Report

Smart Plant Watering System

Member

Teeranai Sangtaera No.65125056
Thanaphat tenghirun No.65126955

This report is part of the course Microprocessors and Embedded Systems Walailak University term 2 Year 2024

objectives

1. Soil moisture measurement: The system uses a soil moisture sensor to monitor the soil moisture level.

2. Watering control: The system uses an ESP32 microcontroller. to analyze data from sensors and control the water pump to water the plants when necessary.

3. Status tracking: The system will display soil moisture information and watering status via the LCD screen or mobile application.

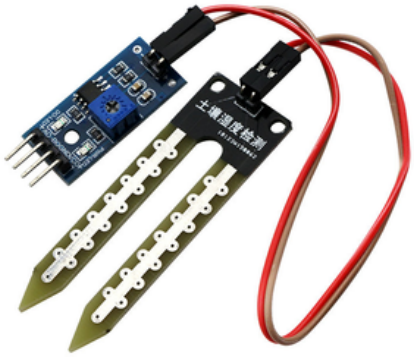
Materials

1. IOXESP32+
2. 5V solid state relay
3. Soil Hygrometer Humidity Detection Module Moisture Water Sensor
4. Mini Micro Submersible DC 2.5-6V
5. Program Arduino

Local : https://th.aliexpress.com/item/32790059510.html

Local : https://components101.com/switches/

Local : https://www.artronshop.co.th/product/59/



Local :https://www.mcucity.com/product/722/

Local :https://www.arduino.cc/en/software

IOXESP32+

IOXESP32+ is an ESP32 development board that uses the latest chip ESP32 ECO V3. Uses 3.3V power supply 700mA (increased from the normal version). Uses FTDI (FT231XQ) to upload programs. Uses USB-C to supply power and upload programs. Change the RESET and IO0 switches to be easier to press. But it still has the same pin position as the normal IOXESP32 board and the NodeMCU-32S, Node32 Lite, making it usable with the original additional board (Shield) and base (Base).

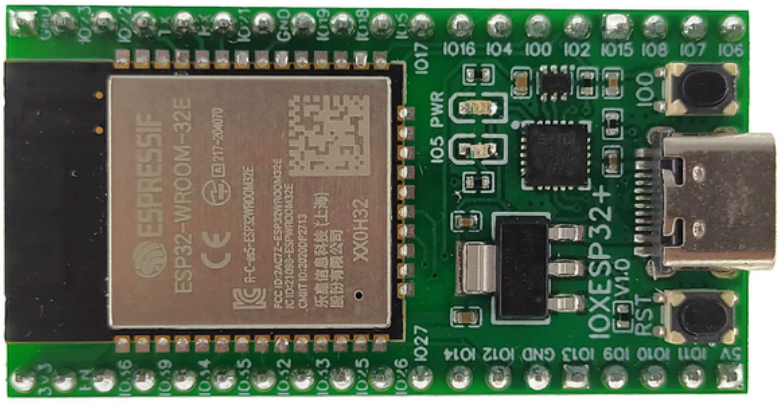


Figure 1 IOXESP32+ board above

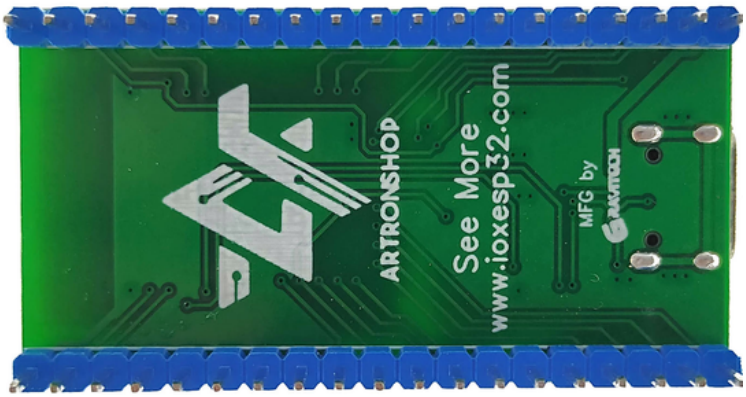


Figure 2 IOXESP32+ board below

The IOXESP32+ isn't a person with duties, but rather a development board for electronics hobbyists and professionals. Here's a breakdown of what it is and how it works:

What is it?

- The IOXESP32+ is a development board built around the ESP32 microcontroller chip.
- This chip integrates Wi-Fi and Bluetooth connectivity, making it useful for creating internet-connected projects.
- The board includes additional components like voltage regulators, buttons, and connectors for attaching other electronics.

How does it work?

- You write code for the ESP32 chip using software like Arduino IDE or MicroPython.
- This code is uploaded to the board through a USB cable.
- The ESP32 chip then executes the code, controlling any attached electronics and interacting with Wi-Fi or Bluetooth networks.

What are the principles?

- The IOXESP32+ follows the principles of open-source hardware and software.
- This means the design and code are openly available for anyone to modify and share.
- This makes it a versatile platform for learning, experimentation, and creating innovative projects.

