APRIL 27, 2021

CAPSTONE PROJECT REPORT

SMART FARMING

CREATED BY:

Abhisha Bhesaniya
Simran Padaniya
Vijay Patel

Abstract

The “Smart Farming” is a Wi-Fi-based project that helps people to monitor the soil moisture and humidity during its dry or wet state and control the amount of water delivered to the crop without any manual interventions.

The system will control the amount of water to the soil. The system maintains logs of soil moisture, temperature, and humidity for future reference. All logs are maintained in a database, which is located on a local or remote webserver.

User of this system can see the database with mobile or browser using ARC(Advance Rest Client) application from remote locations only condition is user should be connected with internate.The application will also be able to display logs and detailed information as needed by the user. The system can be configured based on the type of crops to be cultivated.

The system is efficient in saving water, money, and electricity and will operate automatically without any manual interference. The system can be modified for harvesting electricity and water from natural resources. The system is very reliable. The system is very easy to maintain, and new nodes can be easily added to the entire system. The system is secure from vandalism. The system is economic and has a capability to cover large field area.

Acknowledgement

We extend our gratitude to our management for having provided us with all facilities to build our project successfully. It is a privilege for us to have been associated with Conestoga College, our guide. We express our sincere thanks to our honorable President **John Tibbits,** for providing us with required amenities.

We express our gratefulness to **Mr. Mike Jarabek,** Project mentor, Professor and Program Co-ordinator, Embedded Systems Development for providing us kind advice during the development of the project. We have been greatly benefited by his valuable suggestions and ideas.

We are committed to place our heartfelt thanks to all teaching and non-teaching staff members and all the noble hearts that gave us immense encouragement towards the completion of our project.

Table of Contents

List of Figures:

List of Tables:

Introduction / Problem Statement

Now a days everything has been innovated. An IoT has changed the work pattern it was before. Tools are getting smarter and human labour is decreasing. Nonliving things get speech because of IoT and tell us what is happening around us even if we are not there. IoT has evolved every walk of our life and farming is one of them.

Canada has larger number of farms and arear of this farm is also large so to take care of this farm is very difficult without involvement of technology along with traditional farming practice. Moreover because of the climate of Canada most of the agriculture activities are done in the green house farming. So, in green house farming as we are creating artificial environment to grow the crop we should be more care full in maintaining the environment which helps crop to grow. For that we have to monitor all the parameters continuously and have log of all the data into database so when something goes wrong, we can tell where the problem for example was whether we haven't irrigate the crop on time or humidity or temperature was not perfect for the crop and many more. Many farms have sensors that is used just to sense the value. They are not IoT based they are just like normal sensors. They just sense the parameters and show them on display attached with sensor it does not create any log. Also, the issue is that it is not automated, so we need to go in person if we want the data from the sensor.

The problem with this type of arrangement is as follows:

- Time consuming.
- Difficult to store data.
- Manual intervention is needed.

Sometimes it may happen that due to time issue person may not be able to come to field in person. Thus, this may affect the field. Also, if we are not storing data anywhere thus it becomes difficult to track the quality of soil.

Proposed Solution

The solution for above mentioned problem is Smart Farming. Smart farming is an IoT based project in which we are monitoring the different parameter of farm like temperature and humidity of air and moisture level of soil at this point of time, but we can add more sensor as per user's requirement. All these sensors are connected with Wi-Fi module which is ESP8266 and send the data to the access point which is serially connected with raspberry pi. Once the raspberry pi receive data on serial port, our python code will read the data from serial port and process on the received data and send it into database along with date and time. This is a case with DHT 22 sensor but when it comes to soil moisture sensor when it sends data to access point then access point will check whether the soil needs water, based on the threshold value we have set for dry and wet soil and if soil need access point send command to DC motor to turn on to irrigate the water to crop and when soil moisture comes at required level access point send command to DC motor turn off. Meanwhile soil moisture sensor and DC motor send its status to access point and access point forward it to raspberry pi and raspberry pi will update it into database.

All the sensors send data in predefined frame format which is as follow:

$	Sensor type	Sensor no	No of data	Data 1	Data 2	#	\r

All the elements are separated with coma so that it is easy to separate it in python in raspberry pi when it sends data to database.

Table 1. Table explaining about the elements of pre-defined frame format.

