Systems and Industrial Engineering
University of Arizona

SIE 370 Lab

Monitoring and Data Logging System

1. Objectives

In this lab, you will learn how a distance sensor, temperature sensor, and Liquid Crystal Display work as you implement a laboratory room monitoring and data logging system. You will use Tinkercad for the prototyping before continuing to experiment with a prototype using your Arduino, breadboard, and physical components for the Main Lab.

2. Knowledge Required

Reading Assignment:

Arduino Sensor Tutorial: Ping
https://www.arduino.cc/en/Tutorial/Ping

Parallax PING))) Distance Sensor Datasheet
.PDF uploaded to the Lab folder

Arduino.cc Project Hub
LCD Thermometer using TMP36 Sensor
https://create.arduino.cc/projecthub/VJZ/lcd-thermometer-using-tmp36-sensor-283186

Arduino.cc Project Hub
How to connect DHT11 Sensor with Arduino UNO
https://create.arduino.cc/projecthub/pibots555/how-to-connect-dht11-sensor-with-arduino-uno-f4d239

Arduino User Prompt with Validation
SIE 370 How To
https://d2l.arizona.edu/d2l/le/content/1191780/Home

Arduino.cc Reference
Functions | Time | millis()
https://www.arduino.cc/reference/en/language/functions/time/millis/

3. Prototyping Tasks

3.1. **Task 1** Arduino Sensor Tutorial: Ping

Use Tinkercad to create the project described in the Arduino Built-In tutorial for sensors, Ping. Prototype your system with the model of the HC-SR04 that you plan to use in the Lab (i.e., that your kit contains). Please see image below.

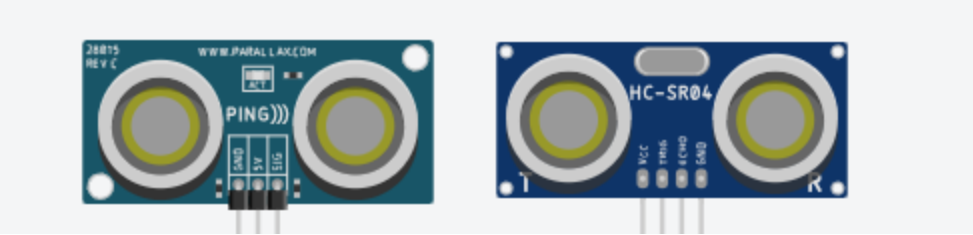

Figure 1 Ultrasonic Distance Sensors available in Tinkercad

3.2. **Task 2** Arduino Project Hub: LCD Thermometer using TMP36 Sensor

Use Tinkercad to create the project described in the Arduino.cc Project Hub LCD Thermometer using TMP36 Sensor (see link above).

3.3. **Task 3** Monitoring and Data Logging System

Re-create the Lab Task Monitoring and Data Logging System Circuit and software on Tinkercad using the instructions for the lab but adapt the instructions for Tinkercad. For example, Tinkercad only has the TMP36 sensor and cannot connect to the Processing App.

Your Tinkercad Serial Monitor output should look something like the following image, since here we are using out Serial.println statements for the carriage return and newline. For the lab, we will instead use Tab-Delimiting.

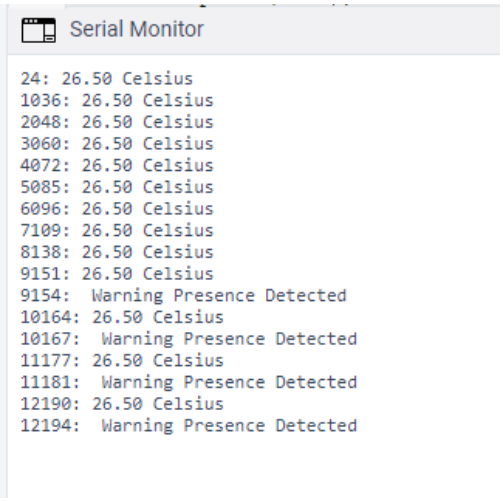

Figure 2 Serial Monitor Output on TInkercad Example

Test your wiring and code here in preparation for Lab and to think of any questions you may be asked during Lab.

