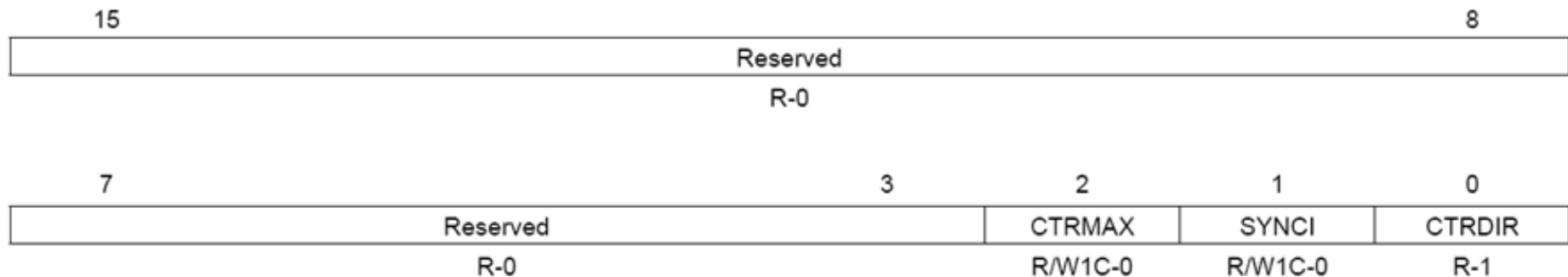
**11** – 停止-停顿模式 (复位时的默认值)



## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 8、时基 (TB) 模块的寄存器 -- TBSTS



➤ **BIT2: 时基计数器最大值锁存状态位。**

为**0**: TBCTR从没到达过最大值。写**0**无反应。

为**1**: TBCTR到达最大值**0xFFFF**。写**1**会清除**0**。

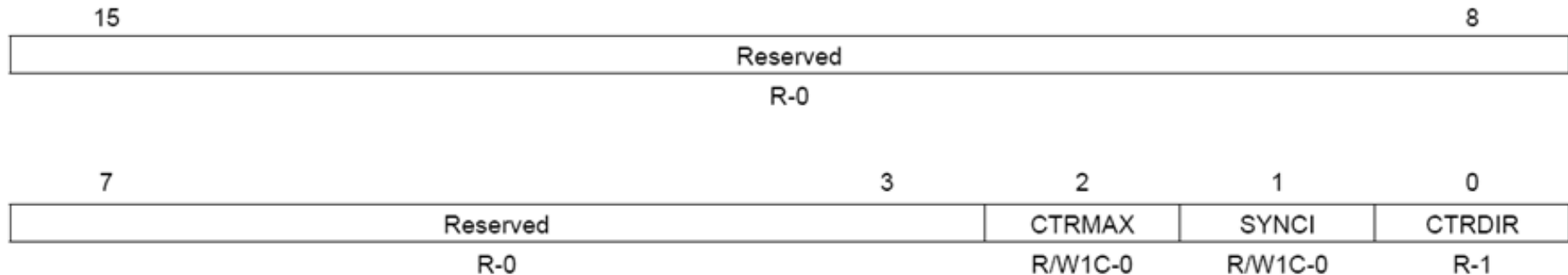




## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 8、时基 (TB) 模块的寄存器 -- TBSTS



➤ **BIT1: 输入同步锁存状态位。**

写**0**没有作用。写**1**会清除已被锁存事件。

读为**0**时，表示没有发生外部同步事件。

读为**1**时，表示发生了一个外部同步事件。

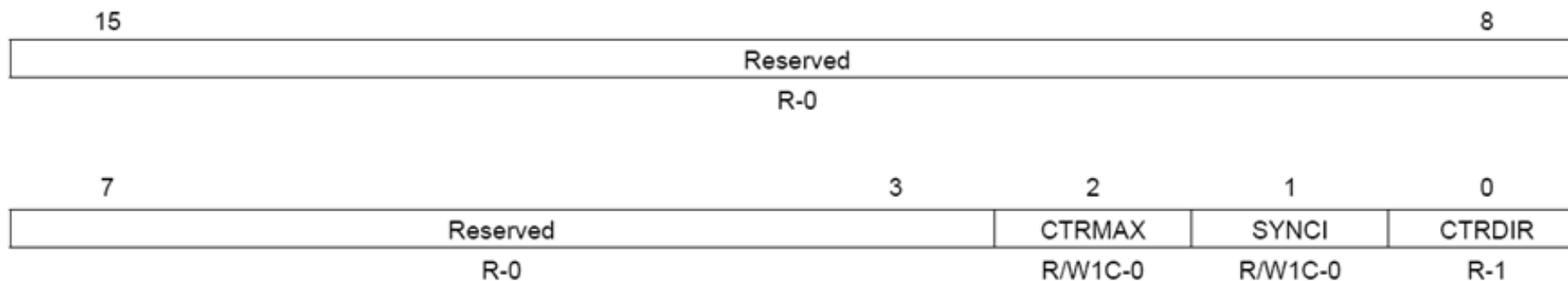




## Lecture7 ePWM

### 三、ePWM子模块功能和配置 -- 时基 (TB)

#### 8、时基 (TB) 模块的寄存器 -- TBSTS



➤ **BIT0: 时基计数器方向状态位。** 只读位。

**0** –当前正在递减。

**1** –当前正在递增。





## Lecture7 ePWM

### 四、ePWM子模块功能和配置 -- 计数器-比较 (CC)

#### 1、计数器-比较 (CC) 的功能

- 指定ePWMxA和/或ePWMxB输出信号的PWM占空比
- 指定ePWMxA或ePWMxB输出信号上发生事件开关的时间



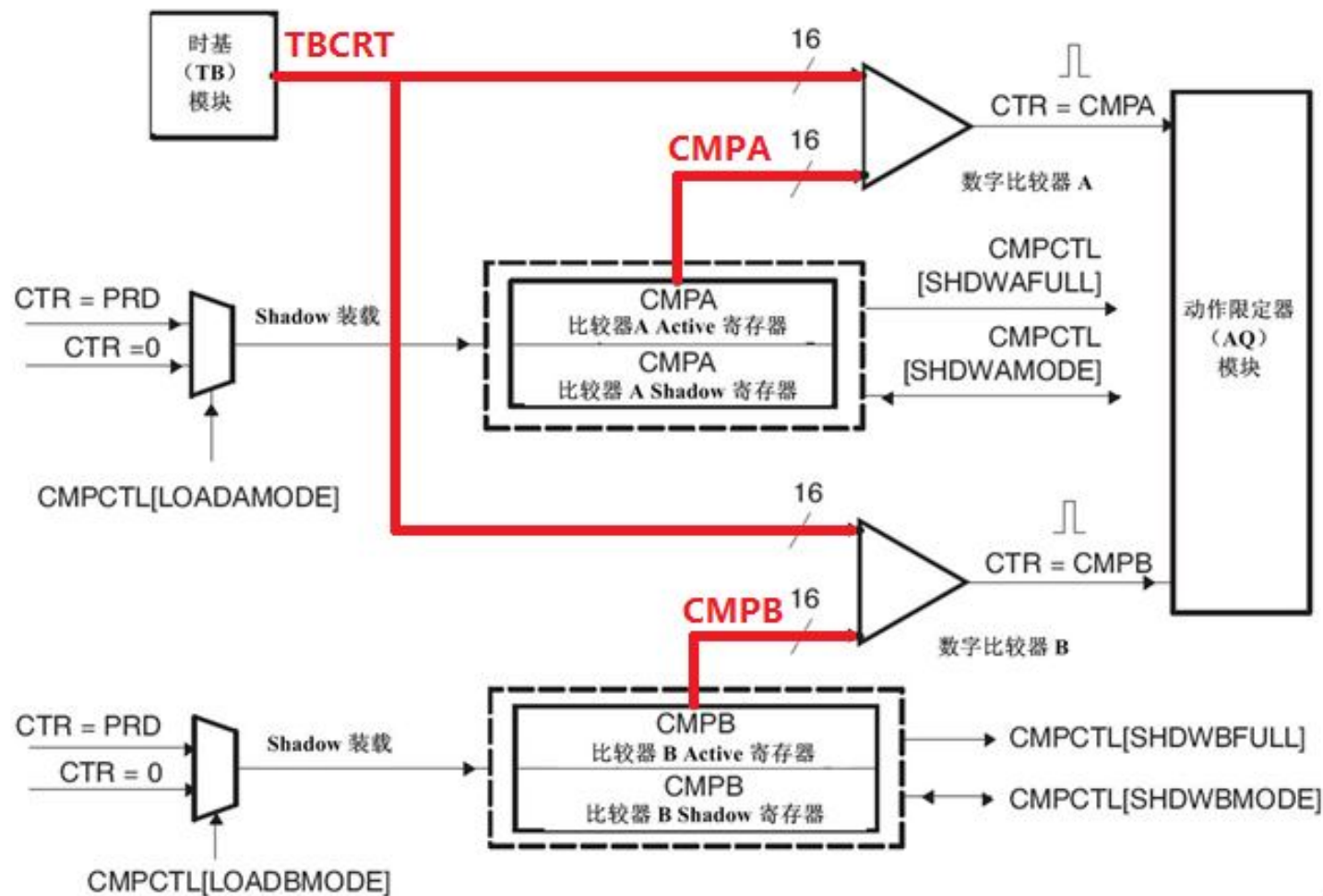




# Lecture7 ePWM

## 四、ePWM子模块功能和配置 -- 计数器-比较 (CC)

### 2、计数器-比较 (CC) 的组成





## Lecture7 ePWM

### 四、ePWM子模块功能和配置 -- 计数器-比较 (CC)

#### 3、计数器-比较 (CC) 的主要信号

信号	描述
CTR = CMPA	TBCTR = CMPA 时基计数器的计数值等于计数器-比较A active寄存器的值
CTR = CMPB	TBCTR = CMPA 时基计数器的计数值等于计数器-比较B active寄存器的值
CTR = PRD	TBCTR = TBPRD: 用于从shadow寄存器那里装载active计数器-比较A和B寄存器
CTR = ZERO	TBCTR = 0x0000: 用于从shadow寄存器那里装载active计数器-比较A和B寄存器





## Lecture7 ePWM

### 四、ePWM子模块功能和配置 -- 计数器-比较 (CC)

#### 4、计数器-比较 (CC) 的shadow寄存器

**使能:** `CMPCTL.SHDWAMODE=1` 和 `CMPCTL.SHDWBMODE=1`

发生以下事件时, shadow寄存器的值送到active寄存器:

-- `CTR = PRD`

-- `CTR = ZERO`

-- `CTR = PRD` 和 `CTR = ZERO`

通过`CMPCTL.LOADAMODE`和`CMPCTL.LOADBMODE`设置

**禁止:** `CMPCTL.SHDWAMODE=0` 和 `CMPCTL.SHDWBMODE=0`

`CMPA/CMPB`直接指向active寄存器。

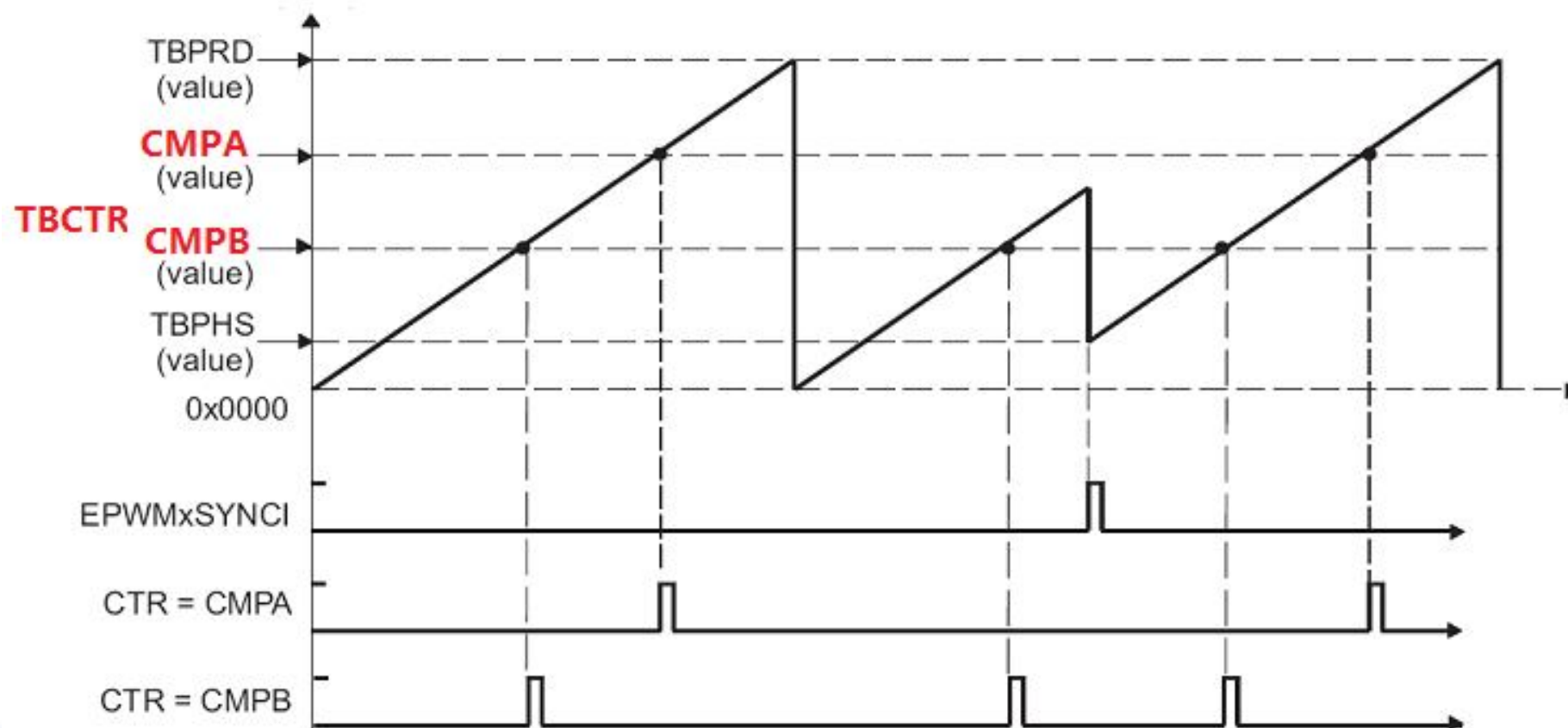




## Lecture7 ePWM

### 四、ePWM子模块功能和配置 -- 计数器-比较 (CC)

### 5、计数器-比较 (CC) 的时序波形 -- 递增

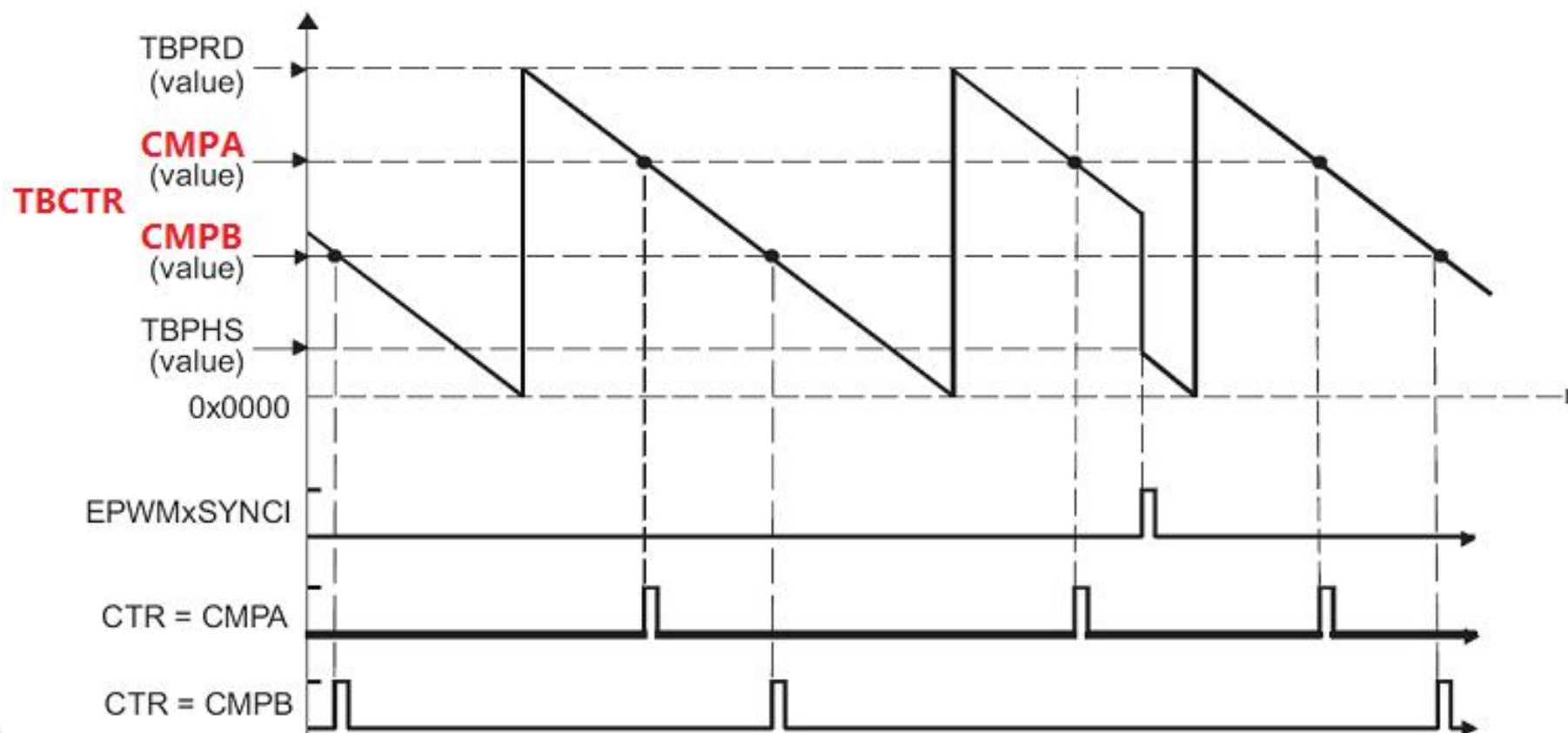




## Lecture7 ePWM

### 四、ePWM子模块功能和配置 -- 计数器-比较 (CC)

### 5、计数器-比较 (CC) 的时序波形 -- 递减



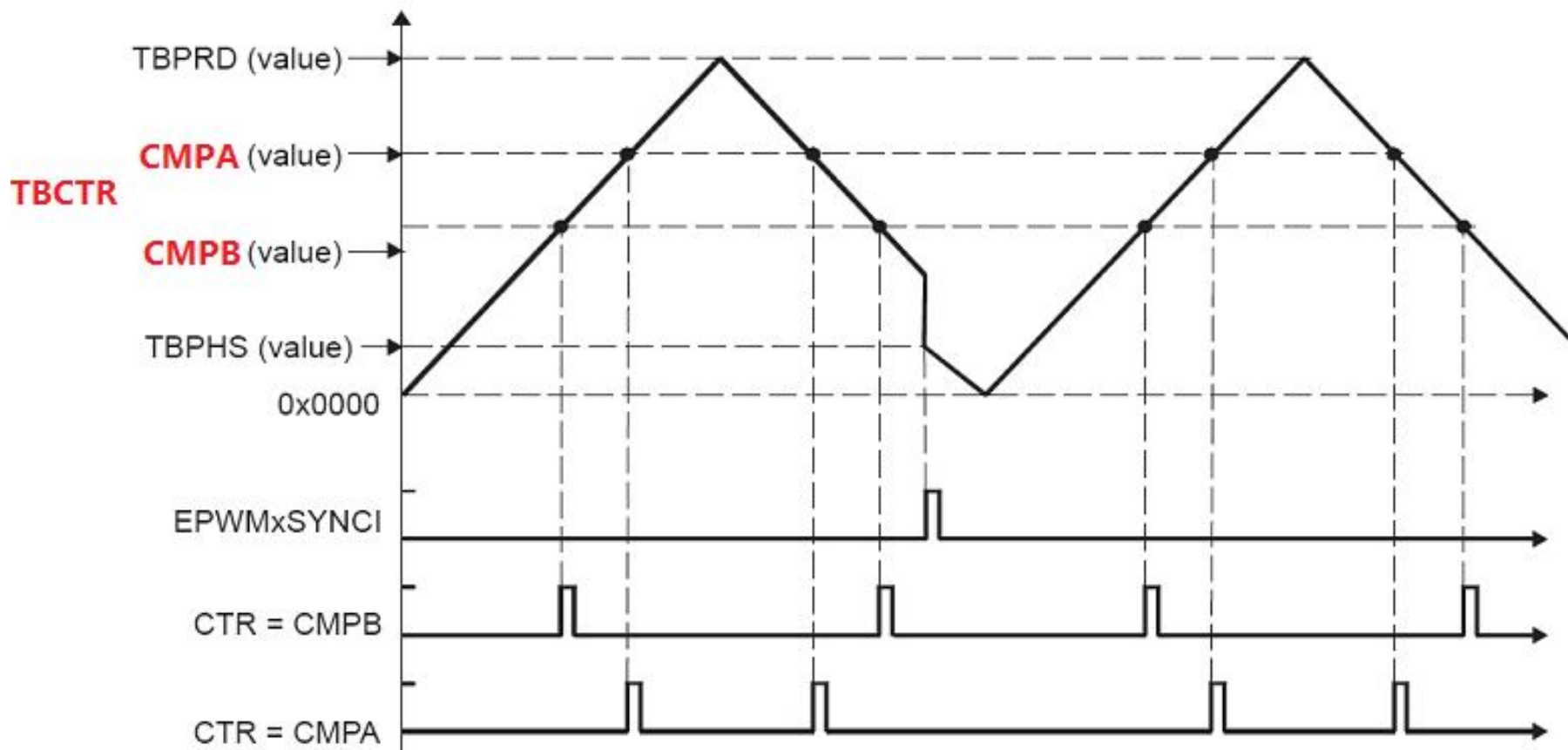




# Lecture7 ePWM

## 四、ePWM子模块功能和配置 -- 计数器-比较 (CC)

### 5、计数器-比较 (CC) 的时序波形 -- 先递增后递减



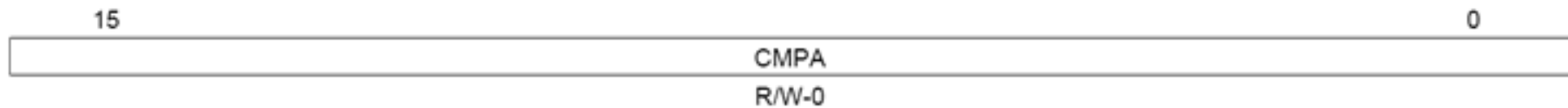




## Lecture7 ePWM

### 四、ePWM子模块功能和配置 -- 计数器-比较 (CC)

#### 6、计数器-比较 (CC) 的寄存器 --CMPA



➤ BIT15-0: 比较值A。0000 – FFFFh

若CMPCTL.SHDWAMODE = 0, CMPA=shadow

若CMPCTL.SHDWAMODE = 1, CMPA=active

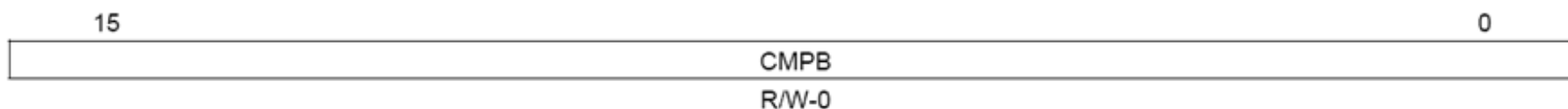




## Lecture7 ePWM

### 四、ePWM子模块功能和配置 -- 计数器-比较 (CC)

#### 6、计数器-比较 (CC) 的寄存器 --CMPB



➤ BIT15-0: **比较值B**。0000 – FFFFh

若CMPCTL.SHDWBMODE = 0, CMPB=shadow

若CMPCTL.SHDWBMODE = 1, CMPB=active





## Lecture7 ePWM

### 四、ePWM子模块功能和配置 -- 计数器-比较 (CC)

#### 6、计数器-比较 (CC) 的寄存器 --CMPCTL

15				10				9	8
Reserved								SHDWBFULL	SHDWAFULL
R-0								R-0	R-0
7		6	5	4	3	2	1	0	
Reserved	SHDWBMODE	Reserved	SHDWAMODE	LOADBMODE		LOADAMODE			
R-0	R/W-0	R-0	R/W-0	R/W-0		R/W-0			

➤ **BIT9: (CMPB) shadow寄存器满状态标志。** (只读)

有新值送到shadow时，该位置1。

shadow值送到active时，该位清零。

➤ **BIT8: (CMPA) shadow寄存器满状态标志。** (只读)

有新值送到shadow时，该位置1。

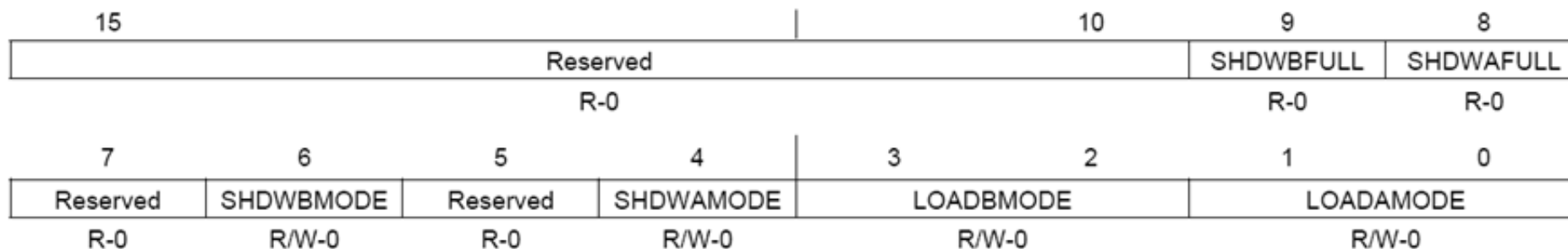
shadow值送到active时，该位清零。



## Lecture7 ePWM

### 四、ePWM子模块功能和配置 -- 计数器-比较 (CC)

#### 6、计数器-比较 (CC) 的寄存器 --CMPCTL



➤ **BIT6: (CMPB) shadow寄存器模式位。**

0 -- 使能shadow。

1 -- 禁止shadow。

➤ **BIT4: (CMPA) shadow寄存器模式位。**

0 -- 使能shadow。

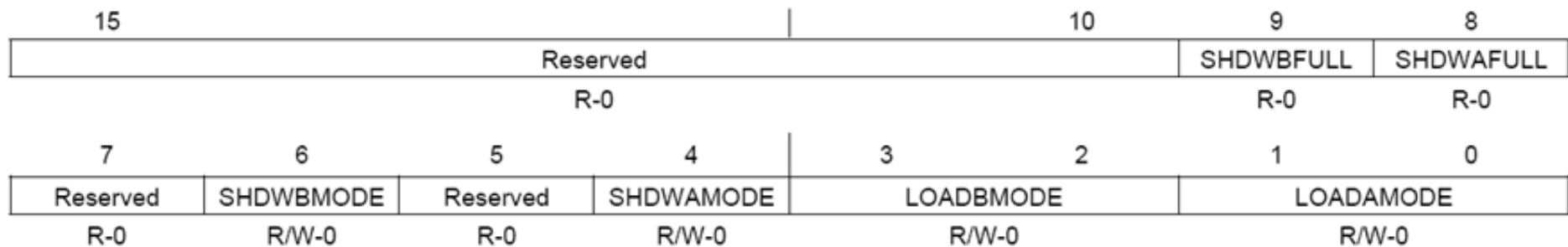
1 -- 禁止shadow。



## Lecture7 ePWM

### 四、ePWM子模块功能和配置 -- 计数器-比较 (CC)

#### 6、计数器-比较 (CC) 的寄存器 --CMPCTL



➤ **BIT3-2: (CMPB) active从shadow装载的模式选择位。**

00 -- CTR=0时装载

01 -- CTR=PRD时装载

10 -- CTR=0 和PRD时装载

11 -- 不装载

➤ **BIT1-0: (CMPA) active从shadow装载的模式选择位。**

00 -- CTR=0时装载

01 -- CTR=PRD时装载

10 -- CTR=0 和PRD时装载

11 -- 不装载



## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

#### 1、动作限定器 (AQ) 的功能

- 指定在发生时基和比较事件时PWM输出引脚的**动作**:
  - 不采取任何动作
  - EPWMxA和/或EPWMxB输出信号变为高电平
  - EPWMxA和/或EPWMxB输出信号变为低电平
  - EPWMxA和/或EPWMxB输出信号来回切换 (toggle)
- 通过软件控制**强制**改变PWM输出信号的状态
- 通过软件来配置和控制PWM**死区**