Specifications of the IOXESP32+ board

- It is an ESP32 development board, 32-bit microcontroller, 2 cores, speed 240 MHz.
- Comes with built-in WiFi (2.4G) and Bluetooth 4.2.
- There is a program storage space (Flash) size 4 MB and RAM (SRAM) size 520 kB.
- Supports connecting 2 channels of I2C, 2 channels of SPI, 3 channels of UART, 2 channels of I2S, 16 channels of ADC (8 channels can be used if using WiFi), 2 channels of DAC, 1 channel of CAN.
- Uses 3.3V voltage as the main operating voltage. There is an LDO IC that converts 5V power from USB-C to 3.3V to power the entire board circuit.
- PCB antenna on the ESP32 module (no need to connect additional WiFi antennas)
- Use genuine equipment for the entire board. Order directly from the manufacturer

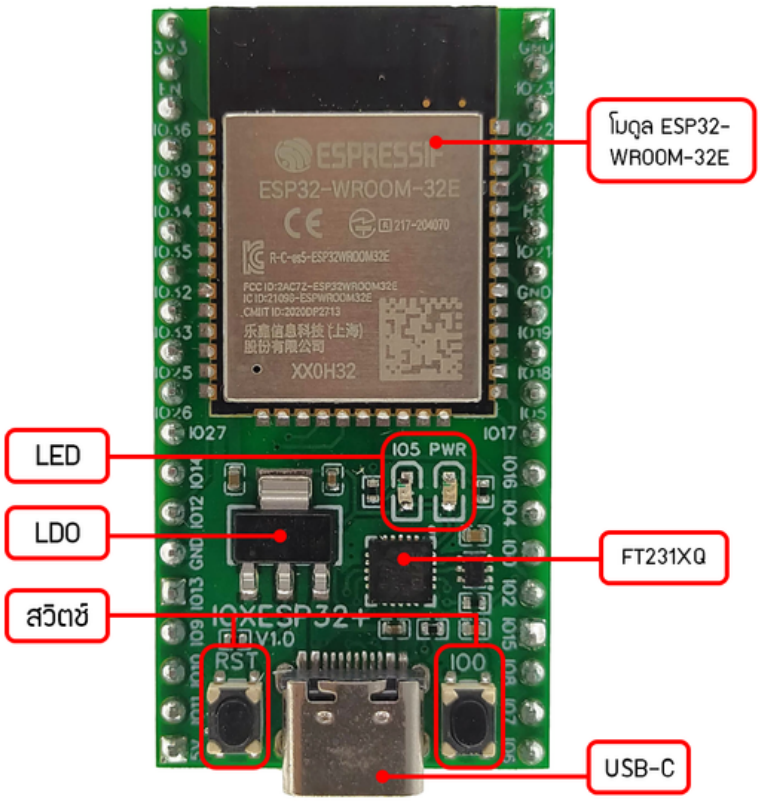


Figure 3 Information

5V solid state relay

A 5V solid state relay (SSR) is an electronically controlled switch used to turn on and off AC or DC loads using a low-voltage DC signal. Unlike traditional mechanical relays that use electromagnets to operate, SSRs use semiconductors to achieve switching, making them more efficient, reliable, and silent.

Figure 4 Example

Here's how a 5V solid state relay works

1. Input: A low-voltage DC signal (typically 5V) is applied to the control input of the SSR.
2. Switching: This signal activates an internal semiconductor, such as a triac or MOSFET, which allows current to flow to the output.
3. Output: The output of the SSR is connected to the AC or DC load that you want to control. When the input signal is present, the load is turned on. When the input signal is removed, the load is turned off.

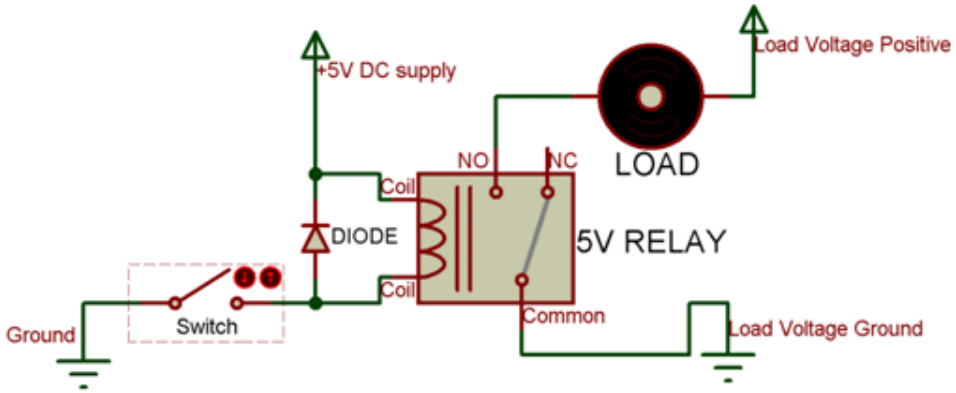


Figure 5 Information

Specifications of the 5V relay board

- Trigger Voltage (Voltage across coil) : 5V DC
- Trigger Current (Nominal current) : 70mA
- Maximum AC load current: 10A @ 250/125V AC
- Maximum DC load current: 10A @ 30/28V DC
- Compact 5-pin configuration with plastic moulding
- Operating time: 10msec Release time: 5msec
- Maximum switching: 300 operating/minute (mechanically)

Applications of Relay

- Commonly used in switching circuits.
- For Home Automation projects to switch AC loads
- To Control (On/Off) Heavy loads at a pre-determined time/condition
- Used in safety circuits to disconnect the load from supply in event of failure
- Used in Automobiles electronics for controlling indicators glass motors etc.