Lab Instructor Demonstration: Demonstrate this working prototype before beginning your wiring for the main lab component with your physical Arduino and components.

3.4. Guiding Questions

a) What is the approximate range of distance measurements for our Parallax PING))) sensor?
b) What is the library you have to use for the LCD display?
c) Briefly compare and contrast the TMP36 and DHT11.

4. Lab Instructions

In this experiment, you will use your real Arduino, breadboard, and components to create a prototype for a Monitoring and Data Logging System. The Ultrasonic Distance Sensor will detect the presence of a person up to and including a distance of 6 inches. In addition, the system will read a temperature sensor (TMP36 or DHT11 / DHT22) and display the current temperature on the LCD. Our data will be logged to a file on our computer named output.txt using the Processing Application.

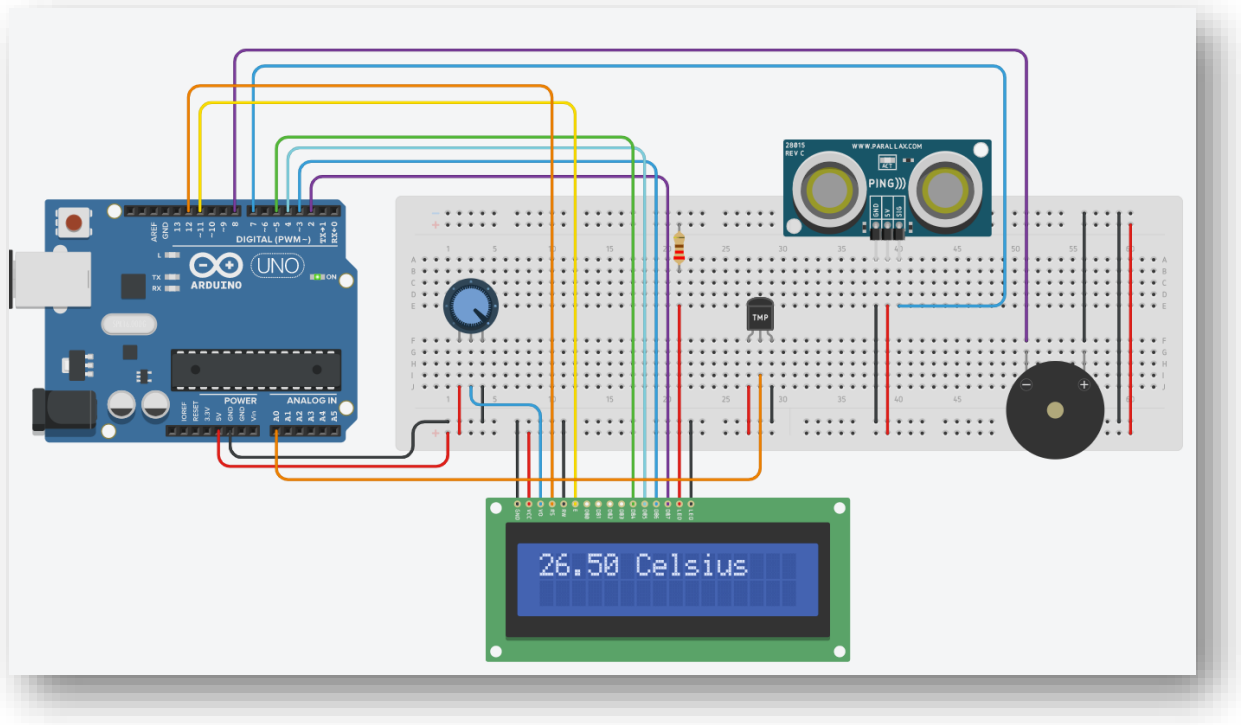

Figure 3 Monitoring and Datalogging System Circuit

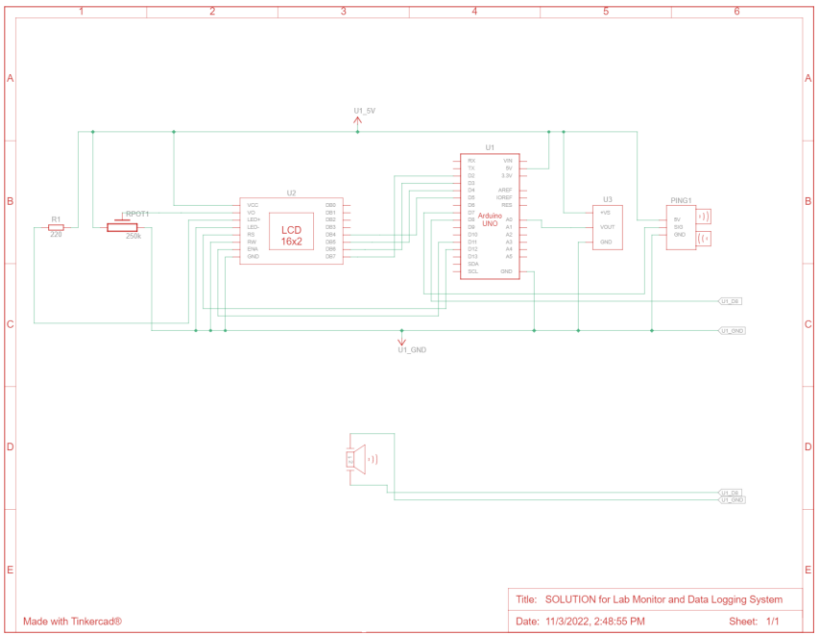

Figure 4 Monitoring and Datalogging Circuit Schematic View

Component List

In addition to your laptop, Arduino, and a breadboard you will need the components in the list below.

Component	Quantity	Arduino Pin
Ultrasonic Distance Sensor	7	7
Piezo Alarm	1	8
LCD	1	LCD(12, 11, 5, 4, 3, 2)
220 Ω Resistor	1	1
Potentiometer	1	LCD Contrast
TMP36 / DHT11	1	TMP36 on Pin A0 DHT11 on Pin 2
