



Lab Worksheet 2

CSE360: Computer Interfacing

Classwork

Task 1

- **Experiment Title: DHT22 sensor utilization**

Objective: To learn how to set up and get results from a DHT22 sensor using an Arduino board.

Materials Needed:

- Arduino board (e.g., Arduino Uno, Elegoo Uno)
- DHT22 sensor
- Jumper wires
- Breadboard

Procedure:

1. Setup the Circuit:

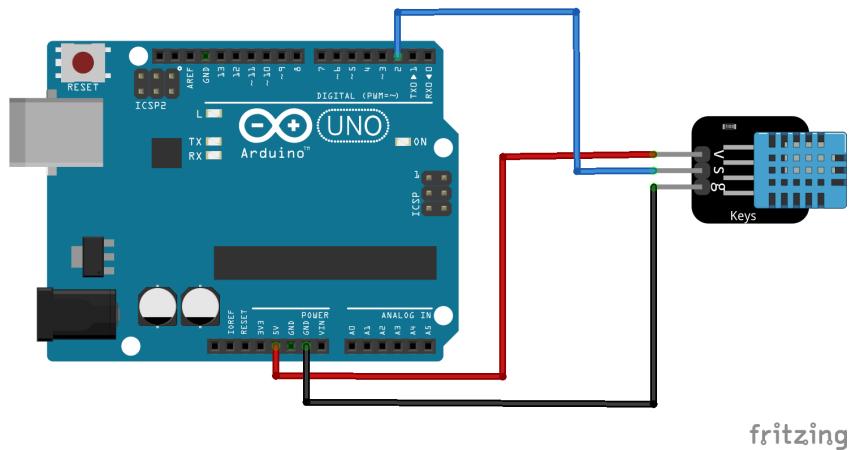
- Connect 3 male to Female wires to the DHT22. It has 3 pins, VCC(5v or 3v), GND(ground) and Data
- Connect the VCC pin with the 5v pin of your arduino
- Connect the GND pin with any GND pin of the arduino board

- Finally select any GPIO pin to connect the data pin. For this tutorial the connected GPIO pin is pin 2.

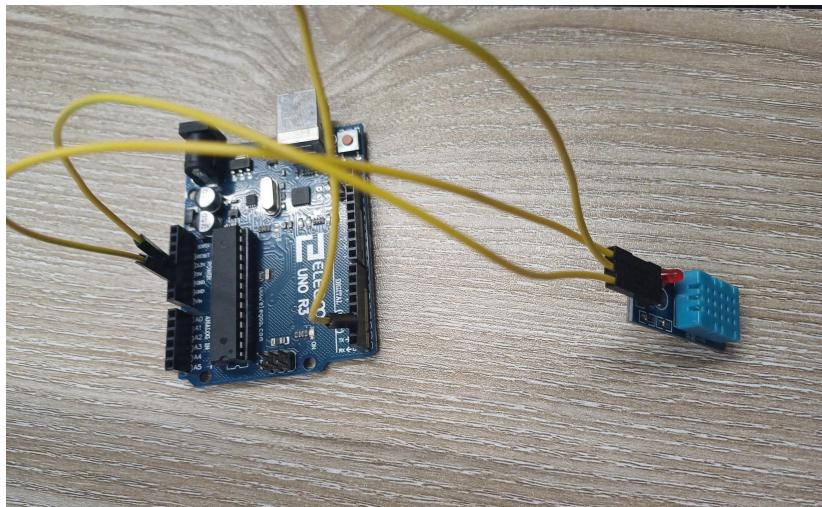
2. Setup the IDE:

- Open a new sketch.
- Connect the arduino
- Open Tools → Port → Select the appropriate port(the one with the arduino uno)
- Again open Tools →manage libraries→ search for **simpleDHT** by **Winlin** → Install
- After installation → File → Examples → Scroll at the bottom and you will find **simpleDHT** → **DHT22Default**
- The entire code will be provided here
- Change the value of serial.begin() to 9600
- Compile and upload.
- Open the serial monitor.

