# PWM Review

Last updated 11/5/18

# PWM Review

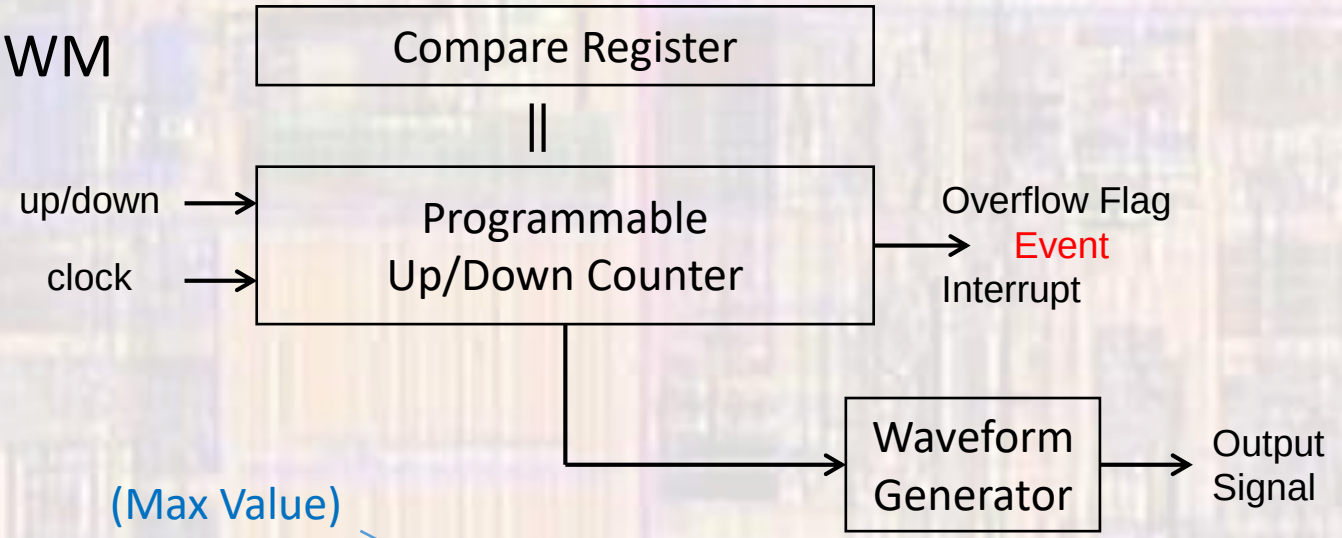
- Advanced Timer Functions
  - Output Compare
    - Sets a flag and/or creates an interrupt when the counter value matches a value programmed into a separate register
  - Input Capture
    - Captures the counter value when a trigger event occurs
    - Sets a flag and/or creates an interrupt
  - Pulse Width Modulation (PWM)
    - Creates an automated PWM signal

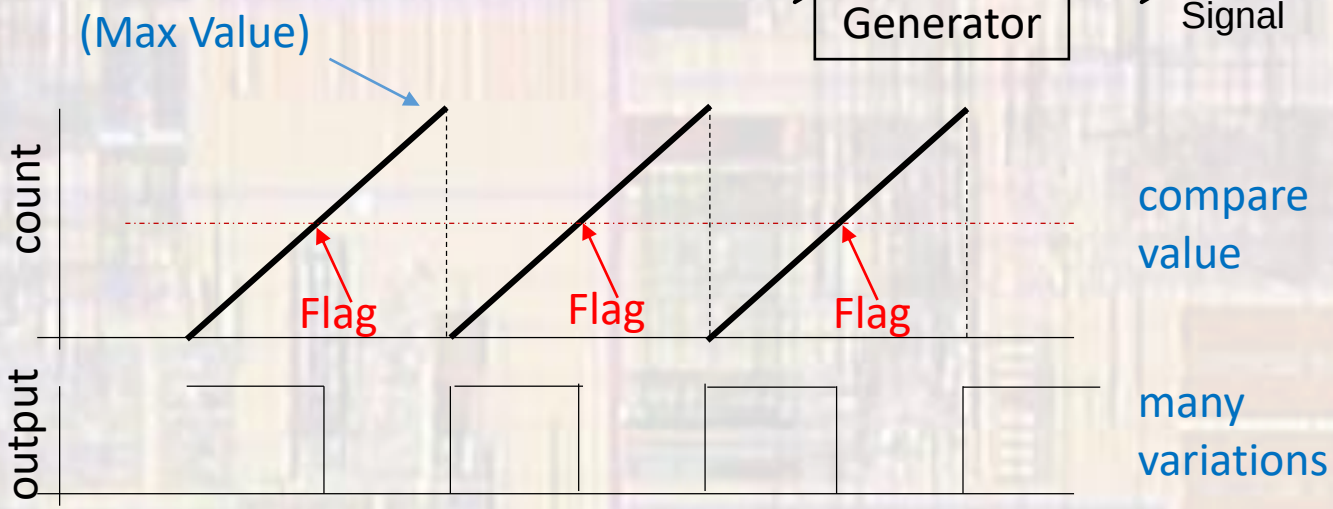
# PWM Review

- Advanced Timer Function
  - Pulse Width Modulation (PWM)
    - Create an output waveform based on two counter values
    - Many variations
  - Delay Counter – in Periodic Mode
    - Count up or down - to or from – the Max Value
  - Compare Register(s)
    - Holds the value to compare with the counter value
    - When matched – create a flag/interrupt

