EE445M/EE360L.6
Embedded and Real-Time Systems/ Real-Time Operating Systems

Lecture 9:

Sensing & Acting,
Input Capture, PWM, Motors

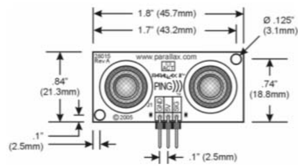
Sharp GP2Y0A21YK

- Infrared distance sensor
  - You will need 5V to power IR sensor
    - Needs 10 mF or larger +5V to Gnd cap for each sensor (supply stabilization)
  - Needs analog LPF
    - Reduces noise
    - Analog input protection
  - Needs digital median filter
  - Needs calibration

See Lecture 7

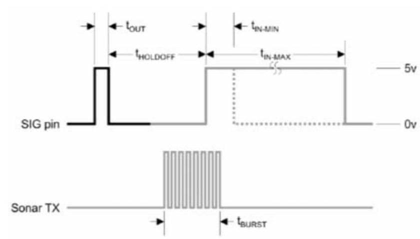
Ping Distance Sensor

- Ultrasound transducers to measure distance
  - Ping)))
    - One **SIG** pin for both input & output
  - HCSR04
    - Two signals: **Trig** output and **Echo** input
  - Need 5V to power
    - Use 5V tolerant input (not all are)



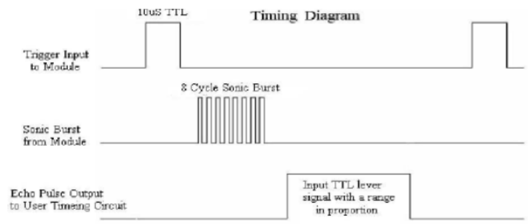
Ping))) Sensor

- Sample 10 times a second
  1) Disable interrupts
  2) Make the **SIG** pin an output
  3) Issue a 5μs output pulse (causing a sound pulse)
  4) Switch the **SIG** pin back to an input
  5) Enable interrupts
  6) Measure time until the echo is received
     - Busy-wait if foreground, interrupt if background



HCSR04 Sensor

- Sample 10 times a second
  1) Disable interrupts
  2) Issue a 10µs output pulse (causing a sound pulse)
  3) Enable interrupts
  4) Measure time until the echo is received
     - Busy-wait if foreground, interrupt if background

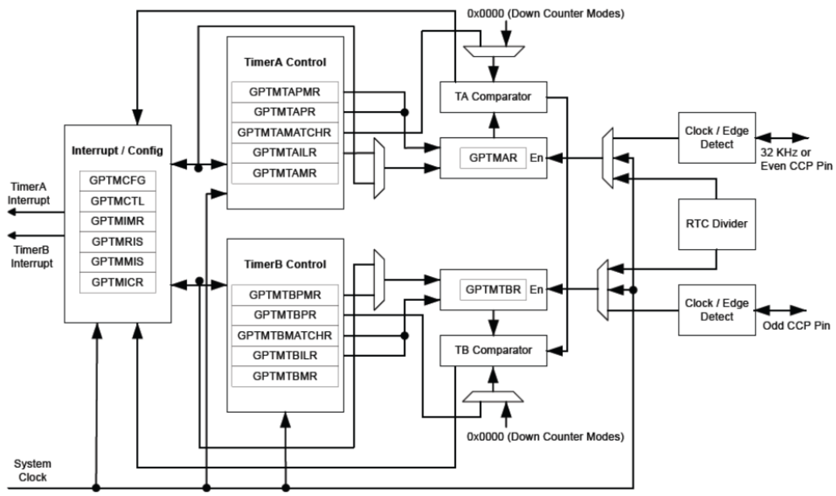


Input Capture

- General purpose timers
  – TM4C123: 6 GP timers (Timer 0…Timer 5)
  – CCPx pins used for input capture
    - CCP0=PD4
- Input edge time (input capture) mode
  – Detect rising/falling input edges
  – Make time measurements on input signals

