Name: **Kaleemulla adil Rahman**
Faculty ID: **183111006**

Rain detector

Using Arduino UNO and rain sensor

Description

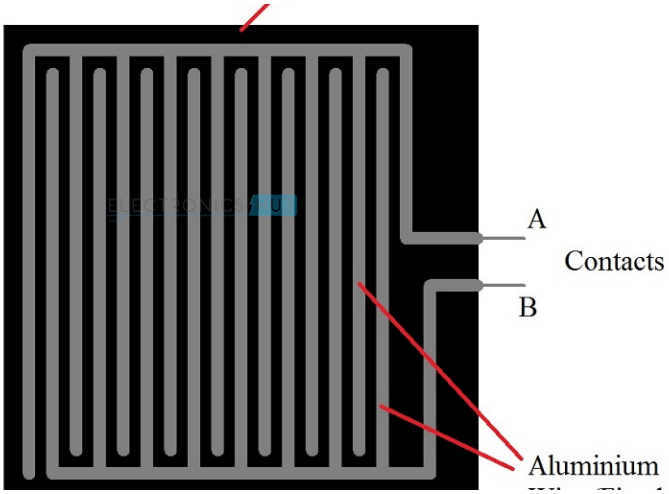
This project helps to detect rain fall by using a rain sensor, arduino uno and buzzer. This sort of mechanism can be used in a lot of field such as the automobile industry and agricultural industry. The main purpose is to alarm rain fall.