Circuit:



fritzing



Code:

```
#include <SimpleDHT.h>

// for DHT22,
//   VCC: 5V or 3V
//   GND: GND
//   DATA: 2
int pinDHT22 = 2;
SimpleDHT22 dht22(pinDHT22);

void setup() {
  Serial.begin(9600);
}

void loop() {
  // start working...
  Serial.println("=====");
  Serial.println("Sample DHT22...");

  // read without samples.
  // We use read2 to get a float data, such as 10.1*C
  // if user doesn't care about the accurate data, use read to get a byte data, such as
  // 10*C.
  float temperature = 0;
```

```

float humidity = 0;
int err = SimpleDHTErrSuccess;

if ((err = dht22.read2(&temperature, &humidity, NULL)) != SimpleDHTErrSuccess) {
    Serial.print("Read DHT22 failed, err="); Serial.print(SimpleDHTErrCode(err));
    Serial.print(","); Serial.println(SimpleDHTErrDuration(err)); delay(2000);
    return;
}

Serial.print("Sample OK: ");
Serial.print((float)temperature); Serial.print(" *C, ");
Serial.print((float)humidity); Serial.println(" RH%");

// DHT22 sampling rate is 0.5HZ.
delay(2500);
}

```

3. Assemble the Experiment:

- Connect the Arduino to your computer using a USB cable.
- Upload the code to the Arduino board.
- Power up the Arduino.

Analysis: Observe the serial monitor as it shows the humidity and temperature at an interval at which the sensor works.

Communication protocol used: UART

Task 2

- **Experiment Title: Connecting LCD display using an I2C module**

Objective: To learn how to interface a LCD display with an Arduino and display any text.

Materials Needed:

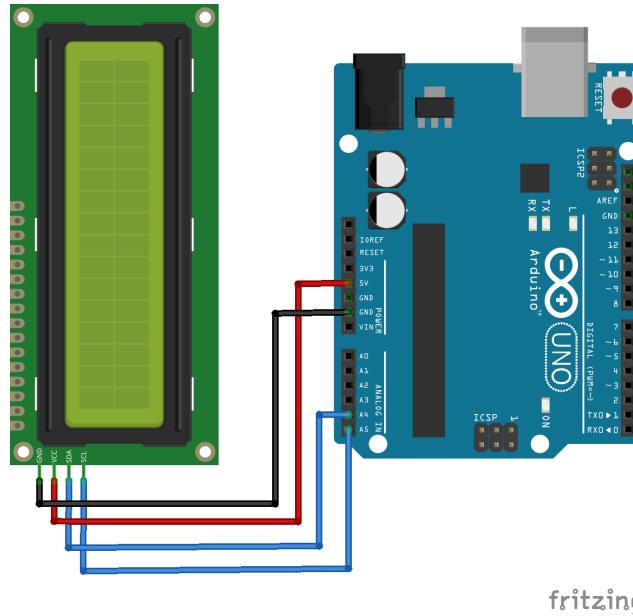
- Arduino board (e.g., Arduino Uno)
- LCD display (I2C module connected)
- Jumper wires
- Breadboard