# PWM Review

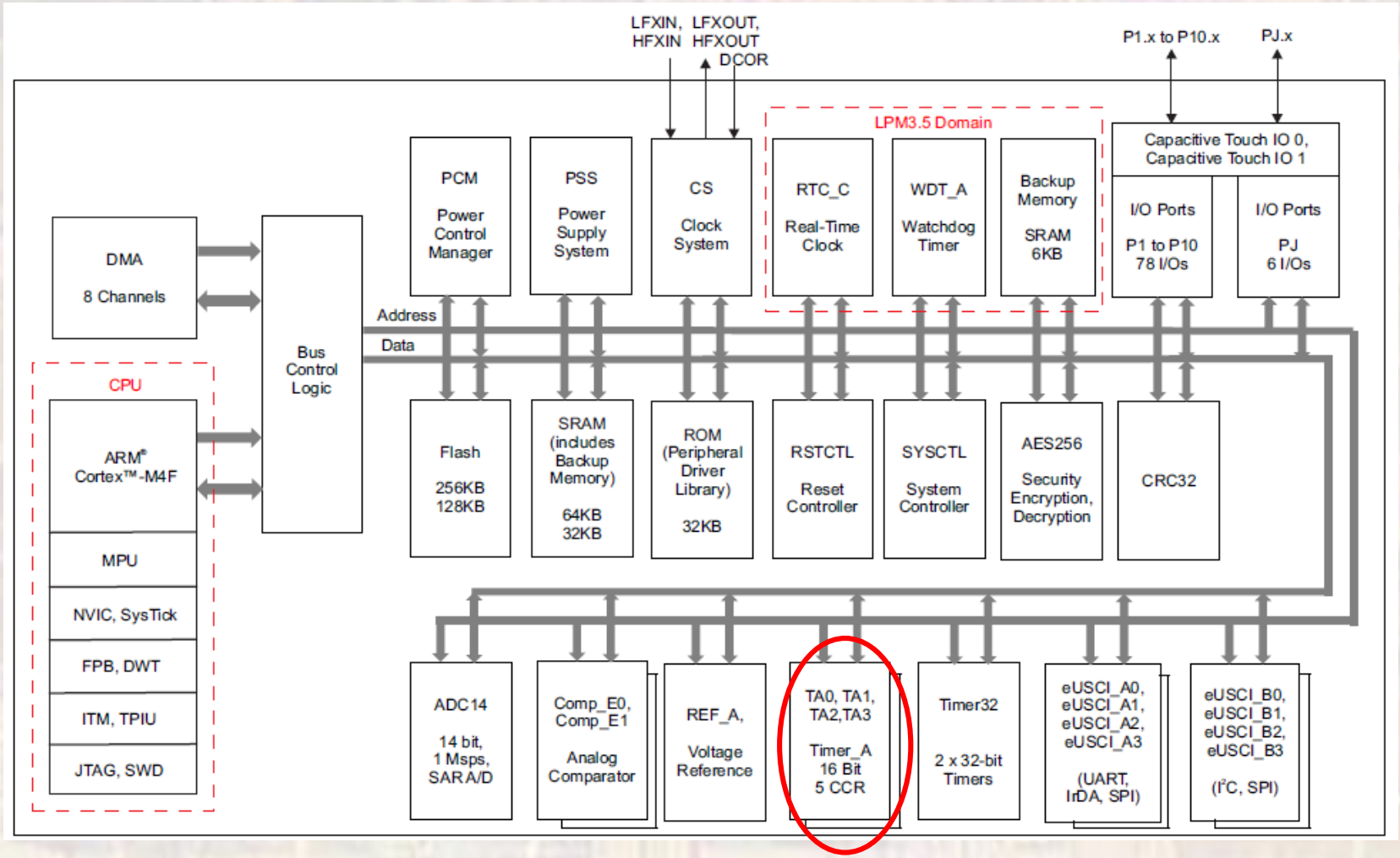
- Advanced Timer Function
  - PWM



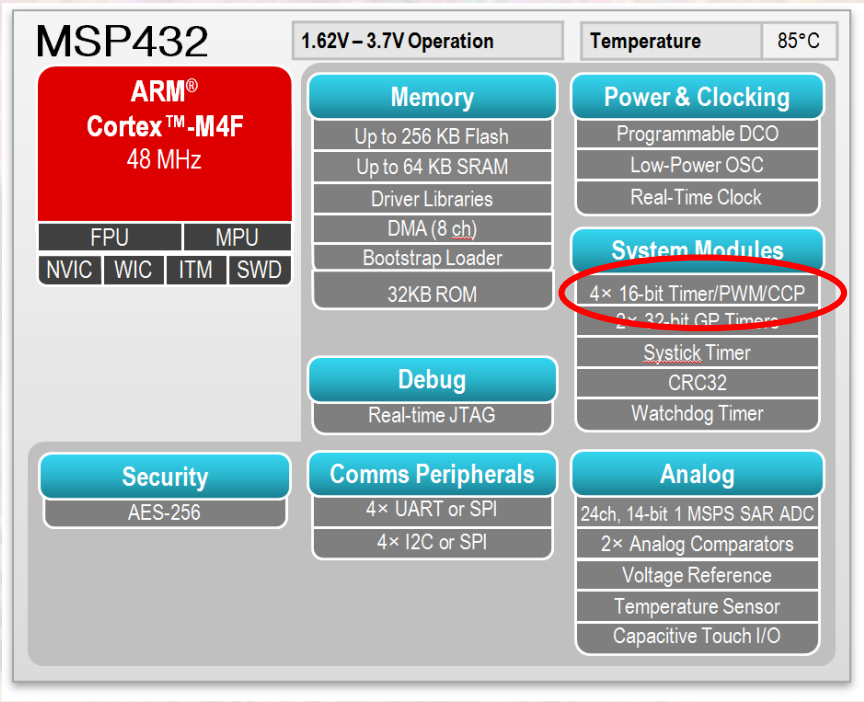
# PWM Review

- MSP432 Timer A



# PWM Review

- MSP432 Timer A
  - ARM (AMBA Compliant) timer
  - 4 – 16 bit timers
  - 3 Timer Modes
    - Up
    - Continuous
    - Up/Down
  - 4 Capture/Compare Modes
    - Timer
    - Capture
    - Compare
    - PWM

