



<http://www.Microcontrol.cn> 微控设计网

中国 MSP430 单片机专业网站

MSP430F2 系列 16 位超低功耗单片机模块原理

## 第 8 章 定时器 A TimerA

版本: 1.2

日期: 2007.4.

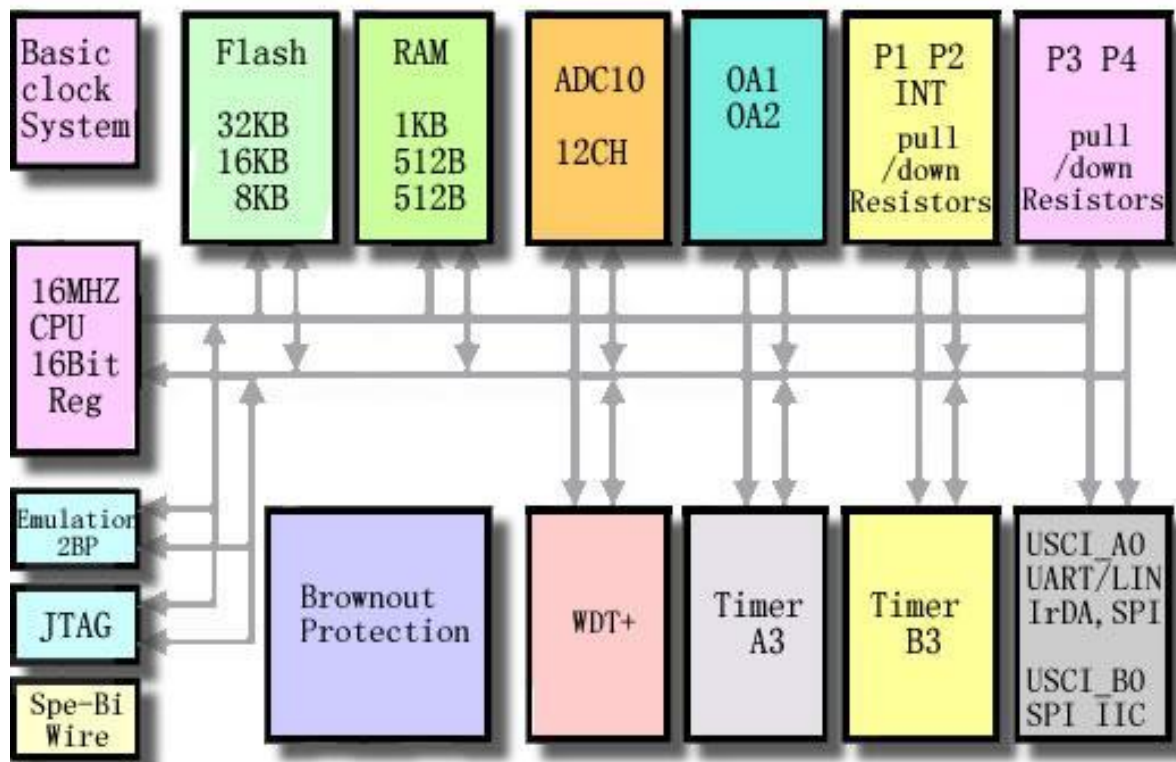
原文: TI MSP430x2xxfamily.pdf

翻译: 李璘 中国计量学院

编辑: DC 微控技术论坛总版主

注: 以下文章是翻译 TI MSP430x2xxfamily.pdf 文件中的部分内容。由于我们翻译水平有限, 有整理过程中难免有所不足或错误; 所以以下内容只供参考. 一切以原文为准。

详情请密切留意微控技术论坛。



## 第八章 定时器 A

定时器 A (Timer\_A, 以后简称为 TA) 是一个 16 位的定时/计数器, 并复合了捕获/比较寄存器。Timer\_A3(拥有 3 个捕获比较器)只存在于 MSP430x2xx, 在 MSP430x20xx 中只有 Timer\_A2 (拥有 2 个捕获比较器)。

章节

8.1 Timer_A 介绍 .....	8-2
8.2 Timer_A 操作方法 .....	8-4
8.3 Timer_A 寄存器 .....	8-19

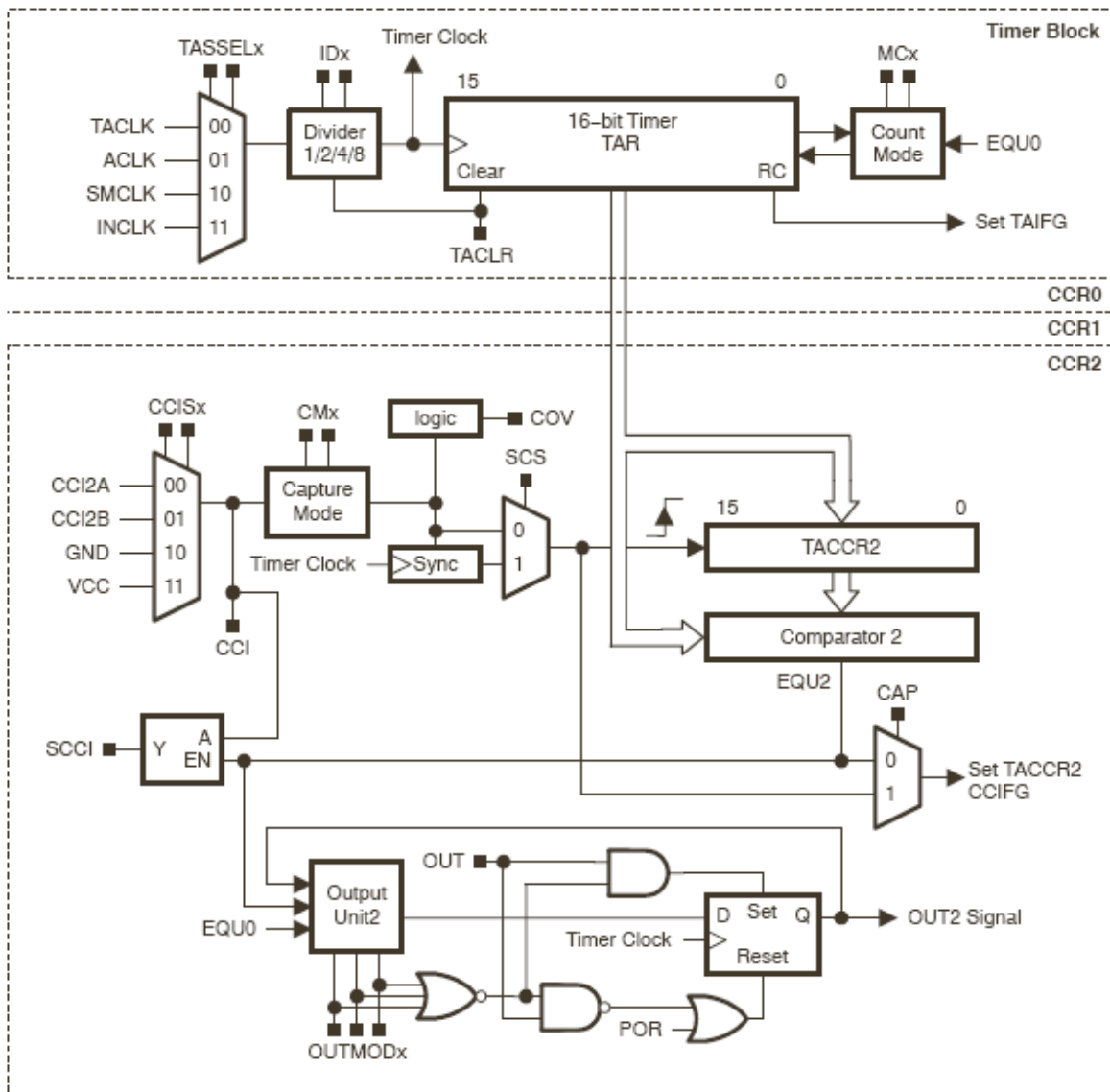
### 8.1 Timer\_A 介绍

TA 是一个 16 位的定时/计数器, 最多拥有 3 个捕获/比较寄存器。TA 可以支持捕获/比较功能、PWM 输出和定时器功能。TA 还有扩展中断的功能, 中断可以由定时器溢出产生或捕获比较寄存器产生。

TA 的特性如下:

- I 4 种操作模式的异步 16 位定时/计数器
- I 可选择配置的时钟源
- I 2 个或 3 个可配置的捕获/比较器
- I 可配置的 PWM 输出
- I 异步输入和输出锁存
- I 对所有 TA 中断快速响应的中断向量寄存器
- I TA 的结构图见图 8-1.

