UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name	:	MUHAMMAD ARIF BIN OSMAN
Date of Birth	:	
Title	:	DEVELOPMENT OF GRAPHICAL USER INTERFACE FOR SMART IRRIGATION SYSTEM
Academic Session	:	

I declare that this thesis is classified as:

☐	CONFIDENTIAL	(Contains confidential information under the Official Secret Act 1997)*
☐	RESTRICTED	(Contains restricted information as specified by the organization where research was done)*
☐	OPEN ACCESS	I agree that my thesis to be published as online open access (Full Text)

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

(Student's Signature)

New IC/Passport Number
Date: 18 FEBRUARY 2022

(Supervisor's Signature)

Ts. Dr. ROSHAHLIZA M RAMLI
Name of Supervisor
Date: 18 FEBRUARY 2022

NOTE : * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

THESIS DECLARATION LETTER (OPTIONAL)

Librarian,
Perpustakaan Universiti Malaysia Pahang,
Universiti Malaysia Pahang,
Lebuhraya Tun Razak,
26300, Gambang, Kuantan.

Dear Sir,

CLASSIFICATION OF THESIS AS RESTRICTED

Please be informed that the following thesis is classified as RESTRICTED for a period of three (3) years from the date of this letter. The reasons for this classification are as listed below.

Author's Name
Thesis Title

Reasons (i)

(ii)

(iii)

Thank you.

Yours faithfully,

(Supervisor's Signature)

Date: 18 FEBRUARY 2022

Stamp: TS. DR. ROSHAHLIZA BINTI M RAMLI
PENSYARAH KANAN
FAKULTI TEKNOLOGI KEJURUTERAAN ELEKTRIK DAN ELEKTRONIK
UNIVERSITI MALAYSIA PAHANG
26600 PEKAN, PAHANG

Note: This letter should be written by the supervisor, addressed to the Librarian, *Perpustakaan Universiti Malaysia Pahang* with its copy attached to the thesis.

SUPERVISOR'S DECLARATION

I/We* hereby declare that I/We* have checked this thesis/project* and in my/our* opinion, this thesis/project* is adequate in terms of scope and quality for the award of the degree of Bachelor Electronics Engineering Technology (Computer System) With Hons.

(Supervisor's Signature)

Full Name : Ts. Dr. ROSHAHLIZA M RAMLI

Position : SENIOR LECTURER

Date : 18 FEBRUARY 2022

(Co-supervisor's Signature)

Full Name :

Position :

Date :

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

(Student's Signature)

Full Name : MUHAMMAD ARIF BIN OSMAN

ID Number : TG18019

Date : 18 FEBRUARY 2022

DEVELOPMENT OF
GRAPHICAL USER INTERFACE
FOR SMART IRRIGATION SYSTEM

MUHAMMAD ARIF BIN OSMAN

Thesis submitted in fulfillment of the requirements
for the award of the degree of
Bachelor of Electronics Engineering Technology (Computer System) With Hons

Faculty of Electrical & Electronics Engineering
UNIVERSITI MALAYSIA PAHANG

FEBUARY 2022

ACKNOWLEDGEMENTS

I am grateful and would like to express my sincere gratitude to my supervisor Ts. Dr. Roshahliza Binti M Ramli for her guidance and support to achieve the objective of this research. I really appreciate guide me from the beginning until the end of this research. Without her assistance and guidance this research would be tougher to achieve. Thank you for spending time for correcting my mistakes and guide me.

I also would like to express very special thanks to Prof Dr Mortaza Mohammed for his suggestion and guidance especially in coding for the microcontroller. Without his guidance in coding of microcontroller this research would have a lot problem to achieve.

My sincere thanks go to all lecturers from Faculty of Electronic Engineering Technology, UMP, who give me knowledge and help me in my education that I will not forget. Also thanks to my friends for their support and giving hand during my study. All this year will be remembered as important memory.

I acknowledge my sincere indebtedness and gratitude to my family for their love and support and sacrifice throughout my life. Their sacrifice had inspired me from the day I have been born until what I have become now. I cannot find the appropriate words that could properly describe my appreciation for their devotion, support and faith in my ability to achieve my dreams.

Lastly, I would like to thanks to any person that contribute in my research. Their comments and suggestion is very helpful for the successful of this study.

ABSTRACT

The agriculture sector must undergo daily monitor that needs to run in precise time to ensure the good quality of the production. Nowadays, climate change had become a serious issue due to the changing led to a big impact on agriculture extension programs to the agriculture sector. Rising temperature and irregular weather patterns impact production and livestock populations. Irrigation imported be closely linked with water-use efficiency to boost productivity and improve food quality. To develop create a smart irrigation system used Wi-Fi and Radio Frequency to monitor and transmitted information about the system via a graphical user interface. Wi-Fi stands for wireless fidelity, a wireless local area network technology developed for use at home and in small regions. Wi-Fi was referred to as the technology surrounding the radio transmission of internet protocol data from an internet connection to a host computer, and it is the popular moniker for the wireless Ethernet 802.11b standard for WLANs (Owens & Karygiannis, 2015). Institute of Electrical and Electronics Engineers (IEEE) defined Wi-Fi as a class of certified wireless networking products that adhere to an industry standard known as "802.11b"(Al-Alawi & Ismail, 2006). In this system Wi-Fi,HC12(RF) and RTC Module DS3231 is used to control and monitor the irrigation system. This system provide easy to use for turn on and turn off solenoid valve. There are two types of on/off solenoid valve, first is regular turn on and off then it has timer on/off for 5,7 and 10 minutes.Auto irrigation also have been integrate to the system , system will turn on valve at 8am and 5pm for 10 minutes everyday. There are 4 soil moisture sensor are used in this prototype. All the sensor can be monitor through the IOT Dashboard. The data from the sensor will be send to the microcontroller then the data will be transfer to the IOT Dashboard and SD Card for offline data tracking. The IOT Dashboard can also analyze and view the collected data in the form of graphs, for better understanding. These data can improve the production.

TABLE OF CONTENT

TITLE PAGE	**i**
SUPERVISOR DECLARATION	**iv**
STUDENT DECLARATION	**v**
ACKNOWLEDGEMENTS	**vii**
ABSTRACT	**viii**
TABLE OF CONTENT	**ix**
LIST OF TABLES	**xi**
LIST OF FIGURES	**xii**
LIST OF ABBREVIATIONS	**xiv**
LIST OF APPENDICES	**xv**
CHAPTER 1 INTRODUCTION	
1.1 Project Background	**1**
1.2 Problem Statement	**2**
1.3 Research Objective	**3**
1.4 Research Scope	**3**
CHAPTER 2 LITERATURE REVIEW	
2.1 Introduction	**4**
2.2 Wi-Fi Technology and its Advantages	**5**
2.3 LPWAN Technology	**5**
2.4 LoRaWan as LPWAN Technology	**6**
2.5 Comparison of LoRa with other Technologies	**8**
2.5.1 LoRaWAN vs Zigbee	**8**
2.5.2 LoRaWAN vs SigFox	**9**

2.5.3 LoRaWAN vs NB-Iot 9

2.5.4 Channel Access Technique 9

2.5.5 Benefits of Smart Agriculture 10

2.5.6 Challenges in Smart Agriculture 10

2.6 LoRaWAN Application in Agriculture Machinery 11

2.7 Wireless Sensor Network in Agriculture 12

2.8 Network Connectivity Technologies 13

2.9 Application in Agriculture 15

CHAPTER 3 METHODOLOGY

3.1 Introduction 16

3.2 Local Controller Circuit 16

3.3 Local Controller Circuit Flowchart 17

3.4 Receiver Circuit 18

3.5 Receiver Circuit Flowchart 19

3.6 Local Controller Circuit IOT Dashboard 19

3.7 Local Controller Circuit IOT Dashboard Flowchart 20

3.8 On/Off Scheduling 21

3.9 On/Off Scheduling Flowchart 23

3.10 Sensor Monitoring System 24

3.11 Sensor Monitoring System Flowchart 25

3.12 Data Logging 26

3.13 Data Logging Flowchart 28

CHAPTER 4 RESULTS AND DISCUSSION

4.1 Introduction 29

4.2 Project Outcome **30**

4.3 Discussion **28**

CHAPTER 5 CONCLUSION AND RECOMMENDATION

5.1 Conclusion **33**

5.1 Recommendation **33**

REFERENCES **34**

APPENDICES **38**

LIST OF TABLES

Table 2.1 Wireless Network Technology **8**

LIST OF FIGURES

Figure 3.1	Local Controller Circuit	**16**
Figure 3.2	Transmitter Circuit	**17**
Figure 3.3	Transmitter Circuit Flowchart	**17**
Figure 3.4	Receiver Circuit	**18**
Figure 3.5	Receiver Circuit Flowchart	**18**
Figure 3.6	Local Controller Circuit IOT Dashboard	**19**
Figure 3.7	Local Controller Circuit IOT Dashboard Flowchart	**21**
Figure 3.8	IOT Dashboard Event Section	**21**
Figure 3.9	Part Of Coding For Offline Scheduling	**21**
Figure 3.10	On/Off Schedulling Flowchart	**23**
Figure 3.11	Sensor Circuit	**23**
Figure 3.12	Sensor Monitoring IOT Dashboard	**24**
Figure 3.13	Sensor Monitoring IOT Dashboard Flowchart	**26**
Figure 3.14	Data Logging IOT Dashboard	**26**
Figure 3.15	Sensor Readings In SD Card	**26**
Figure 3.16	Data Logging Flowchart	**27**
Figure 4.1	Full Setup System	**28**
Figure 4.2	Position Of Soil Moisture Sesnor	**28**
Figure 4.3	IOT Dashboard Live Reading	**29**
Figure 4.4	Soil Moisture Reading Graph	**29**
Figure 4.5	IOT Dashboard Valve On State	**30**
Figure 4.6	Installation Of All Component	**30**

LIST OF ABBREVIATIONS

IOT	Internet Of Things
LoRa	Long Range

LIST OF APPENDICES

Appendix A:	Coding	**27**
Appendix B:	Project Process	**94**

CHAPTER 1

INTRODUCTION

1.1 Project Background

The Internet of Things (IoT) is gaining traction in almost every sector, including banking, industry, consumer electronics, automotive, and many others. In today's world, everything is linked to everything else in some way. From our offices, we can monitor the lights and appliances in our homes. Activities in industries and other fields are monitored and observed from distance. Controlling street lights from distance, using a smart food order system, and so on are no longer ideas. Gone are the days when monitoring and control required the presence of an individual on the scene.

With the introduction of IoT, chip manufacturers have made substantial investments, and the industry is rapidly expanding, bringing a plethora of new technologies and solutions to market. There are several different types of networks that has been used in IoT technology, and they are classified according to their transmission and scale. One of widely used network is Wi-Fi. Wi-Fi is a wireless networking technology that connects devices to the Internet, including computers, mobile devices, and other equipment. It allows these devices, as well as many others, to communicate with one another, forming a network.

Implementation of IoT in agriculture can be seen in the smart irrigation system. Farmers typically work on big plots of land to cultivate a variety of crops. It is not always possible for one person to keep constant monitoring of all of the fields. A given patch of land may receive more water, resulting in waterlogging, or it may receive far less or no water at all, resulting in dry soil. In either situation, the crops may be harmed, and the farmer may lose money. The Smart Irrigation System is based on Internet of Things (IoT) technology which able to regulate the water pump in irrigating the farm area as well as monitor the moisture, temperature, and pressure of the soil. The Wi-Fi module and controller are used to control the smart irrigation system. It also has sensors built in to receive and transfer data to and from the cloud.

In agriculture, an example of IoT technology implementation is a greenhouse production environment controlling and monitoring device. In the agriculture manufacturing system, critical temperature, humidity, and soil signals are collected in real time and transmitted through wireless networks via an M2M (machine to machine) support platform. The art and science of cultivating soil, growing crops, and raising livestock is known as agriculture. It necessitates the processing and distribution of human-grade plant and animal products. Agriculture produces the bulk of the world's food and textiles. Agricultural products include cotton, wool, and leather. (JIchun Zhao & Jun-feng Zhang & Jian-Xin Guo, 2010).

1.2 Problem Statement

Risk in the agricultural sector is caused by a variety of factors, including natural, biological, climatic, and input and production prices, all of which have negative consequences (Agwe, Fissha, Nair, & Larson, 2009; AIT/UNEP, 2011; Jain & M, 2006).

Climate change is one of the most significant threats to agricultural extension programs, affecting farms and farmers in a variety of ways. Climate change effects, according to Mulder (2017), should not be overlooked in the pursuit of sustainable growth. Extension service providers, on the other hand, are not yet professionally trained and prepared to assist farmers in managing agricultural risks through immediate and low-cost solutions (Ali, Man, Abd Latif, Muharam, &Zobidah Omar, 2018).

Rising temperatures, irregular weather patterns, increased intensity of droughts, floods, and cyclones have all been evidence of climatic change, resulting in massive losses in agricultural production and livestock populations (Jain & M, 2006). These events will have a low quality of crop production as a sequence.

