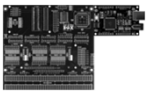
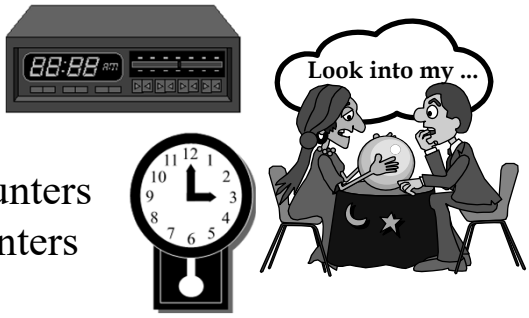


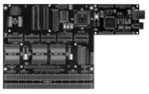
EEL 3744

Menu

- Introduction to OCx and Timers
- PWM
- Basics of XMEGA counters
- Lab 3 and Simple Counters



See docs/examples on web-site:
doc8331 (Sec 14), doc8385 (Sec 16)

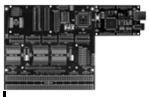


EEL 3744

Timer/Counter System

- Microcontrollers are equipped with precision timing systems (much more precise than RTI/RTC systems)
- The Timer/Counter (TC) system is essentially a counter that increments or decrements based on one of these
  - > Regular clock pulses and a timer prescaler (timer)
  - > Irregular event pulses (counter)
- Useful for
  - > Timing
  - > Periodic Interrupts or Event Generation
  - > Pulse Width Modulation
  - > Event counting
  - > Signal Measurements