Frame field	Explanation
$	Starting of string
Sensor type	Which sensor is sending data. It will send, 1 – if soil moisture 2 – humidity temperature sensor and 3 – DC water pump
Sensor number	Which sensor is sending data. For example, If we have 2 soil moisture sensors, then which one is sending data. 1 or 2.
No of data	How many data the sensor is sending. For example, DHT22 is sending 2 data. Humidity and Temperature then it will send 2 in the string.
Data 1	It will send the first data of the sensor.
Data 2	It will send the second data, if the number of data is 2.
#	Denotes end of the string
/r	To let access-point know about the end of reading string

We are using database which is created by our professor for storing data. We are using json format to send data into database. We can access the database using ARC application form chrome.

Block Diagram.

Following diagram is showing the functional diagram of our solution,

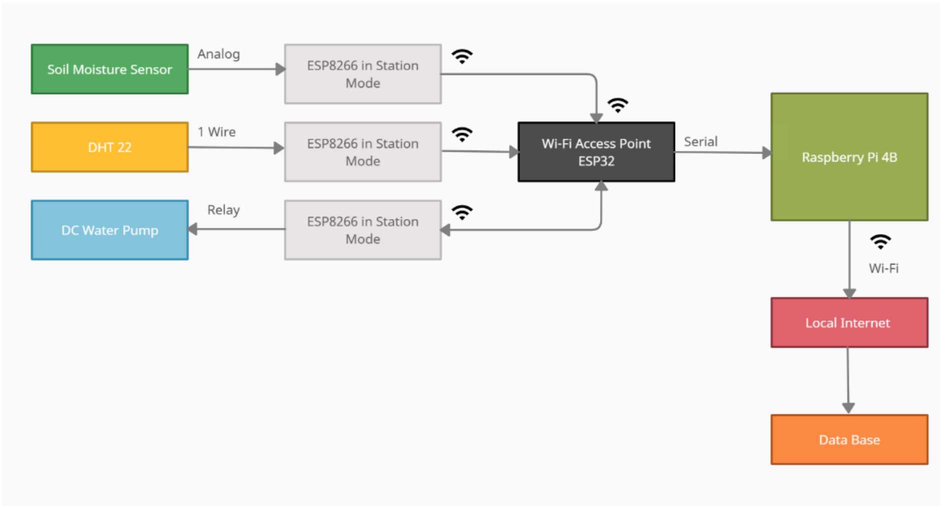

Figure 1. Block diagram of the system.

Flow Chart.

Following flowchart will show the show the flow of our product.