See book Section 8.1

General Purpose Timer



Lecture 9 J. Valvano, A. Gerstlauer EE445M/EE380L.6 7

Input Capture Mode

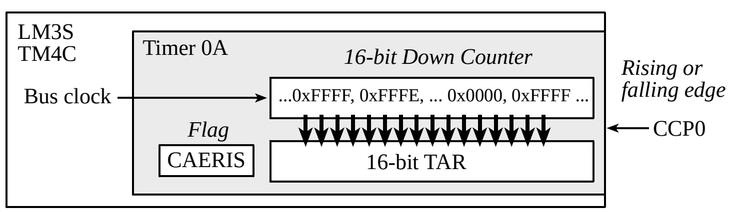


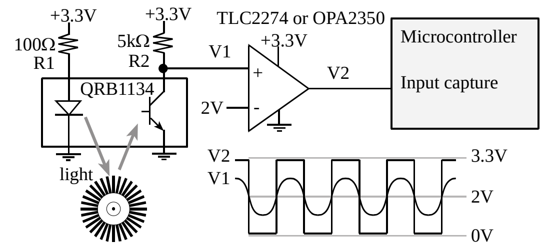
Figure 8.2. Rising or falling edge of CCP0 causes the counter to be latched into TAR, setting CAERIS.

- Generate edge based interrupts
- Count events
- Measure period
- Measure pulse width

Lecture 9 J. Valvano, A. Gerstlauer EE445M/EE380L.6 8

Event Counting

- Count wheel turns (tachometer)



InputCapture_4C123.zip

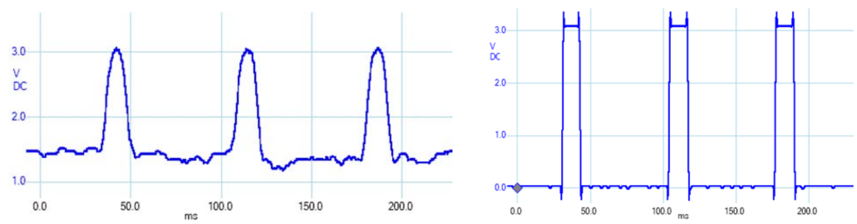


Figure 8.4. Measured V1 and V2

Period Measurement

- Init
  - Select clock period, Δt (measurement resolution)
  - TIMER0_TAILR_R = 0xFFFF (reload=wraparound)
  - Choose edge (rise or fall)
  - Arm interrupt on capture
- ISR
  - Poll to see which channel (if needed)
  - Now = captured time (TIMER0_TAR_R)
  - Period = Last – Now
  - Last = Now
  - Acknowledge interrupt
  - Save/process period

PeriodMeasure_4C123.zip

Resolution, Precision, Range

- How to choose the resolution?
  - Determine minimum & maximum robot speed
  - Convert speed to tachometer period

Period	7100	
	4	holes/rotation
Resolution	10	μsec
Speed	3.521127	rps
Speed	211.2676	RPM

- How to detect speed too slow (period too large)?
  - Clear a counter on each tachometer edge
  - AddPeriodicThread
    - Increment the counter on each rollover 0000 to FFFF
    - If counter >= 2, then wheel is stopped