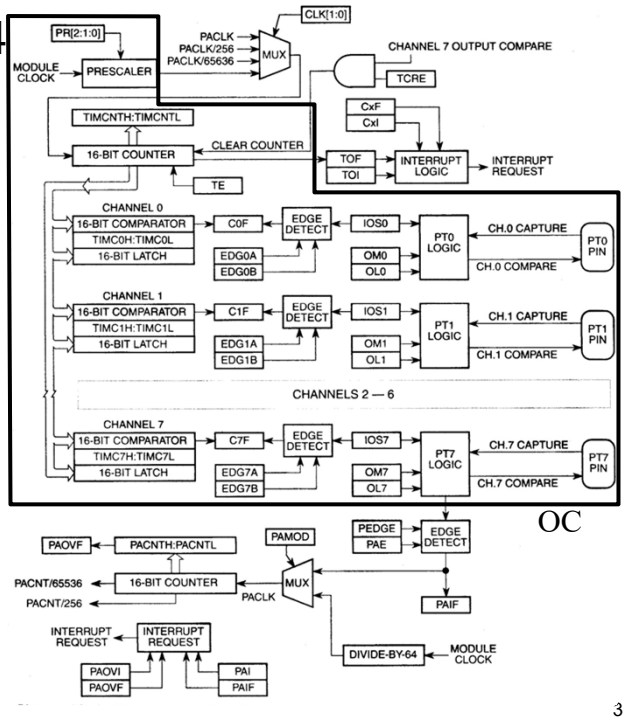
See doc8385, sec 16

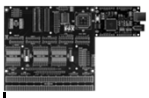


EEL 3744

68HC12 Timer System Block Diagram: OC System

S&HE: Fig 10.1

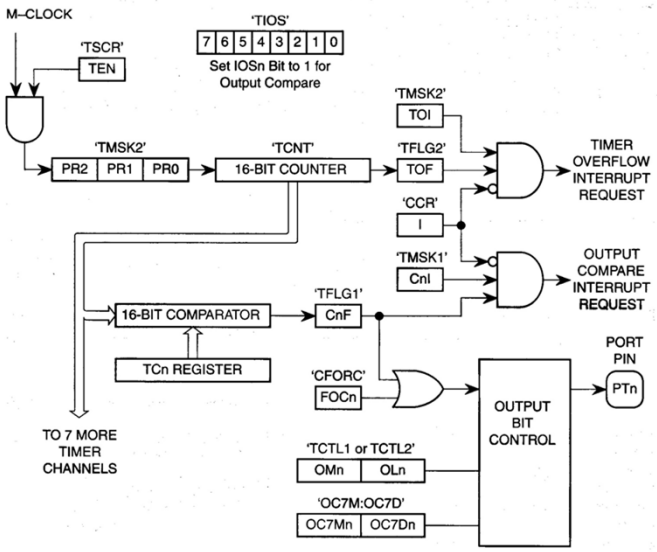


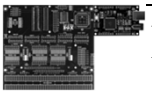


EEL 3744

68HC12 Main Timer System

S&HE: Fig 10.3





EEL 3744

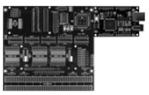
68HC12 TIOS: Timer Input Capture/Output Compare Select Register & DDRT

- TIOS - Timer Input Capture/Output Compare Select Register
  >0 = IC; **1= OC**

	7	6	5	4	3	2	1	0	
$0080	IOS7	IOS6	IOS5	IOS4	IOS3	IOS2	IOS1	IOS0	**TIOS**
RESET	0	0	0	0	0	0	0	0	

- DDRT - Data Direction Register for Port T
  >0 = Input; **1 = Output**

	7	6	5	4	3	2	1	0	
$00AF	DDRT7	DDRT6	DDRT5	DDRT4	DDRT3	DDRT2	DDRT1	DDRT0	**DDRT**
RESET	0	0	0	0	0	0	0	0	



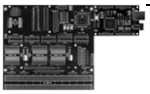
EEL 3744

68HC12 TCNT and TSCR

	7	6	5	4	3	2	1	0	
$0084	Bit 15	-	-	-	-	-	-	Bit 8	**TCNT** High
RESET	0	0	0	0	0	0	0	0	
	7	6	5	4	3	2	1	0	
$0085	Bit 7	-	-	-	-	-	-	Bit 0	**TCNT** Low
RESET	0	0	0	0	0	0	0	0	

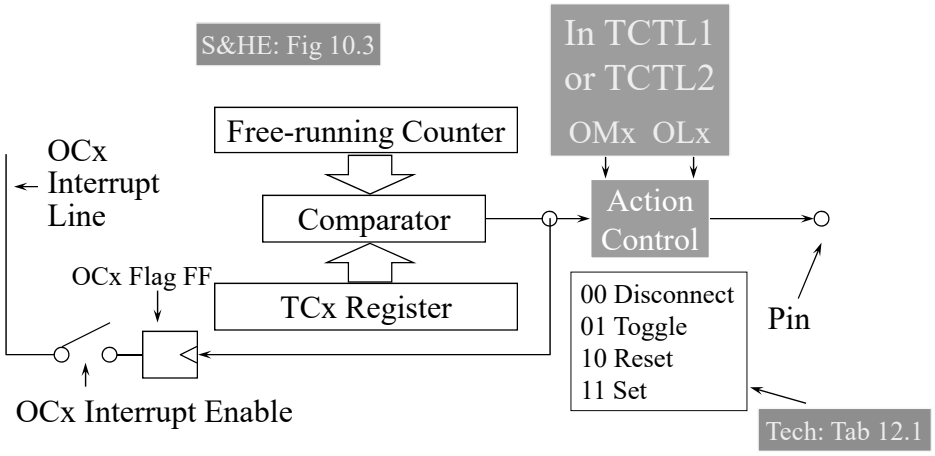
	7	6	5	4	3	2	1	0	
$0086	**TEN**	TSWAI	TSBCK	TFFCA	0	0	0	0	**TSCR**
RESET	0	0	0	0	0	0	0	0	

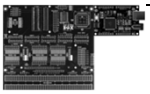
- TEN (Timer Enable) in TSCR:
  >0 = disable
  >**1 = enable**



EEL 3744 **68HC12** Output Compare Block Diagram (from OM,OL)

- Block Diagram of Output Compare (from OM,OL)



