**16BCB0056**

**RAVI SHANKAR**

**Description and other similar type of components**

Our project Smart gardening system, in our project we are implementing the system using a Wi-Fi module, Arduino UNO, Soil moisture sensor, Light sensor, water pump and an LED. What happens after we start our system is that the moisture levels and light levels are continuously updated to Thingspeak platform for the monitoring purpose and if the water is less then system turns on the soil moisture sensor and it runs for about 5 seconds. If there is less light then the artificial light is turned on until the light levels come to normal levels.

The similar components that we can use instead of what we used are…... we can replace the Arduino Uno with a base Atmega328p microcontroller or we can use other versions of Arduino with basic functions of different microcontroller or we can use a computer like Raspberry pi. We can replace the water pump with a solenoid valve and we can also use humidity, temperature, Rain sensors incase if we want to know how the situation is around our gardens

**Working Principle**

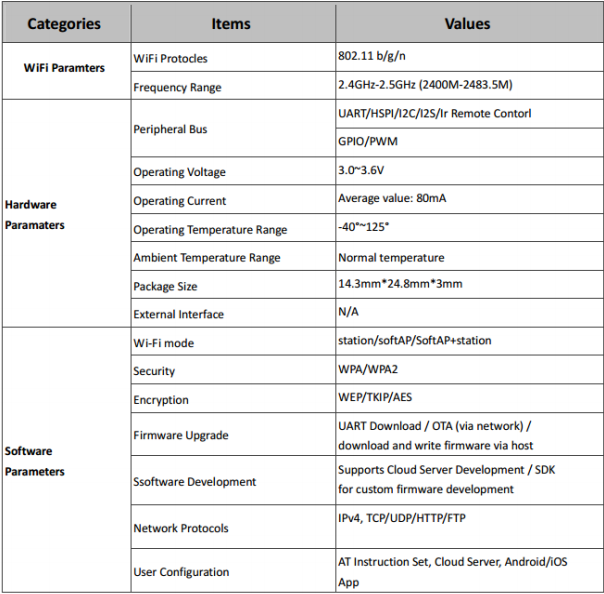
After we start the system, it will be connected to our Hotspot via given username and password and if it is connected then the values of our moisture sensor is collected from A0 and light sensor values are collected from A1. Then it is connected to our think speak account via the API key provided and which is connected to “api.thingspeak.com” and using the CIPSEND we send our data and then the connection is closed by CIPCLOSE command to Wi-Fi module. Now the sensor values will be compared to the min/max data and then accordingly the functions are executed. If the soil moisture value is more than the required moisture value then the water pump pumps the water for about 5 seconds then the code will be continued, it will run like that until the values come to desired levels. If the light is less, then the artificial light is turned on So that the tree will get sufficient light