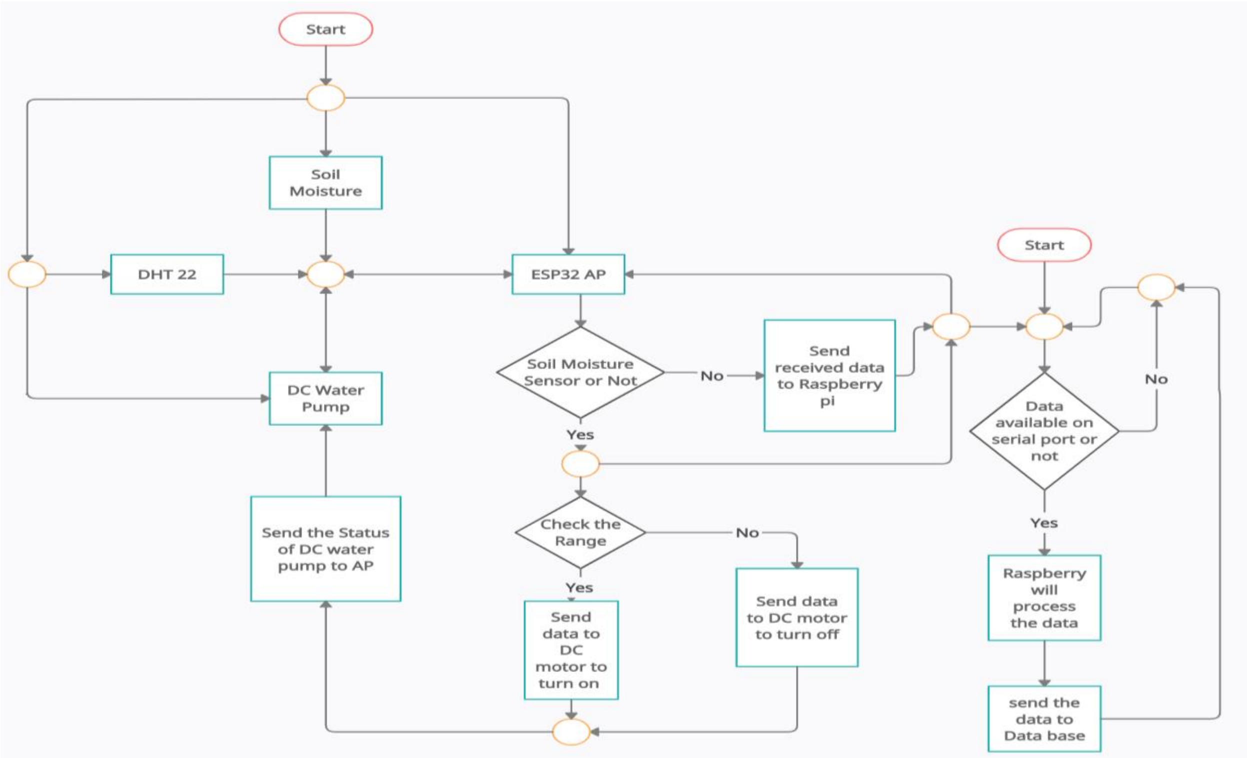

Figure 2. Flowchart of the system.

Technology/Key components

For this project we have used the following technologies and components:

Raspberry Pi4b:

The Raspberry Pi is a little computer that fits in the palm of your hand. Before you can do anything awesome, you need to configure it and install an operating system. Windows, Ubuntu and Raspbian are the OS supported by Raspberry Pi.

Figure 3.Image of Raspberry Pi 4B

("8GB Raspberry Pi 4 on sale now at $75 - Raspberry Pi", 2021)

Table 2. Specification table of raspberry pi.

Features	**Specification**
Processor	Quad core Cortex-A72(ARM v8) 64-bit SoC @ 1.5GHz
SDRAM	4GB
Bluetooth	5.0 and BLE
USB	• 2 USB 3.0 ports • 2 USB 2.0 ports
GPIO header	40 pins
Micro-HDMI	2 (up to 4kp60 supported)
Display port	2-lane MIPI DSI
Camera	2-lane MIPI CSI
Operating temperature	0-50 degrees C ambient

Esp8266

The Development Kit NodeMCU ESP8266 is having GPIO, PWM, IIC, 1-Wire and ADC all in one board. The Nodemcu ESP8266 Wi-Fi Module is a SOC with integrated TCP/IP protocol stack that can give any microcontroller access to your Wi-Fi network. It follow 802.11 b/g/n Wi-Fi protocol. It is having 1MB flash memory. Out power is +19.5dBm in 802.11 b mode. It is having integrated low power 32 bit CPU, we can use it as application processor.

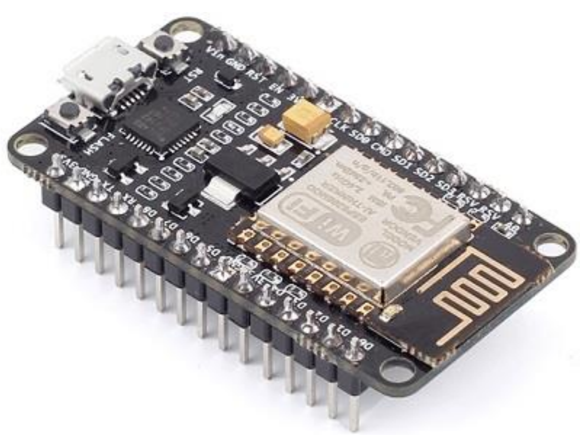

Figure 4 Image of ESP8266

(Santos, 2021)

Esp32

The Development kit NodeMCU ESP32 is having Wi-Fi, Bluetooth V4.2 and BLE all in one board. Moreover, this dev kit has all the peripherals like ADCs, DACs, I^2C, UART, CAN 2.0, SPI, I^2S, RMII, PWM and more. This module has all IEEE 802.11 standard security features like, WFA, WPA/WPA2 and WAPI along with that flash encryption, AES, SHA-2, RSA, Elliptic curve cryptography (ECC), random number generator(RNG). ESP32 has Tensilica xtensa 32-bit LX6 microprocessor, clock frequency is up to 240 Mhz and it has ultra low power co-processor that allows you to do ADC conversions, computation, and level thresholds when in deep sleep.