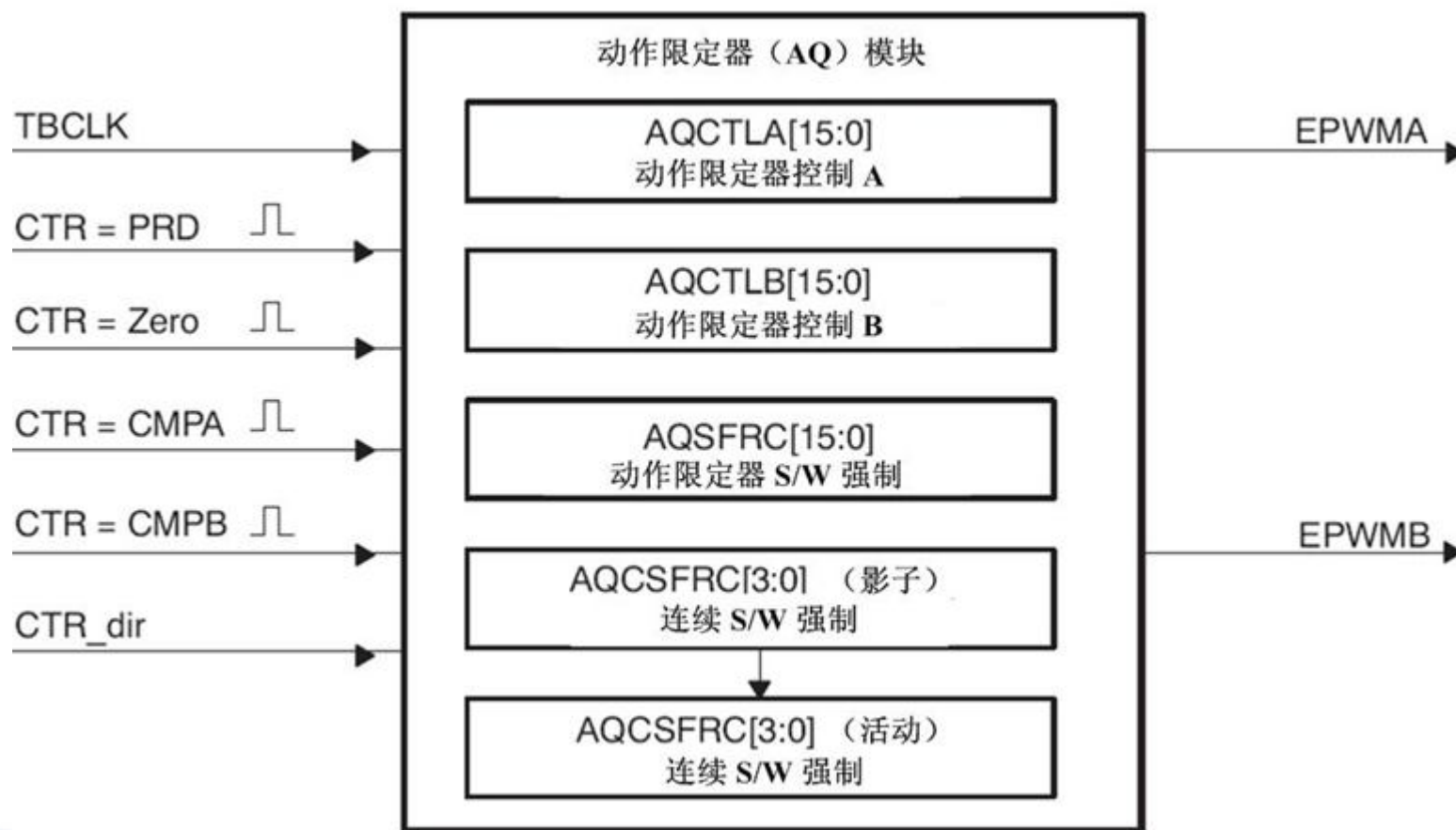




## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

#### 2、动作限定器 (AQ) 的组成





## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

#### 3、动作限定器 (AQ) 的信号

信号	描述
CTR = PRD	TBCTR = TBPRD时基计数器的计数值等于周期值
CTR = ZERO	TBCTR = 0x0000时基计数器的计数值等于零
CTR = CMPA	TBCTR = CMPA时基计数器的计数值等于计数器-比较A
CTR = CMPB	TBCTR = CMPB时基计数器的计数值等于计数器-比较B
软件强制事件	由软件发起的异步事件

以上信号都可以改变PWM输出引脚的状态。





## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

#### 4、动作限定器 (AQ) 的动作

EPWMxA和EPWMxB输出信号上的动作有:

- **置 1** (高电平) : 输出信号设置成高电平
- **清零** (低电平) : 输出信号设为低电平
- **切换** (Toggle) : 输出信号取反
- **Do Nothing** (不动作) : 输出信号保持为当前设置值。

但Do Nothing事件仍然可以触发中断和ADC开始转换。





## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

#### 5、动作限定器 (AQ) 的优先级

原则：事件越后发生优先级越高，软件强制事件优先级最高。

递增计数模式		递减计数模式	
优先级	事件	优先级	事件
1(高)	S/W	1(高)	S/W
2	CTR=TBPRD	2	CTR=0
3	CTR=COMPB(CBU)	3	CTR=COMPB(CBD)
4	CTR=COMPB(CAU)	4	CTR=COMPB(CAD)
5(低)	CTR=0	5(低)	CTR=TBPRD





## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

#### 5、动作限定器 (AQ) 的优先级

原则：事件越后发生优先级越高，软件强制事件优先级最高。

先递增后递减计数模式		
优先级	CTR=0到TBPRD-1	CTR=TBPRD到1
1(高)	S/W	S/W
2	CTR=COMPB (CBU)	CTR=COMPB (CBD)
3	CTR=COMPB (CAU)	CTR=COMPB (CAD)
4(低)	CTR=0	CTR= (TBPRD)





## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

#### 6、CMPA/CMPB大于TBPRD时的行为

计数器模式	递增过程: CAU/CBU	递减过程: CAD/CBD
递增计数模式	若 $CMPA/CMPB \leq TBPRD$ ，可发生该事件。 若 $CMPA/CMPB > TBPRD$ ，不发生该事件。	永不发生
递减计数模式	永不发生	若 $CMPA/CMPB < TBPRD$ ，可发生该事件。 若 $CMPA/CMPB \geq TBPRD$ ，该事件在 $CTR = TBPRD$ 时发生。
先递增后递减计数模式	若 $CMPA/CMPB < TBPRD$ ，该事件在 $CTR = CMPA/B$ 时发生。 若 $CMPA/CMPB \geq TBPRD$ ，该事件在 $CTR = TBPRD$ 时发生。	若 $CMPA/CMPB < TBPRD$ ，该事件在 $CTR = CMP/B$ 时发生。 若 $CMPA/CMPB \geq TBPRD$ ，该事件在 $CTR = TBPRD$ 时发生。





## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

#### 7、PWM波形产生原理

- 使用递增计数模式产生一个**非对称**的PWM
- 使用递减计数模式产生一个**非对称**的PWM
- 使用“先递增后递减”计数模式产生一个**对称**的PWM