We can monitor the values in Thingspeak website or a mobile application

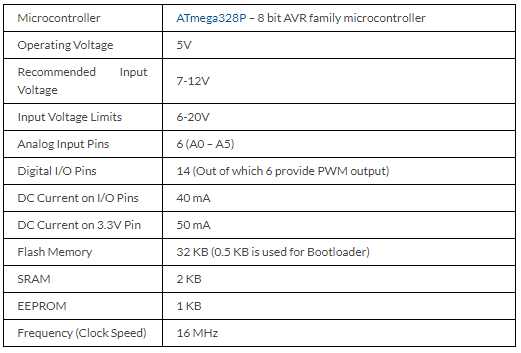


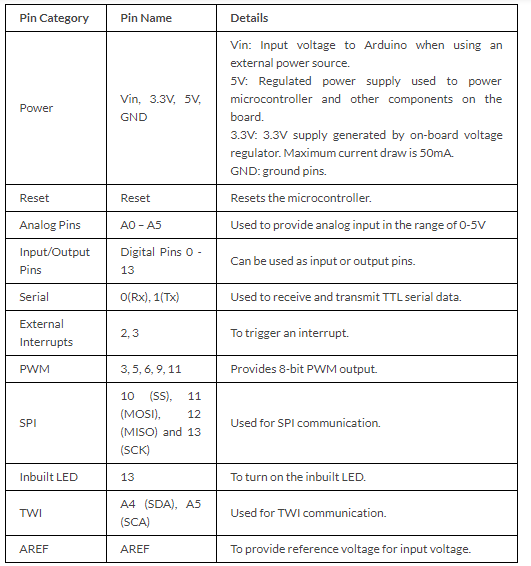
**Data Sheet**

**ESP8266 :**

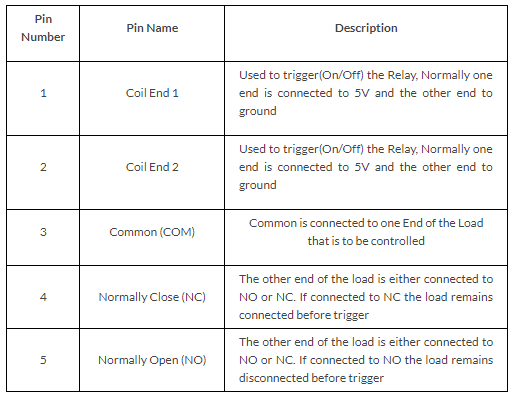


**Arduino:**





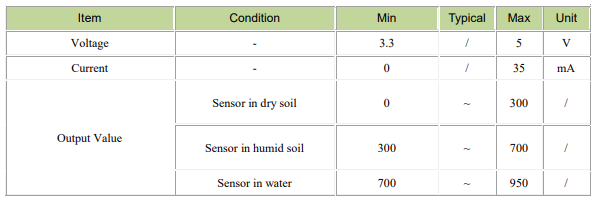
**Relay:**



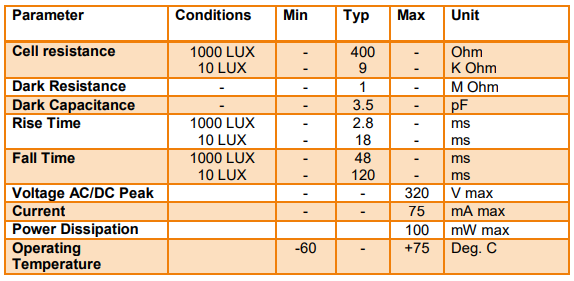
**5v Water pump :**

|  |  |  |
| --- | --- | --- |
| S.no | Type | Description |
| 1 | DC Voltage | 2.5-6V |
| 2 | Maximum lift | 40-110cm / 15.75″-43.4″ |
| 3 | Flow rate | 80-120L/H |
| 4 | Outside diameter of water outlet | 7.5mm / 0.3″ |
| 5 | Inside diameter of water outlet | 5mm / 0.2″ |
| 6 | Diameter | 24mm / 0.95″ |
| 7 | Length | 45mm / 1.8″ |
| 8 | Height | 30mm / 1.2″ |
| 9 | Material | engineering plastic |

