Lab 2: Analog and Digital Sensors

Due Monday July 31, 2017 @ 11:59pm PT on bCourses

Part 1: Simple Sensors

Exercise 1: Basic Digital Input

Items:

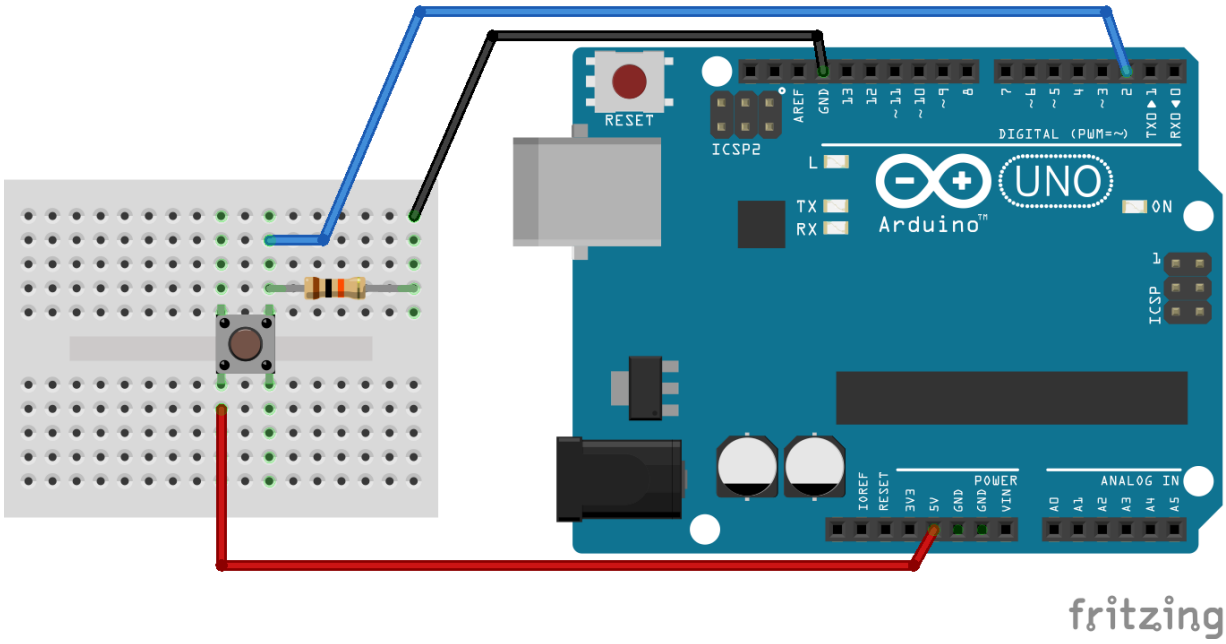
- **1x Arduino Uno Board with USB Cable**
- **1x Switch, Toggle, Pushbutton, or Momentary Switch**
- **1x 10k Resistor (brown-black-orange)**
- **Jumper Wires**

Deliverables:

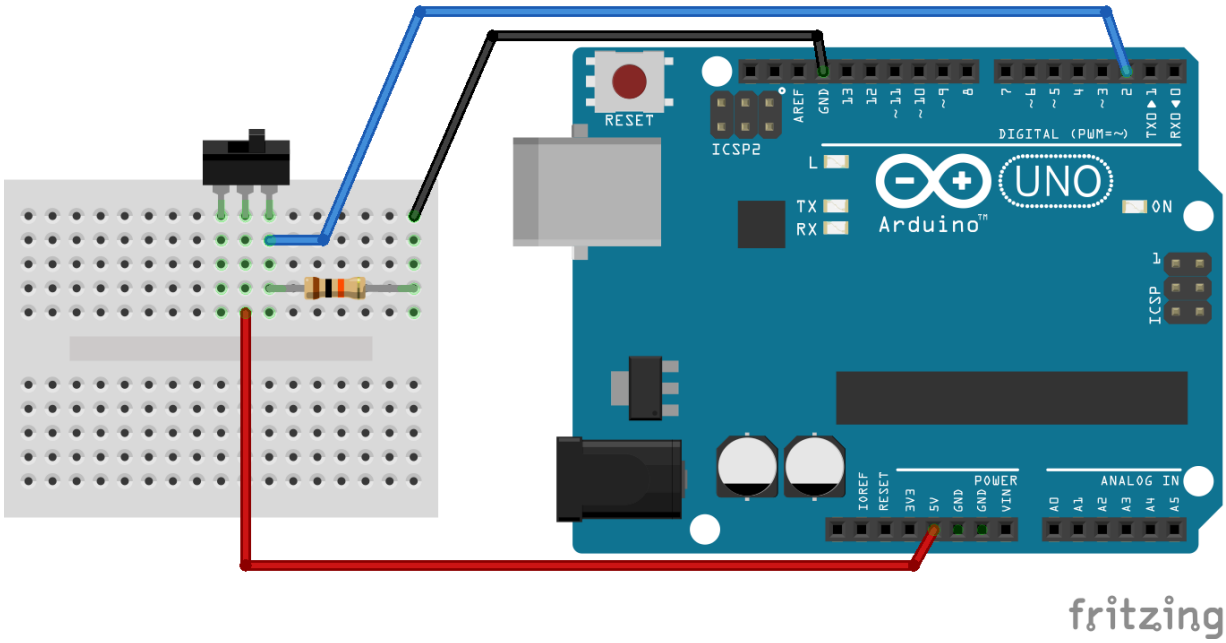
- **None**

Directions:

1. **Using either a button or a toggle switch, setup the circuit shown below.**



If you are using a button or momentary switch, position the button as shown above. Note that the left pins are connected within the button and that the right pins are connected as well. Connect the left (or right) pins of the button to 5v and connect the other pins to pin 2 on the Arduino. Lastly, connect pin 2 to ground through a 10k resistor (This resistor is referred to as a pull-down resistor because when the button is open or removed, the digital signal is pulled to ground).



If you are using a toggle switch, connect the center pin of the switch to 5v and connect the right (or left) pin to pin 2 on the Arduino. Lastly, connect pin 2 to ground through a 10k resistor (This resistor is referred to as a pull-down resistor because when the switch is off or removed, the digital signal is pulled to ground).

2. Open the Arduino Button example (under File->Examples->02.Digital->Button) and upload it to your Arduino board. Note that you may be using a switch instead of a button, but the functionality/code is the same.

Exercise 2: Basic Analog Input

Additional Items:

- 1x 10k Trimpot with Knob

Deliverables:

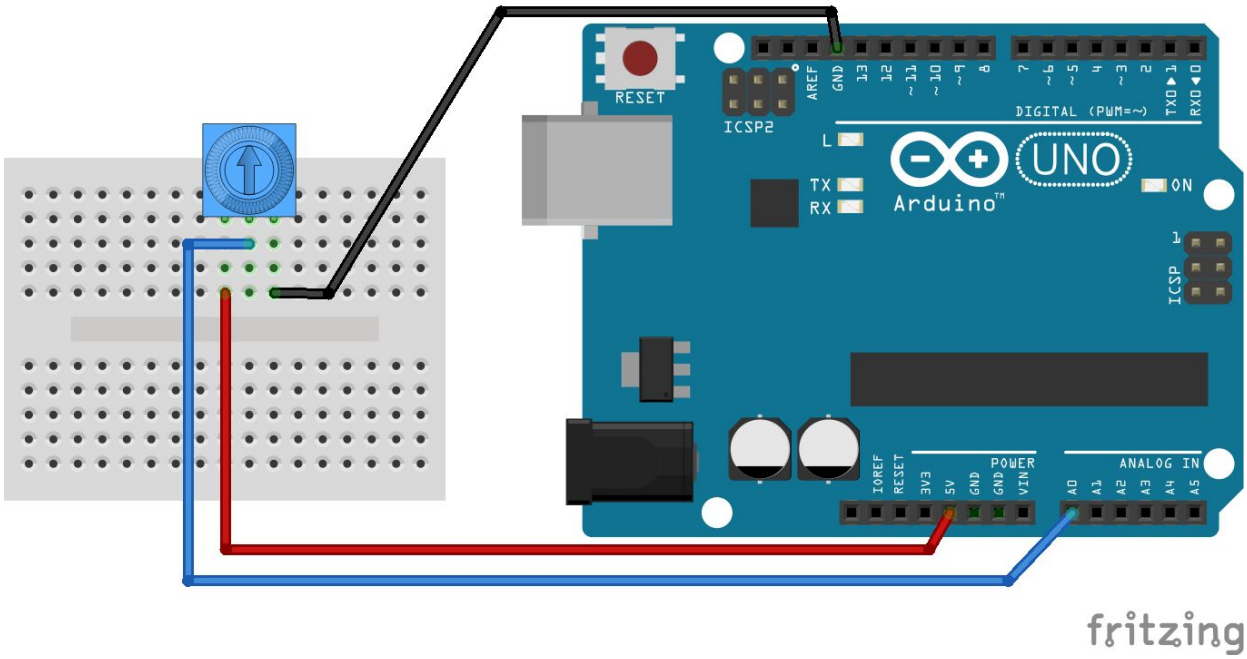
- None

Directions:

1. A trimpot is a trimmable potentiometer, also known as a variable resistor. With a 10k trimpot, you can change the resistance of the device from (around) 0 to 10k ohms by simply turning the

knob. When connected to 5v and ground, this means that we can change the analog output of the trimpot from 5v (0 ohms) to 0v (10k ohms).

2. To use the trimpot, connect the center pin to analog pin A0 and the outside pins to ground and 5v (it does not matter which is which, but switching ground and 5v will change the voltage output), as shown below.



3. Open the Arduino AnalogReadSerial example (under File->Examples->01.Basics-> AnalogReadSerial) and upload it to your Arduino board. Make sure you understand what the program is doing. Open the serial monitor and see how the values change as you adjust the knob of the trimpot. Note that the value 1023 indicates an output voltage of 5v.

Exercise 3: Temperature Sensor

Additional Items:

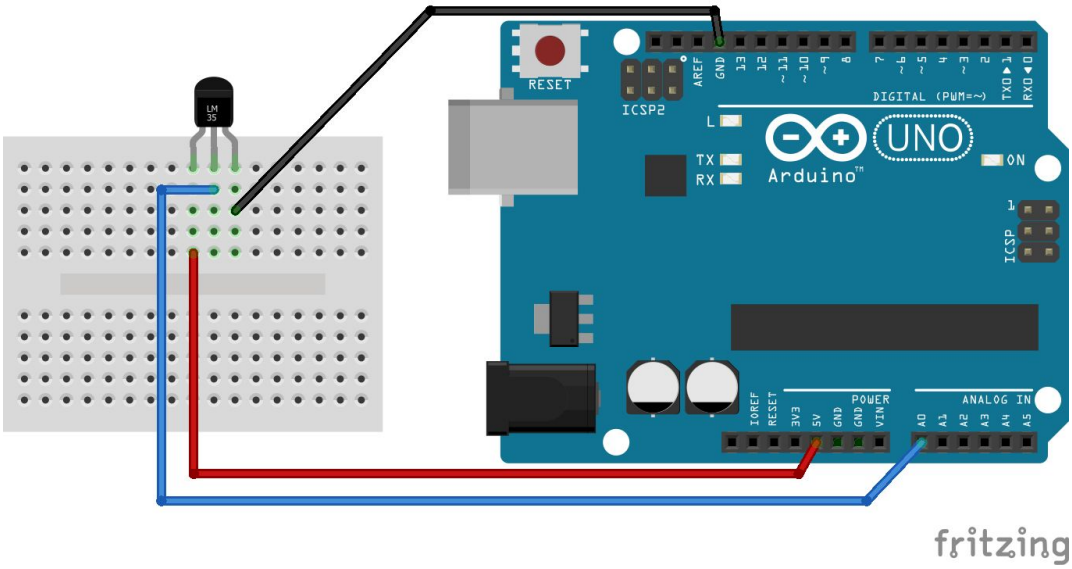
- 1x Temperature Sensor (TMP36)

Deliverables (1.5 pts):

- Submit a file named TemperatureSerial.ino containing the program described below. Additionally, in your lab report, copy & paste your Arduino code into a code box.
- In your lab report, include the formula for converting millivolts to degrees Celsius and your response to Questions.

Directions:

1. Setup the circuit shown below. Connect the center pin to analog in A0, the right pin to ground (or as labeled on the sensor), and the left pin to 5V (or as labeled on the sensor). Be careful not to mix up the left and the right pins or the sensor will quickly become very hot.