Material required

1. Arduino Uno
2. Rain sensor
3. Buzzer
4. Breakboard
5. Connecting wires

Rain sensor

Consist of 2 boards,

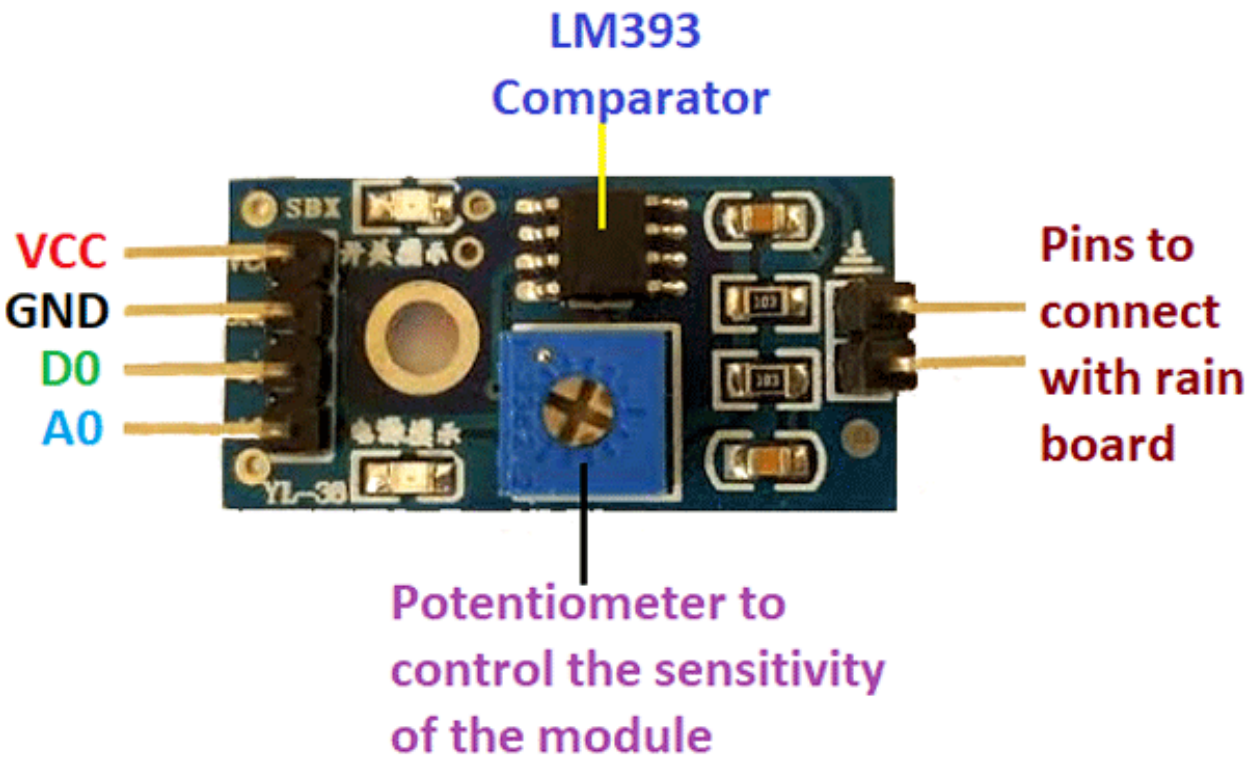


RAIN BOARD

Rain Board

The **Rain board** module consists of two copper tracks, designed in such a way that under the dry conditions they provide high resistance to the supply voltage, and this output voltage of this module will be 5V. This module's resistance gradually decreases with respect to an increase in the wetness on the board. As the **resistance decreases, its output voltage also decreases** with respect to the wetness on the module.

Control Board



CONTROL BOARD

Control Board module controls the sensitivity and **converts the analog output to digital output**. If the analog value is below the threshold value of the control board, the output is digital low, and If the analog value is higher than the threshold value, the output is digital high. For this comparison and conversion, an LM393 **OP-Amp Comparator** is used. An Op-Amp comparator is an interesting circuit that can be used to compare two different voltage values

The **Rain control module** which is shown below consists of 4 pins to connect the Arduino namely VCC, GND, D0, A0 and two more pins to connect the rain board module. In summary, the rain board module detects

the rainwater, and the control board module is used to control the sensitivity and compare and convert the analog values to digital values.