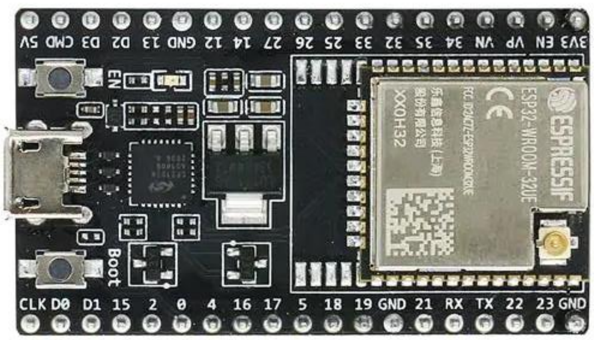

Figure 5 Image of ESP32

Soil Moisture Sensor and Water Pump Kit

This kit has soil moisture sensor Relay to control DC motor, one plastic pipe and connecting wires. Soil moisture sensor is analog sensor it includes capacitive type humidity measurement component. This sensor has ability to provide fast response and cost effective. DC motor is waterproof you can deep entire pump in to water and then it will work as submersible pump.

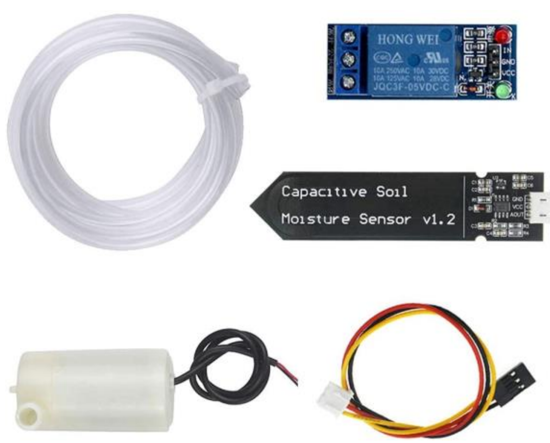

Figure 6. Image of soil moisture sensor and DC water pump kit

DHT22 Sensor

DHT22 is humidity & temperature sensor with digital output. This sensor includes a resistive type humidity measurement component and an NTC temperature measurement component, and connects to a high performance 8 bit microcontroller offering excellent quality, fast response, anti-interference ability and cost effectiveness.

Figure 7 Image of DHT22

Background

To read this document reader must have the knowledge of IoT concept, how Wi-Fi protocol works and what Raspberry Pi is. Reader should have minimum knowledge in microprocessor and microcontroller as our ESP8266 and ESP32 has microcontroller. It controls most of the things in this project.

The person should understand electronic circuits and how to connect sensors with microcontroller. Moreover, Person must have knowledge of reading and understanding the datasheets for sensors used for this project.

The person should have some experience in testing and debugging as it helps them to understand what to do when they face the problem. Along with that they must have knowledge of reading flowchart and block diagram as this will help them to understand the flow of process of this project.

This is all basic requirement for a person to read and understand the document.

Methodology

For our project we have followed following steps:

- We had spent quite few times on deciding the project topic and then we have finalized Smart Farming.
- We had read datasheet of related components that we are planning to use.
- Based on the reading analysis we had finalized following components:
  a. ESP8266
  b. ESP32
  c. DC Water Pump with Soil moisture sensor
  d. DHT22
  e. Raspberry Pi 4.
- We Have installed Arduino IDE in our system and Raspbian in Raspberry Pi.
- After installing Arduino IDE, we have installed required libraries in it.
- After that we have started working on code.
- For coding we have programmed ESP8266 as Station to connect DHT22, DC water pump and Soil moisture sensor.
- Later, we have programmed ESP32 as Access Point.
- After creating the basic network of sensors, we have programmed Raspberry Pi such that it will receive data from access point serially and then split received string.
- Later the required data from string is sent to the Non-SQL mongo db. database with current time and data format using json format.
- After developing the code, we have soldered our components on PCB.
- We were using API Rest Client to see our data from the database.

