

Name _____

CM _____ Section _____

ME430 - Mechatronics

Examination II

February 3, 2009

Problem	Score	Instructor Check Off
1	/ 25	
2	/ 25	
3	/ 25	
4	/ 25	
Total	/100	

Show all work for credit

You may use your computer, the class website, and any class notes.

You may not contact other humans.
(If you can contact aliens, go for it.)

When you finish a problem call your instructor over to check you off on the front page.

For all problems:

The only code in the program should be the code which is necessary to accomplish the task—points will be deducted if there is extra stuff that we need to sort through. Before the end of the test, place the code in the ANGEL drop box.

Problem 1 – PWM Piezo Buzzer

Starting from “template.c”, create an MPLAB/C18 program that uses a PWM to create three sequential one-second tones on the buzzer. The first one-second tone should be at 250 Hz, the second tone should be at 750 Hz, and the third tone should be at 500 Hz. All tones should use a 10% duty cycle. After the three tones play, the buzzer should go silent.

Problem 2 – ADC

Starting from “template.c”, create an MPLAB/C18 program that uses the potentiometer to move a light back and forth on the four LEDs. The program should continuously update the lit LED location based on the potentiometer position. (Use the ADC to read the value of the potentiometer.)

The reading from the potentiometer will range from 0 to 1023. If the reading is between 0 and $1023/4$ turn on light RB3 (the leftmost light). If the reading is between $1023/4$ and $2*1023/4$, turn on RB2, and so forth. This will make the light appear to “follow” the potentiometer as you rotate it.

Problem 3 – Pushbutton Interrupts

Starting from “template with interrupts.c”, create an MPLAB/C18 program that uses interrupts to count how many times we push the button RB0. Store the counter as a global variable, and set up the watch window to display the variable.

Problem 4 – Timer interrupts

Starting from “template with interrupts.c”, create an MPLAB/C18 program that uses Timer 0 interrupts to count in seconds on the LCD screen. (Note that you are required to use Timer 0 interrupts for credit on this part.)

When the program starts the LCD should be blank. After 1 second, it should put a “1” on the LCD screen. After 2 seconds, it should replace the “1” with a “2”, then after 3 seconds it should say “3”, and so forth. You don’t need to worry about what happens after 9 seconds.