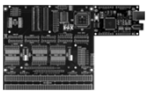


EEL 3744 **68HC12** Input Capture / Output Compare Registers

- TCx - Timer Input Capture / Output Compare x

	7	6	5	4	3	2	1	0	
$0090	Bit 15	-	-	-	-	-	-	Bit 8	TC0
$0091	Bit 7	-	-	-	-	-	-	Bit 0	
$0092	Bit 15	-	-	-	-	-	-	Bit 8	TC1
$0093	Bit 7	-	-	-	-	-	-	Bit 0	○
				○ ○ ○					○
									○
$009C	Bit 15	-	-	-	-	-	-	Bit 8	TC6
$009D	Bit 7	-	-	-	-	-	-	Bit 0	
$009E	Bit 15	-	-	-	-	-	-	Bit 8	TC7
$009F	Bit 7	-	-	-	-	-	-	Bit 0	

RESET = $0000

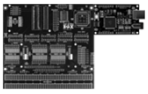


EEL 3744

XMEGA 16-bit Timer/Counter Type 0 and Type 1

See doc8331, Sec 14 & doc8385, Sec 16

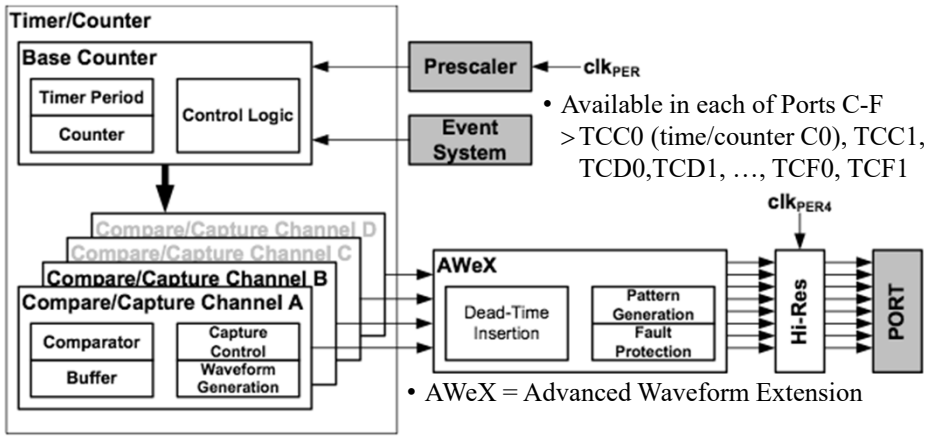
- XMEGA has a set of eight 16-bit timer/counters (TC)
- Two TCs can be combined to create a 32-bit TC
- A TC consists of a base counter and a set of compare or capture (CC) channels
  > Waveform generation available
- TC 0 has four CC channels
  > TC 0 has the split mode feature that split it into two 8-bit Timer/Counters with four compare channels each
- TC 1 has two CC channels

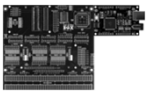


EEL 3744

XMEGA Timer/Counter type 0 and type 1

See doc8331, Fig 14-1

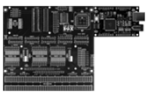




EEL 3744

XMEGA Prescaler

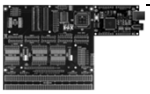
- Timer is limited by size of counter register and rate at which counter changes (with no timer prescaler)
  - >8-bit maxes at count of 256 (~128 μs, if CLK=0.5 μs)
  - >16-bit maxes at count of 65535 (~32,767 μs, if CLK=0.5 μs)
- Prescaler modifies the standard timer clock frequency by a chosen value
  - >Allows timer to be clocked at a desired rate
- Forces a tradeoff between resolution and range
  - >Important with smaller 8-bit and 16-bit counters



EEL 3744

XMEGA Prescaler Example

- If clock is at 2 MHz (with no prescaler)
  - ➔ Period (ticks) = 0.5 μs [= 1/(2 MHz)]
  - >An 8-bit counter incrementing at every tick will max out at 256 ticks or 128 μs
  - >A 16-bit counter will overflow after 64k = 65,536 ticks or 32.768 ms
- If clock is at 32 MHz ➔ Period (of ticks) = 31.25 ns
  - >If 16-bit counter with **no** prescaler, then counter overflows at 2.048 ms [= 64k * 31.25 ns]
  - >If 16-bit counter with **1024 prescaler**, then counter overflows at 2.1 s [≈ 64k * 1024 * 31.25 ns]

