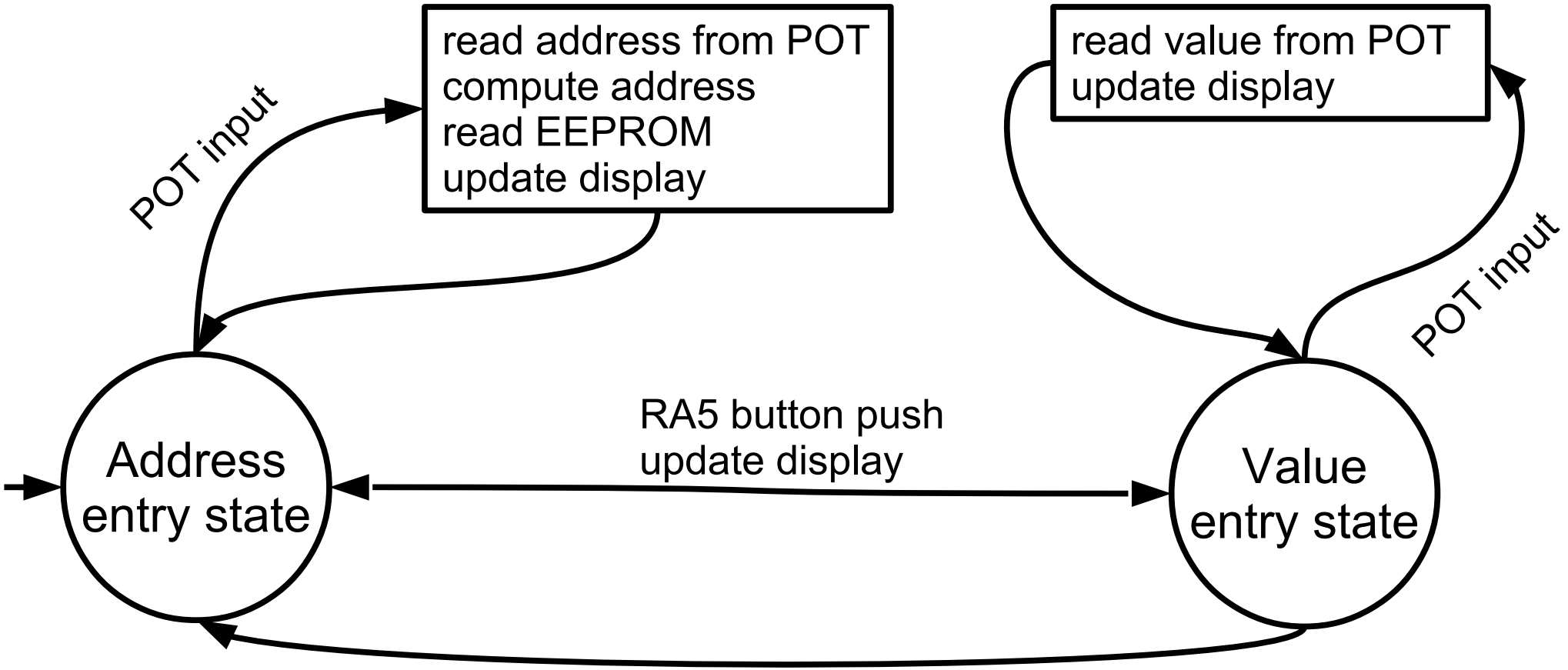
Lab 6 & 7

- Who has hearing sensitivity?
- Let me know if you are having issues with the sound
- I've uploaded a DEM 2 piece of sample code
  - Should be able to get to work with the LCD library
  - Simply reads the pot
  - Scales the value and generates sound
  - Outputs to the LCD

