Group - 11

Soumava Paul 16EE10056
Yerramsetty Rohit 16EE10055

Experiment 6

Humidity, Moisture, Gas Sensing, Accelero-/Gyro, Magnetometer

Aim:

a. Interfacing a humidity sensor (DHT11) and display on PC via serial monitor
b. Interfacing a soil moisture sensor and display on PC via serial monitor
c. Interfacing a gas sensor (MQ2) and display on PC via serial monitor
d. Interfacing accelero-/gyro (GY61 ADXL335 or GY521 MPU6050) and magnetometer (QMC 5883L) and display on PC via serial monitor
e. Interfacing all the sensors and display on PC via Processing

Apparatus Required:

1. Arduino Uno Board
2. humidity sensor – DHT11 (4 pin)
3. Resistor (1k)
4. Soil Moisture Sensor (FC-38)
5. LM393 Comparator
6. Gas sensor (MQ-7)
7. Match Sticks (for Smoke)
8. Accelero/Gyrometer
9. Magnetometer
10. PC with Arduino IDE installed.

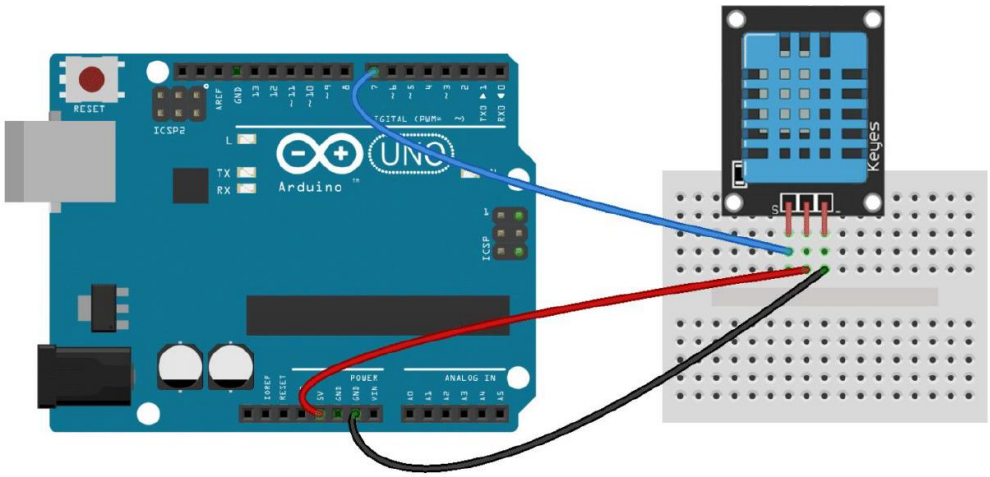
Theory:

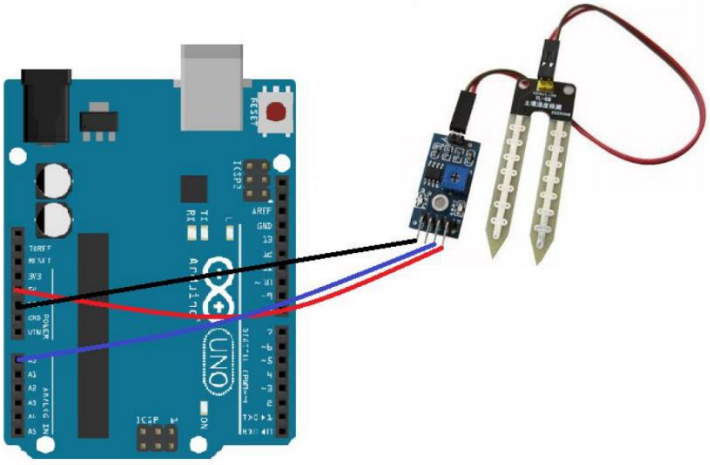
DHT11 detects water vapor by measuring the electrical resistance between two electrodes. The humidity sensing component is a moisture holding substrate with electrodes applied to the surface. When water vapor is absorbed by the substrate, ions are released by the substrate which increases the conductivity between the electrodes. The change in resistance between the two electrodes is proportional to the relative humidity. Higher relative humidity decreases the resistance between the electrodes, while lower relative humidity increases the resistance between the electrodes.