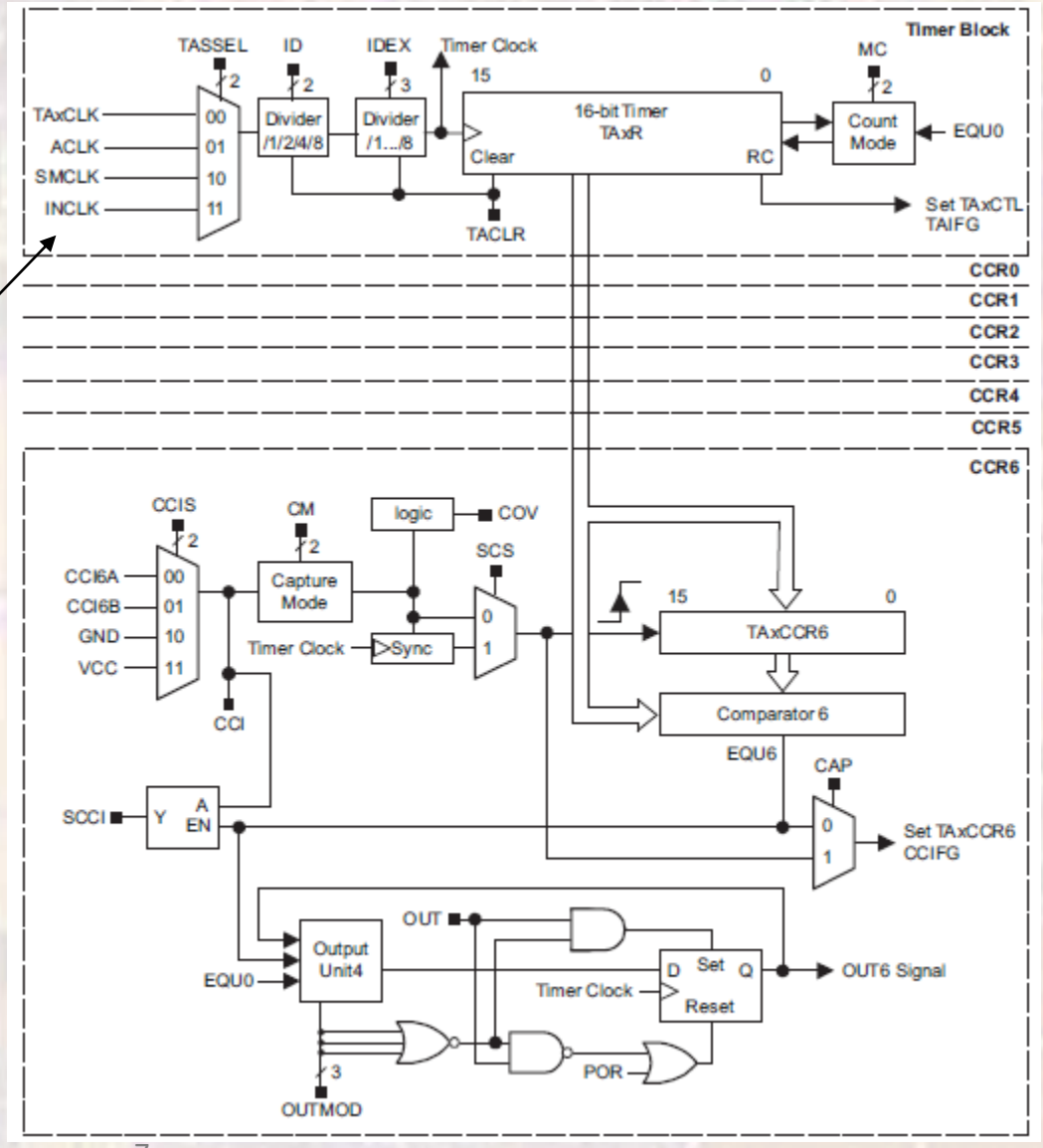


# PWM Review

- MSP432 Timer A
  - 4 Timers - each with 5 capture/compare blocks

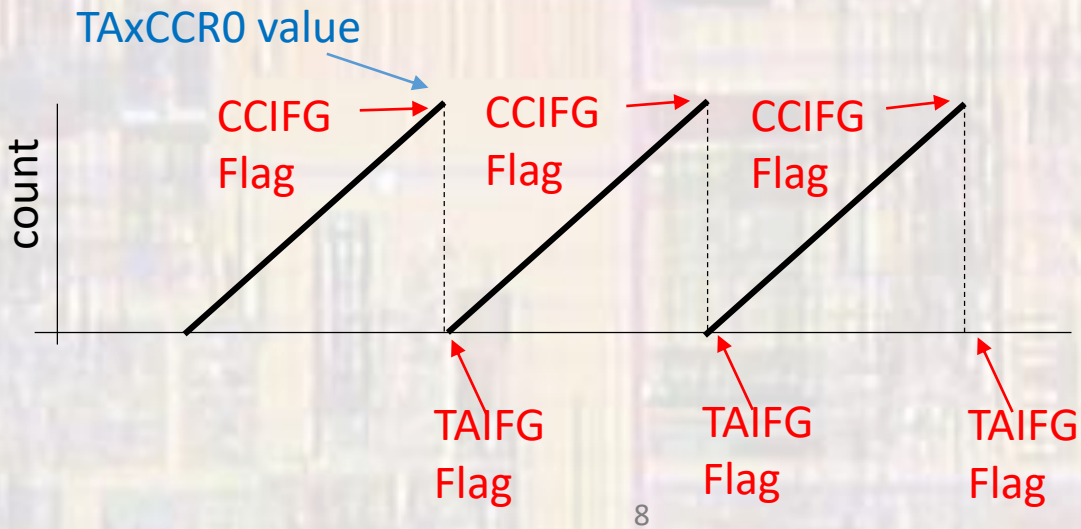
Timer

1 of 5 Capture/Compare Blocks



# Timers

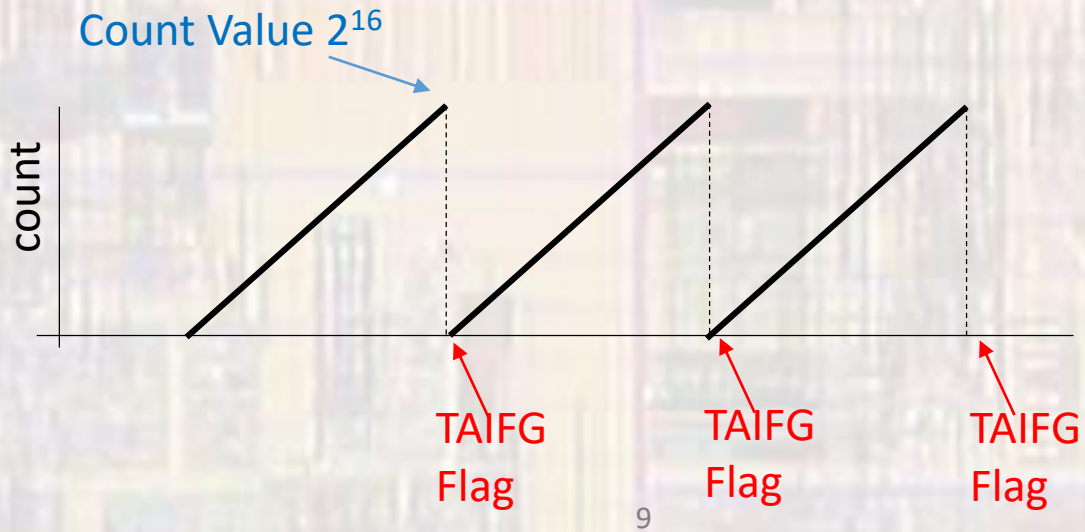
- MSP432 Timer A
  - Timer - Up mode
    - Max count value is $2^{16}$
    - TAxCCR0 register – holds the upper limit for the count
    - 2 Flags
      - CCIFG – set when counter reaches TAxCCR0value
      - TAIFG – set when the counter wraps around to 0




1 clock separation between the two flags

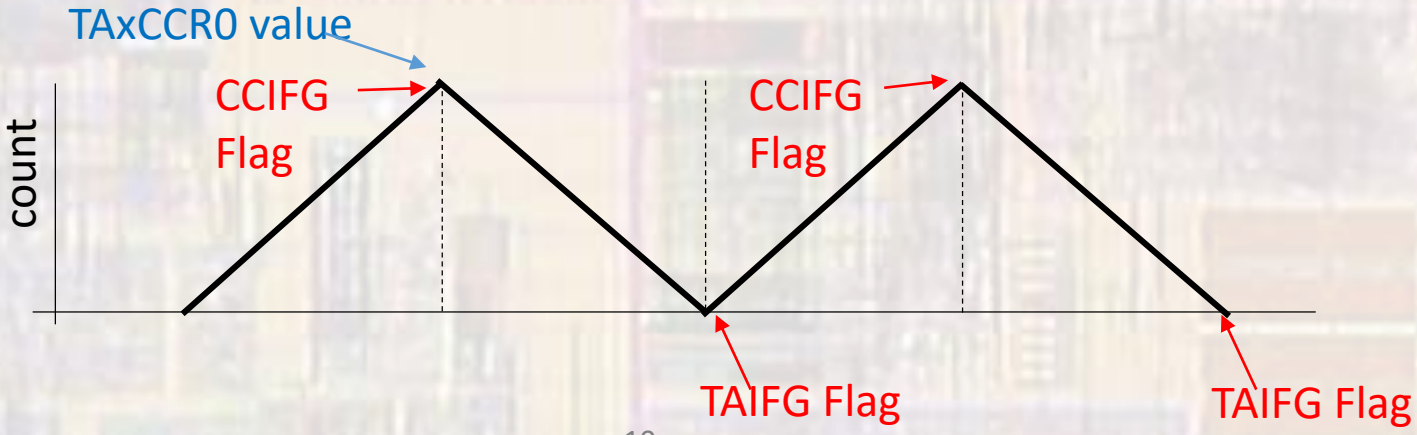
# PWM Review

- MSP432 Timer A
  - Timer - Continuous mode
    - Count value is $2^{16}$
    - TAIFG – set when the counter wraps around to 0