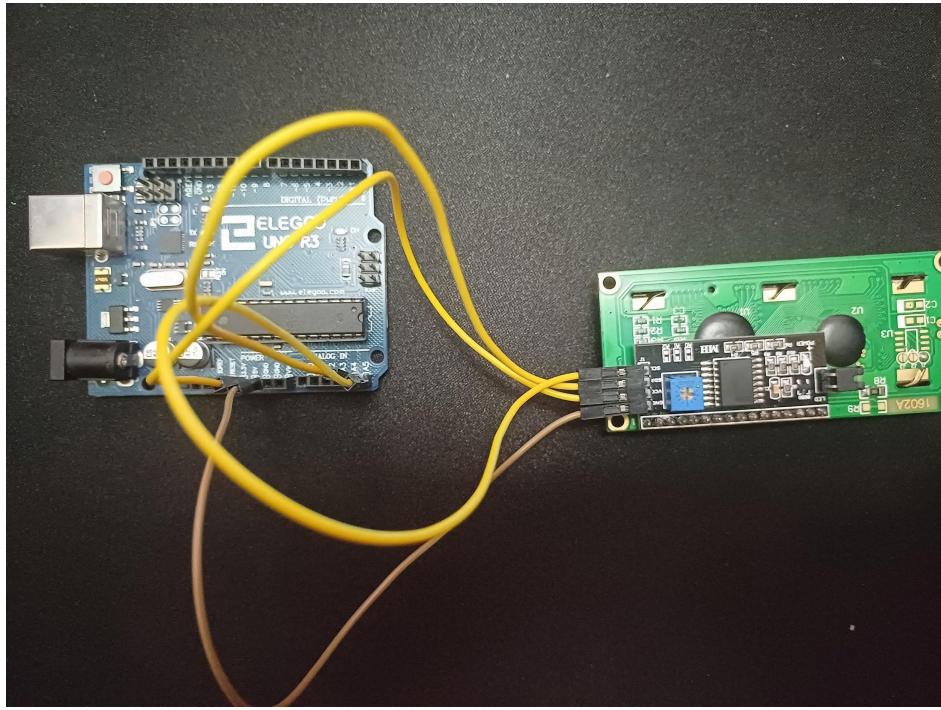
Procedure:

1. Setup the Circuit:

- Connect 4 male to female wires with the 4 pins of the I2C module.(I2C module will have 4 pins, GND, VCC, SDA, SCL)
- Connect the GND pin with any ground pin on the arduino board
- Connect the VCC with the +5v pin of the arduino
- As we will be sending analog data through SDA and SCL channels, connect to any ANALOG IN pin of the arduino board. For the tutorial, we have connected them to SDA → A4 and SCL → A5.

Circuit:





2. Setup the IDE:

- Go to this link: [GitHub - fdebrabander/Arduino-LiquidCrystal-I2C-library](https://github.com/fdebrabander/Arduino-LiquidCrystal-I2C-library): Library for the LiquidCrystal LCD display c+
- connected to an Arduino board.
- From here download the zip file
- Open a new sketch
- Select the appropriate port number after connecting the arduino
- Go to Sketch → Include Library → Add .ZIP Library
- From downloads upload the library.
- Go to File → Examples → Arduino liquid Crystal I2C Master → Hello world
- Compile the code and upload it. Your LCD should display “Hello World”

Code:

```
#include <Wire.h>
```

```
#include <LiquidCrystal_I2C.h>
```

```
// Set the LCD address to 0x27 for a 16 chars and 2 line display
LiquidCrystal_I2C lcd(0x27, 16, 2);
```

```

void setup()
{
    // initialize the LCD
    lcd.begin();

    // Turn on the blacklight and print a message.
    lcd.backlight();
    lcd.print("Hello, world!");
}

void loop()
{
    // Do nothing here...
}

```

3. Putting the experiment together:

- Connect the Arduino to your computer using a power supplying USB cable.
- Upload the code to the Arduino board.
- Power up the Arduino.

Analysis: Observe the LCD display as the Arduino sends in the data which the LCD displays following the I2C protocol

Communication protocol used: I2C

Task 3

- **Experiment Title: Combining task 1 and task 2(Showing the DHT22 values using the LCD display)**

Objective: To learn how to interface an LCD display with an Arduino board and how to interface a DHT22 sensor with an arduino board at the same time so that the analog values of DHT22 sensor can go through the arduino to the LCD display.