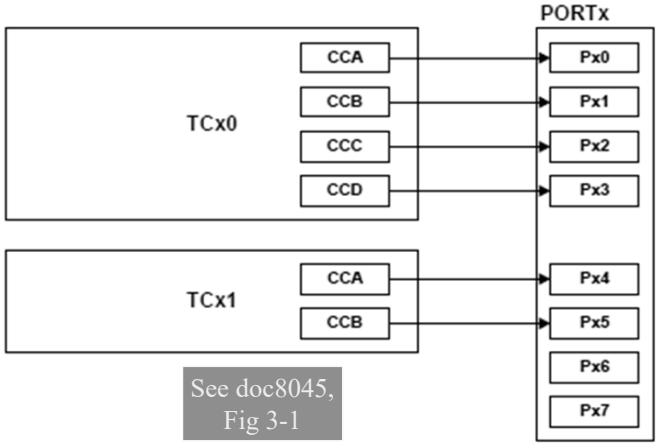


EEL 3744

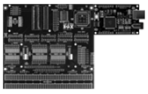
XMEGA Timer/Counter Type 0/1 OC I/O Port Pin Mapping

- Timer TCxn, where x indicates the port (C, D, E, or F) and n is the TC number within PORTx.
  - > Example: TCD0 is Timer/Counter 0 connected to PORTD

See doc8045, Sec 3.1



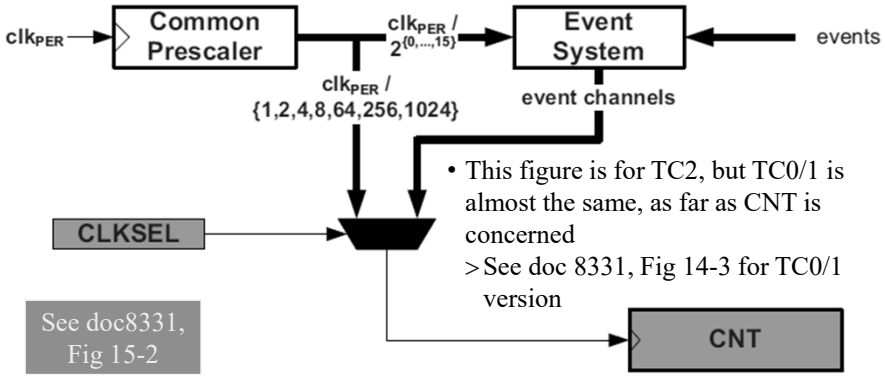
See doc8045, Fig 3-1



EEL 3744

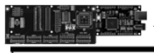
XMEGA TC Clock Sources

- The timer/counter can be clocked from the peripheral clock (clk_{PER}) and from the event system



- This figure is for TC2, but TC0/1 is almost the same, as far as CNT is concerned
  - > See doc 8331, Fig 14-3 for TC0/1 version

See doc8331, Fig 15-2



EEL 3744

XMEGA TC Clock Sources

Symbol	Clock selection
TC_CSEL_OFF_gc	TC off (no clock selected)
TC_CSEL_DIV1_gc	$f_{CLK,SYS}$
TC_CSEL_DIV2_gc	$f_{CLK,SYS}$ / 2
TC_CSEL_DIV4_gc	$f_{CLK,SYS}$ / 4
TC_CSEL_DIV8_gc	$f_{CLK,SYS}$ / 8
TC_CSEL_DIV64_gc	$f_{CLK,SYS}$ / 64
TC_CSEL_DIV256_gc	$f_{CLK,SYS}$ / 256
TC_CSEL_DIV1024_gc	$f_{CLK,SYS}$ / 1024
TC_CSEL_EV0_gc	Event channel 0
TC_CSEL_EV1_gc	Event channel 1
TC_CSEL_EV2_gc	Event channel 2
TC_CSEL_EV3_gc	Event channel 3
TC_CSEL_EV4_gc	Event channel 4
TC_CSEL_EV5_gc	Event channel 5
TC_CSEL_EV6_gc	Event channel 6
TC_CSEL_EV7_gc	Event channel 7