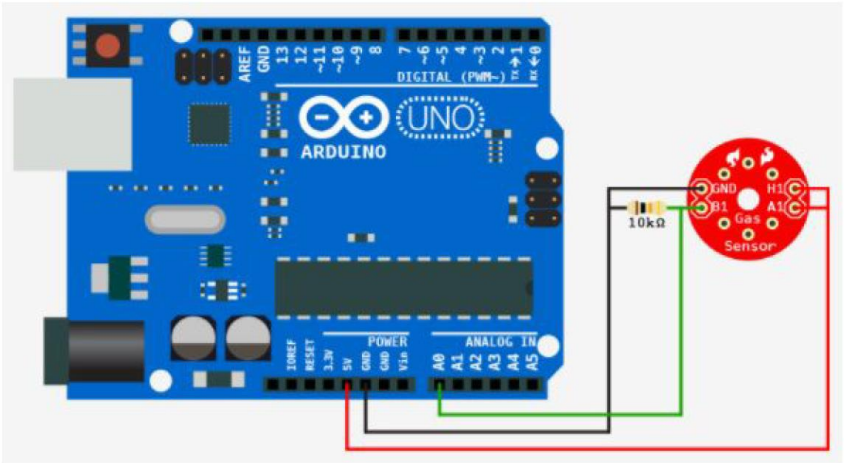
To connect the soil moisture sensor FC-38 in the digital mode, we will connect the digital output of the sensor to the digital pin of the Arduino. The sensor module contains a potentiometer with it, which is used to set the threshold value. This threshold value is then compared with the sensor output value using the LM393 comparator which is placed on the sensor module. The LM393 comparator will compare the sensor output value and the threshold value and then give the output

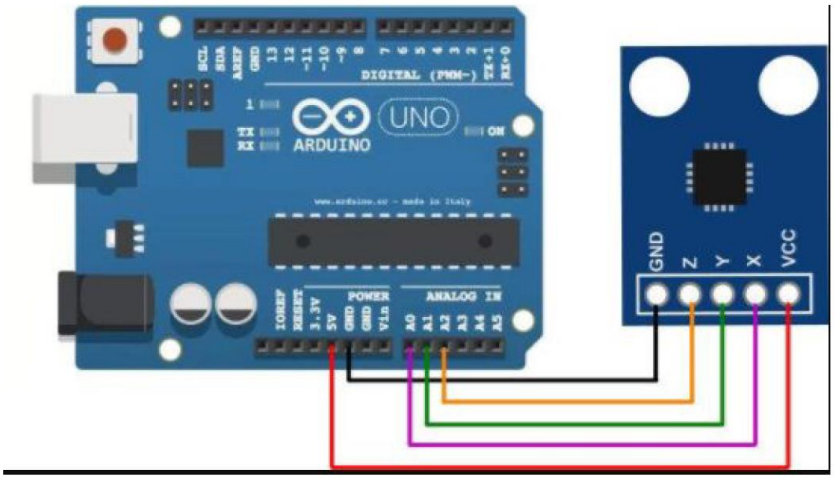
through the digital pin. When the sensor value will be greater than the threshold value, then the digital pin will give us 5V and the LED on the sensor will light up and when the sensor value will be less than this threshold value, then the digital pin will give 0V and the light will go down.

Circuit Diagrams:









Codes and Demos:

Humidity_Sensor

```
#include <dht.h>

dht DHT;

#define DHT11_PIN 7

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

void loop()
{
  int chk = DHT.read11(DHT11_PIN);
  Serial.print("Temperature = ");
  Serial.println(DHT.temperature);
  Serial.print("Humidity = ");
  Serial.println(DHT.humidity);
  delay(1000);
}
```

Soil_Moisture_Sensor

```
int sensor_pin = A0;

int output_value ;

void setup() {

   Serial.begin(9600);

   Serial.println("Reading From the Sensor ...");

   delay(2000);

   }

void loop() {

   output_value= analogRead(sensor_pin);

   output_value = map(output_value,1023,0,0,100);

   Serial.print("Moisture : ");

   Serial.print(output_value);

   Serial.println("%");

   delay(1000);

   }
```

