PROJECT 7

Rain Water Sensor Using Blynk

1. INTRODUCTION:-

Recently I bought a rain sensor for hooking it up to my arduino. I have a home garden and watering it in my area is troublesome. I live in an area where we get unexpected rains. So I wanted to make a system that would check for rain on all times and alert me whenever it rains. The system notifies me on my mobile and shows the intensity of rain on my screen.

Rain sensors are used in the detection of water beyond what a humidity sensor can detect.

COMPONENTS:-

1.WEMOS

2.RAIN WATER SENSOR

APPLICATION:-

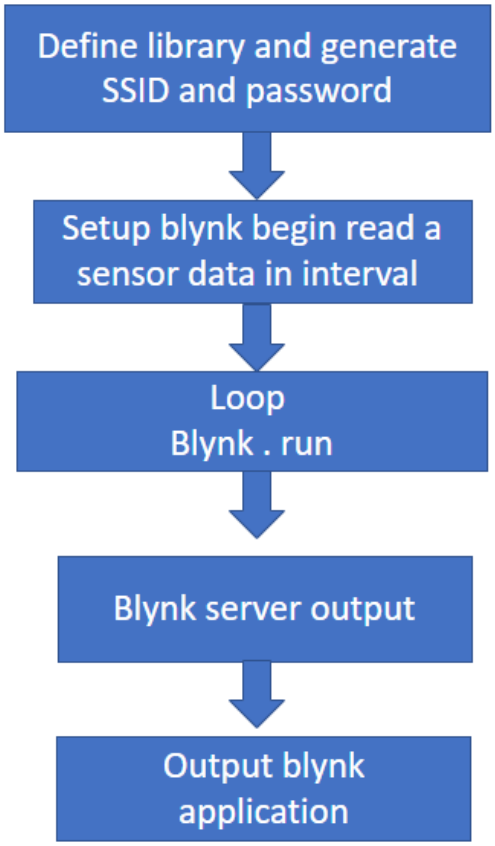
The applications of rain sensor include the following.

- This sensor is used as a water preservation device and this is connected to the irrigation system to shut down the system in the event of rainfall.
- This sensor is used to guard the internal parts of an automobile against the rainfall as well as to support the regular windscreen wiper's mode.
- This sensor is used in specialized satellite communications aerials for activating a rain blower over the opening of the aerial feed, to get rid of water droplets from the mylar wrap to keep pressurized as well as dry air within the waveguides.

OBJECTIVES:-

Rain Alarm Project is a simple but very useful project that detects Rain (Rain Water) and automatically triggers an alarm or buzzer. ... Rain water detector will detect the rain and make an alert; rain water detector is used in the irrigation field, home automation, communication, automobiles etc.

FLOW CHART :-



PROGRAMMING: -

```
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
```

```
char auth[] = "HILbiRWdNjbPkPr4NZ0MD8C_e9YWF_6E";//Enter your Auth token
char ssid[] = "RajibPaul";//Enter your WIFI name
char pass[] = "0563780324";//Enter your WIFI password

BlynkTimer timer;

void moisture() {
  int rainSensor = analogRead(A1);
  rainSensor = map(rainSensor, 0, 1023, 0, 100);
  Blynk.virtualWrite(V0, rainSensor);
  Serial.println(rainSensor);
}
void setup() {
  Serial.begin(9600);
  Blynk.begin(auth, ssid, pass);
  timer.setInterval(100L, moisture)
}


void loop()
{
  Blynk.run();
  timer.run();
}
```

```
#define BLYNK_PRINT Serial         // Uncomment for debugging

#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

#include <ESP8266mDNS.h>  // For OTA with ESP8266
#include <WiFiUdp.h>  // For OTA
#include <ArduinoOTA.h>  // For OTA

// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
char auth[] = "07DQbnsxODMhV3t0HC4QFZ5ZeHKIoFF6";

// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "Tshimologong-General";
char pass[] = "letsinnovate";

#define PULSE_PIN D2  //gpio4
#define FLOW_CALIBRATION 8.2
#define VPIN_TOTAL_LITERS      V1
#define VPIN_FLOW_RATE         V2
#define VPIN_FLOW_MILLI_LITERS  V3
#define VPIN_RESET             V4
```

```
#define OTA_HOSTNAME "Test Flow Sensor Water Meter"



BlynkTimer timer;

volatile long pulseCount = 0;
float flowRate;
unsigned int flowMilliLitres;
unsigned long totalMilliLitres;
float totalLitres;
float totalLitresold;
unsigned long oldTime;



BLYNK_CONNECTED() { // runs once at device startup, once connected to server.

  Blynk.syncVirtual(VPIN_TOTAL_LITERS); //gets last know value of V1 virtual pin

}
```

```
// Restores last know value of V1 virtual pin which we got it from blynk server
BLYNK_WRITE(VPIN_TOTAL_LITERS)
{
  totalLitresold = param.asFloat();

}



BLYNK_WRITE(VPIN_RESET) {  // reset all data with button in PUSH mode on virtual pin V4
  int resetdata = param.asInt();
  if (resetdata == 0) {
    Serial.println("Clearing Data");
    Blynk.virtualWrite(VPIN_TOTAL_LITERS, 0);
    Blynk.virtualWrite(VPIN_FLOW_RATE, 0);
    flowRate = 0;
    flowMilliLitres = 0;
    totalMilliLitres = 0;
    totalLitres = 0;
    totalLitresold = 0;
  }
}
```

```
void pulseCounter()
{
  pulseCount++;
}

void flow()
{

  if ((millis() - oldTime) > 1000)   // Only process counters once per second
  {
    detachInterrupt(PULSE_PIN);
    flowRate = ((1000.0 / (millis() - oldTime)) * pulseCount) / FLOW_CALIBRATION;
    oldTime = millis();
    flowMilliLitres = (flowRate / 60) * 1000;
    totalMilliLitres += flowMilliLitres;
    totalLitres = totalLitresold + totalMilliLitres * 0.001;
    unsigned int frac;

    // Print the flow rate for this second in liters / minute
    Serial.print("flowrate: ");
    Serial.print(int(flowRate));  // Print the integer part of the variable

    Serial.print(".");          // Print the decimal point
```

```
   frac = (flowRate - int(flowRate)) * 10; // Determine the fractional part. The 10 multiplier gives us 1 decimal place.
   Serial.print(frac, DEC) ;      // Print the fractional part of the variable
   Serial.print("L/min");


   Serial.print("  Current Liquid Flowing: ");  // Print the number of liters flowed in this second
   Serial.print(flowMilliLitres);
   Serial.print("mL/Sec");


   Serial.print("  Output Liquid Quantity: ");  // Print the cumulative total of liters flowed since starting
   Serial.print(totalLitres);
   Serial.println("L");


   pulseCount = 0;  // Reset the pulse counter so we can start incrementing again


   attachInterrupt(PULSE_PIN, pulseCounter, FALLING);    // Enable the interrupt again now that we've finished sending output
  }


}



void sendtoBlynk()  // In this function we are sending values to blynk server
```