Agriculture is the world's greatest consumer of freshwater, and irrigation practices can be environmentally, economically, and socially unsustainable, squandering water, energy, and money, drying up rivers and lakes, lowering crop yields, damaging fish and animals, and polluting water (Aberra, 2004). More food, profits, livelihoods, and ecological advantages must be produced at lower social and environmental costs to minimize the amount of water consumed. Water management should therefore balance the need for water for agriculture and the need for a sustainable environment (Bin Abdullah, 2006). Consequently, irrigation must be closely linked with water-use efficiency to boost productivity and improve food quality, especially in those areas where problems of water shortages or collection and delivery are widespread (GarcíaTejero, Durán-Zuazo, Muriel-Fernández, & Rodríguez-Pleguezuelo, 2011).

The income to the agricultural labor class shows as labor expense to agricultural commodity producers. Labor costs are included in the broad category of operational costs, which also include fertilizer, pesticides, gasoline, fodder, seeds, and irrigation fees. All of these costs are likely to be impacted by better communication, which has an impact on prices, availability, and demand (Ghosh, 2002).

1.3 Research Objective

This project has objectives as follows:

1. To develop smart irrigation system using Wi-Fi and RF frequency.

2. To develop GUI for smart irrigation system.

3. To test the develop GUI for smart irrigation system.

4. To test the proposed methods in real environment.

1.4 Research Scope

LPWAN system for agriculture will be test at the pre-nursery at FTKEE. The test site will have the width of 0.7 hectares. Besides that, this LPWAN system for agriculture will also be test on 99,000 seed that have been planted on the site. There are also water tank with 2.4m of length and 7.8m of width will be connect to the agriculture machine. Frequency band will be use for LPWAN in range of 920-925Mhz. The distance of the office that will be use monitor all the sensors is about around 2KM so we will use gateway that can reach up to 2KM. For the local network to control the solenoid valve, we will install router. This router will connect to the microcontroller to able to control the solenoid valve. This router also needs to reach up to 300m range.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter discusses the differences between LPWAN and other technologies, as well as wireless sensor networks and connectivity technology in agriculture, in this chapter. Also, describe LPWAN technology and its use in agriculture.

Wireless sensor networks have recently been used in a variety of applications, including agriculture data collection, industrial control, logistics management, and meteorological monitoring. For wireless underground sensor networks, LoRa offers a new communication solution. There is a LoRa propagation testing node seen.

In agriculture, the deployment of IoT devices, wireless sensors, and sensor networks can be extremely beneficial in terms of monitoring the environment and growing crops, but having a network to support those devices is needed to make the most of those tools. LPWAN has recently been recognized as a suitable technology for agricultural use. LoRa is an LPWAN representative network. Because of its long range and low power capabilities, it can be used in IoT for agriculture. The majority of studies to date have focused on LoRa communication capabilities in urban, mountain, and maritime environments, with little attention paid to agriculture use cases (Daeunyin, 2018).

2.2 Wi-Fi Technology and its Advantages

Wi-Fi stands for wireless fidelity, a wireless local area network technology developed for use at home and in small regions. Wi-Fi was referred to as the technology surrounding the radio transmission of internet protocol data from an internet connection to a host computer, and it is the popular moniker for the wireless Ethernet 802.11b standard for WLANs (Owens & Karygiannis, 2015). Institute of Electrical and Electronics Engineers (IEEE) defined Wi-Fi as a class of certified wireless networking products that adhere to an industry standard known as "802.11b"(Al-Alawi & Ismail, 2006).

Wi-Fi's primary goal was to provide in-building broadband coverage. It was being designed to give coverage on university campuses, hotels, private residences, restaurants, and airports. The Wi-Fi network architecture is made up of several APs (Access Points) or one or more APs and one or more clients. One client has a direct connection to AP (Shailandra, 2012). The Wi-Fi network makes use of radio signals to connect to the internet or a mobile operator's network for connectivity or to access internet services. It only provides services up to the link layer, hence it was reliant on wired IP infrastructure for end-to-end connectivity (Henry & Luo, 2002).

Mobility and the absence of unsightly connections are two major advantages of wireless networks. Besides, setting up a Wi-Fi network is oftentimes the most cost-effective option to get the necessary connection with the surroundings. Because the cost of a single wireless adapter is dropping nearly daily, using Wi-Fi to create a vast network area is the most cost-effective option (Li, Hu, Chen, & He, 2011). Over and above that, it is easy to set up. Setting up a wireless network may appear to be a difficult operation, but it is rather simple. Wi-Fi networks don't need to be set up by a professional, and there are no holes to drill or wires to run through walls (Henry & Luo, 2002). Not to forget, Wi-Fi have fast data transfer rates. 802.11g is the fastest commercially available Wi-Fi technology, with transmission speeds of up to 54 megabits (Mb) per second (6.75 megabytes) (Shailandra, 2012).

2.3 LPWAN Technology

Wireless sensor networks (WSN) are evolving into low-power wide-area networks (LPWAN) for long-range Internet of Things (IoT) applications. In contrast to conventional broadband networks, which concentrate on high data rates and low latency, these networks focus on low cost, scalability, extended range, and energy efficiency for end user devices. Smart metering, precision irrigation, smart houses, and intelligent transportation systems are only a few of the big applications for LPWAN. The following four main performance metrics for LPWAN technologies are required to enable these applications (U.Raza, 2017).

LPWAN technology can be broadly classified into two categories, namely lincense LPWAN technology (for example, NB-IoT, LTE-M-IoT, EC-GSM-IoT, and 5G IoT) and LPWAN technologies that are not licensed (SigFox, Telensa, LoRa, and RPMA).

One of the LPWAN technology is Long Range (LoRa) is a wireless digital data transmission technology for the Internet of Things (IoT) that can transmit data over long distances (more than 10 km in rural areas) while consuming little electricity.

2.4 LoRaWAN as LPWAN Technology

Low Power Wide Area (LPWA) technology has received a lot of attention in recent years as it gains traction in the application of the Internet of Things (IoT) definition (Georgiou & Raza, 2017). Many research centres have addressed the issue of sensor networks, resulting in a concerted effort to improve WSN network efficiency. The Internet of Things (IoT) definition has the potential to transform and improve the way we conduct everyday business, allowing us to address current problems such as the energy crisis, resource scarcity, globalisation impacts, and pollution. The concept's potential is enormous, and it aims to address the M2M (Machine-toMachine) industry's major challenges (Raza et al., 2017).

Depending on the operating zone, most LPWAN (Low-Power Wireless Area Networks) run in the ISM (Industrial, Scientific, and Medical) unlicensed frequency band at 169 MHz, 433 MHz, 868/915 MHz, and 2.4 GHz. Communication protocols such as SigFox, LoRa, Weightless, and Ingenu are examples of this (Lavric & Petrariu, 2018). The LoRa Alliance is standardising the MAC communication protocol and system architecture for the WSN network known as LoRaWAN. LoRaWAN networking is a solution for the Internet of Things. The protocol employs a network topology known as a star-of-stars, in which every Lora node has direct access to the Gateway module (the sink node) (LoRa-Alliance, 2014).

One of the primary benefits of LoRaWAN is that it operates on the unlicensed ISM (Industrial, Scientific, and Medical) frequency, which eliminates the need for a license. The only restriction is imposed by the duty cycle parameter, which varies depending on the geographical region. Lora systems run on batteries (Lavric & Petrariu, 2018).

The LoRa Alliance divides devices into three categories: class A, class B, and class C. A LoRa node, in addition to duty cycle control, has the ability to accept packets from the Gateway module only during specific time slots in order to achieve power efficiency (LoRa-Alliance, 2014).

LoRa modulation employs the chirp spread spectrum (CSS) technique, which provides resistance to interference, a large communication link budget, and low-power efficiency while avoiding multi-path propagation fading and the Doppler effect. Additionally, the LoRa technology supports localization techniques (Catak & Celikkaya, 2019).

2.5 Comparison of LoRa with other Technologies

Every technology has its own characteristics, benefits, and drawbacks. IoT systems cannot be served by a single technology. Each technology differs from the others in a number of ways. In terms of specifications and usage, applications vary from one another. We can only select a technology from the current technologies that is ideally suited for the particular application based on the requirement. Wi-Fi is the most widely used technology for long-distance communication that has recently grown. For short distance communication, we have Bluetooth, ZigBee, and other technologies that can and are being used in a variety of IoT applications (ShilpaDevalal, 2018). LoRa, NB-IoT, Wi-Fi, Bluetooth, Zigbee, and sub-1GHz are some of the technical solutions used in IoT to transmit data over a network. However, in most situations, long transmission distance and energy efficiency cannot be assured at the same time in wireless communication network systems, since longer transmission distance necessitates more energy. Table 2.1 depicts the different types of wireless network technology. All wireless network technologies are distinguished by various characteristics such as connection range, velocity, energy consumption to operate the device, construction cost, communication cost, and network application.

Table 2.1 shows that comparison wireless technology. From that knows that LoRa is the one use for longest distance. LoRa is the best wireless network technology for our project because LoRa includes long distance communication. LoRa is use to communicate from the end node and until the application server in this project.

Table 2.1 Wireless Network Technology

Wireless Network Technology	Distance	Velocity	Construction Cost	Communication Cost	Energy Consumption	Application
Lora	Longest	Slowest	Low	Median	Free	Outdoor Sensors
3G/4G	Long	Fastest	High	High	Dataflow change	Call & Internet
Wifi	Short	Fast	Highest	Low	Free	Home network
Bluetooth	Shortest	Median	Low	Low	Free	Phone accessories
ZigBee	Shorter	Slow	Low	Low	Free	In-door Equipment
NB-IoT	Long	Low	Medium	Medium	Medium	Smart metering
Sigfox	Long	Low	Low	Low	Low	Electricity meters

2.5.1 LoRaWAN vs ZigBee

ZigBee is a 2.4 GHz wireless protocol based on IEEE 802.15.4. You can run a variety of application profile stacks on top of it, and since it's regulated by a standards body, it can work with multiple manufacturers. ZigBee is a personal area network protocol based on a high-level networking protocol. It is made up of low-power digital radios. These are ideal for small-scale projects that need data to be transferred over short distances. It has a length of ten to one hundred meters. These are mesh networks that send data over long distances by sending it to a number of intermediate devices. Since this consumes electricity, it is not appropriate for low-power applications.

LoRa, on the other hand, is built on a star topology that prevents the transfer of data to intermediate devices, resulting in a significant reduction in power consumption. Mesh networks are also only suitable for short to mediumrange communications, lacking the longrange capabilities of LoRa technology (KanchanKaushal, Taranvir Kaur and Jaspinder Kaur, 2014).

2.5.2 LoRaWAN vs SigFox

Sigfox, like LoRa, uses Differential Binary Phase Shift Keying (DBPSK) modulation in an Ultra Narrow Band (UNB) with a 100 Hz message width to connect with base stations. As a result of the low noise levels, Sigfox has a high receiver sensitivity, low power consumption, and a low-cost antenna design. The main disadvantage of this technology is that it uses UNB, which has a low data rate of only 100 bits per second, making it the slowest of the LP-WAN technologies(Kais Mekki, Eddy Bajic, Frédéric Chaxel and Fernand Meyer, 2019)

2.5.3 LoRaWAN vs NB-Iot

Unlike LTE-M, NB-IoT was designed specifically for low-power stationary sensors. It uses SC-FDMA in the uplink and OFDMA in the downlink, and operates in the licenced spectrum like LTE-M. LTE-M has a larger data rate (1 Mbps vs. 60 kbps) and bandwidth (1.4 MHz vs. 180 kHz) than NB-IoT and, as a result, has a shorter latency of up to 100 ms, allowing for real-time voice over LTE communication (VoLTE)(Y. Kabalcı and M. Ali, 2019).

The most widely used LP-WAN technology that operates on licenced frequencies is NBIoT. By 2023, it is expected that NB-IoT and LoRa would account for over 85% of all LP-WAN connections, hence these two technologies are frequently contrasted. LoRa uses less energy and has a longer battery life (Bharat Chaudhari, Marco Zennaro and Suresh Borkar, 2020).

2.5.4 Channel Access Technique

Different communication channels are used in a LoRaWAN network, which are configured and monitored by Gateway devices. The number of assigned channels is determined by geographic constraints or other wireless network configurations (AlexandruLavric, 2018). When a LoRa node sends a packet, it chooses one of the channels at random and transmits it, without performing carrier sense type verification or using a predetermined initialization time slot. This access method is a specific ALOHA style access, with the difference that the packet length is variable.

2.5.5 Benefits of Smart Agriculture

Agriculture can be benefits in so many different aspects. One of that is enables farmers to increase yields by using the least amount of resources possible, such as water, agricultural inputs, and seeds. Drones and robotics are used in smart agriculture to assist in a variety of ways. This facilitates data collection and aids in wireless tracking and control.

Approaching smart irrigation systems has become a top priority to provide farmers with a smart tool that will assist them in producing high-quality crops. In the agriculture sector, smart irrigation has several advantages. The first benefit of a smart irrigation system in agriculture would be monitoring. Data includes the amount of rainfall, leaf wetness, temperature, humidity, soil moisture, salinity, climate, pest movement, and human activities to help understand the patterns and processes of the crops. The acquisition of such a complete record allows for optimal decision-making to improve the quality of agricultural output, reduce risk, and generate more revenue (Nawandar & Satpute, 2019).