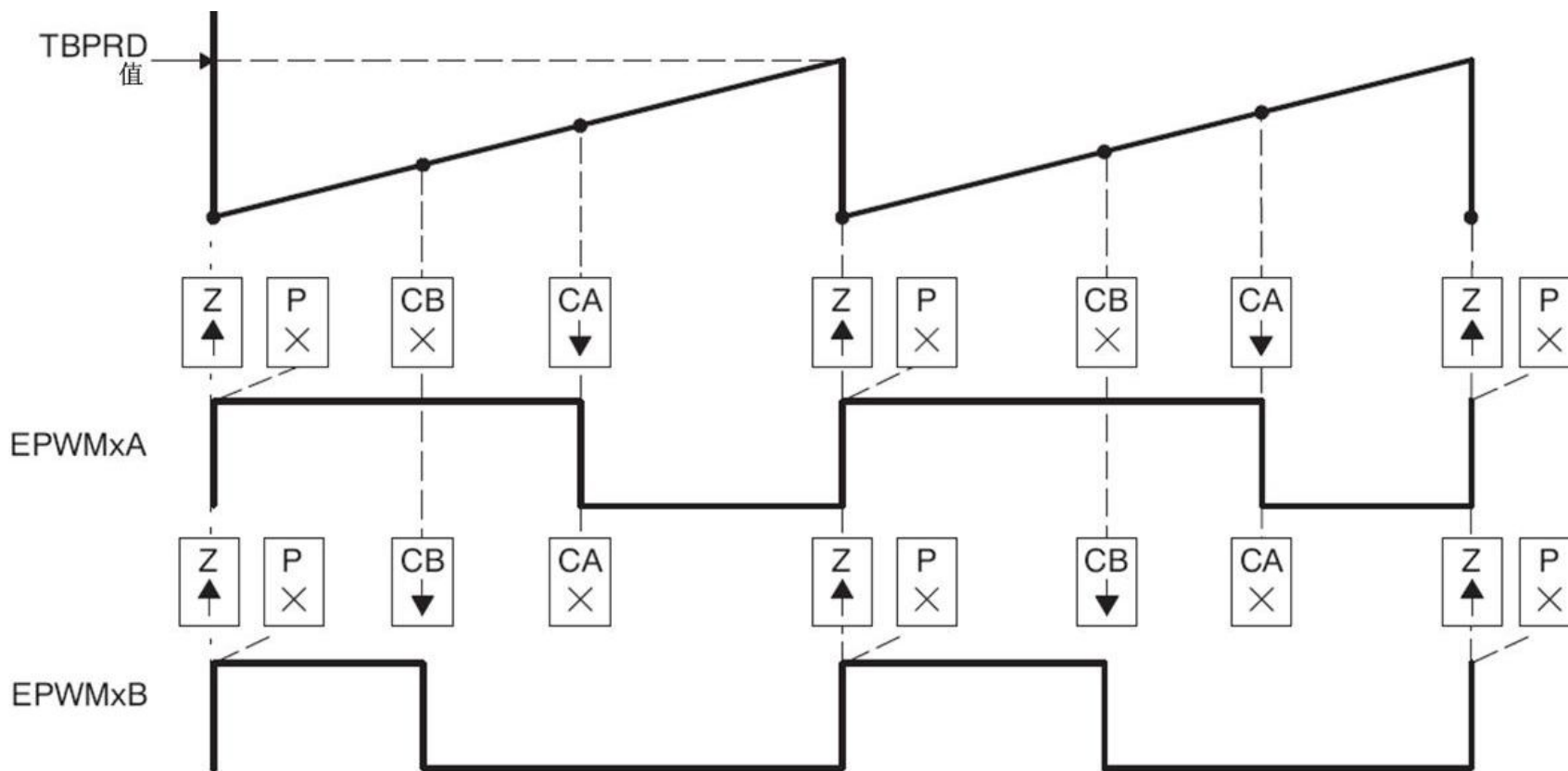


## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

### 7、PWM波形产生原理 -- 非对称PWM

递增、单边沿、单独调制、高有效



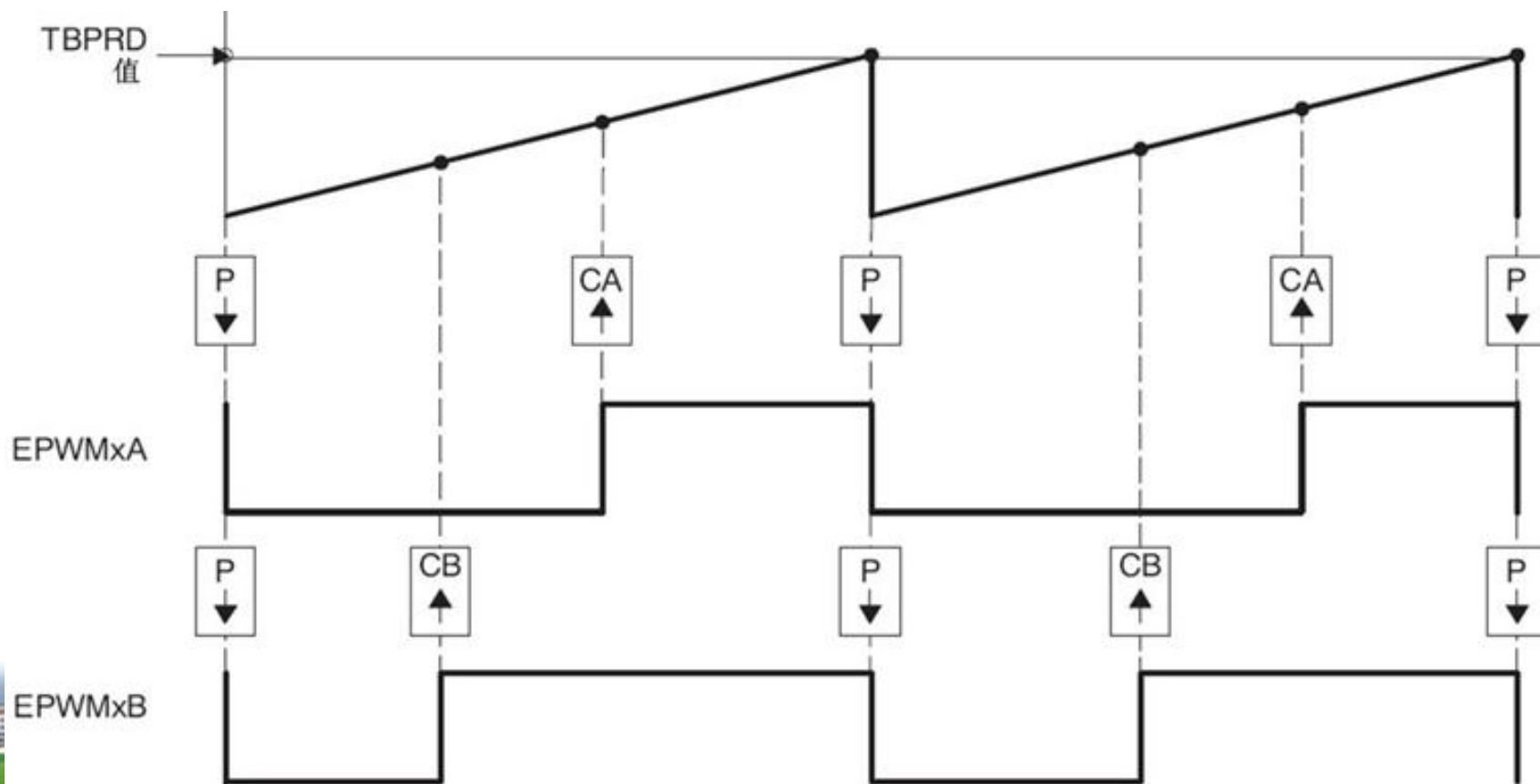


## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

### 7、PWM波形产生原理 -- 非对称PWM

递增、单边沿、单独调制、低有效



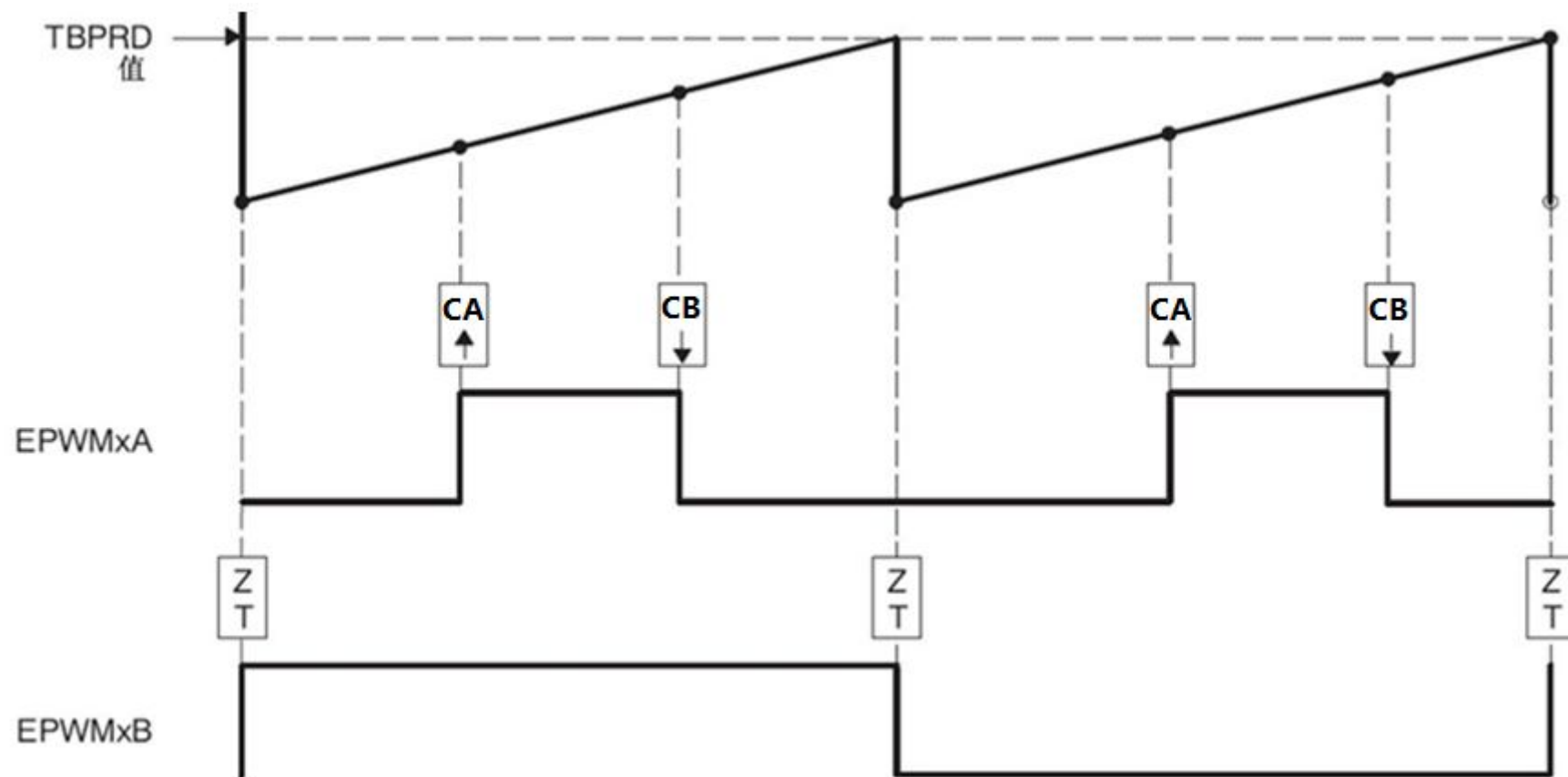


## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

### 7、PWM波形产生原理 -- 非对称PWM

递增、双边沿、单独调制ePWMxA



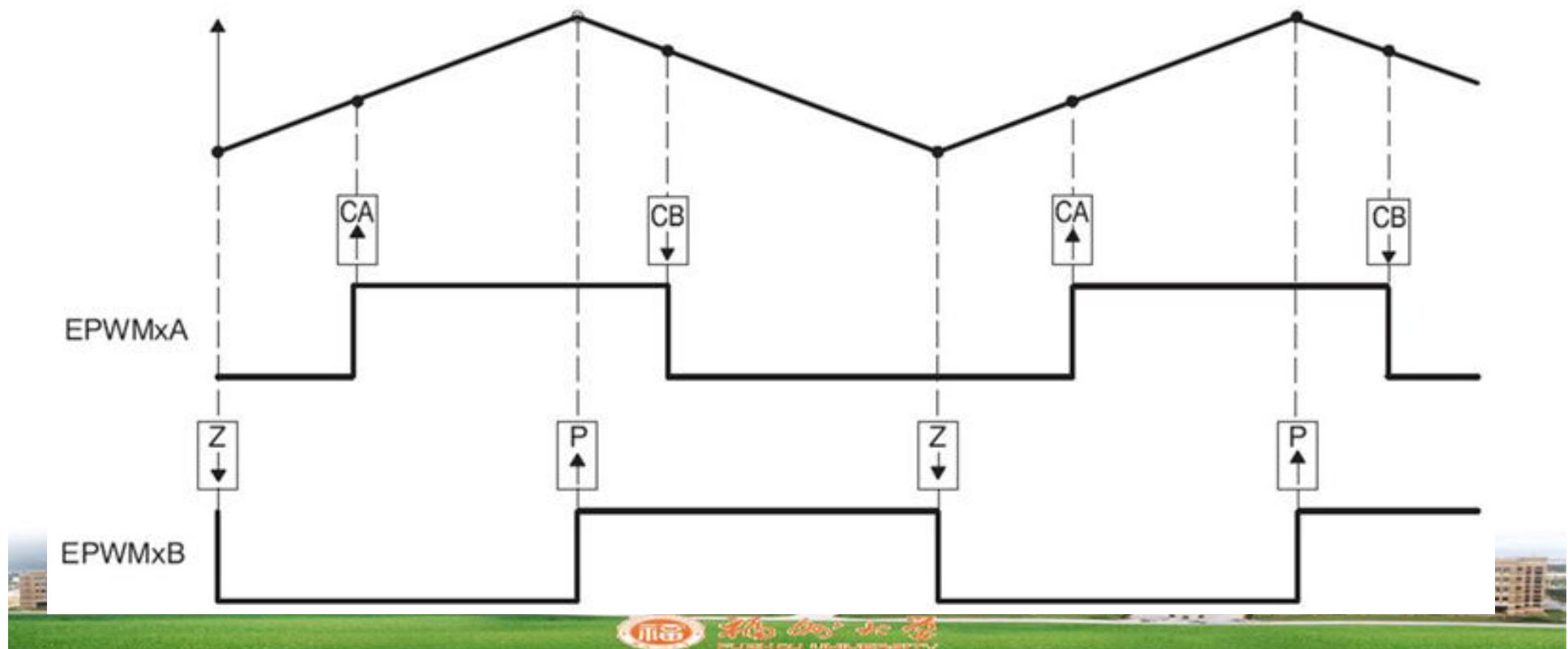


## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

### 7、PWM波形产生原理 -- 非对称PWM

先递增后递减、双边沿非对称、单独调制、低电平有效



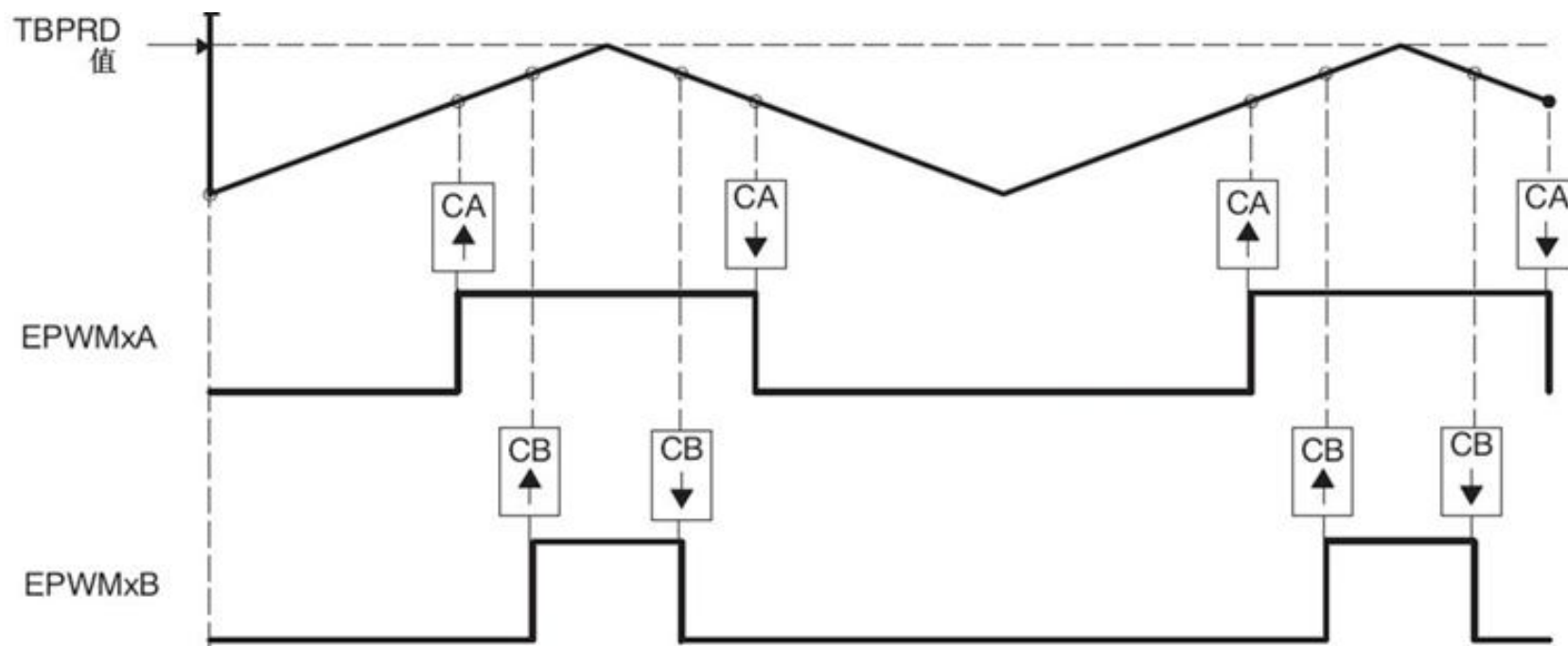


## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

### 7、PWM波形产生原理 -- 对称PWM

先递增后递减、双边沿、单独调制、低电平有效





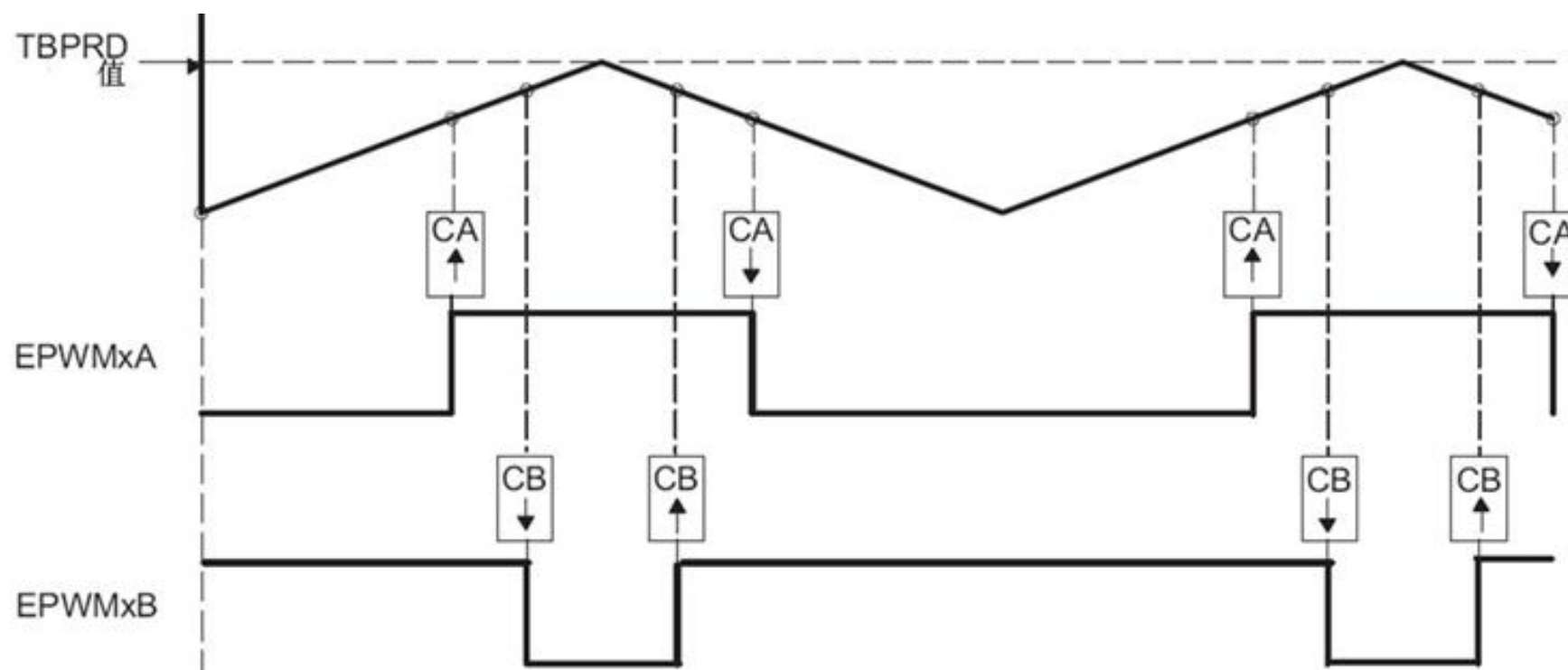


## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

### 7、PWM波形产生原理 -- 对称PWM

先递增后递减、双边沿、互补、低电平有效

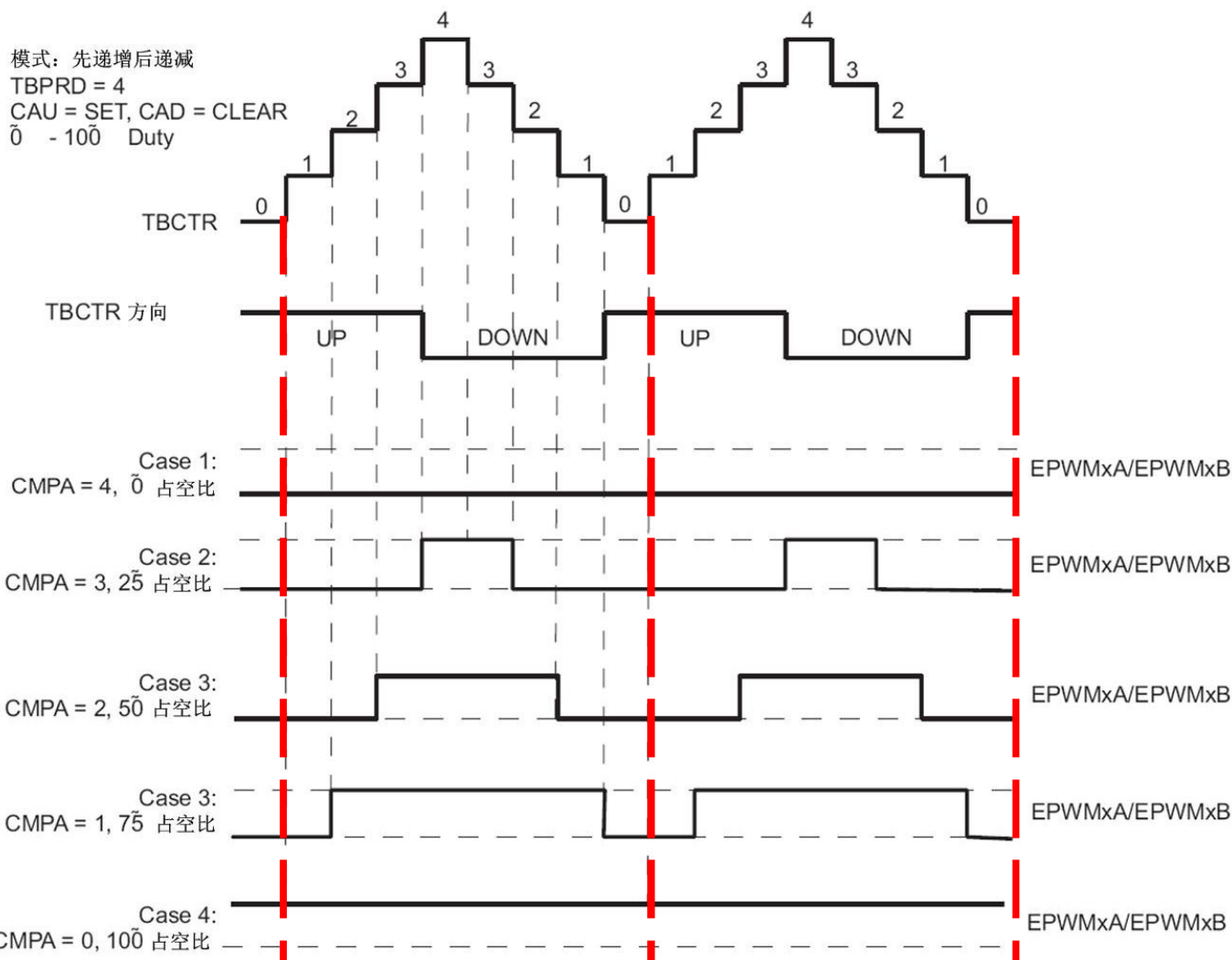




# Lecture7 ePWM

## 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

### 7、PWM波形产生原理 -- 对称PWM

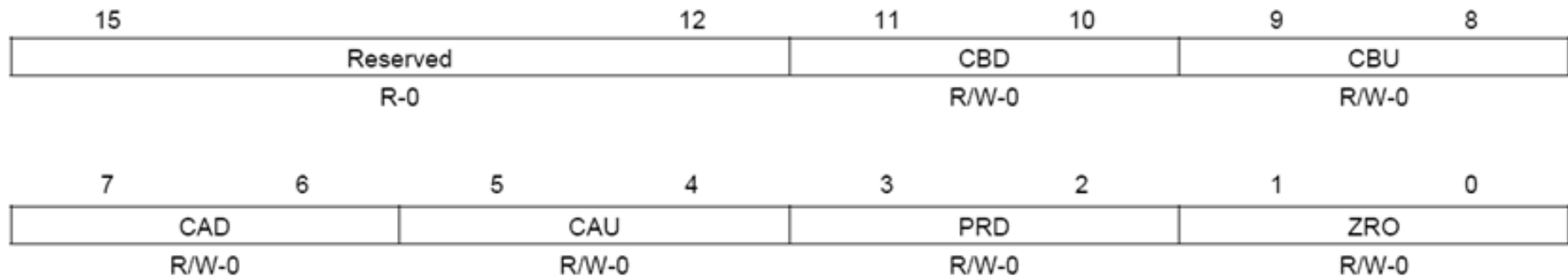




## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

#### 8、动作限定器 (AQ) 的寄存器 -- AQCTLA



➤ ePWMxA 引脚输出电平动作 (ZRO、PRD、CAU、CAD、CBU、CBD) :

00 – NoThing (不动作)

01 --清零

10 --置位

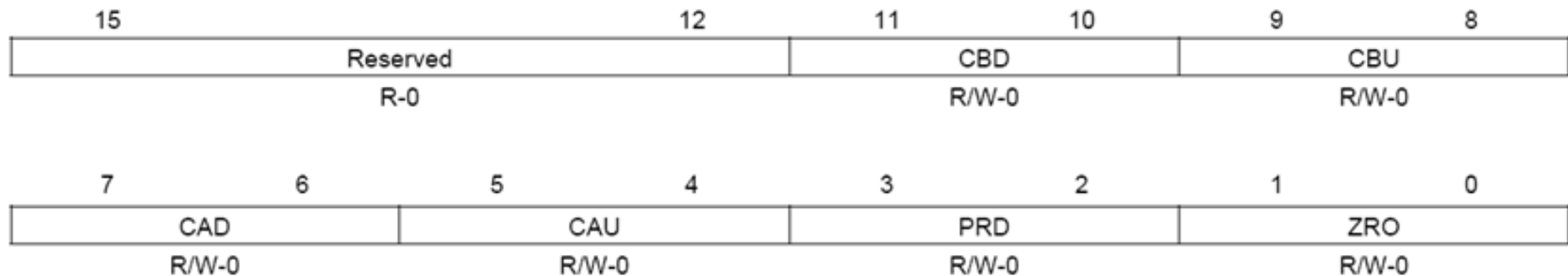
11 --切换



## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

#### 8、动作限定器 (AQ) 的寄存器 -- AQCTLB



➤ ePWMxB引脚输出电平动作 (ZRO、PRD、CAU、CAD、CBU、CBD) :

00 – NoThing (不动作)

01 --清零

10 --置位

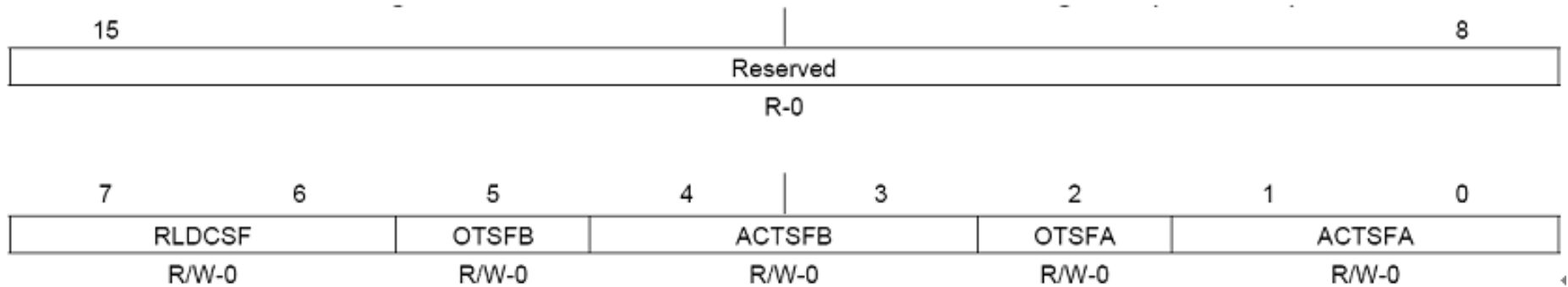
11 --切换



## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

### 8、动作限定器 (AQ) 的寄存器 -- AQSFR



➤ **BIT7-6: AQCFSFR从shadow寄存器重载选择位。**

**00** -- 在事件计数器等于0时装载

**01** --在事件计数器等于周期时装载

**10** --在事件计数器等于0或周期时装载

**11** --立即装载

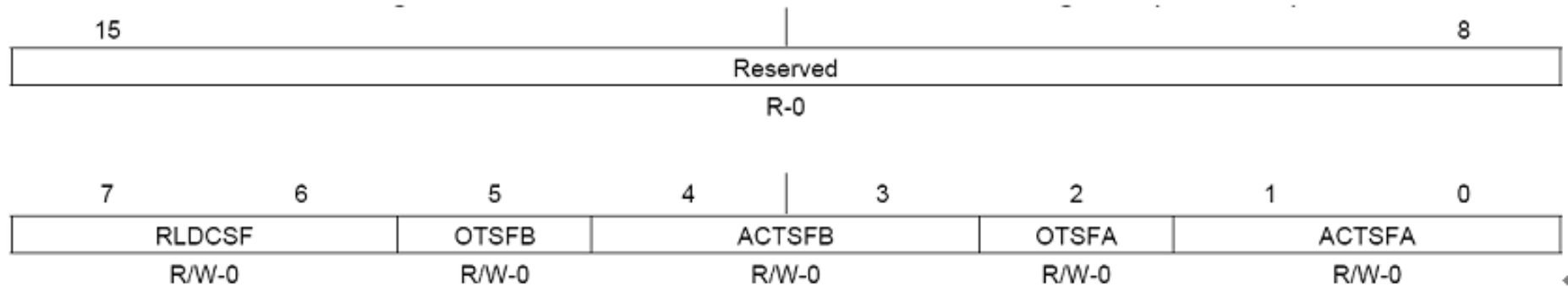




## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

#### 8、动作限定器 (AQ) 的寄存器 -- AQSFR



➤ **BIT5: ePWMxB上的“单次软件 (S/W) 强制事件”。**

写0无效，读总是为0。

写1启动一个单次软件强制事件。

➤ **BIT4-3: ePWMxB单次软件强制事件发生时的动作**

00 – NoThing (不动作)      01 --清零

10 --置位                      11 --切换

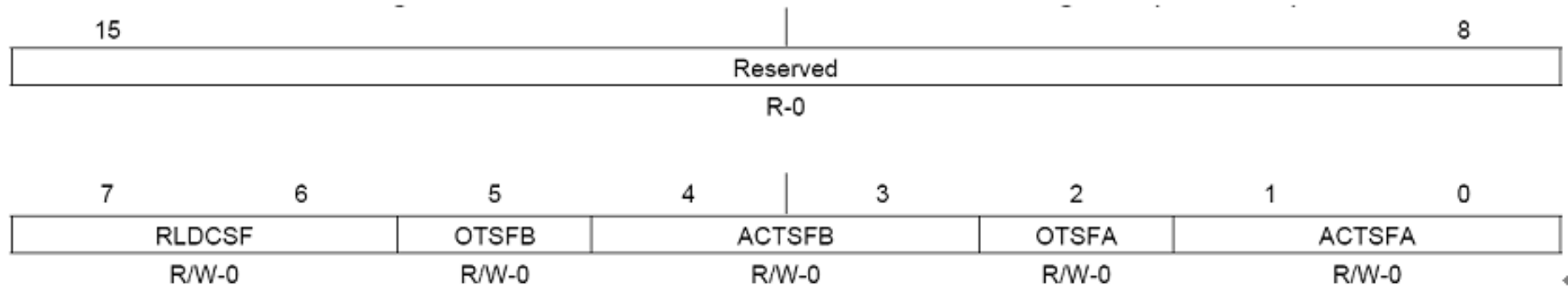




## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

### 8、动作限定器 (AQ) 的寄存器 -- AQSFR



➤ **BIT2: ePWMxA上的“单次软件 (S/W) 强制事件”。**

写0无效，读总是为0。

写1启动一个单次软件强制事件。

➤ **BIT1-0: ePWMxA单次软件强制事件发生时的动作**

00 – NoThing (不动作)      01 --清零

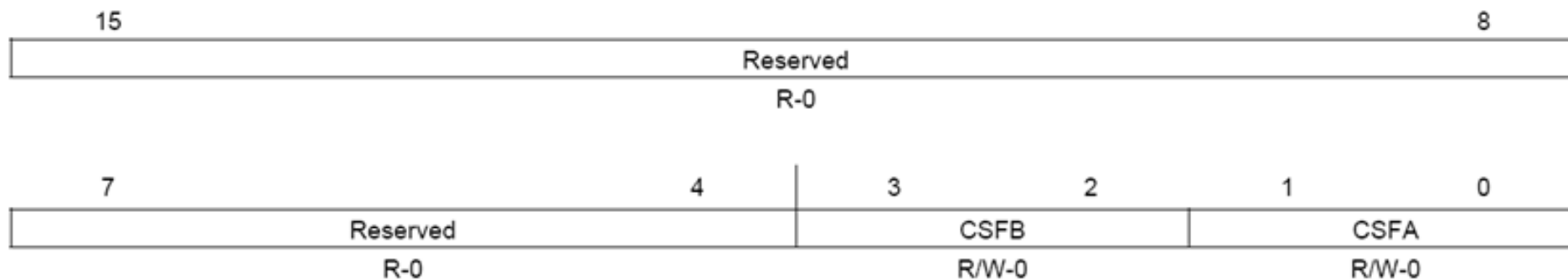
10 --置位                      11 --切换



## Lecture7 ePWM

### 五、ePWM子模块功能和配置 -- 动作限定器 (AQ)

### 8、动作限定器 (AQ) 的寄存器 -- AQCSFRC



➤ **BIT3-2:** ePWMxB连续软件强制位。

➤ **BIT1-0:** ePWMxA连续软件强制位。

**00** - 强制禁止，即无效

**01** -连续将ePWMxA/B输出信号强制变为低电平

**10** -连续将ePWMxA/B输出信号强制变为高电平

**11** -软件强制禁止无效



## Lecture7 ePWM

### 六、ePWM子模块功能和配置 -- 死区 (DB)

#### 1、死区 (DB) 的功能

- 控制上端开关和下端开关之间传统的**死区互补**关系
- 指定输出的**上升沿延迟值**
- 指定输出的**下降沿延迟值**
- 完全**旁路**死区模块。这样，PWM波形将无需修改直接通过
- 使能半周期计时方式，以便获得**双倍分辨率**

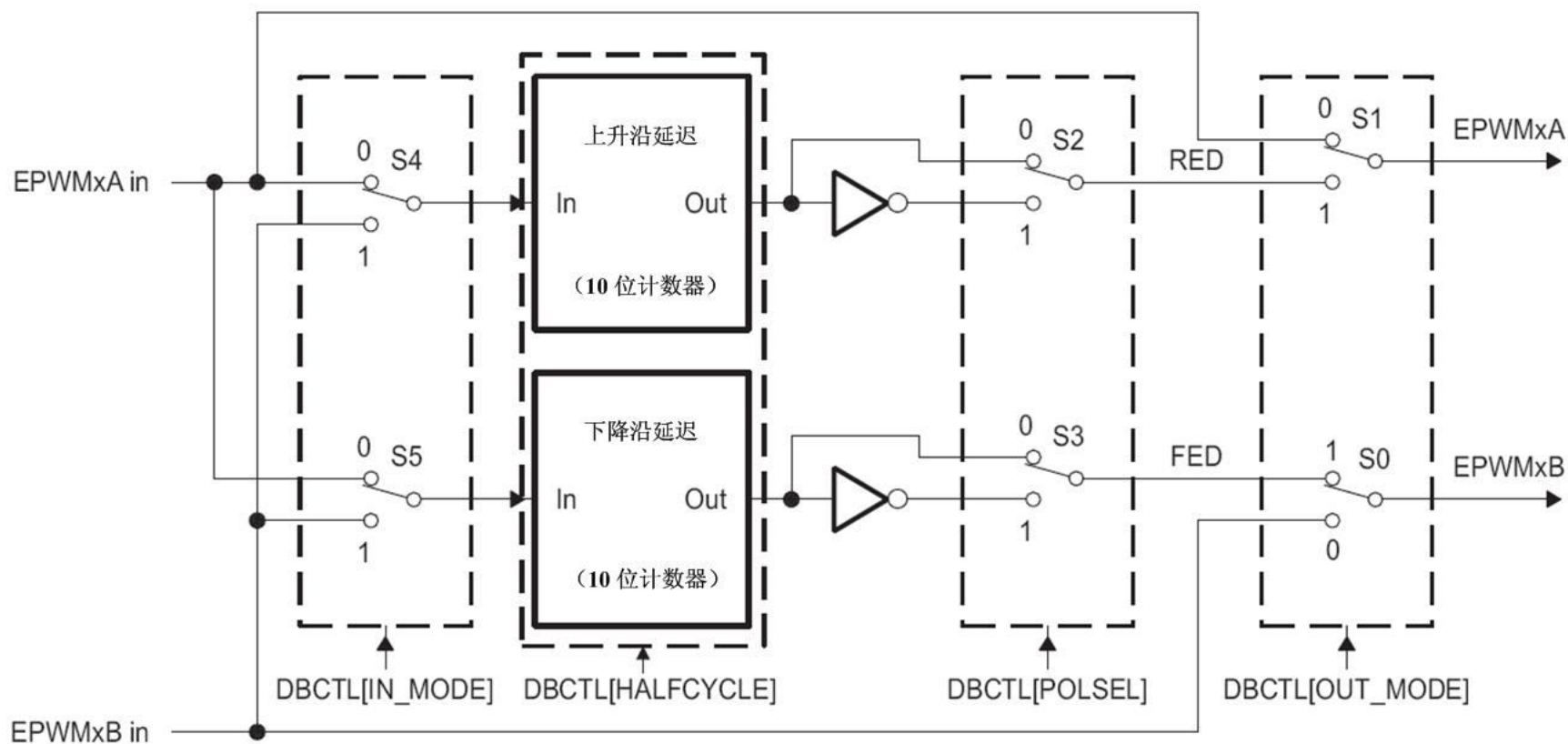




# Lecture7 ePWM

## 六、ePWM子模块功能和配置 -- 死区 (DB)

### 2、死区 (DB) 的组成





## Lecture7 ePWM

### 六、ePWM子模块功能和配置 -- 死区 (DB)

#### 3、死区 (DB) 的操作要点

➤ 输入源选择:

输入模式	上升沿延迟信号源	下降沿延迟信号源
00 (默认)	EPWMxA In	EPWMxA In
01	EPWMxA In	EPWMxB In
10	EPWMxB In	EPWMxA In
11	EPWMxB In	EPWMxB In





## Lecture7 ePWM

### 六、ePWM子模块功能和配置 -- 死区 (DB)

#### 3、死区 (DB) 的操作要点

##### ➤ 半周期计时:

分辨率翻倍。(即计数器以“ $2 \times \text{TBCLK}$ ”计时)

##### ➤ 输出模式控制:

输出模式: 上升沿延迟、下降沿延迟、下降沿与上升沿两种延迟、旁路。(DBCTL.OUT\_MODE位配置)

##### ➤ 极性控制

输出是否反相。(DBCTL.POLSEL位配置)







## Lecture7 ePWM

### 六、ePWM子模块功能和配置 -- 死区 (DB)

#### 4、死区 (DB) 的典型工作模式

模式	模式描述	DBCTL			
		S3	S2	S1	S0
1	EPWMxA/B无延迟	×	×	0	0
2	高电平有效互补 (AHC)	1	0	1	1
3	低电平有效互补 (ALC)	0	1	1	1
4	高电平有效 (AH)	0	0	1	1
5	低电平有效 (AL)	1	1	1	1
6	EPWMxA无延迟	0或1	0或1	0	1
	EPWMxB下降沿延迟				
7	EPWMxA上升沿延迟	0或1	0或1	1	0
	EPWMxB无延迟				