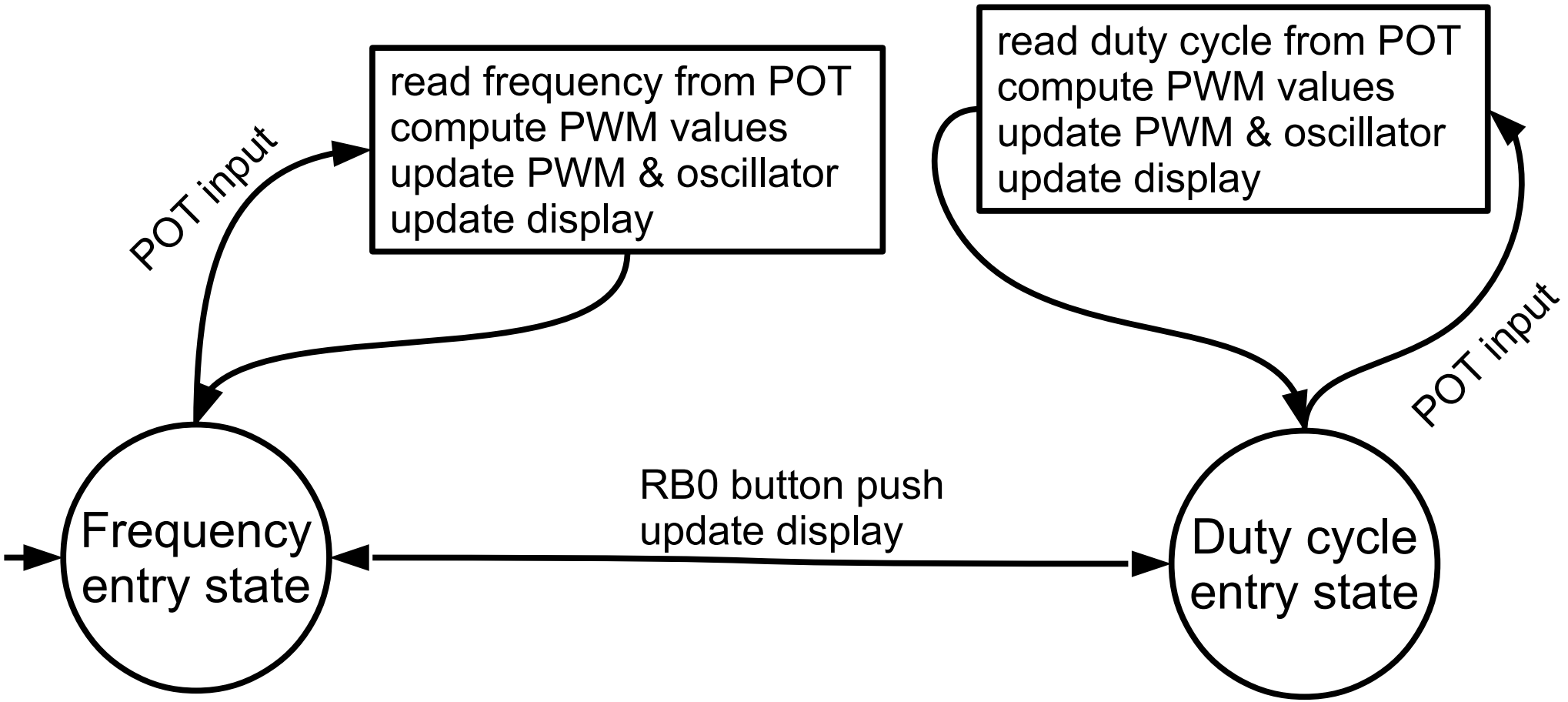
Lab 5 States & Transitions
(PIC18 Explorer Board)



Lab 5 Additional Information

- Have a 16 bit unsigned base address that is added to the 8 bit address read from the POT
  - #define BASE_EEPROM_ADDRESS 0x7F00
  - Allows us to access the full EEPROM address space
- 8MH oscillator should work
- For the POT input
  - Make left justified
  - Just read the high 8 bits (ADRESH)
- When initializing the EEPROM
  - Make sure the port directions are correct

Lab 6 States & Transitions



Lab 6 Additional Information

- I used 250kHz – 16,000kHz clock frequencies
- Limit frequency to [20Hz, 800Hz]
- Remember floating point versus integer division
  - I've seen this many times cause unhappiness
- Use CCP1 & Timer6 (I used prescale of 16 for all)
- At the lower oscillator speeds, my code runs pretty slow (16MHz versus 250kHz is 64:1)
- Compute PWM period in seconds
  - 1/frequency (frequency in Hz)
- Determine oscillator frequency
  - Find largest oscillator frequency that has a max period greater than or equal to the PWM period