Result / Analysis

- Basic prototype of smart farming is created with the network of sensors.
- A clear idea of prototype had been developed to use in real world.
- Required software had been installed.
- Proper working code has been developed.
- Sensors were able to talk to Access point using Wi-Fi.
- Code in Raspbian has been developed as per requirement.
- Decision based code for soil moisture sensor and DC motor has been developed.
- Code to sent data over database has been developed.
- All the things are working as per the design.

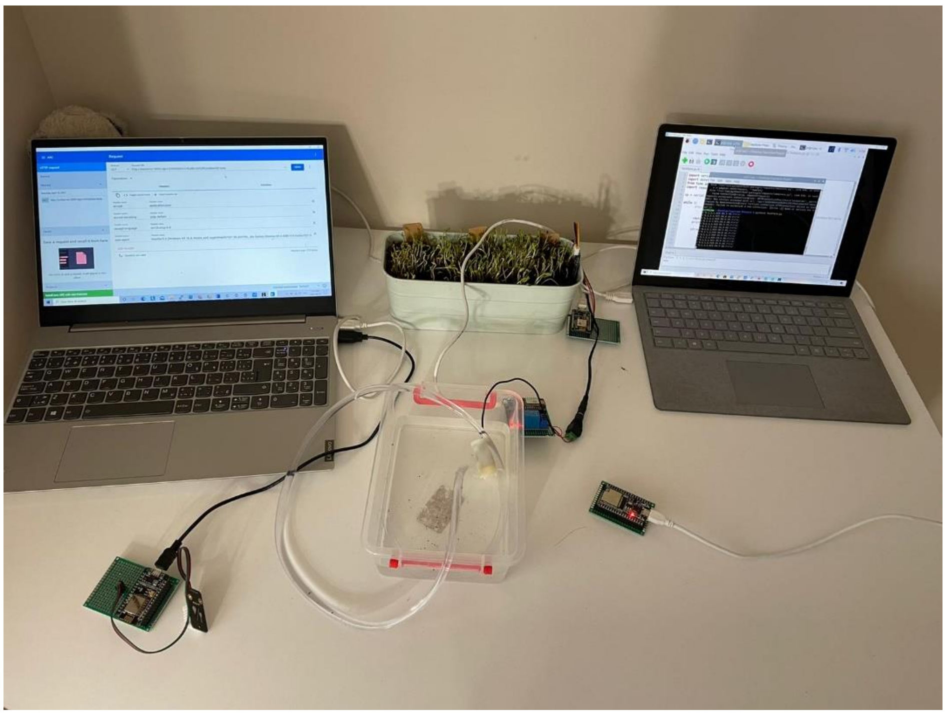

Figure 8. Image of Prototype

Cost Analysis

Table 3: Table of cost analysis.

Sr. NO	ITEM	PRICE(CAD)
1.	Raspberry pi 4B/4GB	86.02
2.	HDMI Cable	12.42
3.	ESP32	16.65
4.	ESP8266 (3 pis set)	30.49
5.	Raspberry pi power supply	12.50
6.	Soil moisture sensor + DC Motor + Relay	20.32
7.	DHT22	14.85
	Total	162.75

Recommendations

- We faced problem in connecting DHT22 with ESP8266 we were not getting data on the esp8266's pin on which the output of DHT22 is connected. And it is because of the wire the wire which we have used to connect was broken from inside so there was no connectivity and because of that we were not getting data. We were only getting nan in the output. So, always check the connecting wire's connectivity before using it.
- In the very beginning when we were trying to connect ESP8266's STA to AP at that time we were not be able to do that our access point was connecting with our phone but not with our STA so we figure out that there is problem with STA and then we came to know that even if we are using ESP8266 in station mode we have to set Wi-Fi mode to station in code.
- Other then that we were facing some syntax error in python while we were separating the string which was received from Access point. We solved this error by reading post on stack overflow. And some pin connection error in DC motor so always read the pin diagram carefully to avoid this kind of mistake.

Future Scope

- At this point of time, we are sending date and time in database in string format, and it takes to much space in database so to overcome this problem we can send time in the format of Epoch time in future.
- Moreover, we can program built in Wi-Fi of raspberry pi as Access point so that we can remove access point from our project, and it will reduce the cost.
- Furthermore, we can add more sensors like sensors to measure quality of water, leaf sensor to measure the heath of crop, weather station so that if there is chances of rain then we can stop irrigation and it will save water other then that we can add camara for the security.
- We can develop android/IOS application so that it will be easy to use our product and can access the database easily.