Lecture 9 J. Valvano, A. Gerstlauer EE445M/EE380L.6 11

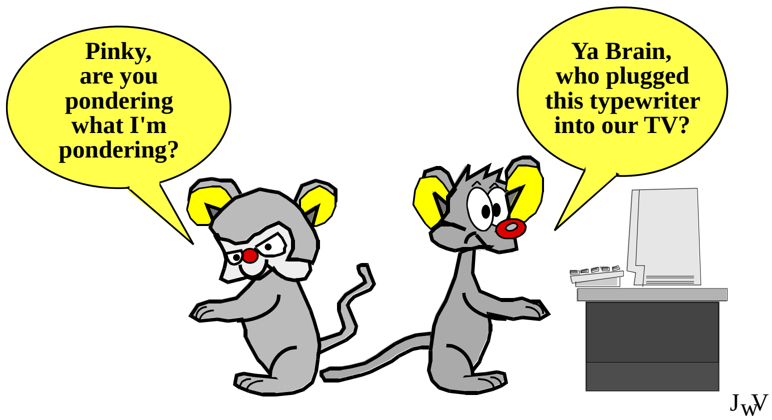
Ping Distance Measurement

- Input pulse width
  - Time $\mathbf{t_{IN}}$ for sound to travel back and forth
  - $\mathbf{t_{IN}} = 2\ \mathbf{d/c}$ (**c**: speed of sound)
- Measure using input capture
  - Rising edge: record TAR
  - Falling edge: calculate distance $\mathbf{d} = \mathbf{c} * \mathbf{t_{IN}}/2$

Lecture 9 J. Valvano, A. Gerstlauer EE445M/EE380L.6 12

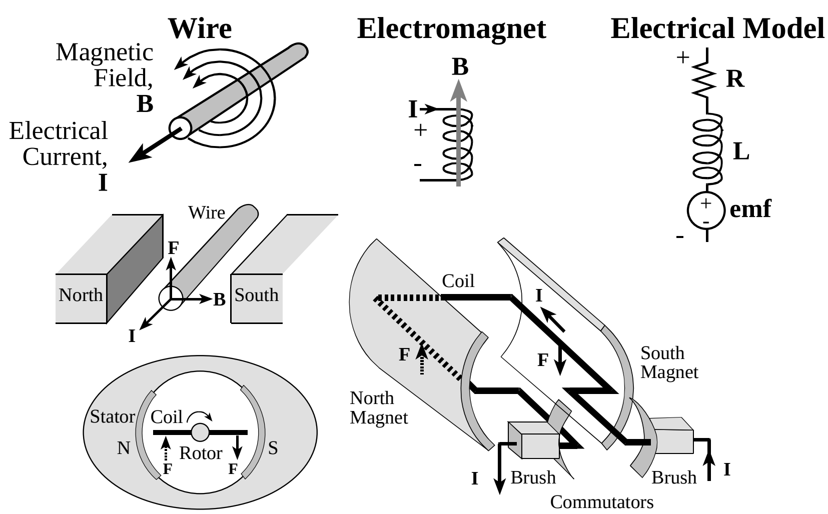
Motor Interfacing

- Motor physics
- Transistor-level interface



Lecture 9 J. Valvano, A. Gerstlauer EE445M/EE380L.6 13

Motor Physics



Lecture 9 J. Valvano, A. Gerstlauer EE445M/EE380L.6 14

Digital Interfacing

V_{OL} is defined as the voltage at maximum I_{OL}

Family	*Example*	I_{OH}	I_{OL}	I_{IH}	I_{IL}	*fan out*
Standard TTL	7404	0.4 mA	16 mA	40 µA	1.6 mA	10
Schottky TTL	74S04	1 mA	20 mA	50 µA	2 mA	10
Low Power Schottky	74LS04	0.4 mA	4 mA	20 µA	0.4 mA	10
High speed CMOS	74HC04	4 mA	4 mA	1 µA	1 µA	
LM3S/LM4F 2mA-drive	LM3S811	2 mA	2 mA	2 µA	2 µA	
LM3S/LM4F 4mA-drive	LM3S811	4 mA	4 mA	2 µA	2 µA	
LM3S/LM4F 8mA-drive	LM3S811	8 mA	8 mA	2 µA	2 µA	

Electrical specifications
- See Chapter 24 of TM4C123
- 5V tolerant?
- PD0, PD1 ⇔ PB7,PB6

All GPIO signals are 5-V tolerant when configured as inputs except for PD4, PD5, PB0 and PB1, which are limited to 3.6 V.