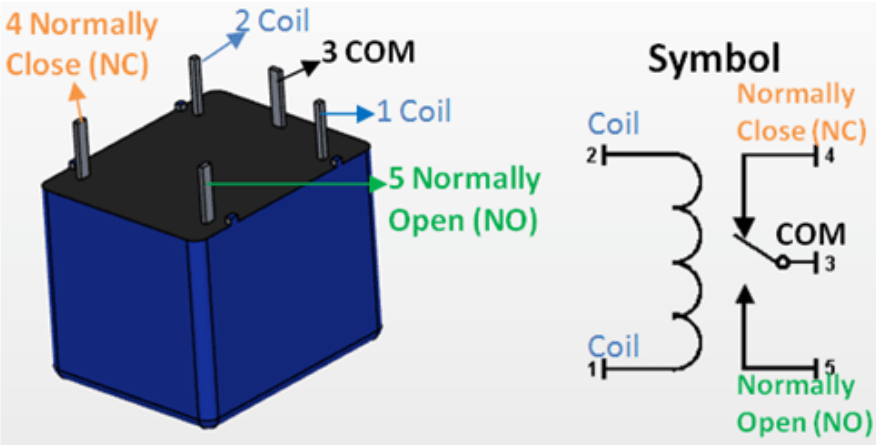


Figure 6 Information_2

Soil Moisture Sensors

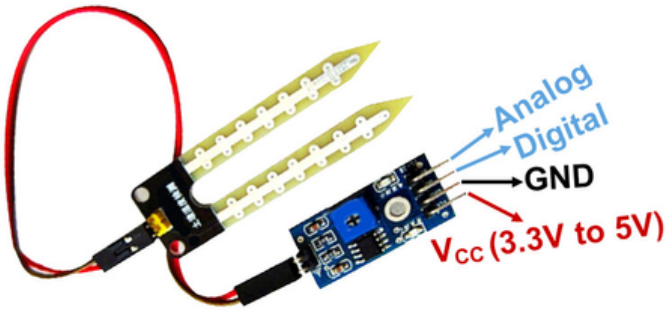


Figure 7 Information

This soil moisture sensor module is used to detect the moisture of the soil. It measures the volumetric content of water inside the soil and gives us the moisture level as output. The module has both digital and analog outputs and a potentiometer to adjust the threshold level.

Soil Moisture Sensor Module Features & Specifications

- Operating Voltage: 3.3V to 5V DC
- Operating Current: 15mA
- Output Digital - 0V to 5V, Adjustable trigger level from preset
- Output Analog - 0V to 5V based on infrared radiation from fire flame falling on the sensor
- LEDs indicating output and power
- PCB Size: 3.2cm x 1.4cm
- LM393 based design
- Easy to use with Microcontrollers or even with normal Digital/Analog IC
- Small, cheap and easily available

Brief about Soil Moisture Sensor Module

This Moisture sensor module consists of a Moisture sensor, Resistors, Capacitor, Potentiometer, Comparator LM393 IC, Power and Status LED in an integrated circuit.

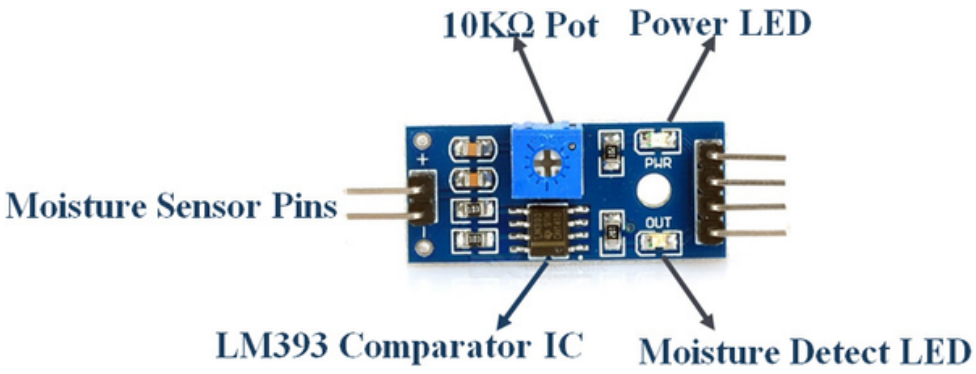


Figure 8 Information_2

How to Use Soil Moisture Sensor Module

Moisture sensor module consists of four pins i.e. VCC, GND, DO, AO. Digital out pin is connected to the output pin of LM393 comparator IC while the analog pin is connected to Moisture sensor. The internal Circuit diagram of the Moisture sensor module is given below.

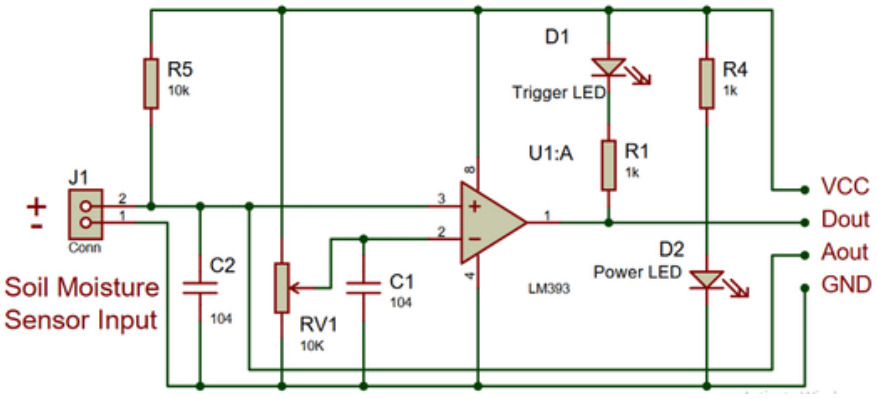


Figure 9 Logic

Using a Moisture sensor module with a microcontroller is very easy. Connect the Analog/Digital Output pin of the module to the Analog/Digital pin of Microcontroller. Connect VCC and GND pins to 5V and GND pins of Microcontroller. After that insert the probe inside the soil. When there is more water presented in the soil, it will conduct more electricity that means resistance will be low and the moisture level will be high.