# PWM Review

- MSP432 Timer A
  - Timer - Up/Down mode
    - Max count value is $2^{16}$
    - Allows for 2x period
    - TAxCCR0 register – holds the upper limit for the count
    - 2 Flags
      - CCIFG – set when counter reaches TAxCCR0value
      - TAIFG – set when the counter reaches 0



# PWM Review

- MSP432 Timer A
  - Capture/Compare Block – Compare Mode/PWM Mode
    - 5 Compare blocks in each timer
    - Desired Compare values stored in TAxCCRn (n = 0:6)
    - CCIFG flag is set
    - Internal signal EQUn is set – modifies output modes
    - Input signal CCI is latched into SCCI
    - Used to create periodic interrupts or PWM signals

# PWM Review

- MSP432 Timer A
  - Timer Output Block
    - Automatically create output signals
    - 8 modes – set by OUTMODx bits
    - Outputs based on EQU0 and EQUn signals (relative compare values)

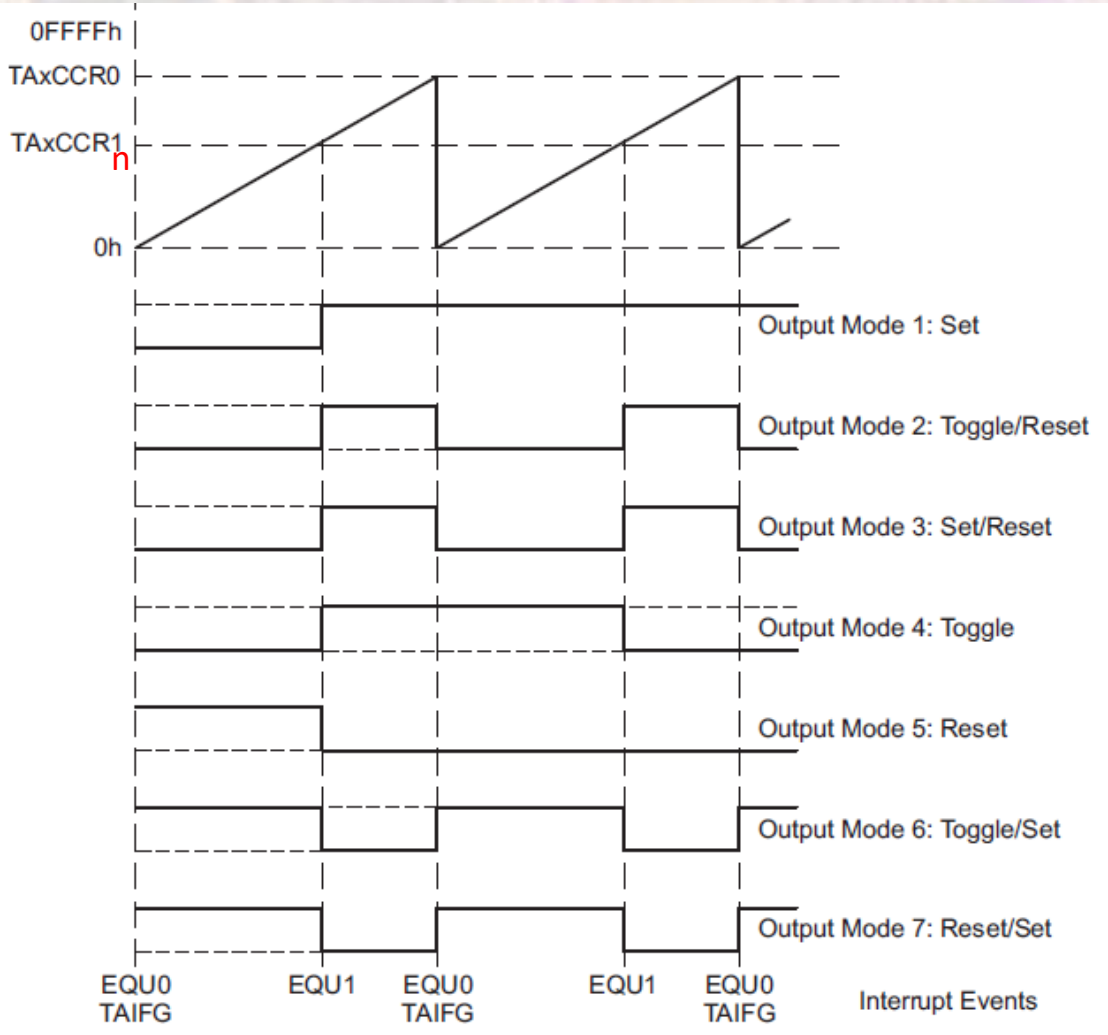
**Table 17-2. Output Modes**

| OUTMODx | Mode | Description |
|---|---|---|
| 000 | Output | The output signal OUTn is defined by the OUT bit. The OUTn signal updates immediately when OUT is updated. |
| 001 | Set | The output is set when the timer *counts* to the TAxCCRn value. It remains set until a reset of the timer, or until another output mode is selected and affects the output. |
| 010 | Toggle/Reset | The output is toggled when the timer *counts* to the TAxCCRn value. It is reset when the timer *counts* to the TAxCCR0 value. |
| 011 | Set/Reset | The output is set when the timer *counts* to the TAxCCRn value. It is reset when the timer *counts* to the TAxCCR0 value. |
| 100 | Toggle | The output is toggled when the timer *counts* to the TAxCCRn value. The output period is double the timer period. |
| 101 | Reset | The output is reset when the timer *counts* to the TAxCCRn value. It remains reset until another output mode is selected and affects the output. |
| 110 | Toggle/Set | The output is toggled when the timer *counts* to the TAxCCRn value. It is set when the timer *counts* to the TAxCCR0 value. |
| 111 | Reset/Set | The output is reset when the timer *counts* to the TAxCCRn value. It is set when the timer *counts* to the TAxCCR0 value. |