Furthermore, precise, and timely weather forecasting data, such as climate changes and rainfall, may boost productivity. Such information can assist farmers in the planning stage and save labor costs (Chen, Xu, Liu, Hu, & Wang, 2014). When compared to employees physically evaluating the field using trucks or walking, the usage of IoT in agriculture will help save time and money when inspecting massive areas. The ability to use IoT to determine when and where pesticides or insecticides should be applied will save costs and waste (Asplund & Nadjm-Tehrani, 2016).

2.5.6 Challenges in Smart Agriculture

Farmers must understand and learn how to use technology in order to use smart farming equipment. Adopting smart agriculture farming on a wide scale across countries is a major challenge. The internet must be available at all times in order for smart agriculture to work. Many developed countries' rural areas do not meet these criteria. In addition, the internet connection is slower.

2.6 LoRaWAN Application in Agriculture Machinery

The LoRaWAN network was created to link thousands of sensors, modules, and appliances over a wide network for IoT applications (Nolan et al., 2016). It's a network protocol that's mostly used in Smart City applications where large network coverage is needed, but it's also being used in almost every other social aspect where its properties are appropriate. The use of the LoRaWAN protocol is increasingly increasing, as is its integration into a variety of social and development processes (Cukurova, 2017).

Aside from the LoRa Alliance's key goal of implementing LoRaWAN in Smart City applications, this protocol has a lot of potential in food production and agricultural IoT applications. Its long-range data transmission capability allows it to cover large areas (fields) and set up sensor nodes to 15 kilometers away from the base station in line of sight (Davcev, Mitreski, Trajkovic, Nikolovski, & Koteli, 2018; LoRa-Alliance, 2014).

In contrast to ZigBee's mesh network topology, LoRaWAN's star of stars topology does not require additional modules to serve as routers, lowering the network's total cost and complexity. Aside from the long-range transmission and topology that is best suited for agricultural applications, LoRaWAN has a low power consumption, making it ideal for batterypowered applications. This is ideal for agricultural applications because it eliminates the need for external power in the fields; instead, the sensors will operate on battery power for years (Cukurova, 2017; LoRa-Alliance, 2014).

LoRaWAN may be used in a variety of agricultural and food processing applications. Sensor nodes that calculate environmental parameters such as air temperature, humidity, soil moisture, and so on can be managed using LoRaWAN. It can also be used to control a variety of actuators (e.g. automatic sprinkler valve for irrigation purposes or automatic poultry feeder). It can also be used in applications such as tracking cattle's location with GPS sensors or calculating their temperature in farms (Bellini & Arnaud, 2016).

2.7 Wireless Sensor Network in Agriculture

Wireless Sensor Network (WSN) is defined by (Matin, 2012) as a self-configured and infrastructure-less wireless network that has been used in monitoring physical or environmental conditions, such as temperature, sound, vibration, pressure, motion, or pollutants. WSN is also being used to cooperatively pass their data through the network to the main location or sink where the data can be commemorated and evaluated.

Wireless Sensor Networks are made up of sensors that are dispersed around an area to track and control it based on physical phenomena. Those small devices are called sensors (Vieira, Coelho, Da Silva, & Da Mata, 2003; Xijun, Limei, & Lizhong, 2009). Such operations can be carried out automatically using computing resources and suitable technology. We can make proper decisions for each zone in the farm using WSN in Precision agriculture. Unlike traditional networks, which assume that the user is a human agent, WSNs are focused on the physical world, particularly the data (Karl & Willig, 2005).

For modern precision agriculture tracking, wireless sensor network technologies are increasingly being used. The benefits of a wireless sensor network in agriculture include high accuracy, increased production efficiency while lowering costs, low power consumption, and distributed data collection (Sahota, Kumar, Kamal, & Huang, 2010). Agriculture faces numerous challenges, and humanity's survival is dependent on agriculture and water, so precision agriculture monitoring is essential, and the demand for environmental monitoring and remote control in agriculture is rapidly increasing.

There have been few studies on how WSN can be used in agriculture. Miranda used a closed-loop irrigation method with distributed soil water measurements to calculate irrigation amounts (Kim, Evans, & Iversen, 2008). For soil moisture data from data loggers to a central computer logging location, Shock used radio transmission (De Lima, E Silva, & Neto, 2010). Wall and King looked at prototypes for plug-and-play smart soil moisture sensors and sprinkler valve controllers, as well as distributed sensor network architectures for site-specific irrigation automation (Xijun et al., 2009). Hyun-Joong Kang simulates the output of a sensor node in a greenhouse environment with low power activity, as well as the impact of crop growth when an inter-node contact point is blocked (Vieira et al., 2003).

2.8 Network Connectivity Technologies

Sensors, actuators, cloud computing, and a variety of other innovations have advanced significantly over the last decade. By the introduction of a slew of new types of low-cost small wireless devices as the internet of things (IoT) takes shape (Bembe, AbuMahfouz, Masonta, & Ngqondi, 2019). IoT is accomplished by combining data-oriented applications with many device-oriented smart objects with identifiable addresses to provide intelligent services internetbased and related technologies (Gazis, 2017; Xu et al., 2018).

Unfortunately, all layers of the protocol stack face difficulties in the advancement of IoT systems. Because of the fragmented market, which includes a wide variety of goods and standards from various organizations, networking is the most difficult task. This takes into account the number of connected objects, their location, battery life, data efficiency, and protection (Kobo, Abu-Mahfouz, & Hancke, 2017; Louw, Niezen, Ramotsoela, & AbuMahfouz, 2016; Ntuli & Abu-Mahfouz, 2016). There is still no perfect network connectivity solution, which explains why there are so many different network connectivity technologies on the market.

The traditional Internet of Things infrastructure is focused on multi-hop short-range communication technologies, which were previously used for wireless personal area networks (WPAN) and wireless local area networks (WLAN) (WLAN). ZigBee, Bluetooth, and other short-range transmission technologies are examples (Gubbi, Buyya, Marusic, & Palaniswami, 2013; Miorandi, Sicari, De Pellegrini, & Chlamtac, 2012).

Due to their small coverage areas and network topologies known for their energy efficiency, WPAN and WLAN are constrained in their implementation. Beyond short-range connectivity, IoT systems have additional technical requirements. Long-range innovations are included in the requirements. This leads to the realization that the future landscape of IoT network access will be heterogeneous. Both short-range and long-range transmission technologies can be used in various geographical areas or the same one. Long-term technical demands include universal coverage and low energy consumption. Low prices, low data speeds, and many devices per access node (base station, gateway, or access point) are also included (Nolan, Guibene, Kelly, Europe, & Email, 2016).

2.9 Application in Agriculture

The Smart Irrigation system was referred to as an Internet of Things-based technology that could automate the irrigation process by evaluating soil moisture and weather conditions. Smart irrigation technologies helped in saving water outdoors (Elijah, Rahman, Orikumhi, Leow, & Hindia, 2018). In agriculture, the IoT was used to provide farmers with decision-making tools and automation technologies that seamlessly combine products, knowledge, and services for increased production, quality, and profit. In comparison to standard automatic system timers, which irrigated on a user-determined set schedule, smart irrigation controllers and sensors have been created to reduce outdoor water use by irrigating based on plant water needs (Zhao, Zhang, Feng, & Guo, 2010).

There are several important technical elements to consider while deploying an IoT device. The range of communication distance, data throughput, battery life, mobility, latency, security and resilience, and cost of gateway modems should all be examined when considering wireless connectivity (García, Parra, Jimenez, Lloret, & Lorenz, 2020; Zhao et al., 2010).

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter clearly defines the research methods used to conduct the study and explains how the necessary data and information to address the research objectives and questions was collected, presented and analysed.

3.2 Local Controller Circuit

Figure 3.1 below show the main circuit for controlling solenoid valve. This circuit will connect to the IOT Dashboard through Wi-Fi connection. This main circuit divide by two section which is transmitter and receive

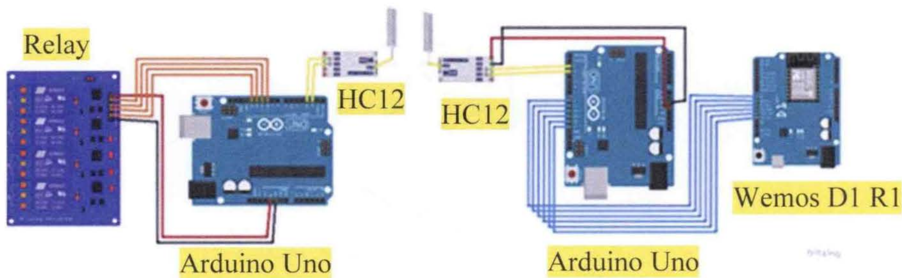


Figure 3.1 Local Controller Circuit

Figure 3.2 below show the transmitter circuit, this circuit using two microcontroller that connected using parallel communication to communicate each other using binary code. Arduino Wi-Fi the only microcontroller that connected to the IOT Dashboard will communicate with database. For Arduino Uno will communicate with receiver circuit using HC12 through the radio frequency.

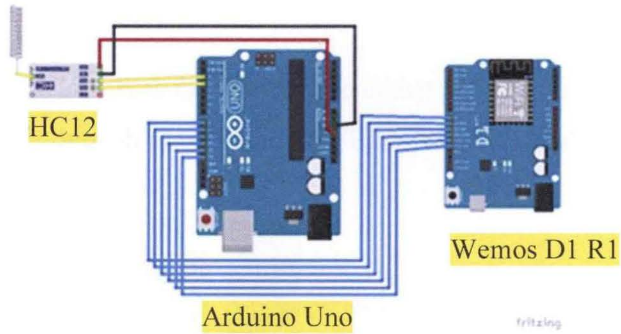


Figure 3.2 Transmitter Circuit

3.3 Local Controller Circuit Flowchart

Figure 3.3 below show the flowchart on how the local controller circuit works.

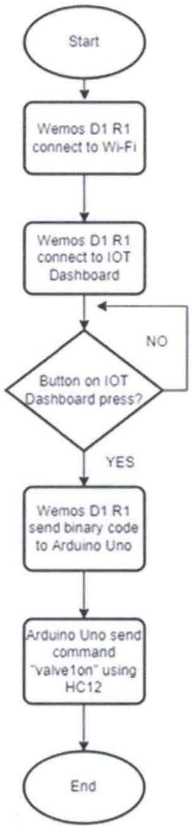


Figure 3.3 Transmitter Circuit Flowchart

3.4 Receiver Circuit

Figure 3.4 below show the receiver circuit that will communicate with transmitter circuit using HC12 through radio frequency. There are RTC modules and SD Card that connect to the receiver circuit.

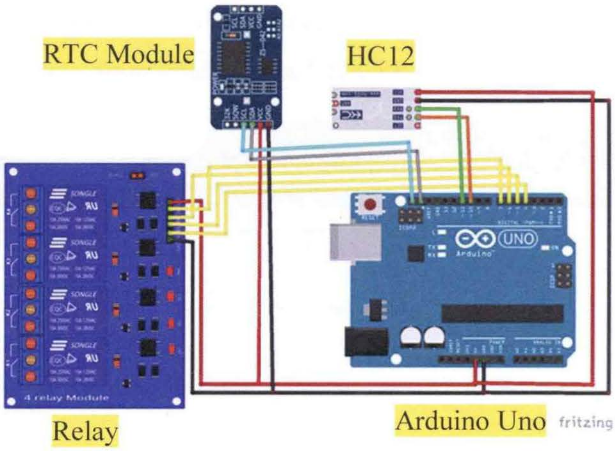


Figure 3.4 Receiver Circuit.

3.5 Receiver Circuit Flowchart

Figure 3.5 below show the flow on how the receiver circuit works after it receive the command from the transmitter circuit.

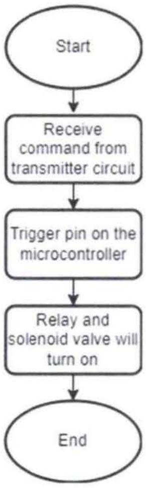


Figure 3.5 Receiver Circuit Flowchart

3.6 Local Controller Circuit IOT Dashboard

Figure 3.6 below show the IOT Dashboard that connected to the LC circuit. From the IOT Dashboard user will able to control ON/OFF relay and ON/OFF relay using timer 5/7/10 Minutes. Each relay has 5 widgets, From the left is regular ON/OFF button for relay. Next to it is ON/OFF relay for 5 minute, 7 minutes and 10 minute. Last widget is time remaining is where the countdown for ON/OFF relay using timer will show up.