Lecture 9 J. Valvano, A. Gerstlauer EE445M/EE380L.6 15

Motor Interface

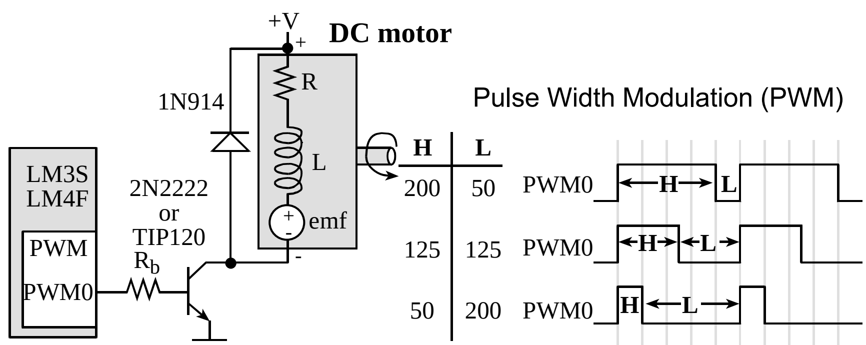
- Darlington transistor
  - TIP120 (NPN)
  - h_{fe} = 1000
  - I_{ce} = 3A

$I_b = I_{coil} / h_{fe}$ = 1A/1000 = 1mA

$R_b \le (V_{OH}-V_{be})/I_b$ = (3-2.5)/1mA = 0.5 kΩ

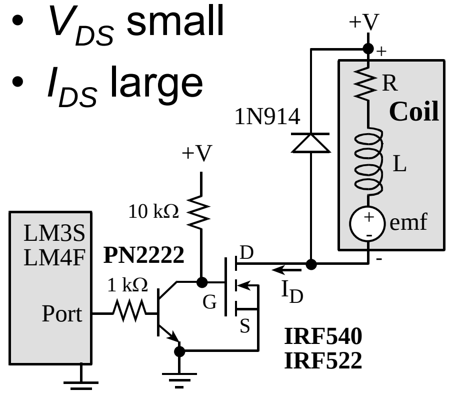
R_b = 100 Ω

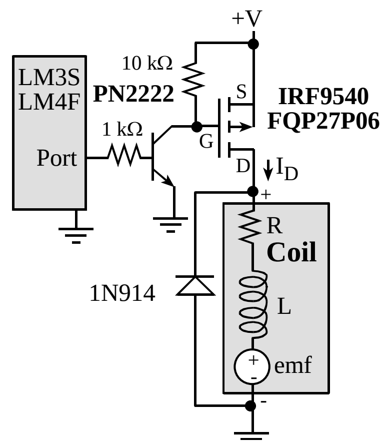
V_{CE} depends on current



MOSFET Interface

- V_{GS} turns on
- V_{DS} small
- I_{DS} large

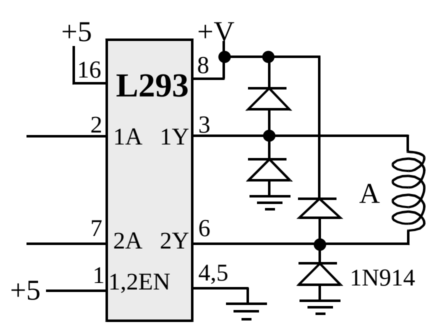


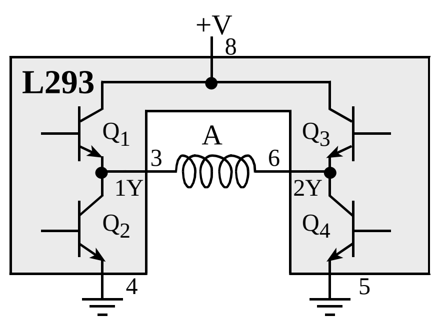


Lecture 9 J. Valvano, A. Gerstlauer EE445M/EE380L.6 17

H-bridge Interface

- Both directions (forward & backward)