图 8-1.TA 结构图



## 8.2 Timer\_A 的操作方法

TA 模块由用户软件来配置，TA 的配置将在下面的章节讨论。

### 8.2.1 16 位定时/计数器

16 位定时/计数器寄存器 TAR，随着时钟信号的每个上升沿增/减(这由操作模式决定)。TAR 可以被软件读写。另外，定时器在溢出时可以产生中断。TAR 可以由 TACL R 位清除，如果 TA 处于 up/down 模式，TACL R 置位也会清除时钟分频器和计数方向。

**注意:**对 TA 寄存器的修改

建议在进行修改定时器的操作(对中断允许、中断标志和 TACL R 的操作除外)时，先停

止定时器，以避免产生未知的误操作。当定时器时钟和 CPU 时钟不同步时，对 TAR 的读会由于定时器的运行而导致所读的结果是不可预料的。因此，当定时器运行时，需要多读几次，通过软件多数表决的方式来原因正确的读数。对 TAR 的写操作是立即生效的。

### 时钟源的选择和分频

定时器的时钟源可以是内部时钟源 ACLK, SMCLK, 或外部源 TACLK 和 INCLK。时钟源由 TASSEL 位来选择，所选择的时钟可以通过 IDx 位进行 2、4 或 8 分频，当 TACLK 置位时，分频器复位。

### 8.2.2 启动定时器

定时器可以通过以下 2 种方式启动或重新启动：

- I 当定时器计数到 MCX>0 并且时钟源处于活动状态时
- I 当定时器模式为 up 或 up/down 模式时（即单调增和增减模式），定时器可以通过写 0 到 TACCR0 来停止计数。定时器可以通过写一个非 0 的数值来重新开始计数。在这种情况下，定时器从 0 开始增计数。

### 8.2.3 定时器模式控制

定时器有 4 种操作模式，见表 8-1，他们分别是停止、单调增、连续和增减模式。操作模式由 MCx 位来选择。

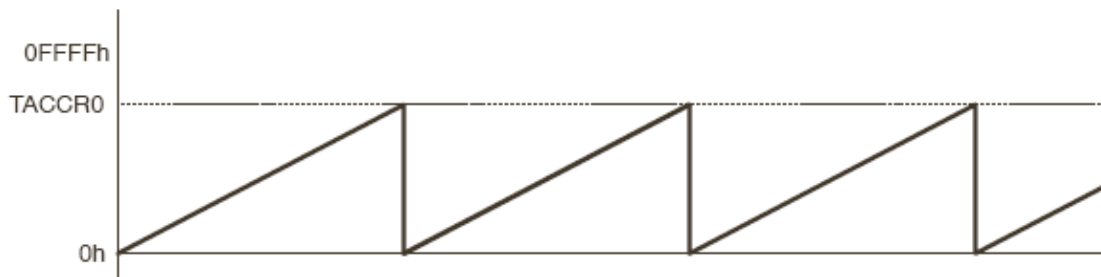
表 8-1 定时器模式

MCx	模式	说明
00	停止模式	定时器暂停
01	单调增模式	定时器循环地从 0 增到 TACCR0 的值
10	连续模式	定时器循环地从 0 连续增加到 0FFFFH
11	增减模式	定时器循环地从 0 增到 TACCR0 的值再连续减至 0

#### (1) 单调增模式

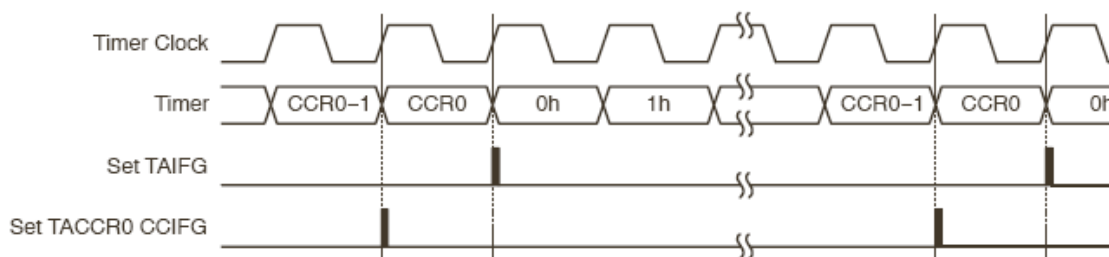
单调增模式用于计数周期不是 0FFFFH 的情况。定时器重复增计数值寄存器 TACCR0 的值，而 TACCR0 的值取决于定时周期，如图 8-2，定时器计数周期为 TACCR0+1。当定时器的值等于 TACCR0 时，定时器就回到 0 重新计数。如果当定时器的值大于 TACCR0，而此时选择单调增模式，定时器立即从 0 重新开始计数

图 8-2. 单调增模式



当定时器计数到 TACCR0 的值时，中断标志 CCIFG 位置位。当定时器由 TACCR0 返回 0 时，TAIFG 中断标志置位。图 8-3 说明了标志置位循环

图 8-3. 单调增模式标志位的变化



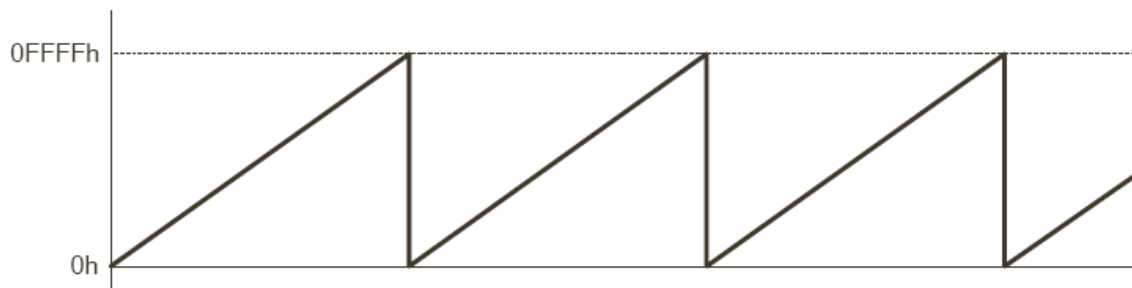
### 修改周期寄存器 TACCR0

在定时器在运行时修改 TACCR0，如果新的周期值大于或等于旧的周期值，或大于当前的定时器计数值，那么定时器立刻开始执行新的周期计数。如果新周期小于当前的计数值，那么定时器回到 0。但是，在回到 0 之前会多一个额外的计数。