Lab 6 Additional Information

- Compute the pulse width in seconds
  - Pulse width = period in seconds*(duty cycle/100)
- Compute period and pulse width values for PWM
  - Tuesday slides have the formulas
- Set PR6 and 8 MSBs of CCPR1L
  - Period and pulse width
- Set pulse width 2 LSBs
  - (PW floating point value – integer value)/0.25
  - Value should be 00, 01, 10, or 11
  - Set CCP1CONbits.DC1B
- Update the display

Lab 6 Additional Information

- Setting RC2 to input, instead of output, should allow you to exercise your code without generating sound (or pull jumper J9)
- Equations from data sheet
  - 14-1 gives PRx setting (period)
  - 14-2 gives CCPRxL setting (pulse width)
  - 14-3 gives duty cycle

Equation 14-1

$$PWM\ Period = [(PRx) + 1] \bullet 4 \bullet TOSC \bullet (TMRx\ Prescale\ Value)$$

Note 1: Tosc = 1/Fosc

- Fosc = 8MHz
- PWM frequency = 20kHz
  - PWM period = 0.00005 seconds (1/20000)
- Prescale 1
- 0.00005 = [PRx + 1]*4*(1/8000000)*1
  - PRx = ((8000000*0.00005)/4) – 1 (solve for PRx)
- PRx = 99

From Microchip PIC18(L)F2X/4XK22 data sheet

Equation 14-2

$$\text{Pulse Width} = (CCPRxL{:}CCPxCON\langle 5{:}4\rangle) \bullet T_{OSC} \bullet (\text{TMRx Prescale Value})$$

- Fosc = 8MHz
- Pulse width = 0.000025 seconds (1/2 of period)
- Prescale 1
- 0.000025 = (50*4)*(1/8000000)*1
- CCPxCON = 00 is equivalent to multiplying by 4
  - The ":" in the equation is simulating 2 digit floating point math since the 4 (clock to instruction) factor is missing