# PWM Review

- MSP432 Timer A

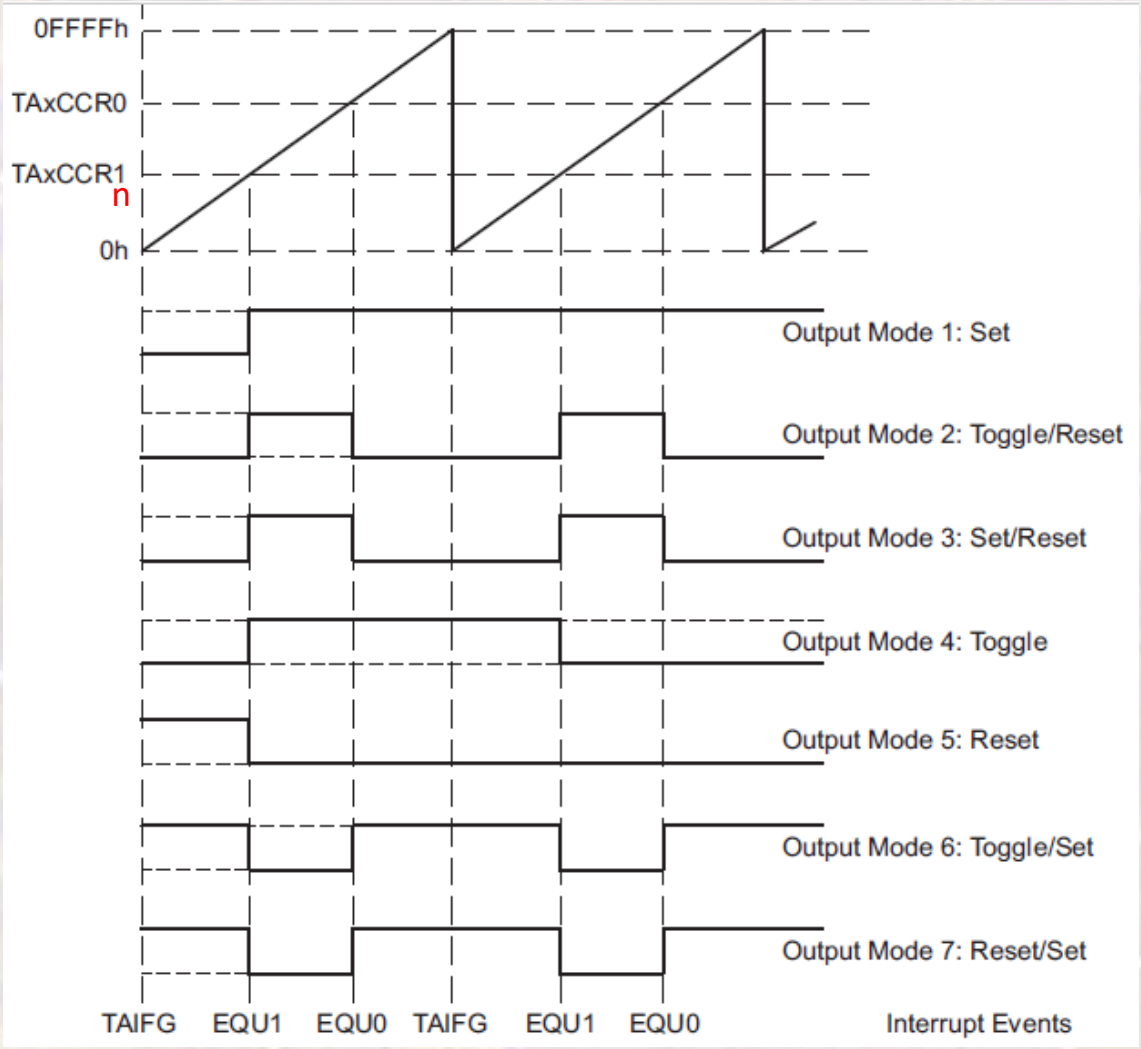
Timer in UP mode



# PWM Review

- MSP432 Timer A

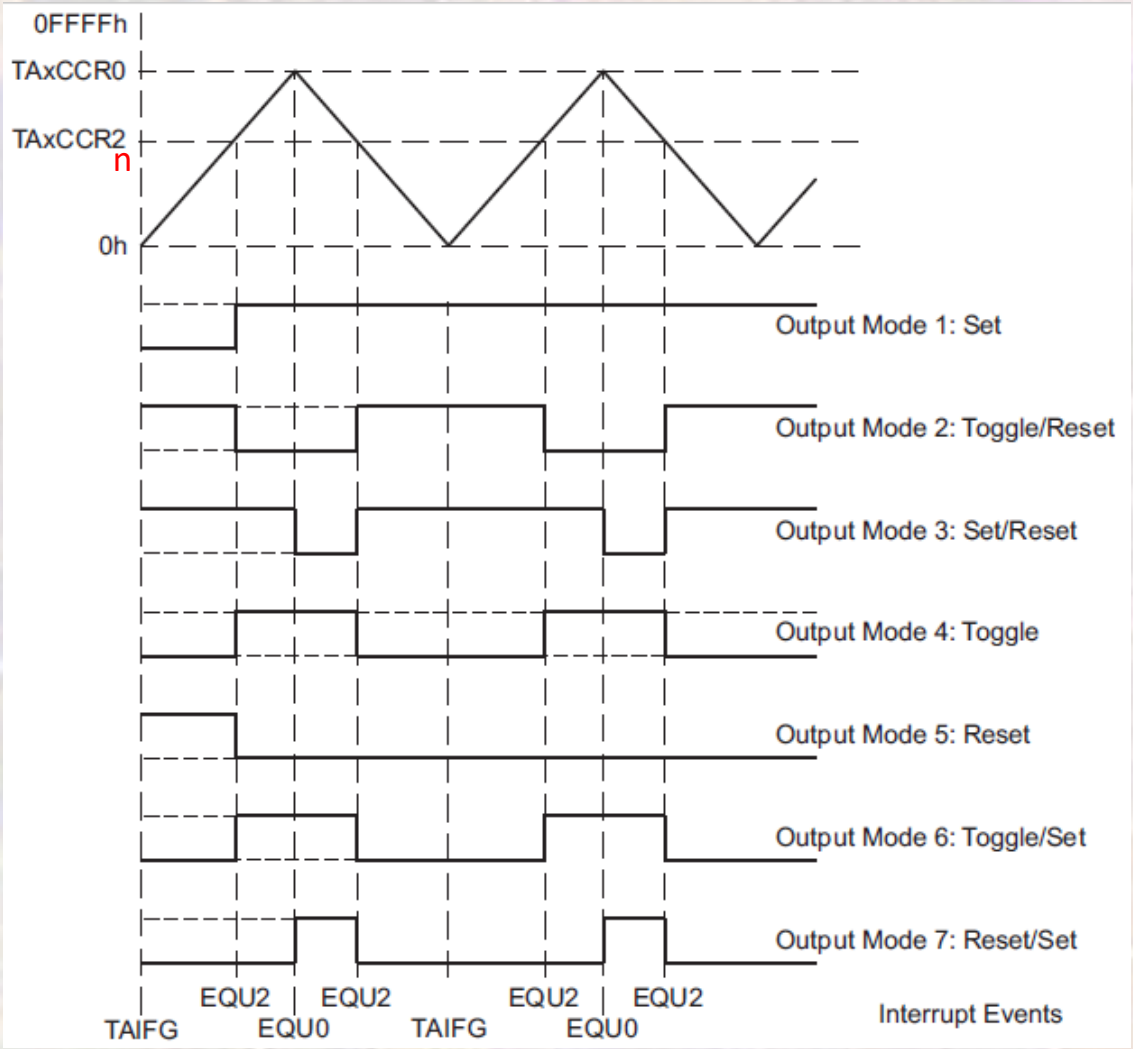
Timer in Continuous mode



# PWM Review

- MSP432 Timer A

Timer in Up/Down mode



# PWM Review

- MSP432 Timer A
  - Interrupts
    - 2 interrupt vectors
    - TAxCCR0 vector
      - Used for the CCIFG flag for TAxCCR0 only
    - TAxIV vector
      - CCIFG flag set in capture mode on TAxCCRn
      - CCIFG flag set in compare mode on TAxCCRn
      - TAIFG flag

