EE 285 Lab 12

This lab focuses on using the Arduino as an embedded platform. The purpose of these simple "LED-blinking" exercises is to become familiar with using the Arduino and connecting simple components rather than looking at any specific programming skills.

Use the class notes and the Arduino programs (sketches) on the GitHub as guides to help implement the exercises described below.

The Arduino software reference page — https://www.arduino.cc/en/Reference/HomePage — might be useful in learning the details of the commands that will needed in the following exercises.

There is no report required for this lab. Demonstrate each section to your lab instructor. Each section is worth 25 points — the entire lab is worth 100 points.

A. Setting up

1. Obtain an Arduino board and a USB cable from the instructor. (Or use your own, if you've brought one.) Connect the Arduino to your computer.
2. Start up the Arduino software on the computer.
3. Bring up the basic "Blink" example program and download it to the Arduino. Note the operation of the program on the Arduino.
4. Change the program to increase the frequency of the blinking. Determine the highest blink rate that you can detect with your eye. As you increase the frequency, keep the LED "on" and "off" times equal.

B. External LEDs

Use the solderless breadboard to wire up one red LED and one green LED to pins 5 and 6. Include current limiting resistors in series with the LEDs and don't forget to make the ground connection between the Arduino and the LEDs on the board.

In writing the programs below, you can use the Arduino `delay()` function or you can use the improved timer approach as described in class. Of course, you must initialize everything in the setup portion of the program.

1. Write a program that will blink the red LED 10 times (1 blink = on for 0.5 s and off for 0.5 s) and then blink green LED once (on for one second). The total sequence cycles repeatedly.
2. Modify the program so that the red LED blinks 5 times (same on/off times) and the the green LED blinks 5 times (on for 0.25 seconds then off for 0.25 seconds). The sequence cycles repeatedly.

C. Poor man's pulse-width modulation.

Use the solderless breadboard to wire up one LED (red or green, your choice) to pin 5. Include a current limiting resistor and the ground connection.

Write a program that:

1. Does 1000 blinks with where the LED is on for 1 ms and off for 4 ms.
2. Does 1000 blinks with where the LED is on for 2 ms and off for 3 ms.
3. Does 1000 blinks with where the LED is on for 3 ms and off for 1 ms.
4. Does 1000 blinks with where the LED is on for 4 ms and off for 1 ms.
5. The sequence cycles repeatedly.

One loop will last 20 seconds (4 * 5 ms * 1000 = 20 s). Note that the LED will be blinking on and off so fast that you will not see individual blinks with your eye. Instead, you will see many periods averaged together.

Again, you can use the Arduino `delay()` function or you can use the improved timer approach as described in class. Obviously, you will need to make good use of loops to implement this program. What do you see happening with the LED as the on/off times change?

D. Analog input

Connect a potentiometer to the Arduino, One outside pin connects to 5 V, the other outside pin connects to ground, and then connect the center pin (the wiper) to A0 — analog input 0. (Your lab instructors can help you set up potentiometer, if needed.) As the potentiometer setting is adjusted, the voltage on the center pin will vary between 0 and 5 V.

1. Write a program that will measure and print out the voltage level of center pin of the potentiometer. You will need to use `analogRead(0)` to get the analog voltage on pin 0. Recall that the input voltage will be represented as integer with value between 0 and 1023. IN the program, have the integer printed to the serial monitor once every half-second or so. While the program is running, slowly adjust the potentiometer through its full range and see measured values printed out as you make the adjustments.
2. Modify the above program so that instead of printing out the 10-bit digital number (0 to 1023), the actual voltage level (0 to 5) is printed.
3. Write a program that will vary the blink rate of an LED based on the setting of the potentiometer.

   Use the following prescription:
   — if the potentiometer voltage is less than 1 V, then the rate is 1 sec on and 1 sec off,
   — if the voltage is between 1 V and 2 V, the rate is 0.5 sec on and 0.5 sec off,
   — if the voltage is between 2 V and 3 V, the rate is 0.25 sec on and 0.25 sec off,
   — if the voltage is between 3 V and 4 V, the rate is 0.125 sec on 0.125 sec off, and
   — if the voltage is above 4 V, the rate is 0.062 sec on and 0.062 sec off.

   Demo the operation of this program to your lab instructor.