

ECE 376 - Final Exam: Name _____

Open-Book, Open Note, Calculators and Matlab permitted. Individual Effort.

1) Binary Outputs: Assume a 6411 NPN transistor (if needed)

- $V_{be} = 0.7V$
- $V_{ce(sat)} = 0.2V$
- $\beta = 300$
- $\max(I_c) = 6A$

1a) Give a circuit which allows a PIC to turn on and off a 40mW LED at 2mA

- $I_d = 20mA$
- $V_d = 2.0V$

1b) Give a circuit which allows a PIC to turn on and off 12V DC motor, which draws up to 3A @ 12V

- $V(\text{motor}) = 12V$
- Current draw $< 3A$

2) Analog Inputs: A CdS light sensor has the following resistance
 - lux (light intensity) relationship

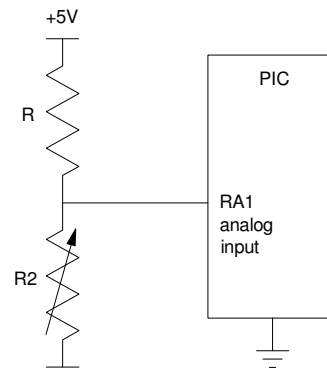
$$R_2 = 1000 \cdot \left(\frac{10}{Lux} \right)^{0.6} \text{ Ohms}$$

If the A/D reading is 417, determine

- The voltage,
- The resistance,
- The light level in Lux, and
- The resolution (the smallest change in Lux you can detect)

with the following circuit. Assume

- $R = 900 + 100 \cdot (\text{your birth month}) + (\text{your birth date})$



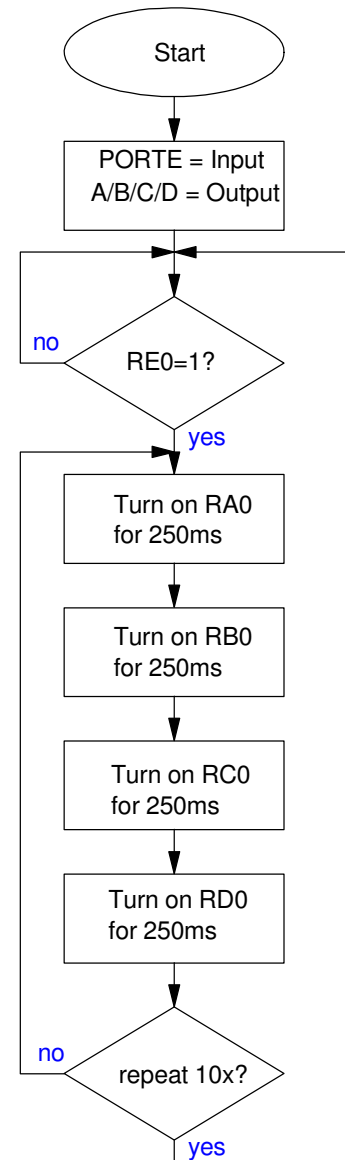
R 900 + 100*mo + day	A/D reading 0..1023	Voltage A/D = 712	R A/D = 712	Resolution smallest change in Lux you can measure
	712			

3) C-Coding **without** interrupts: Write a C program for driving Christmas tree lights. Assume each output pin is connected to an LED. When pin RE0 goes high, the LEDs are to be turned on one at a time in the following sequence

- RA0 - RB0 - RC0 - RD0 - repeat ten times

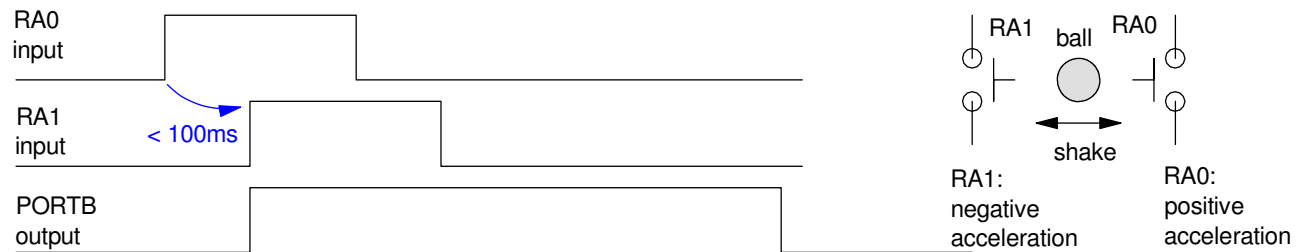
with a 250ms delay between each light. Write the corresponding C code

```
void main(void) {  
    ADCON1 = 0x0F;  
}
```



4) C Coding without interrupts: Christmas Present: Write a C program that turns on lights if the PIC is shaken back and forth in less than 100ms.

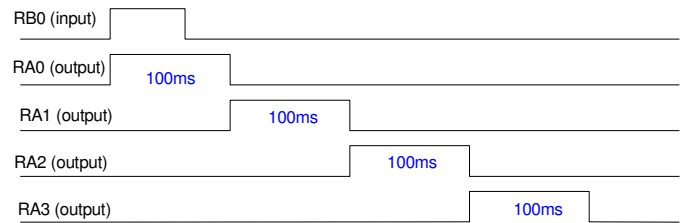
- RA0: 5V if the PIC board experiences positive acceleration
- RA1: 5V if the PIC board experiences negative acceneration
- PORTB: Connected to eight LEDs



```
void main(void) {  
    ADCON1 = 0x0F;  
}
```

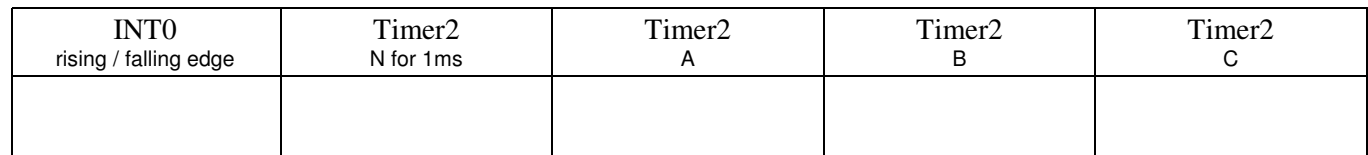
5) C-Coding **with** interrupts: Write a C which uses interrupts to turn on the lighs for a Christmas tree:

- When pin RB0 goes high (INT0 interrupt)
- Pins RA0, RA1, RA2, then RA3 go high sequentially
- Each pin goes high for 100ms (controlled by Timer0 interrupt)
- Once RA3 turns off, the process stops until the next INT0 interrupt



INT0 initialization rising or falling edge	Timer0 Initialization Pre-scalar = ?
INT0 Interrupt Service Routine start the light show when RB0 goes high	Timer0 Interrupt Service Routine Trigger every 100ms Turn on RA0 then RA1 then RA2 then RA3 three times then stop (until next INT0 iinterrupt)
<pre>if (INT0IF) {</pre>	<pre>if (TMR1IF) {</pre>

- If RB0 goes high three times in less than 500ms
- PORTC goes high (0xFF) for one second



Main Loop if needed	INT0 Interrupt Count edges On 1st edge, start 500ms counter If 3 edges in < 500ms, turn on PORTC for 1sec	Timer2 every 1ms
while(1) {	if (INT0IF) {	if (TMR2IF) {