### 连续模式

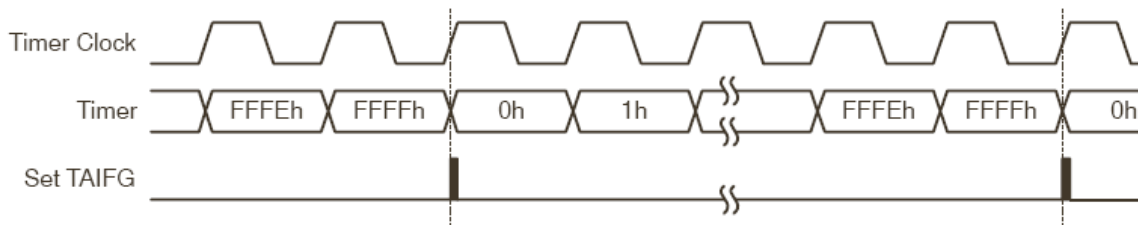
在连续模式中，定时器重复计数到 0FFFFH，然后重新从 0 开始增计数，如图 8-4。捕获比较寄存器 TACCR0 以及其他捕获比较器工作方式一样。

图 8-4.连续模式



当定时器从 0FFFFH 到 0 时，TAIFG 中断标志置位。图 8-5 表示了标志位的设置

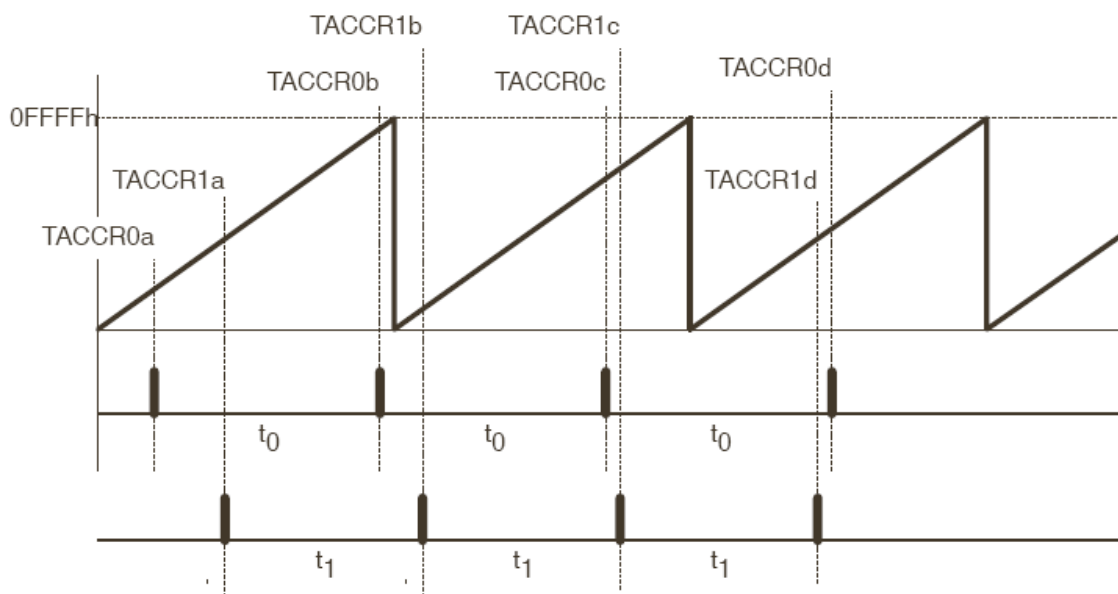
图 8-5. 连续模式标志位的设置



### 连续模式的使用

连续模式可以用于产生统一的时间间隔和输出频率。当每个时间间隔完成时就产生一个中断。下一个时间间隔的值在进入中断服务子程序时写入  $TACCR_x$ 。图 8-6 显示了 2 个独立的时间间隔  $t_0$  和  $t_1$  写入捕获比较寄存器。在该应用中，时间间隔由硬件控制，与中断响应没有冲突。如果想产生多于 3 个的时间间隔可以使用所有的捕获比较寄存器。

图 8-6. 连续模式时间间隔

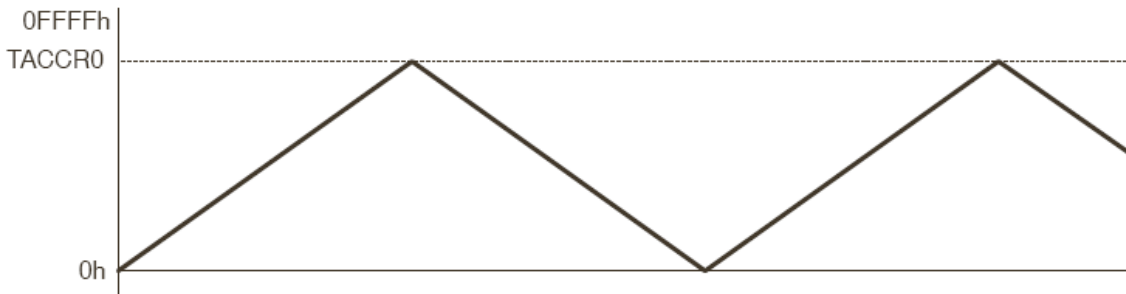


时间间隔可以由其他模式产生， $TACCR_0$  也可以作为周期寄存器使用。如果旧的  $TACCR_0x$  的数据之和与新的相比，要比  $TACCR_0$  大，那么操作就会复杂得多。当旧的  $TACCR_x$  的值加上  $t_x$  比  $TACCR_0$  的值大，那么  $TACCR_0$  的值必须被减掉以获得正确的时间间隔。

### 增减模式

增减模式在定时器周期不是  $0FFFFH$  且需要产生对称的脉冲时使用。定时器增计数到  $TACCR_0$  再从  $TACCR_0$  减计数到 0，如图 8-7，周期是  $TACCR_0$  值的 2 倍。

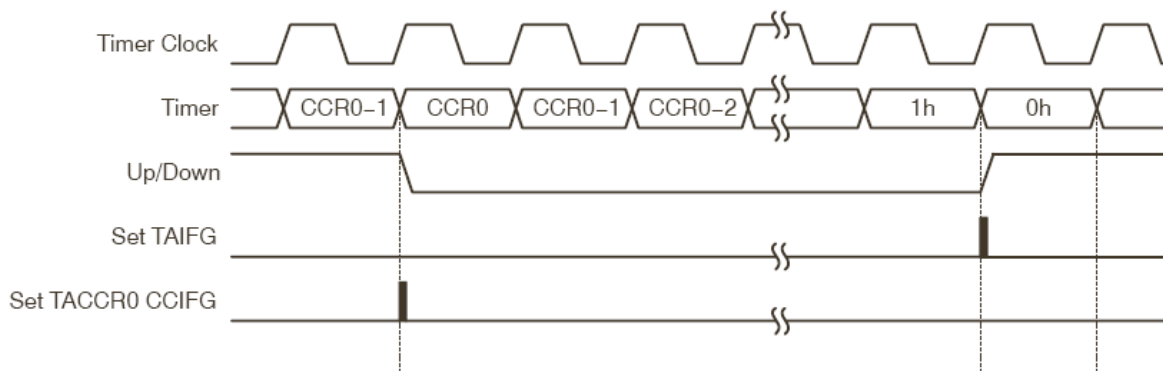
图 8-7. 增减模式



该模式下，计数方向是固定的，即让定时器停止后再重新启动定时器，它就会沿着停止时的计数方向和数值开始计数。如果要从 0 开始，就需要将 TACLRL 置位来清除方向。TACLRL 位也会清除 TAR 的值和定时器的时钟分频。