- V_{OH} = +V-1.4, V_{OL} = *1.2*

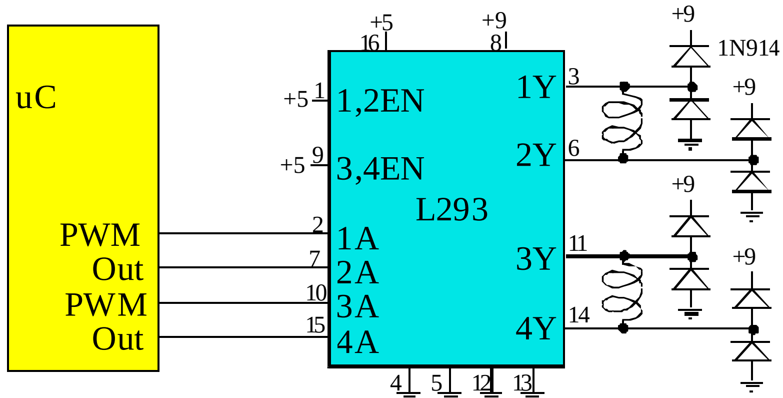




Lecture 9 J. Valvano, A. Gerstlauer EE445M/EE380L.6 18

H-bridge Interface (V1)

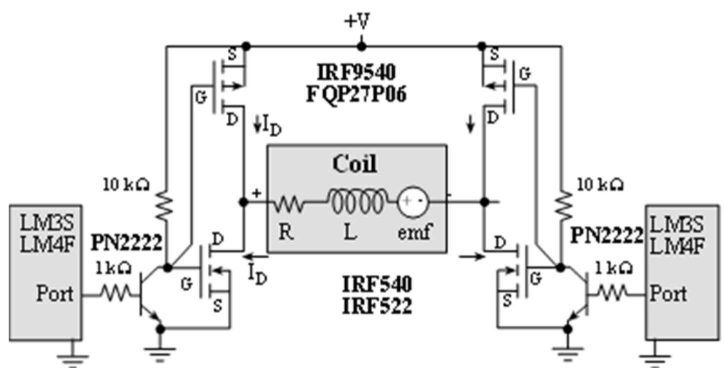
- PWM controls power
- Out controls direction



Lecture 9 J. Valvano, A. Gerstlauer EE445M/EE380L.6 See L6203.pdf 19

H-bridge Interface (V2)

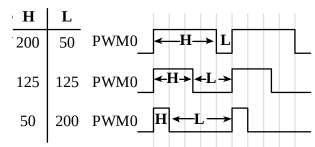
- One Port is PWM controlling power
- Other port controls direction



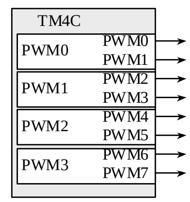
Lecture 9 J. Valvano, A. Gerstlauer EE445M/EE380L.6 20

Pulse Width Modulation (PWM)

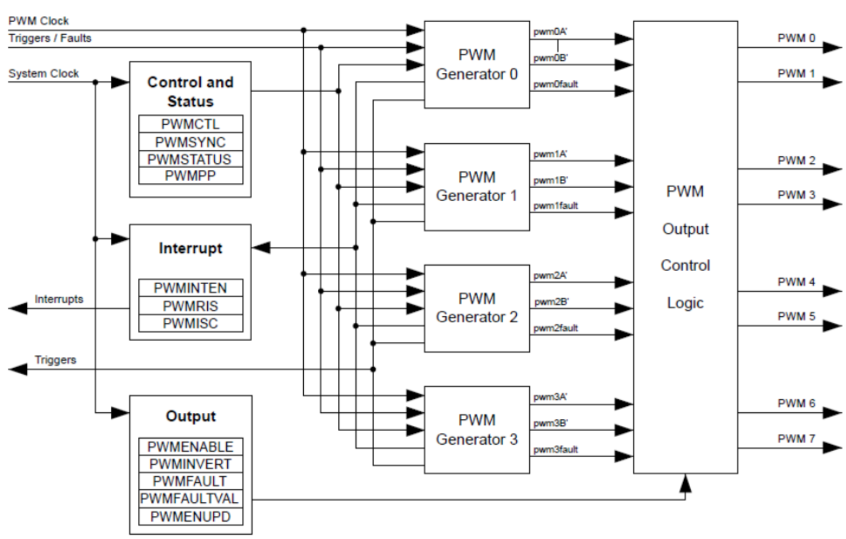
- Generate output waveform
  - Period = High + Low
  - Duty cycle = High / Period