Mini Micro Submersible DC 2.5-6V

The 1 PC Mini Micro Submersible DC 2.5-6V refers to a small, submersible water pump powered by a direct current (DC) voltage between 2.5 and 6 volts. It's commonly used in various applications requiring a tiny pump, such as:

- Aquariums and small fountains: Circulating water and creating water features.
- DIY projects: Building miniature water cooling systems, science experiments, or decorative projects.
- Model making: Powering miniature boats or water features in models.
- Hydroponics systems: Delivering nutrient-rich water to plants.

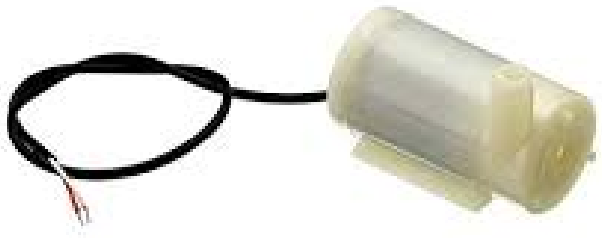


Figure 10 Example

Function:

- Pumps water when submerged and powered by a DC power source between 2.5 and 6 volts.
- Can lift water to a height ranging from 40 to 110cm (depending on voltage and model).
- Flow rate typically between 80 and 120 liters per hour.
- Often made of ABS plastic for durability and water resistance.
- Has a low noise level due to its brushless motor design.

How to use it:

1. Connect the pump: Attach the red wire to the positive terminal of your power source and the black wire to the negative terminal.
2. Submerge the pump: Ensure the entire pump is submerged in water.
3. Power it on: Supply the required DC voltage between 2.5 and 6 volts.
4. Adjust flow (optional): Some models have adjustable flow rates.

Specifications:

- newly
- DC voltage: 2.5-6V
- Working current: 130-220mA
- Power: 0.4-1.5W
- Maximum lift: 40-110cm/15.75"-43.4"
- Flow rate: 80-120L/hr.
- Outer diameter of water outlet: approx. 7.5mm/0.3"
- Inner diameter of water outlet: Approx. 4.7mm/0.18"
- Diameter: Approx. 24mm/0.95"
- Length: Approx. 45mm/1.8"
- Height: Approx. 33mm/1.30"
- Wire length: about 15-20 cm. (Red: "+", Black (White): " - ")
- Material: Engineering plastic
- Driving mode: Brushless DC design, magnetic driving
- Continuous working life 500 hours

Arduino IDE

Makers, students & professionals have been using the classic Arduino IDE (Integrated Development Environment) ever since Arduino was born.
The Arduino IDE 2 is an improvement of the classic IDE, with increased performance, improved user interface and many new features, such as autocompletion, a built-in debugger and syncing sketches with Arduino Cloud.

The Arduino IDE 2 features a new sidebar, making the most commonly used tools more accessible.

Figure 11 Arduino IDE 2

- Verify / Upload - compile and upload your code to your Arduino Board.
- Select Board & Port - detected Arduino boards automatically show up here, along with the port number.
- Sketchbook - here you will find all of your sketches locally stored on your computer. Additionally, you can sync with the Arduino Cloud, and also obtain your sketches from the online environment.
- Boards Manager - browse through Arduino & third party packages that can be installed. For example, using a MKR WiFi 1010 board requires the
- Arduino SAMD Boards
- package installed.
- Library Manager - browse through thousands of Arduino libraries, made by Arduino & its community.
- Debugger - test and debug programs in real time.
- Search - search for keywords in your code.
- Open Serial Monitor - opens the Serial Monitor tool, as a new tab in the console.

Features

The Arduino IDE 2 is a versatile editor with many features. You can install libraries directly, sync your sketches with Arduino Cloud, debug your sketches and much more. In this section, some of the core features are listed, along with a link to a more detailed article.