在 TACCR0 中，CCIFG 中断标志和 TAIFG 中断标志在一个周期中只置位一次，由 1/2 定时器周期隔开。当定时器计数到由 TACCR0-1 变到 TACCR0 时，CCIFG 置位；而定时器完成减计数从 0001h 到 0000h 时，TAIFG 置位。图 8-8 表示了标志位的置位状况。

图 8-8. 增减模式下的标志位



### 改变 TACCR0 周期寄存器

当定时器运行时，改变 TACCR0 的值，如果正处于减计数的情况，定时器会继续减到 0，新的周期在减到 0 后开始；如果正处于增计数状态，新周期大于等于原来的周期，或比当前计数值要大，定时器会增计数到新的周期；如果正处于增计数状态，新周期小于原来的周期，定时器立刻开始减计数，但是，在定时器开始减计数之前会多计一个数。

### 增减模式的使用

增减模式支持在输出信号之间有死区时间的应用（参阅 TA 输出章节）。例如，避免出现过载情况，2 个输出驱动一个 H 桥不能同时为高。在图 8-9 的例中， $t_{dead}$  为

$$t_{dead} = t_{timer} \times (TACCR1 - TACCR2)$$

$t_{dead}$ ——同时输出时必须没有反应的时间段

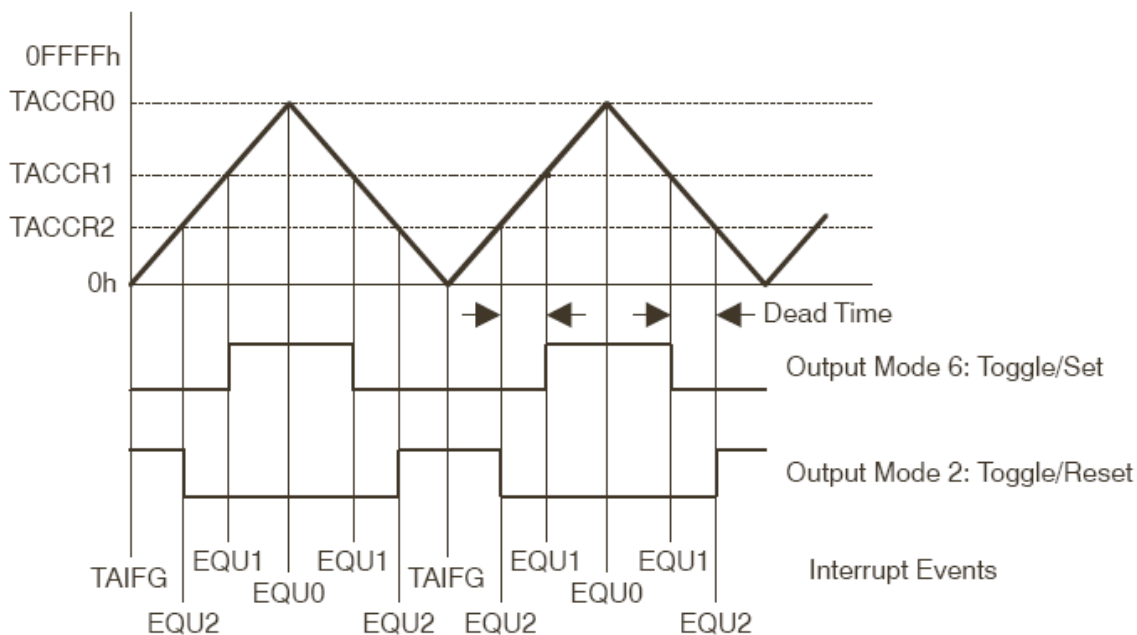


timer——定时器时钟周期

TACCRx——捕获比较寄存器 x 的内容

TACCRx 寄存器并不是缓冲，写入时立即更新，因此，任何所要求的死区时间不会自动保留。

图 8-9. 增减模式的输出



## 8.2.4 捕获比较模块

定时器 A 中有 2 个或 3 个相同的捕获比较模块 TACCRx，其中的任何一个模块可以用于定时器数据的捕获或产生时间间隔。

### 捕获模式

当 CAP=1 时，选择捕获模式。捕获模式用于记录时间事件，比如速度估计或时间测量。捕获输入 CCIxA 和 CCIxB 连接外部的引脚或内部的信号，这通过 CCISx 位来选择。CMx 位选择捕获输入信号触发沿：上升沿、下降沿或 2 者都捕获。捕获事件发生于所选择的输入信号的触发沿。如果发生了捕获事件：

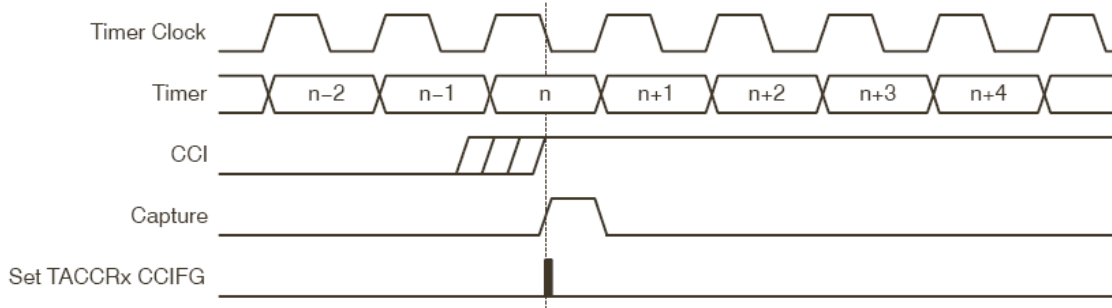
定时器的值复制到 TACCRx 寄存器中

### 中断标志位 CCIFG 置位

在任何时刻，可以通过 CCI 位读取输入信号的电平。MSP430x2xx 系列的器件允许 CCIxA 和 CCIxB 连接于不同的信号（请参考器件手册）。捕获信号可能会和定时器时钟不同步，并导致竞争条件的发生。将 SCS 位置位可以在下个定时器时钟使捕获同步，见图例 8-10