Materials Needed:

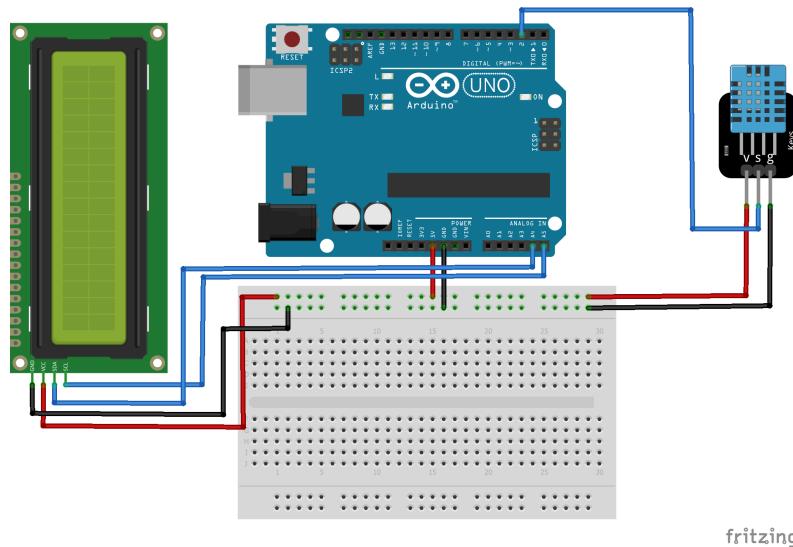
- Arduino board
- LCD display with I2C module
- DHT22
- Jumper wires
- Breadboard

Procedure:

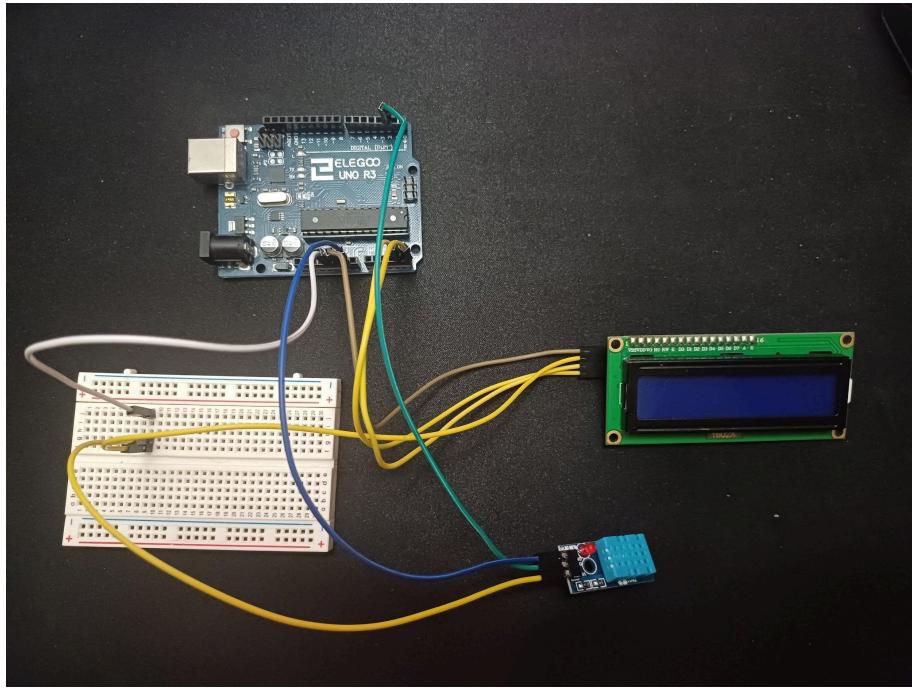
1. Setup the Circuit:

- As both of our devices need the +5V pin, take a male to male wire and connect one end to the +5v pin of the arduino and another end to the breadboard.
- Take the DHT22 sensor and connect it as described in task 1.(Vcc to the new +5v on the breadboard, data to pin 2, GND to any ground pin).
- Take the LCD with the I2C module and connect it as described in task 2.(Vcc to the new +5v on the breadboard, SDA → A4 and SCL → A5 and GND to any ground pin on the arduino.