2. **Open the Arduino AnalogReadSerial example (under File->Examples->01.Basics-> AnalogReadSerial), change the delay from 1 millisecond to 1 second, save the program as TemperatureSerial, and upload it to your Arduino board.**
3. **Convert the analog reading to mV and output to the Serial Monitor.**
4. **Questions: Read the datasheet at https://www.sparkfun.com/products/10988 for the TMP36.**
    a. **According to the datasheet, what is the "sensitivity" of this sensor?**
    b. **What is the specified temperature range?**
    c. **What is the output (mV) of the sensor at 25 degrees Celsius?**
    d. **What is the maximum output (mV) of the sensor?**
    e. **What is the (typical) accuracy at 25 degrees Celsius?**

    The objective is to gain experience and comfort reading data sheets. This is a critical and practical aspect of any project involving sensors.

5. **Question:**
    a. **Derive a formula in terms of V_{out} to convert the voltage reading to degrees Celsius. Include this formula in your code and in your lap report.**
        i. **Note: The temperature should be calculated as a float.**
        ii. **Hint: Use a linear equation with a point-slope form.**
        iii. **Hint: Do not use -40 or 125 degrees Celsius in your formula.**
        iv. **Hint: 0 mV is NOT 0 degrees Celsius.**
6. **Print the temperature in degrees Celsius and in degrees Fahrenheit to the Serial monitor.**
7. **Questions:**
    a. **According to the sensor, approximately what is the current temperature, in degrees Celsius and in degrees Fahrenheit, in the room?**

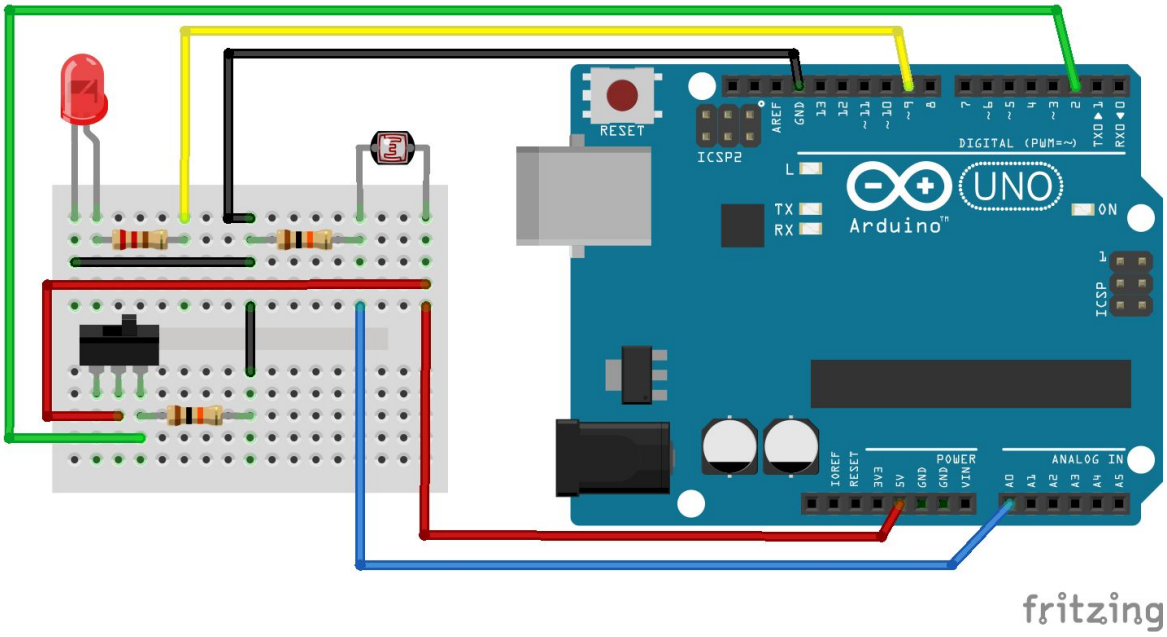
Exercise 4: Putting Everything Together

Deliverables (1.5 pts):

- **Submit a file named LEDSystem.ino containing the program described below. Additionally, in your lab report, copy & paste your Arduino code into a code box.**
- **In your lab report, respond to Questions.**

Directions:

1. **Setup and program a system that fades an LED according to the light intensity but only if the switch/button is on. Your circuit might look something like the image below. Save your program as LEDSystem.ino.**



   a. **Hint: You may find the following helpful.**

```
int mappedVal = map(val, 0, 1023, 0, 255);
```

2. **Questions:**
   a. **What is the procedure (i.e. sequence of events) executed by your system? Explain briefly in your lab report.**
   b. **How many digital inputs, digital outputs, analog inputs, and analog outputs have you used?**