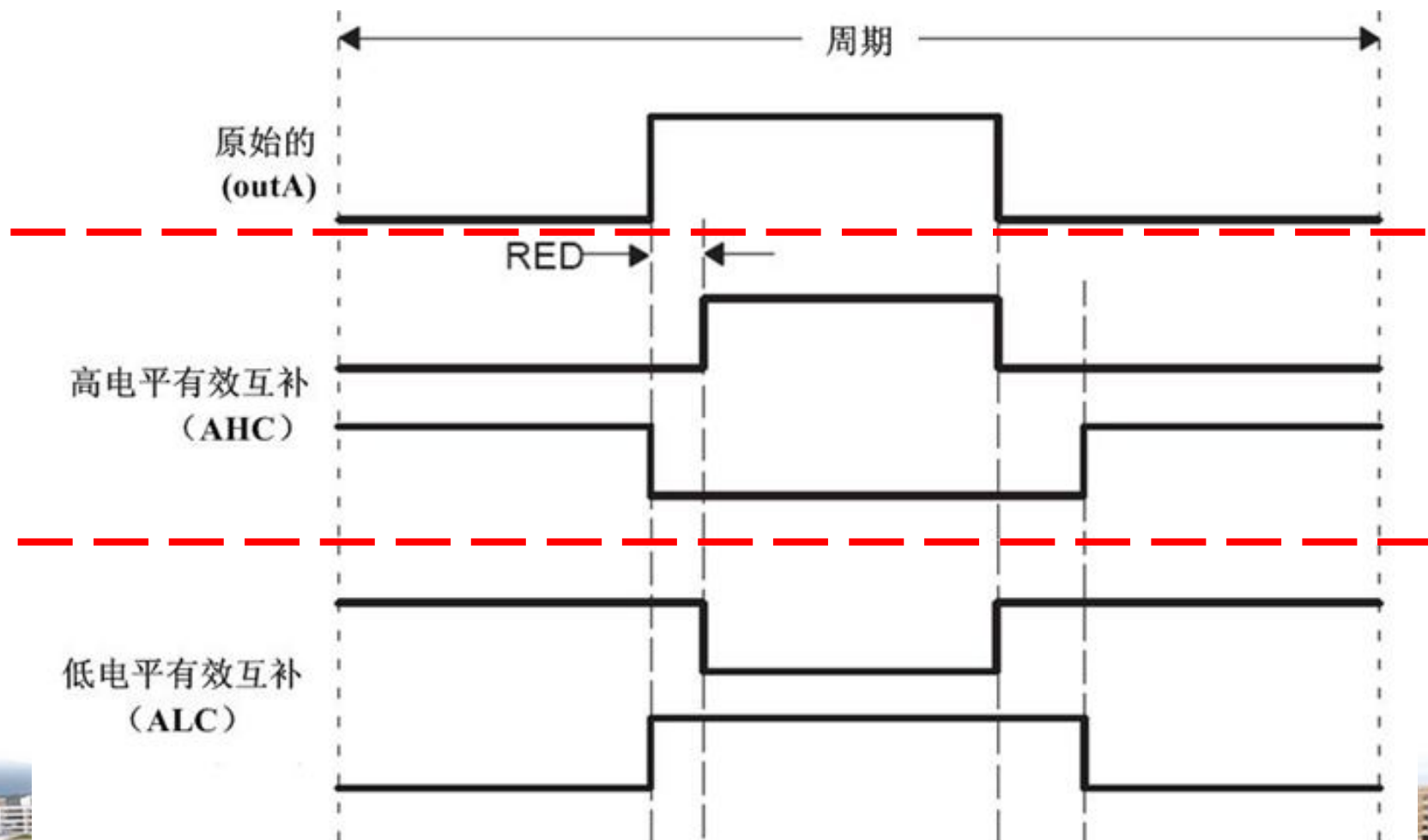




## Lecture7 ePWM

### 六、ePWM子模块功能和配置 -- 死区 (DB)

#### 4、死区 (DB) 的典型工作模式

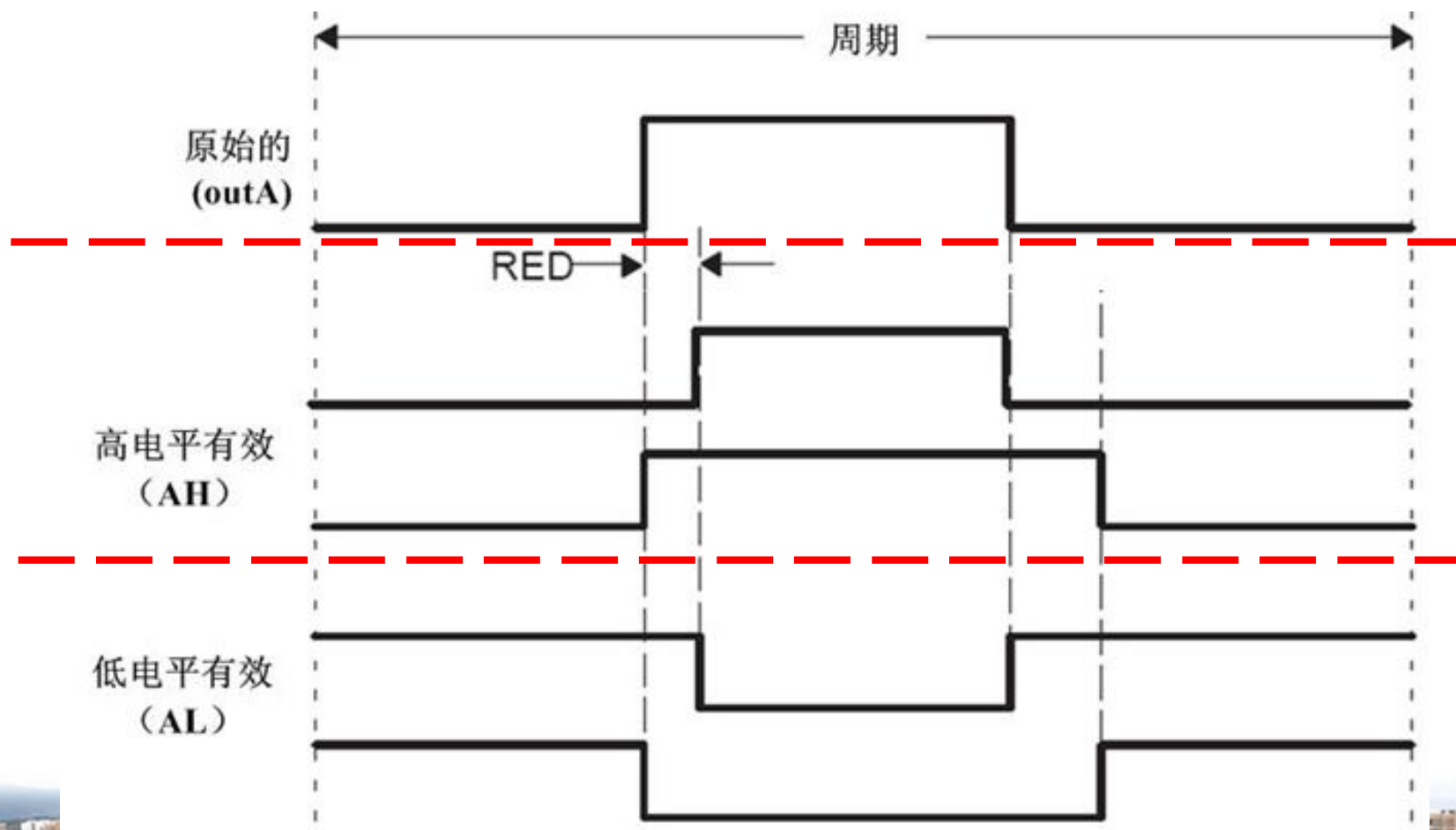




## Lecture7 ePWM

### 六、ePWM子模块功能和配置 -- 死区 (DB)

#### 4、死区 (DB) 的典型工作模式





## Lecture7 ePWM

### 六、ePWM子模块功能和配置 -- 死区 (DB)

#### 5、死区 (DB) 的延迟时间值

上升沿延迟时间:  $RED = DBRED \times T_{TBCLK}/k$

下降沿延迟时间:  $FED = DBFED \times T_{TBCLK}/k$

**DBRED、DBFED是10位寄存器: 0~1023**

使能半周期计时: **k=2**, 否则**k=1**。

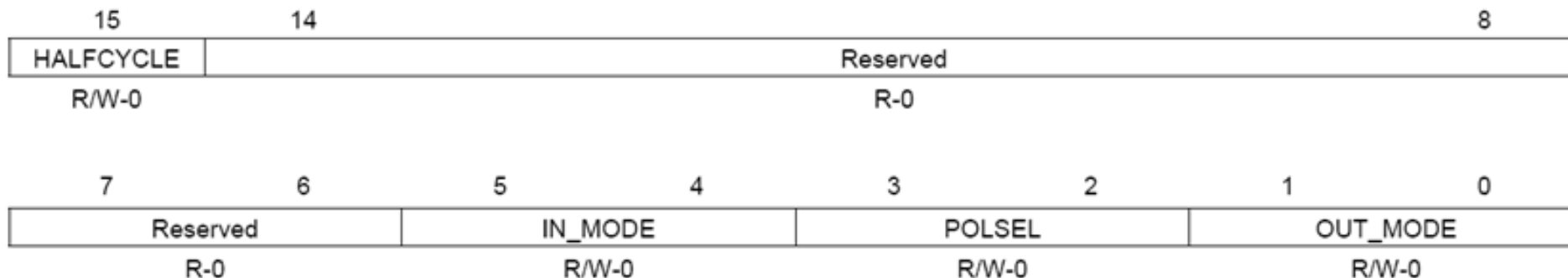




# Lecture7 ePWM

## 六、ePWM子模块功能和配置 -- 死区 (DB)

### 6、死区 (DB) 的寄存器 -- DBCTL



- **BIT15: 半周期计时使能位。** 0-半周期禁止，1-半周期使能。
- **BIT5-4: 输入模式控制位。** 位5控制S5，位4控制S4。

输入模式	上升沿延迟信号源	下降沿延迟信号源
00 (默认)	EPWMxA In	EPWMxA In
01	EPWMxA In	EPWMxB In
10	EPWMxB In	EPWMxA In
11	EPWMxB In	EPWMxB In

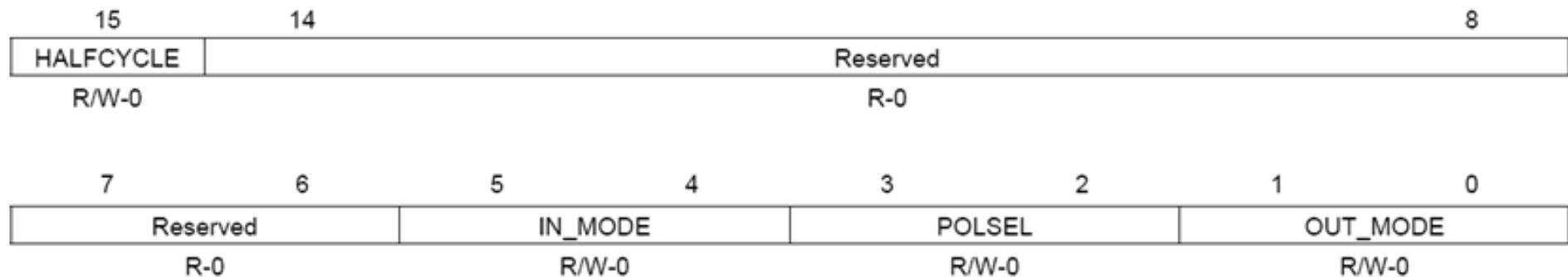




## Lecture7 ePWM

### 六、ePWM子模块功能和配置 -- 死区 (DB)

#### 6、死区 (DB) 的寄存器 -- DBCTL



➤ **BIT3-2: 输出极性选择位。** 位3控制**S3**，位2控制**S2**。

**00 - AH模式。EPWMxA/B都不反相（缺省）**

**01 - ALC模式。EPWMxA反相**

**10 - AHC模式。EPWMxB反相**

**11 - AL模式。EPWMxA/B都反相**



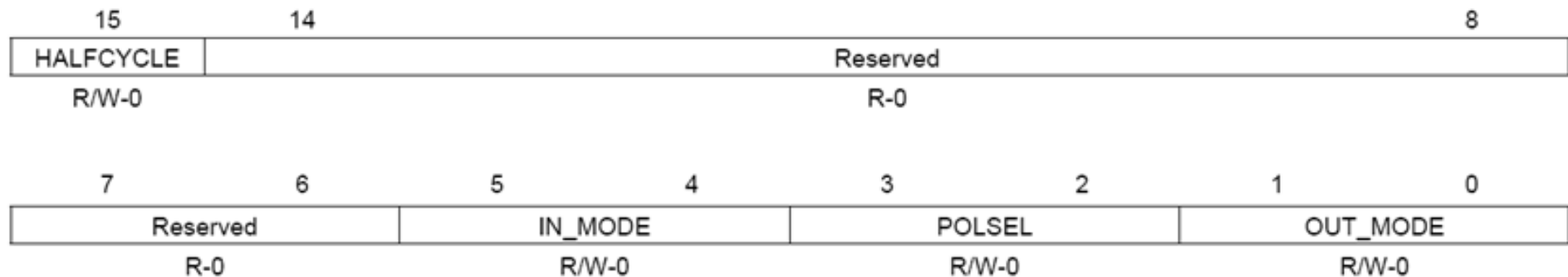




## Lecture7 ePWM

### 六、ePWM子模块功能和配置 -- 死区 (DB)

#### 6、死区 (DB) 的寄存器 -- DBCTL



➤ **BIT1-0: 输出模式选择位。** 位1控制**S1**，位0控制**S0**。

**00 - EPWMxA/B都旁路**

**01 - EPWMxA旁路， EPWMxB输出下降沿延迟**

**10 - EPWMxA输出上升沿延迟， EPWMxB旁路**

**11 - EPWMxA输出上升沿延迟， EPWMxB输出下降沿延迟**

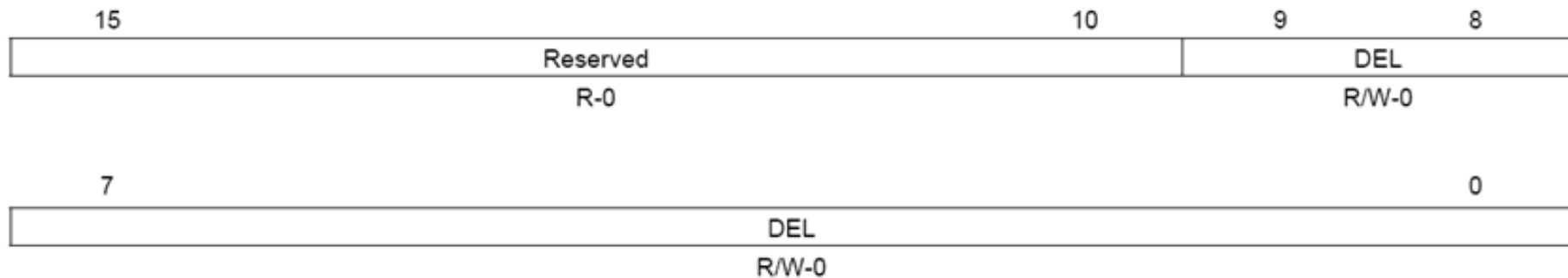




## Lecture7 ePWM

### 六、ePWM子模块功能和配置 -- 死区 (DB)

#### 6、死区 (DB) 的寄存器 -- DBRED



➤ **BIT9-0: 上升沿延迟计数值。DEL=0~1023**

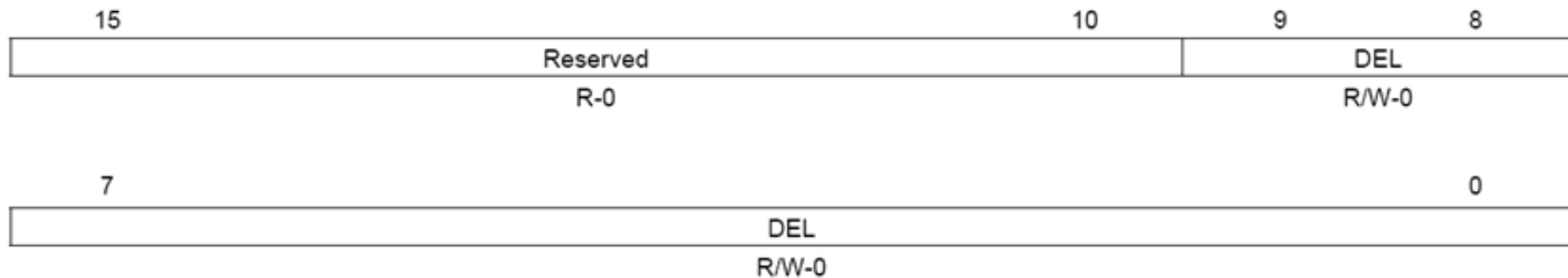




## Lecture7 ePWM

### 六、ePWM子模块功能和配置 -- 死区 (DB)

#### 6、死区 (DB) 的寄存器 -- DBFED



➤ **BIT9-0: 下降沿延迟计数值。0~1023**





## Lecture7 ePWM

### 七、ePWM子模块功能和配置 -- PWM斩波（PC）

#### 1、PWM斩波（PC）的功能

- 创建一个斩波（**载波**）频率
- 被斩脉冲串中第一个脉冲的**脉宽**
- 第一个以后的脉冲的**占空比**
- 完全**旁路**PWM斩波模块。这样，PWM波形将无需修改直接通过

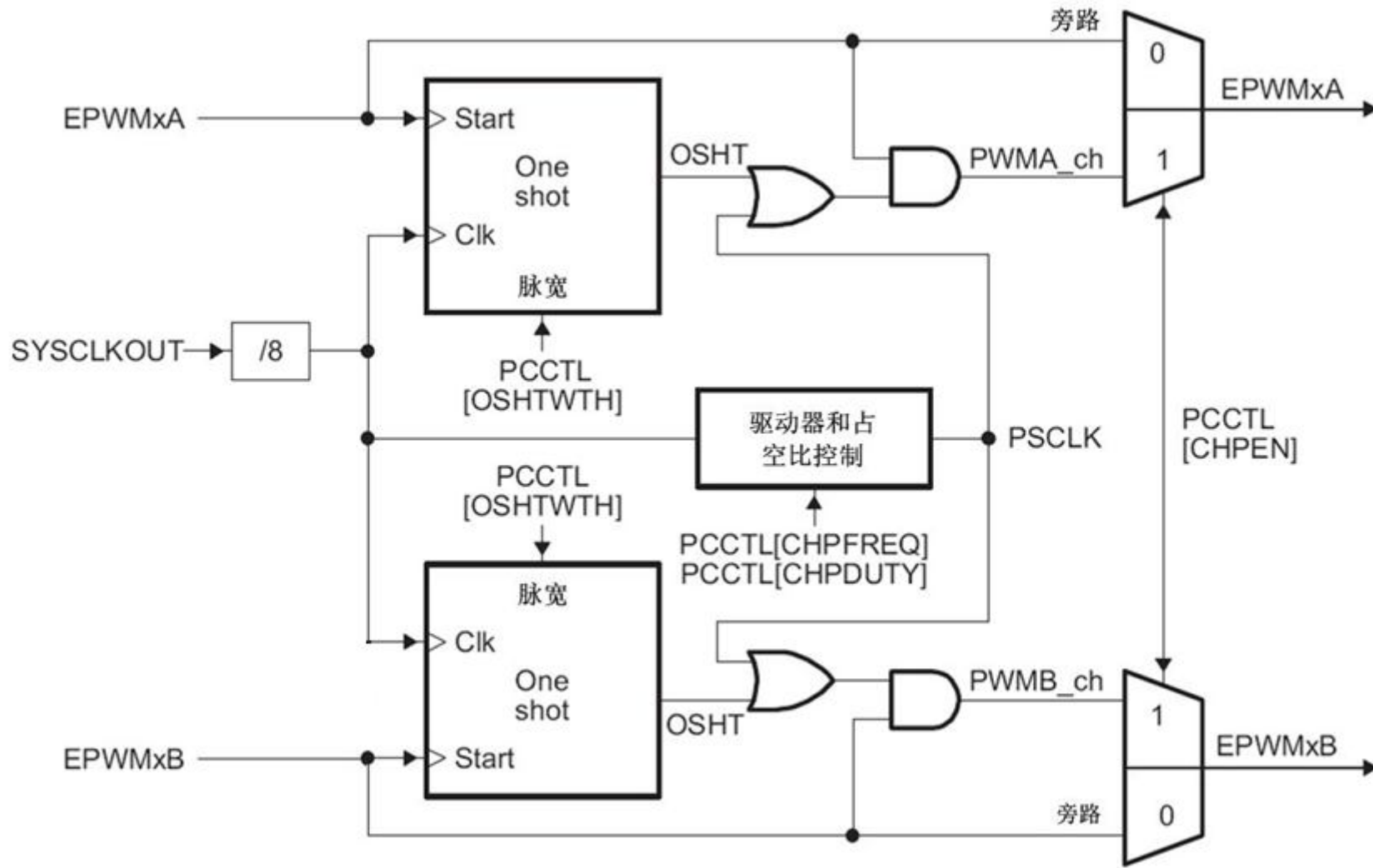




# Lecture7 ePWM

## 七、ePWM子模块功能和配置 -- PWM斩波 (PC)

### 2、PWM斩波 (PC) 的组成

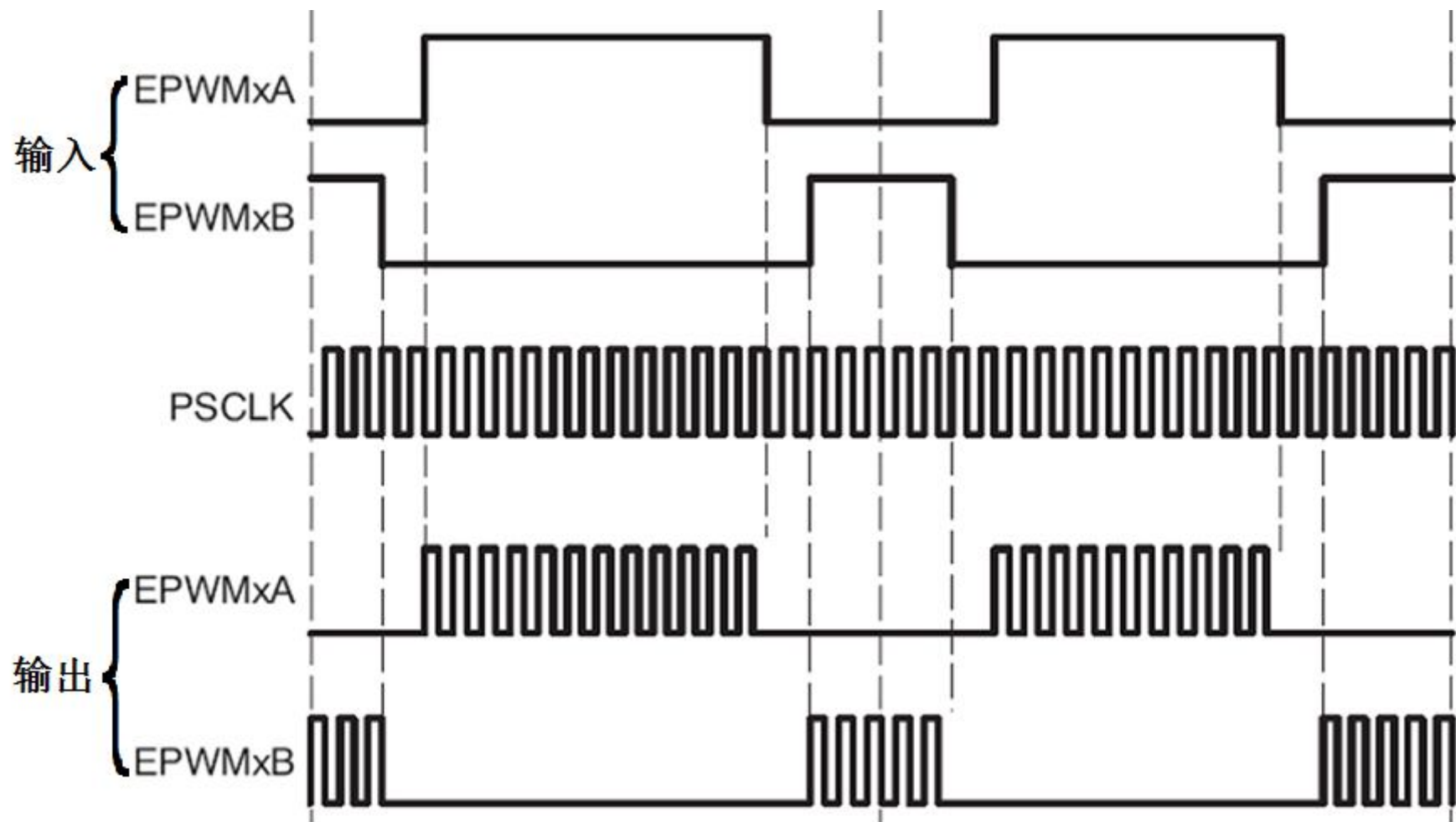




## Lecture7 ePWM

### 七、ePWM子模块功能和配置 -- PWM斩波（PC）

#### 3、PWM斩波（PC）的原理





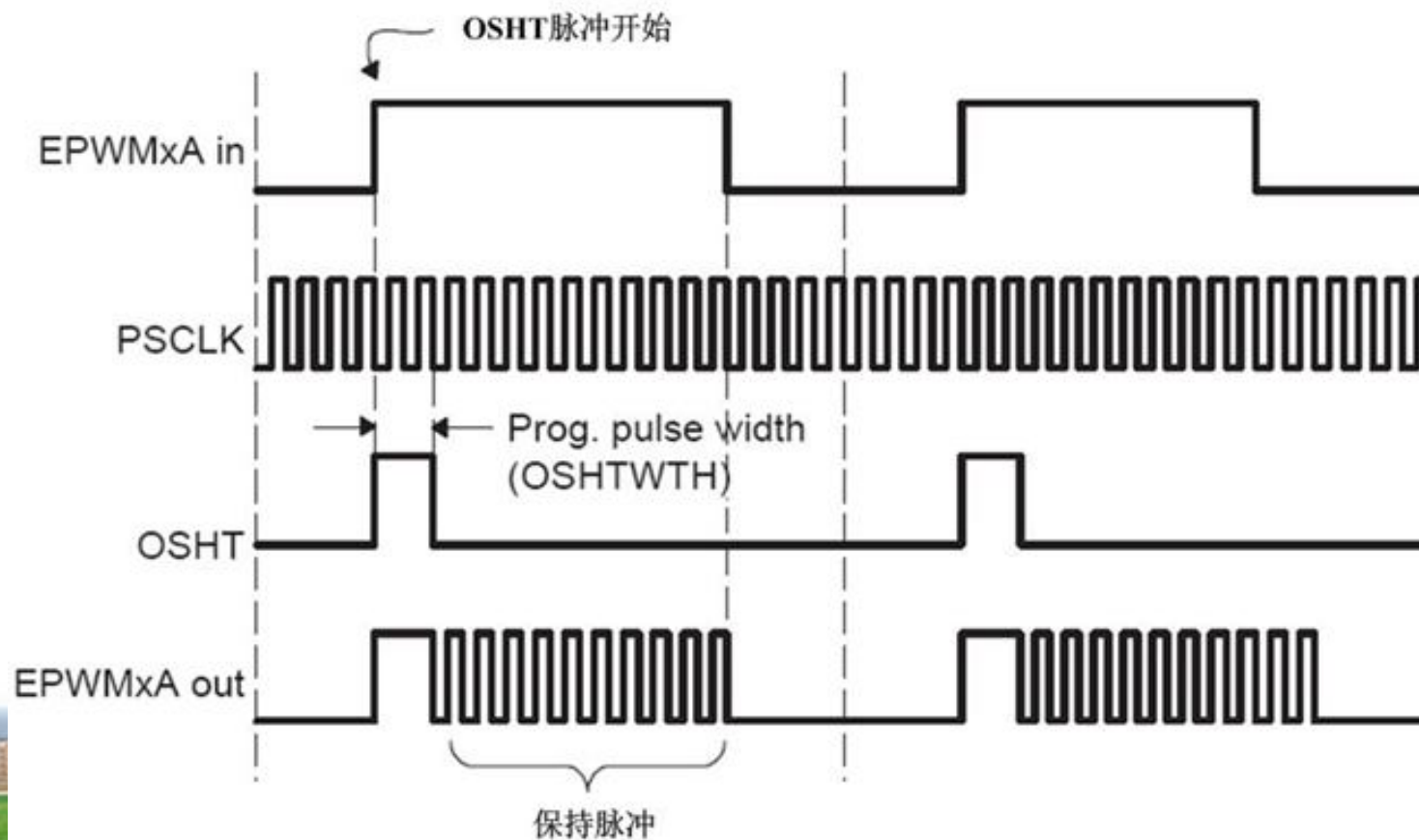


## Lecture7 ePWM

### 七、ePWM子模块功能和配置 -- PWM斩波 (PC)

#### 4、PWM斩波 (PC) 的one-shot脉冲

第一个脉冲的周期:  $T_{1stpulse} = T_{SYSCLKOUT} \times 8 \times OSHTWTH$





## Lecture7 ePWM

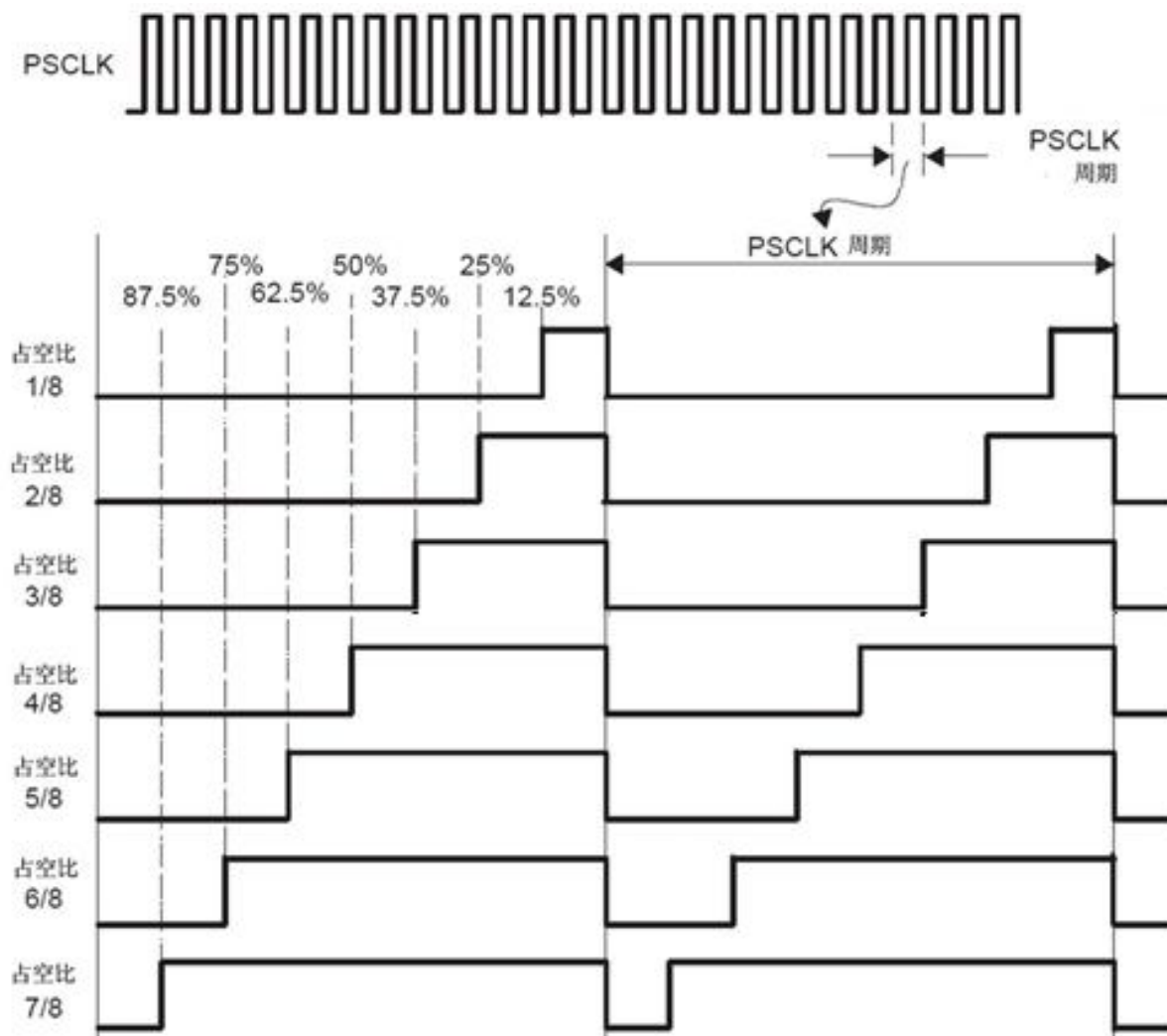
### 七、ePWM子模块功能和配置 -- PWM斩波 (PC)