Sketchbook

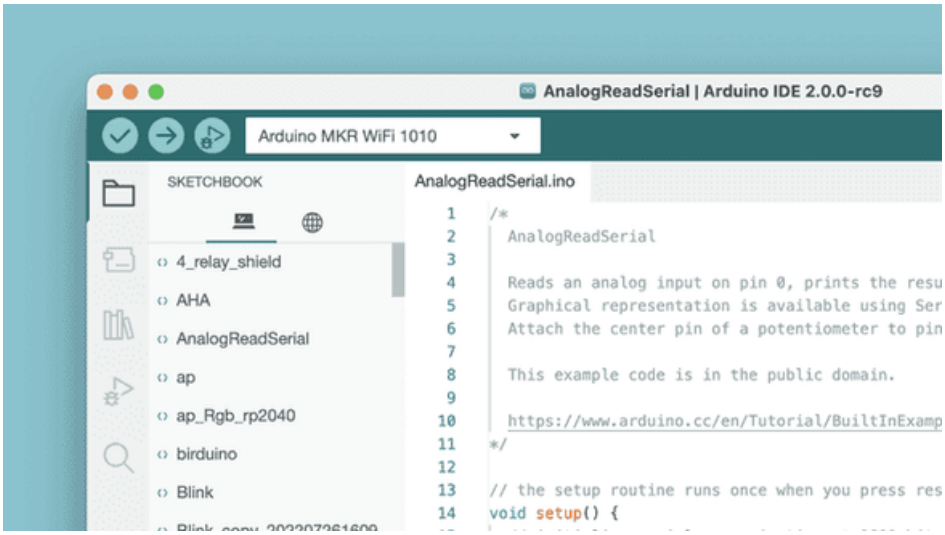


Figure 12 Arduino Sketchbook

Boards Manager

Figure 13 Arduino Boards Manager

Library Manager

Figure 14 Arduino Library Manager

Serial Monitor

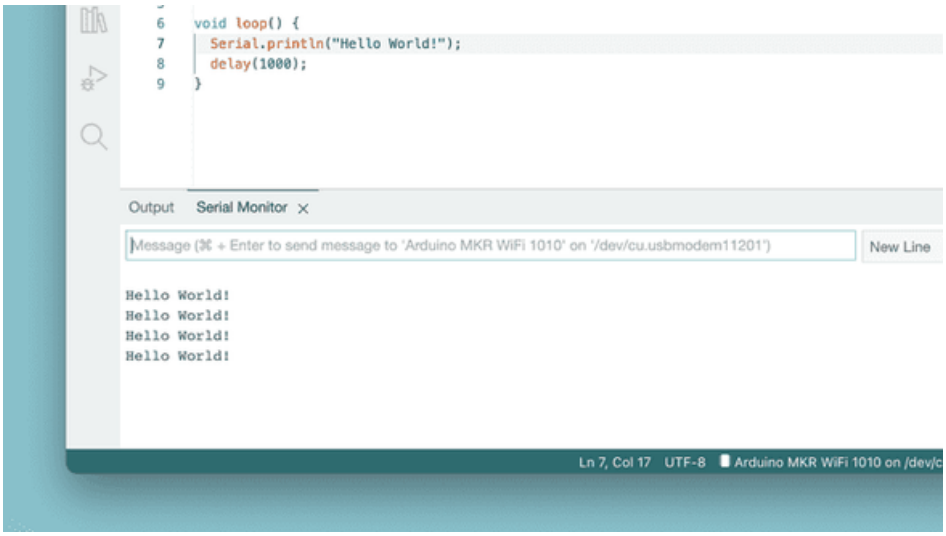


Figure 15 Arduino Serial Monitor

Circuit Diagram

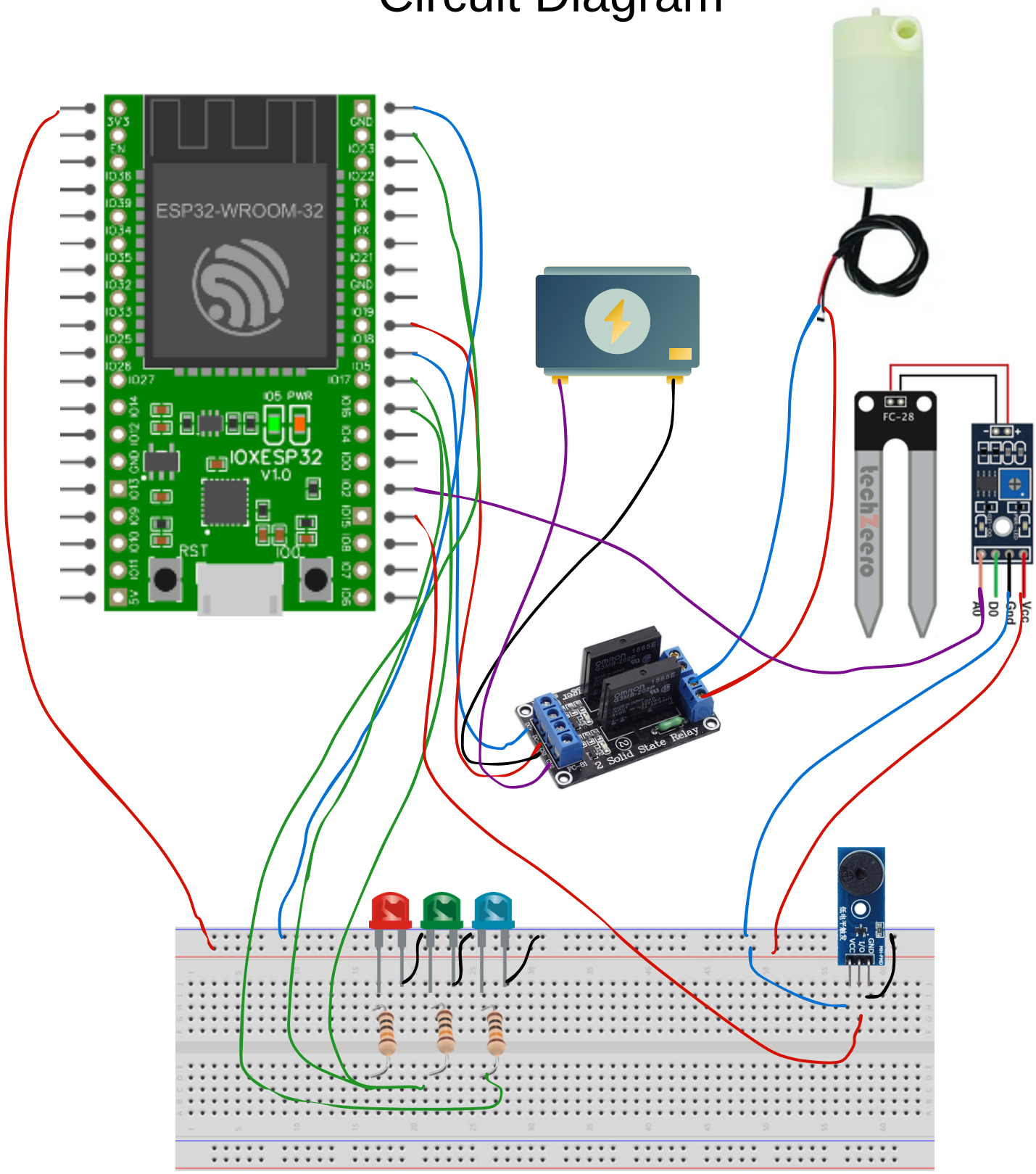


Figure 16 Curcuit Diagram

results and discussions

Curcuit

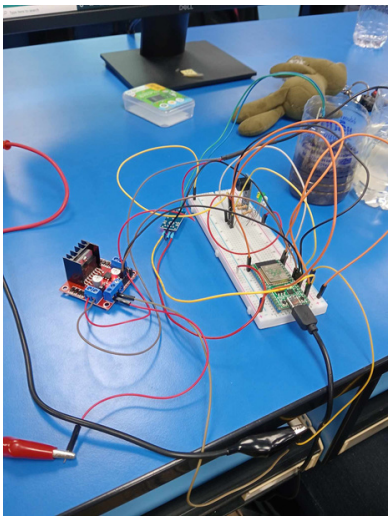


Figure 17 Result 1

Figure 18 Result 2

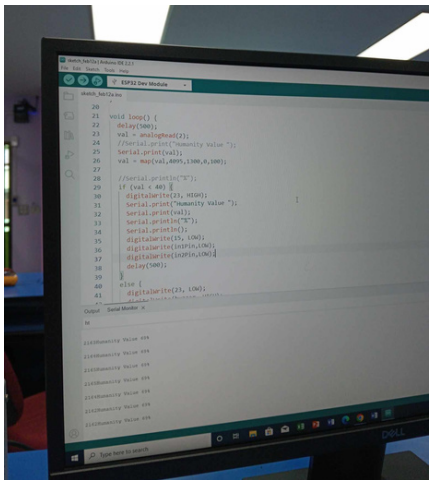


Figure 19 Result 3

This circuit will use the ESP32, Buzzer, water pump, Soil moiture, which first starts by connecting VCC 3.3V and connecting GND of the ESP32 board. Then the LED will use GPIO 16 pin as green, GPIO 17 as red and GPIO. 23 yellow, then connect the wire to the resistor before Connect a resistor to the LED and the negative pin of the LED to GND.

Next, the connection to Soil Moiture, which selects GPIO 2 pin and the other pin connects to GND, is completed. Next, the Buzzer must connect VCC and GND, while the In pin chooses to connect to GPIO 15, where Buzzer LOW is on, HIGH is off.