From Microchip PIC18(L)F2X/4XK22 data sheet

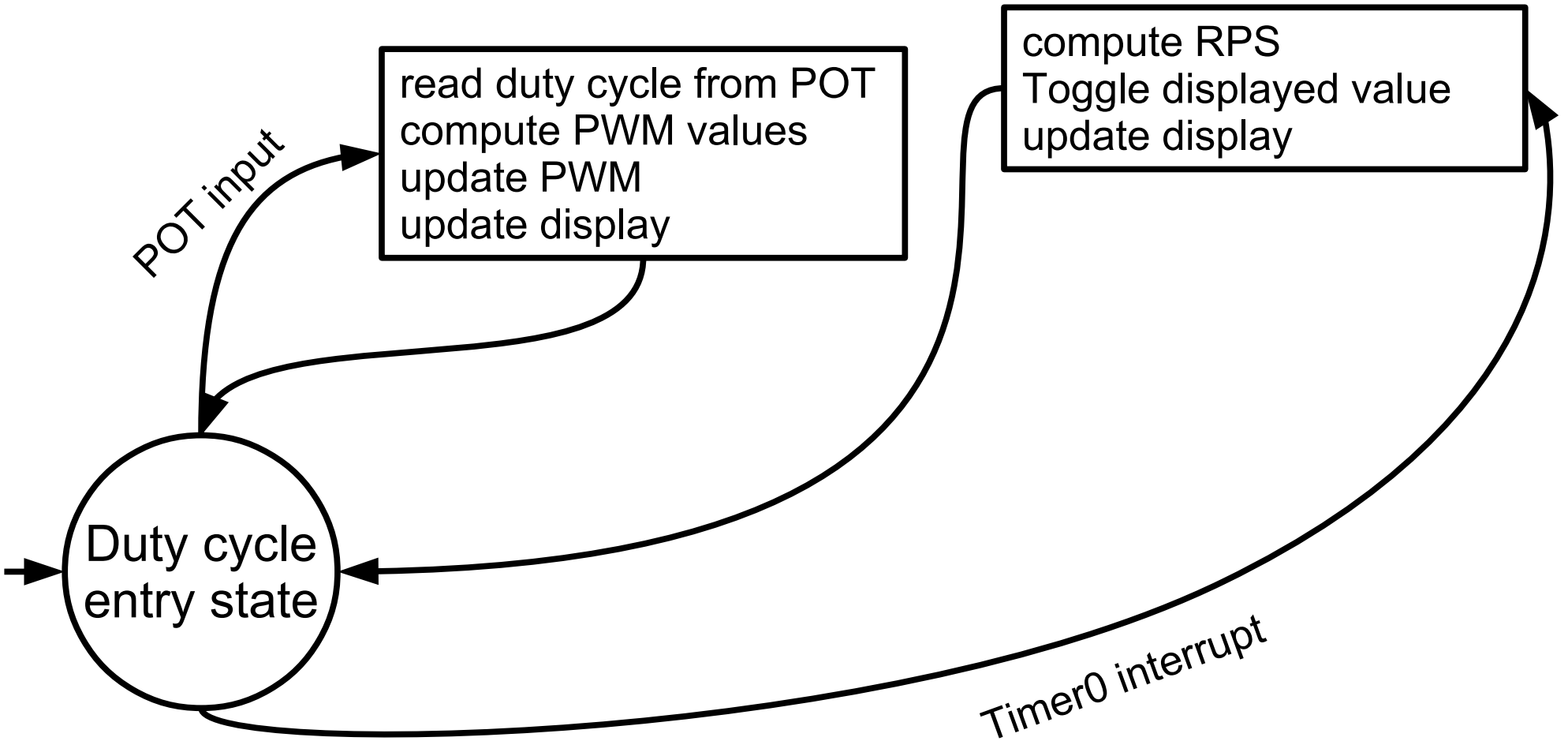
Equation 14-3

$$\text{Duty Cycle Ratio} = \frac{(CCPRxL:CCPxCON<5:4>)}{4(PRx + 1)}$$

- CCPRxL = 50
- CCPxCON = 00
- PRx = 99
- 0.5 = (50*4)/(4*[99 + 1])

From Microchip PIC18(L)F2X/4XK22 data sheet

Lab 7 States & Transitions



Lab 7 Additional Information

- Use CCP2 and Timer2 for PWM
- Use Timer0 for 1 second interrupt
  - Compute RPS
  - Toggle between RPS & duty cycle display
- Use Timer1 to get optical interrupter count
- Setup the pins
  - RD7 & RD2 are outputs
  - RC5 and RA0 are inputs
- Set PR6 to 99 (see lab 6 PWM slides)
  - PWM period of approximately 20kHz
- Port B pullups disabled

Lab 7 Additional Information

- Enable interrupts and interrupts for Timer0
- In main loop
  - Get duty cycle from POT
  - If current pot value != last pot value
    - Update CCPR2L
- Timer0 is 8 bit
  - Will need global counter to get 1 second
- ISR
  - Update display
  - If end of 1 second window
    - Compute RPS
    - Toggle output to display

Lab 6 Table

pulse frequency (Hz)	pulse period (ms)	achieved pulse period (ms)	pulse width (ms)	oscillator frequency (MHz)	PRx	CCPRxL: CCPxCON<5:4>	CCPRxL	CCPxCON<5:4>	timer prescale	duty cycle ([0,100])	period error (%)	pulse width error (%)
20	50.000	49.920	25.000	0.25	194	390	97	2	16	50	0.16	0.16
50										43		
100										30		
257										40		
411	2.433	2.432	0.657	4	151	164	41	0	16	27	0.04	0.14
530										55		
650										60		
722										77		
870										32		
950										25		
1,000										51		
20,000										53		
											< 0.48	< 0.50

The table should be on blackboard

Lab 6 & Lab 7

- Same partner for both
- New partner that you have never worked with in this class
- Any volunteers for working alone?
- Filling out the table should validate your math is correct
- We'll figure out next week if we will have a lab 8 or how we will handle it
- Lab 6 & 7 due next Thursday
  - If you are working on lab 6 & no DEM 2 board available, the display and computations should work on the PIC18
  - Lab 8 may help alleviate a hardware shortage