/*

Cyber Physical Systems Project

Author: Aanchal Chaturvedi

Water Leak Detection System

created by using - UCTronics Ultimate Starter

Kit for Arduino

Design Specifications for this sketch -

This sketch

is designed to determin water leaks in building and detect humidity and temperature

at the time of leakage.

The data recorded from the sensors is sent to Adafruit

Cloud

Parts required:

- 1 Mega2560 R3

- 1 ESP8266 Module

- 1 DHT-11 Sensor Module

- 1 Grove Water Level Sensor Module

- Male

to Male Jumper Wires

- Male to Female Jumper Wires

- 1 830 Tie Point Breadboard

Library Used:

Wifi Esp

https://www.arduino.cc/reference/en/libraries/wifiesp/

Adafruit MQTT

https://www.arduino.cc/reference/en/libraries/adafruit-mqtt-library/

DHT Sensor Library

https://www.arduino.cc/reference/en/libraries/dht-sensor-library/

References:

https://github.com/UCTRONICS/uctronics_arduino_kits/blob/master/Code/Lesson_25_water_level_detection_sensor_module/Lesson_25_water_level_detection_sensor_module.ino

Arduino Examples > Adafruit MQTT Library > mqtt_esp8266 https://github.com/esp8266/Arduino

*/

```
#include "WiFiEsp.h"
#include "Adafruit_MQTT.h"
#include
 "Adafruit_MQTT_Client.h"
#include <DHT.h>
#include <DHT_U.h>

/**********************
 DHT Pin Settings ******************************/
//Input pin for DHT Sensor
 is Pin 7
```

```
#define DHTPIN A1

//We are using is DHT 11 type of DHT sensor
  Module
#define DHTTYPE DHT11
//create a DHT object/instance
//DHTPIN -
  source pin for DHT sensor on the microcontroller
//DHTTYPE - type of DHT Sensor
  Module

DHT dht(DHTPIN, DHTTYPE);

/******************Water Level Sensor
  Settings *************************/

// Configuration for the Water Level
  Sensor analog input and output
#define WL_POWER_PIN  3
#define WL_SIGNAL_PIN
  A0

/******************Active Buzzer pin ***********************************/
int
  buzzerPin = 4; //definition digital 8 pins as pin to control the buzzer

/************************
  WiFi Settings ********************************/

char ssid[] = "**********";
          // your network SSID (name)
```

```
char password[] = "********";       //
  your network password
int status = WL_IDLE_STATUS;     // the Wifi radio's status

/************************
  Adafruit.io Setup *********************************/

#define AIO_SERVER      "io.adafruit.com"
#define
  AIO_SERVERPORT  1883                   // use 8883 for SSL
#define AIO_USERNAME
    "************"
#define AIO_KEY        "***************"

WiFiEspClient
  client;

// Setup the MQTT client class by passing in the WiFi client and
  MQTT server and login details.
Adafruit_MQTT_Client mqtt(&client, AIO_SERVER,
  AIO_SERVERPORT, AIO_USERNAME, AIO_KEY);

//publish to temperature feed located
  at aanchal0431/feeds/WaterLevel
Adafruit_MQTT_Publish inputWaterLevel = Adafruit_MQTT_Publish(&mqtt,
  AIO_USERNAME "/feeds/WaterLevel");
//publish to humidity feed located at aanchal0431/feeds/humidity
Adafruit_MQTT_Publish
  inputHumidity = Adafruit_MQTT_Publish(&mqtt, AIO_USERNAME "/feeds/humidity");
//publish
  to temperature feed located at aanchal0431/feeds/temperature
```

```
Adafruit_MQTT_Publish
 inputTemperature = Adafruit_MQTT_Publish(&mqtt, AIO_USERNAME "/feeds/temperature");
//publish
 to temperature feed located at aanchal0431/feeds/temperature
Adafruit_MQTT_Publish
 inputonoff = Adafruit_MQTT_Publish(&mqtt, AIO_USERNAME "/feeds/onoff");



void
 setup() {

 //baud rate for mega board
 Serial.begin(115200);

 //The serial connection at RX1, TX1 is open and 9600 is the baud rate for ESP8266

 Serial1.begin(9600);

 Serial.println("Cyber Physical Systems Project
 - Water Leakage Detection Sensor");
 delay(2000);

 // Connect to WiFi

 Serial.println();
 WiFi.init(&Serial1);
 // check for the presence of
 the shield