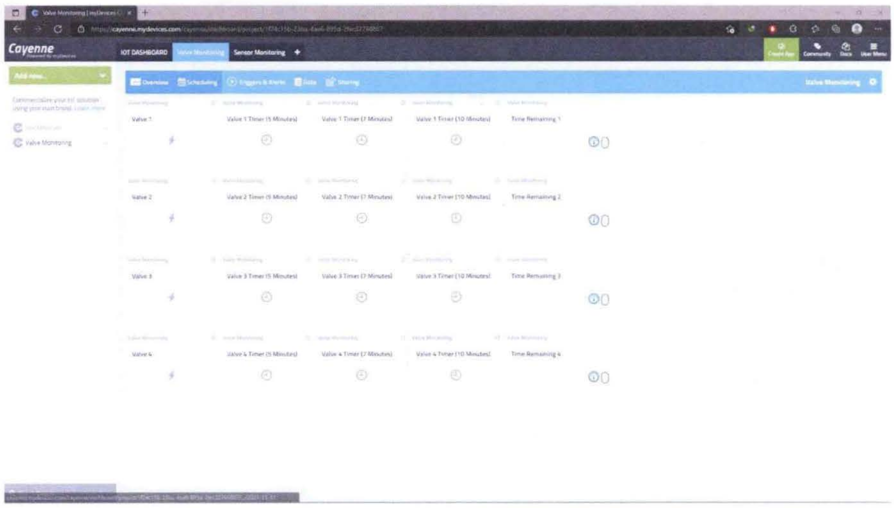


Figure 3.6 Local Controller Circuit IOT Dashboard

3.7 Local Controller Circuit IOT Dashboard Flowchart

Figure 3.7 below shows the system flow from IOT dashboard until the command send to the receiver circuit.

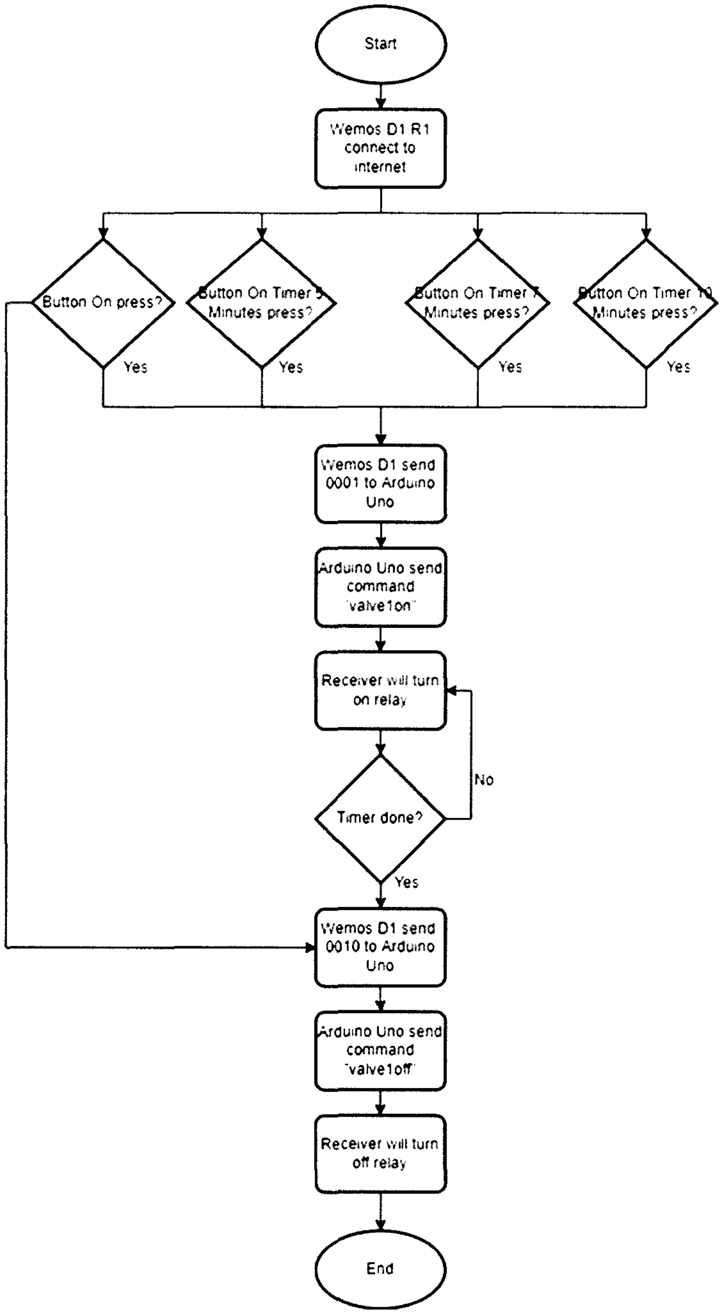


Figure 3.7 Local Controller Circuit IOT Dashboard Flowchart

3.8 On/Off Scheduling

There are two methods that we used to create the on/off scheduling. First method is set the scheduling using IOT Dashboard through online. Second method is set the scheduling by coding in the microcontroller based on the real time clock module.

IOT Dashboard is capable to set time when user want to On/Off relay on the calendar. This feature will benefit user to manage irrigation system wisely. This feature only can be executed when the microcontroller is connected to the internet connection. Figure 3.8 below show the scheduling system on the IOT Dashboard.

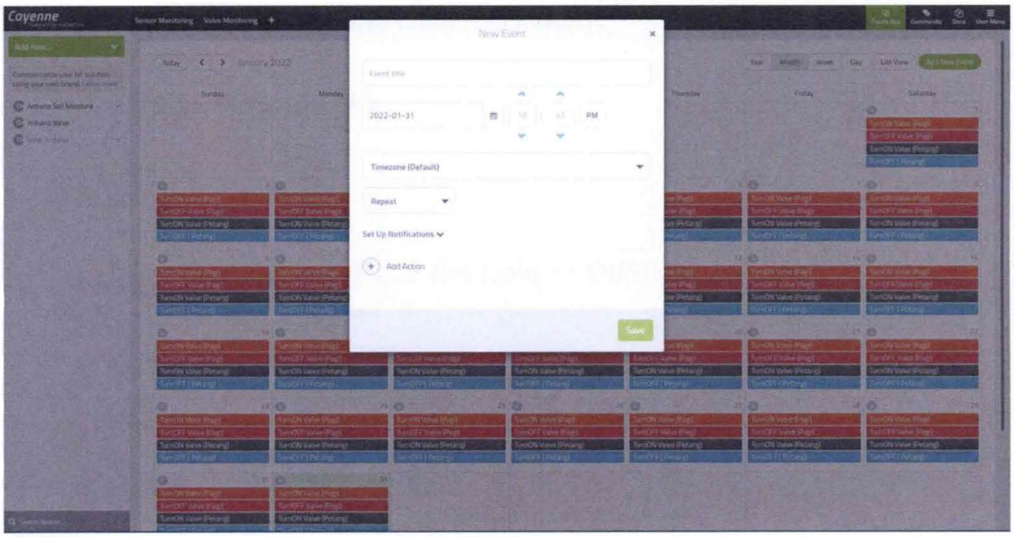


Figure 3.8 IOT Dashboard Event Section.

For offline method everything will set on the coding and it will fix until we change it in the coding. Figure 3.9 below shows the part of the coding to set the time for on/off scheduling.

```
//Variable to set time on/off

const int OnHour = 8; //SET TIME TO ON RELAY (24 HOUR FORMAT)

const int OnMin = 00;

const int OffHour = 8; //SET TIME TO OFF RELAY

const int OffMin = 10;
```

```
const int OnHour2 = 5; //SET TIME TO ON RELAY (24 HOUR FORMAT)

const int OnMin2 = 0;

const int OffHour2 = 05; //SET TIME TO OFF RELAY

const int OffMin2 = 10;

//command to turn on/off

if(t.hour == OnHour && t.min == OnMin){

  digitalWrite(4,LOW);

  }

  else if(t.hour == OffHour && t.min == OffMin){

    digitalWrite(4,HIGH);

  }

    if(t.hour == OnHour2 && t.min == OnMin2){

  digitalWrite(4,LOW);

  }



  else if(t.hour == OffHour2 && t.min == OffMin2){

    digitalWrite(4,HIGH);

  }
```

Figure 3.9 Part Of Coding For Offline Scheduling

3.9 On/Off Scheduling Flowchart

Figure 3.10 below show the process of on/off scheduling, microcontroller will check the time on RTC module if 8.00AM and 5.00PM valve will automatically turn on then after 10 minutes the valve will automatically turn off. For online method IOT Dashboard will give command to the microcontroller when to turn on based on current local time.

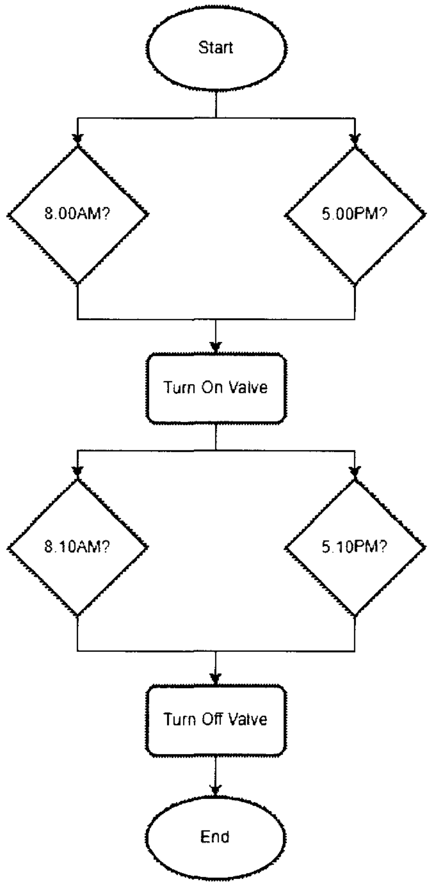


Figure 3.10 On/Off Scheduling Flowchart

3.10 Sensor Monitoring System

For sensor monitoring system we use four soil moisture sensor to check the moisture of the soil in real time. All the readings will be display on the IOT Dashboard through the Wi-Fi.

Figure 3.11 below shows the circuit for the sensor system. Sensor circuit will communicate with IOT Dashboard through Wi-Fi. Each reading from the sensor will be update every 2 second to the IOT Dashboard.

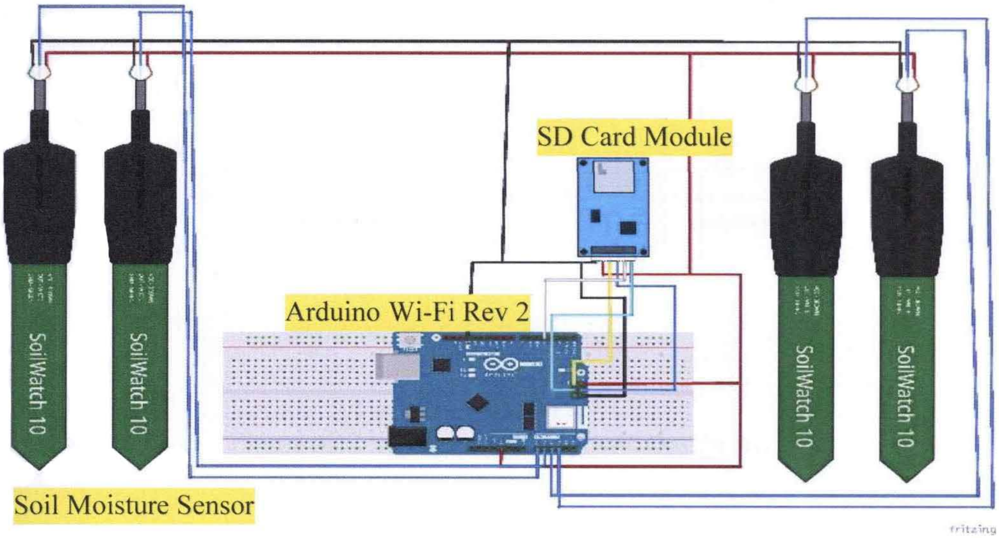


Figure 3.11 Sensor Circuit

Figure 3.12 below shows the sensor monitoring IOT Dashboard will display all the sensor reading that have been send through Wi-Fi from the microcontroller. Each sensor has two widgets to display the readings. For the first widget will display the readings in numerical value and for the second widget will display in graph system. Reading from the sensor will be update every two second to the IOT Dashboard.

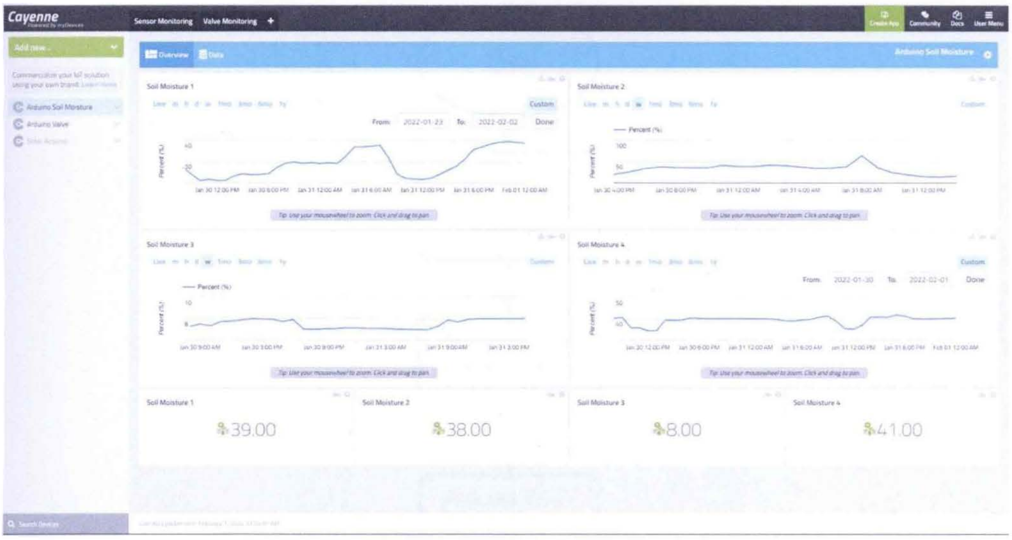


Figure 3.12 Sensor Monitoring IOT Dashboard

3.11 Sensor Monitoring System Flowchart

Figure 3.13 below show the flow on how the data from the sensor will be transfer to the Cayenne database and display on the IOT Dashboard.