Appendix / Schematics

Following are the schematics of the sensors which we have developed as a part of our project.

Schematics of DHT22 and ESP8266

This schematic is made for sensing temperature and humidity of air. In this the DHT22 is connected to ESP8266.

Following connections are made:

- Power Supply – 3v3
- Gnd – Gnd
- Signal – D1

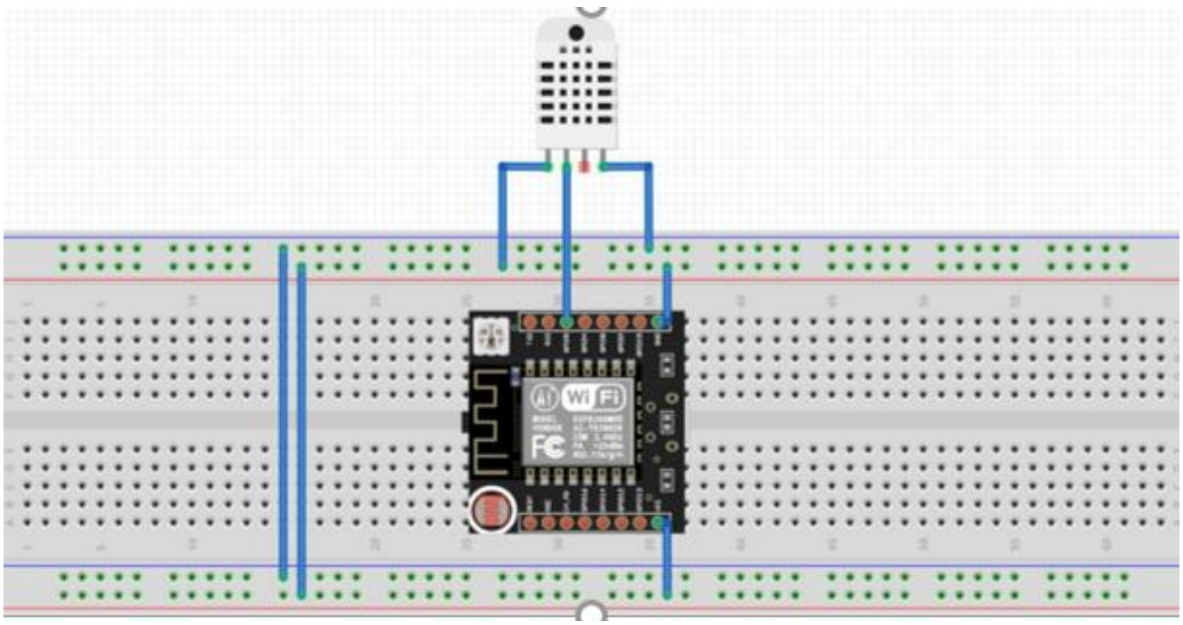

Figure 9. Schematics of DHT22 connected with ESP8266.

Schematics of Soil Moisture Sensor and ESP8266

This schematic is made for measuring Soil Moisture. In this the Soil Moisture sensor is connected to ESP8266.

Following connections are made:

- Power Supply – 3v3
- Gnd – Gnd
- Signal – A0.

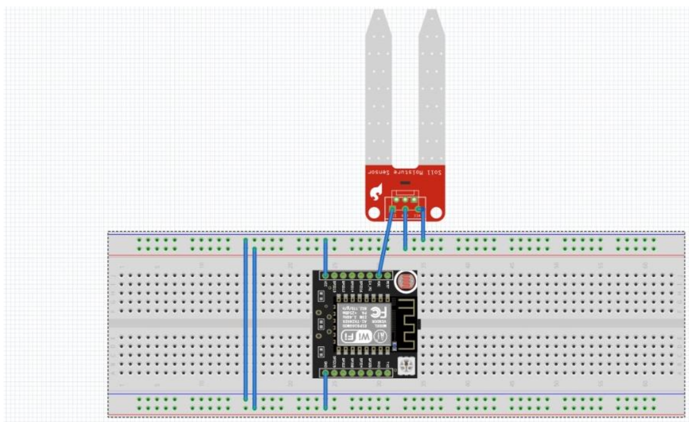

Figure 10. Schematics of soil moisture sensor connected with ESP8266

Schematics of DC water Pump and ESP8266

This schematic is made supplying water to plant. In this the submersible motor is connected to ESP8266 and DC power supply of 5v is given though DC power supply.