Laptop Running Processing	1	Connected to the Serial Communications Port for Arduino

Figure 5 Lab Component List

4.1. **Monitoring and Data Logging System Requirements**

4.1.1. The system shall check every second (1000 ms) to:
- 4.1.1.1. Detect presence (6 inches or less from the sensor for our testing purposes)
  - 4.1.1.1.1. If presence detected, then
    - 4.1.1.1.1.1. The system shall sound the Piezo Buzzer at a frequency of 440 hz. for a duration of 100 ms
    - 4.1.1.1.1.2. The system shall write to the Serial Monitor to log the detection in the format:
      - 4.1.1.1.1.2.1.1. Current milliseconds since program began
      - 4.1.1.1.1.2.1.2. Warning: Presence Detected
- 4.1.1.2. Check and display the current temperature
  - 4.1.1.2.1. Use the proper calculation to convert the temperature to degrees Celsius
  - 4.1.1.2.2. Display the current temperature
    - 4.1.1.2.2.1. In the Serial Monitor
      - 4.1.1.2.2.1.1.1. Current milliseconds since program began
      - 4.1.1.2.2.1.1.2. Degrees Celsius n the first line of the LCD

Observations:

You will want to check the initial output with the Arduino IDE Serial Monitor. Remember that your lines need to be Tab-Delimited (i.e., using Serial.print("\t") instead of Serial.Println()) for the Processing Application output, so your output at this phase with the physical Arduino will look more like this in your Serial Monitor.