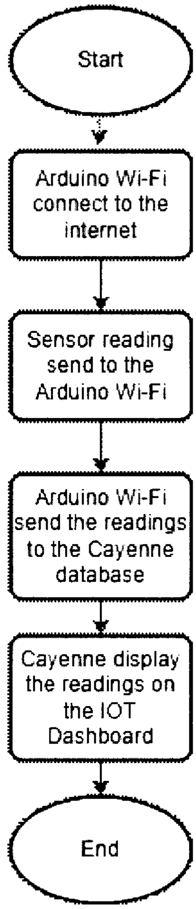


Figure 3.13 Sensor Monitoring Flowchart

3.12 Data Logging

Smart irrigation system has two methods for storing data that have collected from the sensor. The first method is store data in the Cayenne database. Second method is data will be store in the SD card using SD card module.

Figure 3.14 below shows the data logging system in the IOT Dashboard. All the sensor reading will be recorded and can be access through data logging system. This system can filter type of device and range of time on when user want to analyse the data. All the data can be download from the data logging and it will be in the excel format to give user advantage to sketch graph easier and analyse the data.

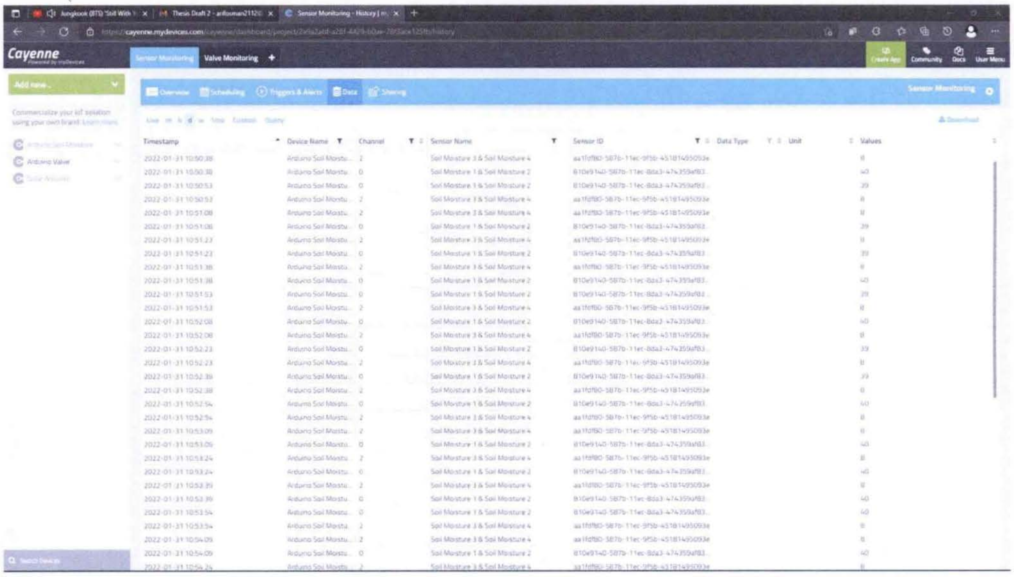
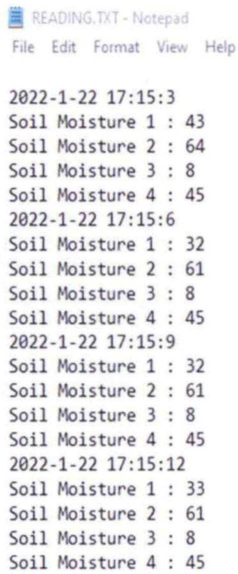


Figure 3.14 Data logging IOT dashboard

Figure 3.15 below show the example of the output data from the SD card for the offline method. All the data will be store every 3 second and the data will synchronize with the online data.


READING.TXT - Notepad
File Edit Format View Help

2022-1-22 17:15:3
Soil Moisture 1 : 43
Soil Moisture 2 : 64
Soil Moisture 3 : 8
Soil Moisture 4 : 45
2022-1-22 17:15:6
Soil Moisture 1 : 32
Soil Moisture 2 : 61
Soil Moisture 3 : 8
Soil Moisture 4 : 45
2022-1-22 17:15:9
Soil Moisture 1 : 32
Soil Moisture 2 : 61
Soil Moisture 3 : 8
Soil Moisture 4 : 45
2022-1-22 17:15:12
Soil Moisture 1 : 33
Soil Moisture 2 : 61
Soil Moisture 3 : 8
Soil Moisture 4 : 45

Figure 3.15 Sensor Readings In SD Card

3.13 Data Logging Flowchart

Figure 3.16 below show the flow on how the data from the sensor will be transfer to the Cayenne database and SD card.

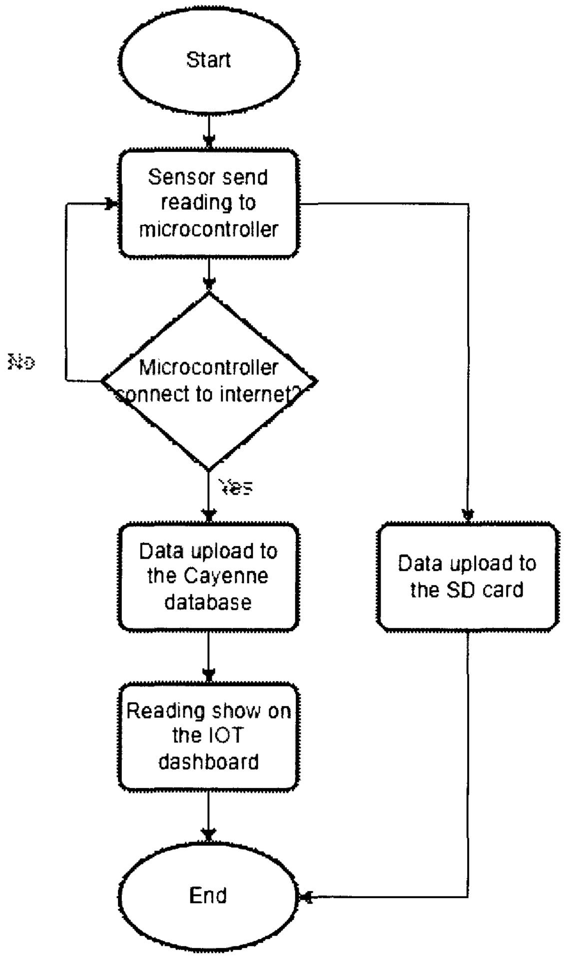


Figure 3.16 Data Logging Flowchart

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

In this chapter, we will discuss the results and detailed discussion of the final prototype in this project.

4.2 Project Outcome

Figure 4.1 below shows the final prototype that have been installed in the pre-nursery UMP. Figure 4.2 below shows on how the soil moisture sensor is placed.

Figure 4.1 Full Setup System Figure 4.2 Position Of Soil Moisture Sensor

4.3 Discussion

For this project we managed to collect data from the soil moisture sensor and test the prototype for two weeks. All the readings and result can be collect from the IOT dashboard and SD card. Figure 4.3 below show the live reading on the pre nursery that have been uploaded by microcontroller.

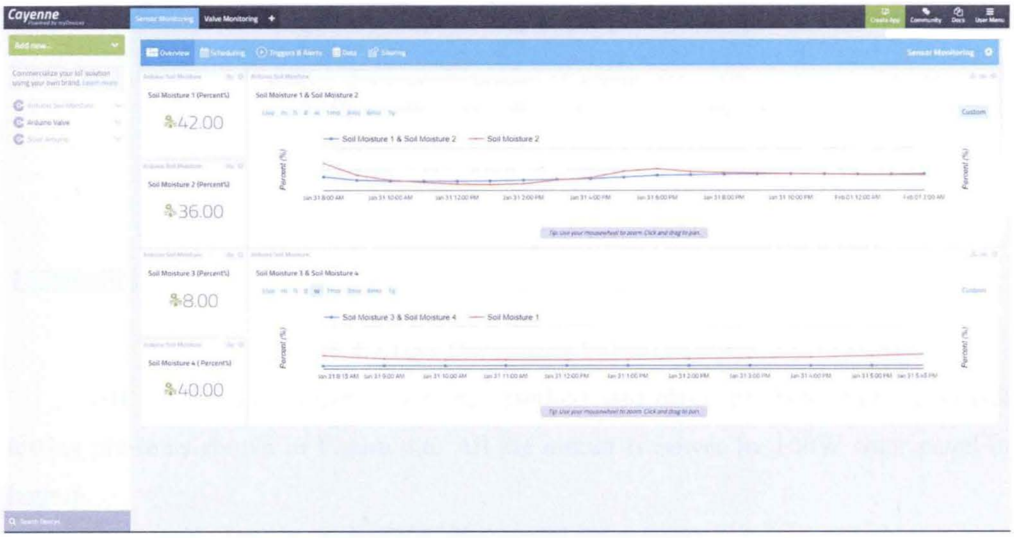


Figure 4.3 IOT Dashboard Live Reading

Figure 4.4 below show the average reading that have been collect. The reading of sensor will increase when time reached 8 am and 5 pm because the valve energized by coil to flow the water the soil.

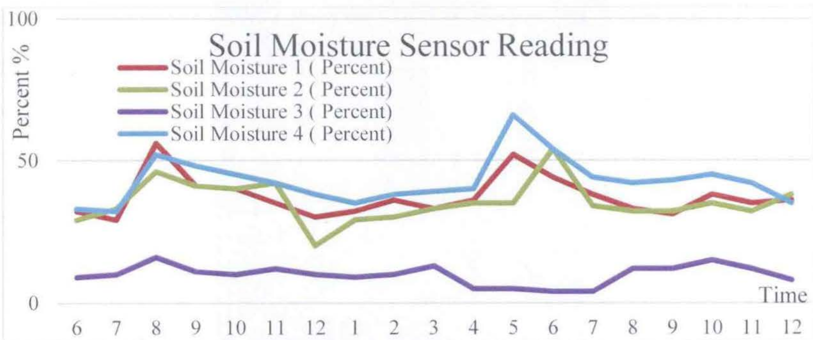


Figure 4.4 Soil Moisture Reading Graph

This final prototype give ability to user able to control solenoid valve by using web base application as long as transmitter circuit is connect to the internet. Figure 4.5 below show the example when user turn on solenoid valve.

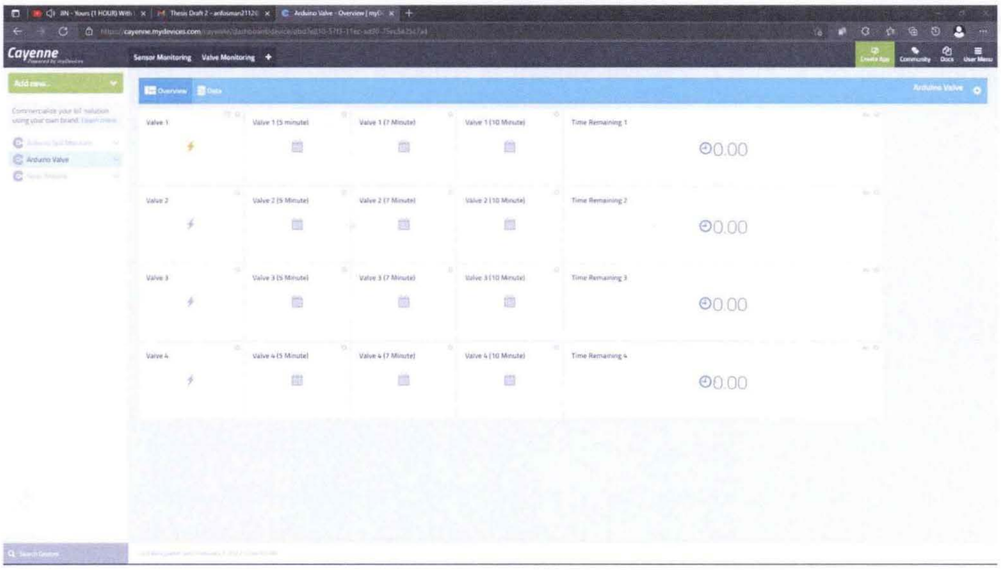


Figure 4.5 IOT Dashboard Valve On State

All the circuit components were installed into electrical box after the circuit testing phase as shown in Figure 4.6. All the circuit is power by 100W solar panel or battery.