We'll use these

TC not running

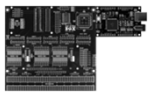
Called **TC_CLKSEL_DIV4_gc** in include file

- The available clock source selections for the XMEGA TC modules
  - > A selection of prescaler outputs from 1 to 1024 is directly available (**CLKSEL**)
  - > The whole range of time prescalings from 1 to 2^{15} is available through the event system (**HCNT** | **LCNT**)

See doc8045, Tab 3-1

See doc8331, Sec 15.4

15

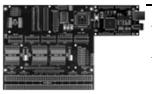


EEL 3744

XMEGA Timer/Counter Terminology

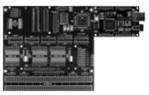
See doc8331, Sec 14.2.1

- "**Timer**" is used when the timer/counter clock control is handled by an internal source
- "**Counter**" is used when the clock control is handled externally (e.g., counting external **events**)
- **CC** = **Compare** / **Capture**
  - > When used for compare operations, the **CC** channels are referred to as "**compare channels**"
  - > When used for capture operations, the **CC** channels are referred to as "**capture channels**"



EEL 3744 **XMEGA** Timer/Counter Terminology

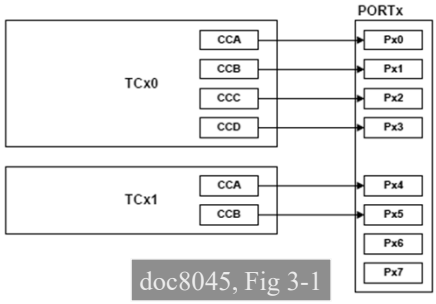
- **CCx** – Compare/Capture registers
  - >In Capture mode, if a capture event is triggered, the current **CNT** value is loaded into the enabled **CCx** register
    - – Used to time intervals between pulses, determine high and low points of input signals, and to define time between two input signals
  - >In Compare mode, the **CNT** register is constantly compared to the **CCx** registers
    - – If **CNT** = **CCx**, then a match event occurs



EEL 3744

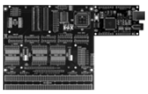
XMEGA CCx

- The compare or capture channels consist of a set of 16-bit registers named CCx[H:L], where x indicates the channel (A, B, C, D)
  - >Timer0, with 4 channels, has
    - – CCA[H:L], CCB[H:L], CCC[H:L] and CCD[H:L]
  - >Timer1, with 2 channels has
    - – CCA[H:L] and CCB[H:L].
  - >Each CCx[H:L] register has an associated buffer register CCxBUF[H:L].



doc8045, Fig 3-1

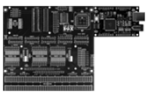
See doc8045, Sec 3-5



EEL 3744 **XMEGA** Timer/Counter Terminology

See doc8331, Sec 14.2.1

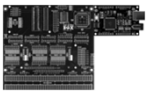
- **CNT** – Count register is incremented or decremented every clock cycle (possibly modified by prescaler)
  - >Used by TC module to perform compare/capture operations
  - >May be read or written to as needed
- **PER** – Period register holds the "TOP" value for the TC count



EEL 3744 **XMEGA** Timer/Counter Terminology

See doc8331, Sec 14.2.1

- **BOTTOM**: When the counter reaches zero
- **MAX**: The counter reaches MAXimum (all ones)
- **TOP**: The counter reaches TOP when it becomes equal to the highest value in the count sequence
  - > The TOP value can be equal to the period (PER) or the compare channel A (CCA) register setting
    - – This is selected by the waveform generator mode
- **UPDATE**: The timer/counter signals an update when it reaches BOTTOM or TOP, depending on the waveform generator mode