# PWM Review

- MSP432 Timer A
  - Interrupts
    - TAxIV vector generator
      - CCIFG and TAIFG flags are combined into a single interrupt vector
      - The highest priority enabled interrupt generator is stored in the TAxIV register
        - Can be evaluated or added to PC to execute the ISR
      - Reading TAxIV clears the current interrupt flag and activates the next highest pending interrupt

# PWM Review

- MSP432 Timer A
  - Calculation examples
    - Timer in UP mode
    - Tclk = 12MHz
    - Desire a 1KHz 50% duty cycle square wave

    12MHz clk → 83.33ns/clk
    1KHz → 1ms period
    →12,000 12MHz clks / 1KHz period
    for a square wave – 6000 12MHz clks low, 6000 12MHz clks high
    12,000 clks is well within our 65,536 maximum limit

# PWM Review

- MSP432 Timer A
  - Calculation examples

    for a square wave – 6000 12MHz clks low, 6000 12MHz clks high

    Choose set/reset mode

    Max = 12,000 clks

    Target = 6000 clks

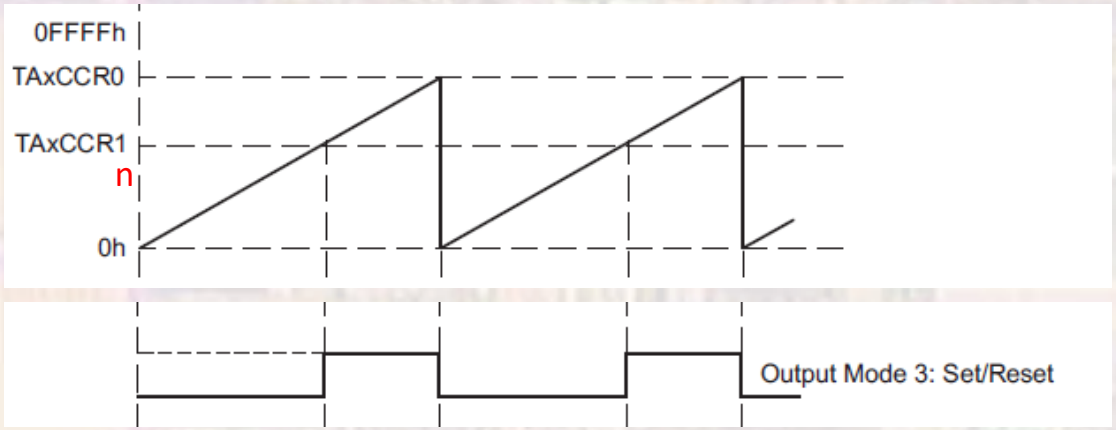
    Up mode

    Fcounter = 12MHz

    TA0CCR0 = 12000

    TA0CCR1 = 6000

    → 1KHz Square wave



# PWM Review

- MSP432 Timer A
  - Calculation examples
    - Timer in UP mode
    - Tclk = 12MHz
    - Desire a 1KHz 25% duty cycle square wave

Solution

# PWM Review

- MSP432 Timer A
  - Calculation examples
    - Timer in UP mode
    - Tclk = 12MHz
    - Desire a 100Hz 75% duty cycle square wave

Solution

# PWM Review

- MSP432 Timer A
  - Timer A Registers

**Table 17-3. Timer_A Registers**

| Offset | Acronym | Register Name | Section |
|---|---|---|---|
| 00h | TAxCTL | Timer_Ax Control | Section 17.3.1 |
| 02h to 0Eh | TAxCCTL0 to TAxCCTL6 | Timer_Ax Capture/Compare Control 0 to Timer_Ax Capture/Compare Control 6 | Section 17.3.3 |
| 10h | TAxR | Timer_Ax Counter | Section 17.3.2 |
| 12h to 1Eh | TAxCCR0 to TAxCCR6 | Timer_Ax Capture/Compare 0 to Timer_Ax Capture/Compare 6 | Section 17.3.4 |
| 2Eh | TAxIV | Timer_Ax Interrupt Vector | Section 17.3.5 |
| 20h | TAxEX0 | Timer_Ax Expansion 0 | Section 17.3.6 |

# PWM Review

- MSP432 Timer A
  - Control Register

**Figure 17-15. TAxCTL Register**

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| Reserved | | | | | | TASSEL | |
| rw-0 | rw-0 | rw-0 | rw-0 | rw-0 | rw-0 | rw-0 | rw-0 |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| ID | | MC | | Reserved | TACLR | TAIE | TAIFG |
| rw-0 | rw-0 | rw-0 | rw-0 | rw-0 | w-0 | rw-0 | rw-0 |

# PWM Review

- MSP432 Timer A
  - Control Register

**Table 17-4. TAxCTL Register Description**

| Bit | Field | Type | Reset | Description |
|---|---|---|---|---|
| 15-10 | Reserved | RW | 0h | Reserved |
| 9-8 | TASSEL | RW | 0h | Timer_A clock source select<br>00b = TAxCLK<br>01b = ACLK<br>10b = SMCLK<br>11b = INCLK |
| 7-6 | ID | RW | 0h | Input divider. These bits along with the TAIDEX bits select the divider for the input clock.<br>00b = /1<br>01b = /2<br>10b = /4<br>11b = /8 |
| 5-4 | MC | RW | 0h | Mode control. Setting MCx = 00h when Timer_A is not in use conserves power.<br>00b = Stop mode: Timer is halted<br>01b = Up mode: Timer counts up to TAxCCR0<br>10b = Continuous mode: Timer counts up to 0FFFFh<br>11b = Up/down mode: Timer counts up to TAxCCR0 then down to 0000h |
| 3 | Reserved | RW | 0h | Reserved |
| 2 | TACLR | RW | 0h | Timer_A clear. Setting this bit resets TAxR, the timer clock divider logic, and the count direction. The TACLR bit is automatically reset and is always read as zero. |
| 1 | TAIE | RW | 0h | Timer_A interrupt enable. This bit enables the TAIFG interrupt request.<br>0b = Interrupt disabled<br>1b = Interrupt enabled |
| 0 | TAIFG | RW | 0h | Timer_A interrupt flag<br>0b = No interrupt pending<br>1b = Interrupt pending |

$$Fclk = CLK_{src} / 2^{ID} / 2^{TAIDEX}$$

# PWM Review

- MSP432 Timer A
  - Counter Register

**Figure 17-16. TAxR Register**

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| TAxR | | | | | | | |
| rw-0 | rw-0 | rw-0 | rw-0 | rw-0 | rw-0 | rw-0 | rw-0 |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TAxR | | | | | | | |
| rw-0 | rw-0 | rw-0 | rw-0 | rw-0 | rw-0 | rw-0 | rw-0 |