- PWM generators
  - TM4C123: 2 modules
    - 4 generators per module
    - 2 PWM signals per generator



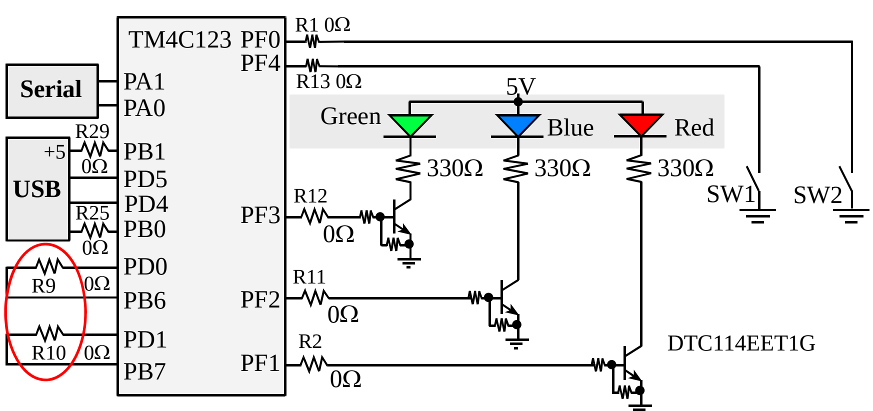
PWM Module



TM4C123 Alternate Function

IO	Ain	0	1	2	3	4	5	6	7	8	9	14
PA0		Port	U0Rx							CAN1Rx		
PA1		Port	U0Tx							CAN1Tx		
PA2		Port		SSI0Clk								
PA3		Port		SSI0Fss								
PA4		Port		SSI0Rx								
PA5		Port		SSI0Tx								
PA6		Port			I$_2$C1SCL		M1PWM2					
PA7		Port			I$_2$C1SDA		M1PWM3					
PB0	USB0ID	Port	U1Rx						T2CCP0			
PB1	USB0VBUS	Port	U1Tx						T2CCP1			
PB2		Port			I$_2$C0SCL				T3CCP0			
PB3		Port			I$_2$C0SDA				T3CCP1			
PB4	Ain10	Port		SSI2Clk		M0PWM2			T1CCP0	CAN0Rx		
PB5	Ain11	Port		SSI2Fss		M0PWM3			T1CCP1	CAN0Tx		
PB6		Port		SSI2Rx		M0PWM0			T0CCP0			
PB7		Port		SSI2Tx		M0PWM1			T0CCP1			
PC4	C1-	Port	U4Rx	U1Rx		M0PWM6		IDX1	WT0CCP0	U1RTS		
PC5	C1+	Port	U4Tx	U1Tx		M0PWM7		PhA1	WT0CCP1	U1CTS		
PC6	C0+	Port	U3Rx					PhB1	WT1CCP0	USB0epen		
PC7	C0-	Port	U3Tx						WT1CCP1	USB0pflt		
PD0	Ain7	Port	SSI3Clk	SSI1Clk	I$_2$C3SCL	M0PWM6	M1PWM0		WT2CCP0			
PD1	Ain6	Port	SSI3Fss	SSI1Fss	I$_2$C3SDA	M0PWM7	M1PWM1		WT2CCP1			
PD2	Ain5	Port	SSI3Rx	SSI1Rx		M0Fault0			WT3CCP0	USB0epen		
PD3	Ain4	Port	SSI3Tx	SSI1Tx				IDX0	WT3CCP1	USB0pflt		
PD4	USB0DM	Port	U6Rx						WT4CCP0			
PD5	USB0DP	Port	U6Tx						WT4CCP1			
PD6		Port	U2Rx			M0Fault0		PhA0	WT5CCP0			
PD7		Port	U2Tx					PhB0	WT5CCP1	NMI		
PE0	Ain3	Port	U7Rx									
PE1	Ain2	Port	U7Tx									
PE2	Ain1	Port										
PE3	Ain0	Port										
PE4	Ain9	Port	U5Rx		I$_2$C2SCL	M0PWM4	M1PWM2			CAN0Rx		
PE5	Ain8	Port	U5Tx		I$_2$C2SDA	M0PWM5	M1PWM3			CAN0Tx		
PF0		Port	U1RTS	SSI1Rx	CAN0Rx		M1PWM4	PhA0	T0CCP0	NMI	C0o	
PF1		Port	U1CTS	SSI1Tx			M1PWM5	PhB0	T0CCP1		C1o	TRD1
PF2		Port		SSI1Clk		M0Fault0	M1PWM6		T1CCP0			TRD0
PF3		Port		SSI1Fss	CAN0Tx		M1PWM7		T1CCP1			TRCLK
PF4		Port					M1Fault0	IDX0	T2CCP0	USB0epen		