Then connect the relay which will use GPIO pins 18,19 and connect VCC to GND. Then the relay will connect to the Water pump to control the Water pump if we give the input in a clockwise direction. Will be able to send water forward But if we're late Will send suction water back

Figure 20 Result 4

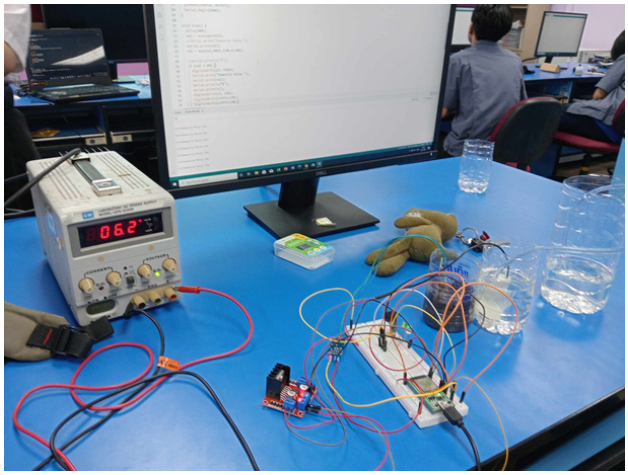


Figure 21 Result 5

Then, the results obtained from testing the entire circuit are The first thing is to try to make the humidity lower than 40%. You will find that the sensor has measured it to be lower than 40%. Then the program starts working and everything is according to the conditions, that is, there is a value showing less than 40% and there is a loud warning sound. up, then the water pump starts working. The relay will control the operation of the Water pump.

Then experiment with humidity at 41%-60% to see that the conditions are met. and the red light turns on After that, no further work occurred because it was still in normal condition.

Later, if the humidity value is 60% or more, the green light will turn on and the relay will order the water pump to stop working.