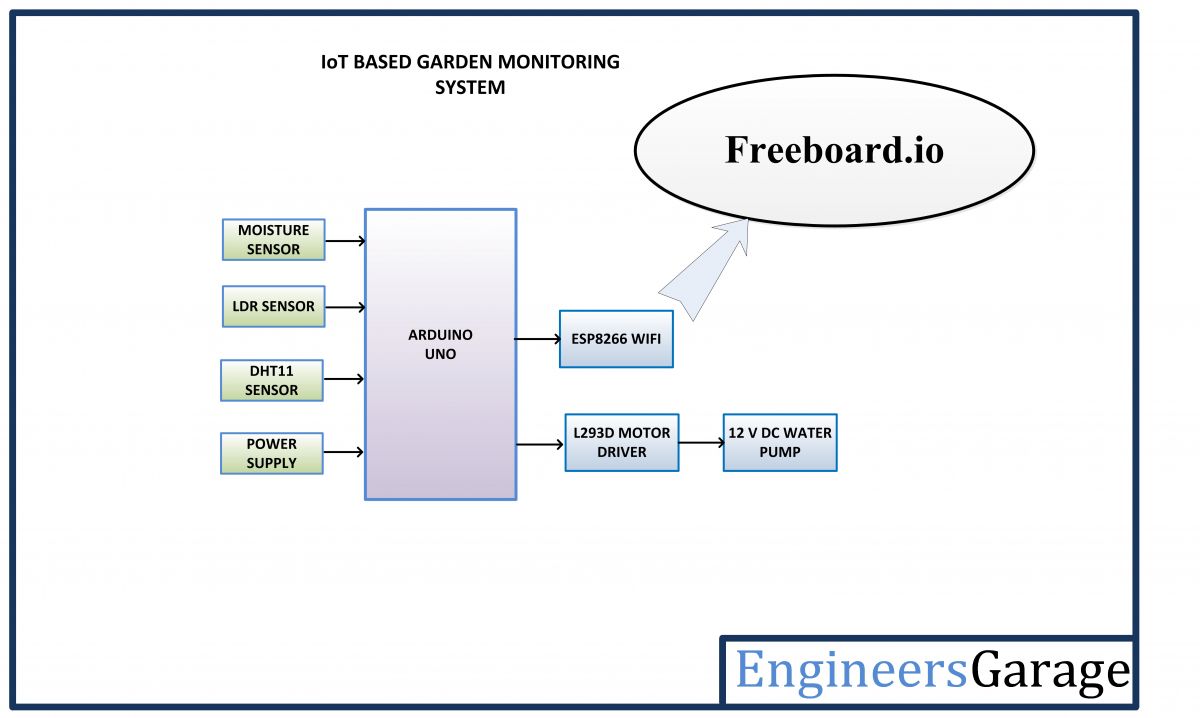
**Soil moisture sensor:**



**LDR sensor:**



**Block Diagram**



122

A0

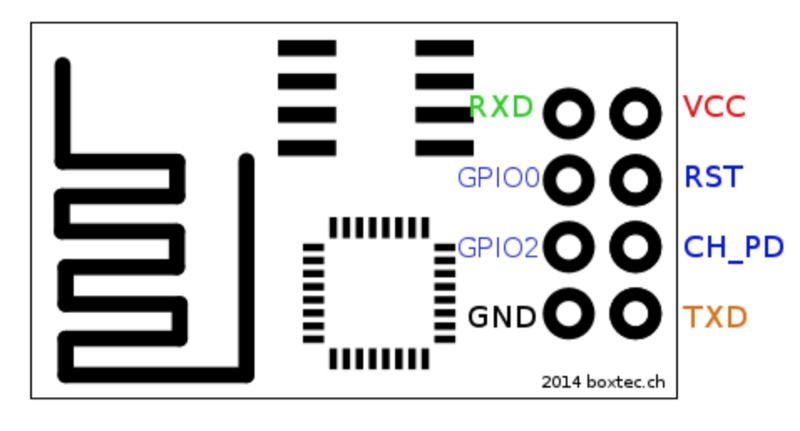
A1

10

11

5v water pump

**Interface Diagram**

****

GND

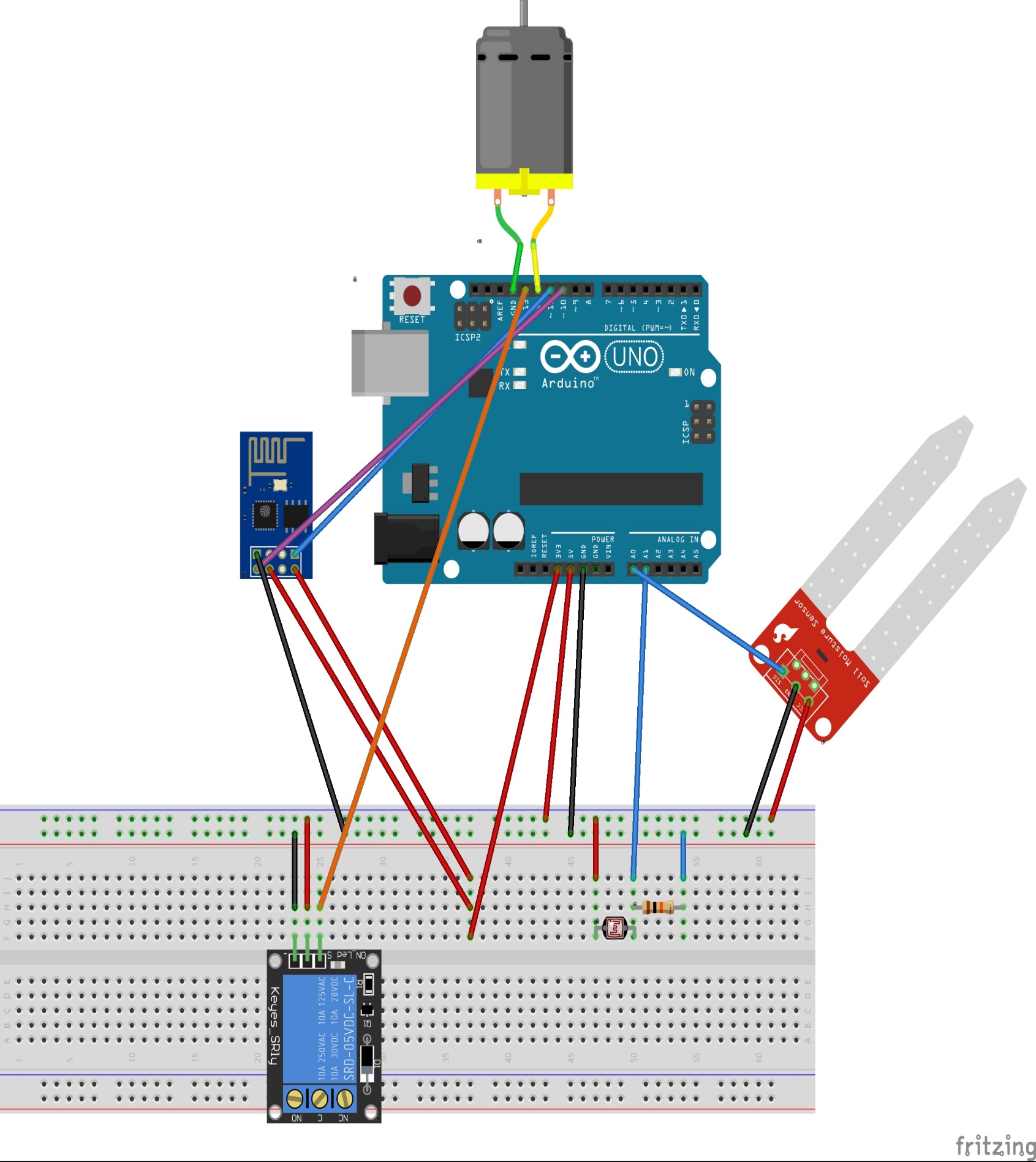
PIN 10

PIN 11

3.3 V

3.3 V

* **Not used**

****

GND

PIN 13

PIN 12

A1

A0

3.3V

GND

5V

**Code for Read and Write**

#include <SoftwareSerial.h>

#define pump 12

#define led 13

#define MOS A0

#define LDR A1

int mos\_val;

int lig\_val;

String AP = "chris";

String PASS = "";

String API = "DPFZR0CGYU5SI8T5";

String HOST = "api.thingspeak.com";

String PORT = "80";

int countTrueCommand = 1;

int countTimeCommand;

boolean found = false;

SoftwareSerial esp8266(10, 11);

void setup() {

Serial.begin(9600);

esp8266.begin(115200);

pinMode(led, OUTPUT);

pinMode(pump, OUTPUT);

sendCommand("AT", 5, "OK");

sendCommand("AT+CWMODE=3", 5, "OK");

sendCommand("AT+CWJAP=\"" + AP + "\",\"" + PASS + "\"", 20, "OK");}

void loop() {

String getData = "GET /update?api\_key=" + API + "&field1=" + String(read\_mos()) + "&field2=" + String(read\_lig());

sendCommand("AT+CIPMUX=1", 5, "OK");