#### 5、PWM斩波 (PC) 的占空比

通过CHPDUTY位

选择7个占空比

12.5%~87.5%

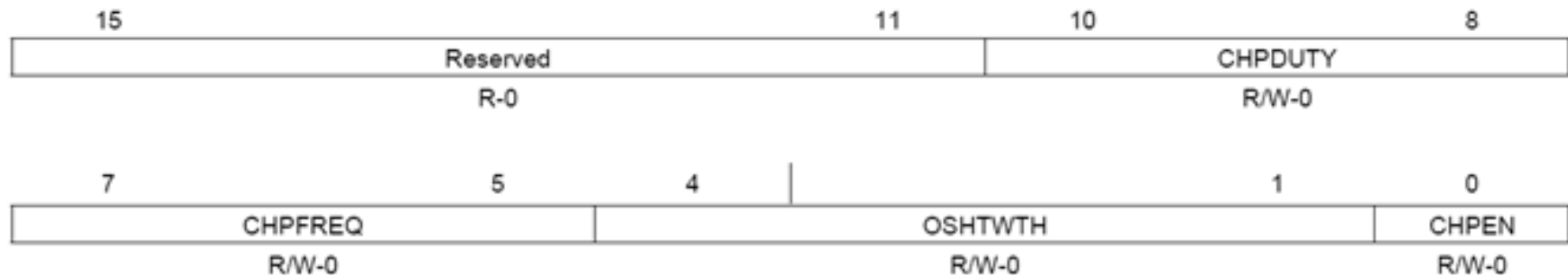




## Lecture7 ePWM

### 七、ePWM子模块功能和配置 -- PWM斩波 (PC)

### 6、PWM斩波 (PC) 的寄存器 -- PCCTL



➤ **BIT7-6: 斩波时钟频率 (CHPDUTY)。**

**000 - 占空比= 1/8 (12.5%)      001 - 占空比=2/8 (25.0%)**

**010 - 占空比=3/8 (37.5.0%)      011 - 占空比=4/8 (50.0%)**

**100 - 占空比=5/8 (62.5%)      101 - 占空比=6/8 (75.0%)**

**110 - 占空比=7/8 (87.5%)      111 - 保留**

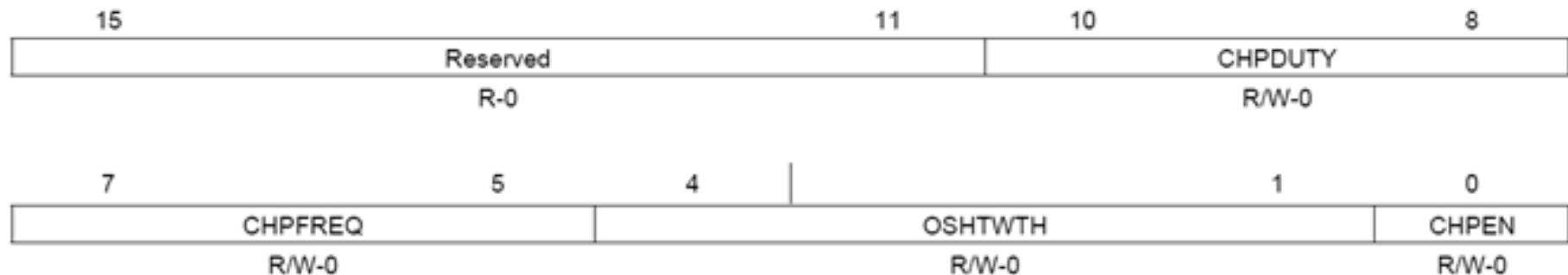




## Lecture7 ePWM

### 七、ePWM子模块功能和配置 -- PWM斩波 (PC)

### 6、PWM斩波 (PC) 的寄存器 -- PCCTL



➤ **BIT10-8: 斩波时钟频率 (CHPFREQ)。**

**000 – 不分频      001 – 2分频**

**010 – 3分频      011 – 4分频**

**100 – 5分频      101 – 6分频**

**110 – 7分频      111 – 8分频**

$$F_{PS} = \frac{SYSCLKOUT}{8 \times CHPFREQ}$$

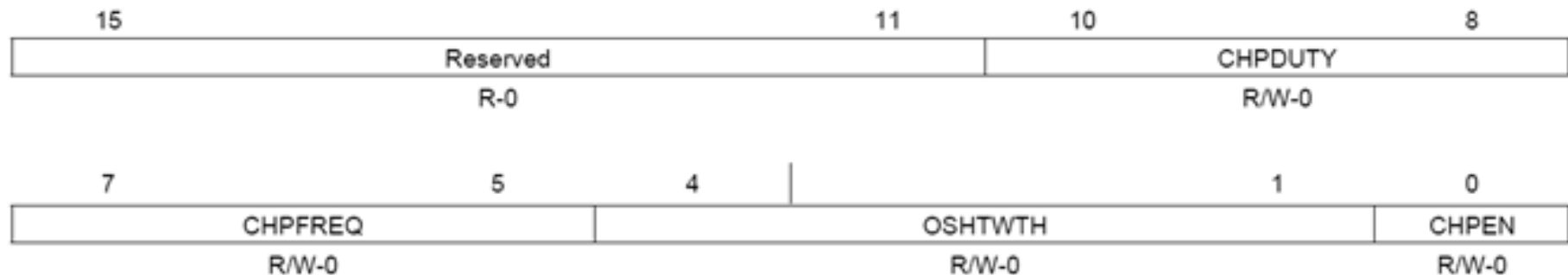




## Lecture7 ePWM

### 七、ePWM子模块功能和配置 -- PWM斩波 (PC)

### 6、PWM斩波 (PC) 的寄存器 -- PCCTL



- **BIT4-1: 第一个脉冲的宽度 (OSHTWTH)。**

第一个脉冲的周期:  $T1stpulse = T_{SYSCLKOUT} \times 8 \times OSHTWTH$

- **BIT0: PWM斩波使能位。**

**0** – 斩波禁止 (旁路)

**1** – 斩波使能





## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 1、触发区 (TZ) 的功能

- 输入**触发信号** ( $\overline{TZ1} - \overline{TZ6}$ ) 被灵活地映射到任意**ePWM**模块
- 设置故障时**ePWM**引脚**动作**: 置高、置低、高阻、不动作
- 配置**ePWM**对每个触发区信号作出**反应的频率**:
  - **One-shot** (单次) (如用于短路或过流保护)
  - **Cycle-by-cycle** (周期) (如用于限流保护)
- 让触发区发起一个**中断**
- 支持**软件强制**触发
- 完全**旁路**触发区模块



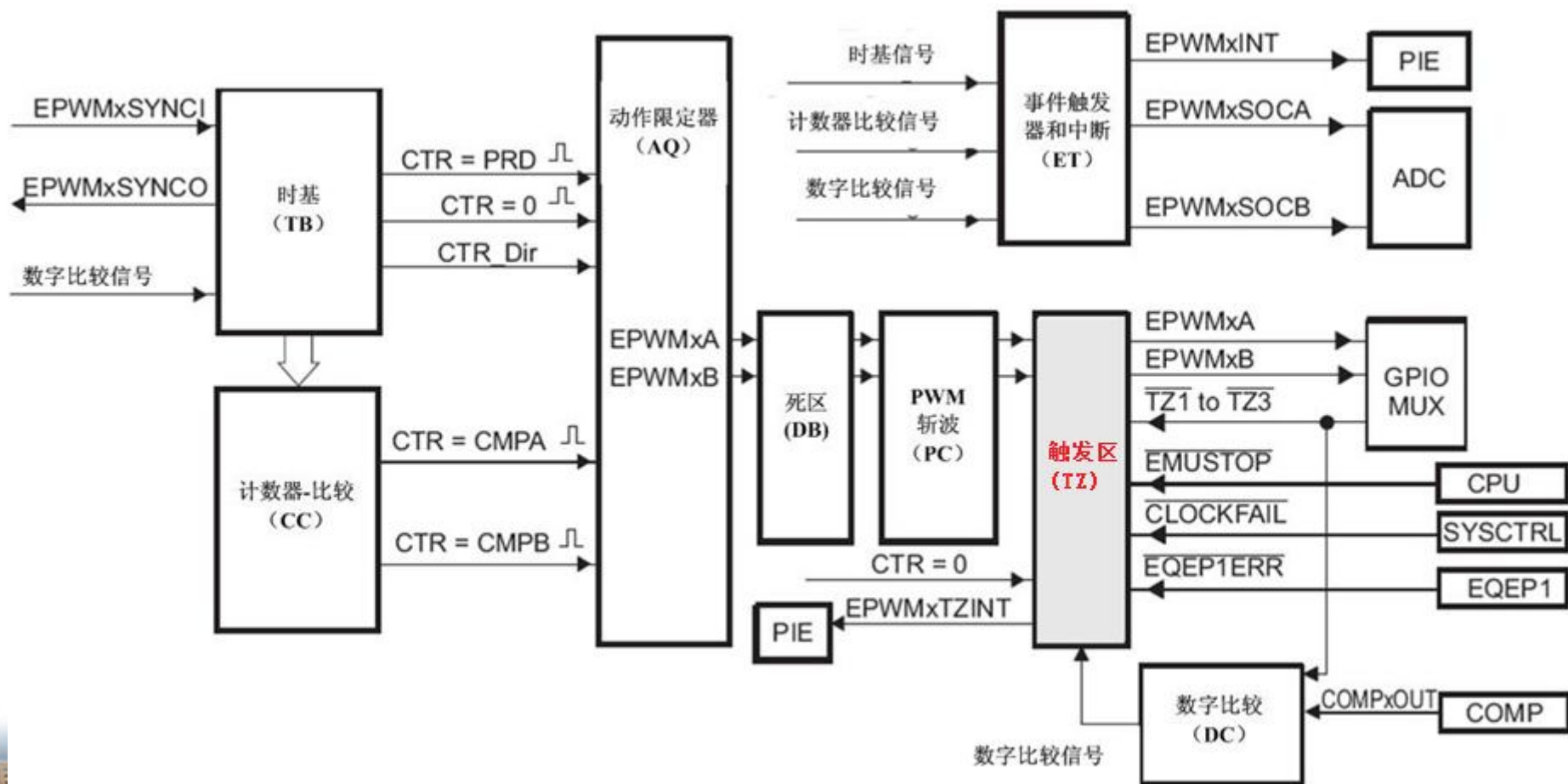




# Lecture7 ePWM

## 八、ePWM子模块功能和配置 -- 触发区 (TZ)

### 2、触发区 (TZ) 的逻辑

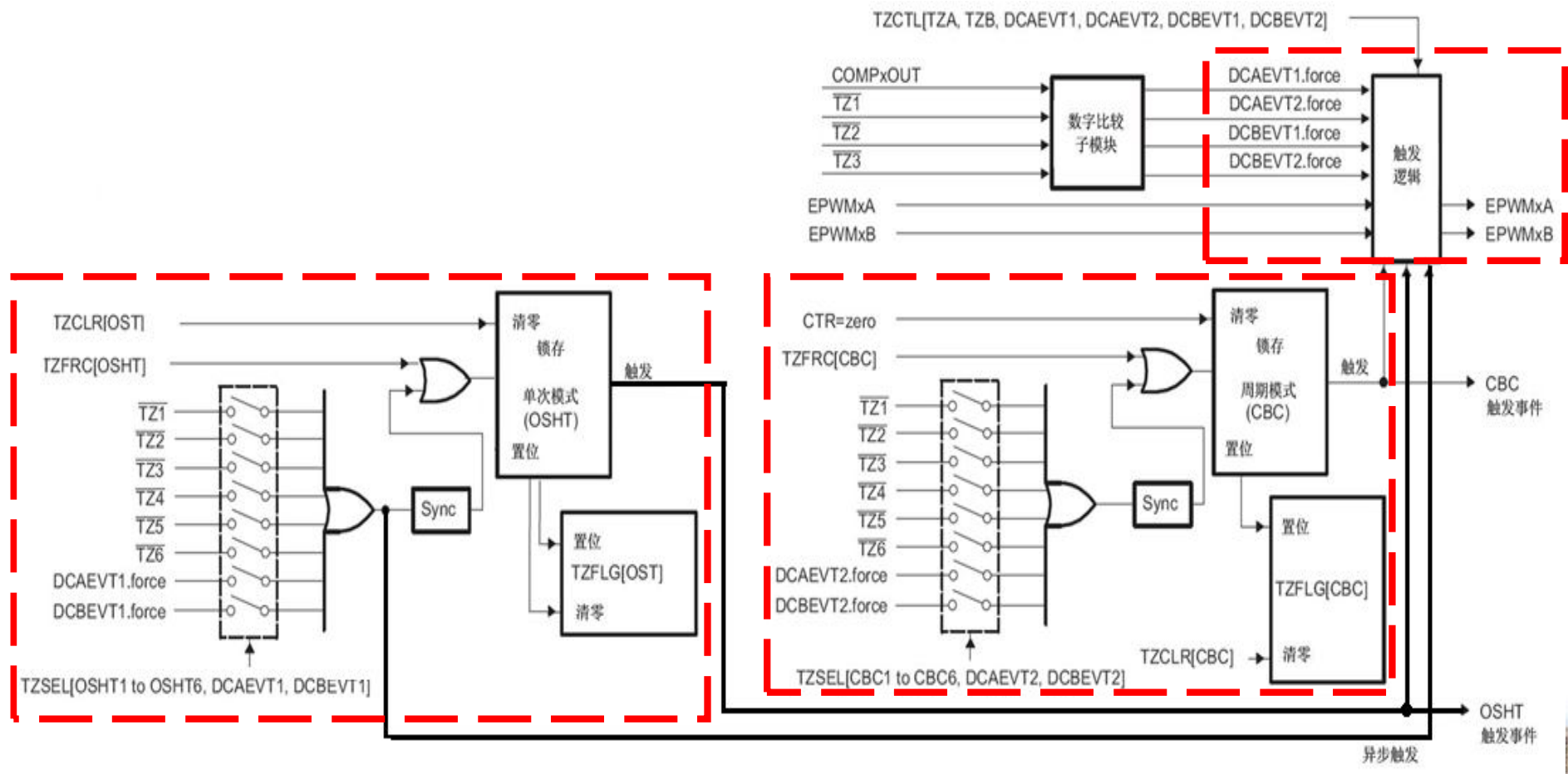




# Lecture7 ePWM

## 八、ePWM子模块功能和配置 -- 触发区 (TZ)

### 2、触发区 (TZ) 的逻辑





## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 3、触发区 (TZ) 的主要信号

##### 1)、 $\overline{TZ1} - \overline{TZ6}$

- 低电平有效的输入信号。其中一个变低，表示发生了错误触发事件 (**trip event**)。
- $\overline{TZ1} - \overline{TZ3}$ 由**GPIO**复用引脚外部输入。 $\overline{TZ4}$ 由那些含**EQEP1**模块的器件上的**EQEP1ERR**反相信号提供。 $\overline{TZ5}$ 与系统时钟失效 (**clock fail**) 逻辑电路连接， $\overline{TZ6}$ 由**CPU**的**EMUSTOP**输出提供。





## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 3、触发区 (TZ) 的主要信号

##### 1)、 $\overline{TZ1} - \overline{TZ6}$

- 输入最短脉宽为“**3\*TBCLK**”的低电平脉冲。
- $\overline{TZ1} - \overline{TZ3}$ 可以在**GPIO MUX**内被数字滤波。
- 异步触发保证，输入有效事件在时钟失效时仍可触发。





## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 3、触发区 (TZ) 的主要信号

##### 2) 数字比较A/B事件

输入: **DCAEVT1、DCAEVT2、DCBEVT1、DCBEVT2**

##### 3) PWMxA、PWMxB

输入: 来自**PWM斩波器 (PC)**

输出: 送到**GPIO**复用引脚输出波形







## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 4、触发区 (TZ) 的触发模式

##### 1)、周期性触发 (CBC) :

- 当发生周期性触发事件时，**EPWMxA/B**按设置动作。
- 周期性触发事件标志置位 (**TZFLG.CBC=1**) 。
- 若使能，则产生**EPWMx\_TZINT**中断信号。
- **CTR = 0**时，若触发事件不再存在，那么**EPWMxA/B**的故障触发条件自动清零。
- **TZFLG.CBC**必须手动清零。







## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 4、触发区 (TZ) 的触发模式

##### 2)、单次触发 (OSHT) :

- 发生单次触发事件时，， **EPWMxA/B**按设置动作。
- 单次触发事件标志置位 (**TZFLG.OST=1**)
- 若使能，则产生**EPWMx\_TZINT**中断信号。
- **EPWMxA/B**上单次故障触发条件必须通过手动清除 (写**TZCLR.OST**位)，否则一直保持。





## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 5、触发区 (TZ) 的优先级

**TZCTL**寄存器上其冲突动作的优先级如下:

输出**EPWMxA**:

**TZA (最高) → DCAEVT1 → DCAEVT2 (最低)**

输出**EPWMxB**:

**TZB (最高) → DCBEVT1 → DCBEVT2 (最低)**

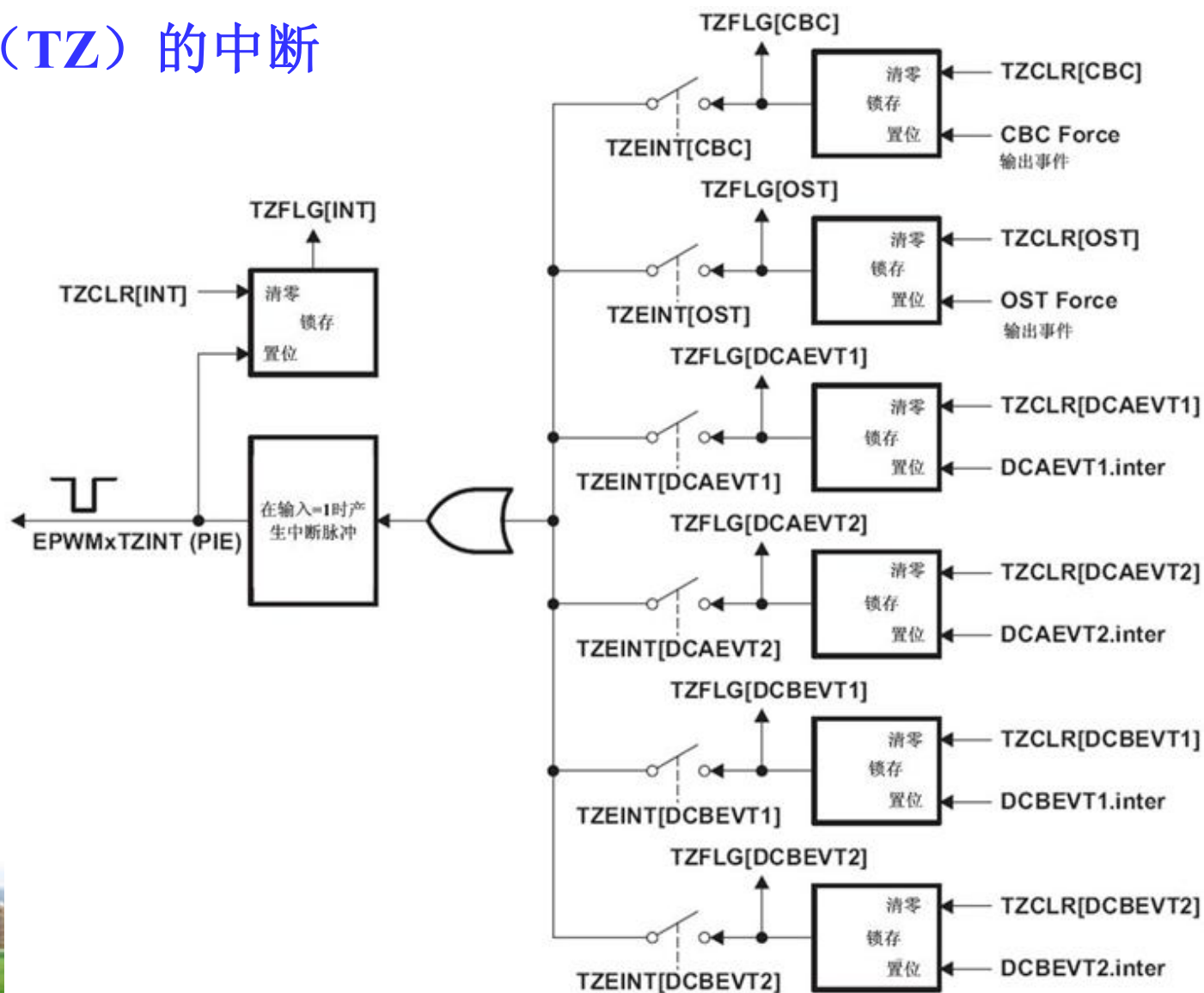




# Lecture7 ePWM

## 八、ePWM子模块功能和配置 -- 触发区 (TZ)

### 5、触发区 (TZ) 的中断







## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 6、触发区 (TZ) 的寄存器 -- TZSEL

15	14	13	12	11	10	9	8
DCBEVT1	DCAEVT1	OSHT6	OSHT5	OSHT4	OSHT3	OSHT2	OSHT1
R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
7	6	5	4	3	2	1	0
DCBEVT2	DCAEVT2	CBC6	CBC5	CBC4	CBC3	CBC2	CBC1
R-0		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

➤ **BIT15: 数字比较输出B事件1单次触发选择位。**

**0 – 禁止将DCBEVT1作为ePWM模块的单次触发源**

**1 – 将DCBEVT1作为ePWM模块的单次触发源**

➤ **BIT14: 数字比较输出A事件1单次触发选择位。**

**0 – 禁止将DCAEVT1作为ePWM模块的单次触发源**

**1 – 将DCAEVT1作为ePWM模块的单次触发源**



## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 6、触发区 (TZ) 的寄存器 -- TZSEL

15	14	13	12	11	10	9	8
DCBEVT1	DCAEVT1	OSHT6	OSHT5	OSHT4	OSHT3	OSHT2	OSHT1
R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
7	6	5	4	3	2	1	0
DCBEVT2	DCAEVT2	CBC6	CBC5	CBC4	CBC3	CBC2	CBC1
R-0		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

➤ **BIT13-8: 触发器 (TZ6-TZ1) 单次触发选择位。**

**0** – 禁止将触发器x作为ePWM模块的单次触发源

**1** – 将触发器x作为ePWM模块的单次触发源







## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 6、触发区 (TZ) 的寄存器 -- TZSEL

15	14	13	12	11	10	9	8
DCBEVT1	DCAEVT1	OSHT6	OSHT5	OSHT4	OSHT3	OSHT2	OSHT1
R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
7	6	5	4	3	2	1	0
DCBEVT2	DCAEVT2	CBC6	CBC5	CBC4	CBC3	CBC2	CBC1
R-0		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

➤ **BIT7: 数字比较输出B事件2周期性触发选择位。**

**0 – 禁止将DCBEVT2作为ePWM模块的周期性触发源**

**1 – 将DCBEVT2作为ePWM模块的周期性触发源**

➤ **BIT6: 数字比较输出A事件2周期性触发选择位。**

**0 – 禁止将DCAEVT2作为ePWM模块的周期性触发源**

**1 – 将DCAEVT2作为ePWM模块的周期性触发源**



# Lecture7 ePWM

## 八、ePWM子模块功能和配置 -- 触发区 (TZ)

### 6、触发区 (TZ) 的寄存器 -- TZSEL

15	14	13	12	11	10	9	8
DCBEVT1	DCAEVT1	OSHT6	OSHT5	OSHT4	OSHT3	OSHT2	OSHT1
R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
7	6	5	4	3	2	1	0
DCBEVT2	DCAEVT2	CBC6	CBC5	CBC4	CBC3	CBC2	CBC1
R-0		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

➤ **BIT5-0: 触发器 (TZ6-TZ1) 周期性触发选择位。**

**0** – 禁止将触发器x作为ePWM模块的周期性触发源

**1** – 将触发器x作为ePWM模块的周期性触发源





## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 6、触发区 (TZ) 的寄存器 -- TZCTL

15	12	11	10	9	8		
Reserved				DCBEVT2	DCBEVT1		
R-0				R/W-0	R/W-0		
7	6	5	4	3	2	1	0
DCAEVT2		DCAEVT1		TZB		TZA	
R/W-0		R/W-0		R/W-0		R/W-0	

➤ **BIT11-10:** 数字比较输出B事件2在EPWMxB上的动作。

00 – 高阻                      01 – 强制高

10 – 强制低                  11 – 不动作

➤ **BIT9-8:** 数字比较输出B事件1在EPWMxB上的动作。

00 – 高阻                      01 – 强制高

10 – 强制低                  11 – 不动作



## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 6、触发区 (TZ) 的寄存器 -- TZCTL

15	12	11	10	9	8		
Reserved				DCBEVT2	DCBEVT1		
R-0				R/W-0	R/W-0		
7	6	5	4	3	2	1	0
DCAEVT2		DCAEVT1		TZB		TZA	
R/W-0		R/W-0		R/W-0		R/W-0	

➤ **BIT11-10:** 数字比较输出A事件2在EPWMxA上的动作。

00 – 高阻                      01 – 强制高

10 – 强制低                  11 – 不动作

➤ **BIT9-8:** 数字比较输出A事件1在EPWMxA上的动作。

00 – 高阻                      01 – 强制高

10 – 强制低                  11 – 不动作



## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 6、触发区 (TZ) 的寄存器 -- TZCTL

15	12	11	10	9	8		
Reserved				DCBEVT2	DCBEVT1		
R-0				R/W-0	R/W-0		
7	6	5	4	3	2	1	0
DCAEVT2		DCAEVT1		TZB		TZA	
R/W-0		R/W-0		R/W-0		R/W-0	