Circuit:



fritzing



2. Write Code to operate the Arduino and LCD Display and the DHT22 sensor:

- Open the Arduino IDE on your computer.
- Create a new sketch.
- Copy the code below.
- Select the appropriate port and board.
- Compile and upload the code

Code:

```
#include <SimpleDHT.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

// for DHT22,
// VCC: 5V or 3V
// GND: GND
// DATA: 2
int pinDHT22 = 2;
SimpleDHT22 dht22(pinDHT22);
LiquidCrystal_I2C lcd(0x27, 16, 2);
```

```

void setup() {
  Serial.begin(9600);
}

void loop() {
  // start working...
  Serial.println("=====");
  Serial.println("Sample DHT22...");

  // read without samples.
  byte temperature = 0;
  byte humidity = 0;
  int err = SimpleDHTErrSuccess;
  if ((err = dht22.read(&temperature, &humidity, NULL)) != SimpleDHTErrSuccess) {
    Serial.print("Read DHT22 failed, err=");
    Serial.println(err);
    delay(1000);
    return;
  }

  lcd.begin();

  // Turn on the blacklight and print a message.
  lcd.backlight();
  lcd.print("temperature ");
  lcd.print((int)temperature);
  lcd.setCursor(0, 1);
  lcd.print("humidity ");
  lcd.print((int)humidity);
  delay(1000);
}

```

3. Putting the experiment together:

- Connect the Arduino to your computer using a USB cable.

- Upload the code to the Arduino board.
- Power up the Arduino.

Analysis: Observe the LCD display for the results of the DHT22 sensor to appear. Note any issues such as the text not displaying correctly or the display being blank.

Communication protocol used: Serial interface for the sensor and Inter-Integrated Circuit (I2C) for the LCD display

Task 4

- **Experiment Title: Using a sonar sensor to determine the distance to an object.**

Objective: To learn how the arduino board interfaces with the sonar sensor.

Materials Needed:

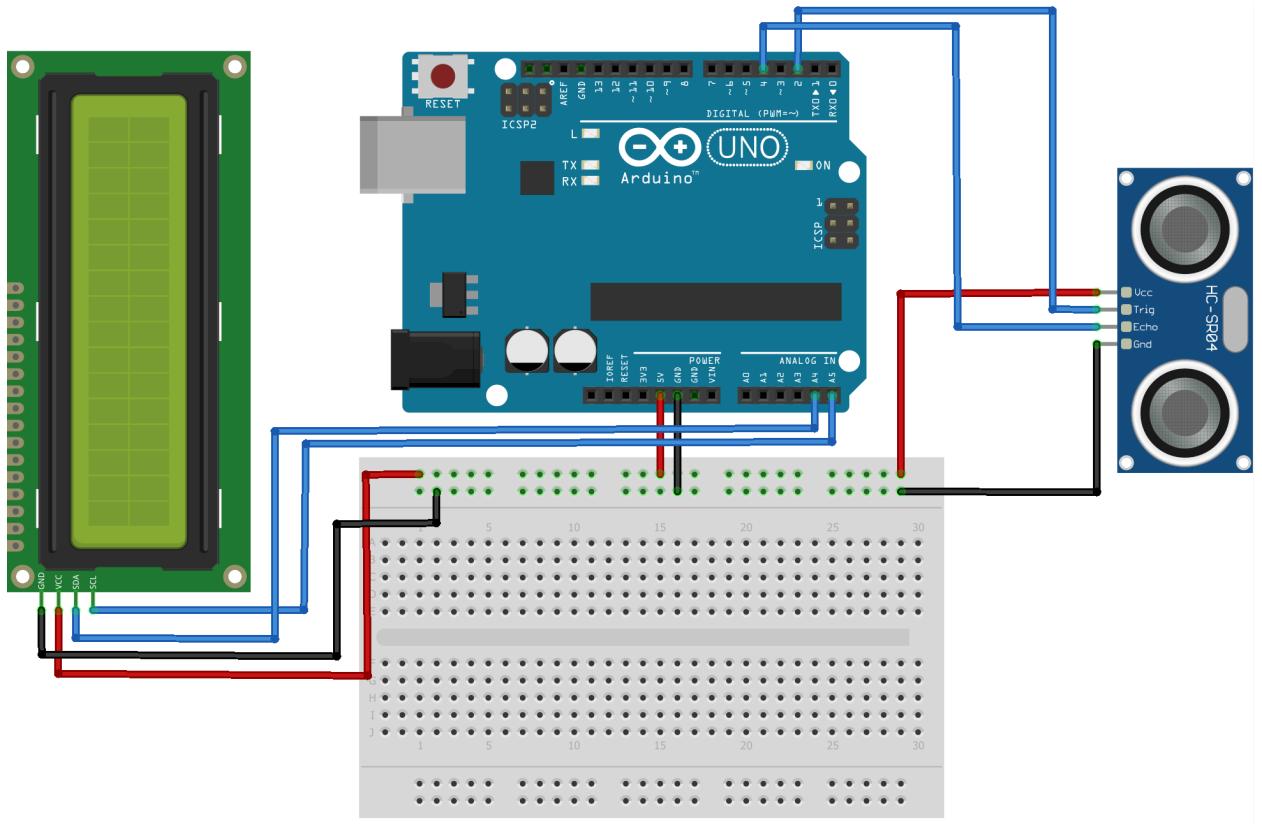
- Arduino board
- HC-SR04 Sonar sensor
- LCD display with I2C module
- Jumper wires
- Breadboard