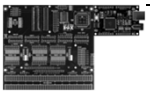


EEL 3744

XMEGA TC Modes of Operation

- Normal mode
  - > We will discuss the below other modes (below) later:
    - – Frequency Generation mode
    - – Single Slope PWM
    - – Dual Slope PWM, overflow on TOP
    - – Dual Slope PWM, overflow on TOP and BOTTOM
    - – Dual Slope PWM, overflow on BOTTOM

See doc8045, Sec 3.8

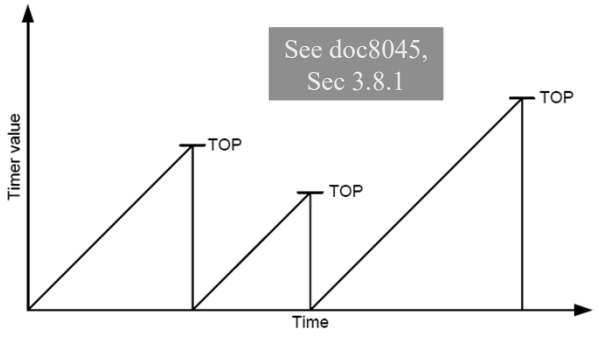


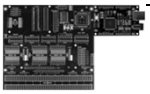
EEL 3744

XMEGA TC Normal Mode

- In Normal Mode, the counter will count in direction set by the DIR bit in CTRLF for each clock until it reaches TOP (when counting up), set by PER[H:L], or BOTTOM (zero, when counting down)
- When TOP is reached when up-counting the counter will be set to zero when the next clock is given
  - > If the TC is down-counting the value will wrap around to the value in PER[H:L] after reaching BOTTOM

See doc8045, Sec 3.8.1



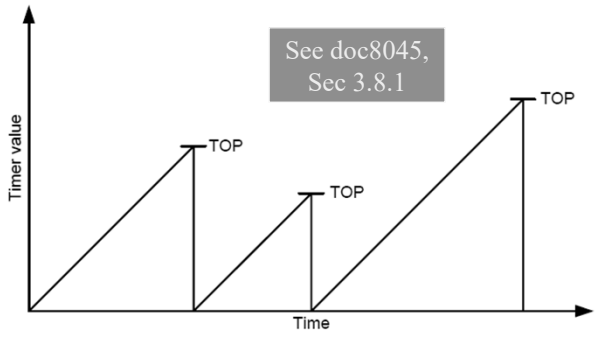


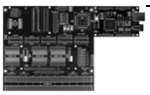
EEL 3744

XMEGA TC Normal Mode

- Changing the counter value while the counter is running is allowed
- The write access has higher priority than count, clear, or reload and will be immediate
  - > However, if the value written is outside the BOTTOM-TOP boundary the counter either has to count down until TOP is reached or count up until wraparound (passing **MAX**) for the timer to re-stabilize to the period time

See doc8045, Sec 3.8.1



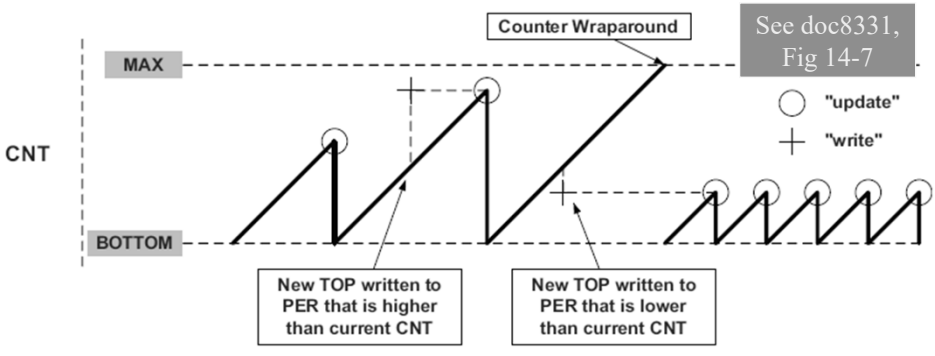


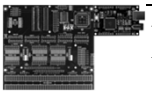
EEL 3744

XMEGA TC Normal Mode (PER)

