**Coding Arduino UNO for Simulation**

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd(0x27, 16, 2);

int LDR;

int LDRPin=A3;

int LDR2;

int LDRPin2=A2;

int LDR3;

int LDRPin3=A1;

#include <Servo.h>

Servo myservo1;

Servo myservo2;

void setup()

{

lcd.init();

lcd.backlight();

myservo1.attach(A0);

myservo2.attach(10);

Serial.begin(9600);

}

void loop(){

lcd.setCursor(0, 0);

lcd.print("sensor=9V");

LDR=analogRead(LDRPin);

Serial.println(LDR);

delay(100);

LDR2=analogRead(LDRPin2);

Serial.println(LDR2);

delay(100);

LDR3=analogRead(LDRPin3);

Serial.println(LDR3);

delay(100);

if(LDR>500){

myservo1.write(180);

myservo2.write(0);

}

if(LDR2>500){

myservo1.write(150);

myservo2.write(60);

}

if(LDR3>500){

myservo1.write(0);

myservo2.write(0);

}

}

**Coding Arduino UNO for Hardware**

#include <SoftwareSerial.h>

SoftwareSerial esp(6, 7); //Pin 6 and 7 act as RX and TX. Connect them to TX and RX of ESP8266

#define DEBUG true

int number;

String mySSID = "abdo990"; // Wi-Fi SSID

String myPWD = "12345678"; // Wi-Fi Password

String myAPI = " XWBKJTA6VSKP5X7D"; // WRITE API Key

String myHOST = "api.thingspeak.com";

String myPORT = "80";

String myFIELD = "field1";

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd(0x27, 16, 2);

int LDR;

int LDRPin=A3;

int LDR2;

int LDRPin2=A2;

int LDR3;

int LDRPin3=A1;

#include <Servo.h>

Servo myservo1;

Servo myservo2;

#define ANALOG\_IN\_PIN A0

float adc\_voltage = 0.0;

float in\_voltage = 0.0;

float R1 = 30000.0;

float R2 = 7500.0;

float ref\_voltage = 5.0;

// Integer for ADC value

int adc\_value = 0;

int LED1=2;

int LED2=3;

void setup()

{

Serial.begin(115200);

esp.begin(115200);

lcd.init();

lcd.backlight();

myservo1.attach(11);

myservo2.attach(10);

pinMode(LED1,OUTPUT);

pinMode(LED2,OUTPUT);

Send\_AT\_Cmd("AT+RST", 1000, DEBUG);

Send\_AT\_Cmd("AT+CWMODE=1", 1000, DEBUG);

Send\_AT\_Cmd("AT+CWJAP=\""+ mySSID +"\",\""+ myPWD +"\"", 1000, DEBUG);

delay(1000);

}

void loop()

{

adc\_value = analogRead(ANALOG\_IN\_PIN);

adc\_voltage = (adc\_value \* ref\_voltage) / 1024.0;

in\_voltage = adc\_voltage\*(R1+R2)/R2;

Serial.print("Input Voltage = ");

Serial.println(in\_voltage, 2);

lcd.setCursor(0, 0);

lcd.print("voltage=");

lcd.setCursor(9, 0);

lcd.print(in\_voltage, 2);

delay(500);

if(in\_voltage>7){

lcd.setCursor(5, 1);

lcd.print("LED1");

digitalWrite(LED1,HIGH);

digitalWrite(LED2,LOW);

}

if(in\_voltage<7){

lcd.setCursor(5, 1);

lcd.print("LED2");

digitalWrite(LED1,LOW);

digitalWrite(LED2,HIGH);

}

LDR=analogRead(LDRPin);

Serial.println(LDR);

delay(100);

LDR2=analogRead(LDRPin2);

Serial.println(LDR2);

delay(100);

LDR3=analogRead(LDRPin3);

Serial.println(LDR3);

delay(100);

if(LDR>500){

myservo1.write(180);

myservo2.write(0);

}

if(LDR2>500){

myservo1.write(150);

myservo2.write(60);

}

if(LDR3>500){

myservo1.write(0);

myservo2.write(0);

}

number = random(100); // Send a random number between 1 and 100

String sendData = "GET /update?api\_key="+ myAPI +"&"+ myFIELD +"="+String(in\_voltage, 2);

Send\_AT\_Cmd("AT+CIPMUX=1", 1000, DEBUG); //Allow multiple connections

Send\_AT\_Cmd("AT+CIPSTART=0,\"TCP\",\""+ myHOST +"\","+ myPORT, 1000, DEBUG);

Send\_AT\_Cmd("AT+CIPSEND=0," +String(sendData.length()+4),1000,DEBUG);

esp.find(">");

esp.println(sendData);

Serial.print("Value to be sent: ");

Serial.println(number);

Send\_AT\_Cmd("AT+CIPCLOSE=0",1000,DEBUG);

Serial.println("Done!");

Serial.println("");

delay(10000);

}

String Send\_AT\_Cmd(String command, const int timeout, boolean debug)

{

Serial.print(command);

Serial.println(" ");

String response = "";

esp.println(command);

long int time = millis();

while ( (time + timeout) > millis())

{

while (esp.available())

{

char c = esp.read();

response += c;

}

}

if (debug)

{

//Serial.print(response);

}

return response;

}