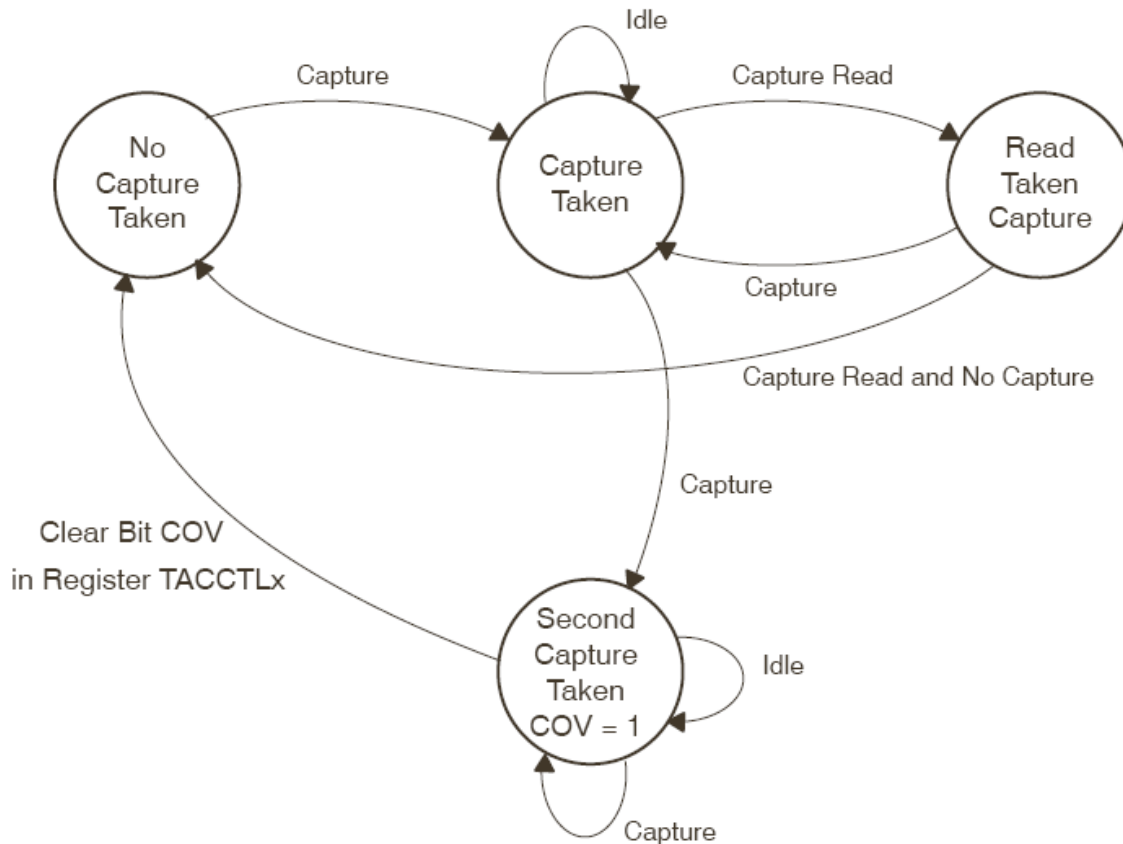


图 8-10. 捕获信号 (SCS=1)



如果一个第二次捕获在第一次捕获的值被读取之前发生，捕获比较寄存器就会产生一个溢出逻辑，COV 位在此时置位，如图 8-11，COV 位必须软件清除。

图 8-11. 捕获循环



通过软件初始化捕获

捕获可以由软件初始化。CMx 位可以配置捕获的触发沿。CCIS1=1 和 CCIS0 位可以捕获电压在 VCC 和 GND 之间的信号，初始化捕获器举例

```
MOV #CAP+SCS+CCIS1+CM_3,&TACCTLx ; 配置 TACCTLx
```

XOR #CCIS0,&TACCTLx ; TACCTLx = TAR

### 比较模式

比较模式通过将 CAP=0 来进入。比较模式用于选择 PWM 输出信号或在特定的时间间隔中断。当 TAR 计数到 TACCRx 的值时：

- I 中断标志 CCIFG=1;
- I 内部信号 EQUx=1;
- I EQUx 根据输出模式来影响输出信号
- I 输入信号 CCI 锁存到 SCCI

### 8.2.5 输出单元

每个捕获比较模块包含一个输出单元。输出单元用于产生如 PWM 这样的信号。每个输出单元可以根据 EQU0 和 EQUx 产生 8 种模式的信号。

#### 输出模式

输出模式由 OUTMODx 位来确定，如表 8-2。对于所有模式来说（出来模式 0），OUTx 信号随着定时器时钟的上升沿而改变。输出模式 2, 3, 6 和 7 对输出单元 0 无效，因为在这些模式下，EQUx = EQU0。

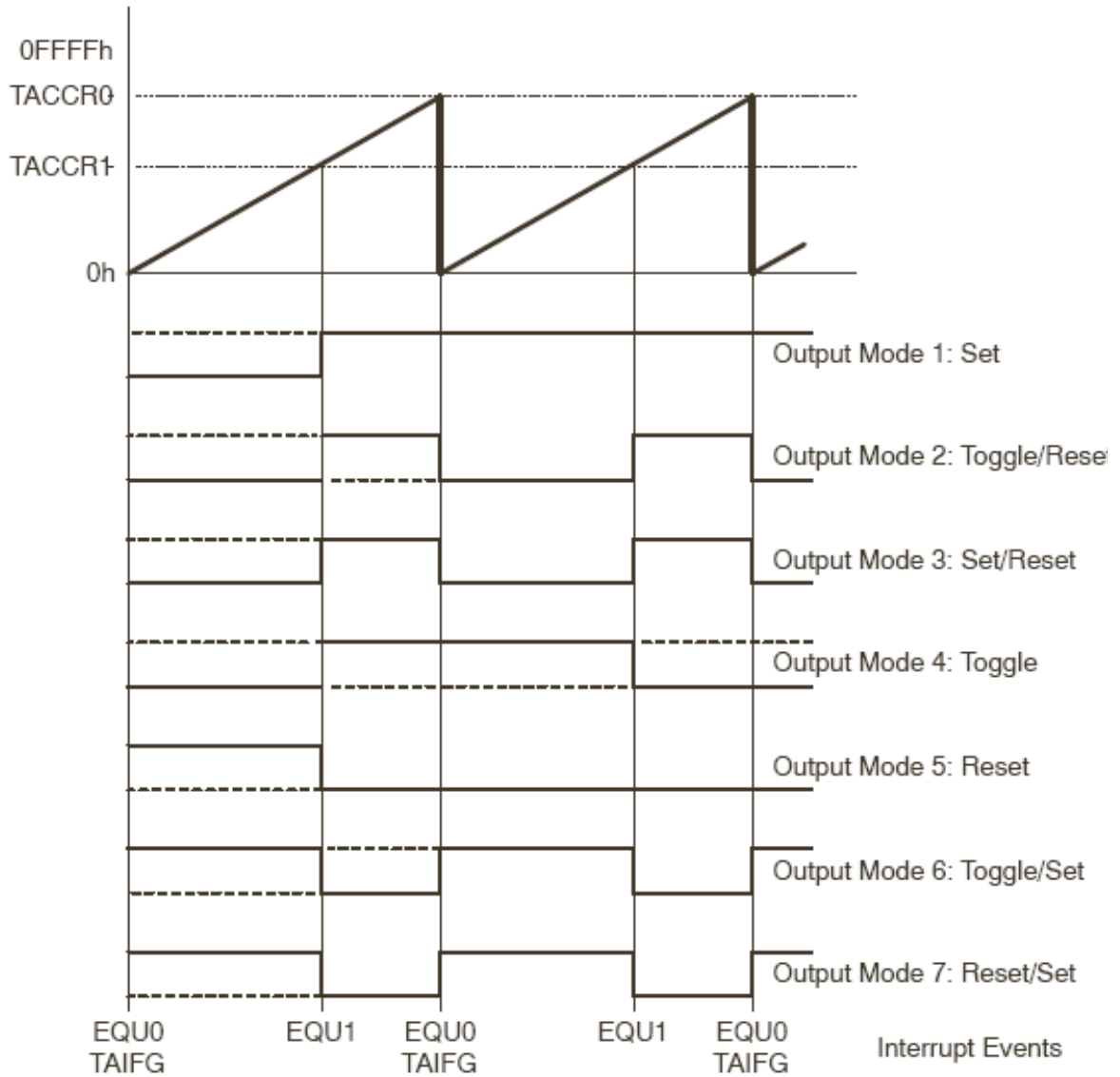
表 8-2.输出模式

OUTMODx	模式	说明
000	输出	输出信号 OUTx 由 OUTx 位定义。当 OUTx 位更新时，OUTx 信号立刻更新
001	置位	当定时器计数到 TACCRx 值时，输出置位，并保持置位直到定时器复位或选择了另一个输出模式。
010	翻转/复位	当定时器计数到 TACCRx 值时，输出翻转。当定时器计数到 TACCR0 值时，输出复位。
011	置位/复位	当定时器计数到 TACCRx 值时，输出置位。当定时器计数到 TACCR0 值时，输出复位。
100	翻转	当定时器计数到 TACCRx 值时，输出翻转。输出信号的周期是定时器周期的 2 倍。
101	复位	当定时器计数到 TACCRx 值时，输出复位，并保持复位直到选择了另一个输出模式。
110	翻转/置位	当定时器计数到 TACCRx 值时，输出翻转。当定时器计数到 TACCR0 值时，输出置位。
111	复位/置位	当定时器计数到 TACCRx 值时，输出复位。当定时器计数到 TACCR0 值时，输出置位。