LaunchPad Board



Notice R9 and R10

PWM Channels

- Use PWM channel
  - Choose PWM outputs
  - Runs at 16-bit precision
  - Fix the period (10 times faster than time constant)
  - Prescaled clock determines resolution
    - high+low sets the precision
    - Choose as large as possible (prescale as low as possible)
- Example
  - 2 ms period, bus clock = 80 MHz
  - Prescale divide by 2, so clocks at 40 MHz, i.e. 25ns
  - high+low= 50000
  - Precision is 50000 alternatives or 16 bits
  - Duty cycle range is 0 to 100%
  - Duty cycle resolution is 100%/50000= 0.002%

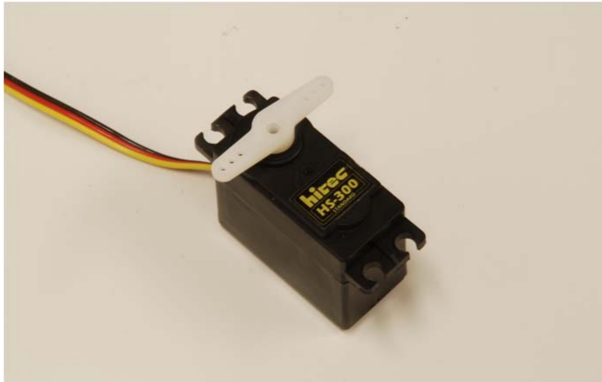
16-Bit PWM Output

```
// period is 16-bit number of PWM clock cycles in one period (3<=period)
// duty is number of PWM clock cycles output is high  (2<=duty<=period-1)
// PWM clock rate = processor clock rate/SYSCTL_RCC_PWMDIV
//                = BusClock/2 (in this example)
void PWM0_Init(uint16_t period, uint16_t duty){
  volatile uint32_t delay;
  SYSCTL_RCGCPWM_R |= 0x0001;      // 1)activate PWM
  SYSCTL_RCGCGPIO_R |= 0x0020;     // 2)activate port F
  delay = SYSCTL_RCGCGPIO_R;       // allow time to finish activating
  GPIO_PORTF_AFSEL_R |= 0x01;           //  enable alt funct on PF0
  SYSCTL_RCC_R |= SYSCTL_RCC_USEPWMDIV; // 3) use PWM divider
  SYSCTL_RCC_R &= ~SYSCTL_RCC_PWMDIV_M; //    clear PWM divider field
  SYSCTL_RCC_R += SYSCTL_RCC_PWMDIV_2;  //    configure for /2 divider
  PWM_0_CTL_R = 0;                 // 4) re-loading mode
  PWM_0_GENA_R = (PWM_X_GENA_ACTCMPAD_ONE|PWM_X_GENA_ACTLOAD_ZERO);
  PWM_0_LOAD_R = period - 1;       // 5) cycles needed to count down to 0
  PWM_0_CMPA_R = duty - 1;         // 6) count value when output rises
  PWM_0_CTL_R |= PWM_X_CTL_ENABLE; // 7) start PWM0
  PWM_ENABLE_R |= PWM_ENABLE_PWM0EN;    // enable PWM0
}
void PWM0_Duty(uint16_t duty){
  PWM_0_CMPA_R = duty - 1;         // 6) count value when output rises
}
```