➤ **BIT3-2: 触发器 (TZ6-TZ1) 在EPWMxB上的动作。**

**00 – 高阻                      01 – 强制高**

**10 – 强制低                  11 – 不动作**

➤ **BIT1-0: 触发器 (TZ6-TZ1) 在EPWMA上的动作。**

**00 – 高阻                      01 – 强制高**

**10 – 强制低                  11 – 不动作**



## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 6、触发区 (TZ) 的寄存器 -- TZEINT



- **BIT6: DCBEVT2**中断使能位。      0 –禁止      1-使能。
- **BIT5: DCBEVT1**中断使能位。      0 –禁止      1-使能。
- **BIT4: DCAEVT2**中断使能位。      0 –禁止      1-使能。
- **BIT3: DCAEVT1**中断使能位。      0 –禁止      1-使能。
- **BIT2: OST**中断使能位。      0 –禁止      1-使能。
- **BIT1: CBC**中断使能位。      0 –禁止      1-使能。

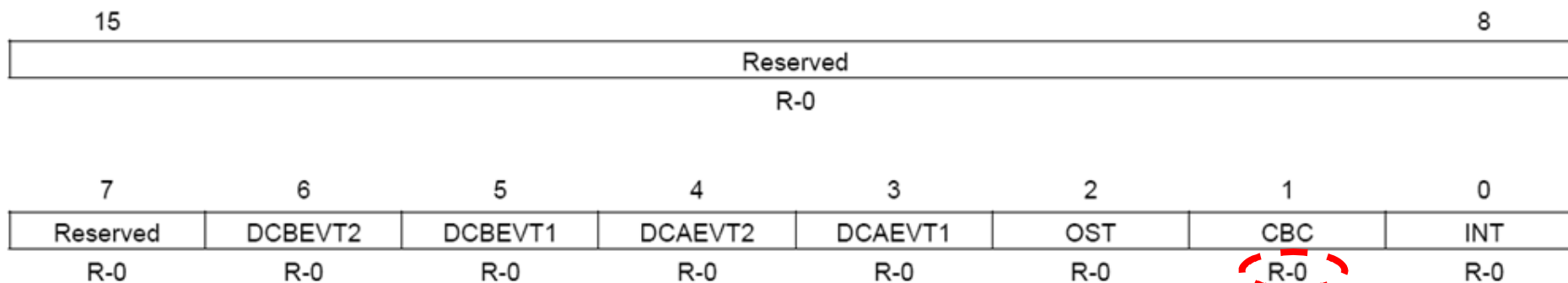




## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 6、触发区 (TZ) 的寄存器 -- TZFLG



- **BIT6: DCBEVT2锁存状态标志位。** 0-无触发; 1-有触发。  
均为只读位。
- **BIT5: DCBEVT1锁存状态标志位。** 0-无触发; 1-有触发。
- **BIT4: DCAEVT2锁存状态标志位。** 0-无触发; 1-有触发。
- **BIT3: DCAEVT1锁存状态标志位。** 0-无触发; 1-有触发。

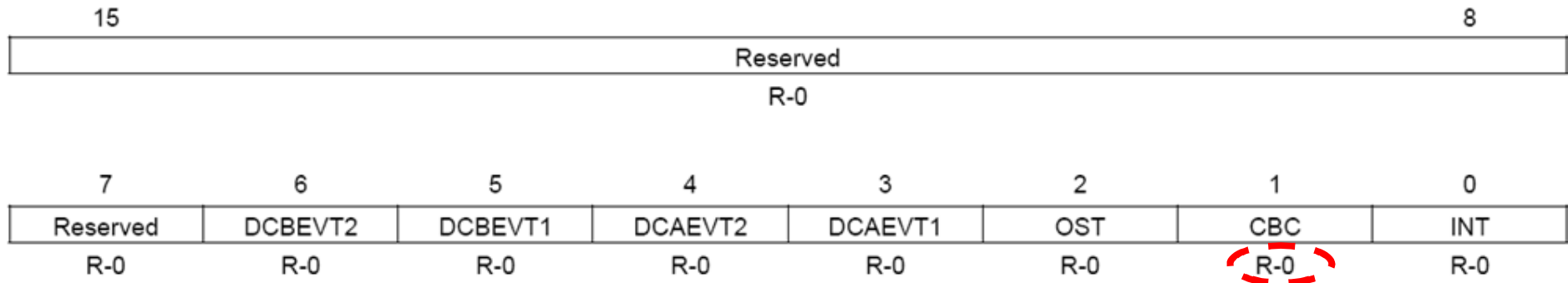




## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 6、触发区 (TZ) 的寄存器 -- TZFLG



➤ **BIT2: OST锁存状态标志位。** 0-无触发; 1-有触发。

只读位, 通过TZCLR寄存器可以清零。

➤ **BIT1: CBC锁存状态标志位。** 0-无触发; 1-有触发。

➤ **BIT0: 触发中断状态标志位。** 0-无中断; 1-有中断。

该位可通过向TZCLR寄存器写入合适值来清零。





## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 6、触发区 (TZ) 的寄存器 -- TZCLR



➤ **BIT6: DCBEVT2清零标志位。**

读总为0；写0 -无效；写1-清除DCBEVT2事件故障触发条件。

以下3位功能相同。

➤ **BIT5: DCBEVT1清零标志位。** 0 -无触发； 1-有触发。

➤ **BIT4: DCAEVT2清零标志位。** 0 -无触发； 1-有触发。

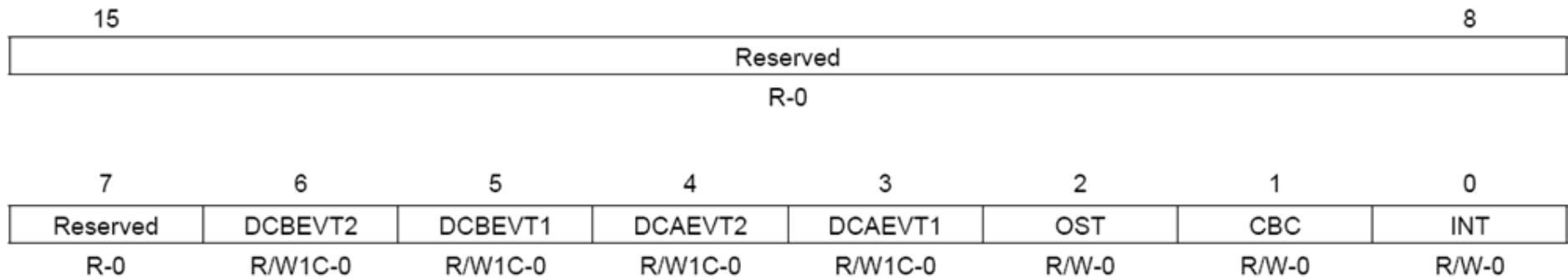
➤ **BIT3: DCAEVT1清零标志位。** 0 -无触发； 1-有触发。



## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 6、触发区 (TZ) 的寄存器 -- TZCLR



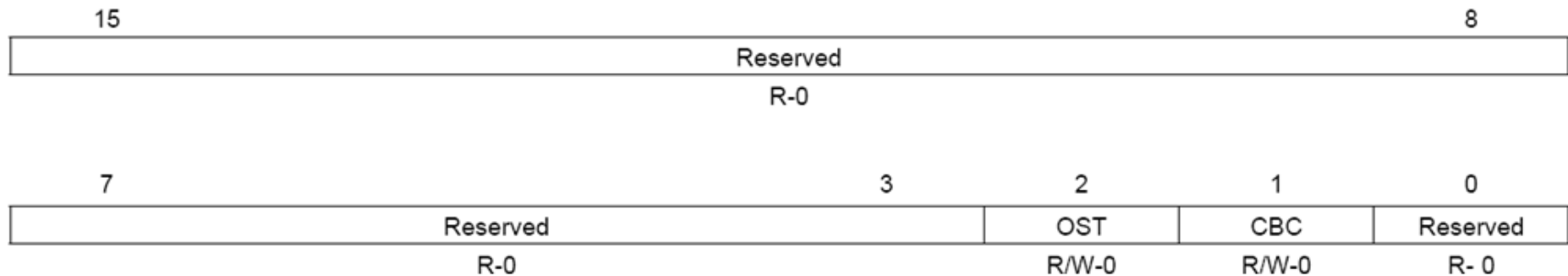
- **BIT2: OST锁存清零标志位。** 读总为0；  
写0 –无效；写1-清除OST事件故障触发条件。
- **BIT1: CBC锁存清零标志位。** 读总为0；  
写0 –无效；写1-清除OST事件故障触发条件。
- **BIT0: 触发中断清零标志位。** 读总为0；  
写0 –无效；写1-清除触发中断标志。



## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 6、触发区 (TZ) 的寄存器 -- TZFRC



➤ **BIT2: OST软件强制位**。读总为0;

写0 -无效; 写1-强制发生OST事件并TZFLG.OST=1。

➤ **BIT1: CBC软件强制位**。读总为0;

写0 -无效; 写1-强制发生CBC事件并TZFLG.CBC=1。

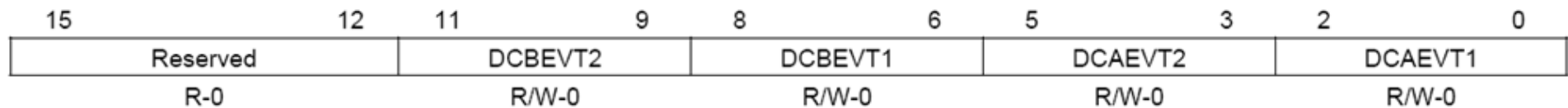




## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 6、触发区 (TZ) 的寄存器 -- TZDCSEL



➤ **数字比较事件输出选择位。** 各个位域值代表如下：

**000** – 禁用

**001** - DCBH =低电平， DCBL = 任意值

**010** - DCBH =高电平， DCBL = 任意值

**011** - DCBL =低电平， DCBH = 任意值

**100** - DCBL =高电平， DCBH = 任意值

**101** - DCBL =高电平， DCBH = 低电平

**11x** - 保留





## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 7、触发区 (TZ) 的实例

**实例一：**  $\overline{TZ1}$ 上的一个单次触发事件要求将EPWM1A、EPWM1B置低电平，将EPWM2A、EPWM2B置高电平。

•ePWM1寄存器配置如下：

**TZSEL[OSHT1] = 1：** 将 $\overline{TZ1}$ 作为ePWM1的一个单次触发事件源

**TZCTL[TZA] = 2：** 在发生触发事件时EPWM1A将被强制变低

**TZCTL[TZB] = 2：** 在发生触发事件时EPWM1B将被强制变低

•ePWM2寄存器配置如下：

**TZSEL[OSHT1] = 1：** 将 $\overline{TZ1}$ 作为ePWM2的一个单次触发事件源

**TZCTL[TZA] = 1：** 在发生触发事件时EPWM2A将被强制变高

**TZCTL[TZB] = 1：** 在发生触发事件时EPWM2B将被强制变高



## Lecture7 ePWM

### 八、ePWM子模块功能和配置 -- 触发区 (TZ)

#### 7、触发区 (TZ) 的实例

**实例二：**  $\overline{TZ5}$ 上的一个周期性触发事件会将EPWM1A、EPWM1B拉低。

$\overline{TZ1}$ 或 $\overline{TZ6}$ 上的一个单次触发事件会将EPWM2A变为高阻状态。

•ePWM1寄存器配置如下：

**TZSEL[CBC5] = 1:** 将 $\overline{TZ5}$ 作为ePWM1的一个周期性触发事件源

**TZCTL[TZA] = 2:** 在发生触发事件时EPWM1A将被强制变低

**TZCTL[TZB] = 2:** 在发生触发事件时EPWM1B将被强制变低

•ePWM2寄存器配置如下：

**TZSEL[OSHT1] = 1:** 将 $\overline{TZ1}$ 作为ePWM2的一个单次触发事件源

**TZSEL[OSHT6] = 1:** 将 $\overline{TZ6}$ 作为ePWM1的一个单次触发事件源

**TZCTL[TZA] = 0:** 在发生触发事件时EPWM2A将变为高阻状态

**TZCTL[TZB] = 3:** EPWM2B将忽略触发事件



## Lecture7 ePWM

### 九、ePWM子模块功能和配置 -- 事件触发器 (ET)

#### 1、事件触发器 (ET) 的功能

- 接收BT、CC和DC子模块产生的事件输入
- 使用时基方向限定递增 (up) /递减 (down) 事件
- 经预分频逻辑电路发出中断请求和ADC开始转换
  - 每个事件
  - 每两个事件 (每隔一个事件)
  - 每三个事件 (每隔两个事件)
- 通过事件计数器和标志将事件的产生过程变成完全可视
- 允许软件强制进行中断和ADC开始转换

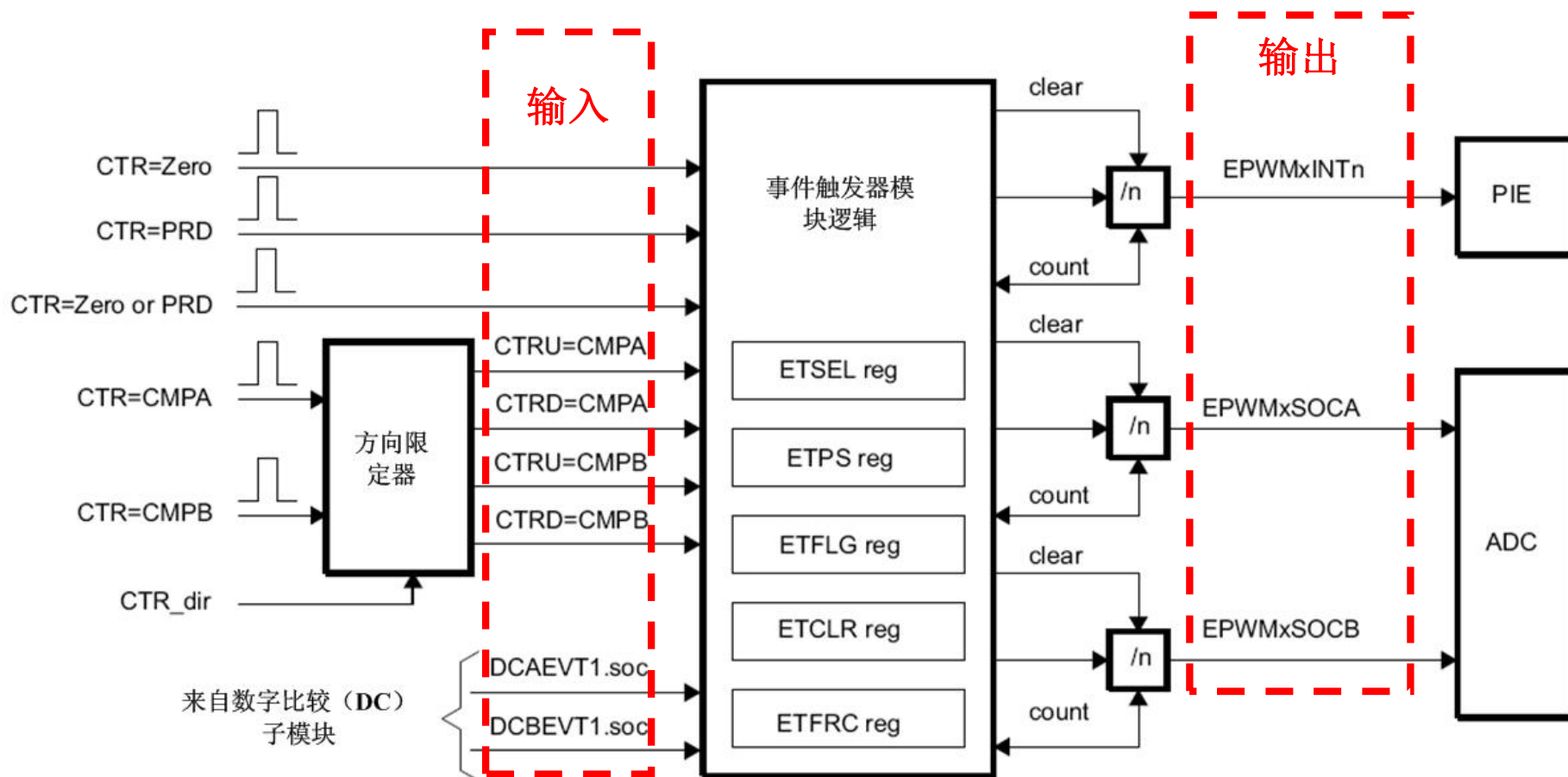




# Lecture7 ePWM

## 九、ePWM子模块功能和配置 -- 事件触发器 (ET)

### 2、事件触发器 (ET) 的组成 -- 框图



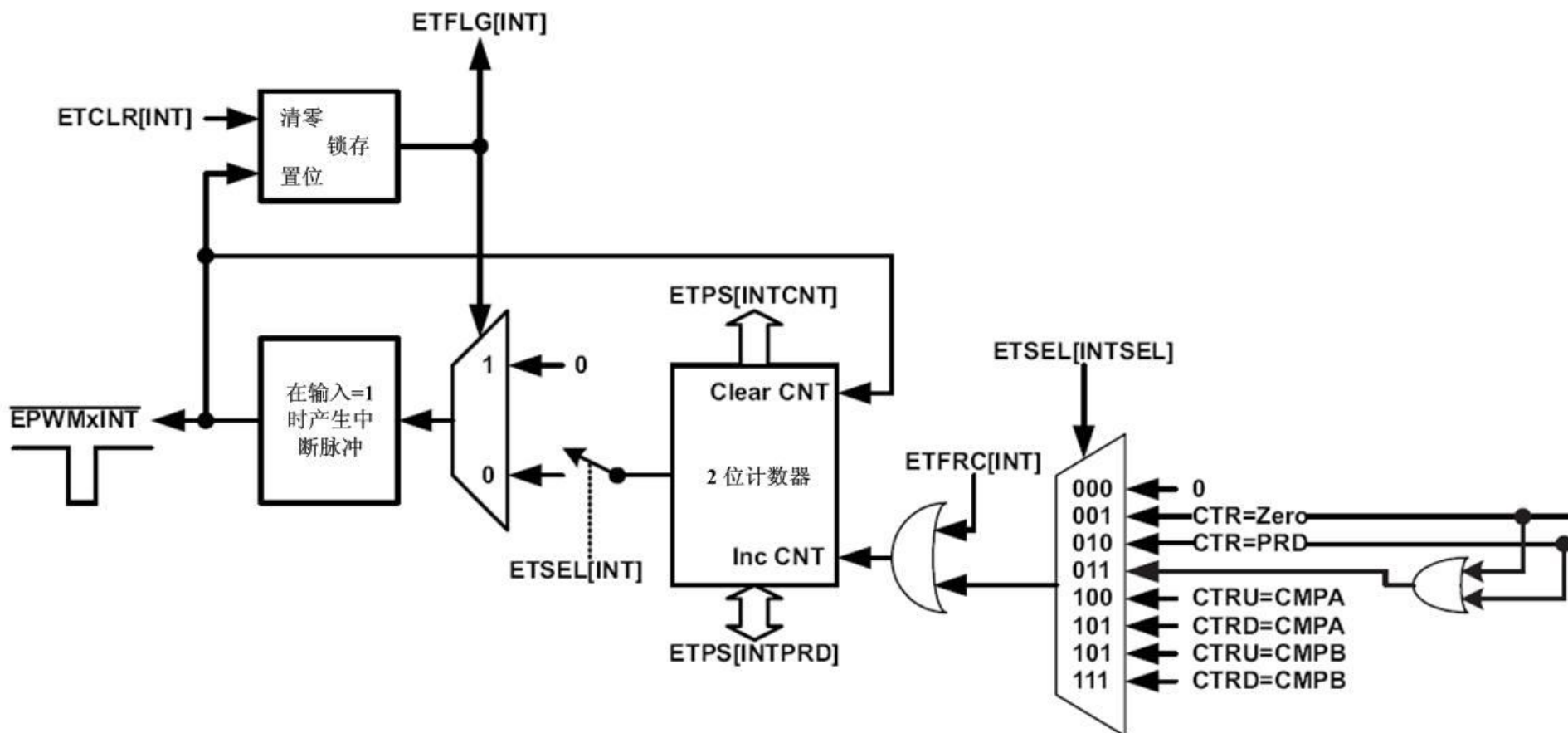




# Lecture7 ePWM

## 九、ePWM子模块功能和配置 -- 事件触发器 (ET)

### 2、事件触发器 (ET) 的组成 -- 中断发生器

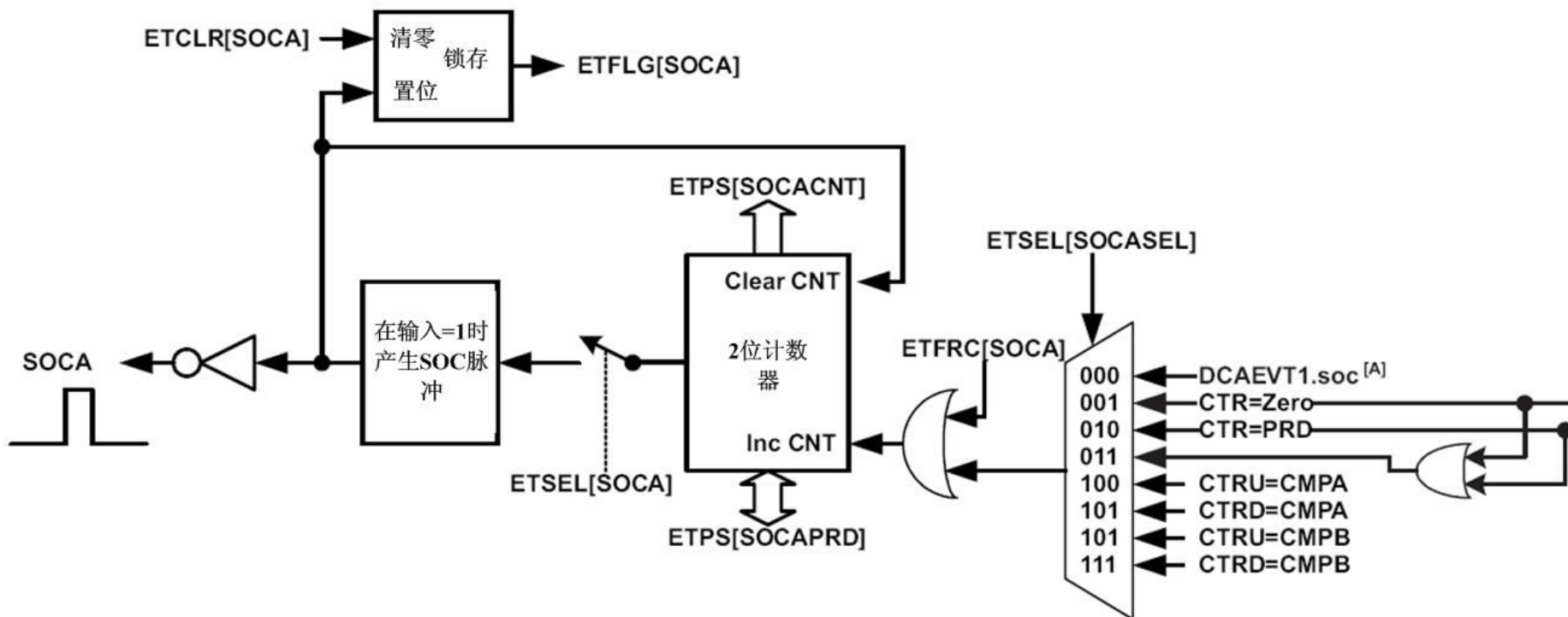




# Lecture7 ePWM

## 九、ePWM子模块功能和配置 -- 事件触发器 (ET)

### 2、事件触发器 (ET) 的组成 -- SOCA发生器



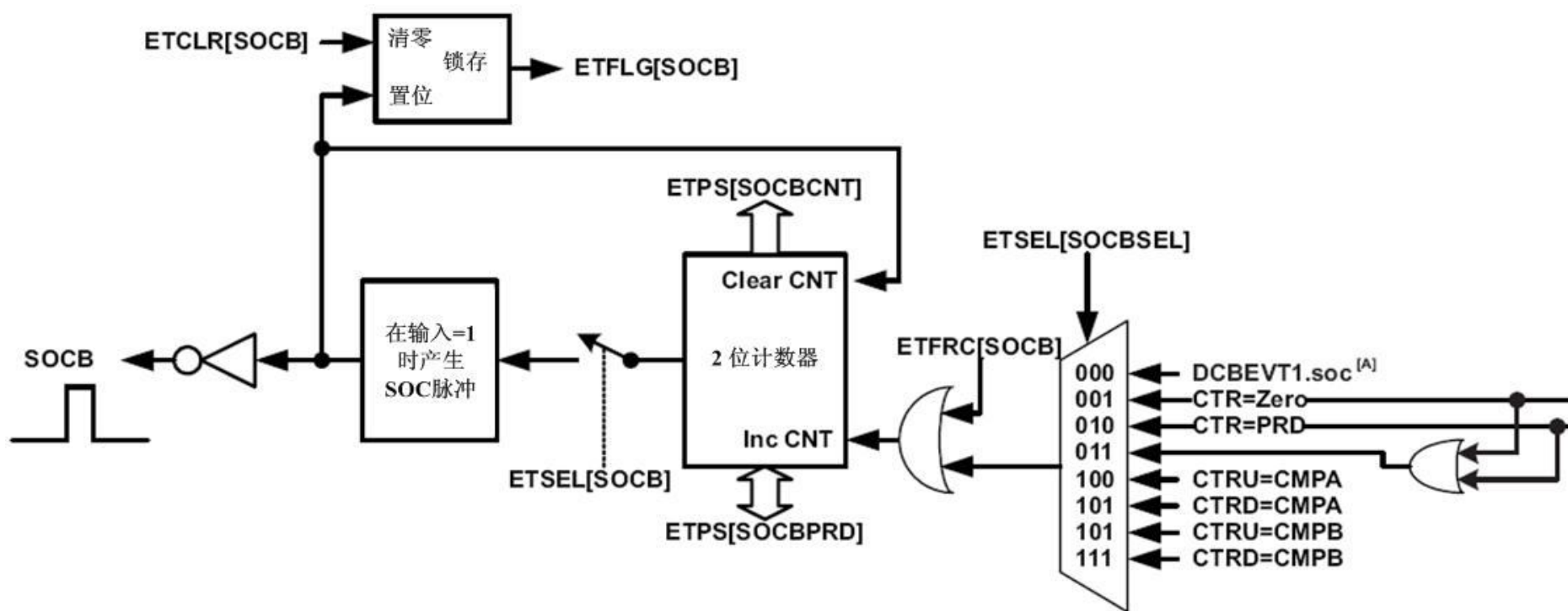




# Lecture7 ePWM

## 九、ePWM子模块功能和配置 -- 事件触发器 (ET)

### 2、事件触发器 (ET) 的组成 -- SOCB发生器





## Lecture7 ePWM

### 九、ePWM子模块功能和配置 -- 事件触发器 (ET)

#### 3、事件触发器 (ET) 的寄存器 -- ETSEL

15	14	12	11	10	8
SOCBEN	SOCBSEL	SOCAEN	SOCASEL		
R/W-0	R/W-0	R/W-0	R/W-0		
7	4	3	2	0	
Reserved	INTEN	INTSEL			
R-0	R/W-0	R/W-0			

- **BIT15: EPWMxSOCB使能位。**      **0 – 禁止      1 – 使能**
- **BIT14-12: EPWMxSOCB选项位。** 决定何时产生**SOCB**
  - 000 - DCBEVT1.soc事件**      **001 - CTR=0**
  - 010 - CTR=PRD**      **011 - CTR=0或PRD**
  - 100 - CAU**      **101 - CAD**
  - 110 - CBU**      **111 - CBD**