#### 输出举例——定时器处于增模式

当定时器计数到 TACCRx 的值或从 TACCR0 到 0 时, OUTx 信号根据输出模式而改变。如图 8-12 使用了 TACCR0 和 TACCR1

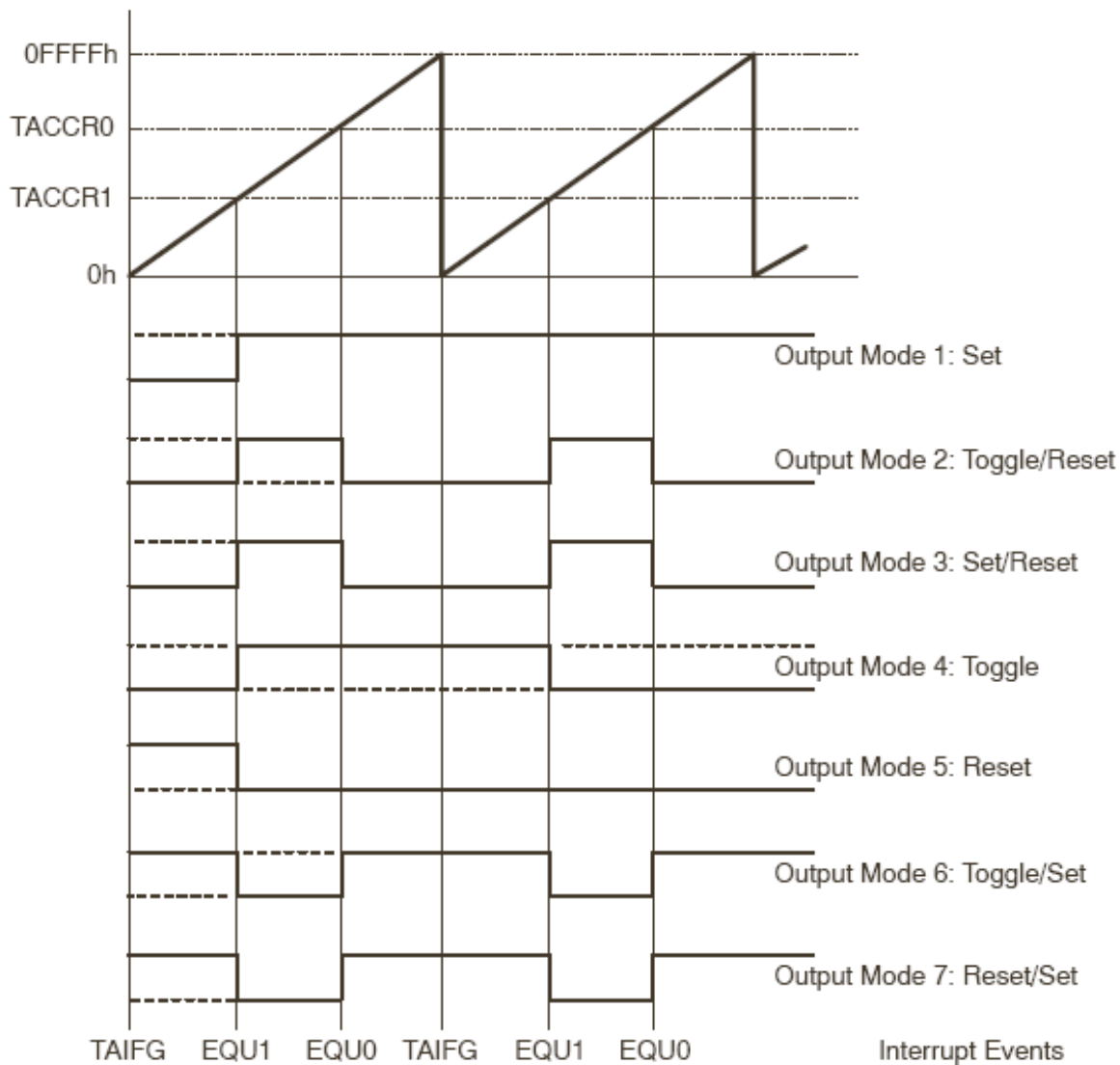
图 8-12. 输出举例——定时器处于增模式



输出举例——定时器处于连续模式

当定时器计数到 TACCRx 和 TACCR0 时, OUTx 信号按选择的输出模式发生改变。如图 8-13 所示

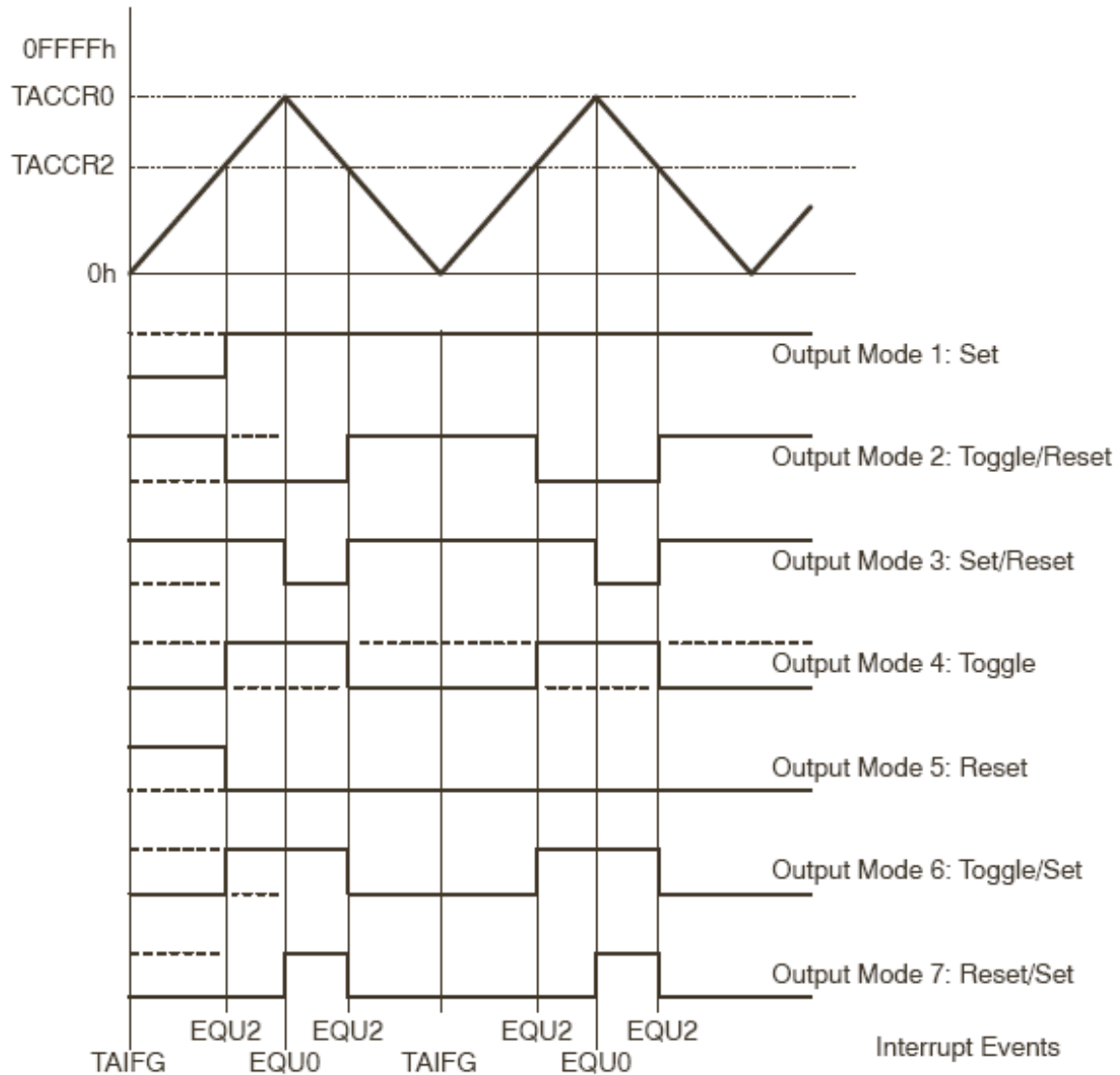
图 8-13. 输出举例——定时器处于连续模式



### 输出举例——定时器处于增减模式

当定时器的值在任一计数方向上出现了等于 TACCR<sub>x</sub> 和等于 TACCR0 的值时，OUT<sub>x</sub> 信号按选择的输出模式发生改变，如图 8-14

图 8-14. 输出举例——定时器处于增减模式



### 注意：输出模式的切换

当需要在输出模式之间进行切换时，`OUTMODx` 的一个位必须在过度时保持置位，除非是切换到模式 0，否则会由于或非门解码输出模式 0 而导致出现脉冲干扰。输出模式之间的安全切换的方法之一是用输出模式 7 作为过度状态，例如：

`BIS #OUTMOD_7,&TACCTLx`；设置为输出模式 7

`BIC #OUTMODx,&TACCTLx`；清除不需要的位

### 8.2.6 定时器 A 的中断

16 位定时器 A 有 2 个中断向量：

I TACCR0 CCIFG 的 TACCR0 中断向量

## I 所有其他 CCIFG 和 TAIFG 的 TAIV 中断向量

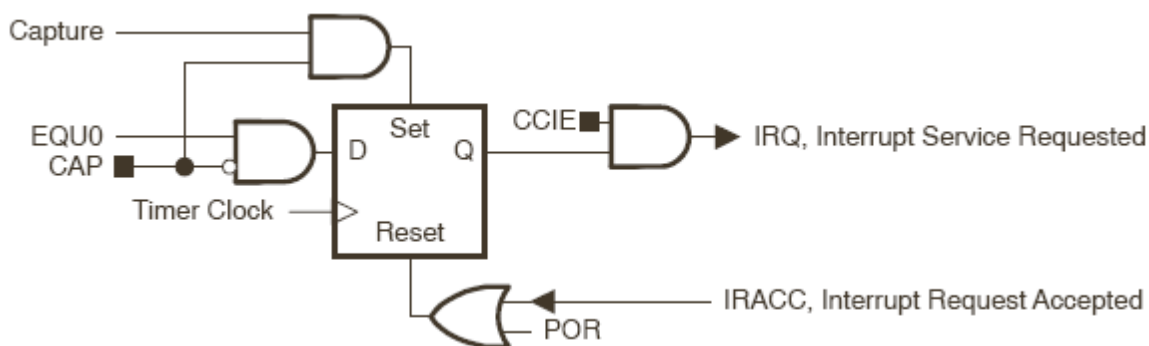
在捕获模式下，当一个定时器的值捕获到相应的 TACCRx 寄存器时，CCIFG 标志置位。在比较模式下，如果 TAR 计数到相应的 TACCRx 值时，CCIFG 标志置位。软件可以清除或置位任何一个 CCIFG 标志。当响应的 CCIE 和 GIE 置位时，CCIFG 标志就会产生一个中断。

### TACCR0 中断

TACCR0 CCIFG 标志拥有定时器 A 的最高中断优先级，并有一个专用的中断向量，如图 8-15。当进入 TACCR0 中断后，TACCR0 CCIFG 标志自动复位。

如图 8-15。

图 8-15 捕获比较 TACCR0 中断标志



### TAIV, 中断向量发生器

TACCR1 CCIFG, TACCR2 CCIFG, 和 TAIFG 标志共用一个中断向量。中断向量寄存器 TAIV 用于确定它们中的哪个要求响应中断。最高优先级的中断在 TAIV 寄存器中产生一个数字（见寄存器说明），这个数字是规定的数字，可以在程序中识别并自动进入相应的子程序。禁止定时器 A 中断不会影响 TAIV 的值。