Gas_Sensor_3

```
int sensorValue;
int output;
void setup() {
 Serial.begin(9600);
}

void loop() {
  sensorValue = analogRead(A0);
  output = map(sensorValue, 1023, 0, 100, 0);
  Serial.println(output);
  delay(1000);
 }
```

Accelerometer

```
const int xpin = A0;                  // x-axis of the accelerometer
const int ypin = A1;                  // y-axis
const int zpin = A2;                  // z-axis (only on 3-axis models)
void setup(){
  // initialize the serial communications:
  Serial.begin(9600);
}
void loop(){
  int x = analogRead(xpin);  //read from xpin
  int y = analogRead(ypin);  //read from ypin
  int z = analogRead(zpin);  //read from zpin
  float zero_G = 512.0; //ADC is 0~1023  the zero g output equal to Vs/2
                        //ADXL335 power supply by Vs 3.3V
  float scale = 102.3;  //ADXL335330 Sensitivity is 330mv/g
                        //330 * 1024/3.3/1000
  Serial.print(((float)x - 331.5)/65*9.8);  //print x value on serial monitor
  Serial.print("\t");
  Serial.print(((float)y - 329.5)/68.5*9.8);  //print y value on serial monitor
  Serial.print("\t");
  Serial.print(((float)z - 340)/68*9.8);  //print z value on serial monitor
  Serial.print("\n");
  delay(500);
}
```

Magnetometer

```
#include <Wire.h>
#include <HMC5883L.h>
HMC5883L compass;
void setup(){
  Serial.begin(9600);
  Wire.begin();
  compass = HMC5883L();
  compass.SetScale(1.3);
  compass.SetMeasurementMode(Measurement_Continuous);
}
void loop(){
  MagnetometerRaw raw = compass.ReadRawAxis();
  MagnetometerScaled scaled = compass.ReadScaledAxis();
  float xHeading = atan2(scaled.YAxis, scaled.XAxis);
  float yHeading = atan2(scaled.ZAxis, scaled.XAxis);
  float zHeading = atan2(scaled.ZAxis, scaled.YAxis);
  if(xHeading < 0) xHeading += 2*PI;
  if(xHeading > 2*PI) xHeading -= 2*PI;
  if(yHeading < 0) yHeading += 2*PI;
  if(yHeading > 2*PI) yHeading -= 2*PI;
  if(zHeading < 0) zHeading += 2*PI;
  if(zHeading > 2*PI) zHeading -= 2*PI;
  float xDegrees = xHeading * 180/M_PI;
  float yDegrees = yHeading * 180/M_PI;
  float zDegrees = zHeading * 180/M_PI;
  Serial.print(xDegrees);
  Serial.print(",");
  Serial.print(yDegrees);
  Serial.print(",");
  Serial.print(zDegrees);
  Serial.println(";");
  delay(1000);
}
```

Demo Videos: https://photos.google.com/share/AF1QipN6Ous6xtIR3VMxhGRkKb-IIBJtOXim9Og2LWjeGCqo0geNmWYaKHSz7kV1zLei0A?key=QzB0VFZrWW56T0hyN1ctVmU3cnhFd1JrVWZDdXhn

Since we need to show only the output of Serial Monitors of all the sensors,we can use this single code for all the sensors in Processing

Code:

```
import processing.serial.*;
Serial myPort; // Create object from Serial class
String val; // Data received from the serial port
void setup()
{
// I know that the first port in the serial list on my mac
// is Serial.list()[0].
// On Windows machines, this generally opens COM1.
// Open whatever port is the one you're using.
String portName = Serial.list()[0]; //change the 0 to a 1 or 2 etc. to match your port
myPort = new Serial(this, portName, 9600);
}
void draw()
{
if ( myPort.available() > 0)
{ // If data is available,
val = myPort.readStringUntil('\n'); // read it and store it in val
}
println(val); //print it out in the console
}
```

Discussions

Various sensors were studied and were implemented to read various analog signals. The soil moisture sensor was used to identify the moisture content in a soil sample, we do this by setting a range after observing completely dry soil and some wet soil. The humidity sensor is connected in a pretty straightforward way in which we simply read the humidity present in the atmosphere which changes when we blow air close to the sensor. The gas sensor is used to identify the CO content in atmosphere which we change by burning a piece of paper and observing the change in the CO content in atmosphere. The gyroscope is used to detect the change in orientation in all the 3 directions.