CSCE 236 Embedded Systems, Spring 2019
Homework 3

Due: Thursday, Feb 14th, 2019 (start of class)

Instructions: This homework is an individual assignment, collaboration is not allowed. Show your work and describe your reasoning to get partial credit if your solution is incorrect. Unless otherwise specified, assume problems refer to the Arduino board we are using. This assignment is out of 100 points, but is equally weighted with other homework assignments.

Name: _____

Problem 1. *(5 pts) (To be completed at end of assignment) Approximately how much time did the total assignment take? Which problem took longest and how much time did it take? List any resources you used to complete this assignment (e.g. websites, datasheets, office hours, discussions with others, etc.). Be specific and if you did not use any other resources include the statement "I did not use outside resources for this assignment."*

Problem 2. Schematics

a). *(5 pts) In Lab 1, Figure 1, the circuit for connecting the LEDs is shown. The LEDs "drop" a fixed voltage and the resistors serve to limit the amount of current that flows through the LEDs. The drops for each of these are RED 2.0V, GREEN 3.2V, and BLUE 3.2V*[1]*. How much current flows through each of these resistors (and therefore the same amount flows through the corresponding LED, which determines the brightness)?*

b). *(5 pts) If nothing else is using any current, how many LEDs could be controlled by processor? Hint: Look at how much overall current the processor can supply/sink.*

[1] As an example, for the BLUE LED, which drops 3.2V the voltage across the corresponding resistor will be 1.8V (since 1.8V + 3.2V = 5.0).

c). *(5 pts) Draw a schematic showing how you would wire the three LEDs (RGB) to a single output pin while keeping the brightness of each individual LED the same as in the original case (assuming in this case that the pin can supply sufficient current).*

Problem 3. Timers and PWM. For this problem assume a CPU frequency of 18MHz.

a). *(10 pts) Write the C code to configure the registers (e.g.* ***`TCCR0A`****, etc.) to set the 8-bit Timer0 in Fast PWM mode with a frequency as close to 500Hz as possible. Comment each line of code to indicate how you are configuring it. Remember to assume a 18MHz clock frequency. Hint: You should try all of the possible clock prescalers to determine which gives you the best value.*

b). *(5 pts) What is the actual frequency that the timer will run at?*

c). *(5 pts) If you used Timer1 instead, could you get closer to an actual frequency of 500Hz? Configure the registers for Timer1 for 500Hz and make sure to show and comment your code. What is the actual frequency you achieve with your configuration?*

d). *(5 pts) Describe the differences between Fast PWM and Phase Correct PWM mode and why you may want to use Phase Correct PWM in some circumstances. You may want to include a picture of the signals to help with your description.*

Problem 4. Input

a). *(10 pts) In Lab 1, you configured the button by using a pullup resistor. However, as you learned in class, the input pins also have the ability to activate an internal pullup resistor. Write the C code (setting registers DDRx, etc) below to activate the internal pullup resistor of pin PC2. Also give the "Arduino" version using functions we discussed in class to set pins. Instructor sign off required: You should also implement this and show the instructor the functionality of using your button in Problem 5 without the resistor.*

Instructor sign off: _____

Problem 5.

For this problem you should complete the sections in morse.c where STUDENT CODE is indicated. To do this problem, you will need to include morse.c and morse.h in your sketch (use the menu Sketch->Add File. To call functions from morse.c, you will need to put #include "morse.h" in your main sketch file (note that it should be in quotes, not in < >).

Complete the morse.c code so you will be able to send Morse code blinks. You should make sure that you are able to specify any of LED_RED, LED_GREEN, or LED_BLUE as the LED to output blinks or any combination of them (e.g. LED_BLUE | LED_RED). I would recommend creating helper functions that turn on or off LEDs so if you switch the pin that controls the LEDs, you only have to change code in one or two places. All of the code described here should be in a single program that runs at the same time.

You must also turn in your code on Canvas. *Failing to electronically turn in your code will result in a 10 point penalty on this assignment. Points may also be deducted for no header, coding errors, poor style, or poor commenting. Note,* ***also need to turn in a printout of your code for this assignment****.*

a). *(5 pts) In morse.c, the Morse blink pattern (dots and dashes) for each character are stored in a single byte. Read the code and describe how this is done and what the meaning of each bit is.*

b). *(10 pts) Now write the C code to turn on and off the LEDs by setting the registers (e.g. DDRx, etc.). What pins did you connect the LEDs to? How did you configure these pins as output? How do you turn the LED on and off? Make sure to include the relevant code here.*

c). *(10 pts) Write code that will blink "Hi" when the board starts with "H" blinked with the green LED and "i" blinked using the red LED. Instructor sign off required, no written answer needed.*

Instructor sign off: _____

d). *(10 pts) Now implement code that will output "d" (for dot), "D" (for dash), or "s" (for a long pause) over the serial port depending on how long the button is pressed. Use fixed times for your dot, dash, and long pauses (e.g. any press under a second is a dot and anything over a second is a dash and any pause over a second is a long pause). Note that bouncing buttons can cause problems, so beware. Describe how you implemented this. Instructor sign off and written answer required.*

Instructor sign off: _____

e). *(10 pts) Finally, implement code that will turn on the Red, Green, or Blue led if the Morse code for 'r', 'g', or 'b' is entered, respectively. Keep the LED on for approximately 1 second and then turn it back off. Instructor sign off required, no written answer needed.*

Instructor sign off: _____

Do not forget to fill in the amount of time you spent on this assignment and resources you used in Question 1.