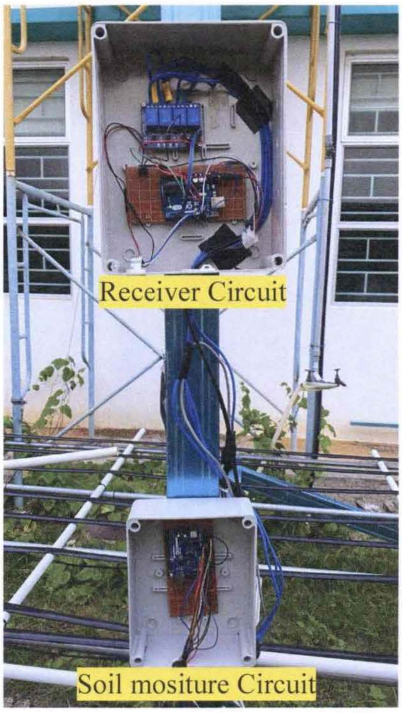


Figure 4.6 Installation Of All Component.

Connection for Arduino Wi-Fi and transmitter circuit is coded to connect to a personal Wi-Fi that have been installed at the pre nursery. There is no delay during performance of the sending and fetching data.

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

Our team managed to build smart irrigation system powered by 100W solar panel that can monitor moisture of soil by using soil moisture sensor, control on and off solenoid valve and build system for scheduling on and off solenoid valve. There are many try and error for component to get the most suitable components for the project. Arduino is used as main component to control and data transfer through Wi-Fi with IOT Dashboard.

5.2 Recommendation

There are lots of improvement can be done to this project, for example by adding more sensors such as PH sensor, temperature, humidity and etc. Integrate more smart function on the system such as trigger solenoid valve when soil moisture reading in certain condition. Lastly, recreate transmitter circuit became more simple and easy to troubleshoot and became more future proof.

REFERENCES

A. Berni and W. Gregg, "On the utility of chirp modulation for digital signaling", IEEE Transactions on Communications, vol. 21, no. 6, pp. 748-751, June 1973.

Al-Alawi, & Ismail, A. (2006). WiFi Technology : Future Market Challenges and Opportunities Adel Ismail Al-Alawi Department of Management Information Systems , College of Information Technology. Journal of Computer Science, 2(1), 13–18.

Asplund, M., & Nadjm-Tehrani, S. (2016). Attitudes and Perceptions of IoT Security in Critical Societal Services. IEEE Access, 4, 2130–2138. https://doi.org/10.1109/ACCESS.2016.2560919

Bharat Chaudhari, Marco Zennaro and Suresh Borkar, "LPWAN Technologies: Emerging Application Characteristics Requirements and Design Considerations", Future Internet, vol. 12, pp. 46, 2020

Bellini, B., & Arnaud, A. (2016). DIEstro Motion sensor platform for cattle oestrus detection.

Bembe, M., Abu-Mahfouz, A., Masonta, M., &Ngqondi, T. (2019). A survey on low-power wide area networks for IoT applications. Telecommunication Systems, 71(2), 249–274. https://doi.org/10.1007/s11235-019-00557-9

Catak, M., &Celikkaya, B. (2019). Limits of Number of Motes at Smart City Scenarios for LoRaWAN. Engineering, Technology & Applied Science Research, 9(3), 4116–4119. https://doi.org/10.48084/etasr.2693

Cukurova, C. C. (2017). Low Power Wide Area Network (LPWAN) and Internet of Things Adaptation in Agricultural Machinery. 4(January), 18–23. https://doi.org/10.21276/sjavs.2017.4.1.4

Chen, S., Xu, H., Liu, D., Hu, B., & Wang, H. (2014). A vision of IoT: Applications, challenges, and opportunities with China Perspective. IEEE Internet of Things Journal, 1(4), 349–359. https://doi.org/10.1109/JIOT.2014.2337336

Cukurova, C. C. (2017). Low Power Wide Area Network (LPWAN) and Internet of Things Adaptation in Agricultural Machinery. 4(January), 18–23. https://doi.org/10.21276/sjavs.2017.4.1.4

C. N. Cabaccan, F. R. G. Cruz, and I. C. Agulto 2017 "Wireless Sensor Network for Agricultural Environment Using Raspberry Pi Based Sensor Nodes," 2017ieee 9th Int. Conf. Humanoid, Nanotechnology, Inf. Technol. Commun. Control. Environ. Manag., Pp. 1–5

Daeunyin, Jiwon Chung, Yulim Cho, Hyunji Song, DaehanJin, Sojeong Kin, SungwookKo, Anthony Smith, "AN experimental LoRa performance evolution in Tree farm" IEEE Instrumentation and Measurement Society, 2018.

Davcev, D., Mitreski, K., Trajkovic, S., Nikolovski, V., &Koteli, N. (2018). IoT agriculture system based on LoRaWAN. IEEE International Workshop on Factory Communication Systems - Proceedings, WFCS, 2018-June, 1–4.https://doi.org/10.1109/WFCS.2018.8402368

De Lima, G. H. E. L., E Silva, L. C., &Neto, P. F. R. (2010). WSN as a tool for supporting agriculture in the precision irrigation. 6th International Conference on Networking and Services, ICNS 2010, Includes LMPCNA 2010; INTENSIVE 2010, 137–142. https://doi.org/10.1109/ICNS.2010.26

Elijah, O., Rahman, T. A., Orikumhi, I., Leow, C. Y., & Hindia, M. N. (2018). An Overview of Internet of Things (IoT) and Data Analytics in Agriculture: Benefits and Challenges. *IEEE Internet of Things Journal*, *5*(5), 3758–3773. https://doi.org/10.1109/JIOT.2018.2844296

García, L., Parra, L., Jimenez, J. M., Lloret, J., & Lorenz, P. (2020). IoT-based smart irrigation systems: An overview on the recent trends on sensors and iot systems for irrigation in precision agriculture. *Sensors (Switzerland)*, *20*(4). https://doi.org/10.3390/s20041042

Gazis, V. (2017). A Survey of Standards for Machine-to-Machine and the Internet of Things. IEEE Communications Surveys and Tutorials, 19(1), 482– 511. https://doi.org/10.1109/COMST.2016.2592948

Georgiou, O., & Raza, U. (2017). Low Power Wide Area Network Analysis: Can LoRa Scale? IEEE Wireless Communications Letters, 6(2), 162–165. https://doi.org/10.1109/LWC.2016.2647247

Gubbi, J., Buyya, R., Marusic, S., &Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. Future Generation Computer Systems, 29(7), 1645–1660. https://doi.org/10.1016/j.future.2013.01.010

Henry, P. S., & Luo, H. (2002). *WiFi : What ' s Next ?* (December), 66–72.

J. Petajajarvi, K. Mikhaylov, A. Roivainen, T. Hanninen and M. Pettissalo, "On the coverage of LPWANs: range evaluation and channel attenuation model for LoRa technology", 2015 14th International Conference on ITS Telecommunications (ITST), pp. 55-59, 2015.

J. J. Kang, H. Larkin and T. H. Luan, "Enhancement of Sensor Data Transmission by Inference and Efficient Data Processing" in Applications and Techniques in Information Security, Singapore:Springer, pp. 81-92, 2016

KanchanKaushal, Taranvir Kaur and Jaspinder Kaur, "ZigBee based Wireless Sensor Networks" in International Journal of Computer Science and Information Technologies, India:CTIEMTShahpur, 2014.

Karl, H., &Willig, A. (2005). PROTOCOLS AND ARCHITECTURES FOR WIRELESS SENSOR NETWORKS.

Kim, Y., Evans, R. G., &Iversen, W. M. (2008). Remote sensing and control of an irrigation system using a distributed wireless sensor network. IEEE Transactions on Instrumentation and Measurement, 57(7), 1379–1387. https://doi.org/10.1109/TIM.2008.917198

Kobo, H. I., Abu-Mahfouz, A. M., &Hancke, G. P. (2017). A Survey on Software-Defined Wireless Sensor Networks: Challenges and Design Requirements. IEEE Access, 5(c), 1872–58 1899. https://doi.org/10.1109/ACCESS.2017.2666200

Lavric, A., &Petrariu, A. I. (2018). LoRaWAN communication protocol: The new era of IoT. 2018 14th International Conference on Development and Application Systems, DAS 2018 - Proceedings, 74–77. https://doi.org/10.1109/DAAS.2018.8396074

Li, L., Hu, X., Chen, K., & He, K. (2011). The applications of WiFi-based Wireless Sensor Network in Internet of Things and Smart Grid. *Proceedings of the 2011 6th IEEE Conference on Industrial Electronics and Applications, ICIEA 2011*, 789–793. https://doi.org/10.1109/ICIEA.2011.5975693

LoRa-Alliance. (2014). What is LoRaWAN® Specification - LoRa Alliance®. Retrieved May 4, 2021, from https://lora-alliance.org/about-lorawan/

Louw, J., Niezen, G., Ramotsoela, T. D., & Abu-Mahfouz, A. M. (2016). A key distribution scheme using elliptic curve cryptography in wireless sensor networks. IEEE International Conference on Industrial Informatics (INDIN), 0, 1166– 1170. https://doi.org/10.1109/INDIN.2016.7819342

Matin, M. A. (2012). Wireless Sensor Networks - Technology and Protocols. In Wireless Sensor Networks - Technology and Protocols. https://doi.org/10.5772/2604

Miorandi, D., Sicari, S., De Pellegrini, F., &Chlamtac, I. (2012). Internet of things: Vision, applications and research challenges. Ad Hoc Networks, 10(7), 1497–1516. https://doi.org/10.1016/j.adhoc.2012.02.016

Nawandar, N. K., & Satpute, V. R. (2019). IoT based low cost and intelligent module for smart irrigation system. *Computers and Electronics in Agriculture*, *162*(May), 979–990. https://doi.org/10.1016/j.compag.2019.05.027

N. Rajput, N. Gandhi and L. Saxena, "Wireless Sensor Networks: Apple Farming in Northern India", 2012 Fourth International Conference on Computational Intelligence and Communication Networks, pp. 218-221, 2012

Nolan, K. E., Guibene, W., Kelly, M. Y., Europe, I. L., & Email, I. (2016). An Evaluation Of Low Power Wide Area Network Technologies For The Internet Of Things. 439–444.

Ntuli, N., & Abu-Mahfouz, A. (2016). A Simple Security Architecture for Smart Water Management System. Procedia Computer Science, 83, 1164–1169. https://doi.org/10.1016/j.procs.2016.04.239

Owens, L., & Karygiannis, T. (2015). *Wireless Network Security. 1.*

RashmiSharan Sinha, Yiqiao Wei and Seung-Hoon Hwang, "A survey on LPWA technology: LoRa and NB-IoT", Dongguk University-Seoul Republic of Korea, March 2017, [online] Available: www.sciencedirect.com.

Raza, U., Kulkarni, P., &Sooriyabandara, M. (2017). Low Power Wide Area Networks: An Overview. IEEE Communications Surveys and Tutorials, 19(2), 855–873. https://doi.org/10.1109/COMST.2017.2652320

Sahota, H., Kumar, R., Kamal, A., & Huang, J. (2010). An energy-efficient wireless sensor network for precision agriculture. Proceedings - IEEE Symposium on Computers and Communications, 3(1), 347–350. https://doi.org/10.1109/ISCC.2010.5546508

ShilpaDevalal, and A. Karthikeyan, "LoRatechnologyan overview" ICECA 2018.

ShilpaDevalal, and A. Karthikeyan, "LoRatechnologyan overview" ICECA 2018.

Shailandra, K. (2012). An overview of Technical aspect for WiFi Networks Technology. *International Journal of Electronics and Computer Science Engineering*, *1*, 28–34.

U. Raza et al., "Low Power Wide Area Networks: An Overview", IEEE Commun. Surveys & Tutorials, vol. 19, no. 2, pp. 855-873, 2017

Vieira, M. A. M., Coelho, C. N., Da Silva, D. C., & Da Mata, J. M. (2003). Survey on wireless sensor network devices. IEEE International Conference on Emerging Technologies and Factory Automation, ETFA, 1(January), 537–544. https://doi.org/10.1109/ETFA.2003.1247753

Xijun, Y., Limei, L., &Lizhong, X. (2009). The application of wireless sensor network in the irrigation area automatic system. Proceedings - International Conference on Networks Security, Wireless Communications and Trusted Computing, NSWCTC 2009, 1, 21–24. https://doi.org/10.1109/NSWCTC.2009.118

Xu, J., Yao, J., Wang, L., Ming, Z., Wu, K., & Chen, L. (2018). Narrowband internet of things: Evolutions, technologies, and open issues. IEEE Internet of Things Journal, 5(3), 1449–1462. https://doi.org/10.1109/JIOT.2017.2783374

Y. Kabalcı and M. Ali, "Emerging LPWAN Technologies for Smart Environments: An Outlook", 2019 1st Global Power Energy and Communication Conference (GPECOM), pp. 24-29, 2019.