对 TAIV 的读写会自动复位最高优先级的挂起中断标志。如果另一个中断标志置位，在结束原先的中断响应后会，该中断响应立即发生。例如，当中断服务子程序访问 TAIV 时，如果 TACCR1 和 TACCR2 CCIFG 标志位置位，TACCR1 CCIFG 自动复位。在中断服务子程序的 RETI 命令执行后，TACCR2 CCIFG 标志会产生另一个中断。

### TAIV 软件示例

以下软件说明了 TAIV 的使用和操作。TAIV 的值加入 PC 指针来自动跳转到相应的子程序。右边空白处的数字表明 CPU 每条指令需要的周期。不同的中断源的软件包含中断响应时间和返回中断周期，但并不包含任务本身的执行时间。响应时间定义为：

- I 捕获比较模块 TACCR0 为 11 个时钟周期
- I 捕获比较模块 TACCR1/8 为 16 个时钟周期

## I 定时器溢出标志 TAIFG 置位为 14 个时钟周期

; TACCR0 CCIFG 的中断处理		周期数
CCIFG_0_HND		
; ... ; 中断响应开始		6
RETI		5
; TAIFG, TACCR1 和 TACCR2 CCIFG 的中断处理		
TA_HND ...	; 中断响应	6
ADD &TAIV,PC	; 加偏移量跳转到标号 table_3	3
RETI	; Vector 0: 无中断	5
JMP CCIFG_1_HND	; 中断 2: TACCR1	2
JMP CCIFG_2_HND	; 中断 4: TACCR2	2
RETI	; 中断 6: 保留	5
RETI	; 中断 8: 保留	5
TAIFG_HND	; 中断 10: TAIFG Flag	
... ; 开始任务		
RETI		5
CCIFG_2_HND	; 中断 4: TACCR2	
... ; 开始任务		
RETI ; 返回主程序		5
CCIFG_1_HND	; 中断 2: TACCR1	
... ; 开始任务		
RETI	; 返回主程序	5

## 8.3 定时器 A 寄存器

定时器 A 寄存器如表 8-1: († MSP430x20xx 中不存在)

表 8-3. 定时器 A 寄存器

寄存器	简写	寄存器类型	地址	初始状态
TA 控制寄存器	TACTL	读/写	0160h	POR 复位
TA 计数器	TAR	读/写	0170h	POR 复位
TA 捕获比较控制寄存器 0	TACCTL0	读/写	0162h	POR 复位
TA 捕获比较 0	TACCR0	读/写	0172h	POR 复位
TA 捕获比较控制寄存器 1	TACCTL1	读/写	0164h	POR 复位
TA 捕获比较 1	TACCR1	读/写	0174h	POR 复位
TA 捕获比较控制寄存器 2	TACCTL2†	读/写	0166h	POR 复位
TA 捕获比较 2	TACCR2†	读/写	0176h	POR 复位
TA 中断向量寄存器	TAIV	只读	012Eh	POR 复位