**Table 17-5. TAxR Register Description**

| Bit | Field | Type | Reset | Description |
|---|---|---|---|---|
| 15-0 | TAxR | RW | 0h | Timer_A register. The TAxR register is the count of Timer_A. |

# PWM Review

- MSP432 Timer A
  - Capture/Compare Control Register

4

**Figure 17-17. TAxCCTL0 to TAxCCTL6 Register**

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| CM | | CCIS | | SCS | SCCI | Reserved | CAP |
| rw-0 | rw-0 | rw-0 | rw-0 | rw-0 | r-0 | r-0 | rw-0 |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| OUTMOD | | | CCIE | CCI | OUT | COV | CCIFG |
| rw-0 | rw-0 | rw-0 | rw-0 | r | rw-0 | rw-0 | rw-0 |

4

**Table 17-6. TAxCCTL0 to TAxCCTL6 Register Description**

| Bit | Field | Type | Reset | Description |
|---|---|---|---|---|
| 15-14 | CM | RW | 0h | Capture mode<br>00b = No capture<br>01b = Capture on rising edge<br>10b = Capture on falling edge<br>11b = Capture on both rising and falling edges |
| 13-12 | CCIS | RW | 0h | Capture/compare input select. These bits select the TAxCCR0 input signal. See the device-specific data sheet for specific signal connections.<br>00b = CCIxA<br>01b = CCIxB<br>10b = GND<br>11b = VCC |
| 11 | SCS | RW | 0h | Synchronize capture source. This bit is used to synchronize the capture input signal with the timer clock.<br>0b = Asynchronous capture<br>1b = Synchronous capture |

# PWM Review

- MSP432 Timer A
  - Capture/Compare Control Register

| | | | | |
|---|---|---|---|---|
| 10 | SCCI | RW | 0h | Synchronized capture/compare input. The selected CCI input signal is latched with the EQUx signal and can be read via this bit. |
| 9 | Reserved | R | 0h | Reserved. Reads as 0. |
| 8 | CAP | RW | 0h | Capture mode<br>0b = Compare mode<br>1b = Capture mode |
| 7-5 | OUTMOD | RW | 0h | Output mode. Modes 2, 3, 6, and 7 are not useful for TAxCCR0 because EQUx = EQU0.<br>000b = OUT bit value<br>001b = Set<br>010b = Toggle/reset<br>011b = Set/reset<br>100b = Toggle<br>101b = Reset<br>110b = Toggle/set<br>111b = Reset/set |
| 4 | CCIE | RW | 0h | Capture/compare interrupt enable. This bit enables the interrupt request of the corresponding CCIFG flag.<br>0b = Interrupt disabled<br>1b = Interrupt enabled |
| 3 | CCI | R | 0h | Capture/compare input. The selected input signal can be read by this bit. |
| 2 | OUT | RW | 0h | Output. For output mode 0, this bit directly controls the state of the output.<br>0b = Output low<br>1b = Output high |

# PWM Review

- MSP432 Timer A
  - Capture/Compare Control Register

| Bit | Field | Type | Reset | Description |
|---|---|---|---|---|
| 1 | COV | RW | 0h | Capture overflow. This bit indicates a capture overflow occurred. COV must be reset with software.<br>0b = No capture overflow occurred<br>1b = Capture overflow occurred |
| 0 | CCIFG | RW | 0h | Capture/compare interrupt flag<br>0b = No interrupt pending<br>1b = Interrupt pending |

# PWM Review

- MSP432 Timer A
  - Capture/Compare Register

4

**Figure 17-18. TAxCCR0 to TAxCCR6 Register**

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| TAxCCRn | | | | | | | |
| rw-0 | rw-0 | rw-0 | rw-0 | rw-0 | rw-0 | rw-0 | rw-0 |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TAxCCRn | | | | | | | |
| rw-0 | rw-0 | rw-0 | rw-0 | rw-0 | rw-0 | rw-0 | rw-0 |

4

**Table 17-7. TAxCCR0 to TAxCCR6 Register Description**

| Bit | Field | Type | Reset | Description |
|---|---|---|---|---|
| 15-0 | TAxCCR0 | RW | 0h | Compare mode: TAxCCRn holds the data for the comparison to the timer value in the Timer_A Register, TAxR.<br>Capture mode: The Timer_A Register, TAxR, is copied into the TAxCCRn register when a capture is performed. |

# PWM Review

- MSP432 Timer A
  - Interrupt Register

**Figure 17-19. TAxIV Register**

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| TAIV | | | | | | | |
| r0 | r0 | r0 | r0 | r0 | r0 | r0 | r0 |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TAIV | | | | | | | |
| r0 | r0 | r0 | r0 | r-0 | r-0 | r-0 | r0 |

**Table 17-8. TAxIV Register Description**

| Bit | Field | Type | Reset | Description |
|---|---|---|---|---|
| 15-0 | TAIV | R | 0h | Timer_A interrupt vector value<br>00h = No interrupt pending<br>02h = Interrupt Source: Capture/compare 1; Interrupt Flag: TAxCCR1 CCIFG; Interrupt Priority: Highest<br>04h = Interrupt Source: Capture/compare 2; Interrupt Flag: TAxCCR2 CCIFG<br>06h = Interrupt Source: Capture/compare 3; Interrupt Flag: TAxCCR3 CCIFG<br>08h = Interrupt Source: Capture/compare 4; Interrupt Flag: TAxCCR4 CCIFG<br>~~0Ah = Interrupt Source: Capture/compare 5; Interrupt Flag: TAxCCR5 CCIFG~~<br>~~0Ch = Interrupt Source: Capture/compare 6; Interrupt Flag: TAxCCR6 CCIFG~~<br>0Eh = Interrupt Source: Timer overflow; Interrupt Flag: TAxCTL TAIFG; Interrupt Priority: Lowest |

# PWM Review

- MSP432 Timer A
  - Clock Divider Register

**Figure 17-20. TAxEX0 Register**

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| Reserved | | | | | | | |
| r0 | r0 | r0 | r0 | r0 | r0 | r0 | r0 |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Reserved | | | | | TAIDEX(1) | | |
| r0 | r0 | r0 | r0 | r0 | rw-0 | rw-0 | rw-0 |

(1) After programming TAIDEX bits and configuration of the timer, set TACLR bit to ensure proper reset of the timer divider logic.