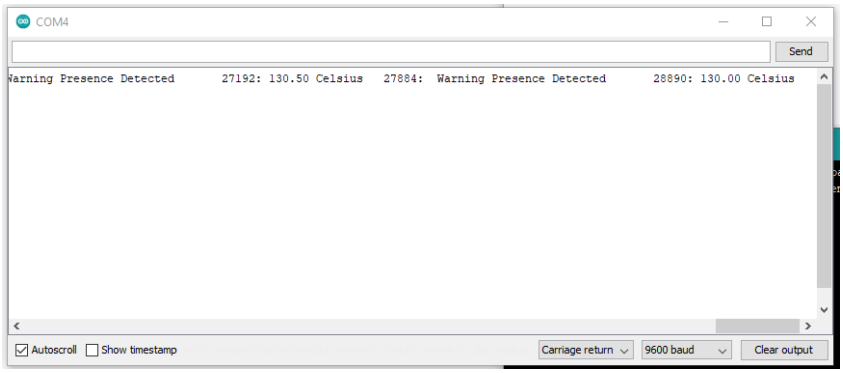

Figure 6 Arduino IDE Serial Monitor Example

Make sure you close your Arduino IDE Serial Monitor before connecting with your Processing Application. Your output in the console of that application will look like the following.

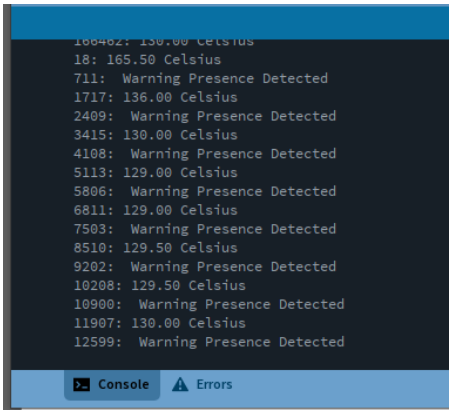

Figure 7 Processing Application Console Output Example

Finally, make sure you close your Processing Application correctly so that it flushes the output to the file and saves it before closing. Your output in output.txt will look like the following.

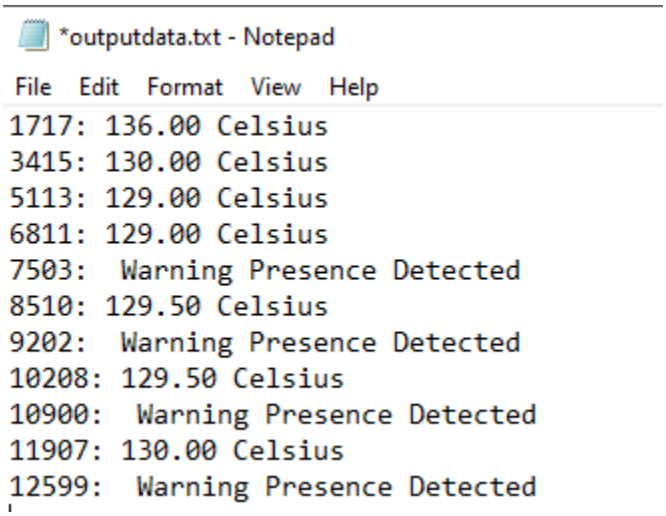

Figure 8 Logging File (output.txt) Content Example

Lab Instructor Sign-Off: Make sure you ask your Lab Instructor to review your work before dismantling the circuit or leaving the room.

Submit: your code for this with your in a text file (ASCII text file) Post Lab Report.

Guiding Questions

4.2. What is the purpose of the potentiometer in this circuit?
4.3. Explain how the Ultrasonic Range Finder works. You may use the illustration in addition to your own wording from the datasheet with attribution.
4.4. What are the steps you had to take in the Processing Sketch:
  4.4.1. To find the correct port number for the Arduino output?
  4.4.2. Write the output to a file and save it on the host computer (your laptop)?