PWM_4C123.zip
PWMDual_4C123.zip

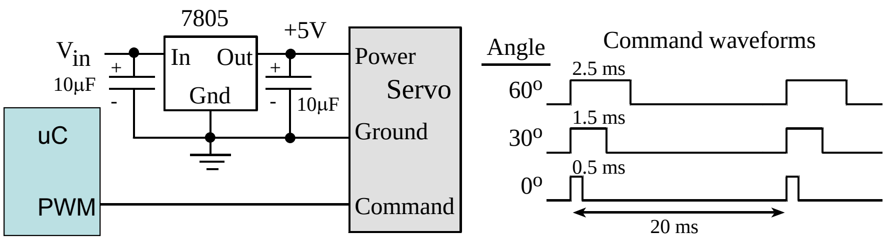
Servo Motor

- Simple digital interface (built in controller)
- Duty cycle controls angle



Servo Interface

- Needs its own +5V regulator
- Duty cycle controls angle

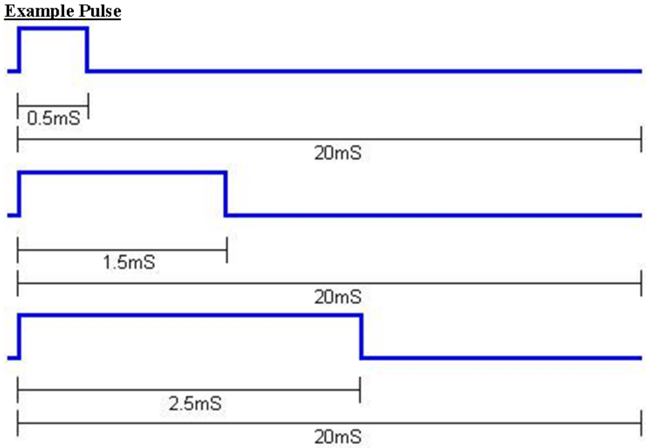


Servo Software

- Duty cycle controls angle

Position	Pulse Width
Minimum	0.5ms
Center	1.5ms
Maximum	2.5ms

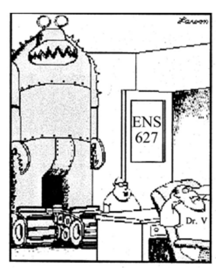
Example Pulse



Lecture 9 J. Valvano, A. Gerstlauer EE445M/EE380L.6 29

Robot Interfacing (Lab 6)

- Power design kit
  - Protoboard
  - Connector for battery
  - 7805 regulator
  - Socket for L293
  - Eight diodes
  - Two motor connectors (0.156in header)
  - Two 4.7uF electrolytic capacitors



"My project's ready for grading, Dr. Big Nose... Hey! ... I'm talking to you, squid brain!"

Lab6.sch

Lecture 9 J. Valvano, A. Gerstlauer EE445M/EE380L.6 30

Summary

- Be careful of the currents
- Sensors are noisy
- Time lag makes it unstable
- Component testing
- Visualization and control