- Counter **period** is changed by writing a new **TOP** value to the period register

See doc8331, Fig 14-7





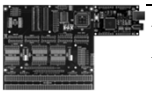
EEL 3744 **XMEGA** CTRLA Register

• CTRLA – Controls the clock source for timers

TCpx_CTRLA	7	6	5	4	3	2	1	0	
+0x00	-	-	-	-	CLKSEL3	CLKSEL2	CLKSEL1	CLKSEL0	**CTRLA**
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

CLKSEL[3..0]	Group Config	Description
0000	Off	None (ie., timer/counter in OFF state)
0001	DIV1	Prescaler: Clk
0010	DIV2	Prescaler: Clk/2
0011	DIV4	Prescaler: Clk/4
0100	DIV8	Prescaler: Clk/8
0101	DIV64	Prescaler: Clk/64
0110	DIV256	Prescaler: Clk/256
0111	DIV1024	Prescaler: Clk/1024
1nnn	EVCHn	Event channel n, n=[0,…,7]

doc8331: Table 14-3



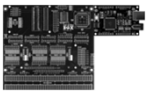
EEL 3744 **XMEGA** CNT Register

• CNTL – LSB of Register Pair for CNT

TCpx_CNT	7	6	5	4	3	2	1	0	
+0x20	CNT7	CNT6	CNT5	CNT4	CNT3	CNT2	CNT1	CNT0	**CNTL**
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

• CNTH – MSB of Register Pair for CNT

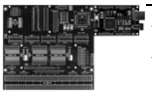
TCpx_CNT+**1**	7	6	5	4	3	2	1	0	
+0x21	CNT15	CNT14	CNT13	CNT12	CNT11	CNT10	CNT9	CNT8	**CNTH**
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	



EEL 3744

Storing to 16-bit Registers

- When storing to a 16-bit register (like the TC's PER, CC, our CNT registers) it is generally necessary to write to **BOTH** bytes of the register in order to have it take effect



EEL 3744

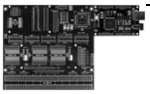
XMEGA PER Register

- PERL – LSB of Register Pair for PER

TCpx_PER	7	6	5	4	3	2	1	0	
+0x26	PER7	PER6	PER5	PER4	PER3	PER2	PER1	PER0	**PERL**
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- PERH – MSB of Register Pair for PER

TCpx_PER+**1**	7	6	5	4	3	2	1	0	
+0x27	PER15	PER14	PER13	PER12	PER11	PER10	PER9	PER8	**PERH**
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	



EEL 3744

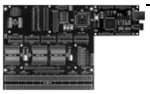
XMEGA CCx Register

• CCxL – LSB of Register Pair for CCx

TCpx_CCy	7	6	5	4	3	2	1	0	
	CCx7	CCx6	CCx5	CCx4	CCx3	CCx2	CCx1	CCx0	**CCxL**
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

• CCxH – MSB of Register Pair for CCx

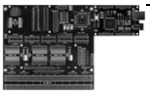
TCpx_CCy+**1**	7	6	5	4	3	2	1	0	
	CCx15	CCx14	CCx13	CCx12	CCx11	CCx10	CCx9	CCx8	**CCxH**
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	



EEL 3744

Summary for Counter part of Lab 3

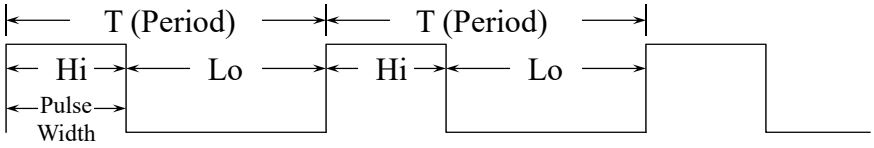
- Select a counter (using one of Ports C through F)
- Setup a 16-bit counter (TCpx, where p is C, D, E, or F and x is 0,1, or 2)
- Setup up the counter to count from 0 to 255 (and repeat)
  >Initialize TCpx_PER
- Setup PortC or PortF for 8 outputs (PORTp_DIR*)
  >These outputs will be the low 8 bits of the 16-bit counter
- Set the counter increment rate
  >Initialize TCpx_CTRLA
- Make output port continuously display the count