Yoo, S. E., Kim, J. E., Kim, T., Ahn, S., Sung, J., & Kim, D. (2007). A2S: Automated agriculture system based on WSN. *Proceedings of the International Symposium on Consumer Electronics, ISCE*, 0–4. https://doi.org/10.1109/ISCE.2007.4382216

Zhao, J. C., Zhang, J. F., Feng, Y., & Guo, J. X. (2010). The study and application of the IOT technology in agriculture. *Proceedings - 2010 3rd IEEE International Conference on Computer Science and Information Technology, ICCSIT 2010*, *2*, 462–465. https://doi.org/10.1109/ICCSIT.2010.5565120

APPENDICES

Appendix A: Coding

Wemos D1 R1 (Transmitter Circuit)

```
//#define CAYENNE_DEBUG

#define CAYENNE_PRINT Serial

#include <CayenneMQTTESP8266.h>




// WiFi network info.

char ssid[] = "Ayep";

char wifiPassword[] = "Huaweitestok";


// Cayenne authentication info. This should be obtained from the Cayenne
Dashboard.

char username[] = "5874db20-0ba7-11ec-8da3-474359af83d7";

char password[] = "ebef037c3d62799163bc59dbb390388904a37638";

char clientID[] = "8ad1d850-4e93-11ec-8da3-474359af83d7";


int D, C, B, A;
```

```
// variable for valve 1

byte FiveMinTimer1;

byte SevenMinTimer1;

byte TenMinTimer1;

unsigned long expireTime1;

unsigned long time1;

unsigned long previousmillis1;



// variable for valve 2

byte FiveMinTimer2;

byte SevenMinTimer2;

byte TenMinTimer2;

unsigned long expireTime2;

unsigned long time2;

unsigned long previousmillis2;



// variable for valve 3

byte FiveMinTimer3;
```

```
byte SevenMinTimer3;

byte TenMinTimer3;

unsigned long expireTime3;

unsigned long time3;

unsigned long previousmillis3;



// variable for valve 4

byte FiveMinTimer4;

byte SevenMinTimer4;

byte TenMinTimer4;

unsigned long expireTime4;

unsigned long time4;

unsigned long previousmillis4;



void setup() {

  pinMode(D12, OUTPUT);

  pinMode(D11, OUTPUT);

  pinMode(D10, OUTPUT);
```

```
  pinMode(D9, OUTPUT);

  pinMode(D8, OUTPUT);




  Serial.begin(9600);

  digitalWrite(D12, LOW);



  Cayenne.begin(username, password, clientID, ssid, wifiPassword);



}



void loop() {

  Cayenne.loop();



  time1 = millis();

  previousmillis1 = millis();



  time2 = millis();
```

```
previousmillis2 = millis();



time3 = millis();

previousmillis3 = millis();



time4 = millis();

previousmillis4 = millis();



if (FiveMinTimer1 == 1 || SevenMinTimer1 == 1 || TenMinTimer1 == 1)

{

 if (expireTime1 < previousmillis1)

 {

  timerDone1();

 }

}



if (FiveMinTimer2 == 1 || SevenMinTimer2 == 1 || TenMinTimer2 == 1)

{
```

```
  if (expireTime2 < previousmillis2)
  {
   timerDone2();
  }
}

if (FiveMinTimer3 == 1 || SevenMinTimer3 == 1 || TenMinTimer3 == 1)
{
  if (expireTime3 < previousmillis3)
  {
   timerDone3();
  }
}

if (FiveMinTimer4 == 1 || SevenMinTimer4 == 1 || TenMinTimer4 == 1)
{
  if (expireTime4 < previousmillis4)
  {
```

```
    timerDone4();

   }

  }

  delay(200);

}



void timerDone1()

{

  digitalWrite(D11, LOW);

  digitalWrite(D10, LOW);

  digitalWrite(D9, HIGH);

  digitalWrite(D8, LOW);

  //set data ready

  digitalWrite(D12, HIGH);

  delay(100);

  digitalWrite(D12, LOW);

  FiveMinTimer1 = 0;

  SevenMinTimer1 = 0;
```

```
  TenMinTimer1 = 0;

  expireTime1 = 0;

  previousmillis1 = 0;

}



void timerDone2()

{

  digitalWrite(D11, LOW);

  digitalWrite(D10, HIGH);

  digitalWrite(D9, LOW);

  digitalWrite(D8, LOW);

  //set data ready

  digitalWrite(D12, HIGH);

  delay(100);

  digitalWrite(D12, LOW);

  FiveMinTimer2 = 0;

  SevenMinTimer2 = 0;

  TenMinTimer2 = 0;
```

```
 expireTime2 = 0;

 previousmillis2 = 0;

}



void timerDone3()

{

 digitalWrite(D11, LOW);

 digitalWrite(D10, HIGH);

 digitalWrite(D9, HIGH);

 digitalWrite(D8, LOW);

 //set data ready

 digitalWrite(D12, HIGH);

 delay(100);

 digitalWrite(D12, LOW);

 FiveMinTimer3 = 0;

 SevenMinTimer3 = 0;

 TenMinTimer3 = 0;

 expireTime3 = 0;
```

```
 previousmillis3 = 0;

}



void timerDone4()

{

 digitalWrite(D11, HIGH);

 digitalWrite(D10, LOW);

 digitalWrite(D9, LOW);

 digitalWrite(D8, LOW);

 //set data ready

 digitalWrite(D12, HIGH);

 delay(100);

 digitalWrite(D12, LOW);

 FiveMinTimer4 = 0;

 SevenMinTimer4 = 0;

 TenMinTimer4 = 0;

 expireTime4 = 0;

 previousmillis4 = 0;
```

```
}



void OnValve1()

{

  digitalWrite(D11, LOW);

  digitalWrite(D10, LOW);

  digitalWrite(D9, LOW);

  digitalWrite(D8, HIGH);

  //set data ready

  digitalWrite(D12, HIGH);

  delay(100);//beri masa kpd Receive untuk baca

  digitalWrite(D12, LOW);

}

void OffValve1()

{

  digitalWrite(D11, LOW);

  digitalWrite(D10, LOW);

  digitalWrite(D9, HIGH);
```

```
  digitalWrite(D8, LOW);

  //set data ready

  digitalWrite(D12, HIGH);

  delay(100);

  digitalWrite(D12, LOW);

}



void OnValve2()

{

  digitalWrite(D11, LOW);

  digitalWrite(D10, LOW);

  digitalWrite(D9, HIGH);

  digitalWrite(D8, HIGH);

  //set data ready

  digitalWrite(D12, HIGH);

  delay(100);//beri masa kpd Receive untuk baca

  digitalWrite(D12, LOW);

}
```

```
void OffValve2()
{
 digitalWrite(D11, LOW);
 digitalWrite(D10, HIGH);
 digitalWrite(D9, LOW);
 digitalWrite(D8, LOW);
 //set data ready
 digitalWrite(D12, HIGH);
 delay(100);
 digitalWrite(D12, LOW);
}

void OnValve3()
{
 digitalWrite(D11, LOW);
 digitalWrite(D10, HIGH);
 digitalWrite(D9, LOW);
```

```
  digitalWrite(D8, HIGH);

  //set data ready

  digitalWrite(D12, HIGH);

  delay(100);//beri masa kpd Receive untuk baca

  digitalWrite(D12, LOW);

}



void OffValve3()

{

  digitalWrite(D11, LOW);

  digitalWrite(D10, HIGH);

  digitalWrite(D9, HIGH);

  digitalWrite(D8, LOW);

  //set data ready

  digitalWrite(D12, HIGH);

  delay(100);

  digitalWrite(D12, LOW);

}
```

```
void OnValve4()
{
 digitalWrite(D11, LOW);
 digitalWrite(D10, HIGH);
 digitalWrite(D9, HIGH);
 digitalWrite(D8, HIGH);
 //set data ready
 digitalWrite(D12, HIGH);
 delay(100);//beri masa kpd Receive untuk baca
 digitalWrite(D12, LOW);
}


void OffValve4()
{
 digitalWrite(D11, HIGH);
 digitalWrite(D10, LOW);
 digitalWrite(D9, LOW);
```

```
  digitalWrite(D8, LOW);

  //set data ready

  digitalWrite(D12, HIGH);

  delay(100);

  digitalWrite(D12, LOW);

}



// Coding valve 1 start

CAYENNE_IN(V2) //send 1 for ON 2 for OFF

{

  int currentValue = getValue.asInt();

  if (currentValue == 1)

  {

    OnValve1();

  }

  else

  {

    OffValve1();
```

```
  }

}



CAYENNE_IN(V3) // 5 minutes valve 1

{

 FiveMinTimer1 = getValue.asInt();



 if (FiveMinTimer1 == 1)

 {

  expireTime1 = (millis() + 300000L);

  Cayenne.virtualWrite(V2, 1);

  OnValve1();

 }



 else

 {

  expireTime1 = 0;

 }
```

```
}

CAYENNE_IN(V4) // 7 minutes valve 1
{
 SevenMinTimer1 = getValue.asInt();

 if (SevenMinTimer1 == 1)
 {
  expireTime1 = (millis() + 420000L);
  Cayenne.virtualWrite(V2, 1);
  OnValve1();
 }

 else
 {
  expireTime1 = 0;
 }
```

```
}



CAYENNE_IN(V5) // 10 minutes valve 1

{

 TenMinTimer1 = getValue.asInt();



 if (TenMinTimer1 == 1)

 {

  expireTime1 = (millis() + 600000L);

  Cayenne.virtualWrite(V2, 1);

  OnValve1();

 }



 else

 {

  expireTime1 = 0;

 }

}
```

```
CAYENNE_OUT(V10) // Time remaining valve 1
{
 unsigned long TimeRemain1 = ((expireTime1 - time1) / 1000) / 60;
 if (FiveMinTimer1 == 1 || SevenMinTimer1 == 1 || TenMinTimer1 == 1)
 {
  Cayenne.virtualWrite(V10, TimeRemain1);

  if (TimeRemain1 == 0)
  {
   Cayenne.virtualWrite(V2, 0);
   Cayenne.virtualWrite(V3, 0);
   Cayenne.virtualWrite(V4, 0);
   Cayenne.virtualWrite(V5, 0);
  }
 }
 else
 {
```

```
   Cayenne.virtualWrite(V10, 0);
  }
}

// Coding valve 1 end

// Coding valve 2 start

CAYENNE_IN(V6) //send 3 for ON 4 for OFF
{
  int currentValue = getValue.asInt();
  if (currentValue == 1)
  {
   OnValve2();
  }
  else
  {
```

```
   OffValve2();
  }
}


CAYENNE_IN(V7) // 5 minutes valve 2
{
 FiveMinTimer2 = getValue.asInt();


 if (FiveMinTimer2 == 1)
 {
   expireTime2 = (millis() + 300000L);
   Cayenne.virtualWrite(V6, 1);
   OnValve2();
 }


 else
 {
   expireTime2 = 0;
```

```
 }
}

CAYENNE_IN(V8) // 7 minutes valve 2
{
 SevenMinTimer2 = getValue.asInt();

 if (SevenMinTimer2 == 1)
 {
  expireTime2 = (millis() + 420000L);
  Cayenne.virtualWrite(V6, 1);
  OnValve2();
 }

 else
 {
  expireTime2 = 0;
 }
```

```
}

CAYENNE_IN(V9) // 10 minutes valve 2
{
 TenMinTimer2 = getValue.asInt();

 if (TenMinTimer2 == 1)
 {
  expireTime2 = (millis() + 600000L);
  Cayenne.virtualWrite(V6, 1);
  OnValve2();
 }

 else
 {
  expireTime2 = 0;
 }
}
```

```
CAYENNE_OUT(V20) // Time remaining valve 2
{
 unsigned long TimeRemain2 = ((expireTime2 - time2) / 1000) / 60;
 if (FiveMinTimer2 == 1 || SevenMinTimer2 == 1 || TenMinTimer2 == 1)
 {
  Cayenne.virtualWrite(V20, TimeRemain2);

  if (TimeRemain2 == 0)
  {
   Cayenne.virtualWrite(V6, 0);
   Cayenne.virtualWrite(V7, 0);
   Cayenne.virtualWrite(V8, 0);
   Cayenne.virtualWrite(V9, 0);
  }
 }
 else
 {
```

```
  Cayenne.virtualWrite(V20, 0);
 }

}

// Coding valve 2 end

// Coding valve 3 start

CAYENNE_IN(V11) //send 5 for ON 6 for OFF
{
 int currentValue = getValue.asInt();
 if (currentValue == 1)
 {
  OnValve3();
 }
 else
 {
```

```
   OffValve3();
  }
}


CAYENNE_IN(V12) // 5 minutes valve 3
{
  FiveMinTimer3 = getValue.asInt();


  if (FiveMinTimer3 == 1)
  {
    expireTime3 = (millis() + 300000L);
    Cayenne.virtualWrite(V11, 1);
    OnValve3();
  }


  else
  {
    expireTime3 = 0;
```

```
  }

}



CAYENNE_IN(V13) // 7 minutes valve 3

{

  SevenMinTimer3 = getValue.asInt();



  if (SevenMinTimer3 == 1)

  {

    expireTime3 = (millis() + 420000L);

    Cayenne.virtualWrite(V11, 1);

    OnValve3();

  }



  else

  {

    expireTime3 = 0;

  }
```

```
}

CAYENNE_IN(V14) // 10 minutes valve 3
{
 TenMinTimer3 = getValue.asInt();

 if (TenMinTimer3 == 1)
 {
  expireTime3 = (millis() + 600000L);
  Cayenne.virtualWrite(V11, 1);
  OnValve3();
 }

 else
 {
  expireTime3 = 0;
 }
}
```

```
CAYENNE_OUT(V30) // Time remaining valve 3
{
  unsigned long TimeRemain3 = ((expireTime3 - time3) / 1000) / 60;
  if (FiveMinTimer3 == 1 || SevenMinTimer3 == 1 || TenMinTimer3 == 1)
  {
    Cayenne.virtualWrite(V30, TimeRemain3);


    if (TimeRemain3 == 0)
    {
      Cayenne.virtualWrite(V11, 0);
      Cayenne.virtualWrite(V12, 0);
      Cayenne.virtualWrite(V13, 0);
      Cayenne.virtualWrite(V14, 0);
    }
  }
  else
  {
```

```
   Cayenne.virtualWrite(V30, 0);
  }
}

// Coding valve 3 end

// Coding valve 4 start

CAYENNE_IN(V15) //send 7 for ON 8 for OFF
{
  int currentValue = getValue.asInt();
  if (currentValue == 1)
  {
    OnValve4();
  }
  else
  {
```

```
   OffValve4();
  }
}


CAYENNE_IN(V16) // 5 minutes valve 4
{
  FiveMinTimer4 = getValue.asInt();


  if (FiveMinTimer4 == 1)
  {
    expireTime4 = (millis() + 300000L);
    Cayenne.virtualWrite(V15, 1);
    OnValve4();
  }


  else
  {
    expireTime4 = 0;
```

```
  }
}

CAYENNE_IN(V17) // 7 minutes valve 4
{
  SevenMinTimer4 = getValue.asInt();

  if (SevenMinTimer4 == 1)
  {
    expireTime4 = (millis() + 420000L);
    Cayenne.virtualWrite(V15, 1);
    OnValve4();
  }

  else
  {
    expireTime4 = 0;
  }
```

```
}


CAYENNE_IN(V18) // 10 minutes valve 4

{

 TenMinTimer4 = getValue.asInt();


 if (TenMinTimer4 == 1)

 {

  expireTime4 = (millis() + 600000L);

  Cayenne.virtualWrite(V15, 1);

  OnValve4();

 }


 else

 {

  expireTime4 = 0;

 }

}
```

```
CAYENNE_OUT(V21) // Time remaining valve 4

{

 unsigned long TimeRemain4 = ((expireTime4 - time4) / 1000) / 60;

 if (FiveMinTimer4 == 1 || SevenMinTimer4 == 1 || TenMinTimer4 == 1)

 {

  Cayenne.virtualWrite(V21, TimeRemain4);



  if (TimeRemain4 == 0)

  {

   Cayenne.virtualWrite(V15, 0);

   Cayenne.virtualWrite(V16, 0);

   Cayenne.virtualWrite(V17, 0);

   Cayenne.virtualWrite(V18, 0);

  }

 }

 else

 {
```

```
  Cayenne.virtualWrite(V21, 0);

 }



}



// Coding valve 4 end
```