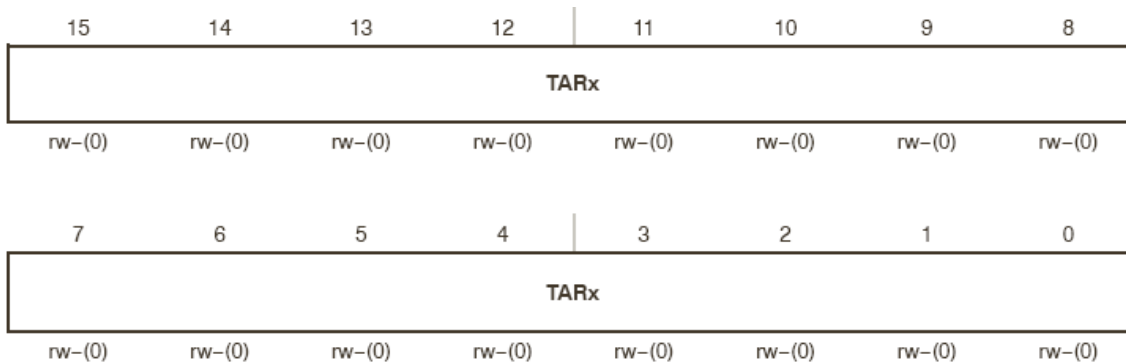
## TACTL, Timer\_A 控制寄存器





未使用	15-10	未使用位
<b>TASSELx</b>	9-8	TA 时钟源选择 00 TACLK 01 ACLK 10 SMCLK 11 INCLK
<b>IDx</b>	7-6	输入分频。这些位为输入时钟分频选择 00 /1 01 /2 10 /4 11 /8
<b>MCx</b>	5-4	模式控制，当 TA 不用于节省功耗时，将 MCx=00h 00 停止模式:定时器停止 01 增模式:定时器计数到 TACCR0 10 连续模式: 定时器计数到 0FFFFh 11 增减模式: 定时器计数到 TACCR0 然后减到 0000h
<b>Unused</b>	3	未使用位
<b>TACLR</b>	2	定时器清零位。该位置位会复位 TAR，时钟分频和计数方向。TACLR 位会自动复位并读出值为 0
<b>TAIE</b>	1	TA 中断允许。改位允许 TAIFG 中断请求 0 中断禁止 1 中断允许
<b>TAIFG</b>	0	TA 中断标志位 0 无中断挂起 1 中断挂起

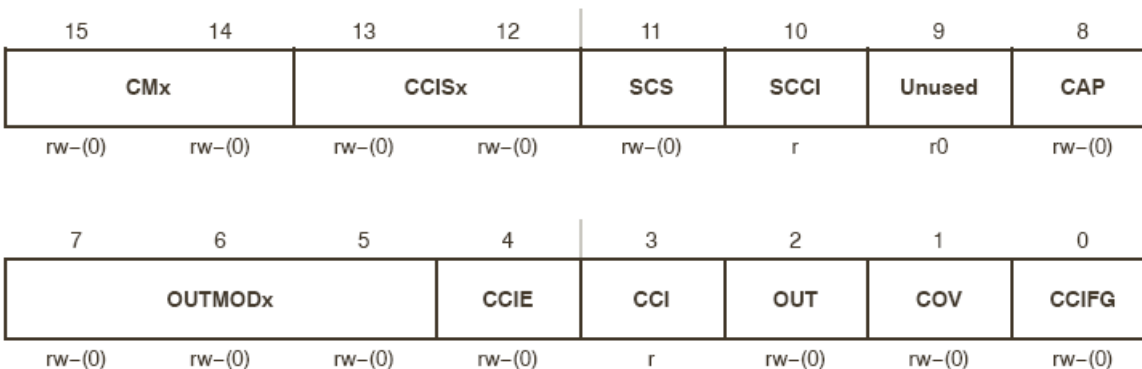
### TAR, Timer\_A 寄存器



TARx 位 15-0

Timer\_A 寄存器. TAR 寄存器是 Timer\_A 的计数器。

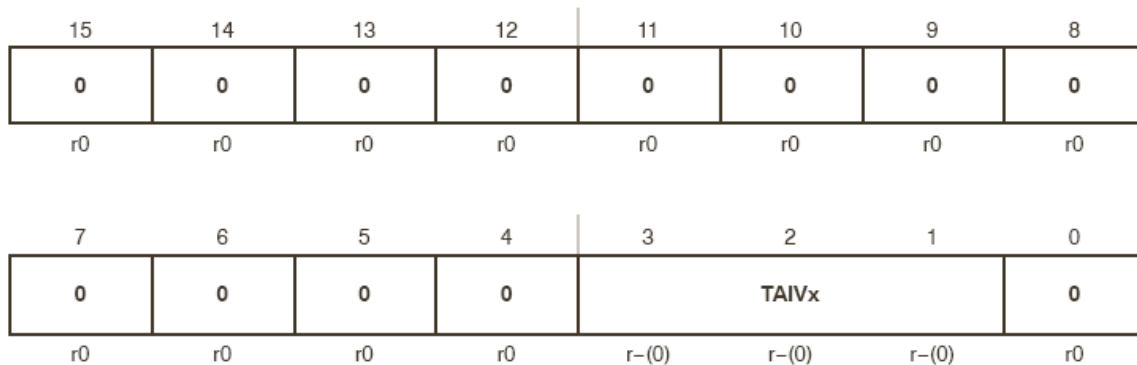
**TACCTLx, 捕获比较控制寄存器**



<b>CMx</b>	15-14	捕获模式 00 不捕获 01 上升沿捕获 10 下降沿捕获 11 上升和下降沿都捕获
<b>CCISx</b>	13-12	捕获比较选择, 该位选择 TACCRx 的输入信号, 详见器件手册 00 CCIxA 01 CCIxB 10 GND 11 VCC
<b>SCS</b>	11	同步捕获源, 该位用于将捕获通信和时钟同步 0 异步捕获 1 同步捕获
<b>SCCI</b>	10	同步的捕获/比较输入, 所选择的 CCI 输入信号由 EQUx 信号锁存, 并可通过该位读取
<b>Unused</b>	9	未使用
<b>CAP</b>	8	捕获模式

		0 比较模式 1 捕获模式
<b>OUTMODx</b>	7-5	输出模式位。由于在模式 2, 3, 6 和 7 下 EQUx = EQU0, 因此这些模式对 TACCR0 无效 000 OUT 位的值 001 置位 010 翻转/复位 011 置位/复位 100 翻转 101 复位 110 翻转/置位 111 复位/置位
<b>CCIE</b>	4	捕获比较中断允许位, 该位允许相应的 CCIFG 标志中断请求 0 中断禁止 1 中断允许
<b>CCI</b>	3	捕获比较输入。所选择的输入信号可以通过该位读取
<b>OUT</b>	2	对于输出模式 0, 该位直接控制输出状态 0 输出低电平 1 输出高电平
<b>COV</b>	1	捕获溢出位。该位表示一个捕获溢出发生。COV 必须由软件复位。 0 没有捕获溢出发生 1 有捕获溢出发生
<b>CCIFG</b>	0	捕获比较中断标志位 0 没有中断挂起 1 有中断挂起

**TAIV, Timer\_A 中断向量寄存器**



TAIVx Bits 15-0 Timer\_A 中断向量值

TAIV 的内容	中断源	中断标志	中断优先级
<b>00h</b>	无中断挂起	—	
<b>02h</b>	捕获比较 1	TACCR1 CCIFG	最高
<b>04h</b>	捕获比较 2 (MSP430x20xx 中没有)	TACCR2 CCIFG	

06h	保留	—	
08h	保留	—	
0Ah	定时器溢出	TAIFG	
0Ch	保留	—	
0Eh	保留	—	最低



MSP430F22x4 开发板

专业提供 **MSP430** 单片机开发工具