EEL 3744

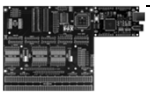
PWM Signal

- Pulse Width Modulation (PWM) Signal
  - >Period (T) is constant



- Duty Cycle is the ratio of "on time" to "off time" during one period
  - >Duty Cycle = (High Time / Period) * 100%

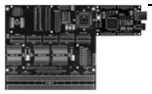
$$\textbf{Duty Cycle (\%)} = \frac{\textbf{Hi}}{\textbf{T}} \times \textbf{100}$$



EEL 3744

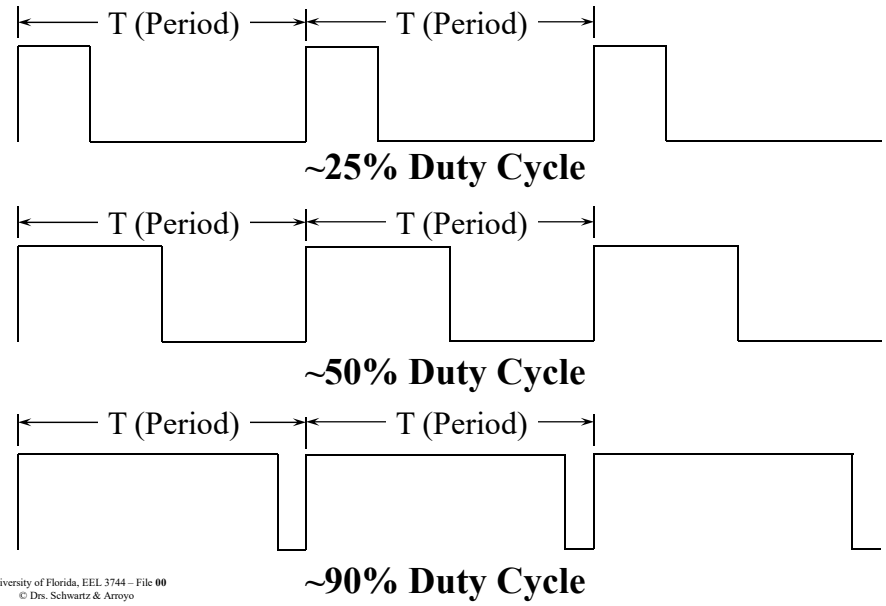
Pulse Width Modulation (PWM)

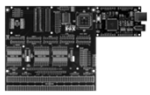
- PWM is a method of controlling **analog** circuits with **digital outputs** by delivering energy through a **sequence of pulses**
- A signal (square wave) is generated by controlling when to turn a digital signal on or off
- Used for controlling
  - >Servos
  - >Motors
  - >Speakers
  - >Etc.



EEL 3744

PWM Duty Cycle

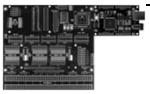




EEL 3744

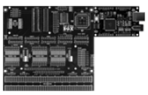
RGB and PWM

- Color is often specified using RGB (red, green, and blue)
- Each of R, G, and B can take on a value between 0 and 255 (0xFF)
- An RGB LED only allows 0 (Gnd) and 1 (Vcc) for each of the colors
- Use PWM with set duty cycles for each of the colors to effectively generate the 256 values



EEL 3744 Summary for Counter part of Lab 4

- Set up the counter and output port for the RGB LEDs
- Use PWM to approximate an RGB color scheme
- Test the RGB colors as described in the lab handout
  >Your TA will specify different RGB values in your lab



EEL 3744

The End!