**Table 17-9. TAxEX0 Register Description**

| Bit | Field | Type | Reset | Description |
|---|---|---|---|---|
| 15-3 | Reserved | R | 0h | Reserved. Reads as 0. |
| 2-0 | TAIDEX | RW | 0h | Input divider expansion. These bits along with the ID bits select the divider for the input clock.<br>000b = Divide by 1<br>001b = Divide by 2<br>010b = Divide by 3<br>011b = Divide by 4<br>100b = Divide by 5<br>101b = Divide by 6<br>110b = Divide by 7<br>111b = Divide by 8 |

$$Fclk = CLK_{src} / 2^{ID} / 2^{TAIDEX}$$

# PWM Review

- MSP432 Timer A

| TIMER A | Info Sheet | |
|---|---|---|
| | | |
| n=3:0 | x=4:0 | |
| | | |
| TIMER_An -> | CTL | Timer Control |
| | CCTL[x] | Capture/Compare Control |
| | R | Counter |
| | CCR[x] | Capture/Compare |
| | EX0 | Clock Divider |
| | IV | Interrupt Vector |

| | | | |
|---|---|---|---|
| TA0_0_IRQHandler | INTISR[8] | Timer_A0 | TA0CCTL0.CCIFG |
| TA0_N_IRQHandler | INTISR[9] | Timer_A0 | TA0CCTLx.CCIFG (x = 1 through 4), TA0CTL.TAIFG |
| TA1_0_IRQHandler | INTISR[10] | Timer_A1 | TA1CCTL0.CCIFG |
| TA1_N_IRQHandler | INTISR[11] | Timer_A1 | TA1CCTLx.CCIFG (x = 1 through 4), TA1CTL.TAIFG |
| TA2_0_IRQHandler | INTISR[12] | Timer_A2 | TA2CCTL0.CCIFG |
| TA2_N_IRQHandler | INTISR[13] | Timer_A2 | TA2CCTLx.CCIFG (x = 1 through 4), TA2CTL.TAIFG |
| TA3_0_IRQHandler | INTISR[14] | Timer_A3 | TA3CCTL0.CCIFG |
| TA3_N_IRQHandler | INTISR[15] | Timer_A3 | TA3CCTLx.CCIFG (x = 1 through 4), TA3CTL.TAIFG |

# PWM Review

- MSP432 Timer A

| PORT | | | | | | DIR | PSEL[1:0] |
|---|---|---|---|---|---|---|---|
| P7.3 | TA0 | 0 | CCI | 0 | A | 0 | 01 |
| | | | Out | 0 | A | 1 | 01 |
| P2.4 | TA0 | 1 | CCI | 1 | A | 0 | 01 |
| | | | Out | 1 | A | 1 | 01 |
| P2.5 | TA0 | 2 | CCI | 2 | A | 0 | 01 |
| | | | Out | 2 | A | 1 | 01 |
| P2.6 | TA0 | 3 | CCI | 3 | A | 0 | 01 |
| | | | Out | 3 | A | 1 | 01 |
| P2.7 | TA0 | 4 | CCI | 4 | A | 0 | 01 |
| | | | Out | 4 | A | 1 | 01 |
| P7.1 | TA0 | | CLK | | | 0 | 01 |
| | | | | | | | |
| P8.0 | TA1 | 0 | CCI | 0 | A | 0 | 01 |
| | | | Out | 0 | A | 1 | 01 |
| P7.7 | TA1 | 1 | CCI | 1 | A | 0 | 01 |
| | | | Out | 1 | A | 1 | 01 |
| P7.6 | TA1 | 2 | CCI | 2 | A | 0 | 01 |
| | | | Out | 2 | A | 1 | 01 |
| P7.5 | TA1 | 3 | CCI | 3 | A | 0 | 01 |
| | | | Out | 3 | A | 1 | 01 |
| P7.4 | TA1 | 4 | CCI | 4 | A | 0 | 01 |
| | | | Out | 4 | A | 1 | 01 |
| P7.2 | TA1 | | CLK | | | 0 | 01 |

| PORT | | | | | | DIR | PSEL[1:0] |
|---|---|---|---|---|---|---|---|
| P8.1 | TA2 | 0 | CCI | 0 | A | 0 | 01 |
| | | | Out | 0 | A | 1 | 01 |
| P5.6 | TA2 | 1 | CCI | 1 | A | 0 | 01 |
| | | | Out | 1 | A | 1 | 01 |
| P5.7 | TA2 | 2 | CCI | 2 | A | 0 | 01 |
| | | | Out | 2 | A | 1 | 01 |
| P6.6 | TA2 | 3 | CCI | 3 | A | 0 | 01 |
| | | | Out | 3 | A | 1 | 01 |
| P6.7 | TA2 | 4 | CCI | 4 | A | 0 | 01 |
| | | | Out | 4 | A | 1 | 01 |
| P4.2 | TA2 | | CLK | | | 0 | 10 |
| | | | | | | | |
| P10.4 | TA3 | 0 | CCI | 0 | A | 0 | 01 |
| | | | Out | 0 | A | 1 | 01 |
| P10.5 | TA3 | 1 | CCI | 1 | A | 0 | 01 |
| | | | Out | 1 | A | 1 | 01 |
| P8.2 | TA3 | 2 | CCI | 2 | A | 0 | 01 |
| | | | Out | 2 | A | 1 | 01 |
| P9.2 | TA3 | 3 | CCI | 3 | A | 0 | 01 |
| | | | Out | 3 | A | 1 | 01 |
| P9.3 | TA3 | 4 | CCI | 4 | A | 0 | 01 |
| | | | Out | 4 | A | 1 | 01 |
| P8.3 | TA3 | | CLK | | | 0 | 01 |

# PWM Review

- MSP432 Timer A
  - Examples – 10KHz square wave on TA2-OUT3A (P6.6)
    - assume 48MHz clock → SMCLK = 12MHz

    10KHz → 100us period  12MHz → 83.33ns period
    →1200 12MHz clocks in each 10KHz period
    or 600 12MHz clocks in each half period of 10KHz

    50% duty cycle → 600 clocks high, 600 clocks low
    → CCRx =600, CCR0 = 1200 in RESET/SET mode

# PWM Review

- MSP432 Timer A
  - Examples – 10KHz square wave on TA2-OUT3A (P6.6)

```
// setup pin P6.6 for use as TA2-OUT3
P6->SEL0 |= 0x40;
P6->SEL1 &= ~0x10;
P6->DIR |= 0x40;

//
// setup timer A2 for 10KHz square wave
//
// note P6.6 is TimerA2-OUT3
// SMCLK is 12MHz with a 48MHz HSMCLK
//
// 10KHz square wave requires TA2-CCR2=600, TA2-CCR0=1200
//
//       SMCLK   /1  Up      no_int
// xxxx xx10     00  01  x0     0    x
TIMER_A2->CTL = 0x0210;
//
// no_cap    asyn   comp   r/s  no_int
// 00    xx   0   xx  0    111    0    xxxx
TIMER_A2->CCTL[3] = 0x00E0;
//
TIMER_A2->CCR[0] = 1200;
TIMER_A2->CCR[3] = 600;
```