Following connections are made on Relay:

- Power Supply – 3v3
- Gnd – Gnd
- Signal – D1.

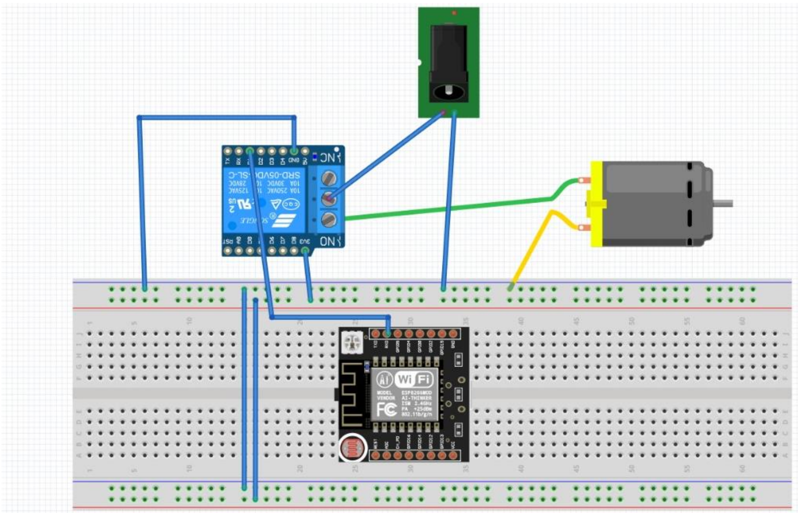

Figure 11. Schematics of DC water pump connected with ESP8266 through relay.

Raspberry pi with ESP32

ESP 32 is connected with raspberry pi on raspberry pi's one USB port.

References

- 8GB Raspberry Pi 4 on sale now at $75 - Raspberry Pi. (2021). Retrieved 26 April 2021, from https://www.raspberrypi.org/blog/8gb-raspberry-pi-4-on-sale-now-at-75/

- Santos, R. (2021). Best ESP8266 Wi-Fi Development Board - Buying Guide 2020. Retrieved 26 April 2021, from https://makeradvisor.com/best-esp8266-wi-fi-development-board/

- (2021). Retrieved 26 April 2021, from https://www.digikey.ca/en/products/detail/schtoeta-engineering-limited/ESP32-DEVKITC-VIE/12091811?utm_adgroup=RF%2FIF%20and%20RFID&utm_source=google&utm_medium=cpc&utm_campaign=Smart%20Shopping_Product_RF%2FIF%20and%20RFID&utm_term=&productid=12091811&gclid=Cj0KCQjwyZmEBhCpARIsALIzmnJFpvuEgjlyu7TWU4NleOs7qxlXTAPC6TweadRCwfA7SVi85MgYis0aAlyIEALw_wcB

- (2021). Retrieved 26 April 2021, from https://www.amazon.ca/WayinTop-Automatic-Irrigation-Watering-Capacitive/dp/B07TLRYGT1/ref=dp_prsubs_3?pd_rd_i=B07TLRYGT1&psc=1

- (2021). Retrieved 26 April 2021, from https://www.digikey.ca/en/products/detail/dfrobot/SEN0193/6588605?utm_adgroup=Evaluation%20Boards%20-%20Expansion%20Boards%2C%20Daughter%20Cards&utm_source=google&utm_medium=cpc&utm_campaign=Shopping_Product_Development%20Boards%2C%20Kits%2C%20Programmers_NEW&utm_term=&productid=6588605&gclid=Cj0KCQjwyZmEBhCpARIsALIzmnIn4E7SWCeh4trIi1wFG2y1JxV3N9Vic-8gl5DkILDaa0gniiqiAhgaAkk6EALw_wcB

- (2021). Retrieved 30 April 2021, from https://www.raspberrypi.org/products/raspberry-pi-4-model-b/specifications/

- The Internet of Things with ESP32. (2021). Retrieved 30 April 2021, from http://esp32.net/