 if (WiFi.status() == WL_NO_SHIELD) {
  Serial.println("WiFi
 shield not present");
```

```
  // don't continue
}

if (status != WL_CONNECTED)
{
  // attempt to connect to WiFi network
  Serial.print("Attempting
to connect to WPA SSID: ");
  Serial.println(ssid);
  // Connect to WPA/WPA2
network//
  status = WiFi.begin(ssid, password);
  while (WiFi.status()
!= WL_CONNECTED) {
    delay(500);
  }

  Serial.println("You're
connected to the network");
}

//setup for Water Level Sensor

pinMode(WL_POWER_PIN, OUTPUT);   // configure  pin as an OUTPUT

//initialization
for DHT sensor
dht.begin();

//buzzer pin mode
pinMode(buzzerPin,
```

```
 OUTPUT);
}

void loop() {

 MQTT_connect();

 delay(10000);//1
 seconds delay between each run

 //Humidity and Temperature Sensor code

 float humidity = dht.readHumidity();
 float temperature = dht.readTemperature(true);



 // Water Level Sensor Code
 long waterLevel = getWaterLevel();
 Serial.println("Water
 Level is: " + waterLevel);

 //calibrations for humidity and temperature
 reading is based on the average of
 //the readings before
 humidity = humidity
 + 14;
 temperature = temperature - 2;

 if (! inputWaterLevel.publish(waterLevel))
 {
  Serial.println(F("Failed"));
```

```
} else {
  Serial.println(F("OK!"));

}
if (! inputHumidity.publish(humidity)) {
  Serial.println(F("Failed"));

} else {
  Serial.println(F("OK!"));
}

if (! inputTemperature.publish(temperature))
{
  Serial.println(F("Failed"));
} else {
  Serial.println(F("OK!"));

}

// validate water level values
if (isnan(waterLevel)) {
  Serial.println("Unable
to read from the Water Level sensor!");
  //end loop without executing the
remaining code
  return;
}

//validate temperature and humidity
values
if (isnan(humidity) || isnan(temperature)) {
```

```
  Serial.println("Unable
to read from the DHT sensor!");
  //end loop without executing the remaining
code
  return;
}

if (waterLevel > 0 && waterLevel <= 100) {

  long onStatus = 10;
  soundAlarm();
  if (! inputonoff.publish(onStatus))
{
    Serial.println(F("Failed"));
  } else {
    Serial.println(F("OK!"));

  }
} else {
 long onStatus = 0;
 if (! inputonoff.publish(onStatus))
{
    Serial.println(F("Failed"));
  } else {
    Serial.println(F("OK!"));

  }
}

//Print Humidity and Temperature sensor readings and Water
Level Sensor in serial monitor
```

```
  Serial.print("Water Level =  ");
  Serial.print(waterLevel);

  Serial.print("    ");
  Serial.print("Humidity = ");
  Serial.print(humidity);

  Serial.print("%   ");
  Serial.print("Temperature = ");
  Serial.print(temperature);

  Serial.println("F");

}

void MQTT_connect() {
  int8_t ret;

  // Stop if already connected.
  if (mqtt.connected()) {
    return;

  }
  Serial.print("Connecting to MQTT... ");
  uint8_t retries = 3;

  while ((ret = mqtt.connect()) != 0) { // connect will return 0 for connected

    Serial.println(mqtt.connectErrorString(ret));
    Serial.println("Retrying
  MQTT connection in 5 seconds...");
```

```
    mqtt.disconnect();
    delay(5000);
  // wait 5 seconds
    retries--;
    if (retries == 0) {
      // basically
  die and wait for reset
      while (1);
    }
  }
  Serial.println("MQTT
  Connected!");
}

void soundAlarm(){
    digitalWrite(buzzerPin, HIGH);
  //Set PIN 8 feet as HIG67890]\\H = 5 v
    delay(2000);                //Set
  the delay time2000ms
    digitalWrite(buzzerPin, LOW); //Set PIN 8 feet for LOW
  = 0 v
    delay(2000);                //Set the delay time2000ms
}




long
  getWaterLevel(){
   digitalWrite(WL_POWER_PIN, HIGH);  // turn the sensor ON

   delay(10);                  // wait 10 milliseconds
```

```
  long waterLevel
 = analogRead(WL_SIGNAL_PIN); // read the analog value from sensor
  digitalWrite(WL_POWER_PIN,
 LOW);   // turn the sensor OFF
  return waterLevel;
}
```