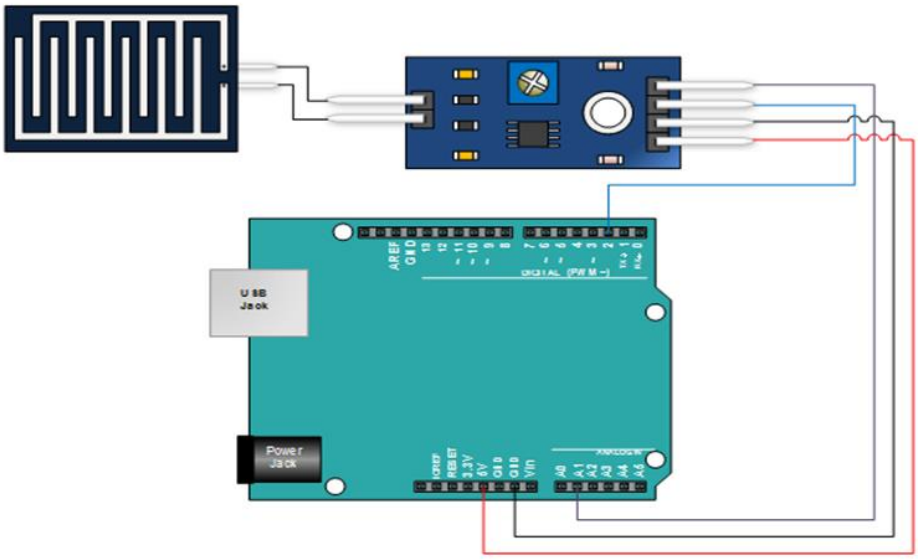
```
{
  Blynk.virtualWrite(VPIN_TOTAL_LITERS, totalLitres);        // Total water consumption in liters (L)
  Blynk.virtualWrite(VPIN_FLOW_RATE, flowRate);          // Displays the flow rate for this second in liters / minute (L/min)
  //  Blynk.virtualWrite(VPIN_FLOW_RATE, flowMilliLitres);  // Displays the number of liters flowed in second (mL/Sec)

}

void setup()
{
  Serial.begin(57600);
  Blynk.begin(auth, ssid, pass);
  ArduinoOTA.setHostname(OTA_HOSTNAME);  // For OTA - Use your own device identifying name
  ArduinoOTA.begin();  // For OTA

  pulseCount       = 0;
  flowRate         = 0.0;
  flowMilliLitres   = 0;
  totalMilliLitres  = 0;
  oldTime          = 0;
  totalLitresold = 0;

  pinMode(PULSE_PIN, INPUT);  // Initialization of the variable "PULSE_PIN" as INPUT (D2 pin)
```

```
  attachInterrupt(PULSE_PIN, pulseCounter, FALLING);


  timer.setInterval(10000L, sendtoBlynk); // send values blynk server
every 10 sec


}



void loop()
{


  Blynk.run();
  ArduinoOTA.handle();  // For OTA
  timer.run();
  flow();


}
```

HARDWARE CONNECTION:-

1. Connect rain water sensor to wemos
2. Connect pin A0 to A1
3. Connect pin GND to GND
4. Connect pin 5V to 5V

CURCUIT DIAGRAM:-


USB
Jack
Power
Jack
AREF
GND
DIGITAL (PWM~)
RESET
3.3V
5V
Vin
A0
A1
A2
A3
A4
A5