Procedure:

1. Setup the Circuit:

- Connect 4 male to female wires to the 4 pins of the sensor. The sensor has 4 pins, **VCC, GND, TRIG and ECHO**.

- Connect the LCD display according to the previous experiments.
- TRIG of sonar sensor → pin 2
- Echo of the sonar sensor → pin 4



fritzing

2. Write Code to operate the Arduino and LCD Display and the Sonar sensor:

- Open the Arduino IDE on your computer.
- Create a new sketch.
- Copy the code below.
- Select the appropriate port and board.
- Compile and upload the code

Code:

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
```

```
LiquidCrystal_I2C lcd(0x27, 16, 2);
```

```
int trigPin = 2;  
int echoPin=4;  
long distance;  
long distanceInch;  
long duration;
```

```
void setup(){  
lcd.begin();  
lcd.backlight();  
  
pinMode(trigPin, OUTPUT);  
pinMode(echoPin, INPUT);  
lcd.setCursor(0, 0);  
  
}  
  
}
```

```
void loop() {  
ULTRASONIC();  
lcd.clear();  
lcd.setCursor(0,0);  
lcd.print("DISTANCE CM:");  
lcd.print(distance);  
lcd.setCursor(0,1);  
lcd.print("DISTANCE INCH:");  
lcd.print(distanceInch);  
delay(1000);  
//lcd.setCursor(0,1);  
// lcd.print("DISTANCE MM:");  
// lcd.print(distanceInch);  
}  
void ULTRASONIC() {  
digitalWrite(trigPin, LOW);  
delayMicroseconds (2);
```

```
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
duration= pulseIn(echoPin, HIGH);
distance = duration*0.034/2;
distanceInch = duration*0.0133/2;

}
```

3. Putting the experiment together:

- Connect the Arduino to your computer using a USB cable.
- Upload the code to the Arduino board.
- Power up the Arduino.

Analysis: Observe the LCD display for the results of the Sonar sensor to appear. Note any issues such as the text not displaying correctly or the display being blank.

Communication protocol used: I2C

Try it yourself!

- **For this evaluation, you must take an LED, and make it blink when either the humidity reaches 90 or the temperature reaches a 32.**