## Lecture7 ePWM

### 九、ePWM子模块功能和配置 -- 事件触发器 (ET)

#### 3、事件触发器 (ET) 的寄存器 -- ETSEL

15	14	12	11	10	8
SOCBEN	SOCBSEL	SOCAEN	SOCASEL		
R/W-0	R/W-0	R/W-0	R/W-0		
7	4	3	2	0	
Reserved	INTEN	INTSEL			
R-0	R/W-0	R/W-0			

➤ **BIT11: EPWMxSOCA使能位。**      **0 – 禁止**      **1 – 使能**

➤ **BIT14-12: EPWMxSOCA选项位。** 决定何时产生SOCA

**000 - DCAEVT1.soc事件**

**001 - CTR=0**

**010 - CTR=PRD**

**011 - CTR=0或PRD**

**100 - CAU**

**101 - CAD**

**110 - CBU**

**111 - CBD**



## Lecture7 ePWM

### 九、ePWM子模块功能和配置 -- 事件触发器 (ET)

#### 3、事件触发器 (ET) 的寄存器 -- ETSEL

15	14	12	11	10	8
SOCBEN	SOCBSEL		SOCAEN	SOCASEL	
R/W-0	R/W-0		R/W-0	R/W-0	
7	4	3	2	0	
Reserved		INTEN	INTSEL		
R-0		R/W-0	R/W-0		

➤ **BIT3: EPWM<sub>x</sub>\_INT**中断使能位。

**0** – 禁止      **1** – 使能

➤ **BIT2-0: EPWM<sub>x</sub>\_INT**中断选项位。

**000** – 保留

**001** - CTR=0

**010** - CTR=PRD

**011** - CTR=0或PRD

**100** - CAU

**101** - CAD

**110** - CBU

**111** - CBD



## Lecture7 ePWM

### 九、ePWM子模块功能和配置 -- 事件触发器 (ET)

#### 3、事件触发器 (ET) 的寄存器 -- ETPS

15	14	13	12	11	10	9	8
SOCBCNT		SOCBPRD		SOCACNT		SOCAPRD	
R-0		R/W-0		R-0		R/W-0	
7		4	3	2	1	0	
Reserved				INTCNT		INTPRD	
R-0				R-0		R/W-0	

➤ **BIT15-14: EPWMxSOCB计数器寄存器。**

**00** – 没事件发生

**01** – 发生了1个事件

**10** – 发生了2个事件

**11** – 发生了3个事件

➤ **BIT13-12: EPWMxSOCB产生周期。**

**00** – 不产生EPWMxSOCB

**01** – 第一个事件产生

**10** – 第二个事件产生

**11** – 第三个事件产生



## Lecture7 ePWM

### 九、ePWM子模块功能和配置 -- 事件触发器 (ET)

#### 3、事件触发器 (ET) 的寄存器 -- ETPS

15	14	13	12	11	10	9	8
SOCBCNT		SOCBPRD		SOCACNT		SOCAPRD	
R-0		R/W-0		R-0		R/W-0	
7		4	3	2	1	0	
Reserved			INTCNT		INTPRD		
R-0			R-0		R/W-0		

➤ **BIT11-10: EPWMxSOCA计数器寄存器。**

**00** – 没事件发生

**01** – 发生了1个事件

**10** – 发生了2个事件

**11** – 发生了3个事件

➤ **BIT9-8: EPWMxSOCA产生周期。**

**00** – 不产生EPWMxSOCA

**01** – 第一个事件产生

**10** – 第二个事件产生

**11** – 第三个事件产生





## Lecture7 ePWM

### 九、ePWM子模块功能和配置 -- 事件触发器 (ET)

#### 3、事件触发器 (ET) 的寄存器 -- ETPS

15	14	13	12	11	10	9	8
SOCBCNT		SOCBPRD		SOCACNT		SOCAPRD	
R-0		R/W-0		R-0		R/W-0	
7		4	3	2	1	0	
Reserved				INTCNT		INTPRD	
R-0				R-0		R/W-0	

➤ **BIT3-2: EPWM<sub>x</sub>\_INT计数器寄存器。**

**00** – 没事件发生

**01** – 发生了1个事件

**10** – 发生了2个事件

**11** – 发生了3个事件

➤ **BIT1-2: EPWM<sub>x</sub>\_INT产生周期。**

**00** – 不产生中断

**01** – 第一个事件产生

**10** – 第二个事件产生

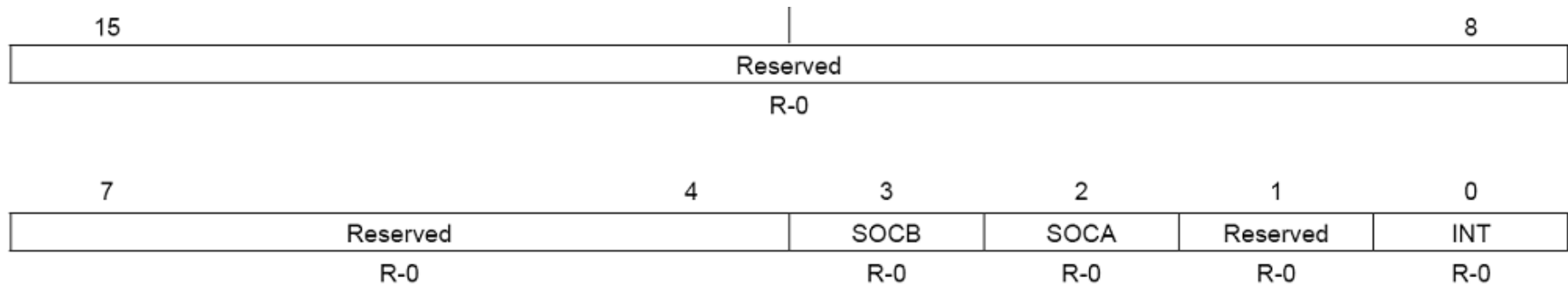
**11** – 第三个事件产生



## Lecture7 ePWM

### 九、ePWM子模块功能和配置 -- 事件触发器 (ET)

#### 3、事件触发器 (ET) 的寄存器 -- ETFLG



➤ **BIT3: EPWMxSOCB**的状态标志位。

0 – 没有SOCB事件发生

1 –有SOCB事件发生

➤ **BIT2: EPWMxSOCA**的状态标志位。

0 – 没有SOCA事件发生

1 –有SOCA事件发生

➤ **BIT0: EPWMx\_INT**的中断标志位。

0 – 没有中断事件发生

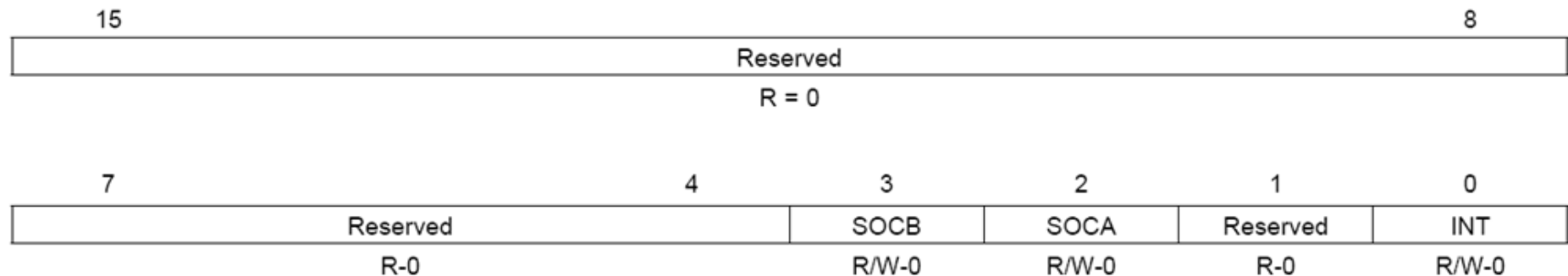
1 –有中断事件发生



## Lecture7 ePWM

### 九、ePWM子模块功能和配置 -- 事件触发器 (ET)

#### 3、事件触发器 (ET) 的寄存器 -- ETCLR



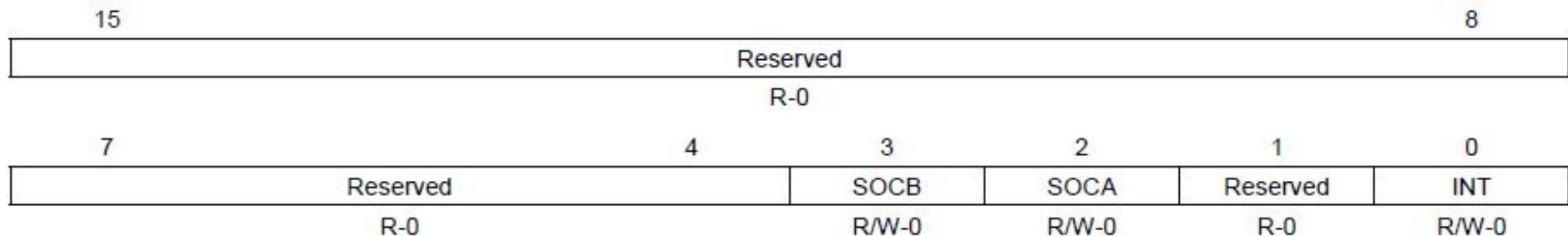
- **BIT3: EPWMxSOCB的标志清零位。读总为0。**  
写0 – 无效      写1 – 清SOCB标志位
- **BIT2: EPWMxSOCA的标志清零位。读总为0。**  
写0 – 无效      写1 – 清SOCA标志位
- **BIT0: EPWMx\_INT的标志清零位。读总为0。**  
写0 – 无效      写1 – 清EPWMx\_INT标志位



## Lecture7 ePWM

### 九、ePWM子模块功能和配置 -- 事件触发器 (ET)

#### 3、事件触发器 (ET) 的寄存器 -- ETFRC



- **BIT3: EPWMxSOCB的强制位。读总为0。**  
写0 – 无效      写1 – 产生一个SOCB且SOCBFLG置位
- **BIT2: EPWMxSOCA的强制位。读总为0。**  
写0 – 无效      写1 – 产生一个SOCA且SOCAFLG置位
- **BIT0: EPWMx\_INT的强制位。读总为0。**  
写0 – 无效      写1 – 产生一个INT且INTFLG置位



## Lecture7 ePWM

### 九、ePWM子模块功能和配置 -- 事件触发器 (ET)

#### 4、正确产生EPWMx\_INT的初始化步骤

当ePWM外设时钟使能时，由于ePWM寄存器没有正确初始化可能会导致伪事件（**spurious event**），进而使得中断标志可能置位。ePWM外设正确初始化的顺序如下：

1. Disable global interrupts (CPU INTM flag)
2. Disable ePWM interrupts
3. Set TBCLKSYNC=0
4. Initialize peripheral registers
5. Set TBCLKSYNC=1
6. Clear any spurious ePWM flags (including PIEIFR)
7. Enable ePWM interrupts
8. Enable global interrupts



## Lecture7 ePWM

### 十、ePWM子模块功能和配置 -- 数字比较 (DC)

#### 1、数字比较 (DC) 的功能

- 比较器 (COMP) 输出和触发区信号输入 (TZ1~TZ3) 创建事件与滤波事件 (**filtered event**)
- 比较事件要不被滤波要不就被直接送到触发区、事件触发器和时基子模块
- 滤波产生屏蔽 (**blank**) 事件，捕获触发事件的TBCTR值



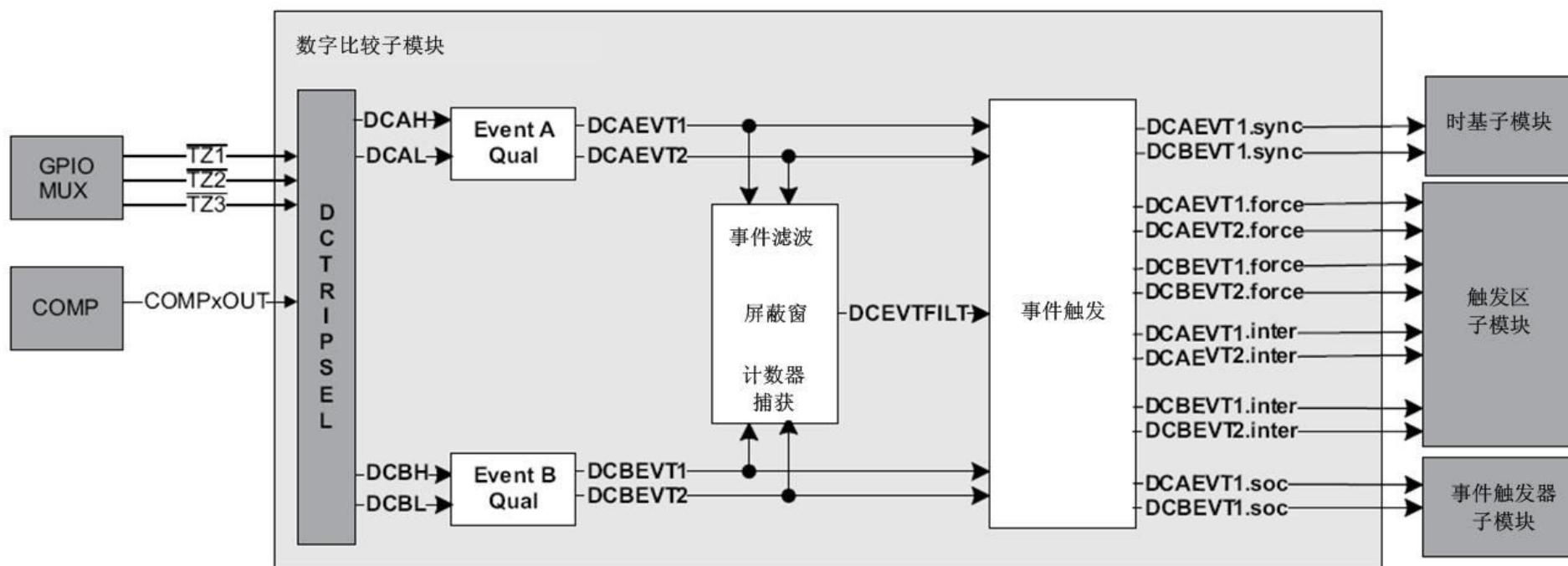




# Lecture7 ePWM

## 十、ePWM子模块功能和配置 -- 数字比较 (DC)

### 2、数字比较 (DC) 的组成





## Lecture7 ePWM

### 十、ePWM子模块功能和配置 -- 数字比较 (DC)

#### 3、数字比较 (DC) 的输出信号

➤ **强制 (force)** 信号→TZ:

影响EPWMxA/B引脚上的输出状态 (通过配置)  
被选作单次触发或周期性触发源

➤ **中断 (inter)** 信号→TZ → PIE:

通过TZEINT寄存器使能, 通过TZCLR可清除中断  
触发一个EPWMxTZINT中断

➤ **开始转换 (soc)** 信号→ET → ADC:

通过ETSEL配置产生SOC信号 (A/B两路)

➤ **同步 (sync)** 信号→TB:

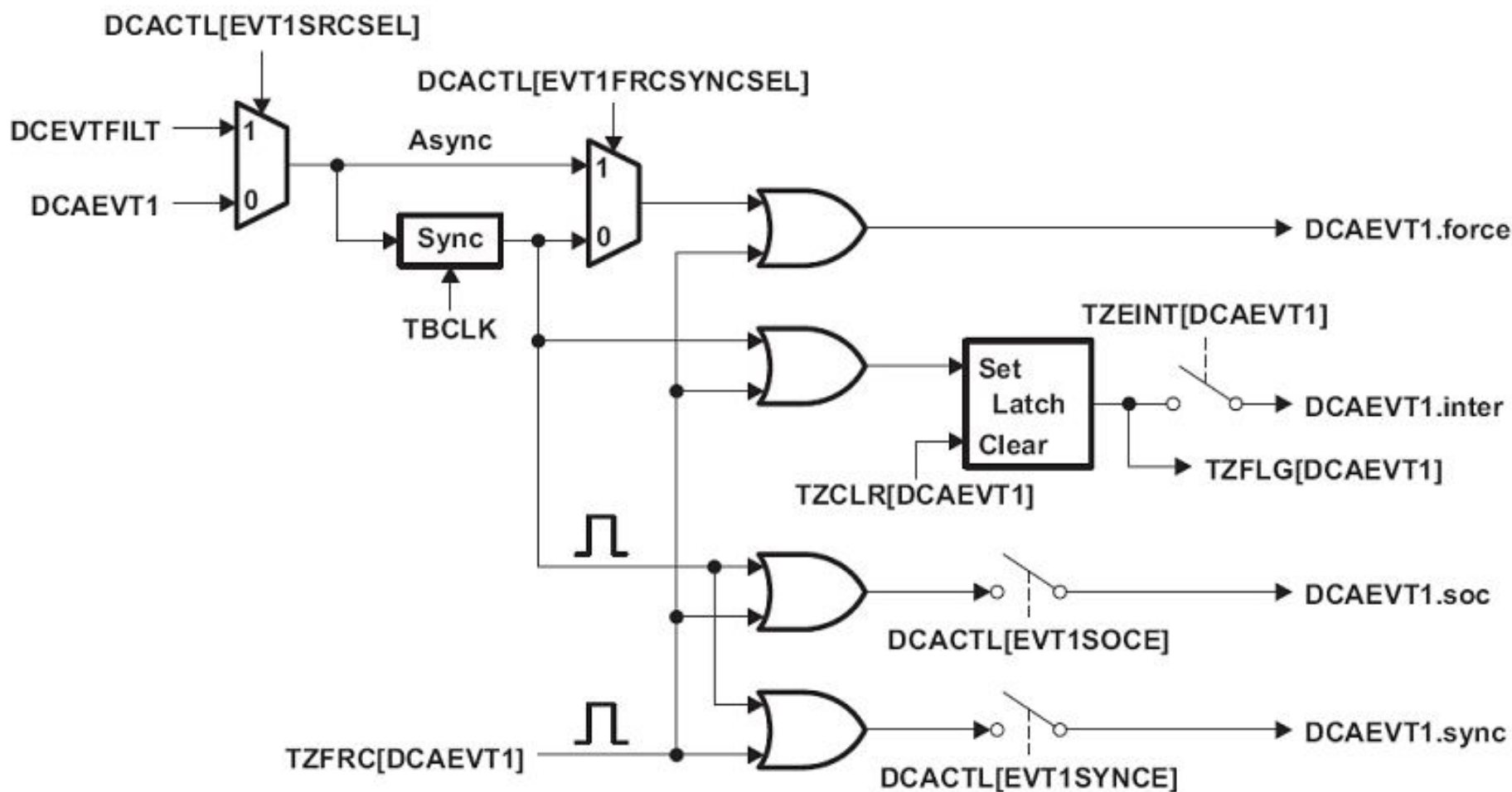




## Lecture7 ePWM

### 十、ePWM子模块功能和配置 -- 数字比较 (DC)

#### 4、数字比较 (DC) 的输出逻辑 -- DCAEVT1

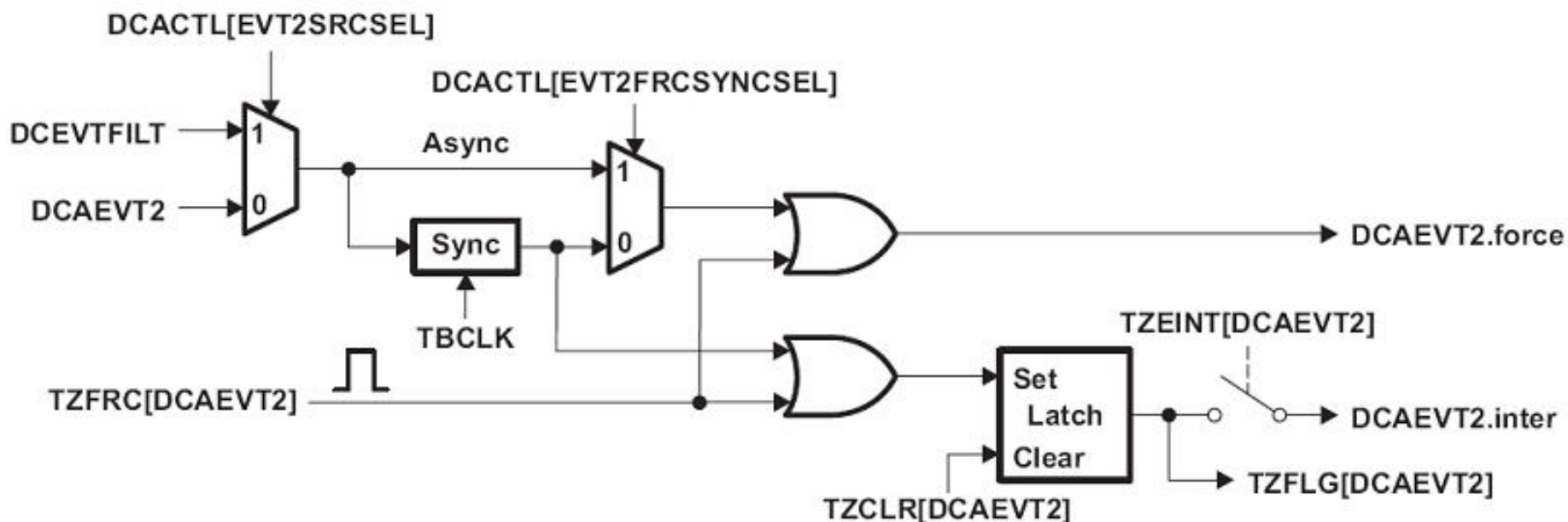




## Lecture7 ePWM

### 十、ePWM子模块功能和配置 -- 数字比较 (DC)

#### 4、数字比较 (DC) 的输出逻辑 -- DCAEVT2

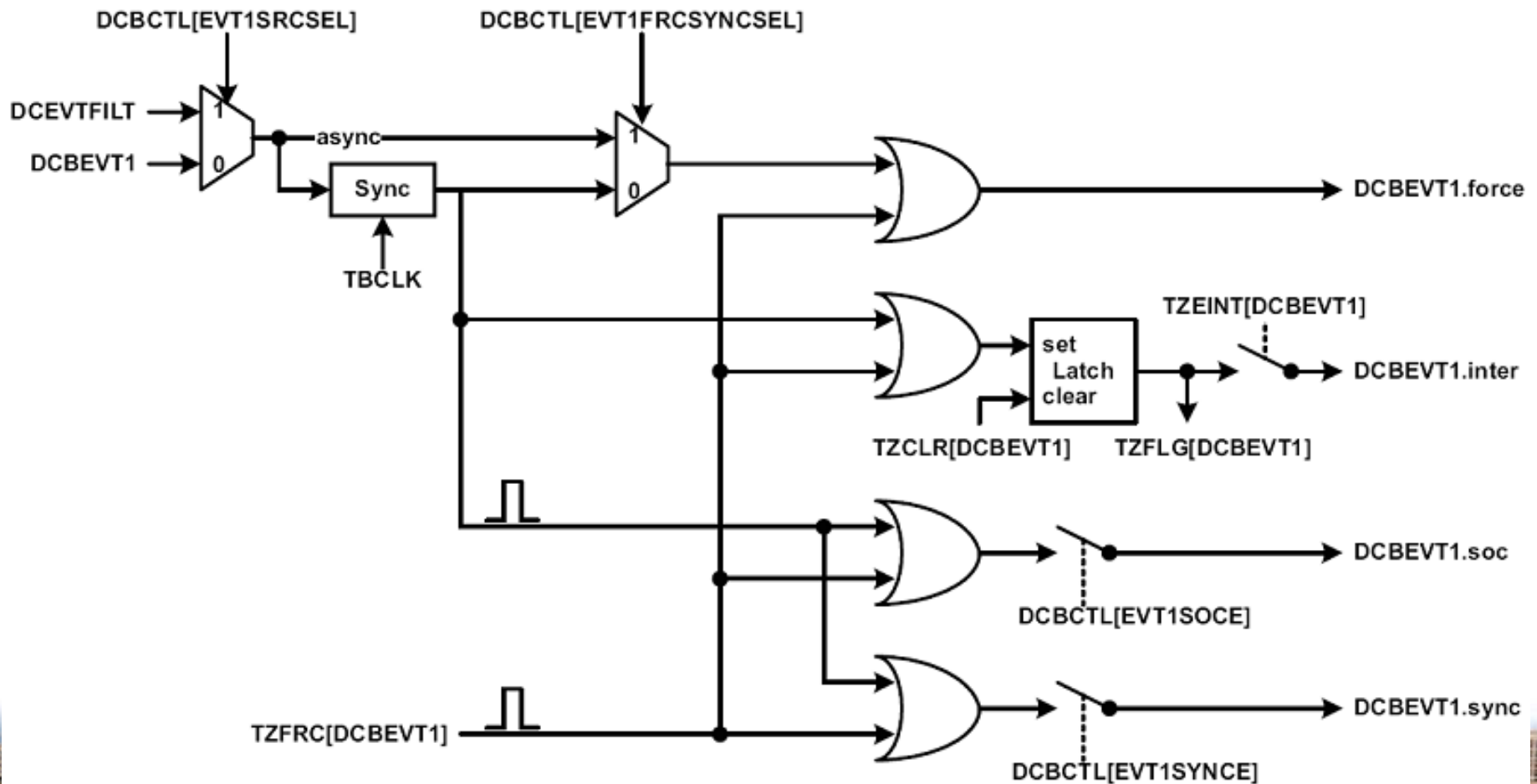




# Lecture7 ePWM

## 十、ePWM子模块功能和配置 -- 数字比较 (DC)

### 4、数字比较 (DC) 的输出逻辑 -- DCBEVT1

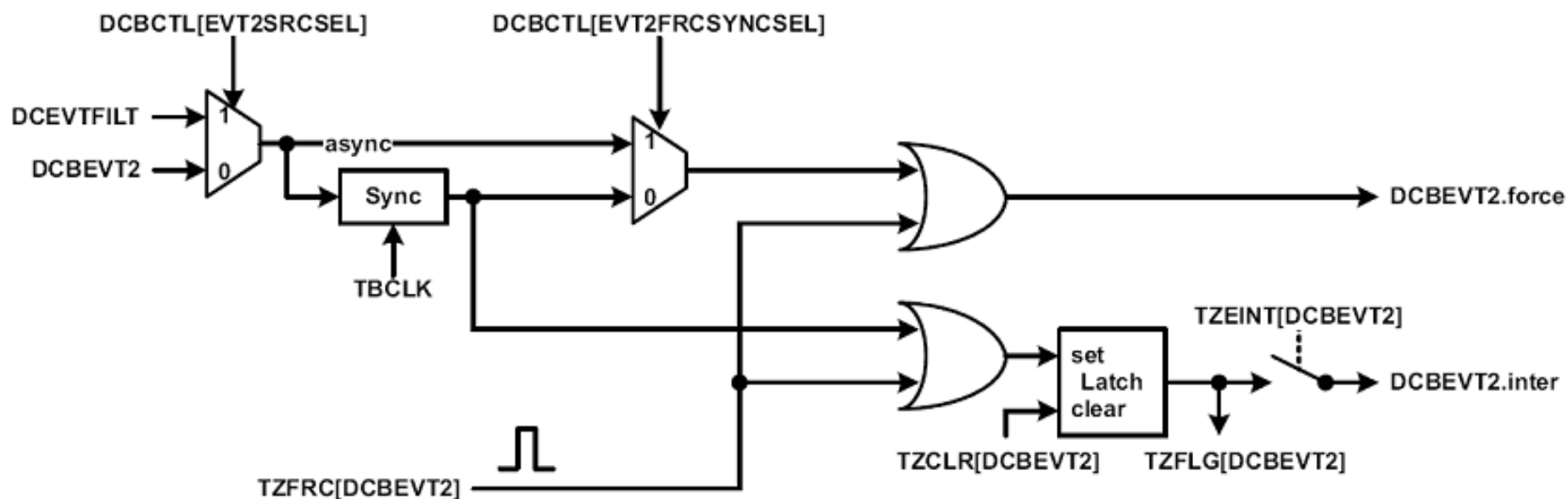




## Lecture7 ePWM

### 十、ePWM子模块功能和配置 -- 数字比较 (DC)

#### 4、数字比较 (DC) 的输出逻辑 -- DCBEVT2



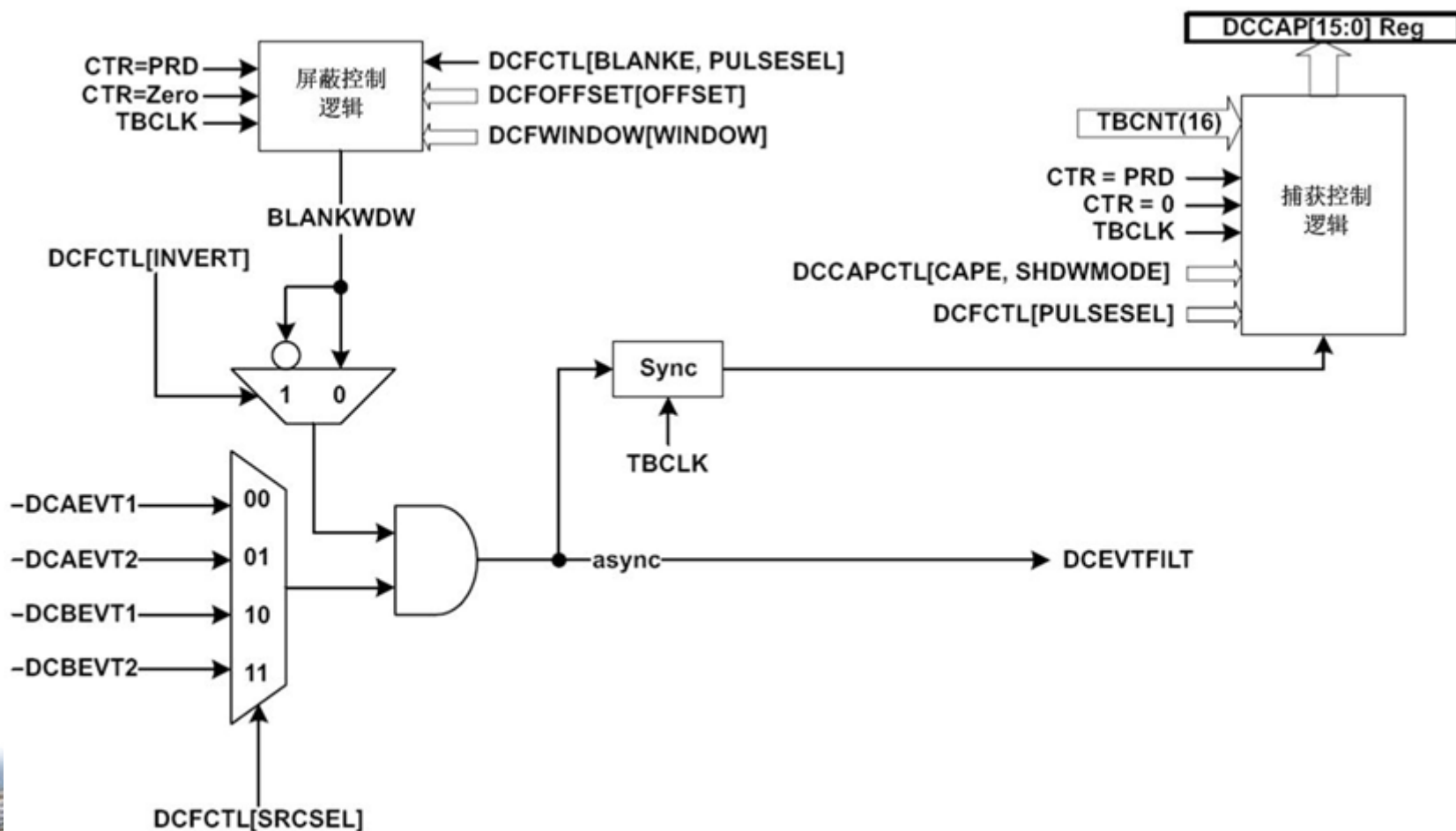




# Lecture7 ePWM

## 十、ePWM子模块功能和配置 -- 数字比较 (DC)

### 4、数字比较 (DC) 的输出逻辑 -- DCEVTFILT

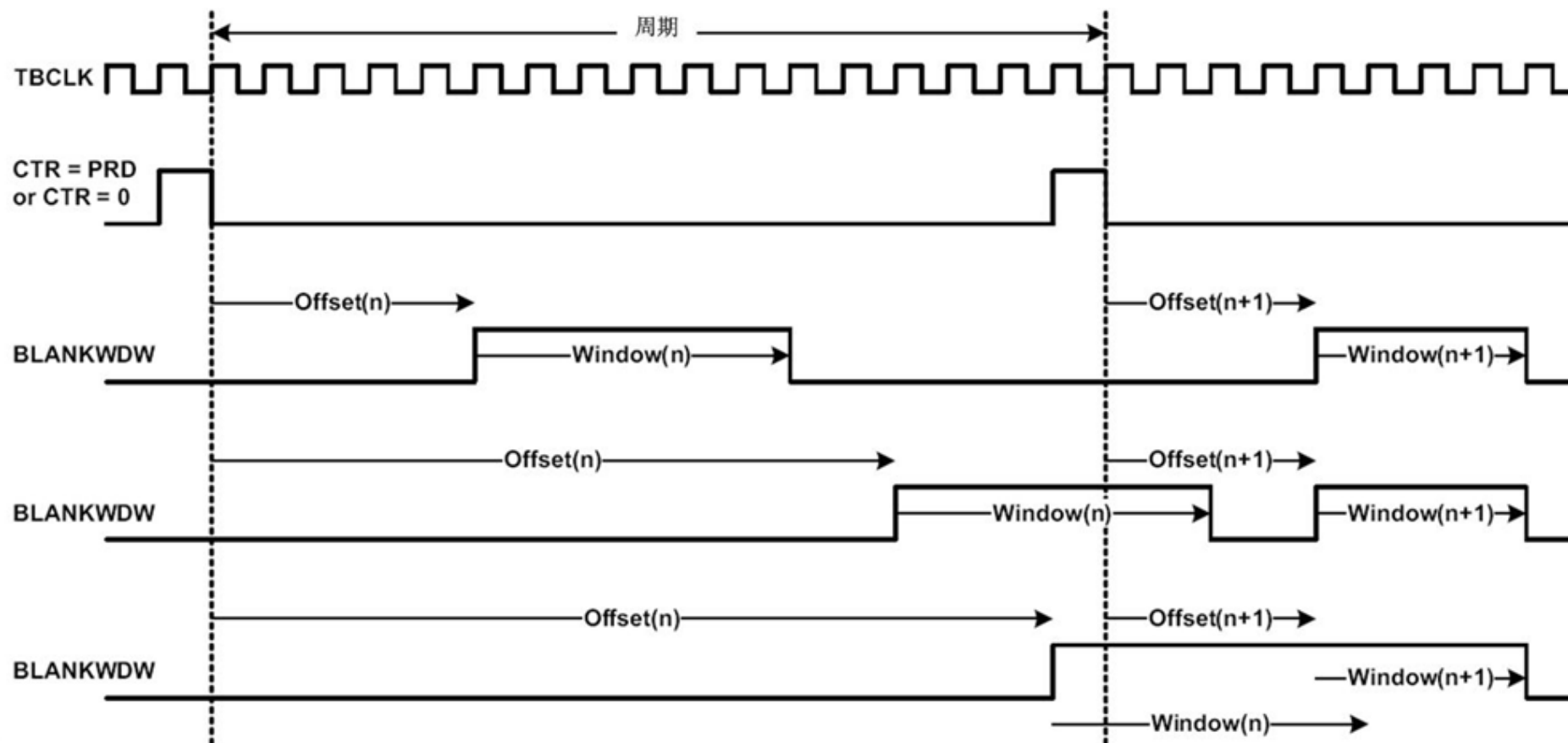




## Lecture7 ePWM

### 十、ePWM子模块功能和配置 -- 数字比较 (DC)

#### 4、数字比较 (DC) 的输出逻辑 -- 屏蔽窗时序

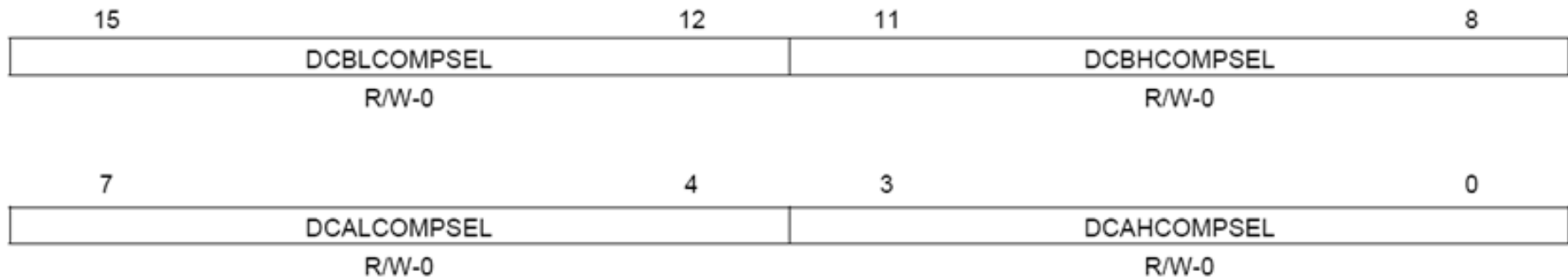




## Lecture7 ePWM

十、ePWM子模块功能和配置 -- 数字比较 (DC)

5、数字比较 (DC) 的寄存器 -- DCTRIPSEL



➤ 数字比较输入选择位: **DCBL**、**DCBH**、**DCAL**、**DCAH**

0000 –  $\overline{\text{TZ1}}$ 输入

0001 –  $\overline{\text{TZ2}}$ 输入

0010 –  $\overline{\text{TZ3}}$ 输入

1000 – COMP1OUT输入

1001 – COMP2OUT输入

1010 – COMP3OUT输入

其他值 – 保留



## Lecture7 ePWM

### 十、ePWM子模块功能和配置 -- 数字比较 (DC)

#### 5、数字比较 (DC) 的寄存器 -- DCACTL

15	Reserved				10	EVT2FRC SYNCSEL		9	EVT2SRCSEL		8						
				R-0			R/W-0			R/W-0							
7	Reserved				4	EVT1SYNCE		3	EVT1SOCE		2						
				R-0			R/W-0			R/W-0							
												1	EVT1FRC SYNCSEL		0	EVT1SRCSEL	
												R/W-0			R/W-0		

➤ **BIT9: DCAEVT2强制同步信号选择位。**

0 –源是同步信号

1 –源是异步信号

➤ **BIT8: DCAEVT2源信号选择位。**

0 –源是DCAEVT2信号

1 –源是DCEVTFILT信号

➤ **BIT3: DCAEVT1 SYNC使能/禁能位。**

0 –禁止产生SYNC信号

1 –可以产生SYNC信号



## Lecture7 ePWM

### 十、ePWM子模块功能和配置 -- 数字比较 (DC)

#### 5、数字比较 (DC) 的寄存器 -- DCACTL

15	10	9	8		
Reserved		EVT2FRC SYNCSEL	EVT2SRCSEL		
R-0		R/W-0	R/W-0		
7	4	3	2	1	0
Reserved		EVT1SYNCE	EVT1SOCE	EVT1FRC SYNCSEL	EVT1SRCSEL
R-0		R/W-0	R/W-0	R/W-0	R/W-0

➤ **BIT2: DCAEVT1 SOC使能/禁能位。**

**0** –禁止产生**SOC**信号

**1** –可以产生**SOC**信号

➤ **BIT1: DCAEVT1强制同步信号选择位。**

**0** –源是同步信号

**1** –源是异步信号

➤ **BIT0: DCAEVT1源信号选择位。**

**0** –源是**DCAEVT1**信号

**1** –源是**DCEVTFILT**信号



## Lecture7 ePWM

### 十、ePWM子模块功能和配置 -- 数字比较 (DC)

#### 5、数字比较 (DC) 的寄存器 -- DCBCTL

15	10	9	8		
Reserved		EVT2FRC SYNCSEL	EVT2SRCSEL		
R-0		R/W-0	R/W-0		
7	4	3	2	1	0
Reserved		EVT1SYNCE	EVT1SOCE	EVT1FRC SYNCSEL	EVT1SRCSEL
R-0		R/W-0	R/W-0	R/W-0	R/W-0

➤ **BIT9: DCBEVT2强制同步信号选择位。**

0 –源是同步信号

1 –源是异步信号

➤ **BIT8: DCBEVT2源信号选择位。**

0 –源是DCBEVT2信号

1 –源是DCEVTFILT信号

➤ **BIT3: DCBEVT1 SYNC使能/禁能位。**

0 –禁止产生SYNC信号

1 –可以产生SYNC信号





## Lecture7 ePWM

### 十、ePWM子模块功能和配置 -- 数字比较 (DC)

#### 5、数字比较 (DC) 的寄存器 -- DCACTL

15	10	9	8		
Reserved		EVT2FRC SYNCSEL	EVT2SRCSEL		
R-0		R/W-0	R/W-0		
7	4	3	2	1	0
Reserved		EVT1SYNCE	EVT1SOCE	EVT1FRC SYNCSEL	EVT1SRCSEL
R-0		R/W-0	R/W-0	R/W-0	R/W-0

➤ **BIT2: DCBEVT1 SOC使能/禁能位。**

**0** –禁止产生**SOC**信号

**1** –可以产生**SOC**信号

➤ **BIT1: DCBEVT1强制同步信号选择位。**

**0** –源是同步信号

**1** –源是异步信号

➤ **BIT0: DCBEVT1源信号选择位。**

**0** –源是**DCBEVT1**信号

**1** –源是**DCEVTFILT**信号



## Lecture7 ePWM

### 十、ePWM子模块功能和配置 -- 数字比较 (DC)

#### 5、数字比较 (DC) 的寄存器 -- DCFCTL

15	13	12	8				
Reserved		Reserved					
R-0		R-0					
7	6	5	4	3	2	1	0
Reserved	Reserved	PULSESEL	BLANKINV	BLANKE	SRCSEL		
R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0		

➤ **BIT5-4: 屏蔽&捕获参照的脉冲选择位。**

**00 –TBCTR = TBPRD**

**01 –TBCTR = 0x0000**

**1x – 保留**

➤ **BIT2: 屏蔽窗反相位。**

**0 – 不反相**

**1 – 反相**

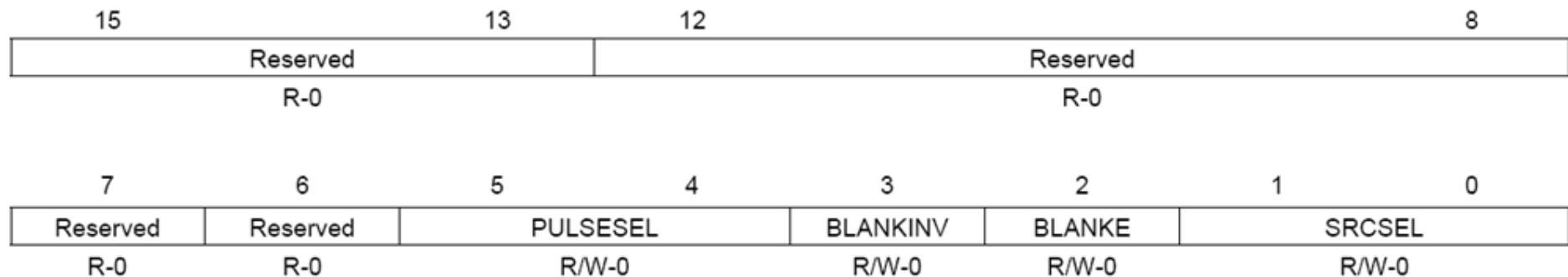




## Lecture7 ePWM

### 十、ePWM子模块功能和配置 -- 数字比较 (DC)

#### 5、数字比较 (DC) 的寄存器 -- DCFCTL



➤ **BIT2: 屏蔽窗使能位。**

**0 – 禁止                      1 – 使能**

➤ **BIT1-0: 滤波模块的信号源选择位。**

**00 – 源是DCAEVT1信号**

**01 – 源是DCAEVT2信号**

**10 – 源是DCBEVT1信号**

**11 – 源是DCBEVT2信号**

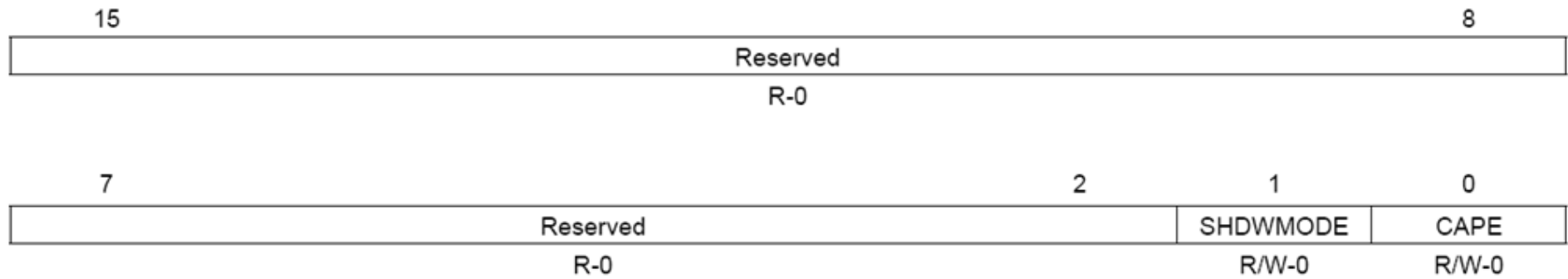




## Lecture7 ePWM

### 十、ePWM子模块功能和配置 -- 数字比较 (DC)

#### 5、数字比较 (DC) 的寄存器 -- DCCAPCTL



➤ **BIT2: DCCAP的shadow模式选择位。**

**0 –使能DCCAP=shadow**

**1 – 禁止DCCAP=active**

➤ **BIT1-0: DCCAP使能位。**

**0 – 禁止**

**1 – 使能**

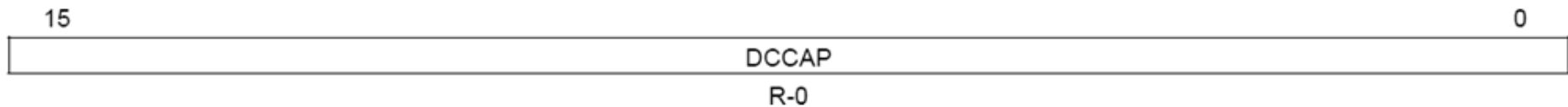




## Lecture7 ePWM

### 十、ePWM子模块功能和配置 -- 数字比较 (DC)

#### 5、数字比较 (DC) 的寄存器 -- DCCAP



➤ **BIT15-0: 数字比较时基计数器捕获值。0000-FFFFh**

使能时基计数器捕获 (**DCCAPCLT.CAPE=1**) :

捕捉时基计数器 (**TBCTR**) 在滤波事件 (**DCEVTFLT**)

从低到高跳变时的值。另外，捕获事件被忽略，直到下一周期或**0** (由**DCFCTL.PULSESEL**位选择)。

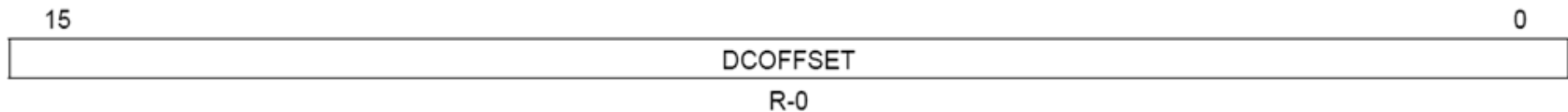




## Lecture7 ePWM

### 十、ePWM子模块功能和配置 -- 数字比较 (DC)

#### 5、数字比较 (DC) 的寄存器 -- DCOFFSET



➤ **BIT15-0: 屏蔽窗偏移量。0000-FFFFh**

该寄存器具有shadow寄存器。

该值指定从屏蔽窗参考点距离屏蔽窗开始的**TBCLK**周期数。



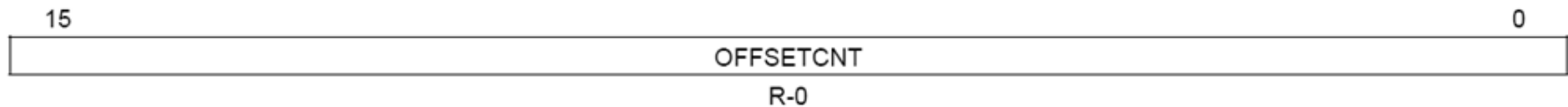




## Lecture7 ePWM

十、ePWM子模块功能和配置 -- 数字比较 (DC)

5、数字比较 (DC) 的寄存器 -- **DCOFFSETCNT**



➤ **BIT15-0: 屏蔽窗偏移量计数器。0000-FFFFh**

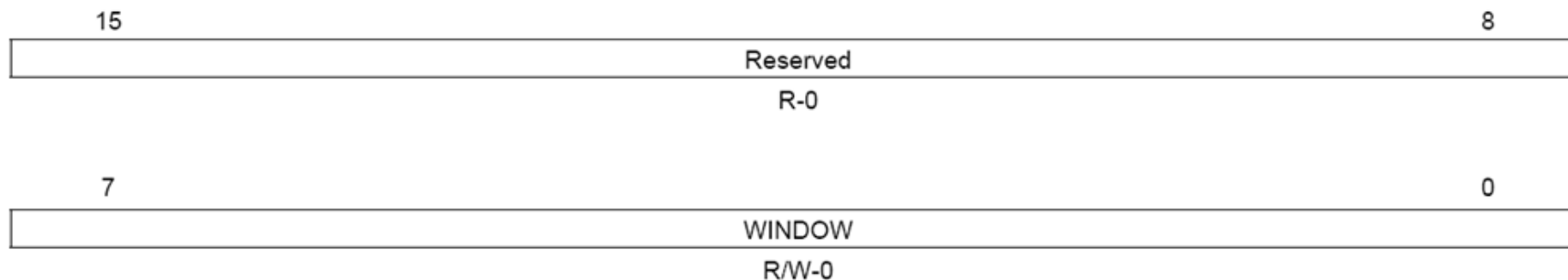




## Lecture7 ePWM

### 十、ePWM子模块功能和配置 -- 数字比较 (DC)

#### 5、数字比较 (DC) 的寄存器 -- DCFWINDOW



➤ **BIT7-0: 屏蔽窗宽度。00-FFh**

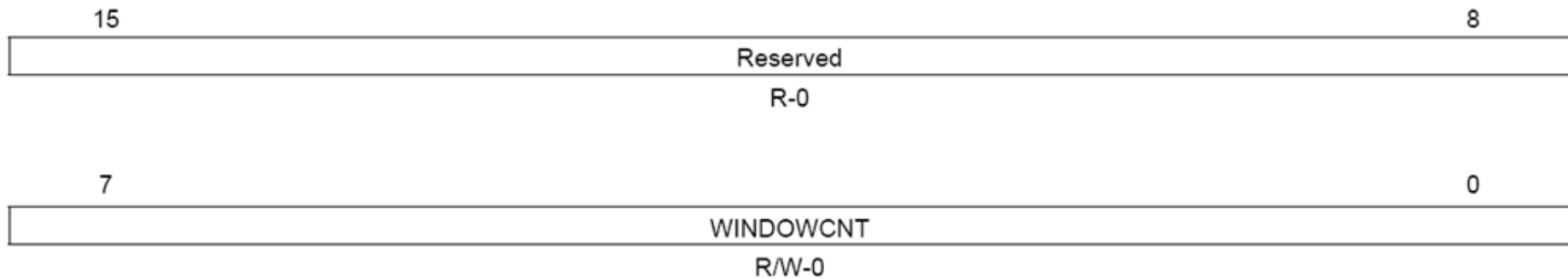




## Lecture7 ePWM

### 十、ePWM子模块功能和配置 -- 数字比较 (DC)

#### 5、数字比较 (DC) 的寄存器 -- DCFWINDOWCNT



➤ **BIT15-0: 屏蔽窗宽度计数器。00-FFh**





## Lecture7 ePWM

### 十一、ePWM编程实现

#### 1、DSP2802x\_EPWM.h寄存器定义

**EPwm1Regs.寄存器名.bit.位名**

**EPwm2Regs.寄存器名.bit.位名**

**EPwm3Regs.寄存器名.bit.位名**

**EPwm4Regs.寄存器名.bit.位名**





## Lecture7 ePWM

### 十一、ePWM编程实现

#### 2、DSP2802x\_EPwm.c函数子程序

- InitEPwmGpio();**            引脚复用设置、禁止内部上拉。
- InitEPwm1Gpio();**        引脚复用设置、禁止内部上拉。
- InitEPwm2Gpio();**        引脚复用设置、禁止内部上拉。
- InitEPwm3Gpio();**        引脚复用设置、禁止内部上拉。
- InitEPwm4Gpio();**        引脚复用设置、禁止内部上拉。
- InitEPwmSyncGpio();** **EPWMSYNCl、EPWMSYNCO**引脚复用设置、  
使能内部上拉。 **GPIO2\GPIO32\GPIO33**
- InitTzGpio();**            **TZ1、TZ2、TZ3**引脚复用设置、禁止上拉、异步





## Lecture7 ePWM

### 十一、ePWM编程实现

#### 3、EPWM初始化流程

- 使能**外设时钟**（PCLKCR1.0-3位置1、PCLKCR0.2位清0【同步】）
- **GPIO引脚复用设置**（ePWMxA/B、EPWMSYNCI、EPWMSYNCO、TZ1、TZ2、TZ3）
- 初始化**中断**（参考PIE章节）
- 时基（**TB**）设置（分频、周期值、计数模式、同步）
- 计数器-比较（**CC**）设置（占空比）
- 动作限定器（**AQ**）设置（6个时刻PWM引脚**动作**）







## Lecture7 ePWM

### 十一、ePWM编程实现

#### 3、EPWM初始化流程

- 死区（**DB**）设置（死区宽度、模式）
- 斩波（**PC**）设置（载波频率、占空比、第一个脉宽）
- 触发区（**TZ**）设置（触发选择、动作、中断）
- 事件触发器（**ET**）设置（中断使能、**SOC**使能、事件频率、）
- 数字比较（**DC**）设置（输入、使能信号force\inter\soc\sync、屏蔽窗）
- 使能同步时钟（**PCLKCR0.2位置1【同步】**）
- 编写中断服务子程序（**ISR**）（修改占空比）





**Thank You & Question?**