Arduino Uno (Transmitter Circuit)

```
#include <SoftwareSerial.h>



SoftwareSerial HC12(2, 3); //RX, TX



int DataReady, digit , D, C, B, A;



const int ledPin = 13;

void setup() {

  Serial.begin(9600);

  HC12.begin(9600);
```

```
  pinMode(12, INPUT);

  pinMode(11, INPUT);

  pinMode(10, INPUT);

  pinMode(9, INPUT);

  pinMode(8, INPUT);

  pinMode(ledPin, OUTPUT);




}

void loop() {

  // read the state of

  DataReady = digitalRead(12);



  if (DataReady == HIGH) {

    D = digitalRead(11);

    C = digitalRead(10);

    B = digitalRead(9);
```

```
 A = digitalRead(8);


 digit = D * 8 + C * 4 + B * 2 + A; // akan dapat nilai 0 hingga 7

 Serial.println(digit);


 if (digit == 1)

 {

  digitalWrite(ledPin, HIGH);

  HC12.print("VALVE1ON");             // Send Command to HC-12

  delay(500);

 }


 if (digit == 2)

 {

 digitalWrite(ledPin, LOW);

 HC12.print("VALVE1OFF");            // Send Command to HC-12

 delay(500);

}
```

```
if (digit == 3)
{
digitalWrite(ledPin, HIGH);
HC12.print("VALVE2ON");            // Send Command to HC-12
delay(500);
}
if (digit == 4)
{
digitalWrite(ledPin, LOW);
HC12.print("VALVE2OFF");            // Send Command to HC-12
delay(500);
}
 if (digit == 5)
{
digitalWrite(ledPin, LOW);
HC12.print("VALVE3ON");            // Send Command to HC-12
delay(500);
```

```
}
  if (digit == 6)
 {
 digitalWrite(ledPin, LOW);
 HC12.print("VALVE3OFF");            // Send Command to HC-12
 delay(500);
}
  if (digit == 7)
 {
 digitalWrite(ledPin, LOW);
 HC12.print("VALVE4ON");            // Send Command to HC-12
 delay(500);
}
  if (digit == 8)
 {
 digitalWrite(ledPin, LOW);
 HC12.print("VALVE4OFF");            // Send Command to HC-12
 delay(500);
```

```
  }

  delay(500);

}

}
```

Arduino Uno (Receiver Circuit)

```
#include <SoftwareSerial.h>

#include <DS3231.h>



SoftwareSerial mySerial(10, 11); //RX, TX



DS3231  rtc(SDA, SCL);

Time t;



const int OnHour = 8; //SET TIME TO ON RELAY (24 HOUR FORMAT)

const int OnMin = 00;

const int OffHour = 8; //SET TIME TO OFF RELAY

const int OffMin = 10;
```

```
const int OnHour2 = 5; //SET TIME TO ON RELAY (24 HOUR FORMAT)

const int OnMin2 = 0;

const int OffHour2 = 05; //SET TIME TO OFF RELAY

const int OffMin2 = 10;


void setup() {

  Serial.begin(9600);

  rtc.begin();

  mySerial.begin(9600);




 pinMode(4, OUTPUT);

 digitalWrite(4, LOW);

 pinMode(5, OUTPUT);

 digitalWrite(5, LOW);

 pinMode(6, OUTPUT);

 digitalWrite(6, LOW);
```

```
 pinMode(7, OUTPUT);

 digitalWrite(7, LOW);

}



void loop() {



  t = rtc.getTime();

  Serial.print(t.hour);

  Serial.print(" hour(s), ");

  Serial.print(t.min);

  Serial.print(" minute(s)");

  Serial.println(" ");

  delay (1000);



  if(mySerial.available() > 1){ //Read from HC-12 and send to serial monitor

    String input = mySerial.readString();

    Serial.println(input);
```

```
if (input=="VALVE1ON")

{digitalWrite(4, LOW);

delay(500);}


 if (input=="VALVE1OFF")

{digitalWrite(4, HIGH);

delay(500);}


  if (input=="VALVE2ON")

{digitalWrite(5, LOW);

delay(500);}


  if (input=="VALVE2OFF")

{digitalWrite(5, HIGH);

delay(500);}


  if (input=="VALVE3ON")

{digitalWrite(6, LOW);
```

```
delay(500);}



  if (input=="VALVE3OFF")

{digitalWrite(6, HIGH);

delay(500);}



 if (input=="VALVE4ON")

{digitalWrite(7, LOW);

delay(500);}



 if (input=="VALVE4OFF")

{digitalWrite(7, HIGH);

delay(500);}

}

delay(20);



if(t.hour == OnHour && t.min == OnMin){

  digitalWrite(4,LOW);
```

```
  }

  else if(t.hour == OffHour && t.min == OffMin){
    digitalWrite(4,HIGH);
  }

    if(t.hour == OnHour2 && t.min == OnMin2){
  digitalWrite(4,LOW);
  }

  else if(t.hour == OffHour2 && t.min == OffMin2){
    digitalWrite(4,HIGH);
  }
}
```

Arduino Wi-Fi Rev 2 (Soil Moisture Circuit)

```
//#define CAYENNE_DEBUG

#define CAYENNE_PRINT Serial
```

```
#include <CayenneMQTTWiFi.h>



#include <SPI.h>

#include <WiFiNINA.h>

#include <SPI.h>

#include <SD.h>



#define Pin A0

#define Pin A1

#define Pin A2

#define Pin A3



File myFile;





char ssid[] = "Ayep";

char wifiPassword[] = "Huaweitestok";
```

```
char username[] = "5874db20-0ba7-11ec-8da3-474359af83d7";

char password[] = "ebef037c3d62799163bc59dbb390388904a37638";

char clientID[] = "a512ba10-4db5-11ec-ad90-75ec5e25c7a4";



void setup()

{

 Serial.begin(9600);

 Cayenne.begin(username, password, clientID, ssid, wifiPassword);



  if (!SD.begin(4)) {

  Serial.println("initialization failed!");

  while (1);

 }

 myFile = SD.open("Reading.txt", FILE_WRITE);



 // if the file opened okay, write to it:

 if (myFile) {

  Serial.print("Writing to Reading.txt...");
```

```
    myFile.println();

    // close the file:

    myFile.close();

    Serial.println("done.");

  } else {

    // if the file didn't open, print an error:

    Serial.println("error opening Reading.txt");

  }

}



void loop(){

   Cayenne.loop();



     // Wait a few seconds between measurements.

  delay(2000);

  myFile = SD.open("Reading.txt", FILE_WRITE);



   int x1 =0;
```

```
int percent1 = 0;

x1 = analogRead(A0);
percent1 = map(x1, 1023, 200 , 0, 100);

int x2 =0;
int percent2 = 0;

x2 = analogRead(A1);
percent1 = map(x2, 1023, 200 , 0, 100);

int x3 =0;
int percent3 = 0;

x3 = analogRead(A2);
percent3 = map(x3, 1023, 200 , 0, 100);

int x4 =0;
```

```
int percent4 = 0;

x4 = analogRead(A3);

percent4 = map(x4, 1023, 200 , 0, 100);

Serial.print("Soil Moisture 1 : ");

Serial.println(percent1);

Serial.print("Soil Moisture 2 : ");

Serial.println(percent2);

Serial.print("Soil Moisture 3 : ");

Serial.println(percent3);

Serial.print("Soil Moisture 4 : ");

Serial.println(percent4);

// if the file opened okay, write to it:

if (myFile) {

myFile.print("Soil Moisture 1 : ");
```

```
myFile.print(percent1);

myFile.print("\r");

myFile.print("Soil Moisture 2 : ");

myFile.print(percent2);

myFile.print("\r");

myFile.print("Soil Moisture 3 : ");

myFile.print(percent3);

myFile.print("\r");

myFile.print("Soil Moisture 4 : ");

myFile.print(percent4);

myFile.print("\r");

 // close the file:

 myFile.close();



} else {

 // if the file didn't open, print an error:

 Serial.println("error opening Reading.txt");

}
```

```
}

CAYENNE_OUT(V0)
 {
  int x1 =0;
  int percent1 = 0;

  x1 = analogRead(A0);
  percent1 = map(x1, 1023, 200 , 0, 100);

 Cayenne.virtualWrite(V0,percent1);
 }
 CAYENNE_OUT(V1)
 {
  int x2 =0;
  int percent2 = 0;

  x2 = analogRead(A1);
```

```
 percent2 = map(x2, 1023, 200 , 0, 100);



Cayenne.virtualWrite(V1,percent2);

}

CAYENNE_OUT(V2)

{

 int x3 =0;

 int percent3 = 0;



 x3 = analogRead(A2);

 percent3 = map(x3, 1023, 200 , 0, 100);



Cayenne.virtualWrite(V2,percent3);

}

CAYENNE_OUT(V3)

{

 int x4 =0;

 int percent4 = 0;
```

```
  x4 = analogRead(A3);

  percent4 = map(x4, 1023, 200 , 0, 100);



 Cayenne.virtualWrite(V3,percent4);

}
```

RTC Module Coding

```
#include <DS3231.h>



// Init the DS3231 using the hardware interface

DS3231  rtc(SDA, SCL);



void setup()

{

  // Setup Serial connection

  Serial.begin(115200);

  // Uncomment the next line if you are using an Arduino Leonardo
```

```
  //while (!Serial) {}



  // Initialize the rtc object

  rtc.begin();



  // The following lines can be uncommented to set the date and time

  rtc.setDOW(WEDNESDAY);     // Set Day-of-Week

  rtc.setTime(12, 0, 0);     // Set the time to 12:00:00 (24hr format)

  rtc.setDate(1, 1, 2022);   // Set the date

}



void loop()

{

  // Send Day-of-Week

  Serial.print(rtc.getDOWStr());

  Serial.print(" ");



  // Send date
```

```
  Serial.print(rtc.getDateStr());

  Serial.print(" -- ");



  // Send time

  Serial.println(rtc.getTimeStr());



  // Wait one second before repeating :)

  delay (1000);

}
```

Appendix B: Project Progress