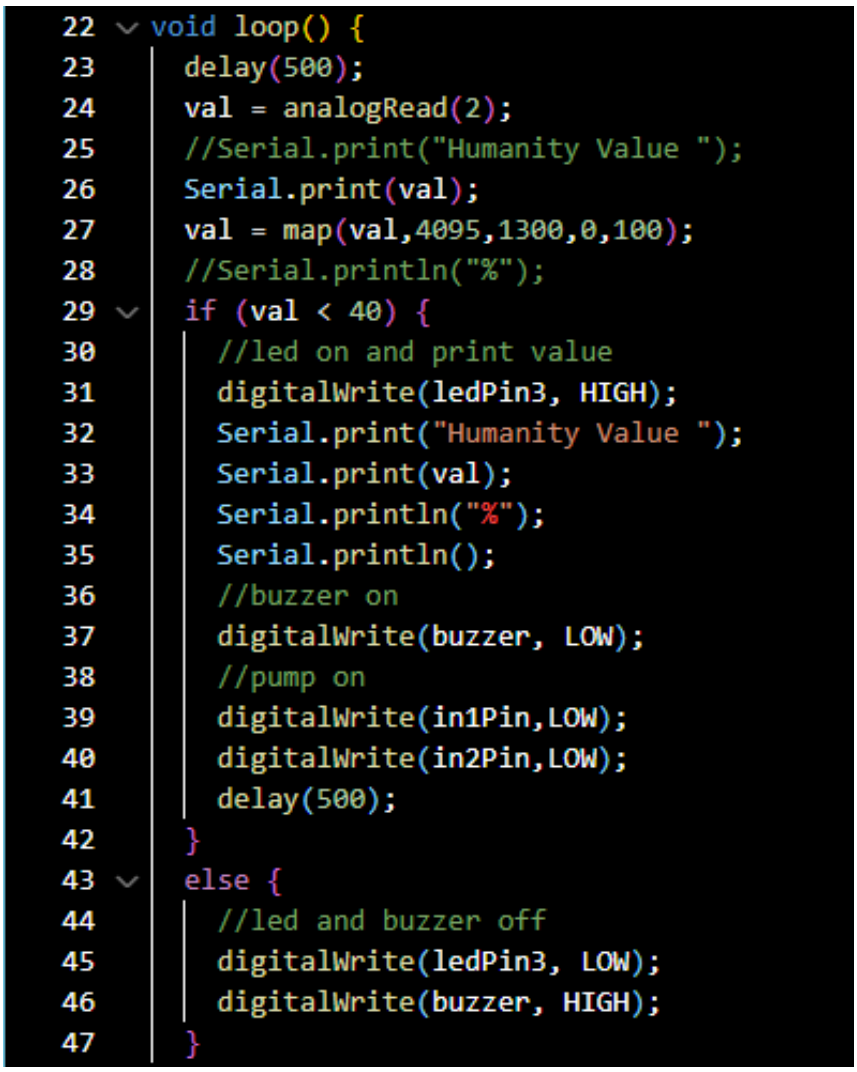
It can be seen that all tested The actual results met the expected needs. No errors occurred. unless The electrical power that is supplied to the water pump to work may need to be higher than 5V as seen from the power supply in the picture, but overall the results are quite good.

Explain about Code

```
1   int ledPin1 = 16; //LED 16 Green
2   int ledPin2 = 17; //LED 17 White
3   int ledPin3 = 23; //LED 23 Yellow
4   const int in1Pin = 18;  // H-Bridge input pins
5   const int in2Pin = 19;
6   // GPIO 18,19 = waterpump
7   int buzzer=15; // Buzzer 15 low = 1 , high = 0
8   int analogPin = 2;
9   int val = 0;
10  void setup() {
11    // sets the pin as output
12    pinMode(ledPin1, OUTPUT);
13    pinMode(ledPin2, OUTPUT);
14    pinMode(ledPin3, OUTPUT);
15
16    pinMode(buzzer,OUTPUT);
17    pinMode(in1Pin, OUTPUT);
18    pinMode(in2Pin, OUTPUT);
19    Serial.begin(9600);
20  }
21
```

Figure 22 Code 1

This section of code is the setup of various values that will all be used in running the program. Lines 1-9 are setting values from LED, Input-Output, Buzzer, relay, all connected to the GPIO of the ESP32 board. Then, lines 10-19 are settings. PinMode of each channel, which is Input or Output?



```
22  void loop() {
23    delay(500);
24    val = analogRead(2);
25    //Serial.print("Humanity Value ");
26    Serial.print(val);
27    val = map(val,4095,1300,0,100);
28    //Serial.println("%");
29    if (val < 40) {
30      //led on and print value
31      digitalWrite(ledPin3, HIGH);
32      Serial.print("Humanity Value ");
33      Serial.print(val);
34      Serial.println("%");
35      Serial.println();
36      //buzzer on
37      digitalWrite(buzzer, LOW);
38      //pump on
39      digitalWrite(in1Pin,LOW);
40      digitalWrite(in2Pin,LOW);
41      delay(500);
42    }
43    else {
44      //led and buzzer off
45      digitalWrite(ledPin3, LOW);
46      digitalWrite(buzzer, HIGH);
47    }
```

Figure 23 Code 2

This section is the code that repeats itself as the program runs. Because it is in the Loop function, in each new loop there will be The delay is 0.5 seconds. Line 24 is used to set a variable to receive the value of the sensor that measures humidity. Then, line 26 in shows the value of the data that is still raw. Line 27 is a map of the value to match the sensor. With the conversion to percentage, the highest value is 4095 and the lowest 1300 and set 0 - 100% to match the front value.

Next, the if else condition asks for the condition: If the humidity is less than 40%, when should it be given? The yellow light is on. and shows the humidity value and has a loud buzzer sound as a warning along with the water pump that will start pumping water. As for the else condition, if it is not true at any time, turn off the LED and turn off the buzzer.

```
48  if (val > 40 && val < 60) {
49    //led on and print value
50    digitalWrite(ledPin2, HIGH);
51    Serial.print("Humanity Value ");
52    Serial.print(val);
53    Serial.println("%");
54    Serial.println();
55    //buzzer off
56    digitalWrite(buzzer,HIGH);
57    delay(500);
58  }
59  else {
60    //ledand buzzer off
61    digitalWrite(ledPin2, LOW);
62    digitalWrite(buzzer, HIGH);
63  }
```