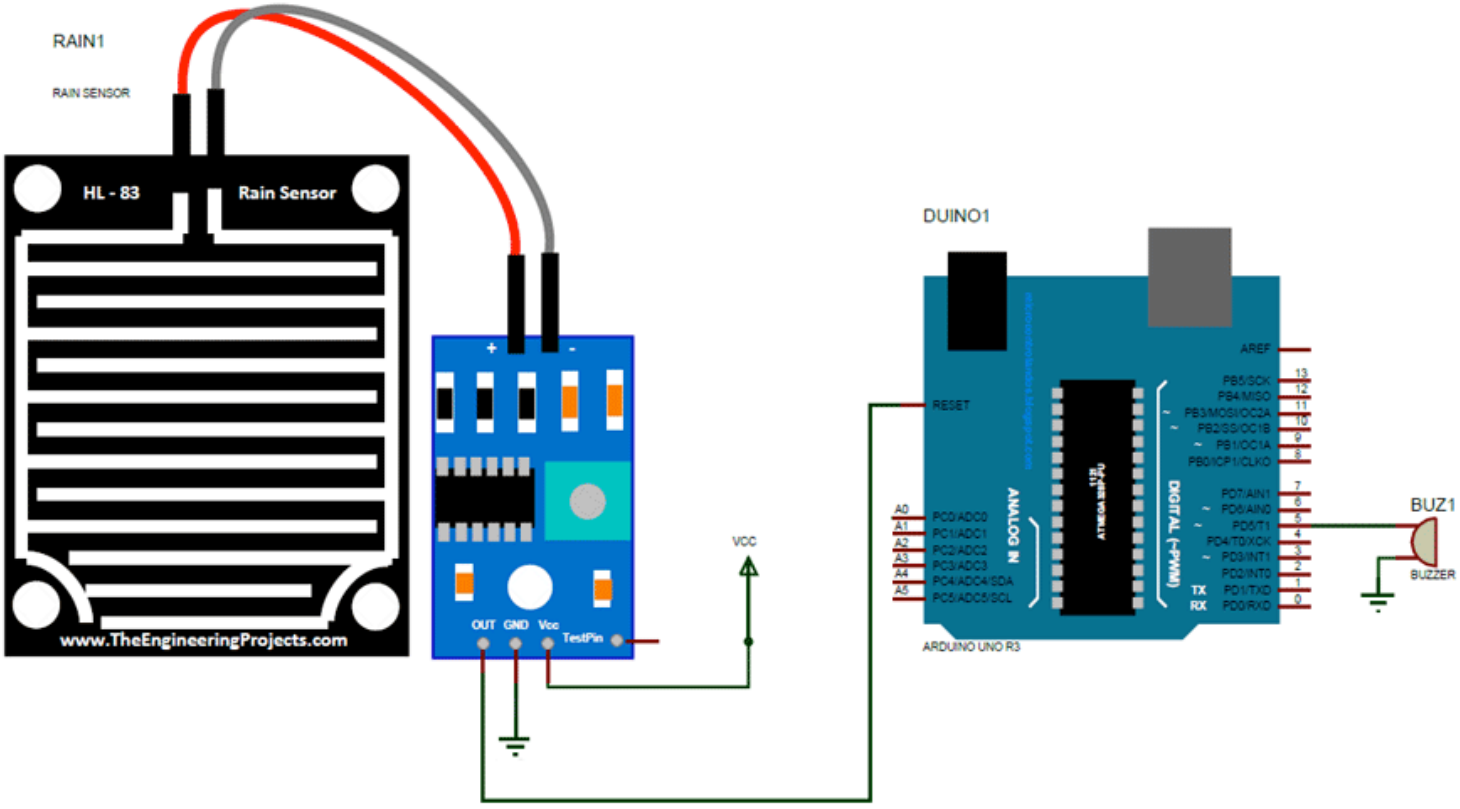
The mechanism

due to the dryness on the rain board module, it offers high resistance to the supply voltage. This voltage appears on the output pin of the rain board module as 5V. This 5V is read as 1023 if read by an analog pin of the Arduino.

When the rain board is fully wet, and the resistance offered by it is minimum, the output voltage will be as low as possible (approx. 0). This 0V is read as 0 value if read by an analog pin of the Arduino.

Formula to find ADC can be given by, **ADC = (analog voltage value X 1023)/5**. By using this formula you can convert any analog voltage to t Arduino analog read value.

The circuit



TYPE TO ENTER A CAPTION.

The rain gauge module, which is shown in the circuit diagram is connected to the control board. The control board's VCC pin is connected to the 5V supply. The ground pin is connected to ground. If needed, the D0 pin is connected to any digital pin of the Arduino, and that pin must be declared as an output pin in the program. The problem we face with the D0 pin is that we can't get the exact value of the output voltage. If the output crosses the threshold voltage, then the control module can sense the change in the output. We need to operate the buzzer, even if there is a considerable change in the output voltage in the rain board module. Due to these reasons, the A0 pin is connected to the analog pin of Arduino, which makes monitoring the change in output easy. The buzzer, which is used as a signal to the user, can be connected to any digital pin of the Arduino. If the buzzer needs more than 5V, then try to connect a relay circuit or a transistor and then connecting the load to it.

The Code and Explanation

// here I will be defining A0 as rain and pin 5 as a buzzer, Also setting 2 variables as "x" and "y" as integer and "y" value as 10. If you want the buzzer to buzz for even little amount of rain then keep the value of "y" less.

```
#define rainfall A0
#define buzzer 5
int x;
int y=10;
void setup() {
  Serial.begin(9600);
  pinMode(buzzer,OUTPUT);
  pinMode(rain,INPUT);
  }
```

// here I started the serial communication and setting the buzzer. Setting the rain pin as an output pin and input pin.

```
void loop() {

 x = analogRead(rain);
 Serial.println("LOL");
 Serial.println(x);
 x = map(x,0,1023,225,0);
```

```
16.  Serial.println(x);

// here If the read sensor value is greater than the set of the value then the program will enter a loop, prints the message on the serial monitor and switches on the buzzer.

17.  if(x >= y){
18.   Serial.println("rain detected");
19.   digitalWrite(buzzer,HIGH);
20.  }
21.  else{
22.   digitalWrite(buzzer,LOW);
23.  }
24.  delay(200);
25. }
```

This is the end of a detailed explanation of my assignment with Arduino Uno building an rainfall detector. Instead of a buzzer we can use a lot of other motors like a viper operator for the car windshield.

Thank you.