sendCommand("AT+CIPSTART=2,\"TCP\",\"" + HOST + "\",80", 15, "OK");

sendCommand("AT+CIPSEND=2," + String(getData.length() + 4), 10, ">");

esp8266.println(getData);

countTrueCommand++;

delay(10000);

sendCommand("AT+CIPCLOSE", 5, "OK");

if (mos\_val > 800){ digitalWrite(pump, HIGH);

delay(5000);

digitalWrite(pump, LOW);}

if (lig\_val < 780) {digitalWrite(led, HIGH);}

else {digitalWrite(led, LOW);}

Serial.println();}

int read\_lig() {

lig\_val = analogRead(LDR);

Serial.print("Light value: ");

Serial.println(lig\_val);

return lig\_val;}

int read\_mos() {

mos\_val = analogRead(MOS);

Serial.print("Moisture value: ");

Serial.println(mos\_val);

return mos\_val;}

void sendCommand(String command, int maxTime, char readReplay[]) {

Serial.print(countTrueCommand);

Serial.print(". at command => ");

Serial.print(command);

Serial.print(" -> ");

while (countTimeCommand < (maxTime \* 1)) {

esp8266.println(command);

if (esp8266.find(readReplay)){

found = true;

break;}

countTimeCommand++;}

if (found == true){

Serial.println("OK");

countTrueCommand++;

countTimeCommand = 0;}

else {

Serial.println("Fail");

countTimeCommand = 0;}

found = false;

}