Figure 24 Code 3

In this section, in the if else condition, if the humidity value is between 40% - 60%, the red light will turn on. and show the thickness value and the buzzer is not turned on. The else condition is to have all LEDs and buzzer turned off.

```
64  if (val > 60) {
65    //led on and print value
66    digitalWrite(ledPin1, HIGH);
67    Serial.print("Humanity Value ");
68    Serial.print(val);
69    Serial.println("%");
70    Serial.println();
71    //buzzer off
72    digitalWrite(buzzer, HIGH);
73    //pump off
74    digitalWrite(in1Pin,HIGH);
75    digitalWrite(in2Pin,LOW);
76    delay(500);
77  }
78  else {
79    //led and buzzer off
80    digitalWrite(ledPin1, LOW);
81    digitalWrite(buzzer, HIGH);
82  }
83  delay(100);
84 }
```

Figure 25 Code 4

In this section, in the if else condition, if the humidity value is more than 60%, the green light will turn on. and shows humidity values and the buzzer is not used, it will order the water pump to stop working. The else condition will order the LED and Buzzer to be turned off.

Circuit testing video

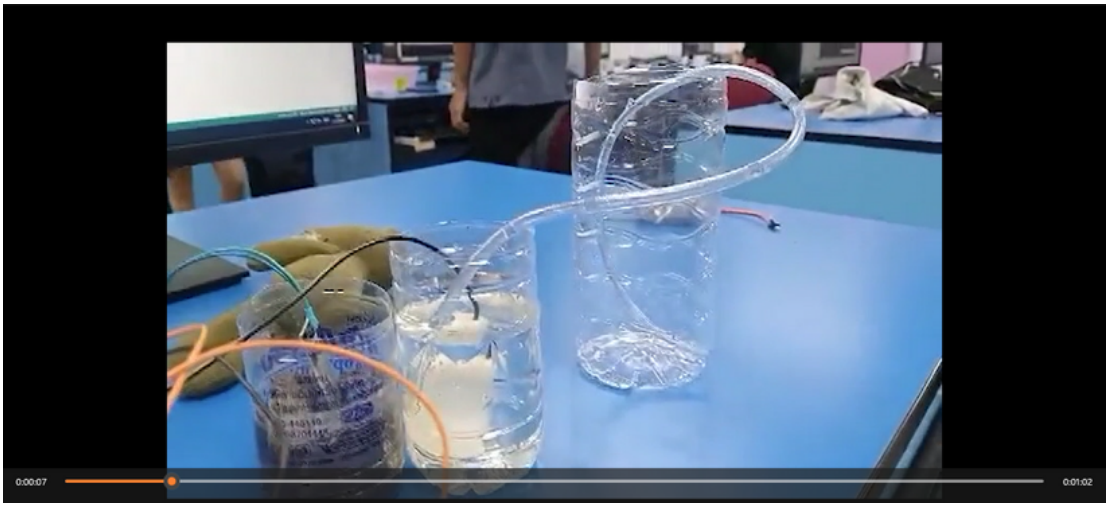


Figure 26 Video

Link : https://drive.google.com/file/d/1rTq6piaXnBBZGzGWGgAqTSOUGHaSvrrD/view?usp=sharing

Summary

The message describes a circuit utilizing an ESP32 microcontroller to monitor soil moisture levels and control a water pump accordingly, along with additional features such as LED indicators and a buzzer for alerts. Here's a breakdown of how it works:

1. **Initialization**: The ESP32 board is powered by connecting VCC to 3.3V and GND to ground.

2. **LED Setup**: LEDs are connected to GPIO pins (16 for green, 17 for red, and 23 for yellow). Each LED is connected through a resistor to limit current and then to ground.

3. **Soil Moisture Sensor**: The soil moisture sensor is connected to GPIO 2 pin and ground.

4. **Buzzer Connection**: The buzzer is connected by powering VCC and GND and connecting the In pin to GPIO 15.

5. **Relay Setup**: The relay is controlled by GPIO pins 18 and 19, and VCC is connected to GND. The relay controls the water pump's operation.

6. **Water Pump Operation**: The relay is used to control the water pump. When activated in a clockwise direction, water is pumped forward. If deactivated, water is suctioned back.

7. **Functionality**:
   - When soil moisture is below 40%, the program triggers actions. The buzzer emits a warning sound, and the water pump starts via the relay.
   - If moisture is between 41% and 60%, the red LED turns on, indicating normal conditions. No further action is taken.
   - If moisture is 60% or higher, the green LED turns on, and the relay stops the water pump.

8. **Conclusion**: The circuit's functionality meets the intended requirements, providing effective monitoring and control of soil moisture levels. However, it's noted that the water pump may require a power supply higher than 5V, and this should be considered for proper operation.

In summary, this setup utilizes the ESP32 microcontroller along with various components to create an automated system for maintaining optimal soil moisture levels, incorporating feedback mechanisms such as LED indicators and a buzzer for user alerts.

Bibliography

Arduino cc. (2024). *Arduino Documentation.*

Retrieved from https://docs.arduino.cc/

AliExpress. (2023). *1 PC Mini Micro Submersible DC 2.5-6V.*

Retrieved from https://th.aliexpress.com/item/32790059510.html

Components. (2023). *Soil Moisture Sensor Module.*

Retrieved from https://components101.com/modules/

Madetech. (2012). *Soil Moisture Datasheet.*

Retrieved from https://www.instrumart.com/assets/smr110-Datasheet.pdf

Artronshop. (2024). *IOXESP32*+ บอร์ดพัฒนา *ESP32* รุ่นอัพเดเวล.

Retrieved from https://www.artronshop.co.th/product/59/ioxesp32

Artronshop. (2024). *Soil Hygrometer Humidity Detection Module.*

Retrieved from https://www.mcucity.com/product/722/